Space/Error Tradeoffs for Lossy Wavelet Reconstruction

Jonathan Frain and R. Daniel Bergeron
Computer Science, University of New Hampshire

Multiresolution Data Problem

Represent same data at different resolutions

Each lower resolution

- should take less space
- will introduce additional error into the representation

Research goals

- explore new approaches for generating low resolution data
- quantify space/error tradeoffs

Our approach

3D wavelet transformations produce 8 sets of wavelet coefficients: 1 <u>summary</u>, 7 <u>detail</u>

Summary coefficients can be easily used to approximate original data: ⅛ the size, and some error

Can reconstruct original data with <u>no error</u> using all 7 sets of detail coefficients, but no space savings

Can we use <u>subsets</u> of detail coefficients to get better space/error tradeoffs?

Outline

Wavelet overview

Lossy reconstruction

- Traditional approaches
- Our scheme

Evaluation process

Results

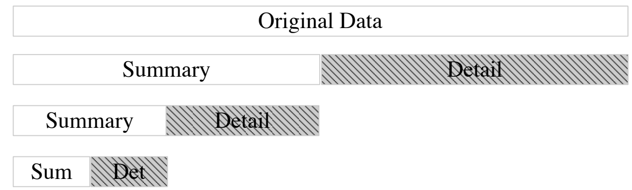
Wavelet Overview

Wavelets provide a powerful tool for multiresolution data generation

1-dimensional wavelet transformation

- N data values map to N/2 summary values and N/2 detail values
  - summary data is the lower resolution representation
  - detail data encapsulates the "error"
- lossless transformation: can reconstruct the original data from the summary and associated detail

Multiresolution 1D Wavelet



Each summary is a coarser representation of previous.

Can reconstruct higher resolution exactly (with some numerical roundoff error) from lower resolution summary and detail.

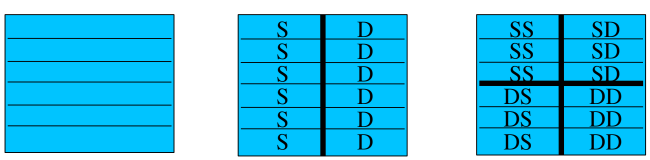
If ignore detail, can reconstruct approximation of higher resolution from lower summary (assuming 0 for detail).

2D Wavelets

Given a 2-D array of input data
- apply 1D wavelet to each row
- apply 1D wavelet to resulting columns

Summary is ¼ input size

3 sets of detail coefficients
- SD: Summary of Detail
- DS: Detail of Summary
- DD: Detail of Detail

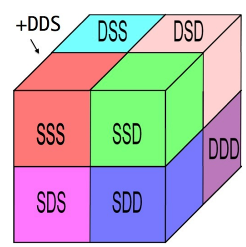


SD: summary of detail

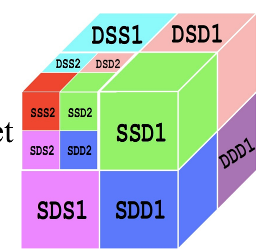
3D Wavelets

Extend 2D: apply 1D wavelets in 3 directions
- across all rows in all planes
- down all columns in all planes
- into all "piles" at each row/column position

Results in:
- summary that is ⅛ size of original
- 7 details: SSD, SDS, SDD, DSS, DSD, DDS, DDD



2-level wavelet: apply 3D wavelet to SSS



Traditional Lossy Reconstruction

Easiest: use only summary; set all details to 0

- low memory; high error

Save *n* highest magnitude coeff, set rest to 0 [Matias et al.]

- need to store coord position along with each wavelet coefficient

Save highest impact coeffients [Sacharidis et al.]

- Tree structure yields more efficient coord position storage
- Extension to 2D and 3D not obvious

Lossy Reconstruction New Approach

Examine influence of the different detail blocks (subbands)

- Do some blocks encapsulate more error than others?
- If so, is there any consistency among different data sets?

Can coefficient *precision reduction* yield better space/error tradeoffs

- Would 4 sets of detail coefficients stored as bytes yield less error than 1 set stored as float?

Precision Reduction

Wavelet detail coefficients

- represent error components
- magnitudes are generally *much smaller* than data values (and summary coefficients)

Try representing them as byte, rather than float

- determine <u>range</u> of a subband of wavelet coefficients
- use a byte to represent a mapping into range
  - linear mapping is simplest
  - software allows an arbitrary “lookup” table to be used

Evaluation Platform

Software that can

- apply Haar wavelet at multiple levels
- compute Haar wavelet reconstruction using arbitrary subset of wavelet coefficient blocks (subbands)
- compute a variety of error measures for reconstructed data
  - Sum absolute point error (SAPE)
  - Sum squared error (SSE) evaluated at each point of original data
  - Root mean squared absolute error (RMSE)
  - Normalized RMSE
  - Max error
  - Average absolute error

Evaluation Process

For a variety of data sets

- Apply 3 levels of Haar wavelet transforms
- For each level, reconstruct data at original resolution
  - use each coefficient block by itself (8 reconstructions)
    - choose block with the lowest error: Best1
  - reconstruct data with Best1 plus each of the other 7
    - choose block pair with lowest error: Best2
  - repeat to get Best3, Best4, Best5, Best6 and Best7
- For each "best" combination, try precision reduction
  - reconstruct / compute error using a byte for detail coefficients
  - reconstruct / compute error using a byte for all coefficients

Evaluation Criteria

Only measured *error* and *space*

Did *not* consider computation costs

- Brute force implementation; no effort to optimize (or even improve) software efficiency
- For very large files in a network environment, communication costs are dominant
- Even local disk I/O is likely to be more critical than CPU time

Error Measures

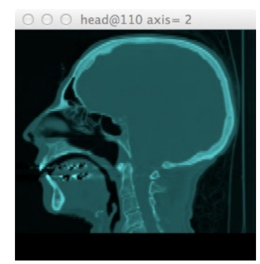
Software computes 6 different error measures

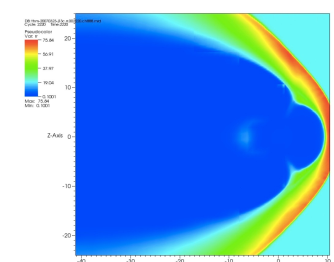
- *Max absolute* best for defining precision needed for guaranteeing minimum error [Woodring et al.], but not very good measure of overall error
- We focused on *sum absolute point error* (SAPE) and *sum squared error* (SSE)
- In practice, want spatial *region*-based error, not one value

Primary Data Sets

Primary data sets

- *cthead* - 3D CAT scan of a head
- *OpenGGCM* - 3D simulation of solar winds as they approach the Earth; used 1 attribute from a typical "interesting" time step





Secondary Data Sets

We also tested other data sets

- Other time steps from OpenGGCM simulation
- Walnut from voreen.org, European Inst of Molecular Imaging
- Stagbeetle from TU Wien, Austria
- Stanford brain

What do we want to learn?

Do the wavelet subbands contribute (significantly) different information to the reconstruction?

- I.e., how much error is introduced for each subband that is ***not*** used in the reconstruction?

If so, are these differences consistent:

- for different wavelet resolution levels for same data set?
- for different data sets?

Can wavelet coefficients be stored at lower precision without significant increase in error?

Summary of (reported) tests

Apply 3 levels of wavelets to *cthead* and *openGGCM*

- report error for each level

Detail Elimination with full precision

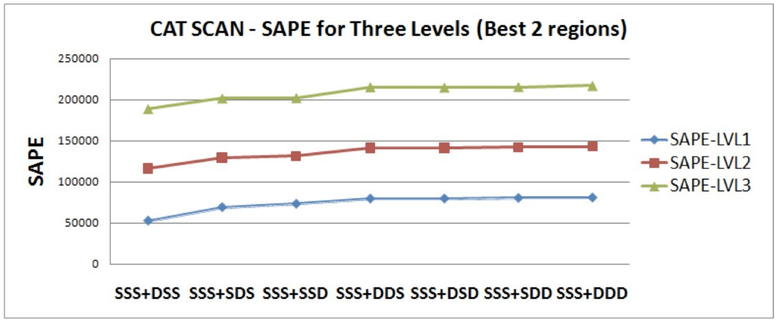
- Order detail coefficients by reconstruction error when using SSS and exactly 1 detail subband (Best 2)
- Identify Best 3, Best 4, …, Best 7 subbands
- Show error reduction for best 1-8 subbands: how much does each subband reduce the error?

Detail elimination with precision reduction (PR)

- Repeat above steps apply PR for details and for all coeff.
- Show space/error tradeoffs

Detail elimination (no PR) Best 2 : Ct Scan

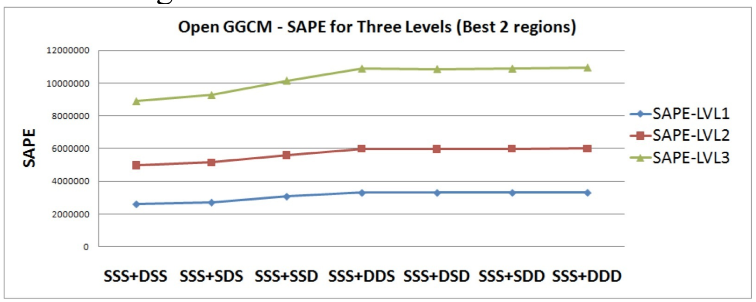
Best wavelet subband is always summary (SSS)
So, looking for "best" detail subband



Subband region order is same for all resolutions:
DSS→SDS→SSD→DDS→DSD→SDD→DDD

Detail elimination (no PR)
Best 2 : OpenGGCM

Best wavelet subband is always summary (SSS)
Looking for “best” detail subband



Subband region order is same for all resolutions:
DSS→SDS→SSD→DDS→DSD→SDD→DDD

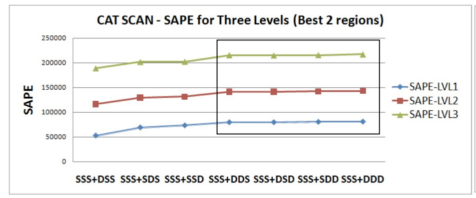
Best 2 Comparison
CT vs GGCM

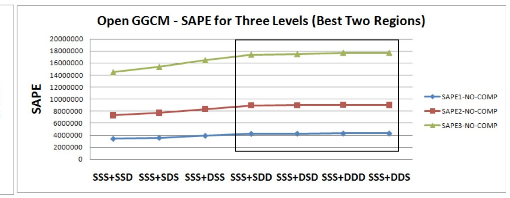
Exact order of regions is same (not always true, though)

CT: DSS→SDS→SSD→DDS→DSD→SDD→DDD

GGCM: DSS→SDS→SSD→DDS→DSD→SDD→DDD

Note that last 4 are barely distinguishable -- order isn’t really relevant.

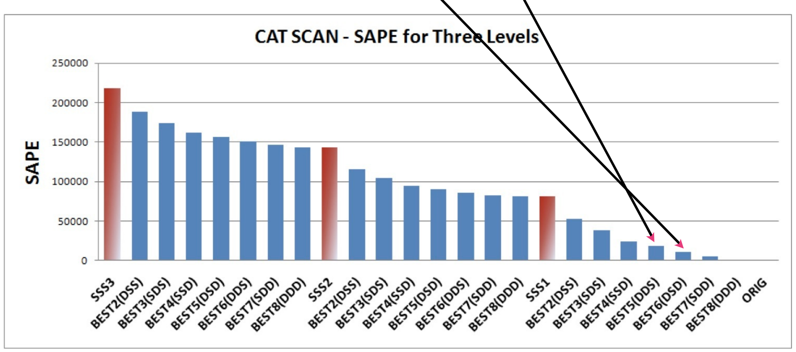




Best 2-8: CT data

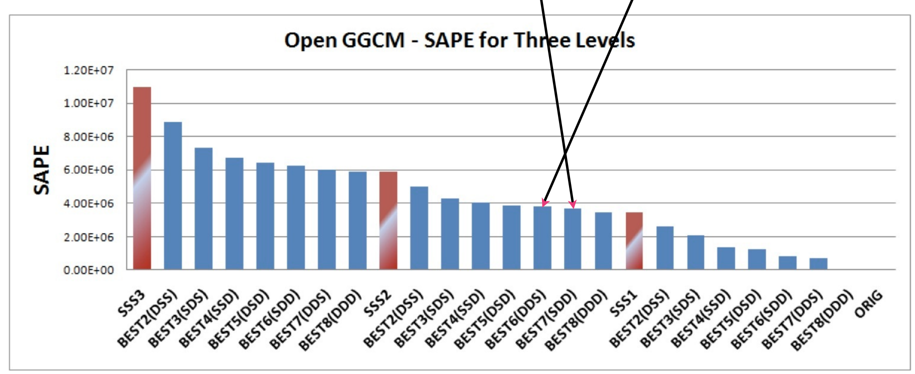
Show the error for <u>best</u> 2, 3, 4 … 8 subbands

First 4 region <u>order</u> is same for all: SSS→DSS→SDS→SSD
Last 4 swap 2 in res 1: →DSD→DDS→SDD→DDD

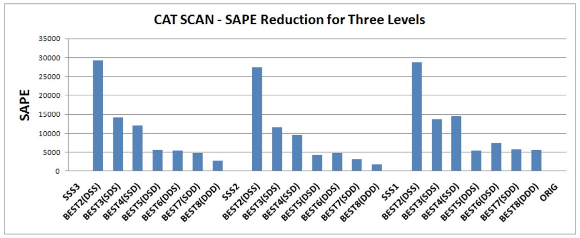


Best 2-8: GGCM data

First 5 region <u>order</u> same for all: SSS→DSS→SDS→SSD→DSD
Last 4: swap 2 in res 2: →SDD→DDS→DDD

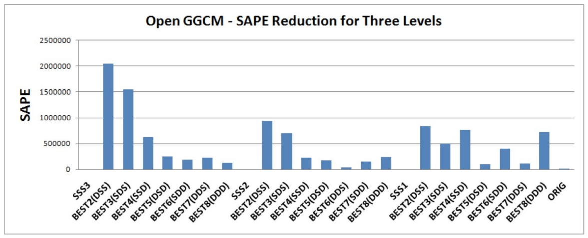


Error reduction by subband



CT: 1st detail subband at each level restores greatest amount of the error; next 2 about the same

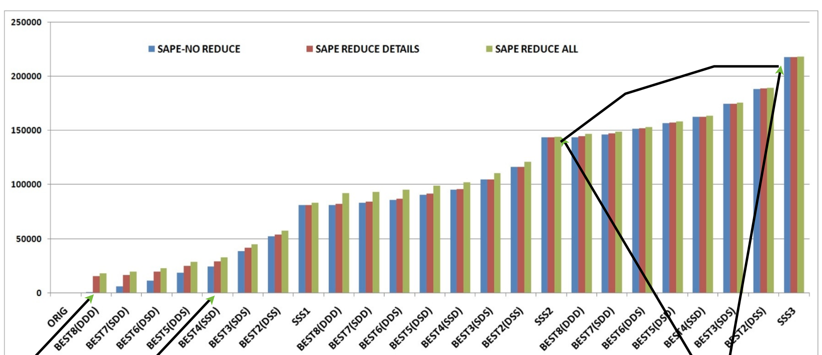
GGCM: 1st detail subband at each level restores greatest amount of the error; not so consistent for rest



Precision Reduction Tests

- Precision reduction approach
  - use 1 byte to represent a coefficient value
  - byte indexes into a table of 256 values
  - linear mapping to the coefficient range
- Compare full precision results with
  - reduced precision of all detail coefficients
  - reduced precision of summary and detail coefficients

PR for CT data



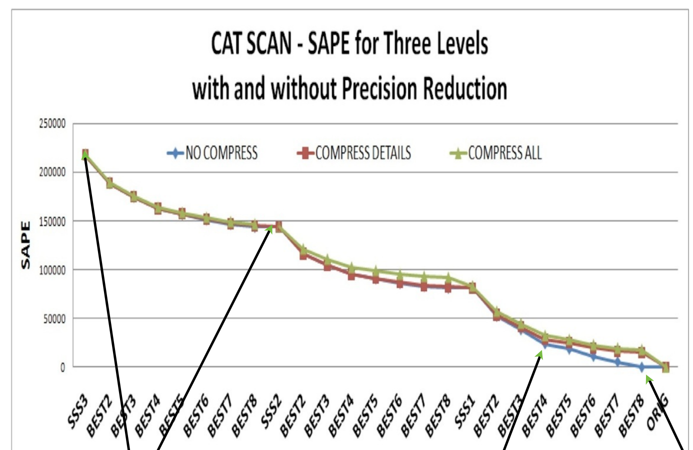
Only PR

½ coeff are 0

Effect of PR is irrelevant compared to resolution error

Missing coeff error quickly dominates precision reduction error

PR for CT data 2

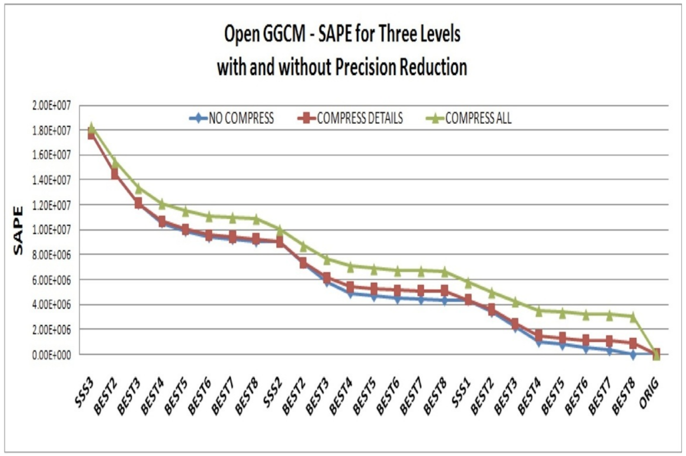


Effect of PR is irrelevant compared to resolution error

½ coeff are 0

Only PR

PR for GGCM data



Missing coeff error quickly dominates precision reduction error

Error vs Space

We have a range of options for representing data

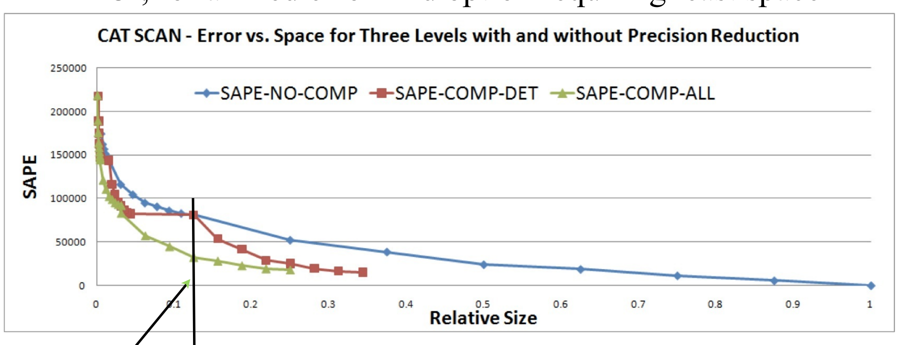
- 3 resolution levels
- 0-7 detail subbands
- without precision reduction, applying precision reduction to the details, and applying precision reduction to all coefficients

Graphs show 24 points representing 3 resolution levels and 8 choices of wavelet subbands -- ordered by increasing size

The 3 precision reduction options are graphed separately

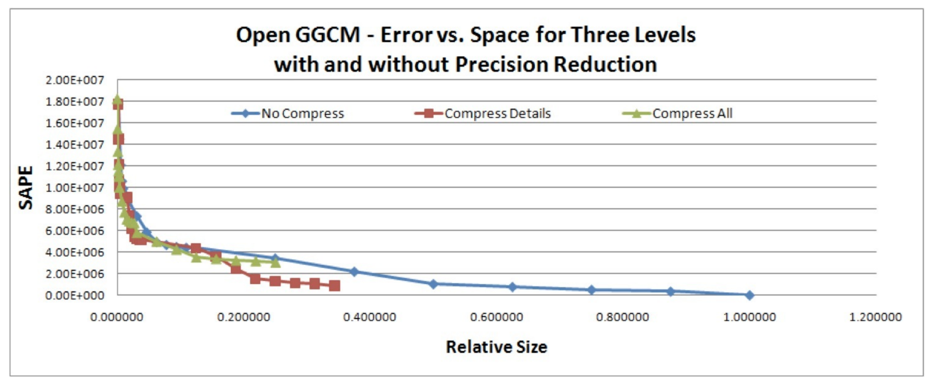
Error vs Space: CT

For a fixed size, find option with least error
Or, for a fixed error find option requiring least space



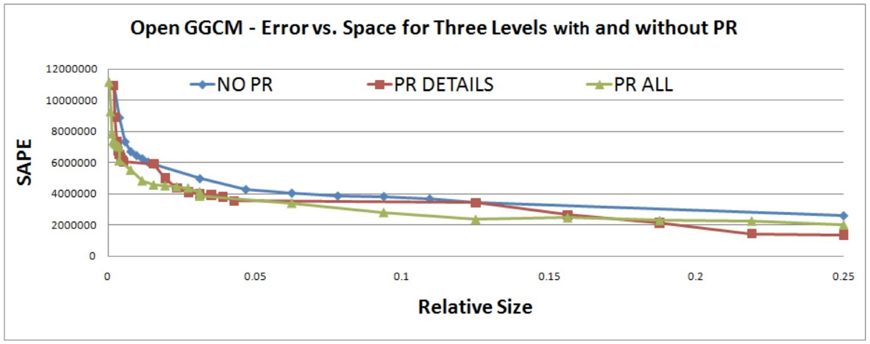
Space ⅛ of original: resolution 3, with 3 detail subbands with all data using reduced precision

Error vs Space: GCCM



Error vs Space: GCCM

With tail truncated



Consistency of Results CT vs GGCM

2-way order: identical

Best2 - Best4 order matches 2-way order

Best5 - Best7 order: some differences with 2-way

- but magnitude of the differences were very small

Precision reduction results very similar except compressing summary more successful for CT data

- may be an artifact of our implementation: CT data is normalized to 0,1

Secondary Data Sets

Stag beetle
- Best2 order: DSS→SDS→SSD→others
- Best3 - Best7: match Best2 order

Stanford brain
- Best2 order: DSS→SDS→SSD→others
- Best3 - Best7: match Best2 order

Walnut
- Best2 order: SDS→SSD→DSS→SDD→others
- Best3 - Best7: Res 2 and 3 reversed DSS and SDD

Two different GGCM time steps
- Consistent with other GGCM results

Observations

Data tests we have run so far are encouraging
- Wavelet subbands aren't all created equal
- Detail elimination results are reasonably consistent
  - Best2 subband ordering is a good approximation to Best3-Best7 order
  - Without any pre-processing, DSS, SDS and SSD are good candidates for first 3 subbands; though order is not always optimal, it's good
- Precision reduction results are reasonably consistent
  - Could provide basis for heuristics

Conclusions

Using selected detail subbands for data reconstruction can provide better space/error tradeoffs than traditional approaches

Representing detail coefficients as bytes can provide significant space savings for only small additional error

Weaknesses

Our precision reduction implementation is naive

- it generates only 1 mapping table for all subbands; there should be a separate table for each subband
- we use a simple linear mapping; a nonlinear mapping might yield lower error

Are our test data sets really representative?

- CT and MR data has pretty low underlying precision (12-14 bits); is that skewing the results?
- all data sets we tested have lots of "empty" space.

A single global measure for error is too limiting

- need to generate and use local error in this context

Future Work

Re-implement precision reduction

Incorporate a local error model as part of detail elimination and precision reduction

Test other data sets

Devise more efficient reconstruction

Incorporate these ideas into an adaptive resolution / multiresolution software framework

Extensions

Adaptive resolution decision (in progress)

- partition space into regions with own retention decision

Evaluation using other wavelet functions

Apply wavelet transform to saved detail blocks

- Save all details at lower res = 1 detail block at same res

Implement more sophisticated error functions

Use existing visualization tool to experiment with error visualization