

CS 205 Project – M4

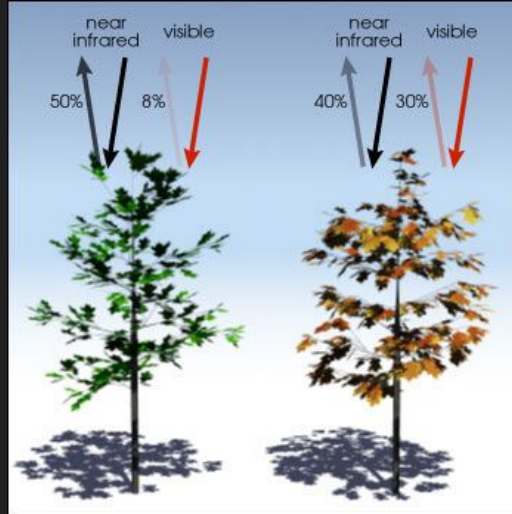
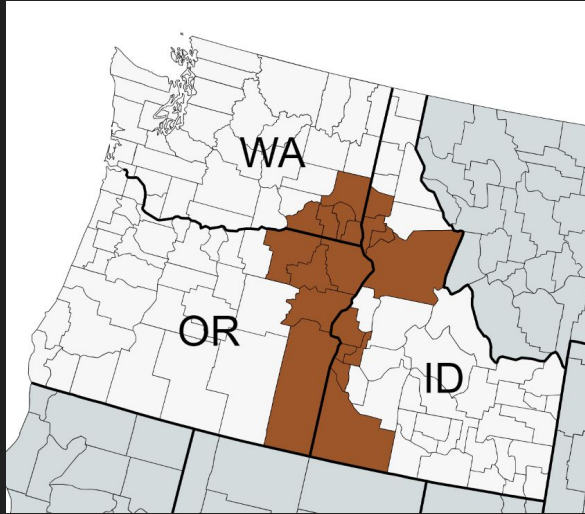
Parallelizing Crop Phenology
reports via NDVI datasets



Team 18:

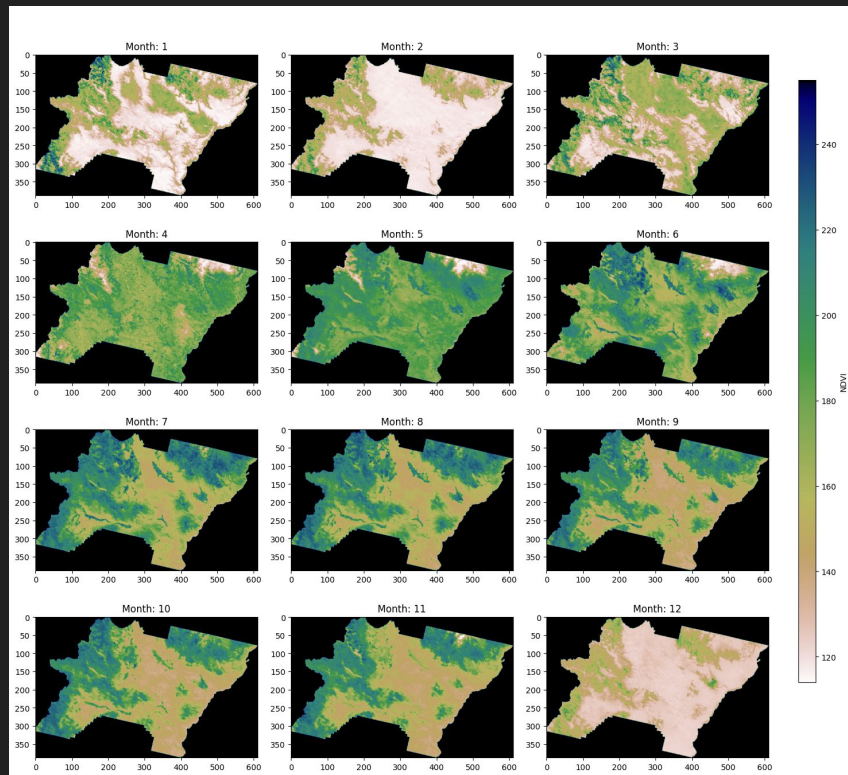
Elaine Swanson, Alex Johnson,
Saketh Mynampati, Alyssa Ross

Background & Significance



- Normalized Difference Vegetation Index (NDVI)
- Helpful to agricultural and urban planning; climate science
- Focus on specific counties in OR, WA, and ID from 2001-2023

Scientific Goals and Objectives

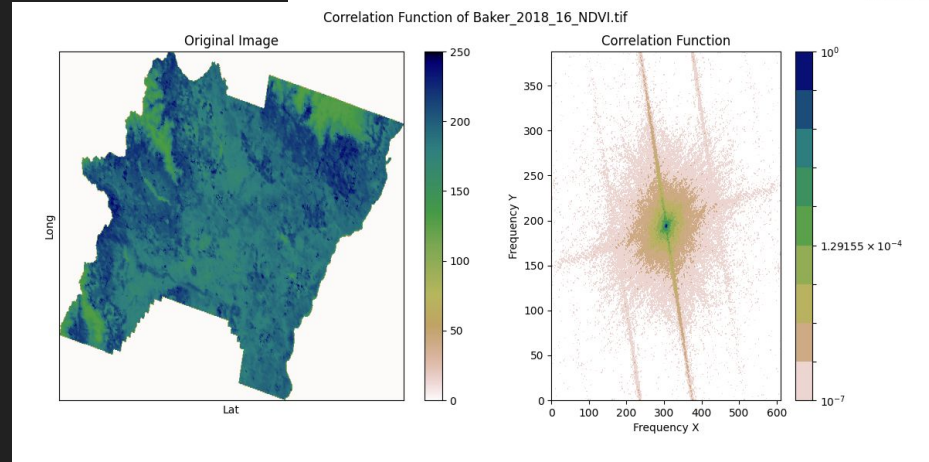
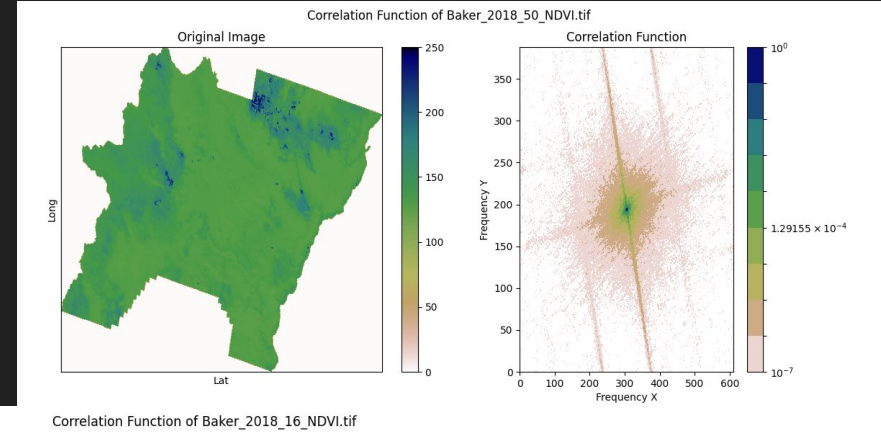


Some questions our work can address:

1. How can we effectively monitor trends of vegetation health over large geographic areas using satellite imagery?
2. What are the best methods to process and analyze this data to deliver timely, actionable insights?
3. Can enhanced data processing improve decision-making in agriculture, environmental conservation, urban planning, etc?

Framework Overview

- .tif Handling with GDAL
- Matrix multiplication optimizations:
correlation function
- Thread Management
- Memory Management



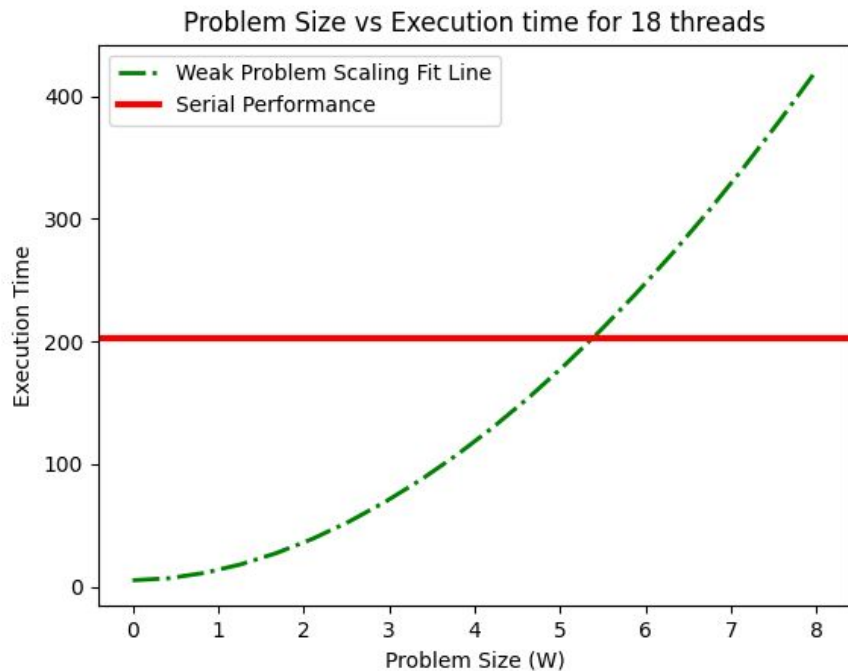
Libraries & Computational Techniques



```
#define ACCUMULATE\  
    accum[0] = _mm512_fmadd(img_slice, f_col[0], accum[0]);\  
    accum[1] = _mm512_fmadd(img_slice, f_col[1], accum[1]);\  
    accum[2] = _mm512_fmadd(img_slice, f_col[2], accum[2]);\  
    accum[3] = _mm512_fmadd(img_slice, f_col[3], accum[3]);\  
    accum[4] = _mm512_fmadd(img_slice, f_col[4], accum[4]);\  
    accum[5] = _mm512_fmadd(img_slice, f_col[5], accum[5]);\  
    accum[6] = _mm512_fmadd(img_slice, f_col[6], accum[6]);
```

transforms.cpp excerpt

Performance Analysis - Metrics and Scaling Tests



Strong Scaling: $S_p = 535.206$

Work size: $W_s = 10$ MB,

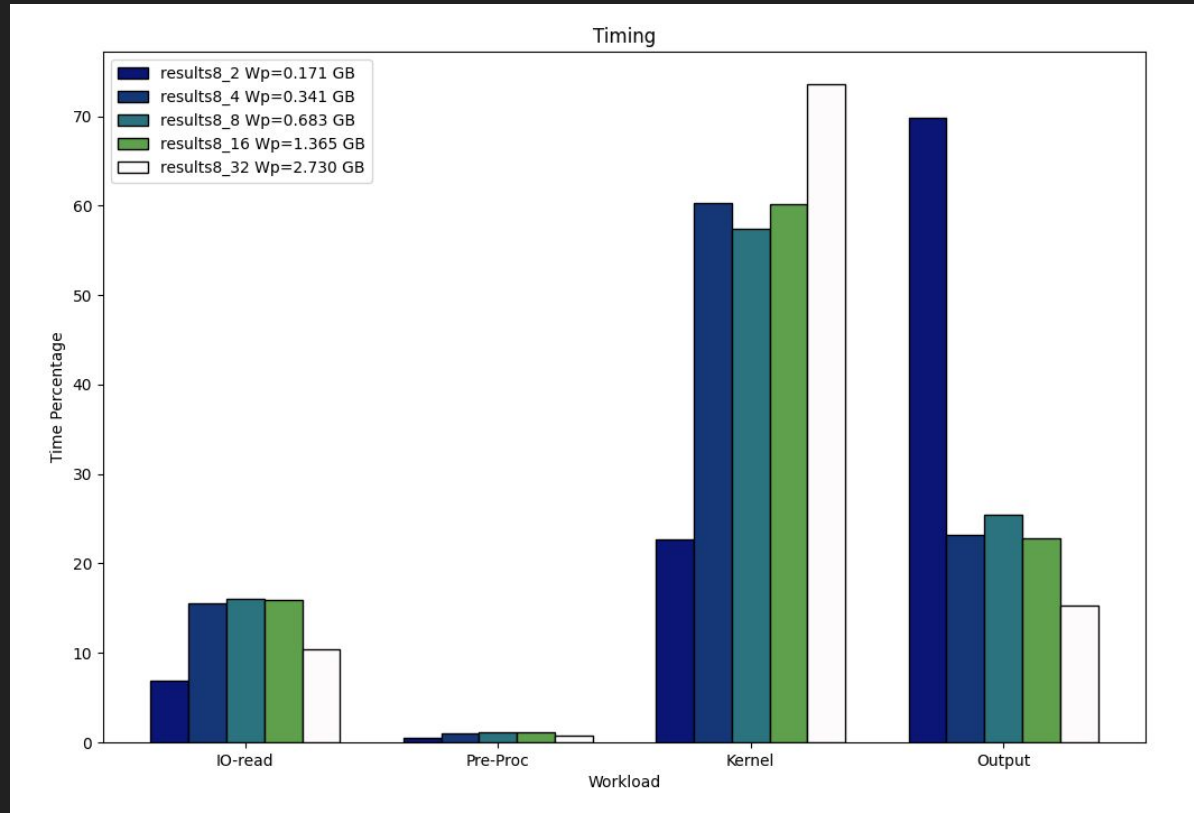
$T_s = 202$ s

Weak Scaling: $S_w = 501.5$

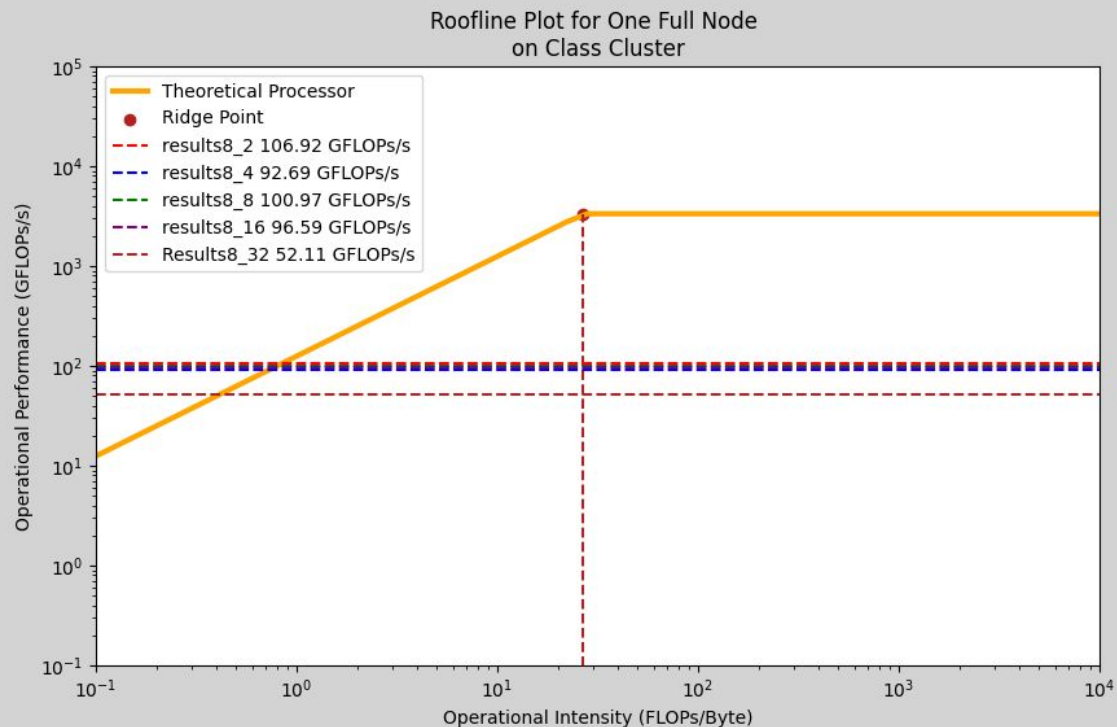
Execution Time: 202 s

$E = 13.811$ (due to cache
opt./vectorization being 16x)

Performance Analysis - I/O and Memory Requirements



Performance Analysis - I/O and Memory Requirements



Conclusion/Future Work

- Metrics
 - High Efficiency due to spatial locality and 18 threads
- Measurement
 - Despite the high efficiency, there are still slight updates that can be done
 - Padding
- Future Work
 - We could interleave computation with I/O in a more deliberate manner to increase peak performance.
 - Cache optimizations are very important to wall time
 - More could achieve higher performance

