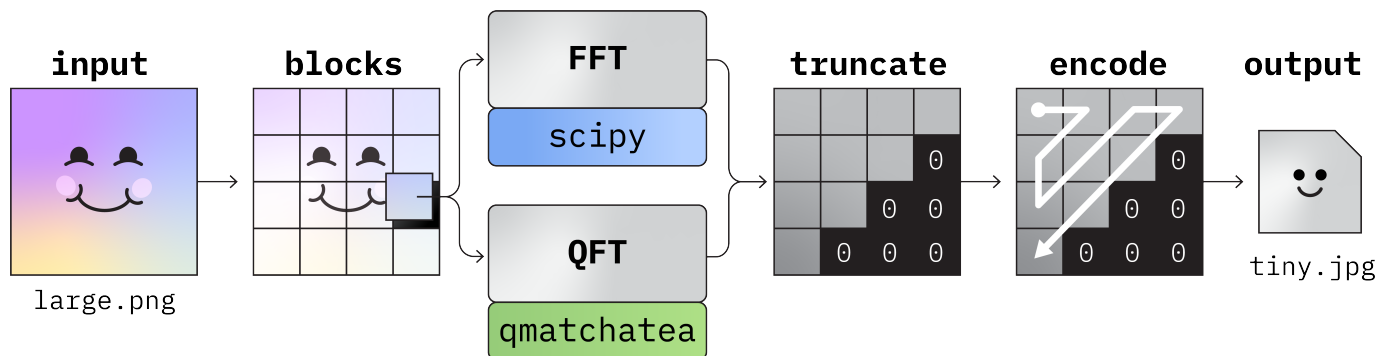


Topic 7: Data compression with quantum Fourier transform

What to expect

In this project you will learn how to compress images using the jpeg algorithm. In particular, you will study how it is possible to use a quantum Fourier transform in the process. You will use qiskit to write the quantum circuit of the quantum Fourier transform and tensor network techniques to simulate it. A basic understanding of quantum mechanics and Python is necessary, while familiarity with Qiskit and tensor networks is beneficial but not required.

Overview



The Fourier Transform is the cornerstone of frequency analysis and the fundamental step in many signal processing algorithms, from noise cancelling to audio and image compression. By working in the Fourier space, we can optimally approximate an input signal, for example, by cutting the frequencies that are not visible/audible by a human. This is the basic principle behind lossy image (JPEG) and audio (MPEG) compression, which gave us the ubiquitous jpg and mp3 codecs that we are so familiar with.

The best available (classical) implementation of the Fourier transform is the *Fast Fourier Transform* (FFT) algorithm. Despite being highly optimized, it scales as $O(N \log_2 N)$, where N is the number of entries of the vector we are considering. Without loss of generality, we constraint ourselves in the case $N = 2^n$, such that the FFT scaling can be expressed as $O(2^n n)$. Interestingly, the quantum version of the Fourier transform (QFT), presents an exponential speedup in the scaling $O(n^2)$, where we can now identify n with the number of qubits. For this reason, the QFT is at the basis of many quantum algorithms, such as the celebrated Shor's algorithm, over classical methods for the notoriously complex problem of number factorisation⁹.

In this project we will tackle the compression of images using the standard JPEG compression codec. First, we will use the classical FFT algorithm and then, we will implement the QFT algorithm using matrix product operators on qmatchatea. While a speedup is not guaranteed when running the QFT on a classical machine, an advantage is achievable when processing sufficiently large signals¹⁰, or by running the algorithm on a real quantum device.

Tasks

- Implement the JPEG image compression algorithm using the FFT.
- Write the quantum circuit, implement the QFT algorithm in qiskit.
- Modify the application of the FFT in the JPEG algorithm with a QFT.
- Apply the algorithm to different images and different scales. Compare the computational time and the scaling for the classical FFT and the QFT.
- **Optional:** Repeat the analysis using approximate quantum Fourier transform. Study how the approximation influences the performances.

⁹P. Shor, in Proceedings 35th annual symposium on foundations of computer science (1994), pp. 124–134.

¹⁰J. Chen et al., PRX Quantum 4, 040318 (2023).