LA-4891-MS

AN INFORMAL REPORT

MASTER

Two-Dimensional Search and Interpolation on a Distorted Rectangular Grid.

Program SI ϕ G2D

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ATOMIC ENERGY COMMISSION
CONTRACT W-7408-ENG. 36

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

In the interest of prompt distribution, this LAMS report was not edited by the Technical Information staff.

Printed in the United States of America. Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy \$3.00; Microfiche \$0.95

LA-4891-MS
An Informal Report
UC-32

ISSUED: February 1972

Two-Dimensional Search and Interpolation on a Distorted Rectangular Grid.

Program SI ϕ G2D

by

Robert A. Jameson
Rene S. Mills

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

TWO-DIMENSIONAL SEARCH AND INTERPOLATION
ON A DISTORTED RECTANGULAR GRID - PROGRAM SIΦG2D

by
Robert A. Jameson
and
Rene S. Mills

This program is used to restore data from a known, but distorted, rectangular grid to a true grid.

A. Determine square of grid in which data point lies.

Sample digitizing pattern for fiducial grid.

Typical distorted fiducial grid square containing a data point.

Line ①

$$x = ny + b$$

$$b = x_1 - \frac{\Delta x \textcircled{1}}{\Delta y \textcircled{1}} y_1$$

$$x = \frac{\Delta x \textcircled{1}}{\Delta y \textcircled{1}} (y - y_1) + x_1 \quad (1)$$

$$\Delta x \textcircled{1} = x_3 - x_1$$

$$\Delta y \textcircled{1} = y_3 - y_1$$

Line ②

$$y = \frac{\Delta y \textcircled{2}}{\Delta x \textcircled{2}} (x - x_3) + y_3 \quad (2)$$

The grid squares must be examined successively to see which one contains the data point, (x,y).

1. Look to left of line ③:

Substitute data point y into the equation for line ③, solve for $\bar{x}$. If $\bar{x} \geq x$, then the point is to the left of line ③.

2. Look below line ②:

Substitute x into the equation for line ②, solve for $\bar{y}$. If $\bar{y} \geq y$, then the point is below line ②.

3. In the same manner, look to right of line ① and above line ④.

4. A square of type $\triangle A$ with sides 6-12 and 7-13 is fictitious, but can be calculated through - e.g., there will not be any points to left of side 7-13.

B. Interpolate within the square.

The fractional distance along opposite sides is equal in an ideal grid, and this condition must apply to the real grid, assuming linear distortion.

Typical ideal rectangular grid.

1. x-Interpolation

Along side ②:

$$\Delta x_{②} = x_4 - x_3$$

$$\Delta y_{②} = y_4 - y_3$$

$$\text{Distance} = \sqrt{\Delta x^2 + \Delta y^2}$$

$$f \cdot D = \sqrt{\Delta x^2 + \Delta y^2}; 0 \leq f \leq 1$$

$$= \sqrt{(f\Delta x)^2 + (f\Delta y)^2} \quad (3)$$

So the coordinates of any point between (x_3, y_3) and (x_4, y_4) can be written as

$$(x_3, y_3) + (f\Delta x_{②}, f\Delta y_{②}) \quad (4)$$

Same along side ④:

$$(x_1, y_1) + f\Delta x_{④}, f\Delta y_{④}; \quad (5)$$

These two points and the data point must be on the same straight line, with slope = m and intercept = b . This gives 3 equations in unknowns f , m and b ; must solve for f ; choose x as the independent variable so the resulting slope will be finite:

$$x = my + b \quad (6)$$

$$x_3 + f\Delta x_{②} = m(y_3 + f\Delta y_{②}) + b \quad (7)$$

$$x_1 + f\Delta x_{④} = m(y_1 + f\Delta y_{④}) + b \quad (8)$$

The equations are non-linear; one must watch for special cases.

Eliminating b :

$$x_3 + f\Delta x_{②} = m(y_3 + f\Delta y_{②}) + x - my$$

$$x_1 + f\Delta x_{④} = m(y_1 + f\Delta y_{④}) + x - my$$

$$\Delta x_{②} f = m(y_3 - y) + (x - x_3) + m\Delta y_{②} f$$

$$\Delta x_{④} f = m(y_1 - y) + (x - x_1) + m\Delta y_{④} f$$

$$[\Delta y_{④} \Delta x_{②} - \Delta y_{②} \Delta x_{④}] f$$

$$= m[\Delta y_{④} (y_3 - y) - \Delta y_{②} (y_1 - y)]$$

$$+ [\Delta y_{④} (x - x_3) - \Delta y_{②} (x - x_1)] \quad (9)$$

$$af = bm + c$$

$$m = \frac{a}{b} f - \frac{c}{b} = Af + B \quad (\text{Note: } B = -\frac{c}{b}) \quad (10)$$

Substituting:

$$0 = \Delta y_{④} f^2 + [A(y_1 - y) + B\Delta y_{④} - \Delta x_{④}] f$$

$$+ [B(y_1 - y) + (x - x_1)] \quad (11)$$

$$= Cf^2 + Df + E \quad (12)$$

$$(\text{Note: if } C < 10^{-4}, f = -\frac{E}{D}) \quad (13)$$

Then:

$$f = -\frac{D}{2C} \pm \sqrt{\left(\frac{D}{2C}\right)^2 - \frac{E}{C}} \quad (14)$$

Choose root $0 \leq f \leq 1$

2. y-Interpolation

For y -interpolation, use equations with y as the independent variable:

$$y = nx + b$$

$$y_1 + f\Delta y_{①} = n(x_1 + f\Delta x_{①}) + b$$

$$y_2 + f\Delta y_{③} = n(x_2 + f\Delta x_{③}) + b$$

$$[\Delta y_{①} \Delta x_{③} - \Delta y_{③} \Delta x_{①}] f$$

$$= n[\Delta x_{③} (x_1 - x) - \Delta x_{①} (x_2 - x)]$$

$$+ [\Delta x_{③} (y - y_1) - \Delta x_{①} (y - y_2)] \quad (15)$$

$$af = bn + c$$

$$n = \frac{a}{b} f - \frac{c}{b} = Af + B \quad (\text{Note: } B = -\frac{c}{b}) \quad (16)$$

$$0 = A\Delta x_1 f^2 + [A(x_1 - x) + B\Delta x_1 - \Delta y_1]f + [B(x_1 - x) + (y - y_1)] \quad (17)$$

$$= Cf^2 + Df + E \quad (18)$$

3. Special cases:

a. x-Interpolation

$$\Delta x_2 f = m(y_3 - y) + (x - x_3) + m\Delta y_2 f \quad (19)$$

$$\Delta x_4 f = m(y_1 - y) + (x - x_1) + m\Delta y_4 f \quad (20)$$

$$\begin{aligned} & [\Delta y_4 \Delta x_2 - \Delta y_2 \Delta x_4]f \\ &= m[\Delta y_4 (y_3 - y) - \Delta y_2 (y_1 - y)] \\ &+ [\Delta y_4 (x - x_3) - \Delta y_2 (x - x_1)] \quad (21) \end{aligned}$$

$$(1) \text{ If } x = x_1 \text{ or } x = x_3; f = 0: \quad (22)$$

$$(2) \text{ If } \Delta y_2 = \Delta y_4 = 0:$$

Eq. (21) $\rightarrow 0 = m \cdot 0 + 0$

Going back to Eqs. (19) and (20)

$$\Delta x_2 f = m(y_3 - y) + (x - x_3)$$

$$\Delta x_4 f = m(y_1 - y) + (x - x_1)$$

Eliminate m:

$$f = \frac{(y_1 - y)(x - x_3) - (y_3 - y)(x - x_1)}{\Delta x_2 (y_1 - y) - \Delta x_4 (y_3 - y)} \quad (23)$$

$$(3) \text{ If } \Delta y_2 = 0, \text{ but } \Delta y_4 \neq 0:$$

Using Eqs. (19) and (20), and eliminating f from the left sides;

$$\begin{aligned} m &= \frac{[\Delta x_4 (x - x_3) - \Delta x_2 (x - x_1)]}{\Delta x_2 \Delta y_4 f - [\Delta x_4 (y_3 - y) - \Delta x_2 (y_1 - y)]} \\ &= \frac{A}{Bf - G} \quad (24) \end{aligned}$$

Substituting in Eq. (19) yields another quadratic:

$$\begin{aligned} & [\Delta x_2 \cdot B]f^2 - [\Delta x_2 \cdot G + B(x - x_3)]f \\ &+ [G(x - x_3) - A(y_3 - y)] = 0 \quad (25) \end{aligned}$$

$$Cf^2 + Df + E = 0 \quad (26)$$

(Note: $D = -[]$ in Eq. (25))

(4) If $\Delta y_2 \neq 0$, but $\Delta y_4 = 0$; the quantity B in Eq. (24) changes to:

$$B = \Delta x_4 \cdot \Delta y_2 \quad (27)$$

b. y-Interpolation

$$(1) \text{ If } y = y_1, \text{ or } y = y_3; f = 0. \quad (28)$$

$$(2) \text{ If } \Delta x_1 = \Delta x_3 = 0:$$

$$f = \frac{(x_2 - x)(y - y_1) - (x_1 - x)(y - y_2)}{\Delta y_1 (x_2 - x) - \Delta y_3 (x_1 - x)} \quad (29)$$

$$(3) \text{ If } \Delta x_1 = 0, \text{ but } \Delta x_3 \neq 0:$$

$$\begin{aligned} n &= \frac{[\Delta y_3 (y - y_1) - \Delta y_1 (y - y_2)]}{[\Delta x_3 \Delta y_1]f - [\Delta y_3 (x_1 - x) - \Delta y_1 (x_2 - x)]} \\ &= \frac{A}{Bf - G} \quad (30) \end{aligned}$$

$$\begin{aligned} & [\Delta y_1 \cdot B]f^2 - [\Delta y_1 \cdot G + B(y - y_1)]f \\ &+ [G(y - y_1) - A(x_1 - x)] = 0 \quad (31) \end{aligned}$$

$$Cf^2 + Df + E = 0$$

(4) If $\Delta x_1 \neq 0$, but $\Delta x_3 = 0$; the quantity B in Eq. (30) is

$$B = \Delta x_1 \Delta y_3$$

C. Coordinate system rotation

If the coordinate system is essentially vertical and horizontal, a rotation as the points are entered will eliminate most of the special cases above.

$$\begin{aligned} (x + jy)(\cos \theta + j \sin \theta) &= (x \cos \theta - y \sin \theta) \\ &+ j(x \sin \theta + y \cos \theta) \quad (32) \end{aligned}$$

Determination of the grid square is the same as in section A. Interpolation is accomplished using Eqs. (9) - (14) and (15) - (18) with the condition of Eq. (13) to avoid the case where the slope of opposite sides is equal.

D. Program listing

Format is for pepper-pot tests of the LAMPF H⁺ injector system.

```

PROGRAM SIGG2D (INPUT,OUTPUT,PUNCH,TAPE9=OUTPUT,TAPE10=INPUT)
DIMENSION NSTART(25,3), NDOTS(25,3), X(4000), Y(4000)
PEWIND 9

```

```

C      NX,NY = NUMBER OF X AND Y LINES OF GRID. CHANGE AS NECESSARY
      NX=6
      NY=6
C      READ RAW DATA INTO FILES, ROTATE IT 30 DEGREES
C      NSWITCH=0 FOR GRID, 1 FOR DATA, 2 FOR SECONDARY DOTS IF DESIRED
C      NUMBER OF PTS IN GRID AND NUMBER OF DATA PTS ARE COUNTED
      A=SDG(30.) $ R=CDG(30.)
      IFLAG=NPHOTO=NPOINT=NSTART(1,1)=NSTOP=NDOTS(1,1)=KOUNT=0
1     READ 3, NSWITCH, YP, XP
3     FORMAT (I1,13X,2F5,0)
      IF (EOF,10) 13=5
5     NPOINT=NPOINT+1
      Y(NPOINT)=XP*A+YP*R
      X(NPOINT)=XP*R-YP*A
      IF (NSWITCH.NE.0) GO TO 9
      ISWITCH=0
      IF (IFLAG.NE.0) GO TO 7
      NPHOTO=NPHOTO+1
      NTRACE=1
      IFLAG=1
      NSTART(NPHOTO,NTRACE)=NSTOP+1
7     NSTOP=NPOINT
      NDOTS(NPHOTO,NTRACE)=NSTOP-NSTART(NPHOTO,NTRACE)+1
      GO TO 1
9     IFLAG=0
      IF (NSWITCH.EQ.ISWITCH) GO TO 11
      ISWITCH=NSWITCH
      NTRACE=NTRACE+1
      NSTART(NPHOTO,NTRACE)=NSTOP+1
11    NSTOP=NPOINT
      NDOTS(NPHOTO,NTRACE)=NSTOP-NSTART(NPHOTO,NTRACE)+1
      GO TO 1
13    CONTINUE
      NG=NX*NY
      NK=NG-NX-1
      DO 61 I=1,NPHOTO
        LSTART=NSTART(I,1)
        LDOTS=NDOTS(I,1)
C      CHECK FOR PROPER NUMBER OF GRID POINTS
        IF (LDOTS.EQ.NG) GO TO 17
        PRINT 15, I
15     FORMAT (* GRID BAD, PHOTO*,I6)
        GO TO 61
C      NTRACE WILL EQUAL 2 IF ONLY ONE SET OF POINTS PER PHOTO ARE DIGITIZED
17    DO 59 ITR=2,NTRACE
        MSTART=NSTART(I,ITR)
        MDOTS=NDOTS(I,ITR)
        DO 57 J=1,MDOTS
          K=MSTART+J-1
          DO 25 L=1,NK
            X1=X(LSTART+L-1) $ Y1=Y(LSTART+L-1)
            X2=X(LSTART+L) $ Y2=Y(LSTART+L)
            X3=X(LSTART+L-1+NX) $ Y3=Y(LSTART+L-1+NX)
            X4=X(LSTART+L+NX) $ Y4=Y(LSTART+L+NX)
            DX1=X3-X1 $ DY1=Y3-Y1
            DX2=X4-X3 $ DY2=Y4-Y3
            DX3=X4-X2 $ DY3=Y4-Y2
            DX4=X2-X1 $ DY4=Y2-Y1
C      LINE 1 IS RIGHT, 2 IS TOP, 3 IS LEFT, 4 IS BOTTOM OF SQUARE
C      SEE IF X IS TO LEFT OF LINE 3
            XL=(Y(K)-Y2)*(DX3/DY3)+X2
            IF (XL-X(K)) 25,25,19
19     XL=(Y(K)-Y1)*(DX1/DY1)+X1
            IF (X(K)-XL) 25,25,21

```

```

C      SEF IF Y IS BELOW LINE 2
21     YL=(X(K)-X3)*DY2/DX2+Y3
      IF (YL-Y(K)) 25,25,23
23     YL=(X(K)-X1)*DY4/DX4+Y1
      IF (Y(K)-YL) 25,25,27
25     CONTINUE

C      INTERPOLATE WITHIN SQUARE ON X
27     IFLAG=1YF=0
      IF (Y1-Y(K)) 29,31,29
29     IF (Y3-Y(K)) 33,31,33
31     F=0.
      GO TO 43
33     DIV=1./((DY4*(Y3-Y(K))-DY2*(Y1-Y(K)))
      A=((DY4*DX2-DY2*DX4)*DIV
      B=((DY2*(X(K)-X1)-DY4*(X(K)-X3))*DIV
      C=A*DY4
      D=A*(Y1-Y(K))+B*DY4-DX4
      F=B*(Y1-Y(K))+X(K)-X1
35     IF (C.LT..0001) F=-E/D
      IF (C.LT..0001) GO TO 43
      F=SQRT((D/(2*C))*2-E/C)
      F1=(-D/(2*C))+F
      F2=F1-2*F
      IF (F1.LT.0..OR.F1.GT.1.) GO TO 37
      F=F1
      IFLAG=1
37     IF (F2.LT.0..OR.F2.GT.1.) GO TO 39
      IF (IFLAG.EQ.1) GO TO 41
      F=F2
      GO TO 43
39     IF (IFLAG.EQ.1) GO TO 43
C      X AND Y MADE IMPOSSIBLY LARGE TO IDENTIFY ANY BAD POINTS
41     IF (IYF.EQ.0) XP=20.
      IF (IYF.EQ.1) YP=20.
      GO TO 47
43     IF (IYF.EQ.1) GO TO 45
      LX=L-1
      XP=MOD(LX,NX)+F
      GO TO 47
45     LX=L/NX
      YP=LX+F
      GO TO 55

C      INTERPOLATE WITHIN SQUARE ON Y
47     IF (IYF.EQ.1) GO TO 55
      IYF=1 & IFLAG=0
      IF (X1-X(K)) 49,51,49
49     IF (X2-X(K)) 53,51,53
51     F=0.
      GO TO 45
53     DIV=1./((DX3*(X1-X(K))-DX1*(X2-X(K)))
      A=(DY1*DX3-DY3*DX1)*DIV
      B=(DX1*(Y(K)-Y2)-DX3*(Y(K)-Y1))*DIV
      C=A*DX1
      D=A*(X1-X(K))+B*DX1-DY1
      E=B*(X1-X(K))+Y(K)-Y1
      GO TO 35
55     X(K)=XP
      Y(K)=YP
57     CONTINUE
59     CONTINUE
61     CONTINUE

```



```

C      PRINT AND PUNCH COMPRESSED DATA DECKS. (FOR PEPPER POT, LAST POIN
C      ALWAYS CENTER OF QUARTZ)  MODIFY OUTPUT AS DESIRED.
      DO 73 I=1,NPHOTO
      DO 71 J=2,NTRACE
      MDOTS=NDOTS(I,J)-1
      MPOINT=NSTART(I,J)
      PRINT 63
63     FORMAT (21*)
      PRINT 65, MDOTS,I
      PUNCH 65, MDOTS,I
65     FORMAT (214)
      LDOTS=MDOTS*MPOINT-1
      PRINT 67, (X(K),Y(K),K=MPOINT,LDOTS)
      PUNCH 67, (X(K),Y(K),K=MPOINT,LDOTS)
      PRINT 69, X(LDOTS+1),Y(LDOTS+1)
      PUNCH 69, X(LDOTS+1),Y(LDOTS+1)
67     FORMAT (10F8.3)
69     FORMAT (F6.1,F8.1)
71     CONTINUE
73     CONTINUE
      END

```