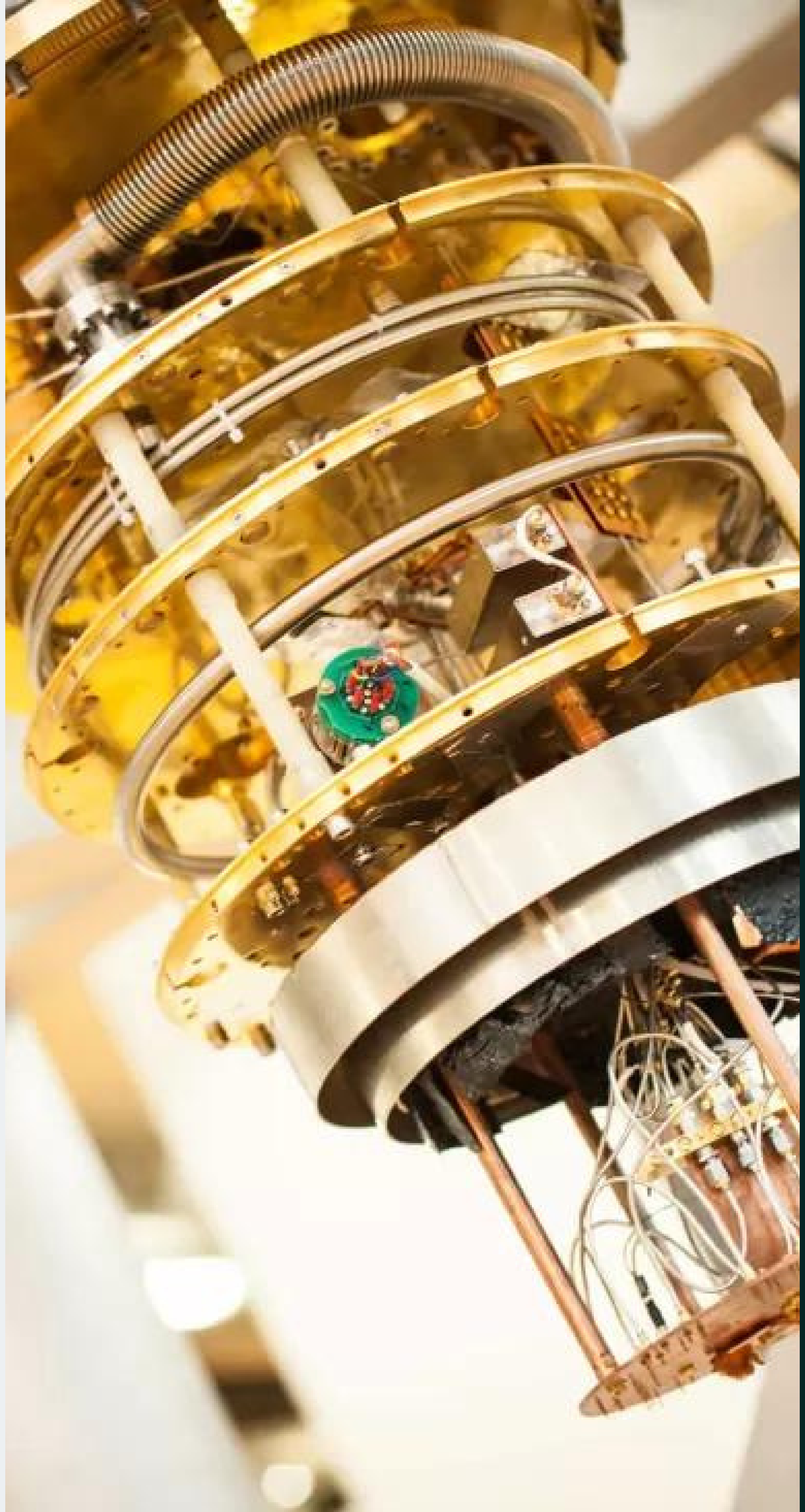
Project Presentation



QCourse 551-1

Project 27: QNN Algorithm Implementation

Mentor: Tal Michaeli

Team Members: Ashmit Gupta
Asif Saad
Roman Ledenov

Implementing a Hybrid Network to classify the MNIST Dataset

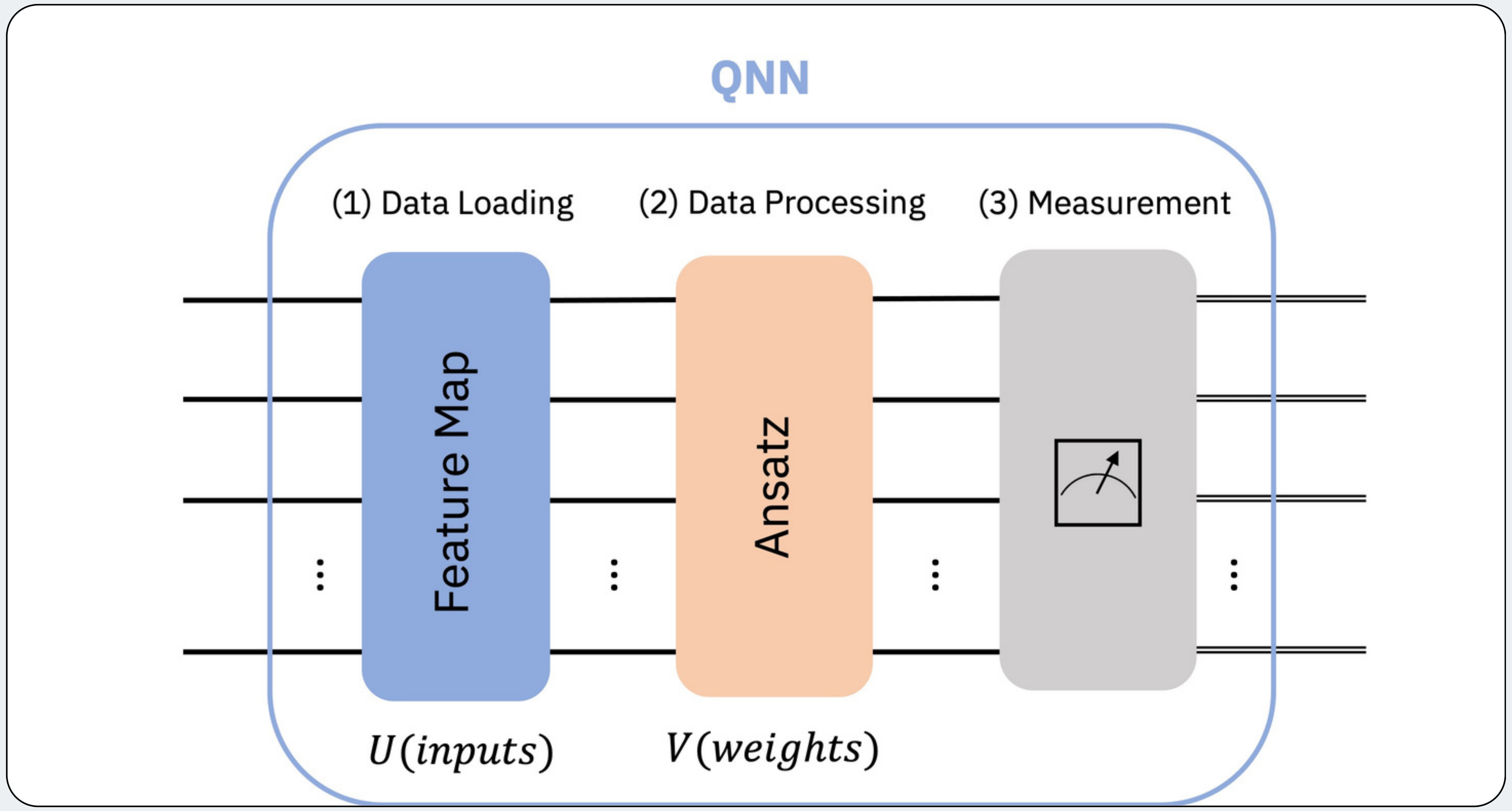
Agenda

Quantum Neural Network

Quantum Neural Networks are quantum algorithms based on parametrized quantum circuits that can be trained in a variational manner using classical optimizers.

These algorithmic models can be trained to find hidden patterns in data similar to their classical counterparts. These models can load classical data (inputs) into a quantum state, and later process it with quantum gates parametrized by trainable weights.

A Quantum layer take in classical data and return classical data.



Quantum Neural Network Algorithm Implementation

Problem Statement: Create a hybrid network consisting of both classical and quantum layers to classify the MNIST dataset.

Project Milestones

REQUIRED UPDATE

WE ARE STILL HERE!

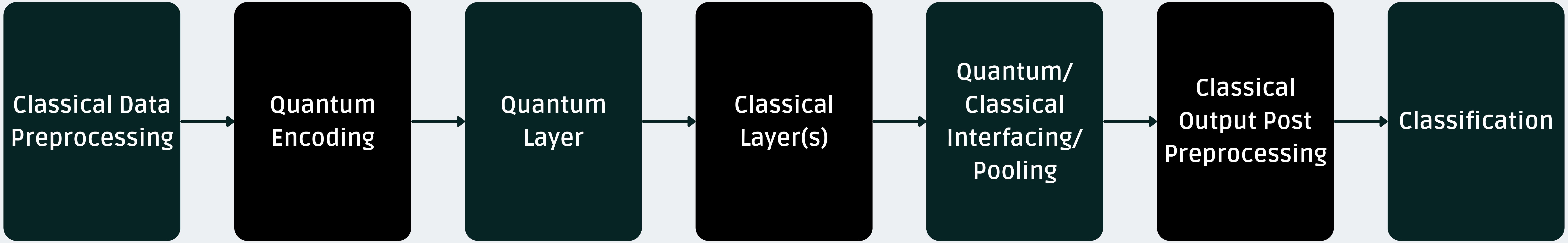
01	02	03	04	05	06
WEEK 1-3	**WEEK 4-5**	**WEEK 6-7**	**WEEK 8-9**	**WEEK 10-11**	**WEEK 12**
Discuss project, understanding the prerequisites and creating a work plan.	Pre-processing MNIST dataset and encoding it into a Quantum Circuit.	Design and Implement circuit for Quantum Layer.	Training and Testing of our QNN,	Extend our research and start writing our project research paper.	Finishing up and final presentation.

Re-evaluating QNN Architecture

This Hybrid QNN architecture combines classical and quantum processing for efficient MNIST digit classification. Classical data preprocessing is followed by quantum encoding, where classical information is translated into quantum states.

The Quantum Layer utilizes quantum gates to perform computations, followed by Classical Layer(s) for further processing. The Quantum/Classical interfacing or pooling layer integrates quantum and classical information, leading to classical output post-processing. Finally, the classification step provides the ultimate prediction.

This hybrid approach leverages the strengths of quantum computing while maintaining compatibility with classical methods, offering a promising paradigm for enhanced machine learning tasks on quantum devices.



Deciding on Image Dataset

Initially MNIST had 28x28 pixel images, and with the help of pytorch transformer, it was reduced to 8x8 pixel. However, it was still too much for a quantum circuit to take on.

Coarse-Grained 4x4 pixel MNIST Dataset has been found, and better suited with our requirement.

```
[[  1  63  86  48]
 [  1 121  46   0]
 [  0   7 162   2]
 [ 50 132  46   0]]
```

Fig: Part of training dataset obtained as Coarse-grained.

The dataset is segmented into testing and training files, and training file contains 6000 image of 4x4 pixel

Updating encoding scheme

Previously, we had to utilise at least, 32 qubits to encode the MNIST dataset. However, this time we have found way to reduce the encoding qubits down to 4, with the help of encoding along X,Y,Z axis across the whole bloch sphere.

```python
def angle_encode_image(image, size):
    # Create a quantum circuit
    qc = QuantumCircuit(size, size)

    # Angle encode pairs of pixel values into the quantum state
    for i in range(size):
        angle_rx=image[i,0]*(2*np.pi/255)
        angle_ry=image[i,1]*(2*np.pi/255)
        angle_rz=image[i,2]*(2*np.pi/255)
        angle_rx1=image[i,3]*(2*np.pi/255)

        qc.rx(angle_rx,i)
        qc.ry(angle_ry,i)
        qc.rz(angle_rz,i)
        qc.rx(angle_rx1,i)

    return qc
```

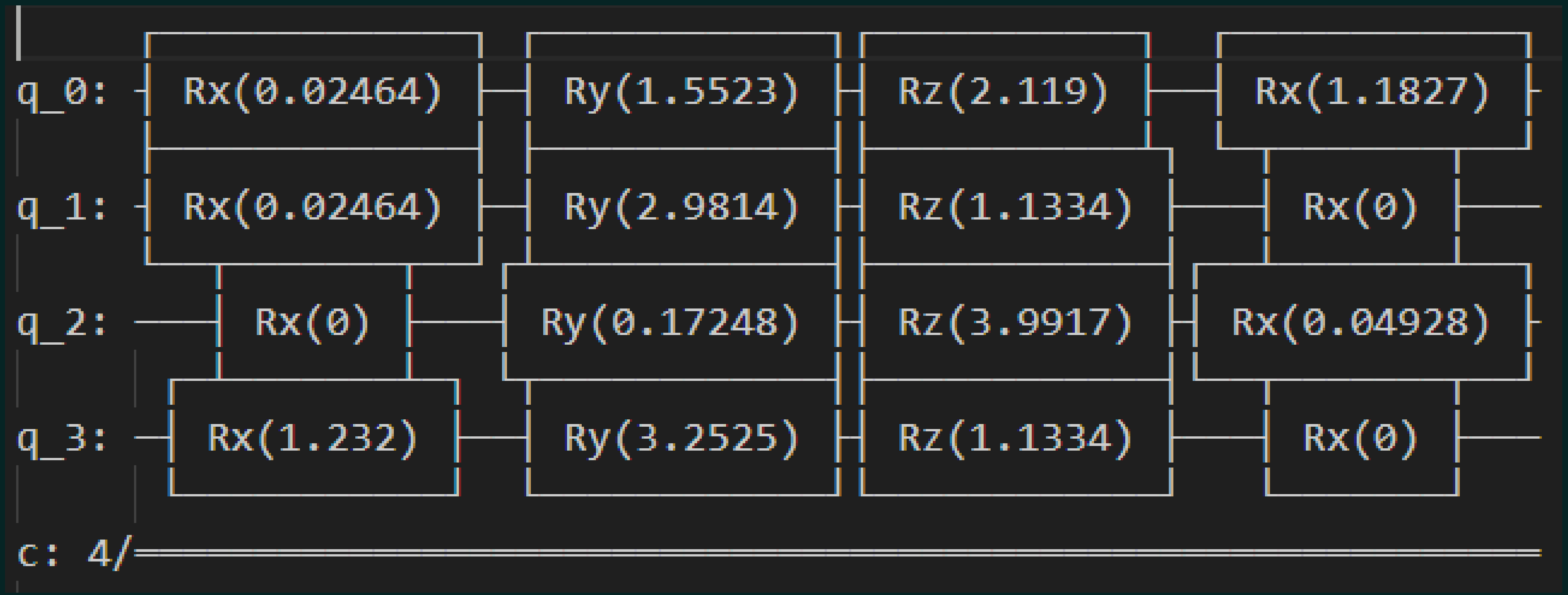
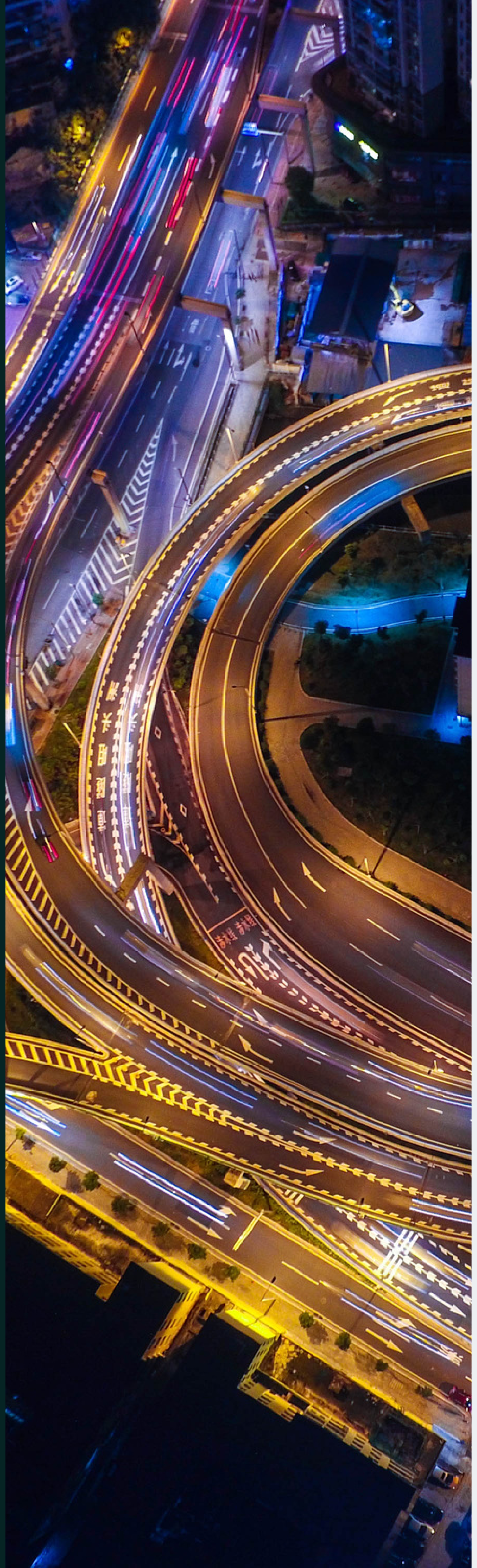


Fig: Snapshot of a quantum circuit encoding image dataset into 4 qubits.

Future Plans



Quantum Neural Layers

- Deciding on unitary operations to embed into each quantum neurons.
- Alinging classical optimisation procedure to manipulate unitary matrix.

Pooling

- Combines outputs of classical and quantum computation.
- Two approaches: Direct interfacing and Indirect interfacing.
- Various types of pooling layers (e.g., Averaging, Max)

Challenges and Obstacles

- Extraction procedure to obtain classical information from quantum space.
- Course of action to transform any image into 4x4 pixels.
- How to cope with PQC as quantum layers and integrate with classical optimiser.

Resources

- Paper on Quantum On-Chip Training with Parameter Shift and Gradient Pruning
- Classiq QNN User Guide: https://docs.classiq.io/latest/user-guide/built-in-algorithms/qml/qnn/
- Project Repository on GitHub

Conclusion

Goal:
To create an hybrid Quantum Neural Network which can classify MNIST Dataset Images.

Current Stage:
Dec Resolving the quantum layers' neurons calculation and integrate with Adam Optimisation and also placing convolutional and pooling operation as Qauntum Circuit.

Short term plans:

- Deciding on Unitary matrix and its parameters for each neurons.
- Optimisation methods to integrate with neurons unitary parameters.

Thank you!

Do you have any questions?