WSDC Subsystem Peer Review

Frame Co-addition

Frank Masci
IPAC/Caltech

Outline

1. Atlas (co-add) Image deliverables and strategy
2. Co-add pipeline overview
3. Preprocessing steps:
   - outlier detection strategy and plans
   - background/frame-overlap matching
4. Co-addition goals and philosophies
5. The AWAIC algorithm
6. Some examples
7. Noise characterization
8. To do list

Deliverables (Co-add products)

- Digital Image Atlas will consist of co-adds that combine multiple frame exposures within pre-defined regions on the sky in each of the four bands: 3.3, 4.7, 12 and 23μm

- For each band, the plan is to have three products (all same dimensions):
  - Main intensity co-add image
  - Associated depth-of-coverage map indicating effectively the number of unmasked (good) pixel contributions
  - Uncertainty co-add image that contains the 1-σ error estimate in the co-add signal for every pixel

- Baseline plan is for an Atlas Image footprint to consist of 2048 × 2048 pixels at 1 arcsec/pixel
  - gives ≈ 34.1 arcmin linear size
  - with ~1 arcmin *minimum* linear overlap between tiles arranged along iso-declination bands ⇒ 136,090 tiles

- Initial proposal was to have 1.375 arcsec/pixel (half the native pixel scale of bands 1-3, a quarter of band 4)
  - gives linear sizes ≈ 47 arcmin.
  - With ~1 arcmin *minimum* linear overlap, ⇒ 70,985 tiles on the sky. Sizes and scales are TBD.

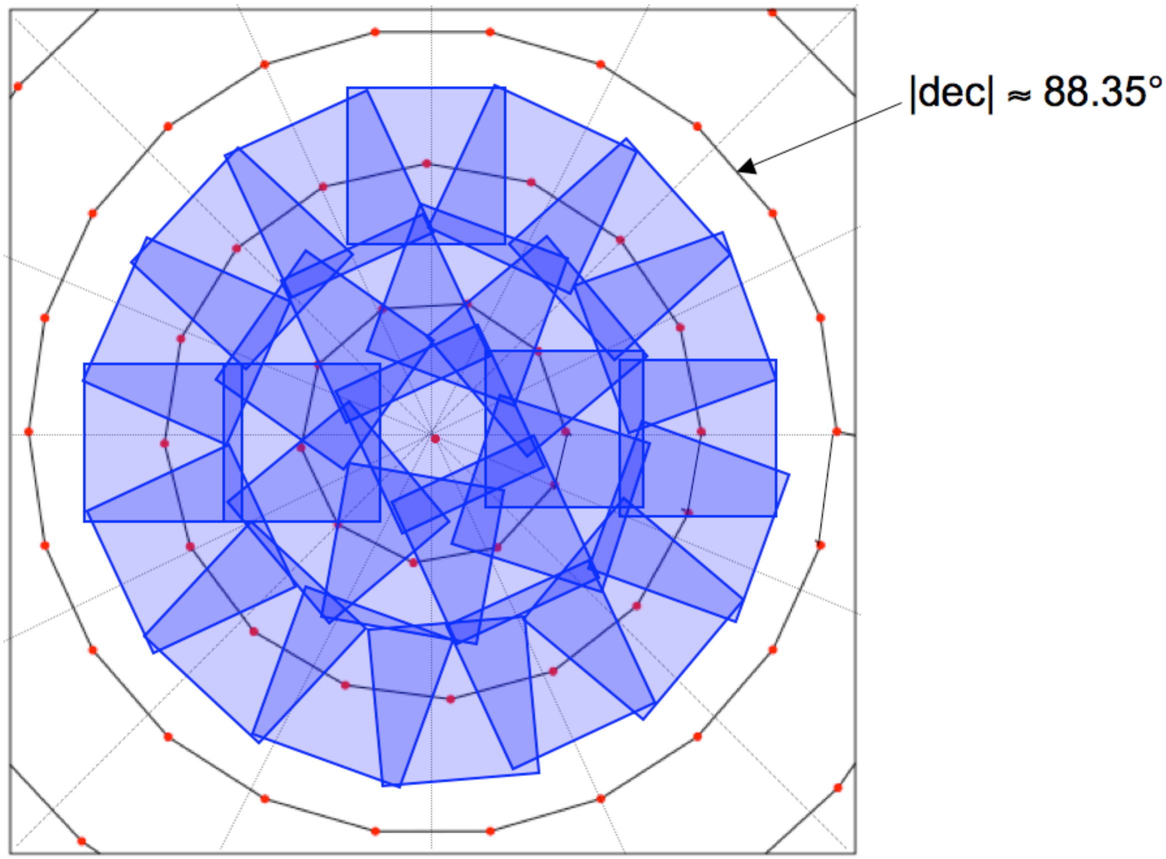
- Given 4 bands and 3 co-add products per band, Atlas Image archive will be ~15 TB and 28 TB (uncompressed) for pixel scales of 1.375 and 1 arcsec/pixel respectively. Assumes 2k × 2k pixel tiles, 1 arcmin *min* overlap and 4 Bytes/pixel.

- Atlas Image products will be accessed via a WISE image server interfacing with IRSA. Users can retrieve a co-add portion based on sky location, spatial extent, orientation and pixel scale. Maximum query size TBD.

Tiling Geometry

Below is a schematic of the centers of the Atlas Image footprints (tiles) at an equatorial pole. This assumes:

- Position angles of all image tiles are zero;
- A *minimum* overlap between any two adjacent tiles of 1 arcmin in both Dec and RA;
- Tile sizes of ~34.1 arcmin (2k × 2k pixels at 1 arcsec/pixel);
- Tiles are aligned within 325 iso-declination bands.



COADD Process Overview

INPUTS:
- instrumentally and phot. calibrated, pointing refined science frames (level-1B): bands 1-3:1016^2; band 4: 508^2;
- accompanying "bad" pixel masks (*also first pass radhits*);
- accompanying uncertainty (sigma) frames;
- list is queried from region centered on predefined sky tile.

Interpolate frames (using PRF as kernel or other?) onto common co-add grid.

Outlier detection on interp. pixel stacks using robust methods (e.g., median, MADs, quantiles). *Frame masks updated.*

?

Not always: check for consistency between computed uncerts and frame repeatability using χ^2 stats. *Update uncerts if inconsistent.*

- Gain/throughput matching (multiplicative);
- *Photometric zero point derived for co-add.*
- Frame background matching (additive).

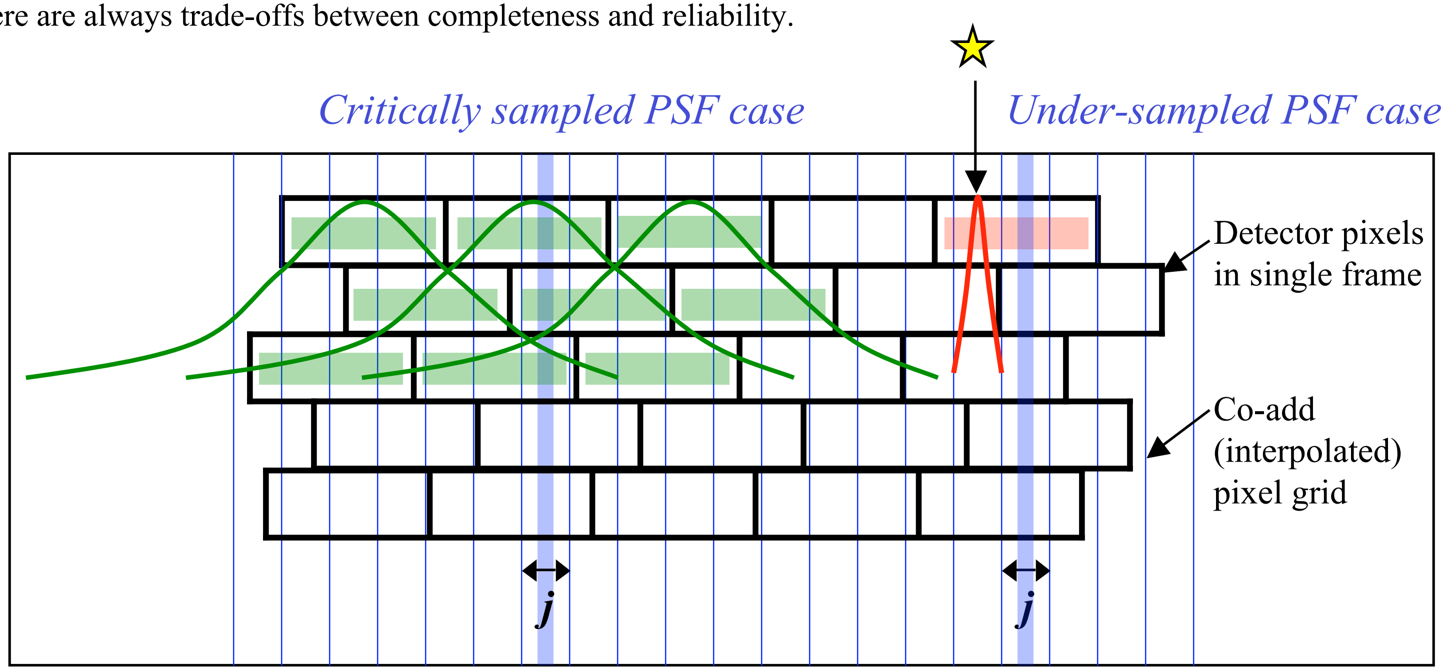
Co-addition of all good (unmasked) detector pixels using **AWAIC:** *A WISE Astronomical Image Co-adder.* Uses a more optimal interpolation procedure.

OUTPUTS:
- main intensity image (rate $\propto$ DN/t units), MAGZP in HDR.
- coverage map/image
- uncertainty image
- all 2048 x 2048; 1″ / pix (TBD)
- QA metrics

Atlas Image FPG; products registered in database and indexed. Will interface with IRSA's WISE image server.

Outlier Detection

- Single frame outlier detectors (e.g., gradient filters) are not effective at picking up PRF-like transients and artifacts. We will take advantage of the redundancy in multiple frame exposures and flag inconsistent measurements in the temporal domain.
- This is a black art. For a lack of better and faster methods from other projects/literature, we adopt the brute force approach where all frames are first projected and interpolated onto a common grid, then an outlier identification algorithm is applied to each pixel stack. More later.
- It helps to have good sampling of the PSF for temporal outlier rejection! If so, more than one detector pixel from the same frame (shaded *green*) can contribute (through its PRF) to co-add pixel *j*.
- If under-sampled (even moderately so), there is the possibility of flagging real sources as outliers! Here the *red pixel* will be erroneously identified as an outlier with respect to samples stacked in co-add pixel *j*. => Reliability plummets. Solution: Make the outlier search window bigger and bump up the threshold. But will now miss the weak outliers.
- There are always trade-offs between completeness and reliability.



Outlier Detection method..

- **Note:** WISE will be better than critically sampled across all bands. From June '07 CDR:
  PSF_{FWHM} / pix scale ≈ 2.2, 2.5, 4.2, 3.4 pixels for bands 1, 2, 3 and 4 respectively.

- We need a generic outlier detection/flagging method that:
  (1) Allows outlier information to be efficiently propagated to the co-adder. Back-tracking to find the offending input pixels to mask will be difficult.
  (2) Avoids memory overflow when performed at the poles where WISE will achieve coverage-depths of 800-1000 frames.

Main steps:

1. Project and interpolate frame pixels onto a common grid and store values from all image stacks. To circumvent memory overflow, we can partition the co-add grid into sub-areas, then identify outliers in each separately.
2. For each pixel stack in interpolated space, compute *robust* moment measures of their flux distribution and save as image files on disk, e.g., median and sigma measures from either quantiles or the MAD measure:

$$\sigma_j \approx 0.5(p_{84} - p_{16}) \quad \text{or} \quad \sigma_j \approx 1.483\, median\{|p_i - median\{p_i\}|\}$$

3. When building the final co-add, read in the median and robust sigma images, re-project and re-interpolate the input frame pixels onto the co-add grid. If an interpolated value p_i in stack j satisfies the following criteria, omit the input pixel entirely from the co-add.

$$p_i > median\{p_i\} + t_{thres}\sigma_j$$

or

$$p_i < median\{p_i\} - b_{thres}\sigma_j$$

Outlier Detection method..

The above method has the advantage that:

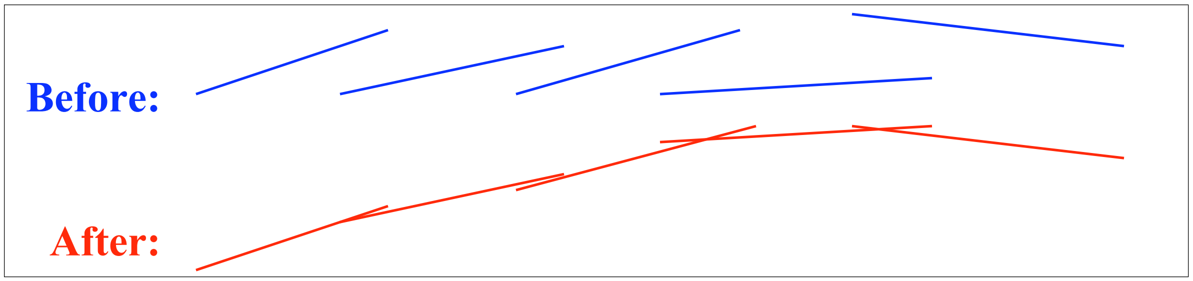
- Outliers are identified and omitted on the fly during the co-addition process. There is no fussing with propagation via frame masks. If needed for QA (or detecting moving objects), the outliers can be stored in a ‘co-add mask’ (TBD).
- It provides an independent *robust* measure of sigma as represented by the redundant measurements across multiple frame exposures. These can be used to check (and re-scale if necessary) the computed co-add sigmas that are initiated upstream using an error model.

In the upcoming months, the plan is to:

- Implement the above method in software.
- Use simulated data frames containing outliers (cosmic rays) of different strengths, including ones at grazing incidence that cause streaks. Explore completeness and reliability across all bands. Tune thresholds.
- Supplement with a single frame (“sharp edged artifact”) outlier detector to see if there are improvements in completeness and reliability across the different outlier regimes and types.

Background Matching

- Goal: obtain seamless (or smooth) transitions between frames across overlaps in a co-add. We want to equalize background levels on frame-to-frame scales but preserve natural background variations if possible.
- Varying background levels may be due to varying instrumental / terrestrial effects and transients.



- The *Spitzer* method:
  - involves global minimization of weighted differences of pixel values I in overlap regions between all pairs of frames. For frame pairs m, n and pixels k^m, k^n therein, the metric is:

$$\mathbf{L} = \sum_{m,n=1 (m \neq n)}^{N_{images}} \sum_{k \in overlap} \frac{(I^n(k^n) - I^m(k^m))^2}{\sigma_n^2(k^n) + \sigma_m^2(k^m)}$$

  - Solve for global offsets ε^n for each input frame where $I^n(corrected) = I^n - \varepsilon^n$
  - Needs to be made more robust to avoid sources affecting the offset estimates (e.g., smooth images first with a low pass filter?).
  - Also, if gradients vary wildly between frames (no reason why they should), we may get residual non-matching backgrounds on *local* scales. Reason: this is a global method that corrects for offsets only.

Background Matching

- The *2MASS* method:
  - Project/interpolate a frame onto co-add grid;
  - Compute the median of the difference between this incoming frame and the pixel values already in the co-add (if any) in the region where they overlap;
  - "Background correct" the incoming frame by subtracting this median difference from its pixels;
  - Build the co-add by re-projecting the background subtracted frame pixels;
  - Repeat above steps for each incoming frame;

  This method is more robust than the *Spitzer* method (less subject to source effects).

- The *Montage* method:
  - Uses a *local* minimization (least squares) method where the overlap area of each image is considered with respect to its neighbors.
  - Both gradient and offset corrections are computed by using planar surface fits to difference images in the overlap regions.
  - A real possibility!

Background Matching

- Note: before any generic background matching algorithm is applied, gain/throughput matching of the input frames according to differences in photometric zero-points *may* first need to be applied.
- This will be needed if significant (*systematic*) differences between the input frame photometric zero points are found - e.g., due to varying instrumental effects.
- This step will also involve derivation of a single photometric zero-point for the co-add.

In the upcoming months, the plan is to:

- Experiment with the above methods on simulated WISE data with varying backgrounds (instrumental + real).
- Need to understand the *Montage* method better (technical details).
- Explore limitations of the different approaches.
- Want a method which is fast, robust and simple enough to implement or import from existing software.

Co-addition Goals

- To optimally combine all the available measurements into a *faithful* representation of the sky given all the instrumental effects, limitations, transients, cosmic rays etc.. have been accounted for.
- Another way of looking at this (which I prefer) is to ask: what model or representation of the sky propagates through the measurement process to yield the observations within measurement error?

- The measurement process is effectively a filtering operation performed by the instrument's Point Response Function (PRF):

$$Sky \rightarrow \underbrace{PSF \otimes \Pi}_{PRF} \times \textit{sampling by pixels} \rightarrow \textit{measurements}$$

- The PRF represents the real transfer function. Each pixel collects light (information) from its vicinity with an efficiency described by the PRF. The better the sampling, the larger its domain of awareness.

- The PRF represents the most *optimal* interpolation kernel for use in co-addition and 'reconstructing the sky' from the measurements. For detector measurements D_i, the flux in a co-add pixel j is given by:

$$f_j = \frac{\sum_i r_{ij} D_i}{\sum_i r_{ij}} \quad ; \quad r_{ij} = \text{response at location } j \text{ from a detector pixel at } i.$$

- In co-adders that use overlap-area weighted averaging (e.g., MOPEX, *Montage*, other..), the interpolation weights are the actual overlap areas $r_{ij} = a_{ij}$. In fact, for severely under-sampled PSFs, the above method reduces to area-weighted averaging. In this limit, the PRF becomes top-hat.

Co-addition in AWAIC

- AWAIC - A WISE Astronomical Image Co-adder. **What makes it 'wise'?**

My approach was motivated by two sources:

- First, Martin Weinberg's theoretical work when designing the 2MASS co-adder with Gene Kopan. They asked: what is the most *optimal* interpolation kernel?
  - Optimality criterion: what continuous function can reproduce an *observed* point source profile (in pixelated space) with minimum variance?
  - 2MASS assumed an analytic fit to the seeing profile (2D Gaussian ⊗ with square pixel response) for its interpolation kernel.
  - This played more the role of a "*smoothing kernel*" for reducing pixel-shape bias. In Martin's words: "*square pixels have nothing to do with Nature*". **Note:** 2MASS data was under-sampled.

- Second, if we use the PRF as the interpolation kernel, resolution enhancement (HIRES) through an iterative Richardson-Lucy like process is possible.
  - This was suggested to me by a person with the initials JWF. When I heard it, I was hooked on the idea.
  - **NOTE:** HIRES'ing is not in the WISE baseline plan.
  - HIRES is CPU intensive, much initial set-up is needed, and some cleanup is needed afterwards.
  - Incidentally, a PRF-interpolated co-add results after the very first iteration of the HIRES process.

Co-addition in AWAIC

Advantages of AWAIC over other co-adders *for WISE* (e.g., MOPEX, 2MASS, *Montage*):

- PRF interpolation reduces the impact of bad/masked pixels if the data are well sampled (even close to critical). This leads to effectively non-zero coverage on the sky due to the extended PRF tails.
- Uncertainty estimation from a propagated error model, χ^2 sanity checks, and correlated noise corrections.
- The output signal and noise co-adds can be combined to define the *most optimal* matched filter for point source detection.
  - High frequency noise is smoothed out without affecting the point source signal sought for => SNR of peaks is maximized.
  - This will benefit processing at the WSDC since a point source catalog is one of its release products.
  - Topic for a separate peer review.
- Allows for any interpolation kernel to be specified, i.e., matched to the observations.
- HIRES capability as mentioned (not in WISE baseline plan).

Disadvantages:

- An extended interpolation kernel will "smear" sharp-edged features such as CR spikes on the co-add grid. These can masquerade as real sources if not properly flagged beforehand. Smearing is minimized in area-weighted (or top-hat PRF) interpolation methods where spike artifacts remain more pronounced.
- Noise is correlated on larger spatial scales in the co-add. Must be accounted for when performing photometry off the co-add - either aperture or profile fitting. Correlations are minimized for compact (top-hat) PRFs.

AWAIC Processing Flow

INPUTS:
- instrumentally and photometrically calibrated, background matched science frames with best (refined) WCS and distortion in FITS headers;
- accompanying "bad" pixel masks (with suspect outlier)
- accompanying uncertainty (sigma) frames;
- list of FPA position-dependent PRFs.
- Processing params: pixel size ratios, desired WCS of co-add

↓

Initializations, check inputs for consistency, assign defaults.

↓

Set-up output WCS of co-add (internal) **cell grid** onto which to map an up-sampled PRF with same pixel scale.

↓

AWAIC BRAIN: **Main processing loop (see more detailed flowchart):** project pixels with distortion correction; map and interpolate detector PRF values onto internal cell grid; build up "mean" co-add using PRF-weighted averaging; optionally iterate if HIRES desired.

→

Down-sample and trim all internal cell grid products to desired final co-add pixel scale and dimensions.

↓

OUTPUTS:
- Main intensity image;
- Uncertainty image;
- Depth-of-coverage map;
- Log of I/O, processing status and runtime;
- QA diagnostics;

National Aeronautics and Space Administration
Jet Propulsion Laboratory
California Institute of Technology

AWAIC Brain

Initialize all co-add cell grid pixels *j* to a "flat" model image: $f_j^n = 1$ (*n* = iteration number)

1 Read/store:
- image data for single frame *m*
- corres. uncertainty data
- corres. pixel mask data

2 Given <u>unmasked</u> detector pixel *i* in frame *m*, project (and undistort) into co-add cell grid frame using "fast" projection algorithm (see Fig. pg.17)

3 Transform PRF centered on pixel *i* to frame of cell grid. Interpolate PRF pixels onto cell grid pixels by matching to nearest neighbor or using bilinear interp. (see Fig. pg.18)

4 Multiply PRF (r_{ij}) of pixel *i* with underlying model image (f_j) to obtain predicted flux for detector pixel *i*:

$$F_i^n = \sum_j r_{ij} f_j^n$$

5 Compute "measured-to-predicted" correction factor for detector pixel *i*:

$$C_i^n = \frac{D_i}{F_i^n}$$

6 Increment PRF-weighted (and inverse-variance weighted) average correction factor cell-grid arrays:

$$\langle C_j^n \rangle = \frac{\sum_i \frac{r_{ij}}{\sigma_i^2} C_i^n}{\sum_i \frac{r_{ij}}{\sigma_i^2}}$$

7 Increment uncertainty and coverage map arrays. Coverage map:

$$N_j = \sum_i r_{ij}$$

Repeat steps 2[1]→7 for all pixels *i* in frame *m*, [and all frames in input list]

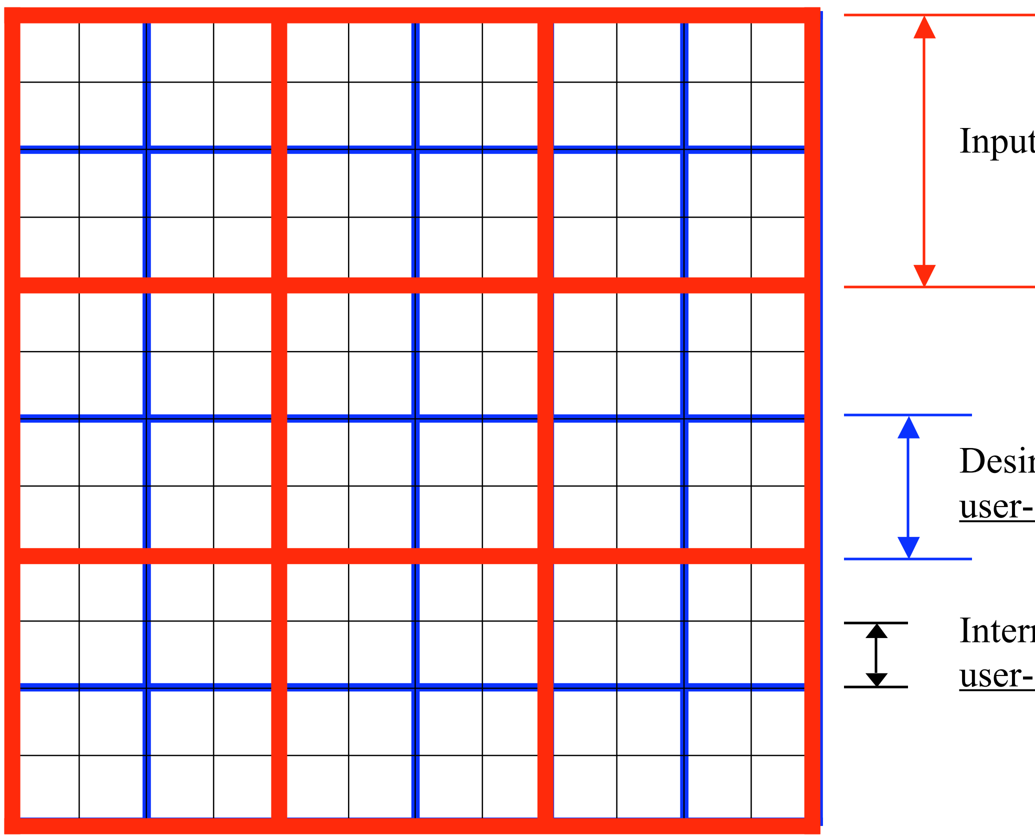
8 Update "model" co-add image array:

$$f_j^{n+1} = f_j^n \langle C_j^n \rangle$$

If after simple co-add <u>stop here</u>, else..

if HIRES desired (n > 1)

Input vs. Co-add vs. Cell Pixel Grids



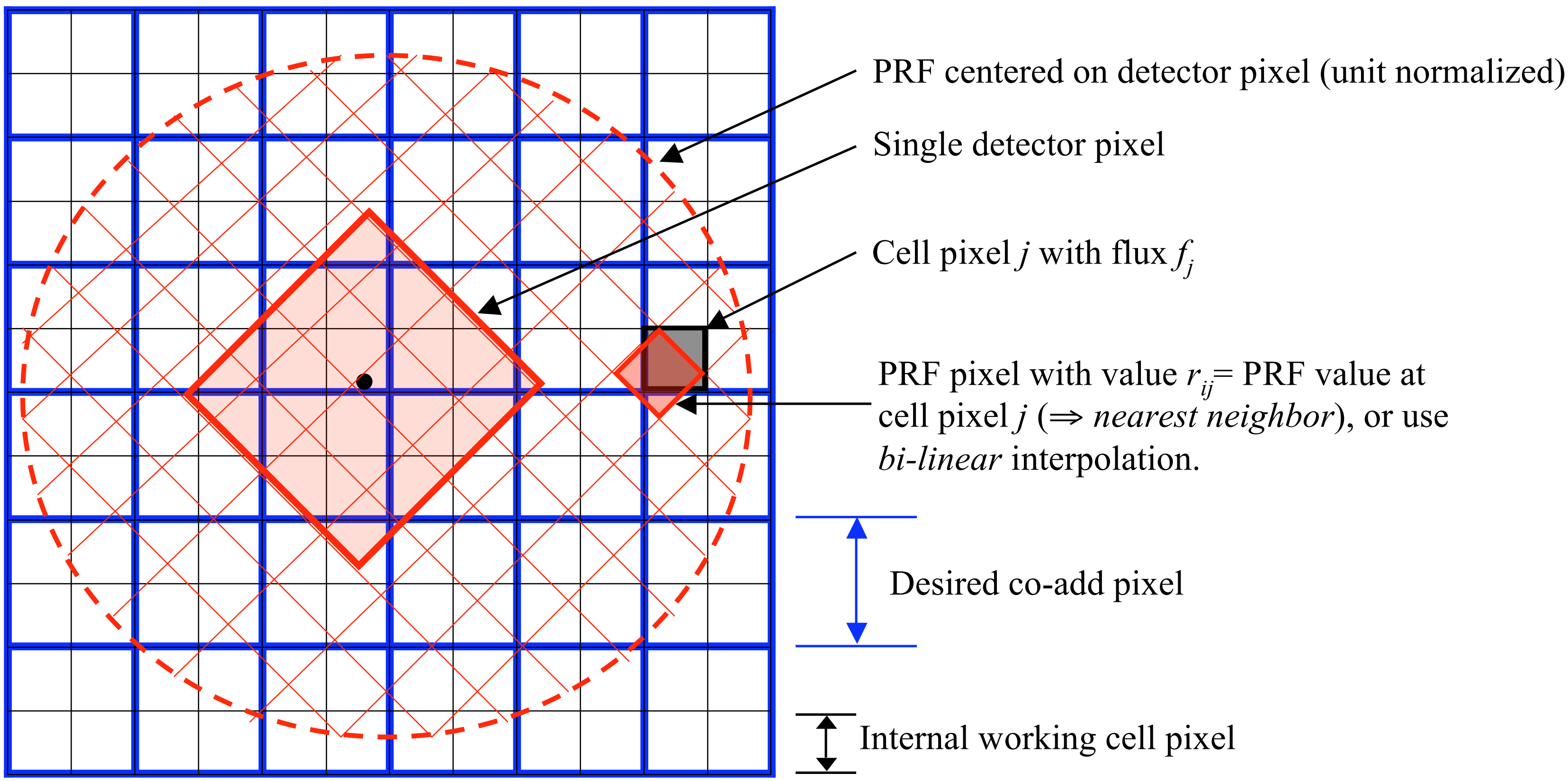
Input detector pixel

Desired co-add pixel
user-specifiable: co-add/input pix. scale factor

Internal working cell pixel
user-specifiable: cell/co-add pix. scale factor

- cell grid pixel size driven by accuracy to which a PRF can be positioned (have spatial and temporal variations).
- PRF (via PSF) is subject to thermal fluctuations in optical system.
- In addition, have a relative frame-to-frame registration uncertainty.

PRF Placement and Interpolation



- PRF is sampled on same scale as an internal cell pixel (offline calibration once cell size optimized).
- Flux from cell pixel contributing to measured flux in detector pixel $i = r_{ij} * f_i$
- At end, co-add values in cell grid are down-sampled to desired final co-add pixel scale.

Examples

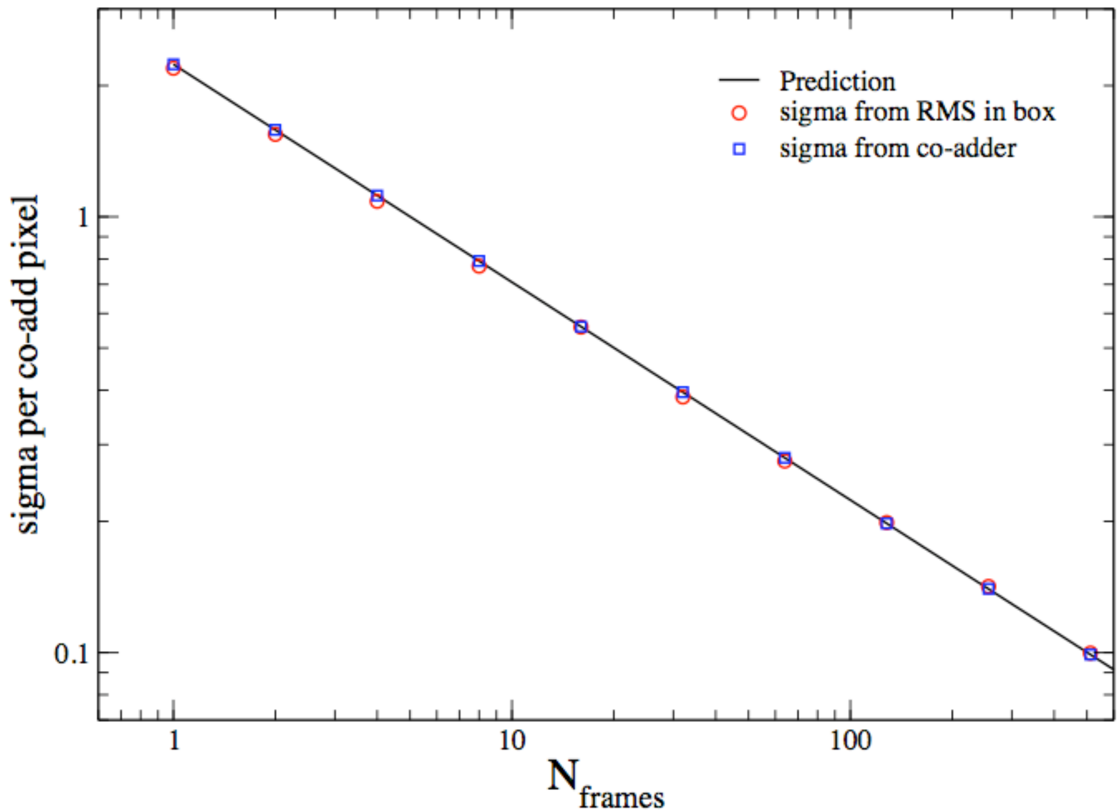
See:
http://web.ipac.caltech.edu/staff/fmasci/home/wise/awaic.html#EG

Focus on:

- Galaxy NGC 2403 with AWAIC and *Montage*
- *Spitzer*-IRAC observations of the North Ecliptic Pole region with AWAIC and MOPEX
- Impact of masked/missed pixels (e.g., saturated stellar cores)

Noise Characterization

- Co-add pixel noise scales as expected with the input number of frame overlaps (depth-of-coverage): $\propto N^{-1/2}$
- But, uncertainty in total flux measured in an aperture (or profile fit) off co-add must account for correlated noise.
  - See formalism outlined in SDS document.
  - Ignoring correlations would lead us to over-estimate Signal-to-Noise ratios => over-confident about our measurements!
  - Plan on providing a look-up table or graph of correction factors as a function of aperture size to correct for correlations.



To do List / Plans

- Explore, test and implement an outlier detection/flagging method.
- …a background matching method or import components from existing software.
- Test the above on simulated data that uses current undertanding of the WISE detectors.
- More accurate, faster interpolation methods when mapping PRF onto co-add grid. E.g., default to an internal analytic function?
- Selection of specific mask bits to flag against when reading bad-pixel masks. Not all flavors of "badness" will be fatal.
- Make specification of output co-add pixel sizes more generic (i.e., allow non-integral fractions of the input/native pixel scale to be specified).
- NaN out values in the co-add whose depth-of-coverage is below some TBD fractional value.
- Software to check consistency of computed/propagated uncertainties using χ^2 tests.
- Speed optimization, e.g., consider using FFTs for convolution operations.