Wide-field Infrared Survey Explorer

Subsystem Design Specification

Instrumental Calibration (ICAL)

Version 6.8, 15-February-2010

Prepared by: Frank Masci

Infrared Processing and Analysis Center
California Institute of Technology

WSDC D-D018

Concurred By:

Roc Cutri, WISE Science Data Center Manager

Tim Conrow, WISE Science Data Center Lead Architect

Frank Masci, WISE Science Data Center Cognizant Engineer/Scientist

Stefanie Wachter, WISE Science Data Center Instrument Calibration Scientist

Deborah Padgett, WISE Science Data Center Instrument Calibration Scientist

Revision History

Date	Version	Author	Description
September 24, 2007	1.0	Frank Masci	Initial Draft
October 12, 2007	1.1	Frank Masci	- preserve order of keywords when propagating from input FITS HDU to output HDU
January 4, 2008	1.3	Frank Masci	- 8-bit input ground mask, - update and propagate 31 bit proc mask, - xy history qa, - lot more qa metrics, rowcol-cut plots
March 13, 2008	1.5	Frank Masci	- implemented all QA metrics according to SIS
March 25, 2008	1.6	Frank Masci	- included parameter to rescale final uncert images using fudge factor
June 2, 2008	1.9	Frank Masci	- implemented SVG for qa plots
August 13, 2008	2.1	Frank Masci	- implemented amplifier channel for w1,w2 and quadrant statistics for w3,w4, - wrote to metadata table, and generated two new SVG plots
September 3, 2008	2.3	Frank Masci	- implemented command-line inputs: -gainmap, -rdmap; i.e., gain and read-noise calibration FITS files to over-ride previous single values
September 18, 2008	2.4	Frank Masci	- removed -coeffs and replaced with input meta file: -metatbl. Coeffs and other cal params will come from meta table
September 19, 2008	2.6	Frank Masci	- implemented quadratic non-linearity model
October 21, 2008	3.0	Frank Masci	- updated to read LSB trunc and uncscal parameters from simcal-meta.tbl; - also updated error model to use trunc factors
December 12, 2008	3.2	Frank Masci	- added -reref switch to not

			remove reference-pixel signal, but still strip reference pix borders at end
January 11, 2009	3.3	Frank Masci	- included droop correction per quadrant for w3 and w4 with unc propagation. Based on SDL's initial characterization
January 27, 2009	3.4	Frank Masci	- implemented use of low-frequency responsivity maps with uncertainty propagation - included 2^T bit truncation factor for DEB output in non-linearity calibration
February 14, 2009	3.5	Frank Masci	- implemented single frame outlier/spike detection and new QA metrics
February 22, 2009	3.6	Frank Masci	- implemented two methods to compute local robust pseudo-chi2 metrics: uncLocScal1,2 written to QA metatable. - also optionally and dynamically rescale prior uncertainties if -uncflg set using method specified by -methscal <*method*>
February 24, 2009	3.7	Frank Masci	- compute new QA metric: intRatMedRange, redefined: intI16_84Range,intMed16ptile, intMed84ptile
February 24, 2009	3.8	Frank Masci	- fixed dreadful bug in outlier detection subroutine; - added final level-1a values to pixel history QA file
March 23, 2009	3.9	Frank Masci	- optional gzipping of final msk and unc values and optional prior precision reduction of unc values by truncating *n* significant figs so will gzip better; - new params: -sigfig, -trcomp, -gzcomp implemented
March 24, 2009	4.0	Frank Masci	- only apply droop correction [using ground algorithm] if frac of sat pixels are below some tolerance specified by *satthres* param
March 24, 2009	4.1	Frank Masci	- added CL options -fm and

			-fltbits for dynamically/on-orbit generated flat-field mask and fatal template bitstring. Latter used to transfer fatal bits to unreliable flat-field bit in output frame mask
April 21, 2009	4.2	Frank Masci	- assume input linearity cals include sum coeff factors; - if discrim < 0 or if badcal indicated in static bit 6, do not linearize input pixel; - implemented linear quadratic extension for mobs > mobsmax where mobsmax is read from cal-meta.tbl file
May 16, 2009	4.3	Frank Masci	- implemented alternative uncertanty model tuned for May 2009 Ned sims onwards with new params read from meta-ical.tbl; new model referred to uncmod=2; old model=1
May 18, 2009	4.4	Frank Masci	- removed ITUT associated metrics: ratio_rmsmadlow_trim, med_trim, rmsmad_lower_trim from meta table output since too slow
May 27, 2009	4.5	Frank Masci	- qa on ref pixels for w3,w4 too, even though no corr algorith avail; - put conditional proc to not generate all QA if -saveflt,-savelin specified unless -allqa also spec; - force skyoff=0 if in -saveflt,-savelin mode unless -allqa spec; - perform thresholding/filtering using thres's in meta-cal file to support dynacal; write usability flags to hdrs and qa meta fil
June 5, 2009	4.6	Frank Masci	- stop appending output to output QA meta-data table file; - add -w(band) to output file name; explicitly write product frame names to table header

June 26, 2009	4.9	Frank Masci	- read FPA temp telem. keyword for dynacal; check for anneal and temp keyword existence and value ranges; - print warnings on which conditions fail dynacal filtering
July 11, 2009	5.0	Frank Masci	- pass 0 as the type when reading FITS values with getpdl(), i.e., as represented by input BITPIX and not floats; - reset num pixels to 1000 if num=0 to avoid QAPlotter from choking;
July 16, 2009	5.2	Frank Masci	- write scanId, frmNum to output metadata table; - don't blindly filter on all non-zero masked pixels when computing QA stats for int and unc frames
July 31, 2009	5.5	Frank Masci	- set NaNs in int and unc frames for fatal pixels set in mask; - properly scan for temp and anneal keys under svlin and svflt modes
August 5, 2009	5.6	Frank Masci	- allow for gzipped L0 input FITS images
September 1, 2009	5.7	Frank Masci	- included new CL param -ksiz and meta.tbl parameter ksize to control size of median filter for single frame outlier detection; previously hardcoded at 3
September 14, 2009	5.8	Frank Masci	- included new input meta.tbl param: rnflg to use read-noise map in DEB DN units directly for error model 1; also removed EXPTIME, TSAMP since not used and also irrelevant error checking
September 28, 2009	5.9	Frank Masci	- added DTANNEAL and FPA temp values to output QA metadata files so Heidi can implement ticket 224
October 6, 2009	6.2	Frank Masci	- changed negative DTANNEAL error checking to "warn" instead of "die"
December 31, 2009	6.3	Frank Masci	- nonlinearity calibration had a

			bug that caused large DN values (close to dynamic range) to be turned into NaNs because discriminant in quadratic solution came out negative. These rouge NaNs wrecked havoc in glitch detector downstream, whereby -NaNs (yes, with a negative flavor) were causing bit 31 (the sign bit) to be erroneously turned on in the frame masks
January 8, 2010	6.4	Frank Masci	- propagate input cal dir used to output FITS headers
January 11, 2010	6.5	Frank Masci	- avoid choking when whole image is NaN'd, e.g., w3, w4 due to anneals, the plan is still to have all saturated pixels NaN'd at the end. Dummy histograms (with dummy in their title) will be made if this occurs; - also, fixed a bitwise precedence warning from PDL
January 15, 2010	6.6	Frank Masci	- prelim quad-to-quad droop correction with new params: gfrac and refbase* read from wise-meta.tbl. Also disabled call to old (ground) droop algorithm
January 15, 2010	6.7	Frank Masci	- testing of droop algorithm revealed code had to be rearranged, optional droop refinement is now performed at the end
February 12, 2010	6.8	Frank Masci	- implemented split-droop and residual split-banding corrections

Table of Contents

1. INTRODUCTION

1.1 Purpose and Scope

This Subsystem Design Specification (SDS) document describes the basic requirements, assumptions, definitions, software-design details, algorithms, QA, and necessary interfaces for the Instrumental CALibration (ICAL) subsystem of the WISE Science Data System (WSDS). It will be used to trace the incremental development of this subsystem, and contains sufficient detail to allow future modification or maintenance of the software by developers other than the original developer. This document is an evolving document as changes may occur in the course of science instrument hardware design, characterization and maturity of operational procedures.

The purpose of ICAL is to read in a single band image-frame and correct it for instrumental signatures, e.g., dark current, non-uniform responsivity, non-linearity, electronic and other "detector-based" signatures, and residual bias and/or gain signatures. By "detector-based", we mean artifacts associated with read-out from the arrays. Artifacts and signatures induced by telescope optics (e.g., ghosting, glints, distortion, PRF non-isoplanicity), as well as image persistence (latents) are discussed in other SDS documents. ICAL also initiates a prior-uncertainty model for each frame pixel for propagation and updating downstream as calibrations are applied. A "frame-processing" bit-mask is also initiated and updated. This stores the status of processing on each pixel as well as prior knowledge of the location of bad-pixels. The primary outputs from the ICAL pipeline are a calibrated intensity frame, uncertainty frame, mask image, QA metrics and plots.

1.2 Document Organization

This document is organized along the major themes of Requirements; Other Software Interfaces; Assumptions; Functional Descriptions and Dependencies; Input/Output; Algorithm Descriptions, Quality Assurance, and Testing.

The material contained in this document represents the current understanding of the capabilities of the major WISE systems and sub-systems. Areas that require further analysis are noted by TBD (To Be Determined) or TBR (To Be Resolved). TBD indicates missing data that are not yet available. TBR indicates preliminary data that are not firmly established and subject to change.

1.3 Applicable Documents

- WSDC Functional Requirements Document, *WSDC D-R001* (FRD – Level 4 Requirements):
http://web.ipac.caltech.edu/staff/roc/wise/docs/WSDC_Functional_Requirements_all.pdf

- WSDS Functional Design Document, *WSDC D-D001* (FDD):
http://web.ipac.caltech.edu/staff/roc/wise/docs/WSDS_FDD_v1.pdf

- WSDC Science Data Quality Assurance Plan, *WSDC D-M004* (QAP):
  http://web.ipac.caltech.edu/staff/roc/wise/docs/QA_Plan_WSDC_2007-03-01.pdf

- Software Interface Specification (SIS), *WSDC D-I101* – Frame Processing Mask:
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/InstruCal01.txt

- Software Interface Specification (SIS), *WSDC D-I105* – Level-1a frame QA metadata:
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/QAoutput_icl04.txt

- Software Interface Specification (SIS), *WSDC D-I137* – "Dynamic flats" QA metadata:
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/QAoutput_icl06.txt

- Specifications for calibration receivables, *WSDC D-I005*:
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/GroundCalSpecs.pdf

- WISE Digital Electronics Box (DEB) processing description (SDL/06-070; Jan 2006):
  http://web.ipac.caltech.edu/staff/roc/wise/docs/sdl06-070-.pdf

- Instrumental Calibration Peer Review (09/28/2007):
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/InstruCal_PeerReview.pdf

- Instrumental Calibration Critical Design Review (01/29/2008):
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/InstruCal_CDRJan08.pdf

- Science Team Meeting presentation (01/08/2009):
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/ICal_STmtgJan09.pdf

- Science Team Meeting presentation (01/28/2010):
  http://wise2.ipac.caltech.edu/proj/fmasci/ICal_STmtgJan10.pdf

- The Droop Effect: to correct or not correct:
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/Droop_question.pdf

- Droop and its Manifestations:
  http://wise2.ipac.caltech.edu/proj/fmasci/droop.html

- Proposed Raw-Pixel Error Model, *WSDC D-T001*:
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/WiseErrorModel.pdf

- Subsystem Design Specification (SDS), *WSDC D-D011* – "tempcal":
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/sds-tempcal.pdf

- Subsystem Design Specification (SDS), *WSDC D-D012* – "flatcal":
  http://web.ipac.caltech.edu/staff/fmasci/home/wise/sds-flatcal.pdf

- Subsystem Design Specification (SDS), *WSDC D-D021* – “compflat”: *http://wise2.ipac.caltech.edu/staff/fmasci/sds-wsdc-D021-compflat.pdf*

- A Comparison of Responsivity Estimation Methodologies: *http://wise2.ipac.caltech.edu/staff/fmasci/FlatTesting.html*

- Dynamic Calibration Plans: *http://wise2.ipac.caltech.edu/staff/fmasci/dynacal_plans.txt*

1.4 Requirements

Below we summarize the requirements pertaining to ICAL goals and processing. These are from the WSDC Functional Requirements Document (§1.3).

• *L4WSDC-042*: The WSDS Pipeline processing shall remove the instrumental signature from Level 0 image frames.
• *L4WSDC-037*: The WSDC Pipelines subsystem shall convert raw WISE science and engineering data into calibrated images and extracted source lists from which the preliminary and final WISE data products will be derived.
• *L4WSDC-039*: Within 3 days from receipt of a given data set at the WSDC all data shall be processed through the WSDS Scan/Frame pipeline which performs basic image calibration and source extraction from on images from individual orbits. The results of this processing step shall be Level 1 source extractions and image data, which are loaded into the WISE Level 1 extracted Source Working Database (L1WDB) and Image Archive allowing access by the WISE Science Team for external quality assessment.
• *L4WSDC-024*: The WSDC shall generate and maintain an archive of the calibrated, single epoch WISE images for the duration of the project for use by the Project Team. The purposes of this archive are quality assurance, transient analysis and moving object identification. Self-derived Demonstration Define duration of project.
• *L4WSDC-060*: The WSDC archive shall provide a web-based interface to enable selection, display and retrieval of any or all single-epoch images and combined Atlas Images based on position or time of observation for the purpose of quality assurance, validation and analysis. The goal shall be to also allow image selection on any image metadata parameter.
• *L4WSDC-012*: Flux measurements in the WISE Source Catalog shall have a SNR of five or more for point sources with fluxes of 0.12, 0.16, 0.65 and 2.6 mJy at 3.3, 4.7, 12 and 23 microns, respectively, assuming 8 independent exposures and where the noise flux errors due to zodiacal foreground emission, instrumental effects, source photon statistics, and neighboring sources (traceable to Level-1).
• *L4WSDC-013*: The root mean square error in relative photometric accuracy in the WISE Source Catalog shall be better than 7% in each band for unsaturated point sources with SNR>100, where the noise flux errors due to zodiacal foreground emission, instrumental effects, source photon statistics, and neighboring sources. This requirement shall not apply to

sources that superimposed on an identified artifact (traceable to Level-1).

1.5 Acronyms

ADU	Analog Digital Unit
ADC	Analog-to-Digital Conversion
ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
CPU	Central Processing Unit
CR	Cosmic Ray
CVZ	Continuous Viewing Zone
DEB	Digital Electronics Box
DN	Data Number
FDD	Functional Design Document
FEB	Focal-plane Electronics Box
FRD	Functional Requirements Document
FITS	Flexible Image Transport System
FOV	Field of View
FPA	Focal Plane Array
HgCdTe	Mercury Cadmium Telluride
I/O	Input / Output
IOC	In-Orbit Checkout
ISO	International Organization for Standardization
ICAL	Instrumental CALibration
IPAC	Infrared Processing and Analysis Center
LSB	Least Significant Bit
MAD	Median Absolute Deviation
MED	Median
MOD	Mode
PRF	Point Response Function
RMS	Root Mean Square deviation from the mean
RSS	Root Sum Squared
SDL	Space Dynamics Lab
SDS	Subsystem Design Specification
SIS	Subsystem Interface Specification
SiAs	Silicon Arsenide
SNR	Signal-to-Noise Ratio
SUR	Sample-Up-the-Ramp
SVG	Scalable Vector Graphics format
TBD	To Be Determined
TBR	To Be Resolved
2MASS	Two Micron All Sky Survey
QA	Quality Assurance

QAP	Quality Assurance Plan
Q-to-Q	Quadrant-to-Quadrant
W1	WISE band 1 (3.4 μm)
W2	WISE band 2 (4.6 μm)
W3	WISE band 3 (12 μm)
W4	WISE band 4 (22 μm)
WDB	Working Data Base
WISE	Wide-field Infrared Survey Explorer
WSDC	WISE Science Data Center
WSDS	WISE Science Data System

2 OVERVIEW

WISE shall downlink image data frames consisting of 1024 × 1024 pixels for bands 1, 2 and 3 with a projected angular scale of 2.75 arcsec/pixel, and 512 × 512 pixels for band 4 with a scale of 5.5 arcsec/pixel. This corresponds to a FOV of ≈ 47 × 47 arcmin on the sky for all bands. Prior to source extraction to support pointing reconstruction and eventually co-addition, frames first need to be processed to remove instrumental signatures. Uncertainty images and masks indicating unusable pixels also need to be generated to support downstream processing at the WSDC. The ICAL pipeline is a script written in Perl, named *instruframecal*.

3 INPUT/OUTPUT SPECIFICATION

3.1 Inputs

instruframecal takes all of its inputs from the command-line, which is set up by a startup wrapper script (*wsfical*) and controlled by the WSDS pipeline executive, or, it can be set up manually and executed standalone. Prior to parsing the command-line inputs, default values for the optional input parameters are assigned. Table 1 summarizes all command-line inputs, their purpose, and default assignments.

Option	**Description**	**Data-type / Format**	**Units**	**Default**
-inpfile	Input FITS file name (raw, level-0 science image): 32-bit floating point / pixel. Must have suffix: "int-0.fits" in filename	Char*256	Null	Required input
-caldir	Input pathname to prepend to calibration files if paths not provided with filenames below	Char*256	Null	./ (current directory)
-m	Input bad-pixel mask FITS file name: either 8-bit / pixel, or 32-bit integer mask. If latter,	Char*256	Null	Required input

	specify –dm switch below			
-dm	Switch to indicate that input mask (-m) is 32-bit / pixel (long int). Otherwise an 8-bit 'static' mask is assumed and a 32-bit ‘dynamic’ mask is initialized from that	Null	Null	8-bit mask assumed for –m
-d	Input dark calibration FITS file name: 32-bit floating point / pixel	Char*256	Null	Required input
-du	Input dark calibration uncertainty FITS file name	Char*256	Null	None used
-l	Input non-linearity calibration FITS file name: 32-bit floating point / pixel	Char*256	Null	Required input
-lu	Input non-linearity calibration uncertainty FITS file name	Char*256	Null	None used
-f	Input flat-field calibration FITS file name: 32-bit floating point / pixel	Char*256	Null	Required input
-fu	Input flat-field calibration uncertainty FITS file name	Char*256	Null	None used
-fm	Input flat-field (dynamic on-orbit) calibration mask FITS file name.	Char*256	Null	None used - static information in ground mask assumed to apply
-fltbits	Input fatal bitstring template (in decimal) for input dynamic flat-field calibration mask (-fm) for flagging bad or unreliable responsivity corrections in output frame mask	I*2 int	Null	0
-lof	Input low-frequency correction flat-field calibration FITS file name: 32-bit floating point / pixel	Char*256	Null	None used
-lofu	Input low-frequency correction flat-field calibration uncertainty FITS file name	Char*256	Null	None used
-s	Input sky-offset calibration FITS file name: 32-bit floating point / pixel	Char*256	Null	Required input
-su	Input sky-offset calibration uncertainty FITS file name	Char*256	Null	None used
-metatbl	Input parameter meta-data table file	Char*256	Null	Required input
-unc	Switch to generate uncertainties and perform uncertainty propagation	Null	Null	None generated
-savelin	Switch to save intermediate 'lincor' product for dynamic flat-field estimation	Null	Null	None saved
-saveflt	Switch to save intermediate 'flatcor' product for dynamic	Null	Null	None saved

	sky-offset estimation			
-saveall	Switch to save all intermediate products for debugging. Will be generically named: <inpfile>-<intprodname>.fits	Null	Null	None saved
-outbase	Output FITS file basename (e.g., if outbase = '1a', output FITS file = <inpfile>-1a.fits). If –unc specified, output uncertainty = <inpfile>-unc-1a.fits. Output mask = <inpfile>-msk-1a.fits	Char*256	Null	Required input
-outdir	Pathname for output FITS file products	Char*256	Null	Required input
-qadir	Pathname for output QA diagnostic files	Char*256	Null	Required input
-archdir	Pathname for archivable products. Default = -outdir	Char*256	Null	–outdir <path>
-xyhist	x,y coordinate of a single pixel in 'active' region of native input frame for generating processing history (output to QA)	R*4 float	Native pixels	int[CRPIX1, CRPIX2]
-refmeth	Reference-pixel or offset correction method: 1 = average; 2 = median	I*2 int	Null	1
-gainmap	Input gain map calibration FITS file name. Required if –gain not specified and –unc specified. Overrides –gain	Char*256	Null	–gain <input> if –unc specified
-gain	Detector gain value. Required if –gainmap not specified and -unc specified	R*4 float	e- / SUR ADU	–gainmap <input> if –unc specified
-rdmap	Input read-noise map calibration FITS file name. Required if –rd not specified and –unc specified. Overrides -rd	Char*256	Null	–rd <input> if –unc specified
-rd	Detector 1-sigma read-noise value. Required if -rdmap not specified and -unc specified	R*4 float	e- / SUR read / pixel	–rdmap <input> if –unc specified
-ufudge	Scaling factor for final uncertainty image. If specified, overrides 'uncscal' factor from –metatbl input table	R*4 float	Null	read from –metatbl <inp>
-uncflg	Switch to dynamically rescale prior-uncertainty image using local RMS fluctuations based on -methscal <method>. If set, overrides value from -ufudge above and 'uncscal' factor from -metatbl input table	Null	Null	0
-methscal	Method to use for dynamic uncertainty scaling factor if -uncflg was set; 1 => "84-16%-tile/median(unc)";2 => "low-tail	I*2 int	Null	2

	StdDev from mode/mode(unc)";			
-trat	Threshold for ratio: 'frame pix value / local median filtered value' above which a pixel is to be declared an outlier spike and hence flagged in mask. If specified, overrides 'thresrat' parameter from -metatbl input table	R*4 float	Null	If not provided, required in –metatbl <inp>
-ksiz	Size of n x n median filter to use for outlier spike detection; must be odd number (3 is conservative). If specified, overrides 'ksize' parameter from -metatbl input table	I*2 int	pixels	If not provided, required in –metatbl <inp>
-sigfig	Number of significant figures to retain in unc outputs if -trcomp switch is set	I*2 int	Null	3
-trcomp	Switch to perform lossy compression on unc outputs by retaining n significant figures as specified by -sigfig <n>	Null	Null	0
-gzcomp	Switch to compress output msk, unc FITS file products using gzip	Null	Null	0
-refq1	Band 3 or 4 baseline reference pixel signal [DN] over quadrant 1 (upper right). If specified, overrides 'refbase1' parameter from -metatbl input table	R*4 float	DN	If not provided, required in –metatbl <inp>
-refq2	Band 3 or 4 baseline reference pixel signal [DN] over quadrant 2 (upper left). If specified, overrides 'refbase2' parameter from -metatbl input table	R*4 float	DN	If not provided, required in –metatbl <inp>
-refq3	Band 3 or 4 baseline reference pixel signal [DN] under quadrant 3 (lower left). If specified, overrides 'refbase3' parameter from -metatbl input table	R*4 float	DN	If not provided, required in –metatbl <inp>
-refq4	Band 3 or 4 baseline reference pixel signal [DN] under quadrant 4 (lower right). If specified, overrides 'refbase4' parameter from -metatbl input table	R*4 float	DN	If not provided, required in –metatbl <inp>
-reref	Switch to subtract zero reference pixel signal from active region	Null	Null	0
-allqa	Switch to generate all QA products with complete processing to make final calibrated product if either -	Null	Null	0

	savelin or -saveflt were specified (i.e., otherwise only partial QA generated specifically for intermediate linearized and/or flattened products)			
-dbg	Switch to print debug info. to stdout and ancillary FITS imgs	Null	Null	0
-v	Switch to increase verbosity to stdout	Null	Null	0

Table 1: Command-line inputs and options

If *instruframecal* is executed with a "-help2" (e.g., as "instruframecal –help2"), a command-line synopsis and tutorial is printed on the screen. This is a summary of the information presented in Table 1. Execution using "–help" yields a shorter summary of the I/O.

3.2 Input Detector Calibrations and Parameter Files

Specifications for the format and content of all calibration files are outlined in the following document: *http://web.ipac.caltech.edu/staff/fmasci/home/wise/GroundCalSpecs.pdf*

The pertinent detector calibrations used in ICAL are below. Each also has its accompanying uncertainty ("unc") and mask ("msk") images where applicable [not shown]. The Root Filename is a recommended naming practice and is not required by the software. Here, the <*origin*> prefix can be either "gnd" for ground/lab calibration, "flt" for flight/on-orbit calibration, or "sim" for simulated. The w<*n*> refers to the band: *n* = 1, 2, 3 or 4. The "Where from?" column indicates where the calibration product will come from for each band: Ground (G), In-Orbit Checkout (IOC), Survey Mode (SM), or a combination of these.

Root Filename	Description	Allowed Formats	Where from?
<*origin*>dark-w<*n*>-int.fits	Dark current/bias	1024^2 (w1-3); 512^2 (w4) FITS; 32-bit/pix float	**G:** all bands **IOC:** w1,2 cover on **SM:** w3,4 self cal
<*origin*>lincal-w<*n*>-est<*c*>.fits	Non-linearity correction for co-efficient "*c*" of linearity model	1024^2 (w1-3); 512^2 (w4) FITS; 32-bit/pix float	**G:** all bands **IOC:** all bands (coarse check) **SM:** no
<*origin*>flat-w<*n*>-int<*c*>.fits	Responsivity correction	1024^2 or 1016^2 (w1-3); 512^2 or 508^2 (w4) FITS; 32-bit/pix float	**G:** all bands **IOC:** all bands **SM:** all bands
<*origin*>lowflat-w<*n*>-int<*c*>.fits	Low-frequency responsivity corr. map	1024^2 or 1016^2 (w1-3); 512^2 or 508^2 (w4) FITS; 32-bit/pix float	**G:** no **IOC:** no **SM:** all bands
<*origin*>skyoff-w<*n*>-int.fits	Sky-offset correction	1024^2 or 1016^2 (w1-3); 512^2 or 508^2 (w4) FITS; 32-bit/pix float	**G:** no **IOC:** all bands (tune) **SM:** all bands

<*origin*>gain-w<*n*>-map.fits	Gain map [e-/SUR DN]	1024^2 (w1-3); 512^2 (w4) FITS; 32-bit/pix float	**G:** all bands **IOC:** maybe **SM:** maybe
<*origin*>rnoise-w<*n*>-map.fits	Read-noise map [e-/pix]	1024^2 (w1-3); 512^2 (w4) FITS; 32-bit/pix float	**G:** all bands **IOC:** maybe **SM:** maybe
<*origin*>mask-w<*n*>-msk.fits	Static bad-pixel mask	1024^2 (w1-3); 512^2 (w4) FITS; 8-bit/pix	**G:** all bands **IOC:** all bands **SM:** all bands (track transients)
wise-meta.tbl	Table containing band-dependent processing parameters. See below for example.	IPAC ASCII table (see below)	**G:** all bands **IOC:** all bands (tune) **SM:** all bands (tune)

Table 2: Input calibration files

Below is an example meta-data table (–metatbl <input> in Table 1) and the expected parameters for ICAL at the time of writing.

```
\ metadata for instrumental calibration pipeline (wsfical)
\ last modified: F. Masci,  2010-10-02 22:47:00 PDT
\ version == 2.0
\ revision = $Id$
\
| name                   | band | hdrname | type | value           | comment
|
|  c                     |  i   |  c      |  c   |  c              |  c
|
\
\ Absolute flux for magnitude zero
\
 cal:flt:magzero_flux      1      -         r       309.54          Absolute flux for magnitude
zero [Jy]
 cal:flt:magzero_flux      2      -         r       171.79          Absolute flux for magnitude
zero [Jy]
 cal:flt:magzero_flux      3      -         r       31.676          Absolute flux for magnitude
zero [Jy]
 cal:flt:magzero_flux      4      -         r       8.3635          Absolute flux for magnitude
zero [Jy]
\
\ Magnitude zeropoints for pre-pcal mdex processing of frames
\
 cal:flt:magzp_fr_mdex     1      -         r       20.730          Frame ZP for pre-pcal mdex
 cal:flt:magzp_fr_mdex     2      -         r       19.590          Frame ZP for pre-pcal mdex
 cal:flt:magzp_fr_mdex     3      -         r       17.540          Frame ZP for pre-pcal mdex
 cal:flt:magzp_fr_mdex     4      -         r       12.980          Frame ZP for pre-pcal mdex
\
\ Magnitude zeropoints for post-pcal mdex processing of coadds
\
 cal:flt:magzp_co_mdex     1      -         r       20.500          Coadd ZP for mdex
 cal:flt:magzp_co_mdex     2      -         r       19.500          Coadd ZP for mdex
 cal:flt:magzp_co_mdex     3      -         r       17.500          Coadd ZP for mdex
 cal:flt:magzp_co_mdex     4      -         r       13.000          Coadd ZP for mdex
\
\ VTC offets to go from packet time to mid-exposure point
\
 cal:vtc_offset            1      -         r       -4.853          Band 1 offset to mid-exposure
from source pkt VTC time
 cal:vtc_offset            2      -         r       -4.853          Band 2 offset to mid-exposure
from source pkt VTC time
```

```
 cal:vtc_offset           3      -         r       -4.853         Band 3 offset to mid-exposure
from source pkt VTC time
 cal:vtc_offset           4      -         r       -4.853         Band 4 offset to mid-exposure
from source pkt VTC time
\
\ Exposure constants
\
 ical:frint               0      -         r        11            Frame-to-frame interval
 ical:tsamp               0      -         r        1.1           Per-sample exposure time
 ical:exptime             1      -         r        7.7           Per-frame exposure time
 ical:exptime             2      -         r        7.7           Per-frame exposure time
 ical:exptime             3      -         r        8.8           Per-frame exposure time
 ical:exptime             4      -         r        8.8           Per-frame exposure time
\
\ SUR coefficients
\
 ical:coeff1              1      -         i       0              SUR coefficient 1
 ical:coeff2              1      -         i      -7              SUR coefficient 2
 ical:coeff3              1      -         i      -5              SUR coefficient 3
 ical:coeff4              1      -         i      -3              SUR coefficient 4
 ical:coeff5              1      -         i      -1              SUR coefficient 5
 ical:coeff6              1      -         i       1              SUR coefficient 6
 ical:coeff7              1      -         i       3              SUR coefficient 7
 ical:coeff8              1      -         i       5              SUR coefficient 8
 ical:coeff9              1      -         i       7              SUR coefficient 9
\
 ical:coeff1              2      -         i       0              SUR coefficient 1
 ical:coeff2              2      -         i      -7              SUR coefficient 2
 ical:coeff3              2      -         i      -5              SUR coefficient 3
 ical:coeff4              2      -         i      -3              SUR coefficient 4
 ical:coeff5              2      -         i      -1              SUR coefficient 5
 ical:coeff6              2      -         i       1              SUR coefficient 6
 ical:coeff7              2      -         i       3              SUR coefficient 7
 ical:coeff8              2      -         i       5              SUR coefficient 8
 ical:coeff9              2      -         i       7              SUR coefficient 9
\
 ical:coeff1              3      -         i      -4              SUR coefficient 1
 ical:coeff2              3      -         i      -3              SUR coefficient 2
 ical:coeff3              3      -         i      -2              SUR coefficient 3
 ical:coeff4              3      -         i      -1              SUR coefficient 4
 ical:coeff5              3      -         i       0              SUR coefficient 5
 ical:coeff6              3      -         i       1              SUR coefficient 6
 ical:coeff7              3      -         i       2              SUR coefficient 7
 ical:coeff8              3      -         i       3              SUR coefficient 8
 ical:coeff9              3      -         i       4              SUR coefficient 9
\
 ical:coeff1              4      -         i      -4              SUR coefficient 1
 ical:coeff2              4      -         i      -3              SUR coefficient 2
 ical:coeff3              4      -         i      -2              SUR coefficient 3
 ical:coeff4              4      -         i      -1              SUR coefficient 4
 ical:coeff5              4      -         i       0              SUR coefficient 5
 ical:coeff6              4      -         i       1              SUR coefficient 6
 ical:coeff7              4      -         i       2              SUR coefficient 7
 ical:coeff8              4      -         i       3              SUR coefficient 8
 ical:coeff9              4      -         i       4              SUR coefficient 9
\
\ DEB offsets
\
 ical:offset              1      -         r       1024           Offset added to scaled slope
from DEB
 ical:offset              2      -         r       1024           Offset added to scaled slope
from DEB
 ical:offset              3      -         r       1024           Offset added to scaled slope
from DEB
 ical:offset              4      -         r       1024           Offset added to scaled slope
from DEB
\
\ DEB LSB-truncations
\
 ical:trunc                1      -         i       3               Number of LSBs truncated in
DEB
 ical:trunc                2      -         i       3               Number of LSBs truncated in
DEB
 ical:trunc                3      -         i       2               Number of LSBs truncated in
DEB
 ical:trunc                4      -         i       2               Number of LSBs truncated in
DEB
\
\ DEB gains (not used in ICal, only informatory). From Doc SDL/09-257.
\
 ical:debgain              1      -         r       3.75            DEB gain in e-/DEB ADU
 ical:debgain              2      -         r       4.6             DEB gain in e-/DEB ADU
 ical:debgain              3      -         r       6.83            DEB gain in e-/DEB ADU
 ical:debgain              4      -         r       4.73            DEB gain in e-/DEB ADU
\
\ FEB gains (not used in ICal, only informatory). From Doc SDL/09-419.
\
 ical:febgain              1      -         r       5.74            FEB gain in e-/SUR ADU
 ical:febgain              2      -         r       6.86            FEB gain in e-/SUR ADU
 ical:febgain              3      -         r       12.83           FEB gain in e-/SUR ADU
 ical:febgain              4      -         r       8.86            FEB gain in e-/SUR ADU
\
\ To support ical:uncmod = 1 below: rnflg=1 => input read-noise map is in
\ DEB DN units, otherwise if rnflg=0, read-noise map is in FEB electron units.
\
 ical:rnflg                1      -         i       1               Read-noise units usage flag
 ical:rnflg                2      -         i       1               Read-noise units usage flag
 ical:rnflg                3      -         i       1               Read-noise units usage flag
 ical:rnflg                4      -         i       1               Read-noise units usage flag
\
\ Scaling factors for final model uncertainties (1.0 => model is perfect)
\
 ical:uncscal              1      -         r       1.70            Scaling factor for final
model uncertainty
 ical:uncscal              2      -         r       1.36            Scaling factor for final
model uncertainty
 ical:uncscal              3      -         r       1.36            Scaling factor for final
model uncertainty
 ical:uncscal              4      -         r       1.60            Scaling factor for final
model uncertainty
\
\ Baseline reference pixel signals (e.g., long-run median or mode) for
\ droop correction (bands 3 and 4 only)
\
 ical:refbase1             3      -         r       250.6           Baseline reference pix signal
at upper right [DN]
 ical:refbase2             3      -         r       250.2           Baseline reference pix signal
at upper left [DN]
 ical:refbase3             3      -         r       247.4           Baseline reference pix signal
at lower left [DN]
 ical:refbase4             3      -         r       248.6           Baseline reference pix signal
at lower right [DN]
 ical:refbase1             4      -         r       252.2           Baseline reference pix signal
at upper right [DN]
 ical:refbase2             4      -         r       249.3           Baseline reference pix signal
at upper left [DN]
 ical:refbase3             4      -         r       245.2           Baseline reference pix signal
at lower left [DN]
 ical:refbase4             4      -         r       245.8           Baseline reference pix signal
at lower right [DN]
\
\ Minimum tolerable fraction of good reference pixels per quadrant for
\ droop correcting bands 3 and 4. Smaller => bias (under-correction)
\ can result; bigger => more cases will use neighboring active region
\ with good droop-corr.
\
 ical:gfrac                3      -         r       0.5             Min. fraction of good
reference pixels
```

```
 ical:gfrac                4      -         r       0.5             Min. fraction of good
reference pixels
\
\ Flag to perform final droop refinement for band 3 or band 4 frames whose
\ quadrants had too many bad reference pixels (i.e., <=gfrac)
\
 ical:drpflag              3      -         i       1               Flag to perform final droop
refinement
 ical:drpflag              4      -         i       1               Flag to perform final droop
refinement
\
\ Threshold for initial detection of intra-quadrant splits due to droop
\ using first derivative of column-cut quantile vectors for bands 3 and 4
\ in native DEB DN.
\
 ical:splithres            3      -         r       4.0             Threshold for split detection
[DN]
 ical:splithres            4      -         r       4.0             Threshold for split detection
[DN]
\
\ Post-threshold for retention of intra-quadrant splits using quantile
\ values in strips on either side of splits. For bands 3 and 4.
\
 ical:postsplithres        3      -         r       4.0             Post-threshold for split
detection [DN]
 ical:postsplithres        4      -         r       4.0             Post-threshold for split
detection [DN]
\
\ Lower-tail fraction for computing quantiles for split corrections in
\ band 3 and 4 quadrants.
\
 ical:falow                3      -         r       0.1            Lower-tail fraction to compute
quantiles
 ical:falow                4      -         r       0.1            Lower-tail fraction to compute
quantiles
\
\ Threshold ratios for outlier-spike detection (bigger value => unlikely
\ to detect)
\
 ical:thresrat             1      -         r       10.0            Flux/med-filt threshold ratio
 ical:thresrat             2      -         r       10.0            Flux/med-filt threshold ratio
 ical:thresrat             3      -         r       10.0            Flux/med-filt threshold ratio
 ical:thresrat             4      -         r       10.0            Flux/med-filt threshold ratio
\
\ Median filter kernel sizes for outlier-spike detection (must be odd;
\ 3 => conservative, hard edged clusters not completely blanketed)
\
 ical:ksize                1      -         i       5               Median filter kernel size
 ical:ksize                2      -         i       5               Median filter kernel size
 ical:ksize                3      -         i       5               Median filter kernel size
 ical:ksize                4      -         i       5               Median filter kernel size
\
\ Max DEB DN for quadratic non-linearity model with linear extension above this
\
 ical:mobsmax              1      -         i       22500           Maximum DEB value [DN] for
quad. NL model
 ical:mobsmax              2      -         i       17500           Maximum DEB value [DN] for
quad. NL model
 ical:mobsmax              3      -         i       32000           Maximum DEB value [DN] for
quad. NL model
 ical:mobsmax              4      -         i       32000           Maximum DEB value [DN] for
quad. NL model
\
\ Uncertainty model: 1 = WSDC's (uses explicit gains + RNs); 2 = Ned's to
\ support v3 sims - see below for params
\
 ical:uncmod               1      -         i       1               Uncertainty model method
 ical:uncmod               2      -         i       1               Uncertainty model method
 ical:uncmod               3      -         i       1               Uncertainty model method
 ical:uncmod               4      -         i       1               Uncertainty model method
```

```
\
\ BSCALEs to support ical:uncmod = 2
\
 ical:bscale               1      -         r       5.2             BSCALE in microJy/DEB DN
 ical:bscale               2      -         r       4.2             BSCALE in microJy/DEB DN
 ical:bscale               3      -         r       1.95            BSCALE in microJy/DEB DN
 ical:bscale               4      -         r       14.7            BSCALE in microJy/DEB DN
\
\ BZEROs to support ical:uncmod = 2
\
 ical:bzero                1      -         r       -665.6          BZERO in microJy
 ical:bzero                2      -         r       -537.6          BZERO in microJy
 ical:bzero                3      -         r       -499.2          BZERO in microJy
 ical:bzero                4      -         r       -3763.0         BZERO in microJy
\
\ Readnoise (in microJy not e-) to support ical:uncmod = 2
\
 ical:sigrn                1      -         r       6.45            Sigma_RN in microJy
 ical:sigrn                2      -         r       5.75            Sigma_RN in microJy
 ical:sigrn                3      -         r       41.2            Sigma_RN in microJy
 ical:sigrn                4      -         r       131.4           Sigma_RN in microJy
\
\ Stack dark noise (in microJy not e-) to support ical:uncmod = 2
\ for pattern/no-pattern noise sims
\
 ical:sigdrk               1      -         r       6.45            Sigma_Drk in microJy
 ical:sigdrk               2      -         r       5.75            Sigma_Drk in microJy
 ical:sigdrk               3      -         r       41.2            Sigma_Drk in microJy
 ical:sigdrk               4      -         r       131.4           Sigma_Drk in microJy
\
\ Double parameter to support ical:uncmod = 2
\
 ical:double               1      -         r       99.49           double in microJy
 ical:double               2      -         r       74.1            double in microJy
 ical:double               3      -         r       5135            double in microJy
 ical:double               4      -         r       9892            double in microJy
\
\ Fatal bitstring template for setting masked pixels to NaNs in
\ intensity and uncertainty frames (523804 => bits 2,3,4,9..18)
\
 ical:fatalbits            1      -         r       523804          fatal bitstring [decimal]
 ical:fatalbits            2      -         r       523804          fatal bitstring [decimal]
 ical:fatalbits            3      -         r       523804          fatal bitstring [decimal]
 ical:fatalbits            4      -         r       523804          fatal bitstring [decimal]
\
\ Thresholds for metrics on intermediate ical products to support dynamic
\ calibration (tempcal: prefix = "t"; flatcal = "f"). These are used to
\ assign quality dynamic-cal usability flags to each frame.
\
 ical:tdtanneal            1      DTANNEAL  r       0.0             Closest time since last
anneal [sec]
 ical:tfpatemp             1      B1FEBFT1  r       45.0            Maximum FPA Temperature [K]
 ical:tminelat             1      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:tmaxelat             1      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:tminglat             1      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:tmaxglat             1      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:tnspikelo            1      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:tnspikehi            1      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:tnumsat              1      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:tminillum            1      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:tmaxillum            1      -         r       250.0           Maximum median
illumination/pixel [DEB DN]
```

```
 ical:tmaxrmsmad           1      -         r       150.0           Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:tmaxrmspti           1      -         r       150.0           Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:tmaxrat              1      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
 ical:fdtanneal            1      DTANNEAL  r       0.0             Closest time since last
anneal [sec]
 ical:ffpatemp             1      B1FEBFT1  r       45.0            Maximum FPA Temperature [K]
 ical:fminelat             1      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:fmaxelat             1      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:fminglat             1      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:fmaxglat             1      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:fnspikelo            1      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:fnspikehi            1      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:fnumsat              1      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:fminillum            1      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:fmaxillum            1      -         r       200.0           Maximum median
illumination/pixel [DEB DN]
 ical:fmaxrmsmad           1      -         r       100.0           Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:fmaxrmspti           1      -         r       100.0           Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:fmaxrat              1      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
\
 ical:tdtanneal            2      DTANNEAL  r       0.0             Closest time since last
anneal [sec]
 ical:tfpatemp             2      B2FEBFT1  r       45.0            Maximum FPA Temperature [K]
 ical:tminelat             2      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:tmaxelat             2      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:tminglat             2      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:tmaxglat             2      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:tnspikelo            2      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:tnspikehi            2      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:tnumsat              2      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:tminillum            2      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:tmaxillum            2      -         r       250.0           Maximum median
illumination/pixel [DEB DN]
 ical:tmaxrmsmad           2      -         r       100.0           Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:tmaxrmspti           2      -         r       100.0           Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:tmaxrat              2      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
 ical:fdtanneal            2      DTANNEAL  r       0.0             Closest time since last
anneal [sec]
 ical:ffpatemp             2      B2FEBFT1  r       45.0            Maximum FPA Temperature [K]
 ical:fminelat             2      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:fmaxelat             2      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:fminglat             2      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
```

```
 ical:fmaxglat             2      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:fnspikelo            2      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:fnspikehi            2      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:fnumsat              2      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:fminillum            2      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:fmaxillum            2      -         r       150.0           Maximum median
illumination/pixel [DEB DN]
 ical:fmaxrmsmad           2      -         r       100.0           Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:fmaxrmspti           2      -         r       100.0           Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:fmaxrat              2      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
\
 ical:tdtanneal            3      DTANNEAL  r       0.0             Closest time since last
anneal [sec]
 ical:tfpatemp             3      B3FEBFT1  r       100.0           Maximum FPA Temperature [K]
 ical:tminelat             3      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:tmaxelat             3      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:tminglat             3      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:tmaxglat             3      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:tnspikelo            3      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:tnspikehi            3      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:tnumsat              3      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:tminillum            3      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:tmaxillum            3      -         r       10000.0         Maximum median
illumination/pixel [DEB DN]
 ical:tmaxrmsmad           3      -         r       1000.0          Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:tmaxrmspti           3      -         r       1000.0          Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:tmaxrat              3      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
 ical:fdtanneal            3      DTANNEAL  r       3000.0          Closest time since last
anneal [sec]
 ical:ffpatemp             3      B3FEBFT1  r       7.3             Maximum FPA Temperature [K]
 ical:fminelat             3      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:fmaxelat             3      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:fminglat             3      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:fmaxglat             3      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:fnspikelo            3      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:fnspikehi            3      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:fnumsat              3      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:fminillum            3      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:fmaxillum            3      -         r       5000.0          Maximum median
illumination/pixel [DEB DN]
 ical:fmaxrmsmad           3      -         r       300.0           Maximum tolerable pixel RMS
(from intSigMADMED)
```

```
 ical:fmaxrmspti           3      -         r       300.0           Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:fmaxrat              3      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
\
 ical:tdtanneal            4      DTANNEAL  r       0.0             Closest time since last
anneal [sec]
 ical:tfpatemp             4      B4FEBFT1  r       100.0           Maximum FPA Temperature [K]
 ical:tminelat             4      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:tmaxelat             4      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:tminglat             4      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:tmaxglat             4      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:tnspikelo            4      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:tnspikehi            4      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:tnumsat              4      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:tminillum            4      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:tmaxillum            4      -         r       10000.0         Maximum median
illumination/pixel [DEB DN]
 ical:tmaxrmsmad           4      -         r       1000.0          Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:tmaxrmspti           4      -         r       1000.0          Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:tmaxrat              4      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
 ical:fdtanneal            4      DTANNEAL  r       3000.0          Closest time since last
anneal [sec]
 ical:ffpatemp             4      B4FEBFT1  r       7.3             Maximum FPA Temperature [K]
 ical:fminelat             4      ELAT0     r       -90.0           Minimum ecliptic latitude
[deg]
 ical:fmaxelat             4      ELAT0     r       90.0            Maximum ecliptic latitude
[deg]
 ical:fminglat             4      GLAT0     r       -90.0           Minimum galactic latitude
[deg]
 ical:fmaxglat             4      GLAT0     r       90.0            Maximum galactic latitude
[deg]
 ical:fnspikelo            4      -         i       258064          Max. number of low-tail
outliers (glitches)
 ical:fnspikehi            4      -         i       258064          Max. number of high-tail
outliers (glitches)
 ical:fnumsat              4      -         i       258064          Max. number of saturated
pixels (at any SUR sample)
 ical:fminillum            4      -         r       0.0             Minimum median
illumination/pixel [DEB DN]
 ical:fmaxillum            4      -         r       3000.0          Maximum median
illumination/pixel [DEB DN]
 ical:fmaxrmsmad           4      -         r       250.0           Maximum tolerable pixel RMS
(from intSigMADMED)
 ical:fmaxrmspti           4      -         r       250.0           Maximum tolerable pixel RMS
(from intMed16ptile)
 ical:fmaxrat              4      -         r       5               Maximum tolerable ratio:
(84%tile-Med)/(Med-16%tile)
```

3.3 Output Files

There are three FITS image products generated by *instruframecal*: the main calibrated intensity frame, accompanying uncertainty frame, and mask image. The uncertainty frame is optional and is only generated if the –unc switch is specified. The output file names are constructed from the input filename (–inpfile) by replacing the “0.fits” suffix (i.e., the level-0 input) with the string specified by the –outbase input. The uncertainty and mask filenames will have the “int” string (as required in the –inpfile input) replaced by “unc” and “msk” respectively.

The uncertainty image contains a 1-sigma error estimate in the intensity signal for every pixel. These uncertainties are initiated by an error-model and updated using formal error-propagation as calibrations are applied downstream. They can then be optionally rescaled for consistency with local RMS pixel fluctuations using the –ufudge, or –uncflg and –methscal parameters. For more details, see the algorithm description in §4.3. Quality Assurance (QA) metrics are written to a table specified by –metatbl and archived under –archdir. QA plots (in SVG format) are generated at the end of processing under –qadir. All metrics and plots are discussed in §5.

3.4 Pixel Units

The pixel units of the input level-0 science frame are expected to be in raw DEB units. These go something like “*a* + *b*slopefit*”, where *slopefit* is the value of the fitted slope (or rate) to on-board Sample-Up-the-Ramp (SUR) data in units of DN/sample, and *a* and *b* are constants (see Eq. 1). The units of the output intensity frames are expected to go something like *c*slope_fit*, where the *a* was removed by e.g., dark subtraction and the *b* re-scaled by scaling operations (e.g., flat-fielding) to create an effective new scaling factor *c*. The output pixel values will therefore remain in units of “scaled-slope” throughout WSDS processing. These are therefore relative units. A calibration factor (or magnitude zero-point offset) is computed downstream in the WSDS pipeline using photometry relative to a standard star network and inserted into headers downstream.

4 ICAL PROCESSING AND ALGORITHMS

4.1 Overview of Processing Steps

Figure 1 captures all the major processing steps in the ICAL subsystem. The cyan/bluish-colored boxes indicate steps generic to all bands. Yellow is specific to bands 1 and 2, and red is specific to bands 3 and 4. Intermediate inputs for the two dynamic calibration-creation steps: *compflat* and *dynacal* are also shown (green boxes). These modules are not explicitly executed in *instruframecal*. Products from these software modules are used in *instruframecal* to perform the flat-field and sky-offset corrections respectively. These modules are described in separate SDS documents (docs D-D021 and D-D011 respectively; see references in §1.3). Algorithms for the steps shown in Figure 1, where available, are described below.

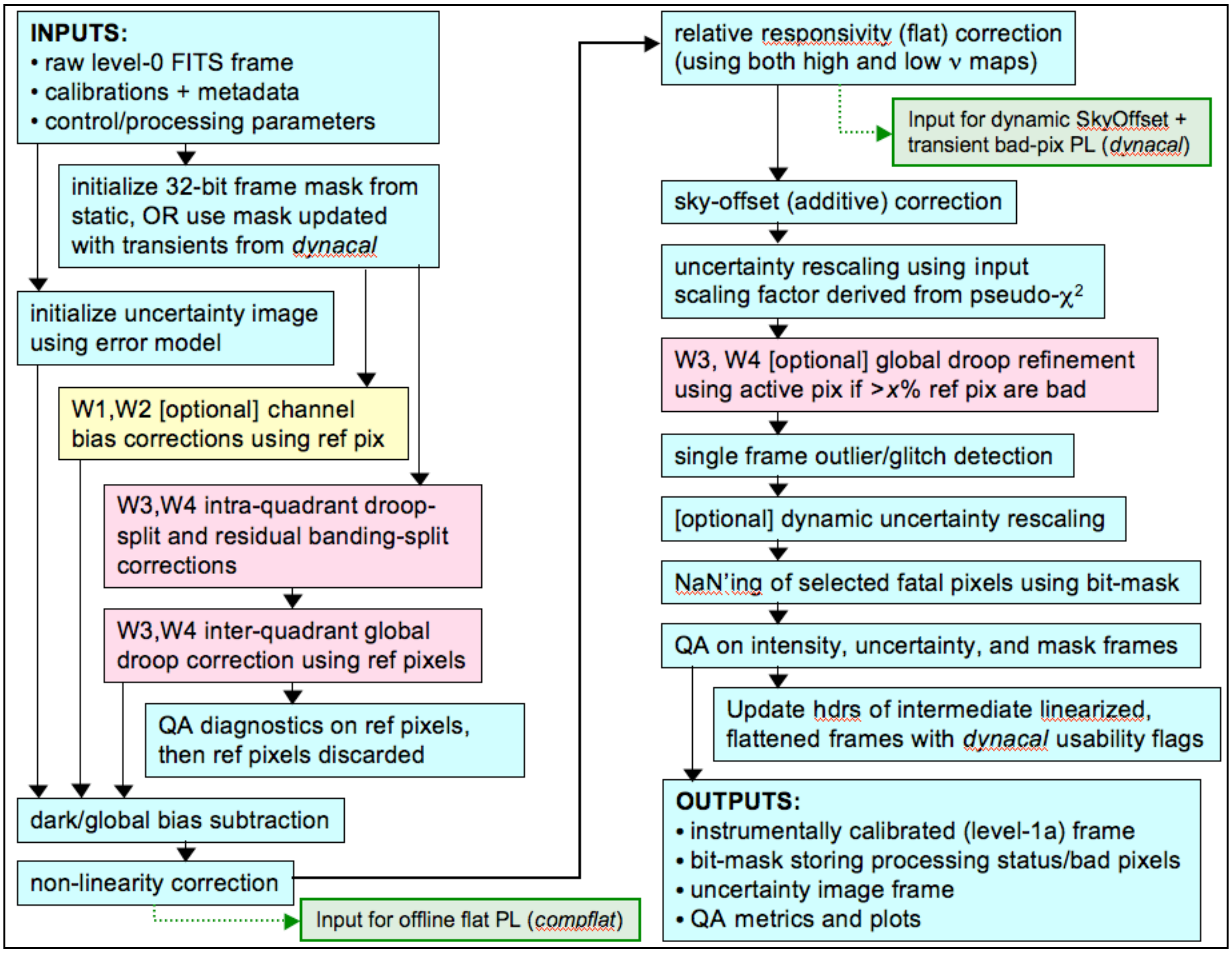
INPUTS:
• raw level-0 FITS frame
• calibrations + metadata
• control/processing parameters
initialize 32-bit frame mask from static, OR use mask updated with transients from dynacal
initialize uncertainty image using error model
W1,W2 [optional] channel bias corrections using ref pix
W3,W4 intra-quadrant droop-split and residual banding-split corrections
W3,W4 inter-quadrant global droop correction using ref pixels
QA diagnostics on ref pixels, then ref pixels discarded
dark/global bias subtraction
non-linearity correction
Input for offline flat PL (compflat)
relative responsivity (flat) correction (using both high and low ν maps)
Input for dynamic SkyOffset + transient bad-pix PL (dynacal)
sky-offset (additive) correction
uncertainty rescaling using input scaling factor derived from pseudo-χ²
W3, W4 [optional] global droop refinement using active pix if >x% ref pix are bad
single frame outlier/glitch detection
[optional] dynamic uncertainty rescaling
NaN'ing of selected fatal pixels using bit-mask
QA on intensity, uncertainty, and mask frames
Update hdrs of intermediate linearized, flattened frames with dynacal usability flags
OUTPUTS:
• instrumentally calibrated (level-1a) frame
• bit-mask storing processing status/bad pixels
• uncertainty image frame
• QA metrics and plots

Figure 1: ICAL processing flow in *instruframecal* script

4.2 Mask Initialization

By default, the frame processing (dynamic) mask is initialized from a static calibration mask (input –m) containing the locations of known bad-hardware pixels characterized on the ground and possibly updated on-orbit. The static mask is expected to be in 8-bits/pixel FITS format, while the dynamic mask generated there from will have 32-bit/pixel (long integer) format. The input static bits are copied into the first 8 bits of the dynamic mask.

In addition to transferring bad hardware pixels, other specialized bits are set according to pixel values in the input raw science frame. In general, the maximum value in a raw science frame (from the DEB) is 32,752. Above this (up to 32,767), the values 32,752 + *n* are reserved to encode SUR numbers *n* where ramps begin to saturate or are physically broken. The specific mask bits (in the 32-bit mask) assigned to flag anomalous downlink values range from 9 to 18 (see below). At the time of writing, the mask-bit definitions are as follows (from SIS *WSDC D-I101*; see reference in §1.3):

Bit #	Condition
0	from static mask: excessive dark current
1	from static mask: excessive read noise not due to high dark current
2	from static mask: dead or very low responsivity
3	from static mask: low responsivity
4	from static mask: high responsivity
5	from static mask: saturated anywhere in ramp
6	from static mask: high, uncertain, or unreliable non-linearity
7	from static mask: broken pixel or -'ve SUR (raw frame value = 32767)
8	reserved
9	broken pixel or intrinsically -'ve SUR (downlink value = 32767)
10	saturated in sample read 1 (downlink value = 32753)
11	saturated in sample read 2 (downlink value = 32754)
12	saturated in sample read 3 (downlink value = 32755)
13	saturated in sample read 4 (downlink value = 32756)
14	saturated in sample read 5 (downlink value = 32757)
15	saturated in sample read 6 (downlink value = 32758)
16	saturated in sample read 7 (downlink value = 32759)
17	saturated in sample read 8 (downlink value = 32760)
18	saturated in sample read 9 (downlink value = 32761)
19	reserved
20	reserved
21	new/transient bad pixel from dynamic masking (tempcal)
22	flat-fielding (responsivity correction) unreliable (ical)
23	sky-offset correction unreliable (tempcal)
24	reserved
25	contains probable latent flux (tempcal)
26	non-linearity correction unreliable (ical)
27	contains cosmic-ray or outlier that cannot be classified (awod)
28	contains positive or negative spike-outlier (ical)
29	reserved
30	reserved
31	not used: sign bit

Alternatively, the input mask (–m) may already be a dynamic 32-bit mask that was updated with transient bad pixels using on-orbit image data by the *dynacal* (*tempcal*) pipeline. If so, then the –dm switch should be set to indicate this. The resulting 32-bit mask is further updated downstream by ICAL.

4.3 Uncertainty Initialization

One-sigma uncertainties for each pixel in the input raw science frame are initialized using a non-trivial error Poisson + read-noise error model that accounts for correlated noise in the ramps. This is only performed if the –unc switch was specified on input. Below we show the main equation for the noise-variance per pixel. A derivation with simulations is given in document WSDC D-T001 (referenced in §1.3).

The error model is derived from the DEB equation, where in general the observed (down-linked) value *m* for a pixel is computed on-board from a linear combination of Samples Up the Ramp (SUR) values y_i in DN:

$$m = \frac{1}{2^T}\left[O + \sum_{i=0}^{N} c_i y_i\right], \qquad \text{(Eq. 1)}$$

where nominally $N = 8$; c_i are the SUR coefficients: {0,-7,-5,-3,-1,1,3,5,7} for bands 1 & 2, and {-4,-3,-2,-1,0,1,2,3,4} for bands 3 & 4; *O* is an offset (= 1024 for all bands), and *T* is the number of LSB truncations performed on the DEB output (= 3, 3, 2, 2 for bands 1, 2, 3, 4 respectively).

The noise variance in *m* for a pixel in a WISE array is given by:

$$\sigma_m^2 \approx \left(\frac{m2^T - O}{2^{2T} g K_N}\right)\left[\sum_{i=0}^{N} i c_i^2 + 2\sum_{i=0}^{N}\sum_{j>i}^{N} i c_i c_j\right] + \left(\frac{\sigma_{RN}^2}{2^{2T} g^2}\right)\sum_{i=0}^{N} c_i^2, \qquad \text{(Eq. 2)}$$

where:

m, *O* and *T* are defined in Eq. 1 and the text below it;

$g =$ gain in electrons/DN in any SUR;

c_i with $i \in \{0, 1, 2, 3 \ldots N\}$ are the predetermined SUR coefficients;

$N =$ maximum number of samples $-$ 1. Nominally $N = 8$;

$$K_N = \sum_{i=0}^{N} i c_i;$$

$\sigma_{RN} =$ readnoise in electrons/pixel in any SUR.

The inputs required for the uncertainty estimation are gain and read-noise maps (or individual values), SUR weighting coefficients, and the DEB processing parameters: *offset* (*O* in Eq. 2) and *trunc* (*T* in Eq. 2) read from the –metatbl input. These uncertainties are updated downstream as each "uncertain" calibration is applied. There is an option (either –ufudge or *uncscal* from –metatbl) to rescale the final output uncertainties. This allows one to obtain consistency with the RMS in spatial noise fluctuations about the background in the intensity image frame. Also, one can use the –uncflg and –methscal options to dynamically rescale the uncertainties using a search of the lowest local intensity RMS to compute a pseudo χ^2 metric ~ rms/σ_{model}. Reference pixels are discarded from the uncertainty frame at the end of computations and the "active" regions (1016 × 1016 for bands 1,2,3; 508 × 508 for band 4) are propagated along.

4.4 Reference Pixel Bias Corrections (W1 and W2)

The original intent of this step was to correct for readout-amplifier dependent biases in the W1 and W2 arrays using the (4-pixel wide) reference border. Following ground characterization, an algorithm was recommended by Teledyne back in June 2007 (the manufacture of the HgCdTe arrays) - see Appendix I of document WSDC D-I005 (referenced in §1.3). However, this algorithm does not perform as expected. It is therefore currently turned off in production and plans are underway to replace it.

4.5 Droop Corrections in W3 and W4

Droop manifests itself in two forms: (i) intra-quadrant splitting, usually bisecting a saturated source or region, and (ii) global quadrant-to-quadrant (Q-to-Q) effects leading to a depression of one or more quadrants followed by an increase (or rebound) in the quadrant signals in subsequent frames. During the rebound phase, the quadrants also become unstable at the location of bad-pixel clusters and the banding/split structure seen in the dark becomes apparent, lasting for 10-20 frames. The residual bands (in excess of those seen in the static dark calibration) are also referred to as stationary splits and can be corrected using the same algorithm as for intra-quadrant droop-splits. Both forms of droop have been confirmed to be additive in nature and it is assumed that the banding-split residuals are additive too so all these can be corrected using offset-matching. There is also a final (optional) refinement step for cases where a global Q-to-Q correction could not be computed. We describe the algorithms (in order of execution) below. See also Figure 1 for the location of these steps in the pipeline flow.

4.5.1 Intra-quadrant droop-split, including residual banding-split corrections

Before outlining the steps, we note that this version of the algorithm corrects splits using exclusively the active pixels. An initial version attempted to use the reference pixels at the top and bottom of each quadrant and demarcated by the split locations. This approach was aborted since (i) splits are often associated with heavily drooped cases where a large fraction of the reference pixel values are truncated at zero (represented as values 32767 in the downlink, which means the DEB slopes are really < 0). This implies that any metric that uses the reference pixels (e.g., median) will be biased high. There are indeed some cases with a good fraction of usable reference pixels, but their occurrence is infrequent to warrant mixing algorithms. The number of usable reference pixels becomes even lower when multiple splits are present, since one must consider only those pixels within regions separated by splits. The situation is worst for W4 where there are only two rows of pixels to work with; (ii) Furthermore, as will become apparent below, optimal correction for splits (whether droop or residual-banding related), requires that metrics be computed off a pre-calibrated frame, e.g., one that has been dark and responsivity corrected. This is not possible for the reference pixels because there is no attempt to calibrate them in processing. The current calibrations do not preserve reference pixel information that can be used in any meaningful manner downstream.

1. We start with raw L0 frame and create an internal regularized spit-detection image. This image is made by subtracting the static dark, dividing by the static responsivity map, and replacing bad pixels (using the static bad-pixel mask) with a median of the respective quadrants. The latter is needed since the bad pixels are usually represented as saturated/high values in the input frame and these interfere when trying to search for droop-related splits across real saturated sources downstream. It's important to note that this is a temporary working frame stored in memory (and can be optionally written to a FITS file) which serves only for detecting candidate splits and computing corrections for further dissemination. Actual corrections are applied to the input L0 frame for propagation downstream, as described below. The dark and responsivity corrections are needed because first, the strong edge vignetting in a L0 frame makes the computation of offset corrections close to a frame edge unreliable, and second, the banding-related split corrections are intended to be *residual* corrections on top of the static dark that will be subtracted further downstream. If we did not subtract the dark, we run the risk of removing the entire 'static' banding structure using our split corrections and this will need to be re-factored back (in some cumbersome way) before the final dark is subtracted in the pipeline.

2. The active regions of each quadrant are extracted from the internal working frame. Next, we search for candidate splits in each quadrant. We first collapse the columns in a quadrant (or take a cut across them) by computing the quantiles corresponding to some fraction *falow* in the low-tail of the data distribution in each column. *falow* is a parameter specified in the *wise-meta.tbl* file and at the time of writing, *falow*=0.1. Note that, *falow*=0.5 corresponds to the median. We choose to work with quantiles in the low-tail of the histogram because they are *less* biased by the presence of bright extended structure and sources. I.e., such measures will be closer to the underlying background. W3 and W4 are background dominated for the most part, but occasionally, bright extended structure can occur. A median is not as robust against such structure than some low-tail quantile and hence is more likely to over-estimate a droop offset correction. Therefore, low-tail quantiles are more representative of the background values and relative offsets derived there from are expected to better track horizontal shifts in the pixel distribution due to droop. An example comparing median and quantile differences across a split is shown in Figure 2. Given the noise, the real offset appears to be closer to the 10%-tile (0.1 quantile) difference of ~7 DN, and not the much larger median difference of ~54 DN.

3. Given the 1-D vector of column-quantile values from step 2 for a quadrant, we compute its first derivative (just the first difference array of pairwise values in the increasing *x*-direction). This vector is thresholded using the input parameter *splithres* to find the locations of all the candidate splits. Droop-related splits are retained by finding those splits which intercept at least one saturated pixel (with pixel values in the range 32753 ≤ value ≤ 32761 in the L0 frame). Splits associated with the residual banding are also retained by comparing the candidate split list with the known stationary banding-split locations. The idea here is to be complete in the split detections by setting *splithres* to be relatively low (e.g., 4 DN).

4. For a candidate split, the lower-quantile values in strips of width 7 pixels (at the time of writing) on either side of the split at a distance of ± 2 pixels from the split location are computed. Note that these are still computed from the internal temporary working frame. These strips contain more pixels than the single columns used for the initial thresholding in 3 and therefore provide a more reliable measure to filter out spurious splits. For a given split, the quantile difference of its adjacent strips is then thresholded against a second (but higher) threshold parameter (*postsplithres*, e.g., ~6 - 10) to determine whether the split magnitude is significant and worth correcting.

5. If a split survives after the post-thresholding step in 4, the quantile difference in adjacent strips is then used as the offset correction to apply to *all active and reference pixels in the quadrant* on one side of the split *in the original L0 frame*. It doesn't matter which side is corrected. I happened to pick the left side throughout. Reference pixels are also corrected for self-consistency since these are used to compute global Q-to-Q corrections below (see Section 4.5.2). Note that only pixel columns at distances > ± 2 pixels from the split are corrected since the split transition region is not necessarily a sharp step. The transition region is treated next.

6. Finally, we correct split transition columns which span a width of 5 pixels (± 2 pixels including the location of the split). For these columns, we use the original column quantile vector from step 2. For each column, offset corrections are computed using the low-tail single column quantile and the quantile in the strip on the uncorrected side of the split. The goal is to equalize the level in the split transition region with other regions in the quadrant.

7. If there is more than one split in the quadrant (e.g., other splits related to droop, banding-residuals, or both), we iterate on steps 4, 5, and 6 until all splits and transitions are corrected. To ensure this, the internal working temporary frame is self-consistently updated following each correction to ensure corrections for other splits do not become biased. Again, each split correction is applied to the input L0 active and reference pixels, and the process is repeated for each quadrant.

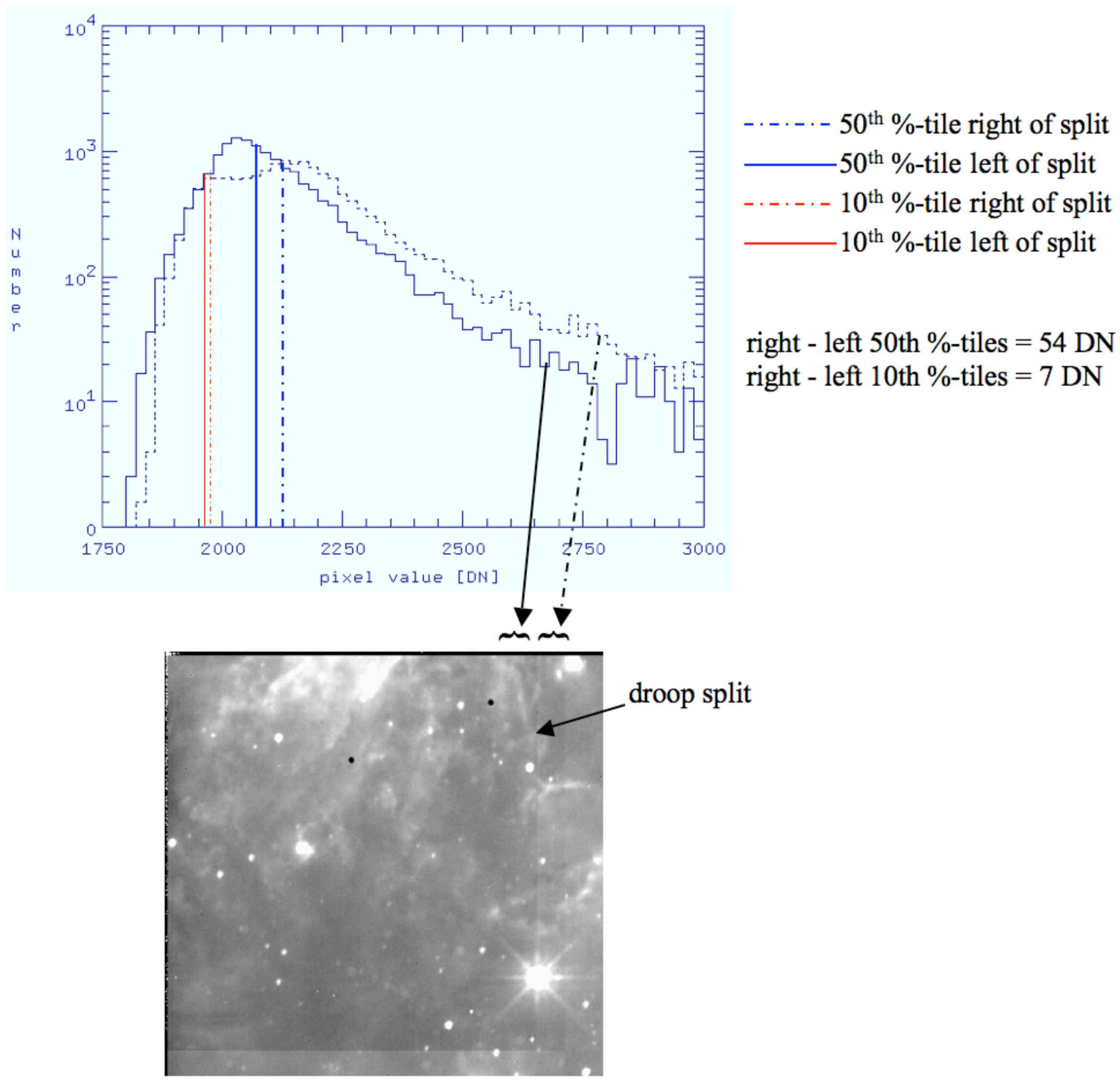

Figure 2: Example of a droop-related split in the top left quadrant of frame 00630x242-w3. This compares the median and lower 10%-tile estimates in the histogram of pixel values in narrow strips on either side of the split. The nebulosity appears to have *more* of an adverse effect on the median.

4.5.2 Global inter-quadrant (Q-to-Q) droop corrections

Following the split corrections to equalize levels within quadrants, Q-to-Q relative offsets are then computed and applied to equalize levels over the whole frame. This step exclusively uses the reference pixels where possible (which may have been updated by the split correction step above). The correction is computed for each separate quadrant by first computing the median of all good reference pixels (above or below it) where 'good' means those pixels with value < 32767 (where 32767 is equivalent to a DEB instrinsic slope value < 0). This median is only computed from the good pixels if their fraction in a reference region is > some threshold *gfrac* (where currently *gfrac* = 0.5). The 32767 values cannot be reset to zero and included in the median estimate since otherwise the reference signal will be over-estimated. We found that a 'good' reference pixel fraction of > 0.5 minimizes such biases.

The median reference value for each quadrant is then differenced against some long-term median reference signal for that quadrant to compute a relative offset correction. These long-run medians are given by the input *refbaseN* parameters for W3, W4 (where *N* = quadrant 1,2,3,4). These baselines are stable over long periods and appear to represent the natural levels that the reference signals return to following a droop and/or rebound event (see Figure 3). The relative corrections are then applied to the active pixels in each respective quadrant. This brings them in line with what one would have observed if no droop or rebound event occurred.

If the fraction of good reference pixels is ≤ *gfrac* for any quadrant, no correction is applied to the active pixels. Instead, a flag is set to indicate this and propagated downstream to the optional droop-refinement step (see Section 4.5.3).

We close this section with a note on the order of application of the split and Q-to-Q corrections. The splits are corrected first since this removes an ambiguity on which side of a split to correct first. The split corrections are applied to both active and reference pixels in a self-consistent manner so that the updated reference pixels can be used in the Q-to-Q step as described above. Furthermore, this order removes a potential bias on a median reference signal if it turns out a split exactly bisects a quadrant straight down the middle, i.e, 50% of the quadrant pixels lie on one side. The median reference signal will lie somewhere across the split transition (which has some slope) and will not be representative of the reference signal in either half of the drooped/undrooped region. Initial equalization of levels across splits is therefore key before attempting the Q-to-Q correction.

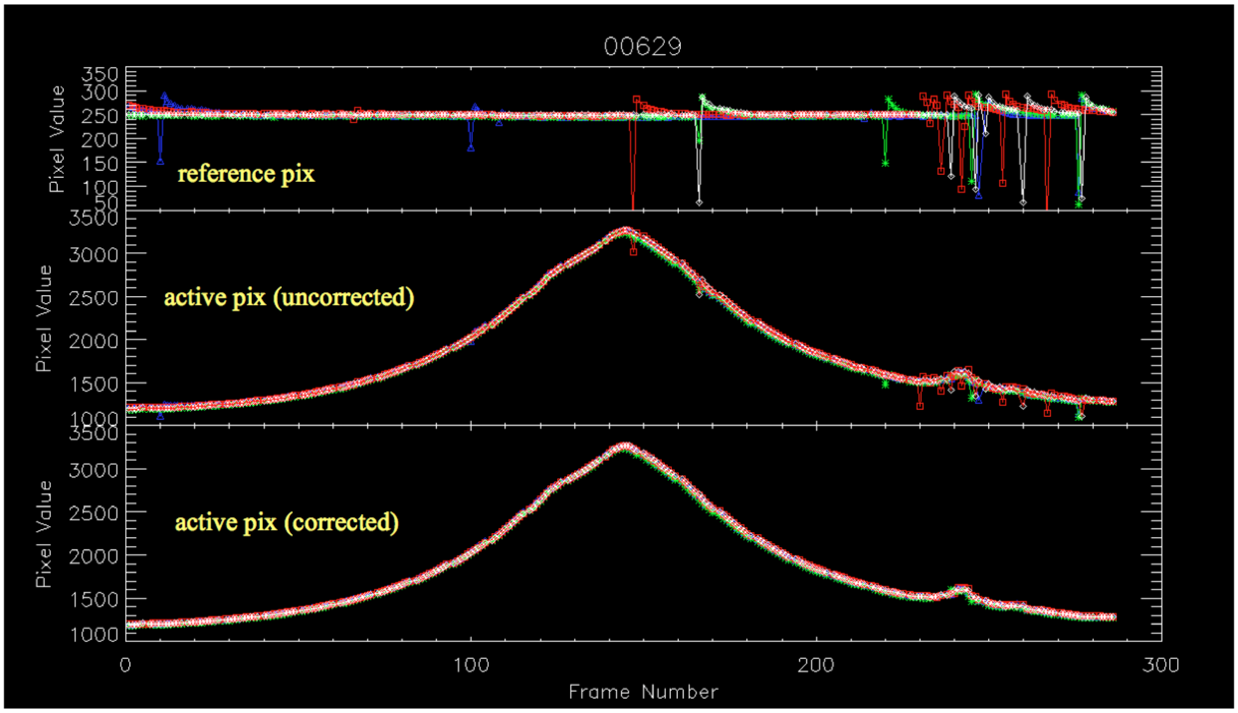

Figure 3: run of reference pixel medians (top), corresponding active pixel medians before Q-to-Q correction (middle), and after correction (bottom) for all frames in a scan. Figure courtesy of Doug Hoffman.

4.5.3 Optional final global droop refinement using active pixels

If a global correction using reference pixels alone (as described in Section 4.5.2) could not be computed for any quadrant, e.g., because there were an insufficient number of good reference pixels (their fraction is < some threshold *gfrac*), we attempt to compute an offset using the active pixels in neighboring quadrants where a correction *was* possible. This step is only performed if the *drpflag* parameter is set to 1 in *wise-meta.tbl*. If set, this step is performed after all instrumental calibrations have been applied in order to minimize biases from quadrant-dependent instrumental residuals.

We first compute lower-tail quantile values corresponding to the *falow* input parameter (currently = 0.1) in strips of width 50 pixels for W3 and 25 pixels for W4 running up and down the center the frame. Figure 4 shows an example of a correction to be computed for Q1. The correction for this quadrant uses the strips colored solid red. If a Q-to-Q droop correction using reference pixels was possible for Q2 (i.e., its good pixels satisfied > *gfrac*), then the offset correction for Q1 (which is added to *all* active pixels of Q1) is given by the quantile difference: *quant*(Q2) – *quant*(Q1). If a correction was not possible for Q2, then we look at the bottom quadrant, Q4, and use the active pixels in its strip to compute the offset correction for Q1, i.e., *quant*(Q4) – *quant*(Q1). The same procedure is used for the other quadrants that failed to be corrected in the initial global Q-to-Q step. E.g., for Q3, the correction sought for is first *quant*(Q4) – *quant*(Q3), but if Q3 failed to be corrected in the Q-to-Q step, we would try using *quant*(Q2) – *quant*(Q3).

If no correction is possible after two attempts to use two neighboring "good" quadrants (because their Q-to-Q reference pixel corrections were not possible), we give up and leave it at that. Note that we avoid using strips aligned in the horizontal direction above/below the quadrants since the existence of incorrectly corrected splits (either from droop or residual banding) could bias the results.

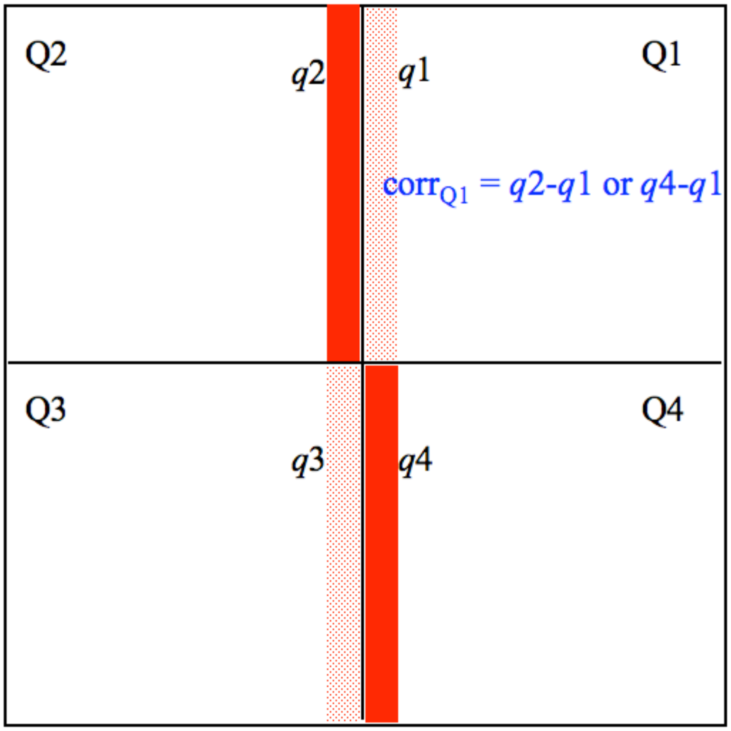

Figure 4: Schematic for final (optional) droop refinement if reference pixel method for global Q-to-Q droop correction for any quadrant could not be performed, e.g., due to an insufficient number of good reference pixels. The solid red strips in Q2 or Q4 are used to correct Q1, more specifically, it's whichever of Q2 or Q4 was successfully corrected upstream using reference pixels.

4.6 Dark Subtraction

The process of dark-current subtraction is a straightforward operation. We should note that any DEB bias (i.e., the implicit offset $O/2^T$ in Eq. 1) is also removed by this step since the input dark calibration is also assumed to contain this bias. This assumes any upstream reference pixel bias subtraction (§4.4) was not applied since otherwise the DEB bias would have been implicitly removed. Following subtraction of the dark calibration, the input uncertainty frame is RSS'd with the dark calibration uncertainty and propagated along.

4.7 Non-linearity Correction

Detector linearity is yet to be fully characterized. Nevertheless, a quadratic correction model has been implemented as a placeholder since experience with these detectors has shown that a quadratic model provides an adequate fit. This model is based on fitting the lab SUR data with the following model:

$$y_i = \alpha i^2 + \beta i, \qquad \text{(Eq. 5)}$$

where

$y_i =$ measured SUR value at sample $i = 0,1,2\ldots8$, and α,β are the fit coefficients.

It is assumed that the SUR data has been zero-baselined so that no intercept is required, i.e., $y_0 = 0$. The fitting procedure to derive the non-linearity coefficients is described in document WSDC D-I005 (referenced in §1.3).

Given that Eq. 5 is the lab calibration fit to specific ramp data, we can generalize to any other observed ramp with underlying *linear count rate* β_l by scaling the "time" *i* at which the linear counts are equal, i.e., if $\beta i = \beta_l i'$ at times *i* and i', then $i = \beta_l i'/\beta$. We can then transform Eq. 5 into a new generic expression for the counts in a ramp:

$$y_{i'} = \left(\frac{\alpha}{\beta^2}\right)\beta_l^2 i'^2 + \beta_l i', \qquad \text{(Eq. 6)}$$

A linear signal output from the DEB (i.e., assuming a detector was perfectly linear) can be written in terms of the true *linear* ramp count ($\beta_l i$) and SUR coefficients using Eq. 1:

$$m_{lin} = \frac{1}{2^T}\left[\sum_{i=0}^{N} c_i \beta_l i\right], \qquad \text{(Eq. 7)}$$

where it is assumed a dark along with the DEB bias ($O/2^T$) has been removed. On re-arranging, the true linear rate can be written in terms of the linear DEB signal as follows:

$$\beta_l = \left(\frac{2^T}{\sum_{i=0}^{N} c_i i}\right) m_{lin}. \qquad \text{(Eq. 8)}$$

Substituting Eq. 8 into Eq. 6, we obtain:

$$y_i = \left(\frac{\alpha}{\beta^2}\right)\Omega^2 m_{lin}^2 i^2 + \Omega m_{lin} i, \qquad \text{(Eq. 9)}$$

where

$$\Omega = \frac{2^T}{\sum_{i=0}^{N} c_i i}.$$

Substituting Eq. 9 into the DEB formula (Eq. 1) and rearranging terms, the observed DEB value (m_{obs}) for a pixel can be written in terms of its linearized counterpart (m_{lin}) *after* subtraction of the dark and DEB bias ($O/2^T$) as follows:

$$m_{obs} = C m_{lin}^2 + m_{lin}, \qquad \text{(Eq. 10)}$$

where $C = \frac{\alpha}{\beta^2} \frac{2^T \sum_{i=0}^{N} c_i i^2}{\left[\sum_{i=0}^{N} c_i i\right]^2}$,

and generally we expect $C \leq 0$ with $C = 0$ implying a perfectly linear response. The quantity α/β^2 in Eq. 10 is provided in an input calibration file (in FITS format for each pixel) along with its uncertainty. C is computed internally in *instruframecal*. Eq. 10 can be inverted to solve for the linearized signal m_{lin}:

$$m_{lin} = \frac{2m_{obs}}{1 + \sqrt{1 + 4Cm_{obs}}}. \qquad \text{(Eq. 11)}$$

The quantity in the square root is the discriminant:

$$D = 1 + 4Cm_{obs} \qquad \text{(Eq. 12)}$$

and we require $D \geq 0$ for a physical solution. If $D < 0$ due to e.g., a bad non-linearity calibration fit, a noisy input pixel, or a ramp which does not follow a quadratic model, we reset this to $D = 0$ implying $m_{lin} = 2m_{obs}$. This is the maximum value possible for a linearized signal under the quadratic model given the input. Another way of saying this is that we require $m_{obs} \leq -1/(4C)$, otherwise, the measurement could not have come from the detector with the given non-linearity model. If $D < 0$, a bit is set in the frame processing mask (§4.2) to indicate an unreliable correction.

The 1-σ uncertainty in a pixel signal after correcting for non-linearity using Eq. 11 can be derived by Taylor expanding about m_{lin} to first order, squaring, and taking expectation values to give the general variance:

$$\sigma^2(m_{lin}) \approx \left(\frac{\partial m_{lin}}{\partial m_{obs}}\right)^2 \sigma^2(m_{obs}) + \left(\frac{\partial m_{lin}}{\partial m_{obs}}\frac{\partial m_{obs}}{\partial C}\right)^2 \sigma_C^2. \qquad \text{(Eq. 13)}$$

Evaluating the derivatives, the uncertainty can be written:

$$\sigma(m_{lin}) = \frac{\sqrt{\sigma^2(m_{obs}) + m_{lin}^4\sigma_C^2}}{1 + 2Cm_{lin}}, \qquad \text{(Eq. 14)}$$

where σ_C is the uncertainty in the calibration factor C. If $D < 0$ in Eq. 12 where we force $m_{lin} = 2m_{obs}$, the uncertainty is reset to $\sigma(m_{lin}) = 2\sigma(m_{obs})$.

4.8 Responsivity (Flat-field) Correction

The responsivity correction uses two input calibrations, the nominal "high-frequency" flat (input –f, and an *optional* "low-frequency" flat (input –lof). The latter is used to correct for throughput/path-length variations as light goes through the entire WISE optical train. This includes surface brightness variations due to distortion, i.e., the convergence and/or divergence of light rays when they reach the focal plane. If specified, the low-frequency flat (f_{lo}) is multiplied into the high-frequency flat (f_{hi}) to yield an effective flat, i.e., $f_{eff} = f_{lo}f_{hi}$. The input frame pixel signals S_i are then corrected for responsivity variations via $S_i \leftarrow S_i / f_{eff}$. The low-frequency flat is expected to be static throughput the mission. It will be derived in IOC and used in processing as described here if variations are significant. The high-frequency flats will initially come from the ground and then created dynamically on-orbit (see document WSDC D-D021 referenced in §1.3) from intermediate products saved from a first pass run of *instruframecal* (see Fig. 1). They are then applied to the science frames on a subsequent rerun of *instruframecal*.

The 1-σ uncertainty in a pixel signal after responsivity correction is computed by combining the uncertainty in the pre-corrected input signal $\sigma(S_{in})$ and uncertainties in the high- and low-frequency flats (inputs –fu and –lofu respectively):

$$\sigma(S_{out}) = S_{out}\sqrt{\frac{\sigma^2(S_{in})}{S_{in}^2} + \frac{\sigma^2(f_{hi})}{f_{hi}^2} + \frac{\sigma^2(f_{lo})}{f_{lo}^2}}. \qquad \text{(Eq. 15)}$$

4.9 Sky-Offset Correction

The purpose of the "sky-offset" calibration (sometimes referred to as an "illumination" correction) is to correct for short-term variations in the bias and/or dark structure over an array that cannot be captured by ground calibrations. The ground calibrations are designed to remove instrumental signatures that are more-or-less static in the long-term. If the short-term systematic variations are not removed, they will persist as residuals and impact photometric accuracy. These can be corrected by computing a zero-mean (or zero-median) image from $N \sim 50$ - 100 frames within a moving block window along the WISE orbit (the "sky-offset" frame), then subtracting this from all the frames in that window. Further details are given in document WSDC D-D011 referenced in §1.3.

The sky-offset calibrations are made by first executing *instruframecal* on on-orbit science frames to create intermediate responsivity-corrected frames (see Fig. 1). These will have a dynacal usability flag (keyword TDYNAFLG = 0 or 1) written to their FITS header indicating whether it's suitable for a skyoffset. TDYNAFLG=1 means the frame is suitable. This flag is determined using a number of QA metrics whose thresholds (at the time of writing) are shown in the *wise-meta.tbl* calibration file (Section 3.2). The responsivity-corrected frames are collated and sky-offsets created by the *dynacal* (*tempcal*) pipeline. This pipeline also detects transient bad-pixels and records them in the dynamic masks. The sky-offsets are then applied to the science frames on a subsequent rerun of *instruframecal*. Following subtraction of the sky-offset from a frame in its respective moving window, the input uncertainty frame is RSS'd with the sky-offset uncertainty and propagated along.

4.10 Single frame outlier/glitch detection

As shown in Figure 1, this step is performed after all instrumental calibrations are applied, but before selected bad pixels in the static input mask are replaced by NaNs in the processed frame. There are two input parameters controlling this step: (i) *–trat*, which specifies the threshold for the ratio: *'regularized pixel value / local median filtered value at that pixel'*. Above this threshold, a pixel is declared an outlier (see below for details) and flagged in the 32-bit mask; (ii) *–ksiz*, which specifies the kernel size *n* of the *n* x *n* median filter. This is an odd number ≥ 3.

The '*regularized pixel value*' is defined as *abs*(*input pixel value - background*) + 1, where *abs* is the absolute value function. The *background* is estimated by computing medians within squares over a 10 x 10 grid on the frame. This block-median image is subtracted from the input frame and a constant of "1" added. This constant is to avoid inadvertent division by zero when the background subtracted frame is divided by the local median filtered image. The local median filtered image is computed using a small *n* x *n* kernel (where currently *n* = 5). This latter operation replaces each pixel by a median of its neighbors and itself. The ratio [*abs*(*input pixel value - background*) + 1] / [*local median filtered value*] is then thresholded using the *–trat* parameter to find and tag outliers, primarily spikes and streaks with hard edges. This ratio is designed such that pixels with soft-edges, e.g., with neighbors whose signal declines no faster than that represented by the PRF are avoided. Note that a kernel size (*-ksiz*) of *n* = 3 (the smallest possible) is conservative and will miss hard-edged clusters of outliers larger or equal to the 3 x 3 kernel size.

5 QUALITY ASSURANCE OUTPUTS

5.1 Metrics

Below we list the intensity (int), uncertainty (unc), mask (msk), and reference (ref) pixel QA metrics computed at the end of processing in *instruframecal*. These are written to a table in IPAC format (–metatbl input). In operations, this table will store the metrics for a frameset (i.e., all associated band-frames) since metrics will be appended on subsequent reruns of *instruframecal*. An example with all columns can be found in WSDC D-I105 (referenced in §1.3). Below we only show the metric names and definitions.

```
|name                     |comment
|c                        |c
  ical:refNum               ref pixels: Number of reference pixels
  ical:refMin               ref pixels: Minimum pixel value
  ical:refMax               ref pixels: Maximum pixel value
  ical:refMean              ref pixels: Mean pixel value
  ical:refMedian            ref pixels: Median pixel value
  ical:refSigMADMED         ref pixels: Sigma scaled from Median Absolute Deviation from Median
  ical:refSigLTMADMED       ref pixels: Sigma from Lower-Tail Median Absolute Deviation from
Median
  ical:refMed16ptile        ref pixels: Median - 16th percentile
  ical:refMed84ptile        ref pixels: 84th percentile - Median
  ical:mskNumGood           msk pixels: Number of 'good' (unmasked) frame pixels in active region
  ical:mskNumTotBad         msk pixels: Number of 'bad' pixels (with mask value > 0)
  ical:mskNumStaticBad      msk pixels: Number of static 'bad' pixels (from ground cal and
updated long-term)
  ical:mskNumDynaBad        msk pixels: Number of dynamically masked pixels in ical processing
  ical:mskNumSat            msk pixels: Number of saturated pixels (at any SUR sample)
  ical:intNumNaN            int pixels: Number of NaNs and |frame values| > machine maximum
  ical:intMin               int pixels: Minimum pixel value
  ical:intMax               int pixels: Maximum pixel value
  ical:intMean              int pixels: Mean pixel value
  ical:intMedian            int pixels: Median pixel value
  ical:intMedITUT           int pixels: Median after Iterative Trimming of Upper-Tail
  ical:intFuzMode           int pixels: (Fuzzy) Mode pixel value
  ical:intStdDev            int pixels: Standard (RMS) Deviation from mean (unbiased population
estimate)
  ical:intSigMADMED         int pixels: Sigma scaled from Median Absolute Deviation from Median
  ical:intSigLTMADMED       int pixels: Sigma from Lower-Tail Median Absolute Deviation from
Median
  ical:intSigLTMADFM        int pixels: Sigma from Lower-Tail Median Absolute Deviation from
Fuzzy Mode
  ical:intSigLTMADITUT      int pixels: Sigma from Lower-Tail MAD from Median after Iterative
Trimming of Upper-Tail
  ical:intMed16ptile        int pixels: Median - 16th percentile
  ical:intMed84ptile        int pixels: 84th percentile - Median
  ical:intMod16ptile        int pixels: Mode - 16th percentile
  ical:intMod84ptile        int pixels: 84th percentile - Mode
  ical:intI16-84Range       int pixels: Inter-84th - 16th percentile range
  ical:intRatMRange         int pixels: Ratio: (84th percentile - Mode) / (Mode - 16th
percentile)
  ical:intMeanCh1           int pixels: Mean pixel value for amplifier channel 1
  ical:intMedianCh1         int pixels: Median pixel value for amplifier channel 1
  ical:intStdDevCh1         int pixels: Standard (RMS) Deviation from mean (unbiased) for
amplifier channel 1
  ical:intSigLTMADMEDch1    int pixels: Sigma from Lower-Tail Median Absolute Deviation from
Median for amplifier channel 1
   ~
   ~
   repeated for channels 2..16 (HgCdTe arrays) or quadrants 2..4 (SiAs arrays)
   ~
   ~
  ical:uncMin               unc pixels: Minimum pixel value
  ical:uncMax               unc pixels: Maximum pixel value
  ical:uncMean              unc pixels: Mean pixel value
  ical:uncMedian            unc pixels: Median pixel value
  ical:uncI16-84Range       unc pixels: Inter-84th - 16th percentile range
  ical:uncRatLTMADMED-Med   unc pixels: Ratio: intSigLTMADMED / uncMedian
  ical:uncRatLTMADITUT-Med  unc pixels: Ratio: intSigLTMADITUT / uncMedian
  ical:uncRat16ptile-Med    unc pixels: Ratio: intMed16ptile / uncMedian
```

Below are descriptions of the not-so-obvious metrics and their purpose, labeled by their generic suffixes as they appear above. We first note that all frame statistics (unless specified in the above list) exclude prior-masked pixels, NaNs, and values exceeding machine minimum/maximum limits. It's also important to note that the various sigma (data-scale) measures are not redundant. It is prudent to measure some of the important metrics in several different ways since a specific algorithm may not be robust against unforeseen glitches and irregular behavior in the data. Each sample metric is also subject to uncertainty and bias (relative to the 'true' value of the underlying population), and it's important not to be mislead by any one of them.

SigMADMED

This is an estimate of sigma using the Median Absolute Deviation (MAD) *from the median*. It is rescaled for consistency with the standard deviation of a normal distribution. This measure is more robust against outliers (including sources) than the sigma estimated from the sample standard deviation. Therefore it can be used to estimate the background RMS, assuming the background level is stationary. We should mention that this measure can still be biased for highly skewed distributions (e.g., severe source contamination affecting the high tail). This metric is defined as:

$$\sigma_{MAD} = 1.4826 median\left\{\left|p_i - median\{p_i\}\right|\right\}.$$

SigLTMADMED

This is essentially the same as SigMADMED above except that only pixel values in the *lower tail* are used when computing the MAD, i.e., with $p_i < median\{p_i\}$. This measure is even more resistant to outliers located in the upper tail (i.e., sources), although it can exhibit more variation (less efficiency) than the SigMADMED on average.

MedITUT & SigLTMADITUT

Both these metrics are returned by the same method. Broadly speaking, these represent the *median* and lower tail σ_{MAD} (using $p_i < median\{p_i\}$) of the pixel distribution after iterative trimming of the upper-tail. The trimming is done incrementally and stops when $mean \leq median$ is satisfied for the remaining sample. The philosophy here is that for a symmetric distribution, *mean* ≈ *median*, and when upper-tail outliers are present, *mean* > *median*. The goal is to remove just enough of the high tail such that the remaining sample can provide a robust measure of the central moment, presumably close to the mode. The upper-tail trimming threshold T_U at each iteration is currently fixed at $5\sigma_{MAD}$ where σ_{MAD} uses only the lower-tail values ($p_i < median\{p_i\}$). After each iteration, T_U is decremented by $\Delta T = 0.1$ relative to the previous value: $T_U \leftarrow T_U - \Delta T$ so that progressively more of the sample can be trimmed. The iterations terminate when either *mean* ≤ *median* or when the maximum number of iterations (= $T_U/\Delta T$; currently = 5/0.1 = 50) is reached, whichever occurs first. A flowchart describing the algorithm is shown in Figure 5.

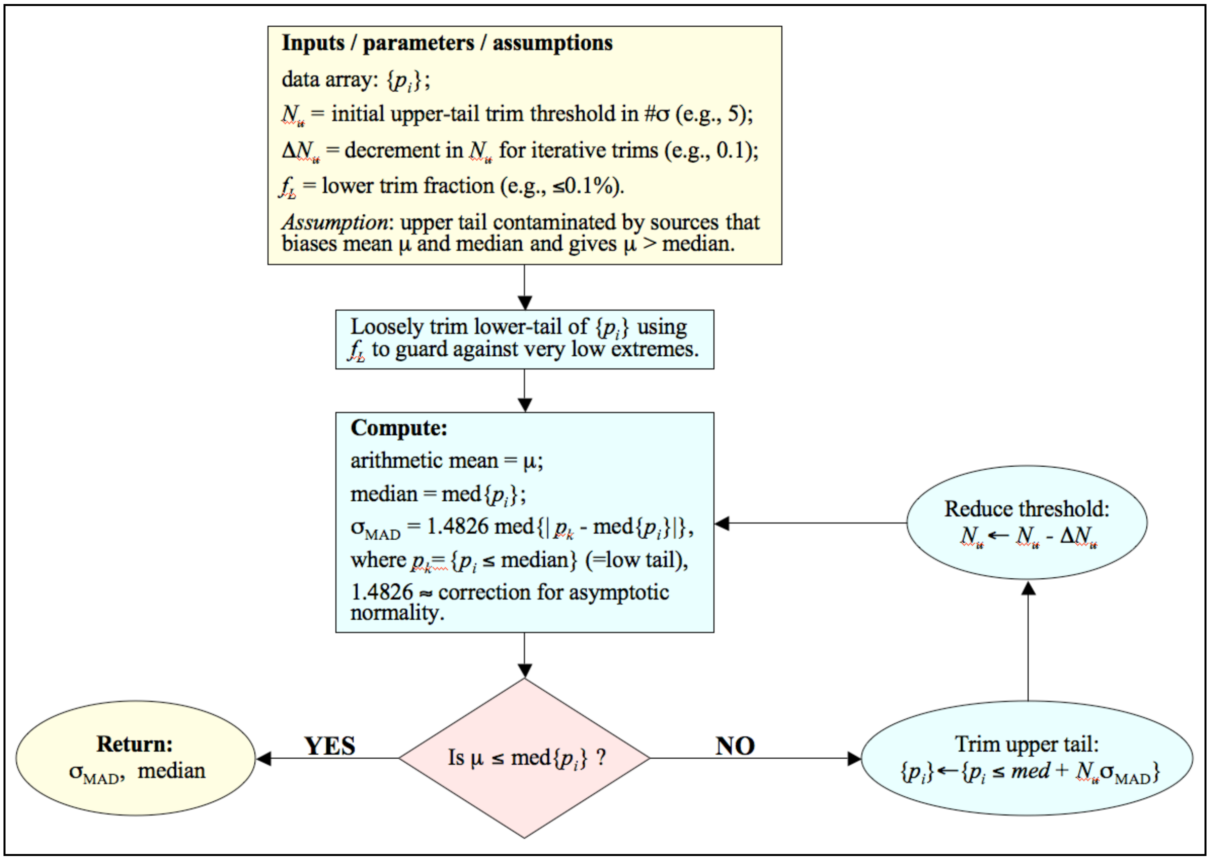
Inputs / parameters / assumptions
data array: $\{p_i\}$;
N_u = initial upper-tail trim threshold in #σ (e.g., 5);
ΔN_u = decrement in N_u for iterative trims (e.g., 0.1);
f_L = lower trim fraction (e.g., ≤0.1%).
Assumption: upper tail contaminated by sources that biases mean μ and median and gives μ > median.
Loosely trim lower-tail of $\{p_i\}$ using f_L to guard against very low extremes.
Compute:
arithmetic mean = μ;
median = med$\{p_i\}$;
$\sigma_{MAD} = 1.4826\ \mathrm{med}\{|p_k - \mathrm{med}\{p_i\}|\}$,
where $p_k = \{p_i \leq \mathrm{median}\}$ (=low tail),
1.4826 ≈ correction for asymptotic normality.
Reduce threshold:
$N_u \leftarrow N_u - \Delta N_u$
Return:
σ_{MAD}, median
YES
Is $\mu \leq \mathrm{med}\{p_i\}$?
NO
Trim upper tail:
$\{p_i\} \leftarrow \{p_i \leq med + N_u \sigma_{MAD}\}$

Figure 5: Sigma estimation from lower-tail Median Absolute Deviation after Iterative Trimming of Upper Tail (the MADITUT method)

FuzMode

This metric represents an estimate of the most frequently occurring pixel value. It represents a "fuzzy" measure since it is based on an approximate binning method and requires a sample size of >~ 500. This is usually always satisfied when all pixels of a WISE frame are used, but the estimate will be highly inaccurate for small samples. The method first partitions the histogram of all image pixel values into 10 equal area bins (or "10%-tiles"); it then takes the bin with the smallest width as the one most probable to contain the mode since this bin is located near the peak of the histogram; the median of the data in this bin is then used as an estimate of the mode.

SigLTMADFM

This is essentially the same as the SigLTMADMED metric but uses the "FuzMode" (see previous description) instead of the median (i.e., lower tail is $p_i < mode\{p_i\}$). It is expected to be more immune to the presence of outliers than SigLTMADMED depending on how well the mode approximates the central moment compared to the median.

Med16ptile & Med84ptile

These represent the quantile differences: $\Delta Q_L = q_{0.5} - q_{0.16}$ and $\Delta Q_U = q_{0.84} - q_{0.5}$ of the frame pixel values. For normally distributed data, these should approximately equal the standard deviation (σ). When there is severe source contamination or bright extended structure, (which affects the high tail), we expect $\Delta Q_U > \Delta Q_L$. Simulations have shown that $\Delta Q_U >\sim 1.5\ \Delta Q_L$ implies the presence of heavy source contamination.

Mod16ptile & Mod84ptile

These are based on the same principle as Med16ptile & Med84ptile above except that the "FuzMode" instead of the median ($q_{0.5}$) is used as an estimator of the first moment. The mode is less immune to "heavy tailed" outliers (or severe source contamination) and hence the Mod16ptile measure provides an even more robust measure of sigma about the background, assuming it's stationary of course.

RatMRange

This measure represents the ratio "Mod84ptile / Mod16ptile", and as described above can be used to infer the level of contamination by outliers (including real sources) in a frame. It is expected to be more sensitive to the presence of outliers (or skewness in general) than the ratio: "Med84ptile / Med16ptile".

RatLTMADMED-Med & RatLTMADITUT-Med & Rat16ptile-Med

These represent the respective ratios:

- **int**SigLTMADMED / **unc**Median
- **int**SigLTMADITUT / **unc**Median
- **int**Med16ptile / **unc**Median

where the metrics therein correspond to the prefixed frame types: **int** = *intensity frame* and **unc** = *uncertainty frame*. These can be viewed as pseudo-χ^2 measures that quantify in a broad sense, the statistical compatibility of uncertainty estimates with the final intensity data. In other words, whether the pixel uncertainties are compatible with the overall RMS fluctuation about the background in a frame. If so, then these ratios should be ≈ 1. Even though robust sigma estimates are used, it's important to note that high source confusion, bright extended structure, and/or a strongly varying background can make these ratios significantly deviate from 1.

5.2 Frame-based Diagnostic Plots and Files

Six other QA diagnostic files are generated. These are written to the directory specified by the –qadir command-line input. Five of these are plots in SVG format:

- **<frameId>-w<band>-intqa-1a.svg :** histogram of the intensity frame pixels.
- **<frameId>-w<band>-offsetcorqa-1a.svg :** histogram of the raw reference-pixels.
- **<frameId>-w<band>-cutsqa-1a.svg :** row and column cut profiles, i.e., median over column/row pixels versus row/column pixel number.
- **<frameId>-w<band>-chanMqa-1a.svg :** first moments (mean and median) versus amplifier channel number 1…12 for bands 1 and 2, or, versus quadrant number 1…4 for bands 3 and 4.
- **<frameId>-w<band>-chanSqa-1a.svg :** second moments (standard deviation and a robust version: SigLTMADMED [see §5.1]) versus amplifier channel number 1…12 for bands 1 and 2, or, versus quadrant number 1…4 for bands 3 and 4.

Another diagnostic is the pixel processing history file (generic name: **<frameId>-w<band>-historyqa-1a.txt**) in ASCII format. This shows the value for a pixel at specified location *x,y* in the intensity and uncertainty frames after each processing step in *instruframecal*. The pixel location is specified via the –xyhist command-line input and must fall in the "active" (non-reference pixel) region of the native frame. An example is as follows:

```
Processing history for _native_ pixel (586 702):

  Step,  Intensity Image,  Uncertainty Image
  lev-0/errmod:  1115,  12.7164335250854
  offsetcor:  1310.59997558594,  12.7164335250854
  darksub:  1037.7685546875,  12.8951892852783
  lincor:  1042.26391601562,  13.007513999939
  flatcor:  1040.61376953125,  13.032377243042
  skycor:  1040.61376953125,  17.7240333557129
  level-1a:  1040.61376953125,  17.7240333557129
```

Acknowledgments

The author is indebted to the Project Office, the folks at SDL, Instrument Calibration Scientists, and QA Scientists at IPAC for illuminating discussions and advice.