Using CUDA to Accelerate Radar Image Processing

Aaron Rogan
Richard Carande
9/23/2010

Approved for Public Release by the Air Force on 14 Sep 2010, Document Number 88 ABW-10-5006

Company Overview

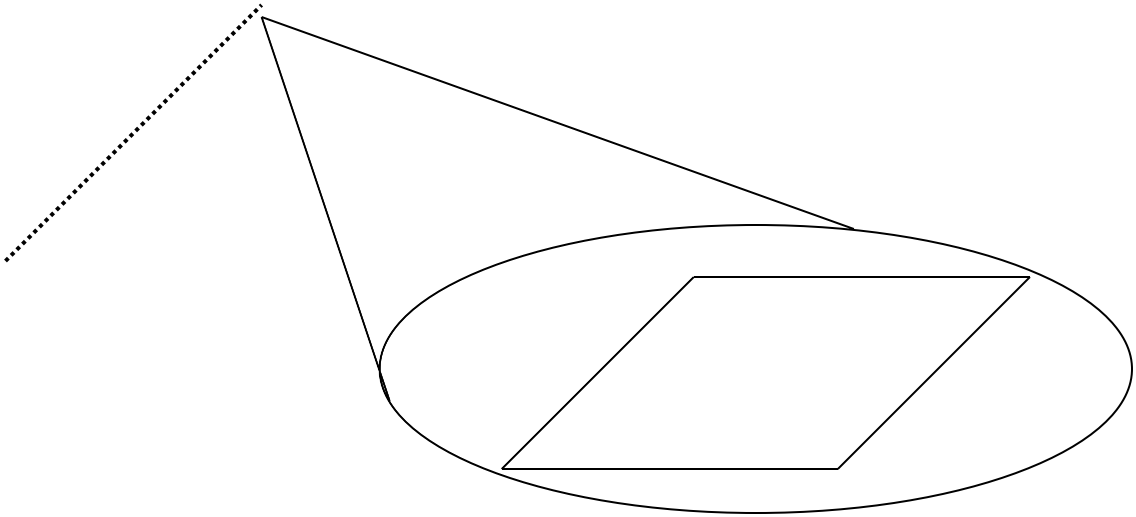
- Neva Ridge Technologies
  - Small company
    - 10 full time employees
  - Located in Boulder, CO
- What we do?
  - Radar image processing
    - Airborne sensors
    - Spaceborne satellites
  - Radar exploitation
    - Coherent change detection (CCD)
    - Amplitude change detection (ACD)
    - Interferometry

Outline

- What is synthetic aperture radar (SAR)?
- SAR processing and exploitation
- Problem statement
  - Air Force's current approach
  - Neva Ridge's GPU approach
- Algorithms overview
  - Standard backprojection (BP)
  - Fast backprojection (FBP)
- Results
- Future work
- Review

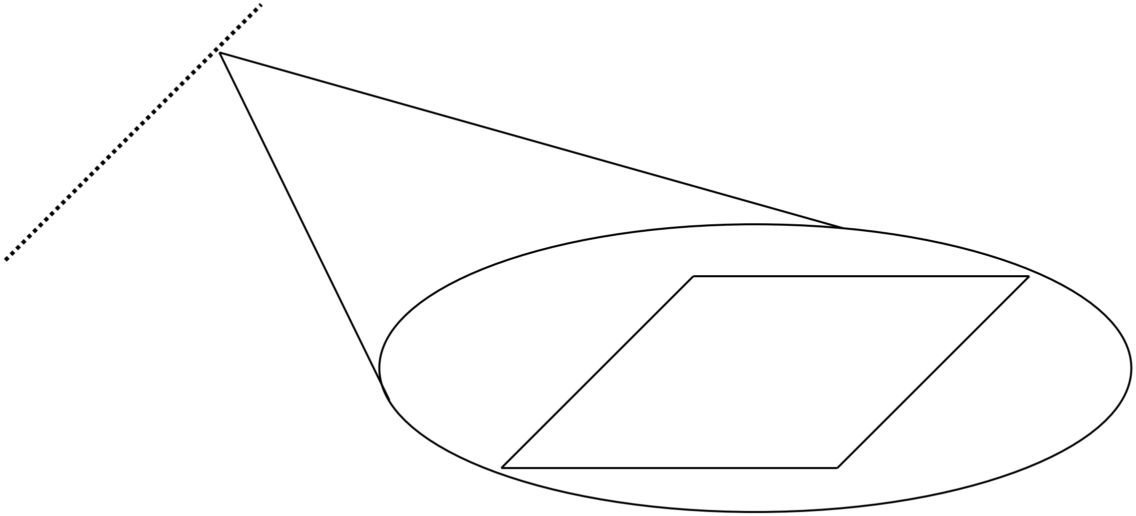
What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



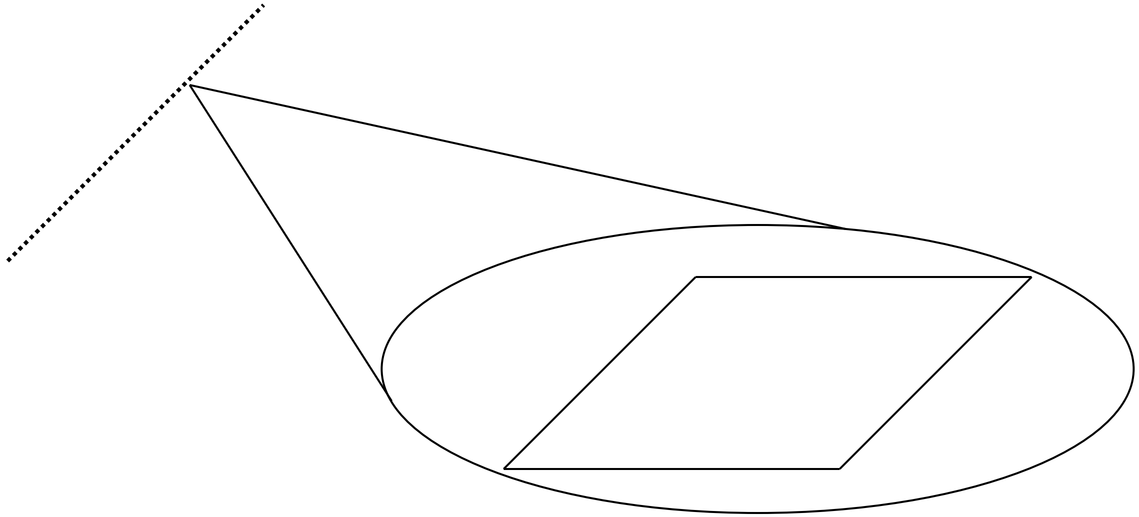
What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



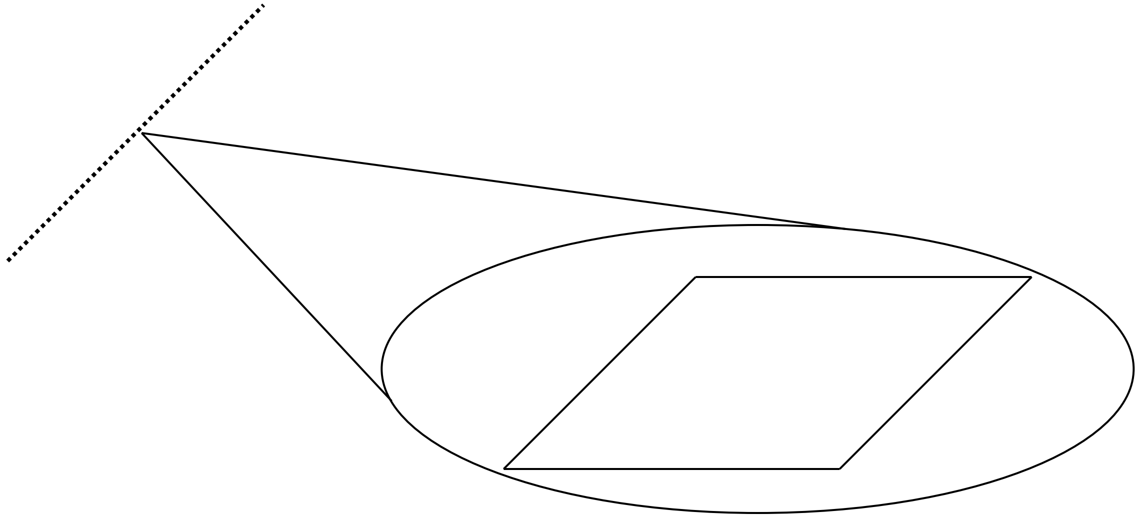
What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



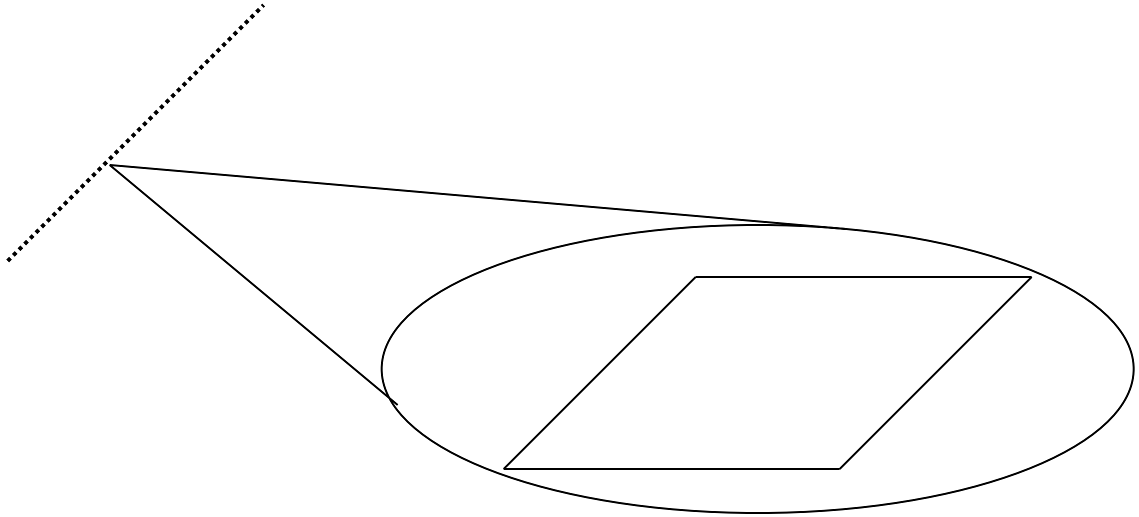
What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



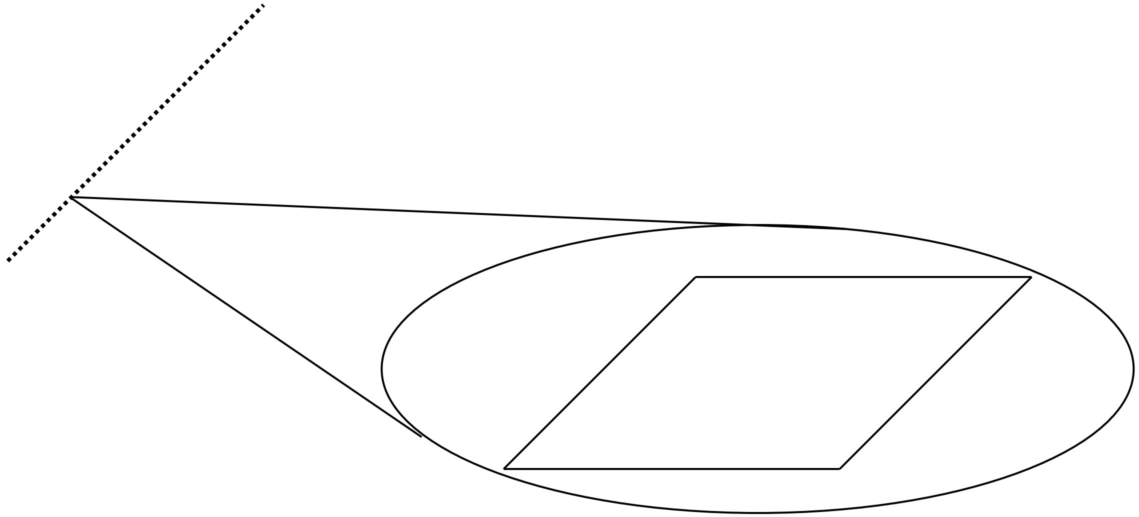
What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



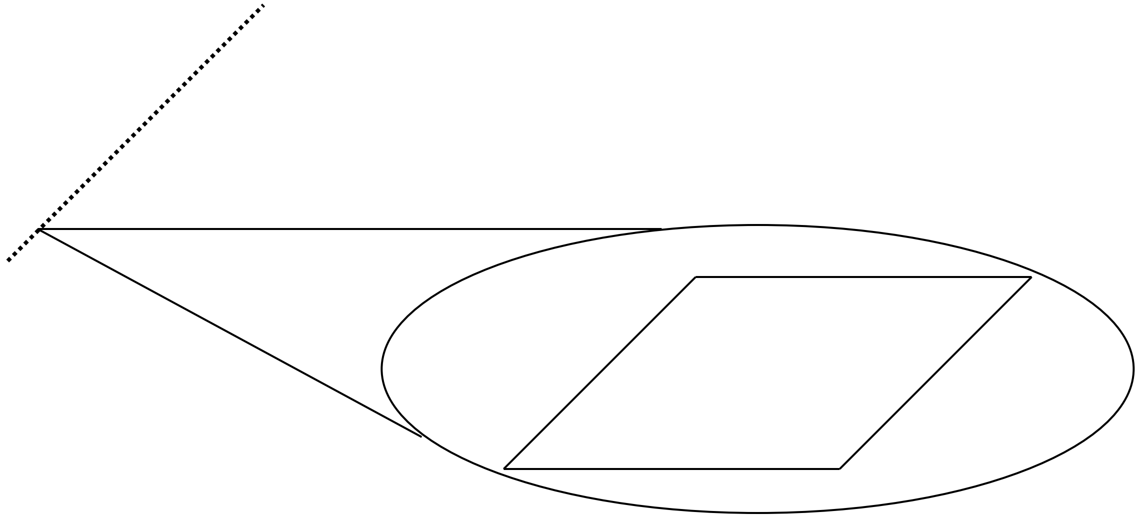
What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



What is SAR?

- Pulsed radar system
  - FM modulated pulses used
    - High range resolution
  - All weather
  - Day or night
- Synthesis large antenna
  - Coherent combinations of adjacent pulses
    - High cross range resolution
- Operating Modes
  - Stripmap
  - Spotlight
  - Combination



SAR Processing

- Algorithms
  - Frequency domain approaches
    - Chirp Scaling, Omega – K , Polar Format
    - Benefit
      - Fast $O(N^2 \log N)$
    - Drawback
      - Approximate solution
        - » Lower image fidelity
  - Time domain approaches
    - Backprojection
    - Benefit
      - Exact solution
      - Highest image fidelity possible
    - Drawback
      - Very slow $O(N^3)$

Example Imagery
(ka band ~4in resolution)

Courtesy of Sandia National Labs

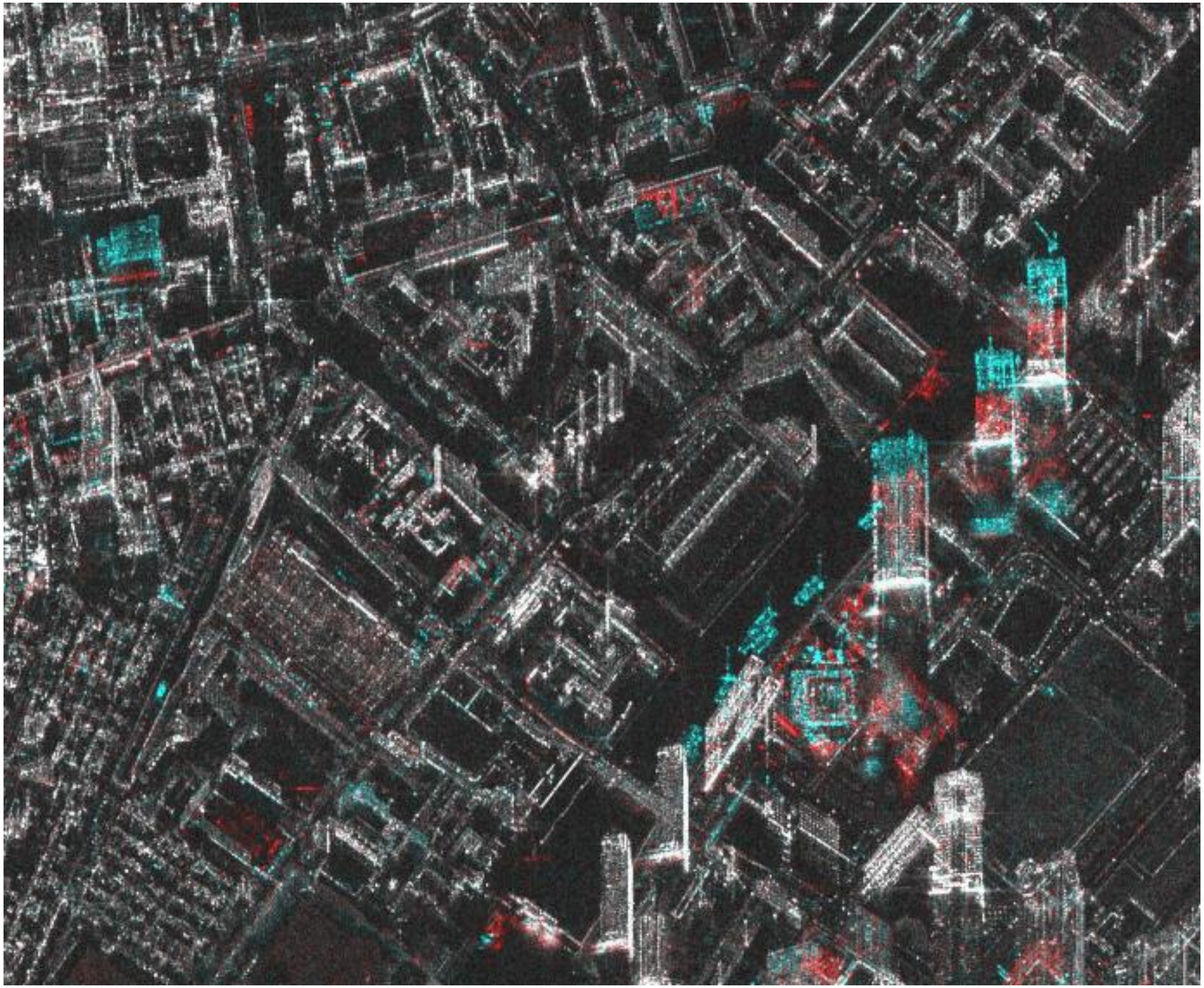
Example Imagery
(ku band ~ 1m resolution)

Courtesy of Sandia National Labs

SAR Exploitation

- Coherent change detection
  - Detect changes using phase and magnitude data
    - Detect sub wavelength changes
  - Lower image resolution due to averaging
- Amplitude change detection
  - Detect changes using magnitude only
    - Less sensitive than CCD
  - Maintains underlying image resolution
- Interferometry
  - Create elevation models of the underlying scene
  - Monitor land subsidence and uplift rates

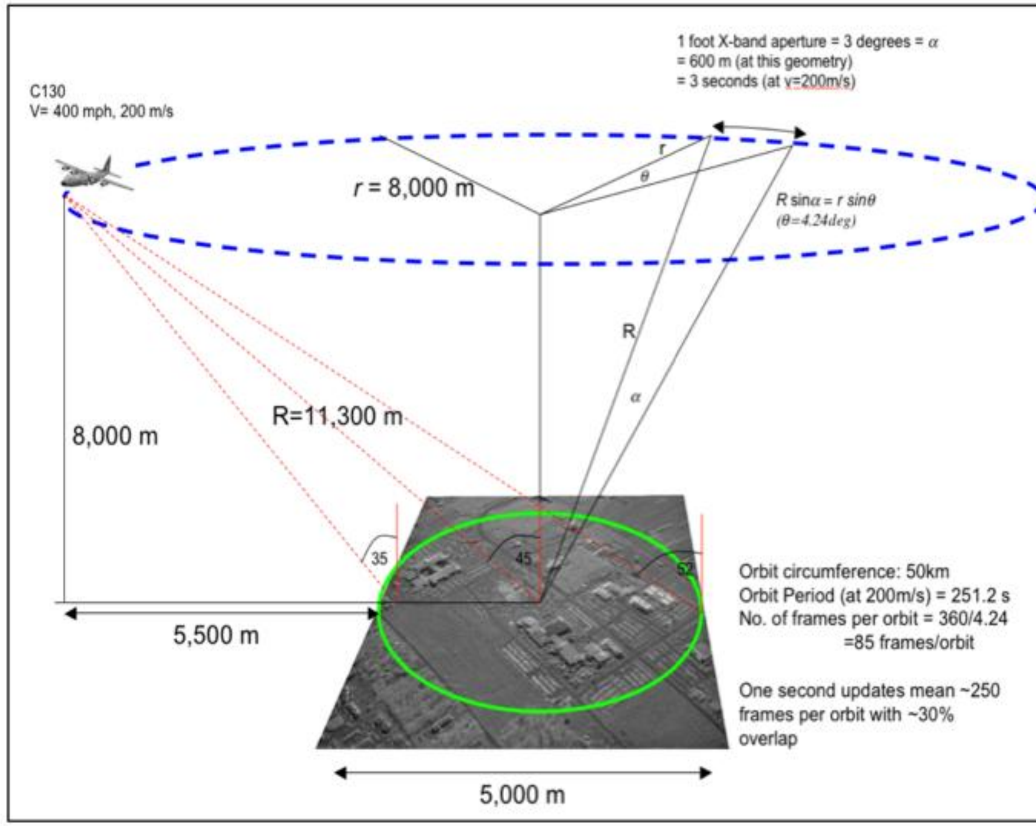
ACD Example Imagery



Courtesy of Infoterra

Problem of Interest

- Air Force contract
- Gotcha Sensor
  - Circular SAR
  - Capbilities
    - VideoSAR
      - Persistent Surveillance
    - Change detection
    - 3D SAR
    - Interferometry



Current Processing Capabilities

- AFRL's Gotcha data
  - 5km spot with ~16k x 16k output pixels
  - SGI Altix ICE 8200 Enhances LX, Xeon X5560 quad core 2.8 GHz super computer
    - 2048 cores
    - 308th fastest super computer in the world (http://www.top500.org/list/2009/06/400)
    - 23 TFLOPS
    - $ 2.2 Million+
    - $ 96 K/TFLOP
  - Real time processing
  - Non-portable solution
    - Can't deploy with sensor

Alternative Capabilities

- Fast Backprojection
- Use GPU for parallelization
  - – Single form factor ATX case
  - – Multiple GPU devices in single tower
  - – Portable and cheaper alternative solution
    - Current Gen. Hardware: nVIDIA GTX 480
      - – 480 Cores, 1345 GFLOPS
      - – ~$500, ~$380/TFLOP

Backprojection Algorithms

- Benefits
  - High image fidelity
  - Easy to parallelize
  - Process directly to a digital elevation model
    - Eliminates height of focus
- Drawbacks
  - $O(N^3)$ standard BP
  - $O(N^2 \log(N))$ fast BP
    - Almost same performance as Fourier based algorithms
    - Still not fast enough for real time applications
  - CPU real time processing capabilities are not portable

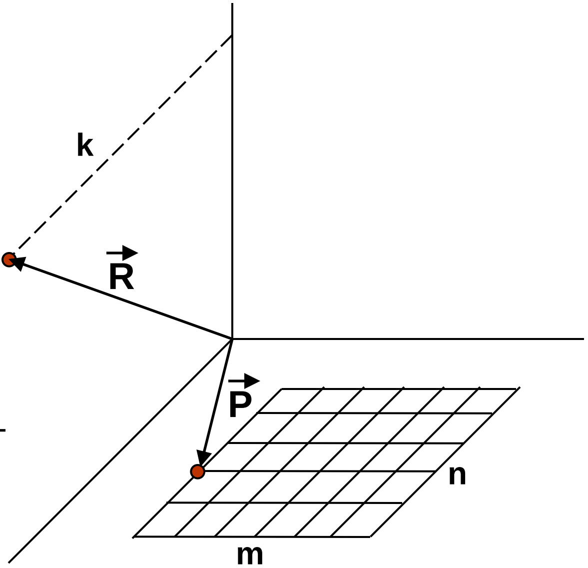
Standard Backprojection

- Input data:
  - k discretely sampled compressed pulses, $S_k(t)$
- Output data:
  - n x m selected output grid, $I(n,m)$ (not necessary planar grid)
- Goal
  - Sum the contribution from each pulse to each pixel in the selected output grid
- Simplified backprojection expression

$$I(n,m) = \sum_{k=1}^{N} S_k(\tau)$$

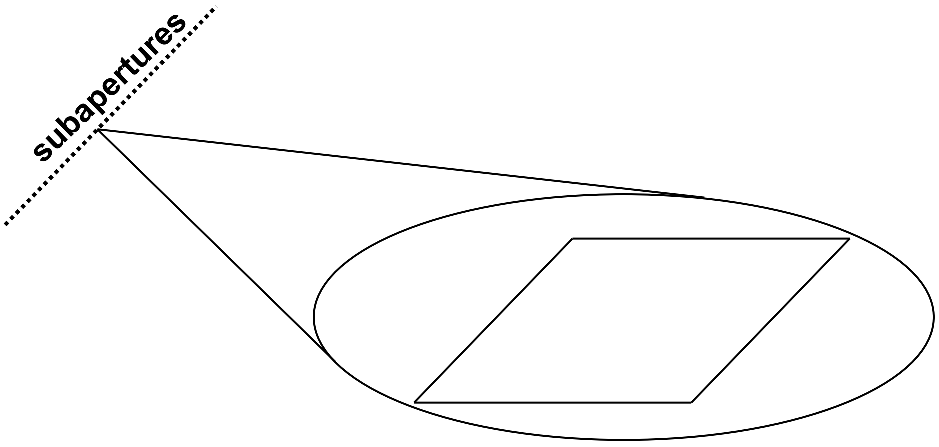
- Two way travel time, τ

$$\tau = \frac{2}{c}\sqrt{(R_x - P_x)^2 + (R_y - P_y)^2 + (R_z - P_z)^2}$$



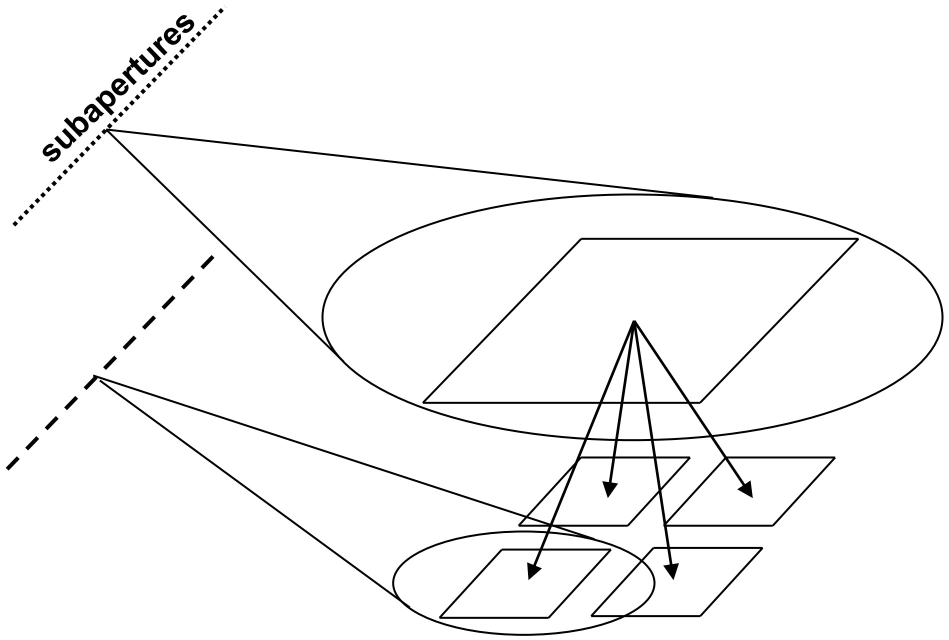
Fast Back Projection

- Recursive algorithm
  - Partition image
  - Digitally steer pulses
    - Trim range samples
    - Interpolate samples to be centered on subpatch
  - Low pass filter
  - Coherently combine adjacent pulses
- Backproject after recursion ends
- Not limited to quadtree algorithm



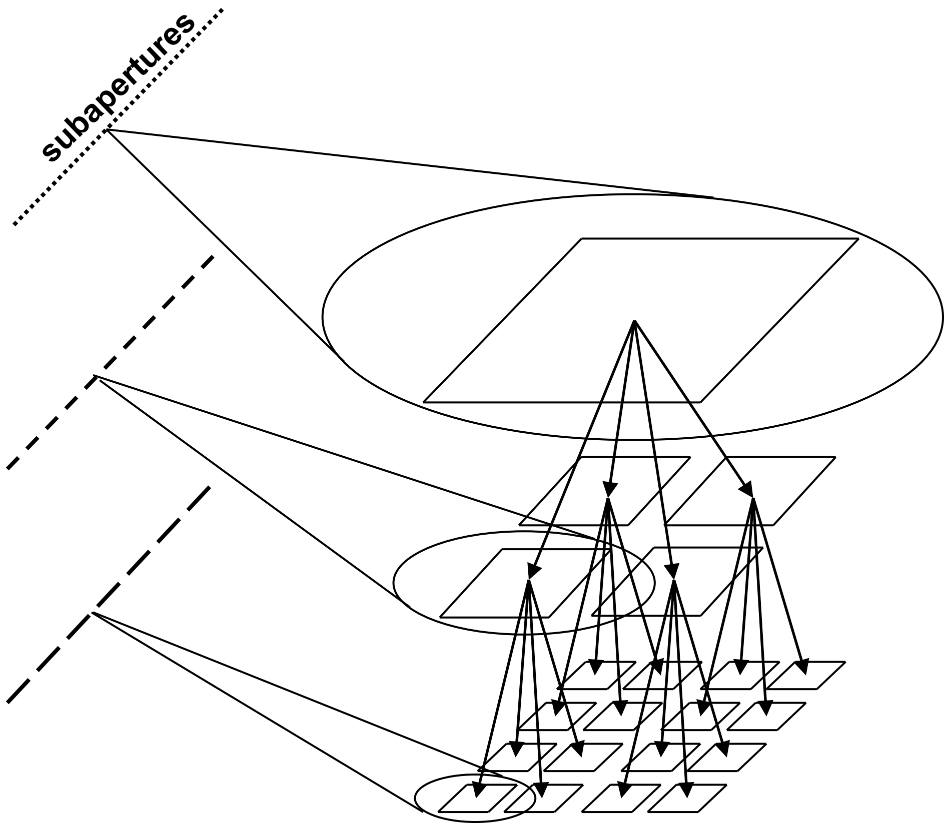
Fast Back Projection

- Recursive algorithm
  - Partition image
  - Digitally steer pulses
    - Trim range samples
    - Interpolate samples to be centered on subpatch
  - Low pass filter
  - Coherently combine adjacent pulses
- Backproject after recursion ends
- Not limited to quadtree algorithm



Fast Back Projection

- Recursive algorithm
  - Partition image
  - Digitally steer pulses
    - Trim range samples
    - Interpolate samples to be centered on subpatch
  - Low pass filter
  - Coherently combine adjacent pulses
- Backproject after recursion ends
- Not limited to quadtree algorithm



Parallel Implementation

- Back projection is "embarrassingly parallel"
- All pixels and pulses are independent

```
standardBackproject
  for k = 1:N
    for i = 1:n
      for j = 1:m
        calculate travel time, τ, from sensor position k to image pixel (i, j)
        interpolate continuous sample S_k(τ) from discretely sample pulse S_k(t)
        add contribution from pulse to image pixel I(i, j) = I(i, j) + S_k(τ)
```

```
parallelBackproject
  for k = 1:N
    kernelBackproject

kernelBackproject
  get image indices (idx, idy)
  calculate travel time, τ, from sensor position k to image pixel (idx, idy)
  interpolate continuous sample S_k(τ) from discretely sample pulse S_k(t)
  add contribution from pulse to image pixel I(idx, idy) = I(idx, idy) + S_k(τ)
```

Parallel Implementation Cont'd

- First algorithm written in CUDA
  - non funded research
  - Quick turnaround (only 1 week development time)
  - Demonstrate capabilities to Air Force
  - Naïve approach
- Problems with the naïve approach
  - Does not make use of shared memory or constant or texture cache
  - Partition camping not addressed
  - Global memory access not necessarily coalesced

Algorithms Tested

- Standard backprojection
  - Neva Ridge BP processor
    - CPU version
    - GPU version
  - Air Force BP processor
    - CPU only algorithm
- Fast backprojection
  - Neva Ridge FBP processor
    - CPU version
    - GPU version

CPU vs. GPU
Benchmark Hardware

- Intel Core i7 920
  - 2.67 GHz quad core processor
- 12 GB Kingston Memory
  - Tri-channel
  - DDR3 1333 MHz memory
- nVIDIA Tesla C1060 GPU
  - 240 stream processors
  - 1.3 GHz core clock speed
  - 4 GB GDDR3 800 MHz memory

Image Analysis (CPU BP)

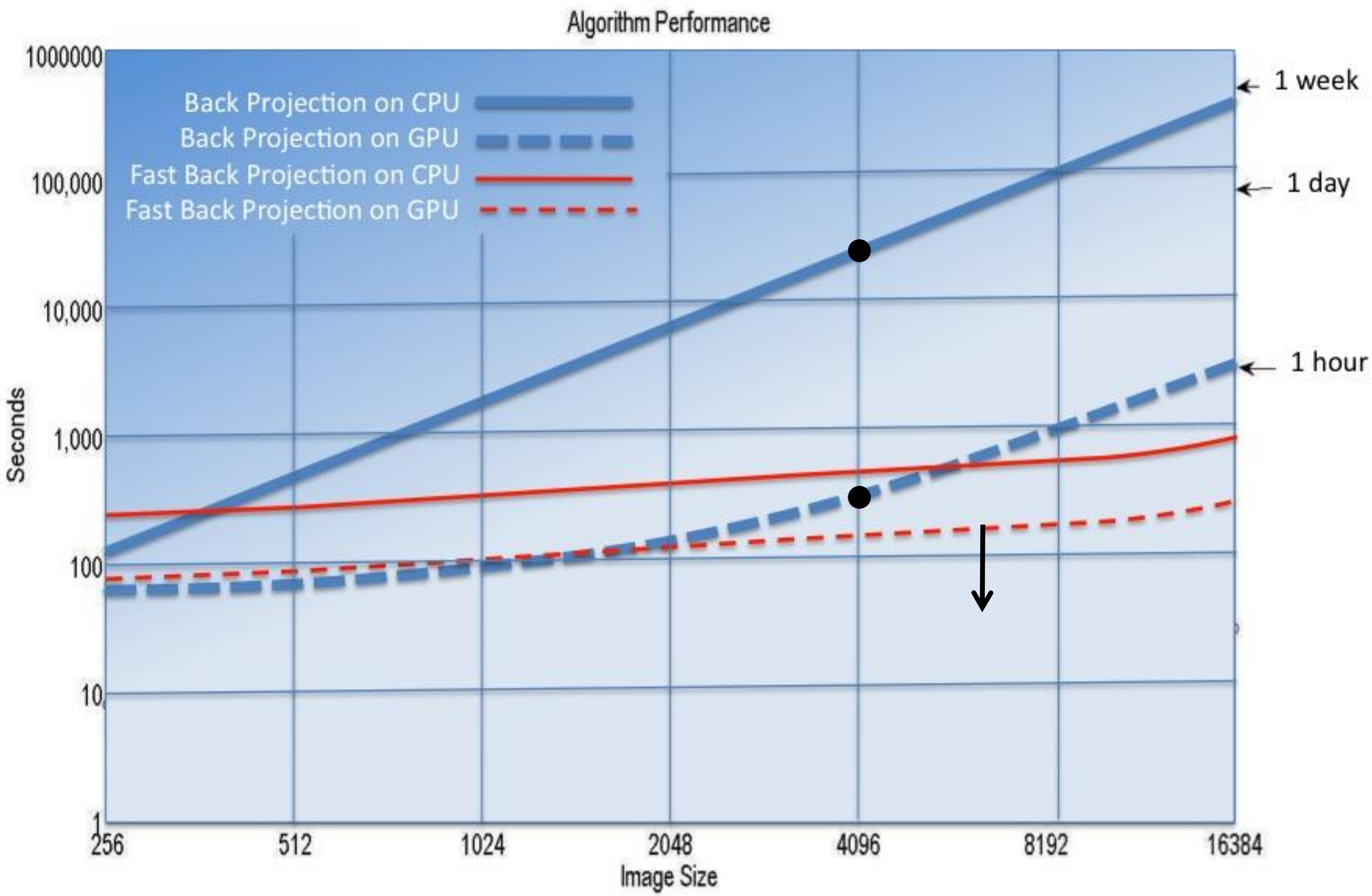
Gotcha
2006
4k x 4k pix
8k pulses
~10 hours
~3º ap.

Image Analysis (GPU BP)

Gotcha
2006
4k x 4k pix
8k pulses
~5.5 mins
~3º ap.

CPU vs. GPU Benchmark

8k pulses
~3º ap.



Future Work

- Optimize GPU memory management
  - Utilize shared memory vs. global memory when possible
  - Coalesce global memory access
  - Eliminate partition camping
- Optimize CPU algorithm
  - Implement multi-CPU algorithm
    - Thread sensor position calculations
- Parallelize more of the FBP algorithm
  - Preliminary results have a ~15x improvement in both interpolation and presumming stages
- Implement multi-GPU solution
- Utilize C2050 and C2070 GPGPUs

Summary

- FBP and standard BP algorithms are "embarrassingly parallel"
- Current parallel approaches are not portable
- GPU solution is highly portable
- GPU solution have tremendous potential
  - Currently naïve approach yields almost 3 orders of magnitude improvement
- GPU results have minor artifacts
- Future changes could further speed up and improve accuracy of GPU approach