MANIFOLD-REGULARIZED LATENT DIFFUSION MODEL FOR PARTITION-ROBUST OPEN SET WIRELESS SIGNAL RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Wireless Signal Recognition (WSR) leverages deep learning to perform blind estimation of signal attributes, serving as the core technology for spectrum monitoring and wireless security. Real-world spectrum environments are open and dynamic, which requires models to accurately classify known signals while reliably detecting unknown types. This defines the open set WSR problem. Despite its importance, robust open set WSR remains underexplored due to a fundamental barrier: partition sensitivity, where model performance and rankings fluctuate across different data partitions, undermining evaluation consistency and deployment reliability. This challenge is particularly acute for wireless signals: channel-induced perturbations cause substantial intra-class feature variation, while inherent structural similarities among signal types render inter-class boundaries fragile. To address this, we formalize the problem as Partition-Robust Risk Minimization (PRRM), which minimizes a surrogate risk that jointly optimizes expected performance and suppresses risk variance. The variance decomposes into sample-level variance and class-composition variance. Guided by PRRM, we introduce DualDiff, which employs manifold-regularized latent diffusion to construct a smooth feature manifold that reduces sensitivity to sample-level perturbations, and geometry-aware fine-tuning to stabilize class boundaries against subset composition changes. Extensive experiments demonstrate that DualDiff achieves strong open set recognition performance while maintaining consistent rankings across diverse data partitions.

1 INTRODUCTION

Wireless signals serve as the primary medium for information exchange in modern communication systems Alsharif et al. (2020)Li et al. (2019). Wireless Signal Recognition (WSR) leverages deep learning to identify key attributes of received signals, such as modulation type Bitar et al. (2017), without prior knowledge. The resulting identification provides a foundation for subsequent signal processing and downstream tasks. WSR has been widely adopted in spectrum monitoring and radio systems Eldemerdash et al. (2016). Existing WSR models, such as IQFormer Shao et al. (2024) and TS-DDAE Liu et al. (2026), have achieved strong classification results on standard benchmarks.

Due to the dynamic nature of the real-world spectrum environment, unknown modulation types and new protocols emerge continuously, and these signals are absent from the training Trichopoulos et al. (2022). Therefore, open set WSR becomes a crucial task in evaluating the practicality of WSR models, which requires them not only to accurately classify known signals, but also to reliably detect and reject unknown types Gong et al. (2021)Hanna et al. (2020). Despite its importance, robust open set WSR remains underexplored. A fundamental barrier is partition sensitivity: model performance and rankings fluctuate substantially across different data partitions, undermining evaluation consistency and deployment reliability. This challenge is particularly acute for wireless signals due to two signal-specific properties. First, wireless channels act as continuous physical processes, inducing multi-scale perturbations that cause substantial intra-class feature variation across varying conditions. Second, distinct signal types are often separated only by fine-grained geometric distinctions in the discriminative parameter space, rendering inter-class boundaries inherently fragile and prone to drift under subset composition changes. These properties jointly exacerbate partition sensitivity and limit the effectiveness of generic open set recognition approaches Hanna et al. (2020).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To overcome partition sensitivity, we formalize the problem as Partition-Robust Risk Minimization (PRRM). Theoretical analysis reveals that sensitivity stems from uncontrolled risk variance, which decomposes into sample-level variance induced by intra-class deviations and class-composition variance caused by changes in the known class subset. Although some approaches acknowledge signal perturbations Li et al. (2025), they do not explicitly reduce the sensitivity of learned features to these perturbations, leaving perturbation variance uncontrolled. More importantly, no existing work addresses class-composition variance. Without geometric constraints, decision boundaries drift as the known subset changes, rendering models fragile under varying partitions. Guided by PRRM, we propose DualDiff, a framework that addresses both variance sources through a two-stage design. To reduce sample-level variance, we employ manifold-regularized diffusion to construct a smooth feature manifold in the latent space. By learning to denoise perturbed features back onto this manifold, the encoder produces representations that are less sensitive to intra-class deviations while retaining class-relevant signal structure. For class-composition variance, we employ geometry-aware fine-tuning, combining angular margins with uniformity regularization to anchor class prototypes to a stable geometric configuration, thereby limiting boundary drift. Theoretical analysis shows that this design reduces both variance sources, contributing to more stable performance across partitions. Extensive experiments demonstrate that DualDiff improves the Open Set Classification Rate (OSCR) by 3.18% on average over 13 baselines, with ablations confirming the contribution of each component.

Our contributions can be summarized as follows:

- We identify partition sensitivity as a critical yet underexplored barrier in open set WSR, and formalize it as Partition-Robust Risk Minimization (PRRM). Our analysis reveals that sensitivity stems from uncontrolled risk variance, which decomposes into sample-level variance and class-composition variance.
- Guided by PRRM, we propose DualDiff, a framework that addresses both variance sources. DDualDiff employs manifold-regularized latent diffusion to construct a deviation-robust feature manifold, reducing sensitivity to intra-class deviations, and geometry-aware fine-tuning to stabilize decision boundaries against class-composition changes.
- Extensive experiments demonstrate that DualDiff achieves strong open set WSR performance, surpassing 13 baselines by an average of 3.18% in OSCR. DualDiff also maintains consistent rankings across diverse partition settings. Ablation studies confirm the contribution of each component.

2 PRELIMINARY

2.1 WIRELESS SIGNALS

Wireless signals can be typically represented as In-phase and Quadrature (IQ) data, denoted by the complex number $x = I + jQ, j = \sqrt{-1}$. In practice, IQ data of length L are stored as a two-row real matrix $x \in \mathbb{R}^{2\times L}$, where each row corresponds to the in-phase and quadrature components, respectively. In addition to the time-domain IQ representation, the frequency-domain spectrum, obtained via the Discrete Fourier Transform (DFT), i.e., $X[k] = \sum_{n=0}^{L-1} x[n] \cdot exp(-j2\pi kn/L), k = 0, 1, \cdots, L-1$, provides complementary information about the signal's frequency structure. The real and imaginary parts of the spectrum are similarly stored as a real matrix in $\mathbb{R}^{2\times L}$.

The task of WSR is to identify the attributes of a received signal, such as its modulation type, by learning a mapping: $f_\theta : \mathbb{R}^{2\times L} \to \mathcal{Y}$, where $\mathcal{Y} \in \{1, \cdots, C\}$ is the set of class labels. Current WSR models can be broadly categorized into two types. The first type is traditional deep learning methods. These methods typically utilize supervised labeling information to guide deep neural networks in learning signal representations, thereby adapting to specific downstream tasks Zhang et al. (2025a). Many CNN and Transformer-based architectures, such as AMC_Net Zhang et al. (2023) and IQFormer Shao et al. (2024), have achieved excellent performance on single tasks. However, they are not suitable for the rapidly evolving and diverse wireless signal environment due to the requirements for a large amount of labeling information. The second type is pre-training methods based on self-supervised learning. These methods utilize extensive learning from large amounts of unlabeled data to model and learn general representations of wireless signals. For example, SpectrumFM Zhou et al. (2025) adopts a "mask-reconstruction" paradigm inspired by language models to build a base model adaptable to various WSR tasks. TS-DDAE Liu et al. (2026) employs

a diffusion-based denoising objective in the original signal space for pre-training to better adapt to waveform signals with continuously varying values.

In the open set setting, the model must classify known signals while detecting unknown types that are absent from the training data. Formally, the decision rule is defined as

$$f_\theta(x) = \begin{cases} \arg\max_{y \in \mathcal{C}_K} p(y|x), & \Gamma(x) \geq \tau \\ \text{unknown}, & \Gamma(x) < \tau \end{cases}, \tag{1}$$

where $\Gamma(x) = \max_{c \in \mathcal{C}_K} P(y = c|x)$ is the confidence scoring function, τ is the discrimination threshold, and $\mathcal{C}_K$ is the set of known class labels. For evaluation, we adopt the OSCR that integrates performance over all possible thresholds and thus avoids manual threshold selection Vaze et al. (2021). While specialized open set WSR methods exist, such as Open-ICL Yang et al. (2024) and FSOS-AMC Zhang et al. (2025b), they primarily focus on unknown detection under static assumptions and do not consider partition sensitivity.

2.2 Denoising Diffusion Models

In recent years, diffusion models have become a prominent paradigm for generative modeling due to their superior generation quality and stable training characteristics Yang et al. (2023). Taking the classic diffusion model DDPM as an example, in the forward noise addition process, T steps of Gaussian noise are continuously added to the original data x_0 to form noisy data. Formally, the relationship between noisy data x_t at step t and x_0 can be directly established as $x_t = \alpha_t x_0 + \beta_t \varepsilon$, where $\alpha_t^2 + \beta_t^2 = 1, t \in (0, T]$, and $\varepsilon \sim \mathcal{N}(0, I)$. In the reverse denoising process, a learnable neural network parameterized by θ is trained, such as U-Net Ronneberger et al. (2015), to gradually denoise the data, thereby forming an optimization objective $\mathcal{L}(x_0; \theta) = ||\varepsilon - \varepsilon_\theta(x_t, t)||^2$. LDM Rombach et al. (2022) further introduces a latent diffusion mechanism, enhancing semantic representation capabilities by performing a noise addition and denoising process in a low-dimensional latent space. Although diffusion models are initially geared towards generative tasks, recent research has shown their significant potential in representation learning as well. For example, DDAE Xiang et al. (2023) and other works have verified that the denoising diffusion process can serve as an effective self-supervised pre-training objective, thus forming the "noise-reconstruction" pre-training paradigm different from the text "mask-reconstruction" Devlin et al. (2019). The learned representations exhibit excellent transfer performance in discriminative tasks such as image classification.

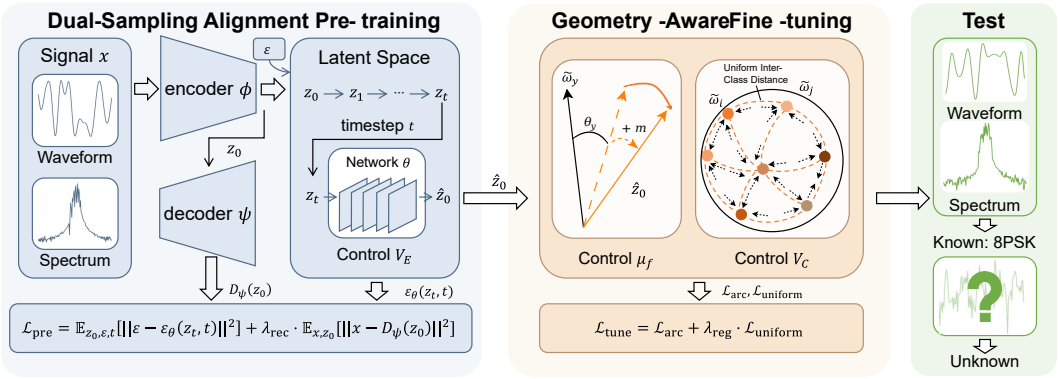


Figure 1: The overall architecture of DualDiff. DualDiff consists of two stages. In the manifold-regularized diffusion pre-training stage, the encoder maps signals to a latent space where a diffusion process constructs a smooth feature manifold, reducing sensitivity to intra-class deviations (V_E). In the geometry-aware fine-tuning stage, ArcFace and uniformity regularization stabilize class boundaries against subset composition changes (V_C) while optimizing the expected classification risk μ_f. The learned embeddings are used for close set and open set WSR during test.

In the field of wireless signal processing, the application of diffusion models is still in its early stages. RF-Diffusion Chi et al. (2024) utilizes time-frequency domain diffusion processes to generate

signal samples, primarily for data augmentation. TS-DDAE Liu et al. (2026), drawing inspiration from DDAE, constructs a diffusion model-based pre-training paradigm for WSR. However, these methods operate directly in the original signal space, which does not explicitly shape the geometry of the learned feature space, and limits its ability to learn features that are robust to channel-induced perturbations while maintaining clear separation among structurally similar signal types.

3 METHODOLOGY

To address partition sensitivity, we propose the Partition-Robust Risk Minimization (PRRM) objective, which reveals that the partition sensitivity stems from perturbation variance V_E and class-composition variance V_C. This indicates that achieving stable open set WSR performance requires reducing dual variance sources, not merely optimizing mean performance. Guided by PRRM, we introduce DualDiff,a framework that addresses dual variance through a two-stage design. As illustrated in Figure 1, DualDiff first employs the manifold-regularized latent diffusion in the pre-training stage to construct a smooth feature manifold, reducing sensitivity to sample-level perturbations and thereby suppressing V_E. Then, it employs a geometry-aware regularization to stabilize class boundaries against subset composition changes thereby suppressing V_C, while optimizing the expected classification risk μ_f. Details are listed in the following sections. The neural networks used are presented in Appendix B.

3.1 PARTITION-ROBUST RISK MINIMIZATION

For any class $c \in \mathcal{Y}$, we define the class prototype of wireless signals as $\mu_c = \mathbb{E}[x|c]$, then the sample-level perturbation is $\epsilon = x - \mu_c$, so that a received signal can be decomposed into $x = \mu_c + \epsilon$, where $\mathbb{E}[\epsilon|c] = 0$, and ϵ captures all stochastic deviations from the class-conditional mean, encompassing both the intrinsic waveform variability due to differing information content and the perturbations introduced by wireless channels. Then, we define a meta-distribution $P(S)$ over data partitions. A partition S is induced by two independent stochastic mechanisms. The first is the class subset sampling $\mathcal{C}_K \in \{\mathcal{C} \subset \mathcal{Y} : |\mathcal{C}| = K\}$, which selects the set of K known classes. The second is the sample-level deviation $\epsilon \sim P(\epsilon|c)$, which characterizes the intra-class variability for each sample. We can assume that class selection $\mathcal{C}_K$ and sample-level deviation ϵ are independent, as $\mathcal{C}_K$ is determined by operational factors such as deployment scenarios, while ϵ arises from the joint randomness of the signal source and the channel. Schmid et al. (2025). Therefore, we have $P(S) = P(\mathcal{C}_K) \cdot P(\epsilon)$.

For a model with parameter θ, its performance on partition S is $F(S;\theta) \in (0, 1]$, and the risk function is defined as $f(S;\theta) = 1 - F(S;\theta)$. We denote the expectation and variance of F as $\mu_F = \mathbb{E}[F]$ and $\sigma_F^2 = \text{Var}[F]$, respectively. To ensure that the model exhibits consistent and optimal performance across any set of partitions $\{S_i\}_{i=1}^N$, we formalize the objective as maximizing the product of performance across partitions, which is equivalent to maximizing the mean of the logarithmic performance. Formally, our objective is

$$\max_\theta \prod_{i=1}^{N} F(\theta; S_i) \iff \max_\theta \frac{1}{N} \sum_{i=1}^{N} \log F(\theta; S_i) \approx \max_\theta \mathbb{E}_{S\sim P(S)}[\log F(\theta; S)]. \tag{2}$$

Applying Taylor's formula to $g(y) = \log(y)$, we obtain its second-order expansion around $y = \mu_F$. Substituting this into $y = F$ and taking the expectation, we get

$$\mathbb{E}[\log F] = \log \mu_F - \frac{\sigma_F^2}{2\mu_F^2} + \mathbb{E}[R_3(F)], \tag{3}$$

where $R_3(F)$ is the remainder term. While the remainder is non-zero, the second-order term captures the dominant source of risk fluctuation. We adopt the PRRM objective as a principled surrogate by

$$\max_\theta (\log \mu_F - \frac{\sigma_F^2}{2\mu_F^2}) \iff \min_\theta (-\log(1 - \mu_f) + \frac{\sigma_f^2}{2(1 - \mu_f)^2}), \tag{4}$$

where $\mu_F = 1 - \mu_f$, and $\sigma_F^2 = \sigma_f^2$. From Equation 4, we observe that optimizing mean performance alone, i.e., minimizing μ_f does not guarantee consistent performance across partitions, since the variance σ_f may remain large. This explains why existing models exhibit partition-sensitive open set performance. We then apply the Law of Total Variance to decompose the risk variance and get

$$\sigma_f^2 = \text{Var}_S[f] = \mathbb{E}_{\mathcal{C}_K}[\text{Var}_\epsilon(f|\mathcal{C}_K)] + \text{Var}_{\mathcal{C}_K}(\mathbb{E}_\epsilon[f|\mathcal{C}_K]). \tag{5}$$

Defining $V_E = \mathbb{E}_{\mathcal{C}_K}[\text{Var}_\epsilon(f|\mathcal{C}_K)]$ as the sample-level variance and $V_C = \text{Var}_{\mathcal{C}_K}(\mathbb{E}_\epsilon[f|\mathcal{C}_K])$ as the class-composition variance, PRRM requires minimizing μ_f, V_E, and V_C. Reducing V_E calls for feature representations that are less sensitive to intra-class deviations while retaining class-relevant signal structure. Since wireless channels act as continuous physical processes, the induced perturbations span a continuum of magnitudes rather than discrete corruption levels, demanding a regularizer that captures deviations at all scales simultaneously. Reducing V_C requires stabilizing decision boundaries against changes in the known class subset. Since distinct signal types are separated by fine-grained geometric distinctions in the modulation parameter space rather than by coarse semantic gaps, inter-class boundaries are inherently susceptible to drift under subset changes. These two objectives motivate the two-stage design of DualDiff.

3.2 Dual-Sampling Alignment Pre-training with Diffusion

Guided by PRRM, we first address the sample-level variance V_E, which arises because intra-class deviations cause the encoder output to fluctuate around the class-level feature structure. Reducing V_E requires the encoder to learn representations that remain stable despite these deviations. In wireless signals, such deviations span a continuum of magnitudes: channel conditions vary continuously, and different information realizations within the same modulation type produce distinct waveform instances, jointly causing features to deviate smoothly and progressively in the latent space. The multi-step forward process of diffusion naturally mirrors this progressive degradation. Each noise level corresponds to a specific degree of channel-induced deviation, and the reverse process learns to project perturbed features back onto the manifold at every scale. We therefore construct a smooth feature manifold via latent diffusion-based pre-training. Combined with a reconstruction objective that preserves class-relevant information, this training encourages the encoder to concentrate features on the manifold, reducing sensitivity to perturbations and suppressing V_E.

Formally, an encoder E_ϕ maps the signal x to an initial latent space $z_0 = z = E_\phi(x)$. The forward process gradually adds T steps of Gaussian noise to z_0, yielding noisy features $z_1, z_2, \cdots, z_T$. The relationship between z_t at step t and z_0 is given by $z_t = \alpha_t z_0 + \beta_t \varepsilon$, where $\alpha_t^2 + \beta_t^2 = 1, \varepsilon \sim \mathcal{N}(0, I)$. As our objective is to construct the manifold structure instead of recovering the transmitted signal from the received one, we use the Gaussian perturbation in the forward process. In the encoder's latent space, the aggregate effect of statistically independent deviation sources approximates a Gaussian distribution by the central limit theorem, making this choice a principled surrogate for composite intra-class variability.

Subsequently, we need to establish a denoising optimization objective in the reverse process. Since our task is to maximize the marginal log-likelihood of the signal $\log P(x)$, and direct optimization is difficult, we adopt the Evidence Lower Bound (ELBO) approach and derive the optimization objective. Details can be seen in Appendix A. So the overall pre-training loss is

$$\mathcal{L}_{\text{pre}} = -\mathbb{E}_{q_\phi(z|x)}[\log P_\psi(x|z)] + D_{KL}(q_\phi(z|x)||P_\theta(z)). \tag{6}$$

A decoder D_ψ reconstructs x from z, capturing residual variations. For the first term, assuming that the decoder output follows a Gaussian distribution $P_\psi(x|z) = \mathcal{N}(D_\psi(z), \sigma^2 I)$, the negative log-likelihood is $-\log P_\psi(x|z) \propto ||x - D_\psi(z)||^2$, and taking the expectation, we get the reconstruction loss $\mathcal{L}_{\text{recon}} = \mathbb{E}_{x,z\sim q_\phi(z|x)}[||x - D_\psi(z)||^2]$. This term requires that the latent variables z retain enough information to recover the signal structure, so that the diffusion regularization does not over-smooth the latent space and erase the fine-grained features essential for discrimination. For the KL term $D_{KL}(q_\phi(z|x)||P_\theta(z))$, we can directly use the optimization result of DDPM Ho et al. (2020), expressed as $\mathcal{L}_{\text{diff}} = \mathbb{E}_{z_0,\varepsilon,t}[||\varepsilon - \varepsilon_\theta(z_t, t)||^2]$. This term trains the model to learn the density structure of the feature manifold, encouraging known-class features to concentrate on the manifold while unknown-class features fall into low-density regions. Finally, our pre-training loss function is

$$\mathcal{L}_{\text{pre}} = \mathbb{E}_{z_0,\varepsilon,t}[||\varepsilon - \varepsilon_\theta(z_t, t)||^2] + \lambda_{\text{rec}}\mathbb{E}_{x,z}[||x - D_\psi(z)||^2], \tag{7}$$

where λ_{rec} is a hyperparameter balancing manifold structure and signal reconstruction. During pre-training, we sample an IQ signal x from the dataset, a diffusion step $t \sim U(0, T)$, and a standard Gaussian noise $\varepsilon \sim \mathcal{N}(0, I)$. The latent variable $z = z_0$ is obtained through the encoder E_ϕ, and the z_t is computed by the forward noising process. The whole network is optimized according to Equation 7. Theoretically, this pre-training constrains the feature deviation induced by ϵ, satisfying $V_E \leq L_\Phi^2 \eta(\lambda_{\text{rec}})$, where $\eta(\cdot)$ decreases with the strength of the diffusion objective. The detailed proof can be found in Appendix A.

3.3 GEOMETRY-AWARE FINE-TUNING

Then, we need to address the class-composition variance V_C, which arises from decision boundary drift as the known subset $\mathcal{C}_K$ changes. In WSR, this drift is particularly pronounced because inter-class discrimination relies on fine-grained angular and amplitude separations in the complex signal plane, rather than on coarse semantic distinctions. Without explicit geometric constraints, these boundaries shift as the known subset changes, inflating V_C. To reduce V_C, we enforce geometric stability by anchoring class prototypes to a globally stable configuration on the hypersphere, thereby limiting boundary drift across partitions.

We achieve this through a fine-tuning strategy that combines angular discrimination with geometric regularization. First, we employ the ArcFace loss Deng et al. (2019), which is defined as

$$\mathcal{L}_{\text{arc}} = -\log \frac{\exp(s \cdot \cos(\angle_y + m))}{\exp(s \cdot \cos(\angle_y + m)) + \sum_{i \neq y} \exp(s \cdot \cos \angle_i)}, \tag{8}$$

where s is the scaling factor, m is the angular margin, and $\angle_i$ denotes the angle between the feature z and classifier weight w_i for class i. The ArcFace loss serves two purposes. First, it optimizes the expected risk μ_f by carving the angular separation between classes, The angular margin directly corresponds to the minimum resolvable separation between adjacent modulation states in the complex plane, ensuring that the learned feature space preserves the geometric resolution required for discrimination. Second, it enforces alignment between sample features z and their corresponding classifier weights w_c, so that the weights serve as geometric proxies for class prototypes μ_c. To reduce V_C, we further introduce a weight-based uniformity regularizer, which is defined as

$$\mathcal{L}_{\text{uniform}} = \log \sum_{i \neq j} \exp(-||\tilde{w}_i - \tilde{w}_j||^2), \tag{9}$$

where $\tilde{w}_i = w_i / ||w_i||$ is the normalized weight for class i. This defines a repulsive potential on the hypersphere whose global minimum corresponds to a Simplex Equiangular Tight Frame (ETF) structure Wang & Isola (2020). Minimizing this loss encourages the weights of any subset $\mathcal{C}_K$ to align with this canonical configuration. This anchoring is essential in WSR, where unknown carrier phase offsets and frequency drifts introduce global rotations in the signal representation; a stable reference configuration ensures that decision boundaries remain invariant to such channel-induced geometric transformations. Due to the weight-prototype alignment enforced by ArcFace, this regularization stabilizes the positions of class prototypes on the feature manifold, limiting boundary drift induced by subset changes. Theoretical analysis shows that V_C decreases as the deviation of subset weights from the global configuration decreases, details can be seen in Appendix A.

Combining the two objectives, we get the final fine-tuning loss function

$$\mathcal{L}_{\text{tune}} = \mathcal{L}_{\text{arc}} + \lambda_{\text{reg}} \cdot \mathcal{L}_{\text{uniform}}, \tag{10}$$

where λ_{reg} balances discriminative power and geometric stability. During fine-tuning, the encoder maps the signal x to a latent feature z_0, a small amount of noise is added according to the forward process, and the denoising network produces the final embedding $\hat{z}_0$. This embedding, together with the label y, are used to compute the loss in Equation 10. Through the combination of manifold-regularized diffusion pre-training and geometry-aware fine-tuning, DualDiff reduces both V_E and V_C while optimizing the expected risk μ_f, addressing all three terms in the PRRM objective.

4 EXPERIMENT

4.1 EXPERIMENTAL SETTINGS

DualDiff is pre-trained and fine-tuned on Tesla A100 with PyTorch Paszke et al. (2019) for implementation. The hyper-parameters are optimized with the Optuna Akiba et al. (2019), whose detailed values are listed in Appendix E. The other implementation details can be found in Appendix D.

Datasets. In our experiments, we use three datasets that cover two typical WSR tasks. They are RML2016.10A O'shea & West (2016), RML2022 Sathyanarayanan et al. (2023) for Automatic Modulation Classification (AMC) Meng et al. (2018), and RadChar Huang et al. (2023) for Wireless Technology Classification (WTC) Bitar et al. (2017). These three datasets contain several classes

Table 1: The partition information used in our experiments.

Dataset	Setting	Known Class	Unknown Class
RML2016.10A	0	8PSK, AM-DSB, AM-SSB, BPSK, CPFSK, GFSK	PAM4, QAM16, QAM64, QPSK, WBFM
	1	BPSK, QPSK, AM-DSB, AM-SSB, WBFM	GFSK, PAM4, CPFSK, QAM16, QAM64, 8PSK
	2	BPSK, QPSK, AM-DSB, WBFM, QAM16, PAM4	GFSK, CPFSK, AM-SSB, QAM64, 8PSK
RML2022	0	8PSK, AM-DSB, AM-SSB, BPSK, CPFSK, GFSK	PAM4, QAM16, QAM64, QPSK, WBFM
	1	BPSK, QPSK, AM-DSB, AM-SSB, WBFM	GFSK, PAM4, CPFSK, QAM16, QAM64, 8PSK
RadChar	0	coherent_pulse_train, linear_frequency_modulated	barker_code, frank_code, polyphase_barker_code

(modulation and wireless technology types) under various Signal-to-Noise Ratio (SNR) conditions. Details about tasks and datasets are listed in Appendix C. To evaluate open set WSR performance, we partition RML2016.10A, RML2022, and RadChar into 6 settings by random select known classes. See Table 1 for detailed partition information. The *Setting-0* and *Setting-1* in RML2016.10A are used to verify performance inconsistencies under different partitioning conditions. The *Setting-0* and *Setting-2* in RML2016.10A aim to demonstrate that even with the same number of known/unknown classes, the performance of baseline models still vary. The *Setting-0* and *Setting-1* of RML2016.10A and RML2022 are the same to illustrate that even within the same category, different data distributions still result in inconsistencies. For known classes, we use 80% for training and 20% for testing.

The evaluation metric used in our experiments is the Open Set Classification Rate (OSCR), which is a comprehensive metric that measures the closed-set and open set performance without manually selecting threshold, and is widely used in computer vision open set recognition Vaze et al. (2021). In the following results, we present the OSCR results of all test samples (denoted as *Overall*), and the best results across samples at each SNR level (denoted as *Best*).

Baselines. We use 13 baselines following the setting of TS-DDAE Liu et al. (2026), including AMC_Net Zhang et al. (2023), CGDNN Njoku et al. (2021), CNN2 O'Shea et al. (2018), GRU2 Hong et al. (2017), MCNet Huynh-The et al. (2020), MSNet Zhang et al. (2021), ResNet Liu et al. (2017), Transformer Vaswani et al. (2017), VGG O'Shea et al. (2018), IQFormer Shao et al. (2024), FSOS_AMC Zhang et al. (2025b), SpectrumFM Zhou et al. (2025), and TS-DDAE. The first 10 models are traditional supervised training models, the FSOS_AMC is a model specially designed for open set WSR, and the SpectrumFM and TS-DDAE are following the pre-training and fine-tuning paradigm based on their original pipelines.

4.2 Main Results

In this section, we give the comparison of the results between the DualDiff and the other 13 baselines. The OSCR results are listed in Table 2. The additional results are presented in Appendix F. The results demonstrate that DualDiff achieves SOTA performance with superior partition robustness. We analyze the findings along three dimensions of sensitivity targeted by our experimental design.

Robustness to Partition Shifts. Comparing *Setting-0* and *Setting-1* on RML2016.10A reveals severe partition sensitivity in baselines: performance and relative rankings fluctuate drastically. This confirms that standard models overfit to specific partitions and lack generalization. In contrast, DualDiff maintains consistent top-1 rankings and stable performance across both settings, validating its effectiveness in suppressing sample-level variance and achieving partition robustness.

Robustness to Class Composition Shifts. *Setting-0* and *Setting-2* on RML2016.10A share identical known/unknown class counts but differ in class composition. Baselines exhibit substantial perfor-

Table 2: The overall OSCR of different models (percentage). The best performance is represented in **bold**, and the second-best performance is underlined.

Model	RML2016.10A						RML2022				RadChar	
	Setting-0		Setting-1		Setting-2		Setting-0		Setting-1		Setting-0	
	Overall	Best	Overall	Best	Overall	Best	Overall	Best	Overall	Best	Overall	Best
AMC_Net	40.37	54.59	26.87	27.19	41.97	50.82	53.66	69.20	41.65	43.44	73.87	93.15
CGDNN	28.81	27.85	26.67	25.83	47.34	69.12	46.93	68.17	39.51	39.72	69.21	84.87
CNN2	33.77	39.22	27.10	31.05	45.68	65.97	48.32	51.95	44.90	47.46	65.45	75.68
GRU2	36.01	43.34	29.25	31.27	47.25	62.80	47.14	49.61	38.04	37.22	71.31	80.62
MCNet	30.09	38.49	28.76	28.96	8.110	9.070	53.69	69.24	49.01	56.14	69.29	81.95
MSNet	34.71	41.50	25.37	23.59	38.99	44.64	48.09	53.92	39.22	41.30	72.13	86.67
ResNet	37.92	57.12	29.04	31.75	43.57	59.35	45.50	67.59	37.86	55.35	40.81	45.05
Transformer	37.99	49.20	25.35	23.94	43.78	55.00	48.85	63.18	36.12	37.95	64.35	92.58
VGG	33.77	39.09	32.85	40.77	44.90	61.60	48.86	59.35	38.61	39.61	71.79	87.24
IQFormer	36.24	41.84	29.30	29.87	45.93	57.49	48.57	52.12	40.92	40.51	57.50	67.64
FSOS_AMC	31.90	45.78	21.26	28.30	30.48	50.14	54.73	68.31	48.63	55.79	77.27	90.41
SpectrumFM	37.92	50.25	24.74	22.97	41.57	52.41	51.09	57.83	40.07	43.10	61.53	68.57
TS-DDAE	35.58	54.37	31.84	41.81	46.20	59.58	42.73	51.32	39.41	42.89	80.47	96.44
DualDiff	**46.90**	**70.25**	**37.15**	**47.11**	**51.61**	**71.83**	**57.06**	**69.34**	**49.98**	**56.27**	**81.17**	**98.58**

mance variations between these settings, indicating sensitivity to subset composition rather than merely class counts. This instability arises from uncontrolled class subset variance, where decision boundaries drift with subset changes. DualDiff mitigates this via geometry-aware fine-tuning and ensures consistent performance regardless of subset composition.

Robustness to Distribution Shifts. RML2016.10A and RML2022 share class subsets but differ in data distributions. Baselines show inconsistent transferability, with performance gaps varying significantly across datasets, highlighting distribution-induced fragility. DualDiff generalizes effectively across both distributions, achieving SOTA results on RML2022 while maintaining stability. This shows that the diffusion alignment successfully decouples semantics from distribution-specific perturbations, enabling robust recognition under various conditions.

Overall Performance. Aggregating across all settings, DualDiff outperforms 13 baselines by an average of 3.18% in OSCR. Compared to a specific baseline such as AMC_Net, DualDiff achieves at least a 7.58% improvement in OSCR performance. Compared to the diffusion-style pre-trained model TS-DDAE, DualDiff achieves a performance improvement of 7.94%, further demonstrating its strong unknown class recognition capability. DualDiff maintains the best OSCR performance while the other baselines exhibit severe ranking inconsistencies, further demonstrating the partition sensitivity problem of the current models. By optimizing PRRM, our DualDiff exhibits stable and excellent OSCR performance.

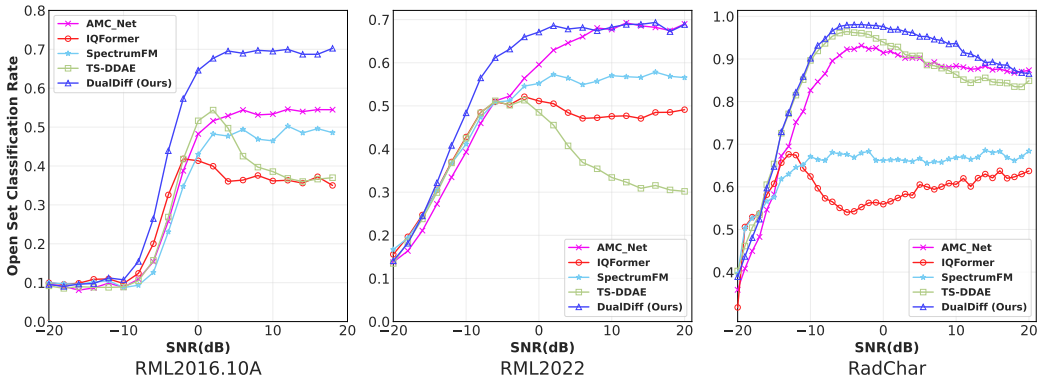


Figure 2: The OSCR performance curves under different SNR conditions (left: RML2016.10A, center: RML2022, right: RadChar).

Performance under Various SNR Conditions. We plot the results of several methods including AMC_Net, IQFormer, SpectrumFM, TS-DDAE and DualDiff, under various SNR conditions in Figure 2 (all using *Setting-0* as an example). It can be seen that under extremely low SNR conditions (-20dB), the noise is much stronger than the signal itself, making model evaluation difficult. Under low SNR conditions (-10dB to 0dB), DualDiff's open set discrimination capability is still higher than that of other models. This further demonstrates DualDiff's adaptability to different environments. For the RadChar dataset, we observe that models perform well at low SNR, and when the SNR is high, these models perform worse. We attribute this to the fact that the RadChar dataset's known class contains *linear_frequency_modulated* (LFM), and the unknown class contains *frank_code*. Frank code is a polyphase coded signal, designed to approximate linear frequency modulation through discrete step frequency variations. From a signal processing perspective, Frank code is a discrete polyphase approximation of LFM. The two share a high degree of similarity Cunxiang et al. (2021). As the SNR increases, the frequency structure of Frank codes becomes clearer and sharper, fully exposing their similarity to LFM. The features extracted by the model cannot effectively distinguish between the two, leading to Frank samples being misclassified as LFM with high confidence. This significantly decreases the accuracy of identifying unknowns, thereby lowering the OSCR. Therefore, for signals, different training strategies need to be considered for different data distributions.

Table 3: The ablation study (percentage) of DualDiff.

Model	RML2016.10A						RML2022				RadChar	
	setting-0		setting-1		setting-2		setting-0		setting-1		setting-0	
	Overall	Best	Overall	Best	Overall	Best	Overall	Best	Overall	Best	Overall	Best
DualDiff	**46.90**	**70.25**	**37.15**	**47.11**	**51.61**	**71.83**	**57.06**	**69.34**	**49.98**	**56.27**	**81.17**	**98.58**
w/o recon	43.90	65.38	34.19	42.13	50.68	70.76	47.10	54.33	47.82	54.33	69.82	92.74
w/o pretrain	42.38	59.80	31.67	36.88	49.57	71.00	41.48	51.29	49.60	55.72	64.49	84.59
only CE loss	42.06	60.09	27.17	25.60	47.32	64.34	50.41	65.51	35.72	42.08	74.31	97.07
w/o uniform	45.50	68.16	37.01	46.33	47.28	61.75	53.77	67.48	39.82	40.33	76.10	96.77

4.3 Ablation Study

In this section, we evaluate the contribution of each component via ablation studies. We consider four variants: (1) *w/o recon*, removing the reconstruction loss; (2) *w/o pretrain*, fine-tuning without pre-training; (3) *only CE loss*, replacing fine-tuning objectives with cross-entropy; and (4) *w/o uniform*, removing the uniformity regularizer. As shown in Table 3, removing any component degrades performance, confirming the synergistic effectiveness of the full DualDiff design. The substantial drop in *w/o pretrain* validates the diffusion alignment's role in suppressing sample-level variance, while the gap with *w/o recon* indicates that reconstruction is vital for preserving semantic fidelity during denoising. Furthermore, the inferior results of *only CE loss* and *w/o uniform* underscore the necessity of geometry-aware fine-tuning. ArcFace and uniformity regularization jointly stabilize decision boundaries and carve explicit rejection regions, capabilities that standard classification losses lack. Additional ablation results are provided in Appendix F.4.

5 Conclusion

In this work, we address the challenge of partition sensitivity in open set WSR, where model performance and rankings fluctuate across different data partitions. We formalize this challenge as Partition-Robust Risk Minimization (PRRM), which reveals that sensitivity stems from sample-level variance V_E and class-composition variance V_C. Guided by PRRM, we propose DualDiff, a framework that addresses both variance sources through a two-stage design. In the pre-training stage, manifold-regularized latent diffusion constructs a smooth feature manifold tailored to the continuous intra-class deviations of wireless signals, effectively suppressing V_E. In the fine-tuning stage, geometry-aware regularization anchors class prototypes to resolve the fine-grained geometric ambiguities inherent in signal spaces, thereby stabilizing boundaries and reducing V_C. Extensive experiments across several datasets and multiple data partitions demonstrate that DualDiff achieves strong OSCR performance, maintaining stable top rankings across diverse partitions. DualDiff thus establishes a theoretically sound and reliable training method for open set WSR.

AI USE STATEMENT

In this work, we used generative AI tools to assist in the writing of proofs and translation. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to edit software code and edit a research paper to improve readability. We have reviewed all AI-assisted work by carefully reading and correcting the AI-generated code and content. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This research strictly adheres to data usage regulations. All experiments are based on public datasets, with a commitment not to process any private information. However, WSR can support beneficial applications (e.g., spectrum monitoring, anomaly detection) but may also enable surveillance or signal intelligence. Deployment should comply with local laws and be restricted in contexts that infringe on privacy or civil liberties. We encourage the academic community to work together to enhance the understanding and implementation of responsible AI.

REPRODUCIBILITY STATEMENT

To facilitate the reproduction of our results, Section 4.1 and the Appendix E provide a thorough description of our experimental setup, and hyperparameters. The source code and scripts will be made publicly available upon this paper's acceptance. We have also listed all required external libraries and the usage of our code repository. We provide the demo of our repository in the supplementary materials and promise to continue to develop and maintain it in the future.

REFERENCES

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2623–2631, 2019.

Mohammed H Alsharif, Anabi Hilary Kelechi, Mahmoud A Albreem, Shehzad Ashraf Chaudhry, M Sultan Zia, and Sunghwan Kim. Sixth generation (6g) wireless networks: Vision, research activities, challenges and potential solutions. *Symmetry*, 12(4):676, 2020.

Naim Bitar, Siraj Muhammad, and Hazem H Refai. Wireless technology identification using deep convolutional neural networks. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–6. IEEE, 2017.

Guoxuan Chi, Zheng Yang, Chenshu Wu, Jingao Xu, Yuchong Gao, Yunhao Liu, and Tony Xiao Han. Rf-diffusion: Radio signal generation via time-frequency diffusion. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, pp. 77–92, 2024.

Xie Cunxiang, Zhang Limin, and Zhong Zhaogen. Quasi-lfm radar waveform recognition based on fractional fourier transform and time-frequency analysis. *Journal of Systems Engineering and Electronics*, 32(5):1130–1142, 2021.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Yahia A Eldemerdash, Octavia A Dobre, and Mengüç Öner. Signal identification for multiple-antenna wireless systems: Achievements and challenges. *IEEE Communications Surveys & Tutorials*, 18 (3):1524–1551, 2016.

Jialiang Gong, Xiaowei Qin, and Xiaodong Xu. Multi-task based deep learning approach for open-set wireless signal identification in ism band. *IEEE Transactions on Cognitive Communications and Networking*, 8(1):121–135, 2021.

Samer Hanna, Samurdhi Karunaratne, and Danijela Cabric. Open set wireless transmitter authorization: Deep learning approaches and dataset considerations. *IEEE Transactions on Cognitive Communications and Networking*, 7(1):59–72, 2020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Dehua Hong, Zilong Zhang, and Xiaodong Xu. Automatic modulation classification using recurrent neural networks. In *2017 3rd IEEE international conference on computer and communications (ICCC)*, pp. 695–700. IEEE, 2017.

Zi Huang, Akila Pemasiri, Simon Denman, Clinton Fookes, and Terrence Martin. Multi-task learning for radar signal characterisation. In *Proceedings of the 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*, pp. 1–5, 2023. doi: 10.1109/ICASSPW59220.2023.10193318.

Thien Huynh-The, Cam-Hao Hua, Quoc-Viet Pham, and Dong-Seong Kim. Mcnet: An efficient cnn architecture for robust automatic modulation classification. *IEEE Communications Letters*, 24(4): 811–815, 2020.

Chenxi Li, Changxiong Xia, and Qun Wan. Specific emitter identification based on feature fusion. In *2025 17th International Conference on Signal Processing Systems (ICSPS)*, pp. 1241–1245. IEEE, 2025.

Xiaofan Li, Fangwei Dong, Sha Zhang, and Weibin Guo. A survey on deep learning techniques in wireless signal recognition. *Wireless Communications and Mobile Computing*, 2019(1):5629572, 2019.

Xiaoyu Liu, Diyu Yang, and Aly El Gamal. Deep neural network architectures for modulation classification. In *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pp. 915–919. IEEE, 2017.

Yaoqi Liu, Jin Wang, Hui Wang, and Chuan Shi. TS-DDAE: A novel temporal-spectral denoising diffusion autoencoder for wireless signal recognition model pre-training. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RKDkqkkZ5m.

Fan Meng, Peng Chen, Lenan Wu, and Xianbin Wang. Automatic modulation classification: A deep learning enabled approach. *IEEE Transactions on Vehicular Technology*, 67(11):10760–10772, 2018.

Judith Nkechinyere Njoku, Manuel Eugenio Morocho-Cayamcela, and Wansu Lim. Cgdnet: Efficient hybrid deep learning model for robust automatic modulation recognition. *IEEE Networking Letters*, 3(2):47–51, 2021.

Timothy J O'shea and Nathan West. Radio machine learning dataset generation with gnu radio. In *Proceedings of the GNU radio conference*, volume 1, 2016.

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

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Venkatesh Sathyanarayanan, Peter Gerstoft, and Aly El Gamal. Rml22: Realistic dataset generation for wireless modulation classification. *IEEE Transactions on Wireless Communications*, 22(11): 7663–7675, 2023.

Luca Schmid, Tomer Raviv, Nir Shlezinger, and Laurent Schmalen. Blind channel estimation and joint symbol detection with data-driven factor graphs. *IEEE Transactions on Communications*, 2025.

Mingyuan Shao, Dingzhao Li, Shaohua Hong, Jie Qi, and Haixin Sun. Iqformer: A novel transformer-based model with multi-modality fusion for automatic modulation recognition. *IEEE Transactions on Cognitive Communications and Networking*, 2024.

Georgios C Trichopoulos, Panagiotis Theofanopoulos, Bharath Kashyap, Aditya Shekhawat, Anuj Modi, Tawfik Osman, Sanjay Kumar, Anand Sengar, Arkajyoti Chang, and Ahmed Alkhateeb. Design and evaluation of reconfigurable intelligent surfaces in real-world environment. *IEEE Open Journal of the Communications Society*, 3:462–474, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *International conference on learning representations*, 2021.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pp. 9929–9939. PMLR, 2020.

Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15802–15812, 2023.

Chen Yang, Zhixi Feng, Shuyuan Yang, and Qiukai Pan. Open-icl: Open-set modulation classification via incremental contrastive learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(5):8853–8866, 2024.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM computing surveys*, 56(4):1–39, 2023.

Hao Zhang, Fuhui Zhou, Qihui Wu, Wei Wu, and Rose Qingyang Hu. A novel automatic modulation classification scheme based on multi-scale networks. *IEEE Transactions on Cognitive Communications and Networking*, 8(1):97–110, 2021.

Hao Zhang, Fuhui Zhou, Hongyang Du, Qihui Wu, and Chau Yuen. Revolution of wireless signal recognition for 6g: Recent advances, challenges and future directions. *IEEE Communications Surveys & Tutorials*, 2025a.

Hao Zhang, Fuhui Zhou, Qihui Wu, and Chau Yuen. Fsos-amc: Few-shot open-set learning for automatic modulation classification over multipath fading channels. *IEEE Internet of Things Journal*, 2025b.

Jiawei Zhang, Tiantian Wang, Zhixi Feng, and Shuyuan Yang. Amc-net: An effective network for automatic modulation classification. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Fuhui Zhou, Chunyu Liu, Hao Zhang, Wei Wu, Qihui Wu, Tony QS Quek, and Chan-Byoung Chae. Spectrumfm: A foundation model for intelligent spectrum management. *IEEE Journal on Selected Areas in Communications*, 2025.

A Additional Proof

In this section, we provide the detailed proof of the theory proposed in Section 3. First, we give the ELBO derivation of the diffusion model for the optimization objective. Then, we demonstrate that the two-stage optimization strategy effectively suppresses both perturbation variance and class-composition variance, thereby minimizing the Partition-Robust Risk. Based on Equations 4 and 5, we know that the model should optimize the μ_f, V_E, and V_C to achieve stable open set WSR performance.

A.1 ELBO Derivation

Starting from the marginal log-likelihood $\log P(x)$, we introduce the variational posterior $q_\phi(z|x)$ (encoder) to derive the Evidence Lower Bound (ELBO) by

$$\log P(x) = \log \int P_\psi(x|z)P_\theta(z)dz = \log \int q_\phi(z|x)\frac{P_\psi(x|z)P_\theta(z)}{q_\phi(z|x)}dz. \tag{11}$$

Based on Jensen's inequality, we can obtain

$$\begin{aligned}\log P(x) &\geq \int q_\phi(z|x)\log\frac{P_\psi(x|z)P_\theta(z)}{q_\phi(z|x)}dz \\ &= \int q_\phi(z|x)[\log P_\psi(x|z) + \log P_\theta(z) - \log q_\phi(z|x)] \\ &= \mathbb{E}_{q_\phi(z|x)}[\log P_\psi(x|z)] - D_{KL}(q_\phi(z|x)||P_\theta(z)).\end{aligned} \tag{12}$$

Here, $q_\phi(z|x)$ is the encoder distribution mapping signals to latent space, $P_\psi(x|z)$ is the decoder likelihood, and $P_\theta(z)$ is the diffusion prior over latent representations. The KL term enforces the latent code to follow the prior manifold. In practice, we replace the explicit KL minimization with the DDPM denoising score matching objective, which learns $P_\theta(z)$ by reversing perturbation accumulation. This yields the diffusion loss $\mathcal{L}_{\text{diff}} = \mathbb{E}[\|\epsilon - \epsilon_\theta(z_t, t)\|^2]$, effectively bounding the KL divergence via score matching. The final pre-training loss combines both terms: $\mathcal{L}_{\text{pre}} = \mathcal{L}_{\text{diff}} + \lambda_{\text{rec}}\mathcal{L}_{\text{recon}}$.

A.2 Bounding Sample-level Variance V_E

We first derive an upper bound on V_E in terms of the feature deviation induced by intra-class deviations, and then show how the pre-training objective constrains this deviation.

From variance to feature deviation. For a fixed subset $\mathcal{C}_K$ with fixed class c and class prototype μ_c, let $Y = f(\mu_c + \epsilon)$. Applying the inequality $\text{Var}(Y) \leq \mathbb{E}[(Y-a)^2]$ for any constant a, choose $a = f(\mu_c)$, we have

$$\text{Var}_\epsilon[f \mid \mathcal{C}_K] \leq \mathbb{E}_\epsilon[(f(\mu_c+\epsilon) - f(\mu_c))^2 \mid \mathcal{C}_K]. \tag{13}$$

Assuming that the risk function f is L_Φ-Lipschitz with respect to the encoder output $\Phi(x)$, we obtain

$$(f(\mu_c+\epsilon) - f(\mu_c))^2 \leq L_\Phi^2\|\Phi(\mu_c+\epsilon) - \Phi(\mu_c)\|^2. \tag{14}$$

Taking expectation over ϵ, we get

$$\text{Var}_\epsilon[f \mid \mathcal{C}_K] \leq \mathbb{E}_\epsilon[(f(\mu_c+\epsilon) - f(\mu_c))^2 \mid \mathcal{C}_K] \leq L_\Phi^2\mathbb{E}_\epsilon[\|\Phi(\mu_c+\epsilon) - \Phi(\mu_c)\|^2 \mid \mathcal{C}_K]. \tag{15}$$

Finally, taking expectation over $\mathcal{C}_K$, we obtain

$$V_E = \mathbb{E}_{\mathcal{C}_K}[\text{Var}_\epsilon[f \mid \mathcal{C}_K]] \leq L_\Phi^2\mathbb{E}_{\mathcal{C}_K,\epsilon}[\|\Phi(\mu_c+\epsilon) - \Phi(\mu_c)\|^2]. \tag{16}$$

Here, L_Φ is the Lipschitz constant of the encoder Φ, which measures the sensitivity of the encoder output to input deviations. For neural networks with bounded weights (ensured by weight decay) and smooth activation functions, L_Φ exists and is finite.

Bounding feature deviation via the pre-training objective. Let Φ^* be the optimal encoder obtained from pre-training with the loss defined in Equation 7. Define the feature perturbation deviation as $\Delta z = \Phi^*(\mu_c+\epsilon) - \Phi^*(\mu_c)$. We compare Φ^* with a hypothetical variant encoder $\tilde{\Phi}$ that maps all perturbed signals of class c to the same feature, i.e., $\tilde{\Phi}(\mu_c+\epsilon) = \tilde{\Phi}(\mu_c)$ for all ϵ, so that $\Delta z = 0$ under $\tilde{\Phi}$.

Here, we have two observations. First, encoding perturbation information into the latent code can reduce the reconstruction loss by at most a finite gain $G_{\max}$, i.e., $L_{\text{recon}}(\tilde{\Phi}) - L_{\text{recon}}(\Phi^*) \leq G_{\max}$. This assumption holds when the perturbation ϵ has finite second moments and the signal space is bounded, which is satisfied in practical WSR settings. Second, encoding perturbation information forces the latent feature away from the feature manifold $\mathcal{M}_z$, incurring a diffusion penalty. Specifically, assuming that the diffusion model learns an energy function $E(z) = -\log p_\theta(z)$ that grows approximately quadratically with the distance to the manifold in a local neighborhood, we have the lower bound $L_{\text{diff}}(\Phi^*) \geq \kappa \cdot \mathbb{E}[\|\Delta z\|^2]$, where $\kappa > 0$ is a curvature constant characterizing the local geometry of the feature manifold. For the variant encoder $\tilde{\Phi}$, since $\Delta z = 0$, the features lie on the manifold, yielding $L_{\text{diff}}(\tilde{\Phi}) \approx 0$.

By the optimality of Φ^* with respect to the pre-training loss, we have $L_{\text{pre}}(\Phi^*) \leq L_{\text{pre}}(\tilde{\Phi})$. Substituting the loss decomposition $L_{\text{pre}} = L_{\text{diff}} + \lambda_{\text{rec}} L_{\text{recon}}$, we obtain

$$L_{\text{diff}}(\Phi^*) + \lambda_{\text{rec}} L_{\text{recon}}(\Phi^*) \leq L_{\text{diff}}(\tilde{\Phi}) + \lambda_{\text{rec}} L_{\text{recon}}(\tilde{\Phi}). \tag{17}$$

Rearranging terms and applying the bounds yields

$$\kappa \cdot \mathbb{E}[\|\Delta z\|^2] \leq \lambda_{\text{rec}} \left(L_{\text{recon}}(\tilde{\Phi}) - L_{\text{recon}}(\Phi^*) \right) \leq \lambda_{\text{rec}} G_{\max}. \tag{18}$$

Therefore,

$$\mathbb{E}[\|\Delta z\|^2] \leq \frac{\lambda_{\text{rec}} G_{\max}}{\kappa}. \tag{19}$$

Substituting this into Equation 16, we obtain:

$$V_E \leq L_\Phi^2 \cdot \frac{\lambda_{\text{rec}} G_{\max}}{\kappa} = L_\Phi^2 \eta(\lambda_{\text{rec}}). \tag{20}$$

Interpretation of the bound. This result establishes a monotonic relationship between the pre-training hyperparameters and the sample-level variance. V_E increases with λ_{rec}, which encourages the encoder to retain deviation details for reconstruction. Moreover, V_E decreases with the strength of the diffusion objective captured by κ. This reveals a trade-off that increasing λ_{rec} preserves more signal information for lower expected risk μ_f, but at the cost of higher perturbation variance.

We note that this bound is intended as a qualitative characterization rather than a practical quantitative guarantee. The constants L_Φ, $G_{\max}$, and κ depend on the specific network architecture, data distribution, and optimization trajectory, and are generally difficult to estimate precisely in closed form. The value of this analysis lies in identifying the structural relationship between the two loss terms and the resulting variance, and in providing principled guidance for hyperparameter selection. Empirical validation of this trade-off is provided in the hyperparameter analysis in Appendix F.4.

A.3 Bounding Class Subset Variance V_C

We first prove that the uniformity loss has a unique global minimum corresponding to the Simplex ETF configuration, and then show how this geometric structure constrains V_C.

The uniformity loss is minimized at the Simplex ETF configuration. As defined in Equation 9, $||w||^2 = 1$, so $||\tilde{w}_i - \tilde{w}_j||^2 = 2 - 2\tilde{w}_i^T \tilde{w}_j$. Minimizing $\mathcal{L}_{\text{uniform}}$ is equivalent to minimizing the function

$$\sum_{i \neq j} \exp(2\tilde{w}_i^T \tilde{w}_j). \tag{21}$$

The function $h(x) = \exp(2x)$ is strictly convex. By Jensen's inequality, for a fixed sum of inner products $\sum_{i \neq j} \tilde{w}_i^T \tilde{w}_j$, the function is minimized when all pairwise inner products are equal:

$$\sum_{i \neq j} \exp(2\tilde{w}_i^T \tilde{w}_j) \geq K(K-1) \exp\left(\frac{2 \cdot \sum_{i \neq j} \tilde{w}_i^T \tilde{w}_j}{K(K-1)}\right), \tag{22}$$

with equality if and only if $\tilde{w}_i^T \tilde{w}_j = \alpha$ for all $i \neq j$. Consider the squared norm of the sum of weights

$$\|\sum_{i=1}^{K} \tilde{w}_i\|^2 = \sum_{i=1}^{K} ||w_i||^2 + \sum_{i \neq j} \tilde{w}_i^T \tilde{w}_j = K + \sum_{i \neq j} \tilde{w}_i^T \tilde{w}_j \geq 0. \tag{23}$$

This implies $\sum_{i\neq j}\tilde{w}_i^T\tilde{w}_j \geq -K$. Substituting into the equiangular condition $\sum_{i\neq j}\tilde{w}_i^T\tilde{w}_j = K(K-1)\alpha$, we obtain the lower bound as

$$\alpha \geq -\frac{1}{K-1}. \tag{24}$$

Since $\sum_{i\neq j}\tilde{w}_i^T\tilde{w}_j$ is monotonically increasing with α, the global minimum is achieved at the smallest feasible α. Thus, the optimum satisfies

$$\alpha^* = -\frac{1}{K-1}, \tag{25}$$

where $\sum_{i=1}^{C}\tilde{w}_i = 0$. This configuration corresponds exactly to a Simplex ETF, which exists if and only if $d \geq K-1$. Under this condition, the weights are maximally separated and centered, completing the proof.

Bounding V_C via weight deviation. Let $g(\mathcal{C}_K) = \mathbb{E}_\epsilon[f|\mathcal{C}_K]$ be the average risk depending on weights $W_{\mathcal{C}_K}$. Let $W^* = \{w_c^*\}_{c\in\mathcal{Y}}$ be the global optimal weight configuration minimizing the uniformity loss over all classes. For a partition with known class subset $\mathcal{C}_K$, we define $W^*_{\mathcal{C}_K} = \{w_c^*\}_{c\in\mathcal{C}_\mathcal{K}}$ as the restriction of W^* to the subset. Then we can define the reference risk $\overline{g}_K = g(W^*_{\mathcal{C}_K})$, which is constant due to the symmetry of W^*. Using $\mathrm{Var}(Y) \leq \mathbb{E}[(Y-a)^2]$ with constant $a = \overline{g}_K$, we have

$$V_C \leq \mathbb{E}_{\mathcal{C}_K}[(g(W_{\mathcal{C}_K}) - \overline{g}_K)^2] = \mathbb{E}_{\mathcal{C}_K}[(g(W_{\mathcal{C}_K}) - g(W^*_{\mathcal{C}_K}))^2]. \tag{26}$$

Applying the Lipschitz property with respect to weights, then

$$(g(W_{\mathcal{C}_K}) - g(W^*_{\mathcal{C}_K}))^2 \leq L_W^2||W_{\mathcal{C}_K} - W^*_{\mathcal{C}_K}||^2. \tag{27}$$

Taking expectation, and we can have

$$V_C \leq L_W^2 \cdot \mathbb{E}_{\mathcal{C}_K}[||W_{\mathcal{C}_K} - W^*_{\mathcal{C}_K}||^2] \tag{28}$$

From weight deviation to prototype stability. The uniformity loss drives weights toward the Simplex ETF structure. Since this potential is strongly convex on the quotient space of configurations, increasing the regularization strength λ_{reg} exerts a stronger force pulling weights toward the ETF minimum. This yields a weight deviation bound $||W_{\mathcal{C}_K} - R\cdot W^*_{\mathcal{C}_K}|| \leq \delta(\lambda_{\text{reg}})$, where R is a rotation matrix and $\delta(\cdot)$ decreases with regularization strength λ_{reg}.

The rotational ambiguity is resolved by the ArcFace loss, which aligns sample features with their corresponding classifier weights through angular margins. This alignment encourages the class $\mu_c = \mathbb{E}[z|c]$ converges to the direction of w_c, with an alignment error bounded by the margin and optimization convergence $||\mu_c - w_c|| \leq \epsilon_a$, where ϵ_a depends on the angular margin m and optimization convergence. The total deviation of prototypes from the global structure is therefore bounded by

$$||\mu_{\mathcal{C}_K} - \mu^*_{\mathcal{C}_K}|| \leq ||W_{\mathcal{C}_K} - W^*_{\mathcal{C}_K}|| + ||\mu_{\mathcal{C}_K} - W_{\mathcal{C}_K}||. \tag{29}$$

Combining the weight deviation bound and the alignment error, the class-composition variance satisfies

$$V_C \leq L_W^2 \cdot (\delta(\lambda_{\text{reg}}) + \epsilon_a)^2. \tag{30}$$

Interpretation of the bound. This result establishes a monotonic relationship between the regularization strength and the class-composition variance V_C decreases as λ_{reg} increase ($\delta(\cdot)$ is monotonically decreasing), and as the ArcFace alignment improves (ϵ_a decreases with better convergence). The constants L_W, $\delta(\lambda_{\text{reg}})$, and ϵ_a depend on the network architecture, optimization trajectory, and data distribution, and are generally difficult to estimate precisely.

The bounds rely on the Lipschitz continuity of the risk function with respect to weights. Since neural networks with bounded weights and smooth activation functions are Lipschitz continuous, the constant L_W exists and is finite.

Overall, the ArcFace loss limits the μ_f, the pre-training loss limits the V_E and the uniformity loss limits the V_C. This proves DualDiff minimizes the partition-robust risk of Equation 4.

B NEURAL NETWORK ARCHITECTURE

In our work, we need to introduce three networks. First is the encoder E_ϕ. Since our task is recognition, rather than generative, we do not need a particularly robust generative encoder-decoder design (like VAE), so we simplify the network here. Simultaneously, because IQ data contain both temporal and spectral information of the signal. Feature extraction from only the IQ data that describe the temporal relationship is insufficient. Therefore, we need to extract the IQ time series x and the spectrum X of the signal simultaneously. To extract features of both modalities, we use one-dimensional convolution (Conv1D), and perform computation in the real domain. This maintains computational efficiency while preserving the data relationships in the complex domain. This process can be represented as:

$$\boldsymbol{x}^{conv} = \text{Conv1D}(\boldsymbol{x}), \boldsymbol{X}^{conv} = \text{Conv1D}(\boldsymbol{X}), \boldsymbol{x}, \boldsymbol{X} \in \mathbb{R}^{B\times 2\times L}. \tag{31}$$

Subsequently, referencing the fusion approach of IQFormer, we use another Conv1D to fuse the features of the two modalities, forming the initial encoded representation.

$$z = \text{Conv1D}(\text{BN}(\text{Conv1D}(\text{Concat}(\boldsymbol{x}^{conv}, \boldsymbol{X}^{conv}))), \tag{32}$$

where $\text{BN}(\cdot)$ is the batch normalization, and the $\text{Concat}(\cdot,\cdot)$ denotes the concatenation operation.

Next is the denoising network. Considering that the signal exhibits global dependencies across different feature dimensions after being encoded into the feature space, a transformer-based model is suitable for feature extraction. Currently, there are many signal representation networks based on transformers. We directly adopt the TS-Net temporal encoder used in the TS-DDAE model, which is compatible with the diffusion model and can reasonably encode the noisy steps t into the network.

Finally, regarding the design of the decoder D_ψ, similar to the encoder, we do not need to adopt a strict model architecture for generative tasks. Therefore, we adopt a simple feature decoding, i.e., a linear output after normalization, formalized as $x = \text{LN}(\text{SiLU}(\text{Linear}(z)))$, where $\text{LN}(\cdot)$ is the LayerNorm module, $\text{SiLU}(\cdot)$ is the activation function, and $\text{Linear}(\cdot)$ is the linear layer.

C DETAILS OF DATASETS AND TASKS

C.1 DATASETS

In this section, we present details of datasets we use in this paper. The total statistics are listed in Table 4, where the SNR denotes the signal-to-noise ratio for RML2016.10A, RML2022, RadChar. The RML2016.10A and the RML2022 are the most typical benchmark for AMC task. The RadChar is a synthetic radar signal dataset that contains "coherent_pulse_train", "barker_code", "polyphase_barker_code", "frank_code", and "linear_frequency_modulated" five types of wireless technology. We use the "RadChar-Baseline" with 1,000,000 samples in our experiments. The RML2016.10A, RML2022, and RadChar take 80% of the data for training and 20% for testing, respectively, where data from different classes are sampled uniformly.

Table 4: The statistics of the datasets used in this work.

Dataset	RML2016.10A	RML2022	RadChar
number of samples	220,000	462,000	1,000,000
length of each sample	128	128	512
number of classes	11	11	5
min SNR (SIR)	-20	-20	-20
max SNR (SIR)	18	20	20
SNR (SIR) interval	2	2	1

C.2 TASKS

We evaluate our model with two tasks including the AMC, and WTC. Here we give descriptions of each task.

Automatic Modulation Classification (AMC). In wireless communication, modulation is the process of mapping digital information (bit stream) onto an analog carrier signal in order to transmit it efficiently in the channel. The AMC task aims to identify the modulation type of the received signals.
Wireless Technology Classification (WTC). Wireless technology type is a type of wireless communication technology standard or system that has specific physical layer waveform structure, modulation method, frame format, multiple access mechanism, bandwidth characteristics, and protocol specifications. The WTC task aims to recognize the wireless technology type given the received signals.

D Implementation Details

In this section, we supplement the details regarding the implementation of our model.The model architectures used are described in Appendix B. So, we describe the classifier architecture used during fine-tuning. The output representation, which has the shape $\mathbb{R}^{B\times d\times L}$ with d denoting the hidden dimension, is first aggregated across the temporal dimension via a pooling layer, and then fed into an Multilayer Perceptron (MLP) to produce the final classification logits.

Then, we give details about the training configuration. The pre-training initialization method is the default PyTorch initialization method, i.e., a uniform distribution bounded by $1/\sqrt{(\text{in_features})}$. During pre-training, the optimizer is "AdamW", with parameters: weight_decay=0.01. The lr for RadChar is 2e-5, and 2e-4 for other datasets. Our pre-training learning rate strategy is "ReduceLROnPlateau" according to the pre-training loss, with parameters: mode="min", factor=0.5, patience=200, min_lr=1e-12. Moreover, we take an early stop mechanism with patience=500 according to the loss. During the fine-tuning, we modify some settings. The optimizer is the same. The learning rate strategy is "ReduceLROnPlateau" according to the valid set loss, with parameters: mode="min", factor=0.5, patience=3, min_lr=1e-8. The patience of the early stop mechanism is 15 according to the valid set loss. The final model chosen for testing is the one with the best valid accuracy. The whole network is pre-trained and fine-tuned using Tesla A100.

Table 5: The hyperparameters used in our experiments.

Parameter name	Description	Value
num_layers	The number of model layers	4
max_step	The maximal diffusion steps	1000
timestep	The step used in fine-tuning	84
ratio	The ratio between the temporal noise and the spectral noise	0.414
min_noise	The minimal noise added to signals	5.4e-7
max_noise	The maximal noise added to signals	6.4e-5
lambda_rec	The ratio used in Equation 7	0.57
lambda_reg	The ratio used in Equation 10	1.98
m	The interval in Equation 8	0.19
s	The scaling factor in Equation 8	27.31

E Usage of Hyper-Parameters in Experiments

The hyper-parameters we use in our experiments are listed in Table 5, where we list the parameter name, the description of each parameter, and the value. We use the same hyper-parameters for all of the experiments related to DualDiff.

F Additional Experimental Results

F.1 Efficiency Comparison

In this section, we present an efficiency analysis of DualDiff. To ensure a fair comparison, we only compare DualDiff with SpectrumFM and TS-DDAE, all of which are pre-training models for WSR. The evaluation dataset is the RML2016.10A with input length 128, and the example setting is the

Table 6: The efficiency comparison among SpectrumFM, TS-DDAE and DualDiff on RML2016.10A.

Model	Parameters	Pre-training time	Inference time	Memory use
SpectrumFM	7.98M	1,319s	8.16s	10,162MB
TS-DDAE	10.3M	75.46s	7.21s	8,338MB
DualDiff	6.65M	50.52s	4.75s	5,078MB

setting 0. The results are shown in Table 6, where the Pre-training time and the Inference time are the time the model takes to complete one epoch training or inference. From the results, we can find that, with a similar number of parameters, DualDiff's pre-training speed is significantly faster than SpectrumFM, the inference overhead is also much lower, and the memory consumption is low. Compared to TS-DDAE, since the original implementation of TS-DDAE has a larger parameters, we consider TS-DDAE's running efficiency to be similar to DualDiff's. In practice, we also use the *torch.compile* to further optimize our model, and the running can be faster.

Table 7: The repeated OSCR results (percentage) of DualDiff.

	RML2016.10A						RML2022				RadChar	
	setting-0		setting-1		setting-2		setting-0		setting-1		setting-0	
	Overall	Best	Overall	Best	Overall	Best	Overall	Best	Overall	Best	Overall	Best
Result 0	46.90	70.25	37.15	47.11	51.61	71.83	57.06	69.34	49.98	56.27	81.17	98.58
Result 1	47.18	69.87	37.41	46.63	49.42	66.62	57.90	71.41	52.97	63.30	79.21	98.75
Result 2	47.06	69.87	39.82	51.74	51.57	71.55	56.32	68.47	51.32	60.22	84.37	99.18
Result 3	46.54	65.87	36.61	45.83	50.27	69.97	57.96	72.23	50.07	57.87	81.17	98.08
Result 4	44.99	63.66	36.04	45.31	51.63	71.83	56.61	68.47	52.28	61.57	82.71	96.09
Mean	46.53	67.90	37.41	47.32	50.90	70.36	57.17	69.98	51.32	59.85	81.87	98.14
Std	0.801	2.660	1.296	2.294	0.903	1.994	0.664	1.554	1.183	2.519	1.704	1.082

F.2 The Results of Multiple Experiments

In this section, we repeat the DualDiff experiment with five different seeds, and the results are shown in Table 7. By calculating the mean and variance, we can see that the average result of DualDiff is still better than the best benchmark, demonstrating its stability.

Table 8: The estimation of V_E and V_C (percentage).

Model	AMC_Net	GRU2	ResNet	VGG	IQFormer	TS-DDAE	DualDiff
$1-\mu_f$	36.40	37.50	36.78	37.17	37.16	37.87	45.22
V_E	1.85	2.34	6.76	7.24	1.61	11.49	2.31
V_C	45.83	55.06	34.93	32.26	46.51	36.97	36.36
$V_E + V_C$	47.68	57.40	41.69	39.50	48.12	48.46	**38.67**

F.3 Estimation of V_E and V_C

To validate the dual-variance control mechanism, we estimate the sample-level variance V_E and class subset variance V_C on RML2016.10A. Following the PRRM objective, variance reduction must be evaluated in conjunction with expected risk μ_f to avoid trivial solutions where poor performance yields low variance. We select representative baselines including AMC_Net, GRU2, ResNet, VGG, IQFormer, TS-DDAE, and report their μ_f to ensure a fair comparison. Furthermore, we restrict the analysis to high-SNR samples ($\geq$ 0dB). Under this condition, channel-induced perturbations are minimized, and the estimated V_E primarily reflects the model's sensitivity to intrinsic waveform variability across different samples within the same class. The results are presented in Table 8. DualDiff achieves the lowest total variance ($V_E + V_C$), significantly outperforming all baselines. While some models exhibit slightly lower individual terms, they suffer from higher variance in the

other term, leading to larger total fluctuation. DualDiff effectively balances both variance sources while maintaining superior mean performance, confirming that its stability stems from genuine variance suppression rather than performance trade-offs.

Table 9: The additional ablation OSCR results (percentage) of DualDiff.

Model	RML2016.10A			RML2022		RadChar
	setting-0	setting-1	setting-2	setting-0	setting-1	setting-0
DualDiff	**46.90**	**37.15**	**51.61**	**57.06**	**49.98**	**81.17**
std DDPM	44.14	34.88	50.03	52.19	49.62	61.71
w/o arcface	39.70	28.12	47.59	48.78	41.29	75.25
IQFormer (original)	36.24	29.30	45.93	48.57	40.92	57.50
IQFormer (w/ arcface)	42.05	33.30	46.34	56.23	48.88	78.20
TS-DDAE (original)	35.58	31.84	46.20	42.73	39.41	80.47
TS-DDAE (w/ arcface)	37.46	36.90	48.83	52.38	49.30	80.32

F.4 Additional Ablation Study

In this section, we conduct additional ablation studies on various model variants to further validate the rationality and effectiveness of our design. Specifically, we introduce two variants for our proposed method: (1) pre-training with a standard DDPM (*std DDPM*), and (2) substituting the ArcFace loss with the standard Cross-Entropy (CE) loss (*w/o arcface*). Furthermore, to evaluate the generalizability of the ArcFace loss, we apply it to strong baselines, creating *IQFormer (w/ arcface)* and *TSDDAE (w/ arcface)* by integrating the ArcFace loss into their original architectures. The corresponding results are summarized in Table 9.

The comparison with the *std DDPM* demonstrates substantial performance gains for our full model across all datasets, highlighting the necessity of employing the Latent Diffusion Model (LDM) to effectively align the dual-sampling processes. Similarly, the comparison with the *w/o arcface* variant reveals noticeable performance degradation, underscoring the critical role of the ArcFace loss in enhancing feature discriminability. Moreover, comparing the original baselines with their *w/ arcface* counterparts reveals significant improvements. For instance, integrating the ArcFace loss into IQFormer yields remarkable accuracy boosts (e.g., increasing from 36.24% to 42.05% on RML2016.10A setting-0, and from 57.50% to 78.20% on RadChar). This indicates that the designed ArcFace loss is not only crucial for our DualDiff framework but also highly compatible with and beneficial to other advanced architectures, effectively facilitating better class separation and feature alignment. By integrating the two modules, DualDiff still achieves the best performance, further demonstrating its effectiveness in open-set WSR.

Table 10: The estimation of V_E and V_C of ablation results.

	V_E	V_C	$V_E + V_C$
DualDiff	**2.31**	**36.36**	**38.67**
DualDiff (w/o arcface loss)	2.40	64.00	66.40
DualDiff (w/o pretrain)	2.63	45.43	48.06

Furthermore, we present the estimation result of V_E and V_C following Table 8. The results are presented in Table 10. The results demonstrate that removing the diffusion pre-training component led to an increase in the sample-level variance V_E, while removing the ArcFace loss component resulted in a significant increase in the V_C of the model, further validating our theory.

F.5 Additional Main Results

To rigorously evaluate partition robustness beyond curated settings, we conduct additional experiments on RML2016.10A using extended randomly sampled partition configurations. We randomly generate 96 different partitions settings and provide the OSCR results compared with all baselines. As stated

Table 11: The additional main results of various partitions on RML2016.10A.

	Avg_Mean	Avg_Std	Rank_Mean	Rank_Std
AMC_Net	0.4486	0.0647	4.09	2.03
CGDNN	0.2466	0.1214	9.91	2.16
CNN2	0.1937	0.0945	11.26	1.55
GRU2	0.1124	0.0915	12.27	1.42
MCNet	0.2794	0.1097	9.91	1.76
MSNet	0.4186	0.0699	5.86	2.05
ResNet	0.4007	0.0673	6.60	1.70
Transformer	0.4447	0.0745	4.06	2.31
VGG	0.4005	0.0690	6.62	1.80
FSOS_AMC	0.2618	0.1169	9.81	2.65
IQFormer	0.4338	0.0722	4.87	2.25
TSDDAE	0.4644	0.0653	3.35	2.50
DualDiff	0.4851	0.0696	2.38	1.92

in Appendix F.4, the Arcface loss may benefit the OSCR results of baselines, so we replace the CE loss in baselines with the Arcface loss for fair evaluation. The results are listed in Table 11. Among numerous random partitions, our DualDiff maintains a relatively stable ranking and excellent performance, thus validating the partition robustness of our model.

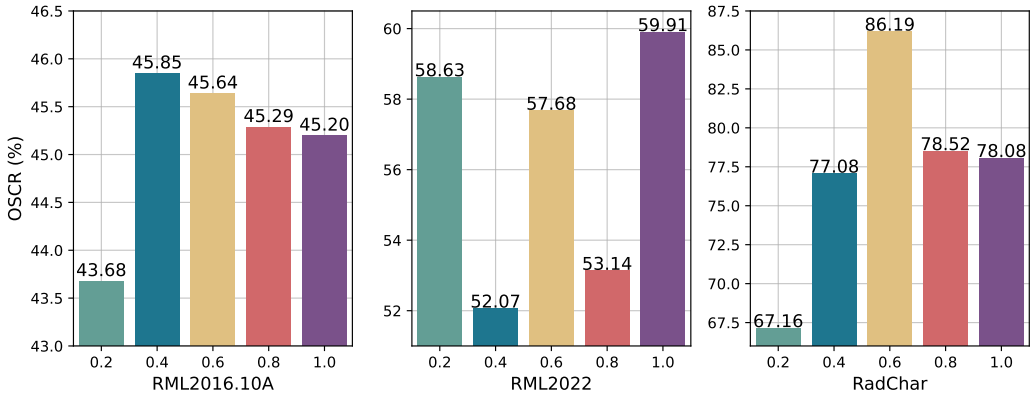


Figure 3: The OSCR results of different λ_{rec} value ranging from 0.2 to 1.0 (left: RML2016.10A, center: RML2022, right: RadChar).

F.6 Hyper-Parameter Analysis

Then, we analyze the impact of hyperparameter λ_{rec} and λ_{reg} in Section 3, which is shown in Figure 3 and Figure 4. The optimal hyper-parameters vary across datasets, reflecting the need to balance variance suppression and expected risk minimization. As derived in Appendix A.2, increasing λ_{rec} linearly increases the upper bound of sample-level variance V_E, as it encourages the encoder to retain deviation details for reconstruction. However, decreasing λ_{rec} too much may cause the encoder to discard essential semantic information, leading to higher expected risk μ_f. Similarly, while larger λ_{reg} suppresses class subset variance V_C by anchoring weights to the ETF structure, excessive regularization may constrain the model's flexibility, harming discriminative performance. Therefore, the performance curves in Figure 3 and Figure 4 exhibit clear peaks, demonstrating the optimal trade-off points where DualDiff achieves minimal PRRM objective by balancing dual variance control with risk minimization. This empirical observation strongly corroborates our theoretical analysis.

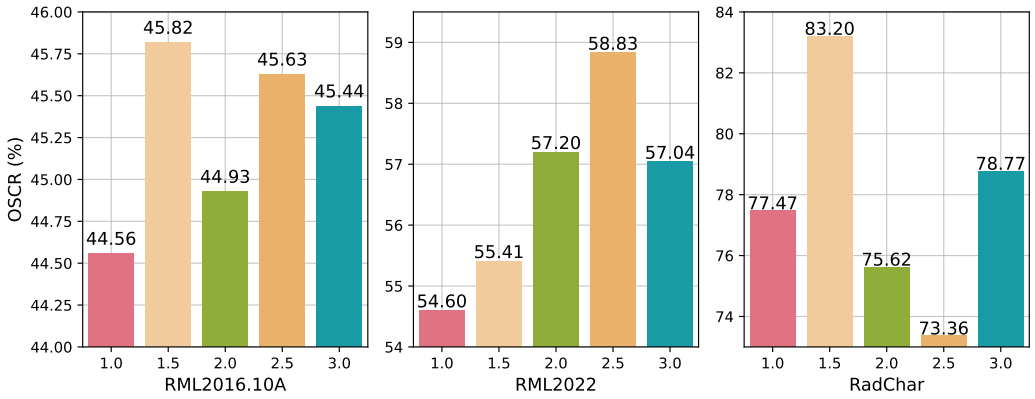


Figure 4: The OSCR results of different λ_{reg} value ranging from 1.0 to 3.0 (left: RML2016.10A, center: RML2022, right: RadChar).

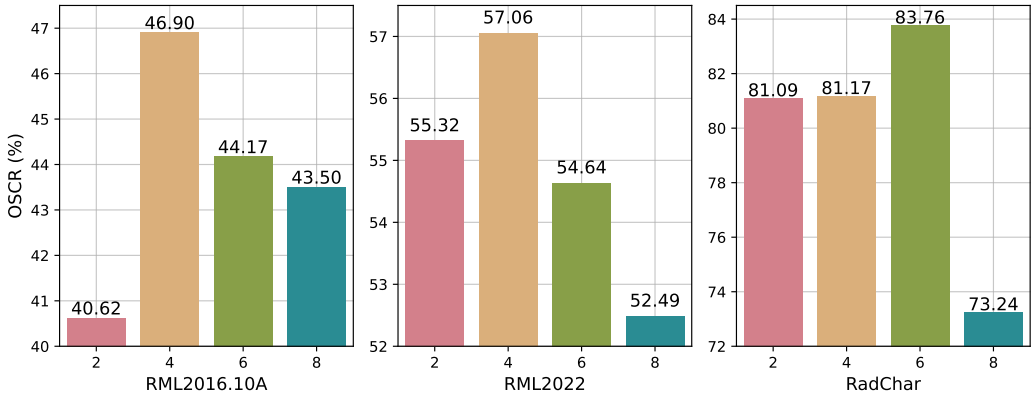


Figure 5: The OSCR results of different number of layers ranging from 2 to 8 (The dataset from left to right is RML2016.10A, RML2022, and RadChar).

F.7 The Effect of Model Size

We analyze the impact of encoder depth on performance to determine the optimal configuration. As shown in Figure 5, OSCR improves as depth increases and peaks at 4 layers. Beyond this point, performance saturates and slightly declines, indicating that 4 layers provide sufficient capacity to capture signal semantics while maintaining generalization on the given data scale. This result validates our choice of 4 layers as the default configuration, achieving the best trade-off between representational power and computational efficiency. The pre-training strategy ensures robust performance at this depth without requiring excessive model complexity.