

## Prerequisite

1. Understand image processing in Julia, please check YouTube Video: [Working With Images in Julia | Week 1, lecture 3 | 18.S191 MIT Fall 2020 | Grant Sanderson](#).
2. Understand fast Fourier transformation for image processing, please check the following videos by Steve Brunton:
  - [The Fast Fourier Transform \(FFT\)](#). It may require some knowledge about complex numbers.
  - [Image Compression and the FFT](#)
3. Understand tensor network, please check the following repository: [Tutorial on Tensor Networks](#).
4. Understand basic optimization theory, please check:
  - The 3blue1brown video: [Gradient descent, how neural networks learn | Deep Learning Chapter 2](#)
  - Manifold optimization: YouTube video [Manopt.jl: Optimisation on Riemannian Manifolds | Ronny Bergmann | JuliaCon 2022](#) and Julia package [Manopt.jl](#).

## Get started

1. Go through the code in `examples/img_process.jl`. It may require some knowledge about manifold optimization, please check the documentation page of [Manifolds.jl](#). Manifold optimization is very similar to gradient based optimization in machine learning, but with some additional constraints.
2. Read the note `note/main.typ` to understand the theory underlying the code.

## Tasks

1. Use GPU to speed up the code, please check the documentation page of [CUDA.jl](#).
2. Setup some image datasets and train the tensor network on the datasets.
3. Compare the performance with the Fourier basis.