LARS Information Note 022175

INTERPOLATION OF ERTS-1 MULTISPECTRAL SCANNER DATA

C. D. McGillem

The Laboratory for Applications of Remote Sensing

Purdue University, West Lafayette, Indiana

1975

LARS Information Note 022175

INTERPOLATION OF MSS DATA

C. D. McGillem

1. Introduction

When continuous data is discretized through sampling there is inevitably distortion that occurs when the data is reproduced. Furthermore, there is no simple way to increase the scale of the sampled data set. A good example of a data set having this problem is the output of the ERTS Multispectral Scanner System. This data set is sampled at a discrete set of spatial coordinates and is quantized in amplitude. When the data is reproduced for visual observation, there is an intrinsic graininess in the picture due to the sampling. When an attempt is made to enlarge the picture, difficulty is encountered because data exists only at discrete spatial coordinates and there is no data between these points. One method of attacking this problem is to repeat each point a number of times thus generating a picture that is enlarged in proportion to the number of repetitions of the points. As can be imagined this type of enlargement does not lead to an improvement of the picture but rather serves to make some of the details more easily seen. Figure 1 shows this type of enlargement of a portion of an ERTS frame taken from the Washington, D.C. region. Figure 1a is a reproduction of the data using each point only once. Figure 1b is a 4 x 4 enlargement in which each point is repeated sixteen times. The graininess and artificial appearance of the enlarged picture are very evident on close inspection. From a distance the appearance is somewhat more acceptable but still has a very artificial look.

An alternative method of producing an enlarged picture is to compute new values between the original sample points by means of an appropriate kind of interpolation. It is the purpose of this paper to describe three

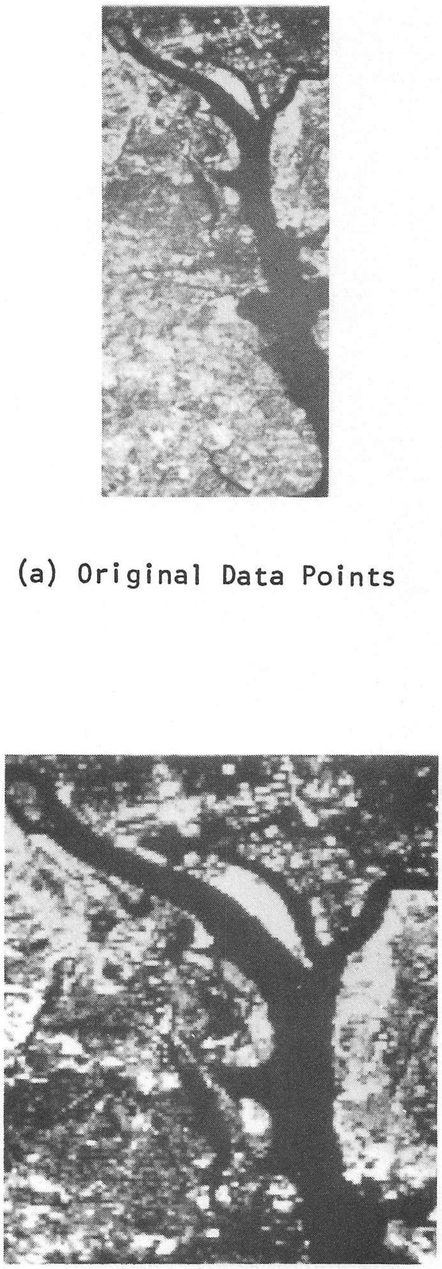
The work described in this report was sponsored by the National Aeronautics and Space Administration under Grant Number NGL-15-005-112 and Contract NAS9-14016. T-1039/4.

such interpolation procedures and to illustrate the results obtained by applying them to the same area as shown in Figure 1.

The mathematical bases of the interpolation procedures will be presented first and then the results of applying these procedures will be given.

2. Mathematical Bases of Interpolation

There are various rationales for selecting interpolating functions. However, none has been found that indicates the appropriate function to use for interpolating the ERTS MSS data so as to minimize any errors between the interpolated image and the true original image. There are reasons to believe that such a rationale may exist and when it has been properly defined a new kind of interpolation may be appropriate. In the meantime three more or less conventional procedures will be used: polynomial interpolation, trigonometric interpolation and sinc function interpolation. The first two methods involve passing an appropriate curve through the data points surrounding the range of coordinates where the interpolation is to be performed and then computing the interpolated values from the curve. The third method consists of taking a weighted sum of all data points in the set to compute the interpolated values. The weighting function is the sinc function and this type of interpolation is exact for samples taken from a continuous function having no spatial frequency components higher than one half the sampling frequency. These procedures will be considered individually.



(a) Original Data Points

(b) Repeated Points 4 x 4

Fig. 1 Washington D.C. area Run 72041900 10/11/72 Ch 3, Lin 1129-1257, Col 1217-1345

3. Polynomial Interpolation (POLYINT)

By passing a nth order polynomial through n + 1 points it is possible to compute interpolated values from the resulting polynomial. There are several equivalent ways of looking at this type of data modification. Among the most useful are: computing interpolated values as a weighted sum of surrounding points; two dimensional convolution; and processing in the frequency domain. These various approaches will be illustrated in the following example of interpolating three intermediate points between equally spaced samples.

The technique of achieving two dimensional interpolation will be to carryout a sequential operation: interpolate in x-direction first and then interpolate the modified data in the y-direction. This process is equivalent to assuming that the interpolation surface can be represented as the product of functions involving only one coordinate, i.e., if the two-dimensional interpolating surface is g(x,y) then it is assumed

$$g(x,y) = g_1(x)\ g_2(y) \qquad (1)$$

The proper weighting for polynomial interpolation can be computed from the Lagrange interpolation formula as follows:

$$f(x) = L_o\ f(X_o) + L_1\ f(X_1) + \ldots\ L_n\ f(X_n) \qquad (2)$$

where

$$f(X_k) = \text{kth sample value}$$

$$L_k = \prod_{\substack{j=0 \\ j\neq k}}^{n-1} (X-X_j) \Bigg/ \prod_{\substack{j=0 \\ j\neq k}}^{n-1} (X_k-X_j) \qquad (3)$$

For a cubic polynomial four sample values are required and the coefficients can be written as follows. (Note the change in indices so that x varies from x_o to x_1 in the central portion of the interpolation region.)

$$L_{-1} = \frac{(x-x_o)(x-x_1)(x-x_2)}{(x_{-1}-x_o)(x_{-1}-x_1)(x_{-1}-x_2)}$$

$$L_o = \frac{(x-x_{-1})(x-x_1)(x-x_2)}{(x_o-x_{-1})(x_o-x_1)(x_o-x_2)}$$

$$L_1 = \frac{(x-x_{-1})(x-x_o)(x-x_2)}{(x_1-x_{-1})(x_1-x_o)(x_1-x_2)}$$

$$L_2 = \frac{(x-x_{-1})(x-x_o)(x-x_1)}{(x_2-x_{-1})(x_2-x_o)(x_2-x_1)} \qquad (4)$$

Considerable simplification results when the samples are equally spaced and when the coordinate x is expressed as a fraction of the sampling interval. For this case letting $u = x/\Delta x$ the interpolation equation becomes.

$$f(u) = -\frac{1}{6}u(1-u)(2-u)f(-1) + \frac{1}{2}(1+u)(1-u)(2-u)f(o)$$
$$+ \frac{1}{2}(1+u)(u)(2-u)f(1) - \frac{1}{6}(1+u)(u)(1-u)f(2) \qquad (5)$$

For a specific interpolation interval the coefficients can be evaluated and a specific equation determined. As an example consider the interpolation of three equally spaced values between the original values. For this case $u = \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$ and the coefficients are

u	L_{-1}	L_0	L_1	L_2
1/4	-7/128	105/128	35/128	-5/128
1/2	-8/128	72/128	72/128	-8/128
3/4	-5/128	35/128	105/128	-7/128

The resulting equations are

$$f(0) = f(0)$$

$$f(\tfrac{1}{4}) = -7/128f(-1) + \frac{105}{128}f(0) + \frac{35}{128}f(1) - \frac{5}{128}f(2)$$

$$f(\tfrac{1}{2}) = -8/128f(-1) + \frac{72}{128}f(0) + \frac{72}{128}f(1) - \frac{8}{128}f(2)$$

$$f(\tfrac{3}{4}) = -5/128f(-1) + \frac{35}{128}f(0) + \frac{105}{128}f(1) - \frac{7}{128}f(2) \qquad (6)$$

This set of equations can be written in matrix form as

$$\begin{bmatrix} f(0) \\ f(\frac{1}{4}) \\ f(\frac{1}{2}) \\ f(\frac{3}{4}) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-7}{128} & \frac{105}{128} & \frac{35}{128} & \frac{-5}{128} \\ \frac{-8}{128} & \frac{72}{128} & \frac{72}{128} & \frac{-8}{128} \\ \frac{-5}{128} & \frac{35}{128} & \frac{105}{128} & \frac{-7}{128} \end{bmatrix} \begin{bmatrix} f(-1) \\ f(0) \\ f(1) \\ f(2) \end{bmatrix} \qquad (7)$$

$$\underline{f}_u = \underline{A}\,\underline{f}_n$$

The process of interpolating can also be interpreted as discrete convolution. By augmenting the original time series with zeros at points where interpolated values will occur the interpolated time series can be written as

$$\{f_{u\ aug}\} * \{h\} \qquad (8)$$

$$(\ldots f_{-2},0,0,0,f_{-1},0,0,0,f_0,0,0,0,f_1,0,0,0,f_2,\ldots)*$$
$$(0,-5,-8,-7,0,35,72,105,128,105,72,35,0,-7,-8,-5) \times \frac{1}{128} \qquad (9)$$

This representation will be considered in more detail later when the three interpolation schemes are compared.

An equivalent process can be carried out in the frequency domain. Taking the discrete Fourier transform of the above equation we have

$$f(\frac{k2\pi}{NT}) = F_D\{f_{uaug}\} = \sum_{o}^{4N-1} f(n\frac{\Delta T}{4})\ \varepsilon^{\frac{-j2\pi nk}{4N}} \quad k=0,\ 1\ldots L/N-1 \qquad (10)$$

but since $f(n\Delta T/4) = 0$ for all n not a multiple of 4 this can be simplified to

$$F_D\{f_{uaug}\} = \sum_{o}^{N-1} f(nT)\ \varepsilon^{\frac{-j2\pi nk}{N}} \quad k=0,1..L/N-1 \qquad (11)$$

Since $F(k\frac{2\pi}{NT})$ is periodic in k with period of N it is evident that the above transform is just the original spectrum with three replications following it.

In order to carry out the convolution by multiplication of the DFT it is necessary that the functions have the same number of points and that a sufficient number of zeros be added in the time domain to prevent alising. For the present instance this means adding 16 zeros after the data points. To keep the total number of points a power of two the number of data points should be set equal to $2^k - 16$. For example if k = 7 the number of data points is 128 - 16 = 112 and the convolution is

$$(f_o, f_1 \ldots f_{127}, 0 \ldots 0) * (h_o, h \ldots h_{15}, 0, 0..0)$$

The processing operation would then be

$$\{f_{inter}\} = \mathcal{F}_D^{-1}\ \{F(k)\ H(k)\} \qquad (12)$$

where $\mathcal{F}_D^{-1}$ is the inverse discrete Fourier transform operator and F(k) and H(k)are the discrete Fourier transform of {f(n)} and {h(n)} respectively. These operations can be carried out using the fast Fourier transform algorithm.

If the interpolation is to be done one line at a time it appears that matrix multiplication would be more rapid than using the FFT. However, if a number of lines are to be processed simultaneously the transform method may reduce the computation time.

4. Trigonometric Polynomial Interpolation (TRIGINT)

By passing a nth order trigonometric polynomial through n + 1 data points an interpolation can be performed by computing intermediate values from the polynomial. This process is equivalent to computing the Fourier series expansion of a function having sample values corresponding to the data subset. One reason for using trigonometric interpolation is that the errors are uniformly distributed over the interpolation interval as compared to polynomial interpolation in which the errors are much greater near the end points.[1] In the present case this is not a major consideration since the interpolated values all lie in the center interval of a multi-interval data set. However, because it does utilize different basis functions for performing the interpolation it was selected as one of the methods to be studied.

The rate at which the coefficients of a Fourier series approximation of a particular function decrease is determined by the smoothness of the function. If the function is continuous and has a continuous first derivative, the coefficients decrease at least as rapidly as $\frac{1}{k^2}$ where k is the order of the harmonic. One way of obtaining these conditions for an arbitrary function represented by N sample values is as follows.

(i) Choose the interval (0,N) as one half the period of a period function.

(ii) Subtract away from the data the linear trend from the first to the last point, i.e., form a new data set

$$z_k = \frac{y_N - y_o}{N} k + y_o \qquad (13)$$

This will made $z_o = z_N = 0$ and thereby make the periodic function continuous.

(iii) Reflect the data set $\{z_k\}$ around the origin as an odd function. This will cause the periodic function to have a continuous derivative at z_N and z_{-N}.

(iv) Expand this new data set in a sine series according to the formulas.

$$z(t) \quad = \quad \sum_{k=1}^{N-1} b_k \sin \frac{\pi k t}{N} \qquad (14)$$

$$b_k \quad = \quad \frac{2}{N} \sum_{n=1}^{N-1} z_n \sin \frac{\pi k}{N} \qquad (15)$$

Both b_o and b_N are indeterminant since the sine functions are zero for both of these cases.

Finally the interpolated values z(t) are computed and the linear trend of the data added in again giving

$$y(t) = z(t) + \frac{y_N - y_o}{N} t + y_o \qquad (16)$$

If the interpolation is restricted to the center interval then we can replace t by $u = t - \frac{N}{2}$ and allow u to vary from 0 to 1. Thus, the equation for interpolation becomes

$$y(u) = z(\frac{N}{2} + u) + (\frac{y_N - y_o}{N_o})(u + \frac{N_o}{2}) + y_o \qquad (17)$$

As an example consider a six point interpolation for which N = 5

$$b_k = \frac{2}{5} \sum_{n=1}^{4} z_n \sin \frac{\pi k n}{5}$$

$$b_1 = 0.4[.58779z_1 + .95106z_2 - .58779z_3 - .95106z_4]$$

$$b_2 = 0.4[.95106z_1 + .58779z_2 - .58779z_3 - .95106z_4]$$

$$b_3 = 0.4[.95106z_1 - .58779z_2 - .58779z_3 + .95106z_4]$$

$$b_4 = 0.4[.58779z_1 - .95106z_2 + .95106z_3 - .58779z_4] \qquad (18)$$

$$z(u) = b_1 \sin \frac{\pi}{5} (u+2.5) + b_2 \sin \frac{2\pi}{5} (u+2.5) + b_3 \sin \frac{3\pi}{5} (u+2.5)$$
$$+ b_4 \sin \frac{4\pi}{5} (u+2.5) \qquad (19)$$

$$= b_1 \cos \frac{\pi}{5} u - b_c \sin \frac{2\pi}{5} u - b_3 \cos \frac{3\pi}{5} u + b_4 \sin \frac{4\pi}{5} u \qquad (20)$$

$$z(u) = \sum_{k=1}^{N-1} b_k \sin \frac{\pi k(u+N/2)}{N} \qquad (21)$$

$$= \sum_{k=1}^{N-1} \sum_{n=1}^{N-1} \frac{2}{N} z_n \sin \frac{\pi kn}{N} \sin \frac{\pi k(u+N/2)}{2} \qquad (22)$$

This can be put into the weighting function form by combining all the coefficients that multiply the same sample value. The weighting function form is as follows.

$$p(u) = \frac{2}{N} \sum_{k=1}^{N-1} z_n \sin \frac{\pi kn}{N} \sin \frac{\pi k(u+N/2)}{N} \qquad n < u < n + 1 \qquad (23)$$

5. Sinc Function Interpolation (SINCINT)

When a signal is bandlimited to frequencies no higher than one half of the sampling frequency it is possible to exactly reconstruct the signal from its samples. The expression is

$$x(t) = \sum_{k=-\infty}^{\infty} x(k) \text{ sinc } (t-k) \tag{24}$$

where sinc t is defined as $\sin\pi t/\pi t$. The above operation is clearly the convolution of sinc t with samples of x(t) as discussed earlier.

When there are actually higher frequency components present this reconstruction leads to the bandlimited function that most closely fits the original data set.

One problem that arises immediately in using this interpolation procedure is the requirement for a very large data set. The reason for requiring a large number of points is the very slow rate at which the interpolating coefficients fall off away from the point being interpolated. The large width of the interpolating pulse means that it is possible to get undesirable edge effects that will persist through the interpolated picture if there is a sharp discontinuity at the start of the image. These difficulties can be essentially eliminated by using either a Fourier series expansion of the data set or the discrete Fourier transform which is virtually the same thing. Using the latter approach the interpolation is accomplished as follows:

(i) The DFT of the data set is calculated. The highest frequency component will be one half the sampling frequency.

(ii) Under the assumption that the data is bandlimited the spectrum is extended to any arbitrary higher frequency by adding zeros for the higher frequency samples.

(iii) The augmented (with zeros) spectrum is now inverted to give the original function but with data points spaced by an amount determined by the number of zeros added to the original spectrum. If there were originally N samples and kN zeros are added to the spectrum then there will be (k+1)N points in the reconstructed signal. Their the scaling will be k+1.

This operation is carried out most rapidly by means of the fast Fourier transform. In order to use the FFT the number of original data points must be an integral power of two and the scale factor k+1 must also be a power of two.

If there is concern about the presence of large discontinuities in the data that may lead to anticipatory or trailing oscillations, it is possible to modify the interpolation process by multiplying the spectrum by an appropriate Hamming window function that will taper the high-frequency response so as to reduce such oscillations. This capability is included in the interpolation routine and is provided as an option to the user.

6. Interpolation as a Convolution Operation

Interpolation can be viewed as the result of convolving samples of a signal with an appropriate interpolating pulse. Comparison of the interpolating pulse shapes corresponding to different interpolation schemes provides considerable insight into their relative performances. The basis of the interpolation-convolution analogy is the fact that both may be considered as weighted sums of the original data points. For interpolation we have

$$f(u) = \sum_{k=-\infty}^{\infty} f(k)\ c_k(u) \tag{25}$$

$$= \ldots c_{-1}(u)f(-1)+c_o(u)f(o)+c_1(u)f(1)+\ldots \tag{26}$$

where $c_k(u)$ is the interpolation coefficient corresponding to the point $f(k)$. For example, in the case cubic polynomial interpolation there are only four non-zero coefficients, namely

$$\begin{aligned} c_{-1}(u) &= -\frac{1}{6}u(1-u)(2-u) \\ c_o(u) &= \frac{1}{2}(1+u)(1-u)(2-u) \\ c_1(u) &= \frac{1}{2}u(1+u)(2-u) \\ c_2(u) &= -\frac{1}{6}u(1+u)(1-u) \end{aligned} \tag{27}$$

The convolution operation can be expressed as

$$f(u) = \{\sum_{K=-\infty}^{\infty} f(u)\ \delta(u-k)\} * p(u)\} \tag{28}$$

$$= \Sigma\ f(k)\ p(u-k) \tag{29}$$

$$= \ldots p(u+1)f(-1) + p(u)f(o) + p(u-1)f(1)+\ldots \tag{30}$$

where p(u) is the interpolating pulse. From (26) and (30) we obtain the following correspondence between the interpolating pulse and the weighting function.

$$p(u-k) = c_k(u) \qquad 0 < u < 1$$

or

$$p(u) = c_k(u+k) \qquad \begin{matrix} 0 < u+k < 1 \\ -k < u < 1-k \end{matrix}$$

For the cubic interpolation coefficients we obtain

$$\begin{aligned} p(u) &= c_2(u+2) = -\frac{1}{6}(u+2)(u+3)(-1-u) && -2 < u < -1 \\ &= c_1(u+1) = \frac{1}{2}(u+1)(u+2)(1-u) && -1 < u < 0 \\ &= c_o(u) = \frac{1}{2}(1+u)(1-u)(2-u) && 0 < u < 1 \\ &= c_{-1}(u-1) = -\frac{1}{6}(u-1)(2-u)(3-u) && 1 < u < 2 \end{aligned}$$

Substituting u = -u into the above equations shows that p(u) is an even function of u and so only the positive values need be calculated.

Similar calculations can be carried out for the other interpolation methods and the convolution operator determined. Figure 2 shows the convolution pulses for the three kinds of interpolation considered here: polynomial, trigonometric and sinc function.

There is clearly much similarity among the three interpolation operations. The most noticeable difference is the extended nature of the sinc function compared to the other two. Because of this it is possible to get significant edge effects if there is a large discontinuity in amplitude at the edge of the image. It is evident that all of the interpolation schemes will give some overshoot when a step discontinuity in amplitude is encountered. However this has not been found noticeable in the processed images -- possibly because of the limited number of gray shades that can be viewed on the display.

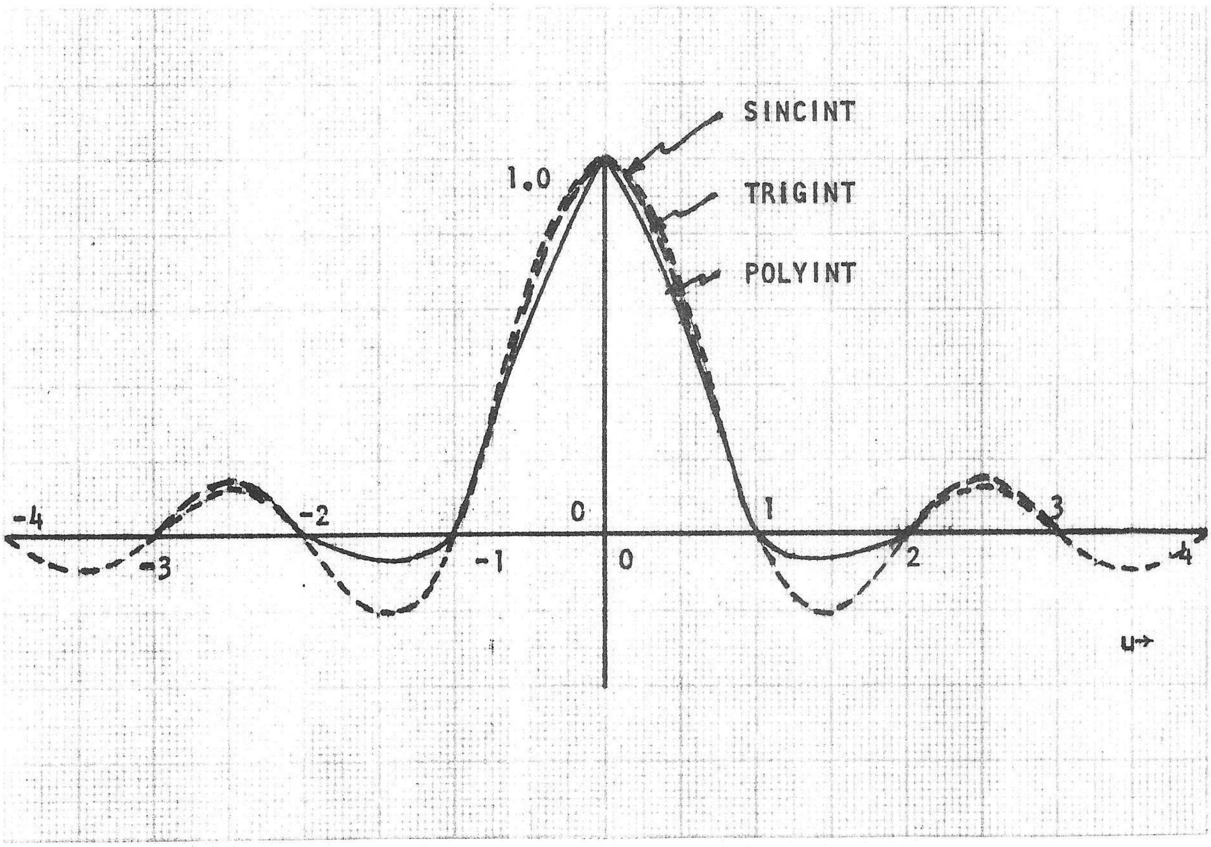


Figure 2 Comparison of Interpolating Pulses

7. Examples of Interpolated Imagery

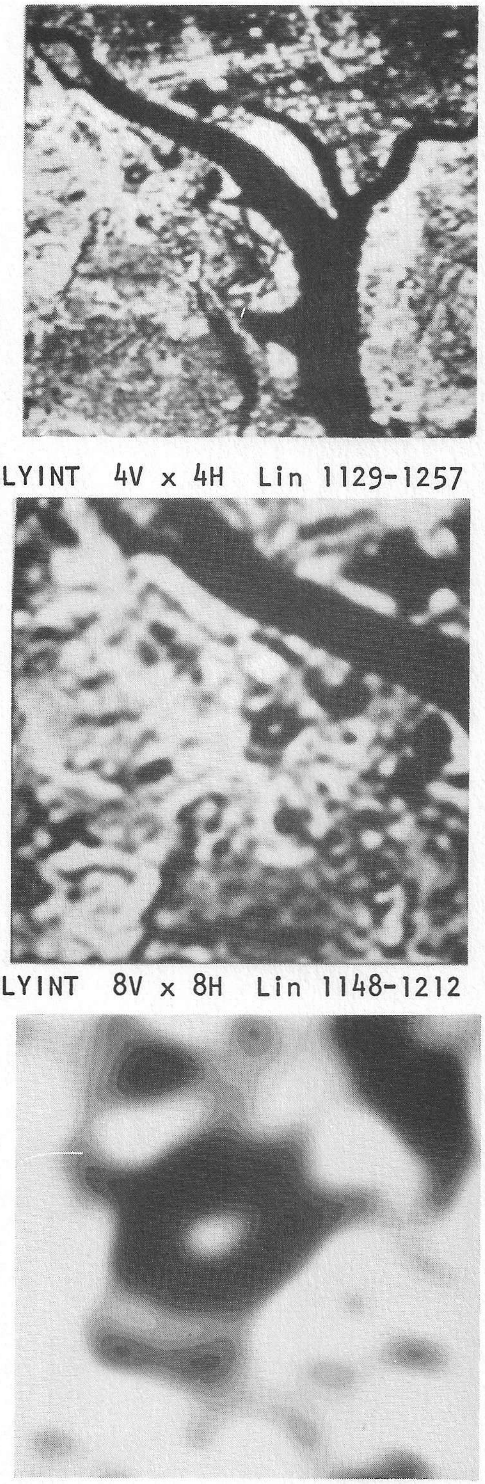
A number of examples of interpolated ERTS imagery are shown in the accompanying figures. Using POLYINT, the area in the vicinity of the Pentagon Building is shown with magnification of 4 x 4, 8 x 8 and 32 x 32 in Figure 3. The smooth transition between points of greatly differing contrast is evident. No overshoot or ghosting is evident in these images.

Figure 4 shows the Pentagon interpolated with SINCINT. In Figure 4(a) the image is displayed with 16 gray levels and is seen to be very similar to the comparable image obtained using POLYINT and shown in Figure 3(c). Figure 4(b) and 4(c) show the effect of reducing the number of gray levels displayed and illustrate the type of contouring that can be obtained in this manner.

Figure 5 shows the same area interpolated using TRIGINT. However, in this case a different magnification is used in the vertical and horizontal directions to correct for the scale differences of ERTS images in these directions. The scaling ratio used is 38:27 which is quite close to the correct value of 79:56 specified in the ERTS Data Users Handbook. Figure 5(a) shows 16 gray levels which Figure 5(b) and 5(c) are binary images with the thresholds set at the 4th and 3rd gray levels respectively of the original image.

Figure 6 shows a comparison of TRIGINT and SINCINT using a different ERTS frame. Figure 6(a) is the original data set while 6(b), 6(c) and 6(d) are 4 x 4 magnifications using repeated points, TRIGINT and SINCINT respectively. Figure 4(d) has a somewhat different appearance than 4(c). The texture appears to be mottled somewhat. This is a general characteristic of images interpolated by SINCINT and has not been satisfactorily explained as yet. One possibility is that the SINCINT process is providing a type of enhancement in which adjacent points are more completely separated than in the other kinds of interpolation.

It can be concluded from examination of the above examples of interpolated imagery that the scene appears to be enlarged without introducing any major changes in its appearance. In many cases details of the scene are more easily discerned.

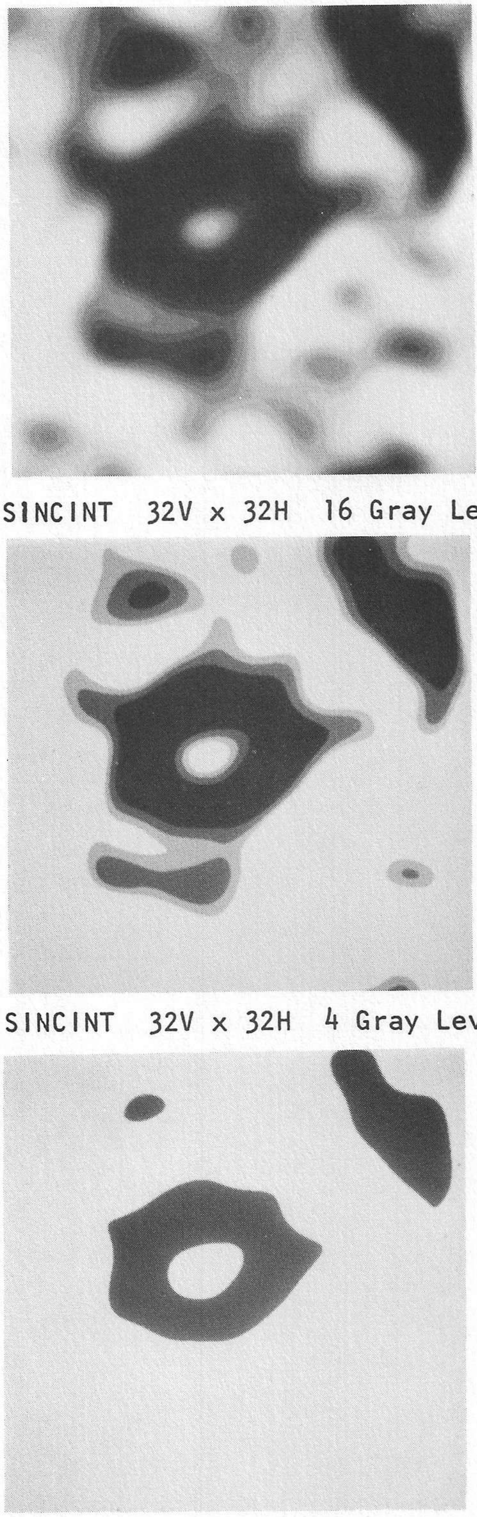


(a) POLYINT 4V x 4H Lin 1129-1257 Col 1217-1345

(b) POLYINT 8V x 8H Lin 1148-1212 Col 1213-1277

(c) POLYINT 32V x 32H Lin 1172-1188 Col 1242-1258

Fig. 3 Pentagon
Run 72041900 10/11/72 Ch 3

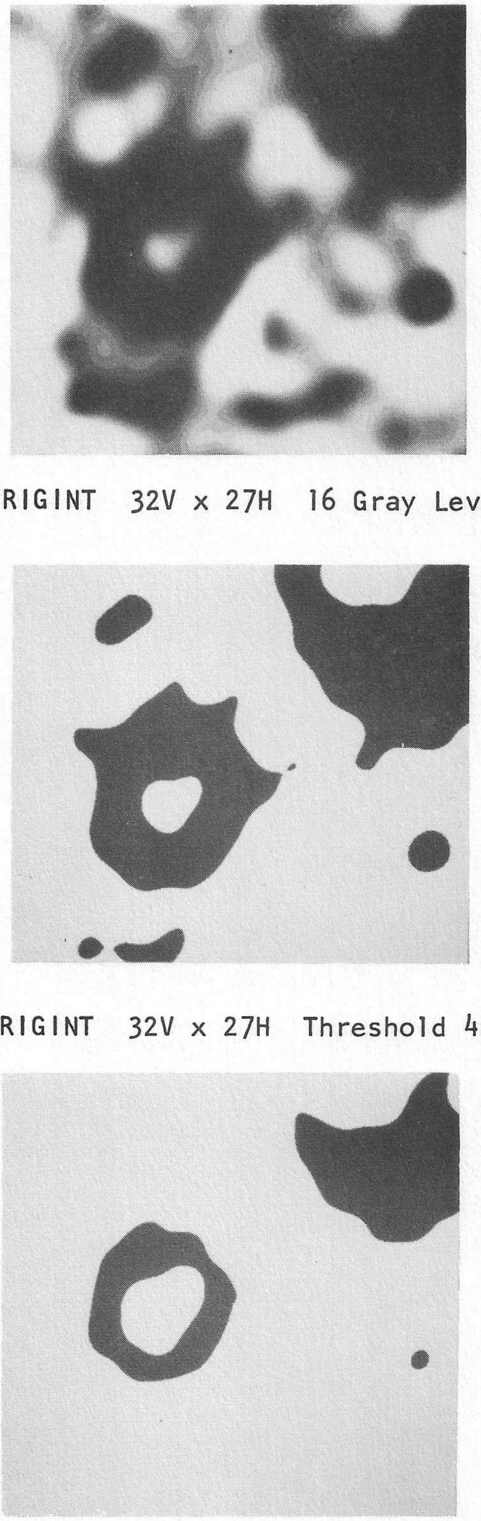


(a) SINCINT 32V x 32H 16 Gray Levels

(b) SINCINT 32V x 32H 4 Gray Levels

(c) SINCINT 32V x 32H 2 Gray Levels

Fig. 4 Pentagon
Run 72041900 10/11/72 Ch 3
Lin 1172-1182 Col 1242-1258

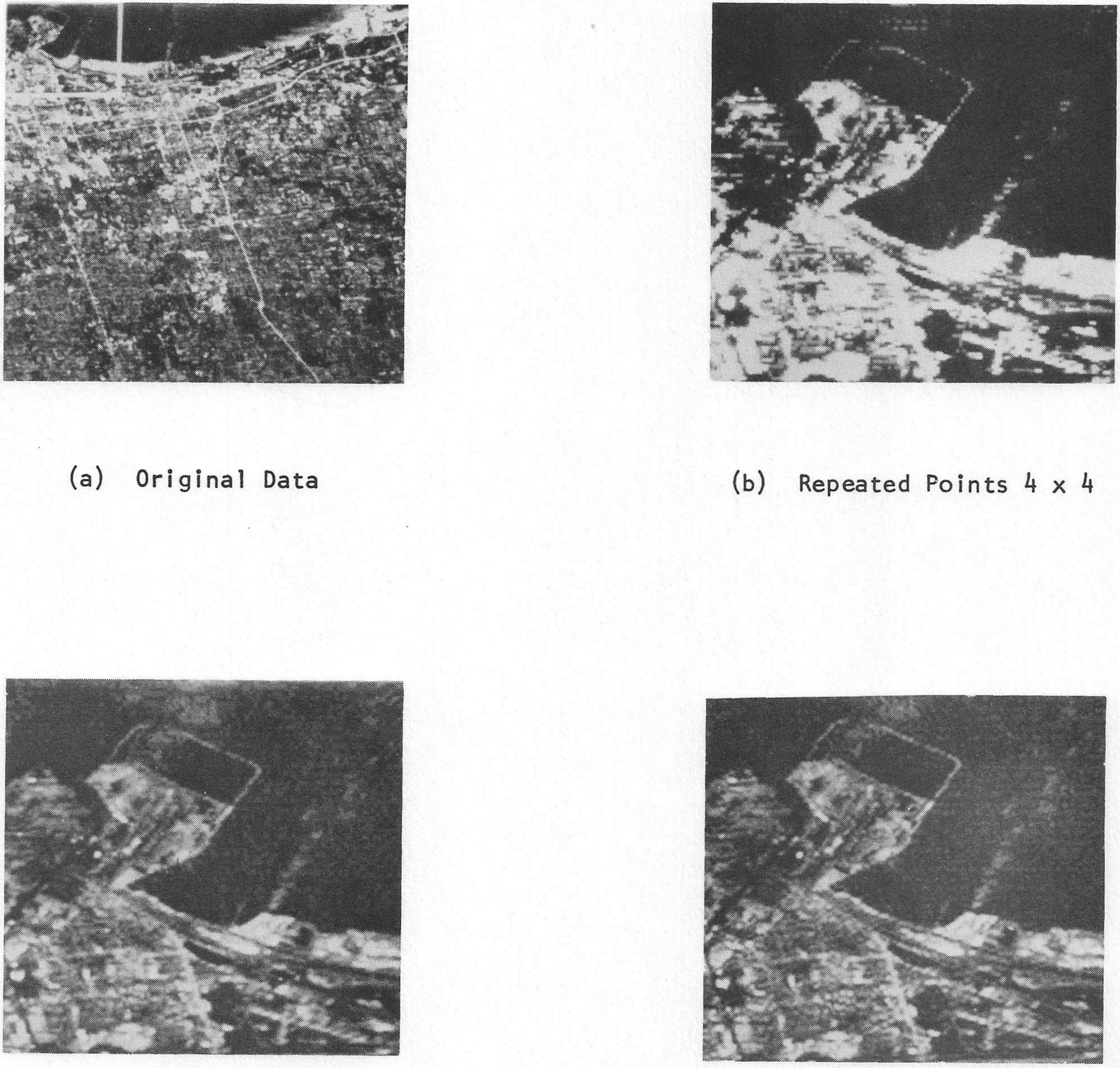


(a) TRIGINT 32V x 27H 16 Gray Levels

(b) TRIGINT 32V x 27H Threshold 4th Level

(c) TRIGINT 32V x 27H Threshold 3rd Level

Fig. 5 Pentagon
Run 72041900 10/11/72 Ch 3
Lin 1172-85 Col 1242-1260



(a) Original Data

(b) Repeated Points 4 x 4

(c) TRIGINT 4 x 4

(d) SINCINT 4 x 4

Fig. 6 Calument Area Run 72059900 Chan. 2
Line 1326-1453 Col 1360-1487