PHYSICAL REVIEW APPLIED **22,** 024011 (2024)

Enhanced measurement of neutral-atom qubits with machine learning

L. Phuttitarn,[1] B. M. Becker,[1] R. Chinnarasu,[1] T. M. Graham,[1] and M. Saffman[1,2,*]

[1] *Department of Physics, University of Wisconsin-Madison, 1150 University Avenue, Madison, Wisconsin 53706, USA*

[2] *Infleqtion, Inc., Madison, Wisconsin, 53703, USA*

(Received 20 November 2023; revised 1 May 2024; accepted 24 June 2024; published 5 August 2024)

We demonstrate qubit-state measurements assisted by a supervised convolutional neural network (CNN) in a neutral-atom quantum processor. We present two CNN architectures for analyzing neutral-atom qubit readout data: a compact five-layer single-qubit CNN architecture and a six-layer multiqubit CNN architecture. We benchmark both architectures against a conventional Gaussian-threshold analysis method. In a sparse array (9-μm atom separation) which experiences negligible crosstalk, we have observed up to 32% and 56% error reduction for the multiqubit and single-qubit architectures, respectively, as compared to the benchmark. In a tightly spaced array (5-μm atom separation), which suffers from readout crosstalk, we have observed up to 43% and 32% error reduction in the multiqubit and single-qubit CNN architectures, respectively, as compared to the benchmark. By examining the correlation between the predicted states of neighboring qubits, we have found that the multiqubit CNN architecture reduces the crosstalk correlation by up to 78.5%. This work demonstrates a proof of concept for a CNN network to be implemented as a real-time readout-processing method on a neutral-atom quantum computer, enabling faster readout time and improved fidelity.

DOI: 10.1103/PhysRevApplied.22.024011

I. INTRODUCTION

Scalable quantum computation requires precise high-fidelity initialization, control, and measurement of the quantum state of a large number of qubits. The number of sequential operations that can be performed is limited by qubit decoherence, resulting from imperfect qubit control mechanisms and unintended interaction with the environment. The leading approach to overcome this limitation is midcircuit-measurement-based quantum error correction (see Ref. [1] and references therein). The ability to make high-fidelity measurements with minimal collateral disruption to the system is not only relevant to initialization and final readout—it is also essential to achieving quantum error correction.

Neutral-atom quantum computing has matured remarkably in recent years. Single- and two-qubit gates have been demonstrated on neutral atoms with fidelity well above 99% [2,3]. Multiqubit quantum circuits [4,5] and midcircuit measurements have also been demonstrated [6–11]. Qubit-state measurements in a neutral-atom array are achieved by probing the array with light detuned from a cycling transition. The resulting fluorescence is captured with a high quantum efficiency imaging device such as an electron-multiplying charge-coupled device (EMCCD) or scientific CMOS (sCMOS) sensor, producing a grayscale image of the neutral-atom array. Conventionally, the state of the qubit is then determined by integrating the photon counts over regions of interest (ROIs) and applying a linear threshold that optimally separates the probability distributions of the two states (see Fig. 3). The fidelity of the state detection is dependent upon the separability of the two distributions, which in turn depends on the signal-to-noise ratio (SNR). To achieve a fidelity above 99% with this method, the typical probing period is tens of milliseconds. This is a significant delay, given that the longest gate operation only takes several microseconds. Shortening the exposure time decreases the probing period but also reduces the SNR and the fidelity. One could increase the power of the probing laser to compensate, but too much power can cause atom loss through heating. This compromise limits the SNR and imposes a lower bound on the probing period, given a desired measurement fidelity and an acceptable rate of qubit loss.

It is possible to further reduce the probing period without loss of fidelity by using more efficient image-analysis algorithms. Examples include independent component analysis [12], the Bayesian inference algorithm [13] and supervised deep neural networks (DNNs) [14–16]. In this work, we present enhanced state detection using a convolutional neural network (CNN). CNNs are a subcategory of DNNs that are especially well suited for image classification. Supervised DNNs have already been

---

[*] Contact author: msaffman@wisc.edu

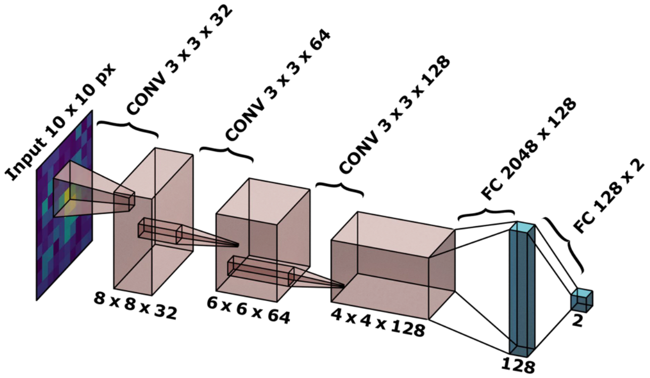


FIG. 1. The CNN-site network structure. The readout containing multiple sites in the array of qubits is preprocessed and sectioned into single sites before feeding into the network. The preprocessing consists of mean subtraction and normalization. For more detailed process description, see Appendix A. Parameters of the CNN-site design are given in Table I. The single-qubit readout is fed into the three convolutional layers. The final features are extracted on the first FC layer and connected to the output layer. The two nodes on the output layer represent the probability of the two quantum states.

demonstrated to improve readout fidelity on other leading quantum computing platforms, including trapped-ion [17,18], superconducting [19,20], and quantum dot qubits [21,22]. This work demonstrates the use of deep learning for neutral-atom qubit-state detection.

II. CNN DESIGNS

Since state detection for neutral-atom qubits is performed by imaging the atoms, a CNN is ideal for the task. The principles of how CNNs work are well known and we refer to the existing literature for more details [23]. In what follows, we focus on the particulars of the CNNs used in our experiments.

We have implemented and benchmarked two different CNN architectures for qubit-state detection. The first architecture, CNN-site, is a single-qubit classifier. Analyzing one qubit at a time, it replaces the conventional linear-threshold method for real-time applications, achieving superior detection fidelity. The second architecture, CNN-array, is a multiqubit classifier. Because it analyzes the entire qubit array in parallel, it is crosstalk aware. Compared to CNN-site, it boasts improved detection fidelity on large tightly spaced arrays at the expense of real-time computational efficiency. Both architectures are implemented in PYTHON using the KERAS application-programming interface (API) [24] and the TensorFlow library [25] with graphics processing unit (GPU) acceleration.

A. CNN-site

CNN-site is a compact five-layer single-qubit-state classifier, as shown in Fig. 1. It consists of three convolutional (CONV) layers followed by two fully connected (FC) layers. The activation function is a rectified linear unit ($g(\mathbf{z}) = \max(\mathbf{0}, \mathbf{z})$) for all layers except the final FC layer, which uses a softmax activation function [26]. As input, it receives a cropped monochromatic image of a single site of the atomic array. The input images are preprocessed by mean subtraction and normalization (for details, see Appendix A). The output nodes represent the weighted probabilities of the atom being in a dark or bright state.

TABLE I. The filter shape, output shape, and number of trainable parameters in each layer of CNN-site.

Layer	Filter shape	Output shape	Number of parameters
Input	No filter	10 × 10 × 1	0
CONV	3 × 3 × 1 × 32	8 × 8 × 32	320
CONV	3 × 3 × 32 × 64	6 × 6 × 64	18 496
CONV	3 × 3 × 64 × 128	4 × 4 × 128	73 856
Flatten	No filter	2048	0
FC	2048 × 128	128	262 272
FC	128 × 2	2	258
Total trainable parameters:			355 202

The dimensions of the CNN layers have been experimentally varied in order to discover the set of parameters that consistently converge to a high state-detection fidelity during training. Appreciably shrinking the network results in sporadic convergence failures (as indicated by low fidelities corresponding to random guessing) due to nondeterministic initialization of the network weights. On the other hand, appreciably increasing the width or depth of the network does not yield further fidelity improvements—the fidelities are almost identical amongst networks that do converge. This reflects the relative simplicity of the input images; the task can be completed with only three convolutional layers because higher-complexity features, which additional convolutional layers could extract, are not present in the data. We believe that CNN-site could be further optimized without any detrimental impact to detection fidelity but this would require further iterative fine tuning and would not alter the conclusions of this work. The number of parameters is already very small in comparison to typical CNNs. Its low parameter count makes it quick enough for real-time applications, including atom rearrangement and midcircuit measurement if accelerated using a neural-network accelerator (NNA) or a field-programmable gate array (FPGA).

B. CNN-array

CNN-array is a six-layer multiqubit-state classifier that improves state detection in closely spaced arrays where crosstalk between the neighboring sites is appreciable. As input, CNN-array accepts preprocessed images of the full

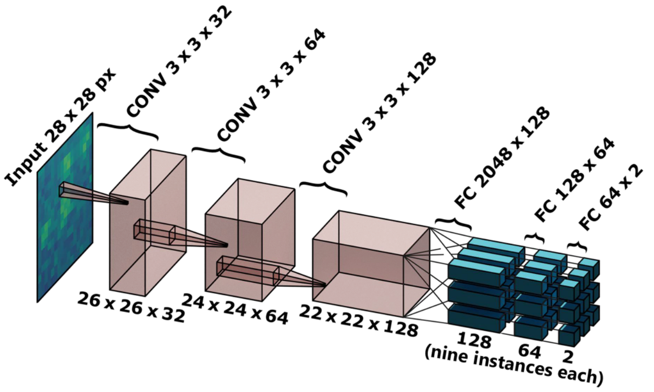


FIG. 2. The CNN-array network structure. The raw readout image is preprocessed and fed directly into the convolutional layers. Parameters of the CNN-array design are given in Table II. The features detected by the last convolutional layers are connected to multiple FC networks, each responsible for the classification of a single qubit. For the 3 × 3 array, there are nine independent FC networks.

atomic array. Its architecture, pictured in Fig. 2, is identical to that of CNN-site except that (a) the input image is larger, (b) there is an additional FC layer, and (c) there are multiple instances of each FC layer, one per qubit in the array. It produces $\mathcal{N}$ simultaneous outputs with the same format as CNN-site, indicating the two-state probabilities of each qubit in the array. Since each instance of the FC layers can "see" the features of all of the other qubits in the array, CNN-array can identify and accommodate correlations between neighboring qubits. Owing to the larger input image and the all-to-all fan-out of the connections of the FC layers, CNN-array contains approximately 210 times more parameters than CNN-site, making real-time implementation challenging, even with an NNA or FPGA. Optimization is likely possible and is a subject for future work.

C. Gaussian threshold and square threshold

For completeness, we present the conventional Gaussian-threshold and square-threshold methods, which we have used as our performance benchmark.

First, the region of interest of each atom (the location in the image where the atom is expected to be found) for each site is determined by averaging all the readout images in the entire data set and applying a circular Gaussian fit to the result. This locates the center of the distribution, corresponding to the central position of the atom, and the standard deviation σ, corresponding to the size of the atom in pixels. For the Gaussian-threshold method, the resulting best-fit parameters are then used to generate a two-dimensional (2D) Gaussian mask for each site: a value in the range 0.0–1.0 for each pixel in the image. For the square-threshold method, we replace the Gaussian mask with a binary square mask with sides of length 2σ.

TABLE II. The output shape and number of trainable parameters in each layer in CNN-array. Compared to CNN-site, the number of trainable parameters is substantially higher (approximately 210 times).

Layer	Filter shape	Output shape	Number of parameters
Input	No filter	$28 \times 28 \times 1$	0
CONV	$3 \times 3 \times 1 \times 32$	$26 \times 26 \times 32$	320
CONV	$3 \times 3 \times 32 \times 64$	$24 \times 24 \times 64$	18 496
CONV	$3 \times 3 \times 64 \times 128$	$22 \times 22 \times 128$	73 856
Flatten	No filter	61 952	0
FC	$(2048 \times 128) \times 9$	128×9	74 613 888
FC	$(128 \times 64) \times 9$	64×9	74 304
FC	$(64 \times 2) \times 9$	2×9	1170
Total trainable parameters:			74 782 034

The mask is multiplied element-wise with each readout image and then summed over all pixels, returning a single integrated value per atom. The quantum state of the atom is predicted to be a bright (dark) state if the integrated value is above (below) a predetermined threshold. The threshold itself is determined by collating the integrated data of many images of randomly loaded arrays. The histograms of the collated data are shown in Fig. 3(b) (left and right), along with the threshold (dotted line). The distribution resembles a mixture of two Gaussian distributions, one for each binary state (dark or bright). The parameters of the two-Gaussian mixture can be obtained by applying a curve fit to the histogram. The threshold that maximizes the separation between the two states is the intersection point of the two fitted Gaussian curves.

III. EXPERIMENTAL METHODS

To collect data, Cs atoms are loaded into a 3 × 3 tweezer array generated by a 2D acousto-optical deflector (AOD). The array is imaged with 852-nm light that is red detuned from the $6s_{1/2}, f = 4$–$6p_{3/2}, f = 5$ transition by 9γ ($\gamma = 2\pi \times 5.2$ MHz). Fluorescence light is collected via the two opposing 0.7-NA objective lenses. State measurements are performed destructively by first pushing out one of the states, followed by occupancy imaging with a hyperfine repump laser to prevent atoms going dark due to Raman transitions. Although state measurements can be performed nondestructively [27], the destructive method used here provides for the lowest possible uncertainty in state measurements. The two collection paths are projected onto two adjacent regions of the EMCCD sensor (Andor Ixon 897).

In order to train and evaluate the performance of the two CNN networks, we have acquired a data set consisting of noisy input data paired with the corresponding "ground-truth" labels. We have collected data at site spacings of 5 µm and 9 µm, varying the readout time (secondary-path state separation) from 10 ms (2.17σ) to 100 ms (66.6σ).

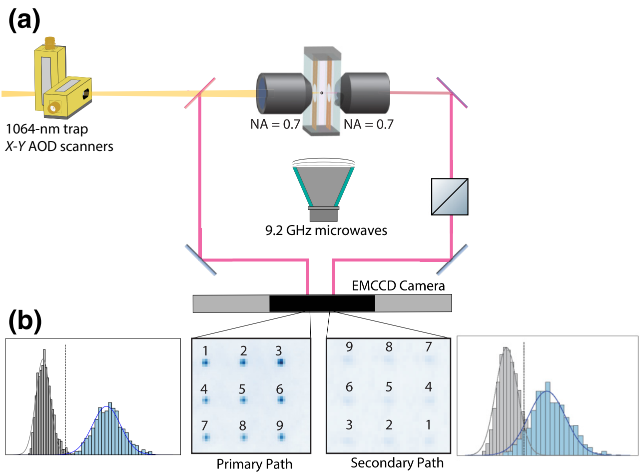


FIG. 3. (a) The experimental layout for dual-path readout collection. Atoms are cooled and loaded into a 3×3 array of optical tweezers. To read out the qubit state, the array is illuminated with 852-nm light and transition fluorescence is collected via two opposing high-NA lenses. The two collection paths are projected onto separate regions of the EMCCD sensor. A nonpolarizing beam splitter is placed in the secondary path to attenuate the photon collection and generate noisy data for benchmarking. The nonattenuated path is used to generate labels for supervised learning and benchmarking. (b) The averaged readout image and histogram from a 10-ms integration, the shortest readout time in the data set. The solid line represents the two Gaussian fits used in conventional analysis methods. Conventionally, a threshold (shown as a dashed line) is computed that optimally separates the two Gaussian distributions and is used to infer the qubit state. The state-detection fidelity depends entirely on the separability of the two distributions. Note that the histograms pictured are based on an ensemble of images. The threshold is computed based on this ensemble, whereas state detection is performed on the camera counts from a single image of a single atom at a time. The count distribution has a state separation of 4.5σ and 2.1σ from the primary and secondary paths, respectively. Using the Gaussian-threshold method, a state-detection fidelity $>99.8\%$ is maintained in all data sets for the nonattenuated path. The combined attenuation elements result in a secondary-path fluorescence efficiency of 60% of the primary path.

The conventional linear-threshold method is applied to the high-SNR images in order to generate the "ground-truth" labels with $> 99.8\%$ fidelity in all sets. Each data set contains 3000 single shots of the 3×3 array, with approximately 50% occupancy for each readout.

We do not expect the small labeling error in the data set to create an upper bound on the achievable fidelity. CNNs and DNNs are known to be robust against small amounts of unbiased mislabeled data [28,29]. The fraction of labeling errors in our data set ($< 0.2\%$) is small enough to not impose an upper limit on the achievable fidelity with a supervised-learning method. For a system with lower readout fidelity such that there is a significant amount of mislabeled data, conventional training methods can be modified with a transfer-learning method [30,31], semisupervised learning [32] and unsupervised learning [33].

Each data set is then randomly divided into three subsets with the distribution 60% training, 20% validation, and 20% test. The training subset is used as input for all iterations of CNN training. The validation subset is used between training epochs to monitor the performance of the partially trained CNN and detect overfitting. The test subset is reserved for final performance evaluation, to ensure that the CNNs are benchmarked on data to which the CNNs have no prior access. For specifics about the CNN training procedure, see Appendix A. All quoted fidelities have been evaluated on the test subset.

IV. RESULTS

In this section, we present the CNN classification performance compared to conventional methods. We first consider the performance under low-crosstalk conditions (9-μm array spacing) and then we examine the case in which there is increased crosstalk (5-μm array spacing). Crosstalk occurs when a percentage of photons emitted by an atom strike the EMCCD array at the location corresponding to a different site in the atomic array. This increases the photon count for the "wrong" atom, making correct state detection more challenging.

Performance is measured in terms of classification fidelity [20], defined as

$$\mathcal{F} = 1 - \frac{P(B_p|D) + P(D_p|B)}{2}, \tag{1}$$

where D and B are the true dark and bright qubit states, respectively, and D_p and B_p are the predicted dark and bright states. $P(B_p|D)$ and $P(D_p|B)$ represent the false-bright and false-dark prediction probabilities, respectively. To quantify the prediction correlation between two sites due to crosstalk, we define the cross-fidelity [34] as

$$\mathcal{F}^{\mathrm{CF}}_{ij} = 1 - \langle P(D_i|B_j) + P(B_i|D_j) \rangle , \tag{2}$$

where D_i and B_i are the dark and bright states of qubit i and D_j and B_j are the dark and bright states of qubit j. In the absence of crosstalk, the state of qubit i is independent of qubit j and the probability of qubit i being in the bright state given that qubit j is in the dark state, $P(D_i|B_j)$, is 0.5. This yields $\mathcal{F}^{\mathrm{CF}}_{ij} = 0$ for the crosstalk-free case. Positive (negative) $\mathcal{F}^{\mathrm{CF}}_{ij}$ values indicate correlation (anticorrelation) between sites i and j.

We define the relative infidelity factor η in terms of CNN infidelity $(1 - \mathcal{F}_{\mathrm{CNN}})$ and the Gaussian-mask infidelity

$(1 - \mathcal{F}_\sigma)$ as

$$\eta = \frac{(1 - \mathcal{F}_\sigma) - (1 - \mathcal{F}_{\mathrm{CNN}})}{1 - \mathcal{F}_\sigma}. \tag{3}$$

This measures the infidelity reduction, or the percent reduction in incorrect state predictions, that the CNN achieves as compared to the conventional method.

In Fig. 4, we show the performance of all four state-detection methods for the 9-µm spacing array as a function of the readout time. At this spacing, the effects of crosstalk are relatively small, which we have confirmed by comparing the cross-fidelity of the neighboring sites versus non-neighboring sites for the square mask in Table III. (We use the square-mask statistics as a reference because that method has no built-in mechanism to mitigate crosstalk.) Both CNN architectures show a reduction in measurement infidelity relative to the conventional methods at readout times up to 90 ms. CNN-site achieves up to 56% reduction in infidelity (vertical dotted line), or up to 29% reduction in readout time (horizontal dotted line) when compared to the Gaussian method. CNN-array achieves up to 35% reduction in infidelity and 14% reduction in readout time. This demonstrates that the CNNs outperform the conventional method even in a low-crosstalk configuration. The superior performance of CNN-site, despite its low parameter count, may indicate that it is better optimized for this particular task than CNN-array. In this crosstalk-free configuration, the large number of trainable parameters in CNN-array does make it more susceptible to overfitting.

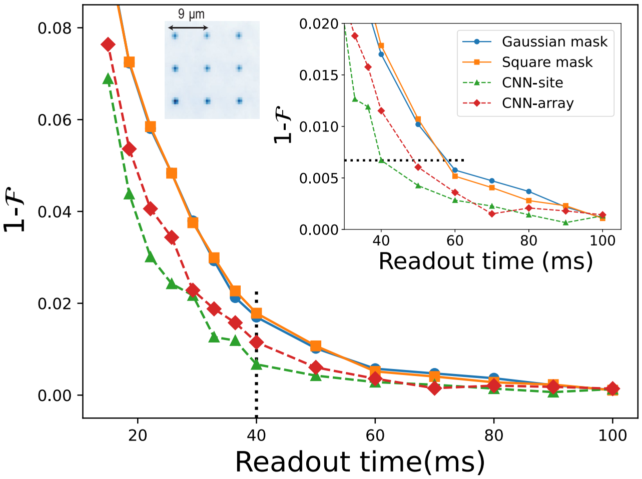


FIG. 4. The measurement infidelity from different analysis methods at 9-µm site spacing. The greatest error reduction between CNN-site and the Gaussian method is observed at 40 ms, marked with a vertical dashed line. The inset shows the details at a readout time above 30 ms. The horizontal dashed line presents the biggest reduction of readout time on CNN-site while maintaining the same fidelity compared to the Gaussian method.

TABLE III. The cross-fidelities at 40 ms readout time in the 9 µm spacing array. The cross-fidelity between the central site and the nearest neighbor sites (C-NN) is given in rows 1–5. The cross-fidelity quantifies the predicted correlation between the central site (site 5) and its nearest neighbors due to crosstalk. The cross-fidelity between the edge sites (E-E) is given in rows 6–12. The edge sites are separated far enough such that there is negligible crosstalk. The averaged values for C-NN and E-E are given in rows 5 and 12. The C-NN cross-fidelities are comparable to the E-E cross-fidelities suggesting there is no observable crosstalk at this separation in this configuration.

	C-NN	Gaussian	Square	CNN-site	CNN-array
1	$\mathcal{F}^{\mathrm{CF}}_{52}$	0.0218	0.0072	0.0048	0.0043
2	$\mathcal{F}^{\mathrm{CF}}_{54}$	0.0300	0.0267	0.0236	0.0238
3	$\mathcal{F}^{\mathrm{CF}}_{56}$	0.0075	0.0025	0.0090	0.0048
4	$\mathcal{F}^{\mathrm{CF}}_{58}$	0.0065	0.0186	0.0046	0.0095
5	$\langle \lvert \boldsymbol{\mathcal{F}^{\mathrm{CF}}_{5j}} \rvert \rangle$	**0.0165**	**0.0138**	**0.0060**	**0.0082**
	E-E	Gaussian	Square	CNN-site	CNN-array
6	$\mathcal{F}^{\mathrm{CF}}_{13}$	−0.0086	−0.0131	−0.0059	−0.180
7	$\mathcal{F}^{\mathrm{CF}}_{79}$	−0.0134	−0.0172	−0.0108	−0.0118
8	$\mathcal{F}^{\mathrm{CF}}_{17}$	0.0024	0.0151	−0.0004	0.0069
9	$\mathcal{F}^{\mathrm{CF}}_{39}$	0.0095	0.0040	0.0007	0.0004
10	$\mathcal{F}^{\mathrm{CF}}_{19}$	−0.0098	−0.0108	−0.0163	−0.0160
11	$\mathcal{F}^{\mathrm{CF}}_{37}$	−0.0030	−0.0006	0.0007	−0.0055
12	$\langle \lvert \boldsymbol{\mathcal{F}^{\mathrm{CF}}_{ij}} \rvert \rangle$	**0.0101**	**0.0078**	**0.0058**	**0.0096**

We also observe no significant differences in performance between the square threshold and the Gaussian threshold.

At 5-µm separation, the effects of crosstalk become apparent. This is evident if we compare the square-mask cross-fidelities in Table IV to those in Table III. While the edge-to-edge baseline values are similar, the average nearest-neighbor cross-fidelity increases from 0.0138 to 0.0428 when we decrease the spacing from 9 µm to 5 µm.

As anticipated, the crosstalk significantly impacts state-detection performance and further differentiates the four methods. In Fig. 5(a), we show the performance of all four state-detection methods for the 5-µm-spaced array. CNN-array achieves an infidelity reduction of up to 43% (vertical dotted line) and a readout-time reduction up to 25% (horizontal dotted line) compared to the Gaussian mask. CNN-site achieves a more modest 32% infidelity reduction and 20% readout-time reduction. Unlike the 9-µm case, the Gaussian mask performs significantly better than the square mask.

Analyzing the fidelity of individual sites in Fig. 5(b), we observe higher infidelity on the central site when compared to the four corner sites for all methods. The corresponding cross-fidelities in Table IV support this observation, showing a positive correlation between the central site and its four nearest neighbors. CNN-array is the most

TABLE IV. The cross-fidelities at 36 ms readout time in the 5 μm spacing array. The cross-fidelity between the central site and the nearest neighbor sites (C-NN) is given in rows 1–5. The cross-fidelity quantifies the predicted correlation between the central site (site 5) and its nearest neighbors due to crosstalk. The cross-fidelity between the edge sites (E-E) is given in rows 6–12. The edge sites are separated far enough such that there is negligible crosstalk. The averaged values for C-NN and E-E are given in rows 5 and 12.

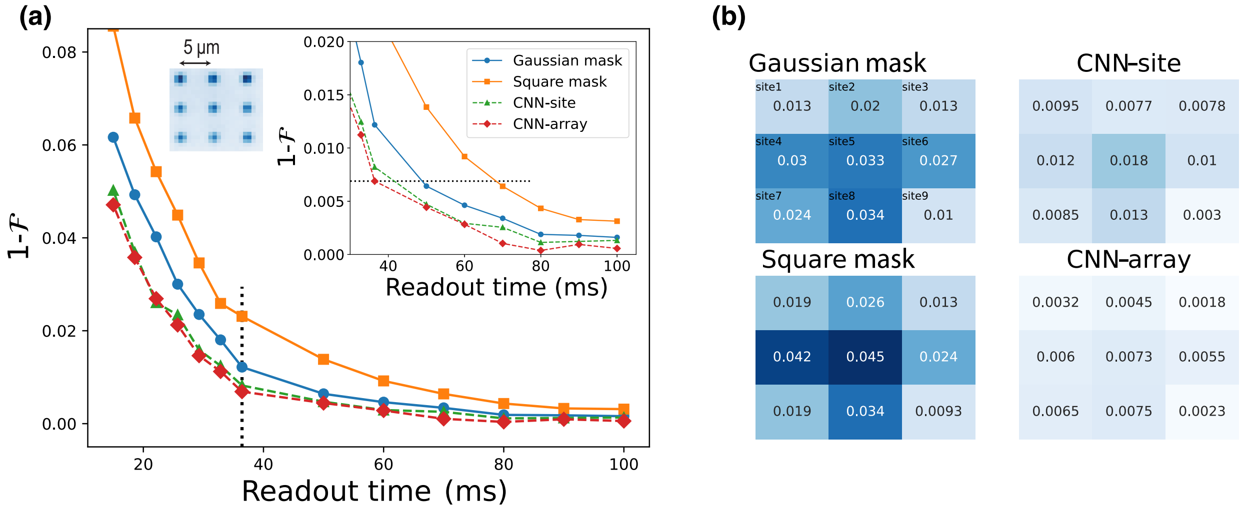
	C-NN	Gaussian	Square	CNN-site	CNN-array
1	$\mathcal{F}^{\mathrm{CF}}_{52}$	0.0298	0.0524	0.0151	0.0099
2	$\mathcal{F}^{\mathrm{CF}}_{54}$	0.0211	0.0298	0.0146	0.0099
3	$\mathcal{F}^{\mathrm{CF}}_{56}$	0.0218	0.0284	0.0120	0.0022
4	$\mathcal{F}^{\mathrm{CF}}_{58}$	0.0392	0.0607	0.0120	0.0020
5	$\langle\lvert\boldsymbol{\mathcal{F}}^{\mathbf{CF}}_{\mathbf{5}j}\rvert\rangle$	**0.0280**	**0.0428**	**0.0134**	**0.0060**
	E-E	Gaussian	Square	CNN-site	CNN-array
6	$\mathcal{F}^{\mathrm{CF}}_{13}$	0.0110	0.0139	0.0205	0.0110
7	$\mathcal{F}^{\mathrm{CF}}_{79}$	−0.0036	−0.0005	−0.0094	−0.0036
8	$\mathcal{F}^{\mathrm{CF}}_{17}$	0.0038	0.0091	−0.0016	0.0038
9	$\mathcal{F}^{\mathrm{CF}}_{39}$	0.0028	0.0090	0.0031	0.0028
10	$\mathcal{F}^{\mathrm{CF}}_{19}$	−0.0068	−0.0043	−0.0098	−0.0068
11	$\mathcal{F}^{\mathrm{CF}}_{37}$	0.0070	0.0078	0.0080	0.0070
12	$\langle\lvert\boldsymbol{\mathcal{F}}^{\mathbf{CF}}_{ij}\rvert\rangle$	**0.0058**	**0.0074**	**0.0087**	**0.0058**

effective at mitigating crosstalk, showing almost no difference between the nearest-neighbor and edge-to-edge cross-fidelities (0.0060 and 0.0058, respectively). That corresponds to a 78% cross-fidelity reduction compared to the Gaussian mask. We attribute the superior performance of CNN-array to its simultaneous awareness of all sites in the array. Unexpectedly, CNN-site is also successful at mitigating some of the crosstalk, achieving a 50% cross-fidelity improvement. In CNN-site, we attribute this improvement to the spatially aware utilization of features on the edges of the (10 × 10)-pixel input to detect subtle correlations.

To better understand the effect of crosstalk on state-detection performance, we compare in Fig. 6 the histograms of the central site with and without neighbors for the Gaussian- and square-threshold methods. When neighboring sites are occupied, we observe broadening and upward biasing of the distributions due to bleed-over of fluorescence from neighboring sites, making the two states more difficult to separate.

Finally, in order to gauge the advantages of our method under normal operating conditions, we have applied the CNN detection method to measurements taken on the nonattenuated primary path at 5-μm spacing. We have obtained the training-data and "ground-truth" labels by performing a sequence of three measurements on each randomly loaded atom array, as described in Appendix B. The first and last high-fidelity measurements are 20 ms long, with $> 4.5\sigma$ separation. In Fig. 7, we observe improvement with up to 83% reduction of infidelity. This improvement is consistent with results from the dual-path setup. This makes it possible to reduce the probing time from 15 ms to 9.8 ms while preserving the readout fidelity above 99.5%.

We have also characterized the processing time needed by the CNN compared to the conventional method. On



Gaussian mask

0.013	0.02	0.013
0.03	0.033	0.027
0.024	0.034	0.01

CNN-site

0.0095	0.0077	0.0078
0.012	0.018	0.01
0.0085	0.013	0.003

Square mask

0.019	0.026	0.013
0.042	0.045	0.024
0.019	0.034	0.0093

CNN-array

0.0032	0.0045	0.0018
0.006	0.0073	0.0055
0.0065	0.0075	0.0023

FIG. 5. Results from the 5-μm site-spacing array. (a) The measurement infidelity with different analysis methods at different readout times. The greatest error reduction is observed at 36 ms, marked with a vertical dashed line. The inset shows the details at readout times above 30 ms. The horizontal dashed line presents the biggest reduction of readout time on CNN-array while maintaining the same fidelity compared to the Gaussian method. (b) The individual site infidelity at 36-ms readout time. Higher infidelity is observed around the central site on the array, suggesting the presence of crosstalk from neighboring-site fluorescence.

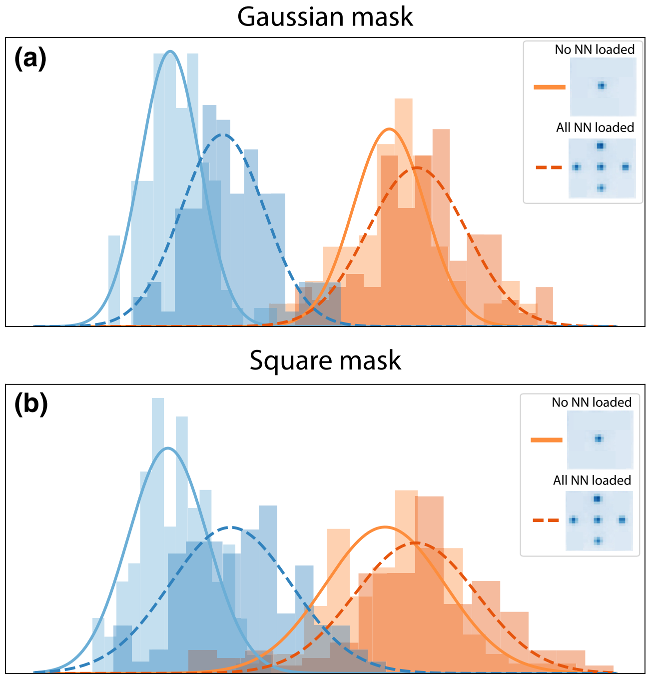


FIG. 6. The shift of the integrated distribution due to the absence and presence of atoms in neighboring sites on (a) the Gaussian mask and (b) the square mask. The overlap between the two states increases substantially in the square-mask method, resulting in poor fidelity.

a personal computer (Ryzen 1700 CPU and GTX1080 GPU), we have found that the average CNN-site and CNN-array single-site inference times are 97 ± 1 μs and 303 ± 23 μs, respectively, compared to the conventional processing time of 11 ± 1 μs. Taking into account the increased inference time together with the reduced probing time, our method reduces the total measurement time by 50%. The processing delay can be further reduced by optimizing the CNN network and using hardware acceleration.

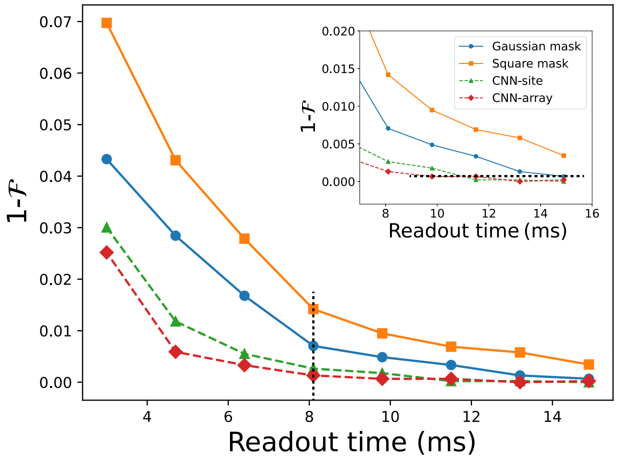


FIG. 7. The measurement results from the primary nonattenuated path for a 3 × 3 array with 5-μm spacing. The greatest fractional error reduction between CNN-site and the Gaussian method is marked with a vertical dashed line. The inset shows the details for readout times above 7 ms. The horizontal dashed line shows that CNN-site at 11.5 ms achieves the same fidelity as the Gaussian method at 15 ms.

V. CONCLUSIONS

We have presented two neural network architectures for state detection of neutral atoms. Both architectures have outperformed conventional state-detection methods by a statistically significant margin. We have observed a 43% reduction in infidelity and 78% cross-fidelity reduction at 5-μm site-to-site spacing with CNN-array, demonstrating a superior ability to mitigate intersite crosstalk. CNN-site has also performed better than conventional methods, with a 32% reduction in infidelity and 50% cross-fidelity improvement. To confirm that improvement is not attributable only to the mitigation of crosstalk, we have also evaluated the performance of the neural network at 9-μm site spacing, where the effects of crosstalk are small. In this configuration, CNN-site has been able to reduce the infidelity up to 57% compared to the Gaussian-mask method. Based on these results, we conclude that a neural network can reduce the readout time by up to 29% while maintaining the same fidelity, opening up the possibility of faster high-fidelity measurements on neutral-atom processors. In future work, we plan to accelerate the CNNs on an FPGA as a proof-of-concept demonstration of fast real-time state detection, which is a necessary ingredient for measurement-based quantum error correction [8].

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers as part of the Q-NEXT center, National Science Foundation (NSF) Award No. 2016136 for the Quantum Leap Challenge Institutes (QLCI) center Hybrid Quantum Architectures and Networks, and NSF Award No. 2210437.

APPENDIX A: METHODS

This appendix describes the step-by-step procedure used to initialize and train the neural networks.

1. Preprocessing

The primary- and secondary-path images are acquired from the camera as raw grayscale images. The primary-path images are fed into the conventional Gaussian-mask analysis method (see Sec. II C) in order to extract the “ground-truth” labels—a binary value for each site in the array, where “1” corresponds to the bright state and “0” to the dark state. This results in a binary array of size

(n_{images}, n_{sites}), encompassing all the labels for the entire data set. Prior to being sent to the CNN for processing, the secondary-path images are first normalized and zero centered: $I_{ij} = (I_{ij} - \mu)/\alpha$, where μ is the average pixel intensity over the entire training subset, and $\alpha = I_{\max} - I_{\min}$, where $I_{\max}$ and $I_{\min}$ are the maximum and minimum pixel intensity in the training subset. The indices i and j refer to the coordinates of each pixel in the image.

2. Image partitioning (only for CNN-site)

The input images for CNN-site are generated by cropping the array image into individual 10 × 10 pixel images at predefined site locations. For each site, the site center is determined by performing a Gaussian fit over the averaged training image. Averaging the training images improves the SNR and ensures that all sites contain the (averaged) image of an atom. Once the pixel coordinates at the center of a site have been identified, a 10 × 10 crop centered on those coordinates is extracted. This is repeated for all sites in the array. All of the extracted site images together form an array of shape (n_{image}, n_{site}, 10, 10), where n_{image} refers to the number of original (uncropped) images. This array is reshaped to a four-dimensional matrix of size ($n_{\mathrm{image}} \times n_{\mathrm{site}}$, 10, 10, 1), which is passed to the CNN as input data (TensorFlow processes the images in batches, not one at a time). A separate array of this type is produced for each of the training, test, and validation subsets. The array of “ground-truth” labels is transformed to a 1D vector of size ($n_{\mathrm{image}} \times n_{\mathrm{site}}$). The CNN does not receive these labels as input; they are used by TensorFlow to evaluate the CNN and manage the training process.

3. Training

The network is trained using the TensorFlow library, via the KERAS API. The network parameters are randomly initialized to begin with, then trained using the preprocessed secondary-path images and corresponding “ground-truth” labels. The hyper-parameters used for training are listed in Table V. The training progress is supervised by monitoring the loss over the validation set at the end of every training epoch and saving a snapshot of the best-so-far network parameters. At the end of the training process, only the set of parameters corresponding to the lowest-achieved validation loss is kept. Validation loss refers to the loss metric evaluated on the validation subset. A lower validation loss corresponds to better performance at the task. The “Maximum epoch” number is chosen to be long enough to avoid underfitting the data. In other words, the training continues until overfitting occurs, such that the training accuracy will approach 100% by the end of the training process, while the validation accuracy begins to drop. Training to the point of overfitting does not harm the final result, because of this monitoring process. Roughly speaking, the optimal network parameters are achieved when the training accuracy (loss) and validation accuracy (loss) diverge, as seen in Fig. 8. This process of monitoring and selection ensures that the network parameters are a sufficiently good fit to the classification task and that the resulting network still generalizes well to data that it has not seen before.

TABLE V. The training hyperparameters. Adam [35] is a stochastic gradient-descent algorithm with adaptive learning rates from estimation of the first and second order of the gradients. The initial learning rate determines the step size of the first parameter updates toward the minimum of a loss function. The learning rate for the further iterations are dynamically determined by the Adam optimizer. “Maximum epoch” is the number of complete passes through the training data set during training. The batch size is the number of samples passed before the network parameters are updated during training.

Parameter	CNN-site	CNN-array
Optimizer	Adam	Adam
Initial learning rate	1×10^{-4}	5×10^{-4}
Maximum epoch	40	30
Batch size	64	16

Note that the CNN needs to be retrained from scratch for every new data set acquired. For example, CNN-site has a completely different set of training parameters for

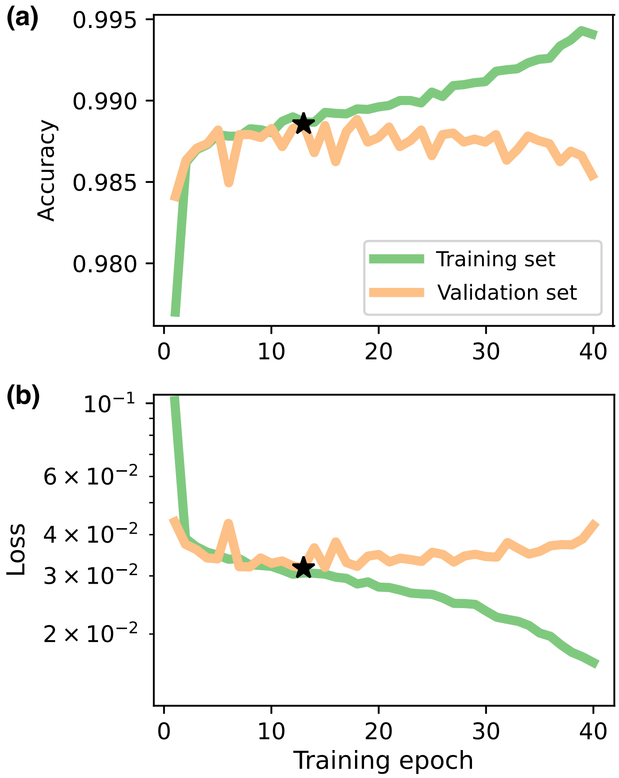


FIG. 8. The (a) accuracy and (b) loss over the training epoch. The optimum parameter set is extracted from the epoch with the lowest validation loss, marked in the figure as ★.

the “9-μm 40-ms readout time” data set as compared to the “9-μm 20-ms readout time” data set or the “5-μm 20-ms readout time” data set. We have found that it is not difficult to modify the CNN architecture in such a way (namely, by incorporating an “embedding” layer and providing a data-set identifier as an additional input) that a single parameter set can handle all data sets equally, without significant loss of accuracy. This could be an important enhancement for real-time applications in which it is desirable to change the parameters of the experiment without acquiring a new data set and retraining the CNN. We consider this to be a topic for future research.

APPENDIX B: ALTERNATIVE DATA-SET GENERATION METHOD

For a system with only a single imaging path, a training data set can be generated by performing a sequence of A-B-A measurements on each randomly loaded array, where A is a high-fidelity measurement and B is a noisy measurement. The data are then postselected to keep only the A-B-A sequences in which the first and last A measurements show identical array occupation (no atom loss), according to the conventional analysis method. Postselection discards any data that are mislabeled due to atom loss during or after the first measurement. The postselected A measurements are then used to generate the “ground-truth” labels, while the corresponding B measurements constitute the inputs to the CNN.

[1] R. Acharya, *et al.*, Suppressing quantum errors by scaling a surface code logical qubit, Nature **614**, 676 (2023).

[2] B. Nikolov, E. Diamond-Hitchcock, J. Bass, N. L. R. Spong, and J. D. Pritchard, Randomized benchmarking using nondestructive readout in a two-dimensional atom array, Phys. Rev. Lett. **131**, 030602 (2023).

[3] S. J. Evered, D. Bluvstein, M. Kalinowski, S. Ebadi, T. Manovitz, H. Zhou, S. H. Li, A. A. Geim, T. T. Wang, N. Maskara, H. Levine, G. Semeghini, M. Greiner, V. Vuletić, and M. D. Lukin, High-fidelity parallel entangling gates on a neutral-atom quantum computer, Nature **622**, 268 (2023).

[4] T. M. Graham, *et al.*, Multi-qubit entanglement and algorithms on a neutral-atom quantum computer, Nature **604**, 457 (2022).

[5] D. Bluvstein, H. Levine, G. Semeghini, T. T. Wang, S. Ebadi, M. Kalinowski, A. Keesling, N. Maskara, H. Pichler, M. Greiner, V. Vuletić, and M. D. Lukin, A quantum processor based on coherent transport of entangled atom arrays, Nature **604**, 451 (2022).

[6] E. Deist, Y.-H. Lu, J. Ho, M. K. Pasha, J. Zeiher, Z. Yan, and D. M. Stamper-Kurn, Mid-circuit cavity measurement in a neutral atom array, Phys. Rev. Lett. **129**, 203602 (2022).

[7] K. Singh, C. E. Bradley, S. Anand, V. Ramesh, R. White, and H. Bernien, Mid-circuit correction of correlated phase errors using an array of spectator qubits, Science **380**, 1265 (2023).

[8] T. M. Graham, L. Phuttitarn, R. Chinnarasu, Y. Song, C. Poole, K. Jooya, J. Scott, A. Scott, P. Eichler, and M. Saffman, Mid-circuit measurements on a neutral atom quantum processor, Phys. Rev. X **13**, 041051 (2023).

[9] M. A. Norcia, *et al.*, Mid-circuit qubit measurement and rearrangement in a ^{171}Yb atomic array, Phys. Rev. X **13**, 041034 (2023).

[10] S. Ma, G. Liu, P. Peng, B. Zhang, S. Jandura, J. Claes, A. P. Burgers, G. Pupillo, S. Puri, and J. D. Thompson, High-fidelity gates and mid-circuit erasure conversion in an atomic qubit, Nature **622**, 279 (2023).

[11] J. W. Lis, A. Senoo, W. F. McGrew, F. Rönchen, A. Jenkins, and A. M. Kaufman, Mid-circuit operations using the omg-architecture in neutral atom arrays, Phys. Rev. X **13**, 041035 (2023).

[12] T. Xia, M. Lichtman, K. Maller, A. W. Carr, M. J. Piotrowicz, L. Isenhower, and M. Saffman, Randomized benchmarking of single-qubit gates in a 2D array of neutral-atom qubits, Phys. Rev. Lett. **114**, 100503 (2015).

[13] M. Martinez-Dorantes, W. Alt, J. Gallego, S. Ghosh, L. Ratschbacher, Y. Völzke, and D. Meschede, Fast nondestructive parallel readout of neutral atom registers in optical potentials, Phys. Rev. Lett. **119**, 180503 (2017).

[14] A. Syberfeldt and F. Vuoluterä, Image processing based on deep neural networks for detecting quality problems in paper bag production, Procedia CIRP **93**, 1224 (2020). 53rd CIRP Conference on Manufacturing Systems 2020.

[15] A. Impertro, J. F. Wienand, S. Häfele, H. von Raven, S. Hubele, T. Klostermann, C. R. Cabrera, I. Bloch, and M. Aidelsburger, An unsupervised deep learning algorithm for single-site reconstruction in quantum gas microscopes, Commun. Phys. **6**, 166 (2023).

[16] L. R. B. Picard, M. J. Mark, F. Ferlaino, and R. van Bijnen, Deep learning-assisted classification of site-resolved quantum gas microscope images, Meas. Sci. and Technol. **31**, 025201 (2020).

[17] A. Seif, K. A. Landsman, N. M. Linke, C. Figgatt, C. Monroe, and M. Hafezi, Machine learning assisted readout of trapped-ion qubits, J. Phys. B: At. Mol. Opt. Phys. **51**, 174006 (2018).

[18] Z.-H. Ding, J.-M. Cui, Y.-F. Huang, C.-F. Li, T. Tu, and G.-C. Guo, Fast high-fidelity readout of a single trapped-ion qubit via machine-learning methods, Phys. Rev. Appl. **12**, 014038 (2019).

[19] S. Bravyi, S. Sheldon, A. Kandala, D. C. Mckay, and J. M. Gambetta, Mitigating measurement errors in multiqubit experiments, Phys. Rev. A **103**, 042605 (2021).

[20] B. Lienhard, A. Vepsäläinen, L. C. G. Govia, C. R. Hoffer, J. Y. Qiu, D. Ristè, M. Ware, D. Kim, R. Winik, A. Melville, B. Niedzielski, J. Yoder, G. J. Ribeill, T. A. Ohki, H. K. Krovi, T. P. Orlando, S. Gustavsson, and W. D. Oliver, Deep-neural-network discrimination of multiplexed superconducting-qubit states, Phys. Rev. Appl. **17**, 014024 (2022).

[21] J. Darulová, M. Troyer, and M. C. Cassidy, Evaluation of synthetic and experimental training data in supervised machine learning applied to charge-state detection of quantum dots, Mach. Learn.: Sci. Technol. **2**, 045023 (2021).

[22] Y. Matsumoto, T. Fujita, A. Ludwig, A. D. Wieck, K. Komatani, and A. Oiwa, Noise-robust classification of single-shot electron spin readouts using a deep neural network, npj Quantum Inf. **7**, 136 (2021).

[23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, Massachusetts, 2016). www.deeplearningbook.org

[24] F. Chollet, *et. al.*, KERAS, https://github.com/fchollet/keras (2015).

[25] M. Abadi, *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems, (2015), software available from tensorflow.org.

[26] The softmax activation function is defined as $g\left(z_{c'}\right) = \frac{e^{z_{c'}}}{\sum_{d=1}^{C_{\text{out}}} e^{z_d}}$ This function normalizes the sum of the elements such that $\sum_{c'=1}^{C_{\text{out}}} z_{c'} = 1$. Each element value is then the weighted probability.

[27] M. Kwon, M. F. Ebert, T. G. Walker, and M. Saffman, Parallel low-loss measurement of multiple atomic qubits, Phys. Rev. Lett. **119**, 180504 (2017).

[28] F. S. Fard, P. Hollensen, S. Mcilory, and T. Trappenberg, in *2017 International Joint Conference on Neural Networks (IJCNN)* (IEEE, New York, 2017), p. 2652.

[29] D. Rolnick, A. Veit, S. Belongie, and N. Shavit, Deep learning is robust to massive label noise, arXiv:1705.10694.

[30] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, in *2014 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, New York, 2014), p. 1717.

[31] H. Azizpour, A. S. Razavian, J. Sullivan, A. Maki, and S. Carlsson, in *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (IEEE, New York, 2015), p. 36.

[32] J. Weston, F. Ratle, H. Mobahi, and R. Collobert, in *Neural Networks: Tricks of the Trade: Second Edition*, edited by G. Montavon, G. Orr, and K.-R. Müller (Springer-Verlag, Berlin, 2012), p. 639.

[33] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, and T. Brox, in *Proceedings of the 27th International Conference on Neural Information Processing Systems—Volume 1*, NIPS’14, edited by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (MIT Press, Cambridge, Massachusetts, 2014), p. 766.

[34] Z. Chen, *et al.*, Exponential suppression of bit or phase errors with cyclic error correction, Nature **595**, 383 (2021).

[35] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.