Johannsen

LARS Information Note 121069
Purdue University

GUIDE TO USE OF THE FAST FOURIER TRANSFORM ALGORITHM FOR TWO DIMENSIONAL IMAGERY CORRELATION

by

Paul E. Anuta

Introduction

The traditional method of correlating two (2) dimensional discrete data arrays is a numerical integration and shifting procedure which is extremely time consuming for large arrays. Two dimensional correlation is required as a part of the imagery registration system under development at LARS. Methods of increasing the computation speed for correlation have been investigated and the Fast Fourier Transform was observed to be up to about forty times faster than the numerical integration approach for the cases studied. This information note explains some of the basic characteristics of the Fast Fourier Transform correlation function algorithm and outlines the steps necessary to obtain the desired results using the FFT.

Two Dimensional Imagery Correlation

A. Basic Correlation Function

The one dimensional registration system developed during 1968 was the first step toward solution of the general multispectral image registration problem. Experience gained from development and use of this system stimulated the development of new techniques to improve the speed and accuracy of the registration system. The major problem with the one dimensional system was that the correlation area available in one line or column segment was often inadequate to achieve lock-on between two widely separated wavelength bands. The reflectance properties of an area tend to differ markedly from its thermal emission properties thus correlation between reflective imagery and thermal infrared imagery becomes difficult. Also, the cross-coupling effect is severe for separate orthogonal correlation of lines and columns. The essence of the cross-coupling effect is that the misregistration in one dimension can cause correlation in the other dimension to fail due to the displacement of that coordinate. Assume two image arrays A and B are misaligned by ΔC columns and with no line misregistration. Then correlation of a column in A with the corresponding column in the B array may fail since they are ΔC columns apart in terms of matching context. The same is true for a line shift. Other problems such as the difficulty of rotating the lines and columns led to the decision to store and process the imagery in two dimensional form. Two dimensional multichannel imagery processing requires a square law increase in computer core space. The decision to implement two dimensional processing was made possible by the acquisition of additional core memory for the LARS computer which doubled the original memory space.

The basic function which must be implemented for discrete cross correlation is:

$$\phi(k,l) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} x(i+k, j+l) y(i,j)}{\sum_{i=1}^{M} \sum_{j=1}^{N} x^2(i+k, j+l) \sum_{i=1}^{M} \sum_{j=1}^{N} y^2(i,j)}$$

$\phi(k,l)$ is the two dimensional correlation function for the $\{X\}$ and $\{Y\}$ image arrays.

k and l are the shift variables and their limits depend on the search range required for the registration task being considered.

The bracket notation $\{X\}$ will be used to denote the set of all image points identified by X. The shift variables k and l vary over a range determined by the size of the x array and by the expected amount of misregistration. The x and y data sets are corrected so as to have zero mean and the denominator sum of the squares terms normalize the function so that 100% positive correlation of $\{x\}$ and $\{y\}$ is indicated by a plus one and 100% negative correlation by a minus one. In the research done to date the shift parameters k and l were varied plus and minus ten to twenty picture points in each direction and the summation ranges M and N were varied from four to thirty-two picture points and only square areas have been used, i.e., $M = N$. The size of the correlation function is as follows:

Let Δk = maximum shift in k shift variable

Δl = maximum shift in l shift variable

Then the correlation function $\phi(k,l)$ has $2\Delta k + 1$ points in the k direction and $2\Delta l + 1$ in the l direction. Again equal shift limits have been used in the research to date ($\Delta k = \Delta l$). This correlation function was implemented along with basic card input, storage allocation, and correlation function printout programming.

B. Fast Fourier Transform Correlation Method

The computing time necessary for straightforward evaluation of the two dimensional correlation function proved to be excessive; on the order of two minutes per evaluation in the typical case. An alternate method of computing the correlation function exists which employs the Fourier Transform. The discovery of the Fast Fourier Transform algorithm enabled a great reduction in the time required to compute the Fourier Transform, thus a fast means of computing the correlation function was made available. The time required to compute an n point transform using the Fast Fourier Transform varies as nlogn instead of approximately n^2 for conventional numerical integration evaluation. Thus a time savings in computation of the correlation function from two factors can be achieved: (1) the Fourier Transform method itself, and (2) the Fast Fourier Transform algorithm. Therefore, the transform method was implemented in the Version II registration system.

Certain problems unique to the use of the finite Fourier Transform presented themselves and the solutions to them bear mentioning here since a significant amount of time was spent in solving them. The convolution theorem of applied mathematics states that multiplication in the frequency domain is analogous to convolution in the time domain and this is expressed mathematically as:

$$C(k) = \int_{-\infty}^{\infty} x(t)y(k-t)dt = FT^{-1}\left[X(f)Y(f)\right] \qquad \text{B1}$$

Where

x,y(t)	two time functions
C(k)	is the convolution between two time functions shifted by k units with respect to each other
X,Y(f)	are the Fourier Transforms of the two time functions. The transform is a function of the frequency variable f.
FT^{-1}	signifies the inverse Fourier Transform operation

Cross correlation is defined in the same way as convolution except that one of the two functions is not reversed but is simply shifted on its axis. The correlation function is expressed as:

$$\phi(k) = \int_{-\infty}^{\infty} x(t)y(t-k)dt = FT^{-1}\left[X(f)Y^{*}(f)\right] \qquad \text{B2}$$

Where $\emptyset(k)$ is the correlation function of the shift variable k
$Y^*(f)$ is the complex conjugate of the Fourier Transform of $y(t)$

Straight-forward application of the above expression to computation of the correlation function of discrete data using the discrete Fourier transform leads to problems causing to completely erroneous results. The Fourier Transform algorithm computes the discrete N term Fourier series of and N point function. Inherent in the operation of the transform algorithm is the assumption that the function being transformed is periodic. The resulting N point transform is also periodic. The result of the application of the above expression in the discrete case is a cyclical convolution function which in most cases will give an erroneous point of maximum correlation. This problem can be alleviated by increasing the size of the transform and including zero values for the range of shift desired. For $x(k)$ defined at M discrete points (M even) $k = 0, 1, \ldots M-1$ and $y(k)$ defined at $N<M$ discrete points (N even), let $y(k) = 0$ for $k = 0, 1, \ldots \frac{M-N}{2} - 1$ and $k = \frac{M+N}{2}, \ldots M-1$. Then executing the correlation process using the transform will give the correct result for a shift of $\frac{N-M}{2}$ points in each direction.

Thus:

$$\emptyset(k) = \sum_{i=0}^{M-1} x(i)y(i+k) \qquad k = 0, \pm 1, \ldots \pm \frac{M-N}{2} \qquad \text{B3}$$

is computed using the Fast Fourier Transform by multiplying the M point transforms of $\{x\}$ and $\{y\}$ constructed with M-N zeros in the expanded $\{y\}$ array as follows:

$$\emptyset(k) = FT^{-1}\left[X(f)Y^*(f)\right] \qquad k = 0, 1, \ldots, M-1 \qquad \text{B4}$$

The $k = 0$ point is the correlation for no shift, $k = 1$ one point shift in the positive direction and so on up to $\frac{N-N}{2}$ points of shift in one direction. The $\frac{N+N}{2}$ point is the correlation for maximum shift in the opposite direction and the $k = M-1$ value is a one point shift in the negative direction. This split is due to the cyclic property of the transform and behavior of this type must be accounted for in any system which uses the transform technique. The values of $\phi(k)$ for $k = \frac{M-N}{2} + 1$ up to $k = \frac{M+N}{2} - 1$ are invalid and are not used. They represent the correlation of y shifted such that values of $y(i)$ for $i>k$ are wrapped around the end of $\{x\}$ and are being correlated with $x(0)$, $x(1)$, etc., which is meaningless in most cases. This picture is changed if the zeros are included in the $\{y\}$ array at different points. It can be stated in general that the only valid correlation function points are those with the same index values as the zero points in the $y(i)$ function.

The cyclic convolution elimination problem becomes more complicated for the two dimensional case. The $\{y\}$ points are surrounded by zeros on four sides to fill it out to the size of the larger $\{x\}$ array. The valid correlation points are identified in the same manner as for the one dimensional case except that the four quadrants of the valid correlation function are at the four edges of the total cyclical correlation function. Specifically let the $\{x\}$ array be of size M by M $\{x(i,j,)\ \ i = 0, \ldots M-1, j = 0, 1, \ldots M-1\}$ and y' be N x N $\{y'(i,j),\ i = 0 \ldots N-1, j = 0 \ldots M-1\}$ with $N<M$. (M and N even) The N x N points are assumed to be in the center of an M x M size array and this array is padded out with zeros such that:

$$y(i,j)=0 \qquad i,j = 0, 1, \ldots \frac{M-N}{2} \text{ and } i,j = \frac{M+N}{2}, \ldots M-1$$

$$y(i,j)=y'(i-s,j-s) \qquad i,j = \frac{M-N}{2} + 1, \ldots, \frac{M+N}{2} - 1 \qquad \text{B5}$$

$$S= \frac{M-N}{2} \qquad y(i,j) \text{ is an } M \times M \text{ array}$$

This is the same basic format as for the one dimensional case. Similarly the valid correlation function points lie in the first $\frac{M-N}{2}$ points in each corner of the two dimensional square result array.

In order to prevent data sets with large average values from "swamping" the correlation by effectively introducing a large square pulse into the data the average value of the $\{y\}$ data points is removed before padding in the zeros. The average value of $\{x\}$ is also removed to mimimize the magnitude of the correlation function. A step by step account of the operations necessary for two dimensional correlation are now described.

1. Select the size in image points of the area to be covered by the correlation integral. Assuming it is square let this value be N (N even). (The rectangular case is a trivial extension of the square case.) Next select the maximum shift of one array with respect to the other. Let this be Δ. The correlation function will then have $2\Delta + 1$ points in each direction; plus and minus Δ and one for zero shift.
2. Step one defines the necessary size for the base or $\{x\}$ array. It is: $N + 2\Delta$ square and this is called M. The average of the $M \times M$ $\{x\}$ array is removed and the average of the $N \times N$ $\{y\}$ array is removed. The $\{y\}$ set is placed in the center of an $M \times M$ array which is padded out with zeros as defined above.
3. The $M \times M$ Fourier Transform of $\{x\}$ and $\{y\}$ is computed using the Fast Fourier Transform algorithm HARM (SHARE Program No. SDA 3425 or similar versions). The complex conjugate of the Y transform is

taken; the X and Y transforms are multiplied and the inverse transform is taken which produces the total M x M cyclical correlation function with the valid points at the corners of the array. Mathematically these operations are expressed as follows. The two dimensional transformation is:

$$X(f,g) = \frac{1}{M^2} \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} x(i,j) W_M^{-fi} W_M^{-gj} \qquad W_M = e^{\frac{2\pi\sqrt{-1}}{M}}$$

Y(f,g) is computed in the same way.

Then: $$\emptyset(k,l) = \sum_{f=0}^{M-1} \sum_{g=0}^{M-1} \left[X(f,g)Y^*(f,g)\right] W_M^{kf}\, W_M^{lg} \qquad \text{B6}$$

4. The resulting total correlation function $\emptyset(k,l)$ is partitioned and the quadrants are interchanged to place the zero shift point in the center of a $2\Delta + 1$ by $2\Delta + 1$ two dimensional correlation function as follows: $\left(\Delta = \frac{M-N}{2}\right)$

$$\psi\,(k+\Delta,\ l+\Delta) = \emptyset(k,l) \qquad k,l = 0, 1, \ldots, \Delta$$

$$\psi\,(k+\Delta,\ l-M+\Delta) = \emptyset(k,l) \qquad k = 1, \ldots, \Delta;\quad l = M-\Delta-1, \ldots, M-1$$

$$\psi\,(k-M+\Delta+1,\ l+\Delta) = \emptyset(k,l) \qquad k = M-\Delta-1, \ldots, M-1;\quad l = 0, \ldots, \Delta$$

$$\psi\,(k-M+\Delta+1,\ l-M+\Delta+1) = \emptyset(k,l) \qquad k,l = M-\Delta-1, \ldots, M-1 \qquad \text{B7}$$

The function ψ is a $2\Delta + 1$ by $2\Delta + 1$ array of correlation values with the zero shift point in the center. This array is the output of the correlation routine.

A core storage saving scheme can be used when employing the Fast Fourier transform routine for transforming real data. The transform algorithm is written to be able to process complex data thus each input data point is a double computer word. To perform correlation two separate arrays are transformed, i.e., the {x} and {y} data sets. The total number of computer words required for the M x M transform is thus $2\cdot 2\cdot M^2$. The core saving method is based on the fact that the real part of the Fourier Transform of real data is even about the zero frequency point and the imagery part is odd. This fact is implicit in the following development for the one dimensional case. We wish to compute X and Y(k) from x(j) and y(j). This is expressed by the inverse transforms:

$$x(j) = \sum_{k=0}^{M-1} X(k)\, W_M^{jk} \qquad \text{B8}$$

$$y(j) = \sum_{k=0}^{M-1} Y(k)\, W_M^{jk} \qquad W_M = e^{\frac{2\pi\sqrt{-1}}{M}} \qquad \text{B9}$$

One real data set {x} is placed in the real part of the input data array and the other {y} is placed in the imaginary part so that a complex array is formed:

$$\xi(j) = x(j) + iy(j) \qquad \text{B10}$$

The {ξ} array is then transformed to the complex frequency domain forming Z(k):

$$\xi(j) = \sum_{k=0}^{M-1} Z(k)\, W_M^{jk} \qquad \text{B11}$$

To get the X and Y transform from the transform Z the following development is used: Multiply the y transform expression B9 by $i = \sqrt{-1}$ and then add to and subtract it from the x transform expression B8 which produces:

$$x(j) \pm iy(j) = \sum_{k=0}^{M-1} (X(k) \pm iY(k))\, W_M^{jk} \qquad \text{B12}$$

The complex conjugate of B11 can be written in terms of an inverse transform by setting k' = M-k as follows:

$$\tilde{\xi}(j) = \sum_{k=0}^{M-1} \tilde{Z}(k)\, W_M^{-jk} = \sum_{k'=0}^{M-1} \tilde{Z}(M-k)\, W_M^{jk'} \qquad \text{B13}$$

(The tilda indicates complex conjugate.)

Equating coefficients of expression B11 with those of B12 having the plus sign (+) gives:

$$Z(k) = X(k) + i\, Y(k) \qquad \text{B14}$$

and equating coefficients of B13 with those of B12 having a minus sign (-) gives.

$$\tilde{Z}(M-k) = X(k) - i\, Y(k) \qquad \text{B15}$$

Solving these two expressions for X and Y gives.

$$X(k) = \tfrac{1}{2}(Z(k) + \tilde{Z}(M-k)) \qquad \text{B16}$$

$$Y(k) = \tfrac{1}{2}(z(k) - \tilde{Z}(M-k)) \qquad \text{B17}$$

Thus the X and Y transforms can be resolved from the transform of the complex combination by applying the above expression. This is implemented in the correlation program thereby cutting the core requirement for array storage in half. For the 32 x 32 point array being used this is a saving of $2 \cdot 32 \cdot 32 = 2048$ words or 8192 bytes of storage.

C. Correlation Function Characteristics

The two dimensional correlation function computation techniques discussed in parts A and B form the core of the Version 2 image registration system. The computer implementation of these functions has been carried out and analysis of correlation results on imagery from a variety of flight lines is continuing. When two scenes are correlated a $2\Delta + 1$ by $2\Delta + 1$ square array of correlation values is produced. The point of maximum correlation is the one with maximum value. The position of the maximum value in the array gives the misregistration of the two image scenes being correlated. If the maximum point is at the center of the array then the two scenes are in registration to within one image point. In order to pictorially print out the correlation function the values are scaled from 0 to 9 and the array is printed as a rectangular box. Contour lines of constant correlation function value ~~have~~ are then drawn in to better illustrate the variation in correlation as the two images are moved with respect to each other. Two forms of distance measure are computed: Picture distance and Euclidian distance. Picture distance is the row and column misregistration and the Euclidian distance is the root of the sum of the squares of the row and column misregistration. The picture distance is used for registering the imagery and the Euclidian distance is used for control and evaluation purposes.

The Fast Fourier transform method described above significantly reduces the time required to compute the correlation function compared to the numerical integration approach. Table 1 presents a comparison of the time required to compute the correlation function by the two methods. The numerical integration time refers to the conventional method of computing the correlation function discussed in part A. The Fast Fourier Transform time is the time to compute the averages of the two data arrays, set up the complex array, take a forward and a reverse two dimensional fourier transform, unscramble the two transforms as required by the core saving method discussed above, and to extract the valid correlation function points from the total correlation function. The time savings using the Fast Fourier Transform averages about an order of magnitude and this savings has a great impact on the usefulness of digital registration methods.

Table 1 Time Comparison of Numerical Integration and Fast Fourier Transform Methods of Computing Correlation Function

Correlation	Maximum Shift in Both Directions							
	± 5 points		± 10		± 15		± 20	
Area Points Sq.	Numerical Integration (sec.)	Fast Fourier Transform (sec.)	Num Int. (sec.)	FFT (sec.)	Num Int. (sec.)	FFT (sec.)	Num Int. (sec.)	FFT (sec.)
4	1.5	.70	5.4	3.2	11.7	13.9	20.4	13.6
8	5.7	3.4	20.5	3.2	44.5	13.9	77.8	13.6
12	12.6	3.4	45.6	3.2	99.2	13.9	173.3	13.6
16	22.4	3.4	80.8	14.4	175.6	13.9	306.9	13.6
20	34.9	3.4	125.9	14.4	273.8	13.9	478.4	13.6
24	50.1	15.1	181.1	14.4	393.7	13.9	688.1	13.6

Survey of References

The techniques discussed in this information note are based on materials from two basic sources, References 1 and 2. This material is duplicated and expanded on in many other sources. A general list of references is included below for those seeking further knowledge on the subject of FFT applications.

REFERENCES

1. James W. Cooley, "Complex Finite Fourier Transform Subroutine," IBM Program Description Document, IBM Watson Research Center, Yorktown Heights, New York, October 6, 1966.

2. G. D. Bergland, "A Guided Tour of the Fast Fourier Transform," IEEE Spectrum, July, 1969, pp. 41-52.

BIBLIOGRAPHY

1. J. W. Cooley, P. A. Lewis, P. D. Welch, "The Fast Fourier Transform Algorithm and its Applications," IBM Research Report RC1743, IBM Watson Research Center, Yorktown Heights, New York, February 9, 1965.

   This is a very good report covering almost all applications and goes into detail via derivations and proofs of many theorems. It is to be published more formally in the near future.

2. Cooley, Lewis, Welch, "Applications of the Fast Fourier Transform to Computation of Fourier Integrals, Fourier Series, and Convolution Integrals," IEEE Transactions on Audio and Electroacoustics, Vol. AU 15, No. 2, June 1967, pp. 79-84.

   This is a compact and limited version of the above report which may be more accessible than the IBM report.

3. W. M. Gentleman, G. Sande, "Fast Fourier Transforms for Fun and Profit," 1966 Fall Joint Computer Conference, AFIPS Proceedings, Vol. 29, Spartan Books, Washington, D.C.

4. R. Bracewell, "The Fourier Transform and its Applications," McGraw-Hill, New York, 1965.

   This is a good book on all aspects of the Fourier Transform but does not deal with numerical techniques or the Fast Fourier Transform.

5. H. D. Helms, "Fast Fourier Transform Method of Computing Difference Equations and Simulating Filters," IEEE Transactions on Audio and Electroacoustics, Volume Au-15, No. 2, June 1967.

   This paper explains in more detail the "select saving" and "overlap adding" methods described in reference 2 by Bergland of economizing the time required for convolution using the FFT.

6. T. G. Stockham, "High-speed Convolution and Correlation," 1966 Spring Joint Computer Conference, AFIPS Proc. Vol. 29, Spartan Books, Washington, D.C, 1966, pp. 229-233.

7. R. C. Singleton, "A Method for Computing the Fast Fourier Transform with Auxiliary Memory and Limited High-Speed Storage," IEEE Transactions on Audio and Electroacoustics, Vol. AU-15, No. 2, June 1967, pp. 91-98.

Reference 2 by Bergland contains a large 62 entry bibliography on the FFT subject.

For parallel processing work the following may be of interest:

8. M. C. Pease, "An Adaptation of the Fast Fourier Transform for Parallel Processing," Journal of the Association for Computing Machinery, Vol. 15, No. 2, April 1968, pp. 252-264.