NATIONAL RADIO ASTRONOMY OBSERVATORY

COMPUTER DIVISION

INTERNAL REPORT #16
February, 1974

THE COMPUTATION OF TWO-DIMENSIONAL CONVOLUTION

BY

S. HUANG

TABLE OF CONTENTS

I. INTRODUCTION

CONVOLU is a program, written in FORTRAN, which performs a two-dimensional convolution between two sets of real data (see Section V for Complex Data). For example, beam and map. The FFT technique is used in order to speed up the computation.

The FFT program used here is assuming that the data block is a N x N even square matrix, where N = power of 2, with the zero-phase at the center of the data block. Therefore, if the size of data block is smaller than N x N, the program will fill in zeroes around the data block to make it a N x N matrix. Figure 1 shows the reference zero is at the center (X) of a 4 x 4 data set block.

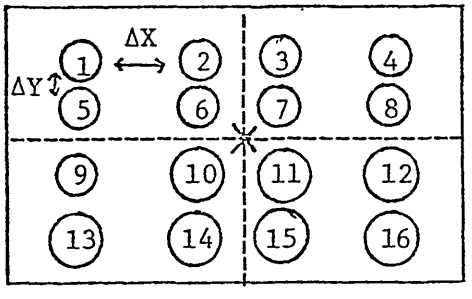

Figure 1
Data Block

Therefore, assuming ΔX, ΔY are the intervals between each point along the X & Y axes (the dash lines), the coordinates for points 6, 7, 10, and 11 are $(-\frac{1}{2}\Delta X, \frac{1}{2}\Delta Y)$, $(\frac{1}{2}\Delta X, \frac{1}{2}\Delta Y)$, $(-\frac{1}{2}\Delta X, -\frac{1}{2}\Delta Y)$ and $(\frac{1}{2}\Delta X, -\frac{1}{2}\Delta Y)$ respectively.

Because of this special arrangement for the data in the FFT technique, the user has to arrange his data block in such way, even number of data points on each quadrant of the data block with the zero at the center, before the FFT can be used. Figure 2 shows the form of the data block supplied by the user. Figure 3 shows how the program filled in zeroes around the data block.

1	1	2	3
2	1	3	1
1	0	1	1
2	1	2	1

Figure 2

Data Block Given By The User

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1	1	2	3	0	0
0	0	2	1	3	1	0	0
0	0	1	0	1	1	0	0
0	0	2	1	2	1	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 3

Zeroes Filled In By The Program

II. PROGRAM CONSIDERATIONS

The Program flow is very easy to follow and understand. The main program called several subroutines. But only two subroutines are concerned by the user.

1. GETBE: GETBE is to get the beam data. It could be read in by cards, one row at one time, or called MODELB to supply the data. The input card in section III will show you how to supply the data.
2. GETMAP: GETMAP is to get the map data. They are being treated as same as GETBE with options of getting the data from the cards or by calling MODELM.

Assuming the beam array has the size of (M0, N0), where M0, N0 $\leq$ M, N and the map array has the size of (M2, N2), where M2, N2 $\leq$ M, N then M, N = power of 2, also

= largest of (M0, N0) & (M2, N2).

Suppose (M0, N0) = (10, 14) & (M2, N2) = (12, 38). Then we will choose (M, N) = (16, 64) as it will cover (M0, N0) and (M2, N2). However, since the FFT package is written in square matrix, so we will choose (M, N) = (64, 64).

III. INPUT AND OUTPUT

The input sequence is as follows:

	Column	Format	Variable	Identification
Card 1	1-10	I10	M	Row size of convolution array, power of 2.
	11-20	I10	N	Column size of convolution array, power of 2. Normally, M=N.
Card 2	1-10	I10	Mo	Row size for the beam, even no.
	11-20	I10	No	Column size for the beam, even no
	21-30	I10	IB	Indication of beam array input. If=1, read in by cards. If=other no., call MODELB.
Block of Card 3	1-10	F10.0	BEAM(1)	Beam array from the cards. They are read in one row at a time. Always starting a new row with a new card. There are 8 values on each card.
	11-20	F10.0	BEAM(2)	
	-	-	-	
	-	-	-	
	71-80	F10.0	BEAM(8)	
Card 4	1-10	I10	M2	Row size for map, even no.
	11-20	I10	N2	Column size for map, even no.
	21-30	I10	IM	=1, read in map array =other no., call MODELM.
Block of Card 5	1-10	F10.0	MAP(1)	Map array from the cards.
	11-20	F10.0	MAP(2)	They are treated same as the beam array.
	-	-	-	
	-	-	-	
	71-80	F10.0	MAP(8)	
Card 6	1-10	I10	ROW1	Starting row for printout.
	11-20	I10	ROW2	Ending row for printout.
	21-30	I10	COL1	Starting Column for printout.
	31-40	I10	COL2	Ending Column for printout.

The output now is to print out the values of the convolution row by row with the size specified by (ROW1, ROW2) and (COL1, COL2).

The nos. ROW1, ROW2, COL1 and COL2 are read in from Card 6. For the example in section II, we could specify ROW1 = 18, ROW2 = 48 and COL1 = 7, COL2 = 56 which will printout the center block of convolution values and presumably will cover the significant parts of the convolution. This is done to reduce the enormous amount of output when M and N get too big, like 128 and 256.

IV. EXAMPLE OF MODELB AND MODELM

```
      Call MODELB (MO, NO, BEAM, J)
C   MO, NO is the size of the beam.
C   BEAM is the beam data for each row, they have No numbers of
          real data.
C   J is the Jth row of the beam data

      Call MODELM (M2, N2, MAPDAT, J)
C   M2, N2 are the size of the map.
C   MAPDAT is the size of N2, each row of the map data.  They are
          real numbers.
C   J is the Jth row of the map data.
```

V. PROGRAM RESTRICTION AND MODIFICATION

The program is listed in the Appendix I. Here 256 x 256 is used as the maximum size for both beam and map. The user can alter this to suit his desires, but we found that this is not necessary in the real practice. The input and output of beam and map here

are assumed the real data. However, the FFT package is applicable for both real and complex data. Therefore, the user can slightly modify the input and output subroutines (GETBE, GETMAP, DISPLY) to input the complex data and output the complex convolution values. The real and imaginary parts of the complex no. is stored adjacent to each other. It should be very easy to modify in the above subroutines.

APPENDIX I

Program Listing

```
FORTRAN IV G LEVEL  20                  MAIN                DATE = 74050        18/28/16              PAGE 0001

0001                DEFINE FILE 11(256,512,U,ID1), 13(256,512,U,ID3)
0002                COMPLEX Z0(256), ZI (256)
0003                INTEGER F11, F13
0004                COMMON  Z0,ZI, F11,F13
0005                F11=11
0006                F13=13
0007                READ (5,5) M,N
0008                CALL GETBE ( M,N)
0009                CALL FILLW (N, 1)
0010                CALL FFT2 ( N,F11,1)
0011                CALL GETMAP(M,N)
0012                CALL FFT2 (N,F13,1)
0013                CALL XFFT(M,N)
0014                CALL FILLW (N, 2)
0015                CALL FFT2 ( N,F13, 2
0016                CALL DISPLY (M,N)
0017              5 FORMAT(2I10)
0018                END
```

```
FORTRAN IV G LEVEL  20                  GETBE               DATE = 74050        18/28/16              PAGE 0001

0001                SUBROUTINE GETBE ( M, N)
                  C---------------
                  C   IB=1       READ IN DATA BY CARDS
                  C      =OTHER  CALL MODELB TO GET THE DATA
                  C---------------
0002                COMPLEX Z0 (256)
0003                INTEGER F11, F13
0004                REAL BEAM(256), DUMMY(256)
0005                COMMON  Z0, BEAM, DUMMY ,F11,F13
0006                READ (5,5) MO,NO,IB
0007                JUMP= (N-MO)/2
0008                WRITE(6,998)
0009                DO 100 J=1, MO
0010                IF (IB .EQ. 1) GO TO 50
0011                CALL MODELB ( MO, NO, BEAM, J)
0012                GO TO 60
0013              50 READ (5, 10) (BEAM(I), I=1, NO)
0014              60 WRITE(6, 999) (BEAM (L), L=1, NO)
0015                CALL FILLO ( BEAM, MO,NO,M,N, ZO)
0016                WRITE(F11' JUMP+J) (ZO(I), I=1, N)
0017             100 CONTINUE
0018                CALL FILLOR ( MO, NO, M,N , F11)
0019             998 FORMAT(' ', / 2X, 'BEAM ARRAY:', /)
0020             999 FORMAT(' ', 10E12.5)
0021               5 FORMAT(8I10)
0022              10 FORMAT (8F10.0)
0023                RETURN
0024                END
```

```
FORTRAN IV G LEVEL  20                  MODELB          DATE = 74050        18/28/16            PAGE 0001

0001                SUBROUTINE MODELB ( MO, NO, BEAM, J)
0002                REAL BEAM (256)
0003                REAL X,Y,AN,BN
0004                X= -( MO /2 * 3- 1.5)
0005                Y= -( MO /2 *4. -2.)
0006                AN= 3.
0007                BN= 4.
0008                DO 30 K=1, NO
0009             30 BEAM (K)= 20.* EXP ( -2.772* ((X+(K-1)*AN)**2/121.+ (Y+ (J-1)*BN
                   C    )**2/400.)     )
0010                RETURN
0011                END
```

```
FORTRAN IV G LEVEL  20                  GETMAP          DATE = 74050        18/28/16            PAGE 0001

0001                SUBROUTINE GETMAP (M,N)
                   C----------
                   C    IM=1       READ IN DATA FROM THE CARDS
                   C       =OTHER  CALL MODELM TO GET THE DATA
                   C----------
0002                COMPLEX ZO (256)
0003                REAL MAPDAT(256), DUMMY (256)
0004                INTEGER F11, F13
0005                COMMON ZO, MAPDAT, DUMMY, F11,F13
0006                READ (5,5) M2,N2,IM
0007                JUMP= (M-M2)/2
0008                WRITE(6, 998)
0009                DO 100 J=1, M2
0010                IF ( IM .EQ. 1) GO TO 50
0011                CALL MODELM ( M2,N2, MAPDAT, J)
0012                GO TO 60
0013             50 READ (5, 10) (MAPDAT (I), I=1, M2)
0014             60 WRITE(6, 999) (MAPDAT(I), I=1, N2)
0015                CALL FILLO ( MAPDAT, M2,N2, M,N,ZO)
0016                WRITE(F13' JUMP+J) ( ZO (L), L=1,N)
0017            100 CONTINUE
0018                CALL FILLOR ( M2,N2, M,N, F13)
0019             5  FORMAT(8I10)
0020             10 FORMAT(8F10.0)
0021            998 FORMAT(' ', /, 2X 'MAP ARRAY: ', /)
0022            999 FORMAT(' ', 10F12.5)
0023                RETURN
0024                END
```

```
FORTRAN IV G LEVEL  20                  MODELM          DATE = 74050        18/28/16            PAGE 0001

0001              SUBROUTINE MODELM ( M2,N2, MAPDAT,J)
0002              REAL MAPDAT (256)
0003              REAL  XO,YO,SX,SY
0004              XO=  13.5
0005              YO= 6.5
0006              SX=  10.
0007              SY= 5
0008              DO 100 I=1, N2
0009            100 MAPDAT(I)= 35* EXP (-2.772*(((I-XO)/SX)**2 + ((J-YO)/SY)**2))
0010              RETURN
0011              END
```

```
FORTRAN IV G LEVEL  20                  XFFT            DATE = 74050        18/28/16            PAGE 0001

0001              SUBROUTINE XFFT ( M,N)
0002              COMPLEX ZO(256), ZI(256)
0003              INTEGER F11, F13
0004              COMMON ZO, ZI, F11, F13
0005              DO 100 J=1 ,M
0006              READ (F11'J) ( ZI ( L), L=1,N)
0007              READ (F13'J) ( ZO ( L), L=1,N)
0008              DO 50 L=1,N
0009            50  ZO(L)= ZO(L)* ZI(L) /N/N
0010              WRITE(F13'J) ( ZO ( L), L=1,N)
0011            100 CONTINUE
0012              RETURN
0013              END
```

```
FORTRAN IV G LEVEL  20                  FILLOR          DATE = 74050        18/28/16            PAGE 0001

0001              SUBROUTINE FILLOR ( MO, NO, M, N , IFILE)
0002              COMPLEX ZO(256), ZI (256)
0003              INTEGER F11,F13
0004              COMMON ZO, ZI, F11, F13
0005              DO 10 I=1, N
0006            10  ZO(I)=0.
0007              JUMP= (N-MO)/ 2
0008              NBRE1= JUMP + MO
0009              DO 50 I=1,JUMP
0010              WRITE(IFILE' I) (ZO(K), K=1,N)
0011              WRITE(IFILE' NBRE1+ I) (ZO(L),L=1,N)
0012            50  CONTINUE
0013              RETURN
0014              END
```

```
FORTRAN IV G LEVEL  20                  FILLO              DATE = 74071          10/20/

0001                SUBROUTINE FILLO  ( ZI, MO,NO,M,N, ZO)
0002                REAL ZI(256)
0003                COMPLEX ZO (256)
0004                NZERO= (N-NO)/2
0005                DO 20 J=1, NZERO
0006                ZO(J)=0.
0007             20 ZO((N+ NO)/2+J)=0.
0008                MZERO= NZERO+1
0009                KZERO= NZERO+NO
0010                K=1
0011                DO 30 J=MZERO, KZERO
0012                ZO (J)= ZI
0013                K= K+1
0014             30 CONTINUE
0015                RETURN
0016                END

FORTRAN IV G LEVEL  20                  DISPLY             DATE = 74071          10/20/18              PAGE 0001

0001                SUBROUTINE DISPLY (M,N; ROW1,ROW2,COL1,COL2)
0002                COMPLEX ZO(256)
0003                INTEGER ROW1,ROW2,COL1,COL2
0004                INTEGER F11,F13
0005                REAL RMAP(256), DUMMY(256)
0006                COMMON ZO, RMAP, DUMMY, F11,F13
0007                WRITE(6, 998)
0008                READ(5,.995) ROW1,ROW2,COL1,COL2
0009                DO 100 I=ROW1,ROW2
0010                READ ( F13' I) (ZO(L), L=1, N)
0011                DO 50 J=COL1,COL2
0012             50 RMAP (J)= ZO (J)
0013                WRITE(6,997) I
0014                WRITE(6,999) (RMAP(L), L=COL1,COL2)
0015            100 CONTINUE
0016            995 FORMAT(4I10)
0017            997 FORMAT(' ','ROW', I5)
0018            998 FORMAT(//,' ', 'THE CONVOLUTION OF THE BEAM AND  MAP: './/)
0019            999 FORMAT(' ', 10F12.5)
0020                RETURN
0021                END

FORTRAN IV G LEVEL  20                  FILLW              DATE = 74071          10/20/18              PAGE 0001

0001                SUBROUTINE FILLW (N, IF )
0002                INTEGER N, N5
0003                COMPLEX W1(128),W2(128),W3(128), WX,WY,WW
0004                COMMON/WEIGHT/ W1,W2,W3
0005                SS= -6.2831853/N
0006                IF ( IF .EQ.  2 ) SS= -SS
0007                N5= N/2
0008                DO 20 I=1,N5
0009                ANG1= I*SS
0010                ANG2= (I*2-1)*SS/4
0011                W1(I)= CMPLX (COS(ANG1), SIN(ANG1))
0012                W2(I)= CMPLX (COS(ANG2), SIN(ANG2))
0013                W3(I)= CMPLX (COS(ANG2),-SIN(ANG2))
0014             20 CONTINUE
```

```
0001            SUBROUTINE FFT2(N, FILE , IF)
                C-------
                C    THIS SUBROUTINE IS TO PERFORM A N X N FAST FOURIER
                C     TRANSFORM,WHERE N IS THE POWER OF TWO
                C     THE ZERO-PHASE IS AT THE CENTER
                C    COMPLEX INPUT DATA IS USED
                C-------
0002            INTEGER N, FILE
0003            INTEGER*2  BITREV(256) , N1, N2,N3,N4,N5,N6
0004            INTEGER*2  II,IJ, IF1,IF2, IP,IQ,IR,IK, KK
0005            COMPLEX                 Z1(256),Z2(256),     WX,WY, WW,WU
0006            COMPLEX W1(128), W2(128),W3(128)
0007            COMPLEX Z(512)
0008            COMMON  Z
0009            COMMON/WEIGHT/ W1,W2,W3
0010            EQUIVALENCE (Z(1), Z1(1)), (Z(257), Z2(1))
0011            N1= N/2
0012            N4= N/4
                C ------
                C    THE RIGHTMOST MATRIX
                C ------
0013            N5= N1+1
0014            DO 40 J=1, N1
0015            READ (FILE' J) Z1
0016            READ (FILE' J+N1) Z2
0017            N2= N5-J
0018            DO 50 I=1,N1
0019            N6= I+N1
0020            N3= N5-I
0021            WX=  Z1(I)*W3(N3)+ Z1(N6)*W2(I)
0022            WY= Z1(I)*W2(N3)+ Z1(N6)*W3(I)
0023            WW= Z2(I)*W3(N3)+  Z2(N6)* W2(I)
0024            WU= Z2(I)*W2(N3)+ Z2(N6)* W3(I)
0025            Z1(I)= WX* W3(N2)+ WW* W2(J)
0026            Z1(N6)= WY* W3(N2)+ WU* W2(J)
0027            Z2(I)= WX* W2(N2)+ WW* W3(J)
0028            Z2(N6)= WY* W2(N2)+ WU* W3(J)
0029      50    CONTINUE
0030            WRITE(FILE' J+N1) Z2
0031            WRITE(FILE' J) Z1
0032      40    CONTINUE
                C-------
                C   THE CENTER  MATRICES
                C-------
0033            IF ( N .LE. 4) GO TO 122
0034            N2=N4
0035            I=1
0036            N3= N+1
0037      60    CONTINUE
0038            II=2**(I-1)
0039            IJ=II*2
0040            DO 130 ITT=1,2
0041            DO 130 IP=1,II
0042            KK= II
0043            DO 130 IP=1,N2
0044            IF1= (IP-1)*N2*2+IR
0045            IF2= IF1+N2
0046            IF ( ITT .EQ. 2) IF1= N3-IF1
```

```
0047              IF ( ITT .EQ. 2) IF2= N3-IF2
0048           65 READ (FILE' IF1)        Z1
0049              READ (FILE' IF2)        Z2
0050              DO 80  IQ=1,II
0051              IK=II
0052              DO 80 II=1,N2
0053              N5= (IQ-1)*N2*2+IL
0054              N6= N5+N2
0055              DO 70 ITER=1,2
0056              WX= Z1(N5)+ Z1(N6)
0057              WY= ( Z1(N5)- Z1(N6))* W1( IK)
0058              WW= Z2(N5)+ Z2(N6)
0059              WU= ( Z2(N5)- Z2(N6))* W1(IK)
0060              Z1(N5)= WX+ WW
0061              Z1(N6)= WY+ WU
0062              Z2(N5)= ( WX-WW) *W1(KK)
0063              Z2(N6)= (WY-WU) * W1(KK)
0064              N5= N3-N5
0065              N6= N3-N6
0066           70 CONTINUE
0067              IK= IK+IJ
0068           80 CONTINUE
0069              WRITE (FILE' IF2)  Z2
0070              WRITE (FILE' IF1)  Z1
0071              KK= KK+IJ
0072           130 CONTINUE
0073              N2= N2/2
0074              I=I+1
0075              IF (  N2 .NE. 1) GO TO 60
                C----------
                C     THE LEFTMOST MATRIX
                C----------
0076           122 CONTINUE
0077              DO 150 ITT=1,2
0078              DO 150 I=1,N4
0079              IF2= I*2
0080              IF1= IF2-1
0081              IF ( ITT .EQ. 2) IF1= N3-IF1
0082              IF ( ITT .EQ. 2) IF2= N3-IF2
0083              READ( FILE' IF1) Z1
0084              READ ( FILE'IF2) Z2
0085              DO 125 J=1,N4
0086              IJ= J*2
0087              II=IJ-1
0088              DO 125 ITER=1,2
0089              WX= Z1(II)+ Z1(IJ)
0090              WY= Z2(II)+ Z2(IJ)
0091              WW= (Z1(IJ)- Z1(II))* W1(N4)
0092              WU=(Z2(IJ)-Z2(II))* W1(N4)
0093              Z1(II)= (WU-WW)* W1(N4)
0094              Z1(IJ)= ( WY-WX)* W1(N4)
0095              Z2(II)= WW+ WU
0096              Z2(IJ)= WX+ WY
0097              II= N3-II
0098              IJ= N3-IJ
0099           125 CONTINUE
0100              WRITE(FILE' IF2) Z2
0101              WRITE(FILE' IF1) Z1
```

```
0102          150 CONTINUE
             C ------
             C   BIT REVERSING
             C ------
0103              BITREV(1)=0
0104              N2= N
0105          155 II=N/N2
0106              N2= N2/ 2
0107              DO 160 I=1,II
0108          160 BITREV(I+II)= BITREV(I)+N2
0109              IF ( N2 .NE.1) GO TO  155
0110              DO 200 I=1,N
0111              IF ( BITREV(I) .LT. I-1) GO TO 20(
0112              READ(FILE'I) Z1
0113              IF ( BITREV(I) .EQ.I-1) GO TO 165
0114              READ(FILE' BITREV(I)+1)  Z2
0115          165 DO 170 J=1,N
0116              IF ( BITREV(J) .LE. J-1) GO TO 17(
0117              WX= Z1(J)
0118              WY= Z2(J)
0119              IJ= BITREV(J)+1
0120              Z1(J)= Z1(IJ)
0121              Z2(J)= Z2(IJ)
0122              Z1(IJ)= WX
0123              Z2(IJ)= WY
0124          170 CONTINUE
0125              IF ( BITREV(I) .EQ. I-1) GO TO 190
0126              WRITE( FILE'BITREV(I)+1)  Z1
0127              WRITE(FILE'I) Z2
0128              GO TO 200
0129          190 WRITE(FILE' I) Z1
0130          200 CONTINUE
0131              RETURN
0132              END
```