GPU-Based Large-Scale Scientific Visualization

Johanna Beyer, Harvard University
Markus Hadwiger, KAUST

Course Website:
Part 4 -
Display-Aware Visualization and Processing
MOTIVATION

goal: perform computations at output resolution

250 megapixels < 1 megapixel visible

current resolution: 250 megapixels
< 1 megapixel visible

resolution level 0
resolution level 3
DISPLAY-AWARE IMAGE OPERATIONS

Input Resolution
(level 0)

Output Resolution
(level 3)

Display Region

Compute Resolution
(level 4)

Compute Region
Dyadic image pyramids

- **Mipmaps** [Williams 1983]: texture mapping (standard on GPUs)
- **Gaussian/Laplacian pyramids** [Burt and Adelson 1983]: image processing/compression
Dyadic image pyramids

- **Mipmaps** [Williams 1983]: texture mapping (standard on GPUs)
- **Gaussian/Laplacian pyramids** [Burt and Adelson 1983]: image processing/compression
Dyadic image pyramids

- **Mipmaps** [Williams 1983]: texture mapping (standard on GPUs)
- **Gaussian/Laplacian pyramids** [Burt and Adelson 1983]: image processing/compression
IMAGE PYRAMIDS

Dyadic image pyramids

- **Mipmaps** [Williams 1983]: texture mapping (standard on GPUs)
- **Gaussian/Laplacian pyramids** [Burt and Adelson 1983]: image processing/compression
- **Sparse pdf maps** [Hadwiger et al. 2012]

Laplacian pyramid

level 0

level 1

level 2

level 3
**IMAGE PYRAMIDS**

Dyadic image pyramids

- **Mipmaps** [Williams 1983]: texture mapping (standard on GPUs)
- **Gaussian/Laplacian pyramids** [Burt and Adelson 1983]: image processing/compression
- **Sparse pdf maps** [Hadwiger et al. 2012]

Local Laplacian filtering [Paris et al. 2011]
ANTI-ALIASING IN IMAGE PYRAMIDS

level 0
ANTI-ALIASING IN IMAGE PYRAMIDS

level 0

level 4
ANTI-ALIASING IN IMAGE PYRAMIDS

level 0

level 4
ANTI-ALIASING IN IMAGE PYRAMIDS

level 0

level 4, standard

level 4

level 4 standard
ANTI-ALIASING IN IMAGE PYRAMIDS

level 0

level 4, sparse pdfmaps

level 4, ground truth
NON-LINEAR IMAGE OPERATORS

Apply non-linear operation to each pixel
• Color map or non-linear contrast adjustment
• Bilateral filtering: range weight
• Smoothed local histogram filtering [Kass and Solomon 2010]
• Local Laplacian filtering [Paris et al. 2011]: point-wise, non-linear re-mapping
Compute Laplacian pyramid coefficient
• Adjust local contrast via point-wise non-linearity; then downsample

Same as local color mapping, then downsampling
• Cannot apply the re-mapping function to the downsampled image!
• Need to compute ground truth (pyramid!) or proper “anti-aliasing”
LOCAL LAPLACIAN FILTERING: SCALABILITY

Night Scene Panorama: 47,908 x 7,531 pixels (361 Mpixels)

- Every downsampled pixel results from the entire pyramid above it
- Sparse PDF maps allow direct computation!
Sparse PDF Maps Concept
SPARSE PDF MAPS

Represent distribution of pixel values in footprint in original image
SPARSE PDF MAPS

Represent distribution of pixel values in footprint in original image
SPARSE PDF MAPS

Represent distribution of pixel values in footprint in original image

$pdf_p(r^*)$
SPARSE PDF MAPS

Represent distribution of pixel values in footprint in original image

$pdf_p(r)$

level 2

level 0

convolution

histogram smoothing

estimated pdf
Represent distribution of pixel values in footprint in original image

Apply non-linear operation

\[ E[t_p(X_p)] = \frac{1}{w_p} \int_0^1 t_p(r) pdf_p(r) \, dr \]
EXAMPLE 1: DOWN-SAMPLED IMAGE

\[
E [t_p (X_p)] = \frac{1}{w_p} \int_0^1 t_p(r)pdf_p(r) \, dr
\]

\[t_p(r) = r\]

\[w_p = 1\]
\[ E \left[ t_p(X_p) \right] = \frac{1}{w_p} \int_0^1 t_p(r)pdf_p(r) \, dr \]

\[ t_p(r) = \text{color map} \]

\[ w_p = 1 \]
EXAMPLE 2: COLOR MAPPING

\[ E \left[ t_p \left( X_p \right) \right] = \frac{1}{w_p} \int_{0}^{1} t_p(r)pdf_p(r) \, dr \]

\[ t_p(r) = \text{color map} \]

\[ w_p = 1 \]

plus: bilateral filtering, local Laplacian filtering in linear time, ...
Fast Local Laplacian Filtering
Computation
GREEDY APPROXIMATION: MATCHING PURSUIT

Spatial filter $W$: 5 x 5
1 coefficient chunk
(# coefficients == 1 * # pixels)
GREEDY APPROXIMATION: MATCHING PURSUIT

Spatial filter $W$: 3 x 3
1-3 coefficient chunks
(# coefficients == 1-3 * # pixels)
Data Structure
SPDF MAPS DATA STRUCTURE

\[ V(p_n, r_n) = c_n \]

(index, count)_p

(r_n, c_n)
SPDF MAPS DATA STRUCTURE

conceptual

\[ V(p_n, r_n) = c_n \]

index image

\[ (index, count)_p \]

coefficient image

\[ (r_n, c_n) \]
Display-Aware Gigapixel Image Processing
Out-of-Core Processing

- Divide data into smaller tiles, process each tile independently (e.g., 256x256)
- Image operations are performed only on requested sub-tiles (display-aware)
- Rendering based on tiled data, using GPU-based virtual memory approach
GPU-based virtual memory architecture [Hadwiger et al. 2012]
Results
COLOR MAPPING GIGAPIXEL IMAGES

NASA Blue Marble bathymetry: 21,601 x 10,801 pixels (233 Mpixels)
GIGAPIXEL LOCAL LAPLACIAN FILTERING

original

details reduced

details enhanced
details reduced
VISIBLE HUMAN (512 X 512 X 1884)

original volume
VISIBLE HUMAN (512 X 512 X 1884)

original volume

octree (averaging) →

fine to coarse
VISIBLE HUMAN (512 X 512 X 1884)

sparse pdf volumes →

original volume

octree (averaging) →
BLOOD VESSELS (1024 X 1024 X 1024)

sparse pdf volumes $\rightarrow$

original volume

octree (averaging) $\rightarrow$
SUMMARY

Display-aware processing with flexible new image pyramid (spdf map)
• Consistent, sparse representation of pixel footprint pdfs
Unified evaluation of many important non-linear image operations
• Local Laplacian filtering for gigapixel images

Efficient CUDA implementation
• Pre-computation costly, but only performed once
• Run time storage and computation similar to standard pyramids

Sparse PDF maps for images:
Hadwiger et al., Sparse PDF Maps for Non-Linear Multi-Resolution Image Operations, Siggraph Asia 2012

Sparse PDF volumes for volume rendering:
Sicat et al., Sparse PDF Volumes for Consistent Multi-Resolution Volume Rendering, IEEE Scientific Visualization 2014
GPU-Based Large-Scale Scientific Visualization

Johanna Beyer, Harvard University
Markus Hadwiger, KAUST

Course Website:
Wrap-Up, Summary
LARGE-SCALE VISUALIZATION PIPELINE

Data

Processing

Visualization

Image

Data Pre-Processing
Filtering
Mapping
Rendering

Data Structures
On-Demand Processing
Acceleration Metadata
Ray-Guided Rendering

Scalability

on-demand?
RAY-GUIDED VOLUME RENDERING

- Working set determination on GPU
- Single-pass rendering
- Traversal on GPU
- Virtual texturing
VOLUME RENDERING OF SEGMENTED DATA

- Empty space skipping essential
- Efficient culling is basis for empty space skipping
  - Compact and scalable data structure (to millions of objects)
  - Hierarchical culling algorithm
- Hybrid approaches
  - Image-order vs. object-order
  - Deterministic vs. probabilistic
THANK YOU!

Johanna Beyer, Harvard University
Markus Hadwiger, KAUST

Course Website: