Experimental Realization of a Quantum Support Vector Machine

Zhaokai Li,[1,2] Xiaomei Liu,[1] Nanyang Xu,[1,2,*] and Jiangfeng Du[1,2,†]
[1]Hefei National Laboratory for Physical Sciences at the Microscale and Department of Modern Physics, University of Science and Technology of China, Hefei 230026, China
[2]Synergetic Innovation Center of Quantum Information and Quantum Physics, University of Science and Technology of China, Hefei 230026, China
(Received 1 December 2014; revised manuscript received 17 February 2015; published 8 April 2015)

The fundamental principle of artificial intelligence is the ability of machines to learn from previous experience and do future work accordingly. In the age of big data, classical learning machines often require huge computational resources in many practical cases. Quantum machine learning algorithms, on the other hand, could be exponentially faster than their classical counterparts by utilizing quantum parallelism. Here, we demonstrate a quantum machine learning algorithm to implement handwriting recognition on a four-qubit NMR test bench. The quantum machine learns standard character fonts and then recognizes handwritten characters from a set with two candidates. Because of the wide spread importance of artificial intelligence and its tremendous consumption of computational resources, quantum speedup would be extremely attractive against the challenges of big data.

DOI: 10.1103/PhysRevLett.114.140504 PACS numbers: 03.67.Ac, 07.05.Mh, 76.60.-k

As an approach to artificial intelligence (AI), machine learning algorithms have the ability to optimize their future performance by learning from existing samples [1]. When human beings are unable to give explicitly programmed instructions in many practical problems, the machines could reveal the hidden patterns in large data sets and become "intelligent." There are two major types of machine learning, supervised and unsupervised, differing from whether the training data are labeled in advance. Among the supervised machines, the support vector machine (SVM) is a popular learning machine which learns from training data and classifies vectors in a feature space into one of two subgroups [2]. The machine first converts the training data to feature space and find the optimal hyperplane which can divide the two subgroups. Then the machine could use the hyperplane to classify a new unknown instance which subgroup it belongs. The computational complexity of SVM in time is $O[\mathrm{poly}(NM)]$, i.e., polynomial in NM, where N is the number of dimensions of the feature space and M is the number of training samples. To achieve desirable performance, the machines often require a huge amount of storage and computational resources.

Quantum algorithms may be utilized to reduce resource costs, as they have provided impressive speedup over their classical counterparts in many fields [3–6]. In the area of AI, Rebentrost *et al.* recently reported a quantum algorithm for SVM (QSVM) [7] based on recently developed techniques [8–10], which can achieve $O[\log(NM)]$ performance in both training and classifying processes. The exponential speedup is achieved by utilizing a quantum circuit to calculate the inner products of the vectors in parallel [9] and to convert the SVM training to an approximate least-squares problem which is subsequently solved by the quantum matrix inversion algorithm [10,11].

Here, we report an experimental implementation of the QSVM algorithm. To be more comprehensive, we apply the algorithm to a popular problem, the optical character recognition (OCR) problem [12], to classify handwritten characters in a candidate set. Since a realistic OCR problem in the QSVM needs more qubits than those that can be implemented in the state-of-art quantum computing technologies, here, for the purpose of demonstration, we restrict the problem to the minimal case where only two characters ("6" and "9") are in the candidate set and only two features (horizontal and vertical ratios, which are defined in the following) are considered in the problem. This allows us to demonstrate this quantum artificial intelligence algorithm on a four-qubit nuclear spin quantum processor at ambient conditions.

A usual OCR process often contains four stages [12]: (i) preprocessing the image, (ii) dividing the sentence image into character images, (iii) extracting the features of the image, and (iv) classifying the character image. Here, the machine is only designed to recognize a single character image; thus, the second stage for segmentation is not necessary. The quantum SVM here works as follows: first, the machine is trained with the printed images of the standard fonts of the two characters 6 and 9 [Fig. 1(a)], and then, a series of the handwritten images of character 6 or 9 [Fig. 1(b)] are provided to the machine for recognition (classification), i.e., to determine which character group it belongs. It is worth noting that the machine actually works with a vector formed by the features of each image, and before being fed to the machine, every image should be preprocessed, such as resizing the pixels and calculating the

(a)

Training data (printed characters)		label
6	$\vec{x}_1 = (0.987, 0.159)$	$y(\vec{x}_1) = +1$
9	$\vec{x}_2 = (0.354, 0.935)$	$y(\vec{x}_2) = -1$

(b)

Handwritten characters			
6	(0.997, -0.072)	9	(0.338, 0.941)
9	(0.147, 0.989)	6	(0.999, 0.025)
6	(0.999, -0.030)	9	(0.439, 0.899)
6	(0.987, -0.161)	9	(0.173, 0.985)

FIG. 1 (color online). Samples of optical character recognition for SVM. (a) The standard font (Times New Roman) images of characters "6" and "9" for training SVM. $\vec{x}_1$ and $\vec{x}_2$ are the feature vectors extracted from the sample images, which are labeled as the "positive" and "negative" class, respectively $(y = \pm 1)$. (b) The arbitrarily chosen handwritten samples for testing the SVM and their corresponding feature vectors. (See Ref. [13] for detailed derivation).

features. In our case, the feature values are chosen as the vertical ratio and the horizontal ratio, calculated from the pixels in the left (upper) half over the right (lower) half. A linear conversion and a normalization are applied to the feature vectors to make them suitable for our quantum processor with only four qubits (See Ref. [13] for details). After preprocessing, the two printed images of the standard fonts are represented by $\vec{x}_1 = (0.987, 0.159)$ for character 6 and $\vec{x}_2 = (0.354, 0.935)$ for character 9. The vectors of handwritten characters, which have the feature of clustering and are linearly separable, are also provided in Fig. 1.

In this Letter, we will define the images of the character 6 as "positive class" and that of 9 as "negative class," denoted by the variable $y(= \pm 1)$. The task for a SVM algorithm is to classify a new vector $\vec{x}_0$ into one of these two classes, by learning from the training samples which are labeled previously. In the training process, the machine finds a maximum-margin hyperplane which separates the training samples by their classes. Mathematically, we construct the hyperplane with $\vec{w} \cdot \vec{x} + b = 0$ so that $\vec{w} \cdot \vec{x}_i + b \geq 1$ for $\vec{x}_i$ in the positive class, and $\vec{w} \cdot \vec{x}_i + b \leq -1$ for $\vec{x}_i$ in the negative class. The optimization objective is to find the optimal hyperplane which could maximize the distance $2/|\vec{w}|$ between two classes, subject to the constraints $y_i(\vec{w} \cdot \vec{x}_i + b) \geq 1$ for all i.

Once the parameters of the hyperplane have been obtained, the classification process for a new vector $\vec{x}_0$ is to find which side of the hyperplane this vector locates. The classification result for $\vec{x}_0$ is given by

$$y(\vec{x}_0) = \mathbf{sgn}(\vec{w} \cdot \vec{x}_0 + b). \tag{1}$$

If $y(\vec{x}_0) = +1$, we will classify the vector $\vec{x}_0$ as positive class, which means the related handwritten character is recognized as 6. For the vectors $\vec{x}_0$ which makes $y(\vec{x}_0) = -1$, the recognition result is 9.

In the SVM algorithm, the normal vector $\vec{w}$ is represented by $\vec{w} = \sum_{i=1}^{M} \alpha_i \vec{x}_i$, where α_i is the weight for the ith training vector $\vec{x}_i$. Thus, in this formation, α_i and b are the hyperplane parameters to be optimized. In the least-squares approximation of SVM [14], these parameters can be obtained by solving a linear equation $\tilde{F}(b, \alpha_1, \alpha_2, \ldots, \alpha_M)^T = (0, y_1, y_2, \ldots, y_M)^T$. Here, $\tilde{F}$ is a $(M+1) \times (M+1)$ matrix with the essential part as the kernel matrix K (see note [15] for the explicit form). In this Letter, we adopt the linear kernels $K_{i,j} = k(\vec{x}_i, \vec{x}_j) = \vec{x}_i \cdot \vec{x}_j$.

To access the training vectors quantum mechanically, it is assumed that we have the training-data oracles that could return the state $|\vec{x}_i\rangle = 1/|\vec{x}_i| \sum_{j=1}^{N} (\vec{x}_i)_j |j\rangle$, which is the quantum counterpart of the training data $\vec{x}_i$ [7]. Starting from the initial state $(1/\sqrt{M}) \sum_{i=1}^{M} |i\rangle$, the training-data oracles are used to prepare the state $|\chi\rangle = (1/\sqrt{N_\chi}) \sum_{i=1}^{M} |\vec{x}_i||i\rangle|\vec{x}_i\rangle$, with $N_\chi = \sum_{i=1}^{M} |\vec{x}_i|^2$. After discarding the training set register, the quantum density matrix is reduced to the kernel matrix K up to a constant factor $\mathbf{tr}K$.

The second step is to optimize the hyperplane parameters b and α_i. The quantum algorithm for solving linear equations has been proposed [10] and experimentally realized [16–18] with exponential speedup. Using the same method, the hyperplane parameters are determined by the matrix inversion: $(b, \vec{\alpha}^T)^T = \tilde{F}^{-1}(0, \vec{y}^T)^T$. The quantum register is first initialized into $|0, \vec{y}\rangle = (1/\sqrt{N_{0,y}})(|0\rangle + \sum_{i=1}^{M} y_i |i\rangle)$. Then, by performing the matrix inversion of F, the quantum state is transferred to

$$|b, \vec{\alpha}\rangle = \frac{1}{\sqrt{N_{b,\alpha}}} \left(b|0\rangle + \sum_{k=1}^{M} \alpha_i |i\rangle \right). \tag{2}$$

Here, $N_{b,\alpha}$ and $N_{0,y}$ are normalization factors. Note that, since the matrix inversion is successful only when the ancillary qubit is on $|1\rangle$ [10], the following operation on $|b, \vec{\alpha}\rangle$ should be a conditional operation (Fig. 2).

With the optimized parameters b and α_i, the classification results $y(\vec{x}_0)$ in Eq. (1) can be represented by

$$y(\vec{x}_0) = \mathbf{sgn}\left(\sum_{i=1}^{M} \alpha_i (\vec{x}_i \cdot \vec{x}_0) + b \right). \tag{3}$$

This could be reproduced by the overlap of the two quantum states: $y(\vec{x}_0) = \mathbf{sgn}(\langle \tilde{\nu}_0 | \tilde{u} \rangle)$, with the training-data state $|\tilde{u}\rangle = (1/\sqrt{N_{\tilde{u}}})[b|0\rangle|0\rangle + \sum_{i=1}^{M} \mathrm{abs}(\vec{x}_i) \alpha_i |i\rangle|\vec{x}_i\rangle]$ and the query state $|\tilde{\nu}_0\rangle = (1/\sqrt{N_{\tilde{x}_0}})[|0\rangle|0\rangle + \sum_{i=1}^{M} \mathrm{abs}(\vec{x}_0)|i\rangle|\vec{x}_0\rangle]$ [7]. Here, the training-data state $|\tilde{u}\rangle$ could be easily obtained by calling the

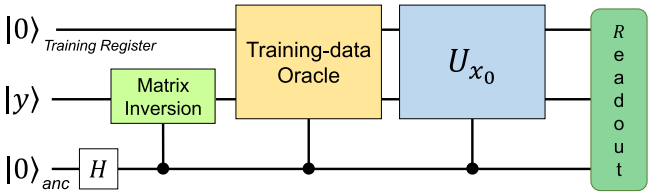


FIG. 2 (color online). The schematic diagram of the quantum SVM. H is the Hadamard gate. The matrix inversion is employed to acquire the hyperplane parameters. Then, the training-data oracle is applied to prepare the training-data state $|\tilde{u}\rangle$. The information of vector $\vec{x}_0$ is introduced by U_{x_0} and then used for classification. The auxiliary phase estimation register for matrix inversion is not shown.

training-data oracle on $|b, \vec{\alpha}\rangle$ and an additional training register (Fig. 3).

To introduce the information of the query vector $\vec{x}_0$, a unitary inverse operation U_{x_0} is applied to transfer $|\tilde{\nu}_0\rangle$ to $|00\rangle$, i.e., $U_{x_0}|\tilde{\nu}_0\rangle = |00\rangle$. Then, the expansion coefficients $\langle 00|U_{x_0}|\tilde{u}\rangle = \langle\tilde{\nu}_0||\tilde{u}\rangle$ will produce the classification result $y(\vec{x}_0)$ [19]. It is noted that the unitary operations are conditional operations controlled by an ancillary qubit. Hence, the final state will be $|\psi\rangle = |\phi\rangle|1\rangle_A + |00\rangle|0\rangle_A$, with $|\phi\rangle = U_{x_0}|\tilde{u}\rangle$, and $|0\rangle_A$ and $|1\rangle_A$ denoting the states of the ancillary qubit. By measuring the expectation value of the coherent term $O \equiv |00\rangle\langle 00| \otimes (|1\rangle\langle 0|)_A$, the classification result will be revealed by

$$y(\vec{x}_0) = \mathbf{sgn}(\langle 00|\phi\rangle) = \mathbf{sgn}(\langle\psi|O|\psi\rangle). \quad (4)$$

If the expectation value is greater than zero, the classification result will be positive $[y(\vec{x}_0) = +1]$; otherwise, it will be negative $[y(\vec{x}_0) = -1]$.

Now we turn to the experimental implementation of the quantum algorithm for OCR. The widely utilized nuclear magnetic resonance (NMR) technique [20–24] is employed to realize the four-qubit quantum processor. The experiments are carried out on a Bruker AV-400 spectrometer at 306 K. The sample used is the ^{13}C-iodotrifluroethylene (C_2F_3I) dissolved in d-chloroform, which consists of a ^{13}C nuclear spin and three ^{19}F nuclear spins. The natural Hamiltonian of this weakly coupled spin system in the rotating frame is

$$H_{\mathrm{NMR}} = \sum_{j=1}^{4} \pi\nu_j\sigma_z^j + \sum_{1\le j<k\le 4} \frac{\pi}{2} J_{jk}\sigma_z^j\sigma_z^k, \quad (5)$$

with the parameters shown in Fig. 3(a). The deviation density matrix of the thermal equilibrium state is $\rho_{\mathrm{eq}} = \sum_{i=1}^{4} \gamma_i\sigma_z^i$, where γ_i represents the gyromagnetic ratio of each nuclear spin.

The experimental procedure consists of three parts: (i) the pseudopure state (PPS) preparation [25], (ii) building the kernel matrix K, and (iii) machine learning and classification. Starting from the thermal state ρ_{eq}, the

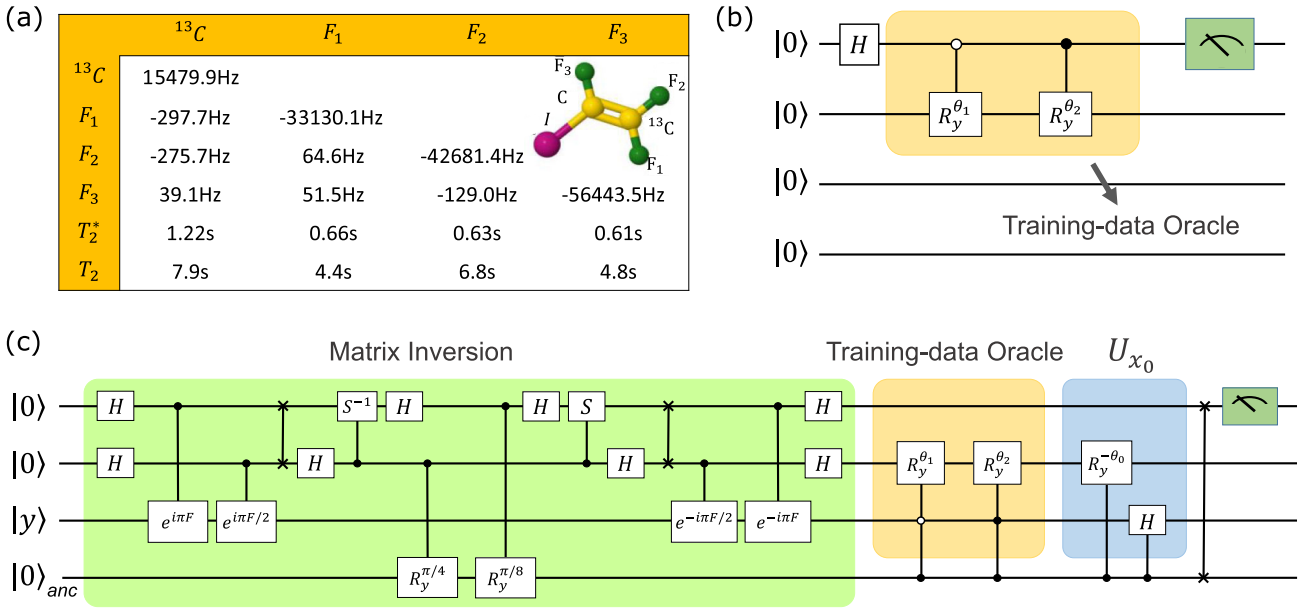


FIG. 3 (color online). (a) Properties of the ^{13}C-iodotrifluroethylene. The chemical shifts ν_i and scalar coupling constants (J_{jk}) are on and below the diagonal of the table, respectively. The chemical shifts are given with respect to the reference frequencies of 100.62 MHz (Carbon) and 376.48 MHz (Fluorines). (b) The quantum circuit for building the kernel matrix K. The rotation angles $\theta_i = \mathrm{arccot}[(x_i)_1/(x_i)_2], i = 0, 1, 2$. After discarding the training-data register (the second qubit), the desired kernel matrix K is obtained as the quantum density matrix of the first qubit. (c) The quantum circuit for classification. H and S are the Hadamard and phase gates, respectively.

line-selective method is employed to prepare the PPS [26]: $\rho_i = (1-\varepsilon/16)\mathbf{I}_{16} + \varepsilon|0000\rangle\langle 0000|$. Here, $\mathbf{I}_{16}$ represents the 16×16 identity operator and $\varepsilon \approx 10^{-5}$ is the polarization.

The quantum circuit of the second part is shown in Fig. 3. Here, the first qubit (^{13}C) serves as the probe qubit, and the second qubit (training set register) is encoded with the information of the two standard sample of characters 6 and 9. The training-data oracle is realized by using two controlled rotations here. After that, the density matrix of the probe qubit will be the kernel matrix K up to a constant factor $\mathbf{tr}K$. We utilized the quantum state tomography technique [27] to measure the density matrix $K/\mathbf{tr}K$ as

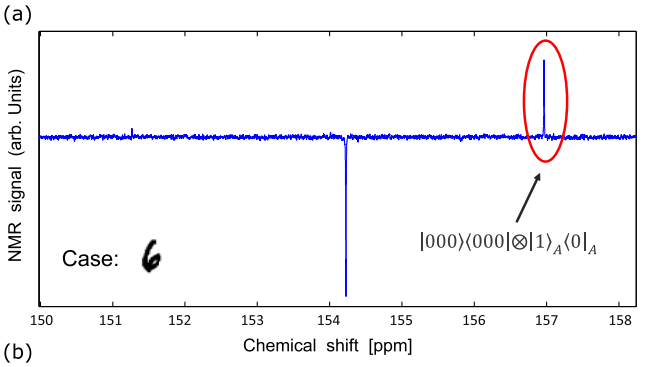
$$\begin{pmatrix} 0.5065 & 0.2425 \\ 0.2425 & 0.4935 \end{pmatrix},$$

which is used to perform a matrix inversion in the next step. It is noted that this process can be avoided by adopting the technique of Lloyd *et al.* [8].

The third part of the experiment is classification. The labels of training samples y_i are encoded into the quantum register. In this implementation, we make the nonoffset reduction, i.e., $b = 0$. Then, the linear equations to be solved are reduced to $F\vec{\alpha} \equiv (K + \gamma^{-1}\mathbf{I}_2)\vec{\alpha} = \vec{y}$, where $\mathbf{I}_2$ is the 2×2 identity matrix and $\gamma = 2$ is a user-specified weight. Here, by using a rotation along the y axis, i.e., $R_y^{-\pi/4} = e^{i\pi\sigma_y/4}$, the system register (the third qubit) is prepared into $|y\rangle = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}/\sqrt{2}$. Then, the first two qubits are utilized here as the phase estimation register to realize the matrix inversion [10,11] that generates the state $F^{-1}|y\rangle$ on the third qubit. In this stage, the weights α_i of the support vectors have been solved and stored into the system register, when the forth qubit is on state $|1\rangle$. By calling the training-data oracle again, the weights are encoded to the coefficient of related support vectors $|x_i\rangle$, leading to the preparation of the training-data state $|\tilde{u}\rangle$. Then the inverse operation U_{x_0}, which relates to the query state x_0, is applied. It is noted that these operations after matrix inversion F^{-1} should be conditional rotation, as shown in Fig. 2. In experiment, we pack the circuit for matrix inversion into one shaped pulse optimized by gradient ascent pulse engineering (GRAPE) method [28], with the length of each pulse being 20 ms and the number of segments being 1000. The remaining part of the circuit is packed into another GRAPE pulse of 25 ms and 2000 segments. All the pulses have theoretical fidelities over 99% and are designed to be robust against the inhomogeneity of radio-frequency pulses.

Finally, the classification result is read out through the expectation value of the coherent term of the ancillary qubit $|000\rangle\langle 000| \otimes |1\rangle_A\langle 0|_A$. The expression is slightly different from Eq. (4), where the auxiliary registers for matrix inversion are not shown. In the experiment, the information of the ancillary qubit is transferred to the ^{13}C spin by a SWAP gate and then is read out through the ^{13}C spectrum [29]. If the corresponding peak in the ^{13}C spectrum is upward [e.g., as in Fig. 4(a)], the classification result will be positive, which means the character recognition result is 6. On the other hand, when the peak is downward, the classification result will be 9. In the experiment, we performed eight different recognition tasks, corresponding to the eight handwritten characters in Fig. 1. The results are shown in Fig. 4(b), which are in accord with human observation. The whole duration of the quantum algorithm is less than 50 ms, and hence, the decay caused by the relaxation is negligible.

In the experiment, the main sources of errors come from the imperfection of pulse sequence and the statistic fluctuation of the signal strength in NMR spectra. The inhomogeneity of magnetic fields is mostly diminished owing to the robustness of the GRAPE pulses. In the first step, the measured value of the kernel matrix ($K/\mathbf{tr}K$) has a fluctuation of 0.01, which leads to an experimental fidelity of 0.99, compared to the exact solution. In the classification part, the imperfection of the kernel matrix leads to a deviation of around 0.01 in the classification result. The pulses used have theoretical fidelities over 99%, leading to



(b)

Hand-written characters	6	9	6	6	9	6	9	9
Experimental indicators								
Amplitude	0.2234	-0.2247	0.2205	0.2496	-0.1775	0.2092	-0.1421	-0.2278
Recognition results	6	9	6	6	9	6	9	9

FIG. 4 (color online). (a) The recognition results given by the quantum OCR. The results are indicated by the orientation of the labeled peak in the ^{13}C spectrum. The upward peak in this spectrum represents a positive result, which classifies the incoming handwritten character as 6. (b) The recognition results corresponding to the handwritten characters in Fig. 1(b). Rows 1–4 represent the handwritten characters, experimental indicators, amplitude of the coherent term, and recognition results, respectively.

a fluctuation of around 0.02. The error originated from the statistic fluctuation of around 0.01 also contributes to the errors. The total experimental error does not exceed 0.04 and has no impact on the recognition results (positive or negative).

To conclude, we have demonstrated the quantum learning algorithm for artificial intelligence on a four-qubit quantum processor, i.e., a quantum learning machine. As an example, we utilized the quantum support vector machine to solve a minimal optical character recognition problem. In the experiment, the quantum machine was trained with the standard fonts of the characters 6 and 9, and then was used to classify new-coming handwritten characters. The successful classification shows the ability of our quantum machine to learn and work, which is an elementary requirement of artificial intelligence. This work sheds light on the bright future of the age of big data by making use of exponential speedup provided by quantum theory. If the quantum processor had the capacity for handling significantly more qubits (60–300 qubits), it would be able to learn the entire (classical) information generated each year and even that of the universe [9].

This work was supported by the National Key Basic Research Program of China (Grant No. 2013CB921800), the National Natural Science Foundation of China (Grants No. 11227901, No. 91021005, and No. 61376128), the Strategic Priority Research Program (B) of the CAS (Grant No. XDB01030400) and the Fundamental Research Funds for the Central Universities.

*Corresponding author.
nyxu@ustc.edu.cn
†Corresponding author.
djf@ustc.edu.cn

[1] E. Alpaydin, *Introduction to Machine Learning* (MIT Press, Cambridge, MA, 2010), p. 584.
[2] C. Cortes and V. Vapnik, Mach. Learn. **20**, 273 (1995).
[3] P. Shor, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science* (IEEE, New York, 1994), p. 124.
[4] L. K. Grover, Phys. Rev. Lett. **79**, 325 (1997).
[5] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2000).
[6] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. OBrien, Nature (London) **464**, 45 (2010).
[7] P. Rebentrost, M. Mohseni, and S. Lloyd, Phys. Rev. Lett. **113**, 130503 (2014).
[8] S. Lloyd, M. Mohseni, and P. Rebentrost, Nat. Phys. **10**, 631 (2014).
[9] S. Lloyd, M. Mohseni, and P. Rebentrost, arXiv:1307.0411.
[10] A. W. Harrow, A. Hassidim, and S. Lloyd, Phys. Rev. Lett. **103**, 150502 (2009).
[11] N. Wiebe, D. Braun, and S. Lloyd, Phys. Rev. Lett. **109**, 050505 (2012).
[12] O. Due Trier, A. K. Jain, and T. Taxt, Pattern Recognition **29**, 641 (1996).
[13] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.114.140504 for the details of the image preprocessing and the database of handwritten characters.
[14] J. A. K. Suykens and J. Vandewalle, Neural Processing Letters **9**, 293 (1999).
[15] $\tilde{F}(1,1)=0$; for $i,j=1,\ldots,M$, $\tilde{F}(1,i+1)=\tilde{F}(i+1,1)=1$, $\tilde{F}(i+1,j+1)=F(i,j)$, with $F\equiv(K+\gamma^{-1}\mathbf{I}_M)$. Here, K is the kernel matrix, and I_M is $M\times M$ identity matrix. The user-specified weight γ determines the relative weight of the training error and the SVM objective [14].
[16] X.-D. Cai, C. Weedbrook, Z.-E. Su, M. C. Chen, M. Gu, M.-J. Zhu, L. Li, N.-L. Liu, C.-Y. Lu, and J.-W. Pan, Phys. Rev. Lett. **110**, 230501 (2013).
[17] J. Pan, Y. Cao, X. Yao, Z. Li, C. Ju, H. Chen, X. Peng, S. Kais, and J. Du, Phys. Rev. A **89**, 022313 (2014).
[18] S. Barz, I. Kassal, M. Ringbauer, Y. O. Lipp, B. Dakic, A. Aspuru-Guzik, and P. Walther, Sci. Rep. **4**, 6115 (2014).
[19] In the case of the inverse operation, U_{x_0} is inaccessible, the classification results could also be obtained by using a swap test [7].
[20] I. L. Chuang, L. M. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, Nature (London) **393**, 143 (1998).
[21] J. A. Jones, M. Mosca, and R. H. Hansen, Nature (London) **393**, 344 (1998).
[22] J. F. Du, N. Y. Xu, X. H. Peng, P. F. Wang, S. F. Wu, and D. W. Lu, Phys. Rev. Lett. **104**, 030502 (2010).
[23] D. Lu, N. Xu, R. Xu, H. Chen, J. Gong, X. Peng, and J. Du, Phys. Rev. Lett. **107**, 020501 (2011).
[24] Z. Li, H. Zhou, C. Ju, H. Chen, W. Zheng, D. Lu, X. Rong, C. Duan, X. Peng, and J. Du, Phys. Rev. Lett. **112**, 220501 (2014).
[25] N. A. Gershenfeld and I. L. Chuang, Science **275**, 350 (1997).
[26] X. Peng, X. Zhu, X. Fang, M. Feng, K. Gao, X. Yang, and M. Liu, Chem. Phys. Lett. **340**, 509 (2001).
[27] J.-S. Lee, Phys. Lett. A **305**, 349 (2002).
[28] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbruggen, and S. J. Glaser, J. Magn. Reson. **172**, 296 (2005).
[29] X. Peng, Z. Luo, W. Zheng, S. Kou, D. Suter, and J. Du, Phys. Rev. Lett. **113**, 080404 (2014).