IUESIPS ENHANCEMENTS

Prepared for

GODDARD SPACE FLIGHT CENTER

By

COMPUTER SCIENCES CORPORATION

Under

Contract NAS 5-24350
Task Assignment 670

Prepared by:

B. E. Turnrose 4/13/79
B. E. Turnrose Date

A. D. Mallama 4/17/79
A. D. Mallama Date

C. A. Harvel 4/17/79
C. A. Harvel Date

B. R. Dantzler 4/23/79
B. R. Dantzler Date

Approved by:

P. M. Perry 4/16/79
P. M. Perry Date
Section Manager

R. F. Williams 4/16/79
R. F. Williams Date
Department Manager

ABSTRACT

This document reports the scientific analysis and software support provided to the Laboratory for Astronomy and Solar Physics for the improvement of the International Ultraviolet Explorer Spectral Image Processing System (IUESIPS) in the performance of Task Assignment 670 (IUESIPS Enhancements), Contract NAS 5-24350, during the period from August 1978 to February 1979.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

Section 5 (Cont'd)

LIST OF ILLUSTRATIONS

Figure

LIST OF TABLES

Table

LIST OF TABLES

Table (Cont'd)

SECTION 1 - INTRODUCTION

1.1 BACKGROUND

The International Ultraviolet Explorer (IUE) spacecraft has been in operation for one year, as has the software system which translates raw IUE images into meaningful astronomical data, the International Ultraviolet Explorer Spectral Image Processing System (IUESIPS). A number of deficiencies have long been identified in the routine processing provided by IUESIPS, and from the perspective of the first several months of operation of IUESIPS, an effort was mounted to improve the data reduction capabilities of IUESIPS so that it might more adequately cope with the known instrumental effects impressed upon IUE images. This effort was embodied in the six-month Task Assignment 670 (IUESIPS Enhancements) under which the Computer Sciences Corporation (CSC) has provided both scientific analysis and software development relevant to the delineation and implementation of system improvements.

1.2 OBJECTIVES

The purpose of this document is to report the task activities which have been performed to effect the various system enhancements. It is hoped that the consolidation here of the relevant results in each specific scientific area addressed (see Section 1.3) will clarify not only the work performed but also the possible directions in which future analyses might profitably proceed.

1.3 SCOPE

This document describes the task activities within the four principal areas of concern:

1. Wavelength calibration procedures
2. High dispersion reduction procedures
3. Low dispersion reduction procedures
4. Geometric and photometric correction procedures

In some of these areas (most notably the last), task activities were maintained at low levels in accordance with various IUE Project three-agency agreements regarding assignments of primary responsibilities among the agencies. In such cases, these circumstances are documented herein, and, whenever possible, suggestions for future development are presented.

SECTION 2 - WAVELENGTH CALIBRATION PROCEDURES

2.1 CALIBRATION DATA BASE

IUE wavelength calibrations are performed using platinum emission-line spectral images which allow a correspondence between wavelength, spectral order, and pixel location to be defined for all spectral formats. All such platinum-line wavelength calibration (WLC) images are processed by the applications program WAVECAL2, which fits simple dispersion relations to the observed spectral format by regression analysis. Operationally, the dispersion relations are fully described by the set of fitted parameters (coefficients in the dispersion relations) called dispersion constants.

An important ancillary input to the calibration procedure is the so-called "line library," or calibration data base, relevant to the particular mode (dispersion and camera) under consideration. Each line library is comprised of a list of platinum emission lines (each line is designated by an arbitrary identification number generally called simply the "line number") and corresponding laboratory wavelengths and echelle order numbers. The wavelength identifications in the line libraries form the basis for the entire calibration and thus are pivotal quantities. In the following sections, the analyses of these identifications in the various line libraries are discussed.

2.1.1 Low Dispersion

2.1.1.1 Short Wavelength Prime (SWP) Line Library

In low dispersion a linear relationship between pixel location and wavelength is the standard dispersion relation in current use. In this case, the task of wavelength calibration reduces to the fitting of four constants: a zero point and a scale factor for both the image line and sample directions (see Equations 2-4 and 2-5). In the specific case of the SWP calibration, the calculated dispersion constants were observed to vary in a more or less random fashion,

with the scale factors exhibiting excursions of up to ±2 percent. The nature of these variations, coupled with the relative constancy of the actual locations of the five or six platinum lines measured on each calibration image to provide a starting point for the regression analysis, was indicative of a problem in the data base rather than a genuine variation to the scale of the spectral format. Accordingly, a number of different calibration solutions were analyzed to identify the deficiencies in the line library. As a fiducial, the solution for SWP 2025 was used, it having been previously identified as a reasonably accurate solution on an empirical basis by comparing known wavelengths in spectra of planetary nebulae with the wavelengths assigned by this solution. The scale-factor constants for this solution are as follows:

Sample direction scale factor: - .46732999
Line direction scale factor: .37548275

The wavelength calibration (WLC) images used in the line library analysis are listed in Table 2-1 along with the ratios of the computed scale factors to the scale factors for SWP 2025, and the standard deviations (σ) of the solutions in both the line and sample directions, in pixel units. Since some of the original WAVECAL2 solutions for the images in Table 2-1 were allowed to include terms quadratic in wavelength, a linear least squares routine was written and implemented on the IBM S/360 computers to recalculate strictly linear solutions for all eight images to facilitate intercomparison; Table 2-1 uses these results.

Examination of the WAVECAL2 runs for these images showed that in some cases the same pixel positions were calculated for several pairs of adjacent platinum lines. The lines in these pairs were less than 12 angstroms apart. These lines were subsequently removed from the line library, as described below.

Table 2-1. Parameters of SWP Low Dispersion Wavelength Solutions With Original Line Library

IMAGE NUMBER	SAMPLE SCALE FACTOR RATIO	σ(SAMPLE)	LINE SCALE FACTOR RATIO	σ(LINE)
1535	0.9972	0.75	0.9984	0.65
1212	1.0073	3.31	1.0078	2.96
1202	1.0086	3.90	1.0130	2.80
1234	1.0226	3.52	1.0223	2.44
1455	0.9997	3.33	1.0007	2.48
1834	0.9945	2.74	0.9930	2.15
2138	1.0184	1.77	1.0176	1.53
2190	1.0154	1.95	1.0166	1.73

The WAVECAL2 solutions for the set of WLC images were further examined to attempt to correlate the inclusion of certain platinum lines with the occurrence of poor solutions. In this way, a list of "suspicious" entries in the data base was compiled, and several test executions of WAVECAL2 were made to determine the degree of improvement resulting from the exclusion of such lines. Finally CalComp plots of an extracted platinum spectrum and photowrite hardcopy images were examined to determine what, if anything, was improper about the suspect entries in the line library. The results were the following:

- All lines below line 8 were too faint
- Line 15 was contaminated by a reseau
- Lines 19 and 20 were a blend
- Lines 24, 25, and 26 were a blend
- Lines 28 and 29 were a blend
- All lines above line 33 were cut off of the SWP image
- Lines 22 and 29 were misidentified, with the correct wavelengths unknown
- Lines 31 and 32 were misidentified, but the correct wavelengths were known

As result of this analysis, lines 15, 19, 20, 22, 24, 25, 26, 28 and 29 were removed from the line library. All lines below line 8 and above line 33 were also removed, and lines 31 and 32 were properly identified. This modified line library gave consistently excellent results, as can be seen in Table 2-2. Later, it was found that line 32 was used less than half of the time by WAVECAL2, so it too was removed from the line library. Table 2-3 shows the final SWP low dispersion line library now in standard use.

Table 2-2. WAVECAL2 Results Using Improved SWP Low Dispersion Line Library

IMAGE NUMBER	NUMBER OF PLATINUM LINES USED	σ(SAMPLE)	σ(LINE)	SAMPLE SCALE FACTOR	LINE SCALE FACTOR	SAMPLE SCALE FACTOR RATIO	LINE SCALE FACTOR RATIO	SAMPLE ZERO POINT	LINE ZERO POINT
1535	11	0.59	0.52	-.46603244	.37562867	0.9972	1.0003	.97903861E3	-.26322392E3
1212	14	0.30	0.31	-.46632053	.37563376	0.9978	1.0004	.97966946E3	-.26340385E3
1202	14	0.36	0.48	-.46622410	.37594431	0.9976	1.0012	.97949527E3	-.26406980E3
1234	14	0.49	0.60	-.46820332	.37605965	1.0019	1.0015	.98351848E3	-.26347848E3
1455	12	0.34	0.42	-.46640896	.37550197	0.9980	1.0001	.98004251E3	-.26329656E3
1934	14	0.48	0.55	-.46684653	.37485095	0.9991	0.9983	.98150905E3	-.26181985E3
2133	12	0.32	0.31	-.46564309	.37536507	0.9964	0.9997	.98091529E3	-.25983874E3

Table 2-3. Final SWP Low Dispersion Line Library

LINE NUMBER	WAVELENGTH (Å)
8	1380.494
9	1403.896
10	1429.230
11	1482.829
12	1509.288
13	1524.725
14	1554.900
16	1604.010
17	1621.658
18	1635.210
21	1723.128
23	1753.822
27	1812.940
30	1883.051
31	1913.230
33	1971.520

Test executions of WAVECAL2 were made in order to determine the advisability of allowing terms quadratic in wavelength to enter the dispersion relations in the low dispersion mode, using the new data base. Whereas it had been previously determined that properly executed linear solutions well represent the true dispersion in this mode (note the small standard deviations in Table 2-2), it was found that the quadratic results were erratic in their agreement with the linear results. On this basis it was determined that quadratic solutions were potentially detrimental and therefore not to be used. Table 2-4 illustrates the inconsistent results obtained in the quadratic solutions by comparing the predicted line and sample pixel locations corresponding to the wavelengths 1000 Å, 1500 Å, and 2000 Å for the linear and the quadratic cases for the eight WLC images under study. The unreliable results yielded by the quadratic tests are attributed to the inherent instability of extrapolating a quadratic solution well beyond the range of the input data base, particularly at short wavelengths.

2.1.1.2 Long Wavelength Prime (LWR) Line Library

There were no known a priori problems with this line library, but it was examined to see if any improvements could be made. Analysis of WAVECAL2 runs using this library showed that lines 13, 14, 15, 24, 26, and 30 were never found by the program and that line 38 was almost never found. A CalComp plot of an extracted WLC spectrum and a photowrite hardcopy image were examined, and the following conclusions were drawn:

- Line 14 was misidentified
- Lines 13, 15, and 24 were too faint
- Lines 26 and 30 were blended with other lines
- Lines 38 and 39 were a blend

Consequently, these lines were removed from the LWR low dispersion line library leaving the revised data base listed in Table 2-5.

Table 2-4. Comparison of Linear and Quadratic Fits for SWP Low Dispersion

	SAMPLE NUMBER LINEAR			SAMPLE NUMBER QUADRATIC			LINE NUMBER LINEAR			LINE NUMBER QUADRATIC		
[illegible]	λ = 1000	λ = 1500	λ = 2000	λ = 1000	λ = 1500	λ = 2000	λ = 1000	λ = 1500	λ = 2000	λ = 1000	λ = 1500	λ = 2000
1035	513.01	279.99	46.98	555.38	279.08	36.92	112.41	300.22	488.04	81.24	300.83	449.91
1212	513.35	280.10	47.03	512.81	280.17	47.26	112.23	300.04	487.86	111.38	300.05	487.60
1200	513.27	280.16	47.05	513.27	280.16	47.05	111.87	299.84	487.81	108.13	299.91	487.15
1234	515.34	281.22	47.12	503.23	281.54	40.58	112.58	300.61	488.64	103.39	300.53	492.45
1455	513.63	280.43	47.22	538.03	279.41	62.53	112.20	299.95	487.70	124.70	299.39	487.78
1834	514.61	281.16	47.71	516.99	281.10	48.05	113.03	300.45	487.88	109.62	300.53	487.39
2139	515.27	282.45	49.63	516.22	282.47	48.98	115.52	303.20	490.88	120.67	303.00	494.38
2190	513.58	281.11	48.64	490.97	281.99	30.04	113.37	301.25	489.14	133.79	300.50	504.21

Table 2-5. Final LWR Low Dispersion Line Library

LINE_NUMBER	WAVELENGTH_(Å)
10	1913.230
11	1937.840
12	2037.119
16	2144.920
17	2175.360
21	2290.710
23	2440.797
25	2489.157
27	2539.968
28	2628.815
29	2703.867
31	2734.770
32	2772.490
33	2793.965
34	2830.128
35	2876.430
36	2896.469
37	2930.652
40	3000.790
41	3065.608

2.1.2 High Dispersion

There were no known problems with the high dispersion line libraries, but an investigation of them was initiated to see if any improvements could be made. Analysis of 21 WAVECAL2 runs showed that for the SWP library, 65 lines were found less than half of the time, and 29 lines were never found. For the LWR library, 57 lines were found less than half of the time, and 15 lines were never found in a total of 24 WAVECAL2 runs. It has not yet been determined why WAVECAL2 has difficulty finding these lines, which are listed in Tables 2-6 and 2-7.

Further analysis of the high dispersion wavelength calibration procedures on another front has recently begun. A set of 26 high dispersion WLC images obtained by D. A. Klinglesmith of GSFC over a several week period in August 1978 is being studied to seek possible additions to the high dispersion line libraries and to ascertain the extent to which the wavelength calibration of target images may be improved by more frequent measurement of WLC images. In this analysis the 26 WLC images are being reduced as normal target images using the standard wavelength calibration appropriate to the time period of their acquisition. This investigation will be continued as part of a follow-on task.

2.2 CORRECTION PROCEDURE FOR OLD SWP LOW DISPERSION CALIBRATIONS

Prior to day number 221 of 1978 the line library used for the SWP low dispersion wavelength calibration was in error, and therefore the wavelengths assigned to extracted data during this period are also in error. In order to correct these wavelengths without reprocessing spectra already extracted a correction formula has been derived, which can be used to obtain corrected wavelengths as a function of (1) the old (incorrect) dispersion constants, (2) the new (correct) dispersion constants, and (3) the old (incorrect) wavelength. The new dispersion constants to be used here are those calculated using the new

Table 2-6. SWP High Dispersion Library Lines Found Less Than Half the Time by WAVECAL2

LINE NUMBER	LINE NUMBER	LINE NUMBER
6	148*	252*
12	1531	254
14	1549	257*
16	1559	258
18	156*	259
19	1659	261
22	167*	266*
24	173	267*
34	180	269*
37	187.	271*
43	192*	275
56	206	276*
59	214	281*
64	229	282*
889	235*	283*
100	241*	285*
102*	242	287*
110*	243*	288*
1211	244*	289*
128*	248	293*
1301*	249	294
138*	250*	

*Line was never found.

Table 2-7. LWR High Dispersion Library Lines Found Less Than Half the Time by WAVECAL2

LINE NUMBER	LINE NUMBER	LINE NUMBER
860	52	155
861	521	1552
862	551	1553*
863*	561*	164
864	623*	1811
865	64	183
866*	67	187
869*	672	1891
870*	674	1951
23	68	204
34	69	5212
351	73	2122
353	773*	214
362	83*	850
37	112	952*
381	114*	855
382	136*	856*
44	149	857*
50*	153	859

*Line was never found.

(corrected) line library (see Section 2.1.1.1) and the same platinum calibration image that was used to determine the old dispersion constants. The old dispersion constants needed by the formula are those printed on the CalComp header label.

The correction formula has the following form:

$$\lambda = d + m\,\lambda_0 \tag{2-1}$$

where λ and λ_0 are the new and the old wavelengths respectively and d and m are constants defined in terms of the old and new dispersion constants

$$m = \frac{b_2\, b_2' + a_2\, a_2'}{b_2^2 + a_2^2} \tag{2-2}$$

$$d = \frac{b_2\,(b_1' - b_1) + a_2\,(a_1' - a_1)}{b_2^2 + a_2^2} \tag{2-3}$$

where primed values are the old dispersion constants and un-primed values are the new constants. The a's are the dispersion constants defining the location of the extracted pixel in the sample direction and the b's are the analogous constants for the line direction, such that,

$$\text{sample number} = a_1 + a_2 \cdot \lambda \tag{2-4}$$

$$\text{line number} = b_1 + b_2 \cdot \lambda \tag{2-5}$$

The new and the old set of dispersion constants each define a locus of points (the dispersion line) in the (sample number, line number) plane, which should

follow the low dispersion spectral order. A point on either of these loci has an associated wavelength: λ for the new dispersion constants, λ_0 for the old constants. The wavelength correction formula was derived such that it assigns to a given λ_0, associated with a point (s_0, l_0) on the old dispersion line, the wavelength λ, associated with a point (s, l) on the new dispersion line, where the point (s, l) is that point on the new dispersion line closest to (s_0, l_0). See Appendix A for a derivation of the formula.

The accuracy of the formula is being tested by using it to correct the wavelengths of emission lines obtained from spectra extracted using several different sets of the "old" dispersion constants. The corrected wavelengths, un-corrected wavelengths and the true wavelengths can then be compared. The results of these tests will be published in the IUE Newsletter, along with a complete table of "good" dispersion constants.

2.3 DSPCON: AUTOMATIC REGISTRATION OF SPECTRAL ORDERS

In addition to the calibration refinements discussed above, a program has been introduced to effect automatic registration of the spectral orders. This program replaces the operator intervention which is otherwise required to shift the zero-point terms of the dispersion constant data sets so that the coordinates of the spectral orders as computed from the dispersion constants will coincide with the actual locations. If this shift is not applied, either by the operator or by the DSPCON program, the data extraction routines may obtain data that is displaced by several pixels from the centers of the spectral orders.

The use of this program has several advantages over operator intervention. Running DSPCON requires only about fifteen seconds compared to the total of several minutes required for the operator to run the dispersion-relation overlay program OSCRIBE, to make a determination of the shift, and to type in the offsets in the manual mode. Thus, DSPCON speeds up production. Also, the program makes the offset calculation independent of any personal equation

of an operator: whereas DSPCON uses a template to compute the offset and applies it in a direction perpendicular to the dispersion, an operator has to judge the offset by eye and usually applies the correction all in the Y direction or all in the X direction. Lastly, DSPCON puts the computed offsets for both zero-point terms into the output dispersion constant data sets in such a way that they may be added to the labels of the extracted spectra.

A listing of the IUESIPS source code of DSPCON is given in Appendix B.

2.3.1 Method

Program parameters and the initial input dispersion constant data set are entered. Twelve line and sample coordinates where the spectrum is expected to cross are calculated using the input dispersion constants and the default or specially input wavelengths. A sample area of pixels centered on each coordinate is read from the geometrically corrected image. Processing of each area depends on whether the image is trailed or untrailed, but in both cases a "shift" corresponding to the needed change in the dispersion constants is computed.

In the untrailed case a square array of pixels is read in and rotated through 45 degrees to roughly align it with the spectral orders. Then row sums are created, and a template (corresponding to the expected intensity distribution perpendicular to wavelength) is translated sequentially over the array of row sums to find the best correlation. This best fit is transformed to give the shift.

In the trailed case, the sample areas are 3 rows by 40 samples each. Row sums here are generated by adding the elements of the 3 rows whose samples are aligned with the orders. Then a template is applied as in the untrailed case.

In the case that a low signal-to-noise ratio or a saturated sample area is detected, the computed shift is flagged.

Once all twelve sample areas have been evaluated their inferred shifts are each corrected by a small amount corresponding to the difference between the integer pixel numbers sampled and the real number computed (see paragraph two of this section). The corrected shifts are averaged and an rms deviation is computed. The averaged shift is than transformed to the line and sample coordinates needed to refine the dispersion constants. If more than eight samples are flagged due to low signal-to-noise ratio or saturation, or if the rms deviation is greater than 1, the program calls for an abnormal termination to alert the Image Processing Specialist that the shift must be derived by eye. If the processing proceeds normally, the zero-point dispersion constant terms are corrected and written to the first record of the output dispersion constant data set, and this data set is flagged.

2.3.2 EXECUTE Statement Format

EXEC, DSPCON,(IN1, IN2), OUT1,, P;

where

- IN1 is the spectral image
- IN2 is the uncorrected dispersion constant data set
- OUT1 is the corrected dispersion constant data set

2.3.3 Parameters

- Fixed location:

  CAMERA, DISP, AP, TRAIL, RECNO, SPCWAV

where

CAMERA = SWP for short wavelength prime

LWR for long wavelength redundant

DISP = LOW for low dispersion
HIGH for high dispersion

AP = SML for small aperture (not used currently)
LRG for large aperture (not used currently)

TRAIL = NOTR for an untrailed spectrum
TR for a trailed spectrum

RECNO = Integer value of record number of input dispersion constant data set to be read; the default, zero, causes the last record to be used

SPCWAV = FALSE to use default wavelengths
TRUE to indicate that special wavelengths follow

- Keyword (must follow fixed location parameters):

  WAVES, X1, X2, ..., X12

  ORDERS, Y1, Y2, ..., Y12

  X1 etc. are the special wavelengths used to sample data from the images.

  Y1, etc., are the orders corresponding to the wavelengths referred to above; blanks are entered if the spectral image is low resolution.

SECTION 3 - HIGH DISPERSION REDUCTION PROCEDURES

3.1 BACKGROUND REMOVAL ALGORITHMS

3.1.1 The Algorithms

3.1.1.1 Current Production Method--Subtraction of Inter-Order Spectrum

The current production algorithm for high dispersion data reduction corrects the on-order spectral data for background by subtracting the average of the inter-order data on both sides of it. This method has two shortcomings. Firstly, the inter-order data consists of background and a contamination from on-order data. Secondly, the on-order data is contaminated by orders other than itself. The first effect is the more serious, and it results in background values that are too high. The effect of this on the reduced fluxes is that they are too low. This especially true at short wavelengths where the orders are crowded and contamination is at its worst.

3.1.1.2 Heap's Method

The problems with the current production method can be ameliorated if the effects of contamination can be corrected. S. R. Heap of GSFC has created an algorithm and a VICAR program (Reference 1) whereby the known spread of a spectral order can be used to correct on-order and inter-order data for the effects of contamination so that the subtraction results in a flux that is, in principle, more accurate than that resulting from the method described above. The details of this algorithm are described in a working paper by Heap. To put it briefly, two order-profile parameters are defined: F, a measure of the strength of the spectrum at the location of the inter-order extraction, and G, a measure of the strength in the adjacent order where that data is extracted. These values are applied to the inter-order, and on-order data, respectively, to remove contamination, and the subtraction is then performed.

3.1.1.3 Constant Background Method

A third algorithm takes advantage of the relatively flat background that is observed. If we take a value of background from a portion of the image that is relatively uncontaminated by spectral data, and subtract that from the order data we can at least be sure that the subtraction does not include heavy contamination from nearby orders. The problem with this method is that the background is not perfectly constant, and if we are subtracting a large background from a small signal even a few per cent variation can translate into large errors in the resulting flux. Furthermore, it does not treat the problem of on-order contamination from adjacent orders.

3.1.2 Test Procedures

To evaluate the merits of each of the three algorithms described above a FORTRAN program, BKGTST, was devised which creates a small test image, extracts the data, corrects it using the three algorithms being tested, and evaluates the results of each one.

First, an analytic expression made up of cosine terms is evaluated at 500 points to produce a simulated intensity function. This function was designed to resemble a stellar spectrum with a variable continuum, strong and weak absorption lines, and a few emission lines. The dispersion is such that a weak line covers only a few pixels and a strong line covers about twenty pixels. Next, a two dimensional gaussian spread is applied such that the full width at half maximum is 2.4 pixels, roughly the size measured on an IUE spectrum, and the resulting point spread function is distributed in five simulated orders on a 100 x 100 simulated image. An inverse-square point spread function is added to represent halation. The integrated intensity in halation is chosen to equal 0.2 times the intensity of the gaussian, as was found in Snijder's study (Reference 2). Finally, a slowly changing background is added, whose peak-to-peak variation is 10 percent.

Data extraction is accomplished by passing a simulated slit over appropriate portions of the on-order and inter-order data. After the raw data has been extracted, the three algorithms (subtraction of inter-order, Heap's method, and the constant background method) are applied, as described above, to produce three sets of reduced data.

These reduced data sets are compared to the input analytic intensity function to evaluate the results of the three methods. For each method the program computes the average of the function, the rms deviation from the input function, and it plots the residual to the input function (in the sense reduced minus input) on a graph adjacent to a graph showing the analytic function; see Figures 3-1 and 3-2.

3.1.3 Test Results

Our results indicate that substantial improvements can be made by using either the constant background method or Heap's method as opposed to using the current production method of subtracting the inter-order background directly without applying corrections. We have found that the current production method results in a reduced flux that is consistently too low, due to the subtraction of contamination along with the inter-order data, as described in sections above. This is shown by the averages as in Table 3-1.

The constant background method works very well for spectra with a high signal-to-noise (S/N) ratio. This result is to be expected since five percent errors in a small background are small when measured in terms of the signal which is much larger. In the case of high S/N ratio the constant background method appears to work better than the other two. Standard deviations are listed in Table 3-2.

When there is a low S/N ratio, however, Heap's method appears to be the best suited. Her algorithm takes into account the variation of the background by actually measuring it, whereas the constant background method does not. In

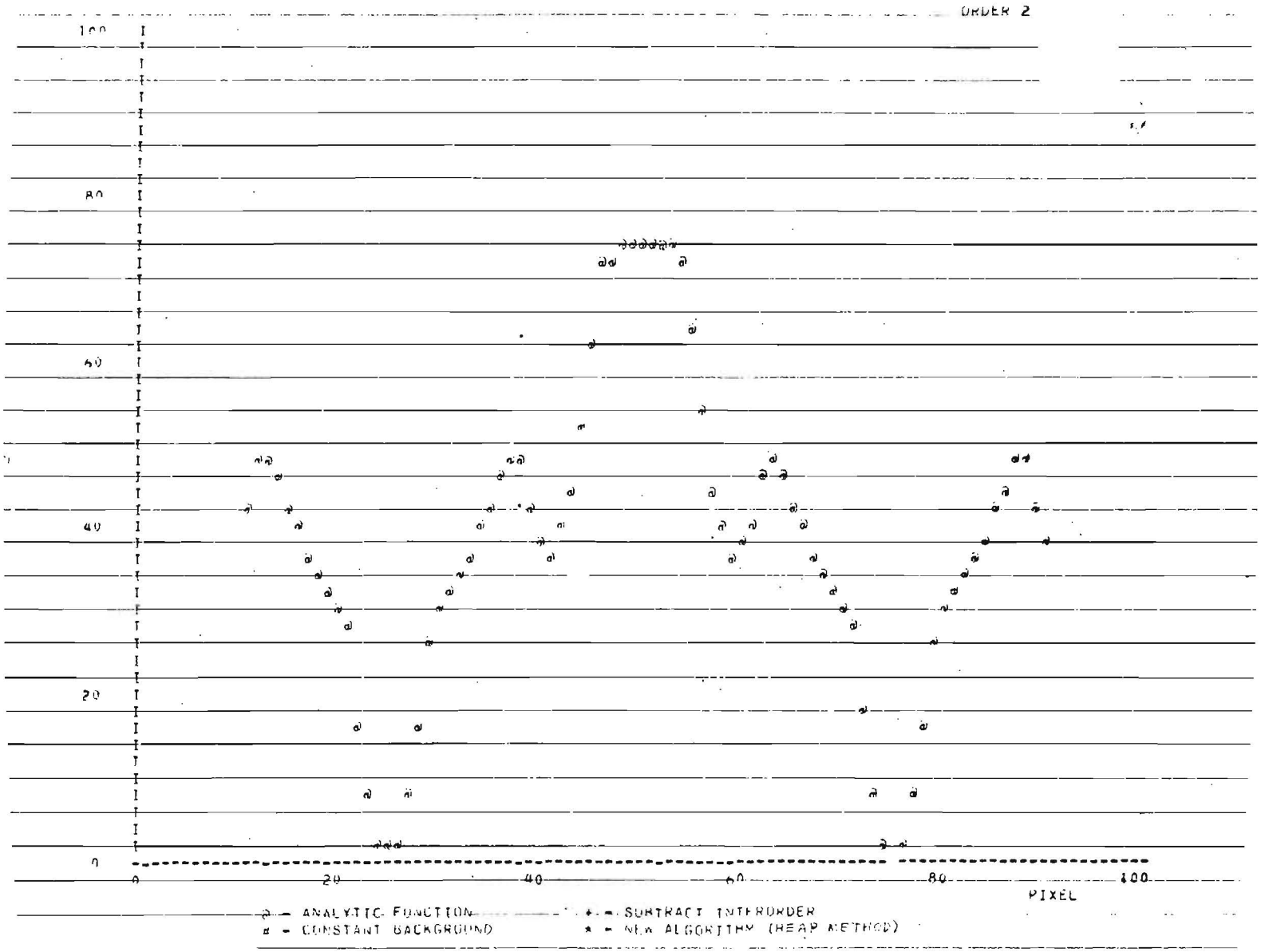

Figure 3-1. The Input Analytic Function for Background Removal Analyses

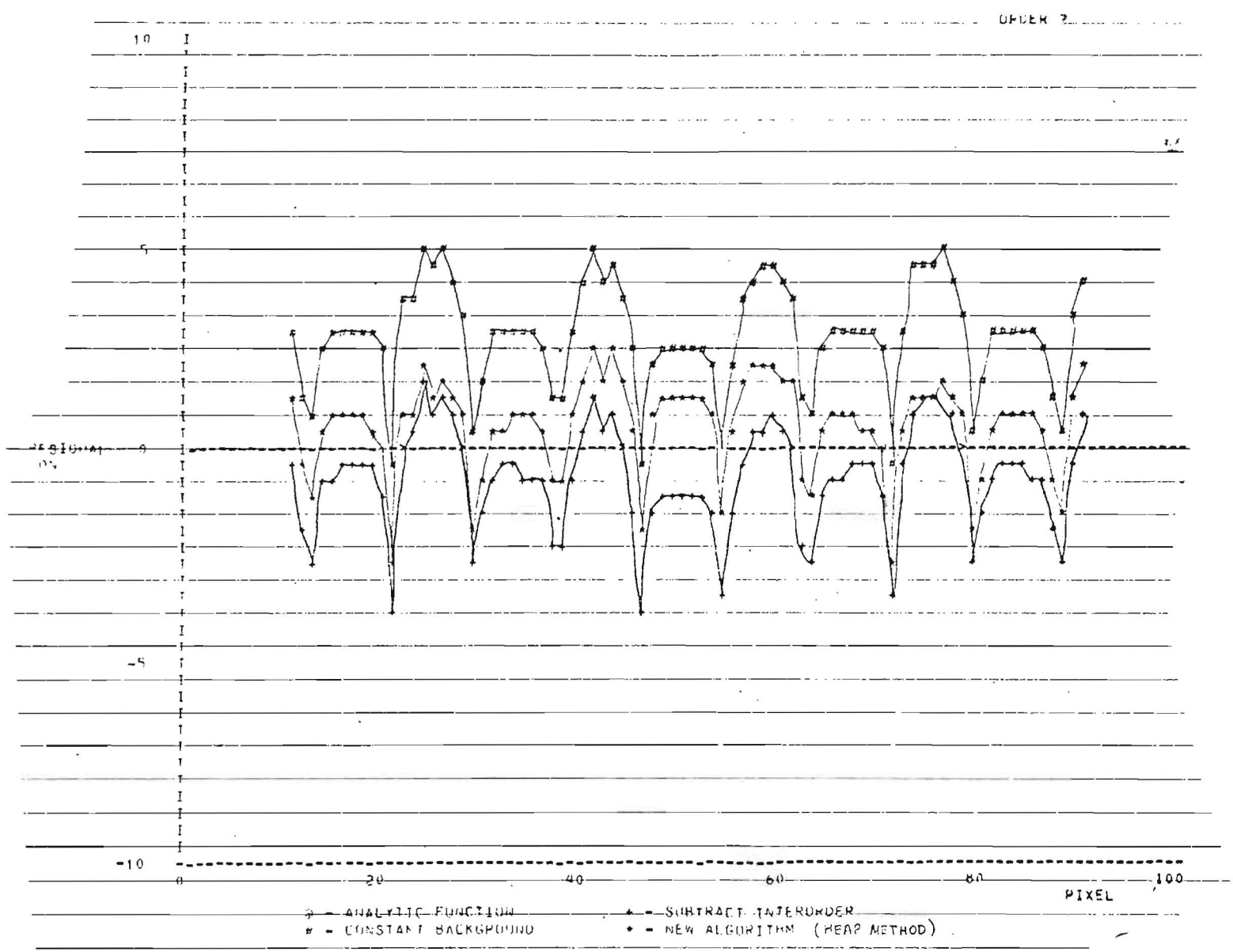

Figure 3-2. Residuals of Various Solutions From the Input Analytic Function

Table 3-1. Average Function Values in Background Removal Analysis
(background $\approx$ 2 x signal, halation included, all orders similar)

Function	Average
Analytic (Input)	39.3
Current Production	37.9
Constant Background	38.9
Heap's Method	40.4

Table 3-2. Standard Deviations for High S/N case in Background Removal Analysis (background $\approx$ 0.2 x signal, halation included, orders 2 and 4 constant, analytic function average = 39.4)

Function	Standard Deviation
Current Production	2.1
Constant Background	1.1
Heap's Method	1.7

the case of a large background, even an error of a few percent in the background can lead to substantial errors in the inferred net signal. Further, Heap's method takes into account the overlap of on-order data from adjacent orders, which becomes appreciable at the short wavelength end of the spectrum. Table 3-3 shows standard deviations for the high background case.

We conclude that Heap's method and the constant background method could substantially improve the data analysis, especially if the former is used for low S/N spectra and the latter for high S/N.

3.2 PDP-11/40 FORTH ANALYSIS

In order to use a program such as the one due to Heap (see Section 3.1) the spread of a spectral order perpendicular to the dispersion (i.e., the order profile) must be known as a function of position in the image. The GSFC Code 685 PDP-11/40 computer and the Johnson Space Center (JSC) FORTH system were used to analyze several images provided by D. A. Klinglesmith in an effort to determine empirically the variation of the order profile. Under the JSC FORTH system, programs were written and debugged to (1) do point-to-point extractions in an IUE image (similar to the "Intensity Plot" routine on the EDS), (2) plot the extracted profiles on the TEKTRONIX terminal using the DECWRITER for control, and (3) store extracted profiles on disk. FORTH programs (words) to contour plot IUE images and to do gaussian curve fitting using least squares were modified and used in this analysis.

It was determined that the one-pixel-high extraction done by the point-to-point profile word produced data which were too noise contaminated to use in the order profile analysis. Therefore, efforts to modify this word so that it could use an extraction slit of any desired height are currently in progress.

Some sample outputs from the system are shown in Figures 3-3, 3-4, 3-5, and 3-6. Figure 3-3 illustrates the trajectories of five point-to-point extractions across the orders of a high dispersion image, shown here as a contour

Table 3-3. Standard Deviations for Low S/N Case in Background Removal Analysis.
(background $\approx$ 2 x signal, halation included, order 2 and 4 constant, analytic function average = 39.4)

Function	Standard Deviation
Current Production	2.2
Constant Background	3.6
Heap's Method	1.5

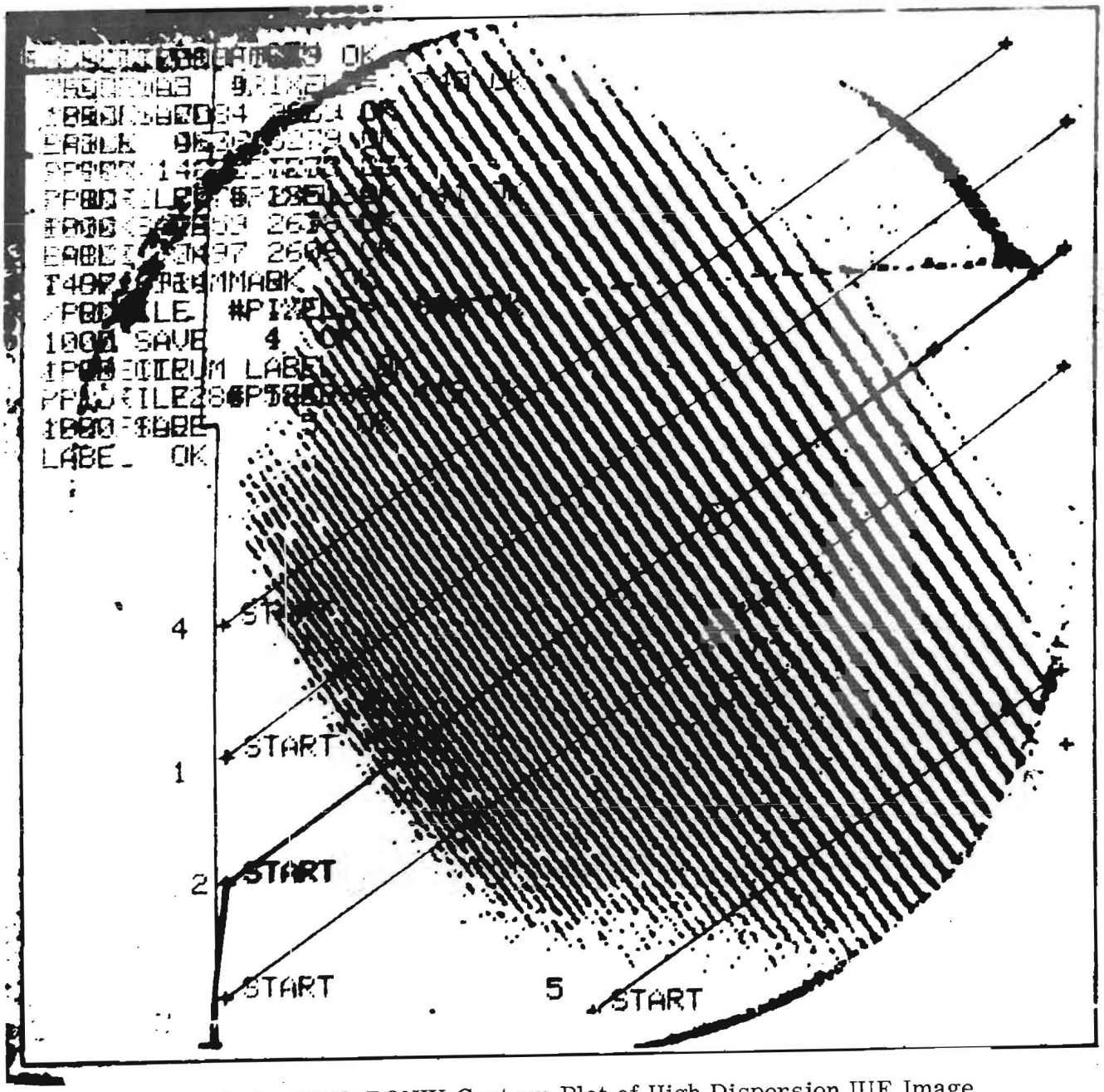

Figure 3-3. TEKTRONIX Contour Plot of High Dispersion IUE Image

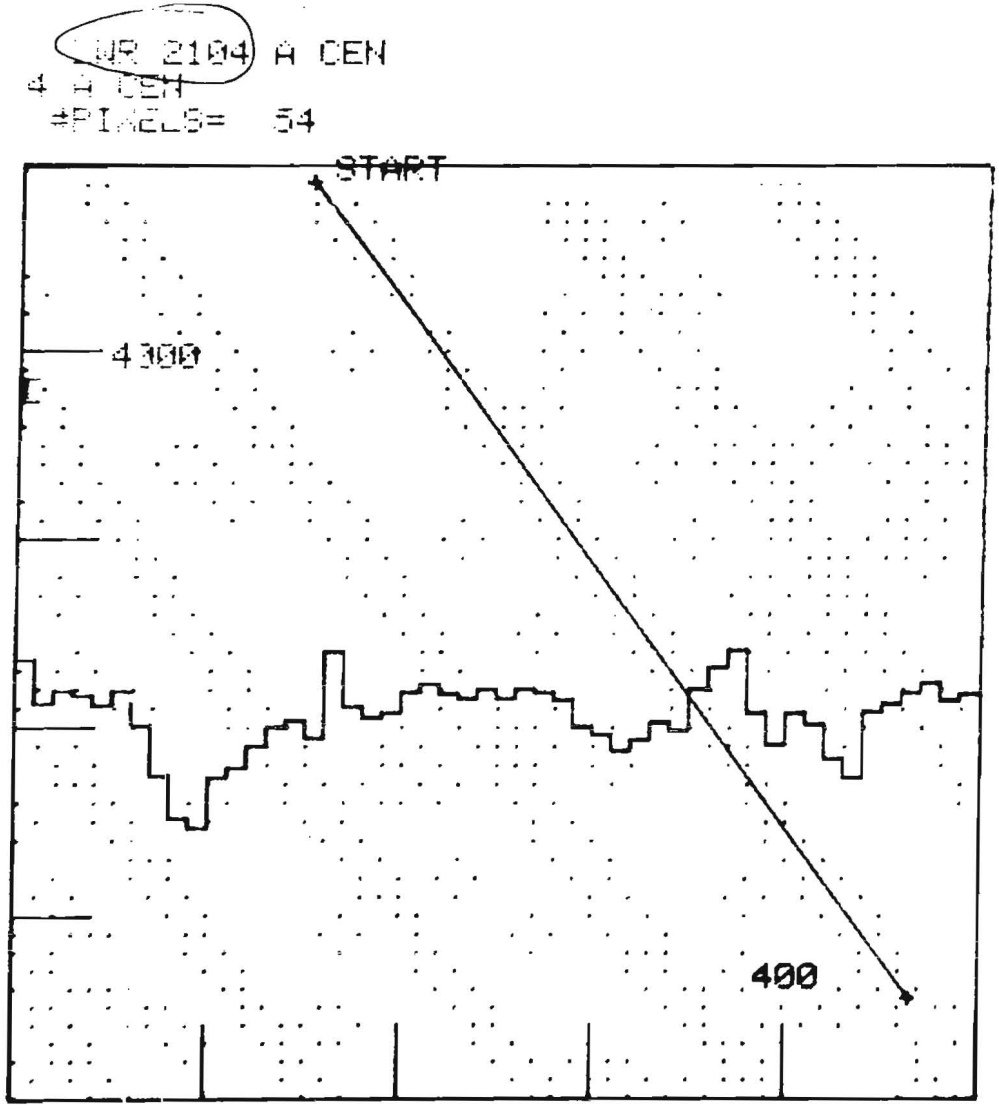

Figure 3-4. TEKTRONIX Contour Plot of a Small Portion of an IUE High Dispersion Image.

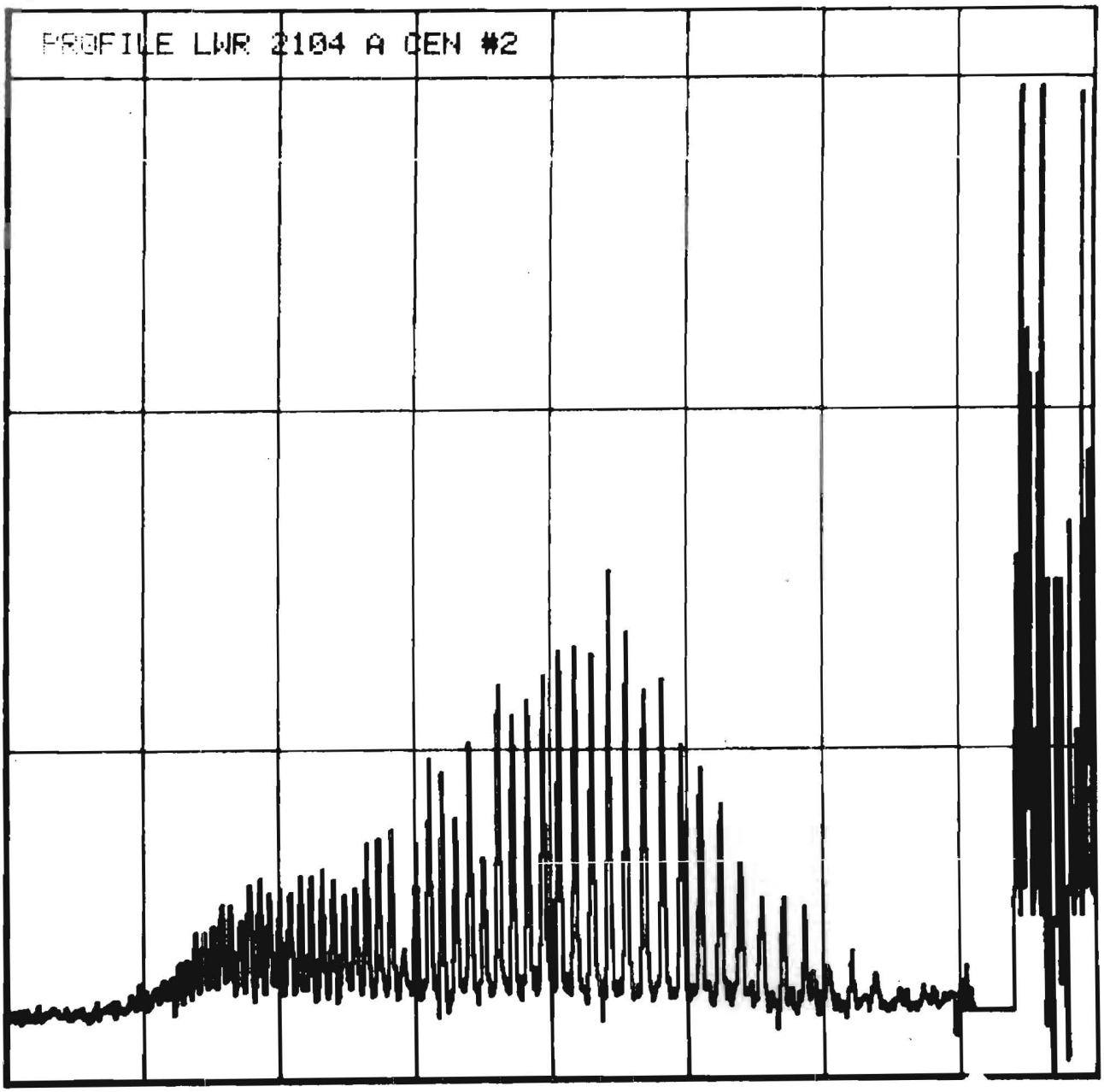

Figure 3-5. TEKTRONIX Plot of a Point-to-Point Profile

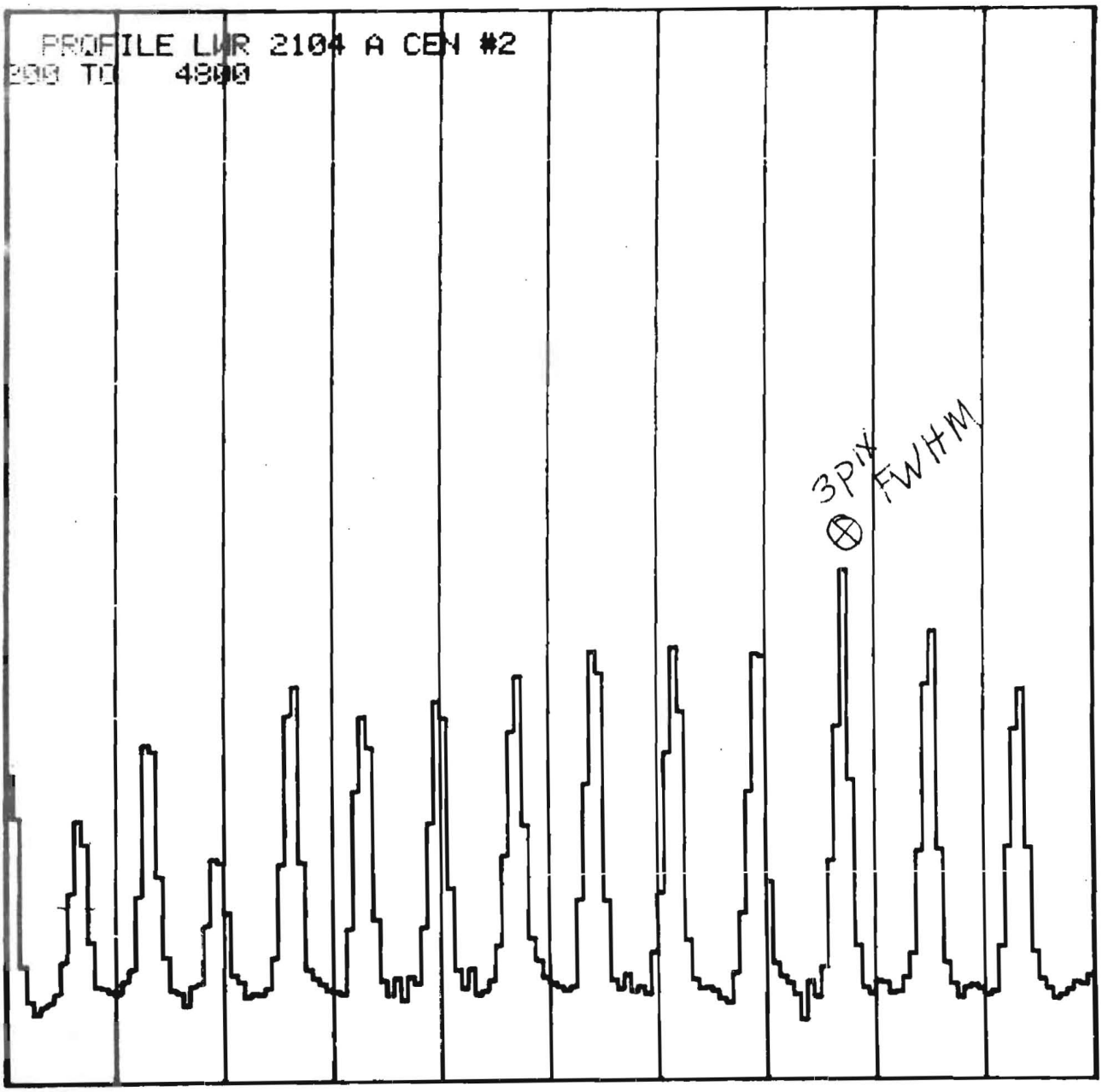

Figure 3-6. Expanded View of a TEKTRONIX Point-to-Point Plot

plot. Figure 3-4 is an expanded view of the TEKTRONIX contour plot, showing seven spectral orders. The straight line running along one of the orders is the path of a point-to-point data extraction. A plot of the extracted data is shown superimposed, running from left to right. Figure 3-5 shows the extracted data for the point-to-point plot labeled "2" in Figure 3-3, and Figure 3-6 shows an expanded view of a portion of this point-to-point extraction. From these data, the full width at half maximum (FWHM) of the most intense order is seen to be approximately 3 pixels in the direction perpendicular to the echelle dispersion.

SECTION 4 - LOW DISPERSION REDUCTION PROCEDURES

4.1 BACKGROUND REMOVAL ALGORITHMS

The current low dispersion reduction procedures are relatively more satisfactory than the high dispersion reduction procedures. Whereas the effects of scattered light, halation, radiation background and the like are present in low dispersion images just as in high dispersion images, their effects are much less deleterious because of the presence of only one order (or at most one spectrum for each entrance aperture). For this reason, the definition of procedures and algorithms to deal with these effects in low dispersion was deferred in favor of emphasizing procedures for the high dispersion reduction.

The techniques described in Section 3.1 which were used to evaluate the various high dispersion background-correction algorithms (which include scattered light and halation effects) may be generalized to the case of low dispersion. It may be anticipated that the best results would be obtained by modeling the tests after actual low dispersion order profiles which would in general be different from high dispersion profiles because the echelle grating is not used in low dispersion.

4.2 POSSIBLE MODIFICATIONS TO EXTLOW

A class of possible improvements to the low dispersion extraction routine EXTLOW has been defined by numerous people over the past year. This area of activity is one which was identified at the outset of the task as having low priority, being primarily the responsibility of other personnel. Consequently, no implementation of these modifications was undertaken, and we here outline them only for completeness.

- Modification of the slit function used to extract the gross spectrum so that the effective slit is precisely perpendicular to the dispersion.

Currently, the slit axis is at 45 degrees to the scan line direction, which is within about 6 degrees of being perpendicular to the dispersion direction. A redefinition of the slit axis will improve the resolution of the extracted spectrum of significantly extended objects. A further improvement might result from reducing the slit width as well.

- Extraction of spectra from both apertures simultaneously (i.e., in a single execution of EXTLOW).

- As an alternative to the above, a modification of EXTLOW so as to calculate the slit-integrated gross and background fluxes directly from the single-pixel spatially-resolved spectra (by co-addition of the appropriate spectra) rather than from separate integrated extractions. This would both save time and assure exact correspondence between sums of the spatially-resolved spectra and the slit-integrated spectra.

- Fitting of the extracted background spectrum with a polynomial to suppress high-frequency noise which is undesireable. Such noise may be due to discrete radiation events, reseaux, uncorrected fixed pattern noise (see Section 5), etc. The current method does a running-average smoothing of the extracted background which attenuates the high frequencies at the expense of introducing low-frequency biases.

SECTION 5 - GEOMETRIC AND PHOTOMETRIC CORRECTION PROCEDURES

5.1 SCOPE OF ANALYSIS

The analysis of the geometric and photometric correction procedures used by IUESIPS was identified by the three-agency IUE Coordination Meeting in September 1978 as a responsibility of the United Kingdom (SRC) IUE Project group. As a result, little direct analysis in this area was performed at GSFC under Task 670. In support of the UK efforts, a magnetic tape containing the found, smoothed, and extrapolated reseau positions for 28 TFLOOD calibration images used during the Guest Observer phase of IUE operations was created and sent to the UK group. Some limited analysis of these data was done at GSFC (Section 5.3), but the bulk of the task activities comprised a compilation of theoretical considerations in the area of fixed-pattern-noise analysis and reseau-stability analysis. These considerations form the basis for possible future improvements in the IUESIPS geometric and photometric correction methods, and they are addressed in Sections 5.2 through 5.5 which follow.

5.2 FIXED PATTERN NOISE

The term "fixed pattern noise" (FPN) is used herein to describe gray-level irregularities or inhomogeneities which are correlated from image to image, i.e., signal errors which to some degree repeat in all images. FPN is an observed phenomenon in raw IUE imagery at all scales from slowly-varying tube response to single-pixel structure. In fact the current geometric and photometric correction procedures compensate the former type of FPN (large-scale structures) very well, but the small-scale structure is rather inadequately compensated, presumably due largely to small inaccuracies in the registration of the photometric intensity transfer function (ITF) with the image being corrected (these inaccuracies arising from imperfections in the geometric correction). A quantitative assessment of the effects of FPN and the feasibility of

removing it requires a noise model and the analysis of actual imagery. The analysis outlined below has been suggested by R. A. White of CSC (Reference 3) as a means of providing such an assessment.

5.2.1 Noise Model

Three sources of error in the observed gray level of a given pixel are considered:

1. Additive FPN (independent of illumination level)
2. Multiplicative FPN (proportional to illumination level; e.g., photocathode irregularities)
3. Additive random noise (zero mean with standard deviation a function of illumination level)

That is,

$$\tilde{DN}_{ij}(I_{ij}) = DN(I_{ij}) + a_{ij} + b_{ij} I_{ij} + n_{ij}(I_{ij}) \qquad (5\text{-}1)$$

where $\tilde{DN}_{ij}(I_{ij})$ is the observed gray level at pixel i of line j, $DN(I_{ij})$ is the noise-free or "true" value of the gray level, I_{ij} is the illumination level at the pixel, a_{ij} and b_{ij} are coefficients independent of I_{ij}, and $n_{ij}(I_{ij})$ is the (additive) random noise. Equation 5-1 is written assuming that gray level is a linear function of illumination over an illumination range equal to the maximum error expected. For the following we will further assume that we are dealing with uniformly illuminated images, so that $I_{ij} \equiv I$ is spatially constant.

The coefficients a_{ij} and b_{ij} may be expressed as the sum of a constant (or slowly varying) component and a spatially-varying component:

$$a_{ij} = \bar{a} + c_{ij} \qquad (5\text{-}2)$$

where

$$\bar{a} \equiv \langle a_{ij} \rangle_R \tag{5-4}$$

$$\bar{b} \equiv \langle b_{ij} \rangle_R \tag{5-5}$$

and where $\langle a_{ij} \rangle_R$ signifies the mean value of a_{ij} computed within a region R about the point (i, j). This definition implies that c_{ij} and d_{ij} have zero mean within R, and since n_{ij} (I) is assumed to have zero mean,

$$\langle \tilde{DN}_{ij} (I) \rangle_R = \langle DN (I) \rangle_R + \bar{a} + \bar{b} I \tag{5-6}$$

or

$$\langle \tilde{DN}_{ij} (I) \rangle_R = DN (I) + \bar{a} + \bar{b} I \tag{5-7}$$

Thus from equations 5-1, 5-2, 5-3, and 5-7,

$$\tilde{DN}_{ij} (I) = \langle \tilde{DN}_{ij} (I) \rangle_R + c_{ij} + d_{ij} I + n_{ij} (I) \tag{5-8}$$

or

$$\tilde{DN}_{ij} (I) = \langle \tilde{DN}_{ij} (I) \rangle_R + E_{ij} (I) \tag{5-9}$$

where

$$E_{ij}(I) \equiv c_{ij} + d_{ij} I + n_{ij}(I) \qquad (5\text{-}10)$$

so that $E_{ij}(I)$ is the position-dependent FPN error term.

5.2.2 Analysis of Imagery

If a series of at least three uniformly illuminated images be considered, it is possible to estimate c_{ij}, d_{ij}, and n_{ij} for each point by a least-squares fit. If it is assumed initially that $n_{ij}(I) = 0$, then the apparent random noise $n_{ij}(I)$ for each level I may later be calculated from the best-fit c_{ij} and d_{ij} values as

$$n_{ij}(I) = E_{ij}(I) - c_{ij} - d_{ij} I \qquad (5\text{-}11)$$

Properties of the FPN such as spatial coherence or variability with time can be analyzed by examining these arrays. (For example, auto-correlations of the c_{ij} and d_{ij} arrays will determine the spatial coherence of the FPN, and cross-correlation of the $E_{ij}(I)$ arrays for different values of I will determine the short-time temporal stability of the noise pattern). In general, a three-phase analysis could be envisioned wherein a set of images recorded at one time would be analyzed in phase one to determine the noise parameters at that time, the spatial characteristics of the noise parameters would be analyzed in phase two, and these results would be compared to those for images acquired at other times to analyze time variability of the FPN in phase three.

The characterization of the FPN as described above can be used to address the question of how to minimize the effects of FPN. For example, the random noise component is the lower limit for residual noise to be expected in spectral images; in general, to this must be added the average effects of short-term

fluctuations in the FPN. Together, these data speak to the feasibility of correcting for the presence of FPN. Furthermore, the properties of the autocorrelation function of the FPN indicate the extent to which a smoothing or misregistration of the FPN would reduce the effectiveness of the removal process. Together with measurements of the "irreducible" misregistration of the FPN in the ITF with the FPN in spectral images due to geometric-correction error (Section 5.3), the measured noise parameters can be used to determine whether, for example, ITF files should be smoothed and whether the present geometric-correction pixel-resampling algorithm is appropriate (e.g., alternative interpolators are available).

5.3 RESEAU-POSITION ANALYSIS

5.3.1 Reseau-Mark Location Techniques

R. A. White (CSC) has suggested ways in which reseau-measurement techniques might be analyzed and perhaps improved in order to define the geometric distortion of IUE images more accurately (Reference 3). The existing cross-correlation technique used by the IUESIPS program FNDRES is well-suited only for flat-field (i.e., uniformly-illuminated) images, since it is known not to perform well in the presence of image intensity gradients such as those found in typical spectral images. Three possibly better alternative techniques for locating reseaux are suggested along with means of evaluating the performance of each: (1) normalized cross-correlation technique, (2) masked-summation technique, and (3) binary edge-registration technique.

5.3.1.1 Normalized Cross-correlation Technique.

This technique removes biases in the reseau locations caused by uniform gradients in image gray level by defining the variance V between the image and a

reseau-shaped template (at a given template alignment relative to the image) in a normalized fashion:

$$V = \sum_{R} [(DN_t - \bar{DN}_t)/\sigma_t - (DN_i - \bar{DN}_i)/\sigma_i]^2 \qquad (5\text{-}12)$$

where the following definitions hold: R is the region of the image overlaid by the template, DN_t is the gray-level value for a pixel in the template; DN_i is the gray-level value of the corresponding image pixel (for the given template alignment), and the mean values are

$$\bar{DN}_t \equiv < DN_t >_R \qquad (5\text{-}13)$$

$$\bar{DN}_i \equiv < DN_i >_R \qquad (5\text{-}14)$$

and the standard deviations are defined by:

$$\sigma_t^2 \equiv < (DN_t - \bar{DN}_t)^2 >_R \qquad (5\text{-}15)$$

$$\sigma_i^2 \equiv < (DN_i - \bar{DN}_i)^2 >_R \qquad (5\text{-}16)$$

The best template alignment, which defines a discrete-pixel reseau location, is that for which V is a minimum. This method is superior to unnormalized cross-correlations, but is not completely effective in eliminating biases due to strongly non-uniform intensity gradients near spectral lines.

5.3.1.2 Masked-Summation Technique

This approach makes use of the fact that reseau profiles are basically binary in character (reseaux are dark, backgrounds are light) so that locating a reseau

mark is equivalent to locating a reseau-shaped area of the image having a minimum average gray level value. Operationally, a reseau-shaped mask is defined within which the gray-level values are summed for various alignments of the mask relative to the image; the best alignment is that for which the masked gray-level sum is minimum. A mask slightly smaller than the reseau marks will provide the sharpest discrimination in practical applications where the observed reseaux have somewhat diffuse edges. Gray-level-value gradients may be compensated for by fitting polynomials to the gray level values in the immediate vicinity of the reseaux.

5.3.1.3 Binary Edge-Registration Technique

This approach utilizes edge correlations between templates and images rather than correlations of gray-level values. In this case, gray-level-value gradients in the images become considerably less troublesome. In a typical algorithm, one first surveys the image to calculate an "edginess" parameter for each pixel by a digital filter. A second pass through the image generates a corresponding binary image with gray level = 1 for pixels whose "edginess" parameter exceeds a chosen threshold value (most conveniently computed from the histogram of edginess parameters calculated by the initial survey), and gray level = 0 for all other pixels. The detection algorithm consists of finding the maximum coincidence of "edge pixels" with a pre-defined reseau mark edge template.

A disadvantage of this technique for IUE images is that reseau marks are small, so that each reseau has only a limited number of edge pixels; also, the degree of coincidence between template and image is rather sensitive to slightly undersized or oversized edge templates.

In all three techniques, the final fractional-pixel reseau locations are determined by two-dimensional interpolation.

Ideally, these various reseau-location techniques would be tested against a substantial number of actual images in order to evaluate their relative

effectiveness. To do this, it would be useful to calculate the statistics of the correlations for each method, addressing, for example, the peak correlation value, the mean correlation value over the search area used, the standard deviation, the ratio of peak-minus-mean value over the search area used, the standard deviation, and the distribution function of correlation values. These statistics can be used, along with the observed susceptability of each method to image-gradient biases, to choose a preferred reseau-locating algorithm.

5.3.2 Experimental Measurements

An analysis of the behavior of reseau positions as a function of time or more fundamental variables such as camera temperature is essential to an understanding of the optimal frequency of obtaining calibration images and the limiting accuracy of the inferred geometric distortions. As a first step in such an analysis program, the group of twenty-eight extrapolated reseau sets pertinent to the initial months of the IUE Guest Observer phase collected for delivery to the U.K. IUE project was examined. These reseau sets were all measured on the standard TFLOOD & WLC calibration images (tungsten flood lamp plus platinum wavelength calibration lamp exposures) by the standard cross-correlation algorithm (applications program FNDRES). These images include both low and high dispersion platinum exposures spanning the time period GMT days 79 through 263 of 1978. Table 5-1 lists the short wavelength prime (SWP) camera or long wavelength redundant (LWR) camera image numbers from which the reseaux were derived, the GMT day number, and the dispersion of the superimposed platinum spectrum, if any (H = high, L = low).

To examine changes in the reseau sets (for a given camera), the VICAR program RESOVDU (which displays in plotted form the displacements of all reseaux of a test set relative to the reseaux of a chosen fiducial set) was executed for each set with LWR 1834 and SWP 2025 (each from day 197) as fiducials. Examples of the extremes of the displacements observed are shown in Figures 5-1

Table 5-1. Parameters of Reseau Sets Used in Stability Analysis (1 of 2)

Camera	Image No.	GMT Day	WLC Dispersion
LWR	1190	79	H
	1199	81	L
	1222	83	L
	1421	120	H
	1485	133	L
	1548	143	H
	1590	152	L
	1643	160	L
	1713	173	L
	1754	181	L
	1834	197	L
	1972	216	L
	1989	217	H
	2027	221	L
	2146	234	H
	2411	263	L
SWP	1202	79	L
	1206	79	H
	1212	80	L
	1455	121	L
	1529	132	L

Table 5-1. Parameters of Reseau Sets Used in Stability Analysis (2 of 2)

Camera	Image No.	GMT Day	WLC Dispersion
SWP	1538	133	H
	1679	151	L
	1753	160	L
	1887	181	L
	2025	197	L
	2211	217	H
	2701	263	L

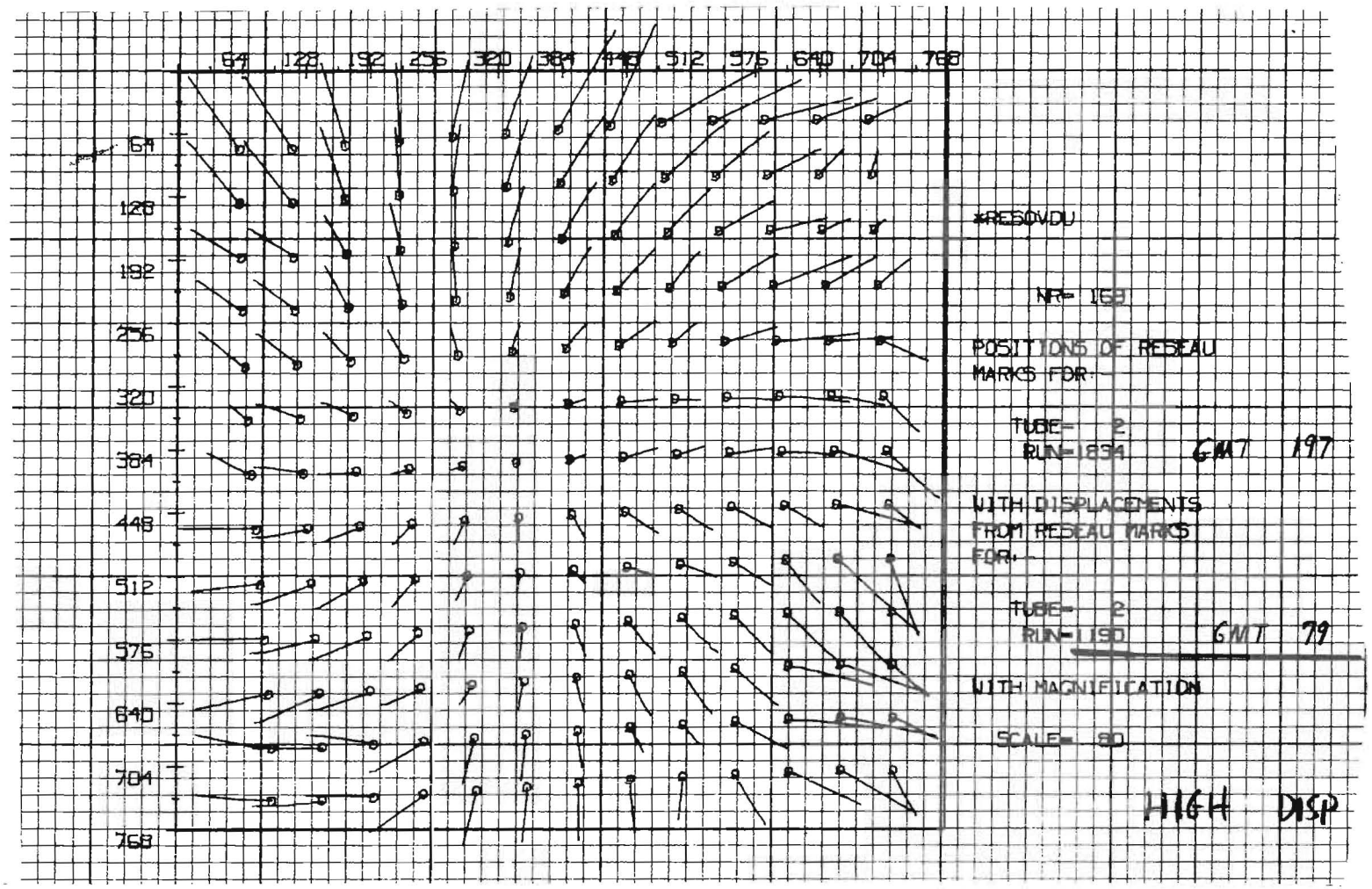

Figure 5-1. Reseau Movement for LWR 1190 Relative to LWR 1834

and 5-2 for LWR and SWP, respectively. The circles represent the locations of the 169 reseaux of the fiducial set on the pixel scale given around the border of the figure; the lines represent reseau displacements relative to the reseaux of the test set, but on a scale 80 times greater than the scale for the fiducial locations (so that a displacement of one pixel corresponds to a vector approximately one half of an inch long).

Figures 5-1 and 5-2 indicate that there are obvious changes between reseau sets for a given camera (i.e., reseaux do "move"). However, these figures are representative of the worst differences seen in the sample of reseau sets used; average displacements from the fiducial positions are almost always at the subpixel level. Figures 5-1 and 5-2 show displacements which are apparently understandable in terms of camera temperature or voltage changes, although so far no correlation with recorded values for such parameters has been attempted. In the LWR camera, the full set of plots indicates no secular or uniquely time-dependent trends in reseau motion, but for the SWP camera there is a marginal indication that the reseau sets measured tend to fall into two groups: those prior to day 151 which yield plots resembling Figure 5-2 and those later than day 151 which are extremely similar to the day 197 fiducial reseaux. The significance of this has yet to be assessed.

The reseau sets that have been examined thus far are not sufficiently numerous or spaced closely enough in time to allow a determination of the driving influences or limiting accuracies. More data are required, particularly to examine short-time-scale variability and to search for any significant correlations between measurable physical variables and reseau displacements. Such data can be used to quantify and hopefully minimize the irreducible geometric-correction error associated with measuring reseau locations on a finite (and thus far, in practice, a rather sparse) set of calibration images.

One may reasonably contemplate extending the experimental analysis several steps further to attempt assessments of intensity-dependent "beam pulling"

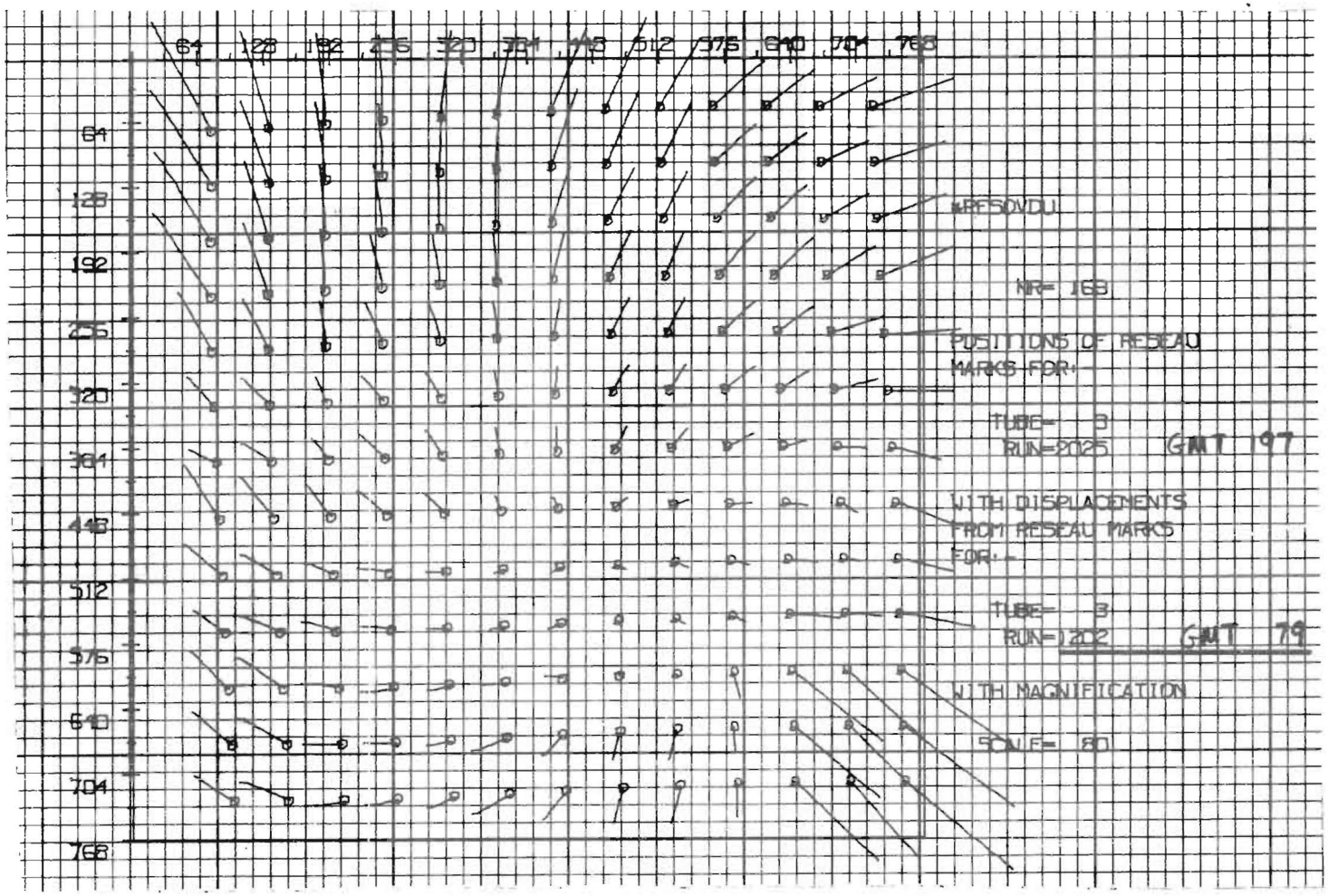

Figure 5-2. Reseau Movement for SWP 1202 Relative to SWP 2025

effects which modify the geometric distortion, along lines pointed out by R. A. White (Reference 3). For example, the measurement of reseaux on a set of ITF-component flat-field images can be used to deduce the nature of the dependence of observed reseau locations on average intensity level, and the comparison of those positions with positions measured in spectral images (i.e., images with intensity gradients near reseaux) can be used to quantify the dependence of observed reseau locations on local intensity structure. It would probably be useful to fit the actual spatial intensity distributions of the images with quadratic surfaces over a number of different scales and plot the differential reseau displacements against the line and sample direction gradient and curvature terms in order to attempt to correlate reseau locations with parameterizable image structure. To the extent that it may be possible to reasonably parameterize beam pulling effects in this way, it would be possible to improve the accuracy of the geometric distortion correction.

5.4 ALTERNATIVES TO CURRENT GEOMETRIC CORRECTION METHODS

It is an observed fact that the bilinear-interpolation pixel-resampling algorithm used in the standard program GEOM causes a loss of resolution of fine-scale structures visible in raw images. Furthermore, the fact that the resampling operation is performed in nonlinear DN space (DN is proportional to integrated charge, not incident light intensity) leads to a loss of photometric integrity, particularly in regions of sharp gray-level-value gradients. Various aspects of these problems could conceivably be mitigated by modifying the current geometric correction procedures, as discussed in the following sections. It appears possible to address successfully the problem of loss of spectral resolution in several ways, although the loss of resolution of fine-scale fixed pattern noise (FPN) is intrinsically more difficult to eliminate.

5.4.1 Optimal Resampling in GEOM

The geometric correction program GEOM performs a pixel-resampling operation in the raw image in order to define the corrected image. Presently, this resampling is done by means of a bilinear interpolation which is described in detail elsewhere (Reference 4). This type of resampling is known to induce a "smearing" because features which are sharply defined in the raw image are in general redistributed over more pixels in the corrected image. (It is this redistribution of pixel values which induces the photometric error (mentioned above) since the input gray levels are not linearly related to intensity). This redistribution of pixel values may be altered by changing the interpolating algorithm - for example, replacing the bilinear interpolation with a nearest-neighbor interpolation.

Another interpolation algorithm which treats the resampling in an "optimal" way is that discussed by Lorre (Reference 5). In this case, one makes use of the fact that insofar as the spectral format is concerned, the Sampling Theorem is satisfied: i.e., the spectral resolution in the raw image is of the order of 2 or 3 pixels, so that the signal (spectrum) is sufficiently bandlimited that the sampling frequency exceeds the Nyquist rate, with the result that the original (continuous) signal may be reconstructed at any point by properly interpolating between the observed (sampled) signal. The proper optimal interpolator according to Lorre is described by:

$$I(x, y) = \sum_{i=x-\Delta}^{x+\Delta} \sum_{j=y-\Delta}^{y+\Delta} I(i, j) \frac{\sin[\pi(x-i)]}{\pi(x-i)} \frac{\sin[\pi(y-j)]}{\pi(y-j)} \qquad (5\text{-}17)$$

where $I(x, y)$ is the interpolated intensity at fractional pixel location (x, y), $I(i, j)$ is the sampled intensity at the rounded integer pixel location (i, j), and Δ is the half-width of the interpolation filter. (Equation 5-17 is the

spatial-domain equivalent to a Fourier-domain square filter which truncates all but the central lobe of the Fourier transform of the sampled image, thus leaving the Fourier transform of the continuous image).

Lorre states that ground-based SIT vidicon spectra may be geometrically corrected without a degradation in resolution by utilizing Equation 5-17. In the case of IUE spectral images, it seems clear that similar preservation of spectral resolution can be achieved by this interpolator, although the finest structure of the FPN will still not be preserved, it being effectively "undersampled" in IUE images (FPN is a camera-induced phenomenon essentially unaffected by the optical resolution of the scientific instrument, and it is known to contain fine-scale structure).

Within the framework of the geometric-correction step now part of the standard IUESIPS processing, the interpolation scheme in Equation 5-17 would be an improvement to the current bilinear technique and would presumably be time-effective as long as the sinc function weights could be stored for lookup (e.g., in a grid of 0.05-pixel spacing over the filter half-width Δ). One should note that the FPN "resolution" will not in general be much improved, however, and that the photometric error associated with interpolation in a non-linear space is not eliminated.

5.4.2 Elimination of Explicit GEOM

An alternative approach which has the advantage of addressing the loss of photometric accuracy mentioned above as well as the loss of spectral resolution is to eliminate the geometric correction of target images. Under this method, one constructs ITF files from raw images which are registered with respect to the target images in piecewise fashion with reference to local reseau measurements. Some registration will always be necessary because it has been established that reseau positions do "move", if only at the subpixel level (Section 5.3.2).

Note that the raw image is imagined to be held fixed in this approach (i.e., it is not resampled) so that full spectral and FPN resolution in the target image is maintained and photometric accuracy is preserved because there is no resampling in non-linear space. However, it appears to be extremely difficult, if not impossible, to maintain the finest-scale FPN structure through the creation and registration of the ITF, since some resampling or interpolation within the ITF is necessitated by the inevitable reseau motion. Displacements of reseaux among the images comprising the "raw image" ITF may necessitate a "differential GEOM" of some of the component images to refer them to the reseau positions of some fiducial exposure (e.g. the 60 percent UVITF exposure). Even if this is not required, an ITF interpolation or effective resampling will generally be necessary to achieve registration with the raw target image being processed. The fact that either type of resampling is needed will lead to loss of "resolution" of the FPN structure, regardless of the fact the differential displacements may be only fractions of a pixel. The seriousness of this effect depends upon the spatial characteristics of the FPN (e.g., as manifested by the auto-correlation function of the FPN - see Section 5.2).

The apparent trade offs between the various methods now in use or discussed here may thus be summarized:

1. The current GEOM procedure induces loss of spectral resolution in the target image and loss of FPN fine structure in the target image and the ITF. It also leads to photometric error.

2. The optimal-interpolator GEOM eliminates loss of spectral resolution but still suffers FPN smearing in both the target image and the ITF. Photometric error is still present.

3. The elimination of GEOM also eliminates the loss of spectral and FPN resolution in the target image, and preserves photometric accuracy. FPN fine structure is inevitably smeared in the ITF and/or in its registration with the target image.

5.5 IMPROVED TREATMENT OF SATURATED PIXELS

The current photometric correction program FICOR5 treats saturated input pixels in a rudimentary way: all input pixels with DN above a parameter-specified level SATLIM (standard value = 254) are defined to be "saturated" and given an output value of 32,512 + DN (i.e., 32,767) in the photometrically-corrected image. Such pixels are then flagged by the spectral extraction programs when they contribute to extracted flux points. This approach is less than ideal for two reasons: 1) a constant SATLIM of 254 makes it impossible to flag out-of-range input pixels (i.e., those with DN greater than maximum corresponding DN in the ITF) in regions of the tube where the sensitivity is low, and 2) the artificially-large output flux value is confusing to Guest Observers and no doubt does unnecessary injustice to the photometry in cases where, e.g., only one or two contributing pixels (out of perhaps 9 or more) for an extracted flux measurement are "saturated".

An alternative approach is suggested here, requiring small modifications in both FICOR5 and the extraction routines. On the premise that all out-of-range (and thus, literally, uncalibrated) pixels should be flagged, change the DN test in FICOR5 to look for input DN equal to or exceeding the corresponding maximum DN in the ITF. If the test is met, assign an output pixel value equal to the maximum ITF intensity level (no artificial offsets). Assign a new parameter to the extraction routines equal to the maximum ITF intensity level, and if any pixel contributing to an extracted flux equals this value, flag the flux point as out-of-range by the epsilon field as is currently done for "saturated" pixels.

This technique is more encompassing than the one in current use, as it identifies all uncalibrated pixels, and it leads to a behavior of out-of-range fluxes in the extracted spectrum which is intuitively reasonable; i.e., "saturated" flux distributions would be flat topped.

APPENDIX A - DERIVATION OF WAVELENGTH TRANSFORMATION FORMULA

Problem: Derive an equation which will allow the transformation of wave lengths, λ_0, determined using one set of dispersion constants (say, the "1st set") to the wavelengths, λ, which we would have found if we had used some other set (say, the "2nd set").

Given: a'_1, a'_2, b'_1, b'_2 $\equiv$ 1st set of dispersion constants

a_1, a_2, b_1, b_2 $\equiv$ 2nd set of constants

λ_0 $\equiv$ The wavelength determined using the "1st set"

λ $\equiv$ The wavelength determined using the "2nd set"

$$l_0 = b'_1 + b'_2 \lambda_0 \equiv \text{line number (1st set)} \qquad \text{(A-1)}$$

$$s_0 = a'_1 + a'_2 \lambda_0 \equiv \text{sample number (1st set)} \qquad \text{(A-2)}$$

$$l = b_1 + b_2 \lambda \equiv \text{line number (2nd set)} \qquad \text{(A-3)}$$

$$s = a_1 + a_2 \lambda \equiv \text{sample number (2nd set)} \qquad \text{(A-4)}$$

We want to assign to an already extracted data point located at (s_0, l_0) on the line defined by equations (A-1) and (A-2) a new wavelength, λ, which will in turn define a point (s, l) through equations (A-3) and A(4). Moreover we want to choose λ such that (s, l) is as close to (s_0, l_0) as possible; see Figure A-1 This figure depicts a small area of the s-l plane (the image plane) and shows two dispersion lines. The symbols used are defined as follows:

ds $\equiv$ The minimum distance from a point on one dispersion line to the other dispersion line

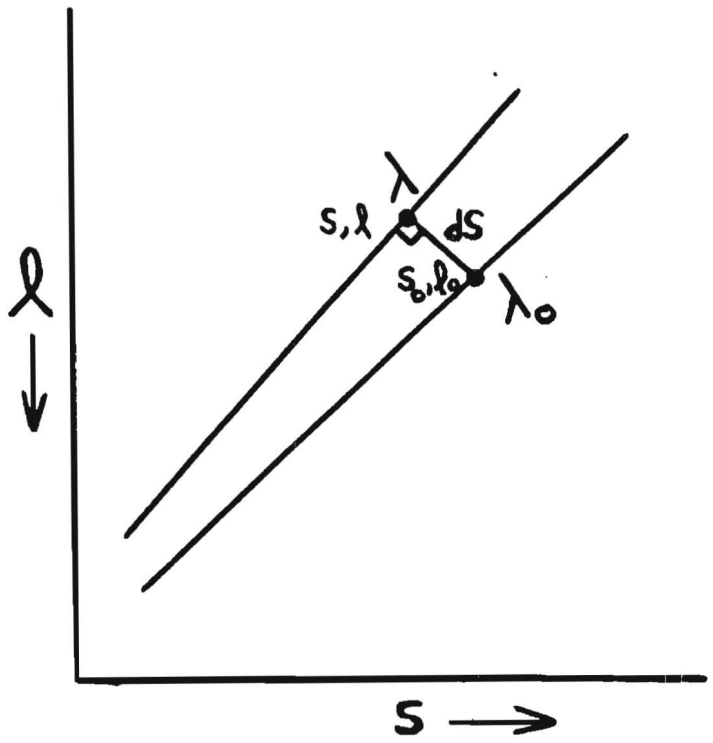

Figure A-1. Geometric Relationship of Two Dispersion Lines

s = The sample number

l = The line number

λ = The wavelength

The subscript zero refers to the 1st set of quantities. Therefore, we want to minimize the quantity

$$ds = \sqrt{(s - s_0)2 + (l - l_0)^2} \qquad \text{(A-5)}$$

From the above equations for l and s in terms of the a's and b's we have:

$$l = b_1 + b_2 \left(\frac{s - a_1}{a_2}\right)$$

$$l = \left(b_1 - \frac{a_1}{a_2} b_2\right) + \left(\frac{b_2}{a_2}\right) s \qquad \text{(A-6)}$$

and thus

$$ds = \left[\left(s - s_0\right)^2 + \left\{\left(b_1 - \frac{a_1}{a_2} b_2\right) + \left(\frac{b_2}{a_2}\right) s - l_0\right\}^2\right]^{1/2} \qquad \text{(A-7)}$$

We take the first derivative of ds with respect to s and set this equal to zero:

$$D_s\,(dS) = \frac{1}{2}\left(dS^2\right)^{-1/2}\left\{2\left(s - s_0\right) + \frac{2a_2}{b_2}\left[\left(b_1 - \frac{a_1}{a_2} b_2\right) + \left(\frac{b_2}{a_2}\right) s - l_0\right]\right\}$$

$$= 0 \qquad \text{(A-8)}$$

Solving equation (A-8) for s we obtain

$$s = \frac{s_0 + \frac{b_2}{a_2}\left[l_0 - \left(b_1 - \frac{a_1}{a_2} b_2\right)\right]}{\left[1 + \left(\frac{b_2}{a_2}\right)^2\right]} \qquad \text{(A-9)}$$

and using this s in Equation (A-4) we have

$$\lambda = \frac{s - a_1}{a_2} \qquad \text{(A-10)}$$

which gives us the new value of λ in terms of the 1st set of dispersion constants and s_0 and l_0.

Now since s_0 and l_0 are related to the 2nd set of dispersion constants and λ_0 through equations (A-1) and (A-2) we have

$$\lambda = \frac{\left(a'_1 + a'_2 \lambda_0\right) + \frac{b_2}{a_2}\left[\left(b'_1 + b'_2 \lambda_0\right) - \left(b_1 - \frac{a_1}{a_2} b_2\right)\right] - \frac{a_1}{a_2}}{\left[1 + \left(\frac{b_2}{a_2}\right)^2\right] a_2}$$

which can be simplified to

$$\lambda = \frac{a_2 (a'_1 - a_1) + b_2 (b'_1 - b_1)}{a_2^2 + b_2^2} + \lambda_0 \left(\frac{a_2 a'_2 + b_2 b'_2}{a_2^2 + b_2^2}\right) \qquad \text{(A-11)}$$

$$\lambda = d + m\lambda_0 \tag{A-12}$$

where

$$d = \frac{a_2 (a'_1 - a_1) + b_2 (b'_1 - b_1)}{a_2^2 + b_2^2} \tag{A-13}$$

and

$$m = \left(\frac{a_2 a'_2 + b_2 b'_2}{a_2^2 + b_2^2} \right) \tag{A-14}$$

This equation (A-12) can be used to transfer between any two sets of dispersion constants. It should be noted however that if the two dispersion lines diverge by more than a few pixels the resulting extracted spectrum may be substantially in error. When the equation is being used to correct for errors caused by the bad line library the results obtained are quite accurate since the two dispersion lines never differ in direction by more than 0.003 Radian.

APPENDIX B - DSPCON SOURCE CODE LISTING (IUFSIPS)

```
      SUBROUTINE DSPCON
C PROGRAM DSPCON (=DISPERSION CONSTANTS) SAMPLES PORTIONS OF THE          00000010
C SPECTRAL IMAGE AND REFINES THE VALUES OF THE OFFSET DISPERSION          00000020
C CONSTANTS.
C
C SIGMA-9 VERSION 2
C
C CALLS SUBROUTINE OFFSET
C
C
C                                                                         00000040
C     MALLAMA     CSC     7/5/78                                        00000050
C                                                                         00000060
C     LAST MODIFIED: 9/18/78     BY: MALLAMA
C
      LOGICAL*4      CW,WVSEC                                           00000070
      INTEGER*4      PIXMAT(15,15),PIXOUT(768),HEADER(18),IBUF(256),    00000080
     &               IX(12),IY(12),BUF(360),PIXTRL(41,3)
      REAL*4         CAMCON(4)/'LWP ','LWR ','SWP ','SWR '/,            00000100
     &               DSPCOM(2)/'HIGH','LOW '/,                          00000110
     &               APCOM(2) /'LRG ','SML '/,PAR(49),X(12),Y(12),DELX(12) 0000012
     &              ,TRAIL(2)/'NOTR','TR  '/,AORDER/'ORDE'/,
     &               ATRUE/'TRUE'/,AFALSE/'FALS'/,AWAVE/'WAVE'/,
     &               DELY(12),SHIFT(12),CORSHF(12),CORR(12),AINC/39.0/  00000130
     &              ,WAVELN(12)
C OFFSETS FOR LARGE APERTURES (LWP(LINE,SAMPLE),LWR,SWP,SWR)              00000140
C     REAL*4         ALRGAP(8)/0.,0.,19.5,-17.5,-20.0,-17.0,0.,0./      00000150
      REAL*4         ALRGAP(8)/8 * 0./
C ORDERS SAMPLED (LW,SW)                                                  00000160
      INTEGER*4      IORDR(24)/86,86,90,90,94,94,98,98,78,78,82,82,
     &                        82,82,86,86,90,90,94,94,74,74,78,78/
C SAMPLED WAVELENGTHS (HIGH DISP(LW,SW),LOW DISP)                         00000180
      REAL*4                                                            00000190
     & WAVHI(24)/2700.,2690.,2580.,2570.,2470.,2460.,
     &           2370.,2360.,2970.,2960.,2830.,2820.,
     &           1690.,1680.,1610.,1600.,1540.,1530.,
     &           1475.,1465.,1870.,1860.,1775.,1765./,
     & WAVLOW(24)/2500.,2600.,2700.,2800.,2900.,3000.,
     &           3100.,3200.,2100.,2200.,2300.,2400.,
     &           1500.,1550.,1600.,1650.,1700.,1750.,
     &           1800.,1850.,1300.,1350.,1400.,1450./
C NOMINAL DISPERSION CONSTANTS (HIGH DISP(LWP(SMP,LIN),LWR,SWP,SWR),LOW)00000240
      REAL*8         HICON(1)
C     & 14 * 0.D0,                                                      00000260
C     &-.650891504682632D 04, .161968112878176D 00,-.585191449433930D-06,00000265
C     & .000000000000000D 00, .211323526328300D 00, .000000000000000D 00,00000270
C     & .391540106841424D-06,                                           00000275
C     & .147064984599250D 05,-.271312018758039D 00, .894269779708860D-06,00000280
C     & .000000000000000D 00, .227440198795110D 00,-.304247744684358D-06,00000285
C     & .000000000000000D 00,                                           00000290
C     & .197071360857575D 04,-.191925470027741D 00, .134413263418483D-05,00000295
C     & .000000000000000D 00,-.467709326385780D 00, .000000000000000D 00,00000300
C     & .000000000000000D 00,                                           00000305
C     &-.302745924357619D 04,-.177630776173760D 00, .143575070513906D-05,00000310
C     & .000000000000000D 00, .374678486729839D 00, .000000000000000D 00,00000315
C     & .000000000000000D 00,                                           00000320
```

```
C     & 14 * 0.D0/                                                          00000325
      REAL*8          LOWCON(1)
C     & 4 * 0.D0,                                                           00000335
C     &-.29887948960335550 03, .30239743255812600 00,                       00000340
C     &-.26683084422090800 03, .22598205300902300 00,                       00000345
C     & .98997454383199800 03,-.47309338366730600 00,                       00000350
C     &-.27182548549915500 03, .38091663548361100 00,                       00000355
C     & 4 * 0.D0/                                                           00000360
      REAL*8 DCCON(42),DCSHFT(3)
      EQUIVALENCE (PAR(11),CAMERA),(PAR(13),DISP),(PAR(15),AP)
     &           ,(PAR(19),IRECDC),(PAR(20),SUBWAV),(BUF(7),DCCON(1))
     &           ,(IWVSEC,IWVSEC),(PAR(17),TR),(BUF(91),DCSHFT(1))
C INPUT CAMERA, DISPERSION, APERTURE, AND DISPERSION RECORD NUMBER.
      CALL PARAM(IND,PAR,49)
C     WRITE(108,999)(PAR(I),I=11,37),IND
  999 FORMAT(' INPUT PARAMETERS',4(1X,2A4),1X,I2,1X,2A4,
     & /,1X,2A4,6(1X,I3),1X,2A4,6(1X,F6.1),1X,I10)
C CHECK INPUT                                                               00000460
      ICAM = 0                                                              00000470
      IDSP = 0                                                              00000480
      IAP  = 0                                                              00000490
      ITRAIL=0
      DO 100 I = 1,4                                                        00000500
  100 IF(CAMERA.EQ.CAMCOM(I)) ICAM = I                                      00000510
      DO 200 J = 1,2                                                        00000520
      IF(DISP.EQ.DSPCOM(J))    IDSP = J                                     00000530
      IF(AP.EQ.APCOM(J))       IAP  = J                                     00000550
  200 IF(TR.EQ.TRAIL(J))       ITRAIL=J
      WRITE(108,902)CAMCOM(ICAM),DSPCOM(IDSP),APCOM(IAP),TRAIL(ITRAIL),
     & IRECDC,SUBWAV
  902 FORMAT(///,' CAMERA:',1X,A4,/,' DISP:',3X,A4,/,' AP:',5X,A4,      00000570
     &       /,' TRAIL:',2X,A4,/,' DREC:',3X,I4,/,' SUBWAV:',1X,A4)
C PROCESS SUBSTITUTE WAVELENGTHS AND ORDERS
      IF(SUBWAV.EQ.AFALSE)GOTO300
C DETERMINE IF WAVELENGTHS OR ORDERS COME FIRST
      WVSEC=PAR(22).EQ.AORDER
      WRITE(108,109)WVSEC,IWVSEC
  109 FORMAT(' WVSEC',L4,1X,' IWVSEC',I4)
C HIGH OR LOW DISPERSION
      IF(IDSP.EQ.2)GOTO120
      DO 110 I=1,12
      WAVHI(I+(ICAM-1)/2*12)=PAR(23+I+IWVSEC*14)
  110 IORDR(I+(ICAM-1)/2*12)=PAR(37+I+IWVSEC*14) + 0.1
      WRITE(108,119)(WAVHI(I+(ICAM-1)/2*12),I=1,12),
     & (IORDR(I+(ICAM-1)/2*12),I=1,12)
  119 FORMAT(//,' SUBSTITUTE WAVELENGTHS AND ORDERS',/,12(2X,F6.1),/,
     & 12(2X,I3))
      GO TO 300
  120 CONTINUE
      DO 130 I=1,12
  130 WAVLOW(I+(ICAM-1)/2*12)=PAR(23+I+IWVSEC*14)
      WRITE(108,139)(WAVLOW(I+(ICAM-1)/2*12),I=1,12)
  139 FORMAT(//,' SUBSTITUTE WAVELENGTHS',12(2X,F6.1))
  300 CONTINUE
      IF(ICAM.EQ.0.OR.IDSP.EQ.0.OR.IAP.EQ.0.OR.ITRAIL.EQ.0) GO TO 9999
C READ IN DISPERSION CONSTANTS
```

```
      CALL OPEN(IREC,3,0,1,0,0)
      CALL READ(IND,3,0,0,0,360,IBUF,0)
      CALL RCDBIN(IBUF(9),NL)
      CALL CLOSE(IND,3,0)
      CALL OPEN(IND,3,0,0,0,0)
C READ LAST RECORD OR SPECIFIC RECORD
      IRECNO = IREC + NL
      IF(IRECDC.NE.0) IRECNO = IREC + IRECDC
      CALL READ(IND,3,IRECNO,0,0,360,BUF,0)
      CALL CLOSE(IND,3,0)
      IF (IRECDC.NE.0) WRITE(108,101)IRECDC,DCCON
      IF (IRECDC.EQ.0) WRITE(108,101)NL,DCCON
  101 FORMAT(///,' DC RECORD',I4
     &       //,'  INPUT DCS:',11(/,4(1X,D22.15)))
C CALCULATE X,Y COORDINATES TO BE SAMPLED                          00000590
      IF(IDSP.EQ.1) GO TO 450                                      00000600
C LOW DISPERSION                                                   00000610
      DO 400 I = 1,12                                              00000122
      WAVELN(I) = WAVLOW((ICAM-1)/2*12+I)
      X(I) = DCCON(22) + DCCON(23)*WAVELN(I)
  400 Y(I) = DCCON( 1) + DCCON( 2)*WAVELN(I)
      GO TO 600                                                    00000125
C HIGH DISPERSION                                                  00000121
  450 DO 500 I = 1,12                                              00000127
      WAVE = WAVHI((ICAM-1)/2*12+I)                                00000128
      WAVELN(I) = WAVE
      IORDER = IORDR((ICAM-1)/2*12+I)                              00000129
      X(I) = DCCON(22) + DCCON(23)*IORDER*WAVE
     & + DCCON(24)*(IORDER*WAVE)**2 + DCCON(25)*IORDER
     & + DCCON(26)*WAVE + DCCON(27)*IORDER**2*WAVE
     & + DCCON(28)*IORDER*WAVE**2
  500 Y(I) = DCCON( 1) + DCCON( 2)*IORDER*WAVE
     & + DCCON( 3)*(IORDER*WAVE)**2 + DCCON( 4)*IORDER
     & + DCCON( 5)*WAVE + DCCON( 6)*IORDER**2*WAVE
     & + DCCON( 7)*IORDER*WAVE**2
  600 CONTINUE                                                     00000780
C TRANSFORM TO X,Y'S FOR LARGE APERTURE IF NEEDED                  00000790
      IF(IAP.EQ.2)GOTO800                                          00000800
      DO 700 I=1,12                                                00000810
      X(I) = X(I) + ALRGAP(2*ICAM)                                 00000820
  700 Y(I) = Y(I) + ALRGAP(2*ICAM-1)
  800 CONTINUE                                                     00000840
C COMPUTE PIXEL VALUES                                             00000850
      DO 900 I = 1,12                                              00000812
      IX(I) = X(I)                                                 00000870
      DELX(I) = X(I) - IX(I)                                       00000880
      IY(I) = Y(I)                                                 00000890
  900 DELY(I) = Y(I) - IY(I)                                       00000900
      WRITE(108,901)X,IX,DELX,Y,IY,DELY
  901 FORMAT(///,'  XS',3X,12F9.3,
     &          /,' IXS',3X,12I9,
     &          /,' DXS',3X,12F9.3,
     &          /,'  YS',3X,12F9.3,
     &          /,' IYS',3X,12I9,
     &          /,' DYS',3X,12F9.3)
C INPUT SAMPLE PIXELS AND COMPUTE OFFSETS                          00000950
```

```
C TILT CW OR CCW? (NO DATA AVAILABLE FOR LNP OR SWR)                    00000960
      IF(ICAM.EQ.2) CW = .FALSE.                                        00000970
      IF(ICAM.EQ.3) CW = .TRUE.                                         00000980
      IF(IDSP.EQ.1) CW = .NOT.CW                                        00000990
      TOTSHF = 0.                                                       00001000
      TOTCOR = 0.                                                       00001010
      ISAMP = 12
      DO 1200 I = 1,12                                                  00001020
      CALL OPEN(IND,2,0,0,ABUF,6)
      IF(ITRAIL.EQ.1)ITX = IX(I) - 8                                    00001040
      IF(ITRAIL.EQ.1)ITY = IY(I) - 8                                    00001050
      IF(ITRAIL.EQ.2)ITX = IX(I) - 21
      IF(ITRAIL.EQ.2)ITY = IY(I) - 2
      WRITE(108,905)IX(I),IY(I),WAVELN(I)
  905 FORMAT(/////,' ITX',I5,6X,'IIY',I5,6X,' WAVELENGTH',F9.2)
      IRECNO = IIY + IND
      IF(ITRAIL.EQ.2) GO TO 1125
      DO 1100 J = 1,15                                                  00001120
      IRECNO = IRECNO + 1
 1100 CALL READ(IND,2,IRECNO,1,ITX,15,PIXMAT(1,J),AUTOCK)
      GO TO 1175
 1125 DO  1150 J=1,3
      IRECNO = IRECNO + 1
 1150 CALL READ(IND,2,IRECNO,1,ITX,41,PIXTRL(1,J),AUTOCK)
 1175 CALL CLOSE(IND,2,0)
      IF(ITRAIL.EQ.1)CALL OFFSET(PIXMAT,CW,SHIFT(I),CORR(I))
      IF(ITRAIL.EQ.2)CALL OFFTRL(PIXTRL,CW,SHIFT(I),CORR(I))
      IF(SHIFT(I).GT.98.)ISAMP=ISAMP-1
      IF(SHIFT(I).GT.98..AND.I.EQ.0)SHIFT(I) = 0.
      IF(SHIFT(I).GT.98..AND.I.NE.0)SHIFT(I)=SHIFT(I-1)
      CORSHF(I) = SHIFT(I)
      IF(ITRAIL.EQ.1)CORSHF(I) = SHIFT(I) - DELY(I)
      TOTSHF = TOTSHF + CORSHF(I)                                       00001160
 1200 TOTCOR = TOTCOR + CORR(I)                                         00001170
C COMPUTE AVERAGE SHIFT                                                 00001230
      AVESHF = TOTSHF / 12.                                             00001240
      WRITE(108,903)CORSHF,SHIFT,CORR,TOTCOR,TOTSHF,AVESHF             00001180
  903 FORMAT(///,' CORSHF',12(1X,F9.3),                                 00001190
     &          /,'  SHIFT',12(1X,F9.3),                               00001200
     &          /,'  CORR',12(1X,1PE9.2),
     &          /,' TOTCOR',5X,0PE9.2,6X,'TOTSHF',6X,F9.3,
     &         //,' AVESHF',5X,F9.3)
C COMPUTE X,Y COMPONENTS                                                00001250
      AVEY = AVESHF * COS(AINC/57.3) * 0.707                            00001260
      AVEX = AVESHF * SIN(AINC/57.3) * 0.707                            00001270
      IF(.NOT.CW) AVEX = -AVEX                                          00001280
C COMPUTE R.M.S. DEVIATION                                              00001290
      SUMDEV = 0.                                                       00001300
      DO 1300 I = 1,12                                                  00001310
 1300 SUMDEV = SUMDEV + (CORSHF(I)-AVESHF)**2
      AVSUDV = SUMDEV / 12.
      RMS = SQRT(AVSUDV)                                                00001330
C PUT RMS TEST HERE (NEED MANUAL INTERVENTION ?)                        00001340
      WRITE(108,904) RMS                                                00001350
  904 FORMAT(//,' RMS DEVIATION OF SHIFTS',5X,F9.3,//)
C ABEND IF RMS GT 1
```

```
      IF(RMS.LE.1.)GOTO910
      WRITE(108,907)
  907 FORMAT(////,' ********** RMS GT 1 **********',////)
      CALL ABEND
  910 CONTINUE
C ABEND IF LESS THAN 5 POINTS USED
      IF(ISAMP.GE.4)GOTO920
      WRITE(108,908)ISAMP
  908 FORMAT(////,' ******** ONLY',I2,' SAMPLES FITTED ********',////)
      WRITE(108,909)
  909 FORMAT(' LOW S/N AND/OR SATURATED SPECTRUM'////)
      CALL ABEND
  920 CONTINUE
C UPDATE CONSTANT TERMS
      DCCON(22) = DCCON(22) + AVEX
      DCCON( 1) = DCCON( 1) + AVEY
C PUT SHIFTS IN DC RECORD
      DCSHFT(1) = AVEY
      DCSHFT(2) = AVEX
C FLAG TO BE READ BY LABMOD TO INDICATE DSPCON PUT OFFSETS IN DC
      DCSHFT(3) = 1.D0
      WRITE(108,906)AVEX,DCCON(22),AVEY,DCCON(1)
  906 FORMAT(    ' X SHIFT',F9.3,5X,' X CONSTANT',1X,D22.15,
     *       //,' Y SHIFT',F9.3,5X,' Y CONSTANT',1X,D22.15//)
C WRITE TO DISK
      CALL OPEN(IND,1,1,0,0,0)
      CALL WRITE(IND,1,0,0,0,400,BUF,0)
      CALL CLOSE(IND,1,0)
 9999 CALL END
      STOP                                                          00001370
      END                                                           00001380
      SUBROUTINE OFFSET(SAMPL,CW,SHIFT,BSTCOR)
C THIS SUBROUTINE CALCULATES THE SHIFT REQUIRED FOR EACH 15 X 15
C PIXEL SAMPLE; SIGMA 9 VERSION 2
C
C       MALLAMA       CSC        JULY 21,1978
C
C       LAST MODIFIED:  9/18/78      BY:   MALLAMA
C
      INTEGER*4 SAMPL(15,15),ROTSMP(15,8)                           00000030
      INTEGER*4 ROWSUM(15),SML,SML2                                 00000040
      REAL*4    TEMP(7),CORR(15)                                    00000050
      LOGICAL*4 CW                                                  00000060
      WRITE(108,102) ((SAMPL(I,J),I=1,15),J=1,15)                   00000070
  102 FORMAT(////,' INPUT MATRIX',///,15(/,15I5))
C TRANSFORM BY 45 DEGREES                                           00000090
C BASED ON CW PARAMETER                                             00000100
      IF (CW) GO TO 110
      DO 100 I=1,15                                                 00000120
      DO 100 J=1,8                                                  00000130
  100 ROTSMP(I,J)=SAMPL(I/2+J,J-(I+1)/2+8)                          00000140
      GO TO 130                                                     00000150
  110 DO 120 I=1,15                                                 00000160
      DO 120 J=1,8                                                  00000170
  120 ROTSMP(I,J)=SAMPL((I+1)/2-J+8,I/2+J)                          00000180
  130 WRITE(108,103) ((ROTSMP(I,J),J=1,8),I=1,15)                   00000190
```

```
  103 FORMAT(/////,' ROTATED MATRIX',///,15(/,8I5))                    00000200
C SUM ROWS IN ROTATED SAMPLE MATRIX                                    00000210
      DO 300 I=1,15                                                    00000220
      ROWSUM(I)=0                                                      00000230
      DO 200 J=1,8                                                     00000240
  200 ROWSUM(I)=ROWSUM(I)+ROTSMP(I,J)                                  00000250
  300 CONTINUE                                                         00000260
      WRITE(108,104) ROWSUM                                            00000270
  104 FORMAT(/////,' ROTATED MATRIX ROW SUMS',5X,15I5)                 00000280
C FIND SECOND LARGEST AND SECOND SMALLEST ROW SUMS                     00000290
      SML=1                                                            00000300
      SML2=1                                                           00000310
      LRG=1                                                            00000320
      LRG2=1                                                           00000330
      DO 400 I=1,15                                                    00000340
      IF(ROWSUM(I).LT.ROWSUM(SML)) SML=I                               00000350
  400 IF(ROWSUM(I).GT.ROWSUM(LRG)) LRG=I                               00000360
C
C NEW VERSION:  JUST USE LARGEST ROW SUM
C
C     GO TO 1000
      IF(SML.EQ.1)SML2=2                                               00000370
      IF(LRG.EQ.1)LRG2=2                                               00000380
      DO 450 I=1,15                                                    00000390
      IF(I.EQ.SML) GOTO425                                             00000400
      IF(ROWSUM(I).LT.ROWSUM(SML2)) SML2=I                             00000410
  425 IF(I.EQ.LRG) GOTO450                                             00000420
      IF(ROWSUM(I).GT.ROWSUM(LRG2)) LRG2=I                             00000430
  450 CONTINUE                                                         00000440
      WRITE(108,106) LRG,LRG2,SML,SML2                                 00000450
  106 FORMAT(' LARGEST, SECOND LARGEST, SMALLEST, SECOND SMALLEST',    00000460
     & 4(I4))                                                          00000470
C SPECIAL LOGIC FOR TRAILED SPECTRUM IN LARGE APERTURE
C     IF(ITRAIL.NE.2.OR.IAP.NE.1)GOTO500
C     AVEROW = (ROWSUM(LRG2)+ROWSUM(SML2))/2.
C     LCORR = 4
C     DO 475 I = 4,12
C     CORR(I) = (ROWSUM(I) - AVEROW)**2
C 475 IF(CORR(I).LT.CORR(LCORR))LCORR=I
C     GOTO750
C NORMALIZE TEMPLATE                                                   00000480
  500 TEMP(1) = ROWSUM(SML2)                                           00000490
      TEMP(7) = ROWSUM(SML2)                                           00000500
      TEMP(3) = ROWSUM(LRG2)                                           00000510
      TEMP(4) = ROWSUM(LRG2)                                           00000520
      TEMP(5) = ROWSUM(LRG2)                                           00000530
      TEMP(2) = (ROWSUM(LRG2)+ROWSUM(SML2))/2.                         00000540
      TEMP(6) = (ROWSUM(LRG2)+ROWSUM(SML2))/2.                         00000550
      WRITE(108,107) TEMP                                              00000560
  107 FORMAT(' NORMALIZED TEMPLATE VALUES',7(F9.2))                    00000570
C FIT WITH TEMPLATE                                                    00000580
C COMPUTE CORRELATIONS, AND FIND SMALLEST                              00000590
      LCORR = 4                                                        00000600
      DO 700 I=4,12                                                    00000610
      CORR(I) = 0.
      DO 600 J=1,7                                                     00000630
```

```
  600 CORR(I) = CORR(I) + (TEMP(J)-ROWSUM(I+J-4))**2
      IF(CORR(I).LT.CORR(LCORR)) LCORR=I                                 00000660
  700 CONTINUE                                                           00000670
  750 BSTCOR = CORR(LCORR)                                               00000680
      WRITE(108,108) (CORR(I),I=4,12),CORR(LCORR)                        00000690
  108 FORMAT(' CORRELATIONS',9(1PE10.2),//,' BEST CORRELATION',5X,       00000700
     & 1PE10.2)                                                          00000710
C. INTERPOLATE BETWEEN ROWS FIT WITH TEMPLATE                            00000720
      DEL = 0                                                            00000730
      IF(LCORR.EQ.4.OR.LCORR.EQ.12) GOTO800                              00000740
      DEL = (CORR(LCORR-1) - CORR(LCORR)) /                              00000750
     &      (CORR(LCORR-1) + CORR(LCORR+1) - 2*CORR(LCORR))              00000760
     &      -0.5                                                         00000770
  800 SHIFT = LCORR + DEL - 8.0
      IF(.NOT.CW)SHIFT=-SHIFT
C SATURATION CONDITION
      IF((FLOAT(ROWSUM(LRG2))/FLOAT(ROWSUM(SML2)).GT.1.5)
     & .AND. (ROWSUM(LRG2).LT.2000)) GO TO 801
      WRITE(108,802)
  802 FORMAT(//,' LOW S/N OR SATURATED SAMPLE')
      SHIFT = 99.
  801 WRITE(108,105) LCORR,DEL,SHIFT
C1000 IF(LRG.EQ.1.OR.LRG.EQ.15) GO TO 900
C     DEL = (ROWSUM(LRG-1) - ROWSUM(LRG)) /
C    &      (ROWSUM(LRG-1) + ROWSUM(LRG + 1) - 2*ROWSUM(LRG))
C    &      - 0.5
C 900 SHIFT = LRG + DEL - 8.0
C     WRITE(108,105) LRG,DEL,SHIFT
  105 FORMAT(    //,' BEST ROW              ',I2,
     &           //,' ROW INTERPOLATION     ',F8.4,                     00000820
     &           //,' TOTAL SHIFT           ',F8.4)                     00000830
      RETURN                                                             00000840
      END
      SUBROUTINE OFFTRL(PIXTRL,CW,SHIFT,BSTCOR)
C THIS SUBROUTINE CALCULATES THE SHIFT REQUIRED IN THE CASE OF A
C TRAILED SPECTRUM
C
C     MALLAMA   CSC   10-3-78
C
C     LAST MODIFIED BY: 10-4-78   BY: MALLAMA
C
      INTEGER*4       PIXTRL(41,3),PIXSUM(39),SML
      LOGICAL*4       CW
      REAL*4          CORR(39),TEMP(27)
C COMPUTE ROWSUMS
      IF(.NOT.CW)GO TO 200
      DO 100 I = 1,39
  100 PIXSUM(I) = PIXTRL(I+2,1) + PIXTRL(I+1,2) + PIXTRL(I,3)
      GO TO 400
  200 DO 300 I = 1,39
  300 PIXSUM(I) = PIXTRL(I,1) + PIXTRL(I+1,2) + PIXTRL(I+2,3)
  400 CONTINUE
      WRITE(108,499)((PIXTRL(I,J),I=1,41),J=1,3),PIXSUM
  499 FORMAT(//,'PIXTRL 1,2,3, PIXSUM',
     &       3(/,41I3),//,3X,39I3)
C FIND LARGEST AND SMALLEST ELEMENT AND AVERAGE
```

```
      LRG = 1
      SML = 1
      DO 500 I = 1,39
      IF(PIXSUM(I).LT.PIXSUM(SML))SML=I
  500 IF(PIXSUM(I).GT.PIXSUM(LRG))LRG=I
      WRITE(108,501)SML,LRG
  501 FORMAT(//,' SML',I3,5X,'LRG',I3)
C     AVE = (PIXSUM(LRG) + PIXSUM(SML))/2.
C TEST FOR S/N AND SATURATION
      APXSML = FLOAT(PIXSUM(SML))
      APXLRG = FLOAT(PIXSUM(LRG))
      IF(APXLRG/APXSML.GT.1.5 .AND. APXLRG.LT.764.99) GO TO 600
      WRITE(108,599)
  599 FORMAT(//,' LOW S/N OR SATURATED SAMPLE')
      SHIFT = 99.
      GO TO 900
  600 CONTINUE
C CREATE TEMPLATE
      TEMP(1) = APXSML
      TEMP(27) = APXSML
      TEMP( 2) = (APXSML*3. + APXLRG)/4.
      TEMP(26) = TEMP(2)
      TEMP( 3) = (APXSML + APXLRG)/2.
      TEMP(25) = TEMP(3)
      TEMP( 4) = (APXSML + APXLRG*3.)/4.
      TEMP(24) = TEMP(4)
      DO 610 I = 5,23
  610 TEMP(I) = APXLRG
      WRITE(108,619)TEMP
  619 FORMAT(/,' TEMP',27F4.0)
C FIT WITH TEMPLATE
      LCORR = 14
      DO 630 I = 14,26
      CORR(I) = 0.
      DO 620 J = 1,27
  620 CORR(I) = CORR(I) + (TEMP(J)-PIXSUM(I+J-14))**2
      IF(CORR(I).LT.CORR(LCORR))LCORR=I
  630 CONTINUE
      BSTCOR = CORR(LCORR)
      WRITE(108,639)(CORR(I),I=14,26),CORR(LCORR)
  639 FORMAT(' CORRELATIONS',13(1PE10.2),//,' BEST CORRELATION',5X,
     & 1PE10.2)
C INTERPOLATE BETWEEN ROWS
      DEL = 0.
      IF(LCORR.EQ.14.OR.LCORR.EQ.26)GOTO800
      DEL = (CORR(LCORR-1) - CORR(LCORR))/
     &      (CORR(LCORR-1) + CORR(LCORR+1) - 2.*CORR(LCORR))
     &      - 0.5
  800 SHIFT = 20. - (LCORR+DEL)
      WRITE(108,899)LCORR,DEL,SHIFT
  899 FORMAT(//,' BEST ROW                ',I2,
     &        //,' ROW INTERPOLATION       ',F8.4,
     &        //,' TOTAL SHIFT             ',F8.4)
C FIND HALF-MAX POINTS ON EACH SIDE OF MAXIMUM
C     LCORR1 = 1
C     LCORR2 = LRG
```

```
C      DO 700 I = 1,LRG
C      CORR(I) = (PIXSUM(I) - AVE)**2
C 700  IF(CORR(I).LT.CORR(LCORR1))LCORR1=I
C      DO 800 I = LRG,39
C      CORR(I) = (PIXSUM(I) - AVE)**2
C 800  IF(CORR(I).LT.CORR(LCORR2))LCORR2=I
CCCOMPUTE MIDPOINT OF HALF-MAX'S
C      AMID = (LCORR1 + LCORR2)/2.
C      SHIFT = (20. - AMID)
CCDIAGNOSTICS
C      WRITE(10A,999)SML,LRG,AVE,LCORR1,LCORR2,AMID,SHIFT
C 999  FORMAT(//,' SML',I3,3X,'LRG',I3,3X,'AVE',F5.1,3X,'LCORR1,2',
C     & 2I3,3X,'AMID',F5.1,//,' SHIFT',F10.4)
  900  RETURN
       END
```

REFERENCES

1. S. R. Heap, Goddard Space Flight Center, Private Communication, 1978

2. T. Snijders, University College London, IUE Working Paper, 1977

3. R. A. White, Computer Sciences Corporation, private Communication, 1978

4. P. M. Perry, and B. E. Turnrose, "IUE Image Processing Overview and Mathematical Description," Computer Sciences Corporation, CSC/TM-77/6250, 1977

5. J. J. Lorre, "Application of Digital Image Processing Techniques to Astronomical Imagery, 1978," JPL Publication 78-91, 1978