iPTF/ZTF Image Differencing & Extraction

Frank Masci & the iPTF/ZTF Team

LSST - ZTF joint meeting, November 2014

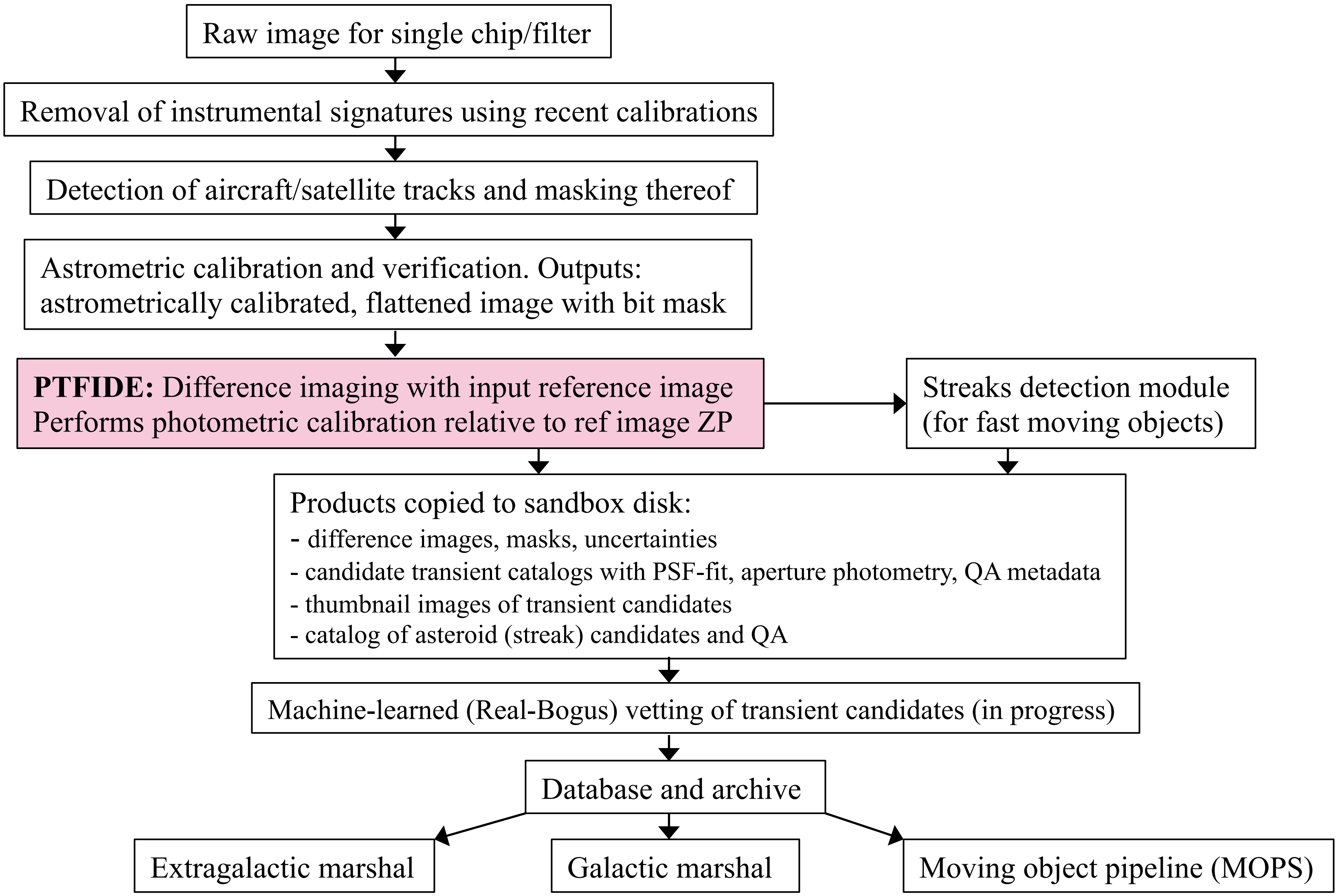
http://web.ipac.caltech.edu/staff/fmasci/home/miscscience/masci_lsst_ztf_Nov2014.pdf

Goals and desiderata

- **PTFIDE:** Image Differencing and Extraction engine for iPTF, ZTF (and the future…)
- **Difference imaging**: discover transients by suppressing everything that's static in space and time
- Given the complexity and heterogeneity of the PTF / iPTF surveys, we wanted a tool that:
  - is **flexible:** robust to instrumental artifacts, adaptable to all seeing conditions, little tuning
  - could operate in a **range of environments:** high source density, complex backgrounds and emission
  - could probe a **large discovery space:** pulsating & eruptive variables, eclipsing binaries, SNe, asteroids
  - **maximizes** the reliability of candidates to streamline/ease vetting process downstream
  - **optimal:** maximizes signal-to-noise of detected candidates
  - is **photometrically accurate:** obtain reasonably accurate "first look" light curves (AC photometry)
  - had **preprocessing steps** customized for the iPTF instrument/detector system
- Existing off-the-shelf methods and tools (as of ~3 years ago) were not flexible or generic enough
- Developed over last two years (on a tight budget) and tuned in response to on-sky performance

The "real-time" operations pipeline at IPAC/Caltech

PTFIDE has been running in the real-time (nightly) pipeline for ~ 18 months. Can also execute offline on archival data.



PTFIDE processing flow

- Single image exposure (*sci*)
- Reference (*ref*) image + catlog
- Parameters

ref/*sci* source matching for throughput (gain) matching of *sci* to *ref* (need < 1% accuracy)

Relative astrometric refinement of *sci* to *ref*: orthogonal shifts only (need <~ 0.2″ accuracy)

Reproject, resample & "warp" reference image onto *sci* image frame; update masks

Match smoothly varying spatially-dep backgrounds

make point-source cutouts / coadds from *sci* & *ref* for matching-kernel derivation

Derive PSF-matching convolution kernels to match seeing/resolutions (spatially dependent)

Apply PSF-matching kernels and compute two image differences: *sci* – *ref* & *ref* – *sci* with masks and uncerts

Quality assurance metrics for difference images; good/bad decision before proceeding

Estimate spatially varying PSF, extract candidates using PSF-fitting & aperture photom;
→ catalog + metadata for both difference images

Loose filtering of transients using PSF-fitting metrics

Compute source metrics and "features" to support machine-learned vetting

OLD white paper: *http://web.ipac.caltech.edu/staff/fmasci/home/miscscience/ptfide-v4.0.pdf*

Reference Image Creation

- Outlier-trimmed averages of stacks of the "best quality" science exposures in terms of seeing (FWHM), limiting depth, and astrometric accuracy
- Best seeing images used because goal (at first) is to always convolve reference image prior to differencing with science image (more later)
- Typically require at least 8 "good" science exposures (satisfying all criteria) for a given field/chip
- Input image pixels are weighted according to 1/(image seeing FWHM)
- Throughput (gain) matching of input science exposures to a common global photometric zero-point
- Relative refinement of astrometry (and distortion) solutions between input images
- Pixels are "de-warped" and interpolated using Lanczos kernel of order 3:

$$L(x, y) = sinc(x)\ sinc(x/3) sinc(y)\ sinc(y/3), \qquad -3 < \{x, y\} < 3$$

  - optimal for PSFs that are >~ critically sampled below some high-*v* since has *sinc*-like properties
  - compact "support" minimizes spreading of bad/saturated pixels and aliasing
  - uncorrelated input noise remains closely uncorrelated
- Sources are extracted using both aperture and PSF-fit photometry
- Reference images and catalogs are archived and registered in a DB for fast retrieval
- (Re)create manually if an existing reference image is bad or not available for a new field location

PTFIDE: reference image to science frame reprojection

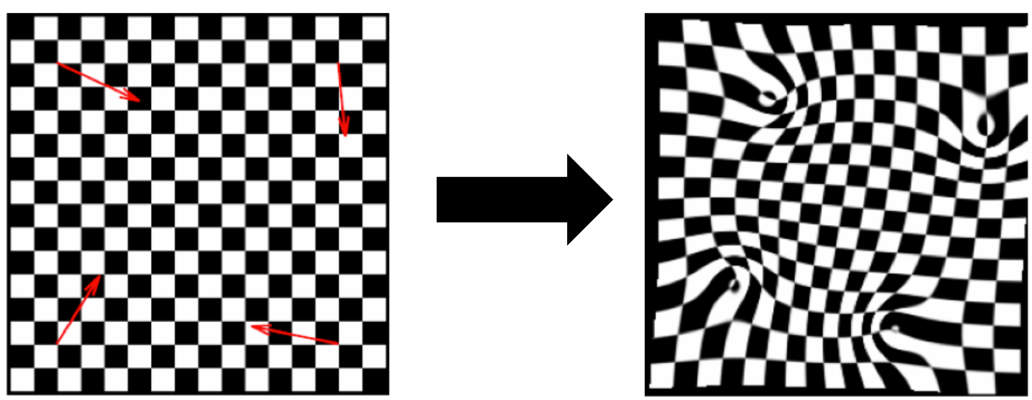
- Reference image is “warped” onto science image grid using science image distortion polynomial coefficients, calibrated upstream as part of astrometric calibration
- Distortion coefficients are calibrated per image and follow the **non-standard** PV convention, e.g:

```
PV1_0   =                   0. / Projection distortion parameter
PV1_1   =                   1. / Projection distortion parameter
PV1_2   =                   0. / Projection distortion parameter
PV1_4   =  0.00135794022943969 / Projection distortion parameter
PV1_5   = 0.000497809862082518 / Projection distortion parameter
```

  etc..

- Reason: used by *Astromatic* software suite (*SCAMP*, *SWarp*...)
- Also represented in SIP (Simple Image Polynomial) format in FITS headers
- Interpolation of input reference image pixels onto science grid uses Lanczos kernel of order 3
- Astrometric/distortion calibration of science image is crucial. If wrong, astrometry of reprojected reference image will also be wrong and residuals will result in difference image (more later)

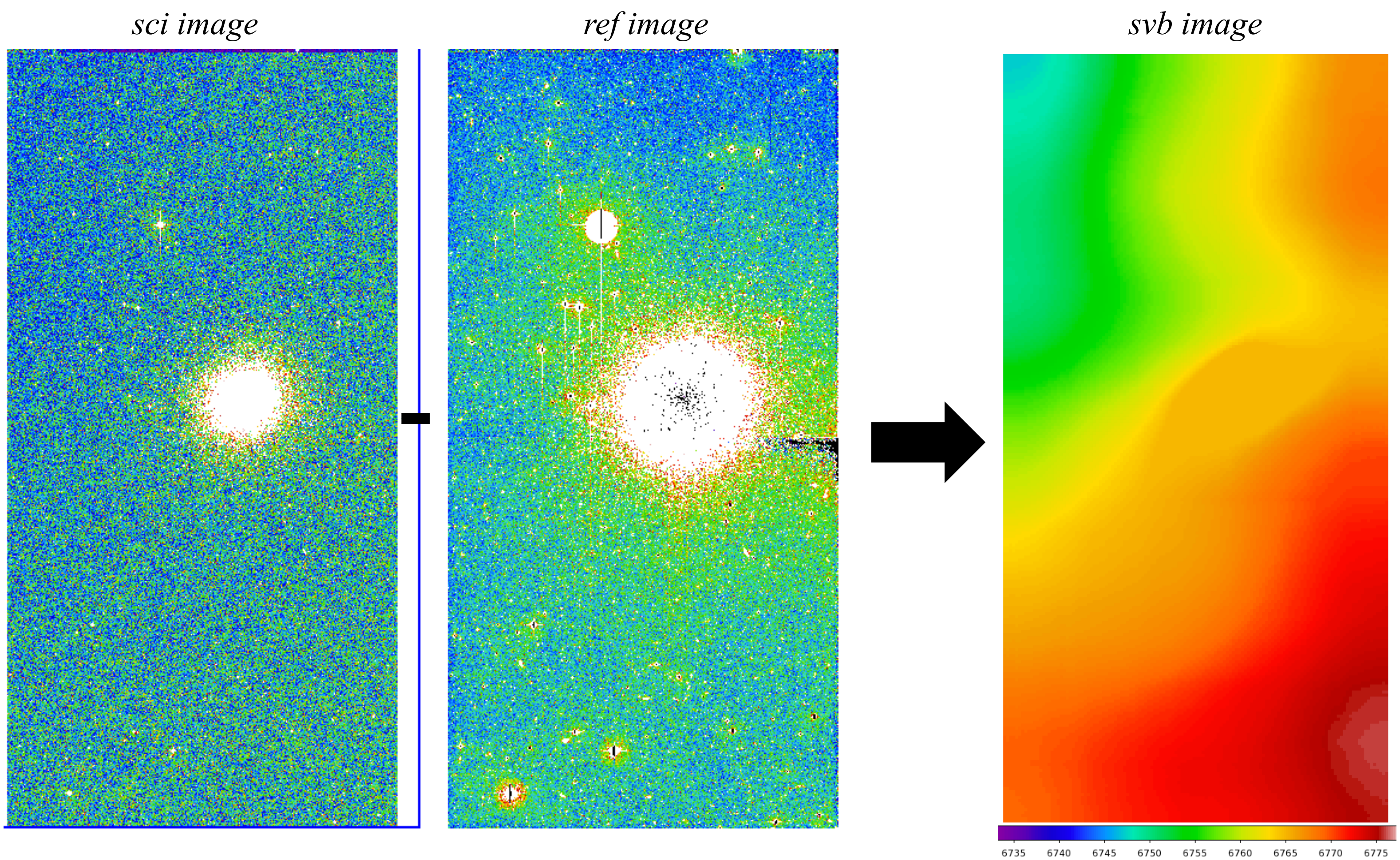
ref image grid



sci image grid

PTFIDE: differential spatially-dependent background matching

- Compute low-pass filtered, smoothly-varying differential background (SVB) and correct science image to match reference image: $sci_{new} = sci_{old} - \langle sci_{old} - ref_{resampled} \rangle_{filt}$
- Matched backgrounds => helps improve photometric accuracy on difference images later

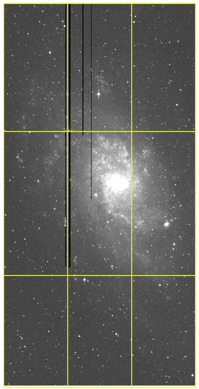


Prepare inputs for PSF-matching

- In general, an observed image I (science exposure) can be modeled from a (higher S/N, better "seeing") reference image R, a PSF-matching convolution kernel K, differential background dB, and noise:

$$I_{ij} = \left[K(u,v) \otimes R_{ij} \right] + dB + \varepsilon_{ij}$$

- Before we derive K (later), need *accurate* representations of PSF shapes from the science and reference images as a function of position on the focal plane
- Estimation of convolution kernel K is sensitive to noise in input images, hence need to mitigate noise
- Generate PSF-representations with high S/N by stacking (co-adding) point-source cutouts from *sci* and *ref* images
- To model spatial variations, generate PSF co-adds over a N x N grid
  - where $N = 3$ for now => nine 11.5' x 23' partitions per chip, with some overlap
- Typically require a minimum of 20 "clean" (filtered) point sources per partition
- Enforce a maximum of N_{max} =150 point sources (for run-time reasons!). This still gives us reasonable S/N.
- If number of sources > N_{max}, use brightest N_{max} point sources available
- Initial (naïve) method used entire image partitions from *sci* and *ref* images as inputs for estimating K
  - solution was severely affected by large number of pixels containing just noise (no signal)
  - obtain more optimal solutions if isolate point-sources and build S/N therefrom



single chip (~ 0.57° x 1.15°) with **M33**

Prepare inputs for PSF-matching (detailed processing flow)

Filter ***ref*-image** extractions to retain N "clean" point sources for each pre-defined image partition; need $N_{min} <= N < N_{max}$

↓

Cut out image stamps from *sci* and *ref* images using (more accurate) *ref*-image point-source positions only. Want to preserve any relative shift that can be modeled by kernel solution later

↓

Reinterpolate stamp cut-outs using Lanczos3 kernel to correct for integer-pixel rounding so can ~ reconstruct input source centroids

↓

Background-subtract stamps and flux-normalize using *ref*-image fluxes only. Want to preserve any relative gain that can be modeled by kernel solution later (= kernel sum)

↓

Background-subtract stamps and flux-normalize using *ref*-image fluxes only. Want to preserve any relative gain that can be modeled by kernel solution later (= kernel sum)

↓

For each *sci* and *ref* image partition, stack (co-add) point source stamps using outlier-trimmed, inverse-variance weighted averaging

↓

Further regularize PSF image co-adds and replace any spatial outliers using winsorization. Careful not to remove real PSF signal!

→

QA metrics on PSF products: RMS and pixel-sum ratios (= gain ratios)

↓

Check pixel-sum ratios of PSF products (= residual gains) across partitions and correct PSFs of partitions with large residual gains

↓

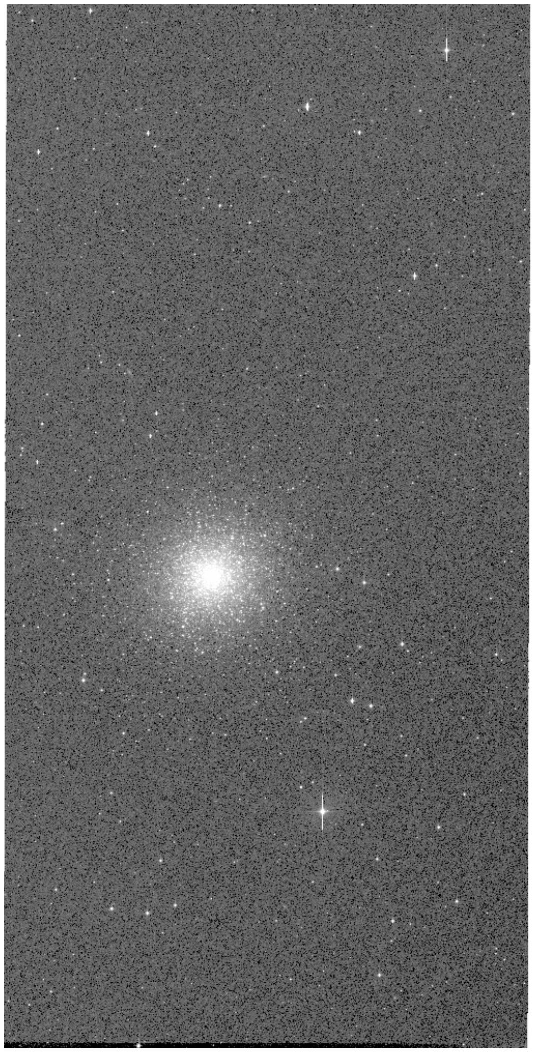
If partition had < N_{min} sources or PSFs have RMS > *threshold*, assign PSFs from partition with min RMS

↓

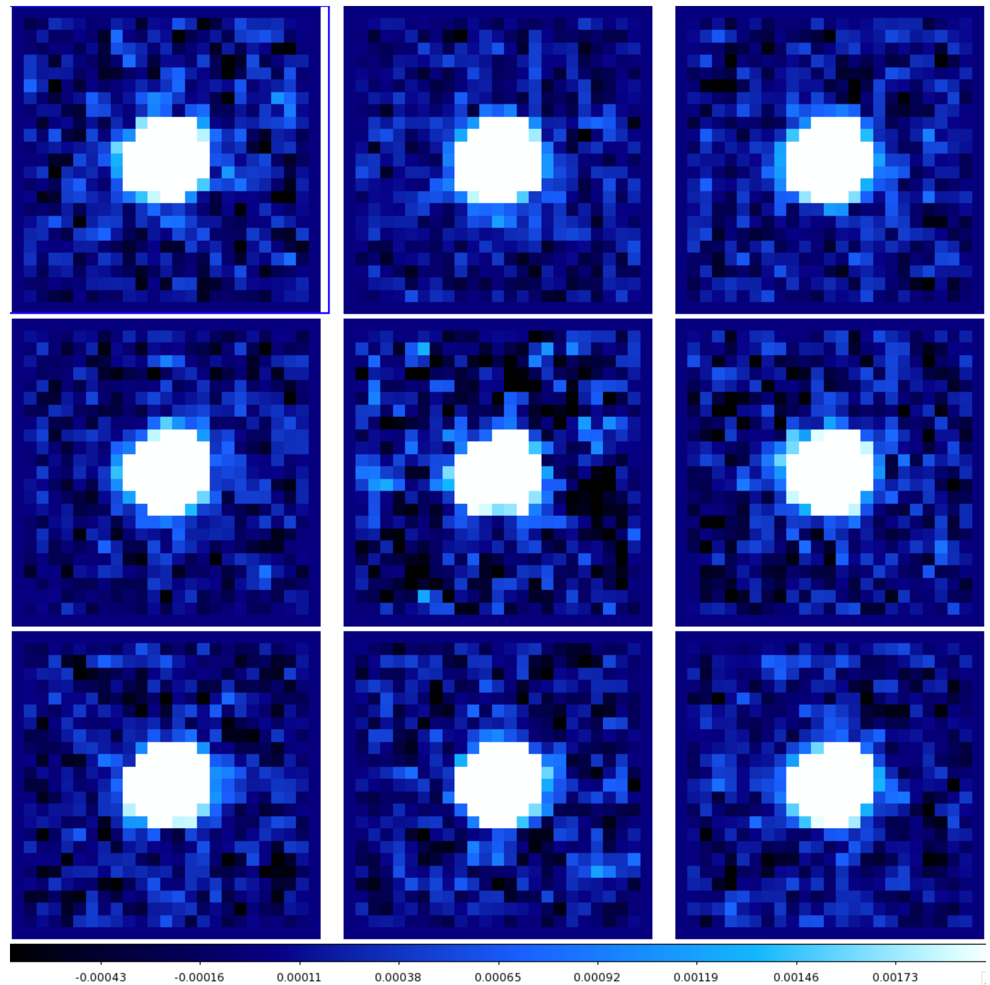
Two high S/N PSF-images per *sci* and *ref* image partition for deriving PSF-matching convolution kernel

Example input PSF-images for deriving PSF-matching kernel

***science** image exposure with M13*

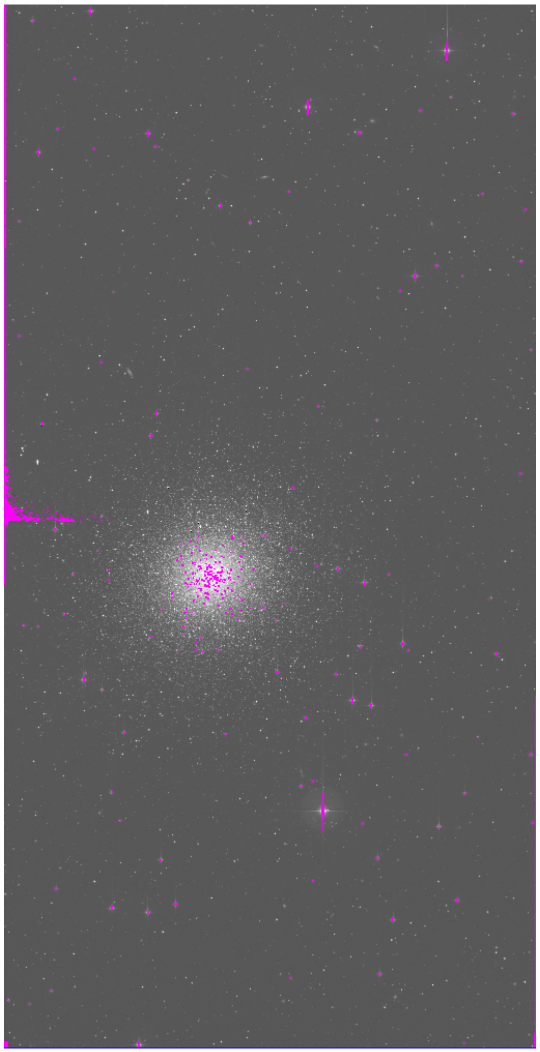


*PSF (co-add) products over **sci-image** partitions*

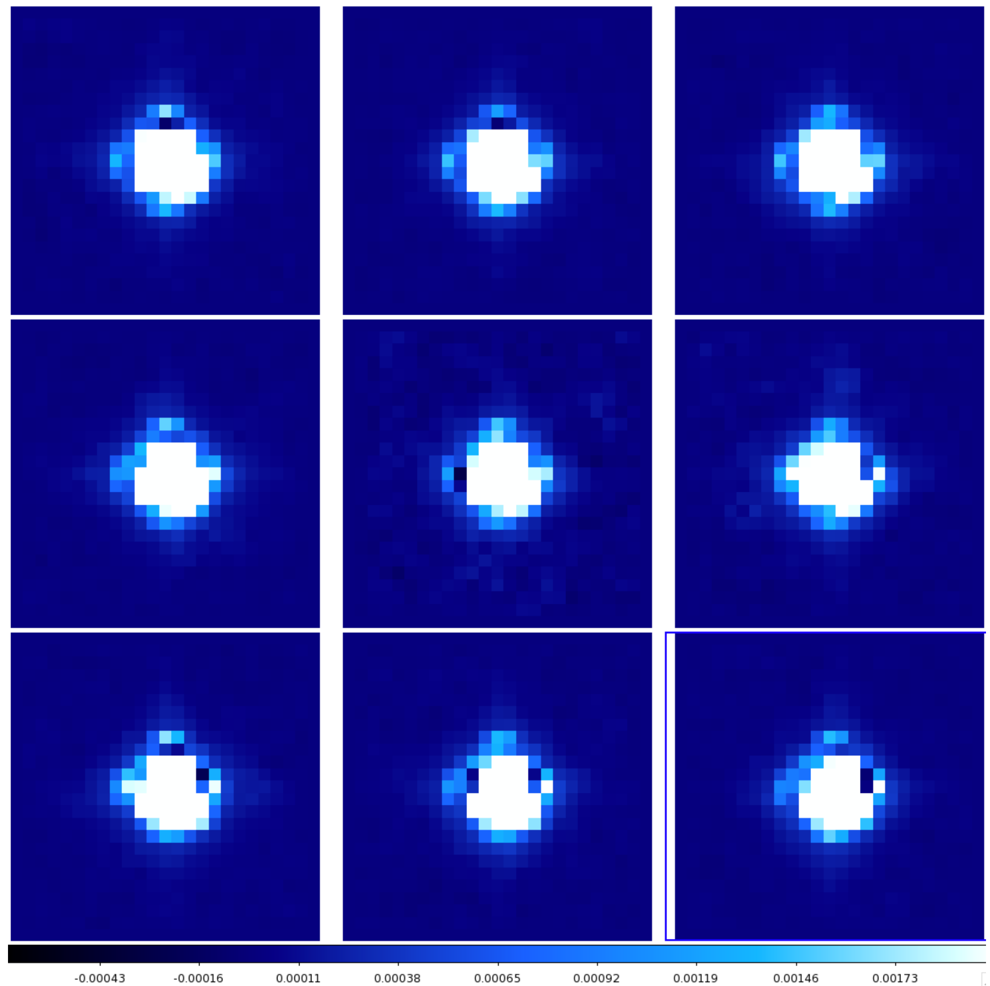


Example input PSF-images for deriving PSF-matching kernel

reference *image with M13*



PSF (co-add) products over ***ref-image*** *partitions*



Derivation of PSF-matching kernel

- Recall, we can model observed image I (science exposure) from a higher S/N, better "seeing" reference image R, PSF-matching convolution kernel K, differential background dB, and noise term:

$$I_{ij} = \left[K(u,v) \otimes R_{ij}\right] + dB + \varepsilon_{ij}$$

- PSF-matching entails finding an optimum convolution kernel K by minimizing some cost function, e.g., chi-square:

$$\chi^2 = \sum_{i,j}\left[\frac{I_{ij} - \left[K(u,v) \otimes R_{ij}\right] - dB}{\sigma_{ij}}\right]^2 \equiv \left(I - M\right)^T \Omega_{\text{cov}}^{-1}\left(I - M\right)$$

where M is the "model" image:

$$M_{ij} = \left[K(u,v) \otimes R_{ij}\right] + dB$$

- Customary to represent K as a linear combination of n basis functions K_i with coefficients a_i:

$$K(u,v) = \sum_i^n a_i K_i(u,v)$$

- n parameters of K can be solved using standard linear-least squares via $\partial\chi^2/\partial a_i = 0$ and inverting the matrix system

Initial derivation of PSF-matching kernel

Traditional method (until about 2007 and still popular today):

- Decompose K into a sum of Gaussian basis functions modified by shape-morphing polynomials (e.g., Alard & Lupton, 1998; Alard 2000). Coefficients are then estimated. Implemented in *HOTPANTS* and *DIAPL*.

$$
\begin{aligned}
K_{a_i}(u,v) = &\sum_{p1,q1} a_{p1q1} u^{p1} v^{q1} \exp\left[\frac{-\left(u^2+v^2\right)}{2\sigma_1^2}\right] \\
&+ \sum_{p2,q2} a_{p2q2} u^{p2} v^{q2} \exp\left[\frac{-\left(u^2+v^2\right)}{2\sigma_2^2}\right] \\
&+ \sum_{p3,q3} a_{p3q3} u^{p3} v^{q3} \exp\left[\frac{-\left(u^2+v^2\right)}{2\sigma_3^2}\right], \\
&+ \sum_{p4,q4} a_{p4q4} u^{p4} v^{q4} \exp\left[\frac{-\left(u^2+v^2\right)}{2\sigma_4^2}\right],
\end{aligned}
$$

$$
a_{p_n q_n}(x,y) = \sum_{r_n s_n} a_{r_n s_n} x^{r_n} y^{s_n},
$$

where

$n = 1,\ 2,\ 3,\ 4$ and

$0 \le r_n + s_n \le 2.$

- For PTF images, found that the following polynomial orders and Gaussian widths worked for some fraction of data:

$$
\begin{aligned}
&0 \le p_1 + q_1 \le 4 && \sigma_1 = 0.40625 \text{ pixels} \\
&0 \le p_2 + q_2 \le 4 && \sigma_2 = 0.65 \text{ pixels} \\
&0 \le p_3 + q_3 \le 2 && \sigma_3 = 1.04 \text{ pixels} \\
&0 \le p_4 + q_4 \le 2 && \sigma_4 = 1.664 \text{ pixels.}
\end{aligned}
$$

- Total number of coefficients in fit (free parameters) was 252. Certainly had enough stars (sufficient #D.O.F.)
- Experimented with this method at first, but found parameterization was not “expressive” or general enough
- Difficult to tune for an entire survey and execute lights out with no intervention

Derivation of PSF-matching kernel in PTFIDE

- Method in PTFIDE discretizes the kernel $K(u,v)$ into L x M pixels and then estimates the pixel values therein, K_{lm}, directly. Provides a "free form" basis expressed as a 2D array of delta functions:

$$K(u,v) = K_{lm}\delta(u-l)\delta(v-m)$$

- Model image in χ^2 cost function on pg. 12 can be written:

$$M_{ij} = dB + \sum_l \sum_m K_{lm} R_{(i+l)(j+m)}$$

- The "best" or optimal values of K_{lm} and dB are those that minimize χ^2, i.e.,

$$\left.\frac{\partial \chi^2}{\partial K_{lm}}\right|_{lo,mo,dBo} = 0: \qquad K_p \sum_{i,j} \frac{R_{(i+lo)(j+mo)} R_{(i+l)(j+m)}}{\sigma_{ij}} + dB_o \sum_{i,j} \frac{R_{(i+lo)(j+mo)}}{\sigma_{ij}} - \sum_{i,j} \frac{I_{ij} R_{(i+lo)(j+mo)}}{\sigma_{ij}} = 0$$

$$\left.\frac{\partial \chi^2}{\partial dB}\right|_{lo,mo,dBo} = 0: \qquad \left(\sum_p K_p\right) \sum_{i,j} \frac{R_{(i+lo)(j+mo)}}{\sigma_{ij}} + dB_o - \sum_{i,j} \frac{I_{ij}}{\sigma_{ij}} = 0$$

$p = 1,2,3 \ldots LM$ = row index of matrix system for corresponding l_o, m_o pair:

$l_o = -(L-1)/2 \ldots (L-1)/2;$

$m_o = -(M-1)/2 \ldots (M-1)/2$

- Leads to a simultaneous system of $LM + 1$ equations in $LM + 1$ unknowns; can be written in vector/matrix form:

$$AX = B$$

- Vector X contains the LM kernel pixel unknowns K_p and differential background estimate dB_o

Derivation of PSF-matching kernel in PTFIDE

- Delta-function-basis is more flexible; *K* can take on more general (unconstrained) shapes
- Can compensate for bad *local* astrometry
  - but only constant (or *slowly varying*) shifts within an image partition

- Also branded as the "**PiCK**" method: **Pixelated Convolution Kernel** method
- **Not new:** similar to method proposed by Bramich (2008); also explored by Becker et al. (2012)
- Only parameters to tune are size of *K* (*L* x *M* pixels) and thresholds for selecting point sources to create PSFs
- *sci* – *ref* difference image for a partition is given by:

$$D_{ij} = I_{ij} - dB_o - \left[K_{lm} \otimes R_{ij} \right]$$

- A measure of the relative gain (residual) between *sci* and *ref* images is given by

$$K_{sum} = \sum_l \sum_m K_{lm} = \sum_p K_p$$

- Can use this as a diagnostic to validate (or refine local) photometric zero-point calibration

PSF-matching kernel: SVD analysis and regularization

- A challenge with the PiCK method is that least-squares solution to K can be dominated by noise if input science and reference image pixels are noisy, even slightly so.
- Biggest limitation is building enough S/N for every pixel in PSF-image inputs => need sufficient number of point sources per partition (typically >~ 50).

- Effective number of degrees of freedom: #PSF-image pixels – (#kernel pixels + 1) = 25x25 – (9x9 + 1) = 543
  - Size of K selected to be small enough to avoid over-fitting, but large enough to avoid biased solutions across expected range of seeing (the so called "bias versus variance" tradeoff)

- One can solve for the kernel unknowns in X using a naïve inversion of the matrix system $A.X = B$:

$$X = A^{-1}B$$

- However, as mentioned, solution could be dominated by noise, especially when A is close to singular. As a check, we use singular value decomposition (SVD) to solve the matrix system and help with possible regularization:

$$A = VWV^T$$

where V is an orthogonal matrix and W is diagonal, containing the eigenvalues, w_i, of A

PSF-matching kernel:
SVD analysis and regularization

- Since A is a real symmetric matrix, SVD is equivalent to an eigenvector (spectral) decomposition and allows us to examine the basis vectors contributing to the kernel solution. Noisy (high frequency) components V_i in V can then be truncated (or reset to zero) to compute a better-conditioned pseudo-inverse matrix:

$$A^{-1} = V\left[diag\left(1/w_i\right)\right]V^T$$

- Leads to "smoother" kernel solutions with a small change in overall χ^2 (or a tiny, but affordable increase at worst)

- Equivalently, solution vector containing kernel values K_{lm} (and differential background offset dB) can be written:

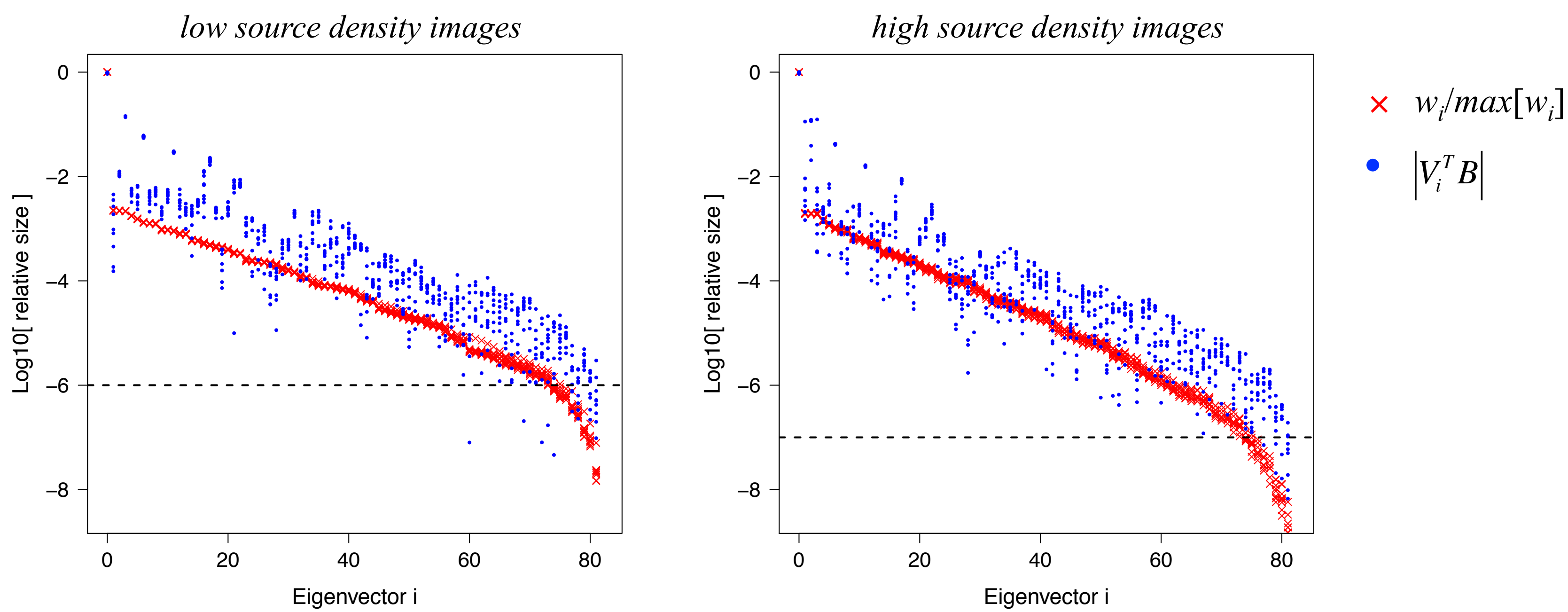
$$X = \sum_{i=1}^{N}\left(\frac{1}{w_i}V_i^T B\right)V_i, \qquad \sigma^2(X) = \sum_{i=1}^{N}\left(\frac{V_i}{w_i}\right)^2$$

- Noisy basis vectors can be identified by examining the N eigenvalues w_i of matrix A (smaller => relatively noisier) or absolute magnitude of the (dot-product) coefficients $\left|V_i^T B\right|$

- For some k where $w_k/max[w_k] < T$, reset $1/w_i$ to 0 for all $i > k$ in expansion above to obtain regularized solution

PSF-matching kernel:
SVD analysis and regularization

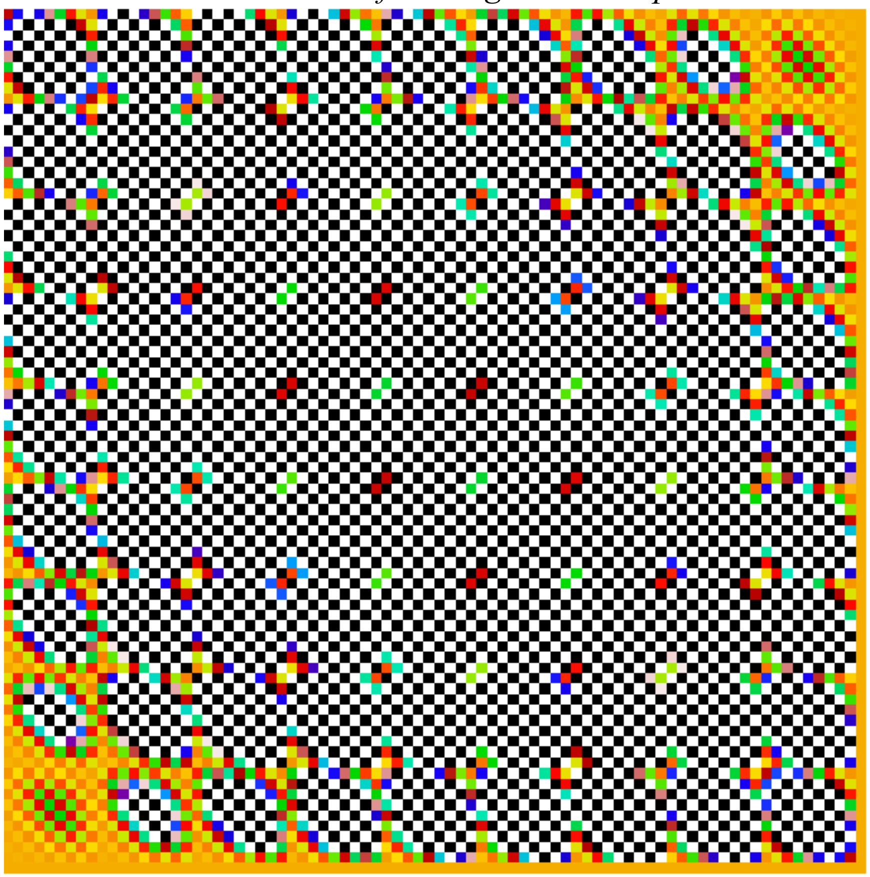
- Relative threshold T for clipping eigenvectors was tuned using difference images across different environments. Conservatively set to not throw away legitimate high frequency information and keep $\Delta\chi^2$ small.
- Or formally $\Delta\chi^2 \;<\sim \sqrt{2d.o.f}$)
- The following quasi-dynamic thresholding works well: $T = min\{10^{-6}, 10^{th}$ percentile in $w_k/max[w_k]\}$



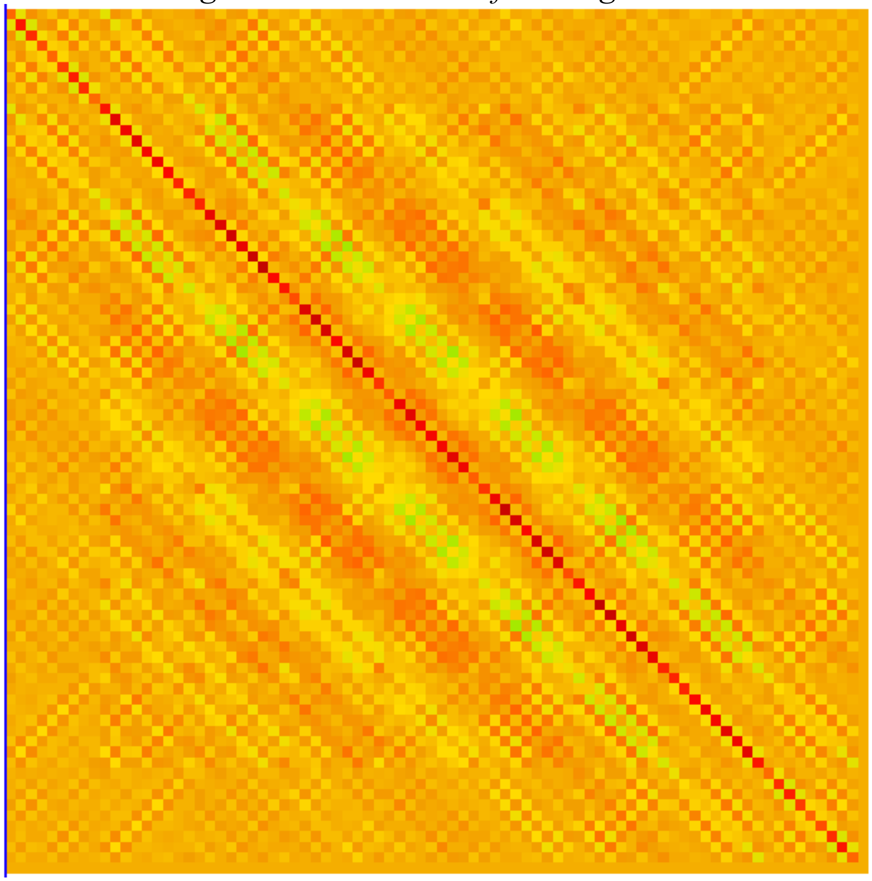
Naïve and pseudoinverse matrices A^{-1} to solve $A.X = B$

- For a single image partition (bottom left corner) in M13 test images on slides 10 and 11
- Regularized version using SVD (with noisiest eigenvectors removed) => better conditioned!

Direct inversion of A using LU decomposition

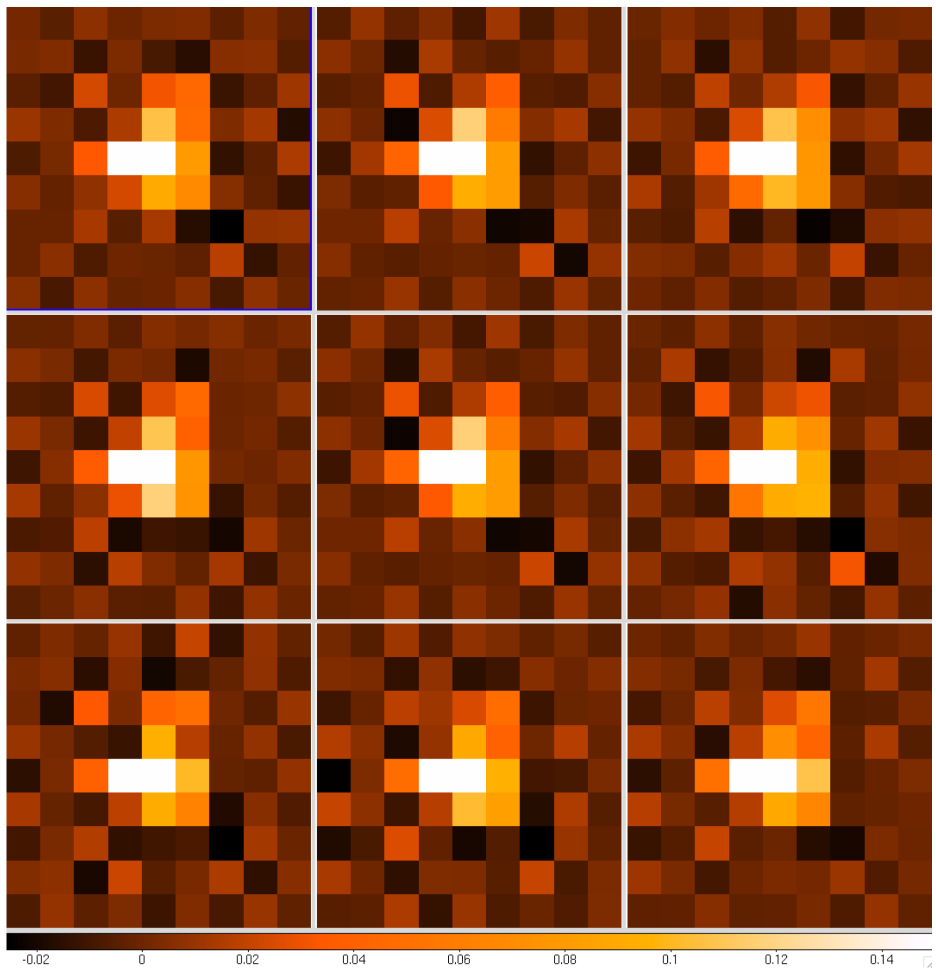


Regularized inversion of A using SVD



Putting it all together

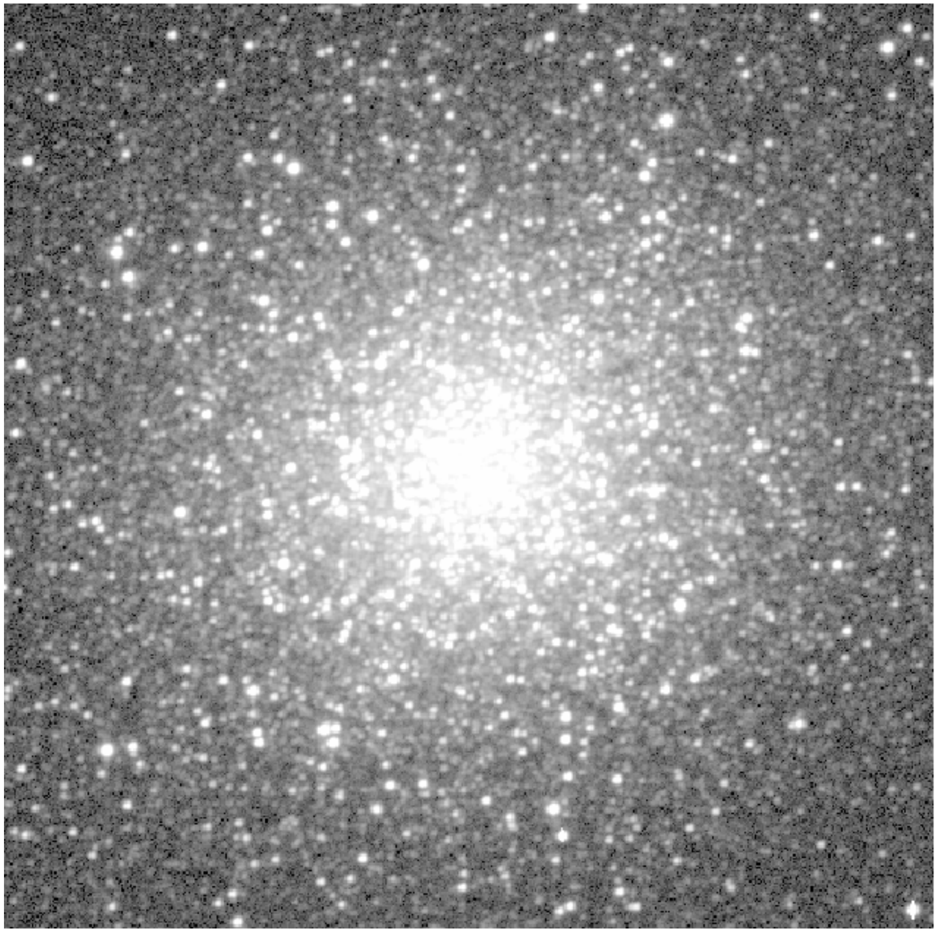
Example convolution kernels to match *sci* and *ref* image PSFs for the M13 test images on slides 10 and 11



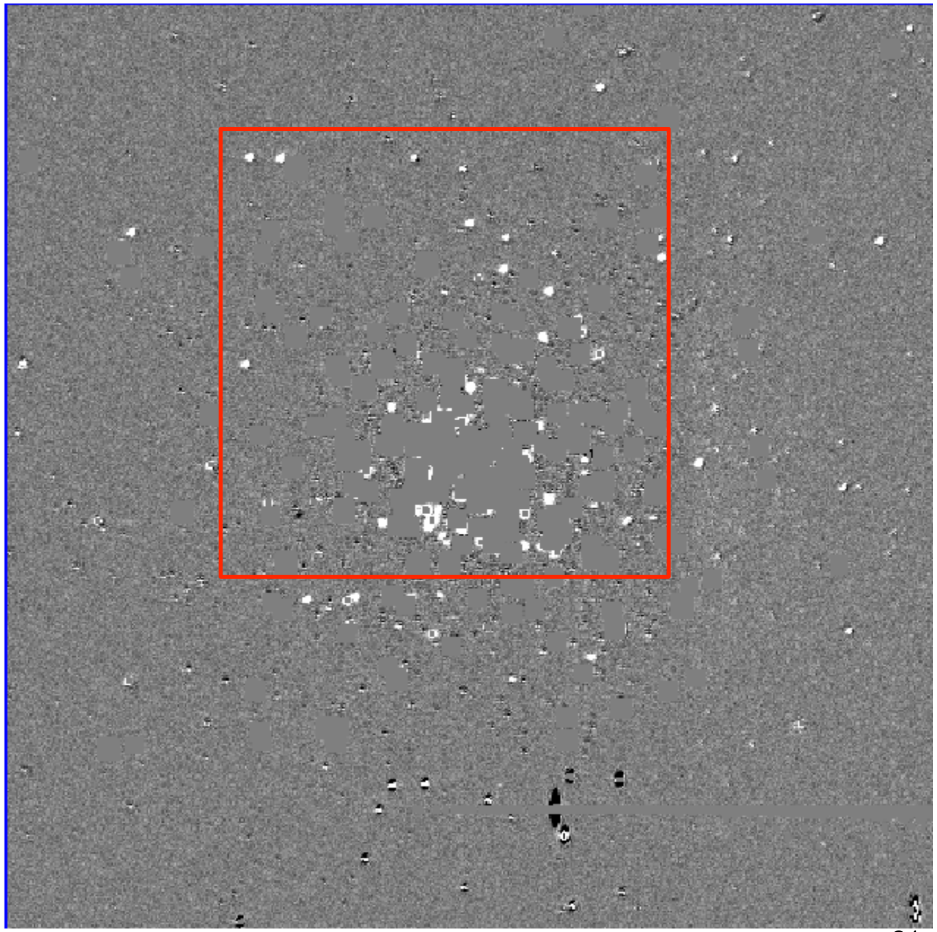
Zoom on M13 globular cluster

- lots of RR Lyrae variables!
- bad/saturated pixels in difference replaced by zero here

science image exposure (~ 9' x 9' *zoom*)



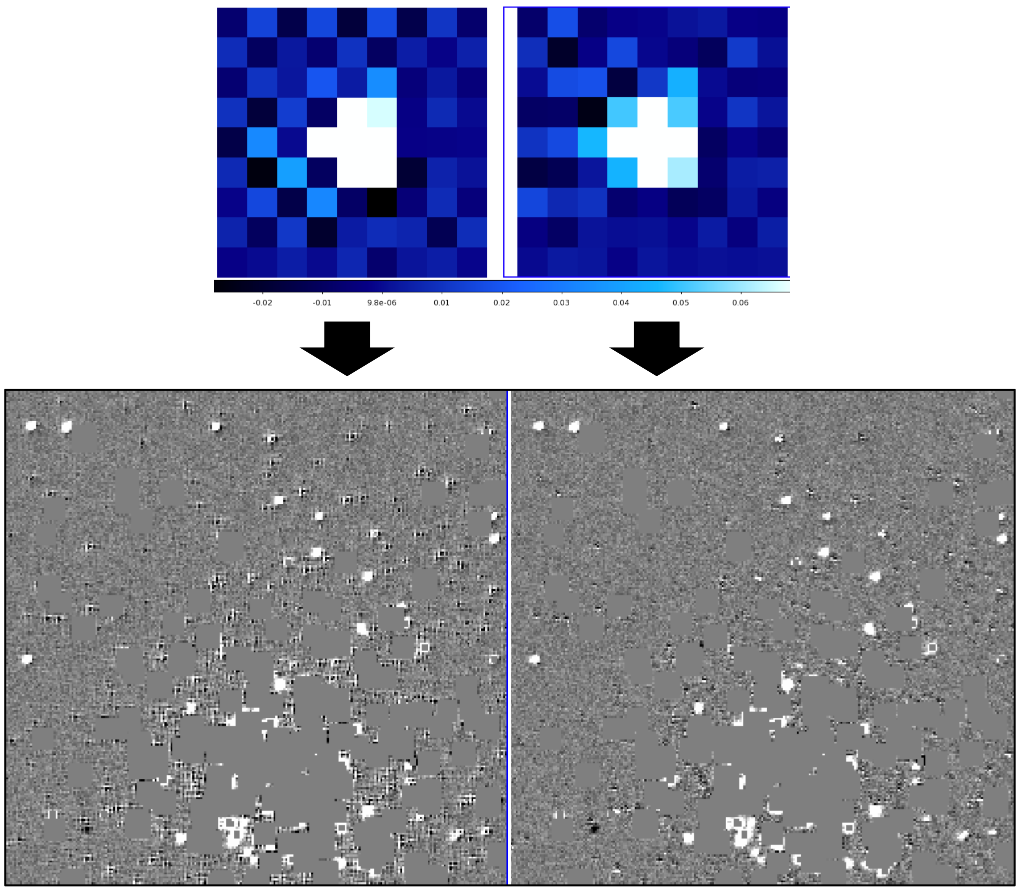
sci – K (x) *ref difference image*



Further zoom on M13 globular cluster

Comparison of difference images using PSF-matching kernels with/without regularization (via a truncated SVD)

unregularized Kernel *regularized Kernel*



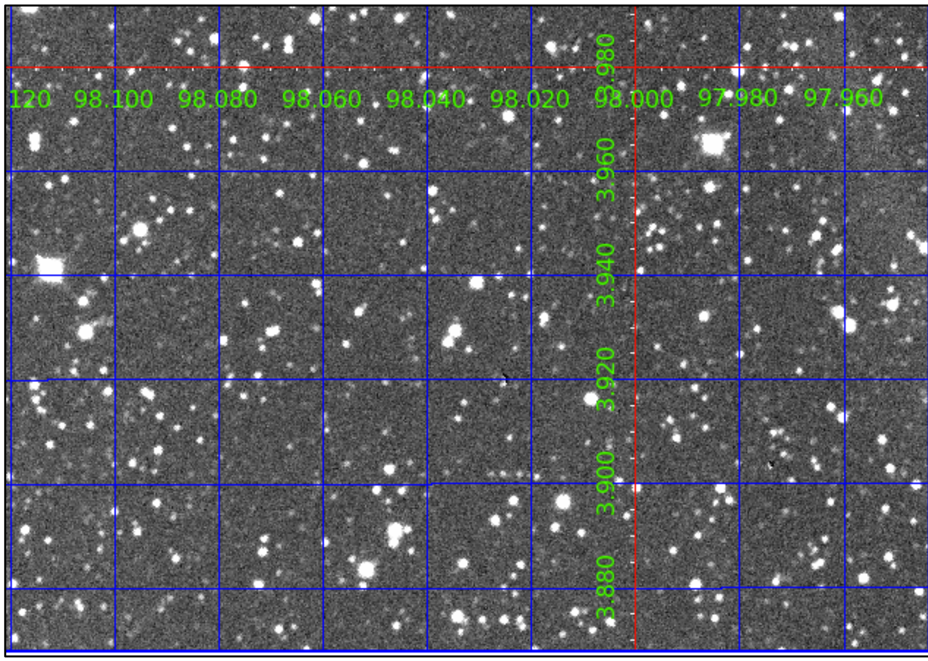
Sci – K (x) *Ref difference images*

“Good” difference in Galactic Plane

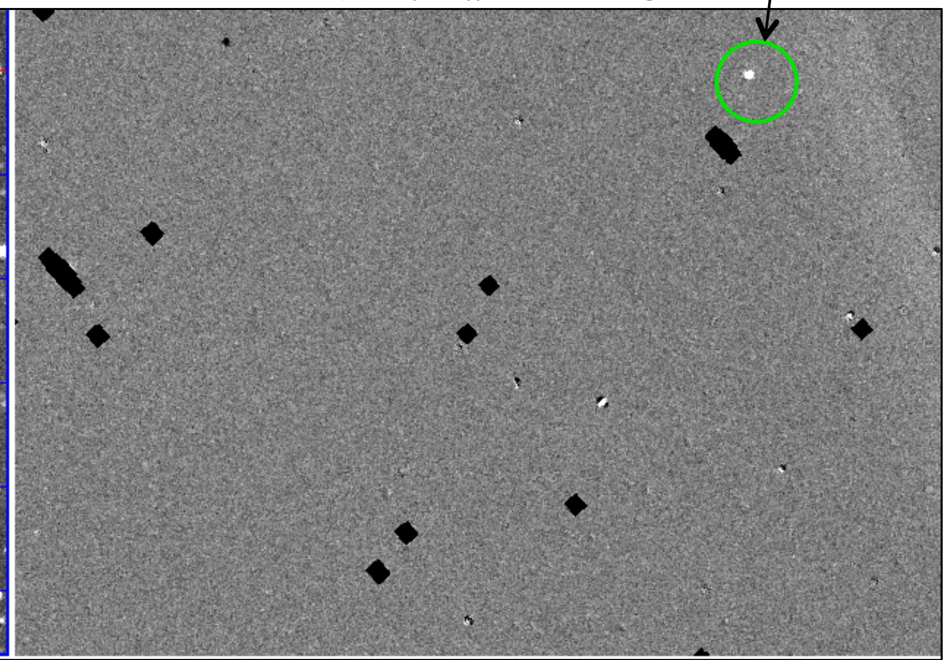
When upstream astrometric/distortion calibration is near perfect, it works!

known variable

science image exposure (~ 10’ x 7’ *zoom*)



Sci – K (x) *Ref difference image*

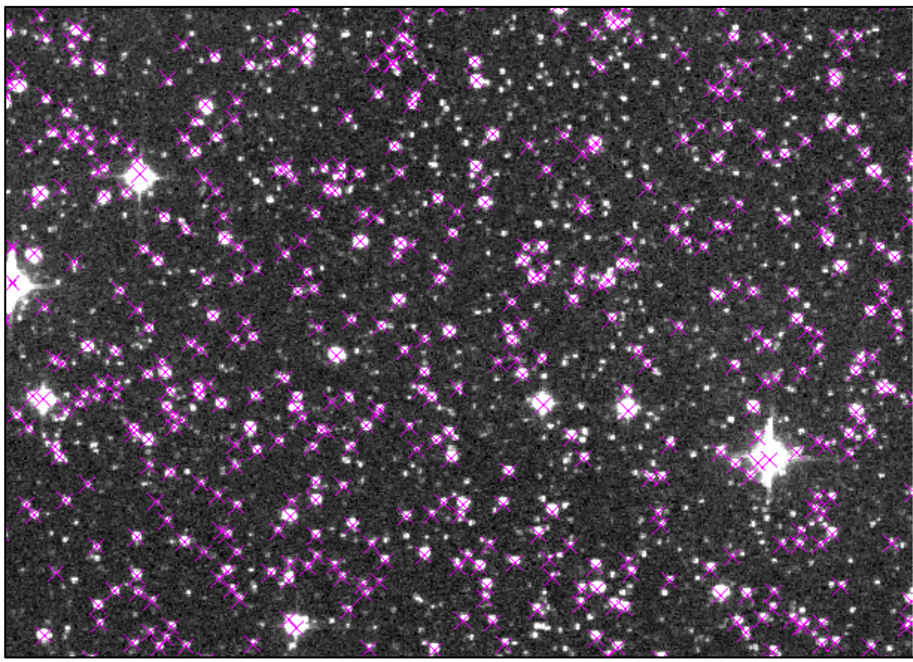


coordinate grid is galactic

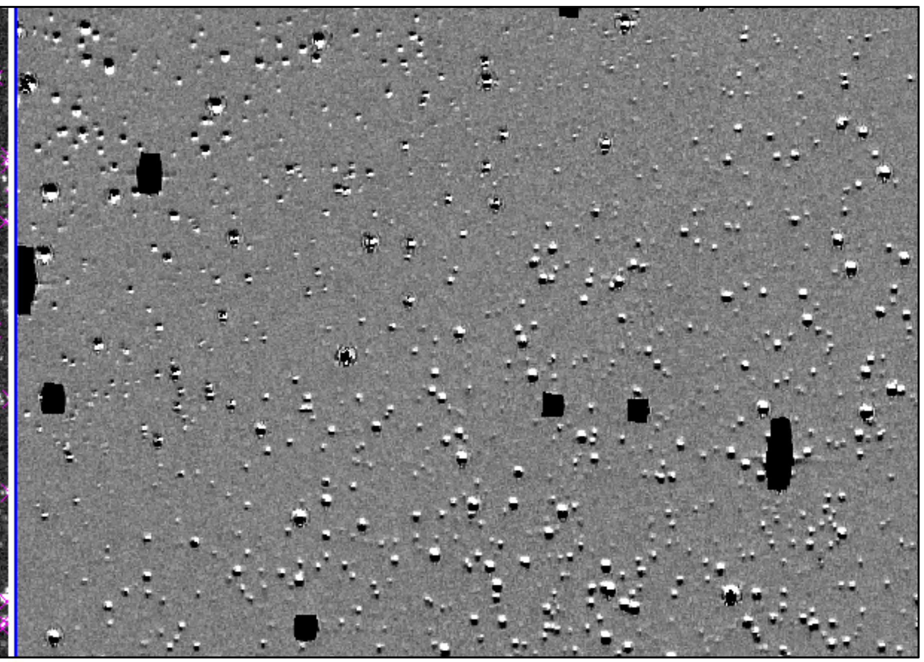
“Bad” difference in Galactic Plane

- When upstream astrometric/distortion calibration was “slighty” wrong
- Bad distortion calibration => spatially-dependent astrometric residuals => usually fast variations on small scales that are difficult to correct/compensate using PSF-matching kernel
- Too complex to include in kernel model! Won’t have enough *d.o.f.* to enable fit

science image exposure (~ 12’ x 8’ *zoom*)



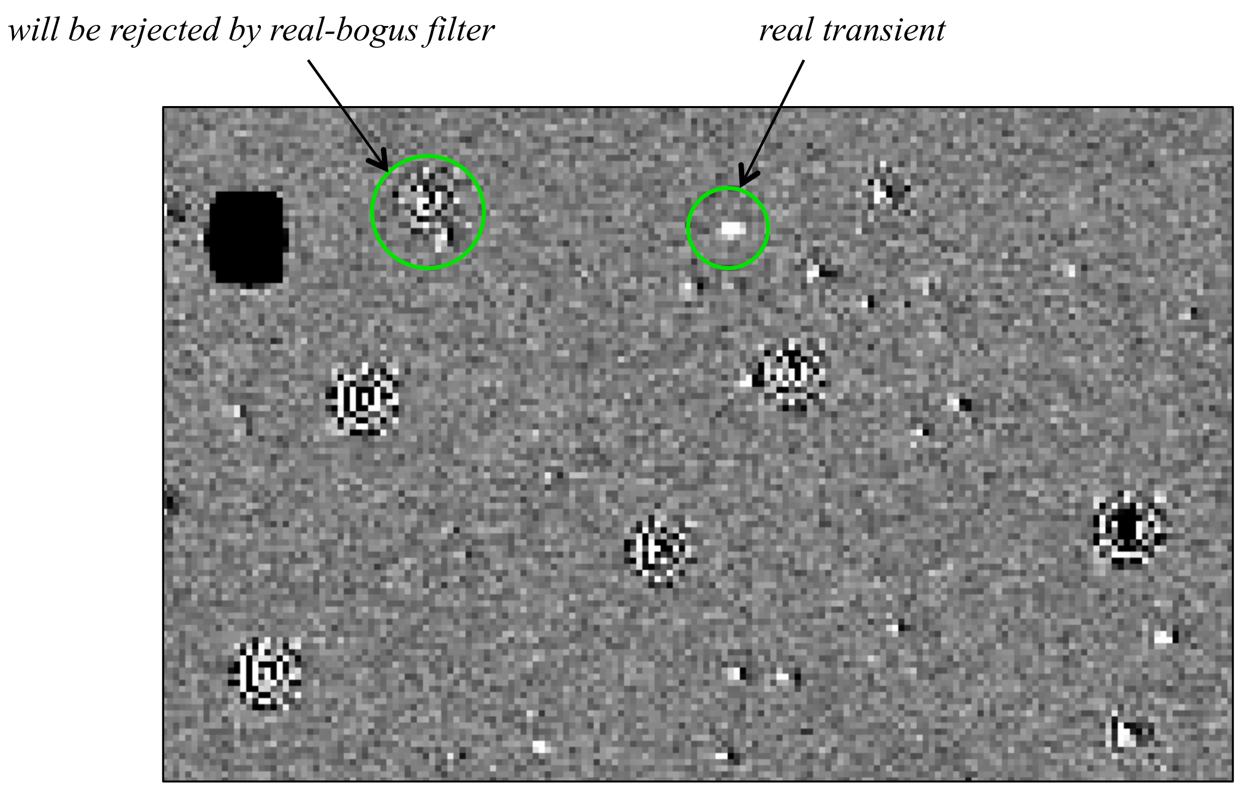
Sci – K (x) *Ref difference image*



magenta crosses: 2MASS positions

Another hiccup: convolution direction

- When a reference image (*inadvertently*) has a larger PSF *FWHM* than seeing *FWHM* in science exposure and direction of convolution is fixed to always convolve reference, convolution is ill-posed and residuals result
- Can be easily fixed by convolving science exposure instead prior to differencing
- There is an option in PTFIDE to automate the selection of images to derive/apply convolution kernels
- However, to minimize ambiguities due to noise, plan is to always convolve reference (known *a-priori* to be sharper)



Candidate transient photometry

- Performed using both PSF-fitting and aperture photometry on difference images

- PSF-fitting provides better photometric accuracy to faint fluxes; de-blending ability (if subtractions bad!)

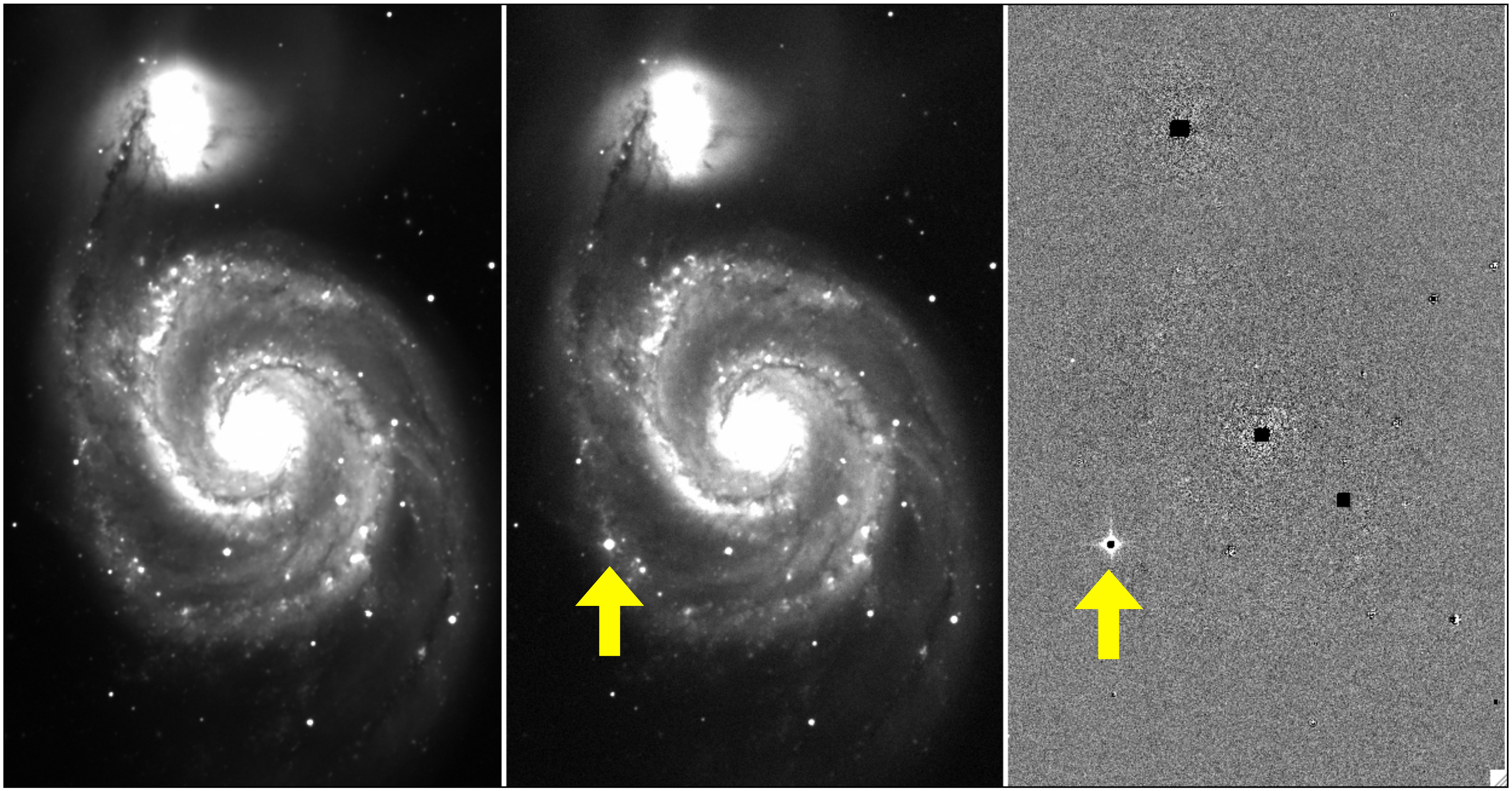
- Where does PSF that's used on a difference-image come from?
  - due to linearity of convolution and differencing process, spatially varying PSF is derived using deeper (and cleaner) *reprojected* and *kernel*-convolved reference image
  - this PSF is used both for detection (point-source matched-filtering) and fitting (photometry)

- Provides diagnostics to distinguish point sources from glitches (false-positives) in diff. images
  - maximizes reliability of difference-image extractions since most transients are point sources

- Above assumes accurate PSF-estimation (over chip) and astrometry prior to differencing

- Aperture photometry, source-shape metrics, and a plethora of other metrics are also generated

SN 2011dh (PTF11eon) in Messier 51



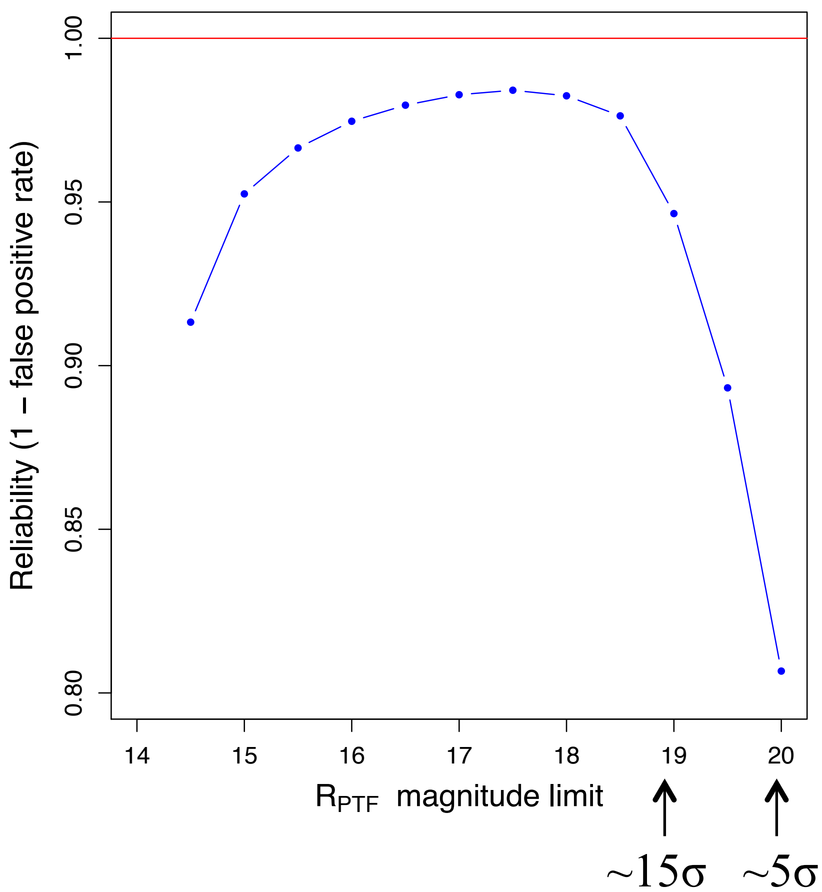
Reference image = co-add of 20 R exposures (pre-outburst)

R exposure on June 19, 2011
Type IIb supernova ~ $10^9 L_\odot$

Difference image:
sci exposure - reference

Performance: real vs. bogus (reliability)

- with no real-bogus vetting yet in place, explored reliability of raw extractions using a simulation
- took 350 real, moderately dense *R*-band frames, derived spatially-varying PSFs, then simulated point source transients with random positions and fluxes
- executed PTFIDE to create diff images and extract candidates with **fixed** threshold (S/N = 4) and filter params.



$$R = \frac{\#\ \text{matched to truth}\ (< R_{mag})}{\#\ \text{total extracted}\ (< R_{mag})}$$

Difference-image based metrics to support machine-learned vetting

Loaded into a database table during real-time processing

Metric name	Description
isdiffpos	t = positive difference, f = negative difference
medksum	Median pixel-sum of all raw convolution kernels
minksum	Minimum pixel-sum of all raw convolution kernels
maxksum	Maximum pixel-sum of all raw convolution kernels
medkdb	Median differential background over all raw convolution kernels (DN)
minkdb	Minimum differential background over all raw convolution kernels (DN)
maxkdb	Maximum differential background over all raw convolution kernels (DN)
medkpr	Median 5th to 95th percentile pixel range of all raw convolution kernels
minkpr	Minimum 5th to 95th percentile pixel range of all raw convolution kernels
maxkpr	Maximum 5th to 95th percentile pixel range of all raw convolution kernels
zpdiff	Photometric zero point of difference image (mag)
nbadpixbef	Number of bad pixels before PSF-matching
nbadpixaft	Number of bad pixels after PSF-matching
medlevbef	Median level before PSF-matching (DN)
medlevaft	Median level after PSF-matching (DN)
avglevbef	Average level before PSF-matching (DN)
avglevaft	Average level after PSF-matching (DN)
medsqbef	Median of squared differences before PSF-matching (DN^2)
medsqaft	Median of squared differences after PSF-matching (DN^2)
avgsqbef	Average of squared differences before PSF-matching (DN^2)
avgsqaft	Average of squared differences after PSF-matching (DN^2)

Continued….

Difference-image based metrics continued…

Metric name	Description
chisqmedbef	Chi-square from median before PSF-matching
chisqmedaft	Chi-square from median after PSF-matching
chisqavgbef	Chi-square from average before PSF-matching
chisqavgaft	Chi-square from average after PSF-matching
scibckgnd	Modal bckgnd level in science image after gain and bckgnd matching (DN)
refbckgnd	Modal bckgnd level in ref image after gain, bckgnd matching,resampling (DN)
scisigpix	Robust sigma/pixel in science image after gain and background matching (DN)
refsigpix	Robust sigma/pixel in ref image after gain, bckgnd matching,resampling (DN)
scimaglim	Expected 5-sigma mag limit of sci image after gain & bckgnd matching (mag)
refmaglim	Expected 5-sigma limit of ref image after gain, bckgnd matching, resampling
diffbckgnd	Median background level in difference image (DN)
diffpctbad	Percentage of difference image pixels that are bad/unusable (%)
diffsigpix	Robust sigma/pixel in difference image (DN)
diffmaglim	Expected 5-sigma magnitude limit of difference image (mag)
sciinpseeing	Seeing (point source FWHM) of input science image (pixels)
refinpseeing	Seeing (point source FWHM) of input reference image (pixels)
refconvseeing	Seeing (point source FWHM) of reference image after convolution (pixels)
ncandscimrefraw	Number of candidates from sci - ref diff image before internal filtering
ncandscimreffilt	Number of candidates from sci - ref diff image after internal filtering
ncandrefmsciraw	Number of candidates from ref - sci diff image before internal filtering
ncandrefmscifilt	Number of candidates from ref - sci diff image after internal filtering
ncandscimrefgood	Number of candidates from sci - ref diff image likely to be real using cuts
ncandrefmscigood	Number of candidates from ref - sci diff image likely to be real using cuts
ncandscimrefratio	ratio: ncandscimreffilt/#sci extractions
ncandrefmsciratio	ratio: ncandrefmscifilt/#sci extractions
status	Good/bad difference image (1/0) based on internal image QA filtering

Candidate-transient metrics (features) to support machine-learned vetting

Also loaded into a database table during real-time processing

Metric name	Description
magpsf	Magnitude from PSF fit (mag)
sigmagpsf	1-sigma uncertainty in PSF-fit magnitude (mag)
flxpsf	Flux from PSF fit (DN)
sigflxpsf	1-sigma uncertainty in PSF-fit flux (DN)
snrpsf	flxpsf / sigflxpsf
magap	Magnitude from aperture photometry (mag)
sigmagap	1-sigma uncertainty in magap (mag)
flxap	Flux from aperture photometry (DN)
sigflxap	1-sigma uncertainty in flxap (DN)
sky	Local sky background level (DN)
nneg	Number of negative pixels in a 7x7 box
nbad	Number of bad pixels in a 7x7 box
distnr	Distance to nearest reference image extraction (arcsec)
magnr	Magnitude of nearest reference image extraction (mag)
sigmagnr	1-sigma uncertainty in magnr (mag)
chi	Chi value from PSF fit
sharp	Sharpness value from PSF fit
nneg2	Number of negative pixels in a 5x5 box
nbad2	Number of bad pixels in a 5x5 box
magdiff	Magnitude difference: magap - magpsf (mag)

Continued….

Candidate-transient metrics (features) continued…

Metric name	Description
aimage	Windowed RMS along major axis of source profile (pixels)
aimagerat	Ratio: aimage / fwhm
bimage	Windowed RMS along minor axis of source profile (pixels)
bimagerat	Ratio: bimage / fwhm
elong	Elongation = aimage / bimage
fwhm	FWHM from Gaussian profile fit (pixels)
seeratio	Ratio: fwhm / (average fwhm of science image)
arefnr	aimage (major axis RMS) of nearest reference image extraction (pixels)
brefnr	bimage (minor axis RMS) of nearest reference image extraction (pixels)
normfwhmrefnr	Ratio: (fwhm of nearest ref image extraction) / (average fwhm of ref image)
mindistoedge	Distance to nearest edge in frame (pixels)
elongnr	Elongation of nearest reference image extraction (= arefnr/brefnr)
magfromlim	Magnitude difference: diffmaglim - magpsf (mag)
ksum	Pixel sum of psf-matching kernel for image partition (= gain residual)
kdb	Delta bckgnd associated with psf-matching kernel for image partition (DN)
kpr	5th to 95th percentile pixel range of psf-matching kernel for partition

Summary / Lessons learned

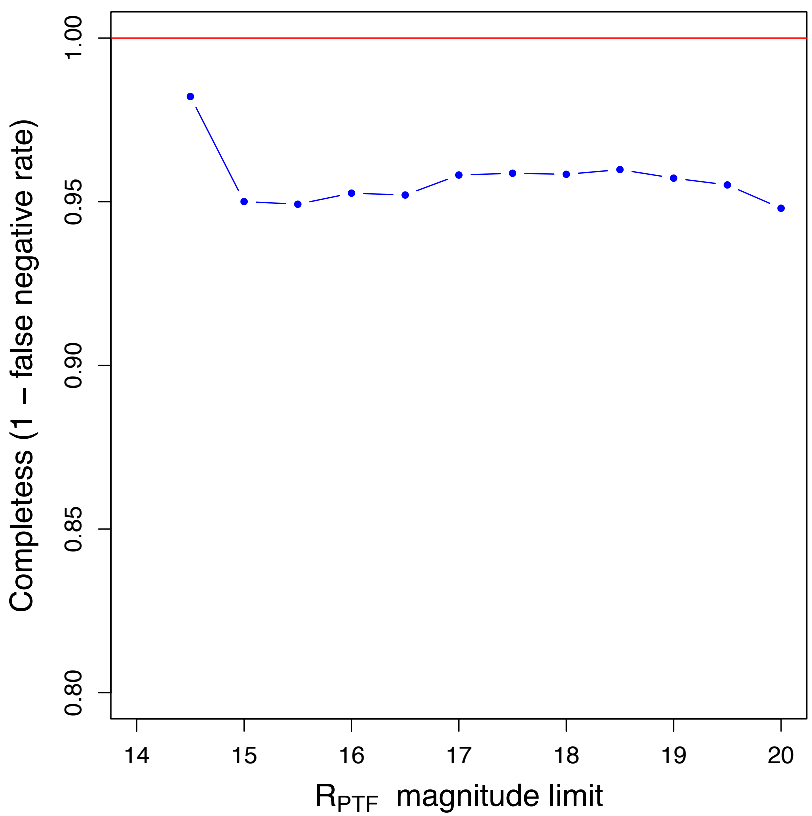
- The transient-discovery engine PTFIDE is now running in near real-time at IPAC/Caltech to support discovery and archival research for the intermediate Palomar Transient Factory (iPTF)
- Algorithms and software are generic. Plan to use on future projects: ZTF…
- Machine-learned vetting (real-bogus) infrastructure is currently in progress (training phase)
- Validation and testing continues, particularly in crowded fields

- Things to note from (limited) experience:
  - Need optimal instrumental calibration of science exposures: astrometry and Field-of-View distortion calibration must be accurate
  - PSF-matching kernel: ensure have enough stars (to build S/N) on spatial scales at which PSF is expected to vary: want maximal #D.O.F. that avoids over-fitting and minimizes bias
  - Automated vetting (QA) system to weed out false positives from difference images, or at least store source metrics in a DB for later: provides feedback for tuning thresholds
  - Have a reference image library in place, together with QA: update products as better quality science images become available (if needed)
  - Need accurate *absolute* astrometric and photometric calibration of reference images if used for relative calibration (refinement) of science exposures before differencing

Back up slides

Performance: completeness

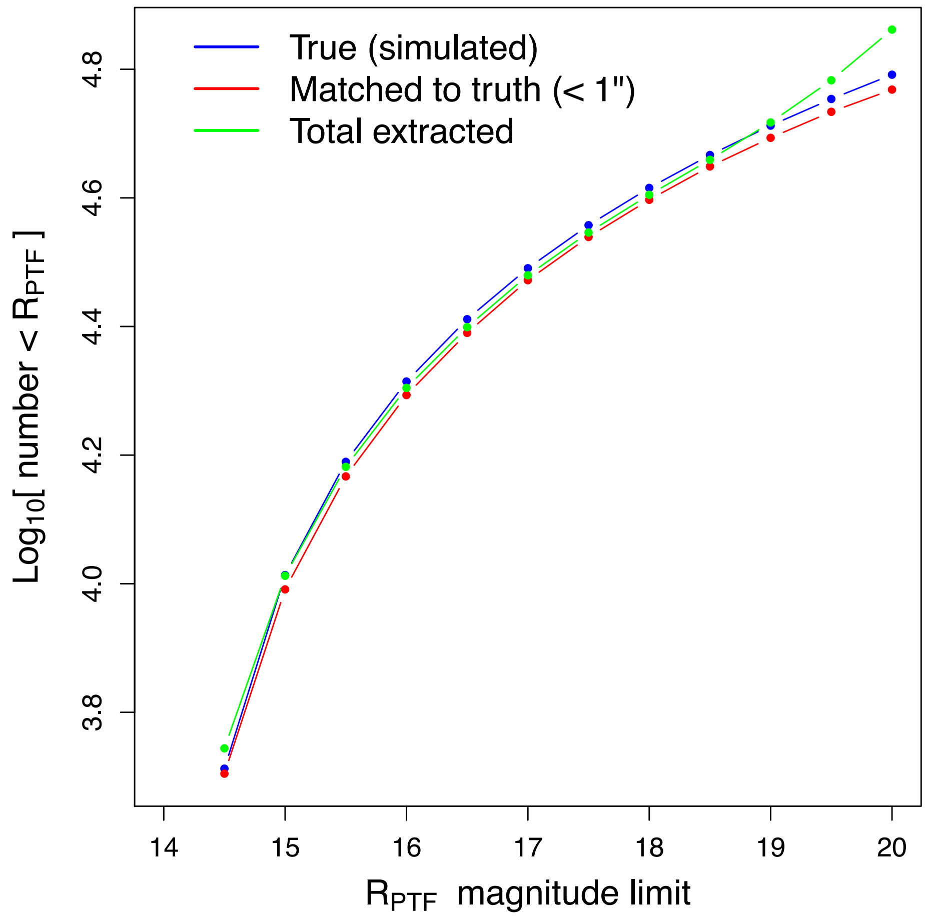
- took ~350 real, moderately dense *R*-band frames, derived spatially-varying PSFs, then simulated point source transients with random positions and fluxes.
- executed PTFIDE to create diff images and extract candidates with **fixed** threshold (S/N = 4) and filter params.



$$C = \frac{\#\ matched\ to\ truth\ (< R_{mag})}{\#\ total\ truth\ (< R_{mag})}$$

Performance: #extractions vs "truth"

- took ~350 real, moderately dense *R*-band frames, derived spatially-varying PSFs, then simulated point source transients with random positions and fluxes.
- executed PTFIDE to create diff images and extract candidates with **fixed** threshold (S/N = 4) and filter params.



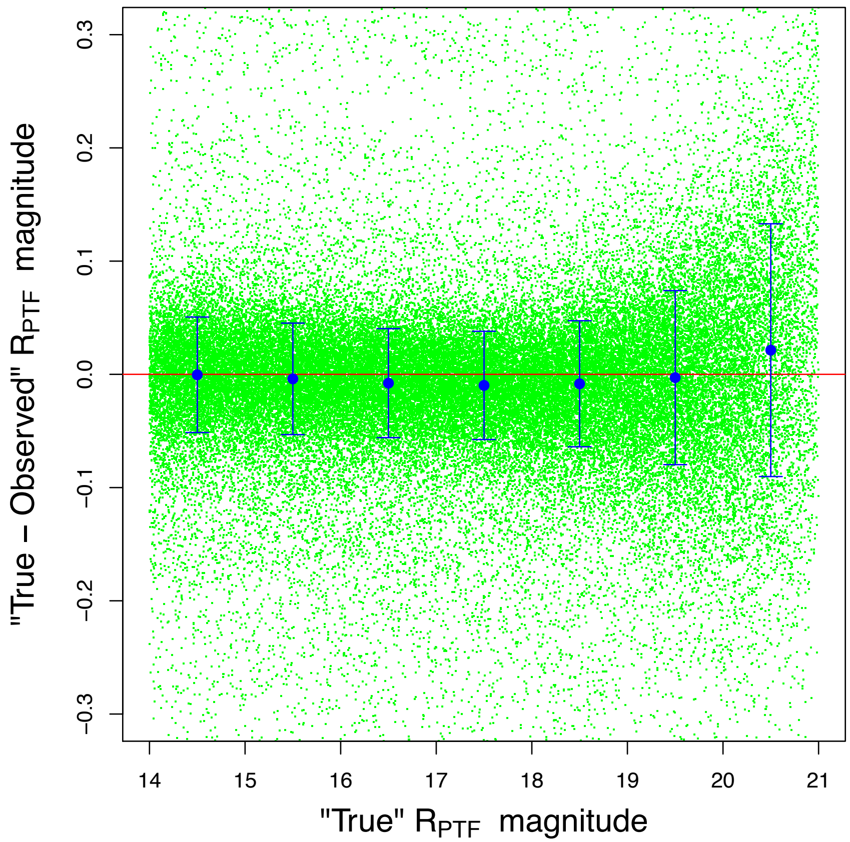
Completeness and Reliability:

$$C = \frac{\#\ matched\ to\ truth\ (< R_{mag})}{\#\ total\ truth\ (< R_{mag})}$$

$$R = \frac{\#\ matched\ to\ truth\ (< R_{mag})}{\#\ total\ extracted\ (< R_{mag})}$$

Performance of PSF-fit (AC) photometry

- took ~350 real, moderately dense *R*-band frames, derived spatially-varying PSFs, then simulated point source transients with random positions and fluxes
- then executed PTFIDE to create diff images and extract candidates
- difference image (AC) fluxes are consistent with truth



SN PTF10xfh

Type Ic supernova in NGC 717 at ~ 65 Mpc (Yi Cao, private communication)