RESEARCH TRIANGLE INSTITUTE

RTI/1796/00-01F

NASA CR-159287

# CONCEPTS FOR ON-BOARD SATELLITE IMAGE REGISTRATION

## Final Report

### Volume One

Prepared for

NASA

National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia

RESEARCH TRIANGLE PARK, NORTH CAROLINA 27709

# CONCEPTS FOR ON-BOARD SATELLITE IMAGE REGISTRATION

Final Report

Prepared Under Contract NAS1-15768

by

W. H. Ruedger, D. R. Daluge, and J. V. Aanstoos

Research Triangle Institute
Research Triangle Park, North Carolina 27709

Prepared for

NASA

National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia

June 1980

# PREFACE

This report was prepared by the Research Triangle Institute, Research Triangle Park, North Carolina, under Contract NAS1-15768. The work has been administered by the Electron Devices Research Branch of the Flight Electronics Division, Langley Research Center, National Aeronautics and Space Administration. Mr. W. L. Kelly IV served as Technical Representative.

These studies began on 29 March 1979 and were completed on 15 June 1980. Mr. W.H. Ruedger served as Project Leader. Dr. D.R. Daluge and Mr. J.V. Aanstoos completed the project team. Dr. W. E. Snyder, North Carolina State University, served as consultant to the program.

This volume covers tasks completed under the initial contract. Subsequent volumes cover additional efforts and include: 1) recommendations for the reference to be used in IAS demonstration hardware evaluation, 2) the impact of microelectronics technology advances on on-board signal processing, and, 3) the impact of data editing on data packetization and ground software requirements.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (continued)

LIST OF ILLUSTRATIONS (continued)

LIST OF TABLES

# 1.0 INTRODUCTION

The satellite data acquisition and handling system currently implemented by NASA is now operating at capacity and as such provides severe limitation on data throughput. NASA's mission model for the near-to-medium future indicates significant increases and/or changes in the data acquisition systems, data rates, and user requirements. To anticipate this new era of space observation requirements, NASA is embarking on the NASA End-to-End Data System (NEEDS) program. This program is an attempt to significantly increase the effectiveness and efficiency of the system that couples the user of space data with the sensors that acquire this data. The NEEDS program will therefore address the identification, development, and demonstration of data handling and processing techniques and technologies which are required to accomplish this.

More specifically, the NEEDS program goals present a requirement for on-board signal processing to achieve user-compatible, information-adaptive data acquisition. One very specific area of interest, which this study addresses, is the preprocessing required to register imaging sensor data which has been distorted by anomalies in subsatellite point position and/or attitude control. This study brings attention to the concepts and considerations involved in using state-of-the-art positioning systems such as the Global Positioning System (GPS) in concert with state-of-the-art attitude stabilization and/or determination systems to provide the required registration accuracy. Aspects of the study include an examination of the accuracy to which a given image picture-element can be located and identified, the determination of those algorithms required to augment the registration procedure, and consideration of the technology impact on performing these procedures on-board the satellite. The signal processing functions comprise a major constituent of the Information Adaptive System (IAS), a significant module of the NEEDS concept. The IAS essentially consists of the spaceborne portion of NEEDS exclusive of telemetry, support and housekeeping systems. A block diagram of the IAS is shown in figure 1-1. The signal processing discussed in this report is resident within the Data Pre-Processor shown in the figure.

Section 2.0 of this report discusses the general approach to registration. In this section it is pointed out that a similar study was completed by TRW[1-1] at the outset of the RTI program and it was therefore advantageous to take advantage of this as a point of departure. Section 3.0 discusses hardware implementation aspects, a demonstration hardware procurement specification is discussed in Section 4.0, a summary and recommendations are contained in Section 5.0, and the appendix presents an in-depth study of high-speed interpolation for resampling.

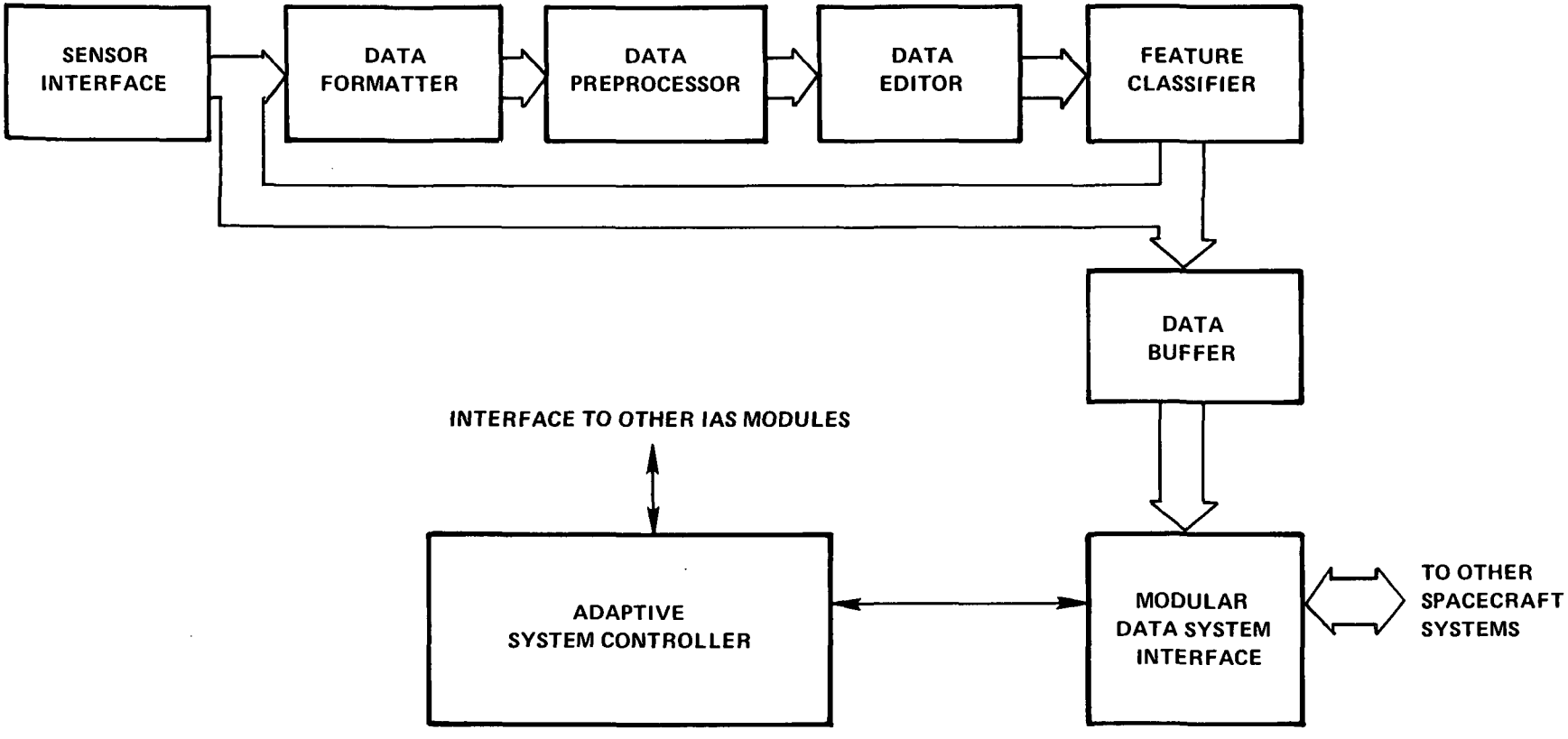



Figure 1-1. INFORMATION ADAPTIVE SYSTEM BLOCK DIAGRAM

## 2.0 IMAGE REGISTRATION APPROACH

As discussed previously, the major theme of the study was a conceptual definition of on-board signal processing to achieve image registration. About the time of initiation of study effort, TRW released the draft report of a study very similar in scope to this effort. In order to maximize the return of the RTI study, it was decided to use the TRW study as a point of departure and to build upon it with recommendations for deviations or revisions. These recommendations were to range from a completely different approach to minor "vernier" revisions. The general conclusion reached was that the basic TRW approach was sound, but that minor changes could achieve better registration accuracy with negligible hardware impact. This section documents those comments.

Major subsections will discuss inputs available for implementing on-board registration, namely position from GPS-PAC and attitude from state-of-the-art star trackers, will address the architecture approach, will consider the impact of choice of map projection, and will conclude with a brief discussion of the issues in use of ground control points for sub-pixel registration.

A major point discussed in Section 2.3 is the advantage in the use of windowing for high speed interpolation. That discussion characterizes interpolators in a way which explains the superiority of certain approaches and facilitates the design of interpolators with pre-specified frequency response characteristics.

### 2.1 Position and Attitude Information

The basic ingredient for real-time image registration is a source of measurements from which the pixel location in geoidal co-ordinates can be obtained. This requires measuring satellite position and attitude accurately and in real-time. The following discussion presents a brief description of techniques for achieving these measurements. For Landsat-D, position is to be accurately measured by means of a Global Positioning System (GPS) receiver flown on board while attitude is to be derived by means of a star-tracker augmented inertial system. Included in this section is a discussion of the GPS system (GPS-PAC) and a survey of current and projected star-tracker systems.

#### GPS-PAC Receiver/Processor Assembly

This discussion describes the specifications for the GPS receiver to be flown on Landsat-D. It is essentially a condensation of the Applied Physics Laboratory Design Specification [2-1].

The Receiver/Processor assembly (R/PA) is a major module of the GPS system and includes the receivers, processor/software, synthesizer, time code generator and the power supply. The R/PA and its relationship to GPS-PAC is shown in figure 2-1.

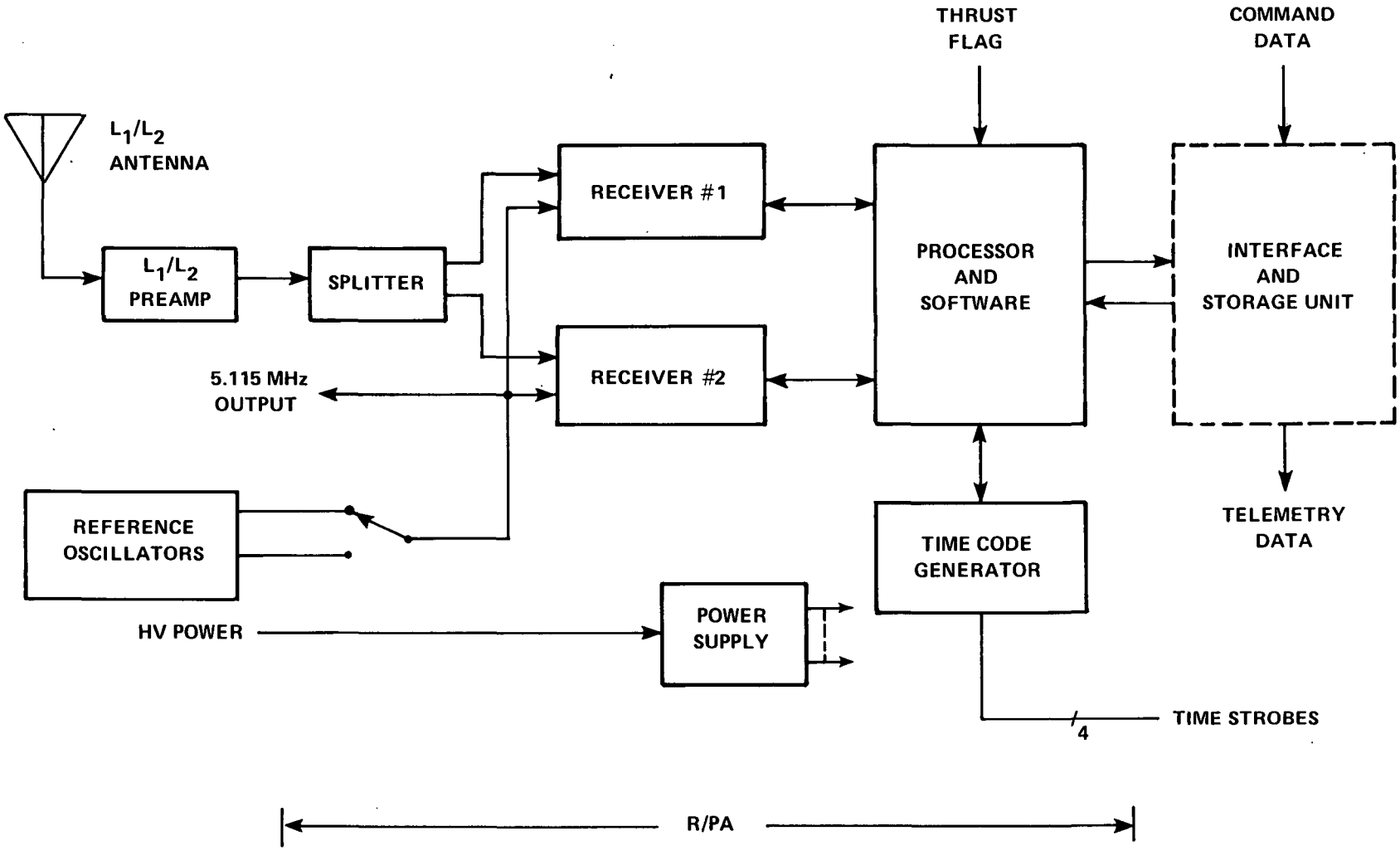



Figure 2-1. R/PA MAJOR COMPONENTS AND INTERFACES [2-1]

GPS-PAC is to provide accurate measurement and prediction of host vehicle navigation parameters in real-time. It will provide (by command) either simultaneous (dual receiver) or sequential (single receiver) processing of the dual L-band signal pairs ($L_1/L_2$ are used for ionospheric group delay compensation). The processor will have provision for selecting the satellite constellation with minimum geometric-dilution-of-precision (GDOP). The host vehicle - NDS dynamics are specified as:

range - 18,000 to 27,000 km
range rate - $\pm$ 9 km/sec
range acceleration - $\pm$ 16 $m/s^2$
range jerk - $\pm$ 0.02 $m/s^3$

While in navigate mode, satellites are tracked sequentially for both simultaneous and sequential configurations with not more than six seconds dwell time on each satellite. When no satellites are in view, the processor enters a dead-reckoning mode with a stored host vehicle dynamics model. The nominal navigation "cycle period" is 6 seconds.

The Kalman navigation filter outputs host vehicle position, velocity, and GPS system time on command with the output rate selected on command. Selected data files are also output at selected rates on command. The specified accuracy is:

pseudo range $\leq$ 1.5 meters
delta-pseudo range $\leq$ 0.02 meters

Time-to-first-fix (TTFF) is < 470 seconds and in case of outage, Time-to-subsequent-fix (TTSF) is < 190 seconds for short outage (15 minutes dead reckoning) and < 445 seconds for long outage (50 minutes dead reckoning).

GDOP is reviewed at least every three minutes.

Attitude Determination and Control Considerations

As mentioned previously, the two critical parameters to determine pixel location are satellite attitude and sub-satellite point position. This discussion presents comments on the capability to determine satellite attitude. Data have been obtained from TRW [1-1] and from a study done by Boeing [2-2] to determine trade-offs in selection of an attitude control and determination system for the Space Test Program Standard Satellite. Material presented has been excerpted from these reports in some instances directly.

TRW reports (see Table 2-1) that the pointing accuracy available from the Modular Attitude Control System (MACS) on LANDSAT-D is on the order of 0.0027 degrees which translates to 33 meters at 705 km. This exceeds the performance goal of registration to one-half pixel. TRW concludes then that utilization of a star tracker is required if Ground Control Points are not used. Table 2-2 shows an error summary of the BBRC CT-401 star tracker (considered as a candidate). The result indicates that it also is not capable of supporting registration to within one-half pixel and TRW

Table 2-1. Estimated Normal On-Orbit Performance Summary (Per Axis) [1-1]

| Error Source | Short-Term (60 seconds) Attitude Stability (1 σ) μrad | Long-Term (20 minutes) Attitude Stability (1 σ) μrad | ACS Pointing Accuracy (1 σ)* μrad | |
|---|---|---|---|---|
| | | | Random Varying | Biases |
| ATTITUDE REFERENCE SYSTEM | | | | |
| • Gyros | 1.0 | 8.0 | 8.0 | |
| • Star tracker (filtered) | | | 25.0 | 19.4 |
| Subtotal ARS | | | 45.6 | |
| ATTITUDE CONTROL SYSTEM | | | | |
| • Gyro noise and quantization | 1.8 | 1.8 | 1.8 | |
| • Controller dynamics | 1.5 | 1.5 | 4.4 | |
| • RW mass imbalance | 0.015 | 0.015 | 0.015 | |
| • | | | | |
| APPENDAGE MOTION | | | | |
| • Solar array | 2.8 | 2.8 | 2.8 | |
| • Ku-band antenna and mast | 0.1 | 0.1 | 0.1 | |
| • TM (HAC design)** | 1.0 | 1.0 | 1.0 | |
| • MSS | 0.4 | 0.4 | 0.4 | |
| STRUCTURE | | | | |
| • Misalignments (STA LOS/cube) | --- | --- | 5.0 | |
| • Thermal shifts | --- | 1.0 | 1.0 | |
| Total (RSS) | 4.0 | 8.9 | 46.3 (0.0027°) | |

Table 2-2. BBRC CT-401, $8^0$ x $8^0$ FOV Star Tracker Characteristics [1-1]

| Error Source | Error Without Compensation | 1 σ Error With External Compensation (arc-sec) | Comments |
|---|---|---|---|
| Slowly Varying Biases | | | |
| • Null offset and nominal dynamic lag | 18 arc-sec (1 σ) | --- | Calibrated out by bench test |
| • Aging (null shifts, dynamic lag shifts, etc.) | 4 arc-sec (1 σ) | 4.0 | No further compensation |
| Total biases (RSS) | 18.4 | 4.0 | |
| Errors Appearing Random for Discrete Star Observation | | | |
| • Nonlinearity and nonorthogonality | 10 arc-min (peak) | 3.0 | Bench calibration by polynominal fit: 60 coefficient |
| • Temperature and power sensitivity ($\pm 30^0$C) | 2 arc-sec/$^0$C | | |
| • External magnetic field (0.4 gauss) | 10 arc-sec (peak) | 2.5 | Correction factors/terms obtained from bench test |
| • Star intensity ($\Delta M_v$ = 3) | 30 arc-sec (1 σ) | 2.5 | |
| • Scale factor changes (aging) | 10 arc-sec (1 σ) | 1.5 | Occasional updates from ground-based Kalman filter |
| Subtotal (RSS) | 203.5 | 4.9 | |
| Noise | | | |
| • NEA (Noise Equivalent Angle) | 5 arc-sec (1 σ) ($\tau$ = 0.5 sec, $M_v$ = 6) | 5.0 | No compensation |
| Total random errors for discrete star observation (RSS) | 203.6 | 7.0 | |
| Total 1 σ accuracy ($\sum$ Bias + Random) | 222 arc-sec | 11 arc-sec | |

concludes an advanced star tracker is required. They mention a high accuracy stellar-inertial attitude reference system being developed by JPL which has an accuracy of 7 arc-seconds and project that technology should be able to drive this to 3 arc-seconds corresponding to about 10 meters errors in along and cross track directions due to pitch and roll variations.

The Boeing study survey shown in Table 2-3 further supports the idea that accuracies over and above the CT-401 system are probably not currently available. Boeing alludes to achieving increased accuracy through use of the space sextant. This sensor is being developed by the Martin Marietta Aerospace Corporation and was planned for test in 1979. It is more than a star sensor in that is provides star, moon and earth fixes for navigation state determinations. Projected accuracy is on the order of 0.3 arc-second. General features of this system are listed in Table 2-4.

## 2.2 Comment on TRW Processing Algorithm

The processing flow to compute a map projection location of a given pixel is as shown in Figure 2-2. Computations involved consist of a series of co-ordinate transformations required to take pixel location from scanner co-ordinates to map projection co-ordinates with various Euler angles, position, and attitude as inputs Flow in Figure 2-2 is shown right to left in order to be compatible with the generalized matrix equation:

$$\begin{bmatrix}\text{OUTPUT}\end{bmatrix} = \begin{bmatrix}A\end{bmatrix}\begin{bmatrix}B\end{bmatrix}\begin{bmatrix}C\end{bmatrix}\begin{bmatrix}\text{INPUT}\end{bmatrix}$$

In general this computation is performed only for a few pixels per scan and distance along-scan for other pixels calculated from the scan geometry. Figure 2-3 shows the processor functional organization recommended by TRW [1-1]. After review of the approach, RTI concluded that it is sound and implementable. Several minor revisions which increase performance are discussed here and include the along-scan distance expression, the map projection, the resampling algorithm, latitude/longitude determination, and an effect ignored by TRW, the local earth "tilt" due to geocentric co-ordinates. Other comments on along-scan distance calculation and on resampling which were generated early in the study are included as appendices. Those comments are pertinent but do not possess the relevance of the topics discussed in the main body of the report.

### Map Projection

The goals of map projection image registration are:

1. to represent the image in a form which is independent of satellite and scan parameters,

Table 2-3. STAR SENSORS [2-2]

| MANUFACTURER | ACCURACY | POWER (WATTS) | WEIGHT (LBS) | LIFE-REL | COST RECURRING | REMARKS |
|---|---|---|---|---|---|---|
| ITT-GILF | 20 $\widehat{SEC}$ | 12 | 14 | N.D. ** | 75 K | STRAPDOWN TRACKER ELMS |
| BBRC CS 200 | 1.0 $\widehat{MIN}$ | 0.5 | 3 | .994/180 DAYS | 90K | SCANNER 3-8 BIT WORDS PER STAR CROSSING DEVELOPED FOR OSO |
| BBRC CT 401 * | 10 $\widehat{SEC}$ | 3.0 | 11 | .955/1 YR | 120 K | STRAPDOWN TRACKER DEVELOPED FOR SAS-C |
| TRW POLARIS | 30 $\widehat{SEC}$ | 7.0 | 9.0 | .94/YR | 100 K | STRAPDOWN TRACKER ACQUISITION FOV: 25° CIRC TRACKING FOV: 34 MIN |
| TRW SRA | 0.5° | 0.5 | 2.5 | .98/3 YRS | 100 K | PIONEER SCANNER SOLID STATE |
| MARTIN MARIETTA AEROSPACE | 1.0 $\widehat{SEC}$ | 30 PEAK 7 ON 2% DUTY CYCLE | 25 INCLUDING ELEC-TRONICS | 5 YRS, 90% CONFIDENCE | 250 K INCLUD-ING DRIRU PKG | SPACE SEXTANT IS PLANNED FOR FLIGHT TEST IN 1979. DESIGNED FOR CISLUNAR MISSIONS |

*NASA STANDARD

**No Data

Table 2-4. SPACE SEXTANT [2-2]

New sensor development for satellite autonomous navigation and attitude determination in combination with on-board computation

Sensor provides precise included angle measurements between lunar limb and star of visual magnitude ($\leq +2$)

- Timing pulses generated by precise speed control (phase lock loop) of sextant wheel produces included angle measurements between sensor optical axes
- Engineering model of sextant is under development and will be tested to demonstrate operation and accuracy

Space sextant is primarily intended for navigation function but inherent accuracy of angle determination makes it attractive as attitude reference sensor

- Projected accuracy (not demonstrated) as attitude reference is 0.3 arc-second
- Reference AIAA G&C Conference Paper No. 75-1073

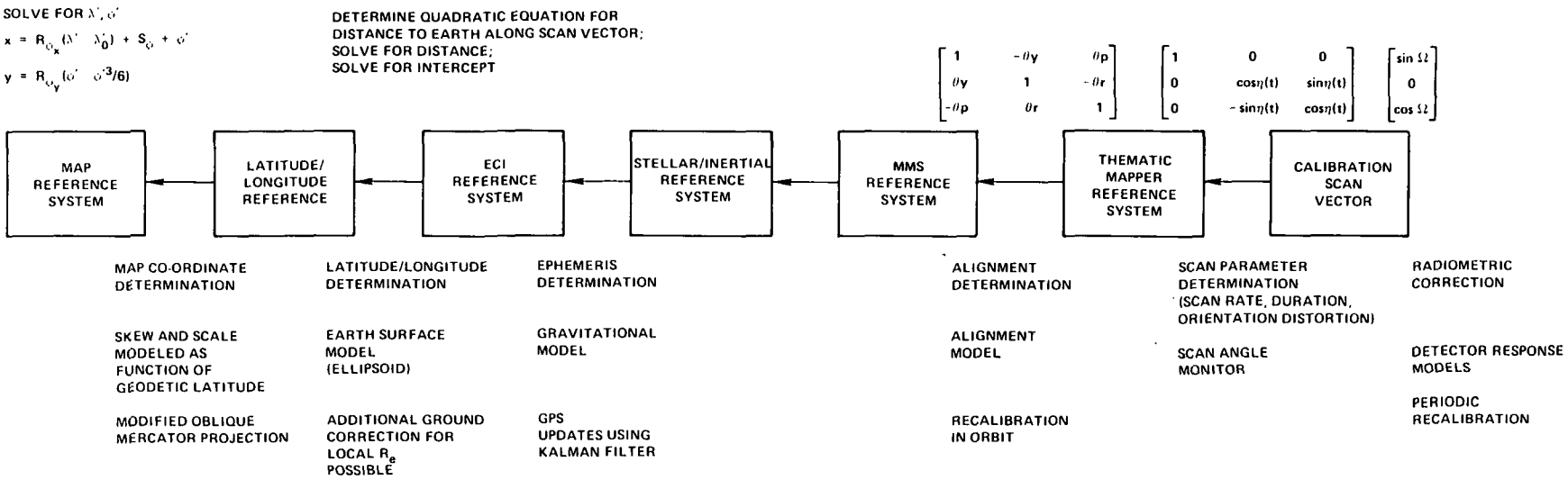



Figure 2-2. Map Projection Position Computation (Flow is from right to left as discussed in text)

**"PAGE MISSING FROM AVAILABLE VERSION"**

2. to represent the frame area with essentially true scale (some scale error must be tolerated when a curved surface is represented on a plane),
3. to locate pixels absolutely.

Further, since on-board real-time image registration limits storage and computational complexity:

4. scan lines should be nearly aligned with an axis of the projection,
5. rational functions or polynomial approximations of trigonometric and other functions should be used.

The first goal would be achieved by adopting a standard perspective to which all images are corrected, but two images could be compared directly only if the image centers coincided. Thus, perspective representation is to be avoided.

The Oblique Mercator Projection will be modified here for:

.computational efficiency

.earth rotation

.earth ellipticity

The regular projection is defined by:

$$\tan \lambda' = \cos i \tan \lambda + \sin i \tan \phi / \cos \lambda$$

$$\sin \phi' = \cos i \sin \phi - \sin i \cos \phi \sin \lambda$$

$$X = Re\ \lambda'$$

$$Y = Re \ln \tan\left(\frac{\pi}{4} + \frac{\phi'}{2}\right)$$

where $\phi$ is (spherical earth) latitude

$\lambda$ is longitude

$\phi'$ and $\lambda'$ are latitude and longitude relative to the new ground track equator

Re is earth radius

and, X and Y are the projection coordinates.

First, note that

$$\ln \tan\left(\frac{\pi}{4} + \frac{v}{2}\right) = \ln\left(\frac{1+\tan\frac{v}{2}}{1-\tan\frac{v}{2}}\right)$$

$$= 2\left[\tan\frac{v}{2} + \frac{\tan^3\left(\frac{v}{2}\right)}{3} + \frac{\tan^5\left(\frac{v}{2}\right)}{5} + \ldots\right]$$

$$\doteq v + \frac{v^3}{6}$$

The error of approximation is less than .0002 meters for

$$y = Re \ln \tan\left(\frac{\pi}{4}+\frac{\phi'}{2}\right) \text{ when } |\phi'|<.0145 \text{ radians}$$

The error is about 3.25 meters if only the linear term is used. The price of the improved accuracy is quite small. The error can be virtually eliminated by the inclusion of one more term.

Thus, one can use $Y = Re(\phi' + \phi'^3/6)$

The earth's rotation results in a latitude-dependent heading error which skews the scan lines with respect to the Oblique Mercator Projection equator (see Figures 2-4 and 2-5). Since latitude changes only by about 1.664 within a frame, the skew may be treated as a constant over a frame.

Thus, one can correct for the rotational skew by using

$$X = Re\ \lambda' + S\phi*\phi'$$

where the parameter $S\phi$ is chosen for frame center

$$S\phi = \tan(He) \doteq He + \frac{He^3}{3}$$

where He is the rotational heading error with a maximum value of about $4^\circ \doteq .07$ radians (see Figure 2-5).

It is convenient to move the zero of transformed longitude to frame center:

$$X = Re(\lambda' - \lambda o') + S\phi*\phi^1$$

where $\lambda o'$ is the original transformed longitude at frame center.

The earth's ellipticity remains to be accounted for. The goal is to map with essentially true scale at all latitudes, although the frame area changes with latitude, because of the variation in earth radius and orbital altitude. The along-scan distance expression may be used to choose an appropriate scaling factor.

$$d = Re\left[\sin^{-1}\left(\frac{Ro}{Re}\sin\Psi\right) - \Psi\right]$$

$$= (Ro-Re)\ \Psi \doteq Re\ \phi' \text{ for small } \Psi$$

Thus, the appropriate scaling factor is the local earth radius, which may be adequately modeled as a latitude-dependent parameter $R\phi_y$.

The final modified transform is

$$X = R\phi_x*(\lambda' - \lambda o') + S\phi*\phi^1$$
$$Y = R\phi_y*(\phi' + \phi'^3/6)$$

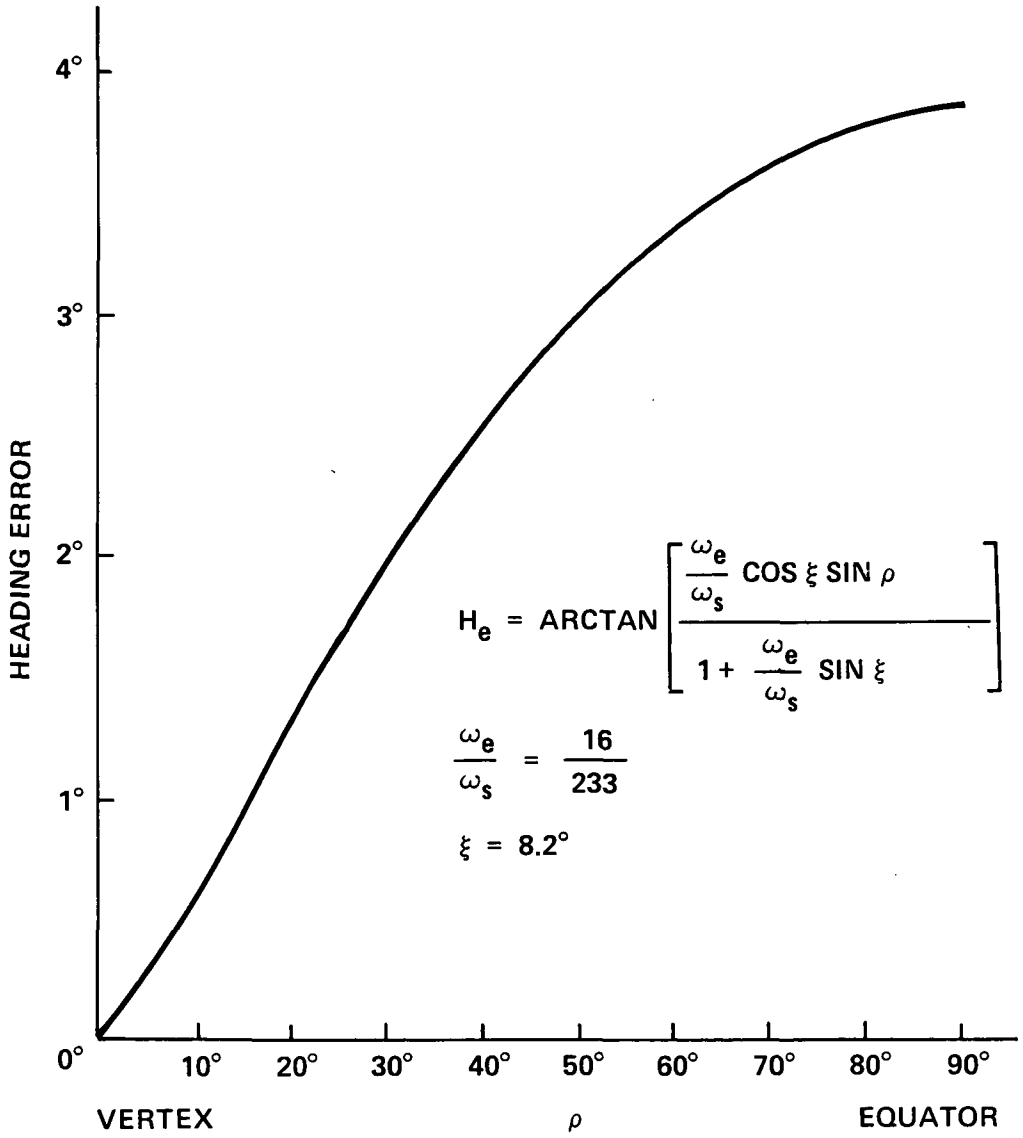



Figure 2-4. Heading Error Resulting Solely From Earth Rotation

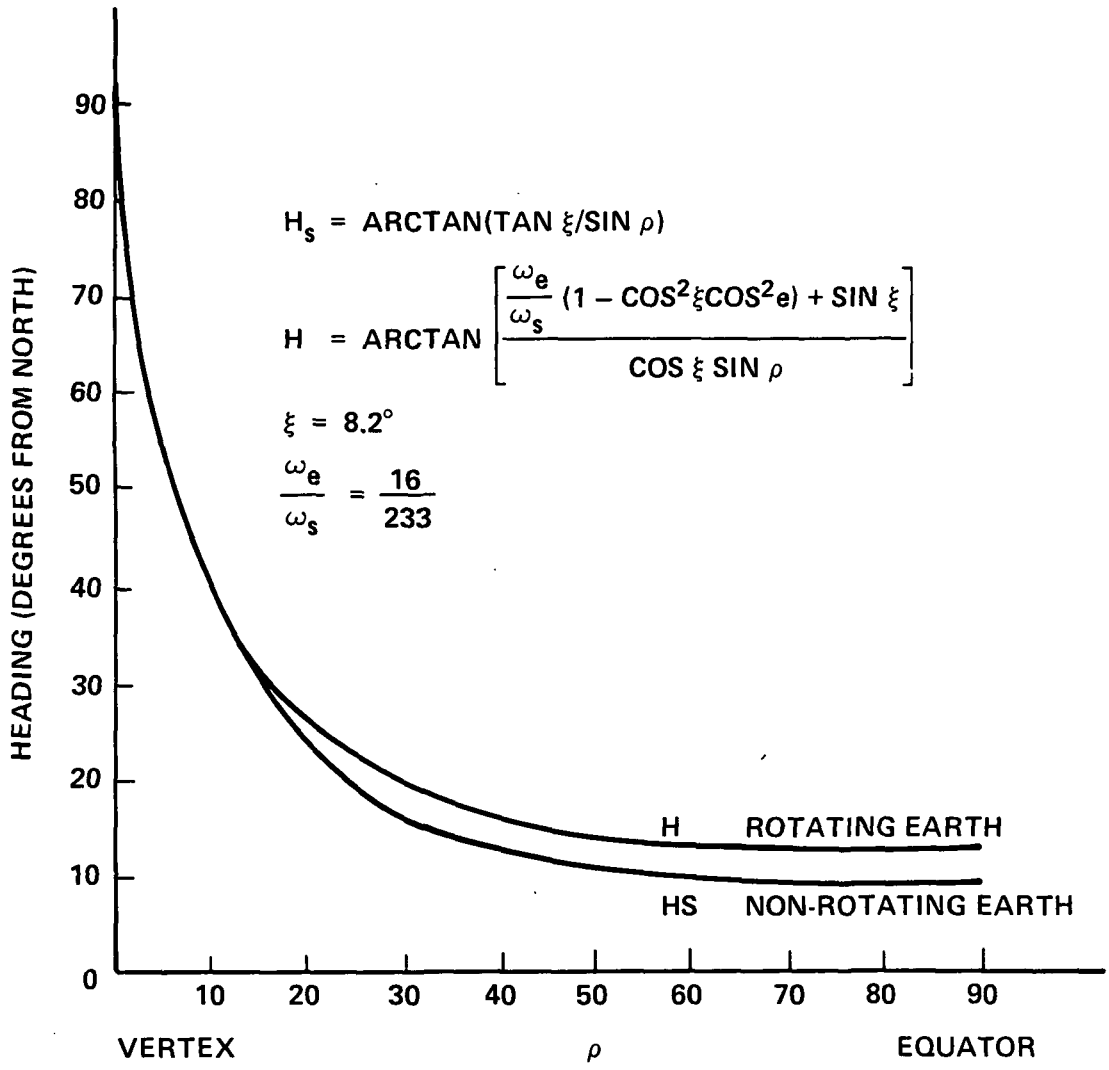



Figure 2-5. Satellite Heading For Rotating And Non-Rotating Earth

where

$$R\emptyset_y = \frac{ab\sqrt{1 + \tan^2 \emptyset_c}}{\sqrt{a^2 \tan^2 \emptyset_c + b^2}}$$

here, $a = 6378$ km (equatorial radius)

$b = 6357$ km (polar radius)

$\emptyset_c$ = geocentric latitude

and $R\emptyset_y$ is chosen at frame center

(Note: $\tan \emptyset_c = \left(\frac{b}{a}\right)^2 \tan \emptyset$, where $\emptyset$ is geodetic latitude)

The parameter $R\emptyset_x$ must be chosen according to satellite velocity, local earth curvature, and local rate of change of transformed longitude with respect to surface distance. These can be adequately modeled as functions of latitude.

The Oblique Mercator Projection is used with geocentric latitude. The scale change within a scan is small, because the scan is virtually east-west.

Latitude Longitude Determination

At this point, the scan vector has been expressed in ECI coordinates. An earth surface model is used to determine the scanning longitude/latitude. The equation of the mean sea level is

$$\frac{x^2+y^2}{a^2} + \frac{z^2}{b^2} = 1$$

or,

$$x^2+y^2+\frac{a^2}{b^2} z^2 = a^2$$

where $a = 6378$ km

$b = 6357$ km

With the satellite at

$$\overline{R} = (x_o, y_o, z_o)$$

and unit scan vector

$$\overline{U} = (U_x, U_y, U_z),$$

a quadratic equation for intercept is

$$[(X_o+U_x\rho)^2+(Y_o+U_y\rho)^2] + \frac{a^2}{b^2}[(Z_o+U_z\rho)^2] = a^2$$

Combining like terms, one obtains

$$\left[U_x^2+U_y^2+\frac{a^2}{b^2}U_z^2\right]\rho^2$$

$$+2\left[X_oU_x+Y_oU_y+\frac{a^2}{b^2}Z_oU_z\right]\rho^2$$

$$+\left[X_o^2+Y_o^2+\frac{a^2}{b^2}Z_o^2-a^2\right]=0$$

The solution $\rho$ is the distance to intercept. It is the smaller of the two solutions and care must be taken in solution:

$$\rho = \frac{2C}{-B+\sqrt{B^2-4AC}}$$

where A, B, C are the quadratic, linear and constant coefficients, respectively. (This expression avoids the differencing of similar quantities, since B is negative.)

The intercept point in ECI coordinates is

$$\bar{R}_I = (X_o+U_x\rho,\ Y_o+U_y\rho,\ Z_o+U_z\rho)$$

and the geocentric latitude is

$$\emptyset = \arctan\frac{(Z_o+U_z\rho)}{||R_I||}$$

The longitude is

$$\lambda = \omega_e t + \arctan\left(\frac{Y_o+U_y\rho}{X_o+U_x\rho}\right)$$

where $\omega_e$ is the earth's angular velocity and t is the time since the ECI system last coincided with the rotating earth coordinate system.

"Tilt" introduced by geocentric co-ordinates

For highest registration accuracy, an earth-sensing satellite such as Landsat-D can determine the map projection coordinates directly, but the required time and computation are excessive for pixel-wise use. This suggests some sort of interpolation procedure. If the map projection along-scan coordinate has good scale accuracy, the problem can be treated as that of estimation of along-scan ground distance for the reference ellipsoid as a function of the scan angle parameter, but the perspective asymmetry resulting from geocentric scanning rather than geonormal scanning must be accounted for. Unless it is, there will be up to $\pm$ 11 meters of error at the ends of the $\pm$ 92 km. scan, more than half of the half-pixel error budget of Landsat-D.

A simplified along-scan distance expression

The locus of points on the earth imaged by a line array of detectors will be called the scan arc. For each pixel there is a satellite-to-imaged area vector, which will be called the scan vector. The angle between a particular scan vector and the scan vector at center scan will be called the scan angle. It will be assumed that the scan arc can be accurately modeled as an arc of a great circle on a sphere of radius $R_{scan}$.

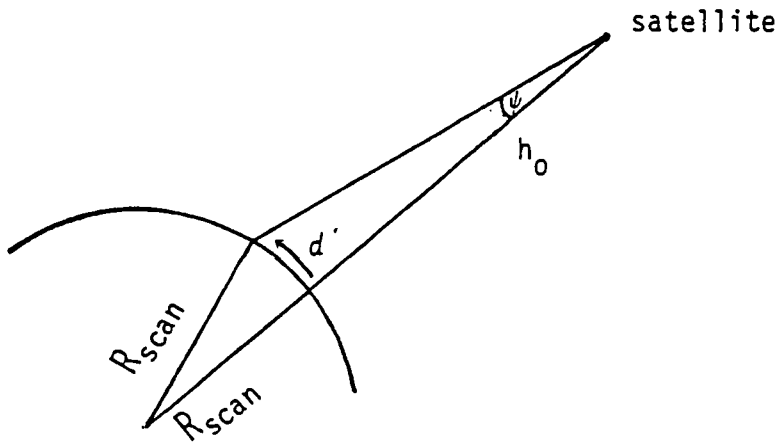



Figure 2-6. Scanning Geometry

In Figure 2-6, $h_o$ is the satellite altitude, measured along the normal to the scan arc, and d is the along-scan distrance to be calculated as a function of the scan angle $\Psi$.

From the law of sines:

$$\frac{h_o+R_{scan}}{\sin(\pi-\Psi-\frac{d}{R_{scan}})} = \frac{h_o+R_{scan}}{\sin(\Psi+d/R_{scan})} = \frac{R_{scan}}{\sin\Psi}$$

$$\sin(\Psi + d/R_{scan}) = \frac{h_o+R_{scan}}{R_{scan}} \sin\Psi$$

$$d/R_{scan} = \sin^{-1}\left(\frac{h_o+R_{scan}}{R_{scan}} \sin\Psi\right) - \Psi$$

$$d = R_{scan}\left[\sin^{-1}\left(\frac{h_o+R_{scan}}{R_{scan}} \sin\Psi\right) - \Psi\right]$$

This equation is exact for a spherical earth model with $R_{scan} = R_e$, the earth radius.

The extension to the reference ellipsoid

Numerical calculations show that the expression derived above is a good approximation if $\Psi$ is measured from the normal to the ellipsoid, but that a small asymmetry is introduced when $\Psi$ is measured from the geocentric direction. This can be seen most clearly near the pole, where the scan arc is exactly meridional (see Figure 2-7), but the asymmetry is seen at all non-zero latitudes to some extent (unless the satellite has a polar orbit).

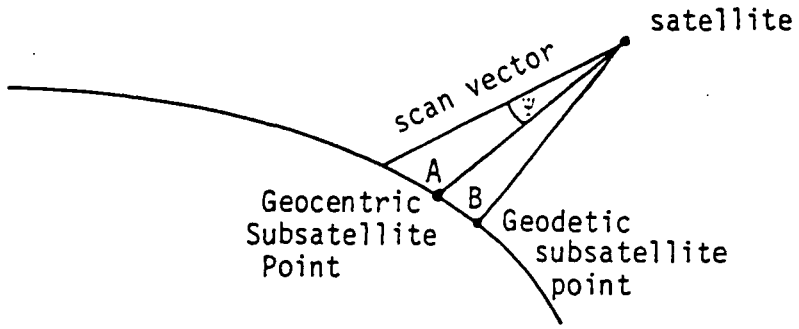



Figure 2-7. Perspective asymmetry results from the fact that the scan vector at center scan is not normal to the scan arc.

Because of the perspective asymmetry, the poleward half of the scan is underestimated while the anti-polar half is overestimated as shown in Table 2-5. For best accuracy, the scan arc should be centered at the geodetic sub-satellite point and $R_{scan}$ should be the radius of curvature of the scan arc, but very good accuracy (except for the asymmetry) is obtained with:

$h_o$ = geocentric altitude

$R_{scan} = R_e$ = earth radius

and $\Psi$ measured from the direction of the geocentric subsatellite point

A simple modification, using the concept of a scan angle bias, eliminates most of the asymmetry while preserving the simple form of the distance expression.

The scan angle bias correction

Figure 2-8 shows the scanning geometry as seen from the satellite position.

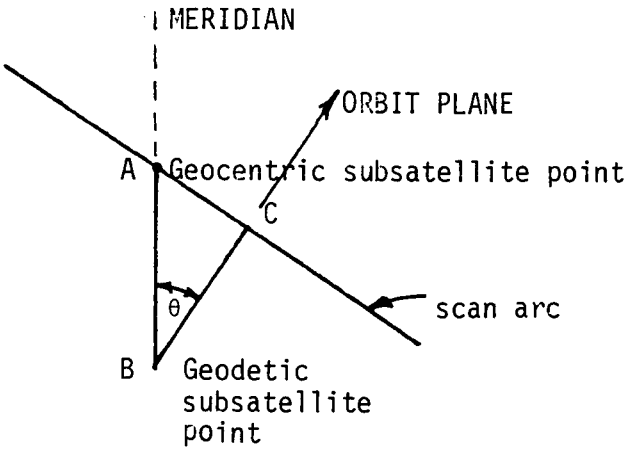



Figure 2-8 The scan vector is biased from the normal direction

On the locally-defined reference sphere, ABC defines a right spherical triangle. The angle $\Theta$ is the angle between the orbit plane and the meridonal plane. The scan angle $\Psi$ corresponding to point C is approximately

$$\Psi_o = \Delta \phi \sin \Theta$$

where $\Delta \phi$ is the difference between geocentric latitude and geodetic latitude.

Table 2-5. Total Ground Distance Perspective Error

| Latitude at Center Scan | Error at Scan Edge in Meters | |
|---|---|---|
| | Poleward Half | Equatorial Half |
| 0° | -0.1 | -0.1 |
| 9.9° | -2.1 | +2.1 |
| 19.8° | -4.1 | +4.2 |
| 29.7° | -5.9 | +6.2 |
| 39.5° | -7.5 | +8.0 |
| 49.3° | -8.9 | +9.5 |
| 59.0° | -9.9 | +10.7 |
| 68.4° | -10.7 | +11.6 |
| 77.1° | -11.1 | +12.2 |
| 81.8° | -11.3 | +12.4 |

To put this into a more useable form, note that

$$\Delta \phi \doteq \tan \Delta \phi = \frac{\left(\frac{a}{b}\right)^2 \tan \phi - \tan \phi}{1 + \left(\frac{a}{b}\right)^2 \tan^2 \phi}$$

$$= \frac{(a^2 - b^2) \sin \phi \cos \phi}{b^2 \cos^2 \phi + a^2 \sin^2 \phi} = \frac{(a^2 - b^2)}{(a^2 - b^2)} \sin \phi \cos \phi R_E^2$$

where $\phi$ = geocentric latitude

$R_e$ = earth radius

$a$ = equatorial radius = 6378.165 km.

$b$ = polar radius = 6356.783 km.

and $\sin \theta = \cos i / \cos \phi$

where $i$ = orbital inclination.

Thus, one obtains

$$\Psi_o = \frac{(a^2 - b^2)}{a^2 \quad b^2} \cos i \sin \phi R_e^2$$

$$= 2.36258\ 10^{-11} \sin \phi R_e^2$$

The accuracy of this expression has been verified by numerical calculation of the angle between the center scan vector and the scan arc.

To use this scan angle bias correction, one merely adds the bias value to each scan angle (a poleward angle is considered positive) and then subtracts the appropriate biasing value from the ground distance. That is, the corrected distance $d'(\Psi)$ is defined by

$$d'(\Psi) = d(\Psi + \Psi_o) - d(\Psi_o)$$

An empirically-derived bias constant provides slightly improved results:

$$\Psi_o = 2.11185\ 10^{-11} \sin \phi R_e^2$$

The asymmetry is essentially removed and the ground distance expression $d'(\Psi)$

is within 0.7 meters at all latitudes for a $\pm$ 92 km. scan at an orbital inclination of $81.8^{\circ}$.

## 2.3 High Speed Interpolation - Background and Key Results

When continuous data are being sampled rapidly, it is not always possible to take samples at exactly the right points. For example, an earth-imaging satellite may take data with equal scan angle increments when data are really needed with equal ground distance increments. Thus, it is necessary to estimate data values between the actual sample points. One might also need to estimate the position of the peak value of the data from samples near the peak, as when locating a landmark in an image.

Two high-speed interpolation procedures are commonly used for estimation. The nearest-neighbor method uses the nearest actual sample as the estimate and linear interpolation fits a piecewise-linear function to the data to allow estimation. These methods are acceptable for certain applications, for example, for data in which adjacent samples are very highly correlated, but they are inadequate for interpolation of high-resolution digital imagery.

A family of moderately-efficient interpolation procedures, known as cubic convolutions, have been espoused by Rifman, et al, [2-3]. These methods fit piecewise-cubic functions to the data samples to permit estimation. Some of these methods have been shown to perform exceptionally well on real satellite imagery. The simpler methods produce interpolation artifacts, such as blockiness or blurring, which may even be visible to the eye when the interpolated imagery is displayed, but these effects are significantly reduced with cubic convolution. These methods seem to have been derived somewhat intuitively, by requiring certain properties of the fitted cubic polynomial (although Simon [2-4] has provided a stochastic criterion for optimal interpolation of noisy data).

This study has characterized interpolators in a way which, for band-limited data, explains the superiority of cubic convolution over the simpler approaches and facilitates the design of interpolators with specified frequency response characteristics.

If $f(x)$ is a continuous data function sampled at integer values, then the interpolation estimate $\hat{f}$ is defined by

$$\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(k)\, h(x-k)$$

where h(x) is the convolution kernel. If $\hat{f}(x)$ is to be exact for each of the actual sample points then h(k) = 0 for each integer k except h(0) = 1. In view of this, the interpolator can always be expressed as

$$h(x) = w(x)\ \text{sinc}\ (x) = w(x)\frac{\sin(\pi x)}{\pi x}$$

where sinc (x) is the kernel which implements exact interpolation of bandlimited data when the sample set is infinite and w(x) is a "window function" which tends to compensate for the errors associated with the use of a finite sample set. Windows are used, for example, to design a near-optimal finite impulse response digital low pass filter. The significant points here are that windows can be used to design bandlimited interpolators and that every interpolator can be associated with a (possibly discontinuous) window function. One can alternately regard the function w(x) as a window for the sample impulses or as a window for the sinc function.

By varying the window, one achieves trigonometric polynomial interpolation or Lagrange polynomial interpolation or methods favoring other classes of data. The cubic convolution method proposed by TRW [1-1] for Landsat-D approximates trigonometric polynomial interpolation, with much simpler computation. It also corresponds quite closely to that obtained using the optimal window for spectral estimation [2-5].

There are two points of detail. First, for maximum speed, the convolution weights, that is, the samples of h used in the interpolation sum

$$\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(k)\ h(x-k)$$

can be stored in a lookup table. In this case, computational complexity plays no role in the choice of the interpolator. Second, for correct interpolation of constants, it may be necessary to modify the interpolator slightly. There is a straightforward procedure for this. One can calculate the mean of the data, interpolate only the deviations from the mean and then reintroduce the mean. When the details of this are written out, it is found that this is equivalent to interpolation with a slightly modified kernel, and this kernel interpolates constants exactly.

## 2.4 Use of ground control points for sub-pixel registration

TRW [1-1] indicates and this study corroborates that on-board real-time image registration is probably limited to accuracies on the order of one pixel. An obvious way to improve upon this is to incorporate ground control points (GCP) into the registration process.

As a first step in investigating sub-pixel registration, a literature search was conducted to ascertain the extent of previous work in this area. The results of this search were very disappointing. No relevant literature was revealed which related either to sub-pixel accuracy or to on-board use of ground control points.

While the latter was not surprising the former was. As a result, it was decided to investigate the differences encountered in using GCP's for the attainment of coarse (~1 pixel) or fine ($\sim\frac{1}{5}$ pixel) registration. The following discussion pursues this in further detail.

In coarse landmark registration, one uses a large-scale landmark occupying many pixels. If a Landsat scan comprises about 16 lines of detectors, the landmark is likely to overlap several scans. A large body of water with well-defined edges might be used as a large-scale landmark, for example. Changes in water level will often produce significant image variation.

The standard coarse registration procedure is to compare a small reference image with equal-size subimages of the data image to be registered using the normalized correlation or sum of differences similarity measures. The subimage of the data image is called a "window". The window and the reference image form a "window pair".

The similarity measure has the form

$$\sum XijRij \quad \text{(correlation)}$$

or

$$\sum |Xij - Rij| \quad \text{(sum of differences)}$$

where i,j define row and column coordinates relative to a corner of the window (Xij) and the corresponding corner of the reference image (Rij).

If correlation is to be normalized, it must be divided by

$$\left(\sum X_{ij}^2\right)^{1/2}$$

and by

$$\left(\sum R_{ij}^2\right)^{1/2}$$

Normalized correlation should have a peak at the exact match point, while the sum of differences should have the value zero there, and, hence, requires no normalization. Of course, these sums are calculated only for a discrete set of windows, offset from each other by integer pixel values. Thus, either method can locate the match point only within one pixel in each dimension. Other techniques are used to refine the landmark registration point to sub-pixel accuracy.

The sum of differences similarity measure allows the use of an efficient thresholding strategy. This similarity measure, combined with such a thresholding strategy, is referred to as the SSDA (Sequential Similarity Detection Algorithm). It is more efficient, by far, on any digital computer (it is about 50 times as efficient on a general purpose computer, because it has very few multiplications, requires no normalization, and terminates computation quickly for grossly-mismatched window pairs). Correlation is, however, preferred at low signal-to-noise ratios (say, less than 2).

Coarse registration of several large-scale landmarks allows the determination of a registration transformation, which registers the entire image to roughly the same degree of accuracy (especially if the registration errors are primarily of low spatial frequency). By this procedure, or by an alternative procedure, small-scale landmarks may be registered within one pixel.

Small-scale landmarks can differ significantly from large-scale landmarks. Examples of small-scale landmarks are highway intersections and airports. These may fit into one Landsat scan of about 16 lines (480 meters). Except for snow cover, they are largely invariant in spectral signature.

In registration refinement, the emphasis can shift from minimization of the probability of gross error ("false lock") to minimization of the fine-scale error because the target landmark is already known to lie in an area without false match points.

There seem to be three basic approaches to registration refinement. One alternative is to choose a landmark with sharply-defined features, fit some sort of smooth surface to the data image points and then determine the extremum of this estimated data surface, whose coordinates are known. This procedure might be used, for example, in the case of a "beacon". In general, landmarks may not provide sharp extrema, so that registration errors may remain large.

A second method replaces the data function by the similarity measure function for window pairs of moderate size (say, 16x16 or 32x32 pixels). The similarity measure function (e.g., correlation) will usually have a sharper extremum than the data function itself. A smooth surface is again used to estimate the exact registration offset.

A third method is based on sophisticated edge detection techniques. The method proposed by Tisdale [2-6] is of this type. His technique involves three phases: an edge determination phase, an edge skeletonization phase, and a matching phase. An accuracy of 0.2 pixel is claimed. To achieve this accuracy, rather large landmarks comprising many city blocks were required.

The following are variations on these techniques which may provide some advantages for a real-time registration system.

If two-dimensional sampled data are interpolated via any convolutional kernel, then the centroid of the interpolated function is exactly the centroid of the sample impulses. Thus, it may be useful to provide the coordinates of the centroid of the landmark reference image and estimate it from the data. If correlation is used, the centroid should be a good first estimate of the correlation-peak (ideally, the correlation would be an autocorrelation and the correlation surface would then be symmetric).

Subpixel Registration Refinement

It may be necessary to use landmarks (ground control points) for absolute registration refinement to achieve the desired registration accuracy. The standard registration techniques, similarity detection and correlation, are intended primarily for registration to one pixel accuracy. What is needed is a "super-resolution" technique.

One possible approach is to use resampling techniques to simulate high-resolution sampling and then use correlation (correlation is preferred over similarity detection for the refinement process). Disadvantages of this approach include: dependence upon a specific resampling technique and computational complexity. (Three such calculations would presumably

be used to determine an affine transformation for registration refinement.)

There is a simple alternative based upon the use of image features which are essentially one-dimensional, such as highways and airport runways or features which have well-defined edges, such as water boundaries. It is a much simpler procedure to estimate the position, in one dimension, of a well-defined linear feature. Given the coordinates of a point on the section of highway and a vector indicating its orientation, one fits a one-dimensional curve in the direction normal to the highway, using a two-dimensional interpolator, and then estimates the highway's position in one-dimension. This is the projection of the registration error vector in the specified direction. Six of these one-dimensional landmarks will usually define an affine registration transformation to register an entire image (just as 3 full two-dimensional registrations do). Other advantages are:

landmark availability
concise landmark data packets
redundancy easily accommodated

Of course, twice as many of these one-dimensional landmarks must be used to fix the same number of parameters.

The first step involves estimation of the edge's intersection with each row and each column of a small sub-image (about 10 x 10 pixels). This might be implemented as an analog process. Then, a least squares linear fit would be used to estimate the true edge position. The map projection distance error is then simply computed. This is the projection of the local registration desplacement vector along the normal to the edge.

If $w = \binom{x}{y}$ is the position vector in the original coordinate system and $w' = \binom{x'}{y'}$ is the position vector in the affine-transformed coordinate system, then

$$w' = Aw + w_o$$

where $w_o$ defines the translational component of the affine transformation and A is a non-singular matrix defining the scaling and rotation.

Six one-dimensional displacement estimations serve in general to determine the 6 parameters of the transformation. The 6 equations are of the form

$$(w_i' - w_i) \cdot u_i = (Aw_i + w_o - w_i) \cdot u_i = d_i$$

where $u_i$ is a unit vector in the direction normal to the i'th landmark edge, and $d_i$ is the estimated displacement. (A is a 2 x 2 matrix and $w_o$ has two components, for a total of 6 parameters).

Questions about the existence and stability of solutions must of course, be answered. Intuitively, one expects that the error sensitivity should be good if displacement estimates are made in orthogonal directions at nearby points (and only in orthogonal directions for such points).

A Kalman filter might be used to permit inter-frame use of landmark displacement data.

The affine transformation could be used either to estimate the complete displacement (not just the one-dimensional displacement) at the landmark points, or to re-register the image completely. The latter is probably cumbersome for on-board real-time use.

# 3.0 HARDWARE DESCRIPTION

## 3.1 Introduction

A functional description of one approach to the special-purpose hardware required for radiometric correction and resampling of the image data is presented here. TRW [1-1] proposed a three-stage approach to resampling, see Figure 3-1, with along-scan and across-scan resampling performed separately, each system accessing a common memory system called a skew buffer. This appears to be a sound approach to achieving the required speed with minimal component count and power dissipation. The documentation of the proposed implementation for each stage is somewhat sketchy, and the hardware diagrams presented are a combination of block diagram generality and gate-level detail. Data flow is not obvious in most cases, the width of most data paths being left unspecified. Control signals are not specifically defined as to their respective functions. The report derives rather precise component counts from these diagrams although the parts lists arrived at and compiled in Table 3-1 do not have an obvious correspondence to the hardware diagrams. In the case of the radiometric correction processor, the parts list included with Figure 3-2 does not correspond with the entries for that subsystem in Table 3-1.

TRW presented algorithms for along-scan and cross-scan resampling in flow chart form. Hardware diagrams of subsystems which are to implement these algorithms are presented later, along with the radiometric correction and skew buffer subsystems and a micro-controller to provide control signals for all of these systems. The radiometric correction and along-scan resampling subsystem diagrams are fairly vague. The other diagrams are much clearer, but all are lacking in accompanying operational explanations. The integration of these subsystems into a complete geometric correction processor is illustrated in the block diagram in Figure 3-1. However, the hardware documentation of the individual subsystems would be better understood from the inclusion of functional-level block diagrams which do not include gate-level wiring details and considerations of individual chip organization.

Each of the subsystems will now be considered in detail, and in some cases changes or alternative approaches will be suggested.

## 3.2 Radiometric Correction Processor

TRW's approach to the design of a radiometric correction processor was to store in RAM the breakpoints of piecewise linear approximations (see Figure 3-3) to the sensor calibration curves. For each pixel intensity value, the RAM is accessed

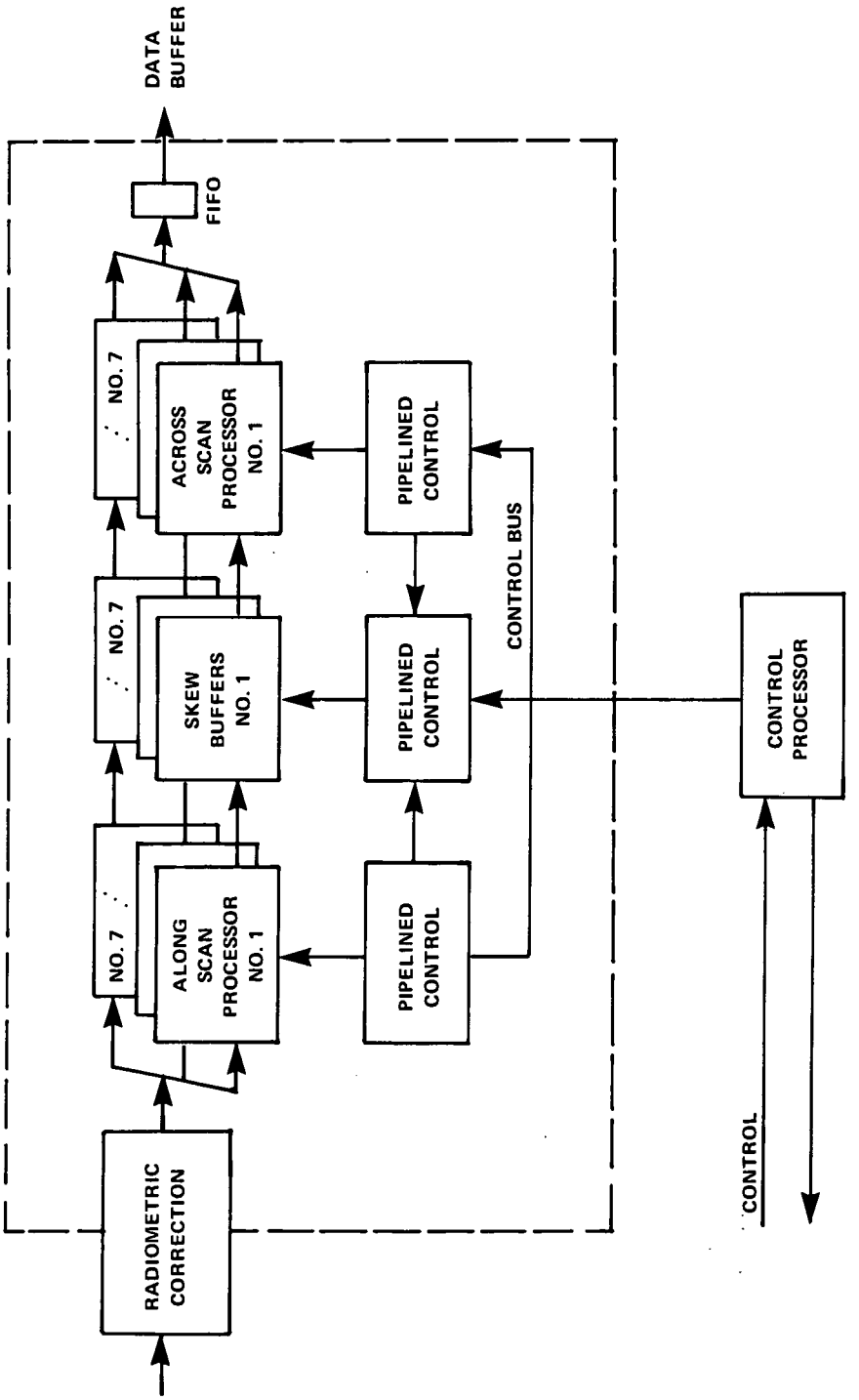



Figure 3-1. Image Processor Block Diagram [1-1]

Table 3-1. Parts List for Special Purpose Hardware [1-1]

| Item | Part Type | Part No. | Device Power (MW) | Device Area (Sq In.) | Function | | | | | | Total Power (MW) | Total Area (Sq In.) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Rad. Correct | Along Line Proc /X7 | Skew Buffer /X6 | Across Line Proc /X7 | Sequencers /X2 | Total Parts | | |
| 1 | Quad Nand | 54LS00 | 8 | 0.32 | 1 | 2/14 | 2 | 1/7 | 1/2 | 26 | 208 | 8.32 |
| 2 | Hex Invert | 54LS04 | 12 | 0.32 | 1 | | | | | 1 | 12 | 0.32 |
| 3 | 2/Mux | 54LS158 | 14 | 0.32 | | 4/28 | | | | 28 | 392 | 8.96 |
| 4 | 2/Mux/Latch | 54LS398 | 36 | 0.4 | | 2/14 | | 2/14 | | 28 | 1,008 | 11.2 |
| 5 | 4 Bit Adder | 54LS283 | 24 | 0.32 | 2 | 6/42 | 6 | 7/49 | | 99 | 2,376 | 31.68 |
| 6 | 8 Bit Latch | 54LS374 | 136 | 0.4 | 6 | 7/49 | 8/48 | 7/49 | 3/6 | 158 | 21,488 | 63.2 |
| 7 | 4 Bit Counter | 54LS161 | 93 | 0.32 | 1 | 4/28 | 4 | 2/14 | | 47 | 4,371 | 15.04 |
| 8 | 4 Bit Up/D Cn | 54LS169A | 100 | 0.32 | | | 8 | | | 8 | 800 | 2.56 |
| 9 | Dual D F/F | 54LS74 | 20 | 0.32 | | | 1 | | | 1 | 20 | 0.32 |
| 10 | Sequencer | 8X02 | 650 | 0.98 | | | | | 1/2 | 2 | 1,300 | 1.96 |
| 11 | 4X4 Scratch | 54LS670 | 135 | 0.32 | | | | 2/14 | | 14 | 1,890 | 4.48 |
| 12 | 8X8 Multip. | MPY8AJ | 1,650 | 1.4 | 2 | 1/7 | | 1/7 | | 16 | 26,400 | 22.4 |
| 13 | 256X4 RAM | 54LS207 | 200 | 0.32 | 4 | 2/14 | | | | 18 | 3,600 | 5.76 |
| 14 | 1KX8 RAM | MK4801 | 725 | 0.84 | | 1/7 | | | | 7 | 5,075 | 5.88 |
| 15 | 512X8 ROM | 93448 | 690 | 0.84 | | 1/7 | | 1/7 | 6/12 | 26 | 17,940 | 21.84 |
| 16 | 55KX1 RAM | TMS4164JL | 125 | 0.32 | | - | 32/192 | | | 192 | 24,000 | 61.44 |
| 17 | Dual JK F/F | 54LS73 | 15 | 0.32 | 1 | - | 1 | | | 2 | 30 | 0.64 |
| 18 | Oscillator | | 100 | 1.0 | | | | | 1 | 1 | 100 | 1.0 |
| | TOTALS | | | | 18 | 210 | 262 | 161 | 23 | 674 | 111,010 | 267 |

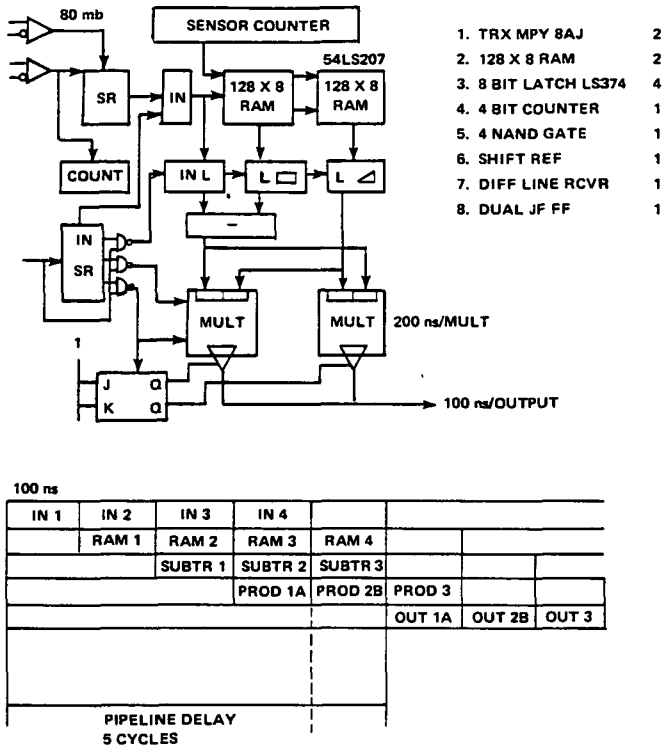



Figure 3-2. Radiometric Correction [1-1]

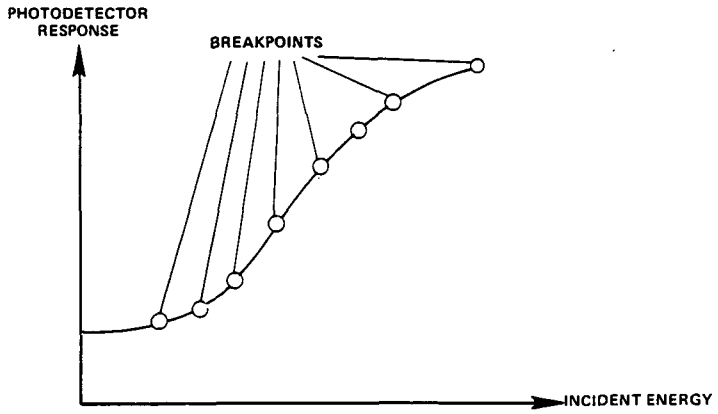



Figure 3-3. Piecewise Linear Approximation to Sensor Response Curve. Curve is Assumed Linear Between Breakpoints [1-1]

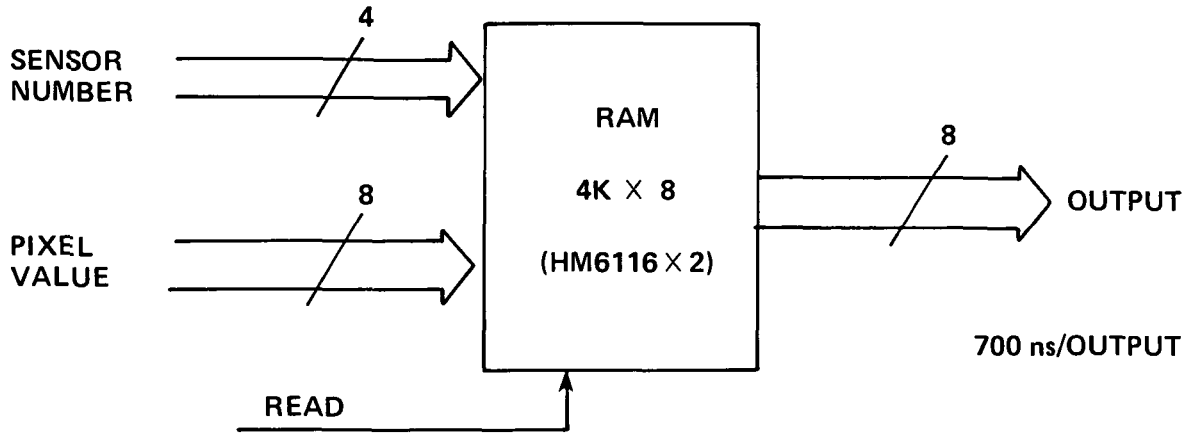
using the sensor number and some of the high-order bits of the intensity. The resulting breakpoints are used with the intensity value in a linear interpolation process involving a subtraction and a multiplication. The circuit which TRW proposes to accomplish this is as shown in Figure 3-2. This diagram shows 256 bytes of RAM being used to store the breakpoints. The overall parts list (Table 3-1) contradicts this and indicates that 512 bytes will be used. Since each sensor has a unique calibration curve, and there are 112 sensors (16 per band, times 7 bands), even the higher figure would only allow storage for 4 breakpoints per sensor. TRW states that "Because the response curves are smoothly varying, relatively few breakpoints need to be stored." However, no further analysis is done to determine just how many breakpoints should be used. The sample sensor response curve shown in the report for illustration purposes contains no fewer than eight breakpoints, which is probably a more reasonable estimate of the number required to achieve acceptable accuracy. In order to achieve the required speed (100 ns/output), TRW's design utilizes two multipliers running in parallel. This is no longer necessary, as high-speed 8-bit multipliers with more than adequate speed and lower power requirements are now available off-the-shelf. (see the survey of high-speed multiplier chips, Table 3-2). The resulting power requirement for the radiometric correction processor, using TRW's parts list (Table 3-1) data except for the multiplier contribution is:

| | | | |
|---|---|---|---|
| 1 | quad nand: | 8 | mw |
| 1 | hex invert: | 12 | |
| 2 | 4 bit adder: | 48 | |
| 6 | 8 bit latch: | 815 | |
| 1 | 4 bit counter: | 93 | |
| 1 | 8 bit multiplier: | 1000 | |
| 4 | 256 x 4 RAM: | 800 | |
| 1 | dual JK F/F: | 15 | |
| | Total Power + | 2791 | mw |

An alternate approach to radiometric correction which uses less power and results in greater accuracy with a much simpler design is shown in Figure 3-4. In this scheme, the entire sensor calibration curves are stored in RAM, instead of merely the breakpoints, thus eliminating the need for interpolation. Since there are 256 points per sensor calibration curve and 16 sensors per band, a total of 4K bytes of RAM is required for each band. The simple table lookup scheme can be implemented with off-the-shelf components using only two IC's per band. Since each band will require a separate lookup table, they can be run in parallel. This significantly reduces the speed requirement, allowing the use of low-power CMOS

Table 3-2. High-Speed Multiplier Chips

| MFR | Part no. | # Bits | Speed (ns) | Power (Watts) |
|---|---|---|---|---|
| AMD | AM25S558 | 8 | 45 | 1.4 |
| TRW | MPY-8HJ-1 | 8 | 45 | 1.0 |
| TRW | MPY-12HJ | 12 | 80 | 2.0 |
| TRW | MPY-16HJ | 16 | 100 | 3.0 |
| TRW | (planned) | 24 | 200 | 3.5 |
| | (The following multipliers include an on-chip accumulator for performing sum-of-product computations). | | | |
| TRW | TDC1008J | 8 | 70 | 1.2 |
| TRW | TDC1009J | 12 | 95 | 2.5 |
| TRW | TDC1010J | 16 | 115 | 3.5 |
| TRW | (being considered) | 16 | 450 | 1.0 |
| TRW | estimates of 3rd generation (ECL) performance | 8, 12, 16 | 50-80 | N.A. |




-- entire calibration curve stored in RAM

-- CMOS static RAM such as Hitachi HM6116 (2Kx8):

access time: 120 ns

power: 175 mw

-- total power = 350 mw/band = 2450 mw

Figure 3-4. Alternate Radiometric Correction Processor (each band).

devices such as the Hitachi HM6116 2Kx8-bit static RAM described in Figure 3-4. Its speed is more than adequate and its low power consumption results in a total power demand from all bands of only 2450, considerably less than that of the original design.

The design suggested above is appropriate for correcting thematic-mapper type of output, in which the number of sensors per band is relatively few (16 in this case). However, if a line scanner is used, each of the 6000 cells per line for each band will have to be corrected. Clearly it would be impractical to store an entire 256-point calibration curve for each of these cells. However, as will be pointed out in the discussion of line scanner specifications (Section 4.0 ), sufficient accuracy is obtainable from a linear correction. This involves for each cell output subtraction of a constant (corresponding to the dark current for that cell) and multiplication by another constant (a scaling factor). Thus only enough memory to store two constants per cell is required, along with a single adder and 8-bit multiplier. As discussed above, the speed of multiplier chips available now is sufficient for only one such device to be required for correction of all seven bands. If the line scanner consists of 16 lines of 6000 cells per band, the requirement is for 16x6000x2 or approximately 192K bytes of random access memory per band.

## 3.3 Line Scanner Buffering

As discussed by TRW, the general purpose processor (GPP) updates the distortion information which it passes to the resampling processors once every scan. However, since the distortion information which the resampling processors use during one scan was actually computed during the previous scan, it is not current and must result in some loss of accuracy. It is proposed that this situation be corrected by buffering one entire scan of image data to give the GPP time to compute up-to-date distortion information for that data. This buffer can be located either in front of or following the radiometric correction processor.

## 3.4 Along-Scan Processor

Before one can adequately comment on the design proposed by TRW or consider alternative architectures, several points concerning the design of the along-scan processor need some discussion. See Figure 3-5 for the algorithm of the along-scan processor as presented in [1-1].

A question arises from the totally different architectures proposed for the along-scan and across-scan processors. Although the distortion calculations are somewhat different in these cases (the existence of the skew buffer takes care of most differences), the interpolation procedure is very similar. It would seem that similarity in the two architectures could prove cost-effective.

For the along-scan processor, a 256 x 8 ROM is used to store the 64 sets of cubic interpolation weights. Each set of weights consists of four values which are used to determine the actual intensity output at grid points. The set of weights selected depends on the distortion calculated at a point. Eight bits derived from the distortion calculation are shown addressing the ROM. Presumably only one value of a set of four weights can be output at a time. Therefore, the addressing scheme for this ROM needs to be augmented. A scheme similar to that used in the across-scan processor for accessing interpolation weights would be preferable.

On the across-scan processor, the distortion calculation supplies six bits of the address with the other two bits being supplied by a counter designating the coefficient within each set of weights. As a pixel is used in an interpolation, the correct weight is addressed in the ROM.

The need for the many multiplexers and their interconnections shown in Figure 3-6 is not obvious. Only the MUX/L at the input to the adder seems to fulfill a useful role in the interpolation calculation. If these multiplexers are in fact part of an ALU chip and are not discrete parts, then their inclusion in Figure 3-6 detracts from an understanding of the along-scan processor. If these multiplexers serve in an active role, (as perhaps, gating to allow ROM coefficients to be stored temporarily in RAM), discussion to this effect would have proved helpful.

In any case, the written explanation of the design of the along-scan processor is sufficiently lacking in detail so as to make its complete review impracticable.

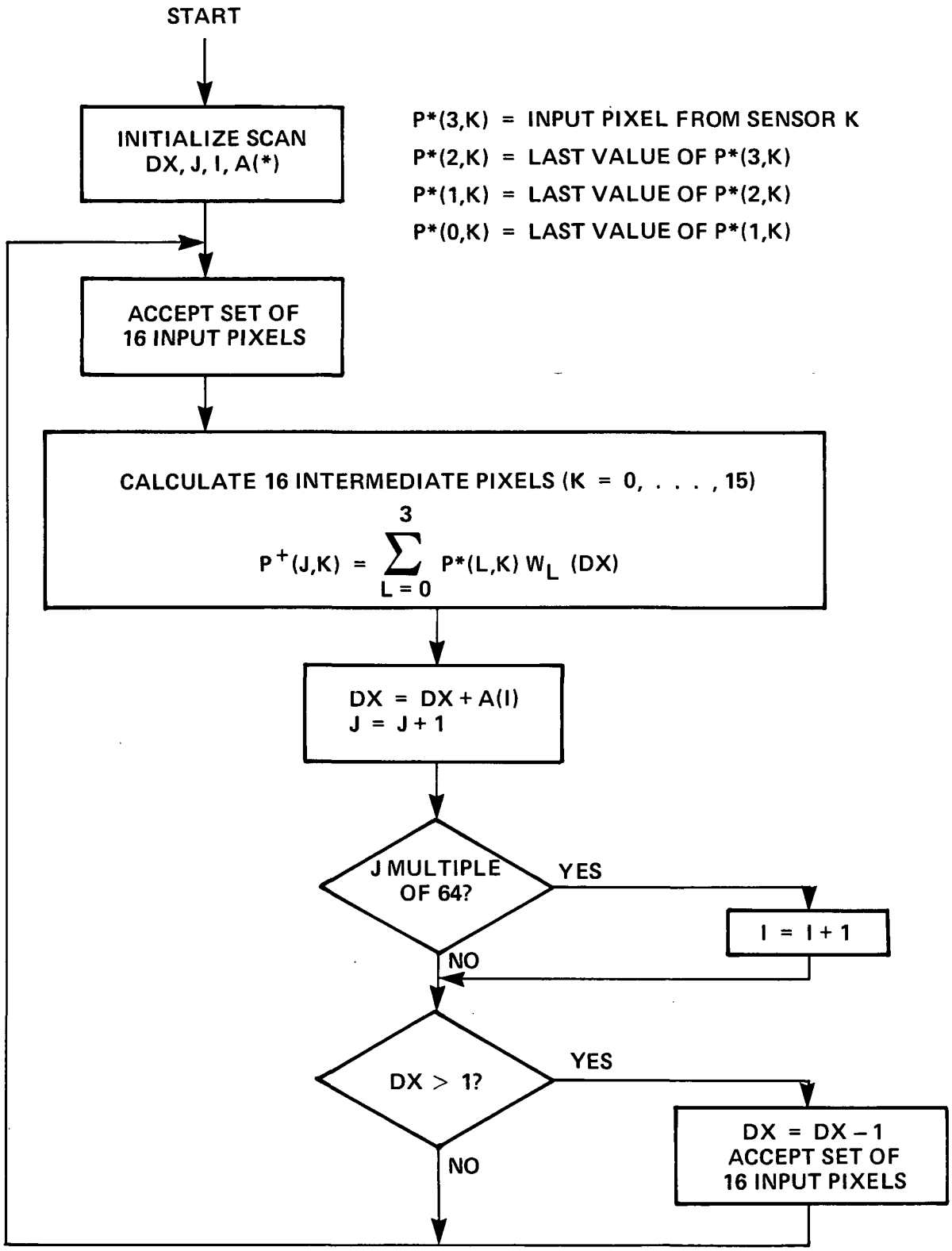
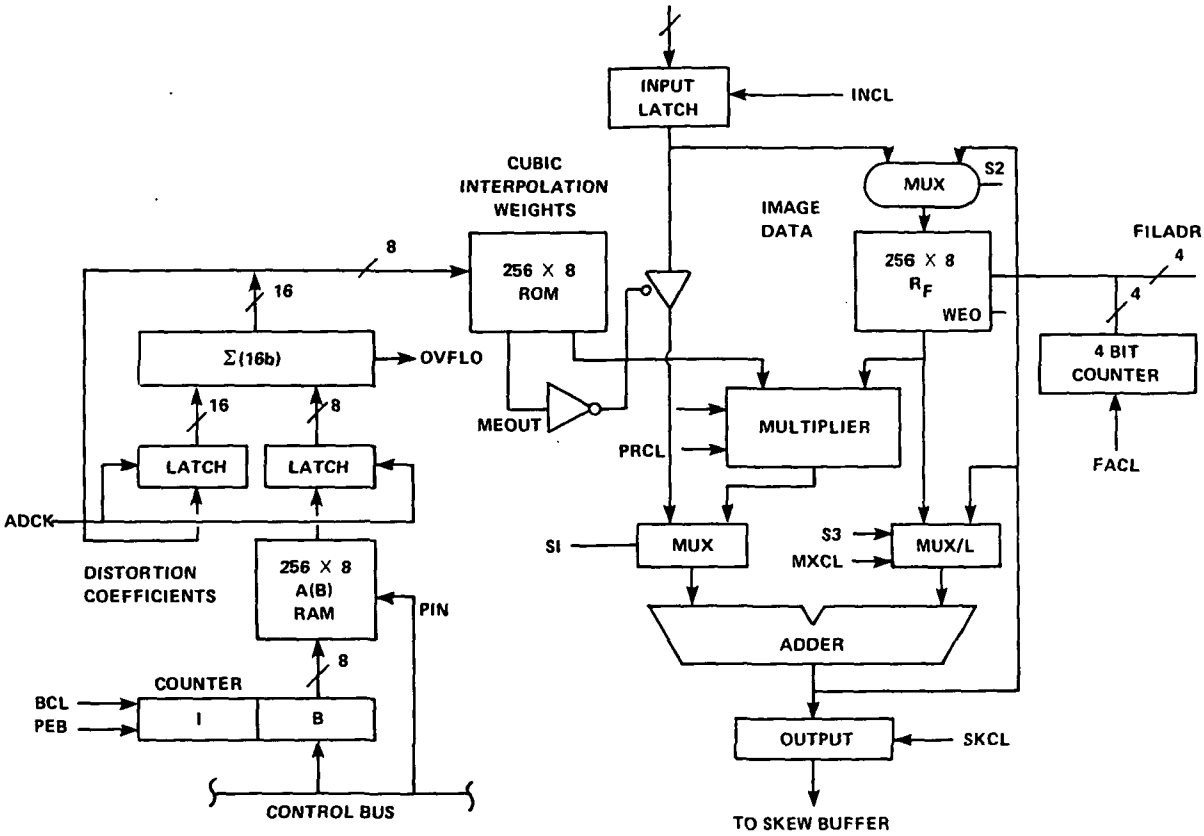



Figure 3-5. Along-Scan Resampling Algorithm (Each Band) [1-1]




CONTROL SIGNAL

| | | | | | |
|---|---|---|---|---|---|
| 1. | COUNTER 1,B | PEB | 9. | MULT IN CL | MUCL |
| 2. | ADDRESS ACCUM | ADCL | 10. | MULT OUT CL | PRCL |
| 3. | INPUT LATCH | INCL | 11. | COEF ROM OUT/RAR OUT | MEOUT |
| 4. | REG FILE MUX SEL | Se | 12. | REG FILE WE | RFWE |
| 5. | ALU MUX SEL | S1 | 13. | FILE SCAN ADDRESS | FACL |
| 6. | ALU M/L SEL | S3 | 14-17. | FILE ADDRESS | FILADR (4 BITS) |
| 7. | RAD RAM WE | RWE | 18. | CLOCK & COUNTER | BCL |
| 8. | MUX/LATCH CP | MXCL | 19. | SKEW BUFFER | SKCL |

MEMORY SIZING

A(B) RAM NEEDS 100 COEFF 1 FOR EACH 100 LINE SEGMENTS

$R_F$ (SAMPLE FILE) = $^{16}_{SENSOR} * {}^{16}_{DELAY\ LINE} = 256$

INTERPOLATION COEF = $64_{VALUES} * 4$ PIXELS = 256

RADIOMETRIC RAM = $16_{SENSOR} * 32_{SEGMENT} * 2_{COEF} = 1024$

Figure 3-6. Along-scan Processor [1-1]

Useful additions to the hardware documentation are functional-level block diagrams of the subsystems. Such a diagram for the along-scan processor is shown in Figure 3-7. This diagram can then be refined in a top-down manner into the detailed version shown in Figure 3-6. This subsystem requires four machine cycles to perform each interpolation, and functions as follows: (Refer to Figure 3-7)

Cycle 1: Input a new pixel value, simultaneously storing it in the local file and forming the product of this value with the currently addressed cubic convolution weight. Increment the 2-bit input pixel counters.

Cycle 2: Form the product of oldest pixel in local file with currently addressed weight. Add this to the previous product. Increment the input pixel counters.

Cycle 3: Form the product of the next-oldest pixel in the local file with the current weight, and add this to the previous sum. Increment the input pixel counters.

Cycle 4: Form the product of the remaining pixel in the local file with the current weight. and add to previous sum. Output this sum to the skew buffer. Increment the output pixel counter. Compute a new distortion DX. Test for DX>1.

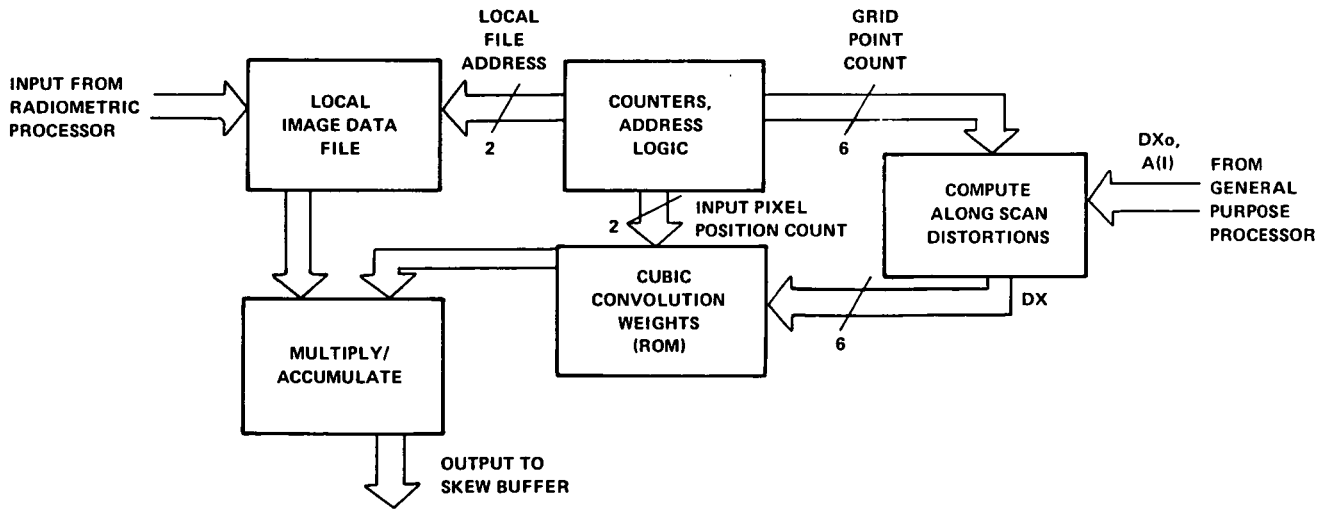



Figure 3-7. Along-scan processor functional block diagram

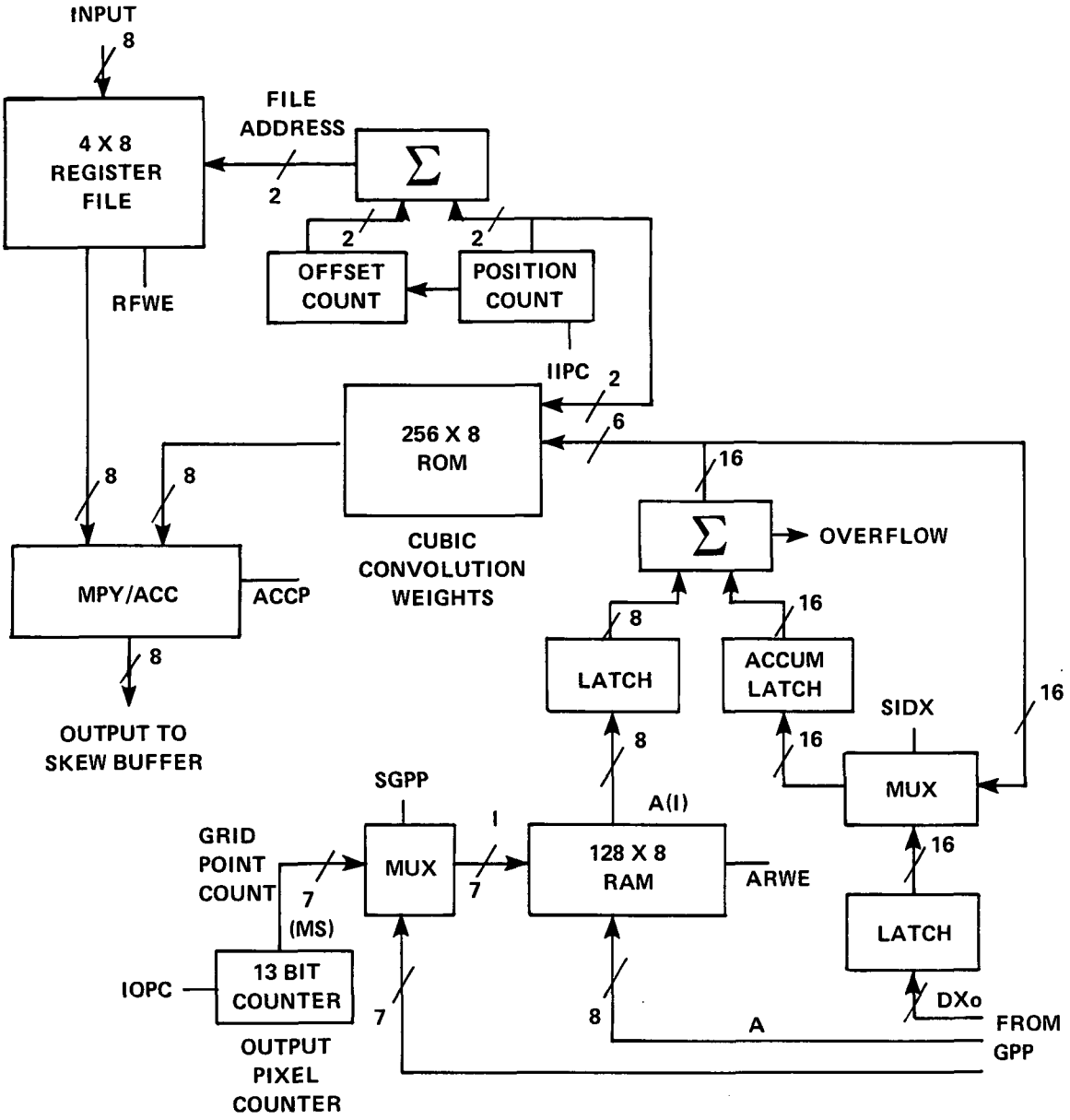
If the test in cycle 4 for DX>1 is true, indicating that no output pixel falls between two input pixels, then an alternate 4-cycle sequence is executed. In this sequence, DX is decremented by one and a new input pixel is read in, but no interpolation is performed and no pixel is output.

As indicated by TRW, the sample period is 600 ns. Thus, the time allotted for each machine cycle is 150 ns. The main functional blocks will now be discussed in detail.

The purpose of the local image data file is to buffer the number of pixels required in the interpolation process, in this case four. In the TRW implementation, this local file also functions to compensate for a displacement between alternate scan lines. In addition, the incoming scan lines for that implementation were multiplexed across sixteen sensors, thus imposing additional size and complexity requirements on the local file design. The TRW documentation does not indicate how these functions are handled, nor is the operation of their local image data file clear at this time.

In the present discussion, buffered line scanner sensor output is assumed, thus greatly simplifying the requirements of the local file. It is assumed that pixel values will be presented to the processor in order, one line at a time. The four most recent input pixels must be saved in such a way that their relative positional information is preserved. One scheme for implementing this without using more than four machine cycles is illustrated in Figure 3-8. Since a new input pixel will be stored in the file on every fourth cycle, the file address must point to the location of the oldest pixel in the file at this point so that it can be written over by the new pixel. This is accomplished by using a 2 bit pixel position counter which is incremented each machine cycle and a 2 bit offset counter which is incremented in every fourth cycle. Thse two counters can actually be implemented as one 4 bit counter as shown. Their sum (ignoring overflow) is used to access the register file. Relative pixel position information is preserved in the pixel position counter, which can then be used in conjunction with the distortion DX to access the cubic convolution weight ROM. Note that the register file must be of the type that can be read and written simultaneously, with a fast access time, since on every fourth cycle the data being written will also be read out and used in forming a product.

The multiply/accumulate function can now be performed on a single chip, further reducing the complexity of the design from that proposed




CONTROL LINES

| | |
|---|---|
| RFWE | REGISTER FILE WRITE ENABLE |
| IIPC | INCREMENT INPUT PIXEL COUNT |
| ACCP | ACCUMULATE PRODUCTS |
| SIDX | SELECT INITIAL DX |
| ARWE | A RAM WRITE ENABLE |
| SGPP | SELECT GPP ADDRESS |
| IOPC | INCREMENT OUTPUT PIXEL COUNT |

Figure 3-8. Along-Scan Processor Detail

by TRW. Chips are now available off-the-shelf which can perform an 8 bit multiplication and addition in 70 nanoseconds (see Table 3-2). These devices have on-chip registers and a flexible control structure, allowing the required sum-of-products operation to be implemented with no additional gates or latches.

This distortion computation is implemented in a straightforward manner, following the algorithm depicted in the flow chart (Figure 3-5). As every 64th output pixel is counted, the grid point counter is incremented, thus indexing a new A(I) value from the RAM. These values represent the average distortions over the 64-pixel intervals, and are accumulated in the adder-latch circuit to produce the distortion estimate DX for the current output pixel. This 6 bit quantity is then used in conjunction with the 2 bit pixel position count to access the cubic convolution weight ROM. The multiplexer is utilized to enable the accumulator latch to be loaded with the initial distortion at the start of a new scan. When DX becomes greater than or equal to one, the overflow signal is asserted, and the controller initiates the 4 cycle sequence in which a new pixel is input without generating a new output pixel. The distortion total (DX) is implicitly reduced by one since the carry bit (the integer part) is lost. Note that the A(I) RAM must be loaded by the general-purpose processor (GPP) in between accesses by the along scan processor. For this reason, a latch is used to store the currently needed value of A(I), leaving the RAM free for writing in between accesses to acquire a new A(I). These accesses will occur once for each output grid point (once every 64 output pixels), or approximately once every 38 microseconds, more than sufficient time for the GPP to write a new value to the RAM.

Table 3-3 illustrates the operation of the along-scan processor of Figure 3-8. The flow of data throughput processor is exemplified using input pixels labelled A, B, C, etc. The example begins with the start of a new scan line, at which time a special control sequence is executed which reads in four new input pixels without generating an output, and the initial distortion $DX_o$ is stored in the distortion accumulator. After this the cycle continues in the normal manner, except for occasional interruptions by the occurence of distortion overflow.

TABLE 3-3. Example of operation of along-scan processor shown in Figure 3-8

| CYCLE | MAJOR CONTROL LINES ASSERTED | OFFSET COUNT | POSITION COUNT | REG FILE ADDR | INPUT PIXEL STORED AT FILE ADDR | COMMENTS |
|---|---|---|---|---|---|---|
| 1 | IIPC, RFWE, SIDX | 0 | 0 | 0 | A | SCAN LINE INITIALIZATION SEQUENCE: DX←DX$_o$ |
| 2 | IIPC, RFWE, ACCP | 0 | 1 | 1 | B | USE FOUR NEW PIXELS IN INTERPOLATION; |
| 3 | IIPC, RFWE, ACCP | 0 | 2 | 2 | C | OUTPUT=A×W(0, DX$_o$)+B×W(1, DX$_o$)+C×W(2, DX$_o$) |
| 4 | IIPC, RFWE, ACCP, IOPC | 0 | 3 | 3 | D | +D×W(3, DX$_o$) |
| 1 | IIPC | 1 | 0 | 1 | B | NORMAL CYCLE: |
| 2 | IIPC, ACCP | 1 | 1 | 2 | C | DX←DX+A(I); |
| 3 | IIPC, ACCP | 1 | 2 | 3 | D | OUTPUT=B×W(0, DX)+C×W(1, DX)+D×W(2, DX) |
| 4 | IIPC, ACCP, RFWE, IOPC | 1 | 3 | 0 | E | +E×W(3, DX) |
| 1 | IIPC | 2 | 0 | 2 | C | |
| 2 | IIPC, ACCP | 2 | 1 | 3 | D | |
| 3 | IIPC, ACCP | 2 | 2 | 0 | E | NORMAL CYCLE |
| 4 | IIPC, ACCP, RFWE, IOPC | 2 | 3 | 1 | F | |
| | | | | | | OVERFLOW DETECTED (DX>1): |
| 1 | IIPC | 3 | 0 | 3 | D | SPECIAL OVERFLOW SEQUENCE: |
| 2 | IIPC | 3 | 1 | 0 | E | READ ONE NEW INPUT PIXEL; |
| 3 | IIPC | 3 | 2 | 1 | F | NO INTERPOLATION; |
| 4 | IIPC, RFWE | 3 | 3 | 2 | G | NO OUTPUT |
| 1 | IIPC | 0 | 0 | 0 | E | |
| 2 | IIPC, ACCP | 0 | 1 | 1 | F | |
| 3 | IIPC, ACCP | 0 | 2 | 2 | G | NORMAL CYCLE |
| 4 | IIPC, ACCP, RFWE, IOPC | 0 | 3 | 3 | H | |

## 3.5 Skew Buffer

The skew buffer is a system of memory and addressing logic designed to interface the along-scan processor to the across-scan processor. The term skew refers to the fact that the scan lines are not exactly parallel to the horizontal axis of the output grid, but are "skewed" slightly due to various error sources. TRW identified the major source of skew in a thematic mapper-based system as variation in the scan duration. This source of skew is virtually eliminated in a line scanner-based system since each entire line is imaged instantaneously, and the timing of the imaging process in the along track direction is controlled electronically rather than mechanically. Some skew will still be realized in a line scanner based system, primarily due to attitude or "pointing" errors.

The consequence of skew on the design of the buffer is increased storage requirement. Note that even if no skew were anticipated, the buffer memory requirement would be four full lines data since the across-scan processor requires four adjacent pixels in the vertical direction for each interpolation, while the along-scan processor outputs sequentially in the horizontal direction one line at a time. If there is skew amounting to as much as N pixels per line (deviation from parallel) in either direction, the buffer storage requirement is increased by 2N full lines of data (6000 bytes per line).

The skew buffer operation is shown schematically in Figure 3-9 and functionally in Figure 3-10, showing two "snapshots" of the skew buffer memory during operation. Data is written into the memory horizontally, one scan line of data to one horizontal line of memory. Data is read out of the buffer four vertically adjacent pixels per interpolation cycle, moving on to the next column with each new cycle. Information on the amount of skew between the input and output lines is used to determine which four pixels in a given column should be used in the current across-scan interpolation. Using this information, the skew buffer output addressing logic will occassionally shift the base address of its four-pixel read sequence up or down (depending on the direction of skew). In the example illustrated in Figure 3-10, a 12-line buffer is shown, thus allowing for maximum of 4 pixels of skew in either direction. Memory locations are labelled by a number which indicates which input scan line the pixel data stored in it came from.

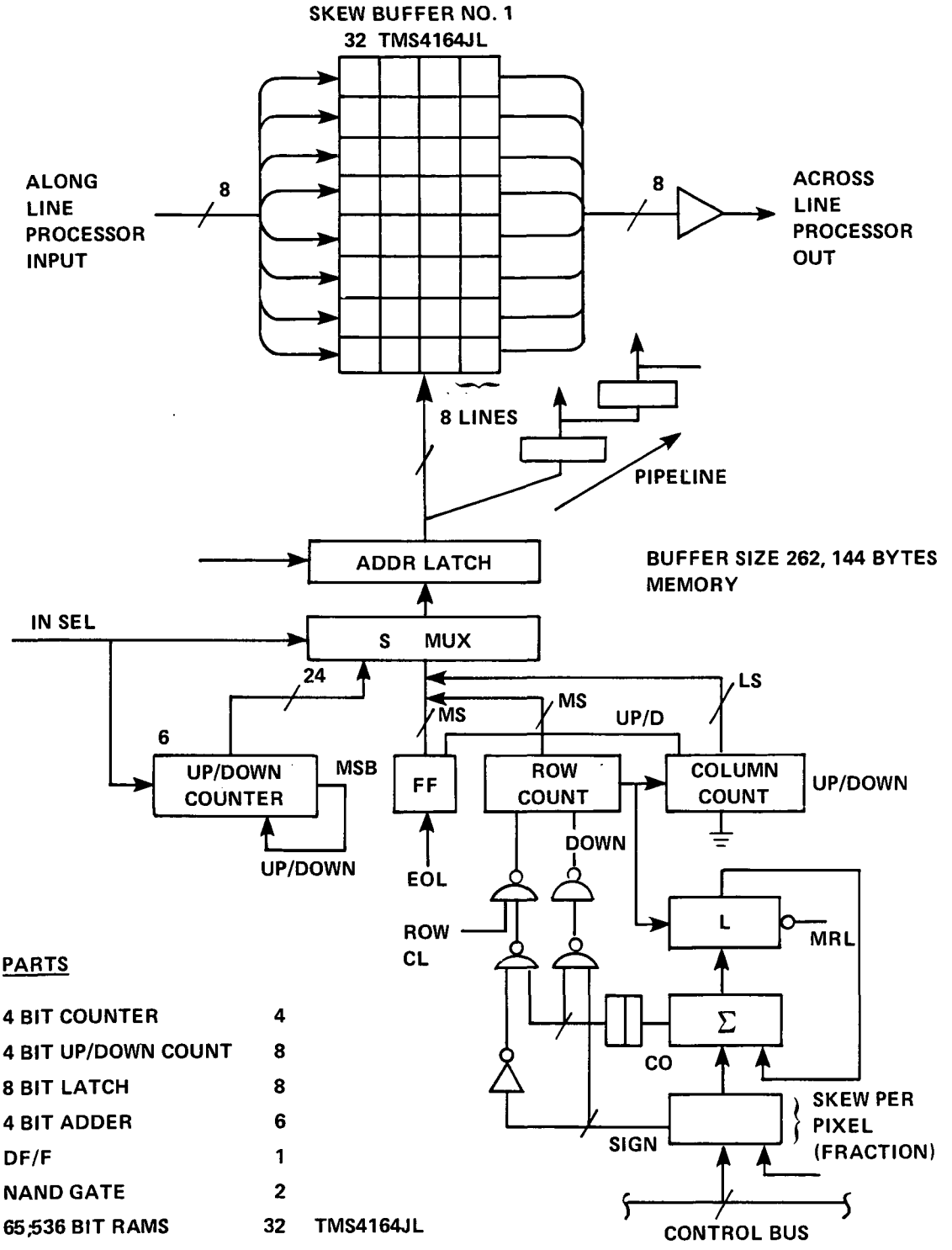



Figure 3-9. Skew Buffer And Address Processor. [1-1]

Encircled groups of four pixels represent data which have been read from the buffer and used by the across scan processor to interpolate a single output pixel. Refer also to the flow chart in Figure 15 to understand this example. Note that the write operations are performed in the same column as the read operations, but the timing is such that the current input pixel is written to the buffer before the read operations are performed.

Figure 3-10A shows the case of the output line being skewed the maximum amount in the positive direction. For reasons of clarity, relatively few pixels per line are illustrated in these examples, thus greatly exaggerating the amount of skew per pixel. At the point in time that this "snapshot" was made all of the encircled groups of four pixels have been read out of the buffer and interpolated by the across scan processor. As this was being done, new intermediate pixel values from scan line 12 were being written into the buffer in row 8. A new cycle has just started in column M, where another intermediate pixel from scan line 12 has been written into row 8 of the buffer, and the next step will be to read the four output pixels from that column labelled 9, 10, 11, and 12. In Figure 3-10B the other extreme case is illustrated: maximum negative skew. In this example, we see the effect of wraparound addressing. This causes the buffer to appear continuous across boundaries, the first row being treated as if adjacent to the last, thus utilizing the available storage most efficiently. Note that if in either of these two cases the skew had been greater than shown, the point at which writing was being done would be "lapped" by the read operations, causing invalid data to be read. This data would be invalid in the sense that the four pixels used in an interpolation would not be adjacent to each other in the input space. Since this could produce blatantly incorrect values of the output pixel intensity estimates, it is important that skew buffer sizing be designed for near worst-case expectations of skew.

The flow chart in Figure 3-11 outlines the algorithm for implementing the operations illustrated in Figure 3-10. Note that all computations on row indices are done modulo N, where N is the number of rows in the buffer. This will implement the "wraparound addressing" effect referred to earlier. At the start of a new line, the column counter COL is reset to 1 and the write row index WROW is incremented (modulo N). The read row index RROW is set to be behind the write row index by an amount equal to the maximum skew to be accommodated in either direction plus 3. This combined with a buffer sizing of 4+2 x MAXSKEW will assure that as long as the skew is

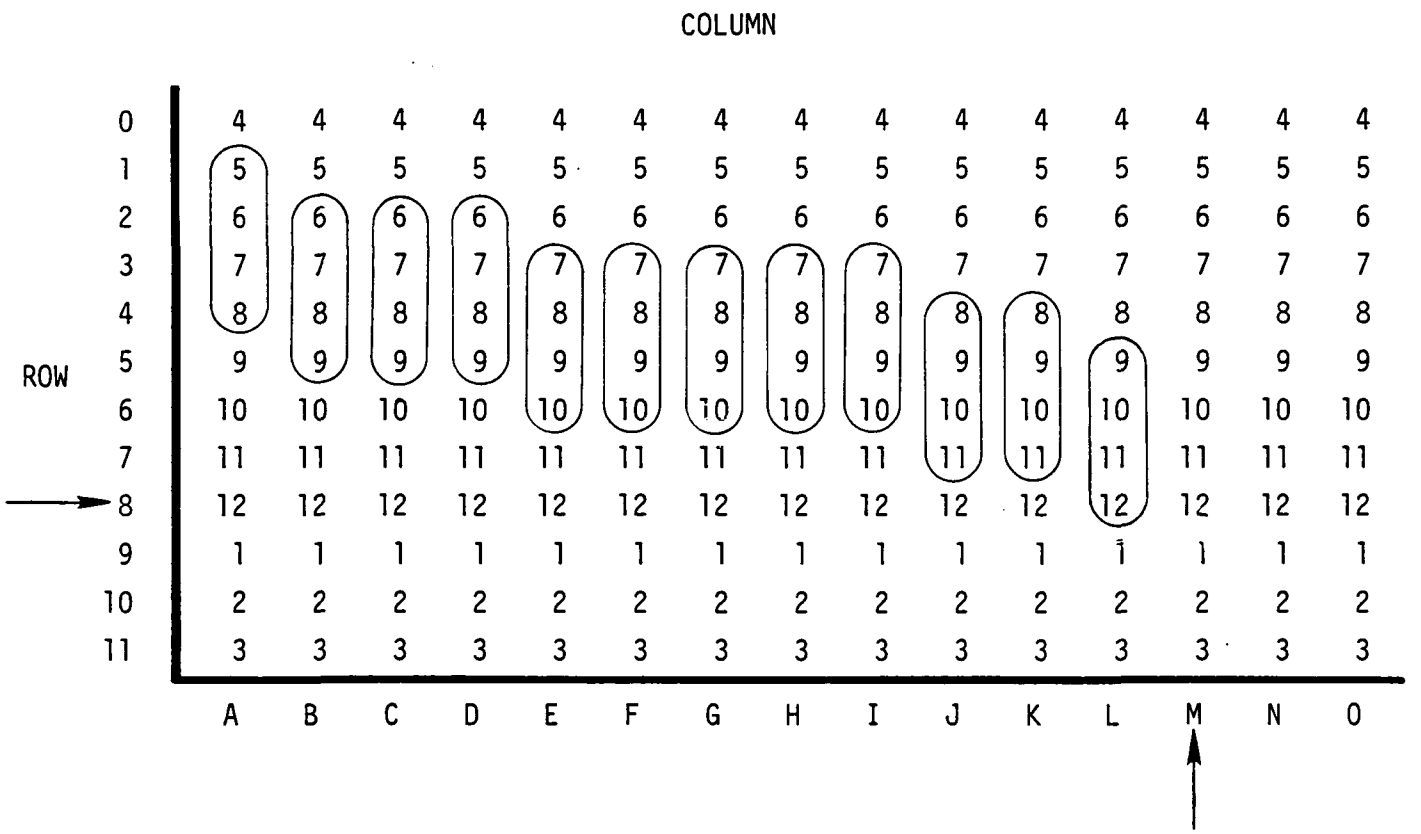



Figure 3-10A. Example of Skew Buffer Operation: Maximum Positive Skew

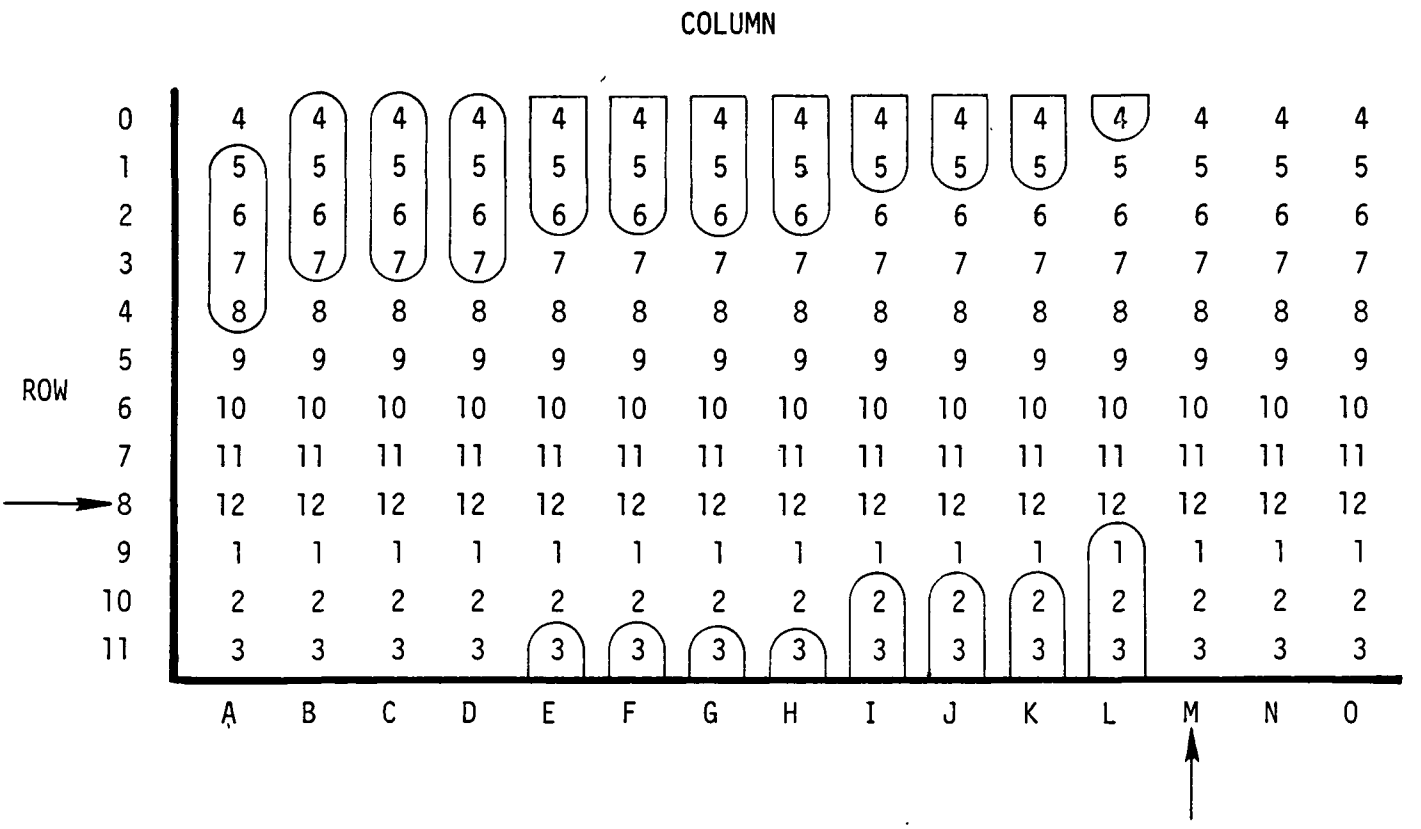



Figure 3-10B. Example of Skew Buffer Operation: Maximum Negative Skew

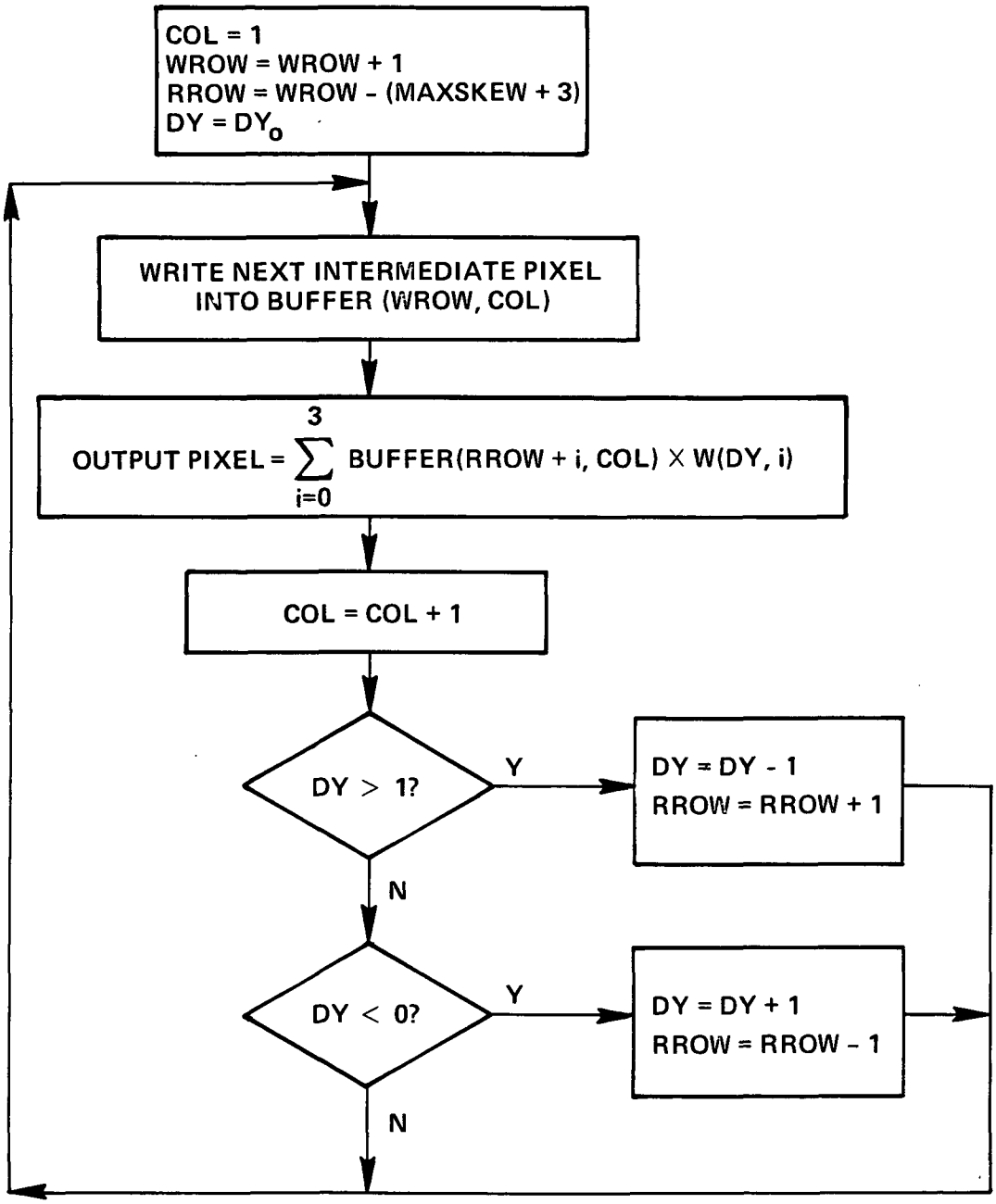



Figure 3-11. Flow chart of skew buffer and across-scan processor operation.

within the design limits no overlapping of new and old data will occur. The value of the vertical distorition DY is set to its initial value $DY_o$ as determined by the general purpose processor (GPP). The value of $DY_o$ will be periodically updated by the GPP, probably at intervals of M lines where M is the number of lines imaged simultaneously by the line scanner. At these points the skew per pixel estimate S will also be updated.

A functional diagram of the hardware to implement this algorithm and the across-scan resampling is shown in Figure 3-12. This is somewhat similar to the TRW designs as discussed in the next section. The major hardware differences are the presence of a 4-byte local input file and associated addressing logic in the TRW design and the distinction made in the TRW design between the vertical distortion per pixel "$\Delta Y$" and the "skew per pixel fraction $\Delta L$". This latter discrepancy is somewhat puzzling since $\Delta Y$ and $\Delta L$ must in fact be the same quantity and it is unclear why the TRW design incorporates the same vertical distortion logic (consisting of two latches and an adder) in both the skew buffer address processor and the across-scan resampling processor. In the design shown in Figure 3-18, this logic is shared by both processors. The absence of a local input file in the present design is made possible by the fact that for each interpolation cycle all four required input pixels are read directly from the skew buffer. In the TRW design, each block of 16 lines was processed one column at a time, requiring 19 input pixels to be utilized in computing 16 output pixels. The proposed timing scheme (Figure 3-13) required every 16th cycle to include four read operations from the skew buffer with no write operations to it. This would cause the input address processor to fall quickly behind the output processor, since only 15 new input pixels are written to the buffer while 16 output pixels are computed. No explanation given in the documentation resolves this apparent timing problem. It appears that an effort was made to keep the required number of cycles per interpolation down to 4, since the resulting 150 nanosecond cycle time was pushing the state of the art in memory speeds. This resulted in increased hardware complexity and possible timing problems. In the present design a 5 cycle interpolation interval is proposed, in which one write operation and four read operations are executed, resulting in a 120 nsec. cycle time. It is believed that in the anticipated time frame high-density RAM of sufficient speed will be available.

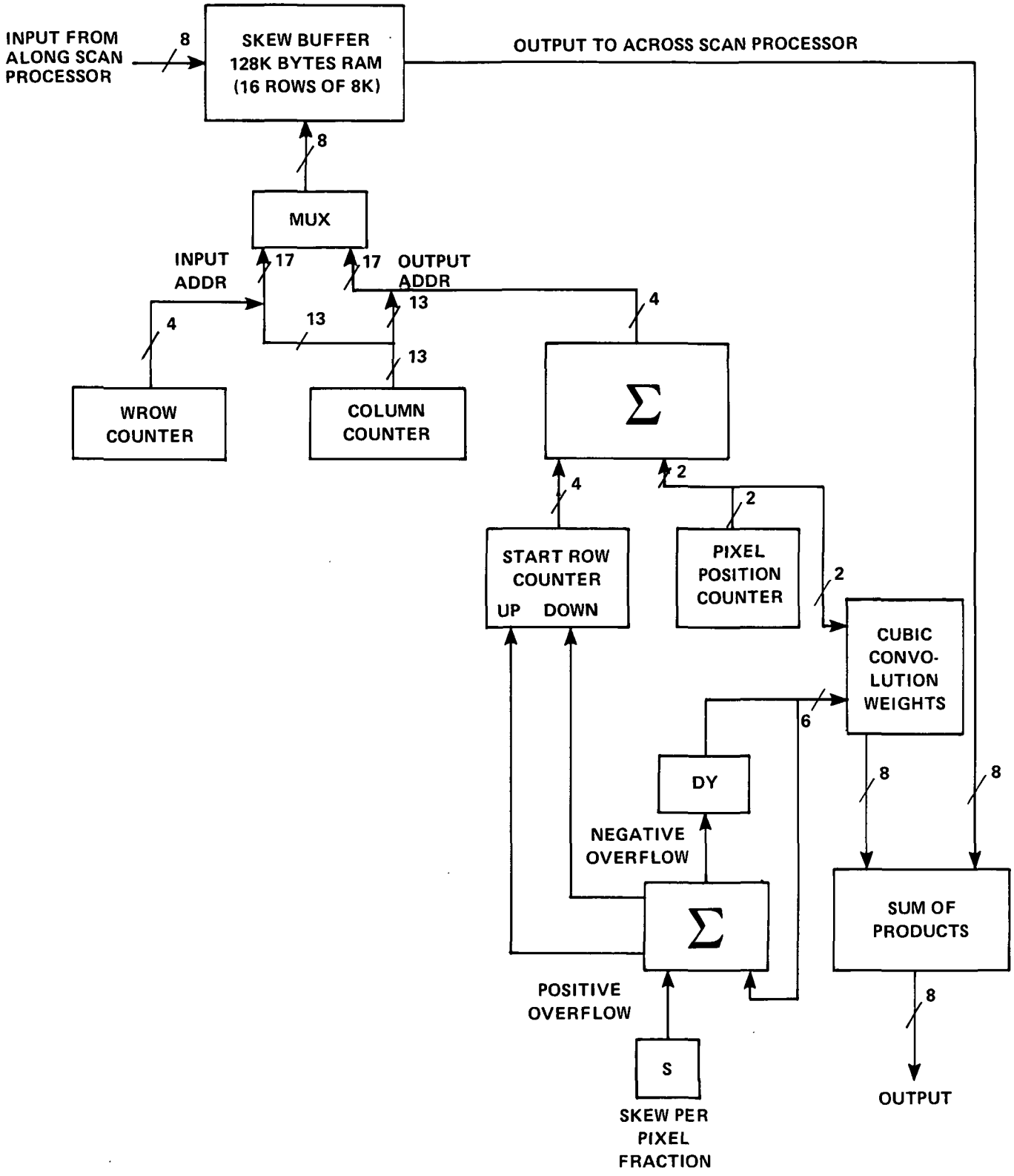



Figure 3-12. Skew buffer and across-scan processor.

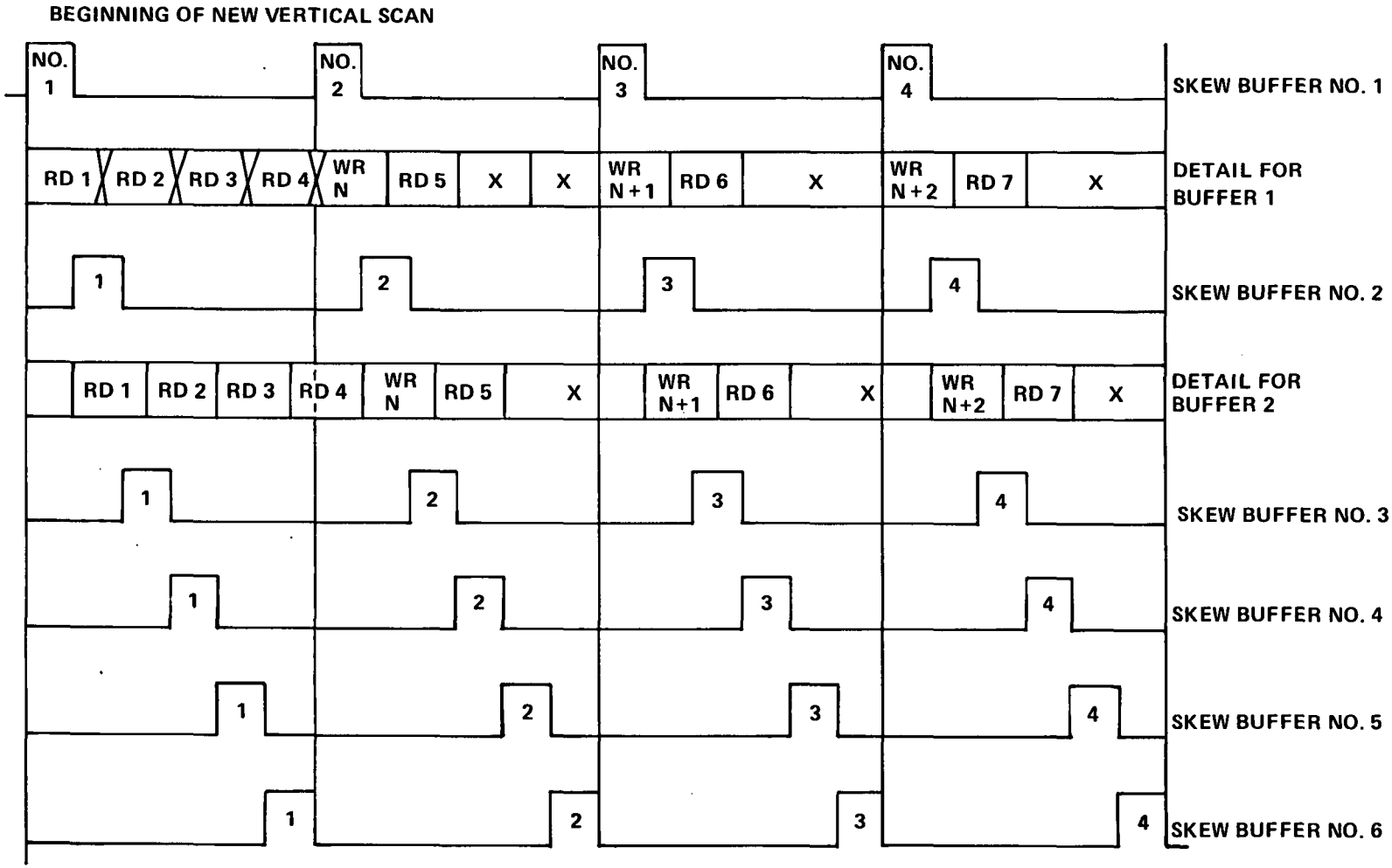



Figure 3-13. Skew Buffer Pipeline Timing [1-1]

## 3.6 Across-Scan Processor

The algorithm and implementation of the across-scan processor proposed by TRW are shown in Figures 3-14 and 3-15.

Since most of the functions of the across-scan processor have been implemented in the skew buffer system, the remaining portion is very simple. Both the accessing of input pixels and the vertical distortion computation have been discussed in the skew buffer section. All that remains is the sum-of-products operation and the cubic convolution weights addressing. These operations are implemented exactly as in the along-scan processor. The complete skew buffer and cross-scan processor diagram has been shown in Figure 3-12. Control lines are not shown here for clarity, but will function in a straightforward manner to implement the algorithm detailed earlier. Note that the registers DY and S must have the capability of being loaded from the GPP.

## 3.7 Microsequencer

For completeness, the microsequencer proposed by TRW for overall control is included as figure 3-16.

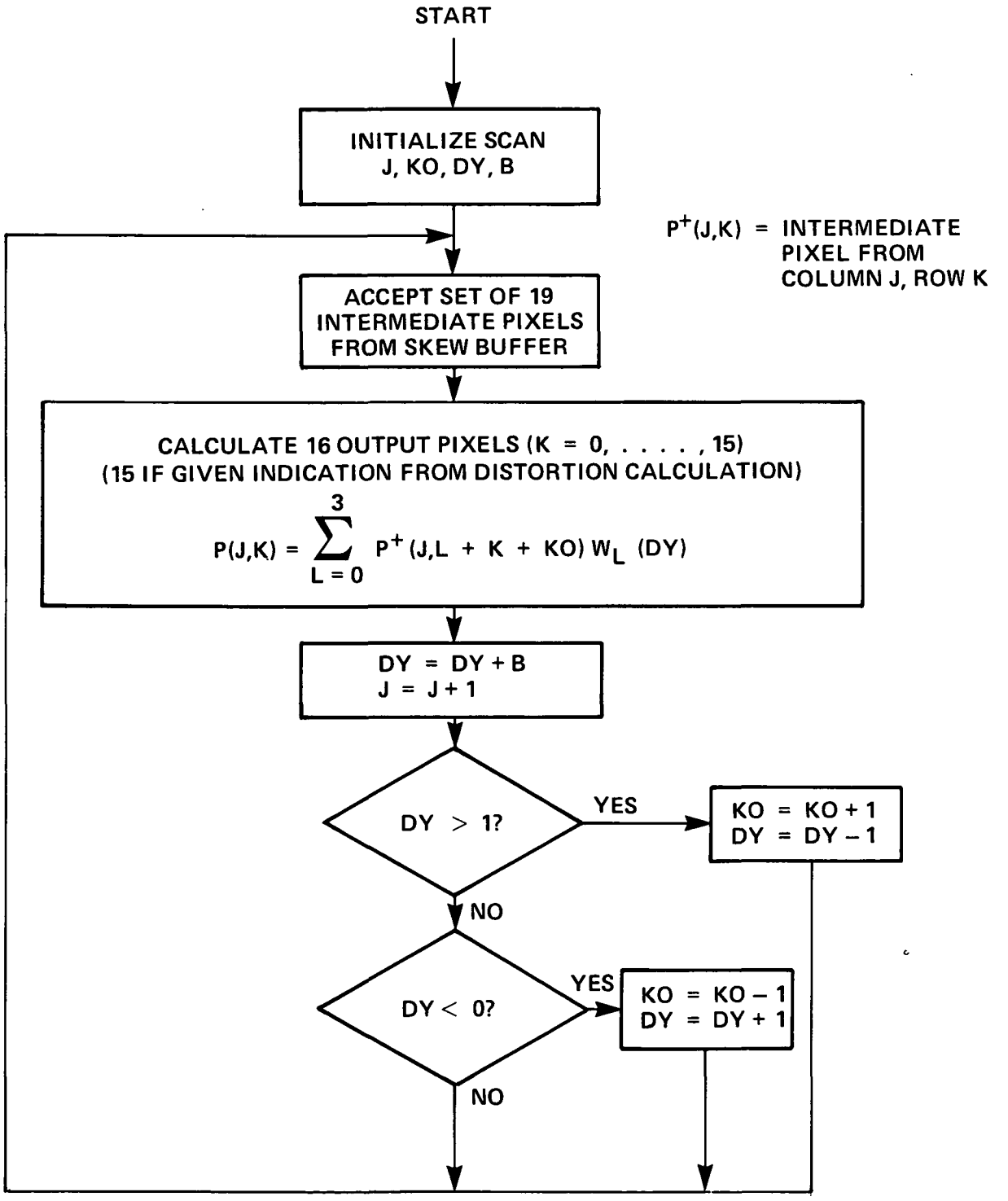



Figure 3-14. Cross-Scan Resampling Algorithm (Each Band) [1-1]

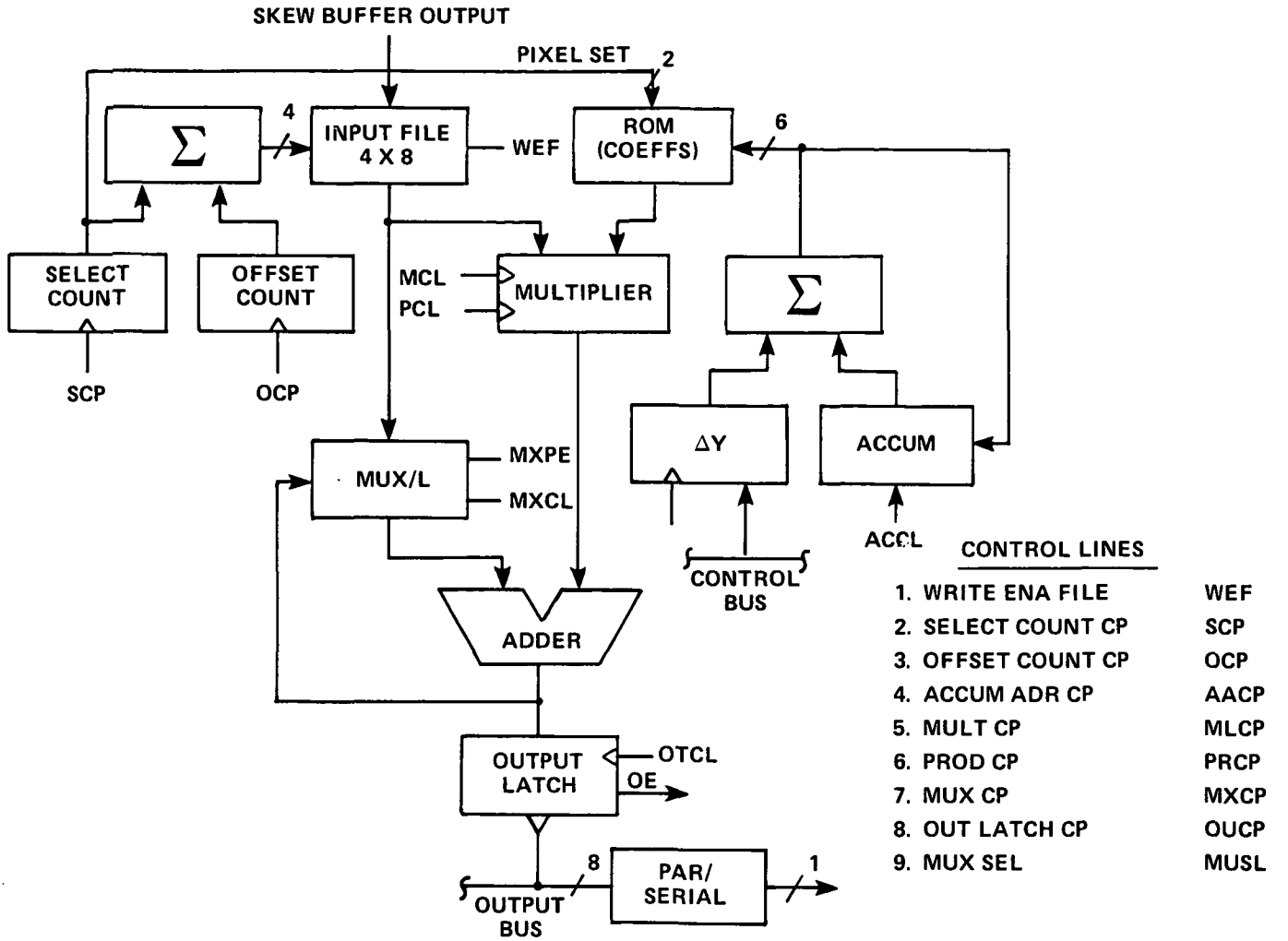



Figure 3-15A. Cross-Scan Processor [1-1]

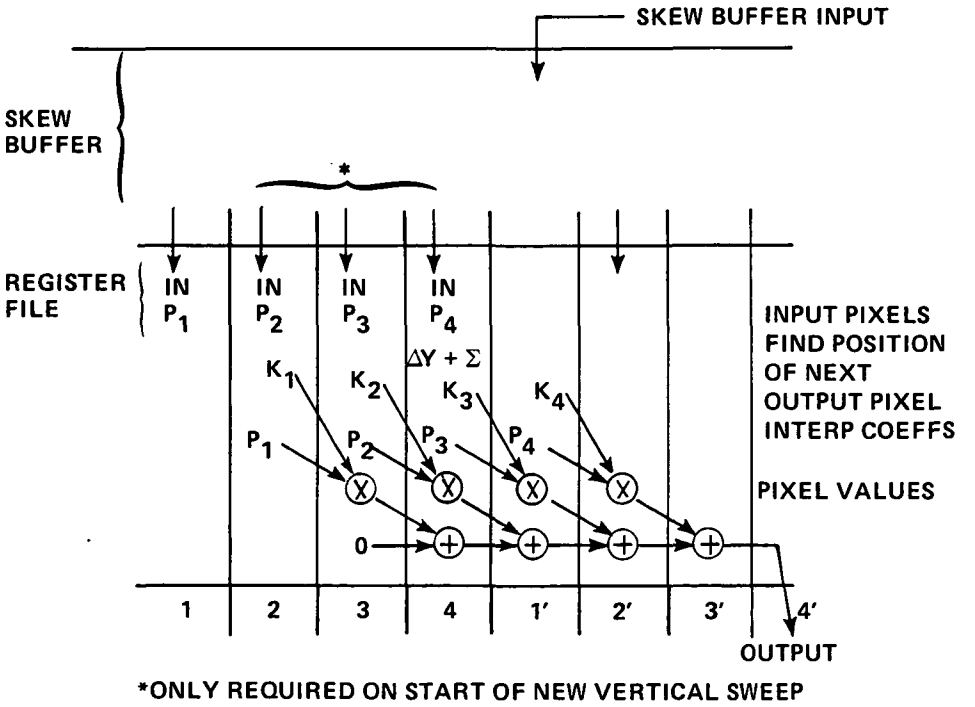
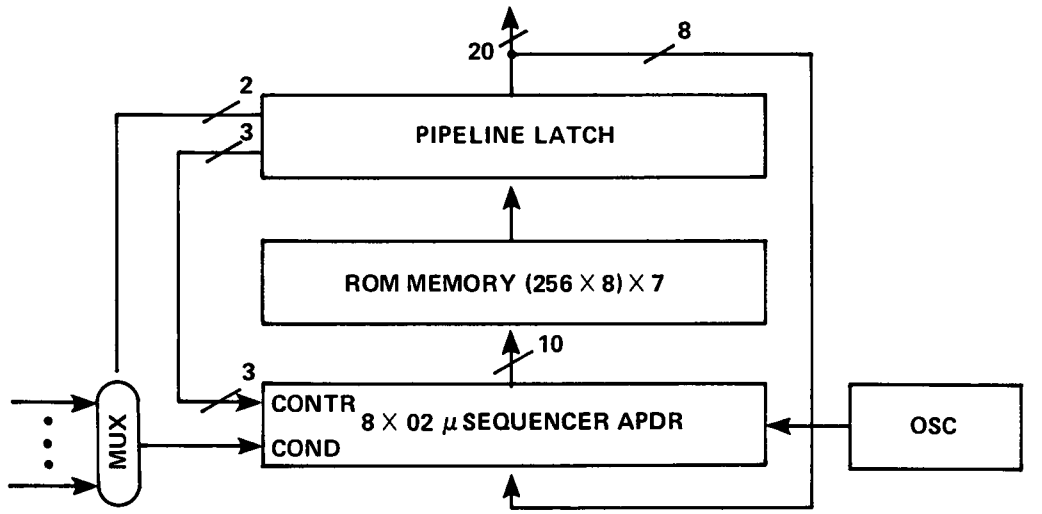
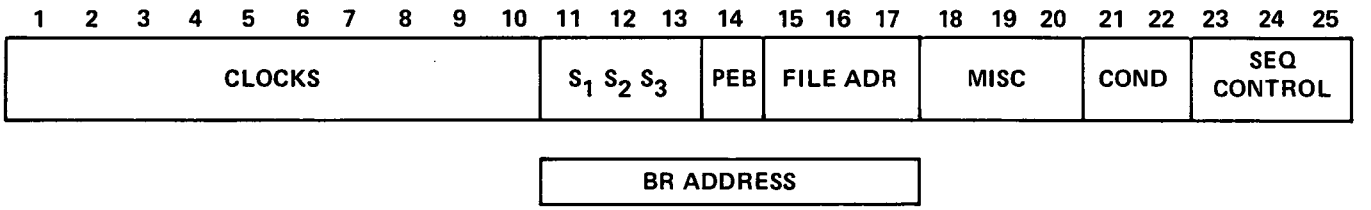



Figure 3-15B. Cross-Scan Processor Operation [1-1]




| | | |
|---|---|---|
| 1. 256 X 8 ROM | 6 | |
| 2. μ SEQUENCER | 1 | 8 X 02 |
| 3. 4:1 MUX | 1 | |
| 4. OCTAL LATCH | 3 | 74LS374 |
| 5. CLOCK OSC | 1 | |

Figure 3-16. Microcontroller [1-1]

# 4.0 SPECIFICATION DEVELOPMENT

## 4.1 Introduction

One task under the study was to develop, in close coordination with NASA, a specification to be used for the procurement of IAS prototype demonstration hardware. This section presents, in narrative form, the major considerations to be incorporated in such a specification. The considerations included here address only the radiometric correction and geometric distortion aspects of the IAS, however, a more comprehensive specification has been developed as an iterative process between RTI and NASA and currently is a part of the procurement package. The following paragraphs were designed to be used as candidate text for this package and as a result are somewhat redundant with respect to other sections of this report.

## 4.2 Background

NASA has throughout the past decade actively conducted programs utilizing earth observing sensor platforms as a mechanism for reconnaissance of earth resources and for observation of the earth environment. As the spatial resolution of these sensors is increased to meet more demanding applications, the volume of data collected by one of these platforms becomes overwhelmingly large. Users are faced with ground processing delays sometimes ranging from weeks to months before usable data is available. To circumvent this, NASA has initiated the NASA End-to-End Data System (NEEDS) program with the objective of improving the efficiency and effectiveness of data throughput.

A primary element of the NEEDS program is the incorporation of on-board signal processing into the satellite sensor platform design. This concept will alleviate much of the demand for ground based data processing by real-time processing the data as it is acquired and by reducing the quantity of data which is telemetered to the ground processing facility. It is the goal of the Phase II NEEDS program to demonstrate this concept.

The emphasis of this procurement is the on-board signal processing hardware required to perform the pre-processing functions of radiometric correction and image registration. The processing requirement results

from a gradually varying departure from linearity of each of the photosensors. The requirement for image registration results from uncertainty in subsatellite position due to imperfect ephemeris control and from error in sensor pointing due to variations in attitude determination and control.

## 4.3 Objective

The objective of the Information Adaptive System Team Task is to design, develop, and demonstrate an adaptive data control and processing system which is capable of interfacing directly with earth resources and environmental monitoring sensors to provide on-board data control, formatting, calibration, preprocessing, data set selection, and information extraction.

The specific objective of this procurement is to provide demonstration hardware to perform the preprocessing functions of radiometric correction and image registration in support of this primary objective.

## 4.4 Technical Requirements

### General

Figure 4-1 shows the principal components of the overall Information Adaptive System. The emphasis of this procurement resides in those functions provided by the data preprocessor. As mentioned elsewhere, these consist of radiometric correction and geometric correction. The functional organization of the preprocessor is shown in Figure 4-2. This organization is the result of a previous study and as such is representative. Bidders are encouraged to review this approach and to take exception if appropriate and to suggest alternatives where cost-effectiveness can be increased. Notice that the general scheme is to perform a pierce-point calculation at several points and then to interpolate to achieve ground distance along-scan. A skew buffer is incorporated to compensate for yaw attitude variations and earth rotation. Resampling is then performed along track to rectify the data to an appropriate map projection. The resampling processor is pipe-lined with parallelism for the seven bands.

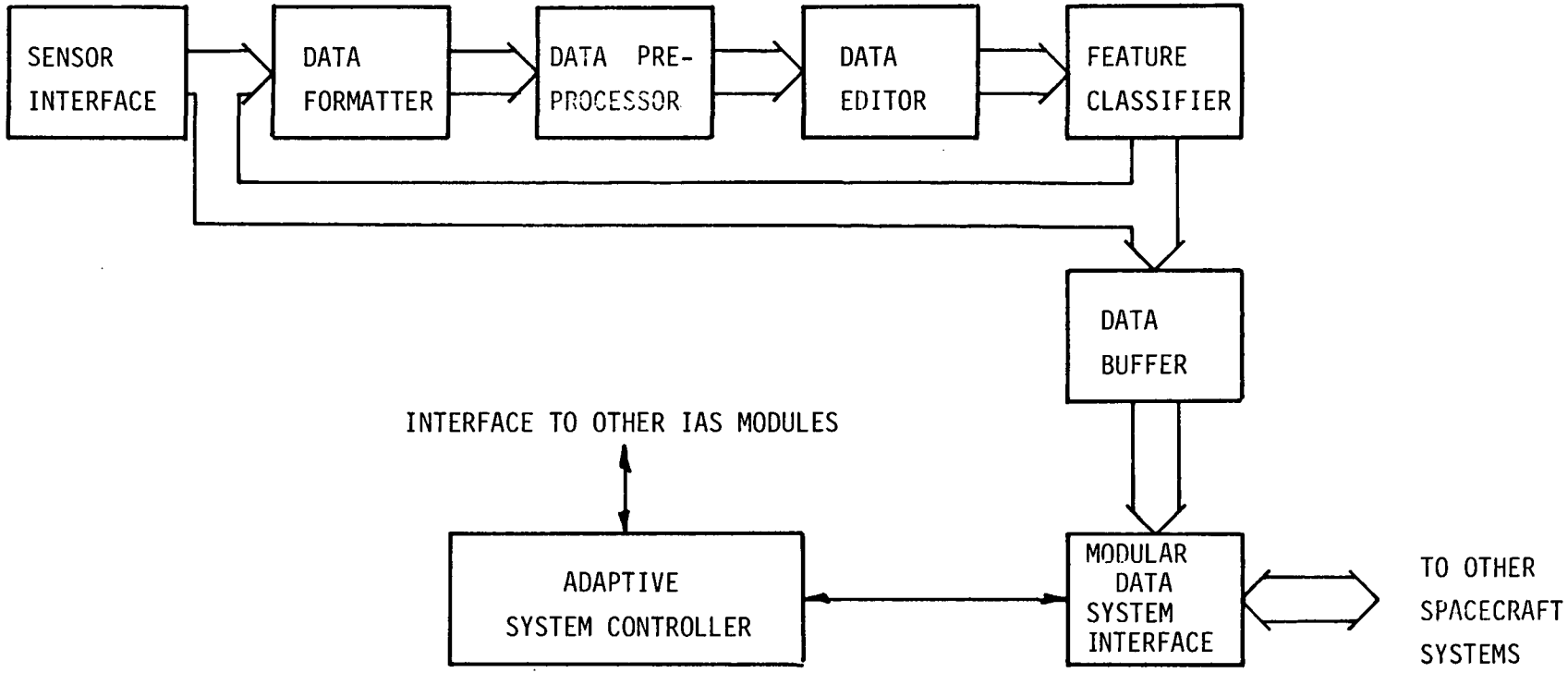
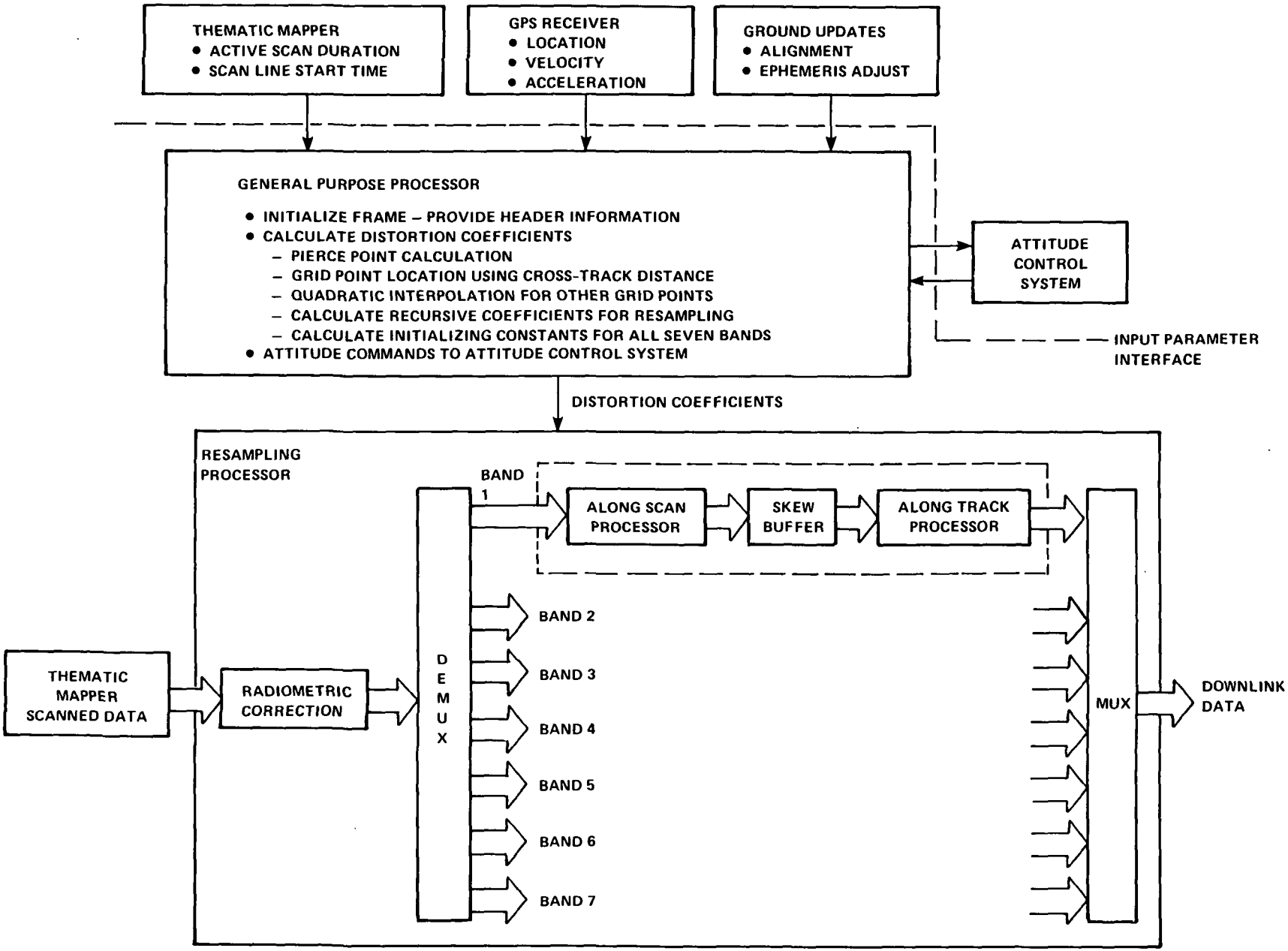



Figure 4-1. INFORMATION ADAPTIVE SYSTEM BLOCK DIAGRAM

FUNCTIONAL OPERATION OF REGISTRATION PROCESSOR




*Double lines represent image data.

Figure 4-2. Functional Operation of Registration Processor [1-1]

Radiometric Correction -

The interpolation circuit operates by each segment of a piecewise linear curve being represented by its characteristic equation -

$$y = a(x - {}^{b}/_{a})$$

and with the 3 most significant bits of the intensity being used as address to the RAM to look up "a" and "b/a".

The design power requirements [1-1] are as follows:

| | |
|---|---|
| memory | 400 mv |
| multipliers | 3300 |
| latch | 544 |
| counter | 93 |
| gates | 8 |
| shift register | 136 |
| flip flop | 15 |
| | 4496 mw/band (∿32 watts total) |

An alternative design would simply use 256 bytes of RAM to store the entire calibration curve, rather than just breakpoints.

For the 16 sensors on one band, a total of 4K bytes of RAM would be required. Using the same 54LS207 chips that TRW proposed, this design would consume 6400 mw per band (∿ 100 watts total). However, the following facts should be observed:

1) The use of bipolar memory chips is not reasonable, since there are now MOS RAMs available with sufficient speed and much lower power.
2) MOS technology can be expected to produce higher densities and lower power in the next 2 years, whereas the multipliers are pushing the state-of-the-art in speed and cannot be expected to change much in that interval.
3) A table look-up calibration scheme is much simpler to build and can be made much more tolerant to small drifts in speed of components than the rather complex interpolation scheme.

It is considered reasonable to project that power consumption with MOS technology is well under half the 6400 mw per band figure and that space-rated hardware is within the state-of-the-art within the anticipated time frame.

Geometric Correction -

General - The raw scan data exhibit spacecraft-dependent and perspective dependent distortions. To permit direct comparisons of

different images of the same region or of adjacent regions, the images must be registered with a map projection. This map projection should be based upon the reference ellipsoid (polar radius 6356.783 km, equatorial radius 6378.165 km). The map projection shall be definable as an invertible function of geocentric latitude and longitude. In determining the true geocentric latitude and longitude which is being scanned, the following factors are to be taken into account:

- satellite altitude above the reference ellipsoid
- variation of earth radius and curvature with latitude
- the direction of scanning with respect to the local normal to the earth's surface
- spacecraft attitude (including the alignment with the sensor module)
- spacecraft ephemeris
- optical distortion

Preferred Map Projection - There are additional constraints for an on-board image registration map projection. It must provide nearly constant scale throughout an image frame, utilize tractable computations and one axis of the map projection should be nearly aligned with the along-scan direction so that scan data storage does not become excessive.

On the other hand, some degree of distortion which is dependent solely upon latitude and longitude can be tolerated if the distortion is removable by subsequent re-registration on the ground. Indeed, some distortion is necessarily associated with the plane representation of a curved surface.

The Space Oblique Mercator Projection is a candidate map projection, but in its straightforward implementation, it is probably too involved computationally for on-board satellite use.

Another candidate is the Oblique Mercator Projection for the reference ellipsoid. For each image frame, the projection would be established with respect to the satellite orbit plane's intersection with the image at frame center. Earth rotation causes the ground track to wander from the transformed equator, so that some additional variation is introduced. This projection is probably still too complex for this application unless simplified algorithms are developed.

A simpler projection is the Oblique Mercator Projection for the sphere. The additional distortion resulting from this projection, rather than the one for the reference ellipsoid, is primarily latitude dependent if a fixed mean earth radius is assumed. The scale in ground distance units will vary slightly but images can be compared, because they have the same distortion at the same ground position. This map projection can be inverted for ground re-registration to further reduce scale variation within a frame.

The full calculation of the map projection coordinates of an image pixel is still time-consuming even for the latter projection. Thus, it may be desirable to choose one or more interpolation algorithms to "fill-in" between precisely-located points. It must, however, be remembered that it is the map projection which is being interpolated, not a direct, physical entity, such as ground distance.

This projection amounts to approximation of the reference ellipsoid by a single sphere of radius approximately 6370 km. Thus, the along scan map coordinate is proportional to ground distance on the approximating sphere (each point on the reference ellipsoid is associated with the unique point on the approximating sphere lying along the same line through earth center).

Computation of Grid Constants - If the latter map projection described in the previous section is chosen, then there are relatively simple interpolation algorithms which calculate ground distance on the approximating sphere.

Computer Compatibility - For each band, the resampling algorithm must be used to process about 225 rows, each comprising about 6167 pixels, in one second (this includes both the distortion calculations and the resampling interpolation).

The line scanner will output, for each pixel, a value which is the result of accumulating charge from photons emitted or reflected from this area. The output of the next pixel sensor will be proportional to the light from a nearby, non-overlapping area of the surface.

Geometric correction requires that the map coordinates of the centers of these areas, be known to within ± 15m prior to resampling. This could be accomplished by computing the map projection for each pixel scanned. However, this is likely to be beyond the speed capabilities of state-of-the-art computers. An acceptable alternative is to provide

special purpose hardware to perform a linear interpolation between points whose map coordinates have been found by more precise methods.

Each pixel thus requires two pieces of information; the intensity at that point (output from the radiometric correction system) and the map coordinates of the center of the pixel. An acceptable mechanism for maintaining the map coordinates is to compute a "distortion", the difference (in the along-scan direction) between the coordinates of center of the pixel being scanned and a reference grid point on the map projection.

Resampling for Intensity Correction - Since the output of a cell of the scanner is proportional to the average intensity from an area on the surface, it can be considered an estimate of the intensity (reflectivity) at a point, the center of the area.

Typically, the pixel centers (grid points) on the map projection do not align exactly with the centers of the pixels being scanned, and it is necessary to provide an estimate of the intensity function at a point other than the center of a scanned pixel.

The estimate of intensity at a point having coordinates x, y can be arrived at, by interpolation from neighboring pixels. In this context, x refers to the along-scan direction, and y refers to the along-track direction.

It is acceptable to perform the interpolation in the x and y directions separately.

The interpolation shall be based on a sum-of-products scheme. The estimated value of a pixel is the weighted average of the nearest four pixels on the same line. Those values are then to be interpolated in the along-track direction. The weights may be derived by a $\frac{\sin(x)}{x}$ interpolation scheme.

Studies to date indicate that a weighted sum of the nearest four pixels provide a good estimate of an intermediate pixel. A choice of $\frac{\sin x}{x}$ ($x + \pi$ for the pixel spacing) weights, provide an optimum estimate only under very restrictive assumptions concerning the sampling process. The contractor shall review and recommend alternative approaches based on actual sampling procedure and by the scanner. Final choice of an interpolation scheme shall be coordinated with the government prior to adoption.

The "along-scan processor" described in [1-1] represents an approach, but not necessarily a final design, which is acceptable to the government. The contractor is expected to review this design and provide an analysis and recommendation to the government prior to adoption.

Along-Track Interpolation - Just as in the along-scan direction, sampled pixel centers cannot be expected to align exactly with map projection grid points. Therefore, intensity interpolation must be done in the along-track direction.

After pixels have been registered in the x direction by the along-scan processor, they must be stored in a memory, so they may be interpolated in the y direction.

The skew described by TRW provides a mechanism for accomplishing this, however, some exceptions must be noted:

(1) Due to the "instantaneous" sampling mechanism of the line scanner, earth motion contributes to blur rather than skew, therefore it is not necessary to compute "skew per pixel" and modify the buffer addressing in this way.

(2) Using a four line buffer with address, wrap-around on lines should be sufficient if read-before-write timing is properly done.

The contractor is expected to provide a detailed simulation of the design to the government prior to hardware implementation.

Accuracy Requirements - The map projection shall be defined by invertible functions of geocentric latitude and longitude for points on the reference ellipsoid. One of the coordinates of the projection shall be in the along-scan direction, with the other essentially normal to it (a skew factor may be necessary to accommodate attitude error and the effect of non-normal scanning of the earth).

A full scan of data is accumulated during a single time interval and then read out sequentially, so that earth rotation produces a slightly selective blurring, but no skew.

The intra-frame variation for the ideal map projection shall be less than five (5) parts in 10,000. That is, there must be a defined scale factor for the frame so that ground distance on the reference ellipsoid is proportional to separation on the map projection to this degree of accuracy (much of this scale variation is "removable" by inverting the functions defining the map projection). The geometric correction shall be correct to within 0.25 pixel for spacing (1 $\sigma$) and to within 0.5 pixel for absolute registration (1 $\sigma$). That is, the map projection coordinates shall be obtained to this degree of accuracy.

In addition, at least three (3) pixels in each frame (separated pairwise by at least 0.25 frame) shall be "position-tagged." That is, their map coordinates, correct to 0.1 pixel (1 $\sigma$) shall be provided, allowing for more accurate re-registration on the ground.

Resampling interpolation should agree to 6 bits with the value obtained by 4 x 4 pixel calculations using the exact value of sin x/x.

Use of Ground Control Points - If the position-tagging described previously cannot be accomplished with the specified accuracy, then there must be a provision for position-tagging refinement to the required accuracy by correlation or Sequential Similarity Detection Algorithm (SSDA) methods when GCP's are available. Performance will be degraded when GCP's are not available.

## 5.0 RECOMMENDATIONS FOR FURTHER RESEARCH

In keeping with the overall goals of the NEEDS program, the following areas are suggested for further study:

1. The continued examination of state-of-the-art technology developments for application to satellite on-board signal processing. This is especially important in the light of the recent microelectronics advances being experienced in the VHSIC and VLSI areas. Any meaningful pursuit of this area should include a survey of related activities in the military community as well as developments in the private sector.
2. A key issue in the NEEDS Phase II hardware demonstration is the performance evaluation. It is important to recognize the proper evaluation procedures and perhaps more importantly the performance evaluation measures. It is recommended that evaluation standards be adopted and that uniform performance measures be established. These should consider image interpretation both numerically, as by computer, and subjectively, as by the human observer.
3. In that the scanning geometry associated with the thematic mapper is not representative of future sensors, the impact of linear and rectangular arrays on the on-board processing should be investigated. Again, a survey of systems employed by the military would provide a useful input.

# APPENDIX A

# HIGH SPEED INTERPOLATION OF SAMPLED DATA

# HIGH-SPEED INTERPOLATION OF SAMPLED DATA

## A.1 INTRODUCTION

The problem of reconstruction of continuous data from a few regularly spaced samples has been approached from several different viewpoints. The most familiar are polynomial interpolation and bandlimited interpolation. Digital interpolation does not generate a continuous reconstruction, but it can optimally simulate an increased sampling rate for bandlimited sequences (see [A-1]). In this paper, the bases of several interpolation techniques are examined, and some of the resulting specific techniques are compared for performance and for facility of digital implementation (for maximum speed, interpolation weights are stored in a lookup table, but if the weights must be computed as they are used, computer-efficient interpolators can be chosen). It is shown that, to some extent, an interpolator can be designed to perform well for a prescribed class of functions. In particular, near-bandlimited* interpolators can be designed for certain frequency response characteristics, much as windowed digital FIR low-pass filters are.

The interpolation may serve two purposes, reconstruction and noise reduction, and these may be somewhat inconsistent goals. Given a stochastic description of the signal and of the noise, an optimal interpolator (i.e., a minimum variance estimator) can be defined [A-2], but significant low-energy constituents of the signal may be severely distorted.

One of the most important applications of high-speed interpolation is satellite image registration, and some of the terminology of this paper is borrowed from the image processing literature. For a thorough discussion of the sampling/interpolation problem, see chapter 12 of [A-3].

---

*A near-bandlimited interpolator is an interpolator which is designed by approximating, in some sense, the spectral response of the ideal (infinite) sinc interpolator. Such interpolators usually differ in the sense in which they approximate a low-pass response, commonly in the spectral distribution of the error of approximation.

Many near-bandlimited interpolators possess a response more like that of a low-pass filter at almost every portion of the spectrum than that of polynomial interpolators.

## A.2 NOTATION AND BACKGROUND

For notational convenience, the continuous data function $f(x)$ is assumed to be sampled at integer values $f(k)$, for $k=0,1,2,\ldots,N-1$. The estimate $\hat{f}(x)$ is to be obtained for the range

$$\frac{N}{2} - 1 < x \leq \frac{N}{2}$$

(it is assumed that other sample sets will be used for other ranges).

The estimator $\hat{f}(x)$ is usually definable as a convolution with a kernel $h(x)$:

$$\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(k)h(x-k)$$

Thus, for each $k$, the kernel $h(x)$ is sampled to provide an interpolation weight as a function of the displacement $d=x-k$ from the estimation point $x$. All of the methods to be considered here are definable in this way. The interpolation condition is that $h(k)=0$ for each integer $k$ except that $h(0)=1$. Kernels for actual implementation will, of course, vanish outside some finite interval.

If $f(x)$ and $h(x)$ are Fourier integrable, then the Fourier transforms $\hat{F}$, $F$, and $H$ of $\hat{f}$, $f$, and $h$, respectively, are related by

$$\hat{F}(\omega)=H(\omega)\sum_{k=-\infty}^{\infty} F(\omega+k) \qquad \text{(normalized frequency)} \quad \text{(A.2.1)}$$

(see chapter 5 of [A-4]). The well-known special case is

$$h(x)=\text{sinc}(x)=\frac{\sin(\pi x)}{\pi x}$$

$$H(\omega)=\begin{cases}1 & \text{if } |\omega| \leq 0.5 \\ 0 & \text{otherwise}\end{cases} \qquad \text{(normalized frequency)}$$

which yields ideal low-pass bandlimited interpolation when an infinite number of samples of $f$ are available. The function $h$ could also be chosen to provide bandpass bandlimited interpolation (see [A-3], page 191), but the low-pass case is always assumed here.

In image processing applications, one usually requires that

$$\sum_{k=-\infty}^{\infty} h(x-k) = 1 \text{ for each } x \qquad \text{(A.2.2)}$$

(i.e., constants are reconstructed exactly), so that the mean intensity

of an image is preserved. Unless otherwise stated, this condition will be assumed.

Symmetry of the kernel is also assumed, except when an odd number of sample weights are required. For example, if one uses nearest neighbor interpolation, the kernel is <u>almost</u> completely defined by

$$h(x) = \begin{cases} 1 & \text{if } |x|<0.5 \\ 0 & \text{if } |x|>0.5 \end{cases}$$

but exact symmetry would require that $h(0.5)=h(-0.5)$, which would lead to the use of two sample values when these are equally displaced from the estimation point. The "tie" may be resolved, for example, by defining $h(0.5)=1$ and $h(-0.5)=0$. In this way, one uses a fixed number of non-zero weights (except when the estimation point coincides with a sample point).

Some properties of the sinc function are listed here for convenient reference:

$\text{sinc}(0)=1$

$\text{sinc}(k)=0$ for $k\neq 0$

$\sum_{k=-\infty}^{\infty} \text{sinc}(x-k)=1$ for each $x$

$\text{sinc}(x)=\text{sinc}(-x)$

$\text{sinc}(x+k)=(-1)^k\left(\frac{x}{x+k}\right)\text{sinc}(x)$ (a "recursion relation")

$\int_{-\infty}^{\infty} \text{sinc}(x)dx = 1$

if $a_n = \int_n^{n+1} \text{sinc}(x)dx$ for $n>0$, then the sequence $a_n$ is alternating, and $|a_n|$ is monotone decreasing

From these properties, it follows that, for each n, there is a number $n< \alpha <n+1$, such that $\int_{\alpha}^{\alpha} \text{sinc}(x)dx=1$.

The significance of the last property is indicated in the following result, which follows directly from equation (A.2.1);

if $f(x)$ is constant and $\hat{f}(x)$ is continuous and

$$\int_{-\infty}^{\infty} h(x)dx = H(o) = 1$$

then $\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(x)\, h(x-k) = f(x) \sum_{k=-\infty}^{\infty} h(x-k) = f(x)$

(that is, the exact constant reconstruction condition is satisfied).

## A.3 APPROACHES TO INTERPOLATION

### A.3.1 Introduction

Fourier analysis and the simple forms of the sinc interpolator tend to lead one toward approximations of ideal bandlimited interpolation, but there are several points which should be made.

From the viewpoint of bandlimited interpolation, the continuous data are contained in the infinite set of sample impulses, but they are corrupted by high-frequency sampling artifacts. Properly-sampled data are then completely separable from the sampling artifacts by ideal low-pass filtering. Thus,

$$f(x) = \hat{f}(x) = \sum_{k=-\infty}^{\infty} f(k)\ \text{sinc}\ (x-k)$$

The first problem is that one can neither obtain nor use an infinite number of samples of f(x), and a truncated function cannot be perfectly bandlimited.

Moreover, the continuous data may not really be bandlimited. For example, if the data have a constant slope (at least, locally), an ideal bandlimited interpolator would reconstruct the data using only low frequencies (i.e., the data's high frequency content would be aliased into low frequencies).

Also, the frequency response of ideal bandlimited interpolation may be inappropriate because the contribution of noise sources may be more significant at one portion of the passband. Simon [A-2] has found that near-bandlimited interpolators are often inferior to near-linear interpolators from the noise standpoint.

The following simple example may be useful: Suppose one uses only two samples, f(0) and f(1). How does one best reconstruct f(x) between 0 and 1? Intuitively, it seems unlikely that two samples provide much information about curvature, thus, a reasonable choice is the linear interpolator, which may be implemented by convolution with the function h(x), where

$$h(x) = \begin{cases} 1-|x| & \text{for } |x| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

The complete piecewise-linear reconstruction of f(x) is defined by

$$\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(k)h(x-k) = \sum_{x-1<k<x+1} f(k)(1-|x-k|)$$

In the (normalized) frequency domain, one has

$$\hat{F}(\omega) = H(\omega) \sum_{m=-\infty}^{\infty} F(\omega+m) = \text{sinc}^2(\omega) \sum_{m=-\infty}^{\infty} F(\omega+m)$$

From this equation, it is seen that there will be aliasing of high frequency content whenever $F(\omega) \neq 0$ for $|\omega| > 0.5$. If, for example, $f(x)$ has exactly constant non-zero slope (locally), it is not bandlimited and there will be some of this aliasing. However, the high frequency content of $h(x)$ causes aliasing of low frequencies of $f(x)$ to high frequencies, too. Wherever $f(x)$ is truly linear, it is interpolated exactly, despite the aliasing, whereas near-bandlimited interpolation would be incorrect.

A thorough comparison of near-bandlimited digital interpolators and a specific alternative, Lagrange polynomial interpolation, is contained in [A-1].

Practical approaches to interpolation can be divided into those which are tied to the concept of bandlimiting and those which are not. Perhaps the most familiar representatives of the two classes are trigonometric polynomial interpolation and Lagrange polynomial interpolation. Near-bandlimited approaches are considered first.

### A.3.2 Trigonometric Polynomial Interpolation

A real periodic function of period N which is also bandlimited, has a finite Fourier series. The finite Fourier series expansion can be obtained by means of a kernel:

$$h_N(x) = \begin{cases} 1+2\sum_{1\leq k<\frac{N}{2}} \text{Cos}\left(\frac{2\pi k}{N} x\right) & (N \text{ odd}) \\ 1+2\sum_{1\leq k<\frac{N}{2}} \text{Cos}\left(\frac{2\pi k}{N} x\right) + \text{Cos}(\pi x) & (N \text{ even}) \end{cases}$$

The special form of $h_N(x)$ for N even results from bandlimiting exactly at one of the Fourier frequencies (see [A-4] for details). This method is always based upon N samples, unless x is itself a sample point.

### A.3.3 Periodic Polynomial Spline Interpolation

This method fits smooth, periodic, piecewise-polynomial functions to the data samples [A-5]. Smoothness tends to produce bandlimiting, as the following development shows.

In view of equation(2.1), each non-negative spectral model for the data defines a unique "exact interpolation response", which will not, in general, correspond to a finite impulse response interpolation.

The spectral model of interest here is

$$F_m(\omega) = \text{sinc}^{2m}(\omega)$$

for an integer m. The optimal kernel is then defined by the aliasing ratio

$$H_m(\omega) = \frac{\text{sinc}^{2m}(\omega)}{\sum_{m=-\infty}^{\infty} \text{sinc}^{2m}(\omega+m)}$$

$$= \begin{cases} 1 & \text{if } \omega = 0 \\ 0 & \text{if } \omega \text{ is an integer} \neq 0 \\ \dfrac{1}{\sum_{m=-\infty}^{\infty} \left(\frac{\omega}{\omega+m}\right)^{2m}} & \text{otherwise} \end{cases}$$

as m increases, $\text{sinc}^{2m}(\omega)$ becomes more concentrated in the lowpass region and is aliased less. Thus, $H_m(\omega)$ approaches the bandlimited ideal as m increases.

Now, it will be noted that, for any positive integers k and N, the expression $H_m\left(\frac{k}{N}\right)$ is precisely the k'th attenuation factor corresponding to N - point periodic 2m-1 degree polynomial spline interpolation (see[A-5]). Thus, at the values $\frac{k}{N}$, the frequency response of periodic polynomial spline reconstruction interpolates the near-ideal $H_m(\omega)$. The case m=2, N=4 is one of the forms of the TRW cubic convolution interpolation (see section A.4.9 and [1-1]).

### A.3.4 Infinite Sample Set Estimation

The sinc function is an ideal interpolator for an infinite set of sample impulses of a properly bandlimited function. Having only a finite set of samples, the "missing samples" must be estimated.

If each estimate is an actual sample value, the mean of the samples, or the mean of any subset of the samples, then constants will be reconstructed exactly (see section A.2). This approach can be used to define trigonometric polynomial interpolation (see section A.3.2), or other, less familiar, techniques.

Example 3.4.1 Trigonometric Polynomial Interpolation

Repeating N samples periodically, one estimates $\hat{f}(k)=f(k \bmod N)$. This leads to the estimator

$$\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(k \bmod N)\ \mathrm{sinc}\ (x-k)$$

$$= \sum_{k=0}^{N-1} f(k) \sum_{j=-\infty}^{\infty} \mathrm{sinc}(x-k+Nj)$$

$$= \sum_{k=0}^{N-1} f(k) \sum_{j=-\infty}^{\infty} \mathrm{sinc}(d+Nj)$$

where, for convenience, the estimation displacement is denoted by d. Because of the properties of the sinc function the latter sum can be rewritten as

$$\mathrm{sinc}(d) \sum_{j=-\infty}^{\infty} (-1)^{Nj} \frac{d}{d+Nj}$$

$$= \mathrm{sinc}(d) \sum_{j=-\infty}^{\infty} (-1)^{Nj} \frac{\frac{d}{N}}{\frac{d}{N}+j}$$

$$= \begin{cases} \frac{\pi d}{N} \cot\left(\frac{\pi d}{N}\right) \mathrm{sinc}(d) & \text{if N is even} \\ \frac{\mathrm{sinc}(d)}{\mathrm{sinc}\left(\frac{d}{N}\right)} & \text{if N is odd} \end{cases}$$

(see [A-5]). Trigonometric identities may then be used to establish the equivalence with trigonometric polynomial interpolation.

Example A.3.4.2 Mean - Sinc Reconstruction

Let $\hat{f}(k)=\overline{f} = \frac{1}{N} \sum_{k=0}^{N-1} f(k)$ for $k \geq N$ and for $k<0$,

and define

$$\hat{f}(x)= \sum_{k=0}^{N-1} f(k)\ \mathrm{sinc}(x-k) + \sum_{\substack{k\geq N \\ k<0}} \overline{f}\ \mathrm{sinc}(x-k)$$

The series converges because

$$\mathrm{sinc}(x-k) = (-1)^{-k}\left(\frac{x}{x-k}\right)\mathrm{sinc}(x)$$

Example 3.4.3 Step - Sinc Reconstruction

Let

$$\hat{f}(k) = \begin{cases} f(0) & \text{for } k<0 \\ f(k) & \text{for } 0 \leq k \leq N-1 \\ f(N-1) & \text{for } k \geq N \end{cases}$$

This results in the estimator

$$\hat{f}(x) = f(0) \sum_{k=0}^{\infty} \mathrm{sinc}(x+k) + \sum_{k=1}^{N-2} f(k)\ \mathrm{sinc}\ (x-k) + f(N-1) \sum_{k=N-1}^{\infty} \mathrm{sinc}(x-k)$$

A.3.5 Truncated Sinc Interpolation

If one truncates the sinc function, it no longer defines ideal bandlimited interpolation, but it will reconstruct constants well, in the mean, if it is truncated at a point $\alpha$ for which

$$\int_{-\alpha}^{\alpha} \mathrm{sinc}(x)\ dx=1$$

(see section A.2). It will, in general, have a better overall frequency response if it is truncated at an integer value (the truncated sinc is then continuous).

In either case, division permits a relatively efficient computation of the set of interpolation weights, using the recursion relation

$$\text{sinc}\,(x-k) = (-1)^{-k} \frac{x}{x-k} \text{sinc}(x)$$

for $x \neq k$

A.3.6 Windowed Sinc Reconstruction

It will be seen (section A.4.1) that simple truncation of the sinc function can introduce substantial ripple into the frequency response, but that the sharp-cutoff characteristic is well-preserved. Windowing is a standard technique for exchanging increased transition bandwidth for decreased ripple, and it can be used for the truncated sinc interpolator, that is, a kernel may be defined as

$$h(x) = w(x)\ \text{sinc}(x)$$

where $w(x)$ is one of the popular analog windows.

Simple truncation corresponds to the use of a rectangular window, and it is optimal in the mean square sense, because of its sharp cutoff (see [A-4]). This property is especially useful if the data spectrum is essentially flat over the passband.

For many data types, including much satellite imagery, the spectral density drops off rapidly with frequency, so that frequency response errors near the edge of the passband are not as significant (in the mean) as low frequency errors. One of the popular analog windows might be chosen on the basis of a spectral model of the data[A-6] (some interpolators are compared for a Gaussian spectral model in section A.5.2 and in [A-2] ). The Kaiser window family is especially convenient, since window characteristics can be varied with a single parameter (see section A.4.12).

The kernels defined in this way will not, in general, satisfy the constant reconstruction condition (Eq A2.2), but there is a simple remedy: interpolate only the deviations from the mean. If $h(x) = w(x)\ \text{sinc}(x)$ and $w(x)$ truncates the sinc function at $\pm N/2$ where N is the fixed sample size, then the kernel $h_1(x)$, where

$$h_1(x) = \begin{cases} h(x) + \frac{1}{N}\left[1 - \sum_{k=-\infty}^{\infty} h(x-k)\right] & \text{if } |x| < \frac{N}{2} \\ 0 & \text{otherwise} \end{cases}$$

implements this procedure. For many common windows, this results in only a slight modification in the overall frequency response.

A.3.7 Lagrange Polynomial Interpolation

This well-known technique fits an N-1 degree polynomial to N samples, using polynomials which vanish on all but one of the sample points. This form of interpolation is often not appropriate for functional approximations because the behavior of the approximated function often differs considerably from that of a low degree polynomial, even if the function has a Taylor's series expansion.

Linear interpolation is the most familiar example. Step function or nearest-neighbor interpolation can be regarded as another example.

A.3.8 Non-Periodic Polynomial Spline

A smooth, piecewise-polynomial reconstruction can be defined with various non-periodic boundary conditions. A common one in the case of the cubic spline is the assumption of zero curvature at the boundary. The resulting (finite) reconstruction has the following property:

Among those interpolating functions which have a continuous second derivative on the interval of definition, this cubic spline "curves the least" in order to fit the data (for an exact statement of this property, see [ A-7 ], page 207).

High curvature is, of course, associated with high frequencies, and this property suggests a tendency toward bandlimiting, which can be seen in the frequency response (section 4.6).

A.3.9 Polynomial Osculatory Interpolation

One alternative to Lagrange interpolation which is sometimes useful in approximating relatively smooth functions is Hermite, or osculatory, approximation. For this method, a continuous function and its derivative(s) are interpolated at several points. Since one does not have direct access to the data in interpolation, one can only estimate the derivatives from the samples, using finite differences.

A.3.10 Discrete Orthogonal System Interpolation

Suppose that a set $\psi_k$ of real, continuous functions, k=0, 1, . . . . ., N-1 defines a discrete orthogonal system over a finite set of sample points, say, at integer values, j=0, 1, . . . . . ., N-1, that is

$$\sum_{j=0}^{N-1} \psi_k(j) \quad \psi_\ell(j) = \begin{cases} 1 & \text{if } k=\ell \\ 0 & \text{if } k\neq\ell \end{cases}$$

then a function could be estimated from its sample values by

$$\hat{f}(x) = \sum_{k=0}^{N-1} \left( \sum_{j=0}^{N-1} f(j)\, \psi_k(j) \right) \psi_k(x)$$

(discrete orthogonality ensures that $\hat{f}(x)$ interpolates $f(x)$). If one of the $\psi_k$ is the constant function, then constants will be reconstructed exactly.

To determine the convolution kernel $h(x)$, rearrange the summations in the definition of $\hat{f}(x)$ to obtain

$$\hat{f}(x) = \sum_{j=0}^{N-1} f(j) \sum_{k=0}^{N-1} \psi_k(j)\psi_k(x)$$

Since this estimation is to be used only on the interval $\left(\frac{N}{2} - 1, \frac{N}{2}\right]$, the displacement $d$ of the estimation point $x$ from the sample point $j$ determines both $x$ and $j$. For example, suppose that $N$ is even and $M<d=x-j\leq M+1$ for some integer $M$. Then

$$M+j<x\leq M+j+1$$

and, hence, $M+j = \frac{N}{2} - 1$

and $x=d+j = d+\frac{N}{2} - 1-M$

Thus, it is possible to define

$$h(d) = h(x-j) = \sum_{k=0}^{N-1} \psi_k(j)\, \psi_k(x)$$

for $-\frac{N}{2} \leq d \leq \frac{N}{2}$

and $\frac{N}{2} - 1 \leq x \leq \frac{N}{2}$

with $h(d) = 0$ otherwise (it is possible that $h$ will have distinct analytic expressions on each unit-length interval).

The first example has been covered elsewhere (section A.3.7), but it provides an excellent example of the procedure.

Example 3.10.1 Quadratic Lagrange Interpolation

The Lagrange auxiliary functions are easily found:

$$\psi_0(x) = \frac{(x-1)(x-2)}{2}$$

$$\psi_1(x) = -x(x-2)$$

$$\psi_2(x) = \frac{x(x-1)}{2}$$

$\psi_0(0)=1$ $\psi_0(1)=0$ $\psi_0(2)=0$

$\psi_1(0)=0$ $\psi_1(1)=1$ $\psi_1(2)=0$

$\psi_2(0)=0$ $\psi_2(1)=0$ $\psi_2(2)=1$

It is clear that these $\psi_k(x)$ form a discrete orthogonal system.

A relatively simple expression for the kernel can be obtained. For $-0.5<d\leq 0.5$, $j=1$, and

$$h(d) = h(x-1) = \sum_{k=0}^{2} \psi_k(1)\,\psi_k(x)$$

$$= \psi_1(x) = 2x-x^2$$

$$= 2(1+d) - (1+d)^2$$

$$= 1-d^2$$

For $0.5<d\leq 1.5$, $j=0$, so that

$$h(d) = h(x-0) = \sum_{k=0}^{2} \psi_k(0)\,\psi_k(x)$$

$$= \frac{(x-1)\,(x-2)}{2}$$

$$= \frac{(d-1)\,(d-2)}{2}$$

For $-1.5<d\leq -0.5$, $j=2$, and

$$h(d) = h(x-2) = \psi_2(x)$$

$$= \frac{x(x-1)}{2} = \frac{(2+d)\,(2+d-1)}{2}$$

$$= \left(\frac{|d|-2)\,(|d|-1}{2}\right)$$

Thus, a compact expression for h(d) is

$$h(d) = \begin{cases} 0 & \text{for } d \leq -1.5 \\ \dfrac{(|d|-2)\,(|d|-1)}{2} & \text{for } -1.5<d\leq -0.5 \\ 1-d^2 & \text{for } -0.5<d\leq 0.5 \\ \dfrac{(|d|-2)\,(|d|-1)}{2} & \text{for } 0.5<d\leq 1.5 \\ 0 & \text{for } d>1.5 \end{cases}$$

Note the slight asymmetry in the definition resulting from the asymmetry of the estimation interval. It is absolutely essential because of the discontinuous kernel. The overall reconstruction is not continuous, in general.

Example 3.10.2 Discrete Cosine Transform Interpolation

The discrete cosine transform G of a function f(k) for $0 \leq k \leq N-1$, is defined by

$$G(k) = \begin{cases} \frac{\sqrt{2}}{N} \sum_{m=0}^{N-1} f(m) & \text{for } k=0 \\ \frac{2}{N} \sum_{m=0}^{N-1} f(m) \cos\left[\frac{(2m+1)\, k\pi}{2N}\right] & \text{for } 1 \leq k \leq N-1 \end{cases}$$

Thus, it is possible to interpolate a continuous function f(x) using

$$\hat{f}(x) = \frac{1}{\sqrt{2}} G(0) + \sum_{k=1}^{N-1} G(k) \cos\left[\frac{(2x+1)\, k\pi}{2N}\right]$$

Since expressions for the kernel are rather involved, they will not be listed here.

This is a non-trivial application of the discrete orthogonal system approach. Another familiar technique which could be defined in this way is trigonometric polynomial interpolation, by means of the orthogonal system of sinusoids of the discrete Fourier transform.

A.3.11 Attenuation Factor Kernel Definition

Suppose that one desires a periodic interpolant (i.e., the samples are repeated to simulate an infinite sample set) and one can specify the Fourier coefficients $P_m$ of the desired (periodic) impulse response. Then the exact Fourier coefficients $H_m$ of the desired kernel h(x) are

$$H_m = N\, P_m\, F(m \bmod N)$$

where $F_k$ denotes the k'th DFT Coefficient [A-5]. This corresponds to taking the N-periodic DFT of the finite data sequence and then "correcting" the infinite (but periodic) transform with the "attenuation factors" $P_m$.

For example, the attenuation factors

$$P_m = \begin{cases} 1 & \text{for } m = 0 \\ \text{sinc}^4\left(\frac{m}{N}\right) \left(\frac{3}{2+\cos\frac{2\pi m}{N}}\right) & \text{for } m \neq 0 \end{cases}$$

result in periodic cubic spline interpolation (see section A.3.3). The kernel is expressible as a cubic piecewise-polynomial (see section A.4.9). One of the forms of TRW's cubic convolution uses this kernel.

### A.3.12 Direct Kernel Construction

An interpolation kernel should have the properties listed in section A.2. Additional properties are sometimes chosen, such as exact constant or slope reconstruction and smoothness conditions (to provide a smooth reconstruction).

If one chooses a family of functions with the required number of degrees of freedom, one may be able to solve the resulting system of equations to define a kernel meeting all of the conditions.

Popular choices are cubic or quartic spline fits to the unit impulse with certain additional conditions [A-2]. The TRW cubic convolution methods are excellent examples of this procedure. They have been shown to perform well on real images and are relatively computer-efficient. The system response for these methods can be quite close to that of the trigonometric polynomial method, despite the computational simplicity of the kernel.

## A.4 SPECIFIC INTERPOLATORS

### A.4.1 Introduction

It has been shown that interpolation can be seen from many points of view. The purpose of this section is to compare the interpolators which result from the approaches described in the previous section. Performance comparisons are left for section A.5.

The forms of several 4-point interpolation kernels are shown in Figure A4-1. Note the great similarity of form which makes kernel comparisons very difficult. The concept of windows which was useful in the design of near-bandlimited interpolators (Section A.3.6), can also be used to facilitate the comparison of interpolators.

Every interpolator can be regarded as a windowed sinc function, although the window is, in some cases, discontinuous. The window serves as a "normalized interpolator", with the sinc function as the "normalizing factor." Kernels can then be discerned and characterized readily.

Interpolators can also be compared in the frequency domain (this is most appropriate for near-bandlimited interpolators, but can be used generally).

For each of the kernels considered, the explicit form of the kernel is provided, together with its "truncation window" or "normalized kernel" and its frequency response. Each of these is real and symmetric (except possibly at a finite number of points), so only the positive half of each domain is indicated.

### A.4.2 Nearest Neighbor (Figure A4-2)

The kernel of this most primitive method is

$$h(d) = \begin{cases} 1 & \text{for } -0.5 < d \leq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

This kernel reconstructs the data as a step function, and very serious aliasing problems arise because h(x) is far from bandlimited. In imagery, the moire' effects of spurious high spatial frequency components of the kernel may be detectable to the eye as "blockiness". The performance may be adequate for reconstructing highly-oversampled data.

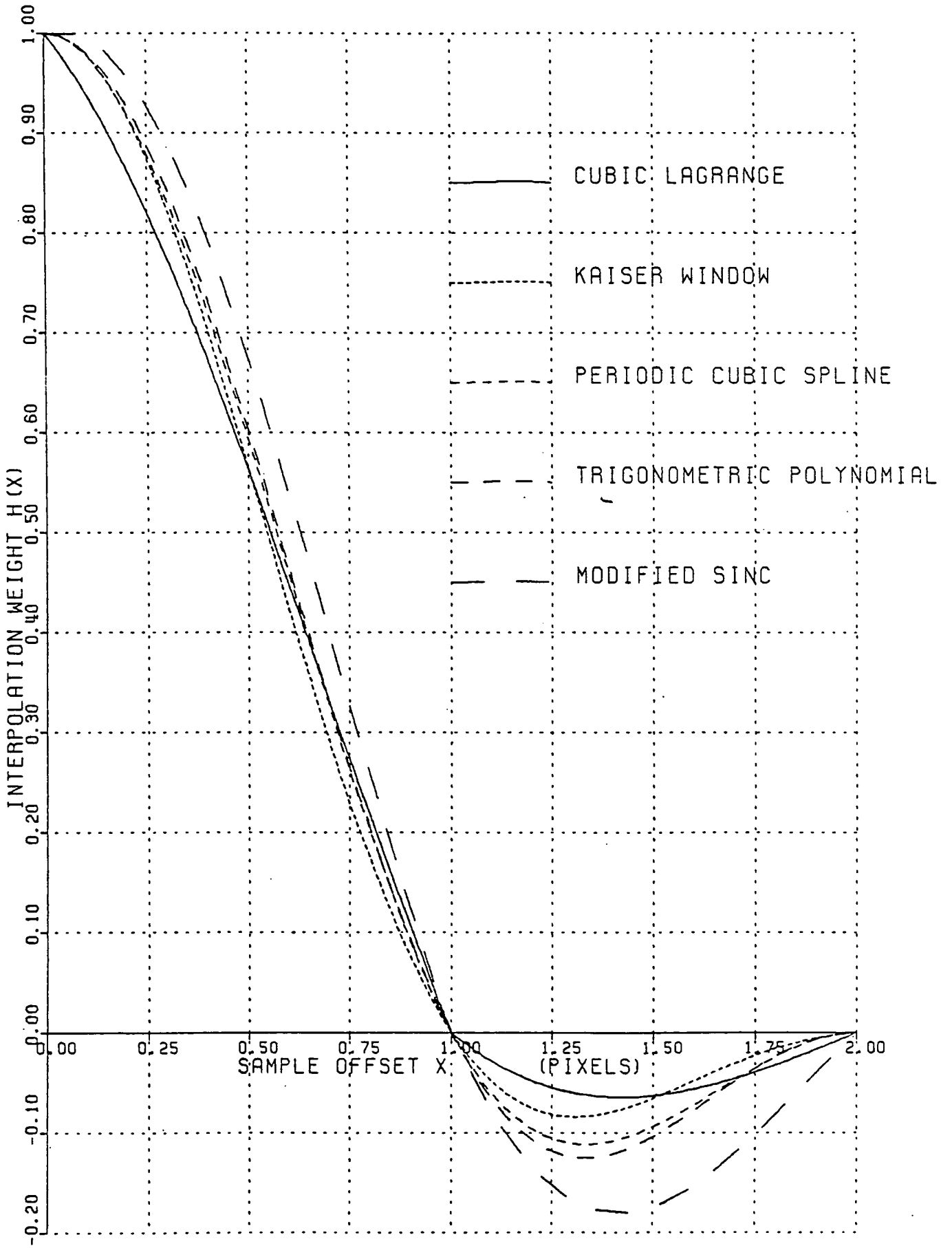



Figure A4-1. 4-POINT INTERPOLATION KERNELS

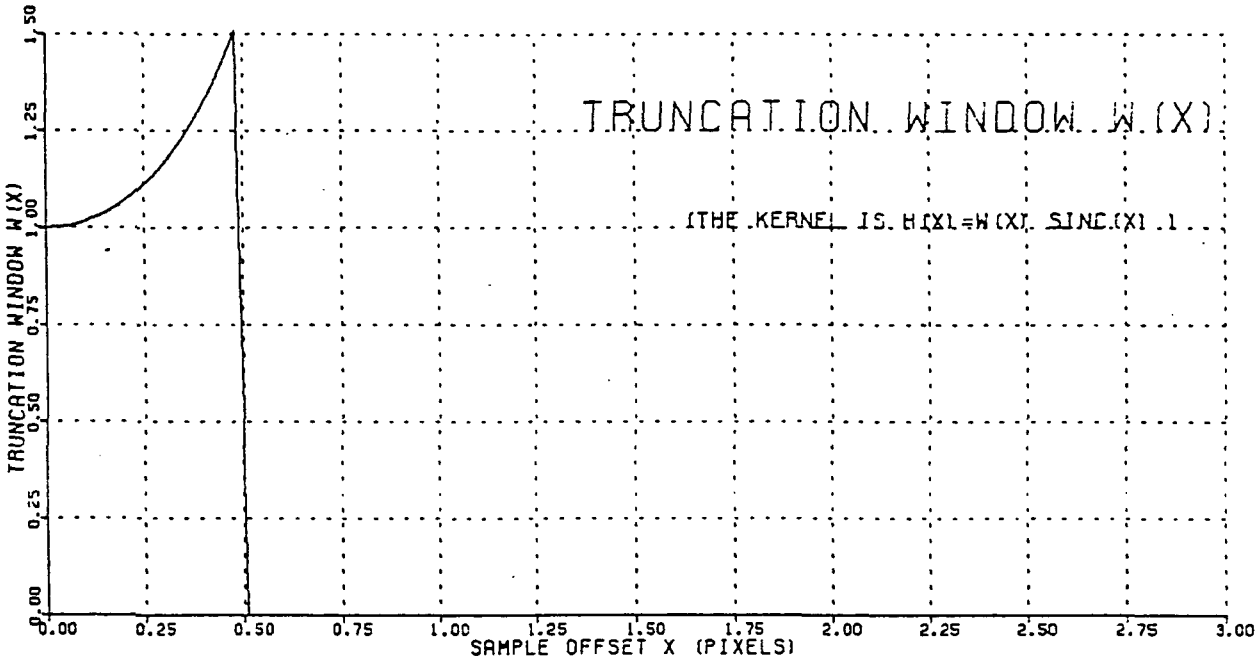
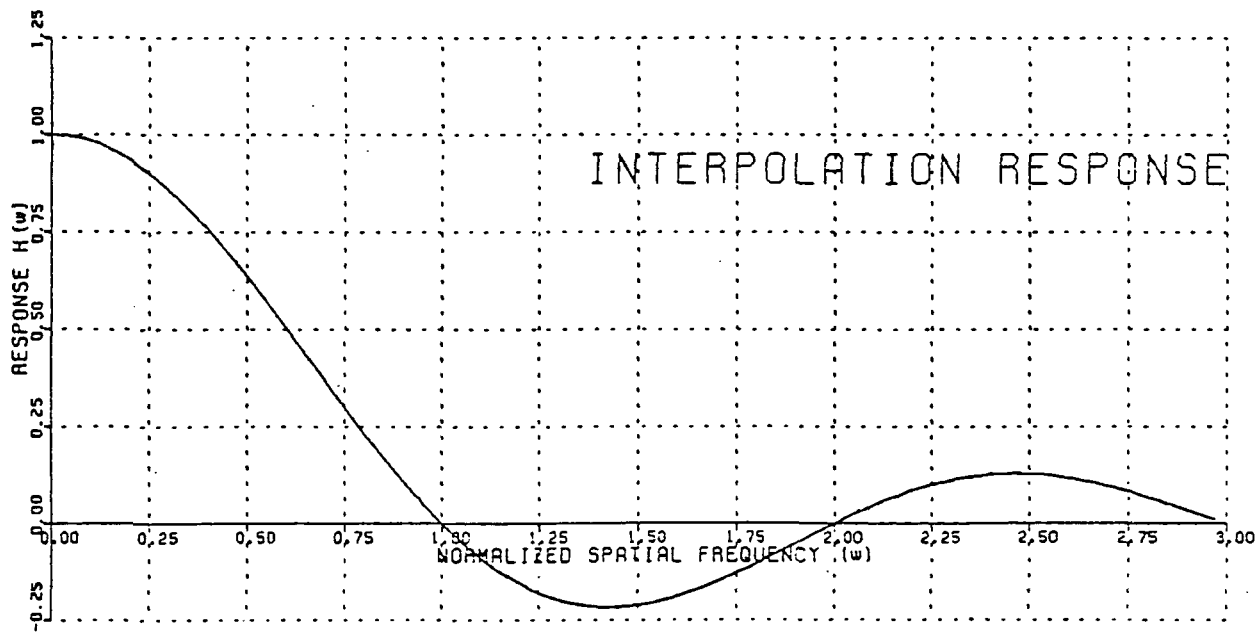



Figure A4-2. NEAREST NEIGHBOR (1 Point)



A.4.3 Linear (Figure A4-3)

Piecewise linear reconstruction may be defined by

$$h(d) = \begin{cases} 1 - |d| & \text{for } |d| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

This method cannot reconstruct curvature of the data function within a sampling interval, of course. In imagery, the resulting lack of resolution may be visible to the eye as "blurriness".

A.4.4 Quadratic Lagrange Interpolation (Figure A4-4)

It was shown in example 3.10.1 that this kernel is

$$h(d) = \begin{cases} 0 & \text{for } d \leq -1.5 \\ \frac{(|d|-1)(|d|-2)}{2} & \text{for } -1.5 < d \leq -0.5 \\ 1-|d|^2 & \text{for } -0.5 < d \leq 0.5 \\ \frac{(|d|-1)(|d|-2)}{2} & \text{for } 0.5 < d \leq 1.5 \\ 0 & \text{for } d > 1.5 \end{cases}$$

A.4.5 Cubic Lagrange (Figure A4-5)

The kernel is

$$h(d) = \begin{cases} \frac{(1-|d|^2)(2-|d|)}{2} & \text{for } |d| \leq 1 \\ \frac{(1-|d|)(2-|d|)(3-d)}{6} & \text{for } 1 < |d| \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

This may be derived as in example 3.10.1.

A.4.6 Cubic Spline (Figure A4-6)

The kernel is

$$h(d) = \begin{cases} (1-|d|)(1+0.8|d|-|d|^2) & \text{for } |d| \leq 1 \\ (1-|d|)(2-|d|)(-|\frac{d}{3}|+\frac{4}{5}) & \text{for } 1 < |d| \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

A cubic spline is formally fit to f(0), f(1), f(2) and f(3); these polynomial weights for $\hat{f}(x)$, $1 \leq x \leq 2$, can then be determined directly.

A.4.7 Cubic Osculatory (Figure A4-7)

Here, a cubic is used to fit data and estimates of the derivative.

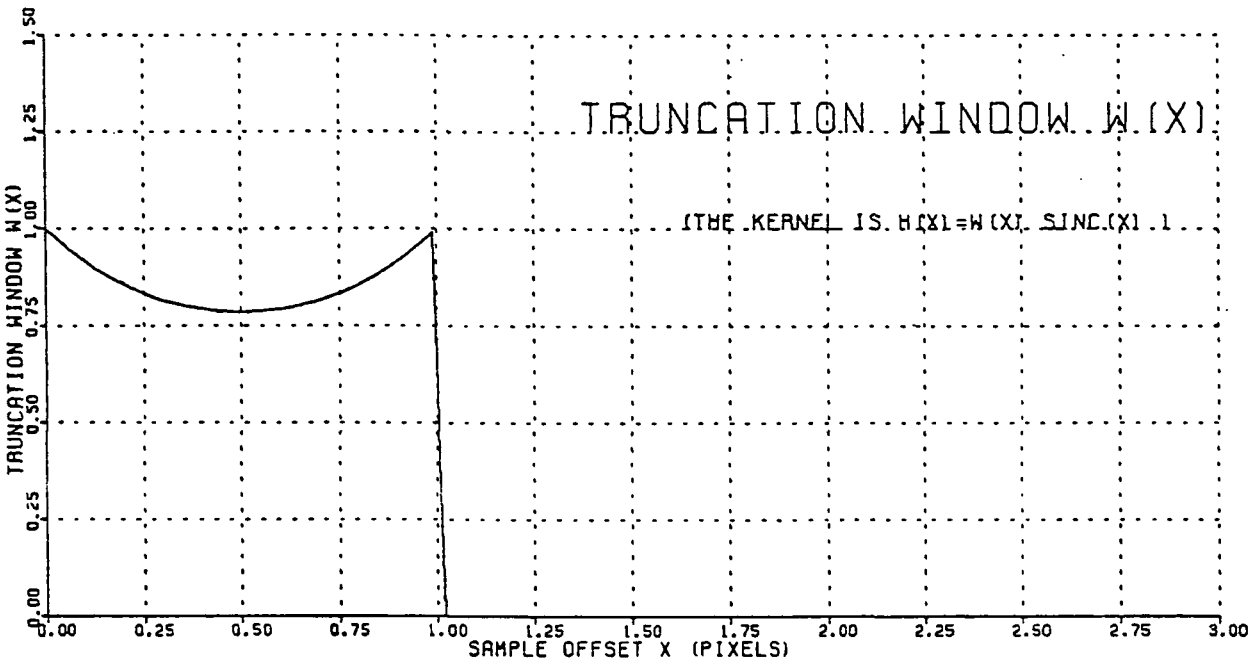
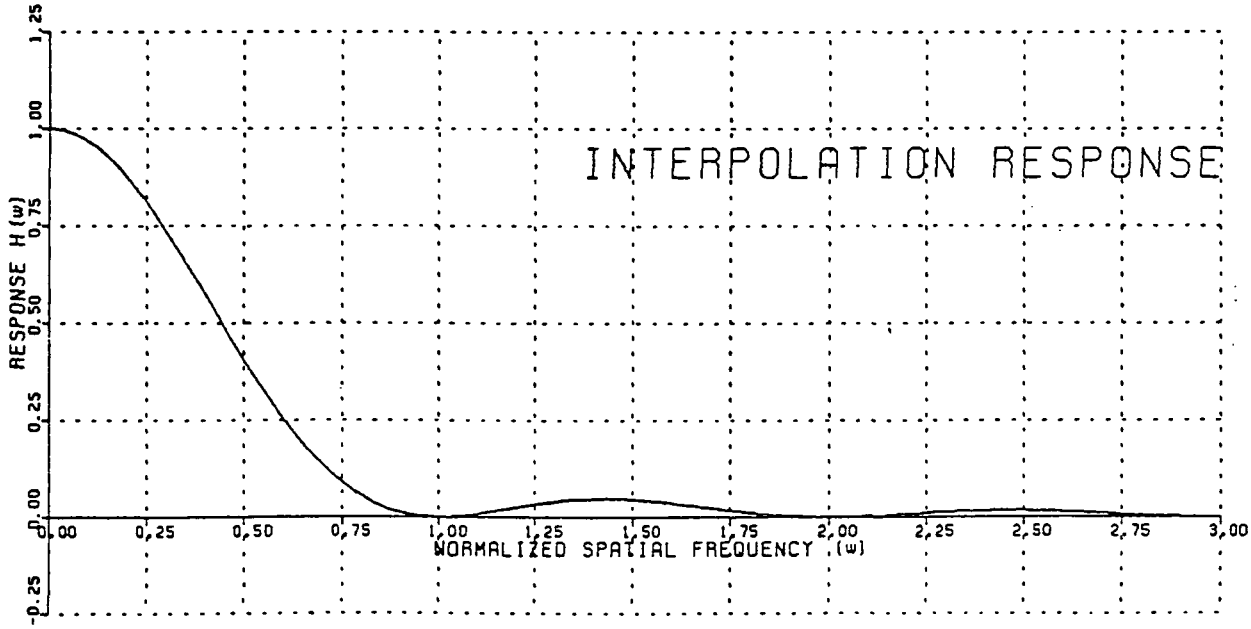



Figure A4-3. LINEAR (2 Points)



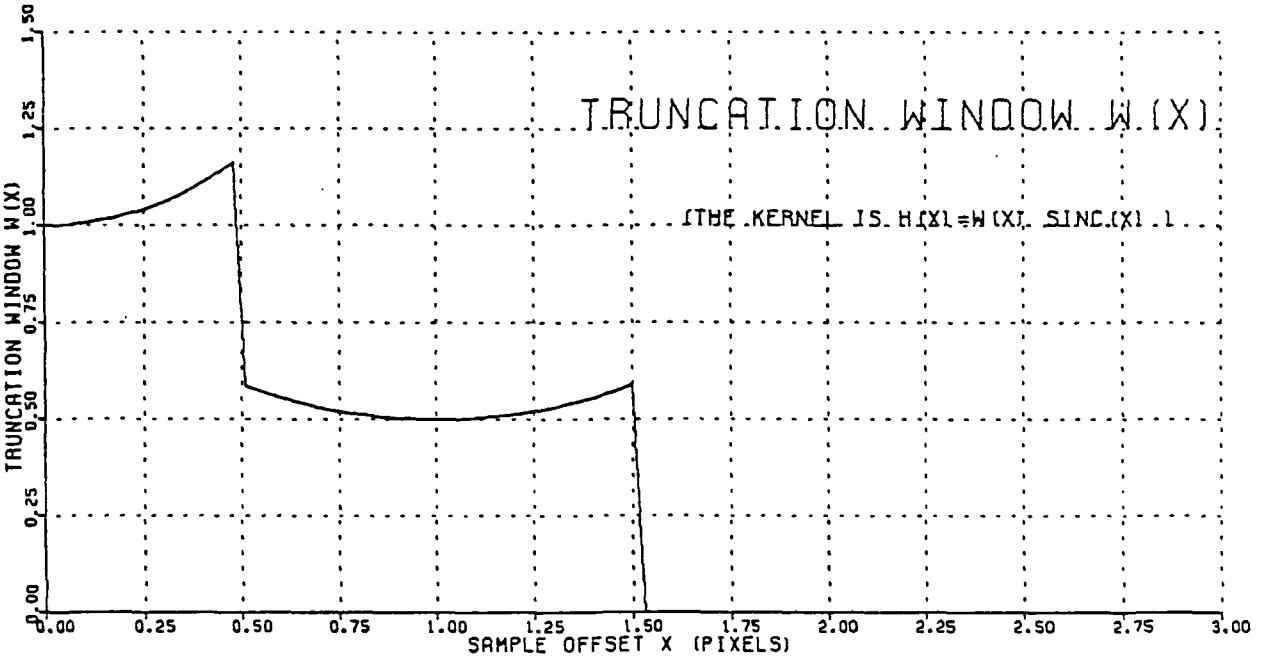
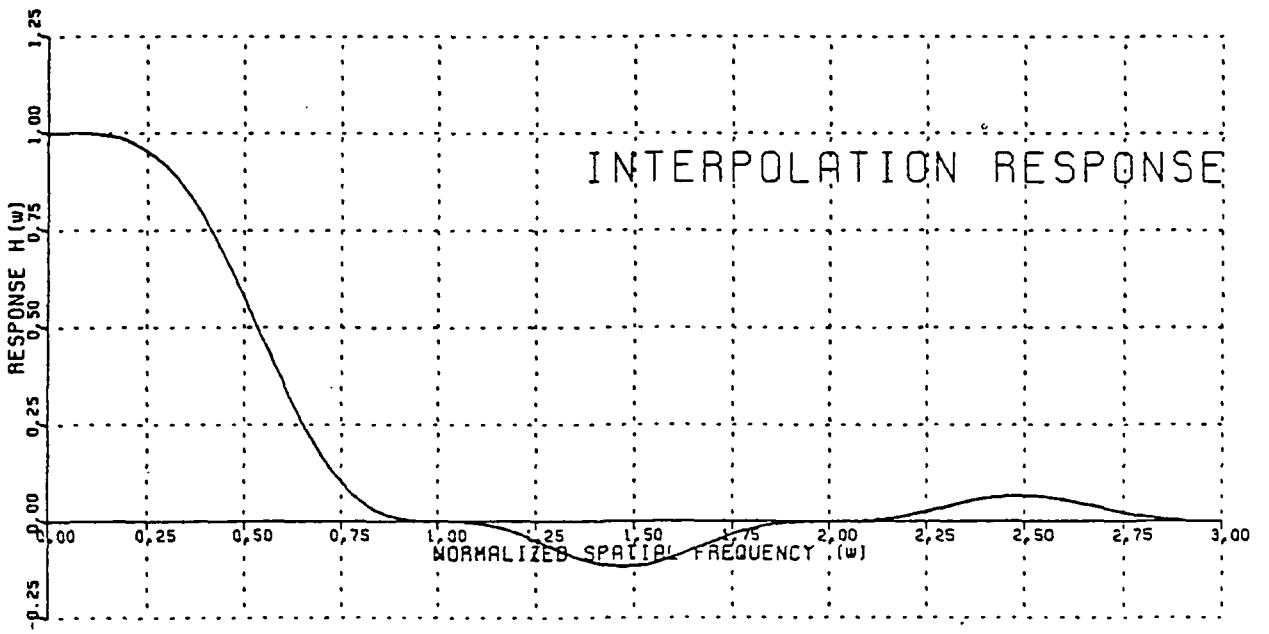



Figure A4-4. QUADRATIC (3 Points)



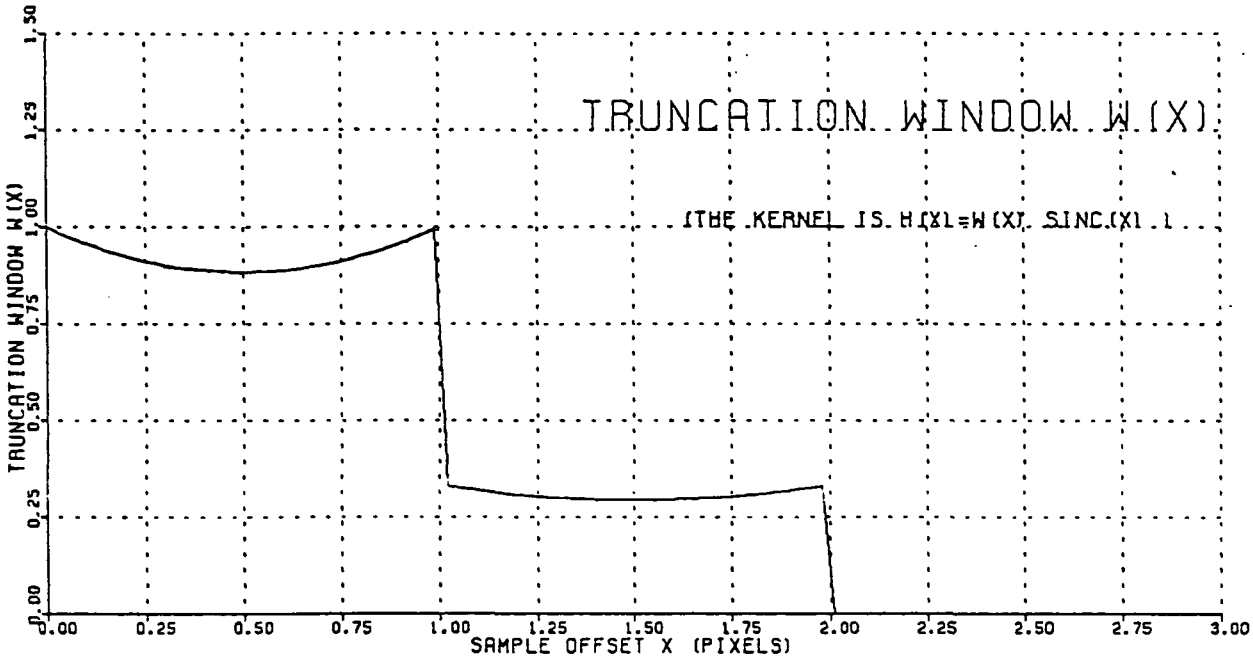
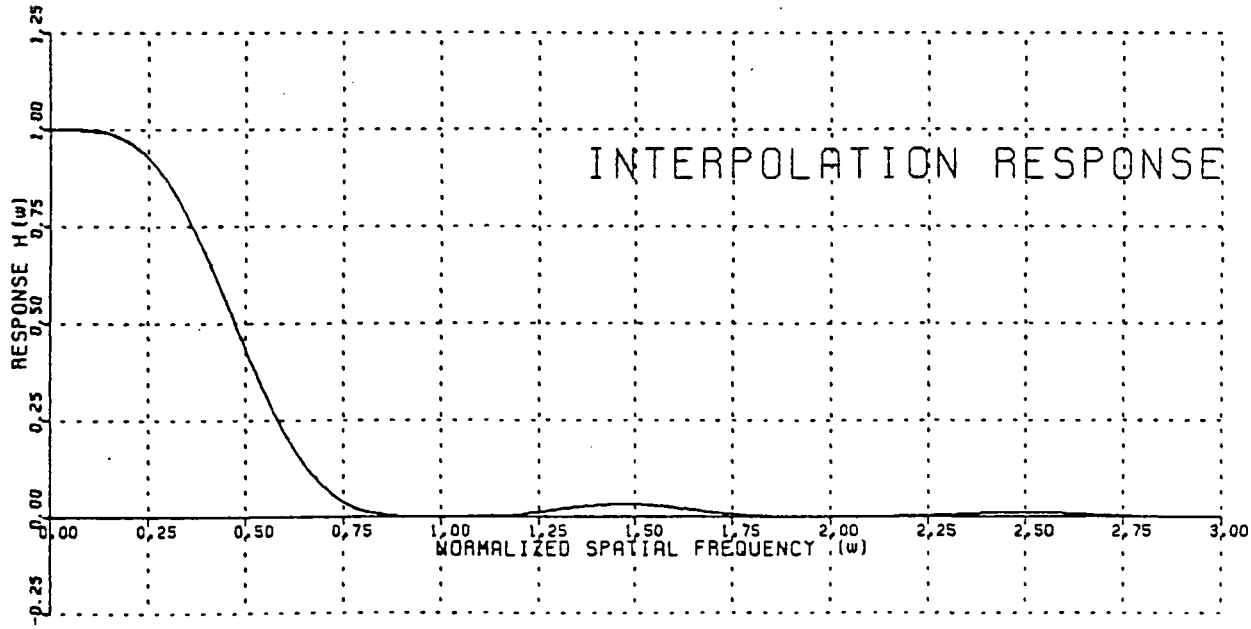



Figure A4-5. LAGRANGE (4 Points)



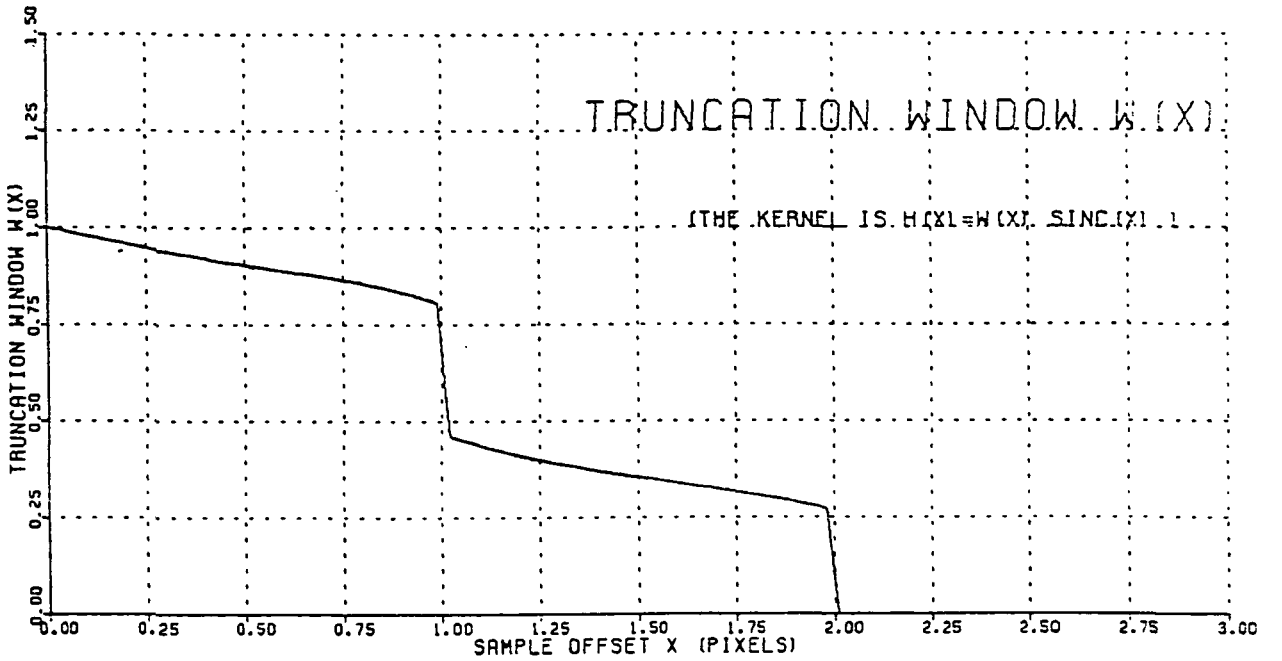
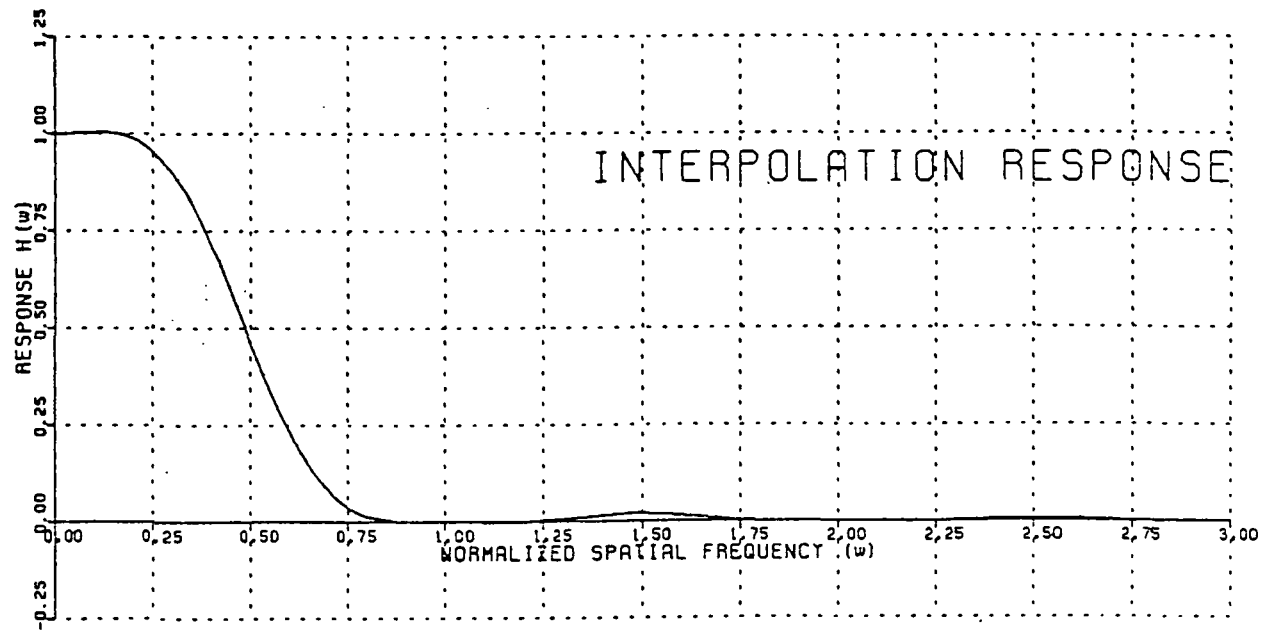



Figure A4-6. CUBIC SPLINE (4 Points)



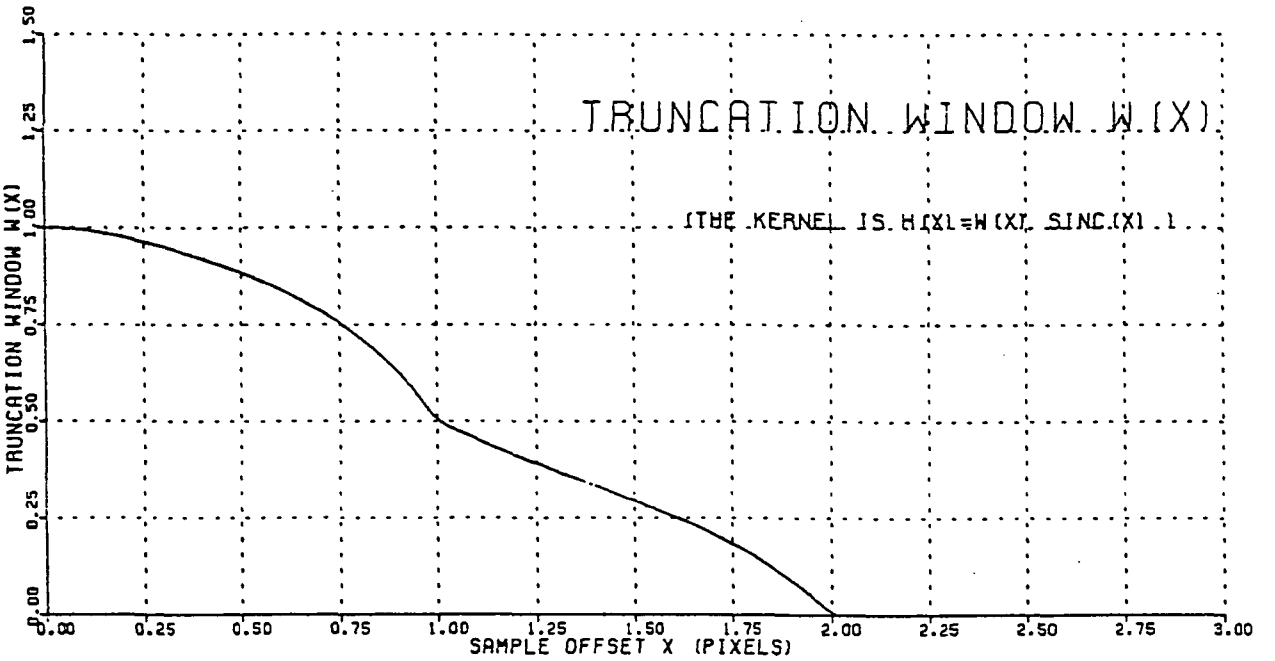
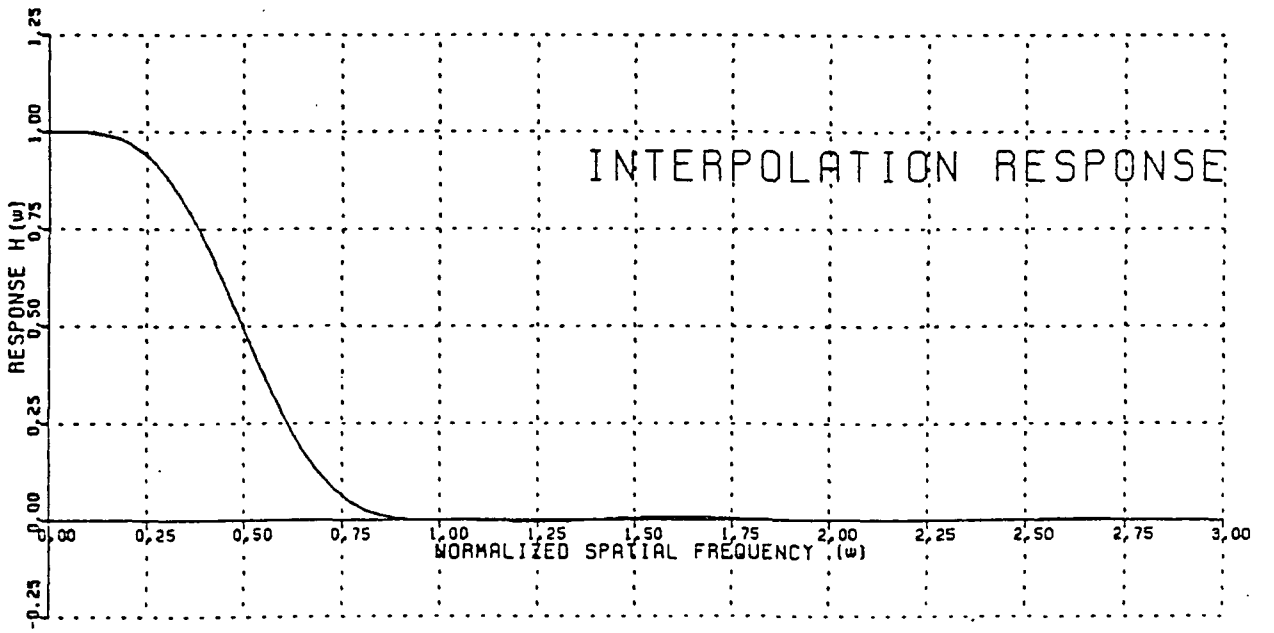



Figure A4-7. CUBIC OSCULATORY (4 Points)



If f(x) is sampled at k=0,1,2,3 then f'(1) is estimated as $\frac{1}{2}$ (f(2)-f(0)) and f'(2) is estimated as $\frac{1}{2}$ (f(3)-f(1)). The kernel which fits a cubic to f(1) and f(2) and the estimates of f'(1) and f'(2) is

$$h(d) = \begin{cases} \dfrac{2-5\,|d|^2+3\,|d|^3}{2} & \text{for } |d|\leq 1 \\ \dfrac{4-8\,|d|+5\,|d|^2-|d|^3}{2} & \text{for } 1<|d|\leq 2 \\ 0 & \text{otherwise} \end{cases}$$

This kernel is continuous and has a continuous derivative.

A.4.8 Trigonometric Polynomial (Figures A4-8a, A4-8b, A4-8c and A4-8d)

The kernel (for N=2,4 or 6) is

$$h(d) = \begin{cases} \dfrac{1}{N}\left[1+2\sum_{k=1}^{\frac{N}{2}-1}\text{Cos}\left(\dfrac{2\pi kd}{N}\right)+\text{Cos}\,(\pi d)\right] & \text{for } |d|\leq \dfrac{N}{2} \\ 0 & \text{otherwise} \end{cases}$$

For N=3, the kernel is

$$h(d) = \begin{cases} \dfrac{1}{3}\left(1+2\,\text{Cos}\left(\dfrac{2\pi d}{3}\right)\right) & \text{for } -1.5<d\leq 1.5 \\ 0 & \text{otherwise} \end{cases}$$

Note that, for the first time, the frequency response exceeds 1 for part of the passband (N≥3).

A.4.9 Periodic Cubic Spline (Figures A4-9a and A4-9b)

If two sample points are used, the kernel is

$$h(d) = \begin{cases} 1-3\,|d|^2+2\,|d|^3 & \text{for } |d|\leq 1 \\ 0 & \text{otherwise} \end{cases}$$

Note that this interpolator, whose response appears quite good, estimates inter-sample curvature (from the two samples).

The 4-point form is a version of TRW cubic convolution [1-1]. The kernel is

$$h(d) = \begin{cases} 1-2.25|d|^2+1.25\,|d|^3 & \text{for } |d|\leq 1 \\ 3-6\,|d|+3.75|d|^2-0.75\,|d|^3 & \text{for } 1<|d|\leq 2 \\ 0 & \text{otherwise} \end{cases}$$

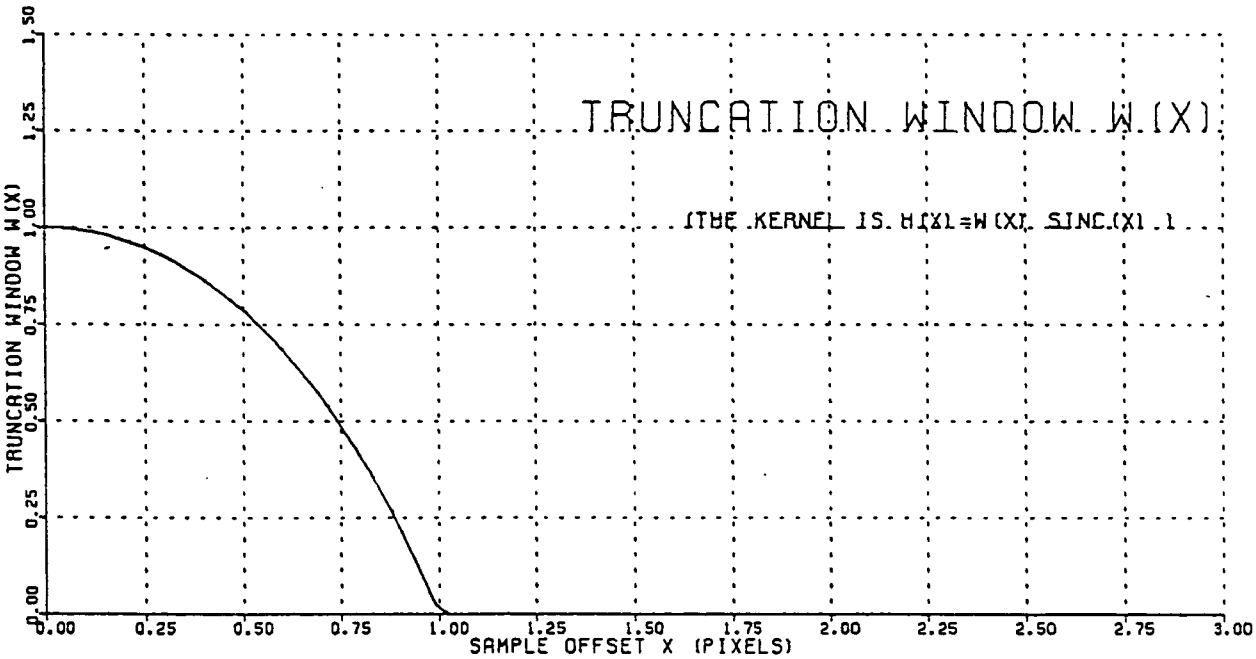
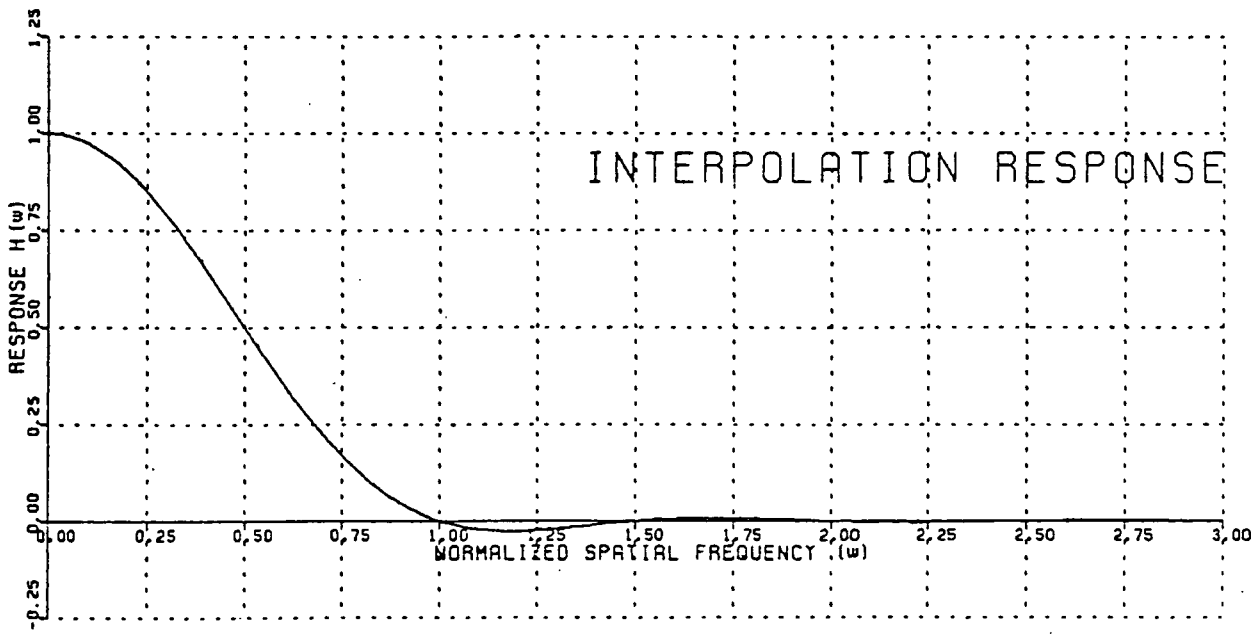



Figure A4-8a. TRIGONOMETRIC POLYNOMIAL (2 Points)



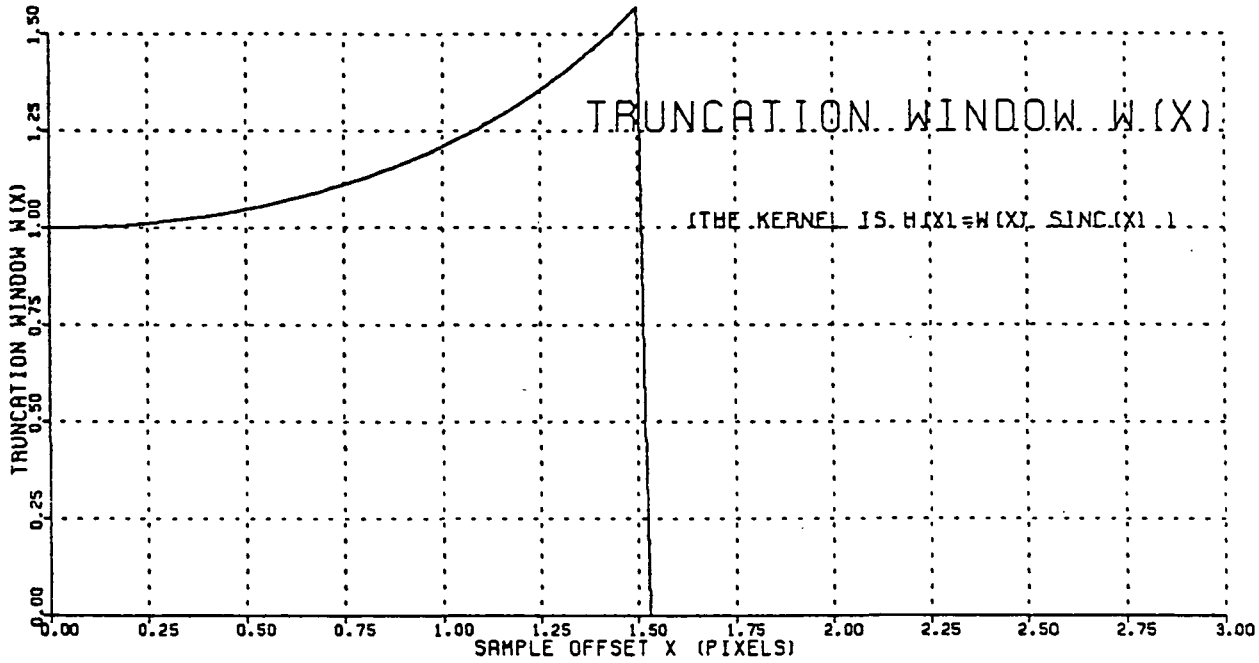
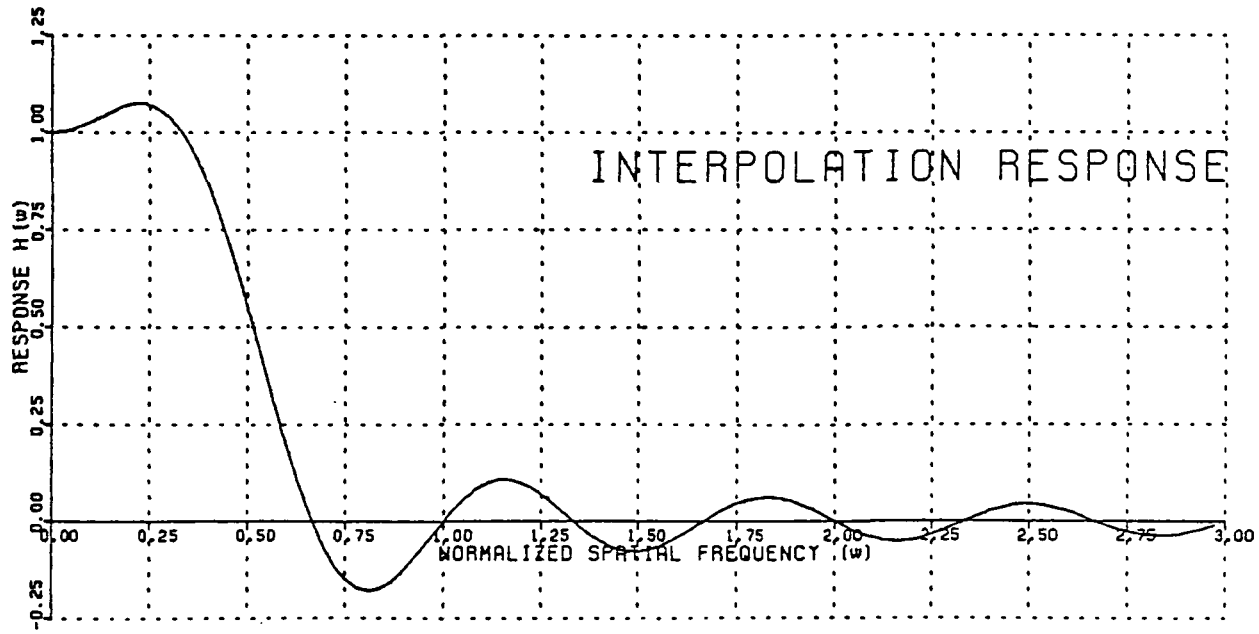



Figure A4-8b. TRIGONOMETRIC POLYNOMIAL (3 Points)



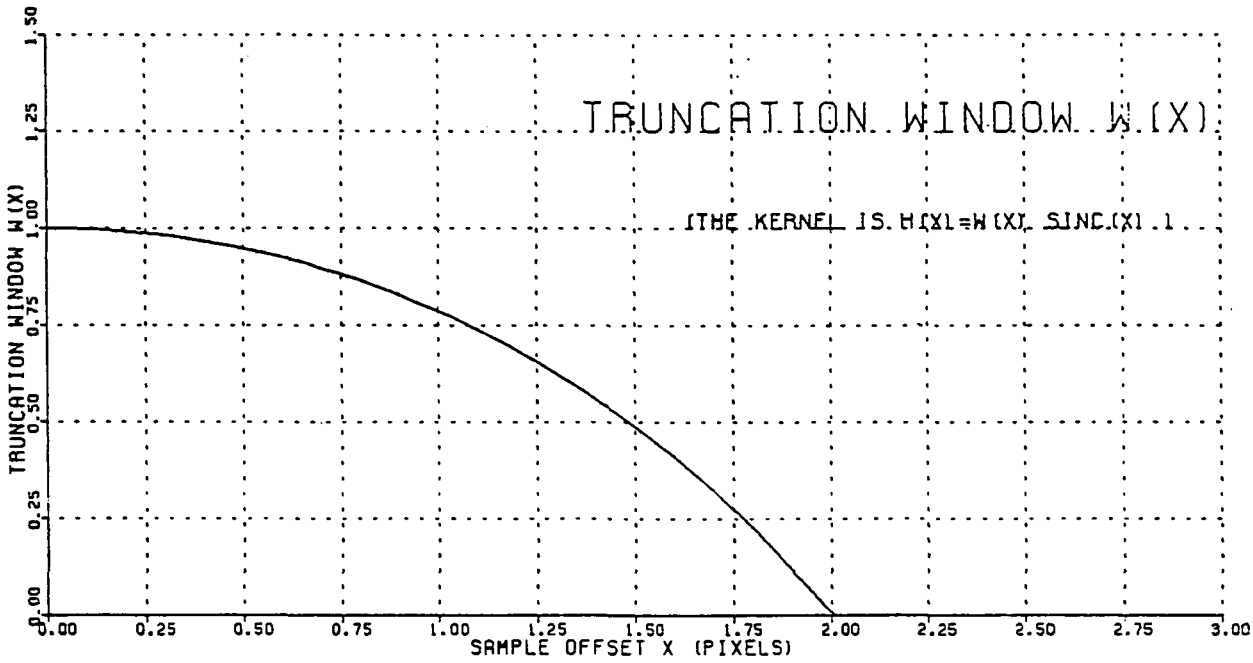
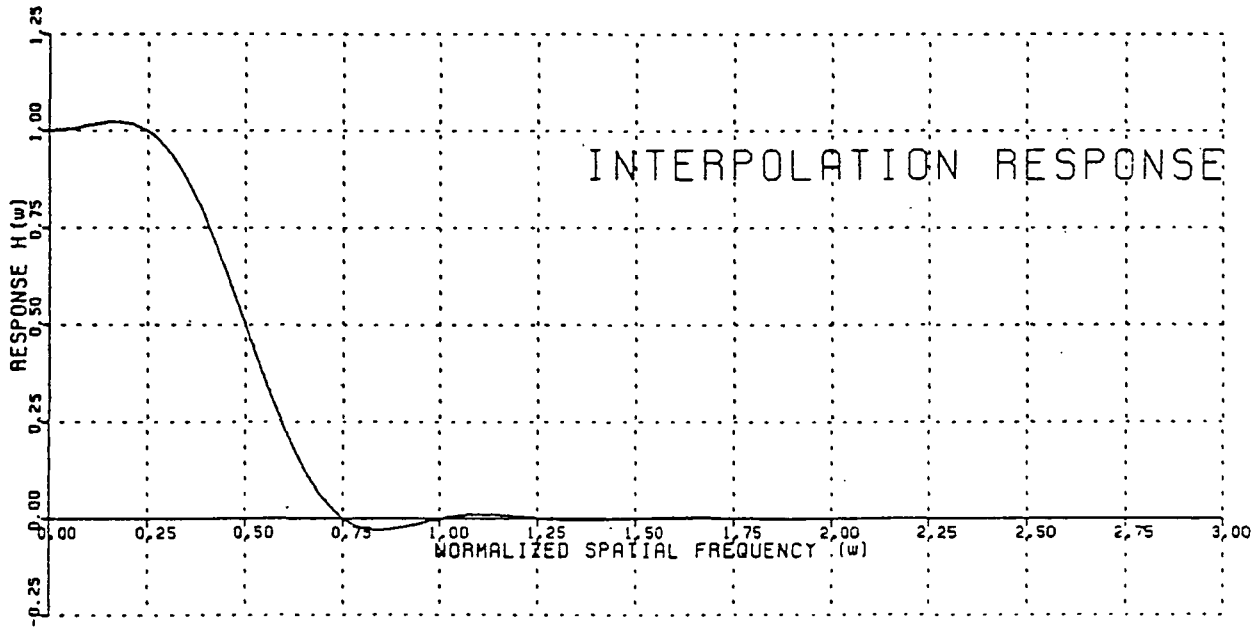



Figure A4-8c. TRIGONOMETRIC POLYNOMIAL (4 Points)



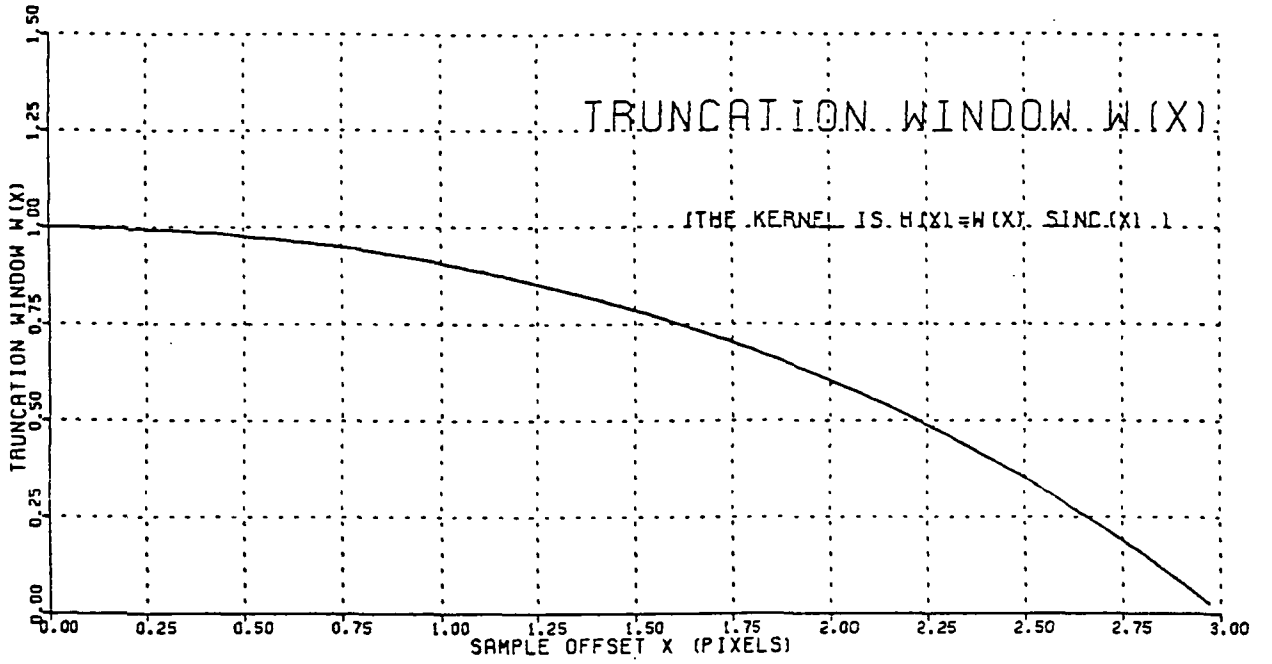
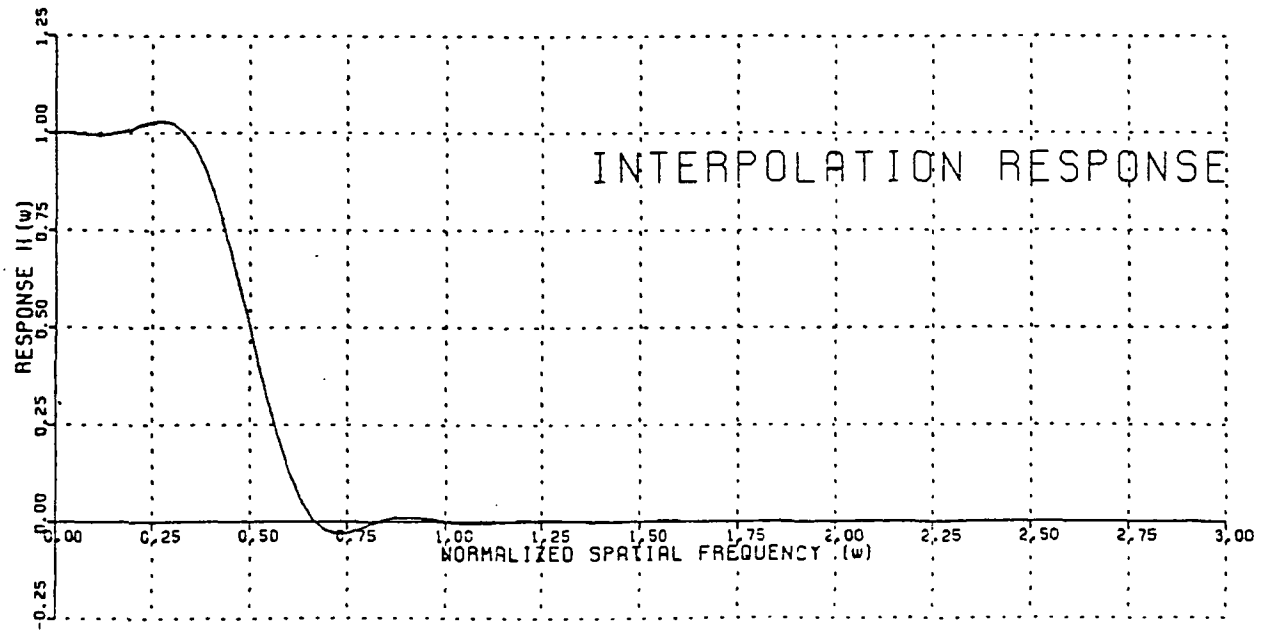



Figure A4-8d. TRIGONOMETRIC POLYNOMIAL (6 Points)



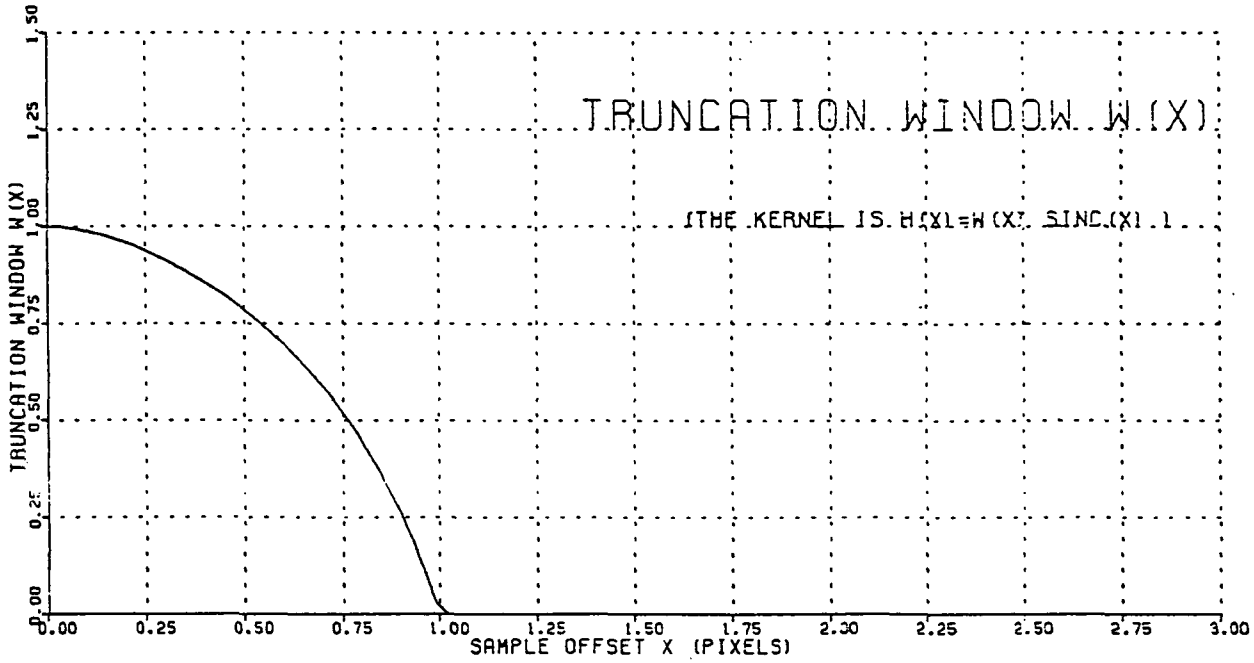
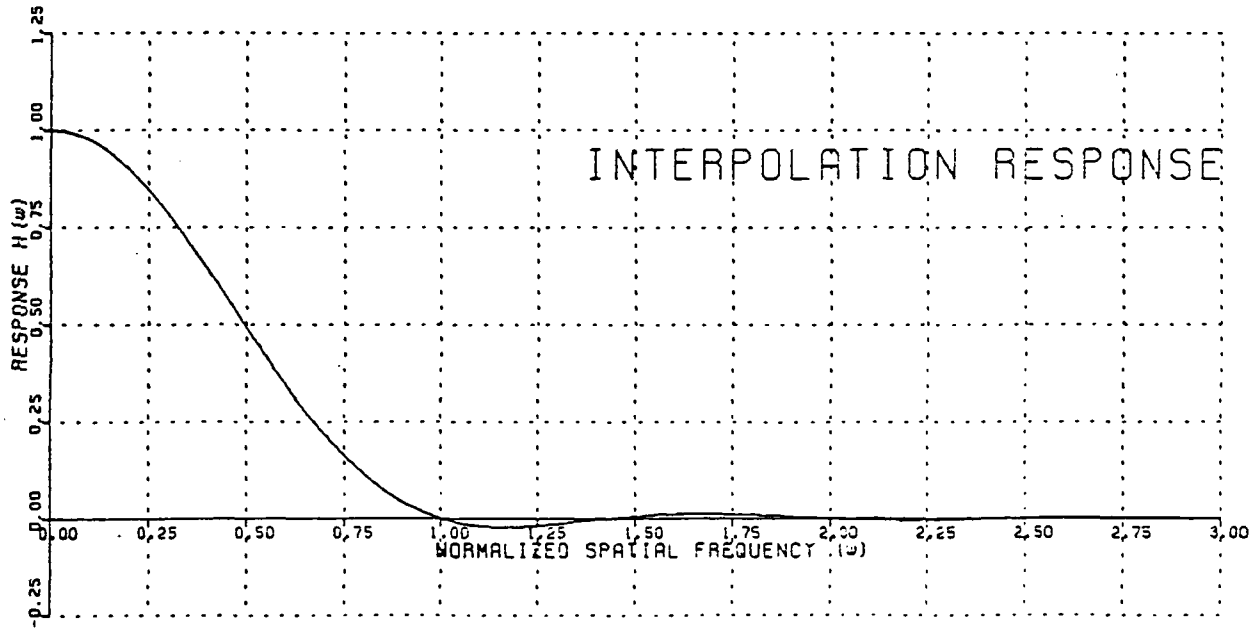



Figure A4-9a. PERIODIC CUBIC SPLINE (2 Points)



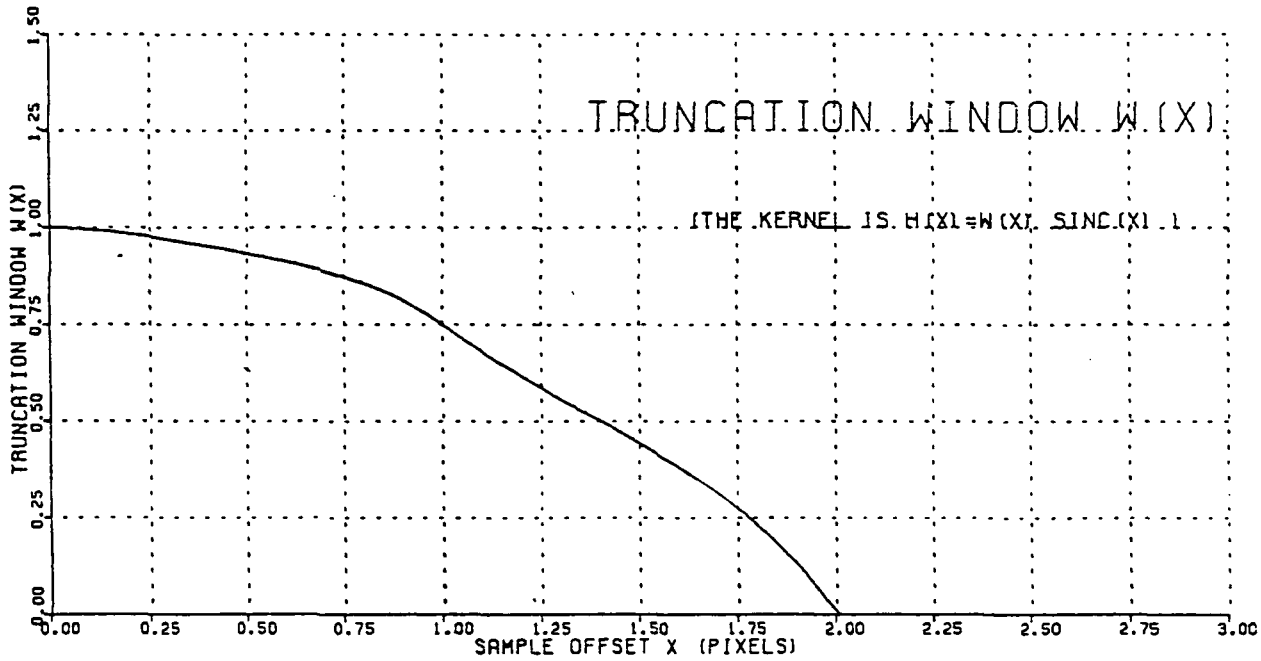
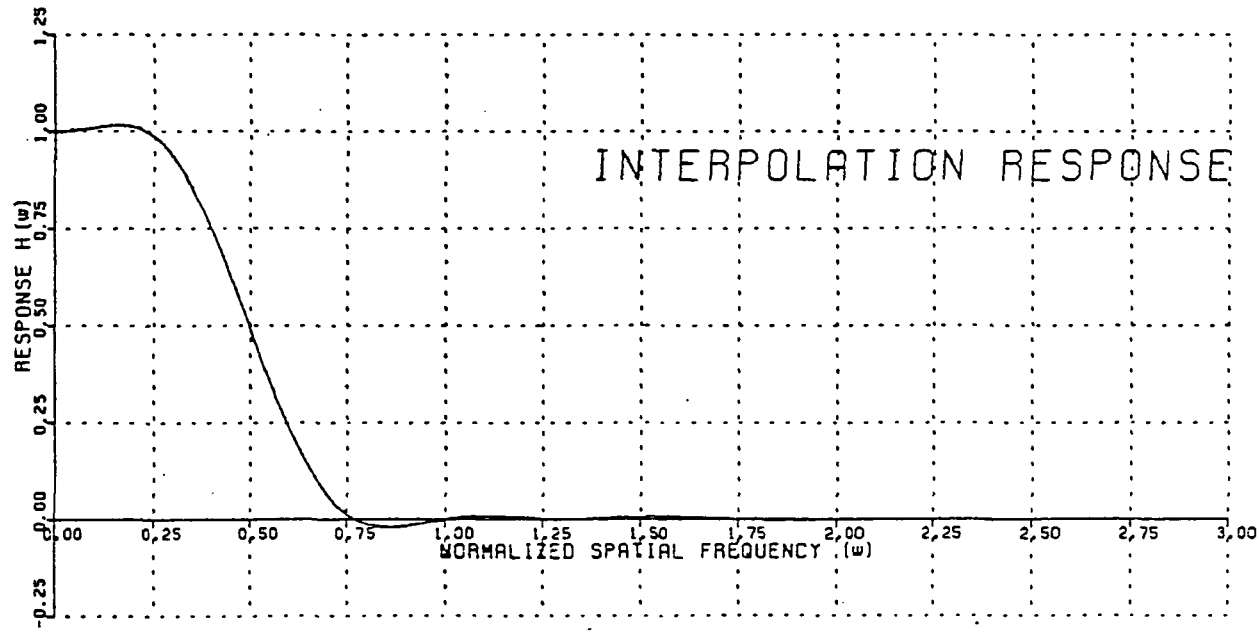



Figure A4-9b. PERIODIC CUBIC SPLINE (4 Points)



The kernel and response are very similar to those of the previous method. The resemblance is even more striking in the next case.

A.4.10 Periodic Quintic Spline (Figure A4-10)

The kernel is

$$h(d) = \begin{cases} \dfrac{32-60|d|^2+45|d|^4-17|d|^5}{32} & \text{for } |d|\le 1 \\ \dfrac{-20(|d|-2)^2 + 35(|d|-2)^4 + 15(|d|-2)^5}{32} & \text{for } 1<|d|\le 2 \\ 0 & \text{otherwise} \end{cases}$$

Compare with figure A4-8a

A.4.11 Truncated Sinc (Figures A4-11a, A4-11b, A4-11c and A4-11d)

The sinc kernel is truncated at $\pm 2$, and the kernel is modified, as described in section A3.5, for exact constant reconstruction. The exact kernel is

$$h(d) = \begin{cases} \text{sinc}(d)+\frac{1}{4}\left(1-\sum_{k=-2}^{1} \text{sinc}(|d|+k)\right) & \text{for } |d|\le 1 \\ \text{sinc}(d)+\frac{1}{4}\left(1-\sum_{k=-3}^{0} \text{sinc}(|d|+k)\right) & \text{for } 1<|d|\le 2 \\ 0 & \text{otherwise} \end{cases}$$

A.4.12 Kaiser-windowed Sinc (Figures A4-12a, A4-12b, and A4-12c)

The kernel is defined as in A4.11, except that sinc(d) is replaced by w(d) sinc(d) where (for the 4-point interpolator)

$$w(d) = \begin{cases} \dfrac{\sinh\left(\beta\sqrt{1-\left(\frac{d}{2}\right)^2}\right)}{(\sinh \beta)\sqrt{1-\left(\frac{d}{2}\right)^2}} & \text{for } |d|\le 2 \\ 0 & \text{otherwise} \end{cases}$$

This is actually a recently-proposed alternative form of the Kaiser window, which is more easily implemented [A-8]. The original form was also tried, with very similar results.

A frequency response very similar to that of any of the near-band-limited interpolators can be obtained by selection of the parameter $\beta$. The ones selected here had near-asymptotic frequency responses near $\omega=0$ and for large $\omega$.

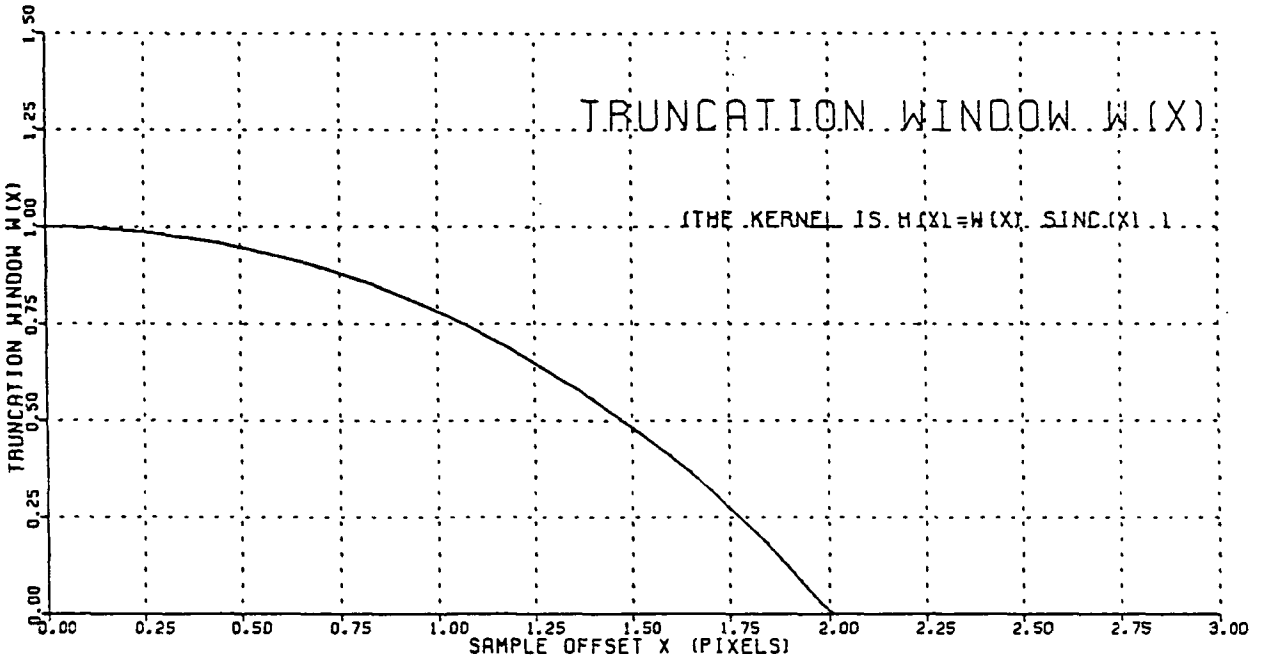
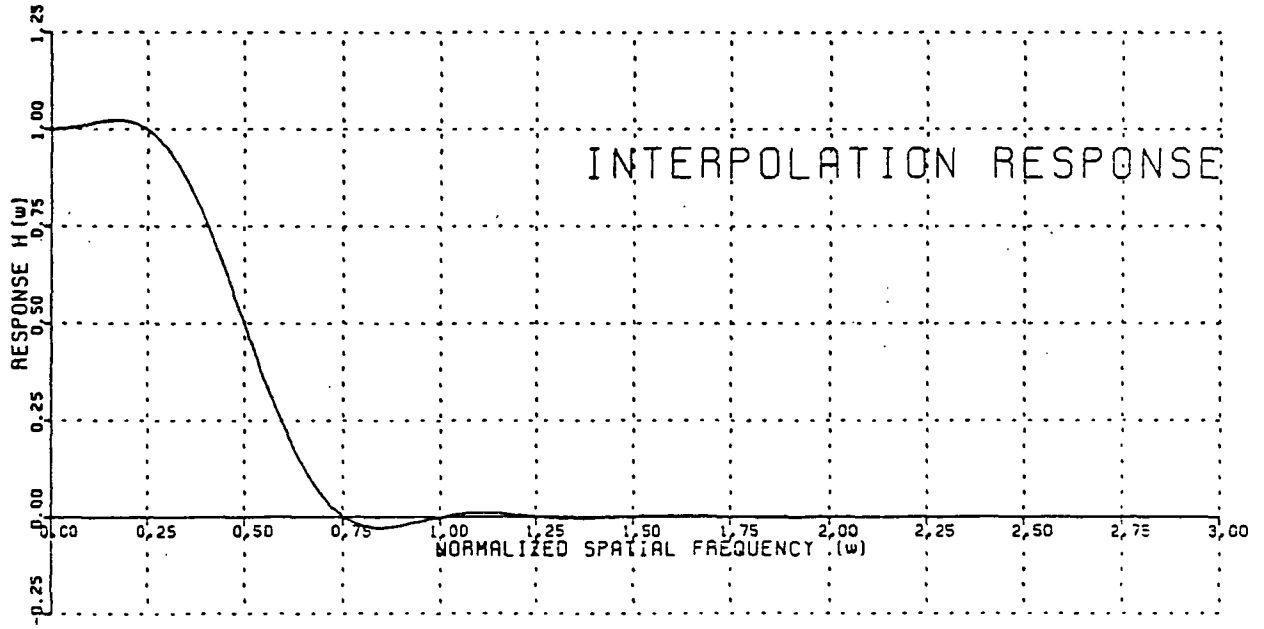



Figure A4-10. PERIODIC QUINTIC SPLINE (4 Points)



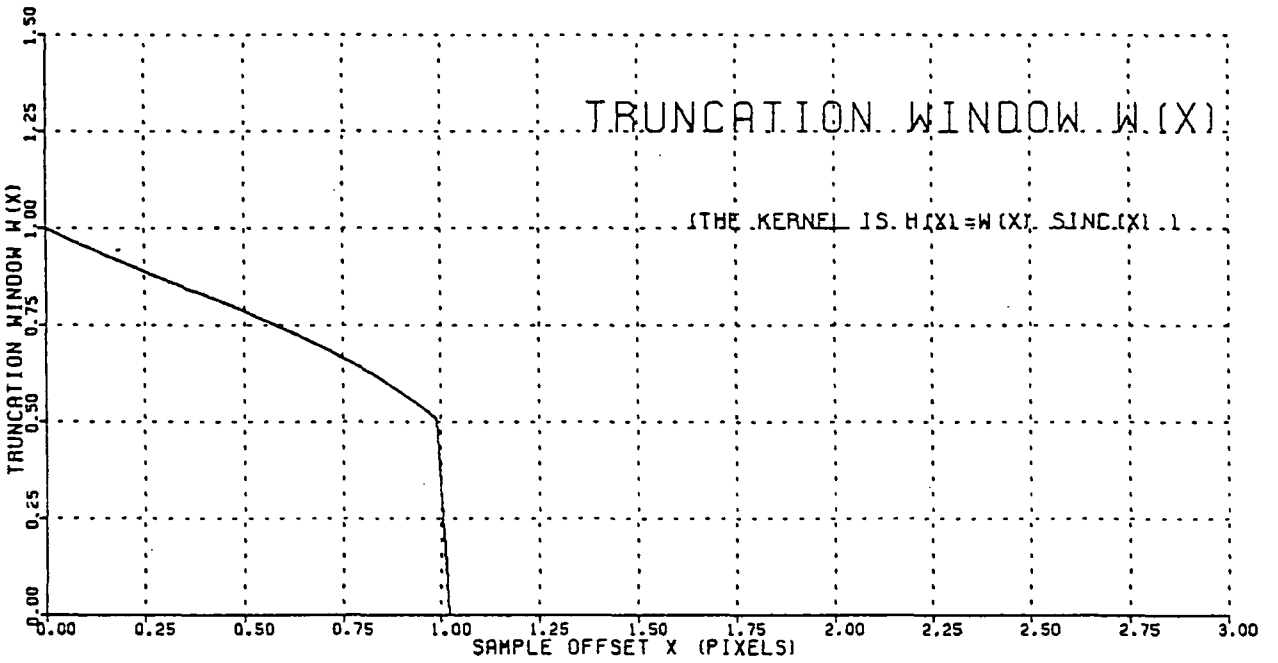
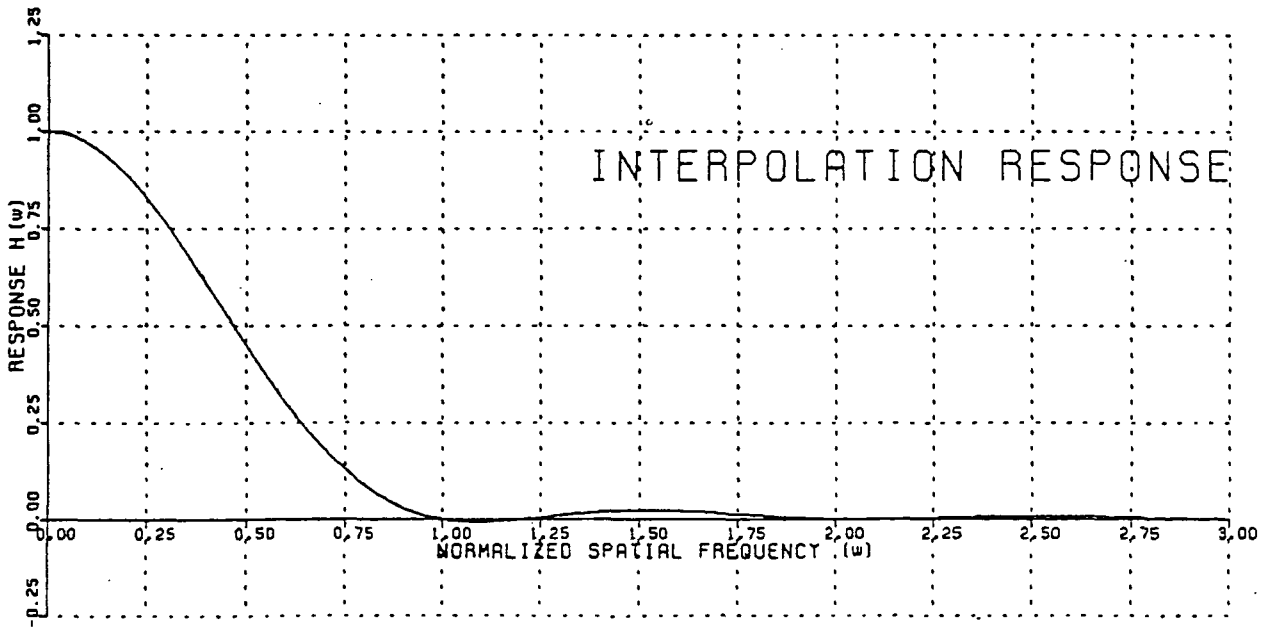



Figure A4-11a. MODIFIED SINC (2 Points)



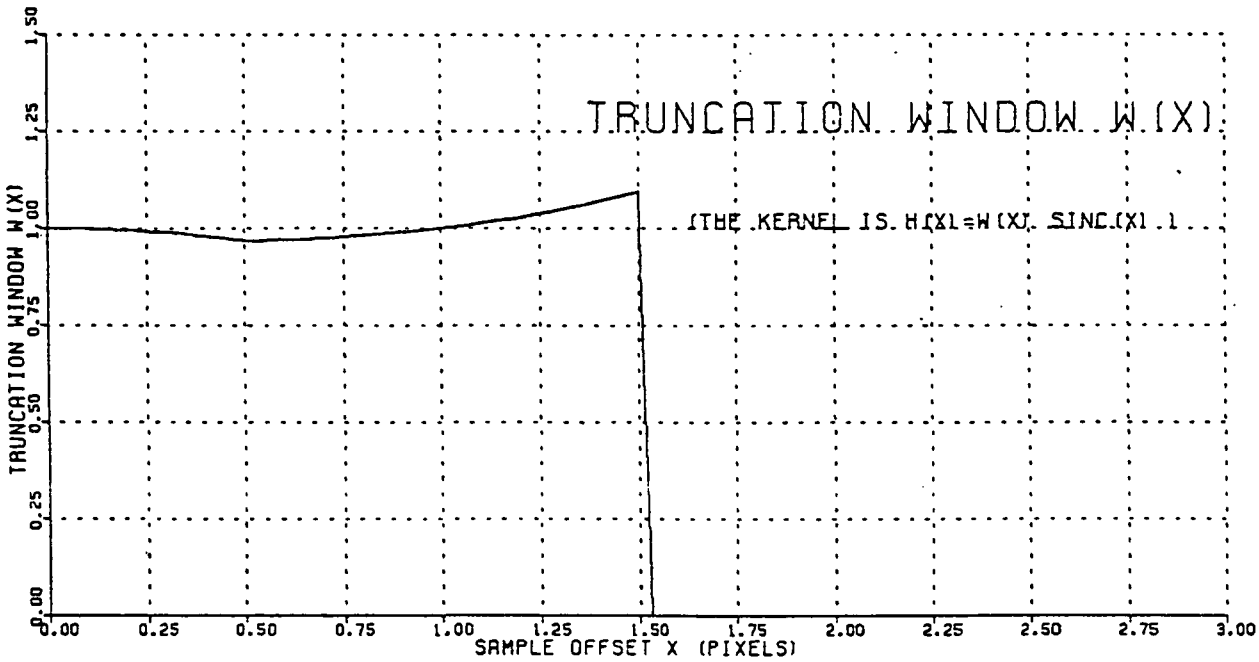
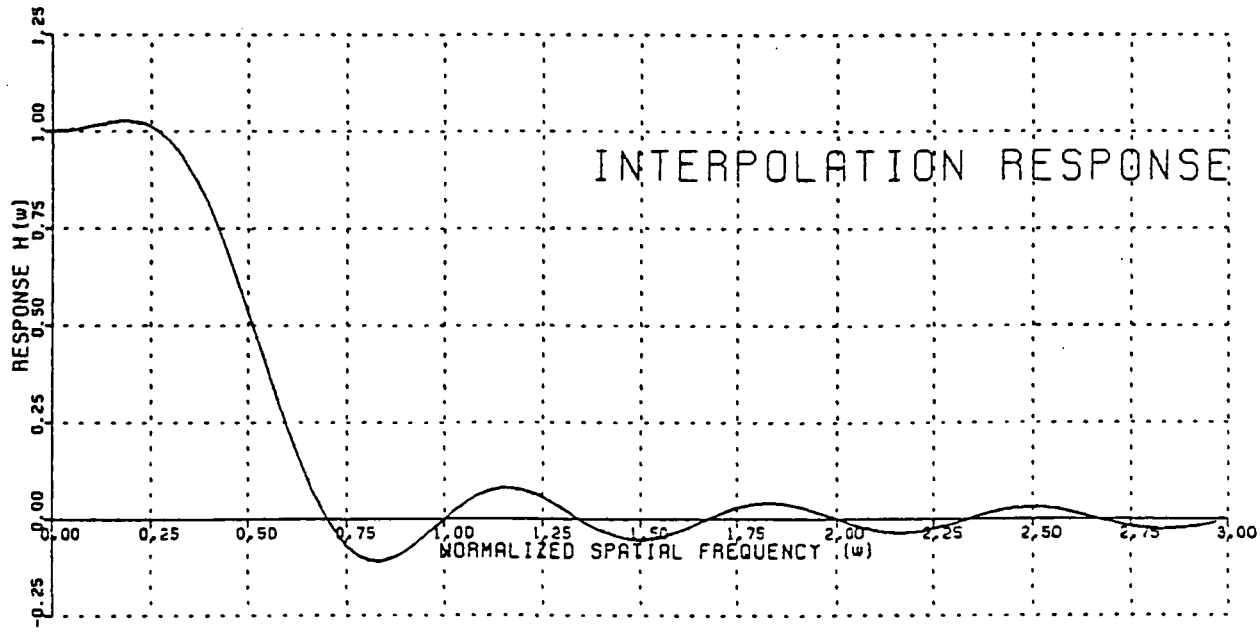



Figure A4-11b. MODIFIED SINC (3 Points)



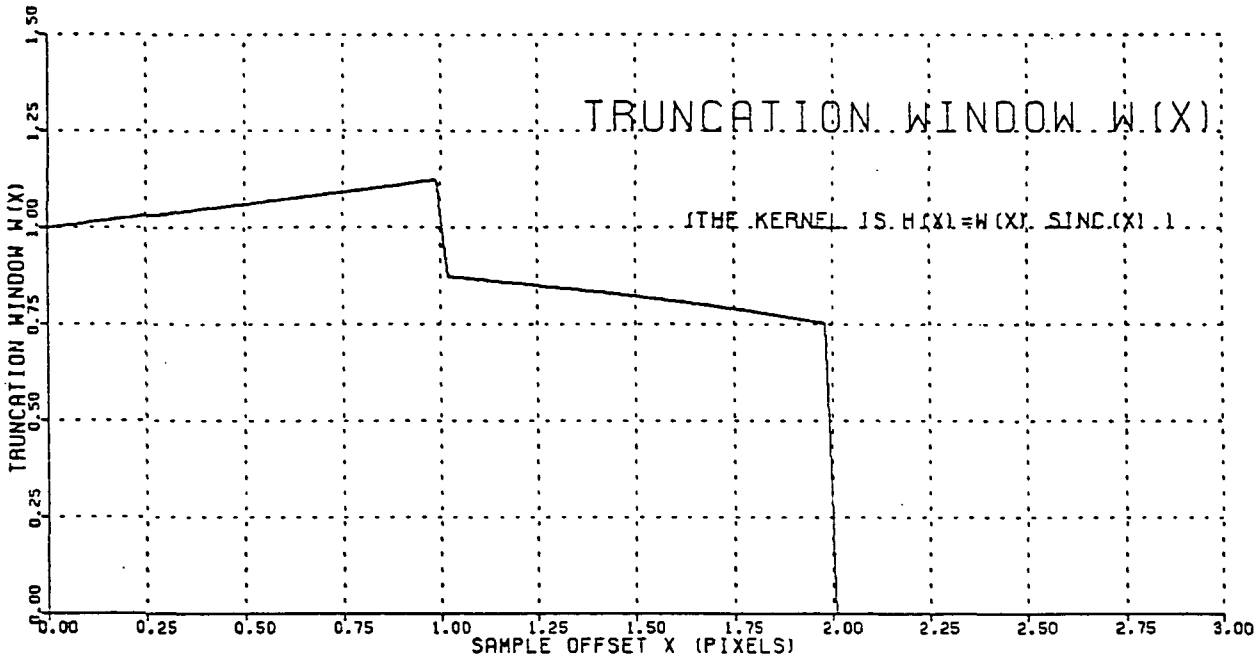
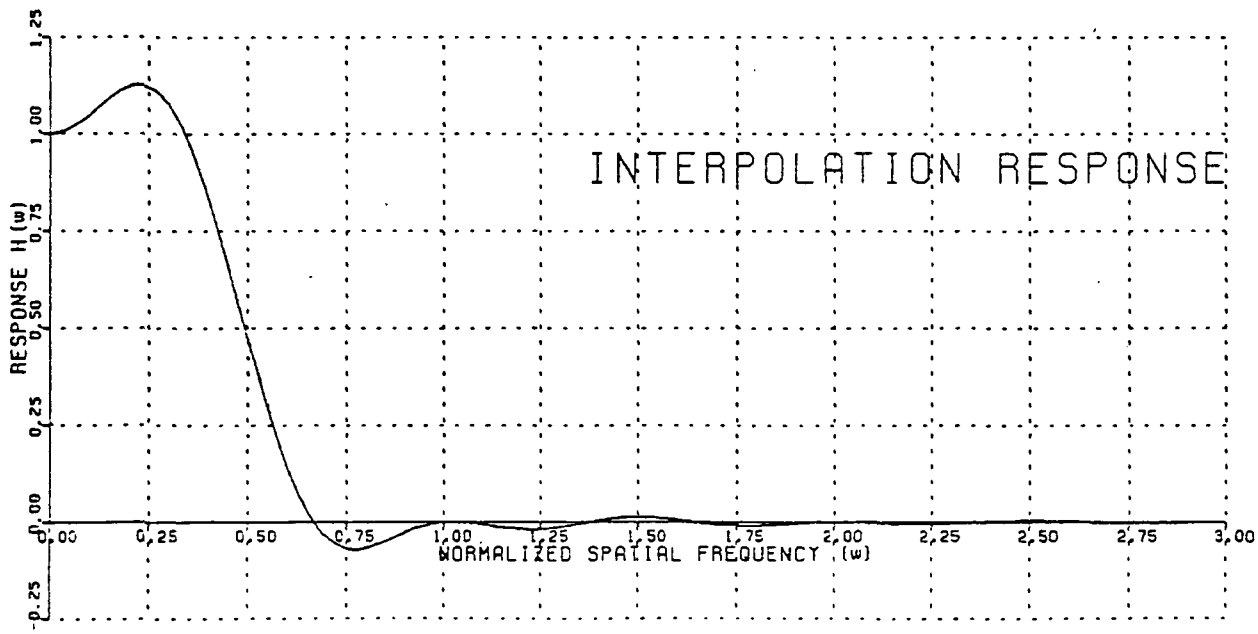



Figure A4-11c. MODIFIED SINC (4 Points)



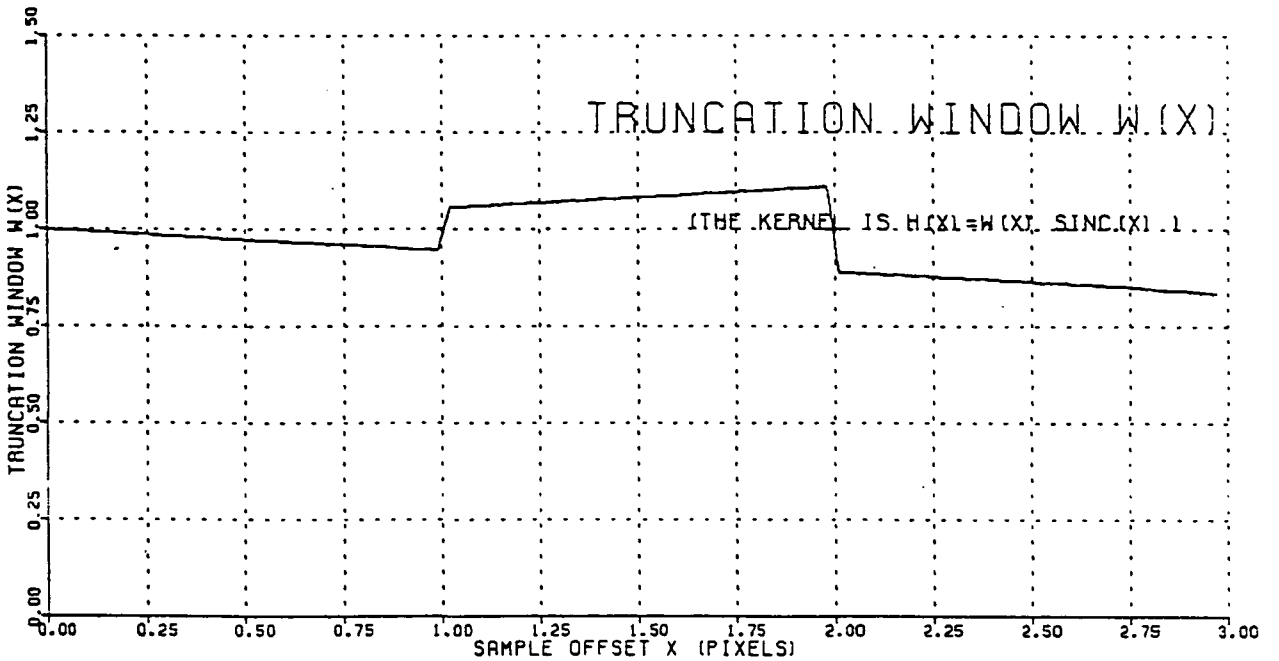
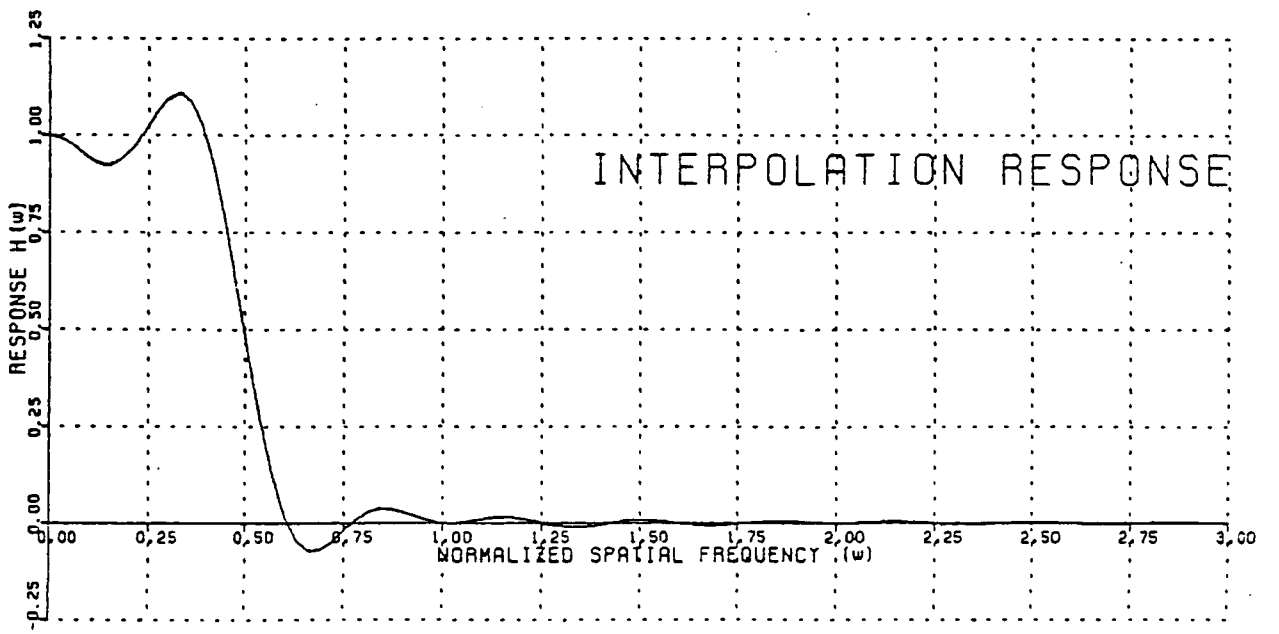



Figure A4-11d. MODIFIED SINC (6 Points)



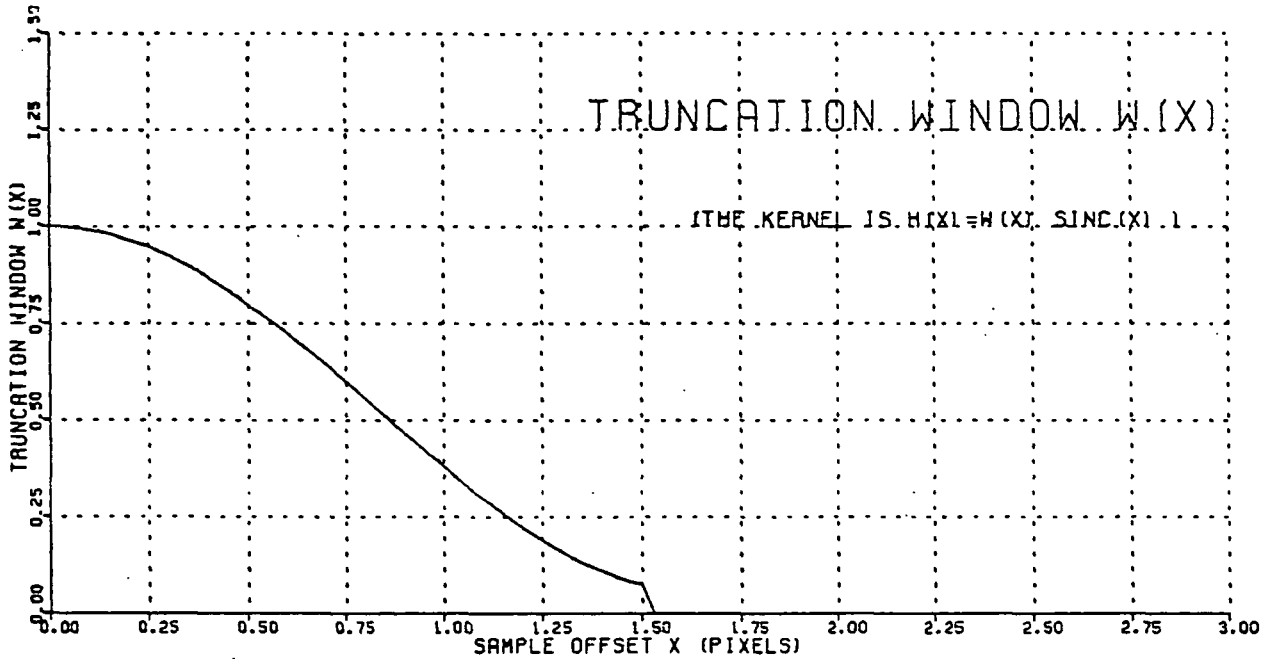
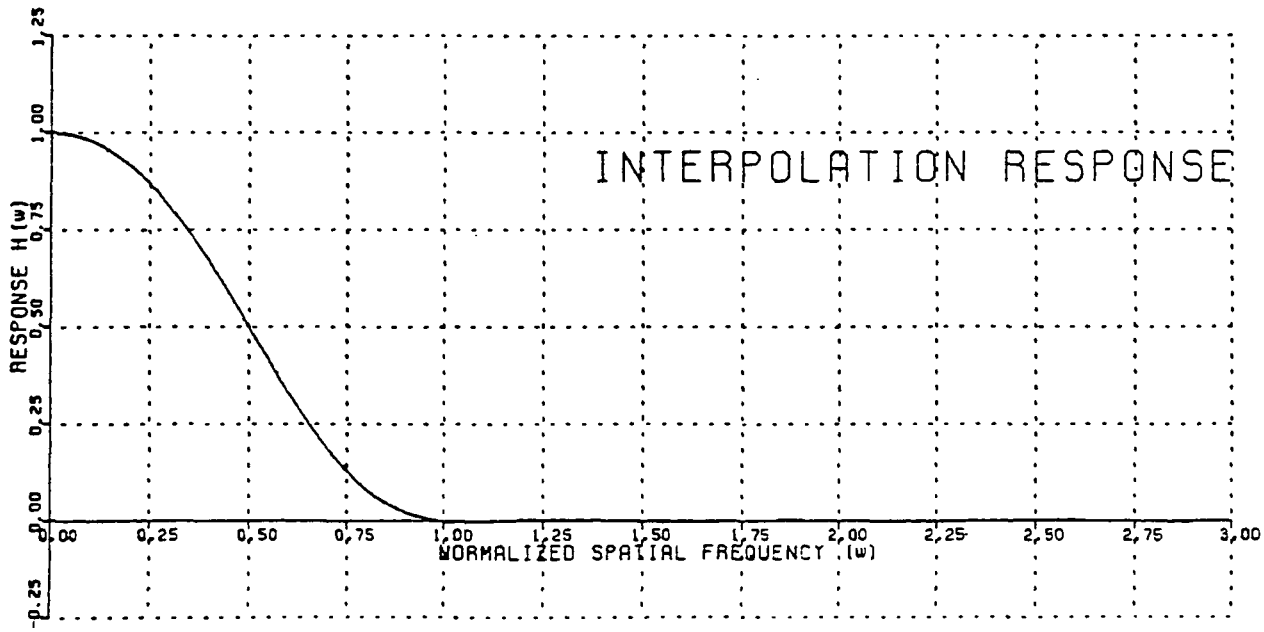



Figure A4-12a. KAISER WINDOW (B=4.95, 3 Points)



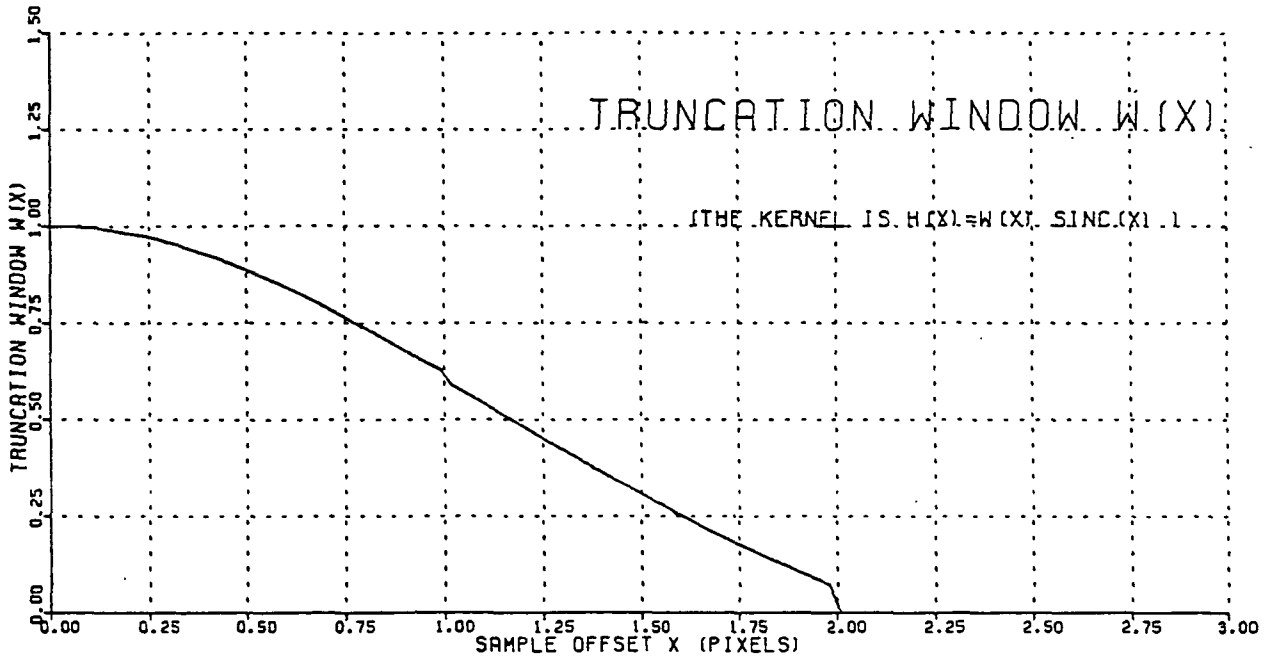
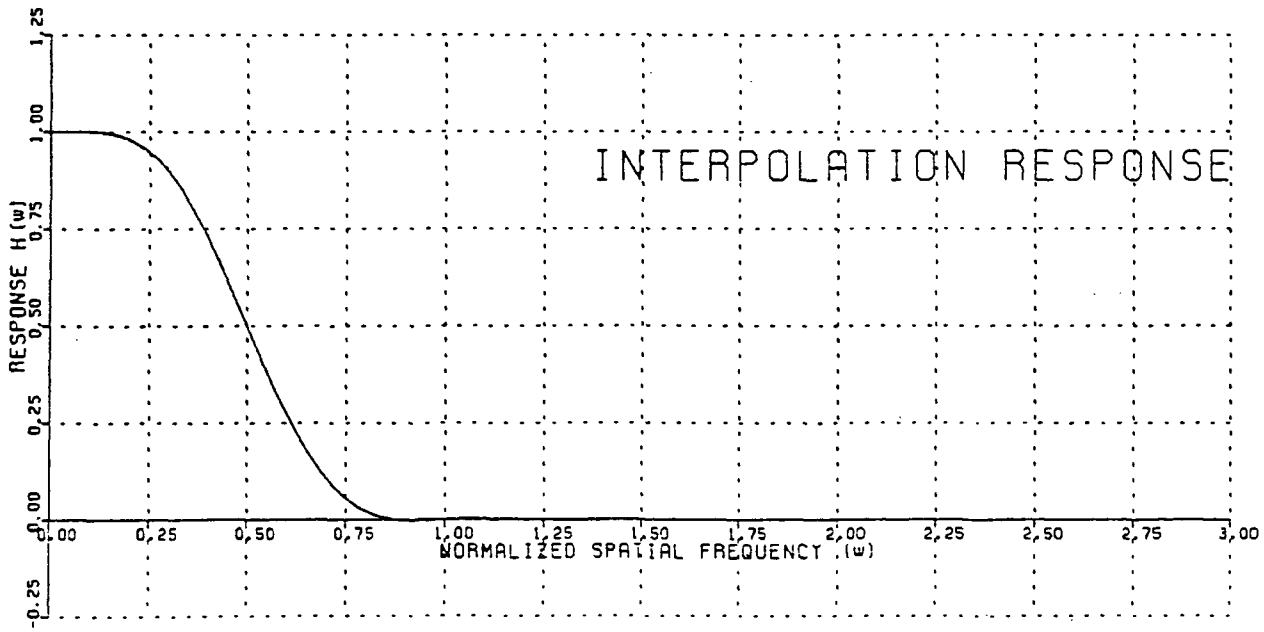



Figure A4-12b. KAISER WINDOW (B=4.73, 4 Points)



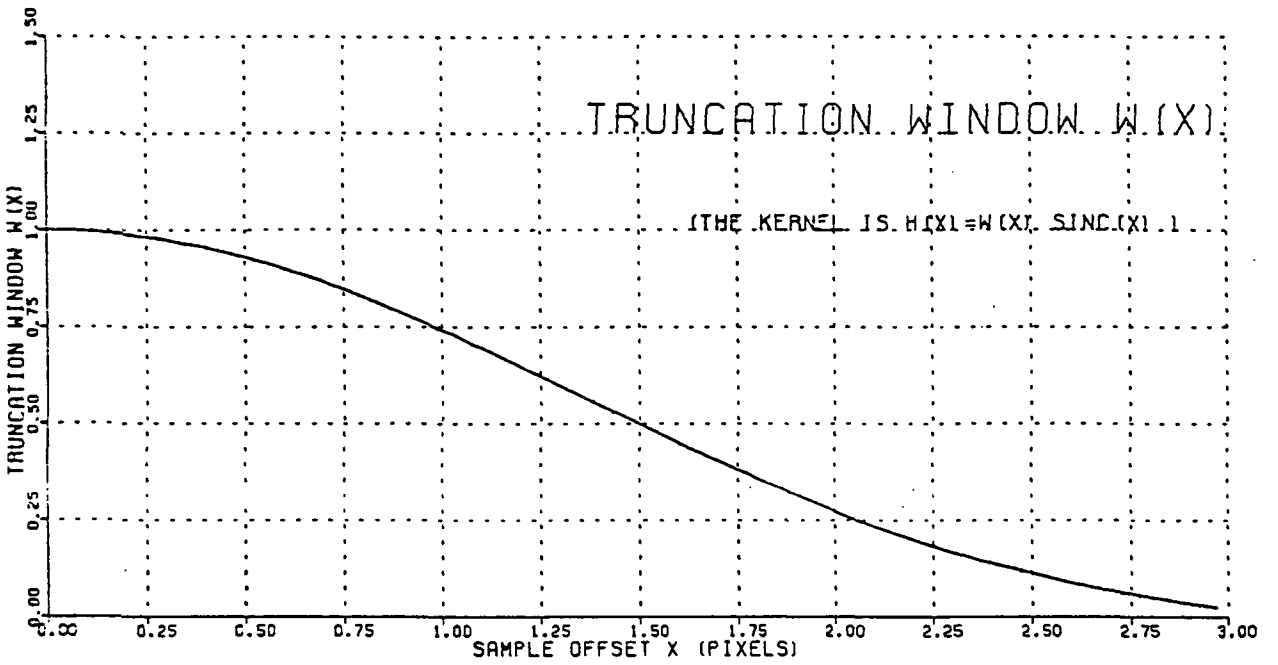
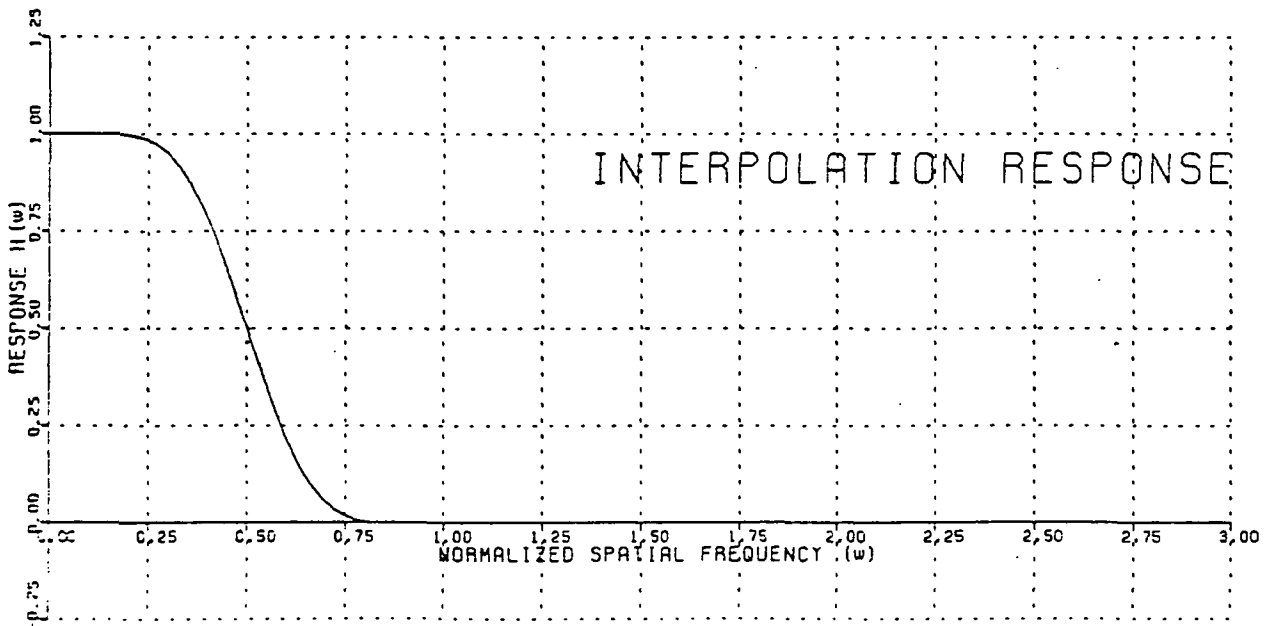



Figure A4-12c. KAISER WINDOW (B=6.25, 6 Points)



A.4.13 Hamming - windowed Sinc (Figure A4-13)

The kernel is defined as in A.4.11, with sinc (d) replaced by w(d) sinc(d) where

$$w(d) = \begin{cases} 0.54 + 0.46 \cos \frac{\pi d}{2} & \text{for } |d| \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

A.4.14 Cosine - windowed Sinc (Figure A4-14)

The kernel is defined as in A.4.11, with sinc (d) replaced by w(d) sinc(d) where

$$w(d) = \begin{cases} \cos \frac{\pi d}{4} & \text{for } |d| \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

A.4.15 Cubic Convolution (Figures A4-15 and A4-9b)

These techniques were apparently derived by direct kernel construction [A-2], but one of the forms is periodic cubic spline interpolation [1-1]. An earlier form approximated the sinc kernel by a smooth, piecewise-cubic function:

$$h(d) = \begin{cases} 1-2|d|^2 + |d|^3 & \text{for } |d| \leq 1 \\ 4-8|d|+5|d|^2-|d|^3 & \text{for } 1 \ |d| \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

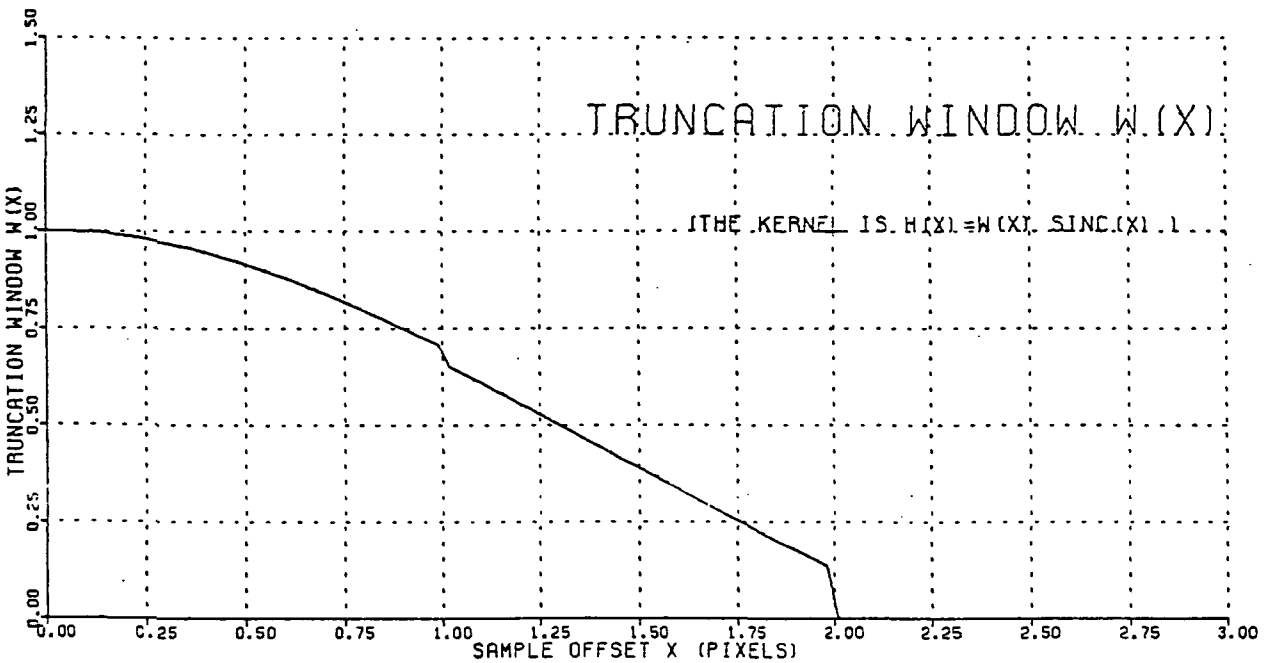
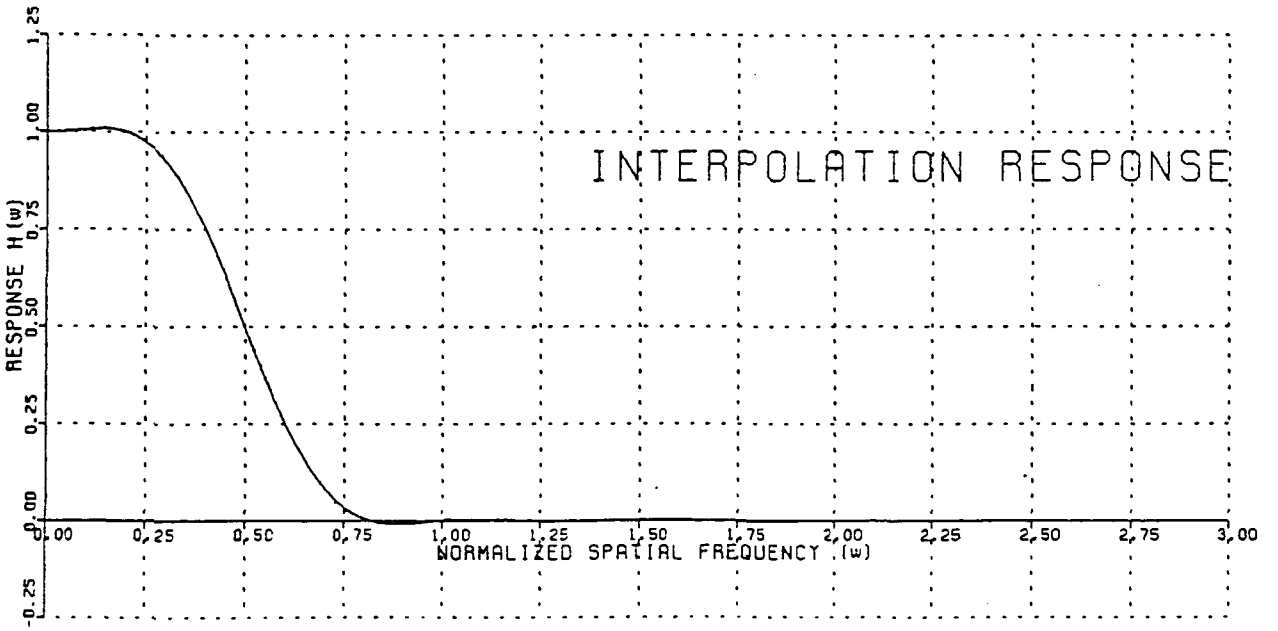



Figure A4-13. HAMMING WINDOW (4 Points)



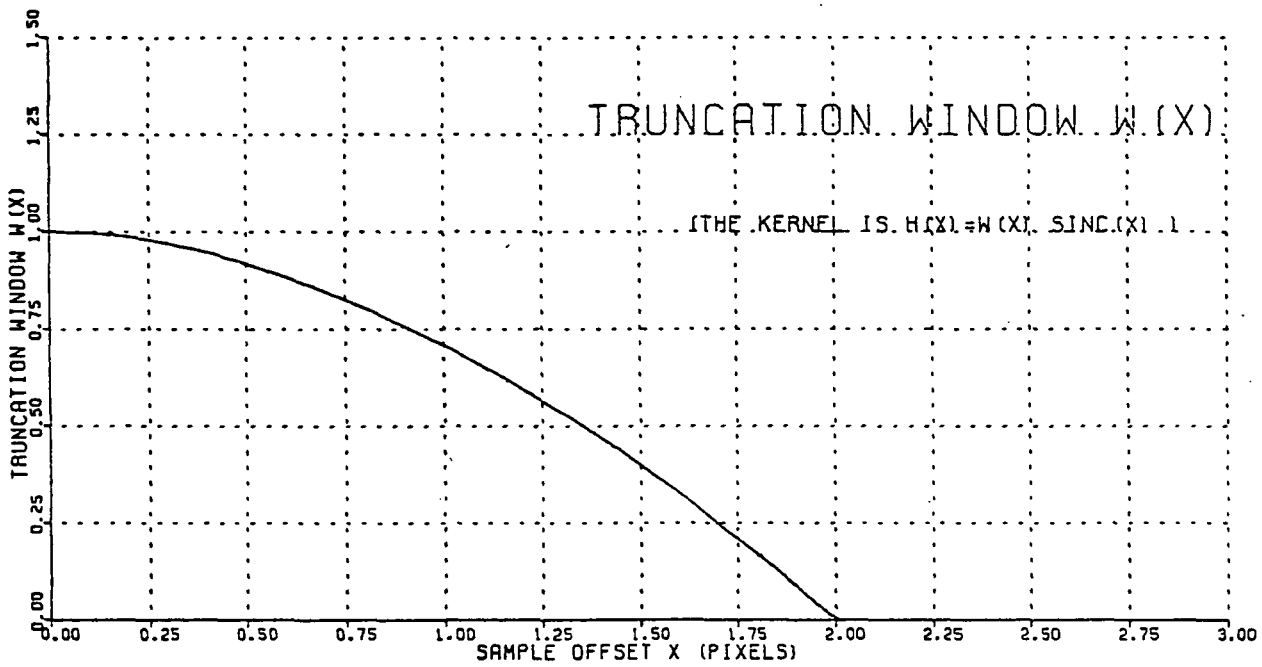
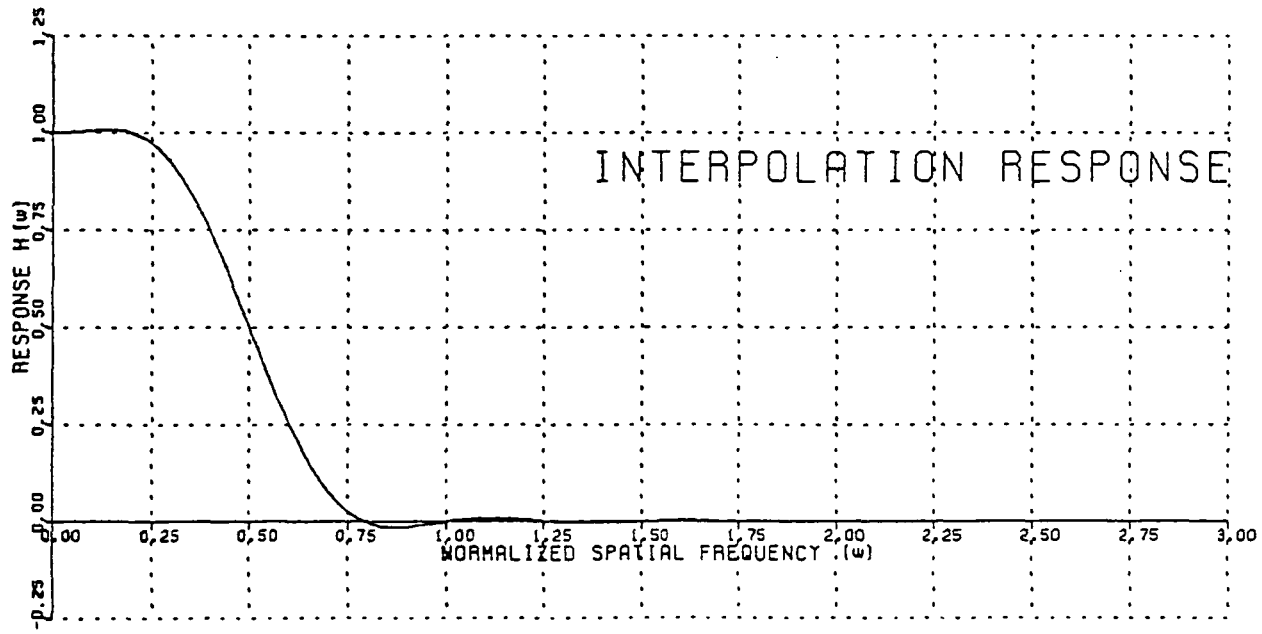



Figure A4-14. COSINE WINDOW (4 Points)



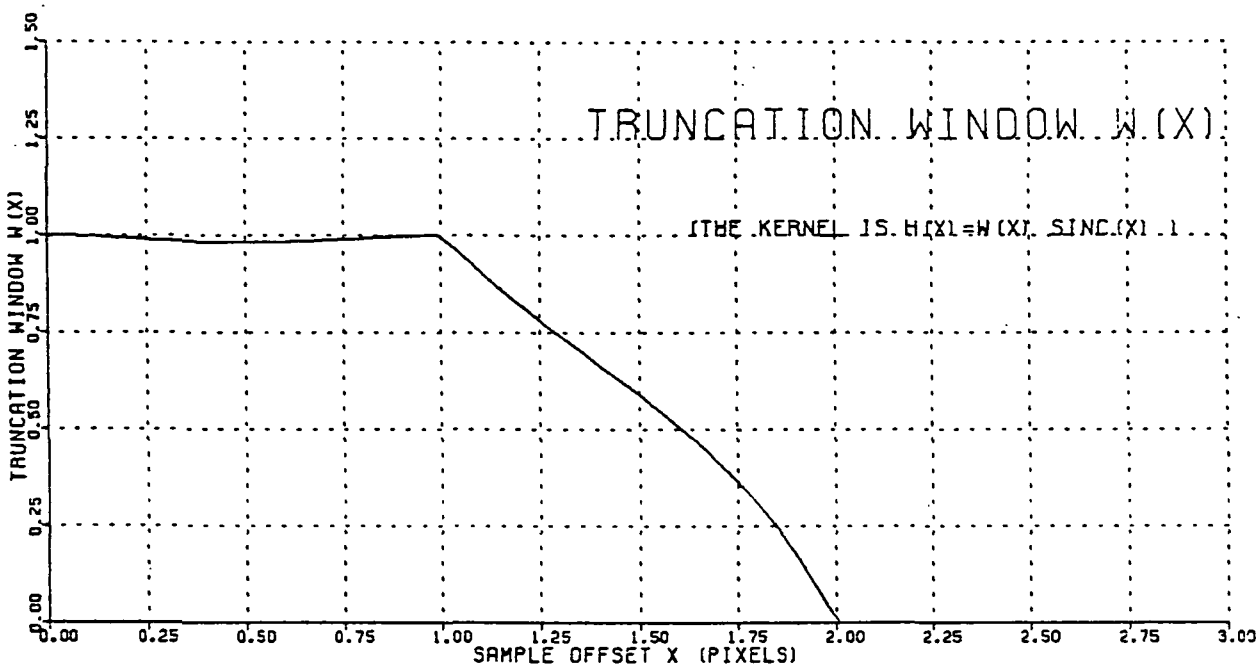
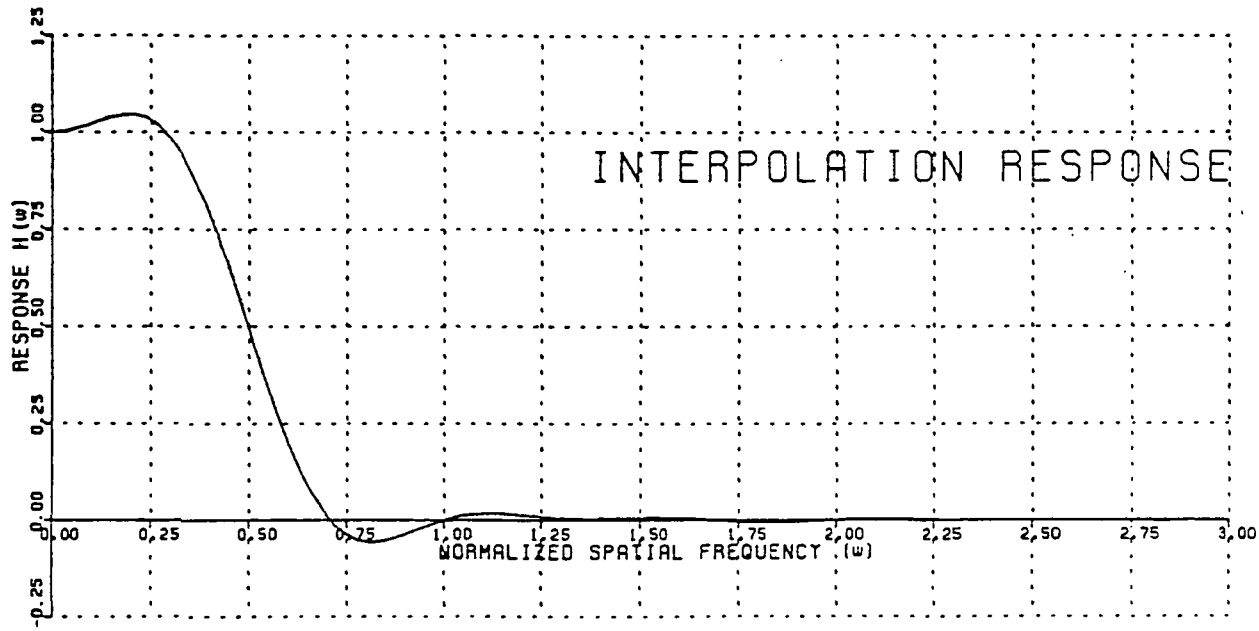



Figure A4-15. CUBIC CONVOLUTION (4 Points)



## A.5 PERFORMANCE COMPARISONS

### A.5.1 Digital Implementation Efficiency

Some of the techniques described in section A.4 are at least moderately computer-efficient. It may, therefore, be feasible to compute the weights as they are needed. Table A5-1 lists these kernels and Table A5-2 lists the number of basic computer operations needed to implement them for image resampling (in the case of the periodic quintic spline, which approximates trigonometric polynomial interpolation, some optimization would reduce the number of multiplications).

Table A5-1. Kernels of 1, 2, 3 and 4 Point Interpolation Methods

| Kernel | Center Portion of Kernel | Second Portion of Kernel |
|---|---|---|
| Nearest Neighbor | 1.0 | 0.0 |
| Bilinear | $1 - \|X\|$ | 0.0 |
| 2 Point Per. Cubic Spline | $1 - 3\|X\|^2 + 2\|X\|^3$ | 0.0 |
| Quadratic | $1 - \|X\|^2$ | $1 - 1.5\|X\| + 0.5*\|X\|^2$ |
| Cubic Osculatory | $1-2.5\|X\|^2+1.5\|X\|^3$ | $2-4\|X\|+2.5\|X\|^2-0.5\|X\|^3$ |
| Cubic Spline | $1-0.2\|X\|-1.8\|X\|^2+\|X\|^3$ | $\frac{24-46\|X\|+27\|X\|^2-5\|X\|^2}{15}$ |
| Cubic Lagrange | $1-0.5\|X\|-\|X\|^2+0.5\|X\|^3$ | $\frac{6-11\|X\|+6\|X\|^2-\|X\|^3}{6}$ |
| Periodic Cubic Spline (4 Pt) (Original TRW Kernel) | $1-2.25\|X\|^2+1.25\|X\|^3$ | $3-6\|X\|+3.75\|X\|^2-0.75\|X\|^3$ |
| Periodic Quintic Spline (4 Pt) | $\frac{32-60\|X\|^2+45\|X\|^4-17\|X\|^5}{32}$ | $\frac{-20(\|X\|-2)^2+35(\|X\|-2)^4+15(\|X\|-2)^5}{32}$ |
| Alternate TRW Kernel | $1-2\|X\|^2+\|X\|^3$ | $4-8\|X\|+5\|X\|^2-\|X\|^3$ |

Table A5-2. Operations per Reconstructed Pixel

| METHOD | "Nearer" Decision | Add./Subt. | Multiply | Divide |
|---|---|---|---|---|
| Nearest Neighbor | 2 | 0 | 0 | 0 |
| Bilinear | 0 | 3 | 4 | 0 |
| 2 point Per. Cub. Spline | 0 | 6 | 9 + 4 | 0 |
| Quadratic | 2 | 20 | 20 + 9 | 0 |
| Lagrange 4 point | 0 | 25 | 30 + 16 | 0 |
| Cubic Osculatory | 0 | 25 | 35 + 16 | 0 |
| Cubic Spline | 0 | 30 | 30 + 16 | 0 |
| Periodic Cubic Spline (4 pt) | 0 | 25 | 30 + 16 | 0 |
| Truncated Sinc (N X N Pixel) | 0 | 2N | $N^2$ + 2N + 2 sinc calculations | 2N |
| Truncated Sinc (NN ± 5 pixels) | 0 | 22 + 2 sinc calculations | 143 | 22 |

A.5.2 The Gaussian Reconstruction Test

The Gaussian function

$$f(x) = \frac{1}{2\pi} \quad e^{-x^2/2}$$

with Fourier transform

$$f(\omega) = e^{-\omega^2/2}$$

was used as simulated data and was reconstructed using various interpolators. The approximation was actually implemented in the frequency domain, to show the spectral distribution of error. The test was also run on the data function $g(x) = f(x - .5)$, the function f shifted by 1/2 pixel.

The basic equation is

$$F(\omega) = H(\omega)\, e^{-j\omega A} \sum_{K=-\infty}^{\infty} \exp[-(\omega + 2k\pi)^2/2 - 2KA\pi j]$$

where A is the data shift (either 0.0 or 0.5 here). The sum was truncated to 21 terms.

The absolute spectral error was plotted. (Figures A5-2a and A5-2b) Among the 4-point methods tested, the periodic cubic spline method produced a minumum value of maximum error. The non-periodic spline method had slightly less error, in the mean, with a slightly higher maximum value.

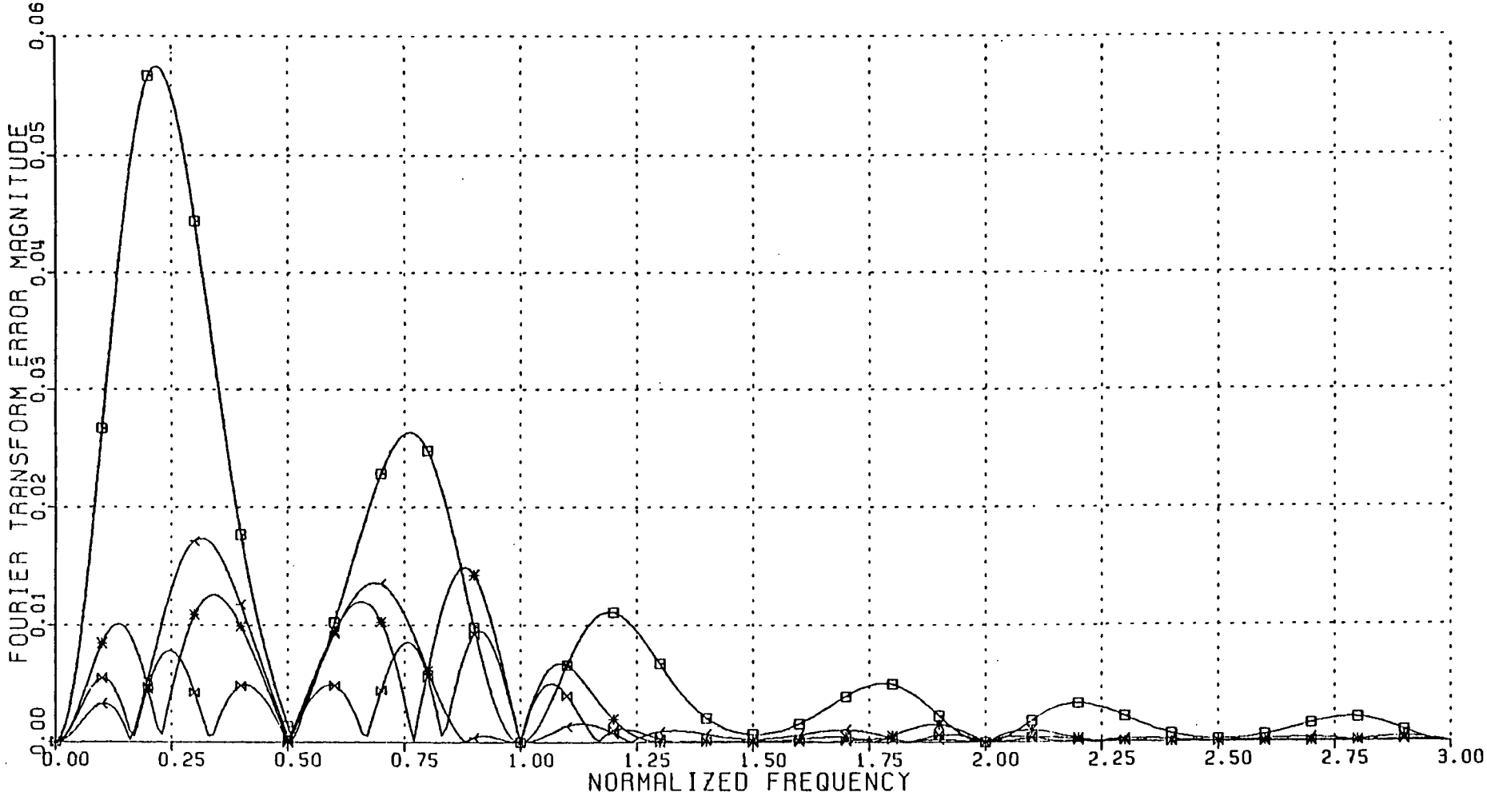



Figure A5-2a. GAUSSIAN RECONSTRUCTION TEST (No Offset)

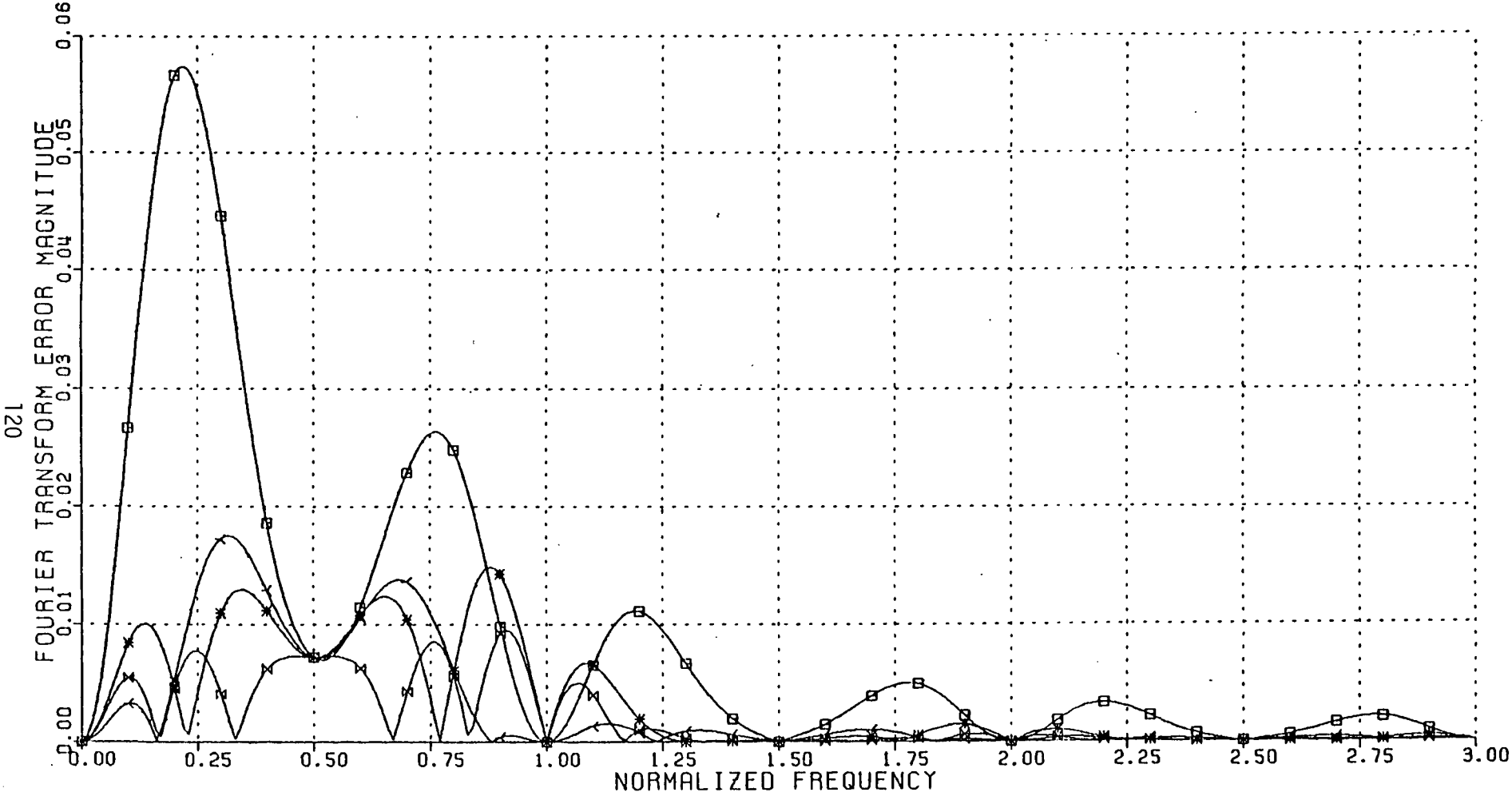



Figure A5-2b. GAUSSIAN RECONSTRUCTION TEST (1/2 Pixel Offset)

### A.5.3 The Bandlimited Reconstruction Test

Portions of each of two 256 x 256 pixel images (with 8 bit resolution) were enlarged by a factor of 3.29 to another image of the same size. For a reference enlargement, the sinc function, truncated at ± 5.5 pixels, was used.

The same parts of these images were then enlarged using several of the alternative techniques and the results were compared with the reference. Mean, mean square, and maximum disparities were noted.

Although one of the images was a portrait of a woman ("Terry") (Figure A5-3a) and the other was a segment of a Landsat scene of North Carolina (Figure A5-3b), similar results were obtained in the two cases. The trigonometric polynomial technique and TRW cubic convolution came out best, with other cubic 4-point techniques also performing well. Linear interpolation did reasonably well, but the nearest-neighbor method did very poorly, as expected (Table 5-3).

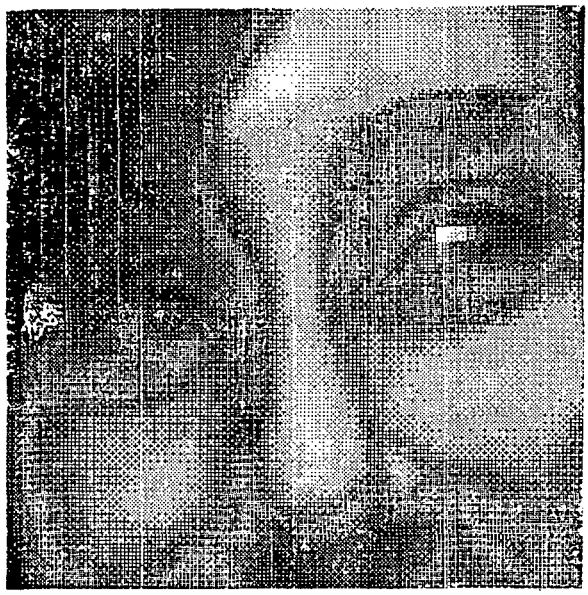


Figure 5.3.1. Enlarged Portion of "Terry"

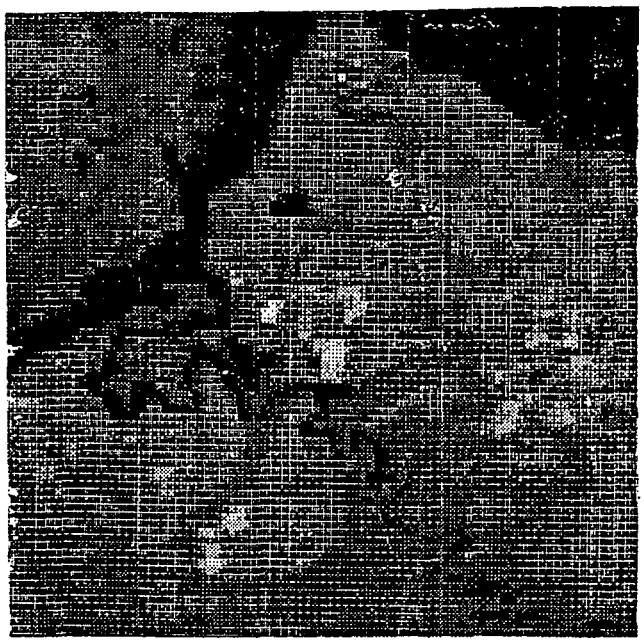


Figure 5.3.2. Enlarged Portion of Landsat Image

Table A5-3 Bandlimited Reconstruction Test Results

| Method | Terry | | | Landsat | | |
|---|---|---|---|---|---|---|
| | Mean $\Delta$ | Mean $\Delta^2$ | Max $\Delta$ | Mean $\Delta$ | Mean $\Delta^2$ | Max $\Delta$ |
| Nearest Neighbor | 3.45 | 11.34 | 189 | 0.49 | 0.87 | 7 |
| Bilinear | 1.14 | 1.63 | 34 | 0.33 | 0.61 | 4 |
| 2 Point Per. Cubic Spline | 1.21 | 1.73 | 32 | 0.33 | 0.61 | 4 |
| Quadratic | 1.39 | 3.73 | 50 | 0.24 | 0.50 | 2 |
| Lagrange 4 Point | 0.92 | 1.33 | 19 | 0.25 | 0.52 | 3 |
| Cubic Osculatory | 0.90 | 1.32 | 20 | 0.23 | 0.48 | 2 |
| Cubic Spline | 0.89 | 1.29 | 18 | 0.28 | 0.55 | 3 |
| Periodic Cubic Spline | 0.84 | 1.24 | 16 | 0.23 | 0.48 | 2 |
| Trig. Polynomial | 0.84 | 1.23 | 16 | 0.23 | 0.49 | 2 |
| Discrete Cosine | 0.85 | 1.25 | 16 | 0.25 | 0.51 | 3 |

### A.5.4 The Unconstrained Reconstruction Test

The same two images were used for this test as for the bandlimited reconstruction test.

First, the fourth row of the image was reconstructed using alternate pixels of the first, third, fifth, and seventh rows. The nearest pixels used for reconstruction were one row and one column away from the reconstruction point (this is the "worst case"). The estimated row was then compared with the actual fourth row of the image. This process of reconstruction of one row out of seven was repeated along the image, for a total of 9000 reconstruction pixels (to avoid end effects, only 250 pixels of each row were used in the comparison). The mean, mean square and maximum disparity were again noted. It should be remarked that the outcome of this test depended solely upon two values of the interpolation kernel, namely, $h(0.5)$ and $h(1.5)$, because only the worst case offset is used. Also dropping alternate rows and columns raises the image bandwidth by decreasing the correlation of adjacent sample values.

With the "Terry" image, the kernels with near-asymptotic responses performed best, with the trigonometric polynomial and TRW cubic convolution techniques also doing well.

With the Landsat image, the two-point methods did best, the near-asymptotic cubic methods were next, and the other cubic methods also did well (Table A5-4).

Table A5-4 Unconstrained Reconstruction Test Results

| Method | Terry | | | Landsat | | |
|---|---|---|---|---|---|---|
| | Mean Δ | Mean $\Delta^2$ | Max Δ | Mean Δ | Mean $\Delta^2$ | Max Δ |
| Nearest Neighbor | 10.03 | 865.47 | 231.00 | 1.05 | 2.46 | 9.00 |
| Bilinear | 7.06 | 350.47 | 123.25 | 0.82 | 1.24 | 8.25 |
| 2 Point Per. Cubic Spline | 7.06 | 350.47 | 123.25 | 0.82 | 1.24 | 8.25 |
| Quadratic | 7.30 | 409.89 | 145.89 | 0.88 | 1.41 | 7.56 |
| Lagrange 4 Point | 6.97 | 349.68 | 123.47 | 0.84 | 1.28 | 8.54 |
| Cubic Osculatory | 6.97 | 349.68 | 123.47 | 0.84 | 1.28 | 8.54 |
| Cubic Spline | 6.99 | 350.13 | 123.51 | 0.85 | 1.31 | 8.59 |
| Periodic Cubic Spline | 7.03 | 351.22 | 126.36 | 0.87 | 1.36 | 8.68 |
| Trig. Polynomial | 7.06 | 352.00 | 127.99 | 0.88 | 1.39 | 8.73 |
| Discrete Cosine | 7.06 | 351.99 | 127.99 | 0.88 | 1.39 | 8.73 |

## A.6 Conclusion

Near-bandlimited interpolators can be designed to have desirable frequency response characteristics by windowing the sinc function. The simple rectangular window yields a sharp-cutoff frequency response with substantial penalties in accuracy at low frequencies. A Kaiser window can be used to obtain a very low ripple response, with a wider transition band. A good compromise is the trigonometric polynomial interpolation window. One of the forms of TRW cubic convolution is very similar in performance to trigonometric polynomial interpolation. Spectral models for the signal and for the noise may aid in the design effort.

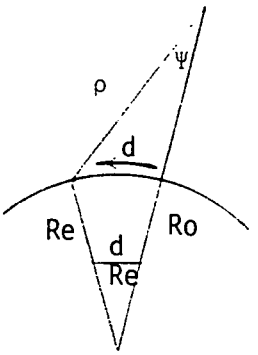
The use of other types of interpolators is based upon a priori information about the class of functions to be interpolated. The discrete orthogonal transform method, the attenuation factor method, direct kernel construction, or one of the classical types of polynomial interpolation may be used in this case. Another possibility is to fit a least-squares line, parabola, or other curve and then to interpolate only the deviations of the samples from the least-squares estimate, using some appropriate interpolator.

# APPENDIX B

# SCANNING GEOMETRY CONSIDERATIONS

## Along-Scan Distance Expression

From the geometry shown below,




d = along scan distance
Re = earth radius (latitude dependent)
Ro = orbital radius
ρ = scan distance to earth

By law of sines (all angles in radians),

$$\frac{Re}{Sin\Psi} = \frac{Ro}{Sin\left[\pi-\left(\Psi+\frac{d}{Re}\right)\right]} = \frac{Ro}{Sin\left(\Psi+\frac{d}{Re}\right)}$$

$$Re\ Sin\left(\Psi+\frac{d}{Re}\right) = Ro\ Sin\Psi$$

Determine,

$$d = Re\left[Sin^{-1}\left(\frac{Ro}{Re}\ sin\Psi\right)-\Psi\right]$$

This is much simpler than the expression given by TRW [1-1].
Also this along-scan distance function is invertible:

$$\Psi = ARCTAN\left(\frac{sin\left(\frac{d}{Re}\right)}{\frac{Ro}{Re} - cos\left(\frac{d}{Re}\right)}\right)$$

Since Re is essentially constant over a scan, the coefficients need be calculated only once per scan with only 2 multiplications and a division required at each point. (Polynomial approximation would use 7 multiplications at each point.)

This expression for d is accurate to within 5 meters. For example,

with Re = 6378
K = 7075/6378
B = -.0550483406
A = .3349163686
c ≐ 91.2077 km

(Compare the exact value d = 91.2123 km and the polynomial approximation value d ≐ 91.1803)

with Re = 6357
K = 7075/6357
B = -.0573081146
A = .3346245445
d ≐ 93.9589

(Compare the exact value 93.9636 and the polynomial approximation value 93.9312)

The error is approximately 32 meters for the polynomial approximation in both cases.

Note that this approach is approximately six times better than the TRW approach.

Further Along-Scan Distance Expression Considerations

As discussed previously, image resampling requires an accurate expression for along-scan distance. The primary factors to be considered are:

- earth modeling
- scan direction

Over a 185 x 185 km region, the earth's curvature has a negligible effect on the along-scan distance, but earth-center scanning (in contrast with earth-normal scanning) can produce a 20 meter error in the along-scan distance at certain latitudes. For example, if the scan at 60° latitude were longitudinal, the North half-scan would be about 75 meters longer than the South half-scan. The satellite heading is approximately 18.5° so that non-longitudinal scanning eliminates a large part of the difference. The remaining error is approximately 75 sin (18.5°) meters.

The along-scan distance expression may be used to avoid the direct computation of latitude and longitude at most points. It is thus useful to optimize the approximate calculation of d.

It is suggested that a rational function approximation be chosen for each scan, as described below, to improve both accuracy and efficiency.

Starting with the series

$$\frac{\sin^{-1}(k \sin x)-x}{(k-1)x} = 1 + \left(\frac{k^2+k}{6}\right) x^2 + \frac{k(9k^2-1)(k^2-1)x^4}{120} + \ldots$$

one obtains the rational function approximation

$$\sin^{-1}(k \sin x)-x \doteq (k-1)x\left(\frac{1+Ax^2}{1+Bx^2}\right)$$

where $B = (1-k)(9k^2-1)/20$

$$A = B + \frac{k^2+k}{6}$$

Here, $k = {}^{Ro}/Re$ and $x = \Psi$, the scan angle,

$$d \doteq Re\left[\sin^{-1}\left(\frac{Ro}{Re} \sin\Psi\right) - \Psi\right] \doteq (Ro-Re)\Psi \frac{1+A\Psi^2}{1+B\Psi^2}$$

with A and B defined as above.

This may be optimized to

$$d \doteq \Psi\left[\frac{\frac{B-(Ro-Re)A}{B^2}}{1/B + \Psi^2} + \frac{(Ro-Re)A}{B}\right]$$

The following discussion details a compensation algorithm for non-normal scanning. The error due to non-normal scanning depends not only upon latitude, but on scan line orientation (via satellite heading).

The along-scan distance expression remains quite simple and can be used more often per scan, replacing cruder approximations.

The effect of changing the scanning to normal scanning would be to fit $\theta = 0$ in the expressions.

Latitude/Longitude And Along-Scan Distance

First, ignore earth motion. Direct determination of latitude/longitude as detailed in earlier memo:

Solve:

$$\left[U_x^2 + U_y^2 + \frac{a^2}{b^2} U_z^2\right] \rho^2$$

$$+ 2 \left[X_o U_x + Y_o U_y + \frac{a^2}{b^2} Z_o U_z\right] \rho$$

$$+ \left[X_o^2 + Y_o^2 + \frac{a^2}{b^2} Z_o^2 - a^2\right] = 0$$

for scan vector length $\rho$

$$\rho = \frac{2C}{-B + \sqrt{B^2 - 4AC}}$$

where A, B, C are the quadratic, linear and constant coefficients above.

The intersection with the reference ellipsoid is at

$$R_I = (X_o + U_x\rho,\ Y_o + U_y\rho,\ Z_o + U_z\rho)$$

$$\phi_c = \arctan\left(\frac{Z_o + U_z\rho}{||R_I||}\right)$$

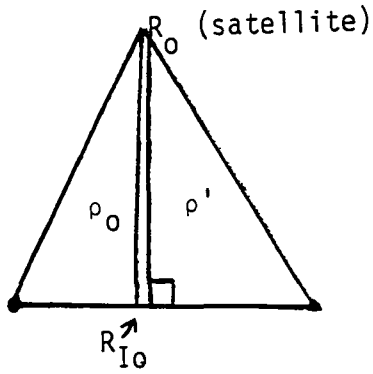
$$\lambda = \omega_e t + \arctan\left(\frac{Y_o + U_y\rho}{X_o + U_x\rho}\right)$$

The previous calculation is to be done at mid-scan and at one scan extreme.

If $R_{Io}$ and $R_{I+}$ are the intersection points,

let

$$d_+ = ||R_{I+} - R_{Io}||$$




Solve for $\rho'$, the perpendicular distance from $R_o$ to the line joining $R_{I+}$ and $R_{Io}$

$$\rho' = \left\| \left(\vec{R}_o - \vec{R}_{Io}\right) - \frac{\left(\vec{R}_o - \vec{R}_{Io}\right) \bullet \left(\vec{R}_{I+} - \vec{R}_{Io}\right)}{\|\vec{R}_{I+} - \vec{R}_{Io}\|} \left(\vec{R}_{I+} - R_{Io}\right) \right\|$$

Note: $\rho' = \rho_o \cos\theta$ where $\theta$ is angular offset due to earth center-pointing altitude $\left(\theta = \arccos\left(\frac{\rho'}{\rho_o}\right)\right)$

The along scan distance expression is the invertible expression (Flat Earth Assumption):

$$d = \rho' \left[\tan(\theta+\Psi) - \tan\theta\right] \pm V\cos\alpha\,\Delta t$$

$$\doteq \rho' \left[\Psi + \frac{(\theta+\Psi)^3}{3} + \frac{2}{15}(\theta+\Psi)^5\right] \pm V\cos\alpha\,\Delta t$$

where $\Delta t$ is the time from mid-scan (see [1-1] page 6-16).

Inverse of Along-Scan Distance

$$\Psi = \arctan\left(\frac{d \pm V\cos\alpha\,\Delta t}{\rho'} + \tan\theta\right) - \theta$$

A suitable approximation for arctan is

$$\arctan X \doteq X - \frac{X^3}{3} + \frac{X^5}{5}$$

$$\text{error} < 10^{-7} * 700 = 7 * 10^{-5} \text{ km}$$

This expression eliminates the need for iterative evaluation of the along-scan distance expression ([1-1], Page 6-17).

(Note: $d \pm V\cos\alpha\ \Delta t$ is the true along-scan distance, taking earth rotation into account.)

# APPENDIX C

# RESAMPLING CONSIDERATIONS

Resampling Algorithm

Resampling interpolation by approximation of the $\sin(\pi x)/\pi x$ kernel using polynomials has been established as a method which avoids loss of resolution and blockiness. The modifications of the TRW convolution methods to be described here relate to the computational efficiency and precision of the method, not to its theoretical basis.

A better polynomial approximation to $\sin(\pi x)/\pi x$ for this application is

$$P(x) = \begin{cases} \dfrac{(x^2-3)*x^2}{2} + 1 & \text{for } |x|<1 \\ \dfrac{[1-(|x|-2)]^2*(|x|-2)}{2} & \text{for } 1<|x|\leq 2 \end{cases}$$

The polynomial approximation P(x)

- .interpolates $\sin(\pi x)/\pi x$ at $0, \pm 1, \pm 2$
- .interpolates its derivative at $0, \pm 1$
- .posesses a continuous second derivative
- .requires only two multiplications and a shift
- .is more accurate at all points
- .has a sharp cutoff lowpass characteristic

A second improvement may be realized by utilization of the property:

$$\frac{\sin\pi(x+k)}{\pi(x+k)} = (-1)^k\left(\frac{x}{x+k}\right)\left(\frac{\sin(\pi x)}{\pi x}\right)$$

Three out of four evaluations are avoided, but a multiplication and a division are necessary. The saving is not great, but if the polynomial P(x) is evaluated only for the nearest neighbor input pixel, and the relationship

$$P(x+k) = (-1)^k\left(\frac{x}{x+k}\right) P(x)$$

is used elsewhere, the very good accuracy of the approximation on [-.5, .5] is automatically transferred to the whole range [-2, 2]. The actual approximating function in this case is no longer a polynomial, but a rational function.

For 16 pixels, the total computational tallies are:

| TRW Method | Revised Method |
|---|---|
| 8 weights ~ 24 mult. | 2 weights ~ 4 mult.+2 shifts |
| +4*16 mult. | 6 weights ~ 7 mult.+6 divides |
| +4*16 mult. | +4*16 mult. |
| = 152 multiplications | +4*16 mult. |
| | = 139 mult.+6 divides+2 shifts |

These tallies ignore the edge effects at scan edges; each method would be similarly affected by these edge effects. At scan edges, output intensities must be interpolated from as few as two input values.

The subject of resampling has been pursued in-depth during the study as described in section 2.3 and in appendix A.

Considerations in Fast Resampling

The material presented in this section was motivated by the need to determine what performance was achievable utilizing fast resampling techniques. The techniques considered are essentially variants of known techniques.

There are several preliminary conclusions to be drawn from this study.

If the data are sampled at four times the Nyquist rate, quadratic resampling using 3 x 3 points gives good results because low frequencies are not heavily aliased.

If the data are sampled at twice the Nyquist rate, cubic spline fitting of the data is to be preferred.

If the data are sampled just at the Nyquist rate, the TRW method or the periodic cubic spline method provides good performance over the passband.

The periodic cubic spline method results in periodic cubic spline reconstruction of the data and, hence, is also quite appropriate for use in the final phase of sub pixel registration using ground control points.

Methods are very easily combined to achieve average properties.

Fast Image Resampling Techniques

The goal of image resampling is to reconstruct continuous image data from discrete samples. The continuous image will not, in general, be reconstructed precisely (unless it has an analytic form involving N parameters and N samples are taken) because the exact resampling method for properly bandlimited data utilizes a weighted combination of an infinite number of samples with the weights given by the sampling kernel

$$\text{sinc}\ (x) = \sin \pi x / \pi x$$

where x is the displacement from the sample point, in pixels.

The resampling methods to be considered here use very few sample points

(1, 2, 3 or 4) with a sampling kernel which, in some sense, approximates the sinc function. Adaptive methods and methods using a variable number of sample points are not treated here.

These methods may be grouped into three categories:

1. Methods based upon standard interpolation techniques (e.g., linear, quadratic, cubic spline)
2. Methods based upon approximation of the exact resampling kernel (the sinc function)
3. Hybrid methods obtained by combining techniques of the previous types in standard or non-standard ways (certain standard combination methods will be described below).

Considerations

The major consideration is, of course, accuracy, but some techniques work very well for certain image types (e.g., polynomial interpolative techniques tend to be good at low spatial frequencies) but not for others (e.g., polynomial interpolative techniques tend to be poor near the edge of the passband.).

The best kernel in the least-squares sense is the truncated sinc function (see [C-1], p. 250) despite the fact that it behaves rather poorly for certain spatial frequencies.

Most of the techniques considered here resample constants exactly. The truncated sinc function does not, however, have this property. Kernels with this property preserve the mean intensity of an image and allow true intensity variations to stand out from extensive areas of very little change. The interpolative methods also resample low-degree polynomials or certain trigonometric polynomials exactly.

A common practice in digital filter design is to minimize the maximum frequency response ripple (i.e., error) in the passband or the stop-band or both. The untruncated sinc function has the response of an ideal low-pass filter and approximations to it can be expected to have somewhat similar characteristics. Thus, if the image has a relatively flat bandlimited spectrum, it is reasonable to judge a kernel by this minimax criterion.

The low-pass filtering effect also tends to minimize "noise" due to detector non-uniformity, however, the very filtering which tends to eliminate the noise tends to smooth sharp "edges" in the data. These edges are frequently desirable constitutents of ground control points ("landmarks") used for high precision image registration by subimage matching.

It is probably safe to summarize the situation by saying that constants ("DC") should be resampled exactly, "edge frequencies" should be resampled with adequate accuracy for image registration and other frequencies should be resampled with an accuracy dependent upon the spectral distribution of typical data.

It should be remarked that the data are not intensities, but intensities corrupted by optical distortion, smoothed by integration over the area of a detector and then corrupted by detector noise. Resampling must be based upon the spectrum of these data.

Another error source in resampling is the quantization of the resampling weights. These weights must be computed, with some roundoff error, from the resampling kernel, or stored, with some degree of precision, in a lookup table. The displacement value, upon which the weights are based, is itself rounded off before computation or table lookup. Moreover, most of the three or four point methods considered here will sometimes yield spurious negative intensity values (since some of the weights are negative).

Only fast resampling techniques, such as those required for on-board satellite real-time resampling of image data, are considered here. Much better results could be had using standard low-pass filter design techniques with many sample points. If absolute maximum processing speed is required, the resampling weights must be stored in a lookup table (unless the weights are trivially related to the displacement, as in nearest-neighbor or linear interpolation resampling).

All of these techniques (except the one or two point methods) have end effects; that is, the method cannot be extended all the way to the end of a scan line. The alternatives are to overscan (i.e., take data slightly beyond the ends of the line of desired sample points) or to adopt another method at the end of the line. A simpler implementation and greater accuracy are achieved if the data are overscanned by a few pixels.

Nearest Neighbor Resampling

The most primitive technique is to estimate the image value at a desired point as the value at the nearest actual sample point. The operation may be described by a kernel

$$h(x) = \begin{cases} 1 \text{ if } |x| < .5 \\ 0 \text{ otherwise} \end{cases}$$

with the understanding that "ties" (i.e., sample points which are equally displaced from a desired point) are to be resolved in some way. Very serious aliasing problems arise because h is not bandlimited.

This kernel reconstructs the data as a step function. The moiré effects of spurious high spatial frequency components may be detectable to the eye as "blockiness" although the resolution is adequate for some purposes, for example, in resampling highly-oversampled data.

Linear Interpolation

This method reconstructs the data as a piecewise-linear function which is exact at the sample points (in two dimensions, the reconstruction is piecewise-planar).

The kernel associated with this method, which is somewhat closer to being bandlimited than the previous one, is

$$h(x) = \begin{cases} 1-|x| & \text{for } |x| < 1 \\ 0 & \text{otherwise} \end{cases}$$

It gives poor resolution, even for moderately low spatial frequencies, because it does not reconstruct the curvature of the data function within pixel intervals.

Two Point Cubic Interpolation

An improved kernel is obtained by fitting a cubic to two points with the assumptions:

- the sampled data are exact
- the derivative is equal to zero at the sample points

(if one estimates the derivative better, one gets ordinary linear interpolation, which yields poorer results)

The kernel is

$$h(x) = \begin{cases} (1-|x|)^2(1+2|x|) = 1-3|x|^2 + 2|x|^3 & \text{for } |x| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

(the graph begins to resemble that of the sinc function).

The kernel h is almost perfectly bandlimited at a normalized frequency of 1 (better than either nearest neighbor or linear interpolation) and its resolution is roughly the mean of that for these two techniques.

The major disadvantage of this method is the computation or table lookup for the weights.

Quadratic Interpolation

This method is akin to Simpson's rule of numerical integration. As with Simpson's rule, a parabola is fit to three data points with good results. The resolution is very good, but there is significant stop-band ripple, although less than with the nearest neighbor method.

The kernel is discontinuous

$$h(x) = \begin{cases} 1-|x|^2 & \text{for } |x| < .5 \\ (2-|x|)(1-|x|/2 & \text{for } .5 < |x| < 1.5 \end{cases}$$

Here, as in NN, ties must be decided; only one point can be weighted using the $1-|x|^2$ branch of the kernel.

Cubic Interpolation

A cubic polynomial is fit to four data points using the kernel

$$h(x) = \begin{cases} (1-|x|^2)(2-|x|)/2 & \text{for } |x| < 1 \\ (1-|x|)(2-|x|)(3-|x|)/6 & \text{for } 1 < |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

This kernel is continuous and the approximation of the sinc function is quite apparent. Resolution is not as good as that of quadratic interpolation, but there is very little stop-band ripple.

Cubic Osculatory Interpolation

The name "osculatory" is used because a cubic is used to fit the data function and its derivatives at two points. The derivatives are estimated, using the mean value theorem:

$$f_2' = (f_3-f_1)/2$$

$$f_3' = (f_4-f_2)/2$$

where $f_1$, $f_2$, $f_3$, and $f_4$ are the data points, respectively.

The kernel which results in osculatory interpolation of the data at the two middle sample points (i.e., the kernel which fits a cubic to $f_2$, $f_3$, $f_2'$, $f_3'$) is

$$h(x) = \begin{cases} (2-5|x|^2 + 3|x|^3)/2 & \text{for } |x| < 1 \\ 2-4|x|+ 2.5|x|^2-0.5|x|^3 & \text{for } 1 < |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} (1-|x|)(2 + 2|x|-3|x|^2)/2 & \text{for } |x| < 1 \\ (1-|x|)(2-|x|)^2/2 & \text{for } 1 < |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

This kernel is continuous and has a continuous derivative, but its resolution is still not as good as that of quadratic interpolation. Its frequency is virtually free of ripple.

Cubic Spline Interpolation

Because of the minimum-curvature property of cubic spline interpolation, fitting cubic splines to four data points tends to bandlimit the data, thus, the action of the sinc function is approximated.

The kernel is

$$h(x) = \begin{cases} (1-|x|)(1 + .8|x|-|x|^2) & \text{for } |x| < 1 \\ (1-|x|)(2-|x|)(-|x|/3 + 4/5) & \text{for } 1 < |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

The resolution is excellent (better than that of the quadratic kernel) but there is a very small amount of stop-band ripple. The derivative of h is discontinuous at 0 and at 1.

Cubic Spline Approximation of the Sinc Function

Instead of trying to bandlimit the data, as in the previous example, one can try to bandlimit and truncate the sinc function. This method was proposed by TRW and seems to provide good overall performance. The TRW kernel is

$$h(x) = \begin{cases} (1-|x|)(1 + |x| - 1.25|x|^2) & \text{for } |x| < 1 \\ (1-|x|)(2-|x|)^2(0.75) & \text{for } 1 < |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

The kernel and its derivative are continuous.

This kernel sacrifices some performance at very low spatial frequencies to obtain good performance throughout the passband.

TRW has also used another kernel

$$h(x) = \begin{cases} (1-|x|)(1 + |x| - |x|^2) \\ (1-|x|)(2-|x|)^2 \end{cases}$$

which has more ripple and a sharper cutoff (it actually provides a more accurate approximation of the sinc function).

Trigonometric Polynomial Interpolation

A four-point Fourier transform of the four sample values yields a trigonometric polynomial reconstruction of the data which interpolates the data at the four sample points. The data, of course, may have constituent frequencies which are not harmonically related to the sampling frequencies.

The kernel is

$$h(x) = \begin{cases} (1 + \cos\pi x + 2\cos(\pi x/2))/4 & \text{for } |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

This kernel gives results which are surprisingly similar to those of the previous technique, although the approaches are quite different. Its characteristic lies between those of the TRW kernels, although it is much closer to the first TRW kernel.

Periodic Cubic Spline Interpolation

The cubic spline interpolation method works quite well, despite the fact that the data cannot have polynomial-like behavior overall, but only locally. This is another application of the discrete Fourier transform which yields a periodic cubic spline interpolation of the data.

The exact continuous Fourier transform of the periodic cubic spline interpolation is calculated by the use of "attenuation factors," which weight the entire infinite but periodic discrete Fourier transform.

The kernel is

$$h(x) = \frac{1}{N}\left[1 + 6\sum_{m=1}^{\infty} \frac{\operatorname{sinc}^4\left(\frac{m}{N}\right)}{2 + \cos\frac{(2\pi m)}{N}} \cos\frac{2\pi m x}{N}\right]$$

The sum is infinite, but the terms become insignificant rapidly because of the fourth power of the sinc function.

If one uses only the first four terms, with N = 4, one has

$$h(x) \doteq \tfrac{1}{4}\left[1 + \frac{192}{\pi^4}\cos\frac{\pi}{2}x + \frac{96}{\pi^4}\cos\pi x + \frac{192}{8\pi^4}\cos\frac{3\pi}{2}x\right]$$

Note that $\frac{96}{\pi^4} \doteq .9855$ so that this kernel differs very slightly from the trigonometric polynomial kernel. The frequency response is essentially the same as that of the first TRW kernel, although it was obtained in an entirely different manner. The TRW kernel has the advantage that it is easy to compute. However, the periodic cubic spline kernel is, when computed exactly, known to provide exact periodic cubic spline reconstruction of the data. If a lookup table is used, the computation required to determine the values of the kernel becomes insignificant.

This method also can be used with 3 points, with results somewhat similar to the second TRW kernel, except for stopband ripple.

Kaiser Window Truncation of Sinc (x)

An alternative Kaiser window, recently proposed, was used to truncate sinc(x) to the interval [-2,2], thus, the effective kernel is

$$h(x) = \begin{cases} (\sin\pi x/\pi x)\sinh\left(\beta\sqrt{1-(x/2)^2}\right)/(\sinh\beta)\sqrt{1-(x/2)^2} & \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } |x| < 2 & \\ 0 \qquad \text{otherwise} & \end{cases}$$

This kernel is known to be virtually equivalent to the original Kaiser window but a little easier to use since only standard functions are required. A choice of $\beta = 4.8$ gives fairly good performance at very low frequencies.

The results are then very similar to those obtained with cubic spline interpolation of the data.

Hamming Window Truncation of Sinc (x)

This kernel is

$$h(x) = \begin{cases} (.54+.46\cos\frac{\pi x}{2})\sin x/\pi x & \text{for } |x| < 2 \\ 0 \quad \text{otherwise} \end{cases}$$

This kernel has a sharp-cutoff response with significant ripple, somewhat like the alternate TRW kernel, except that this kernel does not resample DC correctly.

There are many possible criteria to be applied to these resampling methods. Two of them will be mentioned here.

1. Minimize the maximum error in the passband and/or in the stop-band (exempting the transition band). This is a standard criterion which often guides computer algorithms for digital filter design.

   This is a relatively convenient criterion to apply for nearly bandlimited kernels.

2. Minimize the maximum relative error of interpolation for a test function or class of test functions. It is not convenient to transport this criterion to the frequency domain because it is relative error, not absolute error, which is being measured.

   One might, for example, convolve the kernel with the sinc function itself and compare the result with the ideal which is also the sinc function.

   There are two additional provisos: the zeros of the sinc function (or other test function) must be exempted from consideration and only a section of the test function can be examined.

Combination Methods

The most obvious way to combine two methods of resampling is by the linear combination of two or more kernels which have been defined or the same domain, for example,

if $h_1$ and $h_2$ are defined on [-2,2] and t is any real number,
then $h_3 = th_1 + (1-t)h_2$ is also a kernel defined on [-2,2].

If $h_1$ and $h_2$ resampled constants exactly, $h_3$ will resample constants exactly. The properties of $h_3$ will be "averages" of the propoerties of $h_1$ and $h_2$.

If the interval [-2,2] is subdivided in such a way that resampling will be done using either one kernel of another, but not both, another sort of combination is achieved. For example,

define

$$h(x) = \begin{cases} 1 & \text{for } |x| \leq .25 \\ 1-|x| & \text{for } .25 < |x| < .75 \end{cases}$$

to combine the nearest neighbor and linear interpolation methods on [-1,1].

The Role of Continuous Fourier Transforms and Digital Filtering in Resampling

If the Fourier transform of a signal f is F and the reconstruction is implemented by use of a convolutional kernel h, that is,

$$\hat{f}(t) = \sum_{n=-\infty}^{\infty} f(n)h(t-n)$$

then the Fourier transform $\hat{F}$ of $\hat{f}$ satisfies the equality

$$\hat{F}(\omega) = H(\omega) \sum_{n=-\infty}^{\infty} F(\omega + 2n\pi)$$

where H is the Fourier transform of h (see [A-4], p. 140).

$$\text{If } H(\omega) = 1 \quad \text{for } |\omega| \leq \pi$$
$$\text{and } H(\omega) = 0 \quad \text{for } |\omega| > \pi$$

then h is the sinc function and properly bandlimited functions are reconstructed exactly. If h is truncated (i.e., "time-limited") then it is not bandlimited, thus no truncated approximation of the sinc function will satisfy this condition exactly.

Ordinarily $H(\omega)$ will be a continuous function of $\omega$. Thus, it cannot approximate the response of the sinc function very well near its discontinuity. Some low-frequency energy will be aliased into the higher frequency region, with the largest contribution arising from passband energy near cutoff $\sigma$ which is heavily aliased into the region above cutoff. Some aliasing also occurs because of the side lobes in the response $H(\omega)$, but this can be quite a bit lower.

Those kernels which approximate the sinc function will tend to have a sharper cutoff with more ripple. Those kernels which interpolate the data by a low degree polynomial tend to have a smoother transition with less ripple (e.g., the 4 point cubic osculatory interpolation is a clear example of this).

However, it has been found that kernels which yield periodic cubic spline interpolation of the data are very similar in response to some of those which are obtained from cubic spline approximation of the sinc function. These methods seem to be intermediate between the polynomial interpolation kernels and the kernels obtained by close approximation of sinc x). They are to be recommended when good performance over the passband is required. However, polynomial interpolation methods are to be preferred for accuracy when most energy is at very low frequencies (e.g., cubic spline interpolation is very accurate near DC and has a very gradual dropoff in accuracy to about $\omega = 0.25$). Fortunately these methods may be combined in linear combination to achieve performance at any intermediate level.

## REFERENCES


1-1. TRW-Defense and Space Systems Group, "On-Board Image Registration Study", Final Report, Contract NAS5-23725, (Draft), January 1979.

2-1. Specifications for GPSPAC Receiver/Processor Assembly, Document 7250-9000, Issue C, November 14, 1978.

2-2. Boeing Aerospace Co., "Space Test Program Standard Satellite Attitude Control and Determination Study - Final Report", SAMCO-TR-76-14, October 1975.

2-3. Rifman, S.S, et al, "Experimental Study of Digital Image Processing Techniques for Landsat Data", Final Report, TRW Systems Group, January 1976 (NASA CR-152552)

2-4 Simon, K. W., "Digital Image Reconstruction and Resampling for Geometric Manipulation", Proc. IEEE Symp. on Machine Processing of Remotely Sensed Data, June 3-5, pp. 3A-1-3A-11.

2-5 Papoulis, Athanasios, "Signal Analysis", McGraw-Hill, 1977.

2-6 Tisdale, G.E., and B. Peavey, "A Digital Scene Matching Technique of Geometric Image Correction and Autonomous Navigation" presented at the Flight Mechanics/Estimation Theory Symposium held at Goddard Space Flight Center, Greenbelt, Md. on October 18 and 19, 1978.

A-1 Schafer, Ronald W. and Lawrence R. Rabiner. "A Digital Signal Processing Approach to Linear Interpolation", Proc. IEEE, Vol. 61, pp. 692-702, June 1973.

A-2. Simon, K.W., "Digital Image Reconstruction and Resampling for Geometric Manipulation", Proc. IEEE Symposium on Machine Processing of Remotely Sensed Data, June 3-5, 1975, pp. 3A-1-3A-11.

A-3. Castleman, Kenneth R., "Digital Image Processing", Prentice-Hall, 1979.

A-4. Papoulis, Athanasios, "Signal Analysis". McGraw-Hill, 1977.

A-5. Gautschi, W., "Attenuation Factors in Practical Fourier Analysis". Numer. Math., 18, pp. 373-400.

A-6. Stewart, R.M., "Statistical Design and Evaluation of Filters for the Restoration of Sampled Data", Proc. IRE, V. 44, Feb. 1956, pp. 253-257.

A-7. Johnson, Lee W. and R. Dean Riess, "Numerical Analysis", Addison-Wesley, 1977.

A-8. Knab, J. J., "An Alternate Kaiser Window", IEEE Trans. Acoustics, Speech and Signal Processing, Vol. 27, October 1979.

| 1. Report No. | 2. Government Accession No. | 3. Recipient's Catalog No. |
|---|---|---|
| NASA CR - 159287 | | |

| 4. Title and Subtitle | 5. Report Date |
|---|---|
| Concepts For On-Board Satellite Image Registration | June 1980 |
| | 6. Performing Organization Code |

| 7. Author(s) | 8. Performing Organization Report No. |
|---|---|
| W. H. Ruedger, D. R. Daluge, and J. V. Aanstoos | RTI/1796/00-01F |
| | 10. Work Unit No. |

| 9. Performing Organization Name and Address | 11. Contract or Grant No. |
|---|---|
| Research Triangle Institute<br>P.O. Box 12194<br>Research Triangle Park, North Carolina 27709 | NAS1-15768 |
| | 13. Type of Report and Period Covered |
| 12. Sponsoring Agency Name and Address | Contractor Report |
| National Aeronautics and Space Administration<br>Langley Research Center<br>Hampton, Virginia 23665 | 14. Sponsoring Agency Code |

15. Supplementary Notes

Contract Monitor: W. L. Kelly IV, NASA Langley Research Center

16. Abstract

The NASA-NEEDS program goals present a requirement for on-board signal processing to achieve user-compatible, information-adaptive data acquisition. One very specific area of interest, which this study addresses, is the pre-processing required to register imaging sensor data which has been distorted by anomalies in subsatellite-point position and/or attitude control. This study brings attention to the concepts and considerations involved in using state-of-the-art positioning systems such as the Global Positioning System (GPS) in concert with state-of-the-art attitude stabilization and/or determination systems to provide the required registration accuracy. Aspects of the study include an examination of the accuracy to which a given image picture-element can be located and identified, the determination of those algorithms required to augment the registration procedure, and consideration of the technology impact on performing these procedures on-board the satellite.

| 17. Key Words (Suggested by Author(s)) | 18. Distribution Statement |
|---|---|
| On-Board Signal Processing<br>Image Registration<br>Landsat<br>Thematic Mapper | Unclassified-Unlimited |

| 19. Security Classif. (of this report) | 20. Security Classif. (of this page) | 21. No. of Pages | 22. Price* |
|---|---|---|---|
| Unclassified | Unclassified | | |



For sale by the National Technical Information Service, Springfield, Virginia 22161