BELLCOMM, INC.
955 L'ENFANT PLAZA NORTH, S.W. WASHINGTON, D.C. 20024

B69 04102

SUBJECT: Computer Program Abstract and Write-up for SFLARE - Solar Flare Detection - Case 103-6

DATE: April 25, 1969

FROM: R. R. Singers

MEMORANDUM FOR FILE

Attached for your information is the Computer Program Abstract and Write-up for

SFLARE - Solar Flare Detection.

SFLARE was designed to verify an algorithm to detect solar flares and to show the feasibility of an on-board solar flare monitoring system.

A copy of this program is available from the Computer Program Library.

R. R. Singers

1033-RRS-jf

R. R. Singers

Attachment

Copy to
(See last page of document.)

(NASA-CR-106038) COMPUTER PROGRAM ABSTRACT AND WRITE-UP FOR SFLARE - SOLAR FLARE DETECTION (Bellcomm, Inc.) 25 p

N79-73431

Unclas
00/92 11536

OCT 1969
RECEIVED
NASA STI FACILITY
INPUT BRANCH

COMPUTER PROGRAM ABSTRACT

Identification

TITLE: SFLARE - Solar Flare Detection

AUTHOR: R. R. Singers

SPONSOR: R. K. Agarwal

DATE: January 26, 1969

LANGUAGE: FORTRAN V

KEYWORDS: Solar, Flare, Pattern recognition

Purpose

SFLARE was designed to verify an algorithm to detect solar flares and to show the feasibility of an on-board solar flare monitoring system.

Method

SFLARE uses an algorithm developed by the sponsor to detect solar flar activity. A digitized picture of the sun with flare activity is analyzed with the algorithm detecting flare activity above a given threshold to simulate the monitoring system.

COMPUTER PROGRAM WRITE-UP

Identification

TITLE: SFLARE - Solar Flare Detection

AUTHOR: R. R. Singers

SPONSOR: R. K. Agarwal

DATE: January 26, 1969

LANGUAGE: FORTRAN V

KEYWORDS: Solar, Flare, Pattern recognition

Purpose

SFLARE was designed to verify an algorithm to detect solar flares and to show the feasibility of an on-board solar flare monitoring system.

Method

SFLARE uses an algorithm developed by the sponsor [1] to detect solar flare activity. A digitized picture of the sun with flare activity is analyzed with the algorithm detecting flare activity above a given threshold to simulate the monitoring system.

Usage

SFLARE uses a binary (free format) tape input on unit 2, logical unit B. The tape contains 1001 records each containing 1024 words. Each tape is the digitized results of a solar flare picture.* The picture was divided into 1001 lines (records) with 1024 picture elements (words) per line. These elements were then converted to intensity values between

---

*See [1] for a sample of the pictures which were digitized.

0 and 63 (brightest). The results were then stored on a tape, each line corresponding to a record and each element of a line corresponding to a word of that record (See Figure 3 for Table of Bellcomm tapes used and their corresponding threshold values.).

Output is in the form of a table containing the area, centroid, average and peak intensities of each flare (Figure 1). A picture is then plotted (SC-4020)* of the sun and all detected flares to show relative position (Figure 2a). If any flares of importance (area greater than 158 elements) are detected, one or more of the following pictures are also plotted:

1. the sun and flares of importance 1

   (158< area <396)

2. the sun and flares of importance 2

   (395< area <953)

3. the sun and flares of importance 3(+)**

   (592< area) .

(See Figure 2b.)

Sample Run

Hydrogen line filtered pictures of the sun showing flare activity were digitized at Bell Laboratories and used to test the algorithm. The processed pictures can be found in [1]. Sample outputs from one of the pictures are shown in Figures 1 and 2. In this run, the threshold constant (arbitrary intensity value which distinguishes the flares from their background) was set to 20.

---

*These routines are found on the first two files of $SCPLT.

**3(+) implies the inclusion of flares of importance 4(1895< area) with those of importance 3.

Program Description

All arrays are variably dimensioned for ease of change. The four parameter variables are defined as follows:

NEL - number of elements (e)

NOB - number of objects (ϕ)

NLI - number of lines (L)

LPERO - number of lines per object (L_K)

The relationship between these parameters (e, ϕ, L, L_K) may be found in [1] in the form of a nomograph. The threshold constant, Q, is also a parameter variable.

It should be noted that the algorithm as programmed is not designed for speed. In particular, the search for an unused line and the check for used lines may be refined by "marking" the lines in the master list and doing all searches and/or line checks from the master list. One way of doing this is with the sign bit, the sign of an unused line being + and a used line -. The line check then involves checking only the sign bit of the line involved instead of comparing the line with all previously assigned lines. A similar time decrease is attained in the search for unused lines.

THE NUMBER OF ELEMENTS IS : 529
THE NUMBER OF LINES IS : 73

RESULTS OF FLARE DETECTION:

OBJECT	AREA	CENTROID Y	CENTROID X	INTENSITY PEAK	INTENSITY AVERAGE
1	60	401	640	22	20
2	6	412	731	20	20
3	4	449	873	20	20
4	427	467	864	27	22
5	32	455	885	21	20

FIGURE 1

SOLAR FLARES OF IMPORTANCE [illegible]

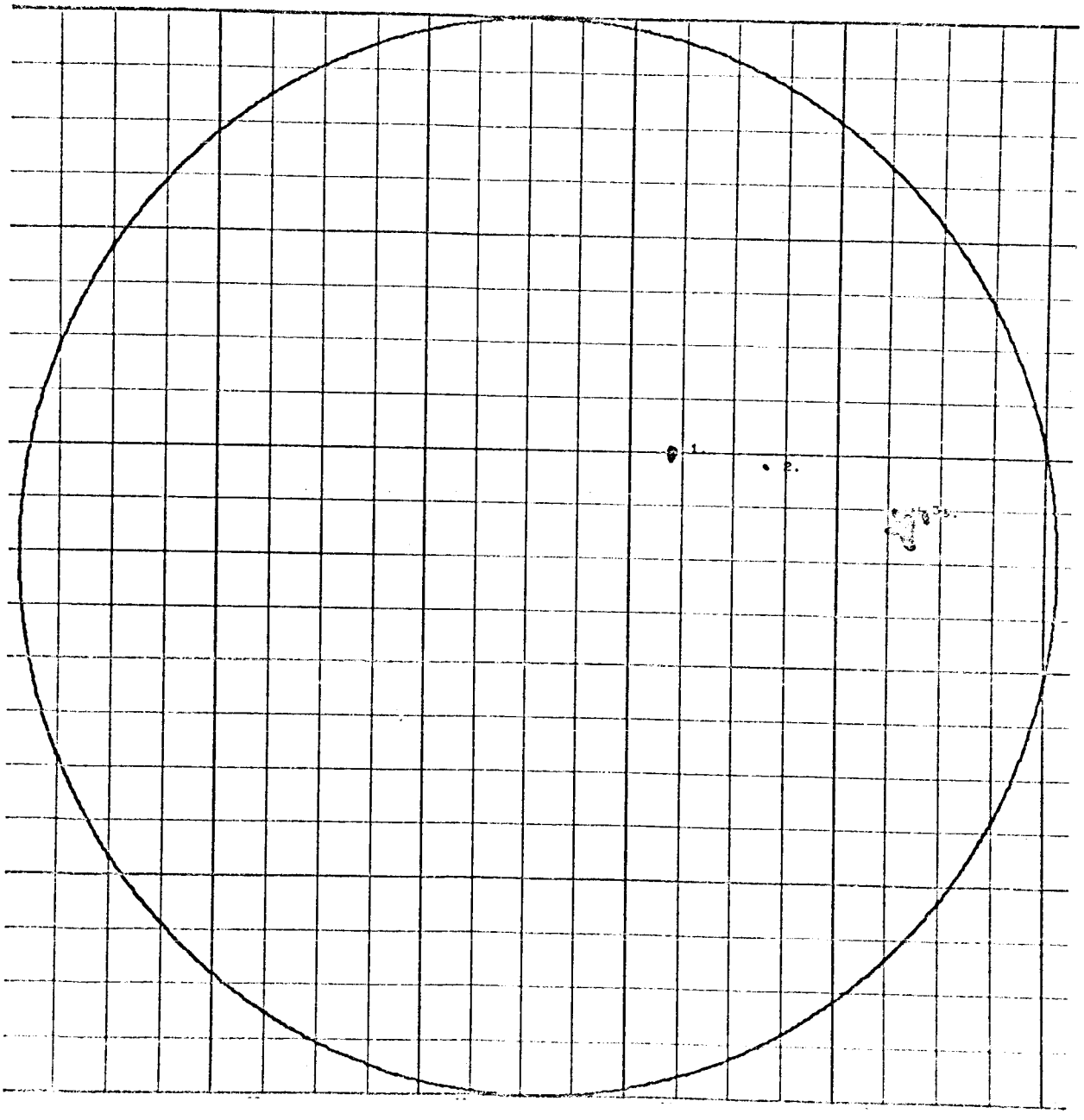

FIGURE 2a

SOLAR FLARES OF IMPORTANCE TWO

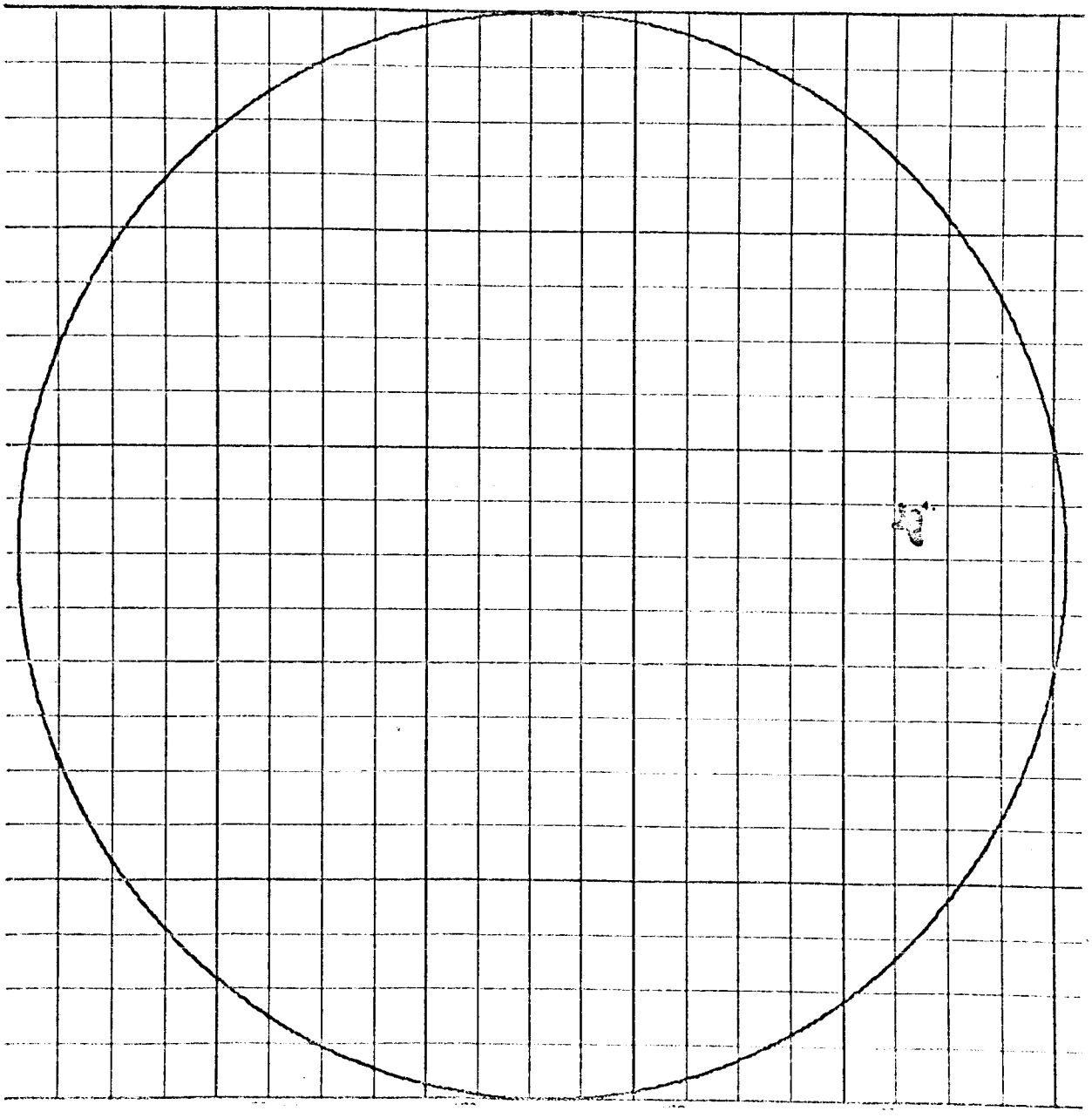

FIGURE 2b

42-381 50 SHEETS
42-382 100 SHEETS
NATIONAL

Table of Bellcomm Solar Flare Tapes

Picture	Tape Number	Threshold Value Q
1	1843	36
2	2180	20
3	1667	27
4	1420	28

Figure 3

REFERENCE

1. "Automatic Spaceborn Solar Flare Detection", R. K. Agarwal, Case 103-7, (to be published).

APPENDIX A

Listing of the program as set up for the sample run.

```
@ .     FOR       SFLARE,SFLARE *
C
C TITLE           SFLARE    -SOLAR FLARE DETECTION
C
C AUTHOR          R.R.SINGERS
C
C SPONSOR         R.K.AGARWAL
C
C DATE            1-26-69
C
C KEY WORDS       SOLAR,FLARE,PATTERN RECOGNITION
C
C PURPOSE         TO VERIFY AN ALGORITHM TO DETECT SOLAR FLARES
C
C METHOD          AN ALGORITHM DEVELOPED BY THE SPONSOR
C
C INPUT        --FILE B
C                P    -A VECTOR OF 1024 ELEMENTS REPRESENTING ONE
C                      GRID SCAN LINE(1000 SCANS PER FRAME).
C
C OUTPUT       --PRINTER
C                NOHL    -NUMBER OF ELEMENTS FOUND TO BELONG TO FLARES.
C                NOLINE  -NUMBER OF LINE SEGMENTS FORMED BY THESE
C                         ELEMENTS.
C                AREA    -AREA OF EACH OBJECT(FLARE).
C                YBAR,   -CENTROID OF EACH OBJECT.
C                XBAR
C                IPEAK,  -PEAK AND AVERAGE INTENSITY OF EACH OBJECT.
C                IAVE
C
C              --SC4020
C                PLOT OF SUN AND ALL FLARES DETECTED.
C                PLOTS OF FLARES OF IMPORTANCE 1,2,OR 3 IF ANY EXIST.
C
C
C       PARAMETER STATEMENTS.
C
                 PARAMETER NEL=4000,NOB=50,NL1=250
                 PARAMETER LPERO=100
C
C
C       THRESHOLD CONSTANT...
C
                 PARAMETER Q=28
C
C
C       SPECIFICATION STATEMENTS.
C
                 IMPLICIT INTEGER (A-Z)
                 DIMENSION P(NEL),PP(NEL)
                 DIMENSION D(NOB,LPERO),L2(NL1),LENGTH(NL1),L(NL1,2),
     .           S(NEL),NOBJ(NOB   ),A(NOB),AR(NOB),ARR(NOB)
                 DIMENSION AE(NOB)
                 COMMON LENGTH,L
C
C       ...DEFINE STATEMENTS
C
                 DEFINE IAV(I)=FLD(0,6,AR(I))
```

```
      DEFINE PIAV(I)=FLD(6,6,AR(I))
      DEFINE DIAV(I)=FLD(12,6,AR(I))
      DEFINE IPEAK(I)=FLD(18,6,AR(I))
      DEFINE PIPEAK(I)=FLD(24,6,AR(I))
      DEFINE DIPEAK(I)=FLD(30,6,AR(I))
      DEFINE AREA(I)=FLD(0,12,ARR(I))
      DEFINE PAREA(I)=FLD(12,12,ARR(I))
      DEFINE DAREA(I)=FLD(24,12,ARR(I))
      DEFINE XBAR(I)=FLD(0,12,A(I))
      DEFINE YBAR(I)=FLD(12,12,A(I))
      DEFINE PXBAR(I)=FLD(24,12,A(I))
      DEFINE PYBAR(I)=FLD(0,12,AE(I))
      DEFINE LL(I,J)=L(I,J)
      DEFINE LINE(I)=LENGTH(I)
      SH=20
C
      READ(2)(P(L),L=1,1024)
 5    CONTINUE
      FRAME=FRAME+1
C
C     INITIALIZE FRAME PARAMETERS.
C
      K=1
      I=0
C
C     READ IN P ARRAY.
C
 10   CONTINUE
C
C     INCREMENT I AND CHECK FOR MAXIMUM.
C
      I=I+1
      IF(I.GT.999) GO TO 20
      READ(2)(P(L),L=I,1024)
      J=0
 15   CONTINUE
C
C     INCREMENT J AND CHECK FOR MAXIMUM.
C
      J=J+1
      IF(J.GT.1024) GO TO 10
C
C     CHECK IF INTENSITY LESS THAN CRITICAL VALUE.
C
      IF(P(J).LT.0) GO TO 15
C
C     ... IF ABOVE, STORE IN MASTER LIST.
C
      FLD(0,11,S(K))=I
      FLD(11,11,S(K))=J
      FLD(22,6,S(K))=P(J)
C
C     INCREMENT K AND CHECK NEXT POINT.
C
      K=K+1
      GO TO 15
 20   CONTINUE
C
```

```
C     END OF FRAME, SAVE THE NUMBER OF ELEMENTS IN THE MASTER LIST.
C
            NOML=K-1
C
C     INITIALIZE PARAMETERS.
C
            I=1
            J=1
            K=1
C
C     START INITIAL LINE SEGMENT.
C
            L(1,1)=S(1)
            LKJ=S(1)
            L(1,2)=1
 25         CONTINUE
C
C     INCREMENT I AND CHECK FOR MAXIMUM.
C
            I=I+1
            IF(I.GT.NOML)GO TO 35
C
C     CHECK IF NEXT POINT IS ON PRESENT LINE.
C
            IF(FLD(0,11,LKJ).NE.FLD(0,11,S(I))) GO TO 30
            IF(FLD(11,11,LKJ)+1.NE.FLD(11,11,S(I))) GO TO 30
C
C     IT IS ... STORE IN LINE AND CONTINUE.
C
            J=J+1
            LKJ=S(I)
            GO TO 25
 30         CONTINUE
C
C     IT IS NOT ... NOTE LENGTH OF PRESENT LINE AND START NEXT ONE.
C
            LENGTH(K)=J
            K=K+1
            J=1
            L(K,1)=S(I)
            LKJ=S(I)
            L(K,2)=I
            GO TO 25
 35         CONTINUE
C
C     SAVE THE LAST LINE LENGTH AND THE NO. OF LINES FORMED.
C
            LENGTH(K)=J
            NOLINE=K
            WRITE(6,500) NOML,NOLINE
C
C     INITIALIZE PARAMETERS.
C
            LB=0
            LE=0
            SP=0
            LP=0
            I=1
```

```
      J=1
      K=1
      C=0
      B=0
      E=0
      D(1,1)=1
      JK=1
40    CONTINUE
C
C     INCREMENT I AND CHECK FOR MAXIMUM.
C
      I=I+1
      IF(I.GT.NOLINE) GO TO 185
      DO 41 T=1,K
      JT=NOBJ(T)
      IF(T.EQ.K) JT=J
      DO 41 KK=1,JT
      IF(D(T,KK).NE.I) GO TO 41
      GO TO 40
41    CONTINUE
C
C     CHECK Y VARIATION.
C
      IF(FLD(0,11,L(I,1))-FLD(0,11,LL(D(K,J),1))-1)45,50,105
45    CONTINUE
C
C     SAME Y - MARK B REGISTER.
C
      B=B+1
      IF(J.EQ.1) GO TO 40
C
C     ...COMPUTE X VARIATION AND MARK SECONDARY LIST IF CONNECTED.
C
      DIST=FLD(11,11,LL(D(K,J-1),1))-FLD(11,11,L(I,1))
      IF(LINE(D(K,J-1)).LT.ABS(DIST)) GO TO 40
      CALL TSUM(I)
      DO 46 II=1,LP
      IF(L2(II).EQ.I) GO TO 40
46    CONTINUE
      LP=LP+1
      L2(LP)=I
      GO TO 40
50    CONTINUE
C
C     Y DIFFERS BY 1 - COMPUTE X VARIATION.
C
      DIST=FLD(11,11,LL(D(K,J),1))-FLD(11,11,L(I,1))
C
C     CHECK IF LINES ARE CONNECTED -IF NOT, NOTE AND CONTINUE.
C
      IF(DIST)65,70,60
60    CONTINUE
      IF(LENGTH(I).GE.ABS(DIST)) GO TO 70
      C=C+1
      GO TO 40
65    CONTINUE
      IF(LINE(D(K,J)).LT.ABS(DIST)) GO TO 105
70    CONTINUE
```

```
C
C       IF CONNECTED, PLACE IN OBJECT AND USE IT AS CURRENT LINE.
C
                J=J+1
                D(K,J)=I
C
C       CHECK E REGISTER - IF NOT EMPTY, CHECK POSSIBLE CONNECTIONS.
C
                IF(E.EQ.0) GO TO 90
                LE=0
 75             CONTINUE
                LE=LE+1
                DIST=FLD(I1,I1,LL(D(K,J),1))-FLD(I1,I1,LL(D(K,J-1)-LE,1))
                IF(DIST.LE.0) GO TO 80
                IF(LINE(D(K,J-1)-LE).LT.ABS(DIST)) GO TO 85
 80             CONTINUE
C
C       ...CONNECTED, MARK AND CONTINUE UNTIL E IS DELETED.
C
                DO 81 II=1,LP
                IF(L2(II).EQ.D(K,J-1)-LE) GO TO 82
 81             CONTINUE
                LP=LP+1
                L2(LP)=D(K,J-1)-LE
 82             CONTINUE
                IF(LE.LT.E) GO TO   75
 85             CONTINUE
C
C       ...NOT CONNECTED, CLEAR E REGISTER.
C
                E=0
 90             CONTINUE
C
C       PUT C REGISTER IN E REGISTER AND CLEAR C.
C
                E=C
                C=0
C
C       CHECK B REGISTER - IF NOT EMPTY, CHECK POSSIBLE CONNECTIONS.
C
                IF(B.EQ.0) GO TO 40
                LB=0
 95             CONTINUE
                LB=LB+1
                DIST=FLD(I1,I1,LL(D(K,J),1))-FLD(I1,I1,LL(D(K,J-1)+LB,1))
                IF(LINE(D(K,J)).LT.ABS(DIST)) GO TO 100
C
C       ...CONNECTED, MARK AND REPEAT UNTIL B IS EMPTY.
C
                DO 96 II=1,LP
                IF(L2(II).EQ.D(K,J-1)+LB) GO TO 97
 96             CONTINUE
                LP=LP+1
                L2(LP)=D(K,J-1)+LB
 97             CONTINUE
                IF(LB.LT.B) GO TO   95
 100            CONTINUE
C
```

```
C       ...NOT CONNECTED, CLEAR B REGISTER.
C
                B=0
                GO TO 90
 105            CONTINUE
C
C       Y DIFFERS BY MORE THAN 1 - CHECK SECONDARY LIST FOR OTHER LINES.
C
                IF(LP.LE.0) GO TO 190
                SP=0
 110            CONTINUE
                SP=SP+1
                IF(SP.GT.LP) GO TO 190
C
C       *****
C       ...INITIALIZE SECONDARY LIST.
C       *****
C
                IDP=L2(SP)
                DO 111 KK=1,J
                IF(D(K,KK).NE.IDP) GO TO 111
                GO TO 110
 111            CONTINUE
                J=J+1
                D(K,J)=IDP
                I=IDP
 115            CONTINUE
C
C       ...CHECK FOR NEXT LINE NOT IN OBJECT ALREADY.
C
                IDP=I+1
                IF(IDP.GT.NOLINE) GO TO 119
                DO 116 II=1,J
                IF(D(K,II).NE.IDP) GO TO 116
                I=I+1
                GO TO 115
 116            CONTINUE
C
C       ...CHECK Y VARIATION.
C
                IF(FLD(D,11,L(IDP,1))-FLD(D,11,LL(D(K,J),1))-1)
     .          118,117,119
 117            CONTINUE
C
C       Y DIFFERS BY 1 - COMPUTE X VARYATION.
C
                DIST=FLD(11,11,LL(D(K,J),1))-FLD(11,11,L(IDP,1))
C
C       CHECK IF LINES ARE CONNECTED- IF NOT,NOTE AND CONTINUE.
C
                IF(DIST)123,124,122
 122    CONTINUE
                IF(LENGTH(IDP).GE.ABS(DIST)) GO TO 123
                C=C+1
                I=I+1
                GO TO 115
 123            CONTINUE
                IF(LINE(D(K,J)).LT.ABS(DIST)) GO TO 119
```

```
 124          CONTINUE
C
C       IF CONNECTED, PLACE IN OBJECT AND USE IT AS CURRENT LINE.
C
              I=IDP
              GO TO 70
 118          CONTINUE
              B=B+1
              I=I+1
              GO TO 115
C
C       ...CHANGE CURRENT LINE IF VARIATION NOT 1 CHECK BACKWARDS.
C
 119          CONTINUE
              IDP=L2(SP)
 120          CONTINUE
              IDP=IDP-1
              IF(IDP.LE.0) GO TO 110
              DO 121 II=1,J
              IF(D(K,II).NE.IDP) GO TO 121
              GO TO 120
 121          CONTINUE
C
C       CHECK Y VARIATION.
C
              IF(FLD(D,II,LL(D(K,J),1))-FLD(D,II,LL(IDP,1))-1)
              125,130,110
 125          CONTINUE
C
C       SAME Y - MARK B REGISTER.
C
              B=B+1
C
C       ...COMPUTE X VARIATION - IF CONNECTED, MARK SECONDARY LIST.
C
              DIST=FLD(II,II,LL(D(K,J-1),1))-FLD(II,II,L(IDP,1))
              IF(LINE(D(K,J-1)).LT.ABS(DIST)) GO TO 120
              DO 126 II=1,LP
              IF(L2(II).EQ.IDP) GO TO 120
 126          CONTINUE
              LP=LP+1
              L2(LP)=IDP
              GO TO 120
 130          CONTINUE
C
C       Y DIFFERS BY 1 - COMPUTE X VARIATION.
C
              DIST=FLD(II,II,LL(D(K,J),1))-FLD(II,II,L(IDP,1))
C
C       CHECK IF LINES ARE CONNECTED - IF NOT, NOTE AND CONTINUE
C
              IF(DIST) 140,150,145
 140          CONTINUE
              IF(LINE(D(K,J)).GE.ABS(DIST)) GO TO 150
              C=C+1
              GO TO 120
 145          CONTINUE
              IF(LENGTH(IDP).LT.ABS(DIST)) GO TO 110
```

```
 150          CONTINUE
C
C       IF CONNECTED, PLACE IN OBJECT AND USE IT AS CURRENT LINE.
C
              J=J+1
              D(K,J)=IDP
C
C       CHECK E REGISTER - IF NOT EMPTY, CHECK POSSIBLE CONECTIONS.
C
              IF(E.EQ.0) GO TO 165
              LE=0
 155          CONTINUE
              LE=LE+1
              DIST=FLD(I1,I1,LL(D(K,J),1))-FLD(I1,I1,LL(D(K,J-1)+LE,1))
              IF(LINE(D(K,J)).LT.ABS(DIST)) GO TO 160
C
C       ...CONNECTED, MARK AND CONTINUE UNTIL E IS DELETED.
C
              DO 156 II=1,LP
              IF(L2(II).EQ.D(K,J-1)+LE) GO TO 157
 156          CONTINUE
              LP=LP+1
              L2(LP)=D(K,J-1)+LE
 157          CONTINUE
              IF(LE.LT.E) GO TO   155
 160          CONTINUE
C
C       ...NOT CONNECTED, CLEAR E REGISTER.
C
              E=0
 165          CONTINUE
C
C       PUT C REGISTER IN E REGISTER AND CLEAR C.
C
              E=C
              C=0
C
C       CHECK B REGISTER - IF NOT EMPTY, CHECK POSSIBLE CONNECTIONS.
C
              IF(B.EQ.0) GO TO 120
              LB=0
 170          CONTINUE
              LB=LB+1
              DIST=FLD(I1,I1,LL(D(K,J),1))-FLD(I1,I1,LL(D(K,J-1)-LB,1))
              IF(DIST.LE.0) GO TO 175
              IF(LINE(D(K,J-1)-LB).LT.ABS(DIST)) GO TO 180
 175          CONTINUE
C
C       ...CONNECTED, MARK AND REPEAT UNTIL B IS EMPTY.
C
              DO 176 II=1,LP
              IF(L2(II).EQ.D(K,J-1)-LB) GO TO 177
 176          CONTINUE
              LP=LP+1
              L2(LP)=D(K,J-1)-LB
 177          CONTINUE
              IF(LB.LT.B) GO TO   170
 180          CONTINUE
```

```
C
C       ...NOT CONNECTED, CLEAR B REGISTER.
C
              B=0
              GO TO 120
 185          CONTINUE
C
C       LAST LINE - CHECK SECONDARY LIST, IF NOT MARKED SAVE RESULTS.
C
              IF(LP.GT.0) GO TO 110
 190          CONTINUE
C
C       END OF OBJECT - REINITIALIZE FOR NEXT OBJECT.
C
              NOBJ(K)=J
C
C       ...FIND THE AREA OF THE PRESENT OBJECT.
C
              AREA(K)=0.0
              DO 191 JJ=1,J
              AREA(K)=AREA(K)+LINE(D(K,JJ))
 191          CONTINUE
              IF(AREA(K).GT.158.AND.AREA(K).LT.396) IMP1=1
              IF(AREA(K).GT.395.AND.AREA(K).LT.953) IMP2=1
              IF(AREA(K).GT.952 ) IMP3=1
              K=K+1
              J=1
C
C       ... CLEAR ALL REGISTERS.
C
              B=0
              C=0
              E=0
              LP=0
              SP=0
C
C       CHECK FOR UNUSED LINE TO START NEXT OBJECT( OR TO VERIFY END).
C
              DO 196 II=JK,NOLINE
              DO 195 KK=2,K
              NO=NOBJ(KK-1)
              DO 195 JJ=1,NO
              IF(D(KK-1,JJ).EQ.II) GO TO 196
 195          CONTINUE
              JK=II
              I=II
              D(K,J)=I
              GO TO 40
 196          CONTINUE
C
C       SAVE THE NUMBER OF OBJECTS.
C
              OBJS=K-1
C
C       FIND THE CENTROID OF THE K-TH OBJECT.
C
              DO 210 K=1,OBJS
              II=NOBJ(K)
```

```
      XK=0
      YK=0
      DO 200 I=1,II
      XK=XK+LINE(D(K,I))*(2*FLD(II,II,LL(D(K,I),1))+LINE(D(K,I)
      )-1)/2
      YK=YK+LINE(D(K,I))*FLD(0,II,LL(D(K,I),1))
 200  CONTINUE
      XBAR(K)=XK/AREA(K)
      YBAR(K)=YK/AREA(K)
C
C     FIND THE PEAK AND AVERAGE OF INTENSITY.
C
      IK=0
      IPEAK(K)=0
      DO 205 I=1,II
      JJ=LINE(D(K,I))
      DO 205 J=1,JJ
      LKJ=LL(D(K,I),2)-1+J
      JK=FLD(22,6,S(LKJ))
      IF(IPEAK(K).LT.JK) IPEAK(K)=JK
      IK=IK+JK
 205  CONTINUE
      IAV(K)=IK/AREA(K)
 210  CONTINUE
      WRITE(6,1000)
      DO 211 K=1,OBJS
      AB=AREA(K)
      AC=YBAR(K)
      AD=XBAR(K)
      AG=IPEAK(K)
      AF=IAV(K)
      WRITE(6,1001) K,AB,AC,AD,AG,AF
 211  CONTINUE
C
C     INITIALIZE PARAMETERS.
C
      IF(FRAME.EQ.1)GO TO 230
      J=1
      K=1
 215  CONTINUE
C
C     COMPUTE RADIAL DIFFERENCE.
C
      SA=XBAR(K)-PXBAR(J)
      SB=YBAR(K)-PYBAR(J)
      SR=SQRT(SA**2+SB**2)
C
C     CHECK IF OBJECTS ARE THE SAME.
C
      IF(SR.GT.SH)GO TO 220
C
C     ...SAME,COMPUTE THE DIFFERENCES IN PEAK AND AVERAGE INTENSITIES
C           AND AREA.
C
      DIPEAK(J)=PIPEAK(J)-IPEAK(K)
      DIAV(J)=PIAV(J)-IAV(K)
      DAREA(J)=PAREA(J)-AREA(K)
      GO TO 225
```

```
 220            CONTINUE
C
C       ...NOT SAME,PICK UP NEXT OBJECT.
C
                K=K+1
                IF(K.LE.OBJS+1)GO TO 215
 225            CONTINUE
                K=1
                J=J+1
                IF(J.LE.POBJS+1)GO TO 215
 230            CONTINUE
C
C       UPDATE PREVIOUS OBJECT.
C
                DO 235 K=1,OBJS
                PIPEAK(K)=IPEAK(K)
                PIAV(K)=IAV(K)
                PAREA(K)=AREA(K)
                PXBAR(K)=XBAR(K)
                PYBAR(K)=YBAR(K)
 235            CONTINUE
                PPOBJS=POBJS
                POBJS=OBJS
C
C       PLOT SOLAR FLARES ...
C
C       DRAW SUN.
C
                CALL SUN(1,P,PP)
C
C       DRAW ALL FLARES.
C
                DO 245 K=1,OBJS
                CALL FLARE(K,P,PP,D)
 245            CONTINUE
C
C       DRAW ALL FLARES OF IMPORTANCE 1 IF ANY EXIST.
C
                IF(IMP1.EQ.0)GO TO 255
C
C       ...DRAW SUN.
C
                CALL SUN(2,P,PP)
C
C       ...DRAW FLARES.
C
                DO 250 K=1,OBJS
                IF(AREA(K).GT.158.AND.AREA(K).LT.396) CALL FLARE(K,P,PP,D)
                )
 250            CONTINUE
 255            CONTINUE
C
C       DRAW ALL FLARES OF IMPORTANCE 2 IF ANY EXIST.
C
                IF(IMP2.EQ.0)GO TO 265
C
C       ...DRAW SUN.
C
```

```
                CALL SUN(3,P,PP)
C
C       ...DRAW FLARES.
C
                DO 260 K=1,OBJS
                IF(AREA(K).GT.395.AND.AREA(K).LT.953)CALL FLARE(K,P,PP,D)
 260            CONTINUE
 265            CONTINUE
C
C       DRAW ALL FLARES OF IMPORTANCE 3 IF ANY EXIST
C
                IF(IMP3.EQ.0)GO TO 275
C
C       ...DRAW SUN.
C
                CALL SUN(4,P,PP)
C
C       ...DRAW FLARES.
C
                DO 270 K=1,OBJS
                IF(AREA(K).GT.952) CALL FLARE(K,P,PP,D
     .          )
 270            CONTINUE
 275            CONTINUE
                IMP1=0
                IMP2=0
                IMP3=0
                CALL EOFTV
                GO TO 5
C
C       FORMAT STATEMENTS.
C
 500            FORMAT(/ ' THE NUMBER OF ELEMENTS IS :',I10/' THE NUMBER
     ,OF LINES IS :'I10/)
 1000           FORMAT(/' RESULTS OF FLARE DETECTION:'/T32,'CENTROID',
     .          T50,'INTENSITY'/T5,'OBJECT',T17,'AREA',T29,'Y',T39,'X',
     .          T47,'PEAK',T56,'AVERAGE')
 1001           FORMAT(6I10)
C
C
                SUBROUTINE SUN(I,P,PP)
C
C       THIS SUBROUTINE DRAWS THE OUTLINE OF THE SUN ON EACH FRAME.
C
C       WHEN I=1,'CALL CAMRAV(935)'IS ALSO CALLED
C
                REAL P,PP,R
                DIMENSION P(NEL),PP(NEL),TOPTL(8)/
     .          '          SOLAR FLARES OF IMPORTANCE                '/
                IF(I.EQ.1) CALL CAMRAV(935)
C
C       BUILD SUN ARRAY.
C
                R=500.
                DO 240 K=0,900
                P(K+1)=R*SIN(.007*K)+512.
                PP(K+1)=500.-R*COS(.007*K)
 240            CONTINUE
```

```
      TOPTL(7)=' -ALL-'
      IF(I.EQ.2)TOPTL(7)=' ONE   '
      IF(I.EQ.3)TOPTL(7)=' TWO   '
      IF(I.EQ.4)TOPTL(7)=' THREE'
C
C     PLOT SUN.
C
      CALL SCPLOT(P,PP,901,1,0,0,2,1,2,42,20.,20.,1,1,1.,1024.,
     .1,1000.,0.,TOPTL,
     .'                    Y-GRID COORDINATES                  ',
     .'                    X-GRID COORDINATES                  ')
C
      RETURN
C
C
C
      SUBROUTINE FLARE(I,P,PP,D)
C
C     THIS SUBROUTINE BUILDS AND DRAWS ONE SOLAR FLARE PER CALL.
C
      REAL P(NEL),PP(NEL),XI,YI,DI
      INTEGER D(NOB,LPERO)
      DIMENSION LENGTH(NLI),L(NLI,2)
      COMMON LENGTH,L
      JK=NOBJ(I)
      JI=1
      DO 10 J=1,JK
      IJ=LINE(D(I,J))-1
      IY=FLD(0,11,LL(D(I,J),1))
      IX=FLD(11,11,LL(D(I,J),1))
      IF(J.EQ.1) XI=IX+IJ+5
      IF(J.EQ.1)YI=IY
      DO 5 JL=0,IJ
      P(JL+JI)=IX+JL
      PP(JL+JI)=IY
5     CONTINUE
      JI=JI+IJ+1
10    CONTINUE
      JI=JI-1
      NL=2
      IF(JI.EQ.1) NL=1
      CALL SCPLOT(P,PP,JI,1,0,0,NL,1,1,42,20.,20.,2)
      DI=1
      IX=NXV(XI)
      IY=NYV(YI)
      CALL LABLV(DI,IX,IY,4,2,3)
C
      RETURN
C
      END
```

BELLCOMM, INC.

Subject: Computer Program Abstract and Write-up for SFLARE - Solar Flare Detection - Case 103-6

From: R. R. Singers

Distribution List

Bellcomm, Inc.

R. K. Agarwal
G. R. Andersen
A. P. Boysen, Jr.
K. R. Carpenter
D. A. Chisholm
D. R. Hagner
H. A. Helm
B. T. Howard
P. F. Long
J. Z. Menard
J. M. Nervik
I. M. Ross
P. S. Schaenman
J. W. Timko
R. L. Wagner
Central Files
Department 1024 Files
Library