Instrument Science Report WFC3 2016-12

Empirical Models for the WFC3/IR PSF

Jay Anderson
Mar 17, 2016

ABSTRACT

The severely undersampled nature of the WFC3/IR PSF introduces significant complications in the analysis of point sources and barely resolved objects. We have made use of observations of an outer field in Omega Centauri in order to construct PSFs in four wide-band WFC3/IR filters: F105W, F125W, F140W, and F160W. Since the F110W PSF is also popular, we used observations of the 47 Tuc calibration field to construct a model for it as well. This document describes the construction of these PSFs, describes their features, and gives a brief tutorial of how to use them to fit objects in images. A follow-on document will make use of the large archive of observations to come up with PSFs for many other WFC3/IR filters. In order to help the community make use of these PSFs, we are currently developing software that can be used to (1) measure stars in images with this PSF and (2) insert stars into flt images such that they can be realistically inserted into drizzled images.

1. INTRODUCTION

One of the challenges of reducing HST data is that the PSF (point-spread function) is generally undersampled by the detector pixels — sometimes severely. The designers of the instruments had to make a compromise between improving the sampling of the scene and covering more of the field. Because many programs do not require Nyquist sampling in order to succeed and because some mitigation for undersampling is possible via dithering, it was decided to cover more field and leave it to observation strategy or post-analysis software to deal with the scene degradation caused by the under-sampling.

Undersampling is not always problematic. When studying point sources, it is not critical to have a well-sampled detector. So long as the PSF is broad enough to spread the flux over a few pixels, an accurate

position and flux can be determined, provided one has access to an accurate PSF model. Spreading the flux over tens of pixels is overkill when there are only three parameters to measure (x/y position and flux). At the other extreme, astrophysical sources are often much larger than the pixels, such that the undersampled nature of the detector does not limit the analysis we can do on them.

That said, there are some astrophysical objects that are barely resolved, and these can indeed be hard to measure in undersampled images. Most faint galaxies are barely resolved by HST, and weak-lensing studies require exquisitely accurate models of the PSF in order to infer a slight preference across the sky for a particular elongation.

Dithering is one way to mitigate the effects of undersampling. By placing each part of the undersampled scene at a variety of locations with respect to the pixel grid, we are able to recover information that is lost in the fat pixels of a single exposure. Unfortunately, recovering and using this information is far from trivial. There are three basic ways to go about it.

(1) One could make use of an accurate point-spread function and fit the pixels of an object in each individual exposure for the parameters of interest (position, flux, shape, etc.). The parameters extracted from each exposure are then combined for analysis. This provides automatic crosschecks and an empirical estimate of errors to help understand how well various parameters have been measured in the mean. This approach works particularly well for point sources, since the particular location of the target with respect to the pixel grid in a particular exposure does not have a large effect on how well we can measure its position and flux.

(2) One could use the forward and reverse distortion solutions and the inter-image transformations to map each pixel in each exposure into the reference frame so that we can treat all the many pixel constraints in a single fit for the parameters of interest. The fact that the pixels come from a dithered set means that they are able to probe structure that is not accessible in any single exposure. This is probably the most accurate way to measure fine structure in the scene, since each pixel is maintained as the direct constraint that it is rather than being combined into an intermediate product.

(3) Finally, it is possible to distill the many observations into a single stack in order to come up with a super-resolved image of the scene that was delivered to the telescope. Unfortunately, there is no single way to reconstruct images that have been degraded by undersampling. One reason for the non-uniqueness is that there is no unique output frame (i.e., plate scale, orientation, etc.). The output frame we choose depends on the number and quality of the dither samplings that we have available and perhaps on the analysis tools we hope to apply. But even if we take as given the specification of the output frame, there is no unique way to go from multiple independent observations of the scene to this output frame. Lauer (1999a) recommends a Fourier approach. Anderson (2014) has explored iterative least-squares-type approaches.

`AstroDrizzle` is the most commonly used software package for constructing higher-resolution images from dithered sets of HST data. It does rigorously perserve flux while at the same time distilling the oversized observed pixels into smaller output pixels in an effort to improve and regularize the sampling somewhat. The sampling achieved by `AstroDrizzle` is a complicated function of the output pixel scale, the drop-size, the specific dithered samplings for each output pixel,

and possible errors in the distortion solution and pixel mapping. The fact that the input pixels do not map in a simple, regular way onto the output pixels makes it difficult to do point-source-related analysis on stacked `AstroDrizzle` products. Such analyses can involve decomposing a galaxy into resolved or unresolved core or halo components, measuring positions for point sources relative to other features in the field, etc.

All three of the above approaches make use of dithered exposures to mitigate undersampling and would benefit from an accurate model of the PSF. The first approach (that of fitting to each exposure separately) is the most explicitly dependent on a PSF model, but all of them would benefit from an understanding of what unresolved sources look like to the detector pixels. The second approach could make use of the PSF in an effort to come up with a smooth deconvolved model to represent the PSF-convolved scene that is sampled by every individual pixel constraint. Even the `AstroDrizzle` approach benefits from an appreciation of what point sources look like through the stacking and resampling process, so as to prevent the tops of stars from being clipped in the drizzling process.

For all these reasons, we are endeavoring to provide accurate PSFs for the WFC3/IR detector to the community. While these PSFs will not enable science directly in most cases, they are a critical step needed to take high-precision analysis to the next level. In the first approach discussed above, PSFs can be used immediately. In the second approach, they must be combined with transformations and analysis software (see Anderson 2014 for an example). In the stacked approach (such as `AstroDrizzle` or other such tools), having access to a PSF can allow stars to be seeded into the raw exposures so that the impact of the stacking procedure on point sources can be assessed directly. This can be particularly useful for astronomical scenes that have a dearth of point sources but many marginally resolved objects.

This document will describe the procedure we used to extract PSFs for five IR filters (F105W, F110W, F125W, F140W, and F160W) from dithered observations of outer fields in Omega Centauri and 47 Tuc. We describe the observations in §2. In Section 3 we describe the "effective" PSF concept and how PSFs can be "observed" in image pixels and in turn used to analyze the pixels. In Section 4 we start with F105W and describe the iterative process that we used to extract and normalize the PSFs. We show the efficacy of the PSFs by comparing positions and fluxes measured with centroids and simple apertures against those determined by PSF fitting. In Section 5, we use our PSFs to explore the properties of the detector pixels. In Section 6 we repeat the operation for the other four filters. Section 7 documents a "standard" format that we have devised for storing our table-based PSFs. Finally, Section 8 summarizes the results and points to software programs that will shortly become available to make use of the PSFs. We provide some basic software in the Appendices B, C, and D.

This document is the first in a series. It will focus exclusively on the PSFs we extract for the four filters used in the Frontier Field program (F105W, F125W, F140W, and F160W) and another commonly used filter, F110W. The second report in the series will provide a software routine that will add artificial stars into `flt` exposures in consistent positions so that the images can be combined using standard `AstroDrizzle` products. The third report in the series will explore the WFC3/IR archive to evaluate how representative these PSF models are for different focus levels. Finally, a fourth report will extend the number of PSFs available to include filters that have enough observed point sources in the archive.

2. OBSERVATIONS

Calibration program CAL-13606 was executed in late 2013 in order to allow the instrument teams to construct PSFs for the filters used in the Frontier Fields project: F105W, F125W, F140W, and F160W for WFC3/IR and F435W, F606W, and F814W for ACS/WFC. The target is the globular cluster Omega Centauri. We pointed the ACS/WFC detector at the center of the cluster, where the star density is ideal for that detector; WFC3/IR observed in parallel a field that was 6′ arcminutes from the center, thus providing a good density of stars for that detector. In this report, we focus on the WFC3/IR observations and reductions.

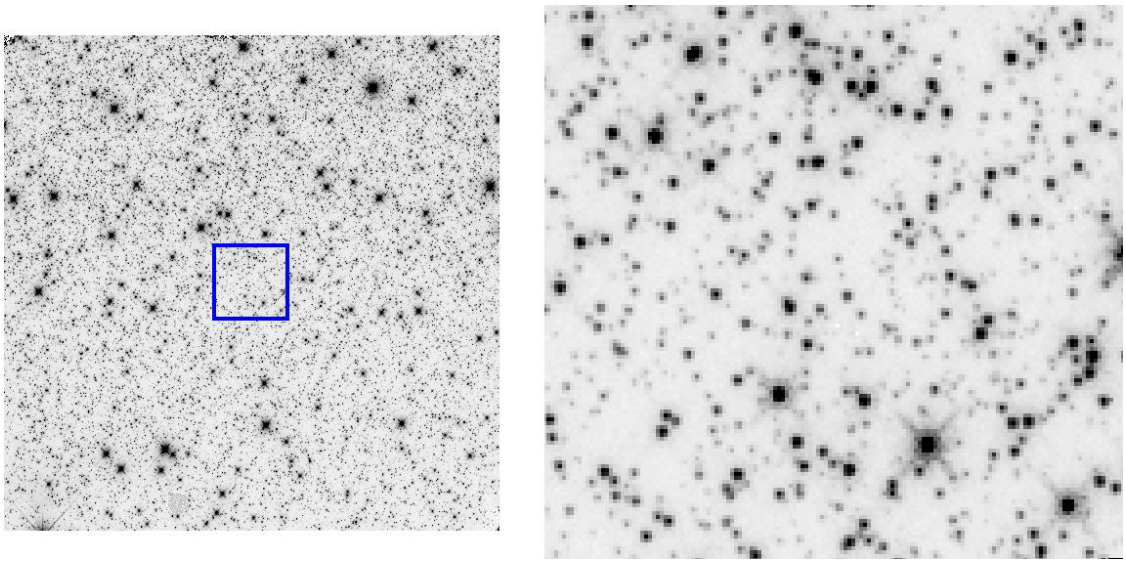

Figure 1: The first image in the F105W set. The left shows the entire image, and the right a ×8 zoom of the region inside the blue box. The log stretch does not do justice to the extreme undersampling suffered by the point sources.

The data for each of the four filters consists of eight 102-s exposures dithered around the edges of a square with a half-width of about 150 pixels. Figure 1 shows the first F105W exposure (`ichi01a1q_flt.fits`). The full exposure frame is on the left, and a ×8 zoom of the blue square is on the right. These images make it clear that there are many bright and isolated point sources that can be used to extract information about the PSF. There are approximately 10,000 detections in each exposure, but only about 1,600 of them are sufficiently bright and isolated to help us solve for the PSF.

3. THE EFFECTIVE PSF MODEL

The model we use for the PSF is essentially the same model developed by Anderson & King in their seminal PASP paper in 2000 (AK00). The main insight in that paper is the distinction between the "instrumental" PSF and what they refer to as the "effective" PSF.

3.1 A TALE OF TWO PSFS

The "instrumental" PSF describes the probability distribution of where photons from a given point source will land at a particular location on the detector. This is what ray-tracing software and sophisticated optics programs provide. While this is of interest to those who optimize telescopes, it is not directly relevant for those of us who model images, since this "instrumental" PSF is always integrated over detector pixels before we see it. At the same time, whenever we need to use the PSF to fit stars, we similarly find that we only need the PSF in pixel-integrated form.

AK00 made the realization that the "instrumental" PSF is an abstraction that we need not concern ourselves with directly: there is never a need to model the "instrumental" PSF; instead we can focus on what they call the "effective" PSF, which is defined to be the instrumental PSF convolved with the profile of a pixel: $\psi_E \equiv \psi_I \otimes \Pi_{DET}$. This realization greatly simplifies the PSF construction and evaluation process.

Since ψ_I and Π_{DET} are both smooth, continuous functions, the "effective" PSF should also be a smooth, continuous function. The instrumental PSF ψ_I tells us the probability for a photon to be recorded at a certain location (Δx, Δy) relative to a star's center in terms of a surface brightness. The effective PSF ψ_E tells us something much more practical: it tells us the probability that a photon will be detected in a given pixel at a particular offset relative to the center of a star. If a star is centered at (x_{star},y_{star}), then pixel [i, j] is offset by

$$(\Delta x, \Delta y) = (i - x_{star}, j - y_{star}) \qquad (E1)$$

with respect to the center of the star, and $\psi_E(\Delta x, \Delta y)$ should tell us what fraction of the star's light will fall in that pixel. Thus, ψ_E is a smooth, continuous two-dimensional function with a domain of (Δx, Δy).

3.2 WAYS TO USE AND EXTRACT THE EFFECTIVE PSF

If we want to model the distribution of light in the vicinity of a point source, then we can model the flux in pixel at [i,j] as:

$$P_{ij} = s_{star} + z_{star} \cdot \psi_E(i - x_{star}, j - y_{star}). \qquad (E2)$$

If we consider all the pixels in the vicinity of a point source and plot P_{ij} as a function of ψ_E, then we see that this is simply the equation of a line with a slope of z_{star} (the star's flux) and an intercept at s_{star} (the background sky value): $P_{ij} = s_{star} + z_{star} \cdot \psi_{ij}$, where $\psi_{ij} \equiv \psi_E(i\text{-}x_{star}, j\text{-}y_{star})$

This equation implies a simple linear relationship between a star's flux and the values of its pixels when we know the location of the star. It is trivial to solve for the flux and sky in this case. But when we do

not know the location of the star, the argument to ψ_E changes, and a non-linear fit is required in order to solve for (x_{star}, y_{star}, z_{star}, s_{star}). Note how easy it is to use the effective PSF in modeling stars. We do not have to integrate it over pixels before we can use it; we simply evaluate it at a set of single points in its domain.

The above equation shows us how to use the effective PSF formalism to fit profiles of stars. But it can also be inverted to let us measure the effective PSF when we know (or have estimates for) a star's parameters. In this case, every pixel in a star's image provides an estimate of the PSF at one point in its domain:

$$\psi_E(\Delta x, \Delta y) = \frac{(P_{ij} - s_{star})}{z_{star}}, \qquad (E3)$$

with (Δx, Δy) coming from ($i-x_{star}$, $j-y_{star}$) in Equation E1. Each star then provides an array of point samplings of the effective PSF, one for each pixel, with the samplings spaced in a grid with one-image-pixel spacing. If we have many stars centered at different pixel phases, it is clear that we have many point-samplings of ψ_E and can distill these samplings into a representative model of the PSF. These concepts are fleshed out more thoroughly with illustrations in Sections 3 and 4 of AK00.

3.3 Representing the Effective PSF

The above discussion makes use of the effective PSF, $\psi_E(\Delta x, \Delta y)$, as a simple two-dimensional function without reference to how it is evaluated or specified. Perhaps the only thing that is *not* easier about using the "effective" rather than the "instrumental" PSF is that sometimes with instrumental PSFs, one can be justified in using an analytical formulation, such as a Gaussian, Lorentzian, or a Moffat profile in its description, since simple telescope optics can often produce such profiles in the image plane. However, telescopes with complicated optics, such as HST, do not lend themselves to simple analytical approximations, so some kind of empirical approach is necessary, and since the model must be done empirically there is no benefit to focusing on the "instrumental" PSF.

Whatever approach we take, it will have to be able to return a value for the function ψ_E given any input offset (Δx, Δy) in the domain. That is the basic role of a PSF: to tell us what fraction of light lands where relative to a point source. We expect that the function will also be smooth and continuous.

The simplest way to specify such a 2-D function is in tabular form. We adopt a set of evenly spaced fiducial points in its (Δx, Δy) domain and specify the value of the function at these points and use some kind of interpolation in between the points. The fiducial points will have to be finely enough spaced to represent the full variation of the function, but not so finely spaced that the individual points will be hard to determine independently. Clearly, since the PSF is undersampled, these fiducial points will have to be specified more finely than one node per image pixel. In practice, we find that ×4 supersampling generally works well. It is hard to specify so many points completely independently, so we impose some additional constraints to ensure smooth and continuous behavior.

There is no hard-and-fast rule about how much smoothness to impose on the PSF gridpoints. When too much is imposed, the PSF is not able to represent the function at its sharpest points (i.e., the peak), and

when too little is imposed, the PSF can contain non-physical high-frequency variations. The benefit of the smoothing step is that we do not have to fine-tune the final grid size and spacing to match the detector perfectly in order to ensure that we have not over-parameterized the problem: the same grid can represent PSFs with very different structure scales. It is worth noting that in all of the instruments that have been treated with this formalism (WFPC2, ACS/WFC, ACS/HRC, WFC3/UVIS and WFC3/IR), we have yet to encounter a PSF that could not be modeled with the same ×4 supersampled function plus smoothness constraints judiciously tailored for each camera. This means that the PSFs we get out are all in the same format and can be used by the same star-fitting subroutines, even though the detectors and cameras are extremely different. This has great benefit, since we do not have to construct new measuring software for every new detector.

3.4 THE NEED FOR A VIRTUOUS CYCLE

Now that we have established the effective PSF formalism, we are ready to start examining the WFC3/IR PSFs. We will begin with the F105W data, since it is the most undersampled of the popular filters.

In §3.2, Equation E3 tells us how each pixel in each star image can be used to tell us the value of the effective PSF at one point, provided that we know the position of the star and its total flux and have an estimate for the background-sky level. Of course, we do not start out with extremely accurate positions or fluxes for stars; such measurements require an accurate PSF with which we fit the profiles. Thus, we can already see that arriving at an accurate PSF will be an iterative process that will involve using positions and fluxes to solve for the PSF and using the PSF to solve for positions, all in a (hopefully) virtuous cycle.

One challenge, however, noted by AK00, is that there is a fundamental degeneracy in the problem. It can be hard to measure systematically accurate positions for stars in undersampled images without already having a good PSF. If our estimate of the PSF is sharper or broader than the true PSF is, then this error will introduce what we call "pixel phase" bias into the measurements[1]. And if the positions we use to construct the PSF are biased in some way, then they will produce a similarly biased PSF, and the positions that we in turn measure from this PSF will reflect the same "pixel-phase" bias.

The key to breaking this degeneracy lies in using dithered data. In a dithered set of images, a given star will land in a variety of locations with respect to the pixel boundaries, and each of these positions will have a different "pixel-phase" bias. As such, the star's average position in a common reference frame will be less biased than the positions measured in the individual frames. We can transform this average position back into the frame of each exposure to arrive at better estimates for the true position of the star in that particular frame. By using this position in our PSF-determination procedure, we can arrive at a more accurate PSF and continue the virtuous cycle.

The cycle thus has three stages: (1) measure raw positions in a set of dithered images, (2) find an average position in the reference frame, and transform this position back into the frame of each individual

[1] Since centroid-type positions make simple assumptions about the PSF, then they often suffer greatly from this bias.

exposure, and (3) use these new positions to determine an accurate PSF. This new PSF can then provide new raw measurements for the first step of the next iteration. The second stage can be complicated when there is distortion in the detector (as there is in every HST detector). We will make use of the forward and reverse distortion solutions that we developed for other projects to effect these transformations, but it is beyond the scope of this effort to delve into the details of the distortion solutions here.

We begin the PSF construction procedure by going through each `_flt` image pixel by pixel and identifying as a possible star every pixel that (1) is brighter than any other pixels within a radius of 7 pixels, (2) has more than 2500 electrons per second in its brightest 2×2 pixels and (3) has fewer than 80,000 electrons before the first read(which would constitute saturation). We measure a position for each star using the centroid (described below) and measure an aperture-based flux using the pixels within a radius of 3.5 pixels. We cross-match the star lists from the eight individual exposures with that from the first exposure and find the transformation from each exposure into the frame of the first exposure (taking distortion into account, naturally). We collate together the star lists in the frame of the first exposure, identifying a usable star wherever five out of the eight exposures find a common source. For the F105W dataset, we found 1037 such common sources. Finally, we determine an average position for the star in the reference frame and transform that average position back into the frame of each exposure, using the inverse distortion solutions. We now have a position and a flux for each star in each individual exposure and can start to examine the PSF.

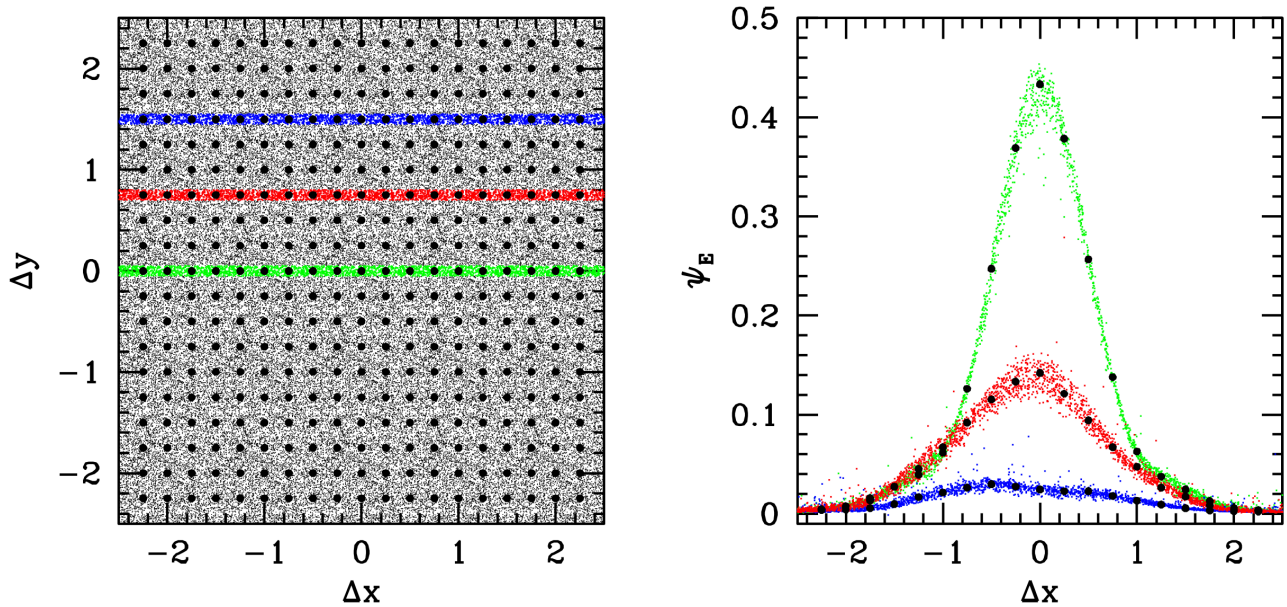

Figure 2: Left: locations of the PSF samplings. Each small point represents a single pixel in a single exposure that tells us what fraction of the star's total light landed in it. Each PSF estimate is plotted according to where that pixel was located with respect to the star's center.. The green, red, and blue strips represent the samplings that land along Δy~constant zones. Right: the PSF estimates along these three strips. It is clear that the effective PSF is a smooth, continuous function.

3.5 EXAMINING THE PSF

Using the positions and fluxes determined above, we use Equation E3 to turn each pixel in the vicinity of each star in each exposure into a point-estimate for the PSF at location $(\Delta x, \Delta y) = (i\text{-}x_{\text{star}}, j\text{-}y_{\text{star}})$. The left panel of Figure 2 shows the locations of all the estimates within the inner 5×5 pixels of the PSF. There are 1037 stars in each of 8 exposures, thus we have about 8000 estimates per 1×1-pixel region (not all stars are found in all exposures, since the dither shifts some stars off the edge of some images).

The green points correspond to the estimates with $|\Delta y| < 0.10$, the red points correspond to those with $|\Delta y - 0.75| < 0.10$, and the blue points correspond to $|\Delta y - 1.5| < 0.10$. The panel on the right shows the raw PSF estimates as a function of Δx along these slices.

The black points on the left of Figure 2 show the fiducial locations every 0.25-pixel in Δx and Δy where we tabulate the function in order to represent it. These same points are shown on the right with the average value of the PSF plotted across each slice.

There is some spread about the averages. Some of this spread is due to the non-negligible height of the Δy bin and the variation of the true PSF within this bin. Some of the spread is due to Poisson-noise errors in the pixels that supply the samplings and positional errors in placement of the samplings or flux errors in determining the samplings. But some of the variation can be traced to the fact that the PSF is not constant across the detector, since we are showing PSF samples all over the detector.

To look for PSF variations with position across the detector, in Figure 3, we divide the image into 3×3 regions and examine the central slice of the PSF for each region. It is clear that the PSF does change shape somewhat across the detector, having more flux in the central pixel (43%) on the right of the detector and less flux (39%) on the left of the detector. Note that the empirical nature of the effective PSF tells us immediately what fraction of a star's flux lands where.

The WFC3/IR detector is indeed undersampled. A pure Gaussian that is Nyquist sampled will have about 20% of its flux in the central pixel. The ACS/WFC detectors have about 22% of the light in their central pixels and the WFC3/UVIS detectors have about 18%. Even though these fractions are close to the Nyquist value, the detectors should still be considered undersampled because there is a significant fraction of the flux in an extended halo, so that the core must be sharper than Nyquist to get such a large fraction in the central pixel.

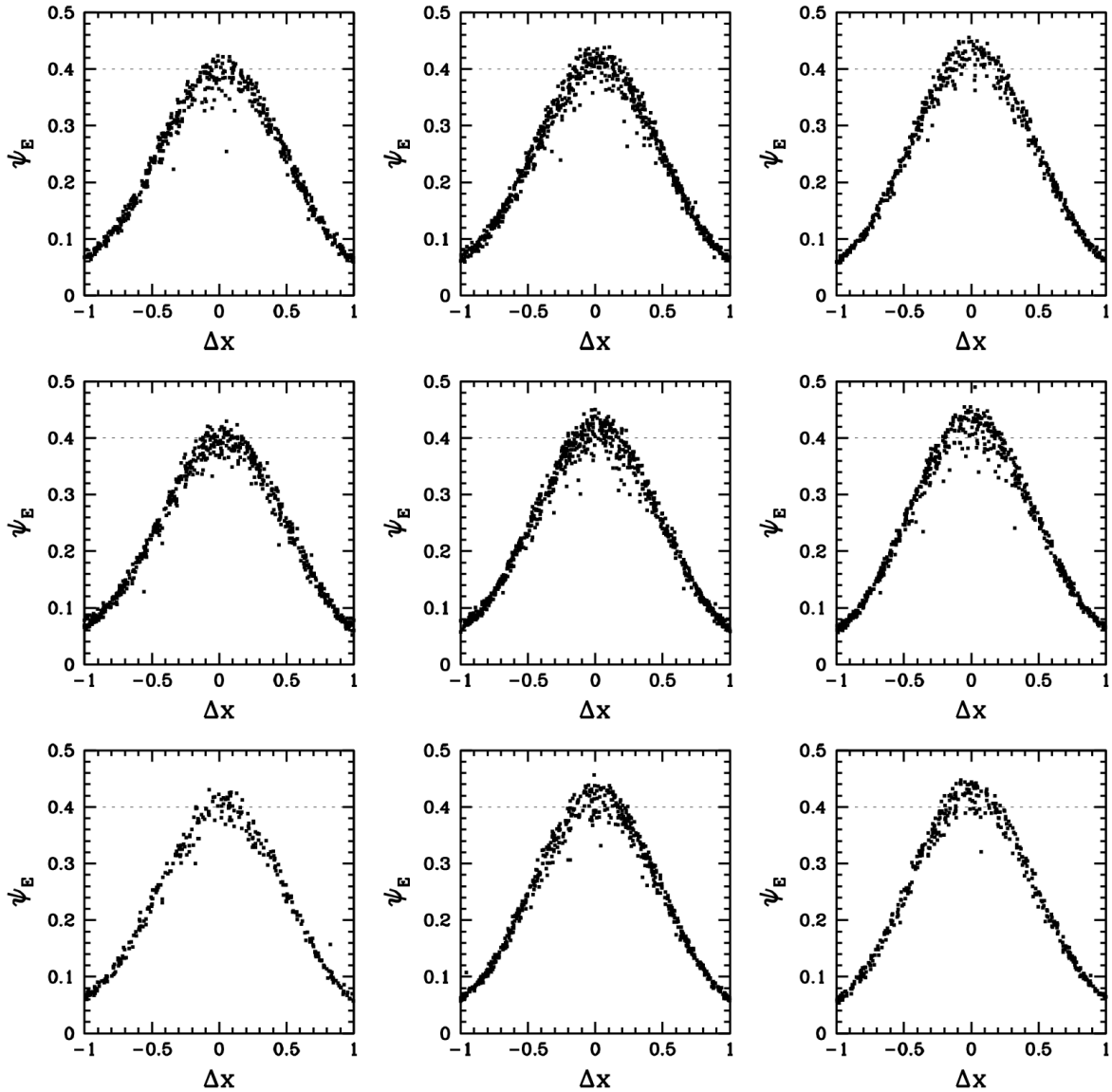

Figure 3: The 1014×1014 WFC3/IR chip has been divided into thirds in x and y and we show the central Δy ~ 0 slice of the PSF for the 9 different regions. The black dots show the samplings from the center panel and the green dots show the samplings from the local panel. It is clear that the PSF has slightly less flux at the center on the left of the chip than on the right.

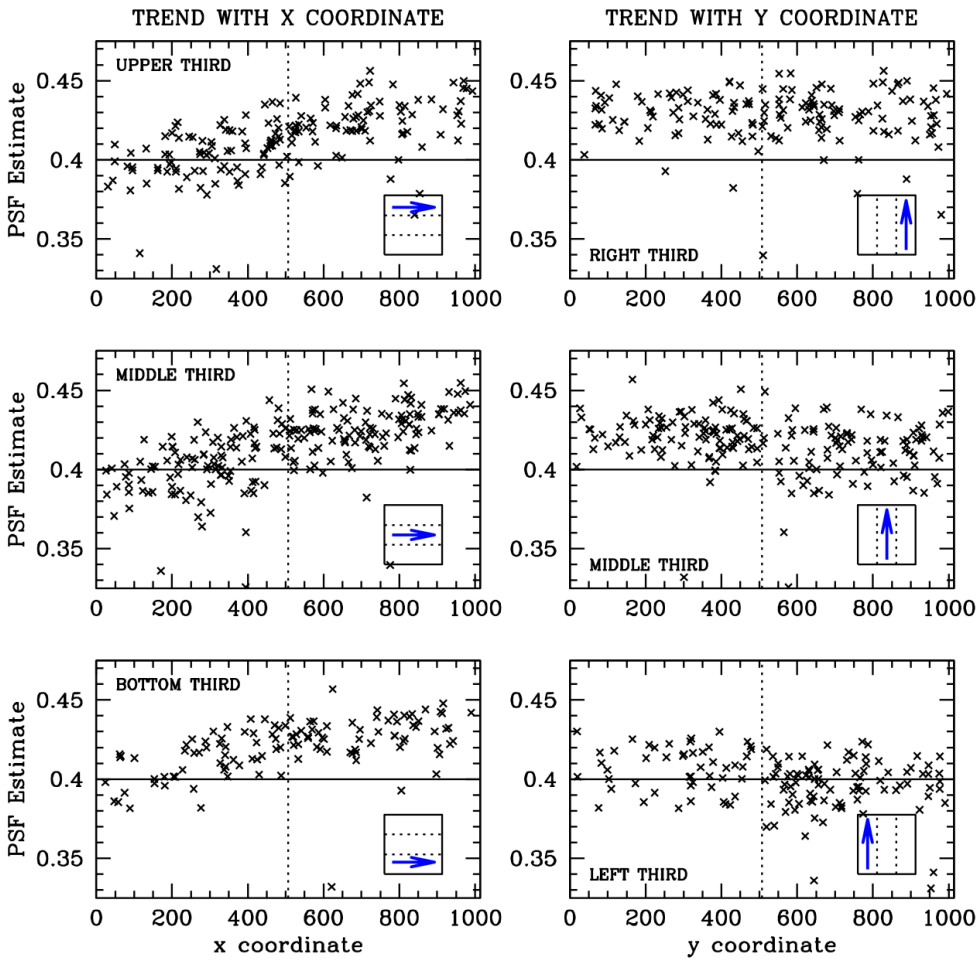

Figure 4: This shows the estimates for the central point of the PSF (|Δx|<0.15 and |Δy|<0.15) as a function of location on the detector. The trends with x (in the left panels) are stronger than those in y.

Figure 4 shows the trends of the central value of the PSF as a function of position on the detector. There is a strong trend with *x* coordinate and less of a trend with *y* coordinate. The trend with *x* appears to be largely linear, but the slope does change over the detector, therefore it is prudent to adopt a set of 9 fiducial PSFs in a 3×3 array: one at the center of the chip and four at the edges and four at the corners. Although the central flux does not change much with *y*, the non-core parts of the PSF *do* vary with both *x* and *y*, so it is prudent to allow PSF variability in both dimensions[2]. The 3×3 model we adopt for the WFC3/IR detector is analogous to the one that was adopted for WFPC2 in AK00.

---

[2] It is worth considering whether the trends of sharpness with *x* coordinate that we see above could be related to the fact that on account of distortion, the size of a projected pixel onto the sky changes and could well intercept more flux when it intercepts more of the telescope beam. It turns out that the pixel-area map (see Figure B.4 in the instrument handbook) has most of its variation in the vertical direction, so the variation of core flux cannot be directly due to that geometric effect.

If we consider a spatially variable PSF, then the PSF is really a four-dimensional function: $\psi_E = \psi_E(\Delta x, \Delta y; i, j)$. In the next section we discuss the detailed parameterization and evaluation.

3.6 PARAMETRIZING THE PSF MODEL

In the previous section, we examined the PSF's behavior, both within its (Δx, Δy) domain and spatially across the detector (i, j). We have discussed in general how we will parameterize the PSF, but here we get into the specifics.

Since the "effective" PSF is a function that does not have a simple analytical form, we parameterize it with a simple look-up table. In undersampled detectors, the PSF can have considerable structure on smaller-than-pixel scales, so our table should be sampled sufficiently finely to deal with that. Our PSF is super-sampled by a factor of 4 in each dimension, such that there are 4×4 = 16 grid-points for every region the size of an image pixel. Even this fine sampling is too sparse to allow simple bi-linear interpolation, so we must use something higher order. We adopt a simple bi-cubic spline to interpolate the grid in-between gridpoints. There are several ways to code up the bi-cubic spline, but the few that we have tried all give similarly good results. The important thing is to allow curvature between the points.

We chose to model only the inner 12.5 pixels of the PSF. Going any further out would require us to include saturated stars in the modeling, since only saturated stars have enough signal beyond 10 pixels to constrain the PSF. It would be difficult to include saturated stars in the modeling, since good positions and fluxes (required for PSF modeling) are complicated to measure for them. Furthermore, since saturated stars extend farther out, we cannot afford to have many in each exposure lest their profiles overlap in a tangled mess. Moreover, the main obvious features of saturated stars are the diffraction spikes, which change orientation with position on the chip, on account of geometric distortion. Our linearly interpolated model of fiducial PSFs would not be a good way to deal with this. Thankfully, the stars we can model well are all unsaturated, and these stars have almost all of their flux within the bounds of our PSF.

So, given the size and the ×4 oversampling, our PSF model consists of an array of 101×101 gridpoints, with the center of the PSF at [51,51]. The PSF at a given location in the image is given simply as a two-dimensional table of numbers. To account for the spatial variability, we have a 3×3 array of PSFs across the detector, so the full WFC3/IR PSF has the dimensions of 101×101×3×3. The first two dimensions are for (Δx, Δy), the domain of the PSF, and the last two are for the spatial variation.

To evaluate the PSF at a particular location in the image, we first determine where the star is with respect to the fiducial locations and linearly interpolate. For WFC3/IR, the fiducial locations 1, 2, and 3 in x are located at $x = 0$, $x = 507$, and $x = 1014$, respectively, with a similar specification for y. The quantity $r_x = 1 + (x-0)/507$ gives the index location in x for the fiducial PSF, with a similar formulation for y. If

we define $n_x = \mathrm{int}(r_x)$ and $f_x = r_x - n_x$, then we find that the PSF at (x,y) is just:

$$\begin{aligned}\Psi_{xy}[i,j] = {} & (1-f_x)(1-f_y)\,\Psi\left[\mathrm{i},\mathrm{j};n_x\quad,n_y\quad\right] \\ & +(\ f_x\ )(1-f_y)\,\Psi\left[\mathrm{i},\mathrm{j};n_x+1,n_y\quad\right] \\ & +(1-f_x)(\ f_y\ )\,\Psi\left[\mathrm{i},\mathrm{j};n_x\quad,n_y+1\right] \\ & +(\ f_x\ )(\ f_y\ )\,\Psi\left[\mathrm{i},\mathrm{j};n_x+1,n_y+1\right].\end{aligned} \tag{E5}$$

Here, the capital Ψ refers to the PSF grid, not the continuous function that is represented by the grid (ψ). If we seek to evaluate the PSF for a given offset $(\Delta x,\Delta y)$, then we can simply evaluate the PSF for that offset for the neighboring fiducial PSFs and compute:

$$\begin{aligned}\psi_{xy}(\Delta x,\Delta y) = {} & (1-f_x)(1-f_y)\,\psi\left(\Delta\mathrm{x},\Delta\mathrm{y};n_x\quad,n_y\quad\right) \\ & +(\ f_x\ )(1-f_y)\,\psi\left(\Delta\mathrm{x},\Delta\mathrm{y};n_x\quad,n_y\quad\right) \\ & +(1-f_x)(\ f_y\ )\,\psi\left(\Delta\mathrm{x},\Delta\mathrm{y};n_x\quad,n_y+1\right) \\ & +(\ f_x\ )(\ f_y\ )\,\psi\left(\Delta\mathrm{x},\Delta\mathrm{y};n_x+1,n_y+1\right).\end{aligned} \tag{E6}$$

Now that we have a table to represent the PSF for a given star, we can use it to determine what fraction of a star's light should land in a particular pixel relative to a star's center. If we have pixel [i, j] and a star centered at $(x_{\mathrm{star}},y_{\mathrm{star}})$, then $\Delta x = i - x_{\mathrm{star}}$ and $\Delta y = j - y_{\mathrm{star}}$, and we need to bicubic-ly interpolate the 101×101 PSF grid at location (r_x, r_y), where $r_x = 51 + 4\times\Delta x$ and $r_y = 51 + 4\times\Delta y$ to evaluate the fraction of light that should fall in that pixel.

4. SOLVING FOR THE PSF

The previous section provided the basis for the PSF model and also showed heuristically how to estimate the PSF from the pixels of star images. It also described the virtuous cycle needed to arrive at a systematically accurate PSF. Here we will show the specific steps that allow us to obtain the PSF.

The virtuous cycle has three steps. Step 1 is the determination of good positions for stars in individual exposures. Step 2 is the averaging process wherein we use the multiple dithered exposures to improve these positions. Finally Step 3 is the extraction of the PSF from the multiple exposures given our current estimates of the flux and position for each star in each one.

4.1 FIRST CYCLE, STEP 1: FITTING STELLAR IMAGES WITH NO PSF

In the very first iteration there is no PSF model available, so we must have a way to determine positions without a PSF. For this we adopt a simple centroid procedure that is designed to work robustly for undersampled data. If pixel $P_{i,j}$ is a local maximum, then we can estimate the position of a source with the following equations:

$$x = i + \frac{1}{2}\frac{P_{i+1,j} - P_{i-1,j}}{P_{i,j} - \min\left(P_{i+1,j}, P_{i-1,j}\right)} \qquad (E4a)$$

$$y = j + \frac{1}{2}\frac{P_{i,j+1} - P_{i,j-1}}{P_{i,j} - \min\left(P_{i,j+1}, P_{i,j-1}\right)}. \qquad (E4b)$$

These equations have the sensible property that if the two pixels on either side of a maximum are the same, it finds the center to be in the middle, and if the central pixel is equal to the adjacent one, then the center is found to be exactly in between the two. As for the flux, we set the flux to be the aperture flux over sky within a radius of 3.5 pixels; this aperture is large enough to be relatively insensitive to the pixel-phase of the star.

4.2 FIRST ITERATION, STEP 2: COMBINING THE INDEPENDENT MEASUREMENTS

The part of the process that allows us to converge on an unbiased PSF model is the combination of data from multiple dithers. As discussed above, the position (and even the flux) that we measure in an individual exposure can be biased if we do not use an accurate PSF model. If we in turn use those positions to re-derive the PSF, then we are just reinforcing these biases. Extra information is needed; and it comes from combining measurements from multiple dithered observations, which place each star at different locations with respect to the pixel grid.

4.2.1 THE NEED FOR A DISTORTION SOLUTION

Conceptually, the procedure is simple, but the complication comes from the fact that we cannot combine positions from different exposures without taking into account distortion. In order to maximize throughput and have as few optical elements in the beam as possible, the designers of HST's cameras opted to accept considerable distortion across the field. HST's square detectors project to parallelograms

on the plane of the sky. This is a linear distortion, but there are considerable non-linear distortions as well. These distortions have been characterized in terms of coefficients that allow us to map pixels to the V2-V3 focal plane; these coefficients are provided in the IDCTAB reference files. Here we do not require such a absolutely-calibrated solution, but merely need the intra-detector solution.

4.2.2 A CONVENIENT DISTORTION SOLUTION: THE STDGDC FILE

The solution we use here was developed for the purposes of high-precision astrometry independently of the official Institute solution. Our solution maps the location of each pixel to its geometrically correct location relative to the middle pixel of the detector. The solution is based on polynomials, but for convenience we have distilled it into images that are the same size of the detector. See Appendix A for a discussion and a hyperlink.

4.2.3 COMBINING THE POSITIONS

Now that we are able to account properly for distortion, we can improve the positions that we will use to extract the PSF. To do this, we transform the positions that were measured in the individual exposures in §4.1 into the reference frame (which here is simply the distortion-corrected frame for the first exposure). This gives us between 5 and 8 estimates for the position of each star in the reference frame. We next determine an average position for each star and an RMS about that average.

The left column of panels in Figure 5 shows from top to bottom the RMS of the residuals in x_{ref} and y_{ref} and m_{ref} (the instrumental magnitude) for the first iteration. The positional residuals are quite large (0.07 pixel or so), but those for the magnitude are quite low (about 0.01 magnitude, or 1%). The reason for the large positional residuals can be seen in the right panels: our initial positions contain significant biases related to the star's pixel phase. These crude positions were derived without access to any PSF model and we had to use a simple centroid algorithm, which is only truly accurate for a top-hat-shaped PSF, since the algorithm we used (like *all* centroid algorithms) presumes a linear transfer of flux with position. Systematic errors in astrometry as a function of pixel phase are typical of such simplistic assumptions, since the true PSF induces a clearly non-linear sloshing of flux from pixel to pixel as a star is moved across the pixel grid.

Once we have an average position for each star in the reference frame, we can use the inverse transformations and the inverse distortion solutions to map the average position back into each individual frame. It is worth noting here that the transformations we use to go from the distortion-corrected frame of each exposure into the reference frame (which is, naturally, also distortion-corrected) are simple 6-parameter linear transformations:

$$u_T = U_0 + A(x_{\mathrm{c}} - X_0) + B(y_{\mathrm{c}} - Y_0) \quad (E5a)$$

$$v_T = V_0 + C(x_{\mathrm{c}} - X_0) + D(y_{\mathrm{c}} - Y_0) \quad (E5b).$$

Here, A, B, C, and D are the linear terms, and X_0, Y_0, U_0, and V_0 are the constant-offset terms. These transformation parameters are specified by a least-squares procedure based on the positions of

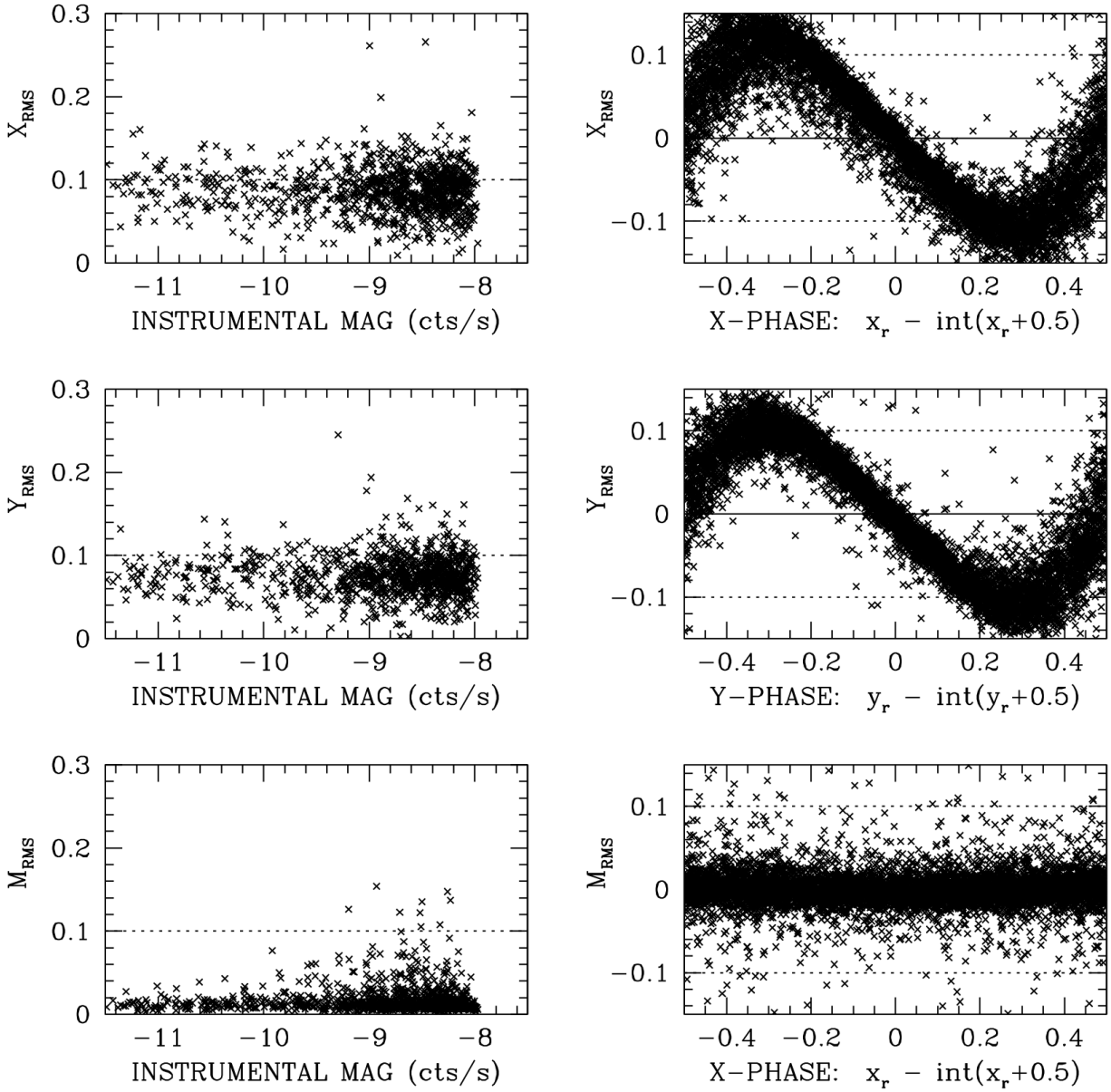

Figure 5: The left panels show the RMS measurement errors from the first iteration, where the positions are centroid positions and the fluxes are simple aperture fluxes. The position residuals are typically 0.075 pixel RMS and the magnitude residuals are typically 0.01 magnitude RMS. In the right panels, we examine the residuals as a function of pixel phase and find significant bias in the astrometric measurments, but not in the photometric measurments.

cross-identified stars in the reference frame and in each individual frame. It may look like there are eight parameters to solve for here (U_0, V_0, X_0, Y_0, and A, B, C and D), but one of the offset terms is arbitrary, so there are really only 6 free parameters to solve for. It turns out that if we take (U_0, V_0) to be the centroid of the cross-identified stars in the reference frame, then (X_0, Y_0) is simply the centroid of the same stars in the individual distortion-corrected frame: $X_0 = \bar{x} = \frac{1}{Ns}\sum_1^{Ns} x_n$ etc. The linear coefficient A is:

$$A = \frac{[\sum(u_n - \bar{u})(x_n - \bar{x})][\sum(y_n - \bar{y})^2] - [\sum(u_n - \bar{u})(y_n - \bar{y})][\sum(x_n - \bar{x})(y_n - \bar{y})]}{[\sum(x_n - \bar{x})^2][\sum(y_n - \bar{y})^2] - [\sum(x_n - \bar{x})(y_n - \bar{y})]^2}$$

The equation for B can be found by swapping x for y. Similarly, the equation for C can be found by swapping u for v in the equation for A, and that for D by swapping x for y and u for v.

Clearly, the upper right panels of Figure 5 shows that there is so much pixel-phase bias in each of the x and y positions that goes into the average that the average positions themselves should also be expected to have considerable remaining systematic error. Even so, these average positions will be less biased (by $\sqrt{8}$, i.e. $\sqrt{N_{obs}}$) than those from the individual exposures, so that we should be able to reduce and eventually remove the errors by iteration.

4.3 STEP 3: EXTRACTING THE PSF

The averaging process described in §4.2 produces an average position for each star in the frame of the first exposure. We use the inverse transformations and the reverse distortion solutions to map this average position of each star into the pixel frame of each individual exposure. We use these mapped positions and the original measured fluxes to extract the PSFs, making use of Equation E3 above.

Dealing with pixel-phase errors. Note that even though we use the average positions when we extract the PSF, we will use the individual fluxes from each individual exposure to solve for the PSF. There could certainly be pixel-phase type errors in photometry as well as in astrometry, but the lower right plot in Figure 5 shows that such errors are much smaller than the astrometric errors (at least along *x;* we inspected a similar plot for *y* and they are equally small).

Pixel-phase errors in photometry would be indicative of a variation in the quantum efficiency (QE) across the face of a pixel. When the PSF is undersampled and there are such QE variations, then a detector will record more total flux for a star if the star's brightest point falls on a more sensitive part of the pixel, and it will record less flux if the brightest point of a star lands in a less-sensitive part of the pixel. This was the case for WFPC2, where stars were 1.5% brighter than average if they landed at the center of a pixel (see Lauer 1999b). This is likely due to the location of photon-absorbing electrodes at the pixel boundaries. The most undersampled NICMOS detector, NIC3, had more than a 10% variation in total flux with pixel phase.

Our PSF model can easily accommodate this variation if we allow it to have different normalization "volume" at different pixel phases. Since it can be a challenge to constrain the PSF in this way and this is clearly a subtle effect for WFC3/IR, while we are solving for the PSF we will choose to normalize the

PSF such that it is consistent with a flat-and-flush pixel response. Later, in §5 we will explore QE variations and adjust the PSF to account for them.

We saw in our initial exploration of the PSF in Section 3.5 that the PSF clearly varies with position on the detector. In the first few iterations of solving for the PSF, we will solve for a single PSF across the entire detector, then later we will allow the PSF to vary spatially.

We have 1037 stars in the F105W data set. Each of these stars is found in at least five out of the eight exposures. For each star, we have a raw position and a flux, which together enable us to convert each pixel in the vicinity of the star into a point sampling of the PSF. To build up the PSF, we go through each of the 101×101 PSF gridpoints and identify the pixel samplings that lie within 0.25 pixel of the gridpoint's location in the Δx and Δy plane. We next subtract the current PSF estimate from the samplings (in the first iteration, $\psi = 0$ everywhere) to determine how much the PSF would have to change at this location to better represent the sampling and construct a robust average of all the residual estimates. We then add all these residual estimates to the current PSF to arrive at an improved estimate of the PSF.

Smoothing. Recall that in formulating the PSF, we opted to have PSF grid that was sampled finely enough to contain more nodes than were absolutely necessary, so that we could use the same PSF parameterization for all detectors. Since the PSF model is highly oversampled relative to the detector pixels (×4), we want to make sure that it doesn't have any unphysical variations, so we impose some mild smoothness constraints. There is no perfect way to do this, since it is never clear *a priori* how much structure the PSF should intrinsically have. In practice, to determine how much smoothing the PSF will tolerate, we increase the amount of smoothing until the residuals to the PSF fit indicate that the model is not able to match the sharpness of the true PSF. We then back up to the smoothest level that allows all model to be as sharp as the data.

The smoothing we adopted for WFC3/IR is a two-step process. First we smooth the PSF with a 3×3-pixel boxcar. This enforces pixel-to-pixel smoothness, but removes some of the true, sharp structure at the center. We then construct the residuals between the original and smoothed PSF and convolve them with an additional kernel, then add what survives back to the boxcar-smoothed PSF to arrive at the final smoothed PSF. At the center of the PSF, where the variation would be expected to be the sharpest, for this secondary smoothing, we use a 5×5 kernel that respects quartic variations[3]. Beyond three pixels we allow only quadratic variations, and beyond 5 pixels we simply adopt the original 3×3 boxcar-smoothed value. We find by looking at the residuals that this is the maximum amount of smoothing that can be done before we start to suppress real structure.

---

[3] This kernel is equivalent to taking each point and fitting the surrounding 5×5 array of gridpoints with a 2-d quartic function and replacing each gridpoint with the value of the fitted quartic at that location. A 2-d quartic has 15 terms, so with 25 constraints it has the effect of making the gridpoints only 60% independent.

Additional constraints on the PSF model. In addition to the smoothing, there are two other constraints we must impose on our PSF models. The first is that it must be normalized. After each iteration, we rescale the value of the PSF such that the sum of the PSF over the inner 5.5 pixels is 1.00. This means that when we fit our PSF to stars, we will get a flux that corresponds to aperture photometry with an aperture of $r = 5.5$ pixels. This normalization is the same for all pixel phases, and for all 3×3 fiducial PSFs across the detector.

The other constraint we must impose concerns the centering. Without such a constraint, a purely empirical model could easily let the PSF wander through the 101×101 grid. It is sensible that we would want the PSF to be centered on the PSF grid, particularly as we will be using bi-linear interpolation to determine PSFs in between the 3×3 fiducial locations across the detector. (If spatially adjacent PSFs are centered differently, we will get two-headed PSFs in between them.)

There is no single way to center the PSF. One could adopt the peak of the PSF as its center, but this point is hard to measure precisely since the gradient is zero there. One could adopt the "centroid" of the PSF, but that is also poorly defined as it can be pulled in strange ways by flux well outside the core. Here, we will define the center of the PSF as the point of maximal radial symmetry over a ~1.5-pixel radius. To find this point, we identify those PSF gridpoints (i, j) that fall within a radius of 6.5 gridpoints (there are 137 of them). We compute an asymmetry metric: $A = \Sigma_{r<1.5} |\Psi_{ij} - \Psi_{-i,-j}|$. We then evaluate the same metric for PSFs that have been re-centered at an array of (δx, δy) shifts. (The re-centering uses the same bi-cubic interpolation as the PSF evaluation.) We identify the optimal center as the one that has the smallest asymmetry metric and then adopt this re-centered PSF as the new PSF. As with the normalization, this is done separately for each of the 3×3 fiducial PSFs.

4.4 Iterating to Improve the PSF

Now that we have all the logistics settled, we can start to construct the PSF. Figure 6 below shows the evolution of the central PSF as we go from the first iteration (red) to the final (9th) iteration (black). The iterations and the averaging process together allow us to improve the positions for the stars (which are the inputs for the PSF), and this in turn allows the PSF to sharpen up to its true value. There is also a small amount of shifting that takes place, indicating that the initial crude centroid was not quite a point of symmetry.

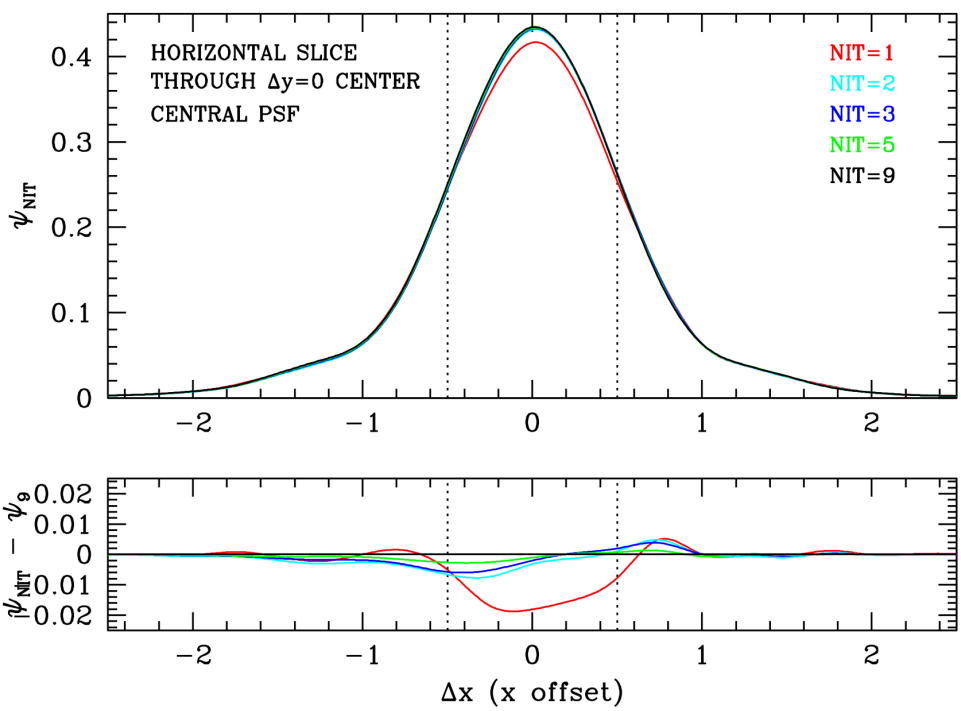

Figure 6: This shows a horizontal slice through the center of the central-field PSF for the final iteration (number 9, shown in black) and previous iterations (in various colors as labeled). The bottom panel shows the difference between the iterations with respect to the final iteration.

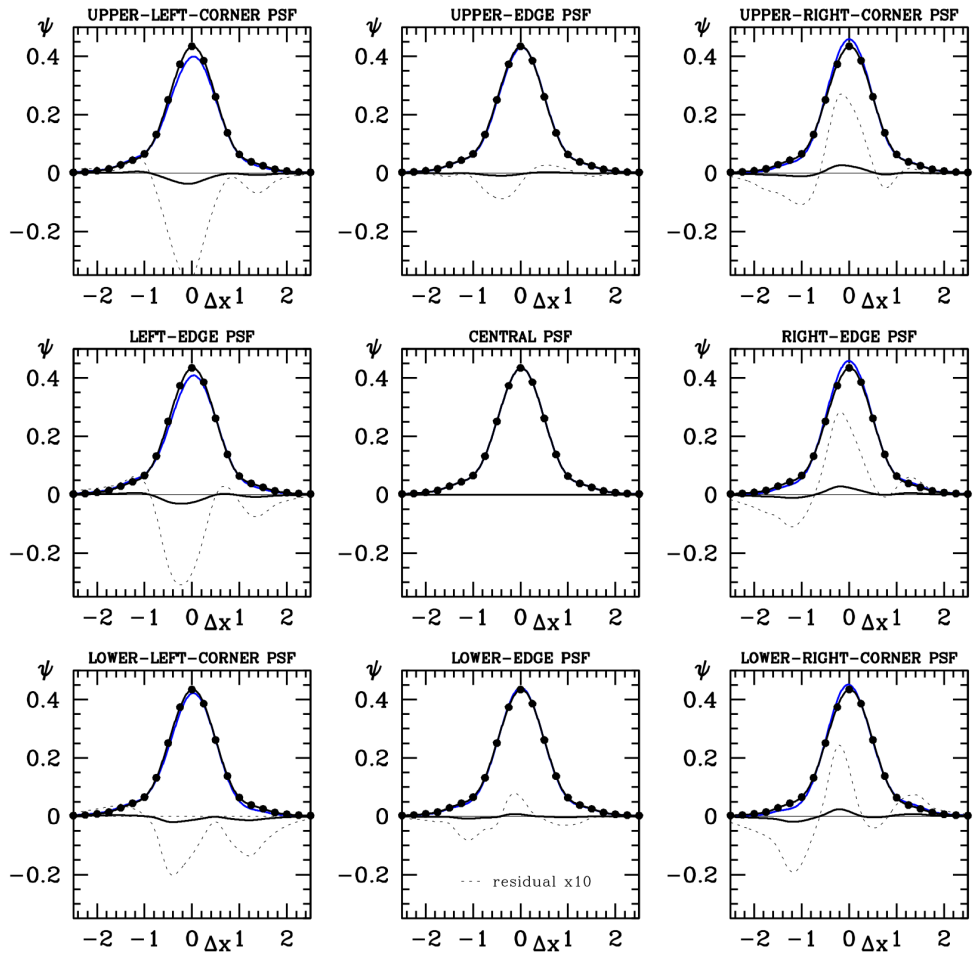

Figure 7: This shows a horizontal slice through the center of the PSF for the 3×3 array of fiducial PSFs after the final iteration. The solid blue line is the PSF as labeled at the top of the panel. The solid black line is the PSF for the central region and the 0.25-pixel-space gridpoints are shown as solid dots (the same in all panels). The solid light line is the difference between the local PSF and the central PSF. The dotted line is the same difference multiplied by 10. It is clear that the PSF varies by up to 10%, both in the core and elsewhere.

Figure 7 shows the central horizontal slice — $\psi(\Delta x, \Delta y=0.0)$ — for the 3×3 array of fiducial PSFs. In each panel, we provide the central PSF for comparison and also report the difference between the two. This shows that the core intensity of the PSF varies by up to 10%, as was seen in Figure 4. It is also clear that the PSF changes in ways other than just the sharpness of the core (which is all we showed in Figure 4), in that sometimes there is an enhancement or deficit on one side of the PSF

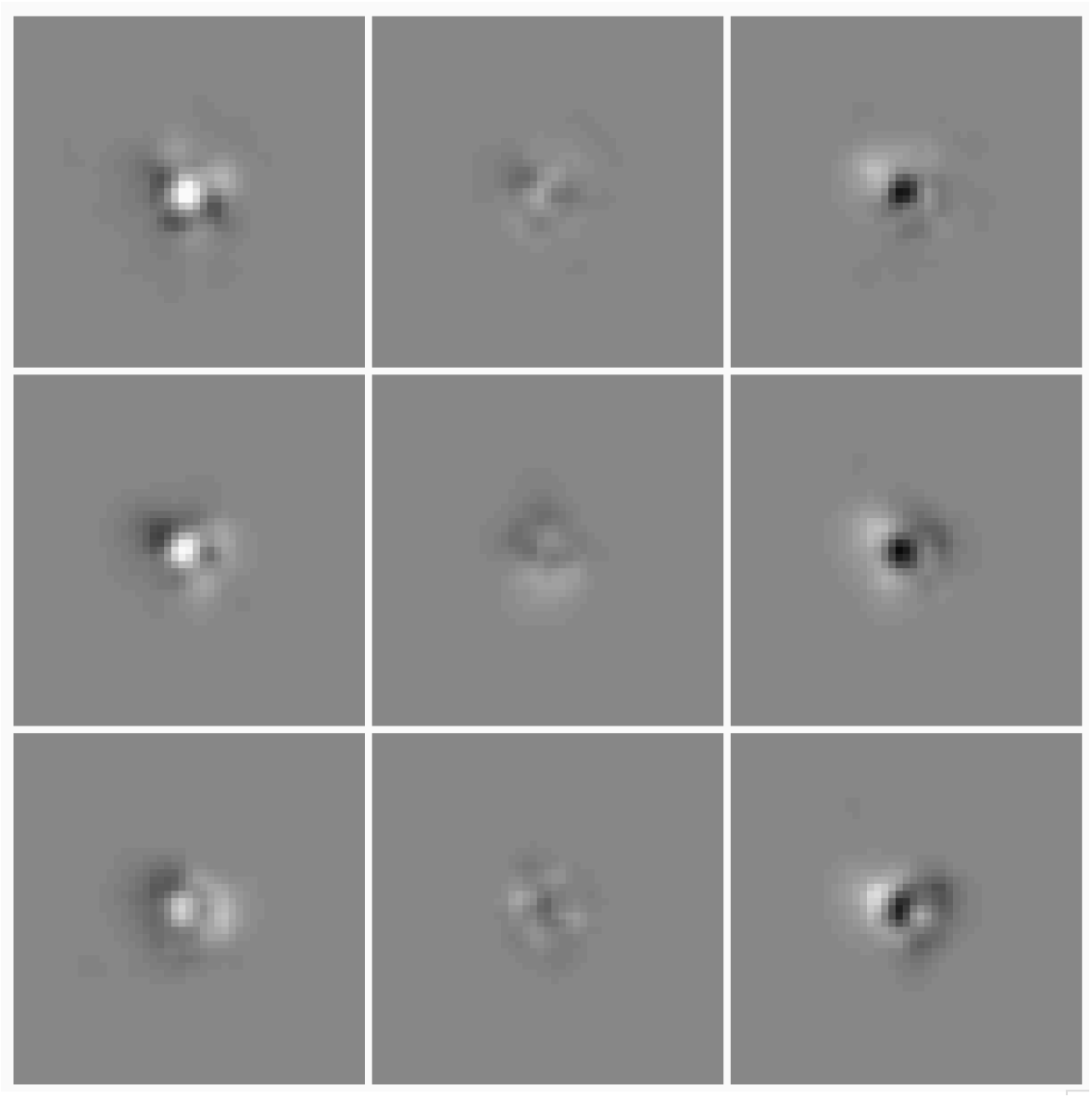

Figure 8: This shows the residuals between the average PSF and the 3×3 array as a grayscale image. Dark represents more flux, and light less flux. The largest variation is about 0.04, which is about 10% of the central value of the PSF.

Figure 8 shows the residuals between the 3×3 PSFs across the detector and the average PSF. It is clear that the biggest trend is the one that we saw in Figure 4, that there is more flux in the core on the right side of the detector than on the left side. But the residual pattern is more complicated in that there are some features in the inner halo that change from panel to panel.

It is worth noting that the changes in the distortion solution across the field cause the area covered by each pixel to change across the field as well. This change is mostly along the *y* axis, therefore the change in PSF sharpness we see from left to right cannot be due to the central pixel changing size and less flux landing in it. The variation of light in the core must be due to more complicated telescope optics.

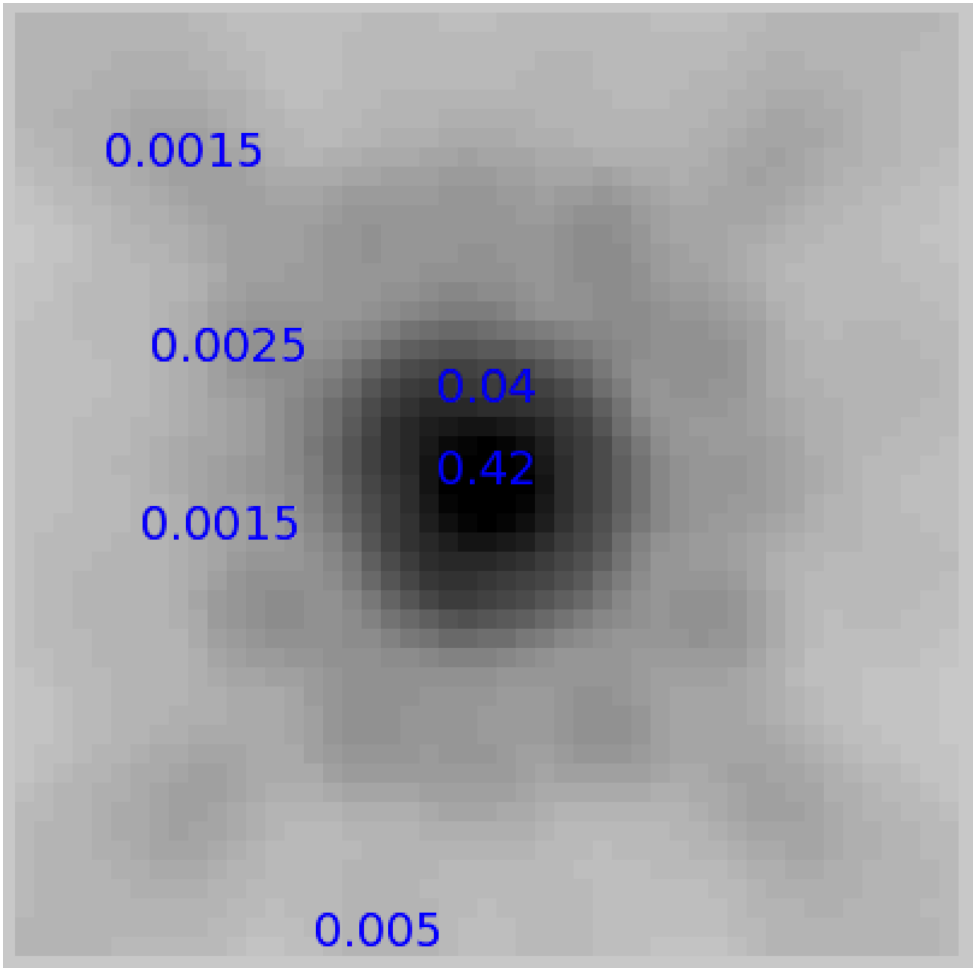

Figure 9: A log image of the final central PSF, with the value of the PSF identified at specific locations. The full PSF is shown here: each pixel represents one of the 101×101 grid-points and the image covers the central 25×25 pixels of the PSF, i.e. from −12.5 to +12.5 in both *x* and *y*.

The PSF is a four-dimensional function that varies by more than 4 orders of magnitude within its own (Δx, Δy) domain and with (i,j) position on the detector. Figure 9 shows a log plot of the PSF, with its value at a few points indicated. A pixel that is centered on the star receives about 42% of the star's light. A pixel in the inner halo that is about 1.25 pixels off from the center receives about 4% of the light. A pixel in one of the bumps in the outer diffraction ring has about 0.25% of the light in it, while a point in a non-bump portion of the outer ring receives about 0.15% of the light. A pixel on the diffraction spike at a radius of about 10 pixels receives about 0.15% of the light, while a minor feature at about the same radius collects about 0.05% of the light in each pixel.

The plots in Figures 6 through 9 show various aspects of the PSF from the first iteration to the last. The iterative process succeeds because the improved PSFs result in improved positions for the stars, which in turn give us in improved transformations and then improved placement of the PSF samplings.

4.5 STEP 1 IN SUBSEQUENT ITERATIONS: FITTING STAR IMAGES WITH A PSF

One of the output products of the first iteration is a complete PSF model, so to get new positions we can fit the inner 5×5 pixels for each star with the a PSF model rather than with the crude centroids we started with. Since the fit for position is highly non-linear, we perform a progressive grid search to identify the best position. We start with a 7×7 grid centered on the middle of the star's brightest pixel with a grid spacing of 0.25 pixel. We identify as the best center the location that minimizes χ^2, where:

$$\chi^2 = \sum_{i,j \in 5\times5} \left(\frac{P_{ij} - s_{\text{star}} - f_{\text{star}}\psi_{ij}}{\sigma_{ij}} \right)^2 \qquad (E5)$$

and $\psi_{ij} \equiv \psi(i\text{-}x_{\text{star}}, j\text{-}y_{\text{star}})$. When σ_{ij} can be approximated as the Poisson limit value of $\sqrt{P_{ij}}$, as it can for these bright stars, then the least-squares solution for f_{star} simply reduces to the sum of the pixels in the aperture divided by the sum of the PSF in the aperture: $f_{\text{star}} = \Sigma(P_{ij}\text{-}s_{\text{star}})/\Sigma\psi_{ij}$. This allows us to evaluate χ^2 for every trial position (x_{star}, y_{star}) and adopt the position with the minimum χ^2 as the best position.

Once we identify the best position within the trial grid, we re-center the grid on the new position and shrink the size of the grid. After nine such iterations, the final grid has spacings of 0.001 pixel. This corresponds to the precision of our positions, though they are not necessarily this accurate: Poisson noise would predict a limiting accuracy of about 0.005 for the brightest stars. This grid search is quite fast and does not slow down the PSF-construction procedure when it is coded in FORTRAN. Since the table-based PSF is trivially easy to differentiate, it would be easy to formulate this fit into a gradient search for the best position. This would converge considerably faster.

Figure 10 on the next page shows the PSF fits for three stars of roughly equal brightness in the first exposure, one centered largely on a pixel, one at the edge between two pixels, and one at the corner between four pixels. The linear nature of the fit with the correct center is clear from the plots: we are simply plotting the observed value on the *y*-axis against the model value on the *x*-axis. The slope is the flux of the star. For each star, we assign a quality-of-fit metric that reflects the fractional absolute value of the residuals to the pixel fit to the inner 5×5 pixels: $q_{\text{star}} = \Sigma|P_{ij} - s_{\text{star}} - f_{\text{star}}\psi_{ij}| / f_{\text{star}}$. The total fractional residual for each of these three is about 4%, typical of the well-fit stars.

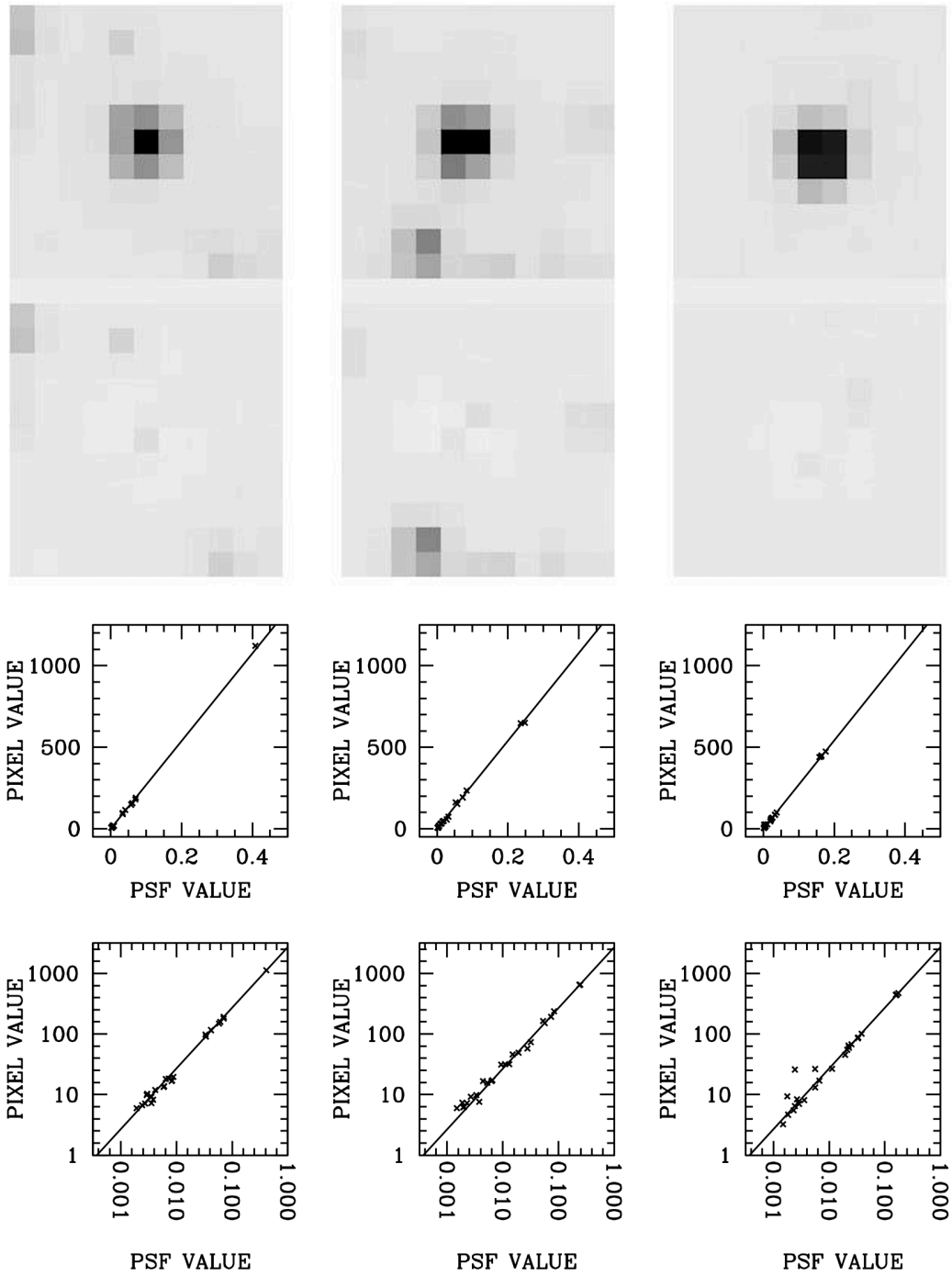

Figure 10: This figure shows the PSF fits to three bright stars in the first F105W exposure, `ichi01a1q_flt.fits`. The top row of plots shows each star as it appears in the image. The first star is centered on a pixel at (200.034, 736.030), the second on the edge between two pixels at (63.483, 787.000), and the third on a corner at (355.499, 256.511). The second row shows the image with the star subtracted (with the same stretch as on top). The third and fourth rows show the fit in linear and log perspective, respectively. All three stars have about 2700 total counts per second, and the third star appears to have a faint neighbor to its upper right (hence the excess of flux in a few pixels).

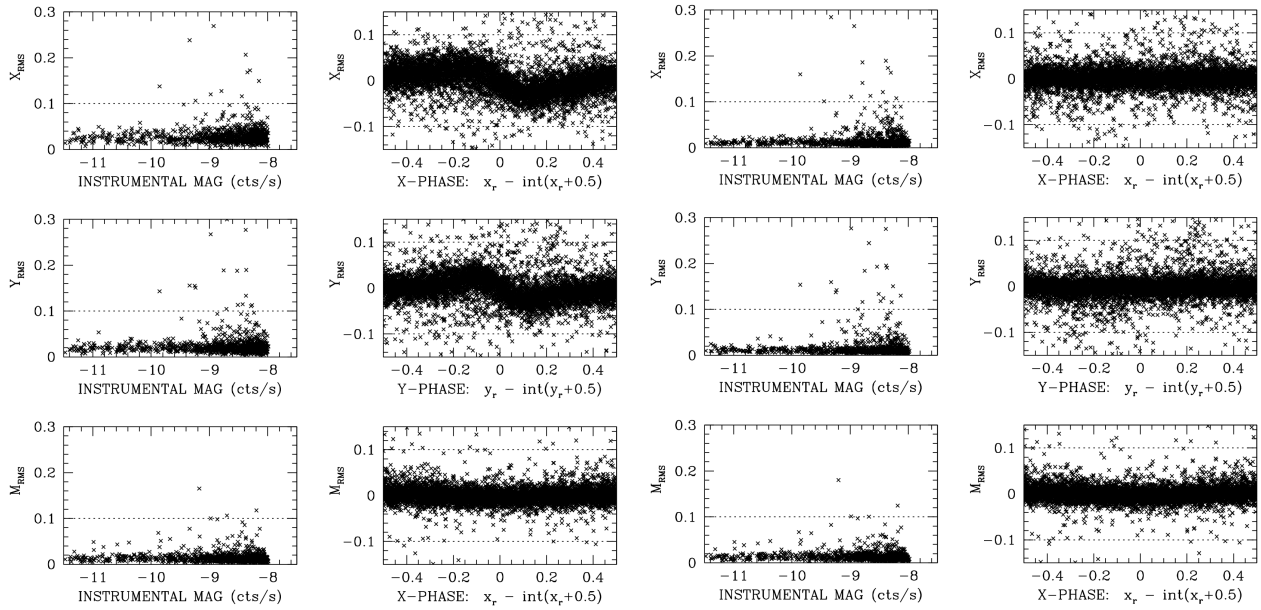

Figure 11: This shows the version of Figure 5 for the second iteration on the left and the final iteration on the right. The leftmost panel shows the final residuals.

4.6 Step 2 After the First Iteration: Transformations and Residuals

The second step in the iterations involves taking the positions measured in the first step and averaging them together to get a more accurate composite position for each source in the reference frame (the distortion-corrected frame of the first exposure). These positions will then be transformed back into the individual frames for use in PSF construction.

Figure 11 above show the results from the second iteration on the left and the final iteration on the right. It is clear that even after just one iteration, the position residuals tighten up considerably: in Figure 5 we saw astrometric residuals of about 0.07 pixel, and here the residuals are 0.02 pixel. The second column of panels shows that the pixel phase errors also improve quickly. Even after just one iteration, the pixel phase residuals are down from an amplitude of 0.10 pixel to less than 0.02 pixel. By the final iteration, the RMS positions are good to about 0.01 pixel and the astrometric pixel-phase errors are well below 0.002 pixel.

5 EXPLORING THE POSSIBILITY OF QE VARIATIONS

The PSFs we extracted above were constructed to be consistent with a flat pixel response (see §4.3). In terms of normalization, this means that the effective PSF has the same normalization for every pixel phase, or:

$$\sum_{\sqrt{(i^2+j^2)}<5.5} \psi(i+\phi_x, j+\phi_y) = 1.0$$

for all pixel phases (ϕ_x, ϕ_y), where ϕ_x and ϕ_y go from −2 to +2. This is what we would expect if the pixels were flush (i.e., having no spaces between them) and flat (they have an even response meaning that a photon is equally likely to be detected at all locations within the pixel). If the pixels are not flush and flat, then we will find that the total number of counts collected for a star depends on its pixel phase. The sharper the PSF is, the larger we would expect this effect to be.

The intra-pixel-sensitivity function has been explored in two previous ISRs. In 2008, McCullough reduced some complicated TV2 (Thermal Vac #2) data and inferred an 8% peak-to-trough variation in the pixel-response function for the F105W data. He found that the centers of the pixels were less sensitive than the edges, but this study had complicated time and spatial normalization issues, so he cautions that it should only guide us regarding things to explore on-orbit. Such large variations have never been seen on orbit. Pavlovsky et al (2012) examined some data of Omega Cen and found the centers to be slightly less sensitive than the edges in F105W, but they did not deem this result to be statistically significant.

In constructing the PSF, the flux that we used to turn the pixels in the vicinity of each star in each exposure into PSF samplings came directly from the exposure itself, so there was no way for our PSF model to include any QE variations. Even so, we can still examine the photometric residuals to see whether the flux that we measure for each star from the flat-QE-normalized PSF has any dependence on pixel phase.

The lower right plots in Figures 5 and 11 showed no strong behavior of magnitude residual against *x* pixel phase. In Figure 12, we re-plot in the left panels the photometric residuals against pixel phase for the final iteration with an expanded scale and see that there actually are pixel-phase trends with an amplitude of about 0.004 magnitude in both x and y. The contour plot on the right shows the two-dimensional structure. It is clear that when a star is centered on a pixel, the detector records about 0.85% more electrons, and if it lands in a corner it records about 0.5% fewer. We note that this is not just a simple aperture correction. While stars are fitted over a 5×5 pixel aperture, the PSF is normalized over a 5.5-pixel-radius aperture, and our photometry represents stars fitted to the PSF over their inner 5×5 pixels. We will explicitly show that this is not an aperture correction below.

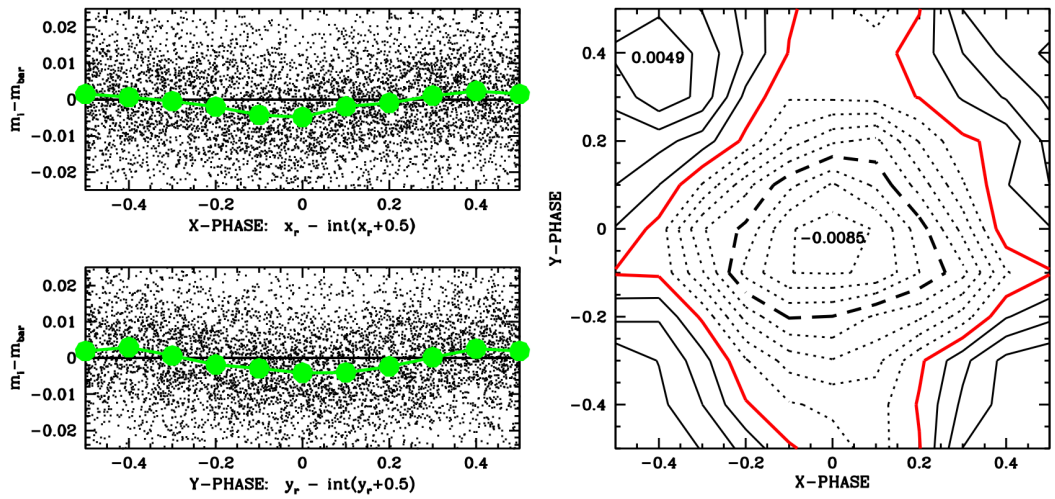

Figure 12: Left panels: photometric residuals plotted against pixel phase for *x* and *y*. Right panel: contour plot showing the average photometric residual as a function of 2-D pixel phase.

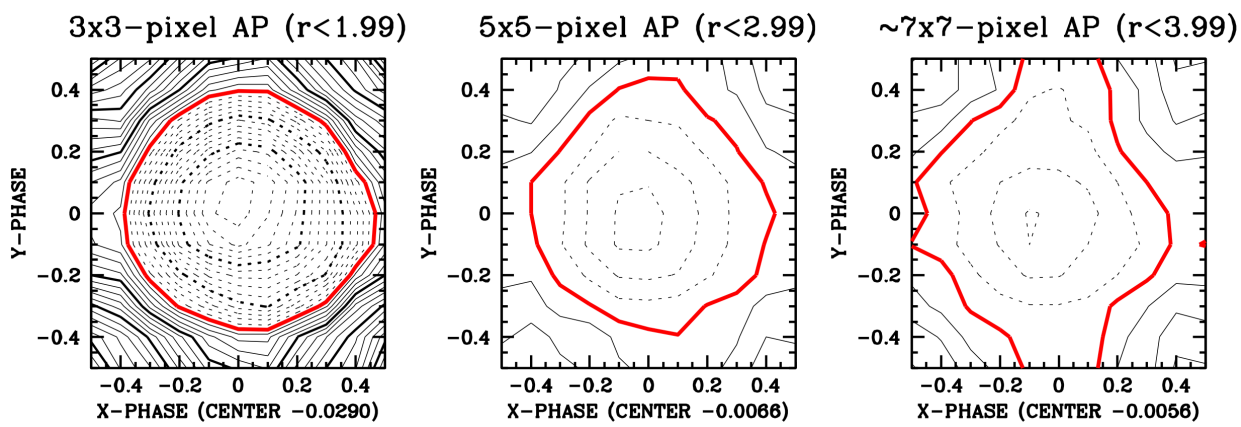

Figure 13: Contour plots showing the trends in photometric residuals as a function of *x* and *y* pixel phases for three different apertures: a small one (3×3 pixels), a medium one (5×5 pixels) and a larger one. The dark contours are 1% and the light contours are 0.2%. The dotted contours are negative. The heavy red contour is at zero.

In an effort to demonstrate that this trend is not simply a function of our PSF formalism, we perform the same calculation with simple aperture photometry on stars that were found in all eight dithers. We went through the eight dithered images and performed generic aperture photometry on every source with three different apertures: a very small 3×3-pixel aperture, a medium-sized 5×5-pixel aperture, and a larger ~7×7-pixel aperture. For each star, we found an average flux and compared the flux for the star in each exposure against the average as a function of pixel phase (after correcting for the pixel-area correction). The contour plots in Figure 13 above show the results.

The plot on the left shows a very strong trend with pixel phase in that a star that is centered on the middle pixel has 3% more flux than average within a 3×3 aperture. Most of this is a simple reflection of the fact that more of the PSF lands within the aperture when the star is centered on the aperture — a textbook aperture correction. When we use larger and larger apertures, this correction should go to zero.

It is clear that the pixel-phase trends are drastically reduced when the aperture radius is increased by just one pixel around the border in the 5×5-pixel aperture of the middle plot. A star that is centered on a pixel is only 0.66% brighter than average. The trend hardly changes at all when the aperture is increased by an additional pixel to 7×7 in the rightmost plot. This residual pattern is therefore not related to the aperture but rather is a reflection of the intrinsic detector properties.

The effect we see could in principle be related to persistence, since it has recently been observed that in the first exposure of an un-dithered series, stars tend to be measured fainter than in subsequent exposures (Knox Long, personal communication; ISR in progress). This can be understood as electrons filling the traps in the pixels and and not getting measured when the pixel is read out. These electrons later come out as persistence. If this effect is non-linear and turns out to affect brighter pixels more than fainter pixels, then a star of a given brightness could suffer more traps when most of its flux is on one pixel. This would phenomenologically explain what we see. However, we have verified that the trend seen in Figures 12 and 13 are the same for all stars, regardless of brightness. Since persistence is highly non-linear with a star's flux, if what we see is related to persistence, we would expect it to be much worse for brighter stars. Since we do not see that, it seems safe to conclude that it is a detector-QE-related effect, which should be linear with the stars' flux.

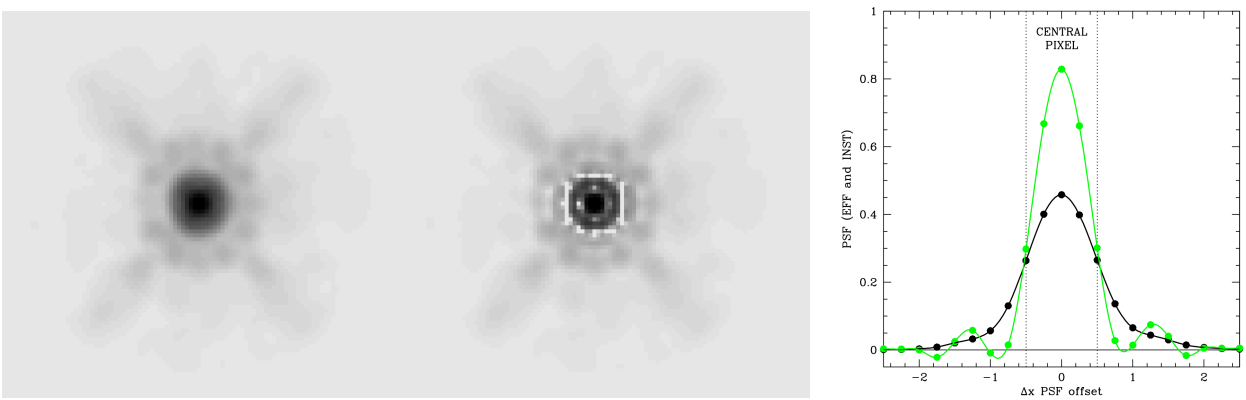

Figure 14: In the figure on the left we show the effective PSF extracted for the F105W filters, and the deconvolved "instrumental" PSF on the right. The plot on the right shows horizontal slices through the center (Δy=0). Note that the instrumental PSF goes slightly negative between the core and the first diffraction ring. In principle, we should impose a positive-only constraint, but since we are focused here on the core, we decided to just use this slightly unphysical PSF.

5.1 Matching QE Variations with Simulations

In order to see how QE variations can affect the total fluxes of stars, we performed some simulations of how a finely sampled PSF might interact with a pixel in detail. The effective PSF already includes the integration over pixels, so it is not directly able to help with these simulations. We took the effective PSF and found the closest "instrumental" PSF to it (recall that $\psi_E = \psi_I \otimes \Pi_{\mathrm{DET}}$) by deconvolving it with an assumed pixel-response function (Π_{DET}). We started by assuming a flat-and-flush pixel-response function, since our ψ_E normalization is consistent with that. We accomplished this deconvolution by iterative forward modeling, applying the same smoothness constraints on the instrumental PSF as we had applied to the original effective PSF. The two PSFs are shown in Figure 14.

The deconvolution shows how truly sharp the F105W PSF is. The surface brightness at the center of the PSF is such that if it were constant across the entire central pixel, 85% of the light would land in that

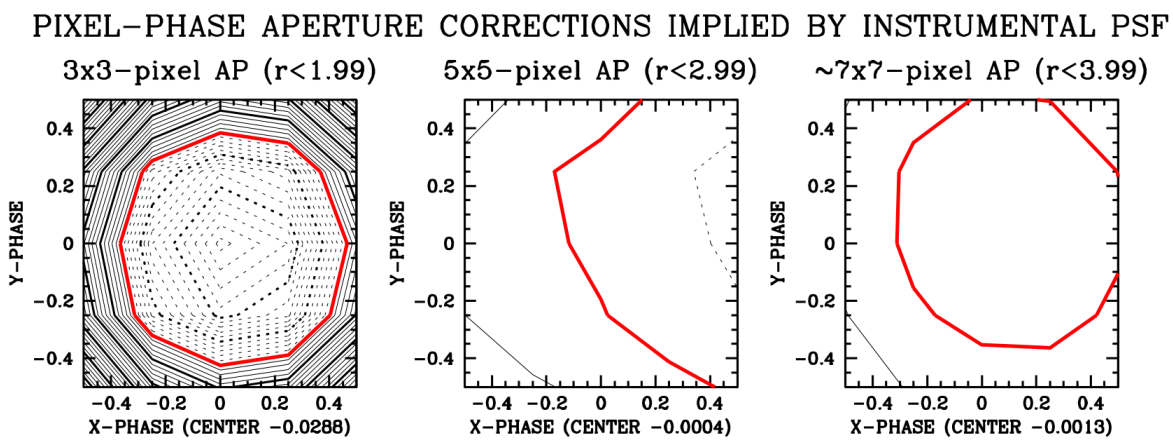

Figure 15: This is the analogous plot to Figure 13 above, except that it is based purely on the deconvolved instrumental PSF, rather than the observed photometry and the assumption of a flat pixel-response function (Π_{DET}). The contours are the same as in Figure 13. The lack of central symmetry in the middle panel is an indication that the asymmetric distribution of light in the diffraction ring dominates the general radial behavior of the PSF.

pixel. The FWHM is considerably less than a pixel. It is clear that the PSF is sharp enough that if there is any variation in sensitivity across the face of the pixel (i.e., Π_{DET} is not a pure top-hat), then we would expect to see variations in the total recorded number of counts with pixel phase.

Now that we have an estimate of the instrumental PSF, we can show what Figure 13 would look like if there is no pixel-response-function variation, in other words, if the contour plot simply represented just an aperture correction for the star's being slightly off-center in the aperture.

Figure 15 shows the simulated photometry in the same way as Figure 13 for the real photometry. It is clear that the pixel-phase-related aperture correction we saw for the 3×3-pixel aperture assuming a top-hat pixel-response function (in the left panel of Figure 13) is very similar to the simulations (same panel of Figure 15): we get about 2.5% more light than average in the 3×3-pixel aperture if the star is centered on the pixel. The other two panels, however, show a very different story. From our instrumental PSF and the top-hat Π_{DET} simulations, we would expect very little pixel-phase-related corrections (less than 0.1%) for the 5×5-pixel aperture or the 7×7-pixel aperture, while we actually observed about 8 times this amount in our empirical photometry (Figure 13)., Figure 13 is clearly telling us something about the pixels.

Since we have a deconvolved instrumental PSF in hand, we can explore how a non top-hat pixel-response function (PRF, which we represent as Π_{DET}) would affect the observed trends in photometric residuals against pixel phase. We explored two different kinds of PRF. The first was a PRF that was flat, but not flush, corresponding to gaps between the pixels. The second was a sinusoid with a maximum at the center of the pixels and the minimum at the edge/corner.

Figure 16 shows the resulting photometric residual trend with pixel phase. We tweaked the amplitude of the perturbations to the PRF to get the same (observed) central excess. We note that in the left plot, the central value is about half as high as the corner value is low, whereas in the right plot, the central value and the corner value are about the same. The latter is close to what we see for the real data (see Figures 12 and 13), but the real data actually has an even higher value at the center than at the edge.

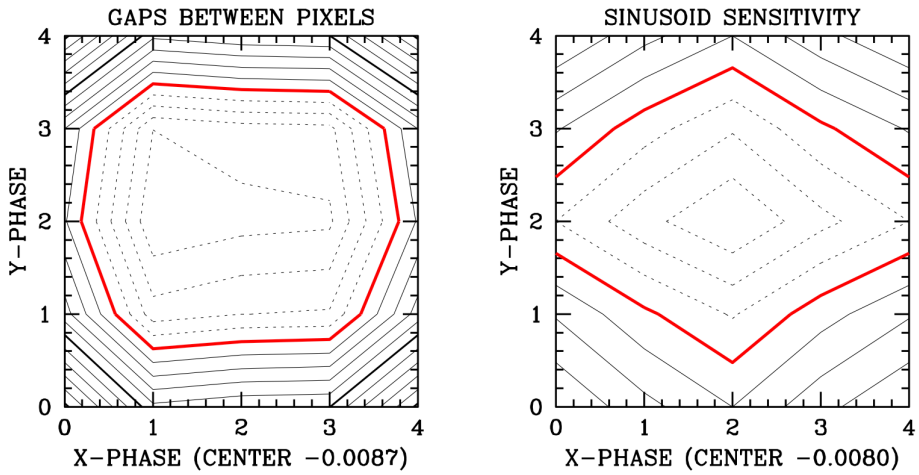

Figure 16: The simulated trend of photometric residuals given the solved-for instrumental PSF and two different assumptions for Π_{DET}: one that has simple gaps between the pixels and one that is sinusoidal.

There is clearly much more to explore in regards to the pixel-response function. One additional degree of freedom is that the PRF could in practice extend beyond the [−0.5:0.5, −0.5:0.5] domain that we have explored here. Such a PRF would reflect "charge diffusion", which happens when a charge is generated in one pixel, but the electron gets collected by an adjacent pixel. That would probably make the behavior even more extreme.

5.2 Fixing the PSFs to Reflect the QE Variations

Even the sharp WFC3/IR PSF is a relatively blunt instrument to explore the pixel-response function (PRF), since the PRF can easily have sub-structure on very small scales since the small electrodes that define the pixels can affect the pixel's QE. One could easily fold in data collected from lab experiments, which often use a laser to explore the PRF in detail. But that is beyond the scope of this effort.

To fix the PSF, we simply need to adjust the sub-pixel grid-point values within the central pixel to account for extra or missing flux that results when the peak of the PSF lands at particular pixel phases. To this end, we distilled Figure 12 into a set of 5×5 values covering the PSF gridpoints in the central pixel. The following table represents the percent excess or deficit for each of these inner gridpoints:

−0.37	−0.12	0.06	−0.12	−0.37
−0.24	−0.06	0.17	−0.01	−0.24
−0.10	0.41	0.81	0.36	−0.10
−0.22	0.12	0.39	0.11	−0.22
−0.37	−0.12	0.06	−0.12	−0.37

This means that the central pixel of the PSF should be increased by 0.0081 (since the PSF is recorded in terms of the fraction of the total light, rather than a percentage). We need to take care at the pixel edges and pixel corners not to double-count this adjustment. Each of the twelve edge pixels should get one half the correction shown above, and each of the four corner pixels should get one quarter the correction.

We made this adjustment to the PSF and in Figure 17, we show with central values of the PSF (±3 about the central grid point of [51,51]) before and after being tweaked to account for this. The boundaries of the central pixel (where $|\Delta y|<0.5$ and $|\Delta x|<0.5$) are shown by a dashed line. No changes are made beyond the central pixel.

In hindsight, it might have been possible to include this normalization in the initial PSF-construction process. However, it would not have been trivial, given the need to apply and remove the pixel-area correction, but it could be done. Perhaps we will include that in the algorithm in the future, but we have not done that here.

BEFORE ADJUSTMENT

0.075	0.107	0.133	0.140	0.126	0.097	0.065
0.097	0.162	0.228	0.259	0.233	0.168	0.098
0.118	0.220	0.326	0.380	0.341	0.236	0.127
0.132	0.250	0.372	0.433	0.384	0.261	0.137
0.133	0.232	0.333	0.379	0.335	0.231	0.125
0.117	0.176	0.232	0.253	0.225	0.162	0.097
0.093	0.117	0.134	0.135	0.122	0.097	0.068

AFTER ADJUSTMENT

0.075	0.107	0.133	0.140	0.126	0.097	0.065
0.097	0.161	0.227	0.258	0.232	0.167	0.098
0.118	0.219	0.327	0.384	0.340	0.235	0.127
0.132	0.250	0.376	0.441	0.386	0.262	0.137
0.133	0.232	0.334	0.382	0.335	0.230	0.125
0.117	0.176	0.231	0.253	0.224	0.161	0.097
0.093	0.117	0.134	0.135	0.122	0.097	0.068

Figure 17: The central 7×7 grid-points of the PSF model — Ψ[48:54,48:54] — before the QE correction (on the left) and after the QE correction (on the right).

The final step is to validate the improvement. Figure 18 shows the residuals corresponding to the new PSF. Again, we have 8 exposures and each residual represents the difference between the observed position/photometry in a given exposure and where the average of all 8 would predict for the star in that exposure. We have shown the astrometric residuals again to demonstrate that the change we made to fix the photometry had a negligible effect on the astrometry. It is clear that the photometry is much better. It is not perfect: the trends have gone down from 0.8% to about 0.1%. But these residual systematic errors are considerably smaller than the precision of an individual measurement.

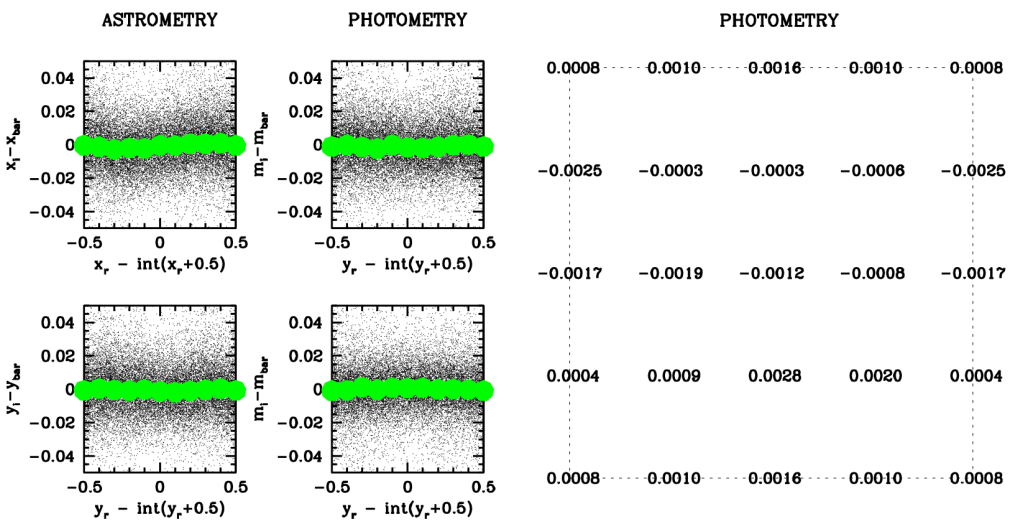

Figure 18: This shows the astrometric (left) and photometric (right) errors as a function of pixel phase for the QE-fixed PSF. On the right, we distill the residuals into two-dimensional quarter-pixel bins.

6 EXTRACTING THE PSF FOR DIFFERENT FILTERS

The previous two sections did an exhaustive job documenting the extraction of the PSF for WFC3/IR observations through F105W. We used the same procedure for four other commonly used filters: F110W, F125W, F140W, and F160W and show the results here.

Figure 19 below shows horizontal slices through the central-field PSF for the 5 wide-band PSFs under study here. It is clear that the fraction of light in the central pixel goes down monotonically from 43% at F105W to about 33% for F160W; the FWHM is seen to increase as the peak goes down. We also show a deconvolved version of the PSF, just for reference. It is interesting to note that although the effective PSF decreases monotonically with radius for all PSFs, the detailed behavior is such that it is necessary to include a diffraction ring in the deconvolved product. As expected, this ring moves outwards as the wavelength goes up, similar to the FWHM.

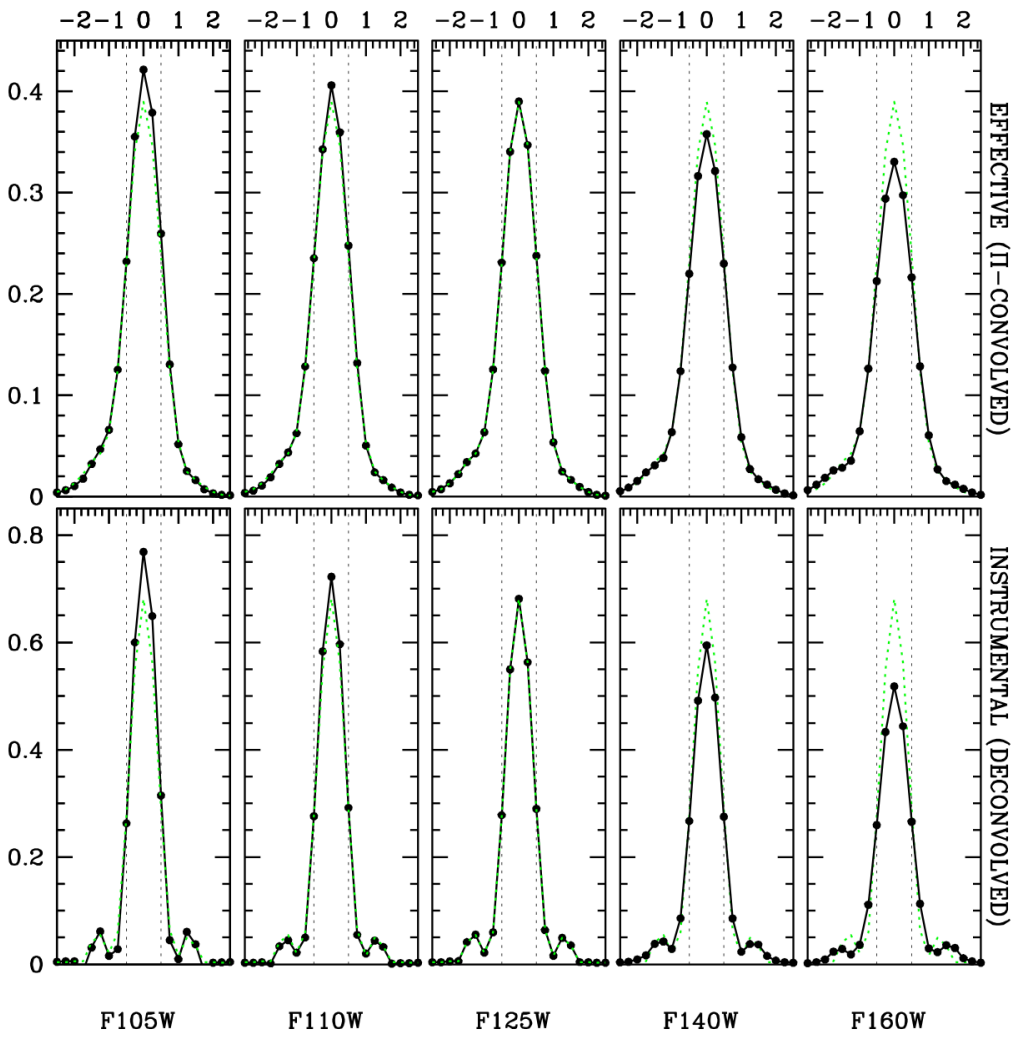

Figure 19: Top: effective PSF profile for $\psi(\Delta x, \Delta y=0)$ for the central-field PSF for the five filters under consideration here. Bottom: For reference, the deconvolved versions (where the deconvolution assumes a flat-flush pixel-response function Π_{DET}.

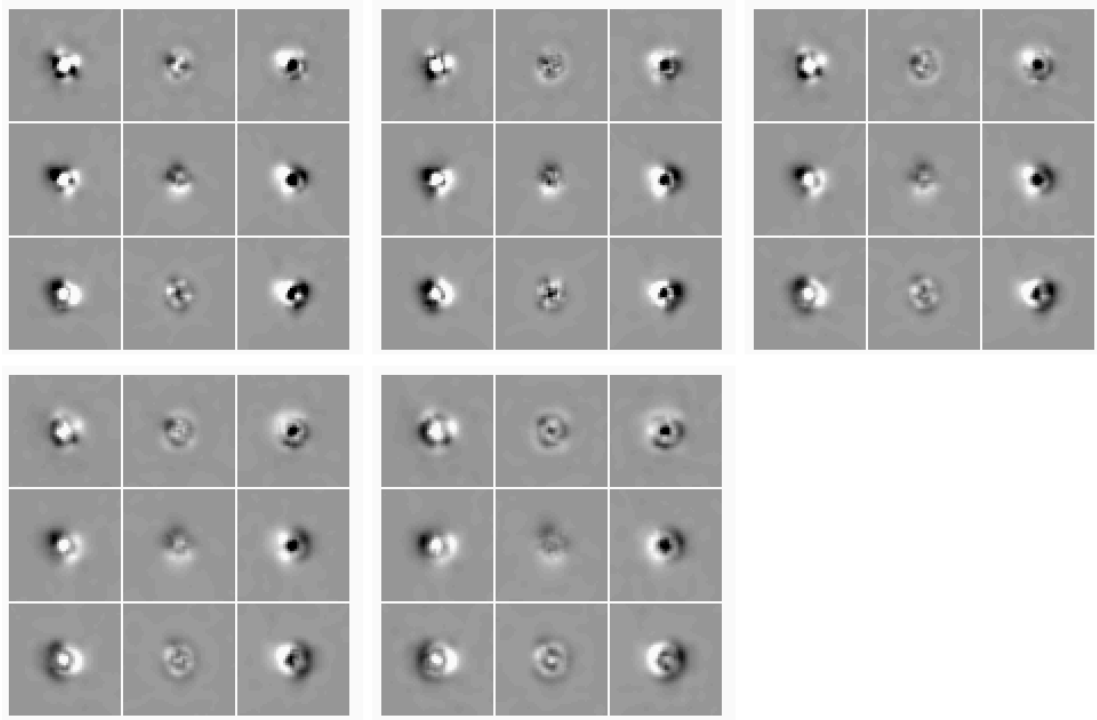

Figure 20: The spatial variation of the five PSFs under consideration, relative to the average for that filter. Top row: F105W, F110W, F125W. Bottom row: F140W and F160W. As in previous plots, dark represents more flux.

Figure 20 above shows how the PSFs for the five filters vary spatially across the detector. There are some similarities: most of the variation in all filters is from left to right. But there are some differences. The PSF in the lower middle is sharper than average for the shorter wavelengths but duller for the longer wavelengths, and that in the upper middle has the opposite behavior.

The data for the four Frontier Fields filters (F105W, F125W, F140W, and F160W) was taken over a period of about 2.5 hours in late 2013, so it is not clear how representative these PSFs are of the average PSF. It is encouraging that the F110W data, which were taken four years earlier (in 2009) of the 47 Tuc calibration field, appear to have PSFs that follow the general trends seen by the other four filters. This suggests that perhaps the WFC3/IR PSF does not change much with focus. We will explore this more fully in a future ISR. It does make sense, though, that the WFC3/IR PSF might be more stable than the WFC3/UVIS PSF, since the longer wavelength means that a given focus change (caused either by breathing or longer-term focus drift) has a smaller impact on the PSF.

We saw above for F105W that since our PSF-extraction routine assumes a pure flat-and-flush pixel response function (Π_{DET}), the photometry based on it contains some residual trends related to the pixel-phase of the star because the true pixel-response function is apparently not a perfect top-hat. We adjusted the F105W PSF so that it would produce unbiased photometry. It is worth exploring this issue for the other filters as well.

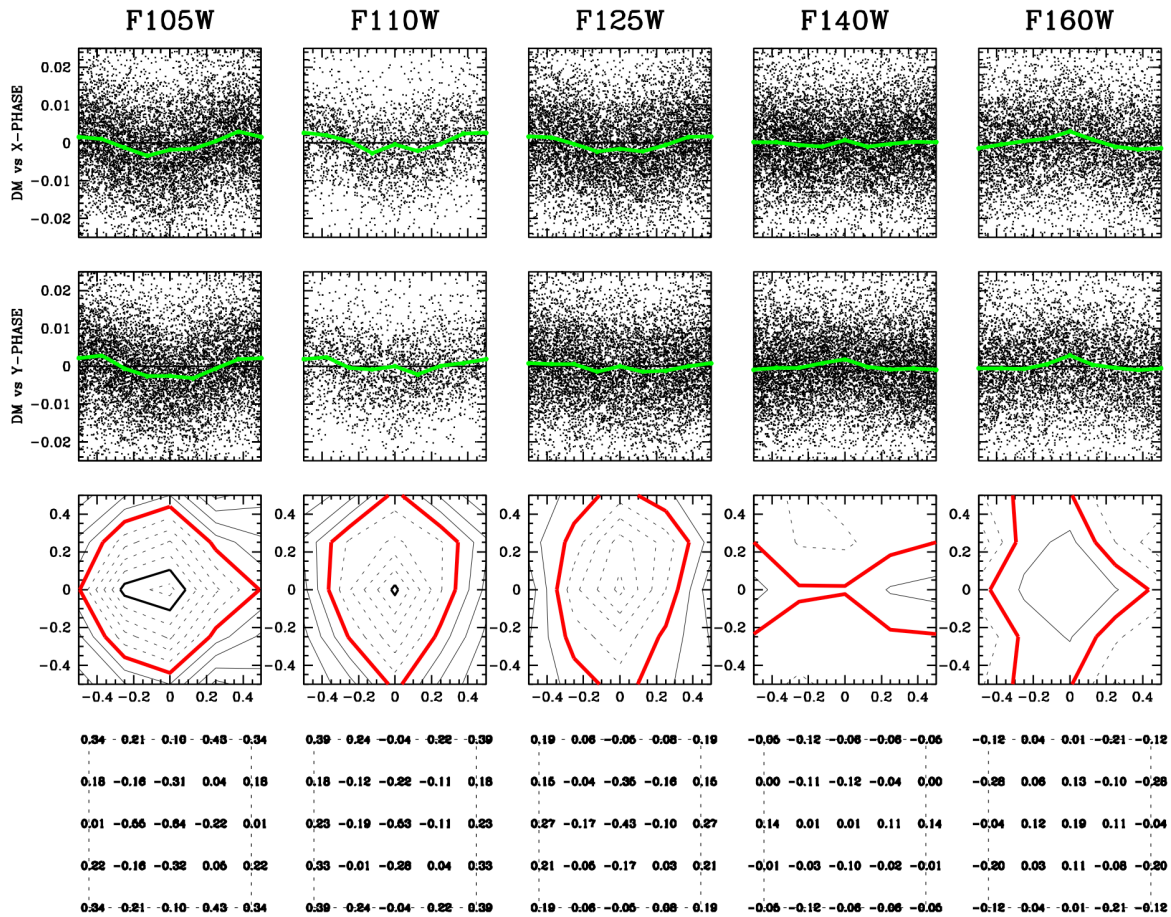

Figure 21: The photometric residuals as a function of pixel phase for the five filters under consideration here, from blue on the let to red on the right. The top two rows show the residuals as a function as a function of 1-d pixel phase against x and y. The third row shows the same data, but distilled into two dimensions as contour plots. Finally the bottom row reports the contour plots as numbers, showing 100× the magnitude residual. Recall that a negative magnitude residual corresponds to a positive excess of flux.

Figure 21 shows the magnitude residuals against *x* and *y* pixel phase for the five filters under consideration here. There is a clear trend with filter. Bluer filters have negative photometric residuals when the star is centered on the pixel, meaning that the centers of the pixels appear to be more sensitive to blue light than the rest of the pixel. Quantitatively, this trend goes from −0.64 for F105W to −0.53 for F110W to −0.43 at F125W, then it is essentially zero for F140W and slightly positive (+0.19) for F160W. (The slight difference between −0.64 here for F105W and what we saw above (−0.81) is just a statistical variation reflecting the fact that we used a slightly different sample of stars.)

We adjusted all the PSFs for the various filters for these QE variations as we did for F105W at the end of the previous section. In this way, the PSFs should be able to measure unbiased fluxes irrespective of the star's pixel phase. These adjusted PSFs are what we will provide to the community.

7 Storing the PSFs in a Standard Format

One of the goals of this exposition is to encourage HST users to start making use of FLT-based PSF models. The treatment here of the WFC3/IR PSF is the first, with more cameras to follow, including WFC3/UVIS and ACS/WFC. We chose to start with WFC3/IR since it suffers less from breathing, thanks to the typically longer wavelengths of light it observes. WFC3/IR is also the most undersampled of the current HST detectors, thus it is more important to have an accurate PSF model for it.

In an effort to help the community make use of these PSFs, we will be producing some software tools that use them for fitting stars in FLT images and for inserting stars into FLT images. These tools will be presented in future ISRs.

The PSF derived here will be stored in what we will call the "standard" PSF format. This is a simple 3-dimensional `real*4` `fits` image where the first two dimensions are 101 and 101, corresponding to the ×4-supersampled (Δx, Δy) domain extending from −12.5 to 12.5. The third dimension is `NAXIS3 =` $N_{\mathrm{PSFs}} = N_x \times N_y$, where N_x and N_y are the numbers of fiducial PSFs along x and y, respectively, and are stored in keywords `NXPSFs` and `NYPSFs`. There is space for up to ten fiducial PSFs along x and y, and the locations of these in the image frame are given by keywords `IPSFX01` through `IPSFX10` for x and `IPSFY01` through `IPSFY10`. The values of 9999 below simply mean that there is no fiducial PSF for that table value. The lines below show the typical header for a STDPSF format image for WFC3/IR:

```
SIMPLE  =                    T      IPSFX07 =                 9999
BITPIX  =                  -32      IPSFX08 =                 9999
NAXIS   =                    3      IPSFX09 =                 9999
NAXIS1  =                  101      IPSFX10 =                 9999
NAXIS2  =                  101      JPSFY01 =                    0
NAXIS3  =                    9      JPSFY02 =                  512
DATE    = '2016-03-31'              JPSFY03 =                 1014
TIME    = '17:12:14'                JPSFY04 =                 9999
BSCALE  =               1.0000      JPSFY05 =                 9999
BZERO   =               0.0000      JPSFY06 =                 9999
NXPSFs  =                    3      JPSFY07 =                 9999
NYPSFs  =                    3      JPSFY08 =                 9999
IPSFX01 =                    0      JPSFY09 =                 9999
IPSFX02 =                  512      JPSFY10 =                 9999
IPSFX03 =                 1014      COMMENT THIS PSF IS FOR WFC3/IR F105W
IPSFX04 =                 9999      COMMENT IT WAS CONSTRUCTED FROM PID-13606 DATA
IPSFX05 =                 9999      COMMENT OF THE OUTSKIRTS OF OMCEN ON DEC 13, 2013
IPSFX06 =                 9999      END
```

In general, we read the 101×101×`NAXIS3` PSFs (Ψ_{3D}) into the four-dimensional Ψ_{4D} array using: Ψ_{4D}(*,*,NXPSF,NYPSF) = Ψ_{3D}(*,*,N3), where N3 = NXPSF + (NYPSF-1)*NXPSFs. To use the PSF, we can either extract a full 101×101-gridpoint PSF appropriate for a given location — Ψ_{xy} as in E5 above — or we can use E6 to interpolate the entire four-dimensional array as we use it: $\psi(\Delta x,\Delta y;x,y)$.

Appendices B through C provide FORTRAN routines that perform the general interpolation given in E6. These one-page routines could very easily be translated into PYTHON or whatever code is needed.

8 CONCLUSIONS

The WFC3/IR detector is the most undersampled of the currently operating detectors on board HST. As such, it stands to benefit considerably from an accurate PSF model. The challenge of PSF modeling is that it must be done in the FLT frame, since that is the frame where pixels constitute direct, well understood constraints on the astronomical scene.

Working in this FLT frame is complicated, however, because of the need to do distortion and pixel-area corrections when working with dithered data sets. On account of these FLT-related complications, many HST users prefer to work in the rectified and combined `AstroDrizzle` frame, however this means they are unable to do precise modeling of shapes or optimal extractions for faint objects.

The PSFs presented in this ISR represent an attempt to make it easier for users to make better use of the FLT frame. There are a few ways to this. One is to use the BUNDLE software documented in Anderson 2014. This software makes use of accurate transformations among the exposures to map the pixels in the vicinity of a source of interest observed in multiple dithered exposures into a common reference frame so that PSF analysis can be performed on all images simultaneously. This approach will make full use of the FLT constraints.

Another way to use these PSFs would be to use the same transformations documented above to insert artificial stars into the FLT images, which can then be drizzled as usual. This would provide a set of reference stars in fields that are bereft of them, so that users can understand how the AstroDrizzle process has affected the profiles of stars. This can help with profile fitting, completeness and other applications. For instance, it is often complicated to drizzle extremely undersampled images since it can be hard to discern stars from cosmic rays and often the tops of stars get clipped off when the star profile is much sharper in an image where the star happens to be centered on a pixel than it is in an image where it is centered near pixel boundaries. Such a difference in sharpness can be erroneously flagged as a cosmic ray. A routine that can add stars to images can help verify that the drizzling process is not adversely affecting stars. A follow-up document will describe a simple software package designed to insert stars into FLT images.

We also aim to follow up on this ISR with a study of the behavior of the WFC3/IR PSF over time, to document how much the shape of the PSF might change with breathing, focus, and other observing circumstances. Finally, we also aim to provide PSFs for additional WFC3/IR filters using images from the archive.

REFERENCES

Anderson, Jay and King, Ivan R. 2000 PASP 112 1360. "Toward High-Precision Astrometry with WFPC2. I. Deriving an Accurate Point-Spread Function"

Anderson, Jay. 2014 ISR/WFC3 2014-24. "Local Pixel Bundles: Bringing the Pixels to the People"

Dressel, L. 2016, "Wide Field Camera 3 Instrument Handbook, Version 8.0" (Baltimore: STScI)

Lauer, Tod. 1999a PASP 111 227, "Combining Undersampled Dithered Images"

Lauer, Tod. 1999b PASP 111 1434, "Photometry of Undersampled Point-Spread Functions"

McCullough, P. 2008, WFC3/ISR 2008-19, "WFC3 TV2 TESTING: IR Intra-pixel Sensitivity"

Pavlovsky, McCullough, Baggett. 2011, WFC3/ISR 2011-19, "IR Intra-Pixel Sensitivity Variance"

Appendix A

The distortion solution we adopted is based on polynomials, but for convenience we have distilled it into images that are the same size of the detector. We bundle these images into a 5-extension `fits` file, a format that we refer to as "STDGDC" format (STandarD Geometric Distortion Correction).

The first extension is a 1014×1014 image that tells us the distortion-corrected x coordinate for each pixel. The second gives the distortion-corrected y coordinate. The image naturally contains only integer locations for the distortion solution, but bi-linear interpolation of this image is easily adequate to determine the distortion-corrected position (x_c, y_c) associated with a raw position (x_r, y_r). The third extension contains a 1014×1014 image that provides the pixel-area map, which accounts for the fact that on account of distortion the pixels do not all have the same projected area onto the sky. (In the presence of distortion, the flat fields can either preserve surface brightness or flux, and the HST pipeline convention is to preserve surface brightness, thus this correction is necessary if we want to convert our measured fluxes into calibrated fluxes.)

The fourth and fifth extensions provide the inverse distortion solution. These images are 1148×1015 pixels in size, corresponding to the entire extent of the distortion-corrected frame. Each pixel maps a particular location in the distortion-corrected frame back into the raw frame. Since we chose to constrain our distortion solution to have no correction for the central pixel of the detector, it often happens that the correction maps to negative (x_c, y_c) coordinates at the edges. We cannot of course have negative pixel locations in an image, so we use the `LTV1` and `LTV2` keywords to allow an offset. If we have a distortion-corrected location of (x_c, y_c), then the corresponding raw detector coordinate can be found by bi-linearly interpolating the 4th and 5th extension images at location: $x = x_c + (\texttt{LTV1} - 1)$ and $y = y_c + (\texttt{LTV2} - 1)$. Both the forward and the reverse distortion solutions are important in the second step of our iterative process.

The STDGDC-format distortion correction image for WFC3/IR[4] can be found on the same page that hosts the PSFs (http://www.stsci.edu/hst/wfc3/analysis/PSF). The file `STDGDC_WFC3IR.fits` contains the five-extension WFC3/IR correction. There are similar files for the UVIS detectors, but for these there is a different file for each filter, since those filters have been shown to introduce a fingerprint onto the distortion solution.

[4] Unlike WFC3/UVIS and the WFC cameras, which have separate solutions for each filter, we have only one solution that works for all filters. We have not explored the variation with filter, but any variation is likely less than 0.01 pixel.

Appendix B

```
c-----------------------------------------------------------------
c
c This FORTRAN routine will evaluate the PSF for a given offset
c at a given location in the image
c
      real function rpsf_phot_ij(dx,dy,iloc,jloc,psf_ij,
     .                           NXPSFs,XLIST_PSFs,
     .                           NYPSFs,YLIST_PSFs)
      implicit none

      real dx, dy
      integer iloc,jloc
      integer NXPSFs, NYPSFs
      real psf_ij(101,101,NXPSFs,NYPSFs)
      integer XLIST_PSFs(NXPSFs)
      integer YLIST_PSFs(NYPSFs)

      integer nx, ny
      real    fx, fy
      real    rpsf_phot
      integer i,j

      nx = 1
    1 continue
      if (iloc.gt.XLIST_PSFs(nx+1).and.nx.le.NXPSFs-2) then
         nx = nx + 1
         goto 1
         endif

      ny = 1
    2 continue
      if (jloc.gt.YLIST_PSFs(ny+1).and.ny.le.NYPSFs-2) then
         ny = ny + 1
         goto 2
         endif

      fx = 1.00*(iloc            -XLIST_PSFs(NX))/
     .          (XLIST_PSFs(NX+1)-XLIST_PSFs(NX))
      fy = 1.00*(jloc            -XLIST_PSFs(NY))/
     .          (XLIST_PSFs(NY+1)-XLIST_PSFs(NY))

      if (NXPSFs.eq.1) then
         fx = 0.0
         NX = 1
         endif
      if (NYPSFs.eq.1) then
         fy = 0.0
         NY = 1
         endif

      rpsf_phot_ij= (1-fx)*(1-fy)*rpsf_phot(dx,dy,psf_ij(1,1,NX  ,NY  ))
     .            + (1-fx)*( fy )*rpsf_phot(dx,dy,psf_ij(1,1,NX  ,NY+1))
     .            + ( fx )*(1-fy)*rpsf_phot(dx,dy,psf_ij(1,1,NX+1,NY  ))
     .            + ( fx )*( fy )*rpsf_phot(dx,dy,psf_ij(1,1,NX+1,NY+1))

      return
      end
```

Appendix C

```
c-----------------------------------------------------------------------------
c
c This routine will evaluate a PSF at a given (dx,dy) offset;
c * for the regions within 4 pixels of the center, it uses bi-cubic
c   interpolation
c * beyond this, it uses simple bi-linear interpolation
c
      real function rpsf_phot(x,y,psf)
      implicit none

      real x, y                ! delta x, delta y offset from the PSF center
      real psf(101,101)        ! the 101x101 PSF model to interpolate

      real listi_6(6)
      real listj_6(6)

      real    rx, ry
      integer ix, iy
      real    fx, fy
      real    dd
      integer i, j, iu, ju
      real    rspline_6

      rx = 51 + x*4
      ry = 51 + y*4
      ix = int(rx)
      iy = int(ry)
      fx = rx-ix
      fy = ry-iy

      dd = sqrt(x**2+y**2)

      rpsf_phot = 0.
      if (dd.gt.12.0) return

      if (dd.gt.4.0) then
         rpsf_phot = (1-fx)*(1-fy)*psf(ix  ,iy  )
     .             + ( fx )*(1-fy)*psf(ix+1,iy  )
     .             + (1-fx)*( fy )*psf(ix  ,iy+1)
     .             + ( fx )*( fy )*psf(ix+1,iy+1)
         return
         endif

      do j = 1, 6
         ju = iy + (j-3)
         do i = 1, 6
            iu = ix + (i-3)
            listi_6(i) = psf(iu,ju)
            enddo
         listj_6(j) = rspline_6(fx,listi_6)
         enddo

      rpsf_phot = rspline_6(fy,listj_6)

      return
      end
```

Appendix D

```
c-------------------------------------------------------------
c
c Given a list of pixels P1 through P6, the coefficients
c A, B, C, D are the spline fit between P3 and P4 that constrains
c the value of the function and its derivatives at P3 and P4.
c
c  f(x-3) = A + B*(x-3) + C*(x-3)**2 + D*(x-3)**3
c
c                                            (P1)
c  A =   1    ( 0   0   12    0    0    0)  (P2)
c  B = ---- * ( 1  -8    0    8   -1    0)  (P3)
c  C =  12    (-2  15  -28   20   -6    1)  (P4)
c  D =        ( 1  -7   16  -16    7   -1)  (P5)
c                                            (P6)
c
c This is a simple spline evaluation where the point of
c evaluation is located between pixels 3 and 4 at 3+dx
c
      real*4 function rspline_6(rx,x)
      implicit none

      real rx
      real x(6)

      integer i
      real    f
      real    A, B, C, D

      if (rx.lt.3.or.rx.gt.4)
     .    stop 'rspline_6 designed for rx between 3 and 4'

      i = int(rx)
      f = rx-i

      A =                     12*x(i)
      B =  1*x(i-2)-08*x(i-1)        +08*x(i+1)-01*x(i+2)
      C = -2*x(i-2)+15*x(i-1)-28*x(i)+20*x(i+1)-06*x(i+2)+01*x(i+3)
      D =  1*x(i-2)-07*x(i-1)+16*x(i)-16*x(i+1)+07*x(i+2)-01*x(i+3)

      rspline_6 = (A + B*f + C*f**2 + D*f**3)/12.00

      return
      end
```