DISTORTION-FREE MAPPING OF VISSR IMAGERY DATA FROM GEOSYNCHRONOUS SATELLITES

F. K. Chan

Scientific Analysts and Consultants, Inc.

4114 Heathfield Road, Rockville, Md. 20853

ABSTRACT

Analysis has been performed for mapping VISSR imagery data so as to eliminate all geometrical distortions. The formulation is rigorous and includes all misalignment angles of the VISSR, the sun sensor and the instantaneous spin axis with the satellite's body axis. It also includes the effects due to the motion of the satellite's suborbital point. All the mapping equations for distortion removal are reduced to simplest forms, and all the algorithms are optimized as much as possible.

An approach is then formulated for implementing these algorithms for in-line operational use. It covers the computations involved in determining benchmarks, the interpolation methodology for filling in the points interspaced between benchmarks, and the correction procedure for computing the radiometric values at the center of the pixel in the distortion-free image. It is also concerned with the time requirements, data storage, and output data accuracy. With the present microprocessor technology, it is concluded that this in-line distortion removal is possible in real-time processing of infra-red but not visible VISSR imagery data.

This work was supported by NOAA Contract Nos: 01-8-M01-1864 and NA-79KAC-00026

SECTION 1 - INTRODUCTION

In the Visible and Infra-red Spin Scan Radiometer (VISSR) data obtained from the present geosynchronous satellites, distortions are observed in the images of the earth. As illustrated in Figure 1.1 which is exaggerated for clarity, these image deformations appear as vertical compression and expansion of the image, non-vertical alignment of the North and South Poles, and multi-representation of some points or omission of other points.

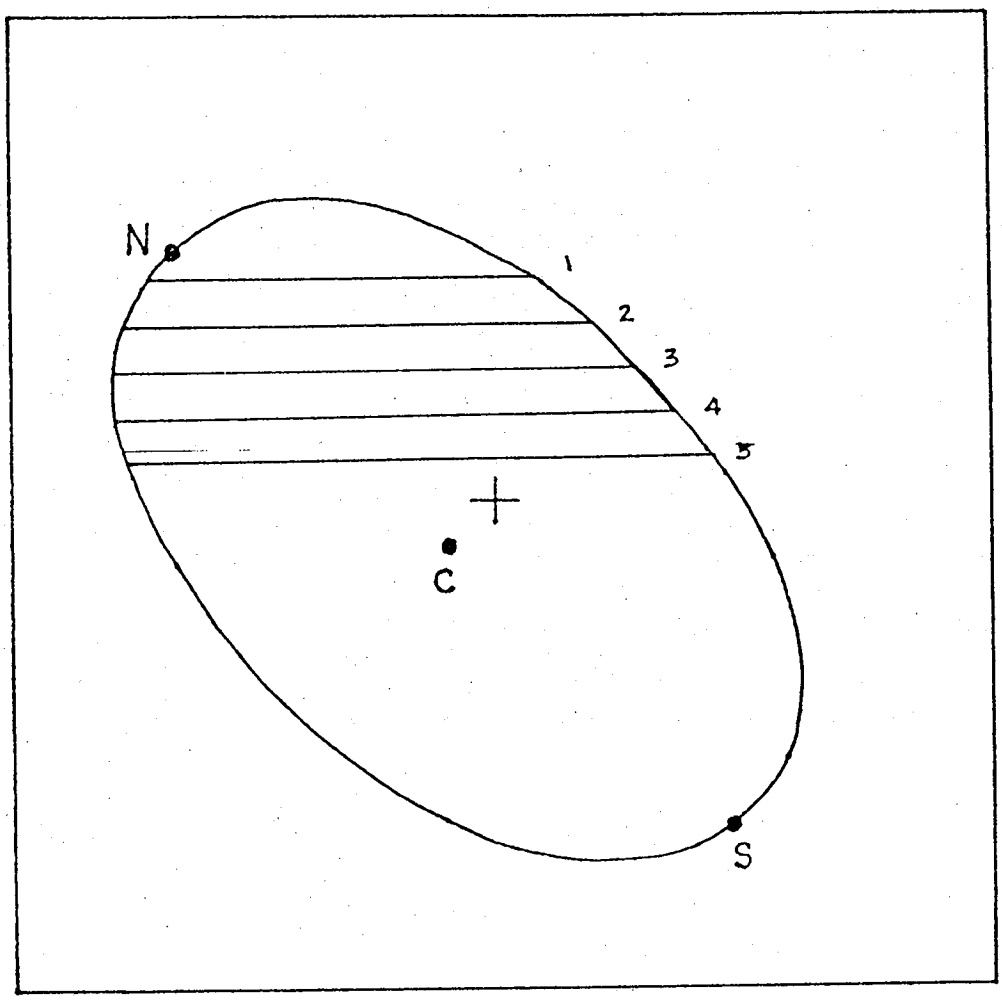


Figure 1.1 Exaggerated VISSR Image of Earth

These distortions may well be explained by considering Figure 1.2 which illustrates the same scan-lines on the projection plane of the earth. Again, for clarity, these scan-lines are depicted to be non-parallel and unevenly spaced to a degree more so than the realistic cases.

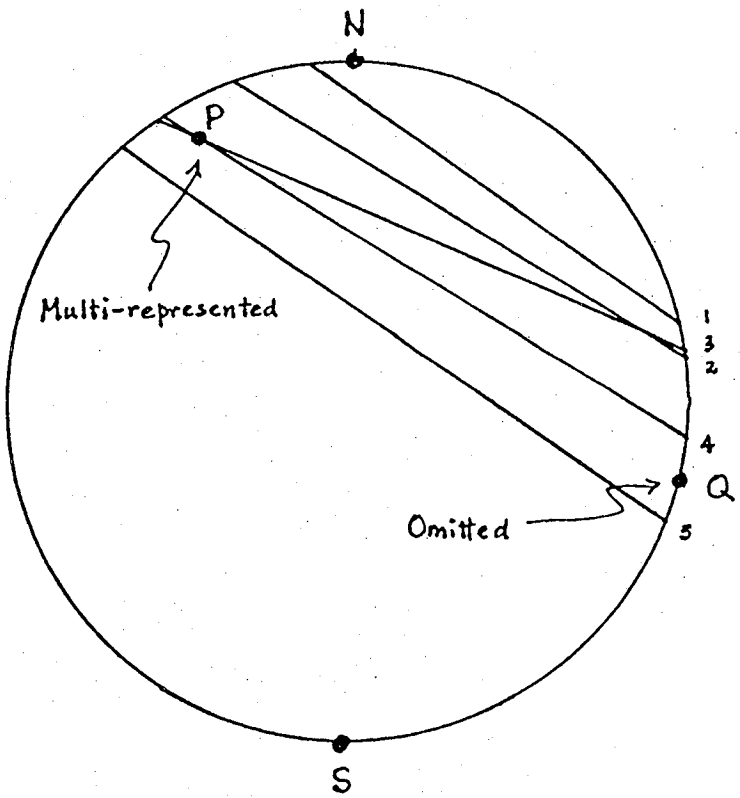
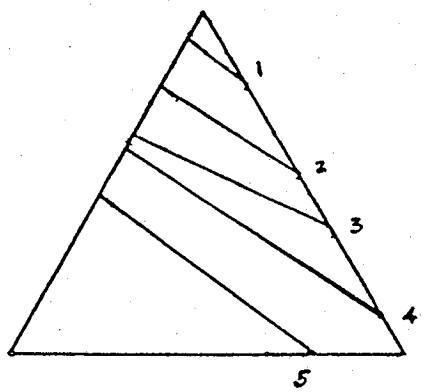
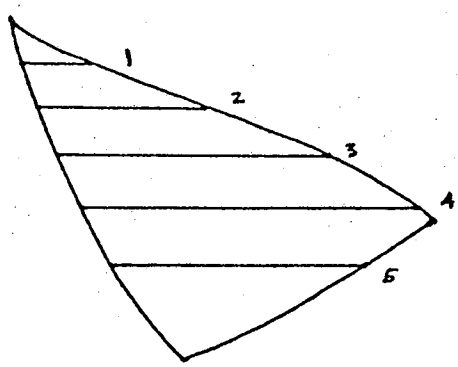


Figure 1.2 Projection Plane Image of Earth

If one were to relate these two images, one would find, for example, that a triangular figure in the projection plane image becomes distorted into a curved figure in the VISSR image. This is illustrated in Figure 1.3 which is obtained by superimposing a triangle on Figure 1.2 and then mapping it onto Figure 1.1.



Projection Plane Image



VISSR Image

Figure 1.3 Distortion of Image

The general causes for these image deformations may be broadly classified as follows:

1. Orbit not circular and equatorial
2. Spin axis not perpendicular to orbital plane
3. Misalignment of the VISSR, the sun sensor, and the instantaneous spin axis with the satellite's body axis
4. Biases due to varying sun size and varying sun elevation effect on threshold of the sun sensor triggering.

To remove these distortions, it is necessary to include all the above factors in the formulation of the mapping equations. However, it is feasible to consider only the first three. The corresponding equations have been derived in Reference 1 in which it was convenient to introduce the following coordinate systems:

The Inertial System: This is well-known and is defined such that the x_I-axis is in the direction of the vernal equinox, the z_I-axis is perpendicular to the equatorial plane (in the direction of the North Pole), and the y_I-axis is given by $\hat{y}_I = \hat{z}_I \times \hat{x}_I$.

The Body System: This system is defined such that the z_B-axis is along the longitudinal axis of the satellite, the x_B-axis is the intersection line between the VISSR stepping plane and the plane perpendicular to the z_B-axis, and the y_B-axis is given by $\hat{y}_B = \hat{z}_B \times \hat{x}_B$.

The VISSR System: This system is defined such that the x_V-axis is in the direction of the mid-scan, the z_V-axis is perpendicular to the x_V-axis and lies in the VISSR stepping plane, and the y_V-axis is given by $\hat{y}_V = \hat{z}_V \times \hat{x}_V$.

The Sun Sensor System: This system is defined such that the x_U-axis

is in the direction of the bisector of the angle fanned out by the sun sensor, the z_U-axis is perpendicular to the x_U-axis and lies in the sun sensor fan plane, and the y_U-axis is given by $\hat{y}_U = \hat{z}_U \times \hat{x}_U$.

The Spin System: Let $\hat{z}_S$ denote the unit spin axis vector around which the satellite is instantaneously rotating. Let $\vec{r}'$ denote the position vector of the satellite. Then, the y_S-axis and the x_S-axis are respectively defined by

$$\hat{y}_S = \overbrace{\vec{r}' \times \hat{z}_S} \tag{1.1}$$

$$\hat{x}_S = \hat{y}_S \times \hat{z}_S \tag{1.2}$$

The Auxiliary System: In this system, illustrated in Figure 1.4, the unit base vectors are defined by the following equations:

$$\hat{x}_A = -\hat{r}' \tag{1.3}$$

$$\hat{y}_A = \overbrace{\hat{z}_I \times \hat{x}_A} \tag{1.4}$$

$$\hat{z}_A = \hat{x}_A \times \hat{y}_A \tag{1.5}$$

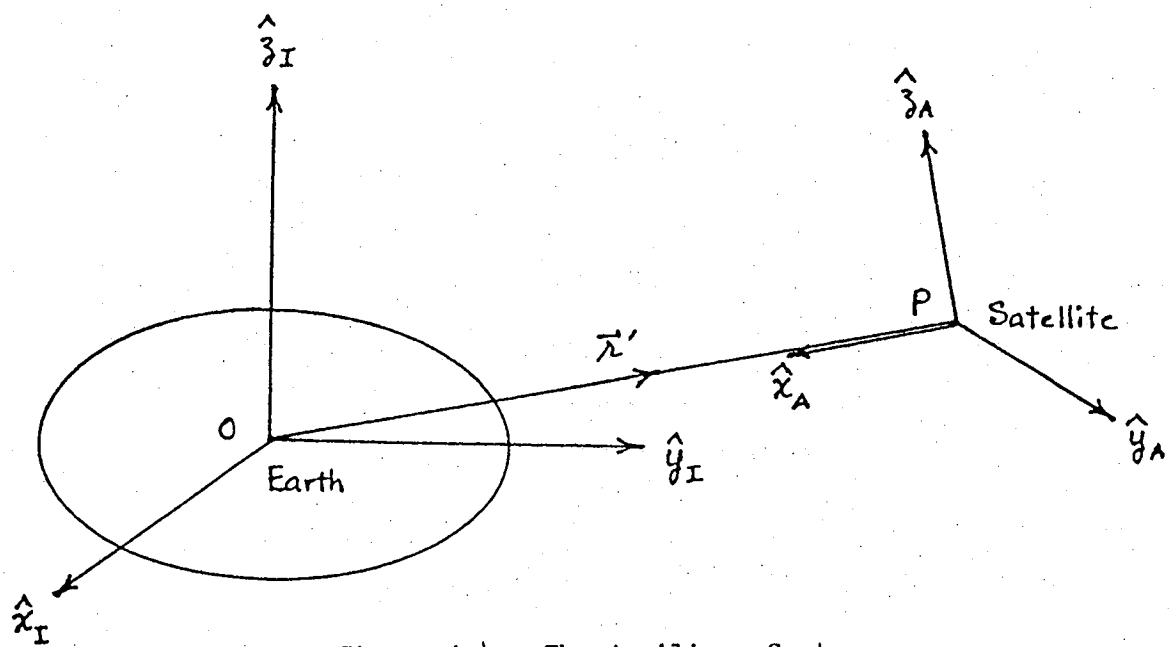


Figure 1.4 - The Auxiliary System

The Normalized System: Consider a system, referred to as the normalized system, as illustrated in Figure 1.5. The origin R of this system is defined at a point $\vec{r}$ on the earth's equatorial plane and fixed in the earth's rotating system. The satellite P at point $\vec{r}'$, however, is not necessarily on the earth's equatorial plane or fixed in the earth's rotating system. In this normalized system, the unit base vectors are defined by the following equations:

$$\hat{x}_N = -\hat{r} \tag{1.6}$$

$$\hat{z}_N = \hat{z}_I \tag{1.7}$$

$$\hat{y}_N = \hat{z}_N \times \hat{x}_N \tag{1.8}$$

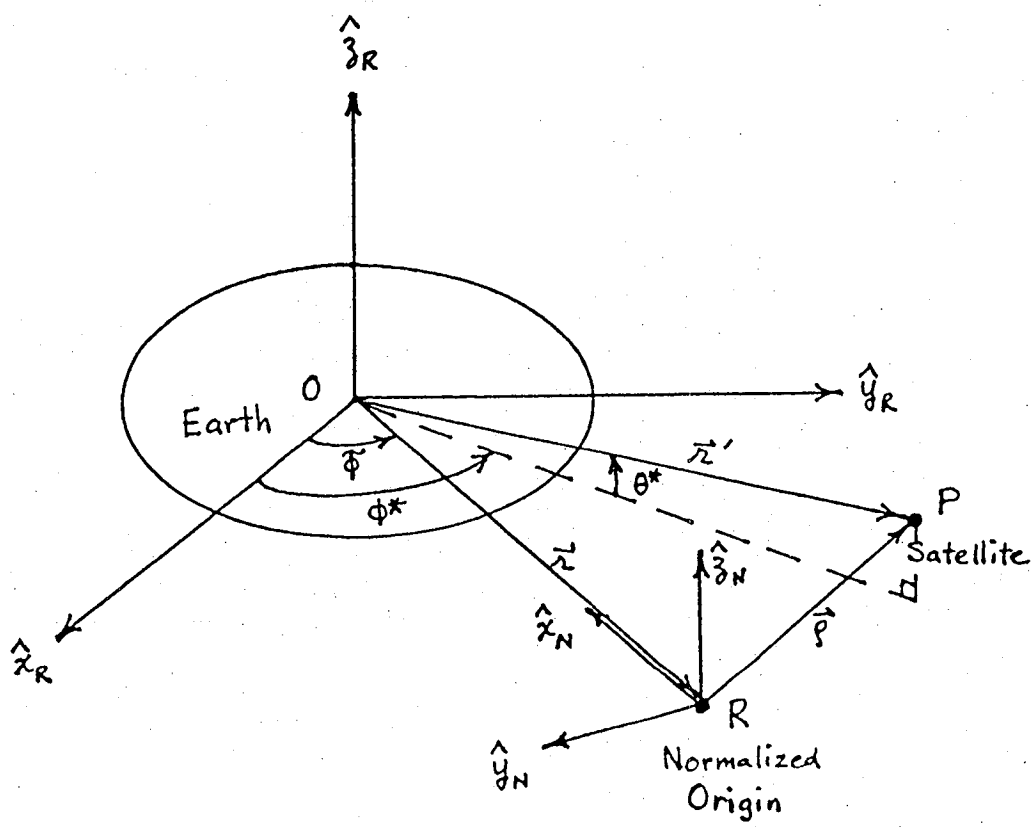


Figure 1.5 - The Normalized System

In the present study, the results obtained in Reference 1 are used to formulate algorithms for mapping the data coming out of the Synchronous Data Buffer (SDB) so as to obtain a distortion-free imagery. Moreover, this rectified imagery also has the desirable feature that it is referred to a normalized satellite position which is therefore the same for all imageries. Thus, if the distortion-free mapping is performed in-line during data processing, the transmitted VISSR data will provide a uniformly compatible data base for all users in their scientific work. Furthermore, it will also facilitate in the future development of a composite data base for different kinds of data obtained from various satellites.

Mapping of the data may be further optimized the use of interpolation with the aid of appropriately chosen benchmarks. Section 2 deals with the computation of these benchmarks, while Section 3 covers the interpolation methodology for filling in the points interspaced between benchmarks. Section 4 is concerned with the correction procedure for computing the radiometric values at the center of the pixel in the distortion-free image. Section 5 discusses the time requirements, data storage and output data accuracy. Section 6 summarizes the results of this study.

SECTION 2 - BENCHMARK COMPUTATIONS

This section deals with the computation of benchmark locations in the normalized distortion-free coordinate system. It discusses the relevant input parameters, computational equations, number of compuationál operations, and the requisite partial derivatives.

2.1 Input Parameters

The relevant input parameters are listed below:

N_η = scan number

N_{ξ_s} = sample number

M_η = mid-scan number

M_{ξ_s} = mid-sample number

$\Delta\eta$ = scan angular width

$\Delta\xi$ = sample angular width

ψ_s = scan angular bias (line bias)

$\vec{r}'$ = orbital position of satellite

α = right ascension of spin axis

δ = declination of spin axis

$\vec{r}$ = normalized position of satellite

a = earth's semi-major axis

c = earth's semi-minor axis

For convenience, two parameters dependent on the above are defined as follows:

$$\varepsilon \equiv \frac{a^2 - c^2}{c^2}$$

$$K^* \equiv \frac{r}{a} \frac{1}{\Delta\xi} \tan^{-1}\left(\frac{a}{r}\right)$$

2.2 Computational Equations

The following equations for computing benchmark coordinates have been extracted from Reference 1. They have been simplified and are listed below in the proper sequence for usage. The exact definitions of cursory intermediate variables may be obtained from the original report.

$$\eta = (M_\eta - N_\eta)\,\Delta\eta \tag{2.1}$$

$$\xi_s = (M_{\xi_s} - N_{\xi_s})\,\Delta\xi \tag{2.2}$$

$$\eta_s = \eta + \psi_s \tag{2.3}$$

$$\hat{u}_s = c_{\eta_s} c_{\xi_s} \hat{x}_s + c_{\eta_s} s_{\xi_s} \hat{y}_s + s_{\eta_s} \hat{z}_s \tag{2.4}$$

$$k = \frac{1}{\sqrt{[1 - (\hat{\lambda}' \cdot \hat{z}_s)^2]}} \tag{2.5}$$

$$k' = \frac{1}{\sqrt{[1 - (\hat{\lambda}' \cdot \hat{z}_I)^2]}} \tag{2.6}$$

$$(T_{13})_{SA} = -(\hat{\imath}' \cdot \hat{\jmath}_S) \tag{2.7}$$

$$(T_{23})_{SA} = k' \hat{\imath}' \cdot (\hat{\jmath}_I \times \hat{\jmath}_S) \tag{2.8}$$

$$(T_{33})_{SA} = k' \left[(\hat{\jmath}_S \cdot \hat{\jmath}_I) - (\hat{\imath}' \cdot \hat{\jmath}_S)(\hat{\imath}' \cdot \hat{\jmath}_I) \right] \tag{2.9}$$

$$T_{SA} = \begin{bmatrix} \frac{1}{k} & 0 & (T_{13})_{SA} \\ -k(T_{13})_{SA}(T_{23})_{SA} & k(T_{33})_{SA} & (T_{23})_{SA} \\ -k(T_{13})_{SA}(T_{33})_{SA} & -k(T_{23})_{SA} & (T_{33})_{SA} \end{bmatrix} \tag{2.10}$$

$$(T_{11})_{AN} = C_{\theta^*} C_{\phi^* - \tilde{\phi}} \tag{2.11}$$

$$(T_{21})_{AN} = C_{\theta^*} S_{\phi^* - \tilde{\phi}} \tag{2.12}$$

$$(T_{31})_{AN} = -S_{\theta^*} \tag{2.13}$$

$$T_{AN} = \begin{bmatrix} (T_{11})_{AN} & -k'(T_{21})_{AN} & -k'(T_{11})_{AN}(T_{31})_{AN} \\ (T_{21})_{AN} & k'(T_{11})_{AN} & -k'(T_{21})_{AN}(T_{31})_{AN} \\ (T_{31})_{AN} & 0 & \frac{1}{k'} \end{bmatrix} \tag{2.14}$$

$$\hat{z}_s = c_\alpha c_\delta \hat{x}_I + s_\alpha c_\delta \hat{y}_I + s_\delta \hat{z}_I \qquad (2.15)$$

$$\hat{x}_s = k\left[(\hat{r}' \cdot \hat{z}_s)\hat{z}_s - \hat{r}'\right] \qquad (2.16)$$

$$\hat{y}_s = \hat{z}_s \times \hat{x}_s \qquad (2.17)$$

$$T_{sI} = \begin{bmatrix} \hat{x}_s \cdot \hat{x}_I & \hat{y}_s \cdot \hat{x}_I & \hat{z}_s \cdot \hat{x}_I \\ \hat{x}_s \cdot \hat{y}_I & \hat{y}_s \cdot \hat{y}_I & \hat{z}_s \cdot \hat{y}_I \\ \hat{x}_s \cdot \hat{z}_I & \hat{y}_s \cdot \hat{z}_I & \hat{z}_s \cdot \hat{z}_I \end{bmatrix} \qquad (2.18)$$

$$\hat{u}_I = T_{sI}\, \hat{u}_s \qquad (2.19)$$

$$A = 1 + \varepsilon (u_z)_I^2 \qquad (2.20)$$

$$B = \vec{r}_I' \cdot \hat{u}_I + \varepsilon (r_z')_I (u_z)_I \qquad (2.21)$$

$$C = \vec{r}_I' \cdot \vec{r}_I' + \varepsilon (r_z')_I^2 - a^2 \qquad (2.22)$$

$$\lambda = \frac{-B - \sqrt{(B^2 - AC)}}{A} \qquad (2.23)$$

$$\hat{u}_N = T_{AN}\, T_{SA}\, \hat{u}_S \tag{2.24}$$

$$\vec{\rho}_N = (r - r' c_{\theta^*} c_{\phi^* - \phi})\hat{x}_N - r' c_{\theta^*} s_{\phi^* - \phi}\, \hat{y}_N + r' s_{\theta^*}\, \hat{z}_N \tag{2.25}$$

$$\vec{\rho}_N' = \vec{\rho}_N + \lambda \hat{u}_N \tag{2.26}$$

$$I^* = K^* \frac{(\rho_y')_N}{(\rho_x')_N} \tag{2.27}$$

$$J^* = K^* \frac{(\rho_z')_N}{(\rho_x')_N} \tag{2.28}$$

I^* and J^* are the coordinates in the normalized distortion-free system. At this stage, for the sake of greater accuracy in subsequent computations, it is preferable not to digitize them.

2.3 Computational Operations

For each benchmark, it may be verified that the computations in equations (2.1) - (2.28) may be achieved by performing:

51 additions

22 divisions

114 multiplications

4 divisions

11 trigonometric function evaluations

2 square root evaluations

Assuming that the following times are required:

Operation	Time (micro-seconds)
Addition	1.5
Subtraction	1.5
Multiplication	6
Division	11
Trigonometric function evaluation	50
Square root evaluation	50

it is seen that about 1487 microseconds are required for each benchmark computation.

2.4 Partial Derivatives

For convenience, it is desirable to choose the set of benchmarks so that they form a rectangular grid in the (N_{ξ_s}, N_η) -space as illustrated in Figure 2.1.

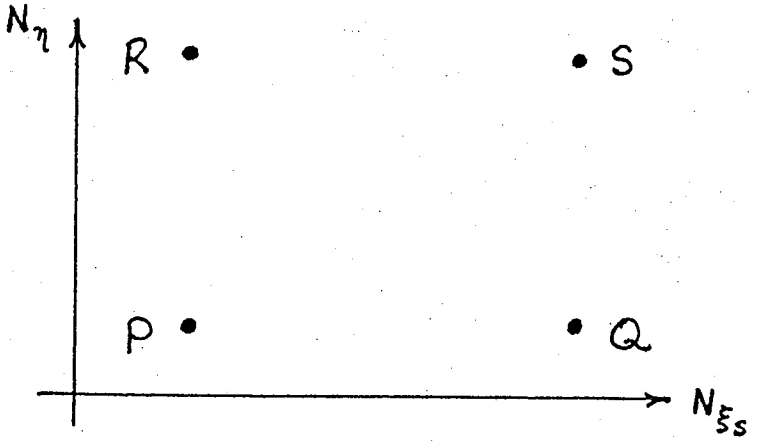


Figure 2.1 - Benchmarks in (N_{ξ_s}, N_η) - Space

Then, it is obvious that the partial derivatives of I^* and J^* with respect to N_{ξ_s} and N_η may be approximated by

$$\left[\left(\frac{\partial I^*}{\partial N_{\xi_s}}\right)_{N_\eta}\right]_P \simeq \frac{(I^*)_Q - (I^*)_P}{(N_{\xi_s})_Q - (N_{\xi_s})_P} \qquad (2.29)$$

$$\left[\left(\frac{\partial J^*}{\partial N_{\xi_s}}\right)_{N_\eta}\right]_P \simeq \frac{(J^*)_Q - (J^*)_P}{(N_{\xi_s})_Q - (N_{\xi_s})_P} \qquad (2.30)$$

$$\left[\left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi s}}\right]_P \simeq \frac{(I^*)_R - (I^*)_P}{(N_\eta)_R - (N_\eta)_P} \qquad (2.31)$$

$$\left[\left(\frac{\partial J^*}{\partial N_\eta}\right)_{N_{\xi s}}\right]_P \simeq \frac{(J^*)_R - (J^*)_P}{(N_\eta)_R - (N_\eta)_P} \qquad (2.32)$$

For each benchmark, the four associated partial derivatives require 6 subtractions and 4 divisions. These operations consume about 53 microseconds.

These partial derivatives are used later in the method of interpolation for mapping points interspaced betweeen the benchmarks.

SECTION 3 - INTERPOLATION COMPUTATIONS

This section is concerned with the mapping of points which do not coincide with the chosen set of benchmarks. It discusses the input data, interpolation methodology, and number of computational operations.

3.1 Input Data

The input data consists of the coordinates (I^*, J^*) and their four associated partial derivatives for each benchmark. This information has already been obtained in Section 2.

3.2 Interpolation Methodology

Suppose there are m interspaced points between the horizontal benchmarks, and n interspaced points between the vertical benchmarks. Figure 3.1 illustrates a basic unit comprising benchmarks (denoted by solid circles) and interspaced points (denoted by open circles), Thus, there are $(m + 1)(n + 1)$ points altogether in a basic unit.

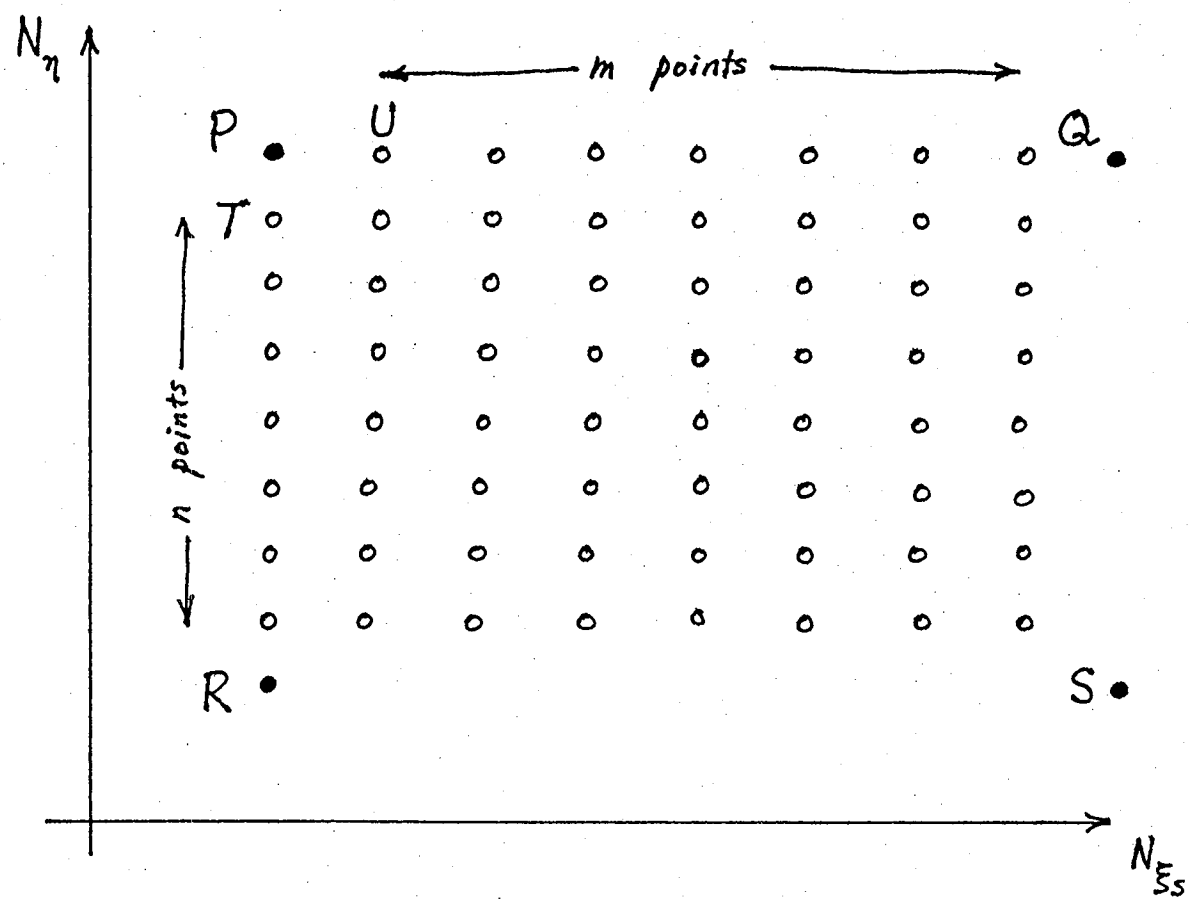


Figure 3.1 - Benchmarks for Interpolation

It is noted that the I^* coordinate of the point T directly below P is given by

$$(I^*)_T = (I^*)_P + \Delta I^* \tag{3.1}$$

But

$$\Delta I^* \simeq \left[\left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi_s}}\right]_P \Delta N_\eta \tag{3.2}$$

and

$$\Delta N_\eta = +1 \,. \tag{3.3}$$

Therefore, it follows that

$$(I^*)_T = (I^*)_P + \left[\left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi s}}\right]_P \tag{3.4}$$

A similar equation holds for any point and the point directly below it. Hence, the n interspaced points between P and R may be mapped by the following iterative algorithm:

Let

$$I_0^* = (I^*)_P \tag{3.5}$$

$$J_0^* = (J^*)_P \tag{3.6}$$

Then, perform the following computations

$$\left.\begin{aligned} I^*_{\ell+1} &= I^*_\ell + \left[\left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi s}}\right]_P \qquad (3.7)\\ J^*_{\ell+1} &= J^*_\ell + \left[\left(\frac{\partial J^*}{\partial N_\eta}\right)_{N_{\xi s}}\right]_P \qquad (3.8) \end{aligned}\right\} \ell = 0, 1, 2, \ldots n$$

Similarly, it is noted that the I^* coordinate of the point U directly to the right of P is given by

$$(I^*)_U = (I^*)_P + \Delta I^* \tag{3.9}$$

But
$$\Delta I^* \simeq \left[\left(\frac{\partial I^*}{\partial N_{\xi s}}\right)_{N_\eta}\right]_P \Delta N_{\xi s} \tag{3.10}$$

and
$$\Delta N_{\xi s} = +1 \tag{3.11}$$

Therefore, it follows that
$$(I^*)_U = (I^*)_P + \left[\left(\frac{\partial I^*}{\partial N_{\xi s}}\right)_{N_\eta}\right]_P \tag{3.12}$$

Again, a similar equation holds for any point and the point directly to the right of it. Hence, the m(n + 1) interspaced points between the columns PR and QS may be mapped by the following iterative algorithm:

Let
$$\left.\begin{aligned} I^*_{0,\ell} &= I^*_\ell \\ J^*_{0,\ell} &= J^*_\ell \end{aligned}\right\} \ell = 0, 1, 2, \ldots n \tag{3.13, 3.14}$$

Then, perform the following computations
$$\left.\begin{aligned} I^*_{(k+1),\ell} &= I^*_{k,\ell} + \left[\left(\frac{\partial I^*}{\partial N_{\xi s}}\right)_{N_\eta}\right]_P \\ J^*_{(k+1),\ell} &= J^*_{k,\ell} + \left[\left(\frac{\partial J^*}{\partial N_{\xi s}}\right)_{N_\eta}\right]_P \end{aligned}\right\} k = 0, 1, 2, \ldots n \tag{3.15, 3.16}$$

After these computations, it is now permissible to digitize I^* and J^* to obtain the rounded integer values I and J.

3.3 Computation Operations

For each interspaced point, it is seen that the computations in equations (3.7) and (3.8) or those in equations (3.15) and (3.16) require 2 additions. Assuming a time of 1.5 microseconds for each operation, therefore about 3 microseconds are required to map each point by interpolation. Allowance is also to be make for converting two real numbers to integer values for each point. This will probably increase the time requirement by a factor of 2 so that about 6 microseconds are required for each point.

SECTION 4 - RADIOMETRIC COMPUTATIONS

This sections considers the methodology of correcting the radiometric values so as to reflect a more realistic value at the center of the pixel in the distortion-free image. It discusses the input data, correction methodology, and number of computational operations.

4.1 Input Data

For each point, the input data consists of the following and may be obtained either from the SDB output data stream or has already been obtained in Section 3:

N_η = scan number

N_{ξ_s} = sample number

$R(N_{\xi_s}, N_\eta)$ = radiometric value of N_η th scan and N_{ξ_s} th sample

$R(N_{\xi_s} - 1, N_\eta)$ = radiometric value of N_η th scan and $(N_{\xi_s} - 1)$th sample

$R(N_{\xi_s}, N_\eta - 1)$ = raidometric value of $(N_\eta - 1)$th scan and N_{ξ_s} th sample

I^* = interpolated horizontal coordinate of (N_{ξ_s}, N_η) in distortion-free system

J^* = interpolated vertical coordinate of (N_{ξ_s}, N_η) in distortion-free system

I = rounded integer value of I^*

J = rounded integer value of J^*

4.2 Correction Methodology

The radiometric value $R(I, J)$ may be obtained from the value $R(I^*, J^*)$ by using the Taylor's series expansion

$$R(I,J) = R(I^*, J^*) + \left(\frac{\partial R}{\partial I^*}\right)_{J^*} (I - I^*) + \left(\frac{\partial R}{\partial J^*}\right)_{I^*} (J - J^*). \quad (4.1)$$

The partial derivatives $\left(\frac{\partial R}{\partial I^*}\right)_{J^*}$ and $\left(\frac{\partial R}{\partial J^*}\right)_{I^*}$ may be written as

$$\left(\frac{\partial R}{\partial I^*}\right)_{J^*} = \left(\frac{\partial R}{\partial N_{\xi_s}}\right)_{N_\eta} \left(\frac{\partial N_{\xi_s}}{\partial I^*}\right)_{J^*} + \left(\frac{\partial R}{\partial N_\eta}\right)_{N_{\xi_s}} \left(\frac{\partial N_\eta}{\partial I^*}\right)_{J^*} \quad (4.2)$$

$$\left(\frac{\partial R}{\partial J^*}\right)_{I^*} = \left(\frac{\partial R}{\partial N_{\xi_s}}\right)_{N_\eta} \left(\frac{\partial N_{\xi_s}}{\partial J^*}\right)_{I^*} + \left(\frac{\partial R}{\partial N_\eta}\right)_{N_{\xi_s}} \left(\frac{\partial N_\eta}{\partial J^*}\right)_{I^*} \quad (4.3)$$

The partial derivatives $\left(\frac{\partial R}{\partial N_{\xi s}}\right)_{N_\eta}$ and $\left(\frac{\partial R}{\partial N_\eta}\right)_{N_{\xi s}}$ may be approximated by

$$\left(\frac{\partial R}{\partial N_{\xi s}}\right)_{N_\eta} \simeq \left(\frac{\Delta R}{\Delta N_{\xi s}}\right)_{N_\eta} = R(N_{\xi s}, N_\eta) - R(N_{\xi s}-1, N_\eta) \qquad (4.4)$$

$$\left(\frac{\partial R}{\partial N_\eta}\right)_{N_{\xi s}} \simeq \left(\frac{\Delta R}{\Delta N_\eta}\right)_{N_{\xi s}} = R(N_{\xi s}, N_\eta) - R(N_{\xi s}, N_\eta - 1). \qquad (4.5)$$

The four remaining partial derivatives on the RHS of equations (4.2) and (4.3) may be obtained as follows: Let A and B be matrices defined by

$$A = \begin{bmatrix} \left(\frac{\partial N_{\xi s}}{\partial I^*}\right)_{J^*} & \left(\frac{\partial N_{\xi s}}{\partial J^*}\right)_{I^*} \\ \left(\frac{\partial N_\eta}{\partial I^*}\right)_{J^*} & \left(\frac{\partial N_\eta}{\partial J^*}\right)_{I^*} \end{bmatrix} \qquad (4.6)$$

$$B = \begin{bmatrix} \left(\frac{\partial I^*}{\partial N_{\xi s}}\right)_{N_\eta} & \left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi s}} \\ \left(\frac{\partial J^*}{\partial N_{\xi s}}\right)_{N_\eta} & \left(\frac{\partial J^*}{\partial N_\eta}\right)_{N_{\xi s}} \end{bmatrix} \qquad (4.7)$$

Then, from the theory of mathematical transformations, we have

$$A = B^{-1} \tag{4.8}$$

which explicitly yields

$$\left(\frac{\partial N_{\xi_s}}{\partial I^*}\right)_{J^*} = \frac{1}{C}\left(\frac{\partial J^*}{\partial N_\eta}\right)_{N_{\xi_s}} \tag{4.9}$$

$$\left(\frac{\partial N_{\xi_s}}{\partial J^*}\right)_{I^*} = -\frac{1}{C}\left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi_s}} \tag{4.10}$$

$$\left(\frac{\partial N_\eta}{\partial I^*}\right)_{J^*} = -\frac{1}{C}\left(\frac{\partial J^*}{\partial N_{\xi_s}}\right)_{N_\eta} \tag{4.11}$$

$$\left(\frac{\partial N_\eta}{\partial J^*}\right)_{I^*} = \frac{1}{C}\left(\frac{\partial I^*}{\partial N_{\xi_s}}\right)_{N_\eta} \tag{4.12}$$

where

$$C = \left(\frac{\partial I^*}{\partial N_{\xi_s}}\right)_{N_\eta}\left(\frac{\partial J^*}{\partial N_\eta}\right)_{N_{\xi_s}} - \left(\frac{\partial J^*}{\partial N_{\xi_s}}\right)_{N_\eta}\left(\frac{\partial I^*}{\partial N_\eta}\right)_{N_{\xi_s}} \tag{4.13}$$

The partial derivatives appearing on the RHS of equations (4.9) - (4.13) may be obtained from equations (2.29) - (2.32), valid for a basic unit defined by benchmarks.

4.3 Computational Operations

For each point, it is seen that the computations in equations (4.1) - (4.5) require 4 additions, 6 subtractions and 6 multiplications. Assuming a time of 1.5 microseconds for each addition or subtraction , and 6 microseconds for each multiplication, therefore about 51 microseconds are required to correct the radiometric value for each point.

SECTION 5 - DISCUSSION

This section discusses relevant topics such as benchmark spacing, time requirements for mapping IR data, real and non-real time computations, data storage and buffering, input and output data accuracy, and computational accuracy requirements.

5.1 Benchmark Spacing

The IR samples have angular widths of about $0.01^{\circ} \times 0.005^{\circ}$ at the satellite position. This corresponds to a resolution of about 4 x 2 miles at the sub-satellite position on the earth's surface. In general, this resolution and the non-linearity of the mapping equations determine the requisite spacing of the benchmarks to be used for interpolation. The analytical approach to obtain this spacing involves comparatively complex mathematical analysis. Alternatively, it is also possible to obtain this value by actually performing the mapping munerically. At this stage, it is felt that the interpolation requirements can be met by choosing the IR benchmarks to be spaced 50 samples horizontally and 25 samples vertically. That is, it probably suffices to choose m = 50 and n = 25 in Section 3. In the full IR imagery, there are 1822 scans each containing 3822 samples. Consequently, about 5,600 benchmarks will be required.

5.2 Time Requirements

In Section 2, the computation of each benchmark and its associated partial derivatives requires about 1540 microseconds. Hence, a set of 5600 benchmarks requires about 8,624,000 microseconds $\simeq$ 8.6 seconds.

In Section 3, the mapping of each sample by interpolation requires about 6 microseconds. Therefore, an IR imagery of about 7×10^6 samples requires about 42 seconds. However, if the entire IR imagery is not to be mapped, then cropping out the edges will probably reduce time by a factor of 2/3 to yield a requirement of about 28 seconds.

In Section 4, the correction of radiometric value at the center of the pixel in the distortion-free image requires about 51 microseconds for each sample. Therefore, an IR imagery of about 7×10^6 samples requires about 357 seconds. Cropping will probably reduce this to about 238 seconds.

Consequently, about 399 seconds or 6.7 minutes will be needed to map the entire IR imagery comprising of coordinates and radiometric values of the samples. This time requirement drops to about 4.4 minutes if cropping is introduced.

If it is desired to map the visible imagery containing 1/2 x 1/2 mile samples, then the above times are increased by a factor of 32. Therefore, about 214 minutes will be required to map the entire imagery comprising of coordinates and radiometric values. If the edges are cropped out, then about 143 minutes will be needed.

5.3 Real and Non-Real Time Computations

From the discussion above, it is seen that it is possible to perform all the mapping computations in real-time in the case of IR imagery, and not possible in the case of visible imagery. However, in the latter case, the crucial point is whether the radiometric corrections are really necessary. If not, then the time requirements drops to 22.4 minutes for the entire imagery, and 14.9 minutes for the cropped imagery. Consequently, visible data-mapping becomes feasible in real-time.

Because the benchmark computation time is so small, it is desirable to perform the benchmark computations in real-time so that the relevant parameters may be easily extracted in-line from the data-stream coming out of the SDB.

5.4 Data Storage and Buffering

Because the imagery obtained from the SDB output data is distorted, it is necessary to store this data in a buffer before the distortion-free mapping can be performed. The buffer size may be estimated by allowing for a maximum 3° offset in the spin axis. Since the satellite is about 6.6 earth radii away, it may be verified that about 100 IR scan-lines (382,200 samples) to be buffered at a time. This will be sufficient to output a horizontal distortion-free line from end to end. In the case of visible data, the corresponding buffer will contain about 800 visible scan-lines (12,230,400 samples). If a realistic situation, the above numbers will probably be reduced by a factor of 3.

5.5 Input and Output Data Accuracy

The data coming out of the SDB will be used as input into the distortion-free mapping software system. The accuracy of this data may be roughly classified as perfect, normal or bad.

Perfect data corresponds to data having errors of less than one pixel (i. e. $\pm$ 2 km at the subsatellite point for IR data). The error in the output data from the distortion-free mapping is therefore determined by the pixel resolution of the benchmarks, the interpolation accuracy of interspaced points, and the correction accur-

acy of the radiometric values. Because the second and third factors depend on the benchmark-spacing, which in turn depends on the pixel resolution, therefore it is estimated that the error bound of the output data is about one pixel (i.e., $\pm$ 4 km for IR data).

Normal data corresponds to data having errors of about one or two pixels. The mapping error is determined by the benchmark accuracy corresponding to normal input error, the interpolation accuracy of interspaced points, and the correction accuracy of the radiometric values. In this case, the error bound of the output data is about two pixels.

Bad data corresponds to data having errors of about 4 or more pixels. The mapping error is determined mainly by the benchmark accuracy corresponding to these bad input errors. In this case, the error of the output data is probably about 5 or more pixels.

5.6 Computational Accuracy Requirements

It is desirable to investigate into the use of 16-bit words in the distortion-free computations.

Because of the complexity of the benchmark computations in equations (2.1) - (2.28), it is quite evident that sufficient accuracy will not be obtained by performing single-precision computations using 16-bit words.

However, for the partial derivatives computations in equations (2.29) - (2.32), the interpolation of interspaced points computaions in equations (3.7) - (3.8) and (3.15) - (3.16), and the radiometric correction computations in equations (4.1) - (4.5), it is possible to achieve the desired accuracy using single-precision computations involving 16-bit words. In this case, perhaps the best way to represent real numbers is as follows:

1 bit for sign of number
11 bits for range of number ($2^{11} - 1 = 2047$)
1 bit for sign of exponent
3 bits for range of exponent ($2^3 - 1 = 7$)

An alternative choice is as follows:

1 bit for sign of number
12 bits for range of number ($2^{12} - 1 = 4095$)
1 bit for sign of exponent
2 bits for range of exponent ($2^2 - 1 = 3$)

This second choice may not be as desirable because of the small range of the exponent.

SECTION 6 - CONCLUSION

From the preceding discussion, it is seen that it is possible to map in real-time the entire IR imagery comprising of coordinates and radiometric values of the samples. However, it is possible to map in real-time the entire visible imagery comprising of only the coordinates of the samples. This conclusion is based heavily on the assumption that it takes 1.5 microseconds for each addition or subtraction, and 6 microseconds for each multiplication.

The following table summarizes the time requirements for IR and visible imagery mapping:

	Entire IR	Cropped IR	Entire VIS	Cropped VIS
Benchmarks	8.6 sec.	5.7 sec.	4.6 min.	3.1 min.
Sample Coordinates	42 sec.	28 sec.	22.4 min.	14.9 min.
Radiometric Values	357 sec.	238 sec.	190.4 min.	126.9 min.
Total	6.8 min.	4.5 min.	217.4 min.	144.9 min.

The following table summarizes the expected accuracy of the distortion-free mapping (DFM) algorithms:

Input Data from SDB	Output Data from DFM
Perfect	1 pixel
Normal	2 pixels
Bad	5 or more pixels

REFERENCES

(1) Chan, F. K., "Distortion-Free Mapping of VISSR Imagery Data from Geosynchronous Satellites", Scientific Analysts and Consultants, Inc. Report. (1978).