THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

MARSHALL / GRANT

FINAL REPORT

IN-74 CR

NASA/MSFC Grant No. NAG8-587

87611

9/1/86-5/31/87

P. 34

THE FEASIBILITY AND DESIGN OF OPTICAL SENSORS FOR MODAL CONTROL

Charles A. Lundquist
Principal Investigator

Prepared for

George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Marshall Space Flight Center, AL 35812

Submitted by

The University of Alabama in Huntsville

July 17, 1987

(NASA-CR-181184) THE FEASIBILITY AND DESIGN OF OPTICAL SENSORS FOR MODAL CONTROL Final Report, 1 Sep. 1986 - 31 May 1987 (Alabama Univ.) 34 p Avail: NTIS HC A03/MF A01 CSCL 20F G3/74

N87-27499

Unclas
0087611

A FINAL REPORT

TO

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MARSHALL SPACE FLIGHT CENTER

ON

FEASIBILITY ASSESSMENT OF OPTICAL SENSORS

FOR

MODAL CONTROL

BY

DR. MICHAEL GREENE
ASSOCIATE PROFESSOR
DEPARTMENT OF ELECTRICAL ENGINEERING
AUBURN UNIVERSITY, AL 36849

JULY 9, 1987

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. INTRODUCTION

The concept of the Pinhole-Occulter Facility has been studied extensively (1-5) and the control system used to point and stabilize it has also been well defined and studied (4-8). Both the scientific and engineering analyses have indicated the need for highly accurate and stable sensors to be used in both the pointing and control of the structure and in the analyses of the scientific data (4,6). The proposed sensors have been designated as a fine line of sight (LOS) sensor and the modal control sensor (MCS) (4,5).

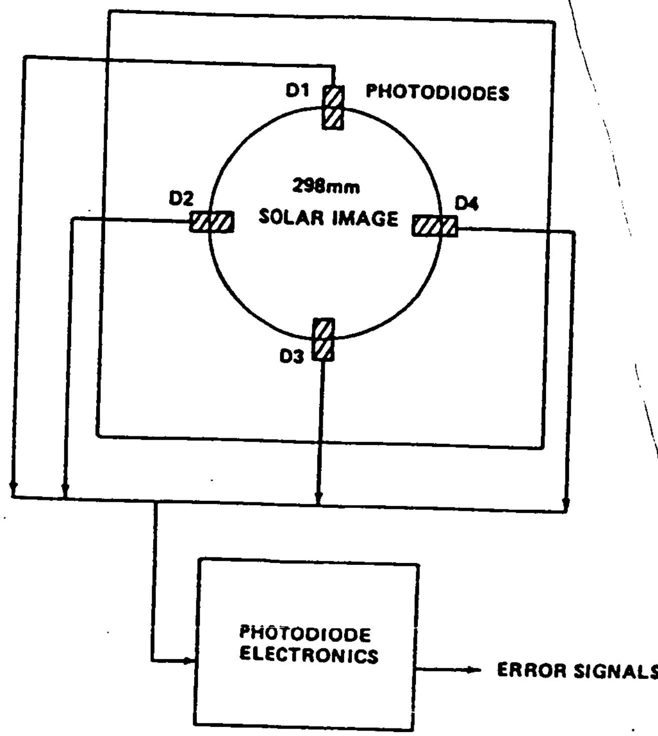
The fine line of sight (LOS) sensor has been studied in the Pinhole Occulter Facility (POF) Phase A study (5). The basic configuration is a 5mm pinhole in the mask of POF which casts a solar image on the detector plane which is 105 ft. away. This would yield a solar image of 11.73 in. at the detector. A montage of four photodetectors would then sense this image and yield pointing errors. This basic configuration of the sensors is shown in figure 1. This basic LOS sensor has been built and studied in the lab by NASA, MSFC. A heliostat projected the solar image onto a mask with a 5mm pinhole and a series of mirrors were used to extend the path length of the light to 105 ft. Four and eight pinhole LOS sensors were also proposed but not actually built or tested in the Phase A study.

A number of sensors for the internal alignment of flexible bodies have been proposed (5,9) but have not been designed with any detail or analyzed in any systematic way. None of the sensors are existent in hardware or software or have they been prototyped. None have been tested. Research in this area appears to be at a minimum even though much work is needed for future systems such as P/OF.

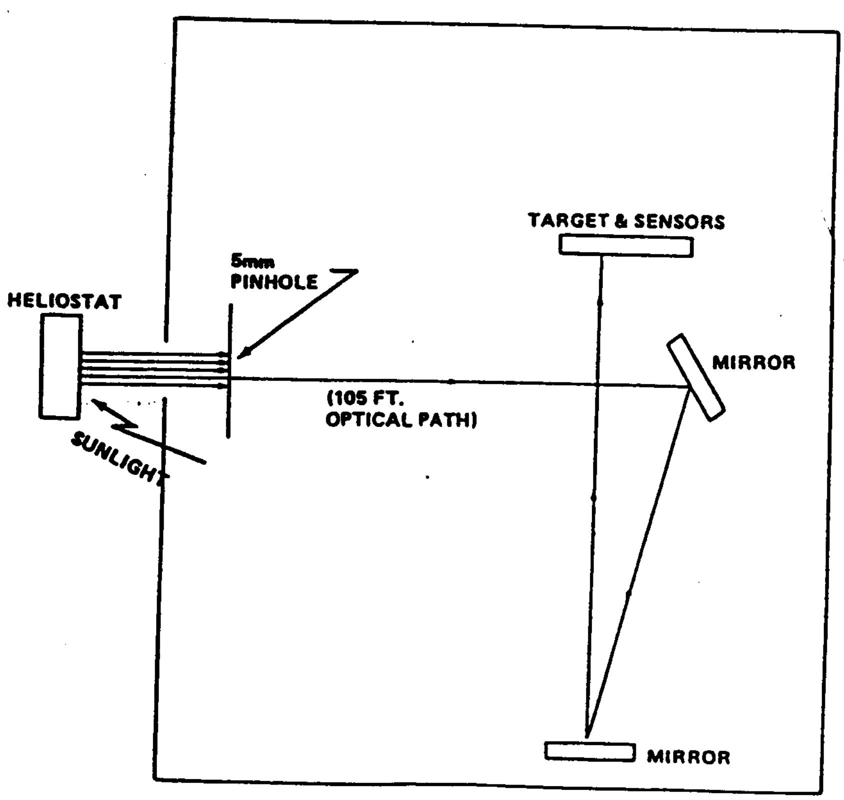
The purpose of this report is to access the feasibility of optical type sensors for control of flexible bodies. The accuracies of such systems were determined via simulation and the sources of potential errors were designated. An initial laboratory design was effected and preliminary results obtained. These results are discussed critically with applications to future studies and system designs.

A number of errors exist in any measurement system. The chief errors occur due to noise, bias, quantization and variations in scale factor. For the proposed system, the error sources were analyzed and an error model developed.

SINGLE PINHOLE
TARGET & SENSORS



ALIGNMENT
TEST SETUP FOR
A SINGLE PINHOLE



II. BACKGROUND

The Pinhole/Occulter Facility is designed to enhance the studies of solar flares, the solar corona, and cosmic X-ray sources. The POF consists of a continuous longeron astromast which connects an occulting mask to a detector plane. The entire assembly is located at the bay of the space shuttle and mounted on a three axes gimbal pointing system as seen in Figure 2.

During launch and landing this boom is stored in a canister 6.42 meters in length [5]. When fully deployed the boom is 32 meters in length with a diameter of .3556 m. With the occulter mounted at the tip of the boom the tip mass is 55 lbs. Since the tip mass is negligible compared to that of the shuttle the boom may be modeled as a fixed/free flexible beam problem. Approaching the problem in this manner it has been determined, from NASTRAN simulation run, that the mast has modes as shown in Table 1 [5].

The candidate modal control sensor (MCS) was proposed (5,9) first by Dr. Frank van Beek and was basically the system shown in figure 3. Here laser diode light sources are used to generate two beams which are reflected off the back side of the POF mask. Two such beams would be used: one would be reflected off a spherical mirror yielding both tilt and position information while the other beam would be reflected off a flat mirror yielding only tilt information.

The MCS provides information to the active modal controller (5) on both the position of the boom tip of P/OF and its rotation relative to the detector plane. The sensors are constructed of laser light diode sources and diode array detectors both at the detector plane. Mirrors on the back of the mask, which is 32m from the detector plane, reflect the light from the sources to the detectors. The sensors used with the curved mirror would provide positional information combined with tilt. The sensors used with the flat mirror would provide only tilt information. Positional information would be obtained analytically from these two measures.

Positional information on both X and Y translations can be obtained using two curved mirrors with detectors for both X and Y from the tilt + position

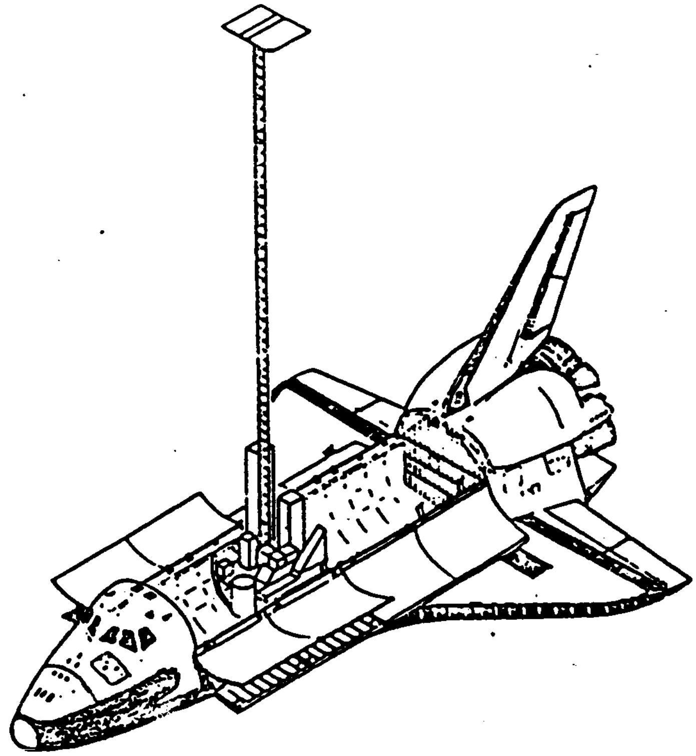
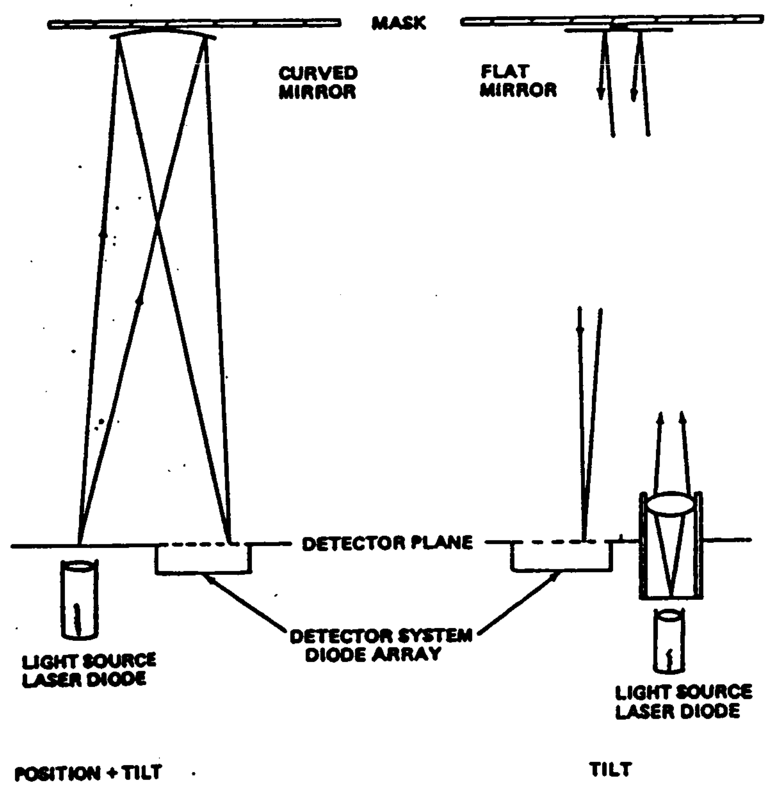


Figure 2

Pinhole Occulter Facility (POF)

Figure 3

POF ALIGNMENT SENSOR FOR ACTIVE MODAL CONTROL



detector. Alternately, a dished mirror could be used in lieu of the two mirrors along with an array detector system for obtaining the same data. A system for measuring the modal deflection about the Z axis (boom roll) has not been presented to date.

MODE NUMBER	FREQUENCY (Hz)
1	.064
2	.064
3	.355
4	.751
5	.751
6	2.361
7	2.361
8	4.872
9	4.872

TABLE 1 BOOM FREQUENCIES

The optical sensor system must be capable of providing deflection information corresponding to the first four modes, at .064, .355, and .751 Hz. This information is used in a feedback control system (5,7) which actively damps the vibrations of the beam. The enhanced stability of the system with the controller provides significant resolution enhancement (5).

III. SYSTEM DESCRIPTION

The basic components of the instrument system for each axes include a laser, lenses, mirror, and photodetector array. The laser and photodetector array are to be mounted at the detector plane while the mirror will be mounted on the underside of the mask. The lenses will be placed between the laser and the mirror to columnate and focus the beam onto the photodetector array. It is desired that none of the optics be placed between the mirror and the photodetector array since the position of the beam will vary due to deflections and rotations of the mask. The basic system schematic is shown in Figure 4.

For laboratory work the photodetector is a linear array with 256 x 1 pixels. Each has a separation of 25 micro meters (um). For the POF, a longer detector with more pixels or several staggered detectors will be required. For the purpose of this study the response of the detector was assumed to be linear. However, future work may need to investigate the effects of pixel response nonuniformity (10).

The general scheme is that the laser will be bore sighted to reflect off the mirror mounted to the underside of the mask and illuminate the detector mounted back at the detector plane. A beam splitter will allow a single laser to be used to illuminate detectors for both the pitch and yaw directions.

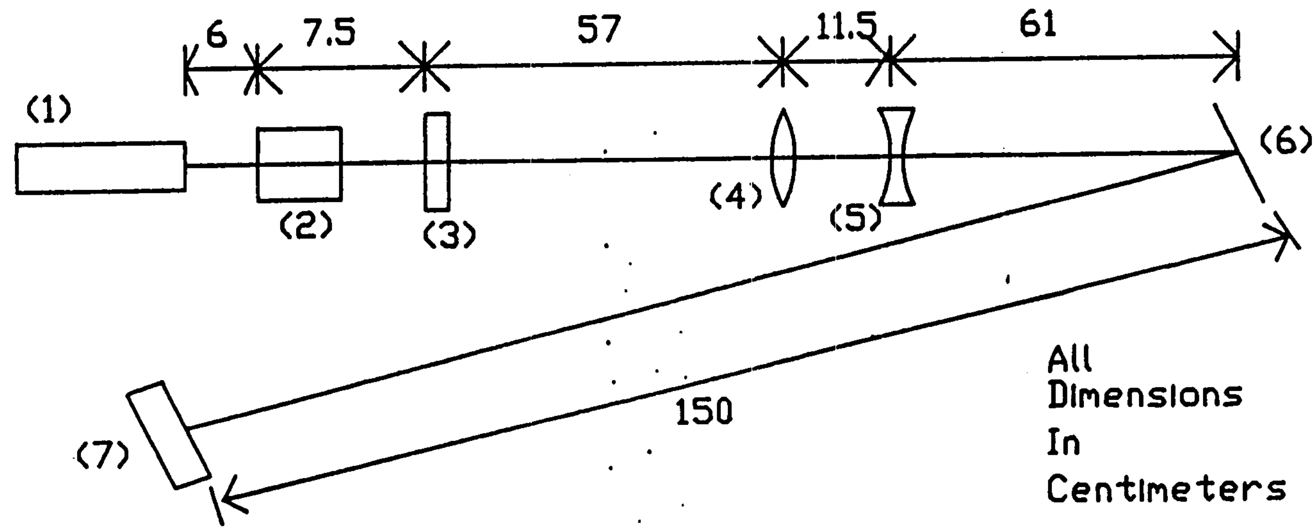
It is desired to detect micrometer deflections and sub-arcsecond rotations of the mask with respect to the detector plane. In order to achieve results consistent with these requirements changes in position of the beam illuminating the detector, due to a disturbance, must be resolved to sub-pixel accuracy. Specifically, position estimation to less than 1/10 of a pixel is desired.

The response of the photodetector is proportional to the intensity of the illuminating source. Assuming the beam to be gaussian the peak response will come from the pixel where the center of the beam is located. It is thus necessary to keep track of the center of the gaussian beam as it moves across the detector.

Figure 4

Laboratory Setup

(1) HeNe Laser (5mW), dia = 1mm, Melles Griot - 05-lhp-151
(2) Microscope Objective, 10x with 10 micrometer pinhole
(3) Round Variable Beamsplitter
(4) Bi-convex lens, f=178 mm (lenses are mounted on a magnetic base)
(5) Bi-concave lens, f=144 mm
(6) Flat Mirror (micro-adjustable)
(7) Linear photodetector, EG&G Reticon 256x1, 25 micrometer pixels mounted on a micro-adjustabe xyz platform



In general, any wave with a gaussian transverse amplitude distribution may be written as,

$$|u(x,y)| = e^{-(x^2+y^2)/W^2}. \qquad (1)$$

For this application it is for more efficient to use two linear arrays, one for the x-axis and one for the y-axis, rather than one area array. Two linear arrays 256 x 1 require the manipulation of only 512 responses compared to an area array 256 x 256 which would require scanning 65,536 pixels.

Since this project is concerned with linear arrays only one axis need be considered at a time. The general equation may then be altered to reflect the difference between a known mean $\bar{x}$ and the pixel response x_i:

$$|u(x)| = e^{-(x_i^2-\bar{x}^2)/W^2}. \qquad (2)$$

From the general formula given in Eq. 2 the intensity of the response of each pixel may be calculated relative to the distance from the center of the beam. Both algorithms will use the formula given in Eq. 2 to compute pixel response.

Since each pixel can only give one response regardless of where the light on it, the array has the effect of discretizing the gaussian beam. The response of each pixel will be taken from the center of the pixel, the position of which will be referred to as x_i, where the subscript i ranges from 1 to 256. The actual position of the mean of the gaussian beam, on the array, will be referred to as $\bar{x}$.

IV. BEAM CENTROID ESTIMATION

Two methods of beam centroid estimation were developed and simulated. The first method, Three Point Centroid, is lower in computational overhead than the second method; Probability Density Centroid. Each method is presented and discussed.

A. THREE POINT CENTROID

A.1. METHOD

The three point centroid algorithm relies on the response from the three most highly illuminated pixels to estimate the location of the mean, $\tilde{x}$. Where $\tilde{x}$ is an estimate of $\bar{x}$. Four possible cases exist for the location of $\bar{x}$ with respect to x_i, x_{i-1}, and x_{i+1}, where x_i is the location of the greatest response, x_{i-1} is the location of the pixel one to the left, and x_{i+1} in the location of the pixel one to the right. the responses at these positions will be referred to as y_i, y_{i-1}, and y_{i+1} respectively. The four cases are:

1. x is exactly between x_{i-1} and x_i (see Figure 5), then
   $\tilde{x} = (x_{i-1} + x_1)/2$;

2. $\bar{x}$ is between the left edge of pixel x_i

   and the center of pixel x_i (see Figure 6), then
   $\tilde{x} = x_i - (1 - y_i / y_2)\ x$

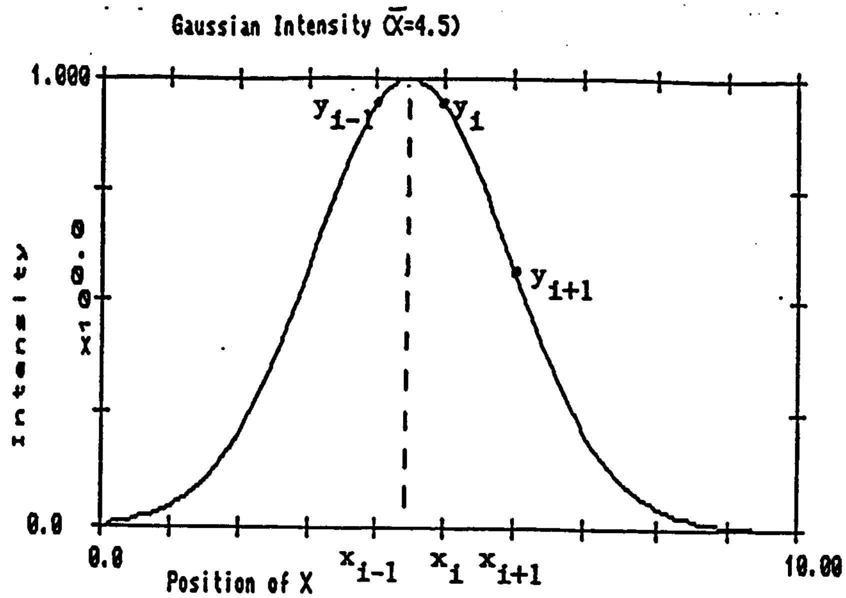
   where $y_1 = y_i - y_{i-1}$

   $y_2 = y_i - y_{i+1}$ and,
   x is the pixel width;

3. $\bar{x}$ is equal to x_i (see Figure 7), then
   $\tilde{x} = x_i$; and

4. $\bar{x}$ is between the center of pixel x_i

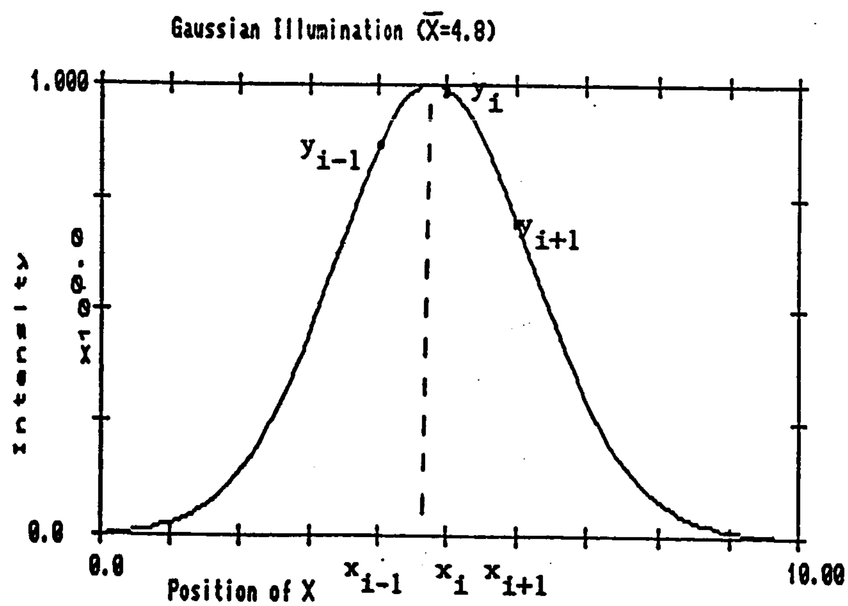
   and the right edge (see Figure 8), then
   $\tilde{x} = x_i + (1 - y_2 / y_1)\ x$.



$y_{i-1} = y_i > y_{i+1}$

$\tilde{x} = (x_{i-1} + x_i) / 2$

Figure 5



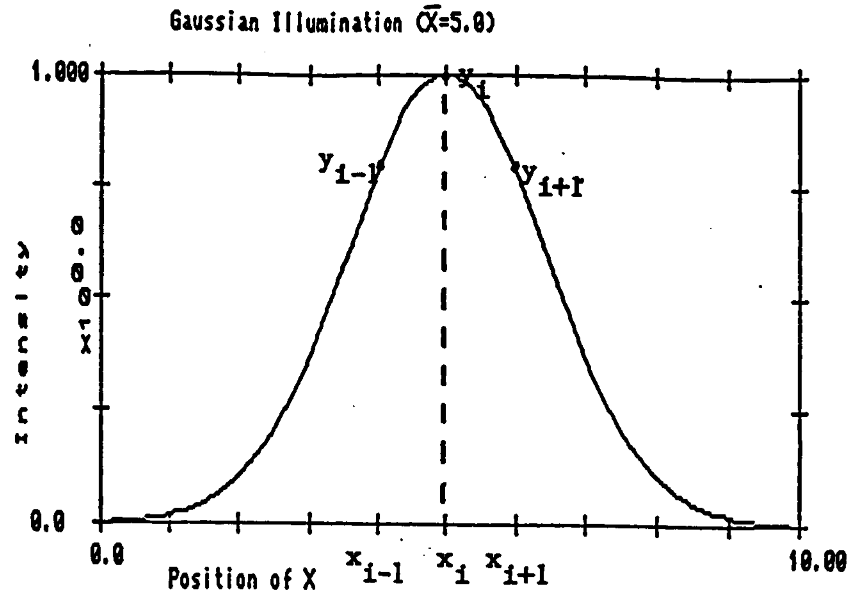
$y_i > y_{i-1} > y_{i+1}$

$\tilde{x} = x_i - (1 - \Delta y_1/\Delta y_2)\,\Delta x \alpha$

$\Delta y_1 = y_{i+1} - y_{i-1}$

$\Delta y_2 = y_i - y_{i+1}$

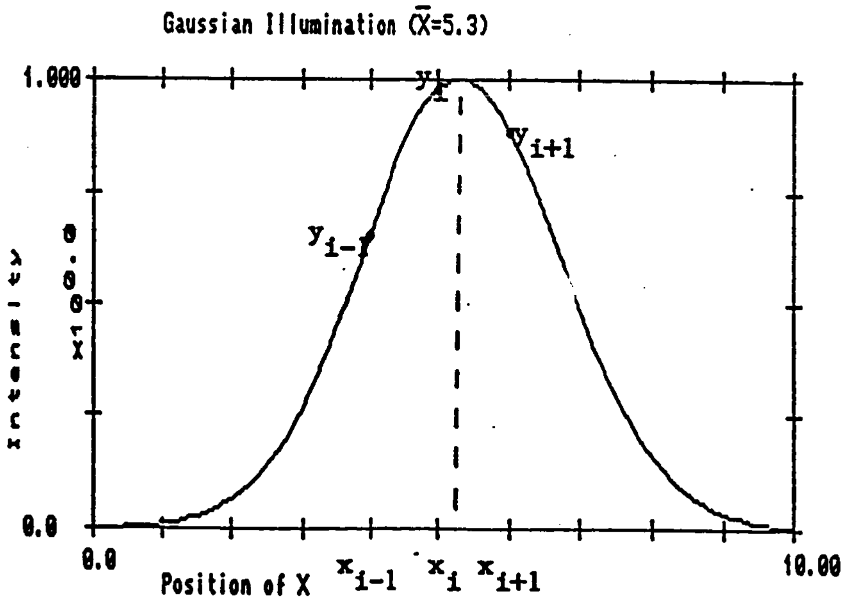
Figure 6



$y_i > y_{i-1} = y_{i+1}$

$\tilde{x} = x_i$

Figure 7



$y_i > y_{i+1} > y_{i-1}$

$\tilde{x} = x_i + (1 - \Delta y_2 / \Delta y_1) \Delta x \alpha$

$\Delta y_1 = y_{i+1} - y_{i-1}$

$\Delta y_2 = y_i - y_{i+1}$

Figure 8

The algorithm uses a linear fit between the response at y_i and y_{i-1} as well as between y_i and y_{i+1}. Since the pixels are seperated by 25um. this does not introduce much error, however, future work will investigate the possibility of using a higher order fit between these points.

A.2. SIMULATION RESULTS

The algorithms presented in section IV. A.1. were simulated on an IBM-PC in FORTRAN. The computer program is given in Appendix A. The actual beam controid, $\bar{x}$, was varied from one edge of an individual pixel to the other edge and the estimated centroid, $\hat{x}$, was calculated the error, $\bar{x}$-$\hat{x}$, was then calculated as a function of actual centroid location, $\bar{x}$. Such calculations were performed at 1W beam widths of 20, 25, 30, 35, 40 and 45um.

Results for the three point centroid algorithm may be seen in Fig. 9. The figure shows six error curves where the vertical axis represents the error between the estimated mean $\hat{x}$ and the actual mean $\bar{x}$ as it is varied across one pixel. Each curve represents a different spot size from 20 to 45 um. in 5 um. increments. The error is minimized when the spot size is 25 um. or when 68% of the intensity is focused on two pixels. For a spot size of 25 um., this results in pointing accuracy to 1/100 of a pixel width.

These results were obtained without any measurement errors introduced into the system. Actual devices have bias and nonlinearity errors of up to $\pm$ 7% between pixels (10). These errors will be incorporated into a more complete model during later work. The errors shown in Figure 9 are, therefore, quantification errors.

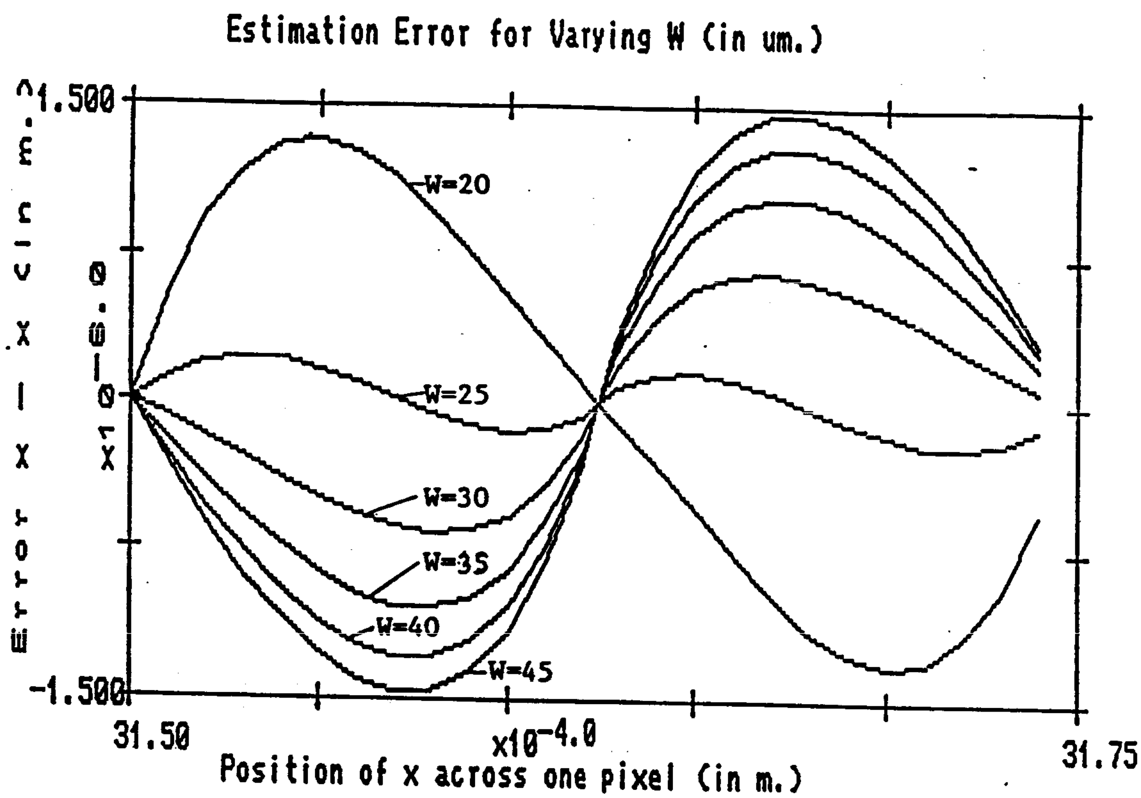

Estimation Error for Varying W (in um.)
1.500
-1.500
0
Error x^ - x (in m.)
x10-6.0
W=20
W=25
W=30
W=35
W=40
W=45
31.50
x10-4.0
31.75
Position of x across one pixel (in m.)

Figure 9

B. PROBABILITY DENSITY CENTROID

B.1. METHOD

The second algorithm uses probability theory to estimate the mean. As mentioned previously the array has the effect of discretizing the guassian beam, hence the mean, $\bar{x}$ may be represented by a discrete random variable. With this in mind the estimate $\tilde{x}$ of $\bar{x}$ may be computed as:

$$\tilde{x} = \sum_{j=1}^{n} x_j P_j \qquad (3)$$

where P_j represents the probability density function of the jth pixel and x_j the position of the jth pixel.

$$P(x_j) = \frac{y_j}{\sum_{i=1}^{n} y} \qquad (4)$$

where y_j is the response of the jth pixel.

Obviously, as with any probabilistic calculation, the accuracy of the estimate improves with the number of samples taken. The error is also very dependent on the spot size, W. As the spot size increases it is necessary to sample more pixels in order to get the same accuracy.

The initial goal has been to compute the RMS error between the estimated $\tilde{x}$ and $\bar{x}$ as $\bar{x}$ is moved across a pixel. The RMS error may be computed by:

$$RMS = \frac{\sqrt{\sum_{i=1}^{n} (i - \bar{x})^2}}{} \qquad (5)$$

where, n is the number of samples.

B.2. SIMULATION RESULTS

The probability algorithm of equation 3-5 was programmed in FORTRAN on an IBM-PC with numeric co-processor. The program is listed on Appendix B. Once again, the actual beam centroid, $\bar{x}$, was varied from one edge of one pixel to the opposite edge. The estimated centroid, $\hat{x}$, calculated as a function of beam centroid location as well as estimation error, $\bar{x}$ - $\hat{x}$. These calculations were performed at the beam widths shown.

Results for the probability density algorithm are shown in Figure (10). This plot is equivalent to Fig. 9 in the data represented. For this set of plots the number of pixels sampled has been fixed at seven. As W increases, the algorithm error also increases since the number of pixels being sampled is not increasing. For the figure shown, the minimum error occurs when the spot size is between 30 and 35um.

Fig. 11 shows a plot of the RMS error versus the number of pixels sampled for the probability density technique. The family of curves differ by the spot size W. As would be expected, as the number of pixels sampled increases the RMS error decreases. This figure also shows how larger the spot sizes need more pixels to be sampled to achieve the same error.

Once again, non-linearities and biases were not considered in the calculation for Figure 10 and 11. Figure 11 gives the quantization errors as a function of number of pixels sampled and beam width. Bias and non-linearity error will be incorporated into a more complete model during later work.

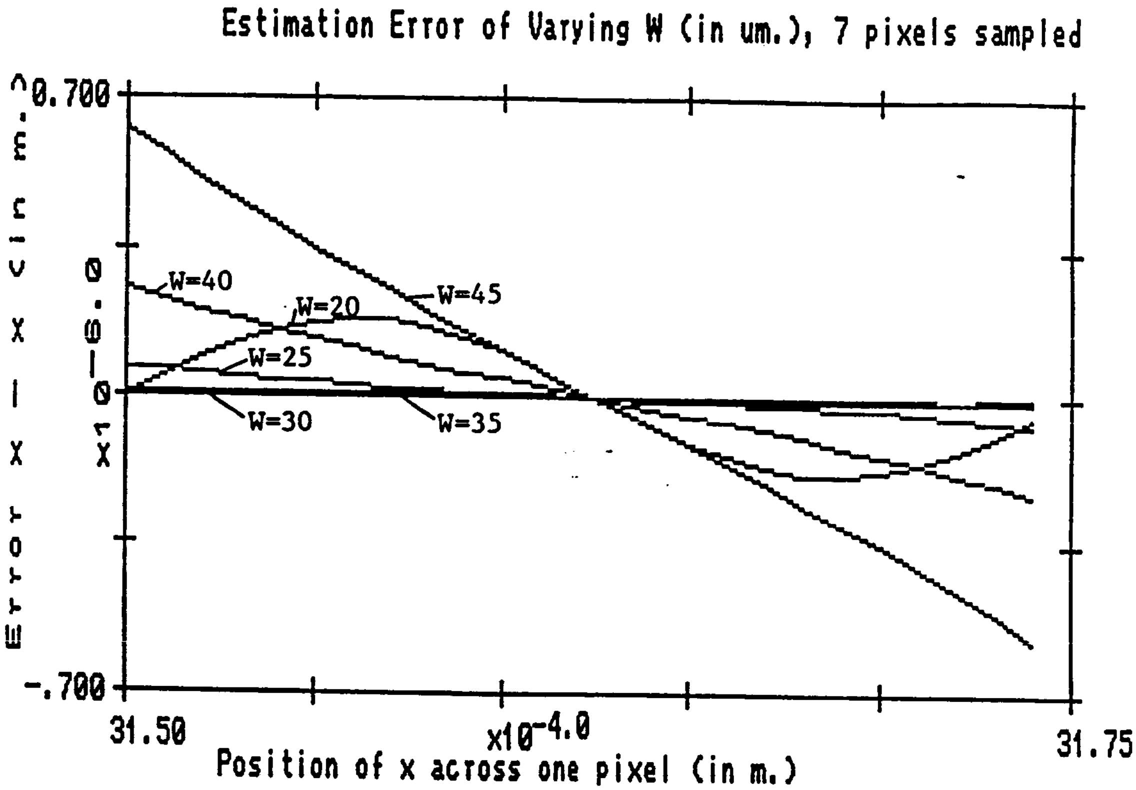
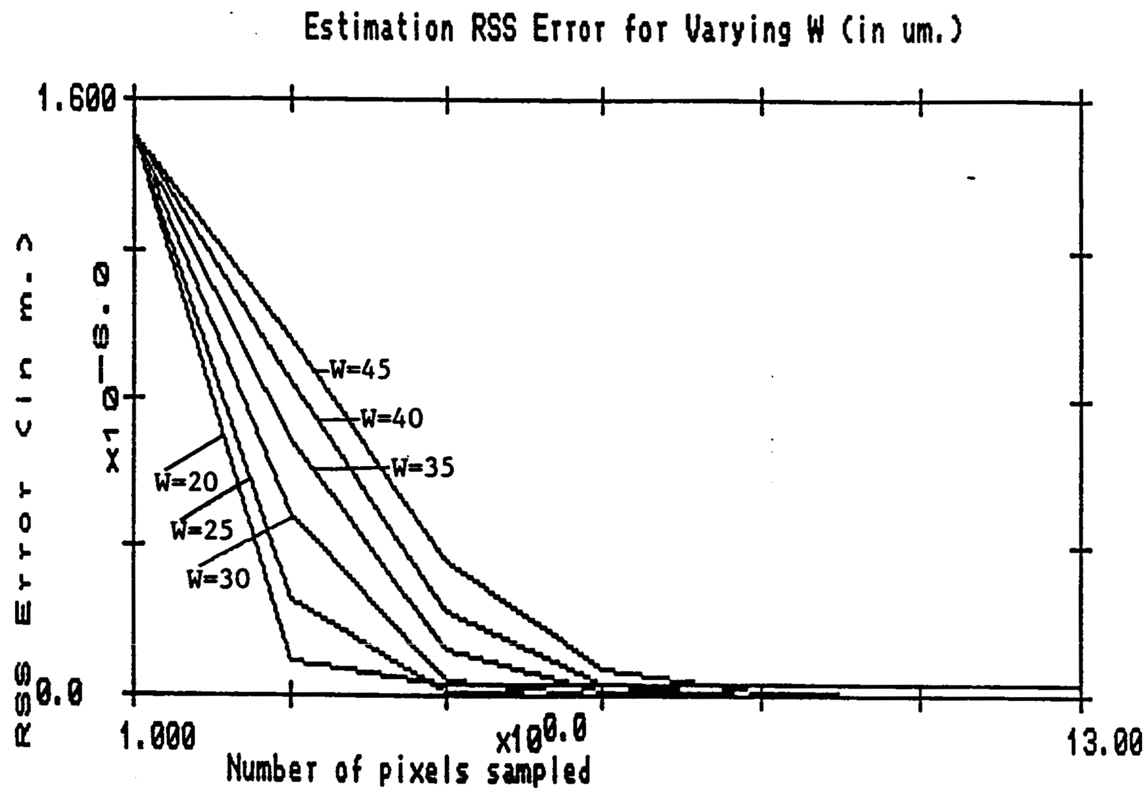

Estimation Error of Varying W (in um.), 7 pixels sampled
0.700
-.700
Error x^ - x (in m.)
x1 0-6.0
W=40
W=20
W=45
W=25
W=30
W=35
31.50
x10-4.0
31.75
Position of x across one pixel (in m.)

Figure 10



Estimation RSS Error for Varying W (in um.)
1.600
0.0
RSS Error (in m.)
X10^-6.0
W=45
W=40
W=35
W=20
W=25
W=30
1.000
x10^0.0
13.00
Number of pixels sampled

Figure 11

C. ESTIMATION DISCUSSION

This section (III) has presented the concept of applying optical sensors for modal control. Consistent with the results contained in other references both algorithms are capable of producing positional estimates to 1/10 of a pixel (11,12). However, the simplicity of the three point centroid makes it more favorable for implementation. The three point centroid requires at most three multiplications and three additions per estimate. Hence, it is the faster of the two algorithms.

Future work for modeling the detector will include compensating the pixel response nonuniformities and biases. These nonuniformities may be measured in the lab for any particular pixel array. Future work will also include noise modeling in the detectors and estimation system.

V. LABORATORY DESIGN AND PRELIMINARY RESULTS

A 1/20 scale model was developed in the lab for the MCS. A schematic of this system is shown in Figure 4. A total pathlength of 3m was demonstrated in UAH's Department of Electrical and Computer Engineering Optics Lab. Photographs of the system have been supplied to the Contracting Officers Technical Representative at MSFC.

For optimal estimation of the centroid using the three centroid algorithm, a beam width of 25um is necessary. In the lab, the smallest beamwidth obtained was 75 um due to the availability of precision optics. At the pixel center and $\pm$ 10 um., the accuracy was $\pm$ 1 um. At $\pm$ 5 um from center, the accuracy was $\pm$ 2 um. These accuracies are due to the gradations of the adjusting micrometer on the pixel array. In the lab setup, the mirror was fixed and the detector array adjusted. Readings were taken at 0, $\pm$ 5 um. and $\pm$ 10 um.

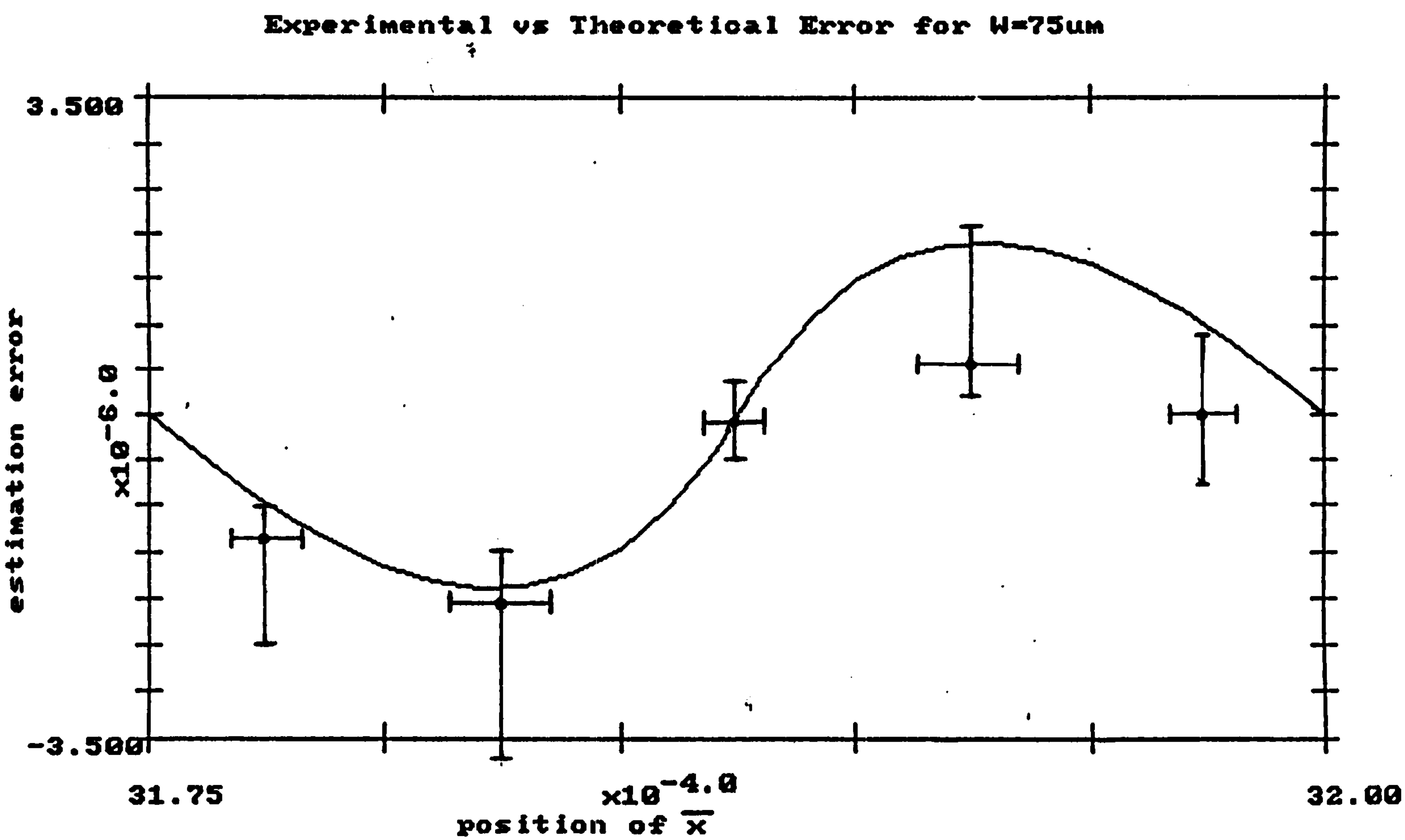
The data are presented in Figure 12. This figure compares the experimental results versus the theoretical (simulation) results using the three point centroid technique. The horizontal variations are due to the micrometer accuracy while the vertical variations are due to noise, bias and non-linearities in the system. The system is drastically affected by air motion induced by sound vibrations and temperature flucuations in the lab. Variations in stray light also influenced the readings. It is interesting to note, however, that the two sets of data agree to a large extent and follow the same general trends. The lab data are repeated in Table 2 and clearly show that except at $\bar{x}$ = -5 um. and -10 um. the position of the centroid can be estimated to within 1/10 of a pixel.

TABLE 2

Lab Data in um

$\bar{x}$	$\bar{x}$ min	$\bar{x}$	$\bar{x}$ max
-5	-8.824	-7.8125	-6.6666
-10	-12.5	-11.875	-11.18
+5	2.88	4.46	5.83
+10	9.21	10.00	10.714

Figure 12



Experimental vs Theoretical Error for W=75um
3.500
-3.500
x10^-6.0
estimation error
31.75
32.00
x10^-4.0
position of x

VI. DISCUSSION AND CONCLUSIONS

Figure 12 and Table 2 clearly demonstrate the feasibility of the proposed modal control sensors. Without any corrections for bias or non-linearities, the lab system responded to nearly the required accuracy. With bias and non-linearity corrections, the system could easily respond to the required degree of accuracy. With noise reduction techniques such a monochromatic filtering at the pixels, the stray light problem also could be minimized. Using precision optics along with corrections and noise reduction on accuracy of 1/20 a pixel could easily be obtained.

The weak link in the system is the optics. Long focal length lenses of quality are expensive and difficult to obtain. In addition, if the optical beam is off axis, aberations are created and the beam is no longer Gaussian. An alternate focusing scheme needs to be used. Current investigations are centered on using linear zone plates. Zone plates do not require critical alignment and their manufacture is easier than long focal length lenses. Their use in a full scale system is more likely, therefore.

VII. BIBLIOGRAPHY

1. Pinhole Occulter Facility Science Working Group Meeting (P/OFSWG) V, Sept. 21-22, 1982, MSFC.

2. Pinhole Occulter Facility Science Working Group Meeting (P/OFSWG) VI., Sept. 7-8, 1983, MSFC.

3. Studies with the Pinhole/Occulter Facility, Tandberg-Hanssen et al, AIAA 21st Aerospace Sciences Meeting, 1983.

4. Pinhole/Coronograph Pointing Control System Integration and Noise Reduction Analysis, Final Report NASA contract #NAS8-34529, 1981, Greene, M.

5. Pinhole Occulter Facility Phase A study report, MSFC, 1984.

6. On the Modeling and Control Of The Pinhole-Occulter Facility, Proc. 15th SSST, p. 56, 1983, Greene, M.

7. Active Modal Damping of The Pinhole/Occulter Facility, COMP. Mech., Adv. Engr. Software, I: 88-90, Greene, M.

8. Robustness of Active Modal Damping Of Large Space Structures, INT. J. Control, in print, 1987.

9. Personal Communication P/OFSWG Sept. 8, 1983, Dr. Frank Van Bech.

10. Optical Sensor for Modal Control, Proc. 19th SSST, p. 179, 1987, Greene, M. and Carter, J.

11. Evaluation of peak location algorithms with sub pixel accuracy for mosaic focal planes, Proc. SPIE, 292, p. 288, 1981, Cox, J.A.

12. Performance Analysis and size optimization of focal planes for point--source tracking algorithm applications, Optical Engr.; 23, p. 167, 1984, Grossman, S. and Emmons, R.

VIII. APPENDICES

A. Appendix A: Three Point Centroid Program

```
$NODEBUG
***********************************************************************
*                                                                     *
*     3PTCEN.FOR          V1.1                                        *
*                                                                     *
*     Author: Jack Carter Jr.                                         *
*     The following program is designed to find the mean of a         *
*     Gaussian waveform which is illuminating a linear photo-         *
*     detector array using a three-point centroid algorithm.          *
*                                                                     *
*     Definition of variables:                                        *
*     W  = Beam Spot Size                                             *
*     YI = Intensity at the given mean of the light source            *
*     YIP1 = Intensity one pixel width to the right                   *
*     YIM1 = Intensity one pixel width to the left                    *
*     XI = The given posistion of the mean of the light source        *
*     XIP1 = The position one pixel width to the right                *
*     XIM1 = The position one pixel width to the left                 *
*     DELX = Pixel width (25um)                                       *
*     DELY1 = (YI - YIM1)                                             *
*     DELY2 = (YI - YIP1)                                             *
*     ALPHA = Constant for tuning results                             *
*     XBAR = Input position of the mean of the light source           *
*     ERR = The error between the estimated mean and XBAR             *
*     MEAN = The estimated mean position                              *
*                                                                     *
***********************************************************************
      IMPLICIT REAL(A-H,O-Z)
      REAL MEAN
*
*  Initialize values for linear array dimensions.
*
      DELX = .000025
      ALPHA = .5
*
*  Open file to store plot data.
*
       OPEN (3,FILE='ERROR.DAT',STATUS='NEW')
*
*  Input the beam spot size W and the pixel number of the
*  location of the mean.
*
       WRITE(*,*) 'INPUT THE PIXEL NUMBER FOR THE LOCATION '
       WRITE(*,*) 'OF THE MEAN. (BETWEEN 1 AND 256)'
       READ(*,*) J
       XI = DELX * (J - .5)
       XIM1 = XI - DELX
       XIP1 = XI + DELX
*
*  Set up a loop to allow W to vary, thus generating a plot file
*  which will have several curves of ERROR vs XBAR with the
*  parameter W
*
```

```
      W = .000020
      DO 200 J=1,6
*
*      WRITE(*,*) 'INPUT W. '
*      READ(*,*) W
*
*  Initialize XBAR to the beginning edge of the pixel so that
*  it may be varied across the pixel and the mean estimated.
*
      XBAR = XI - (DELX/2)
*
*  Vary XBAR across the width of one pixel, in 1um steps
*  (25um) and plot the error (MEAN - XBAR) vs XBAR.
*
      DO 100 I=1,25
*
*  Now evaluate the Gaussian function Y at the three positions of X.
*
     YIM1 = Y(XIM1,XBAR,W)
     YI = Y(XI,XBAR,W)
     YIP1 = Y(XIP1,XBAR,W)
*
*  Evaluate the changes in YIM1, YI and YIP1.
*
     DELY1 = YI - YIM1
     DELY2 = YI - YIP1
*
*  Now that the function has been evaluated there are four cases
*  which must be considered in order to find the actual position
*  of the mean of the light source.
*            (1) YI = YIM1 > YIP1
*            (2) YIP1 = YIM1 < YI
*            (3) YIM1 < YIP1 < YI
*            (4) YIM1 > YIP1 < YI
*
      IF(YI.EQ.YIM1 .AND. YIM1.GT.YIP1) GO TO 10
      IF(YIP1.EQ.YIM1 .AND. YIM1.LT.YI) GO TO 20
      IF(YIM1.LT.YIP1 .AND. YIP1.LT.YI) GO TO 30
      IF(YIM1.GT.YIP1 .AND. YIP1.LT.YI) GO TO 40
*
 10   MEAN = (XI + XIM1)/2
      GO TO 50
*
 20   MEAN = XI
      GO TO 50
*
 30   MEAN = XI + DELX * (1 - (DELY2/DELY1)) * ALPHA
      GO TO 50
*
 40   MEAN = XI - DELX * (1 - (DELY1/DELY2)) * ALPHA
      GO TO 50
*
*  Write the output to the file.
```

```
*
 50     ERR = MEAN - XBAR
        RSSI = RSSI + (ERR * ERR)
*
        WRITE(3,800) XBAR,ERR
 800    FORMAT(2E15.7)
        XBAR = XBAR + .000001
 100    CONTINUE
*
*  Increment W
*
        W = W + .000005
 200    CONTINUE

        RSS = SQRT(RSSI)/I
        WRITE(*,*) 'THE RSS ERROR IS ',RSS
*
        CLOSE(3)
        STOP
        END
**********************************************************************
*                                                                    *
*      FUNCTION Y         V1.1                                       *
*                                                                    *
*      Author: Jack Carter Jr.                                       *
*      date: 8/9/86                                                  *
*                                                                    *
*      The following function evaluates the Gaussian wave front for  *
*      given values of X, XBAR and W.                                *
*                                                                    *
*      Definition of variables:                                      *
*      NUM = -((X - XBAR)**2)  the numerator of the function         *
*      DEN = W * W  the denominator of the function                  *
*                                                                    *
**********************************************************************
      REAL FUNCTION Y(X,XBAR,W)
      REAL NUM, DEN
      REAL X, XBAR, W
*
      NUM = (X-XBAR)**2
      DEN = W * W
      Y = EXP(-(NUM/DEN))
*
      RETURN
      END
```

B. Appendix B: Probability Density Centroid Program

```
$DEBUG
***********************************************************************
*                                                                     *
*                                                                     *
*     PROBW.FOR           V1.1                                        *
*                                                                     *
*     Author: Jack Carter Jr.                                         *
*     The following program is designed to find the mean              *
*     of a gaussian wavefront  using the probability                  *
*     density function.  The output will consist of a plot            *
*     file which will allow the user to plot the estimated            *
*     mean minus the actual vs the actual mean as it moves            *
*     across one pixel.  The output will be a family of               *
*     curves which vary with the spot size W.                         *
*                                                                     *
*     Definition of variables:                                        *
*     W = Beam Spot Size                                              *
*     Y(I) = Array to store intensity levels                          *
*     XI = Incremental value of X (incremented across                 *
*          the photodetector)                                         *
*     XBAR = Input, Desired mean                                      *
*     TEMP = Temporary variable to store intermediate                 *
*            values.                                                  *
*     DELX = Width of the photodetector (in micrometers)              *
*                                                                     *
***********************************************************************

      IMPLICIT REAL (A-H,O-Z)
      REAL MEAN, RL
      INTEGER N1, N2, J, K, L, ITEMP
      REAL YI(26)
*
*    Note: all dimensions are in um.
*

      DELX = .000025
*
*  Input the position of the centroid and the number of pixels
*  to be sampled.
*
      WRITE(*,*) 'INPUT THE POSITION OF THE MEAN '
      WRITE(*,*) 'BETWEEN 1 AND 256.'
      READ(*,*) N
      WRITE(*,*) 'INPUT THE NUMBER OF PIXELS TO BE SAMPLED'
      READ(*,*) ANO
      N1 = N - NINT(ANO/2)
      N2 = N1 + (ANO - 1)
*
*   Open files for output data.
*
      OPEN(3,FILE='ERRDAT.DAT',STATUS='NEW')
      OPEN(4,FILE='RSSDAT.DAT',STATUS='NEW')
      OPEN(5,FILE='RSSPLT.DAT',STATUS='NEW')
```

```
*
*   Vary the value of the beam spot size in order to generate
*   a plot file to allow for multiple plots of ERROR vs XBAR
*   while varying the parameter  W.
*
          W = .000020
*
          DO 300 KK=1,6
          WRITE(4,*) 'W = ',W
*
*   Set up a loop to increment XBAR across the pixel where
*   the centroid is located, in um. increments.
*
        XBAR = (N-1)*DELX
*
        DO 40 II = 1,25
*
*   Reinitialize variables for next calculation.
*
        TEMP = 0.0
        X = 0.0
        J = 0
        K = 0
*
*   Loop to compute the sum of the responses Yi.
*
        DO 10 I=N1,N2
        J = J + 1
        XI = DELX * (I - .5)
        YI(J) = Y(XI,XBAR,W)
        TEMP = TEMP + YI(J)
 10     CONTINUE
*
*   Compute the product of Xi and Yi
*
        DO 20 JJ=N1,N2
        K = K + 1
        PDF = YI(K) * (DELX * (JJ -.5))
        X = X + PDF
 20     CONTINUE
*
*   Now calculate the mean
*
        MEAN = X / TEMP
        ERR = MEAN - XBAR
        WRITE(3,700) XBAR,ERR
        WRITE(4,900) N2-N1+1,XBAR,ERR
        XBAR = XBAR + .000001
        RSS = RSS + ERR*ERR
 40     CONTINUE
        WRITE(4,*) 'THE RSS ERROR IS ',SQRT(RSS)/(II-2)
        WRITE(5,700) REAL(KK),SQRT(RSS)/(II-2)
        W = W + .000005
```

```
 300   CONTINUE

 700   FORMAT(2E15.7)
 900   FORMAT(I5',' E15.7',' E15.7)
       CLOSE (3)
       CLOSE (4)
       CLOSE (5)
       STOP
       END

**********************************************************************
*                                                                    *
*     FUNCTION Y          V1.1                                       *
*                                                                    *
*     Author: Jack Carter Jr.                                        *
*     date: 8/9/86                                                   *
*                                                                    *
*     The following function evaluates the Gaussian wave front for   *
*     given values of X, XBAR and W.                                 *
*                                                                    *
*     Definition of variables:                                       *
*     NUM = -((X - XBAR)**2)  the numerator of the function          *
*     DEN = W * W  the denominator of the function                   *
*                                                                    *
**********************************************************************
      REAL FUNCTION Y(X,XBAR,W)
      REAL NUM, DEN, X, XBAR, W
*
      NUM = (X-XBAR)**2
      DEN = W * W
      Y = EXP(-(NUM/DEN))
*
      RETURN
      END
```