ISOCAM Data Users Manual

Version 4.0 (SAI/95-222/Dc)

Ralf Siebenmorgen,
ISO Science Data Centre, Astrophysics Division of ESA,
Villafranca del Castillo, P.O. Box 50727, 28080 Madrid, Spain

Jean–Luc Starck,
CEA, DSM/DAPNIA, CE-Saclay, F-91191 Gif-sur-Yvette Cedex, France

Marc Sauvage
CEA, DSM/DAPNIA, CE-Saclay, F-91191 Gif-sur-Yvette Cedex, France.

Diego A. Cesarsky,
IAS, CNRS, Bat 121, University of Paris Sud, F-91405 Orsay, France

Joris Blommaert
ISO Science Data Centre, Astrophysics Division of ESA,
Villafranca del Castillo, P.O. Box 50727, 28080 Madrid, Spain

Stephan Ott
ISO Science Data Centre, Astrophysics Division of ESA,
Villafranca del Castillo, P.O. Box 50727, 28080 Madrid, Spain

December 1, 1998

Contents

Chapter 0

Change log

The present version of the ISOCAM Data Users Manual replaces all previous versions.

Compared to earlier versions the manual has been re-structured, with respect to V3.0 the following changes have been applied:

- We include a new Chapter entitled "How to use this manual" (Chapter 2).
- Auto-Analysis.
  Chapter 5, refers to the off-line-processing version (OLP V7.0) of the pipeline.
- ISOCAM Photometry
  Beside a discussion on colour correction there is now some general information on ISOCAM photometry.
- Global error budget.
  Section 7.4 is based on in-orbit measurements.
- Data processing methods
- Caveats

  Caveats applicable to off line processing version OLP 7.0 (Appendix C).

Chapter 1

Introduction

1.1 Purpose and Structure

The ISOCAM Data User Manual is one in a series of five documents that globally explains the ISO data. Four of the five explain the contents of data files from an individual instrument, while the fifth explains the contents of the general ISO Satellite data files. This document describes the files associated with ISOCAM.

The purpose of this document is to indicate to the observer:

- How ISOCAM operates ?
- What processing steps have been carried out on the original data ?
- Where do the uncertainties lie in the data ?
- What other processing can be carried out on the data ?

The document is structured as follows: we give a brief overview of the ISOCAM instrument, its astronomical observing templates (AOT) and the ISO data processing (Chapter 1). A summary of the various ISOCAM data products is given in Chapter 3. The tremendous number of individual calibration measurements are archived in calibration libraries, which are described in Chapter 4. The ISO data are run through an automatic analysis system. An overview of this system is given in Chapter 5. The absolute flux calibration is briefly described in Chapter 6.

Generally an automatic processing of data cannot be as accurate as an interactive analysis. Observers are invited to re-analyze their own data. We give some ideas about individual processing steps in the reduction of ISOCAM data and start with a description of a basic model of ISOCAM. In order to better understand whether an interactive processing is necessary we give some global estimates of uncertainties in the ISOCAM data. All this can be found in Chapter 7. A tutorial of how an observer can start analyzing his data is presented in form of a guided tour in Chapter 8. An introduction to enhanced algorithms for individual data processing steps is given in Chapter 9. Several appendices are attached to give more detailed information on individual subjects. In particular Appendix C should be of interest since it describes the caveats applicable to OLP V7.0 of the automatic processing system.

1.2 How to reach us:

You may contact us via ISO helpdesk:
`helpdesk@iso.vilspa.esa.es`

The information necessary to make best use of your ISOCAM data can be found on our instrument page: `http:/www.iso.vilspa.esa.es`

1.3 Instrument and AOT overview

ISOCAM is the instrument on ISO designed to map selected areas of the sky in the spectral region from 2.5 to 17 μm at various spatial and spectral resolutions. Polarization mapping is also possible.

With ISOCAM one can study spectral features at wavelengths inaccessible from ground; one example is imaging of water vapor emission in the circumstellar environment. The morphology of spatial structures at a scale of a few arcsecs can be investigated in broad and narrow band filters. Spectral imaging is possible at a spectral resolution of up to $\Delta\lambda/\lambda = 50$.

The spatial resolution is limited by the diffraction at longer wavelengths, and by the pointing of the satellite at shorter wavelengths. For a 60 cm diameter telescope the diameter, in arcsec, of the Airy disc $2.4\lambda/\mathrm{D} = 0.8\lambda(\mathrm{micron})$ for the first dark ring. Regarding the optical performance of ISO, images of point sources have been made clearly showing the first Airy diffraction ring, one example is given in Figure 1.1. Some more information on the telescope and the pointing system is given in Sections 7.3, 8.4 and for a detailed description please consult the ISO Satellite Data User Manual.

ISOCAM provides imaging capabilities across a field of view of up to 3 arcmin diameter. It consists of two optical channels, one used at a time, each of which contains a 32×32 pixels infrared detector array. The short wavelength channel (SW) operates between 2.5 and 5.2 μm, the long wavelength channel (LW) between 4 and 17 μm. Each channel includes different magnification lenses (1.5, 3, 6 and 12 arcsec), filters and CVFs. During the ISO thermal vacuum test it was discovered that the output voltage of one column of the LW detector (column 24) was always at zero Volt. In space it was found that because of a design feature of the Fabry mirror another column (column 0) does not always get sufficient light when using the 3 and 6 arcsec lens.

ISOCAM has three main imaging modes: photometric, spectrophotometric and polarimetric. Additionally, in the photometric mode, celestial raster and beam-switching, where observations on the source and on one or more reference positions are carried out, can be performed.

These observation modes are accessible via three dedicated AOTs:

- CAM01 – general observation, single pointing, raster,
- CAM03 – beam-switching,
- CAM04 – spectrophotometry.

Polarimetric observations require the use of the Calibration Up-link System (CUS), rather than being programmed through an AOT. The same holds for the limited number of observations in coronographic mode.

For a detailed description of the instrument and instrument modes please consult the ISOCAM Observer's Manual [24], [34].

1.4 Processing overview

Here we will give a general product description and review the scientific treatments and processing steps applied. ISOCAM provides its user with a total of 41 different ISOCAM specific data products. In addition the user will get three more general ISO products (Files: IIPH, IRPH, ORBIT), for pointing verification. All of the ISOCAM specific products result from one of the following tasks:

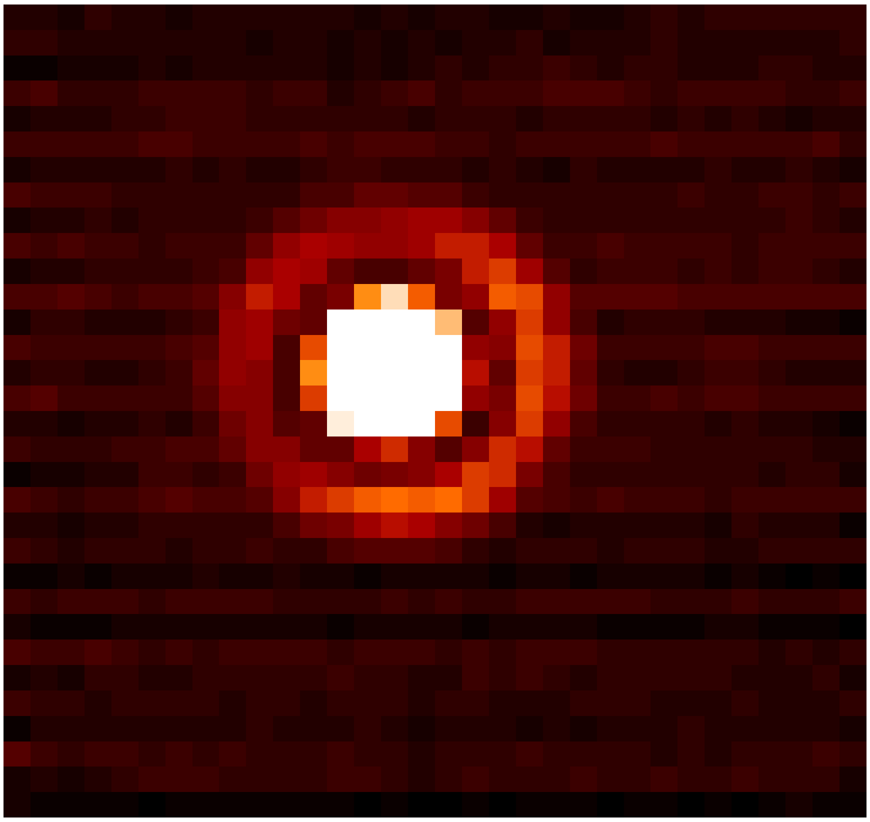

Figure 1.1:

- The Quick Look Assessment task, which analyses on-line the data coming down from the satellite.
- Analysis of in-orbit calibration activities, dark current, flat–field, point-spread function, etc.
- The automatic off-line data analysis chain. All AOT based observations are automatically processed, so as to provide the observer with a first look at the scientific value of the data.
- The ground calibration tasks.

The products are written following the definition of the NOST [29], Flexible Image Transport System (FITS: see NASA/Science Office of Standards and Technology, NOST standard 100-1.0 [29]). For each ISOCAM data product, a full description of its layout, namely the FITS keywords and headers, can be found in the ISO Data Product Document [25] or by reading the FITS header of your ISOCAM. files. In this document you will find more information on the product layout in Section 3.2.4 and Chapter 8.

A global overview of the processing is given in Figure 1.2.

If you would like to check immediately how your observations went, please display some of the file(s) called:

CMOS, CMAP, CUFF, CPSL, CCIM, CSSP

by following the instructions given in Section 8.1 and examine the contents of the CUFF file (Appendix A). Note that the presence of one or all of these data products depends on the AOT used in the observation. CMOS which contains a mosaic of fields, is obviously irrelevant to a staring observation.

The ISO processing is an automated process where the data products:

- Edited Raw Data (ERD),
- Standard Product Data (SPD) and
- Automatic Analysis Products (AA).

are quality checked. This automated process is also referred to as the pipeline. The data coming from the satellite is sent to the ground station in so called Telemetry Distribution Files (TDF). These are binary files that contain the data in a coded (instrument specific) way. These files also contain all the information on the satellite and its subsystems.

We deliver to the user products with information about :

- status of the instrument,
- status of the telescope,
- pointing history,
- and his science data.

Scientific and auxiliary data are extracted from the Telemetry Distribution File in the program *TDF first scan*. TDF first scan produces only files that are used internally in the ISO ground segment and that are needed for the automatic generation of the data products. The scientific data is extracted from the Telemetry Distribution File, decoded and written to the Edited Raw Data (ERD) in *Derive_ERD*. The ERD can be used by the observers as raw data file. It contains all science data for the observation. The ERD contains the scientific data in the form of raw detector readouts, without any calibration or conversion being done.

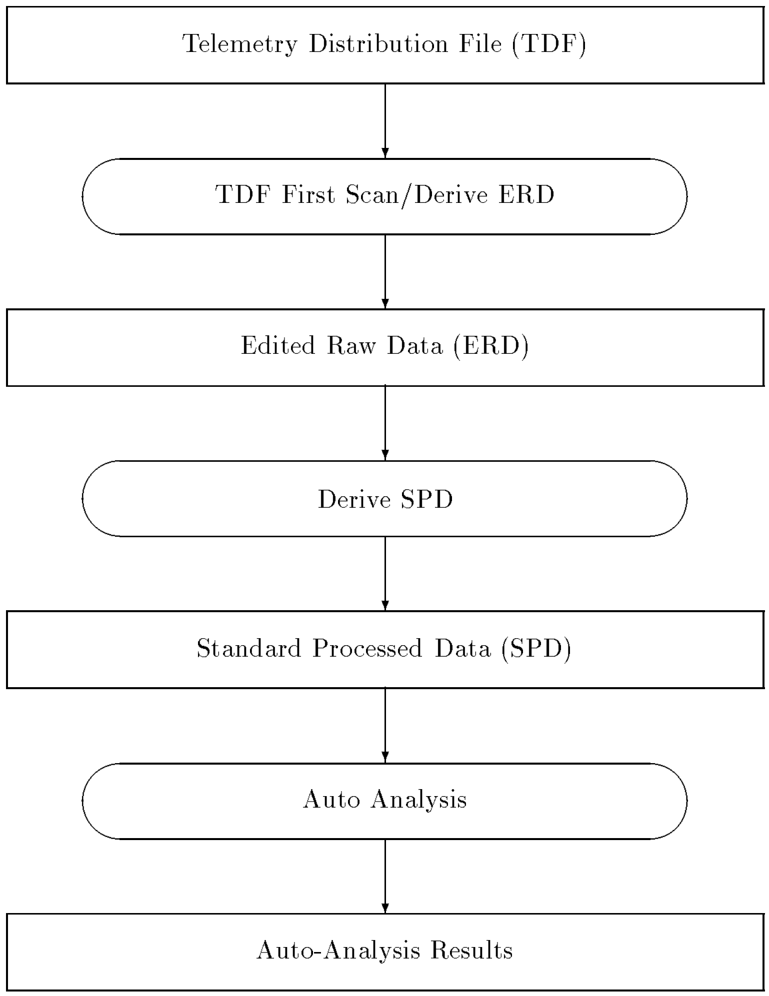

Figure 1.2: Overview of the ISO data processing, called the pipeline.

Then the Standard Product Data (SPD) are derived in the program called *Derive SPD*. Processing steps have been applied to extract most of ISOCAM status and housekeeping information plus statistics on detector data and images have been constructed from the reset and end-of-integration frames present in ERD[1]. SPD is the starting point for the automatic analysis chain (Chapter 5), and generally, the starting point for interactive analysis sessions (see Chapter 8, though the interactive data slicer in CIA (ISOCAM interactive analysis package) starts from the ERD level). For description of the auto-analysis products see (Chapter 5).

[1]Note that to allow for a monitoring of the state of the detector, the reset frames are kept at the SPD stage.

Chapter 2

How to use this manual

How to use this manual depends on what your goals are. If this is the first time you access ISOCAM data you should read Chapters 1 to 4. If you wish to inspect your data go to Chapter 8, where you find a kind of tutorial of data manipulation and visualization. Chapter 7 gives a very basic model of the instrument and consequently outlines the fundamental processing steps which must be applied to get scientifically useful data. To inspect results as given by the automatic processing system you should go to Chapter 5 and Appendix A. The absolute flux calibration is briefly described in Chapter 6.

If you have received already ISOCAM data and you are interested in the most recent results of the automatic processing system then read carefully the caveats given in Appendix C.

We invite astronomers for an interactive analysis of their ISOCAM data, best done with the ISOCAM interactive analysis system, CIA [32] and following the instructions given in the CIA Users Manual [17]. For further information on CIA and how to get it see the ISOCAM instrument page at:
`http://www.iso.vilspa.esa.es`
Some illustrating examples on how different processing methods and algorithms may improve the quality of the final ISOCAM results are given in Chapter 9.

2.1 Reference list of detailed ISOCAM reports

For more experienced users we already here refer to technical reports which give the most detailed description of the calibration status. In order to retrieve these reports please check the ISOCAM web page `http:/www.iso.vilspa.esa.es`

- Transient Behavior of LW Channel of ISOCAM [1], Abergel et al., 1998.

- Comparison of ISOPHOT and ISOCAM/IAS approaches for transient [2], Acosta, 1998.

- ISOCAM Faint Source Report [4], Altieri et al., 1998.

- CAM Calibration Explanatory document [3], Altieri et al., 1998.

- Field distortion for LW [5], Aussel, 1998.

- ISOCAM Dark Current Calibration Report [6], Biviano et al., 1998.
- ISOCAM Flat-field Calibration Report [7], Biviano et al., 1998.
- ISOCAM CVF Calibration Report [8], Biviano, et al., 1998.
- ISOCAM Calibration Error Budget Report [9], Biviano, 1998.
- ISOCAM Photometry Report [10], Blommaert, 1998.
- The ISOCAM Detectors: An Overview [11], Boulade & Gallais, 1998.
- Glitch Effects in ISOCAM Detectors [14], Claret et al.,1998.
- ISOCAM Glitch Library [15], Claret & Dzitko, 1998.
- CIA User's Manual [17], Delaney et al., 1998.
- Cosmic Ray effects on the ISOCAM LW detector [19],Dzitko et al, 1998.
- Report on Trend Analysis of CAM daily Calibration Measurements [20], Gallais, Boulade, 1998.
- ISOCAM PSF Report [30], Okumura, 1998.
- Ghosts in ISOCAM Images [31], Okumura et al., 1998.
- Addendum to the ISOCAM Observer's Manual [34], Siebenmorgen et al., 1996.
- ISOCAM Observer's Manual [24], The ISOCAM Team, 1994.

Chapter 3

The Data Products

3.1 Overview

Though the data products that are distributed are numerous and different in content, they can be ordered in four main categories:

1. Satellite data products (see the ISO Data User Manual [27]). These mainly give information about the attitude of the satellite, which will be useful to carefully reconstruct the images, especially in case of mosaics. Note that this information, i.e. actual coordinates rather than intended ones, is added to each frame header in the CISP.

2. Low-Level ISOCAM products (Section 3.2). Low level implies that few data processing steps have been applied. These products can either be a rearranged version of the Telemetry Distribution File in order to have a more user friendly format or log files describing what actually happened during the observation, or parameter tables such as those giving the correspondence between filter and motor step on the filter wheels.

3. High-Level ISOCAM products (Section 3.2). These contain processed data, as well as some by-products of the processing algorithms (e.g. glitch list). The algorithms used by the Automatic Analysis are described in Chapter 5 of this manual.

4. Calibration products. These contain all the calibration information required to correctly reduce the observations. They are described in Chapter 4 of this manual.

3.2 ISOCAM data products

In this section we describe the data products that fall into the second ("Low-Level ISOCAM products") and third ("High-Level ISOCAM products") category listed above. We give a simple description of the content and use of the products.

3.2.1 Basic definitions

Here we review some definitions, around which most of the products are constructed. There are four basic concepts: the STATE, the operational mode (OP-MODE), the CONFIGURATION and the Astronomical Observation Template (AOT).

1. A STATE corresponds to the finest time division of ISOCAM activities where it is either performing a pointed observation, or at a steady pointing during a raster or a beam-switch observation, or at a steady CVF position in the course of a CVF scan, or at a steady position of the entrance wheel during a polarization measurement.

2. An operational mode (OP-MODE) is a larger time division of ISOCAM activities since it corresponds to one of the following actions:

   (a) ISOCAM is obtaining astronomical data (OBS), in which case, the OP-MODE can be a succession of STATEs.

   (b) ISOCAM is performing a dark current measurement, DRK.

   (c) ISOCAM is performing an internal flat–field calibration, CAL.

   (d) ISOCAM is cleaning one of its detector (CLN).

   These operational modes should be familiar to those astronomers having entered their observations via the Proposal Generation Aids (PGA).

3. a CONFIGURATION is a sequence of chronologically ordered OP-MODEs containing one and only one OBS operational mode and corresponding to the observation of a celestial source. Inside a CONFIGURATION, the following items have one fixed value:

   (a) the beam, i.e. the Fabry-mirror and detector

   (b) the lens, pixel-field-of-view (pfov),

   (c) the integration time (T_{int}),

   (d) the gain,

   (e) the number of stabilization frames (N_{stab}) and exposures (N_{exp}),

   (f) the filter, or, for a CVF observation, the wavelength interval and scanning step.

   With these parameters, the CONFIGURATION will consist of:

   (a) one staring mode observation, or

   (b) one observation of all points in a raster, or

   (c) one observation of all positions in a beam-switch, or

   (d) observation of all requested wavelengths in a CVF scans, or

   (e) observation of all positions of the entrance wheel in a polarization measurement and,

   (f) if requested at the PGA stage, in the order in which they were requested:

   (g) dark (DRK), and/or

   (h) a clean (CLN), and/or

   (i) a flat-field (CAL).

   There can be at most four OP-MODEs in a CONFIGURATION, and each of the four flavors of OP-MODEs: OBS, DRK, CAL or CLN can appear only once. The two following examples show valid CONFIGURATIONs which produce different data outputs:

   CONFIGURATION 1: OBS – CAL – DRK,
   CONFIGURATION 2: DRK – CAL – OBS.

   Both examples give different results. The stabilization time of the detector becomes long when coming from a dark level and the first CONFIGURATION should have been entered into Proposal Generation Aids. It should become clear that the chronological order of OP-MODEs within a CONFIGURATION is important.

4. The Astronomical Observation Template (AOT) is a collection of CONFIGURATIONs performed on one given source position.

The concepts of CONFIGURATION and operational mode are central to the understanding of the contents of ISOCAM's Compact Status History and Standard Product Data files.

Also of importance regarding the data products is the way the detectors have been operated: at a given time, i.e. the start of the individual integration, the pixels are reset to a constant charge which is then read. Then the detector is exposed to the incoming radiation flux and after the selected integration time, the charges are read again. Thus for an individual exposure, there will actually be two frames, called respectively the reset and the end-of-integration (EOI) frames. These two frames are present in the Telemetry Distribution File (TDF) except in the case of $T_{int} = 0.28$s or parallel mode, where an on-board integration takes place. Of course the meaningful exposure is end-of-integration (EOI)−Reset. These two frames will be stored in the Edited Raw Data (ERD). The exposure (end-of-integration (EOI)−Reset) and the reset level are stored in the Standard Product Data (SPD).

3.2.2 Inventory and naming convention

By convention, all products are described by an acronym beginning by C for ISOCAM; the rest of the name tries, but not always succeeds, to provide a mnemonic way of knowing what is inside the product. Here is the list of Data products:

ISOCAM "low–level" products

CSTA	ISOCAM Compact Status
CIER	ISOCAM Image ERD (Edited Raw Data)
CISP	ISOCAM Image SPD (Standard Processed Data)
CPER	ISOCAM Parallel ERD
CPSP	ISOCAM Parallel SPD

ISOCAM "high–level" products

CUFF	ISOCAM User-Friendly log File
CCIM	ISOCAM Images averaged per state, in detector units and coordinates
CMAP	ISOCAM Images averaged per state, calibrated and in celestial coordinates
CMOS	ISOCAM Celestial image mosaic, one image per configuration, calibrated and in celestial coordinates; exists only when more than one position on the sky was observed in the AOT
CGLL	ISOCAM Glitch List
CJAM	ISOCAM Jitter, Memory and Stabilization information
CPSL	ISOCAM Point-Source List
CSSP	ISOCAM Spectra; spectra are computed in case sources are detected at more than one wavelength.

"Basic" calibration files

CSCGCROSS	ISOCAM SW noise/cross talk decorrelation matrices	
CHCGCONV	ISOCAM House keeping interpolation values	
CCG*DEAD	ISOCAM Dead Pixel Map	
CCG*DARK	ISOCAM Dark Current Exposures	
CCG*DFLT	ISOCAM Detector Flat Field Library	
CCG*OFLT	ISOCAM Optical Flat Field Library	(*SW, LW)
CCG*SPEC	ISOCAM Filter & CVF spectral data	
CSWCVF	ISOCAM SW CVF description	
CLWCVF1	ISOCAM LW CVF1 description	
CLWCVF2	ISOCAM LW CVF2 description	
CCG*SLP	ISOCAM CVF spectral line profile	
CCG*PSF	ISOCAM Point-Spread-Function Library	

"High–level" calibration libraries

CCGLWDMOD	ISOCAM LW parameters for the time dependent dark corrections	
CCG*TRANS	ISOCAM Model transients	
CCG*LINEAR	ISOCAM Linearity Correction Library	(*=SW, LW)
CCG*FRAME	ISOCAM Detector astrometric calibration	
CCG*GLITCH	ISOCAM Glitch Model	
CCG*STRAY	ISOCAM Non-dark local light model	

3.2.3 Global organization and relationships

Please select here whichever products you would like to have more information on and consult the subsections specified.

CHCGCONV and CWHEELS are observation-independent as they contain tables translating motor steps on the wheels into meaningful quantities such as filters, lens, etc. (Section 3.2.5).

CUFF and CSTA are information files where one can find respectively the log-file of the Automatic Analysis system and the chronological list of achieved ISOCAM operational modes (OP-MODEs) (Section 3.2.6).

CIER contains the observation's raw data and all of the information on the status of each configuration during the observation (so-called housekeeping data). CPER is the same for parallel mode data. CISP is a more elaborate version of the data than CIER most of the housekeeping information is removed, and images are constructed from the reset and end-of-integration frames present in CIER. The product CISP is the entry of the automatic analysis chain (see Chapter 5), and again CISP or CIER represent the starting point for an interactive analysis session (see Chapter 8). CPSP is similar to CISP but for parallel mode data.

For quick look, CCIM is created. CCIM contains a pure STATE by STATE averaged image in detector coordinates and includes as well associated statistical information describing the averaging process. There is a list of detected glitches in the product CGLL and information on satellite jitter, memory and stabilization can be found in CJAM. Both, CGLL and CJAM, are calculated on a frame by frame basis and are by-products of the analysis chain. CPSL is the catalogue of point sources in the map. CMAP is the calibrated and referenced to celestial coordinates version of CCIM. CMOS is a mosaic made of the elements of CMAP belonging to the same configuration. Obviously, CMOS will only be constructed in case of AOT01 raster or micro–scan and AOT03. In case point sources are detected in more than one filter a spectrum is calculated and stored in CSSP. Products of the automatic analysis chain are described in Chapter 5.

3.2.4 General principles of the FITS structure

ISOCAM generates a very high number of images per observation. Furthermore each of these images comes with a large number of auxiliary parameters giving, e.g. the temperatures of the various subsystems, the position of the wheels and so on.

Rather than producing one FITS file per image, it was decided to keep all the images belonging to one AOT into the same file. This has the advantage of keeping the chronological sequence in order and allows a relatively easier access to trend analyses. (Furthermore, it reduces drastically the amount of files.)

However, this choice results in a rather complicated FITS structure. Indeed it is possible to store a cube in a FITS file, but it is much more complicated to store keywords taking different values for each plane of the cube. Thus it was decided to treat all the information of an exposure, pixel data and keyword values, in the same way, i.e. as binary values, arrange them in lines, one per exposure, and store these lines in a binary FITS table.

The header of this table thus contains (1) standard FITS keywords describing, e.g., the origin and date of creation of the data, (2) ISO specific keywords describing the general parameters of the AOT, the version of the pipeline used to handle and derive the product, and (3) the specific content of the binary table, that is, in the order in which they appear on the line, the name or description of the data and their format.

While ISO products, given as FITS binary table extensions, confirm to the FITS standard, they can not be read by various astronomical packages (which did not implement access to the full FITS standard). Therefore specific tools are provided to transform these FITS files into the most basic FITS structures. Please browse `http://www.iso.vilspa.esa.es` to down load the software called flcam.

Figure 3.1 schematically describes the process leading from observation to FITS files, and the organization of these FITS files, taking the example of the CIER product.

3.2.5 Time independent products

There are two ISOCAM products assumed not to change. They are called CHCGCONV and CWHEELS.

CHCGCONV – ISOCAM House keeping interpolation values

This file translate the housekeeping values into physical units.

CWHEELS – ISOCAM Wheel information table

This file relates the motor step number for the

entrance wheel,
selection wheel,
SW lens wheel,
SW filter wheel,
LW lens wheel or
LW filter wheel,

with the optical name in the path such as "HOLE", "LW10", etc. In the case of the filter wheel, the corresponding central wavelength and filter resolution are specified.

The information regarding wheel positions is always stored as motor steps in ISOCAM data products, CIER, CISP, CCIM, CMAP and CMOS.

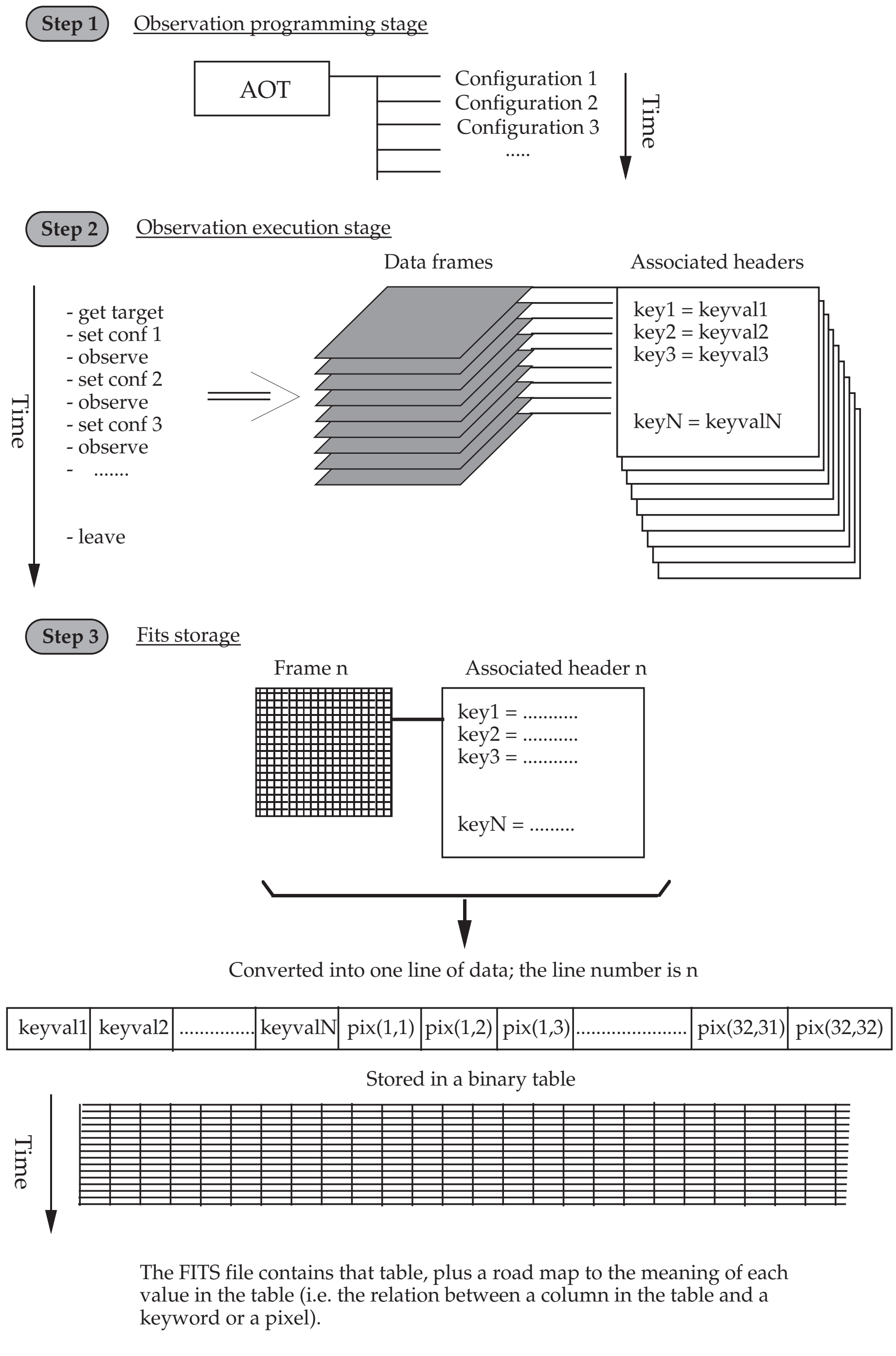

Figure 3.1: A schematic drawing of the process leading from observation to the CIER FITS product. In step 1, the observer enters into Proposal Generation Aids the configurations with which he wishes the observation to be done. In step 2, the satellite receives the order to slew to the observer's target. Once this position as been reached, the On Target Flag is set to 1 and the observation begins. In fact, while it is on, ISOCAM is continually reading its detector and therefore is producing a stream of frames with related keywords. In step 3, each frame and its associated header are converted to integer values and stored in a binary FITS table. The header of the FITS file describes how the values are arranged in the table so that the observer can sort the data.

3.2.6 Information files

There are basically two information files for the user, called CUFF and CSTA.

CUFF – ISOCAM User-Friendly log File

The CUFF file contains the processing history (Section 5.2.8). While by necessity it is a FITS file (since it is a product), the contents of the table are essentially an ASCII file. An example CUFF table is shown in Appendix A. In the file the observer can trace the processes leading to CCIM and CMAP, check the versions of the algorithms for internal consistency of his data, and get an idea of how to proceed for an interactive analysis.

The EOHA, EOHI and EOHC files also contain information about the observation. These files are not ISOCAM specific, therefore we refer to the ISO Data User Manual [27].

CSTA – ISOCAM Compact Status

CSTA summarizes the chronological order of operational modes (OP-MODEs) actually performed during the Target Dedicated Time (TDT) of an AOT, thus it is very useful to search for any differences between what was entered in PGA and what was executed. CSTA is distributed as a binary table FITS file. The header of CSTA basically contains the mandatory FITS keywords necessary to describe the binary table and the origin of the product, plus a number of ISO specific keywords, pertaining either to the observing mode or to the version of the pipeline used to create and handle the product.

The contents of the binary table records are ISOCAM specific. Each line in the table refers to an operational mode (OP-MODE) and describes the succession of all STATES performed during this operational mode. As mentioned above the lines of the binary tables only contain keyword values. The keywords are described in the header of CSTA. This allows to completely reconstruct the configuration of ISOCAM during the operational mode. (For more examples of FITS headers of ISOCAM products please consult the ISO Data Product Document [25].)

Miscellaneous notes regarding the contents of the CSTA:

- The UTC keywords go by pair, the first value is the number of seconds since 1989.0 and the second the remaining fraction of a second (see Section 8.3.1).
- Wheel positions are expressed in motor step counts. Consult CWHEELS to find the correspondence between motor step and optical component (per wheel basis). Files CSWCVF, CLWCVF1 and CLWCVF2 give the correspondence between step counts and CVF nominal central wavelengths.
- In the case of CAM04 (spectrophotometry), the filter wheel position reported in the CSTA file is irrelevant since the wheel steps forward or backwards during the operational mode. There is only one filter wheel position per OPMODE in CSTA.

3.2.7 Entry products for data reduction

Data reduction for ISOCAM can start with either the product CIER or the CISP.

CIER – ISOCAM Image ERD

The CIER is the lowest level data produced delivered by the pipeline. The CIER file is ISOCAM's realization of the so called Edited Raw Data (ERD). By definition, the CIER must contain only raw data; furthermore the data is cast in "natural" computer datum units, viz bytes, integer*2, integer*4, etc. By definition, there is only one CIER per AOT. At this level only a regrouping of records from the raw Telemetry Distribution File takes place but the records themselves remain untouched.

To better understand the contents of the CIER file some digression on ISOCAM's telemetry is in order. ISOCAM down-links two different data streams, called F1 and F2 blocks. These blocks are "floating" in the telemetry flow: they may start anywhere in the telemetry and in fact may even cross telemetry format boundaries, i.e. start in telemetry format NN and end in format NN+1.

The F1 block (length 326 bytes) contains most of ISOCAM's house-keeping parameters and some micro-processor status words; the F2 block (length 2302 bytes) contains all the science data and some selected house-keeping parameters and processor status words. F2 blocks are generated by ISOCAM's processor and sent to the service module for inclusion on the telemetry flow at the end of an on-chip integration period. In fact, two F2 blocks are generated: one containing the end-of-integration pixel voltages and a second one containing the pixel reset voltages. Note that for the so called on-board processing modes, i.e. when the arithmetical difference between end-of-integration and reset voltages is performed by the on board processor, only one F2 block is created. Note also that when two F2 blocks are generated per integration, the processor has been programmed to insert one F1 block immediately after each F2 block. This F1 block contains house-keeping information prevailing at the time the preceding F2 block was generated.

The rate of creation of F2 blocks is related to the on-chip integration time. While the integration proceeds, and therefore no F2 blocks are created, the on-board processor down links F1 block at the maximum possible rate allowed by the telemetry bandwidth: 24 science frames every two seconds, 244 bytes per frame, i.e. 2928 bytes/s. ISOCAM's telemetry then would look like:

```
......F2 F1 F2 F1 F1 F1 (more intervening F1s) F2 F1 F2 F1 F1 ....
```

for no processing on-board; and

```
......F2 F1 F1 F1 F1 (more intervening F1s) F2 F1 F1 F1 .....
```

when data is processed on board.

The requirement was to create a file containing identical size records; this precluded the obvious choice of a CIER holding F2s and F1s records in chronological order. The chosen layout is of the form F2+F1, where the F1 is the block immediately after the F2 block. Notice that for long integration times, a rather large number of F1 blocks will not make it into the CIER and thus the CIER may severely under sample the chronological evolution of ISOCAM's house-keeping.

We may now describe the step used to create CIER (simplified here to not describe the error handling):

1. read the raw telemetry
2. detect an F2 block
   – unpack the raw data, i.e. extract bits from the raw telemetry and copy them to one CIER record as BYTE, INTEGER*2, etc.
3. continue reading raw telemetry
4. intercept the immediately following F1 block
   – unpack the raw data, i.e. extract bits from the raw telemetry and copy them to the remaining of the CIER record as BYTE, INTEGER*2, etc.
5. go back to 1

To summarize: CIER contains all raw telemetry generated by ISOCAM, cast in "normal" computer data type descriptions, at a sampling rate equal to the on-chip integration time. The ISO Data Product Document [25] gives the complete description of the contents of the CIER file.

Most of the keywords relate to house-keeping data. They are defined to fully describe the status of ISOCAM and thus most of them are not of much interest for a general user. However, they can be extremely useful to understand why an observation has failed. Therefore, the observer should not discard the CIER for the reason that most of the interesting information can be found in the CISP.

Examples of house-keeping data are:

settings and status of the black-body, cryo-motor,
various temperatures e.g. of the detectors,
address and contents of RAM/ROM,
s/w status and error codes,
general settings of voltages, etc.

Besides the detector data and information not contained in CISP (see below), the user should be mostly interested in the following keywords:

- Quick-look flag, when ON, indicates that ISOCAM has completed the user-selected waiting period (number of stabilization frames selected in PGA) and a scientifically meaningful observation is taking place. Thus by selecting frames in the CIER that have the Quick-Look flag ON, the observer selects the observation frames of the AOT (remember that ISOCAM detectors are read continuously during the AOT, not only when science can be achieved).
- On Target Flag (OTF), when ON indicates that the intended coordinates for the observation have been reached within the pointing uncertainties.

Finally the CIER contains information such as:

- Gain
- Offset
- Bias
- Data type (Reset, end-of integration (EOI))
- Number of images summed on board, when applicable, i.e. when $T_{\mathrm{int}} = 0.28\,\mathrm{s}$ has been selected
- Sampling rate (only for sampling image mode)

Please consult Chapter 8 for some more details.

CISP – ISOCAM Standard Processed Data

The CISP is essentially a slightly processed and more compact version of the CIER. In the pipeline, the CISP file is the output of Derive SPD and the input of Automatic Analysis System. A thorough description of CISP's contents, as well as IDL based "recipes" to read and manipulate CISP and other ISOCAM products can be found in Chapter 8.

The CISP contains, besides most of the information separately given in the compact status file CSTA:

- The most important housekeeping parameters such as ISOCAM temperature, blackbody current & voltage, etc.
- The instrument time key (a unique time identifier for each frame taken by ISOCAM), ON Target Flag, raster point ID, Quick Look Analysis flag and Compact Status History (CSH) discrepancy flag (for commanding verification). The CSH discrepancy flag indicates records where the wheels did not reach their commanded position yet. Users are advised to exclude data with a bad CSH (= 0) discrepancy flag.

- Detector data as end-of-integration (EOI)−RESET, including the mean, median, RMS of the whole array per exposure.
- Reset frames for each exposure.
- For SW detector data, the blind reference lines (column 33, 34) of the SW detector, which have been subtracted to end-of-integration (EOI)−Reset to correct for electronic perturbations.
- Header keywords.

Equivalent products to CIER and CISP but when ISOCAM is in parallel mode are:

CPER	ISOCAM Parallel ERD
CPSP	ISOCAM Parallel SPD

Chapter 4

ISOCAM Calibration libraries

4.1 Philosophy

The limited life time of ISO has reduced the amount of time for possible calibration measurements. (For a detailed outline on how the calibration time was used please inspect "CAM Calibration Explanatory document [3]". In principle the tremendous number of different optical ISOCAM configurations makes it impossible to schedule all calibration observations for each orbit.

Therefore, results of calibration observations are archived into calibration libraries. These libraries are used by the automatic analysis chain (see Chapter 5) and can be browsed through by the observer to search for darks, flat–fields, point spread functions that best match his observation. It should be noted here that unless specific measurements were scheduled by the observer, calibration will be performed on a best-match basis. We recommend that observers start their own analysis with the most recent calibration files available at `http://www.iso.vilspa.esa.es`. The version of the calibration files used are referenced in the CUFF.

In total there exist 16 individual calibration libraries for each detector. Those files will be described in this chapter.

4.2 Calibration libraries and how they are produced

In the following subsections we describe the calibration files, their content and how they have been derived. The names of the libraries start with "CCG*", where "*" refer to the two possibilities SW or LW detector.

4.2.1 List of Data Files

The calibration files can be sorted in 11 "basic" and 5 "higher level" libraries. In the minimum calibration system only the basic calibration libraries are implemented while the remaining files are currently empty. This set of 15 calibration libraries is also known as the Cal-G files.

“Basic” calibration files

CCG*DEAD	ISOCAM Dead Pixel Map
CCG*DARK	ISOCAM Dark Current Exposures
CCG*DFLT	ISOCAM Detector Flat Field Library
CCG*OFLT	ISOCAM Optical Flat Field Library
CCG*SPEC	ISOCAM Filter & CVF spectral data
CSWCVF	ISOCAM SW CVF description
CLWCVF1	ISOCAM LW CVF1 description
CLWCVF2	ISOCAM LW CVF2 description
CCG*SLP	ISOCAM CVF spectral line profile
CCG*PSF	ISOCAM Point-Spread-Function Library

(*=SW, LW)

“Higher level” calibration libraries

These are at present empty.

CCGLWDMOD	ISOCAM LW parameters for the time dependent dark corrections
CCG*TRANS	ISOCAM Model transients
CCG*LINEAR	ISOCAM Linearity Correction Library
CCG*FRAME	ISOCAM Detector astrometric calibration
CCG*GLITCH	ISOCAM Glitch Model
CCG*STRAY	ISOCAM Non-dark local light model

(*=SW, LW)

4.2.2 CCG*DEAD – ISOCAM * Dead Pixel Map

Contains an image with dead pixels. One example is all pixels of the dead column of the lw detector.

4.2.3 CCG*DARK – ISOCAM * Dark Current Exposures and CCGLWDMOD

Contains the library of Dark Current Exposures. Individual dark exposures of a specific integration time have been deglitched. The units of the CAL-G library darks are ADU/s. In order to scale this to ADU/s/Gain, it is necessary to inspect the GAIN column of the FITS binary table. Note that the definition of gain in this file is as binary and can take the values (0, 1, 2) rather than the real gain values of (1, 2, 4). The observer must divide by the real gain value, i.e. 2^{gain}.

For $T_{\rm int} = 0.28$ s, four images are added on-board, still the library dark is correctly scaled back to ADU/s. Only the frames which are taken on a stabilized detector are averaged. The result is converted to ADU/s. The dark should be subtracted from all ISOCAM images.

Dark exposures are taken once every orbit with an RMS accuracy of about 0.5 ADU (for SW detector) and 0.3 ADU (for LW detector).

The calibration data structures are ordered by increasing integration times, followed by gain and offsets. The dark depend physically on orbital position of the space-craft and possibly also the temperature. For a description of those physical effects and a method on how to best correct for them see: “ISOCAM Dark Current Calibration Report [6]”. The components of the polynomial fits for dark drifts of the LW detector are stored in the LWDMOD calibration file.

4.2.4 CCG*DFLT – ISOCAM * Detector Flat Field Library

To reduce the number of calibration measurements to be performed, we assume that the flat–field is the result of the detector flat-field times the optical path flat–field. Detector flat–fields and optical flat–fields are measured separately by illuminating the detector with an external source (the zodiacal light). For those optical configurations where the zodiacal background has insufficient brightness we obtained flats using the internal calibration device.

The detector flat–field is measured through the 1.5 arc-seconds lens. The smallest lens will minimize effects such as stray-light due to the optical path and is taken to represent best the detector flat–fields.

The exposures are deglitched, dark corrected and scaled to a gain of one. Only those frames which are taken on a stabilized detector are averaged. The result is converted to ADU/s/Gain. The corresponding dark is subtracted. The image is scaled to a median of one using only the central part of the detector array, more precisely from column 10 to 21 and row 10 to 21. This generates the detector flat–field ('DFLT').

It can be assumed that standard flat–fields have accuracies of better than 1%, see ISOCAM Calibration Error Budget Report [9].

The Automatic analysis searches the library for a detector flat that best match the observation in filter, CVF position, selection wheel and integration time. The detector flat physically depends on the entrance wheel position which gives quite an important limitation to the usefulness of the calibration flats. For more details on detector flat–fields see the "ISOCAM Flat–Field Calibration Report[7]".

4.2.5 CCG*OFLT – ISOCAM * Optical Flat–Field Library

The optical flat–field is measured by illuminating the detector through different lenses. Same data processing steps are applied as for the detector flats, see above. Finally, the optical flat–field ('OFLT') is computed by dividing by the corresponding detector flat–field. We search for the corresponding optical flat to find a best match in p.f.o.v., filter, CVF position and selection wheel. As for the detector flat the lens wheel jitter gives an important limitation to the accuracy of the library flats. No optical flat–field will be measured through the CVF filters. For more details on optical flat–fields see the ISOCAM Flat–Field Calibration Report [7].

4.2.6 CCG*SPEC – ISOCAM Filter & CVF spectral data

Filter and CVF spectral characteristics are derived by observing standard calibration stars. A detailed report on ISOCAM photometry including a description of the photometric program followed by the ISOCAM team is given by Blommaert (1998) [10].

The resulting photometric relation between detector units (ADU/s/Gain) and astronomical units (Jansky) are stored in the CCG*SPEC library by the so called SENSITIV parameter.

The dark corrected and de-glitched exposures of measurements towards a calibration standard star are averaged (by taking a pixel mask into account). The resulting image is flat–fielded and background subtracted. The ADU/sec of the source are derived in a circular aperture and a correction factor for the point spread function is applied. This result is then divided by the known Jansky's of the measured source assuming a λ^{-1} spectral shape (see "ISOCAM Photometry Report" [10]) . The so derived conversion factor (ADU/s/Gain) / Jy is written into the SENSITIV parameter.

In the case of CVF measurements of a continuum source: For each CVF position point source photometry is done. The resulting spectrum is divided by the known spectrum of the standard. This will give the spectral response function. The derived conversion value (ADU/s/Gain)/Jy at each CVF position is written into the SENSITIV parameter. For more details on the CVF spectral response function see ISOCAM CVF Calibration Report [8].

For all filters and all CVF steps: The spectral transmission curve $R(\lambda)$ (based on ground based measurements) is stored. The spectral transmission curve (cmp. Chapter 6) as defined by ISOCAM is the product of filter transmission $T(\lambda)$ times the detector quantum efficiency $Q(\lambda)$. Consequently if one wants to compare the observered flux with a model spectrum one must apply Eq. (6.1).

4.2.7 CCGSWCVF – ISOCAM SW CVF description

This product relates the CVF wheel position with lambda in microns. The spectral response function is also written in this file. It is stored as function of each CVF wheel position in the SENSITIV parameter. The result of the ground based wavelength calibration is published in Table 12, 13 and 14 of the ISOCAM Observer's Manual [24] (p. 81).

4.2.8 CCGLWCVF1 – ISOCAM LW CVF1 description

This product relates the CVF wheel position with lambda in microns. The spectral response function is also written in this file. It is stored as function of each CVF wheel position in the SENSITIV parameter. The result of the ground based wavelength calibration is published in Table 15 and 16 of the ISOCAM Observer's Manual [24] (p. 84). For more details on the CVF spectral response function see ISOCAM CVF Calibration Report [8].

4.2.9 CCGLWCVF2 – ISOCAM LW CVF2 description

This product relates the CVF wheel position with lambda in microns. The spectral response function is also written in this file. It is stored as function of each CVF wheel position in the SENSITIV parameter. The result of the ground based wavelength calibration is published in Table 17 and 18 of the ISOCAM Observer's Manual [24] (p. 86). For more details on the CVF spectral response function see ISOCAM CVF Calibration Report [8].

4.2.10 CCG*SLP – ISOCAM CVF spectral line profile

In order to verify and compare the line profile with ground based measurements, narrow lines are scanned in nine CVF positions. For each CVF position point source photometry is done. The resulting spectrum is divided by the Spectral Response Function. The scan will be then divided by the height of the measured peak and is then stored as function of the filter wheel position (FCVF) in the FITS parameter NORMFLUX. This calibration file, which is still empty at present, may also contain information on the observed source and line.

4.2.11 CCG*PSF – ISOCAM * Point-Spread-Function Library

This library contains a set of normalized point source images or Point Spread Function (PSF) placed on different locations of the detector array. Each image has been processed in the standard way: deglitched, dark current subtracted and flat–field corrected.

These point source observations are performed by a 4×4 micro–scan raster with a step size of 3 arcsec for 12 arcsec pfov, and by a 6×6 micro–scan raster with step size of 2 arcsec for the other pfovs in order to obtain a better spatial sampling rate. However this does not allow to achieve a good enough theoretical sampling rate with respect to the Point Spread Function size for every configurations. Therefore the users should be warned when using a Point Spread Function for deconvolution purposes. In the PSF library, the estimated location of each PSF is found under CRPIX1 and CRPIX2 parameter in pixel units.

The micro–scan raster used for the PSF measurements also allows to obtain information about photometric variations as a function of the position of the point source on a pixel. Except for 12 arcsec pfov,

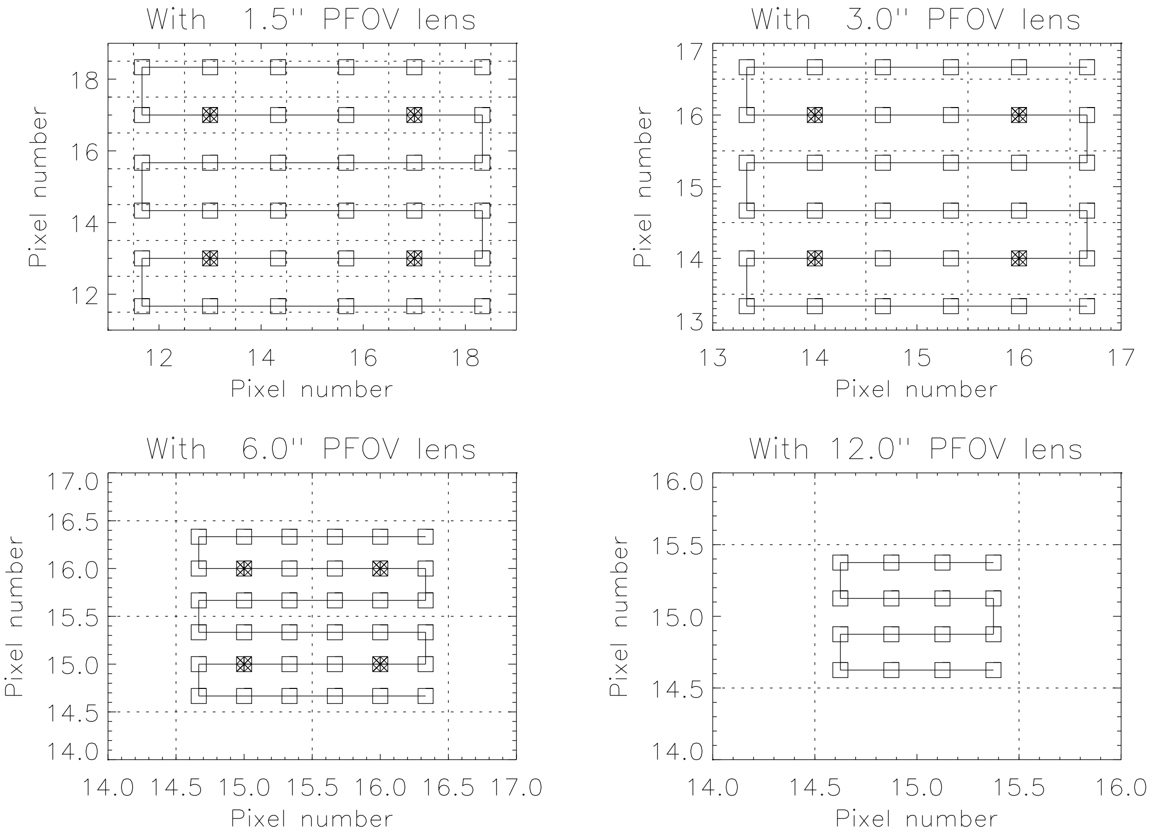

Figure 4.1: Point source displacement along a raster

one raster gives 9 sets of 4 raster points with the same sub-pixel positions (see Figure 4.1). An example of such 4 raster points is shown in Figure 4.1 as 4 black raster points for 1.5, 3, and 6 arcsec pfov. One Point Spread Function is created by averaging these four point source images giving a adequate offset to make them overlapped. This gives a set of 9 PSFs representative of the region defined by the raster extension and free of the absence of data on the column 24. Therefore, the Point Spread Function library contains, for one configuration, a set of 9 PSFs for different locations on the detector array. Figure 4.2 and 4.3 show examples of the micro–scan performed for 3 and 6 pfov PSF measurements. All the rasters for the same configuration are done within one observation time block, pointing on the same star.

All the PSFs of a given configuration are normalized by the flux of the most pixel-centered PSF in the raster which defines the central region of the detector. This global normalisation is supposed to give photometric corrections as a function of the point source position on the pixel and on the detector plane. Therefore, for each configuration, only one PSF is normalised to 1. The normalisation factor is the flux of the reference PSF integrated excluding the edge pixels. This is because of the uncertainty on the field mirror edge position for 6 arcsec pfov. The Point Spread Function library can then be used for the point source photometry which attempts to achieve an accuracy beyond inside-pixel response variations.

However, this photometric correction normalisation is not possible using Point Spread Function library before version 3.5, while the global normalisation of the whole set of PSFs is alterated by the non-corrected transient effect which introduces the flux variation of more than 5% and smears out the variation due to the in-pixel and on-detector response variation.

Finally we give one example of a comparison between the measured and a theoretical Point Spread Function but refer for a discussion to the ISOCAM PSF Report [30].

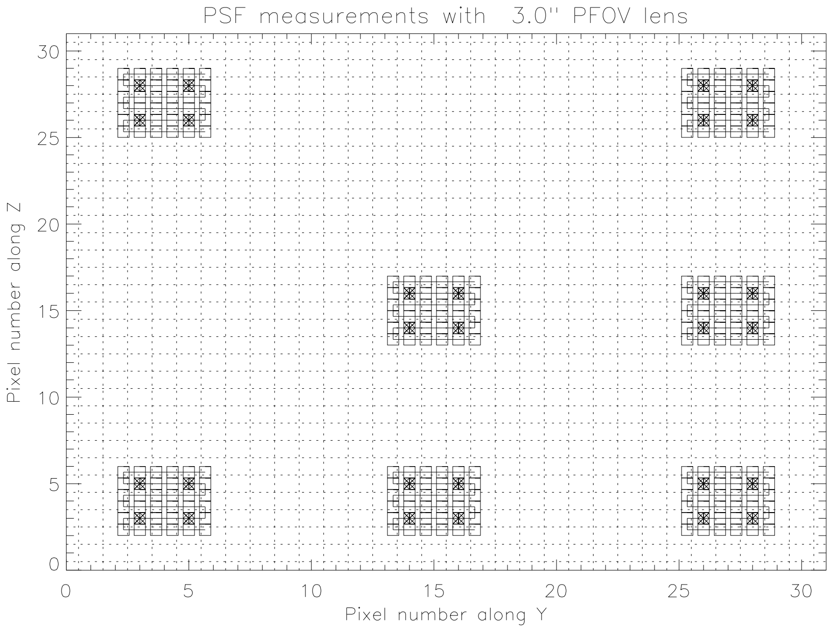

Figure 4.2: PSF measurements on different positions on the detector

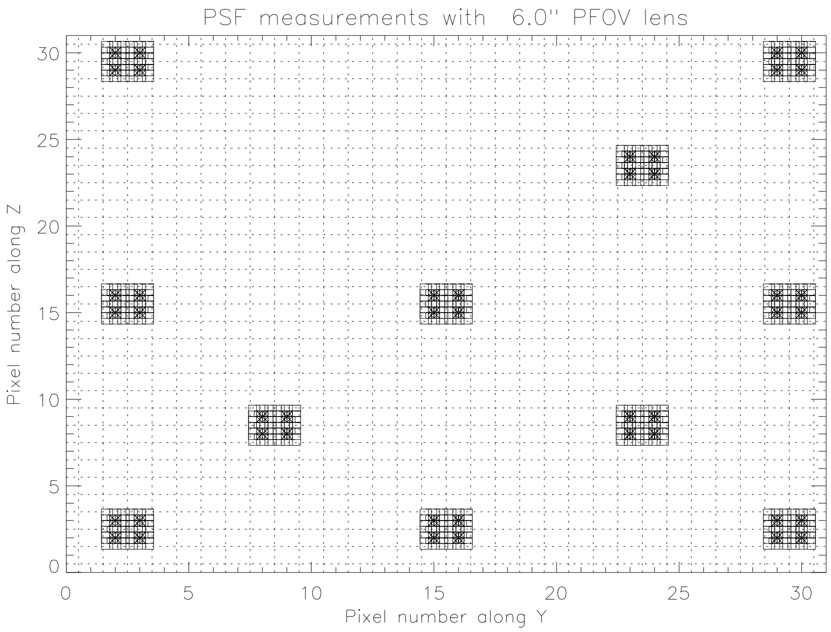

Figure 4.3: PSF measurements on different positions on the detector

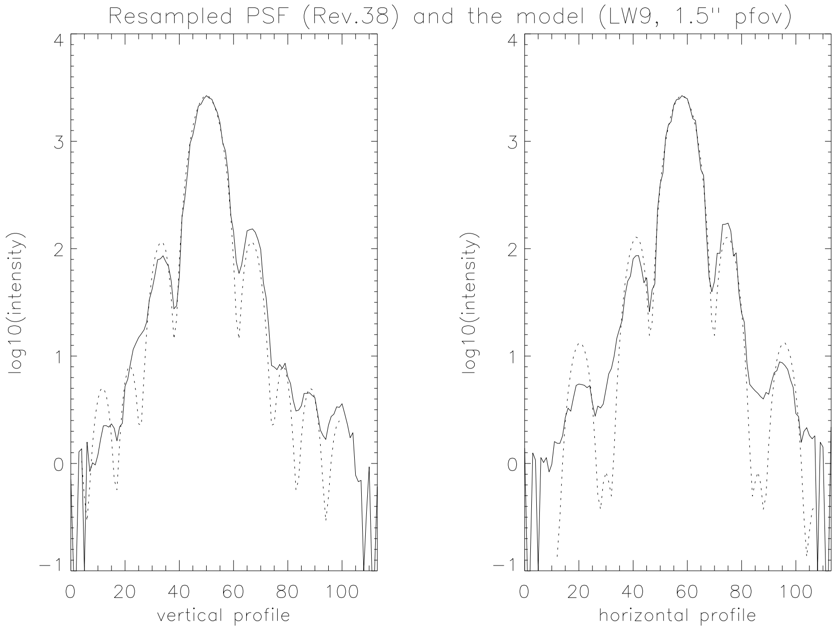

Figure 4.4: Comparison of measured PSF (continuous line) and modeled PSF (dashed line) with LW9 and 1.5 arcsec pfov. *From "THE ISOCAM PSF REPORT"* [30]

4.2.12 CCG*TRANS – ISOCAM * Model transients

This product should contain a model description for the detector memory effects.

4.2.13 CCG*LINEAR – ISOCAM * Linearity Correction Library

This product contains a scaling factor per pixel per ADU/s or may contain for each pixel polynomial coefficients. This kind of scaling factor should be used to correct for detector non-linearities.

4.2.14 CCG*FRAME – ISOCAM * Detector astrometric calibration

This product contains a transformation correcting for field distortion ("pin-cushion") of the images caused by any imperfections in the optics. Note that this file does not give you the best possible correction. Please inspect the report on "Field distortion for LW [5]" on more details.

4.2.15 CCG*GLITCH – ISOCAM * Glitch Model

This product should contain a model for detector response and decay after ionizing radiation and particle impacts.

4.2.16 CCG*STRAY – ISOCAM * Non-dark local light model

This product should contain a model for stray-light affecting the images.

Chapter 5

Auto-Analysis

5.1 Auto-Analysis Processing Overview

The Auto-Analysis is the final stage in the pipeline processing, where astronomically meaningful products are derived. For an overview of the pipeline, see Section 3.

The input to the Auto-Analysis is the following:

- Standard Processed Data (SPD)
- Instrument pointing history (IIPH & IRPH)
- Executed Observation History (EOHA)
- CAL-G calibration libraries

For any given instrument configuration an attempt is made to find the best possible calibration datasets (and of course, from the most up-to-date libraries). If it is not possible to match the desired configuration exactly, then a 'closest fit' is selected and the choice logged in the CUFF product (described below).

The processing involved can be summarised in the following manner:

1. The EOHA is read to identify the type of the observation.
2. The SPD file is read and each ISOCAM state is stored in a Science CAM Dataset (SCD) within memory. Pointing and jitter information is extracted from the IIPH and IRPH files. Dead pixels are identified by inspection of the relevant CAL-G files.
3. Average exposures are calculated in detector coordinates and written to the CCIM product. Note that *no* calibrations or corrections have been applied to these images. CMAP and CMOS are the places for calibrated images and will be produced later.
4. Deglitching is performed by median filtering of 20 exposures. The glitch list product CGLL is written.
5. Each SCD is checked to find at what point the detector is assumed to be stabilized. This is done by inspection of the Quick Look Analysis (Quick Look Analysis) flag. All frames with Quick Look Analysis flag set to 'ON' are considered to be taken on a stabilised detector.[1]

[1] The Quick Look Analysis flag is switched to ON, after the value of Nstab (= Number of stabilisation exposures before a single pointing observation or before the first point of the raster; your PGA input parameter) frames are taken.

6. Each SCD is checked to find readouts where the wheels have not reached their commanded position yet. This is done by inspecting the Compact Status History (CSH) discrepancy flag. All frames with CSH discrepancy flag equal to 0 are considered to be taken with a good instrument configuration

7. Each SCD is checked to find readouts where ISO has not reached its commanded position yet. This is done by inspecting the On Target Flag (OTF) discrepancy flag. All frames with OTF equal to 7 are considered to be taken with a stable pointing

8. *Null Step*: No jitter correction is applied as the amplitude has been found to be so small that it is dominated by the measurement noise. Neither is there any astrometric correction owing to wheel positioning accuracy, bending of the optical axis as a function of the filter wheel or optical distortion away from the optical centre.

9. *Null Step*: There is currently no correction for memory (transient) effects or non-linearity of the detectors.

10. Any parts of the detector not illuminated by sky are identified and masked out. This can occur with certain PFOV/mirror combinations (12" PFOV, 3" PFOV with small mirror).

11. The jitter, memory and stabilisation information is written to the CJAM product.

12. Each SCD is dark-subtracted, flat-fielded and converted to flux (Jy/pixel) using the appropriate data from the calibration libraries. Non-stabilised frames, frames with a bad pointing or instrument configuration, detected glitches and dead pixels are discarded. These calibrated images are written to the CMAP product.

13. The calibrated images for raster, micro-scan and beam-switching AOT are combined into the mosaic product CMOS. In the beam-switching case[2] the background is first subtracted from the source images.

14. The CMAP images are searched for point sources, and each detection is logged in the product CPSL. Note that the CMOS images are *not* searched for point sources.

15. If point sources have been detected and measured in more than one wavelength, then the resulting spectra are written to the CSSP product.

16. A processing summary including any warning or error messages is written to the CUFF product.

5.2 Auto-analysis data products

Below you will find a description of the Automatic Analysis products, together with a brief explanation of the processing steps that go into their creation.

There are a total of eight products that may be produced by the Auto-Analysis, two of which (CMOS and CSSP) are not present for every observation. The end products of the pipeline can be considered to be the calibrated image files CMAP and, if present, CMOS, though actually the point-source list CPSL, spectra-list CSSP and the information file CUFF are produced afterwards.

Note that in general, it is not recommended to attempt to make improvements on the analysis starting from the Automatic Analysis products. We strongly suggest to start re-analysing the data from the SPD or ERD level (products CISP and CIER respectively). A brief introduction to the interactive analysis of ISOCAM data is given in Section 8.

[2] Auto Analysis will try to determine if the observer specified source position as reference and vice-versa. In such a case the CUFF will contain the message "Reversing source and reference positions". Should this reversal not be detected by Auto Analysis, then the CMOS image of the object will be negative.

5.2.1 CCIM – ISOCAM images in detector coordinates

While the ERD and SPD products are organised on a frame-by-frame level, the fundamental unit of the CCIM (Cam Calibration Image, pronounced 'chee-chim') is that of the ISOCAM STATE (see Section 3.2.1). For each state, mean and error images are calculated and converted to ADU/s by dividing by the fundamental integration, the read-out time. These are the *only* operations performed on the CCIM images. It should be understood that these images are purely for calibration and informational purposes and have little astronomical meaning.

The images come in sets of three, one set for each state, which are stored in successive rows of the binary table. In each of these sets are found the mean image (in ADU/s and here called the 'rate' image), the statistical error or 'error rate' image (standard deviation per pixel in ADU/s, to be precise), and the exposure image in seconds (comprising the total exposure time per pixel). As no data has been removed the exposure image is always flat.

Every state of the AOT is present. Thus if a dark-current or flat-field measurement was performed in the AOT then the resultant images will appear in the the CCIM (but not the CMAP). The observation mode appears in the MODE field of the product. This may be any of `OBS` = Observation, `DRK` = Dark, `FLT` = Flat–field, `CLN` = Clean, `IDL` = Idle.

The celestial coordinates RA and DEC are stored in fields of the table and refer to the centre of the detector (these are actual rather than intended positions). The time and date are specified for the start and end of the observation in the (primary) header and for each state in the table.

5.2.2 CGLL – ISOCAM Glitch List

The CGLL product lists all detected glitches. Glitches are time-tagged to UTK. They are indexed to the related CCIM image and their location specified in both detector and celestial coordinates. A median filter of width 20 is applied to the raw data for each pixel over the entire observation, ignoring any dead pixels. The values thus filtered out are listed as glitches.

5.2.3 CJAM – ISOCAM Jitter, Memory and Stabilization information

The CJAM product logs for each frame the calculated jitter (offset) from the nominal pointing, whether the frame contains glitches, whether the detector was stabilised, and whether there were memory effects. In addition it is time-tagged and indexed to CCIM.

The jitter is calculated by comparing the instantaneous attitude solution with the nominal pointing position. Note that the jitter is still calculated even though it is not taken into account in creating the images. Glitches are those determined for CGLL. Stability is currently determined by inspection of the Quick Look Analysis flag. No algorithm is currently implemented for detection and correction of memory effects.

5.2.4 CMAP – ISOCAM Celestial IR image

The CMAP product consists of calibrated flux images aligned with celestial coordinates[3] in units of Jy/pixel. CMAP images are produced only for observation ('OBS') mode, and when the spacecraft is on target and the detector is stabilised.

As with the CCIM, the images come in sets of three. They can be thought of as calibrated versions of the 'good' CCIM images, and indeed, they contain an index pointing to the associated CCIM images[4]. The first of the three gives the flux (Jy/pixel), the second the flux error (Jy/pixel) and the third again

[3] Strictly speaking, the images are in detector coordinates with the FITS keywords specifying the sky orientation.

[4] Indices also exist pointing to dark and flat measurements within the AOT that were used for the calibration. However, this facility is not implemented.

the exposure time (seconds). Here the exposure images are not flat because not all pixels are counted for each frame (i.e. glitched and dead pixels are not included in the averaging).

To calculate the images, the SCD's (not the CCIM images) are averaged again, this time discarding pixels that are dead, glitched, not stabilised, or not illuminated by the sky or not taken with the commanded target position. The 'best' available dark current measurements and flat-fields are extracted or computed from CCGLWDMOD or CCGSWDARK, CCG*DFLAT and CCG*OFLAT CAL-G libraries, and the configurations of these logged in the CUFF. Using these, the averaged images are corrected for dark current and flat-fielded. The calculated error is a combination of the image (assumed Gaussian) noise and the errors on the darks and flat-fields. The factor to convert to flux in units (ADU/s/Jy/pixel) is then extracted from the spectral characteristics CCG*SPEC file (the SENSITIV parameter) and used to convert the units to Jy/pixel.

For each image, the sky projection is specified by the FITS keywords (actually as fields in the table) `CTYPEi`, `CRVALi`, `CRPIXi`, `CDELTi`, `CROTA2` and (redundantly) `CDiiijjj`. The time and date are specified for the start and end of the observation in the (primary) header and for each state in the table.

5.2.5 CMOS – ISOCAM Celestial IR mosaic

A CMOS image is formed by combining the CMAP images into a single "mosaic" image for each instrument configuration. The product is only produced for raster, micro-scan and beam-switching AOT's. The format of this product is the same as for CMAP except that there is no indexing to CCIM images; instead indices refer to the CMAP product.

Rasters and micro-scans receive exactly the same treatment. For the beam switching AOT, the mosaic is built from a series of "difference" images, which are calculated by subtracting the background (mean of the reference positions) from the source image. In the polarisation case separate images are produced for each entrance wheel position. The number of pixels in the mosaic is calculated automatically. It is built by assigning the CMAP pixels to the nearest CMOS pixels and then taking the weighted mean across the image.

5.2.6 CPSL – ISOCAM Point-Source List

The CPSL product contains a list of all the point sources detected in the CMAP images. Each detection is indexed to the appropriate image in CMAP. The detected point-sources are listed in the CUFF.

Point source searching is performed on the CMAP product by calculating point-source likelihoods throughout the flux maps, using a central point spread function extracted from the CAL-G file CCG*PSF, and best matching the instrument configuration.

A (maximum–)likelihood estimate is made of the flux from a single point source of fixed position above a flat background under the assumption that the flux map is accompanied by Gaussian errors.

The positions of the detected sources are then located more accurately by fitting point spread functions with different fractional pixel offsets. Finally the flux is recalculated using the point spread function that provided the best fit.

5.2.7 CSSP – ISOCAM Source Spectrum List

CSSP contains the measured spectrum for each point source detection. The file is only produced if measurements have been made in more than one wavelength and at least one point source has been detected at the same position at two wavelengths (e.g. fixed filter observations) or where AAC has found a point-source at the same position for four or more contiguous CVF steps (e.g. CVF observations). An index exists to the relevant entries in the CPSL.

The spectra consist of the measured fluxes at each wavelength. If there is more than one measurement for a wavelength then a weighted mean is calculated for that wavelength.

5.2.8 CUFF – ISOCAM User-Friendly log File

The CUFF file contains a log of messages generated by the Automatic Analysis system. While by necessity a FITS file (since it is a product), the contents of the table are essentially an ASCII file. An example CUFF table is shown in Appendix A.

5.2.9 Changes since OLP 4.0

Since the pipeline was scientifically validated, the following major improvements were implemented:

- Improvements of (most) calibration files.
- The area of the detector not illuminated by the sky is masked out.
- The mosaic construction algorithm has been improved.
- The dark correction is now a function of time since activation and revolution number (cf "ISOCAM Dark Current Calibration Report [6]".
- The sub-pixel PSF fitting improves the photometric accuracy.

Chapter 6

Photometry with ISOCAM

6.1 Standard star photometry

The absolute flux calibration of the ISOCAM instrument is based on the observations of calibration stars selected from the Ground Based Preparatory Programme (GBPP, P. Hammersley, priv. communication). The visible magnitudes and near-infrared photometry observed in the GBPP have been fitted with a Kurucz model to provide the flux densities at longer wavelengths. The overall spectral energy distribution is believed to be correct within 5% .

The conversion from the measured signal in ADU/Gain/s to the flux density in Jy is given in the so-called SENSITIV parameter which can be found in the CCG*SPEC (see Section 4.2.6). The conversion is given for the reference wavelength of each filter, assuming a $\lambda F(\lambda) = const$ spectrum. In Table 6.1 and 6.2 we give values of the SENSITIV parameter for the LW and SW detector, respectively. A comparison of these values as derived from in-orbit with those given from pre-launch expectations are shown in Fig. 6.1 and 6.2

For a detailed review of the status of the ISOCAM photometry please consult the "ISOCAM Photometry Report [10]".

6.2 Colour Correction of Photometric Measurements

The transmission of optics and filters, as well as the quantum efficiencies of the detector exhibit non-flat spectral dependencies. This has the consequence that two sources radiating the same power but within a given wavelength range with different spectral shape, produce two different signals. The flux density from measurements of a source with a given filter has to be corrected for this effect. Therefore the flux density at the reference wavelength of a certain filter of the heterochromatic photometry of ISOCAM has been scaled to an a priori known spectral shape, a $F(\lambda) \sim \lambda^{-1}$ law. When the observed source has an energy-distribution different from this law a correction has to be applied.

In the CCG*WSPEC cal-G files the spectral transmission curve for each filter is given. The spectral transmission $R(\lambda)$ as defined for ISOCAM is the product of the filter transmission $T(\lambda)$ and the detector quantum efficiency $Q(\lambda)$):

$$R(\lambda) = T(\lambda) \times Q(\lambda)$$

At this stage, no transmission curve for the lenses has been included. To determine the *actual* flux density one has to divide the flux density (derived after dividing the measured ADU/Gain/s by the SENSITIV parameter) by the colour correction factor.

Table 6.1: LW sensitiv values

Filter	λ_{ref} μm	Mean Adu/G/sec/mJy	RMS %
LW1	4.5	0.77	4.1
LW2	6.7	2.32	3.3
LW3	14.3	1.96	4.8
LW4	6.0	0.78	7.5
LW5	6.8	0.35	7.5
LW6	7.7	1.03	6.4
LW7	9.6	1.33	6.7
LW8	11.3	0.78	5.4
LW9	14.9	0.65	2.8
LW10	12.0	4.23	3.9

Table 6.2: SW sensitiv values

Filter	λ_{ref} μm	Mean Adu/G/sec/mJy	RMS (%)	In-orbit/pre-launch
SW1	3.6	0.41	4.6	1.41
SW2	3.3	0.12	4.9	1.56
SW3	4.5	0.26	3.4	1.11
SW4	2.8	0.30	2.9	1.97
SW5	4.0	0.77	4.8	1.39
SW6	3.7	0.19	6.0	1.31
SW7	3.0	0.16	4.5	1.76
SW8	4.05	0.038	7.2	1.29
SW9	3.9	0.074	5.8	1.31
SW10	4.6	0.092	2.5	1.08
SW11	4.26	0.058	5.5	1.17

$$F_{actual}(\lambda_{ref}) = F_{derived}/K(\lambda_{ref})$$

where $K(\lambda_{ref})$ is given by:

$$K(\lambda_{ref}) = \left(\int F(\lambda)/F(\lambda_{ref}) \times \lambda/\lambda_{ref} \times R(\lambda)d\lambda\right) / \left(\int R(\lambda)d\lambda\right) \tag{6.1}$$

Formally the reference wavelengths are arbitrary and they are chosen such that they clearly identify a particular filter and that they are easy to remember. Some reference wavelengths (for the filters: SW1, SW2, LW6, LW8 and LW10) are chosen for comparison with the ISOPHT filters. The LW10 filter gives a direct comparison with the 12 μm filter. For a further discussion see "Reference Wavelengths for ISO: CAM and PHOT filters [28]". The reference wavelengths can be found in Table 6.1 and 6.2.

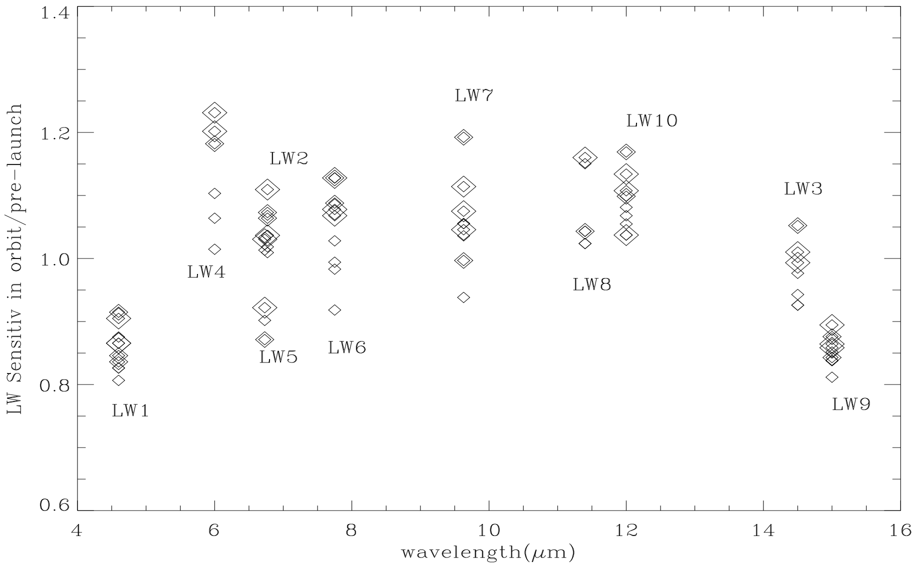

Figure 6.1: The in-orbit LW SENSITIV values compared to the pre-launch values. 1.5, 3 and 6″ PFOV measurements are included. The smallest symbols are 1.5″ and the largest are 6″ measurements. Taken from the *"ISOCAM PHOTOMETRY REPORT" [10]*

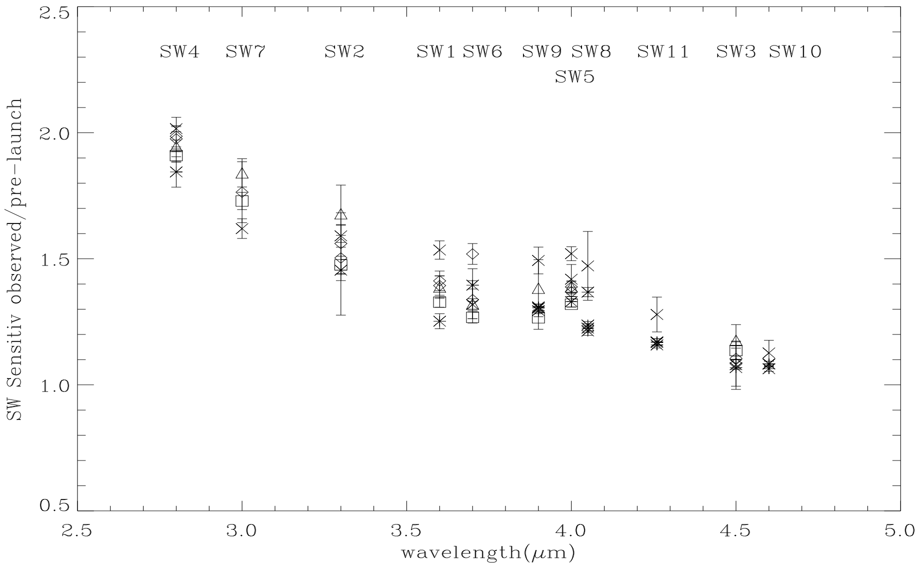

Figure 6.2: The in-orbit SW SENSITIV values compared to the pre-launch values. 1.5, 3 and 6″ PFOV measurements are included. Taken from the *"ISOCAM PHOTOMETRY REPORT" [10]*

Chapter 7

Instrument Model

In the following we give a very brief description on how one may proceed to start re-analyzing ISOCAM data. In general it is not recommended to try and improve on the Automatic Analysis products. Instead we strongly suggest to start re-analyzing the data from the derive SPD (product CISP) or ERD level (product CIER). This chapter describes a basic model of the instrument, gives a guide on how to analyze data in such a simplified context and indicates deviations from this idealized situation. A summary of principle uncertainties in ISOCAM measurements in style of a global error budget is given.

7.1 Basic instrument model

A basic model of ISOCAM can be described by making the following assumptions:

- We are working with a linear detector and linear Analog to Digital Converter.
- The observer has taken a sufficiently high number of exposures, so that the detector reaches stabilization. We neglect memory effects.
- After cosmic ray hits and even after a heavy ion impacts, there is no variation of the pixel response or gain.
- The optic is perfect, so that the Point Spread Function is taken to be identical all over the field of view
- There is no stray-light or any pin–cushion distortion.
- The CVF central wavelength is identical on the entire field of view.
- We neglect any jitter of the satellite.

By taking the basic instrument model it is obvious that we oversimplified the true situation. However, it allows to outline the basic processing steps without having to go into details of describing models for flux transients, glitch histories or non-linear detector behaviors. For a validation of this basic concept see Section 7.3. Some more information on the importance of those individual effects and how they may add up is described in the global error budget of the instrument (Section 7.4). Processing algorithms to correct for such effects are given in Chapter 9.

7.2 Basic processing steps

We compile a list of basic rules to be considered by the observer in order to enhance the success rate of his own data processing. The validity of these guidelines is limited to the basic instrument model.

We recommend to use, as input to the data reduction, the CISP. After having extracted the relevant data frames (see Chapter 8 for basic tools), the first processing step should be to flag invalid exposures, e.g. where the pointing deviates from the nominal position or the instrument was not in the optical configuration of the science observations. At the same step bad pixels (e.g. column 24 of the LW detector) can be best flagged via a mask. Then cosmic ray impacts, so called glitches should be removed. A standard median filtering technique on individual frames as a function of time is probably sufficient as a first step for de-glitching.

One should scale frames to a gain of one and remember to do so when necessary for products taken from the calibration libraries (e.g. dark currents).

The pixel values should be converted from ADU into ADU/s by dividing by the fundamental integration time.

One should apply an algorithm to select only those frames which have good pointing flag, are in the selected instrument configuration of the science observation and frames taken on a stabilized array. Only those frames should be co-added and averaged.

The calibration libraries have to be accessed. We recommend to consult the CUFF product (see Appendix A) in order to clarify which calibration files were used by the Automatic Analysis system. With some experience one can try to find a better matching entry in the libraries CCG*OFLT and CCG*PSF.

Then more or less standard processing steps as known from ground based cameras can be applied. Those are:

- Subtract the dark from the averaged frame.
- Apply flat field corrections.
- Select and subtract the background.

After visualization of the resulting frame one may wish to apply the point source photometry. This can be done in different ways and one may use standard packages of astronomical data reduction systems. Alternatively, one example is to scale the point source to have roughly the volume of one. Make a least square search of the Point Spread Function library in order to find the best match Point Spread Function. Alternatively use the Point Spread Function as in the Automatic Analysis system. Scale best Point Spread Function volume to approximately the source volume. Then increment the scaling factor to minimize least–square difference between scaled Point Spread Function and point source. The optimum scaling factor gives the ADU/s in the point source. Inspect CCG*SPEC to get the scaling factor Jy/(ADU/s) or the spectral response function for CVF scan.

We repeat that the data processing steps described above are only valid working under the assumption of a basic instrument model. It has always taken into considerations that reality is more complicated and far from the idealized situation outlined here. We also strongly encourage observers to use CIA, the ISOCAM interactive analysis package, where all these steps, and more, are implemented in a user-friendly way.

7.3 Deviations from the basic instrument model

Here we summarize known deviations from the basic instrument model. This knowledge is derived from laboratory studies as well as in-orbit experience of the behavior of the ISOCAM instrument.

Detector

For a review of the ISOCAM detector see: "The ISOCAM Detectors: An Overview" [11].

The LW detector has problems to stabilize in time: Even for a simple case, e.g. decreasing transient after a flat illumination, at least 20 images have to be taken before stabilization of the LW detector. In more general circumstances, model results on downward transients suggest a number of about 50 exposures before the LW detector reached a kind of stabilization (see: "Transient Behaviour of LW Channel of ISOCAM [1]" and "Comparison of ISOPHOT and ISOCAM/IAS approaches for transient correction [2]". We recommend to study drift effects of the detectors.

For a description of ISOCAM detector trends see: "Report on Trend Analysis of CAM" [20]. In short we found that: Along one orbit and throughout the mission the dark level changes (see: "ISOCAM Dark Current Calibration Report [6]", while the detector responsivity is stable in time ("ISOCAM Photometry Report [10]".

Optics

There are variations of the Point Spread Function when compared on both axis (see: "ISOCAM PSF Report [30]"; "Field distortion for LW [5]") There is a field distortion and vignetting problem in particular for the 6 and 12 arc-seconds lens. The wavelength calibration of the CVF shows a gradient along the array. For the 6 arc-seconds lens the gradient is about half a CVF resolution element ($\delta\lambda/2$) from the center to the edge of the detector array.

In the CVF ghosts are present for an example see Figure 7.1. Please consult "Ghosts in ISOCAM Images" [31] for more details.

Glitches

Heavy ion or electron impacts, so called glitches, usually induce response or gain variations of several pixels. This is an important limitation to ISOCAM images for detailed description see: "Glitch Effects in ISOCAM Detectors" [14]; "Cosmic Ray effects on the ISOCAM LW detector" [19] and "ISOCAM Glitch Library"[15].

Jitter

The ISO pointing system is described in the ISO Data User Manual [27]. It has been shown to have the following performances:

- random motion of the satellite is 0.5 arcsec (2 σ),
- absolute pointing drift is smaller than 0.1 arcsec h^{-1},
- absolute pointing error (blind pointing accuracy): around 2 arcsec (half cone).

This pointing stability is much better than the pre-launch specifications of the space-craft. However, the jitter has still some impacts on the data and if possible should be corrected, in particular when observing with the 1.5 arcsec lens (see: Section 9.7 and "CIA User's Manual [17]".

7.4 Global error budget

The uncertainties in the ISOCAM measurements depend on a variety of parameters. There should be a specific error budget for each observing mode, depending on a number of parameters such as exposure

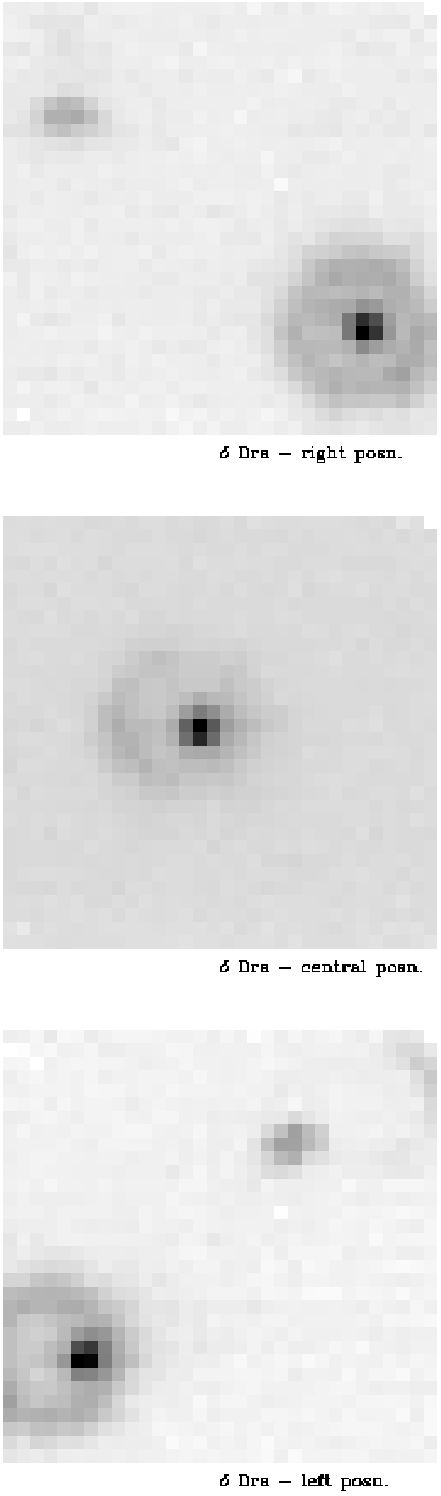

Figure 7.1: Ghost images in three CVF observations of δ Draconis. Intensity scales are logarithmic. *From "THE ISOCAM CVF CALIBRATION REPORT [8]"*

time or optical configuration. The specific error budget will of course depend on the version of calibration libraries and nonetheless on the signal processing treatments used. In Table 7.1, we compile a global list of instrumental effects contributing to the uncertainty budget. The estimated magnitude of the various effects are given for typical circumstances. Column 2 refers to the relative error before data are correted. In column 3 we give typical error magnitudes of residuals still present after utilizing available calibration methods. Those estimates should make the importance of various non-ideal behaviors of the instrument more visible. Examples are:

- the long stabilization times after flux transients,
- the memory effect of the detectors,
- the gain variations after glitches,
- the astrometric uncertainties
- the stray-light induced by CVF filters and
- the field distortion.

One may crudely estimate basic uncertainties of ISOCAM observations by using numbers as specified in Table 7.1. This table should make it obvious that any ISOCAM observer should try to correct for these various effects to the best of his knowledge. For a detailed review of the error budget see: "ISOCAM Calibration Error Budget Report [9]".

Table 7.1: Summary of the error budget for CAM calibration ("The ISOCAM Calibration Error Budget Report [9]"

Noise origin	magnitude	residual after correction
Stabilization (systematic)	40 % source flux	15 % source flux
Saturation	$\geq$ 30 % decrease in responsivity	
Read-out noise	LW: 0.7 ADU/Gain SW: 2 ADU/Gain	
Gain	0.15 % output signal	
Dark current	LW: 0.5 ADU/Gain SW: 0.5 ADU/Gain	0.25 ADU/Gain
Flat field	LW: 0.5 % (high background) SW: $\leq$ 1 %	
Sensitiv	5 %	
PSF (photometric correction)	20 %	few %
PSF (pixel position)	$\leq$ 3 %	
CVF SRF	3 %	
CVF spectral purity	$\leq$ 1 CVF step	
Ghost	$\leq$ 25 %	
Straylight	10 % sky flux	
Astrometry (pointing)	$3''$	
Astrometry (jitter)	$\pm$2 column	
Astrometry (field distortion)	1 pixel	1/3 pixel

Chapter 8

Getting started

This chapter[1] presents the contents of the ISOCAM's products in greater detail than the summary presentations given earlier. Several examples of data manipulation and visualization are given here; the examples assume that the reader has access to IDL [2]. It is also assumed that the environment includes the IDL Astronomical library (Landsman, 1995 [26]). Use a WWW browser to see:
`http://idlastro.gsfc.nasa.gov/homepage.html`
and instructions therein to down load the library.

The idea behind the chapter is to familiarize the reader with the contents of CISP and other FITS files the observer will receive after the demanded observations have been completed in a pure IDL environment without the need for an interactive analysis package. However we strongly recommend the use of the ISOCAM interactive analysis system, CIA [32] by following the instructions given in the CIA Users Manual [18]. For further information on CIA and how to get it see:
`http://www.iso.vilspa.esa.es`

Some more detailed description of the processing methods for glitch removal, detector transient handling, etc, can be found in Chapter 9. Alternatively, the reader may decide to use the CISP data in the framework of a data analysis package or custom made reduction algorithms. Do not expect to find here details on specialized algorithms (deglitching, transients, etc.).

The reader is encouraged to read this chapter next to a workstation in order to follow up the examples presented here.

8.1 Reading and displaying Auto Analysis Products

Before you try your hand at improving on the automatic data reduction, you may want to see first how the pipeline has reduced your data.

Auto Analysis top level products are CMAP and CMOS, see Sections 5.2.4 and 5.2.5. Note, however, that CMOS is generated only for raster observations; indeed, CMOS is a mosaic of individual CMAPs obtained at each raster position. An exception to the rule is the Beam Switched observations (CAM03): a CMOS product is created as the difference between the on–source observations and the mean of the reference observations.

8.1.1 Reading and displaying CMAPs

Use the IDL Astro Library to open the CMAP product:

[1] originally written by Diego A. Cesarsky
[2] IDL is a trademark of Research Systems Inc.

```
IDL> fxbopen,u1,cmap_file_name,1,cmaphdr
```

The string `cmap_file_name` shall contain not only the actual FITS file name but also the necessary path information (device, directory, etc.) using the syntax adequate to the operating system being used.

A first glimpse at the contents of CMAP is obtained by typing:

```
IDL> fxbhelp,u1
```

You will get several lines of information as shown in the Table below (the listing has been cast in two columns for compactness sake). Note that each line in the Table describes the contents of a FITS column[3]; each column has as many elements as indicated by the number of rows. For instance, there will be 102 different values for the variable `CHANNEL`.

```
FITS Binary Table:  ISOCAM stack of maps
Table contains 39 columns, by 102 rows

Col  Name                Type Size |  Col  Name                Type Size
  1  CHANNEL             Asc  1    |   21  CDELT2              Flt  1
  2  INDEX               Lng  1    |   22  CROTA2              Dbl  1
  3  TYPE                Asc  1    |   23  CD001001            Dbl  1
  4  BITPIX              Lng  1    |   24  CD002002            Dbl  1
  5  NAXIS               Lng  1    |   25  CD001002            Dbl  1
  6  NAXIS1              Lng  1    |   26  CD002001            Dbl  1
  7  NAXIS2              Lng  1    |   27  DATE-OBS            Asc  1
  8  BSCALE              Flt  1    |   28  TIME-OBS            Asc  1
  9  BZERO               Flt  1    |   29  MJD-OBS             Dbl  1
 10  BLANK               Lng  1    |   30  DATE-END            Asc  1
 11  BUNIT               Asc  1    |   31  TIME-END            Asc  1
 12  CTYPE1              Asc  1    |   32  MJD-END             Dbl  1
 13  CUNIT1              Asc  1    |   33  WAVELENG            Flt  1
 14  CRVAL1              Dbl  1    |   34  BANDWIDT            Flt  1
 15  CRPIX1              Flt  1    |   35  POLARISE            Asc  1
 16  CDELT1              Flt  1    |   36  OBSINDEX            Lng  1
 17  CTYPE2              Asc  1    |   37  DRKINDEX            Lng  1
 18  CUNIT2              Asc  1    |   38  FLTINDEX            Lng  1
 19  CRVAL2              Dbl  1    |   39  ARRAY               Lng  1156
 20  CRPIX2              Flt  1    |
```

Your CMAP file may contain a different number of rows, i.e. number of images stored in CMAP and a different `ARRAY` size.

The number of rows of CMAP is a multiple of 3. Each sky image stored in CMAP is accompanied by a second equally sized "image" indicating the noise level (per pixel) and a third "image" showing the statistical weight assigned to each pixel.

`ARRAY` data is recorded based on FITS definition of the `BSCALE` and `BZERO` keywords. In order to recreate the original value of `ARRAY`, the scaling factors should be read first[4]:

```
IDL> fxbread,u1,bscale,'BSCALE'
```

[3] Data types are shown using the notation Lng: INTEGER*4, Int: INTEGER*2, Byt: INTEGER*1, Flt: REAL*4, Dbl: REAL*8, Asc: character type. The numeral after the data type indicates the number of elements of the given type. Thus ARRAY for instance, is of type INTEGER*4 ARRAY(1156)

[4] You may specify the column number (8 and 9 for the example) instead of the column name

```
IDL> fxbread,u1,bzero,'BZERO'
```

You may verify that indeed you have read in as many values as there are rows:

```
IDL> help,bscale

BSCALE    FLOAT      = Array(102)
```

We shall proceed now to re-scale `ARRAY` to its original value. Read the auxiliary variable `cmaparray`:

```
IDL> fxbread,u1,cmaparray,'array'
```

Create a new array of type REAL*4 and same dimensions as `cmaparray`:

```
IDL> array = float(cmaparray)
```

and perform the scaling:

```
IDL> for n=0,101 do array(*,n) = float(cmaparray(*,n))*bscale(n) + bzero(n)
```

The variable `array` has now the original values estimated by Auto Analysis but it does not have yet the dimensions corresponding to a "stack" of images:

```
IDL> help,array

ARRAY     FLOAT      = Array(1156, 102)
```

The first index – 1156 – has to be broken up into rows and columns. The number of rows and columns is indicated in FITS by the keywords NAXIS1 and NAXIS2:

```
IDL> fxbread,u1,naxis1,'naxis1'

IDL> fxbread,u1,naxis2,'naxis2'
```

Each one of the 102 images has dimensions `naxis1(n) x naxis2(n), n= 0 to 101` for this example. You may inspect the size of each image:

```
IDL> print,naxis1

IDL> print,naxis2
```

The values of `naxis1(n) and naxis2(n)` should give the same image size for values of n taken by three, i.e. `n=0,1,2; n=3,4,5; etc.`; as explained above. The resulting size of every "trio" of images may differ from each other; this is due to corrections applied by Auto Analysis to individual sky images making up a CMAP averaged image. This means also that some images may have a total size less than 1156 pixels. This is because out of 1156 `array` values per row, the last values may just be fillers: only `naxis1(n)` $\times$ `naxis2(n)` values are relevant.

The last step before being able to visualize CMAP images is to cast `array` in the form of `naxis1` $\times$ `naxis2` per row. This is done using IDL's `reform` command and its keyword `sample` (see Sect.8.5).

Create an array indicating the actual number of pixels (minus one) per row:

```
IDL> nb = naxis1*naxis2 - 1
```

nb pixels (per row) shall be broken up into naxis1 and naxis2 dimensions. The command reform can be combined with the TV display command, tvscl as follows:

```
IDL> for n=0,101,3 do tvscl,reform(array(0:nb(n),n),naxis1(n),naxis2(n),1)
```

Unless you have a magnifying glass at hand, you may want to re-bin the images to, say, 10 times their original size:

```
IDL> for n=0,101,3 do tvscl,re-bin(reform(array(0:nb(n),n),naxis1(n), $

naxis2(n),1),10*naxis1(n),10*naxis2(n),sample=1)}
```

8.1.2 Reading and displaying CMOS

Since one may proceed exactly as explained for CMAP, it is left as an exercise to the reader. However, it may be useful to use differently named IDL variables than used in CMAP in case there is the need to compare CMOS and CMAP images within the same IDL session.

8.2 Reading the CISP FITS file

The main idea of this section is to allow the reader to extract images and general information from the ISO Product Distribution and compare CISP's raw images to those produced by ISOCAM's Auto Analysis. Note that a good starting point to improve on Auto Analysis data reduction is the raw data contained in the CISP file. You will have to learn how to read and interpret the contents of CISP, as well as reading and using some of the calibration files (see Chapter 4) distributed alongside with the data products.

CISP is distributed as a binary FITS table. A CISP file contains all the detector readouts – and selected ISOCAM housekeeping and spacecraft pointing information – pertaining to one given Target Dedicated Time (TDT) period. Note that there is no loss of science information – within the time bounds of the TDT – when creating CISP from CIER. In turn, no information is lost when creating CIER from the raw telemetry. The only operation performed on CIER's detector data in order to obtain CISP is the trivial operation `end_of_integration - reset`. However, `reset` data is also available in the CISP file thereby making it possible to reconstruct the `end_of_integration` information and thus recreating the contents of the raw telemetry.

8.2.1 CISP Primary Header

CISP's primary header contains a number of important data items. The header can be read as follows:

```
IDL> fxread,'fits_file_name',data,hdr
```

The string `fits_file_name` shall contain not only the actual FITS file name but also the necessary path information (device, directory, etc.) using the syntax adequate to the operating system being used.

Note that the `fxread` function will return with the message:

```
% FXREAD: File does not contain a primary array
```

This is indeed the case, as all CISP's information is in its binary extension. However, the variable `hdr` will be correctly returned by the read function. This can be confirmed with:

```
IDL> print,hdr

SIMPLE  =                    T / file does conform to FITS standard
BITPIX  =                    8 / number of bits per data pixel
NAXIS   =                    0 / number of data axes
EXTEND  =                    T / FITS dataset may contain extensions
COMMENT   FITS (Flexible Image Transport System) format defined in Astronomy and
COMMENT   Astrophysics Supplement Series v44/p363, v44/p371, v73/p359, v73/p365.
COMMENT   Contact the NASA Science Office of Standards and Technology for the
COMMENT   FITS Definition document #100 and other FITS information.
ORIGIN  = 'ESA     '           / European Space Agency
TELESCOP= 'ISO     '           / Infrared Space Observatory
INSTRUME= 'CAM     '           / Instrument used
COMMENT    CAM Standard Processed Data
FILENAME= 'CISP08101204'       / File name
DATE    = '10/07/98'           / Creation date 98/191
FILEVERS= '1415    '           / Version ID (derived from creation date)
OLPVERS = 'OLP_70  '           / SOC OLP system version
CALGVERS= 'CALG_40 '           / SOC OLP CAL-G files version
USERNAME= 'PIPELINE'           /
OBJECT  = 'HD97300 '           / Target ID as given by proposer
OBSERVER= 'RSIEBENM'           / Proposer ID in ISO Mission DB
EQUINOX =               2000.0 / Equinox
TMRATE  =                   32 / Telemetry rate in Kbps (Kbits/sec)
EOHAUTCS= '96037143827'        / Approx. UTC of start of observation
EOHAUTCE= '96037154925'        / Approx. UTC of end of observation
EOHAAOTN= 'C01     '           / AOT name
EOHAPLID= 'RSIEHEBE'           / Proposal ID
EOHAOSN = '04      '           / Observation sequence number
EOHAPSN = '00      '           / Pointing sequence number
EOHAPCAT= '4       '           / Proposal category
EOHACIND= '        '           / Calibration indicator
EOHATTYP= '003     '           / Target type
AOTVERS = '02.92   '           / AOT-to-OCT logic version
ATTUTCSL= '96037143235'        / UTC of start time of slew to intended target
ATTUTCS = '96037143800'        / UTC of time of first arrival at intended target
ATTOTFTH=                 10.0 / On-target flag threshold (arc secs)
ATTRA   =            167.46079 / Intended Right Ascension of instrument viewing
ATTDEC  =            -76.61311 / Intended DEClination (with ATTRA)
ATTTYPE = 'R       '           / Type of attitude operation (P/R/T)
ATTRNPTS=                    4 / Number of points per line of the raster
ATTRNLNS=                    4 / Number of lines in whole raster pointing
ATTRDPTS=                    5 / Distance (arcsecs) between adjacent points
ATTRDLNS=                    5 / Distance (arcsecs) between adjacent lines
ATTRORIE=                    0 / ORIENTation flag (0 or 1)
ATTRROTA=                 90.0 / ROTATE of raster pattern (degrees)
ATTGUIDE=                63062 / Guide star reference number
ATTSAANG=                 86.2 / Solar aspect angle (degrees)
ATTERROR=                    0 / CONTINGEncy flag(0=success; 1=target not acq'd)
TREFUTC1=            224001508 / UTC (whole seconds since 01-01-1989)
TREFUTC2=              4066340 / UTC (remaining fraction of second)
TREFUTK =            202404222 / ISO Uniform Time Key (UTK)
```

```
TREFITK =             60239551 / ISO INSTRUMENT Time Key (ITK)
TREFITKU=     0.13999950000000 / ITK unit length in seconds
INSTRA  =            167.46082 /Reference instrument J2000 right ascension (deg
INSTDEC =            -76.61311 /Reference instrument J2000 declination (deg)
INSTROLL=               153.22 /Reference instrument J2000 roll angle (deg)
CINSTRA =            167.46006 /Corrected reference instrument J2000 right asce
CINSTDEC=            -76.61352 /Corrected reference instrument J2000 declinatio
CINSTROL=               153.22 /Corrected reference instrument J2000 roll angle
ISRRSEV =                    0 / RTA maximum severity level
ISRQSEV =                    0 / QLA maximum severity level
ISRRWARN= '??      '           / RA-warning flag from IS user
ISRNOOSL=                    0 / No. of out-of-soft-limit errors
ISRNOOHL=                    0 / No. of out-of-hard-limit errors
ISRNMW  =                    0 / No. of monitor warnings
ISRNCVW =                    0 / No. of command verification errors
ISRNBTW =                    3 / No. of bad telemetry errors
ISRNMVW =                    0 / No. of memory verification errors
ISRNSQLA=                  252 / No. of severe QLA errors
TREFCOR1=            224001507 /UTC of 1st reference time
TREFPHA1=             81.59903 /Orbital phase at TREFCOR1
TREFHEL1=                32.81 /Heliocentric correction +(s) at TREFCOR1
TREFDOP1=                11.87 /ISO velocity towards target (km/s) at TREFCOR1
TREFCOR2=            224003636 /UTC of 2nd reference time
TREFPHA2=             81.62372 /Orbital phase at TREFCOR2
TREFHEL2=                32.72 /Heliocentric correction +(s) at TREFCOR2
TREFDOP2=                11.86 /ISO velocity towards target (km/s) at TREFCOR2
TREFCOR3=            224005765 /UTC of 3rd reference time
TREFPHA3=             81.64842 /Orbital phase at TREFCOR3
TREFHEL3=                32.63 /Heliocentric correction +(s) at TREFCOR3
TREFDOP3=                11.86 /ISO velocity towards target (km/s) at TREFCOR3
VERS1   = '1415/CSTA08101204'  / Version ID of each input file
VERS2   = '0706/CHCGCONV'      / Version ID of each input file
VERS3   = '1261/CSWCVF'        / Version ID of each input file
VERS4   = '1261/CLWCVF1'       / Version ID of each input file
VERS5   = '1261/CLWCVF2'       / Version ID of each input file
VERS6   = '0706/CCGSWDEAD'     / Version ID of each input file
VERS7   = '0706/CCGLWDEAD'     / Version ID of each input file
VERS8   = '0706/CSCGCROSS'     / Version ID of each input file
VERS9   = '0706/CWHEELS'       / Version ID of each input file
VERS10  = '1415/CIER08101204'  / Version ID of each input file
VERS11  = '1415/IRPH08101204'  / Version ID of each input file
VERS12  = '0513/CISR08101204'  / Version ID of each input file
VERS13  = '1350/ORBIT'         / Version ID of each input file
XTENSION= 'BINTABLE'           / Binary table FITS extension
BITPIX  =                    8 / 8 BITS character format
NAXIS   =                    2 / Tables are 2-d character array
NAXIS1  =                 4600 / Characters in a row
NAXIS2  =                 1974 / Number of rows in a table
PCOUNT  =                    0 / Parameter count always 0
GCOUNT  =                    1 / Group count always 1
TFIELDS =                   34 / No of columns in table
TFORM1  = '1J      '           /
```

```
TTYPE1  = 'GPSCTKEY'           / Instrument time key

...

TTYPE32 = 'CISPDATA'           / Detector data (EOI - RESET)
TFORM33 = '1122I   '           /
TTYPE33 = 'CISPRESE'           / Reset level for NORMAL processing
TFORM34 = '64I     '           /
TTYPE34 = 'CISP3334'           / Columns 33 & 34 of SW readout
EXTNAME = 'CISP    '           /
CDARK1  = 'DRKLW015'           / DARK   to use w/config
COFLAT1 = 'OFLLW6.0'           / O-FLAT to use w/config
CDFLAT1 = 'DFLLWLW4'           / D-FLAT to use w/config
CDARK2  = 'DRKLW015'           / DARK   to use w/config
COFLAT2 = 'OFLLW1.5'           / O-FLAT to use w/config
CDFLAT2 = 'DFLLWLW8'           / D-FLAT to use w/config
CDARK3  = 'DRKLW015'           / DARK   to use w/config
COFLAT3 = 'OFLLW1.5'           / O-FLAT to use w/config
CDFLAT3 = 'DFLLWLW5'           / D-FLAT to use w/config
CDARK4  = 'DRKLW015'           / DARK   to use w/config
COFLAT4 = 'OFLLW1.5'           / O-FLAT to use w/config
CDFLAT4 = 'DFLLWLW4'           / D-FLAT to use w/config
CDETE01 = 'LW      '           / Detector ID (obs)
CTINT01 =                   15 / Tint (obs)
CEGAN01 =                    0 / Electronic gain (obs)
CEOFS01 =                    1 / Electronic offset (obs)
CWLNS01 = '6.0     '           / Lens wheel (obs)
CWENT01 = 'HOLE    '           / Entrance wheel (obs)
CWFLT01 = 'LW4     '           / Filter wheel (obs)
CWSEL01 = 'LWFM    '           / Selection wheel (obs)
CDETE02 = 'LW      '           / Detector ID (obs)
CTINT02 =                   15 / Tint (obs)
CEGAN02 =                    0 / Electronic gain (obs)
CEOFS02 =                    1 / Electronic offset (obs)
CWLNS02 = '1.5     '           / Lens wheel (obs)
CWENT02 = 'HOLE    '           / Entrance wheel (obs)
CWFLT02 = 'LW8     '           / Filter wheel (obs)
CWSEL02 = 'LWfm    '           / Selection wheel (obs)
CDETE03 = 'LW      '           / Detector ID (obs)
CTINT03 =                   15 / Tint (obs)
CEGAN03 =                    0 / Electronic gain (obs)
CEOFS03 =                    1 / Electronic offset (obs)
CWLNS03 = '1.5     '           / Lens wheel (obs)
CWENT03 = 'HOLE    '           / Entrance wheel (obs)
CWFLT03 = 'LW5     '           / Filter wheel (obs)
CWSEL03 = 'LWfm    '           / Selection wheel (obs)
CDETE04 = 'LW      '           / Detector ID (obs)
CTINT04 =                   15 / Tint (obs)
CEGAN04 =                    0 / Electronic gain (obs)
CEOFS04 =                    1 / Electronic offset (obs)
CWLNS04 = '1.5     '           / Lens wheel (obs)
CWENT04 = 'HOLE    '           / Entrance wheel (obs)
```

```
CWFLT04 = 'LW4     '           / Filter wheel (obs)
CWSEL04 = 'LWfm    '           / Selection wheel (obs)
```

We shall come back to the contents of the primary header as some numerical values found there are necessary to further processing steps. For a full description of the key words we refer to the ISO Data Product Document (IDPD) available on our ISO web page. Noteworthy items are the version number of the calibration files accessed by the code that generated CISP.

8.2.2 Contents of the CISP Binary Extension

The binary extension of the FITS file can easily be opened for reading with the following command:

```
IDL> fxbopen,u3,'fits_file_name',1
```

The contents of the binary extension can now be listed:

```
IDL> fxbhelp,u3
```

As it was the case with CMAP and CMOS, you will get several lines of information (rearranged in two columns here for conciseness):

```
FITS Binary Table:  CISP
Table contains 32 columns, by 4437 rows

Col  Name               Type Size | Col  Name               Type Size

  1  GPSCTKEY           Lng  1    |  17  CISPCVFS           Byt  1
  2  GPSCRPID           Byt  2    |  18  CISPTEMP           Flt  10
  3  GPSCFILL           Int  1    |  19  CISPBLKB           Flt  2
  4  CISPDEID           Byt  1    |  20  CISPGAIN           Byt  1
  5  CISPCNFG           Byt  1    |  21  CISPOFFS           Byt  1
  6  CISPQFLG           Byt  1    |  22  CISPBSFG           Int  1
  7  CISPPROC           Byt  1    |  23  CISPMEAN           Flt  1
  8  CISPDISC           Int  1    |  24  CISPMEDI           Flt  1
  9  CISPPFOV           Int  1    |  25  CISPSRMS           Flt  1
 10  CISPITIM           Int  1    |  26  CISPICNT           Int  1
 11  CISPFCVF           Int  1    |  27  CISPACSA           Byt  1
 12  CISPWAVE           Flt  1    |  28  CISPAOBS           Byt  1
 13  CISPEWHL           Int  1    |  29  CISPBIAS           Flt  10
 14  CISPSWHL           Int  1    |  30  CISPDATA           Int  1024
 15  CISPWDTH           Int  1    |  31  CISPRESE           Int  1122
 16  CISPOBST           Byt  1    |  32  CISP3334           Int  64
```

Your actual list will certainly have a different number for the row count (4437 for the example here). The row count is the number of ISOCAM detector readouts present in the FITS file.

The meaning of each Column label appearing in the above table is the following:

1. GPSCTKEY ISOCAM's time key, see Section 8.3 for more details.

2. `GPSCRPID` ISO's raster point ID. When applicable, GPSCRPID(0) gives the raster M point along GPSCRPID(1), the raster N or line number identifier. Staring, beam switch and CVF observations have a raster point ID of (1,1). A raster point ID of (0,0) indicates an invalid observation.

3. `GPSCFILL` Filler

4. `CISPDEID` ISOCAM detector identification: 0 for SW; 1 for LW

5. `CISPCNFG` Configuration number within the AOT

6. `CISPQFLG` Composite quality flag, constructed as `CISPQFLG` = `QLA_FLAG` + 2 × `OTF` + 4 × `OTF_upper`. `QLA_FLAG` is set to one by the AOT logic to indicate that the demanded stabilization time (via PGA) has elapsed; `OTF` is ISO's on target flag, synchronized with ISOCAM's readout electronics; `OTF_upper` is the logical AND of individual `OTF` in the case of an accumulated mode observation.

7. `CISPPROC` ISOCAM on–board processing mode: 0 for normal mode; 1 for accumulated mode; 2 for sampling mode

8. `CISPDISC` CIER/CSTA discrepancy flag. It warns that individual CISP rows (more precisely, the corresponding CIER rows) indicate different house-keeping parameters than the values recorded in the CSTA file for the corresponding configuration and operational mode. In general, CISPDISC should be ignored by CISP users except for the very rare case of motor positions different from their commanded positions. For the sake of completeness, the algorithm used to construct CISPDISC is presented here.

```
First byte
bit  0: detector ID does not match its CSTA value
bit  1: Tint does not match its CSTA value
bit  2: ENTW does not match its CSTA value
bit  3: SELW does not match its CSTA value
Second byte
bit  4: FLTW does not match its CSTA value
bit  5: LNSW does not match its CSTA value
bit  6: EntWhl motor position differs from its commanded position
bit  7: SelWhl motor position differs from its commanded position
Third byte
bit  8: LnsWhl motor position differs from its commanded position
bit  9: FltWhl motor position differs from its commanded position
bit 10: CNF in CIER differs from CNF in CSTA (ISOCAM team only, ignore)
bit 11: OPM in CIER differs from OPM in CSTA (ISOCAM team only, ignore)
Fourth byte
bit 12: AOBS in CIER differs from FOBS in CSTA (ISOCAM team only, ignore)
bit 13: AAOT in CIER differs from AAOT in CSTA (ISOCAM team only, ignore)
```

9. `CISPPFOV` Lens wheel step number. See Figure 8.1 for the correspondence between Step Number and optical component

10. `CISPITIM` Elementary integration time in CAM Time Units

11. `CISPFCVF` Filter wheel step number

12. `CISPWAVE` Wavelength of chosen filter [microns]

13. `CISPEWHL` Entrance wheel step number

14. `CISPSWHL` Selection wheel step number

15. `CISPWDTH` Equivalent width of filter [milli–microns]
16. `CISPOBST` Op–mode: 0 for mode changing, 1 for OBS; 2 for DRK; 3 for CAL; and 4 for CLEAN or FLOOD
17. `CISPCVFS` CVF Scan on or off: 0 for OFF; 1 for ON
18. `CISPTEMP` Array of selected temperatures in and around ISOCAM's focal plane unit; units of [K].
19. `CISPBLKB` Black body voltage and current. Dissipated power is given by their product.
20. `CISPGAIN` Electronics gain (internal on–board processor code). Actual gain is $\texttt{GAIN} = 2^{\texttt{CISPGAIN}}$.
21. `CISPOFFS` Electronics offset
22. `CISPBSFG` Beam Switching Flag (relevant only for Beam Switching observations). This datum is constructed as follows: $\texttt{CISPBSFG} = 256 \times \texttt{Cycle\_number} + \texttt{Reference\_number}$. Furthermore, add 16 to the result if observing a Reference position.
23. `CISPMEAN` Mean value of `current_image - previous_image`
24. `CISPMEDI` Median value of `current_image`
25. `CISPSRMS` RMS value of `current_image - previous_image`
26. `CISPICNT` Current internal image counter
27. `CISPACSA` Accumulated or Sampled count; use together with CISPPROC. The actual number of accumulated/sampled images is the value of this datum plus one.
28. `CISPBIAS` Array of selected polarization bias for current detector.
29. `CISPDATA` 32×32 array of pixel values, in units of analog to digital units (ADU). Derived as `CISPDATA = end_of_integration - reset`. The difference is performed on–board for Accumulated and Sampled modes, or it is computed by *Derive_SPD* for Normal mode.
30. `CISPRESE` 32×32 array of pixel `reset` values (LW detector) or 34×33 array of pixel `reset` values (SW detector). Null array for accumulated or sampled on–board mode. Units of ADU.
31. `CISP3334` Values of 32 pixels of columns 33 and 34 of SW detector. Units of ADU.

8.2.2.1 Wheel step number and optical components

Most ISOCAM data products report wheel positions by their angular displacement with respect to some fix and arbitrary zero position; angular displacement is measured in step numbers [5].

8.3 Time and time scales in ISO products

This section on times and time scales is necessary for those who may try to correlate their ISOCAM observations with other ISO products.

There are several time definitions applicable to various ISO products. This is not altogether a satisfactory situation as it lends itself to possible errors, needless complications and potentially to a loss of precision. To further complicate matters a given time is some times expressed in seconds and fraction, and other times as `yydddhhmmss`.

[5] Step numbers is in fact the count of gear teeth from the zero position

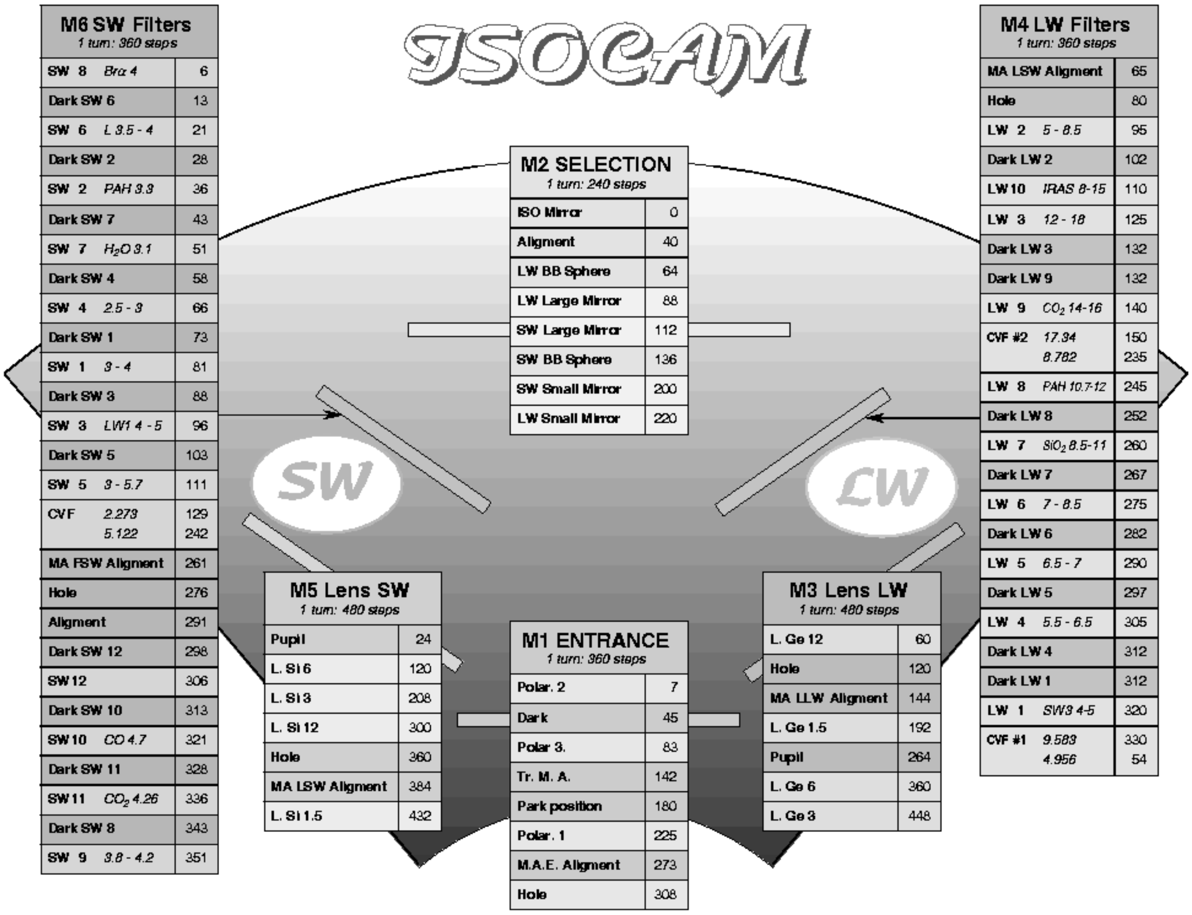

Figure 8.1: We give the correspondence between step number and optical component. Note that only optical elements are shown, viz. lenses, filters and polarizers. That is, although lens and filter wheels are moved to special positions during dark operations, the corresponding numbers of those steps are not shown.

The official description of time and timing can be found in document ISO Architectural Design Document [23] (for off–line software); (see also the ISO Data User Manual [27]). The Architectural Document states that there are essentially only two physical different scales of time, as far as off–line analysis software is concerned: Ground Station Time (GST) and on–board time (measured by various oscillators on–board. The on–board time referred to in the design document is the format number: the format number is generated by the On Board Handling System (OBDH) as a counter incrementing by one every 24th frame reading pulse (remember that 24 frames per format are transferred from the focal plane units to the OBDH subsystem). This is because a new telemetry format is generated every two seconds, the format counter is a clock ticking at 0.5 Hz. Although this time looks adequate, a second time was introduced into the system: the already mentioned GST – called UTC sometimes. There may be a systematic and varying delay between the on–board and the GST clock, but this delay is "absorbed" by the telemetry reception and time tagging on ground systems.

Another two times are defined, related to the GST. The first is the so called Uniform Time Key, UTK, defined as:

- a normal signed I*4 quantity.
- increasing at a rate of 24 Hz.
- divisible by 48 on a format boundary.
- initialized to zero at an arbitrary time close to ISO launch.
- having a constant linear relation to the real spacecraft time, as long as the latter increments nominally.
- guaranteed to increase with time, even across a non–nominal jump of the real spacecraft clock, and even in a case of loss of telemetry frame 0.

The fourth time is the Instrument Time Key (ITK). The ITK was introduced because some ISO instruments required a finer clock division than the 24 Hz afforded by UTK. The ITK for ISOCAM is defined as UTK, viz. initialized to zero same time before launch and monotonically increasing throughout the mission. It is computed in ISOCAM Time Units (1 CAMTU $\sim$ 0.14s, is derived from the ISOCAM on–board clock and it is the time unit used to measure the on–chip integration time).

8.3.1 Transforming ITK to UTK

The ITK to UTK transformation is necessary as several general products are UTK time tagged, particularly the Instrument Instantaneous Pointing History.

The relationship between UTK and ITK of ISOCAM is given by the following expression:

$$\mathtt{UTK(t)} = \mathtt{UTK(t_0)} + \mathtt{const} \times [\mathtt{ITK(t)} - \mathtt{ITK(t_0)}]$$

$\mathtt{t_0}$ is an arbitrary time; the value of UTK and of ITK at this arbitrary time are given in CISP's primary header.

Remember that UTK is expressed in units of 24 Hz and ITK in units of CAMTU. It follows that `const` above is given by:

$$\mathtt{const} = \mathtt{CAMTU} \times \mathtt{24.0}$$

Actual values of $\mathtt{CAMTU}, \mathtt{UTK(t_0)}$ and $\mathtt{ITK(t_0)}$ can be found in CISP's primary header:

Value of CAMTU:

```
IDL>  camtu = fxpar(hdr,'TREFITKU')
```

IKEY value at time t_0:

```
IDL>  itk0 = fxpar(hdr,'TREFITK')
```

UTK value at time t_0:

```
IDL>  utk0 = fxpar(hdr,'TREFUTK')
```

See the values read:

```
IDL> print,camtu,itk0,utk0

     0.1399995    28777108    96694560
```

for the example at hand.
We can now create the UTK array corresponding to CISP's ITK array. First read in CISP's ITK array:

```
IDL> fxbread,u3,itkey,'GPSCTKEY'
```

Next, create CISP's UTK array (note that we use the implicit indexing rules of IDL, namely **utkcisp** will be created as an array of same dimensions as **itkey**):

```
IDL> utkcisp = 24.0d0*camtu*(itkey - itk0) + utk0
```

The **utkcisp** array thus obtained will be the entry point to the IIPH information. Note, however, that **utkcisp** is estimated at a sampling rate equal to the on–chip integration time; the pointing informations is itself sampled every 0.5 seconds – or 12 UTK units. For a typical integration time of 15 CAMTU, **utkcisp** is sampled every 50.4 UTK units, some four times coarser sampling than the IIPH information.

8.4 ISOCAM Pointing Information

The general pointing information of ISO is detailed in ISO Data User Manual [27]. We give here some of the information relevant specific to the ISOCAM instrument. The Pointing information is distributed via the Instrument Instantaneous Pointing History (IIPH) product.
To open the binary IIPH product proceed as you did for the CISP file:

```
IDL> fxbopen,u4,'iiph_fits_file',1
```

Note that the IIPH file is being opened in unit **u4** as not to disturb CISP which was opened in unit **u3**.
Read the RA, DEC, and UTK:

```
IDL> fxbread,u4,ra,'ra'

IDL> fxbread,u4,dec,'dec'

IDL> fxbread,u4,utkiiph,'utk'
```

RA and DEC are both expressed in degrees. To get a first idea of the behavior of the spacecraft during your observations, you may plot DEC vs. RA:

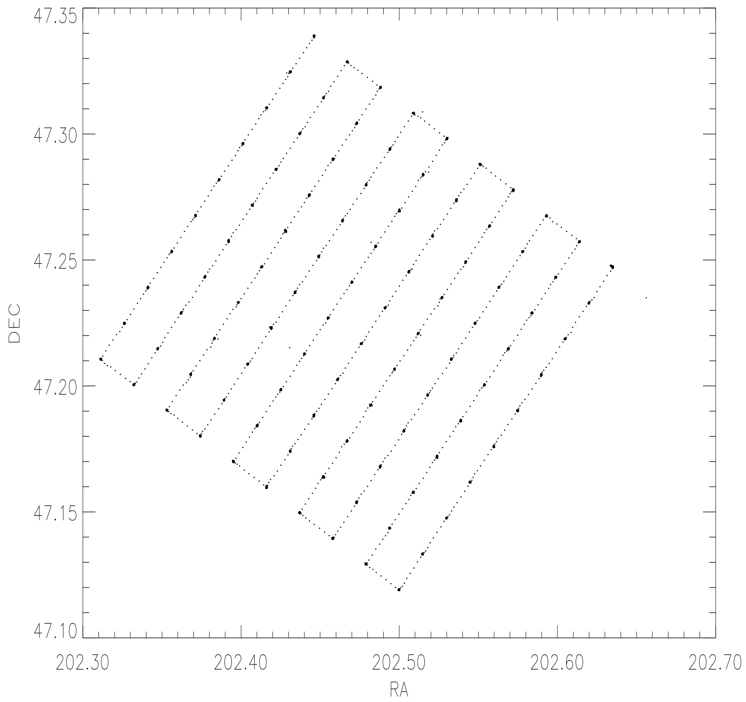

Figure 8.2: DEC vs. RA for a raster observation.

```
IDL> plot,ra,dec,psym=3,xtit='RA',ytit='DEC',ynozero=1
```

If desired, the mean RA and DEC positions may be estimated and printed out in hexadecimal notation:

```
IDL> ramean = total(ra)/n_elements(ra)

IDL> decmean = total(dec)/n_elements(dec)
```

Convert to hexadecimal notation and print out:

```
IDL> radec,ramean,decmean,rahr,ramin,rasec,decdeg,decmin,decsec

IDL>  print,rahr,ramin,rasec

      13       29       54.102055

IDL> print,decdeg,decmin,decsec

      47       13       46.429204
```

For the example at hand: RA = 13H29M54.1S and DEC = 47D13M46.4S.

It is instructive to plot a close up of the RA read from IIPH (our example is based on a raster observation), as a function of **utkiiph**, and to overplot the On Target Flag (OTF) read from CISP, itself as a function of **utkcisp**.

Figure 8.3 was created using the following IDL instructions:

```
IDL> plot,utkiiph,ra,xra=[9.67e07,9.671e07],xtit='UTK time'
```

Read On Target Flag related data in CISP:

```
IDL> fxbread,u3,qflg,'CISPQFLG'
```

The actual On Target Flag is given by bits 1 and 2 of CISPQFLG (bit 0 is the PGA programmable QLA flag; see Section 8.2.2); the factor 500 is just to scale the figure into the RA plot:

```
IDL> oplot,utkcisp,((qflg and 6) eq 6)*500,lin=1
```

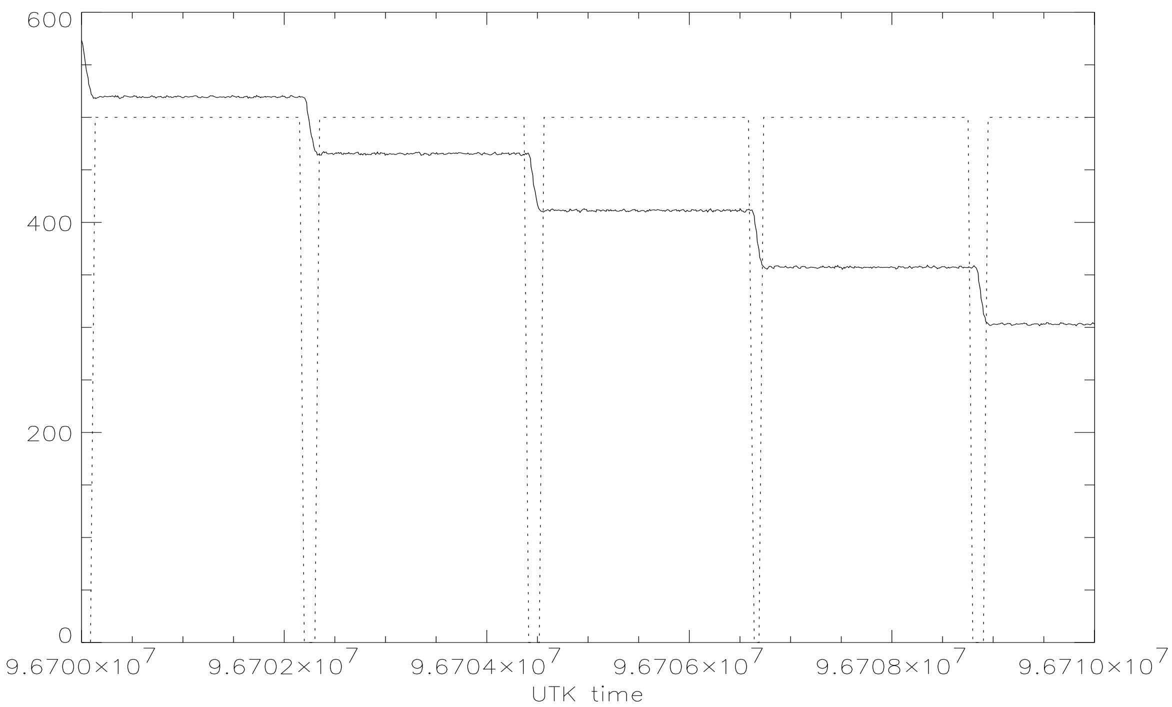

Figure 8.3: A close up of RA vs. UTK (from the IIPH file) and overplot (dotted) of the On Target Flag read from CISP).

8.5 Science data in CISP

The science data in CISP is recorded under the column label CISPDATA. Note that the FITS variety used in ISO does not say explicitly that CISPDATA is of the form 32×32. We shall proceed in two steps to create the 32×32 pixel array:

Read CISPDATA

```
IDL> fxbread,u3,array,'CISPDATA'
```

```
IDL> help,array

ARRAY     INT     = Array(1024, 4437)
```

Indeed, `array` contains all 1024 pixels one after the other; reform to $32 \times 32 \times depth\_of\_array$; where $depth\_of\_array = 4437$ for the example at hand.

```
IDL> pixarr = reform (array,32,32,4437)
```

You may convince yourself that `pixarr` has the required dimensions:

```
IDL> help,pixarr

PIXARR     INT     = Array(32, 32, 4437)
```

How does the raw data look like? A quick look of `pixarr` can be performed now. Load a lookup table:

```
IDL> loadct,15
```

Visualize each image, enlarge to 128×128 pixels:

```
IDL> for n=0, 4436 do tvscl,rebin(pixarr(*,*,n),128,128,sample=1)
```

`sample=1` (faster) causes rebin to create new pixels by pixel replication; a value of 0 (slower) causes rebin to create new pixels by interpolating between existing pixels.

Note that, as far as IDL is concerned, you could have by–passed the step to create `pixarr`:

```
IDL> for n=0, 4436 do tvscl,rebin(reform(array(*,n),32,32,1),128,128,sample=1)
```

Very likely, you will find that the most striking feature of your ISOCAM images is the large number of cosmic hits and the disconnected column 24 (column 23 using IDL's indexing convention) of the LW detector.

8.5.1 Elementary deglitching

For the purpose of this guided tour, you may obtain quite acceptable results using a median filter on the time history of each one of the 32×32 pixels of ISOCAM. Please see Section 9.2 for a description of enhanced deglitching algorithms. Referring again to the `array` data read earlier from CISP, you may try a median filter to eliminate glitches:

```
IDL> array3 = array

IDL> for n=0,1023 do array3(n,*) = median(reform(array(n,*),3)
```

Here a median filter of width 3 was used; you may try other filter widths. For example in OLP a filter width of 20 is used. Figure 8.4 shows the time history of a pixel in a raster observation, and the result of having used a median filter of width 3 (middle plot) and width 7 (lower-most plot).

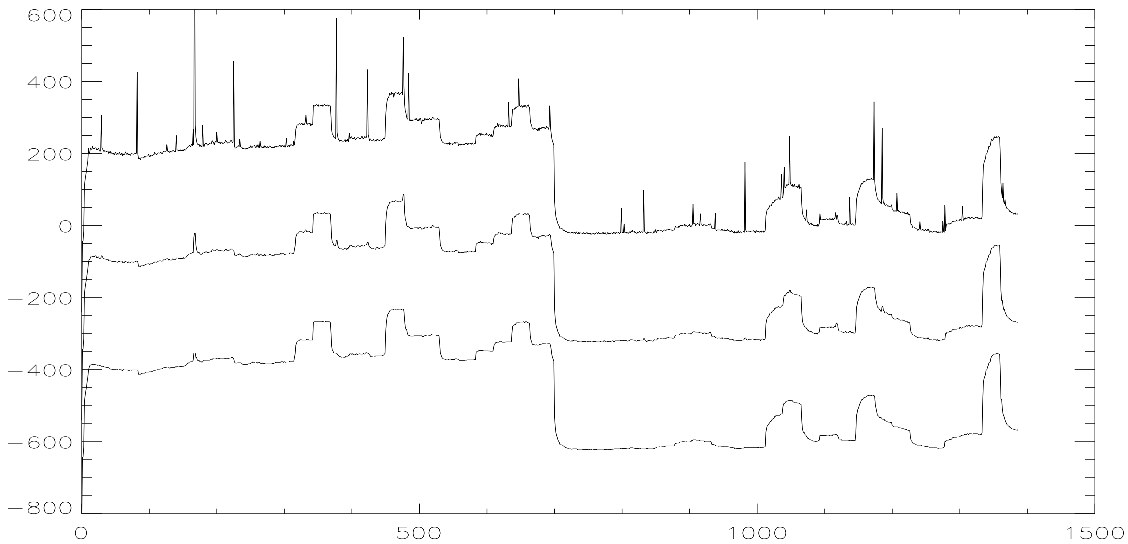

Figure 8.4: The time history of a pixel in a raster observation. Topmost plot shows raw data; the two lower ones show the result of a median filter of width 3 and of width 7 respectively. Plots are vertically shifted by 300 units for clarity.

8.5.2 Detector transient and memory effects

The ISOCAM detectors show relaxation with multiple time constants and physical dependencies. A deeper discussion on the response of the detector after having seen a gradient in the flux, the so called transients, are given in Section 9.6. One illustration is given in Figure 8.5. The example is drawn from a 11 by 11 raster observation where a strong point–like source was observed at M = 3, N = 6. A close up of the time history of the 3×3 surrounding pixels has been plotted, together with an overplot of GPSCRPID(0).

Two important features are apparent in the figure:

1. The up–flux transient. The strong source comes within ISOCAM's field of view at raster step M=3 (y=30, x=2550, in the figure). The point source reaches about 60% of its peak intensity. A close–up examination of the plot reveals that the asymptotic value is not reached during the observation of this raster point although the total duration is of some 45 readouts per raster position. Beware of this behavior when trying to perform accurate photometry.

2. The down–flux step. The number of counts decreases rapidly upon leaving raster point 3 and it can be seen to continue to decrease throughout raster points 2 and well into raster point 1. Obviously, a ghost image of this source will be visible at the two next raster points, M = 2 and M = 1. Beware of this effect as it may lead to the "discovery" of non existing sources.

8.6 Reading and using Calibration files

Calibration libraries were introduced in Chapter 4, here we shall briefly explain how to read them and how to apply their information to CISP's raw data. We shall present, in that order, dark subtraction, flat–field corrections, and calibration from engineering units to astronomical units. For a deeper discussion on data

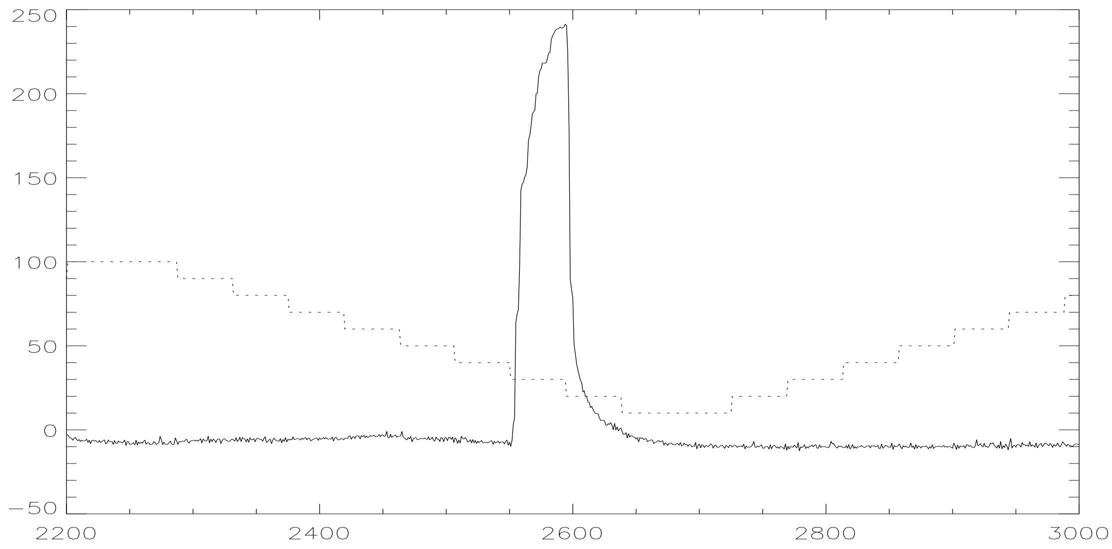

Figure 8.5: A close up of the time history of a point–like source in a raster scan. Dotted line: 10 times the M_raster identifier.

processing methods see Chapter 9. Dark and flat field corrections consist of subtracting the dark current emission from each sky frame, and correcting the resulting image for pixel–to–pixel variable responsivity:

$$corrected\_sky(col, lin) = (sky(col, lin) - dark(col, lin))/flat(col, lin)$$

8.6.1 Dark Current Calibration Files

There are two sets of Dark Current Calibration Files one for each ISOCAM detector, SW and LW. The files are called CCG*DARK. These files are opened and read as explained before for all other FITS files. As usual (the example is based on the LW file):

```
IDL> fxbopen,udark,'name_LWDARK_file',1,hdrdark


IDL> fxbhelp,udark
```

```
FITS Binary Table:  CCGLWDARK
Table contains 24 columns, by 5 rows
Col  Name               Type Size |  Col  Name               Type Size
  1  SPARE              Byt  36   |   13  VMIN               Flt  10
  2  TINT               Int  1    |   14  VMAX               Flt  10
  3  EWHL               Int  1    |   15  BITPIX             Lng  1
  4  SWHL               Int  1    |   16  NAXIS              Lng  1
  5  GAIN               Byt  1    |   17  NAXIS1             Lng  1
  6  OFFSET             Byt  1    |   18  NAXIS2             Lng  1
  7  TEMPERAT           Flt  1    |   19  NAXIS3             Lng  10
  8  TRMS               Flt  1    |   20  BSCALE             Flt  10
  9  TMIN               Flt  1    |   21  BZERO              Flt  10
 10  TMAX               Flt  1    |   22  BLANK              Lng  10
```

```
 11  VOLTAGE                 Flt  1     |   23  BUNIT                  Asc  10
 12  VRMS                    Flt  10    |   24  ARRAY                  Lng  2048
```

Most keywords are known already from the previous examples. The ever present `ARRAY` keyword has here a slightly different meaning than elsewhere. It contains 2048 pixel values: the first 1024 are dark current values; the second 1024 are the rms noise of the previous 1024 values.

To proceed, you should read in numerical values for keywords `TINT, BSCALE, BZERO, ARRAY` and generate `darkarray` applying `bscale` and `bzero` to `ARRAY` as already done for the CMAP file, Section 8.1.1.

Values of `TINT`, column 2 in the table above, are important as they determine which `darkarray(0:1023,row)` is to be subtracted from your sky observations. The proper row is that corresponding to the same elementary integration time as used to observe the sky, given as `CISPITIM` in the CISP file. LW dark currents have been measured and cataloged for the five allowed integration times: 2, 15, 36, 72, and 144 [CAMTU]; SW dark currents are available for 2, 15, 43, 144, and 430 [CAMTU]. Thus, if your LW observations were obtained with an elementary integration time `CISPITIM = 15 [CAMTU]`, you should use `darkarray(0:1023,1)`.

CISP reports as many values of `CISPITIM` as there are rows in the binary file. In principle, all values pertaining to a given configuration `CISPCNFG` and operational mode `CISPOBST` should be identical. In fact, there are two means to make sure that this is indeed the case:

1. Open and read the corresponding CSTA file. The column `CSTATINT` lists the integration times used per operational mode and per configuration. The flag `CISPDISC` (Section 8.2.2) will tell you whether or not the individual CISP rows have used an integration time different from that indicated in the compact status file CSTA.

2. You may use some IDL shortcuts (see Section 8.6.4) to verify that all of your sky observations have been performed with the same integration time.

Catalogued dark current values are normalized to an integration time of 1 sec. The current version of the dark calibration files has been established for `CISPGAIN = 1`, that is GAIN = 2. Hence, before performing the actual data subtraction, both CISP and dark data have to be scaled to the same conditions. It is a good practice to scale all your data to unitary integration time such as seconds and not CAMTU, also the gain should be scaled to one. Hence, you may need to scale both, CISP and dark, to the recommended conditions. Integration time scaling is performed by dividing each pixel value by

$$\mathtt{tint} = \mathtt{CISPITIM} \times 0.1399995$$

One further warning: if your sky data was obtained in the accumulated mode, the elementary integration time is then

$$\mathtt{tint} = \mathtt{CISPITIM} \times (\mathtt{CISPACSA} + 1) \times 0.1399995$$

Dark (and flat field) corrections can be applied to the average sky image rather than to every image used to compute the average. Indeed:

$$corrected\_sky_{aver} = 1/N \times \sum_{n=1}^{N} (sky(n) - dark)/flat$$

$$corrected\_sky_{aver} = (sky_{aver} - dark)/flat$$

However, it is advisable to create individually corrected images

$$corrected\_sky(n) = (sky(n) - dark)/flat$$

which can be displayed and visually inspected one by one.

8.6.2 Flat Calibration Files

Flat field calibration corrections are necessary due to the combined result of two different physical effects: (a) the intrinsic pixel–to–pixel gain variation in the detector itself, and (b) the illumination non homogeneity due to the optical components, for instance vignetting. Each physical phenomenon is described by a different set of flat field correctors: Detector flat corrections and Optical flat corrections. The corrected image is obtained thus:

$$corrected\_sky(col, lin) = [sky(col, lin) - dark(col, lin)]/[detf(col, lin) \times optf(col, lin)]$$

where $detf$ and $optf$ are short for detector flat and optical flat corrections; (col, lin) means that the arithmetical operations are performed for every pixel in a column and a row.

8.6.2.1 Detector Flat Calibration Files

Correcting information is stored in files CCxDFLT S, where x stands for SW or LW. Start with the now usual IDL operations:

```
IDL> fxbopen,u5,'detector_flat_file',1,dflathdr

IDL> fxbread,u5,dfltarray,'array'

IDL> fxbread,u5,bscale,'bscale'

IDL> fxbread,u5,bzero,'bzero'
```

Next scale `dfltarray` back to its true value using `bscale` and `bzero`, see Section 8.1.1.

The current LW file contains detector flat field corrections for filters LW2 and LW3; choose the filter row whose wavelength is closest to that of your own observations. Proceed in the same manner for the SW files.

8.6.2.2 Optical Flat Calibration Files

Optical flat corrections are quite minor when using the 1.5 and 3.0 arcsec/pixel fields of view (there is little, if any, vignetting). In fact, the current calibration files are identically unity for those fields of view.

The current set of LW optical flat correction files contain information for the following combination of filters and field of view:

Row	PFOV	Filter
1	1.5	LW2
2	1.5	LW10
3	1.5	LW3
4	6.0	LW2
5	6.0	LW3
6	6.0	LW10
7	3.0	LW2
8	3.0	LW2

Choose the row most closely matching your observations.

8.6.3 Engineering to flux conversion

Assume that you have read CISP images, normalized them for integration time and gain. Assume further that you have corrected them for dark current and flat field. You have now a series of images where the sky intensity is expressed in engineering units ADU per second normalized to unit gain. The correspondence between flux, expressed in Jy, and ADU, is contained in the calibration tables CCG*SPEC. More precisely, the tables list the number of ADU to be expected from a point source of flux 1 Jy. The usual IDL Astro Library **fxbopen** and **fxbhelp** operations reveal the contents of, say, CCGLWSPEC:

```
FITS Binary Table:  CCGLWSPEC
Table contains 17 columns, by 202 rows

Col  Name                 Type Size           Units            Null

  1  SPARE                Byt  8
  2  EWHL                 Int  1
  3  SWHL                 Int  1
  4  PFOV                 Int  1
  5  FCVF                 Int  1
  6  WAVELENG             Flt  1              micron
  7  BANDWIDT             Flt  1              micron
  8  TRANSMIS             Flt  1
  9  NORMALIS             Flt  1
 10  SENSITIV             Flt  1              (ADU/sec)
 11  CRPIX1               Flt  1
 12  CRVAL1               Flt  1              micron
 13  CDELT1               Flt  1              micron
 14  BITPIX               Lng  1
 15  NAXIS                Lng  1
 16  NAXIS1               Lng  1
 17  ARRAY                Flt  202
```

The parameter of interest here is `SENSITIV`: number of ADU per second. The current version of CCGLWSPEC has 202 rows; each row corresponds to one position of the filter wheel, or wavelength. To convert ADU in the CISP corrected images to Jy, simply divide each pixel value by the corresponding `SENSITIV` value. It was assumed for the estimation of `SENSITIV` that the spectrum is of the form $\nu I_\nu = const$. Please consult Chapter 6 for more information on photometric correction.

8.6.4 Some useful IDL tips

There is an IDL primitive that allows you to pick out well defined subsets from the CISP data set. Imagine that your observation contains several configurations and different operational modes within a configuration. All data pertaining, say, to a sky observation in configuration 2 can be singled out thus:

Read configuration and operational mode values:

```
IDL> fxbread,u3,obst,'CISPOBST'

IDL> fxbread,u3,cnfg,'CISPCNFG'
```

Single out obst = 1 (sky observation) and cnfg = 2:

```
IDL> skycnf2 = where (obst eq 1 and cnfg eq 1)
```

`skyconf2` is an array containing the CISP indices for all sky observations in configuration 2.

You may use `skycnf2` to verify for instance that the elementary integration time has not changed during the sky observations in configuration number 2. Read all the integration times in CISP:

```
IDL> fxbread,u3,tint,'CISPTINT'
```

Verify that the maximum and minimum values are identical:

```
IDL> print,max(tint(skyconf2))

IDL> print,min(tint(skyconf2))
```

Another example; had this been a raster observation, the corresponding raster ID parameters would be:

```
IDL> fxbread,u3,rpid,'GPSCRPID'

IDL> rpidcnf2 = rpid(*,skycnf2)
```

8.7 Conclusion

This chapter was intended to familiarize the reader with the contents of some ISOCAM products and to introduce some simple IDL based tools to handle those products. It is not the intention of this guided tour to provide a full tutorial on data reduction techniques nor to provide a thorough discussion on instrumental effects.

The Automatic Analysis is designed to perform as good a job as is possible in the framework of a "blind" and automated data reduction session. ISO observers who would like to reduce their ISOCAM data more thoroughly should get in touch with the ISO data Center in VILSPA, Spain or also the national ISO data centers (France, Germany, The Netherlands, United Kingdom or IPAC in the United States). Observers who would like to reduce their ISOCAM data on their own can do so by using the ISOCAM interactive analysis system, CIA [32] by following the instructions given in the CIA Users Manual [17]
`http://www.iso.vilspa.esa.es`

Chapter 9

Data processing Methods

This almost self standing chapter gives a presentation of some algorithms which can be used for processing of ISOCAM data. It should be obvious that this chapter can only be rather incomplete. Our goal here is to motivate astronomers to consider an interactive analysis of their ISOCAM data, best with the ISOCAM interactive analysis system (CIA [30], see:
`http://www.iso.vilspa.esa.es`
and to stimulate the search for their own and possibly better ideas for processing their observations.

The purpose of this chapter is to demonstrate that different processing methods can indeed significantly improve the quality of the final ISOCAM image. Some advantages and limitations of individual methods outlined are discussed. As much as possible we try to apply a realistic model of the instrument. Note that most of these algorithms are implemented in ISOCAM Interactive Analysis (see: "CIA User's Manual" [17].). As the understanding of the calibration improves we will develop and implement new algorithms for the ISOCAM data reduction those will be continuously announced on our web page.

9.1 Notation

For one observation, the ISOCAM instrument delivers a set of frame pairs (start of acquisition (Reset) and end of acquisition (EOI)). For LW, the signal corresponds to a simple difference between EOI and Reset. For SW detector, it is more complex, and several operation such as "cross talk correction" must be done. This first calibration operation is already done when we start with the data in the SPD format. Then we have a set of data noted $D(x, y, t, c)$: we have one measurement per pixel position (x, y) in the detector, which is repeated t times, and we have c configurations. There is a new configuration each time we modify the pointing position, the filter, the integration time, the lens, the gain, etc.

Then, in the ideal case, calibration will consist of :

1. Normalizing the data to ADU/s/Gain by

$$D_1(x, y, t, c) = \frac{D_0(x, y, t, c)}{gain * T_{int} * N_{accu}}$$

   where $gain$ is the electronic gain, $tint$ is the integration time, $Naccu$ is the number of frames already added ($Naccu$ is greater than one only in the accumulation mode, used for $T_{int} = 0.28s$ or in the CAM parallel mode).

2. Subtraction of the dark current ($dark$).

$$D_2(x, y, t, c) = D_1(x, y, t, c) - dark(x, y)$$

The corresponding dark is extracted from CCGLWDMOD or CCG*DARK (see "ISOCAM Dark Current Calibration Report [6]", Section 4.2.3).

3. Search the CCG*DFLT and CCG*OFLT for suitable optical and detector flat–field measurements, respectively $oflat$ and $dflat$ and build from them the composite flat-field, $flat$:

$$flat(x,y) = \frac{oflat(x,y) \times dflat(x,y)}{mean(oflat(10:21,10:21) * dflat(10:21,10:21)}$$

   Note the normalization procedure of ISOCAM flat–fields (see: ISOCAM Flat–Field Calibration Report [7], Section 4.2.4). Care should be taken not to forget to normalize the composite flat–field, e.g. both $oflat$ and $dflat$ are normalized but their product is not.

4. Division by the composite flat–field ($flat$)

$$D_3(x,y,t,c) = \frac{D_2(x,y,t,c)}{flat(x,y)}$$

5. Averaging the values corresponding to the same sky position and the same configuration to get the final image ($Image$) and the corresponding noise image (RMS). Here we consider the noise image (RMS) to include pure but all statistical errors.

$$Image(x,y,c) = mean(D_3(x,y,1..t,c))$$

$$RMS(x,y,c) = sigma(D_3(x,y,1..t,c))$$

6. In case of raster observation, reconstruct the final raster map $Raster(x,y)$ from all images $Image(x,y,c)$ by standard routines as in the IDL Astronomical library (Landsman, 1995 [26]).

In reality one has to consider several problems (Section 7.3):

- after dark correction one still may find same residual dark pattern
- the calibration library flats are not perfect,
- cosmic rays hit the detector,
- transient behavior of the detector. Every time a detector pixel is illuminated successively by a source and the background, as the detector is scanning the sky, the transition between the two flux levels is not instantaneous.
- spacecraft jitter.
- field of view distortion: the resolution of each pixel depends on its position in the detector.
- ghosts (see: "Ghosts in ISOCAM Images [31]".
- CVF position and/or wavelength shifts (see: ISOCAM CVF Calibration Report [8], Section C.3.3.4 and 6).

Studies have been done and continue in order to resolve all these problems, and some solutions have been proposed. We present in the following some of the methods to correct for those effects.

9.2 Cosmic Ray Impact Suppression

Several methods to deglitch the data have been studied. Here we present four algorithms suitable for deglitching ISOCAM data.

9.2.1 Manual Deglitch

A way to deglitch the data consists of looking at all data, and interactively suppress the data where a glitch occurs. This method is obviously robust, but requires to spend a lot of time. However, it can complement automatic procedures which are never perfect.

9.2.2 Temporal deglitch

For the temporal evolution of a pixel in a data cube, (v(i) = cube(x,y,*), where i is an index of the individual exposures), we compute the median value m and the standard deviation σ by a k-sigma clipping method. Values of v(i) which verified: $abs(v(i) - m) > k\sigma$ are considered as a glitch detection. There is the free parameter k, where k = 5 gives usually reasonable results. This automatic method has the advantage of the simplicity, but is not robust at all due to the stabilization problem which biases the mean value. Indeed, we can compare v(i) to the mean value only if the data are well stabilized. Furthermore, glitches introduce gain variation which modify the mean value. This method gives reasonable results in case of bright point sources.

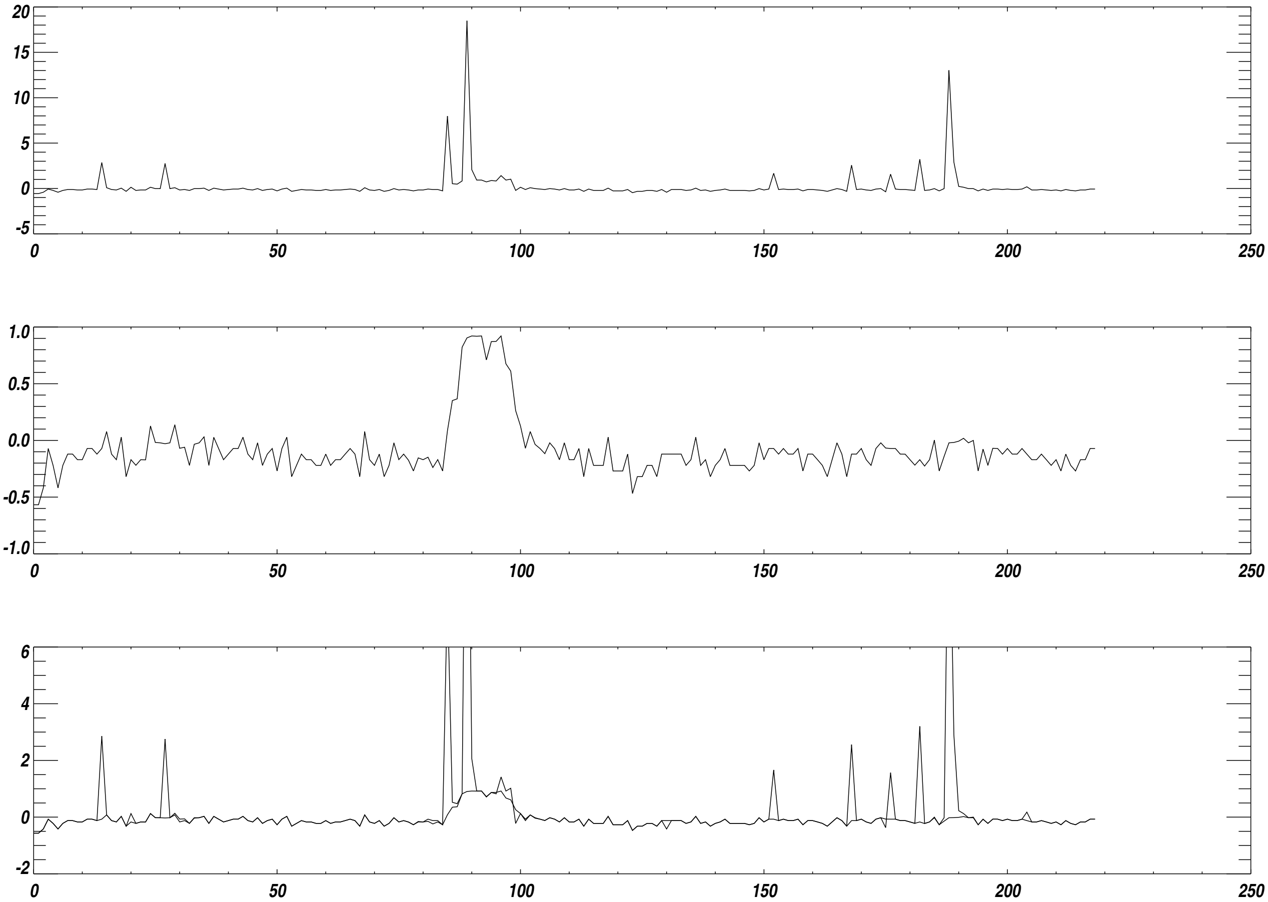

Figure 9.1: Temporal evolution of a pixel in a data cube. we plot the flux in ADU vz. time, given as exposure index. Original data (top), deglitched data (middle), and both over-plotted (bottom). Note the gain variation of about 1 ADU in between frame 85 and 105.

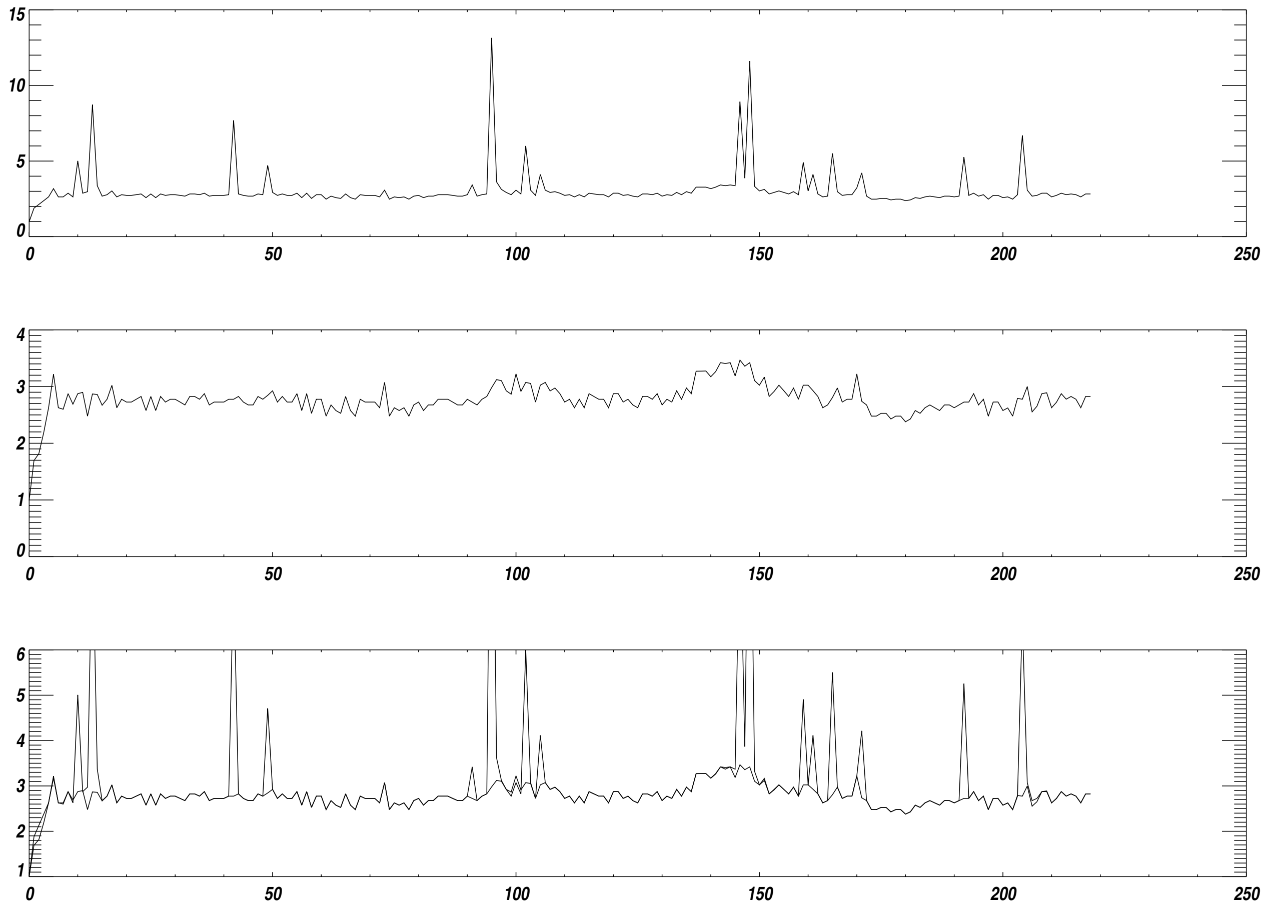

Figure 9.2: Temporal evolution of a pixel in a data cube. We plot the flux in ADU vz. time, given as exposure index. Original data (top), deglitched data Original data (top), deglitched data (middle), and both over-plotted (bottom).

9.2.3 Spatial and temporal Deglitch

The spatial deglitch consists to:

1. calculate first the difference cube: $cubediff(x, y, i) = cube(x, y, i) - cube(x, y, i-1)$
2. calculate for each frame of $cubediff$ the standard deviation $(s(i))$ and the mean $(m(i))$
3. calculate $s = median(s(i))$ and $m = median(m(i))$
4. for all pixels in $cubediff$ we speak of a detection of a glitch, if $cubediff(x, y, i) > m + ks$

This method has the advantage to be relatively efficient when the cube is not stabilized. It uses both spatial and temporal information. But this method does not detect long glitches which remain in several exposures, or when several glitches hit the same pixels in a short duration. It often produces satisfactory results in the case of faint and extended sources such as the zodiacal light.

9.2.4 Median deglitching

A quite robust method is the use of a median filter. Such a median deglitching algorithm is used in the automatic processing system. Some examples of the goodness of this method are given in Section 8.5.1.

9.2.5 Deglitching using Multi-resolution Median Transform

As the glitch structures can have different sizes, we need a multi-resolution tool in order to make an efficient automatic detection. The Multi-resolution Median Transform (MMT) is an alternative to the wavelet transform (Starck et al., 1996). It is a non-linear multi-resolution transform, and is particularly useful every time we have structures with large dynamics. This is the case of the deglitching problem. The idea developed here is the following: as we observe the same position in the sky during n exposures, we cannot have any structure in our signal which has a temporal size lower than $n * tint$. It means that all the significant structures (i.e. not due to the noise) at small scales are due to the glitches. Figures 9.1 and 9.2 show the results after such a treatment. We see that the noise and the signal are not modified during this operation.

Like most current methods the MMT is not perfect but robust and works for non stabilized data. It helps to identify gain variations after a strong or multiple glitch event. It takes more computational time than previous methods (if we omit the manual one). One real limitation is that we cannot detect glitches which are longer than or equal to $n * tint$. That means that the more exposures can be used to deglitch the better will be the result. More details can be found in Starck et al. (1998). The method often fails in case of bright point sources.

9.2.6 Conclusion

Several methods for deglitching of ISOCAM data have been presented here. Among them, deglitching using the MMT is often but not always the most successful approach. Still, in some "special and horrible" cases none of the methods proposed give satisfactory results. Two of such special cases are shown in figure 9.3 and figure 9.4. The latter figure demonstrates a none-recoverable gain variation of order 1 ADU. In the bottom of figure 9.4 one should notice the gain variation due to the strongest glitch. Even worse, the gain variation affects not only the pixel hit by the glitch, but often also the neighboring pixels as well.

A perfect deglitching method would certainly be a method associating the glitches found to a model which allows to correct for the gain variation. For more information on glitches see: "Glitch Effects in ISOCAM Detectors [14]"; "Cosmic Ray effects on the ISOCAM LW detector [19]" and "ISOCAM Glitch Library [15]".

9.3 Dark Subtraction

One has to subtract the dark current from the image, for both the SW and LW channels. The standard procedure is to use LWDMOD file, so to scale a dark image by a parameter which is a function of orbital position and Julia date ("ISOCAM Dark Current Calibration Report [6]". (Section 4.2.3). Such a procedure gives the best results but is not always applicable (.e.g. for $T_{int} = 0.28$s or the SW detector.

Here we compare a FFT dark subtraction method with the dark subtraction using the CCG*DARK. The latter method gives often acceptable results but sometimes fails. The reason is a combination of long-term drifts in the dark current, and low-signal that will make these drifts dominate over the photon and readout noise. This situation is quite easy to recognize as the LW dark current shows strong odd-even stripes (see Figure 9.5), which are not completely gone when the dark correction fails. To provide the reader with orders of magnitudes for these effects, Table 9.1 lists, for all integration times, the spatial mean of the noise on the calibration dark measurement (each dark is the result of the average of a given number of frames, therefore each pixel in the dark measurement has an associated RMS, we report here

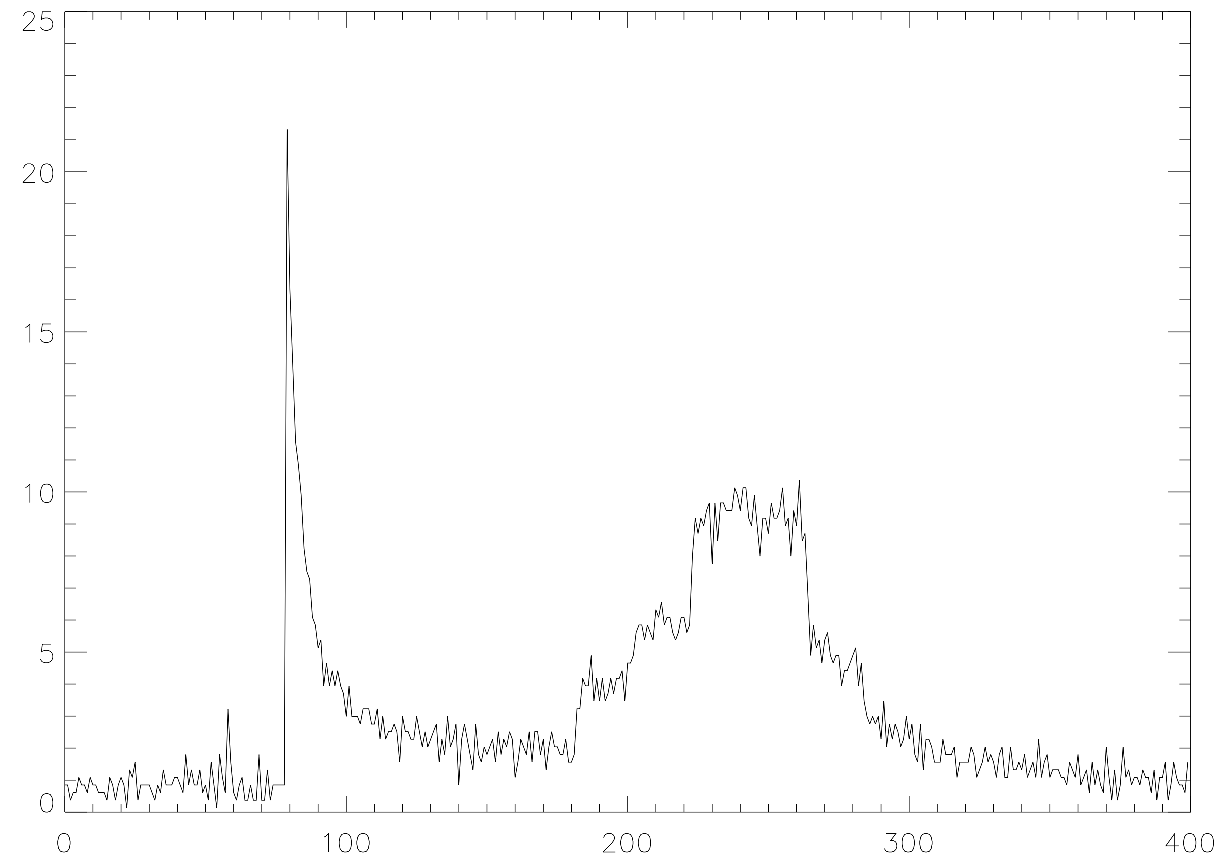

Figure 9.3: Glitch with very long duration. The flux in ADU is plotted against time given by the exposure index.

the mean of these RMS), and, separately for the even and odd lines, the mean, median and rms of the dark level. Finally, this table also lists typical values of residuals (mean and rms) that can be obtained when the calibration dark is used (see also Table 9.2). These values were obtained by correcting very long dark measurements by the calibration dark. These long measurements were performed to derive the time behavior of ISOCAM LW dark [6]. As can be seen from the table, the dark correction is not perfect and there remains a residual whose amplitude is larger than the noise in the calibration dark. Furthermore a clear even-odd pattern remains (see Figure 9.5) as indicated by the relatively large dispersion.

The offset between the calibration dark and the actual dark can only be derived from a study of the time behavior of the dark [6]. The pattern, however, can be removed with appropriate analysis (e.g. Starck and Pantin, 1996).

It was suggested to remove the dark pattern using a linear transformation law (Van Buren and Aussel, 1996 [40]) or a Maximum Entropy Filtering Method (Starck and Pantin, 1996). However those techniques are less robust than the method which removes the pattern in the Fourier space. The algorithm is described in (Starck et al, 1998).

To provide quantitative information on the quality of this correction we performed the following experiment: we used dark measurements performed during "dark orbits [6]" and subtracted the corresponding calibration dark from these measurement. The FFT method was applied to the residuals. In Table 9.2

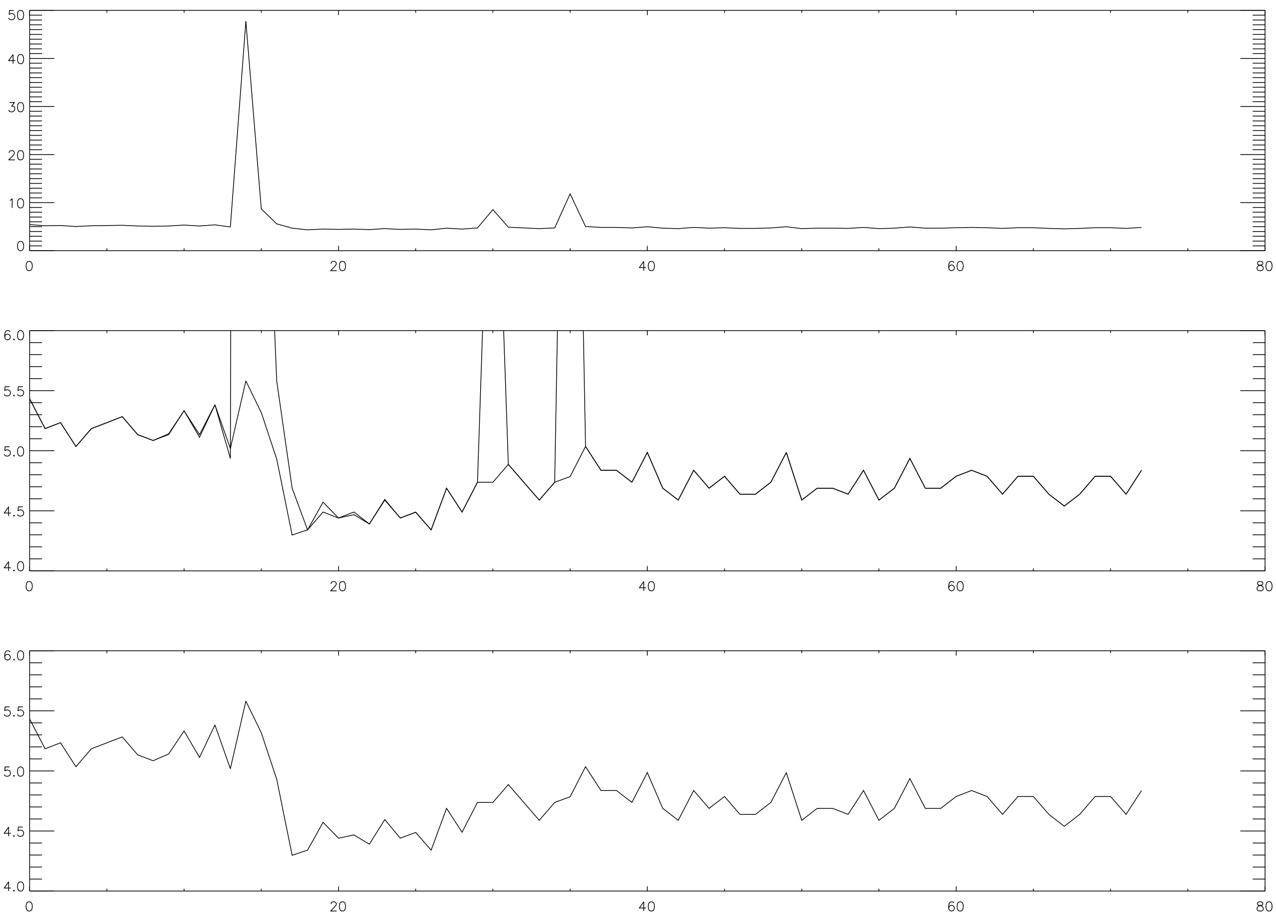

Figure 9.4: Glitch with gain variation. The flux in ADU is plotted against time given by the exposure index. Top, original data, middle both original and deglitched data over-plotted, and bottom, deglitched data. Note the gain variation of about 1 ADU, which remains until the end of the observations.

we give, for the most often used integration times of 2.1, 5.04 and 10.08 s, the mean and rms around the mean for the residuals after calibration dark removal, the mean and rms around the mean for the residuals after the FFT dark correction. As can be seen from the table, the mean value of the residuals is almost unchanged after the FFT dark correction while the rms has been divided by two. More striking is the effect in even-odd pattern. In the last two columns of Table 9.2 we list the difference between the mean of the even lines and the mean of the odd lines for the residuals after calibration dark correction and those after application of the FFT dark correction. After FFT dark correction the remaining difference becomes barely significant.

9.4 Flat Field Correction

The flat field is separated in two in the calibration library: the optical flat and the detector flat. The goal of this separation is to minimize the number of configurations to calibrate. The data must be divided by the two flats. However, the library flat fields rarely correspond exactly to the observer's flat field as there exists a small jitter in the position of the wheels of the instrument. This jitter is intentional and was introduced to prevent the wheels from getting stuck at the extremely low temperature ISO is

T_{int} (s)	σ_N	Even Lines			Odd Lines			Dark mes - Cal	
		Mean	Median	rms	Mean	Median	rms	Mean	rms
	ADU $Gain^{-1}\,s^{-1}$	ADU $Gain^{-1}\,s^{-1}$			ADU $Gain^{-1}\,s^{-1}$			ADU $Gain^{-1}\,s^{-1}$	
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
0.28	0.66	-46.7	-43.5	10.9	-28.4	-26.4	6.0		
2.1	0.16	-10.3	-9.87	1.53	-7.08	-6.83	$8.15\ 10^{-1}$	-0.58	0.27
5.04	$7.0\ 10^{-2}$	-4.87	-4.68	$6.95\ 10^{-1}$	-3.63	-3.53	$3.55\ 10^{-1}$	-0.31	0.16
10.08	$4.6\ 10^{-2}$	-2.57	-2.48	$3.63\ 10^{-1}$	-2.04	-1.98	$1.88\ 10^{-1}$	-0.18	0.07
20.16	$3.6\ 10^{-2}$	-1.25	-1.21	$1.92\ 10^{-1}$	-1.04	-1.02	$1.05\ 10^{-1}$	-0.16	0.05

Table 9.1: Dark levels and associated dispersions for the LW channel of ISOCAM. Column (1) is the integration time in seconds, (2) is the spatial mean of the noise (1σ) in the calibration dark measurement. As the LW dark shows a strong separation between the odd and even lines (due to the different amplification chains), we have listed the levels separately for the even and odd lines. The mean dark value, the median dark value, and the 1σ dispersion around the mean are listed for the even lines (columns 3 to 5) and odd lines (columns 6 to 8). Finally in the last two columns, we have used data obtained during "dark" revolutions (CAM is kept closed for a whole revolution and is continuously read out) to exemplify the amplitude of dark drifts by subtracting the calibration dark from these dark measurements. The spatial mean and rms are listed in column (9) and (10). All dark values are in ADU $Gain^{-1}\,s^{-1}$. No values are listed for the 0.28s integration time since due to technical constraints, it cannot be measured for a complete orbit.

T_{int}	Calibration Dark		FFT dark correction		Mean(even lines) - Mean(odd lines)	
	Mean	rms	Mean	rms	calg	FFT
s	ADU $Gain^{-1}\,s^{-1}$		ADU $Gain^{-1}\,s^{-1}$		ADU $Gain^{-1}\,s^{-1}$	
(1)	(2)	(3)	(4)	(5)	(6)	(7)
2.10	-0.597	0.279	-0.588	0.126	-0.455	$5.31\,10^{-3}$
5.04	-0.302	0.133	-0.300	0.068	-0.207	$2.59\,10^{-3}$
10.08	-0.174	$6.86\,10^{-1}$	-0.173	$3.60\,10^{-1}$	-0.105	$1.17\,10^{-3}$

Table 9.2: Quantitative information on the performances of the FFT dark subtraction method. This table compares the quality of dark current correction between the CCG*DARK method, i.e. using a library dark, and the standard+FFT method, where the subtraction of a library dark is followed by FFT filtering. To make this comparison, we are using dark measurements (typically between 10 and 20 per integration times) obtained during "dark orbits [6]". Column (1) gives the integration time in seconds. We only display results for the most commonly used integration times. Columns (2) and (3) list the spatial mean and rms on the library dark corrected images. Columns (4) and (5) give the same information once these images have been FFT filtered. One can see that the mean signal is little affected (<2%) while the rms is divided by ~2. In column (6) and (7) we compare the even and odd lines of the images at the two stages of dark correction by computing the difference between the mean of the even lines and the mean of the odd lines. One can see that while this difference is quite significant after only the library dark correction, it is almost insignificant after FFT filtering.

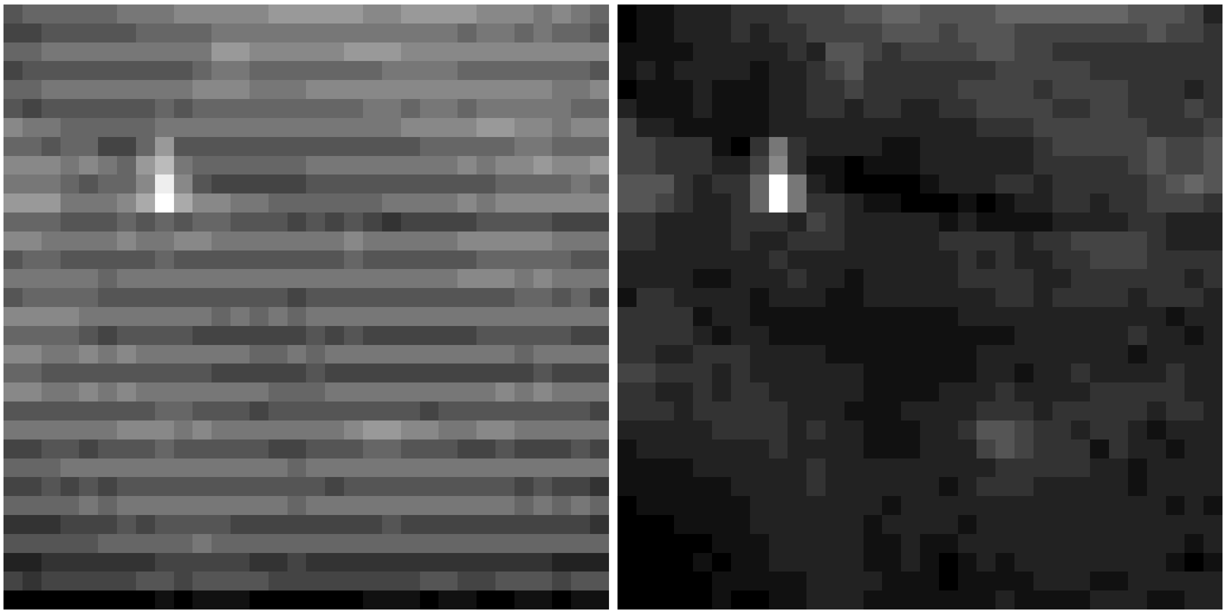

Figure 9.5: Image before dark subtraction (left) and after (right). Notice the dark pattern, which is visible as a change of the signal by comparing odd and even lines in the original image.

operated. This means that for two observations with the same configuration, the optical flat field can be significantly different. Different methods have been tested in order to derive a flat from the data, but they apply only in the case of a raster observation.

9.4.1 Library Flats

The flat field is separated in two flat fields in the calibration library: the optical flat and the detector flat. The data have to be divided by the multiplication of the two flats. In some cases, particularly for observations using a raster mode on faint sources embedded in a flat extended emission, e.g the zodiacal light, a flat field can be derived from the data by choosing interactively a region where there is no object, and only background. For an automatic flat field estimation, we take advantage of the fact that, in such observations, all pixels of the detector spend more time observing the background than the object

9.4.2 Manual Flat Field Estimation

A flat field may be derived from the data, particularly for those in raster mode, just by choosing interactively the region where there is no object, but only background.

9.4.3 Automatic Flat Field Estimation

If we make the assumption that a pixel of the detector sees more time the background than the object, a flat field is easily obtained by the following procedure:

1. Averaging all frames belonging to the same configuration. We obtain one image per configuration, i.e. a cube of averaged data $C_a(x, y, c)$.

2. For each pixel (x, y) of the detector, we take the median of $C_a(x, y, *)$. We get M(x,y).

3. The flat is equal to M normalized by the the mean of its center region: $flat = M/mean(M(10:21, 10:21))$.

This methods works well for long raster observations on strong background fields including only faint sources.

9.5 Iterative Flat Field Estimation for Raster with Overlap

9.5.1 Introduction

In order to improve the estimated flat field, we can use the fact that a sky point is seen by several pixels of the detector. We consider:

- R = number of raster positions
- $a = 32^2$ number of pixels of the detector.
- u = detector overlap ($u = 0$ for no pixel overlap, and $u = 1$ for pixel overlap)
- P = number of pixels of the final image

The number of unknown variables X should be less than the number D of data. We have $X = P + a$ (final image + flat), and the number of data is equal to $D = R \cdot a$. The final image size depends on both the overlap factor and the number of raster position:

$$P = a + a(R-1)(1-u) = a(R - uR + u) \tag{9.1}$$

then $X = a(R - uR + u) + a \leq Ra$ implies that

$$R \geq \frac{1+u}{u} \tag{9.2}$$

or

$$u \geq \frac{1}{R-1} \tag{9.3}$$

These relations give only an idea of the limits, because they do not take into account effects such as transients, field of view distortion, noise, etc. As a rule of thumb one can consider that in practice an increased number of raster positions and pixel overlap will result in a better determination of the system.

9.5.2 Iterative Flat-Field Estimation

We represent the reduced data (without flat field correction) by $image(x, y, c)$, where x, y is the position on the detector, and c the raster position number or configuration number. From $Image$ and from an initial flat $flat_0$, we can build a raster image $R_0(k, l)$.

$$P(Image/flat_0) = R_0 \tag{9.4}$$

From the sky image R_0, we can simulate our data by applying what we call an *inverse projection*. We obtain the set $M(x, y, c)$

$$M(x, y, c) = P^{-1}(R_0(k, l)) \tag{9.5}$$

M can be considered as a model of what we should get on the detector if the *true* sky image is R_0. This model can be used to obtain an estimation $flat_1$ of the flat. This new flat will then lead to a new rater image R_1

$$P(Image/flat_1) = R_1 \tag{9.6}$$

or in a more general way, we have the iterative flat field correction

$$P(Image/flat_{n+1}) = R_{n+1} \tag{9.7}$$

The problem is now to find $flat_{n+1}$ knowing $Image$ and our sky model M. The least square solution gives

$$flat_{n+1}(x, y) = \frac{\sum_c \frac{Image(x,y,c)M(x,y,c)}{RMS(x,y,c)^2}}{\sum_c \frac{M(x,y,c)^2}{RMS(x,y,c)^2}} \tag{9.8}$$

Furthermore, we can assume that the flat is varying with the time. Then the equation 9.7 becomes

$$P(\frac{Image(x, y, c)}{flat_{n+1}(x, y, c)}) = R_{n+1} \tag{9.9}$$

and if we assume a linear variation constant over the whole image, we have

$$P(\frac{Image(x, y, c)}{(\alpha_{n+1}(c)flat_{n+1}(x, y)}) = R_{n+1} \tag{9.10}$$

we have now one unknown more. At each iteration, we compute α_{n+1}, and $flat_{n+1}$ by

$$flat_{n+1}(x, y) = \frac{\sum_c \frac{\alpha_n(c)Image(x,y,c)M(x,y,c)}{RMS(x,y,c)^2}}{\sum_c \frac{\alpha_n(c)M(x,y,c)^2}{RMS(x,y,c)^2}} \tag{9.11}$$

$$\alpha_{n+1}(c) = \frac{\sum_{x,y} M(x, y, c)flat_n(x, y)}{\sum_{x,y} flat_n(x, y)^2} \tag{9.12}$$

at each iteration, the flat and α are normalized

$$\begin{aligned} flat_{n+1}(x, y) &= flat_{n+1}(x, y)/mean(flat_{n+1}(10:21, 21:21)) \\ \alpha_{n+1}(c) &= \alpha_{n+1}(Last\_Config) \end{aligned}$$

9.5.3 Conclusion

Library flat fields can be used to an accuracy similar to pre-launch expectations (see Table 7.1). The automatic flat field method produces generally good result for raster observations. Its limitation is given by the need that a pixel of the detector sees more often a strong background than a source also the object must be faint enough to reduce the transient problem.

For raster observation, the flat field estimation can be improved by an iterative method, using the fact that a sky point is seen by different pixels (Starck et al, 1998).

For more details on flats see "ISOCAM Flat-field Calibration Report [7]".

9.6 Stabilization

The flux transient behavior of the ISOCAM LW detector is strong source of uncertainty, consequently a number of different methods have been suggested to correct for this source of error. The more straight forward procedures are not using any physical model to correct for flux transients. Those methods are looking for the frames (method 1) or the pixels (method 2 and 3) which are 90% stabilized. The next methods described try to fit the transients using a physical model of the (LW) detector. The vision method assumes a given behavior of the signal, and use it for transient suppression. The last method uses a simplified of the CAM-LW response.

9.6.1 Find the first stabilized frame

This method consists in searching the first stabilized frame, and to use only the frames after stabilization. The algorithm is:

1. Compute the mean of the first 10% of frames ($fstart$) and the mean of the last 20% of frames ($fend$), by:

   $fstart = mean(cube(*, *, 0 : istart))$
   and
   $fend = mean(cube\_in(*, *, iend : nframe))$

2. Derive vector (v_{mean}) representing the mean value of each frame in the cube.

3. Estimate when the total array is stabilized by the condition:

   in case $v_{mean}(fstart) > v_{mean}(fend)$:
   $1.1 * v_{mean}(fend) > v_{mean}$,

   and

   in case $v_{mean}(fstart) < v_{mean}(fend)$:
   $0.9 * v_{mean}(fend), v_{mean}$.

This assumes that the whole array is stable or that the whole array is not stable. Hence, the method does not take into account pixel-to-pixel variation in stabilization times. One simply improves this method by considering the mask.

9.6.2 Find the first stabilized pixel at each position without using a model

This method is similar than the previous algorithm, but treats all pixels individually and not the array at once. No model is used to find the 90% stabilization level. An IDL script reads:

```
From = Frame 0 and To = Last frame
Delta = abs(From - To)
for each pixel i,j of the cube
        the pixel is stabilized when
              cube(i,j,s) < To + 0.1 * Delta(i,j)
           and cube(i,j,s) > To - 0.1 * Delta(i,j)
```

9.6.3 Find the first stabilized pixel at each position using a model

This method find pixels which are not stabilized at 90% stabilization level. We apply the transient model as given in the ISOCAM Observer's Manual [24]. for the estimation of the number of stabilized frames. The number N_f of frames we need to reach a stabilization at a given detector position is calculated by:

$$N_f = N_0 * \Delta^{\alpha} / J_{max}^{\beta} \tag{9.13}$$

where N_0,α,β are parameters and J the flux. For LW, parameters we use $N_0 = 275$, $\alpha = 0.00$, $\beta = 0.60$ and for SW $N_0 = 269$, $\alpha = 0.24$, $\beta = 0.67$. We call Δ the absolute value of difference between the start and final flux and the maximum of both values J_{max}. Pixels which are not within the 10% stabilization are suppressed.

9.6.4 Transient Model Fitting with IPAC Model

A transient routine has been developed at IPAC. A description of the IPAC model can be found by Van Buren and Kong 1995 [41]; see also Ganga et al. 1998 [21] for an new IPAC model. In the IPAC model only decreasing flux transients are modeled. The downward transients are given by:

$$J(t) = c_0 + \frac{c_1}{1.0 + c_2 e^{-\frac{t}{c_3}}} \tag{9.14}$$

where c_0, c_1, c_2, c_3 are the coefficients for the non-linear squares fit:

$$\begin{aligned} c_0 &= \beta j_\infty && (9.15) \\ c_1 &= (1-\beta) j_\infty && (9.16) \\ c_2 &= \frac{(j_\infty - j_0)}{j_0 - \beta j_\infty} && (9.17) \\ c_3 &= \tau && (9.18) \end{aligned}$$

and j_0 is the data after transition, j_∞ the last data point, and τ is the time value when $J(\tau) = c_0 + c_1 + c_2 e^{-1}$, and $\beta = 0.6$. Figure 9.6 shows the result of such a treatment on simulated and real data.

For each pixel, we sample a specified number of data points in time sequence and fit them with a mathematical model.

The algorithm is:

1. Detect the starting point of transient behavior within the input data cube.

2. For each pixel, sample a specified number of data points in time sequence and put them in a array.

3. Determine the initial guesses for the least squares fit from the data.

4. Perform a least squares fit. If outliers are detected, deletes them and do the fit once more.

5. Replace the input cube with the fitted data.

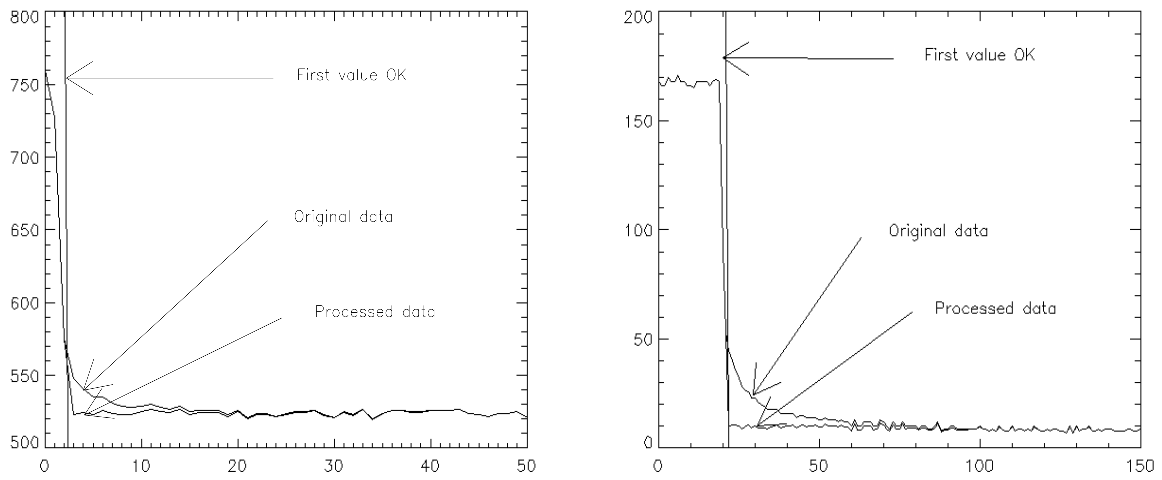

Figure 9.6: Transient modeling on simulated data (left) and real data (right). All values at left of the vertical bar are not used by the next treatments.

9.6.5 Transient Model Fitting with IAS Model

A transient method has been developed at IAS. In the IAS model up and downward transients can be treated. The IAS model is given by:

$$T(t, J_{min}, J_{max}) = J_{min} + \beta \cdot \Delta + 2\Delta \cdot (1-\beta) \cdot (\frac{1}{1+exp(-\frac{t-t_0}{\tau})} - 0.5) \qquad (9.19)$$

with $\Delta = J_{max} - J_{min}$.

The relaxation time or damping constant τ is estimated in order to have a 90% stabilization after N_f frames. N_f is the number of frames we need to reach a stabilization and is calculated by Eq. 9.13.

For upward transient ($J_\infty = J_{max}$, $J_0 = J_{min}$):

$$J_u(t, J_0, J_\infty) = T(t, J_{min}, J_{max}) \qquad (9.20)$$

and for downward transient ($J_\infty = J_{min}$, $J_0 = J_{max}$), we use:

$$J_d(t, J_0, J_\infty) = J_{max} + J_{min} - J_u(t, J_\infty, J_0) \qquad (9.21)$$

9.6.6 Transient Model Fitting with SACLAY Model

This model takes into account the very low transition (in time) of a pixel after a strong exposure. Real data show that this kind of downward transitions can be barely fitted by the previous models. A good example of such a transient is the companion of M51 in the large ISOCAM raster map of this galaxy (see Figure 9.7). Here one can improve by splitting the transient function into two parts:

- a first part where data are quickly decreasing and they can be fitted by an downward exponential and

- a second part where data are decreasing very slowly but which cannot be seen as the tail of the decreasing exponential of the first part.

In the Saclay model a second decreasing exponential with a different relaxation time is introduced:

- For downward transient

$$J_d(t) \;=\; J_\infty + (J_0 - J_\infty).\beta.exp(-\frac{t}{\tau}) + \beta'.exp(-\frac{t}{\tau'})$$

  where $\tau' >> \tau$;

- and for upward transient

$$J_u(t) \;=\; J_\infty - (J_\infty - J_0).\beta.exp(-\frac{t}{\tau})$$

The low decrease of the transient is modeled by a larger time constant τ'. This model has two advantages: it leads to a complete fit of the data (and not only the beginning or the end of the transient), and it also allows us to work with the first points to guess the stabilized value J_∞.

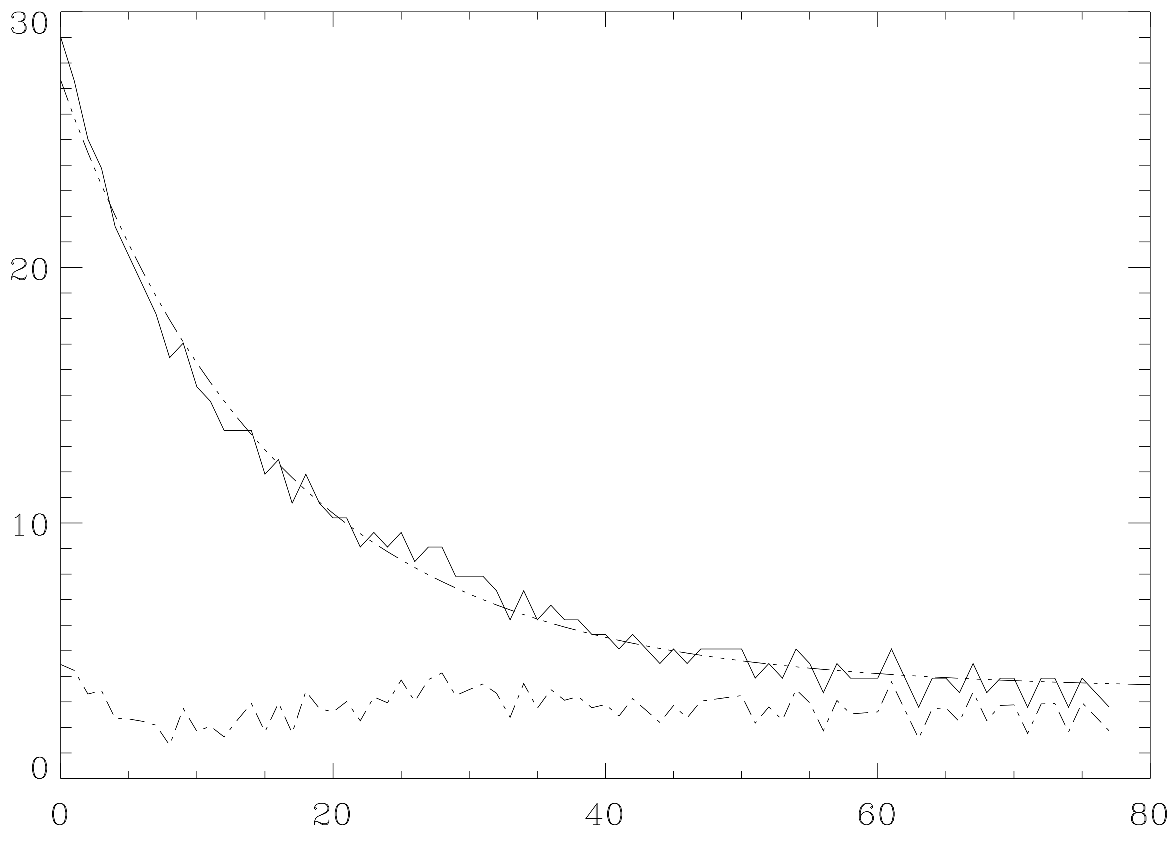

Figure 9.7: Example of transient fit using the SACLAY model.

Figure 9.7 shows the result of a fit, and figure 9.8 shows the raster image of M51 before (left) and after (right) transient correction. It is a raster 10×10, with LW2, an integration time 2.1 seconds, and 40 frames per raster position. The SACLAY model leads to quite good results (no ghosts in the final image whereas there were four ghosts before treating M51).

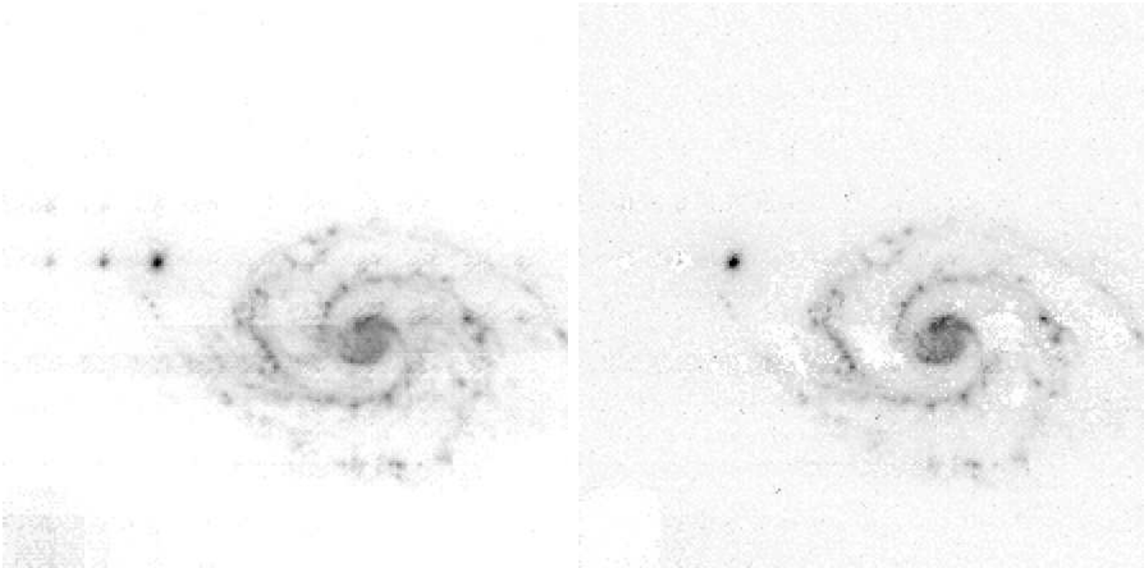

Figure 9.8: M51 without any transient correction (left), and with transient correction using the SACLAY model (right).

9.6.7 Vision Model

Transients correction can be done by trying to understand how the information is distributed in the signal. For example, we can consider that a signal for a given configuration c, and at a detector position (x, y) is

$$D(x, y, 0...t, c) = L(x, y, c) + N(t) \tag{9.22}$$

where $L(c, x, y)$ is the *true* flux we should have observed, which is a constant, and $N(t)$ corresponds to the noise fluctuation. $N(t)$ can be obtained by applying an adaptive filtering to the data (see Section 9.10). The $L(x, y, c)$ estimate is more difficult, and we assume an interpretation of the behavior of the signal, i.e. we fix a vision model.

We call the filtered data F, so that the original data becomes:
$D(x, y, t, c) = F(x, y, t, c) + N(t)$

An example of vision model which would detect a long downward transient is given by:

$$\begin{array}{rl} \text{if} & (F(x, y, First, c) - F(x, y, Last, c)) > k\sigma \\ \text{and} & (F(x, y, First, c+1) - F(x, y, Last, c+1)) > k\sigma \text{ then } L(x, y, c) = L(x, y, c+1) \end{array}$$

with σ as standard deviation of the noise.

Under the assumption that the last frame is the stabilized, an example for a vision model which detect a short downward transient is given by:

$$\begin{array}{rl} \text{if} & (F(x, y, First, c) - F(x, y, Last, c)) > k\sigma \\ \text{and} & (F(x, y, First, c+1) - F(x, y, Last, c+1)) < k\sigma \text{ then } L(x, y, c) = F(x, y, Last, c) - N(Last) \end{array}$$

and for an upward transient:

if $(F(x,y,Last,c) - F(x,y,First,c)) > k\sigma$ then $L(x,y,c) = F(x,y,Last,c) - N(Last)$

These laws are applied for all pixels at each configuration, starting from the last configuration. This allows the treatment of very long downward transients because as long as the transient is detected $L(x,y,c) = L(x,y,c+1)$.

9.6.8 Simplified model of the CAM-LW response

The idea is to describe the response $s(t)$ of one given pixel as a function of the input flux $i(t)$ (in units of ADU $\mathrm{Gain}^{-1}\,\mathrm{s}^{-1}$, i.e. proportional to the incoming flux density, dark subtracted but not flat-field or distortion corrected):

$$s(t) = ri(t) + (1-r)\int_{-\infty}^{t} i(t')\frac{e^{-\frac{t-t'}{\tau}}}{\tau}dt' \tag{9.23}$$

with the time constant τ (in seconds) inversely proportional to the input flux:

$$\tau = \frac{\alpha}{i(t')} labeleq : model2 \tag{9.24}$$

It is easy to verify that, for a step of flux going from i_1 to i_2, this formula gives an instantaneous response equal to to $r \times (i_2 - i_1)$, followed by an exponential variation for the remaining $(1-r) \times (i_2 - i_1)$, with two time constants inversely proportional to i_1 and i_2.

¿From preliminary fitting of the model with the transient responses recorded during a dedicated orbit (revolution 16, during the Performance Verification phase, where cycles of filter sequences were performed), the values of r and α have been adjusted to $r = 0.6$ and $\alpha = 1200$ s (ADU $\mathrm{Gain}^{-1}\,\mathrm{s}^{-1})^{-1}$ (uncertainties are around 10 %). The low signal to noise ratio has not allowed the detection of any significant variations of these values between pixels, though these are likely to occur.

The camera is not read continuously but rather we obtain a discrete series of readouts for all t_j. Thus we can rewrite equation (9.23) in this way:

$$S(t_i) = rI(t_i) + (1-r)\sum_{j=-\infty}^{i-1} int_{t_j}^{t_{j+1}} I(t_j)\frac{e^{-(\frac{t_i - t'}{\tau_j})}}{\tau_j}dt' \tag{9.25}$$

with $\tau_j = \alpha / I(t_j)$.

During an integration, it is assumed (1) that one given pixel sees the same point in the sky and (2) that the configuration of ISOCAM is constant. The effects due to the jitter during the integration are neglected. Thus $I(t)$ is constant between t_j and t_{j+1} and the integral of equation (9.25) can be computed for all the terms in the summation. We have:

$$S(t_i) = rI(t_i) + (1-r)\sum_{j=-\infty}^{i-1} I(t_j)e^{\frac{t_j - t_i}{\tau_j}}(e^{\frac{t_{j+1}-t_j}{\tau_j}} - 1) \tag{9.26}$$

This simplified model allows us to reproduce the response after a step of flux with a precision of typically 5-10%. It fails especially for large flux steps (typically 80 ADU $\mathrm{Gain}^{-1}\,\mathrm{s}^{-1}$): upward transients strongly depart from a simple exponential behavior due to unpredicted oscillations, while downward transients seem to be described more precisely by an hyperbolic curve rather than with an exponential curve. The method is in fact not adapted to measure the brightness of strong point sources, since it neglects the spatial charge coupling, and no methods actually exists allowing this kind of transient to be correctly treated.

The inversion method of the model is described in (Starck et al, 1998).

9.6.9 Conclusion

The three first models suppress the frames or the pixels which do not seem stabilized. However, sometimes no data at all are stabilized during one configuration of the camera. This is the case for the companion of M51 (Figure 9.7). One can see that the transient during the next two configurations lead to a "ghost structure" in the final reconstructed image.

The vision model is a "hard" transient correction, and helps to produce good quality images quickly. The vision method can lead to the loss of information, so must be used with caution.

The quality of fitting methods cannot guarantee that the result is photometrically correct (Starck et al, 1998). We thus caution observers against fitting methods for transient correction as they offer little control on the validity of the resulting photometry.

For the inversion of the simplified CAM-LW response model, one should say that the resulting correction is *not* perfect, since it is limited in precision by the accuracy of the model of the CAM-LW response we use which is obviously oversimplified. In particular, for large flux steps, the memory effect is predicted to be damped too fast. The model of the response we use is approximate. We have already pointed out that upward and downward transients resulting from large flux steps (typically 80 ADU $\mathrm{Gain}^{-1}\,\mathrm{s}^{-1}$) are not symmetrical and depart from a simple exponential behavior, while the model is symmetrical and exponential. The photometric accuracy of the LW channel of ISOCAM is actually of 5-10% after applying the transient correction algorithm we have presented in this section. It is not better for strong sources than for faint sources, because of the over-simplification of the model of the response we use.

None of the transient models produce perfect transient correction, and improvements will certainly be done by using a more realistic CAM-LW response model. It is the direction currently followed by Coulais and Abergel (1998) [16].

9.7 Jitter correction

Spacecraft jitter can cause a slight displacement of a source from IMAGE to IMAGE. This displacement is limited to about $0.5''$ with a standard deviation of less than $0.1''$. The jitter seems to reasonably follow a normal distribution, though there is some evidence that the ISOCAM x-axis jitter, du is anti-correlated to the y-axis jitter, dv. Figure 9.9 shows the distribution of jitter offsets for one particular observation. Jitter is most noticeable when a point source is observed. You can check for jitter in your data inspecting the IMAGEs in your CUBE while watching for sub-pixel shifting of a source.

In general image registration can be applied by optimizing the correlation function between successive frames, however for ISOCAM this approach often fails. We perform jitter correction in a two step approach: Firstly the jitter offsets from IMAGE to IMAGE are computed best by Gaussian or PSF fitting of a point like source in the image and secondly these offsets are applied to the data.

9.7.1 Computing the jitter offsets

It is recommended that the data are at least dark corrected and deglitched before attempting to compute the jitter. In general, the jitter computation is performed as follows:

1. All the IMAGEs corresponding to a single ISO pointing are extracted from the CUBE. These IMAGEs are averaged and the maximum pixel, $[i_{max\_avg}, j_{max\_avg}]$ is taken as a first estimate of the position of the center of the source.

2. A sub-cube is made from the IMAGEs, centered on $CUBE[i_{max\_avg}, j_{max\_avg}, k]$ and of radius *jitter*, where *jitter* is the expected maximum jitter amplitude and defaults to 1 pixel. For *each* frame of this sub-cube, the maximum pixel, $[i_{max}, j_{max}]_k$ is determined.

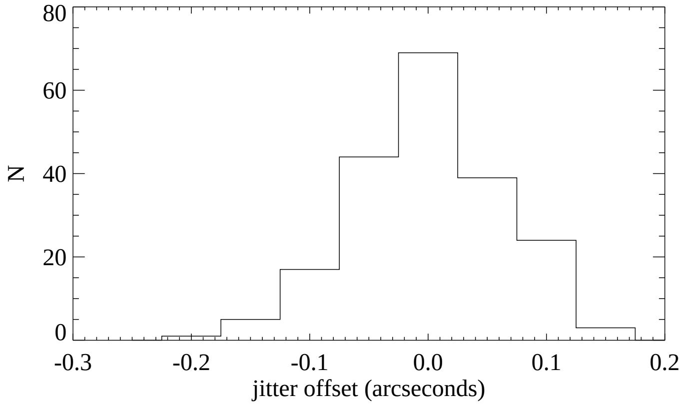

Figure 9.9: Distribution of jitter offsets as computed with the *gauss* method. Data are from a CAM calibration observation of HIC 73005. In this particular case the jitter standard deviation is observed to be about $0.065''$.

3. A new sub-cube, $SUBCUBE$ is again made from the IMAGEs, this time *each* sub-cube frame, $SUBCUBE[i, j, k]$, is centered on $CUBE[[i_{max}]_k, [j_{max}]_k, k]$ and of radius $bsize$.
4. A fit is applied to *each* sub-cube frame $SUBCUBE[i, j, k]$ and the center of the fitted function is taken as the observed source location, $[i_{source}, j_{source}]_k$
5. The jitter offsets in the x and y-axis, du and dv are taken as
   $du_k = [i_{source}]_k - median([i_{source}]_k)$
   $dv_k = [j_{source}]_k - median([j_{source}]_k)$

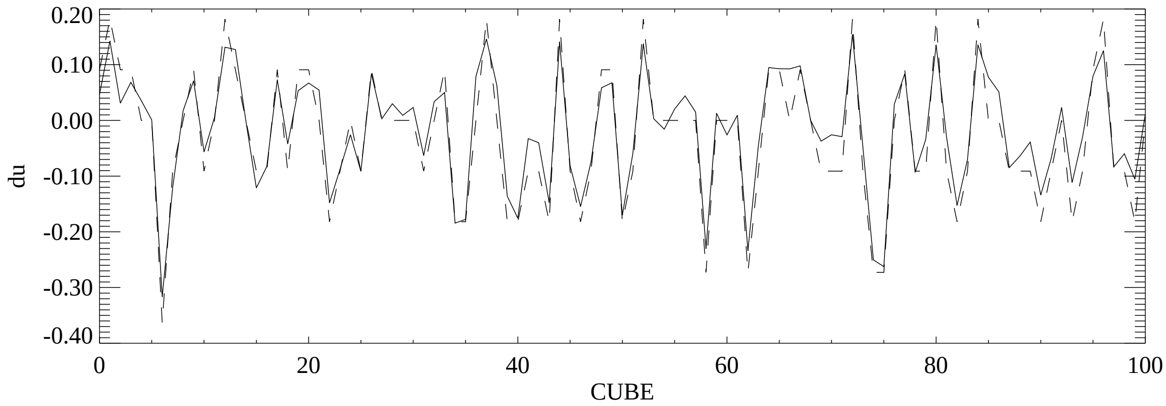

Figure 9.10: Comparison of jitter computation methods. Jitter offsets computed with the *gauss* method are represented by the solid line, and jitter offsets computed with the *psf* method are represented by the dashed line. The data used in this example are from an observation using the $1.5''$ PFOV and the LW7 filter.

Jitter computation is a very interactive process. A Gauss or PSF fitting procedure to a point like source can be used with some success. So far the use of a optimized correlation function has failed.

9.7.2 Applying jitter offsets

The application of the jitter offsets du, dv is the most difficult part of jitter correction. Two methods which have been tested with some success are

- We can attempt to correct the jitter by shifting each IMAGE by du, dv before averaging. The limitation of this method is the shifting algorithm. So far all attempts to perform sub-pixel shifting causes some smoothing and this defeats the purpose of removing the jitter.
- A projection method which attempts to correct jitter by applying du, dv as an astrometric correction. In the example below, the entire CUBE is projected onto a MOSAIC image. In attempting to make a crude correction of the jitter a magnification factor of 3 is applied to IMAGEs before projection.

9.8 Average

All the frames belonging to the same configuration have to be averaged. Bad pixels, or pixels where glitches or transients have been detected should not be taken into account for the average. Then for each pixels and for each configuration of the instrument, we get:

- the mean value of the data cube: $Image(x, y, c)$.
- the standard deviation of the data cube : $RMS(x, y, c)$.
- the number of pixels which where used for computing the mean value: $NPIX(x, y, c)$.

9.9 Field of View Distortion

Field distortion in ISOCAM is mostly due to the off-axis mirror that directs the light beam toward each detector and to the fact that the ISOCAM field of view is an off-axis part of the full FOV of the ISO telescope. The field distortion was measured for the LW channel $6''$ and $3''$ lenses, using calibration observations of fields that contained many stars. No measurements have been made for the $1.5''$ lens because they are difficult to perform as the amplitude of the satellite jitter is of the order of the quantities to be measured. Also the distortion with the $1.5''$ lens seems to be negligible.

ISOCAM also suffers from lens wheel jitter. In order to avoid any mechanical blocking, the gear wheel has been designed with a small play. Therefore, the position at which the lens stops is not fixed.

The field of view distortion was measured in dedicated calibration observations and analysed following the techniques by Holtzman et al. (1995) [22] developed for HST WFPC. For a detailed discussion see: "Field distortion for LW [5]".

9.10 Filtering

In particular for faint source extraction it has been demonstrated that a simple median filtering of a number of about 100 exposures is already quite a good tool to correct for long term drifts. Another step forward is to use an adaptive filtering in the pyramidal median transform or the *à trous* algorithm allows the suppression of the noise without removing the significant structures due to the signal (Starck et al., 1996). For ISOCAM data, the filtering can be done on the final image, but also on the non reduced data. This is done by applying a 1D filtering on each pixel during the whole observation.

Figure 9.11 show the result after an adaptive filtering. Filtering the data for the calibration helps for the transient fit, or the flat building.

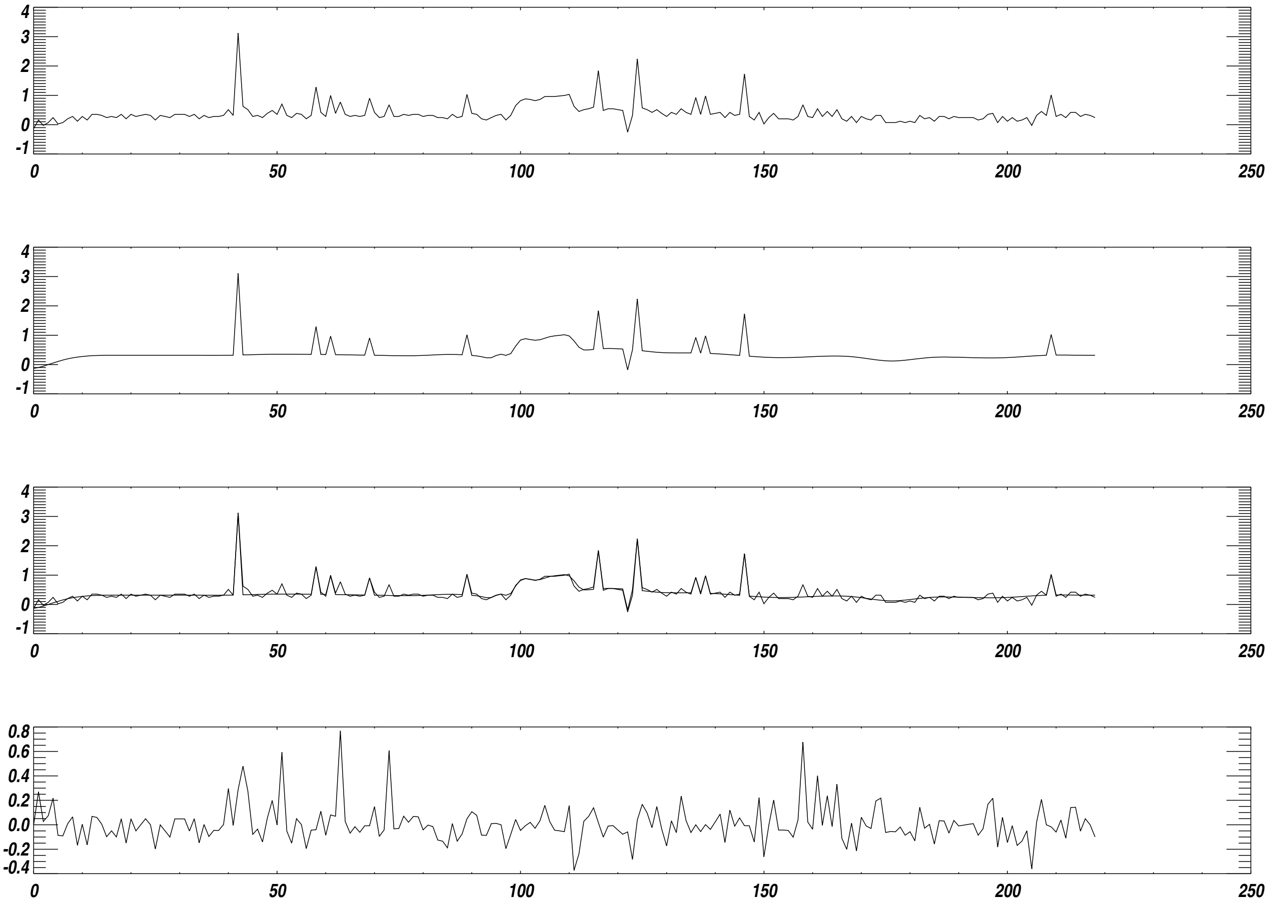

Figure 9.11: From top to bottom, original data, filtered data, original and filtered data overplotted, deglitched and filtered data.

9.11 Deconvolution

Deconvolution has been successfully applied to a rare number of cases only. We give here some general comments: Consider an image characterized by its intensity distribution (the "data") $I(x,y)$, corresponding to the observation of a "real image" $O(x,y)$ through an optical system. If the imaging system is linear and shift-invariant, the relation between the object and the image in the same coordinate frame is a convolution:

$$I(x,y) = (O * P)(x,y) + N(x,y) \tag{9.27}$$

$P(x,y)$ is the point spread function (PSF) of the imaging system, and $N(x,y)$ is an additive noise. In Fourier space we have:

$$\hat{I}(u,v) = \hat{O}(u,v)\hat{P}(u,v) + \hat{N}(u,v) \tag{9.28}$$

We want to determine $O(x,y)$ knowing $I(x,y)$ and $P(x,y)$. This inverse problem has led to a large amount of work, the main difficulties being the existence of: (i) a cut-off frequency of the PSF, and (ii) the additive noise.

Equation 9.27 is always an ill-posed problem. This means that there is not a unique least-squares solution of minimal norm $\| I(x,y) - P(x,y) * O(x,y) \|^2$. Some constraints need to be added in order to have an unique solution. Different approaches can be used to regularized the problem. The wavelet transform is well appropriate to express these constants (Starck et al., 1996; Pantin and Starck, 1996). These deconvolution methods are include in the multi-resolution package (**MR/1** software, contact: murtagh@ulst.ac.uk).
Warning: The deconvolution of ISOCAM images is not only a matter of using an appropriate deconvolution method. It seems more important to ensure that all spurious features which could effect the quality of the image are removed. Potential disturbing features could be residual of glitches and flux transients or jitter. Furthermore the PSF must be of sufficient quality as well. For caveats on the quality of ISOCAM PSF we refer to the ISOCAM PSF Report [30].

9.12 Object detection

Once the data are reduced, we may want to extract the astronomical objects. One detection procedure is to extract the object from its wavelet coefficients (Bijaoui and Rué, 1995; Starck et al, 1997). The algorithm consists of detecting and connecting the significant wavelet coefficients belonging to the same object. Then the problem of reconstruction consists of seeking an image $\tilde{I}$ such that its wavelet coefficients are the same as those of the detected object. If $\mathcal{T}$ describes the wavelet transform operator, and P_w the projection operator in the subspace of the detected coefficients (i.e. all coefficients set to zero at scales and positions where nothing was detected), the solution is found by minimization of

$$J(\tilde{I}) = \| W - (P_w \circ \mathcal{T}) \tilde{I} \| \tag{9.29}$$

where W represents the detected wavelet coefficients of the object. A complete description of algorithms for minimization of such a functional can be found in (Starck, Murtagh and Bijaoui, 1998).

Acknowledgment

The authors would like to thank the overall ISOCAM Team for the continuously work over the last decade making the success of the ISO mission possible. We are grateful for comments and various suggestions given by Prof. Harm Habing.

Bibliography

[1] Abergel, et al., Transient Behaviour of LW Channel of ISOCAM, 1998.

[2] Acosta, Comparison of ISOPHOT and ISOCAM/IAS approaches for transient, 1998.

[3] Altieri, et al., CAM Calibration Explanatory document, 1998.

[4] Altieri, et al., ISOCAM Faint Source Report, 1998.

[5] Aussel, Field distortion for LW, 1998.

[6] Biviano, et al., ISOCAM Dark Current Calibration Report, 1998.

[7] Biviano, et al., ISOCAM Flat-field Calibration Report, 1998.

[8] Biviano, et al., ISOCAM CVF Calibration Report, 1998.

[9] Biviano, ISOCAM Calibration Error Budget Report, 1998.

[10] Blommaert, ISOCAM Photometry Report, 1998.

[11] Boulade & Gallais, The ISOCAM Detectors: An Overview, 1998.

[12] Bijaoui, J.L. Starck, and F. Murtagh, Restauration des Images Multi-Echelles par l'Algorithme à Trous, Traitement du Signal, 3, 11, 1994.

[13] Bijaoui, A., Rué F., 1995, A multiscale vision model adapted to the astronomical images, Signal Processing, 46, pp 345–362.

[14] Claret, et al., Glitch Effects in ISOCAM Detectors, 1998.

[15] Claret and Dzitko, ISOCAM Glitch Library 1998.

[16] Coulais, Abergel, Transient correction of the ISOCAM data with the Fouks-Schubert model: first results, IAS Technical Report, 1998.

[17] Delaney, et al., CIA User's Manual, 1998.

[18] Delattre, About Transients, technical report, CEA, SAP Saclay, 1996.

[19] Dzitko, et al., Cosmic Ray effects on the ISOCAM LW detector, 1998.

[20] Gallais & Boulade, Report on Trend Analysis of CAM daily

[21] Ganga, Kong, Van Buren, Transient Removal with the IPAC method, IPAC, 1998.

[22] Holtzman, J.A., Burrows, C.J., Casertano, S., Hester, J.J, Trauger, J.T., Watson, A.M., and Worthey, G., 1995, PASP, 107, 156.

[23] ISO Architectural Design Document, 1994, ESA document ISO–SSD–9105.

[24] ISOCAM Observer's Manual, The ISOCAM Team, 1994.

[25] ISO Data Product Document, 1996, ESA documnet: ISO-SSD-9111A.

[26] Landsman, W.B., 1995, The IDL Astronomy User's Library, in: Astronomical Data Analysis Software and Systems IV, ed. R.A. Shaw, H.E. Payne, J.J.E. Hayes, ASP Conference Series Vol. 77, p. 437.

[27] Leech, ISO Data User Manual, 1998, ESA document: SAI-95-218-Dc.

[28] Moneti, et al., Reference Wavelengths for ISO: CAM and PHOT filters, 1997

[29] NOST, 1983, Definition of the Flexible Image Transport System, NASA/Science Office of Standards and Technology, NOST standard 100-1.0, NASA Goddard Space Flight Center.

[30] Okumura, ISOCAM PSF Report, 1998.

[31] Okumura, Perault, Congval, Ghosts in ISOCAM Images, 1998.

[32] Ott et al., Design and Implementation of CIAthe ISOCAM Interactive Analysis System, ASP Conference Series, Vol. 125, 1997.

[33] E. Pantin and J.L. Starck, Deconvolution of Astronomical Images using the Multiresolution Maximum Entropy Method, Astronomy and Astrophysics, Suppl. Ser., 118, pp 575–585, 1996.

[34] Siebenmorgen et al., Addendum to the ISOCAM Observer's Manual, 1996.

[35] J.L. Starck, F. Murtagh, and A. Bijaoui, Image Processing and Data Analysis: The Multiscale Approach, Cambridge University Press, 1998.

[36] J.L. Starck, et al, ISOCAM Data Processing, to appear in AA.

[37] Starck, Pantin, 1996, Second Order Dark Correction, Technical Report, CEA Saclay.

[38] J.L. Starck, R. Siebenmorgen, and R. Gredel, Spectral Analysis by the Wavelet Transform, Astrophysical Journal, Vol. 482, pp 1011–1020, 1997.

[39] J.L. Starck, F. Murtagh, and R. Gastaud, A New Entropy Measure Based on the Wavelet Transform and Noise Modeling, IEEE Transactions on Circuits and Systems II on Multirate Systems, Filter Banks, Wavelets, and Applications, 45, 8, 1998.

[40] Van Buren, Aussel, ISOCAM-LW dark frames are simply related, Technical Report, 1996.

[41] Van Buren, and Kong, "Infrared Processing and Analysis Center Interoffice Memorandum", technical report, IPAC, 1995.

Appendix A

Example of the content of a CUFF

A.1 Some Explanation

Firstly let us explain some basic characteristics of this file. Remember that it is written by the Auto-Analysis.

A.1.1 Acronyms

A number of acronyms found in the CUFF are listed below:

RATE	ADU/s
SCD	Science ISOCAM dataset. Each one of these corresponds to a single configuration of ISOCAM.
EWHL	Entrance Wheel position
SWHL	Selection Wheel position
FCVF	Filter wheel position
PFOV	Lens wheel position
TINT	Integration time
CRPIXi	Reference pixels (as for the FITS keywords)

A.1.2 Producing an overview of the observation

In the beginning, AAC produces an overview of the observation like:

```
AAC v7.0.13 starting at 11:47:50 on 10-JUL-98
CCD v4.4 calibration data release
+--------+---------+---------+--------+--------+---------+---------+--------+
This is the general summary of ISOCAM(081012,04)
T=4322s C01 [4*4] micro-scan observation of HD97300
RA(J2000)=11 09 50.596 : DEC(J2000)=-76 36 47.21 : ROLL(J2000)=153.22
```

The first three lines indicate the version os the AAC software and the used Cal-G files. The next three lines contain the TDT-number, the length and type of the observation, together with the target name (as given by the observer) and the pointing direction.

A.1.3 Checking the consistency of the data

When AAC checks the consistency of ISO's movements, it produces messages such as:

```
Overruling OTF when final RPID <> [2,1]
No useful data in CSCD081012040103
```

The first message means that the On Target Flag stayed high while the raster-point number changed. AAC discards the data.
The second message indicated that the data taken by ISOCAM is not reduced to an image, because, e.g. no stable pointing was achieved yet or the camera was not yet in the commanded configuration.

A.1.4 Calibration Library Access

Accessing calibration libraries produces messages such as:

```
Extraction of components from CAL-G files
Spec    FCVF    EWHL
Tried:  140.00  308.00
Got:    140.00  308.00
Drkm    TINT
Tried:  15.00
Got:    15.00
Dflt    EWHL    FCVF    SWHL    PFOV    TINT
Tried:  308.00  140.00  220.00  192.00  15.00
Got:    308.00  125.00  220.00  192.00  36.00
Oflt    PFOV    EWHL    FCVF    SWHL    TINT
Tried:  192.00  308.00  140.00  220.00  15.00
Got:    192.00  308.00  125.00  220.00  36.00
Spec    FCVF    EWHL
Tried:  245.00  308.00
Got:    245.00  308.00
Psf     PFOV    FCVF    CRPIX1  CRPIX2  EWHL    SWHL    TINT
Tried:  192.00  140.00  16.00   16.00   308.00  220.00  15.00
Got:    192.00  140.00  17.00   14.00   308.00  220.00  2.00
```

Three lines are produced for each calibration library access. The first line contains first the name of the library (stripped of its prefix, in the first case here, 'Spec', the spectral characteristics), then the parameters it was attempting to match, in order of precedence. The second line contains the values it was attempting to match, i.e. the values in the SCD. The third line contains the values actually chosen and used from the calibration library.

A.1.5 SCD Summary

A summary such as the following is generated for each SCD:

```
LW(OBS)  06/02/96 14:39:03 to 06/02/96 14:42:52 CSCD081012040106
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
14.0-15.8 micron @ 1.5" per pixel
CCIM(16:18) 109 planes
<RATE>=-8.2ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=228.9s
CGLL 551 glitches
CJAM maximum jitter [-0.5:+0.2,-0.3:+0.2] pixels
```

The first line contains the detector (here `LW`), the observation mode (which can be `OBS` = Observation, `DRK` = Dark, `FLT` = Flat–field, `CLN` = Clean, `IDL` = Idle), the time range and an internal SCD identifier. The second line shows the attitude of the detector. The third line has the wavelength range and the pixel-field-of-view size. The fourth and fifth lines summarise the CCIM entries: firstly the range of rows in the product referred to (here 16 to 18) and the number of planes averaged to construct them, then the calculated rate and statistical error (ADU/s) and the average exposure (in seconds). The sixth line gives the number of glitches detected in pixels and the last the maximum calculated jitter in pixels.

A.1.6 Map summary (CMAP or CMOS)

A CMAP or CMOS summary looks like this:

```
CMAP(1:3) 06/02/96 14:39:03 to 06/02/96 14:42:52
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
14.0-15.8 micron LW RPID=[1,1] image of [32*32] [1.5"] pixels
<FLUX>=2.89mJy/arcsec^2  <FLUX_ERR>=0.21mJy/arcsec^2  <EXPOSURE>=69.0s
CCIM(OBS:DRK:FLT)=(16:0:0)
```

The first line shows the name of the product, the range of rows in the product it refers to (so in this case the first 3 of CMAP, these three rows corresponding to the flux image, the RMS image and the exposure image). The second line contains the ISOCAM attitude at the center of the map. The third has the wavelength, raster position and image and pixel size. The fourth line indicates the mean flux and error (in $mJy/arcsec^2$) and the exposure (in seconds). The final line (present for a CMAP summary only) refers to the row numbers in the CCIM product of the observation measurement from which the CMAP was derived, as well as the related dark and flat–field measurements if applicable.

A.1.7 Point Source Summary

A point-source summary is produced for each CMAP.

```
ISOC110948.3-763649 11 09 48.313 -76 36 49.1    168.9 +/- 7.0   mJy
ISOC110950.6-763641 11 09 50.687 -76 36 41.0    277.6 +/- 8.4   mJy
ISOC110946.6-763640 11 09 46.631 -76 36 40.8     91.6 +/- 5.1   mJy
```

The first line gives the range of rows referred to in the CPSL product and the time range. The second shows the wavelength range. Following this is a summary for each row in the CPSL (and therefore each point source detection). These depict the source name (automatically generated), the position in RA and DEC, the flux and (statistical) error in mJy.

If there is a wavelength range then the CSSP is also summarized in the same vein, with an entry for each point source. A (self-explanatory) entry looks like this:

```
ISOC110946.8-763640 5.4-15.8 micron
```

A.2 A complete example

The example shown here is a real but longer CUFF cut down to size.

```
AAC v7.0.13 starting at 11:47:50 on 10-JUL-98
CCD v4.4 calibration data release
+--------+---------+---------+--------+--------+---------+--------+--------+
```

```
This is the general summary of ISOCAM(081012,04)
T=4322s C01 [4*4] micro-scan observation of HD97300
RA(J2000)=11 09 50.596 : DEC(J2000)=-76 36 47.21 : ROLL(J2000)=153.22
Overruling OTF when final RPID <> [1,1]
Overruling OTF when final RPID <> [2,1]
No useful data in CSCD081012040105
+--------+---------+---------+--------+--------+---------+--------+--------+
Extraction of components from CAL-G files
Spec    FCVF    EWHL
Tried:  140.00  308.00
Got:    140.00  308.00
Drkm    TINT
Tried:  15.00
Got:    15.00
Dflt    EWHL    FCVF    SWHL    PFOV    TINT
Tried:  308.00  140.00  220.00  192.00  15.00
Got:    308.00  125.00  220.00  192.00  36.00
Oflt    PFOV    EWHL    FCVF    SWHL    TINT
Tried:  192.00  308.00  140.00  220.00  15.00
Got:    192.00  308.00  125.00  220.00  36.00
Spec    FCVF    EWHL
Tried:  245.00  308.00
Got:    245.00  308.00
Psf     PFOV    FCVF    CRPIX1  CRPIX2  EWHL    SWHL    TINT
Tried:  192.00  305.00  16.00   16.00   308.00  220.00  15.00
Got:    192.00  305.00  19.00   15.00   308.00  220.00  2.00
+--------+---------+---------+--------+--------+---------+--------+--------+

LW(IDL)  06/02/96 14:38:28 to 06/02/96 14:38:53 CSCD081012040101
5.4-6.6 micron
CCIM(1:3) 1 plane
<RATE>=-3.4ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=25.2s
CGLL 800 glitches

LW(IDL)  06/02/96 14:38:53 to 06/02/96 14:38:55 CSCD081012040102
5.4-6.6 micron
CCIM(4:6) 1 plane
<RATE>=-5.7ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=2.1s
CGLL 3 glitches

LW(OBS)  06/02/96 14:38:57 to 06/02/96 14:38:59 CSCD081012040103
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
5.4-6.6 micron @ 6" per pixel
CCIM(7:9) 1 plane
<RATE>=-6.6ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=2.1s
CGLL 0 glitches
CJAM maximum jitter [+0.0:+0.0,+0.0:+0.0] pixels

LW(OBS)  06/02/96 14:38:59 to 06/02/96 14:39:01 CSCD081012040104
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
5.4-6.6 micron @ 6" per pixel
CCIM(10:12) 1 plane
```

```
<RATE>=-10.9ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=2.1s
CGLL 0 glitches
CJAM maximum jitter [+0.0:+0.0,+0.0:+0.0] pixels

LW(OBS)  06/02/96 14:39:01 to 06/02/96 14:39:03 CSCD081012040105
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
5.4-6.6 micron @ 1.5" per pixel
CCIM(13:15) 1 plane
<RATE>=11.1ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=2.1s
CGLL 248 glitches
CJAM maximum jitter [+0.0:+0.0,+0.1:+0.1] pixels

LW(OBS)  06/02/96 14:39:03 to 06/02/96 14:42:52 CSCD081012040106
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
14.0-15.8 micron @ 1.5" per pixel
CCIM(16:18) 109 planes
<RATE>=-8.2ADU/s  <RATE_ERR>=0.0ADU/s  <EXPOSURE>=228.9s
CGLL 551 glitches
CJAM maximum jitter [-0.5:+0.2,-0.3:+0.2] pixels
+--------+---------+---------+--------+--------+---------+--------+--------+

CMAP(1:3) 06/02/96 14:39:03 to 06/02/96 14:42:52
RA(J2000)=11 09 48.464 : DEC(J2000)=-76 36 39.62 : ROLL(J2000)=153.23
14.0-15.8 micron LW RPID=[1,1] image of [32*32] [1.5"] pixels
<FLUX>=2.89mJy/arcsec^2  <FLUX_ERR>=0.21mJy/arcsec^2  <EXPOSURE>=69.0s
CCIM(OBS:DRK:FLT)=(16:0:0)

CMAP(4:6) 06/02/96 14:42:52 to 06/02/96 14:43:49
RA(J2000)=11 09 49.903 : DEC(J2000)=-76 36 39.68 : ROLL(J2000)=153.22
14.0-15.8 micron LW RPID=[2,1] image of [32*32] [1.5"] pixels
<FLUX>=2.91mJy/arcsec^2  <FLUX_ERR>=0.21mJy/arcsec^2  <EXPOSURE>=52.3s
CCIM(OBS:DRK:FLT)=(19:0:0)

CMAP(7:9) 06/02/96 14:43:49 to 06/02/96 14:44:46
RA(J2000)=11 09 51.343 : DEC(J2000)=-76 36 39.74 : ROLL(J2000)=153.22
14.0-15.8 micron LW RPID=[3,1] image of [32*32] [1.5"] pixels
<FLUX>=2.89mJy/arcsec^2  <FLUX_ERR>=0.22mJy/arcsec^2  <EXPOSURE>=52.2s
CCIM(OBS:DRK:FLT)=(22:0:0)
+--------+---------+---------+--------+--------+---------+--------+--------+

CMOS(1:3)     06/02/96 14:39:03 to 06/02/96 14:57:01
RA(J2000)=11 09 50.596 : DEC(J2000)=-76 36 47.21 : ROLL(J2000)=153.22
14.0-15.8 micron LW image of [46*46] [1.5"] pixels
<FLUX>=2.23mJy/arcsec^2  <FLUX_ERR>=0.07mJy/arcsec^2  <EXPOSURE>=392.0s

MOS(4:6)     06/02/96 14:57:49 to 06/02/96 15:14:18
RA(J2000)=11 09 50.601 : DEC(J2000)=-76 36 47.19 : ROLL(J2000)=153.23
10.7-11.9 micron LW image of [46*46] [1.5"] pixels
<FLUX>=2.58mJy/arcsec^2  <FLUX_ERR>=0.06mJy/arcsec^2  <EXPOSURE>=424.9s
+--------+---------+---------+--------+--------+---------+--------+--------+

CPSL(1:3) 14:39:03 06/02/96 to 14:42:52 06/02/96
```

```
14.0-15.8 micron
ISOC110948.3-763649 11 09 48.313 -76 36 49.1    168.9 +/- 7.0   mJy
ISOC110950.6-763641 11 09 50.687 -76 36 41.0    277.6 +/- 8.4   mJy
ISOC110946.6-763640 11 09 46.631 -76 36 40.8     91.6 +/- 5.1   mJy

CPSL(4) 14:42:52 06/02/96 to 14:43:49 06/02/96
14.0-15.8 micron
ISOC110951.0-763640 11 09 51.031 -76 36 40.2    250.8 +/- 8.7   mJy
+--------+---------+---------+--------+--------+---------+---------+--------+

CSSP(1:67)
ISOC110948.3-763649 5.4-15.8 micron
ISOC110950.4-763642 5.4-15.8 micron

+--------+---------+---------+--------+--------+---------+---------+--------+
AAC v7.0.13 finishing at 12:03:48 on 10-JUL-98
```

Appendix B

Data products sorted by AOT

B.1 List of AOT1 specific products

CCSH	ISOCAM Continuous Compact Status
CDER	ISOCAM Diagnostic Edited Raw Data
CIER	ISOCAM Image ERD
CSTA	ISOCAM Compact Status
CISP	ISOCAM Standard Processed Data
CCIM	ISOCAM Images in detector coordinates
CGLL	ISOCAM Glitch List
CJAM	ISOCAM Jitter, Memory and Stabilization information
CMAP	ISOCAM Celestial IR image
CMOS	ISOCAM Celestial IR mosaic
CPSL	ISOCAM Point-Source List
CSSP	ISOCAM Source Spectra
CUFF	ISOCAM User-Friendly log File
CISR	ISOCAM Instrument Station Report
COLR	ISOCAM CAM Off-Line Processing Report
CWHEELS	ISOCAM Wheel information table
CSCGCROSS	ISOCAM SW noise/cross talk decorrelation matrices
CHCGCONV	ISOCAM House keeping interpolation values
CCG*DARK	ISOCAM *W Dark Current Exposures
CCG*DEAD	ISOCAM *W Dead Pixel Map
CCG*DFLT	ISOCAM *W Detector Flat Field Library
CCG*OFLT	ISOCAM *W Optical Flat Field Library
CCG*FRAME	ISOCAM *W Detector astrometric calibration: Currently incondite
CCG*GLITCH	ISOCAM *W Glitch Model: Currently empty
CCG*LINEAR	ISOCAM *W Linearity Correction Library
CLWCVF1	ISOCAM LW CVF segment 1 description table
CLWCVF2	ISOCAM CAM LW CVF segment 2 description table
CCGLWDMOD	ISOCAM CAM LW Dark Current Model Library
CCG*PSF	ISOCAM *W Point-Spread-Function Library

CCGLWSHIFT	ISOCAM LW astrometric shift correction
CCG*SPEC	ISOCAM *W Filter & CVF spectral data
CCG*STRAY	ISOCAM *W Non-dark local light model
CSWCVF	ISOCAM SW CVF description table
CCG*TRANS	ISOCAM *W Model transients: Currently empty

(*=S, L)

B.2 List of AOT3 specific products

CCSH	ISOCAM Continuous Compact Status
CDER	ISOCAM Diagnostic Edited Raw Data
CIER	ISOCAM Image ERD
CSTA	ISOCAM Compact Status
CISP	ISOCAM Standard Processed Data
CCIM	ISOCAM Images in detector coordinates
CGLL	ISOCAM Glitch List
CJAM	ISOCAM Jitter, Memory and Stabilization information
CMAP	ISOCAM Celestial IR image
CMOS	ISOCAM Celestial IR mosaic
CPSL	ISOCAM Point-Source List
CSSP	ISOCAM Source Spectra
CUFF	ISOCAM User-Friendly log File
CISR	ISOCAM Instrument Station Report
COLR	ISOCAM CAM Off-Line Processing Report
CWHEELS	ISOCAM Wheel information table
CSCGCROSS	ISOCAM SW noise/cross talk decorrelation matrices
CHCGCONV	ISOCAM House keeping interpolation values
CCG*DARK	ISOCAM *W Dark Current Exposures
CCG*DEAD	ISOCAM *W Dead Pixel Map
CCG*DFLT	ISOCAM *W Detector Flat Field Library
CCG*OFLT	ISOCAM *W Optical Flat Field Library
CCG*FRAME	ISOCAM *W Detector astrometric calibration: Currently incondite
CCG*GLITCH	ISOCAM *W Glitch Model: Currently empty
CCG*LINEAR	ISOCAM *W Linearity Correction Library
CLWCVF1	ISOCAM LW CVF segment 1 description table
CLWCVF2	ISOCAM LW CVF segment 2 description table
CCGLWDMOD	ISOCAM CAM LW Dark Current Model Library
CCG*PSF	ISOCAM *W Point-Spread-Function Library
CCGLWSHIFT	ISOCAM LW astrometric shift correction
CCG*SPEC	ISOCAM *W Filter & CVF spectral data
CCG*STRAY	ISOCAM *W Non-dark local light model
CSWCVF	ISOCAM SW CVF description table
CCG*TRANS	ISOCAM *W Model transients: Currently empty

(*=S, L)

B.3 List of AOT4 specific products

CCSH	ISOCAM Continuous Compact Status
CDER	ISOCAM Diagnostic Edited Raw Data
CIER	ISOCAM Image ERD
CSTA	ISOCAM Compact Status
CISP	ISOCAM Standard Processed Data
CCIM	ISOCAM Images in detector coordinates
CGLL	ISOCAM Glitch List
CJAM	ISOCAM Jitter, Memory and Stabilization information
CMAP	ISOCAM Celestial IR image
CPSL	ISOCAM Point-Source List
CSSP	ISOCAM Source Spectra
CUFF	ISOCAM User-Friendly log File
CISR	ISOCAM Instrument Station Report
COLR	ISOCAM CAM Off-Line Processing Report
CWHEELS	ISOCAM Wheel information table
CSCGCROSS	ISOCAM SW noise/cross talk decorrelation matrices
CHCGCONV	ISOCAM House keeping interpolation values
CCG*DARK	ISOCAM *W Dark Current Exposures
CCG*DEAD	ISOCAM *W Dead Pixel Map
CCG*DFLT	ISOCAM *W Detector Flat Field Library
CCG*OFLT	ISOCAM *W Optical Flat Field Library
CCG*FRAME	ISOCAM *W Detector astrometric calibration: Currently incondite
CCG*GLITCH	ISOCAM *W Glitch Model: Currently empty
CCG*LINEAR	ISOCAM *W Linearity Correction Library
CLWCVF1	ISOCAM LW CVF segment 1 description table
CLWCVF2	ISOCAM LW CVF segment 2 description table
CCGLWDMOD	ISOCAM CAM LW Dark Current Model Library
CCG*PSF	ISOCAM *W Point-Spread-Function Library
CCGLWSHIFT	ISOCAM LW astrometric shift correction
CCG*SPEC	ISOCAM *W Filter & CVF spectral data
CCG*STRAY	ISOCAM *W Non-dark local light model
CSWCVF	ISOCAM SW CVF description table
CCG*TRANS	ISOCAM *W Model transients: Currently empty

(*=S, L)

B.4 List of AOT5 specific products

CCSH	ISOCAM Continuous Compact Status
CDER	ISOCAM Diagnostic Edited Raw Data
CIER	ISOCAM Image ERD
CSTA	ISOCAM Compact Status
CISP	ISOCAM Standard Processed Data
CCIM	ISOCAM Images in detector coordinates
CGLL	ISOCAM Glitch List
CJAM	ISOCAM Jitter, Memory and Stabilization information
CMAP	ISOCAM Celestial IR image
CMOS	ISOCAM Celestial IR mosaic
CPSL	ISOCAM Point-Source List
CSSP	ISOCAM Source Spectra
CUFF	ISOCAM User-Friendly log File
CISR	ISOCAM Instrument Station Report
COLR	ISOCAM CAM Off-Line Processing Report
CWHEELS	ISOCAM Wheel information table
CSCGCROSS	ISOCAM SW noise/cross talk decorrelation matrices
CHCGCONV	ISOCAM House keeping interpolation values
CCG*DARK	ISOCAM *W Dark Current Exposures
CCG*DEAD	ISOCAM *W Dead Pixel Map
CCG*DFLT	ISOCAM *W Detector Flat Field Library
CCG*OFLT	ISOCAM *W Optical Flat Field Library
CCG*FRAME	ISOCAM *W Detector astrometric calibration: Currently incondite
CCG*GLITCH	ISOCAM *W Glitch Model: Currently empty
CCG*LINEAR	ISOCAM *W Linearity Correction Library
CLWCVF1	ISOCAM LW CVF segment 1 description table
CLWCVF2	ISOCAM LW CVF segment 2 description table
CCGLWDMOD	ISOCAM CAM LW Dark Current Model Library
CCG*PSF	ISOCAM *W Point-Spread-Function Library
CCGLWSHIFT	ISOCAM LW astrometric shift correction
CCG*SPEC	ISOCAM *W Filter & CVF spectral data
CCG*STRAY	ISOCAM *W Non-dark local light model
CSWCVF	ISOCAM SW CVF description table
CCG*TRANS	ISOCAM *W Model transients: Currently empty

(*=S, L)

Note that AOT5 observations have been up-linked via the calibration up-link system. The AOT name used by this system is 'C99'

B.5 List of specific products for ISOCAM parallel

CCSH	ISOCAM Continuous Compact Status
CDER	ISOCAM Diagnostic Edited Raw Data
CPER	ISOCAM Parallel Image ERD
CSTA	ISOCAM Compact Status
CPSP	ISOCAM Parallel Standard Processed Data
CCIM	ISOCAM Images in detector coordinates
CGLL	ISOCAM Glitch List
CJAM	ISOCAM Jitter, Memory and Stabilization information
CMAP	ISOCAM Celestial IR image
CMOS	ISOCAM Celestial IR mosaic
CPSL	ISOCAM Point-Source List
CSSP	ISOCAM Source Spectra
CUFF	ISOCAM User-Friendly log File
CISR	ISOCAM Instrument Station Report
COLR	ISOCAM CAM Off-Line Processing Report
CWHEELS	ISOCAM Wheel information table
CSCGCROSS	ISOCAM SW noise/cross talk decorrelation matrices
CHCGCONV	ISOCAM House keeping interpolation values
CCG*DARK	ISOCAM *W Dark Current Exposures
CCG*DEAD	ISOCAM *W Dead Pixel Map
CCG*DFLT	ISOCAM *W Detector Flat Field Library
CCG*OFLT	ISOCAM *W Optical Flat Field Library
CCG*FRAME	ISOCAM *W Detector astrometric calibration: Currently incondite
CCG*GLITCH	ISOCAM *W Glitch Model: Currently empty
CCG*LINEAR	ISOCAM *W Linearity Correction Library
CLWCVF1	ISOCAM LW CVF segment 1 description table
CLWCVF2	ISOCAM LW CVF segment 2 description table
CCGLWDMOD	ISOCAM CAM LW Dark Current Model Library
CCG*PSF	ISOCAM *W Point-Spread-Function Library
CCGLWSHIFT	ISOCAM LW astrometric shift correction
CCG*SPEC	ISOCAM *W Filter & CVF spectral data
CCG*STRAY	ISOCAM *W Non-dark local light model
CSWCVF	ISOCAM SW CVF description table
CCG*TRANS	ISOCAM *W Model transients: Currently empty

(*=S, L)

Appendix C

Caveats applicable to OLPv7

C.1 Introduction

Several products, calibration files and documentation are sent to ISO Observers.

At the present time, several caveats are applicable to the contents of those files. The following sections apply only to version **V7.0** of the pipeline software.

Caveats below are given separately for the documentation, for the calibration files; and for the actual data products.

C.2 Calibration Files

For completeness, the full list of calibration files and a one line summary description is given below (note that similar files for LW and SW are named xW):

File Name	Short Description
CCGxWDARK	xW Detector Dark Current Exposure Library
CCGxWDEAD	xW Detector Dead Pixel Map
CCGxWDFLT	xW Detector Flat Field Library
CCGLWDMOD	LW parameters for the Time Dependent Dark correction
CCGxWFRAME	xW Detector astrometric calibration (currently empty)
CCGxWGLITCH	xW Glitch Model (currently empty)
CCGxWLINEAR	xW Linearity Correction Library (currently empty)
CCGxWOFLT	xW Optical Flat Field Library
CCGxWPSF	xW Point-Spread-Function Library
CCGLWSHIFT	LW optical axis shift (currently only for polarizers)
CCGxWSPEC	xW Filter and CVF spectral characteristics
CCGxWSTRAY	xW Non-dark local light model (currently empty)
CCGxWTRANS	xW Model transients for memory calibration (currently empty)
CLWCVF1	LW CVF segment 1 description table
CLWCVF2	LW CVF segment 2 description table
CSWCVF	SW CVF description table
CSCGCROSS	SW Noise/cross talk decorrelation matrices
CHCGCONV	House Keeping interpolation values
CWHEELS	Wheel Information Table

C.3 Products

ISO's off–line processing has three levels of processing, resulting in several intermediate and final products. The description below applies particularly to ISOCAM data[1] (other ISO instruments apply different processing levels before generating intermediate products)

- Raw telemetry to edited telemetry. Intermediate products include: CIERtdtnb [2], CSTAtdtnb, IIPHtdtnb, IRPHtdtnb, etc.
- Edited telemetry to ISOCAM raw images and calibrated engineering parameters. Intermediate product: CISPtdtnb
- Raw images to "astronomical" images and ancillary information. Products: CCIMtdtnb, CMAPt-dtnb, CMOStdtnb, etc.

C.3.1 First processing level

C.3.1.1 CSTA files

There is a known minor problem. Sometimes, CSTA first record indicates that an observation was performed in the parallel mode configuration, yet ISOCAM is the prime instrument (CSTA parameters CSTAFOBS = 3 and CSTACONF = 1). This is a "left–over" from the parallel mode, i.e. ISOCAM's configuration before the start of the current TDT. Usually there is only one readout in this mode and the corresponding CIER and CISP data can well be disregarded.

The CSTA file is not reliable for the "C99" observations, i.e. observations performed with the Calibration Uplink System.

C.3.1.2 CIER files

When CSTA contains a spurious record as indicated in C.3.1.1, CIER's first record(s) will indicate that the detector readouts belongs to CONF = 1. This is a left over from the previous parallel mode and not the first configuration as defined in PGA and the user may safely disregard the corresponding readouts.

C.3.1.3 IIPH files

The pointing stability of the ISO spacecraft is so good that a dominant component of the RA, DEC fluctuations during fixed pointing reported in the IIPH seems to be due to measurement noise rather than reflecting real jitter of the viewing direction. In any case, the amplitude of the apparent fluctuations is less than the smallest CAM pixel. On larger angular scales, the IIPH is reliable for following the course raster and beam–switch observations.

Beware of occasional bumps in the IIPH during stable pointing that have been traced to the effects of cosmic ray glitches in the star trackers CCD. Recognition of events of this type may be helped by the fact that large apparent excursions in the pointing direction are not accompanied by the OTF going down. Beware, also, that drifts could conceivably be caused by a thermally–induced shift between the optical axes of ISO and the Star tracker that would be invisible to the attitude system but could be visible in CAM images. We are not aware that any episodes of this type have occurred but please report back any suspicious data you may find.

The data elements called CRA and CDEC are currently unreliable and should be ignored.

For all observations done during the instrument activation sequence the pointing reported for the prime instrument is incorrect. Also the INSTRUME header keyword and the aperture field is wrongly filled. Instead of the prime instrument it gives QSS and Z0.

[1] Please refer to the ISO Data Products document [25] for explanation of contents and layout of data products

[2] The notation tdtnb represents the eight–digit part of products' names.

C.3.1.4 IRPH files

For all observations done during instrument activation sequence the pointing reported for the prime instrument is incorrect. Also the INSTRUME header keyword and the aperture field is wrongly filled. Instead of the prime instrument it gives QSS and Z0.

C.3.2 Second processing level

C.3.2.1 CISP files

When CSTA contains a spurious record as indicated in C.3.1.1, CISP's first record(s) will indicate that its detector readout belongs to CONF = 1. This is a left over from the previous parallel configuration and not the first configuration as defined in PGA and the user may safely disregard the corresponding readouts.

C.3.3 Third processing level

AAC is an automatic procedure intended to give the astronomer a set of analysis products. While every effort is made to ensure that these are as reliable as possible, including up–to–date calibrations, detailed and quantitative interpretation leading to publications will generally demand the more flexible and specific scrutiny potentially achievable with careful manual reduction of the data.

It has been found that AAC products have the "right look" when visualized on a computer screen. While it is believed that equatorial coordinates reported by AAC are correct to 3*arcsec PFOV + 3 arcsec (but see below for further discussion), estimated flux values, on the other hand, are currently estimated to be generally accurate to within ± 20 % for fixed filters and ± 30 % for CVFs of the indicated flux, pending ongoing more detailed assessment of the many factors affecting calibration (transients, glitches, wheel–jitter...) and of the data reduction algorithms. This holds also true for polarization observations done with LW2, LW3, LW7, LW9 and LW10. For polarization observations done with other filters, as a first order approximation, the flux has to be augmented by 20%.

CAM data analysis, including AAC **V7.0** , requires the exercise of caution as follows :

1. AAC uses only time–independent flat–field corrections available in the CAL–G libraries, leading to the possibility of less accurate fluxes compared with those calculated using optimised flat and dark corrections possible using interactive procedures, especially at the edges of the field of view.

2. AAC makes no transient or memory corrections, leading to the possibility of flux errors and the detection of spurious point sources.

3. Repositioning accuracy of the lens- and filter wheels leads to flat–fielding and minor coordinate errors, typically less than 2 pixels.

4. Bending of ISOCAM optical axis as a function of filter or of CVF position leads to coordinate errors of up to one pixel.

5. Optical distortion away from the optical centre leads to celestial position and integrated flux errors; see discussion in the IDUM.

6. CVF effective wavelength away from the optical centre leads to wavelength errors amounting to the equivalent of one CVF step.

7. AAC makes no straylight or ghost corrections leading to the possibility of flux errors and the detection of spurious sources.

8. Errors calculated by AAC are only statistical and take no account of any systematic effects.

9. Observations performed near the extremes of the CVF ranges, i.e. outside the range recommended in the Cookbook, are less accurately calibrated.
10. Non-linearity of detectors is not taken into account. This affects in particular the SW detector when a very bright point source is observed.

The observer should bear these in mind when assessing the contents of the individual AAC products discussed below :

C.3.3.1 CCIM files

There are no known problems with these files.

C.3.3.2 CMAP products

- CMAPs are constructed using the reference celestial coordinates from the IRPH files with no correction for the IIPH pointing jitter, which as discussed above, is of amplitude small enough to be dominated by measurement noise, but still large enough to give image blur of about 1 arcsec in size.
- As indicated in the CAM cookbook, straylight generated ghosts can lead to spurious extended features within the data. Ghosts seem to affect mainly CVF data, at any field of view other than 1.5 arcsec [8], [31].
- Lens- and filter- wheel jitter can lead to imperfect flat–fielding at the edges of CMAPs, where errors up to a factor of 5 may occur.
- In case of SW observations of a strong source the dark current may be overestimated, which leads to an apparent negative background.

C.3.3.3 CMOS products

Since CMOS are created from CMAPs, all the caveats above apply. There are no further warnings.

C.3.3.4 CPSL

Despite the flux uncertainty reported in CPSL – which is a statistical error value – flux values are currently estimated to be generally accurate within ± 20 % for fixed filters and ± 30 % for CVFs of the indicated flux. This holds also true for polarization observations done with LW2, LW3, LW7, LW9 and LW10. For polarization observations done with other filters, as a first order approximation, the flux has to be augmented by 20%.

- Wavelengths and fluxes near the extreme wavelength end ranges of the CVF range are unreliable.
- Lens- and filter- wheel jitter can introduce spurious point sources in CPSL.
- For a variety of calibration and algorithmic related causes the point–source detection algorithm is prone to find some spurious sources for the SW detector or around strong gradients.
- AAC **V7.0** does not correct for detector stabilization and/or transient effects following a flux step. This may lead to contaminated fluxes or spurious hot spots in the CMAPs. In general, 60 percent of the true flux is detected after one readout. When leaving the source in a raster or beam switched observation, the signal decreases to 30–40 percent of the on–source value *but* beware of a long decay tail. The decay tail may create “ghost” sources.

- As there are no corrections for varying CVF wavelength across the detector, off–axis sources suffer a wavelength shift that may amount to the equivalent of one CVF step.
- Unrecognized glitches may be mistaken as sources.
- Astrometry is subject to the following errors :
  - No corrections are available yet for the variation of ISOCAM's optical axis as a function of filter (up to 2 arcsec), CVF position (up to 6 arcsec from one LW segment to the other) and for LW/SW separation (SW images are displaced -6 arcsec (Y–direction) and -3 arcsec (Z–direction) with respect to LW images. In extreme cases, a celestial source detected at one wavelength may be "re–detected" with different RA, DEC at other wavelengths.
  - No corrections are made for optical distortion (up to one pixel).
  - No corrections are made for any ISO pointing error (up to 21 arcsecs for revolutions done in the Spacecraft Verification Phase, around 4 arcsecs until revolution 451, and less than 2 arcsecs from revolution 452 onwards.

C.3.3.5 CSSP files

AAC **V7.0** collates spectral information for those sources listed in CPSL, so all the CPSL caveats apply here as well. In particular, the same source which is detected twice or more at apparently different positions in different filters will correspondingly give rise to two or more spectra.

C.3.3.6 CUFF

Flux error estimates reported in CUFF are statistical values and do not take into account systematic effects. Indicated average fluxes pertaining to CMAPs and CMOSs are reliable except for the end range of CVFs and in case of SW observations with a strong point source.

C.3.3.7 CJAM and CGLL files

CJAM and CGLL contain auxiliary information. The contents of CGLL has been verified and it is reliable.

Appendix D

Acronyms

ADC	Analog to Digital Converter
ADU	Analog Digital Units
AOT	Astronomical Observation Template
CAL	Procedure when ISOCAM is performing an internal flat–field calibration
CAMTU	ISOCAM Time Units
CCG*DARK	ISOCAM Dark Current Exposures
CCG*DEAD	ISOCAM Dead Pixel Map
CCG*DFLT	ISOCAM Detector Flat Field Library
CCGLWDMOD	LW parameters for the Time Dependent Dark correction
CCG*FRAME	ISOCAM Detector astrometric calibration
CCG*GLITCH	ISOCAM Glitch Model: Currently empty
CCG*LINEAR	ISOCAM Uniform Illumination Library
CCG*OFLT	ISOCAM Optical Flat Field Library
CCG*PSF	ISOCAM Point-Spread-Function Library
CCG*SPEC	ISOCAM Filter & CVF spectral data
CCG*STRAY	ISOCAM Non-dark local light model
CCG*TRANS	ISOCAM Model transients: Currently empty
CCIM	ISOCAM Images in detector coordinates
CD ROM	Compact Disk Read Only Medium
CGLL	ISOCAM Glitch List
CIA	ISOCAM Interactive Analysis package
CIER	ISOCAM Image ERD
CIDT	ISOCAM Instrument Dedicated Team
CISP	ISOCAM Standard Processed Data
CIST	ISOCAM Instrument Science Team
CJAM	ISOCAM Jitter, Memory and Stabilization information
CLN	ISOCAM procedure to clean the LW array
CLWCVF1	ISOCAM LW CVF1 description
CLWCVF2	ISOCAM LW CVF2 description
CMAP	ISOCAM Celestial IR image
CMOS	ISOCAM Celestial IR mosaic

(*=SW, LW)

CPER	ISOCAM ERD for parallel mode
CPSL	ISOCAM Point-Source List
CPSP	ISOCAM Standard Processed Data for parallel mode
CRPIXi	Reference pixels (as for the FITS keywords)
CSSP	ISOCAM Source Spectra
CSTA	ISOCAM Compact Status
CSWCVF	ISOCAM SW CVF description
CUFF	ISOCAM User-Friendly log File
CUS	Calibration Uplink System
CVF	Circular variable filter
CWHEELS	ISOCAM Wheel information table
DDS	Digitized Sky Survey
DFLT	Detector flat–field
DRK	Procedure when ISOCAM is performing a dark current measurement
ERD	Edited Raw Data
EWHL	Entrance Wheel position
FCVF	Filter wheel position
FLT	Flat–field
GST	Ground Station Time
IIPH	Instrument Instantaneous Pointing History
IRPH	Instrument Reference Pointing History
ISO	Infrared Space Observatory
ISOCAM	Infrared Space Observatory instrument: camera
ITK	Instrument time key
LW	Long wavelength channel of ISOCAM
MMT	Multiresolution Median Transform
Nstab	Number of stabilization exposures before a single pointing observation or before the first point of the raster
OBS	Procedure when ISOCAM is obtaining astronomical data
OBDH	On Board Handling System
OFLT	Optical flat–field
OLP	Off Line Processing
OP-MODE	ISOCAM Operational mode
ORBIT	Orbital parameter file
OTF	On Target Flag
PFOV	Pixel field of view and lens wheel position
PGA	Proposal Generation Aids
PSF	Point Spread Function
QLA	Quick Look Analysis
RATE	ADU/s
RMS	Root mean square
SCD	Science ISOCAM dataset. Each SCD correspond to a single CONFIGURATION of ISOCAM
SPD	Standard Processed Data
SW	Short wavelength channel of ISOCAM
SWHL	Selection Wheel position
TDF	Telemetry Distribution file
TDT	Target Dedicated Time
TINT	Integration time
UTC	Universal time (conventional)
UTK	Uniform time key