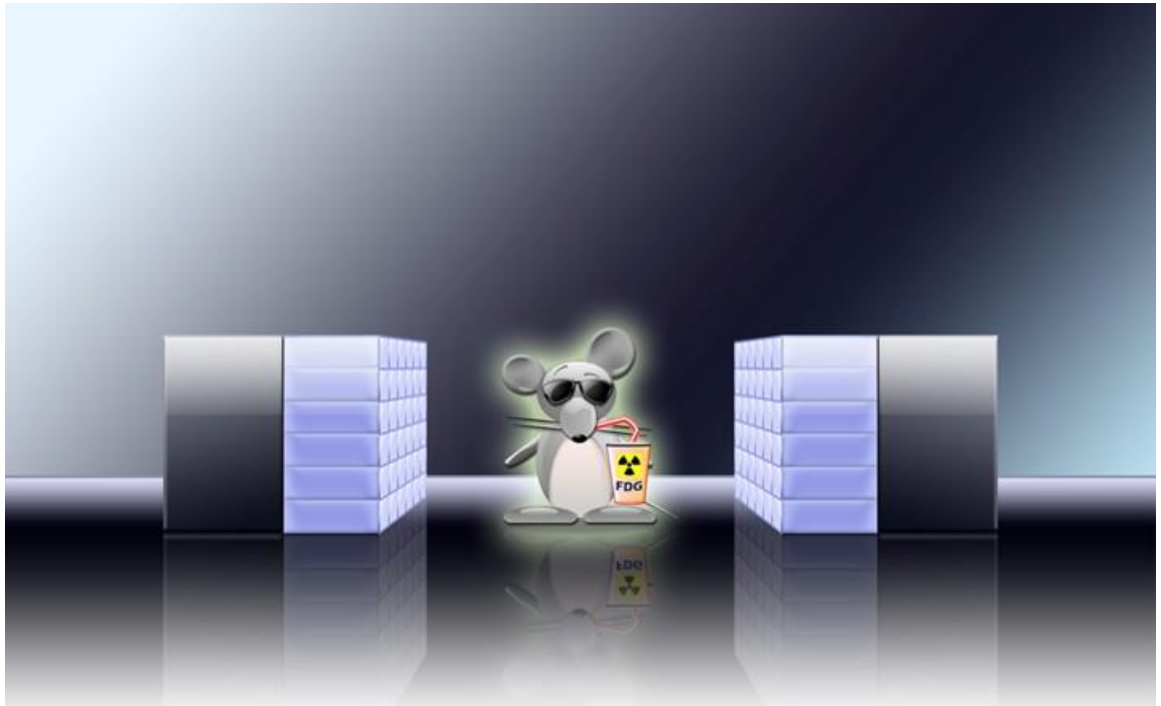
Application of commercial Graphics Processor Units (GPU) to image reconstruction in Medical Physics

Carmela Luongo
Summer School Seminar
Pisa, July 25, 2017

Outline

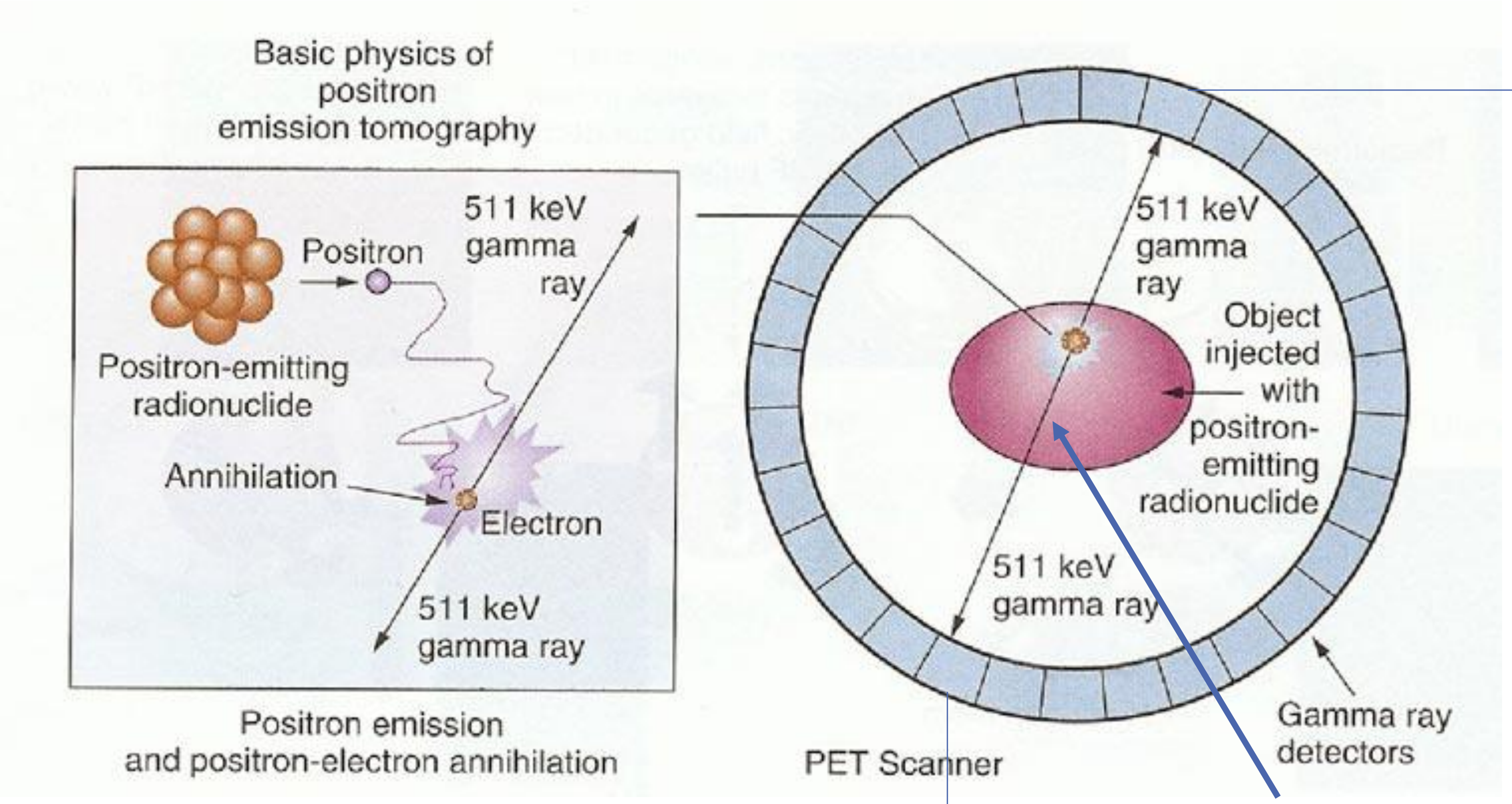
- PET system
- Image Reconstruction
- Implementation
- Quality analysis
- CPU vs GPU approach



Positron Emission Tomography (PET)

- PET is a molecular imaging technique that uses radiolabeled molecules to image molecular interactions of biological processes in vivo
- PET imaging can measure the spatial distribution of active functional processes in living tissue
- Emission imaging
- Functional imaging

Physics in PET



Coincident events

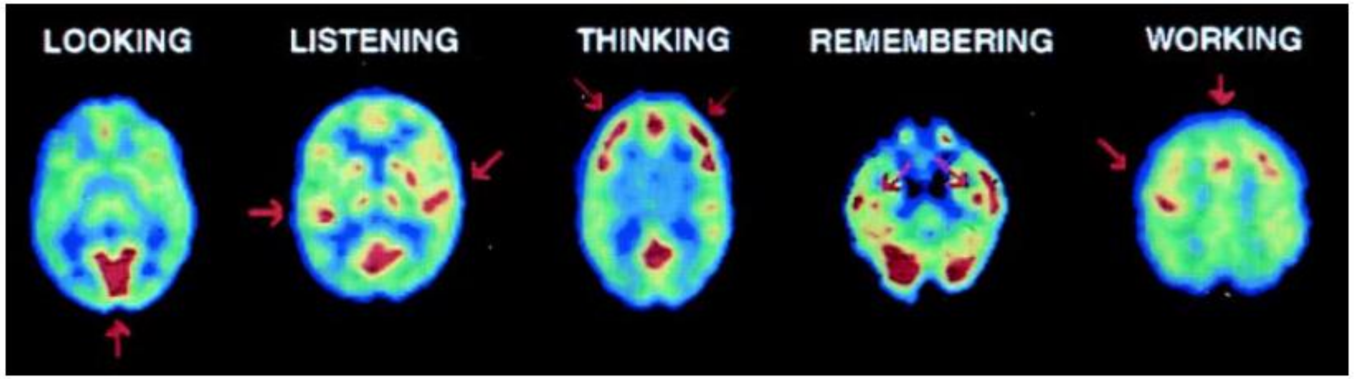
Line of Response (LOR)

IMAGE RECONSTRUCTION

Tomography

- The word *tomography* is derived from Greek:
  - *Tomos* -> section, slice, cut
  - *Graphō* -> to write
- Cross-sectional images of the radiotracer distribution in the subject
- Insight of the physiology and pathology of the patient

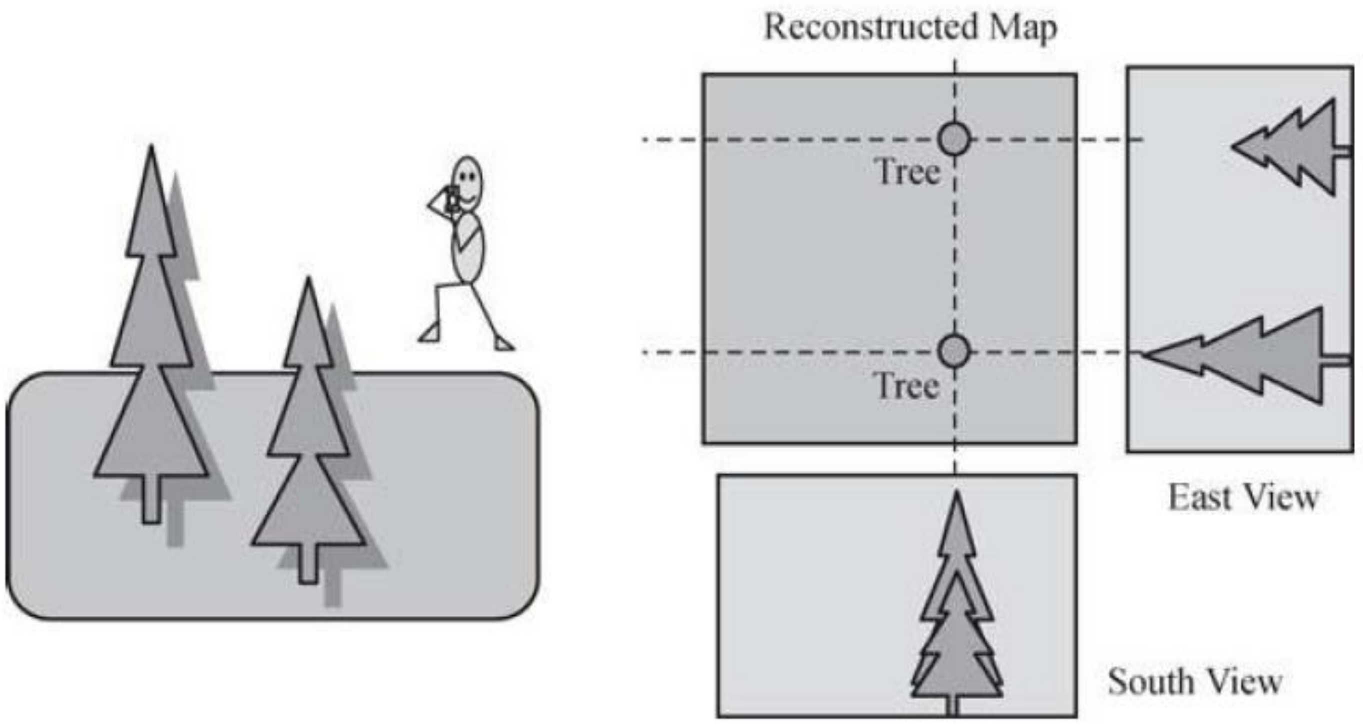
Cutting open to see what is inside



A. Del Guerra, N. Belcari, and M. Bisogni. Positron Emission Tomography: Its 65 years. Nuovo Cimento Rivista Serie, 39:155–223, April 2016.

Basic idea: projections

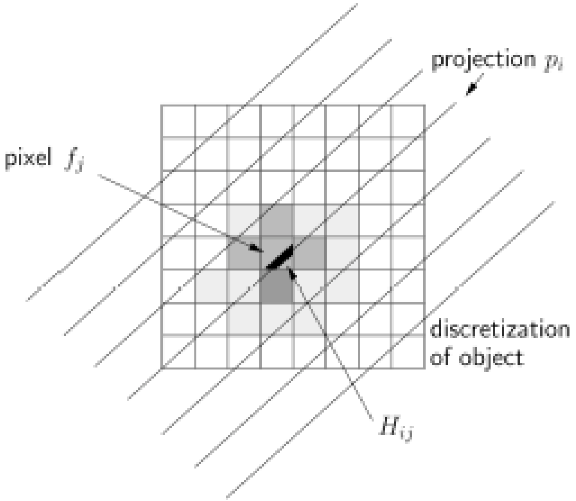
- Two trees in a park
- Make two pictures from east and south
- Try to create a map of the park



A photo is a **projection** of an object onto a plane

Iterative Image Reconstruction

- The image is discretized in voxels
- Image reconstruction can be obtained by solving a system of linear equations



$$P = AX \quad \Rightarrow \quad X = A^{-1}P$$

If the inverse matrix of A exists

P: projections
X: reconstructed image
A: coefficient matrix of the system (*System Response Matrix*)

System Response Matrix

- A is not square in general -> generalized inverse

- A is huge and cannot be inverted!

A: $N_{LOR} \times N_{Image\ voxels}$

- 10^7 LORs $\times$ 10^6 Pixels
- 4 bytes elements $\Rightarrow$

4×10^{13} bytes $\approx$ 40000 GB

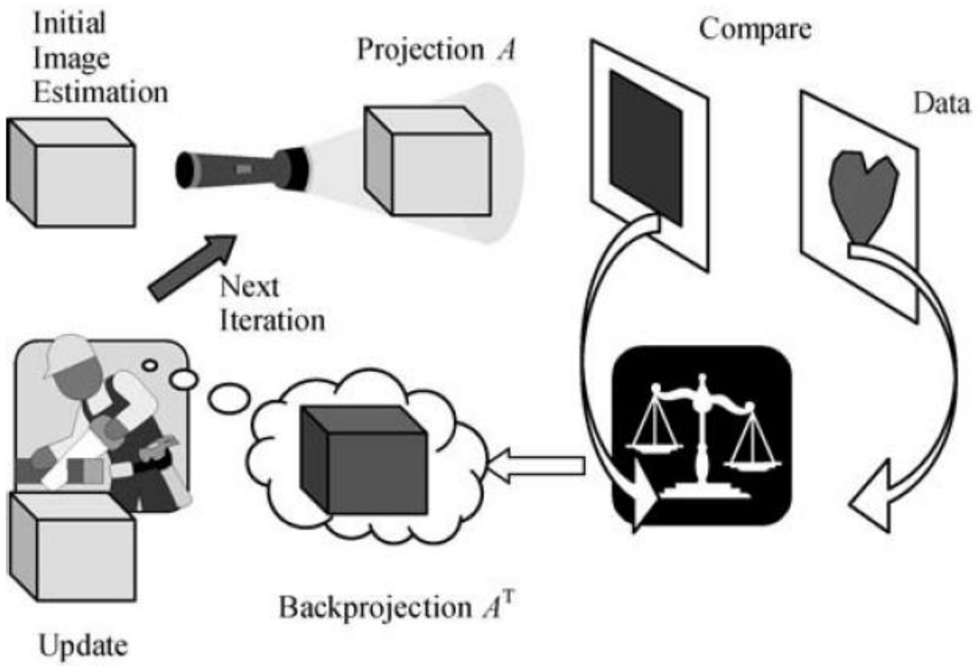
- The discrete system is *ill-conditioned*
  → the solution is unstable for small
  perturbations of the data

➡ Solve inverse problem iteratively

Iterative Algorithms

- The reconstruction problem is solved iteratively

- Objective function
- Optimization algorithm
- System Response Matrix (SRM)



Maximum Likelihood Expectation Maximization (ML-EM)

- Hypothesis: measurements are independent random variables (**Poisson**)
- Idea: find the activity values which maximise the probability of the measured values -> **likelihood function**
- Maximum Likelihood = maximizes the likelihood function
  - Maximum = the image generating the measured data (LOR)
- Expectation Maximization: iterative algorithm to find the ML estimate
- Needs many iterations in order to obtain good images (30-100)

ML-EM

$$\boldsymbol{X^{k+1} = X^k \cdot \frac{1}{A^T 1} \cdot \left(A^T \cdot \frac{p}{AX^k} \right)}$$

Sensitivity

Backprojection

Forward projection

1. Initial guess for the image (uniform)
2. Forward projection: simulate measurements from estimate
3. Compare projections with measured data (ratio)
4. Back projection: improve image estimate
5. Update image weighted by sensitivity
6. Repeat until convergence

What is A?

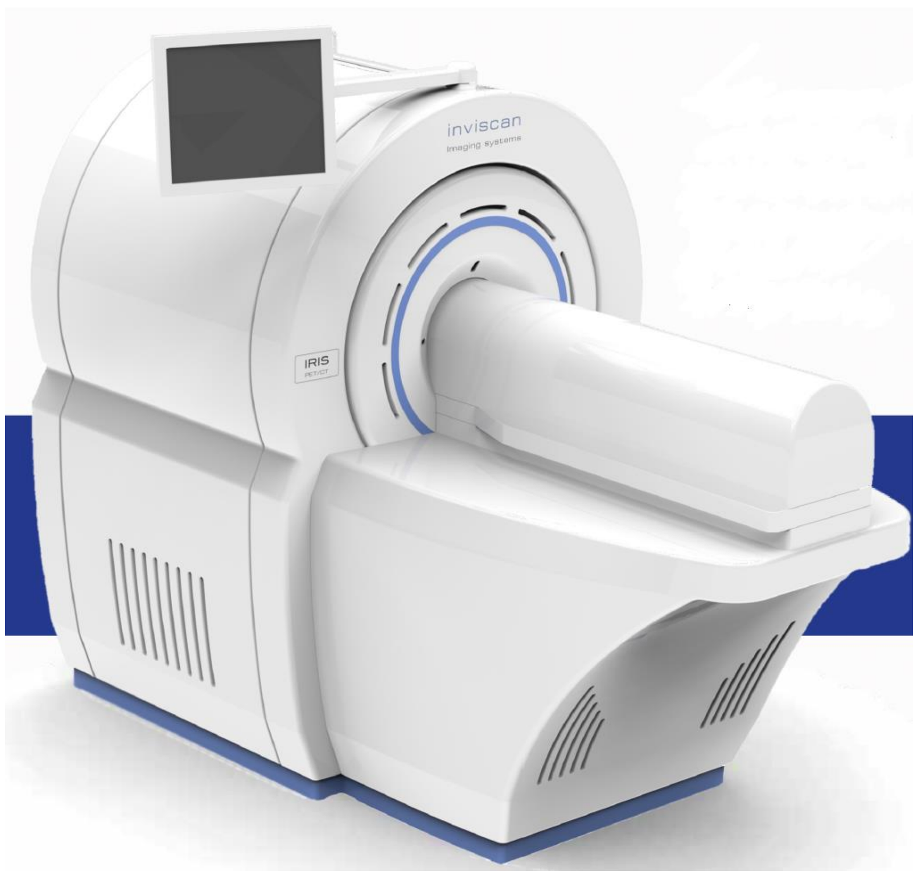
- A_{ij}: probability that a photon pair emitted in the voxel j is detected in the LOR i

- A is scanner dependent

- A is physics dependent → the more physics the better the reconstruction will be → too big model

IMPLEMENTATION

SCANNER IRIS PET



- PET-CT tomography
- 16 detectors arranged on 2 octagonal rings
  - PMT optically coupled with a segmented LYSO of 27x26 (702) crystals
- Coincidence scheme 1 vs 6 detectors → 48 detector pairs
- Number of LORs = $702^2 \cdot 48 \sim 24$ million
- Number of FOV pixels = $101 \cdot 101 \cdot 120 \sim 10^6$
- Crystal size = $1.6 \times 1.6 \times 12\ \mathrm{mm}^3$
- Crystal pitch = $1.7\ \mathrm{mm}$
- Axial FOV = $95\ \mathrm{mm}$
- Ring diameter = $110\ \mathrm{mm}$

CPU implementation

- Ray-tracing → Siddon algorithm
- SRM pre-calculated and stored on disk
- Limited by RAM
- Reduce the number of redundant LOR → Symmetries
- Slow reconstruction
- Dedicated workstation to perform the reconstruction

GPU solution

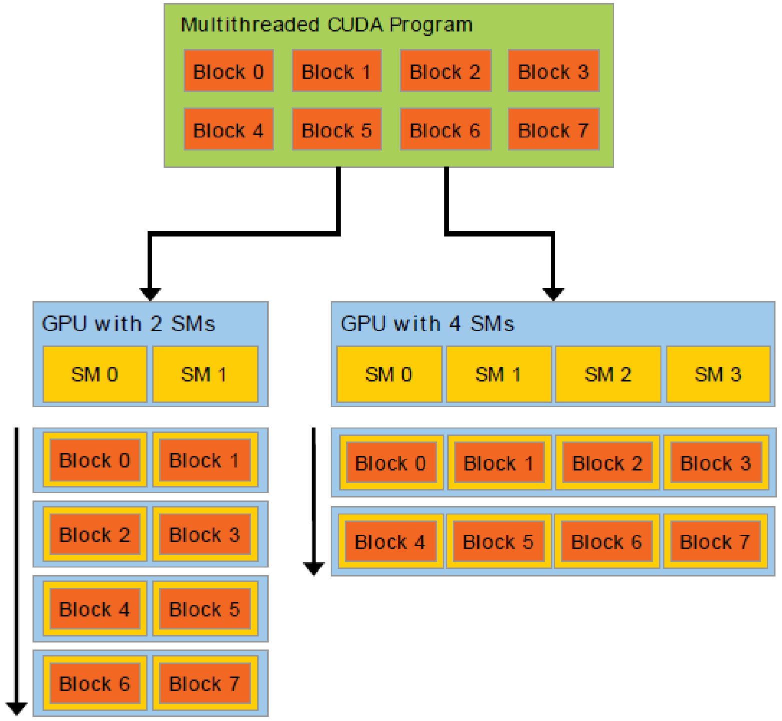
- **Graphics Processing Unit (GPU):** SRM calculated on-the-fly

- **Forward projection** accumulates image data along projective lines
  - Line-projection operations are independent → inherent parallel nature
- **Back projection** distributes projection values back into the image data along the same lines

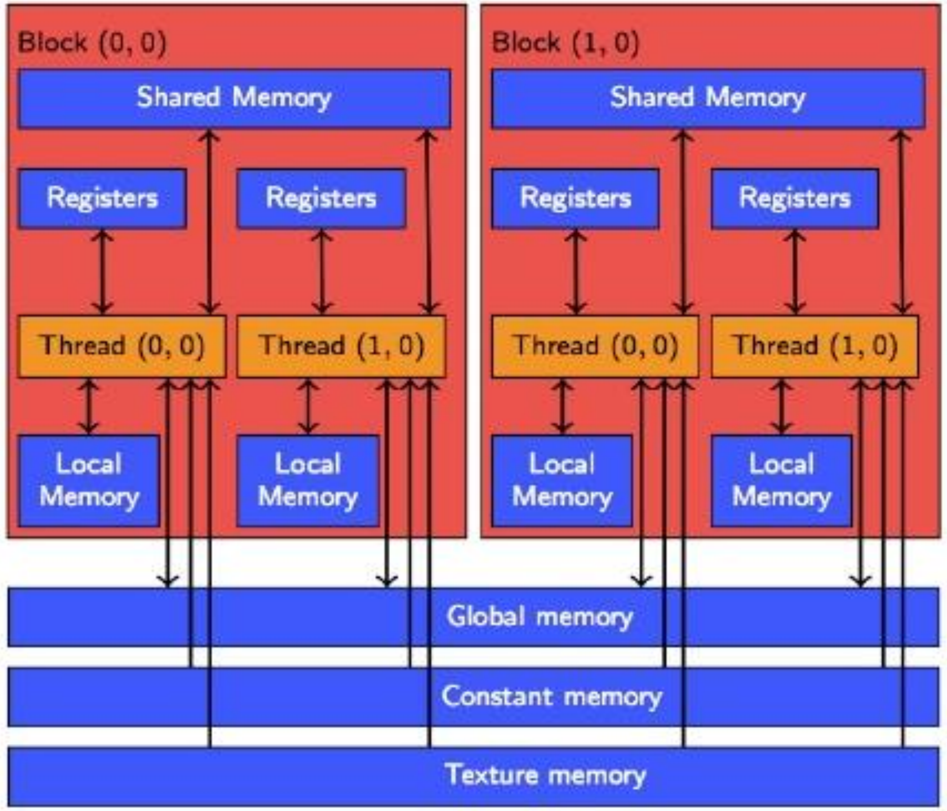
- NVIDIA CUDA architecture – CUDA C

CUDA

- Software environment that allows developers to use C as a high-level programming language
- **Host**: CPU and system's memory
- **Device**: GPU and its memory
- **CUDA-C → kernels → threads → blocks**
- **Streaming Multiprocessors (SMs) → warps**
- **Compute Capability**



Device Memory Hierarchy

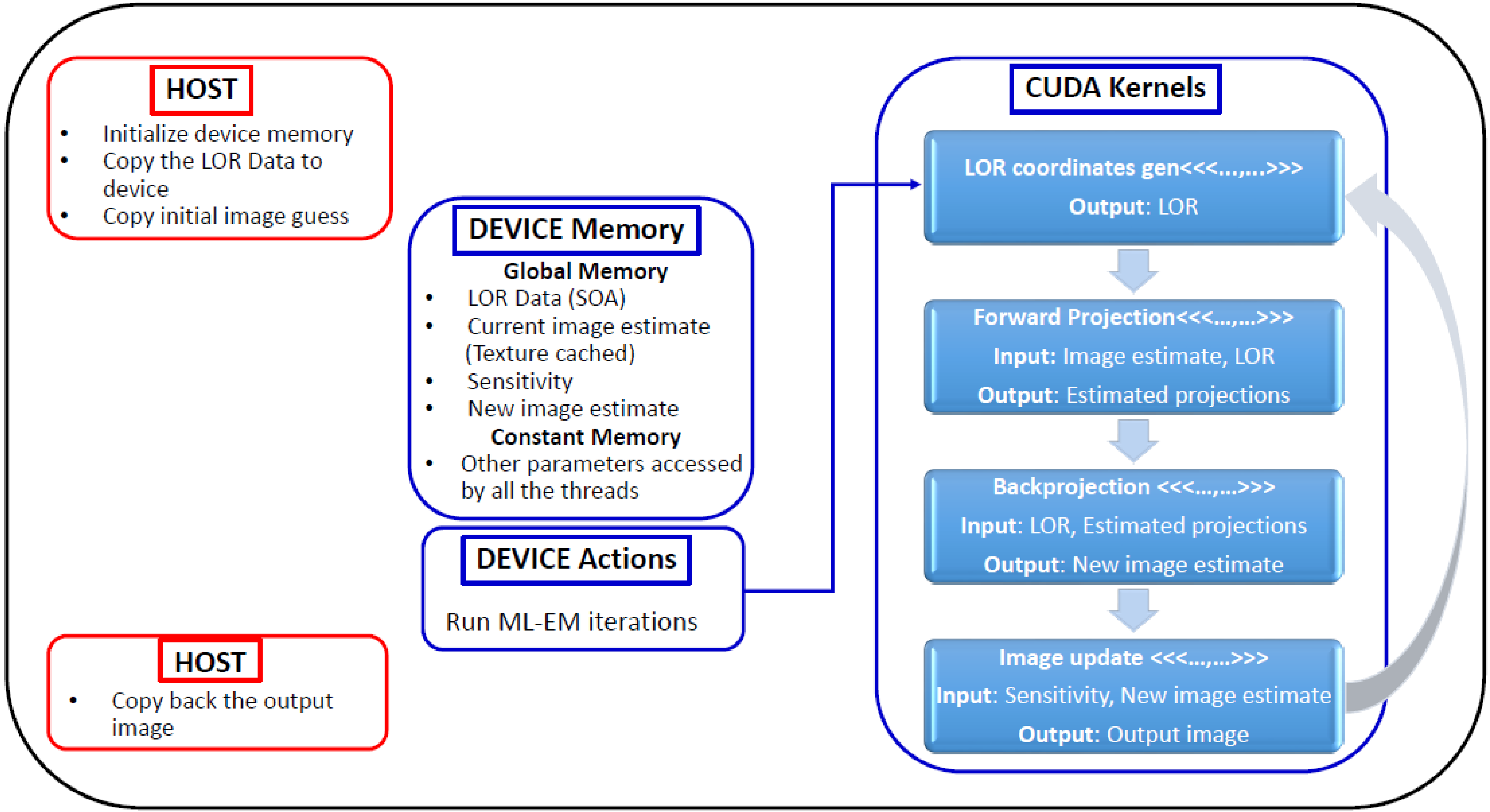


- Registers
- Shared Memory
- Global Memory
- Costant Memory (read-only)
- Texture Memory (read-only)

GPU implementation

- Parallelization of each stage of the EM iteration (4 CUDA kernels)
- Two main kernels:
  - **Line-driven Forward Projection**
  - Forward projections of LORs are independent from each other
  - Each thread in the thread block processes a line independently
  - Each thread computes the sum of all activity along one projection path
  - **Line-driven Back Projection**
  - Each thread re-distributes the activity back to its original path
  - Different pattern of access to memory
  - Race condition problems → atomic operations

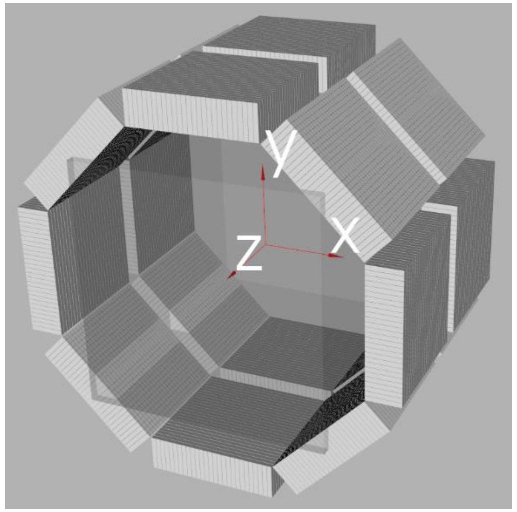
Layout of the algorithm

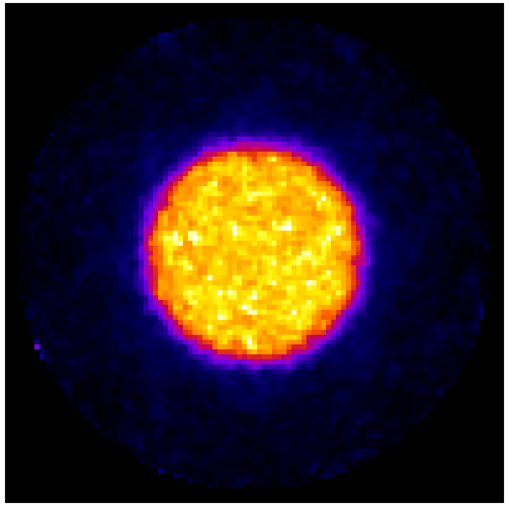


RESULTS

Preliminary results

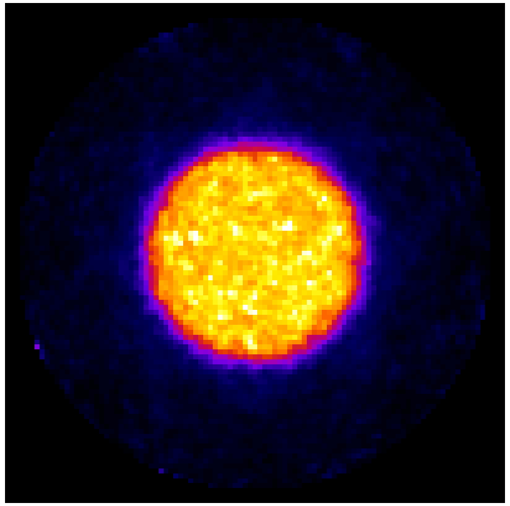
- Monte Carlo GATE for PET
- Cylinder filled with uniform Fludeoxyglucose (FDG) solution
- 40 mm diameter – 24 mm height





CPU reconstruction

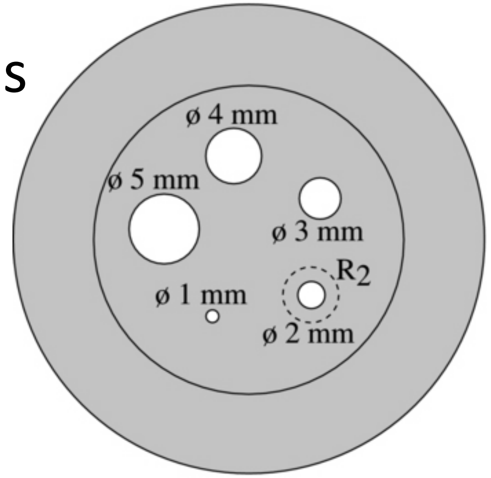
% difference between the two images is 10^{-4}



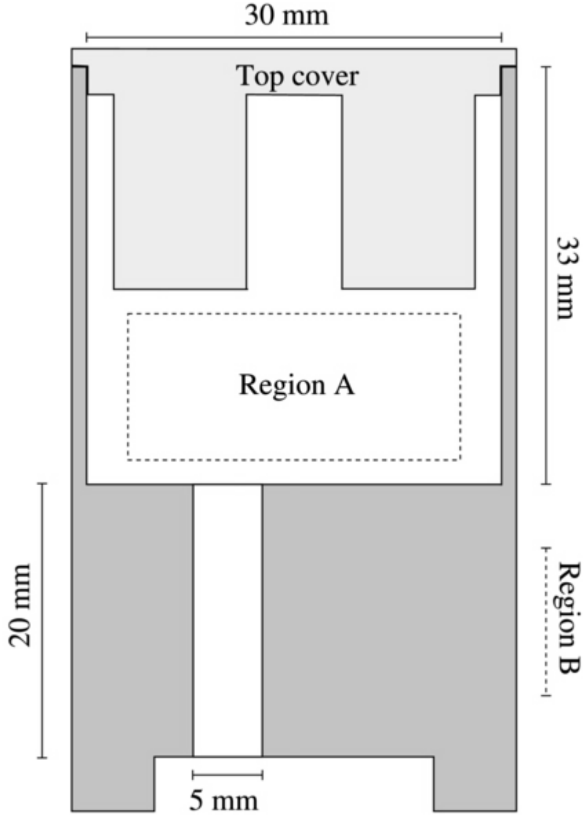
GPU reconstruction

Image Quality

- National Electrical Manufacturers Association (NEMA) standard
  - For small animal imaging
- Image quality phantom
- Simulated acquisition of 20 minutes



Transverse section

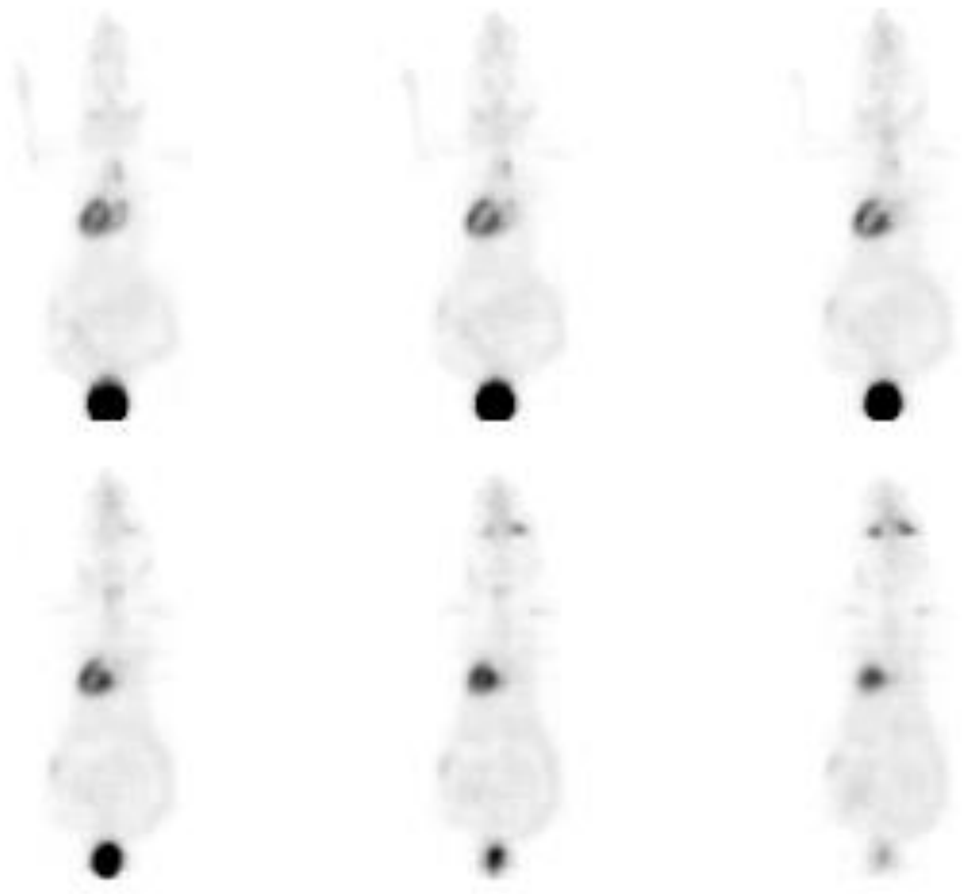


Coronal section

Animal Studies

- A 33 g mouse injected with ^{18}F-FDG
- Scan 60 minutes after FDG injection
- 30 iterations of the reconstruction algorithm
- 6 consecutive horizontal slices of the image
- Bladder and heart are well visible

Courtesy of Dr. Piero Salvadori and Dr. Daniele Panetta, Istituto di Fisiologia Clinica (IFC) Pisa



BENCHMARK

Hardware

- **CPU Intel i7 3770 – 3.4 GHz – 8 cores**

- **GeForce GTX TITAN (Kepler)**
  - Compute Capability 3.5
  - 14 Multiprocessors
- **GeForce GTX980 Ti (Maxwell)**
  - Compute Capability 5.2
  - 22 Multiprocessors

- Max dimension size of a thread block (x, y, z): (1024, 1024, 64)
- Max number of threads per block: 1024
- Total amount of shared memory per block: 49152 bytes
- Total number of registers per block: 65536

Performance: benchmark NEMA quality

- Number of rays per iteration $\approx 13 \times 10^8$

ML-EM	CPU	GPU Kepler	GPU Maxwell
Sensitivity	235 sec	94 sec	49 sec
Forward Projection	226 sec	28 sec	17 sec
Back Projection		83 sec	45 sec

GPU achieves speed up factor of **5** with respect to CPU

GPU **3.5** times faster than CPU

Conclusions

- We implemented an iterative algorithm for PET image reconstruction on GPU
- This computing application fits the capabilities of massively parallel architectures like GPUs
- The GPU’s advanced capabilities were originally used primarily for 3D game rendering
- Now those capabilities are being harnessed more broadly to rapidly solve large problems having substantial inherent parallel nature, such as image reconstruction
- We made a comparison with an existing CPU implementation with respect to image quality and processing time
  - The reconstructed images are "identical"
  - GPU implementation shows speed up factor of 5 with respect to CPU implementation