NASA Contractor Report 182094

HEAVY NUCLEUS COLLECTOR (HNC) PROJECT FOR THE NASA LONG DURATION EXPOSURE FACILITY (LDEF)

Gregory Tarle

THE UNIVERSITY OF MICHIGAN
Ann Arbor, Michigan

Contract NAS1-17820
September 1990

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

Project Overview

The primary goal of the HNC experiment was to obtain high resolution composition measurements for cosmic ray nuclei in the platnum-lead and actinide region of the periodic table. Secondary objectives include studies of selected groups of elements of lower charge. These goals were to be realized by orbiting a large area array of dielectric nuclear track detectors be in space for several years. In this time we would collect sufficient actinide nuclei to determine the nucleosynthetic age of the cosmic radiation and the relative mix of r- and s-process elements in the cosmic ray source.

The detector was to consist of approximately 50 trays assembled in pressurized canisters. Each tray would contain 8 half-stacks (4 stacks total) and an event thermometer which would record the temperature of each event at the time of exposure. Each stack would contain seven layers of Rodyne-P, CR-39 and Cronar plastic track detectors interleaved with Copper stripping foils.

Upon return to earth, the detectors will be removed for analysis. Ultraheavy nuclei would have left "tracks" through the detector sheets that would be made visible after etching in a hot sodium hydroxide solution.

The primary responsibility of the University of Michigan under was to produce and assemble the plastic track detectors for the HNC experiment project. The tragic loss of the Space Shuttle Challenger first caused an indefinite delay and later a cancellation of the HNC project. During the final years, it was decided to use some of the remaining HNC funds to upgrade the University of Michigan automatic microscope system that would be used to measure large numbers of nuclear that would be returned from such a mission.

This report contains two parts. The first is a log that describes the activities carried out to produce the detector stacks for the HNC experiment. The second part describes the automatic microscope system that was developed to scan and measure nuclear tracks.

Part I

Fabrication of the HNC detectors—Project Log

Laboratory Set-Up

The following measures were taken to guarantee security and quality control during the detector stack fabrication and assembly:

Bonded stores: Room 148A of the Randall Laboratory of Physics at the University of Michigan was designated for pre-flight storage of all stack materials except CR-39 (see bonded cold stores below). Bonded stores is secured by an 8ft. chain-link fence padlock.

Bonded cold stores: Room 148 contained an $8' \times 8' \times 8' \times 7''$ Norlake walk-in freezer with holding temperature of $-14°F$, to guarantee CR-39 detector sheet preservation over pre-flight storage periods. CR-39 sheets were packaged and placed in storage immediately after inspection. Bonded cold stores is secured by a lock on the freezer door.

Light filters: Yellow light filters were installed over all florescent tubes in the sheet handling area (Room 148) to block damaging UV radiation.

Cleanroom: A filtered clean-room within bonded stores (Room 148A) ensured a dust-free environment for the stacking operations (see Appendix E for clean-room layout). The cleanroom contains the following equipment for use during the receipt inspection, serializing and stacking procedures:

- IBM PCXT, monitor and printer
  To run coordinating stacking program (see below)
- Sartorious scale and windshield
  To guarantee detector sheet uniformity by weight (see Appendix C for tolerances).
- Tektronix Digital Plotter and diamond scribe
  For sheet serializing (see Appendix H for scribe design).
- Heat sealer, aged dry air, Marvelseal packaging material
  For packaging completed stacks and sheet materials in air/water/light tight seal
- Cleanroom operator garb including nylon lint free hats, smocks, gloves, booties and particle masks.

Stacking program: The stacking procedure was coordinated by the HNC serializing and stacking program "BLDSTK.BAS" operable on the IMB PCXT. The program leads the operator through the procedure in a step-by-step fashion. Program includes:

- log in of operator, date and time
- inventory of available sheet materials to guarantee proper set-up
- weighing procedure to discard sheets outside set tolerances (see Appendix C)
- scribing procedure to serialize sheets based on serializing code (see Appendix A)
- stacking procedure to stack sheets in proper order (see Appendix B for stacking configuration)
- packaging and labeling procedure using the plotter to print identifying labels
- documentation including:

- continuous printout of stacking procedure
- floppy diskette data files with sheet serial numbers and weights

CR-39 Detector Sheet Manufacture

CR-39 detector sheets were manufactured from monomer with greater than 90% diallyl diglycol carbonate (DADC) into 10 mil thick sheets which are laser cut to size following NASA blueprint #LC-819728.

Manufacturing Log:

- Monomer supplied by SFOS under the trade name "CAD Monomer Allymer" 5-55 gallon drums received at UM 1/4/85 (UM P.O.#L03450, $1318.75 per drum)
- CAD monomer tested
  - by UM for viscosity 2/4-2/13/85 (see Appendix K)
  - by Noury Chemicals for chemical and physical properties 4/4/85 (see Appendix J)
- two one-gallon samples taken from each monomer drum were stored in Room 148A
- 1% DOP (dioctyl phthalate) and 0.01% N-445 (Nauguard 445 high performance amine antioxidant) by weight was added to the intermixed monomer
- Monomer and additives were returned to drums and shipped to American Acrylics and Plastics (AA&P) to be manufactured into CR-39 sheets 5/6/85
- CR-39 received from AA&P:
  - 320 sheets (UM P.O.#L00921, $60 per sheet) manufactured from standard Barberton monomer with 1%DOP, not CAD-M
  - 44 crates received 6/24-4/14/86 (UM P.O.#L07673) 100 sheets per crate
- CR-39 laser cut by Laser Machining, Inc.:
  - 120 sheets, 7/3/85, UM P.O.#L19876
  - 220 sheets, 10/24/85, amendment to above P.O.#
  - 148 sheets, 10/28/85, sent to LaRC for Q/T sheets
  - 220 sheets, 11/8/85, for Calibration sheets
  - 35 sheets, 11/11/85, for Calibration tabs

Rodyne-P Detector Sheet Manufacture

Rodyne-P detector sheets were manufactured from Mobay bisphenol-Apolycarbonate resin into 5 and 10 mil thicknesses which are die-cut to NASA blueprint #LC-819727 specifications. Some sheets contain 0.01% by weight N-512 antioxidant and most contain DOP.

Manufacturing log:

- Rodyne-P supplied by Coburn Plastics Corp:
  - 500 yards on 20 rolls received at UM 1/25/85 (UM P.O.#L00313)
  - 7 rolls 20 mil, with N-512
  - 5 rolls 20 mil, without N-512
  - 4 rolls 10 mil, with N-512
  - 4 rolls 10 mil, without N-512
  - 13,175 sheets received at UM 12/4/85 (UM P.O.#L31158)
  - 12,000 sheets 10 mil
  - 1 lot (1300-1500) sheets 5 mil
- 1100 sheets of 10 mil Rodyne-P die cut by Coburn Plastics Corp. for test stacks received at UM 2/6/85
- 600 sheets 10 mil Rodyne-P die cut by UM Physics instrument shop for Q/T stacks, received 12/5/85

Cronar Detector Sheet Manufacture

Cronar detector sheets were manufactured by DuPont in 4 mil thicknesses as CR42 polyester graphics art film which is die cut to NASA blueprint #LC-819730 specifications. Manufacturing log:

- Cronar supplied by Northern Graphics supply
  13,500 sheets (133 boxes) received at UM 11/1/85
- Cronar sheets die cut for Q/T stacks by UM physics instrument shop
  - 400 cut sheets received 12/5/85
  - additional cut sheets received 4/22/86

Copper Sheet Manufacture

Copper sheets were manufactured from type-101 OFHC Certified copper ASTM F6877 into 10 mil thick sheets which were die-cut to NASA blueprint #LC-819729 specifications. Manufacturing log:

- Copper supplied by Olin Brass
  5900 lbs. (approx. 6775 sheets) received at UM 4/14/86 (UM P.O.#C97746, approx. $15,197.75)
- 1 case copper sheets die cut for Q/T stacks by UM physics instrument shop, received 4/22/86

Calibration Sheets and Calibration Tabs

Calibration sheets are plastic track detectors exposed to relativistic ^{56}Fe nuclei (for CR-39 sheets) or ^{238}U nuclei (for Rodyne-P and Cronar sheets) at the Lawrence Berkeley Bevelac. They were to be used to monitor post flight thermal annealing of tracks.

Calibration tabs were detectors irradiated with the calibration sheets. They do not fly with the experiment but remain in storage until experiment completion for comparative analysis and identification.

Calibration packets each containing enough calibration sheets and calibration tabs for one canister were packaged in Marvelseal and sent to Bevelac at the University of California in Berkeley for irradiation.

See Appendix B for exact placement of calibration sheets in the detector stacks.

Qualification and Test Stack Assembly

Qualification and test (Q/T) stacks were built to test the overall detector and canister design for the HNC project. They are assembled identically to proposed detector stacks except that unirradiated detector sheet is substituted for each calibration sheet in the stack configuration. The completed stacks were sent to the NASA Langley Research Center for testing.

HNC Project Mothballing

Due to the space shuttle Challenger accident, further work on the HNC project was originally postponed until a new flight date could be scheduled. To preserve the detectors until such time, all plastic sheet materials were double-bagged in Marvelseal packaging material and purged with dry air before being sealed and placed in bonded cold stores. All copper sheets remained crated in bonded stores.

Bonded Cold Stores Inventory

CR-39 Packet No.	Date Inspected	No. of Sheets	Date in Cold Stores
1	9-26-85	100	10-4-85
2	10-2-85	100	10-5-85
3	10-21-85	100	10-5-85
4	10-5-85	100	10-10-85
5	10-10-85	100	10-14-85
6	10-14-85	100	10-18-85
7	10-10-85	100	10-19-85
8	10-30-85	100	11-1-85
9	10-19-85	100	11-2-85
10	11-1-85	100	11-5-85
11	11-6-85	100	11-6-85
12	11-6-85	100	11-7-85
13	11-8-85	70	11-11-85
14	11-8-85	100	11-12-85
15	11-14-85	100	11-14-85
16	11-14-85	100	11-15-85
17	11-15-85	100	11-15-85
18	11-15-85	100	11-19-85
19	11-19-85	100	11-20-85
20	11-20-85	100	11-22-85
21	11-22-85	100	11-25-85
22	11-22-85	100	11-26-85
23	12-5-85	100	12-5-85
24	12-5-85	100	12-11-85
25	1-14-85	100	1-16-85
26	bad oven charts	44	1-20-86
27	1-22-86	100	1-24-86
28	1-22-86	100	1-24-86
29	1-28-86	100	2-7-86
30	2-11-86	100	2-12-86
31	rejects	100	2-20-86
32	rejects	100	2-20-86

33	rejects	100	2-20-86
34	3-3-86	100	3-4-86
35	3-31-86	100	4-8-86
36	4-14-86	100	5-1-86
37	5-1-86	100	5-1-86
38	8-8-86	100	8-12-86
39	8-11-86	100	8-12-86
40	rejects	73	8-12-86
crate 1	cal shts./tabs	125	
crate 2	cal shts./tabs	125	

Rodyne-P

Packet No.	Date Pkgd.	No. of Sheets	date in cold stores	thickness (mil)
0001	3/5/87	125	3/13/87	10
0002	3/5/87	150	3/13/87	10
0003	3/5/87	125	3/13/87	10
0004	3/6/87	125	3/13/87	10
0005	3/9/87	125	3/13/87	10
0006	3/9/87	125	3/13/87	10
0007	3/9/87	125	3/13/87	10
0008	3/10/87	125	3/13/87	10
0009	3/10/87	125	3/13/87	10
0010	3/10/87	125	3/13/87	10
0011	3/10/87	125	3/13/87	10
0012	3/10/87	125	3/13/87	10
0013	3/10/87	125	3/13/87	10
0014	3/10/87	125	3/13/87	10
0015	3/10/87	125	3/13/87	10
0016	3/10/87	125	3/13/87	10
0017	3/10/87	200	3/13/87	5
0018	3/10/87	200	3/13/87	5
0019	3/10/87	200	3/13/87	5
0020	3/10/87	200	3/13/87	5
0021	3/10/87	170	3/13/87	5
0022	3/11/87	125	3/13/87	10
0023	3/11/87	125	3/13/87	10
0024	3/11/87	125	3/13/87	10
0025	3/11/87	125	3/13/87	10
0026	3/11/87	125	3/13/87	10
0027	3/11/87	125	3/13/87	10
0028	3/11/87	125	3/13/87	10
0029	3/11/87	125	3/13/87	10
0030	3/12/87	125	3/13/87	10
0031	3/12/87	125	3/13/87	10
0032	3/12/87	125	3/13/87	10
0033	3/12/87	125	3/13/87	10
0034	3/12/87	125	3/13/87	10
0035	3/12/87	125	3/13/87	10
0036	3/12/87	125	3/13/87	10
0037	3/12/87	125	3/13/87	10
0038	3/12/87	125	3/13/87	10
0039	3/12/87	125	3/13/87	10
0040	3/12/87	150	3/13/87	10

0041	3/12/87	125	3/13/87	10
0042	3/12/87	150	3/13/87	10
0043	3/13/87	150	3/13/87	10
0044	3/13/87	125	3/13/87	10
0045	3/13/87	150	3/13/87	10
0046	3/13/87	150	3/13/87	10
0047	3/13/87	150	3/13/87	10
0048	3/19/87	125	3/25/87	10
0049	3/19/87	125	3/25/87	10
0050	3/19/87	125	3/25/87	10
0051	3/19/87	125	3/25/87	10
0052	3/19/87	125	3/25/87	10
0053	3/19/87	125	3/25/87	10
0054	3/19/87	125	3/25/87	10
0055	3/19/87	125	3/25/87	10
0056	3/19/87	125	3/25/87	10
0057	3/19/87	125	3/25/87	10
0058	3/20/87	125	3/25/87	10
0059	3/20/87	125	3/25/87	10
0060	3/20/87	125	3/25/87	10
0061	3/20/87	125	3/25/87	10
0062	3/20/87	125	3/25/87	10
0063	3/20/87	125	3/25/87	10
0064	3/23/87	125	3/25/87	10
0065	3/23/87	125	3/25/87	10
0066	3/23/87	125	3/25/87	10
0067	3/23/87	125	3/25/87	10
0068	3/23/87	125	3/25/87	10
0069	3/23/87	125	3/25/87	10
0070	3/23/87	125	3/25/87	10
0071	3/23/87	125	3/25/87	10
0072	3/23/87	125	3/25/87	10
0073	3/23/87	125	3/25/87	10
0074	3/23/87	125	3/25/87	10
0075	3/24/87	125	3/25/87	10
0076	3/24/87	125	3/25/87	10
0077	3/24/87	125	3/25/87	10
0078	3/24/87	125	3/25/87	10
0079	3/24/87	125	3/25/87	10
0080	3/24/87	125	3/25/87	10
0081	3/24/87	125	3/25/87	10
0082	3/24/87	125	3/25/87	10
0083	3/24/87	125	3/25/87	10

0084		3/24/87	125	3/25/87	10
0085		3/24/87	125	3/25/87	10
0086		3/24/87	125	3/25/87	10
0087		3/24/87	125	3/25/87	10
0088		3/24/87	125	3/25/87	10
Misc. A		3/25/87	150	3/26/87	10
Misc. B		3/25/87	200 (QT shts)	3/26/87	5
Misc. C		3/25/87	approx. 150 (thick.var.)	3/26/87	10
Misc. D		3/25/87	150	3/26/87	10
Misc. E		3/25/87	approx. 100 (thick.var)	3/26/87	10
Misc. F		3/25/87	150	3/26/87	10
Misc. G		3/25/87	150	3/26/87	10
Misc. H		3/25/87	150	3/26/87	10
Misc. O	L	3/25/87	100 (some cut,?)	3/26/87	5
	M	3/25/87	approx. 100 (some ?)	3/26/87	10
	N	3/25/87	approx. 100 (some ?)	3/26/87	10

Cronar

Packet No.	Date Pkgd.	No. of Sheets	Date Into Cold Stores
0001	2/9/87	300	3/2/87
0002	2/9/87	300	3/2/87
0003	2/9/87	300	3/2/87
0004	2/9/87	300	3/2/87
0005	2/9/87	300	3/2/87
0006	2/9/87	300	3/2/87
0007	2/9/87	300	3/2/87
0008	2/11/87	300	3/2/87
0009	2/11/87	300	3/2/87
0010	2/11/87	300	3/2/87
0011	2/11/87	300	3/2/87
0012	2/11/87	300	3/2/87
0013	2/12/87	300	3/2/87
0014	2/12/87	300	3/2/87
0015	2/12/87	300	3/2/87
0016	2/27/87	300	3/2/87
0017	2/27/87	300	3/2/87
0018	2/27/87	300	3/2/87
0019	2/27/87	300	3/2/87
0020	2/27/87	300	3/2/87
0021	2/27/87	300	3/2/87
0022	2/27/87	300	3/2/87
0023	2/27/87	300	3/2/87
0024	3/2/87	300	3/9/87
0025	3/2/87	300	3/9/87
0026	3/2/87	300	3/9/87
0027	3/2/87	300	3/9/87
0028	3/3/87	300	3/9/87
0029	3/3/87	300	3/9/87
0030	3/3/87	300	3/9/87
0031	3/3/87	300	3/9/87
0032	3/3/87	300	3/9/87
0033	3/3/87	300	3/9/87
0034	3/3/87	300	3/9/87
0035	3/3/87	300	3/9/87
0036	3/3/87	300	3/9/87
0037	3/3/87	300	3/9/87
Misc. I	3/25/87	300	3/26/87
Misc. J	3/25/87	300	3/26/87
Misc. K	3/25/87	300	3/26/87
Cal. Sheets	3/3/87	389 (1200-?bad)	3/9/87

Appendix A
Sheet Serializing Code

Calibration sheets are scribed with an 11-digit serial number. All other sheets are scribed with a 9-digit number. The codes are as follows:

Cal-sheet identification (cal-sheets only)	CS
Canister number designation	01-
Stack-quad location identification	A-D
Canister side location	1-2
Layer numbers	1-7
Layer sheet numbers	01-12
Stack sheet numbers	01-72

Example: 05C160933

The above example identifies a hypothetical sheet as located in:

1) the fifth (5th) CANISTER, in
2) Quad "C", on
3) CANISTER SIDE "One" (1), in
4) LAYER NUMBER "Six" (6), being
5) LAYER SHEET NUMBER, "Nine" (09), and is
6) STACK SHEET NUMBER, "Thirty-Three" (33)

Appendix B

HNC Stack Configuration

No.	Item	No.	Item	No.	Item	No.	Item	No.	Item	No.	Item	No.	Item
01	R	12	CR	23	CU	34	C+	45	C	56	C	67	C
02	R+	13	CU	24	C	35	C	46	C	57	C	68	CU
03	CU	14	C	25	C	36	C	47	C	58	CU	69	R
04	C+	15	C	26	C	37	C	48	CU	59	R+	70	R
05	C	16	C	27	C	38	CU	49	R	60	R	71	R
06	C	17	C	28	CU	39	R+	50	R	61	R	72	CR
07	C	18	CU	29	R	40	R	51	R	62	CR		
08	CU	19	R	30	R	41	R	52	CR	63	CU		
09	R	20	R+	31	R	42	CR	53	CU	64	C+		
10	R+	21	R	32	CR	43	CU	54	C	65	C		
11	R	22	CR	33	CU	44	C	55	C	66	C		

CU = copper
R = Rodyn-P
C = Cronar
CR = CR-39
+ = skeleton etch

Appendix C

Sheet Weight Tolerances

The following weight tolerances were incorporated into the serializing and stacking program "BLDSTK.BAS" to ensure detector sheet uniformity and proper material identification by the operator during the stack assembly procedure.

Material	Tolerance max	(in grams) min
10 mil Rodyne-P	59	45
4 mil Cronar	32	21
10 mil CR-39	70	50
10 mil copper	376	368

Appendix D
HNC Correspondence References

AKZO-CHEMIE America
Noury Chemicals
8401 W. 47th Street
Attn: Bob, Margevich, Technical Development Manager

American Acrylics & Plastics
25 Charles Street
Stratford, CN 06497
Attn: George Barry, President

Coburn Plastics Corp.
1650-T Corporate Rd. W
Lakewood, NJ 08701
Attn: Paul Riske, Manager Industrial Films

Laser Machining, Inc.
500 Laser Drive
P.O. Box 219
Somerset, WI 54025
Attn: Ted Bauer, Robert Bosworth

Northern Graphics Supply
2216 Midland Road
Saginaw, MI 48603

Olin Brass
19 Atterbury Blvd., Suite 11-B
Hudson, OH 44236

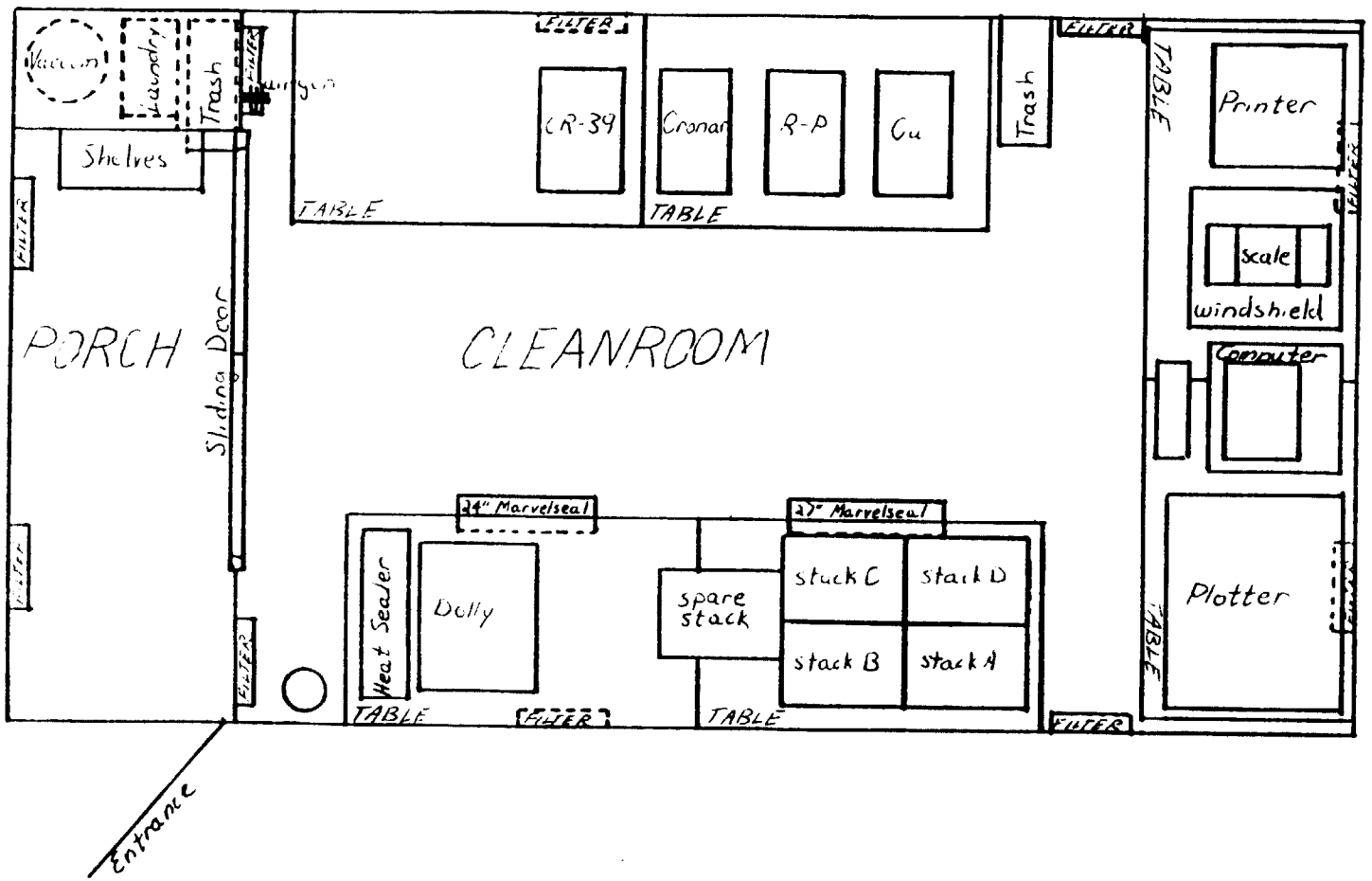
O'Neal, Bob
Project Manager
Mail Stop 258
NASA Langley Research Center
Hampton, VA 23665

Price, Professor Buford
Department of Physics
University of California, Berkeley
Berkeley, CA 94720

Sola Optical
3600 Lakeville Hwy.
Petaluma, CA 94952

Appendix E

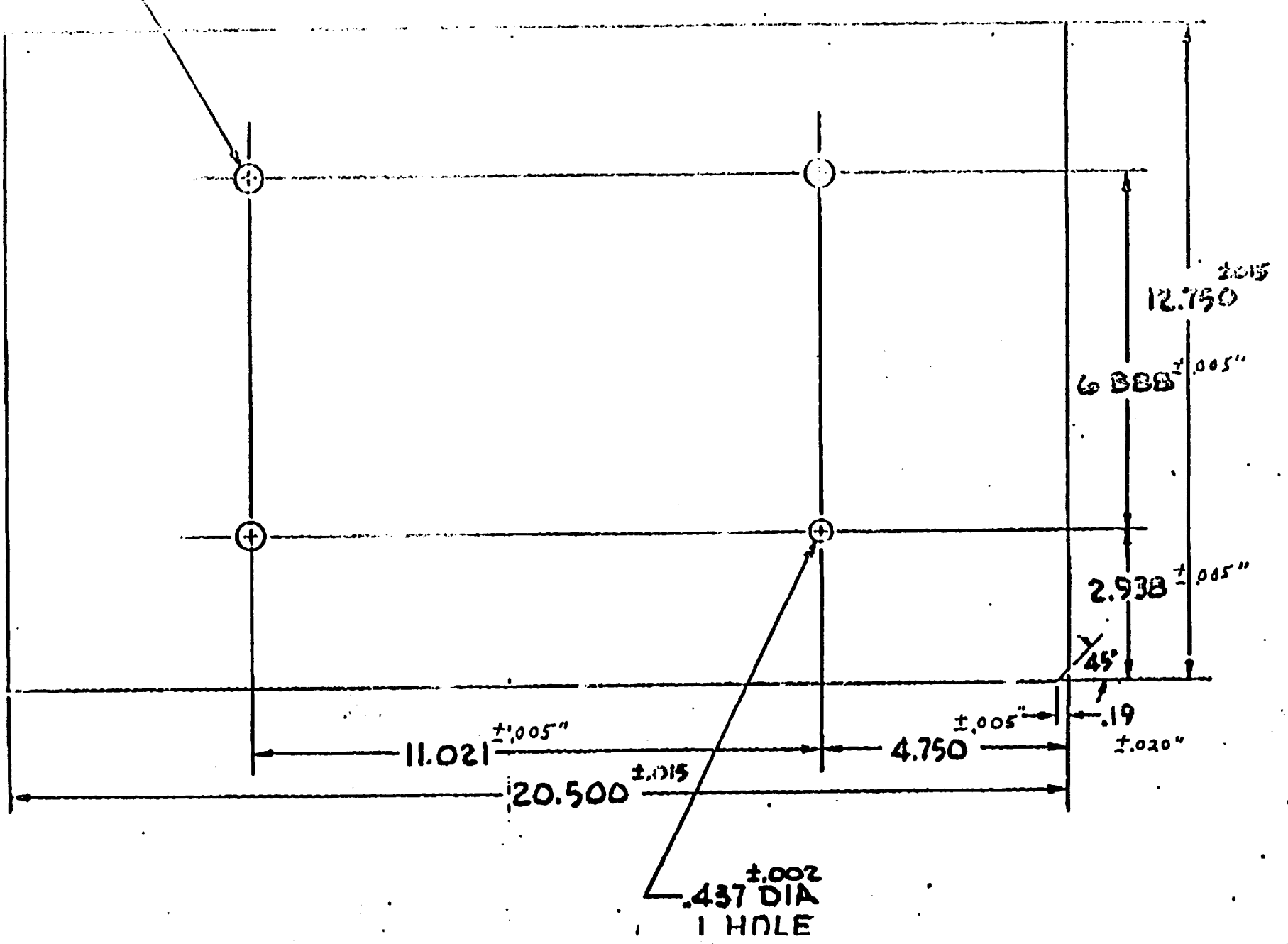
Cleanroom Layout for Stacking and Serializing



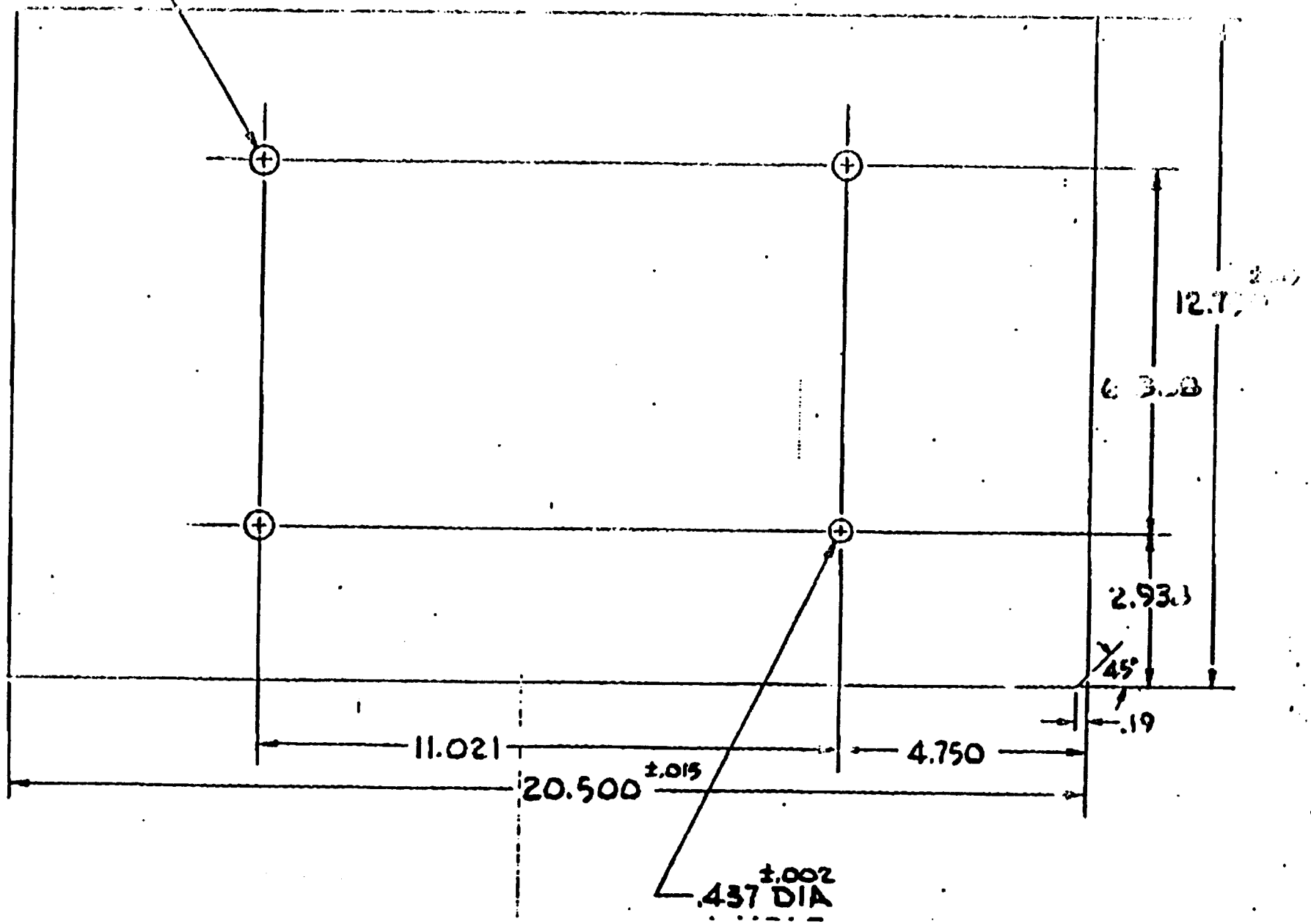
scale 0 6 12 18 24 30 36 42 48 54 60 in

0 1 2 3 4 5 ft

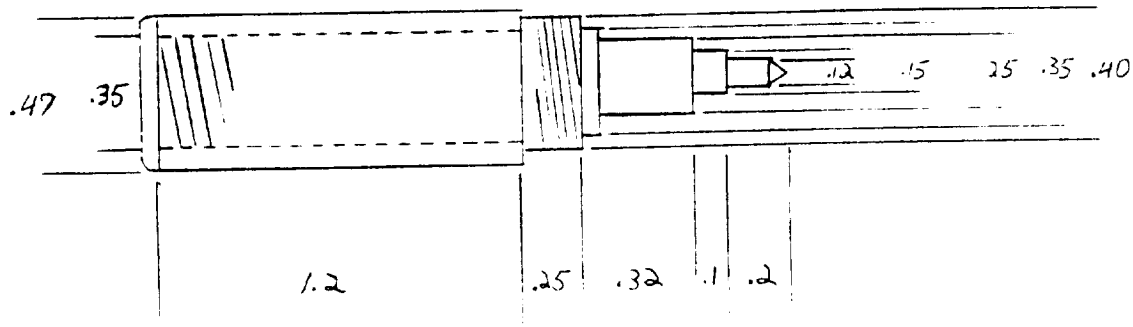
Appendix F
Copper and Cronar Sheet Dimensions



Appendix G
Rodyne-P and CR-39 Sheet Dimensions



Appendix H
Diamond Scribe



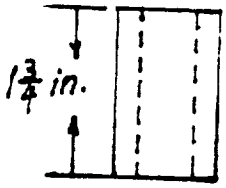
all dimensions in inches

Appendix I

Temporary Stacking Hardware

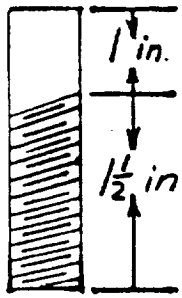
The following hardware was manufactured at the University of Michigan to be used during the stacking operation for stabilizing purposes. All temporary stacking hardware was removed from the stacking fixtures upon completion of the stacking operation and before installation of the mounting hardware.

A) Aluminum Bushing



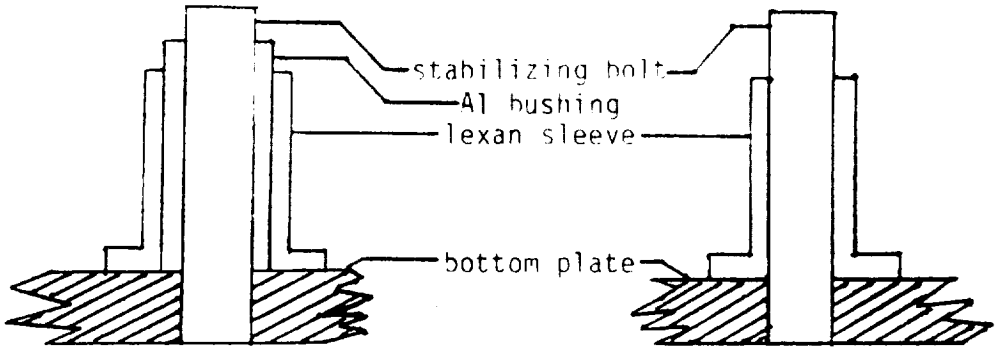
O.D. 0.435 in.
I.D. 0.325 in.
quantity: 15

B) Stabilizing Bolt



O.D. 0.30 in.
steel
5/16 - N.C.
quantity: 20

Installation of the Temporary Stacking Hardware (cross-section)



Appendix J

Analysis of CAD-N Monomer
Akzo Chemie America, Noury Chemicals
Ref: ARCS-39
Project Number 0684075039

Eight samples of CAD-N Monomer were received from the University of Michigan for analysis. These samples were analyzed by wet methods, physical analysis, HPLC, and gas chromatography. The results of the wet and physical analyses are shown in Table 1. The viscosity was done by the falling ball method. Karl Fischer titration was used to measure water. The pH was measured on a 10% solution in 50/50 water-ethanol. Inorganic and organic hydrolyzable chlorine was measured using the Noury procedure.

Table 1

Wet and Physical Analysis of CAD-N Monomer Samples

	SAMPLE NO.							
	1	2	3	4	5	6	7	8
Viscosity (mpa · s)*	18.5	18.7	18.5	18.4	18.5	17.9	18.5	19.2
Water (ppm)	797	791	733	677	749	1186	541	477
Color (APHA)	5	5	5	5	5	5	5	5
pH	5.2	5.7	5.5	5.5	5.6	4.8	4.6	4.7
Refractive Index	1.4530	1.4522	1.4522	1.4521	1.4520	1.4525	1.4517	1.4527
Inorganic (ppm)	2.5	2.3	2.4	4.2	4.2	1.8	11.8	.9
Organic Hydrolyzable Chlorine								

*Dynamic viscosity millipascals x seconds

Sample Code:

1-5 = Allymer drums no. 1-5
6 = misc.
7 = #3178-5-16
8 = #5239 #5

High performance liquid chromatography (HPLC) was used to determine the monomer, dimer and trimer content. The allyl diglycol carbonate was also measured. It was also observed that all samples contained some polymer. The HPLC results are shown in Table 2.

Table 2

HPLC Analysis of CAD-N Monomer Samples

	SAMPLE NO.							
	1	2	3	4	5	6	7	8
n = 1	92.4	92.1	86.6	93.6	88.0	93.0	86.8	94.1
n = 2	4.5	4.3	3.8	4.5	4.1	3.9	5.3	4.9
n = 3	0.6	0.5	0.4	0.3	0.3	0.8	0.8	0.4
Allyl Diglycol Carbonate	1.6	1.6	1.4	1.6	1.3	1.5	0.9	0.5

Gas chromatography was used to measure the acrolein (ppm), allyl alcohol (ppm) and diallyl carbonate (%) content. The results are shown in Table 3.

Table 3

Gas Chromatographic Analysis of CAD-N-Monomer

	SAMPLE NO.							
	1	2	3	4	5	6	7	8
Acrolein (ppm)	68	40	178	73	49	256	127	147
Allyl Alcohol (ppm)	58	29	32	43	36	88	73	94
Diallyl Carbonate (%)	0.25	0.21	0.25	0.21	0.22	0.31	0.31	0.33

Appendix K

Viscosity Measurements of CAD - Monomer

The viscosity of the monomer in each of the five drums received 1/4/85 (UM P.O. #L03450) was measured at the University of Michigan prior to the monomer use in the manufacture of CR-39 sheets.

The following data was obtained using the "Falling Ball Viscometer Method" (viscometer accuracy: 0.2 - 1.0%):

Sample No.	Centipoise	Avg. Centipoise	Denisty
1	15.0059		
1	14.9531	14.9669	1.1406
1	14.9418		
2	14.8605		
2	15.3168	15.0151	1.1407
2	14.8651		
3	14.9288		
3	15.0082	15.0325	1.1408
3	15.1107		
4	14.8005		
4	14.9124	14.9014	1.1411
4	14.9914		
5	14.9731		
5	15.0597	15.0532	1.1412
5	15.1268		

The following data was obtained from density measurements used to check the accuracy of the viscosity measurements:

Sample No.	Mass (g)	Volume (cc)
1	114.06	100
2	114.07	100
3	114.08	100
4	114.11	100
5	114.12	100
avg. density = 1.14088		

Part II
Description of the University of Michigan Automatic Nuclear Track Measuring Microscope

1. Introduction

In this section the hardware and software for the track measuring microscope system are briefly described. The original goal of the group was to develop a system capable of locating and measuring 1 million cosmic-ray events per year automatically, for use on a variety of experiments requiring large collection areas and precise charge resolution. While the superior charge resolution of CR-39 plastic is well known, the limitations imposed by existing image processing technology make accurate identification and measurement of nuclear tracks in a reasonable time a formidable task. The system that has evolved in our lab over the past four years represents a compromise between speed and accurate identification and measurement suitable for our purposes. The time required for measurement approaches that indicated above. While further improvements in hardware could improve our analysis time slightly, advances in the development of image processing systems since the inception of the current project (circa 1985) hold the prospect of dramatically improving the time required for analysis and software development for experiments of this type. Still, it is hoped that the techniques and algorithms described in this section will find application in future systems.

The procedure for analysis of CR-39 data will be described briefly here and in more detail in section 4. This procedure begins by determining a coordinate system on the sheet of interest, which will serve as a reference coordinate system for all subsequent track position identification. This coordinate system is described by two parameters, the location of the point (0,0) and a rotation angle, θ. All sheet coordinates are given in terms of the (x,y) location of the stage at the point of interest relative to a position on the sheet which defines the origin (0,0). In practice, the origin of the coordinate system is taken to be the centroid of a selected track which is normally incident to the surface of the detector. The position of a second track is then used to define a reference coordinate system ($\theta = 0$). If the sheet is removed from the stage, and then replaced, any rotation introduced in the position of the sheet may be removed by locating the reference tracks and comparing their stage coordinates to those of the reference locations of these same tracks. In this way all subsequent scans of the sheet may be referenced to identical coordinate systems.

Next, each side of each sheet is scanned at low power (typically 3.2X or 10X) and the locations of all candidate tracks are identified and their coordinates written to disk, along with various parameters which have been calculated for these tracks. In to eliminate the background of surface defects and non-penetrating tracks from low energy particles from consideration as candidate events, the opposite side of this surface is also scanned and penetrating events are identified by coincidence between these two surfaces, after the paths of the events in the plastic are reconstructed.

The results of this scan are then used to make final measurements of track parameters at high power where these measurements can be made more accurately to improve the charge resolution that can be obtained from the plastic.

These procedures have been used on data obtained from accelerator experiments and also on balloon flight data. The performance of the system is discussed in section 5.

2. Microscope Hardware

A schematic diagram of the automated measuring system is shown in figure 1. The microscope consists of a Technical Instrument Measuremaster II stage with 5 Leitz objectives of 3.2X, 10X, 20X, 32X and 50X magnification. The microscope is equipped with a Hamamatsu C1000 Vidicon camera. The output of the camera is directed to a VICOM model 1850 image processing system, where the image is digitized and subsequent analysis is performed. Mass storage is provided by a CDC 450 Megabyte hard disk, where the results of the image analysis are collected. Data files can be transferred to and from the VICOM by means of a DMA link with a DEC microVAX, where the results can be further analyzed and plotted without using CPU time dedicated to image analysis.

The stage of the microscope system has a 6X6 "travel on 1mm pitch lead screws and is controlled by two 25,000 step/revolution Compumotor stepping motors and Compumotor 2100 series indexers. The position of the stage is monitored by a Heidenhain length measuring system, giving absolute position information accurate to $\pm 1\mu$m over its entire length of travel. The focus of the microscope is monitored by a Leitz Laser Autofocus system.

The VICOM image processor is built around a Motorola VM03-4 microcomputer, and includes a fast point processor and array processor in a pipeline architecture. This allows various image operations, such as region of interest processing, histogramming and 3X3 pixel convolution, to be accomplished at the frame rate (approximately 1/30 second). These capabilities have proved to be extremely useful in the design of the image processing algorithms required to smooth image data and extract edges from potential tracks, as described below. The images are digitized by a 8 bit/pixel analog to digital converter. The digitized images are written to a 512X512X2 byte segment of memory dedicated to image storage. Three additional segments of memory are available to store the result of image processing operations. The VICOM also has some graphics capabilities, with 4 bits per word in image memory reserved for graphics display.

The microcomputer also controls the stage and autofocus via an RS-232 link with the Compumotor indexers. The motion of the stage is initiated by serial command strings sent by the microcomputer. The indexers contain their own 6802 microprocessor, so the operation of controlling and monitoring the position of the stage is divorced from the VICOM microprocessor itself, and real-time analysis on digitized images can proceed while stage movement is in progress.

The Leitz Laser Autofocus system can be controlled via an operator keypad or by means of programmable bits on the Compumotor indexers which can be accessed from the RS-232 link. In practice, both methods of control are used during the plastic analysis, as discussed below. The focus axis is also equipped with a Heidenhain length measuring system and special electronics to give .1 μm position resolution over the entire length of travel. Optionally, the position of the stage can be controlled by a Compumotor stepping motor to this same accuracy.

During analysis, the plastic is mounted on a specially constructed aluminum stage equipped with a vacuum line that holds the sheet in place during analysis and keeps the sheet flat to within .001". This design restricts us to the use of reflected light for viewing samples. When viewing samples using a low power, long focal length objective, a reflective paper backing is used to increase image contrast. This also improves the performance of the laser autofocus by reducing reflection from the laser off of the aluminum stage.

The entire microscope apparatus is mounted on a large marble block and isolated from vibrations by a specially designed microscope stand with sand-filled legs.

3. Analysis Software

The Motorola VM03-4 microcomputer in the model 1850 VICOM image processing system runs under the VERSAdos operating system and includes a resident PASCAL compiler and Motorola assembler for 68000 based systems. VICOM provides a library of general-purpose image processing routines and software necessary to program the special purpose hardware.

One of the original goals in developing the microscope software was to provide an environment for quickly setting up and measuring sheets by an essentially unskilled worker with a minimal amount of training. Another goal was to make the program entirely modular to save time in development of the routines required for the plastic analysis. To these ends we have designed the system around a central menu program, which provides an easily learned user interface as well as the capacity to load and run the various analysis programs and monitor their status. This structure allows the development of the different analysis programs to occur simultaneously, and avoids the problems inherent in developing all of the analysis routines in a single monolithic structure. While menu system remains resident in memory, the data analysis programs can be swapped in and out of memory under control of the menu and the limited memory space on the VICOM is very efficiently used.

The menu user interface is mouse driven and can be operated on a Tektronix 4014 compatible terminal. The VERSAdos operating system is built around a real-time multi-tasking kernel, and offers wide variety of sophisticated intertask communication facilities suitable for our purposes. The menu also contains a shared segment of memory containing the data structures necessary for passing information between analysis programs. Figure 2 shows a schematic diagram of the menu system and associated analysis tasks.

The heart of our analysis system are the image processing routines. The accuracy with which these identify and measure tracks largely determines the resolution which the experiment will have, and the speed at which these routines execute is the primary constraint on total analysis time. Our philosophy throughout has been to dedicate the VICOM entirely to low-level image processing tasks, while data reduction and further analysis done on the microVAX. The problems that this has introduced into our analysis procedure are discussed below. The VICOM image processor contains a large library of PASCAL-callable subroutines and a resident PASCAL compiler, and thus we have chosen to program the analysis programs in PASCAL, and to use assembly language for portions of the programs where speed is essential.

As discussed previously, the microscope image is produced in reflected light. Tracks produced with various angles of incidence appear as ellipses with various internal features which generally reflect considerably less light than the horizontal plastic surface. The goal of our routines is to convert these images to a list of digital information which can be processed to give us the dimensions and orientation of the ellipse which best approximates the perimeter of each track. Since all of the necessary information about each track (major and minor ellipse axes, and ellipse orientation) can be derived from its perimeter points

and since these points constitute a smaller and more manageable set of data than the pixels that make up the ellipses area, we have concentrated on developing edge enhancement operations which allow the track perimeter to be identified and segmented from the background of the image. An alternative scheme, which involves the segmentation of the ellipses by using an optimal threshold selection (on the basis of a histogram of the entire image) was attempted in the early phases of the development of our system. This method was not considered practical because a) the tracks are not uniformly dark, and therefore a single threshold grey level value used to select the track from background will not faithfully reproduce the extent of the track, and b) the result of the thresholding is a list of pixels which make up the interior points of the track ellipse, and, as mentioned above, this is a larger set of pixels than are needed to evaluate the ellipse parameters. Furthermore, algorithms used to select the edge points from this set of pixels are computationally expensive, i.e., these algorithms can not be implemented as spatial operations on a single image.

There is a considerable literature on the subject of detecting edges in machine vision (for reviews see Davis 1975; Peli and Malah 1982; Levaldi 1983) and a large class of these detectors fall in the enhancement/threshold category. In these edge detectors, discontinuities in an image attribute are enhanced or accentuated by some spatial operator. If the enhanced discontinuity is sufficiently large, that is, greater than some threshold level, an edge is deemed present (figure 3). This type of edge detection scheme particularly is desirable from the point of view of the hardware we have available, and from the point of view of the software required for feature-list extraction and subsequent analysis. As mentioned previously, our image processing facilities include the capabilities for point and convolution operations to be performed quickly, and the list of pixels that must be analyzed as a result of the enhancement/threshold operation ellipse parameters is relatively small. An obvious choice for the spatial operator is a derivative or second derivative operation. Discontinuities in the image intensity appear as a peak in the grey level at the location of the edge.

An alternative method of detecting edges in images has been proposed by Marr and Hildreth (1980) and Marr, Ullman, and Poggio (1979) as a possible mechanism for the elucidation of the 'raw primal sketch' in early visual systems. This scheme, which is presumably implemented neurally in the visual system, uses a two-dimensional extension of a result known as Logan's theorem (Logan 1977). It states that the zero crossings of a band-pass filtered function can be used to determine the original function up to a multiplicative constant, provided the function satisfies some conditions for being "well-behaved". In this method, which is discussed more fully in appendix A, the image is convolved with a filter which combines a Gaussian smoothing operation with a Laplacian filter. The result of this sequence of operations is an approximation of a one octave band-pass filter. The zero crossings are then detected and identified with edges. While the theoretical motivations for this method of edge detection are not entirely sound, it is attractive because it does not rely on a threshold and is therefore applicable to images which contain large contrast variation. Also, the Laplacian filter is spherically symmetric, and the edges which are extracted do not show the directional dependence apparent in some

edge detection schemes that involve directionaly dependent spatial derivative operators.

The edge detection algorithm that we have implemented is an enhancement/threshold operation which incorporates the ideas presented by Marr et. al. Although we have not implemented the zero crossing detection that this procedure suggests, the thresholding scheme that we use for determining the gray level peaks in the second order derivative of the smoothed image is adequate for the purposes of interest, while remaining easy to implement in software. We have not used convolutions which are identical to those of Marr et. al., but instead have chosen a set of 3X3 convolutions which correspond to a spatial smoothing filter followed by a Laplacian operation. The necessity of the smoothing becomes apparent by studying the results of a Laplacian operation on an image with and without smoothing. This smoothing tends to eliminate noise in the image which has a scale smaller than the size of the 3X3 pixel mask, such as digitization errors, or very small scale imperfections in the plastic surface, while preserving large scale features, such as track edges (see appendix A). This reduces the background noise in the thresholded image, and simplifies the task of segmenting tracks from this background.

Our first attempt at using the ideas presented above used the convolution masks shown in figure 4, followed by a gray level threshold to produce a binary edge map in image memory. The image containing this map was then searched for the occurrence of an "on" pixel, and a list of all 8-neighbor continguous pixels was found (using a depth-first search algorithm) and stored in a linked list structure, which was then used to compute the ellipse parameters. This algorithm worked well if the track edge was well defined, but failed for certain commonly occurring track imperfections. This tended to break the continuity of the track, making it difficult to identify contiguous segments of pixels with single tracks.

The next step in the evolution of the edge detection algorithms was the implementation of a threshold/tracking detector (appendix B). The program is conceptually very simple. After the smoothing/Laplacian operations have been performed on an image, the pixel rows are scanned for the occurrence of a single pixel above a certain gray level. After a pixel has been found, its position is recorded on a stack, and a list of adjacent pixels are scanned for a value above some second threshold. If a pixel is found that satisfies these conditions this pixel is stored a linked list of track candidate pixels. If more than one pixel is found above threshold, only the one with the highest gray level is accepted. This procedure then continues at the location of the pixel most recently found. The search ends when a) the location of the next pixel is equal to the start position of the search, in which case the linked list is terminated, and a track is assumed to be found, or b) the search reaches a "dead end", that is, the start position has not been found, and no adjacent pixels satisfy the threshold condition for the search to continue. In this latter case, the linked list of pixels for the candidate track is deleted from memory. Thus each track has a list of pixels which forms a continuous ring, which helps to remove from consideration edges formed by objects which are not tracks (an example is a surface scratch in the plastic which appears as a long straight edge). A track candidate is then tested to make sure that the number of pixels found lies above some certain value; if the track does not pass this cut the linked list is also deleted from memory. Control is then returned to the section of

the program that scans image memory for pixels above threshold. In practice, to speed up this operation, only select rows of the image are scanned; the spacing between these rows are chosen such that they are smaller than the size of the objects of interest.

In the process of performing this search the x and y coordinates of the pixels are summed to provide first-order moments for calculating the centroid of the track. All of this procedure is written in optimized assembly language to provide the fastest possible speed.

After the entire image has been scanned in the manner described above the linked lists of candidate tracks is passed to a PASCAL routine which does the subsequent processing. The primary reason for using the linked list as opposed to an array structure is that the linked list is more flexible when dealing with pixel lists of varying size. For example, if an array structure were used, the arrays would have to be made large enough to accommodate tracks viewed at high power, while at low power, most of this memory would be wasted. Also, many of these arrays would have to be used to store the results at low power, when a large number of tracks may appear in an image. When measuring at high power, on the other hand, in general only one track is of interest to the analysis. The linked list structure is easy to implement in assembly language; this type of data structure appears in PASCAL, and the entire structure can be constructed in assembly language and accessed from PASCAL, as necessary.

The parameters of interest in the surface analysis of the tracks are the major and minor axes of the ellipse forming the track mouth, and the orientation of this ellipse. These quantities, along with the etch rate of the bulk surface of the plastic, v_G, suffice to determine the charge of the particle, given that the particle is relativistic.

These parameters are calculated using a least-squares fit to the pixel list. The method is given in appendix C. The parameters are calculated from the ellipse moments:

$$M_{mn} = \sum x^m y^n$$

where x and y are the x and y coordinates of a pixel and the sum runs over each pixel in a track. The derivation of the ellipse parameters from the moments for the most general case is quite complicated and requires 15 moments to be calculated. In practice we use an approximation to this general solution, which requires only 10 moments for all the parameters (major and minor axis, orientation angle) to be calculated. The relative complexity of the 10 and 15 moment solutions can be inferred from appendix C, where the solutions to the problems for these two cases have been given using the the algebraic manipulation program REDUCE. The relative simplicity of the 10 moment solution (requiring the solution of a 3X3 system of linear inhomogeneous equations, compared with a 5X5 system for the 15 moment case) is offset by the fact that the centroid of the ellipse must be determined precisely for the fit to be accurate in this approximation. Since the centroid is determined using the M_{10} and M_{01} moments of the ellipse, the determination of the centroid makes it essential that the edge segmentation algorithm find the *entire* edge of the ellipse, and that this edge have uniform thickness. A portion of the edge of an ellipse, for example,

would give a bad fit in the 10 moment case, since the centroid of the full ellipse would not be accurately determined by the moments derived from the portion of the ellipse found. The threshold/tracking algorithm that we have implemented satisfies both of these constraints: the edge is of single pixel width, and the completeness of the edge is assured by the requirement that the last pixel in the search be the pixel from which the search was started.

In order to increase the speed of the analysis, none of the parameter calculations are done on the VICOM at low power. Instead, the moments are summed and written directly to disk, along with the sheet x and y coordinates of the track. Each image is also assigned a number and each track candidate within the image assigned a unique 'event number'. These moment files are then transferred to the microVAX, where ellipse parameters are calculated and subsequent analysis is performed. A problem with this approach is that it does not allow cuts on the data to be made in real time based on chi-square test or track radius. Thus the output of the 'low power' scans tend to large (typically 10,000 to 100,000 events), because of the number of surface imperfections that are initially identified as track candidates. This has not been considered to be a serious drawback, however, since most of the surface imperfections (low energy α-particle tracks, and 'bubbles' in the plastic surface) are of a spherical or elliptical shape and can not be cut from the data set on the basis of chi-square test of the fit alone. Also, the cut on pixel number, while not providing as convenient a way to eliminate undesirable features in an image as a cut on minor axis, does provide a way to place some cut on object size. The cuts on minor axis can be done later on the microVAX, as the analysis requires.

4. Analysis Procedures

As mentioned earlier, the scan is divided into two parts, a 'low power' scan (typically 10X), in which the surface of the plastic is scanned rapidly and rough calculations are made of the ellipse parameters, and a 'high power' scan (typically 50X), in which the track parameters are measured accurately. Given these two magnifications, the difference in the accuracy of the determination of ellipse major and minor axes is easily estimated from the edge position resolution at 10X and 50X. At 10X, each pixel corresponds to 2.5 μm, and the number of pixels in a typical track is approximately $N = 30$. Thus while the position of a pixel only specifies a position to $\pm 1.25\mu$m, the position of the edge can be given with an uncertainty $\sigma_{Edge} = 1.25/\sqrt{N}$ or 0.23μm. A similar calculation applies for tracks measured at 50X. Here the micron/pixel scale factor is approximately 0.5, and the number of pixels per track is about $N = 400$ for most of the samples we have analyzed, so $\sigma_{Edge} = 0.5/\sqrt{N}$ or roughly 0.03μm. While these values for σ_{Edge} can be realized in practice if successive measurements are made while the track is viewed at one position under the camera, the actual resolution that we obtain is limited by the effect of image distortion, ($\pm 2\%$ over the entire field of view). The accuracy in determining the major or minor axis is roughly $\pm 0.2\mu$m at 50X.

The image distortion also must be taken into account when calculating the sheet positions (in μm) from the centroids of the objects calculated in pixel coordinates during a low power scan. At 10X, at typical low power scan magnification, the field of view is $1200\,\mathrm{X}\,950\mu\mathrm{m}^2$, so that a 2% effect is roughly 20μm over the entire field of view, a significant fraction of an average track dimension. A linear micron/pixel scale factor is thus not sufficient to accurately give the sheet positions of tracks given their centroid x and y pixel coordinates. In order to eliminate this effect, we have developed a program to map the pixel coordinates to sheet coordinates using a table lookup. This is done by raster scanning a single track across the screen and tabulating track centroid pixel coordinates for a given stage position. The centroid of the track can be estimated to sub-pixel accuracy (see the arguments above); the sheet coordinates are limited by the accuracy of the position measuring apparatus to $\pm 1\mu$m. After this table is accumulated, the results are inverted using a bilinear interpolation scheme to give the sheet coordinates for each point on a rectangular grid of pixel coordinates. A typical grid dimension is $10\,\mathrm{X}\,10$. This grid is stored as an array in the analysis program and serves as a look-up table for calculating the sheet position of arbitrary pixel values, again using a bilinear interpolation process. Using this method we have reduced the effects of camera distortion to the approximately .5% level over the viewing area.

While this look-up table could be used in principle to calculate the sheet coordinates for every pixel in a track, this has not been implemented in our system. In general, the tracks do not take up a significant area on the image, and, at high power, the tracks are well centered so that the micron/pixel scale factor is roughly constant over the area of interest, and a scale factor is used to calculate the sheet coordinates of a given pixel. Since the scale factors are different for the x and y directions (due to the fact that the pixel aspect ratio is 4:3 for the RS-170 television standard), two scale factors must be

calculated. This is done in a manner similar to that described above. The average values for the x and y pixel/micron conversion factors are calculated for a number of points in a region around the center of the image by moving a single track in a raster scan across the screen. The position in pixel space of the centroid of the track is compared with the stage position, and the average of the micron/pixel scale factor thus deduced is used in subsequent calculations.

The analysis of the plastic begins with a scan at low magnification of both sides of a sheet of plastic. This is a raster scan of an entire sheet of plastic (or a region selected interactively from the menu), where each 'frame' (the image at a particular location in the scan) is processed to give sheet coordinates and approximate ellipse parameters for all potential tracks identified. Adjacent frames overlap one another by an amount which is adjustable from the menu, so that track candidates which lie on the boundary of a particular frame will not be excluded.

The focus is maintained by a Leitz laser autofocus system. For the low power scan, the autofocus is not under control of the computer and no analysis time is devoted to this task. This gives an improvement in speed over an earlier system which used a 'focus map' to determine the appropriate focus for each point on the sheet, by interpolating from the z-axis position at focus for a grid of (x,y) coordinates on the sheet. Also, this focus map had to be constructed each time the sheet was placed on the stage. The autofocus is switched on at the beginning of the scan by a control keypad available to the operator.

The images are first analyzed using a simple gray level threshold to discard images which are unlikely to contain tracks. The threshold is selected such that the gray levels of the majority of surface defects or non-penetrating tracks, which are shallow and therefore not as dark in reflected light as tracks will well developed etch cones, will be below the chosen threshold level. The result of this threshold is a binary image (all pixels either 0 or 1), with the value 1 corresponding to interior points of possible tracks. The number of pixels with the value 1 ('on' pixels) is then used as a cut to determine whether the image requires further processing. This counting is done by histogramming the gray levels of the binary image, the contents of the '1' bin in the histogram being the number of pixels that correspond to possible track interior points. If the number of 'on' pixels after thresholding is above a certain value, the image is processed further to segment the edges of the track candidates; otherwise, the image is discarded and the next frame in the scan is digitized and processed. Because the thresholding and subsequent histogramming can be done very fast in hardware (see section 2), this method provides a fast method of skipping uninteresting frames in the low magnification scan, if the density of tracks is low enough that there are a considerable number of frames without track candidates, and if the surface is smooth enough that most of the uninteresting features can be eliminated by a simple thresholding process. A cut is also placed on the maximum number of pixels that a frame may contain and still be considered for processing. This allows the program to quickly skip over areas, such as the mounting holes for the plastic sheets, without subjecting them to edge analysis.

In practice the threshold level and the pixel number cut are determined by viewing

the scanned images on the screen and adjusting these parameters interactively, so that the algorithm detects most of the events considered interesting by the operator. In some cases of eliptical tracks, the result of thresholding may be to only detect some small portion of the interior of the track, and the pixel number cut must be chosen low enough so that this possibility is taken into account.

If the frame passes these pixel number cuts, it is subjected to edge analysis discussed previously (see also appendix A). The only real-time cut placed on the objects found is a minimum pixel number cut, and again this can be set interactively from the menu program to select tracks appropriate to the analysis. In this way small imperfections in the plastic which are segmented by the threshold/tracking routine as possible tracks are discarded from further processing. The pixel lists and zero-order moments are collected in the threshold/tracking routine, and the linked list of pixel values is passed to PASCAL, where the appropriate moments are summed and written to disk, along with the track sheet coordinates, frame number and event number. After the scan of the entire sheet is finished, this file is sent via a DMA link to be processed on the microVAX, where ellipse parameters (major axis, minor axis, and angle relative to the given sheet coordinate system) are calculated. The files are then searched for duplicate events that may arise from the overlap between frames. We have developed a very fast algorithm for duplicate removal, this is discussed in appendix D.

The number of tracks which can are classified as 'noise' can be significantly reduced by requiring that each event be matched with a corresponding event on the opposite side of the plastic, so that non-penetrating tracks and surface imperfections are eliminated. This requires that the tracks appearing on the top surface be projected onto the bottom surface, and that these tracks be matched to a track found on the low magnification scan of the bottom surface. Zenith angles for tracks are easily calculated using a well known formula (Somogyi 1980), which depends on the major and minor axes, and the general etch rate of the plastic, v_G. Using this information, the sheet thickness and the calculated azimuth angle, the trajectory of the particle through the plastic can be reconstructed. Because of the large number of events that need to be handled, an N^2 algorithm for matching would be wasteful of computer resources. We have therefore written a matching program which is based on $N\log N$ sorting and a fast search routine (Press et al. 1986). The events from the bottom surface scan are sorted using the $N\log N$ sorting routine 'heapsort' (Press et al. 1986) by increasing y coordinate. These sorted events are then written to a random access file. For each event in the top data file, the trajectory through the plastic is reconstructed, and the position of the x and y coordinates of the event on the bottom of the sheet is calculated. Since, for a given zenith angle, the azimuthal angle not unambiguously specified (the azimuthal angle for a given ellipse orientation could be given by θ or $\theta + \pi$), the coordinates that must be searched for a match are in two x and y regions on the lower surface. A search by bisection is performed to locate the track with the lowest y value within one of these calculated regions in the random access file. and the x,y parameters are searched sequentially for until a match is found, or until the highest possible y value consistent with the search regions specified above has been found. Since both the sorting

routine and the search routine run in a time which is proportional to $N\log N$, the entire routine runs in essentially $N\log N$ time, and a dramatic time saving is achieved over the 'brute force' N^2 algorithm. The match is further constrained by placing cuts on major and minor axes and azimuth angle, except for tracks with incident angles of approximately 0°, where the azimuth angle is not specified. Good results have been achieved at 10X by requiring that the major and minor axes for the events to be matched agree to within 1μm, and the azimuth angle to within 10°, with a search area of $50\,\mathrm{X}\,50\mu\mathrm{m}^2$ from the calculated point. For these values, more than 75% of the tracks that pass the cuts are 'good' events.

The next phase in the analysis process is the high magnification scan. The files produced by the matching algorithm are loaded onto the VICOM via the DMA link and used to drive the microscope to the sheet locations of track candidates. The edge segmentation operation is essentially the same as that for the low power scan (see section 3). The number of tracks to be processed is generally much less than the number processed in a high power scan; the track parameters are calculated on the VICOM, and written directly to disk. This allows us to perform additional cuts on the measured tracks. As of now the only cut that is available is track minor axis, a 'goodness of fit' cut has not yet been implemented.

For the high magnification scan, the autofocus is controlled by the VICOM microcomputer. The autofocus is switched on while the stage is moving to a track coordinate, and switched off prior to stopping. This prevents the focus from being lost as the track comes into the field of view, and the laser spot is incident on the surface of the track cone. The time required for the stage to move from one sheet coordinate to another is easily calculated given the velocity profile of the stage. The Compumotor indexers, which send TTL control signals to the autofocus via available programmable output bits, can be instructed to delay processing of the autofocus commands via an ASCII string sent from the VICOM microcomputer. In this way the appropriate delay can be generated before the autofocus is disabled upon reaching a track coordinate.

The time required to complete a high power scan is currently limited by the stage movement between successive screen coordinates. The problem of minimizing the travel time between a number of points on a surface has been well studied, and an approximate solution based on 'simulated annealing' is given in Press et al. (1986). Because, as a result of our matching method, the high power scan files are sorted according to y coordinate, this travel time is certainly not optimized. We plan to implement some sort of solution to this problem in the future. Our high magnification scan routine works either automatically, or with operator assistance. In the latter case, the movement of the stage to successive track locations is initiated by a command sent from the menu task. The operator can also choose to analyze a track via another menu call, and also to record the track parameters in a file. This method of control has proved useful in debugging our analysis routines.

5. System Performance

We have tested the performance of the track analysis system on CR-39 data obtained in accelerator experiments and on a balloon flight. An attempt has been made using the accelerator data to determine the charge resolution that can be obtained with the system, and to investigate systematic variations in track measurements with sheet position and other effects. A full description of these experiments has been written (Dowell 1987), and a brief discussion of the results will be given here. The full analysis of the balloon data is still in progress, and here a discussion of the results obtained so far will be given.

The accelerator exposures were done at the Berkeley Bevelac to a a $1.7A$ GeV ^{56}Fe beam, using a stack of 6" X 6" X 0.072" CR-39 sheets with an air-gap of 1.13 cm. The stack was composed of repeating modules of two sheets in contact, followed by the air gap, for a total of 110 sheets. These sheets were etched at 40 ° C in 6.25M NaOH for 480 hours. The resulting Fe primary tracks were approximately 50μm in diameter. A 2" X 2" region of the 37th sheet in the stack was scanned and approximately 19,000 objects recorded. The scan took roughly four days to complete. A cut was placed on the iron peak (R_{Fe}= 24.3μm), so that other peaks could be easily identified. Using the semi-automatic measuring program, the fronts of sheets 36 and 38 and the back of sheet 38 were scanned at high magnification, using the coordinates from the low power scan of sheet 37. The operator was responsible for identifying tracks and eliminating bad events from the data file. These were mainly surface imperfections which produced shallow circular pits with a characteristic bright spot in the center, and were easily discarded from the data by inspection of the image on a video screen. The data from the three surfaces was correlated and and charge changing interactions eliminated from the data set. After these cuts the data set consisted of 951 events. The histograms of these surfaces are shown in figure 5-7. A histogram of average radius measurements, figure 8, was used to determine the charge resolution of the CR-39. In order to identify the charge peaks in the histograms correctly, a calibration of the Fe tracks was made, since the cut on the low magnification data set was severely limited the number of Fe and Mn tracks which appeared in the high magnification scans.

The charge resolution is given by $\sigma/\Delta R$, where σ is standard deviation of a single charge peak, and ΔR is the separation between two adjacent peaks in figure 8. The charge resolution was calculated for each peak, $12 \leq Z \leq 24$, and the mean charge resolution was determined to be $\sigma_Z = (0.211 \pm 0.0012)/\sqrt{n}$, where n is the number of scanned surfaces. This is about the same resolution reported by other authors (Gerbier, Williams, Price and Guoxiao 1987) for CR-39(DOP). Figures 9-11 are plots of R_1 vs. R_2 where R_1 and R_2 represent radii measurements of an individual non-interacting track on two different sheets. The individual track measurements differed by less than 1.5μm between two surfaces.

A study was also made to try to discover any systematic differences in the radii of the tracks as a function of sheet position. The sheet was divided into 3" X 3" regions (9 total), and the average radius for each charge species in each each region was calculated. The weighted distribution of radius variation for each region, normalized to an arbitrary region (which we labeled with the number 9) was calculated using:

$$W_i = \sum_{Z=12}^{24} \left(\frac{R_{Z_i}}{R_{Z_9}}\right)\left(\frac{N_{Z_i}}{N_{i_{tot}}}\right)$$

where R_{Z_i} is the average radius for charge Z in region i, R_{Z_9} is the average radius for charge Z in region 9, N_{Z_i} is the number of radii with charge Z in region i, and $N_{i_{tot}}$ is the total number of tracks in region i. The weighted distributions for each region differed by less than 0.5% from region 9. This result implies that variations in track measurements with respect to position is negligible.

The above measurements were done with a system which in which most of the track segmentation and fitting routines were written in PASCAL. Since then, in an attempt to improve the speed of the analysis, much of the segmentation routine has been written in assembly language, and the fitting done on the microVAX from the raw moment data derived on the VICOM. We have also developed the software for matching tracks on adjacent surfaces of the sheets, and the software for the autofocus to allow us perform high power scans automatically. The performance of the improved system has been tested on data collected during a balloon flight. 2080 sheets (6" X 6 " X 0.010") of CR-39(DOP) were flown from Prince Albert, Saskatchewan, for 5.5 hours at a float altitude of 120,000 feet. The plastic was etched for 504 hours in 6.25M NaOH. Subsequent analysis was performed on the sheets at low magnification to determine the location and approximate major and minor axes of the events. A typicai scan provided lists of 60,000 event candidates. Each side of the sheet took approximately 24 hours to scan at 10X magnification, this time depending somewhat on surface quality and number of track candidates found. The matching procedure between the front and back surfaces of a single sheet yielded roughly 600 events. The cuts placed on the data were: 1.0μm on major and minor axis, and 8 $^\circ$ on azimuth angle. The search area was set equal to 3 times the major axis of the ellipse. The number of events found was not highly dependent on this parameterization, and it was chosen because the events at large zenith angle showed larger deviations from calculated positions than those with smaller zenith angle. Figure 12 shows a histogram of b (minor axis) for one particular sheet. This values for b must be corrected for the deviation in the center of the etch cone from the center of the track mouth (a function of zenith angle) to provide information on (Z/β). A more convenient way to study this is to plot the ratio of track etch rate to bulk etch rate, $s = v_T/v_G$ which gives a unique value for (Z/β) independent of zenith angle. This is shown in figure 13 for the data of figure 12. The value of s shows a sharp peak around the value of 1.5. This corresponds well with the value of s found for relativistic iron in a previous study (Tarle et al. 1981). A plot of θ is shown in figure 14. Ideally, the distribution of tracks in theta should have a peak at around 40 $^\circ$, while our data shows a peak at less than 30 $^\circ$. The discrepancy can be attributed to a low detection efficiency at large θ, since the tracks at large angle of incidence tend to be less dark than those near $\theta = 0$ and are likely to be missed by the gray level threshold used to pick 'interesting' frames. This is even more likely for tracks with large zenith angle and low (Z/β).

The charge resolution in this experiment is ultimately to be determined by matching the events in the plastic with electronic data from the to give positive identification of iron events, and measuring the events found at low power with a high magnification scan. This work is in progress. Note that a cut on low β events is implicitly performed by the requirement that the major and minor axes differ by no more than 1.0μm, but the exact parameters for this cut have not yet been quantified.

Appendix A: Edge Detection Algorithms

The first stage in the segmentation of an edge concerns the detection of intensity changes in an image. The general idea behind this detection is that a sudden intensity change in an image will give rise to a peak in the first and second derivative, or a zero crossing in the second derivative of an image. Marr and Hildreth (1980) have argued that the most satisfactory method for detection of intensity changes is to locate the zero crossings in an image convolved with the operator $\nabla^2 G$ where ∇^2 is the two dimensional Laplacian operator and G stands for the two dimensional Gaussian distribution

$$G(x,y) = e^{-\frac{x^2+y^2}{2\pi\sigma^2}}$$

which has standard deviation σ. $\nabla^2 G$ is a circularly symmetric Mexican hat-shaped operator, whose distribution in two dimensions may be expressed in terms of the radial distance r from the origin by

$$\nabla^2 G = \frac{-1}{\pi\sigma^4}\left(1 - \frac{r^2}{2\sigma^2}\right)e^{\frac{-r^2}{2\pi r^2}}.$$

Where I is the raw image data. By the convolution theorem, $(\nabla^2 G) * I = \nabla^2(G * I)$, so this process can be thought of as the convolution of I with the operator $(\nabla^2 G)$, or equivalently, as the operation of ∇^2 on the smoothed image $(G * I)$.

There are two basic ideas behind this choice of filter. The first is that the Gaussian part of it, G, blurs the image, wiping out all structure on scales much smaller than the space constant σ of the Gaussian. If the σ is chosen small enough, the effect is to eliminate much of the noise in the image that is due to small scratches in the plastic, or single 'bright' pixels. Marr and Hildreth chose a Gaussian, instead of a pillbox or some other smoothing function, because it is a smooth function localized in both the spatial and frequency domains, and as such it is unlikely to introduce changes that were not present in the original image.

The second idea concerns the derivative part of the filter, ∇^2. In many image processing applications, first order derivatives are used to detect edges in some combination like

$$|\nabla I(x,y)| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$$

where $I(x,y)$ is the intensity at the point (x,y) in the image, and the partial derivatives can be represented by the differences:

$$\frac{\partial I}{\partial x} = f(x,y) - f(x-1,y)$$

$$\frac{\partial I}{\partial y} = f(x,y) - f(x,y-1)$$

The above method is computationally very expensive, and some means must be found of computing the square and square root operation on an entire image. The following

approximation to the above is generally used:

$$| \nabla I(x,y) | = | \frac{\partial I}{\partial x} | + | \frac{\partial I}{\partial y} |$$

However this approximation is strongly directionally dependent, and thus some edges with a given orientation (notably along the x or y direction) will be enhanced more than others. Thus it becomes difficult to define a threshold which will give good results in selecting the edge from background.

The solution of Marr and Hildreth to the problem of simplifying the computation and eliminating directional dependence is to use the spherically symmetric second derivative operator (∇^2). In their approach, the edges are then chosen by detecting the zero crossings of the resulting image. This has the advantage over the threshold method in that it is applicable over a wide range of contrasts. The optimal threshold selection in an enhancement/thresholding scheme is generally sensitive to the edge contrast and if this contrast varies widely no one threshold value may be appropriate. On the other hand, the existence of a zero crossing does not depend in any way on the contrast of the edge.

We used a method of extracting edges which is based on Marr and Hildreth's but differs from theirs based on the limitations of our hardware and the time allowed for processing. We have chosen the same set of operations $(\nabla^2 G) * I = \nabla^2 (G * I)$ but represent G and ∇^2 by 3X3 convolutions, these are shown in figure 4. The smoothing operation, figure 4b defined over a 3X3 area, eliminates noise occurring over these scales. The Laplacian mask, figure 4a, is the discrete approximation to a spherically symmetric function. We have chosen to detect the maxima of this function, instead of zero crossings, since the contrast in our image does not vary over a wide range, and the illumination is uniform. The threshold detection algorithm is described in Appendix B.

It is interesting to note that a recent theorem of Logan (1977) states that a one octave bandpass function can be reconstructed, up to a multiplicative constant, by a knowledge of its zero crossings alone. While the $(\nabla^2 G)$ function is certainly not a pure one-octave bandpass filter (its bandwidth at half power is 1.25 octaves for the parameters used in the paper), it is nevertheless evident that the zero crossings of a band limited function are rich in information. This theorem forms the basis for a theory of the early visual system (Marr and Hildreth 1980).

Appendix B. Threshold/Tracking Algorithm

The threshold/tracking algorithm that we use to locate edges from the grey level maxima in the laplacian image has been briefly described in section 3. Figure 15 shows a simplified flow chart for this process. The smoothing/laplacian algorithm produces an image which has both negative and positive peaks in the gray level at the location of edges. Figure 16 shows the result of a Laplacian algorithm on an ideal edge. The threshold tracking algorithm scans the rows of image memory for a negative gray level value below some threshold, t. When a pixel which satisfies this criteria has been found, the eight-neighbor adjacent pixels are searched to find a neighbor with the lowest gray level value below some second threshold, t'. If one is found, it defines the starting direction for the search. From the most recent pixel found, 3 neighboring pixels are searched for a grey level below the value of t'. Figure 16a and 16b show the pixel search list for two given directions. Pixel 2 represent the most recent pixel found in the search, and pixel 1 is the pixel found previous to pixel 2. The pixel search list is defined such that the 3 pixels searched are those most nearly forming a straight line with the direction defined by the pixels 1 and 2. This prevents the search from tracing back over itself, and breaking the continuity of the track at locations which the gray level minima form 'blobs' on the track edge. Each successive pixel found in the search defines a new list of pixels to be searched for an edge pixel, on the basis of its location and the direction it defines with the previous pixel. Figure 17c shows how this search process proceeds in relation to pixels which make up the track edge to be found. The search continues until a) none of the pixels in the search list satisfy the criteria for an edge pixel (gray level less than t'), or b) one of the pixels in the search list is the pixel with which the search began. Condition b) is to assure that the track found represents the complete edge of an ellipse. If condition a) is satisfied, it is assumed that the edge found is not a track edge, and the list of 'found' pixels is deleted from memory. This eliminates unwanted edges found from consideration as possible track candidates. One example of this type of edge is that produced by a scratch in the plastic, which appears as a long straight line of pixels in the processed image.

While the condition that the entire track edge be found may seem difficult to assure, in practice the track edges in CR-39(DOP) are sufficiently well defined that the search procedure outlined above gives good results even with tracks produced at large zenith angle (see figure 14) where the contrast variation between the edge of the ellipse and the surface of the sheet is very small. Certain track imperfections, caused by dust or 'bubbles' in the plastic surface may cause the search to lose the edge and only a portion of the track will be found; by condition b) above the track is then discarded from the analysis. However, from the known density of tracks in the plastic thus far analyzed, the effect is known to be less than 10%.

The remainder of this appendix is a code listing for the threshold/ tracking algorithm, as implemented on our system. This code can be run on any system based on the Motorola 68000 microprocessor, with appropriate changes in the system dependent definitions listed at the beginning of the program.

```
BUZZ      IDNT      1,1
          NOPAGE
          XDEF      BUZZ,PREBUZZ
          SECTION   1
*
* BUZZ12 incorporates both modifications of BUZZ7 and BUZZ8. This is
* the algorithm which is currently being used. DC 12/9/88.
* Added MAXPIXEL routine 12/20/88 DC.
*
* PROCEDURE BUZZ(MINPIXEL,MAXPIXEL : WORD; SPARSE : WORD; VAR NO_OBJECTS : WORD
*                VAR PIX_NO_LIST : NO_ARRAY; VAR PV : PTR_ARRAY;
*                VAR RSUM,CSUM : INT_ARR);
*
* TYPE POINTER1 = ^PIXEL_REC;
*      PIXEL_REC = RECORD
*                    COORD : ARRAY[1..2] OF WORD;
*                    NEXT : POINTER1;
*                  END;
*      PTR_ARRAY = ARRAY[1..500] OF POINTER1;
*      NO_ARRAY = ARRAY[1..500] OF WORD;
*      INT_ARR = ARRAY[1..500] OF INTEGER;
*
* ROW AND COLUMN VALUES GO FROM 0 TO 511
* CURRENT REGION OF INTEREST : ROWS (5 - 465)
*                            : COLUMNS (36-455)
*
* START ADDRESS IS $800000+(INITIAL ROW)*1024+(INITIAL COLUMN)*2
* BINARY IMAGE IS ASSUMED TO BE IMAGE #0
*
* PREBUZZ(INIT_ROW,INIT_COL,FNAL_ROW,FNAL_COL,NUM_COL,CCOUNT,RCOUNT : WORD;
*         STRT_ADD,Z1_BEG,Z2_BEG : INTEGER)
*
* PREBUZZ LOADS THE FOLLOWING MEMORY LOCATIONS:
*
* NUM_COL   = FNAL_COL-INIT_COL
* CCOUNT    = FNAL_COL-INIT_COL+2
* RCOUNT    = FNAL_ROW-INIT_ROW
* STRT_ADD  = $800000+INIT_ROW*1024+INIT_COL*2
* Z1_BEG    = $800000+(INIT_ROW-1)*1024+(INIT_COL-1)*2
* Z2_BEG    = $800000+(INIT_ROW)*1024+(INIT_COL-1)*2
*
* (PREVIOUS VALUES USED HAVE BEEN:   INIT_ROW=5,INIT_COL=36
*                                    FNAL_ROW=465,FNAL_COL=455)
*
PIXELVAL  EQU       $01
INIT_ROW  DC.W      0
INIT_COL  DC.W      0
FNAL_ROW  DC.W      0
FNAL_COL  DC.W      0
NUM_COL   DC.W      0
CCOUNT    DC.W      0
RCOUNT    DC.W      0
STRT_ADD  DC.L      0
Z1_BEG    DC.L      0
Z2_BEG    DC.L      0
*
STACK1    DS.L      200
STACK2    DS.L      200
*
HEAPBEG   DS.L      50000
HEAPEND   EQU       *
          DC.L      0
HSBASE    DC.L      0
ACCUM     DC.W      0
RETPTR    DC.L      0                  * RETURN POINTER TO PASCAL
ARRPTR    DC.L      0                  * POINTER TO ARRAY AREA
```

```
OBJPTR    DC.L      0
NODEPTR   DC.L      0
MINPIXEL  DC.W      0
MAXPIXEL  DC.W      0
EDGFLG    DC.W      0
NUMUPD    DC.W      0
DIRECT    DC.W      0
EDGMARK   DC.W      0
NEWADD    DC.L      0
INITDR    DC.W      0
R_SUM     DC.L      0
C_SUM     DC.L      0
CLOSED    DC.W      0
DEADFLG   DC.W      0
CHKADD    DC.L      0
JTABLE    DC.L      DIR1                * JUMP TABLE FOR SEARCH ROUTINES
          DC.L      DIR2
          DC.L      DIR3
          DC.L      DIR4
          DC.L      DIR5
          DC.L      DIR6
          DC.L      DIR7
          DC.L      DIR8
          SECTION   9
PREBUZZ   EQU       *
          MOVE.L    (A7)+,RETPTR
          MOVE.L    (A7)+,Z2_BEG
          MOVE.L    (A7)+,Z1_BEG
          MOVE.L    (A7)+,STRT_ADD
          MOVE.W    (A7)+,RCOUNT
          MOVE.W    (A7)+,CCOUNT
          MOVE.W    (A7)+,NUM_COL
          MOVE.W    (A7)+,FNAL_COL
          MOVE.W    (A7)+,FNAL_ROW
          MOVE.W    (A7)+,INIT_COL
          MOVE.W    (A7)+,INIT_ROW
          MOVE.L    RETPTR,A4
          JMP       (A4)
*
BUZZ      EQU       *
*
* REGISTERS : D0  ROW COUNTER
*             D1  FINAL COLUMN
*             D2  ROW INCREMENT FOR SPARSE SCAN
*             D3  COLUMN COUNTER
*             D4  MEMORY INCREMENT = (D2*1024)
*             D5  LOW ORDER WORD = $0000
*
*             A0  POINTER TO CURRENT ROW
*             A1  POINTER TO CURRENT PIXEL
*
          MOVE.L    (A7)+,RETPTR
          MOVE.L    (A7)+,C_SUM
          MOVE.L    (A7)+,R_SUM
          MOVE.L    (A7)+,A4            * POINTER TO ARRAY OF POINTERS
          MOVE.L    (A7)+,ARRPTR        * POINTER TO ARRAY OF PIXEL NUMBERS
          MOVE.L    (A7)+,OBJPTR        * POINTER TO NUMBER OF OBJECTS
          MOVEQ     #0,D2
          MOVE.W    (A7)+,D2
          MOVE.W    (A7)+,MAXPIXEL
          MOVE.W    (A7)+,MINPIXEL
          MOVEM.L   A3/A5-A6,-(A7)      * SAVE THESE REGISERS ON THE STACK
*
*                                       * NOW SET THE BORDER PIXELS OF THE
*                                       * IMAGE TO #4. PRESERVE A4 AND D2!
*
```

```
         MOVEQ     #4,D7
         MOVE.L    #1024,D6
         MOVEA.L   Z1_BEG,A0
         MOVE.W    CCOUNT,D0
         MOVEA.L   A0,A1
LOOPA    MOVE.W    D7,(A1)+
         DBF       D0,LOOPA
*
         SUBQ      #2,A1
         ADDA.L    D6,A1              * D6 = 1024
         MOVEA.L   A1,A0
         MOVEA.L   Z2_BEG,A3
         MOVEA.L   A3,A5
         MOVE.W    RCOUNT,D0
*
LOOPB    MOVE.W    D7,(A1)
         MOVE.W    D7,(A5)
         ADDA.L    D6,A0
         ADDA.L    D6,A3
         MOVEA.L   A0,A1
         MOVEA.L   A3,A5
         DBF       D0,LOOPB
*
         MOVE.W    CCOUNT,D0
LOOPC    MOVE.W    D7,(A3)+
         DBF       D0,LOOPC
*
         MOVE.L    #10,D0
         MOVE.L    D2,D4              * D2 = NUMBER OF ROWS SKIPPED
         LSL.L     D0,D4              * D4=(D2*1024)
*
         LEA       ACCUM,A6           * CLEAR THE LOCATION FOR NUMBER OF
         MOVE.W    #0,(A6)            * OBJECTS COUNTER
         LEA       HEAPBEG,A6         * A6 IS THE HEAP POINTER
         MOVE.L    ARRPTR,A5          * A5 POINTS TO THE CURRENT
*                                     * POSITION INTHE PIXEL # ARRAY
         MOVEA.L   STRT_ADD,A3        * INITIALIZE REGISTERS
         MOVEQ     #0,D0
         MOVE.W    INIT_ROW,D0
         MOVE.L    #$FF70,D5          * PIXELVAL = VALUE OF "ON" PIXEL
         MOVEQ     #0,D3
         MOVEQ     #0,D1
         MOVEQ     #0,D6
         MOVE.W    FNAL_COL,D1
         MOVE.W    FNAL_ROW,D6
*
*        CLR.W     (A5)               * PIXEL NUMBER ARRAY IS OF SIZE WORD
LOOP1    MOVE.W    NUM_COL,D3         * D3 IS USED TO COUNT 'INNER' LOOP
         ADD.W     D2,D0              * ADD ROW INCREMENT D2 TO D0
         CMP.W     D6,D0              * ROW + INCREMENT > FINAL ROW?
         BHI.S     DONE               * THEN FINISH
         ADD.L     D4,A3              * ADD ADDRESS INCREMENT D4 TO A1
         MOVE.L    A3,A1
         MOVE.L    A3,A2
         ADD.L     #$80000,A2
LOOP2    ADDQ.L    #2,A2
         CMPI.W    #0,(A1)+
         BNE.S     PROCEED
         CMP.W     -2(A2),D5
         BGE       CHECK
PROCEED  DBF       D3,LOOP2
         BRA.S     LOOP1
*
DONE     MOVE.L    OBJPTR,A4
         MOVE.W    ACCUM,(A4)
```

```
        MOVEM.L   (A7)+,A3/A5-A6
        MOVE.L    RETPTR,A4
        JMP       (A4)
*
* REGISTERS : D0 ROW COUNTER
*             D1 COLUMN COUNTER
*             D2 COLUMN COUNTER
*             D3 JUMP TABLE COUNTER
*
*             A0 POINTER TO JUMP TABLE BASE
*             A1 POINTER TO "CURRENT" PIXEL
*             A3 POINTER TO JUMP TABLE ENTRY
*             A4 POINTER TO STACK1
*             A5 POINTER TO STACK2
*             A6 HEAP POINTER
*
SEARCH  EQU       *
        MOVEM.L   D0-D6/A0-A3,-(A7)   * SAVE EVERYTHING ON THE STACK
        MOVE.L    C_SUM,A3
        CLR.L     (A3)                * CLEAR ROW AND COLUMN SUM LOCATIONS
        MOVE.L    R_SUM,A3
        CLR.L     (A3)
        LEA.L     JTABLE,A0
        MOVE.L    A7,HSBASE           * HSBASE=HARDWARE STACK BASE
        SUBQ      #2,A1               * A1 IS THE CURRENT PIXEL POINTER
        SUBQ      #2,A2
        SUB.W     D3,D1               * D1=COLUMN # OF CURRENT PIXEL
        MOVE.L    A1,NEWADD
        MOVE.L    A1,CHKADD
        MOVE.W    D0,-(A7)
        MOVE.W    D1,-(A7)
        MOVE.L    A1,-(A7)
        MOVE.W    #8,(A1)             * MARK THIS PIXEL
        ADDQ.L    #1,(A5)             * INCREMENT NUM OF PIXELS COUNTER
        MOVE.L    A6,(A4)+            * PUSH A NEW NODE POINTER ON
*                                     * THE STACK
        MOVE.L    A6,NODEPTR          * SAVE NODE POINTER IN CASE OF
*                                     * DELETE
        MOVE.W    D0,(A6)+            * PUSH THIS FIRST PIXEL ON THE STACK
        MOVE.W    D1,(A6)+
        ADDQ      #4,A6
        MOVE.L    A6,-4(A6)
*
        BSR       INIT                * FIND AN INITIAL DIRECTION
*
HRESET  MOVE.W    #$FFFF,D4
        MOVE.W    #0,EDGFLG           * RESET EDGE FLAG
        MOVE.W    D0,D6               * SAVE THE INITIAL ROW/COLUMN IN
        MOVE.W    D1,D7               * REGISTERS D6/D7
        MOVE.L    NEWADD,A1
        MOVE.L    NEWADD,A2
        ADD.L     #$80000,A2
        MOVE.W    DIRECT,D3
        SUBQ.W    #1,D3
        MULU.W    #4,D3
SRCH    MOVE.L    0(A0,D3.W),A3       * JUMP TO SEARCH SUBROUTINE
        JMP       (A3)
*
*
*
JMPRET  MOVE.W    EDGFLG,D6           * CHECK IF EDGE PIXEL FOUND
        CMPI.W    #1,D6
        BEQ.S     EDGEH               * BRANCH TO EDGE HANDLER
        MOVE.W    DIRECT,D6           * CHECK TO MAKE SURE SOME PIXEL WAS
        CMPI.W    #0,D6               * FOUND
        BEQ.S     DEADEND             * DEAD END HANDLER IF DIRECTION=0
```

```
*
          MOVE.W    D0,(A6)+            * OTHERWISE WE ARE IN BUSINESS
          MOVE.W    D1,(A6)+            * PUT PIXEL'S ROW,COLUMN COORD. ON
          ADDQ      #4,A6               * HEAP
          MOVE.L    A6,-4(A6)
          MOVE.W    #1,(A3)             * SET BIT IN CURRENT PIXEL
          MOVE.L    R_SUM,A3
          ADD.L     D0,(A3)
          MOVE.L    C_SUM,A3
          ADD.L     D1,(A3)
          ADDQ.W    #1,(A5)             * INCREMENT THE PIXEL # COUNTER
          ADDQ.W    #1,NUMUPD
          BRA.S     HRESET              * ADD THIS CURRENT PIXEL TO STACK
*
*
*
DEADEND   CMPA.L    HSBASE,A7
          BEQ       DISCARD
          MOVE.L    A1,CHKADD
          MOVE.W    #1,DEADFLG
          BRA.S     EDGEH2
*
*
*
EDGEH     MOVE.W    #1,EDGMARK
          CMPA.L    HSBASE,A7           * IS THE STACK EMPTY?
          BEQ.S     SDONE               * BRANCH = END OF SEARCH
EDGEH2    MOVE.L    (A7)+,NEWADD        * POP A CURRENT POINTER OFF STACK
          MOVE.W    (A7)+,D1
          MOVE.W    (A7)+,D0
*
*
*
                                        * FIND A NEW DIRECTION
          CMPI.W    #4,INITDR
          BGT.S     SUBDIR
          ADDI.W    #4,INITDR
          MOVE.W    INITDR,D3
          MOVE.W    D3,DIRECT
          BRA       HRESET
SUBDIR    SUBI.W    #4,INITDR
          MOVE.W    INITDR,D3
          MOVE.W    D3,DIRECT
          BRA       HRESET
*
*
*
SDONE     MOVE.L    #0,-4(A6)           * SEAL OFF THE LINKED LIST
          MOVE.W    CLOSED,D6
          CMPI.W    #1,D6
          BNE.S     DISCARD
          MOVE.W    EDGMARK,D6          * WAS ONE PIXEL AN EDGE PIXEL?
          CMPI.W    #1,D6
          BEQ.S     DISCARD             * BRANCH IF IT WAS
          MOVE.W    MINPIXEL,D6         * IS THE PIXEL COUNT OK?
          CMP.W     (A5),D6
          BHI.S     DISCARD             * BRANCH IF IT ISN'T
          MOVE.W    MAXPIXEL,D6
          CMP.W     (A5),D6
          BLT.S     DISCARD
          ADDQ.W    #1,ACCUM
          ADDQ.W    #2,A5               * INCREMENT AND CLEAR PIXEL NUMBER
          CLR.W     (A5)                * COUNTER
          ADD.L     #4,R_SUM
          ADD.L     #4,C_SUM
          MOVE.L    R_SUM,A3
          CLR.L     (A3)
```

```
          MOVE.L    C_SUM,A3
          CLR.L     (A3)
          CLR.W     CLOSED
          CLR.W     DEADFLG
          MOVE.L    HSBASE,A7
          MOVEM.L   (A7)+,D0-D6/A0-A3
          BRA       PROCEED
*
*
*
DISCARD   MOVE.L    NODEPTR,A6          * REPLACE OLD NODE POINTER
          CLR.W     (A5)                * CLEAR THE PIXEL # COUNTER
          SUBQ.L    #4,A4               * RESET NODE POINTER SP
          MOVE.L    C_SUM,A3
          CLR.L     (A3)
          MOVE.L    R_SUM,A3
          CLR.L     (A3)
          CLR.W     EDGMARK
          CLR.W     DEADFLG
          MOVE.L    HSBASE,A7
          MOVEM.L   (A7)+,D0-D6/A0-A3
          BRA       PROCEED
*
*
*
EDGE      MOVE.W    #1,EDGMARK
          MOVE.W    #1,EDGFLG
          MOVEQ     #0,D3
          BRA       JMPRET
*
*
*
CLOSE2    MOVE.W    #1,CLOSED
          BRA       SDONE
*
*
*
DIR1      EQU       *
          MOVE.W    #0,DIRECT
          MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -2(A1),D5           * IMAGE
          BEQ.S     NEW10                * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -2(A1),D5           * BEEN LOOKED AT
          BEQ       DEADH8
          MOVEQ     #4,D5
          CMP.W     -2(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -2(A1),D5
          BEQ       TST88
NEW10     CMP.W     -2(A2),D4           * NEW MINIMUM VALUE
          BLT.S     TEST11              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -2(A2),D4           * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -2(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          SUBQ.W    #1,D1               * ALTER ROW COLUMN COUNTERS
          MOVE.W    #8,DIRECT
*
*
TEST11    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -1026(A1),D5        * IMAGE
          BEQ.S     NEW11                * NO? CONSIDER IT A NEW CANDIDATE
```

```
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -1026(A1),D5        * BEEN LOOKED AT
          BEQ       DEADH1
          MOVEQ     #4,D5
          CMP.W     -1026(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -1026(A1),D5
          BEQ       TST81
NEW11     CMP.W     -1026(A2),D4        * NEW MINIMUM VALUE
          BLT.S     TEST12              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -1026(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -1026(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          SUBQ.W    #1,D1
          SUBQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
          MOVE.W    #1,DIRECT
*
*
TEST12    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -1024(A1),D5        * IMAGE
          BEQ.S     NEW12               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -1024(A1),D5        * BEEN LOOKED AT
          BEQ       DEADH2
          MOVEQ     #4,D5
          CMP.W     -1024(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -1024(A1),D5
          BEQ       TST82
NEW12     CMP.W     -1024(A2),D4        * NEW MINIMUM VALUE
          BLE       JMPRET              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -1024(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -1024(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          SUBQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
          MOVE.W    #2,DIRECT
          BRA       JMPRET              * RETURN TO SEARCH
*
*
DIR2      EQU       *
          MOVE.W    #0,DIRECT
          MOVE.W    #0,DIRECT
          MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -1026(A1),D5        * IMAGE
          BEQ.S     NEW20               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -1026(A1),D5        * BEEN LOOKED AT
          BEQ       DEADH1
          MOVEQ     #4,D5
          CMP.W     -1026(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -1026(A1),D5
          BEQ       TST81
NEW20     CMP.W     -1026(A2),D4        * NEW MINIMUM VALUE
          BLT.S     TEST21              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -1026(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -1026(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
```

```
          MOVE.W    D7,D1
          SUBQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
          SUBQ.W    #1,D1
          MOVE.W    #1,DIRECT
*
*
TEST21    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -1024(A1),D5        * IMAGE
          BEQ.S     NEW21               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -1024(A1),D5        * BEEN LOOKED AT
          BEQ       DEADH2
          MOVEQ     #4,D5
          CMP.W     -1024(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -1024(A1),D5
          BEQ       TST02
NEW21     CMP.W     -1024(A2),D4        * NEW MINIMUM VALUE
          BLT.S     TEST22              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -1024(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -1024(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          SUBQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
          MOVE.W    #2,DIRECT
*
*
TEST22    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -1022(A1),D5        * IMAGE
          BEQ.S     NEW22               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -1022(A1),D5        * BEEN LOOKED AT
          BEQ       DEADH3
          MOVEQ     #4,D5
          CMP.W     -1022(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -1022(A1),D5
          BEQ       TST03
NEW22     CMP.W     -1022(A2),D4        * NEW MINIMUM VALUE
          BLE       JMPRET              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -1022(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -1022(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          SUBQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
          ADDQ.L    #1,D1
          MOVE.W    #3,DIRECT
          BRA       JMPRET              * RETURN TO SEARCH
*
*
DIR3      EQU       *
          MOVE.W    #0,DIRECT
          MOVE.W    #0,DIRECT
          MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -1024(A1),D5        * IMAGE
          BEQ.S     NEW30               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -1024(A1),D5        * BEEN LOOKED AT
          BEQ       DEADH2
          MOVEQ     #4,D5
          CMP.W     -1024(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
```

```
            MOVEQ     #8,D5
            CMP.W     -1024(A1),D5
            BEQ       TST82
NEW30       CMP.W     -1024(A2),D4      * NEW MINIMUM VALUE
            BLT.S     TEST31            * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    -1024(A2),D4      * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       -1024(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            SUBQ.W    #1,D0
            MOVE.W    #2,DIRECT
*
*
TEST31      MOVEQ     #0,D5             * IS THIS PIXEL MARKED IN BINARY
            CMP.W     -1022(A1),D5      * IMAGE
            BEQ.S     NEW31             * NO? CONSIDER IT A NEW CANDIDATE
            MOVEQ     #1,D5             * MARKED WITH A 1? IT HAS ALREADY
            CMP.W     -1022(A1),D5      * BEEN LOOKED AT
            BEQ       DEADH3
            MOVEQ     #4,D5
            CMP.W     -1022(A1),D5
            BEQ       EDGE              * BRANCH=EDGE FOUND
            MOVEQ     #8,D5
            CMP.W     -1022(A1),D5
            BEQ       TST83
NEW31       CMP.W     -1022(A2),D4      * NEW MINIMUM VALUE
            BLT.S     TEST32            * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    -1022(A2),D4      * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       -1022(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            SUBQ.W    #1,D0             * ALTER ROW COLUMN COUNTERS
            ADDQ.W    #1,D1
            MOVE.W    #3,DIRECT
*
*
TEST32      MOVEQ     #0,D5             * IS THIS PIXEL MARKED IN BINARY
            CMP.W     2(A1),D5          * IMAGE
            BEQ.S     NEW32             * NO? CONSIDER IT A NEW CANDIDATE
            MOVEQ     #1,D5             * MARKED WITH A 1? IT HAS ALREADY
            CMP.W     2(A1),D5          * BEEN LOOKED AT
            BEQ       DEADH4
            MOVEQ     #4,D5
            CMP.W     2(A1),D5
            BEQ       EDGE              * BRANCH=EDGE FOUND
            MOVEQ     #8,D5
            CMP.W     2(A1),D5
            BEQ       TST84
NEW32       CMP.W     2(A2),D4          * NEW MINIMUM VALUE
            BLE       JMPRET            * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    2(A2),D4          * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       2(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            ADDQ.W    #1,D1             * ALTER ROW COLUMN COUNTERS
            MOVE.W    #4,DIRECT
            BRA       JMPRET            * RETURN TO SEARCH
*
*
DIR4        EQU       *
            MOVE.W    #0,DIRECT
            MOVE.W    #0,DIRECT
            MOVEQ     #0,D5             * IS THIS PIXEL MARKED IN BINARY
```

```
        CMP.W   -1022(A1),D5        * IMAGE?
        BEQ.S   NEW40               * NO? CONSIDER IT A NEW CANDIDATE
        MOVEQ   #1,D5               * MARKED WITH A 1? IT HAS ALREADY
        CMP.W   -1022(A1),D5        * BEEN LOOKED AT
        BEQ     DEADH3
        MOVEQ   #4,D5
        CMP.W   -1022(A1),D5
        BEQ     EDGE                * BRANCH=EDGE FOUND
        MOVEQ   #8,D5
        CMP.W   -1022(A1),D5
        BEQ     TST83
NEW40   CMP.W   -1022(A2),D4        * NEW MINIMUM VALUE
        BLT.S   TEST41              * NEW ADDRESS IN BINARY IMAGE
        MOVE.W  -1022(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
        LEA     -1022(A1),A3
        MOVE.L  A3,NEWADD
        MOVE.W  D6,D0
        MOVE.W  D7,D1
        SUBQ.W  #1,D0
        ADDQ.W  #1,D1
        MOVE.W  #3,DIRECT
*
*
TEST41  MOVEQ   #0,D5               * IS THIS PIXEL MARKED IN BINARY
        CMP.W   2(A1),D5            * IMAGE
        BEQ.S   NEW41               * NO? CONSIDER IT A NEW CANDIDATE
        MOVEQ   #1,D5               * MARKED WITH A 1? IT HAS ALREADY
        CMP.W   2(A1),D5            * BEEN LOOKED AT
        BEQ     DEADH4
        MOVEQ   #4,D5
        CMP.W   2(A1),D5
        BEQ     EDGE                * BRANCH=EDGE FOUND
        MOVEQ   #8,D5
        CMP.W   2(A1),D5
        BEQ     TST84
NEW41   CMP.W   2(A2),D4            * NEW MINIMUM VALUE
        BLT.S   TEST42              * NEW ADDRESS IN BINARY IMAGE
        MOVE.W  2(A2),D4            * NEW ADDRESS IN LAPLACIAN IMAGE
        LEA     2(A1),A3
        MOVE.L  A3,NEWADD
        MOVE.W  D6,D0
        MOVE.W  D7,D1
        ADDQ.W  #1,D1               * ALTER ROW COLUMN COUNTERS
        MOVE.W  #4,DIRECT
*
*
TEST42  MOVEQ   #0,D5               * IS THIS PIXEL MARKED IN BINARY
        CMP.W   1026(A1),D5         * IMAGE
        BEQ.S   NEW42               * NO? CONSIDER IT A NEW CANDIDATE
        MOVEQ   #1,D5               * MARKED WITH A 1? IT HAS ALREADY
        CMP.W   1026(A1),D5         * BEEN LOOKED AT
        BEQ     DEADH5
        MOVEQ   #4,D5
        CMP.W   1026(A1),D5
        BEQ     EDGE                * BRANCH=EDGE FOUND
        MOVEQ   #8,D5
        CMP.W   1026(A1),D5
        BEQ     TST85
NEW42   CMP.W   1026(A2),D4         * NEW MINIMUM VALUE
        BLE     JMPRET              * NEW ADDRESS IN BINARY IMAGE
        MOVE.W  1026(A2),D4         * NEW ADDRESS IN LAPLACIAN IMAGE
        LEA     1026(A1),A3
        MOVE.L  A3,NEWADD
        MOVE.W  D6,D0
        MOVE.W  D7,D1
        ADDQ.W  #1,D0
```

```
          ADDQ.W    #1,D1               * ALTER ROW COLUMN COUNTERS
          MOVE.W    #5,DIRECT
          BRA       JMPRET              * RETURN TO SEARCH
*
DIR5      EQU       *
          MOVE.W    #0,DIRECT
          MOVE.W    #0,DIRECT
          MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     2(A1),D5            * IMAGE
          BEQ.S     NEW50               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     2(A1),D5            * BEEN LOOKED AT
          BEQ       DEADH4
          MOVEQ     #4,D5
          CMP.W     2(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     2(A1),D5
          BEQ       TST84
NEW50     CMP.W     2(A2),D4            * NEW MINIMUM VALUE
          BLT.S     TEST51              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    2(A2),D4            * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       2(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          ADDQ.W    #1,D1
          MOVE.W    #4,DIRECT
*
*
TEST51    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     1026(A1),D5         * IMAGE
          BEQ.S     NEW51               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     1026(A1),D5         * BEEN LOOKED AT
          BEQ       DEADH5
          MOVEQ     #4,D5
          CMP.W     1026(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     1026(A1),D5
          BEQ       TST85
NEW51     CMP.W     1026(A2),D4         * NEW MINIMUM VALUE
          BLT.S     TEST52              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    1026(A2),D4         * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       1026(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          ADDQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
          ADDQ.W    #1,D1
          MOVE.W    #5,DIRECT
*
*
TEST52    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     1024(A1),D5         * IMAGE
          BEQ.S     NEW52               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     1024(A1),D5         * BEEN LOOKED AT
          BEQ       DEADH6
          MOVEQ     #4,D5
          CMP.W     1024(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     1024(A1),D5
          BEQ       TST86
```

```
NEW52   CMP.W   1024(A2),D4      * NEW MINIMUM VALUE
        BLE     JMPRET          * NEW ADDRESS IN BINARY IMAGE
        MOVE.W  1024(A2),D4     * NEW ADDRESS IN LAPLACIAN IMAGE
        LEA     1024(A1),A3
        MOVE.L  A3,NEWADD
        MOVE.W  D6,D0
        MOVE.W  D7,D1
        ADD.W   #1,D0           * ALTER ROW COLUMN COUNTERS
        MOVE.W  #6,DIRECT
        BRA     JMPRET          * RETURN TO SEARCH
*
DIR6    EQU     *
        MOVE.W  #0,DIRECT
        MOVE.W  #0,DIRECT
        MOVEQ   #0,D5           * IS THIS PIXEL MARKED IN BINARY
        CMP.W   1026(A1),D5     * IMAGE
        BEQ.S   NEW60           * NO? CONSIDER IT A NEW CANDIDATE
        MOVEQ   #1,D5           * MARKED WITH A 1? IT HAS ALREADY
        CMP.W   1026(A1),D5     * BEEN LOOKED AT
        BEQ     DEADH5
        MOVEQ   #4,D5
        CMP.W   1026(A1),D5
        BEQ     EDGE            * BRANCH=EDGE FOUND
        MOVEQ   #8,D5
        CMP.W   1026(A1),D5
        BEQ     TST85
NEW60   CMP.W   1026(A2),D4     * NEW MINIMUM VALUE
        BLT.S   TEST61          * NEW ADDRESS IN BINARY IMAGE
        MOVE.W  1026(A2),D4     * NEW ADDRESS IN LAPLACIAN IMAGE
        LEA     1026(A1),A3
        MOVE.L  A3,NEWADD
        MOVE.W  D6,D0
        MOVE.W  D7,D1
        ADDQ.W  #1,D0
        ADDQ.W  #1,D1
        MOVE.W  #5,DIRECT
*
*
TEST61  MOVEQ   #0,D5           * IS THIS PIXEL MARKED IN BINARY
        CMP.W   1024(A1),D5     * IMAGE
        BEQ.S   NEW61           * NO? CONSIDER IT A NEW CANDIDATE
        MOVEQ   #1,D5           * MARKED WITH A 1? IT HAS ALREADY
        CMP.W   1024(A1),D5     * BEEN LOOKED AT
        BEQ     DEADH6
        MOVEQ   #4,D5
        CMP.W   1024(A1),D5
        BEQ     EDGE            * BRANCH=EDGE FOUND
        MOVEQ   #8,D5
        CMP.W   1024(A1),D5
        BEQ     TST86
NEW61   CMP.W   1024(A2),D4     * NEW MINIMUM VALUE
        BLT.S   TEST62          * NEW ADDRESS IN BINARY IMAGE
        MOVE.W  1024(A2),D4     * NEW ADDRESS IN LAPLACIAN IMAGE
        LEA     1024(A1),A3
        MOVE.L  A3,NEWADD
        MOVE.W  D6,D0
        MOVE.W  D7,D1
        ADD.W   #1,D0           * ALTER ROW COLUMN COUNTERS
        MOVE.W  #6,DIRECT
*
*
TEST62  MOVEQ   #0,D5           * IS THIS PIXEL MARKED IN BINARY
        CMP.W   1022(A1),D5     * IMAGE
        BEQ.S   NEW62           * NO? CONSIDER IT A NEW CANDIDATE
        MOVEQ   #1,D5           * MARKED WITH A 1? IT HAS ALREADY
        CMP.W   1022(A1),D5     * BEEN LOOKED AT
```

```
            BEQ       DEADH7
            MOVEQ     #4,D5
            CMP.W     1022(A1),D5
            BEQ       EDGE              * BRANCH=EDGE FOUND
            MOVEQ     #8,D5
            CMP.W     1022(A1),D5
            BEQ       TST87
NEW62       CMP.W     1022(A2),D4       * NEW MINIMUM VALUE
            BLE       JMPRET            * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    1022(A2),D4       * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       1022(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            ADDQ.W    #1,D0             * ALTER ROW COLUMN COUNTERS
            SUBQ.W    #1,D1
            MOVE.W    #7,DIRECT
            BRA       JMPRET            * RETURN TO SEARCH
*
DIR7        EQU       *
            MOVE.W    #0,DIRECT
            MOVE.W    #0,DIRECT
            MOVEQ     #0,D5             * IS THIS PIXEL MARKED IN BINARY
            CMP.W     1024(A1),D5       * IMAGE
            BEQ.S     NEW70             * NO? CONSIDER IT A NEW CANDIDATE
            MOVEQ     #1,D5             * MARKED WITH A 1? IT HAS ALREADY
            CMP.W     1024(A1),D5       * BEEN LOOKED AT
            BEQ       DEADH6
            MOVEQ     #4,D5
            CMP.W     1024(A1),D5
            BEQ       EDGE              * BRANCH=EDGE FOUND
            MOVEQ     #8,D5
            CMP.W     1024(A1),D5
            BEQ       TST86
NEW70       CMP.W     1024(A2),D4       * NEW MINIMUM VALUE
            BLT.S     TEST71            * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    1024(A2),D4       * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       1024(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            ADDQ.W    #1,D0
            MOVE.W    #6,DIRECT
*
*
TEST71      MOVEQ     #0,D5             * IS THIS PIXEL MARKED IN BINARY
            CMP.W     1022(A1),D5       * IMAGE
            BEQ.S     NEW71             * NO? CONSIDER IT A NEW CANDIDATE
            MOVEQ     #1,D5             * MARKED WITH A 1? IT HAS ALREADY
            CMP.W     1022(A1),D5       * BEEN LOOKED AT
            BEQ       DEADH7
            MOVEQ     #4,D5
            CMP.W     1022(A1),D5
            BEQ       EDGE              * BRANCH=EDGE FOUND
            MOVEQ     #8,D5
            CMP.W     1022(A1),D5
            BEQ       TST87
NEW71       CMP.W     1022(A2),D4       * NEW MINIMUM VALUE
            BLT.S     TEST72            * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    1022(A2),D4       * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       1022(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            ADDQ.W    #1,D0             * ALTER ROW COLUMN COUNTERS
            SUBQ.W    #1,D1
```

```
          MOVE.W    #7,DIRECT
*
*
TEST72    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -2(A1),D5           * IMAGE
          BEQ.S     NEW72               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -2(A1),D5           * BEEN LOOKED AT
          BEQ       DEADH8
          MOVEQ     #4,D5
          CMP.W     -2(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -2(A1),D5
          BEQ       TST88
NEW72     CMP.W     -2(A2),D4           * NEW MINIMUM VALUE
          BLE       JMPRET              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    -2(A2),D4           * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       -2(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          SUBQ.W    #1,D1               * ALTER ROW COLUMN COUNTERS
          MOVE.W    #8,DIRECT
          BRA       JMPRET              * RETURN TO SEARCH
*
DIR8      EQU       *
          MOVE.W    #0,DIRECT
          MOVE.W    #0,DIRECT
          MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     1022(A1),D5         * IMAGE
          BEQ.S     NEW80               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     1022(A1),D5         * BEEN LOOKED AT
          BEQ       DEADH7
          MOVEQ     #4,D5
          CMP.W     1022(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     1022(A1),D5
          BEQ       TST87
NEW80     CMP.W     1022(A2),D4         * NEW MINIMUM VALUE
          BLT.S     TEST81              * NEW ADDRESS IN BINARY IMAGE
          MOVE.W    1022(A2),D4         * NEW ADDRESS IN LAPLACIAN IMAGE
          LEA       1022(A1),A3
          MOVE.L    A3,NEWADD
          MOVE.W    D6,D0
          MOVE.W    D7,D1
          ADDQ.W    #1,D0
          SUBQ.W    #1,D1
          MOVE.W    #7,DIRECT
*
*
TEST81    MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
          CMP.W     -2(A1),D5           * IMAGE
          BEQ.S     NEW81               * NO? CONSIDER IT A NEW CANDIDATE
          MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
          CMP.W     -2(A1),D5           * BEEN LOOKED AT
          BEQ       DEADH8
          MOVEQ     #4,D5
          CMP.W     -2(A1),D5
          BEQ       EDGE                * BRANCH=EDGE FOUND
          MOVEQ     #8,D5
          CMP.W     -2(A1),D5
          BEQ       TST88
NEW81     CMP.W     -2(A2),D4           * NEW MINIMUM VALUE
```

```
            BLT.S     TEST82              * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    -2(A2),D4           * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       -2(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            SUBQ.W    #1,D1               * ALTER ROW COLUMN COUNTERS
            MOVE.W    #8,DIRECT
*
*
TEST82      MOVEQ     #0,D5               * IS THIS PIXEL MARKED IN BINARY
            CMP.W     -1026(A1),D5        * IMAGE
            BEQ.S     NEW82               * NO? CONSIDER IT A NEW CANDIDATE
            MOVEQ     #1,D5               * MARKED WITH A 1? IT HAS ALREADY
            CMP.W     -1026(A1),D5        * BEEN LOOKED AT
            BEQ       DEADH1
            MOVEQ     #4,D5
            CMP.W     -1026(A1),D5
            BEQ       EDGE                * BRANCH=EDGE FOUND
            MOVEQ     #8,D5
            CMP.W     -1026(A1),D5
            BEQ       TST81
NEW82       CMP.W     -1026(A2),D4        * NEW MINIMUM VALUE
            BLE       JMPRET              * NEW ADDRESS IN BINARY IMAGE
            MOVE.W    -1026(A2),D4        * NEW ADDRESS IN LAPLACIAN IMAGE
            LEA       -1026(A1),A3
            MOVE.L    A3,NEWADD
            MOVE.W    D6,D0
            MOVE.W    D7,D1
            SUBQ.W    #1,D0               * ALTER ROW COLUMN COUNTERS
            SUBQ.W    #1,D1
            MOVE.W    #1,DIRECT
            BRA       JMPRET              * RETURN TO SEARCH
*
*
*
*
*
INIT        MOVE.L    #0,D3
            MOVE.W    #$FFC0,D4
            MOVE.L    NEWADD,A1
            MOVE.L    A1,A2
            ADD.L     #$80000,A2
INIT1       CMPI.W    #0,-1026(A1)
            BNE.S     INIT2
            CMP.W     -1026(A2),D4
            BLE.S     INIT2
            MOVE.W    -1026(A2),D4
            MOVE.W    #1,DIRECT
INIT2       CMPI.W    #0,-1024(A1)
            BNE.S     INIT3
            CMP.W     -1024(A2),D4
            BLE.S     INIT3
            MOVE.W    -1024(A2),D4
            MOVE.W    #2,DIRECT
INIT3       CMPI.W    #0,-1022(A1)
            BNE.S     INIT4
            CMP.W     -1022(A2),D4
            BLE.S     INIT4
            MOVE.W    -1022(A2),D4
            MOVE.W    #3,DIRECT
INIT4       CMPI.W    #0,2(A1)
            BNE.S     INIT5
            CMP.W     2(A2),D4
            BLE.S     INIT5
```

```
            MOVE.W    2(A2),D4
            MOVE.W    #4,DIRECT
INIT5       CMPI.W    #0,1026(A1)
            BNE.S     INIT6
            CMP.W     1026(A2),D4
            BLE.S     INIT6
            MOVE.W    1026(A2),D4
            MOVE.W    #5,DIRECT
INIT6       CMPI.W    #0,1024(A1)
            BNE.S     INIT7
            CMP.W     1024(A2),D4
            BLE.S     INIT7
            MOVE.W    1024(A2),D4
            MOVE.W    #6,DIRECT
INIT7       CMPI.W    #0,1022(A1)
            BNE.S     INIT8
            CMP.W     1022(A2),D4
            BLE.S     INIT8
            MOVE.W    1022(A2),D4
            MOVE.W    #7,DIRECT
INIT8       CMPI.W    #0,-2(A1)
            BNE.S     CHOOSE
            CMP.W     -2(A2),D4
            BLE.S     CHOOSE
            MOVE.W    -2(A2),D4
            MOVE.W    #8,DIRECT
CHOOSE      CMPI.W    #0,DIRECT
            BEQ       SDONE
            MOVE.W    DIRECT,D3
            MOVE.W    D3,INITDR
            RTS
*
*
CHECK       EQU       *
            CMPI.W    #0,-8(A1)
            BNE       PROCEED
            CMPI.W    #0,-6(A1)
            BNE       PROCEED
            CMPI.W    #0,-4(A1)
            BNE       PROCEED
            CMPI.W    #0,(A1)
            BNE       PROCEED
            CMPI.W    #0,2(A1)
            BNE       PROCEED
            CMPI.W    #0,4(A1)
            BNE       PROCEED
            CMPI.W    #0,-2050(A1)
            BNE       PROCEED
            CMPI.W    #0,-1026(A1)
            BNE       PROCEED
            CMPI.W    #0,1022(A1)
            BNE       PROCEED
            CMPI.W    #0,2046(A1)
            BNE       PROCEED
            BRA       SEARCH
*
*
*
DEADH1      BSR       CHK8
            CMP.W     #1,DEADFLG
            BNE       DISCARD
            LEA       -1026(A1),A2
            CMP.L     CHKADD,A2
            BNE       DISCARD
            MOVE.W    #1,CLOSED
            BRA       SDONE
```

```
*
*
*
DEADH2  BSR       CHK8
        CMP.W     #1,DEADFLG
        BNE       DISCARD
        LEA       -1024(A1),A2
        CMP.L     CHKADD,A2
        BNE       DISCARD
        MOVE.W    #1,CLOSED
        BRA       SDONE
*
*
*
DEADH3  BSR       CHK8
        CMP.W     #1,DEADFLG
        BNE       DISCARD
        LEA       -1022(A1),A2
        CMP.L     CHKADD,A2
        BNE       DISCARD
        MOVE.W    #1,CLOSED
        BRA       SDONE
*
*
*
DEADH4  BSR       CHK8
        CMP.W     #1,DEADFLG
        BNE       DISCARD
        LEA       2(A1),A2
        CMP.L     CHKADD,A2
        BNE       DISCARD
        MOVE.W    #1,CLOSED
        BRA       SDONE
*
*
*
DEADH5  BSR       CHK8
        CMP.W     #1,DEADFLG
        BNE       DISCARD
        LEA       1026(A1),A2
        CMP.L     CHKADD,A2
        BNE       DISCARD
        MOVE.W    #1,CLOSED
        BRA       SDONE
*
*
*
DEADH6  BSR.S     CHK8
        CMP.W     #1,DEADFLG
        BNE       DISCARD
        LEA       1024(A1),A2
        CMP.L     CHKADD,A2
        BNE       DISCARD
        MOVE.W    #1,CLOSED
        BRA       SDONE
*
*
*
DEADH7  BSR.S     CHK8
        CMP.W     #1,DEADFLG
        BNE       DISCARD
        LEA       1022(A1),A2
        CMP.L     CHKADD,A2
        BNE       DISCARD
        MOVE.W    #1,CLOSED
        BRA       SDONE
```

```
*
*
*
DEADH8  BSR.S   CHK8
        CMP.W   #1,DEADFLG
        BNE     DISCARD
        LEA     -2(A1),A2
        CMP.L   CHKADD,A2
        BNE     DISCARD
        MOVE.W  #1,CLOSED
        BRA     SDONE
*
*
*
CHK8    CMPI.W  #8,-1026(A1)
        BNE.S   CHK8A
        LEA     -1026(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8A   CMPI.W  #8,-1024(A1)
        BNE.S   CHK8B
        LEA     -1024(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8B   CMPI.W  #8,-1022(A1)
        BNE.S   CHK8C
        LEA     -1022(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8C   CMPI.W  #8,2(A1)
        BNE.S   CHK8D
        LEA     2(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8D   CMPI.W  #8,1026(A1)
        BNE.S   CHK8E
        LEA     1026(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8E   CMPI.W  #8,1024(A1)
        BNE.S   CHK8F
        LEA     1024(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8F   CMPI.W  #8,1022(A1)
        BNE.S   CHK8G
        LEA     1022(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
        RTS
CHK8G   CMPI.W  #8,-2(A1)
        BNE.S   CHK8H
        LEA     -2(A1),A2
        CMP.L   CHKADD,A2
        BEQ     CHK8R
CHK8H   RTS
CHK8R   ADDQ.L  #4,A7           * RESET SYSTEM STACK
        CMPI.W  #1,DEADFLG
        BEQ     DISCARD
        BRA     CLOSE2
```

```
*
*
*
TST81   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      -1026(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST82   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      -1024(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST83   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      -1022(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST84   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      2(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST85   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      1026(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST86   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      1024(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST87   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      1022(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
TST88   CMPI.W   #1,DEADFLG
        BEQ      DISCARD
        LEA      -2(A1),A3
        CMPA.L   CHKADD,A3
        BNE      DISCARD
        BRA      CLOSE2
        END
```

Appendix C: Ellipse Fitting

The algorithm we use for fitting an ellipse to our list of data points is discussed in a paper by Agin (1980). The most general equation for an ellipse is given by

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0.$$

For a given pixel coordinate (x_i,y_i) we let the pointwise error be given by

$$\xi_i = G(x_i, y_i) = Ax_i^2 + Bx_iy_i + Cy_i^2 + Dx_i + Ey_i + F.$$

Since this quantity is not positive definite, it must be squared to give the aggregate error, Ξ:

$$\Xi = \sum \xi^2 = \sum (Ax_i^2 + Bx_iy_i + Cy_i^2 + Dx_i + Ey_i + F)^2$$

where the sum runs over all points in the ellipse. The idea here is to minimize this aggregate error by chosing appropriate values of the coefficients $(A, B, \ldots)$. This is done by obtaining the partial derivatives with respect to the coefficients and setting the resulting equations equal to zero. The result is five equations in the five unknown coefficients:

$$AM_{04} + BM_{31} + CM_{22} + DM_{30} + EM_{21} + M_{20} = 0$$

$$AM_{31} + BM_{22} + CM_{13} + DM_{21} + EM_{12} + M_{11} = 0$$

$$AM_{22} + BM_{13} + CM_{04} + DM_{12} + EM_{03} + M_{02} = 0$$

$$AM_{30} + BM_{21} + CM_{12} + DM_{20} + EM_{11} + M_{10} = 0$$

$$AM_{21} + BM_{12} + CM_{03} + DM_{11} + EM_{02} + M_{01} = 0$$

where $M_{04} = \sum y_i^4$, $M_{31} = \sum x_i^3 y_i$, etc. The general solution to this problem has been solved on the symbolic manipulation program REDUCE, and the solution is given at the end of this appendix. The solution requires 15 moments to be summed and for this reason we refer to it as the "15 moment" solution.

In view of the complexity of the solution above, we have chosen to sum the ellipse moments in a coordinate system which has its origin at the centroid of the track of interest. The result is that the coefficients D and E in the equation for the ellipse above can be set equal to zero, and the equation for an ellipse becomes

$$Ax^2 + Bxy + Cy^2 - 1 = 0$$

where we have set $F = -1$ so that $A = (1/a^2)$ and $B = (1/b^2)$ for an ellipse with semi-major axis parallel to the x-axis, this is the standard form for and ellipse with the semi-major axis and semi-minor axes given by a and b, respectively.

The aggregate error then becomes

$$\Xi = \sum \xi^2 = \sum (Ax_i^2 + Bx_i y_i + Cy_i^2)^2$$

and the minimization results in three equations in three unknowns

$$AM_{40} + BM_{31} + CM_{22} = M_{20}$$

$$AM_{31} + BM_{22} + CM_{13} = M_{11}$$

$$AM_{22} + BM_{13} + CM_{04} = M_{02}$$

which can easily be solved analytically for the unknown coefficients. The number of moments required for the solution to this problem is 10, the 8 moments in the equations above, plus the two first order moments (M_{10} and M_{01}), used in calculating the ellipse centroid. This is referred to as the "10 moment solution". The solution for A, B, and C is given in terms of the moments at the end of this appendix (we have given the output in REDUCE format to facilitate the comparison of the complexity of the 15 moment solution and the 10 moment solution). Once A, B, and C are known, the major a, and minor axes b, and the azimuth angle ϕ can be calculated as

$$\phi = \frac{\arctan(\frac{A-B}{C})}{2}$$

$$a^2 = \cos(2\phi)/(A - C) + (A + C)\cos(2\phi)$$

$$b^2 = \cos(2\phi)/(C - A) + (A + C)\cos(2\phi).$$

Though this solution requires much less computation than the full 15 moment calculation, it has the disadvantage that the centroid of the ellipse must be known accurately for the fit to be correct. Thus this solution will not work if the points that are to be fit represent, say, half of an ellipse, because by summing the 1st order moments the centroid of the ellipse that these points represent is not determined accurately. The threshold/ tracking detector that we use requires the pixel list form a 'closed loop' as a requirement that an edge be accepted as a track candidate; thus we can be assured that any track that is accepted by the threshold/tracking routine will be fit well by the 10 moment solution (see appendix B).

REDUCE 3.2, 15-Apr-85 ...

$$A := (- ((M22^2 - M31*M13)*M02 - (M22*M13 - M31*M04)*M11 - (M22*M04 - M13^2)*M20))/((2*M31*M13 + M04*M40)*M22 - M22^3 - M31^2*M04 - M13^2*M40)$$

$$B := (((M22^2 - M04*M40)*M11 - (M22*M31 - M13*M40)*M02 - (M22*M13 - M31*M04)*M20))/((2*M31*M13 + M04*M40)*M22 - M22^3 - M31^2*M04 - M13^2*M40)$$

$$C := (- ((M22^2 - M31*M13)*M20 - (M22*M31 - M13*M40)*M11 - (M22*M40 - M31^2)*M02))/((2*M31*M13 + M04*M40)*M22 - M22^3 - M31^2*M04 - M13^2*M40)$$

*** END OF RUN

$$
\begin{aligned}
A := -(& -((((M30*M04 - M03*M31)*M21 - (M13*M31 - M22^2)*M11 + (M13*M22 - M31*M04)*M20 + 2*M21^2*M13 + M30*M03*M22)*M12 \\
& - ((M13^2 - M22*M04)*M20 + (M13*M22 - M31*M04)*M11 + M30*M03*M13)*M21 - (2*M21*M22 + M30*M13)*M12^2 + (M13^2 \\
& - M22*M04)*M30*M11 + (M13*M31 - M22^2)*M03*M20 + M12^3*M31 - M21^3*M04 + M21^2*M03*M22)*M01 - (((M02*M31 \\
& - 3*M20*M13)*M11 + M11^2*M22 + M02*M20*M22)*M03 - (M03^2*M11 - M11*M02*M04 + M02^2*M13)*M30)*M21 + (((4* \\
& M02*M22 - 3*M20*M04)*M11 - M30*M03*M02 - M11^2*M13 + M02^2*M31 - M02*M20*M13)*M21 + (2*M11^2*M31 - M11 \\
& *M20*M22 + M02*M20*M31 - 2*M20^2*M13)*M03 + (M11^2*M04 - M02^2*M22)*M30 - M21^2*M03*M11)*M12 + (((M13^2 \\
& M22*M04)*M11 + (M13*M22 - M31*M04)*M02 + M03^2*M31)*M21 - ((M13^2 - M22*M04)*M02 + M03^2*M22)*M30 + ((\\
& M13*M31 - M22^2)*M02 - (M13*M22 - M31*M04)*M11 + M21^2*M04 + 2*M30*M03*M13)*M12 - (M21*M13 + M30*M04 \\
& + M03*M31)*M12^2 - (M13*M31 - M22^2)*M03*M11 + M12^3*M22 - M21^2*M03*M13)*M10 - (2*(M02*M31 - M20*M13)* \\
& M11 + M21^2*M02 - 2*M21*M03*M20 + M30*M03*M11 + M11^2*M22 - M20^2*M04)*M12^2 - ((M13^2 - M22*M04)*M20 - (M13* \\
& M31 - M22^2)*M02)*M11^2 + (M21*M11 + M30*M02)*M12^3 - (M03^2*M20 - M11^2*M04 + M11*M02*M13 + M02^2*M22 - M02 \\
& *M20*M04)*M21^2 - (M11*M13 - M02*M22)*M30*M03*M11 - (M11*M31 - M20*M22)*M03^2*M20 + (M13^2 - M22*M04)*M02* \\
& M20^2 - (M13*M31 - M22^2)*M02^2*M20 + (M13*M22 - M31*M04)*M11^3 - (M13*M22 - M31*M04)*M11*M02*M20 - M12^4*M20 + \\
& M21^3*M03*M02))/(2*(((M13^2 - M22*M04)*M11 + (M13*M22 - M31*M04)*M02 + M03^2*M31)*M30 + ((M13*M31 - M22^2)*M20 \\
& + (M13*M40 - M31*M22)*M11)*M03)*M21 + 2*(((M13*M31 - M22^2)*M02 - (M13*M22 - M31*M04)*M11)*M30 - ((M13* \\
& M31 - 2*M22^2 + M04*M40)*M11 - (M13*M22 - M31*M04)*M20 + (M13*M40 - M31*M22)*M02)*M21 - ((M13*M40 - \\
& M31*M22)*M20 - (M31^2 - M22*M40)*M11)*M03 + (M30*M04 - M03*M31)*M21^2 + M21^3*M13 + M30^2*M03*M13)*M12 \\
& (2*(M30*M13 - M03*M40)*M21 - 2*(M13*M40 - M31*M22)*M11 + (M31^2 - M22*M40)*M02 + (M22^2 - M04*M40)*M20 + 3*M21^2*
\end{aligned}
$$

M22 + M30^2*M04 + 2*M30*M03*M31)*M12^2 - ((M13^2 - M22*M04)*M02 + M03^2*M22)*M30^2 - ((M13^2 - M22*M04)*M20 + 2*(
M13*M22 - M31*M04)*M11 + (M22^2 - M04*M40)*M02 + 2*M30*M03*M13 + M03^2*M40)*M21^2 + 2*(M21*M31 + M30*M22)*
M12^3 - (M13^2*M40 - 2*M13*M31*M22 + M31^2*M04 + M22^3 - M22*M04*M40)*M11^2 + (M13^2*M40 - 2*M13*M31*M22 + M31^2*M04
+ M22^3 - M22*M04*M40)*M02*M20 - 2*(M13*M31 - M22^2)*M30*M03*M11 - (M31^2 - M22*M40)*M03^2*M20 - M12^4*M40 -
M21^4*M04 + 2*M21^3*M03*M22)

B - ((((M30*M13 - M03*M40)*M21 - (M13*M40 - M31*M22)*M11 + (M22^2 - M04*M40)*M20 + M21^2*M22 + M30^2*M04 + M30*M03*M31)*
M12 - ((M13*M22 - M31*M04)*M20 + (M22^2 - M04*M40)*M11 - M30*M03*M22)*M21 - (M21*M31 + 2*M30*M22)*M12^2 + (M13*
M22 - M31*M04)*M30*M11 + (M13*M40 - M31*M22)*M03*M20 + M12^3*M40 - M21^2*M30*M04 - M30^2*M03*M13)*M01 - (((M30*
M04 - M03*M31)*M21 - (M13*M40 - M31*M22)*M02 + (M22^2 - M04*M40)*M11 + M21^2*M13 - M30*M03*M22)*M12 - ((
M13*M22 - M31*M04)*M11 + (M22^2 - M04*M40)*M02 + M30*M03*M13 + M03^2*M40)*M21 + ((M13*M22 - M31*M04)*M02
+ M03^2*M31)*M30 - (M21*M22 - M03*M40)*M12^2 + (M13*M40 - M31*M22)*M03*M11 - M21^3*M04 + 2*M21^2*M03*M22)*M10
+ (((M02*M31 + M20*M13)*M11 - 2*M30*M03*M11 - 2*M11^2*M22 + M02^2*M40 - 2*M02*M20*M22 + M20^2*M04)*M21 + ((3*M02*
M22 - M20*M04)*M11 - M02^2*M31 - M02*M20*M13)*M30 + (2*M11^2*M40 - 2*M11*M20*M31 + M02*M20*M40 - M20^2*
M22)*M03 - M21^3*M02 + M21^2*M03*M20 - M30^2*M03*M02)*M12 - (((M02*M40 - 3*M20*M22)*M11 + M02*M20*M31 + M20^2
*M13)*M03 + (M03^2*M20 - 2*M11^2*M04 + 2*M11*M02*M13 + M02^2*M22 - M02*M20*M04)*M30)*M21 + ((M02*M31 + M20*
M13) - 2*M11*M22)*M30*M03*M11 + ((M02*M22 - 2*M20*M04)*M11 + M30*M03*M02 + M02*M20*M13)*M21^2 - ((2*M02*M40
- M20*M22)*M11 - M21^2*M11 - M21*M30*M02 - M30*M03*M20 - M02*M20*M31)*M12^2 - ((M13*M22 - M31*M04)*M20 - (
M13*M40 - M31*M22)*M02)*M11^2 + (M03^2*M11 - M11*M02*M04 + M02^2*M13)*M30^2 - (M11*M40 - M20*M31)*M03^2*M20 + (
M13*M22 - M31*M04)*M02*M20^2 - (M13*M40 - M31*M22)*M02^2*M20 + (M22^2 - M04*M40)*M11^3 - (M22^2 - M04*M40)*M11*M02
*M20^3 - M12^3*M21*M20)/(2*(((M13^2 - M22*M04)*M11 + (M13*M22 - M31*M04)*M02 + M03^2*M31)*M30 + ((M13*M31 - M22^2)*M20

$$+ (M13*M40 - M31*M22)*M11)*M03)*M21 + 2*(((M13*M31 - M22^2)*M02 - (M13*M22 - M31*M04)*M11)*M40 - ((M13*$$

$$M31 - 2*M22^2 + M04*M40)*M11 - (M13*M22 - M31*M04)*M20 + (M13*M40 - M31*M22)*M02)*M21 - ((M13*M40 -$$

$$M31*M22)*M20 - (M31^2 - M22*M40)*M11)*M03 + (M30*M04 - M03*M31)*M21^2 + M21^3*M13 + M30^2*M03*M13)*M12$$

$$(2*(M30*M13 - M03*M40)*M21 - 2*(M13*M40 - M31*M22)*M11 + (M31^2 - M22*M40)*M02 + (M22^2 - M04*M40)*M20 + 3*M21^2*$$

$$M22 + M30^2*M04 + 2*M30*M03*M31)*M12^2 - ((M13^2 - M22*M04)*M02 + M03^2*M22)*M30^2 - ((M13^2 - M22*M04)*M20 + 2*(M13*M22 - M31*M04)*M11 + (M22^2 - M04*M40)*M02 + 2*M30*M03*M13 + M03^2*M40)*M21^2 + 2*(M21*M31 + M30*M22)*$$

$$M12^3 - (M13^2*M40 - 2*M13*M31*M22 + M31^2*M04 + M22^3 - M22*M04*M40)*M11^2 + (M13^2*M40 - 2*M13*M31*M22 + M31^2*M04$$

$$+ M22^3 - M22*M04*M40)*M02*M20 - 2*(M13*M31 - M22^2)*M30*M03*M11 - (M31^2 - M22*M40)*M03^2*M20 - M12^4*M40$$

$$M21^4*M04 + 2*M21^3*M03*M22)$$

C. $$- ((((M30*M13 - M03*M40)*M21 - (M13*M40 - M31*M22)*M11 + (M31^2 - M22*M40)*M02 + 2*M21^2*M22 + M30*M03*M31)*M12 + ((M13*M31 - M22^2)*M11 + (M13*M40 - M31*M22)*M02 - M30*M03*M22)*M21 - (2*M21*M31 + M30*M22)*M12^2 - (M13*M31 - M22^2)*M30*M02 - (M31^2 - M22*M40)*M03*M11 + M12^3*M40 - M21^3*M13 + M21^2*M03*M31)*M10 - (((3*M02*M31 - M20*M13)*M11 - M11^2*M22 - M02*M20*M22)*M30 - ((3*M02*M40 - 4*M20*M22)*M11 + M30*M03*M20 + M11^2*M31 + M02*M20*M31 - M20^2*M13)*M21 - (M11*M40 - M20*M31)*M03*M20 + M21^3*M11 + 2*M21^2*M30*M02 + M30^2*M03*M11)*M12 - ((M13*M31 - M22^2)*M20 + (M13*M40 - M31*M22)*M11 + 2*M30*M03*M31)*M21 - ((M13*M40 - M31*M22)*M20 - (M31^2 - M22*M40)*M11 + M21^2*M31 - M30^2*M13)*M12 + (M21*M40 - M30*M31)*M12^2 - (M30*M13 + M03*M40)*M21^2 - (M13*M31 - M22^2)*M30*M11 - (M31^2 - M22*M40)*M03*M20 + M21^3*M22 - M30^2*M03*M22)*M01 - ((2*M11^2*M13 - M11*M02*M22 - 2*M02^2*M31 + M02*M20*M13)*M30 + (M11^2*M40 - M20^2*M22)*M03)*M21 - (2*(M02*M31 - M20*M13)*M11 - M30*M03*M11 - M11^2*M22 + M02^2*M40)*M21^2 - ((M13*M31 - M22^2)*M20 - (M31^2 - M22*M40)*M02)*M11^2 + (M21^2*M20 + M21*M30*M11 + M30^2*M02 - M11^2*M40 + M11*M20*M31 - M02*M20*M40 + M20^2*M22)*M12^2 + (M11*M13 - M02*M22)*M30^2*M02 + (M11*$$

M31 - M20*M22)*M30*M03*M11 + (M13*M31 - M22^2)*M02*M20^2 + (M13*M40 - M31*M22)*M11^3 - (M13*M40 - M31*M22)*M11*

M02*M20^2 - (M31^2 - M22*M40)*M02^2*M20^3 - M12^3*M30*M20^4 + M21^3*M02^4 - M21^3*M03*M20^3)/(2*(((M13^2 - M22*M04)*M11 + (M13*

M22 - M31*M04)*M02 + M03^2*M31)*M30^2 + ((M13*M31 - M22^2)*M20^2 + (M13*M40 - M31*M22)*M11)*M03^2)*M21^2 + 2*((

(M13*M31 - M22^2)*M02^2 - (M13*M22 - M31*M04)*M11)*M30^2 - ((M13*M31 - 2*M22^2 + M04*M40)*M11^2 - (M13*M22 - M31

*M04)*M20^2 + (M13*M40 - M31*M22)*M02^2)*M21^2 - ((M13*M40 - M31*M22)*M20^2 - (M31^2 - M22*M40)*M11^2)*M03^2 + (

M30*M04 - M03*M31)*M21^2 + M21^3*M13^2 + M30^2*M03^2*M13^2)*M12^2 - (2*(M30*M13 - M03*M40)*M21^3 - 2*(M13*M40 - M31*

M22)*M11^3 + (M31^2 - M22*M40)*M02^2*M11^2 + (M22^2 - M04*M40)*M20^2*M11^2 + 3*M21^2*M22^2 + M30^2*M04^2 + 2*M30*M03*M31)*M12^2 - (

(M13^2 - M22*M04)*M02^2 + M03^2*M22^2)*M30^2 - ((M13^2 - M22*M04)*M20^2 + 2*(M13*M22 - M31*M04)*M11^2 + (M22^2 - M04*M40

)*M02^2 + 2*M30*M03*M13^2 + M03^2*M40^2)*M21^2 + 2*(M21^3*M31^2 + M30*M22^3)*M12^2 - (M13^2*M40^2 - 2*M13*M31*M22^2 + M31^2*

M04^2 + M22^3 - M22*M04*M40)*M11^2 + (M13^2*M40^2 - 2*M13*M31*M22^2 + M31^2*M04^2 + M22^3 - M22*M04*M40)*M02^2*M20^2 - 2*(

M13*M31 - M22^2)*M30*M03*M11^2 - (M31^2 - M22*M40)*M03^2*M20^4 - M12^4*M40^4 - M21^4*M04^4 + 2*M21^3*M03*M22^3)

D = -((((M11*M31 - M02*M40)*M03^2 - (M11*M04 - M02*M13)*M30^2)*M21^2 - (M11*M13 - 2*M02*M22 + M20*M04)*M21^2)*(M11*M22^2 + M02

*M31^2 - 2*M20*M13)*M30*M03 - (M13*M31 - M22^2)*M02^2*M20^2 + (M13*M22 - M31*M04)*M11*M20^2 + (M31^2 - M22*M40)*M02^2*

(M22^2 - M04*M40)*M11^2)*M12^2 - (((M13^2 - M22*M04)*M30^2 + (M13*M40 - M31*M22)*M03^2)*M21^2 - ((M13*M31 - 2*M22^2 +

M04*M40)*M21^2 + (M13*M22 - M31*M04)*M30^2 - (M31^2 - M22*M40)*M03^2)*M12^2 - (M13^2*M40^2 - 2*M13*M31*M22^2 + M31^2*

M04^2 + M22^3 - M22*M04*M40)*M11^2 - (M13*M31 - M22^2)*M30^2*M03^2 - (M13*M22 - M31*M04)*M21^2 + (M13*M40 - M31*M22)*

M12^2)*M01^2 - (((M13^2 - M22*M04)*M20^2 - (M13*M31 - M04*M40)*M02^2)*M11^2 - (M11^2 - M02*M20)*(M13*M22 - M31*M04)^2 - (

M11*M13 - M02*M22)*M30*M03^2 - (M11*M40 - M20*M31)*M03^2 - (M13*M40 - M31*M22)*M02^2)*M21^2 + (((M13*M31 - M22^2)

*M20^2 - (M31^2 - M22*M40)*M02^2)*(M13*M40 - M31*M22)*M11^2*M03^2*M11^2 - (2*((M13*M22 - M31*M04)*M21^2 - (M13*M40

- M31*M22)*M03^2)*M12^2 + (M13^2*M40^2 - 2*M13*M31*M22^2 + M31^2*M04^2 + M22^3 - M22*M04*M40)*M02^2*(M13^2 - M22*M04

$$
\begin{aligned}
E = {} & (((M11 \cdot M13 - M20 \cdot M04) \cdot M30 - (M11 \cdot M40 - M20 \cdot M31) \cdot M03) \cdot M21 + ((M13 \cdot M31 - M04 \cdot M40) \cdot M20 - (M31^2 - M22 \cdot M40) \cdot M02) \cdot M11 \\
& - (M11^2 - M02 \cdot M20) \cdot (M13 \cdot M40 - M31 \cdot M22) + (M11 \cdot M31 - M20 \cdot M22) \cdot M30 \cdot M03 + (M11 \cdot M22 + M02 \cdot M31 - 2 \cdot M20 \cdot M13) \cdot M21^2 + \\
& (M11 \cdot M04 - M02 \cdot M13) \cdot M30^2 - (M13 \cdot M22 - M31 \cdot M04) \cdot M20^2) \cdot M12 - (((M13^2 - M22 \cdot M04) \cdot M30 + (M13 \cdot M40 - M31 \cdot M22) \cdot M03) \cdot \\
& M21 - ((M13 \cdot M31 - 2 \cdot M22^2 + M04 \cdot M40) \cdot M21 + (M13 \cdot M22 - M31 \cdot M04) \cdot M30 - (M31^2 - M22 \cdot M40) \cdot M03) \cdot M12 - (M13^2 \cdot M40 - 2 \\
& \cdot M13 \cdot M31 \cdot M22 + M31^2 \cdot M04 + M22^3 - M22 \cdot M04 \cdot M40) \cdot M11 - (M13 \cdot M31 - M22^2) \cdot M30 \cdot M03 - (M13 \cdot M22 - M31 \cdot M04) \cdot M21^2 + \\
& (M13 \cdot M40 - M31 \cdot M22) \cdot M12^2) \cdot M10 - (((M13^2 - M22 \cdot M04) \cdot M20 - (M13 \cdot M31 - M22^2) \cdot M02) - (M13 \cdot M22 - M31 \cdot M04) \cdot M11) \cdot M30 \\
& \cdot M11 - (2 \cdot ((M13 \cdot M31 - M22^2) \cdot M30 - (M13 \cdot M40 - M31 \cdot M22) \cdot M21) \cdot M12 + (M13^2 \cdot M40 - 2 \cdot M13 \cdot M31 \cdot M22 + M31^2 \cdot M04 + M22^3 -
\end{aligned}
$$

$$
\begin{aligned}
&) \cdot M21^2 + 2 \cdot (M13 \cdot M31 - M22^2) \cdot M21 \cdot M03 - (M31^2 - M22 \cdot M40) \cdot M03^2 - (M22^2 - M04 \cdot M40) \cdot M12^2) \cdot M10 - ((M11 \cdot M31 - M20 \\
& \cdot M22) \cdot M03^2 - (M13^2 - M22 \cdot M04) \cdot M02 \cdot M20 + (M13 \cdot M31 - M22^2) \cdot M02^2 + (M13 \cdot M22 - M31 \cdot M04) \cdot M11 \cdot M02) \cdot M30 + ((M11 \cdot \\
& M22 - 2 \cdot M02 \cdot M31 + M20 \cdot M13) \cdot M21 - (M11 \cdot M40 - M20 \cdot M31) \cdot M03 - (M02 \cdot M22 - M20 \cdot M04) \cdot M30) \cdot M12^2 - (2 \cdot M11 \cdot M22 - \\
& M02 \cdot M31 - M20 \cdot M13) \cdot M21^2 \cdot M03 + (M11 \cdot M04 - M02 \cdot M13) \cdot M21^3 + (M02 \cdot M40 - M20 \cdot M22) \cdot M12^3) / (2 \cdot (((M13^2 - M22 \cdot M04) \cdot M11 \\
& + (M13 \cdot M22 - M31 \cdot M04) \cdot M02 + M03^2 \cdot M31) \cdot M30 + ((M13 \cdot M31 - M22^2) \cdot M20 + (M13 \cdot M40 - M31 \cdot M22) \cdot M11) \cdot M03) \cdot M21 \\
& + 2 \cdot (((M13 \cdot M31 - M22^2) \cdot M02 - (M13 \cdot M22 - M31 \cdot M04) \cdot M11) \cdot M30 - ((M13 \cdot M31 - 2 \cdot M22^2 + M04 \cdot M40) \cdot M11 - (M13 \cdot M22 - \\
& M31 \cdot M04) \cdot M20 + (M13 \cdot M40 - M31 \cdot M22) \cdot M02) \cdot M21 - ((M13 \cdot M40 - M31 \cdot M22) \cdot M20 - (M31^2 - M22 \cdot M40) \cdot M11) \cdot M03 + \\
& (M30 \cdot M04 - M03 \cdot M31) \cdot M21^2 + M21^3 \cdot M13 + M30^2 \cdot M03 \cdot M13) \cdot M12 - (2 \cdot (M30 \cdot M13 - M03 \cdot M40) \cdot M21 - 2 \cdot (M13 \cdot M40 - M31 \cdot M22 \\
&) \cdot M11 + (M31^2 - M22 \cdot M40) \cdot M02 + (M22^2 - M04 \cdot M40) \cdot M20 + 3 \cdot M21^2 \cdot M22 + M30^2 \cdot M04 + 2 \cdot M30 \cdot M03 \cdot M31) \cdot M12^2 - ((\\
& M13^2 - M22 \cdot M04) \cdot M02 + M03^2 \cdot M22) \cdot M30^2 - ((M13^2 - M22 \cdot M04) \cdot M20 + 2 \cdot (M13 \cdot M22 - M31 \cdot M04) \cdot M11 + (M22^2 - M04 \cdot \\
& M40) \cdot M02 + 2 \cdot M30 \cdot M03 \cdot M13 + M03^2 \cdot M40) \cdot M21^2 + 2 \cdot (M21 \cdot M31 + M30 \cdot M22) \cdot M12^3 - (M13^2 \cdot M40 - 2 \cdot M13 \cdot M31 \cdot M22 + \\
& M31^2 \cdot M04 + M22^3 - M22 \cdot M04 \cdot M40) \cdot M11^2 + (M13^2 \cdot M40 - 2 \cdot M13 \cdot M31 \cdot M22 + M31^2 \cdot M04 + M22^3 - M22 \cdot M04 \cdot M40) \cdot M02 \cdot M20 \\
& 2 \cdot (M13 \cdot M31 - M22^2) \cdot M30 \cdot M03 \cdot M11 - (M31^2 - M22 \cdot M40) \cdot M03^2 \cdot M20 - M12^4 \cdot M40 - M21^4 \cdot M04 + 2 \cdot M21^3 \cdot M03 \cdot M22)
\end{aligned}
$$

```
M22*M04*M40)*M20 - (M13^2 - M22*M04)*M30^2 + 2*(M13*M22 - M31*M04)*M21*M30 - (M31^2 - M22*M40)*M12^2 - (M22^2
 - M04*M40)*M21^2)*M01 ((M11*M31 + M02*M40 - 2*M20*M22)*M21 + (2*M11*M22 - M02*M31 - M20*M13)*M30)*M12^2
+ ((M11*M22 - 2*M02*M31 + M20*M13)*M30*M03 + (M13^2 - M22*M04)*M20^2 - (M13*M31 - M22^2)*M02*M20 - (M13*M40 - M31*
 M22)*M11*M02 - (M22^2 - M04*M40)*M11^2)*M21 - ((M11*M04 - M02*M13)*M30 - (M02*M40 - M20*M22)*M03)*M21^2 ((
 M13*M31 - M22^2)*M20 - (M13*M40 - M31*M22)*M11 - (M31^2 - M22*M40)*M02)*M03*M20 - (M11*M13 - M02*M22)*M30^2 +
 M03 + (M11*M40 - M20*M31)*M12^3 - (M02*M22 - M20*M04)*M21^3)/(2*(((M13^2 - M22*M04)*M11 + (M13*M22 - M31*M04)*M02
 + M03^2*M31)*M30 + ((M13*M31 - M22^2)*M20 + (M13*M40 - M31*M22)*M11)*M03)*M21 + 2*(((M13*M31 - M22^2)*M02
 - (M13*M22 - M31*M04)*M11)*M30 - ((M13*M31 - 2*M22^2 + M04*M40)*M11 - (M13*M22 - M31*M04)*M20 + (M13*M40
 - M31*M22)*M02)*M21 - ((M13*M40 - M31*M22)*M20 - (M31^2 - M22*M40)*M11)*M03 + (M30*M04 - M03*M31)*
 M21^2 + M21^3*M13 + M30^2*M03*M13)*M12 - (2*(M30*M13 - M03*M40)*M21 - 2*(M13*M40 - M31*M22)*M11 + (M31^2 - M22*
 M40)*M02 + (M22^2 - M04*M40)*M20 + 3*M21^2*M22 + M30^2*M04 + 2*M30*M03*M31)*M12^2 - ((M13^2 - M22*M04)*M02 +
 M03^2*M22)*M30^2 - ((M13^2 - M22*M04)*M20 + 2*(M13*M22 - M31*M04)*M11 + (M22^2 - M04*M40)*M02 + 2*M30*M03*M13
 + M03^2*M40)*M21^2 + 2*(M21*M31 + M30*M22)*M12^3 - (M13^2*M40 - 2*M13*M31*M22 + M31^2*M04 + M22^3 - M22*M04*M40)
*M11^2 + (M13^2*M40 - 2*M13*M31*M22 + M31^2*M04 + M22^3 - M22*M04*M40)*M02*M20 - 2*(M13*M31 - M22^2)*M30*M03*M11
 (M31^2 - M22*M40)*M03^2*M20 - M12^4*M40 - M21^4*M04 + 2*M21^3*M03*M22)
```

*** END OF RUN

Appendix D: Duplicate Track Elimination

The most direct, or 'brute force' way to solve this problem is using an N^2 algorithm, which compares the x and y coordinates of every track record in a file to those of every other record. However, for a file containing 50,000 tracks, this method takes prohibitively long. The easiest way to reduce this time is to take advantage of the sequentiality of each frame's position in the low power scan file. If the number of rows and columns scanned is included in the file header, and if the frame index number is included as an element in the track record, then one may choose to compare the coordinates of each track only to those of tracks in immediately adjacent frames.

To insure that all necessary frames are checked for duplicates, we search the current frame, the subsequent frame to the right or left (depending on which way the scan progressed; this frame does not exist if the current frame marks the end of a row), and each of the three frames most nearly beneath the current frame (again subject to discretion at edges). Figure 16 shows the search pattern we use for a frame in a row with an original search direction from left to right. In order for this scheme to work, tracks must be in a file which allows direct access keyed to frame number. If no duplicates are found for a given track, it is written to a new file; if a duplicate is found, both tracks are first written to a "duplicate" file, and then only the second of the of the two (sequentially) is retained and written to the new file. The program typically requires twenty minutes to eliminate all duplicates in a file of 50,000 track records.

REFERENCES

1. Abdou, I. E., and Pratt, W. K. *Proceedings of the IEEE* **67**, 753-763 (1979).
2. Agin, G. J., *Fitting Ellipses and General Second Order Curves*, The Robotics Institute, Carnegie-Mellon University, Report CMU-RI-TR-81-5.
3. Brumm, A. P., Senior Thesis, University of Michigan, 1989.
4. Davis, L. S., *Computer Graphics and Image Processing* **4**, 248-270 (1975).
5. Dowell, M. L., Senior Thesis, University of Michigan, 1987.
6. Gerbier, G., Williams, W. T., Price, P. B. *Phys. Rev. Lett.* **39**, 2535-2538 (1987).
7. Levaldi, S. in *Fundamentals of Computer Vision*, O. D. Faugeras, ed., Cambridge University Press (1982).
8. Logan, B. F. *Bell Sys. Tech. J.* **56**, 487-510 (1977).
9. Marr, D. and Hildreth, E. *Proc. R. Soc. Lond.* **B207**, 187-217 (1980).
10. Marr D., Ullman, S., and Poggio, T. *J. Opt. Soc. Am.* **69**, 914-916 (1979).
11. Peli, T. and Malah, D. *Computer Graphics and Image Processing* **20**, 1-21 (1982).
12. Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. *Numerical Recipes*, Cambridge University Press, (1986).
13. Somogyi, G. *Nucl. Instrum. Meth.* **173**, 21-42, (1980).
14. Tarlé, G., Ahlen, S. P. and Price, P. B. *Nature* **293**, 556-558 (1981).

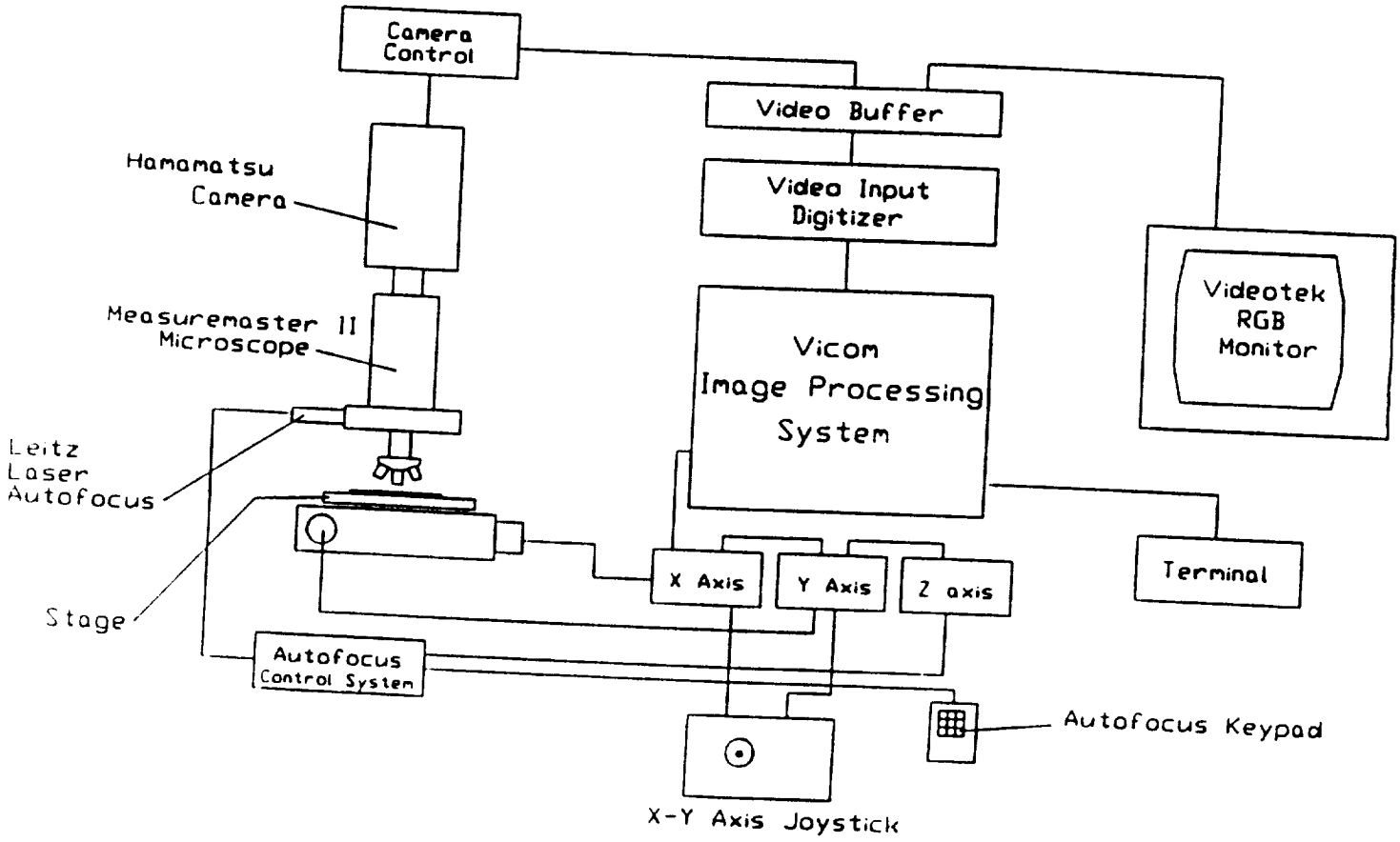


Figure 1. Schematic diagram of the microscope system.

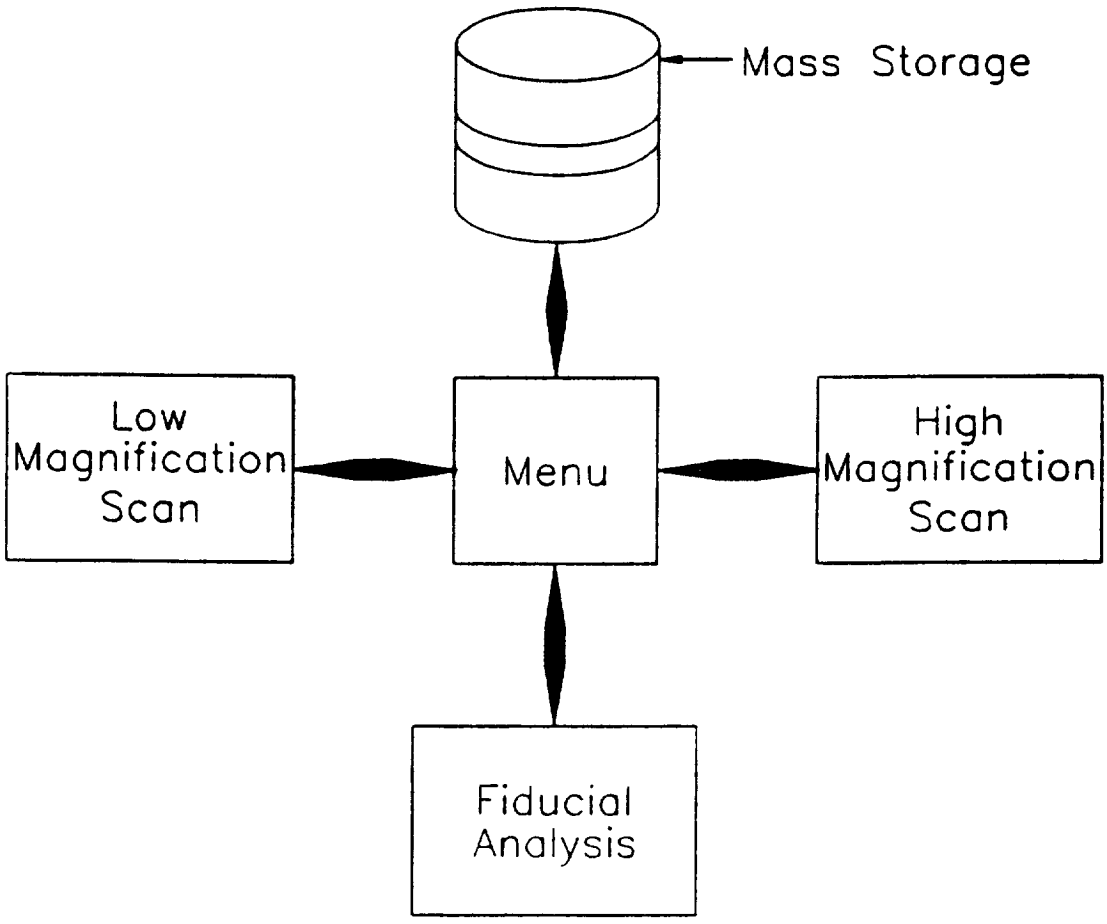


Figure 2. Diagram of the menu system and associated tasks

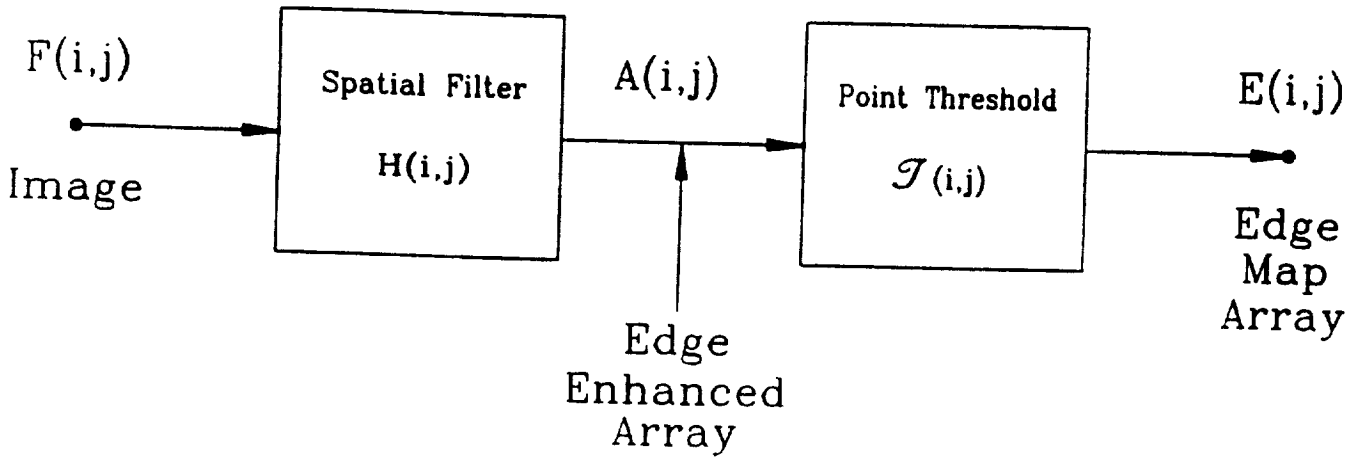


Figure 3. Enhancement/thresholding edge detection sytem

a.

- 1	- 1	- 1
- 1	8	- 1
-1	- 1	- 1

b.

1	1	1
1	1	1
1	1	1

Figure 4a. Laplacian Mask

Figure 4b. Smoothing Mask

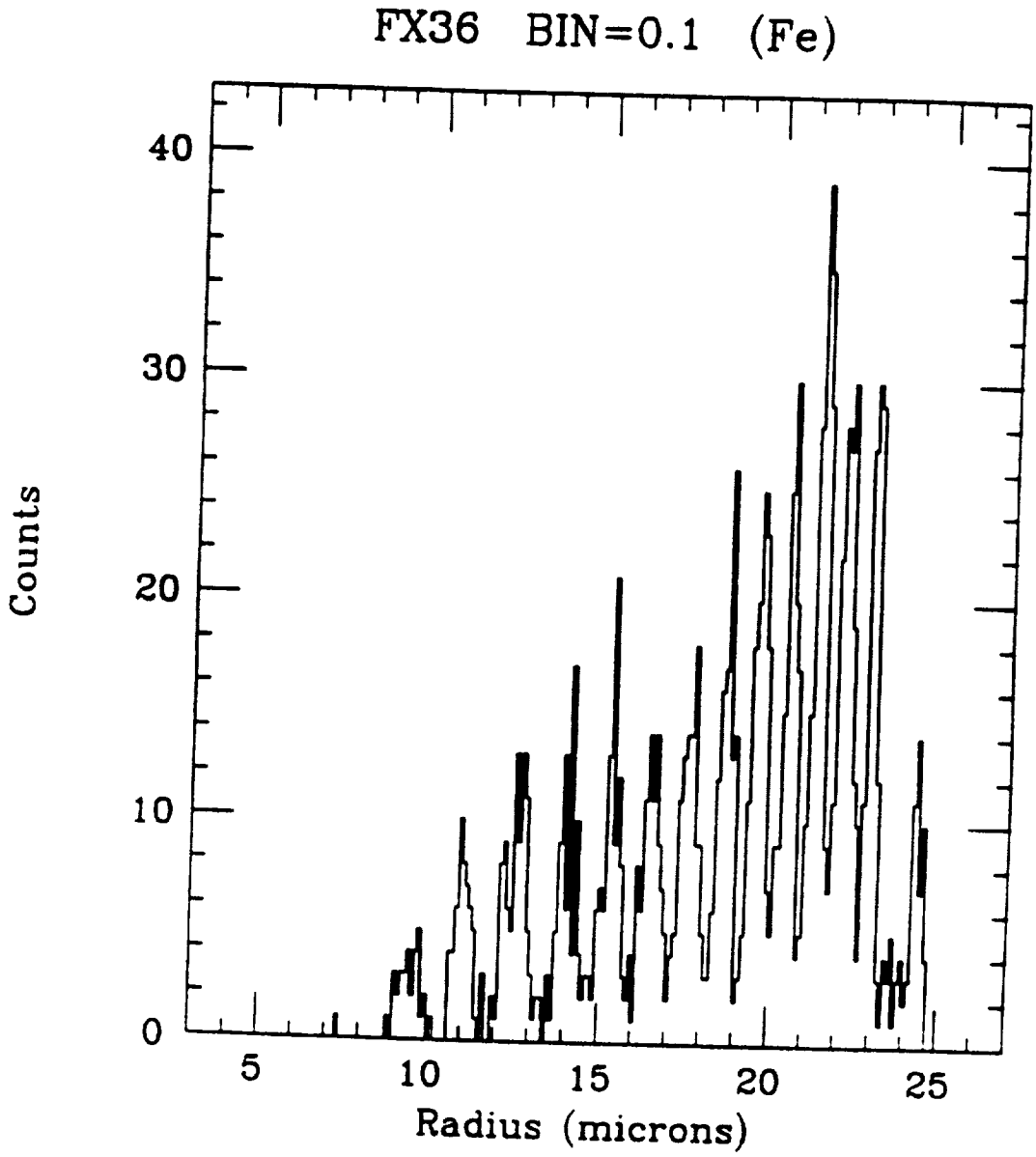


Figure 5. Track radius histogram for sheet #36 (front)

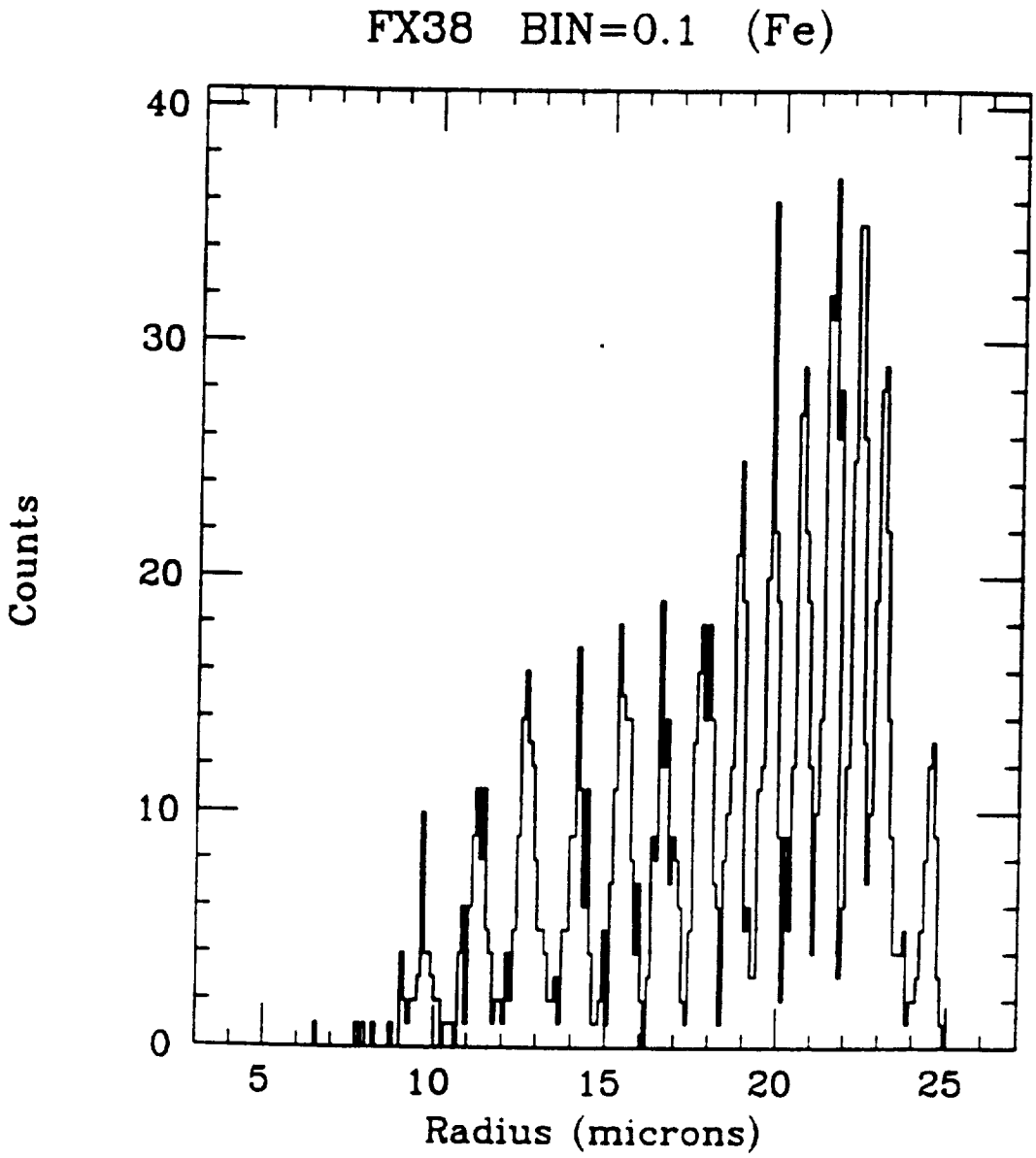


Figure 6. Track radius histogram for sheet #38 (front)

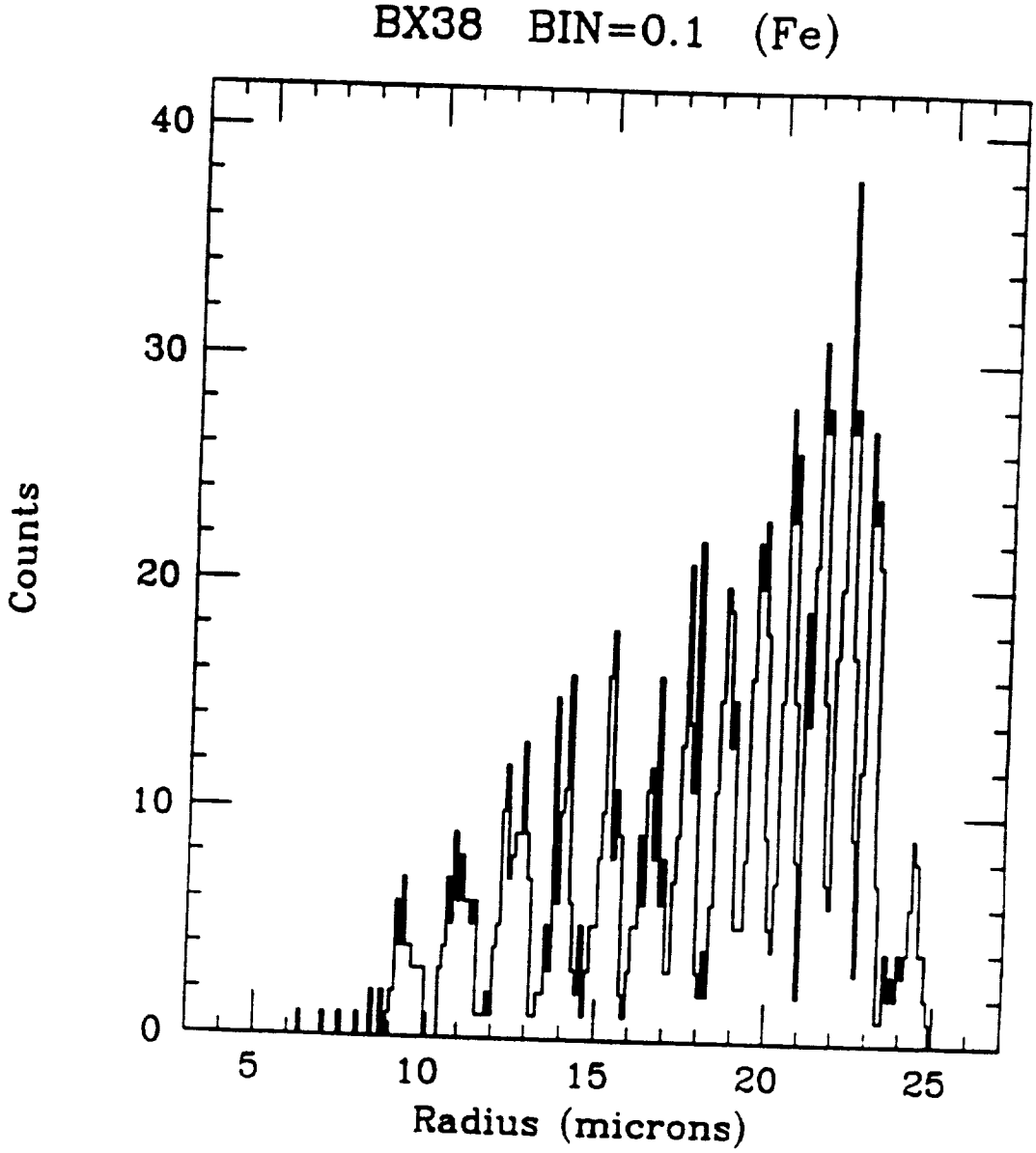


Figure 7. Track radius histogram for sheet #38 (back)

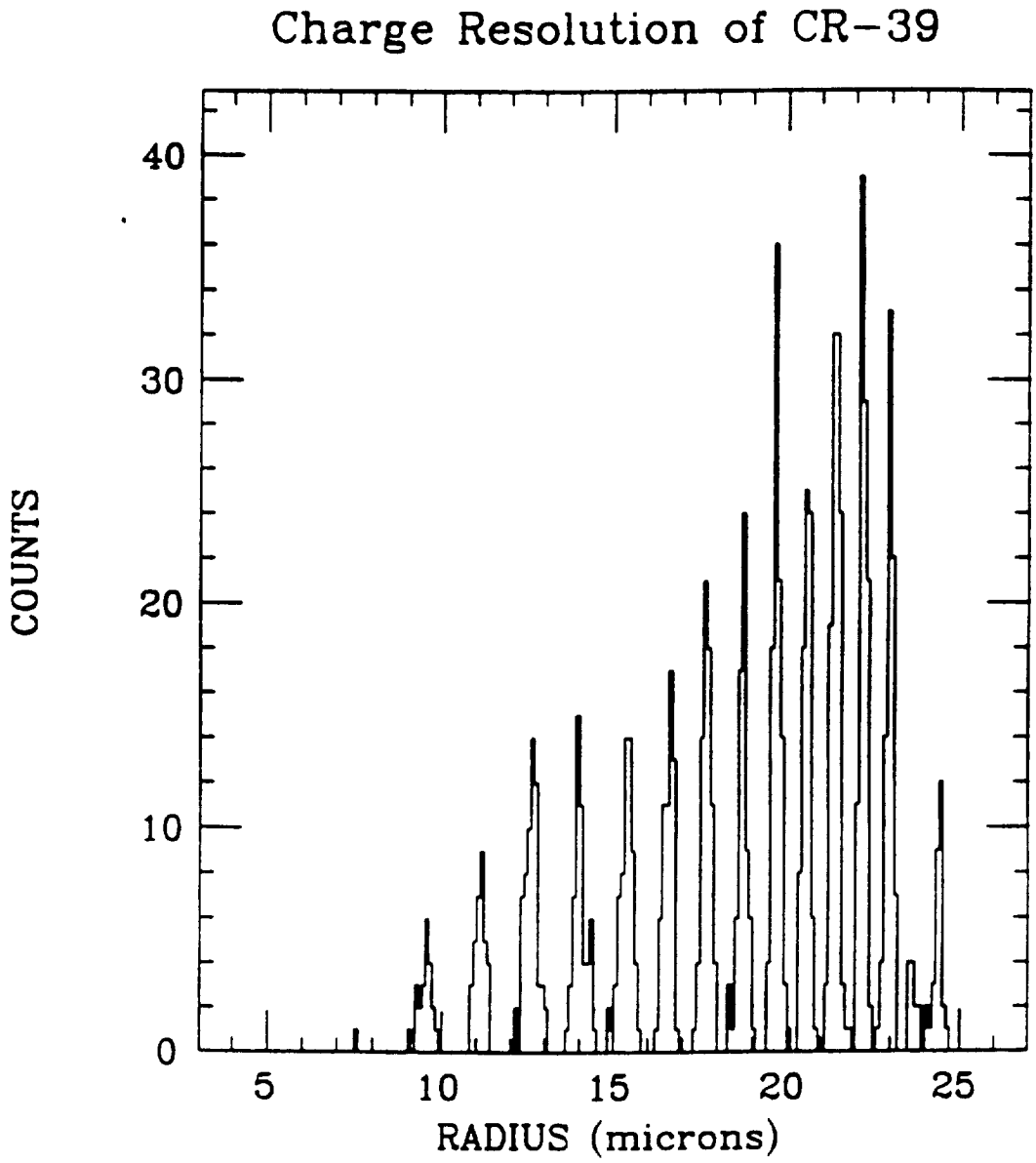


Figure 8. Average track radius for sheets #36 and #38

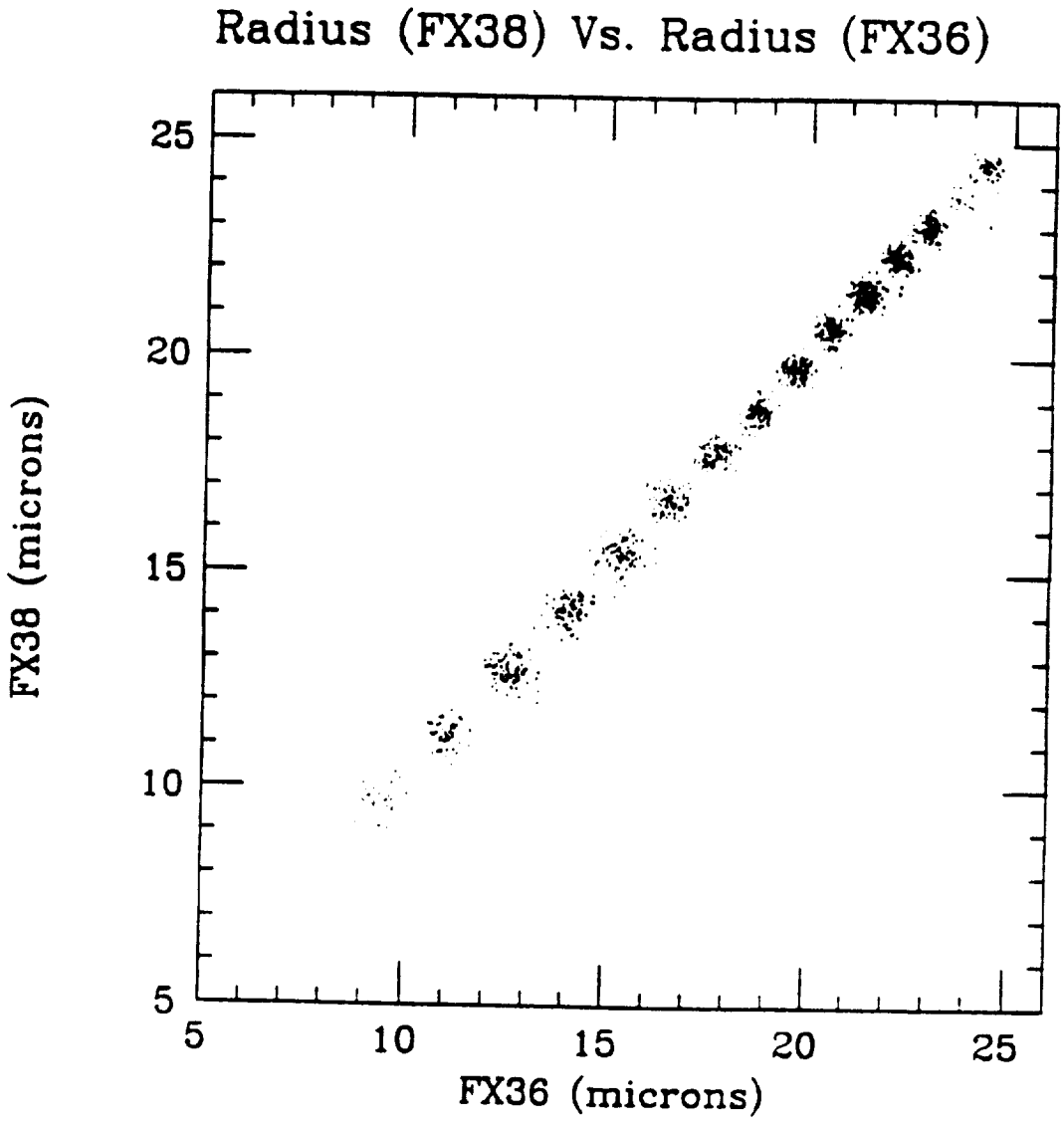


Figure 9. Plot of R_1 vs. R_2 for sheet #38 (front) and sheet #36 (front)

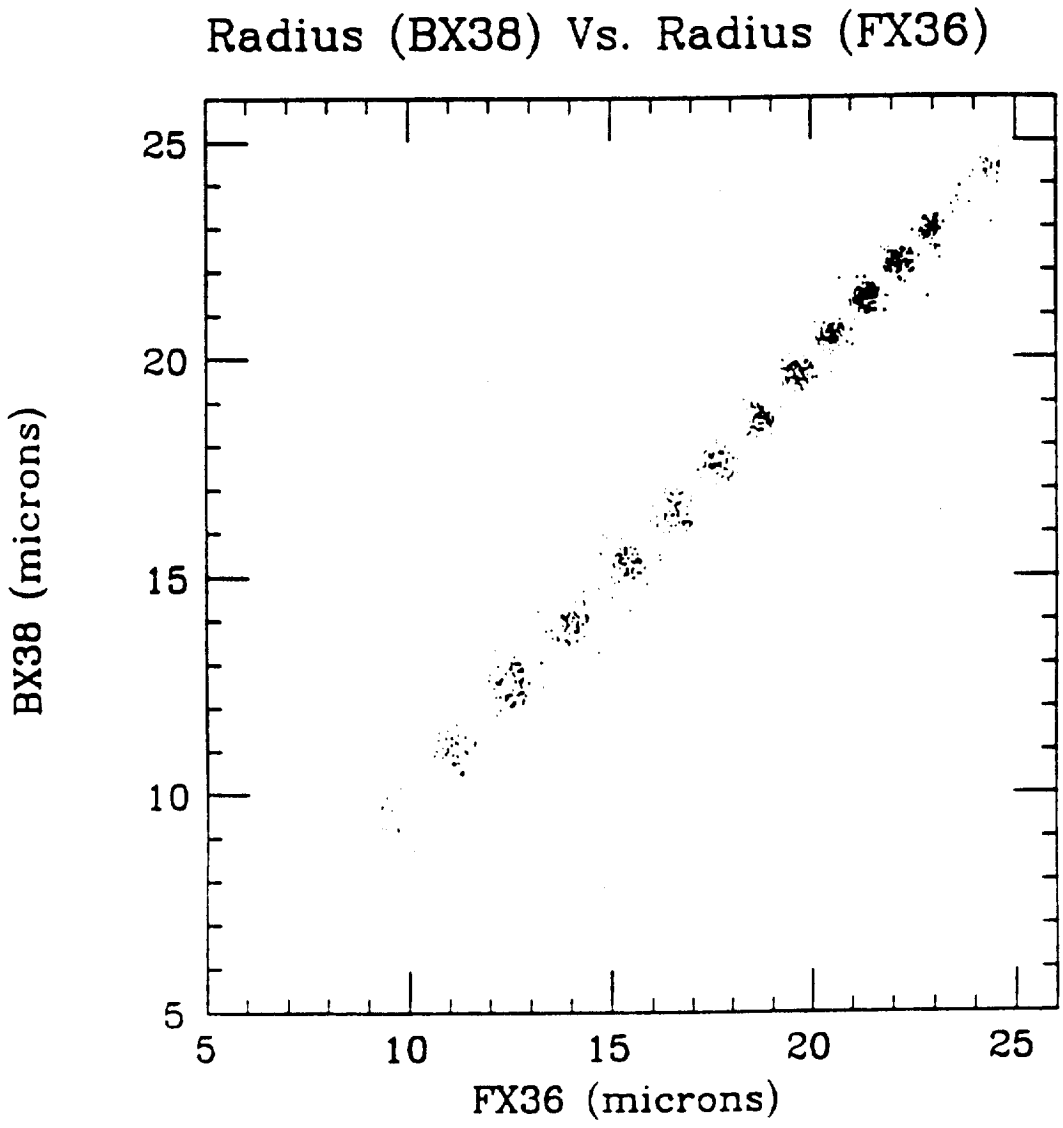


Figure 10. Plot of R_1 vs. R_2 for sheet #38 (back) and sheet #36 (front)

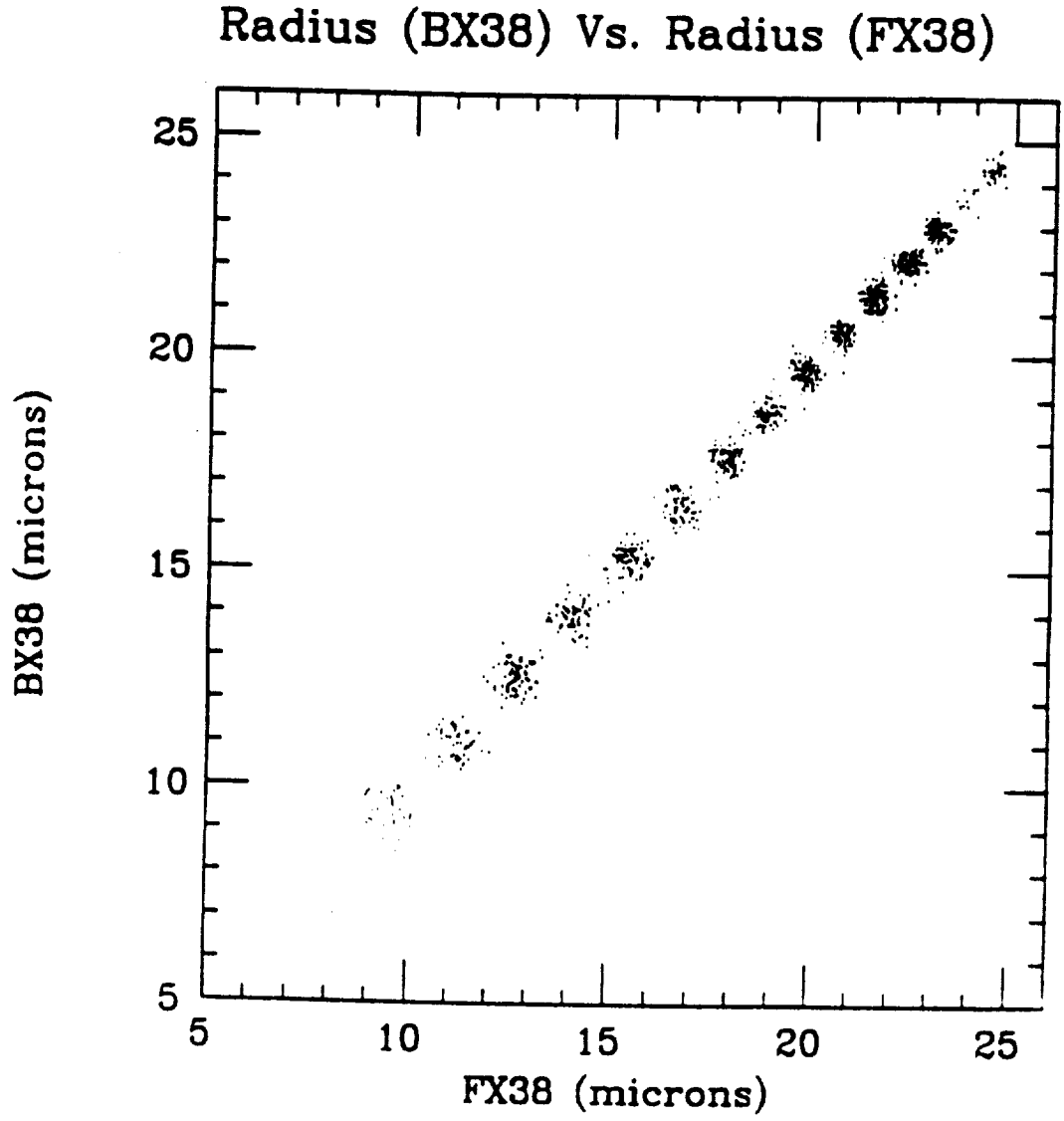


Figure 11. Plot of R_1 vs. R_2 for sheet #38 (back) and sheet #38 (front)

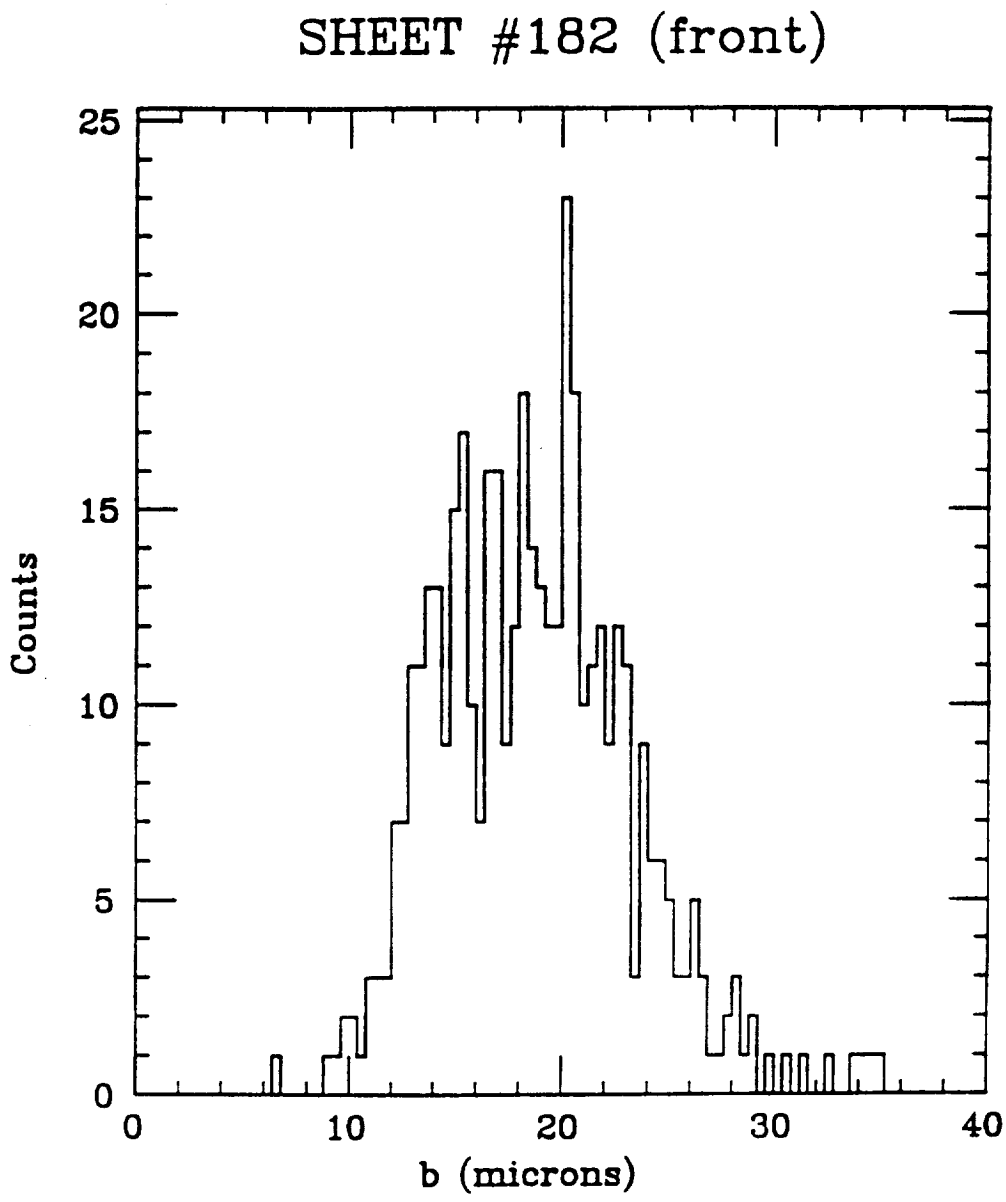


Figure 12. Histogram of track minor axis, balloon data, sheet #182

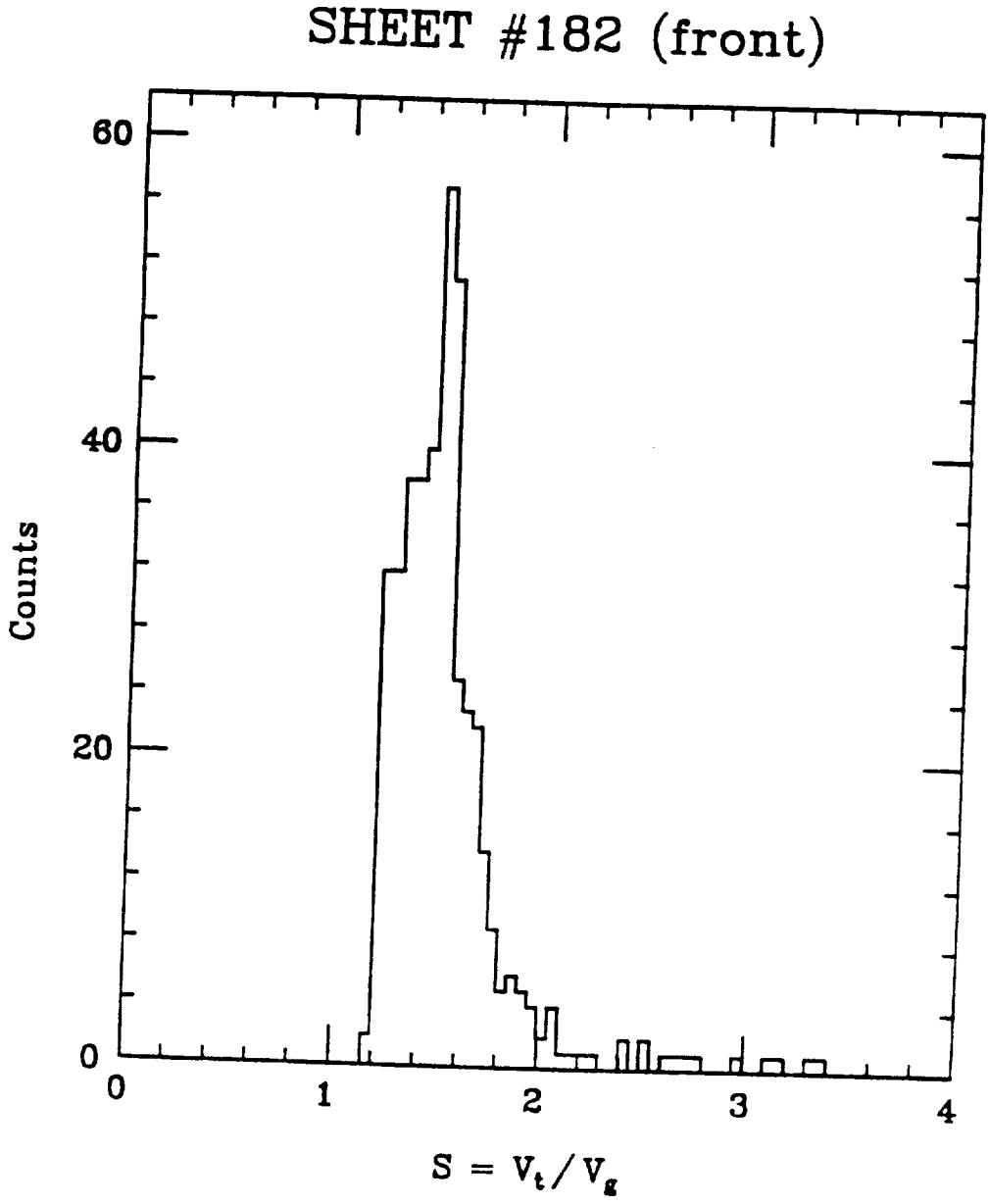


Figure 13. Histogram of s, balloon data, sheet #182

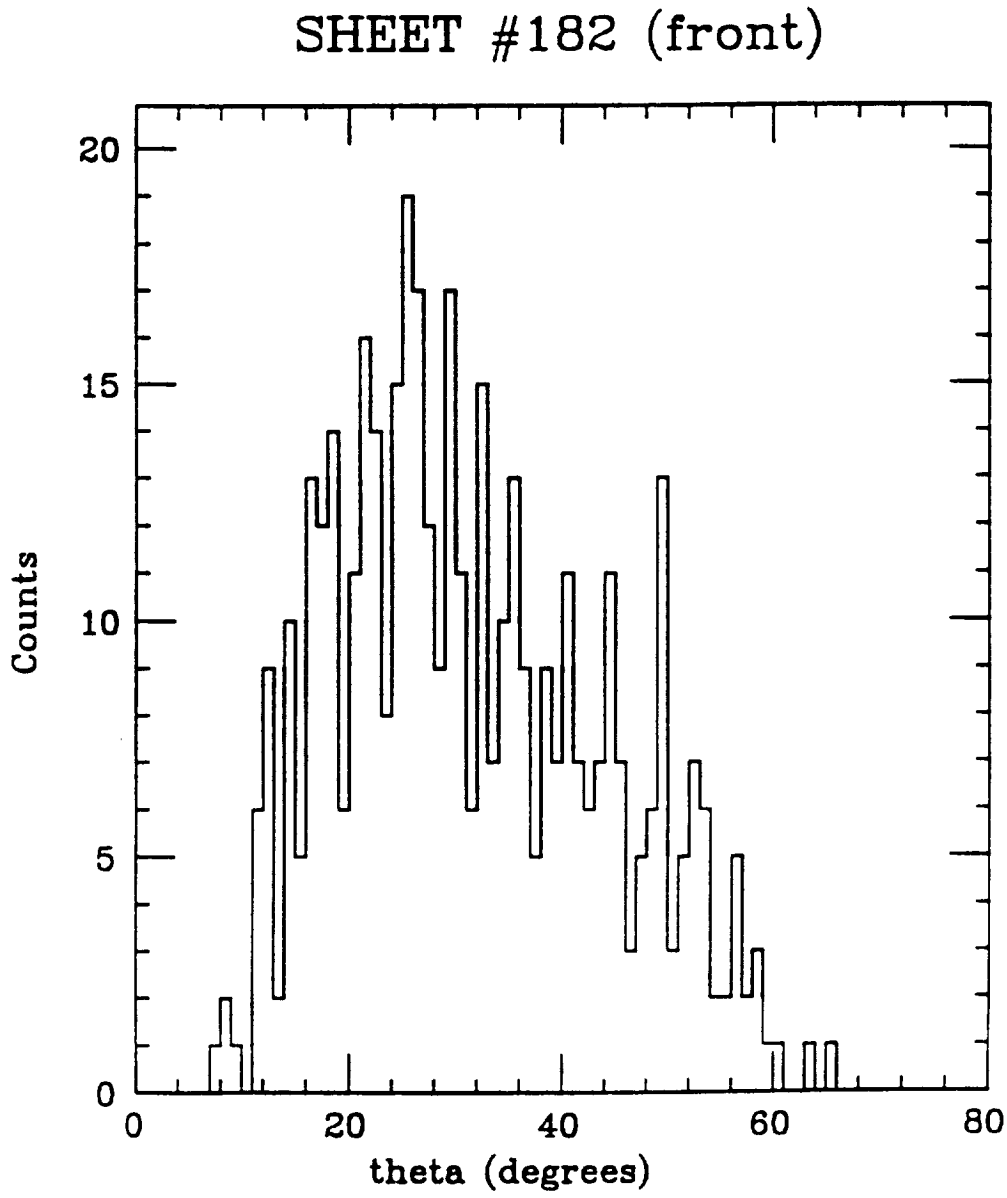


Figure 14. Histogram of zenith angle, balloon data, sheet #182

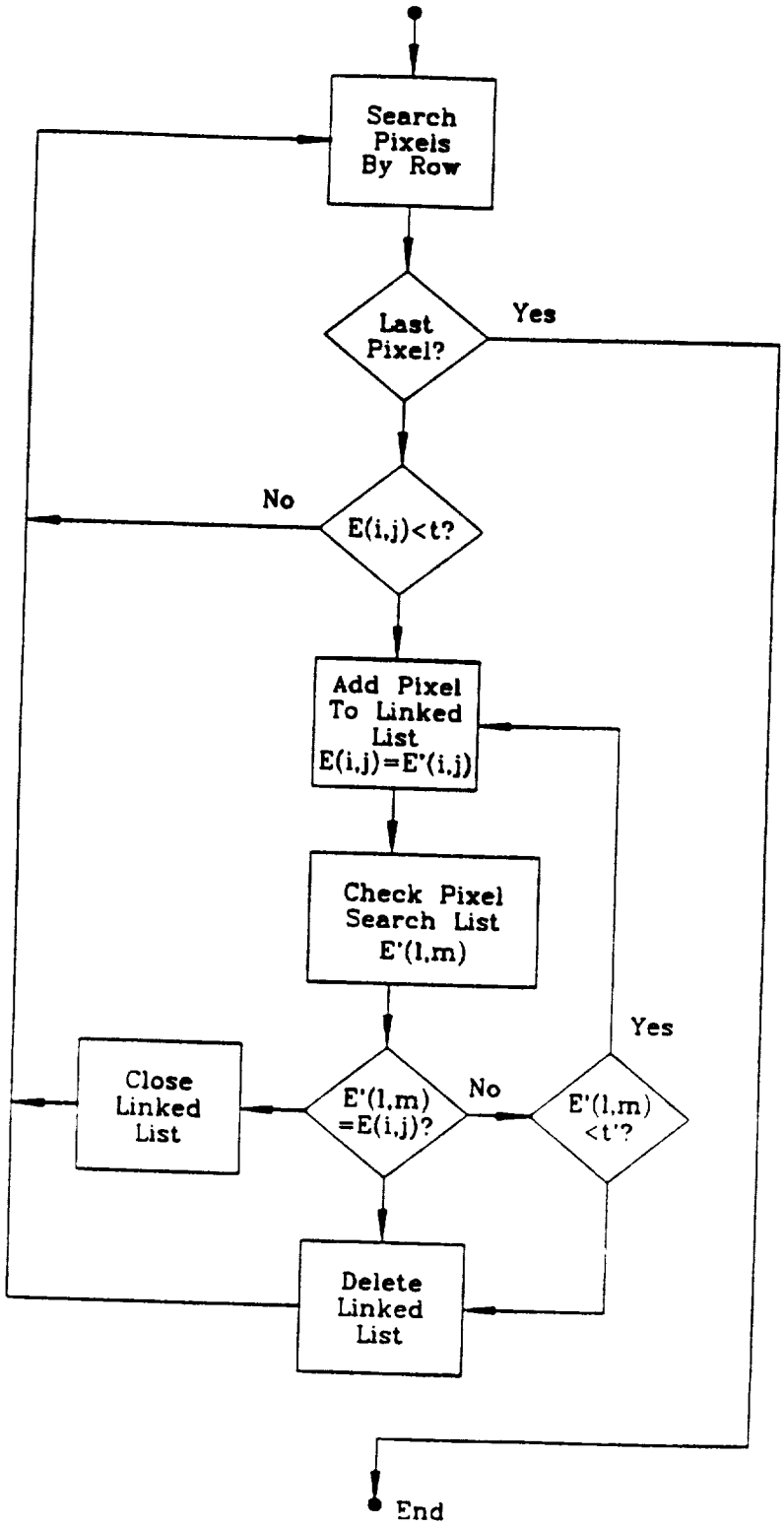


Figure 15. Threshold/tracking algorithm flow chart

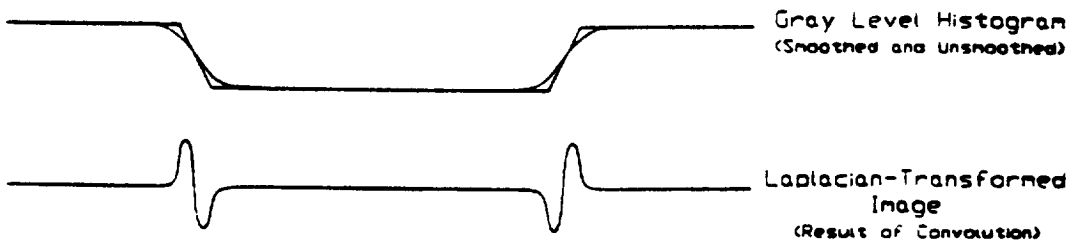


Figure 16. Results of Laplacian operation on an ideal edge

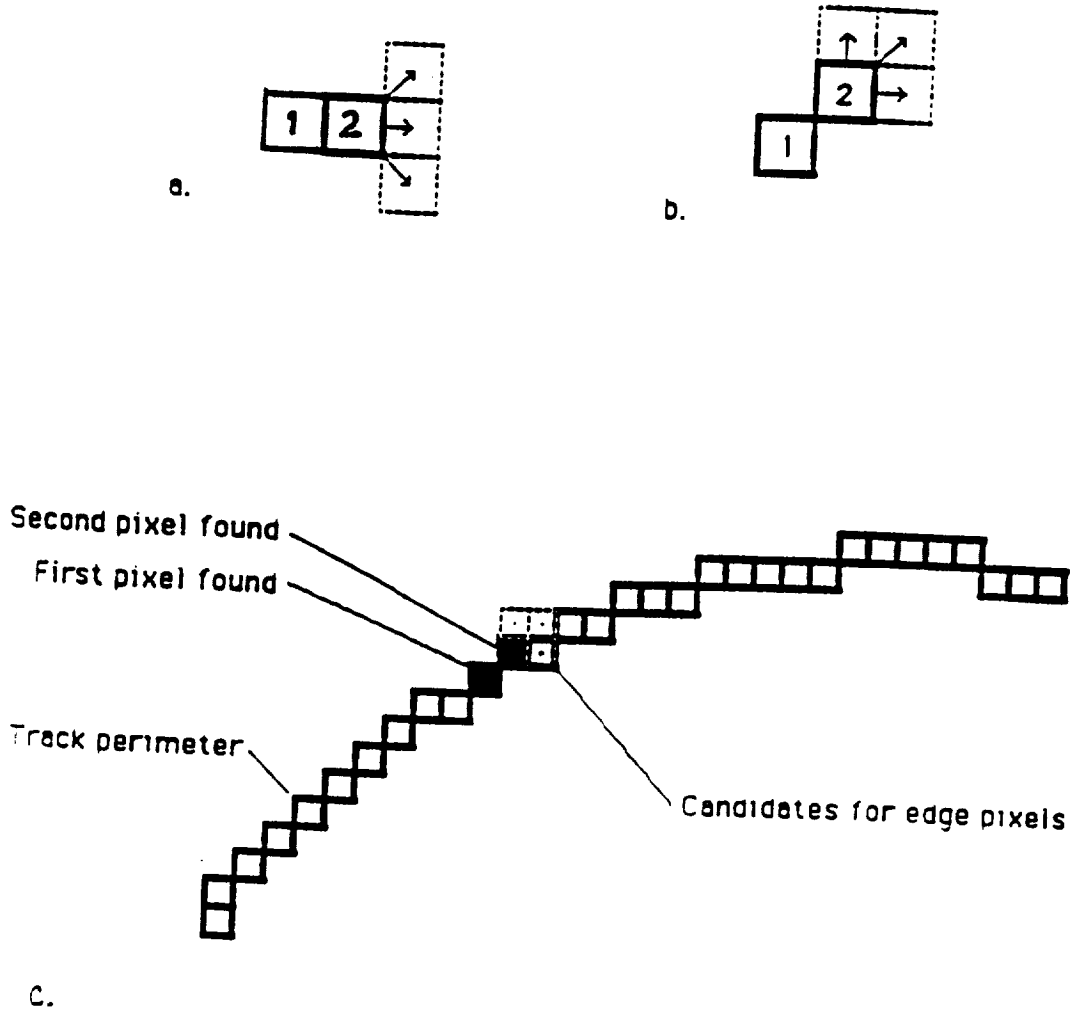


Figure 17. Pixel search list for threshold/tracking algorithm

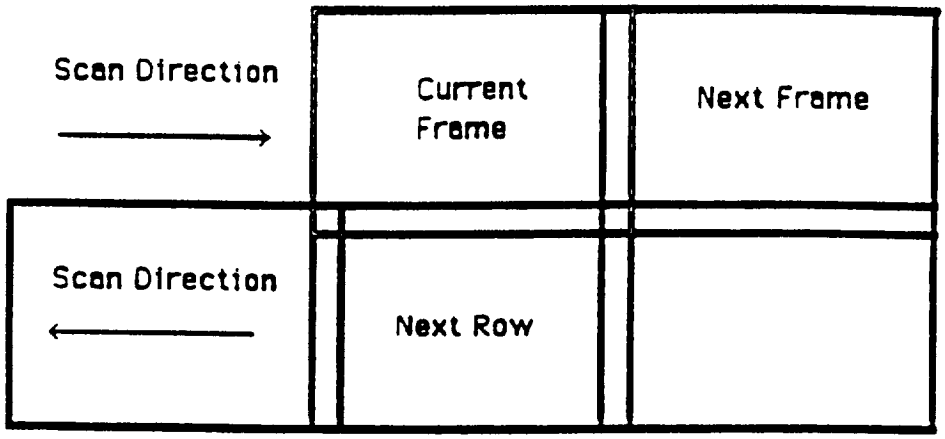


Figure 18. Frame definition for duplicate track elimination

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-182094		

4. Title and Subtitle	5. Report Date
Heavy Nucleus Collector (HNC) Project for the NASA Long Duration Exposure Facility (LDEF)	September 1990
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Gregory Tarle	
	10. Work Unit No. 871-00-00-04

9. Performing Organization Name and Address	11. Contract or Grant No.
The University of Michigan Department of Physics Ann Arbor, MI 48109-1120	NAS1-17820

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Langley Research Center Hampton, VA 23554-5225	Contractor Report
	14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor: William H. Kinard
Final Report

16. Abstract

The primary goal of the HNC experiment was to obtain high resolution composition measurements for cosmic ray nuclei in the platnum-lead and actinide region of the periodic table. Secondary objectives include studies of selected groups of elements of lower charge. These goals were to be realized by orbiting a large area array of dielectric nuclear track detectors in space for several years. In this time we would collect sufficient actinide nuclei to determine the nucleosynthetic age of the cosmic radiation and the relative mix of r- and s-process elements in the cosmic ray source.

The detector was to consist of approximately 50 trays assembled in pressurized canisters. Each tray would contain 8 half-stacks (4 stacks total) and an event thermometer which would record the temperature of each event at the time of exposure. Each stack would contain 7 layers of Rodyne-P, CR-39 and Cronar plastic track detectors interleaved with copper stripping foils.

Upon return to Earth, detectors would be removed for analysis. Ultraheavy nuclei would have left tracks through the detector sheets that would be made visible after etching in a hot sodium hydroxide solution.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
cosmic ray, heavy nuclei, space, LDEF	Unclassified - Unlimited Subject Category 73

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	94	A05