

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
INFORMATION AS POSSIBLE

SPECTRAL TRANSFORMATIONS IN THE "SOFI" COMPLEX FOR
PROCESSING PHOTOGRAPHIC IMAGES ON THE ES COMPUTER, PART I

A. S. Debabov and D. A. Usikov

(NASA-TM-75632-Pt-1) SPECTRAL
TRANSFORMATION IN THE SOFI COMPLEX FOR
PROCESSING PHOTOGRAPHIC IMAGES ON THE ES
COMPUTER, PART 1 (National Aeronautics and
Space Administration) 22 p HC A02/MF A01

N82-18556

Unclas
G3/35 07976

Translation of: "Spektral'nyye preobrazovaniya v
komplekse obrabotki fotograficheskikh isobrazhenii
"SOFI" na ES-EVM" Chast' I, Academy of Sciences
USSR, Institute of Space Research, Moscow, Report
Pr-394, 1978, pp. 1-20.



NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D. C. 20546
JUNE 1979

STANDARD TITLE PAGE

1. Report No. NASA TM-75632	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle SPECTRAL TRANSFORMATIONS IN THE "SOFI" COMPLEX FOR PROCESSING PHOTO- GRAPHIC IMAGES ON THE ES COMPUTER. PART I		5. Report Date JUNE 1979	6. Performing Organization Code
		8. Performing Organization Report No.	10. Work Unit No.
7. Author(s) A.S. Debabov and D.A. Usikov		11. Contract or Grant No. NASw-3198	
9. Performing Organization Name and Address SCITRAN Box 5456 Santa Barbara, CA 93108		13. Type of Report and Period Covered Translation	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code	
15. Supplementary Notes Translation of "Spektral'nyye preobrazovaniya v komplekse obrabotki fotograficheskiky isobrazhenii 'SOFI' na ES-EVM". Chast' I, Academy of Sciences USSR, Institute of Space Research, Moscow, Report Pr-394, 1978, pp 1 - 20			
16. Abstract A description is given of three programs catalogued in the form of object modules in the library of a system for processing photographic images on the "SOFI" ES-1040 computer at the Institute of Space Research of the USSR Academy of Sciences: PFT is the subprogram of the multi-dimensional BPF of real-valued information, in the operative computer memory; INRECO - subprogram-interface between the real and complex formats for representing two-dimensional spectra and images; FFT2 - subprogram for calculating the correlation functions of the image using the previous subprograms.			
17. Key Words (Selected by Author(s))		18. Distribution Statement Unclassified - Unlimited	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 22	22. Price

SPECTRAL TRANSFORMATIONS IN THE "SOFI" COMPLEX FOR
PROCESSING PHOTOGRAPHIC IMAGES ON THE ES COMPUTER. PART I

A. S. Debabov and D. A. Usikov

INTRODUCTION

13*

The study [1] determined the tasks and principles for organizing a system of mathematical processing of photographic images (the "SOFI" complex). Spectral transformations play no small role in the programming support for the complex. Since the information is inputted to the computer in discrete form, we shall be speaking of discrete transformations (DT).

The DT of the numerical file $C(NNN)$ represents a series of linear transformations of the subfiles $f^{(N)}$, selected from the file according to a definite rule:

$$f_k^{(N)} = C(K), \quad k=0, 1, \dots, N-1; \quad 1 \leq K \leq NNN, \quad (1)$$

where K is an integral function of k ; N -- DT base. Drawing an analogy between the subfile and the vector column in N -dimensional space, we may write:

$$f^{(N)} = \beta^{(N)} \prod_{l,k} f_k^{(N)} = \beta^{(N)} \sum_{k=0}^{N-1} a_{k,l} f_k^{(N)}, \quad l=0, 1, \dots, N-1. \quad (2)$$

Mark $^{\circ}$ over the symbol of the subfile means that its $^{\circ}$ -transforms, i.e. linear combinations of the subfile elements, occupy the position of the subfile elements in the computer memory.

Π and a -- symbols of the DT matrix of the dimensionality N

* Numbers in margin indicate pagination of original foreign text.

x N and its elements; β -- normed factor. In the general case to implement DT (2), we require $\sim N^2$ operations of addition-multiplication (\circ), and a memory for $\sim N^2$ numbers -- elements of the DT matrix.

Since 1965, methods have been vigorously developed for the rapid transformations (RT) [2,3,4,5], based on the possibility of representing the DT matrices $\Pi^{(N)}$ in the form of Kronecker products of the slightly filled matrices $Q^{(N,i)}$ ($i = 1, 2, \dots, m$). In this case, (2) is transformed into the DT series

$$f_i^{(N)} = \beta^{(N,i)} Q_{\rho, \kappa}^{(N, m+1-i)} f_{\kappa}^{(N)}, \quad (3)$$

where $f_{\kappa}^{(N)} = f_{\kappa}^{(N)}$, $f_{\rho}^{(N)} = f_{\rho}^{(N)}$.

If $Q^{(N,i)} = L_i$ - are block-diagonal matrices, the number of operations (\circ) and the volume of additional memory are reduced by a factor of γ_{rt} , where

$$\gamma_{rt} = N / \sum_{i=1}^m L_i. \quad (4)$$

"Acceleration" of the DT on a computer is achieved by:

a) finding the factorization method $\Pi^{(N)} = Q^{(N,1)} \dots Q^{(N,m)}$, for which γ_{rt} (formula (4)) is maximum;

b) constructing the optimum algorithm for the DT series (formula (3)) with allowance for the layout of the subfile in the computer memory (formula (1));

c) optimum translation of the algorithm into a system of actual computer commands;

d) technical improvements in the computer: special commands, special processors, etc.

Algorithms and programs for one-dimensional Fourier RT (RTF) are the most widely used at the present time. The module of the multidimensional RTF described below is "more rapid" than its predecessors in all points; with respect to (d) we have in mind the advantages of the EC computer as a third generation machine; with respect to a), b) -- the advantages of the RTF are a real-valued function [6], extraction of a normed multiplier, inversion, equal multipliers, exception of transposition (A. S. Debabov), the extraction of 0.1 multipliers, and recurrent computations of trigonometric functions (N. I. Kalinin); with respect to c) -- programming on ASSEMBLER of cyclical DT procedures, the selection of optimum command sequences based on results of tests for rapid operation (A. S. Debabov). The DTF of a file of $\sim 10^5$ numbers on one or several measurements on the EC-1040 computer is performed in a time of approximately 10 seconds. /5

For several applications, the Fourier transforms of a real-valued file should be presented in the form of a complex file, because from the latter we must extract the purely real and the purely imaginary transforms, etc. For a two-dimensional file, a special subprogram performs similar transformations, which is described below. The third subprogram makes it possible to calculate the image correlation functions with the help of the first two. The example given here may be regarded as the text of the module described.

I. STRUCTURE OF INFORMATION IN THE COMPUTER MEMORY

In accordance with the principles for organizing the

"SOFI" complex [1], the exchange of information between the user programs and the DT modules is performed by using the unlabelled general region of the form

COMMON FICT(394),NC,N,NN,NA,NK, C(NNN)... , (5)

where FICT is the region of the control program [1]; ~~NC=NK~~ -- the control parameters of the DT modules described and the auxiliary routines; NNN -- total dimensions of the file of numbers of the REAL * 4 type.

During the operation of the user program phase with the DT modules, requiring a memory of Z_n K bytes, and Z_p K bytes, $NNN < 256(Z_p - Z_n)$. Thus, if $Z_p = 256K$, and $Z_n < 16K$, the user may write up to 15 files of the form A2(64,64), or up to 3 files of B2(128,128), or a three-dimensional file C3(32,32,32), etc. in the operator (5) producing the program in the memory. /6
Some files may be first written as complex lengths of 8, for example, CMP(128,128), which is equivalent to the real file description C3(2,128,128).

In the DT subroutine, the file C(NNN) may be written with its first element C(I) and the rules (I) for selecting the subfiles, which are determined by the values of the parameters NC-NK. As an example, we give the rule for selecting the subfiles on three measurements of the file C3(1,2,3), which are in operation in the FPT modulus at $|NC| = I$: the parameters NA and NK determine the boundaries of C3 in the file field C(NNN): $C3(I,I,I)=C(NA), C3(NI,N2,N3)=C(NK)$; the parameter N determines the base of the first measurement (row of two-dimensional subfiles): $NI = N$, with the selection of the k-th element of the i-th subfile of the first measurement ($0 \leq K < N$), ($0 \leq N2 \times N3$) (I) has the form: $K = NA + K + i \cdot N$: the parameter NN determines the base of the two-dimensional subfiles first and second

measurements: $N_2 = NN/N$, (1) for the k -th element of the j -th subfile of the second measurement ($0 \leq k < N_2$), ($0 \leq j < N$), selected from [1] of the l -th two-dimensional subfile such that:
 $k = N_2 l + k_0 + j + l \cdot NN$, where $0 \leq l < N$; the base of the third measurement (numbers of the two-dimensional subfiles $N_3 = (N_2 + 1 - N_2)/NN$, and the rule (1): $k = N_3 l + k_0 + j + m$, where $0 \leq m < NN$.

The transforms of the subfiles of the corresponding measurements are selected according to the same rules as the subfiles until DT is performed.

In the INRECO module, the elements of the real two-dimensional subfile C_2 are selected according to the rules;
 $C_2(i, j) = C(N_2 i + j, N)$, and the corresponding elements of the complex subfile $C_{CP}(i, j) = C(U:K + 2^e i, 2^e j + N + \theta)$, where $0 \leq i < N$, $0 \leq j < NN/N$, $e = 0$ corresponds to the real part; $e = 1$ -- the imaginary part.

2. RTF ALGORITHM OF A REAL-VALUED FUNCTION

17

The canonical form of the DTF record of the complex function $f_k^{(N)}$ of the discrete whole-number argument k on the base N ($0 \leq k \leq N-1$) [2] is:

$$\hat{p}_{f_c}^{(N)} = \frac{1}{N} \Phi_{e,k}^{(N)} = \frac{1}{N} \sum_{k=0}^{N-1} W_N^{k\ell} f_k^{(N)}, \quad \ell = 0, 1, \dots, N-1, \quad (6)$$

and thus the inverse transformation:

$$f_k^{(N)} = [\Phi_{k,\ell}^{(N)}]^{-1} \hat{p}_{f_c}^{(N)} = \sum_{\ell=0}^{N-1} \overline{W_N^{k\ell}} \hat{p}_{f_c}^{(N)} = \overline{\Phi_{k,\ell}^{(N)}} \hat{p}_{f_c}^{(N)}, \quad (7)$$

where $W_N = \exp(i2\pi k/N) = \cos(2\pi k/N) + i \sin(2\pi k/N) = (C_N, S_N)$, $W_N^* = (C_N, -S_N)$,

the value of the upper index for W_N is the exponent, and for C_N and S_N -- the multiplicity factor of the angle to the base angle $2\pi/N$: $C_N^p = \cos(2\pi p/N)$.

The multidimensional DTF may be presented in the form of a sequence of one-dimensional DTF; for the function of the two arguments $f^{(N_1, N_2)}$, on the base $NN = N_1 + N_2$:

$$\hat{F}_{C_1, C_2}^{(N_1, N_2)} = \frac{1}{NN} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} W_{NN}^{k_1(1+N_2)+k_2(2+N_1)} F_{k_1, k_2}^{(N_1, N_2)} = \frac{1}{NN} \Phi_{C_1, C_2}^{(N_2)} \left[\Phi_{C_1, C_2}^{(N_1)} F_{k_1, k_2}^{(N_1, N_2)} \right], \quad (8)$$

since $W_{NN}^{PN_2} = W_{N_1}^P$.

Substituting $\text{Im } \hat{f}_k^{(N)} = 0$ into (7) and setting $\text{Re } \hat{f}_k^{(N)} = \hat{d}_k^{(N)}$ for odd N , we obtain (and we set):

$$\begin{aligned} \text{Im } \hat{f}_0^{(N)} = \text{Im } \hat{f}_{N/2}^{(N)} = 0, \quad \text{Re } \hat{f}_0^{(N)} = \hat{d}_0^{(N)}, \quad \text{Re } \hat{f}_{N/2}^{(N)} = \hat{d}_{N/2}^{(N)}, \\ \text{Re } \hat{f}_l^{(N)} = \text{Re } \hat{f}_{N-l}^{(N)} = \hat{d}_l^{(N)}, \quad \text{Im } \hat{f}_l^{(N)} = -\text{Im } \hat{f}_{N-l}^{(N)} = -\hat{d}_{N-l}^{(N)}, \\ (l = 1, 2, \dots, N/2 - 1). \end{aligned} \quad (9)$$

In this notation, the conversion formula takes the form:

$$\hat{d}_k^{(N)} = [D_{k, l}^{(N)}]^{-1} \hat{d}_l^{(N)} = \hat{d}_0^{(N)} + (-1)^k \hat{d}_{N/2}^{(N)} + 2 \sum_{l=1}^{N/2-1} [\hat{d}_l^{(N)} C_{N-l}^{kN} + \hat{d}_{N-l}^{(N)} S_{N-l}^{kN}] \quad (10)$$

i.e., the cosine and sine coefficients of the Fourier series may be written toward to each other. /8

We may obtain the algorithm of the rapid reverse DTF for $N = 2^M$ by comparing the expansion of (10) for odd and even

elements of the subfile $\hat{b}^{(2n)}$ with the expansion (10) of auxiliary subfiles $\hat{b}^{(n)}$ and $\hat{b}^{(n)}$, selected according to the rule:

$$\hat{b}_i^{(n)} \leftrightarrow \hat{b}_{2i}^{(2n)} \quad , \quad \hat{b}_i^{(n)} \leftrightarrow \hat{b}_{2i+1}^{(2n)} \quad , \quad (i=0, 1, \dots, n-1).$$

For example, for even elements for all $i=0, 1, \dots, n-1$:

$$\hat{b}_i^{(n)} = \hat{b}_0^{(n)} + (-1)^i \hat{b}_{n/2}^{(n)} + 2 \sum_{j=1}^{n/2-1} [\hat{b}_j^{(n)} C_n^{ij} + \hat{b}_{n-j}^{(n)} S_n^{ij}] ;$$

$$\hat{b}_{2i}^{(n)} = \hat{b}_0^{(n)} + \hat{b}_n^{(n)} + 2 \left\{ (-1)^i \hat{b}_{n/2}^{(n)} + \sum_{j=1}^{n/2-1} [\hat{b}_j^{(n)} C_{2n}^{2ij} + \hat{b}_{n-j}^{(n)} S_{2n}^{2ij} + \hat{b}_{n-j}^{(n)} C_{2n}^{2i(n-j)} + \hat{b}_j^{(n)} S_{2n}^{2i(n-j)}] \right\}.$$

Considering that we obtain

$$C_{2n}^{ij} = C_n^{ij} \quad , \quad S_{2n}^{ij} = S_n^{ij} \quad , \quad C_n^{i(n-j)} = C_n^{ij} \quad , \quad S_n^{i(n-j)} = -S_n^{ij}$$

1. $\hat{b}_0^{(n)} = \hat{b}_0^{(2n)} + \hat{b}_n^{(2n)}$,
2. $\hat{b}_{n/2}^{(n)} = 2 \hat{b}_{n/2}^{(2n)}$,
3. $\hat{b}_j^{(n)} = \hat{b}_j^{(2n)} + \hat{b}_{n-j}^{(2n)}$,
4. $\hat{b}_{n-j}^{(n)} = \hat{b}_{2n-j}^{(2n)} - \hat{b}_{n+j}^{(2n)}$,

($1 \leq j \leq n/2-1$). . Performing similar, although more cumbersome calculations for the odd elements, we obtain:

5. $\hat{b}_0^{(n)} = \hat{b}_0^{(2n)} - \hat{b}_n^{(2n)}$,
6. $\hat{b}_{n/2}^{(n)} = 2 \hat{b}_{3n/2}^{(2n)}$,
7. $\hat{b}_j^{(n)} = \text{Re}(Z \bar{W}_{2n}^j)$,
8. $\hat{b}_{n-j}^{(n)} = \text{Im}(Z \bar{W}_{2n}^j)$,

(12)

where $\text{Re} Z = \hat{b}_j^{(n)} - \hat{b}_{n-j}^{(n)}$, $\text{Im} Z = \hat{b}_{2n-j}^{(2n)} + \hat{b}_{n+j}^{(2n)}$, $n=1$:

Formulas (12) hold for any $n \geq 1$, and at $n = 1$:

$$\hat{b}_0^{(n)} = \hat{b}_0^{(n)} = \hat{b}_0^{(n)} + \hat{b}_1^{(n)} \quad , \quad \hat{b}_1^{(n)} = \hat{b}_0^{(n)} = \hat{b}_0^{(n)} - \hat{b}_1^{(n)} .$$

Assuming $d_i^{(n)} = d_i^{(n)}$ for $n = N/2$ and applying (12) in the case $n = \frac{N}{2}, \frac{N}{4}, \dots, 2, 1$ to sequentially ordered auxiliary subfiles, whose number doubles with each step, at the locations of the elements $d_i^{(n)}$ we obtain the shuttled elements of the subfile $d_i^{(n)}$, i.e., the DT result (10). The number of the element ℓ is connected with the number of the element $k = \ell$ by the law of binary inversion:

if

$$\begin{aligned} \ell &= e_{N-1}2^{N-1} + e_{N-2}2^{N-2} + \dots + e_12 + e_0 \\ \ell &= e_02^{N-1} + e_12^{N-2} + \dots + e_{N-2}2 + e_{N-1} \end{aligned} \quad (13)$$

then

where $e_l (l=0, 1, \dots, N-1)$ assumes values of either 0 or 1. For brevity, we shall call the procedure and the result of transposing the subfile elements according to the rule (13) as inversion.

Direct RTF of the real-valued function:

$$\hat{d}_\ell^{(N)} = \frac{1}{N} D_{\ell, k}^{(N)} d_k^{(N)} = \frac{1}{N} \sum_{k=0}^{N-1} \delta_{\ell, k}^{(N)} d_k^{(N)}, \quad (14)$$

where

$$\delta_{\ell, k}^{(N)} = \begin{cases} C_N^{k\ell} & \text{for } 0 \leq \ell \leq N/2, \\ S_N^{k(N-\ell)} & \text{for } N/2+1 \leq \ell \leq N-1, \end{cases}$$

produces the DT sequence described above (including inversion), accomplished in reverse order. The algorithm for changing from the subfiles $\hat{f}_i^{(n)}, \hat{f}_i^{(n)}$ to the subfile $\hat{f}_i^{(n)}$ is produced by solving a linear system of equations (12) relative to $\hat{f}_i^{(n)}$.

The number of arithmetic operations for RTF(10) at $N \gg 3$:

$[(N-7/2)+6]^{*+}$ and $[(3/2)N(N-1)+2]^{*+}$, and with the indirect use of the algorithm (10) -- $[N(N-2)]^{*+}$ and $[N(N/2+2)]^{*+}$. During transformation of the m -dimensional files, having the identical base N in all the measurements, the number of operations increases by a factor of mN^{m-1} ($m = 3, 3\dots$). Direct RTF (14) requires as many as $^{*+}$ and $[N-2]$ fewer $^{*+}$ than the reverse RTF (multiplication by two, we refer to operations of the type of ' + '), if we let the user combine multiplication of each transform element by $1/N$ (in the multidimensional case -- by $1/(N1 \cdot N2 \cdot \dots \cdot Nm)$) with the usually necessary multiplication by values calculated previously, which we do. /10

If the user is interested in complex representations of the transform (6), in the one-dimensional case, we may change to it by using the formulas (9). For many measurements, the change algorithm is complicated. Thus, the two-dimensional DTF of the real-valued function $\Lambda_{k1, k2}^{(N1, N2)} = \text{Re } F_{k1, k2}^{(N1, N2)}$

$$\hat{\Lambda}_{l1, l2}^{(N1, N2)} = \frac{1}{N1 \cdot N2} \sum_{k2=0}^{N2-1} \delta_{l2, k2}^{(N2)} \sum_{k1=0}^{N1-1} \delta_{l1, k1}^{(N1)} \Lambda_{k1, k2}^{(N1, N2)}. \quad (15)$$

Comparing (15) and (8), at $l1 = 0, N1/2$ and at $l2 = 0, N2/2$ we obtain the formulas for connecting $\hat{\Lambda}$ and \hat{F} of the type (9), and at $1 \leq l1 \leq N1/2-1, 1 \leq l2 \leq N2/2-1$:

$$\begin{aligned} \hat{\Lambda}_{l1, l2}^{(N1, N2)} &= \frac{1}{4} \text{Re} \left\{ \hat{F}_{l1, l2}^{(N1, N2)} + \hat{F}_{l1, N2-l2}^{(N1, N2)} + \hat{F}_{N1-l1, l2}^{(N1, N2)} + \hat{F}_{N1-l1, N2-l2}^{(N1, N2)} \right\}, \\ \hat{\Lambda}_{l1, N2-l2}^{(N1, N2)} &= -\frac{1}{4} \text{Im} \left\{ \begin{array}{c} \vdots \\ - \vdots \\ + \vdots \\ - \vdots \end{array} \right\}, \\ \hat{\Lambda}_{N1-l1, l2}^{(N1, N2)} &= \frac{1}{4} \text{Im} \left\{ \begin{array}{c} \vdots \\ + \vdots \\ - \vdots \\ - \vdots \end{array} \right\}, \\ \hat{\Lambda}_{N1-l1, N2-l2}^{(N1, N2)} &= \frac{1}{4} \text{Re} \left\{ \begin{array}{c} \vdots \\ - \vdots \\ - \vdots \\ + \vdots \end{array} \right\}. \end{aligned} \quad (16)$$

The formulas (9), (16), their inversions with respect to \hat{F} and analogous formulas for the case $\text{Re } F = 0, \Lambda = \text{Im } F$ represent the algorithm for the subprogram INRECO described below.

3. FFTD SUBPROGRAM

The FFTD module performs direct (14, 15) and reciprocal (10) DTF for the real-valued subfiles (REAL # 4) of one or several measurements with the bases $N = 2^M$ (M is a whole number) according to each measurement or according to several measurements. /11 It is assumed that the processed files are fully located in the computer operational memory. Their Fourier transforms (increased by a factor of N) occupy the locations of the subfiles in the case of direct transformation; reciprocal transformation restores the subfiles (within an accuracy of the factor N).

The DTF is performed by using the RTF algorithm, described in the previous section (12, 13). Inversion is performed by separate transformation to FFTD at $NC = 0$. The rules for selecting the subfiles were described in Section I. The subfiles selected according to the general law were developed in parallel. The cases $j = n/4$; $C_{2n}^{2n} = S_{2n}^{2n}$, were extracted in the algorithm (12), which leads to a reduction in the number of operations.

The text of the FFTD subprogram in the FORTRAN-4 language is shown in Appendix 1. The FFTD module, cataloged in the library of target modules of the "SOFI" complex [1] in the Institute of Space Research, Academy of Science USSR retains the logic of this text, but the binary cycles 11, 22, 29 are replaced by invocations of the FPM module, which is written

in ASSEMBLER with allowance for the results of the tests on high-speeds groups of machine commands performed on the ES-1040. One result of this study was an increase in the actual operation of the FFTD by a factor of 3.5, as compared with the FORTRAN option. As an example, the direct and reciprocal two-dimensional RTF of the file B2(128,128) require together of 5-6 seconds ES-1040 processor time whereas the translation of the text (Appendix 1) on the BESM-6 computer requires 20-30 seconds to perform these procedures.

At $M \leq 10$, the normed result of the direct reciprocal DTF coincides with the initial (in terms of all the elements) subfile (which differ by 5 orders of magnitude in terms of the module), with a relative error of $\leq 10^{-5}$. When these procedures are repeated many times, the error increases linearly. At $M > 10$ the error increases due to the inexact values of the factors G_{2n}^i , S_{2n}^i , which are not stored in the memory in this variation of the RTF, but are calculated recurrently from G_{2n} , S_{2n} in the DT process.

/12

The module call from the user program, written in FORTRAN, is performed as follows: CALL FFTD. Before the call, it is necessary to write the total region by the operator of the type (5), assign the necessary values of the parameters NC, N, NN, NA, NK and fill the processing section of the file C(NNN) with the necessary information.

FFTD takes into account the values of the five parameters
 INTEGER * 4: NA and NK -- respectively the number of the first and last elements of the processed group of subfiles in the file field C(NNN): $I \leq NA < NK \leq NNN$;

N -- base of the subfiles of the first measurement: N1 - N:
 NN -- base of two-dimensional subfiles of the first and second measurements: N2 - NN/N;

NC -- splitting parameter: at NC = 0 only the inversion of subfiles of first measurement is performed, if $NN/N < 16$; in the opposite case -- inversion of subfiles of the first and second measurements and if $(NK+I-NA)/NN \geq 16$, the section is subjected to inversion for three measurements: NC > 0 -- direct RTF inversion; NC < 0 -- reciprocal RTF:
at

- |NC| = 1 - using three measurements
- |NC| = 2 - using the first two measurements,
- |NC| = 3 - using the first measurement (rows)
- |NC| = 4 - using the second measurement (columns),
- |NC| = 5 - using the third measurement (number of pages).

The memory used by the module is 2.8 K.

4. INRECO SUBPROGRAM

The INRECO module accomplishes the exchange of data between real - A2 and complex - CMP two-dimensional files with identical dimensions, determined by the user on overlapping segments of the file C(NNN). In one of the files, either the function of two discrete arguments or its transform (8) or (15) is /13 given.

Access to the FORTRAN program: CALL INRECO. The parameters: NC - NK. The meaning of the parameters N, NN, NC is the same as for FFTD (see Sec. 3); NA -- number of the first element of the real file; NK -- number of the real part of the first element of the complex file in the C(NNN) field; the splitting parameter NC assumes values from 1 to 8, for 0 = odd values of NC XX=RE -- procedures with the real part of the complex function; for even NC XX = IM -- procedures with the purely imaginary part:

at NC = 1 the XX transform is carried from A2 into CMP;

at NC = 2 the XX transform from A2 is added to the CMP;
 NC = 3,4 the XX transforms are extracted from CMP and
 moved to A2;
 NC = 5,6 the A2 is carried to the XX part of CMP;
 NC = 7,8 the XX part of CMP is carried to A2.
 Thus, operations at NC \leq 4 pertain to the Fourier image spectra;
 at NC \geq 5 -- to the images themselves.

The occupable memory is 3.0 K.

5. FFT2 SUBPROGRAM

The FFT2 module combines three frequently encountered DT procedures of two-dimensional files, determined in the user program as complex:

- 1) direct DTF of the two-dimensional file (image) with a zero imaginary part;
- 2) inverse DTF of the complex two-dimensional Fourier transform (it was not apparent previously that the imaginary portion of the result is null);
- 3) multiplication of the complex image spectrum by the complex conjugate of the "reference" spectrum.

/14

Each procedure is carried out for individual access to FFT2:

- 1 - at NC $>$ 0,
- 2 - at NC \leq 0,
- 3 - at NC = 0.

For procedures 1 and 2, the user must determine on the file

field C(NNN) one or two real files of the same dimensions as the complex ones: $NI = N$, $N2 = NN/N$; NK designates the origin of the complex file; and NA designates the position of the first element of the first of the working real files. For procedure 3, the user determines two complex files of the dimensions given above, whose initial address corresponds to the $C(NK)$ and $C(NA)$ addresses; the conjugate is given according to elements of the second, and the result is written in the first file. The second complex file and the working real files may be determined in the same section of the field (CNNN).

FFT2 calls the FFTD and INRECO modules. It may serve as an example of access to this sub. The FFT2 text is presented in Appendix II. The occupable memory is 0.9 K.

6. COMPUTATION OF CORRELATION FUNCTIONS. EXAMPLE

An example of using the moduli described above, and also their text, is the program whose text is presented in Appendix III. The bases of e files (16, 16) were selected for ease of visualization of the image function and the correlation function, performed by the PR2 module. The KREST module performs generation of the reference and the images. Print-outs from the PR2 are given in Appendix IV:

a) The reference-cross with the center coordinates (6,8). /15
The coordinates are selected to the right and below the element (1,1). In view of the periodic nature of the images, for negative values of a certain coordinate the origin of the coordinate will be one of the positions designated by a cross;

b and c) Images No. 1 and NO. 2 represent two crosses with

the parameters of the reference, shifted with respect to the last and with respect to each other. The dotted line combines the cross contours. The arrows are vectors of the reference center to the center of the image element;

d and e) The correlation functions of the images No. 1 and No. 2 with the reference a), obtained as the result of calculations using the program (Appendix IV). The arrows are the vectors from the "corresponding" origin of the coordinates to the position of the correlation function maximum; the dashed line is drawn on the level of the values of the correlation function module, which equals half of the maximum value.

In conclusion, the authors would like to thank N. I. Kalinin, T.E. Krenkelya and B. I. Kolosov for valuable discussions.

1. Zolotukhin, V.G., Kolosov, B. I., Usikov, D. A., Borisenko, V. I., Mosin, S. T., Gorohiov, V. N., Opisaniye sistemy programm matematicheskoy obrabotki na EVM Yedinoy Serii (YeS) fotograficheskikh izobrazheniy semli, poluchennyky s kosmicheskiki apparatov (kompleks "SOFT"), [Description of a system of programs for mathematical processing by unified computers (ES) of photographic images of the earth obtained from spacecraft ("SOFI" equipment)]. Preprint of the IKI AN SSSR, Moscow, 1977, Preprint 336.
2. Trakhtman, A. M., Trakhtmaya, V. A. Osnovy teorii diskretnykh signalov na konechnykh intervalakh (Bases of theory of discrete signals at finite intervals). Moscow, Izdatel'stvo "Sovetskoye radio", 1975.
3. Usikov, D. A., Primeneniye abstraktnogo garmonicheskogo analiza dlya bystrogo raspoznavaniya izobrazheniy (Use of abstract harmonic analysis for rapid identification of images). Preprint IKI AN SSSR, Moscow, 1977, preprint 335.
4. Cooley, J. V., Tuckey, J. V. An algorithm for the Machine Calculation of Complex Fourier Series. "Mathem. Comput." April 1965, V. 19, 297-301.
5. Tkhabisimov, D. A. Bystryy korrelyatsionnyy analiz na gruppe dvizheniy i odnorodnykh masshtabnykh preobrazovaniy ploskosti $(M(2) \times R_+)$, (Rapid correlation analysis for a group of motions and homogeneous scales of the $(M(2) \times R_+)$ plane transformations). Preprint IKI AN SSSR, Moscow, 1977, Preprint 367.
6. Soroko, L. M., Strizh, T. A. Spektiral'nyye preobrazovaniya na tsifrovyykh vychislitel'nykh mashinakh (Spectral transformations on digital computers). Dubna, 1972.

APPENDIX I

DOO/ES FORTRAN IV V.M. 4.0		0000 0000 0000	
0001	SUBROUTINE PPTO	0007	U=0-Z
0002	COMMON PIG(304), NC, N, NN, NA, NK, C(1)	0008	IF(N)40, 12, 14
0003	50 A=UBRT(.5)	0009	C(I)=0.507K
0004	51 IA=1	0010	C(I)=708-NK
0005	NS=N	0011	ACTO 22
0006	52 IF(NC=NC-10)3, 1, 2	0012	17 C(I)=(0.5)A0
0007	1 IA=N	0013	11 C(I)=(1-Y)A0
0008	NS=NN	0014	22 GOTU 22
0009	GOTU 3	0015	10 IF(N)10, 20, 21
0010	2 IA=NN	0016	10 X=N-N-V0
0011	NS=NK-NA+1	0017	U=N*0.508
0012	3 NS=NA-IA-1	0018	GOTU 21
0013	53 IP=IA	0019	20 Y=(N)A0
0014	NS=0.5A	0020	U=1.00Y)A0
0015	54 IF(NC)5, 35, 10	0021	21 C(I)=0.5Y
0016	4 IP=IP+2	0022	C(I)=0.5Y
0017	IF(IP=NS)10, 6, 7	0023	C(I)=0.5Y
0018	5 IP=NS	0024	22 CONTINUE
0019	6 A=AU	0025	X=0
0020	7 D=UBRT(.5-.50A)	0026	IF(N)23, 26, 26
0021	A=UBRT(.5-.50A)	0027	23 KK=IR=KK
0022	IF(NC)0, 50, 10	0028	5=0
0023	8 IF(IP=2-NS)6, 10, 50	0029	5=K
0024	9 X=A	0030	GOTO 14
0025	A=X*X-D*D	0031	24 KK=IR-KK+2
0026	D=2.*X*D	0032	IF(N-IA)50, 14, 28
0027	10 IP=2-IP	0033	5=A*0.508
0028	DO 11 K=NA, NS	0034	F=A*X*D
0029	J1=IP-K	0035	GOTO 14
0030	DO 11 J=J1, NK, IP2	0036	26 IF(NC)27, 50, 30
0031	IR=IP	0037	27 IF(IP-IA)50, 31, 28
0032	X=C(J)	0038	28 DO 29 K=NA, NS
0033	Y=C(I)	0039	J1=IR-K
0034	C(J)=Y-X	0040	DO 29 J=J1, NK, IP
0035	11 C(I)=Y-X	0041	29 C(J)=0.5C(J)
0036	IR=IP/2	0042	IP=IP/2
0037	IF(IR=2*IA)26, 13, 12	0043	IF(IP=NS)10, 9, 9
0038	12 B=A	0044	30 IF(IP=NS)4, 32, 50
0039	F=D	0045	31 IF(NC=2)40, 33, 34
0040	53 KK=IA	0046	32 IF(NC=2)34, 33, 40
0041	14 N=KK-IR/2	0047	33 IF(IA=N)1, 40, 40
0042	DO 42 K=NA, NS	0048	34 IF(IA=N)1, 2, 40
0043	J1=IP-K	0049	35 IF(N=10) 30, 135, 135
0044	DO 42 J=J1, NK, IP2	0050	36 IP2=IP+2
0045	IR=J-KK	0051	I2=N
0046	I2=I-IP	0052	N=N/2
0047	IX=I3-IP	0053	NS=20-N
0048	X=C(I)	0054	DO 37 K=NA, NS
0049	Y=C(IX)	0055	DO 37 I3=K, NK, I2
0050	V=C(I2)	0056	IR=N+I3
0051	B=C(I2)	0057	J1=IP, I3
0052	Z=C(I3)	0058	
0053	IF(NC)15, 50, 18	0059	
0054	15 C(I2)=B+Z	0060	
0055	C(I3)=V-Z	0061	
0056	Y=X+V	0062	
0057	37 CONTINUE	0063	
0058		0064	
0059		0065	
0060		0066	
0061		0067	
0062		0068	
0063		0069	
0064		0070	
0065		0071	
0066		0072	
0067		0073	
0068		0074	
0069		0075	
0070		0076	
0071		0077	
0072		0078	
0073		0079	
0074		0080	
0075		0081	
0076		0082	
0077		0083	
0078		0084	
0079		0085	
0080		0086	
0081		0087	
0082		0088	
0083		0089	
0084		0090	
0085		0091	
0086		0092	
0087		0093	
0088		0094	
0089		0095	
0090		0096	
0091		0097	
0092		0098	
0093		0099	
0094		0100	
0095		0101	
0096		0102	
0097		0103	
0098		0104	
0099		0105	
0100		0106	
0101		0107	
0102		0108	
0103		0109	
0104			
0105			
0106			
0107			
0108			
0109			
0110	DO 37 J=J1, IR, IP2	0118	IP=0.5P
0111	KK=J+IP-IA	0119	IF(IP=N)36, 38, 38
0112	DO 37 I=N, KK, IA	0120	38 IF(N/N-10)40, 39, 39
0113	I=N-NS	0121	39 IF((NK-NA+1)/N-10)33, 34, 34
0114	B=C(I)	0122	40 PRINT 41, N, NN, NA, NK, IA, IP, NC
0115	C(I)=C(IX)	0123	41 FORMAT(0X, '0001', 7I10)
0116	C(IA)=B	0124	59 RETURN
0117	37 CONTINUE	0125	END

APPENDIX II

DOU/ES FORTRAN IV V.M 2.0

FFT2

```

0001      SUBROUTINE FFT2
0002      COMMON FICT(304),NC,N,NA,NA,NK,C(1)
0003      NZ=NK
0004      IF(NZ) 2,3,1
0005      1 NC=7
0006      CALL INREC0
0007      NC=U
0008      NK=NA-NN-1
0009      CALL FFTD
0010      NC=2
0011      CALL FFTD
0012      NK=NZ
0013      NC=1
0014      CALL INREC0
0015      RETURN
0016      2 NC=3
0017      CALL INREC0
0018      NC=4
0019      NA=NA-NN
0020      CALL INREC0
0021      NC=3
0022      NK=NA-NN-1
0023      NA=NA-NN
0024      CALL FFTD
0025      NC=0
0026      CALL FFTD
0027      NK=NZ
0028      NC=5
0029      CALL INREC0
0030      NA=NA-NN
0031      NC=6
0032      CALL INREC0
0033      NA=NA-NN
0034      RETURN
0035      3 ND=NA-NK
0036      NE=NK-2*NN-2
0037      DO 4 I=NK,NE,2
0038      X=C(I)*C(I+ND)+C(I+1)*C(I+ND+1)
0039      C(I+1)=C(I-1)*C(I+ND)-C(I)*C(I+ND+1)
0040      4 C(I)=X
0041      RETURN
0042      END

```

APPENDIX III

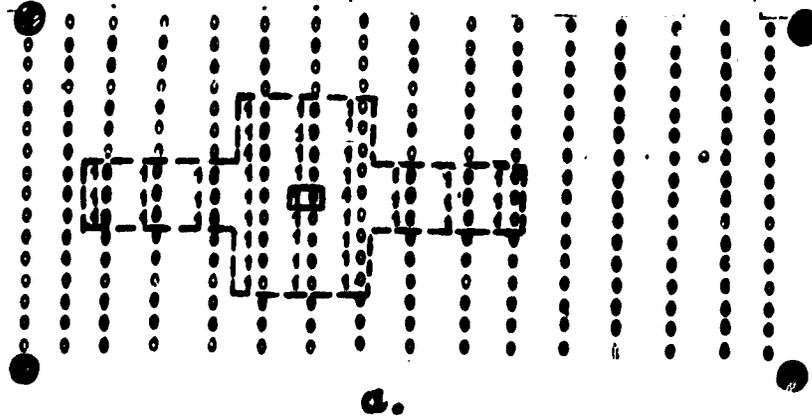
```

DO 8/85 FORTRAN IV V.M 2.0
0001 COMPLEX A(10,10),B(10,10)
0002 COMMON /107(1004),NC,N,NH,NA,NE,A,B(10,10),E(10,10),B
0003 N=10
0004 NH=NH
0005 NK=1
0006 NA=NH*2+NH
0007 CALL KRST(7,0,1,4,10.)
0008 NC=1
0009 CALL FFT2
0010 DO 1 I=1,N
0011 DO 1 J=1,N
0012 B(I,J)=A(I,J)
0013 1 CONTINUE
0014 DO 1 I=5,11*5
0015 IS=2
0016 CALL KRST(IS,12,1,4,10.)
0017 IS=IS-17
0018 CALL KRST(IS,5,1,4,10.)
0019 CALL PR2
0020 NC=1
0021 CALL FFT2
0022 NC=0
0023 NA=NA+2+NH
0024 CALL FFT2
0025 NA=NA-2+NH
0026 NC=-1
0027 NK=1
0028 CALL FFT2
0029 CALL PR2
0030 5 CONTINUE
0031 STOP
0032 END

```

reference
generation
reference DTF
Rewriting reference spectrum from A to B
generation and visualization of image
image DTF
multiplication of spectrum A and B
inverse DTF
visualization of correlation function

APPENDIX IV



9T

