Julian Merten

Joint cluster reconstructions on GPUs

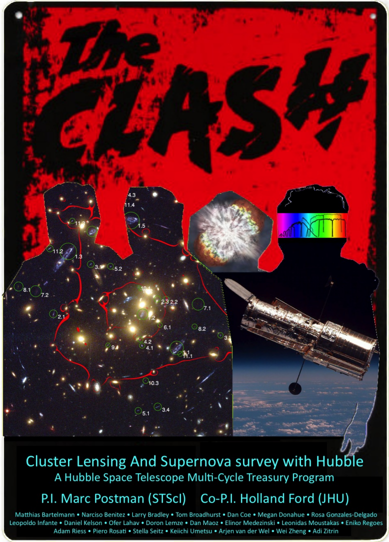
Institut für Theoretische Astrophysik
Zentrum für Astronomie
Universität Heidelberg
INAF - Osservatorio Astronomico di Bologna

July 20th, 2010

with: Massimo Meneghetti (OABO)
& Matthias Bartelmann (ITA)

Great things to come: The CLASH

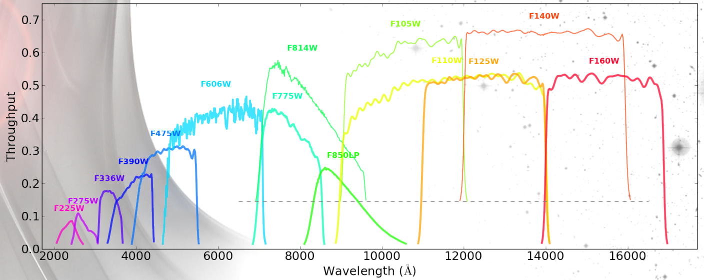
One of three HST/MCT programmes. Start September 2010 (3 cycles).



Science Drivers

- To map the dark matter in galaxy clusters
- To detect SN out to redshifts $z > 1.5$
- To detect and characterise $z > 7$ galaxies
- To study the galaxies in and behind the clusters

CLASH



CLASH Facts

- 25 X-ray clusters
- 524 orbits
- ACS + WFC3 obs.
- 14 (16) wave bands
- wide follow-ups with SUBARU

Our contribution

- Full pipeline calibration
- nonparametric DM profiles
- weak-lensing shapes + flexion
- magnification maps for the high-z guys (Maruša's talk)

The basic idea of an "inverse" method (Bartelmann96)

Cluster lensing in a box

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

$$\partial = \partial_1 + i\partial_2 \qquad \partial^* = \partial_1 - i\partial_2$$

$$\psi \qquad \alpha = \partial\psi$$

$$2\gamma = \partial\partial\psi \qquad 2\kappa = \partial^*\partial\psi$$

$$2F = \partial^*\partial\partial\psi \qquad 2G = \partial\partial\partial\psi$$

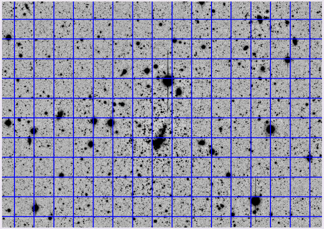
Statistical approach

$$\chi^2(\psi) = \chi_1^2 + \chi_2^2 + \chi_3^2 + \ldots$$

Possible constraints:

- Ellipticities of background sources
- Flexion (JM10 in prep.)
- Multiple image systems (Bradač05+)
- Critical curve estimates (JM09+)

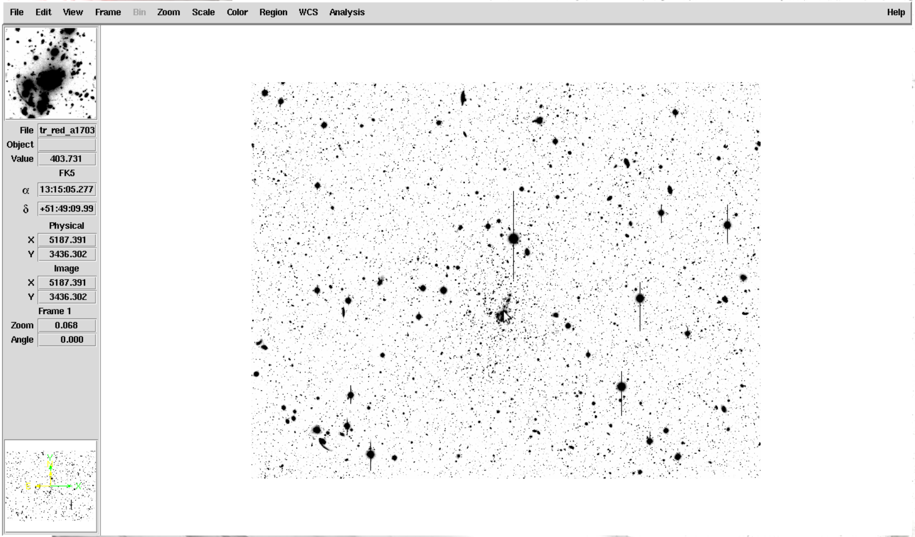
The trick



$$\frac{\partial\chi^2(\psi_k)}{\partial\psi_l} \overset{!}{=} 0$$

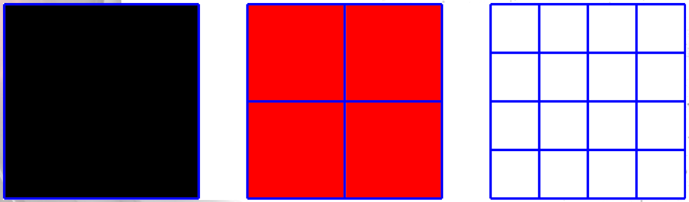
$$\Rightarrow \mathcal{B}_{lk}\psi_k = \mathcal{V}_l$$

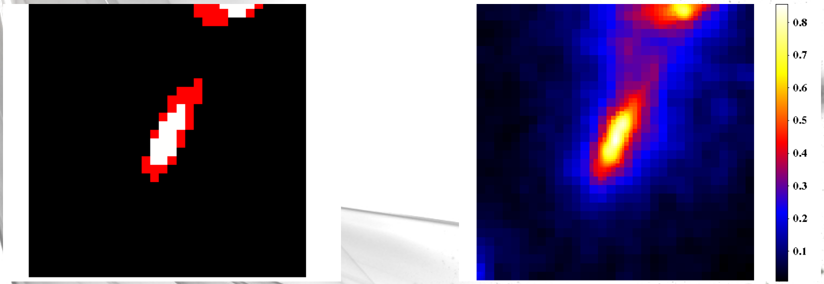
A problem of different scales (JM10, Bradač09)



(Abell 1703 in SUBARU r-band)

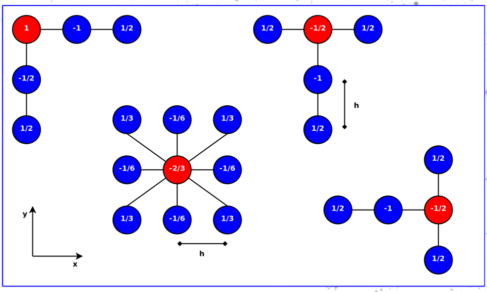
A problem of different scales *(JM10, Bradač09)*





Making it all work: Numerics (JM09)

- α, γ, κ, F and G can be expressed by derivatives of ψ via finite differences.
- Linearisation of the problem.
- 2-level iteration scheme with simple regularisation (Bradač05).



Problem Runtime

$$\mathcal{B}_{lk}\psi_k = \mathcal{V}_l$$

$$\mathcal{B}_{lk} \sim a_i b_j C_{ij} D_{il} E_{jk}$$

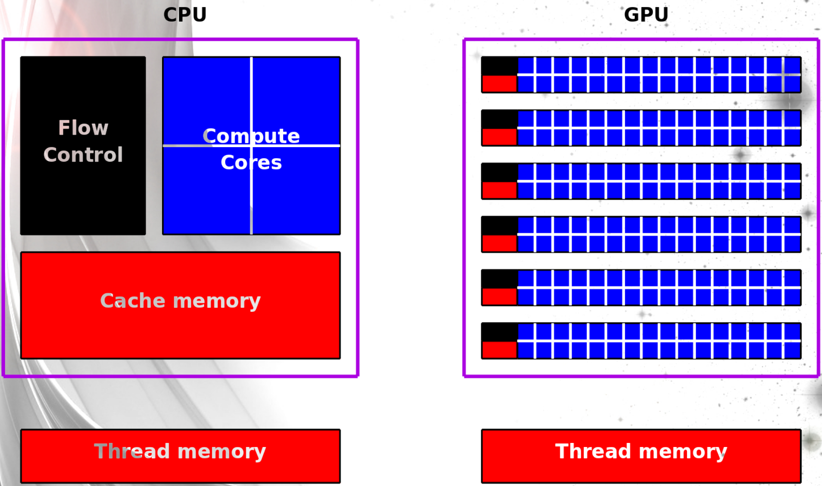
$$\mathcal{V}_l \sim a_i b_j C_{ij} E_{il}$$

$$l, k, i, j \sim \mathcal{O}(\text{grid\_dim}^2)$$

Geeky implementation facts

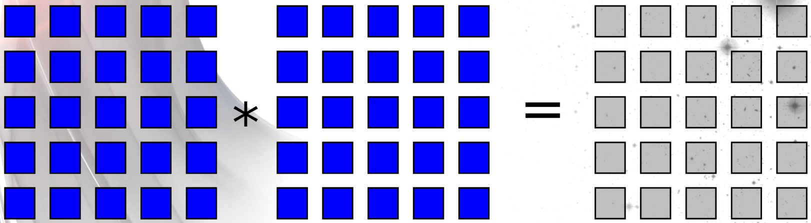
- Parallel C++ code
- medium sized $\sim$ 12000 lines
- Uses GSL, LAPACK, ATLAS, MPI
- Fully documented, including user manual
- and...`CUDA`...

*Data-parallel single node **GPU** Parallelisation*



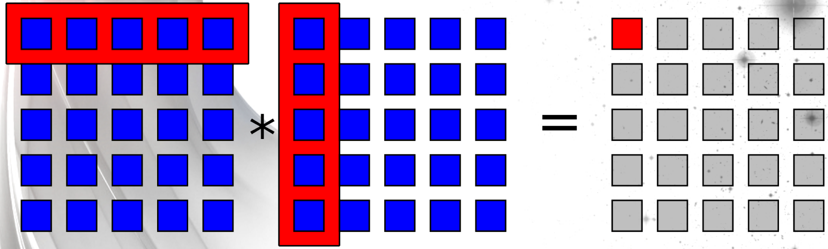
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



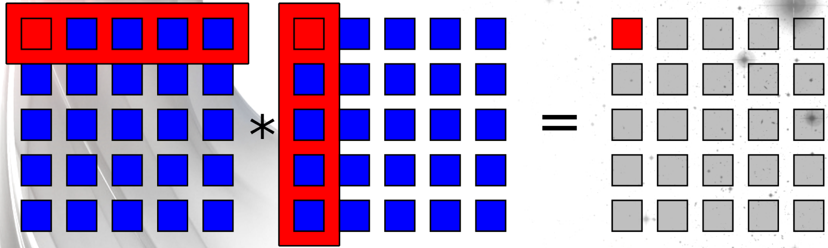
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



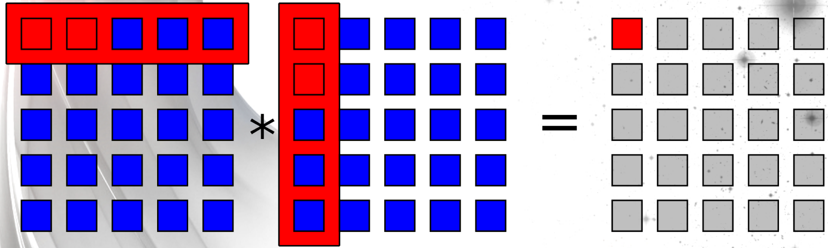
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



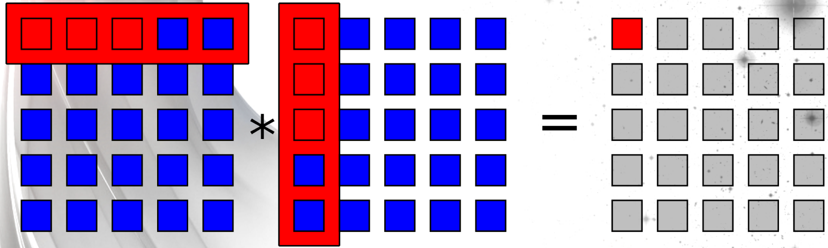
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



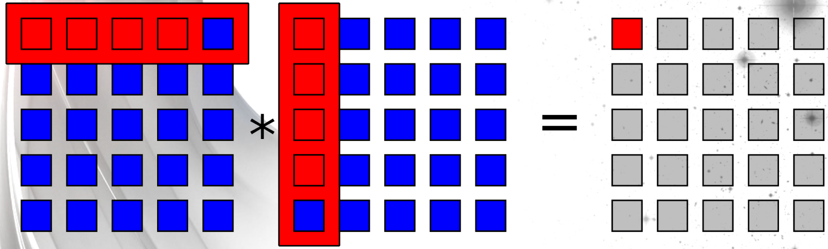
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



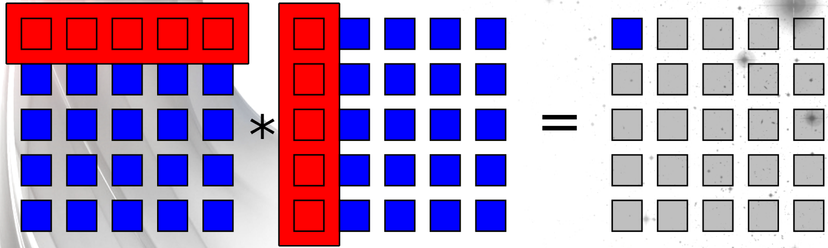
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



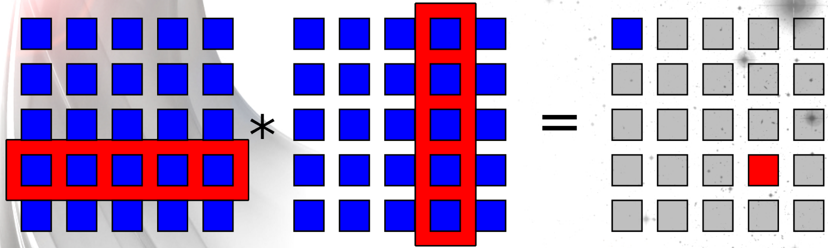
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for $\Rightarrow$ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



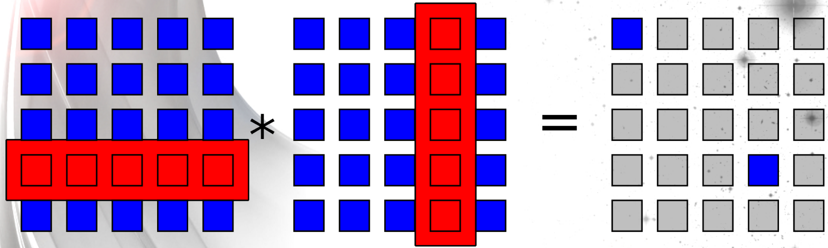
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



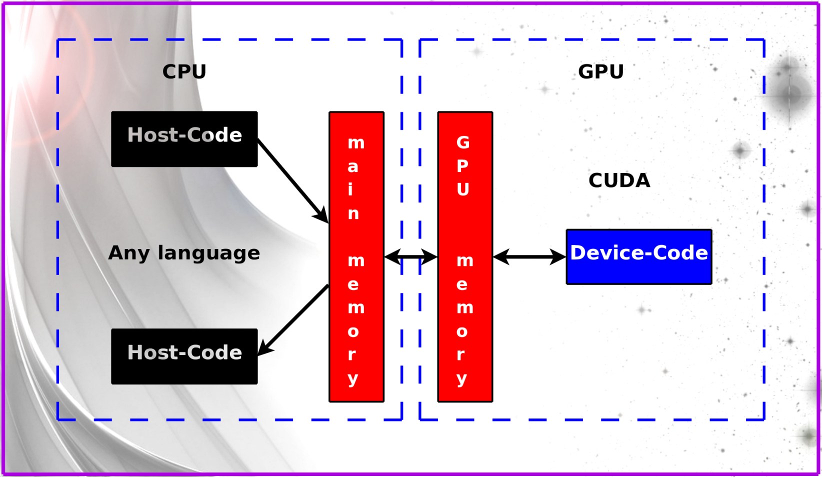
- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

*Data-parallel single node **GPU** Parallelisation*



- One single GPU allows for massive parallelisation at a fraction of the cost of a CPU cluster, if problem is suited for ⇒ **Data-parallel**.

GPUs in practice *(JM10 in prep.)*

NVIDIA Tesla C1060

- 240 streaming cores
- 4 GB DDR3 GPU memory
- 933 GFLOPS peak performance
- `Fermi` cards out by now, update planned

Speed-up

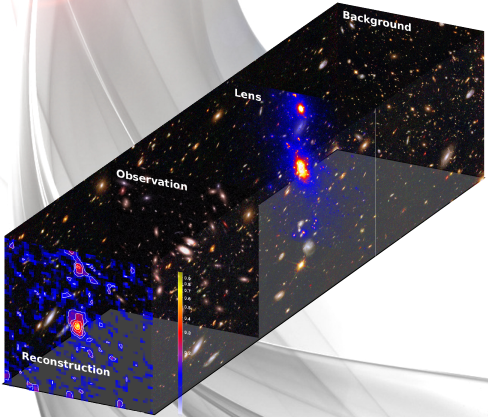
- Calculate:

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

- one-core CPU: `82.3 s`
- 240 core GPU: `1.03 s`

Realistic lensing simulations: **SkyLens** *(Meneghetti, JM 08/10)*

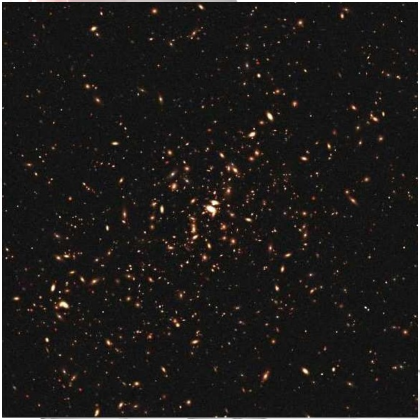
Developers: Massimo Meneghetti, Peter Melchior, Fabio Bellagamba, JM

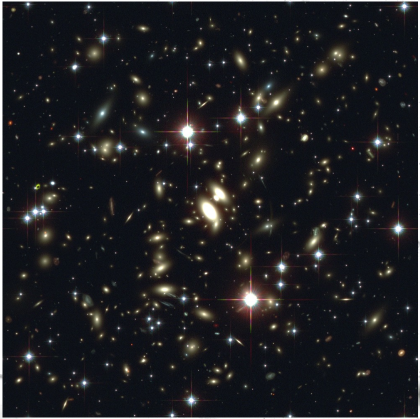


Name	Description
D	aperture diameter
g	detector gain
A_{pix}	pixel area
$F(\lambda)$	used filter
$M(\lambda)$	mirror filter curve
$O(\lambda)$	optics filter curve
$C(\lambda)$	CCD filter curve
FoV	total field-of-view
RON	detector readout-noise
f	flat-field accuracy
a	residual flat-field error
PSF	PSF model
t_{exp}	exposure time
$A(\lambda)$	atmospheric extinction
m_{a}	airmass
$\mathrm{SED}_{\mathrm{sky}}$	sky-background emission
$\mathrm{SED}_{\mathrm{gal}}$	background population
α	deflection angle map

Realistic lensing simulations: **SkyLens** *(Meneghetti, JM 08/10)*

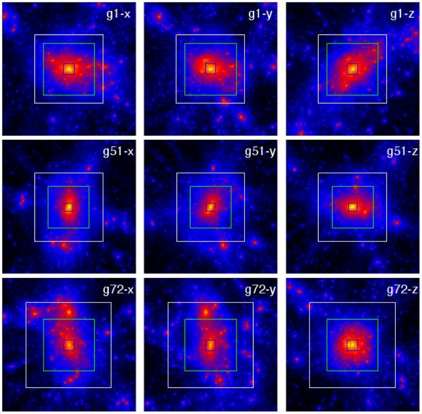
Developers: Massimo Meneghetti, Peter Melchior, Fabio Bellagamba, JM

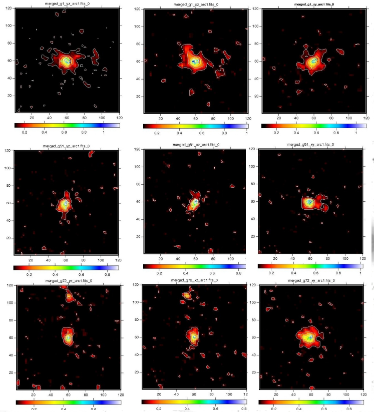




Realistic lensing simulations: **SkyLens** *(Meneghetti, JM 08/10)*

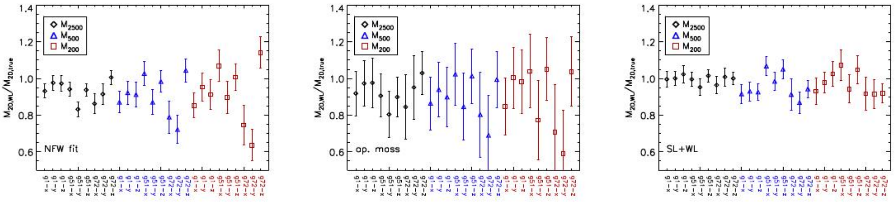
Developers: Massimo Meneghetti, Peter Melchior, Fabio Bellagamba, JM



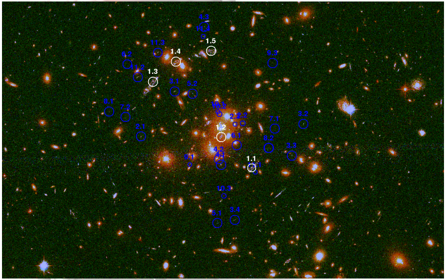


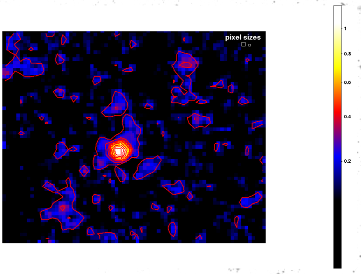
Realistic lensing simulations: **SkyLens** *(Meneghetti, JM 08/10)*

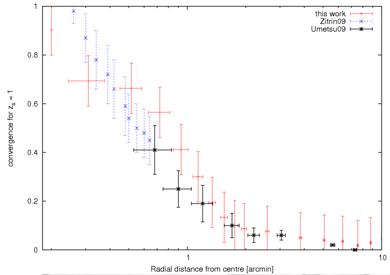
Developers: Massimo Meneghetti, Peter Melchior, Fabio Bellagamba, JM

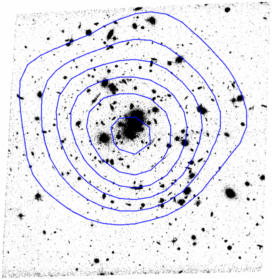


CL0024 (with: T. Broadhurst, A. Zitrin, K. Umetsu)

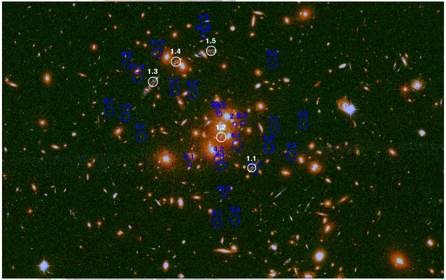


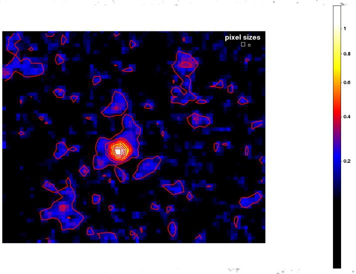


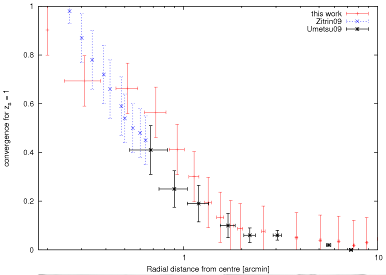


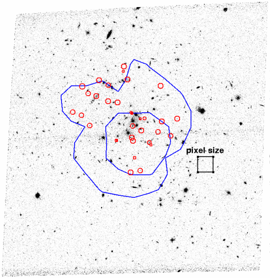


CL0024 (with: T. Broadhurst, A. Zitrin, K. Umetsu)









Conclusions

1. Looking forward to see the `CLASH`:
   - Cluster DM profiles
   - Strong lensing pecularities
   - High-z universe
2. Joint methods, combining several lensing constraints are able to reliably reconstruct clusters in a nonparametric way.
3. GPUs are able to outperform “classical” compute clusters at a fraction of the cost, if your code is suited for such an implementation.