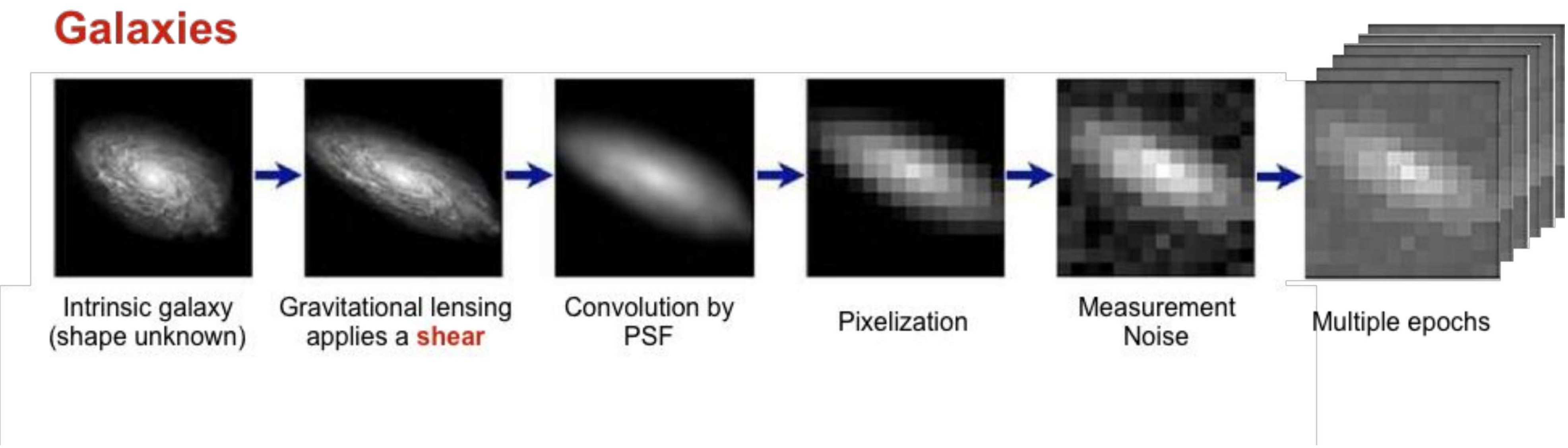
State of the Art In PSF Modeling

Mike Jarvis

St Patrick’s Day, 2020

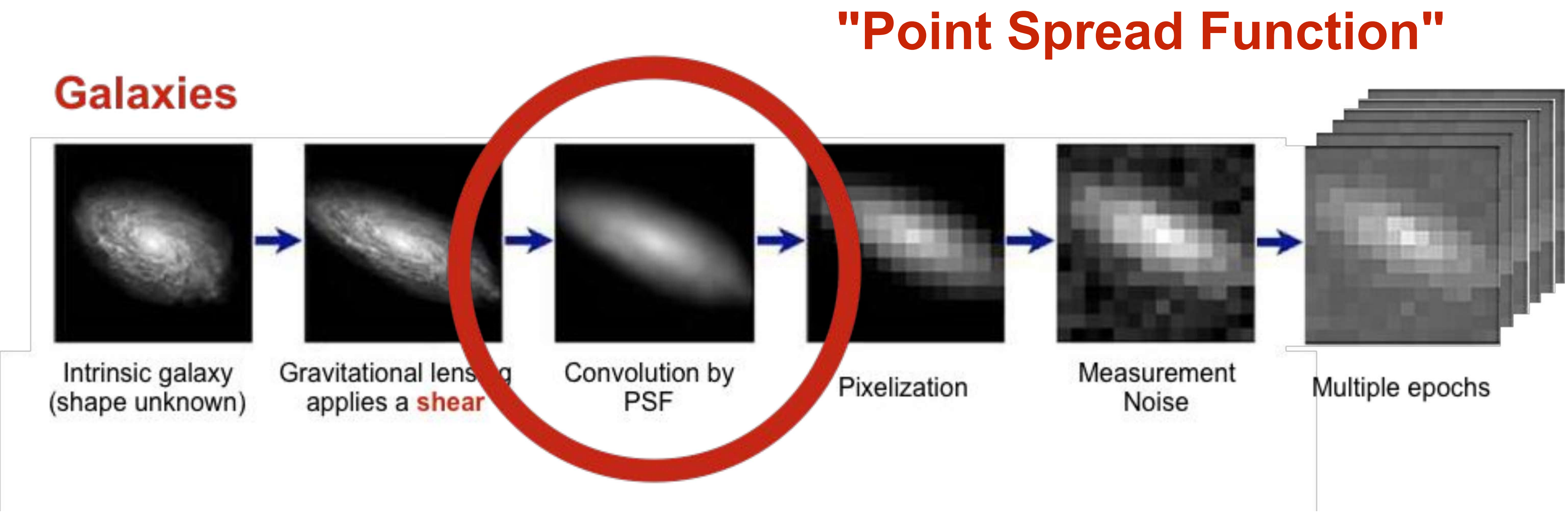
Rubin Observatory Algorithms Workshop

Observing shapes and fluxes of galaxies



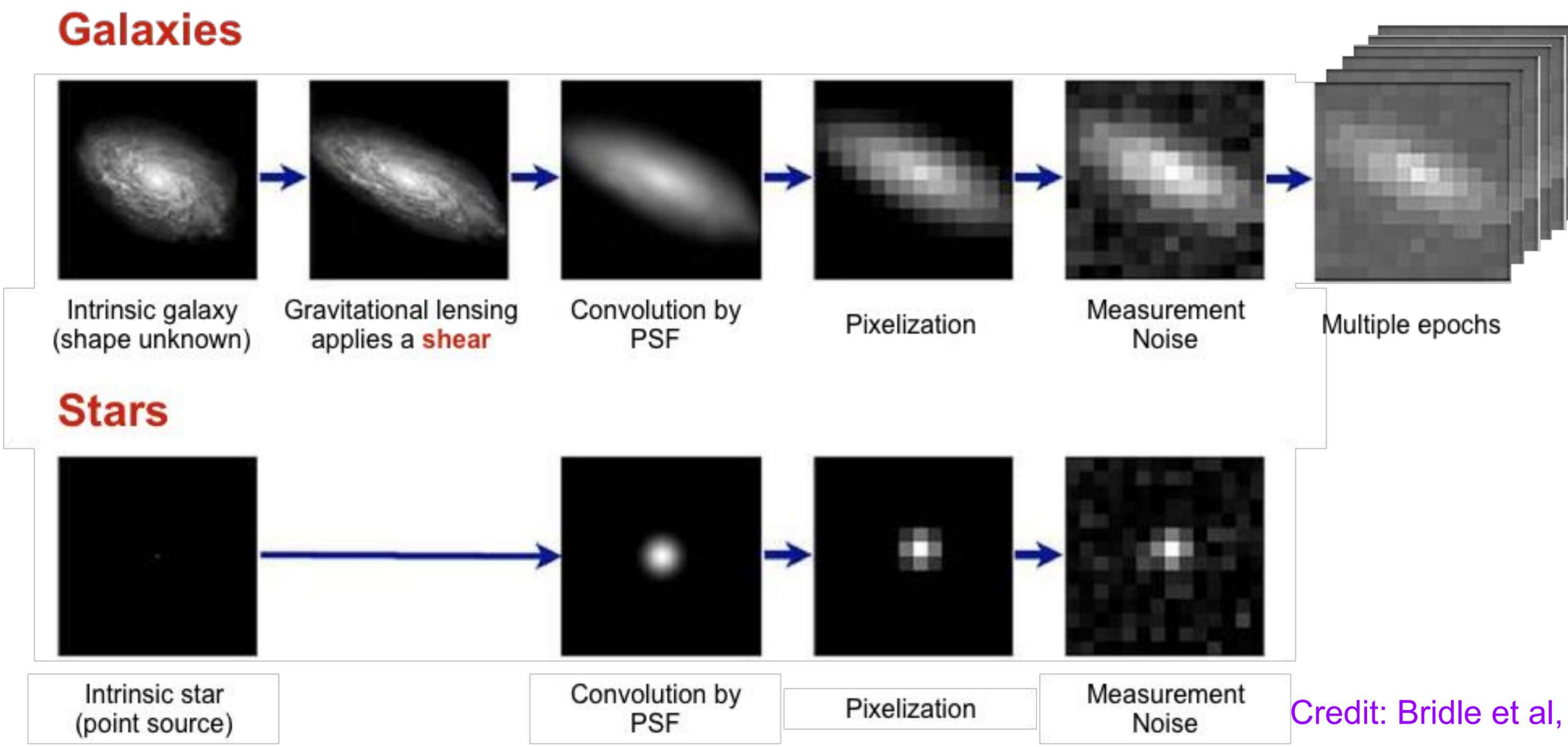
Credit: Bridle et al, 2008

Observing shapes and fluxes of galaxies



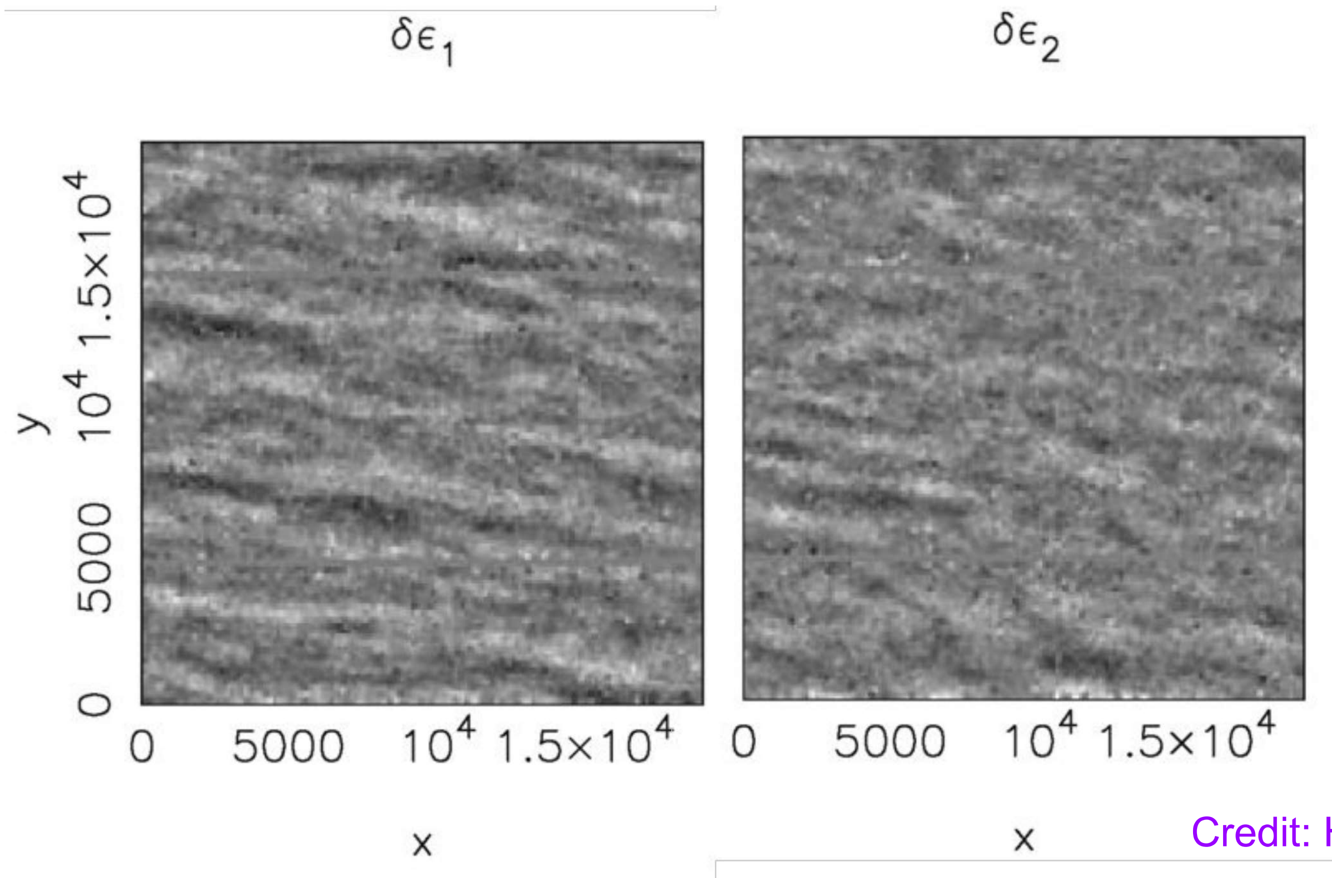
Credit: Bridle et al, 2008

Observing shapes and fluxes of galaxies



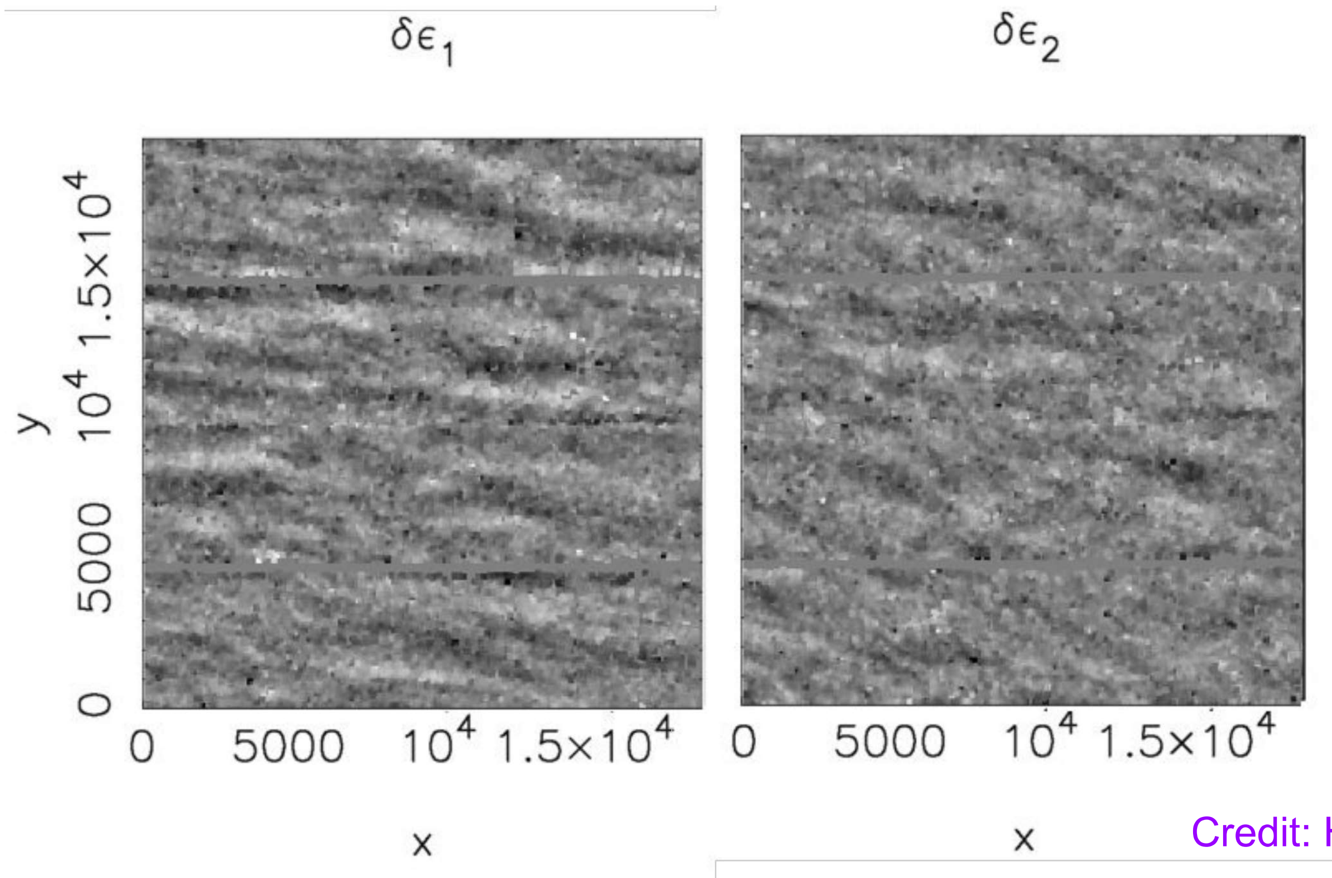
Credit: Bridle et al, 2008

Atmospheric PSF Variation



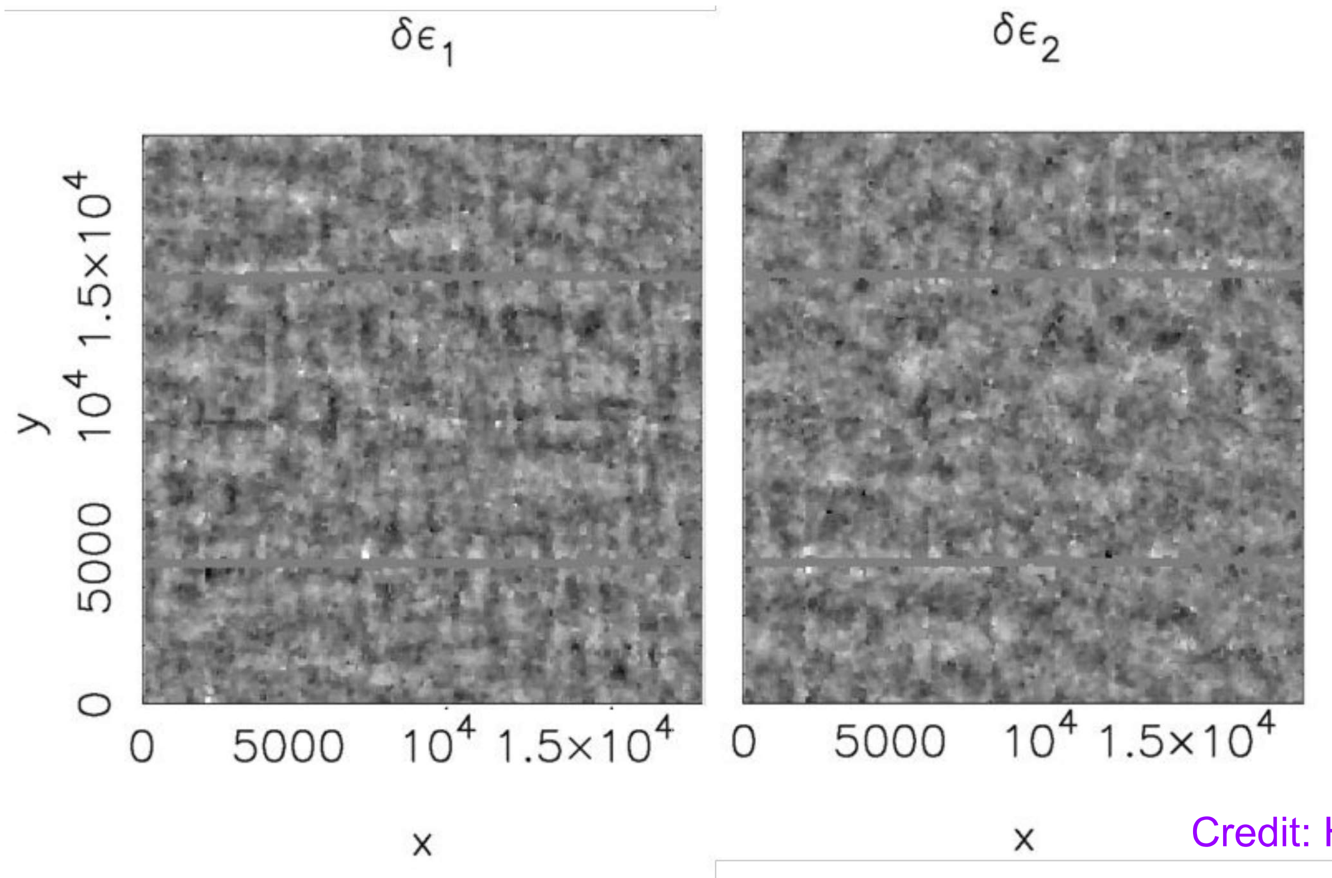
Credit: Heymans et al, 2011

Atmospheric PSF Variation



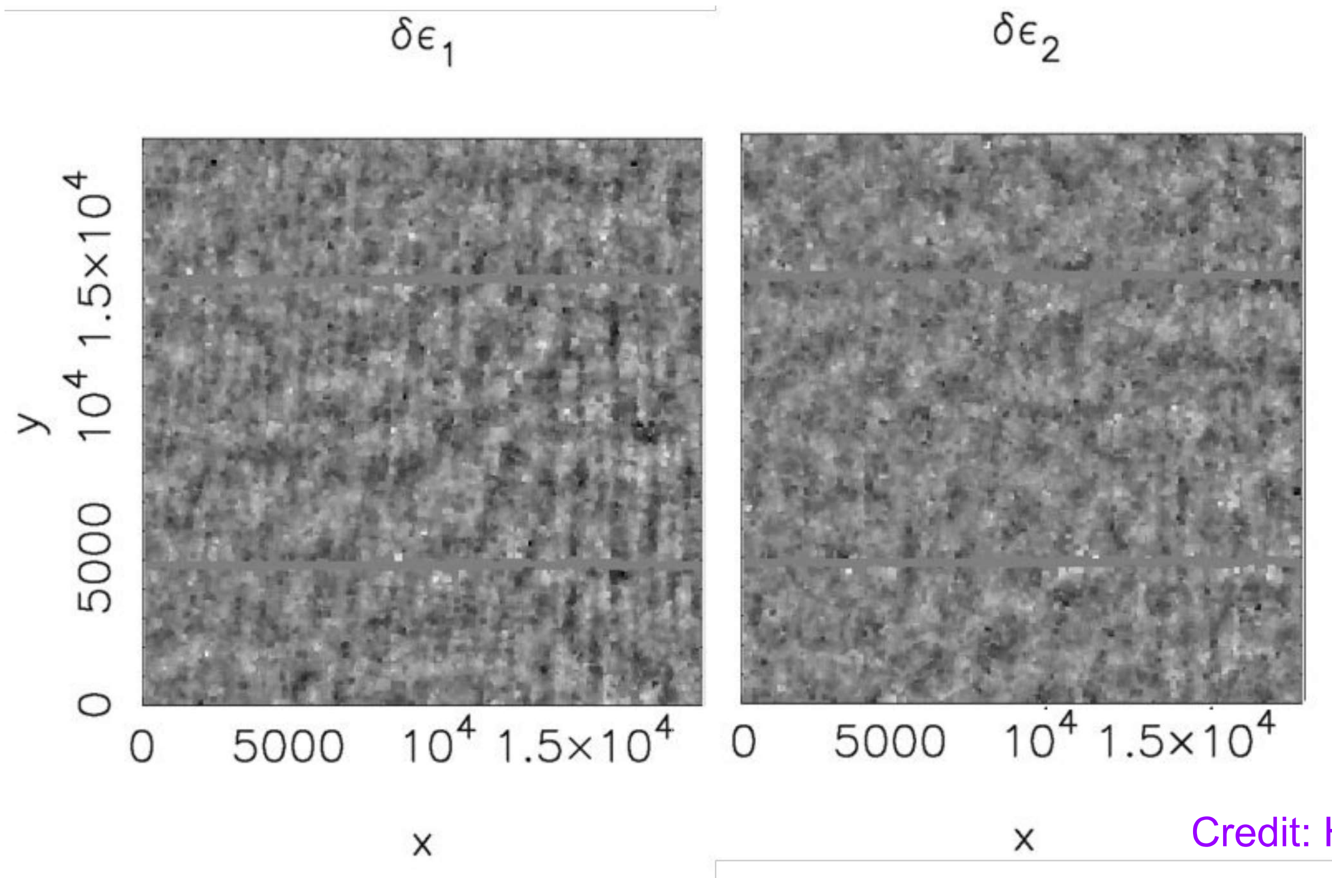
Credit: Heymans et al, 2011

Atmospheric PSF Variation



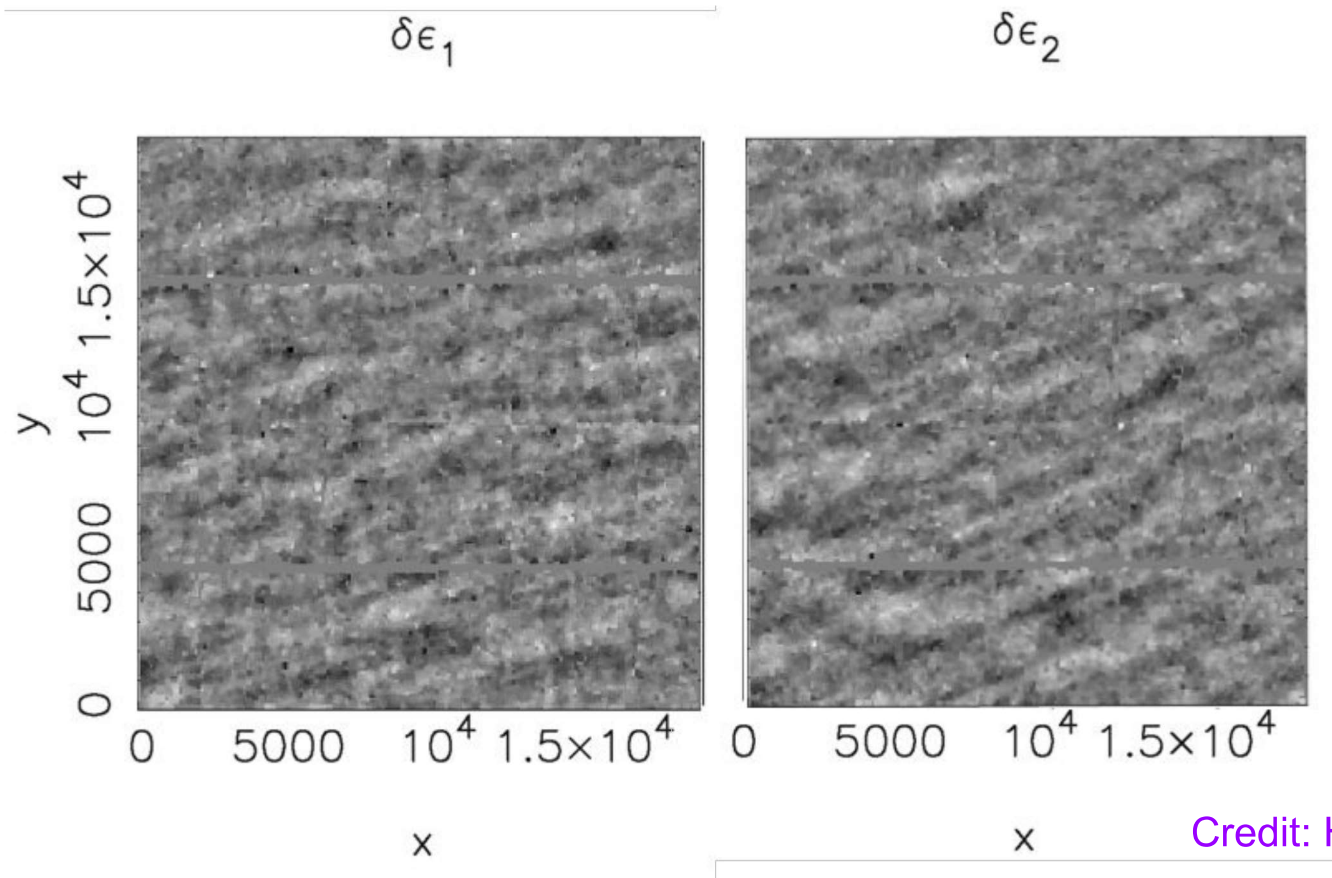
Credit: Heymans et al, 2011

Atmospheric PSF Variation



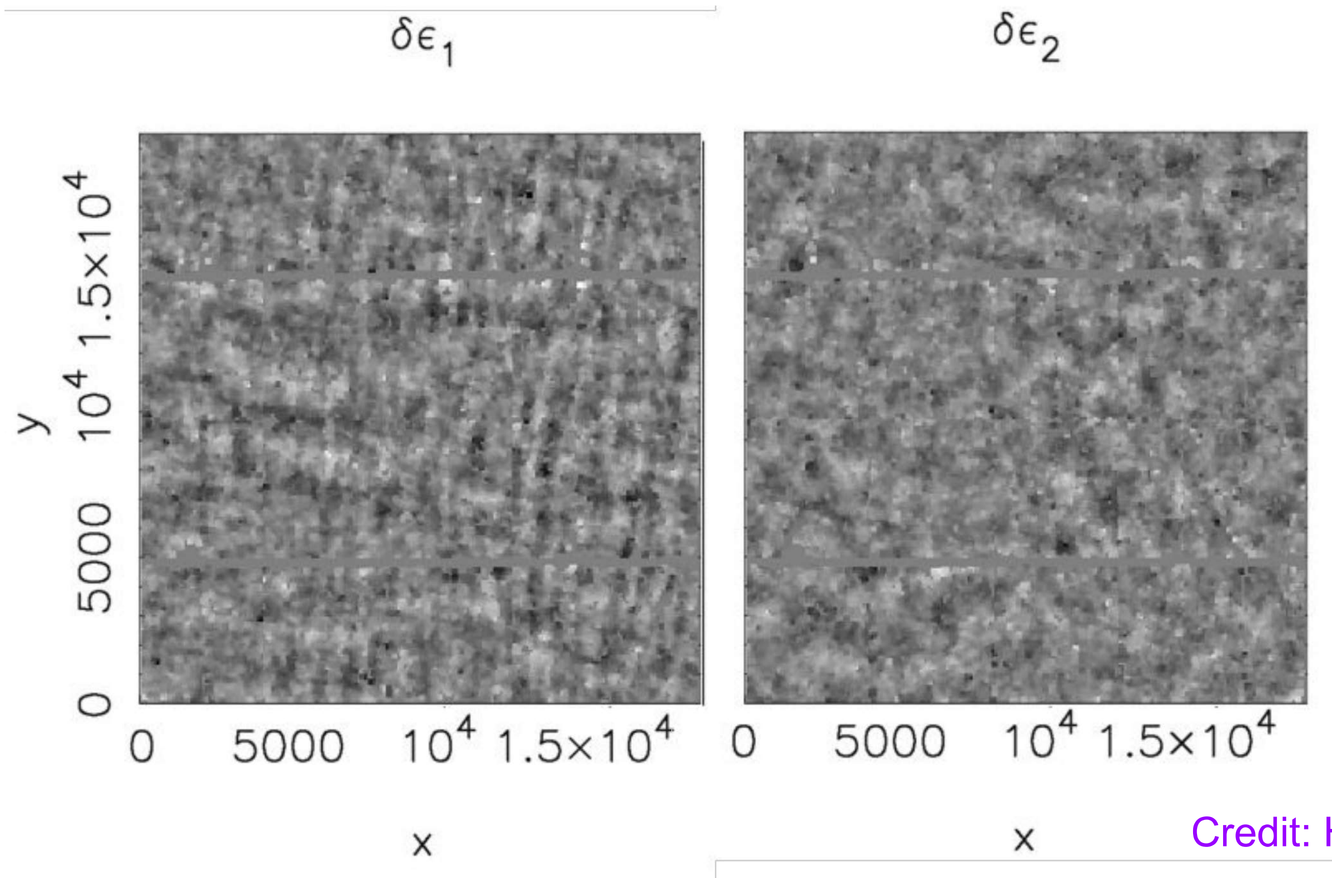
Credit: Heymans et al, 2011

Atmospheric PSF Variation



Credit: Heymans et al, 2011

Atmospheric PSF Variation



Credit: Heymans et al, 2011

Optical PSF Variation

DECam Wavefront PSF Model

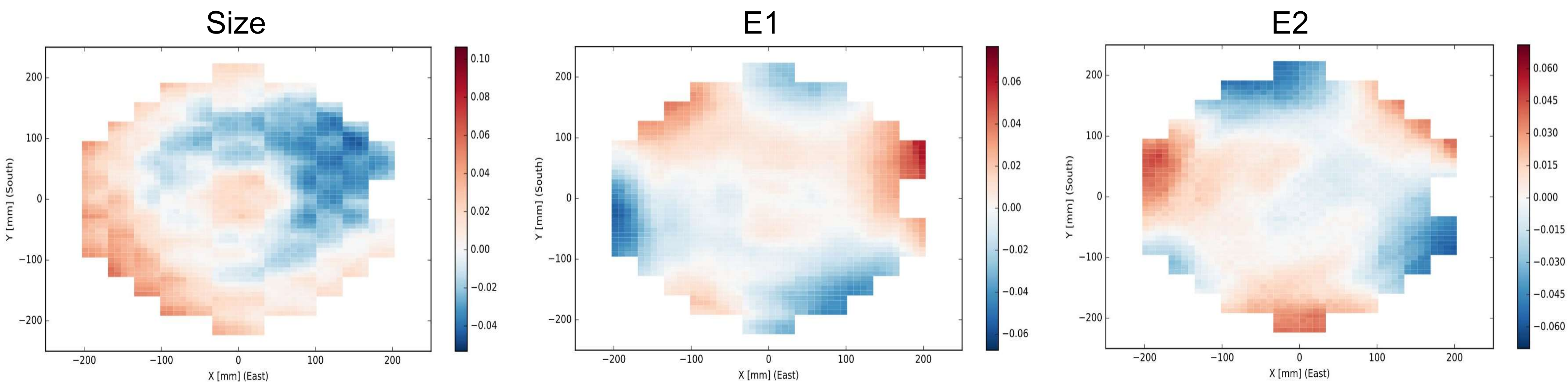


Image Credit: C. Davis

Optical + Atmospheric PSF Variation

DECam Observed Stars

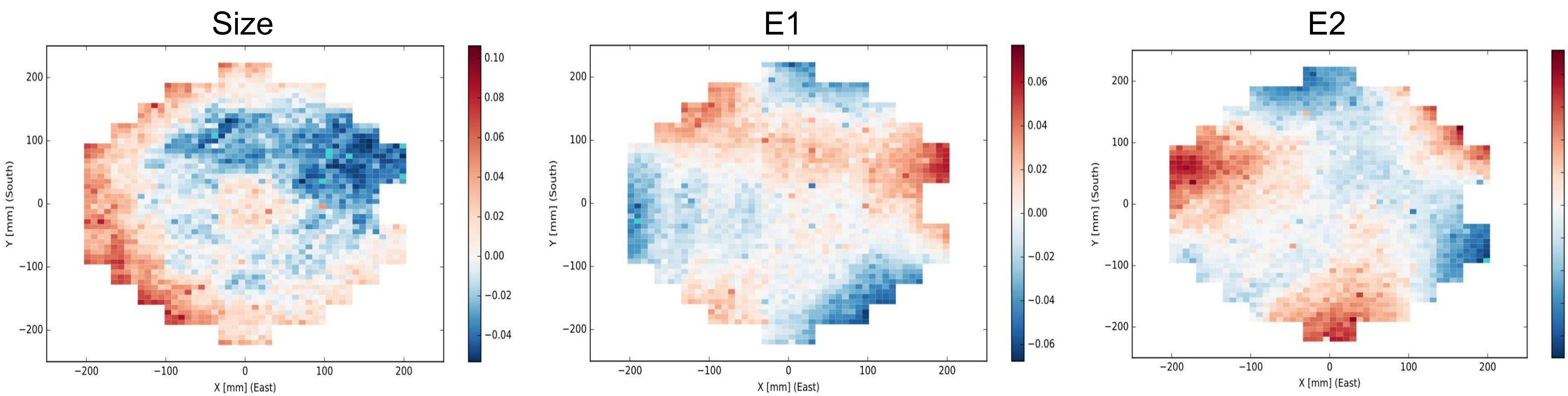


Image Credit: C. Davis

PSFEx, developed by Emmanuel Bertin, is currently the most widely used software for modeling the PSF.

- Used by DES SV and Y1 analyses
- Used by DES Y3, except for Weak Lensing applications.
- Used (in modified form) by HSC analyses
- Lots more: GAMA, BCS, WISCy, NGVS-IR, VISTA Orion A, ...

- Models the PSF as a linear combination of basis vectors.
- Typically the basis vectors are small images (vectors of pixel values).
- Interpolates these basis vectors using a polynomial across CCD.

Piff is a PSF estimation software package being developed initially for DES, but now also targeting LSST simulations. Specifically, the goal is to improve upon the current state of the art, PSFEx, in terms of PSF residuals.

Contributors (so far): Mike Jarvis, Chris Davis, Pierre-François Leget, Erin Sheldon, Josh Meyers, Gary Bernstein, Aaron Roodman, Pat Burchat, Daniel Gruen, Ares Hernandez, Andres Navarro, Flavia Sobreira, Reese Wilkinson, Joe Zuntz, Sarah Burnett, Alex Drlica-Wagner, Eric Charles

Improvements of Piff over PSFEx:

- Modular design makes it easier to allow for different models for surface-brightness profile an interpolation scheme.
- Models PSF profile in “sky” coordinates (local tangent plane) rather than pixel coordinates, which allows the WCS (including, e.g., tree rings) to be removed before PSF interpolation.
- Can use all CCDs in an exposure as part of a single solution.
- Uses iterative improvement, which helps with missing data.
- Can include color terms in PSF model.
- Includes hooks for a mode that would “learn” hyper parameters from multiple exposures, which can inform solutions for individual exposures.
- Includes code to read and use the output file.

Available Models

- Gaussian = Gaussian profile with fitted size, ellipticity
- Moffat = Moffat profile with fixed beta and fitted size, ellipticity
- Kolmogorov = Kolmogorov profile with fitted size, ellipticity
- Any Galsim GSObject (also with fitted size, ellipticity)
- PixelGrid = 2d grid of pixels representing the surface brightness profile. Note that the pixels do not need to match the size of the pixels in the data.
- Optical = arbitrary Zernike aberrations and pupil plane mask.

Available Interpolations

- Mean = simplest possible interpolation, just the mean PSF profile
- Polynomial = polynomial in u,v (sky coord) typically, but may provide any other terms to use for interpolation as well. E.g. color.
- BasisPolynomial = like Polynomial, but allows the fit to be delayed to compute the model parameters and the interpolation coefficients at once by solving a single matrix equation.
- kNNInterp = k-nearest neighbors, a simple non-parametric interpolation
- GPInterp = Gaussian process, optionally using PCA for components
- GPInterp2pcf = Gaussian process, using an anisotropic 2-point correlation function measured from the data directly as the GP kernel.

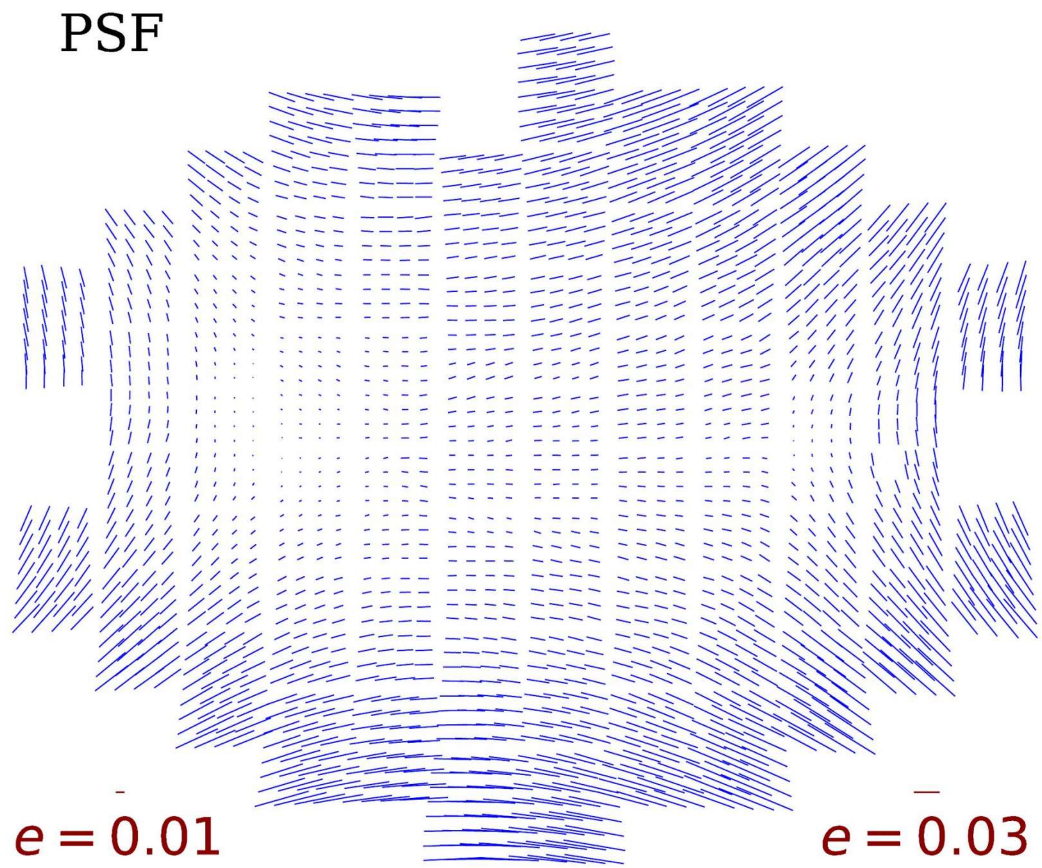
Other options and features

- Can use weight mask as noise level in each pixel
- Can apply signal portion of the Poisson noise based on model rather than realized pixel values
- Can reject input stars for various reasons (too faint, mask, neighbors, etc.)
- Can impose a maximum S/N so not dominated by a few bright stars
  - This doesn't remove the bright stars, but treats them as though their S/N is the maximum.
- Can use any WCS function available as a GalSim WCS type.
- Can reject outliers based on chisq of model and data after 2nd iteration
  - "Outliers" is a modular feature, so potentially other options for how to do this, but chisq is the only one implemented yet.

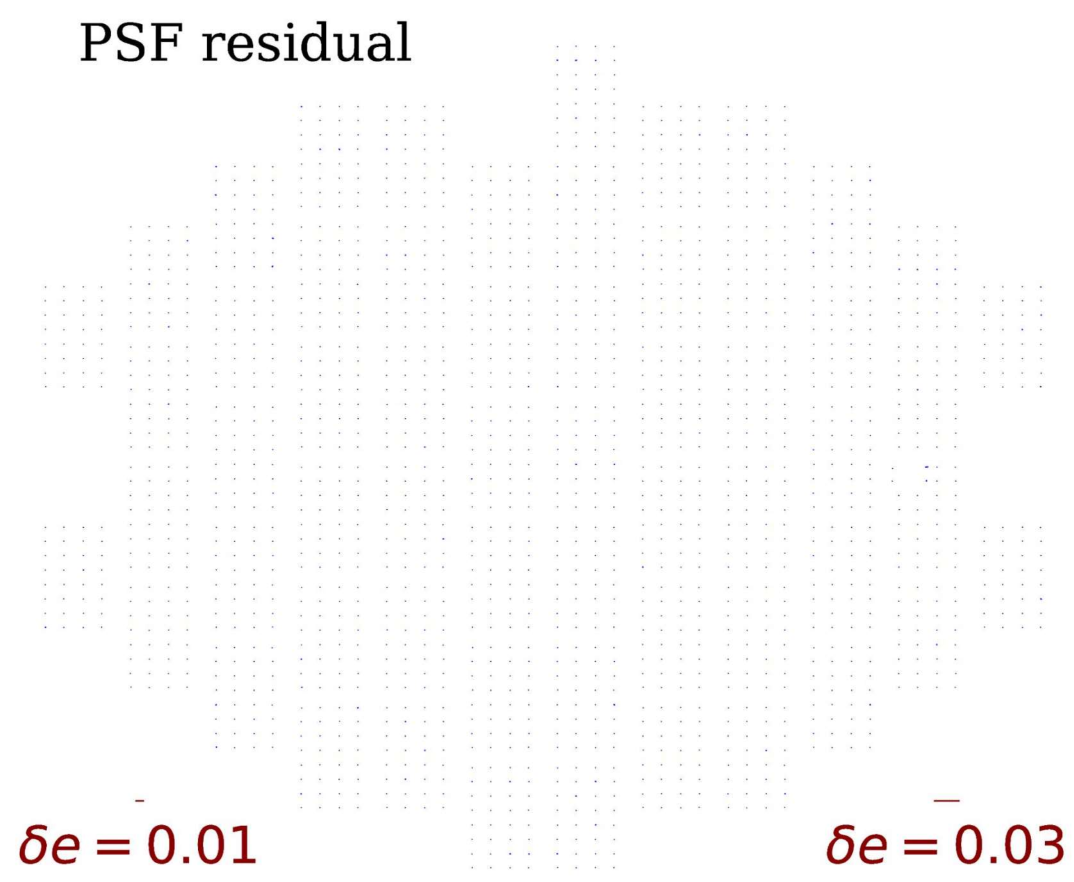
Configuration run on DES Y3 data

- PixelGrid model, which is similar to PSFEx in spirit (although not in detail). Model is a 17x17 grid of pixels, whose fluxes are independently solved for.
- Model pixel scale = 0.30 arcsec. Slightly larger than real pixels to help stabilize fit for images with few stars.
- Interpolation is a 3rd order polynomial. We delay the matrix solution to do all interpolant and model parameters at once, so can handle missing data.
- Our WCS (world coordinate system) comes from a global astrometric solution that includes edge distortion and tree rings (called pixmappy).
- We have a brighter/fatter correction in place, but remove the brightest 1.2 magnitude of stars where the correction is not quite good enough.
- We reserve 20% of good stars to use for testing the quality of the fits.

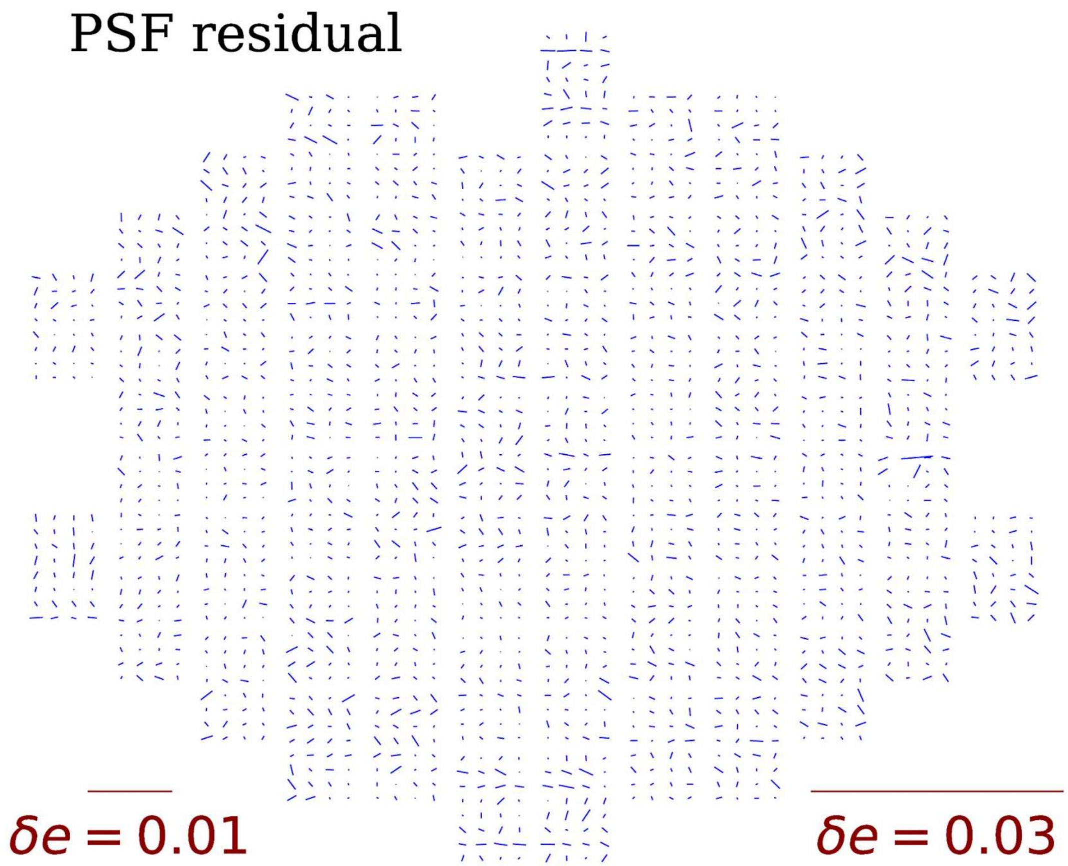
DES Y3: Mean raw PSF pattern



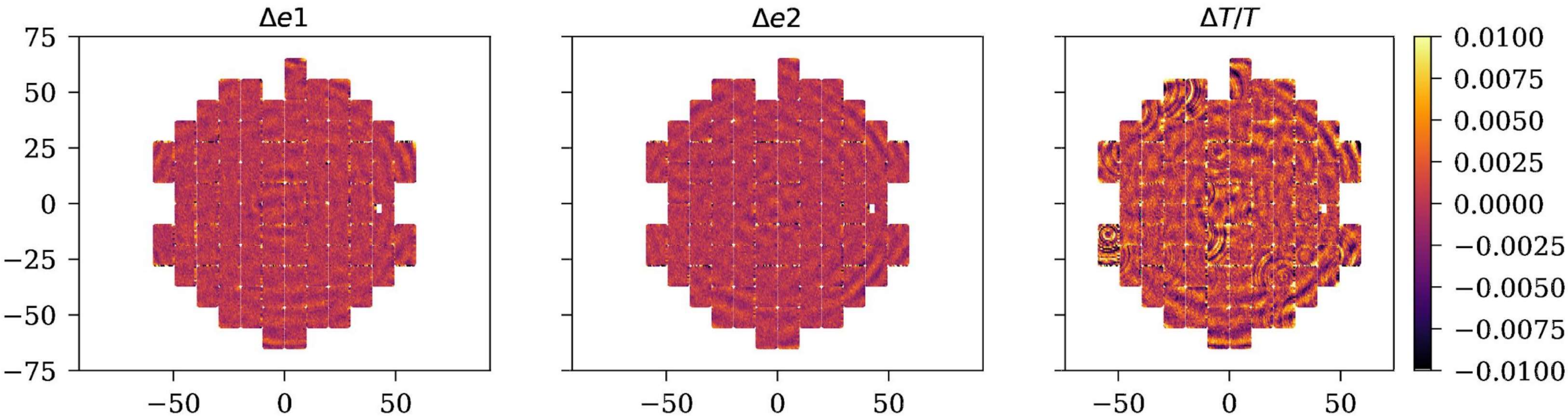
DES Y3: Mean residuals by position



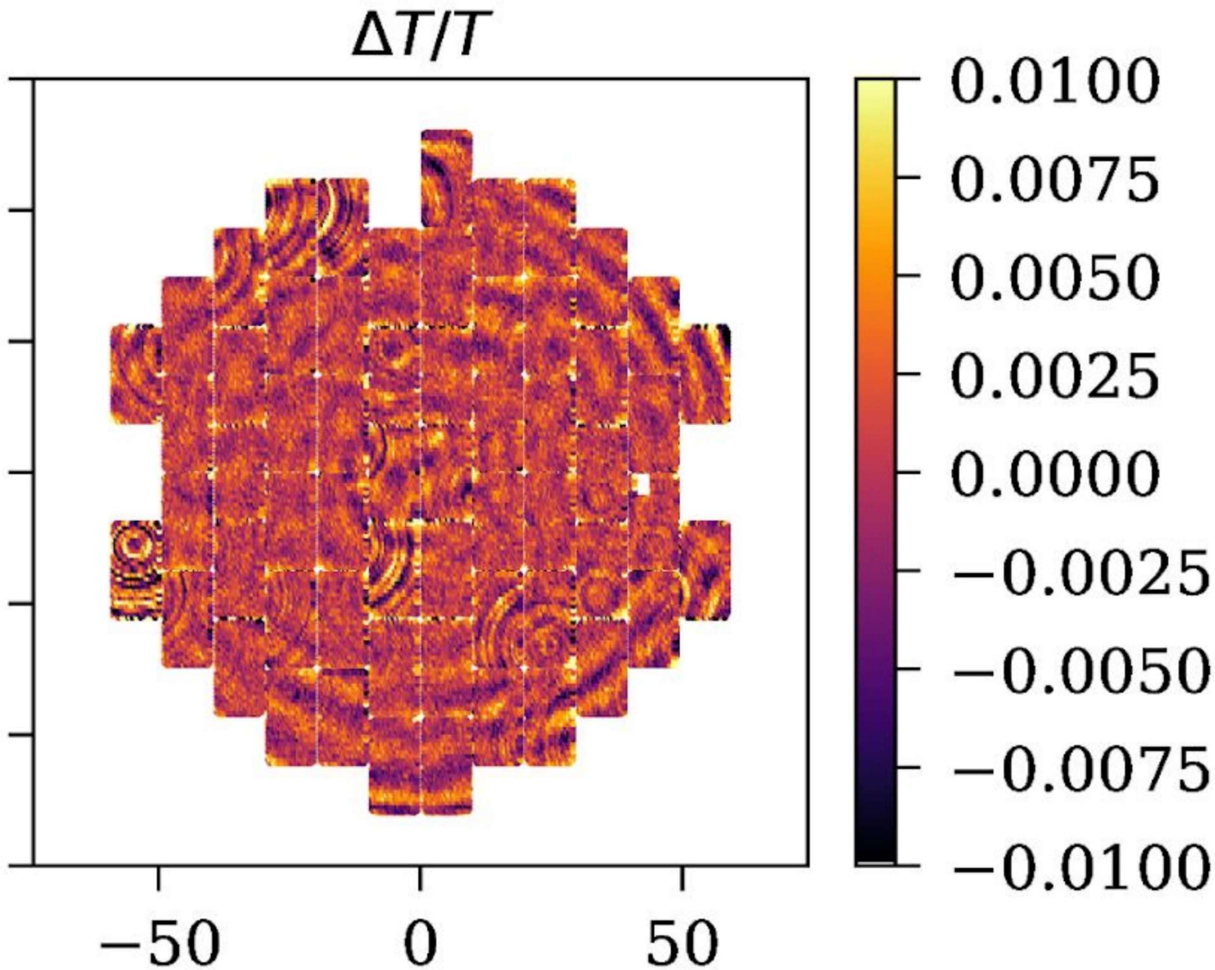
DES Y3: Mean residuals by position



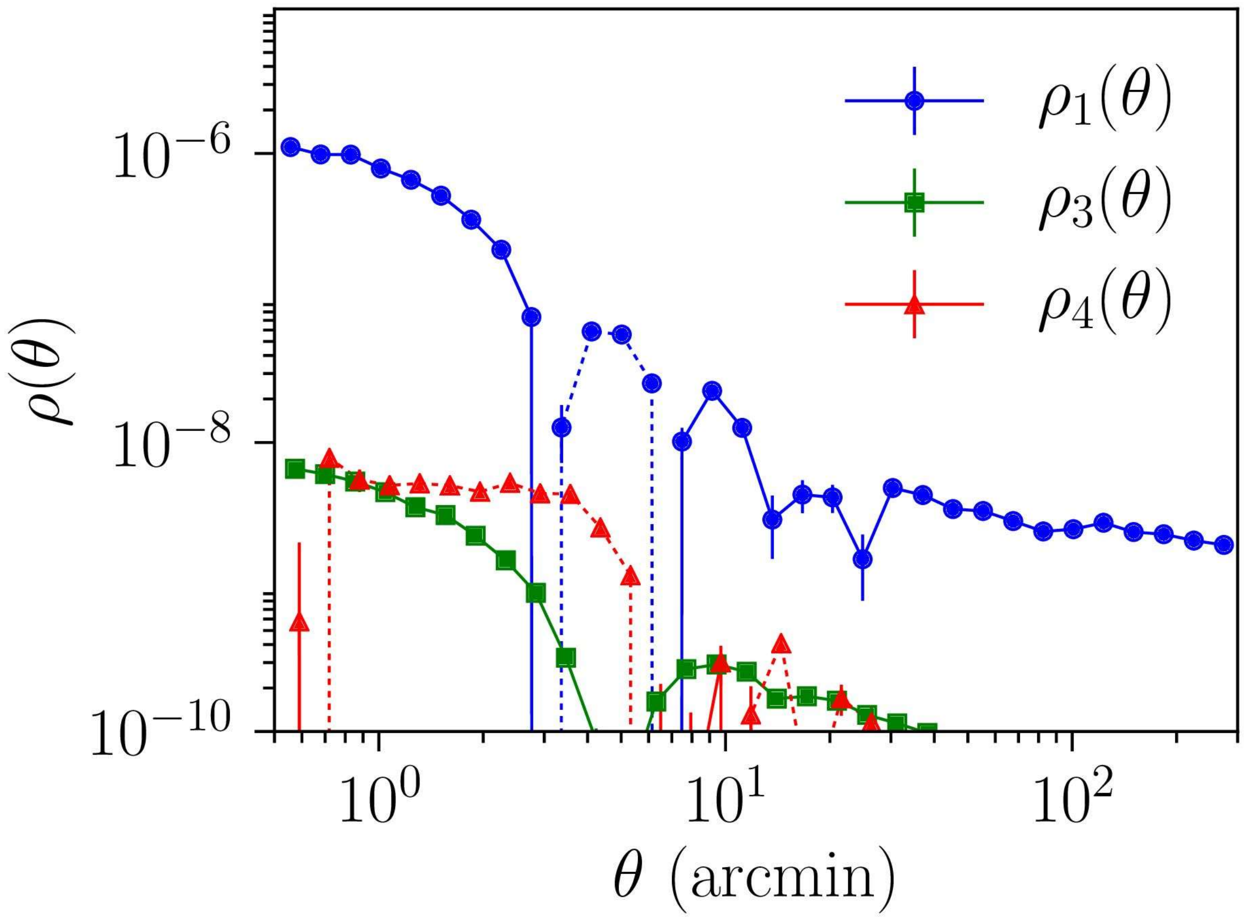
DES Y3: Mean residuals by position



DES Y3: Mean residuals by position



DES Y3: Rho statistics (RIZ combined)



Measure shapes and sizes of PSF model and data for reserve stars.

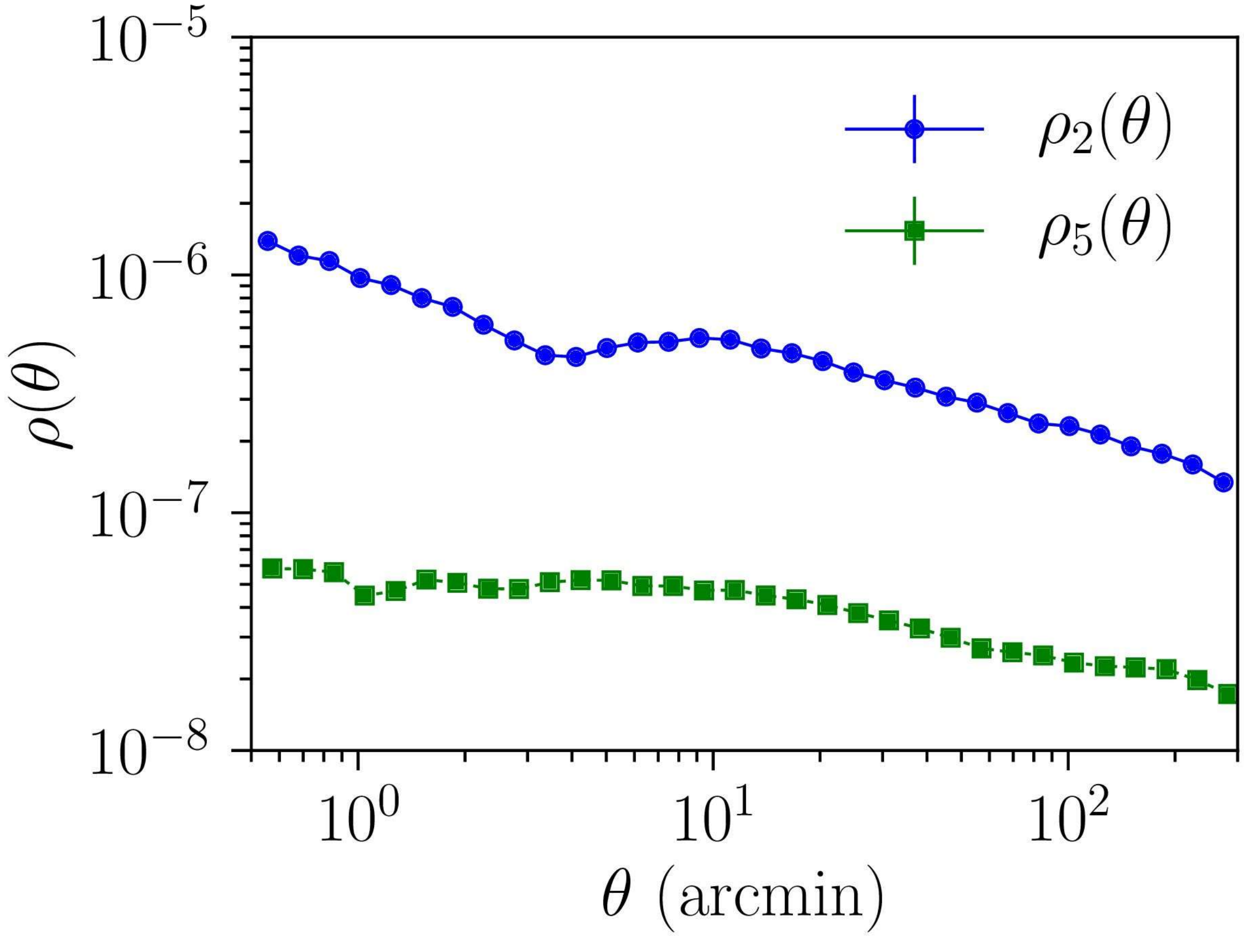
Rho1 is auto-correlation of the residual shapes de = (e_piff - e_star)

Rho3 is auto-correlation of residual sizes times PSF shape e_star (T_piff - T_star) / T_star

Rho4 is the cross correlation of these two values.

All three are additive systematics in xi+ modulated by a factor of order unity.

DES Y3: Rho statistics (RIZ combined)

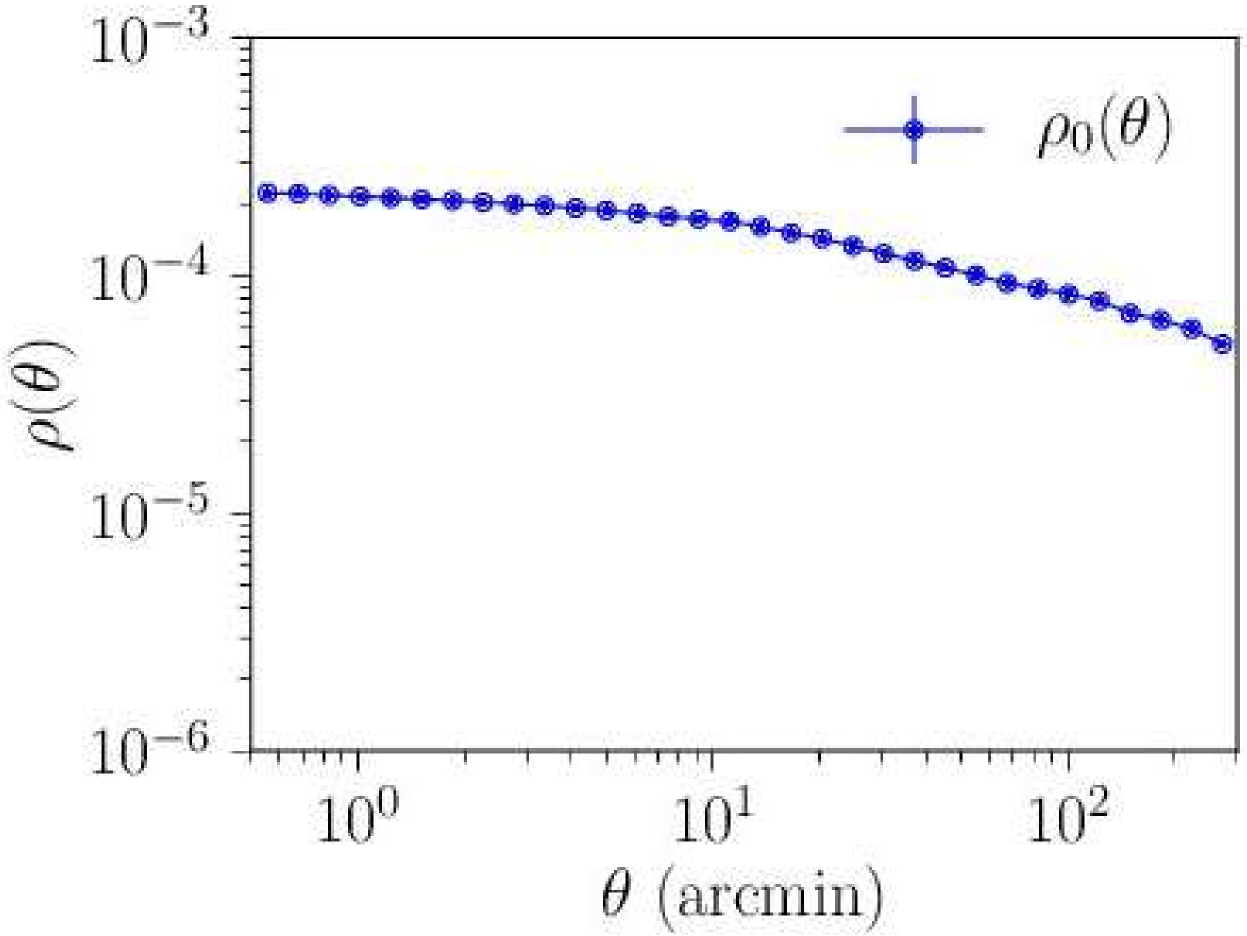


Rho2 is cross-correlation of the residual shape de with e_star

Rho5 is cross-correlation of residual size edT/T with e_star

These are additive systematics in xi+, but modulated by “leakage” factor, which we call alpha, that we think is ~0.01 (depends on the shear method used though).

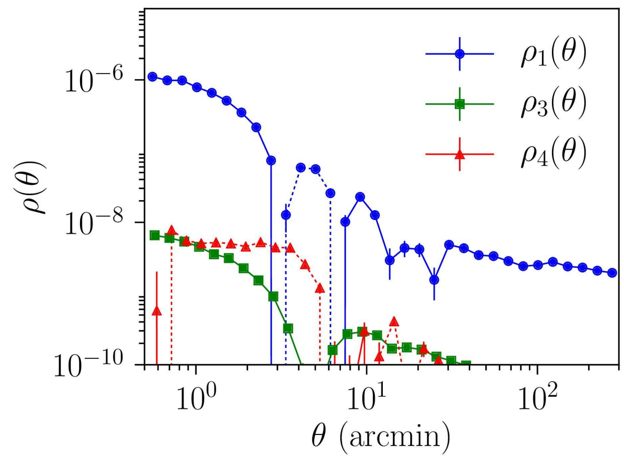
DES Y3: Rho statistics (RIZ combined)

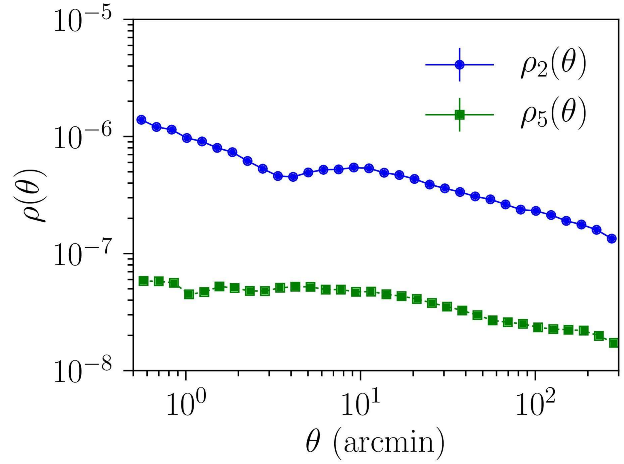


For comparison, rho0 is what we call the auto-correlation of the real PSF shapes, e_star.

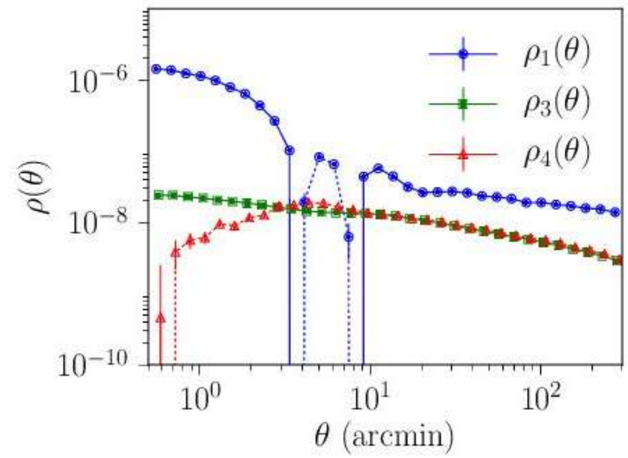
Compare with PSFEx models

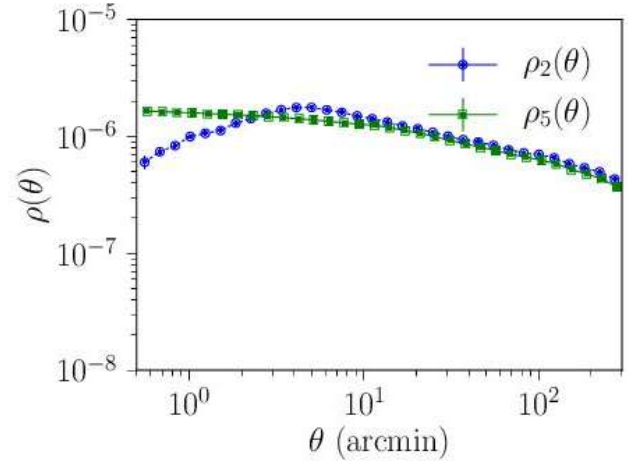
Piff





PSFEx



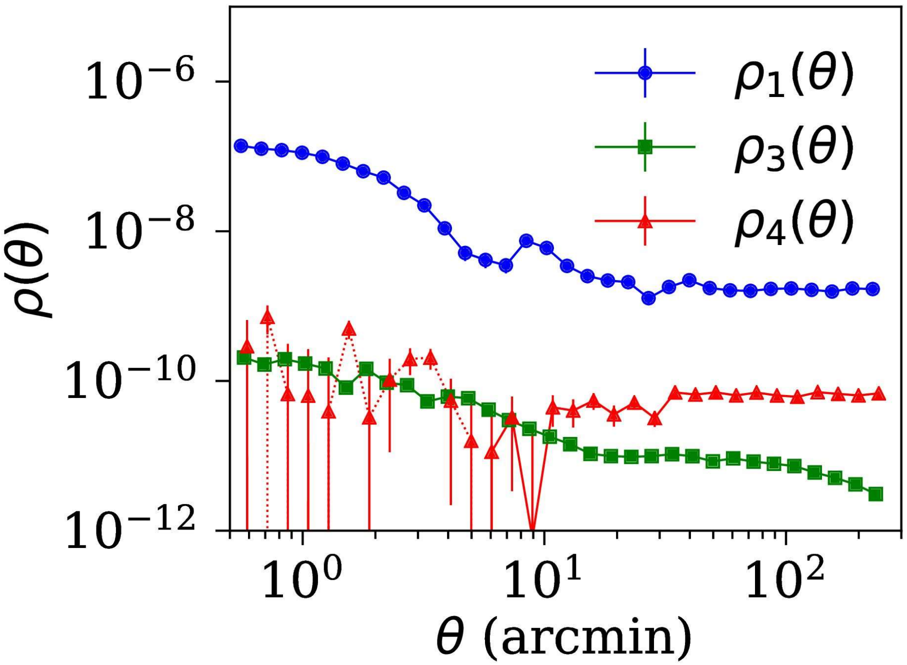


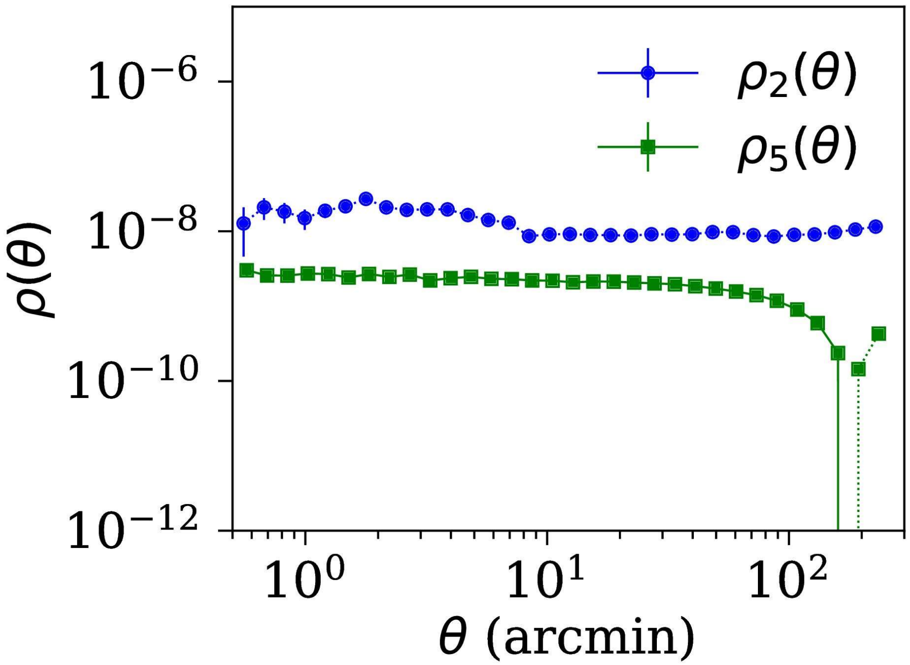
DC2 Simulation -- Simulated LSST Images

- Use truth “Gaia” catalog for input stars
  - Used LSST g band limits, which is not Gaia G band. Oops.
  - Remove saturated stars and gross outliers immediately on input.
- Fit PSF independently for each CCD
- Use “Pixel Grid” model at the location of each star.
- Interpolate with 2nd order polynomials.
- Delay solution to fit both model and interpolation coefficients at once.
- Finished 9202 of 9271 visits. (Other 69 have failures.)

DC2 Simulation

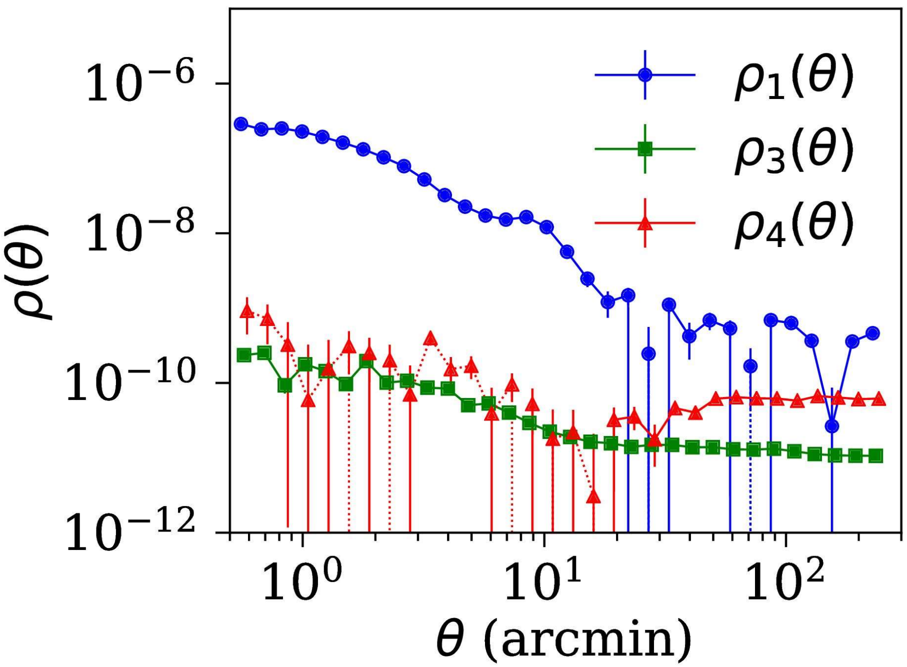
All bands (ugrizy)

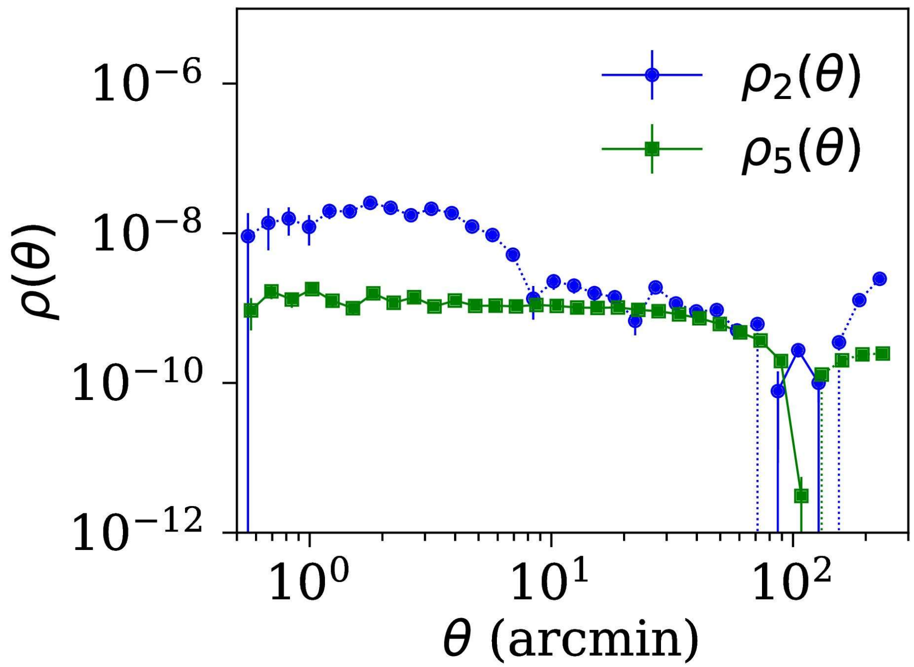




DC2 Simulation

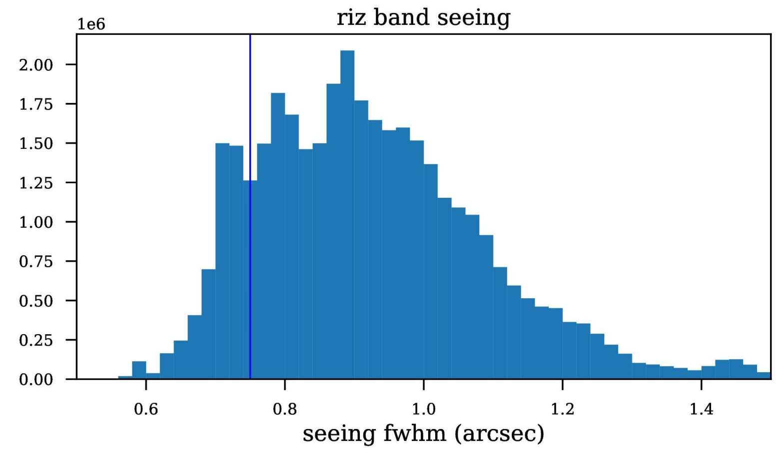
"Weak Lensing" bands (riz)

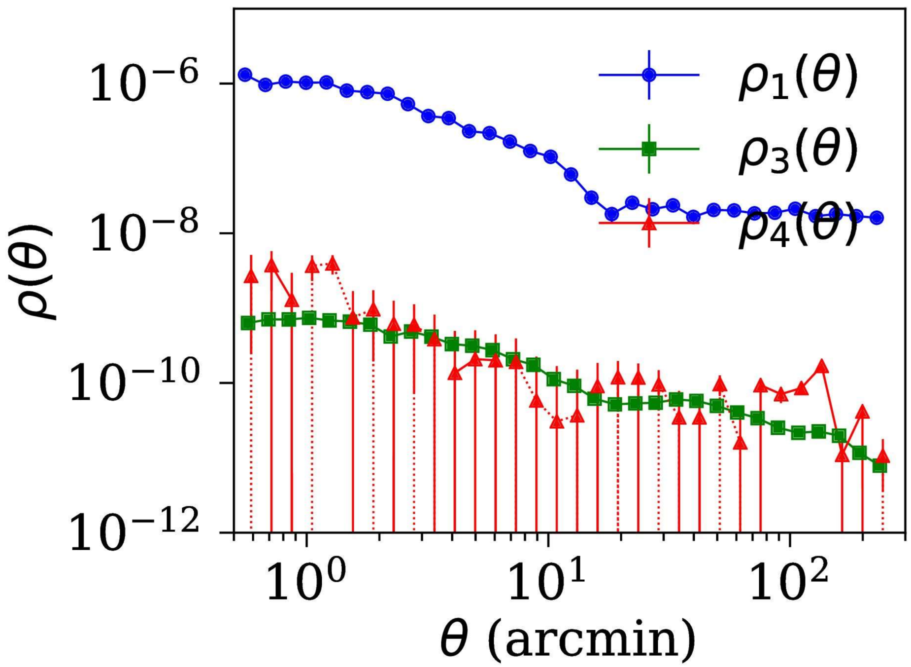


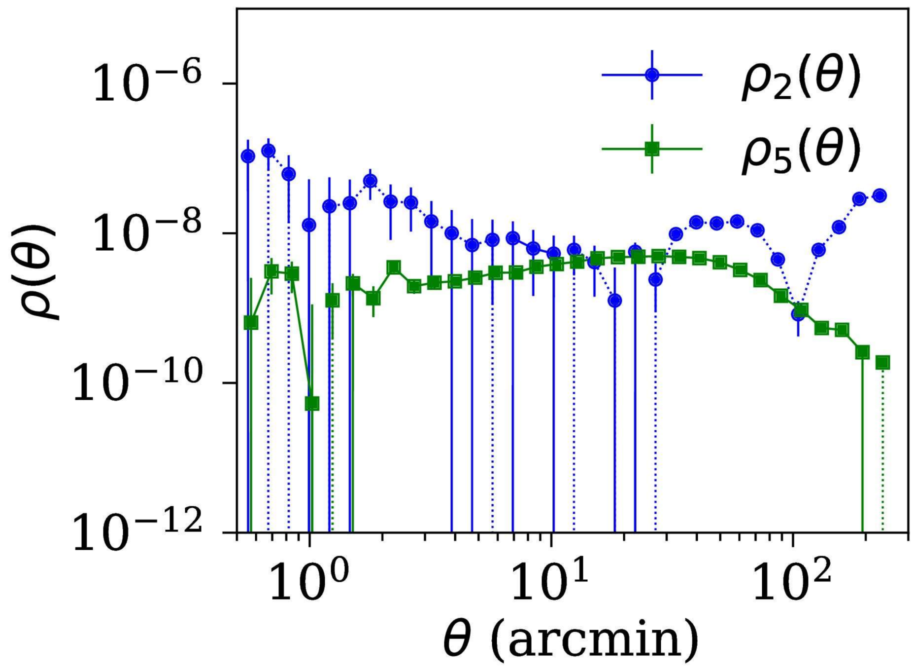


DC2 Simulation

Good seeing data
FWHM < 0.75
(17% of DC2 visits)

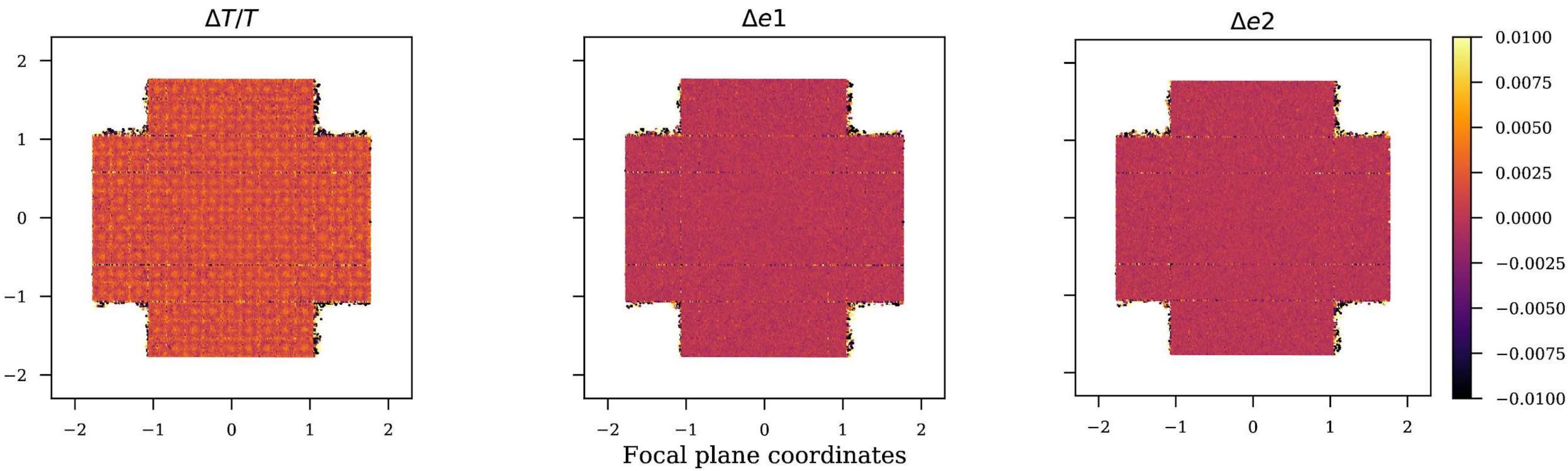


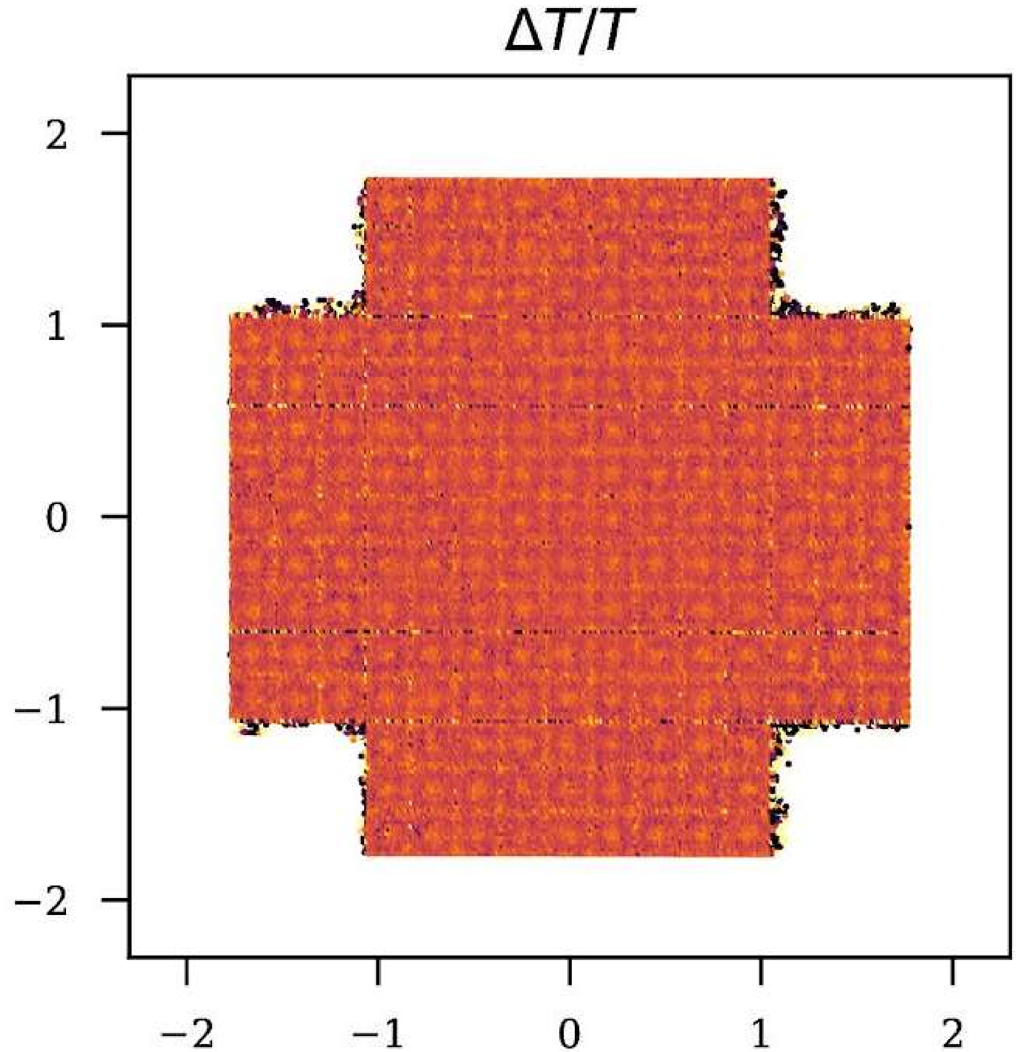




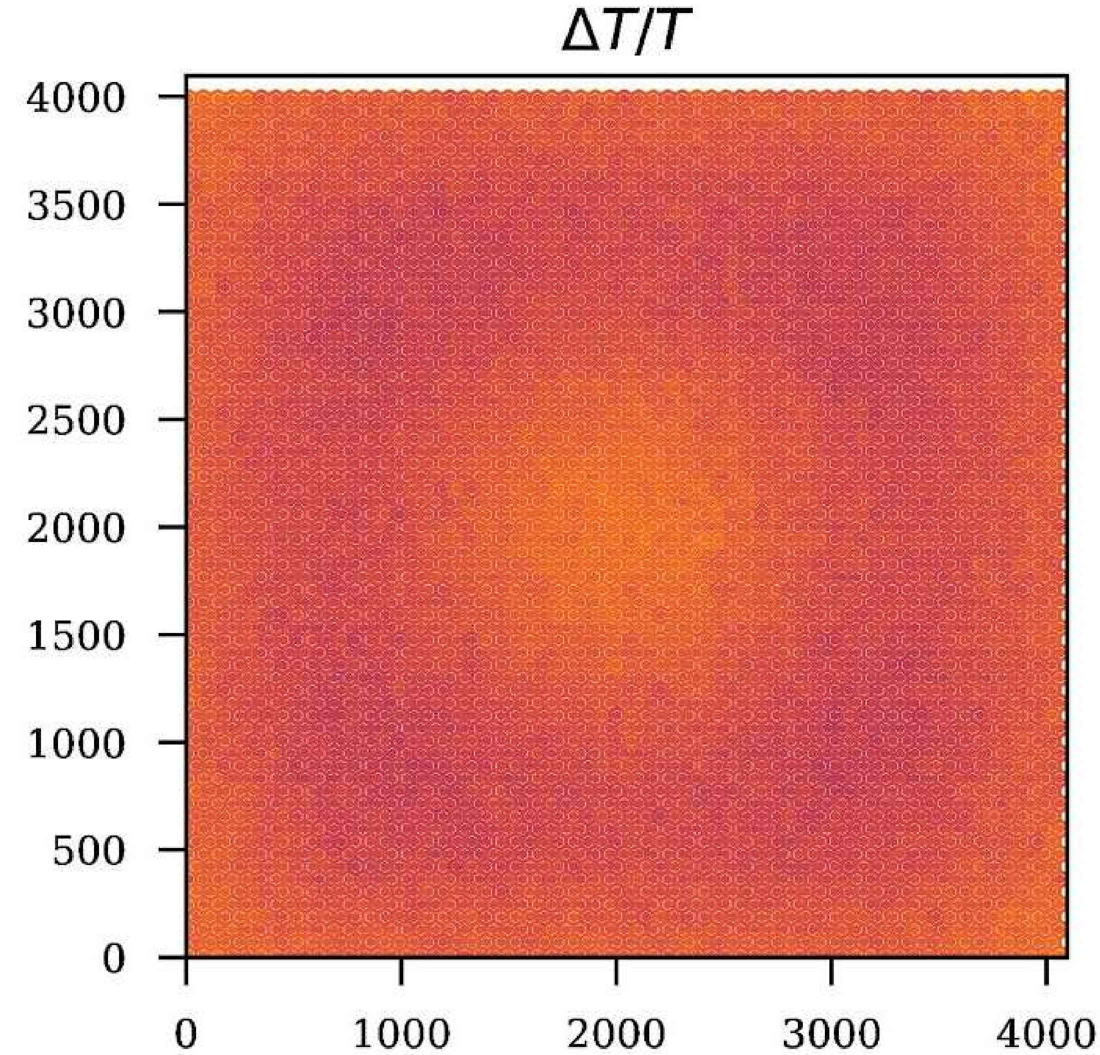
DC2 Simulation

“Weak Lensing” bands (riz)





ΔT/T
2
1
0
−1
−2
−2
−1
0
1
2



ΔT/T
4000
3500
3000
2500
2000
1500
1000
500
0
0
1000
2000
3000
4000

Possibly Improved PSF Model

PixelGrid with BasisPolynomial interpolation is working well, but we think it can be further improved.

We are currently working on a PSF model that includes the following:

- Use simple VonKarman model with Gaussian process interpolation for the atmospheric component
- Use Optical model based on reference wavefront images (from donuts) and only a few free parameters constrained by each in focus exposure.
- Include a small Gaussian component for charge diffusion in chip coordinates.

Gaussian Process interpolation

- Pierre-François Leget has developed a GP interpolation scheme that uses an anisotropic correlation function for the GP kernel.
- This is expected to perform well on the atmospheric component where the underlying physics is close to a real gaussian process
- Interpolation gives the best linear unbiased estimator given a single realization of a Gaussian random field.
- Requires knowledge of the 2-point correlation function for the kernel.
- We measure that directly with TreeCorr using 2D binning option.
- Performs well on simulations that only have an atmospheric component.

Gaussian Process interpolation

Simulated anisotropic Gaussian random field in DECam-like FOV

Piff solution with 2nd order polynomial on each chip

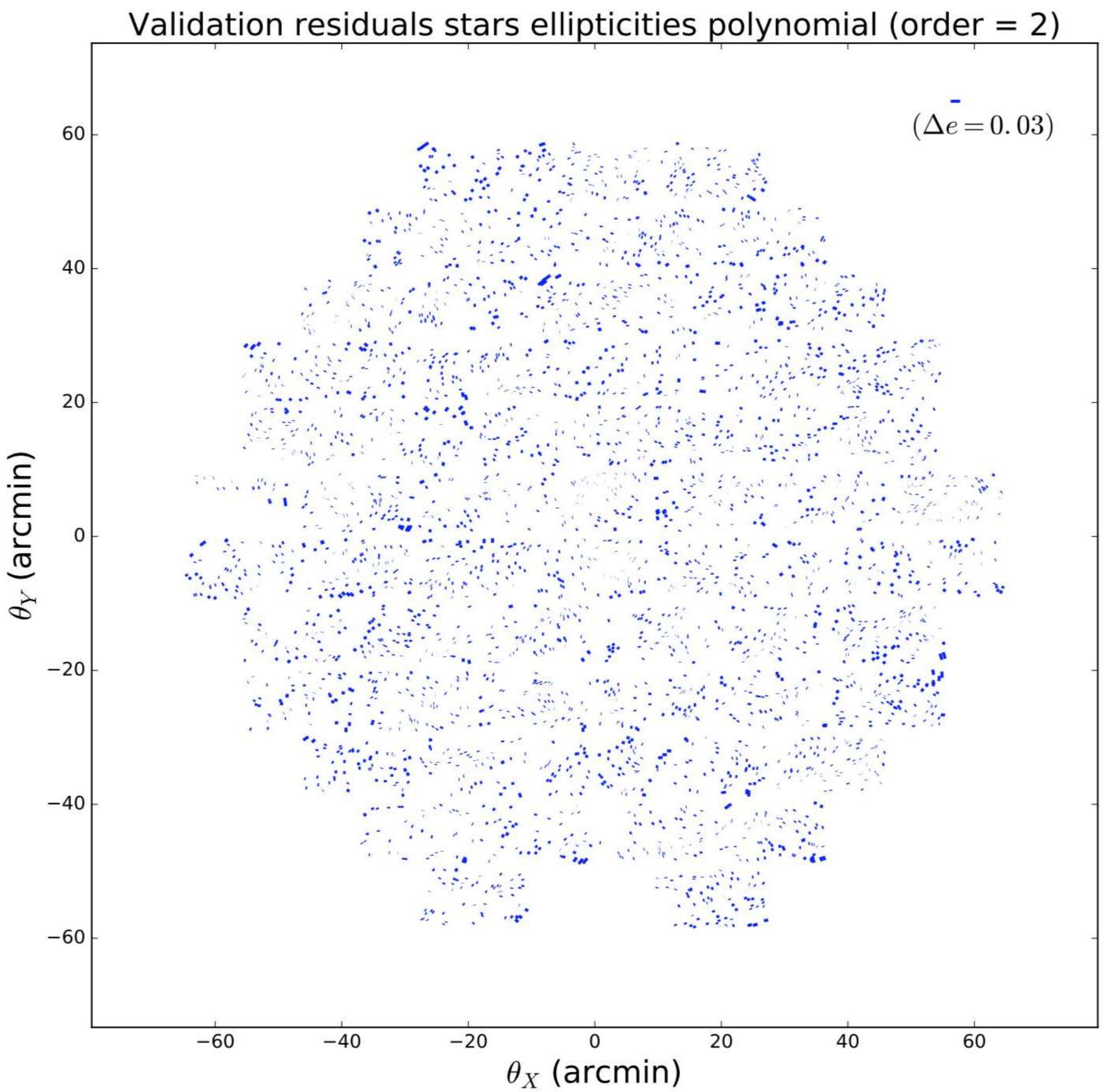


Image credit: P.F. Leget

Gaussian Process interpolation

Simulated Gaussian random field in DECam-like FOV

Piff solution with isotropic GP

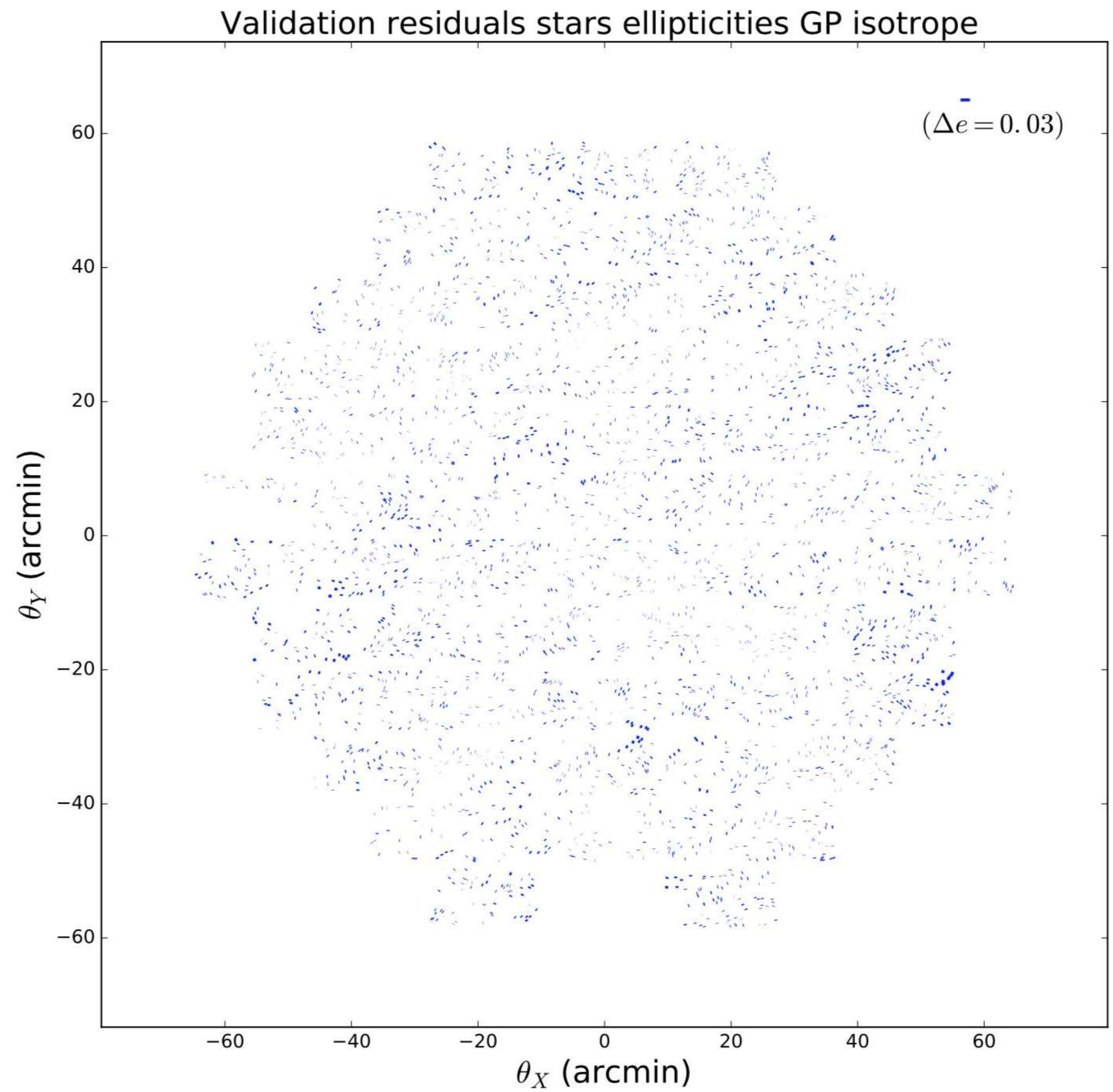


Image credit: P.F. Leget

Gaussian Process interpolation

Simulated Gaussian random field in DECam-like FOV

Piff solution with anisotropic GP

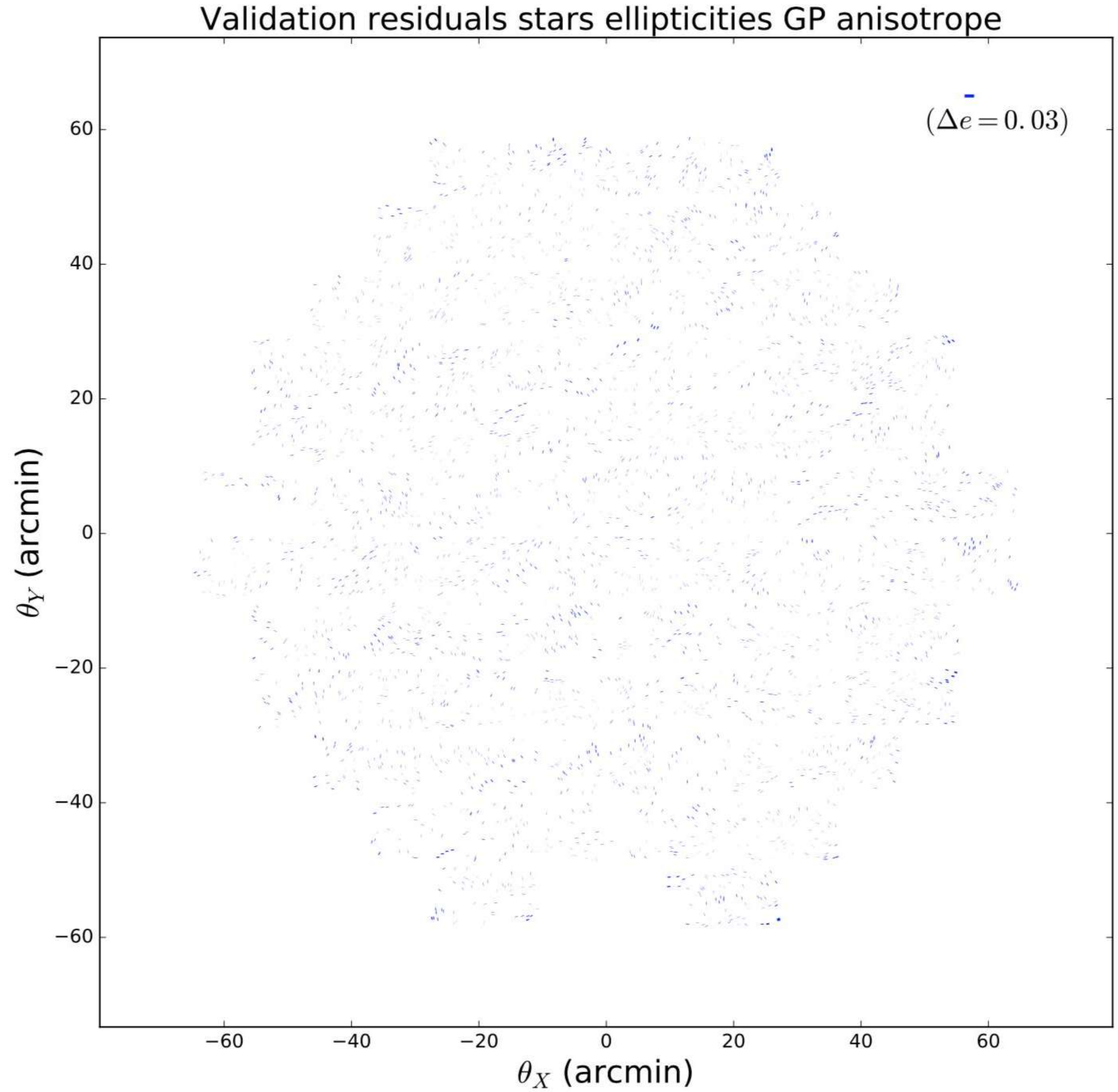


Image credit: P.F. Leget

Optical Model

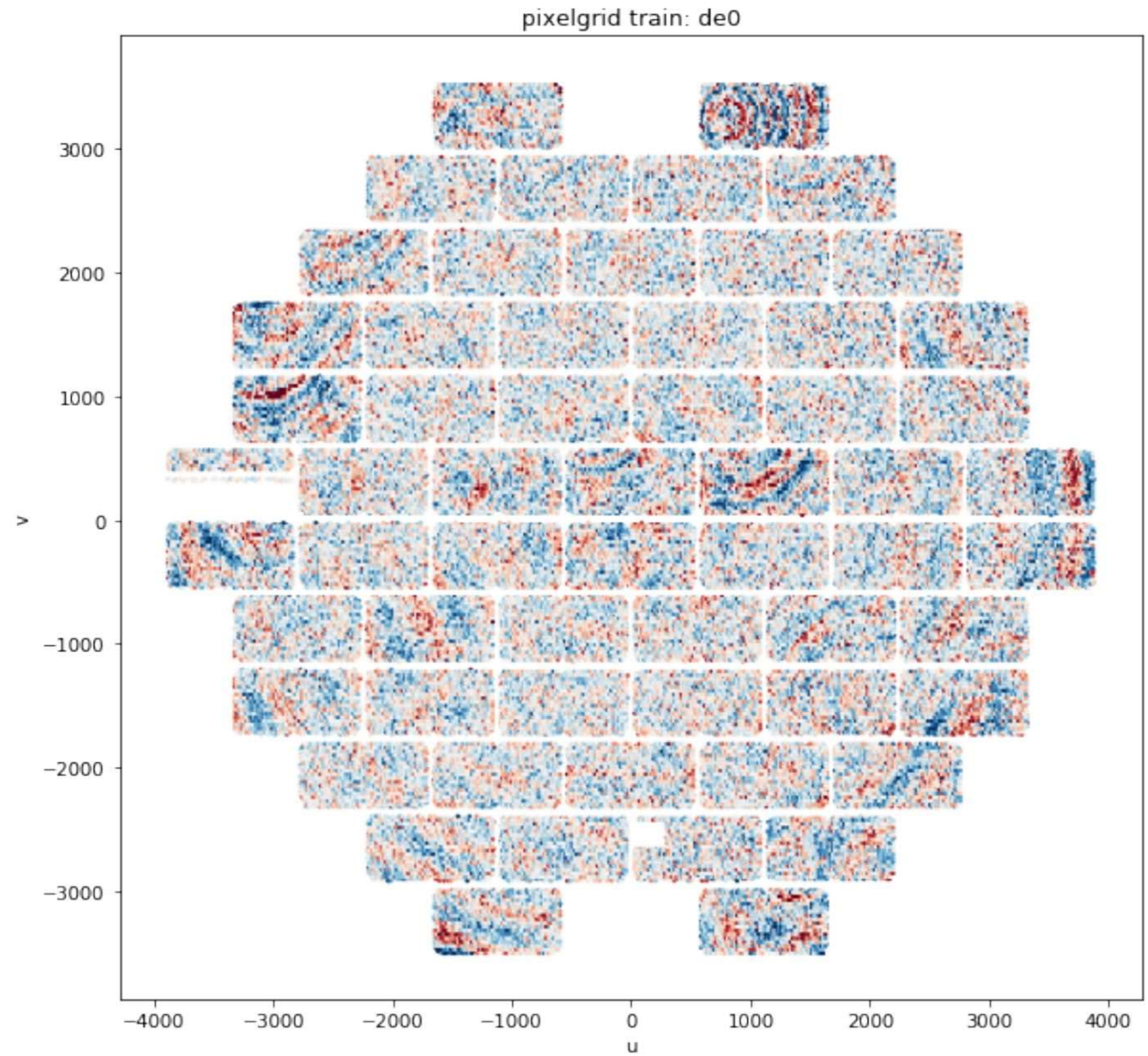
- Josh, Aaron, Ares and Chris have been developing an optical model of DECam and telescope optics
- Most of the model is fixed from out-of-focus images (donuts) to measure Zernike modes across field of view
- In-focus images have a small number of free parameters:
    - Zernike 4-11 (defocus, astigmatism, coma, trefoil, spherical)
- This model is convolved by a Kolmogorov atmospheric component.
- We also need to separately apply a mean residual that is not well captured by this model (cf. tree rings in dT). This step is called “meanify”.

Optical Model

Average residual for
1000 DES Y3 exposures

PixelGrid model
BasisPolynomial interpolation
(3rd order per CCD)

Image credit: C. Davis



Optical Model

Average residual for
1000 DES Y3 exposures

Optical + Atmosphere model

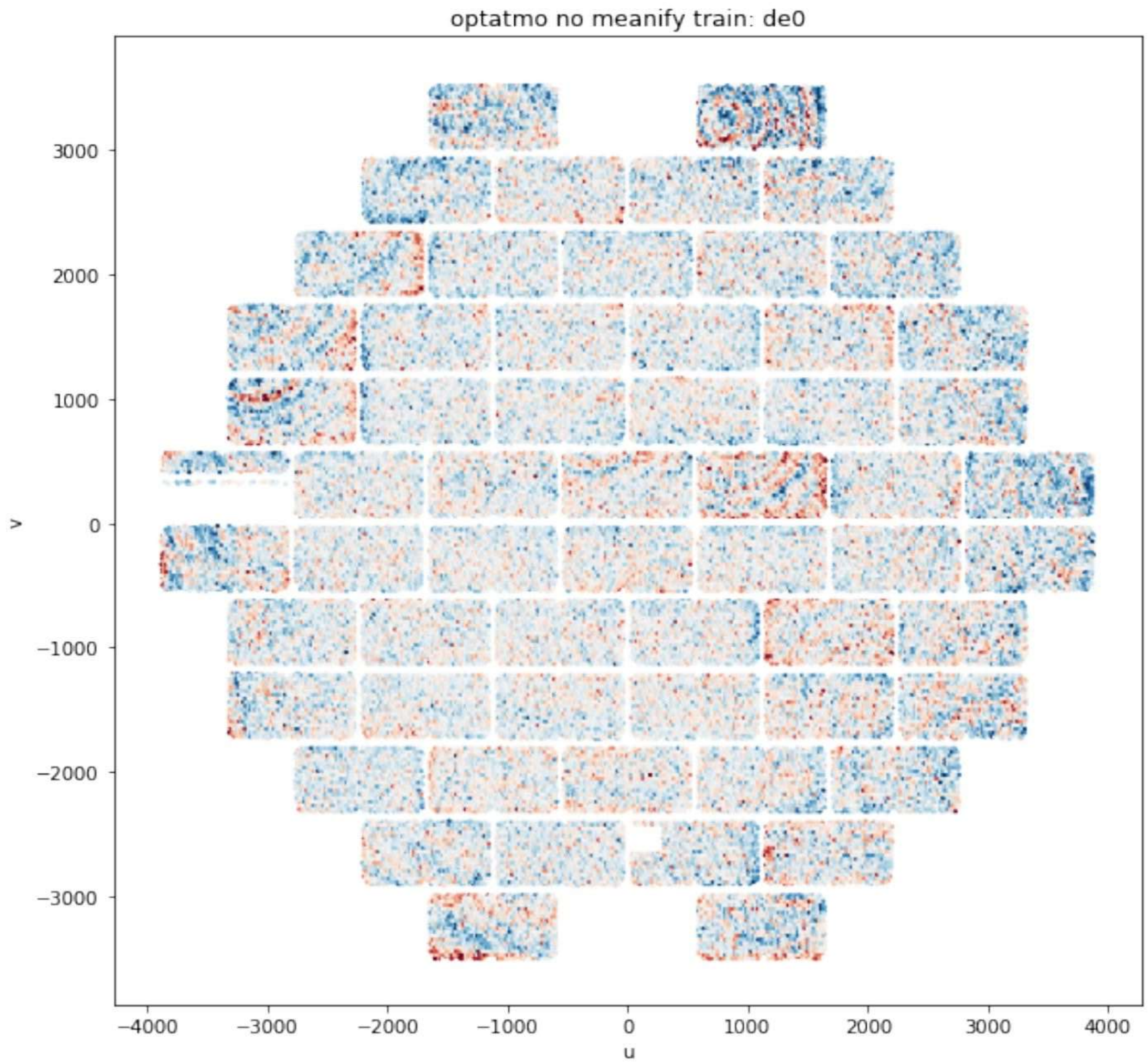


Image credit: C. Davis

Optical Model

Average residual for
1000 DES Y3 exposures

Optical + Atmosphere Model
With “mean” correction

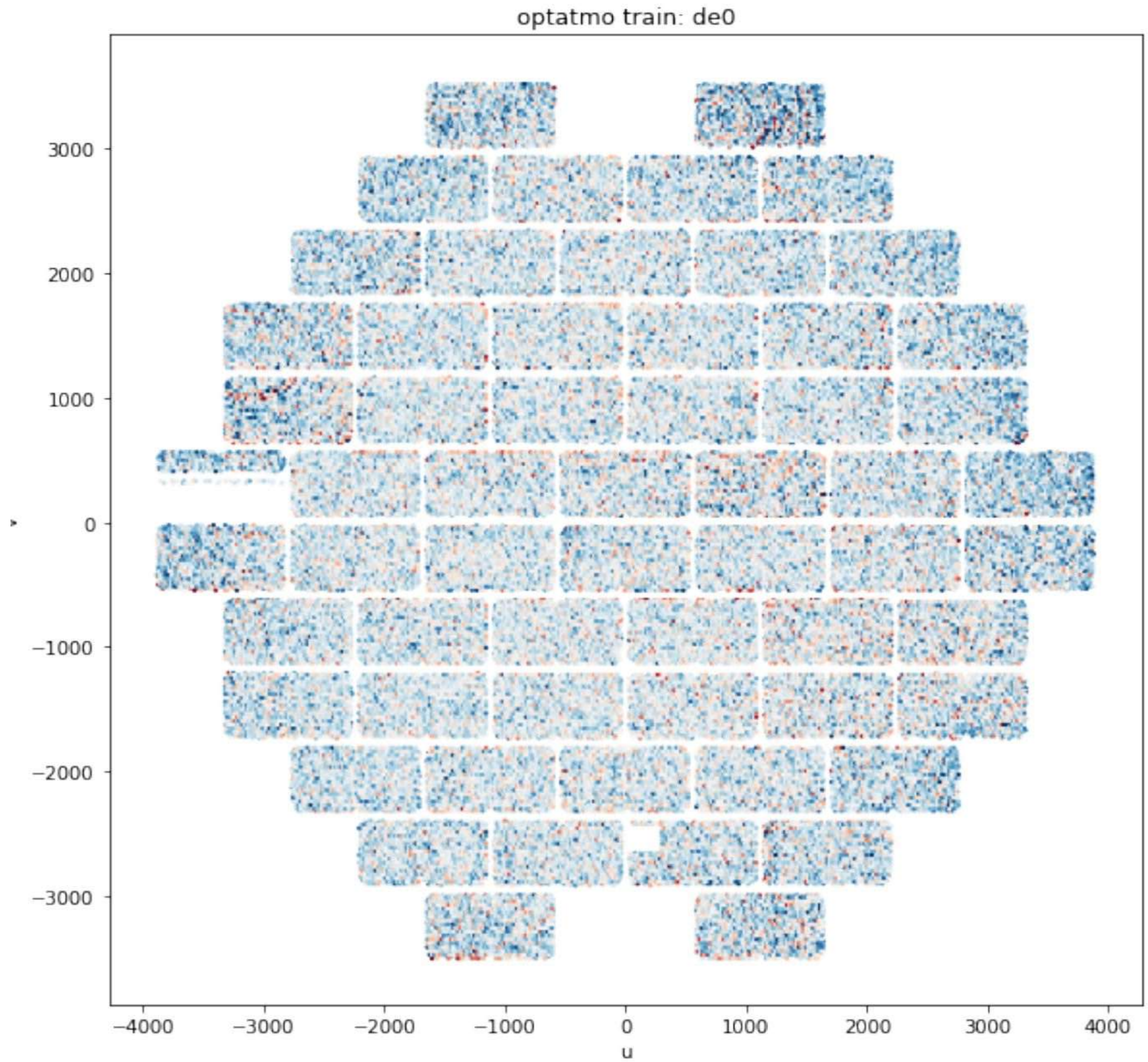


Image credit: C. Davis

Planned Research and Development

- Optical + Atm (GP) model
  - Development to combine these models is actively underway.
  - Will then try it out on DES data
  - Will need further work to develop LSST analog.
- DC2 failures
  - Investigate cause of failures. Seems to be due to singular matrices during solve.
  - Look for algorithmic improvements to help fix.
  - Try different configuration parameters. (Larger model pixel size, 3rd order interp, ...)
- Investigate performance on very good seeing data
  - Seems OK so far for the range of expected LSST seeing.
  - Have not yet tested at best HSC seeing.
- Color dependence
  - Piff has hooks for color dependence of PSF, but we have not yet tested it.
  - Plan to try to include some kind of color dependence in DES Y6 run.

Extra Slides

```
modules:
    - galsim_extra

input:

    dir: /astro/u/mjarvis/work/y3_piff/y3a1-v01/226650
    image_file_name: D00226650_z_c13_r2355p02_immasked.fits
    image_hdu: 0
    badpix_hdu: 1
    weight_hdu: 2

    cat_file_name: D00226650_z_c13_r2355p02_stars.fits
    cat_hdu: 1

    x_col: x
    y_col: y
    sky_col: sky
    ra: TELRA
    dec: TELDEC
    gain: 1.0

    max_snr: 100
    min_snr: 20

    stamp_size: 25

    wcs:
        type: Pixmappy
        dir: /astro/u/mjarvis/rmjarvis/DESWL/psf/
        file_name: zone029.astro
        exp: 226650
        ccdnum: 13

output:
    #dir: /astro/u/mjarvis/work/y3_piff/y3a1-v01/226650
    file_name: D00226650_z_c13_r2355p02_piff.fits

psf:

    model:
        type: PixelGrid
        scale: 0.30
        size: 17
        start_sigma: 0.8  # The code is pretty robust to this being larger than the actual size
                          # of the PSF, but if it's much smaller, there can be problems.

    interp:
        type: BasisPolynomial
        order: 3

    outliers:
        type: Chisq
        nsigma: 4
        max_remove: 0.01

verbose: 2
```

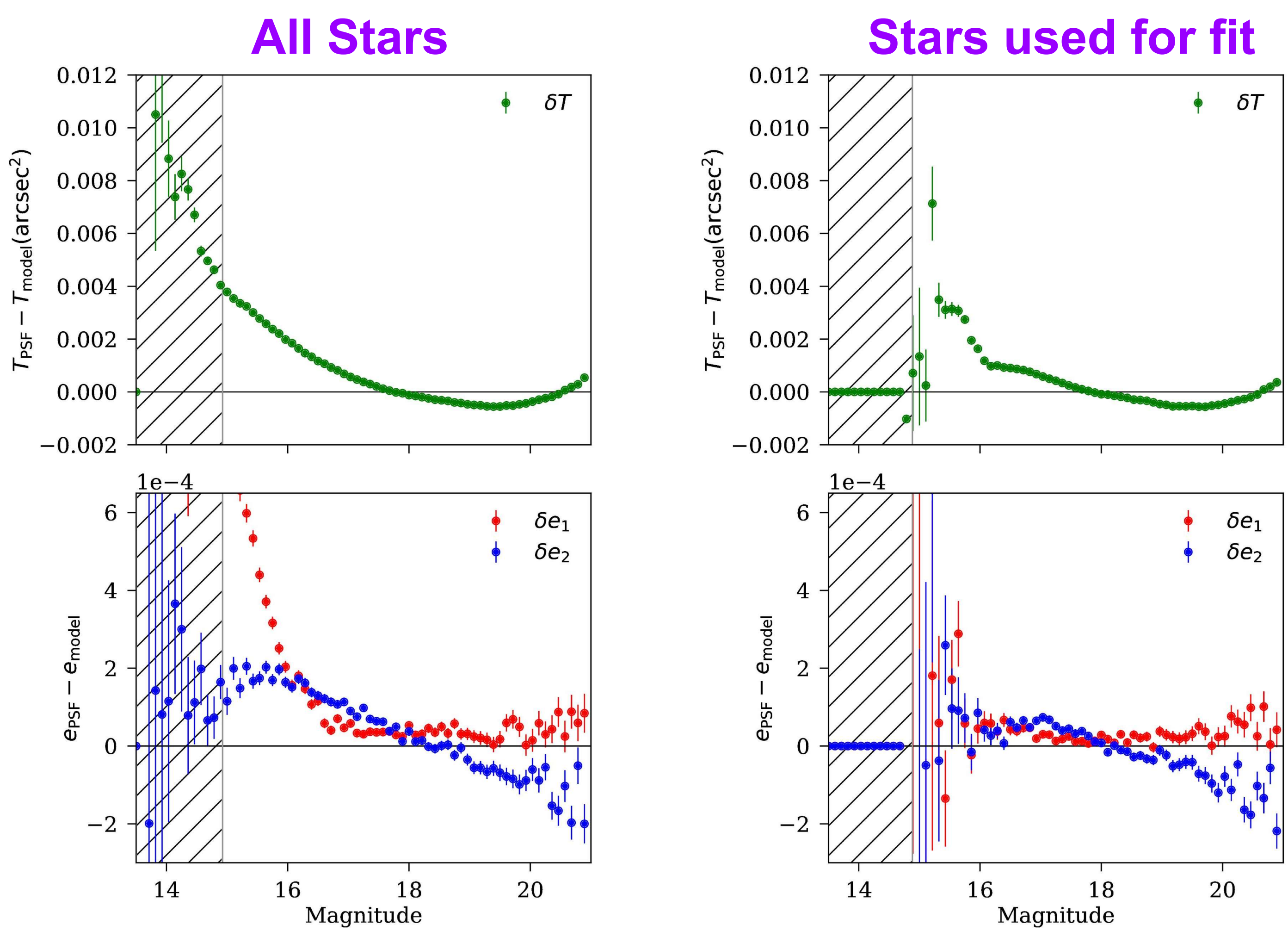
Can modify any parameters you want on the command line (e.g. in a for loop in a script):

```
piffify piff.yaml input.image_file_name=image input.cat_file_name=cat
output.file_name=output input.wcs.file_name=pixmappy_file
input.wcs.exp=exp input.wcs.ccdnum=ccdnum
```

Or can call this from within python

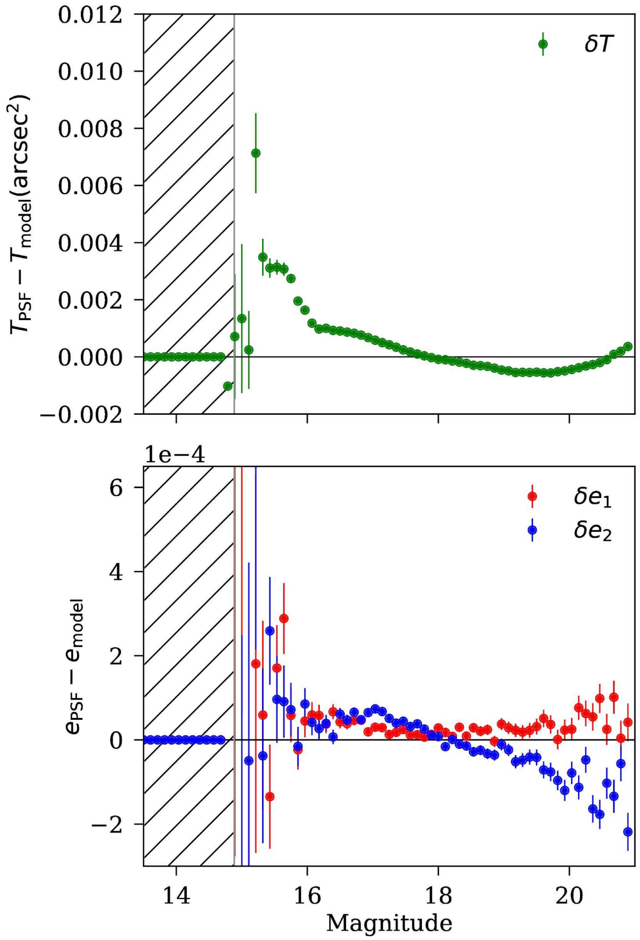
```
config = piff.read_config(piff_config)
config['input']['image_file_name'] = img_file
config['input']['cat_file_name'] = piff_cat_file
config['output']['file_name'] = psf_file
config['input']['wcs']['file_name'] = pixmappy
config['input']['wcs']['exp'] = exp
config['input']['wcs']['ccdnum'] = ccdnum
piff.piffify(config, logger)
```

DES Y3: Mean residuals by magnitude



Compare with PSFEx models

Piff



PSFEx