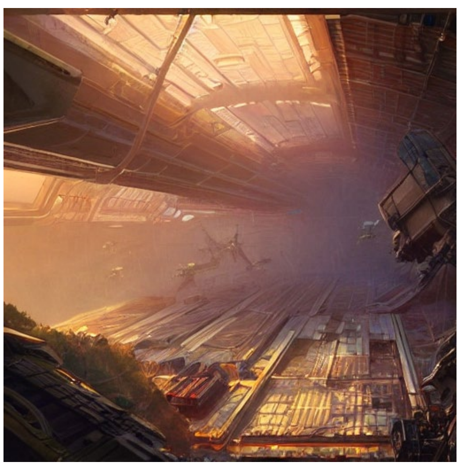
Image Classification with Quantum Machine Learning

Daiwei Zhu and Jason Iaconis

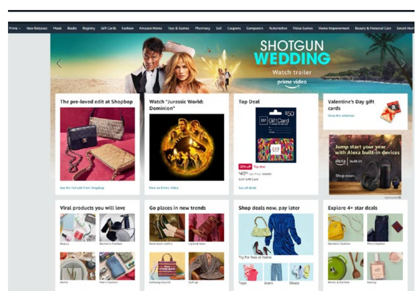
Introduction:

Prof. Feynman's spaceship has crashed on an unknown planet. Many things exploded during the accident, including his suitcase. Oh no—his favorite T-shirts were in the suitcase!

Luckily Prof. Feynman's spacecraft is a cargo craft, which carries in the cargo bay a whole mountain of items including an Xbox, guitar, dry-aged A5 Wagyu ribeye, California-king-sized bed, and of course, t-shirts. (Sounds more like an exotic holiday, right?)



It's not to say Prof. Feynman doesn't still have a problem. Look at the size of the cargo bay. When the cargo bay AI system was working, retrieving an item was super easy—all he needed to do was search via the AI's search interface:



But now the whole AI system is gone with the crash, Prof. Feynman has to look up everything manually. It's the third day already, and t-shirts are still missing. Luckily, he was also carrying an IonQ quantum computer, and it survived the crash. Feynman came up with a brilliant idea: having you design an image classification algorithm to help him sort out useful items from the cargo bay. Yes, it's going to be a quantum image classifier, since the IonQ quantum computer is all that is left.

Ok here's what you need to do to help Prof. Feynman find t-shirts.

IonQ Challenge #2: Dataset and the big picture.

The first step of image classification is encoding images captured by the camera into quantum circuits. This way the quantum computer can "see" the item.

You will be given an image dataset **(Fashion-MNIST)**. Your task is to make a data loading scheme that encodes the images into a quantum state as lossless as possible. Because Feynman doesn't have many classical computing resources, the encoded image should be interpretable by simple measurements at the end of the circuit.
Link: https://drive.google.com/file/d/11Vo2dJ4q--5hnsfuhCMvCO6EFcnjNx4q/view?usp=sharing

Data preprocessing is allowed, but higher preference is given to solutions that can directly store as much of the state as possible in the wave function (i.e., solutions that don't use much classical processing).

Newcomer Track

Task 1:
To get ourselves warmed up, let's implement an encoding commonly known as the product state encoding. To be specific, encode an 8-dimensional vector $\vec{v} = \sum v_i \vec{e}_i$ into 8 qubits by creating the state:
$|\psi\rangle = \prod_{i=0}^{7} {R^{(i)}}_x (f(v_i))|0\rangle$ here ${R^{(i)}}_x$ stands for a single-qubit x rotation applied on qubit "i". $f(x)$ is a function that maps values onto the rotation angle of the gate.

Write a Qiksit (or Cirq) circuit to encode the 8-dimensional vector [1,5,2,6,3,7,4,8] using this encoding scheme. Explain your method, simulate your circuit, and then reconstruct the 8-dimensional vector from the measurements of the quantum circuit.

Hint: If you want to encode a single number between 0 to 1. I can do it by rotating a single qubit along the x axis. $|\psi\rangle = R_x(0.5\pi)|0\rangle$ *. Then I can retrieve the encoded value as the probability of getting 1 from the computational basis measurements.*

Task 2:
As a second warm-up, Let's play with a more challenging encoding. Try to encode the same 8-dimensional vector [1,5,2,6,3,7,4,8] into the amplitude of the 3-qubit quantum state $|\psi\rangle = \frac{1}{Z}\sum_{i=0}^{7} v_i |i\rangle$, where $|i\rangle$ are a set of basis states $\frac{1}{Z}$ and is the normalization factor.
Explain your method, simulate your circuit, and then reconstruct the 8-dimensional vector from the measurements of the quantum circuit.

Task 3:

The above two schemes are just two examples of image encoding, each has its pros and cons. Now that we are warmed up with the concept of encoding, let's dive deeper:

a) Discuss the pros and cons of the two encoding methods you implemented in Task 1 and Task 2.

b) Propose a new method that finds a balance between the advantages and disadvantages of these two methods. Explain how your new scheme works - describe how a quantum circuit is constructed according to the encoded image and how the image reconstructed according to the measurement results of the quantum circuit?

Task 4:

Now, we will use the method we developed above to load the small 3x3 image below into a quantum state:

0	0.5	0
0.8	1	0.8
0	0.5	0

Demonstrate your encoding scheme, and run the scheme using a simulator.

Hint: Here you may assume that the quantum gates are noiseless and use as many qubits as your simulator can support.

Advanced Track

Task 5:

Unfortunately, we still need to try harder to process real-world images. On one hand, actual quantum hardware is susceptible to noise. Also, the number of qubits available is limited. On the other hand, real images can be very large, for example, the images in Fashion-MNIST contain 28x28 pixels. Propose a new scheme to encode 28x28 Fashion-MNIST images that are compatible with the currently available hardware: 11 qubits with ~96% 2Q-gate fidelity. Your scheme can be an extension of what you proposed in task 3 or it could be a different one. Make sure to explain how the circuit is constructed and how images are recovered from the circuit measurements.

Hint: Lossy encoding is fine but the best performance will be where information within the image is preserved. ***Don't just encode the index of the image!***

Task 6:

Demonstrate your encoding scheme using a simulator and then run your scheme on actual QPU HW, with the provided dataset. Explain the feasibility of your scheme and show at least 5 images reconstructed from your actual QPU results. In your explanation consider QPU resources, system noise, and the quality of the reconstructed images.

Task 7:
Train a parametric quantum circuit to perform binary classification on the provided dataset (Fashion-MNIST). To be specific, you will perform the following steps to distinguish 50 images of shoes from 50 images of t-shirts.

Hint: Suggested steps:

1. *Integrate parametrized quantum circuits with the circuits encoded with images (from task 5).*
2. *Design a method to interpret the measurement results of the integrated circuit to perform the binary classification. For example, measure one particular qubit in the computational basis. Then assign positive inferences if <Z>>0 , or negative inferences inf <Z><=0.*
3. *Train the parametrized quantum circuits to minimize the inference error (mismatch between inference and label). You will need a cost function for training, possible options include mean square error (MSE), cross entropy loss, etc.*

You may find this repository for a quantum circuit training helpful:
https://github.com/daiweiionq/open_acess_teaching_notebooks

Requirements:

1. You need to properly explain your methods, the training procedure, and the results.
2. You should perform the training on a simulator.
3. You need to perform inference of at least 10 images on a QPU, using the parametric circuit trained on the simulator.

Final Note: The goal of this problem is to explore quantum computing approaches to machine learning problems. To this end, the final score will take into account the degree of quantumness in the program. Preference will be given to creative solutions with a minimal amount of classical processing in the final solution.