SPACE TELESCOPE SCIENCE INSTITUTE

Instrument Science Report NICMOS 2009-004

NICMOS Photometric Calibration Pipeline

A. C. Viana, R. S. de Jong, A. M. Koekemoer
June 29, 2009

ABSTRACT

We present the details of the NICMOS Photometric Calibration Pipeline used to generate aperture photometry. We present an overview of the different corrections employed, explain components of the pipeline and also outline the types of photometric and statistical data products created. In de Jong et al. (in preparation) this photometry will be used to determine the photometric calibration constants stored in the PHOTTAB reference files.

Introduction

The goal of the NICMOS photometric calibration program is to create accurate tables containing a linear fit to the sensitivity of the instrument as a function of temperature for each camera and filter combination. This is necessary to correct for the temperature dependent nature of the NICMOS detector (Bergeron, 2009, ISR in preparation). Specifically, calibration is accomplished through the creation of FITS reference tables (`*_pht.fits`). The location and name of the reference tables are stored in the PHOTTAB header keyword in the FITS image header. The `calnica` task in Pyraf performs the photometric correction using these reference tables and in the process populates the PHOTFLAM header keyword with an appropriate value corresponding to the camera, filter and temperature for that exposure. There is a separate reference file for before and after the installation of the NICMOS Cooling System (NCS) to account for the change in temperature regime. All these reference files were created by Roelof de Jong using the output of the NICMOS photometric calibration pipeline and are available on the

WWW at:

http://www.stsci.edu/hst/observatory/cdbs/SIfileInfo/NICMOS/NICPhot

Observations

The pipeline runs over a database of files from the NICMOS photometric calibration program. This database contains over 5,000 pre-NCS and 8,000 post-NCS exposures from 28 proposals containing observations of 19 different standard stars taken from 1997 through 2008. These proposals were predominately PI'ed by Roelof de Jong and contain primarily observations of P330E (a Solar analog star) and G191B2B (a white dwarf) which are well understood stars useful for photometric calibration (Bohlin et al 2005, NICMOS ISR 2005-002). The following tables gives the details of the proposal IDs used to generate this dataset:

Table 1: pre-NCS proposal IDs used to generate the calibration and their observation dates.

Proposal ID	Dates	Stars
7034	04/08/97 – 04/09/97	P330E
7049	04/29/97 – 06/08/97	P330E, OPHS1, P177D
7050	04/25/97 – 05/25/97	P330E
7055	04/22/97 – 04/30/97	G191B2B, P035R
7152	05/22/97 – 05/23/97	P330E, P177D
7607	06/12/97 – 09/21/97	P330E, P177D
7691	08/05/97 – 08/07/97	P330E, G191B2B, OPHS1
7693	09/23/97 – 01/31/98	P330E, P177D
7696	01/12/98 – 01/15/98	P330E, G191B2B
7816	01/14/98 – 01/18/98	P330E, G191B2B, BRI0021, P177D, CSKD12, GD153
7902	02/09/98 – 05/29/98	P330E, P177D
7904	12/16/97 – 12/26/97	P330E, G191B2B, BRI0021, P177D, CSKD12, GD153
7959	06/07/98 – 06/18/98	P330E

Table 2: post-NCS proposal IDs used to generate the calibration and their observation dates.

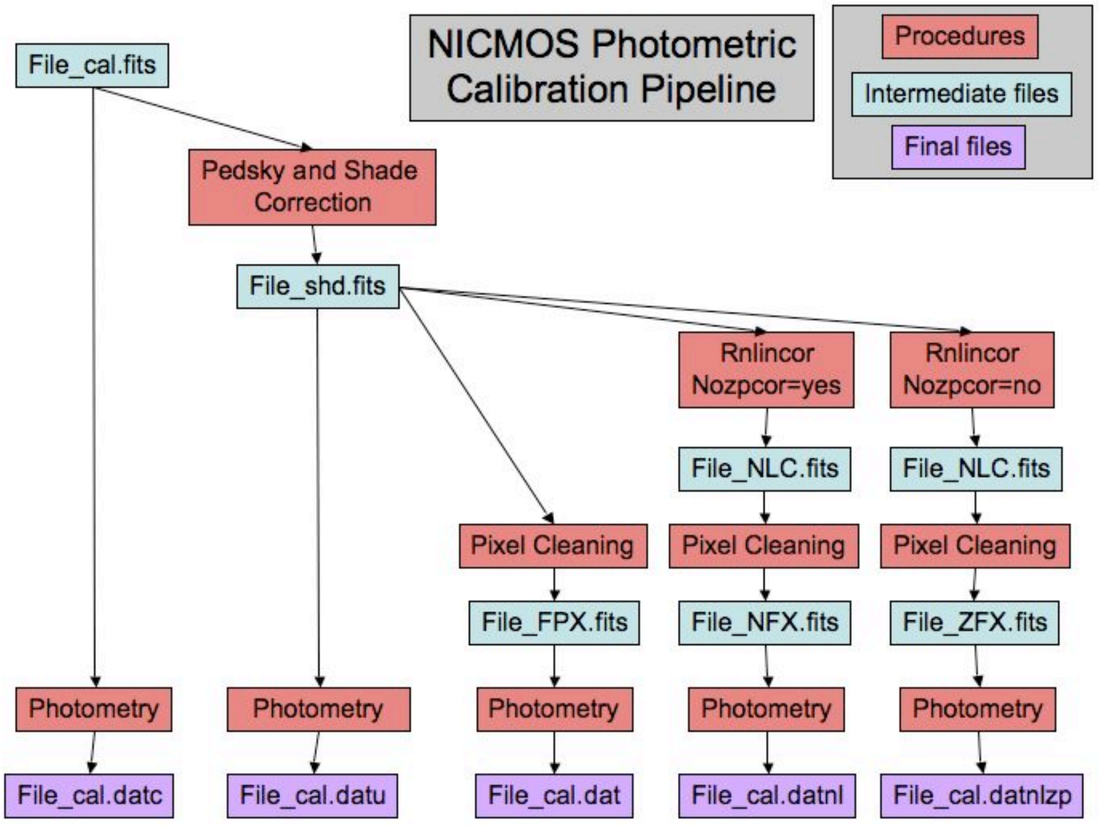
Proposal ID	Dates	Stars
8986	05/31/02 – 06/07/02	P330E, G191B2B
9325	07/20/02 – 08/14/02	P330E
9639	09/14/02 – 01/14/03	P330E
9995	12/01/03 – 07/12/04	P330E
9997	10/27/03 – 03/17/04	P330E, G191B2B, GD71, P177D
9998	09/05/03 – 01/19/04	P330E, G191B2B, GD71, P177D, SNAP7, GD153, P041C
10381	11/09/04 – 07/08/05	P330E, G191B2B
10446	02/21/05 – 02/21/05	P330E
10454	02/07/05 – 04/23/05	BD17D4708, SNAP2, WD1657343
10725	11/14/05 – 07/14/06	P330E, G191B2B
10726	11/07/05 – 11/17/05	GL390
11060	11/15/06 – 07/31/07	P330E, G191B2B
11061	10/10/06 – 10/12/06	2M003618, GD71, P177D, WD1057719, GD153, P041C, VB8, WD1657343
11319	04/05/08 – 07/01/08	P330E
11826	09/05/08 – 09/05/08	G191B2B

Pipeline Overview

The photometric calibration pipeline is a series of Python and CL scripts that performs image corrections, generates aperture photometry and provides statistical summaries. The software for the pipeline was developed by Roelof de Jong, Megan Sosey and Alex Viana over the course of several years. It calls several standard Pyraf tasks including RNLINCOR, MINMAX, PEDSKY, FIT1D and PHOT. All images were processed using the latest version of `calnica` (4.4.0) prior to processing using the photometric pipeline.

Figure 1 traces a single `*_cal.fits` file through all the steps of the pipeline. As the figure shows the pipeline creates five different types of output files for each input file. Four of these file types, `.datc, .datu, .dat` and `.datnlzp`, are only used to verify the effectiveness of each of the corrections. The fifth extension, `.datnl`, is the only extension used to generate the reference files.

Figure 1: A diagram of the NICMOS Photometric Pipeline. Note that the one input file goes through all the branches of the diagram and produces five different outputs.



Corrections

The Pipeline performs three corrections for effects that are not completely removed in the standard `calnica` pipeline. This is done to ensure that these effects do not affect the corrections in the PHOTTAB reference file. As shown in Figure 1 the effects removed in the pipeline are:

- The pedestal effect
- The shading effect
- The countrate-dependent nonlinearity

Additionally, a pixel cleaning algorithm is applied to remove any bad pixels on the source.

Shading Removal

Shading is a noiseless pixel-dependent bias which takes the form of a ripple from the first to last pixel in the direction of pixel clocking during readout. Physically, it is caused by the rapid heating for the readout amplifier during readout in each quadrant with an amplitude of up to several hundred electrons in NIC2 and smaller for NIC1 and NIC3. The shading depends primarily on the time interval since the last readout of a pixel. For more information see the NICMOS Data Handbook Section 4.2.2.

The shading effect is already corrected by the `calnica` software but the photometric pipeline performs an additional subtraction for uncorrected residuals. The effect is corrected using the FIT1D task in Pyraf. This tasks creates a fit to each row or column of the image. In this case the output image is the difference between the original and the 1D fit to remove the shading. FIT1D is run using the parameters in Table 3.

Table 3: Summery of the parameters used to run FIT1D.

Parameter	**Discription**
Type = “difference”	Sets the output image to be the difference of the image and 1D fit.
Axis = 2 (or 1)	Sets the 1D fits to be taken along the image columns (2) or rows (1).
Sample = sample	Gives the input list of variable to be use such that that coronographic hole is omitted.
Naverage = -3	Sets the number of points to be combined into a fitting point to be 3. The negative sign sets to combining method to the median.
Function = Chebyshev	Sets the fitting function to by a chebyshev polynomial.
Order = 1	Set the order of the fitting function to be 1st order.
Low_reject = 5	Rejection limit below the fit in units of residual sigma.
High_reject = 5	Rejection limit above the fit in units of residual sigma.
Niter = 4	Number of rejection iterations.

Pedsky Correction

The pedestal correction corrects for a time-variable bias, which is usually uniform across each quadrant of NICMOS but different in each quadrant. From readout to readout each quadrant can exhibit a range of variability in the amplitude of this effect. In order to correct for this effect the bias must be determined independent of the sky and background level present and then removed. This removal is accomplished using the PEDSKY task in Pyraf with the parameters show in Table 4.

Table 4: Summery of the parameters used to run PEDSKY.

Parameter	Discription
Salgori = “iter”	Sets the output image to be the difference of the image and 1D fit.
Rmedian-	Set the 1D fits to be taken along the image columns.

Countrate-Dependent Nonlinearity Correction

NICMOS suffers from a countrate dependent non-linearity which influences accurate photometry, especially at the shorter wavelengths. This anomaly means that the ratio of the measured flux to the incident flux is not strictly 1:1, nor is it linear as a function of incident flux. As a result the level of measured flux must be corrected to reflect the actual incident flux. This correction is accomplished using the RNLINCOR task in Pyraf.

However, since the ratio of incident to measured flux is a unitless quantity, the actual flux values have to be taken into account to produce a useful calibration. In doing this we take advantage of the fact that both our standard calibration stars (P330E and G191B2B) have well-understood and nearly-identical flux levels. Thus, we take the mean flux of these two stars as our "Zero-Point" and calibrate our countrate correction around that value. Using the mean of the two stars introduces a small but acceptable amount of error into our final calibration. More information about this anomaly can be found at:

http://www.stsci.edu/hst/nicmos/performance/anomalies/nonlinearity.html

Pixel Data Quality and Sigma Clipping

The photometry pipeline contains a routine to replace pixels with "bad" data quality values or extreme flux values with the local average in order to prevent those pixels from affecting the photometry in the event that they occur within the photometric aperture. The following is the correction algorithm employed:

1. The routine first flags all the pixels in the input `*_cal.fits` file that do not have a data quality tag found in Table 5. Flagged pixels mainly consist of bad (hot/cold) pixels with a data quality value of 32.

Table 5: Accepted (unflagged) FITS data quality bit values and descriptions. All other values will be flagged for correction.

Data Quality Value	Description
0	No Know Problems
64	Saturated Pixel
2048	Signal in the 0th Read
2112	Both 64 and 2112

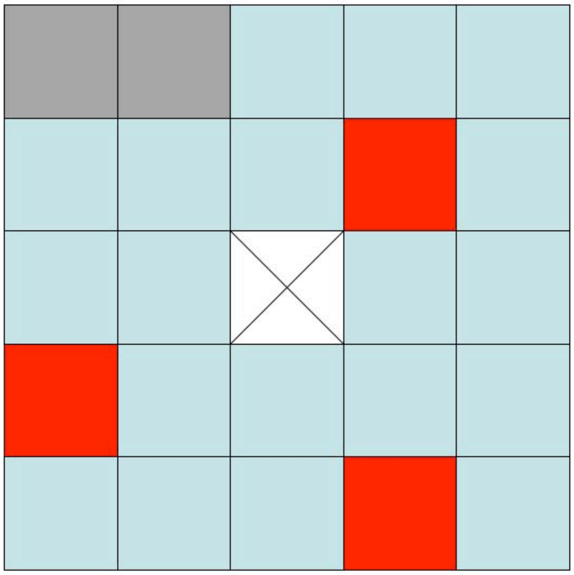
2. Next, the central pixel value of each unflagged pixel is compared with the average and RMS of its 24 nearest neighbors, e.g. a 5x5 grid but excluding the central pixel, as show in Figure 2. This is done excluding the neighbor pixels that were also flagged for their data quality values. Specifically the central pixel must satisfy the following "sigma clipping" condition or it will also be flagged:

$$|\text{pixel value - local average}| \leq \text{tolerance} \bullet \text{local RMS}$$

Where the local average and the local RMS are computed over the same 5x5 grid in Figure 2. The tolerance level is set to 5 for NIC1 and NIC2 exposures and 16 for NIC3 to account for the NIC3 camera's larger pixel field of view which makes that camera highly undersampled. A minimum of 2 other pixels in the local neighborhood must be “good” (unflagged) for the pixel to be flagged. Otherwise it is left unflagged due to a lack of information. In practice however, flagged pixels are sufficiently scattered to prevent this from occurring.

3. Once all the data quality and sigma pixels have been flagged these flagged pixels are replaced with average of the local 5x5 grid. This replacement average is calculated excluding any pixels flagged for data quality and sigma rejection so that the effects from these pixels do not affect the average. Again, the criteria of at least 2 good pixels in the local 5x5 grid must be met for the replacement with the local average to occur.

Figure 2: This is an example of the pixel cleaning algorithm at work on a 5x5 region. The center pixel is the one being evaluated. The 2 grey pixels did not pass the data quality condition (step 1) and the 3 red pixels did not pass the sigma condition (step 2). If the central pixel were flagged for not passing the data quality test or sigma clipping test it would be replaced with the average of the 19 unflagged light blue pixels (step 3).



Photometry

Once all the corrections have been applied aperture photometry is performed using the PHOT task in Pyraf with the parameters in the following tables:

Table 6: PHOT parameters in the PHOTPARS Pyraf package

Parameter	Description
coords=coofile	The file containing the coordinates of the star.
weighting="constant"	All pixels have the same weight

Table 7: PHOT parameters in the DATAPARS Pyraf package

Parameter	Description
readnoi=rdnoi	The noise model used
gain="adcgain"	The image header keyword defining the gain parameter whose units are assumed to be electrons per adu.
scale=1.0	The scale of the image in user units, e.g arcseconds per pixel. All APPHOT distance dependent parameters are assumed to be in units of scale. If scale = 1.0 these parameters are assumed to be in units of pixels.
exposure="samptime"	The image header exposure time keyword.

Table 8: PHOT parameters in the CENTERPARS Pyraf package

Parameter	Description
calgorithm=calg	Sets the centering algorithm to “centriod” for standard execution and “none” if recentering is performed.
cbox=7	The width of the subraster used for object centering in units of the DATAPARS scale parameter.
maxshift=2.5	The maximum permissible shift of the center with respect to the initial coordinates in units of the scale parameter.

Table 9: PHOT parameters in the FITSKYPARS Pyraf package

Parameter	Description
annulus=annul	The inner radius of the annular sky fitting region in units of the DATAPARS scale parameter.
salgori="median"	Compute the median of the sky pixel distribution. This algorithm is a useful for computing sky values in regions with rapidly varying sky backgrounds and is a good alternative to "centroid".
skyvalu=0	The constant for constant sky subtraction.
dannulus=dannul	The width of the annular sky fitting region in units of the DATAPARS scale parameter.
aperture=aper	Set the radius of the photometric aperature.

Table 10: PHOT camera dependent values.

Parameter	Value		
	NIC1	NIC2	NIC3
Aperture	11.5	6.5	5.5
Annulus	46.5	26.5	20.5
Dannulus	5.0	5.0	5.0
rdnoise	26.0	26.0	30.0

Each star position is automatically determined using a custom script called FINDMAX which creates an output coordinate file use by subsequent steps in the pipeline. FINDMAX essentially uses the IRAF MINMAX task to locate the star in each image but avoids the coronagraphic hole in NIC2. In the rare situations where FINDMAX fails the coordinate files have been manually generated to correctly input the location of the star.

PHOT uses these coordinates as the center of the photometric aperture. The sky value is calculated from a concentric annulus also centered on the input coordinates. The radius of the aperture, and the inner and outer radii of the sky annulus are all set by the keywords in Table 10.

Once the photometry is completed all the data is gathered into tables containing the outputs of PHOT for analysis. The details of how this analysis is performed can be found in de Jong et al. 2009 (NICMOS ISR, in preparation).

Statistics

As a final feature the pipeline calculates the error associated with each observation. It does this by breaking down the total error given by PHOT into different components and then further calculating the flatfield error. The purpose of this is to determine the contribution of each component to the total error and thereby identify possible areas for improved calibration.

Typical random noise on an image is modeled as Poisson noise and goes as the square root of the number of observations (electrons). However, since there is usually a gain factor we would normally multiply the number of ADU by the gain to return the number of electrons. PHOT performs all these calculations, including a gain correction, based on the assumption that the flux units are given in ADU. However, NICMOS images record flux as ADU/s. As a result the components will still add up to the total error but will not be in the same ADU/s units as the image. The flatfield error however will be in units of ADU/s because it is calculated outside of the PHOT task.

The error components of the total error are calculated using 5 different variables:

- Gain - The electrons per ADU (called “epadu” in PHOT).
- Area - The area of the aperture in pixels.
- Stdev - The standard deviation of the best estimate of the sky value per pixel.
- Nsky - The number of sky pixels.
- Flux - The total flux in the aperture (excluding the sky flux).

From these five variables four different statistics are calculated. The first three statistics as well as all five variables can be found in the help file for the Pyraf/IRAF task PHOT. All of these variables are generated using the PHOT task. Note that the PHOT task pulls the gain value from the ADCGAIN header keyword.

1. The object error appropriate for flux in units of ADU/s due to Poisson noise from the object is given by:

$$\sigma_{object} = \frac{\sqrt{flux(ADU/s) \bullet gain(e^{-}/s)}}{gain(e^{-}/s)} = \sqrt{\frac{flux}{gain}}$$

2. The sky flux error is the error in the sky value per pixel scaled to the number of pixels in the aperture and is given by:

$$\sigma_{Sky\_Flux} = \sqrt{area \bullet stdev^2}$$

3. The variance in the sky value per pixel scaled to the aperture area and then divided by the number of pixels used to calculate the sky gives the sky estimate error:

$$\sigma_{Sky\_Estimate} = \sqrt{area^2 \bullet \frac{stdev^2}{nsky}}$$

   This measures the error introduced by subtracting the sky flux from the flux in the aperture. Again, note that these three statistics when summed in quadrature are exactly equivalent in value and computation to the "error" output in the PHOT task.

4. Lastly, the flatfield error is calculated using the error array in the flatfield associated with each image and used to process that image in `calnica`. This error (σ_{flat}) is given as an error per flux unit and the individual pixels errors add in quadrature. Thus the equation for the flatfield error over the area of the aperture is given by:

$$\sigma_{Flat\_Error} = \sqrt{\sum_{area} (\sigma_{flat} \bullet flux)^2}$$

References

- Bergeron et al 2009, NICMOS ISR in preparation
- Bohlin et al 2005, NICMOS ISR 2005-002
- Thatte, D. and Dahlen, T. et al. 2009, “NICMOS Data Handbook”, version 8.0, (Baltimore, STScI)
- De Jong et al 2009, NICMOS ISR in preparation