Julian Merten

New ideas in lensing reconstructions

INAF - Osservatorio Astronomico di Bologna
Institut für Theoretische Astrophysik
Zentrum für Astronomie
Universität Heidelberg

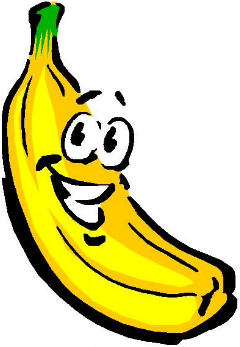
March 4th, 2010

with:

Massimo, Lauro, Barbara, Paolo
Matthias Bartelmann, Ralf Klessen, Peter Melchior, Massimo Viola
and hopefully at least with two more PhD students soon

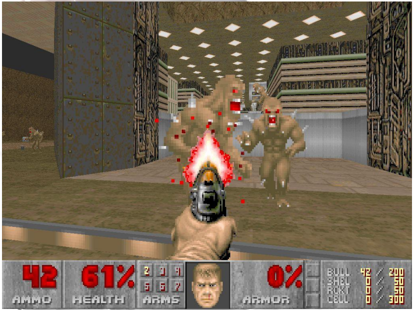
Outline

64
amatbe rc meaty
ammeans rc goo has become a zombie
05
11
25m
Infection
4
2
BUNGIE



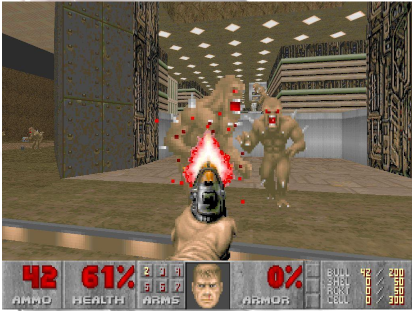
The advent of GPU's...or the art of shooting monsters

1993

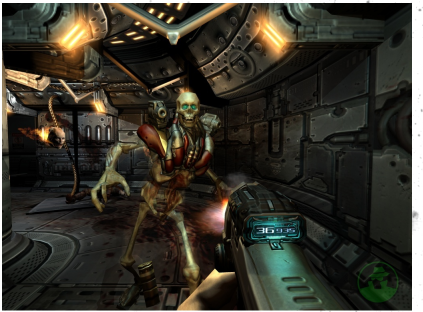


The advent of GPU's...or the art of shooting monsters

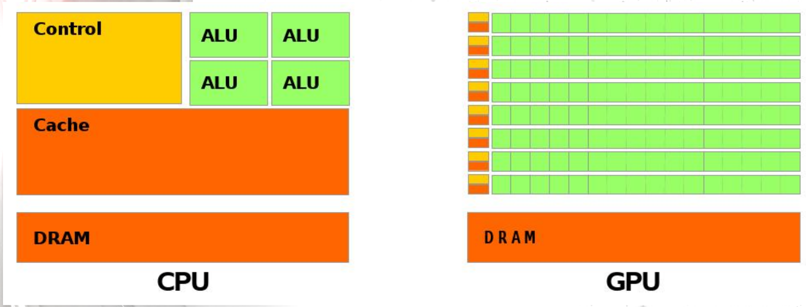
1993



2004



The concept of GPU's



- One single GPU allows for massive parallelisation
- Desktop computers become comparable in performance to clusters, at $\sim 1/1000$ of the cost.
- A problem has to be specially suited for GPU-parallelisation
  $\Rightarrow$ Data-parallel
- Codes have to be ported, separate host and device code

GPU-System in Bologna: Jabba the Hutt

NVIDIA Tesla C1060

- 240 streaming cores
- 4 GB DDR3 GPU memory
- 933 GFLOPS peak performance
- CUDA interface (C-based)

NVIDIA C for CUDA

- Interface to address GPU's
- C functionality with C++ syntax
- Recently, double support
- Some libraries: BLAS, FT
- Special compiler, no constraints for the host-code

Codes to be ported

1. `SaWLens` (JM)
   - Nonparametric, combined lensing reconstructions on AMR grids.
   - Already implemented in MPI.
   - Parallelisation strategy: adaptive-averaging of background galaxies, covariance determination, build-up of linear systems of equations.
2. `SkyLens` (Massimo)
   - Creation of extremely realistic lensing scenarios.
   - Ported to C++.
   - Parallelisation strategy: background galaxy population, ray-tracing
3. `SimLens??` (Massimo, Francesco)
   - Creation of deflection angle maps from numerical simulations.
   - Based on ray-tracing.
   - Parallelisation strategies: light rays are independent, different source redshift distributions.

HPC-Europa proposal was submitted on Saturday.

A first small test

- The toy problem:
  - Simulate a typical SaWLens problem
  - Calculate a typical coefficient matrix

$$\mathcal{B}_{lk} = a_i b_j \mathcal{C}_{ij} \mathcal{D}_{ik} \mathcal{E}_{jl},$$

  while using Albert's sum convention.
  - Dimensions:

$$l, k \in [0, ..., 2499], \quad i, j \in [0, .., 15]$$

- Competitors:
  1. Jabba's CPU: Intel XEON quadcore @ 2.5 GHz, one core used
  2. Jabba's GPU: NVIDIA Tesla C1060 @ 1.2 GHz, 240 cores used
- The runtime:
  1. CPU: `82.3 s`
  2. GPU: `1.03 s`

There is not much to comment on that.

Flexion...or weak lensing goes bananas

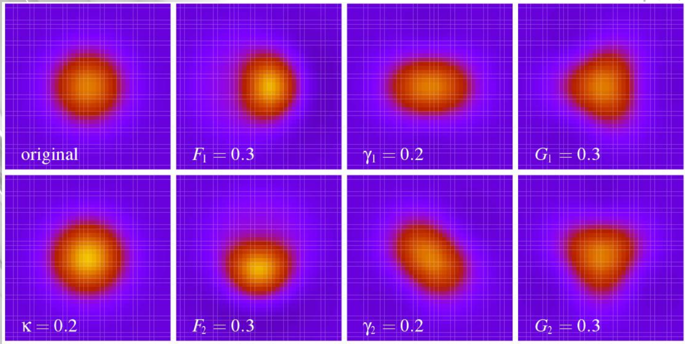
$$\boldsymbol{\theta} = \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} \rightarrow \theta = \theta_1 + i\theta_2$$

$$\beta = \theta - \alpha(\theta, \psi)$$

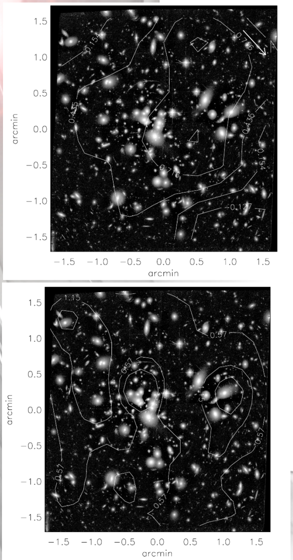
$$I(\theta) = I^{(s)}[\beta(\theta, \psi)]$$

$$\partial := \partial_1 + i\partial_2 \qquad \partial^\dagger := \partial_1 - i\partial_2$$

$$\begin{aligned} \partial\psi &= \alpha & s &= 1 \\ \partial^\dagger\partial\psi &= 2\kappa & s &= 0 \\ \partial\partial\psi &= 2\gamma & s &= 2 \\ \partial\partial^\dagger\partial\psi &= 2\mathcal{F} & s &= 1 \\ \partial^3\psi &= 2\mathcal{G} & s &= 3 \end{aligned}$$

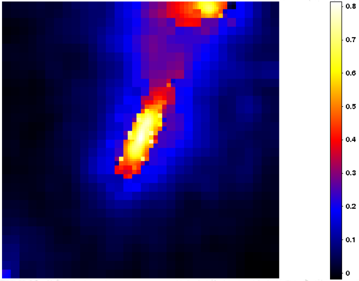


What this is all about: (Leonard et al. 2007)

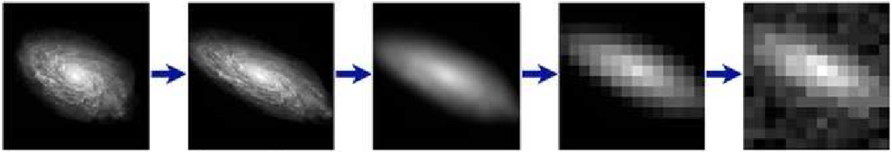


SaWLens 2.0 (JM in prep.)

- Flexion is already implemented
- WL + SL + Flexion + ...
- AMR grids
- on GPU's: runtime $\mathcal{O}(minutes)$

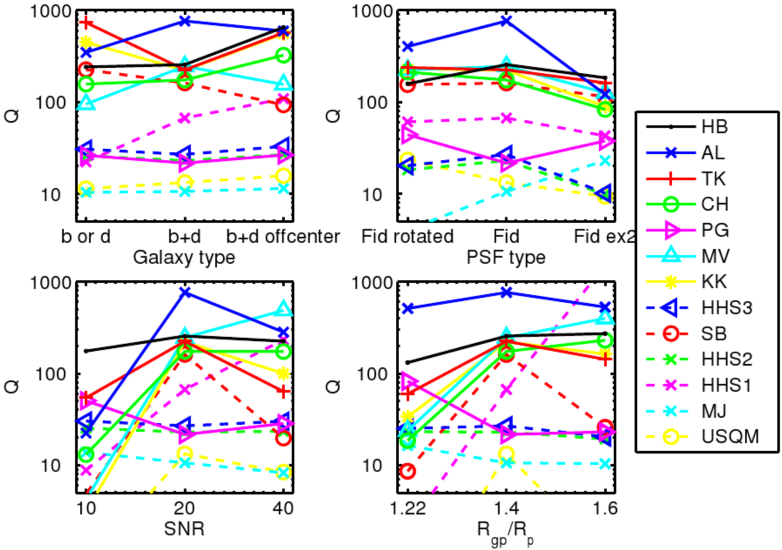


BUT:



Rank	Author	Method	Q
1	HB	CVN Fourier	211
2	AL	KK99	131
3	TK	Lensfit	119
4	CH	KSBf90	52.3
5	PG	gfit	32.0
6	MV	KKshapelets with flexion	28.6
7	KK	KKshapelets	23.0
8	HHS3	GaussStackForwardGaussCleaned	22.4
9	SB	im2shape	20.1
10	HHS2	GaussStackForwardGauss	19.9
11	HHS1	Gauss	12.8
12	MJ	BJ02 deconvolved shapelets	9.80
13	USQM	USQM	1.22

The bright side and an idea as a side effect



The dark side, but we are working on it

- Most model based techniques will not work, because the model cannot describe Flexion
- An exception are 'Shapelets', but they seem to have severe problems
- You can think about several solutions:
  1. Build a model which can describe flexion.
  2. Try to fix the Shapelets-approach (Sersiclets)
  3. Go back to direct techniques (KSB strikes back)