

QAMP 2025

Enhancing Quantum Diffusion Models for Complex Image Generation

**Jeongbin Jo¹, Santanam Wishal², Shah Md Khalil Ullah³, and
Shan Kowalski⁴**

¹Department of Physics , Yonsei University , jeongbin033@yonsei.ac.kr

²Asia Cyber University , santawishal17@gmail.com

³Khulna University of Engineering and Technology , hamimkhandakar222@gmail.com

⁴Eleven Dimension LLC , shanjdk2012@gmail.com

January 23, 2026

Abstract

In this study, we propose a Hybrid Quantum-Classical U-Net architecture designed for complex image generation. By integrating a classical encoder-decoder structure with a Quantum Bottleneck, the model leverages the expressivity of quantum circuits while maintaining structural efficiency. The architecture employs Skip Connections to preserve semantic information, which is critical for maintaining image sharpness.

Contents

1	Intrduction	2
1.1	Quantum Convolutional Neural Network	3
2	Diffusion Model	3
2.1	Classical Diffusion Model	3
2.1.1	Forward Process: Diffusion Process	3
2.1.2	Reverse Process: Generative Process	4
2.1.3	Loss Function	4
2.1.4	Continuous-Time Formulation: SDE	4
2.2	Quantum Diffusion Model	5
2.2.1	Forward Process: Depolarizing Channel	5
2.2.2	Reverse Process: Quantum Denoising	5
2.2.3	Loss Function: Fidelity or Trace Distance	5

3	Quantum Data Encoding	6
3.1	Basis Encoding	6
3.2	Amplitude Encoding	6
3.3	Angle Encoding	6
3.4	Phase Encoding	6
3.5	Dense Angle Encoding	7
4	Quantum Circuit	7
4.1	General Variational Ansatzes	7
4.2	Ansatzes for Quantum Machine Learning and Vision	7
4.3	Optimal General Two-Qubit Ansatz (Vatan-Williams Decomposition)	8
4.4	Global Correlation and Phase Mixing	8
5	Adaptive Non-Local Measurement	9
5.1	Trainable Hermitian Construction	10
5.2	High-Dimensional Feature Extraction	10
6	Circuit Benchmarking and Evaluation	10
6.1	Expressibility via KL Divergence	11
6.2	Entangling Capability via Meyer-Wallach Measure	11
6.3	Visualizing the State Space	12
6.4	Architecture	12
7	Experimental Results	13
8	Conclusion	15

1 Introduction

Quantum Machine Learning (QML) is an emerging interdisciplinary research field that integrates principles of quantum computing with classical machine learning techniques to enhance data processing, pattern recognition, and optimization tasks. By exploiting quantum mechanical phenomena such as superposition, entanglement, and quantum interference, QML aims to provide computational advantages over classical learning algorithms, particularly for high-dimensional and complex datasets.

In classical machine learning, the performance of algorithms is often constrained by computational complexity and memory requirements as data size and model depth increase. Quantum computing, on the other hand, offers a fundamentally different computation paradigm, where information is represented using quantum bits (qubits) that can

exist in multiple states simultaneously. This property allows quantum algorithms to explore large solution spaces more efficiently than their classical counterparts in certain scenarios.

One of the key motivations behind QML is the potential for speedup in tasks such as linear algebra operations, optimization, and sampling, which form the backbone of many machine learning models. Quantum-enhanced algorithms, including quantum support vector machines, variational quantum circuits, and quantum neural networks, have been proposed to accelerate learning processes and improve model expressiveness.

Despite its promising advantages, QML also faces several challenges. Current quantum hardware, commonly referred to as Noisy Intermediate-Scale Quantum (NISQ) devices, suffers from limited qubit counts, short coherence times, and gate errors. These hardware constraints restrict the depth and scalability of quantum models, often requiring hybrid quantum-classical approaches for practical implementations. Additionally, the theoretical quantum advantage of many QML algorithms remains an open research question, as classical algorithms continue to improve rapidly.

In summary, Quantum Machine Learning represents a promising yet evolving research direction. While it offers the potential for enhanced computational power and novel learning capabilities, significant challenges related to hardware limitations, noise, and algorithmic scalability must be addressed before widespread real-world adoption becomes feasible.

1.1 Quantum Convolutional Neural Network

2 Diffusion Model

2.1 Classical Diffusion Model

Diffusion models are a class of generative models in machine learning inspired by non-equilibrium thermodynamics. The goal is to learn a diffusion process that destroys structure in data, and then learn a reverse process that restores structure to generate new data samples from noise.[\[3\]](#)

2.1.1 Forward Process: Diffusion Process

The forward process is defined by a Markov chain that gradually adds Gaussian noise to the data according to a variance schedule β_1, \dots, β_T .

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t\mathbf{I}\right) \quad (1)$$

Using the notation $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$, we can sample x_t at any arbitrary time step t directly from x_0 in closed form:

$$q(x_t|x_0) = \mathcal{N}\left(x_t; \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}\right) \quad (2)$$

This property allows for efficient training by eliminating the need to iterate through all previous timesteps.

2.1.2 Reverse Process: Generative Process

The generative process[5] is defined as the reverse Markov chain. Since the exact reverse posterior $q(x_{t-1}|x_t)$ is intractable, we approximate it using a neural network with parameters θ :

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \quad (3)$$

Here, $\mu_\theta(x_t, t)$ is the predicted mean, and $\Sigma_\theta(x_t, t)$ is the covariance (often fixed to $\beta_t \mathbf{I}$ or learned).

2.1.3 Loss Function

Training is performed by optimizing the variational lower bound (VLB) on the negative log-likelihood.

$$\mathcal{L} = \mathbb{E}[-\log p_\theta(\mathbf{x}_0)] \leq \mathbb{E}_q \left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right] \quad (4)$$

This can be rewritten as a sum of KL divergence[8] terms:

$$\mathcal{L} = \mathbb{E}_q \left[\underbrace{D_{KL}(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T))}_{L_T} + \sum_{t>1} \underbrace{D_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))}_{L_{t-1}} \underbrace{- \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0} \right] \quad (5)$$

In practice, Ho et al. found that a simplified objective function yields better sample quality. The objective is to predict the noise ϵ added to x_0 :

$$\mathcal{L}_{\text{simple}}(\theta) = \mathbb{E}_{t, x_0, \epsilon} [\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t)\|^2] \quad (6)$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ and ϵ_θ is a function approximator (e.g., U-Net).

2.1.4 Continuous-Time Formulation: SDE

The discrete diffusion process can be generalized to continuous time using Stochastic Differential Equations (SDEs). The forward process is described by the following Itô SDE[7]:

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w} \quad (7)$$

where $\mathbf{f}(\cdot, t)$ is the drift coefficient, $g(t)$ is the diffusion coefficient, and \mathbf{w} is a standard Wiener process.

Remarkably, the reverse process—generating data from noise—is also a diffusion process governed by the *reverse-time SDE*[1]:

$$d\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - g(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x})] dt + g(t)d\bar{\mathbf{w}} \quad (8)$$

Here, dt represents a negative time step, and $\bar{\mathbf{w}}$ is the Brownian motion in reverse time. The term $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$, known as the *score function*, points towards high-density regions of the data.

Therefore, the core task of the diffusion model is to learn a score-based model $s_\theta(\mathbf{x}, t) \approx \nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ using a neural network (or PQC in our case), allowing us to numerically solve the reverse SDE to generate samples from noise.

2.2 Quantum Diffusion Model

While classical diffusion models operate on probability distributions over classical data, Quantum Diffusion Models (QDMs) extend this concept to the Hilbert space of quantum states. The goal is to generate quantum states (density matrices) from a maximally mixed state.

2.2.1 Forward Process: Depolarizing Channel

Instead of adding Gaussian noise, the forward process in QDM is typically modeled as a standard depolarizing channel acting on a density matrix ρ . For a system of n qubits with dimension $d = 2^n$, the state at step t is given by:

$$\rho_t = \mathcal{E}_t(\rho_{t-1}) = (1 - p_t)\rho_{t-1} + p_t \frac{I}{d} \quad (9)$$

where $p_t \in [0, 1]$ is the depolarization probability (noise schedule). Similar to the classical case, we can express ρ_t directly from the initial state ρ_0 . Let $\alpha_t = \prod_{s=1}^t (1 - p_s)$, then:

$$\rho_t = \alpha_t \rho_0 + (1 - \alpha_t) \frac{I}{d} \quad (10)$$

As $t \rightarrow T$, $\alpha_T \rightarrow 0$, and the state converges to the maximally mixed state $\rho_T \approx \frac{I}{d}$, which contains no information about ρ_0 .

2.2.2 Reverse Process: Quantum Denoising

The reverse process aims to restore the quantum state from the noise. This is modeled by a parameterized quantum circuit (PQC), denoted as a unitary operator $U(\theta)$. The discrete reverse step can be approximated as:

$$\rho_{t-1} \approx \mathcal{D}_\theta(\rho_t, t) = U(\theta_t) \rho_t U^\dagger(\theta_t) \quad (11)$$

For more complex generative tasks, the reverse process may involve ancillary qubits and measurements to simulate non-unitary maps.

2.2.3 Loss Function: Fidelity or Trace Distance

Unlike the KL divergence used in classical models, quantum models often use Fidelity or Hilbert-Schmidt distance to measure the closeness between the generated state and the target state. A common objective is to maximize the overlap with the target pure state $|\psi\rangle$:

$$\mathcal{L}(\theta) = 1 - \mathbb{E} [\langle \psi | \rho_{\text{gen}}(\theta) | \psi \rangle] \quad (12)$$

Alternatively, for density matrices, we minimize the trace distance or maximize the quantum fidelity $F(\rho, \sigma) = (\text{tr} \sqrt{\sqrt{\rho} \sigma \sqrt{\rho}})^2$.

3 Quantum Data Encoding

We referred to IBM Quantum Platform, especially Quantum Machine Learning.[\[6\]](#)

3.1 Basis Encoding

Basis encoding maps a classical P -bit string directly to a computational basis state of a P -qubit system. For a single feature represented as binary bits (b_1, b_2, \dots, b_P) , the quantum state is:

$$|x\rangle = |b_1, b_2, \dots, b_P\rangle, \quad b_i \in \{0, 1\} \quad (13)$$

3.2 Amplitude Encoding

Amplitude encoding maps a normalized classical N -dimensional data vector \vec{x} to the amplitudes of an n -qubit quantum state, where $n = \lceil \log_2 N \rceil$.

$$|\psi_x\rangle = \frac{1}{\alpha} \sum_{i=1}^N x_i |i\rangle \quad (14)$$

Here, $|i\rangle$ is the computational basis state and α is a normalization constant ensuring $\langle\psi_x|\psi_x\rangle = 1$. And α is a normalization constant to be determined from the data being encoded. We use amplitude encoding in the quantum diffusion model.

$$\alpha = \sqrt{\sum_{i=1}^N |x_i|^2} \quad (15)$$

3.3 Angle Encoding

Angle encoding maps each feature x_k to the rotation angle of a qubit using R_Y gates. For a data vector \vec{x} , the state is a product state:

$$|\vec{x}\rangle = \bigotimes_{k=1}^N R_Y(x_k) |0\rangle = \bigotimes_{k=1}^N \left(\cos\left(\frac{x_k}{2}\right) |0\rangle + \sin\left(\frac{x_k}{2}\right) |1\rangle \right) \quad (16)$$

3.4 Phase Encoding

Phase encoding maps data features to the phase of qubits using Phase gates $P(\phi)$, typically applied after Hadamard gates.

$$|\vec{x}\rangle = \bigotimes_{k=1}^N P(x_k) |+\rangle = \frac{1}{\sqrt{2^N}} \bigotimes_{k=1}^N (|0\rangle + e^{ix_k} |1\rangle) \quad (17)$$

3.5 Dense Angle Encoding

Dense angle encoding encodes two features, x_k and x_l , into a single qubit using both Y -axis and Z -axis rotations.

$$|x_k, x_l\rangle = R_Z(x_l)R_Y(x_k)|0\rangle = \cos\left(\frac{x_k}{2}\right)|0\rangle + e^{ix_l}\sin\left(\frac{x_k}{2}\right)|1\rangle \quad (18)$$

Extending this to more features, the data vector $\vec{x} = (x_1, \dots, x_N)$ can be encoded as:

$$|\vec{x}\rangle = \bigotimes_{k=1}^{N/2} (\cos x_{2k-1}|0\rangle + e^{ix_{2k}}\sin x_{2k-1}|1\rangle) \quad (19)$$

4 Quantum Circuit

The choice of quantum circuit ansatz is critical in designing Quantum Machine Learning (QML) models, as it determines the expressivity and trainability of the network within the constraints of Noisy Intermediate-Scale Quantum (NISQ) devices. Based on recent surveys and specific applications in diffusion models, we categorize relevant ansatzes into general-purpose architectures and task-specific designs for image processing.

4.1 General Variational Ansatzes

- **Hardware-Efficient Ansatz (HEA):** Designed to minimize circuit depth and gate count on NISQ devices, HEA utilizes a layered structure of parameterized single-qubit rotations followed by entangling gates (e.g., CNOTs) tailored to the specific connectivity of the quantum hardware [4]. While it offers high implementability, it is known to suffer from barren plateaus if not carefully initialized.
- **Hamiltonian Variational Ansatz (HVA):** Inspired by the Quantum Approximate Optimization Algorithm (QAOA) and adiabatic quantum computation, HVA constructs the ansatz based on the problem Hamiltonian. It is particularly effective for physics-inspired problems but introduces a hierarchical structure to manage complexity [4].

4.2 Ansatzes for Quantum Machine Learning and Vision

For high-dimensional data such as images, specialized ansatzes inspired by tensor networks and convolutional neural networks are preferred.

- **Quantum Convolutional Neural Network (QCNN):** QCNN adapts the structure of classical CNNs to the quantum domain. It consists of alternating convolutional layers (quasilocal unitary evolutions) and pooling layers (measurements or partial traces) to hierarchically reduce dimensionality while extracting features [4]. This structure effectively captures spatial correlations, making it suitable for image generation tasks.

- **Multiscale Entanglement Renormalization Ansatz (MERA):** Originally designed for simulating quantum many-body systems, MERA represents quantum states using a hierarchical tensor network. It efficiently captures correlations at different length scales, which is analogous to the feature extraction process in deep learning [4].

4.3 Optimal General Two-Qubit Ansatz (Vatan-Williams Decomposition)

We adopt the optimal quantum circuit decomposition for general two-qubit interactions as proposed by Vatan and Williams [12]. Specifically, we utilize the explicit circuit construction for the non-local interaction block $N(\alpha, \beta, \gamma)$, which is the core component of the ansatz.

The circuit structure, depicted in Fig. 1, consists of three CNOT gates interleaved with parameterized single-qubit rotations. This configuration is capable of simulating the unitary operator $U = \exp(-i(\alpha\sigma_x \otimes \sigma_x + \beta\sigma_y \otimes \sigma_y + \gamma\sigma_z \otimes \sigma_z))$ up to local basis transformations.

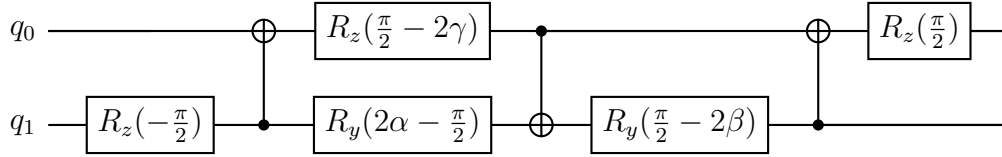


Figure 1: The parameterized quantum circuit used in our Quantum Bottleneck. This structure implements the optimal decomposition for arbitrary two-qubit entangling gates, characterized by the parameters α, β, γ . Note the alternating CNOT direction and the specific rotation angles derived from the canonical decomposition.

The parameters α, β, γ serve as the learnable weights of the quantum diffusion model, allowing the network to capture complex correlations between features represented by qubits q_0 and q_1 .

4.4 Global Correlation and Phase Mixing

While the convolutional layers capture local spatial dependencies, generating high-fidelity images requires the model to understand global structures and high-frequency details. To address this, we introduce a *Phase Mixing* block designed to enhance the expressivity of the ansatz in the frequency domain [schuld2021effect].

The Phase Mixing block consists of two stages:

1. **Star-Topology Entanglement:** We apply a series of Controlled-Rotation Z (CRZ) gates. As shown in the code implementation, the first qubit (q_0) acts as a central node (control), interacting with all other qubits (q_i) via parameterized rotations. This creates a "star" connectivity graph, efficiently distributing information from the central feature to the entire register.

2. **Quantum Fourier Transform (QFT):** Following the entanglement, a QFT operation is applied to the register. This transforms the encoded information from the computational basis to the Fourier basis, effectively mixing the local correlations into global phase information [nielsen2010quantum].

The circuit diagram for this block is illustrated in Fig. 2.

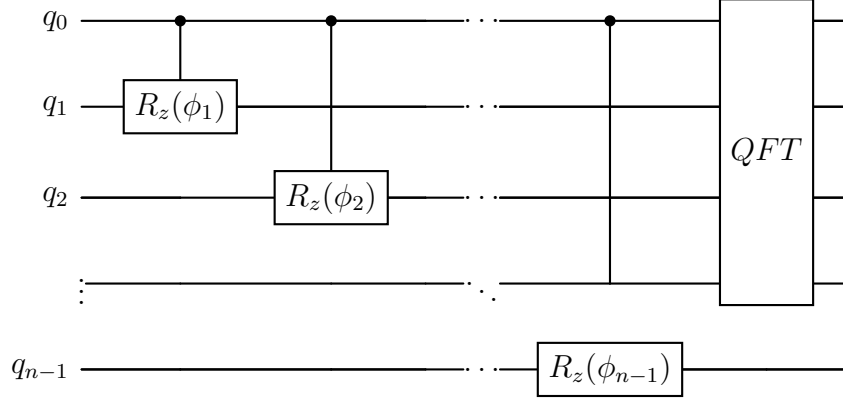


Figure 2: The *Phase Mixing* architecture. It employs a star-topology interaction where the first qubit controls parameterized R_z rotations on all other qubits, followed by a Quantum Fourier Transform (QFT) to enforce global mixing. This structure allows the model to learn complex frequency-domain features essential for image sharpness.

Mathematically, this operation can be expressed as:

$$U_{\text{mix}}(\vec{\phi}) = \text{QFT} \cdot \prod_{i=1}^{n-1} CRZ_{0 \rightarrow i}(\phi_i) \quad (20)$$

where $CRZ_{c \rightarrow t}(\phi) = |0\rangle\langle 0|_c \otimes I_t + |1\rangle\langle 1|_c \otimes R_z(\phi)_t$.

5 Adaptive Non-Local Measurement

In conventional Variational Quantum Circuits (VQCs), information extraction is typically limited to fixed local measurements, such as the expectation values of Pauli-Z operators ($\langle \sigma_z^{(i)} \rangle$). This approach restricts the accessible information to local properties of individual qubits, failing to capture the rich correlations encoded in the entangled quantum state.

To fully leverage the high-dimensional Hilbert space of our quantum bottleneck, we implement the *Adaptive Non-Local Observable (ANO)* framework [9]. Instead of fixed operators, we employ a set of trainable Hermitian observables $\{H_k\}_{k=1}^K$ that act on the entire qubit register.

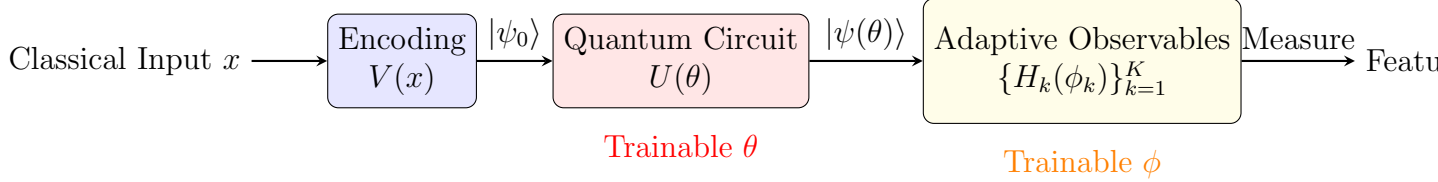


Figure 3: Schematic of the Quantum Bottleneck with Adaptive Non-Local Observables (ANO). Unlike standard VQCs that output N local measurements, our ANO framework employs a set of K trainable Hermitian operators $\{H_k(\phi_k)\}$. This allows the extraction of high-dimensional feature vectors ($K \geq N$) from the entangled quantum state $|\psi(\theta)\rangle$, effectively acting as a learnable quantum lens for super-resolution tasks.

5.1 Trainable Hermitian Construction

For an n -qubit system with dimension $D = 2^n$, a general observable is represented by a $D \times D$ Hermitian matrix. We parameterize the k -th observable $H_k(\phi_k)$ using a complex matrix $M_k(\phi_k) \in \mathbb{C}^{D \times D}$ with learnable weights ϕ_k :

$$H_k(\phi_k) = \frac{1}{2} (M_k(\phi_k) + M_k(\phi_k)^\dagger) \quad (21)$$

This construction guarantees the Hermiticity ($H = H^\dagger$) required for valid quantum measurements while allowing the optimization process to explore the full manifold of observable operators.

5.2 High-Dimensional Feature Extraction

A key advantage of our approach is the ability to decouple the output feature dimension K from the number of physical qubits n . By configuring $K \geq n$, we can extract an over-complete set of features from the latent quantum state $\rho = |\psi(\theta)\rangle\langle\psi(\theta)|$. The output vector $\mathbf{y} \in \mathbb{R}^K$ fed into the classical decoder is given by:

$$y_k = \langle\psi(\theta)|H_k(\phi_k)|\psi(\theta)\rangle = \text{Tr}(\rho H_k(\phi_k)), \quad k = 1, \dots, K \quad (22)$$

By jointly training the circuit parameters θ and the measurement parameters ϕ , the model learns to construct "quantum lenses" that focus on the specific sub-spaces containing the most relevant features for image reconstruction, such as high-frequency edge details.

6 Circuit Benchmarking and Evaluation

Before integrating the Quantum Bottleneck into the full U-Net architecture, we quantitatively evaluated the intrinsic properties of our proposed ansatz. Following the framework established by Sim et al. [11], we utilize two key descriptors: *Expressibility* and *Entangling*

Capability. These metrics ensure that our circuit is sufficiently expressive to model complex latent distributions while effectively capturing non-local correlations required for the diffusion process.

6.1 Expressibility via KL Divergence

Expressibility measures the ability of a parameterized quantum circuit (PQC) to explore the Hilbert space. We quantify this by comparing the distribution of state fidelities generated by our ansatz against the distribution expected from an ensemble of Haar-random states (which represents the maximally expressive uniform distribution).

Let $F = |\langle \psi_\theta | \psi_\phi \rangle|^2$ be the fidelity between two states sampled from the PQC with random parameters θ, ϕ . The probability density function (PDF) of fidelities for Haar-random states in an N -dimensional Hilbert space ($N = 2^n$) is given by $P_{\text{Haar}}(F) = (N-1)(1-F)^{N-2}$.

The expressibility E is defined as the Kullback-Leibler (KL) divergence between the PQC’s fidelity distribution $P_{\text{PQC}}(F)$ and the analytical Haar distribution $P_{\text{Haar}}(F)$:

$$E = D_{KL}(P_{\text{PQC}} || P_{\text{Haar}}) = \int_0^1 P_{\text{PQC}}(F) \ln \left(\frac{P_{\text{PQC}}(F)}{P_{\text{Haar}}(F)} \right) dF \quad (23)$$

A lower value of E indicates that the ansatz can uniformly explore the Hilbert space, approaching the statistical properties of random states. In our experiments, we approximate this integral using a discretized histogram of sampled fidelities.

6.2 Entangling Capability via Meyer-Wallach Measure

To verify the circuit’s ability to generate multi-partite entanglement—a crucial feature for capturing global dependencies in image data—we employ the Meyer-Wallach (MW) measure Q . For a given state $|\psi\rangle$, the MW measure is defined as the average purity of the single-qubit reduced density matrices:

$$Q(|\psi\rangle) = \frac{4}{n} \sum_{k=1}^n D(\iota_k(|\psi\rangle)) = \frac{4}{n} \sum_{k=1}^n \frac{1}{2} (1 - \text{Tr}(\rho_k^2)) \quad (24)$$

where $\rho_k = \text{Tr}_{-k}(|\psi\rangle\langle\psi|)$ is the reduced density matrix of the k -th qubit. The value Q ranges from 0 (product states, unentangled) to 1 (maximally entangled, e.g., GHZ states).

We estimate the *Entangling Capability* of our ansatz by averaging Q over an ensemble of states sampled with uniformly random parameters:

$$\bar{Q} = \frac{1}{S} \sum_{i=1}^S Q(|\psi(\theta_i)\rangle) \quad (25)$$

A high \bar{Q} value confirms that our *ConvUnit* and *PhaseMixing* layers successfully distribute information across the qubit register, facilitating the "global mixing" required for high-resolution image synthesis.

6.3 Visualizing the State Space

In addition to quantitative metrics, we visualize the distribution of single-qubit reduced states on the Bloch sphere. As shown in Fig. 4, the concentration of states within the interior of the Bloch sphere (as opposed to the surface) serves as visual evidence of entanglement, consistent with high Meyer-Wallach values.

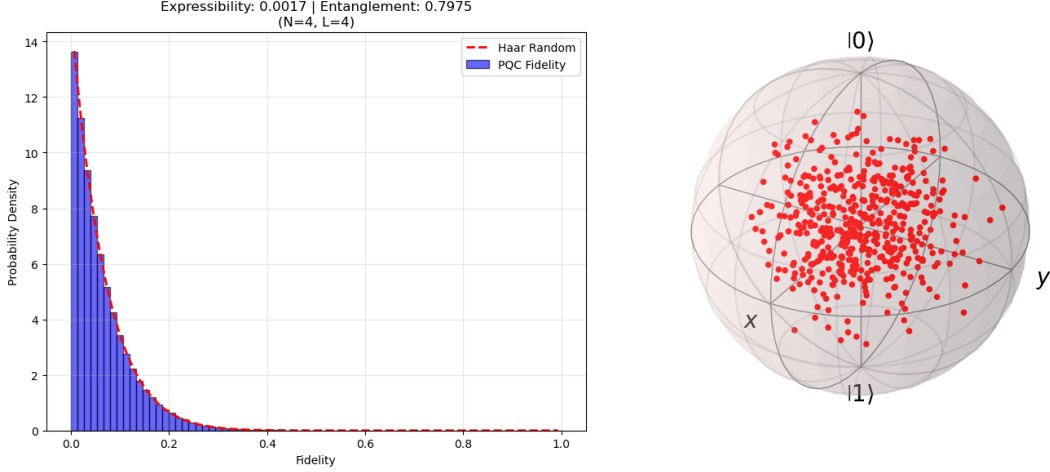


Figure 4: Visualization of the single-qubit reduced state distribution on the Bloch sphere generated by our ANO-based ansatz. Each red point represents the state of the first qubit traced out from the multi-qubit system, sampled over random circuit parameters. (Left) Points distributed strictly on the surface indicate product states with no entanglement. (Right) Points filling the interior volume demonstrate the generation of strong multi-partite entanglement, as the reduced state becomes mixed due to correlations with other qubits. This volumetric coverage visually corroborates the high Meyer-Wallach measure and Expressibility scores.

6.4 Architecture

The proposed model, illustrated in Figure 5, adopts a hybrid quantum-classical U-Net architecture inspired by recent advances in quantum generative diffusion models[2, 10]. The architecture is designed to leverage the expressivity of quantum circuits while maintaining the reconstruction capability of classical networks. It consists of three distinct modules: a classical encoder, a quantum bottleneck (core), and a classical decoder.

First, the **Classical Encoder** transforms the input data (flattened vector) into a lower-dimensional latent representation. To seamlessly interface with the quantum layer, we utilize complex-valued linear layers (**ComplexLinear**) that preserve phase information necessary for quantum state preparation.

Second, the **Quantum Bottleneck** acts as the core generative component. We employ *amplitude encoding* to efficiently map the classical latent vector \mathbf{z} onto an n -qubit quantum state $|\psi\rangle$ by normalizing the complex vector. This state is evolved by a Parameterized

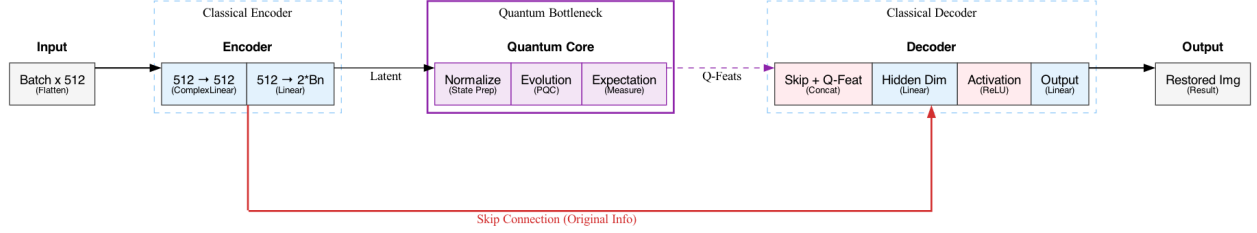


Figure 5: Schematic of the Hybrid Quantum-Classical U-Net Architecture. The model integrates a quantum bottleneck for feature extraction and employs a skip connection to preserve original semantic information during the reverse diffusion process.

Quantum Circuit (PQC), $U(\theta)$, which learns the reverse diffusion dynamics. The resulting quantum state is then measured to extract quantum features (Q-Feats).

Finally, the **Classical Decoder** reconstructs the denoised sample from the quantum features. A crucial design element is the inclusion of a **Skip Connection** (red path in Figure 5). This connection concatenates the original flattened input vector directly with the quantum features in the decoder. This mechanism is essential for mitigating the "bottleneck" problem inherent in low-qubit quantum circuits and preventing the model from converging to trivial identity mappings by preserving the original signal structure.

7 Experimental Results

We evaluated the proposed Hybrid Quantum U-Net using the MNIST dataset. The training was conducted for 100 epochs, and the model demonstrated stable convergence behavior. As shown in Figure 6, the final infidelity loss reached approximately 0.25. Remarkably, the total execution time was only about 3 minutes, highlighting the computational efficiency of our hybrid architecture compared to fully classical counterparts or more complex quantum simulations.

Figure 7 visualizes the forward diffusion process used in our experiment. The input data is gradually corrupted by adding Gaussian noise over discrete time steps t , eventually converging to an isotropic Gaussian distribution.

To analyze the generative capability and scalability of the model, we compared two scenarios: a low-complexity subset (digits 0 and 1) and the full dataset (digits 0 through 9). The generation results are presented in Figure 8.

As observed in Figure 8a, the model successfully generates distinct and sharp images when trained on binary classes. However, as the complexity increases to include all ten digits (Figure 8b), the generated samples exhibit ambiguous shapes and blurring. This suggests that while the current quantum bottleneck is efficient, its limited Hilbert space (number of qubits) constrains the expressibility required to fully capture the complex multimodal distribution of the entire dataset.

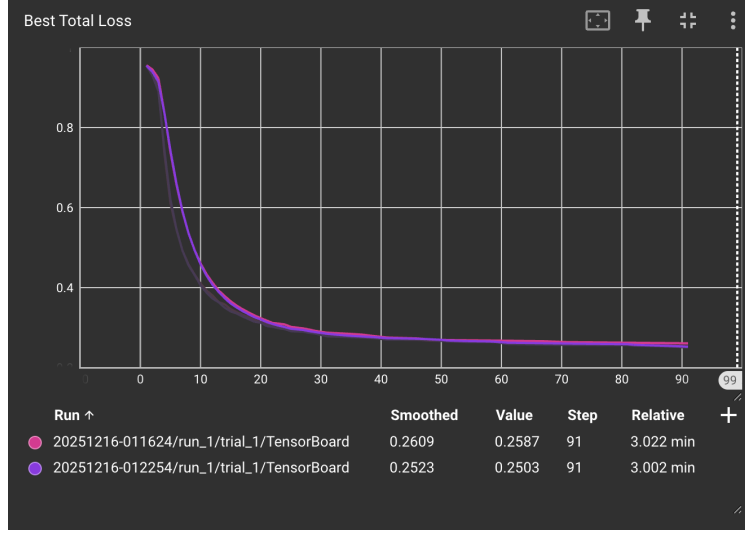


Figure 6: Training Loss Convergence. The model achieves a stable loss of ≈ 0.25 within 100 epochs, demonstrating rapid convergence.

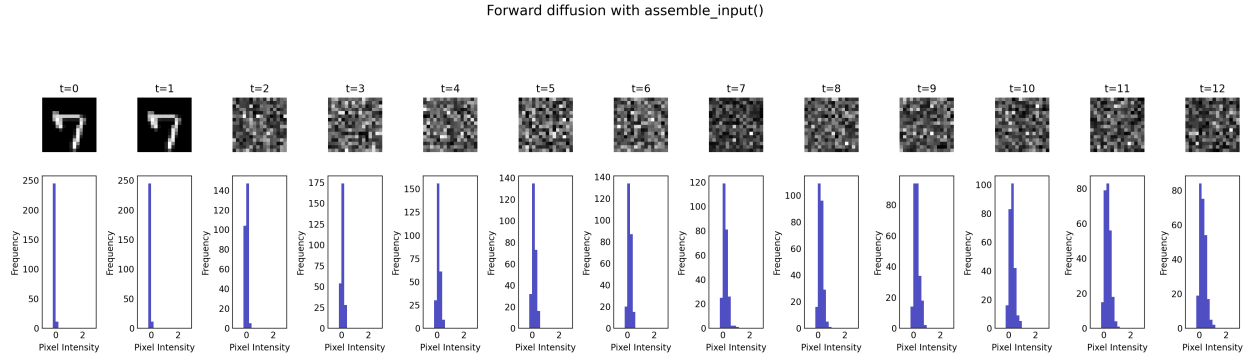
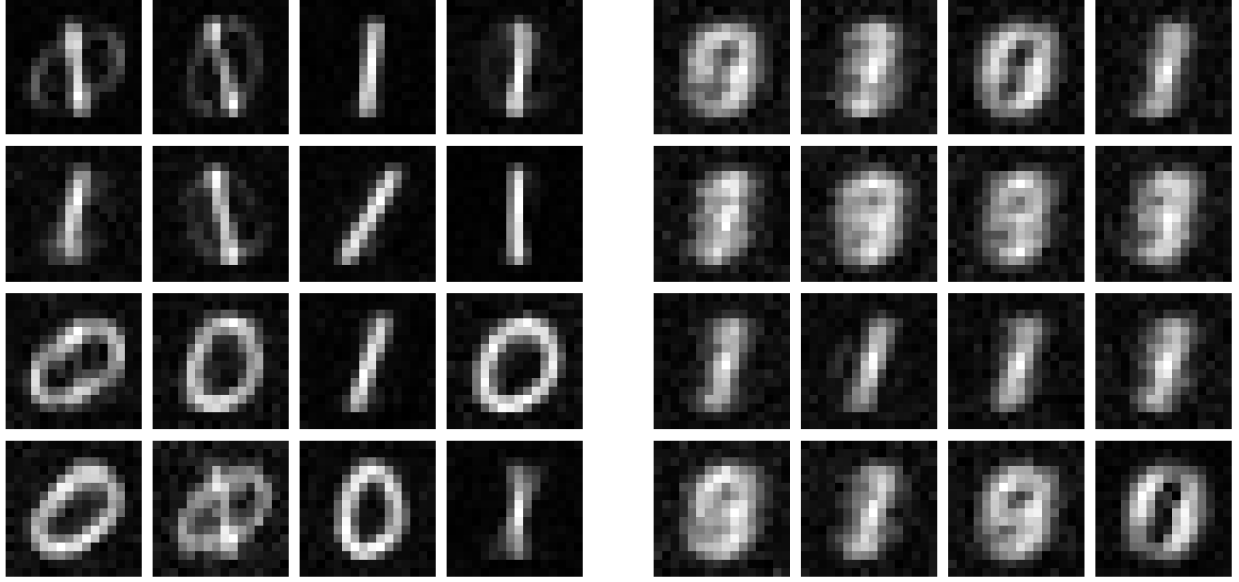


Figure 7: Visualization of the Forward Diffusion Process. The original digits (left) are progressively transformed into random noise (right).



(a) Binary Class (Digits 0, 1)

(b) Multi-Class (Digits 0-9)

Figure 8: Comparison of Generated Samples. (a) shows sharp features for the binary case, while (b) exhibits increased ambiguity and blurring due to higher data complexity.

8 Conclusion

In this study, we proposed a **Hybrid Quantum-Classical U-Net** architecture for complex image generation and validated its effectiveness using the MNIST dataset. The experimental results demonstrated that the proposed model exhibits significant generative capabilities with a reduced number of parameters, confirming the potential of hybrid structures in the NISQ era. The key insights gained, technical challenges faced, and our current research direction are summarized below.

Key Insights and Learnings

- **Structural Efficiency:** Through code implementation (**QuantumUNet**), we verified that the ‘Quantum Bottleneck’ architecture—which compresses features via a classical encoder before mapping them to a quantum state is significantly more efficient than encoding the entire high-dimensional image directly into qubits.
- **Importance of Skip Connections:** We discovered that **Skip Connections**, which combine original classical features with quantum features in the decoder stage, are critical. Without them, the information loss inherent in the quantum circuit significantly degrades image sharpness. This highlights that preserving the **Classical Path** is essential for maintaining reconstruction quality in hybrid models.

Technical Challenges

- **Simulation Overhead (Running Speed):** A major practical barrier was the **ML running speed**. Simulating quantum circuits (especially tensor product operations for entanglement) on classical hardware incurs massive computational costs. This overhead severely constrained our ability to scale up the number of qubits and circuit depth within a reasonable training time.
- **Barren Plateaus & Optimization Landscape:** Beyond computational speed, we faced the fundamental challenge of **Barren Plateaus**. The standard Adam optimizer struggled to navigate the flat loss landscapes typical of parameterized quantum circuits (PQCs). This often led to stagnation or convergence to local minima, particularly when attempting to learn the complex multi-modal distributions of the multi-class MNIST dataset.

Current Status: Enhancing Optimization with QNGD

To address these optimization challenges and ensure convergence stability, we are currently implementing **Quantum Natural Gradient Descent (QNGD)**.

- **Addressing the Optimization Geometry:** Standard optimizers like Adam assume a Euclidean parameter space, which is ill-suited for the quantum state space that follows Riemannian geometry. We are adopting QNGD to adjust the gradient direction by incorporating the **Fubini-Study Metric Tensor** (g_{ij}), which represents the curvature of the quantum information geometry. The update rule is defined as:

$$\theta_{t+1} = \theta_t - \eta g^{-1}(\theta_t) \nabla \mathcal{L}(\theta_t) \quad (26)$$

where η is the learning rate and $g^{-1}(\theta_t)$ corrects the gradient based on the local curvature.

- **Quality over Speed (Trade-off):** We acknowledge that calculating the metric tensor increases the computational cost per step, potentially lengthening the wall-clock training time. However, we accept this trade-off to **mitigate Barren Plateaus** and achieve higher **Fidelity** with fewer training epochs. Our focus has shifted from raw simulation speed to ‘**Optimization Efficiency**’, aiming to overcome the expressivity and convergence limitations of the current model.

References

- [1] Brian D.O. Anderson. “Reverse-time diffusion equation models”. In: *Stochastic Processes and their Applications* 12.3 (1982), pp. 313–326. ISSN: 0304-4149. DOI: [https://doi.org/10.1016/0304-4149\(82\)90051-5](https://doi.org/10.1016/0304-4149(82)90051-5). URL: <https://www.sciencedirect.com/science/article/pii/0304414982900515>.

- [2] Andrea Cacioppo et al. *Quantum Diffusion Models*. 2023. arXiv: [2311.15444](https://arxiv.org/abs/2311.15444) [quant-ph]. URL: <https://arxiv.org/abs/2311.15444>.
- [3] *Diffusion Model*. URL: https://en.wikipedia.org/wiki/Diffusion_model (visited on 12/07/2025).
- [4] Xiaoyu Guo, Takahiro Muta, and Jianjun Zhao. *Quantum Circuit Ansatz: Patterns of Abstraction and Reuse of Quantum Algorithm Design*. 2024. arXiv: [2405.05021](https://arxiv.org/abs/2405.05021) [cs.SE]. URL: <https://arxiv.org/abs/2405.05021>.
- [5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. “Denoising Diffusion Probabilistic Models”. In: *CoRR* abs/2006.11239 (2020). arXiv: [2006.11239](https://arxiv.org/abs/2006.11239). URL: <https://arxiv.org/abs/2006.11239>.
- [6] *IBM Quantum Platform Quantum Machine Learning*. URL: <https://quantum.cloud.ibm.com/learning/en/courses/quantum-machine-learning/data-encoding#methods-of-encoding> (visited on 12/13/2025).
- [7] *Ito calculus*. URL: https://en.wikipedia.org/wiki/It%C3%B4_calculus (visited on 12/07/2025).
- [8] *Kullback-Leibler Divergence*. URL: https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence (visited on 12/07/2025).
- [9] Hsin-Yi Lin et al. *Quantum Super-resolution by Adaptive Non-local Observables*. 2026. arXiv: [2601.14433](https://arxiv.org/abs/2601.14433) [quant-ph]. URL: <https://arxiv.org/abs/2601.14433>.
- [10] *Quantum Diffusion Model*. URL: <https://github.com/AndreaCacioppo/quantum-diffusion-models-code> (visited on 12/13/2025).
- [11] Sukin Sim, Peter D. Johnson, and Alán Aspuru-Guzik. “Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms”. In: *Advanced Quantum Technologies* 2.12 (Oct. 2019). ISSN: 2511-9044. DOI: [10.1002/qute.201900070](https://doi.org/10.1002/qute.201900070). URL: <http://dx.doi.org/10.1002/qute.201900070>.
- [12] Farrokh Vatan and Colin Williams. “Optimal quantum circuits for general two-qubit gates”. In: *Physical Review A* 69.3 (Mar. 2004). ISSN: 1094-1622. DOI: [10.1103/PhysRevA.69.032315](https://doi.org/10.1103/PhysRevA.69.032315). URL: <http://dx.doi.org/10.1103/PhysRevA.69.032315>.