IUE IMAGE PROCESSING
OVERVIEW AND
MATHEMATICAL DESCRIPTION

Prepared for

GODDARD SPACE FLIGHT CENTER

By

COMPUTER SCIENCES CORPORATION

Under

Contract NAS 5-11999
Task Assignment 852

Prepared by:

P. M. Perry 10/17 Date
IUE Image Processing
Resident Astronomer

B. E. Turnrose 10/17/77 Date
IUE Image Processing
Resident Astronomer

Approved by:

Edgar M. Greville 10/17/77 Date
E. M. Greville
Section Manager

Richard F. Williams 10/18/77 Date
R. Williams
Department Manager

ABSTRACT

This document provides an overview of the IUE image processing operation and a mathematical description of those aspects of the operation which directly affect the scientific content of IUE spectral data. Sufficient detail is included to allow the individual guest observer to assess the effects of the image processing algorithms on the original data. This is a working document which will be updated to include all relevant programs as they become available.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Figure

SECTION 1 - INTRODUCTION

1.1 BACKGROUND

The International Ultraviolet Explorer (IUE) spacecraft will be located in a geo-synchronous orbit providing continuous contact with the operations center located at the Goddard Space Flight Center (GSFC). The IUE's scientific payload is a 45-centimeter diameter telescope using an echelle spectrograph for ultraviolet astronomy in the spectral region between 1150 and 3200 Å. This instrument will be utilized to obtain UV spectra of astronomical objects. To accomplish mission objectives, a ground observatory has been established at GSFC consisting of two hardware/software computer systems. The Operations Control Center (OCC) System consists of a Xerox Sigma-5 computer and accompanying command and control software supporting the mission. The Image Processing System will consist of the Experiment Display System (EDS), a Sigma-9 computer and its accompanying IUE Spectral Image Processing System (IUESIPS) software. The remainder of this document is devoted to describing capabilities which affect the image processing system exclusively.

1.2 OBJECTIVES

A raw IUE spectral image consists of a 768-pixel by 768-pixel square array with pixel values (or data numbers, DN) ranging from 0 to 255. The primary objective of this document is to provide a description of the process by which the raw input image, along with the necessary ancillary information, is reduced to generate a corrected output spectral image, a spectral intensity tracing, and/or other output products. Processing functions and individual routines dealt with in this document are treated with sufficient mathematical rigor to enable the investigator to determine for himself the effect of these schemes and routines on his unreduced image data. No attempt is made at instructing the reader in the use of the processing language or in operation of any of the image processing peripherals. Information in these areas may be found elsewhere[1].

1.3 SCOPE

In this document, emphasis is placed on those aspects of the image processing data reduction operation which address the scientific content of the data. Operations whose sole purpose is to move data about for computational expediency are not treated at any great length. Furthermore, within several areas, such as the reduction of low-dispersion spectra, software is currently under development and thus is not included herein. The most significant programs are discussed in detail in the following chapters; quick reference functional descriptions of all programs are given in Appendix A.

SECTION 2 - IUE IMAGE PROCESSING OVERVIEW

2.1 OPERATIONAL ENVIRONMENT

Two interactive consoles that interface with the image processing computers and the IUESIPS software are provided in the Image Processing Control Center (IPCC). The first, an ADDS terminal, gives the image analyst the capability to select and edit the image processing sequence prior to task initiation. The second terminal, an integral part of the Experiment Display System (EDS), incorporates a color Cathode Ray Tube (CRT) controlled by a minicomputer and allows the image analyst to monitor and control the actual process in near-real time. In this way, image processing errors can be identified much sooner and proper corrective measures taken with minimal increase in processing time. There are only two hands-on users of this system, the Image Processing Resident Astronomer and the Image Processing (IP) Specialist. A large portion of the image processing will be carried out routinely by the specialist. In those processing areas, however, where specific scientific judgement is required, the guest observer or resident astronomer will provide the necessary guidance.

2.2 PROCESSING SYSTEM FLOW

A complete discussion of all IUE processing areas and their associated requirements is beyond the scope of this document. To provide an added dimension to the discussion presented in this document, however, a hardware/software interface diagram depicting the spacecraft-to-observer data flow is included as Figure 2-1. In addition to providing a list of the system's input and output, it shows the relationship between the two major computer systems involved, the operating system environments, and the utilization of the attached peripherals.

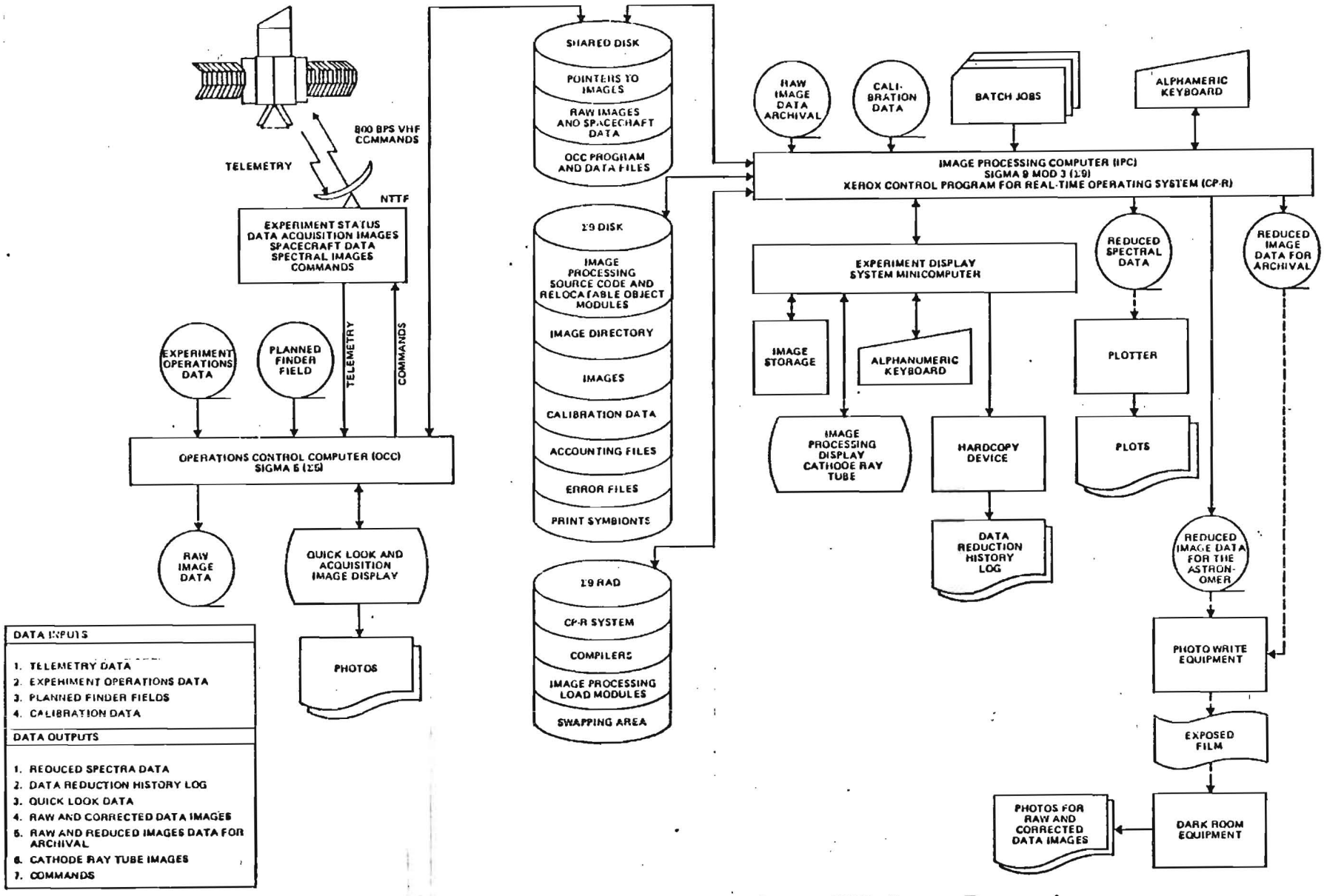

Figure 2-1. The International Ultraviolet Explorer (IUE) Image Processing Data Flow: Spacecraft to Astronomer

(Courtesy of R. Durachka GSFC)

2.2.1 IUESIPS Image Processing

In order to perform a specific function using the IUE Spectral Image Processing System (IUESIPS), the IP specialist creates a processing scheme by selecting and ordering the necessary application programs from a program library. All application programs are general purpose in nature--they are written to handle a wide range of image constraints. Specific constraints for the task at hand are conveyed to the application program by means of parameters which are supplied at scheme creation time.

A label is associated with, and is an integral part of, each image. The label contains three types of information: 1) information concerning the image size, 2) the scientific spacecraft information, and 3) image processing information. The image processing portion of the label contains user supplied information and a system generated history label. The history label provides a permanent record of the image processing operations performed on that image. All input/output processing, label processing, label processing, and routine bookkeeping are handled automatically by IUESIPS.

2.2.2 IUE Experiment Display System (EDS)

Under the IUE image processing configuration, the Experiment Display System (EDS) can perform two distinct functions. 1) The EDS has its own resident image storage disk and software to perform basic image processing functions. The EDS can thus be used as a stand-alone quick-look processor. 2) The IPCC EDS is also linked to the IUE Sigma-9 image processing computer. In this mode it is used to monitor IUESIPS image processing tasks. This is the means by which the image processing specialist can monitor images during the processing, respecify parameters, and restart the processing sequence at any

point in the IUESIPS scheme. The EDS stand-alone capabilities permit the user to:

- display one of two stored images,
- display the full histogram of either image,
- generate a histogram of a subset of a displayed image,
- select one of eight intensity transfer functions which relate the image DN values to the 16 pseudocolors,
- select one of four preprogrammed pseudocolor sets, each containing 16 pseudocolors,
- display an intensity plot of the DN values along a line between any two points in the image,
- expand portions of an image up to nine times,
- translate portions of an image in both X and Y,
- draw a vector between any two selected points of an image.

2.2.3 The Photowrite System

The Optronics Photowrite System is used to produce photographic images on film from magnetic tape. IUE spectral images are written to magnetic tape by an applications program on the Sigma-9. This tape is next brought to the IPCC where it is read by the Photowrite minicomputer. The digital signals received at the input of the writing system are converted to an analog current which is used to modulate a light emitting diode (LED) on an optical carriage system. This carriage is moved along kinematic mounts by a stepping-motor-driven lead screw. A removable film cassette drum is attached directly above the optical carriage. Thus drum rotation provides motion along one scan axis (X) while carriage translation provides scan motion along the other (Y) axis. When the writing process is complete, the film cassette drum is removed and the film is processed by conventional methods in an automatic processor.

SECTION 3 - IMAGE PROCESSING FUNCTIONS

3.1 OVERVIEW OF FUNCTIONS

The IUE image processing system is designed to have the capability to perform a number of separate but related tasks which are referred to as image processing functions. Each image processing function is a generalized operation which may be applied in a variety of contexts; the coordinated performance of the required functions in the proper contexts constitutes the execution of the overall image processing task.

The IUE image processing functions may be conveniently categorized as follows.

1. Transmission error correction (T/M repair)
2. Noise removal (periodic, random)
3. Geometric distortion correction
4. Photometric calibration (SEC response)
5. Background removal (dark current, null, sky, etc)
6. Wavelength determination
7. Spectral intensity determination (I (λ) table generation, or "data extraction")
8. Standard star calibration (system sensitivity correction)
9. Hardcopy output generation (Photowrite, tape, listings, etc)
10. Special processing

The way in which these various functions are combined to provide the final results for any given object spectrum is illustrated in Figure 3-1. For each type of image which is used (e.g., wavelength calibration (WLC) lamp exposure, ultraviolet flood (UVF) lamp exposure, astronomical object, etc), the processing flow is indicated by the arrows connecting successive functions. The interrelationships between the flows for the various images are also shown by connective arrows.

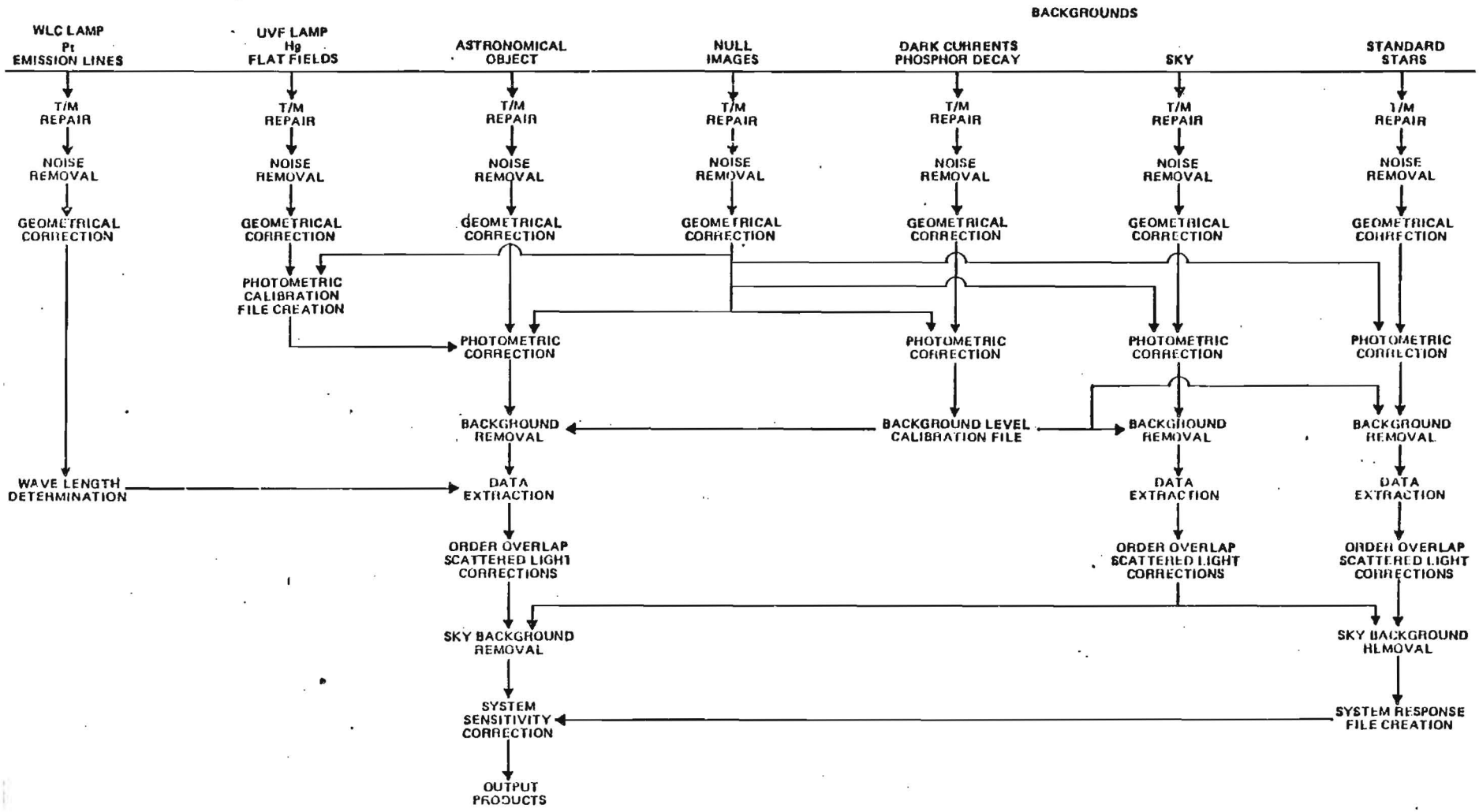

Figure 3-1. Image Processing Functional Overview

(courtesy of D. A. Klinglesmith GSFC)

3.2 IMAGE PROCESSING SCHEMES

The various IUESIPS image processing functions discussed above are realized by combining applications programs into processing schemes. A scheme is an ordered collection of applications programs and is the basic operational unit of IUESIPS. Schemes are inherently flexible entities which are edited and modified as necessary to perform image processing functions in a variety of contexts. The applications programs of which schemes are built, and which are discussed in detail in Section 4 of this document, are static, normally unchanging modules which perform specific operations in much the same way as FORTRAN subroutines do. The power and flexibility of IUESIPS stem in large part from its facility for tailoring schemes to fit the needs of a wide variety of situations.

Since the details of schemes evolve with time as the overall needs of the image processing operation are modified and/or more closely delineated, individual schemes are not listed here. A master library of all current schemes is maintained in the IPCC for reference by guest investigators, and listings of all schemes actually used in the processing of a guest investigator's data will be routinely included among the output products. In the following discussion, a description of the operations performed by the major schemes, such as they currently exist, within each of the functional areas described in Section 3.1 is given.

1. Transmission error correction. The schemes within this functional area are used to repair dislocations of portions of images caused by telemetry errors. Scheme development is required in this area.

2. Noise removal. Periodic noise is most likely to be impressed upon the image during transmission from the spacecraft to the ground. In order to remove this type of noise the down-link data stream must first be correctly reconstructed in time. This means that allowance must be made for the

fact that an image is not transmitted continuously, line by line, but rather 48-image pixels are alternated with 16 bits of spacecraft engineering data. After the data stream is temporally reconstructed, it is transformed from the time domain to the frequency domain by means of the Fourier transform and a power spectrum is obtained. The noise frequencies are next identified by inspection of the power spectrum and are removed from the transformed data stream. Finally, an inverse transform is taken and the gaps containing the spacecraft data are closed. The resultant image is thus free of the identified periodic noise. Corrections for random noise can be made by checking the input array for the existence of spikes, missing lines, blemishes, and background non-uniformity. If desired, random spikes and missing lines may be replaced by weighted averages of neighboring pixels; permanent and quasi-permanent errors may be identified and masked with the aid of blemish maps.

3. Geometric distortion correction. Two principal schemes are employed in this functional area. One scheme is used to perform a complete geometric correction on flat field images for use in intensity transfer function calibration files. The precise locations of fiducial reseau marks are located on each flat field exposure by the applications program FNDRES. These positions are used by the applications program REMRES to smoothly interpolate across the image area surrounding each identified reseau mark, effectively "removing" each mark. Finally, the positional information from FNDRES is used by the applications program GEOM to correct each flat field image for geometric distortion.

   The second scheme is an abbreviated version of the first and is used on actual spectral images. In this case, only GEOM is used. Reseaux are not removed; their locations are inferred from relevant flat field images taken with the same camera.

4. Photometric calibration. Several schemes are required to perform this function. A scheme executing the program VPFCF1 is used to create a pixel-by-pixel SEC vidicon intensity transfer function calibration file from graded exposure, geometrically corrected flat field images. Once this calibration file has been created, another scheme uses the program FICOR1 or MICOR1 to apply this calibration to a given data image.

5. Background removal. The schemes for this functional area are used to subtract unwanted background contributions from IUE images. Scheme development is required in this area.

6. Wavelength determination. The scheme for determining wavelengths on IUE echelle spectra uses the applications program WAVECAL. Initially, WAVECAL is applied to a platinum emission-line calibration spectrum to determine approximate emission-line positions. Then, WAVECAL is re-applied with this input to the same spectrum to render precise wavelength dispersion constants.

7. Spectral intensity determination. This image processing function, sometimes referred to as "data extraction," is performed in the high dispersion case by a scheme which uses the applications program DATEXT. DATEXT calculates the integrated intensity in an artificial slit which is passed along each echelle order, thereby creating a table of I (λ). Additional programs IUESORT and IUEPLOT are also executed to meaningfully arrange the entries of the I (λ) table and to produce a plot of I versus λ. Programs to treat the low dispersion case are under development.

8. Standard star calibration. This function serves to put the instrumental spectral intensities calculated within functional area seven on an absolute flux scale by correcting for the overall system sensitivity. The system sensitivity is determined from the observed instrumental spectral intensities of standard stars for which absolutely calibrated spectra exist. Scheme development is required in this area.

9. Hardcopy output generation. Several output operations are performed by this processing function. Photographic images may be obtained from magnetic tape files using the Photowrite device; the IUESIPS program MASK may be used to add gray-level scale information to output tape images. Ordinary tape and listing outputs are also available.

10. Special processing. This functional area encompasses special processing requirements arising infrequently or on a contingency basis. An example of the type of operation included herein would be correction for echelle order overlap. Scheme development as required will be performed in this area.

SECTION 4 - IMAGE PROCESSING APPLICATIONS PROGRAMS

4.1 OVERVIEW OF PROGRAMS

Applications programs are executable routines which perform specific, standardized image processing operations. The most important applications programs from an astronomical viewpoint--those which directly affect or extract the scientific content of the images--are discussed below in some detail to familiarize the guest investigator with the operations which are performed on the image data. Brief descriptions of these and less critical applications programs are catalogued in Appendix A.

4.2 PROGRAM DESCRIPTIONS

4.2.1 Periodic Noise Removal Routines

The IUESIPS routines which perform the periodic noise removal function are SPLIT, FFT1, POWER, AX, SCALP, and JOIN.

SPLIT expands the IUE image line of 768 samples to a line of 1024 samples. The first 8 samples of the expanded line contain zeroes, followed by the first 48 samples of the IUE image, and then 16 samples of zero values, 48 more samples from the IUE image, and so forth until the end of the line is reached. Instead of padding the image with zeroes linear interpolation may be used to fill the gaps.

FFT1 computes a one-dimensional Fourier transform using the Brenner MIT[2] subroutine FOUR2. For the forward transform the relation

$$G(f) = \frac{1}{N} \sum_{t=0}^{N-1} g(t) \exp(-2 \pi i f t/N) \qquad \text{(Equation 1)}$$

is used, and

$$g(t) = \sum_{f=0}^{N-1} G(f) \exp(+2\pi i f t/N) \qquad \text{(Equation 2)}$$

yields the inverse transform.

The fast Fourier transform (FFT) computational algorithm due to G. W. Cooley and J. W. Tukey[3] is used.

To insure a smooth transition to zero at the data string ends and to minimize "ringing" effects, the data can optionally be apodized before transformation. The apodizing function at any point is defined by

$$W(X) = \begin{cases} 0 \text{ for } X > R \\ 1 \text{ for } X \leq (1-f)R \\ 1 - \dfrac{1}{\text{Cos } \dfrac{\pi}{2(1-f)}} \text{ Cos } \left[\dfrac{\pi X}{2(1-f)R}\right] \text{ for } (1-f)\ R < X < R \end{cases} \qquad \text{(Equation 3)}$$

where f is the fraction of the data string over which the apodizing is done, and R is the interval between the center of the data string and the point where the apodizing function goes to zero. These parameters are shown in Figure 4-1.

The program POWER uses the fast Fourier transform (FFT) to compute the power spectrum of a specified portion of each line of an image and produces a single resultant power spectrum by averaging the separate spectra together as a function of frequency. The square root of this average power spectrum is then displayed on the line printer and a CalComp tape is generated. The actual tranform is computed using the subroutine FOUR2 (see discussion of

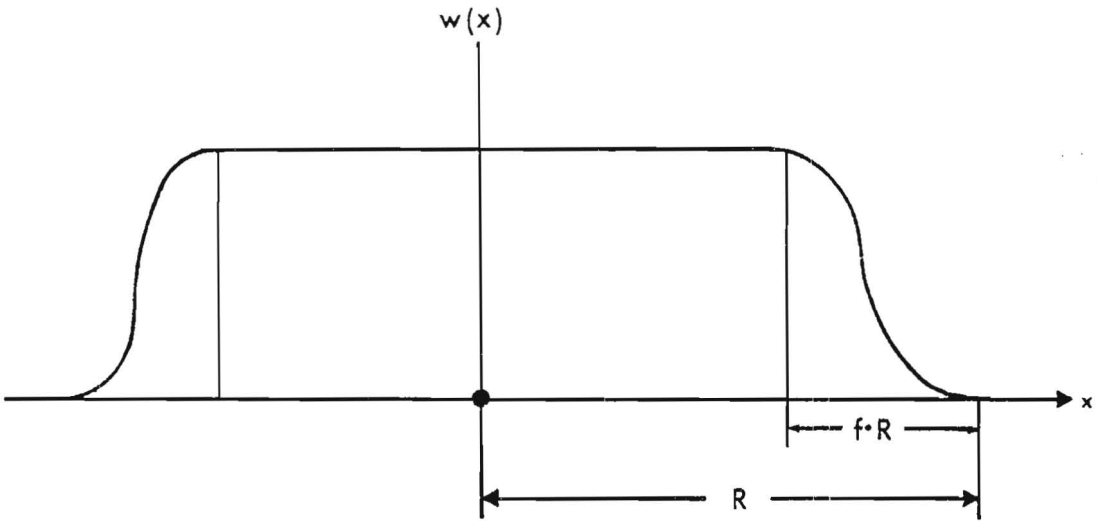

Figure 4-1. FFT1 Apodizing Parameters

program FFT1). The power spectrum is computed by taking the modulus of the real and imaginary Fourier co-efficients. Thus, we have:

$$A(f) = \sqrt{a^2(f) + b^2(f)}$$

with A (f) being the power at frequency f, a (f) the real (cosine) Fourier co-efficient and b (f) the imaginary (sine) co-efficient. A (f) is calculated for all frequencies up to the Nyquist frequency--$f_n = 1/2\,\Delta t$ with Δt being the sampling interval.

AX

AX is an IUESIPS applications program which removes periodicities from a data string at specified frequencies. It acts on the complex images that are produced by FFT1. The real and imaginary parts of the complex Fourier transformed picture at the specified positive and the corresponding negative frequencies are set to zero.

SCALP

SCALP is an IUESIPS applications program which reduces periodicities within a specified range of frequencies to the background level. It acts on the complex images that are produced by an execution of SPLIT followed by FFT1. End point frequencies of the noise spikes to be removed are supplied to the program via parameters. These end points are used to determine the background level adjacent to the noise spike. The Fourier transform between the specified positive frequencies and corresponding negative frequencies is set to the background level. The phase angle remains the same.

JOIN

JOIN performs the reverve function of SPLIT in that it compresses the 1024 image line back to 768 samples. The program would generally be the last to be run in the periodic noise removal scheme.

4.2.2 General Purpose Low Pass Filtering/Averaging Programs

Averaging-type programs find application in image processing in the areas such as: 1) image size reduction, 2) noise reduction, 3) low-pass filtering, and 4) smoothing. Additionally, if the results of computations used in the determination of the average are processed further, statistical information about the image can be obtained. IUESIPS provides three routines in this functional category: BOXAV, BOXFILT, and BOXSTATS. As the names imply these routines convolve rectangular (box) apertures with the image. These routines will be discussed separately below.

BOXAV performs simple non-overlapping averaging of an input image in two dimensions. The sum of points within a rectangle is scaled by the number of points in that rectangle. The initial rectangle (l lines by s samples in size) is placed in the upper left-hand corner of the image, the average is found, and the rectangle is moved to the right s samples. This process continues to the end of the line. If at the end of the line a fraction of a rectangle remains, it is ignored. The box is then shifted down l lines and the process is repeated. As in the case of individual lines, fractions of a box at the bottom edge of an image are ignored. The operation of this filter is shown diagrammatically in Figure 4-2. Because BOXAV performs a non-overlapping average it is useful for image size reduction applications. It should be kept in mind however, that as in the case of all simple averaging programs BOXAV acts as a low pass filter attenuating the high frequency components of the image data.

BOXAV also has the capability of calculating the rms deviation of the mean at each averaged point and placing this value in an array structured exactly the same way as the average value image array. This quantity is given by:

$$S_m = \frac{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2}}{n} \qquad \text{(Equation 5)}$$

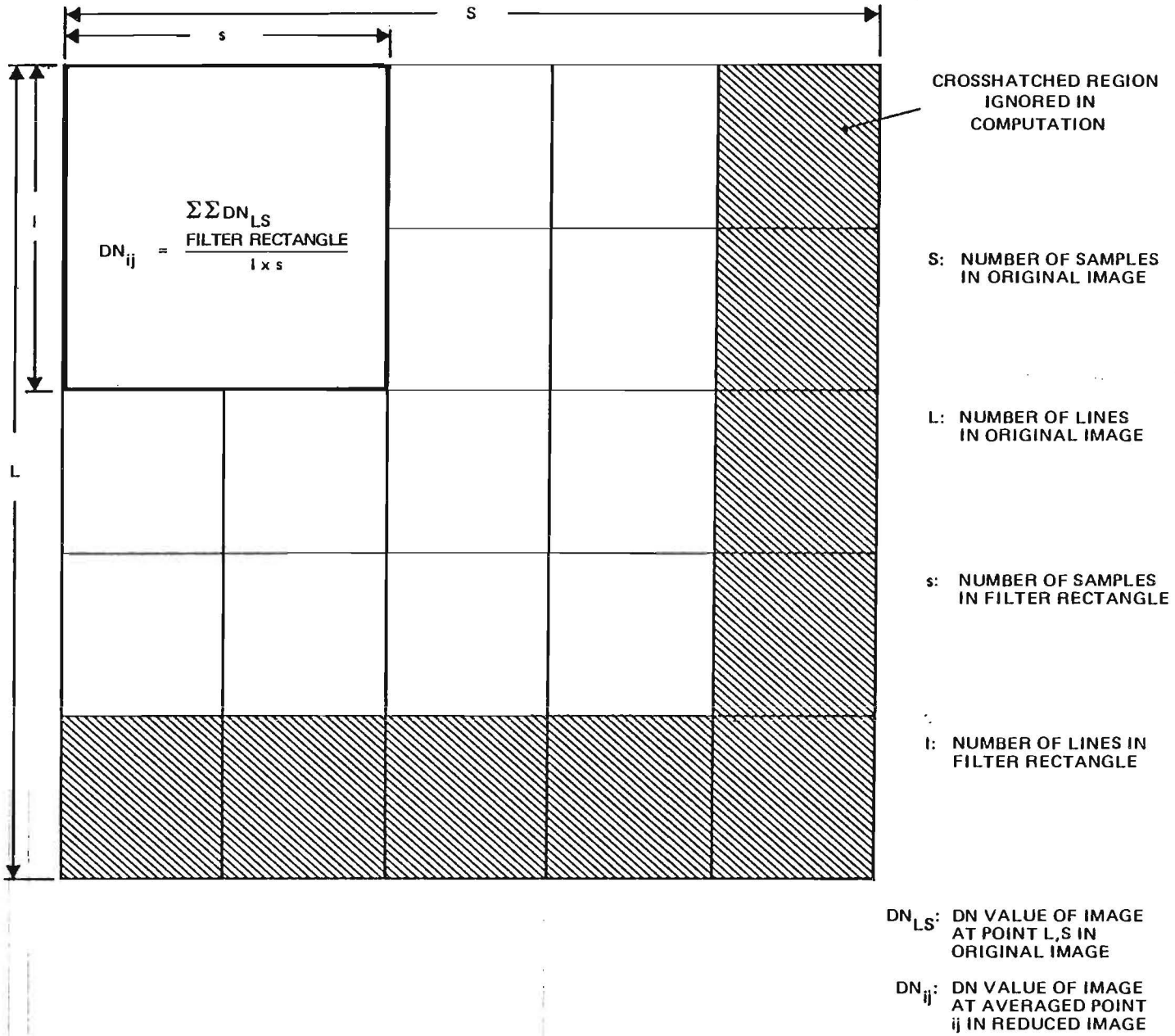

Figure 4-2. A Two-by-Two Pixel BOXAV Filter

The S_m values may be scaled as desired but since the array is an 'image' array all values must be between 0 and 255. All values outside this range will be truncated to fit.

BOXFILT performs a running average of an input image in two dimensions. A filter rectangle is shifted one sample at a time and reflection of the appropriate image data is done at the edges if necessary so the output image will be the same size as the input. The filter rectangle must have an odd number of lines and samples; if an even-sized filter is specified, BOXFILT automatically increases the size by one pixel. The maximum filter rectangle size is 31 x 31 pixels.

BOXSTATS generates rms deviations from respective mean values of a set of contiguous rectangular areas (boxes) of an input image. The mean values and ratios of mean values to rms deviation can also be produced. Fractions of boxes at the high line and high sample edges of an image are not processed. The sum of pixel values in a box is scaled by the number of pixels in the box, thus, this routine acts as a low pass filter when mean values are produced. The default size of the averaging box is 25 x 25 pixels. The default scaling factor by which the output rms deviations are multiplied is 10, while output mean values and the ratio of means to unscaled standard deviations are scaled by an integer with a default of 1. The rms deviations are computed by using Equation 5.

4.2.3 FNDRES

FNDRES is an applications program designed to search an image for fiducial reseau spots. A cross-correlation technique is used to determine the precise location of up to 169 reseaux by searching areas centered on approximately known reseau positions which serve as initial input. FNDRES is ordinarily applied to graded exposure, flat field IUE images which are used in the generation of intensity transfer function calibration files. On such images, the exact locations of the reseaux must be known in order to perform effective "reseaux removal" (see program REMRES) and to perform geometrical image rectification (see program GEOM).

An algorithm very similar to that used to locate emission-line positions in the applications program WAVECAL is used here to define precise reseau locations. An initial requirement is that approximate positions for all relevant reseaux be input to the program. Such approximate positions are measurable on the EDS and are generally determined but once for each of the four IUE cameras. These approximate positions serve as the centers of search areas which may be up to 40 pixels large in both directions.

To perform the cross correlation, an intensity dependent, two-dimensional, "model" reseau is defined for each search area on the basis of the local mean DN value and a two-dimensional, normalized intrinsic reseau shape, or template, T. There may be up to five different templates defined by IUESIPS parameters; when more than one is defined, they are employed sequentially, as needed, in the cross-correlation procedure described below. That is, template number 2 is used only if template number 1 could not successfully locate the current reseau, and so on for additional templates. The maximum allowable template size in the line or sample direction is 10 pixels. The normalization of the templates used in the program is such that a template element with a value of 100 refers to an element of the reseau model with a value equal to the local DN value.

That is,

$$R_k (i,j) = .01 * M_k * T (i,j) \qquad \text{(Equation 6)}$$

where $R_k (i,j)$ is the reseau model for search area k, M_k is the simple average of the DN values within search area k, T (i,j) is a reseau template, and i and j are relative line and sample numbers.

As mentioned, in the search for any given reseau, template number 1 is used first. The model reseau in Equation 6 is cross-correlated with the actual image over the search area and a correlation matrix is computed. Each

element of the correlation matrix is equal to the sum of the squares of the differences between the model reseau and the actual input image, the sum being taken over all pixels encompassed by the model reseau. Other elements of the correlation matrix are generated as the reseau model is effectively moved about the entire search area.

Once all elements of the correlation matrix have been computed for the current reseau search area, the exact reseau position (correlation matrix minimum) is established as follows. Let the integral pixel location for which the correlation matrix is a minimum be denoted by relative line and sample numbers m and n, and let that correlation matrix entry be c (m, n). Then the location of the matrix minimum is refined by means of the interpolations given below, provided that $1 < m < m_{max}$ and $1 < n < n_{max}$, where m_{max} and n_{max} are the line and sample direction sizes of the correlation matrix:

$$\Delta m = \frac{c(m-1,n) - c(m,n)}{c(m-1,n) + c(m+1,n) - 2 * c(m,n)} - 0.5$$

$$\Delta n = \frac{c(m,n-1) - c(m,n)}{c(m,n-1) + c(m,n+1) - 2 * c(m,n)} - 0.5 \qquad \text{(Equation 7)}$$

If the conditions $1 < m < m_{max}$ and $1 < n < n_{max}$ are indeed satisfied, then the exact reseau position is inferred to be at relative line number $m + \Delta m$, relative sample number $n + \Delta n$; otherwise, the location (m,n) is used.

Two tests are made of the significance of this result: 1) the difference between the maximum and the minimum correlation matrix entries must exceed a preselected value (input via an IUESIPS parameter), and 2) the displacement in the line or sample direction of the inferred exact reseau position from the originally input approximate position must be less than a preselected value (also input via an IUESIPS parameter). If both conditions are satisfied, the search is regarded as successful, the new exact position is recorded, and the program cycles to the next reseau spot to begin the search process again with template 1 in the new search area.

If either condition 1 or 2 is not satisfied, the current reseau has not successfully been found with template number 1, and if other templates exist, the entire procedure is repeated within the current search area using template number 2. This process can be repeated until all available templates (up to 5) have been used. If none of the templates are successful, a position of (0, 0) is entered and the program goes on to consider the next reseau spot.

4.2.4 REMRES

REMRES is an applications program which is used to effectively "remove" fiducial reseau spots from an image. A simple averaging procedure is used to replace the observed DN values in small areas centered on the specified positions of up to 169 reseaux with interpolated DN values representative of the local background. REMRES is used mainly in the preparation of graded exposure, flat field images which are required for the generation of intensity transfer function calibration files. In this context, the use of REMRES enables the definition of an interpolated intensity transfer function for pixels lying under fiducial reseaux.

a. Pixel Layout and Definitions

The coordinates of up to 169 reseau spots, usually determined by the program FNDRES, are input to the program. Treating each reseau in sequence, the general procedure is to work with an 8-pixel by 8-pixel area of the image situated about the input reseau position such that the nearest integer rounded reseau center lies on line 5 and sample 5 of the working 8 by 8 array. Figure 4-3 illustrates the layout of a 64-pixel working array in relation to the location of the integer reseau center. The central 4-pixel by 4-pixel array with cross-hatching always has its DN values replaced; DN values for the pixels within the region with single shading immediately adjacent to the central area may be replaced on an individual basis if certain conditions associating them with the reseau itself are met; and the one-pixel wide outer border region serves strictly as a background DN value tie-down point--its pixels are never replaced.

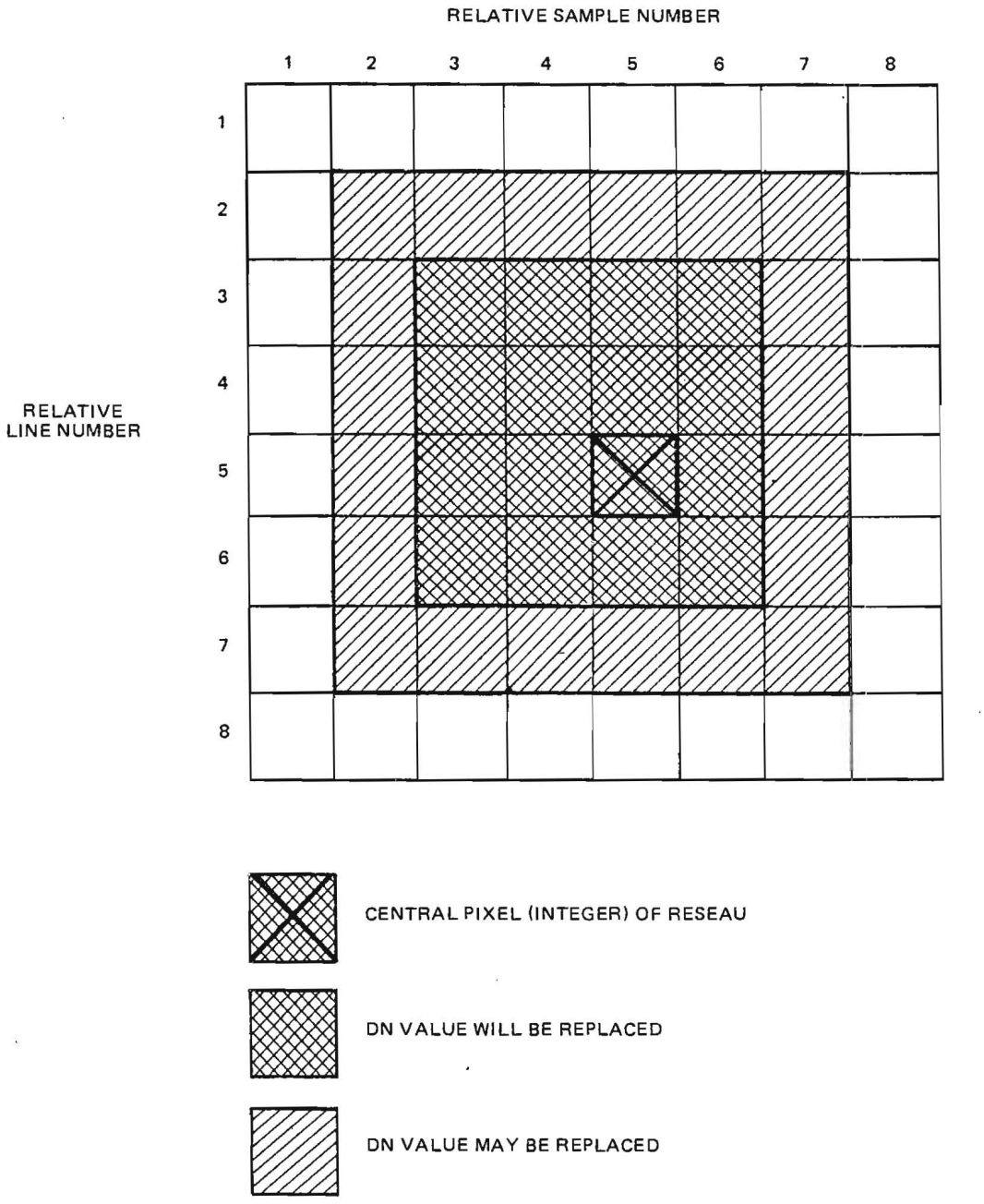

Figure 4-3. Pixel Array for Reseau Removal

The replacement operation for the inner pixels begins with the following considerations. Whenever possible, a pixel within the singly-shaded region is used as a background point. This is not done if and only if that pixel appears to belong to the reseau mark, a condition which is determined by the characteristics of the DN gradient in a single direction. This is best illustrated by specific example: consider the pixel at line 4 sample 2, with DN value DN (4,2). Similarly consider the adjacent pixels on line 4 with DN (4,1) and DN (4,3). If

$$\frac{|DN\ (4,2)\ -\ DN\ (4,3)|}{2} < |\ DN\ (4,2)\ -\ DN\ (4,1)|\qquad \text{(Equation 8)}$$

then pixel (4,2) is defined to be associated with the reseau mark and will have its DN value replaced; it will not be used as a background point--rather, the adjacent outer border pixel (4,1) will be used. An analogous test is made on pixel (4,7), involving pixels (4,6) and (4,8) to determine whether (4,7) will be used as a tie-down point or replaced. Then, a "line-average"--the simple arithmetic mean DN value of the two selected tie-down pixels at either end of line 4--is computed for later use in the interpolation and replacement process. Similar calculations are made for the other three central image lines, (lines 3, 5, and 6): a determination of the tie-down points at either end of these lines is made, and "line averages" are computed. Entirely analogously, tie-down points at constant sample number are determined for the central four samples (3, 4, 5, and 6) and corresponding "column averages" are computed.

Note that this process has determined the fate (to be used as a tie-down point or replaced) of all the pixels within the singly-shaded area except the corner pixels (2,2), (7,2), (2,7), and (7,7). Whereas these corner pixels are never used as tie-down points, they may or may not be replaced, according to

whether the local DN gradient in the diagonal direction indicates they are to be associated with the reseau mark or not. For example, if

$$\frac{|DN\ (2,2)\ -\ DN\ (3,3)|}{2} < |DN\ (2,2)\ -\ DN\ (1,1)| \qquad \text{(Equation 9)}$$

then pixel (2,2) is to be replaced by an interpolated DN value.

b. Replacement (Interpolation) Formulae

The calculation of interpolated DN values for those pixels to be replaced proceeds in the following order: central 4 x 4 pixel array; line and column elements bordering the central array, if required; and finally, any corner points, if required.

For each of the 16 central pixels, the interpolated replacement DN value is the straight mean of the appropriate line and column averages as defined in the preceding section:

$$DN_{new}(i,j) = \frac{\text{line average (i)} + \text{column average (j)}}{2} \qquad \text{(Equation 10)}$$

where i, j are the relative line and sample (column) numbers.

For the non-corner singly-shaded border pixels, replacement DN values, if replacements are indeed to be made, are simply the appropriate line average or sample average. (Note that the question of whether any given one of these border points is to be replaced is determined individually for each such pixel, as described in the preceding section.)

Finally, if any of the corner border pixels are to be replaced, their new DN values are interpolated along the appropriate diagonal line, using the straight mean of the appropriate outermost corner pixel [(1,1), (8,1), (1,8) or (8,8)] and the appropriate corner pixel of the newly replaced central array [(3,3), (6,3), (3,6), or (6,6)].

Once all DN value replacements have been made for the current working region, the next reseau location is processed similarly until all reseaux (≤169) are thus "removed" by interpolation. The final, "cleaned" image is the output product of the program.

4.2.5 GEOM

GEOM is an applications program designed to alter the geometric structure of an image. It is normally used to correct IUE images for geometric distortion introduced by the SEC vidicon, although it may also be used to expand or reduce images with no alteration of shape.

IUE image rectification is performed by means of reference to the square grid of fiducial marks (reseaux) deposited on the faceplate of each camera. Departures of the observed reseaux from their true locations are measured on flat field exposures and are used to characterize the geometric distortion of each SEC tube. Input to GEOM consists of the true (undistorted) locations of the fiducial marks and the observed displacement of each mark (in lines and samples) from its true position. The line and sample displacements (which need not be integers) for any four adjacent fiducial marks define a "displacement rectangle" which describes the local geometric properties of that portion of the image. The 169 fiducial marks on each camera faceplate define 144 such displacement rectangles which GEOM uses to map the observed image into an undistorted format. Using the displacement rectangles as tie-down points, the line and sample displacements for any pixel in the image are determined by a two-dimensional bi-linear interpolation within the appropriate rectangle. For each pixel of the output image, the inferred line and sample displacements are used to find the corresponding pixel (or fractional parts of several pixels) in the input image and thus to define the proper output DN value.

Consider the true square defined by four adjacent fiducial marks located in line numbers M and N, and sample numbers A and B (see Figure 4-4). If

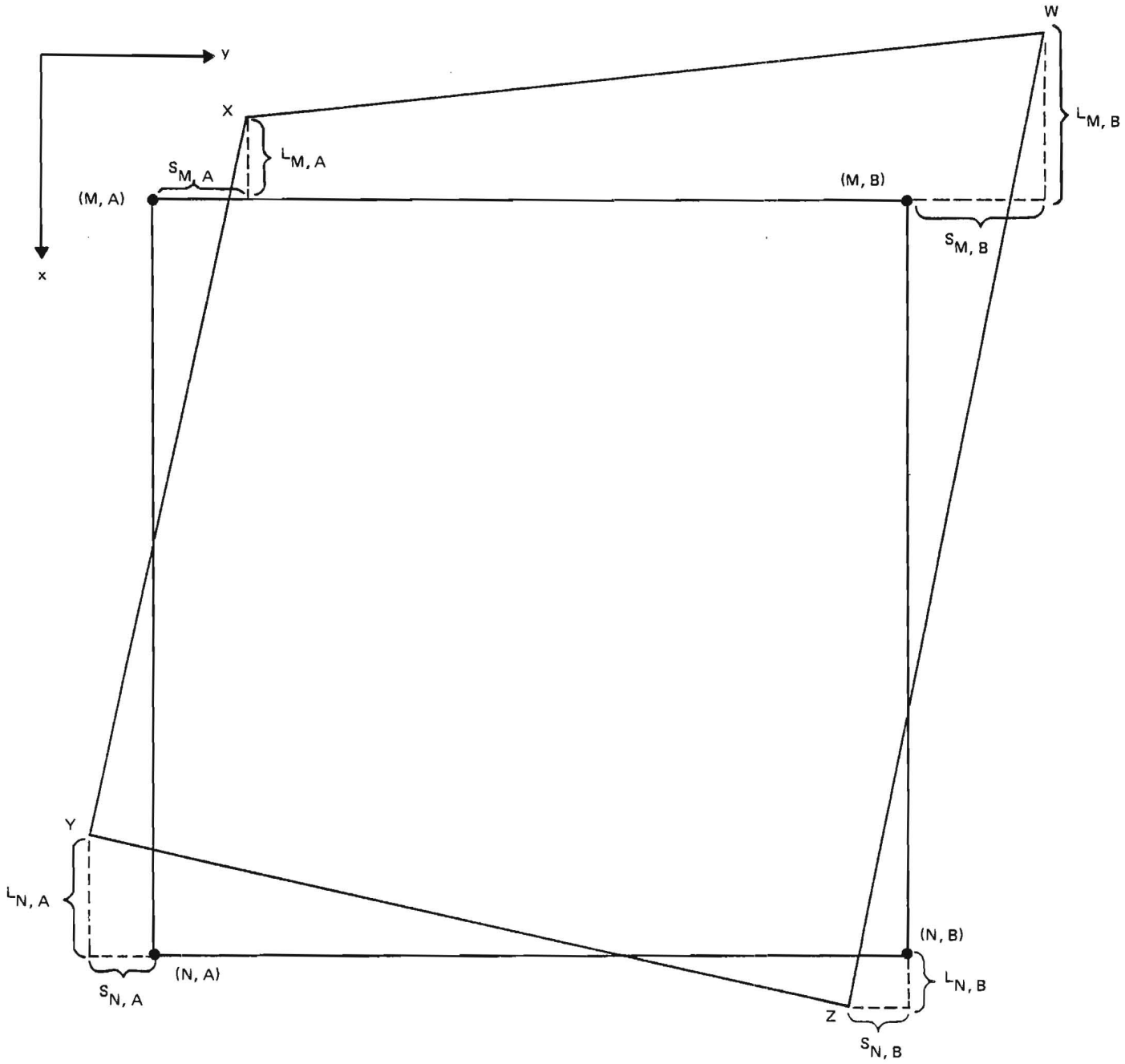

Figure 4-4. Geometric Distortion Displacements

these fiducial marks are observed to reside at the points W, X, Y, and Z of the input (distorted) image, then the projected displacements of each observed point along the line and sample directions are denoted $L_{M,A}$, $S_{M,A}$, $L_{M,B}$, $S_{M,B}$, etc, (see Figure 4-4). These displacements are defined algebraically in the sense "observed-minus-true" in the coordinate system shown in Figure 4-4.

The following interpolation constants are defined in order to establish the pixel correspondence between the true and the distorted images:

$NR = N - M$ = number of lines of the "displacement rectangle".

$NC = B - A$ = number of samples of the "displacement rectangle".

$$F_{S,M} \equiv 1 + \frac{S_{M,B} - S_{M,A}}{NC}$$ = factor by which length of rectangle in sample direction, at line M, is changed (this, and each of the following constants, refers to properties of the distorted image relative to those of the true, or undistorted, image).

(Equation 11)

$$F_{S,N} \equiv 1 + \frac{S_{N,B} - S_{N,A}}{NC}$$ = factor by which length of rectangle in sample direction, at line N, is changed.

$$\Delta F_S \equiv \frac{F_{S,N} - F_{S,M}}{NR}$$ = gradient of the sample direction length factor between lines M and N (change in length factor per line).

$$\Delta S \equiv \frac{S_{N,A} - S_{M,A}}{NR}$$ = gradient in sample number, between lines M and N, for points along left edge of rectangle (change in starting sample per line).

$$F_{L,A} \equiv 1 + \frac{L_{N,A} - L_{M,A}}{NR} =$$ factor by which length of rectangle in line direction, at sample A, is changed.

$$F_{L,B} \equiv 1 + \frac{L_{N,B} - L_{M,B}}{NR} =$$ factor by which length of rectangle in line direction, at sample B, is changed.

$$\Delta F_L \equiv \frac{F_{L,B} - F_{L,A}}{NC} =$$ gradient of the line direction length factor between samples A and B (change in length factor per sample).

$$\Delta L \equiv \frac{L_{M,B} - L_{M,A}}{NC} =$$ gradient in line number, between samples B and A, for points along top edge of rectangle (change in line number per sample).

(Equation 11) (contd)

The above interpolation constants are calculated for all displacement rectangles. For every point (i,j) in the output (undistorted) image (i = line, j = sample), the program determines the appropriate displacement rectangle within which to work, and the point (r + u, p + q) in the input (distorted) image which corresponds to the output pixel is calculated by bi-linear interpolation as follows, where r and p are integers, and u and q are fractional parts less than unity:

$$r + u = M + L_{M,A} + (j - A)\ \Delta L + (i - M)\ (F_{L,A} + (j - A)\ \Delta F_L)$$

(Equation 12)

$$p + q = A + S_{M,A} + (i - M)\ \Delta S + (j - A)\ (F_{S,M} + (i - M)\ \Delta F_S)$$

(Equation 13)

Ordinarily, the point $(r + u, p + q)$ does not coincide precisely with one pixel of the input image. In the most general case, four adjacent pixels in the input image contribute fractionally to the DN value desired, which must then be calculated by four-point interpolation (see Figure 4-5):

$$DN_{out}(i,j) = (1-q)(1-u)\,DN_{in}(r,p) + q(1-u)\,DN_{in}(r,p+1) + (1-q)\,u\,DN_{in}(r+1,p) + q\,u\,DN_{in}(r+1,p+1)$$

(Equation 14)

There are two exceptions to the use of the above expression to obtain $DN_{out}(i,j)$:

1. if $u < .01$, it is replaced by $u = 0$, and if $u > .99$, it is replaced by $u = 1$, so that only two input pixels, on a common line, are used.
2. if any of the four (or two) input pixels specified for actual use in the evaluation of $DN_{out}(i,j)$ lie outside the boundaries of the input image, $DN_{out}(i,j)$ is set equal to zero.

4.2.6 VPFCF1

VPFCF1 is a program which is used to combine a set of up to 10 flat field exposures of increasing exposure time into an SEC vidicon intensity transfer function photometric calibration file containing the relation between output DN value and input light intensity for each pixel. The use of this transfer function allows the non-linearity and non-uniformity of the SEC vidicon response to be corrected on a pixel-by-pixel basis.

Input to the program consists of at least three, but no more than 10 flat field exposures arranged in order of increasing exposure time. In current usage, the first input image is a null exposure, i.e., an image of zero exposure time which is read out following an erase. This is included because the camera read-out process itself induces a non-zero background DN level. If future

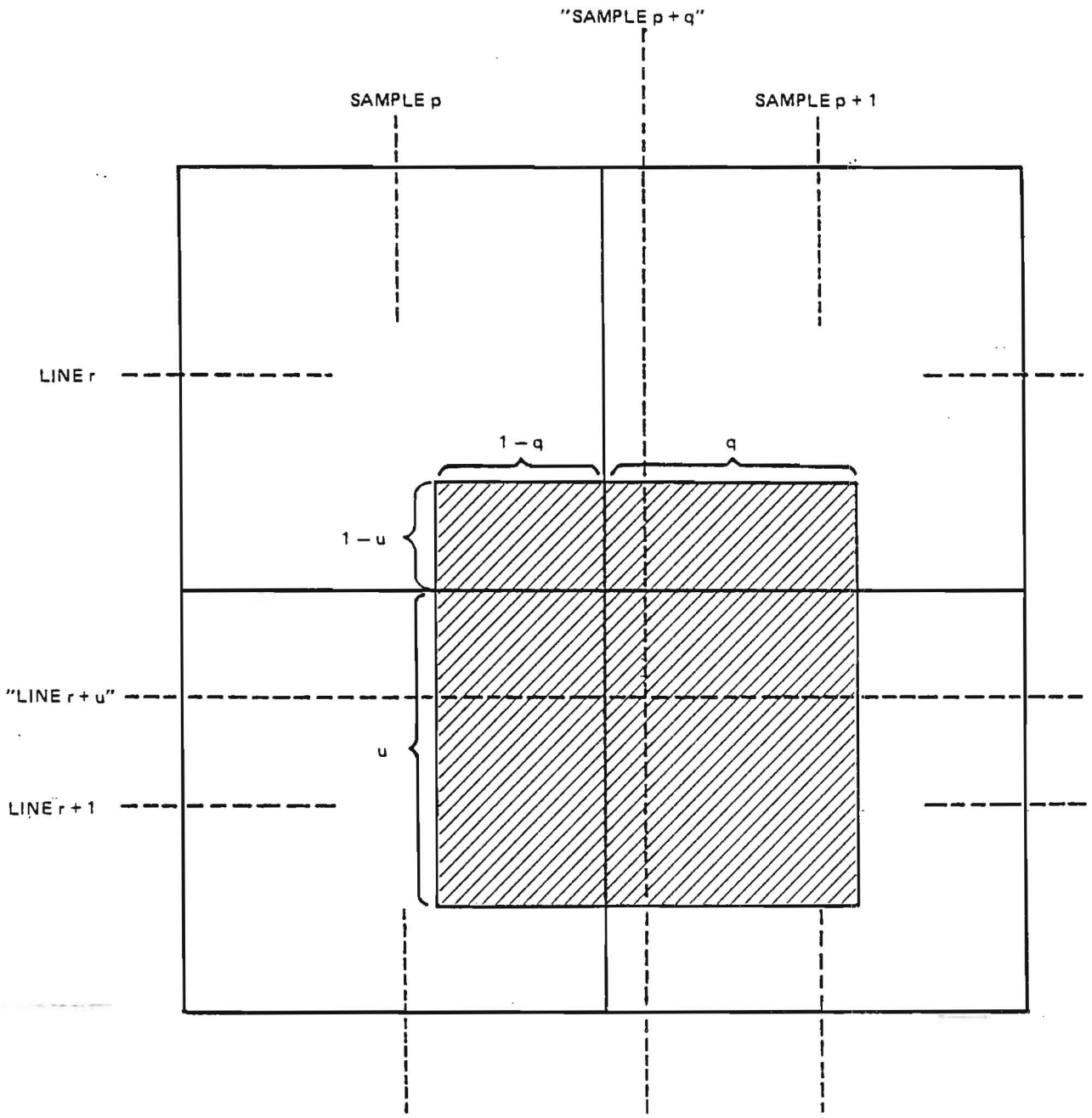

Figure 4-5. Pixel Resampling by Program GEOM

experience indicates that this read-out background is sufficiently variable, it may become necessary to monitor and subtract this background in an entirely separate step, which would mean that the first flat field exposure used in constructing the calibration file would have a non-zero exposure time.

If there are N input images, they are merged into a calibration file which has the form of a VICAR image with 768 lines and 768 * N samples arranged in the manner described below. Line 1 of the output calibration file image holds the DN values from line 1 of all of the input images, interleaved so that the first N samples of output line 1 are the DN values from line 1, sample 1 for each of the input images in order; the next N samples of output line 1 are the DN values from line 1, sample 2 for each of the input images in order, and so forth. In other words, each output image line is constructed by stringing together the N-point intensity transfer functions for each of the pixels in that image line.

In standard usage, the individual intensity transfer function for each pixel is forced to be a montonically increasing function of exposure time. If, as originally input, the DN value of a given pixel for the i th exposure is less than or equal to that for the preceding exposure, VPFCF1 will automatically increase DN (i) so that it is greater than DN (i - 1) (up to the maximum saturation DN value of 255): if DN (i) $\leq$ DN (i - 1), then redefine DN (i) $\equiv$ DN (i - 1) + 1 up to a maximum of 255, where $2 \leq i \leq N$. If so desired, the imposition of monotonicity may be suppressed by means of an IUESIPS parameter.

4.2.7 FICOR1

FICOR1 is an applications program which is primarily used to photometrically correct IUE images for the non-linearity and non-uniformity of the SEC vidicon intensity transfer function. Using the intensity transfer function calibration file generated by the program VPFCF1, FICOR1 determines an incident flux value corresponding to the observed DN value for each pixel by linear interpolation.

FICOR1 first determines the correct portion of the calibration file to be used for the particular pixel under consideration. Then, the flux corresponding to the observed DN value is linearly interpolated from the two calibration file flux values corresponding to the two DN values bracketing the observed DN. Two special situations are handled in the fcllowing ways:

1. If the observed DN value lies outside the range of the calibration file for that pixel, then the flux is set equal to the corresponding extreme (i.e., low or high) calibration file flux for that pixel.

2. If the observed DN value is equal to the saturation DN value for that pixel, the output flux is set equal to the lowest saturation flux value for that pixel, except when the saturation DN value is 255, in which case the output flux is set equal to the highest saturation flux value for that pixel.

The entire set of calibration file flux values may be scaled, if desired, through the use of an IUESIPS parameter FACTOR. The output flux image may be in either half-word or byte format. The flux values in the latter case are always scaled by the two parameters A0 and A1 as follows:

$$FLUX = (LUM - A0)/A1 + 0.5 \qquad \text{(Equation 15)}$$

where LUM is the original flux, and FLUX is the nearest integer rounded output value.

In the case of both half-word and byte output format, the original pixel fluxes LUM are checked to see that they are within an acceptable range of values delimited by the quantities MINLUM and MAXLUM. For half-word output MINLUM = 0.0 and MAXLUM = 32767.0; for byte output MINLUM = A0 and MAXLUM = 255.0 * A1 + A0. Values of LUM outside the acceptable range are assigned the relevant extreme value. Note that this assignment takes place before the scaling indicated in Equation 15 is performed for byte output.

4.2.8 MICOR1

MICOR1 is an applications program which is primarily used to photometrically correct IUE images for the non-linearity and non-uniformity of the SEC vidicon intensity transfer function. In this respect, it is similar to the program FICOR1: using the intensity transfer function calibration file generated by the program VPFCF1, MICOR1 determines an incident flux value corresponding to the observed DN value for each pixel by means of quadratic interpolation.

With the exception of the interpolation algorithm, MICOR1 operates entirely similarly to the program FICOR1, which performs a linear interpolation within the intensity transfer function calibration file. The various other attributes of FICOR1 have been discussed in the section of this document describing that program and are not repeated here; only the interpolation algorithm is discussed below.

For each input pixel, MICOR1 determines the correct portion of the calibration file to be used. Then the actual interpolation is done using three adjacent points on the intensity transfer function: if the input pixel DN value is DN, and if the DN values along the calibration curve for that pixel are denoted DN (i), $i = 1, N \leq 10$, then if,

$$DN(j) < DN \leq DN(j+1) \qquad \text{(Equation 16)}$$

in general the interpolation is done using points $j - 1$, j, and $j + 1$, except that if $DN \leq DN(3)$, points 1, 2, 3 are used even if $DN \leq DN(2)$ as well.

The interpolation scheme used is an approximation to a three-term Taylor expansion in the flux about the first of the three points selected above. In discussing this, the following definitions are convenient:

$$F(j) = \text{flux for point j of the calibration curve}$$

$$\Delta(j) = F(j) - F(j-1) = \text{flux differential between points j and j - 1} \qquad \text{(Equation 17)}$$

$$R(j) = \frac{\Delta(j)}{DN(j) - DN(j-1)} = \text{derivative of flux}$$

Then, assuming Equation 16 to be satisfied, the interpolated flux LUM corresponding to DN is,

$$LUM = F(j-1) + [DN - DN(j-1)]\ R(j) + [DN - DN(j-1)] \times [DN - DN(j)] \frac{[R(j+1) - R(j)]}{[DN(j+1) - DN(j-1)]} \qquad \text{(Equation 18)}$$

For the special case DN ≤ DN (3), Equation 18 is used with j = 2.

4.2.9 WAVECAL

WAVECAL is an IUESIPS applications program which is used to determine the dispersion constants relating the positions of spectral lines across the IUE image (line and sample numbers) to their wavelengths and echelle orders. These data effectively provide the wavelength calibration for the IUE spectrographs and are extracted from images of a platinum lamp (Pt) emission line spectrum which have been corrected for geometric distortion. Wavelength, echelle order number, relative strength, and positional data for up to 296 platinum lamp lines are used to determine the dispersion constants. WAVECAL first determines the exact positions of the Pt lines by means of a cross-correlation technique which searches the areas surrounding the approximate locations of the lines. It then uses these exact positions to compute the

dispersion constants by means of regression analysis. Statistics describing the errors of individual line measurements are generated, lines with anomalously large positional errors are identified, and the regression analysis is repeated excluding the anomalous lines. Up to five regression analyses are allowed.

a. Line Location

Location of the exact positions of the Pt lines is generally effected by a multi-step process. In order for the cross-correlation search to work, the approximate locations of the Pt lines must be known. Ordinarily, these approximate positions will have been calculated from preliminary dispersion constants obtained from a separate execution of WAVECAL in which accurately-measured positions of a small number of Pt lines are used in a preliminary regression analysis involving only two independent variables--see section c for further information. (Typically, five Pt spectral lines might be used for this, their line and sample locations having been measured on the EDS.) Once the approximate positions of the lines are specified, their exact positions are determined as follows. A two-dimensional spectral line shape instrumental profile, or template (TMPL), is defined as an IUESIPS parameter pertaining to all the Pt lines. An exposure value, EXP, characteristic of the particular image in use is also specified as an IUESIPS parameter. Finally, the expected DN strength of each line, INT (k), k = 1296 is specified. From these inputs, a two-dimensional "model spectral line" for the k th line is defined as:

$$\text{MODEL}_k\,(i,j) = \text{EXP} \times \text{INT}\,(k) \times \text{TMPL}\,(i,j) \qquad \text{(Equation 19)}$$

where i = relative line number, j = relative sample number. Typically, the template size is of the order 3-by 3-pixels.

The model flux distribution given by Equation 19 is cross-correlated with the actual image in the vicinity of each Pt line, and a correlation matrix whose entries hold the sum of the squares of the differences between the model flux elements and the actual image elements is defined for limited excursions about the approximate Pt line location. These excursions are typically of the order of ±5 pixels in both the line and sample directions. The inferred exact position of a given Pt line is that pixel location for which the correlation matrix is a minimum. The significance of this result is assessed in two ways: 1) the difference between the maximum matrix entry and the minimum matrix entry must exceed a preselected value (input via an IUESIPS parameter), and 2) the displacement in the line or sample direction of the inferred exact position from the input approximate position must be less than a preselected value (also input by an IUESIPS parameter). If either condition 1 or 2 is not satisfied, that particular line is considered not to have been found, values of zero are entered in the storage array for the line and sample numbers for that emission line, and the line is excluded from the regression analyses.

b. Revised Relative Line Strength Calculation

For each Pt line for which an exact position has been found, a revised DN strength I (k) is defined as that value which best reproduces the observed integrated line strength (in the least-squares sense) given the line shape template and image exposure value:

$$I(k) \equiv \frac{\sum_{ij} DN(i,j) \times TMPL(i,j) \times EXP}{\sum_{ij} (TMPL(i,j) \times EXP)^2} \qquad \text{(Equation 20)}$$

where the sums are made over the pixels included by TMPL (i,j), centered on the exact line position, and the DN (i,j) are the observed pixel DN values.

c. Regression Analysis and Dispersion Constants

Given the exact positions SAMP (k) and LINE (k) for the N Pt lines which were successfully found by the algorithm described in section a, a standard regression analysis is performed using NVAR (≤ 6) independent variables to determine the dispersion constants A_i and B_i:

$$\text{LINE (k)} = A_0 + A_1 Z_{1,k} + A_2 Z_{2,k} + \ldots\ldots + A_6 Z_{6,k} \qquad \text{(Equation 21)}$$

$$\text{SAMP (k)} = B_0 + B_1 Z_{1,k} + B_2 Z_{2,k} + \ldots\ldots + B_6 Z_{6,k} \qquad \text{(Equation 22)}$$

where the $Z_{i,k}$ are combinations of wavelength λ_k and echelle order m_k defined as follows:

$$\begin{aligned} Z_{1,k} &= m_k \lambda_k \\ Z_{2,k} &= (m_k \lambda_k)^2 \\ Z_{3,k} &= m_k \\ Z_{4,k} &= \lambda_k \\ Z_{5,k} &= m_k^2 \lambda_k \\ Z_{6,k} &= m_k \lambda_k^2 \end{aligned} \qquad \text{(Equation 23)}$$

[Note that it is possible to specify NVAR< 6, in which case not all of the variables given in Equation 23 would be utilized in the regression analysis. Specifically, in the case of the initial execution of WAVECAL to determine preliminary dispersion constants and thus approximate locations of all the Pt lines (see section a), NVAR is ordinarily equal to two, and the regression analysis uses

variables $Z_{1,k}$ and $Z_{4,k}$ (the terms in $m\lambda$ and λ). The subsequent executions of WAVECAL to determine the final dispersion constants ordinarily utilize all six variables $Z_{1,k} \ldots . Z_{6,k}$ for each Pt line.]

The dispersion constants thus found are used to compute the positions of the N lines which were used in the regression analysis. Denoting these computed positions by $SAMP_{comp}$ (k) and $LINE_{comp}$ (k), the mean error of a single measurement in the line direction, σ_l, and the mean error of a single measurement in the sample direction, σ_s, are calculated from,

$$\sigma_l \equiv \left(\frac{\sum_{k=1}^{N} (LINE\ (k) - LINE_{comp}\ (k))^2}{N - 1} \right)^{1/2} \qquad \text{(Equation 24)}$$

$$\sigma_s \equiv \left(\frac{\sum_{k=1}^{N} (SAMP\ (k) - SAMP_{comp}\ (k))^2}{N - 1} \right)^{1/2} \qquad \text{(Equation 25)}$$

These quantities serve as the yardsticks by which the individual residuals $SAMP_{comp\ (k)}$ - SAMP (k) and $LINE_{comp}$ (k) - LINE (k) can be measured. If any individual sample direction residual exceeds 2.5 σ_s or any individual line direction residual exceeds 2.5 σ_l, the regression analysis is recomputed with that emission line, and any additional Pt line with similarly large residuals, excluded. New dispersion constants and σ's are computed, and the above process is repeated as necessary, up to a maximum total of five regression analyses.

4.2.10 DATEXT

DATEXT is an IUESIPS applications program which performs several crucial data extraction operations on high dispersion IUE spectral images. Most importantly, DATEXT effectively acts to pass an artificial slit, oriented at approximately 90 degrees to the echelle dispersion, along each echelle order, thereby computing

a slit-integrated flux at each wavelength along each order. DATEXT will normally also be used prior to the slit-integration operation to correct the input intensities for the effect of echelle blaze variation (referred to as "echelle ripple") and may also be used to correct the intensities for the wavelength sensitivity variation of the UV converter.

DATEXT uses the dispersion constants output by the program WAVECAL to perform the precise wavelength calibrations, and outputs an unsorted table of echelle order number, wavelength, integrated slit flux, and qualitative error measure. Subsequent programs may be used to re-order these data into a more useful format.

a. General Processing Flow

DATEXT works on seven lines of the input image at a time. The order of processing is as follows. First, the dispersion formulae which relate line and sample location to wavelength and echelle order number are solved, using the dispersion constants found by the program WAVECAL. That is, for each of the seven input image lines currently being processed, the dispersion formulae are used to determine the sample numbers and wavelengths corresponding to the precise centers of the echelle orders as they cross the image line at an angle of approximately 39 degrees (see Figure 4-6). Second, pixels which do not lie along the exact order centers defined by the dispersion formulae (orders have a non-zero width perpendicular to the echelle dispersion) are associated with the nearest echelle order and an appropriate wavelength is assigned to them if possible. At this point, the corrections for UV converter wavelength sensitivity and echelle ripple may be applied. Finally, the artificial slit, which is oriented at an angle of 45 degrees to the image lines (~perpendicular to the echelle orders), is effectively placed over the intersection of each order with the current fourth image line and the flux from all pixels within the slit is added up and an error estimate is compiled. Once this slit-integrated flux is calculated

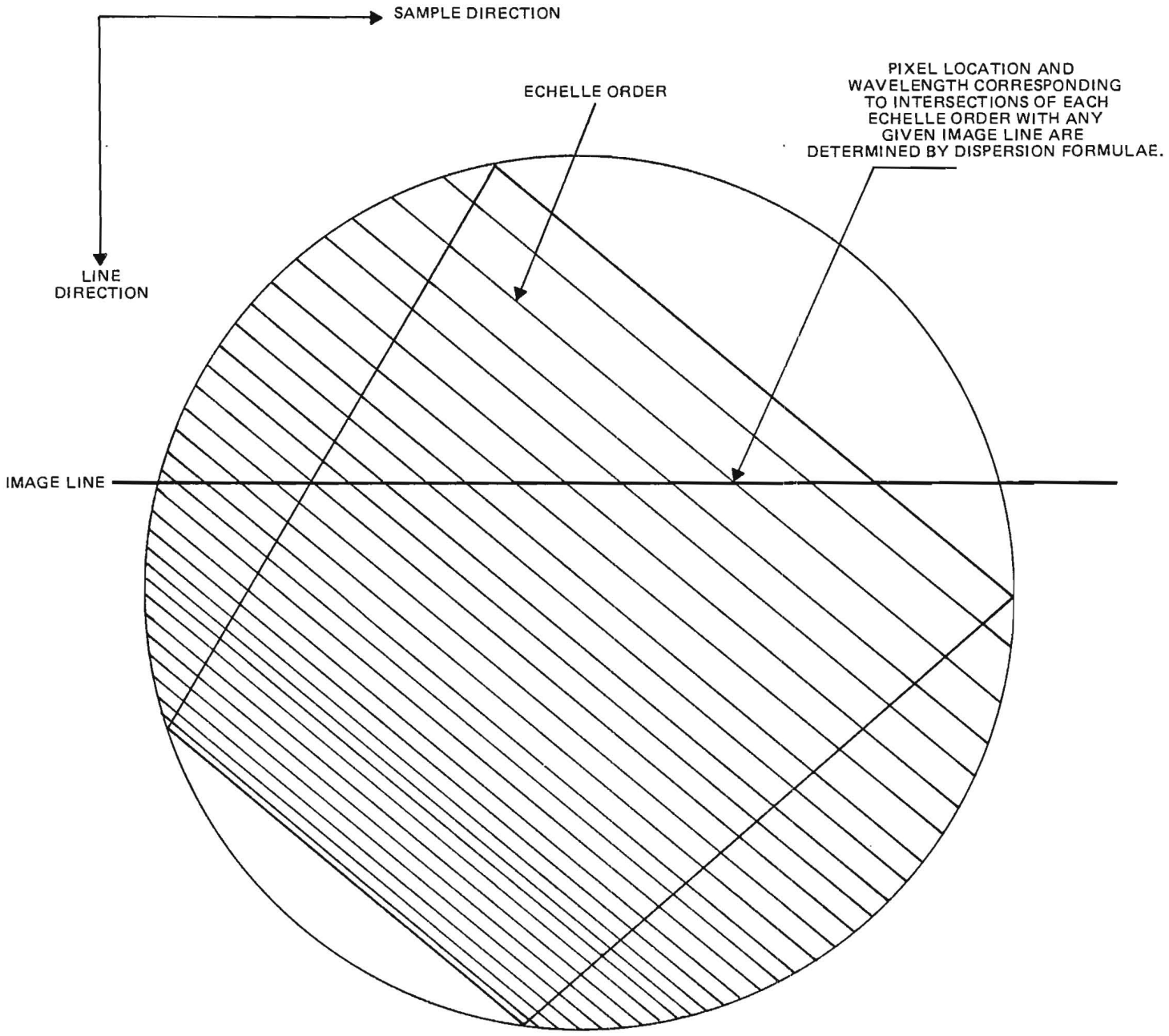

Figure 4-6. Schematic Echelle Image Format

for all orders which cross the current fourth image line, lines two through seven of the current array are rolled up into positions one through six and a new line seven is read in from disk. As above, the pixel locations and wavelengths corresponding to the intersections of the echelle order centers with the new seventh line are computed. Off-order pixels are assigned an order and a wavelength as above, and then the corrections for wavelength sensitivity and echelle ripple may also be applied. Finally the slit integration operation, with the slit now centered on the new fourth image line in the working array (which was line five in the preceding pass), is performed as above. Upon completion of this operation, the bottom six lines of the current seven are again rolled up, and the new seventh line is read in, and the process as described above is repeated until all input image lines have been read in. In this way, the artificial slit is passed, one image line at a time, along each echelle order. No allowance is made for overlap contributions to any given pixel from orders other than the primary assigned order.

b. Location of (Echelle Order, Image Line) Intersections

In the determination of the pixel locations and wavelengths corresponding to the intersections of the various echelle orders with any given image line, (as in Figure 4-6), the dispersion formulae in Equations 26-28 play a pivotal role:

$$\text{LINE} \equiv x = A_0 + A_1 Z_1 + A_2 Z_2 + \ldots\ldots + A_6 Z_6 \qquad \text{(Equation 26)}$$

$$\text{SAMP} \equiv y = B_0 + B_1 Z_1 + B_2 Z_2 + \ldots\ldots + B_6 Z_6 \qquad \text{(Equation 27)}$$

where,

$$Z_1 = m\lambda$$
$$Z_2 = (m\lambda)^2$$
$$Z_3 = m$$
$$Z_4 = \lambda$$
$$Z_5 = m^2\lambda$$
$$Z_6 = m\lambda^2$$

(Equation 28)

The dispersion constants A_i and B_i are determined by the program WAVECAL in a prior processing step. Given a line number x of the input image, the sample number and wavelength at which order m crosses line x are computed in a multi-step process. First, the dispersion formula for x is rewritten into a more convenient form:

$$\lambda^2 (mA_6 + m^2A_2) + \lambda (mA_1 + m^2A_5 + A_4) + (A_0 + mA_3 - x) = 0$$

(Equation 29)

This equation, which is of the form $a\lambda^2 + b\lambda + c = 0$, may be solved for λ subject to several physical contraints. The solution λ_x must be real, positive, and must lie within the main echelle ripple lobe for order m. That is, $\lambda_{min} \leq \lambda_x \leq \lambda_{max}$ where $\lambda_{min} \equiv \frac{K}{m}(1 - \frac{1}{m})$, $\lambda_{max} \equiv \frac{K}{m}(1 + \frac{1}{m})$ and K is a constant input to the program. Depending on the values of a, b, and c, there may be 0, 1, or 2 such roots. If there is no allowable root, the program proceeds to the next echelle order. If there are two allowable roots, λ_x is chosen to be that root closest in value to a benchmark wavelength WAV. (For any order m, WAV is initially set to λ_{min}; if a final wavelength solution (see discussion to follow) existed for that order on the preceding image line, however, then WAV is set to that value instead.)

Given an allowable solution λ_x to Equation 29, the sample direction dispersion formula in Equation 27 is used to evaluate y_x, the sample number corresponding to wavelength λ_x, and echelle order m. At this point, a check is made to insure that the point (x, y_x) lies within the 25 mm diameter sensitive area of the camera faceplate. If y_x is such that this condition is not satisfied, the program proceeds directly to consideration of the next echelle order. Assuming it is satisfied, the integer sample number Y closest to y_x is calculated. Then a similar series of operations is performed starting with order m and sample Y as inputs: an equation similar to Equation 29, except with the A_i replaced by the B_i and with x replaced by Y, is solved for an allowable root λ_y. The line-direction dispersion formula in Equation 26 is then used to evaluate x_y, the line number corresponding to wavelength λ_y and echelle order m.

From these two sets of data, an interpolated final wavelength solution is calcuated as follows:

$$\text{let} \quad \alpha \equiv \tan^{-1}\left[(y_x - Y)/(x_y - x)\right]$$

$$d \equiv \left[(y_x - Y)^2 + (x_y - x)^2\right] \qquad \text{(Equation 30)}$$

$$\beta \equiv (x_y - x)\cos\alpha$$

$$\text{Then} \quad \lambda \equiv \lambda_y - \beta\,(\lambda_y - \lambda_x)/d \qquad \text{(Equation 31)}$$

is the final wavelength solution for the pixel at line x, sample Y, where echelle order m intersects line x.

For each image line x, values of λ and Y are sought for the intersections of relevant echelle orders m. Each camera has associated with it a list of 65 possible echelle orders, in the sequence in which they appear on the image, which is used in the search. The first seven-line processing pass is done using all 65 allowable orders; the lowest and highest order numbers for which

complete solutions were obtained are recorded, and the next new line is searched for intersections only with orders within the above lowest to highest range, extended by four orders at each end. Each new line is thereafter searched using only orders within the range defined by the lowest and highest orders found for the preceding line, plus four at each end.

c. Assignment of Order and Wavelength for Off-Order Pixels

For each image line, once the intersections of the centers of the echelle orders with that line are determined, all other pixels are associated with the nearest echelle order on that line (Figure 4-7). Wavelengths are set by assigning the wavelength of the nearest order along a 45 degree line. Figure 4-8 illustrates schematically the situation for the seven image lines currently being processed. In this representation, pixels are shown as points. The pixel labeled two would be assigned a wavelength equal to the simple mean of the wavelengths for the order-intersection pixels three and four. Pixel one would in turn be assigned the wavelength of pixel two. Similarly, pixel seven would be assigned a wavelength equal to the mean of the wavelengths of pixels five and six, and pixel eight would be assigned the same wavelength as pixel six. In this way, for a given order, all pixels along a 45 degree line approximately perpendicular to the order would be assigned the same wavelength, as one would expect. Of course, wavelength assignments must be consistent with the nearest-order associations described above; pixels can receive wavelength information only from the order with which they have been associated previously. Furthermore, wavelength assignments are contingent upon final wavelength solutions having been successfully obtained for the relevant on-order pixels.

d. Sensitivity corrections

Each pixel which has had an echelle order and wavelength associated with it may have its intensity value corrected for the wavelength sensitivity of the UV converter and the echelle blaze variation. The former correction is currently made as a function of wavelength only (i.e., no pixel-to-pixel positional

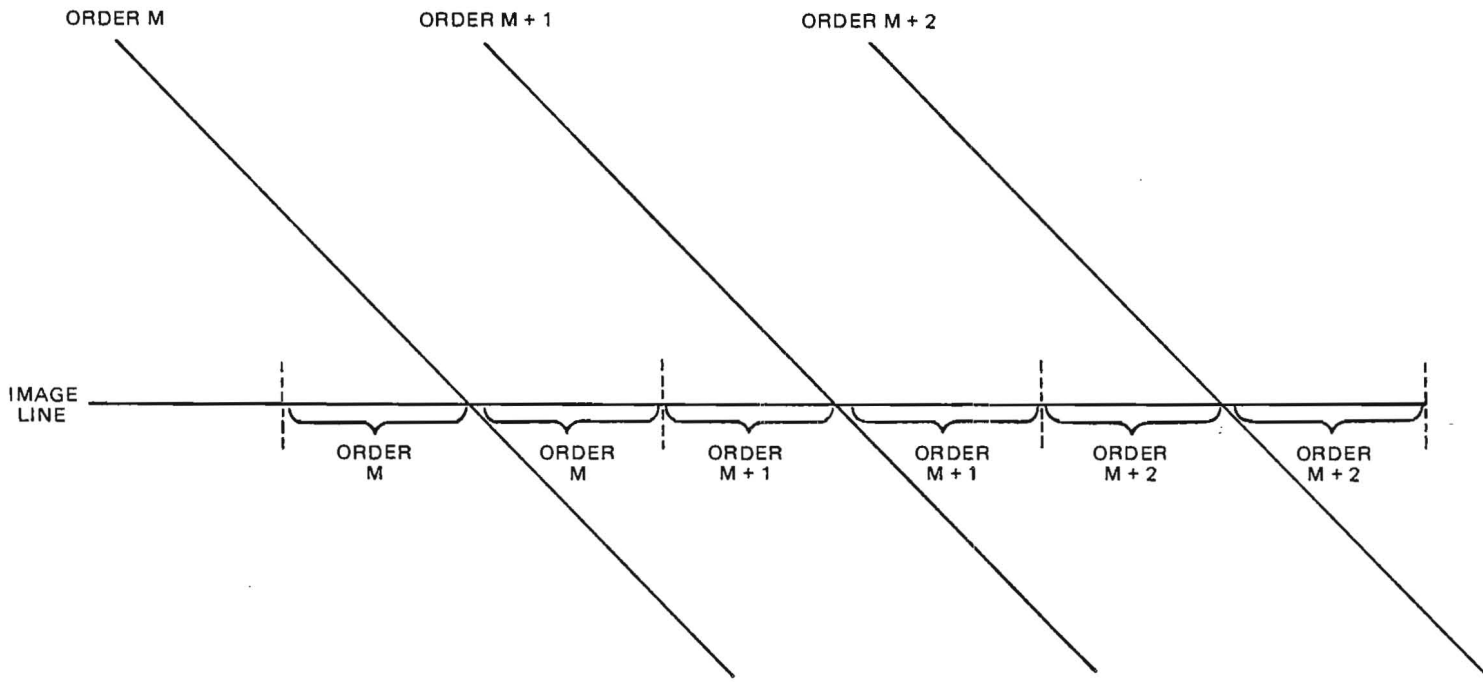

Figure 4-7. Echelle-Order Assignment in DATEXT

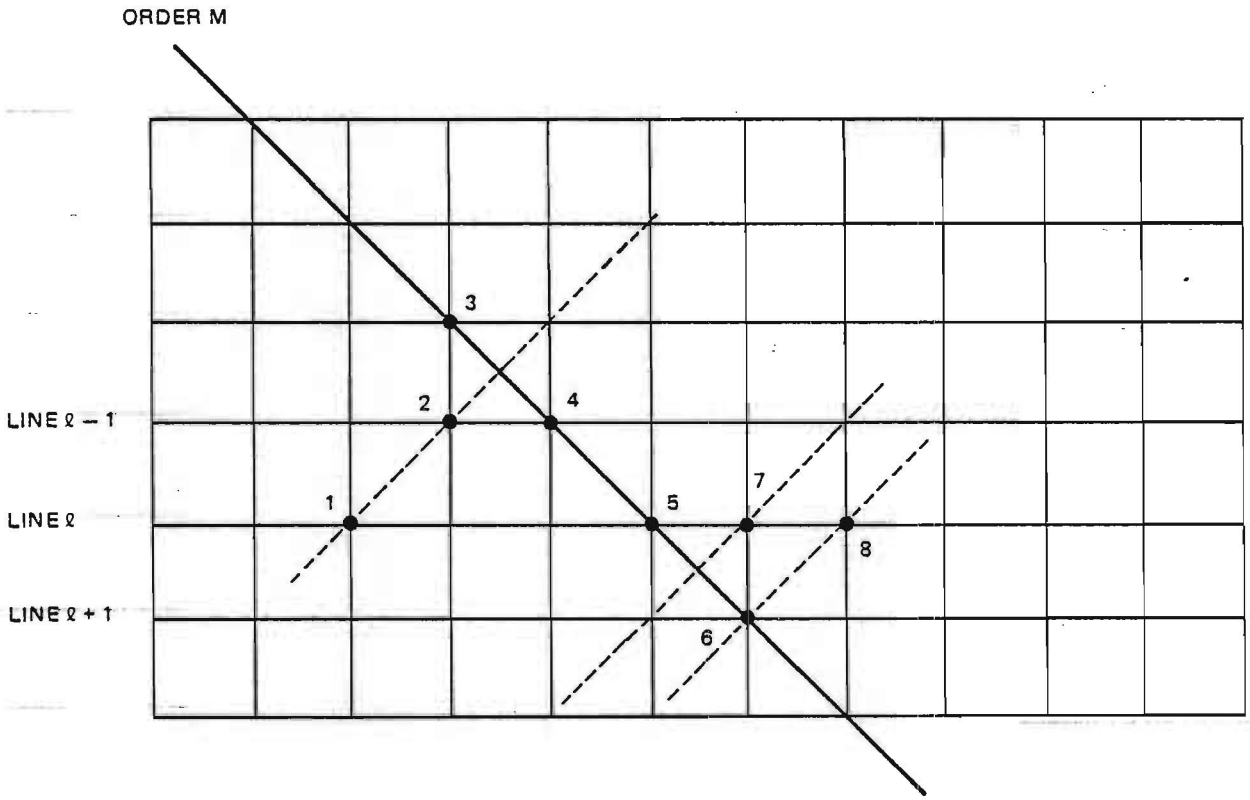

Figure 4-8. Wavelength Assignment in DATEXT

variation). A table of relative logarithmic sensitivities, a_λ, at 12 wavelengths spanning the observed wavelength range is input to the program, and the appropriate a_λ at any intermediate wavelength is computed by simple linear interpolation within the a_λ table. The correction to be applied to the pixel intensity is a multiplicative factor of the form 10^{a_λ}.

The current preliminary echelle ripple sensitivity correction is also a multiplicative factor, r, which depends on wavelength λ and order number m.

Define $W \equiv K/m$ where K is as in section b.

$F \equiv W/m$ (Equation 32)

$\theta \equiv \pi(\lambda - W)/F$

Then if $\theta = 0$, $r = 1$

if $\theta > .90797862\ \pi$, $r = 0$ (Equation 33)

otherwise, $r = (\theta/\sin\theta)^2$

Combining both corrections, output intensity $I' = I_{input} \cdot 10^{a_\lambda} \cdot r$.

e. Artificial Slit Integration of Flux

The analyzing slit which is effectively passed along each order is essentially $\sqrt{2}$ pixels wide, with length $\sqrt{2}$ x IHT pixels; IHT is specified as an input parameter with a maximum and default value of seven. The slit is oriented at 45 degrees to the line and sample directions and is placed at the intersection of each echelle order and the fourth line of the seven currently being processed (this includes only those orders for which an acceptable solution, as described in section b, was found). Figure 4-9 illustrates an analyzing slit with IHT = 7 placed at the intersection of echelle order m and the current fourth line. Here, individual pixels are shown as squares, and all pixels with shading contribute to the integrated flux in the slit; those with double shading contribute with weight = 1, while those with single shading contribute with weight = 1/2. Because of this weighting, the effective slit boundaries are as shown by the dashed line

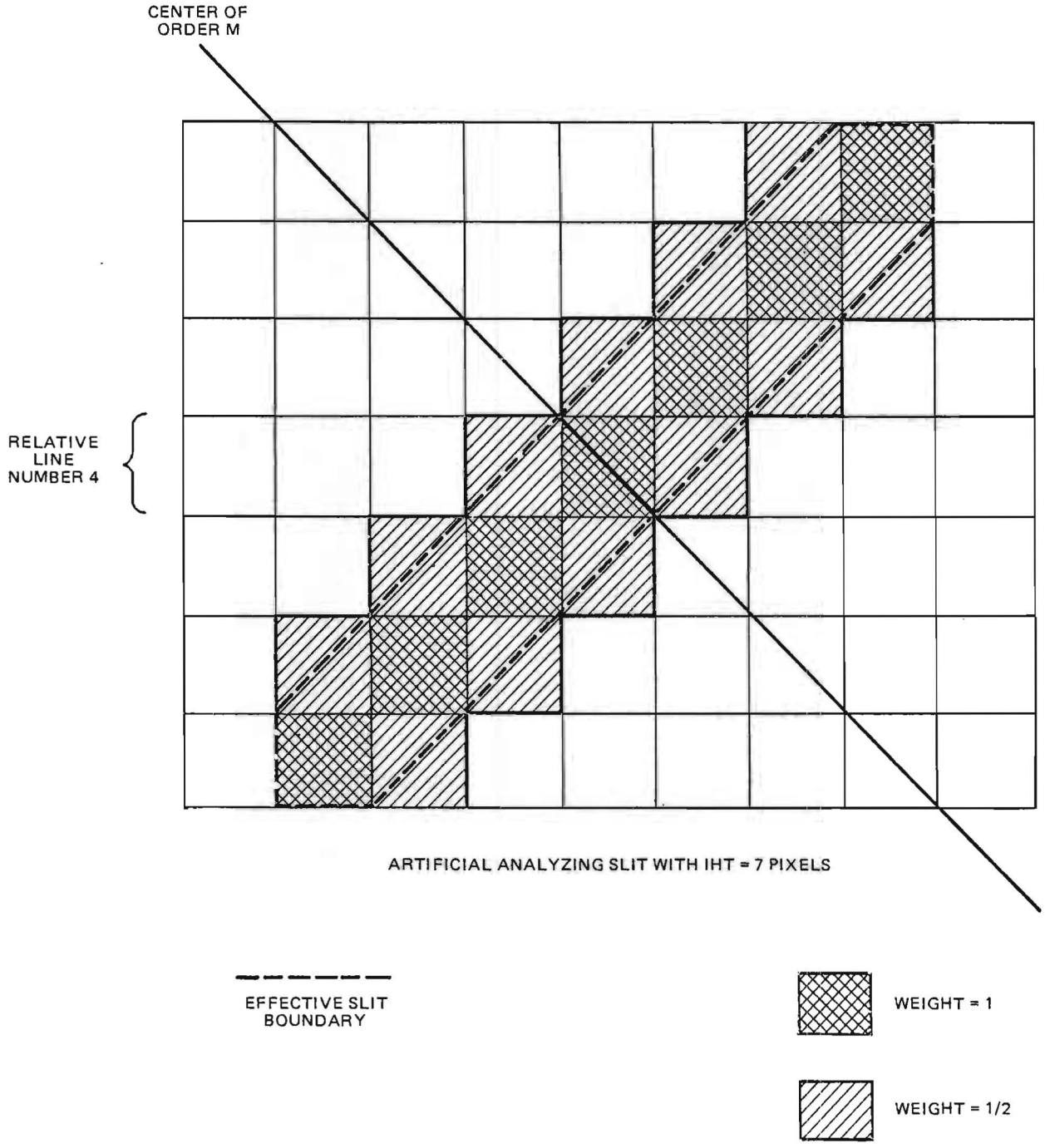

Figure 4-9. Analyzing Slit in DATEXT

in Figure 4-9. The output slit-integrated flux is the weighted sum of the pixel intensities as outlined above; the associated wavelength is the final wavelength solution for the intersection of order m and the current line four; and a qualitative error measure of the output flux is computed as described below.

Each contributing pixel is checked by means of the associated quality-bit array for synchronization or parity error. A check is also made to determine if a contributing pixel is within 5 pixels of the predicted center of a fiducial reseau spot. (The position of one reseau spot and the line and sample spacing between successive spots are input as parameters to the program; the spots will define a square grid since the image will have been geometrically rectified.) A quatity ϵ' is calculated from the sum of the error measures for each contributing pixel, with parity error = -200, sychronization error = -100, and reseau proximity error = -800. Pixels with none of the above faults contribute zero error measure. A positive quantity proportional to the distance of the central slit pixel from the tube center is added to the quantity ϵ' as calculated above.

$$\epsilon = \epsilon' + 0.264 \cdot \text{STEP} \cdot \left[(x - x_c)^2 + (Y - y_c)^2\right]^{1/2} \quad \text{(Equation 34)}$$

where STEP = step size for the image
x = line number of slit center
Y = sample number of slit center (Equation 35)
x_c = line number of tube face center
y_c = sample number of tube face center

The step size may be either 1.0, 2.0, 3.0, or 4.0. This quantity is an observer-specified variable characterizing the SEC vidicon read out mode (the number of addresses in the camera read mechanism per pixel of the image). The positive term of ϵ is scaled such that it is always ≤ 100.

Output data consist of the slit-integrated flux, m, λ, and ϵ. Such data may be compiled for up to 60 echelle orders per image line.

APPENDIX A
FUNCTIONAL DESCRIPTION OF IUESIPS ROUTINES

NAME	FUNCTION
ALGY	performs arithmetic and scaling operations on one or two data sets. In addition, the logarithm of the output may be taken.
APODIZE	creates cosine bell roll-off of an image starting at some radius. It is used to remove edge effects in circular images before doing a two-dimensional FFT.
ASCANS	produces graphs of pixel amplitude along picture lines on the lineprinter. Up to eight lines can be plotted in one pagewidth, and data may be taken from up to nine data sets and presented in various combinations. Scaling can be automatic, each line having a separate origin, or alternatively a single origin may be used.
AUTOSAR	finds pixels whose DN deviates by more than a user-specified tolerance from the average of the corresponding pixels in the immediately preceding and succeeding lines. It replaces those pixel DN values with the average.
AX	removes periodicities from a data string at specified frequencies.
BOXAV	does simple box averaging with nonoverlapping boxes in two dimensions. It is useful for reducing data.
BOXFILT	does running box averaging in two dimensions. The box is shifted in position one pixel at a time to produce a low-pass filter. It is useful for smoothing data.
BOXSTATS	generates the standard deviations from the respective mean values of a set of contiguous box areas of an input picture. The mean values and ratios of mean values to standard deviations can be produced. There are two scaling options and halfword images can be processed and generated.

NAME	FUNCTION
COPYCHK	copies an intensity transfer function calibration file, inserting the diffuser transmission reading into the appropriate label.
DATEXT	performs the data-extraction operations of generating a table of instrumental spectral intensity (flux vs. wavelength) and error estimates. It also applies corrections for echelle ripple and wavelength sensitivity variation of UV converter.
DIFFPIC	performs addition or subtraction between images and writes true or enhanced differences.
DISPLAY	displays an image or portion of an image on the lineprinter using double printing to achieve different gray levels.
EXCHAN	does three types of exchanges. In each line, the samples can be exchanged first for last; in an image, all the lines can be exchanged first for last and in an image both the lines and the samples can be exchanged first for last.
EXPAND	enlarges a picture or selected portion of a picture by repeating each sample N times and each (enlarged) line M times.
FFT1	computes a one-dimensional Fourier transform by the Cooley-Tukey FFT routine. As an option, the data may be apodized before transforming.
FICOR1	will take any image and calculate by linear interpolation the flux at constant optical wavelength which produced the image. The interpolation is carried out using the tables produced by VPFCF1.
FLUXC	operates on images previously processed by the program FICOR1 (or MICOR1). It calculates the transmission of the diffuser at the wavelength of the SEC vidicon intensity transfer function calibration file.

NAME	FUNCTION
FNDRES	locates reseau marks on flat field and bar pattern images using previously measured positions as a guide to the search area. These initial positions can be derived by hand or from a previous run of the FNDRES program. The program uses a cross-correlation technique, seeking minima in the correlation matrix. Checks are made to ensure that a reseau has in fact been located. The new reseau positions are stored on disk for use by distortion correction programs.
GEOM	accepts as input the data produced by PATCH1 and corrects for the geometric distortion using a bilinear technique.
INSECT	combines two pictures of unequal size into one composite picture. It may be used to mosaic two pictures or to replace a portion of one picture with a portion of another.
IUEPLOT	generates Cal Comp plot of flux vs. wavelength for IUE images previously processed by DATEXT and IUESORT. It also generates a printout of wavelength, flux and error estimate for each spectral order.
IUESORT	takes a pseudo VICAR image generated by DATEXT (unsorted table of flux vs. wavelength for various orders) and sorts the data into descending spectral order and ascending wavelengths.
JOIN	compresses a 1024 sample line back to 768 samples (see routine SPLIT).
LINMOD	edits and modifies contents of platinum emission-line library (wavelength, order number, relative intensity, sample number, and line number) used by program WAVECAL.
LIST	is used to display pixel values on the line-printer for any chosen area of an image. In addition, histograms can be obtained, together with mean and standard deviations.

NAME	FUNCTION
LITEXFER	generates intensity-transfer characteristics for up to 50 different specified areas within a flat field, using up to 10 input flat fields. It calculates average DN and sigma for each area within each flat field exposure. Areas may overlap.
MASK	outputs an image into tape for hardcopy representation on the Photowrite device. Gray-scale information and annotations are included.
MICOR1	takes any image and calculates by second-order interpolation the flux at constant optical wavelength which produced the image. The interpolation is carried out using the tables produced by VPFCF1.
OPERATE	adds, subtracts, multiplies or divides two input images. It accepts both byte and and halfword input and produces either byte or halfword output.
OSCRIBE	superposes straight lines on echelle spectral images delineating the various echelle orders as determined by the dispersion formulae output by WAVECAL.
PATCH1	sets up the parameters necessary for the distortion correction program GEOM using the reseau grid positions as provided by FNDRES and RESOGRID.
PICAVE	creates an average picture from up to ten input pictures which may be linearly displaced from one another. It can be used to remove grain noise from pictures.
PIXMAL	performs fast arithmetic transformations on either byte or halfword images. Operations include addition, multiplication, subtraction, and division by a constant, logarithms, exponentiation, powers and roots.
PIXPIK	reduces an input picture by extracting every Mth sample of every Nth line from the input picture.

NAME	FUNCTION
POWER	produces one-dimensional power spectra of one or more lines of an image in the form of graphs on the lineprinter. A binary length transform is used.
PRINTDC	prints out contents of the dispersion-constant library determined by the program WAVECAL.
QSAR	adds or subtracts DN values to pixels in arbitrary rectangles of a picture. Used to correct flaws and in conjunction with VGEN to generage test pictures.
REMRES	erases reseau marks in a picture by replacing the pixels comprising the reseau marks by an average of the surrounding pixel values.
RESODIFF	determines the point-by-point differences between two reseau sets of the same grid structure and outputs these differences for possible further processing. An option exists to produce histograms of these differences.
RESODISP	fits by least squares a square to the central nine reseaux. The 'correct' positions of all the other reseaux are then calculated from that square, and the deviations of the observed positions are displayed.
RESOEDIT	provides a facility for changing, inserting, and deleting reseau positions stored on disk.
RESOFIXL	smoothes the observed set of reseau marks by fitting curves. It discriminates against badly determined reseau marks and interpolates for marks missing entirely.
RESOGENR	adds a set of identical reseau marks to a picture. The reseaux may be of any shape and structure and can be placed anywhere on the picture.

NAME	FUNCTION
RESOGRID	uses the output from FNDRES which gives reseau locations as pixel line and sample and generates the reseau grid description. This describes the line and column of each located reseau in the rectangular grid in a form suitable for the programs PATCH1 and RESOPLOT.
RESOINTP	The observable reseau marks lie inside the circle defined by the faceplate. If an image is corrected using these reseaux directly, spurious discontinuties are introduced near the edge of the tube. To overcome this, the program extrapolates the observed reseaux locations to provide a complete rectangular array over the whole of the picture area.
RESOLIST	lists the DN values within up to 169 15-pixel areas surrounding a set of pixel values--(line, sample) pairs-input into the program. The (line, sample) values defining the centers of each area can be input as parameters or as a secondary input data set. The area centers are first sorted into increasing line order and, if line values are equal, into increasing sample order. The intended use of this program is to list picture areas surrounding reseau marks. Up to six areas can be printed on one page.
RESOLOSE	Reseau marks lie in an almost rectangular grid. This program removes certain subsets of reseau marks from such sets, either actually removing them or replacing the coordinates of "removed" reseau marks with (0, 0).
RESOPLOT	takes the reseau positions as produced by RESOGRID (or input as parameters) and displays them in two-dimensional form on the lineprinter.

NAME	FUNCTION
RESOTRAN	performs linear transformation on a set of reseau-mark coordinates. The transformations possible are rotations, translations, and dilations.
ROTATE	rotates images or segments of images 90 degrees clockwise or counterclockwise.
SAR	removes unwanted lines and blemishes by replacing a rectangle with the average of nearby points. It can also be used to copy images.
SCALP	reduces all periodicities within a specified range of frequencies to the background level. It acts on the complex images that are produced by FFT1.
SPLIT	expands image lines to temporally reconstruct the data stream. This makes allowance for the fact that image lines are not received continuously from the IUE spacecraft.
STRETCH	changes the point-by-point intensity of a picture by generating a transfer function on the domain of DN values.
UNFORM	normalizes the flux distribution falling on the faceplate and provides a calibration of the ultraviolet source. Since the photo diode will read the total flux over a circular area, the normalization is carried out by first finding the average flux in the image over that same area and then dividing all elements by that average. Two output images are produced, the first in half-word form for use by the photometric calibration programs and the other in byte form for display.
VGEN	generates a picture. The picture may be a flat field or have linear line and column increments in pixel values.

NAME	FUNCTION
VPFCF1	reads a set of exposures taken at increasing exposure times and combines the pictures into a table. It ensures that the table has monotonically increasing entries for all pixels. The table is stored on disk or tape with a header block containing information about the wavelength and exposure times used which are extracted from the header labels of the source tapes. This program's primary use is to construct the photometric transfer function tables.
WAVECAL	determines the dispersion constants relating the positions of spectral lines across the IUE image (line and sample numbers) to their wavelengths and echelle orders.
WRL	writes reseaux locations (read from a parameter list) to an old file holding descriptions of the reseaux pattern existing on numbered tubes. Files may hold 'observed locations' or 'corrected locations'. The 'observed locations' may be updated by program PATCH1 to prepare input for the program GEOM which will perform geometric distortion corrections on images taken by one of the numbered tubes.

REFERENCES

1. IUESIPS USERS/OPERATORS GUIDE (in press).

2. Brenner, N. M. 1967, Lincoln Lab. Techn. Note 1967-2, MIT

3. Cooley, G. W., Tukey, J. W., Math. Computation 19, 287, 1967