IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 18 March 2024, accepted 8 April 2024, date of publication 15 April 2024, date of current version 22 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3388490

RESEARCH ARTICLE

Enhanced Modulation Recognition Through Deep Transfer Learning in Hybrid Graph Convolutional Networks

NOPPARUJ SUETRONG[1], ATTAPHONGSE TAPARUGSSANAGORN[2], AND NATTHANAN PROMSUK[1]

[1]Department of Computer Engineering, Faculty of Engineering, Chiang Mai University, Chiang Mai 50200, Thailand
[2]School of Engineering and Technology, Department of ICT, Asian Institute of Technology, Pathum Thani 12120, Thailand

Corresponding authors: Attaphongse Taparugssanagorn (attaphongset@ait.asia) and Natthanan Promsuk (natthanan.p@cmu.ac.th)

This work was supported in part by the Office of the Permanent Secretary, Ministry of Higher Education, Science, Research and Innovation under Grant RGNS 64-061; and in part by the Faculty of Engineering, Chiang Mai University, Chiang Mai, Thailand, and the Teaching Assistant and Research Assistant (TA/RA) Scholarship from Chiang Mai University.

ABSTRACT Nowadays, wireless communication plays a pivotal role in our daily lives, encompassing technologies such as wireless fidelity (Wi-Fi) and the internet of things (IoT). The backbone of the wireless communication is modulation, which involves various techniques with its own unique characteristics. As modulation techniques evolve in intricacy and diversity, the need for modulation recognition becomes apparent. Traditional modulation recognition relies on human intervention to classify modulation types in received signals, a time-consuming and laborious process prone to human error and inefficiency. Consequently, automatic modulation recognition (AMR) is introduced to autonomously classify modulation types without human interventions. In the current era, artificial intelligence (AI), specifically deep learning (DL) has gained prominence, providing numerous advantages across various domains, including AMR. While many DL-based AMR models have been developed, their efficacy reduces at low signal-to-noise ratio (SNR). Consequently, we propose a hybrid DL model for AMR, named the in-phase and quadrature - temporal graph convolutional network (IQ-TGCN) to enhance the recognition performance at low SNR. Integrating graph convolutional network (GCN) and long short-term memory (LSTM) architectures, the IQ-TGCN takes a node feature matrix as input, derived from the magnitude differences between each node. In comparative assessments against other DL models, our model has consistently exhibited superior performance. To enhance its capabilities further, we integrated deep transfer learning, leading to a remarkable 30% improvement in classification accuracy. Notably, at a SNR of 10 dB, IQ-TGCN reached its pinnacle, attaining an impressive accuracy of 99%, all the while significantly reducing training time by nearly threefold.

INDEX TERMS Automatic modulation recognition, deep learning, deep transfer learning, graph convolutional network, long short-term memory, wireless communication.

I. INTRODUCTION

In the realm of telecommunications, wireless communication emerges as a specialized domain facilitating the transmission of data without the constraints of physical constraints. This mode of communication relies on electromagnetic waves, specifically radio waves, to establish connections between two or more points. A tangible embodiment of wireless communication is wireless fidelity (Wi-Fi), which utilizes radio waves in the 2.4 gigahertz (GHz) and 5 GHz bands to seamlessly transmit data among devices such as computers, smartphones, tablets, and various wireless-enabled gadgets. In the contemporary technological landscape, the internet of things (IoT) serves as a compelling example of the practical

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

applications of wireless communication. The IoT forms a network of interconnected physical devices that effortlessly communicate and exchange data over the internet [1]. This cutting-edge technology has pervaded various aspects of our daily lives, showcasing its extensive applicability across diverse domains such as smart homes, intelligent vehicles, smart cities, and interconnected systems [2]. The significance of wireless communication lies in its ability to provide flexible and efficient connectivity, allowing devices to communicate without physical constraints. Wi-Fi, operating within specific frequency bands, facilitates seamless data transmission, fostering connectivity in our interconnected world. Furthermore, the IoT exemplifies the transformative power of wireless communication, creating a network where devices collaborate to enhance efficiency and convenience in myriad aspects of modern living.

Modulation stands as a fundamental principle in wireless communication, playing a pivotal role in encoding information onto the carrier signal—a high-frequency wave. This encoding process is crucial for ensuring the effective transmission of information through the communication channel. Beyond its foundational role, modulation also plays a critical function in enabling the simultaneous existence of multiple signals within the same frequency spectrum, thus preventing interference. As a result, a range of diverse modulation techniques has been developed, each with its unique characteristics. These modulation types serve specific purposes, addressing distinct communication needs within the intricate domain of wireless communication.

In the realm of wireless communication, automatic modulation recognition (AMR) stands out as a burgeoning standard that seamlessly integrates various technologies. This innovative approach is engineered to autonomously identify and categorize the modulation type of a received signal, eliminating the need for human intervention. Its significance becomes pronounced in scenarios where the characteristics of the communication channel are subject to changes. Traditional AMR methodologies primarily fall into two categories: likelihood theory-based AMR (LB-AMR) and feature-based AMR (FB-AMR) [3]. LB-AMR relies on likelihood theory, employing a statistical approach to calculate the probability of observed data given different modulation hypotheses. In contrast, FB-AMR focuses on extracting specific features from the received signal, utilizing this information for modulation classification [4].

Each methodology has its distinct advantages and limitations. LB-AMR, rooted in statistical principles, provides robustness, particularly in scenarios with well-defined statistical models. However, it may face challenges in dynamic environments where the underlying statistical assumptions might not hold, potentially leading to reduced accuracy. On the other hand, FB-AMR offers adaptability to diverse signal characteristics, making it suitable for dynamic environments. Nevertheless, its effectiveness is contingent on the careful selection of relevant features, as suboptimal choices may result in classification errors [5].

In the contemporary era, artificial intelligence (AI), notably deep learning (DL), stands as a transformative force offering numerous advantages across diverse fields [6]. DL, a subfield within the broader domain of machine learning (ML), focuses specifically on developing and training artificial neural networks (ANNs), particularly deep neural networks (DNNs). Distinguished by their deep architectures with multiple layers, DL models excel in learning hierarchical representations of data. This layered approach significantly enhances model performance, allowing for the capture of intricate patterns and features at various levels of abstraction.

However, the utilization of DL in AMR comes with its own set of pros and cons. On the positive side, DL-based AMR leverages sophisticated neural network architectures to discern and categorize modulation types with a high degree of accuracy. The power of DL algorithms enables the automatic extraction of intricate patterns and features from received signals, contributing to precise modulation type identification. Despite these advantages, challenges may arise in terms of computational complexity and the need for substantial amounts of labeled training data [7]. Additionally, DL models may be perceived as black-box systems, making it challenging to interpret their decision-making processes. The integration of DL into AMR demonstrates its potential to achieve accurate modulation type identification by automatically extracting intricate patterns and features. While offering significant benefits, it is crucial to address challenges such as computational complexity and the interpretability of DL models in this context.

While increasing the number of layers or neurons in DL models is a conventional strategy for performance enhancement, it comes with inherent drawbacks. The augmentation of layers or neurons escalates model complexity, necessitating greater computational resources. To address this limitation, transfer learning is introduced. Transfer learning is a technique involving the utilization of a pre-trained model, typically on a substantial dataset or a specific task, as a foundational point for a related task [8]. As previously mentioned, the efficacy of DL models depends on the availability of a substantial amount of labeled data for training. The presence of labeled data assumes a pivotal role in influencing model performance. Specifically, in the context of modulation recognition tasks, the challenge lies in acquiring labeled data that span a diverse array of modulation types while accommodating identical signal variability. This challenge is pronounced in the context of achieving robust generalization under real-world conditions.

Transfer learning offers several advantages. Firstly, it expedites the training process by leveraging the knowledge acquired from the pre-trained model, serving as a valuable starting point. By initializing a model with insights from a pre-trained counterpart, transfer learning enables the model to build upon existing knowledge and adapt more efficiently to the intricacies of the specific task at hand. This not only accelerates training but also enhances the model's ability to

generalize to new data, even when faced with limited labeled samples [9].

However, transfer learning is not without its challenges. On the positive side, it addresses the computational demands associated with increased model complexity. On the flip side, transfer learning may encounter issues when the source and target tasks significantly differ, leading to suboptimal performance. Furthermore, the selection of an appropriate pre-trained model and ensuring its compatibility with the target task are critical considerations.

A. RELATED WORKS

Over the past few years, numerous research groups have contributed DL models for modulation recognition tasks. Liu et al. [10] presented two convolutional neural networks (CNNs). The first CNN model comprised two convolutional layers and two fully connected (dense) layers, while the other featured four convolutional layers and two dense layers. Both models demonstrated the ability to learn from complex time domain vectors, particularly in the in-phase and quadrature (I/Q) data within the RadioML2016.10b dataset. In a distinct approach, Rajendran et al. [11] proposed a long short-term memory (LSTM) model with two LSTM layers and a dense layer. This novel data-driven model aimed at automatic modulation classification (AMC) utilized time domain amplitude and phase (A/P) transformed from the I/Q data sourced from RadioML2016.10a. These contributions underscore the diversity in DL model architectures tailored for modulation recognition, each leveraging specific design choices to optimize performance for distinct datasets and tasks.

Emam et al. [12] undertook fine-tuning of the convolutional long short-term deep neural network (CLDNN) originally proposed by Liu et al. [10]. Their refined model incorporated three convolutional layers, one LSTM layer, and two dense layers. Meanwhile, Ghasemzadeh et al. [13] introduced a graph convolutional network (GCN)-based classifier for AMR. Utilizing I/Q data from the RadioML2018.01a dataset as input, their GCN model transformed the I/Q data into a constellation diagram, represented as an adjacency matrix. Subsequently, the model extracted features from this matrix, employing four dense layers for modulation classification. Furthermore, Feng et al. [14] presented a feature correlation graph convolutional network (FCGCN) for graph structure mapping and individual identification. Their graph structure mapping mapped a signal to a graph structure, represented in an adjacency matrix by extracting features and then evaluating the correlation between the features, while GCN was used for identification. These research endeavors showcase the evolution of DL models in modulation recognition, with Emam et al. refining an existing architecture and Ghasemzadeh et al. and Feng et al. introducing a novel GCN-based approach, both tailored for improved performance in their respective datasets.

In addition, Zhang and Zhu [15] introduced a transfer learning approach for CNN-based AMC. They transformed one-dimensional (1D) time series signals into a two-dimensional (2D) matrix, serving as the input for the CNN model. Employing a layer freezing strategy for transfer learning, they selectively re-trained specific layers, including the first convolutional layer and the first dense layer, to obtain more appropriate features. The results demonstrated that this approach reduced the number of parameters while achieving accuracy close to training the entire network.

Furthermore, the enhancement of model performance extends beyond developing new DL architectures to encompass data preparation for training. Suetrong et al. [16] illustrated the significance of data pre-processing techniques, particularly data normalization, in improving model performance by 30.6%, without compromising the underlying structure. This highlights the holistic approach required for advancing DL-based AMC, considering both architecture refinement and meticulous data preparation techniques to achieve optimal results.

B. CONTRIBUTION

To the best of our knowledge, the proposed in-phase and quadrature - temporal graph convolutional network (IQ-TGCN) model represents a novel contribution to the field of DL-based AMR. The integration of GCN and LSTM in a hybrid architecture is a unique approach, aiming to synergize the strengths of these two neural network models. This novel combination seeks to enhance the model's ability to capture both graph-structured relationships and sequential dependencies within the data, presenting a distinctive solution in the landscape of AMR methodologies.

Furthermore, the inclusion of data normalization techniques and the strategic implementation of transfer learning add novel dimensions to our approach. The meticulous consideration of data pre-processing through normalization ensures the stability and convergence of the training process, contributing to improved model performance. The application of deep transfer learning, leveraging insights gained from one dataset to fine-tune the model for a related dataset, enhances the adaptability and generalization capabilities of the IQ-TGCN model.

The IQ-TGCN model introduces a novel synthesis of GCN and LSTM architectures, accompanied by innovative data normalization and deep transfer learning strategies. This comprehensive and unique approach positions our model as a novel and promising contribution to the advancement of DL-based AMR methodologies.

II. METHODOLOGY

This section provides a comprehensive overview of the research methodology, encompassing the exploration of various DL models and the introduction of the innovative "IQ-TGCN models." The rationale behind this methodology lies in its effectiveness in addressing the research objectives.

The research workflow commences with the extraction of I/Q data from the dataset. This initial step is crucial as it forms the foundation for subsequent analyses. The rationale

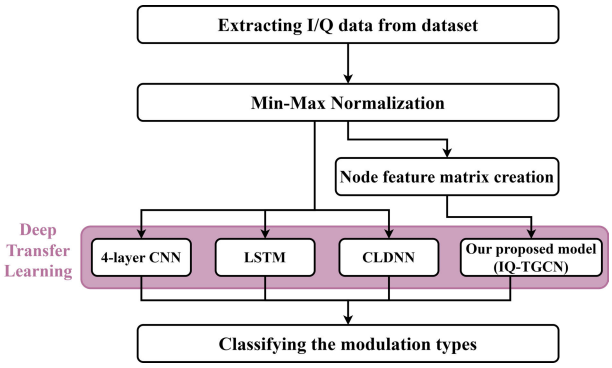


FIGURE 1. Workflow of this research.

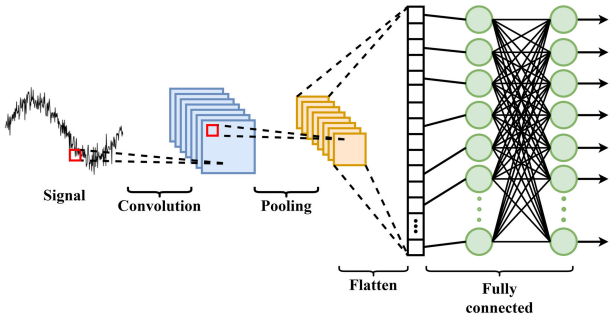


FIGURE 2. Architecture of CNN.

for focusing on I/Q data extraction is rooted in its significance for modulation type classification.

Subsequently, the application of pre-processing techniques becomes essential. Specifically, data representation transformation and normalization are employed to enhance the quality and consistency of the dataset. The rationale behind these techniques lies in their ability to mitigate potential variations and distortions in the raw data, ensuring a more robust input for the DL models.

Following the pre-processing step, the I/Q data is utilized as inputs for a range of DL models. These models include CNN, LSTM, CLDNN, and our proposed IQ-TGCN models. The rationale for this diverse model selection is rooted in the intention to comprehensively explore and compare their effectiveness in classifying modulation types.

To visually capture the overall flow of the research methodology, Fig. 1 illustrates the sequential steps from data extraction to the application of DL models. This visualization aids in providing a clear understanding of the entire process, reinforcing the transparency and reproducibility of the research approach.

A. PRE-PROCESSING TECHNIQUES

The received signal ($r(t)$) in wireless communication systems is mathematically expressed as

$$r(t) = s(t) * h(t) + n(t), \tag{1}$$

where t denotes the time period, $s(t)$ represents the transmitted signal, $h(t)$ characterizes the channel impulse response, $*$ denotes a convolutional sum, and $n(t)$ represents the noise. The $r(t)$ is commonly represented in the I/Q data format. In this study, Min-Max normalization is separately applied to in-phase and quadrature data, a technique known as feature scaling. This process standardizes each feature within a predefined range, ensuring uniform scales and preventing the dominance of specific features. Beyond enhancing overall robustness, Min-Max normalization accelerates optimization techniques, like gradient descent, by expediting convergence [17]. This streamlined approach contributes to the efficiency of DL models, aligning with the study's goal of optimizing dataset preparation for unbiased and swift analyses. The normalized data (x') is determined as

$$x' = \frac{(x - min)}{max - min}(max' - min') + min', \tag{2}$$

where x represents the original data, min signifies the minimum value, max signifies the maximum value, max' represents the new maximum scale, and min' represents the new minimum scale. For the context of this research, the values assigned to max' and min' are 1 and -1, respectively.

B. DEEP LEARNING (DL)

1) CONVOLUTIONAL NEURAL NETWORK (CNN)

CNNs, tailored for grid-structured data like images, feature pivotal layers such as convolutional and pooling layers. Convolutional layers deploy kernels to slide over input data, extracting localized patterns through element-wise operations. This process generates feature maps, capturing spatial information and highlighting relevant patterns. Multiple filters enhance feature diversity, yielding multiple feature maps.

Pooling layers, integral to the network, play a crucial role in downsampling, reducing spatial dimensions, and enhancing feature invariance to scale and orientation changes. Max pooling involves selecting the maximum value within a pooling window, while average pooling calculates the average value in the window [18]. These operations contribute significantly to downsampling, optimizing CNNs for effective pattern recognition.

The holistic structure of CNNs comprises a convolutional layer and an optional pooling layer, succeeded by dense layers, inclusive of the output layer as depicted in Fig. 2. Preceding the entry into the dense layers, a critical flattening process is applied to the high-dimensional feature maps, converting them into a 1D vector [19]. This flattening procedure serves to empower the model in discerning intricate relationships within the data, enhancing its capacity to grasp complex patterns and make informed predictions.

The training process of CNNs incorporates backpropagation [20], wherein optimization algorithms fine-tune weights to minimize a predetermined loss function. For classification tasks, an activation function, typically softmax, is utilized in the output layer. The proficiency of CNNs in capturing spatial hierarchies and local patterns positions them as a crucial

tool in diverse computer vision applications, including image classification and object detection [21], [22], [23].

2) LONG SHORT-TERM MEMORY (LSTM)

LSTM, a specialized form of recurrent neural networks (RNNs), falls within the category of ANNs tailored for handling sequential or time series data. RNNs excel in capturing temporal dependencies through the maintenance of a hidden state that evolves over time. Nevertheless, traditional RNNs encounter difficulties in learning long-term dependencies, often succumbing to challenges like vanishing or exploding gradients [24]. To overcome these limitations, LSTMs were introduced, integrating advanced gating mechanisms and a cell state mechanism [25].

The fundamental innovation within LSTMs revolves around their cell state and gating mechanism, featuring three pivotal gates: the forget gate, the input gate, and the output gate, as illustrated in Fig. 3. The cell state mechanism empowers the network to sustain a long-term memory by selectively updating and propagating pertinent information.

The forget gate plays a crucial role in determining the information retention or discarding from the previous cell state ($c^{(t-1)}$). It takes both the current input ($X^{(t)}$) and the previous hidden state ($h^{(t-1)}$) as input, processing them through a sigmoid (σ) activation function to generate a forget factor ($f^{(t)}$), expressed as

$$f^{(t)} = \sigma(W_{f_x} X^{(t)} + W_{f_h} h^{(t-1)} + b_f), \tag{3}$$

where W_{f_x}, W_{f_h}, and b_f denote the weights for the $X^{(t)}$, the weights for the $h^{(t-1)}$, and the bias for the forget gate, respectively.

The input gate plays a pivotal role in determining the new information to be incorporated into the cell state ($c^{(t)}$). Similar to the forget gate, it takes both $X^{(t)}$ and $h^{(t-1)}$ as input. This input undergoes processing through the σ activation function to generate an update factor ($i^{(t)}$). Simultaneously, the same input is passed through a hyperbolic tangent ($tanh$) activation function to produce a new candidate state ($\tilde{c}^{(t)}$), expressed as

$$i^{(t)} = \sigma(W_{i_x} X^{(t)} + W_{i_h} h^{(t-1)} + b_i), \tag{4}$$

$$\tilde{c}^{(t)} = tanh(W_{\tilde{c}_x} X^{(t)} + W_{\tilde{c}_h} h^{(t-1)} + b_{\tilde{c}}). \tag{5}$$

In these equations, W_{i_x} and W_{i_h} represent the weights associated with the $X^{(t)}$ and the $h^{(t-1)}$ for the input gate, respectively. The bias for the input gate is denoted by b_i, while $W_{\tilde{c}_x}$, $W_{\tilde{c}_h}$, and $b_{\tilde{c}}$ correspond to the weights for the $X^{(t)}$, the $h^{(t-1)}$, and the bias for the $\tilde{c}^{(t)}$, respectively.

Before reaching the output gate, an update occurs in $c^{(t)}$, integrating preserved information from $c^{(t-1)}$ with $f^{(t)}$ and $\tilde{c}^{(t)}$ with $i^{(t)}$, as expressed by

$$c^{(t)} = f^{(t)} \odot c^{(t-1)} + i^{(t)} \odot \tilde{c}^{(t)}, \tag{6}$$

where $\odot$ denotes an element-wise product.

The output gate plays a pivotal role in determining the information from the $c^{(t)}$ to be utilized in generating the output at the current time step. It processes the $X^{(t)}$ and the $h^{(t-1)}$ through the σ activation function to generate an output factor ($o^{(t)}$). Simultaneously, it applies the $c^{(t)}$ to the $tanh$ activation function. The output is obtained as the element-wise product of the $o^{(t)}$ and the $tanh$ of the $c^{(t)}$, expressed as

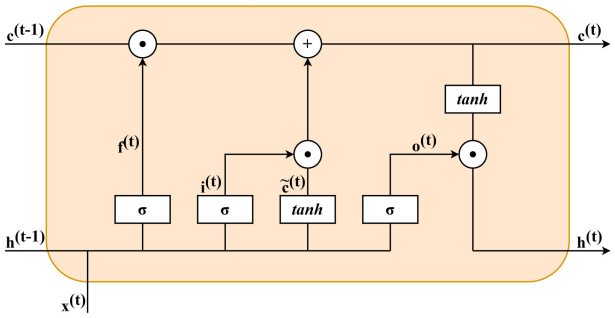


FIGURE 3. Architecture of LSTM cell.

$$o^{(t)} = \sigma(W_{o_x} X^{(t)} + W_{o_h} h^{(t-1)} + b_o), \tag{7}$$

$$h^{(t)} = o^{(t)} \odot tanh(c^{(t)}), \tag{8}$$

where W_{o_x}, W_{o_h}, and b_o represent the weights for the $X^{(t)}$, the weights for the $h^{(t-1)}$, and the bias for the output gate, respectively.

3) CONVOLUTIONAL LONG-SHORT TERM DEEP NEURAL NETWORK (CLDNN)

CLDNN is a hybrid architecture that seamlessly integrates CNN, LSTM, and DNN structures [26]. It strategically leverages the strengths of each component to optimize its performance. CNN is employed to capture local features and spatial dependencies within the input data. Multiple convolutional layers, equipped with diverse filter sizes, collaboratively contribute to reducing frequency variation by capturing features at different scales. This approach enhances the model's robustness to minor variations in the input.

The inclusion of LSTM in the CLDNN architecture is instrumental for capturing long-term dependencies in sequential or time series data [27]. LSTM's proficiency in learning evolving patterns over time makes it particularly well-suited for tasks involving temporal sequences. It adds a crucial layer of sophistication to the model's ability to comprehend and interpret evolving patterns within the input data.

The final integration is achieved through the DNN component, which harmoniously combines the spatial dependencies learned by CNN and the temporal dependencies captured by LSTM. This synthesis occurs through a series of dense layers, effectively mapping the input to the desired output İn essence, CLDNN emerges as a potent hybrid architecture that harnesses CNN for spatial feature learning, LSTM for temporal dependency modeling, and DNN for the seamless integration of these representations. This comprehensive approach empowers CLDNN to tackle diverse tasks with a nuanced understanding of both spatial and temporal aspects of the input data.

4) GRAPH CONVOLUTIONAL NETWORK (GCN)

GCN is a subset of graph neural networks (GNNs), a specialized class designed to process and analyze data organized in graph structures. Graphs, in this context, are composed of interconnected nodes and edges, where nodes represent entities, and edges signify relationships or connections between them. GCN, as an integral part of GNNs, applies convolutional operations on graphs [28], typically represented through an adjacency matrix or a node feature matrix [29]. In a manner reminiscent of traditional CNN, these convolutional operations extract features based on the local neighborhood of each node. Moreover, to manage the complexity of the graph representation, pooling operations can be employed to reduce dimensionality.

In essence, GCNs serve as tailored neural networks for graph-structured data, facilitating effective learning of node and graph-level representations. By extending the foundational principles of traditional CNNs to graph structures, GCNs emerge as valuable tools in domains dealing with interconnected entities. Examples include applications in social network analysis, link prediction, and node classification [30], [31]. Their adaptability to capture intricate relationships within graph data makes GCNs instrumental in addressing challenges related to various fields where entities are interconnected and relationships play a pivotal role.

5) DEEP TRANSFER LEARNING

Deep transfer learning represents an innovative application of transfer learning techniques within the framework of DNNs. As previously discussed, transfer learning entails the utilization of knowledge acquired from one task and its application to another, irrespective of their contextual relationship [32]. Within the realm of deep transfer learning, a pre-trained DNN model serves as the foundational framework for a new task. Typically, the pre-trained model has undergone training on a sizable dataset for a specific task, with its lower layers functioning as adept feature extractors. Fine-tuning is often essential for the remaining layers to adapt seamlessly to the intricacies of the new task.

6) OUR PROPOSED MODEL: IN-PHASE AND QUADRATURE - TEMPORAL GRAPH CONVOLUTIONAL NETWORK (IQ-TGCN)

The proposed model, named IQ-TGCN, presents a hybrid architecture that seamlessly integrates GCN and LSTM components. This model takes a node feature matrix as input, derived by computing the magnitude of differences between each node, resulting in a symmetric matrix. Convolutional operations are subsequently applied to the node feature matrix, effectively capturing relationships within the local neighborhood. The resulting feature map undergoes further processing in the LSTM architecture, facilitating the capture of temporal dependencies. Ultimately, dense layers are incorporated to map the features to the desired output. The architectural layout of the IQ-TGCN model is depicted in Fig. 4. Notably, the IQ-TGCN model leverages deep transfer learning techniques to enhance its overall performance in this research.

TABLE 1. Datasets description.

Details	RadioML2016.10a	RadioML2016.10b
Digital Modulation Types	BPSK, QPSK, 8PSK, 16QAM, 64QAM, GFSK, CPFSK, PAM4	
Analog Modulation Types	WBFM, AM-SSB, AM-DSB	WBFM, AM-DSB
SNR Levels	-20 dB to 18 dB with 2 dB increments	
Total Samples	220,000	1,200,000
Data Format	I/Q data	
Data Dimension	2×128	

III. RESULTS

This section showcases the outcomes of our research, encompassing the datasets employed, the experimental setup, and the ensuing discussion of the results.

A. DATASETS

In this study, we employed the RadioML2016.10a and RadioML2016.10b datasets [33] as they are recognized benchmark datasets extensively utilized in signal processing and wireless communications, particularly for modulation recognition. These datasets, crafted through GNU Radio, feature open-source and synthetic I/Q data. Each signal is structured in a 2×128 format, with distinct in-phase and quadrature data. The signal-to-noise ratio (SNR) levels within these datasets range from -20 dB to 18 dB, with increments of 2 dB. To simulate a real-life environment, the datasets were collected in a complex channel environment, encompassing additive white Gaussian noise (AWGN), selective fading (Rician and Rayleigh), center frequency offset (CFO), sample rate offset (SRO), and other channel interference. [34], [35].

The RadioML2016.10a dataset encompasses eight digital and three analog modulation types, featuring binary phase shift keying (BPSK), quadrature phase shift keying (QPSK), 8-phase shift keying (8PSK), 16-quadrature amplitude modulation (16QAM), 64-quadrature amplitude modulation (64QAM), Gaussian frequency shift keying (GFSK), continuous phase frequency shift keying (CPFSK), and pulse amplitude modulation with 4 levels (PAM4) as digital modulation types. Analog modulation types encompass wideband frequency modulation (WBFM), amplitude modulation with single-sideband (AM-SSB), and amplitude modulation with double-sideband (AM-DSB). This dataset comprises a total of 220,000 I/Q data samples and serves as a foundational resource for deep transfer learning applications.

The RadioML2016.10b dataset bears similarities to the RadioML2016.10a dataset, acting as its expanded counterpart with a substantial size of 1,200,000 samples. Notably, it includes only two analog modulation types: WBFM and AM-DSB. This dataset is specifically utilized for training DL models as the foundational model. The overview of the distinctions between the two datasets is outlined in Table 1.

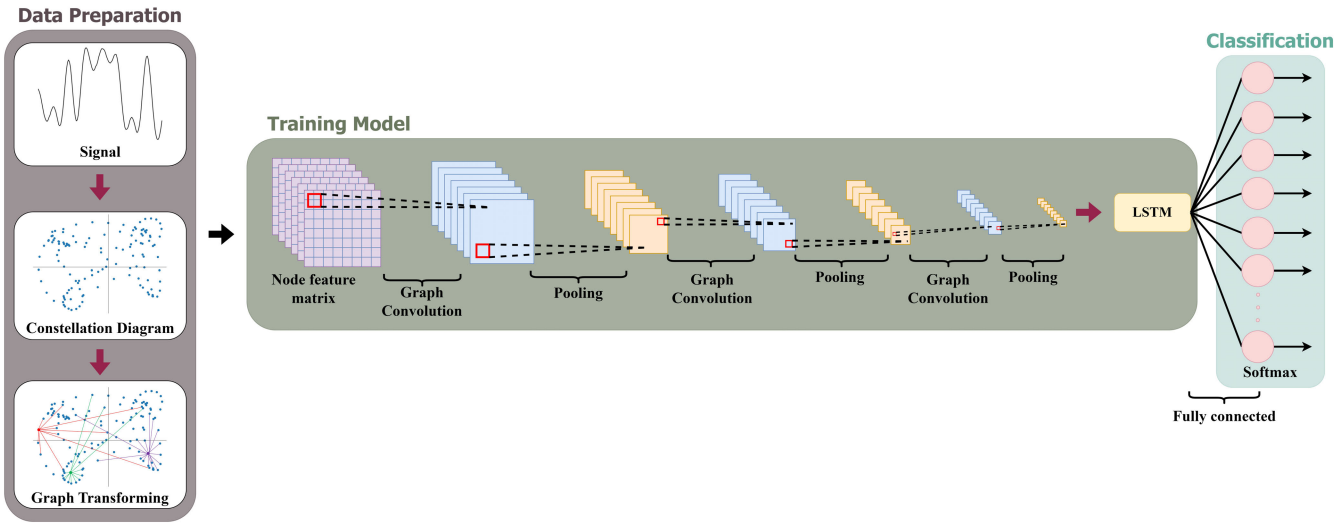


FIGURE 4. Architecture of the proposed in-phase and quadrature - temporal graph convolutional network (IQ-TGCN).

Furthermore, the proposed model and other models are examined under real-world conditions, including the effects of Doppler and interference in a complex channel.

B. EXPERIMENTAL SETUP

In the experimental setup, we bifurcated the process into two key stages: firstly, the development of the base model, and secondly, the integration of deep transfer learning into the models, as depicted in Fig. 5. To conclude, we provided a detailed explanation of the comparison models.

1) CREATING BASE MODEL

In the initial phase, our attention centered on seven digital modulation types sourced from the RadioML2016.10b dataset—namely, BPSK, QPSK, 8PSK, 64QAM, GFSK, CPFSK, and PAM4. A train-test split was executed by randomly selecting 4,200 samples for each modulation type at every SNR level. This process yielded a training set totaling 588,000 samples, with the remaining 1,800 samples per modulation type designated as the testing set for each SNR level. Notably, the dataset adhered to a 0.7:0.3 ratio between the training and testing sets.

After the data was partitioned, each sample underwent Min-Max normalization individually, as outlined in the methodology section. Since our model relies on the node feature matrix as input, we derived this matrix from the I/Q data. Specifically, the I/Q data was represented as a constellation diagram, and the disparities in magnitude between each node were determined. Consequently, a matrix of size 128 $\times$ 128 was generated for each sample. This matrix encapsulates the interrelations between a node in the constellation diagram and all other nodes. Subsequently, the models were trained using the training set and saved as a base model.

The architecture of IQ-TGCN comprises three sets of graph convolutional layers, featuring 48, 96, and 128 filters, each utilizing a 3 $\times$ 3 kernel size. Subsequent to these graph convolutional layers, a max-pooling layer with a 2 $\times$ 2 pooling size and a stride of two is applied. The architecture concludes with a leaky Rectified Linear Unit (*ReLU*) activation function. Given its nomenclature, IQ-TGCN, where 'T' denotes temporal, the subsequent layer is an LSTM layer with 256 nodes and the *tanh* activation function. Dropout layers are strategically employed after each set of graph convolutional layers and the LSTM layer. The final output utilizes a softmax activation function, with seven nodes corresponding to the number of modulation types

2) IMPLEMENTING DEEP TRANSFER LEARNING

In this part, we performed pre-processing techniques analogous to the procedures outlined in the Creating Base Model subsection. These techniques included data splitting for training and testing sets, Min-Max normalization applied to each signal separately, and the creation of the node feature matrix. However, the dataset for this part was changed to the RadioML2016.10a dataset, which is a smaller dataset. Subsequently, the base models from the previous subsection were transferred and used as an initial pre-trained model with this dataset. All layers in the base model were frozen, except for the last layer, the output layer, as the layers above it function as feature extractors. Only the last layer was trained with the training set of the RadioML2016.10a dataset.

3) OTHER DL ARCHITECTURES FOR COMPARISON PURPOSE

In this subsection, fine-tuning was conducted on the CNN from [10], the LSTM architecture from [11], and the the CLDNN architectures from [10] and [12]. The CNN model comprised 4 convolutional layers with a *ReLU* activation function. The first convolutional layer featured 256 filters with a 1 $\times$ 8 kernel size, and the second convolutional layer had 64 filters with a 2 $\times$ 8 kernel size, while the remaining layers had 64 filters with a 1 $\times$ 8 kernel size. Following the

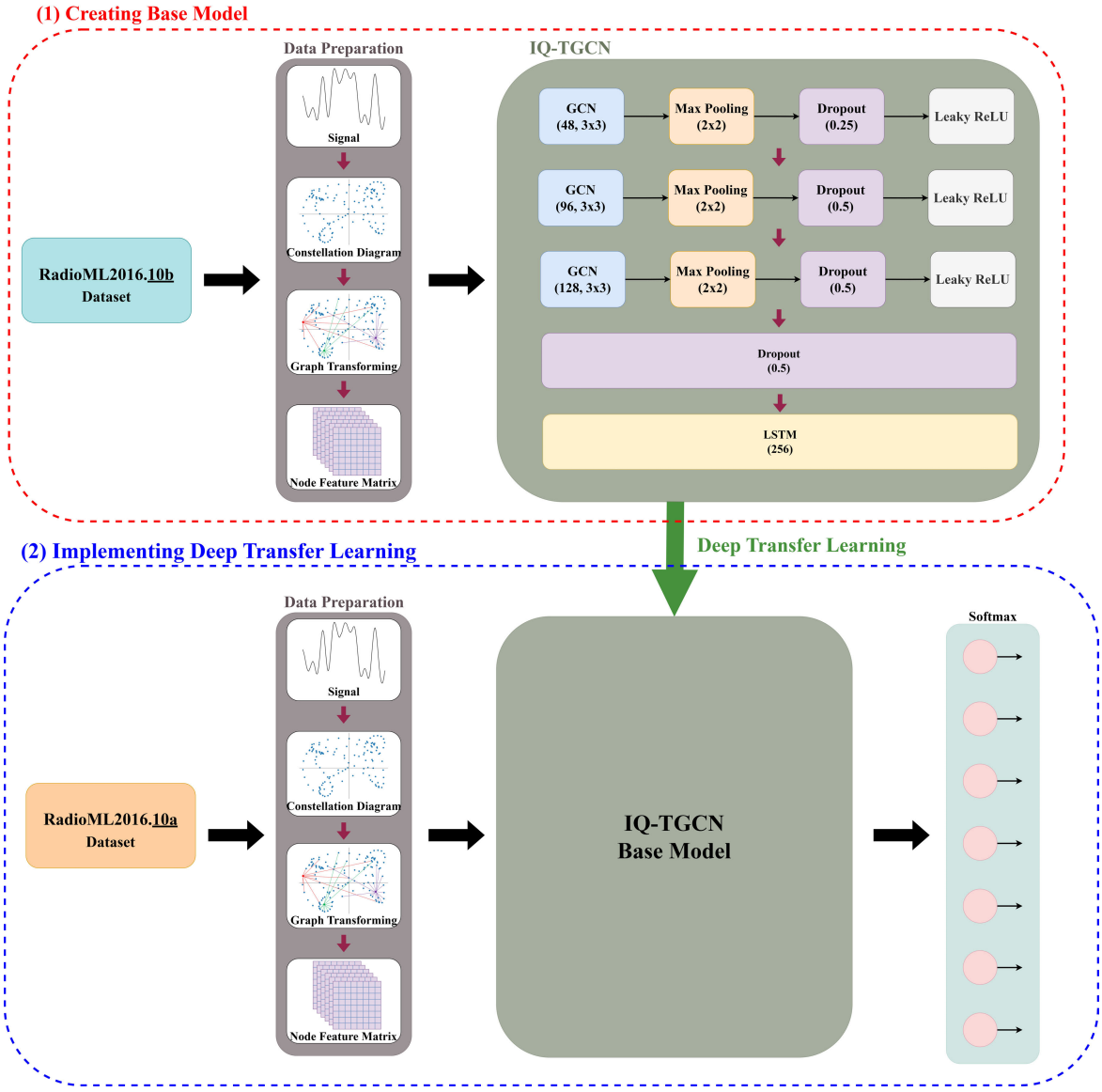


FIGURE 5. Overall experimental setup of the IQ-TGCN model with deep transfer learning.

four convolutional layers, a dense layer with 128 nodes and a *ReLU* activation function, and subsequently, the output layer were included. A dropout layer with a 0.5 rate was inserted between the convolutional layers and between the dense layer and the output layer.

The LSTM model adopted an architecture akin to [11], featuring two LSTM layers. However, there was a modification in the number of nodes, transitioning from 128 and 128 nodes to 256 and 128 nodes, respectively. Subsequently, the output layer was incorporated. To address potential overfitting, a dropout layer with a rate of 0.5 was inserted between the LSTM layers. The activation function for the LSTM layers was the *tanh* activation function.

Regarding the CLDNN model, as previously mentioned, it is a hybrid model that amalgamates the CNN, LSTM, and DNN architectures. Consequently, the first part of the model resembles the four convolutional layers of the CNN model. The second part encompasses an LSTM layer with 64 nodes and the *tanh* activation function. Lastly, a 128-node dense layer with a *ReLU* activation function was positioned after the LSTM layer, followed by the output layer.

C. RESULTS AND DISCUSSION

Initially, all models underwent training using the training set from the RadioML2016.10b dataset, followed by evaluation using the corresponding testing set. Evaluation metrics such as accuracy, precision, recall, and F1-score were employed to assess performance. The results demonstrated that IQ-TGCN achieved superior classification accuracy compared to CNN and LSTM models, slightly outperforming the CLDNN model across the -4 dB SNR to 18 dB SNR range. Refer

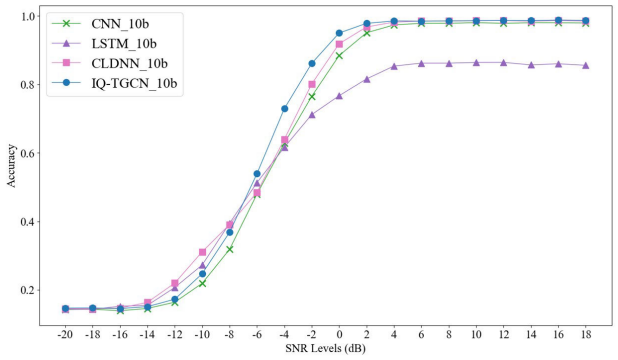


FIGURE 6. Accuracy comparison graph for various DL models across SNR levels in dB units based on the testing set of RadioML2016.10b.

TABLE 2. Classification accuracy of the IQ-TGCN model with the testing set of the RadioML2016.10b.

SNR Levels (dB)	Accuracy	Precision	Recall	F1-Score
-20	0.1424	0.0604	0.1424	0.0762
-18	0.1381	0.0595	0.1381	0.0747
-16	0.1438	0.3439	0.1438	0.0783
-14	0.1533	0.2381	0.1533	0.0911
-12	0.1900	0.2726	0.1900	0.1457
-10	0.2324	0.3139	0.2324	0.1846
-8	0.2757	0.3533	0.2757	0.2258
-6	0.3452	0.3518	0.3452	0.2852
-4	0.4348	0.4573	0.4348	0.3766
-2	0.5786	0.6430	0.5786	0.5634
0	0.7129	0.7591	0.7129	0.7114
2	0.8043	0.8327	0.8043	0.7982
4	0.8733	0.8915	0.8733	0.8680
6	0.8829	0.8972	0.8829	0.8777
8	0.8824	0.8965	0.8824	0.8778
10	0.9057	0.9196	0.9057	0.9020
12	0.9048	0.9169	0.9048	0.9031
14	0.8938	0.9089	0.8938	0.8911
16	0.8957	0.9111	0.8957	0.8931
18	0.9014	0.9128	0.9014	0.8985

to Fig. 6 and Table 2 for a visual representation and detailed breakdown.

The reduced classification accuracy of the LSTM model can be attributed to its challenge in handling input data represented as a series of pairs of values (I/Q data), which lack sequential patterns. This leads to a phenomenon known as the curse of dimensionality, which issues with high-dimensional data. While LSTM excels with sequential or time-series data, its performance diminishes when confronted with high-dimensional input. In contrast, the higher accuracy of the IQ-TGCN model can be attributed to the innovative use of the node feature matrix derived from the constellation diagram. This matrix captures inherent patterns in signal modulation types, including features such as constellation shape and distribution. These distinctive features enhance the model's robustness in distinguishing among different modulation types.

Subsequently, deep transfer learning was executed utilizing the RadioML2016.10a dataset, as previously detailed. The results unequivocally showcased that the application of deep

TABLE 3. Classification accuracy of the transferred IQ-TGCN model with the testing set of the RadioML2016.10a.

SNR Levels (dB)	Accuracy	Precision	Recall	F1-Score
-20	0.1414	0.1032	0.1414	0.0792
-18	0.1390	0.1324	0.1391	0.0734
-16	0.1514	0.1754	0.1514	0.0890
-14	0.1757	0.2401	0.1757	0.1272
-12	0.2033	0.2665	0.2033	0.1657
-10	0.2581	0.3418	0.2581	0.2240
-8	0.3781	0.3943	0.3781	0.3486
-6	0.5091	0.4934	0.5091	0.4679
-4	0.7176	0.7100	0.7176	0.7011
-2	0.8514	0.8476	0.8514	0.8480
0	0.9476	0.9477	0.9476	0.9471
2	0.9733	0.9736	0.9733	0.9733
4	0.9871	0.9873	0.9871	0.9872
6	0.9886	0.9887	0.9886	0.9886
8	0.9810	0.9811	0.9810	0.9810
10	0.9900	0.9900	0.9900	0.9900
12	0.9862	0.9864	0.9862	0.9862
14	0.9871	0.9873	0.9871	0.9872
16	0.9876	0.9878	0.9876	0.9876
18	0.9881	0.9883	0.9881	0.9881

transfer learning can significantly enhance the performance of DL models, with IQ-TGCN exhibiting notable improvements in terms of both model complexity and classification accuracy. Regarding model complexity, a substantial reduction in trainable parameters—from 548,807 to 1,799—was observed. This reduction led to a remarkable decrease in the average training time per epoch, dropping from approximately 32 minutes to around 9 minutes, representing a reduction of around 73%. In the case of the CNN, LSTM, and CLDNN models, the training time after applying deep transfer learning was reduced by about 58%, 50%, and 62%, respectively.

An additional contributing factor to this enhancement lies in the transfer of knowledge from the larger dataset, RadioML2016.10b (used as the base model), to the target model—typically a smaller dataset, such as RadioML2016.10a in this case. The accuracy surged from approximately 43% to nearly 73% at −4 dB SNR, marking an increase of about 30%, as shown in Table 3. It is worth noting that the adequacy of data plays a pivotal role, as evident from the dip in classification accuracy when dealing with smaller datasets, as compared to Figs. 6 and 7, and in comparison to Fig. 7 with type 10a and tfl. In Fig. 7, CNN_10a, LSTM_10a, CLDNN_10a, and IQ-TGCN_10a represent models trained and tested with the RadioML2016.10a dataset, while CNN_TFL, LSTM_TFL, CLDNN_TFL, and IQ-TGCN_TFL are models that incorporate deep transfer learning.

Furthermore, the classification performance of the CLDNN model on the RadioML2016.10a dataset did not significantly surpass that of the CNN model, as illustrated in Fig. 7. This can be attributed to the limitations of the LSTM model within the CLDNN architecture, as it struggled to effectively capture the relationships within I/Q data, ultimately impacting the overall performance of the CLDNN model. In contrast, the superior accuracy achieved by the

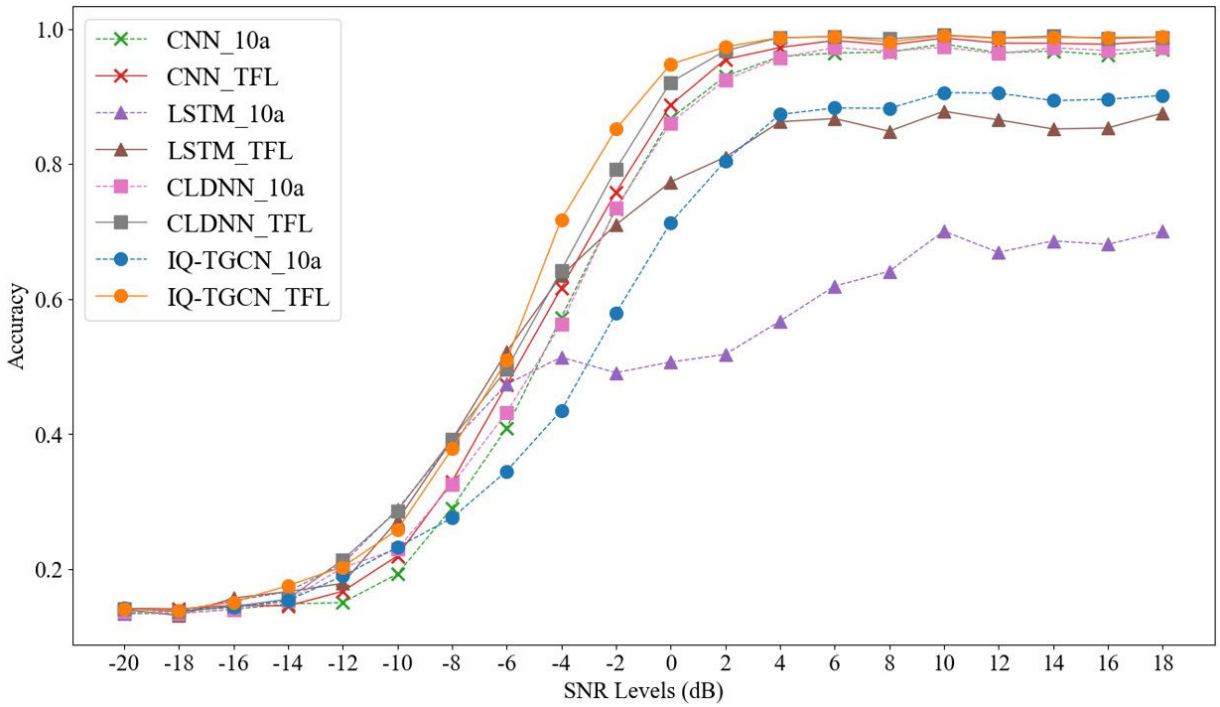


FIGURE 7. Accuracy comparison graph for various DL models across SNR levels in dB units. '10a' denotes training and testing with RadioML2016.10a, while 'TFL' denotes the implementation of deep transfer learning on RadioML2016.10a.

IQ-TGCN model, which is the combination of GCN and LSTM architectures, can be attributed to the node feature matrix. This matrix is constructed based on the magnitude of differences between each node, where each node represents a distinct time step that the LSTM can effectively handle.

Finally, the confusion matrices of the transferred IQ-TGCN model, employed for classifying seven digital modulation types, are depicted in Fig. 8. The SNR levels corresponding to −10 dB, −6 dB, 0 dB, 6 dB, 10 dB, and 18 dB are represented by (a) to (f), respectively. These matrices offer insights into the model's performance, demonstrating improvement as the SNR levels increase. Higher SNR levels indicate a lower ratio of noise. Remarkably, at 0 dB SNR, the model demonstrated an impressive accuracy of approximately 94%, as detailed in Table 3. The model showcased accurate classification for the seven modulation types, surpassing 90% accuracy for each type, with the exception of 8PSK. In Fig. 8(c), it can be observed that 8PSK was occasionally confused with QPSK. This misclassification can be attributed to the inherent similarity between the two modulation types. They are both phase-shifted keying with four and eight possible phase shifts, and some of which overlap. Particularly in the presence of noise, distinguishing between them becomes more challenging.

Examining Figs. 8(d) to 8(f), corresponding to SNR levels greater than 0 dB, the transferred IQ-TGCN model exhibited an exceptional classification accuracy of around 98% for each modulation type. This aligns with the accuracy values presented in Table 3 for SNR levels greater than 0 dB. Notably, the model showcased outstanding performance in accurately classifying each modulation type under these conditions. Conversely, for SNR levels less than 0 dB, as illustrated in Figs. 8(a) to 8(c), the classification accuracy of all DL models, including the transferred IQ-TGCN model, experienced a drop. This reduction can be attributed to the increased uncertainty in the signal under low SNR conditions. Despite the challenges posed by lower SNR levels, the transferred IQ-TGCN model demonstrated improved performance at −4 dB, −2 dB, and −14 dB SNR. This highlights the model's resilience and effectiveness in handling signal classification tasks even in adverse conditions, showcasing its robustness across varying levels of SNRs.

To enhance the realism of our simulations, we augmented the RadioML2016.10a dataset with both the Doppler effect and interference. The interference model we utilized, commonly employed in such scenarios, is the Poisson distributed heterogeneous interference model, known for its ability to introduce additional noise that mimics real-world conditions.

For the Doppler effect, we incorporated a classical model like Clarke's model, which describes the frequency shift due to the motion of either the source or the observer. According to Clarke's model, when the source moves towards an observer, the signal's frequency increases. Hence, we selected velocities of 0 kilometers per hour (km/hr), 100 km/hr, 250 km/hr, and 500 km/hr, which align with speeds typical in fifth-generation (5G) communication [36].

Our findings illustrated that the IQ-TGCN model exhibited consistent performance across varying velocities, as depicted

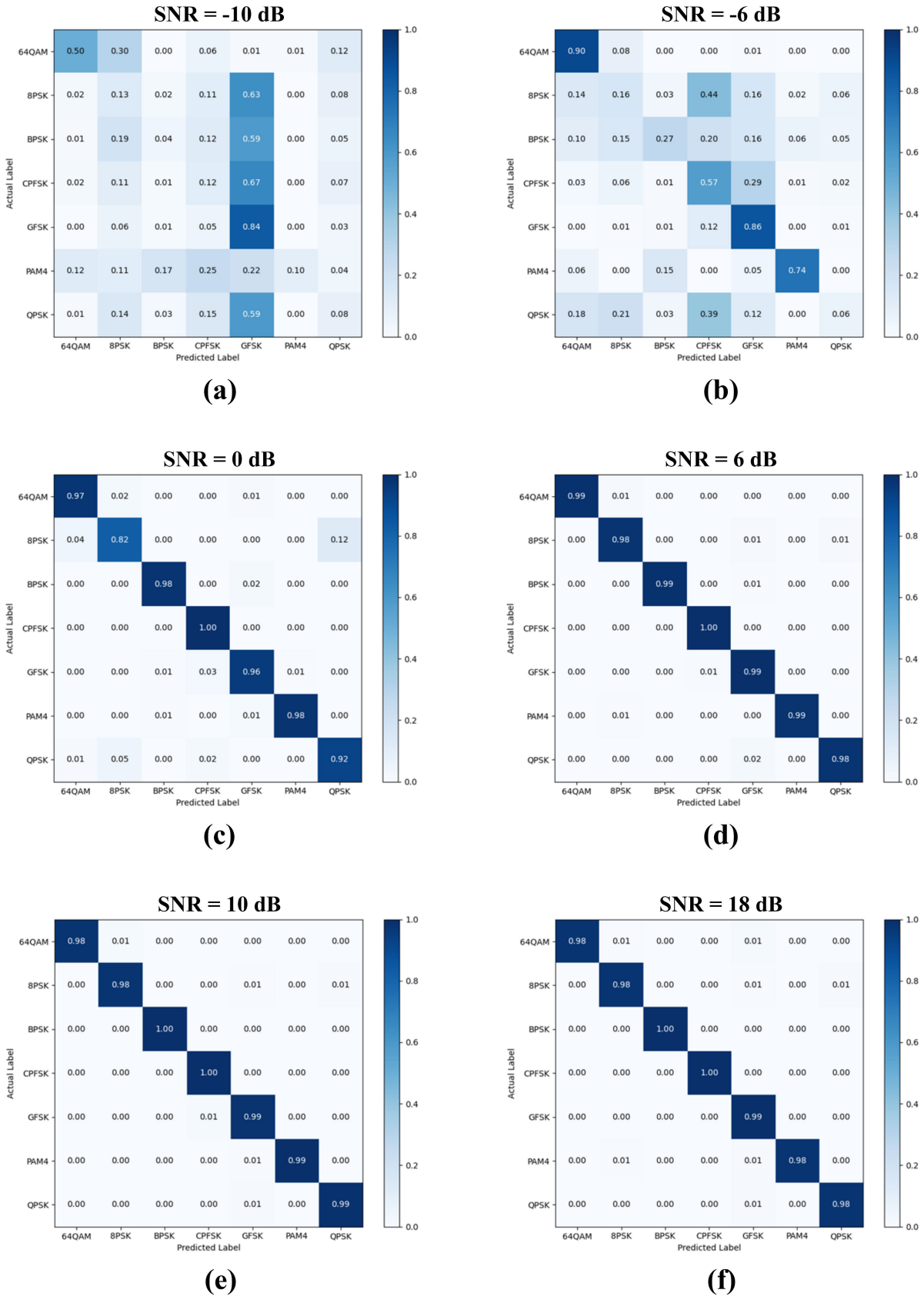


FIGURE 8. Confusion Matrix of IQ-TGCN at different SNR levels; (a) SNR = −10 dB; (b) SNR = −6 dB; (c) SNR = 0 dB; (d) SNR = 6 dB; (e) SNR = 10 dB; (f) SNR = 18 dB.

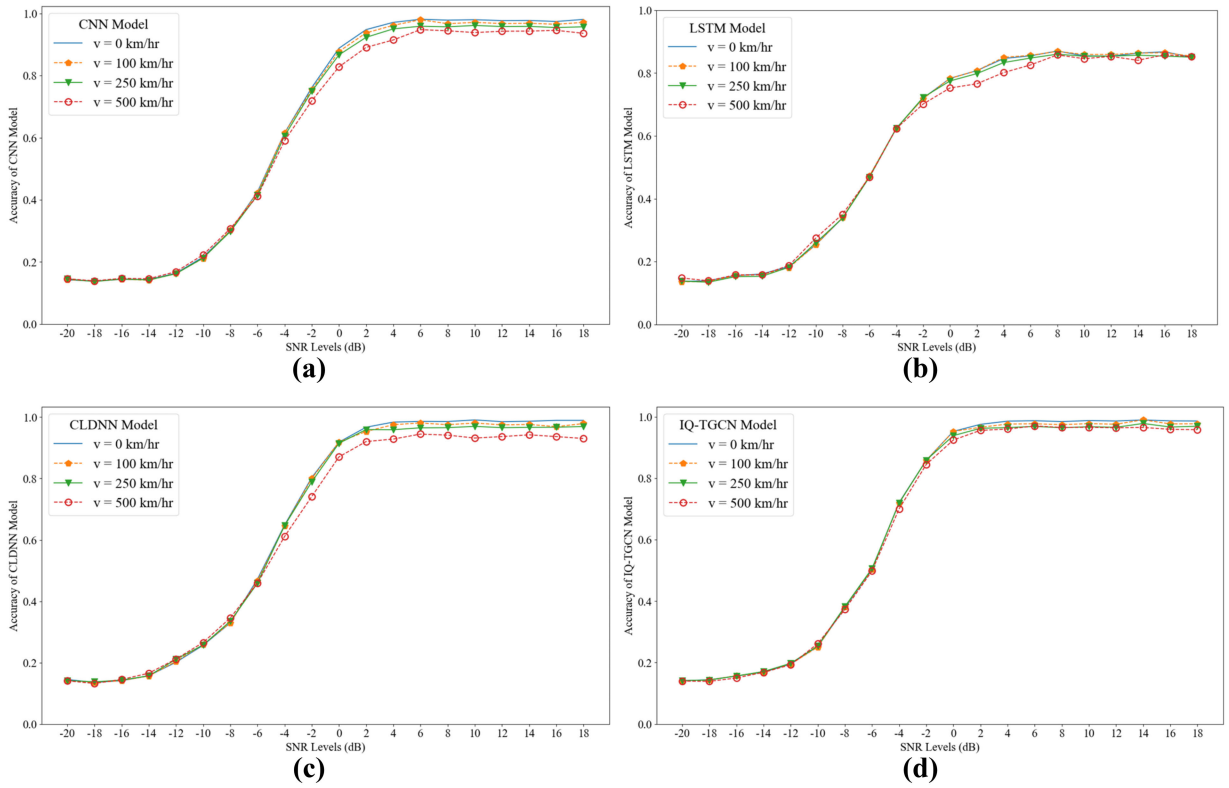


FIGURE 9. Accuracy comparison graph for various DL models with Doppler effect across SNR levels in dB units; (a) CNN model; (b) LSTM model; (c) CLDNN model; (d) IQ-TGCN model.

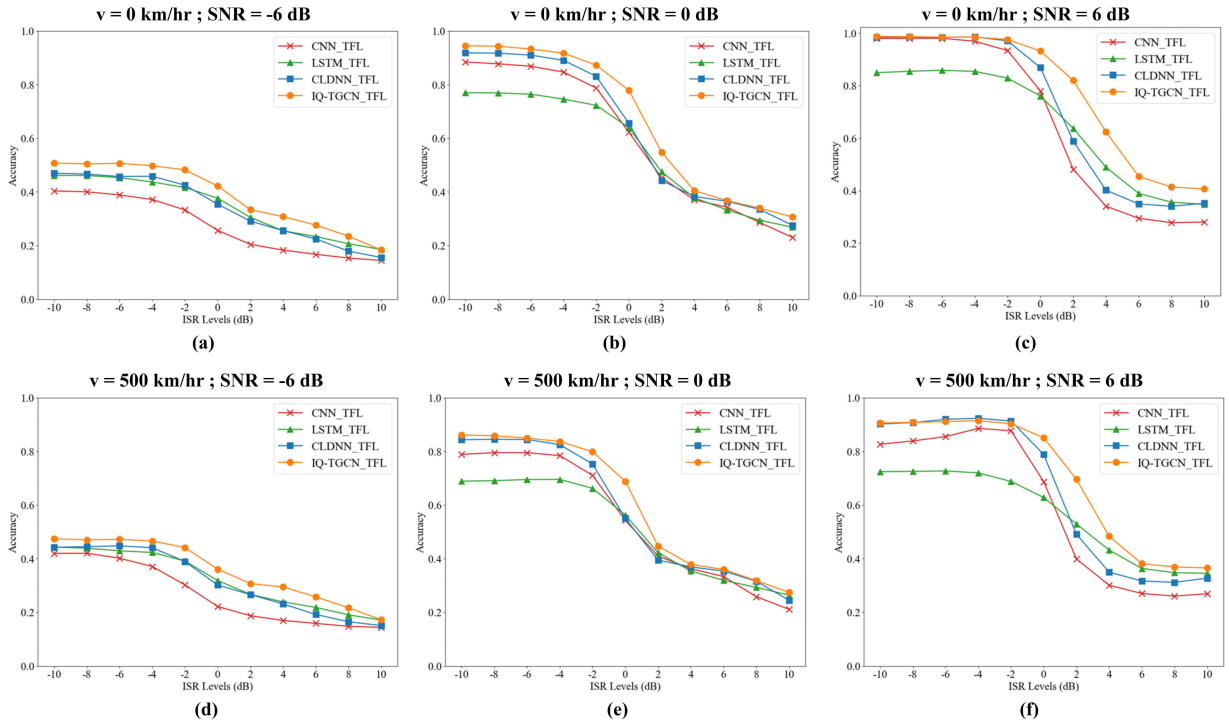


FIGURE 10. Accuracy comparison graph for DL models with Doppler effect and interference across ISR levels: (a) v=0 km/hr, SNR = −6 dB; (b) v=0 km/hr, SNR=0 dB; (c) v=0 km/hr, SNR=6 dB; (d) v=500 km/hr, SNR=−6 dB; (e) v=500 km/hr, SNR=0 dB; (f) v=500 km/hr, SNR=6 dB.

in Fig. 9(d). This resilience can be attributed to the node feature matrix, which effectively captures the relationships among local neighborhoods of each node. Conversely, the CNN, LSTM, and CLDNN models showed greater sensitivity

to Doppler-induced changes in the formation of I/Q data, as illustrated in Figs. 9(a) to 9(c), respectively.

After applying the Doppler effect to the signal, interference was introduced as an additional component. Velocities of 0 km/hr and 500 km/hr, along with SNR levels of −6 dB, 0 dB, and 6 dB, were selected before incorporating the interference into the original dataset signal. To quantify the level of incorporated interference, the interference-to-signal ratio (ISR) was chosen, ranging from −10 dB to 10 dB with 2 dB increments.

For a velocity of 0 km/hr, our findings indicate that the proposed IQ-TGCN model outperformed other models, particularly evident when the SNR levels were −6 dB and 0 dB, as depicted in Figs. 10(a) and 10(b) respectively. At an SNR of 6 dB, the proposed model exhibited similar performance to the CLDNN model when the ISR ranged from −10 dB to −2 dB, and it demonstrated higher accuracy when the ISR exceeded −2 dB, as illustrated in Fig. 10(c). For a velocity of 500 km/hr, the IQ-TGCN achieved higher performance when the SNR level was −6 dB, as shown in Fig. 10(d). However, at SNR levels of 0 dB and 6 dB, the proposed model slightly outperformed the CLDNN model when the ISR ranged from −10 dB to −2 dB, as shown in Fig. 10(e), and achieved similar performance, as shown in Fig. 10(f), respectively.

IV. CONCLUSION AND POSSIBLE FUTURE WORKS

In conclusion, the integration of wireless communication into various technologies has underscored the importance of automatic modulation recognition (AMR) for identifying received signal modulation types without human intervention. This research introduces a hybrid model, in-phase and quadrature - temporal graph convolutional network (IQ-TGCN), which leverages graph convolutional network (GCN) and long short-term memory (LSTM) architectures for AMR. Additionally, deep transfer learning is implemented to enhance the model's classification accuracy without escalating model complexity and training time.

The research methodology involves extracting in-phase and quadrature (I/Q) data from the dataset, followed by custom Min-Max normalization for each signal. The creation of the node feature matrix serves as the input for the IQ-TGCN model. Experimental results reveal that the IQ-TGCN model outperforms fine-tuned convolutional neural network (CNN), LSTM, and convolutional long-short term deep neural network (CLDNN) models with the RadioML2016.10b dataset. It achieves a peak accuracy of around 90%, particularly excelling in the −4 dB to 18 dB SNR range. The pivotal factor contributing to the model's success is its utilization of a node feature matrix derived from a constellation diagram, effectively representing patterns within the diagram and facilitating the classification of different modulation types compared to raw I/Q data.

Moreover, deep transfer learning, when applied to the RadioML2016.10a dataset, proves effective in improving the performance of DL models. The transferred IQ-TGCN model demonstrates maximum accuracy improvement of approximately 30%, accompanied by a 30% reduction in training time through a decrease in trainable parameters. The findings highlight the potential of deep transfer learning to enhance the robustness and efficiency of AMR models in real-world scenarios.

Finally, the overall performance of the proposed IQ-TGCN model declined when exposed to a more complex environment; however, its performance remained superior to that of other DNN models, including CNN, LSTM, and CLDNN models.

Possible future works to enhance this research involve exploring the application of transfer learning across diverse datasets. This could further enhance the model's adaptability in various communication scenarios. Additionally, researching the feasibility of incorporating reinforcement learning techniques for adaptive modulation recognition in dynamic and evolving communication environments could be a promising avenue for future exploration. Finally, considerations for scalability and efficiency improvements, especially for deployment in resource-constrained devices and real-time applications, should be addressed for practical implementation.

ACKNOWLEDGMENT

This work was supported by Erawan High Performance Computing (HPC) Project, Information Technology Service Center (ITSC), Chiang Mai University. The authors extended to Chiang Mai CCS for providing a dedicated server.

REFERENCES

[1] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of Things (IoT): A literature review," *J. Comput. Commun.*, vol. 3, no. 5, p. 164, 2015.

[2] S. Hamdan, M. Ayyash, and S. Almajali, "Edge-computing architectures for Internet of Things applications: A survey," *Sensors*, vol. 20, no. 22, p. 6441, Nov. 2020.

[3] F. Zhang, C. Luo, J. Xu, Y. Luo, and F.-C. Zheng, "Deep learning based automatic modulation recognition: Models, datasets, and challenges," *Digit. Signal Process.*, vol. 129, Sep. 2022, Art. no. 103650.

[4] D. H. Al-Nuaimi, I. A. Hashim, I. S. Z. Abidin, L. B. Salman, and N. A. M. Isa, "Performance of feature-based techniques for automatic digital modulation recognition and classification—A review," *Electronics*, vol. 8, no. 12, p. 1407, Nov. 2019.

[5] T. Zhang, C. Shuai, and Y. Zhou, "Deep learning for robust automatic modulation recognition method for IoT applications," *IEEE Access*, vol. 8, pp. 117689–117697, 2020.

[6] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *Proc. 4th Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, Aug. 2018, pp. 1–6.

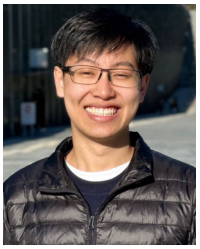
[7] B. Jdid, K. Hassan, I. Dayoub, W. H. Lim, and M. Mokayef, "Machine learning based automatic modulation recognition for wireless communications: A comprehensive survey," *IEEE Access*, vol. 9, pp. 57851–57873, 2021.

[8] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021.

[9] A. W. Salehi, S. Khan, G. Gupta, B. I. Alabduallah, A. Almjally, H. Alsolai, T. Siddiqui, and A. Mellit, "A study of CNN and transfer learning in medical imaging: Advantages, challenges, future scope," *Sustainability*, vol. 15, no. 7, p. 5930, Mar. 2023.

[10] X. Liu, D. Yang, and A. E. Gamal, "Deep neural network architectures for modulation classification," in *Proc. 51st Asilomar Conf. Signals, Syst., Comput.*, Oct. 2017, pp. 915–919.

[11] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, Sep. 2018.
[12] A. Emam, M. Shalaby, M. A. Aboelazm, H. E. A. Bakr, and H. A. A. Mansour, "A comparative study between CNN, LSTM, and CLDNN models in the context of radio modulation classification," in *Proc. 12th Int. Conf. Electr. Eng. (ICEENG)*, Jul. 2020, pp. 190–195.
[13] P. Ghasemzadeh, M. Hempel, and H. Sharif, "A robust graph convolutional neural network-based classifier for automatic modulation recognition," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, May 2022, pp. 907–912.
[14] Z. Feng, H. Zha, C. Xu, Y. He, and Y. Lin, "FCGCN: Feature correlation graph convolution network for few-shot individual identification," *IEEE Trans. Consum. Electron.*, to be published. [Online]. Available: https://ieeexplore.ieee.org/document/10272604, doi: 10.1109/TCE.2023.3322224.
[15] P. Zhang and Z. Zhu, "Transfer learning for CNN based modulation classification in time-variant a WGN channels," in *Proc. IEEE 8th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2022, pp. 1551–1555.
[16] N. Suetrong, A. Taparugssanagorn, and N. Promsuk, "Deep learning-based robust automatic modulation classification using higher order cumulant features," in *Proc. 15th Int. Conf. Inf. Technol. Electr. Eng. (ICITEE)*, Oct. 2023, pp. 1–6.
[17] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, "Normalization techniques in training DNNs: Methodology, analysis and application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 10173–10196, 2023.
[18] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022.
[19] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: An overview and application in radiology," *Insights Imag.*, vol. 9, no. 4, pp. 611–629, Aug. 2018.
[20] X. Zhou, "Understanding the convolutional neural networks with gradient descent and backpropagation," *J. Phys., Conf. Ser.*, vol. 1004, Apr. 2018, Art. no. 012028.
[21] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018.
[22] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," *Remote Sens.*, vol. 13, no. 22, p. 4712, Nov. 2021.
[23] A. Dhillon and G. K. Verma, "Convolutional neural network: A review of models, methodologies and applications to object detection," *Prog. Artif. Intell.*, vol. 9, no. 2, pp. 85–112, Jun. 2020.
[24] N. Promsuk and A. Taparugssanagorn, "Long short term memory network-based interference recognition for industrial Internet of Things," *J. Commun.*, vol. 15, no. 12, pp. 876–885, 2020.
[25] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019.
[26] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584.
[27] D. Ke, Z. Huang, X. Wang, and X. Li, "Blind detection techniques for non-cooperative communication signals based on deep learning," *IEEE Access*, vol. 7, pp. 89218–89225, 2019.
[28] F. Wu, A. H. Souza, T. Zhang, C. Fifty, T. Yu, and K. Q. Weinberger, "Simplifying graph convolutional networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 6861–6871.
[29] M. Chen, Z. Wei, Z. Huang, B. Ding, and Y. Li, "Simple and deep graph convolutional networks," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1725–1735.
[30] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: A comprehensive review," *Comput. Social Netw.*, vol. 6, no. 1, pp. 1–23, Dec. 2019.
[31] S. Xiao, S. Wang, Y. Dai, and W. Guo, "Graph neural networks in node classification: Survey and evaluation," *Mach. Vis. Appl.*, vol. 33, no. 1, pp. 1–19, Jan. 2022.
[32] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Artificial Neural Networks and Machine Learning—ICANN 2018*, Rhodes, Greece. Cham, Switzerland: Springer, 2018, pp. 270–279.
[33] T. O'Shea and N. West, "Radio machine learning dataset generation with GNU radio," in *Proc. GNU Radio Conf.*, 2016, vol. 1, no. 1. [Online]. Available: https://pubs.gnuradio.org/index.php/grcon/article/view/11
[34] W. Zhang, Y. Sun, K. Xue, and A. Yao, "Research on modulation recognition algorithm based on channel and spatial self-attention mechanism," *IEEE Access*, vol. 11, pp. 68617–68631, 2023.
[35] R. Duan, X. Li, H. Zhang, G. Yang, S. Li, P. Cheng, and Y. Li, "A multi-modal modulation recognition method with SNR segmentation based on time domain signals and constellation diagrams," *Electronics*, vol. 12, no. 14, p. 3175, Jul. 2023.
[36] P. Fan and J. Zhao, "5G high mobility wireless communications: Challenges and solutions," *China Commun.*, vol. 13, no. 2, pp. 1–13, 2016.



NOPPARUJ SUETRONG received the Bachelor of Engineering degree in information systems and network engineering from Chiang Mai University, Chiang Mai, Thailand, in 2021, where he is currently pursuing the master's degree with the Department of Computer Engineering. His research interests include wireless communication and machine and deep learning applications.

ATTAPHONGSE TAPARUGSSANAGORN received the B.Eng. degree from Chulalongkorn University, Thailand, in 1997, the M.Sc. degree in electrical engineering from Technische Universität Kaiserslautern, Germany, in 2001, and the Dr.Tech. degree from the University of Oulu, Finland, in 2007. Right after his undergraduate study, he was an Engineer with the Telecommunications Transmission Department, Siemens Ltd., Bangkok, for two years. He gained experience as a Researcher with the Institute of Communications, University of Stuttgart, after his masters study, until 2003. After that, he joined the Centre for Wireless Communications (CWC), University of Oulu, in 2003, until he received the Dr. (Tech.) degree. In 2008, he continued with CWC as a Postdoctoral Researcher and was with Yokohama National University, Yokohama, Japan, as a Visiting Postdoctoral Researcher. After he returned home to Thailand, in 2011, he joined the Asian Institute of Technology (AIT) as an Adjunct Faculty Member. In 2012, he joined the National Electronics and Computer Technology Center (NECTEC) as a full-time Employee. Since August 2015, he has been a full-time Faculty Member with AIT. He is currently an Associate Professor with the Department of ICT. His research interests include signal processing, statistical signal processing (detection and estimation techniques), wireless communications engineering, information theory, the Internet of Things, and machine/deep learning for various applications.

NATTHANAN PROMSUK received the Bachelor of Engineering degree in computer engineering from Chiang Mai University, in 2014, and the Master of Engineering and Doctor of Engineering degrees in telecommunications from the Asian Institute of Technology (AIT), Pathum Thani, Thailand, in 2017 and 2020, respectively. Currently, he is an Assistant Professor with the Department of Computer Engineering, Chiang Mai University. His research interests include digital signal processing, signal detection, interference suppression, the Internet of Things, and machine/deep learning.