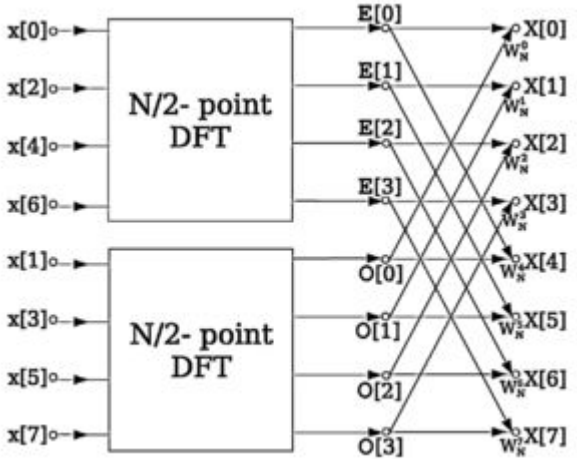
FFT for Image Compression

Mark Koszykowski and Omar Thenmalai

FFT Algorithm

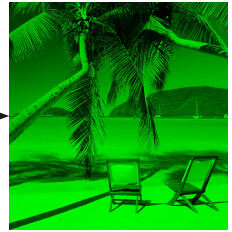
- Cooley-Tukey Radix-2 Inplace, O(Nlog(N))
- Bit reversal preserves order

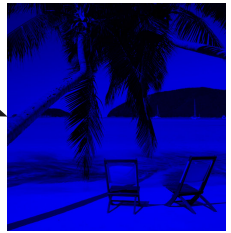


Process

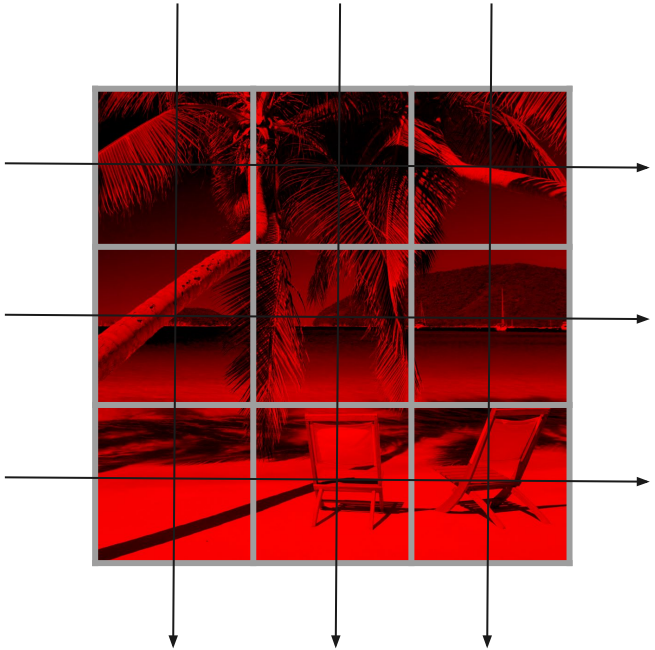
1) Split Color Channels
2) Perform 1D FFTs on each row in the channel, then each column
3) Discard coefficients smaller than threshold
4) Inverse FFT

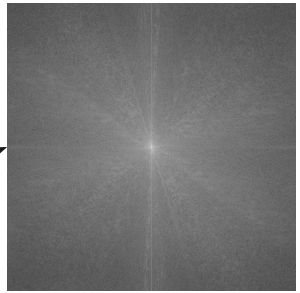
Split Color Channels



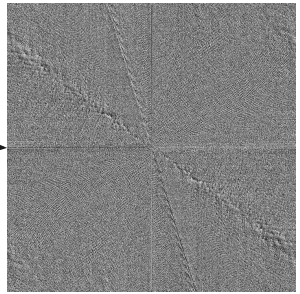


Fourier Transform



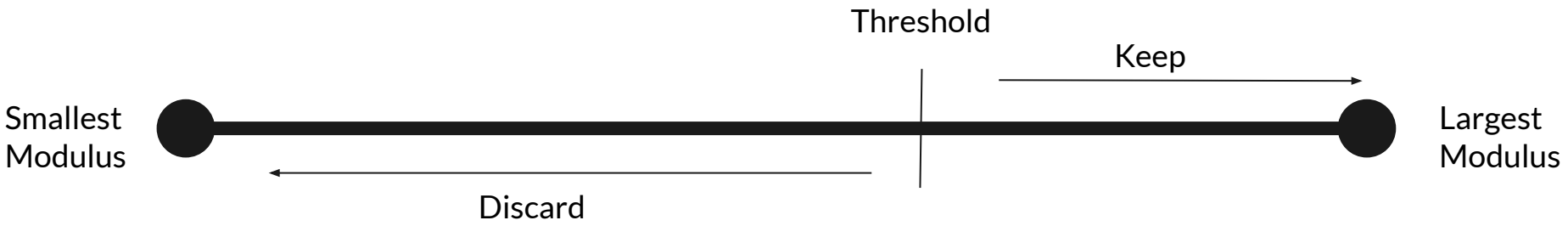


Magnitude



Phase

Thresholding



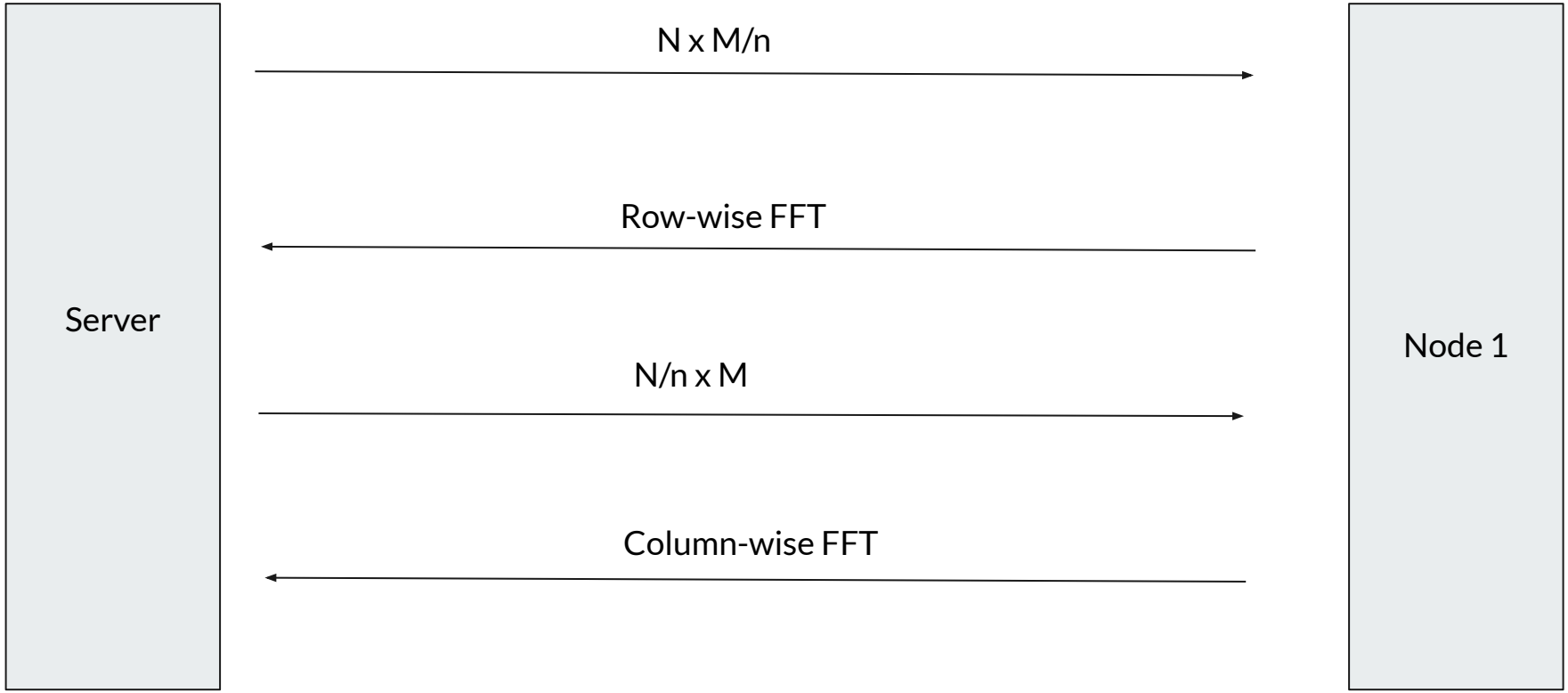
Inverse Fourier Transform

1) Conjugate
2) FFT
3) Conjugate
4) Scale

Architecture

- Static number of EC2 instances serve as workers
- VPC
- Backend server/coordinator
- Static HTML frontend

n = Number of Nodes
NxM Image



Repeat for IFFT and then Compress...

Demo

Future Work

- More efficient implementation for non radix 2 images.
- Dynamically adjusting the distributed algorithm based on number of nodes.
- Convert complex values from objects to primitive types.
- Add ability for multiple connections.