SCALING QUANTUM MACHINE LEARNING WITHOUT TRICKS: HIGH-RESOLUTION AND DIVERSE IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Quantum generative modeling is a rapidly evolving discipline at the intersection of quantum computing and machine learning. Contemporary quantum machine learning is generally limited to toy examples or heavily restricted datasets with few elements. This is not only due to the current limitations of available quantum hardware but also due to the absence of inductive biases arising from application-agnostic designs. Current quantum solutions must resort to *tricks* to scale down high-resolution images, such as relying heavily on dimensionality reduction or utilizing multiple quantum models for low-resolution image patches. Building on recent developments in classical image loading to quantum computers, we circumvent these limitations and train quantum Wasserstein GANs on the established classical MNIST and Fashion-MNIST datasets. Using the complete datasets, our system generates full-resolution images across all ten classes and establishes a new state-of-the-art performance with a single end-to-end quantum generator without tricks. As a proof-of-principle, we also demonstrate that our approach can be extended to color images, exemplified on the Street View House Numbers dataset. We analyze how the choice of variational circuit architecture introduces inductive biases, which crucially unlock this performance. Furthermore, enhanced noise input techniques enable highly diverse image generation while maintaining quality. Finally, we show promising results even under quantum shot noise conditions.

1 INTRODUCTION

Since the advent of ChatGPT (OpenAI, 2025), generative modeling has become one of the most used technologies in the world (Paris, 2023). From coding "copilots" (Yao, 2023) to the generation of realistic-looking images (OpenAI, 2022), or musical compositions (OpenAI, 2019), generative AI is continuously gaining fields of applications, with increasing computation and energy demands (Jegham et al., 2025). Quantum generative modeling (Schuld & Petruccione, 2021) is an emerging field at the intersection of quantum computing and machine learning, focused on using quantum systems to learn, model, and sample from complex data distributions. Just as classical generative models, e.g. Variational Autoencoders (VAEs) (Kingma & Welling, 2014), Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), or Transformers (Vaswani et al., 2017), learn to mimic data distributions, quantum generative models aim to leverage the probabilistic and high-dimensional nature of quantum mechanics to achieve outcomes, potentially superior and intractable for classical computers (Huang et al., 2025). Although the potential advantages of applying quantum generative models to practical problems remain uncertain in terms of performance, there are indications that such systems can be energetically more efficient (Villalonga et al., 2020). Thus, it is crucial to investigate their capabilities on relevant machine learning benchmark tasks empirically.

Image generation is a particularly interesting use case of generative modeling. For example, data augmentation (Islam et al., 2024) for artificial vision systems is used in diverse fields ranging from medical diagnose systems (Motamed et al., 2021) to quality assurance (Wang et al., 2023), in which neural networks are trained to recognize illness or defective parts or products. In both cases, such anomalous images are usually difficult to obtain naturally and synthetic examples need to be created.

State-of-the-art methods for quantum image generation rely on *tricks* to circumvent scaling issues related to high-dimensional (high-resolution) images. We recognize two widely used techniques:

1. *Dimensionality Reduction:* This method uses principal component analysis (PCA) (Stein et al., 2021; Silver et al., 2023; Solanki et al., 2024; Khatun et al., 2024) or neural networks, including autoencoders, (Rudolph et al., 2022; J et al., 2022; Shu et al., 2024; Ma et al., 2025) to generate images in a lower-dimensional *latent* space. The output of the small quantum model is then classically post-processed to recover the original image dimensions.
2. *Patch generation:* This method circumvents high dimensionality by generating smaller patches of the images, where each patch uses a separate quantum generator, usually trained simultaneously (Huang et al., 2021; Tsang et al., 2023; Thomas & Jose, 2024).

Importantly, both methods circumvent high-dimensional data by generating low-dimensional quantum model outputs and may supplement them with classical computation to recover the original image dimensions. As a result, it becomes unclear whether the quantum model plays a non-trivial role in the generation. This is particularly true for the first method type, where a neural network may cover most of the generation. Thus, we consider Tsang et al. (2023), a patch-generation QGAN with one quantum generator per image row, as the previous state-of-the-art and baseline for comparison. Notably, despite these tricks, prior QGANs suffered from limited visual quality and diversity, producing scattered pixels and unrealistic class mixing even on three-class datasets. By presenting a single end-to-end quantum generator for diverse images at full resolution, we provide evidence for the capability and scalability of quantum generative modeling when appropriately designed.

Data of interest are often not arbitrary and have some internal structure, e.g., natural occurring images differ from random pixels. In fact, real images are known to have low-rank structure, evidenced in their fast decreasing power spectrum (van der Schaaf & van Hateren, 1996). This allows for compression algorithms such as JPEG (Wallace, 1992), which is a popular format in classical computing. This structure carries out to the quantum realm, as illustrated in several recent results (both numerical and theoretical) showing that their underlying structure leads to encoding quantum states that are well-captured by tensor-network states and by tensor-network-inspired quantum circuits (Dilip et al., 2022; Iaconis & Johri, 2023; Jobst et al., 2024; Shen et al., 2024). These states can thus be prepared with quantum circuits of depth linear in the number of qubits required for the encoding.

Prior research has explored various aspects of quantum image processing, including the identification of effective quantum encodings (Jobst et al., 2024), the generation of large-scale datasets through quantum circuit-based image encoding (Kiwit et al., 2025), and the application of quantum models to classification tasks (Shen et al., 2024; Kiwit et al., 2025).Here, we present a single end-to-end image quantum generator based on a quantum GAN (QGAN) training with a classical discriminator. In our approach, we use no dimensionality reduction methods and no multiple generators for image patches, and tackle large datasets commonly used in the machine learning field for benchmarking: MNIST (Lecun et al., 1998), Fashion-MNIST (Xiao et al., 2017), and Street View House Numbers (SVHN) (Netzer et al., 2011), for color images. This is possible due to the inductive bias created by an application-specific quantum circuit design inspired by the exponentially compressed encoding scheme. Moreover, we show that multimodal noise input increases the diversity of the generated images. We further explore the performance of training in the presence of shot noise.

2 Background: Quantum Image Representations

The simplest way to encode classical data into the amplitudes of a quantum state is referred to as *amplitude encoding* (Schuld & Petruccione, 2021; Latorre, 2005) that is given by $|\psi(\boldsymbol{x})\rangle = \frac{1}{\|\boldsymbol{x}\|}\sum_{j=0}^{2^N-1} x_j |j\rangle$, where $\boldsymbol{x}$ represents some classical data vector. (For notation conventions, see App. A). This encoding is attractive because it allows for representing an image with 2^n pixels using only n qubits, leading to an exponential reduction in storage requirements compared to a classical representation. Since the state must be normalized, the global scaling information is lost in the encoding. To address this limitation, encodings of the following form have been proposed (Le et al., 2011a;b) for images with 2^N pixels:

$$|\psi(\boldsymbol{x})\rangle = \frac{1}{\sqrt{2^N}} \sum_{j=0}^{2^N-1} |c(\boldsymbol{x}_j)\rangle \otimes |j\rangle \,. \tag{1}$$

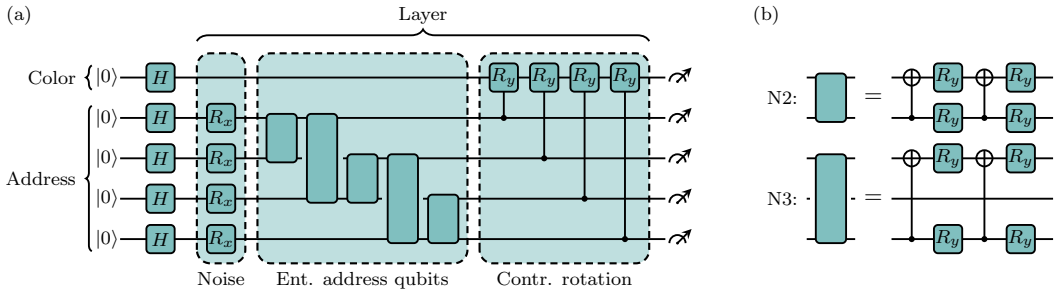


Figure 1: (a) Quantum generator for a 4×4-pixel grayscale image with one layer of noise, entangling and controlled R_y gates. The nearest-neighbor (N2) and next-nearest-neighbor (N3) entangling gates are applied on the address qubits. The final R_y gates rotate the color qubit, each controlled by one address qubit. (b) Gate decompositions of the N2 and N3 entangling gates into CNOT and R_y.

The state $|j\rangle$ of the N so-called *address qubits* tracks the position index j and the state $|c(\boldsymbol{x}_j)\rangle$ encodes the corresponding data value $\boldsymbol{x}_j$. For grayscale images, we use the *flexible representation of quantum images (FRQI)* (Le et al., 2011a;b). In this encoding, $\boldsymbol{x}_j$ is a scalar with the grayscale value of that pixel. We encode this information in the z-polarization of the color qubit

$$|c(x_j)\rangle = \cos(\tfrac{\pi}{2} x_j)\,|0\rangle + \sin(\tfrac{\pi}{2} x_j)\,|1\rangle\,, \tag{2}$$

with the pixel value normalized to $x_j \in [0, 1]$. Thus, combining Eqs. (1) and (2), a 2^N-pixel image is encoded into a state with $N + 1$ qubits. FRQI has been extended to *multi-channel representation of quantum images (MCRQI)* (Sun et al., 2011; 2013) for color images, as detailed in App. B.4.

The order in which the pixels are indexed can change the entanglement entropy of the resulting state (Jobst et al., 2024). Here, we choose hierarchical indexing based on the so-called Z- or Morton order (Latorre, 2005; Le et al., 2011a;b; Jobst et al., 2024): the first two bits of the index j label the quadrant of the image the pixel is in, the next two bits label the subquadrant, and so on. This tends to decrease the entanglement entropy compared to other orderings, resulting in more compressible states (see Jobst et al. (2024) for grayscale images and Kiwit et al. (2025) for color images).

3 Method

In GANs, the generator $G_{\boldsymbol{\theta}}(\boldsymbol{z}) \mapsto \boldsymbol{x}$ aims to map a noise vector $\boldsymbol{z}$ to a sample $\boldsymbol{x}$, indistinguishable from real data, while the discriminator $D_{\boldsymbol{\phi}}(\boldsymbol{x})$ aims to differentiate between real and fake samples. In our setup, the generator is a quantum circuit while the discriminator is a classical convolutional neural network. Both are trained jointly using the gradient-penalized Wasserstein GAN (Gulrajani et al., 2017) scheme. More details are provided in App. B, while we focus here on our main methodological contribution: the design of the quantum generator G including an enhanced noise input.

Application-specific generator design. The quantum generator employs a circuit ansatz with an inductive bias tailored towards the FRQI representation. Analogously for MCRQI, a color-extended task-specific ansatz is proposed in App. B.4. The generator ansatz starts with a layer of Hadamard gates to bring the initial state $|0\rangle^{\otimes n}$ into an equal superposition, which resembles a valid FRQI state of a uniformly gray image. After the Hadamard gates, (multiple) layers of the generator are added. Each layer consists of (i) noise gates, (ii) gates that entangle the the address qubits, and (iii) controlled rotations of the color qubits, as depicted in Fig. 1 and described in the following.

First, the noise is injected at the beginning of each layer by parameterized single-qubit R_x gates (Definitions of quantum gates are provided in App. A). Details on the noise encoding with additional learnable parameters and multiple modes are provided in the second part of this section.

Second, entangling gates are arranged as a ladder alternating between connecting nearest-neighbor (N2) and next-nearest-neighbor (N3) address qubits. Due to the Morton order, as described in Sec. 2, N2 gates mix qubits addressing two different spatial dimensions (vertical and horizontal). Consequently, N3 gates only mix between qubits addressing the same spatial dimension at different scales.

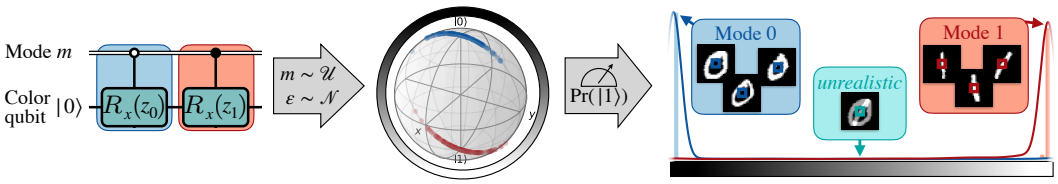


Figure 2: Illustration of multimodal noise modeling (left to right). Quantum circuit perspective of implementing a bimodal mixture distribution via controlled rotations sampling the classical bit m uniformly and ε normally (unimodal). z_0 and z_1 denote the tuned noise (shifted by 0 and π, respectively). In this single-pixel example, noise is injected directly into the color qubit (no address qubits or layering), so layer and qubit indices l, n as in Eq. (3) are omitted. The noise separates the prepared states around $|0\rangle$ and $|1\rangle$ in the Bloch sphere. Measurements yield pixel values via the probability of $|1\rangle$, consistent with FRQI states in Eq. (2). As an example, the distribution resembles the bimodal statistics of the MNIST center pixel for handwritten digits *0* and *1*, with peaks at *0* (black) and *1* (white) and vanishing probability in between, avoiding unrealistic gray pixels.

We refer to repeating these ladders ℓ times as introducing ℓ *sub-layers*. Each sub-layer uses distinct parameters and alternates the direction of qubit connections between top-down and bottom-up. The address qubits are entangled by parameterized two-qubit gates of the form shown in Fig. 1b. These gates realize compressed orthogonal two-qubit transformations, which have proven effective for encoding FRQI states (Kiwit et al., 2025).

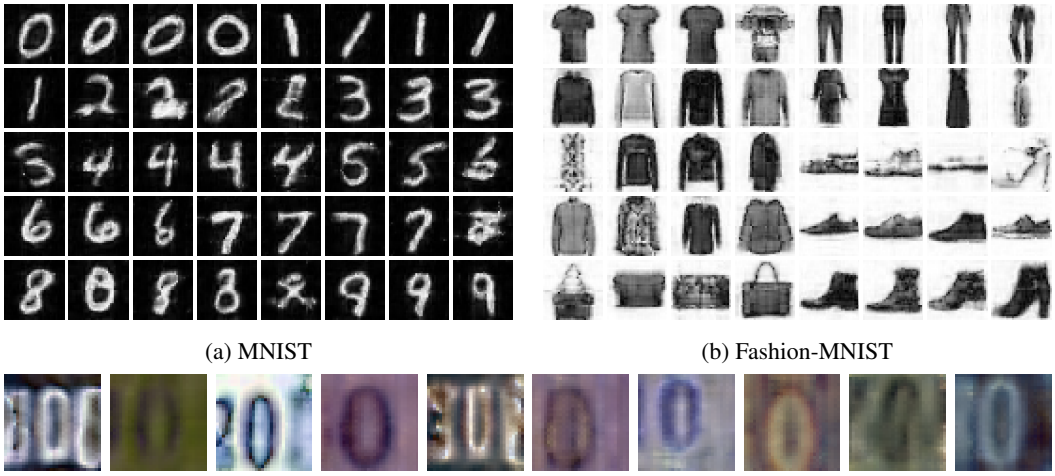
Third, after entangling the address qubits, we rotate the color qubit via parametrized R_y gates, controlled by a single address qubit, that modulates the color of half the pixels in the image while leaving the other half unchanged. The pixels that are affected are those whose corresponding address bit is set to one in the binary representation of their index. More controls (k) affect a smaller fraction ($1/2^k$) of pixels simultaneously. These controlled rotations after the entangling gates let multiple address qubits influence color rotations, emulating extra control qubits. More complex entanglement structures can induce color channel modulations within a specific region and details of the image.

As a final step, the state generated by the quantum circuit must be decoded into an image. It is essential to note that the ansatz does not enforce valid FRQI states, i.e., neither nonnegative real amplitudes nor a uniform superposition over address qubits (uniform pixel distribution upon measurement) are guaranteed. Normalizing/conditioning the computational basis probabilities enables decoding as valid FRQI states via trigonometric inverse functions, as further detailed in App. B.2.

Enhanced noise input. The noise design critically determines the diversity and fidelity of generated samples as poorly chosen noise distributions limit the generator's ability to capture the data variability. Prior QGAN works (Riofrío et al., 2024; Tsang et al., 2023; Ma et al., 2025) use *noise re-uploading* (Pérez-Salinas et al., 2020) which enhances expressivity by inducing a more complex, non-linear dependence on the noise input. In contrast, we introduce two enhancements to the noise sampling and its injection (encoding into the generator layers), enabling more diverse and detailed image generation. While a Gaussian distribution is *unimodal*, with a single peak at its mean, a *multimodal* distribution has multiple peaks, inducing multiple high-probability regions. Pixel distributions of natural images exhibit such multimodality. For example, the distribution over the central pixel of MNIST digits *0* and *1* as depicted in Fig. 2 shows peaks at black and at white.

In the following, we introduce our *noise tuning* technique to use a multimodal noise distribution, inspired by the reparametrization trick (Kingma & Welling, 2014). To generate a sample $\boldsymbol{x}$ we sample the noise vector from a multivariate isotropic Gaussian $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}_N)$, where N is the number of address qubits. The same noise vector is shared across all L generator layers. We then sample the mode index from a discrete uniform distribution $m \sim \mathcal{U}\{1, M\}$ and select the corresponding reparameterization matrices $\mu_m, \sigma_m \in \mathbb{R}^{L \times N}$ that are part of the learnable generator parameters $\boldsymbol{\theta}$. Finally, we apply the element-wise affine transformation $\boldsymbol{z}_{m,l} = \mu_{m,l} + \sigma_{m,l} \odot \boldsymbol{\varepsilon}$ with $l \in \{1, \ldots, L\}$. This results in a (uniform) Gaussian mixture model $\boldsymbol{z} \sim \frac{1}{M} \sum_{m=1}^{M} \mathcal{N}(\boldsymbol{z} \mid \boldsymbol{\mu}_m, \mathrm{diag}(\boldsymbol{\sigma}_m^2))$, where $\boldsymbol{\mu}_m, \boldsymbol{\sigma}_m \in \mathbb{R}^{LN}$ correspond to the flattened matrices μ_m and σ_m. For each single noise component with $n \in \{1, \ldots, N\}$, this can be represented by rotations on address qubit n of the form

$$-\boxed{R_x(z_{m,l,n})}- \quad = \quad -\boxed{R_x(\mu_{m,l,n} + \sigma_{m,l,n}\varepsilon_n)}-. \tag{3}$$



(a) MNIST (b) Fashion-MNIST

(c) SVHN (class *0*)

Figure 3: QGAN samples for (a) MNIST, (b) Fashion-MNIST, and (c) SVHN. For (a) and (b), one image is shown for each of the 40 noise modes used by the large QGANs (64 layers). For each mode, the displayed image is selected as closest to the mean of 500 samples in Euclidean distance. For (c), a 32-layer QGAN generates images restricted to containing the digit *0*. The central digit is consistently a *0*, while extra digits may occur on the sides, reflecting typical house number tags.

In terms of its quantum circuit implementation, this tuning corresponds to M rotation gates encoding unimodal noise components but controlled by *classical* bits encoding the sampled mode index m to realize each mode via a separate controlled gate layer. Figure 2 presents the bimodal case. To the best of our knowledge, multimodal latent distributions have only been explored implicitly in quantum *conditional* models (Liu et al., 2021; Zeng et al., 2023), and their explicit treatment together with noise tuning is novel in QGANs, with only classical analogues reported (Gurumurthy et al., 2017).

4 Results

We designed the experiments with three main objectives: (i) demonstrating the high quality and diversity of the QGAN image generation, (ii) analyzing the impact of our QGAN design choices, and (iii) assessing the transferability to future quantum computers under inevitable shot noise in the generation process. All experiments are conducted in numerical simulation. We evaluate our approach using standard image datasets, including the grayscale MNIST (Lecun et al., 1998; Deng, 2012) and Fashion-MNIST (Xiao et al., 2017) datasets. These datasets contain ten classes of different handwritten digits and clothing photos, respectively. Both have a resolution of 28×28 pixels and are interpolated (bilinear) to 32×32 pixels to match 11-qubit FRQI states. The 32×32-pixel Street View House Numbers (SVHN) color images (Netzer et al., 2011) are represented by 13-qubit MCRQI states. Further details on the datasets are provided in App. C. All images presented are generated from QGANs trained for a fixed number of iterations or, when stated, loaded from a checkpoint that minimizes the maximum mean discrepancy (MMD; see App. D.1). For clarity, images are manually ordered and, where relevant, matched to classes. We vary and indicate the number of generator layers, but place two sub-layers each. Implementation details can be found in App. B.

4.1 Generating samples of high quality and diversity

To demonstrate image generation of high quality and diversity, we train large QGAN models with 64 layers and 40 noise modes for about 50 000 generator updates on the full MNIST and Fashion-MNIST datasets and present the checkpoint that minimizes the MMD metric. As shown in Fig. 3, not only are all ten classes successfully captured with high visual quality, but images also reveal rich intra-class diversity. The depth of the models enables them to represent fine image structures in digits (Fig. 3a), or extreme cases such as the single-pixel-wide straps in the *sandals* class (Fig. 3b), which demand more complex entanglement among the address qubits. The size of the quantum

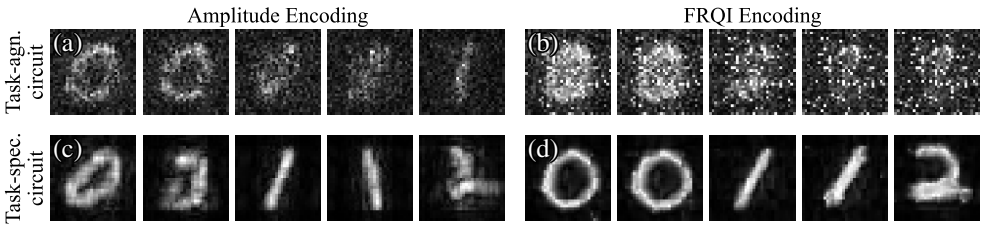


Figure 4: Ablation study highlighting the importance of task-specific model design choices. Panels (a) and (b) show images from the task-agnostic circuit using Amplitude encoding and FRQI encoding, respectively. Panels (c) and (d) show images from the task-specific circuit using Amplitude encoding and FRQI encoding, respectively. Task-specific modifications yield clearer, less distorted digit representations, with combining both proposed design choices leading to the best results (d).

generator may appear large relative to previous works. However, since these works only covered small subsets of classes within these datasets, less expressive models suffice. Similarly, our QGAN framework also learns high-quality images with shallower circuits on these subsets. In App. D.2, we analyze this trade-off in more detail and observe that deeper models are necessary not only to improve image quality on a fixed dataset but also to maintain quality when scaling to all classes.

A colorful extension. The model is also trained on the color dataset, SVHN, restricted to images containing the digit *0*. In this setting, the *0* consistently occupies the central position, while additional digits may appear on the left and right. Consequently, the surrounding context introduces variability, as house numbers naturally contain multiple digits, and the background colors may also differ. Fig. 3c illustrates representative results from a QGAN model with 32 layers of the color-extended task-specific ansatz and 3 modes, trained for nearly 100 000 iterations and evaluated via MMD. One can observe that the central digit is reliably reconstructed as a *0*, while digits occurring to the left often resemble *2*s or *3*s, reflecting the realistic distribution present in the dataset.

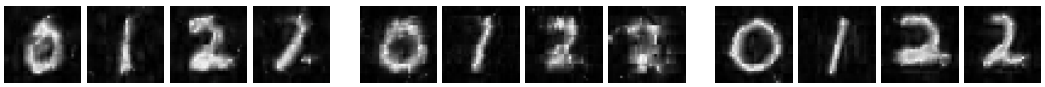
4.2 Impact of task-specific generator design choices

We analyze the impact of the two main design choices in the presented QGAN framework, concerning the generator design and noise techniques, through additional experiments.

Task-specific generator design ablation study. We evaluate the relevance of two generator design choices specific to the task of image generation: (i) the generator circuit ansatz specific to the image state encoding instead of a task-agnostic ansatz, and (ii) the FRQI state representation over simple amplitude image encoding. Compared to the layers in the task-specific ansatz, the task-agnostic ansatz implements entanglement via fixed cyclic N2 controlled-NOT gates, while parameterization occurs only in single-qubit $z - y - z$ rotation sequences. We perform an ablation study that compares the results of QGANs where these design choices are either implemented or omitted. All combinations use 16 layers, and are trained for 15 000 iterations on the digits *0*, *1* and *2*. Furthermore, the enhanced noise inputs (3 modes) may improve even the amplitude encoding and task-agnostic ansatz combination, which most closely resembles the setup by Tsang et al. (2023).

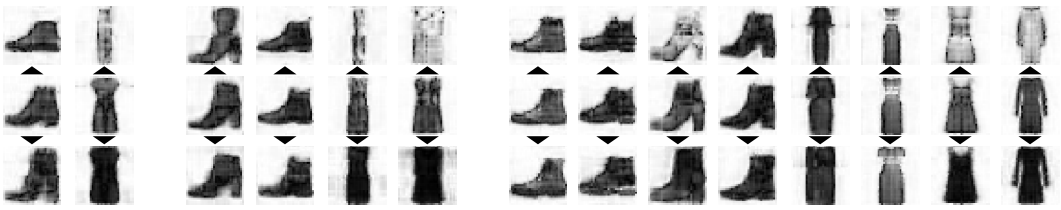
Figure 4 shows the results, revealing the impact of the two design choices. The most pronounced difference in image quality arises from the ansatz choice. The task-agnostic ansatz (Fig. 4a, 4b) produces images with a vague glimpse of digits. Furthermore, this ansatz produces images of limited diversity, particularly omitting classes, such as digit *2*. Formally, this corresponds to mode collapse, which limits QGANs with task-agnostic ansätze from scaling to more classes, as in previous works limited to at most three classes. The task-specific ansatz (Fig. 4c, 4d) clearly achieves what the task-agnostic one fails to model: spatial coherence and defined edges—two main properties of natural images (Simoncelli & Olshausen, 2001). Hence, neighboring pixels exhibit similar colors, with edges clearly defined rather than being fuzzy.

For the image encoding choice, the overall contrast of the digits from the black background is improved when transitioning from amplitude (Fig. 4a, 4c) to FRQI encoding (Fig. 4b, 4d). We observe



(a) Unimodal (b) Multimodal w/o noise tuning (c) Multimodal w/ noise tuning

Figure 5: Comparison of noise inputs: (a) unimodal, (b) fixed multimodal, (c) tuned multimodal. Models were trained on MNIST classes *0–2*, with 3 modes in the multimodal setups. Images are generated after 15 000 training iterations and manually selected to highlight characteristic effects.



(a) $\times 1$ ms./c. (b) $\times 2$ modes/class (c) $\times 4$ modes/class

Figure 6: More input noise modes ("overmoding") diversify generated samples. Three models are trained on all ten Fashion-MNIST classes with a factor of (a) 1, (b) 2, and (c) 4 more noise modes than classes. We present all modes capturing the classes *ankle boot* and *dress*. Three images are shown per mode: the center image is closest to the mean in Euclidean distance, while the outer images closely approximate moves of $\pm 3\sigma$ along the first principal component (indicated by arrows). PCA, based on 1 000 samples per mode, illustrates the primary variability within each mode.

that the saturation is more balanced across different samples and more uniform within each digit. These results support the theoretical expectation of sensitivity in saturation for amplitude-encoded images due to the need of amplitude normalization. FRQI encoding handles the saturation by introducing the color qubit. In addition, the edges are less blurred when switching from amplitude to FRQI encoding under the image-specific ansatz. We tested two image-specific ansatz realizations for amplitude encoding (Fig. 4c): one omits the layer of controlled color-qubit rotations, while the other replaces it with a layer of single-qubit rotations. No substantial differences were observed in the generated images.

From unimodal to multimodal noise through tuning. In the following, we will discuss the role of input noise distributions and injection techniques, centered around generated images from three different experiments presented in Fig. 5. Given that previous QGAN works relied solely on unimodal noise distributions, we start the analysis with unimodal Gaussian noise (Fig. 5a). Pure blending by simply superimposing images of two classes (see *0*s where the inside of the circle is not transparent, e.g., leftmost image in Fig. 5a) is observed less frequently than in previous works (Tsang et al., 2023), which might be due to an improved generator design. However, more pronounced class mixing effects manifest as morphing shapes of distinct classes, such as *1*s appearing as right-leaning with curved tops and faint bottom bars reminiscent of *2*s (rightmost image in Fig. 5a). Although unimodal noise does not suffer from strict mode collapse onto a single digit, we conclude that scaling to datasets with many diverse classes is infeasible.

Introducing a multimodal distribution with three fixed modes (matching the number of classes included for training) mitigates these two mixing effects (Fig. 5b). However, this change is accompanied by a considerable loss in image quality, often obscuring visual class differentiation either (rightmost image in Fig. 5b). A likely reason is that sampling from modes placed at fixed μ_j away from zero results in noise injections that disrupt state preparation due to a systematic rotation in each layer, which the model can control only to a limited extent. Therefore, the proposed *noise tuning* technique, where the mode centers μ_j and widths σ_j effectively become learnable parameters, is crucial for multimodality, generating clearly separated and undistorted images (Fig. 5c).

More modes than classes ("overmoding"). Choosing the number of modes equal to the number of classes is natural, however this information is unavailable in unsupervised datasets. Moreover, in-

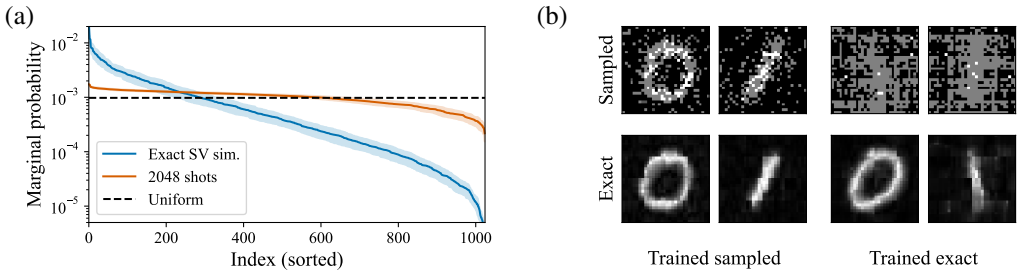


Figure 7: (a) Marginal probabilities of the address qubits sorted by magnitude. Exact state-vector simulations (blue) deviate strongly from the expected uniform distribution (dashed), with many amplitudes nearly zero, whereas finite sampling with 2 048 shots (orange) smooths the distribution toward uniformity. (b) Examples generated from 2 048 shots (top) and from exact probabilities (bottom). Finite-shot sampling introduces statistical noise that smooths the distribution and preserves pixel information. Hence, models trained on sampled data (left) yield clearer, more robust digits, while models trained on exact probabilities (right) tend to produce incomplete or distorted images.

stances of the same class may exhibit very different features (high intra-class variety), and modeling them with more than a single mode might be an appropriate choice. By analogy to *overparameterization*, we call the use of an ansatz with a potential excess of modes *overmoding*. To analyze the effects of overmoding, we train three QGANs on the complete Fashion-MNIST datasets with $\times 1$, $\times 2$, and $\times 4$ more modes than classes for 20 000 iterations (nearly 40 000 in the latter case). Fig. 6 shows generated images after training for the classes *ankle boot* and *dress*, corresponding to three models with 1, 2, and 4 modes per class.

Across all classes, increasing the number of modes enhances intra-class diversity by allowing the model to represent distinct sub-classes. A single mode (Fig. 6a) may already capture some variation, but typically sacrifices visual quality. In contrast, overmoding benefits both diversity and quality. With two modes (Fig. 6b), the model already separates flat vs. heeled boots and short vs. long dresses, which were previously conflated in a single mode. At four modes (Fig. 6c), the separation becomes more fine-grained. For *boots*, one heeled mode varies heel type (from stiletto, via block, to wedge), while another varies heel height. Flat-boot modes capture distinct styles, differing in details such as laces, soles, and pull tabs. *Dresses* are distinguished by sleeve type (long, short, cap, sleeveless/straps) and further vary in length within each mode. The fourth *dress* mode (Fig. 6c) transitions into the *coat* class by altering shape and introducing a zipper line. This overlap highlights the benefit of not conditioning QGAN modes on class labels, allowing the unsupervised model to exploit shared visual structures across classes. More *inter*-class modes are presented in App. D.3.

4.3 Finite measurement shot effects

The marginal distribution of the address qubits of valid FRQI states, after tracing out the color qubit, is uniform due to the sine–cosine structure in Eq. (2). In exact state-vector simulations without shot noise, the quality of the generated samples depends only on the ratio, rather than the absolute values, of the probability amplitudes of $|0\rangle$ and $|1\rangle$ in the color qubit for a given address. However, some basis states may have vanishingly small amplitudes in both $|0\rangle$ and $|1\rangle$. With a finite number of shots, such states are unlikely to be sampled, causing loss of pixel information. Incorporating finite shot noise during training may alleviate this problem. Very low probabilities may exclude information from some pixels, making it easier for the discriminator to detect fake samples and forcing the generator to avoid such cases and thus promoting more uniformly distributed marginal probabilities over the address qubits. Details of our implementation are presented in App. B.5.

Figure 7a illustrates how exact state-vector simulations yield highly uneven marginal probabilities across pixels, with many basis states exhibiting vanishingly small amplitudes. By contrast, sampling with a finite number of shots (2 048 in this example) smooths out the distribution and keeps the probabilities closer to the expected uniform distribution, thereby mitigating the risk of pixels being systematically excluded. This effect also shows in the sampled images in Fig. 7b, where finite shot noise ensures that pixel information is retained more consistently across the image. Together, these

results highlight that incorporating shot noise into training not only prevents the discriminator from exploiting missing pixels but also promotes more robust and uniform sampling behavior.

5 Discussion

In this work, we have made several contributions advancing quantum generative modeling. First, we demonstrated that end-to-end quantum Wasserstein GANs can be trained directly on full-resolution, standard classical image datasets without resorting to dimensionality reduction or patch-wise modeling, thus moving beyond the toy examples that have historically constrained the field. Second, we showed that performance depends critically on the incorporation of inductive biases through carefully designed variational circuit architectures, rather than relying on generic, application-agnostic ansätze. Our findings highlight that task-specific architectural choices are not only a technical detail but a central driver of scalability and generative quality in quantum machine learning. Finally, by benchmarking our training under realistic shot-noise conditions, we provide a practical pathway toward robust quantum image generation. Together, these contributions underscore that progress in quantum generative modeling will come not only from hardware advances but also from principled design choices that align quantum models with the structure of the task.

A common criticism of quantum generative modeling with image encodings such as FRQI is the apparent measurement overhead: while images of N pixels can be encoded using $O(\log(N))$ qubits, recovering a sampled image requires $O(N)$ measurements, seemingly negating the exponential qubit compression by introducing exponential costs at the decoding stage. At first glance, this appears to undermine one of the central motivations for FRQI-based models. However, several points help mitigate this concern. First, it is worth questioning whether this limitation is practically consequential: real-world images are captured by classical devices and do not need exponential classical resources, e.g., memory and time, for capturing or processing. Second, more sophisticated decoding strategies, exploiting known structure of natural images, may reduce the measurement burden while still recovering meaningful image statistics.

We propose the following three ideas for decoding strategies. One could use compressed sensing (Donoho, 2006; Candes & Tao, 2006; Candes et al., 2006) as a post-processing step. This method would act entirely classically: missing pixel intensities, i.e., non-measured states, can often be reconstructed from partial information using structural priors on natural images (Candes & Wakin, 2008; Duarte & Eldar, 2011). While this may not directly resolve the scaling challenge, exploring the asymptotic behavior of compressed sensing in this context could clarify the extent to which the number of required measurements can be meaningfully reduced. Alternatively, one could perform measurements in Fourier space. By applying the Quantum Fourier Transform to the address qubits, as suggested in the original FRQI framework (Le et al., 2011a;b), one could probe the frequency domain rather than pixel space. Since low-frequency components dominate natural images, higher-frequency qubits should naturally decouple, effectively concentrating measurement probability on the relevant low-frequency subspace. Finally, one could use shadow tomography techniques that leverage recent advances tailored to tensor-network states (Akhtar et al., 2023; Bertoni et al., 2024). By exploiting the limited bond dimension characteristic of natural images (Jobst et al., 2024), such methods could drastically reduce the shot complexity of retrieving useful image statistics with error bounds and theoretical guarantees. Pursuing these directions could recast measurement overhead from a perceived limitation into an opportunity for additional streamlining, further aligning quantum generative modeling with the structure of natural data. Exploring these decoding strategies is left for future work.

As a final reflection, it is striking to observe the disparity in resources required by quantum versus classical generative models for the datasets studied here. Our quantum approach achieves competitive synthetic data generation with only 11–13 qubits and on the order of ten thousand trainable parameters, whereas classical models typically rely on ten thousands of bits and hundreds of thousands of parameters. This contrast highlights the remarkable expressive power that quantum computing can bring to machine learning, and we view it as yet another indication of its potential to fundamentally reshape how generative modeling is conceived and implemented.

REPRODUCIBILITY STATEMENT

All experiments in this paper can be reproduced with the provided code and instructions. The complete codebase, which includes training scripts, evaluation notebooks, and configuration files, is included as part of the supplementary material for the submission. If the paper is accepted, we will make the code publicly available as a GitHub repository.

REFERENCES

Ahmed A. Akhtar, Hong-Ye Hu, and Yi-Zhuang You. Scalable and Flexible Classical Shadow Tomography with Tensor Networks. *Quantum*, 7:1026, June 2023. ISSN 2521-327X. doi: 10.22331/q-2023-06-01-1026. URL https://doi.org/10.22331/q-2023-06-01-1026.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN, December 2017.

Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, Shahnawaz Ahmed, Vishnu Ajith, M. Sohaib Alam, Guillermo Alonso-Linaje, B. AkashNarayanan, Ali Asadi, Juan Miguel Arrazola, Utkarsh Azad, Sam Banning, Carsten Blank, Thomas R Bromley, Benjamin A. Cordier, Jack Ceroni, Alain Delgado, Olivia Di Matteo, Amintor Dusko, Tanya Garg, Diego Guala, Anthony Hayes, Ryan Hill, Aroosa Ijaz, Theodor Isacsson, David Ittah, Soran Jahangiri, Prateek Jain, Edward Jiang, Ankit Khandelwal, Korbinian Kottmann, Robert A. Lang, Christina Lee, Thomas Loke, Angus Lowe, Keri McKiernan, Johannes Jakob Meyer, J. A. Montañez-Barrera, Romain Moyard, Zeyue Niu, Lee James O'Riordan, Steven Oud, Ashish Panigrahi, Chae-Yeun Park, Daniel Polatajko, Nicolás Quesada, Chase Roberts, Nahum Sá, Isidor Schoch, Borun Shi, Shuli Shu, Sukin Sim, Arshpreet Singh, Ingrid Strandberg, Jay Soni, Antal Száva, Slimane Thabet, Rodrigo A. Vargas-Hernández, Trevor Vincent, Nicola Vitucci, Maurice Weber, David Wierichs, Roeland Wiersema, Moritz Willmann, Vincent Wong, Shaoming Zhang, and Nathan Killoran. Pennylane: Automatic differentiation of hybrid quantum-classical computations. *arXiv:1811.04968*, July 2022. doi: 10.48550/arXiv.1811.04968. Code available at https://github.com/PennyLaneAI/pennylane.

Christian Bertoni, Jonas Haferkamp, Marcel Hinsche, Marios Ioannou, Jens Eisert, and Hakop Pashayan. Shallow shadows: Expectation estimation using low-depth random clifford circuits. *Phys. Rev. Lett.*, 133:020602, Jul 2024. doi: 10.1103/PhysRevLett.133.020602. URL https://link.aps.org/doi/10.1103/PhysRevLett.133.020602.

Ali Borji. Pros and cons of GAN evaluation measures. *Computer Vision and Image Understanding*, 179:41–65, February 2019. ISSN 1077-3142. doi: 10.1016/j.cviu.2018.10.009.

Ali Borji. Pros and Cons of GAN Evaluation Measures: New Developments, October 2021.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, March 2025. Code available at https://github.com/jax-ml/jax.

E.J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52 (2):489–509, 2006. doi: 10.1109/TIT.2005.862083.

Emmanuel J. Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, 52(12):5406–5425, 2006. doi: 10.1109/TIT.2006.885507.

Emmanuel J. Candes and Michael B. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, 2008. doi: 10.1109/MSP.2007.914731.

Li Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, November 2012. doi: 10.1109/MSP.2012.2211477.

Rohit Dilip, Yu-Jie Liu, Adam Smith, and Frank Pollmann. Data compression for quantum machine learning. *Physical Review Research*, 4:043007, October 2022. doi: 10.1103/PhysRevResearch.4.043007.

D.L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006. doi: 10.1109/TIT.2006.871582.

Marco F. Duarte and Yonina C. Eldar. Structured compressed sensing: From theory to applications. *IEEE Transactions on Signal Processing*, 59(9):4053–4085, 2011. doi: 10.1109/TSP.2011.2161982.

Neocognitron Fukushima. A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks, June 2014.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL http://jmlr.org/papers/v13/gretton12a.html.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved Training of Wasserstein GANs, December 2017.

Swaminathan Gurumurthy, Ravi Kiran Sarvadevabhatla, and R. Venkatesh Babu. DeLiGAN: Generative Adversarial Networks for Diverse and Limited Data. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4941–4949, Honolulu, HI, July 2017. IEEE. ISBN 978-1-5386-0457-1. doi: 10.1109/CVPR.2017.525.

He-Liang Huang, Yuxuan Du, Ming Gong, Youwei Zhao, Yulin Wu, Chaoyue Wang, Shaowei Li, Futian Liang, Jin Lin, Yu Xu, Rui Yang, Tongliang Liu, Min-Hsiu Hsieh, Hui Deng, Hao Rong, Cheng-Zhi Peng, Chao-Yang Lu, Yu-Ao Chen, Dacheng Tao, Xiaobo Zhu, and Jian-Wei Pan. Experimental Quantum Generative Adversarial Networks for Image Generation, September 2021. URL http://arxiv.org/abs/2010.06201. arXiv:2010.06201.

Hsin-Yuan Huang, Michael Broughton, Norhan Eassa, Hartmut Neven, Ryan Babbush, and Jarrod R. McClean. Generative quantum advantage for classical and quantum problems, 2025. URL https://arxiv.org/abs/2509.09033.

Jason Iaconis and Sonika Johri. Tensor network based efficient quantum data loading of images. *arXiv:2310.05897*, October 2023. doi: 10.48550/arXiv.2310.05897.

Tauhidul Islam, Md. Sadman Hafiz, Jamin Rahman Jim, Md. Mohsin Kabir, and M.F. Mridha. A systematic review of deep learning data augmentation in medical imaging: Recent advances and future research directions. *Healthcare Analytics*, 5:100340, 2024. ISSN 2772-4425. doi: https://doi.org/10.1016/j.health.2024.100340. URL https://www.sciencedirect.com/science/article/pii/S277244252400042X.

Arun Pandian J, Kanchanadevi K, Vadem Chandu Mohan, Pulibandla Hari Krishna, and Edagottu Govardhan. Quantum Generative Adversarial Network and Quantum Neural Network for Image Classification. In *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, pp. 473–478, April 2022. doi: 10.1109/ICSCDS53736.2022.9760943. URL https://ieeexplore.ieee.org/abstract/document/9760943.

Nidhal Jegham, Marwan Abdelatti, Lassad Elmoubarki, and Abdeltawab Hendawi. How hungry is ai? benchmarking energy, water, and carbon footprint of llm inference, 2025. URL https://arxiv.org/abs/2505.09598.

Bernhard Jobst, Kevin Shen, Carlos A. Riofrío, Elvira Shishenina, and Frank Pollmann. Efficient MPS representations and quantum circuits from the Fourier modes of classical image data. *Quantum*, 8:1544, December 2024. ISSN 2521-327X. doi: 10.22331/q-2024-12-03-1544. URL https://quantum-journal.org/papers/q-2024-12-03-1544/.

Amena Khatun, Kübra Yeter Aydeniz, Yaakov S. Weinstein, and Muhammad Usman. Quantum Generative Learning for High-Resolution Medical Image Generation, June 2024. URL http://arxiv.org/abs/2406.13196. arXiv:2406.13196.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, 2014. URL http://arxiv.org/pdf/1412.6980v9.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Florian J. Kiwit, Bernhard Jobst, Andre Luckow, Frank Pollmann, and Carlos A. Riofrío. Typical machine learning datasets as low-depth quantum circuits, 2025. URL http://iopscience.iop.org/article/10.1088/2058-9565/ae0123.

Jose I. Latorre. Image compression and entanglement. *arXiv:quant-ph/0510031*, October 2005. doi: 10.48550/arXiv.quant-ph/0510031.

Phuc Q. Le, Fangyan Dong, and Kaoru Hirota. A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. *Quantum Information Processing*, 10:63–84, February 2011a. doi: 10.1007/s11128-010-0177-y.

Phuc Q. Le, Abdullahi M. Iliyasu, Fangyan Dong, and Kaoru Hirota. *A Flexible Representation and Invertible Transformations for Images on Quantum Computers*, volume 372, pp. 179–202. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011b. ISBN 978-3-642-11739-8. doi: 10.1007/978-3-642-11739-8_9.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998. doi: 10.1109/5.726791.

Wenjie Liu, Ying Zhang, Zhiliang Deng, Jiaojiao Zhao, and Lian Tong. A hybrid quantum-classical conditional generative adversarial network algorithm for human-centered paradigm in cloud. *EURASIP Journal on Wireless Communications and Networking*, 2021(1):37, February 2021. ISSN 1687-1499. doi: 10.1186/s13638-021-01898-3.

QuanGong Ma, ChaoLong Hao, NianWen Si, Geng Chen, Jiale Zhang, and Dan Qu. Quantum adversarial generation of high-resolution images. *EPJ Quantum Technol.*, 12(1):3, December 2025. ISSN 2662-4400, 2196-0763. doi: 10.1140/epjqt/s40507-024-00304-3. URL https://epjqt.epj.org/articles/epjqt/abs/2025/01/40507_2024_Article_304/40507_2024_Article_304.html. Number: 1 Publisher: Springer Berlin Heidelberg.

K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Phys. Rev. A*, 98: 032309, Sep 2018. doi: 10.1103/PhysRevA.98.032309. URL https://link.aps.org/doi/10.1103/PhysRevA.98.032309.

Saman Motamed, Patrik Rogalla, and Farzad Khalvati. Data augmentation using generative adversarial networks (gans) for gan-based detection of pneumonia and covid-19 in chest x-ray images. *Informatics in Medicine Unlocked*, 27:100779, 2021. ISSN 2352-9148. doi: https://doi.org/10.1016/j.imu.2021.100779. URL https://www.sciencedirect.com/science/article/pii/S2352914821002501.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2011. ISBN 9781107002173. URL https://www.amazon.com/Quantum-Computation-Information-10th-Anniversary/dp/1107002176?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=1107002176.

OpenAI. Musenet: Creating four-minute musical compositions with up to ten instruments. https://openai.com/index/musenet/, April 2019.

OpenAI. Dall·e 2: Creating more realistic and accurate images. https://openai.com/index/dall-e-2/, April 2022.

OpenAI. Chatgpt. https://chat.openai.com/, 2025. Large language model developed by OpenAI, based on the GPT-4 and GPT-5 architectures.

Martine Paris. Chatgpt hits 100 million users, google invests in ai bot and catgpt goes viral, February 2023. URL https://www.forbes.com/sites/martineparis/2023/02/03/chatgpt-hits-100-million-microsoft-unleashes-ai-bots-and-catgpt-goes-viral/. Accessed: 2025-09-02.

Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José I. Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, February 2020. ISSN 2521-327X. doi: 10.22331/q-2020-02-06-226.

Carlos A. Riofrío, Oliver Mitevski, Caitlin Jones, Florian Krellner, Aleksandar Vuckovic, Joseph Doetsch, Johannes Klepsch, Thomas Ehmer, and Andre Luckow. A Characterization of Quantum Generative Models. *ACM Transactions on Quantum Computing*, 5(2):12:1–12:34, June 2024. doi: 10.1145/3655027. URL https://dl.acm.org/doi/10.1145/3655027.

Manuel S. Rudolph, Ntwali Bashige Toussaint, Amara Katabarwa, Sonika Johri, Borja Peropadre, and Alejandro Perdomo-Ortiz. Generation of High-Resolution Handwritten Digits with an Ion-Trap Quantum Computer. *Phys. Rev. X*, 12(3):031010, July 2022. ISSN 2160-3308. doi: 10.1103/PhysRevX.12.031010. URL https://link.aps.org/doi/10.1103/PhysRevX.12.031010.

M. Schuld and F. Petruccione. *Machine Learning with Quantum Computers*. Quantum Science and Technology. Springer International Publishing, 2021. ISBN 9783030830984. URL https://books.google.de/books?id=-N5IEAAAQBAJ.

Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A*, 99:032331, Mar 2019. doi: 10.1103/PhysRevA.99.032331. URL https://link.aps.org/doi/10.1103/PhysRevA.99.032331.

Kevin Shen, Bernhard Jobst, Elvira Shishenina, and Frank Pollmann. Classification of the Fashion-MNIST dataset on a quantum computer. *arXiv:2403.02405*, March 2024. doi: 10.48550/arXiv.2403.02405.

Runqiu Shu, Xusheng Xu, Man-Hong Yung, and Wei Cui. Variational Quantum Circuits Enhanced Generative Adversarial Network, February 2024. URL http://arxiv.org/abs/2402.01791. arXiv:2402.01791.

Daniel Silver, Aditya Ranjan, Tirthak Patel, Harshitta Gandhi, William Cutler, and Devesh Tiwari. Mosaiq: Quantum generative adversarial networks for image generation on nisq computers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7007–7016, 2023. doi: 10.1109/ICCV51070.2023.00647.

Eero P Simoncelli and Bruno A Olshausen. Natural Image Statistics and Neural Representation. *Annual Review of Neuroscience*, 24(1):1193–1216, March 2001. ISSN 0147-006X, 1545-4126. doi: 10.1146/annurev.neuro.24.1.1193.

Ashish Solanki, Sandeep Singh Kang, Sanjay Singla, and T.S. Gururaja. High-Resolution Fashion Image Generation using Quantum-GAN. In *2024 First International Conference on Technological Innovations and Advance Computing (TIACOMP)*, pp. 118–123, June 2024. doi: 10.1109/TIACOMP64125.2024.00029. URL https://ieeexplore.ieee.org/abstract/document/10742822.

Samuel A. Stein, Betis Baheri, Daniel Chen, Ying Mao, Qiang Guan, Ang Li, Bo Fang, and Shuai Xu. QuGAN: A Quantum State Fidelity based Generative Adversarial Network. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 71–81, October 2021. doi: 10.1109/QCE52317.2021.00023. URL https://ieeexplore.ieee.org/abstract/document/9605352.

Bo Sun, Phuc Q. Le, Abdullah M. Iliyasu, Fei Yan, J. Adrian Garcia, Fangyan Dong, and Kaoru Hirota. A multi-channel representation for images on quantum computers using the RGBα color space. In *2011 IEEE 7th International Symposium on Intelligent Signal Processing*, pp. 1–6, October 2011. doi: 10.1109/WISP.2011.6051718.

Bo Sun, Abdullah M. Iliyasu, Fei Yan, Fangyan Dong, and Kaoru Hirota. An RGB multi-channel representation for images on quantum computers. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 17(3):404–417, March 2013. doi: 10.20965/jaciii.2013.p0404.

Aaron Mark Thomas and Sharu Theresa Jose. VAE-QWGAN: Improving Quantum GANs for High Resolution Image Generation, September 2024. URL http://arxiv.org/abs/2409.10339. arXiv:2409.10339.

Shu Lok Tsang, Maxwell T. West, Sarah M. Erfani, and Muhammad Usman. Hybrid Quantum–Classical Generative Adversarial Network for High-Resolution Image Generation. *IEEE Transactions on Quantum Engineering*, 4:1–19, 2023. ISSN 2689-1808. doi: 10.1109/TQE.2023.3319319. URL https://ieeexplore.ieee.org/abstract/document/10264175. Conference Name: IEEE Transactions on Quantum Engineering.

A. van der Schaaf and J.H. van Hateren. Modelling the power spectra of natural images: Statistics and information. *Vision Research*, 36(17):2759–2770, 1996. ISSN 0042-6989. doi: https://doi.org/10.1016/0042-6989(96)00002-8. URL https://www.sciencedirect.com/science/article/pii/0042698996000028.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Benjamin Villalonga, Dmitry Lyakh, Sergio Boixo, Hartmut Neven, Travis S Humble, Rupak Biswas, Eleanor G Rieffel, Alan Ho, and Salvatore Mandrà. Establishing the quantum supremacy frontier with a 281 pflop/s simulation. *Quantum Science and Technology*, 5(3):034003, apr 2020. doi: 10.1088/2058-9565/ab7eeb. URL https://dx.doi.org/10.1088/2058-9565/ab7eeb.

G.K. Wallace. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992. doi: 10.1109/30.125072.

Ruyu Wang, Sabrina Hoppe, Eduardo Monari, and Marco F. Huber. Defect transfer gan: Diverse defect synthesis for data augmentation, 2023. URL https://arxiv.org/abs/2302.08366.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, August 2017. doi: 10.48550/arXiv.1708.07747. Dataset available at https://github.com/zalandoresearch/fashion-mnist.

Deborah Yao. One year on, github copilot adoption soars, June 2023. URL https://aibusiness.com/companies/one-year-on-github-copilot-adoption-soars. Accessed: 2025-09-02.

Qing-Wei Zeng, Hong-Ying Ge, Chen Gong, and Nan-Run Zhou. Conditional quantum circuit Born machine based on a hybrid quantum–classical framework. *Physica A: Statistical Mechanics and its Applications*, 618:128693, May 2023. ISSN 0378-4371. doi: 10.1016/j.physa.2023.128693. URL https://www.sciencedirect.com/science/article/pii/S0378437123002480.

A NOTATION AND DEFINITIONS

The present work follows the standard notions and definitions commonly found in the quantum computing literature (Nielsen & Chuang, 2011) and is briefly presented here.

We adopt the Dirac (*bra–ket*) notation, where a quantum state labeled by ψ is written as a 'ket' $|\psi\rangle$. For a single qubit, $|\psi\rangle$ may be state zero $|0\rangle$, one $|1\rangle$, or, unlike a classical bit, in a *superposition*

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \quad \text{with} \quad \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1. \tag{4}$$

The coefficients α, β are called *probability amplitudes* for reasons that become clear shortly. The state-vector representation expresses the 'ket' states as column vectors when fixing a basis. The common *computational basis*, used in this work, is composed of the zero and one states as

$$|0\rangle = \boldsymbol{e}^{(0)} = (1 \quad 0)^\top, \qquad |1\rangle = \boldsymbol{e}^{(1)} = (0 \quad 1)^\top, \tag{5}$$

which span the *state* space is $\mathbb{C}^2$ and superpositions are simply basis decompositions. Equipped with the canonical inner product $\langle\phi|\psi\rangle$, this space is a Hilbert space. This definition uses a 'bra', which is the adjoint of the ket $|\psi\rangle$ (conjugate row vector of the state-vector), i.e., $\langle\psi| = |\psi\rangle^\dagger = (|\psi\rangle^\top)^*$.

The *tensor product* $\otimes$ combines single-qubit spaces into the joint state space $\mathbb{C}^{2^n}$ of an n-qubit system. For example, two qubits $|\psi_1\rangle$ and $|\psi_2\rangle$ form the composite state $|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle$, The computational basis naturally generalizes to 2^n states, given by all tensor products of n qubits in $|0\rangle$ and $|1\rangle$, commonly labeled by a bit string or integer label, e.g., $|101\rangle \equiv |5\rangle$. Hence, the n-qubit Hilbert space is spanned by $\{|0\rangle, \ldots, |2^n - 1\rangle\} = \{\boldsymbol{e}^{(0)}, \ldots, \boldsymbol{e}^{(2^n-1)}\}$.

Entanglement distinguishes two types of multi-qubit states. A state $|\psi\rangle \in \mathbb{C}^{2^n}$ is *separable* (unentangled) if a tensor product decomposition into single-qubit states exists $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle$, and *entangled* otherwise. Hence, entangled states cannot be fully described by their subsystems, only by the joint system. In the FRQI representation used here, entanglement corresponds to spatially correlated pixel colors, whereas unentangled states yield pixel colors independent of position.

Quantum states evolve not only linearly $|\psi\rangle \mapsto U\,|\psi\rangle$, but also, which conserves normalization, by a unitary transformation, i..e, $U^\dagger U = UU^\dagger = I$. In the state-vector expression, this action corresponds to a matrix-vector product in a fixed basis. A standard way to express such transformations is through quantum circuits, where unitary operations are decomposed into elementary quantum gates (e.g., see Fig. 1). Two gates combine either sequentially, $U_1 \circ U_2$, corresponding in matrix form to $U_2 U_1$, or in parallel on disjoint subsystems via the tensor/Kronecker product $U_1 \otimes U_2$. The basic single-qubit gates used here are defined in the computational basis as

$$H = \tfrac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{6}$$

Parameterized rotation gates are generated by the Pauli operators X, Y, Z through exponentials

$$R_x(\theta) = e^{-i\theta X/2}, \quad R_y(\theta) = e^{-i\theta Y/2}, \quad R_z(\theta) = e^{-i\theta Z/2}, \tag{7}$$

rotating a qubit about its x-, y-, and z-axis by an angle θ, respectively. Controlled (two-qubit) gates act conditionally, with the control qubit determining whether the operation is applied to the target qubit. Examples include the controlled-NOT (CNOT) and controlled-R_y (in block-matrix notation):

$$\mathrm{CNOT} = \begin{pmatrix} \boldsymbol{I}_2 & 0 \\ 0 & X \end{pmatrix}, \quad cR_y(\theta) = \begin{pmatrix} \boldsymbol{I}_2 & 0 \\ 0 & R_y(\theta) \end{pmatrix}. \tag{8}$$

Note that only multi-qubit gates can alter the entanglement of a state.

Finally, the probabilistic nature of quantum mechanics arises from the fact that quantum states cannot be fully observed: measurements yield probabilistic outcomes and collapse the state to align with the observation. For computational basis measurements, the probability of observing the qubits representing integer $i \in \{0, \ldots, 2^n - 1\}$ is

$$p_i = |\,\langle i|\psi\rangle\,|^2. \tag{9}$$

This probability is the squared magnitude of the corresponding probability amplitude in the superposition of computational basis states (or, put differently, the inner product of the $|\psi\rangle$ and $|i\rangle$). Consequently, the closer $|\psi\rangle$ is to a basis state $|i\rangle$, the higher the likelihood of observing i upon measurement. In a quantum computer, states can typically be prepared repetitively. Therefore, from a number of measurement shots, certain state quantities, such as the (computational basis) probabilities, can be estimated, which are of particular interest to decode the image from an FRQI state.

B METHODOLOGICAL AND IMPLEMENTATION DETAILS

All experiments in this work are implemented as numerical state-vector simulations. For the gradient-based optimization, we use PennyLane (Bergholm et al., 2022) in combination with the just-in-time compilation and vectorization capabilities of JAX (Bradbury et al., 2025) to perform auto-differentiable, GPU-accelerated state-vector calculations.

B.1 GENERATIVE MODELING

The Generative Adversarial Networks (GAN) (Goodfellow et al., 2014) technique was originally proposed for classical neural networks. One neural network functions as the *generator* $G_{\boldsymbol{\theta}}(\boldsymbol{z})$ and learns (parameters $\boldsymbol{\theta}$) to produce samples, based on random noise inputs $\boldsymbol{z}$, that are indistinguishable from the real data. In contrast, another neural network operates as the *discriminator* $D_{\boldsymbol{\phi}}(\boldsymbol{x})$ and concurrently learns (parameters $\boldsymbol{\phi}$) to provide a discrimination signal indicating whether the input is real or generated (fake). GANs can be readily extended to quantum generative models by replacing the generator neural network with a generator quantum circuit, where the generated data sample is constructed from measurement expectation values for continuous-valued outputs (Riofrío et al., 2024), such as images (Tsang et al., 2023). In principle, although not studied in this work, the discriminator could also be a quantum model.

GANs were originally introduced with a discriminator resembling a binary discrimination signal (for classification $D_{\boldsymbol{\phi}}(\boldsymbol{x}) = 1$ for real and $D_{\boldsymbol{\phi}}(\boldsymbol{x}) = 0$ for fake inputs $\boldsymbol{x}$). Due to training instability and problems such as the mode collapse phenomenon (resulting in less diverse samples than the real distribution), the original GAN framework can be improved by the Wasserstein-GAN (WGAN) approach (Arjovsky et al., 2017), where the discriminator now provides a continuous discrimination signal $D_{\boldsymbol{\phi}}(\boldsymbol{x}) \in \mathbb{R}$ that should be maximized for real and minimized for fake inputs $\boldsymbol{x}$. This is described by the following optimization problem, which directly gives rise to the corresponding loss functions that are minimized alternately during training:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\phi}} \underbrace{\mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}_{\boldsymbol{x}}} D_{\boldsymbol{\phi}}(\boldsymbol{x}) \overbrace{-\mathbb{E}_{\boldsymbol{z}\sim\mathbb{P}_{\boldsymbol{z}}} D_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z}))}^{=\mathcal{L}_G(\boldsymbol{\theta})}}_{=-\mathcal{L}_D(\boldsymbol{\phi})} \tag{10}$$

The noise distribution $\mathbb{P}_{\boldsymbol{z}}$ induces the generation distribution $\mathbb{P}_{G_{\boldsymbol{\theta}}}$ through the map from noise to data space that the generator $G_{\boldsymbol{\theta}}(\cdot)$ provides. We utilize batches of size N of generated (and real) data to evaluate the *empirical* loss functions $L_G(\boldsymbol{\theta})$ and $L_D(\boldsymbol{\theta})$, which estimate the expectations over the noise and real data distributions $\mathbb{P}_{\boldsymbol{z}}, \mathbb{P}_{\boldsymbol{x}}$ in $\mathcal{L}_G(\boldsymbol{\theta})$ and $\mathcal{L}_D(\boldsymbol{\theta})$, respectively, by substituting $\mathbb{E}(\cdot) \approx \frac{1}{N}\sum(\cdot)$.

The discriminator is required to be 1-Lipschitz so that its output differences reflect actual distances in input space, preventing it from creating artificial in the loss landscape that would distort the Wasserstein distance. To enforce this condition, it is common practice to add a gradient penalty of the discriminator with respect to its inputs, scaled by a regularization coefficient $\lambda > 0$

$$\mathcal{L}_D(\boldsymbol{\phi}) \leftarrow \mathcal{L}_D(\boldsymbol{\phi}) + \lambda \mathbb{E}_{\hat{\boldsymbol{x}}\sim\mathbb{P}_{\hat{\boldsymbol{x}}}}\Big[\left(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1\right)^2\Big], \tag{11}$$

where these inputs $\hat{\boldsymbol{x}}$ are uniformly distributed $\mathbb{P}_{\hat{\boldsymbol{x}}}$ on lines between pairs of samples from the data distribution $\mathbb{P}_{\boldsymbol{x}}$ and generator distribution $\mathbb{P}_{G_{\boldsymbol{\theta}}}$ (Gulrajani et al., 2017). Again, finite batches of N inputs provide expectation estimates and yield the gradient-penalty version of the empirical loss $L_D(\boldsymbol{\phi})$. In this work, all implementations refer to the Wasserstein GAN method with gradient penalty (WGAN-GP), utilizing a quantum generator, whether it is termed QGAN or QWGAN.

B.2 GENERATOR DECODING: FROM QUANTUM STATES TO IMAGES

As outlined in the main text, the generator ansatz does not enforce valid FRQI states. Therefore, we construct the image solely from the estimated (computational basis) measurement probabilities of the generated state $|G(\boldsymbol{z};\boldsymbol{\theta})\rangle$, and then normalize/condition these probabilities to recover a valid FRQI representation.

Concretely, for a pixel indexed by j, probabilities of observing the color qubit of pixel j in states $|0\rangle$ and $|1\rangle$ are

$$p_{0,j} = \left|(\langle 0| \otimes \langle j|)|G(\boldsymbol{z};\boldsymbol{\theta})\rangle\right|^2, \qquad p_{1,j} = \left|(\langle 1| \otimes \langle j|)|G(\boldsymbol{z};\boldsymbol{\theta})\rangle\right|^2, \tag{12}$$

respectively, following the computational basis measurement definition in Eq. (9). The total probability of measuring information of pixel j is

$$p_j = p_{0,j} + p_{1,j}. \tag{13}$$

For a valid FRQI state, the total probability always equals $1/2^N$ because all 2^N pixels are equally likely to be observed.

Hence, to achieve conformity to the FRQI representation in the decoding process, normalization uncovers the effective color-qubit amplitudes as defined in Eq. (2)

$$a_{0,j} = \sqrt{p_{0,j}/p_j}, \qquad a_{1,j} = \sqrt{p_{1,j}/p_j}. \tag{14}$$

Finally, the pixel value is derived from the FRQI encoding using trigonometric inverse functions as

$$x_j = \tfrac{2}{\pi}\arccos\big(a_{0,j}\big) = \tfrac{2}{\pi}\arcsin\big(a_{1,j}\big). \tag{15}$$

B.3 Discriminator design

The discriminator is implemented as a convolutional neural network (CNN) (Lecun et al., 1998; Fukushima, 1980) designed to distinguish between real and fake images, i.e., those obtained by decoding the quantum states generated by the QGAN. The exact CNN architecture is adopted from the discriminator suggested by Gulrajani et al. (2017) for the MNIST dataset and outlined in the following. Three convolutional layers are used and followed by leaky ReLU activations, which preserve gradient flow in low-activation regions. All convolutions have 5×5 kernels and are applied with a stride of 2, which halves the size in each layer (no pooling is used). The number of convolutional filters is 64, 128, and 256 in the first, second, and third layers, respectively. After the convolutional layers, the outputs are flattened and passed into a fully connected layer that maps the extracted features to a single scalar output without any further activation function.

B.4 Quantum generative modeling of color images

To extend the QGAN framework in this work to generating color images, we first present the extension of the FRQI grayscale encoding to color images proposed by Sun et al. (2011; 2013). Then, we introduce a natural extension of the task-specific, FRQI-based generator ansatz to this more general image encoding. We refer to this new ansatz as the *color-extended task-specific ansatz*.

Quantum image representations for color images. We encode color images with the *multi-channel representation of quantum images (MCRQI)* (Sun et al., 2011; 2013). For each pixel, the data value now has several components, $\boldsymbol{x}_j = (\ x_j^R,\ \ x_j^G,\ \ x_j^B,\ \ x_j^\alpha\ )^\top$, corresponding to the three RGB color channels and a possible fourth α channel indicating the opacity of the image. If only the three RGB channels are available for a given image (as is the case for all color image datasets considered in this work), the image is at full opacity and we can simply set the α channel to zero (Sun et al., 2013) or ignore it in the decoding. The color information of a pixel is then encoded in a three-qubit state as

$$\begin{aligned} |c(\boldsymbol{x}_j)\rangle = \frac{1s}{2}\Big(&\cos(\tfrac{\pi}{2}x_j^R)\,|000\rangle + \sin(\tfrac{\pi}{2}x_j^R)\,|100\rangle \\ &+ \cos(\tfrac{\pi}{2}x_j^G)\,|001\rangle + \sin(\tfrac{\pi}{2}x_j^G)\,|101\rangle \\ &+ \cos(\tfrac{\pi}{2}x_j^B)\,|010\rangle + \sin(\tfrac{\pi}{2}x_j^B)\,|110\rangle \\ &+ \cos(\tfrac{\pi}{2}x_j^\alpha)\,|011\rangle + \sin(\tfrac{\pi}{2}x_j^\alpha)\,|111\rangle\Big), \end{aligned} \tag{16}$$

with normalized values $x_j^R, x_j^G, x_j^B, x_j^\alpha \in [0,1]$. Thus, by inserting this definition in Eq. (1), a color image with 2^N pixels is encoded into a quantum state with $n = N + 3$ qubits. Just as for grayscale images, encoding natural color images via MCRQI results in lowly-entangled states, which are well

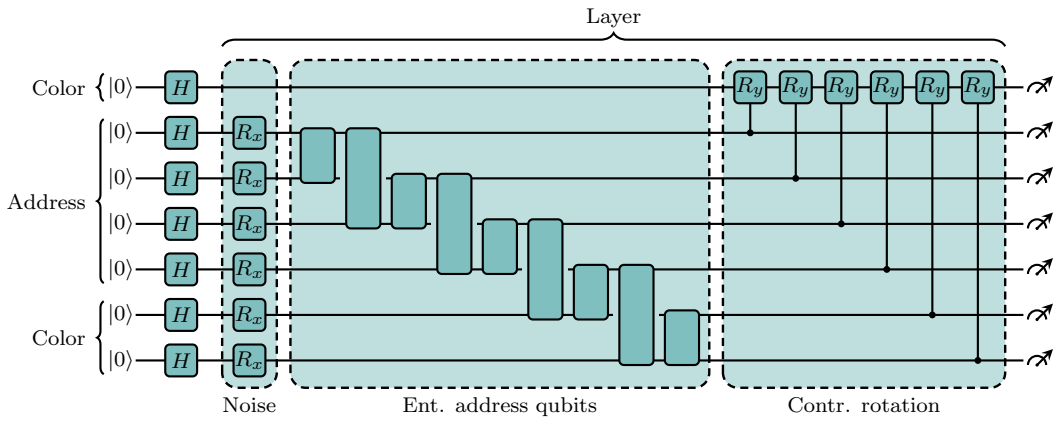


Figure 8: Quantum generator for a 4×4-pixel color image with one layer of noise, entangling and controlled R_y gates. The last two color qubits are interpreted as (channel-) address qubits, analogous to four sub-pixels per pixel, and are integrated into the address qubit register accordingly.

approximated by tensor-network states. To prepare the state *exactly* on a quantum computer, we can essentially reuse the same circuit that prepares an FRQI state and run it for each color channel separately. However, this procedure treats the color channels independently, which is not ideal for generative modeling, as channels should be considered together.

Color-extended task-specific ansatz. In MCRQI, the three color qubits, as defined in Eq. (16), play distinct roles: the first (left) encodes the channel intensity in its z-polarization, while the last two (center and right) specify the channel, i.e., $|00\rangle$ for R, $|01\rangle$ for G, $|10\rangle$ for B, and $|11\rangle$ for α. Interpreting these two channel qubits also as address qubits casts MCRQI into an FRQI perspective, effectively mapping a color image onto a grayscale image of doubled resolution. Within the Morton order, when these channel addressing color qubits are placed as the last two address qubits, this can be interpreted as subdividing each pixel into four sub-pixels. This interpretation aligns with the design of digital displays, where each pixel is divided into RGB sub-pixels that, when sufficiently miniaturized, appear as a single colored pixel to the human eye. We adopt this physical intuition as the basis for our color-extended ansatz to achieve a task-specific design with sufficient inductive bias.

Consequently, extending our grayscale generator ansatz to color images becomes straightforward: the last two color qubits are treated as highest-resolution address qubits. Figure 8 provides a circuit diagram for a 4×4-pixel color image analogous to the 4×4-pixel grayscale example in Fig. 1. As with any address qubit, these two color qubits are affected by noise (the noise vector now includes two more components), are included in the N2 and N3 entangling ladders, and act as control qubits each for two additional R_y gates on the (first) color qubit.

B.5 Training with shot noise

We recall, that the exact probability distribution P is defined as the squared amplitudes of the quantum state produced by the circuit. However, in practice, we only have access to samples from this distribution. Given the unfavorable scaling of the parameter-shift rule (Mitarai et al., 2018; Schuld et al., 2019) for large quantum systems, we focus on assessing the influence of shot noise on the generated distribution, but not the exact impact on the gradient. We define the computational basis $\{|x\rangle\}_{x\in\{0,1\}^n}$, i.e., the set of all bitstrings of length n, where n is the number of qubits. The exact distribution P assigns to each basis state $|x\rangle$ the probability $|\langle x|\psi\rangle|^2$, obtained from the squared amplitudes of the circuit's output state $|\psi\rangle$. In practice, however, we only have access to a finite-shot approximation $\hat{P}$, obtained from measurement samples. To emulate the effect of shot noise while keeping gradients tractable, we compute the per–basis-state deviation $\varepsilon(x) = \hat{P}(x) - P(x)$. We then perturb the exact distribution by this deviation, $\tilde{P}(x) = P(x) + \varepsilon(x)$, and apply a subsequent clipping step to ensure nonnegativity, followed by a renormalization. The gradient flows only

through the exact distribution P, not through the stochastic deviation ε. This procedure closely resembles the reparameterization trick from (Kingma & Welling, 2014). Thus, the gradient is evaluated with respect to the exact distribution P, while still enabling efficient backpropagation during the simulation of quantum circuits. Note that the gradient is affected solely by the clipping step. If no measurement outcomes occur in the basis states corresponding both to $|0\rangle$ and $|1\rangle$, we assign the pixel a neutral gray value before reconstructing the image from the quantum state and feeding it into the discriminator.

B.6 Training hyperparameters

The minibatch size is 64 in most experiments, reduced to 32 for the large (64-layer) MNIST and Fashion-MNIST QGANs and to 16 for the color model, solely due to GPU memory limits. General generator parameters are initialized from a zero-centered normal distribution with variances $\sigma^2_{\text{init}} \in \{0.001, 0.01, 0.025, 0.05\}$, using larger variances for smaller models and vice versa. Noise-tuning parameters are further scaled down by a factor of 10. The discriminator is updated ten times per generator update (all iteration counts in the paper refer to generator updates), with the ratio reduced to $5:1$ for the color model. Both the generator and discriminator are optimized with the *Adam* optimizer Kingma & Ba (2014), using learning rates in $\{0.001, 0.0025, 0.01\}$, typically lower for larger models. For the discriminator, the learning rate is reduced by a factor of 10 in grayscale experiments and 4 in the color model. Training the QGANs largely follows the WGAN-GP setup of Gulrajani et al. (2017), which informs the following choices: Adam hyperparameters are fixed to $\beta_1 = 0.5$ and $\beta_2 = 0.9$, and the gradient-penalty coefficient λ is set to 10 as defined in Eq. (11).

C Datasets

The MNIST dataset (Lecun et al., 1998; Deng, 2012) is a simple and widely used dataset for training machine learning models. It contains grayscale images of handwritten digits between '0' and '9', and associated labels indicating the correct digit. The original images have 28×28 pixels. Here, we use bilinear interpolation to resize them to 32×32 pixels making them suitable for processing on a quantum computer. The class distribution over the $70\,000$ images is approximately uniform, with each class representing between 9% and 11% of the dataset.

The Fashion-MNIST dataset (Xiao et al., 2017) was introduced as a more challenging alternative to MNIST, after it became apparent that MNIST was too easily solved and no longer posed a significant challenge for more sophisticated classification models. The dataset also features $70\,000$ grayscale images with an original resolution of 28×28 pixels, which we again resize to 32×32 pixels using bilinear interpolation. Instead of handwritten digits, the images feature the 10 different clothing articles, T-shirt/top, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, ankle boot. The dataset is balanced over the ten classes. When presented in this work, the colors of the generated images are inverted for Fashion-MNIST for a more intuitive presentation, e.g., of shadings.

The Street View House Numbers (SVHN) dataset (Netzer et al., 2011) offers a natural-image analog to MNIST, comprising RGB 32×32 crops of digits 0–9 taken from Google Street-View scenes that feature real-world background variation. In our experiments, we restrict the corpus to those samples whose central digit is 0. Within the official core split this subset contains roughly $4\,948$ training samples and $1\,744$ test samples.

D Extended experiments and analysis

Experiments and analyses beyond the results presented in the main text (Sec. 4) are discussed here.

D.1 Model selection and evaluation

All samples presented in this work are generated by either a QGAN after being trained for a fixed number of iterations, or a QGAN reloaded from a training checkpoint, which is selected automatically via the maximum mean discrepancy (MMD) metric instead of the lowest-loss checkpoint, a common criterion in generative modeling (Borji, 2019; 2021). Generally, the number of iterations is set before training starts, or training is stopped after a preset time limit, independent of the loss or

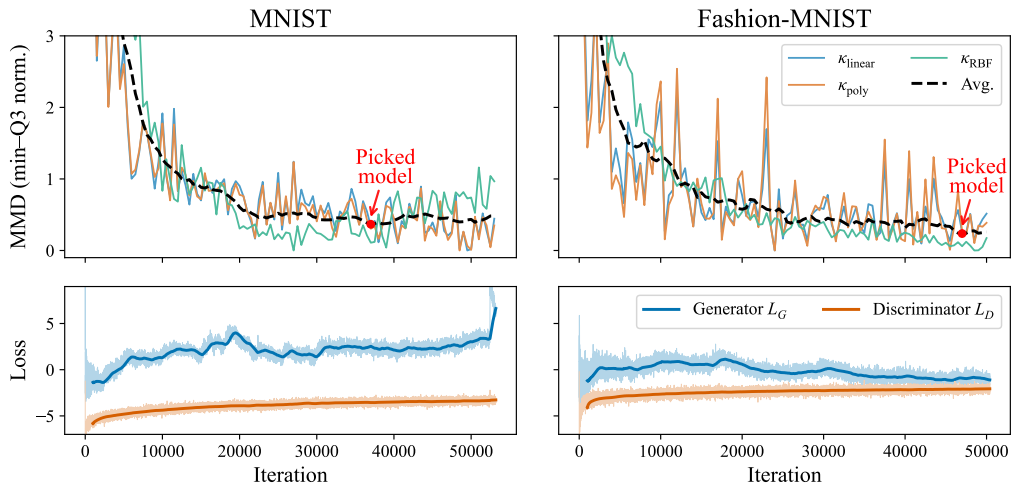


Figure 9: Learning curves of MMD and loss for the largest QGANs (64 layers) on MNIST and Fashion-MNIST. The MMD metric is normalized using its minimum (min) and upper quantile (Q3). The average MMD curve is computed across the three kernels κ_{linear}, κ_{poly}, and κ_{RBF}, followed by a centered moving average over 9 neighboring training checkpoints. The selected model is indicated at the point where the average MMD reaches its minimum. For clarity, the loss curves are additionally smoothed with a moving average over 1 000 iterations.

evaluation metrics. Importantly, in both model selection scenarios, human intervention or selection was not involved to avoid biased or "cherry-picked" results.

Maximum mean discrepancy (MMD). The kernel MMD (Gretton et al., 2012) measures the difference between two probability distributions $\mathbb{P}_{\boldsymbol{x}}$ and $\mathbb{P}_G$, denoting the real data distribution and generator distribution, respectively, in the context of QGAN evaluation. Intuitively, MMD compares similarities within and across datasets, providing a measure of how well the generator mimics the real distribution. For the largest QGAN models in this work, which were used to generate the images in Figs. 3a and 3b, the learning curves of MMD and loss are presented in Fig. 9. The empirical definition of the MMD, based on k samples $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(k)} \sim \mathbb{P}_{\boldsymbol{x}}$ (a random k-sized subset of the training set) and $\hat{\boldsymbol{x}}^{(1)}, \ldots, \hat{\boldsymbol{x}}^{(k)} \sim \mathbb{P}_G$, reads as follows

$$\text{MMD}_\kappa = \frac{1}{k^2}\sum_{i,j}\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) - 2\frac{1}{k^2}\sum_{i,j}\kappa(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j) + \frac{1}{k^2}\sum_{i,j}\kappa(\hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j), \tag{17}$$

where κ denotes the kernel. The kernel κ is a symmetric similarity function assigning high values to similar samples and low values to dissimilar ones. We evaluate MMD using three common kernels: *linear*, *polynomial* (of degree 2), and *radial basis function (RBF)* (with unit bandwidth) kernels. To obtain stable scores, the MMD values are normalized between their minimum and upper quantile for each kernel, avoiding sensitivity to noisy estimates from early underfit stages. The final score to pick the best model is computed by averaging across kernels and applying a centered moving average (window size 9) across neighboring training checkpoints. A checkpoint was created every 500 iterations, and $k = 5000$ samples were used to estimate the MMD.

D.2 Impact of generator depth and dataset complexity on image quality

An extended analysis is presented here to investigate further the relationship between model depth, dataset complexity (in terms of the number of classes), and image quality. The experiments are based on MNIST, comparing models trained on either the complete set of classes or a restricted subset (digits *0* and *1*), while varying generator depths $L \in 8, 16, 32$. The number of modes is set to either 2 or 10, matching the number of classes. Each model is trained for 40 000 iterations, and the checkpoint minimizing the MMD metric is used for image generation. Figure 10 presents the results

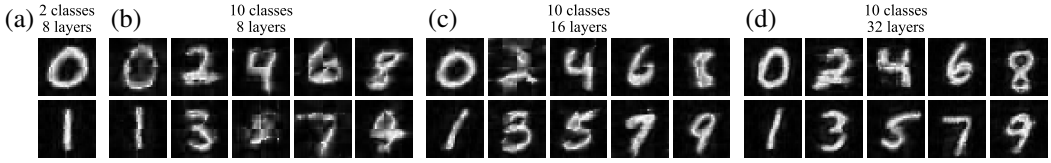


Figure 10: Effect of generator depth and dataset complexity on image quality. Models were trained either on (a) an MNIST subset (digits *0* and *1*) with depth $L = 8$, or on all ten MNIST classes with (b) the same depth $L = 8$ or increased depths of (c) $L = 16$ and (d) $L = 32$. One representative image (manually selected) per class is depicted. Results suggest that increasing the number of classes requires deeper generators to maintain visual quality.

(a) *dress – coat* (b) *t-shirt – dress* (c) *trouser – dress*

Figure 11: Three inter-class modes blend between Fashion-MNIST classes: (a) *dress–coat*, (b) *t-shirt–dress*, and (c) *trouser–dress*. Results are from training a QGAN with 4 times more input noise modes than dataset classes ("overmoding"). As in Fig. 6, each mode is visualized with images representing the mean (center) and $\pm 3\sigma$ variations (outer) along the first principal component. PCA is based on 1 000 samples per mode. Note that (a) *dress–coat* mode was already presented in Fig. 6c.

and clearly shows that as the number of classes increases, deeper generator circuits are required to maintain image quality. Recall that the large models in Sec. 4.1 used 64 layers to capture all MNIST digits at both high quality and diversity.

For the training restricted to two classes, even a shallow generator with $L = 8$ produces high-quality samples (Fig. 10a). This setting exactly matches the number of layers per patch generator used in prior work by Tsang et al. (2023), where 28 such generators were combined for two classes. In contrast to their results, the model here produces images of improved quality, comparable to the results of the much larger 64-layer generators (cf. Fig. 3a). This demonstrates that the gain in quality over previous works is primarily due to our task-specific QGAN design, not merely due to increasing model size. In comparison, Fig. 10b shows images generated from a model with the same number of layers $L = 8$ but now trained on all 10 classes. Here, a decrease in quality is evident, especially when comparing to the *0* and *1* samples in Fig. 10a. Generally, most other classes are captured considerably worse than by the large model as in Fig. 3a. By scaling the model to $L = 16$ (Fig. 10c) and $L = 32$ (Fig. 10d) layers, a successive increase in image quality can be observed. While some images at $L = 16$ (Fig. 10c) already reach a high quality, such as digits *0* and *1* again matching the high quality of the smaller $L = 8$ model when trained on these two classes only (Fig. 10a), it requires $L = 32$ layers (Fig. 10d) to achieve uniformly such quality across all classes.

D.3 Inter-class modes in overmoding

In extension of the analysis on QGAN "overmoding" for the complete Fashion-MNIST dataset, Fig. 11 presents additional modes, analogous to those in Fig. 6. Here, the focus is on *inter-class* modes, which capture images blending between two classes, occurring in the model trained with 40 input noise modes on the 10 classes in Fashion-MNIST. While such blending may initially appear to induce undesired mixing artifacts, it can in fact reflect realistic scenarios. For instance, one mode morphs between *dress* and *coat*, not only adjusting the shape but also introducing a clear line for a zipper (Fig. 11a). Another mode mostly captures *t-shirts* gradually transitioning from a fitted t-shirt into a t-shirt *dress* (Fig. 11b). A third mode gradually brings the legs of a *trouser* closer together until they eventually connect and resemble a *dress*, while the top simultaneously forms proper shoulder caps (Fig. 11c).

LLM USAGE STATEMENT

In accordance with the ICLR 2026 policy on Large Language Model (LLM) usage, we disclose that Grammarly and ChatGPT were utilized for grammar checking, style improvement, and minor text polishing. GitHub Copilot was used to suggest code snippets during development, with all generated code reviewed, tested, and adapted by the authors. No LLMs were used for generating novel research ideas, data analysis, or drafting substantial portions of the manuscript.