

Project Galileo

MIPS Software Process Description For Instrument Calibration GLL SSI

Gary Yagi

May 22, 1995

National Aeronautics and Space Administration



Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91109
D-4264
Version 2

D-4264

Multimission Image Processing Subsystem
Software Process Description
for GLL SSI

Prepared by: _____
Gary Yagi, MIPS Science Software Engineer

Approved by: _____
Ken Klaasen, GLL SSI Science Coordinator

Jerry Clark, Supervisor Planetary Applications Grp

Steve Noland, MIPS System Engineer

Table of Contents

1	INTRODUCTION.....	1
1.1	Purpose.....	1
1.2	Functional Description.....	1
1.3	Applicable Documents.....	2
2	BACKGROUND INFORMATION.....	4
2.1	Basic Notations and Formulae.....	4
2.2	The SSI Camera.....	5
2.2.1	Camera Frame Cycle.....	6
2.2.2	Output Camera Signal.....	6
2.2.3	Environmental Factors.....	7
3	RADIOMETRIC CALIBRATION.....	9
3.1	Input Calibration Data.....	11
3.2	Radiometric properties.....	14
3.2.1	System Gain Constant.....	14
3.2.2	Reciprocity.....	18
3.2.3	Light Transfer Function.....	19
3.3	Calibration Files.....	22
3.3.1	Radiometric Files.....	23
3.3.2	Dark-Current Files.....	23
3.3.3	Blemish Files.....	24
3.3.4	Shutter-Offset File.....	25
3.3.5	Radiometric Catalog.....	26
4	RADIOMETRIC CORRECTION.....	27
4.1	Linear Light-Transfer Function.....	27
4.2	Scaling to Radiometric Units.....	29
5	GEOMETRIC CALIBRATION.....	31
6	GEOMETRIC CORRECTION.....	34
7	IN-FLIGHT CALIBRATION.....	35
7.1	Calibration Plaque Data.....	35
7.2	Star Field Data.....	36
8	PROGRAM DESCRIPTIONS.....	37
8.1	Radiometric Calibration Programs.....	37
8.1.1	LTGEN.....	37
8.1.2	MOMGEN.....	38

8.1.3	CCDNOISE.....	39
8.1.4	CCDRECIP	42
8.1.5	CCDSLOPE	44
8.1.6	PICSUM.....	46
8.1.7	GALGEN	47
8.1.8	BLEMGEN.....	52
8.1.9	BLEMPIC.....	55
8.1.10	Examples	58
8.2	Radiometric Correction Programs.....	64
8.2.1	GALSOS.....	64
8.2.2	GALSOSI	66
8.2.3	Examples	67
8.3	Geometric Calibration Programs.....	68
8.3.1	INTERLOC	68
8.3.2	FIXLOC	69
8.3.3	GRIDLOCB.....	70
8.3.4	GETLOC	71
8.3.5	LOCUS.....	72
8.3.6	Examples	76

Document Change Log

Change Order	Date	Changes
Original	June 11, 1987	New
2.0	November 15, 1994	1) Document updated to reflect changes derived from the 1989 instrument calibration. 2) Program information updated to reflect current capabilities. 3) Document updated to include the new 15 1/6 frame rate to be used during Jupiter operations. 4) Document updated to include new On-Chip Mosaic capability to be used during Jupiter operations.

TBD Items

Page	Closure Date	Item Description
5	spring 1996	Dark current files for On-Chip Mosaic images are needed.

Acronyms and Abbreviations

ADC	Analog-to-Digital
CCD	Charge Coupled Device
DN	Data Number
EDR	Experiment Data Record
GLL	Galileo
GSSC	Galileo SSI Science Processing Subsystem
IRAF	Image Reduction and Analysis Facility
LTF	Light Transfer File
MIPL	Multimission Image Processing Laboratory
MIPS	Multimission Image Processing Subsystem
mm	Millimeter(s)
msec	Millisecond(s)
NL	number of lines
NOAO	National Optical Astronomy Observatories
NS	number of samples
PSF	Point Spread Function
RCP	Reciprocity File
REDR	Raw Experiment Data Record
RMS	Root Mean Square
RTG	Radioisotope Thermoelectric Generators
SSI	Solid State Imaging subsystem
SWR	Software Requirements Document
UDR	Unprocessed Experiment Data Record
VICAR	Video Image Communications and Retrieval software system

1 Introduction

1.1 Purpose

This document defines the software process for the instrument calibration function of the GLL SSI science processing subsystem (GSSC). This software set is designed to support camera development and calibration activities during (1) sensor screening, (2) optical bench testing, (3) environmental testing, and (4) cruise phase of the mission. All functions defined herein are traceable to the Software Requirements Document (SWR) for this subsystem, D-3741.

1.2 Functional Description

Instrument calibration consists of radiometric calibration, geometric calibration, modulation transfer function and point spread function analysis, noise analysis, variable vignetting, light leaks, and color reconstruction. Special purpose software are required to support radiometric, geometric, and color calibration. This version of the document contains the structural design of the radiometric and geometric calibration software. Design specifications for color calibration and in-flight calibration are deferred until more information is available. It is expected that the other aspects of instrument calibration will be supported by general purpose software (e.g., plots, histograms, pixel listings, mean and standard deviation statistics, and Fourier transforms) and are therefore not treated in this document.

The purpose of radiometric and geometric calibration is to measure the radiometric and geometric distortions of the camera and to remove these distortions from the flight imagery so that quantitative scientific measurements may be made from the data.

Figure 1 presents a functional overview of the image processing performed during radiometric and geometric calibration. Input radiometric calibration data consists of flat-field frames from light-transfer and reciprocity sequences and images of star-fields.

The radiometric properties of the camera are measured and reported in the form of image displays, plots, and statistical listings. Radiometric, dark-current, shutter-offset, and blemish files are generated and used later to radiometrically correct the flight image data.

Pre-launch input geometric calibration data consists of images of a geometric grid target, together with measurements of the actual grid target. The geometric camera

distortions are measured and reported in the form of image displays of the distortion vectors and a tabulation of the RMS errors.

1.3 Applicable Documents

1. MIPL Requirements Document, D-4419.
2. MIPL Software Requirements for Galileo SSI Science, D-3741.
3. Functional Requirements, Galileo Orbiter Flight Equipment Solid State Imaging Subsystem, GLL-4-2036B.
4. Galileo Project SSI Calibration Requirements Document, Subsystem Level Calibration, 625-606.
5. K. Klaasen, M. Clary, J. Janesick, "Charge-coupled device for the Galileo Jupiter Orbiter spacecraft", paper update June 7, 1984.
6. Galileo Solid-State Imaging Subsystem Calibration Report: Part 1, K. Klaasen, H. Breneman, D-5880, November 1, 1988.
7. Galileo Solid-State Imaging Subsystem Calibration Report: Part 2, K. Klaasen, D-5880, March 22, 1993.

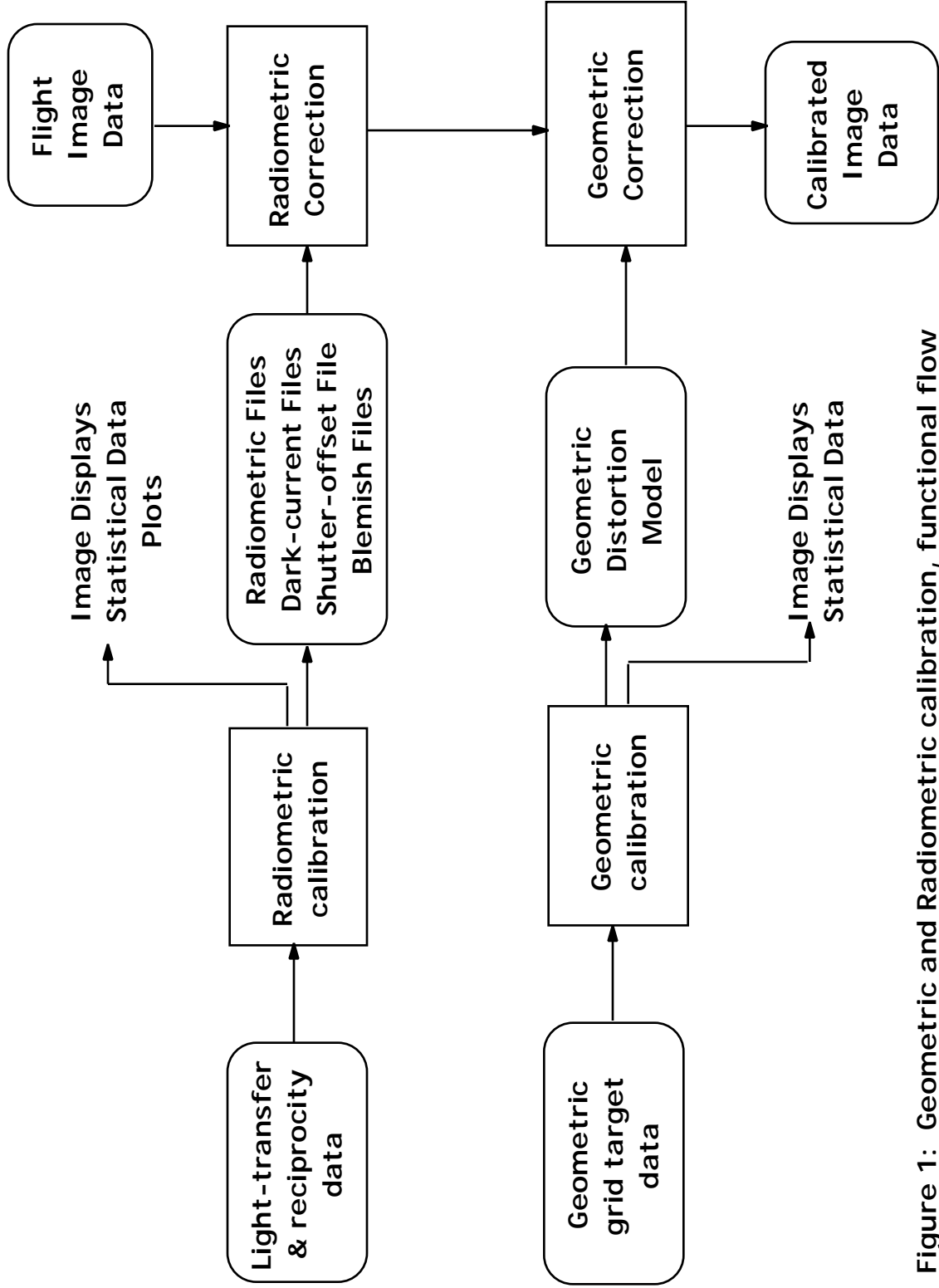


Figure 1: Geometric and Radiometric calibration, functional flow

2 Background Information

2.1 Basic Notations and Formulae

An image may be represented as an $m \times n$ matrix:

$$D = \begin{matrix} d(1,1) & d(1,2) & d(1,3) & \dots & d(1,n) \\ d(2,1) & d(2,2) & d(2,3) & \dots & d(2,n) \\ d(3,1) & d(3,2) & d(3,3) & \dots & d(3,n) \\ \vdots & \vdots & \vdots & & \vdots \\ d(m,1) & d(m,2) & d(m,3) & \dots & d(m,n) \end{matrix}$$

where $d(i, j)$ is the data number (DN) value at line i and sample j .

The sum of two images, $F=D + E$, is obtained by summing each of the corresponding elements: $f(i, j) = d(i, j) + e(i, j)$. We define a term-by-term product of two images $F=D * E$ as the product of each of the corresponding elements: $f(i, j) = d(i, j)e(i, j)$.

The mean DN μ_D and standard deviation σ_D of the image D are computed as follows:

$$\mu_D = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n d(i, j)$$

$$\sigma_D = \sqrt{\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n [d(i, j) - \mu_D]^2}$$

By rearranging the latter formula, we get an equation that is computationally more efficient:

$$\sigma_D = \sqrt{\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n [d(i, j)]^2 - \mu_D^2}$$

Throughout this document, upper case letters are used to represent matrices and lower case letters are used to represent scalar quantities.

2.2 The SSI Camera

The SSI subsystem consists of a 1500 mm telescope, 8 selectable filters mounted on a filter wheel, and an 800x800 CCD sensor. The camera can be operated at 4 gain states, 4 frame rates, at normal or extended exposure, in full-frame or summation mode, in inverted or non-inverted mode, with blemish-protection on or off, and with the data compressor on or off.

The filters consist of a clear filter and 7 narrow-bandpass filters spanning the range from 350 to 1100 nanometers. The CCD has a spectral response from 350 to 1050 nanometers.

The video analog-to-digital converter (ADC) converts the analog video data to eight bits. The SSI has 4 gain states (approximately 10,000, 40,000, 100,000, and 400,000 electrons full-scale). The lowest gain state is scaled to provide full-scale data for full-well of the CCD during summation mode read-out (400,000 electrons). The highest gain state is scaled to provide full-scale for a CCD signal of 10,000 electrons.

The SSI acquires full-frame data at frame rates of 8-2/3, 30-1/3, and 60-2/3 seconds. In summation mode, the SSI acquires data at reduced spatial resolution (400x400) at frame rates of 2-1/3 and 15 1/6 seconds. The 400,000 e^- gain state is achieved only in summation mode.

Summation Mode: In summation mode, the image is reduced to a 400x400 format by electronically summing 2x2 pixel areas. This results in a 4x reduction in data rate. The data compressor cannot be operated in summation mode.

Extended Exposure Mode: For all full frame images, all exposures longer than 800 milliseconds extend past the normal camera prepare cycle. For such exposures, frame read-out is suppressed until the following frame cycle. For summation-mode images taken at 2 1/3 second frame rates, all exposures longer than 400 milliseconds must be taken in this extended-exposure mode. For summation-mode images taken at 15 1/6 second frame rates, all exposures longer than 533 milliseconds must be taken in extended-exposure mode. All such extended exposure frames will contain a correspondingly larger dark current component.

On-Chip Mosaic Mode: The SSI camera may be operated in a multiple-exposure single read-out mode. A typical application is where four images of a target body are acquired by slewing the camera (and stepping the filter wheel) to image the target in

each of the four corners of the CCD array. CCD read-out is suppressed until all four exposures are completed, thus resulting in a 2x2 mosaic. 3x3, 5x5, or NxM mosaics are possible, where N and M are limited only by the pixel diameter of the target and pointing accuracy of the scan platform. Provisions for removing the dark-current from these on-chip mosaics is TBD.

Inverted Mode: The SSI may be operated with inverted parallel clock voltages. This capability suppresses CCD dark-current but results in increased spurious charge.

Blemish-Protection Mode: The SSI may be operated in a blemish-protection mode which reduces the effect of charge spillover into the CCD's serial register in the event that the CCD develops an image degrading blemish.

Under nominal conditions, the inverted and blemish-protection modes will not be used. The camera is switched to these modes only if it develops severe blemish or dark-current problems.

2.2.1 Camera Frame Cycle

Each camera frame cycle is composed of a prepare and a readout cycle. During the prepare cycle, the filter wheel is stopped, light flood is applied, the CCD is erased to reduce the dark-current level and the shutter is activated. During the read cycle the CCD is read by shifting the collected charge from image line to image line (in the column direction), causing the first image line to be shifted into the serial register. That line is then read out serially through the on-chip amplifier. This procedure is repeated until each of the 800 lines has been read out.

2.2.2 Output Camera Signal

The output camera signal D consists of the following components:

$$D = S + D_0 + N$$

where

S is the signal due to exposure to the light source

D_0 is the dark-current signal including any pedestal or offset level

N is the noise

The signal S contains the image information, and is determined by the object scene and the radiometric, geometric, and modulation transfer characteristics of the camera.

The dark-current D_0 consists of two terms: (1) thermally generated charge which builds up on the CCD array and (2) an electronic offset (pedestal). The dark-charge increases linearly with time and is pixel dependent, while the pedestal drifts slowly with time and is a constant for a given frame. At the SSI CCD temperature of -110°C , the thermal dark-current produces charge at a rate less than $10^{-3}\text{ e}^-/\text{pixel}/\text{sec}$. However, local dark-current blemishes, called dark-current spikes, also occur because of defects in the bulk silicon. Charge is produced at these sites at the rate of 0.01 to $10\text{ e}^-/\text{pixel}/\text{sec}$ at -110°C .

The noise N consists of random and periodic components. The standard deviation of N , σ_N , is used as the measure of the noise.

The output signal also contains camera blemishes. Camera blemishes are pixels in the CCD array for which the output signal cannot be radiometrically calibrated. Camera blemishes are removed as a factor from most of the computations by identifying and avoiding blemish areas in the analysis, or by cosmetically removing them via interpolation.

2.2.3 Environmental Factors

The camera performance is effected by the following environmental factors:

Camera Operating Temperature: The nominal CCD temperature during flight is -110°C . The SSI temperature control system is designed to maintain the CCD at this temperature to within 0.5°C .

Jovian Radiation Environment: In the near-Jovian radiation belt, interaction of high-energy particles with the CCD produces unwanted charge. The charge from any given interaction is normally expected to be concentrated within a few picture elements. A degradation in the signal-to-noise ratio in the image results from these interactions.

RTG Radiation: Neutron radiation damage due to the spacecraft's radioisotope thermoelectric generators will increase the number of dark-current spikes over the cruise and orbital phase of the mission.

3 Radiometric Calibration

During radiometric calibration, the camera's response to incident light is measured as a function of amplitude (light-transfer) and wavelength (spectral response). Any spatial dependence in the camera's response is detected by measuring its radiometric properties over its entire field-of-view.

Input data to radiometric calibration consists of light-transfer and reciprocity sequences acquired during optical-bench and thermal-vacuum chamber testing prior to launch, and calibration plaque, and dark-current data acquired periodically throughout the cruise and orbital phases of the mission.

No special software is required to support spectral calibration, and that activity is therefore not discussed here.

Two sets of programs support light-transfer calibration. The first set, consisting of programs LTGEN, MOMGEN, CCDNOISE, CCDRECIP, and CCDSLOPE, measure the radiometric properties of the camera. The second set, consisting of programs PICSUM, GALGEN, BLEMGEN and BLEMPIC, generate radiometric, dark-current and blemish files which may subsequently be used to radiometrically correct and remove camera blemishes from the flight data.

Figure 2 shows the functional flow of the processing performed to measure the radiometric camera properties. Program LTGEN is first executed to create and initialize a Light-Transfer File and a Reciprocity File. The input light-transfer and reciprocity data is then processed by MOMGEN to extract statistical data for a grid of areas uniformly distributed over the camera's field-of-view. These statistical data are stored in the Light-Transfer and Reciprocity Files. The Reciprocity File is accessed by CCDRECIP to determine the shutter offset and camera sensitivity. The Light-Transfer File is then accessed by CCDNOISE to determine the system gain constant, and by CCDSLOPE to determine the slope and offset terms of the linear light-transfer function. The radiometric properties are determined for each area of the grid and global values computed. Plots and displays are generated, in addition to a listing of the statistical results for each area.

Figure 3 shows the functional flow of the processing performed to generate radiometric, and dark-current files. The input light-transfer data is first consolidated by summing all multiple exposures via PICSUM. The data

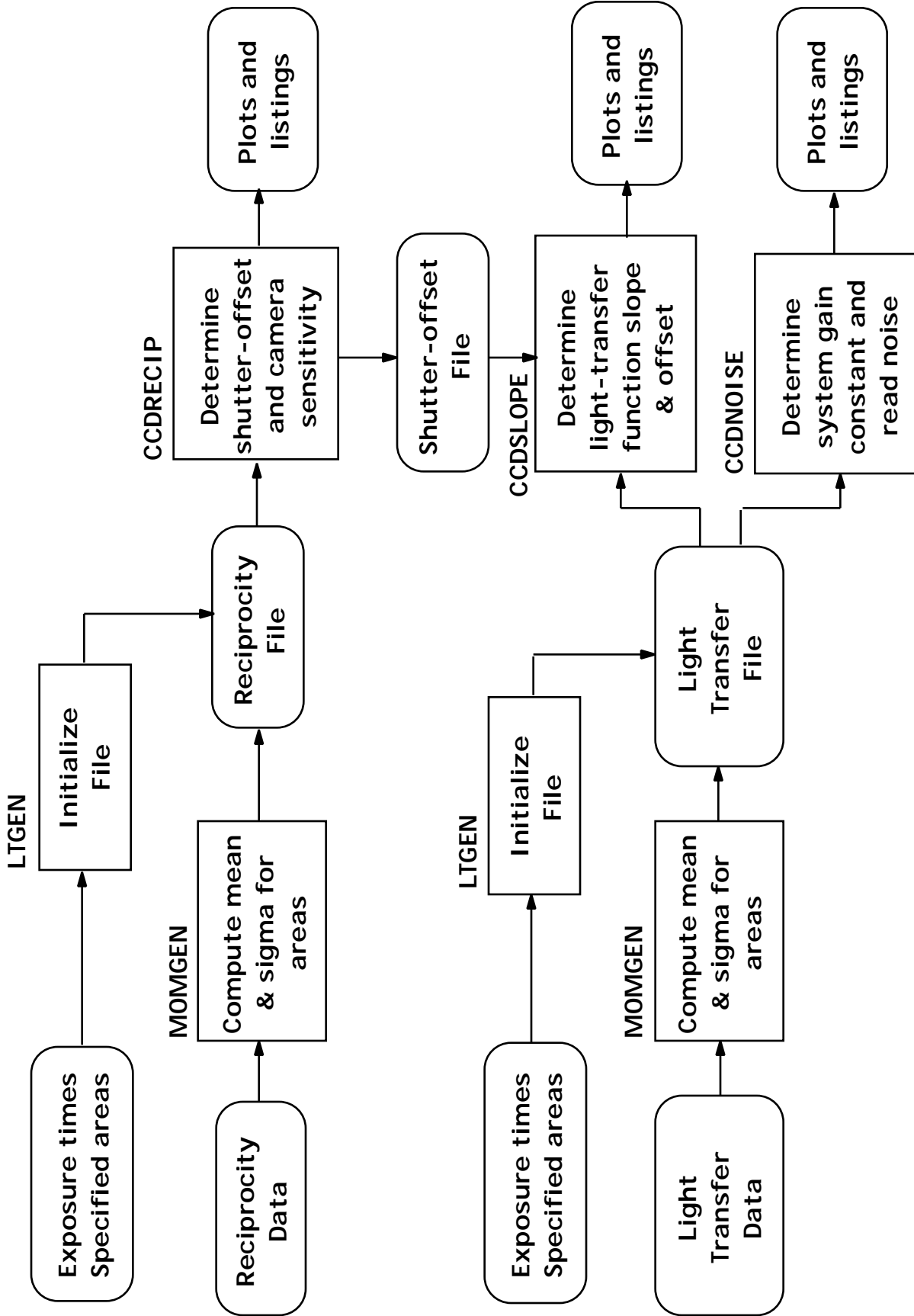


Figure 2: Measuring the radiometric properties of the camera

are then used to generate radiometric files based on a linear model of the light-transfer function via program GALGEN. Outputs from GALGEN consist of a radiometric file containing the slope terms of the linear function, a dark-current file, files containing the maximum absolute error and RMS error of the fit, and a file identifying all low full-well pixels by their saturation DN. The radiometric file may be used by program GALSOS to radiometrically correct the flight data. The radiometric and dark-current files may be displayed as images.

In figure 4, the outputs from GALGEN are input into BLEMGEN, which identifies and classifies the camera blemishes and low full-well pixels. The resulting blemish file may be displayed via program BLEMPIC.

3.1 Input Calibration Data

Input calibration data consists of flat-field frames, i.e. frames that have been uniformly illuminated over the camera's field-of-view. During optical bench calibration, the uniform illumination is provided by a light cannon. During flight, a calibration plaque mounted on the spacecraft is imaged using solar illumination.

During radiometric calibration, two types of imaging sequences are acquired: (1) light transfer sequences, and (2) reciprocity sequences. In a light transfer sequence, the light source is maintained at a constant brightness, while the exposure time is increased in steps from zero to maximum exposure. In a reciprocity sequence, a constant exposure is maintained via inversely proportional changes in the scene radiance and exposure time. During both of these calibration sequences, all other camera parameters are held constant (e.g., temperature, filter position, gain state, frame rate, operating mode). Multiple exposures are normally taken at each exposure setting to reduce the effect of shot noise.

Included at the beginning and/or end of each calibration sequence are frames acquired by reading out the CCD without prior shuttering. The principal component in these frames is the dark-current, since there is no signal term S . These frames (called dark-current frames) are acquired to measure the dark current level and to create dark-current files. Because the dark-current level is affected by temporal and temperature changes, dark-current frames are also periodically acquired throughout the cruise

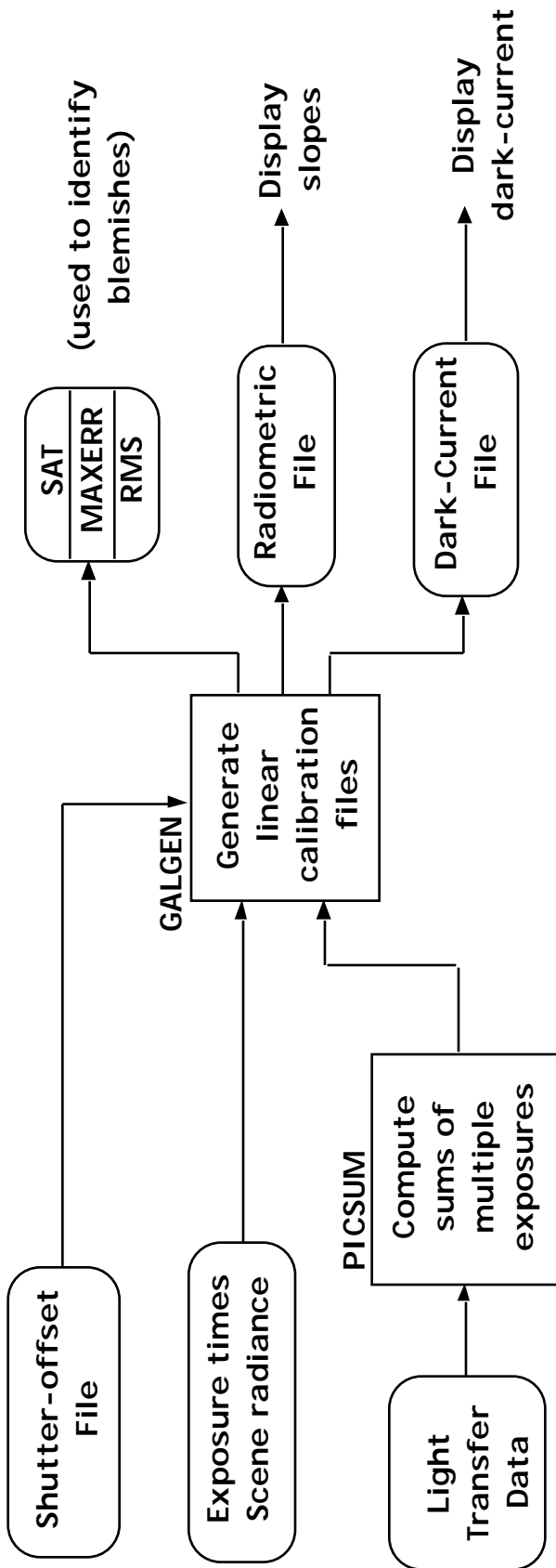


Figure 3: Generating radiometric and dark-current files

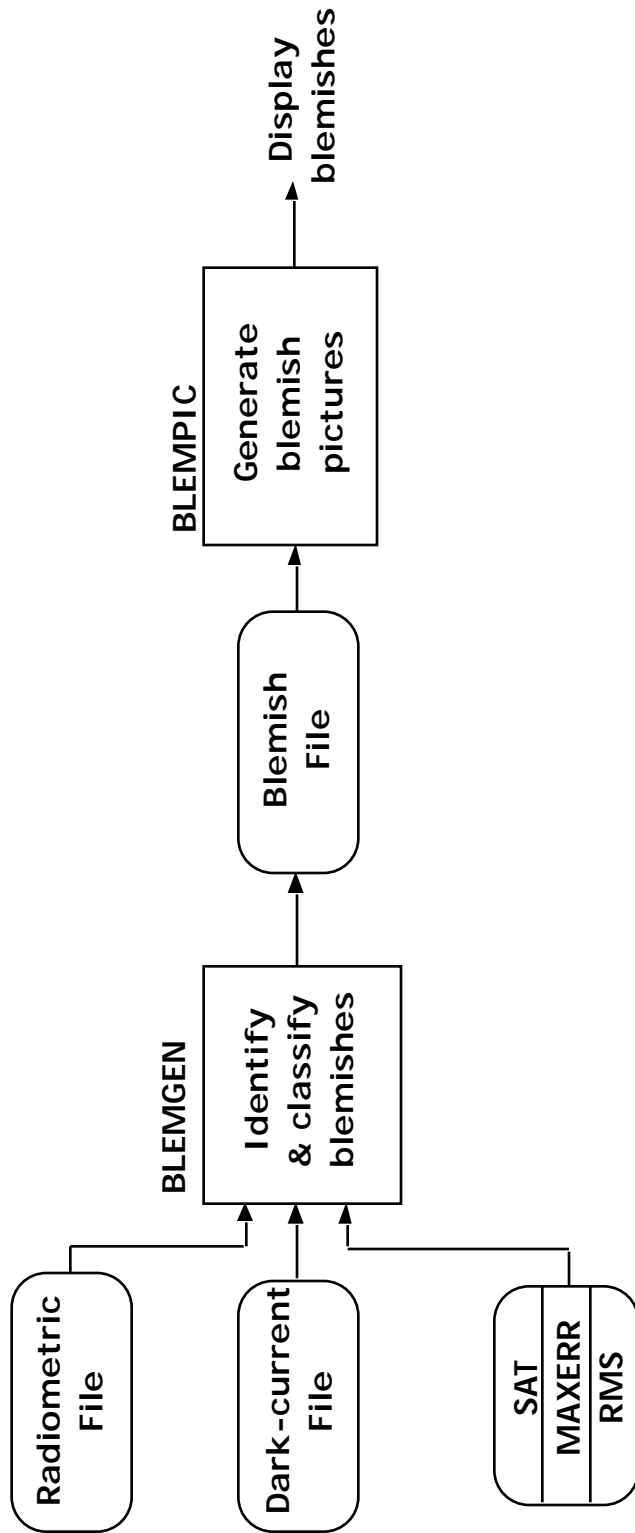


Figure 4: Generating Blemish files

and orbital phases of the mission.

Let $n_0, n_1, n_2, \dots, n_m$ be the number of frames taken at exposure settings 0,1,2,3,...,m. Let $D_k^{(l)}$ represent the k th flat field frame taken at exposure level l . Then a typical light transfer (or reciprocity) sequence consists of the following:

$$\begin{array}{cccccc} D_1^{(0)}, & D_2^{(0)}, & D_3^{(0)}, & \dots & D_{n_0}^{(0)} \\ D_1^{(1)}, & D_{2_1}^{(1)}, & D_3^{(1)}, & \dots & D_{n_1}^{(1)} \\ D_1^{(2)}, & D_2^{(2)}, & D_3^{(2)}, & \dots & D_{n_2}^{(2)} \\ \vdots & \vdots, & \vdots, & & \vdots \\ D_1^{(m)}, & D_2^{(m)}, & D_3^{(m)}, & \dots & D_{n_m}^{(m)} \end{array}$$

where the frames at the zero exposure setting ($D_1^{(0)}, D_2^{(0)}, \dots$) are the dark-current frames.

The superscripts are sometimes omitted when the frames from a fixed exposure are treated, e.g., D_1, D_2 , etc.

3.2 Radiometric properties

3.2.1 System Gain Constant

Let D be a flat field frame with components:

$$D = S + D_0 + N$$

The system gain constant k (in electrons per DN) may be derived from the signal (S) and noise (N) components of the image.

The noise component is due to shot noise and read noise:

$$N(e^-) = \sqrt{\overset{2}{SHOT}(e^-) + \overset{2}{READ}(e^-)}$$

where the suffix (e^-) is used to indicate that the terms are expressed in units of electrons.

The shot noise is caused by the statistical uncertainty in photon arrival, and varies with the light signal level S as follows:

$$SHOT(e^-) = \sqrt{\mu_{S(e^-)}} \quad (1)$$

Therefore:

$$N(e^-) = \sqrt{\mu_{S(e^-)}^2 + \text{READ}(e^-)^2} \quad (2)$$

The system gain constant k can be used to convert this equation into units of DN:

$$N = \sqrt{\mu_S^2 + \text{READ}^2}$$

or

$$= \frac{\mu_S}{N - \frac{\text{READ}^2}{N}} \quad (3)$$

The following paragraphs describe how k and READ are extracted from equation (3) by measuring the mean signal μ_S and total noise N at each exposure level of a light transfer sequence.

Measuring the Signal: Let $D^{(0)}$ be a zero-exposure (or dark-current) frame and let $D^{(l)}$ be a flat-field frame taken at exposure level l :

$$\begin{aligned} D^{(0)} &= D_0^{(0)} + N^{(0)} \\ D^{(l)} &= S^{(l)} + D_0^{(l)} + N^{(l)} \end{aligned}$$

Assuming that the dark current level has been held constant by maintaining a constant temperature and constant time interval between exposures, the dark-current may be removed by subtracting these two frames:

$$D^{(l)} - D^{(0)} = S^{(l)} + N^{(l)} - N^{(0)}$$

Since the noise terms have zero mean values, the mean signal at exposure level l is equal to the difference between the mean DN's of frames $D^{(l)}$ and $D^{(0)}$:

$$\mu_S^{(l)} = \mu_D^{(l)} - \mu_D^{(0)} \quad (4)$$

When multiple frames at each exposure are available, the data are combined to compute an average μ_S . Let $D_1^{(0)}, D_2^{(0)}, D_3^{(0)}, \dots, D_{n_0}^{(0)}$ be flat-field frames taken at zero exposure and let $D_1^{(l)}, D_2^{(l)}, D_3^{(l)}, \dots, D_{n_l}^{(l)}$ be flat field frames taken at exposure level l . The

mean DN at exposure level l is computed by averaging the mean DN's of each frame taken at that exposure:

$$\mu_D^{(l)} = \frac{1}{n_l} (\mu_1^{(l)} + \mu_2^{(l)} + \mu_3^{(l)} + \dots + \mu_{n_l}^{(l)}) \quad (5)$$

where $\mu_i^{(l)}$ is the mean DN for frame $D_i^{(l)}$. Similarly, the mean DN at zero exposure is computed by averaging the mean DN's of all the dark current frames:

$$\mu_D^{(0)} = \frac{1}{n_0} (\mu_1^{(0)} + \mu_2^{(0)} + \mu_3^{(0)} + \dots + \mu_{n_0}^{(0)}) \quad (6)$$

The mean signal μ_s can then be computed using equation (4).

Measuring the Noise: Let $D_1^{(l)}$ and $D_2^{(l)}$ be two flat field frames taken at the same exposure level and under identical conditions:

$$\begin{aligned} D_1^{(l)} &= S^{(l)} + D_0^{(l)} + N_1^{(l)} \\ D_2^{(l)} &= S^{(l)} + D_0^{(l)} + N_2^{(l)} \end{aligned}$$

Since the signal due to the light source and dark current are identical, the difference between the two frames is equal to the difference of their noise terms:

$$D_{12}^{(l)} = D_1^{(l)} - D_2^{(l)} = N_1^{(l)} - N_2^{(l)}$$

Let $N_{12}^{(l)}$ represent the noise level of the difference image $D_{12}^{(l)}$. Assuming that the noise from the two input frames are uncorrelated, the total noise at exposure level l is given by:

$$N_N^{(l)} = \frac{1}{\sqrt{2}} N_{12}^{(l)}$$

A more accurate measure of the noise may be obtained by using multiple frames $D_1^{(l)}, D_2^{(l)}, D_3^{(l)}, \dots, D_{n_l}^{(l)}$. By taking differences $D_i^{(l)} - D_{i+1}^{(l)}$, $i = 1, 2, 3, \dots, n_l - 1$, computing standard deviations $\sigma_{i, i+1}$, and taking the average value of these results, we get:

$$\sigma_N^{(l)} = \frac{1}{(n_l - 1)\sqrt{2}} \left(\sigma_{12}^{(l)} + \sigma_{23}^{(l)} + \sigma_{34}^{(l)} + \dots + \sigma_{n_l-1, n_l}^{(l)} \right) \quad (7)$$

Solving for k: Let the set of frames $\{D_i^{(l)}\}$ be a be a light transfer sequence consisting of multiple frames $i = 1, 2, 3, \dots, n_l$, taken at exposure levels $l = 0, 1, 2, \dots, m$, respectively (see Sec.3.1). Let $\mu_S^{(l)}$ and $\sigma_N^{(l)}$ be the mean signal and total RMS noise at exposure level l , as derived from equations 4 – 7 above. The set of data points $\{\mu_S^{(l)}, \sigma_N^{(l)}\}$, $l = 1, 2, \dots, m$ may be used to compute a least-squares solution for the system gain constant k and read noise σ_{READ} by rearranging Equation 3:

$$\sigma_N^2 = \frac{1}{k} \mu_S + \sigma_{READ}^2$$

The result is a linear equation of the form $y = mx + b$, where $x = \mu_S$ and $y = \sigma_N^2$. The least-squares solution for this linear equation is:

$$m = \frac{\sum_{l=1}^m \mu_S^{(l)} \sigma_N^{(l)2} - \left[\sum_{l=1}^m \mu_S^{(l)} \right] \left[\sum_{l=1}^m \sigma_N^{(l)2} \right]}{\sum_{l=1}^m \mu_S^{(l)2} - \left[\sum_{l=1}^m \mu_S^{(l)} \right]^2} \quad (8)$$

and

$$\sigma_{READ}^2 = \frac{\sum_{l=1}^m \mu_S^{(l)2} \left[\sum_{l=1}^m \sigma_N^{(l)2} \right] - \left[\sum_{l=1}^m \mu_S^{(l)} \right] \left[\sum_{l=1}^m \mu_S^{(l)} \sigma_N^{(l)2} \right]}{\sum_{l=1}^m \mu_S^{(l)2} - \left[\sum_{l=1}^m \mu_S^{(l)} \right]^2} \quad (9)$$

where all the summations are for $l = 1$ to m .

3.2.2 Reciprocity

A fully reciprocal camera system produces a constant video signal output under constant exposure. During acquisition of a reciprocity sequence, the exposure is kept constant by using inversely proportional changes in shutter time and scene radiance. Data from the reciprocity sequence are used to determine the mean camera sensitivity and the shutter offset.

Let D be a flat field frame with components:

$$D = S + D_0$$

where S is the signal and D_0 is the dark-current (noise term ignored). For a reciprocal camera system, the signal S is proportional to the exposure as follows:

$$S = Cl(t - t_0) \quad (10)$$

where

- C is the camera sensitivity in DN per relative foot-Lambert-msec
- l is the luminance of the light-cannon in relative foot-Lamberts
- t is the commanded exposure time in milliseconds
- t_0 is the shutter offset

The actual exposure time differs slightly from the commanded shutter time (shutter offset) because the shutter is mechanically activated.

Solving for μ_c and t_0 : Let the set of frames $\{D_k^{(i)}\}$ be a reciprocity sequence consisting of multiple frames $k = 1, 2, 3, \dots, n_i$ taken at exposure levels $i = 1, 2, \dots, m$ (see Sec.3.1). Let t_i and l_i be the commanded exposure time and scene luminance for exposure level i .

Let $\mu_s^{(i)}$ be the mean signal at exposure level i , as calculated by using equations 4 – 6.

Then

$$\mu_s^{(i)} = \mu_c l_i (t_i - t_0) \quad (11)$$

where μ_c is the mean camera sensitivity. Define a variable x such that:

$$x_i = \frac{\mu_s^{(i)}}{l_i} \quad (12)$$

Given data points $\{t_i, x_i\}$, we solve for the value of μ_c and t_0 which minimizes the function:

$$= \frac{1}{2} \sum_{i=1}^m w_i [\mu_c (t_i - t_0) - x_i]^2$$

where the data points are weighted such that $w_i = l_i^2$. By equating the partial derivatives of F with respect to μ_c and t_0 to zero, we get:

$$\mu_C = \frac{w_i \quad w_i x_i t_i - \quad w_i x_i \quad w_i t_i}{w_i \quad w_i t_i^2 - \left(\quad w_i t_i \right)^2} \quad (13)$$

and

$$t_0 = \frac{\mu_C \quad w_i t_i - \quad w_i x_i}{\mu_C \quad w_i} \quad (14)$$

where all the sums are from $i = 1$ to m .

Because of the acceleration of the shutter blades, which travel in a vertical direction, the shutter offset t_0 is image line dependent. See the program description for CCDRECIP (below) for details on how this line dependence is determined.

3.2.3 Light Transfer Function

The light-transfer function F defines the transformation from exposure to a given scene luminance to output camera signal:

$$D = F(E)$$

where

D is the output image (DN),

E is the exposure (in relative foot-Lambert-msec),

The Galileo SSI nominally has a linear light-transfer function:

$$D = C \ E + D_0 \quad (15)$$

where

C is the camera sensitivity (slope term), and

D_0 is the dark-current (offset term).

The signal term, which results from exposure of the CCD to the light source ($S = C \ E$), is the term-by-term product of matrices C and E , such that each element is of the form $c(i, j) \times e(i, j)$.

The matrices C , D_0 , and E may be regarded as images with real DN values, while the output image D assumes integral DN values. For a fixed camera state (given filter position, gain state, etc.) C and D_0 are camera constants, whereas E is a function of the

object scene. The purpose of light transfer calibration is to determine constants C and D_0 .

If D is a flat field frame from a light-transfer sequence, then the exposure $E = e(i)$ may be computed from the luminance l of the light cannon (in relative foot-Lamberts), the commanded shutter time t (in milliseconds), and the shutter offset t_0 :

$$e(i) = l[t - t_0(i)] \quad (16)$$

The exposure $e(i)$ is line-dependent because the shutter offset $t_0(i)$ is line-dependent.

Each pixel in the CCD array acts as an independent sensor with its own characteristic light-transfer curve:

$$d(i, j) = c(i, j)e(i) + d_0(i, j) \quad (17)$$

where $d(i, j)$, $c(i, j)$, and $d_0(i, j)$ represent the matrix elements corresponding the pixel coordinates (i, j) for images D , C , and D_0 , respectively.

Solving for C and D_0 : Let $D^{(0)}, D^{(1)}, D^{(2)}, \dots, D^{(m)}$ represent the flat-field frames of a light transfer sequence, taken at exposure levels $k = 0, 1, 2, \dots, m$, and let $e_k(i)$ be the corresponding line-dependent exposure levels.

Let $d_k(i, j)$ represent the matrix element corresponding to pixel coordinates (i, j) for matrix $D^{(k)}$. The set of data points $\{e_k(i), d_k(i, j)\}$ may be used to compute a least-squares solution to equation 17 as follows:

$$c(i, j) = \frac{m \sum_k d_k(i, j)e_k(i) - \sum_k d_k(i, j) \sum_k e_k(i)}{m \sum_k e_k^2(i) - \left[\sum_k e_k(i) \right]^2} \quad (18)$$

and

$$d_0(i, j) = \frac{\sum_k d_k(i, j) \sum_k e_k(i) - \sum_k d_k(i, j)e_k(i) \sum_k e_k(i)}{m \sum_k e_k^2(i) - \left[\sum_k e_k(i) \right]^2} \quad (19)$$

where all the sums are from $k=1$ to m .

Solving for C only: Assuming that the dark-current level remains constant throughout the light-transfer sequence, the dark-current frame $D^{(0)}$ may be used to subtract the dark-current from each of the frames in the sequence to extract the signal term S :

$$S^{(k)} = D^{(k)} - D^{(0)} \quad (20)$$

Let $s_k(i, j)$ represent the matrix element corresponding to pixel coordinates (i, j) for matrix $S^{(k)}$. The set of data points $\{e_k(i), s_k(i, j)\}$ may be used to compute a least-squares solution to the equation

$$s(i, j) = c(i, j)e(i) \quad (21)$$

as follows:

$$c(i, j) = \frac{s_k(i, j)e_k(i)}{e_k^2(i)} \quad (22)$$

where the sums are from $k=1$ to m .

Low Full-Well Pixel Test: The linearity of the light-transfer function breaks down when the number of electrons in a potential well exceeds the charge-transfer capacity of the pixel. The system gain constants k are selected such that, for the vast majority of the pixels, the output signal scale is saturated (255 DN) before the charge-transfer capacity of the pixel is exceeded. Those pixels for which this is not the case (low full-well pixels) are identified by determining whether the data points $\{(e_k, d_k)\}$ depart from linearity at the higher exposure levels. For these low full-well pixels, the DN value at which saturation occurs (d_{fw}) is measured and recorded in the blemish file (see section 3.3.3)

Note that, because the CCD array is read out by repeatedly shifting the collected charge from image line to image line, any pixel with a lower than average charge transfer capacity will prevent the charge from any pixel below it from being transferred up the column properly. Therefore, if a column contains a low full-well pixel, all pixels below it (and on the same column) will be a low full-well pixel, with the same saturation DN (d_{fw}).

The low full-well pixel test is described here for the linear model. The test is analogous for the slope model. Starting from the lowest exposure, fit the first n data points to the linear model using equations 18 and 19, and let c and d_0 be the resulting coefficients. The next data point d_{n+1} is tested to determine whether it is sufficiently close to the computed curve. If:

$$ce_{n+1} + d_0 - d_{n+1} < \epsilon_{n+1}$$

where ϵ_{n+1} is some specified error threshold, then add d_{n+1} to the set and recompute c and d_0 . If the difference is greater than or equal to the error, then $d_{fw} = d_n$, and d_{n+1} and

all higher exposure data points are rejected as being in the non-linear region of the light-transfer curve. This test is repeated iteratively until the last data point has been included or the test fails.

The error threshold ϵ_k may be chosen as a linear function of the exposure $t_k(i)$:

$$\epsilon_k = \epsilon_{k1} t_k(i) + \epsilon_{k0}$$

If the low-full-well pixel locations and saturation values are already known prior to analyzing a given light transfer sequence, the low-full-well pixel test can be bypassed.

3.3 Calibration Files

Four types of calibration files are required to perform radiometric correction on the flight data: (1) radiometric files, (2) dark-current files, (3) blemish files, and (4) shutter-offset files. The radiometric files model the camera's light-transfer function with a linear function.

Radiometric files are required for each filter (8) in both full-frame and summation modes (total=16). The files should be scaled for the lowest gain-state. The data taken at higher gain-states can be corrected using the resulting files since the ratios between the gain states are known constants.

A total of 136 dark-current files are required. For full-frame mode, files are required at 3 gain-states (10K, 40K, and 100K) and 3 frame rates (8-2/3, 30-1/3, 60-2/3 seconds). For summation mode, files are required at all 4 gain-states at both 2-1/3 seconds and 15 1/6 seconds frame rates. For all the above combinations, files are required for both normal and extended-exposure, for both inverted and non-inverted modes, and for blemish protection on or off. Additional dark-current files may be required for on-chip mosaic sequences.

Blemish files are required at each filter (8), at the three lowest gain-states for full-frame data, and at all four gain-states for summation mode data (total=56).

Only one shutter-offset file is required. The shutter-offset file is generated by program CCDRECIP.

Creation of separate radiometric and blemish files for inverted/non-inverted modes, blemish protection on or off, and data compressor on or off will be deferred until these modes become necessary. The files will be periodically updated to reflect changes in camera sensitivity, dark-current level, and blemishes.

3.3.1 Radiometric Files

The radiometric file is an image with the same dimension as the SSI frame (800 x 800 for the full-frame format and 400 x 400 for summation-mode frames). Each pixel $z(i,j)$ in the file represents the slope term of the inverse light-transfer function for the lowest gain state:

$$e(i,j) = z(i,j)(d(i,j) - d_0(i,j)) \quad (23)$$

The $z(i,j)$ are computed from equation 18, where $z = 1/c$ and are stored in the file in REAL*4 data format.

3.3.2 Dark-Current Files

Dark current files may be (1) byte data format or (2) 16-bit integer data format. Files in byte format consist of unshuttered (or dark-current) frames, acquired to measure the dark-current level.

Files in 16-bit integer format are scaled to $128 \times$ the mean dark-current level to provide resolution to one-hundredth of a DN. These 16-bit files are generated by combining the data from multiple dark-current frames via summation, or by fitting the data from an entire light-transfer sequence and solving for the dark-current (see equation 19 and program GALGEN in section 8.1.7).

Let $D_1^{(0)}, D_2^{(0)}, D_3^{(0)}, \dots, D_n^{(0)}$ be a set of n dark-current frames taken under identical conditions. The data from these frames may be combined via summation (see program PICSUM in section 8.1.6):

$$D_0 = \frac{128}{n} \sum_{i=1}^n D_i^{(0)}$$

where the resulting output (in 16-bit integer format) has been scaled to $128 \times$ the dark-current level.

3.3.3 Blemish Files

The Blemish File is in 16-bit integer format, and defines blemishes by using vectors of the form (line,samp,CLASS, d_{fw}), where line and samp are the picture coordinates where the blemish occurs, d_{fw} is the DN value at which the pixel saturates at full-well, and CLASS describes the interpolation algorithm to be used to remove the blemish. The vectors are sorted first on line, then on samp.

Two types of blemishes are stored in the Blemish File: 1) permanent blemishes and 2) low full-well pixels. Low full-well pixels are treated as blemishes only when the DN value exceeds the full-well capacity of the pixel. Permanent blemishes are identified by vectors with $d_{fw} = 0$. If $d_{fw} > 0$ then the vector represents a low full-well pixel.

Blemish classification: The blemish classification is used by GALSOS to determine the interpolation algorithm to use in removing the blemishes. Blemishes are classified according to which of the surrounding pixels are good and hence available for interpolation. In order to limit the possible number of permutations, the surrounding pixels are paired, leaving four pairs. If we represent the blemish as a B , and each pair as a number 1 – 4, then we have the following diagram:

$$\begin{array}{ccc} 1 & 2 & 3 \\ 4 & B & 4 \\ 3 & 2 & 1 \end{array}$$

The four least significant bits of the 16-bit CLASS represent each of these pairs in order, such that CLASS = 15 would mean to use all 4 pairs, CLASS = 5 would mean to use pairs 3 and 1, etc. This results in a range of values for CLASS from 0 – 15. If CLASS = 0, then no neighboring pixels are available for interpolation (i.e. it is surrounded by other blemishes or the edges of the image).

The classification scheme is extended to include double-column blemishes B of the form:

$$\begin{array}{ccc} 1 & b & b & 3 \\ 2 & B & b & 2 \\ 3 & b & b & 1 \end{array} \quad \text{or} \quad \begin{array}{ccc} 1 & b & b & 3 \\ 2 & b & B & 2 \\ 3 & b & b & 1 \end{array}$$

where the b represent neighboring blemishes. The resulting CLASS is represented in binary as 10xxx and 11xxx for right and left double-column blemishes, respectively, where the x are 1 or 0 depending on the presence or absence of good pixels at the

corresponding locations. This results in a range of values for *CLASS* from 17–23 and 25–31.

All blemishes of *CLASS* = 0 (termed unclassified blemishes) are set equal to 0 by GALSOS. If a blemish lies on one of the edges of the image, it is assigned *CLASS* = 0.

Note that this blemish classification scheme is oriented towards the removal of single pixel blemishes. The only extended blemishes treated are line blemishes a single pixel in width or column blemishes no more than two pixels in width.

3.3.4 Shutter-Offset File

The shutter-offset file consists of a single record in REAL*4 data format containing the shutter-offset (in milliseconds) for each image line (800 values).

3.3.5 Radiometric Catalog

The Radiometric Catalog contains the location of all radiometric, dark-current, blemish, and shutter-offset files.

For files resident on disk, file locations are specified by disk volume, directory and file name. For files on magnetic tape, file locations are specified by tape volume and file number.

Updating the Catalog: It is possible to edit the Catalog entries to enter new data, and to reflect changes in the location of files. A listing of the Catalog entries may be obtained.

VICAR Program interface to the Catalog: Radiometric correction program GALSOS accesses the correct calibration files by searching the catalog for matching spacecraft-event time, filter position, gain-state, frame-rate, and flags for extended-exposure, summation mode, inverted mode, and blemish protection mode. Outputs consist of the locations for the corresponding radiometric, dark-current, blemish, and shutter-offset files.

4 Radiometric Correction

Radiometric correction is the process of restoring the original scene brightness from the output camera signal. The process consists of applying the inverse light-transfer function

$$E = F^{-1}(D)$$

and scaling the result to some specified radiometric unit. During the correction, any pixel which cannot be calibrated (i.e., permanent blemishes and pixels exposed beyond the limit of a low-full-well column) are cosmetically removed by interpolating over neighboring (corrected) pixels.

The required transformation (inverse light-transfer function) is defined in the radiometric and dark-current files. The blemishes and low full-well pixels are identified in the blemish file.

The Galileo camera is nominally assumed to have a linear response. Each pixel is treated as an independent sensor whose inverse light-transfer function is modeled by a slope term (stored in the radiometric file) and an offset term (stored in the dark-current file). This linear model is used for all radiometric correction performed for systematic processing. The radiometric correction is performed by program GALSOS (figure 5), which applies the inverse function and scales the result to some specified radiometric unit.

4.1 Linear Light-Transfer Function

For the linear light-transfer curve, the inverse function at each pixel coordinate (i,j) is of the form:

$$e(i,j) = z(i,j)(d(i,j) - d_0(i,j))$$

where

$e(i,j)$ is the equivalent lowest gain exposure at pixel (i,j) ,

$d(i,j)$ is the DN value of the pixel,

$z(i,j)$ is the slope term retrieved from the radiometric file, and

$d_0(i,j)$ is the dark-current retrieved from the dark-current file.

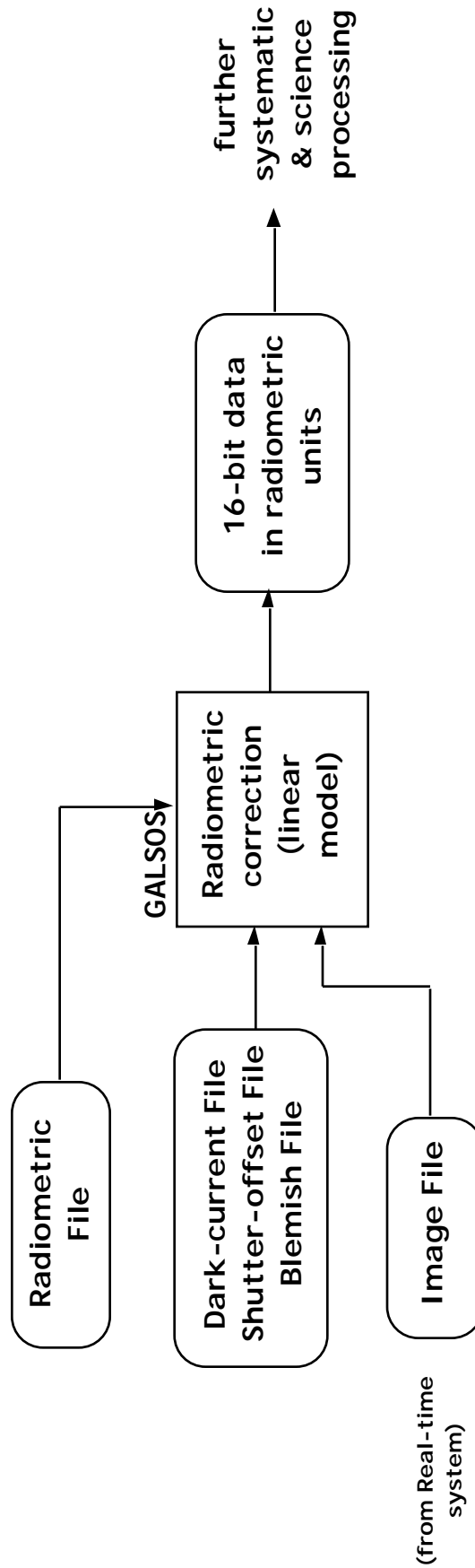


Figure 5: Radiometric correction

4.2 Scaling to Radiometric Units

The radiometric scale of the output picture is specified by a scale factor a_1 in *I/F* units per DN, or a scale factor a_2 in *nanowatts/cm²/steradian/nanometer* per DN.

I/F Units: Let $e(i, j)$ be the computed exposure at pixel coordinate (i, j) resulting from a linear correction. If scale factor a_1 is specified, the output $r(i, j)$ is converted to units of $\frac{1}{a_1} \times (I/F)$, where 10,000 *I/F* units would be produced by normal incidence of sunlight on a Lambert disk at Jupiter's distance from the Sun. See the GALSOS help.

$$r(i, j) = e(i, j) \times \frac{S_1(f)}{a_1 [t - t_0(i)]} \times \frac{D^2}{5.2^2} \quad (24)$$

where

$S_1(f)$ is the conversion factor from ft-Lamberts to *I/F* units,

t is the commanded exposure time in milliseconds,

$t_0(i)$ is the shutter offset at image line i , at the time the image was acquired (in msec),

g is the system gain constant for the gain-state of the image,

g_0 is the system gain constant for the lowest gain-state, and

D is the actual distance from the Sun in A.U.

The conversion factor $S_1(f)$ is a function of filter position f , and assumes a solar illuminated surface at a distance of 5.2 A.U. from the Sun. If scale factor a_2 is specified, the output is converted to radiance units of $\frac{1}{a_2} \times \text{nanowatts/cm}^2/\text{steradian/nanometer}$:

$$r(i, j) = e(i, j) \times \frac{S_2(f)}{a_2 [t - t_0(i)]} \times \frac{1}{g} \quad (25)$$

where $S_2(f)$ is the conversion factor for filter f from foot-Lamberts to units of *nanowatts/cm²/steradian/nanometer* assuming constant scene radiance across the filter bandpass. The scale factor a_2 is typically chosen to be a convenient integer near $S_2(f)/64$ for 16-bit output data. If the scale factor is defaulted, an appropriate scale is automatically selected based on the filter position, exposure, and gain state.

Values for $S_1(f)$ and $S_2(f)$ as a function of filter position f , for the lowest gain-state, and for gain constants g_1 and g_2 (in electrons per DN) are retrieved from tables internal to GALSOS.

5 Geometric Calibration

Figure 6 shows the functional flow of the processing involved in measuring the geometric distortions of the camera. The geometric grid target consists of thin rulings forming a grid of evenly spaced horizontal and vertical lines. The grid intersections are measured to $\pm 10^{-3}$ millimeters on the target itself with a traveling microscope, or to $\pm 2 \mu rad$ when viewed through a collimator using a theodolite. The traveling microscope or theodolite measurements are input into a file via program FIXLOC.

During bench calibration, the target is imaged through a collimator at various orientations (rotations and translations), and at various camera states. For each grid image, the grid intersections are initially located to the nearest pixel via program INTERLOC. Any poorly located intersections may be corrected via INTERLOC or FIXLOC. Program GRIDLOCB is then used to refine the grid locations to sub-pixel accuracy.

Geometric distortion vectors are computed by comparing the actual grid locations with the imaged grid locations via program LOCUS. Before the comparison can be made, the grid intersections within the field-of-view of the image must be extracted from the target measurements via program GETLOC.

Measuring the geometric distortion: The imaged grid locations are compared with the original target measurements by applying a linear transformation which minimizes the RMS residual error between the two sets of data. Two RMS values are computed, the first by allowing the transformation to adjust for differences in scale, translation, and rotation between the two datasets (four degrees of freedom), and the second, by allowing the transformation to correct also for differences in skew and aspect ratio (unconstrained case).

Let (x,y) represent the original measurements of the grid target, and let (u,v) represent the corresponding line-sample coordinates in the imaged grid. The transformation is a linear mapping from the plane of the grid target into the plane of the imaged grid and is of the form:

$$\begin{matrix} u \\ v \end{matrix} = \begin{matrix} a & b \\ c & d \end{matrix} \begin{matrix} x \\ y \end{matrix} + \begin{matrix} e \\ f \end{matrix}$$

The transformation coefficients a, b, c, d, e , and f are computed by fitting the coordinates in the actual grid and imaged grid to this linear equation via the method of least squares.

Let n be the total number of points in each coordinate set, and let (x_i, y_i) and (u_i, v_i) represent the line-sample coordinates for the actual and imaged grid, respectively. Then, for the unconstrained case, the coefficients are chosen such that they minimize the function:

$$= \sum_{i=1}^n \left[(ax_i + by_i + e - u_i)^2 + (cx_i + dy_i + f - v_i)^2 \right]^{\frac{1}{2}}$$

If only four degrees of freedom are permitted, the transformation is constrained by requiring that $a = d$ and $b = -c$ (scale, rotation and offset only).

Adjusting for rotation and offset compensates for the arbitrary orientation of the grid target relative to the camera focal plane. Adjusting for scale compensates for scale differences between the millimeter measurements on the target and the picture scale. The RMS value computed by allowing these four degrees of freedom is a measure of the total geometric distortion of the camera system, excluding measurement errors.

In the unconstrained case, the remaining two degrees of freedom, skew and aspect ratio are also involved. The resulting RMS is a measure of the nonlinear component of the geometric distortion.

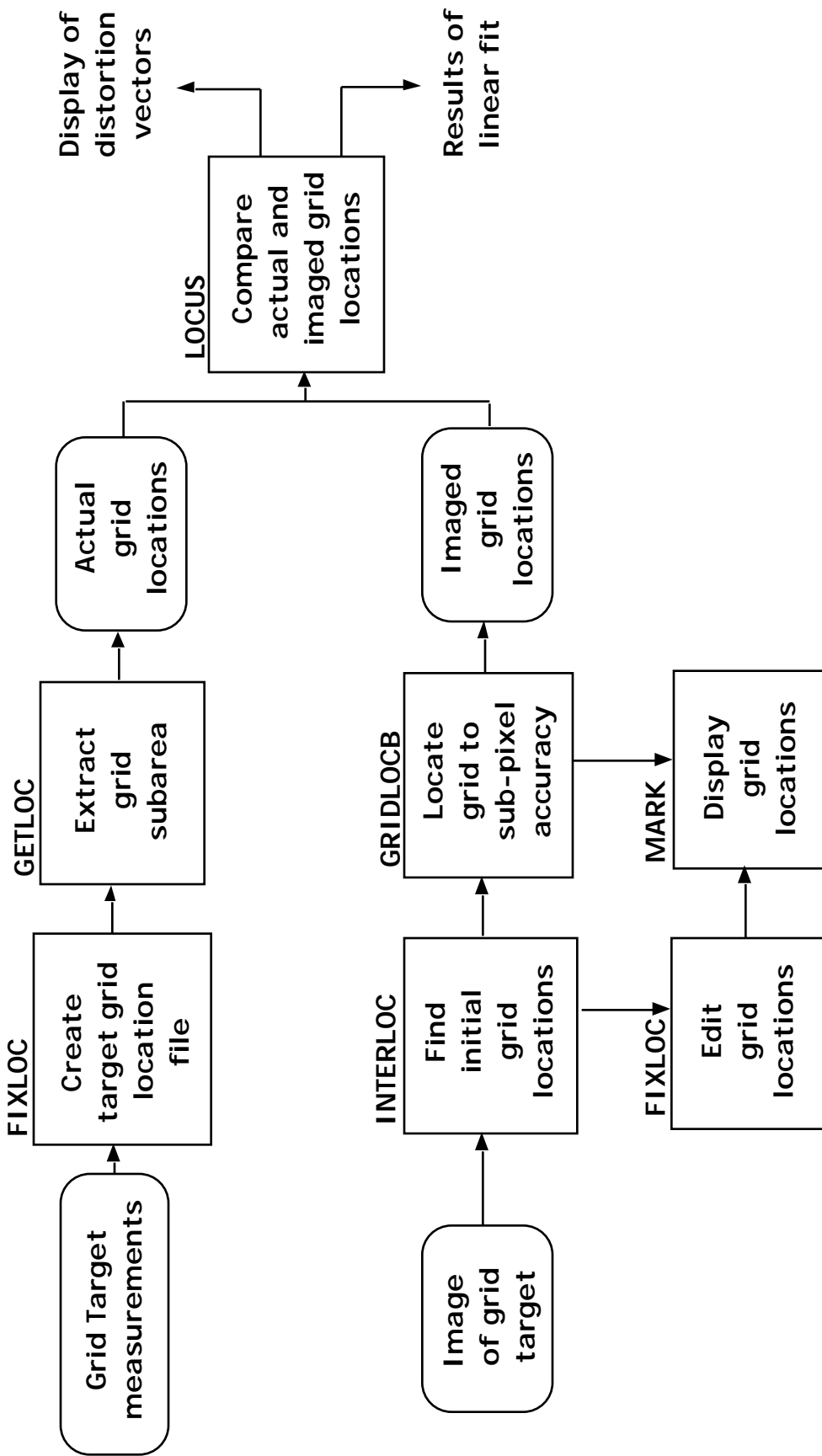


Figure 6: Measuring the geometric distortions of the camera

6 Geometric Correction

Geometric correction is the process of removing the geometric distortion from the output camera signal. Based upon the geometric test results a Geometric Distortion model was developed for the GLL SSI camera.

This model was coded into the VICAR subroutine GLLGCOR. GLLGCOR is called by the VICAR programs associated with image navigation (FARENC, NAV, NAV2), map projection (MAP2, MAP3), and photometric correction (PHOTFUNC).

7 In-Flight Calibration

All radiometric calibration software developed to support pre-launch calibration are also required to support in-flight calibration. This section specifies additional software capabilities required to process calibration data acquired in-flight. Processing requirements in this area are not fully understood at this time and only brief descriptions are presented below. No program descriptions are included. This section is to be expanded as more information becomes available.

7.1 Calibration Plaque Data

The calibration plaque is a flat gray surface mounted on the spinning section of the spacecraft. The spin period is 20 seconds (3 rpm). The plaque is imaged with the spacecraft in the latched configuration using solar illumination. The plaque will thus present a fixed target, but usually with a rapidly changing phase angle.

The plaque is used to acquire light-transfer data so that the radiometric files may be periodically updated. The field-flatness of the plaque may be degraded by contamination from gas emitted by the spacecraft thrusters. Also, individual frames may be contaminated by shadowing from the high-gain antenna. The field-flatness is enhanced because the plaque, as imaged through the 1500 mm optics, is severely out of focus.

The radiometric calibration programs CCDSLOPE and GALGEN require that all input frames be taken under constant illumination. For calibration plaque data, the illumination will vary from frame-to-frame. For absolute radiometric recalibration, constant illumination conditions must be simulated, either by pre-processing the images, or by adjusting the exposure parameters input to these programs.

Recalibration of the relative pixel-to-pixel response and shutter offset can be accomplished using individual plaque flat fields without knowledge of the absolute plaque brightness.

7.2 Star Field Data

Images of star fields acquired through each of the filters are processed to measure the sensitivity and point spread function of the camera. The following processing steps are required:

- Locate centers of the stars to within 0.1 pixels.
- Determine the mean DN and standard deviation of the sky background.
- Compute the parameters of the PSF.
- Plot the radial profile of a star.
- Measure the intensity of the star.

The required functions will be implemented by installing the VMS version of the NOAO IRAF software, consisting of the IRAF command language and a set of software packages designed to support stellar photometry. Enhancements to the IRAF software package may be required to treat smeared star images. Also, software to convert image data from VICAR format to IRAF format and vice-versa are required.

8 Program Descriptions

8.1 Radiometric Calibration Programs

8.1.1 LTGEN

LTGEN is the first of a sequence of programs developed to measure the radiometric properties of the camera at each of a grid of areas, spaced evenly over the camera's field of view. LTGEN is used to create a Light-Transfer File and a Reciprocity File, and to specify the size and location of the areas. The second program, MOMGEN, is used to extract statistical data for each area from each frame of a light-transfer and reciprocity sequence, and to store these data in the Light Transfer and Reciprocity Files. Data analysis routines subsequently access these files to extract camera properties such as:

- System gain constant and read noise (CCDNOISE).
- Light transfer curve slope and offset (CCDSLOPE).
- Shutter offset and camera sensitivity (CCDRECIP).

Execution Statement:

LTGEN IN LTF user-parameters...

LTGEN IN RCP user-parameters...

Input: LTGEN accepts the following inputs:

1. Any image (IN) from the corresponding light-transfer or reciprocity sequence.
2. Commanded exposure times $t_0, t_1, t_2, \dots, t_m$ (in msec) for each exposure level of the sequence.
3. Specifications for the size of the areas and the number of areas to be used.

The areas are normally 20x20 pixels square. For light-transfer sequences, one hundred areas, arranged in 10 rows and 10 columns, are recommended. For reciprocity sequences, 256 areas, arranged in 16 rows and 16 columns are recommended (the finer

grid is used to more accurately measure shutter-offset as a function of image line number).

Output: The LTGEN output is a Light-Transfer (LTF) or Reciprocity File (RCP). The Light-Transfer (or Reciprocity) File contains a header record, followed by one record for each exposure level. The header record contains the size field for each area specified. The exposure records are stored in ascending order, beginning with the extended-exposure dark-current (when present), the normal dark-current, exposure level 1, on up to the highest exposure. The first word of each exposure record contains the exposure time (in msec) for that exposure level.

Processing: The input image is used to define the image format (800x800 for the normal format and 400x400 for summation-mode frames). The user normally specifies the number of rows and columns in the grid, and LTGEN determines the grid spacing to uniformly cover the field-of-view.

The image label of the input image is copied to the LTF or RCP to help identify these files.

8.1.2 MOMGEN

MOMGEN is used to extract statistical data from each frame of a light-transfer (or reciprocity) sequence, and to store these data in the Light-Transfer (or Reciprocity) File. The Light-Transfer File must have been previously initialized via program LTGEN.

Execution Statement:

MOMGEN (D_1, D_2, \dots, D_n) LTF EXPO= t

MOMGEN (D_1, D_2, \dots, D_n) RCP EXPO= t

Inputs: MOMGEN accepts the following inputs:

1. One or more flat-field frames taken at the same exposure setting:

$$D_1, D_2, \dots, D_n$$

2. Commanded exposure time t (in msec) for the given exposure level.

3. Size specifications for the grid of areas, retrieved from the Light-Transfer File (LTF) or Reciprocity File (RCP) header record.

A maximum of 10 flat-field frames may be input at any given exposure setting.

Processing: For each specified $n \times n$ subarea, the following sums are computed:

1. Sum of DNs: $M_1, M_2, M_3, \dots, M_n$
2. Sum of squares of DNs: $M_{11}, M_{22}, M_{33}, \dots, M_{nn}$
3. Sum of cross terms: $M_{12}, M_{23}, M_{34}, \dots, M_{n-1,n}$

where

$$M_k = \sum_{i,j=1}^n d_k(i,j)$$

$$M_{kk} = \sum_{i,j=1}^n [d_k(i,j)]^2$$

$$M_{k-1,k} = \sum_{i,j=1}^n d_{k-1}(i,j)d_k(i,j)$$

and the $d_{k-1}(i,j)$ and $d_k(i,j)$ are the elements of the $n \times n$ subarea for frames D_{k-1} and D_k , respectively.

Output: The computed sums are stored in the corresponding exposure record of the Light-Transfer File (LTF) or Reciprocity File (RCP). Multiple executions of MOMGEN are used to extract and store data for each exposure level into the LTF or RCP.

8.1.3 CCDNOISE

CCDNOISE determines the system gain constant (in electrons per DN) and read-noise floor (in DN) for the camera system.

Execution Statement:

CCDNOISE LTF OUT user-parameters...

Inputs: CCDNOISE accepts the following inputs:

1. Light-Transfer File (LTF).
2. If extended-exposure data is included in the light-transfer sequence, specification of the exposure level at which these data begin.
3. Optional specification of an output plot file to contain signal-to-noise plots of the data.

The Light-Transfer File must have been initialized via LTGEN and loaded with statistical data via MOMGEN.

Processing: CCDNOISE performs the following steps:

1. Read area size specifications from header record of the LTF.
2. Read statistical data from each exposure record of the LTF.
3. For each area, compute the following:
 - Signal and noise $\{\mu_S^{(l)}, \mu_N^{(l)}\}$ for each exposure level.
 - System gain constant k and read noise $READ$.
4. Compute mean value for system gain constant and read noise, and flag all areas deviating by more than 2 sigma from the mean.
5. Re-compute the mean value for system gain constant and read noise, ignoring all flagged area.

For a given $n \times n$ area, and a given exposure level l , the signal and noise terms are computed from the corresponding sums $M_k, M_{kk}, M_{k,k+1}$ retrieved from the LTF. Recall that for frames $D_k^{(l)}$ and $D_{k+1}^{(l)}$, $\mu_k^{(l)}$ is defined to be the mean DN for frame $D_k^{(l)}$ and $\mu_{k,k+1}^{(l)}$ is defined to be the standard deviation for the difference frame $D_{k,k+1} = D_k - D_{k+1}$. If these definitions are applied to the $n \times n$ area instead, then

$$\mu_k = \frac{1}{n^2} \sum_{i,j=1}^n d_k(i,j)$$

$$\mu_{k,k+1} = \frac{1}{n^2} M_k$$

and

$$\sigma_{k,k+1} = \frac{1}{n} \sum_{i,j=1}^n [d_k(i,j) - d_{k+1}(i,j)]^2 - (\mu_k - \mu_{k+1})^2 \quad \frac{1}{2}$$

$$\sigma_{k,k+1} = \frac{1}{n^2} (M_{kk} - 2M_{k,k+1} + M_{k+1,k+1}) - \frac{1}{n^4} (M_k - M_{k+1})^2 \quad \frac{1}{2}$$

The terms $\mu_S^{(i)}$ and $\mu_N^{(i)}$ can then be computed using equations 4 – 6 above. Finally, k and $READ$ can be calculated using equations 8 and 9.

Outputs: CCDNOISE generates the following outputs:

1. Printed listings of:
 - System gain constant and read noise for each area specified.
 - Summary of all areas flagged for bad system gain or read noise.
 - Summary of the mean signal and noise (DN) at each exposure for each area specified.
2. If an output file (OUT) is specified, the centers of all flagged values are stored in a MARK format tiepoint data set.
3. If a PLOT file is specified, a signal vs. noise plot is generated for each of the four corners and for the center of the frame, using data from the good areas in each of these regions. The tabular listing of the plotted points is also printed out.

8.1.4 CCDRECIP

CCDRECIP determines the mean sensitivity (in DN per foot-Lambert-milliseconds) and shutter offset (in msec) for the camera system.

Execution Statement:

CCDRECIP RCP OUT user-parameters...

Inputs: The program accepts the following inputs:

1. A Reciprocity File (RCP) containing statistical data for a grid of areas in the image, for each exposure of a reciprocity sequence.
2. Light cannon luminance's l_i in relative foot-Lamberts, for each exposure $i = 1, 2, 3, \dots, m$.
3. Optional specification of an output plot file to contain reciprocity and shutter offset plots.
4. Optional specification of an output shutter-offset file to contain the shutter-offset as a function of line number.

The Reciprocity File must have been previously initialized via LTGEN and loaded with data via MOMGEN.

Processing: CCDRECIP performs the following steps:

1. Read area size specifications from header record of the Reciprocity File.
2. Read commanded shutter times t_i and area data from each exposure record of the RCP.
3. For each area, compute the following:
 - Mean signal $\mu_s^{(i)}$ for each exposure level $i = 1, 2, 3, \dots, m$.
 - Mean camera sensitivity μ_c and shutter offset t_0 .
4. Compute mean value for μ_c and t_0 by averaging the values for all the areas and flag those areas which deviate by more than 2 sigma from the mean.
5. Re-compute the mean value for μ_c and t_0 , ignoring all flagged area.

For a given $n \times n$ area, and a given exposure level i , the mean signal $\mu_s^{(i)}$ is computed from the corresponding sums M_k from the Reciprocity File, as in CCDNOISE. The terms μ_c and t_0 may then be computed using equations 12 – 14 above.

Outputs: CCDRECIP generates the following outputs:

1. Printed listings of:
 - Sensitivity and shutter offset for each area specified.
 - Summary of all areas with bad values for sensitivity and shutter offset.
 - Mean sensitivity as a function of exposure time.
 - Global value for sensitivity and shutter offset, obtained by combining data from all good areas.
 - Shutter offset as a function of image line number.
2. If an output file is specified, the centers of all flagged values are stored in a MARK format tiepoint data set.
3. If a PLOT file is specified, the following plots are output:
 - μ_s / l vs. t
 - $\mu_s / (lt)$ vs. t
 - $\mu_s / [l(t - t_0)]$ vs. t
 - t_0 vs. image line number
4. If an output shutter-offset file is specified, then a file containing 800 shutter offsets (one for each image line) is generated.

The line dependence of the shutter offset is due to the acceleration of the shutter blades, which travel in a vertical direction. These offsets are calculated by performing a piece-wise linear interpolation through the offsets for all the good areas. These offsets can then be input into CCDSLOPE and GALGEN.

Restrictions:

1. Extended exposure frames should not be included in the reciprocity sequence.

2. The areas must be specified in a regular grid pattern (see GRES parameter in program LTGEN).

8.1.5 CCDSLOPE

CCDSLOPE determines the mean slope μ_c and offset μ_{D_0} of the light transfer curve:

$$D = Cl + D_0$$

where C is the camera sensitivity in relative foot-Lamberts-milliseconds, and D_0 is the dark-current (in DN).

Execution Statement:

CCDSLOPE (LTF,OFFSETS) OUT user-parameters...

Inputs: CCDSLOPE accepts the following inputs:

1. A Light-Transfer File (LTF) containing statistical data for a grid of areas in the image, for each exposure of a light transfer sequence.
2. Shutter-offset file (OFFSETS) as generated by the program CCDRECIP.
3. Luminance l of the light cannon in relative foot-Lamberts.
4. If extended-exposure data is included in the light-transfer sequence, specification of the exposure level at which these data begin.
5. Optional specification of an output plot file to contain plot of mean DN vs. exposure.

The Light Transfer File must have been previously initialized via LTGEN and loaded with data via MOMGEN.

Processing: CCDSLOPE performs the following steps:

1. Read area size specifications from header record of the Light-Transfer File.
2. Read commanded shutter times t_i and area data from the LTF.

3. For each area, compute the following:
 - Mean signal $\mu_s^{(i)}$ for each exposure level $i = 1, 2, 3, \dots, m$.
 - Mean slope μ_c and offset μ_{D_0}
4. Compute mean values for μ_c and μ_{D_0} by averaging the values for all the areas and flag those areas which deviate by more than 2 sigma from the mean.
5. Re-compute the mean value for μ_c and μ_{D_0} , ignoring all flagged areas.

For a given $n \times n$ area, and a given exposure level i , the mean output DN $\mu_D^{(i)}$ is computed from the corresponding sums M_k from the Light-Transfer File, as in CCDNOISE. The terms μ_c and μ_{D_0} may then be computed using equations 18 – 19 above by replacing the d_i by the area means $\mu_D^{(i)}$.

If extended-exposure frames are included in the Light-Transfer File, an adjustment must be made for the differences in dark-current level. This adjustment consists of subtracting the extended-exposure dark-current and adding the normal dark-current to the output DNs for all such frames.

Outputs: CCDSLOPE generates the following outputs:

1. Printed listings of:
 - Mean slope and offset for each area specified.
 - Summary of all areas with bad values for slope and offset.
 - A table listing mean output DN vs. exposure.
 - Global values for slope and offset, obtained by combining data from all good areas.
2. If an output file is specified, the centers of all flagged values are stored in a MARK format tiepoint data set.
3. If a PLOT file is specified, a plot of mean DN vs. exposure is output for each of the four corners, the center of the frame, and for the entire frame, using the data from the good areas in each of these regions.

8.1.6 PICSUM

PICSUM computes the sum of multiple flat-field frames taken at the same exposure level, and optionally checks for saturated samples. The purpose of the program is to consolidate the data from a light-transfer or dark-current sequence for later input to GALGEN or GALSOS.

Execution Statement:

PICSUM (D_1, D_2, \dots, D_n) D user-parameters

Inputs: PICSUM accepts the following inputs:

1. n flat-field frames $\{D_k\}$ taken at the same exposure level.
2. Optional parameters *LSAT* and *HSAT* specifying the low and high saturation thresholds.
3. Optional keyword *ASCALE* specifying automatic scaling of the output.

The input frames may be in byte or 16-bit integer format. However, if the first input contains byte data, then the remaining input frames must contain byte data also. PICSUM will process a maximum of 30 input frames.

Output: The PICSUM output is a summed image D (in 16-bit integer format) where

$$D = D_1 + D_2 + D_3 + \dots + D_n$$

The reason for computing the sum of the frames rather than the average is to minimize quantization errors during subsequent calculations. If automatic scaling is specified, the output is multiplied by $128/n$. The output scale (n or $128/n$) and number of frames summed are inserted in the picture label.

Pixel Saturation Check: If all the input frames are in byte format, PICSUM will optionally check for saturation of the input pixels. Let $d_k(i, j)$ represent the DN value at pixel coordinate (i, j) for input frame D_k . If

$$LSAT < d_k(i, j) < HSAT$$

then $d_k(i, j)$ is accepted as a valid pixel value. If all the DN values at pixel coordinate (i, j) are valid, then their sum is output:

$$d_{out}(i, j) = \sum_{k=1}^n d_k(i, j)$$

If at least half the pixels are valid, then n times the median value is output. Otherwise, -32000 is output to flag the pixel as bad. The saturation check may be suppressed by setting $LSAT = -32000$ and $HSAT = 256$.

If the inputs are in 16-bit data format, then no pixel saturation check is made.

8.1.7 GALGEN

GALGEN generates radiometric files by fitting the data from a light transfer sequence to a linear model of the light-transfer curve.

Execution Statement:

GALGEN ($D^{(0)}, D^{(1)}, D^{(2)}, \dots, D^{(m)}, D^{(ext)}$) (CAL, SAT, ERR, RMS, DC) +
user-parameters

Inputs: GALGEN accepts the following inputs:

1. Flat-field frames from each exposure level of a light-transfer sequence:

$$D^{(0)}, D^{(1)}, D^{(2)}, \dots, D^{(m)}, D^{(ext)}.$$

The frames are entered in order of increasing exposure, beginning with the dark-current $D^{(0)}$. An extended-exposure dark-current $D^{(ext)}$ is appended if extended-exposure mode frames are included in the sequence.

2. The scale of each input frame, as indicated in the picture label.
3. Luminance l of the light cannon in relative foot-Lamberts.
4. Commanded exposure times $t_0, t_1, t_2, \dots, t_m$ (in msec) for each exposure level of the sequence.

5. If extended-exposure data is included in the light-transfer sequence, specification of the exposure level at which these data begin.
6. Shutter-offset file, as output by program CCDRECIP.
7. Specification of the number of data points (n) to use in the initial fit, before applying the low full-well pixel test.
8. Specification of the slope and offset (α_1 and α_0) of the error function used in the low full-well pixel test.
9. A low full-well pixel map defining their locations and thresholds if they are already available.

The input frames may be in byte format, or 16-bit integer format (as output by the program PICSUM).

Outputs: GALGEN creates the following output files:

1. Radiometric file CAL, containing the slope terms of the inverse light transfer function for each pixel (REAL*4 data).
2. SAT file, identifying the low full-well pixels by the DN value at which the pixel reaches full-well (16-bit data).
3. ERR file, containing the maximum absolute difference (in DN) between the input data samples and the fitted curve, at each pixel (16-bit data).
4. RMS file, containing the RMS error for the fit, at each pixel (16-bit data).
5. A dark-current file (DC), containing the offset term of the inverse light transfer function, in 16-bit integer format.

The Radiometric file (CAL) and dark-current file (DC) are used by the program GALSOS to perform radiometric correction on the flight data. The files SAT, ERR, and RMS are used by the program BLEMGEN to identify and classify camera blemishes.

Processing: For each image line, GALGEN computes the exposure $e_i = l(t_i - t_0)$ for each exposure level $i = 0, 1, 2, \dots, m$. The exposure is line dependent because t_0 is a function of line number. For each pixel on the line, the following steps are performed:

1. The DN values for the pixel are retrieved from each of the input frames $D_{(i)}$, $i = 0, 1, 2, \dots, m$. If, at any exposure level i , the input value from $D_{(i)}$ is equal to or greater than 32000 (a flag set by PICSUM), the data from that exposure level and all higher levels are ignored.
2. Normalized DN values $d_0, d_1, d_2, \dots, d_m$ are computed by dividing each summed sample by the corresponding picture scale for each level.
3. The resulting data points $\{(e_i, d_i)\}$ are fit to one of the following models of the light-transfer curve:
 - Slope model: $s = d - d_0 = ce$
 - Linear model: $d = ce + d_0$

For the slope model, coefficient c is computed via equation 22. For the linear model, coefficients c and d_0 are computed using equations 18 and 19.

4. The inverse light-transfer function is computed:

$$e = z(d - d_0)$$

where $z = 1/c$. The slopes z are output to the radiometric file. For the linear model, the d_0 are output to the dark-current file as $128 \times d_0$.

5. If, during the fit, the pixel is determined to have a low-full-well (see Sec 3.2.3), then the DN value at which the pixel saturates (d_{fw}) is stored in the SAT File. If low-full-well pixel information is input to the program, the low-full-well pixel test is bypassed and this information is stored directly into the SAT file.
6. The maximum absolute difference between the input DN values and the computed curve is computed, where

$$\max = \max\{|ce_i + d_0 - d_i|\}$$

and this difference is stored in the ERR File.

7. The RMS error is computed,

$$r_{ms} = \sqrt{\frac{1}{m} \sum_{i=1}^m (ce_i + d_0 - d_i)^2}$$

and stored in the RMS File.

The output files CAL, ERR, RMS, SAT, and DC are images in the same format (800x800) as the flat-field frames.

Let $z(i,j)$ represent the output DN value at pixel coordinate (i,j) for the CAL file. Then

$$z(i,j) = \begin{cases} -1 & \text{for an unsuccessful fit} \\ 1/c & \text{otherwise} \end{cases}$$

The elements of the SAT file are 16-bit integers

$$s(i,j) = \begin{cases} -1 & \text{for an unsuccessful fit} \\ d_{fw} & \text{for low-full well pixels} \\ d_{max} & \text{otherwise} \end{cases}$$

where d_{fw} is the DN value at which the pixel reaches full-well, and d_{max} is a user specified value (usually 32767) indicating a normal pixel.

The elements of the ERR file are 16-bit integers $f(i,j)$, where

$$f(i,j) = \begin{cases} -1 & \text{for an unsuccessful fit} \\ r_{max} & \text{otherwise} \end{cases}$$

The elements of the RMS file are 16-bit integers $r(i,j)$, where

$$r(i,j) = \begin{cases} -1 & \text{for an unsuccessful fit} \\ r_{ms} & \text{otherwise} \end{cases}$$

The DN value -1 is used to indicate a permanent blemish.

The elements of the dark-current file (DC) are 16-bit integers $d(i,j)$, where

$$d(i,j) = \begin{cases} -32768 & \text{for an unsuccessful fit} \\ 128 \times d_0 & \text{otherwise} \end{cases}$$

Extended-exposure data: If extended-exposure mode frames are included in the input light-transfer sequence, then extended-exposure dark-current data must be included as well. An adjustment is made to correct for the difference in dark-current level between extended-exposure mode frames and frames taken at the normal exposures. For the slope model, this adjustment consists of replacing the normal exposure dark-current d_0 by the extended-exposure dark-current (equation 20) for data points d_i taken in extended-exposure mode. For the linear model, the extended-exposure dark-current is subtracted from d_i and the normal dark-current d_0 is added. If all inputs frames used extended-exposure, no adjustment is made, and the output dark-current file (DC) reflects the extended-exposure offset.

8.1.8 BLEMGEN

BLEMGEN accepts as input, the CAL, SAT, ERR, RMS, and DC files generated by the radiometric calibration program GALGEN, and uses these data to automatically identify and classify camera blemishes. The blemishes are recorded in a Blemish File which may later be used by programs GALSOS to remove blemishes from the flight images.

Execution Statement:

BLEMGEN (CAL,SAT,ERR,RMS,DC) BLEM user-parameters

Inputs: BLEMGEN accepts the following inputs:

1. GALGEN output files CAL, SAT, ERR, RMS, and DC (see GALGEN for details).
2. Thresholds *MINSLOPE* and *MAXSLOPE* specifying the valid range for the light-transfer function slope.
3. Thresholds *MINDC* and *MAXDC* specifying the valid range for the dark-current.

4. Threshold *MINSAT* specifying the minimum allowable low full-well saturation DN.
5. Threshold *MAXERR* specifying the maximum allowable absolute difference between the light-transfer data and the fitted curve.
6. Threshold *MAXRMS* specifying the maximum allowable RMS error resulting from the fit.

Outputs: BLEMGEN generates the following outputs:

1. A Blemish File BLEM.
2. Print-out of the following statistical data:
 - Mean and standard deviation of the slope z and dark-current d_0 for all good pixels (i.e., excluding blemishes).
 - Number of permanent blemishes.
 - Number of low full-well pixels.
 - Number of unclassified blemishes.
 - Number of double column blemishes.
 - Total number of blemishes.
 - Histogram of DN values at which low full-well pixels saturate.

The Blemish File is in 16-bit integer format, and defines blemishes by using vectors of the form (line,samp,*CLASS*, d_{fw}), where line and samp are the picture coordinates where the blemish occurs, d_{fw} is the DN value at which the pixel saturates at full-well, and *CLASS* describes the interpolation algorithm to be used to remove the blemish. The vectors are sorted first on line, then on samp.

Two types of blemishes are stored in the Blemish File: 1) permanent blemishes and 2) low full-well pixels. Low full-well pixels are treated as blemishes only when the DN value exceeds the full-well capacity of the pixel. Permanent blemishes are identified by vectors with $d_{fw} = 0$. If $d_{fw} > 0$ then the vector represents a low full-well pixel.

Blemish Identification: A permanent blemish is any pixel which has one of the following characteristics:

1. A light-transfer-function whose slope or offset is excessively high or low.
2. A lower-than-average full well.
3. Light-transfer-function nonlinearities greater than specified thresholds.

The following tests are applied on each pixel of the CCD:

- Let z and d_0 represent the slope and offset terms of the inverse light-transfer function

$$e = z(d - d_0)$$

as extracted from the CAL and DC files, respectively. Then z and d_0 are acceptable if:

$$MINSLOPE < z < MAXSLOPE$$

$$MINDC < d_0 < MAXDC$$

Otherwise, the pixel is identified as a permanent blemish. For the slope model of the inverse function, the thresholds $MINDC$ and $MAXDC$ are ignored.

- Let d_{fw} be the DN value at which full-well is reached, as extracted from the SAT file. Then if

$$d_{fw} < MINSAT$$

the pixel is identified as a permanent blemish. Else, if

$$d_{fw} < 32767$$

the pixel is identified as a low full-well pixel.

- Let $_{max}$ represent the maximum absolute difference (in DN) between the light-transfer data points and the fitted curve, as extracted from the ERR file. Then if

$${}_{max} > MAXERR$$

the pixel is identified as a permanent blemish.

- Let ${}_{rms}$ represent the RMS error resulting from the fit, as extracted from the RMS file. Then if

$${}_{rms} > MAXRMS$$

the pixel is identified as a permanent blemish.

Blemish classification: The blemish classification scheme is described in Section 3.3.3.

Diagnostic Mode: If the keyword *BC* is specified, then the blemish class is replaced by a code which indicates which blemish criteria was used in flagging each blemish:

- 1 = slope was out of range
- 2 = offset was out of range
- 4 = pixel saturation level is less than *MINSAT*
- 5 = maximum error exceeds *MAXERR*
- 6 = RMS error exceeds *MAXRMS*
- 7 = low-full-well pixel

The resulting file is not a valid Blemish File, but is used for diagnostic purposes to indicate which thresholds are being exceeded.

8.1.9 BLEMPIC

BLEMPIC creates images of the permanent blemishes, low-full-well pixels, and total blemishes, as recorded in the Blemish File output by BLEMGEN. The resulting images can be displayed to indicate where the blemishes are spatially located in the image.

Execution Statement:

BLEMPIC BLEM (PBLEM,SAT,TBLEM) NL=800 NS=800

Inputs: BLEMPIC accepts the following inputs:

1. Blemish File (BLEM) as generated by program BLEMGGEN.
2. Integers NL and NS specifying the size of the output image.

Outputs: BLEMPIC generates the following outputs:

1. Image (PBLEM) of the permanent blemishes.
2. Image (SAT) of the low full-well pixels.
3. Image (TBLEM) of all blemishes (TBLEM=PBLEM+SAT).
4. Print-out of the following statistical data:
 - Number of permanent blemishes.
 - Number of low full-well pixels.
 - Number of unclassified blemishes.
 - Total number of blemishes.
 - Histogram of DN values at which low full-well pixels saturate.

Images PBLEM, SAT, and TBLEM are of size $NL \times NS$ (800x800 in the above example) and in byte data format. For summation mode frames, the output image size should be specified as 400x400.

The elements of the image PBLEM are integers $p(i,j)$, such that

$$p(i,j) = \begin{array}{ll} 0 & \text{if pixel (i,j) is not a permanent blemish} \\ n & \text{if the pixel is a permanent blemish} \\ 255 & \text{if the pixel is an unclassified permanent blemish} \end{array}$$

See BLEMGGEN description for the blemish classification scheme.

The elements of the image SAT are integers $s(i,j)$, such that

$$s(i,j) = \begin{cases} 0 & \text{if pixel (i,j) is normal or a permanent blemish} \\ d_{fw} & \text{if the pixel saturates at } d_{fw} \text{ DN} \end{cases}$$

The image TBLEM is similar to PBLEM except that it includes all blemishes (i.e. the union of all permanent blemishes and low-full-well pixels). This image characterizes the case of a high-exposure frame in which all low-full-well pixels are saturated.

8.1.10 Examples

Reciprocity Sequence:

```
LTGEN DC 1 RCP.CLR GRES=(16,20,20) NI=4 +
EXPO=(0,4.167,6.25,8.333,12.5,16.67,25)
```

```
MOMGEN (DC 1,DC 2,DC 3,DC 4) RCP.CLR EXPO=0
```

```
MOMGEN (A1,A2,A3,A4) RCP.CLR EXPO=4.167
```

```
MOMGEN (B1,B2,B3,B4) RCP.CLR EXPO=6.25
```

```
MOMGEN (C1,C2,C3) RCP.CLR EXPO=8.333
```

```
MOMGEN (D1,D2,D3) RCP.CLR EXPO=12.5
```

```
MOMGEN (E1,E2) RCP.CLR EXPO=16.67
```

```
MOMGEN F1 RCP.CLR EXPO=25.
```

```
CCDRECIP RCP.CLR OUT=BADSENSE.DAT +
PLOT=RECIP.DAT OFFSETS=OFFSETS.DAT +
LIGHT=(0,67.6,45.6,35.1,23.7,18.4,11.9)
```

```
MARK (F1,BADSENSE.DAT) BD
```

```
JDISP BD
```

```
DC L PRINT/N OF EED RECIP.DAT
```

The input reciprocity sequence (taken through the clear filter) consists of flat-field frames:

DC1,DC2,DC3,DC4 taken at 0.0 msec exposure (dark-current),

A1,A2,A3,A4 taken at 4.167 msec exposure,

B1,B2,B3,B4 taken at 6.25 msec exposure,

etc.

The flat-field frames must be stored on random-access disk. Extended-exposure frames are not permitted.

The Reciprocity File (RCP.CLR) is initialized by LTGEN. The grid of areas, specified by the GRES parameter, will consist of 256 areas arranged in 16 rows and 16 columns. The size of the areas will be 20×20 pixels. The maximum number of frames input to MOMGEN at any one exposure (*NI*) is 4. The exposure times (in msec), for each exposure level of the sequence, are specified in ascending order (*EXPO* parameter) beginning with the dark-current.

Multiple executions of MOMGEN are used to extract and store data from each exposure level into the RCP. Note that if one of these MOMGEN executions fails, that execution may be rerun without repeating all other executions. Each exposure record of the RCP will contain the data recorded by the last MOMGEN execution at that exposure.

After the entire reciprocity sequence has been processed by MOMGEN, CCDRECIP is executed to determine the camera sensitivity and shutter offset. The light cannon luminance's (in relative foot-Lamberts) must be specified (*LIGHT* parameter) in the same order as the corresponding exposure times in LTGEN. The output file (OUT) will contain the (line,sample) coordinates of all areas whose camera sensitivities differ by more than 2 sigma from the mean. These areas may be displayed by first running program MARK and displaying the resulting file (BD) interactively via the VIDS JDISP command. The plot file (RECIP.DAT) may be printed out using the DCL PRINT command with the *NOFEED* option. The output parameter file OFFSETS.DAT contain the shutter offsets as a function of image line number, and are used as inputs to CCDSLOPE, GALGEN, and GALSOS below.

Light-Transfer Sequence:

```
LTGEN A1 LTF.GRN 'GRID NI=5 +
EXPO=(-1,0,133.33,200,266.67,400,533.33)
```


MOMGEN (DC 1,DC 2,DC 3,DC 4,DC 5) LTF.GRN EXPO=0

MOMGEN (A1,A2,A3,A4,A5) LTF.GRN EXPO=133.33

MOMGEN (B1,B2,B3,B4,B5) LTF.GRN EXPO=200

MOMGEN (C1,C2,C3,C4) LTF.GRN EXPO=266.67

MOMGEN (D1,D2,D3) LTF.GRN EXPO=400

MOMGEN (E1,E2) LTF.GRN EXPO=533.33

MOMGEN (EDC 1,EDC 2,EDC 3) LTF.GRN EXPO=-1

CCDNOISE LTF.GRN OUT=BADNOISE.DAT EXTEXPO=5 +
PLOT=NOISE.DAT

MARK (E1,BADNOISE.DAT) OUT

JDISP OUT

DC L PRINT/N OF EED NOISE.DAT

CCDSLOPE LTF.GRN OUT=BADSLOPE.DAT EXTEXPO=5 +
LIGHT=3.54 PLOT=SLOPE.DAT OFFSETS=OFFSETS.DAT

MARK (G1,BADSLOPE.DAT) OUT

JDISP OUT

DC L PRINT/N OF EED SLOPE.DAT

In this light-transfer sequence (taken through the green filter), the extended-exposure mode data begins at $EXPO = 533.33$ msec. Consequently, in addition to the normal dark-current frames (DC1,DC2,etc.), this sequence includes extended-exposure dark-current frames (EDC1,EDC2,etc.).

The Light Transfer File (LTF.GRN) is initialized by LTGEN. The grid of areas, specified the *GRID* parameter, will consist of 100 areas arranged in 10 rows and 10 columns. The size of the areas will be 20×20 . The maximum number of frames input to MOMGEN at any one exposure is 5. Note that the extended-exposure dark-current is flagged with an exposure time of -1.

Multiple executions of MOMGEN are used to extract and store the data from each exposure into the LTF. Note that if CCDNOISE is to access the LTF, at least 2 frames must be input at each exposure (since frame differences are used to extract the noise).

CCDNOISE is executed to determine the system gain constant and read noise. The *EXTEXPO* parameter specifies that the extended-exposure mode data begins at exposure level 5, where zero exposure is the dark-current. The output file (BADNOISE.DAT) will contain the (line,sample) coordinates of all areas whose gain constants or read noise differ by more than 2 sigma from the mean.

CCDSLOPE is executed to determine the light-transfer function slope and offsets. The input parameter file OFFSETS.DAT was created by CCDRECIP in the previous example. The output file (BADSLOPE.DAT) will contain the (line,sample) coordinates of all areas whose slopes differ by more than 2 sigma from the mean.

The files BADNOISE.DAT and BADSLOPE.DAT may be displayed similarly as in the example for BADSENSE.DAT above. The files NOISE.DAT and SLOPE.DAT may be plotted as in the example for RECIP.DAT above.

Generating calibration files: The same light-transfer data from the previous example may be used to generate the radiometric, dark-current and blemish files:

PICSUM (DC 1, DC 2, DC 3, DC 4, DC 5) DC

PICSUM (A1, A2, A3, A4, A5) A

PICSUM (B1, B2, B3, B4, B5) B

PICSUM (C1, C2, C3, C4) C

PICSUM (D1, D2, D3) D

PICSUM (E1, E2) E

PICSUM (EDC 1, EDC 2, EDC 3) EDC

GALGEN (DC, A, B, C, D, E, EDC) +

(CAL.GRN, SAT, ERR, RMS, DC.GRN) +

EXPO=(0,133.22,200,266.67,400,533.33) +

EXTEXPO=5 LC=3.54 + ERROR=(0,20) SKIP=4 +

OFFSETS=OFFSETS.DAT

BLEMGEN (CAL.GRN, SAT, ERR, RMS, DC.GRN) BLEM.GRN +

MIN SLOPE=0.13 MAX SLOPE=18.2 + MIN DC =3 MAX DC =95 +
 MIN SAT=15 MAX ERR=9 MAX RMS=5

BLEMPIC BLEM.GRN (PBLEM,SAT,TBLEM) NL=800 NS=800

MASKV PBLEM TAPE STRE=(0,1)

MASKV SAT TAPE STRE=(0,1)

MASKV TBLEM TAPE STRE=(0,1)

The light-transfer data are consolidated by PICSUM, resulting in one frame per exposure level. These summed frames are then input into GALGEN, in order of increasing exposure, with the extended dark-current (EDC) appended at the end of the list. The *EXPO* parameter specifies the exposure times corresponding to each of the input frames, excluding the EDC. The *EXTEXPO* parameter specifies that the extended-exposure mode data begins at exposure level 5, where zero exposure is the dark-current. The *NUMB* parameter specifies the number of frames comprising each summed frame. The *ERROR* parameter specifies the slope and offset (α_1 and α_2) of the error function used to detect low full-well pixels. The *SKIP* parameter specifies the number of data points to use in the initial fit (before applying the low full-well pixel test). The output radiometric file (CAL.GRN) and dark-current file (DC.GRN) are used by GALSOS to radiometrically correct the flight data.

The GALGEN output files are input to BLEMGGEN, which generates the blemish file (BLEM.GRN). The blemishes and low full-well pixels may be displayed by executing BLEMPIC and MASKV. The linear stretch specified in MASKV will display the blemishes (any non-zero pixel) as white on a dark background.

If the low-full-well pixels have already been identified, they may be input directly into GALGEN by specifying them as triplets of the form (line,sample, d_{fw}) using the *LFWPIXEL* keyword. This causes the low-full-well pixel test to be suppressed:

GALGEN (DC, A, B, C, D, E, EDC) +
 (CAL.GRN, SAT, ERR, RMS, DC.GRN) +
 EXPO=(0,133.22,200,266.67,400,533.33) +
 EXT EXPO=5 LC=3.54 OFFSETS=OFFSETS.DAT +
 LFWPIXEL=(389,96,219 +

420,269,153 +
 679,351,215 +
 75,466,80 +
 668,520,191 +
 659,578,156 +
 696,594,172 +
 743,673,204 +
 754,790,195)

Displaying the radiometric files:

```
F2 CAL C FUNC="1 /IN1"
  ASTRTCHR C BYTE PERC=1 EXCL=(-9999,0)
```

The input radiometric file CAL contains the slope terms z of the inverse light-transfer function $e = zd + z_0$. Program F2 is first used to invert the slopes ($c = 1/z$). The resulting slope terms c are scaled to byte data using program ASTRTCHR, which applies a histogram stretch. All negative slopes (i.e. blemishes) are excluded when determining the stretch limits. A hardcopy of the output file (BYTE) may then be obtained via MASKV.

The ratio of two radiometric files may be displayed as follows:

```
F2 (CAL1,CAL2) R FUNC="IN1 /IN2"
  ASTRTCHR R BYTE PERC=1 EXCL=(-9999,0)
```

Displaying the dark-current files:

```
FIT DC BYTE PERC=1 'BYTE EXCL=(-32768,-32767)
```

The dark-current file DC is scaled from 16-bit integer format to byte data via program FIT, which applies a histogram stretch. Blemishes, which are encoded as -32768 DN, are excluded from the stretch limit calculation.

8.2 Radiometric Correction Programs

8.2.1 GALSOS

GALSOS is used to radiometrically correct the flight images by using a linear model of the light-transfer function. GALSOS will automatically remove the camera blemishes by interpolating over neighboring pixels.

Execution Statement:

GALSOS IN OUT user-parameters

Inputs: GALSOS accepts the following inputs:

1. Image (IN) to be radiometrically corrected, in byte data format.
2. Radiometric File as generated by program GALGEN.
3. Dark-current file.
4. Shutter offset file as generated by program CCDRECIP.
5. Blemish File as generated by program BLEMGEN.
6. Location of the radiometric, dark-current, blemish, and shutter offset files, either via access to the Radiometric Catalog or via user-specification.
7. Specification of the output radiometric scale via one of the following:
 - Scale factor a_1 in l/F units per DN.
 - Scale factor a_2 in $nanowatts/cm^2/steradian/nanometer$ per DN.

Output: The GALSOS output is a radiometrically corrected image in 16-bit integer format. The radiometric, dark-current, blemish, and shutter offset files used are identified in the output picture label. The radiometric scale is also inserted in the picture label.

Processing: GALSOS will automatically select the correct radiometric, dark-current, blemish, and shutter offset files (via the Radiometric Catalog) by matching the time, filter, and gain-state of the input frame and checking for extended-exposure, summation mode, inverted mode, and blemish protection mode. Alternatively, the user may specify these files via parameter input.

GALSOS performs the following processing steps on each pixel:

1. If the pixel is identified in the blemish file as a low full-well pixel then the pixel is checked for full-well saturation. Let d_{in} be the input DN value of the pixel and d_{fw} is the full-well saturation DN as retrieved from blemish file. If:

$$d_{in} > d_{fw}$$

then the pixel is flagged as a blemish internally to be interpolated over.

2. The radiometric correction is applied:

$$e = z(d_{in} - d_0)$$

where z is retrieved from the radiometric file and d_0 is retrieved from the dark-current file.

3. Pixels which have been flagged internally as blemishes are interpolated over.
4. The exposure e is scaled to the specified radiometric units (see section 4.2).

For each filter (f), GALSOS contains conversion constants $S_1(f)$ and $S_2(f)$, where $S_1(f)$ is the conversion factor from ft-Lamberts to I/F units, and $S_2(f)$ is the conversion factor from ft-Lamberts to $\text{nanowatts/cm}^2/\text{steradian/nanometer}$ assuming the scene radiance is constant over the filter bandpass. A time history of the GALSOS conversion constants and gain constants is maintained.

8.2.2 GALSOSI

GALSOSI is used to reverse the radiometric correction process and restore the raw (uncorrected) image data. GALSOSI may be applied to data which has been corrected using a linear light-transfer function.

Execution Statement:

GALSOSI IN OUT user-parameters

Inputs: GALSOSI accepts the following inputs:

1. A radiometrically corrected image (IN) in 16-bit data format, as output by program GALSOS, or in byte data format, as output by program GALSCALE.
2. Radiometric File as generated by program GALGEN.
3. Dark-current file.
4. Shutter-offset file as generated by program CCDRECIP.
5. Blemish File as generated by program BLEMGEN.
6. Location of the radiometric, dark-current, blemish, and shutter offset files, either via access to the Radiometric Catalog or via user-specification.

The radiometric, dark-current, shutter-offset, and blemish files input must match the files used in the radiometric correction process, as identified in the input image label.

Output: The GALSOSI output is a raw (uncorrected) image in byte data format.

Processing: GALSOSI will automatically select the correct radiometric, dark-current, blemish, and shutter offset files (via the Radiometric Catalog) by reading the file identifiers from the input image label. Alternatively, the user may specify these files via parameter input.

GALSOSI performs the following processing steps on each pixel:

1. The pixel is converted to units of foot-Lambert-milliseconds by reversing the radiometric scaling applied by GALSOS (see section 4.3), as indicated in the input picture label.
2. The radiometric correction is reversed:

$$d_{out} = \frac{1}{z} e + d_0$$

where z is retrieved from the radiometric file and d_0 is retrieved from the dark-current file.

Pixels flagged as blemishes in the blemish file are output as 0 DN. Low-full pixels are treated as normal pixels.

8.2.3 Examples

Applying a radiometric correction:

GALSOS A B DC =DC. GRN CAL=CAL. GRN IOF =1 .

GALSOS is used to radiometrically correct frame A. The *DC* and *CAL* parameters are used to specify the dark-current and radiometric files, respectively. The scale of the output picture (B) will be 1 *I/F* unit per DN. If the *DC* and *CAL* parameters are defaulted, GALSOS will automatically select the dark-current and radiometric file via the Radiometric Catalog.

Reversing the radiometric correction:

```
GALSOSI B A DC=DC.GRN CAL=CAL.GRN BLEM=BLEM.GRN +  
OFFS=OFFSETS.DAT
```

The GALSOS output B of the previous example is input and the raw (uncorrected) image A is restored. The dark-current, radiometric, blemish, and offset files are specified.

8.3 Geometric Calibration Programs

8.3.1 INTERLOC

INTERLOC accepts as input, an image of a geometric grid target. INTERLOC determines the (line,sample) locations of the grid intersections to within one or two pixels. GRIDLOCB may subsequently be used to refine the grid locations to 0.1 pixel accuracy.

Execution Statement:

```
INTERLOC GRID (OGRID,LOC,WORK) user-parameters...
```

Input: INTERLOC accepts the following inputs:

1. An image of a geometric grid target (GRID) in byte data format.
2. The number of horizontal and vertical grid rulings visible in the image.
3. The top (line,sample) starting position of each vertical ruling and the left (line,sample) starting position of each horizontal ruling.

The grid rulings are assumed to be dark lines (low DNs) on a light (high DNs) background. Rotations in the target should not exceed 20 degrees.

Outputs: INTERLOC generates the following outputs:

1. An output image (OGRID) consisting of the input image with the track of each horizontal and vertical trace overlaid as 255 DN.

2. The (line,sample) coordinates of the measured grid intersections (LOC) in MARK format.
3. A scratch dataset (WORK) used for intermediate storage.

The grid coordinates are ordered from left-to-right and from top-to-bottom in the image, and follow the path of the horizontal grid rulings.

Operation: INTERLOC will operate in both interactive and non-interactive modes. In non-interactive mode, the (line,sample) coordinates of the top and left starting positions of each grid ruling must be entered via parameters. In interactive mode, the image is displayed on a video monitor and the top and left starting positions are selected by the user via a trackball driven cursor. The interactive mode also permits the user to correct the positions of specific intersections by selecting them via the cursor.

Processing: Beginning at the specified left starting positions, INTERLOC traces each horizontal grid ruling in a series of one-pixel steps to the right. The direction of each step is determined by the DN values of the three neighboring pixels to the right. The step is always in the direction of the pixel with the lowest DN. The tracing of the vertical grid rulings is analogous. The grid intersections are then determined by scanning the horizontal and vertical tracings for matching coordinates.

8.3.2 FIXLOC

FIXLOC accepts as input the (line,sample) coordinates of the grid intersections, as measured by INTERLOC or GRIDLOCB. FIXLOC may be used to edit the grid intersection by inserting, deleting, or changing the coordinates for specified intersections. FIXLOC also generates a listing of the grid locations.

Execution Statement:

```
FIXLOC LOC OLOC user-parameters...
```

Input: FIXLOC accepts the following inputs:

1. A grid location file (LOC) in MARK format, as generated by INTERLOC or GRIDLOCB.

2. Insert, delete, and/or change parameters for editing specific grid intersections.

Outputs: FIXLOC generates the following outputs:

1. Optional output file to contain the edited grid locations (OLOC).
2. Printed listing of the coordinates of each grid intersection.

If the output file and edit parameters are omitted, a printed listing of the input grid locations will be generated.

8.3.3 GRIDLOCB

GRIDLOCB locates the grid intersections in an imaged geometric grid target to sub-pixel accuracy.

Execution Statement:

GRIDLOCB (F,TF,LOC) OLOC NH= n NV= m

Input: GRIDLOCB accepts the following inputs:

1. A filtered version of the grid target image (F).
2. A filtered version of the transpose of the grid target image (TF).
3. A grid location file (LOC) in MARK format, as generated by program INTERLOC.
4. Integer values n and m specifying the number of horizontal and vertical grid locations visible in the image.

The grid target image must be filtered prior to input to GRIDLOCB. The first filtered version (F) may be generated via program FILTER using a 49×7 filter with identical weights 3,2,1,0,-1,-2,-3 for each line. This filter will enhance the vertical grid rulings, such that pixels immediately to the left of each grid ruling have positive DN values and pixels immediately to the right are negative. The second filtered version (TF) is generated by first transposing the image (using program FLOT with keyword TRANS to reflect the image about the diagonal joining the upper-left and lower-right corners)

and applying the filter above to enhance the horizontal grid rulings. Both F and TF must be in 16-bit integer format to preserve negative DN values.

Output: The GRIDLOCB output consists of a grid location file (OLOC), in MARK format, containing the refined grid locations.

Processing: The position of a grid intersection is refined by locating the horizontal and vertical rulings within a 7×7 area centered about the nominal position (as input from LOC). To locate the vertical ruling, GRIDLOCB searches each line segment of the 7×7 area for a zero DN value or a positive to negative transition (using the filtered version F). If a positive to negative transition is detected, the zero DN point is linearly interpolated. A least-squares fit is applied through the seven zero DN points to derive a (local) linear equation for the center of the vertical ruling. The center of the horizontal ruling is similarly located using the transposed filtered version TF, and its linear equation derived with line and sample interchanged. The intersection is then solved for simultaneously.

GRIDLOCB will reject an intersection if its initial or final position is outside the image or if either vertical or horizontal grid rulings cannot be located. Rejected intersections are flagged as $(-99.0, -99.0)$.

8.3.4 GETLOC

GETLOC accepts as input the theodolite or traveling microscope measurements of the geometric grid target. GETLOC is used to extract the measurements for a specified area of the target.

Execution Statement:

```
GETLOC LOC OLOC SR=r SC=s NR=n NC=m
```

Input: GETLOC accepts the following inputs:

1. The original target measurements (LOC) input as a MARK format file.
2. Integer values r,s,n,m specifying the starting row, starting column, number of rows, and number of columns for the subarea to be extracted.

The input file LOC is assumed to contain the (line,sample) coordinates for a fixed array of grid intersections.

Outputs: The GETLOC output consists of a tiepoint file in MARK format containing the grid locations for the specified subarea.

8.3.5 LOCUS

LOCUS is used to compare two or more sets of coordinates by displaying their differences. The coordinate sets may be linearly fitted to adjust for differences in scale, rotation, etc. Alternatively, LOCUS may be used to display the coordinates from a single set.

Execution Statement:

LOCUS (LOC1,LOC2,...) OPIC SIZE user-parameters...

Input: LOCUS accepts the following inputs:

1. One or more sets of (line,sample) coordinates input via MARK format files (LOC1,LOC2,...)
2. Optional specification of scaling and translation of the input coordinate sets.
3. If more than one set is input, specification of the operational mode (*HOMOGENEOUS*, *PREFER*, or *DETREND*).
4. For *PREFER* or *DETREND* modes, specification of the reference coordinate set and optional linear transformation(s).
5. Optional specification of an exaggeration factor for scaling the vector differences between coordinate sets.

Each input file is assumed to contain one set of coordinates. Up to ten input files may be used. The maximum total number of input coordinates is approximately 15000.

Outputs: LOCUS generates the following outputs:

1. Image display (OPIC) of the differences between the sets of coordinates.

2. Printed listings of the linear transformation coefficients resulting from various least-square fits.

The output image contains small marks on a zero DN background. A mark is produced in the output image for each point in each set of coordinates. Each mark is a five-pixel + sign. The marks for each given coordinate set all have the same DN. The first set of coordinates have the lowest DNs, and the DNs increase linearly to 255 for the successive sets of coordinates. Thus the last set of coordinates always has a DN of 255.

No mark is generated for coordinates not within the boundaries of the output image. The output image has byte data format. Its size is determined by the *NL* and *NS* parameters of the VICAR size field (*SIZE*).

Operation: *LOCUS* has four modes of operation which are selectable by keywords: (1) single-frame, (2) *HOMOGENEous*, (3) *PREFER*, and (4) *DETREND*.

In each of these four modes, the coordinates may be optionally pre-processed to adjust for scale and offset differences. Independent scaling may be performed in the line and sample directions.

If only one set of coordinates is input (single-frame mode), the output is nothing more than a graphic display of each of the input points unprocessed except for optional translation and scaling.

The remaining three modes are multi-frame modes which compare two or more sets of coordinates. Each set must have the same number of points (coordinates). All comparisons between sets assume that corresponding points within the different sets appear in the same order.

The output is a display of vector differences between each of the sets of coordinates and a reference set. In the *HOMOGENEous* mode, the reference set is the computed mean of the coordinate sets. In the remaining two modes (*PREFER* and *DETREND*), one of the input coordinate sets is designated as the reference set. The reference set is displayed as in the single-frame mode. The remaining sets are displayed as displacements from the reference set. The displacement for a given input point is the vector difference between the its coordinates and the coordinates of the reference point, multiplied by the specified exaggeration factor. The reference set is not displayed in the *HOMOGENEous* mode.

In the *PREFER* and *DETREND* modes, a transformation may be applied to each of the non-reference coordinate sets to correct for linear differences between each set and

the reference set. This transformation is applied prior to computing the vector differences.

Let (x,y) represent a point in a given coordinate set, and let (u,v) represent the corresponding point in the reference set. The transformation is a linear mapping from the plane of the given coordinate set into the plane of the reference set and is of the form:

$$\begin{matrix} u & = & a & b & x & + & e \\ v & & c & d & y & + & f \end{matrix}$$

The transformation coefficients a,b,c,d,e , and f are computed by fitting the points in the given coordinate set and reference coordinate set to this linear equation via the method of least squares.

Let n be the total number of points in each coordinate set, and let (x_i,y_i) and (u_i,v_i) represent the line-sample coordinates for the given set and the reference set, respectively. Then the coefficients are chosen such that they minimize the function:

$$= \sum_{i=1}^n \left[(ax_i + by_i + e - u_i)^2 + (cx_i + dy_i + f - v_i)^2 \right]^{\frac{1}{2}}$$

Various constraints may be placed on the transformation to restrict it to translation and rotation in the plane and scaling. The constraints are specified by optional keywords as follows:

- **NOCHANGE:** Identity mapping ($a = d = 1$ and $b = c = e = f = 0$).
- **OFFONLY:** Translation only ($a = d = 1$ and $b = c = 0$).
- **ROTOFF:** Rotation and offset only ($a = d$, $b = -c$ and $a^2 + b^2 = 1$).
- **MAGOFF:** Scale and offset only ($a = d$ and $b = c = 0$).
- **MAGROTOF:** Scale, rotation, and offset ($a = d$ and $b = -c$).
- **UNCONSTR:** Unconstrained (all six degrees of freedom allowed).

The *PREFER* mode is used to compare several coordinate sets to the reference set. One of the above transformations may be applied to each of the sets. In the resulting output, the reference set is displayed in its true position and each of the remaining sets are displayed as displacements from the reference.

The *DETREND* mode is used to compare two coordinate sets (a reference set and a non-reference set) after applying several different transformations to the non-reference set. In the resulting output, the reference set is displayed in its true position and the vector differences remaining after applying each transformation is displayed as displacements from the reference. Thus the *DETREND* mode can be used to try a variety of transformations on one non-reference set, while the *PREFER* mode can be used to try one transformation on a variety of non-reference sets.

In multi-frame modes, LOCUS produces printed listings in addition to the output image. Each non-reference set is compared with the reference set. (In the homogeneous mode, the last coordinate set input is arbitrarily used as the reference set for this purpose.) For each of the transformations described above, the coefficients a, b, c, d, e , and f resulting from a least squares fit are printed out, together with the RMS error.

8.3.6 Examples

Creating the object space grid file: Assume that a 27×27 grid target is used:

```

GEN G1 1 729 LINC=0 SINC=0 'REAL
FIXLOC G1 G2 CHANGE=(1,.101,.102,2,.101,.202,3,.102,.302,...)
FIXLOC G2 G1 + CHANGE=(28,.201,.102,29,.201,.202,30,.201,.302,...)
FIXLOC G1 G2 + CHANGE=(55,.301,.102,56,.301,.202,57,.301,.303,...)
.
.
.
FIXLOC G2 G1 +
    CHANGE=(723,2.701,.102,724,2.701,.202,725,2.701,.303,...)
CFORM G1 OSGRID SO=(1000.,0.)
FIXLOC OSGRID NC=27
LABEL-DELETE OSGRID TASKS=GEN
LABEL-ADD OSGRID +

```



```
ITEMS="TARGET=123456,COMMENT='Object-Space Target +
Measurements'"
```

Program GEN is used to create a file (G1) for the actual target measurements. Then a series of FIXLOC steps are used to enter the actual target measurements (in millimeters) into this file. The measurements are scaled to units of microns via CFORM and FIXLOC is then used to print out a map of the resulting object space grid location file (OSGRID). FIXLOC may also be used to correct any errors in entering the coordinates. Program LABEL is used to identify the file.

Creating the image space grid file: Assume that the input grid image (PIC) contains 20 grid rows and 21 grid columns. The grid intersections are first located interactively to within one or two pixels via program INTERLOC:

```
INTERLOC PIC (OPIC, GRID, WORK) NH=20 NV=21 'INT
MARK (PIC, GRID) OUT
JDISP OUT
FIXLOC GRID GRID1 NC=21 parameters...
```

INTERLOC traces the path of each grid row and grid column (see *HSEARCH* and *VSEARCH* commands) and displays the results in OPIC. The resulting grid intersections are stored in GRID. Program MARK is used to indicate the located intersections by overlaying squares on the input image. The VIDS JDISP command is used to display the results and to measure the coordinates (via the cursor) of any poorly located intersections. FIXLOC is then used to correct any errors. Several iterations of MARK, JDISP, and FIXLOC may be required before all intersections are located correctly.

```
FILTER PIC FPIC PARMS=FP
FLOT PIC TPIC 'TRAN
FILTER TPIC FTPIC PARMS=FP
GRIDLOC B (FPIC, FTPIC, GRID1) G NH=20 NV=21
MARK (PIC, G) OUT
JDISP OUT
FIXLOC G ISGRID DELETE=(21,31)
```

Program FILTER is used to enhance the (vertical) grid columns and to store the results in FPIC. The parameter file FP is assumed to contain the following:

```
'HALF NLW=49 NSW=7 'ASYM
WEIGHTS=( 3,2,1,0,-1,-2,-3 +
          3,2,1,0,-1,-2,-3 +
          3,2,1,0,-1,-2,-3 +
          . . .
          . . .
          . . .
          3,2,1,0,-1,-2,-3)
```

The input grid image (PIC) is then transposed and filtered to enhance the grid rows. The resulting filtered images (FPIC, FTPIC) and initial grid intersections (GRID1) are input into GRIDLOCB, which refines the intersections to sub-pixel accuracy. MARK and the VIDS JDISP command are used to display the results. FIXLOC may be used to delete spurious intersections. The final image space grid location file (ISGRID) should be displayed in hardcopy via MARK and MASKV.

Measuring the geometric distortion:

```
GETLOC OSGRID OSG SR=3 SC=5 NR=20 NC=21
LOCUS (ISGRID, OSG) OPIC 1 (1,1,800,800) +
      (PREFER=1, COORDS=2, MAGROTOF)
LOCUS (ISGRID, OSG) OPIC 2 (1,1,800,800) +
      (PREFER=1, COORDS=2, UNCONST)
```

Program GETLOC is used to extract the subset of the object space grid intersections which correspond to the area visible in the imaged grid. In this example, the subarea begins at row 3 and column 5. Program LOCUS is then used to compare the image space grid (ISGRID) with the object space grid (OSG). The image space grid is used as the reference (*PREFER=1*) so the resulting distortion vectors are scaled to pixels in image space. Two LOCUS runs are used to measure the total (*MAGROTOF*) and non-

linear (*UNCONSTR*) distortions. Hardcopies of the final MARK output and of the LOCUS output files (OPIC1,OPIC2) should be obtained via MASKV.