

Final Project

Jainil Patel

Problem Overview

1. Loop over slide using a 256x256 window and a 128x128 frame grid
2. Apply a 7x7 Gaussian smoother
3. Compute 2D FFT and convert it to magnitude
4. Compute a 16-bin histogram of log-magnitude spectrum
5. Output the mean and std dev across all frames

Architecture - Imaging tools

1. Open SVS files
2. Read frame coordinates
3. Extract a 256x246 RGB window using multithreaded I/O

Architecture - GPU based Gaussian Filtering

I implemented the 7x7 Gaussian kernel, normalized it, and applied it with:

```
F.conv2d(channel_data, self.gaussian_kernel, padding=3)
```

Running this on the GPU dramatically speeds up convolution, which is one of the most expensive parts of the operation

Architecture - FFT

For each smoothed channel:

```
fft_result = fft2(smoothed.squeeze())
```

```
fft_shifted = fftshift(fft_result)
```

```
magnitude = torch.abs(fft_shifted)
```

```
log_magnitude = torch.log(magnitude + 1e-8)
```

This produces a stable frequency-domain response of each window

Architecture - 16 Bin Histogram per Window

The log-magnitude spectrum is then flattened and histogrammed:

```
hist = torch.histc(log_mag_flat, bins=16)
```

Histograms are computed separately for R, G, and B then summed together since the assignment specifies the spectrum should be independent of color.

Architecture - Streaming Statistics

Computes:

1. Running Sum of histogram
2. Running Sum Square of the histogram
3. Total frames processed

Performance Optimizations

1. Batching and Chunking

- a. Windows are processed in GPU batches
- b. Frame coordinates are split into chunks to prevent memory spikes

2. Uniform Frame Sampling

- a. Images with extreme numbers of frame positions, the system samples them uniformly to cap processing load

3. Memory Cleanup

- a. After each batch, unused memory is freed to prevent buildup during long runtimes

Results - Image Content

Over 10 images (10,000 frames) the system processed the following characteristics:

1. Mid-frequency (bins 5-8) have the highest energy
2. Very high-frequency bins have near-zero values (removed by Gaussian smoother)
3. Overall statistics
 - a. Mean = 144.0
 - b. Std dev = 153.57

Results - Time

Runtime of 8mins and 4sec over 10 images.

~484 seconds

~48.4 seconds per image

Conclusion

This design incorporates:

1. Mathematically correct 7x7 Gaussian Filter
2. Efficient FFT computation
3. Memory and performance optimizations

This system successfully speeds up image processing by leveraging GPU usage in heavy computational loads.