ResGCN: A Scalable, Robust and Efficient Approach for Radio Fingerprinting

Fahmida Afrin, Neda Moghim, Sachin S. Shetty
Department of Modeling and Simulation
Old Dominion University

***Abstract*—Radio fingerprinting is a technique that distinguishes wireless devices by exploiting the unique hardware imperfections present in radio frequency (RF) signals. As wireless communication systems become increasingly complex and widely adopted, the need for accurate radio fingerprinting has intensified. Existing methods, such as Convolutional Neural Networks (CNNs), have been widely used for the identification of wireless devices due to their ability to enhance accuracy. However, these methods have significant drawbacks. One major limitation is their reliance on large, labeled datasets for effective training, which can be both difficult and expensive to acquire. Additionally, CNNs struggle significantly to maintain consistent performance across varying temporal conditions, particularly in cross-day scenarios where RF signal characteristics may shift due to environmental changes, device aging, or other temporal factors. This inconsistency undermines the reliability and practicality of CNNs in real-world applications. To address these limitations, this paper proposes a novel approach using Residual Graph Convolutional Networks (ResGCNs). ResGCNs are particularly adept at modeling graph-structured data, enabling them to capture intricate spatial and temporal dependencies within Inphase and Quadrature (I/Q) samples from wireless channels. Our results demonstrate that, ResGCNs not only achieve an impressive 96% accuracy in channel identification using 80% less data compared to CNNs but also achieves a notable 72.35% accuracy in cross-day scenarios, outperforming the CNNs under the same conditions. These findings underscore the potential of ResGCNs as a more efficient and scalable solution for radio fingerprinting.**

***Index Terms*—Radio Fingerprinting, ResGCN, Graph Neural Network, Convolutional Neural Networks, Wireless Devices, Internet of Things, Data Efficiency, Channel Identification, Deep Learning**

I. Introduction

The explosive growth of wireless communication technologies, driven by the widespread adoption of Internet of Things (IoT) devices, 5G networks, and other emerging innovations, has significantly increased the need for precise identification and differentiation of wireless devices. This need is critical for efficiently managing spectrum, and ensuring reliable device authentication in increasingly complex wireless environments. Cryptography-based authentication methods, such as passwords or digital certificates, face considerable challenges in this domain due to the inherent characteristics of wireless communication, including high device mobility, the broadcast nature of transmissions, the diversity of protocols, the lack of a trusted root authority, and an expanded attack surface [1].

To address these challenges, radio fingerprinting has become essential for authenticating devices by analyzing hardware-induced variations in RF signals. These variations, caused by manufacturing imperfections, create unique fingerprints that enhance network security and trustworthiness.

Previous methods for radio fingerprinting, particularly CNNs, have been extensively used due to their ability to extract complex features from raw data. However, these methods come with notable limitations:

-*Inadequate Capture of Complex Dependencies*: CNNs are limited in their ability to capture the intricate spatial and temporal dependencies present in I/Q samples of wireless signals. This limitation arises because CNNs are designed to process data on regular grids, like images, which makes them less effective in modeling the irregular and non-Euclidean nature of graph-structured data inherent in wireless signal processing [2].

-*Scalability Issues*: The performance of CNNs is heavily dependent on the availability of large labeled datasets, which may not always be feasible to obtain. Moreover, as the number of transmitters increases, the complexity of the data grows, leading to longer training times and higher computational costs. These scalability issues make it challenging to deploy CNN-based models in real-time scenarios and large-scale deployments [3].

-*High Feature Engineering Requirements*: Neural Network based radio fingerprinting approaches often require extensive manual feature engineering, relying on domain expertise to extract relevant features from the data. This process is time-consuming and can limit the adaptability of the model to different environments and conditions.

-*Limitations in Addressing Temporal Variations*: CNNs often struggle to maintain robustness against temporal variations, particularly in cross-day scenarios. For example, a neural network trained on RF signals from one day may exhibit significantly lower accuracy when tested on RF signals from the following day [4]. Various strategies have been proposed to address this issue, such as multi-day training [5], [6] data augmentation [7], and adding extra components to transmitters [8], [9]. However, each of these approaches has its limitations. Multi-day training demands large amounts of labeled data collected over multiple days, which may not always be feasible. Data augmentation can introduce data bias, leading to distribution shifts that result in suboptimal performance. Additionally, adding extra components to transmitters is impractical for devices that are already deployed and operational, making it difficult to implement this solution widely.

To address these challenges, we propose the use of Residual Graph Convolutional Networks (ResGCNs) for scalable and efficient radio fingerprinting. ResGCNs represent an advancement over Graph Convolutional Networks (GCNs), which are known for their proficiency in modeling complex data and capturing intricate dependencies across diverse domains. Unlike CNNs that operate on Euclidean data structures, GCNs are designed to handle non-Euclidean data such as graphs, making them particularly suitable for capturing the spatial and temporal dependencies present in wireless channels [10]. By incorporating residual connections, ResGCNs address the vanishing gradient problem, thereby enabling the training of deeper networks and enhancing the model's ability to learn complex feature representations [11].

Our study demonstrates that ResGCNs can achieve comparable accuracy with significantly less data compared to CNNs, making them an ideal choice for large-scale, real-time applications. This scalability benefit is crucial for practical deployment, as it reduces the need for extensive data collection and processing, thereby lowering operational costs and enhancing the feasibility of implementation in diverse settings. Furthermore, the inherent robustness of ResGCNs against temporal variations—further solidifies their potential as a leading solution for radio fingerprinting [12].

The structure of this paper is organized as follows: Section II provides a comprehensive review of related work, examining existing approaches and techniques. Section III presents the proposed method, detailing the system model, data preprocessing steps, and the architecture of the ResGCN model. Section IV covers the training and evaluation procedures, presenting a thorough analysis of the model's performance along with comparisons to existing method. Finally, Section V provides a summary of the findings, highlights the contributions of this work, and outlines potential directions for future research.

II. Related Work

The field of radio fingerprinting has evolved significantly, with the advent of various machine learning (ML) techniques, each exhibiting unique strengths and encountering specific challenges. This section provides an in-depth exploration of these methods, categorized into supervised learning, unsupervised learning, and recent advancements in graph-based models, emphasizing their application to radio fingerprinting.

***1. Supervised Learning Techniques*: ** Supervised learning has been a foundational approach in radio fingerprinting, with methodologies ranging from similarity-based to classification-based techniques.

-*Similarity-based Techniques*: These approaches involve matching observed signatures against reference signatures stored in a master database. While simple, these methods face scalability issues as they rely on predefined databases, which may not account for variations in signal characteristics caused by environmental changes.

Classification-based Techniques: Classification models, such as support vector machines (SVMs) and logistic regression, use features like I/Q imbalance and phase errors. Brik et al. [13] achieved 99% accuracy in identifying WiFi devices using transients and offset-based features in controlled environments. However, these methods are often limited to specific conditions, reducing their applicability in diverse real-world scenarios.

Deep Learning Techniques: The advent of deep learning, particularly CNNs, has revolutionized the field of radio fingerprinting. CNNs have demonstrated substantial potential due to their capability to automatically extract hierarchical features from raw data, as showcased in various modulation recognition tasks. For example, Riyaz et al. [14] demonstrated that CNNs could achieve 90-99% accuracy in transmitter identification, even in complex environments. Despite their success in controlled conditions, CNNs often struggle with overfitting and fail to generalize across varying channel conditions, as noted by Restuccia et al. [8].

***2. Unsupervised Learning Techniques* :**

-*Non-parametric Bayesian Approaches*: In scenarios where prior label information is absent, non-parametric Bayesian methods have proven to be highly effective. These approaches are capable of identifying latent patterns and clusters in data without the necessity of labeled training data. Nguyen et al. [15] employed such a method to detect the number of devices by leveraging device-dependent channel-invariant radio metrics, demonstrating its efficacy on a small ZigBee node test bed. This technique is particularly advantageous in dynamically changing environments where labeled data may not be available, though non-parametric Bayesian approaches can suffer from high computational complexity, scalability issues, sensitivity to hyperparameters, and potential challenges with interpretability and convergence.

***3. Recent Advanced Techniques*: ** The limitations of previous ML and initial deep learning approaches have inspired the exploration of more advanced models that can better capture complex dependencies. Graph Convolutional Networks (GCNs) have emerged as a promising solution for such challenges, offering significant advancements in handling graph-structured data.

-*Graph Convolutional Networks (GCNs)*: GCNs excel in modeling spatial structures and dependencies, making them ideal for tasks like spectrum sensing that require understanding intricate relationships between data points. Studies like [16] demonstrate GCNs' ability to capture relationships within I/Q samples better than CNNs, improving multi-antenna spectrum sensing. Other works, such as [17], highlight GCNs' role in enhancing spectrum sensing efficiency in dense environments and adaptive spectrum sensing in dynamic conditions [18], significantly improving cognitive radio networks. Further research [19] explores how GCNs optimize cooperative decision-making in spectrum sensing, and [20] extends GCN applications to dynamic spectrum access, leveraging historical data for better network performance.

Beyond cognitive radio, GCNs are now being explored in radio fingerprinting due to similar spatial and temporal dependencies in RF signals. Building on GCN successes in spectrum sensing, studies like [20] apply GCNs to indoor localization tasks, showing their versatility in RF domains.

Building upon the foundation laid by GCNs, we propose Residual Graph Convolutional Networks (ResGCNs) for radio fingerprinting. ResGCNs are specifically designed to capture the complex spatial and temporal dependencies within I/Q samples, addressing many of the limitations faced by CNNs. This paper introduces several key contributions:

- We introduce ResGCNs to the domain of radio fingerprinting, leveraging their ability to capture complex spatial and temporal dependencies within I/Q samples.
- By integrating residual connections into the GCN architecture, our approach mitigates the vanishing gradient problem, allowing for deeper network training and improved feature extraction.
- The proposed method demonstrates superior scalability and computational efficiency, handling large datasets effectively and reducing trainable parameters and training times significantly compared to existing CNNs models.
- The proposed ResGCN model achieve an exceptionally high accuracy of 72.35% in the cross-day scenario, highlighting their robustness in capturing complex spatial and temporal dependencies in radio signals across different temporal contexts.

III. Proposed Method

Our implementation of ResGCN-based radio fingerprinting leverages a system model that integrates graph-based learning with deep neural network paradigms. The motivation for applying graph neural networks (GNNs) to RF fingerprinting lies in their ability to model the intricate relationships inherent in wireless signals. While time-domain IQ samples are sequential data and not inherently graph-structured, representing them as a graph enables the capture of underlying dependencies and interactions among samples. Thus the core of the system model involves representing the received wireless signal data as a graph. Each node in the graph corresponds to an I/Q sample, while the edges represent the relationships between these segments. The graph convolution operation in the ResGCN framework aggregates information from neighboring nodes, enabling the extraction of complex features from the data.

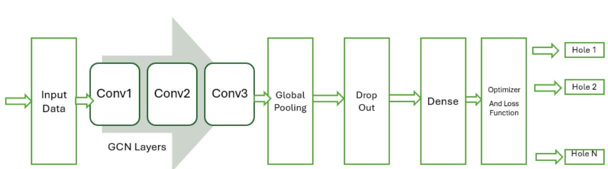

Fig. 1. System Model

Figure 1 illustrates the system model employed in this study, which consists of multiple GCN layers (Conv1, Conv2, Conv3), global pooling, and dense layers, as well as dropout for regularization.The GCN layers are crucial for capturing the spatial and temporal dependencies in the I/Q signals, enabling the model to learn intricate features from raw data that CNNs might miss. Additionally, the use of global pooling and dense layers helps in aggregating information and refining the final predictions, making the model more adaptive, robust, and scalable for real-world applications

Mathematically, the graph convolution operation can be expressed as:

$$H^{(l+1)} = \sigma\left(\hat{A}H^{(l)}W^{(l)}\right)$$

where $H^{(l)}$ represents the feature matrix at the l-th layer, with each row corresponding to a node (data point) and each column representing a feature. $\hat{A}$ is the normalized adjacency matrix, given by $\hat{A} = D^{-\frac{1}{2}}(A+I)D^{-\frac{1}{2}}$, where A is the adjacency matrix, I is the identity matrix, and D is the degree matrix. $W^{(l)}$ denotes the trainable weight matrix at the l-th layer. σ is a non-linear activation function, such as ReLU.

To enhance the model's capacity to learn deep feature representations, residual connections are incorporated, allowing for the flow of information across layers and preventing issues such as the vanishing gradient problem. The incorporation of residual connections can be expressed as:

$$H^{(l+1)} = \sigma\left(\hat{A}H^{(l)}W^{(l)}\right) + H^{(l)}$$

This design ensures that the input feature matrix $H^{(l)}$ is directly added to the output of the graph convolution operation, facilitating better gradient flow and enabling the training of deeper networks.

A. Data Preprocessing

Data Description: The dataset utilized in this study, referred to as Dataset B, originally collected from [4]. In this setup, all the four devices are connected to a single antenna, ensuring that they are equidistant from the receiver acted as transmitters while the fifth device acted as receiver. This configuration allows all devices to experience similar channel and multipath conditions, thereby minimizing variability due to differing physical positions. Data collection spanned two separate days, creating a comprehensive dataset under varied temporal conditions. The I/Q samples collected are used in two distinct evaluation scenarios: same-day and cross-day data.

In the same-day scenario, the dataset is split such that a portion of the I/Q samples from one day is used for training, while the remaining samples from the same day are used for testing. This setup allows for the evaluation of the model's performance under consistent environmental conditions.

In the cross-day scenario, I/Q samples collected on one day are used exclusively for training, and samples from the following day are used for testing. This scenario is particularly challenging as it tests the model's ability to generalize across different temporal conditions, highlighting its robustness against potential variations in channel characteristics due to changes in environmental factors. By employing both same-day and cross-day evaluation methods, the study provides a thorough assessment of the model's capacity to perform consistently under various real-world conditions.

Feature Extraction and Standardization: The initial step involves extracting relevant features from the I/Q samples, focusing on the real and imaginary components, which serve as primary indicators of the signal's characteristics. These features are standardized to a uniform scale, between 0 and 1, to ensure consistency across the dataset. Standardization is vital as it normalizes the data, ensuring that all features contribute equally to the learning process. This adjustment, which results in features having a mean of zero and a standard deviation of one, stabilizes the learning process and enhances convergence during training. While explicit noise reduction techniques are not directly employed, the standardization and careful feature selection effectively minimize the impact of irrelevant or noisy data, focusing on the most meaningful features for classification.

Graph Construction: After feature extraction, the next step is constructing a graph where each node represents an I/Q sample. Nodes are connected based on feature similarities, reflecting the relationships and dependencies among data points. The adjacency matrix, which defines these connections, has been constructed using the K-nearest neighbors (KNN) method, ensuring that nodes with similar features are closely linked. This graph structure allows the ResGCN model to capture both local and global structures within the data, offering a comprehensive understanding of the signal's characteristics.

To enhance the model's robustness and prevent overfitting, random edge flipping has also been applied, which involves randomly altering the connections in the graph's adjacency matrix with a specified probability. This technique introduces variability in the graph structure, simulating different configurations and preventing the model from becoming overly dependent on specific connections. Additionally, edges are weighted according to the similarity between nodes, using metrics such as Euclidean distance and correlation measures. The weighted adjacency matrix is then normalized to ensure numerical stability during graph convolution operations, treating each node and its neighbors equitably and preventing nodes with more connections from dominating the learning process.

B. Model Architecture

The proposed ResGCN model is composed of several key layers, each serving a distinct purpose in the processing pipeline. As depicted in Figure 2, the architecture begins with multiple GCNConv layers, where each layer applies a localized convolution operation on the graph's nodes. These layers are designed to aggregate information from a node's immediate neighborhood, effectively capturing the local structure and relationships within the graph. The inclusion of residual connections between these layers ensures that the model retains information from earlier layers, thus enhancing its ability to learn hierarchical representations. The identity mapping in these residual connections allows for direct pathways for the gradient flow, thereby stabilizing the training process and enabling the construction of deeper networks without encountering the issues associated with deep learning, such as gradient vanishing or exploding [11].

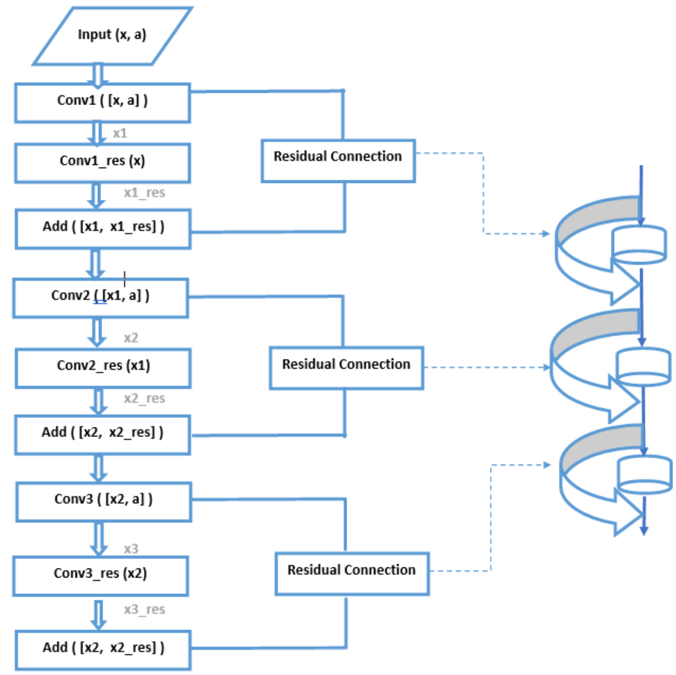

Fig. 2. ResGCN Model Architecture

Following the GCNConv layers, the architecture includes a global pooling layer, specifically designed to aggregate features across all nodes in the graph. This layer plays a crucial role in transforming the graph-level data into a fixed-size output, which is then passed to the dense layers. The dense layers, equipped with ReLU activations, further refine the learned representations and contribute to the model's classification capabilities. A dropout layer is employed before the final output layer to prevent overfitting by randomly setting a fraction of the input units to zero during training. This regularization technique ensures that the model generalizes well to unseen data, a critical requirement for real-world applications.

The next layer of the architecture is a softmax output layer, which produces a probability distribution over the possible classes, facilitating the classification task. Finally, the model utilizes the Adam optimizer, a dynamic algorithm that adjusts the learning rate for each parameter, promoting faster and more stable convergence without overshooting the minimum loss. The model's training is guided by the sparse categorical cross-entropy loss function, ideal for multi-class classification tasks with integer-labeled data. This loss function evaluates the disparity between predicted probabilities and actual labels, penalizing errors and driving the model to refine its parameters for improved classification accuracy of RF signals

IV. Training and Evaluation

The dataset was divided into training, validation, and test sets, adhering to an 80-10-10 split. The evaluation of the model involved two primary setups: Train-Test-One-Day (TTOD) and Train-One-Day-Test-Another (TDTA), also known as Cross-Day Testing. The TTOD setup involved training and testing the model on data collected from the same day, yielding an accuracy of 96%. This high accuracy reflects the model's ability to precisely learn the characteristics of RF signals under consistent conditions. On the other hand, the TDTA setup posed a more significant challenge, requiring the model to generalize across different days with varying channel conditions. In this setup, the model achieved a commendable accuracy of 72.35%, demonstrating its robustness and adaptability to unseen data.

A. Performance Analysis and Discussion

The results obtained from the proposed ResGCN model were benchmarked against two prominent studies: Al-Shawabka et al. [4] and the DeepSense SDR results by Uvaydov et al. [21]. The comparison highlights significant advancements in radio fingerprinting accuracy and the resilience of deep learning models against various wireless channel conditions.

Al-Shawabka et al. [4] investigated the impact of wireless channels on CNN-based radio fingerprinting, particularly focusing on the degradation of accuracy due to channel variations. Their findings indicated a significant drop in accuracy, with TTOD accuracy ranging from 85% in optimal conditions to as low as 8.72% in more challenging scenarios, such as for TDTA. The ResGCN model, in comparison, achieved a TTOD accuracy of 96%, demonstrating its superior capability to maintain high accuracy even when trained and tested on data collected on the same day. This distinct difference underscores the efficacy of ResGCN's architecture in capturing complex relationships in the data and mitigating the effects of channel variations.

In the TDTA setup, which involved training and testing on data collected on different days, Al-Shawabka et al. [4] reported accuracies as low as 8.72% when using equalized I/Q data. This significant drop was attributed to the CNN's inability to generalize across different channel conditions. The ResGCN model, however, managed a much higher TDTA accuracy of 72.35% using the same dataset, indicating a robust performance even in cross-day testing scenarios. This suggests that the ResGCN architecture, with its use of graph-based data representation and residual connections, provides a more stable and accurate identification process by effectively distinguishing hardware-induced features from channel effects. The graphical comparison is illustrated in Figure 3.

Another study which was performed by [21] focused on real-time spectrum sensing and radio identification using a CNN-based approach implemented on an FPGA platform. The study utilized the same dataset and achieved high precision and recall rates of 98% and 97%, respectively, demonstrating

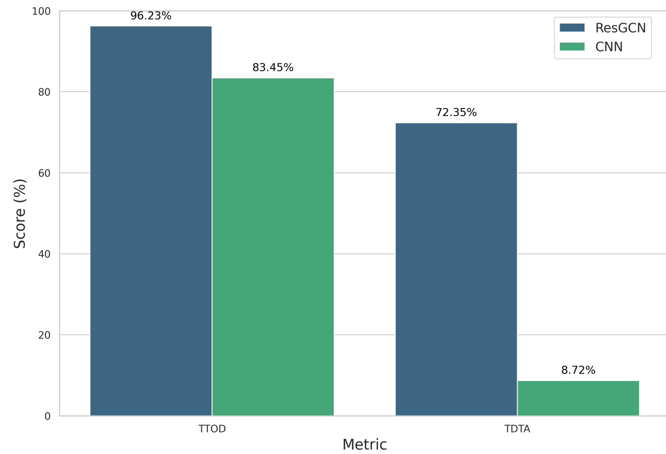

Fig. 3. Performance Comparison: TTOD vs TDTA Accuracy compared with [4]

strong capabilities in accurately identifying spectrum occupancy under diverse conditions. In contrast, the ResGCN model, leveraging the same dataset, achieved a comparable performance, with 96% accuracy and 95% recall value along with a F1 score of 97% as shown in Figure 4. Remarkably, ResGCN achieved these results while utilizing only 20% of the data compared to the CNN used in DeepSense as illustrated in Figure 5. This efficiency highlights a significant advantage of ResGCN in terms of data efficiency, making it particularly suitable for scenarios where data collection is costly or time-consuming. The model's architecture, featuring three GCN layers with residual connections, effectively reduces the number of trainable parameters and accelerates convergence. Consequently, the ResGCN model not only matches the performance metrics of CNN-based models but does so with a fraction of the data, underscoring its robustness and efficiency in learning from less data.

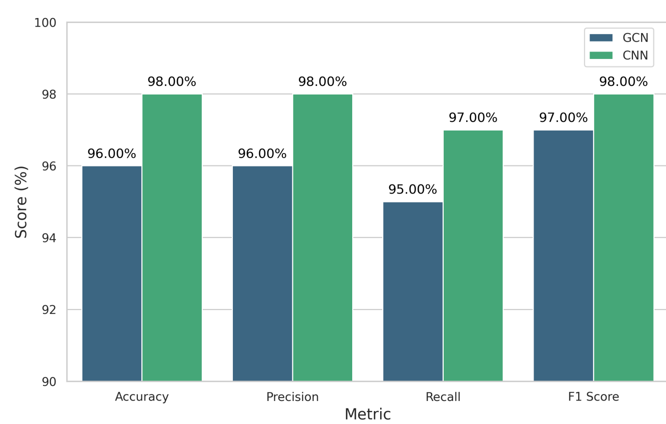

Fig. 4. Performance comparison between Graph Convolutional Networks (GCNs) and Convolutional Neural Networks (CNNs) [21] across four key metrics: accuracy, precision, recall, and F1 score.

Trainable Parameters and Hyperparameter Tuning: The ResGCN model was designed with a focus on minimizing

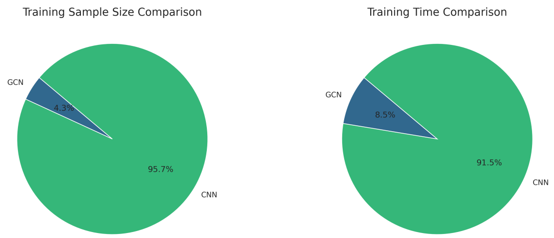

Fig. 5. Comparative analysis of ResGCN and CNN [21] performance in terms of training samples and time. The figure illustrates the superior efficiency of ResGCN, which achieves similar accuracy with much fewer training samples and very short training time compared to CNN [21]

the number of trainable parameters while maximizing the performance. Key hyperparameters in this design include k, the number of nearest neighbors used in graph construction; p, the dropout rate applied during training to mitigate overfitting; and p_flip, the probability of random edge flipping to enhance robustness. A systematic grid search was employed to optimize these hyperparameters by exploring various combinations of p, p_flip and k. Through this process, the optimal values were determined as k=8, p=0.03 and p_flip = 0.03. These specific values were found to provide the best balance between preventing overfitting and maintaining high model accuracy and generalization. The choice of k=8 allowed the model to effectively capture local node relationships without over-complicating the graph structure, which can lead to noise and overfitting. Meanwhile, a dropout rate of p=0.03 was sufficient to introduce regularization, ensuring that the model did not become overly dependent on specific neurons, thereby improving its generalization capability. The parameter p_flip, on the other hand, introduced randomness by flipping edges (removing or adding edges randomly) to further improve model generalization. This configuration, along with a learning rate of 0.0001, batch size of 128 and a dropout rate of 0.4, resulted in a highly efficient training process, both in terms of computational resources and time along with a balanced model accuracy.The hyperparameter tuning process, along with the depiction of how the model's accuracy varies with different fractions of training data, is presented in Figures 6 and 7, respectively.

Upon training with optimized values, the ResGCN model utilized only 22,184 trainable parameters to get similar results while CNN model required 72,452 trainable parameters. This efficiency highlights the effectiveness of the ResGCN architecture in capturing essential features for radio fingerprinting tasks while maintaining a lean and computationally efficient structure. The reduction in trainable parameters not only reduces the computational burden but also minimizes the risk of overfitting, making the ResGCN model a highly efficient alternative for large-scale and real-time applications

V. Conclusion and Future Work

The proposed ResGCN model sets a new benchmark for radio fingerprinting, demonstrating strong potential for real-

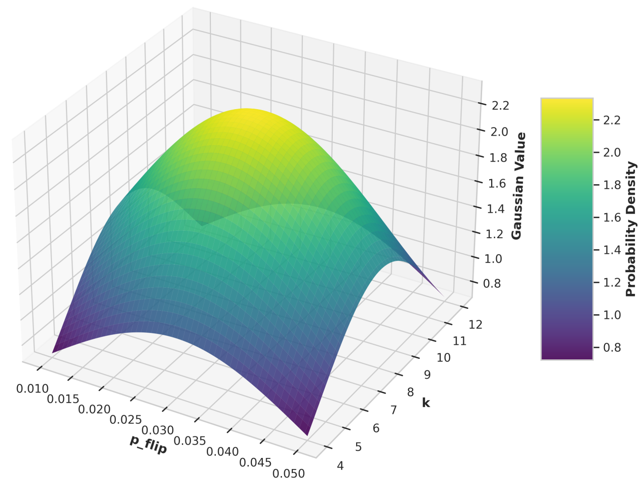

Fig. 6. Hyperparameter tuning

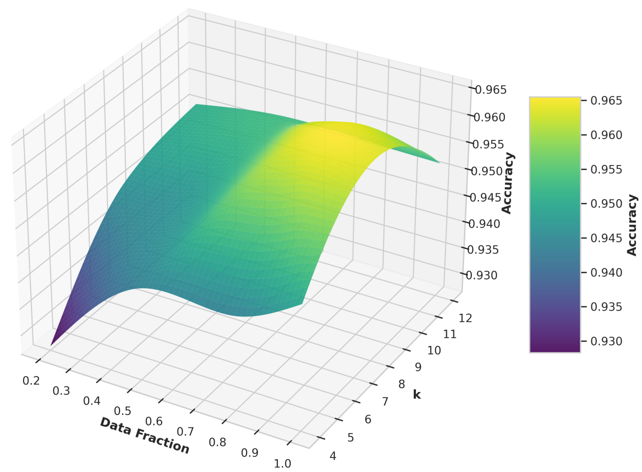

Fig. 7. Surface plot illustrating the variation in model accuracy with respect to different fractions of training data

world applications. Its ability to generalize across different conditions, coupled with a robust architecture, positions it as a valuable tool for future research and development in the field of wireless communications. The observed performance emphasizes the importance of its architectural features. The inclusion of residual connections facilitated the training of a deeper network, allowing the model to learn complex, high-dimensional representations of the data. Furthermore, the use of graph-based representations enabled the model to capture both spatial proximity and feature similarity among data points, which is crucial for accurate classification in dynamic environments.

Despite the promising results, the TDTA accuracy indicates areas for improvement, particularly in enhancing the model's generalizability across different days. Future work could explore the integration of data augmentation techniques, such as synthetic data generation or adversarial training, to improve robustness against varying conditions. Additionally, the development of more sophisticated graph-based architectures, such as Graph Attention Networks (GATs), could potentially offer further improvements in capturing temporal dependencies and reducing overfitting.

References

[1] J. Hall, M. Barbeau, and E. Kranakis, "Radio frequency fingerprinting for intrusion detection in wireless networks," *IEEE Transactions on Defendable and Secure Computing*, vol. 12, pp. 1–35, 2005.

[2] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 94–104, 2015.

[3] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks (2016)," *URL https://arxiv.org/pdf/1602.04105. pdf*.

[4] A. Al-Shawabka, F. Restuccia, S. D'Oro, T. Jian, B. C. Rendon, N. Soltani, J. Dy, S. Ioannidis, K. Chowdhury, and T. Melodia, "Exposing the fingerprint: Dissecting the impact of the wireless channel on radio fingerprinting," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 646–655.

[5] M. Cekic, S. Gopalakrishnan, and U. Madhow, "Robust wireless fingerprinting: Generalizing across space and time," *arXiv preprint arXiv:2002.10791*, 2020.

[6] I. Agadakos, N. Agadakos, J. Polakis, and M. R. Amer, "Chameleons' oblivion: Complex-valued deep neural networks for protocol-agnostic rf device fingerprinting," in *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2020, pp. 322–338.

[7] N. Soltani, K. Sankhe, J. Dy, S. Ioannidis, and K. Chowdhury, "More is better: Data augmentation for channel-resilient rf fingerprinting," *IEEE Communications Magazine*, vol. 58, no. 10, pp. 66–72, 2020.

[8] F. Restuccia, S. D'Oro, A. Al-Shawabka, M. Belgiovine, L. Angioloni, S. Ioannidis, K. Chowdhury, and T. Melodia, "Deepradioid: Real-time channel-resilient optimization of deep learning-based radio fingerprinting algorithms," in *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2019, pp. 51–60.

[9] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "Oracle: Optimized radio classification through convolutional neural networks," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 370–378.

[10] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI open*, vol. 1, pp. 57–81, 2020.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[12] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[13] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 116–127.

[14] S. Riyaz, K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep learning convolutional neural networks for radio identification," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 146–152, 2018.

[15] N. T. Nguyen, G. Zheng, Z. Han, and R. Zheng, "Device fingerprinting to enhance wireless security using nonparametric bayesian method," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 1404–1412.

[16] E. V. Vijay and K. Aparna, "Deep learning-ct based spectrum sensing for cognitive radio for proficient data transmission in wireless sensor networks," *E-Prime-Advances in Electrical Engineering, Electronics and Energy*, vol. 9, p. 100659, 2024.

[17] D. Janu, S. Kumar, and K. Singh, "A graph convolution network based adaptive cooperative spectrum sensing in cognitive radio network," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 2, pp. 2269–2279, 2022.

[18] Y. Li, G. Lu, Z. Li, and Y. Ye, "Graph neural network based cooperative spectrum sensing for cognitive radio," in *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2024, pp. 1–6.

[19] H. Jiang, *Graph Neural Network Based Learning for Dynamic Spectrum Access*. University of Rhode Island, 2021.

[20] Y. Chen, G. Li, Y. Tan, and G. Zhang, "Graph-based radio fingerprint augmentation for deep-learning-based indoor localization," *IEEE Sensors Journal*, vol. 23, no. 6, pp. 6074–6084, 2023.

[21] D. Uvaydov, S. D'Oro, F. Restuccia, and T. Melodia, "Deepsense: Fast wideband spectrum sensing through real-time in-the-loop deep learning," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.