Calibrating Wide Field Surveys

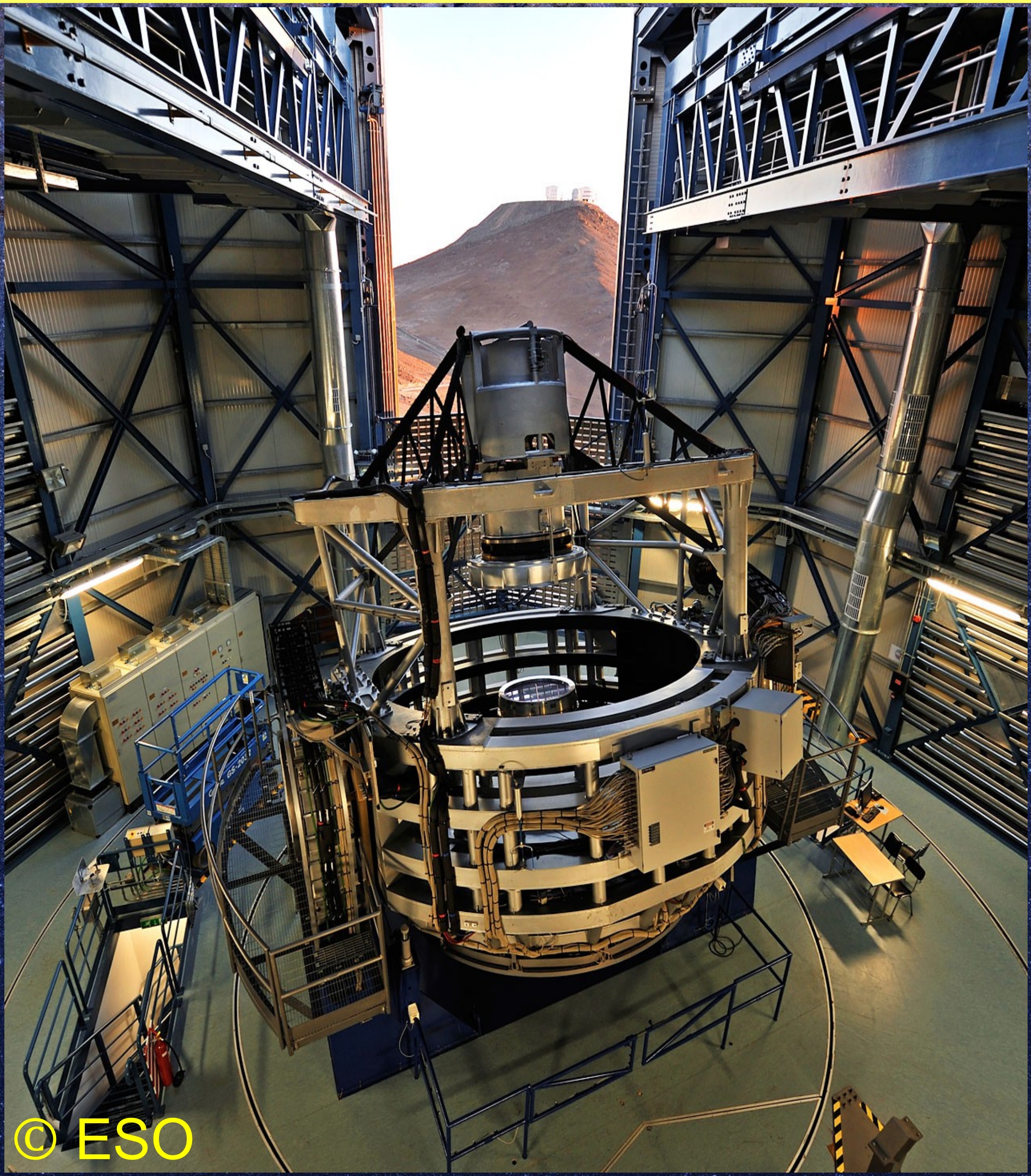
Carlos González Fernández

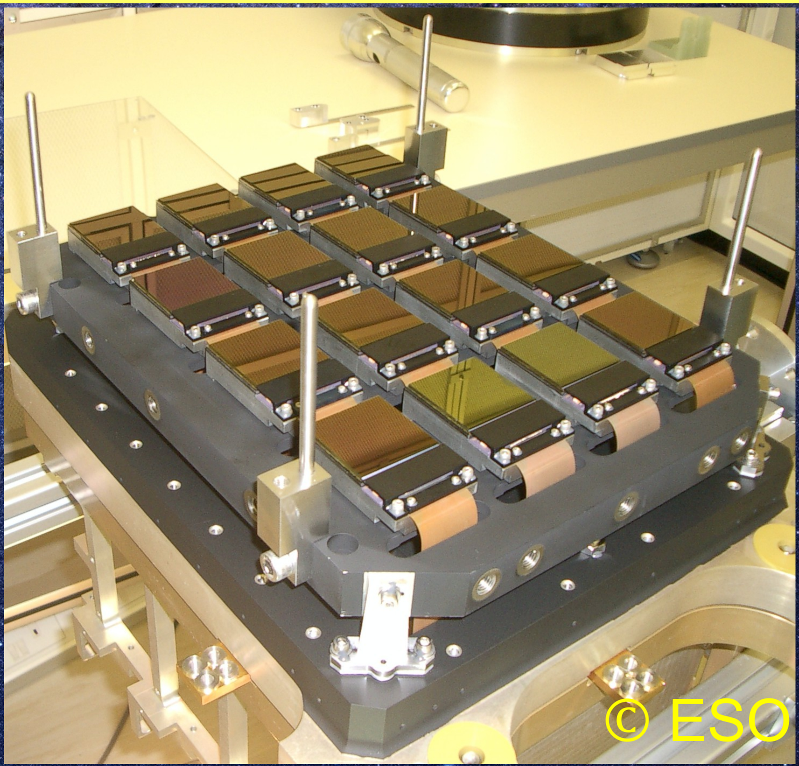
Mike Irwin, Jim Lewis, Eduardo González Solares
Aybuke Kupcu Yoldas, Simon Hodgkin

2017 ESO Calibration Workshop

Santiago, 19th of January, 2017

VISTA survey telescope

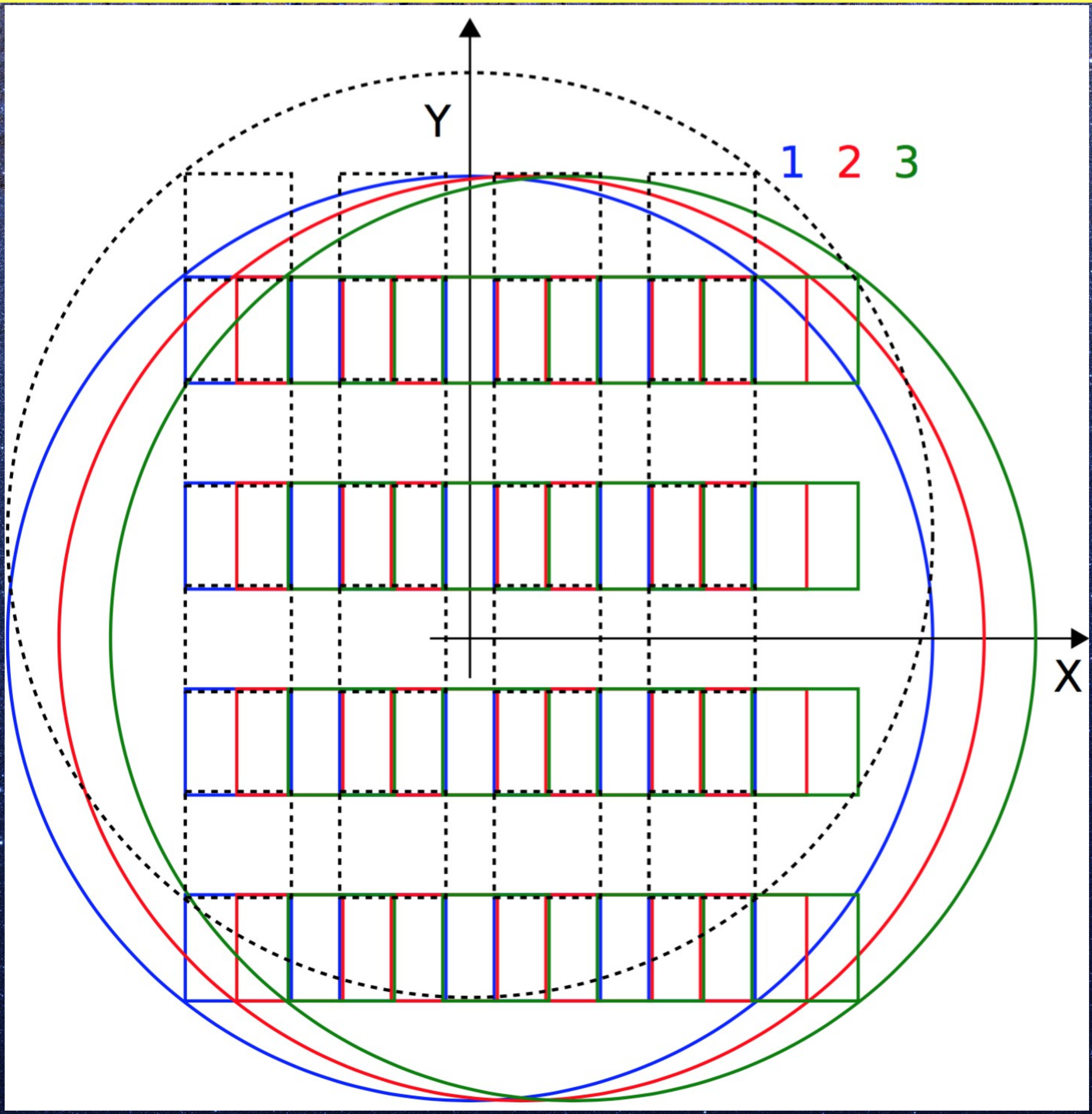


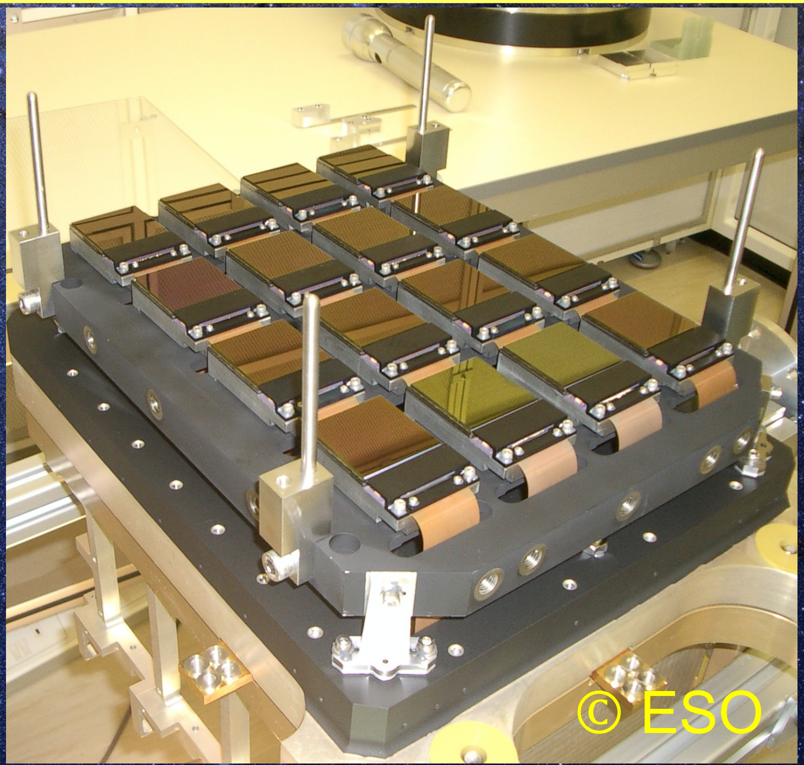


VISTA:
- 4m, fov: 1.65° Ø, px: 0.34”

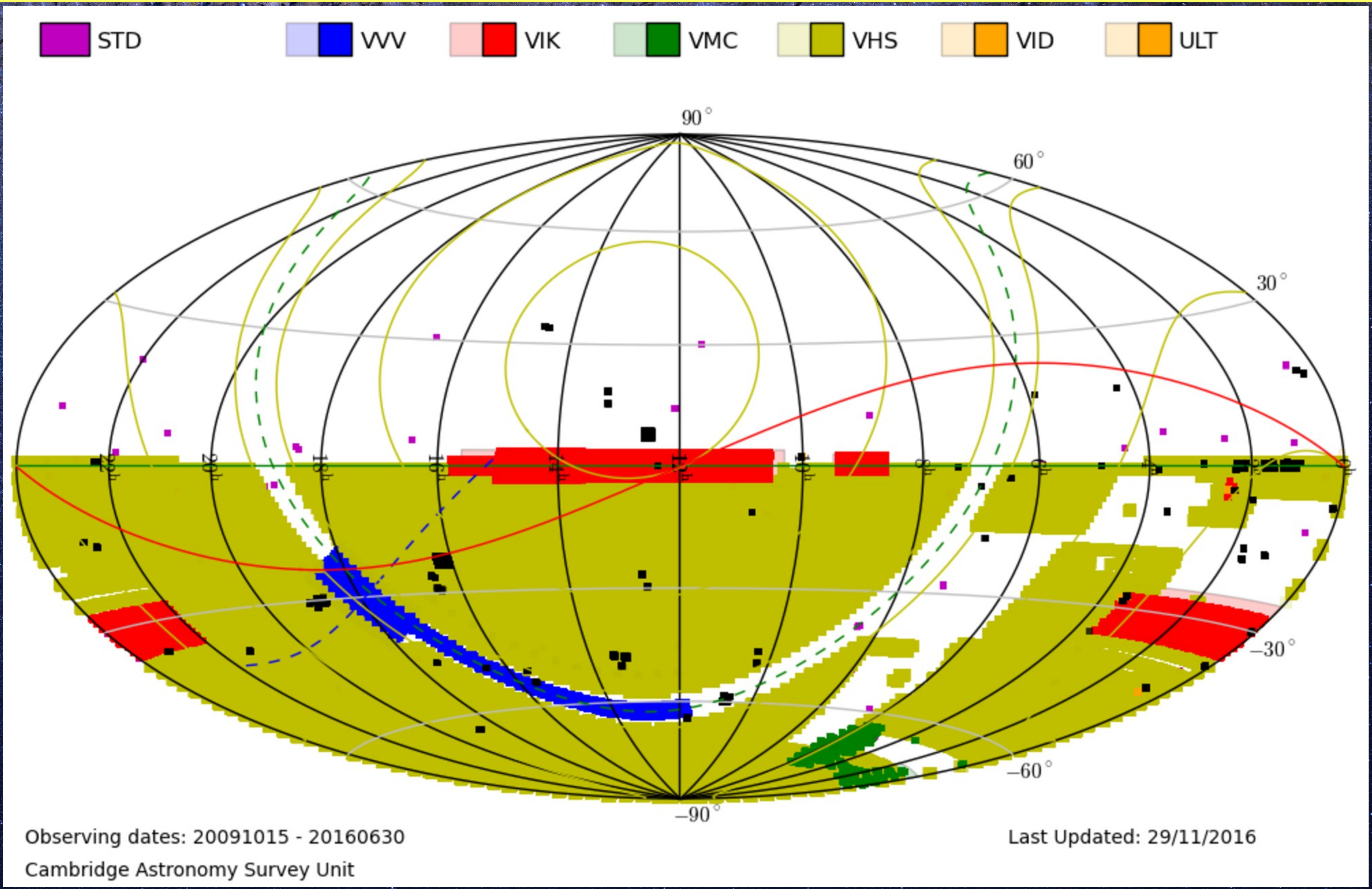
VIRCAM:
- ZYJHKs+NB
- 16×2048^2 Raytheon dets

VISTA survey telescope





VISTA survey telescope



What do we want?

Magnitudes (or fluxes) that are physically consistent between themselves and with other measurements.

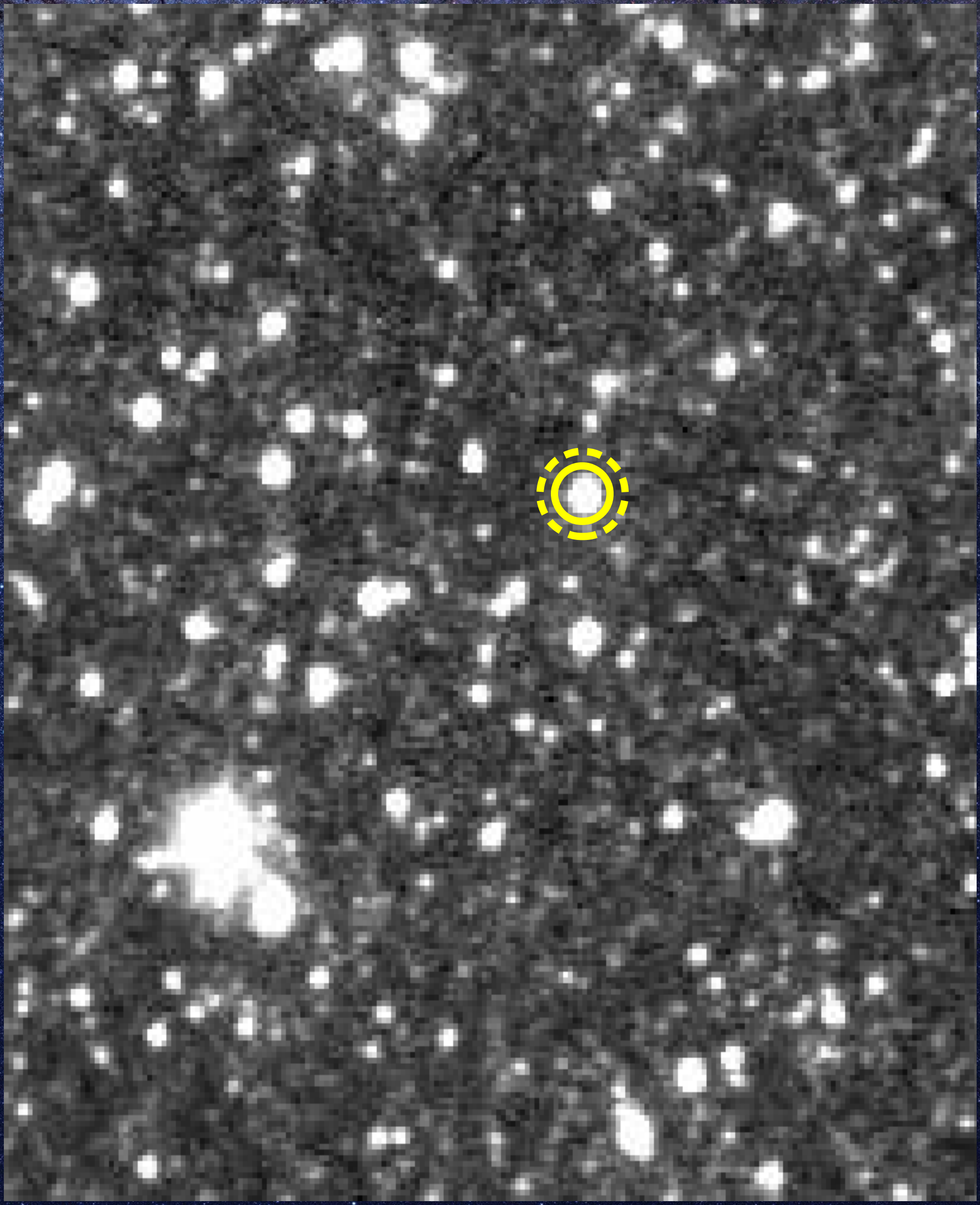
What do we want?

We measure mags as:

$$m = ZP - 2.5 \cdot \log(f_i / t)$$

Normally:

$$f_i = F[S(x, y)] = \sum_i S_i - \sum_j S_j$$

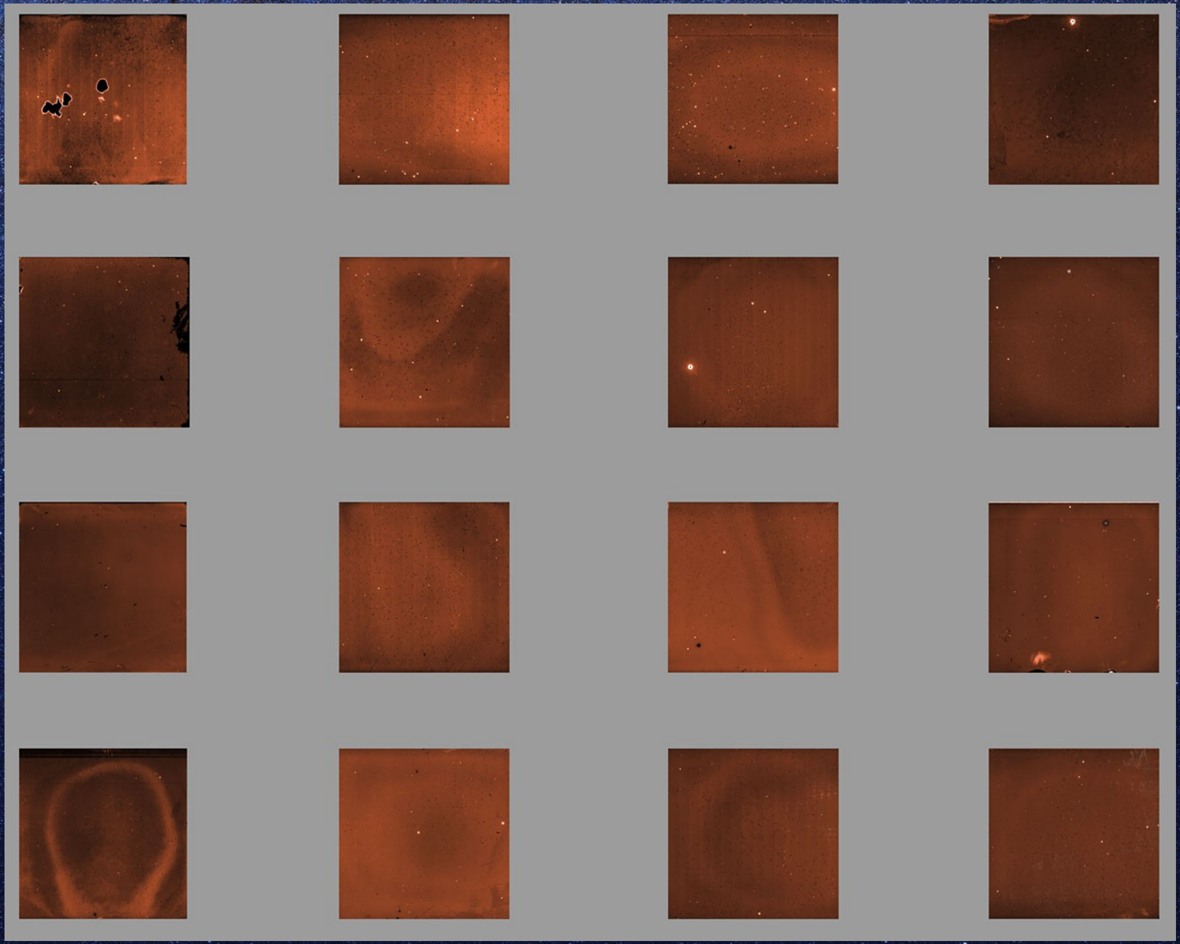


First callibration steps

But not all pixels are created equal:

- Some variation you can correct with a good flat:

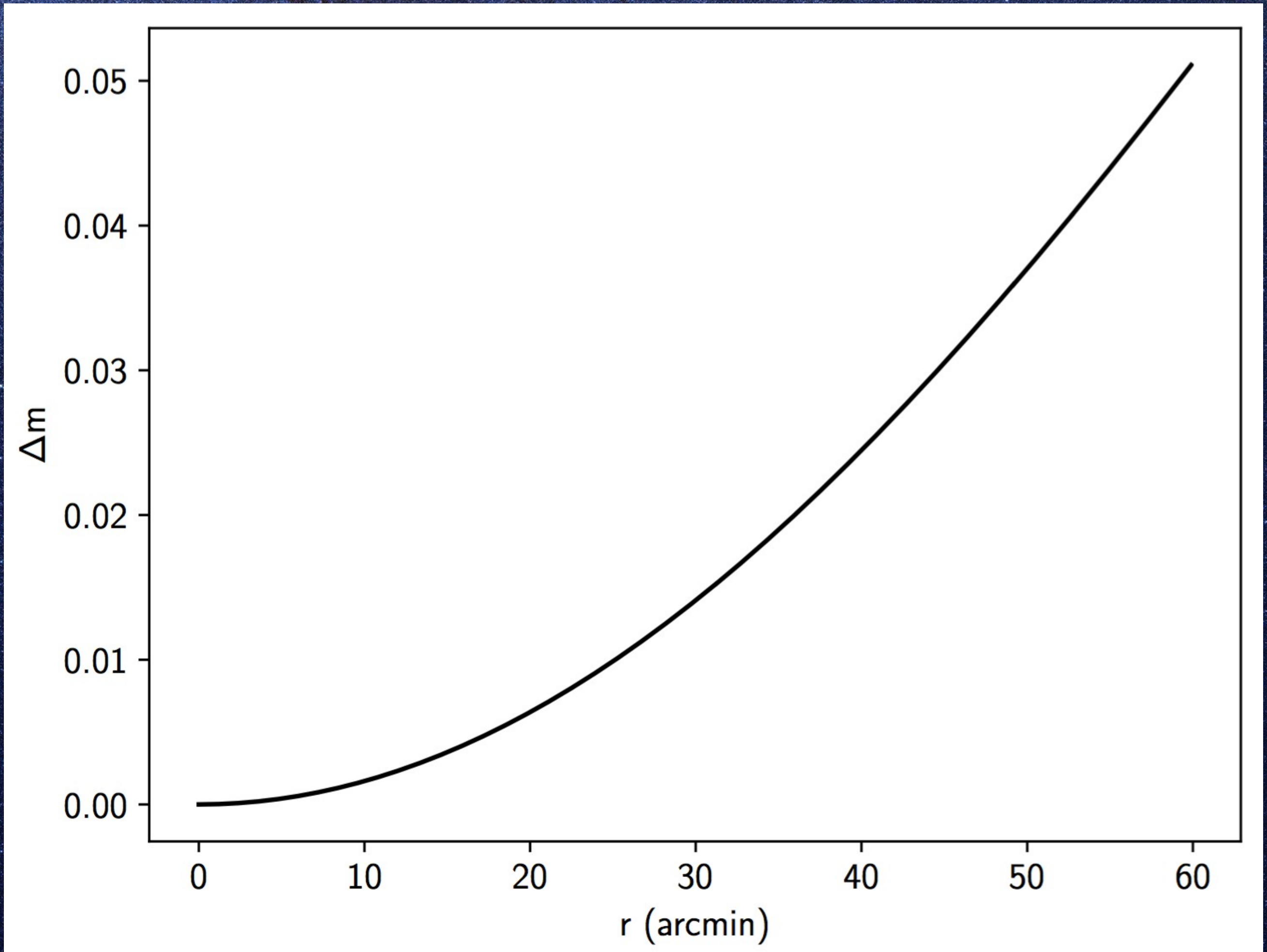
  The sky is always right.



First callibration steps

But not all pixels are created equal:

- Some variation you can correct with a good flat.
- Some you can correct even before you start observing.



What do we want?

We measure mags as:

$$m = ZP - 2.5 \cdot \log\left(f_i / t\right)$$

ZP looks deceptively simple:

$$ZP = \underbrace{\log\left[a\left(t, x, \lambda; SED\right)\right]} + \underbrace{\log\left[T\left(\lambda; t, SED\right)\right] + ZP_{ref}}$$

This is easy to calculate, if everything goes well

For this we normally use reference standard stars

What calibrators to use

Ideally:

- Well distributed in the sky and in magnitude.
- Measured in the same system being calibrated.
- They should be primary calibrators (ZP_{ref}).

What calibrators to use

But in reality:

- Well distributed in the sky and in magnitude.
- Measured in a similar system being calibrated.
- ~~They should be primary calibrators (ZP_{ref}).~~

↓

2MASS, SDSS, APASS, Gaia...

What if the phot. systems differ?

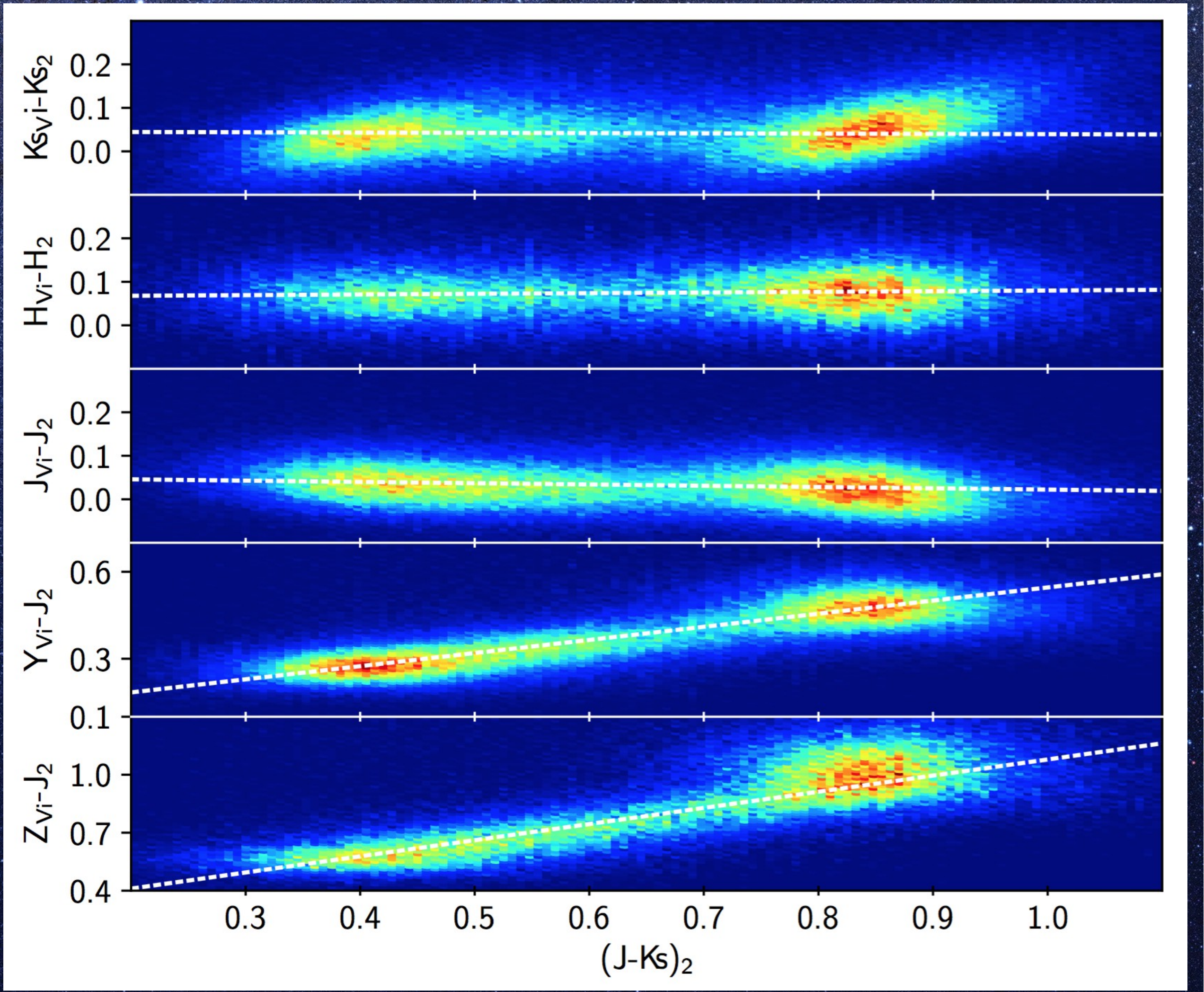
We need to transform from 2MASS into ours.

Under no reddening:

$$m_V = m_2 + C \cdot (J - K_s)_2$$

Sometimes requires risky extrapolation.

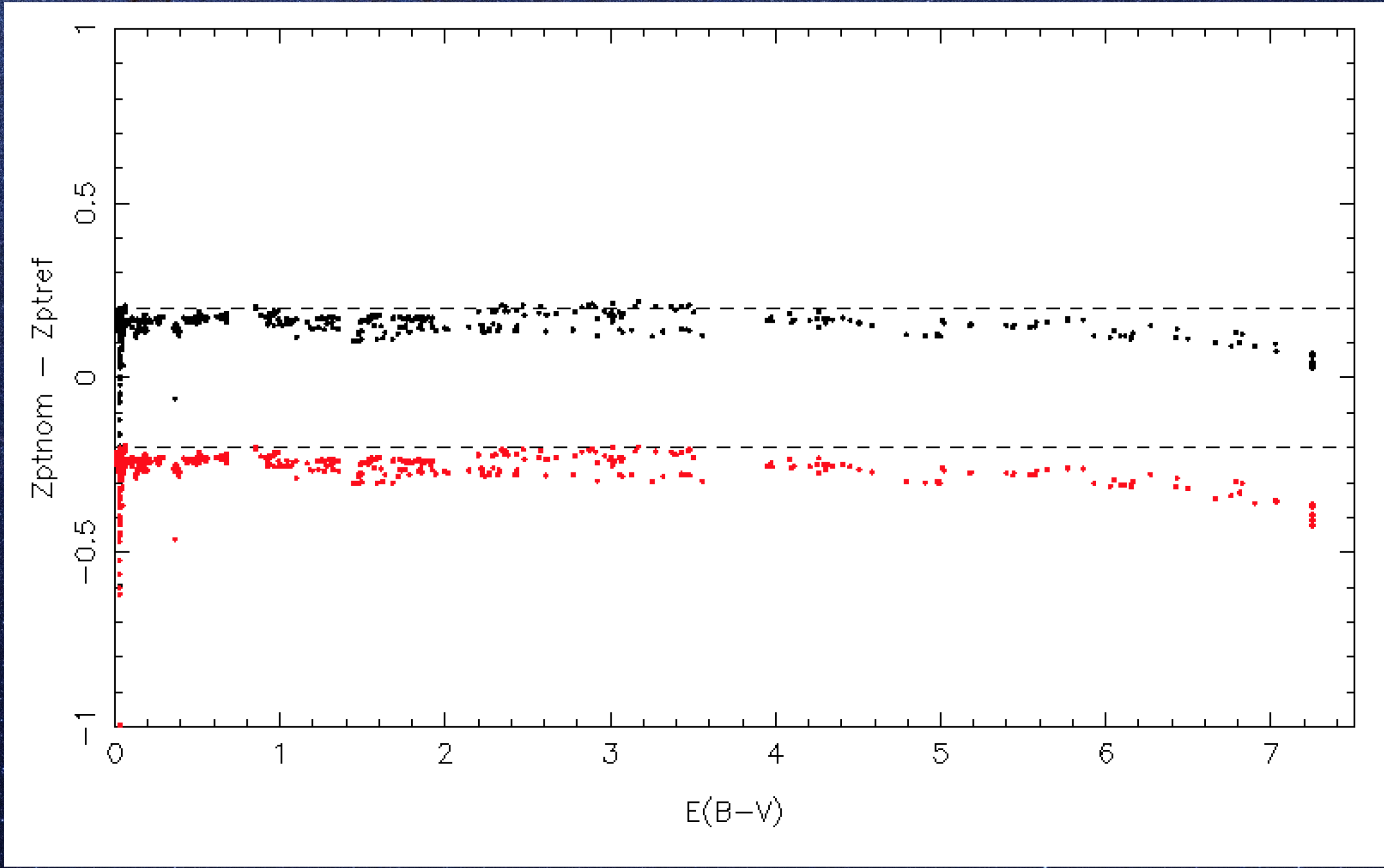
$Z_V = J_2 + (0.86 \pm 0.08) \cdot (J - K_s)_2$
$Y_V = J_2 + (0.46 \pm 0.02) \cdot (J - K_s)_2$
$J_V = J_2 - (0.031 \pm 0.006) \cdot (J - K_s)_2$
$H_V = H_2 + (0.015 \pm 0.005) \cdot (J - K_s)_2$
$Ks_V = Ks_2 - (0.006 \pm 0.007) \cdot (J - K_s)_2$



What if the phot. systems differ?

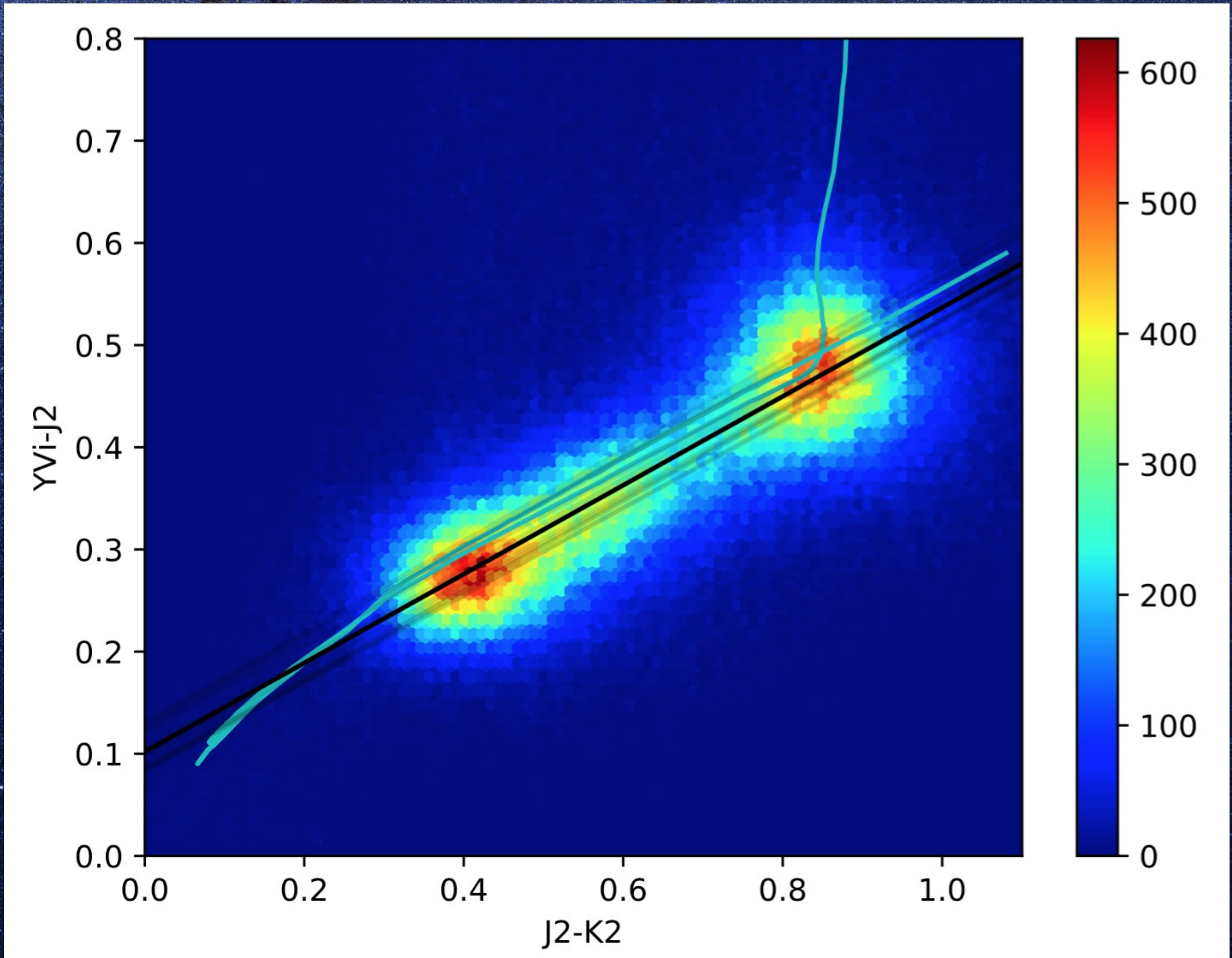
If there is reddening, things can get really complicated, really soon:

$$Y_V = J_2 + C_Y \cdot (J - K_s)_2 + E(B-V) \cdot [A_Y - A_J - C_Y \cdot (A_J - A_K)]$$



...and into a standard system

We need to check our mags. against the standard that defines the system (Vega, AB, etc.)



...and into a standard system

But good A0V stars with low reddening and not too bright are hard to find, particularly in the SH.

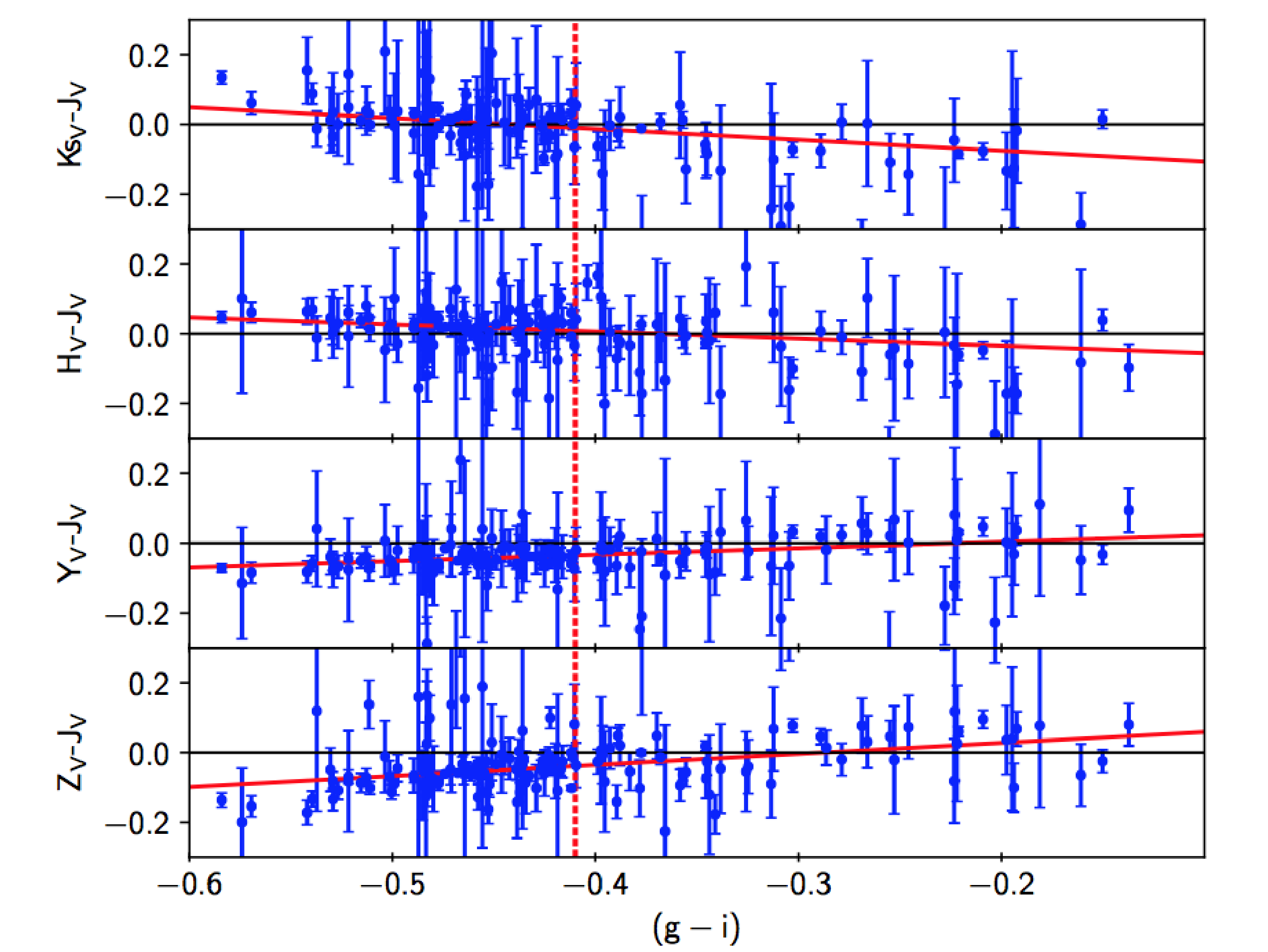
Two steps:

- Internal colour:

$$Z_V - J_V = -0.039 \pm 0.006$$
$$Y_V - J_V = -0.035 \pm 0.004$$
$$H_V - J_V = 0.008 \pm 0.006$$
$$Ks_V - J_V = -0.010 \pm 0.006$$

- Absolute ZP

$$(J_V - J_2) = 0.005 \pm 0.007$$
$$(H_V - H_2) = 0.03 \pm 0.01$$
$$(Ks_V - Ks_2) = 0.00 \pm 0.01$$



Implementing the calibration

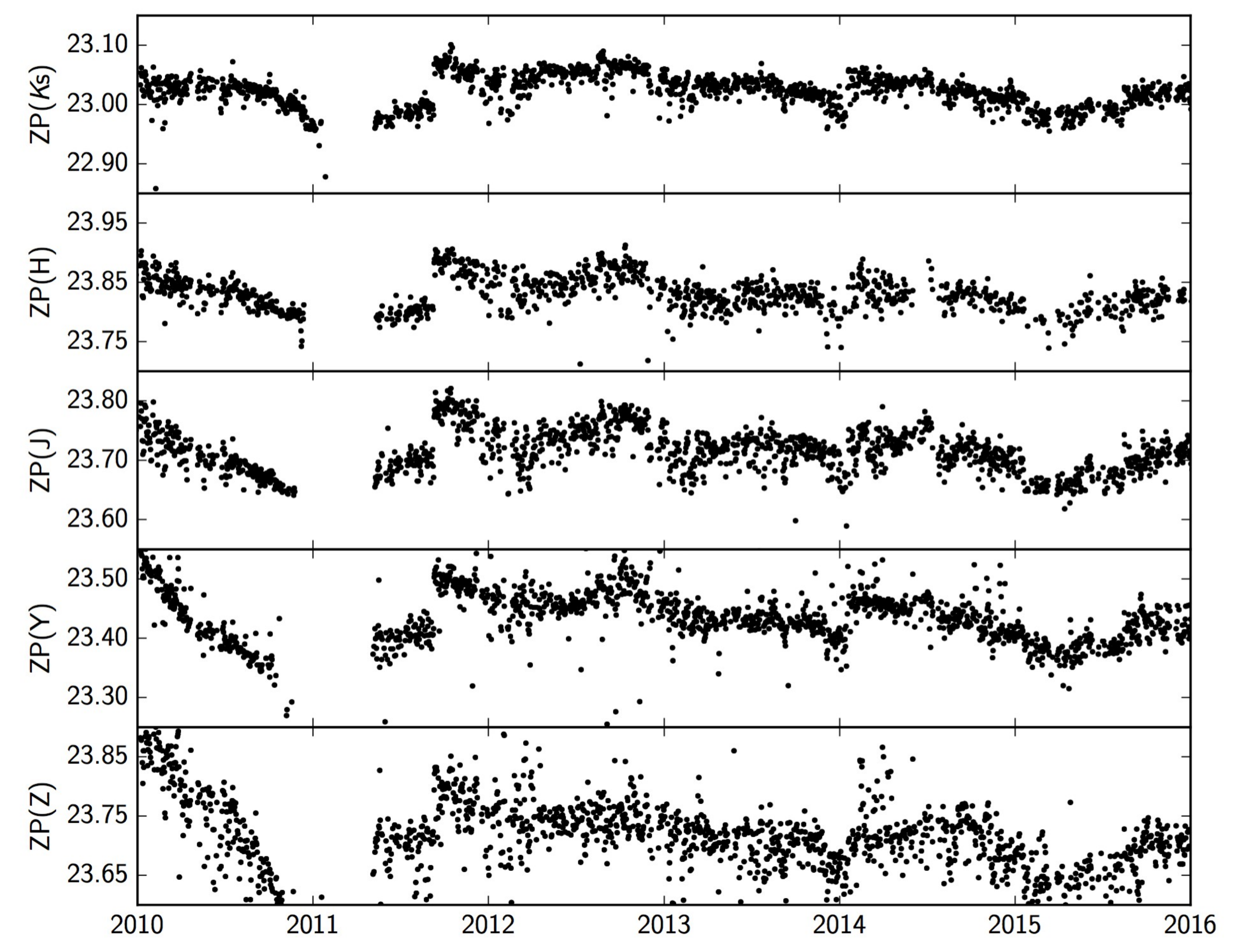
There are two alternative ways to calibrate large datasets:

1.- Real-time (-ish): one ZP per image + one absolute cal.

$$ZP = m_2 - A_2 + C_2 \cdot (J - K_s)_2 + \kappa \cdot (\chi - 1) + \Delta m_{CHIP} + m_i$$

Implementing the calibration

There are two alternative ways to calibrate large datasets:



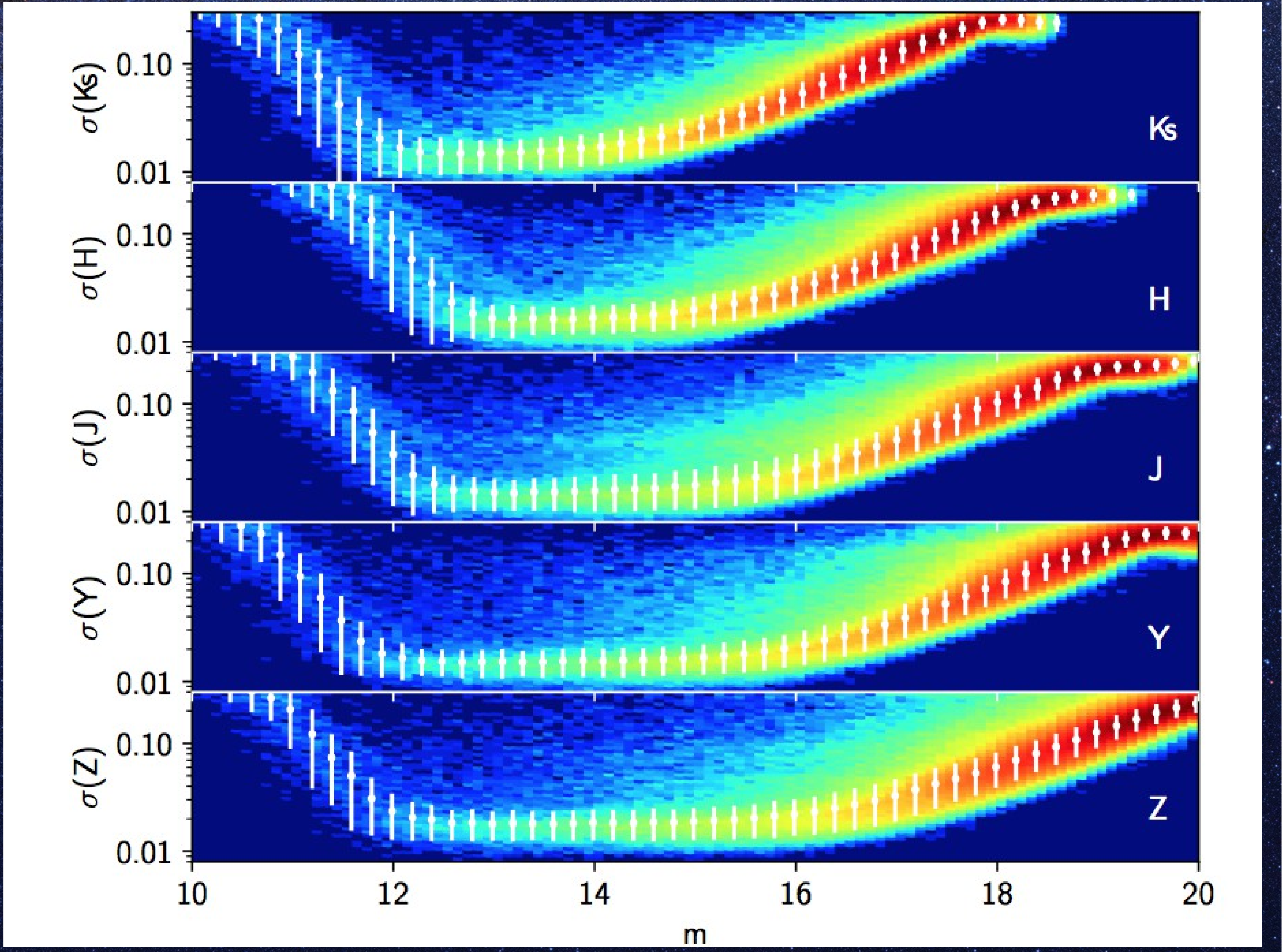
What's the precission you get?

Two types of error:

Absolute

$$(J_V - J_2) = 0.005 \pm 0.007$$
$$(H_V - H_2) = 0.03 \pm 0.01$$
$$(Ks_V - Ks_2) = 0.00 \pm 0.01$$

Internal



Implementing the calibration

There are two alternative ways to calibrate large datasets:

1.- Real-time (-ish): one ZP per image + one absolute cal.

$$ZP = m_2 - A_2 + C_2 \cdot (J - K_s)_2 + \kappa \cdot (\chi - 1) + \Delta m_{CHIP} + m_i$$

2.- Once enough observations are taken: übercal

Implementing übercal(-ish)

Übercal (Padmanabhan et al. 2007) relies on repeated observations:

$$m_1 = ZP_1 - \kappa \cdot (\chi_1 - 1) + (\Delta m_{CHIP})_1 - (m_i)_1$$
$$m_2 = ZP_2 - \kappa \cdot (\chi_2 - 1) + (\Delta m_{CHIP})_2 - (m_i)_2$$

$$\Delta(m_i) = \Delta(ZP) - \kappa \cdot \Delta(\chi) + \Delta[\Delta(m_{CHIP})]$$
$$(m_i)_1 - m_{2MASS} = ZP_1 - \kappa \cdot (\chi_1 - 1) + (\Delta m_{CHIP})_1$$

Implementing übercal(-ish)

Übercal (Padmanabhan et al. 2007) relies on repeated observations:

$$\Delta(m_i) = \Delta(ZP) - \kappa \cdot \Delta(\chi) + \Delta[\Delta(m_{CHIP})]$$
$$(m_i)_1 - m_{2MASS} = ZP_1 - \kappa \cdot (\chi_1 - 1) + (\Delta m_{CHIP})_1$$

$$\begin{pmatrix} 1 & -1 & \Delta(\chi) & 1 & -1 \\ 1 & 0 & \chi_1 - 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} ZP_1 \\ ZP_2 \\ \kappa \\ \Delta m_{C1} \\ \Delta m_{C2} \end{pmatrix} = \begin{pmatrix} \Delta(m_i) \\ (m_i)_1 - m_{2MASS} \end{pmatrix}$$

Implementing übercal(-ish)

$$\begin{pmatrix} 1 & -1 & 0\cdots & \Delta(\chi) & 1 & -1 & 0\cdots \\ & & & \vdots & & & \\ 1 & 0 & 0\cdots & (\chi-1) & 1 & 0 & 0\cdots \\ & & & \vdots & & & \end{pmatrix} \cdot \begin{pmatrix} ZP_1 \\ ZP_2 \\ \vdots \\ \kappa \\ \Delta m_{C1} \\ \Delta m_{C2} \\ \vdots \end{pmatrix} = \begin{pmatrix} \Delta(m_i) \\ \vdots \\ (m_i)_1 - m_{2MASS} \\ \vdots \end{pmatrix}$$

These are extremely large matrixes (10^8-10^9 mag. diffs. and 10^3 coefficients) but mostly empty, so amenable to sparse methods that can solve for *b*:

$$X \cdot b = Y$$

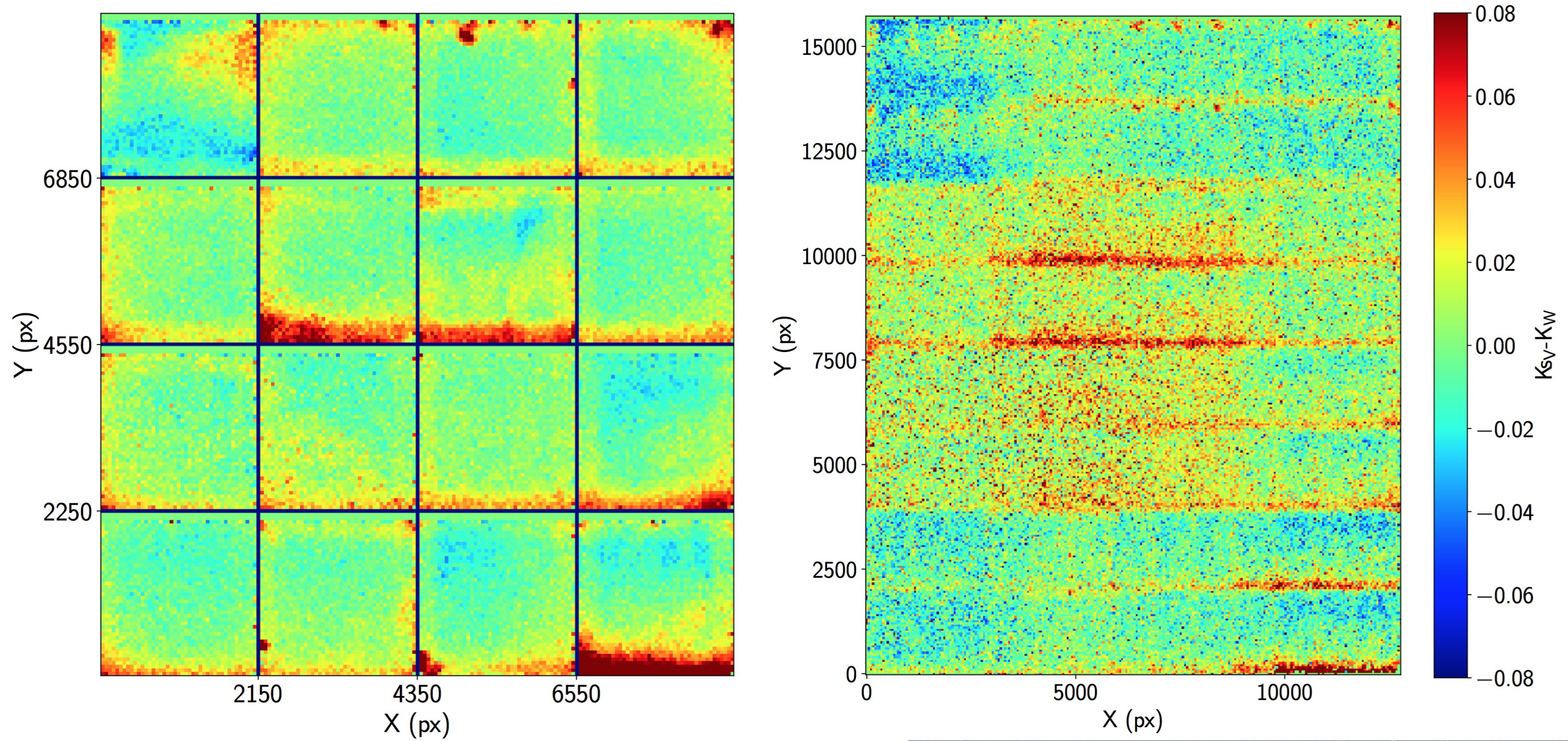
Implementing übercal(-ish)

Why go through all this work?

- Übercal is a single-step calibration.
- Uses all the data available:
  - Repetitions homogeneize the survey (without worrying about interstellar extinction, passbands, etc.)
  - Obs. of standards (that can be selected on well behaved lines-of-sight) set ZPs.
- Allows for a great liberty fine-tuning the cal. eq.

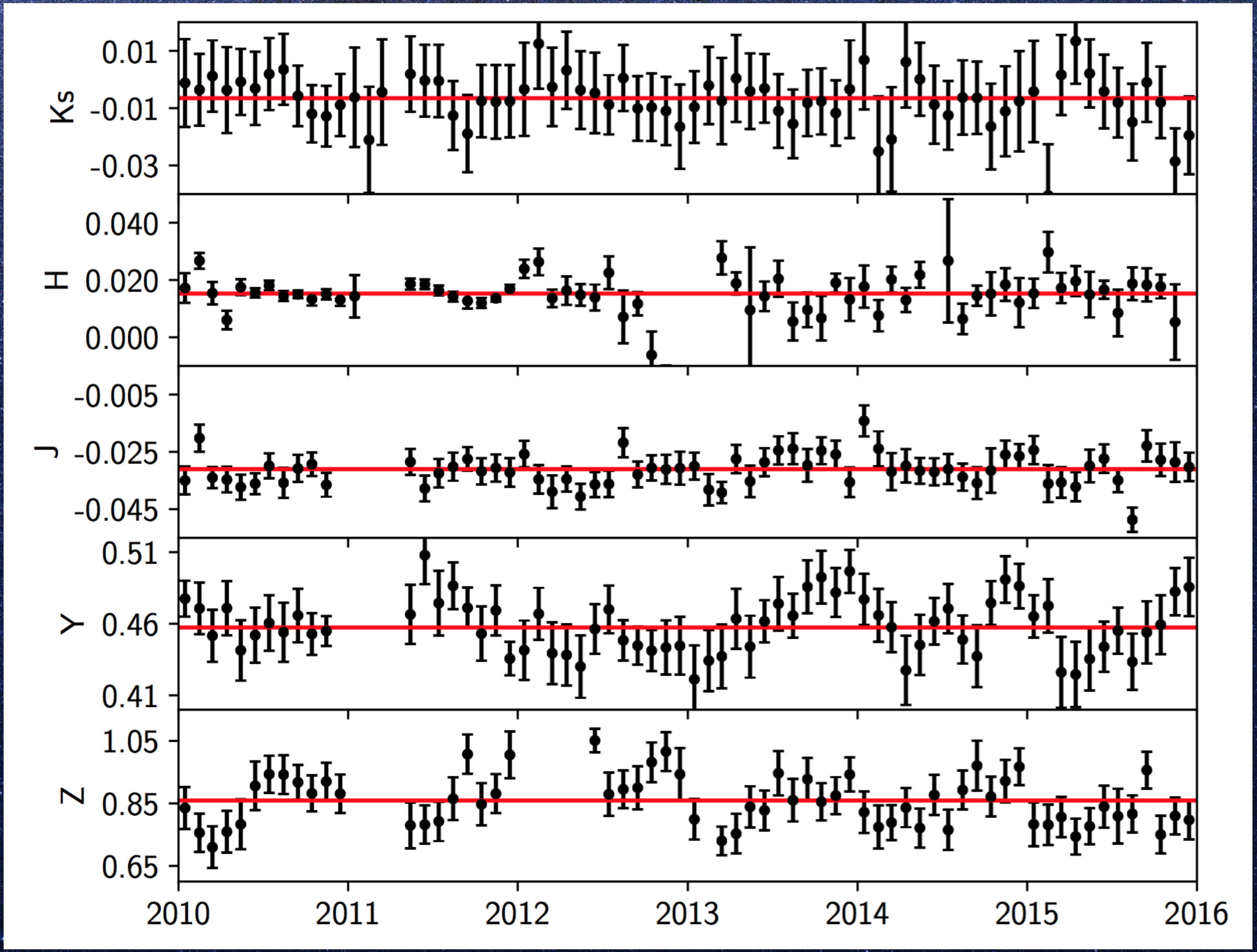
Implementing übercal(-ish)

$$ZP = m_2 - A_2 + C_2(t) \cdot (J - K_s)_2 +$$
$$+ \kappa(x, y) \cdot (\chi - 1) + \Delta m_{CHIP}(t) + m_i + f(x, y, t)$$



Implementing übercal(-ish)

$$ZP = m_2 - A_2 + C_2(t) \cdot (J - K_s)_2 +$$
$$+ \kappa(x, y) \cdot (\chi - 1) + \Delta m_{CHIP}(t) + m_i + f(x, y, t)$$



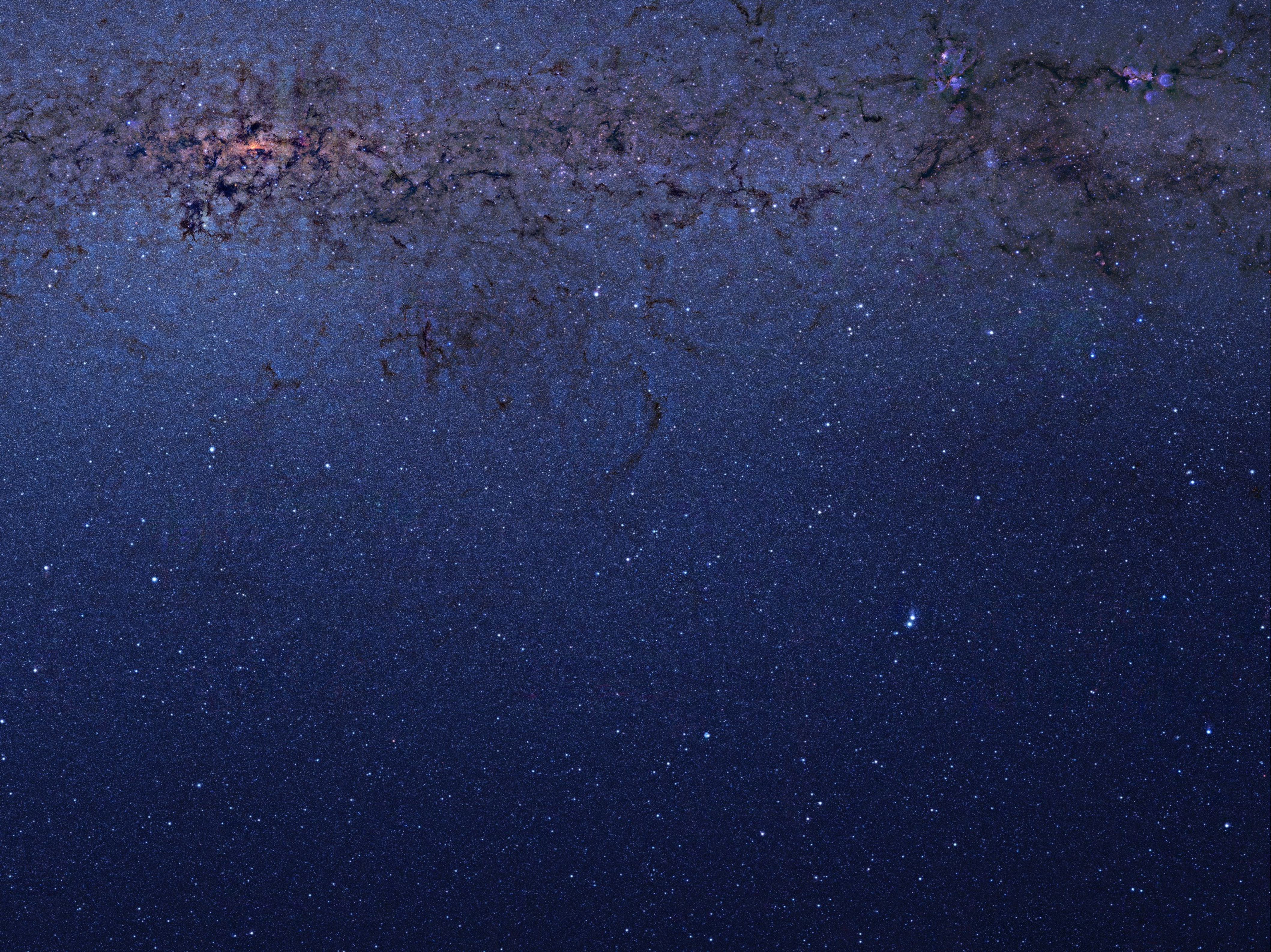
Implementing übercal(-ish)

But it comes at a price:

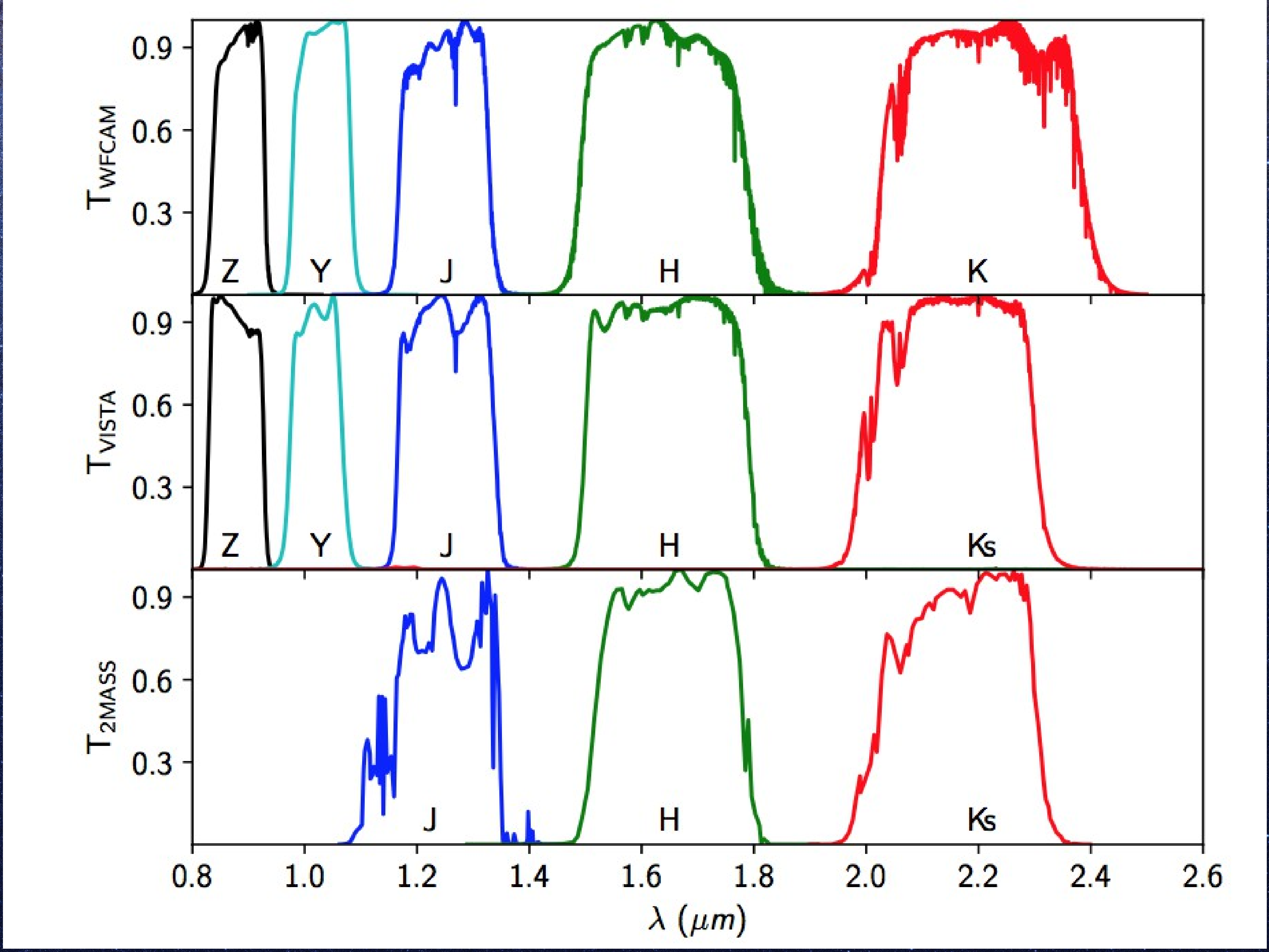
- Although simple code, computationally demanding:
  - Crossmatching & building matrixes can take several days.
  - Some DB design may be required.
- You need to wait until enough observations are taken, or even until they are totally finished.
- It requires a careful planning of the observations, sometimes in a way that is not PI science-optimized.

Some incomplete conclussions

- Classical calibration is required, it offers real-time monitoring and quick science, including ToO programs.
- Classical schemes are complicated to push below precissions & accuracies of a few percent.
- Übercal-like schemes offer added value products once observations/surveys are completed.
- Übercal is the only realistic way to break the 1% precission threshold.
- Accuracy is limited by the availability of true standards.

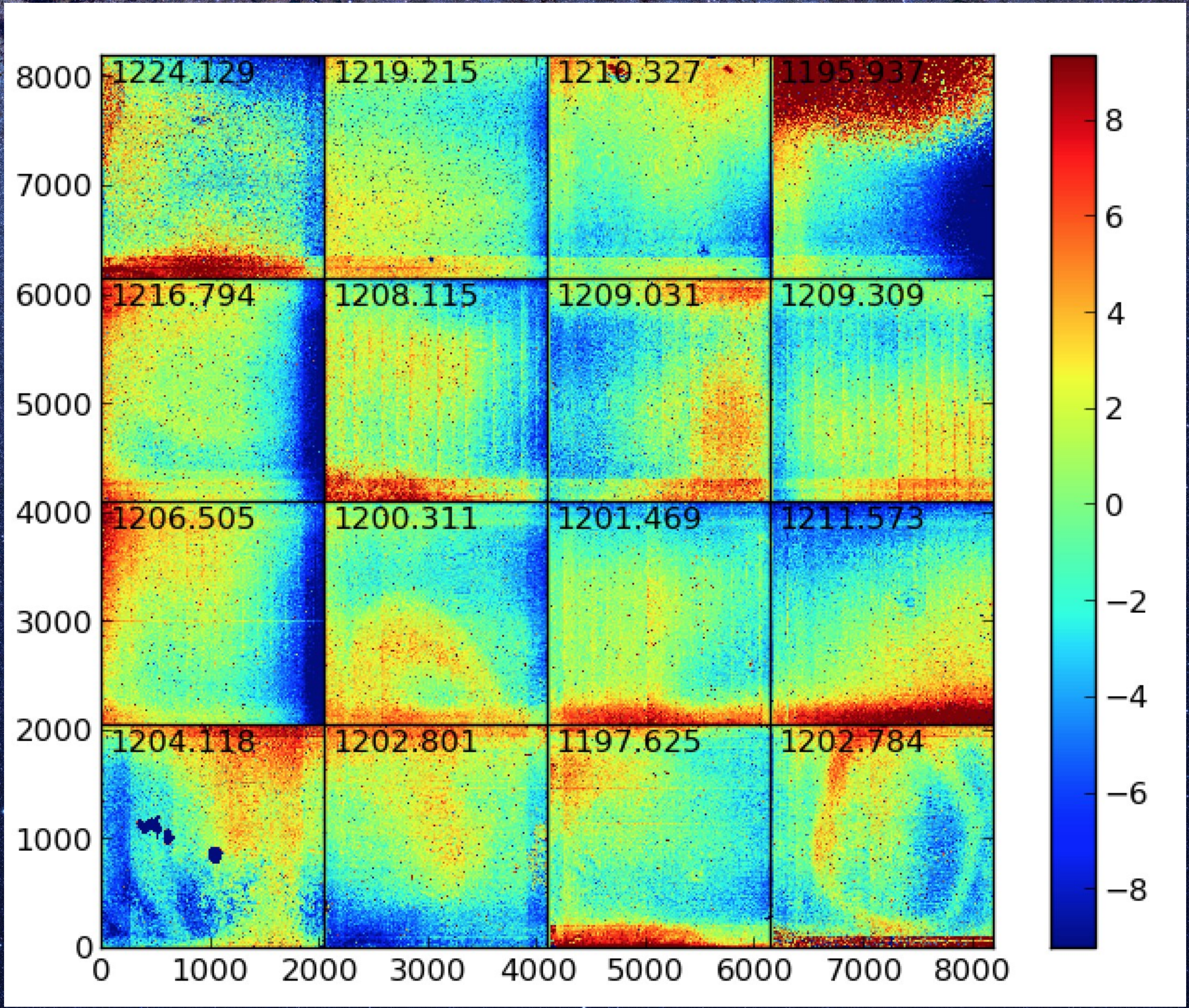


Extra



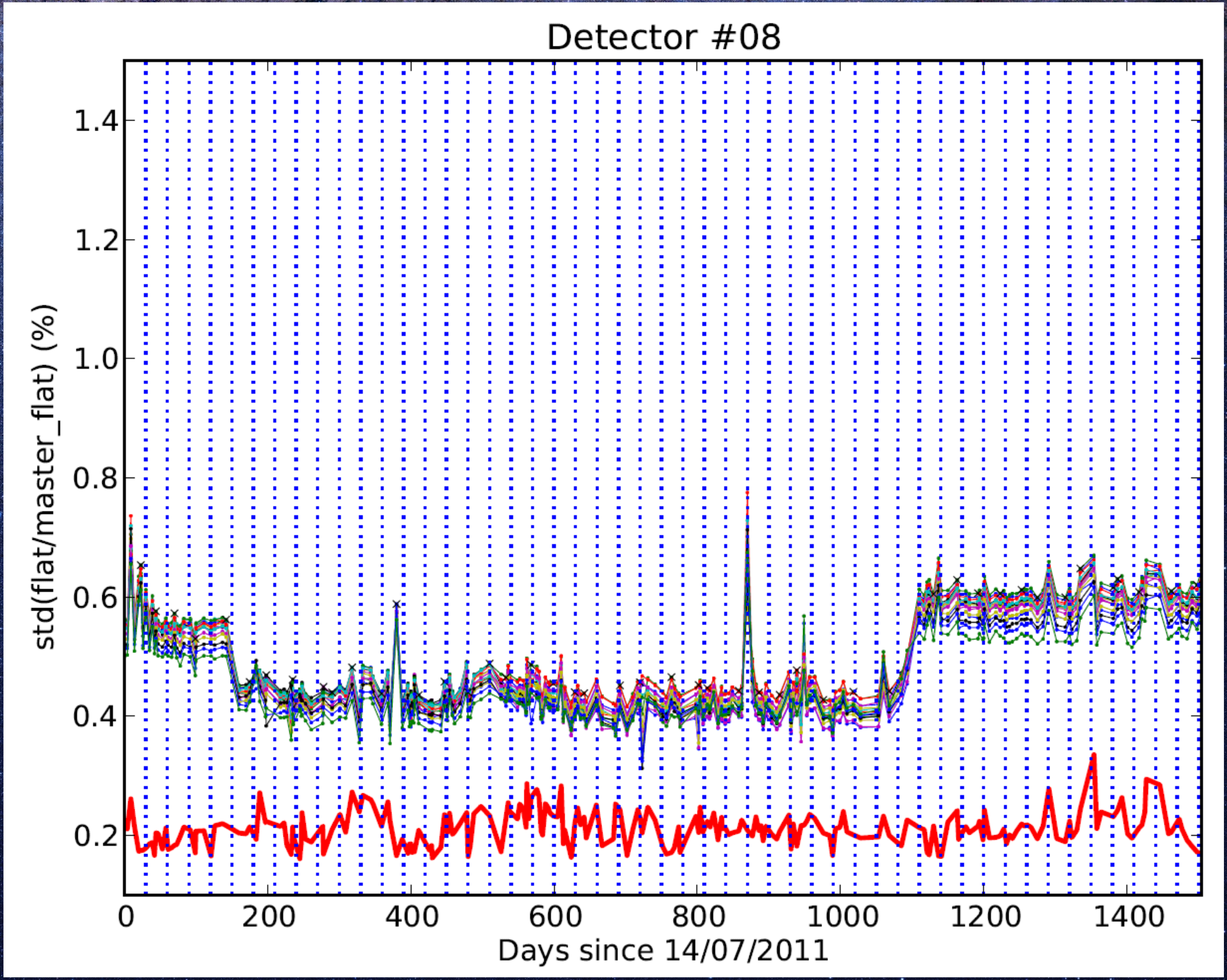
T_{WFCAM}
T_{VISTA}
T_{2MASS}
0.9
0.6
0.3
Z
Y
J
H
K
Ks
0.8
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
λ (μm)

Extra



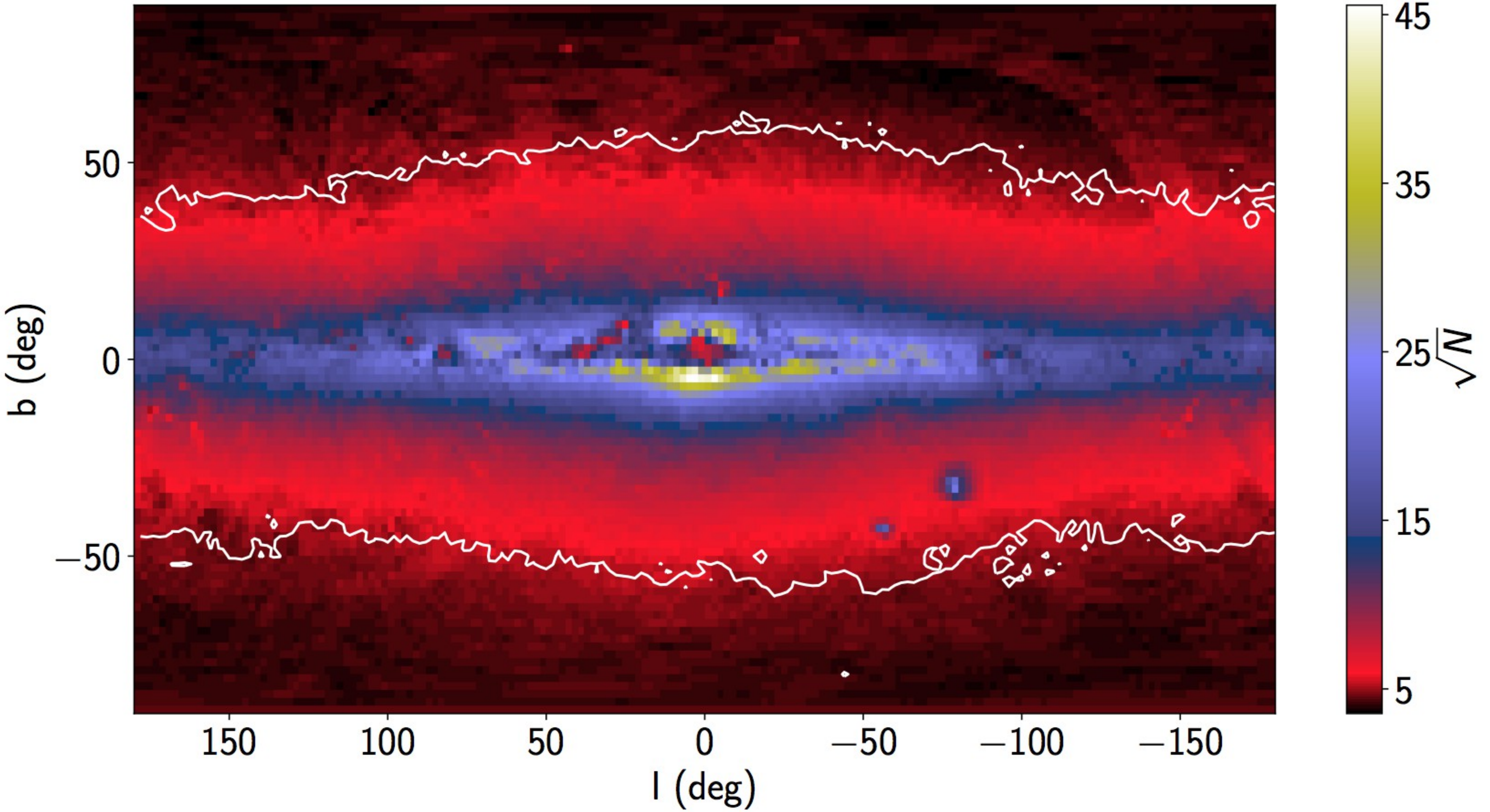
1224.129
1219.215
1210.327
1195.937
1216.794
1208.115
1209.031
1209.309
1206.505
1200.311
1201.469
1211.573
1204.118
1202.801
1197.625
1202.784
0
1000
2000
3000
4000
5000
6000
7000
8000
8
6
4
2
0
−2
−4
−6
−8

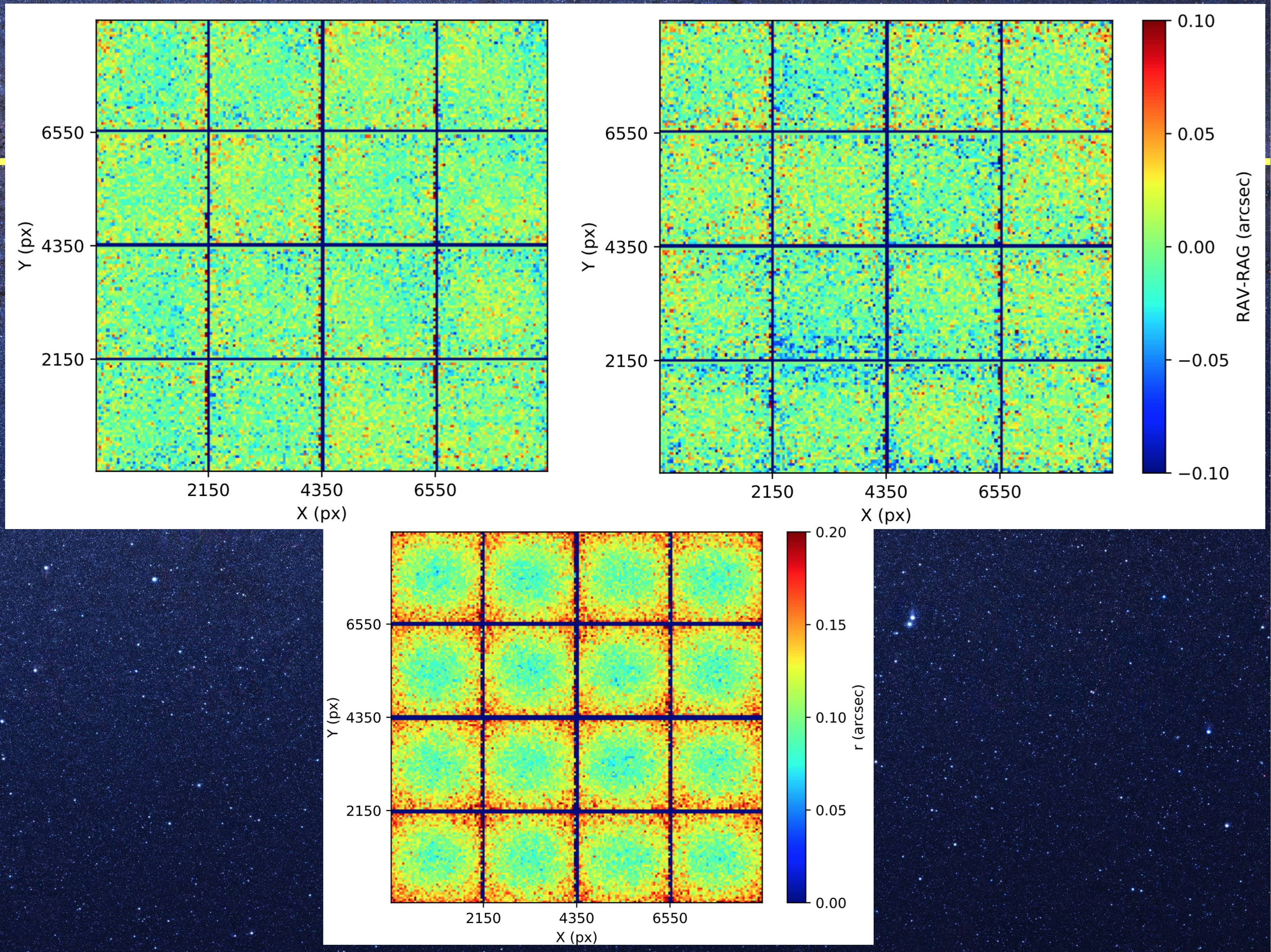
Extra



Detector #08
std(flat/master_flat) (%)
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0
200
400
600
800
1000
1200
1400
Days since 14/07/2011

Extra





6550
4350
2150
Y (px)
2150
4350
6550
X (px)
6550
4350
2150
Y (px)
2150
4350
6550
X (px)
0.10
0.05
0.00
−0.05
−0.10
RAV-RAG (arcsec)
6550
4350
2150
Y (px)
2150
4350
6550
X (px)
0.20
0.15
0.10
0.05
0.00
r (arcsec)