

Robust Parallel Adaptive Smoothing

Danny Gibbs II
Christopher K. Lee

CS205 Final Project
4 December 2011

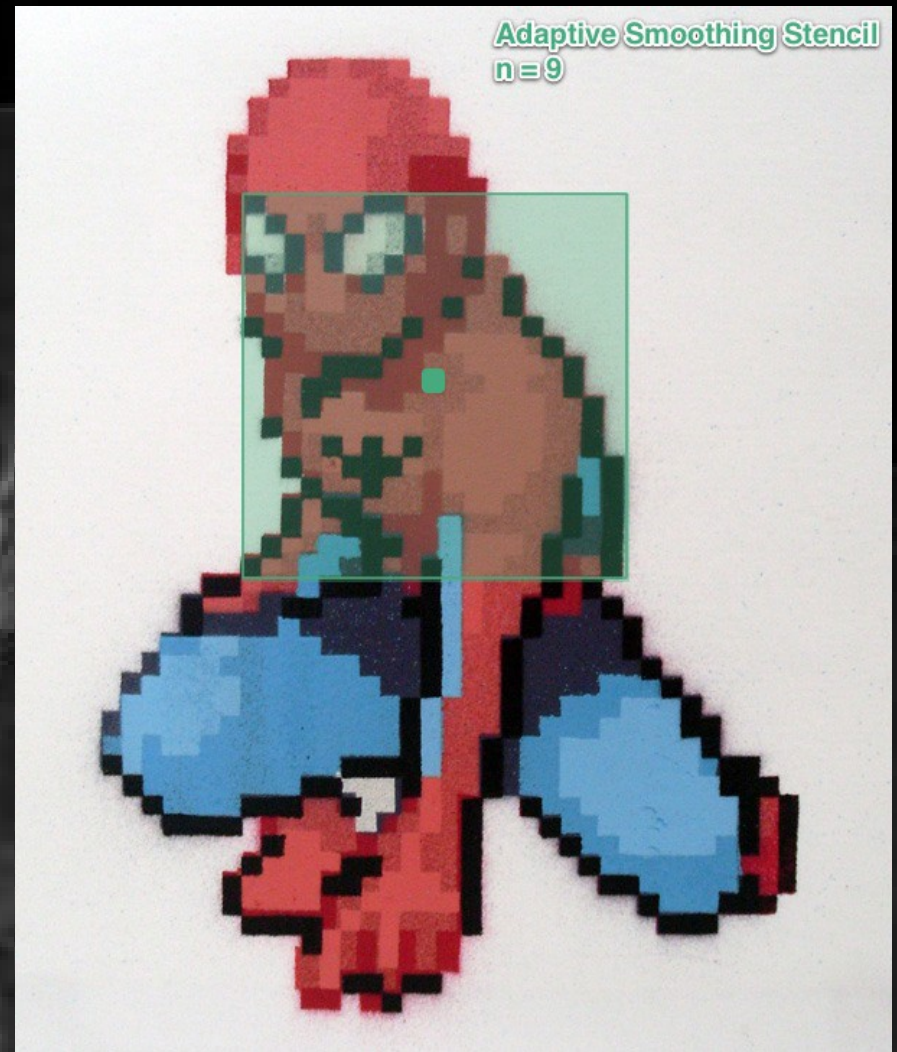
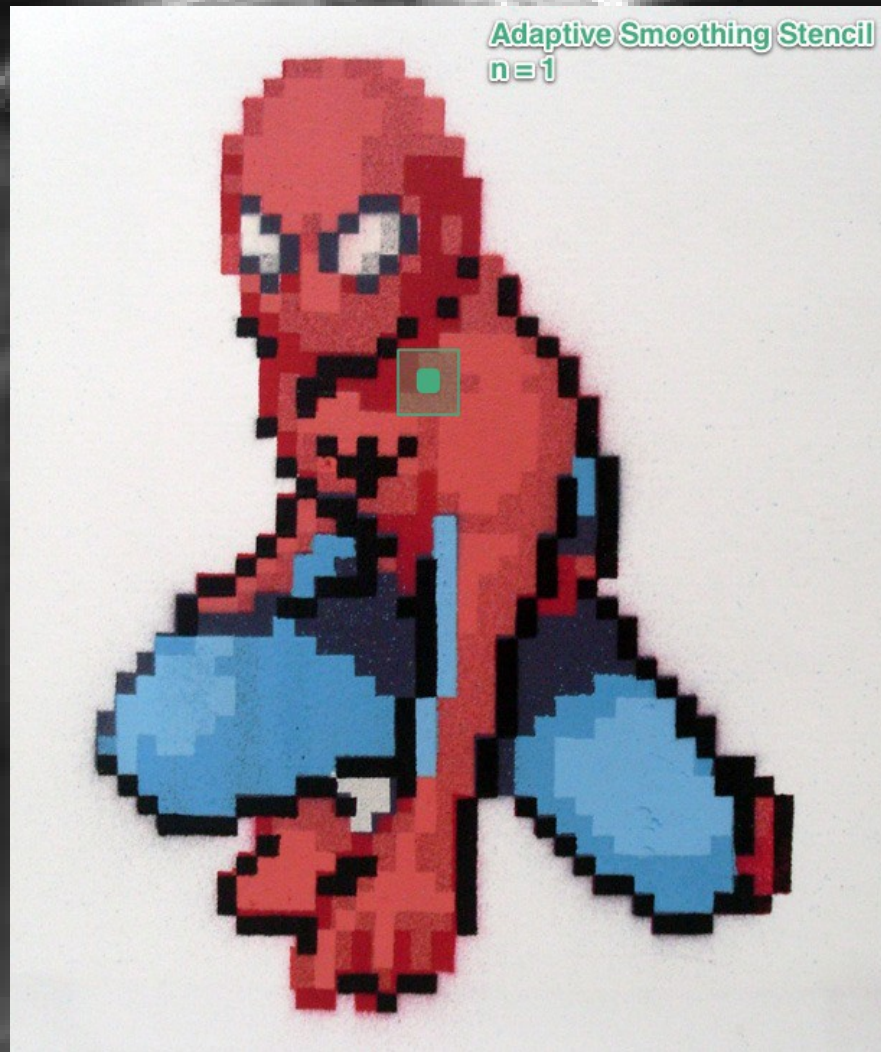
Overview

- Adaptive Smoothing
- CPU and GPU/MPI Implementations
- Results
- Conclusions
- Summary

Adaptive Smoothing

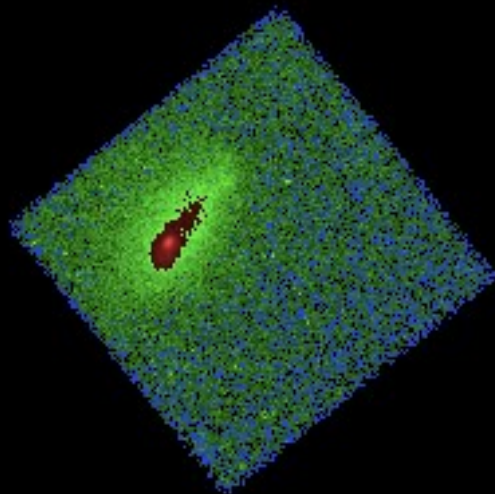
- Independently smooths each pixel
- Preserves small scale features
- Smooths larger features over larger area
- Only scales to Threshold or MaxRad
- Preserves energy (flux)

Example of Adaptive Smoothing: Threshold and MaxRad

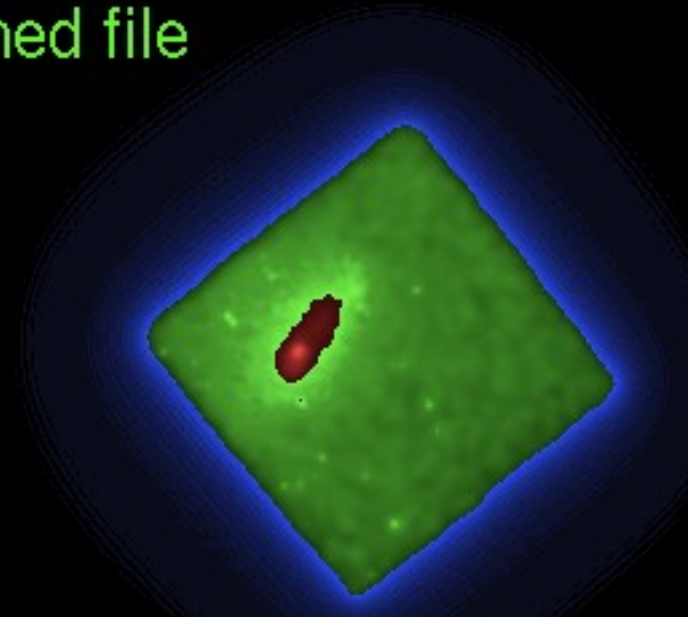


Observation ID: 11759

Event file



Smoothed file



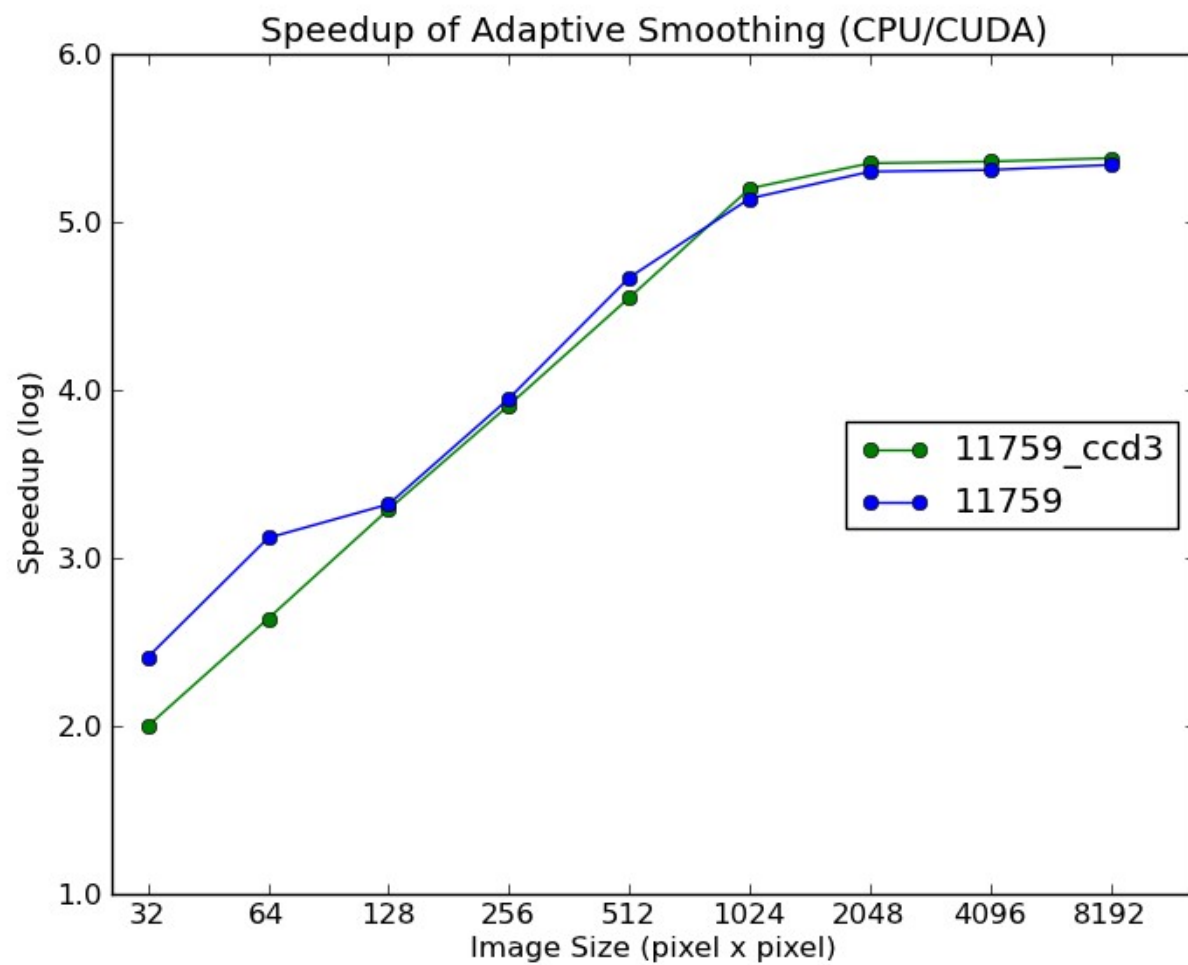
CPU Implementation

- Traditional Chandra Observation
 - dmimgadapt()
 - Worst Case: $O(n^4)$
 - 8192x8192 binned to 1024x1024
 - Run Time: ~ 30 minutes

GPU/MPI Implementation

- GPU
 - Embarrassingly Parallel
 - Three CUDA kernels
 - Use full resolution image
- MPI
 - Process Multiple Observations

Results



Conclusions

- CPU vs GPU
 - Full Resolution Image
 - CPU: ~ 4.5 Days
 - GPU: ~ 1.65 sec
 - Speed Up of $\sim 10^5$

Summary

- Adaptive Smoothing
- CPU and GPU/MPI Implementations
- Results
- Conclusions

Acknowledgments

A black and white astronomical image of a nebula, likely the Carina Nebula, showing intricate filamentary structures and a bright star in the lower-left quadrant. The image is used as a background for the slide.

CIAO: Chandra's data analysis system

Fruscione et al. 2006, SPIE Proc. 6270, 62701V, D.R. Silva & R.E. Doxsey, eds.

Special thanks to Kenny Glotfelty of the Harvard-Smithsonian Center for Astrophysics