Interactive Visualization and On-Demand Processing of Large Volume Data: A Fully GPU-Based Out-Of-Core Approach.

Jonathan Sarton - Nicolas Courilleau - Yannick Remion - Laurent Lucas

CReSTIC – Université de Reims Champagne-Ardenne – France
ICube – Université de Strasbourg – France

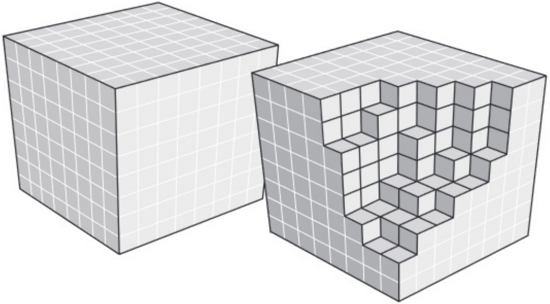
Introduction

Background and motivations

Large volume data, how to

- interactively visualize them
- process them on-the-fly ?

$\rightarrow$ interesting to use **GPUs** !

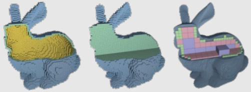
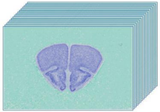
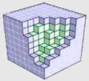


Background and motivations

Large volume data, how to

- interactively visualize them
- process them on-the-fly ?

$\rightarrow$ interesting to use **GPUs** !

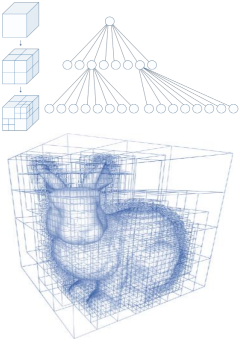
Domain/Application		Data size
Mesh voxelization		~ 100 GB
Histology Electron microscopy		~ 100 GB to several TB
Regular 3D grid		And beyond

Issue : memory occupation

- Large datasets
- $\gg$ GPU and CPU physical memory !
- Interactive manipulation complicated

$\rightarrow$ **Elaborate out-of-core algorithms**

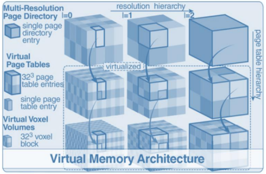
Out-of-core data access

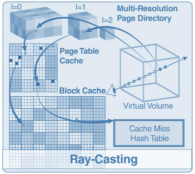


GPU data cache

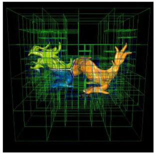
+

Octree Or **Multi-resolution Page Table**

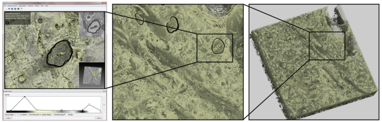




Gigavoxels



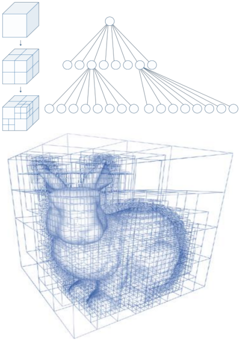
[Crassin et al., ACM SIGGRAPH i3D, 2009]



[Hadwiger et al., IEEE SciVis 2012]

Out-of-core data access

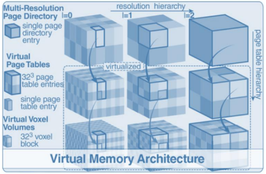
Better for very large volume !!

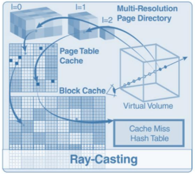


GPU data cache

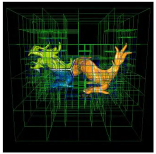
+

Octree **Or** **Multi-resolution Page Table**

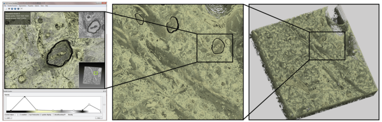




Gigavoxels



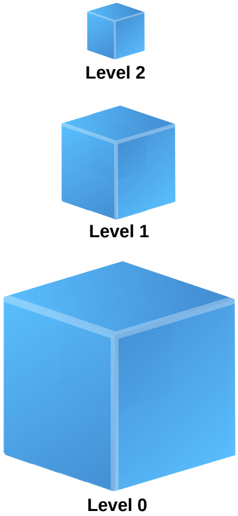
[Crassin et al., ACM SIGGRAPH i3D, 2009]



[Hadwiger et al., IEEE SciVis 2012]

Data representation and storage

- **Multi-resolution**: to choose the desired level of detail
  $\Rightarrow$ Reduces the amount of data



Data representation and storage

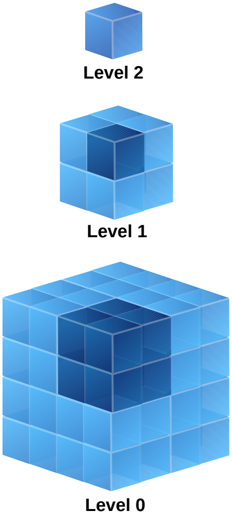
3D mipmap

- **Multi-resolution**: to choose the desired level of detail

  $\Rightarrow$ Reduces the amount of data

- **Bricking**: Volume subdivided into small bricks (e.g 32^3, 64^3).

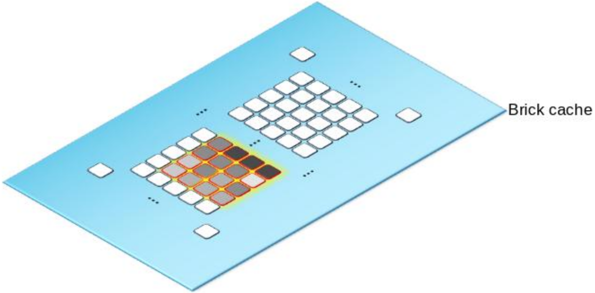
  $\Rightarrow$ Allows the out-of-core approach



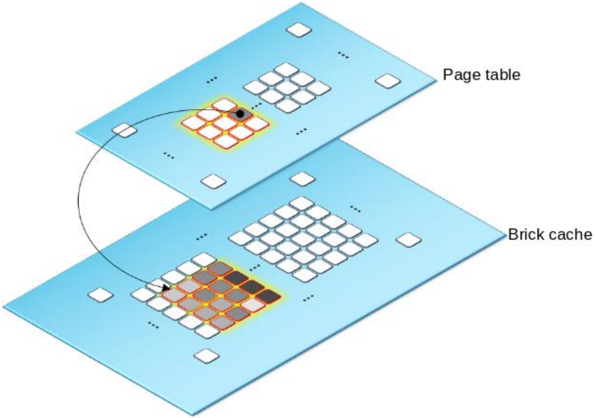
Data compression with LZ4 algorithm

- Loss less
- Good compression ratio
- Real-time decompression

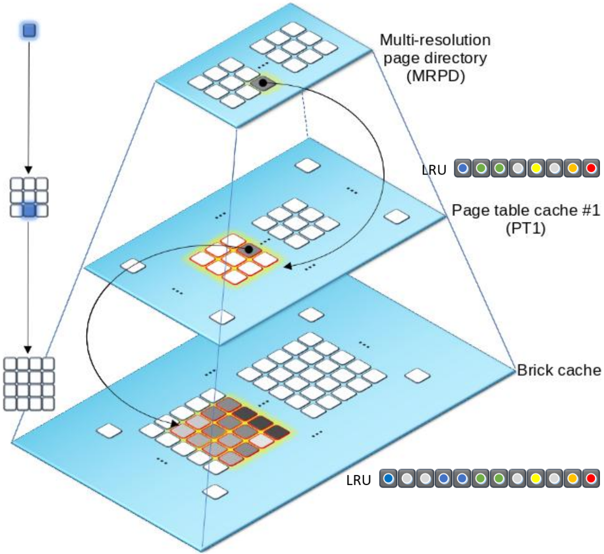
Multi-resolution, multi-level page table hierarchy



Multi-resolution, multi-level page table hierarchy

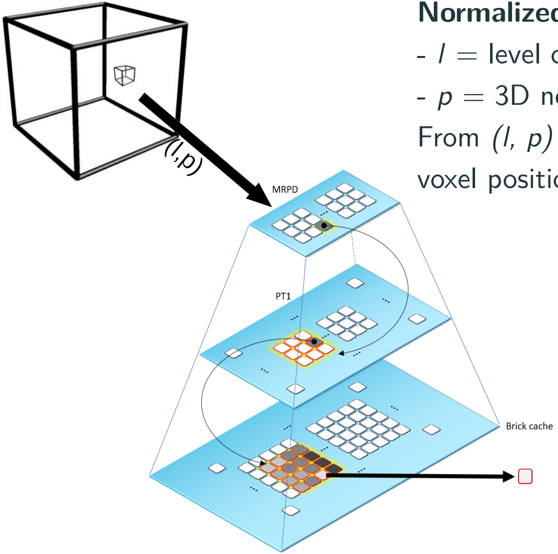


Multi-resolution, multi-level page table hierarchy



- One page = 3D coordinates of the bloc in the next cache level + one flag:
  - Mapped
  - Unmapped
  - Empty
- Implementation: CUDA 3D Textures
- Cache replacement algorithm: Least Recently Used (LRU)

Virtual addressing



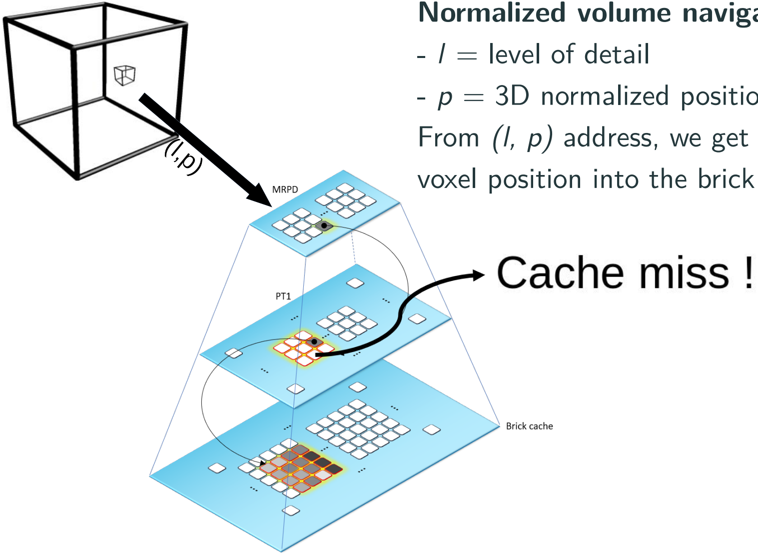
Normalized volume navigation $\rightarrow$ address *(l, p)*

- l = level of detail

- p = 3D normalized position $(x, y, z) \in [0, 1[^3$

From *(l, p)* address, we get the corresponding 3D voxel position into the brick cache.

Cache miss



Normalized volume navigation $\rightarrow$ address *(l, p)*

- l = level of detail

- p = 3D normalized position $(x, y, z) \in [0, 1[^3$

From *(l, p)* address, we get the corresponding 3D voxel position into the brick cache.

Out-of-core data access

How to allow on-demand processing of any part of a large volume during its visualization ?

Cache manager

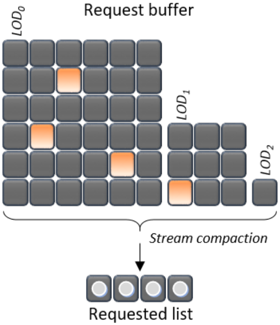
1. Cache usage updates
2. Brick requests management

A GPU data structure fully managed on GPU

Advantages

- Avoids many data transfers between CPU and GPU
- Take advantage of the massively parallel environment of GPUs
- Free the CPU for other eventual processing

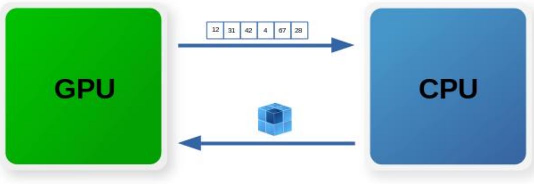
Brick request management on GPU



- Size = number of bricks in the **multi-resolution volume**
- Marked with a timestamp

CPU / GPU transfer

GPU $\rightarrow$ CPU communications
A simple list with the requested brick IDs



GPU $\leftarrow$ CPU communications
Only the bricks ! (With CUDA Zero Copy)

Model in action: interactive visualization & on-demand processing on GPU

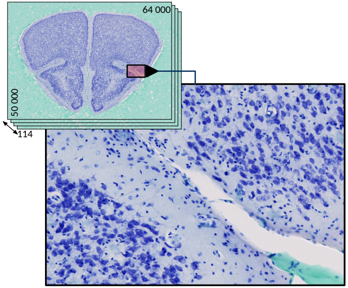
Out-of-core virtual miscroscope

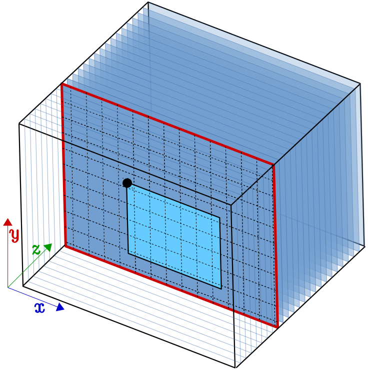
Virtual miscroscope

2D multi-resolution visualization of a high resolution image stack.

Interactive navigation:

- move and zoom in a slide
- navigate through the volume from slide to slide





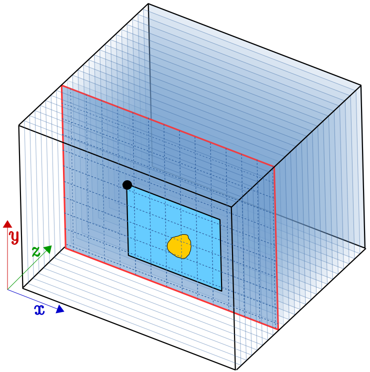
Out-of-core virtual miscroscope

Virtual miscroscope ...

2D multi-resolution visualization of a high resolution image stack.

Interactive navigation:

- move and zoom in a slide
- navigate through the volume from slide to slide



+ on-demand processing

Region-growing from a voxel selected by the user in the screen space

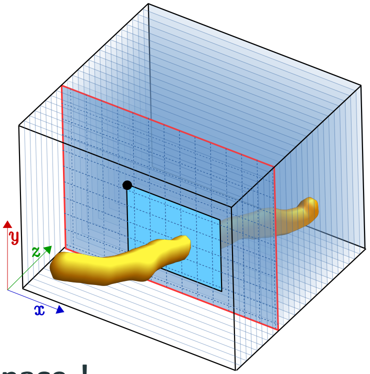
Out-of-core virtual miscroscope

Virtual miscroscope ...

2D multi-resolution visualization of a high resolution image stack.

Interactive navigation:

- move and zoom in a slide
- navigate through the volume from slide to slide



+ on-demand processing

Region-growing from a voxel selected by the user in the screen space

Cache miss due to processing outside the screen space !

Out-of-core virtual miscroscope

Electron micorsocpy dataset

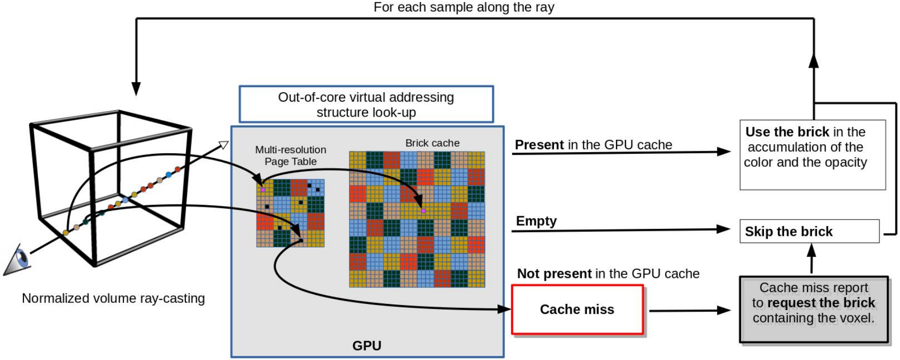
$4096 \times 3072 \times 2130$ 8bits $\approx$ **27 GB**

Rendering performance: $\approx$ **250 FPS**

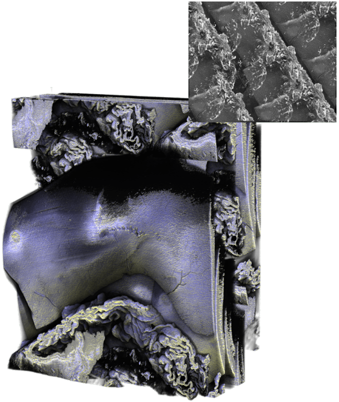
Out-of-core Direct Volume Rendering

Ray-guided approach

- Intuitive visibility selection: no additional culling calculation
- Intuitive out-of-core integration: only load visible bricks on GPU cache



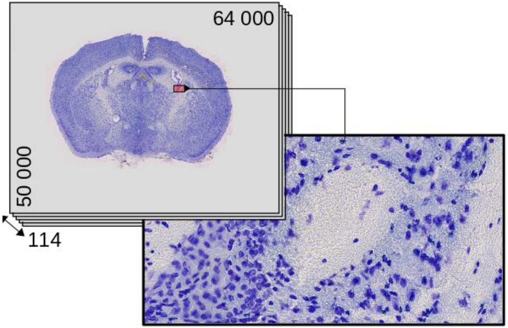
Datasets



Primate hippocampus

Light sheet microscope

$2160 \times 2560 \times 1072$ 16bits $\approx$ **12 GB**



Mouse brain

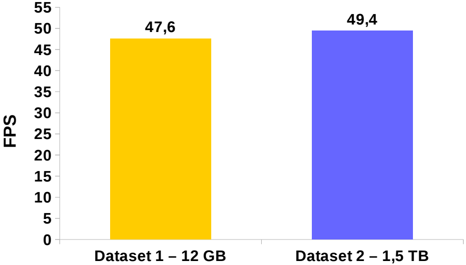
Histological scanner

$64000 \times 50000 \times 114$ RGBA $\approx$ **1.5 TB**

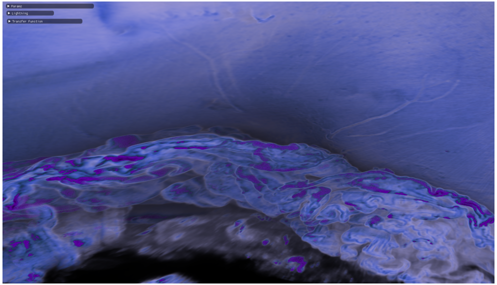
Performances – frames frequency

On a single workstation

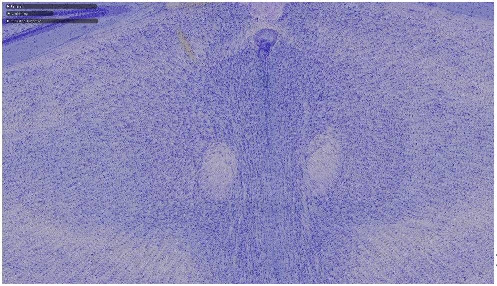
NVidia GeForce Titan X



Primate hippocampus



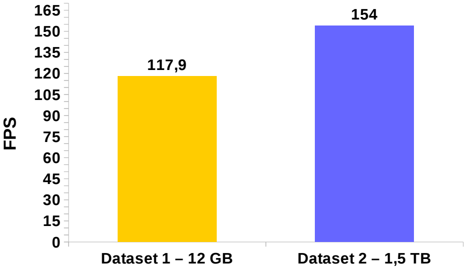
Mouse brain



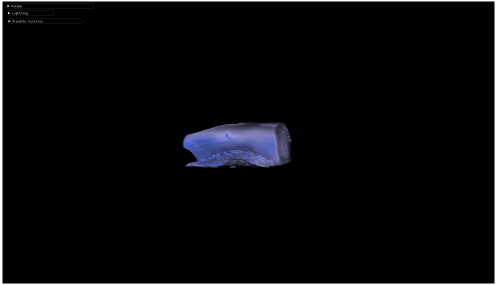
Performances – frames frequency

On a single workstation

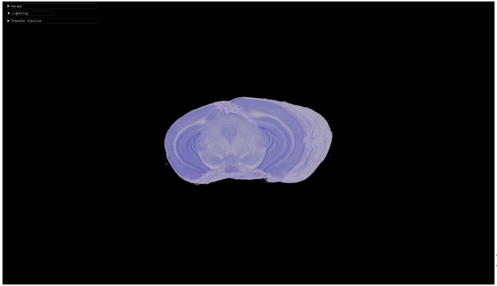
NVidia GeForce Titan X



Primate hippocampus



Mouse brain



Memory occupancy

- Primate hippocampus ($2160 \times 2560 \times 1072 \approx$ **12 GB**)
  - Brick size: $64^3 \Longrightarrow \approx 27000$ bricks (7 LOD)
  - One virtualization level
    $\rightarrow$ Need **1.2 MB** on GPU
- Mouse brain ($64000 \times 50000 \times 114 \approx$ **1.5 TB**)
  - Brick size: $64^3 \Longrightarrow 3.13$ million bricks (10 LOD)
  - One virtualization level $\rightarrow \approx$ **63 MB** needed on GPU
  - Two virtualization levels $\rightarrow \approx$ **13 MB** needed on GPU

Conclusion

Conclusion

- Out-of-core data management: multi-resolution multi-level page table hierarchy
  - Entirely managed on GPU
  - GPU – CPU communication reduced
  - Good rendering frequency even for very large volume of data ($>$ TB)
  - Weak GPU memory and computational footprint
  - General purpose context : interactive visualization & on-demand processing

VIS2019

Interactive Visualization and On-Demand Processing of Large Volume Data: A Fully GPU-Based Out-Of-Core Approach.

Jonathan Sarton - sarton@unistra.fr

Nicolas Courilleau - nicolas.courilleau@neoxia.com

Yannick Remion - yannick.remion@univ-reims.fr

Laurent Lucas - laurent.lucas@univ-reims.fr

Université de Reims Champagne-Ardenne

3DNEUROSECURE calcul intensif et simulation numérique

Université de Strasbourg