

Dynamic Spectrum Sharing Using Deep Learning and Generative Artificial Intelligence Models

Amanda Sheron Gamage

School of Electrical and Electronic
Engineering
The Graduate School
Yonsei University

Dynamic Spectrum Sharing Using Deep
Learning and Generative Artificial Intelligence
Models

Advisor Prof. Seong-Lyun Kim

A Dissertation Submitted to the
School of Electrical and Electronic Engineering
and the Committee on Graduate School
of Yonsei University in partial fulfillment of the
requirements for the degree of
Master of Science

Amanda Sheron Gamage

June 2025

This certifies that the dissertation of Gamage Amanda Sheron
is approved.

Committee Chair: Prof. Seong Lyun Kim

Committee Member: Prof. Jeonghun Park

Committee Member: Prof. Jemin Lee

School of Electrical and Electronic Engineering

Graduate School

Yonsei University

June 2025

ACKNOWLEDGEMENTS

As I complete my master's degree, I am grateful to have been surrounded by wonderful people, which allowed me to enjoy a happy graduate school life. I would like to express my heartfelt thanks to the precious individuals who helped me finish this journey well.

First, I extend my deep gratitude to my advisor, Professor Seong Lyun Kim, for his invaluable guidance and support throughout my research. His expertise and encouragement have been instrumental in shaping my academic journey. I am also thankful to my committee members, Professor Jeonghun Park and Professor Jemin Lee, for their insightful feedback and support. I would like to express my sincere appreciation to my lab members at RAMO, for their support and friendship. Their camaraderie has made my time in the lab enjoyable and fulfilling. I will cherish all the memories I made with the incredible people I met through RAMO in my heart forever.

Next, I want to express my gratitude to the Government of the Republic of Korea and the Ministry of Education for providing me with the opportunity to study in Korea through the Global Korea Scholarship Program (GKS).

Finally, I would like to express my deepest gratitude to my family; *Thatththa, Amma, Aachchi, Nangi & Tharindu* and friends for their unwavering support and encouragement throughout my academic journey. Their love and belief in me have been a constant source of strength, and I am truly grateful for their presence in my life.

TABLE OF CONTENTS

List of Tables	iii
List of Figures	iv
Abstract	vi
Chapter 1: Introduction and Background	1
1.1 Citizens Broadband Radio Service (CBRS)	3
1.2 Background	5
Chapter 2: Methodology	8
2.1 Dataset	8
2.2 Vector Quantized Variational Autoencoder (VQ-VAE)	9
2.3 Generative Adversarial Network (GAN)	11
2.4 Denoising Diffusion Probabilistic Model (DDPM)	13
2.5 Convolutional Neural Network (CNN)	15
2.6 Deep Reinforcement Learning (DRL)	16
2.7 System Architecture	17

2.8 Experimental Setting	18
2.8.1 VQ-VAE	18
2.8.2 GAN	19
2.8.3 DDPM	19
2.8.4 CNN	20
2.8.5 Fréchet Inception Distance (FID)	20
2.8.6 Principal Component Analysis (PCA)	21
2.8.7 DRL	22
Chapter 3: Results	24
Chapter 4: Discussion	31
Chapter 5: Conclusion	34
References	40

LIST OF TABLES

2.1 DRL agent reward structure	23
3.1 Average accuracy for GAN, DDPM, VQ-VAE and Original data . .	28
3.2 Comparison of Generative Models in Terms of Mean Reward and Collision Rate	30

LIST OF FIGURES

1.1	The Citizens Broadband Radio Service (CBRS) spectrum	3
2.1	Training ViC dataset containing the 12 classes [7]	9
2.2	The VQ-VAE architecture	10
2.3	The GAN architecture	12
2.4	Spectrogram images generated by the Generator of the GAN . . .	13
2.5	Diffusion process of the DDPM	14
2.6	EMA sampled images and the change of the learning rate with each time step when training the DDPM	15
2.7	System Architecture	18
2.8	Spectrograms of the signals generated by the simulated CBRS environment in MATLAB	23
3.1	A side-by-side comparison of some of the collision scenarios generated by the DDPM, the GAN, and the VQ-VAE	24
3.2	FID Score Comparison Across Models for Different Classes. . . .	25
3.3	PCA scores for the GAN, DDPM and VQ-VAE	26

3.4 Class-wise classification accuracy for GAN, DDPM, VQ-VAE generated data with original dataset	27
3.5 Performance of DQN agents trained with the four different datasets	29

ABSTRACT

Amanda Sheron Gamage

School of Electrical and Electronic Engineering

The Graduate School, Yonsei University

Dynamic spectrum sharing is a critical component in the efficient utilization of wireless communication resources, particularly within the Citizens Broadband Radio Service (CBRS) band. This thesis presents a novel end-to-end AI-driven framework for intelligent spectrum access that integrates generative modeling, supervised classification, and deep reinforcement learning (DRL). The system is designed to operate as a learning-based alternative to traditional Spectrum Access Systems (SAS), capable of real-time spectrum occupancy detection and autonomous channel decision-making.

To address the challenge of limited labeled data for training, we employ three types of generative AI models—Generative Adversarial Networks (GAN), Vector Quantized Variational Autoencoders (VQ-VAE), and Denoising Diffusion Probabilistic Models (DDPM)—to synthesize high-fidelity spectrogram images representing various CBRS scenarios, including interference and collision cases. These synthetic spectrograms are used to train a convolutional neural network (CNN) for binary classification of collision events. Experimental evaluations demonstrate that DDPM displays the best performance in terms of both visual fidelity and classification accuracy, significantly outperforming GAN and VQ-VAE.

A MATLAB-based CBRS simulation environment is developed, where a DRL

agent, trained using a Deep Q-Network (DQN), interacts with the spectrum environment. The agent receives channel occupancy predictions from the CNN and learns to avoid collisions with Incumbent and Priority Access License (PAL) users while maximizing throughput on General Authorized Access (GAA) channels. Quantitative results show that the DQN agent trained with DDPM-generated data achieved a mean reward of 79.65 with zero collisions per episode, indicating safe and effective channel selection. In comparison, the GAN-trained agent achieved a higher reward of 86.78 but with a small number of collisions (0.74 per episode), while the VQ-VAE and original dataset agents suffered from higher collision rates (12.6 and 17.44 per episode, respectively), despite lower or comparable rewards. These findings highlight the role of generative model quality in ensuring both high performance and operational safety in spectrum access.

The thesis concludes that the integration of generative AI, deep classification, and DRL offers a scalable and intelligent solution to spectrum sharing in CBRS. The modular architecture, high-performance learning models, and simulation results provide a foundation for future extensions, including multi-channel scaling, real-time deployment, and unsupervised learning approaches for spectrum management.

Keywords: Citizens Broadband Radio Service (CBRS), Generative Adversarial Networks (GAN), Vector Quantized Variational Autoencoders (VQ-VAE), Denoising Diffusion Probabilistic Models (DDPM), Deep Q-Network (DQN), Spectrum Access System (SAS), Dynamic Spectrum Sharing, Reinforcement Learning, Convolutional Neural Network (CNN).

CHAPTER 1

INTRODUCTION AND BACKGROUND

Spectrum sharing is a critical approach in modern wireless communications that allows multiple users, systems, or services to access the same frequency bands dynamically or concurrently while minimizing interference. Traditionally, spectrum allocation followed a static model where regulatory bodies, such as the International Telecommunication Union (ITU) or Federal Communications Commission (FCC), assigned exclusive frequency bands to specific services, such as cellular networks, broadcasting, and military communications. However, with the exponential growth of wireless technologies—such as 5G, 6G, IoT, and satellite-based internet—this rigid allocation model has led to spectrum underutilization in some bands and congestion in others. Spectrum sharing mitigates this inefficiency by enabling more flexible and intelligent use of the available spectrum.

There are various models of spectrum sharing, including Dynamic Spectrum Access (DSA), unlicensed spectrum sharing and Licensed Shared Access (LSA). LSA allows secondary users to access licensed spectrum under strict regulatory agreements, while DSA employs cognitive radio technology to detect and utilize underutilized spectrum opportunistically. In unlicensed sharing, multiple users operate in shared bands with interference management protocols. Techniques such as spectrum sensing, cooperative communication,

and machine learning-based allocation further enhance the effectiveness of spectrum sharing by allowing real-time adaptation to spectrum availability.

The need for spectrum sharing is driven by increasing spectrum scarcity, growing wireless traffic, and the demand for more efficient spectrum utilization. It enables cohabitation between different services, such as military and commercial applications, or between terrestrial and satellite networks, improving overall network capacity and reliability. Additionally, spectrum sharing reduces the need for costly spectrum auctions and facilitates innovation by allowing emerging technologies to access frequency bands that would otherwise remain unused. As wireless ecosystems evolve, spectrum sharing will play a pivotal role in ensuring sustainable and scalable connectivity while balancing regulatory, technical, and economic considerations.

Generative AI models have revolutionized machine learning by enabling the synthesis of high-quality data across various domains, including audio, image and text generation and speech synthesis. Unlike traditional discriminative models that learn decision boundaries, generative models aim to approximate complex data distributions, allowing them to generate realistic samples from learned representations. CNNs have been the backbone of deep learning for visual data processing, excelling in tasks such as image and object detection, classification segmentation. By leveraging hierarchical feature extraction through convolutional layers, CNNs capture spatial patterns, making them highly effective for structured data like images.

1.1 Citizens Broadband Radio Service (CBRS)

CBRS in the United States refers to the unlicensed 150MHz spectrum in the 3.5 GHz band. CBRS allows dynamic and efficient allocation of spectrum resources, fostering both commercial and public interest applications. CBRS operates under a three-tiered hierarchy designed to prioritize and protect high-priority users while ensuring equitable access for lower-priority users: Incumbent, Priority Access License (PAL), and General Authorized Access (GAA).

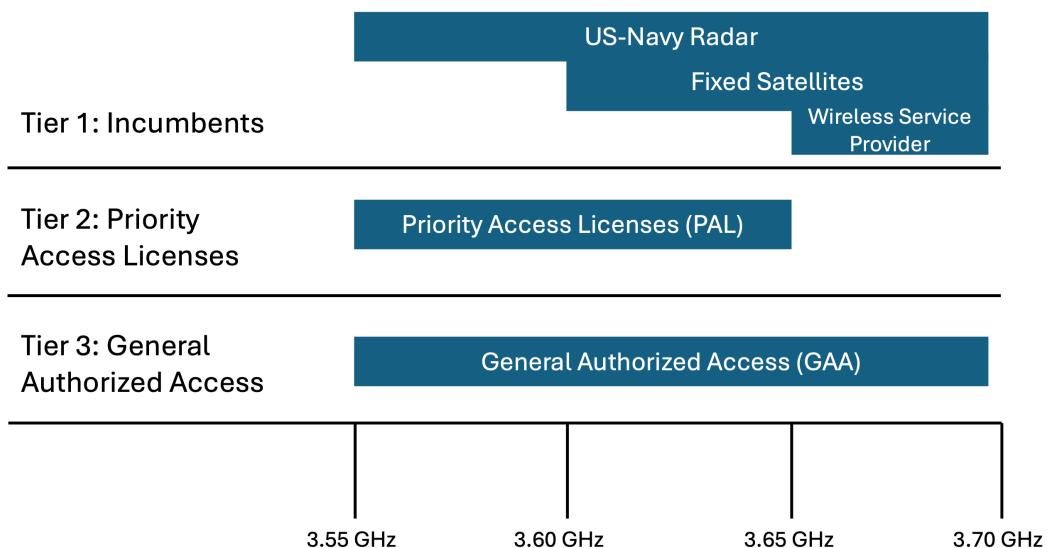


Figure 1.1: The CBRS spectrum

Incumbent users include federal military radar systems and fixed satellite stations. These users have the highest priority and are granted uninterrupted access to the spectrum. Typically, these systems operate along coastal regions and require significant protection from interference.

PAL Users are licensed commercial users who acquire spectrum rights through an auction process. PAL licenses are granted for up to 10 MHz in a

given geographic area and provide a level of protection from interference by lower-tier users. Common applications include private LTE/5G networks for enterprises, utilities, and healthcare.

GAA users are the unlicensed users who can access the spectrum opportunistically in areas where it is not in use by Incumbent or PAL users. GAA users operate under strict rules to prevent interference with the higher-priority tiers. Figure 1.1 illustrates the three tiers of the CBRS spectrum.

Since these tiers simultaneously use the CBRS spectrum, the FCC of the USA mandates that GAA users must not cause interference with PAL or incumbent users, and PAL users must not cause interference with incumbent users. A Spectrum Access System (SAS) is used to manage the spectrum and avoid potential interference [1].

SAS is a cloud-based, automated frequency management system mandated by the FCC to enforce CBRS rules and ensure efficient use of the spectrum. SAS acts as a dynamic coordinator, regulating how the spectrum is shared and preventing interference between users across the three tiers.

The SAS continuously monitors the spectrum usage and allocates channels dynamically to users based on their tier. Incumbent users are always given precedence, followed by PAL users, and then GAA users. To prevent collision scenarios, particularly involving Incumbent users, SAS currently employs several mechanisms, including continuous monitoring, spectrum reallocation for non-priority users, and preemption for Incumbent users.

Preventing interference is crucial in CBRS operations. SAS enforces a priority hierarchy, ensuring GAA users access spectrum only when PAL and Incumbent users are inactive. If a higher-priority user requires access, SAS re-

allocates GAA users dynamically. PAL users are similarly protected through exclusive spectrum assignments [2].

1.2 Background

GANs and diffusion-based generative models have gained significant traction for various applications, including spectrogram augmentation and audio synthesis. In [3], GAN-based models were employed for Radar spectrogram augmentation, leveraging a Joint Distribution Learning (JDL) module for injecting diversity into synthetic data. While effective for enhancing dataset variability, the reliance on adversarial training often poses challenges in terms of stability and convergence, particularly in complex spectrogram domains.

The work in [4] conducted a multimodal comparison of latent Diffusion Models (DDPMs) and GANs for medical image synthesis. Their results demonstrated the superiority of latent DDPMs over Least Squares GAN and Variational Autoencoder GAN, particularly in generating high-quality, diverse images. While this comparison offers insights into the potential of diffusion models, its application was limited to medical imaging and did not address the challenges of dynamic spectrum scenarios, such as those found in CBRS.

The U-Net architecture, proposed in Imagen [5] and later adapted in [6], has shown promise for spectrogram-based high-quality audio synthesis. These approaches exploit the hierarchical encoding-decoding structure of U-Nets to generate fine-grained audio spectrograms but have not been extended to the domain of spectrum-sharing systems or collision detection in CBRS.

Effective CBRS management relies on robust collision detection, yet tra-

ditional rule-based methods struggle in dynamic environments due to scalability and adaptability challenges. Deep learning-based classification offers a promising alternative, but the scarcity of collision data limits training [7].

To address this, we propose a novel spectrum-sharing system that generates spectrogram images of collision scenarios using GAN [8], DDPM [9], and VQ-VAE [10]. These images will train a CNN to detect collisions and alert SAS.

We evaluate the generative models by comparing their impact on CNN classification accuracy and data diversity. Then we simulate a CBRS environment using MATLAB which uses DRL for instead of a SAS to assign channels to users in a rapidly changing environment. While GANs and VAEs have been used to address data limitations, diffusion models have shown state-of-the-art results in natural image generation [4]. However, a direct use generative AIs with CNN for interference detection and DRL for channel allocation is unexplored.

This work employs conditional GAN, conditional DDPM, and VQ-VAE models to generate visualized CBRS datasets and compares their performance based on CNN classification accuracy.

The key contributions of this study are:

1. Introduce a scalable and generalizable spectrum-sharing system for a dynamic CBRS environment.
2. Use Generative models to generate visualized waveforms of a CBRS system.
3. Make a quantitative comparison of three of the most popular generative models currently available, in generating frequency spectrogram

images.

4. Use DRL based channel allocation system with a CNN based collision detection system to manage the CBRS environment.

This document consists of the following: Chapter 2 has the background explanations and the system architecture of this thesis and the Chapter 3 analyzes the results, performance of the generative models and the DRL agent. Chapter 4 includes a discussion of the results and Chapter 5 includes the conclusion and future work.

CHAPTER 2

METHODOLOGY

2.1 Dataset

The dataset utilized for training our generative models and the CNN model was obtained from the research conducted by [7] called the ViC dataset. This dataset comprises 2400 images belonging to 12 different scenarios for two channels.

Each channel in the system can be in one of four states: unused (empty), occupied by a primary user (RADAR), occupied by a secondary user (LTE), or experiencing interference (collision). Since the system involves two channels and two tiers of users, this results in 16 possible channel usage combinations. However, four of these combinations are excluded because the Citizens Broadband Radio Service Device (CBSD) is designed to operate on only one channel at a time. Therefore, the channels can exhibit 12 valid state configurations. Figure 2.1 presents the spectrograms that correspond to these 12 distinct channel states. A dataset of 2000 images was used to train the generative models, while an additional 400 images were used for training the CNN classifier.

The class names presented in Figure 2.1 will be used throughout this paper when discussing the results. Waveforms were visualized using Short Time Fourier Transform (STFT) with a size of 64×64 matrix.

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 9	Class 10	Class 11
Channel 1	Empty	Empty	Radar	Radar	Empty	LTE	Radar	LTE	Collision	LTE	Collision
Channel 2	Empty	Radar	Empty	Radar	LTE	Empty	LTE	Radar	Empty	Collision	Radar
Spectrogram Image											

Figure 2.1: Training ViC dataset containing the 12 classes [7]

2.2 VQ-VAE

The VQ-VAE is a generative model that combines continuous latent representations with discrete, interpretable codes by introducing a quantization mechanism. As illustrated in Figure 2.2 In VQ-VAE, the encoder maps the input x to a continuous latent representation $z_e(x)$ which is then quantized to the closest embedding $z_q(x)$ from a predefined discrete codebook e_1, e_2, \dots, e_k . This process ensures that the latent space is discrete and semantically meaningful. During training, the quantization mechanism minimizes the difference between $z_e(x)$ and $z_q(x)$ using a gradient-through approximation. The decoder then reconstructs the input x from the quantized representation $z_q(x)$. The use of codebook e_1, e_2, \dots, e_k enables the model to structure the latent space compactly, making it highly effective for tasks like image generation and speech synthesis. This discrete representation, denoted by $q(z|x)$, improves reconstruction quality by focusing on meaningful latent structures and allows efficient scaling for large datasets and complex generative tasks. The innovation of replacing continuous latent spaces with quantized embeddings results in a model that bridges the gap between interpretability and high-quality gener-

ation.

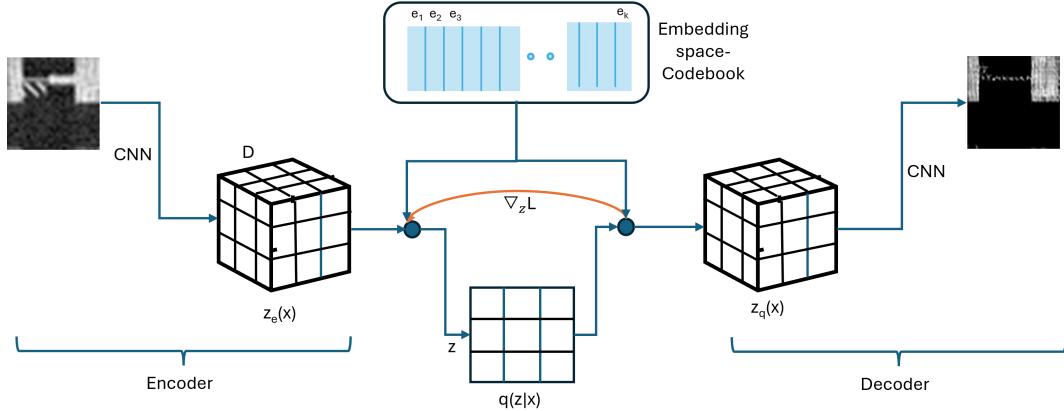


Figure 2.2: The VQ-VAE architecture

In the architecture of the conditional VQ-VAE we used, the encoder is a CNN designed to extract features from the input grayscale images. It consists of three convolutional layers that gradually decrease the spatial resolution while expanding the depth of feature maps. The encoder also incorporates a residual stack, which consists of multiple residual layers to enhance feature extraction by allowing information to bypass intermediate transformations, thereby improving gradient flow and capturing complex patterns.

After encoding, the latent representation is passed through a 1x1 convolution layer to adjust the feature dimensions to match the input requirements of the vector quantizer. This mapping is performed using the Euclidean distance between the latent vectors and the codebook embeddings. To ensure effective learning, the vector quantizer minimizes a combination of the reconstruction loss, a codebook commitment loss, and a latent quantization loss.

The loss function can be expressed as follows:

$$L = \mathbb{E}[\|x - \hat{x}\|^2] + \beta \cdot \mathbb{E}[\|z_{\text{enc}} - \text{sg}(z_{\text{quant}})\|^2] \quad (2.1)$$

where x is the input image, \hat{x} is the reconstructed image, z_{enc} is the latent vector from the encoder, z_{quant} is the quantized vector, and $\text{sg}(\cdot)$ denotes the stop-gradient operator.

The decoder is another CNN that takes the quantized latent representations and reconstructs the original input images. It mirrors the encoder's structure, starting with a convolutional layer followed by the residual stack to refine features, and ends with transposed convolutional layers to upsample the latent representation back to the original input dimensions. The decoder ensures that the reconstructed images retain high fidelity to the input.

We implemented a conditional VQ-VAE model, enabling class-specific image generation. The encoder compresses grayscale input images using three convolution layers followed by residual blocks, while the decoder mirrors this process using transposed convolutions to reconstruct the data. The quantization layer is also class-conditioned, leveraging a codebook with 512 embeddings, each of 64 dimensions.

2.3 GAN

GANs, are composed of two neural components: a generator that synthesizes data samples and a discriminator that assesses their authenticity. These two models are trained in an adversarial framework, where the generator attempts to produce data that the discriminator cannot distinguish from real

data, while the discriminator learns to correctly differentiate between real and generated samples. Through this competitive training, GANs are capable of generating highly realistic images, audio, and other types of data[11]. The corresponding loss functions for the generator and discriminator are expressed below. Figure 2.3 illustrates this architecture. L_D and L_G are the loss

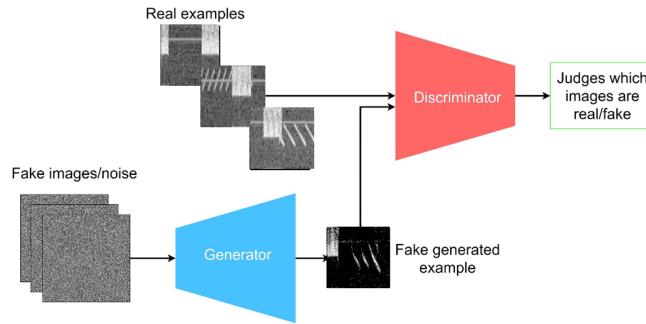


Figure 2.3: The GAN architecture

functions for the Discriminator and the Generator for image x , respectively.

$$L_D = \mathbb{E}[\max(0, 1 - D(x_{real}))] + \mathbb{E}[\max(0, 1 + D(G(z_{fake})))] \quad (2.2)$$

$$L_G = \mathbb{E}[D(G(z_{fake}))] \quad (2.3)$$

We used a conditional GAN with a Generator \mathcal{G} and a Discriminator \mathcal{D} which were trained simultaneously through a min-max game. For the Generator, we used an input noise vector z of 100 dimensions and an embedding layer to convert class labels into embeddings with five convolution layers and ReLU activations in hidden layers. Tanh activation was used in the output layer to produce pixel values in the range [-1, 1]. In the Discriminator, five

convolution layers were used to down-sample the images with Leaky ReLU activations in hidden layers. We used Sigmoid activation in the output layer to produce the probability.

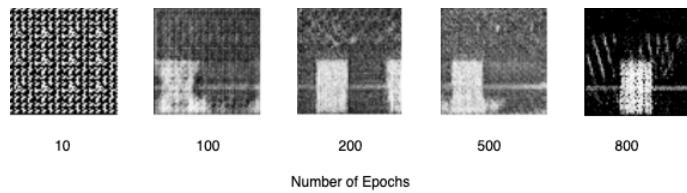


Figure 2.4: Spectrogram images generated by the Generator of the GAN

We generated 1200 images with 100 images per class for the twelve classes and used the pre-trained CNN model to classify the images.

2.4 DDPM

DDPMs are generative models designed for image generation using variational inference. They operate through a Markovian process that involves a finite sequence of steps, denoted as T . The training process consists of two key phases: the forward phase, which gradually introduces noise to the data in a systematic way, and the reverse phase, where the model is trained to progressively remove the noise and reconstruct the original data. Each step in this process acts as a denoising operation, aiming to refine the image quality progressively[9]. Figure 2.5 illustrates the diffusion process of the DDPM.

During the forward process, the clean image y_0 is sampled with Gaussian

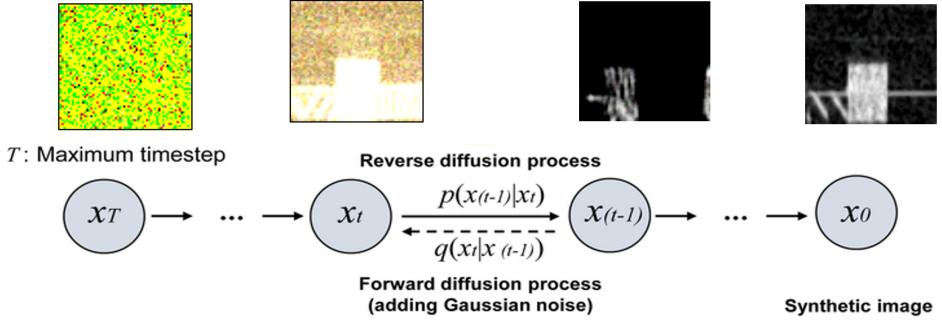


Figure 2.5: Diffusion process of the DDPM

noise of variance β_1, \dots, β_T is added over T time steps.

$$\begin{aligned} q(y_t | y_0) &:= \mathcal{N}(y_t; \sqrt{\bar{\alpha}_t}y_0, (1 - \bar{\alpha}_t)I) \\ &= \sqrt{\bar{\alpha}_t}y_0 + \epsilon\sqrt{1 - \bar{\alpha}_t}, \epsilon \sim \mathcal{N}(0, I) \end{aligned} \tag{2.4}$$

Where $\alpha_t = 1 - \beta_t$

In the reverse process, the added noise is removed step by step to recover the image y_0 .

$$p(y_T) = \mathcal{N}(0, I) \tag{2.5}$$

$$p(y_{(t-1)} | y_t) = \mathcal{N}(y_{(t-1)}; \mu_\theta(y_t, t), \sqrt{\beta_t}I) \tag{2.6}$$

We used a conditional DDPM, and when training the DDPM we enabled mixed precision training and multithreaded data loaders. We sampled the images regularly during the training and we could see the learning rate gradually decreasing with the time steps. For training our model we sampled a timestep $t \sim U[1, T]$ with $T = 2000$ and a 5×10^{-3} learning rate. After training the model, we used the trained weights to generate 1200 images with 100 images per class and used the pre-trained CNN model to classify the images to

their respective classes.

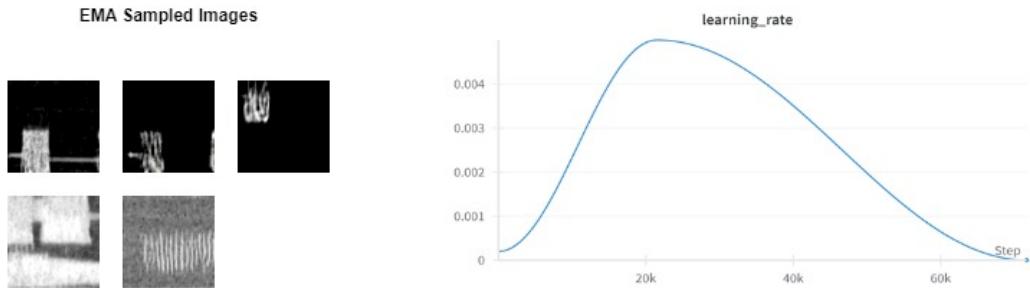


Figure 2.6: EMA sampled images and the change of the learning rate with each time step when training the DDPM

2.5 CNN

CNNs are a class of deep learning models commonly used for analyzing images. They automatically learn spatial feature hierarchies from input data by applying multiple layers of convolutional filters. These filters detect patterns such as edges, textures, and more complex features in the images, making CNNs highly effective for tasks like image classification.

We trained a CNN model that can detect collision scenarios using the generated spectrogram data from the three models. Since the spectrogram images generated by the DDPM, GAN, and VQ-VAE models contained some noise, which could have led to misclassification by conventional CNN architectures, particularly when distinguishing fine-grained features, we employed an enhanced CNN architecture specifically designed to effectively capture and differentiate subtle details within the images, thereby mitigating the impact of noise on classification accuracy.

We used a CNN architecture based on a modified ResNet-50 model, en-

hanced with a Convolutional Block Attention Module (CBAM) to improve its feature extraction and classification capabilities. The architecture comprised convolutional layers for basic feature extraction, residual blocks for efficient gradient propagation, Global Average Pooling (GAP) for summarizing spatial information, and a Fully Connected (FC) layer adapted for classifying the 12 spectrogram image classes.

The model was trained on spectrogram images resized to 448×448 for capturing finer details. The training used the Adam[12] optimizer for adaptive gradient updates and a learning rate scheduler that reduced the learning rate every five epochs to ensure stable convergence.

2.6 DRL

In this work, we employed a DRL agent to learn optimal spectrum access strategies in a simulated CBRS environment. The agent interacts with an environment that mimics the dynamic characteristics of the CBRS band, including incumbent radar signals and LTE-based priority access users. The environment is modeled as a Markov Decision Process (MDP) where the agent observes the occupancy of two communication channels and selects an action from a discrete set: idle, use channel 1, or use channel 2.

We implemented a Deep Q-Network (DQN) agent that approximates the Q-value function using a neural network. The agent receives as input a two-dimensional observation vector representing the current channel state, where each element indicates whether a channel is occupied (1) or free (0), as predicted by the pre-trained CNN model that analyzes the spectrogram of the re-

ceived signal. The agent’s goal is to maximize a cumulative reward by learning a policy that avoids interference with higher-tier users while efficiently utilizing the available spectrum. The reward structure is defined such that selecting an idle channel yields a positive reward, selecting an occupied channel results in a penalty, and remaining idle results in a small negative reward to discourage inactivity.

The neural network used by the DQN agent consists of a feature input layer followed by two fully connected layers with ReLU activations, and a final output layer that estimates Q-values for each possible action. During training, we enabled GPU acceleration to improve performance. The training utilized an experience replay buffer and a target network, and was configured with Double DQN and a smooth target update factor of 10^{-3} to stabilize learning. We set the discount factor $\gamma = 0.99$, and trained the model for 1000 episodes with a maximum of 100 steps per episode.

The environment generates new signal observations by randomly simulating different channel conditions, including ‘empty’, ‘radar’, ‘LTE’, and ‘collision’. These synthetic signals are converted into spectrogram images, resized to 224×224 pixels, and classified by the CNN. This predicted class is used to update the channel occupancy vector, which forms the observation space for the DRL agent.

2.7 System Architecture

We evaluated classification performance across different generative models and computed class-wise and overall accuracy, along with FID scores for each

class. Then, we used a pre-trained CNN model trained with a dataset comprising both original data and the data generated by the three generative models to detect incumbent users. Figure 2.7 illustrates this scenario.

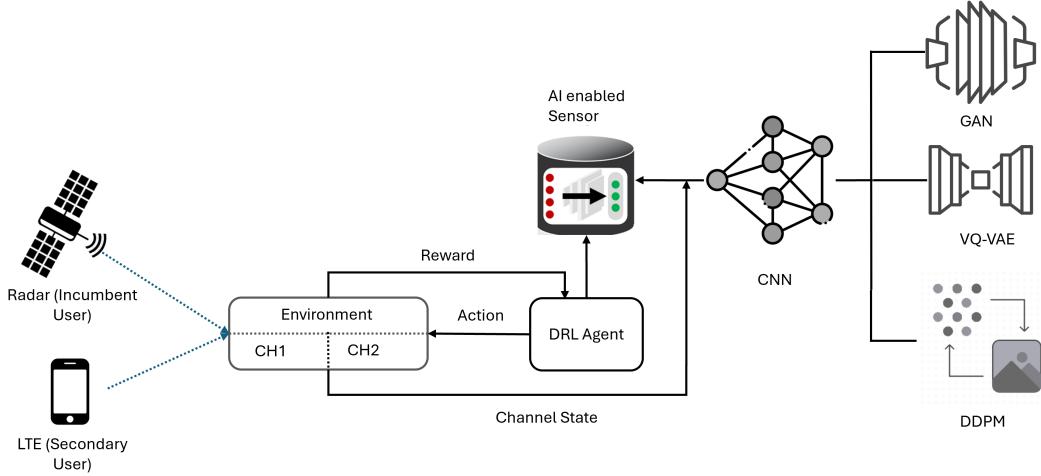


Figure 2.7: System Architecture

2.8 Experimental Setting

2.8.1 VQ-VAE

The conditional VQ-VAE model was designed for class-specific image generation, with an encoder that compresses grayscale inputs through three convolutional layers and residual blocks to enhance feature extraction. The quantization layer, class-conditioned, uses a codebook of 512 embeddings (64 dimensions each) to discretize the latent space. A 1×1 convolution adjusts the encoder's output to fit the quantizer's input requirements. The decoder, structured similarly to the encoder, uses transposed convolutions and residual stacks

to reconstruct high-fidelity images from quantized representations. The model generated 1,200 images, with 100 per class across 12 classes, which were classified using a pre-trained CNN.

2.8.2 GAN

The conditional GAN used in this study consists of a generator and discriminator trained in a min-max game. The generator takes a 100-dimensional noise vector and class embeddings, passing them through five convolutional layers with ReLU activations and a Tanh output to generate pixel values in the range [-1, 1]. The discriminator uses five convolutional layers with Leaky ReLU activations to downsample input images, and a Sigmoid output provides a probability score. The model generated 1,200 images, with 100 per class across 12 classes, which were classified using a pre-trained CNN.

2.8.3 DDPM

The conditional DDPM used in this study generates images through a two-stage Markovian process: a forward process, where Gaussian noise is added to the clean image over $T=2000$ timesteps, and a reverse process, where the model removes the noise step-by-step to reconstruct the original image. Mixed precision training and multithreaded data loaders were utilized for efficiency, and images were sampled during training to monitor progress. The model was trained with a learning rate of 5×10^{-3} , sampling timesteps $t \sim U[1, T]$. After training, 1,200 images (100 per class) were generated and classified using a pre-trained CNN.

2.8.4 CNN

A modified ResNet-50 architecture with a CBAM was used to classify 12 spectrogram image classes, effectively capturing fine-grained details despite noise. The model was trained on 448×448 resized images using the Adam optimizer and a learning rate scheduler to ensure stable convergence. For training, we used a combination of the original dataset and the generated data from each model. The combined dataset was divided into 80% for training and 20% for inference, and the results are based on three separately trained models, each incorporating generated data from one of the three-generation models.

We compared the class-wise and overall accuracy of the CNN classification model trained with data generated by different models for the inference set. Then, we calculated the FID score for each class.

2.8.5 FID

FID is a widely used metric for evaluating the quality of images generated by generative models. It quantitatively compares the distributions of features extracted from real and generated images in a high-level feature space. These features are typically obtained from a pre-trained neural network, such as the Inception-v3 model. The FID score is lower when the two distributions are closer, indicating higher quality and diversity of the generated images [21].

Let μ_r and Σ_r represent the mean and covariance of the features extracted from real images, and μ_g and Σ_g represent the mean and covariance of the features extracted from generated images. The FID score is calculated using the following formula:

$$\text{FID} = \|\mu_r - \mu_g\|_2^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{\frac{1}{2}}) \quad (2.7)$$

Usually, to compute the FID, real and generated images are passed through the pre-trained Inception-v3 network, and features are extracted from a specific layer (often the "pool3" layer). The means and covariances of these features are then computed for both datasets. However, in this paper, we used a ResNet18 model with the final classification layer replaced instead of Inception-v3. The ResNet18 model employed for FID calculation is distinct from the CNN model used for classification. While the ResNet18 was trained exclusively on the original dataset to extract consistent feature representations for real and generated images, the classification CNN was trained on a combined dataset comprising both real and synthetic data. This separation ensures that the FID evaluation remains unbiased by the influence of synthetic data used in the classification task. Finally, the FID score is calculated by substituting these statistics into equation 2.7.

A lower FID Score indicates that the distributions of real and generated features are more similar, implying better quality and diversity of generated images whereas a higher FID score indicates that the generated images are less similar to real images, suggesting poorer quality or lack of diversity.

2.8.6 PCA

PCA is a statistical technique used for dimensionality reduction, facilitating the analysis and visualization of high-dimensional data. By transforming the original variables into a new set of uncorrelated variables, PCA captures the

directions of maximum variance in the data. The first principal component accounts for the largest possible variance, with each succeeding component accounting for the remaining variance under the constraint of being orthogonal to the preceding components. This method simplifies complex datasets while preserving their essential patterns and structures [13].

2.8.7 DRL

The DRL agent observes the current state (e.g., channel occupancy or spectrogram predictions), selects an action (e.g., choose Channel 1, Channel 2, or remain idle), and receives a reward based on the outcome (e.g., successful transmission or interference avoidance). Over repeated episodes, the agent learns to maximize its long-term performance, such as throughput or interference avoidance, by training a neural network to approximate optimal policies. Figure 2.8 depicts Incumbent signals generated by the MATLAB simulation, where the agent learns to select the best channel based on the spectrogram images generated by the three models. The agent's training involved 1000 episodes, with each episode consisting of 100 steps, and it was trained using a reward structure that incentivized efficient spectrum access while avoiding interference with higher-tier users. The agent was rewarded or penalized according to the steps it chose to take. The reward structure was defined as shown in Table 2.1.

The agent was highly rewarded for choosing an empty channel and penalized for choosing a channel occupied by a primary user. The agent was also rewarded for using a secondary channel, but less than when it chose an empty channel. The agent was penalized for choosing a channel occupied by

Table 2.1: DRL agent reward structure

Action	Channel State	Reward
Idle (0)	Both channels empty	-0.1
Idle (0)	Any channel non-empty (primary or collision)	+0.2
Use CH1(1)	CH1 Empty	+1.0
Use CH1(1)	CH1 Primary	-2.0
Use CH1(1)	CH1 Collision	-1.5
Use CH1(1)	CH1 Secondary	+0.5
Use CH2(2)	CH2 Empty	+1.0
Use CH2(2)	CH2 Primary	-2.0
Use CH2(2)	CH2 Collision	-1.5
Use CH2(2)	CH2 Secondary	+0.5

a collision or primary user. The agent was also penalized for remaining idle, but less than when it chose a channel occupied by a primary user.

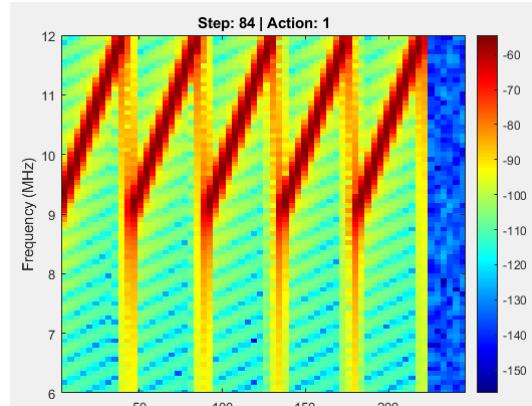


Figure 2.8: Spectrograms of the signals generated by the simulated CBRS environment in MATLAB

CHAPTER 3

RESULTS

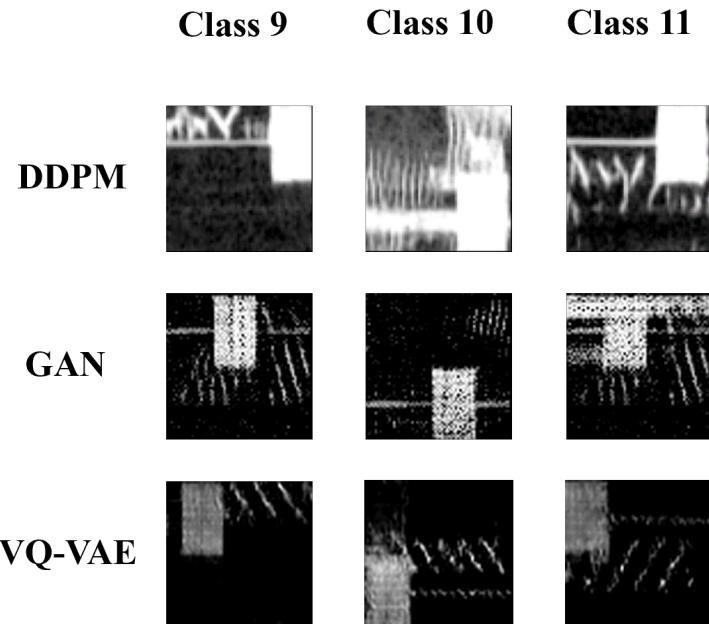


Figure 3.1: A side-by-side comparison of some of the collision scenarios generated by the DDPM, the GAN, and the VQ-VAE

Classes depicted in Figure 3.1: class 9, class 10, and class 11 are all collision scenarios (Refer to Figure 2.1), which were generated by the DDPM, the GAN, and the VQ-VAE.

Figure 3.2 illustrates the change in FID score across the classes for the 3 models. The GAN, DDPM, and VQ-VAE models achieved average FID scores of 1.944, 0.504, and 1.580, respectively. We used a ResNet18 model with the final

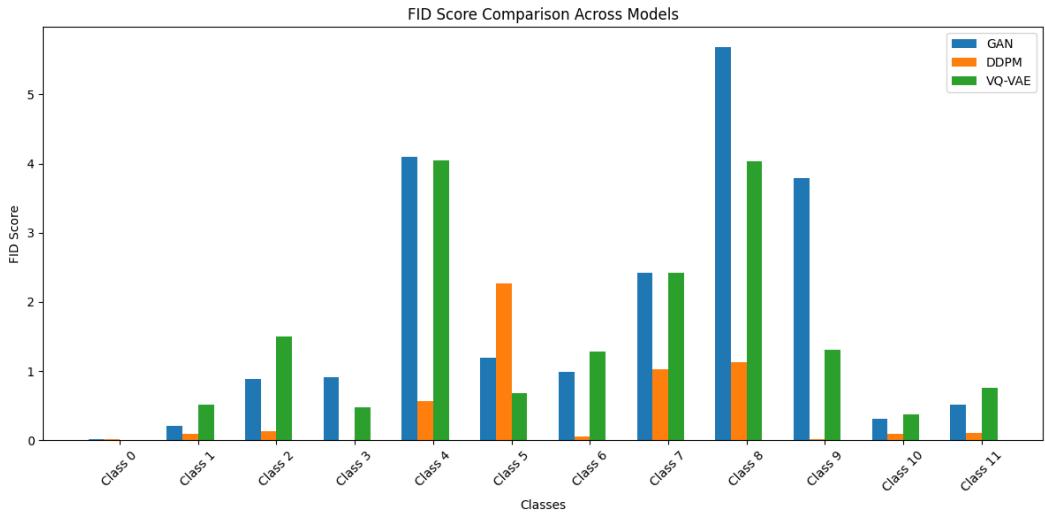


Figure 3.2: FID Score Comparison Across Models for Different Classes.

classification layer replaced to extract features instead of using a pre-trained InceptionV3 network due to the specific nature of our dataset.

Since our dataset lacks the complex textures, patterns, and color information typically found in natural image datasets, the differences between real and generated spectrograms are smaller in the feature space. As a result, the FID scores we obtained are smaller than usual for all three models.

Class 0 consistently shows near-zero FID scores across all three models due to its purely noise-like nature. As observed in the graph, the classes representing collision scenarios—Class 9, Class 10, and Class 11—exhibit significantly lower FID scores for images generated by the DDPM. This suggests that the spectrogram images with collision scenarios generated by the DDPM are more natural and resemble the original dataset better than the other two models. The other two models also have relatively low FID scores for Class 10 and Class 11.

Among the three models, the DDPM also has the lowest average FID score, suggesting that DDPM is the most effective model overall for generating spectrograms that closely resemble the real dataset, as evidenced by its consistently low FID scores across most classes.

The GAN model's high FID scores across many classes suggest model collapse in its generated spectrograms for this dataset, to which GANs are susceptible. Given that our main focus is on generating a diverse dataset to improve the accuracy of a visualized interference classification system, having a wide variety of data is essential.

VQ-VAE model, on the other hand, shows varied performance, outperforming GAN in many cases (e.g., Classes 1, 2, 6, 9) but falling behind DDPM in most.

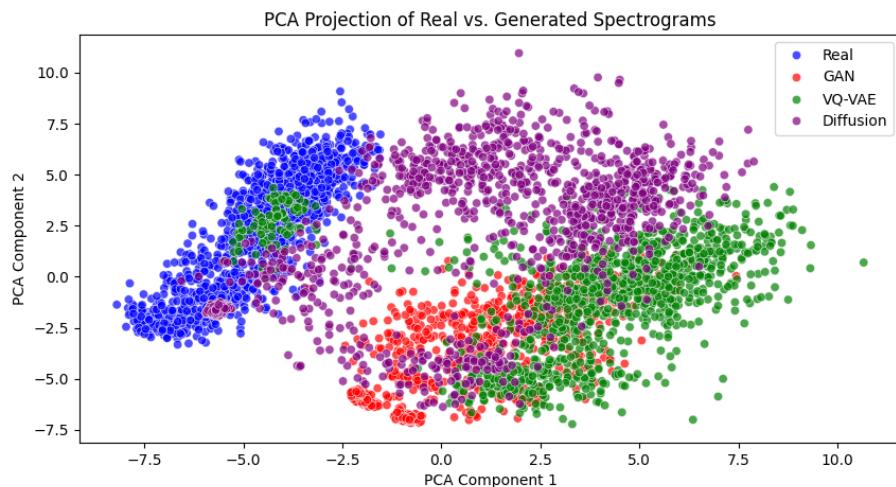


Figure 3.3: PCA scores for the GAN, DDPM and VQ-VAE

The PCA projection in Figure 3.3 indicate that real and generated spectrograms form separate yet overlapping clusters, suggesting that generative models do not simply reproduce existing samples but introduce new varia-

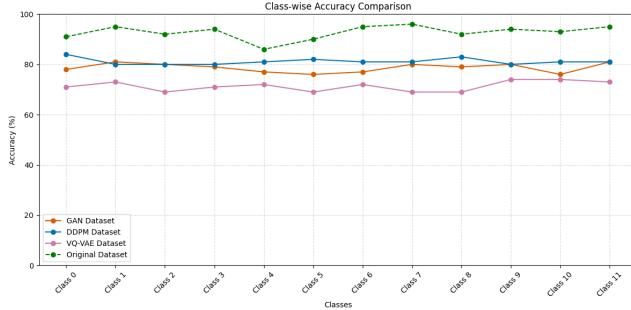


Figure 3.4: Class-wise classification accuracy for GAN, DDPM, VQ-VAE generated data with original dataset

tions. The spread of generated samples in the PCA projection indicates that the models create additional variations, potentially improving classifier generalization. The PCA analysis provides an independent validation of data distribution differences, confirming that generative models add meaningful diversity. However, the PCA projection also shows that the GAN model's generated samples are more clustered and less spread out than the other two models, indicating that the GAN model may not be as effective at generating diverse samples as the DDPM and VQ-VAE models. The DDPM model's generated samples are more spread out than the VQ-VAE model's, suggesting that the DDPM model may be more effective at generating diverse samples than the VQ-VAE model.

Figure 3.4 illustrates the change in the classification accuracy of the CNN model for the spectrogram images generated by the three generative models for the twelve different data classes. Meanwhile, the spectrogram images generated by the DDPM generally outperformed the images generated by other models. Even though we used a CNN model specifically trained to distinguish fine-grained features ignoring the noise, the high noise and blurry features in

the VQ-VAE model seem to have affected the accuracy of the CNN model than the other two models.

Cases	Model	Average Accuracy	FID Score
1	DDPM	81.9%	0.504
2	GAN	78.5%	1.944
3	VQ-VAE	71.3%	1.580
4	Original dataset	92.75%	-

Table 3.1: Average accuracy for GAN, DDPM, VQ-VAE and Original data

Table 1 depicts the average classification accuracy and the FID scores of the spectrogram images, for the three different models and the original data. Although the average accuracy is not far apart for the three generative models, due to the lack of diversity of the generated data by the GAN, and the noisy images by VQ-VAE we can conclude that the DDPM performs better than the GAN or VQ-VAE when generating spectrogram images for a robust spectrum-sharing system.

We can also notice that the classification accuracy for the original dataset is much higher than the other three models. This is expected since the original dataset is a real dataset, and the CNN model was trained on it. However, the DDPM model's generated data is not far behind the original dataset, indicating that the DDPM model can generate high-quality spectrogram images that can be used to train a robust spectrum-sharing system. But when these models were directed to detect collisions in the simulated CBRS environment all three generative models performed better than the original dataset. This is because the original dataset was not diverse enough to train a robust CNN model. We will discuss this later in the chapter.

After calculating the classification accuracy we trained four DQN agents

using the generated data from the three models and the original data. The DQN agents were trained for 1000 episodes. We then evaluated the trained DQNagent based on how well the agent was able to avoid collisions using the CNN classifications of the four models in the simulated MATLAB CBRS environment.

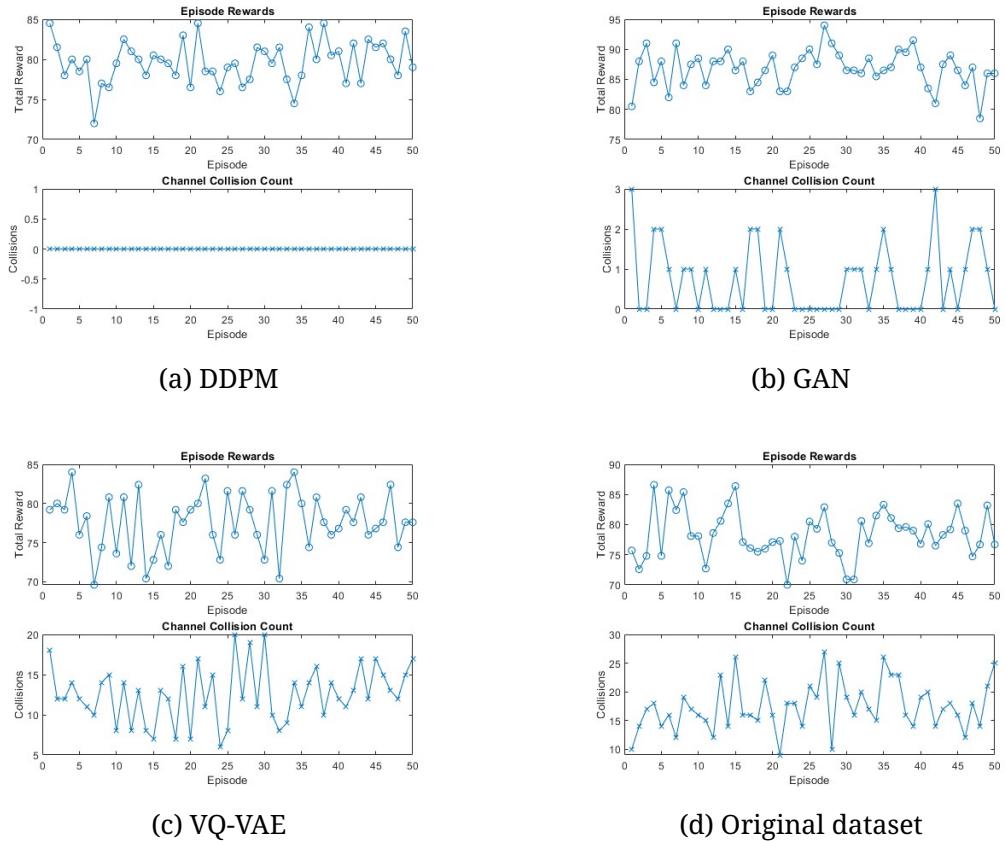


Figure 3.5: Performance of DQN agents trained with the four different datasets

Figure 3.5 and Table 3.1 depicts the performance of DQN agents trained with datasets generated using DDPM, GAN, VQ-VAE, and the original dataset. The DDPM-trained agent exhibits highly stable behavior, achieving a consis-

Table 3.2: Comparison of Generative Models in Terms of Mean Reward and Collision Rate

Model	Mean Reward	Mean Collisions per Episode
DDPM	79.65	0.00
GAN	86.78	0.74
VQ-VAE	77.62	12.60
Original	78.40	17.44

tently high reward across episodes with zero collisions—indicating excellent generalization and safe spectrum selection. The GAN-based agent demonstrates the highest average reward among all models; however, it incurs occasional collisions, suggesting a trade-off between performance and safety. In contrast, the VQ-VAE-trained agent shows high variability in both rewards and collisions, implying that its synthetic data may introduce noise or uncertainty, leading to less reliable policy learning. Lastly, the agent trained on the original dataset performs moderately well in terms of reward but experiences the highest average collision rate, underscoring the benefit of augmenting training with high-quality synthetic data like that from DDPM. These results collectively highlight that generative models, especially DDPM, can significantly enhance the safety and performance of DQN agents in dynamic spectrum access scenarios.

CHAPTER 4

DISCUSSION

In this paper, we presented a deep learning-based system for interference classification in the CBRS spectrum, leveraging synthetic data generated by three distinct generative models: GANs, DDPMs, and VQ-VAEs. A CNN was trained using the datasets produced by each generative model, and the classification accuracies were compared. Then that pre-trained CNN was used to detect Incumbent users in a simulated CBRS environment with a DRL agent for channel allocation. The performance of the DRL agent and the CNN was measured. Additionally, the FID was used to evaluate the quality and diversity of the images generated by the models.

The experimental results show that the images generated by DDPM, GAN, and VQ-VAE achieved classification accuracies of 81.9%, 78.5%, and 71.3%, respectively. Among the three models, DDPM achieved the lowest FID score, indicating that it produced images with the highest diversity while maintaining fidelity. These findings suggest that DDPM-generated datasets when combined with CNN-based classification, provide the most effective approach for interference classification in the CBRS spectrum. The combination of superior classification accuracy and low FID score underscores the advantage of DDPM in this application.

Furthermore, the performance of DQN agents trained using each genera-

tive model was analyzed in terms of reward and safety in a simulated CBRS environment. The DDPM-trained agent showed the best safety profile, maintaining a mean reward of 79.65 with zero collisions per episode, demonstrating strong generalization and conservative channel selection. The GAN-trained agent achieved the highest mean reward of 86.78, though it incurred an average of 0.74 collisions per episode, reflecting a more aggressive but risk-prone strategy. In contrast, the VQ-VAE-trained agent had a lower mean reward of 77.62 and suffered from 12.6 collisions per episode, while the agent trained on the original dataset yielded a reward of 78.4 with 17.44 collisions per episode. These results highlight the effectiveness of DDPM in producing not only high-quality training data but also enabling safe and efficient decision-making by reinforcement learning agents in dynamic spectrum-sharing settings.

While the proposed system is tailored for collision detection and spectrum management within CBRS, its architecture and methodology are inherently adaptable to other dynamic spectrum-sharing frameworks. The use of generative models such as DDPMs, GANs, and VQ-VAEs for creating synthetic collision scenarios and training robust classifiers is not limited to CBRS but can be extended to any system that requires dynamic interference management and real-time decision-making.

For instance, TV White Spaces (TVWS)[14], which allow unlicensed devices to operate in unused portions of the broadcast television spectrum, face challenges similar to CBRS in managing interference between primary and secondary users. The synthetic data generation techniques presented in this work can aid in training deep learning classifiers for detecting spectrum misuse or interference in such systems.

Additionally, this approach is well-suited for 5G and 6G networks, where ultra-dense deployments and diverse spectrum use cases demand advanced interference detection and management capabilities. In these environments, generative models can simulate interference scenarios across various network slices, enabling operators to develop more resilient DSA systems.

An important future research direction is addressing the challenge of distinguishing between PAL and GAA users, particularly in scenarios where their spectral behavior appears identical. One promising approach could involve incorporating network-level information, such as decoding the System Information Block (SIB1) message to retrieve the Mobile Network Code (MNC) of colliding networks. This would allow for identifying the specific networks involved in suspected PAL and GAA collisions. Integrating tools such as Keysight Nemo with the proposed deep learning framework could create a hybrid system where spectral analysis detects potential collisions and network monitoring tools provide additional validation and identification. This complementary methodology offers a pathway for further enhancing spectrum management and interference classification strategies.

CHAPTER 5

CONCLUSION

This thesis presented a comprehensive study on dynamic spectrum sharing in the Citizens Broadband Radio Service (CBRS) environment through the integration of deep learning and generative artificial intelligence models. The growing complexity and demand for wireless communication systems necessitate more intelligent, flexible, and scalable approaches to spectrum management. Traditional static and rule-based systems fail to accommodate the rapid evolution of wireless technologies and user behavior, especially in spectrum-constrained environments like CBRS. In response, we proposed a novel pipeline that synergizes generative models, convolutional neural networks (CNN), and deep reinforcement learning (DRL) to enable dynamic and efficient channel allocation.

A key innovation of this work lies in the use of generative models—GAN, VQ-VAE, and DDPM—to synthesize realistic spectrogram images simulating various channel occupancy scenarios. These include non-interfering states (empty, radar, LTE) and critical interference cases (collision). The synthetic spectrograms were then used to train a CNN-based classifier for automatic collision detection, a critical functionality in any spectrum access system. Our comparative study revealed that DDPM outperformed both GAN and VQ-VAE in terms of Fréchet Inception Distance (FID) and classification accuracy, in-

dicating superior image quality and diversity. These findings align with the emerging consensus in the machine learning community regarding the superiority of diffusion-based models for high-fidelity generation tasks.

To evaluate the practical utility of the generative models and the trained CNN, we implemented a MATLAB-based CBRS simulation environment. In this simulated environment, a DRL agent was tasked with making real-time channel allocation decisions based on the CNN’s predictions. This framework effectively replaces a conventional Spectrum Access System (SAS), offering a decentralized and learning-based alternative that can adapt to previously unseen scenarios. The DRL agent was trained using a Deep Q-Network (DQN) architecture with experience replay and Double DQN to ensure stability. By using a binary channel occupancy vector derived from spectrogram classification, the agent learned to avoid collisions with Incumbent and PAL users while maximizing spectral efficiency for General Authorized Access (GAA) users.

This work not only demonstrates the viability of using AI-driven techniques for spectrum sharing but also provides evidence of their practical superiority over traditional hand-engineered solutions. By generating diverse training data, we overcame one of the primary challenges in supervised learning: the scarcity of labeled samples in complex environments. The trained CNN achieved high accuracy even for subtle or rare collision patterns, and the DRL agent showed robust behavior during spectrum reallocation under dynamically changing conditions.

Furthermore, we demonstrated that combining synthetic data with real training data can enhance classifier generalization. This is particularly impor-

tant for regulatory and commercial deployment, where real-world datasets may be highly imbalanced or incomplete. The DDPM-generated spectrograms not only improved classification performance but also contributed to a more diverse and representative training set, thereby reducing model bias and improving resilience in unseen environments.

Despite these achievements, there are several limitations and open problems that warrant further research. First, while our models distinguish between classes like radar and LTE effectively, they are less adept at distinguishing between PAL and GAA users due to similar waveform characteristics. This is a significant limitation in a practical SAS system, where different policy constraints apply to each user type. Future work could address this challenge by incorporating metadata or control-plane information, such as the Mobile Network Code (MNC) obtained via System Information Block (SIB1) messages, into the decision-making pipeline.

Another future direction involves improving the temporal and frequency resolution of the spectrograms. Currently, we use fixed STFT parameters to generate 64x64 or 224x224 resolution spectrograms, which may limit the fidelity of subtle spectral features. Adaptive time-frequency representations, or even raw I/Q signal modeling using 1D convolutional networks or transformers, may offer better performance for dynamic and complex signals. Furthermore, while this work focused on two channels, scaling to multi-channel ($N > 2$) spectrum environments introduces new challenges in DRL action space, spectrogram representation, and interference modeling.

Expanding the generative models themselves is also a promising path. While conditional DDPM performed best among the tested models, hybrid approaches

such as classifier-guided diffusion, or Score-based Generative Models, could yield even better results. Similarly, self-supervised learning or contrastive learning approaches could reduce the reliance on class labels, enabling spectrum analysis in unsupervised or weakly labeled domains.

From a systems perspective, this research opens the door to real-time, learning-driven SAS implementations. By integrating tools such as Keysight Nemo or open-source LTE/5G sniffers, the DRL agent could be trained or validated in near-real-world conditions. The modular nature of the system separating generation, classification, and control—makes it easy to swap components or upgrade specific functionalities as new models and techniques emerge. In practical deployments, this could enhance spectrum efficiency, reduce operational costs, and accelerate the adoption of shared spectrum systems globally.

In conclusion, this thesis has shown that generative AI models, when combined with CNN-based classification and DRL-based control, offer a powerful and scalable framework for dynamic spectrum sharing in CBRS. The proposed system bridges the gap between data scarcity and real-time decision-making, and it sets a foundation for future research in AI-driven spectrum management. Our experiments demonstrate not only the technical feasibility of such a system but also its strong performance across key metrics of accuracy, diversity, and adaptability. As spectrum becomes an increasingly scarce and valuable resource, AI-based systems like the one proposed here will be essential to ensuring equitable, efficient, and intelligent spectrum usage in the wireless networks of the future.

REFERENCES

- ¹HPE Aruba Networking, “What is cbrs?”, <https://www.arubanetworks.com/faq/what-is-cbrs/>.
- ²Federal Communications Commission, “Amendment of the commission’s rules with regard to commercial operations in the 3550-3650 mhz band”, FCC Report and Order (2015).
- ³Y. Yang, Y. Zhang, Y. Lang, B. Li, S. Guo, and Q. Tan, “Gan-based radar spectrogram augmentation via diversity injection strategy”, IEEE Transactions on Instrumentation and Measurement, vol. 72, pp. 1–12 (2022).
- ⁴P. Dhariwal and A. Nichol, “Diffusion models beat gans on image synthesis”, CVPR (2021).
- ⁵C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi, “Photorealistic text-to-image diffusion models with deep language understanding”, Advances in Neural Information Processing Systems, 35:36479–36494 (2022).

- ⁶G. Zhu, Y. Wen, M.-A. Carbonneau, and Z. Duan, “Edmsound: spectrogram based diffusion models for efficient and high-quality audio synthesis.”, <https://arxiv.org/pdf/2311.08001.pdf> (2023).
- ⁷H. Nam, K. Han, J. Park, H. Ch, and S.-L. Kim, “A spectrum sharing system: visualized interference classification and cnn approach”, 2021 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Los Angeles, CA, USA (2021).
- ⁸I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets”, Advances in Neural Information Processing Systems, vol. 27 (2014).
- ⁹J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models”, NeurIPS 2020 (34th Conference on Neural Information Processing Systems) (2020).
- ¹⁰A. van den Oord, O. Vinyals, and K. Kavukcuoglu, “Neural discrete representation learning”, NIPS’17: Proceedings of the 31st International Conference on Neural Information Processing Systems (2018).
- ¹¹A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative adversarial networks: an overview”, IEEE Signal Processing Magazine, vol. 35, no. 1, pp. 53-65 (2018).

¹²D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization”, 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, May 2015. (2015).

¹³F. L. Gewers, G. R. Ferreira, H. F. de Arruda, F. N. Silva, C. H. Comin, D. R. Amancio, and L. da F. Costa, “Principal component analysis: a natural approach to data exploration”, ACM Comput. Surv. 54, 4, Article 70 (2022).

¹⁴Federal Communications Commission, “White space”, <https://www.fcc.gov/general/white-space> (2011).