Supplemental Material for "Pixel-Translation-Equivariant Quantum Convolutional Neural Networks via Fourier Multiplexers"

Dmitry Chirkov and Igor Lobanov

School of Physics and Engineering, ITMO University, St. Petersburg, Russia.

This document includes a literature table for QCNN-style MNIST results (Sec. 1), additional experiments and analyses (Sec. 2), an explicit specification of the classical baseline architectures used in benchmark comparisons (Sec. 2.3), the full-MNIST classical control used to motivate the translated benchmark choice (Sec. 2.4), an extended classical convolution primer (Sec. 3), complete proofs of the QCS–PCS mismatch lemma and the Fourier-multiplexer theorem (Sec. 4), a longer discussion of MERA-QCNN-style (QCS-equivariant) templates (Sec. 5), derivations behind the Fourier-junction reduction between pooled PCS layers (Sec. 6), discussion of the idealized benchmark model and its resource scaling (Sec. 7), and the gradient-accounting derivation (Sec. 8).

1 QCNN-style models on MNIST: literature table

MNIST [1] is widely used in demonstrations of quantum and hybrid quantum–classical image classifiers, including QCNN-inspired architectures. However, the literature is highly heterogeneous in at least four respects: (i) binary vs multi-class tasks (often with different label subsets), (ii) image resolution and preprocessing (downscaling, PCA-based compression, handcrafted features), (iii) the training/evaluation protocol (data splits, subset selection, number of epochs), and (iv) the readout model (infinite-shot/statevector versus finite-shot sampling, and sometimes implicit assumptions about state preparation). As a result, reported accuracies are rarely directly comparable across papers. This is a general issue in quantum-machine-learning benchmarking, where design choices and classical controls can affect conclusions [2].

In the main text (Sec. **??**), MNIST is used in two fixed benchmark families: a translated-MNIST regime where the classical convolutional and dense baselines are

Work	Task	Input / preprocessing	Reported accuracy (%)
Mattern et al. [3]	10-class	14×14	85–88
Huang et al. [4]	10-class	14×14	96
Li et al. [5]	Binary	4×4 (label pairs)	99.80 (4 vs 5), 91.51 (3 vs 8)
Oh et al. [6]	10-class	10×10	95
Huggins et al. [7]	Binary	8×8	95
Jing et al. [8]	10-class	10×10 and 20×20	94–95
Easom-McCaldin et al. [9]	Binary	9×9	100
Zheng et al. [10]	Binary	8×8	96.65
Hur et al. [11]	Binary	PCA to 16×16	98.1
Huang et al. [12]	Binary / 3-class	8×8	95.8 / 72.1
Li et al. [13]	10-class	32×32	98.97
Wei et al. [14]	10-class	32×32	96.3 and 74.3
This work	10-class	full-MNIST, direct 16×16 / direct 32×32 preprocessing	97.96 / 98.13 (both exact-probability inference)

Notes: Accuracy values are reproduced from the cited papers, higher is better.
Table S1 Overview of representative QCNN-like MNIST results in the literature. Entries are shown in the form reported by the corresponding references. Our main paper uses MNIST with a fixed preprocessing and evaluation protocol and includes within-family matched controls (Sec. **??**).

well separated, and a full-MNIST size sweep used for the main infinite-shot and finite-shot PCS-QCNN analyses (Secs. **??**–**??**).

Table S1 summarizes representative QCNN-like results on MNIST, together with the corresponding task type and input resolution as reported by the authors.

2 Additional experiments and analyses

This section gives additional analyses for the main-text Results section (Sec. **??**): an encoder-scale brightness sensitivity sweep, an extended hyperparameter and parameter-accounting report, a readout entropy diagnostic, a local readout-space loss-landscape probe, and qualitative error analysis. These analyses document the sensitivity of the FRQI angle map, training behavior, finite-shot sensitivity, and residual error structure.

2.1 Hyperparameter sweep and parameter accounting

The experiments reported here use two sweep families. Figure **??**(a) is a translated-MNIST architecture sweep with digits resized to 16×16, placed on a 32×32 canvas, translated by at most 8 pixels along each axis, evaluated with full readout, and trained for 2000 epochs over 3 seeds, with $Q \in \{1, 2, 3, 4, 5\}$ and $n_f \in \{1, 2, 3\}$. Figure **??**(b) is a full-MNIST size sweep with full readout, $Q = 1$, $n_f = 3$, 2000 training epochs, and 3 seeds, across four preprocessing settings: direct 8×8, direct 16×16, centered

Configuration	Total qubits	Quantum params	Classifier params	Readout shape	Final mean acc. (%)
$Q = 1, n_f = 1$	11	4 096	20 490	$32 \times 32 \times 2$	51.29
$Q = 1, n_f = 2$	12	16 384	40 970	$32 \times 32 \times 4$	62.50
$Q = 1, n_f = 3$	13	65 536	81 930	$32 \times 32 \times 8$	66.36
$Q = 2, n_f = 1$	11	8 192	5 130	$16 \times 16 \times 2$	59.02
$Q = 2, n_f = 2$	12	32 768	10 250	$16 \times 16 \times 4$	76.00
$Q = 2, n_f = 3$	13	131 072	20 490	$16 \times 16 \times 8$	76.85
$Q = 3, n_f = 1$	11	9 216	1 290	$8 \times 8 \times 2$	65.02
$Q = 3, n_f = 2$	12	36 864	2 570	$8 \times 8 \times 4$	75.89
$Q = 3, n_f = 3$	13	147 456	5 130	$8 \times 8 \times 8$	79.15
$Q = 4, n_f = 1$	11	9 472	330	$4 \times 4 \times 2$	62.63
$Q = 4, n_f = 2$	12	37 888	650	$4 \times 4 \times 4$	74.44
$Q = 4, n_f = 3$	13	151 552	1 290	$4 \times 4 \times 8$	73.79
$Q = 5, n_f = 1$	11	9 536	90	$2 \times 2 \times 2$	60.46
$Q = 5, n_f = 2$	12	38 144	170	$2 \times 2 \times 4$	70.25
$Q = 5, n_f = 3$	13	152 576	330	$2 \times 2 \times 8$	67.96

Notes: This table corresponds to the translated-MNIST architecture sweep in Fig. **??**(a). All runs use a 32×32 canvas with digits resized to 16×16 and translated by at most 8 pixels per axis. The accuracy column reports the final mean test accuracy over 3 seeds from the runs used for the figure; small differences among high-performing rows should not be interpreted as a statistically resolved ranking.
Table S2 Exact parameter accounting for the PCS-QCNN architecture sweep in Fig. **??**(a).

28×28 in a 32×32 canvas, and direct 32×32. The follow-up reference model reused in Figs. **??**, **??**, and Supplemental Figs. S4, S5, and S6 is the medium-resolution direct 16×16 member of this full-MNIST size sweep.

For square canvas size $2^{n_{\text{idx}}}$ and full readout, the total logical qubit count is $n_{\text{tot}} = 2n_{\text{idx}} + n_f$ (Eq. (**??**)). After $Q - 1$ pooling steps per axis, the remaining active index width is $n_l = n_{\text{idx}} - Q + 1$, so the classifier sees $n_{\text{meas}} = 2n_l + n_f$ effective measured qubits (Eq. (**??**)) and input dimension $D_{\text{out}} = 2^{n_{\text{meas}}}$ (Eq. (**??**)). Because the classifier is a single biased linear layer, its parameter count is exactly $10D_{\text{out}} + 10$. The quantum parameter counts follow directly from the model definition in the main text. Tables S2 and S3 summarize these exact counts and final mean accuracies for the reported sweeps; the full accuracy trajectories are shown in Figs. **??**(a,b).

The translated-MNIST architecture sweep shows that intermediate depths and wider feature registers can improve performance, with the largest displayed mean at $Q = 3, n_f = 3$; with only three seeds and no uncertainty column in the table, this should be read as a descriptive sweep result rather than a resolved architecture ranking. The full-MNIST size sweep likewise depends on image resolution: direct 8×8 gives the lowest final mean (93.21%), the larger three preprocessing choices span 97.96–98.13%, and the direct 32×32 model attains the highest final mean in the displayed sweep (98.13%).

2.1.1 Preprocessing convention for encoder inputs.

For reproducibility, let $x_{u,v} \in [0, 1]$ denote the preprocessed grayscale intensity at pixel (u, v) after resizing, optional canvas placement, and normalization. The encoder maps it to an angle

$$p_{u,v} = a + (b - a)x_{u,v}.$$

Preprocessing	Total qubits	Quantum params	Classifier params	Readout shape	Final mean a
direct 8×8	9	4 096	5 130	$8 \times 8 \times 8$	93.21
direct 16×16	11	16 384	20 490	$16 \times 16 \times 8$	97.96
28×28 on 32×32	13	65 536	81 930	$32 \times 32 \times 8$	97.96
direct 32×32	13	65 536	81 930	$32 \times 32 \times 8$	98.13

Notes: This table corresponds to the full-MNIST size sweep in Fig. **??**(b), which fixes $Q = 1$ and $n_f = 3$ and varies only the preprocessi The 28×28-on-32×32 and direct-32×32 rows share the same quantum and classifier dimensions because both use a 32×32 quantu they differ only in image preprocessing before encoding, with no random translations in either case. The accuracy column reports the f test accuracy over 3 seeds from the runs used for the figure.

Table S3 Exact parameter accounting for the PCS-QCNN size sweep in Fig. **??**(b).

For the PCS-QCNN experiments reported in the paper, $(a, b) = (0, \pi)$. Thus $x_{u,v}$ is the direct input to the brightness map in Eq. (**??**). For $n_f > 1$, the auxiliary feature qubits are initialized in $|0\rangle^{\otimes(n_f-1)}$, so the local feature-register state is $|0\rangle^{\otimes(n_f-1)} \otimes |\phi_{u,v}\rangle$ with the grayscale/color state on the least-significant feature qubit. In the numerical protocol used here, the global $1/\sqrt{N_x N_y}$ prefactor from Eq. (**??**) is omitted during state initialization; measurement compensates for this by dividing the final marginal by the corresponding overall spatial normalization factor.

2.1.2 Block parameterization in the benchmark model.

In the benchmark model, each mode-wise feature-register block $U^{(\ell)}_{k_x,k_y}(m)$ is parameterized as a general $U(2^{n_f})$ unitary via the full Pauli-basis exponential (Eq. (**??**) in the main text, Sec. **??**). Concretely, this includes the all-identity Pauli string and yields 4^{n_f} real parameters per Fourier mode (and, for $\ell \geq 2$, per selected condition branch), and the full multiplexer $\mathcal{B}^{(\ell)}$ is applied as an explicit block-diagonal unitary in the Fourier basis. The identity coefficient is retained because block-local phases become relative phases between Fourier modes or pooling branches in the full multiplexer. For simulation, we treat this multiplexer as a single explicit block-diagonal unitary map. Hardware compilation cost is discussed in Sec. 7.1.

2.1.3 Numerical realization.

The reported training and evaluation use dense-tensor statevector simulation rather than a gate-by-gate hardware emulation. The PCS-QCNN evolution, including Fourier transforms, multiplexers, marginal measurement, and finite-shot sampling, is evaluated directly on the full state tensor. This choice isolates the architectural questions studied in the paper from compilation overhead and hardware noise.

2.2 Brightness-range sensitivity sweep for encoder scaling

Besides the main translated-MNIST architecture sweep and the full-MNIST size sweep, we ran a separate low-data sensitivity sweep for the encoder angle scale. The article-wide convention $(a, b) = (0, \pi)$ was chosen a priori as a one-period real FRQI color-map convention and was not selected by optimizing this sweep. The sweep is included as a post-hoc sanity check of sensitivity to the upper endpoint b, not as a validation procedure for tuning the reported benchmark models. No reported benchmark setting,

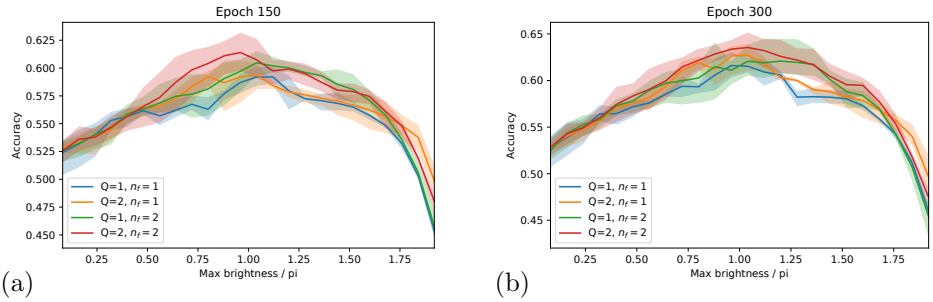

Fig. S1 Brightness-range sensitivity sweep for the FRQI encoder scale. Panels show accuracy at epochs 150 and 300 as a function of the upper endpoint b/π in the map $p = a + (b-a)x$ with fixed lower endpoint $a = 0$. The article-wide benchmark convention $(a, b) = (0, \pi)$ was fixed independently of this sweep. Each curve corresponds to one architecture $(Q, n_f) \in \{1, 2\} \times \{1, 2\}$; lines show the mean over 3 seeds and shaded bands show the seedwise minimum–maximum range.

architecture choice, or quoted main-result accuracy is selected using the accuracy values in this sweep.

The sweep uses translated-MNIST with 20 training examples per class, while keeping the standard test split membership unchanged. Both train and test images are resized to 28×28, placed on a 32×32 canvas, and translated by independently sampled integer offsets of at most 2 pixels per axis. For each independent run, the prepared images are fixed and reused for all epochs and evaluations. The PCS-QCNN is evaluated with full readout for 300 training epochs, train/test batch sizes 256/1600, and 3 seeds. We evaluate PCS-QCNN architectures with feature-qubit counts $n_f \in \{1, 2\}$ and quantum depths $Q \in \{1, 2\}$. For each architecture we keep the lower endpoint fixed at $a = 0$ and vary the upper endpoint as $b = \beta\pi$ with

$$\beta \in \left\{ \frac{2}{25}, \frac{4}{25}, \ldots, \frac{48}{25} \right\},$$

that is, 24 evenly spaced interior points spanning the open interval $(0, 2\pi)$. Accuracy is recorded at epochs 150 and 300, corresponding to the two panels in Fig. S1. For each architecture and each value of β, the line shows the mean over seeds and the band shows the seedwise minimum–maximum range.

The sweep is a post-hoc sensitivity check around the fixed $[0, \pi]$ one-period convention for the local real FRQI color map; it does not imply that longer intervals are strictly redundant for the full coherent image state. With

$$|\phi(p)\rangle = \sin(p)\,|0\rangle + \cos(p)\,|1\rangle\,,$$

one has $|\phi(p+\pi)\rangle = -\,|\phi(p)\rangle$, so the isolated one-qubit ray is revisited after a shift by π. In the FRQI image superposition, however, address-dependent sign changes are relative phases across address components and can affect subsequent QFT/PCS layers. The brightness sweep is therefore reported as a robustness/sensitivity diagnostic only. All reported benchmark families fix the encoder interval to $(a, b) = (0, \pi)$ independently of this sweep.

(a) (b)

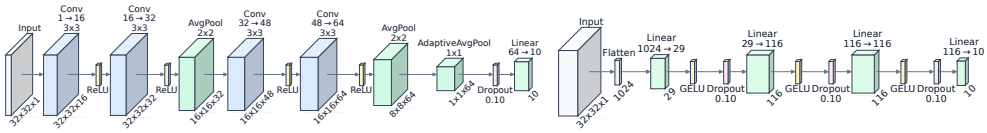

Fig. S2 Classical reference architectures used in Fig. **??**(a). The same 32×32 input models are used for the translated benchmark in the main text and for the full-MNIST control in Sec. 2.4. (a) Convolutional CNN baseline with 47,034 trainable parameters. (b) Pure-dense MLP control with 47,947 trainable parameters.

2.3 Classical baseline architectures for benchmark comparisons

The classical reference models used in Fig. **??**(a) are specified here for reproducibility (main-text Secs. **??** and **??**).

For the CNN baseline (Fig. S2(a)), the input is a 32×32 grayscale canvas produced by the shared preprocessing protocol. In Fig. **??**(a), this architecture is evaluated on the translated benchmark with 16×16 digits placed on a 32×32 canvas, and the same network is also used for the full-MNIST control without translations. The network applies

$$\mathrm{Conv}(1 \to 16, 3 \times 3) \to \mathrm{ReLU} \to \mathrm{Conv}(16 \to 32, 3 \times 3) \to \mathrm{ReLU} \to \mathrm{AvgPool}(2 \times 2)$$

followed by

$$\mathrm{Conv}(32 \to 48, 3 \times 3) \to \mathrm{ReLU} \to \mathrm{Conv}(48 \to 64, 3 \times 3) \to \mathrm{ReLU} \to \mathrm{AvgPool}(2 \times 2)$$

and finally $\mathrm{AdaptiveAvgPool}(1 \times 1) \to \mathrm{Dropout}(0.10) \to \mathrm{Linear}(64 \to 10)$. This model has 47,034 trainable parameters.

For the MLP baseline (Fig. S2(b)), the same 32×32 input is flattened to length 1024 and passed through

$$\mathrm{Linear}(1024 \to 29) \to \mathrm{GELU} \to \mathrm{Dropout}(0.10) \to \mathrm{Linear}(29 \to 116) \to \mathrm{GELU} \to \mathrm{Dropout}(0.10)$$

followed by

$$\mathrm{Linear}(116 \to 116) \to \mathrm{GELU} \to \mathrm{Dropout}(0.10) \to \mathrm{Linear}(116 \to 10).$$

For 32×32 inputs this model has 47,947 trainable parameters.

For comparison, the fixed translated-MNIST PCS-QCNN and the matched random-basis control in Fig. **??**(b) both use $Q = 3$, $n_f = 2$, full readout on the same translated benchmark with 16×16 digits placed on a 32×32 canvas, and 39,434 total trainable parameters (36,864 in the quantum core and 2,570 in the classifier head). This random-basis control is a non-PCS symmetry-breaking ablation of the Fourier construction, not an explicitly QCS-equivariant or MERA-style baseline.

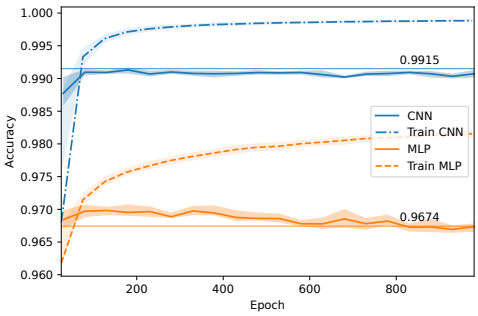

Fig. S3 Classical CNN and MLP controls on the full standard MNIST split resized directly to 32×32 without translations. The architectures are exactly those specified in Fig. S2; only the data regime changes. Solid lines show mean test accuracy, train curves show mean train accuracy, and shaded bands show the 25th–75th percentile range over 3 seeds. The gap is much smaller than on the translated benchmark, which illustrates why non-translated MNIST alone is a less stringent benchmark for testing convolutional inductive bias.

2.4 Full-MNIST classical control without translations

The same CNN and MLP baselines were also evaluated on the full standard MNIST split (60,000 training images and 10,000 test images) after resizing each digit directly to a 32×32 grayscale image, with no canvas translation. The model architectures were kept identical to those specified in Sec. 2.3, so the comparison changes only the data regime and not the parameter budget. Training followed the same fixed-baseline protocol as the main-text classical comparison except for the training horizon: three random seeds, 1000 epochs, train/test accuracy tracked through time, and the same percentile-band summary convention as in Fig. **??**(a).

Figure S3 shows that on this non-translated full-MNIST task the convolutional and dense baselines remain much closer than on translated-MNIST, with final mean test accuracies of 99.15% and 96.74%, respectively. This control is the reason the main text uses the translated benchmark with 16×16 digits on a 32×32 canvas when discussing convolution-sensitive inductive bias: removing translations makes the classical gap much less informative.

2.5 Readout entropy diagnostic

The main text reports initialization-time quantum-gradient diagnostics in Fig. **??**, distinguishing the norm of the empirical-loss gradient from the per-sample RMS gradient norm. The readout entropy diagnostic below checks how finite-shot sampling changes the classifier input. Figure S4 shows the Shannon entropy of the full readout distribution under finite-shot reevaluation for the representative direct-16×16 reference run after epochs $100, 200, \ldots, 2000$ and shot budgets 128, 256, 512, 1024, and 2048; each curve reports the mean over test samples and the shaded band shows the interquartile range. The lower entropy at smaller shot budgets is partly expected for a simple combinatorial reason: an N-shot histogram can have at most N nonzero outcomes. Within that limitation, the entropy increases moderately during training, indicating that sampled readouts spread over more outcomes instead of collapsing onto a small

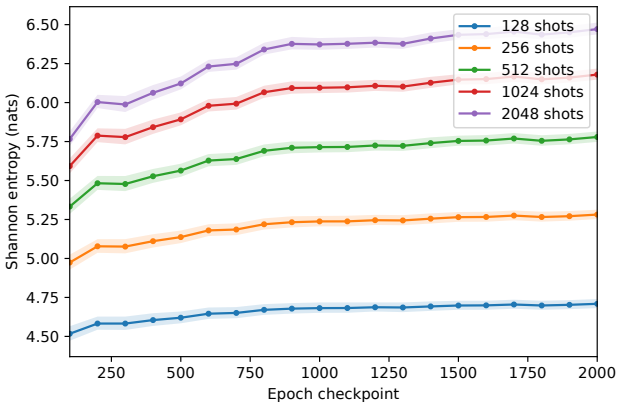

Fig. S4 Mean Shannon entropy of the full readout distribution versus training epoch for the representative direct-16 × 16 reference run after epochs $100, 200, \ldots, 2000$ and finite-shot budgets 128, 256, 512, 1024, and 2048; shaded bands show the interquartile range over test samples.

subset. Thus, finite-shot readout can distort the effective classifier input even when the exact-probability model has a stable gradient signal.

2.6 Geometric interpretation: local loss landscape in readout space

The loss-landscape diagnostic probes the classical head under finite-shot perturbations expected from the multinomial readout model. Figure S5 uses the representative direct-16 × 16 full-MNIST reference PCS-QCNN after 100 and 800 training epochs, shot budget $N = 128$, and an 81×81 grid of coordinates $(\alpha, \beta) \in [-3, 3] \times [-3, 3]$. For each test sample x with label y, let $p(x)$ denote the exact infinite-shot readout distribution produced by the trained quantum part, and let h denote the fixed trained classical head mapping readout vectors to class logits. We define the sample-local multinomial covariance

$$C_x = \frac{\operatorname{diag}(p(x)) - p(x)p(x)^\top}{N}, \qquad N = 128,$$

take its two leading eigenpairs $(\lambda_1(x), e_1(x))$ and $(\lambda_2(x), e_2(x))$, and form the two local perturbation directions

$$u_x = \sqrt{\lambda_1(x)}\, e_1(x), \qquad v_x = \sqrt{\lambda_2(x)}\, e_2(x).$$

These directions define local sigma coordinates in readout-probability space for that sample. At each grid point (α, β), we perturb the exact readout as

$$q_x(\alpha, \beta) = p(x) + \alpha u_x + \beta v_x.$$

No renormalization is applied. A sample contributes to a grid cell only when every component of $q_x(\alpha, \beta)$ is nonnegative and $\sum_j q_{x,j}(\alpha, \beta) \le 1 + 10^{-5}$. Writing $\mathcal{V}_{\alpha,\beta} \subseteq \mathcal{T}$

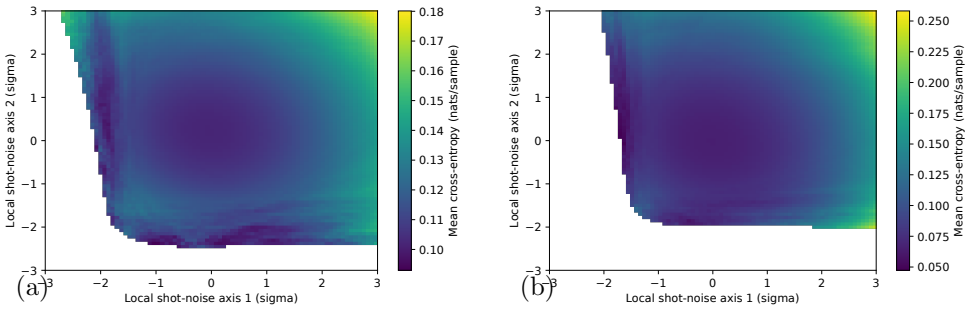

Fig. S5 Readout-space loss landscape for the representative direct-16 × 16 reference PCS-QCNN after (a) 100 and (b) 800 training epochs. Each test sample is perturbed in its own two-dimensional local shot-noise PCA basis, the resulting cross-entropy is averaged over valid test samples on an 81×81 grid covering $[-3, 3]^2$ in sigma units for $N = 128$ shots, and cells with valid fraction below 0.10 are shown in white. Each panel has its own colorbar.

for the subset of test samples satisfying this validity condition at grid point (α, β), the plotted quantities are

$$L(\alpha,\beta) = \frac{1}{|\mathcal{V}_{\alpha,\beta}|} \sum_{(x,y)\in\mathcal{V}_{\alpha,\beta}} \ell(h(q_x(\alpha,\beta)), y), \qquad f_{\text{valid}}(\alpha,\beta) = \frac{|\mathcal{V}_{\alpha,\beta}|}{|\mathcal{T}|},$$

where ℓ is the usual cross-entropy and $\mathcal{T}$ is the full test set. At $(\alpha, \beta) = (0, 0)$, no perturbation is applied, so $q_x(0, 0) = p(x)$ and the heatmap value equals the exact infinite-shot mean test loss of that trained model. White cells indicate grid points with $f_{\text{valid}}(\alpha, \beta) < 0.10$, i.e. fewer than 10% of test samples yield valid perturbed readouts there.

For the model trained for 100 epochs (Fig. S5(a)), the loss rises relatively gradually away from the exact solution over most of the region that remains well defined. After 800 epochs (Fig. S5(b)), the center value is lower, reflecting the lower exact infinite-shot test loss, but the local increase away from the center is steeper and higher losses are reached within only a few shot-noise standard deviations. Because each panel is rendered with its own colorbar, this comparison should be based on the numeric scales and the local rise away from the center rather than on hue alone. In this empirical sense, the 800-epoch model is locally sharper in readout space along typical finite-shot perturbation directions than the 100-epoch model. This local sharpening is consistent with the observed need, in this run, for shot budgets on the order of 10^3 to reliably preserve the gains of longer exact-probability training.

2.7 Qualitative error structure

Figure S6 reports the remaining classification errors for the representative final direct-16 × 16 full-MNIST reference model after 2000 training epochs (confusion matrix and examples of misclassified digits). These errors are not dominated by a single failure mode; rather, they reflect the expected ambiguity between visually similar digits after aggressive downscaling to a 16 × 16 quantum input.

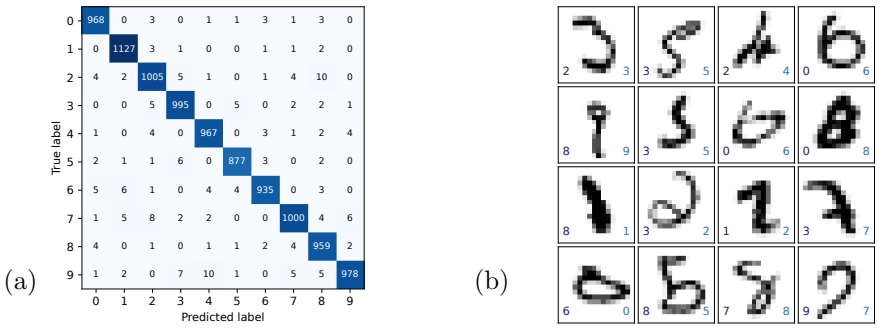

Fig. S6 Qualitative error analysis for the representative final direct-16×16 full-MNIST PCS-QCNN after 2000 training epochs. (a) Confusion matrix (rows: true labels, columns: predicted labels); darker cells indicate higher counts. (b) Examples of misclassified MNIST digits with predicted labels shown in the lower-left corner and true labels in the lower-right corner of each image.

3 Classical convolution primer

The main text (Sec. **??**) states the key symmetry fact (convolution as commutation with translations) and the Fourier diagonalization template [15, 16]. A slightly more detailed classical discussion is reproduced here.

Beyond this algebraic characterization, two recent theory papers are directly relevant to our benchmarking logic. Li *et al.* [17] exhibit settings where convolutional architectures can be learned with fewer samples than parameter-matched fully connected models under standard symmetry-preserving optimization assumptions. Lahoti *et al.* [18] further separate the roles of locality and weight sharing (CNNs vs. locally connected neural networks (LCNs) vs. fully connected neural networks (FCNs)), showing that these architectural priors can produce clear sample-complexity gaps on translation-related tasks. The operational consequence is direct: exposing the effect of inductive bias requires a reduced-data regime in which this bias is visible. Accordingly, the main text uses an explicit translated-MNIST benchmark (1000 training examples per class, with digits resized to 16×16, placed on a 32×32 canvas, and translated by at most 8 pixels per axis; Sec. **??**) in addition to the full-MNIST size sweep.

3.1 Classical convolution and translation symmetry

A classical feedforward network is a composition of layers of the form $z = f(Ax)$, where A is a linear operator and f is a non-linear activation. In many spatial problems, the same local pattern is applied at every position: each output feature at location k depends on inputs in the same relative way at any location.

In one dimension this leads to the familiar convolutional form

$$y_k = \sum_{n=0}^{N-1} a_{k-n}\, x_n,$$

where indices are taken modulo N. Define the circular shift operators T_k by

$$(T_k x)_j = x_{j-k}, \qquad k \in \mathbb{Z},$$

with indices modulo N. A matrix A is circulant if and only if it commutes with all shifts,

$$T_k A = A T_k \qquad \forall k.$$

In that case, multiplication by A is exactly a circular convolution, $y = a \star x$. Enforcing this commutation relation at the architectural level is what weight sharing means in classical CNNs, and it reduces the number of degrees of freedom from N^2 (dense) to N (circulant), producing the classical convolutional inductive bias.

In practical models one works with multiple channels. Then each scalar coefficient becomes a linear map on the channel space and the input/output acquire a channel index:

$$(Az)_{k,j} = \sum_{n=0}^{N-1} \sum_{l} a_{k-n,j,l}\, z_{n,l}.$$

For images the spatial grid is two- (or three-) dimensional. In two dimensions, with translation symmetry along each axis, the shifts act separately on each coordinate. For an input z_{n_1,n_2} we write

$$(T_k \otimes I)\, z_{n_1,n_2} = z_{n_1-k,n_2}, \qquad (I \otimes T_k)\, z_{n_1,n_2} = z_{n_1,n_2-k}.$$

A two-dimensional convolution layer commutes with all translations $T_x \otimes T_y$ and has the standard form

$$(Az)_{x,y,j} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} \sum_{l} a_{x-n,y-m,j,l}\, z_{n,m,l}. \tag{S1}$$

A key structural fact is that circulant (and block-circulant) operators are diagonal in the Fourier basis. Let F_N be the discrete Fourier transform,

$$(F_N x)_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \omega^{kn}, \qquad \omega = e^{-2\pi i/N}. \tag{S2}$$

In two dimensions, $(F_N \otimes F_M)^\dagger A (F_N \otimes F_M)$ becomes block-diagonal, with blocks acting on channels:

$$(\hat{A}\hat{z})_{p_1,p_2,j} = \sum_{l} \hat{a}_{p_1,p_2,j,l}\, \hat{z}_{p_1,p_2,l}. \tag{S3}$$

This Fourier characterization is the classical template we will mirror in the quantum construction.

4 Proofs of the QCS–PCS mismatch and Fourier-multiplexer theorem

This section gives complete proofs of the two structural statements used in the main text: Lemma **??**, which separates qubit cyclic shifts (QCS) from pixel cyclic shifts (PCS) under address encoding, and Theorem **??**, which identifies the unitary commutant of PCS with a Fourier-mode multiplexer. We use the notation of main-text Sec. **??**.

Lemma S1 (Detailed form of the QCS–PCS mismatch)**.** *For pixel-to-qubit encodings, a cyclic translation of pixels can be represented by a cyclic permutation of physical qubits, up to the convention for shift direction. For address/amplitude encodings with a nontrivial index register, the pixel cyclic shift is the modular-addition operator*

$$T\,|j\rangle = |j+1 \bmod N\rangle\,, \qquad N = 2^n,$$

on the computational basis of the index register. This operator is not a cyclic permutation of the physical qubits of that register. Consequently, QCS-equivariance does not in general imply PCS-equivariance.

Proof In a pixel-to-qubit encoding, the computational tensor factors are indexed by pixels. If a basis state is written as

$$|q_0\rangle\,|q_1\rangle\cdots|q_{N-1}\rangle\,,$$

then a one-pixel cyclic translation sends the value stored at each pixel to the neighboring pixel. As an operator on the encoded qubits this is precisely a cyclic permutation of tensor factors, either S or S^{-1} depending on whether the spatial shift is taken to the left or to the right. Thus, in this encoding family, QCS can coincide with the pixel translation symmetry.

For address/amplitude encodings, the spatial coordinate is not a tensor-factor label but the binary value of the index register. The pixel shift is therefore modular addition on the computational basis:

$$T\,|j\rangle = |j+1 \bmod N\rangle\,.$$

Every permutation of physical qubits leaves the all-zero basis state fixed:

$$\Pi\,|0\rangle^{\otimes n} = |0\rangle^{\otimes n}$$

for any qubit permutation Π. For $N > 1$, however,

$$T\,|0\rangle^{\otimes n} = |1\rangle \neq |0\rangle^{\otimes n}\,.$$

Here $|1\rangle$ denotes the integer-one address state on the n-qubit index register. Hence the address-encoded PCS operator T is not equal to any qubit permutation, and in particular it is not equal to a cyclic qubit shift.

It remains to show that commutation with a qubit cyclic shift need not imply commutation with T. For $n \geq 2$, take $U = S$, where S is the cyclic permutation of the n index qubits. Then U commutes with S trivially, but S and T do not commute:

$$ST\,|0\rangle^{\otimes n} = S\,|1\rangle \neq |1\rangle = TS\,|0\rangle^{\otimes n}\,.$$

For the one-qubit case, $S = I$ while $T = X$, so any unitary such as Z commutes with S but not with T. Thus QCS-equivariance alone does not enforce PCS-equivariance under address encoding. □

Theorem S1 (Detailed form of the Fourier-multiplexer characterization)**.** *Let T be the cyclic shift on an index register $\mathcal{H}_{\mathrm{idx}} \cong \mathbb{C}^N$, and let $\mathcal{H}_{\mathrm{feat}}$ be an arbitrary finite-dimensional feature space. A unitary U on $\mathcal{H}_{\mathrm{idx}} \otimes \mathcal{H}_{\mathrm{feat}}$ commutes with $T \otimes I$ if and only if*

$$U = (F_N^\dagger \otimes I) \left(\bigoplus_{k=0}^{N-1} U_k \right) (F_N \otimes I),$$

where F_N is the N-point Fourier transform on the index register and each U_k is a unitary on $\mathcal{H}_{\mathrm{feat}}$. In two and three spatial dimensions, the same statement holds with tensor-product Fourier transforms and with blocks indexed by the corresponding tuple of Fourier modes.

Proof We first prove the one-dimensional statement. Let

$$F_N |j\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{kj} |k\rangle, \qquad \omega_N = e^{-2\pi i/N}.$$

With the shift convention $T|j\rangle = |j+1 \bmod N\rangle$, direct substitution gives

$$F_N T F_N^\dagger = \Omega := \sum_{k=0}^{N-1} \omega_N^k |k\rangle\langle k|.$$

Define

$$B := (F_N \otimes I) U (F_N^\dagger \otimes I).$$

Then

$$[U, T \otimes I] = 0 \quad \Longleftrightarrow \quad [B, \Omega \otimes I] = 0.$$

The eigenvalues ω_N^k of Ω are all distinct. Therefore each Fourier-mode projector $P_k = |k\rangle\langle k|$ is a polynomial in Ω, explicitly

$$P_k = \prod_{r \neq k} \frac{\Omega - \omega_N^r I}{\omega_N^k - \omega_N^r}.$$

If B commutes with $\Omega \otimes I$, it commutes with every $P_k \otimes I$. Hence B preserves each subspace $|k\rangle \otimes \mathcal{H}_{\mathrm{feat}}$, and its off-diagonal Fourier-mode blocks vanish:

$$(P_k \otimes I) B (P_l \otimes I) = 0 \qquad (k \neq l).$$

Thus

$$B = \sum_{k=0}^{N-1} (P_k \otimes I) B (P_k \otimes I) = \bigoplus_{k=0}^{N-1} U_k,$$

where

$$U_k = (\langle k| \otimes I)\, B\, (|k\rangle \otimes I)$$

acts on $\mathcal{H}_{\mathrm{feat}}$. Because B is unitary and block diagonal, each block U_k is unitary. Conjugating back by $F_N \otimes I$ gives the claimed Fourier-multiplexer form.

Conversely, if

$$B = \bigoplus_{k=0}^{N-1} U_k$$

in the Fourier basis, then B is block diagonal with respect to the eigenspaces of $\Omega \otimes I$. It therefore commutes with $\Omega \otimes I$, and the conjugated unitary

$$U = (F_N^\dagger \otimes I) B (F_N \otimes I)$$

commutes with $T \otimes I$.

The multidimensional case is the simultaneous version of the same argument. In two dimensions, set $F = F_{N_x} \otimes F_{N_y}$ and

$$B = (F \otimes I)U(F^\dagger \otimes I).$$

Commutation with $T_x \otimes I$ and $T_y \otimes I$ is equivalent to commutation of B with

$$\Omega_x \otimes I \quad \text{and} \quad \Omega_y \otimes I,$$

where

$$\Omega_x = \sum_{k_x,k_y} \omega_{N_x}^{k_x} |k_x, k_y\rangle \langle k_x, k_y|, \qquad \Omega_y = \sum_{k_x,k_y} \omega_{N_y}^{k_y} |k_x, k_y\rangle \langle k_x, k_y|.$$

Here $\omega_M = e^{-2\pi i/M}$. The joint spectral projector onto a fixed pair (k_x, k_y) is the product of the one-dimensional spectral projectors for Ω_x and Ω_y. Therefore B commutes with every joint projector if and only if it is block diagonal over the joint Fourier modes:

$$B = \bigoplus_{k_x=0}^{N_x-1} \bigoplus_{k_y=0}^{N_y-1} U_{k_x,k_y}.$$

Unitarity of B again implies unitarity of all feature-space blocks. The converse is immediate because such a block-diagonal B commutes with both diagonal translation operators. The three-dimensional statement follows by adding the third commuting generator and the corresponding joint spectral projectors. □

The proof shows a slightly stronger algebraic fact: the same block-diagonal characterization holds for arbitrary linear maps in the commutant of the translation action. Restricting to unitary maps simply restricts the Fourier-mode blocks to be unitary.

5 MERA-QCNN templates and QCS equivariance

The main text (Sec. **??**) uses MERA-QCNN as a representative example of a common QCNN design pattern that enforces cyclic-shift symmetry on physical qubits (QCS). The longer discussion is included here.

5.1 MERA-QCNN as a QCS-equivariant architecture

Figure **??**(a) shows a MERA-inspired QCNN, originally proposed in Ref. [19] and used in several subsequent works. The architecture consists of repeated local unitaries arranged in a multiscale pattern, interleaved with pooling that reduces the number of active qubits, typically implemented by measurements followed by classically controlled operations (or, equivalently, by postponing measurement and replacing it by controlled gates due to standard circuit identities [20]).

The essential structural ingredient is weight sharing along a line of qubits, but full one-site QCS equivariance also requires the layout itself to respect the cyclic action. Many MERA/QCNN templates can be made QCS-equivariant by imposing cyclic weight sharing together with periodic boundary conditions, a periodic or symmetrized placement pattern containing the full cyclic orbit of each local block, and pooling/readout rules compatible with the same qubit permutation. Without these

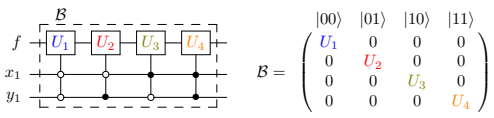

Fig. S7 Decomposition of a block-diagonal multiplexer $\mathcal{B}$ into controlled operations U_i on the feature qubits. The example uses a 2×2 image (one qubit in each of the x and y registers) for clarity.

additional choices, a standard brickwork layout generally commutes only with the subgroup preserved by its tiling and pooling pattern, rather than with the full cyclic shift S. In our terminology, circuits satisfying this commutation property are QCS-QCNNs. As Lemma **??** emphasizes, this is well matched to pixel-to-qubit encodings, but it does not in general enforce the translation action T induced by address encoding, which is the setting of interest here.

6 Multiplexer and Fourier-junction realization

Two circuit-level points are useful for the implementation discussion: (i) how a block-diagonal multiplexer $\mathcal{B}$ can be decomposed into controlled operations, and (ii) how the inverse Fourier transform at the end of one layer and the forward Fourier transform at the start of the next layer collapse across pooling boundaries.

6.1 Fourier cancellation at the interface of PCS-QCNN layers

Figure **??** shows that consecutive PCS layers contain adjacent inverse and forward Fourier transforms on the index registers. In our architecture the pooling step is implemented after the inverse QFT of a layer: in the computational index basis we split the fine index as $r = 2q + s$, measure the parity bit s, discard it, and use the outcome to classically condition the parameters of the next layer. After previous pooling steps, the active index is already the corresponding coarse coordinate; the next pooling step again measures the least significant bit of this current active computational index. Equivalently, successive pooling steps remove the original computational-index bits from least to most significant order. The next-layer QFT $F_{N/2}$ acts only on the surviving coarse index q. In the reduced Fourier-domain representation this spatial-parity pooling appears as an aliasing operation on Fourier modes: the same coarse Fourier label q receives contributions from the pair $k = q$ and $k = q+N/2$. As a result, the only nontrivial Fourier junction between two consecutive PCS layers reduces to a simple isometry on the remaining index qubits, which can be implemented without explicit forward or inverse QFT blocks.

Lemma S2 (Reduction of the Fourier junction)**.** *Let $N = 2^n$ and let the pooling projection in the computational index basis be*

$$P_b \;:=\; \sum_{q=0}^{N/2-1} |q\rangle\langle 2q+b| \;:\; \mathbb{C}^N \to \mathbb{C}^{N/2},$$

where $b \in \{0,1\}$ is the measured parity bit in the split $r = 2q+b$. On the Fourier-input side, introduce the alias-ordering isomorphism

$$A_N : \mathbb{C}^{N/2} \otimes \mathbb{C}^2 \to \mathbb{C}^N, \qquad A_N \,|q\rangle\,|\sigma\rangle = |q + \sigma N/2\rangle\,,$$

so that σ distinguishes the two Fourier modes that alias to the same coarse label q. For $b \in \{0,1\}$ define the postselected junction map

$$K_b \;:=\; F_{N/2}\, P_b\, F_N^\dagger \;:\; \mathbb{C}^N \to \mathbb{C}^{N/2}. \tag{S4}$$

Then

$$K_b A_N \;=\; G^{\,b}\,(I \otimes \langle b|\, H), \tag{S5}$$

where H is the Hadamard acting on the Fourier alias selector σ, $G^{\,b}$ means I for $b = 0$ and G for $b = 1$, and G is the diagonal phase-gradient operator on the remaining $(n-1)$-qubit register defined by

$$G\,|q\rangle \;=\; e^{+2\pi i\, q/N}\,|q\rangle\,. \tag{S6}$$

Equivalently, for $|\psi\rangle = \sum_{k=0}^{N-1} \alpha_k\,|k\rangle$ one has

$$K_b\,|\psi\rangle = \frac{1}{\sqrt{2}} \sum_{q=0}^{N/2-1} e^{+2\pi i\, qb/N}\big(\alpha_q + (-1)^b \alpha_{q+N/2}\big)\,|q\rangle\,. \tag{S7}$$

Proof Let $\omega = e^{-2\pi i/N}$, so that

$$F_N^\dagger\,|k\rangle = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} \omega^{-kr}\,|r\rangle\,.$$

With the decomposition $r = 2q + s$ (so $|r\rangle \equiv |q\rangle\,|s\rangle$) we have

$$P_b F_N^\dagger\,|k\rangle = \frac{1}{\sqrt{N}} \sum_{q=0}^{N/2-1} \omega^{-k(2q+b)}\,|q\rangle = \frac{\omega^{-kb}}{\sqrt{N}} \sum_{q=0}^{N/2-1} \omega^{-2kq}\,|q\rangle\,.$$

Applying $F_{N/2}$ and using $\omega_{N/2} = e^{-2\pi i/(N/2)} = \omega^2$ gives, for each output basis state $|p\rangle$,

$$\langle p|\, K_b\,|k\rangle = \frac{1}{\sqrt{N(N/2)}}\, \omega^{-kb} \sum_{q=0}^{N/2-1} \omega^{2(p-k)q}.$$

The inner sum is a geometric series. Since $\omega^{2(p-k)}$ is an $(N/2)$-th root of unity, it evaluates to

$$\sum_{q=0}^{N/2-1} \omega^{2(p-k)q} = \frac{N}{2}\,\delta_{p,\,k \bmod (N/2)}.$$

Hence

$$\langle p|\, K_b\,|k\rangle = \frac{1}{\sqrt{2}}\,\omega^{-kb}\,\delta_{p,\,k \bmod (N/2)}.$$

Writing $k = q + \sigma\frac{N}{2}$ with $\sigma \in \{0,1\}$ and $q \in \{0,\dots,N/2-1\}$, we obtain

$$K_b\,|q\rangle = \frac{1}{\sqrt{2}}\,\omega^{-qb}\,|q\rangle\,, \quad K_b\,|q+N/2\rangle = \frac{1}{\sqrt{2}}\,\omega^{-(q+N/2)b}\,|q\rangle = \frac{(-1)^b}{\sqrt{2}}\,\omega^{-qb}\,|q\rangle\,.$$

By linearity, for $|\psi\rangle = \sum_k \alpha_k\,|k\rangle$ this yields (S7), since $\omega^{-qb} = e^{+2\pi iqb/N}$. Finally, (S5) follows in the alias ordering $A_N\,|q\rangle\,|\sigma\rangle = |q+\sigma N/2\rangle$: $(I \otimes \langle b|\,H)$ produces the factor $(\alpha_q + (-1)^b\alpha_{q+N/2})/\sqrt{2}$ on each $|q\rangle$, while G^b contributes the phase $e^{+2\pi iqb/N}$ from (S6). □

Lemma S3 (Circuit implementation of the reduced junction)**.** *Let $q = \sum_{j=0}^{n-2} q_j 2^j$ be the binary expansion of the $(n-1)$-qubit index. Then the phase-gradient operator (S6) can be implemented (up to a global phase) as a tensor product of single-qubit Z-rotations:*

$$G \;\equiv\; \bigotimes_{j=0}^{n-2} R_z^{(j)}\Big(+\frac{\pi}{2^{\,n-1-j}}\Big)\,, \tag{S8}$$

where $R_z^{(j)}(\cdot)$ acts on qubit j of the $(n-1)$-qubit index register. Consequently, the reduced junction is implemented in the Fourier alias ordering by: (i) apply H to the alias selector σ in $k = q + \sigma N/2$, (ii) measure it obtaining the same branch label b, and (iii) conditionally on $b = 1$ apply the gates from (S8) to the remaining $(n-1)$ index qubits (do nothing if $b = 0$), i.e., implement G^b with $G^0 = I$ and $G^1 = G$.

Proof For $N = 2^n$ and $q = \sum_{j=0}^{n-2} q_j 2^j$,

$$e^{+2\pi iq/N} = \prod_{j=0}^{n-2} e^{+i\pi q_j/2^{\,n-1-j}}\,.$$

The single-qubit gate $R_z(\theta) = \exp(-i\theta Z/2)$ is diagonal and acts as $|0\rangle \mapsto e^{-i\theta/2}\,|0\rangle$, $|1\rangle \mapsto e^{+i\theta/2}\,|1\rangle$. Therefore, on a computational basis state $|q\rangle = \bigotimes_{j=0}^{n-2} |q_j\rangle$,

$$\left(\bigotimes_{j=0}^{n-2} R_z(\theta_j)\right)|q\rangle = \left(\bigotimes_{j=0}^{n-2} e^{-i\theta_j/2}\right)\left(\prod_{j=0}^{n-2} e^{+i\theta_j q_j}\right)|q\rangle\,.$$

Choosing $\theta_j = +\pi/2^{\,n-1-j}$ yields the q-dependent factor $\prod_j e^{+i\pi q_j/2^{\,n-1-j}} = e^{+2\pi iq/N}$. The remaining prefactor is independent of q, hence it is a global phase. □

The same construction extends directly to two dimensions. In the 2D architecture (registers $\mathrm{Reg}\,x$ and $\mathrm{Reg}\,y$), spatial-parity pooling on the two registers produces outcomes (b_x, b_y), and the reduced Fourier junction factorizes as $K_{b_x}^{(x)} \otimes K_{b_y}^{(y)}$ with $N \mapsto N_x$ and $N \mapsto N_y$ in Lemmas S2-S3. Thus the interface of two consecutive layers can be implemented by two Hadamards, two measurements, and two independent conditional phase-gradient blocks on the remaining index qubits, all in the corresponding Fourier alias orderings.

The hybrid PCS-QCNN has the following canonical form after Fourier cancellation, in notation consistent with main-text Sec. **??**. Lemmas S2-S3 imply that the inverse Fourier transform at the end of one layer and the forward Fourier transform at the

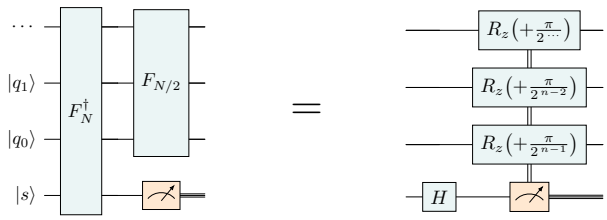

Fig. S8 Reduction of the Fourier junction between two adjacent PCS-QCNN layers. Left: the original junction $F_{N/2}\,\mathcal{M}_s\,F_N^\dagger$, where $\mathcal{M}_s$ measures the spatial parity bit in the split $r = 2q + s$. Right: the reduced implementation from Lemma S3: in the Fourier alias split $k = q+\sigma N/2$, apply H and measure σ, then conditionally (for outcome $b = 1$) apply the single-qubit Z-rotations on the q-register.

start of the next layer can be replaced by a fixed junction map, so trainable parameters remain only inside layer multiplexer blocks.

We describe the 2D specialization ($d = 2$) with registers Reg x and Reg y. Let n_{idx} be the number of index qubits per axis before the first layer, and let Q be the number of quantum layers. At layer $\ell \in \{1, \ldots, Q\}$, the number of active index qubits per axis is $n_\ell = n_{\mathrm{idx}} - \ell + 1$. For $\ell < Q$, pooling measures one qubit per axis and produces $m_\ell = (b_x^{(\ell)}, b_y^{(\ell)}) \in \{0, 1\}^2$.

After cancellation of intermediate Fourier transforms, the only parameterized operations are the multiplexer blocks

$$\mathcal{B}^{(\ell)}(m_{\ell-1}) = \bigoplus_{k_x=0}^{2^{n_\ell}-1} \bigoplus_{k_y=0}^{2^{n_\ell}-1} V_{k_x,k_y}^{(\ell)}(m_{\ell-1}),$$

with $m_0 = 0$ and $V_{k_x,k_y}^{(\ell)}(m_{\ell-1}) \in U(D_f)$ acting on the feature register. The inter-layer junction is fixed and parameter-free: it is implemented by local Hadamards, measurement of pooling qubits, and conditional phase-gradient gates (Lemmas S2-S3).

Equivalently, the same architecture can be written in layer form as

$$U^{(\ell)}(m_{\ell-1}) = (\mathcal{F}^{(\ell)\dagger} \otimes I)\, \mathcal{B}^{(\ell)}(m_{\ell-1})\, (\mathcal{F}^{(\ell)} \otimes I),$$

which is exactly Eq. (**??**) in main-text Sec. **??**. The full measured-and-controlled process defines the output distribution $p_\Theta(z \mid x)$ from main-text Eq. (**??**).

The gradient-accounting discussion below uses the following fact: all trainable parameters are inside $\mathcal{B}^{(\ell)}$, while inter-layer junction maps are fixed. This separation is used in Sec. 8.

7 Idealized benchmark model and resource scaling

The resource scaling summarized in main-text Sec. **??** should be read together with the main modeling caveat: the reported PCS-QCNN is an idealized symmetry benchmark model. It uses direct state initialization, dense-tensor statevector evolution, and

explicit block-diagonal application of fully general Fourier multiplexers. The following discussion gives qubit, depth, and shot-cost accounting for moving such a model toward hardware, where state preparation and multiplexer compilation become central costs.

7.1 Complexity analysis

In the NISQ regime it is important to keep both qubit requirements and circuit depth in view. We begin with the qubit count. For an $N \times N$ image with $N = 2^{n_{\mathrm{idx}}}$, address encoding uses two index registers with $n_{\mathrm{idx}} = \log_2 N$ qubits each. In addition, we use a feature register of size n_f qubits, with the grayscale/color qubit included in n_f by convention. Thus

$$n_{\mathrm{tot}} = 2n_{\mathrm{idx}} + n_f = 2\log_2 N + n_f. \tag{S9}$$

In the reported 16×16 benchmark, $n_{\mathrm{idx}} = 4$ and $n_f \in \{1, 2, 3\}$, hence $n_{\mathrm{tot}} \in \{9, 10, 11\}$.

The key point for circuit depth is that trainable and compiled operations are not the same thing. After cancellation of intermediate Fourier transforms (Sec. 6.1), the trainable operations are precisely the multiplexer blocks $\mathcal{B}^{(\ell)}$. However, on hardware one still has to implement the Fourier transforms and compile the multiplexers into one- and two-qubit gates. The Fourier-transform contribution is modest at the reported image sizes. For an n-qubit QFT, the standard circuit uses n Hadamards and $n(n-1)/2$ controlled-phase gates (plus optional swaps), hence gate count $O(n^2)$ and depth $O(n^2)$ without additional parallelization [20–22]. In PCS-QCNN, QFT acts separately on the x and y registers, so at n_x and n_y qubits per axis this yields a per-layer overhead on the order of $O(n_x^2+n_y^2)$ two-qubit entangling gates. For our 16×16 case ($n_x = n_y = 4$), this cost is small compared to the multiplexer compilation below.

The multiplexer synthesis is the main depth bottleneck. In the Fourier basis, a PCS layer applies a mode-wise block-diagonal unitary of the form

$$\mathcal{B}^{(\ell)}(m) = \sum_{k_x,k_y} |k_x, k_y\rangle\langle k_x, k_y| \otimes U^{(\ell)}_{k_x,k_y}(m), \tag{S10}$$

where each block $U^{(\ell)}_{k_x,k_y}(m)$ acts on the n_f feature qubits. This is a multi-target quantum multiplexer: a uniformly controlled $U(2^{n_f})$ gate with $n_c = n_x + n_y$ control qubits. General decompositions of uniformly controlled gates scale exponentially in n_c in the worst case: one needs $O(2^{n_c})$ controlled blocks, and if each $U^{(\ell)}_{k_x,k_y}(m)$ is treated as a generic $U(2^{n_f})$ unitary, the total two-qubit gate count is $O(2^{n_c}\,4^{n_f})$ up to architecture-dependent constants [23, 24]. In our benchmark, $n_c = 8$ and $n_f \le 3$, so this worst-case compilation cost is still moderate, but it grows quickly with image resolution.

This hardware compilation cost is not paid in the statevector experiments. In our classical simulations we apply each $\mathcal{B}^{(\ell)}$ directly as a block-diagonal unitary map. This benchmark evaluates the PCS-aligned inductive bias under within-family controls: the PCS-QCNN and RBC-QCNN have identical parameter counts, while the classical CNN and MLP controls are closely parameter-matched. Hardware-oriented variants would likely restrict $U^{(\ell)}_{k_x,k_y}(m)$ (e.g., shallow local ansatzes on the feature register

and/or parameter sharing across modes) to reduce depth while preserving exact PCS equivariance at the unitary-block level.

Finally, finite-shot readout introduces a separate resource cost. The measurement cost is proportional to the shot budget N_{shot} required to estimate the readout distribution $p_\Theta(\cdot \mid x)$ with the desired accuracy. This dependence is intrinsic: in hybrid quantum-classical learning, measurement precision is a computational resource and must be treated as part of the model design.

8 Gradient-accounting derivation

The elementary accounting estimate used in Sec. **??** is derived here. The estimate concerns the mean per-coordinate second moment. The key point is that, in the depth-scaling family, the aggregate gradient-energy bound grows at most linearly in Q, whereas the fully general Fourier multiplexer has exponentially many scalar coordinates.

8.1 PCS-QCNN family and parameter count

We work with an index register consisting of d spatial axes. Each axis initially has n_{idx} qubits. A depth-Q PCS-QCNN pools after the first $Q-1$ layers, each time measuring and discarding one computational-index qubit per axis. After pooling, the remaining index register has

$$n_l := n_{\mathrm{idx}} - Q + 1 \tag{S11}$$

qubits per axis, hence

$$D_{\mathrm{idx}} := 2^{d n_l}. \tag{S12}$$

The feature register has n_f qubits and dimension

$$D_f := 2^{n_f}. \tag{S13}$$

The final measurement is performed on the remaining index qubits and the feature register, so

$$D_{\mathrm{out}} := D_{\mathrm{idx}} D_f. \tag{S14}$$

In the depth-scaling family used in the main text,

$$n_l,\ d,\ D_f,\ M \text{ are fixed}, \qquad n_{\mathrm{idx}} = n_l + Q - 1. \tag{S15}$$

Thus the readout dimension remains fixed as Q increases.

At layer $\ell \in \{1, \dots, Q\}$, the active index width per axis is

$$n_\ell = n_{\mathrm{idx}} - \ell + 1 = n_l + Q - \ell, \tag{S16}$$

and the active index dimension is

$$N_\ell = 2^{d n_\ell}. \tag{S17}$$

Let $\mathcal{F}^{(\ell)}$ denote the d-fold QFT on the active index register at layer ℓ. The branch label available to layer ℓ is drawn from

$$\mathcal{M}_\ell := \begin{cases} \{0\}, & \ell = 1, \\ \{0,1\}^d, & \ell \geq 2, \end{cases} \tag{S18}$$

where the singleton label for $\ell = 1$ represents the absence of a previous pooling outcome. This is the benchmark convention: later multiplexers are conditioned on the most recently pooled d-bit outcome, not on the full measurement history.

The Fourier multiplexer at layer ℓ is

$$\mathcal{B}^{(\ell)}(m_{\ell-1}) := \sum_{k \in [N_\ell]} |k\rangle\langle k| \otimes V_k^{(\ell)}(m_{\ell-1}), \tag{S19}$$

where each block $V_k^{(\ell)}(m_{\ell-1}) \in U(D_f)$ acts only on the feature register. The PCS-equivariant unitary block is

$$U^{(\ell)}(m_{\ell-1}) := \big(\mathcal{F}^{(\ell)\dagger} \otimes \mathbb{I}\big)\mathcal{B}^{(\ell)}(m_{\ell-1})\big(\mathcal{F}^{(\ell)} \otimes \mathbb{I}\big). \tag{S20}$$

For $\ell < Q$, this unitary is followed by parity pooling on the computational index register; the final layer is followed by measurement of the remaining index and feature registers. The full measurement-and-feedforward process defines

$$p_\Theta(z \mid x), \qquad z \in [D_{\text{out}}], \tag{S21}$$

where Θ denotes all quantum parameters.

The classical head is a linear map followed by a softmax,

$$q(x) = \text{softmax}\big(W p_\Theta(\cdot \mid x) + b\big) \in \Delta^M. \tag{S22}$$

The unitary PCS blocks are exactly equivariant on their active index registers. Pooling yields a multiscale/coarse equivariance structure rather than exact one-pixel equivariance of the complete readout/classifier pipeline. This unconstrained linear head is not symmetry-tied across readout coordinates, so exact PCS equivariance is a property of the unitary quantum convolutional blocks rather than of the full classifier output. In the reported classifier, translation invariance of the final labels is learned from data rather than imposed exactly. For label $c \in \{1, \ldots, M\}$ the loss is

$$\mathcal{L}(\Theta, W, b \mid x, c) := -\log q_c. \tag{S23}$$

The same notation applies to a minibatch or empirical-risk loss by replacing $\mathcal{L}$ with the corresponding average.

For a dataset or minibatch $\mathcal{D} = \{(x_i, c_i)\}_{i=1}^N$, define the single-sample quantum gradients

$$g_i := \nabla_\Theta \mathcal{L}(\Theta, W, b \mid x_i, c_i). \tag{S24}$$

The gradient used by a first-order optimizer for the averaged loss is

$$G_{\mathcal{D}} := \nabla_{\Theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\Theta, W, b \mid x_i, c_i) = \frac{1}{N} \sum_{i=1}^{N} g_i. \tag{S25}$$

By contrast, the per-sample RMS diagnostic is

$$R_{\mathcal{D}}^2 := \frac{1}{N} \sum_{i=1}^{N} \|g_i\|_2^2. \tag{S26}$$

These are not interchangeable plateau diagnostics. Taking expectation over random initialization,

$$\mathbb{E}\|G_{\mathcal{D}}\|_2^2 = \frac{1}{N^2} \sum_{i=1}^{N} \mathbb{E}\|g_i\|_2^2 + \frac{1}{N^2} \sum_{i \neq j} \mathbb{E}\langle g_i, g_j \rangle. \tag{S27}$$

Thus $R_{\mathcal{D}}^2$ measures the diagonal, single-sample contribution, whereas $\|G_{\mathcal{D}}\|_2^2$ also includes cross-sample alignment or cancellation. For optimization of the empirical risk, $\|G_{\mathcal{D}}\|_2$ is the directly relevant norm; $R_{\mathcal{D}}$ is a diagnostic for whether individual samples still carry gradient signal.

For the full Pauli-exponential ansatz used in the benchmark, each feature block has

$$p_{\text{blk}} = 4^{n_f} \tag{S28}$$

real scalar parameters. The number of quantum-core parameters is

$$P_Q = p_{\text{blk}} \sum_{\ell=1}^{Q} |\mathcal{M}_\ell| N_\ell = p_{\text{blk}} \sum_{\ell=1}^{Q} b_\ell \, 2^{d(n_l+Q-\ell)}, \qquad b_1 = 1, \quad b_\ell = 2^d \ (\ell \geq 2). \tag{S29}$$

In particular,

$$P_Q \geq p_{\text{blk}} \, 2^{d(n_l+Q-1)}. \tag{S30}$$

For completeness, for $Q \geq 2$ the exact closed form is

$$P_Q = p_{\text{blk}} \left[2^{d(n_l+Q-1)} + 2^d \, 2^{dn_l} \frac{2^{d(Q-1)} - 1}{2^d - 1} \right], \tag{S31}$$

so $P_Q = \Theta(2^{dQ})$ at fixed n_l, d, n_f.

8.2 Aggregate gradient-energy estimate

The benchmark initialization samples every quantum Pauli coefficient independently as

$$\theta_{k,m,\alpha}^{(\ell)} \sim \text{Unif}(0, 2\pi). \tag{S32}$$

Independently, the classical head uses fan-in uniform initialization. For head input dimension D_{out}, this means

$$W_{az} \sim \text{Unif}\left(-D_{\text{out}}^{-1/2}, D_{\text{out}}^{-1/2}\right), \qquad b_a \sim \text{Unif}\left(-D_{\text{out}}^{-1/2}, D_{\text{out}}^{-1/2}\right), \qquad a \in [M],\ z \in [D_{\text{out}}], \tag{S33}$$

independently over all entries. Because D_{out} is fixed in Eq. (S15), this head distribution is the same for every depth Q. The uniform quantum initialization fixes the experimental protocol, while the bounds below are deterministic in the quantum parameters and use only independence from the head.

Lemma S4 (Softmax-head gradient bound)**.** *Let*

$$q = \text{softmax}(Wp + b) \in \Delta^M, \qquad \mathcal{L}(p, c) = -\log q_c.$$

Then

$$\|\nabla_p \mathcal{L}(p, c)\|_2^2 \le 2\|W\|_{\text{op}}^2. \tag{S34}$$

Proof The standard cross-entropy/softmax derivative gives

$$\nabla_p \mathcal{L} = W^\top (q - e_c).$$

Since $q \in \Delta^M$, one has $\|q - e_c\|_2^2 \le 2$, and the claim follows. □

Lemma S5 (Fan-in uniform linear-softmax head: directional second-moment bound)**.** *Assume the fan-in uniform head initialization* (S33)*, with $L = D_{\text{out}}$. Fix $p \in \Delta^L$, a label $c \in [M]$, and a deterministic vector $v \in \mathbb{R}^L$. Then*

$$\mathbb{E}_{W,b}\left[\left\langle W^\top(\text{softmax}(Wp + b) - e_c), v\right\rangle^2 \mid p\right] \le \frac{2M}{3}\|v\|_2^2. \tag{S35}$$

Proof Set $q = \text{softmax}(Wp + b)$ and $r = q - e_c$. For every realization of W, b, $\|r\|_2^2 \le 2$, and therefore

$$\left\langle W^\top r, v\right\rangle^2 \le \|W^\top r\|_2^2 \|v\|_2^2 \le 2\|W\|_{\text{op}}^2 \|v\|_2^2 \le 2\|W\|_F^2 \|v\|_2^2.$$

Under (S33), each weight has variance $1/(3L)$, so

$$\mathbb{E}\|W\|_F^2 = ML\,\frac{1}{3L} = \frac{M}{3}.$$

Taking expectation gives (S35). □

Lemma S6 (Bounded tangents of Pauli-exponential blocks)**.** *Let*

$$U(\theta) = \exp\left(i \sum_{\alpha \in \mathcal{P}_{n_f}} \theta_\alpha P_\alpha\right),$$

where $\mathcal{P}_{n_f}$ is the full Pauli-string basis, including the all-identity string. For each coordinate θ_α there exists an operator $K_\alpha(\theta)$ such that

$$\partial_{\theta_\alpha} U(\theta) = K_\alpha(\theta) U(\theta), \qquad \|K_\alpha(\theta)\| \le 1. \tag{S36}$$

Proof Write $A(\theta) = \sum_\beta \theta_\beta P_\beta$. The derivative of the matrix exponential is

$$\partial_{\theta_\alpha} U(\theta) = i \int_0^1 e^{i(1-s)A(\theta)} P_\alpha e^{isA(\theta)}\, ds.$$

Each integrand has operator norm $\|P_\alpha\| = 1$, hence $\|\partial_{\theta_\alpha} U(\theta)\| \le 1$. Taking $K_\alpha(\theta) = (\partial_{\theta_\alpha} U(\theta)) U(\theta)^\dagger$ gives (S36). □

Lemma S7 (Layer sensitivity-energy accounting)**.** *Fix an input x, all model parameters, and a layer ℓ. For each coordinate $\theta^{(\ell)}_{k,\alpha}(m)$ in that layer, let*

$$s^{(\ell)}_{k,m,\alpha}(x) := \partial_{\theta^{(\ell)}_{k,\alpha}(m)} p_\Theta(\cdot \mid x) \in \mathbb{R}^{D_{\text{out}}}.$$

Then

$$\sum_{m \in \mathcal{M}_\ell} \sum_{k \in [N_\ell]} \sum_{\alpha \in \mathcal{P}_{n_f}} \|s^{(\ell)}_{k,m,\alpha}(x)\|_2^2 \le 4 p_{\text{blk}}. \tag{S37}$$

Proof Use the deferred-measurement dilation of the pooling process. For fixed ℓ, write the final measured state as

$$|\Psi(\theta)\rangle = \mathcal{U}_{\Theta(\theta)}\big(\, |\psi(x)\rangle \otimes |0\rangle_{\text{rec}}\,\big),$$

and write the final readout probabilities as

$$p_\Theta(z \mid x) = \langle \Psi |\, \Pi_z \,| \Psi \rangle\,,$$

where the orthogonal projectors $\{\Pi_z\}$ include the marginalization over deferred condition registers. For one coordinate, let $\dot\Psi$ denote the derivative of $|\Psi(\theta)\rangle$ with respect to that coordinate. Then

$$\partial_\theta p_\Theta(z \mid x) = 2\,\mathrm{Re}\left\langle \dot\Psi \middle|\, \Pi_z \,| \Psi \right\rangle,$$

and orthogonality of the projectors gives

$$\|\partial_\theta p_\Theta(\cdot \mid x)\|_2^2 \le 4 \|\dot\Psi\|_2^2.$$

It remains to sum $\|\dot\Psi\|_2^2$ over the coordinates in layer ℓ. Immediately before the layer-ℓ multiplexer, let $|\Phi\rangle$ be the normalized dilated state, including the deferred record register. For branch label m and Fourier mode k, the derivative of the selected block has the form $K_\alpha V^{(\ell)}_k(m)$ with $\|K_\alpha\| \le 1$ by Lemma S6. Thus, for an isometry $\mathcal{W}$ representing the rest of the circuit,

$$\dot\Psi = \mathcal{W}\big(\Pi_m \otimes |k\rangle\langle k| \otimes K_\alpha V^{(\ell)}_k(m) \otimes I\big)\,|\Phi\rangle\,,$$

where Π_m is the projector onto the relevant deferred pooling record (with the singleton branch at $\ell = 1$). Therefore

$$\|\dot\Psi\|_2^2 \le \|(\Pi_m \otimes |k\rangle\langle k| \otimes I)\,|\Phi\rangle\,\|_2^2.$$

For each fixed α, the projectors over (m, k) are mutually orthogonal and sum to at most the identity, so the sum of the right-hand side over (m, k) is at most 1. Summing over the p_{blk} Pauli coordinates gives (S37). □

To combine the two sensitivity estimates, consider first a single sample and condition on the quantum parameters, so the readout vector $p = p_\Theta(\cdot \mid x)$ and all sensitivity vectors

$$s^{(\ell)}_{k,m,\alpha}(x) = \partial_{\theta^{(\ell)}_{k,\alpha}(m)} p_\Theta(\cdot \mid x)$$

are fixed. The chain rule gives

$$\partial_{\theta^{(\ell)}_{k,\alpha}(m)} \mathcal{L} = \left\langle W^\top(\mathrm{softmax}(Wp + b) - e_c), s^{(\ell)}_{k,m,\alpha}(x) \right\rangle .$$

By Lemma S5,

$$\mathbb{E}_{W,b}\left[\left(\partial_{\theta^{(\ell)}_{k,\alpha}(m)} \mathcal{L} \right)^2 \mid \Theta \right] \le \frac{2M}{3} \| s^{(\ell)}_{k,m,\alpha}(x) \|_2^2 .$$

Summing over all coordinates in one layer and applying Lemma S7 gives

$$\mathbb{E}_{W,b}\left[\sum_{m,k,\alpha} \left(\partial_{\theta^{(\ell)}_{k,\alpha}(m)} \mathcal{L} \right)^2 \mid \Theta \right] \le \frac{8M}{3} p_{\mathrm{blk}} .$$

Summing over Q layers and then taking expectation over any independent quantum initialization gives

$$\mathbb{E} \| \nabla_{\Theta_Q} \mathcal{L}(\Theta_Q, W, b) \|_2^2 \le \frac{8M}{3} p_{\mathrm{blk}} Q. \tag{S38}$$

For a minibatch or empirical-risk average, the quantum gradient is the corresponding average of the single-sample gradients, and convexity of the squared norm gives the same bound after averaging over samples.

Finally, let $G_Q = \nabla_{\Theta_Q} \mathcal{L}(\Theta_Q, W, b)$. Dividing (S38) by the deterministic parameter-count lower bound (S30) yields the main-text accounting estimate

$$\mathbb{E}\left[\frac{1}{P_Q} \| G_Q \|_2^2 \right] \le \frac{8MQ}{3\, 2^{d(n_l+Q-1)}} . \tag{S39}$$

For $M = 10$, the numerator constant is $80Q/3$. This is the parameter-count effect discussed in the main text. The aggregate norm $\mathbb{E}\|G_Q\|_2^2$ and the layerwise coordinates relevant to a particular optimizer or dataset are separate trainability quantities. The main text therefore reports both the empirical-loss gradient norm and the per-sample RMS gradient norm directly.

References

[1] LeCun, Y., Cortes, C., Burges, C.J.C.: MNIST Handwritten Digit Database. http://yann.lecun.com/exdb/mnist/. Accessed 2026-04-23 (2010)

[2] Bowles, J., Ahmed, S., Schuld, M.: Better than Classical? The Subtle Art of Benchmarking Quantum Machine Learning Models (2024). https://doi.org/10.48550/arXiv.2403.07059 . https://arxiv.org/abs/2403.07059

[3] Mattern, D., Martyniuk, D., Willems, H., Bergmann, F., Paschke, A.: Variational Quanvolutional Neural Networks with Enhanced Image Encoding (2021). https://doi.org/10.48550/arXiv.2106.07327 . https://arxiv.org/abs/2106.07327

[4] Huang, S.-Y., An, W.-J., Zhang, D.-S., Zhou, N.-R.: Image classification and adversarial robustness analysis based on hybrid quantum-classical convolutional neural network. Optics Communications **533**, 129287 (2023) https://doi.org/10.1016/j.optcom.2023.129287

[5] Li, W., Chu, P.-C., Liu, G.-Z., Tian, Y.-B., Qiu, T.-H., Wang, S.-M.: An image classification algorithm based on hybrid quantum classical convolutional neural network. Quantum Engineering **2022**, 5701479 (2022) https://doi.org/10.1155/2022/5701479

[6] Oh, S., Choi, J., Kim, J.K., Kim, J.: Quantum convolutional neural network for resource-efficient image classification: A quantum random access memory (QRAM) approach. In: Proceedings of the 2021 International Conference on Information Networking (ICOIN), pp. 50–52. IEEE, ??? (2021). https://doi.org/10.1109/ICOIN50884.2021.9333906

[7] Huggins, W., Patil, P., Mitchell, B., Whaley, K.B., Stoudenmire, E.M.: Towards quantum machine learning with tensor networks. Quantum Science and Technology **4**(2), 024001 (2019) https://doi.org/10.1088/2058-9565/aaea94

[8] Jing, Y., Li, X., Yang, Y., Wu, C., Fu, W., Hu, W., Li, Y., Xu, H.: RGB image classification with quantum convolutional ansatz. Quantum Information Processing **21**(3), 101 (2022) https://doi.org/10.1007/s11128-022-03442-8

[9] Easom-McCaldin, P., Bouridane, A., Belatreche, A., Jiang, R.: On depth, robustness and performance using the data re-uploading single-qubit classifier. IEEE Access **9**, 65127–65139 (2021) https://doi.org/10.1109/ACCESS.2021.3075492

[10] Zheng, J., Gao, Q., Lü, J., Ogorzałek, M., Pan, Y., Lü, Y.: Design of a quantum convolutional neural network on quantum circuits. Journal of the Franklin Institute **360**(17), 13761–13777 (2023) https://doi.org/10.1016/j.jfranklin.2022.07.033

[11] Hur, T., Kim, L., Park, D.K.: Quantum convolutional neural network for classical data classification. Quantum Machine Intelligence **4**(1), 3 (2022) https://doi.org/10.1007/s42484-021-00061-x

[12] Huang, F., Tan, X., Huang, R., Xu, Q.: Variational convolutional neural networks classifiers. Physica A: Statistical Mechanics and its Applications **605**, 128067 (2022) https://doi.org/10.1016/j.physa.2022.128067

[13] Li, Y., Zhou, R.-G., Xu, R., Luo, J., Hu, W.: A quantum deep convolutional neural network for image recognition. Quantum Science and Technology **5**(4),

044003 (2020) https://doi.org/10.1088/2058-9565/ab9f93

[14] Wei, S.-J., Chen, Y.-H., Zhou, Z.-R., Long, G.-L.: A quantum convolutional neural network on NISQ devices. APL Photonics **7**(5), 056103 (2022) https://doi.org/10.1063/5.0083444

[15] Bamieh, B.: Discovering Transforms: A Tutorial on Circulant Matrices, Circular Convolution, and the Discrete Fourier Transform (2022). https://doi.org/10.48550/arXiv.1805.05533 . https://arxiv.org/abs/1805.05533

[16] Gray, R.M.: Toeplitz and circulant matrices: A review. Foundations and Trends in Communications and Information Theory **2**(3), 155–239 (2006) https://doi.org/10.1561/0100000006

[17] Li, Z., Zhang, Y., Arora, S.: Why are convolutional nets more sample-efficient than fully-connected nets? In: Proceedings of the 9th International Conference on Learning Representations (ICLR) (2021). https://openreview.net/forum?id=uCY5MuAxcxU

[18] Lahoti, A., Karp, S., Winston, E., Singh, A., Li, Y.: Role of locality and weight sharing in image-based tasks: A sample complexity separation between CNNs, LCNs, and FCNs. In: Proceedings of the 12th International Conference on Learning Representations (ICLR) (2024). https://openreview.net/forum?id=AfnsTnYphT

[19] Cong, I., Choi, S., Lukin, M.D.: Quantum convolutional neural networks. Nature Physics **15**(12), 1273–1278 (2019) https://doi.org/10.1038/s41567-019-0648-8

[20] Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge University Press, Cambridge (2010)

[21] Coppersmith, D.: An Approximate Fourier Transform Useful in Quantum Factoring (2002). https://arxiv.org/abs/quant-ph/0201067

[22] Cleve, R., Watrous, J.: Fast parallel circuits for the quantum fourier transform. In: Proceedings of the 41st Annual Symposium on Foundations of Computer Science, pp. 526–536. IEEE, ??? (2000). https://doi.org/10.1109/SFCS.2000.892140

[23] Bergholm, V., Vartiainen, J.J., Möttönen, M., Salomaa, M.M.: Quantum circuits with uniformly controlled one-qubit gates. Physical Review A **71**(5), 052330 (2005) https://doi.org/10.1103/PhysRevA.71.052330

[24] Shende, V.V., Bullock, S.S., Markov, I.L.: Synthesis of quantum-logic circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **25**(6), 1000–1010 (2006) https://doi.org/10.1109/TCAD.2005.855930