

This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (<u>https://dspace.lboro.ac.uk/</u>) under the following Creative Commons Licence conditions.

COMMONS DEED
Attribution-NonCommercial-NoDerivs 2.5
You are free:
 to copy, distribute, display, and perform the work
Under the following conditions:
Attribution . You must attribute the work in the manner specified by the author or licensor.
Noncommercial. You may not use this work for commercial purposes.
No Derivative Works. You may not alter, transform, or build upon this work.
 For any reuse or distribution, you must make clear to others the license terms of this work
 Any of these conditions can be waived if you get permission from the copyright holder.
Your fair use and other rights are in no way affected by the above.
This is a human-readable summary of the Legal Code (the full license).
<u>Disclaimer</u> 曰

For the full text of this licence, please go to: <u>http://creativecommons.org/licenses/by-nc-nd/2.5/</u>

	Loughborough University of Technology Library	
	AUTHOR BISHOP, NE	
	COPY NO. 0002-63/01	
	VOL NO. CLASS MARK	
· · · · · · · · · · · · · · · · · · ·	-2 11 1983	
	000 0263 01	

Computer Image Processing with

Application to Chemical Engineering.

ЪУ

NICHOLAS EDWARDS BISHOP, B.Tech.

A Master's Degree Thesis.

Submitted in partial fulfilment of the requirements for the award of

Master of Science of the Loughborough University of Technology. July 1972.

Supervisors: P. Allen, Ph.D and

G. Mason, Ph.D

Department of Chemical

Engineering.

C by Nicholas Edwards Bishop.



f

.

•

Summary.

A literature survey covers a wide range of picture processing topics from the general problem of ranipulating digitised images to the specific task of analysing the shape of objects within an image There follows a discussion and development field. of theory relating to this latter task. A number of shape analysis techniques are inapplicable or computationally untenable when applied to objects containing concavities. A method is proposed and implemented whereby any object may be divided into convex components the algebraic sum of which constitute the original. These components may be related by a tree structure.

It is observed that properties based on integral measurements, e.g. area, are less susceptible to quantisation errors than those based on linear and derivative measurements such as diameters and slopes. A set of moments invariant with respect to size, position and orientation are derived and applied to the study of the above convex components. An outline of possible further developments is given.

Acknowledgements.

The author wishes to express his grateful thanks to all members of the Chemical Engineering department who have helped to enable this thesis to be completed. In particular. to Dr. P. Allen who both supervised this research and instigated the image processing project within the department: To Dr. G. Mason, joint supervisor, for a number of stimulating discussions and for painstakingly proof-reading the text of this thesis: To Mr. K. Tilley who has written software for computer operation of the image scanner to obtain digitised images: To Mr. S. Cooper for building and commissioning much of the electronics used in scanner operation: And, not least, to Miss M. Hiorns for typing the manuscript.

Nomenclature.

Unless otherwise defined symbols may be interpreted throughout the text as follows:

d _i , H _i , L _i	Length between two points
g	series filter weights
ij	name of line joining point i
	to point j
dij	length of ij
L	total number of grey levels
MAT	Medial axis transform
MPP	Minimum perimeter polygon
MSP	Minimum sided polygon
Mpq	Moment in p th power of x and
	q th power of y
pq بر	Mpq taken about the centroid
	origin
N	width(length) of a digitised
	picture (in picture points)
pr	probability
r	radial distance from (centroid)
	origin
θ	angle of r to the positive y axis
u, v	variables in Fourier plane
х, у	variables in real plane of
	digitised picture
X(f), Y(f), G(f)	Fourier transforms of x, y and g
x , y	centroid coordinates
2, W	complex variables in the discrete
	Laplace plane

- iii -

List of Figures.

	E E	age.
1	The nature of a digitized boundary.	46A
2	Polar Plot of Boundary on principal axes.	56A
3	Medial Axis Transform of a convex polygon.	37A
4	Medial Axis Transform of a circle,	41A
	ellipse and concave angle.	
5	Five Concavities.	59A
6	Convex Components of a concave boundary	60A
	and their connection tree.	-
7	Elimination of boundary points to find	65A
	convex vertices.	
8	Alternative Segmentations of a concave	71 A
	boundary.	
9	Implicit segmentation of the MAT.	72A
10	Three possible segments of a	85 A
	boundary subset.	
11	Four (lunar dust) particles.	89 A
12	115 sand particles.	91A
13	Two scans of 5 particles from Fig. 12.	91B
	- • -	,
Table	1. Analysis of Lunar Particles.	90 A
Table	2. Comparison of Area, Perimeter and	91 C
	circularity for objects in Fig 13.	
Table	3. Moment Analyses of objects in Fig 13.	92A

- iv -

Contents.

	Page
Summary	1
Acknowledgements	ii
Nomenclature	iii
List of Figures	iv
Contents	v
Introduction	l
Literature Survey	6
Information Theory and Data Handling.	9
Image Coding, Efficient Coding,	9
Sequences, Predictive Coding,	
Run Coding.	·
Data Compression, Fourier Transforms,	12
BIFORE or Hadamard transforms,	
Karhunen Loeve Transforms, Fast	
Transforms, Compression by sampling	59
Reconstruction of pictures by	
projections.	
Picture Processing.	16
Position invariant operation, Point	16
operations, Local operations.	
Hexagonal Pattern (GOLAY) trans-	18
formations.	
Sharpening	19
Digital Filtering	20
2-Dimensional filters in the	
frequency domain.	21
Correlation and Convolution	25

- v ---

	rage
Matched Filtering	25
Image Subtraction	27
Image Restoration	28
Point Spread Functions	28
Recursive Filters	29
Feature Extraction.	31
Ordonnances, Parenthesizing, Principle	32
of Assignment	
Feature extraction from the bulk image.	33
Multiple Linear Regression, Probability	35
of Error, Average Correlation	
Coefficient, Sequential Selection,	
Weighted sum of POE and ACC.	
Blum's Theory (Medial Axis Transform)	37
Scanning Hardware.	43
Development of Theory.	44
Digitised Boundaries.	46
Moments.	53
Boundary Standardisation.	55
Concavities.	59
Machine generation of invariant moments.	67
Segmentation into internal convex sub-area	.s 71
Summary of Findings.	73

Computer Implementation.

Results and Discussion.

vi

89

77

Suggestions for further work	Page 99
Conclusions	103
Appendix A 2-D Integral Transforms.	A 104
Appendix B Glossary of shape describing	B107
terms	
Appendix C List of periodicals.	C1 09
Appendix D Listing of computer subroutines.	D110
Bibliography.	

÷

Introduction.

In the 1950's whilst many dismissed computers as glorified adding machines others failed to appreciate their limitations predicting the 'big brother' machine and humanoid robots. Today, at least the latter of these two predictions appears further from reality than twenty years ago. Indeed, computer development has led man to study in more detail his own thought and reasoning processes. It is mans' combination of eye and brain coupled with manual dexterity which have enabled him to dominate other species.

Like computers, image processing has also been underestimated in potential and in the difficulty of realising this potential. It is remarkable how little has been achieved in this field during the The specific task upon which by past ten years. far the most time, effort and money has been expended is that of character recognition. Some success has been achieved and provided only a pre-determined character set are used a number of moderately priced commercial machines will perform this task with low error rates. Other workers have concentrated upon chromosome and finger print classification, interactive graphic aids, robots and, inevitably, military applications such as aerial reconnaisance.

1

During the same decade the price/performance ratio of small computers has improved by leaps and bounds so that a once expensive novelty is rapidly becoming a commonplace necessity in industry. Unfortunately, too many machines have been sold on the basis of the speed at which they can add two numbers rather than their ability to solve industrial problems. Lack of support in the form of inexpensive peripherals and easy to use software has caused many companies to become disillusioned. One can readily appreciate the frustration of having a sophisticated tool capable of assisting with many important tasks but unable to carry them out for want of specialised 'attachments'.

To avoid the task of interfacing their equipment with numerous small computers, manufacturers of image processing equipment have, in general, supplied their own special purpose Where only relatively simple operations hardware. are to be performed on pictures it is economic to implement them using hardwired circuitry. If units to perform simple operations are produced in modular form then the interconnection of modules may allow some flexibility. However. for more complex operations the increased adaptability of a general purpose computer is undeniable.

- 2 -

In the process industries small computers have assumed many of the tasks previously requiring manual intervention. A limitation has been the relatively few types of information, concerning a process, which could readily be supplied on-line to process control computers. It is hoped that in the future input of visual information will become commonplace, making a valuable contribution to the capability of on-line control systems.

This project is primarily concerned with developing image processing techniques and software for on-line and off-line use on small general purpose computers. Most software has been written in FORTRAN since this language is becoming available on a number of small machines. The techniques used are generally limited most by the data handling facilities, however, so sections which manipulate data in other than single dimensioned vectors have been isolated as small routines. In particular, it has been a policy to make all computer routines of moderate size so that work involved in modification or conversion for another computer would be minimised.

In choosing the subject of research for this thesis several lifetimes of work have readily suggested themselves. The difficulty has been in picking a path amid a maze of false trails and blind alleys in a direction most likely to prove

- 3 -

profitable, moreover, a direction in which a useful amount of development could be implemented in the relatively short time available. The final topic was selected from observations of the shortcomings of work by other researchers and, it is hoped, will prove to be a means of overcoming some of these difficulties.

Literature Survey.

The computer processing of images encompasses such a proliferation of literature that a comprehensive survey would occupy numerous volumes. Fortunately, other authors have surveyed sections of the subject and unnecessary repetition has been avoided.

Every effort has been made to cover the most recent works of authors and research teams, but with new articles appearing each week no such work can remain up to date. A guide to periodicals which frequently contain relevant information has therefore been included as Appendix C.

Special mention must be made of 'Picture processing by computer', by Azriel Rosenfeld(1). This textbook has been used as a framework to introduce basic concepts whilst mentioning more recent articles in the first two sections of the survey. Several other pertinent books have been published, mostly collections of conference papers.

The series of publications 'Advances in Information Systems Science' in which prominent authors discuss specific topics have also been found invaluable. Information has frequently been drawn from this source in attempting to link discussion of various subjects in a logical manner. It is hoped that this survey will provide a useful background knowledge of picture processing to Chemical Engineers (not least the author). The aim has been to complement rather than duplicate topics familiar to Chemical Engineers whilst never straying far from subjects of direct application. Nevertheless, anyone with a fair knowledge of engineering principles and mathematics should still find the survey readable.

The survey consists of three main sections of which the first, Information Theory and Data Handling, is the most general. It is difficult to understate the importance of laying good foundations in the form of careful selection of data structures if image processing procedures are to be implemented efficiently within a limited storage space. The first section reviews this problem in abstract terms before concentrating on the more specific task of coding images.

Picture Processing, the second section, describes a number of operations which may be performed on pictures to enable either a human observer or a computer to more easily discern items of interest. Techniques described have been developed by workers in statistics, optics, and electrical and audio engineering: Differences may be seen to be more in the methods of approach

- 7 -

resulting from a diversity of purpose than in the actual transformations themselves. Researchers in picture processing have increased the usefulness of these techniques by adding logic functions.

The next section dealing with feature extraction reviews work on the problem of measuring picture properties and assessing the relation between such measurements and properties to be estimated. Treatment of the former topic is restricted to avoid overlap with 'Development of Theory', the next part of the thesis.

Finally, brief reference is made to scanning hardware.

INFORMATION THEORY AND DATA HANDLING.

A primary problem in the processing of images is the vast quantity of data arising from the digitisation of a single picture. In this section a number of methods for reducing this volume are discussed and compared.

A digital picture is usually considered as a matrix of points (pixels) each representing the average grey level of a small area. If the number of possible discreet grey values is L=2^X then the total storage requirement for an N by N pixel picture is N².x binary bits, regardless of picture content. Considering a binary image (L=2), where 'l' represents a white point and 'O' a black point, then a completely white picture could ideally be represented by a single 'l' bit. At the other extreme a picture composed entirely of random dots would have a maximum information content, there being no way of predicting the value of any point with more than 50% accuracy, i.e. using standard statistical notation for probability, $pr(0) = pr(1) = \frac{1}{2}$.

In general the grey value at a point may be predicted with better than 50% accuracy from a knowledge of neighbouring points, see for instance Deutsch(2). Simpson(3) has used statistical methods to assess the degree of predictability of individual points and hence the redundancy of information contained in our N^2 .x bits.

-9.

Efficient Coding.

Huffmann(4), shows how storage requirements may be minimised by using a code in which the length of the number denoting each value is inversely proportional to its likelihood of occurence. Suppose there are three grey levels: '1', '2' and '3' and $pr(1) = \frac{1}{2}$, $pr(2) = \frac{1}{4}$, $pr(3) = \frac{1}{4}$. We may use a code only half as long for 'l' since this value is twice as frequent. Let the binary numbers 1, 01 and 00 be the codes for 1, 2 and 3 respectively. Summing the product of code length and probability for each code gives the average code length which will be $1.\frac{1}{2}+2.\frac{1}{4}+2.\frac{1}{4} = 1\frac{1}{2}$ binary bits. Note that no separators are required between successive codes since a code starting with 'O' is known to have 2 bit length and code starting with 'l' only single bit length.

Sequences.

If the probability that the next grey value will be J is to some extent dependent on the current value I then pr_{JI} is called the conditional probability for this occurence. If $pr_{JI} = pr_{J}$ for all pairs of points then the points are said to be independent. Where grey values are interdependent a single code may be used for à predetermined sequence. The length of code should be related to the frequency of occurence of the sequence.

Predictive Coding.

Frequently, neighbouring points will have a high probability of being the same, i.e. pr_{II} > pr_{JI} for I \neq J. For such cases the first grey value

· 10 -

and successive differences may be stored. The difference used may be shifted into the range 1 to L using J-I for J>I and J-I+L for J&I. The relative frequencies of each value will be less equally distributed allowing more efficient coding. An interesting method used by Schwartz(5) takes into account lower frequency of change for the more significant binary digits of successive values. Run Coding.

Where long runs of the same grey value occur it becomes efficient to store value and run length. For binary images, only the run length is required since black and white will alternate.

Data Compression.

The above methods allow a reduction in storage space whilst retaining all the original Unfortunately, efficient coding may information. increase the difficulty of image processing and Since only a small number of final manipulation. results are normally required from each image. redundancy of required information will be much higher than redundancy of pictorial information. It is important to consider the type of processing required together with the above trade-offs in code selection. In practice it may be acceptable and even desirable to discard some of the data in the interests of further compression.

Digitisation produces quantisation errors in both spatial and grey level measurements. Data compression may include smoothing operations which actually reduce this error.

Fourier Transforms.

Slow variations in picture shading and sharply defined edges may be thought of as low and high spatial frequency picture components. Various frequency transformations are available, the best known being the Fourier transform. Since two dimensional Fourier transforms are not well covered in standard texts some useful notes are collected in Appendix A. Andrews(6) discusses the use of Fourier image coding. Since the transform of each point involves the weighted sum of every point on the original image, distortions in the transform

- 12 -

plane become diversified and hence less visually objectionable when the original image is reconstructed.

The transform produces complex results, real parts being dependent on magnitude of the frequency components and imaginary parts on their phase or position. Low frequency components have, in general, much greater magnitude than high frequency ones so that storage of the transform image can be compressed by suitable choice of code. BIFORE or Hadamard Transforms.

In this case the approximating functions are rectangular waves rather than sine and cosine functions. The Hadamard transform produces results which are both position and magnitude dependent. However, in a series of recent articles, Ahmed and Rao(7) have shown that it is possible to construct magnitude and phase spectra. Magnitude terms may be interpreted as the average power of the sum of a fundamental frequency and its odd order harmonics. Unfortunately, because of this summation, the ability to reconstruct the original image is lost. For a fuller description see Appendix A.

Karhunen-Loeve Transforms.

For any group of pictures there is an optimum linear transformation for data compression. For an N by N picture it is necessary to calculate the N^2 by N^2 covariance matrix. This is then diagonalised and the eigenvectors found.

- 13 -

Normally the eigenvalues are ordered in magnitude and a relatively small number retained. Corresponding eigenvectors only are calculated, nevertheless computational effort and storage requirements are huge. Van Emden(8) summarises the way in which various research workers have used different criteria each finding this transformation optimum. In practice it is useful to compare the performance of other compression methods using Karhunen-Loeve as a standard. Andrews(9) recently published a comparison of the three transformations mentioned here together with the Haar transform. As might be expected transforms requiring more computation produce more efficient compression. However, in cases where less than 80% of the information content need be retained he found the difference between Fourier and Hadamard transforms only marginal. Fast Transforms.

By storing intermediate results it is possible to reduce computation required to perform Fourier or Hadamard transformation of a whole picture from N⁴ to $2N^2 \log_2 N$ operations. Each operation is essentially a complex multiplication for a Fourier transform and only an addition or subtraction for a Hadamard transform. Effectively, the Fourier transform uses N different weighting factors, i.e. $\sin 2\pi x/N$ for x = 1, 2, ..., N whilst the Hadamard uses only two, namely, *1, and -1. Hence the term BIFORE or binary Fourier representation.

- 14 -

Compression by Sampling.

Where other methods are unavailable compression by random sampling may be used. Particular mention should be made of the various applications of repeatable pseudorandom sequences. If pseudorandom noise is added to a digital picture prior to grey level quantisation(10) and on reproduction the same noise is subtracted the effect is to alter the quantisation levels in a random manner. This successfully breaks up 'false contours' which would otherwise be produced.

Reconstruction of Pictures from Projections.

If the grey levels for each vertical line of a picture are summed the resulting vector is the vertical projection of the image. Any number of projections may be found in different directions. Gordon(11) has suggested a novel method employing

Monte Carlo technique to produce an image which satisfies the constraints imposed by the projections. Such an image will not normally be unique and he obtained the best results by averaging about twenty images produced using the same constraints. An important application of this technique is in the reconstruction of three-dimensional translucent objects from 2D images at a number of angles. Each point is considered as the projection in the plane normal to the image.

15 -

PICTURE PROCESSING.

A digitised picture may be considered as data and, as such, manipulated by mathematical methods. Picture processing takes into account the pictorial nature of this data and is concerned with the change in appearance of the picture caused by processing.

This section starts with basic definitions, then various methods for the removal of useless data and emphasis of selected classes of features are discussed. Digital filter design, matched filters, and deconvolution techniques will be considered in turn and an attempt made to show the equivalence of these procedures.

Position Invariant Operations.

If the net result of an operation is the same whether the picture is operated upon directly or subjected to complementary shift operations before and after, then the operation is position In this context, the shifting of the invariant. picture matrix is cyclic, i.e. any elements which are shifted beyond the edge of the picture reappear on the opposite edge. This corresponds to the Fourier image representation as a periodic waveform which is repeated to infinity in both X and Y directions. Shift operations correspond to the post-multiplication or pre-multiplication of the picture matrix by a modified unit matrix of the same order.

- 16 -

Point Operations.

If the grey level value of each point on a picture is rescaled by linear transformation, image brightness and contrast are altered. In certain cases (12) cyclic non-linear rescaling may prove advantageous, i.e. false contours are deliberately introduced by dividing the original grey scale into several grey scale ranges. Fine detail in low contrast pictures is greatly enhanced but noise is also amplified.

Local Operations.

These depend not only upon the grey level of the point undergoing the operation but also those in a local neighbourhood, e.g. a local 3x3 Such operations may be mathematical matrix. transformations or include logic functions. To reduce random speckling of images the average grey level of the 8-near neighbours of a pixel may be compared with the value of the pixel itself. If the difference exceeds a threshold the average is substituted. Sklansky(13) has shown that in general the convolution and thresholding operations are not equivalent, hence, it is not possible to find a non-zero aperture convolution which is equivalent to thresholding.

Picture operations may be considered as the mapping of points of the original image onto a point or points of a new transformed image. To conserve space it is frequently convenient to

- 17 -

operate sequentially on each pixel of a single image. Rosenfeld and Pfaltz(14) have demonstrated that parallel operations may be reformulated as equivalent sequential operations.

Hexagonal Pattern Transformations.

A rectangular array of picture points has a number of disadvantages when used in picture processing. One is that the concept of connectivity is not easily defined, since each point has 4, 4-near (horizontally or vertically adjacent) and a further 4, 8-near (diagonally adjacent) neighbours. If connection requires a continuous path of 4-near points then the connectivity of two 8-near points may only be determined by studying the two mutually 4-near neighbours. Conversely, if a continuous path of 8-near points is deemed sufficient it is possible for the connection paths of black and white objects to cross.

In hexagonal arrays all six near neighbours are placed in geometrically similar positions. Golay(15) enumerates the 14 possible binary configurations of 6-near neighbours to a picture cell, Subject to rotation invariance. This compares with more than 70 configurations for the 8 neighbours of a rectangular array.

If each cell composed of 6-near neighbours is tested and assigned a value in the range 0-13 according to its configuration and the value of this number is used to either change or not change the value of the central point then interesting

- 18 -

pictorial transformations occur. In order that serial and parallel operations are not confused the image may be divided into 3 or more subfields so that in each subfield any point is either a surround point or a centre point but not both. Each subfield is operated upon in turn. Repeated application of these Golay transforms may be used for filling holes, counting blobs, 'smoothing' contours and skeletonisation.

Golay et al(16) have constructed a digital logic unit (GLOPR) capable of efficiently performing Golay transforms on pictures held in computer store. The system includes a T.V. scanner, monitor and magnetic tape storage device. This equipment has been used mainly in the classification of blood cells and is marketed as CELLSCAN. Associated software is written in a special Golay Logic Language (GLOL).

Sharpening.

Whereas smoothing involves substitution of the integral of local grey levels for a central point, sharpening is concerned with local grey level derivatives. Pingle(17) gives a simple formulation for the direction and magnitude of steepest grey level gradient using the 3x3 matrix of local points. The sum of the second partial derivatives, or Laplacian operator, may be approximated by the difference between the average of a number of local points and the central grey

- 19

value, i.e. the image subtracted from a blurred copy of itself. Both gradient and Laplacian operators emphasize edges and fine detail but also increase noise.

When used as position invariant operations smoothing and sharpening represent an alteration of the trade-off between resolution and signal to noise ratio. If, however, logical procedures are applied, particularly where the statistics of the picture are well known, a more positive gain in image quality may be achieved. One sophisticated technique includes a test to detect the presence of an edge. Smoothing operations are used at right angles to, or in the absence of, an edge whilst a sharpening procedure is simultaneously applied across the edge itself.

Digital Filtering.

By considering a picture to contain two components, useful information and unwanted information or noise, we may attempt to design a filter which, ideally, would remove all the noise leaving wanted information intact. First it is necessary to study the nature, i.e. the statistics, of the two components. The more disimilar some property of these components the greater will be the success of a discriminent function.

It follows naturally that where pictorial information must be transmitted over a noisy

- 20 -

channel the picture should first be processed to have minimum correlation with the known noise. Optimum separation of noise and picture may then be achieved when the original image is recovered. 2-Dimensional Filters in the Frequency Domain.

If the frequency spectra of wanted signal and unwanted noise are sufficiently different then frequency domain filters may be used successfully. In an earlier section Fourier transforms were discussed and the parallel between grey level transitions and spatial frequencies mentioned.

Seltzer(18), gives a simple procedure for calculating the frequency response of a weighted average of grey level values applied as a convolution operator to each picture point. Consider a filter in which each point along a line is represented by the average value of the three points centred about itself. Let the weights be termed g and the initial and transformed series of points be x and y, then:

$$y_n = \sum_{\substack{k=K+1\\k=K-1}}^{k=K+1} g_{k} \cdot x_{n-k} \quad (EL).$$

where $g_{k-1} = g_k = g_{k+1} = 1/3$. The Fourier transform of y_n can be expressed as:

- 21 -

 $Y(f) = \sum_{n=0}^{n=N-1} \cdot \sum_{k=K+1}^{k=K+1} g_{k} \cdot x_{n-k} \cdot e^{-2\pi j f n}$ $= \sum_{k=K-1}^{k=K+1} g_{k} \cdot \sum_{n=0}^{n=N-1} x_{n-k} \cdot e^{-2\pi j f n}$

where f is the fractional frequency i.e. f=u/N, where u is the frequency, and the distance between neighbouring points is considered to be unity. Now let m=n-k so that n=m+k, hence:~

$$Y(f) = \sum_{k=K-1}^{k=K+1} g_k \cdot e^{-2\pi jfk} \cdot \sum_{n=0}^{n=N-1} x_m \cdot e^{-2\pi jfm}$$

= G(f).X(f)

i.e. the input and output of the filter are related in the frequency domain by the filter transfer function G(f) which is the Fourier transform of the filter weights. Now:-

$$G(f) = \sum_{k=K-1}^{k=K+1} g_k \cdot e^{-2\pi jfk}$$

$$\frac{1/3}{e^{+2\pi jf} + e^{0} + e^{-2\pi jf}}$$

$$\frac{1/3}{\cos 2\pi f + j\sin 2\pi f + 1 + \cos 2\pi f - j\sin 2\pi f}$$

$$\frac{1/3}{1 + 2\cos 2\pi f}$$

This simple derivation shows a number of basic properties of digital filters. Firstly the convolution theorem, i.e. the convolution of equation (E1) is equivalent to element by element multiplication of the transform filter weights in This property holds for the frequency domain. a number of integral transforms including Laplace but it is particularly interesting in the light of fast Fourier transformation since it may require less computation to transform an image, filter, and perform inverse transformation than to apply the same filter as a convolution operation in the image plane. Secondly, we note that it is the symmetry of the filter weights about the central point which causes imaginary (phase shift) terms to cancel out. Applying this constraint to a general 3x3 matrix whose weights may be represented and using the same derivation c b ch as: bat

in two dimensions with the plane x,y transforming to the Fourier plane u,v we obtain the transfer function:

 $G(u,v) = a + 2b(\cos 2\pi u/N + \cos 2\pi v/N) + 2c(\cos 2\pi (u+v)/N) + \cos 2\pi (u-v)/N)$

substituting u'= $2\pi u/N$ and v'= $2\pi v/N$ this simplifies to:-. $G(u,v) = a+2b(\cos u'+\cos v')+2c.\cos u'.\cos v'(E2)$.

- 23 -

If the illuminance of the filtered image is constrained to be unchanged then only two independent variables remain. Ideally, the amplitude response of such a filter should be equal in all directions, equation (E2) shows that this ideal may only be approximated for any finite size filter matrix (unless b=c=0).

Where the problem is to apply a filter of a given frequency spectrum to a digital picture two methods are apparent. Firstly, the transformed image may be weighted by the frequency terms and an inverse transformation made. Alternatively. the inverse transform of the spectrum may be taken and convoluted with the image. The convolution filter will be an N by N matrix but may be approximated by a matrix of lower order. No straightforward method for optimizing this approximation is known. If the matrix terms are truncated at a given order the frequency response of those remaining are subject to oscillation or Better results are achieved when the 'ringing'. filter matrix is itself weighted with coefficients whose values decrease with distance from the centre Where the filter is symmetric, yet another point. method is to fit the coefficients of the known (cosine series) form of filter response to the desired spectrum using the method of least squares.

- 24 -
Correlation and Convolution.

It is frequently required that a particular feature or shape which may occur within an image should be enhanced or that two images should be compared for similarities. This infers the correlation of two pictures or a picture and filter matrix. In fact the filter may be thought of as a picture to which the subject image must be matched. If the two series used in equation (El) are subjected to correlation the result may be expressed as:

 $y_n = \sum_{k=K-1}^{k=K+1} g_k \cdot x_{k-n}$ (E3).

Taking Fourier transforms this simplifies to:

 $Y(f) = G^{*}(f).X(f)$ where $G^{*}(f)$ is the complex conjugate of G(f). If the series g is symmetrical so that the terms in G(f) are real only then it is obvious that correlation and convolution operations are identical.

Matched Filtering.

One would have thought that the above explanation of correlation led directly to matched filter implementation. Namely, produce a filter matrix which is a picture of the required feature and correlate with the given picture.

- 25 -

Alternatively, find the complex conjugate of the filter transform and use a normal digital filter. Unfortunately, two major snags arise. Firstly, although the filter operates regardless of the position of the required feature on the image it is not invariant to changes of size or orientation. In some cases this limitation is not important e.g. in typewritten character recognition where both size and orientation are known. Orientation invariance may be achieved at the expense of mapping the image into an orientation invariant plane e.g. using polar co-ordinate Fourier transforms but it is still necessary to apply separate filters for each size range of the required object.

E ach point on the filtered image is effectively a similarity index whose value reflects the 'goodness of fit' of the required feature centred at that point. A perfect fit will be obtained for any object whose boundaries totally engulf the required feature. This may be overcome by preprocessing the image so that the required feature becomes a contour outline, e.g. by differencing neighbouring picture elements. Provided such an operation may be expressed by a simple matrix it may be combined with the matched filter. Combination filters are often ambiguously refered to as matched filters.

An interesting property of the matched filter is that it is also optimal for finding

- 26 -

distorted copies of a perfect filter shape, or conversely in seeking a less distorted copy of a given image.

Image Subtraction.

This technique is used to show up small differences in different photographs of the same scene. Correlation techniques must be used for precise registration of the two images prior to subtraction and additional difficulties occur from geometrical distortions. Barnea(19) details a method for rapid image registration.

Image Restoration.

If a one point picture can be subjected to the same position invariant degradation as a picture which it is required to restore then clearly the operation required is one of deconvolution using the one point picture as a 'blue print'. Deconvolution may be performed by dividing the Fourier transform of the image by that of the one point picture and taking the inverse of the result. In practice, good results are highly dependent on the original distortion being precisely position invariant, usually only low frequency terms of the two pictures are divided.

Point Spread Functions.

Digitized images are normally degraded Blurring, due to imperfect in the following ways: focussing and inaccurate scanning; Noise from the photosensing head and quantisation, and imperfections in the scanned image. If the degradations can be mathematically modelled compensation may be made using an inverse function. Using the blue print above as a starting point MacAdam has derived an interesting on-line method for improving image quality. Adjustments are made incrementally to the restoring function. At each stage comparisons may be made on a T.V. monitor. The number and range of variable perturbations are greatly reduced by the programme which applies the constraints of

- 28 -

circular symmetry to the restoring function and a predetermined illuminance range to the grey levels of the restored image.

Recursive Filters.

In equations (E1) and (E3) the filtered output series y is derived from the input series x and filter coefficients g. It is possible to design filters in which the output is also dependent upon previous output terms . Unlike time series signals the meaning of 'previous' is open to arbitrary interpretation when applied to pictures. A close analogy exists in the differences between non-recursive and recursive filters and open and closed loop control systems. Recursive filters may produce a more desirable frequency response using fewer filter coefficients but the stability problem must be considered. Usually, recursive filters are designed using 2-dimensional discrete Laplace(zw) transforms, although Fourier methods may be applied (20). The frequency response is generally complex so that both magnitude and phase Hall(21), gives a formulation must be constrained. and illustrates results obtained using a 2nd order Butterworth, maximally flat, recursive filter. In particular, an illustration of oscillation of shading caused by non-linear phase response, i.e. phase shift not proportional to frequency, is given. The filter uses only the unfiltered centre point and two previous filtered values.

- 29 -

Unfortunately, literature concerning digital filtering, especially where applied to images is highly fragmentary. In addition to the works of Andrews(6), Selzer(18) and Rosenfeld(1) previously mentioned two special issues of I.E.E.E. Transactions on Audio and Acoustical Engineering(22) have been found particularly useful.

Feature Extraction.

In previous sections we have considered the suppression or removal of unwanted information and enhancement of features either for the benefit of a human observer or to simplify further computer processing. Some interesting work has been described by Macon(23) on the removal of objects from an image so that neither object nor the resulting space are visually apparent. In this section, however, we are primarily concerned with carrying the enhancement process one stage further so that qualitative measurements may be made and analysed.

Much work has been concentrated on the extraction and classification of chromosomes from digitized images. One research team, headed by Ledley(24), uses a FIDAC scanning system employing a T.V. Scanner to digitise photographic transparencies which are input directly to a computer. Various approaches have been made capitalizing on well defined chromosome features. Klinger et al (25) present the most recent work on this subject and review previous research. An edge tracing algorithm is used and the boundary point list converted to polar co-ordinates with centroid origin. Since chromosomes consist of a number of arms emanating from the centre, the polar plot has a number of peaks at or near the arm tips with heights related to arm length. Unfortunately, the plot $r=f(\theta)$ is not necessarily single valued and there is no guarantee that the origin will be within the enclosed

- 31 •

region boundary. In certain circumstances, however, it is possible to classify regions as two overlapping chromosomes.

Zahn(26) describes an interesting method for analysing the boundary point list. Minor adjacent and opposite changes in directions called inflections are located and deleted. The remaining features are classified according to length of straight segments and radii of curvature. Curves in the same direction in close proximity are grouped as one. A number of higher order features are defined in terms of simpler ones using Backus normal form, a language first used by compiler writers. This hierachical language structure enables high level descriptions of objects to be derived and hence easy comparison and classification. A number of shape describing features are defined and appear together with some from other sources in Appendix B. Ordonnances.

If the N boundary points of an enclosed region are numbered from 1 to N in, say, clockwise order, there are N(N-1)/2 lines joining pairs of points. These lines are called Ordonnances. Let the line joining i to j be ij of length dij and the set of ij's be ordered in decreasing dij sequence. <u>Parenthesizing.</u>

The ordered ij sequence described will not necessarily correspond exactly to similar patterns with minor distortions. Simon et al.(27) give details of a data structure to overcome this limitation.

- 32 -

An error tolerance is decided upon and the sequence parenthesized to show the equivalence of all possible sequences within this tolerance, e.g. A(B C(D)E)F G. This means possible sequences all have A first with B, C or D second. B and C must preceed E although E may itself preceed D.

Principle of Assignment.

When a pattern is being compared with a reference parenthesized list each member must be assigned such that the parenthesis rules are not broken. If only one assignment of corresponding lines of the two patternsconcur with these rules the assignment is said to be possible. When many or zero assignments are available they are described as undetermined and impossible, respectively. As the error tolerance is increased assignment may go from impossible through possible to undetermined. Sometimes the transition from impossible to undetermined is direct.

Feature Extraction from the bulk image.

It has been asserted (1), that the statistics of grey level variations in local areas are related to the textural appearance. Hence by considering sub areas, differences in texture can be determined. Taking moments to a sufficiently high order each sub area could be uniquely characterised although, for ease of computation, only low order statistics are normally used.

If the picture is converted to a binary image by thresholding, contiguous points above the threshold may be considered as chords. If these

- 33 -

chords are measured in any direction then the statistics of their size distributions may be compared between different subareas, at various thresholds, and for different directions. The properties which may feasibly be assessed in this manner include size distribution, orientation, elongation, spacing, variations of packing density, permeability, porocity and surface area. Unfortunately, such properties cannot generally be measured in a direct manner and it is necessary to find relations between measurements and the parameters required. This is essentially a problem of pattern recognition.

Pattern recognition techniques are usually applied by using a training set for which both measurements and parameters are known. An attempt is made to find either a measurement or a linear combination of measurements highly correlated to a parameter. Among better known techniques are multiple linear regression, principle components (Eigenvalues) and factor analysis. The selection of measurements used in the estimation of the parameter may be completely empirical or based on a known physical interpretation of the measurements. The physical interpretation of low order moments are well known hence their frequent use. Since the parameters required are often chosen in order to estimate other parameters e.g. Surface Area to estimate reaction rate, it'is desirable to short cut this process by estimating the required parameters directly, wherever possible, thereby reducing computation and rounding errors.

- 34 -

Mucciardi and Gose(28) compare seven of the more sophisticated methods of choosing 'best' subsets of a given property list for the purposes of pattern recognition. A brief outline is given below.

Multiple Linear Regression.

The list of properties for each object or feature are considered as a point in N-space. The first property selected is that which minimises the sum of the squares of the distances between, property points with all other co-ordinates set to zero and the corresponding N-space property points, to an arbitrary multiplicative constant. The process is repeated to choose subsequent property co-ordinates and their constants. Disadvantages are that the approximation will not necessarily approach a global optimum unless each property variable is unimodal, and that relationships between variables are not considered.

Probability of Error(POE).

The POE of each variable used by itself to classify the objects is considered. The property for which this value is a minimum is selected first. Second, third and fourth properties are those with minimum POE from the set remaining. Average Correlation Coefficient(ACC).

The first property is chosen using the POE test but the second is that with minimum correlation with the first and subsequent choices are of variables with minimum average correlation coefficients with properties already chosen.

25.

Sequential Selection.

Again, the first property is selected using POE but successive variables are those which least often misclassify the object class for which the error rate is highest using the set chosen so far.

Weighted sum of POE and ACC.

The lowest weighted sum of POE and ACC is chosen as the next variable. Mucciardi found the sum 0.1.POE + 0.9.ACC to give best results and that this method was more economical in computation than the Eigenvalue(Karhunen-Loeve) method described earlier.

Blum's Theory.

Perhaps the most interesting of all picture transformations is that first suggested by Blum(36). Each point on the object boundary is considered to initiate circular waves i.e. similar to those caused by a stone in a mill pond. The wave front produced advances parallel to straight sides but interferes where convex corners occur. The interference paths or medial axes together with distance from boundary values give a complete, reversible transformation of the original boundary. Another representation maps each point in the plane of the boundary into a third dimension, 'height', equal to the distance from the nearest boundary point. Medial axes are formed along ridges and troughs of the resulting 3-D figure.

The medial axis transform, or skeleton, of a convex polygon has a particularly simple form. Fig. 3 (a) shows the skeleton and 3 (b) the propagating wavefront of the original boundary. As the skeleton is formed sides of the original figure disappear from the wavefront, producing skeletal branches. Finally the wavefront vanishes to a point. The geometrical construction of the skeleton is straightforward. Skeletal lines are bisectors of adjacent boundary sides. When two bisectors cross a branch is formed and the included side disappears.

. 37 .



Rosenfeld's(37), implementation of this transform uses the digitised picture stored in Each pixel is replaced by a number metrix form. representing its distance from the nearest boundary Unfortunately, the distance value refers point. to the number of near neighbours on the shortest path to the boundary with 8-near and 4-near distances considered equal. If points with lower height values than one or more neighbours are removed a skeleton is produced. Although the original object may be regenerated, this skeleton is not invariant under Rutovitz(38), shows a simpler method of rotation. implementating this Rosenfeld-Blum transformation and discusses a number of further developments. If a skeleton is produced from a digitised image with a number of grey levels then the height coordinate may be weighted according to the grey values on the nearest path to the boundary, i.e. a grey weighted distance transform. Rutovitz also suggests a path-with-distance transform in which the skeletal points are augmented by pointers, in each of the other pixels within the boundary, directed along the nearest path to the boundary. Since each pointer must point in one of eight directions a code may be used in which the numbers 0 to 7 represent these directions. Such a code corresponds closely to the chain code, discussed by Freeman(39), which a

_ 38 _

number of workers have employed for boundary representation.

Philbrick(40), describes equipment, comprising a T.V. monitor and light pen linked directly to computer, used to investigate properties of the above transformation. He illustrates boundaries regenerated from skeletons which have been straightened, i.e. each branch has been approximated by a straight line.

A class of transformations which more closely approximate that of Blum are described by Montanari(41). Effectively the eight neighbouring points are considered to be at their true Euclidean distances. If more distant neighbours and their true distances are also considered specifically then 'height' values will more closely approximate the orthogonal distance to the nearest boundary point. Unfortunately, the computational effort required increases rapidly as more neighbours are taken into account.

In a later article Montanari(42) describes a computer program which derives the true medial axis transformation (M.A.T.) from the boundary analytically. The boundary is considered to be a polygon as in Fig. 3. It is necessary to calculate the height at which each side of the polygon will disappear. Refering to Fig 3(c), the perpendicular

- 39

distance d₂ in which the side of length L₂ disappears can be expressed as:

$$d_2 = \frac{L_2}{\cot \theta_2 - \cot \theta_3}$$

Clearly $L_2 = (x_2 - x_3)^2 + (y_2 - y_3)^2 = a_2^2 + c_2^2$

where $a_2 = x_2 - x_3$ and $b_2 = y_2 - y_3$.

The half angles may be evaluated from:-

$$\theta_2 = \frac{1}{2} \tan^{-1} \frac{(y_1 - y_2)(x_2 - x_3) - (y_2 - y_3)(x_1 - x_2)}{(x_1 - x_2)(x_2 - x_3) - (y_1 - y_2)(y_2 - y_3)}$$

$$= \frac{1}{2} \tan^{-1} \qquad \frac{a_1 b_2 - a_2 b_1}{a_1 a_2 - b_1 b_2}$$

since this expression may vary in value from zero to infinity an equivalent one employing the cosine function may be considered preferable:-

$$\theta_2 = \frac{1}{2} \cos^{-1} \qquad \frac{a_1 a_2 - b_2 b_1}{L_1 \cdot L_2}$$

Notice that d_2 approximates the mean radius of curvature in the region of L_2 . If d values are found for each polygon side and compared the minimum will correspond to the first skeletal branch. The new angle formed on the wavefront will be the sum of those on each side of the edge which vanishes and affects the d values of the two adjacent sides. Use may be made of the relation:-

 $Cot (a + b) = \frac{Cot a.Cot b - 1}{Cot a + Cot b}.$

The minimum remaining d value will form the next branch and the process is repeated. Eventually all the remaining d values (normally 3) are equal and the wavefront vanishes.

Let us consider the transformations of three other shapes shown in Fig. 4. The wavefront of a circle consists of concentric circles which disappear at their centre producing a single point skeleton. The skeleton of an ellipse is a straight line along the major axis starting at the centre of the smallest radius of curvature and vanishing at the centre of the circumscribed circle. A concave angle produces a wave form containing circular arcs between the straight edges, hence no discontinuity in slope, and therefore no skeletal lines until this wavefront impinges upon another travelling in a However, a skeletal line is different direction. formed on the outside of the angle as a perpendicular bisector.

_ 41 _



-41A-

Montanari(42) shows that circular arcs impinging upon straight lines produce skeletal lines which are parabolic arcs. Although analytical treatment of such lines is still fairly straightforward some of the attractive simplicity of convex polygonal skeletons is lost. The height at which each circular arc will impinge upon every other advancing wavefront must be calculated to ensure that skeletal branches are located in order of Computation of all the initial d values occurence. is therefore radically increased. Fortunately. work involved in updating this list of heights as sides disappear is not so drastically affected. However, complications occur when a circular arc impinges on another wavefront. The wavefronts are split into two sections and each subset must thereafter be considered separately.

Scanning Hardware.

Many of the articles referenced describe equipment used in the digitisation of images. The author has already completed a survey and comparison of scanning techniques (29) and this topic is not repeated here. A recent article by Stevens(30) describes a number of European pattern recognition projects.

DEVELOPMENT OF THEORY.

This section gives a detailed account of the theoretical development leading to the practical work of the thesis. Inevitably, the method of critical examination caused one to continually review techniques and in some cases reject them entirely. Some of the pitfalls encountered are discussed since the knowledge gained from them influenced the final choice of method.

One of the problems in reading deeply into a subject is that more and more of ones 'own' ideas become attributable to other writers. Where known, authors concurring or disagreeing with arguments put forward are referenced. Unfortunately, a number of workers have paid far too little attention to problems of numerical stability and computational efficiency. Such considerations have had a considerable influence on ones own choice of method and its implementation.

- 44 -

When a picture contains a number of distinct objects whose boundaries may be determined from some function of the grey levels these boundaries frequently contain all the required information. Prior to the commencement of this thesis the author had written a computer program to extract boundaries of enclosed regions from thresholded pictures(29). In practice, the pictures may be of fibres, animal or plant cells, molecules, powders, polymers, metal grains or even household objects. This thesis is concerned with investigating parameters which describe the shape of enclosed regions.

It is unfortunate that shape descriptors used by humans do not in general correspond to unique scientific measurements. Even a subjective judgement of relative size may differ from corresponding measurements.

The vocabulary of shape describing adjectives used in the English language is little short of enormous. These adjectives may themselves be classified some of the main groups are:-

a) Comparison with a geometrical shapee.g. triangular, elliptical,

prismoidal.

b) Comparison with a familiar objecte.g. pear-shaped, egg-shaped.

- 44A -

c) Subjective assessment of how the object would feel

e.g. smooth, spiky.
d) Subjective assessment of some physical property

e.g. fragile, solid.

Shapes are also described in terms of aesthetic qualities and even identified with human personality traits. In many cases such adjectives are closely related and their use varies considerably from one individual to another. In shape characterisation it is desirable to use parameters which are mutually independent.

The boundary of an enclosed region obtained from a digitised image may be considered as a list of (x,y) coordinates. Items in the list have an order, if for instance the boundary is followed in a clockwise direction, but no logical beginning or end.

45 -

Digitised Boundaries.

Before proceeding with any analysis using scan boundary data it is essential to study the nature of the available information. Assuming a rectangular grid is used Fig 1. shows the measured black/white transition points connected as boundary B. This may be considered as the most likely or 50% boundary when each measurement is regarded in isolation. It is immediately apparent that a boundary in closer correspondence with the original object might result by considering each point in relation to its neighbours and by making certain assumptions concerning the boundary statistics.

Strohl(31) has derived a model for such schemes and uses it to prove, using information theory, that an improvement is achieved provided that 78% or more of points, which may be considered as moved onto a finer grid, are correctly repositioned.

Boundary A in figure 1. is the outermost or 0% boundary, whilst C encloses the area which is known (100%) to lie within the object. This latter area is refered to by Sklansky(13) as the core. Some objects contain a complex core i.e. in more than one piece. A square of side twice the error tolerance has a core which is a single point. Boundaries A and C are the limits of an annular uncertainty region within which the true boundary of the original object must lie. No upper limit

_ 46 _



to boundary perimeter can be estimated but if a string were pulled taught within the annulus a minimum perimeter path would be traced. Sklansky proves that this minimum perimeter polygon (MPP) is unique for a given boundary B. The reverse correspondence is only true, however, provided the error limits are within those specified by Montanari(32). One of the primary difficulties encountered by workers on boundary analysis has been the existence of concavities. These may occur in the 50% boundary despite the original object being convex. Sklansky redefines convexity as possession of a convex MPP thereby removing the above anomaly, whilst enlarging the more easily managed class of boundary. Sklansky(33) describes a computer algorithm to find the MPP based on the taught string principle. It may be observed that the MPP deletes the inflections refered to by Zahn(26) in a more rigorous manner.

Reviewing the various ways in which our boundary information may be considered, in absolute terms we have a number of boundary coordinate measurements each one subject to quantisation error. Hence, one may assert that the true object boundary passes within a certain small distance of each point this distance being equal to or less than the error tolerance. The above remark infers that the boundary must pass through a circular area surrounding the

47

point although in practice this area may be rectangular, it also involves the implicit assumption that the local radius of curvature of the boundary is not significantly less than the linear error tolerance. If this assumption is false, e.g. the boundary zigzags wildly, then the quantised boundary will fall within the zig-zag region tut obviously its actual error will be considerably greater.

Having considered each boundary point by itself we may now attempt to group these measurements to obtain properties. Two methods of grouping are prominently featured in the literature survey. Firstly, all the points may be considered as an ordered list tracing the boundary path. Intuitively. these points form a complete description of the boundary shape and it should only be necessary to manipulate the data into a more easily handled structure. By applying information theory one may devise an efficient code to represent the boundary and attempt to use this representation for the comparison and classification of boundaries. Alternatively, using methods of mathematical approximation one may attempt to fit a suitable function to the boundary points. Ideally, this approximating function should also perform boundary smoothing hence reducing quantisation Since the boundary is a closed planar curve noise. it is not, in general, possible to find a single valued expression such as y = f(x) to represent the

- 48 -

Fortunately, it is possible to use entire boundary. a number of such functions to form a piecewise approximation i.e. each function approximating one piece of the boundary. Clearly, the more complex the piecewise fitting-function becomes the larger the 'piece' of boundary we may expect to satisfactorily approximate. In practice, we may consider two simple functions, namely, straight lines and cubic Since a detailed discussion of approxpolynomials. imation theory is outside the scope of this work we will consider these possibilities in outline only. If piecewise linear approximation is used the boundary produced will resemble the form of that implemented Since the MPP boundary is both by Sklansky(33). unique and has minimum total curvature for any given digitised boundary and error tolerance we may regard it as an optimum linear fit. Extending the fitting function to a cubic equation it would seem reasonable to impose upon this boundary the constraints of passing through the vertices of the MPP. Two additional degrees of freedom of the cubic function may be used either to increase the size of the boundary pieces or to provide continuity of slope, and higher derivatives, at the points where the pieces are joined. These points are called 'knots'. If both degrees of freedom are used for boundary derivative continuity the piecewise fit is by a class of functions known as cubic splines. The actual points where the pieces overlap may be either pre-determined or derived

- 49 -

from the constraints of the annular boundary uncertainty region.

In the particular task of shape analysis we are more interested in the data compression resulting from functional approximation than accurate boundary reconstruction. Sklansky's MPP effectively selects a reduced set of vertex points which summarise both boundary and error tolerance information. However. it does not select the vertices of a minimum sided polygon falling within the uncertainty region. This is a much more complex task and involves non-linear programming along the lines described by Montanari(32). One may consider each boundary point as contributing to a knowledge of the true boundary position by constraining the uncertainty region. Clearly some points contribute more to this knowledge than others. In particular points which are collinear with their neighbours will not alter the uncertainty region if they are omitted. The MPP vertices form a set sufficient for the polygon formed to always lie within the uncertainty region. A minimum sided polygon (MSP) conforming to this constraint would represent optimum data compression but the improvement would not normally justify the additional computation involved. Since the method of Sklansky is straightforward it appears to be a good compromise between degree of compression and computational efficiency.

The second method of grouping previously described is that of ordonnances. Effectively, each boundary point is given a number and all the lines

- 50 -

joining pairs of points are then considered and ordered in decreasing length sequence. Each line is named by the points which it joins. The ordered list of line names is then used as an object description. This description is by its very nature size and position invariant and may be considered to be orientation invariant if the line numbering is carried out in a pre-determined manner e.g. if the longest ordonnance is chosen so as to involve point number 1 and point number I such that I is the minimum value for the two possible remaining orientations and points are labeled in clockwise order around the boundary. Although useful for comparing distorted forms of the same object this method has disadvantages in shape description, not least the large number of ordonnances involved. However, it is interesting to study this set of all possible lines joining pairs of points.

Our analysis of individual points led us to the conclusion that some conveyed more information It seems likely that this statement also than others. holds for ordonnances. Since the error tolerance of each boundary point is equal, the relative error associated with the line joining two points will be inversely proportional to its length. The longest chord from this set will also be the largest value cf Feret's diameter in any direction. In any given direction there will be a longest chord associated with two parallel tangents at right angles to the direction on opposite sides of the boundary. The length of this chord will be equal to Feret's diameter

- 51 -

multiplied by the secant of the angle which the chord makes with the direction concerned. Boundary points involved in all such chords will be the vertices of the convex envelope boundary or convex hull. These vertices will be a subset of those associated with the MPP.

Having considered individual boundary points and the lines formed by joining pairs of them we may move on to consider measurements obtained from any subset of points. One such measurement is the area bounded by an ordered subset. The error associated with area measurement will be the ratio of annular uncertainty region area to 50% boundary area. The error bound on area measurements will be at least twice that of linear measurements of the same average diameter to the area involved but assuming errors are of Gaussian distribution the error variance will diminish by the square root of the number of points used in area calculation.

In the foregoing discussion two problems have continually recurred. Firstly, because of the nature of the boundary information resulting from digitisation, measurements made between points particularly those in close proximity are subject to large errors. Perimeter and local boundary slope are two important parameters falling within this category. Secondly, a number of methods for analysing boundary shape are inapplicable or become computationally impracticable when concavities occur. The practical work of this thesis has been

- 52 -

concentrated upon providing a sclution to these two problems.

The possibility of more closely approximating the original boundary has already been mentioned. However, since we are concerned with boundary shape description it is more straightforward to use descriptors which perform the same smoothing operations implicitly. Integral functions are clearly indicated.

The author had noted in previous work that area measurements were far less susceptible to errors than linear measurements. An obvious extension was to consider two dimensional moments of which area is the zeroeth order member.

Moments.

For the purpose of taking moments it is convenient to consider the object to be constructed of filaments or chords. Considering such a chord in the x direction and taking moments we have:-

xmin

 $\mathbf{x} = \mathbf{x} \mathbf{m} \mathbf{e} \mathbf{x}$

 $\mathbf{x} = \mathbf{x} \min$

xmax

 $x^{p} \cdot dx = \begin{bmatrix} p+1 \\ \frac{x}{p+1} \end{bmatrix}$

Mp =

Clearly any scaling factor s used to normalise the size of the object will have to affect the value of Mp by a factor $s^{(p+2)/2}$. The values of xmin and xmax are known to $\frac{+1}{2}$ a picture point unit hence the error

- 53 -

tolerances of the moments about a filament may be expressed as:- -

$$Mp = \frac{(xmax^{\frac{1}{2}})^{p+1} - (xmin^{\frac{1}{2}})^{p+1}}{p+1}$$

ignoring second and higher order errors :-

$$Mp = \frac{(xmax)^{p+1} - (xmin)^{p+1}}{p+1} \stackrel{\pm}{=} \frac{1}{2} (xmex - xmin)$$

where xmax and xmin are of opposite sign. Assuming that the

errors are normally distributed we may weight the moments of each filament by the inverse square of the relative error magnitude to minimise the variance of the moments calculated. This amounts to weighting longer chords more than short ones.

Another method of moment calculation is to consider each picture point within the object separately. For the case of zero error results should be identical but since only points near the boundary are subject to errors these may now be considered and weighted individually.

Points which have one or more 8-near neighbours of a different colour to themselves may be considered to be in the boundary region. It should be possible to assign to such points a most likely % black area. The improvements made to the exact values of moments calculated may be considered less important than the reduction of error bounds.

- 54 -

In particular, in the comparison of two objects, error values would enable one to determine whether on the basis of a number of moments the two objects were significantly different.

We have previously mentioned the dependence of points in the boundary list upon size, position and orientation. Various measurements have been aiscussed, e.g. ordonnances, which preclude this dependance. Low order moments may also be used to calculate a transformation to give the required Effectively we may transform any independance. boundary list so that it will have similar form to any object of the same shape and refer to such a process as boundary standardisation.

Boundary Standardisation.

Let us assume that Mpq is the moment in the pth power of x and the qth power of y. Clearly MOO = Area and MO1 and M10 are \overline{x} and \overline{y} the coordinates of the centroid. If we also rotate the coordinate axes so that Mll becomes zero and normalise the area to unity the resulting transformation is equivalent to:-

 $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{c} & \mathbf{s} \\ -\mathbf{s} & \mathbf{c} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x} & - & \mathbf{x} \\ \mathbf{y} & - & \mathbf{y} \end{bmatrix}$ $c = \sqrt{Area}$. Cos θ and $\theta = \frac{1}{2} \tan^{-1} \frac{\mu}{\mu}$ $s = \sqrt{Area}$. Sin Θ

where

 μ ll, μ 20 and μ 02 are central moments i.e. equivalent to Mll, M20 and M02 but substituting $x - \bar{x}$ for xand $y - \bar{y}$ for y.

The transformed object will have the following moment values: MOO = 1, MOI = MIO = 0, MII = 0. Within a close approximation, i.e. subject to quantisation errors, all objects of the same shape should result in identical transformations independent of initial size, position or orientation. The parameters Θ , Area and \bar{x} , \bar{y} (the centroid coordinates) may in many instances be useful. However, we are only concerned with shape and will therefore only consider transformed objects.

Moments MO2 and M20 may be regarded as the variance or spread of the object about the x and y axes and hence are two independent shape parameters. Higher moments may be taken but first let us consider the boundary of our normalised object.

Along the lines of Klinger(25) we may use a polar representation of the object with respect to a circle of unit area centred on the centroid origin, see Fig. 2(a). Although this boundary is not necessarily a single valued function for $r \pm f(\theta)$, it will be so for a larger class of objects than Sklansky's convex set. Obviously the areas above and below the axis of Fig. 2(b) are equal. Many of the object features are revealed through studying this boundary e.g. number of zero crossings, enclosed areas between zero crossings and the angles at which zero crossings occur.

_ 56


An interesting possibility is to consider each group of points between zero crossings as a subset and to make measurements upon each of these subsets as a separate entity. Rosenfeld(1) mentions the analysis of pictures using subsets but his main interest lay in the separation of objects within a picture rather than the segmentation of individual objects. If subsets are to be used for shape analysis a primary requirement will be the stability of subset formation when the object undergoes small distortions. One therefore must consider situations in which similar objects will produce different subsets. Unfortunately, as Fig.2(b) shows the angle at which the $r = f(\theta)$ line crosses the unit circle is normally quite small. It is therefore reasonable to suppose that the shape of subsets will vary considerably under small distortions as will the θ values at which zero crossings occur. This form of segmentation was therefore not considered

suitable for further analysis, but the general idea of segmenting the boundary into local subsets was thought to be good.

Some objects have a small internal sub-area within which the centre of a single valued polar transformation $r = f(\theta)$ may be situated although this area need not necessarily include the centroid. Objects possessing such an area are described as star-shaped e.g. by Rosenfeld(1).

- 57 -

An object whose boundary is convex has a single valued polar transform for all internal points.

It is apparent that the boundary would be easier to handle if the two independent variables could be considered as one independent and one dependent variable for all classes of closed boundary. Two such variables are distance along the boundary and slope. The slope is most conveniently expressed in terms of the angle to a principal axis. An interpolation routine is needed to enable the distance along the boundary and the local slope to be evaluated as accurately as possible. Unfortunately, as Hawkins(34) points out, both these variables will be sensitive to small distortions due to quantisation error and image perfections. Indeed, unless the first derivative of the boundary is continuous the line will be disjoint.

Despite the above drawbacks, Zahn(35) has recently published an article in which this transformation is approximated using Fourier descriptors. These are essentially Fourier coefficients of the line manipulated to give size, position and orientation invariance. Zahn states that low order (low frequency) coefficients are good shape descriptors and are resistant to change under small image distortions. He illustrates the reconstruction of a boundary, using low frequency descriptors only, showing a large boundary discontinuity. Nevertheless, the convenience of this transformation may well outweigh its shortcomings.

- 58 -

Concavities.

The problem of concavities has troubled numerous workers. Let us first consider a method of isolating individual concevities from the main boundary in order to study them more closely. Around any object it is possible to draw a line corresponding to a taughtly stretched string. This line is the convex hull boundary, and, if the object is convex, it will stay within a small distance of the main boundary, the difference being attributable to quantisation error. Since area measurements are less susceptible to error than linear measurements we may consider each concavity as the area between the true and convex hull boundaries. The number, size and shape of concavities are of great interest in the analysis of the shape of the parent object. Figure 5 shows five concavities. Notice, in particular, that shape 5(c) suggests the overlap of objects more strongly than the others. The factors involved in reaching this decision may be enumerated :-

1) Acute angled vertex.

- 2) Smooth boundary elsewhere.
- 3) Slope continually increasing except at discontinuity.

Unfortunately, all three properties depend upon local boundary measurements and therefore cannot be assessed accurately. On the other hand all three properties may be considered as features

. 59 .



of the shape formed by the boundary points bordering on the concavity. Shape parameters generated from these points should allow the same assessment to be made whilst being less susceptible to small distortions.

It can be seen from Fig. 5 that the shapes of concavities are in no less general a class than the shape of original objects. A logical method of analysis is therefore to consider each concavity as a new object. We have thus defined a recursive procedure for analysing the shape of any object, namely, consider each object to be a convex hull and a number of concavities, then treat each

The above analysis effectively segments an object into a number of convex components the algebraic sum of which constitute the original object. Figure 6 shows the segmentation and a tree illustrating the connection of convex components. This type of division is uniquely defined for any object. The size of each convex component will be accurate within easily calculated tolerances and most small distortions will have marginal effect. The connection tree is itself an important summary of the object shape.

60



Applying our previous criterion of stability under small distortions we see that although the angle between the convex hull boundary and the true boundary will normally be quite small, distortion of the true boundary will lead to a similar distortion of the convex hull. It follows that provided the distortion of the subset of boundary points on the convex hull is representative of the distortion to the boundary as a whole, comparison of the convex hull and concavities will effectively be compensated for these distortions. Hence a marked gain in immunity to distortions is achieved over the segmentation shown in Fig.2(a).

There remains the problem of situations in which similar objects may give rise to a different component set. This will occur whenever the local radius of curvature on the boundary of a subset is very large. A small distortion will form a new concavity and hence a new convex component. Fortunately, components so formed will be small in size and may thus be ignored by stipulating a threshold area below which any concavity will be included as part of the boundary of its parent component. An additional bonus accrues from this procedure. It will perform a similar action to the formation of the MPP. However. choice of an area threshold rather than one based on the error tolerance of individual boundary points allows a number of distortions to be ignored in addition to those arising from quantisation error. Specifically,

- 61 -

when a single boundary point has been inaccurately measured due to a fault in image preparation or scanning its influence upon the shape description of the object will be largely suppressed. Only one situation has been found in which the convex components of similar objects may be drastically different. When two concavities are in close proximity on an object boundary and the convex hull boundary spanning them has a very high radius of curvature then a small shift in position of the point or points on the portion of the convex hull between these two concavities may cause them to merge into one. Fortunately, this situation will only occur rarely and may easily be detected. A possible solution would be to provide an object description which included an 'either or' structure for this type of occurrence.

Every convex envelope may be considered as a 'host' subset to the boundary sub area which it contains. Suppose a parameter is calculated from all the points within the sub area and an identical type of measurement is then made using all the points in its convex envelope clearly the ratio of these measurements will be constant under a large number of possible distortions.

The above method of segmentation was considered sufficiently stable to be useful in shape description. The convex components formed could be individually

- 62 -

analysed by a number of techniques thereby making its implementation independent of the component features later extracted. Methods for the computerised decomposition of an ordered boundary point list into convex components were therefore examined.

An initial step was the selection of the convex hull vertices from the boundary point list. A simple test to determine these points could be based upon the fact that any point which fell inside a straight line drawn between two other points could be eliminated. Unfortunately, the definition of what constituted 'inside' was found to present problems. Also even assuming this could be overcome the amount of work involved in using this test alone would be Recalling that the convex hull vertices considerable. were the same points which influenced the value of Feret's diameter in any direction, the following model Suppose the object was cut from a was considered. sheet of cardboard and then fixed to another sheet by a drawing pin. If the object was then rotated about the point at which it was fixed the point on the boundary with maximum projection in a given direction would be a convex hull vertex. This model allowed a different test for use in rejecting points. Consider an internal point x, y and two boundary points x_1y_1 and x_2y_2 . Let the angle of the line from $x_0 y_0$ to $x_1 y_1$ to a given axis be θ_1 and the angle of the line from $x_0 y_0$ to $x_2 y_2$ be θ_2 so that the angle

- 63 -

between the two lines is $\theta_2 - \theta_1$. Also, let the lengths of these two lines be H_1 and H_2 . H_2 may be rejected as a convex hull vertex if:-

$$H_1 \cos(\theta_2 - \theta_1) - H_2 \ge 0$$

now $\cos(\theta_2 - \theta_1) = \cos \theta_1 \cos \theta_2 + \sin \theta_1 \sin \theta_2$

and
$$\sin \theta_1 = \frac{x_0 - x_1}{H_1}$$
 and $\cos \theta_1 = \frac{y_0 - y_1}{H_1}$

with similar expressions for $\sin \theta_2$ and $\cos \theta_2$. Substituting in the above inequality we obtain:-

$$H_{1}\left[\frac{(x_{0}-x_{1})(x_{0}-x_{2}) + (y_{0}-y_{1})(y_{0}-y_{2})}{H_{1}H_{2}} - H_{2} > 0\right]$$

multiplying by H2:-

$$(x_0-x_1)(x_0-x_2)+(y_0-y_1)(y_0-y_2) - H_2^2 \ge 0$$

but $H_2^2 = (x_0-x_2)^2 - (y_0-y_2)^2$

substituting and rearranging:-

$$(x_2-x_1)(x_0-x_2)+(y_2-y_1)(y_0-y_2) \ge 0$$

The above condition was found to be necessary but not sufficient for the rejection of the point x_2y_2 . In order to eliminate this point it was found that a further point x_3y_3 on the other side of x_2y_2 from x_3y_1 would have to satisfy the condition.

$$(x_2-x_3)(x_0-x_2)+(y_2-y_3)(y_0-y_2) \ge 0$$

These two conditions are of about the same complexity as the test originally mentioned but appeared to solve the difficulty of whether a point

- 64 -

is inside or outside since this is effectively measured as the distance from x_0y_0 . A computer routine based on this test was implemented and appeared both simple and successful until a group of points were tried which represented a very narrow and highly concave object. For this object a boundary point occurred which subtended an angle more than 90° different to that of both its neighbours. This point was accepted by the routine as a convex hull vertex even though it lay at the 'bottom' of a deep concavity.

A reappraisal of the situation was necessary. It soon became apparent that difficulties arose mainly through attempting to select the convex points using a single test. A more detailed study was made of the nature of the convex hull. Figure 7 shows some of the properties of this hull, in particular it may be seen that the four points at the extremities of the object in x and y directions will always be convex The points at minimum and maximum x coordpoints. inates may be considered to divide the boundary into an upper half and a lower half. No more than one point at each x coordinate along the upper half of the boundary can be on the convex hull, hence the shaded points in Fig. 7 may be eliminated as convex points. A similar test removes points from the lower half of It is also known that the convex hull the boundary. will monotonically rise or fall between each of the neighbouring four extremal points. A test using this fact may also eliminate a number of points. Final

- 65 -



selection may use the property that every convex point will lie outside a line drawn between any other two boundary points. In this context, 'outside', may be tested as above for the upper boundary half and below for the lower boundary.

The area of each concavity may now be calculated as previously defined and those above a threshold size noted as new objects. Points within concavities smaller than the threshold may be considered to lie on the convex hull.

Simple moments may be calculated from the points on each convex hull by any of the methods outlined previously.

Hu(43), showed that simple moments could be converted into parameters invariant with respect to size, position and orientation, thereby obviating the need to transform the boundary point coordinates. He mentions two classes of such invariant moments; absolute invariants, which are totally independent of the above variables and relative invariants in which the values of the variables are standardized. Since the former are computationally difficult to derive, invariants about the principal axes have been used.

Machine Generation of Invariant Moments.

Initially objects are known by an ordered list of boundary points. By definition

Mpq =
$$\int_{-\infty}^{+\infty} x^p \cdot \int_{-\infty}^{+\infty} y^q \, dy \cdot dx$$

gives moments for the object in arbitrary orders of x and y. In digital terms the moment of an object may be considered as the sum of the moments of the digitised squares of which the object is composed, i.e. if \sum_{s} is used for the sum of all the squares (picture points) within the object then

Mpq =
$$\sum_{g} \int_{x=x-\frac{1}{2}}^{x=x+\frac{1}{2}} x^{p} \cdot \int_{y=y-\frac{1}{2}}^{y=y+\frac{1}{2}} y^{q} dy \cdot dx$$

$$= \sum_{y} \frac{(x+\frac{1}{2})^{p+1} - (x-\frac{1}{2})^{p+1}}{p+1} \cdot \frac{(y+\frac{1}{2})^{q+1} - (y-\frac{1}{2})^{q+1}}{q+1}$$

(E4)

(E5)

where it is assumed that each square is of unit side centred at x,y. For squares with equal x values, and y values which differ by unity, summation of their moments results in the cancellation of all but the extremal terms in y. Hence the moment of a chord may be expressed as:-

$$\frac{(x+\frac{1}{2})^{p+1} - (x-\frac{1}{2})^{p+1}}{p+1} \cdot \frac{ymax^{q+1} - ymin^{q+1}}{q+1}$$

so that Mpq =
$$\sum_{c} (\frac{x+\frac{1}{2}}{p+1})^{p+1} - (x-\frac{1}{2})^{p+1}$$
, $\frac{ymax^{q+1}-ymin}{q+1}^{q+1}$

where \sum_{c} indicates the summation of all the chords. It is therefore a simple matter to calculate moments from the boundary points since it is known whether y values are upper (ymax) or lower (ymin).

Recalling the binomial theorem we know that :-

$$(x+y)^{p} = \sum_{n=0}^{n=p} \frac{p!}{(p-n)!n!} \cdot x^{p-n} \cdot y^{n}$$

Scaling the moments for size invariance is a straightforward procedure since MOO is the area. To make the object of unit area it is necessary to divide each linear distance by \sqrt{MOO} , from (E5) we get:-Mpq (normalised) = Mpq/MOD^{(p+q+2)/2} (E6)

Positional invariance may be obtained by shifting the origin of the coordinate axis to the object centroid \bar{x}, \bar{y} where $\bar{x} = MlO/MOO$ and $\bar{y} = MOl/MOO$ Substituting x=x- \bar{x} and y=y- \bar{y} into (E5) we obtain $\mu pq = \sum_{c} (\frac{x+\frac{1}{2}-\bar{x}})^{p+1} - (x-\frac{1}{2}-\bar{x})^{p+1} \cdot \frac{(ymax-\bar{y})^{q+1} - (ymin-\bar{y})^{q+1}}{q+1}$

using the binomial theorem

$$\mu_{rq} = \begin{bmatrix} \sum_{n=0}^{n=p+1} \frac{(p+1)!}{(p+1-n)!n!} (x+\frac{1}{2})^{p+1-n} (-\overline{x})^n - \sum_{n=0}^{n=p+1} \frac{(p+1)!}{(p+1-n)!n!} (x-\frac{1}{2})^{p+1-n} (-\overline{x})^n \\ p+1 \\ \\ \sum_{m=0}^{m=q+1} \frac{(q+1)!}{(q+1-m)!m!} y^{max} q^{q+1-m} (-\overline{y})^m - \sum_{m=0}^{m=q+1} \frac{(q+1)!}{(q+1-m)!m!} y^{min} q^{q+1-m} - \overline{y}^{max} q^{q+1} \\ \\ = 0 \\ q+1 \end{bmatrix}$$

_ 68 .

rearranging:-

 $\mu pq = p!q! \sum_{n=0}^{n=p} \frac{(-\bar{x})^n}{n!} \cdot \sum_{m=0}^{m=q} \frac{(-\bar{y})^m}{m!} \cdot \frac{Mp-n q-m}{(p-n)! (q-m)!}$ (E7)

Substituting values of p and q into (E7) and making the additional substitutions of \bar{x} =MlO/MOO, \bar{y} =MOl/MOO we may obtain the relations given by equations 11 of Hu(43).

If in equation (E4) our squares are of size 24x by 24y we obtain

 $Mpq = \sum_{s} \frac{(x + \Delta x)^{p+1} - (x - \Delta x)^{p+1}}{p+1} \cdot \frac{(y + \Delta y)^{q+1} - (y - \Delta y)^{q+1}}{q+1}$ which for arbitrarily small Δx and $\Delta y \rightleftharpoons \sum_{s} x^{p} \cdot y^{q}$

If the coordinate axes are rotated by the angle θ , x and y are transformed such that $x=x\cos\theta+y\sin\theta$ y=ycos $\theta-x\sin\theta$.

substituting in the above expression:-

$$Mpq(rotated) = \sum_{s} (xcos\theta+ysin\theta)^{p}(ycos\theta-xsin\theta)^{q}$$
$$= \sum_{s} \sum_{n=0}^{n=p} \frac{p!}{(p-n)!n!} (xcos\theta)^{p-n}(ysin\theta)^{n}$$
$$\cdot \sum_{m=0}^{m=q} (-1)^{m} \cdot \frac{q!}{(q-m)!m!} (ycos\theta)^{q-m}(xsin\theta)^{m}$$
$$= \sum_{n=0}^{n=p} \frac{p!}{(p-n)!n!} \sum_{m=0}^{m=q} \frac{q!}{(q-m)!m!} (-1)^{m}$$
$$\cdot \cos\theta^{p+q-n-m} \sin\theta^{n+m}$$

$$\sum_{s} x^{p-n-m} y^{q-m-n}$$

Substituting and rearranging.

Mpq (rotated)=p!q!
$$\sum_{n=0}^{n=p} (\frac{\sin\theta}{n!})^n (\frac{\cos\theta}{p-n})^{p-n} \cdot \sum_{m=0}^{m=q} \frac{(-\sin\theta)^m \cdot (\cos\theta)^{q-m}}{m!} \cdot \frac{(\cos\theta)^{q-m}}{(q-m)!}$$

.Mp-n+m q-m+n (E8)

Combining equations (E6), (E7) and (E8) invariance from size, position and orientation may be obtained. It will be seen that the transformed moments are functions of all simple moments of equal or lower order.

By choosing the angle θ so that $\mu ll^{(rotated)} = 0$ i.e. $\theta = \frac{1}{2} \tan^{-1} \frac{2 \lambda l l}{\mu 20 - \mu 02}$

 θ is determined to be one of four angles at 90° , for all objects which are not symmetrical such that μ ll = 0 and μ 20 = μ 02. To determine θ uniquely a further constraint may be made. A simple choice is to make both μ 03 and μ 30 \geq 0.

Segmentation into internal convex sub-areas.

It is also possible to divide an enclosed region into convex sub-areas by internal segmentation. In order to uniquely specify such a procedure a number of rules are necessary. For instance segmentation shall:-

- 1) Avoid creating new concavities.
- 2) Cause a minimum increase in the number of sides.
- 3) Avoid segmenting convex subsets.
- 4) Avoid producing acute angles.
- 5) Minimise artificial segmenting lengths.
- 6) Exploit symmetry.

As figure 8 shows it is essential to specify the order in which these rules are applied and in some cases to alter this order or specify new rules when a particular rule cannot be adhered to. The figure contains two convex subsets but a new concavity is produced when the two concave points are joined. If rule 1 is given priority and the lower concavity is considered first, three possible segmenting lines occur, two of which are continuations of the lines



adjacent to the concavity and the third perpendicular to the side opposite. Comparing the first two of these options that are shown in Fig. 8 (a) is more attractive since it avoids creating an acute angle. Although shorter than the other two, the perpendicular line creates two new sides (and angles) rather than one as shown in 8 (d). If the shorter collinear line is chosen (as in 8 (a)) then the vertex produced is collinear with one line of the upper concavity hence only one new side is produced by the segmentation Notice, however, that two acute angles are shown. produced and that although a coincidence of the figure is exploited the segmentation itself does not appear natural, the shorter collinear extension shown in 8 (b) appears preferable even though the total number of sides is one greater.

In Fig. 8 (c) rule(1) is contravened, in that either of the segmenting lines in isolation produce a new concave angle. The line joining two concavities appears to be a natural segmenting line and it is interesting to note the effect of concave points upon the M.A.T.

Fig. 9 shows that whilst some concavities produce an internal starting point for the skeleton which then proceeds in opposite directions others merely create a curvilinear extension to existing skeletal branches. There is a close correspondence between this distinction and that of whether



concavities should be considered as evidence of overlapping objects.

Using the line joining concavities in Figs. 8 (d) and 8 (e) we see two more possible segmentations. Since a number of other segmenting lines could have been drawn it is clear that any method for internal segmentation will not be straightforward and is likely to necessitate a large number of rules and exception conditions to avoid ambiguities.

Summary of Findings ..

Let us therefore summarise our findings concerning computer extraction of shape parameters.

We have a list of boundary points subject to known quantisation errors. It is possible to calculate areas contained by groups of these points fairly accurately since errors tend to cancel. However, estimates of local boundary length and slope are highly sensitive to errors and parameters calculated using them, e.g. perimeter, are subject to cumulative error.

Low order moments will have low relative errors and weighting factors may extend their usefulness. Feature extraction may be used in characterisation but great care must be taken to ensure that the features chosen are invulnerable to small distortions. Features whose ratio of area to perimeter is high, i.e. approximately circular,

appear more stable than others since errors occur at the edges. If sub-areas are to be considered some method of segmentation is necessary. The moments and connecting structure of segmented components are thought to be most useful in shape characterisation.

Picture language theorists consider such components as primitives or symbols from which a picture language is constructed. Our primitives are closed planar figures and in general convex polygons. To uniquely specify a general polygom 2n parameters are needed where n is the number of Economies of this number are possible where sides. some parametric relations are assumed e.g. a square requires only four parameters, size, centre coordinates and orientation. Primitives such as the circle and ellipse are infinite sided polygons. with special relations between sides. The medial axis transform may be considered as the characterisation of the object boundary by a number of circles. Each skeletal point is a circle centre, its 'height' being the radius. The area to perimeter ratio of a circle is of course a minimum. Since circles are completely characterised by size and centre position the skeleton gives complete information for the infinite number of circles involved. Since the circle is itself a convex polygon it is not suprising that skeletons of concave curves are This insight could enable the boundary more complex.

74

to be characterised by a smaller number of primitives (circles) and hence a more compact representation. Other primitives might also be used to construct skeletons but this would necessarily require the standardisation of orientation prior to skeleton construction and/or an increase in skeleton aimensionality.

Ideally, it should be possible to produce a list of feature/shape parameters for any given object, each item in the list being independent of the others. Initial list members would indicate general shape categories whilst later members would deal with progressively finer detail. The entire list would completely characterise the object. Similarity of two objects could then be judged by the number of similar list items before a significantly dissimilar one. In practice a number of later members of the list would be discarded as inessential for the required task.

There is some evidence to suggest that the human eye-brain combination functions in a similar manner, the number of list items involved varying according to object complexity and frequency of encounter. It is significant that man finds no difficulty in recognising other individuals, particularly of his own genus, but will not normally be able to distinguish individual sheep by facial features. A shepherd will often have this ability.

_ 75

The inference is that remembered human facial feature lists are usually longer than those for sheep. A more complete discussion of this topic may be found in Walthen-Dunn(44) and its associated references.

COMPUTER IMPLEMENTATION.

Although work on the two problems discussed in Development of Theory has been carried out expressly for this thesis it is important to stress that neither technique could have been used without the background experience and, in particular, the software written during two years of previous research.

Details of the edge following routine have already been mentioned (29) and only the output from this procedure will be described here. Vectors of boundary points in adjacent positions and with consecutive x coordinates are held in a data structure corresponding to a plex, as described by Ross(45). Effectively the list of y coordinates is headed by a number of parameters describing the vector, its interelation with others, and pointers enabling access to other vectors, in which boundary points are continued in either direction. It is therefore a simple matter to access all the boundary points of an object given a single key entry value. Edge following is performed by the subroutine XYNSTACK which appears together with the other routines. mentioned in Appendix D.

Plex manipulation is handled by a series of short subroutines which greatly simplify other programming. To avoid excessive subscripting arithmetic these have been written in PLAN, the

- 77 -

ICL 1900 series assembler language. Other useful software includes a subroutine enabling array sizes to be specified as data(46) and a special error diagnostic/recovery routine.

When a complete enclosed region has been found the key value is passed from XYNSTACK to the subroutine PARTICLE which determines whether this region is in fact an outer boundary or a hole. The key entries for holes are stored so that when an object is found a search for holes which it contains may be made.

When an outer boundary is detected the key value is passed to the subroutine PARPAR which supervises evaluation of particle parameters. The version of this routine given in appendix D first calculates a number of parameters details of which are given in (47). Following an optional call to PARTPLOT, which plots the particle boundary on the graphplotter, control is then passed to TARTMOM which supervises the procedures written for this thesis.

In the author's experience careful planning of program structure results in a more compact end product and considerably assists debugging and subsequent modification. One's first task was therefore to divide the overall problem into smaller units, to be written as FORTRAN subroutines. Since convex decomposition was conceived as a recursive process a supervisory routine was necessary to organise the order in which components would be analysed.

- 78 -

The input to this routine (PARTMOM) comprises the key value enabling access to the boundary point list of an enclosed region and the threshold area value (AMIN) of the smallest significant component. Currently this threshold is set as a constant within the program but could easily be read as data or calculated as a percentage of total object area. FARTMOM first sets up a number of constants including a look up table for factorials. It then sums the total number of points in the boundary list (NTOT). The subroutine MOMSET is next called. This routine evaluates simple moments for the entire enclosed Effectively MOMSET calculates the x and y region. coordinates for every boundary point. If the point is on the upper boundary half then the moment values calculated for each point will be polynomials of x and y values for which the coefficients of y terms will be positive. If the point is on the lower boundary this coefficient must be made negative. Moment contingency values for each point are calculated within the routine SMOMS. Two x values are supplied to this routine XA and XC which are equal to the x coordinate of the current point ± 불. If XA > XC upper boundary values are calculated otherwise a sign reversal occurs in the arithmetic within SMOMS and lower boundary values result. SMOMS adds the moment contingency values for each point into the array MOM in which the simple moments for the entire boundary

- 79 -

are consequently summed. PARTMOM next calls MOMINV which converts simple moments held in the array MOM into invariant moments about the principle axes. Although the equations (E5), (E6) and (E7) are fairly simple to evaluate considerable savings in computation can be achieved by careful choice in the order of evaluation. It had already been decided to store the simple moments within MOM as a triangular array. In practice this structure must be held in a single dimensioned vector. Both space and time in executing subscripting arithmetic are saved using this storage form as opposed to occupying part of a two dimensional rectangular array. Unfortunately, the mapping of the moments of various orders into the appropriate location within the vector must be specified as part of the program. Area and the centroid coordinates \bar{x} and \bar{y} may easily be calculated from the simple moments. Two vectors of factors are then evaluated :-

 $XFAC_n = \frac{(-\overline{x})^n}{n!}$ and $YFAC_n = \frac{(-\overline{y})^n}{n!}$ for n=0,1,2,...R

where R is the highest order of moments calculated. At the same time size normalisation is combined with the scaling factor required in position invariance. The equation used is:-

$$Mpq = \frac{Mpq}{p!q! MOO(p-q-2)/2}$$

- 80 -

The position invariance equation (E6) involves moments of all lower orders in the calculation of each invariant moment. In order to generate these moments in situ i.e. for the results to remain in the same vector (MOM) it is necessary to compute highest order moments first. The items within the vector are therefore made invariant in reverse sequence. The equation is:-

 $Mpq = p!q! \sum_{n=0}^{n=p} XFAC_n \cdot \sum_{m=0}^{m-q} YFAC_m \cdot Mp-n q-m$

It is now necessary to determine the orientation angle Θ or more specifically its sine and cosine. The relation:

 $C2 = 0.5+0.5/\sqrt{2.Mll/(M20-M02)^2}$ is used where C2 will be either $\cos^2\theta$ or $\sin^2\theta$. The ambiguity is resolved by evaluating the expression:-

(2.02-1)Mll-(M2O-MO2).Sin θ .Cos θ which should approximate zero when the correct choice is made. It is now necessary to determine the quadrant in which M3O and MO3 are both positive. This is achieved by effectively subjecting the values of sin θ and cos θ to a 90° rotation and evaluating MO3 and M3O using equation (E7) until the required values are found. The final value θ is then determined. It is not possible to evaluate equation (E7) with values

- 81 -

within MOM remaining in situ unless the new invariant values of moments for each order are stored temporarily. The array YFAC is used for this purpose. Two new sets of factors are also evaluated:

SFAC_n = $\left(\frac{-\sin\theta}{n!}\right)^n$ and CFAC_n - $\left(\frac{\cos\theta}{n!}\right)^m$ for n=0,1,2,...R.

Equation (E7) now becomes:

$$Mpq = p:q: \sum_{n=0}^{n=p} (-1)^n SFAC_n CFAC_{p-n} \cdot \sum_{m=0}^{m=q} SFAC_m CFAC_{q-m}$$

• Mp-n+m q-m+n

At the completion of invariance calculations for each moment order the new values are returned to the appropriate positions in the array MOM. Since the values of MOO, MIO, MOI and MII are known the corresponding locations in MOM are overwritten by area, \bar{x} , \bar{y} and θ respectively.

On returning to PARTMOM a number of factors for use in component normalisation are set up. The precise method chosen for this normalisation is to relativise the size of all components to that of the original object. The same linear scaling factor is used to calculate the centroids of components relative to that of the original object. Values of x and y are measured along the principal axes of this object.

- 82 -

Orientation angles for components are measured relative to these axes. In this way all moment values of components may be considered as shape factors. The area, centroid coordinates and orientation of the original object are preserved as measurements relative to the original digitisation grid. These measurements and the other moments calculated are printed on the line printer using subroutine MOMOUT which also labels each component.

In order to establish which component of an object is associated with a given set of moments a component labeling scheme has been devised. The object as a whole and its convex hull are referenced as components 0 and 1 respectively. Convex sets associated with the concavities of objects are labeled as components 1.1, 1.2, etc. in clockwise order around the boundary. The decimal labeling may be continued to any depth thus component 1.2.3. is the convex set associated with the third concavity around the boundary of the component 1.2.

Subroutine MOMCON was written to handle completely the calculation of simple moments for a single convex component from a clockwise list of contiguous boundary points. Figure 6(b) shows that with each convex component may be associated a level or 'depth'. The original object and it's convex hull are at level 1 whilst their concavities are at level 2 etc. An important parameter which must be supplied as data to

- 83 -

MOMCON has a value which varies according to whether the next component to be found is of odd or even level. It enables MOMCON to determine whether upper or lower boundary points should be sought along a section of boundary comprising upper points of the original object boundary. The list of boundary points from which a convex component is to be extracted is described by three date items. The list vector or 'bead' within which the first boundary point coordinate is stored. the position of this coordinate within the bead and the total number of points in this boundary subset. These data items are held in corresponding elements of the three arrays NS, XS and YS respectively. After extracting this information from the arrays and dtermining whether the first point was an upper or lower point on the original object, MOMCON calls the subroutine CONP.

CONP produces an ordered list of convex points from the list supplied. As mentioned in 'Development of Theory' this task proved more complex than first imagined and even when the methodology described was finally conceived its implementation was far from trivial. Figure 10 shows the boundary of a concavity with three segments marked AB, CD and EF. For ease of manipulation CONP unpacks the list of boundary points into the three arrays previously mentioned which are used in this instance to store the number of each point in the list and its x and y coordinates respectively.

-84-

At the same time the points corresponding to B and D are located. It should be noted that segments AB and EF may not necessarily exist i.e. point C may be the first in the list and/or point D may be the last. In the former case points B and C may be considered to be co-incident. Assuming that the subset supplied is of the form shown in Figure 10 a test is first made to confirm that segment AB exists. The boundary list is then searched up to point B and for each x coordinate that point with the largest y coordinate is selected as a prospective convex point(T1). Convex points are now extracted from those selected using the straight line test previously described(T2). Since in each case non-convex points are eliminated from the list up to point B the resulting convex vertices may be stored in the same locations that these boundary points Effectively, the complete boundary formerly occupied. point list is contracted to a list of convex points only. Tests T1 and T2 are now applied sequentially to the boundary points after point B and up to and including point D. In each case the contracted list is stored in consecutive locations to those occupied by the previously dtermined convex points. A test is now made to establish if D is the last boundary point and if not any remaining points i.e. segment EF, are subjected to the same tests and contraction. The final output is therefore an ordered list of convex points stored in three arrays specifying x and y coordinates and

- 85 -


number of this point in the original boundary point list. Fests Tl and T2 will vary according to whether an upper or lower boundary is sought. In many instances it is required to move corresponding elements of the three arrays mentioned and a small utility routine (MOVEP) has been written for this purpose.

MOMCON now zeroises the moments array (MOM) and calculates simple moments for points spanning segment AF of Fig. 10. This task is performed first since it is known that no concavities may occur in the segment and the data required is readily available. To calculate moments for the rest of the component subroutine MCONSET is called.

As previously mentioned any concavities of area less than AMIN are ignored i.e. boundary points within these concavities are assumed to be on the convex If this assumption were not made areas, and hull. other moments. would be overestimated. MCONSET therefore computes in turn the area of each concavity. The 🔅 procedure is very simple since in all cases the convex hullwill follow a straight line between neighbouring convex points. The y coordinate corresponding to each point on the boundary point on the boundary which this line spans is obtained and subtracted from the coordinate of the point itself. Summing these differences gives the required area. This area is compared with AMIN and moments taken either for points along the convex hull line or for those on the true boundary. In the former case details of the boundary points within the concavity, i.e. their number and

- 86 -

the boad and location within the bead of the first point, are stored in the same manner as the data for the whole component. Once again the same locations within the three arrays may be used since the list of concavities will be shorter than those of convex points and will not be entered until the convex points overwritten are no longer needed. The number of concave subsets so listed is also returned to PARTMOM via MOMCON.

The simple moments returned by MOMCON are treated in a similar way to those output by MOMSET? i.e. they are made invariant by MOMINV and printed Before printing, however, the size, centroid by MOMOUT. coordinates and orientation are relativised with respect to the cooresponding values of the original object. PARTMOM now tests the number of new subsets found. If this value is non-zero then it is entered in the next available location of the array SET. This array holds the number of known components which have yet to be analysed at each level of the connection tree. The next component processed will be the last located as a subset of the previous component. This component is the last listed within the three arrays NS, XS and YS which are used in the manner of a push down stack to hold details of outstanding components. When a component is found to have no concavities PARTMOM subtracts 1 from the value of SET at the current level and test if any outstanding components remain at this level. If not the level indicator (JR) is diminished by 1 and the next level up tested in a similar manner.

- 87 -

In this way components are systematically analysed with minimal storage requirement. The programme is also simplified. When no more components remain control returns to XYNSTACK which continues to process scan data until another enclosed region is completely located.

The edge following routine itself is designed to handle one complete picture but may be called repeatedly allowing any number of pictures to be processed in a single computer run. All input of pictorial information is effected via the input routines PREAD8 and PREAD5 thereby enabling any peripheral to be used for this purpose without change to other routines.

RESULTS AND DISCUSSION.

Figure 11 shows a graphplotter representation of the boundaries of four objects (lunar dust particles). These were obtained from a raster scan which was processed by the edge following routine (XYNSTACK), adjacent boundary points were then joined by straight lines on a Calcomp incremental plotter. The four objects may all be loosely described as compact but each of them contains two or more concavities which should be considered significant for the purpose of shape analysis. The computer output shows that even ignoring size, position and orientation information the objects may very easily be distinguished. This example is, of course, fairly trivial and was used mainly to debug the computer One interesting feature, however, is routines. the concertina undulations of the boundaries in These were caused by backlash during Fig. 11. scanning and may thus be considered as noise. It is apparent that any parameters based upon local boundary properties, e.g. perimeter, would be grossly affected by this fault. Using a convex component moment analysis it is merely necessary that the area threshold AMIN exceeds the size of the small concavities produced for only marginal differences to occur in the resulting moments. Such a choice of threshold will not

- 89 -



- 89A -

prevent the location of 'genuine' concavities provided they are larger than AMIN. The optimum choice of AMIN in this example is not critical but could feasibly become important in the event of more gross distortions. Several strategies have been considered for the automated selection of the threshold. A simple method would be to consider the distribution of concavities versus threshold to be bimodal and to choose AMIN as the minimum frequency between modes.

Object number 5 has a particularly complex shape and proved to be ideal for finding minor errors in program logic which did not affect the processing of the other particles shown. It is interesting to note that of 540 boundary points listed for this particle by XYNSTACK only 25 were convex vertices. 318 of those remaining were found to be within significant concavities, leaving 198 within concavities smaller than AMIN. For this scan, AMIN was set to 0.5% of the area of each particle.

Referring to Table 1, although objects 3 and 4 have the same number of convex components the connection tree of each of the four particles is different. This structure could therefore be used to distinguish them. If only component 0. were calculated the second order moments M20 and M02 would also allow distinction.

Table 1. Analysis of Lunar Particles.

NUMBER OF OBJECT IN SCAN 1

AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORD Y X X Y EQ.DIAMETER -ARITY X v 132.6 151.6 0.874 134.0 129.0 476.3 13804 130 142 136 136 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (0.1380E 05) (263.6)(68.45)0.90276-01 0.7235E-01 (0.7388) 0.1349E-02 -0.2521E-02 -0.1177E-02 0.2925E-02 0.1634E-01 0.1049E-01 -0.4990E-03 0.4579E-02 0.6715E-03 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.022 -0.9266E-03 0.6184E-02 0.7117E-01 0,90558-01 0.3328E-01 0.2269E-(2 0.8928E-03 -0.1875E-02 -0.9040E-03 0.16316-01 0.1011E-01 -0.3822E-03 0.4498E-02 0.5296E=03 MOMENTS OF INCREASING COMPONENT 1.2. ORDER DOWNWARD AND INCREASING POWER OF Y ACRUSS 0.1029E-01 0.1709 -0.5116 0.5468 0.16266-01 1.256 0.5340E-01 0.3073E-01 -0.1072E-02 -0.4038E-03 0.6883 0.6652E-03 -0.1082E-03 0.6298E-02 0.2817E-02 MOMENTS OF INCREASING COMPONENT 1.1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.1213E-01 0.3762 0.3556 1.169 2.634 0.7760E-02 0.3424 0.2819E-03 -0.2197E-02 -0.4227E-01 0.4436E-02 3,076 -0.1922E-02 -0.5993E-03 0.6582E-02

- 90A -

NUMBER OF OBJECT IN SCAN 3

AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FEPET DIAM MAX.CHORD EQ_DIAMETER HARITY X X Y Y X 157.3 222.6 0.684 136.0 171.4 699.2 18224 143 180 136 173 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (0.1822E 05) (201.5)(111.7)0.6159E-01 (1.377)0.1077 -0.4366E-03 -0.2881E-02 0.3029E-03 0.3955E-02 0.7534E-02 -0.6640E-04 0,5103E-02 0,3026E-03 0.22556-01 COMPONENT 1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.055 -0.1391E-01 -0.1200E-01 0.6359E-01 0.6981E-01 0.1028 0.3933E-04 -0.3184E-02 -0.4622E-04 0.4806E = 020.8042E-02 -0.5117E-04 0_4716E-02 0.1049E-03 0.2100E-01 COMPONENT 1.4. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.8889E-02 -0.5829 0.1096 0.9718E-02 1.406 1,421 -0.4158E-03 -0.4879E-02 -0.2104E-01 0.5319 0.2551E-03 0.3367E-03 0.1495E-01 -0.5354E-02 3,426 MOMENTS OF INCREASING COMPONENT 1.3. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.1418E-01 -0.5087 -0.3293 0.1159E-01 2.079 0.6355 -0.6426E-04 -0.2055E-02 0.71485-02 0.7289E-01 0_2516E-03 -0_2121E-04 0_7423E-02 0_1215E-02 0_7604 COMPONENT 1.3.1. MOMENTS OF INCREASING ORDEP DOWNUARD AND INCREASING POWER OF Y ACROSS 0.5048E-02 -0.3336 -0.49132.085 0.7375E-02 1.086 0.1600E-03 -0.1402E-02 -0.1697E-01 0.1879 0,1248E-03 0,7380E-04 0,6813E-02 -0,1486E-01 2.392 COMPONENT 1.2. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.2933E-01 -0.1613 -0.4761 0.3163E-01 -3.080 0.2761 -0.2277E-02 -0.2025E-02 0.2570E-01 0.6831E-02 0.2120E-02 -0.7515E-04 0.7860E-02 -0.2107E-02 0.1786 MOMENTS OF INCREASING COMPONENT 1.1. ORDER DOWNWARD AND INCREASING POWEP OF Y ACROSS 0.7161E-02 -0.3478E-02 0.4640 0.15408-01 0,6418 3.022 0.4798E-03 -0.4876E-02 -0.2691E-01 0.2971 0.1825E-03 0.7464E-02 -0.5373E-02 0.5520E-03 1.109 - 90B -

NUMBER OF OBJECT IN SCAN 4

AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORD X EQ.DIAMFTER -ARITY X Υ. Х Y Y 145 152.4 215.8 0.706 145.0 170.3 677.9 18236 171 164 192 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNWARD AND INCREASING POWER OF Y AUROSS (0.1824E 05) (279.1)(316.1) (2.155) 0.5302F-01 0.1338 0.5242E-02 0.8294E-02 -0.2008E-02 -0.4565E-02 0.5698E-02 -0.5580E-03 0.3596E-01 0.5657E-02 0.1522E-03 COMPONENT 1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.073 -0.1047E-01 -0.3546E-02 0.5274E-01 0.3148E-02 0.1284 0,4855E-02 -0.1951E-02 -0.3499E-02 0.7381E-02 0_4932E-02 -0_4799E-03 0.3338E-01 0.5645E-02 0.1263E-03 MOMENTS OF INCREASING COMPONENT 1.5. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.9377E-02 0.3479 0.3537 0.3539E-01 -0.2023 0.3235 -0.3454E-02 -0.6083E-02 0.2869E-01 0.5016E-01 0.2974E-02 0.1354E-03 0.9681E-02 -0.6578E-02 0.2545 MOMENTS OF INCREASING COMPONENT 1.4. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.9816E-02 0.4069 -0.1833 3.117 0.6679 0.1676E-01 0.1377E-02 -0.4860E-02 -0.3904E-01 0.2845E-01 0.2490E-04 -0.6540E-03 0.9877E-02 0.2238E-01 1.020 MOMENTS OF INCREASING COMPONENT 1.3. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.2539E-01 -0.9993E-02 -0.6190 -1.314 0.4875 0.1785E-01 0.1155E-02 -0.3506E-02 -0.2417E-01 0.7294E-01 0.84366-03 -0.19246-03 0.68206-02 0.4487E-02 0.5048 MOMENTS OF INCREASING COMPONENT 1.2. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.2117E-01 -0.5398E-01 -0.4537 0.4201E-01 0.7207E-01 0.2593 0.2794E-02 -0.8485E-02 -0.2584E-01 0.6037E-01 011855 0.3693E-02 0.9620E-03 0.9724E-02 0.4629E-03 MOMENTS OF INCREASING COMPONENT 1.1. ORDER DOWNUARD AND INCREASING POWER OF Y ACROSS 0.6800E-02 -0.1769 0.6042 0.1338E=01 0.5574 0.8434 0.3458E-03 -0.5576E-02 -0.3354E-01 0.2559 0,1711E-01 1.617 -0.17882-01 -0.1159E-02 0.5828E-02 - 900 -

NUMBER OF OBJECT IN SCAN 5

AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORD EQ.DIAMETER HARITY X X X Y 216.4 318.0 0.681 124.7 297.4 998.9 36780 301 170 166 298 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (0.3678E 05) (227.8)(490.3) 0.4190E-01 (1.469)0.1679 0.1850E-03 -0.1976E-02 -0.2564E-02 0.7650E-02 0.3543E-02 0.4870E-03 0.5876E-02 -0.1279E-02 0.5299E-01 COMPONENT 1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.149 0.1480E-01 0.7796E-02 0.4502E-01 0.2521E-02 0.1441 0.9079E-04 -0.1184E-02 -0.8237E-03 0.3160E-02 0.3873E+02 0.1360E+03 0.5178E-02 -0.4121E-03 0:3935E-01 COMPONENT 1.6. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.9847E-02 -0.5293 0.3561 -2.724 0.1759 = 010.4425 0.3411E-03 -0.3292E-02 -0.1072E-01 0.7464E-01 0.4970E-03 0.6838E-04 0.6790E-02 0.8810E-03 0.3875 COMPONENT 1.5. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.5383E-02 -0.6797 -0.1608 0.1612E-01 -1.173 0.4877 0.5195E-03 -0.5889E-03 -0.2275E-01 0.4671E-01 0.4772E-03 0.3227E-03 0.7077E-02 -0.5853E-02 0.5089 COMPONENT 1.4. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.8007E-01 -0.3375 0.1727 0.1465E-01 0.6071E-01 0.5297 0,2917E-03 -0,3888E-02 -0,1402E-01 0.9877E-01 0.4449E+03 0.3628E+04 0.6890E+02 0.1943E-02 0,5666 COMPONENT 1.4.1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.1525E-01 0.5756 -0.2624 -26.54 1.698 0.6405E-01 -370.0 -1.942 0.1742 0.6249E-02 -4469 0.2982 0.71195-02 -34.87 0.3502E-01 COMPONENT 1.3. MOMENTS OF INCREASING ORDEP DOWNHARD AND INCREASING POWER OF Y ACROSS 0.1225E-01 0.6912 0.2049 0.4644E - 01 - 0.83320.2377 -0.3572E-02 -0.9458E-02 0.2241E-01 0.5872E-01 0.4738E-02 -0.6008E-03 0.9075E-02 -0.5620E-03 0.1530

Y

<u>, i</u>

COMPONEN	NT 1.2.	MOMENTS	OF INCREASIN	G
ORDER DI	HUNHARD AND	INCREASING PO	DWER OF Y ACR	0\$\$
0.38629-01				
0.4369	0.3116			
0.6622E-01	2.514	0.1294		
=0.1309E=02	-0.11315-01	-0.2378E-02	0.2328E+01	
0 10815-01	0 13866-02	0 61696-02	-0 7810F-07	0 4009 -01
U. EUGIE-01	0.1004C=02		-0.30176-03	0.40072-07
COMPONEN	1.2.1	MOMENTS	OF INCREASIN	G
ORDER DO	DUNWARD AND	INCREASING PO	DWER OF Y ACR	055
0.8089E-02				
0.3629	0.3178	1. A.		
0.2184E-01	-1.086	0.5288		
0.2329E=02	-0 3338E-03	-0.2459E-01	0.2218E-01	
0 13636-02	-0 26905-03	0 81675-02	0 7795E-02	0"5513
0.1000C-0C	-0.24/02-03	0.00000-06	0.177JL-02	0.0040
COMPONEN	чт 1 <u>1</u>	MOMENTS	OF INCREASIN	G
ORDER DO	DWNWARD AND	INCREASING PO	OWER OF Y ACR	055
0 3505F-01		x		
-0 1213	0 4005		· · · ·	
- 0,12,5 0,12,68=01	-3 127	0 5/87	· · · ·	
0 7010E-07	J - I G I 	-0 3360E-04	0 10025-01	
0.70106-05	0.000000000	-U.2209E-UI		0 ¹¹ /00/
0.4858E-03	U.1320E-03	- 0,6132E=02	-0.2160E-02	0.0220

Accore critical test of the shape describing parameters was devised by selecting five of the objects shown in Fig. 12. This group of 115 sand particles was scanned twice in mutually perpendicular directions. The 'clipping level', i.e. the grey level threshold between black and white, and the magnification factor were slightly different for each scan. Fig. 13 shows enlarged graphplotter outlines of the five particles from two scans. The objects were chosen to be fairly similar so that the power of the shape descriptors could be assessed.

A previously written subroutine (PARPAR) was used to calculate 'circularity' as defined in appendix B. The values of this shape factor are given in Table 2. for the two scans of five objects. Various methods have been used to compare them in attempting to determine the correct correspondence of objects. Table 2. shows assignments established by (a) considering the object in the second list with nearest circularity to each item in the first, (b) using the same procedure but with the first list considered relative to the second, and (c) by assigning objects in ascending order of circularity. It may be observed that only object number 2 is consistently assigned correctly. Corresponding values of area and perimeter are also compared in Table 2 both parameters allowing correct classification. Consistent values of perimeter are partially attributable to the fact that the two scans were at 90°.

- 91 -





Fig 13. Two scans of 5 particles from Fig. 12.

Table 2

Comparison of area, perimeter and circularity for the two scans of five sand particles in figure 13. Areas and perimeters are normalised to make totals equal for both scans.

Object Within Scan l	Number Scan. Scan 2	Number used for	Area	Area.	Perim -eter	Perim -eter	Circul -arity	Circul -arity
· · _	•	analysis	l	2	l	2	1	2
· 5	82	1	840	823	113.9	113.6	902	882
7	67	2	1158	1172	143.6	145.3	840	823
16	64	3.	792	788	106.8	107.6	934	911
17	75	4	1256	1258	141.4	141.4	889	876
27	66	5	905	907	120.2	117.7	887	894

Predicted correspondence of objects from two

scans using:-

Nearest to value Scan l	in scan 2 e in scan 1 Scan 2	Nearest to value Scan l	in scan 1 e in scan 2 Scan 2	Assignmer of circul Scan 1	nt in order arity valu Scan 2	r le
1	· 3	- 5	1	1	5	
2	2	2	2	2	2	•
3	3	1	3	3	3	
4	5	5	4	· 4	l	•
5	1	4	5	5	4	

Table 3. gives moment analyses of the five objects from both scans. The objects may first be grouped into convex (1 and 3) and non-convex (2, 4 and 5).When dealing with convex or near symmetrical objects it is important to consider the ratio of MO2 and M20 before analyzing higher order If this ratio is close to unity then it moments. is conceivable that a small distortion would cause orientation of the principal axes to change by 180 . Similarly, if M30 is very much smaller than M03 a discrepancy of 90° may occur, changing the signs of moments in odd powers of either x or y. In both cases comparisons may still be made, using elongation and the product M20.M02 and absolute values of odd power moments respectively. For objects 1 and 3 MO2/M20 is approximately 2 and comparison of these moments clearly shows the correct assignment.

The non-convex group may be further sub-divided since object 2 has two concavities whilst 4 and 5 have only one. This remaining pair of objects may easily be distinguished using either second order moments or by comparing the sizes of their concavities.

Shape factors of components 0. and 1. should be identical for convex objects. Numerical accuracy can therefore be assessed by comparison. Items which are not shape factors have been enclosed in brackets. Values in brackets are, in order of

- 92 -

(Object 1). NUMBER OF OBJECT IN SCAN 5 AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORD EO_DIAMETER -ARITY X Y X Y 31 33 32.0 113.9 840 31 39 0.902 3010 32.7 36.2 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (840.0)(325.1)(35.78) 0.1195 0.5439E=01 (-0.9978) 0.1144E-02 0.1001E-02 -0.5491E-03 -0.4955E-03 0.6276E-02 0.3689E-03 0.3750E-02 -0.8818E-03 0.3034E-01 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1,000 0.0000E 00 0.0000E 00 0.1195 0.5439E-01 -0.5183E-07 -0.5492E-03 -0.4954E-03 0.1144E-02 0.1002E-02 0.3034E-01 0.6277E-02 0.3678E-03 0.3752E-02 -0.8838E-03 82 NUMBER OF OBJECT IN SCAN AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORC EQ.DIAMETER -ARITY X X Y X Y Y 799 38 30 33 30 31.9 36.2 0.882 32.7 28.0 113.6 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (799.0)(148.8)(476.9) 0.5476E-01 (0.5381) 0.1186 -0.8797E-03 -0.5620E-03 0.1651E-02 0.1160E-02 0.2990E-01 0.6385E-02 0.2748E-03 0.3720E-02 -0.6144E-03 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.000 0,0000E 00 0.0000E 00 0.5476E-01 0_96258-07 0.1186 -0.8798E-03 -0.5617E-03 0.1651E-02 0.1160E-02 0.2767E-03 0.3715E-02 -0.6054E-03 0.2988E-01 0_6385E-02

– 92A –

Table 3. cont'd. (Object 2, Scan 1). NUMBER OF OBJECT IN SCAN 7 AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FFRET DIAM MAX, CHORE EQ.DIAMETER HARITY X Y X Y X ¥. 38.4 45.7 0.840 30_0 50,7 143.6 1158 34 53 34 52 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (1158.) (412.0) (37.99) 0.5214E-01 (-1.506) 0.1449 0.1721E-03 -0.7276E-02 0.7402E-03 0.2416E-01 0.6087E-02 -0.1964E-03 0.4782E-02 0.7311E-03 0.4772E-01 COMPONENT 1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.035 0.1315E-01 0.5051E-02 0.1513E-01 0.5133E-01 0.1413 0.2021E-03 -0.6553E-02 0.3550E-04 0.1979E-01 0.5754E-02 -0.1119E-03 0.4783E-02 0.3549E-03 0.4364E-01 COMPONENT 1.2. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACRUSS 0_8636E-02 0.4513 -0.2556 0.1131E-010.3894 2.290 0.8794E-03 -0.4012E-02 -0.2982E-01 1,091 -0.1246 0.2396 -0.4301 0.8938 8,932 COMPONENT 1.1. MOMENTS OF INCREASING

ORDER DOWNHARD AND INCREASING POMER OF Y ACROSS 0.2591E-01 0.3749 0.7596E-02 0.7596E-02 0.7137E-04 0.1276E-02 0.1600E-01 0.3866 0.6203E-03 0.1832E-02 0.1730E-01 0.1347E-01 4.853

Table 3. cont'd. (Object 2, Scan 2). NUMBER OF OBJECT IN SCAN 67 ARFA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHOPI E0_DIAMETER -ARITY X v X Y X Y 38 1 46 3 0 823 51.1 29.9 145.3 1138 53 34 53 34 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (1138.) (150.8)(390.3)0.5071E-01 (0.5055E-01) 0.1472 -0.5500E-04 -0.6812E-02 0.1008E-02 0.2378E-01 0.5790E-02 -0.11/9E-03 0.4634E-02 0.6350E-03 0.6350E-03 0.4883E-01 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.054 0.1945E-01 0.7497E-02 0.5003E-01 0.1880E-01 0.1410 0.1671E-04 -0.5740E-02 0.5280E-04 0.1752E-01 0,5420E-02 -0.1199E-04 0.4612E-02 0.1795E-03 0.4267E-01 COMPONENT 1.2. MOMENTS OF INCREASING ORDER DOWNHARD AND INCREASING POWER OF Y ACROSS 0.1494E - 01-0.2798 0.4288 0.5762E-02 0.3928 2.071 -0.7389E-04 0.8654E-03 -0.2874E-01 0.8488E-01 0.9136E-02 -0.3800E-02 -0.2083E-01 0.1189 8,210 COMPONENT 1.1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.3954E-01 0.3566 0.3022 1.296 0.6589E-02 2.801 0.4845E-03 -0.6939E-03 -0.3151E-01 0.2052 0.1991E-01 0.2773E-01 0.4099E-01 0.2928E-01 3.439

÷., .

NUMBER OF OBJECT IN SCAN - 16 AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORD EQ. DIAMETER -ARITY Y Y Y X X 31.8 34.0 0.934 32.0 34.8 106.8 792 35 32 35 33 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNNARD AND INCREASING POWER OF Y ACROSS (792.0)(78.75)(421.3)0.5845E-01 (-2.227) 0.1108 0.6812E+03 -0.1359E+02 -0.9479E-03 0.2134E-02 0.6678E-02 -0.2776E-03 0.4884E-02 0.4748E-03 0°2364E-01 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.000 0.0000E 00 0.0000E 00 0.5845E-01 -0.9467E-07 0.1108 0.6811E-03 -0.1359E-02 -0.9472E-03 0.2134E-02 0.6677E-02 -0.2790E-03 0.4882E-02 0.4700E-03 0.2363E-01 NUMBER OF OBJECT IN SCAN - 64 AREA FERET DIAM MAX, CHORD AREA PERIM CIRCUL MARTIN DIAM PERIM Y X Y X EO.DIAMETER -ARITY X Y 35 32 765 35 33 31.2 34.3 0.911 34.5 30.0 107.6 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (765.0) (381.2)(191.6)0.5792E-01 (-0.6393) 0.1125 0.1087E-02 -0.1435E-02 -0.1810E-02 0.2250E-02 0.24448-01 0.6594E+02 +0.2831E+03 0.4932E+02 0.4926E+03 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1,000 0.0000E 00 0.0000E 00 0.5792E-01 -0.2910E-10 0.1125 0.2250E-02 0_1088E-02 -0.1435E-02 -0.1810E-02 0.24446-01 0.6590E-02 -0.2857E-03 0.4931E-02 0.4918E-03

92D

NUMBER OF OBJECT IN SCAN 17 AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHORD EQ_DIAMETEP -ARITY X Y X. Y 40_0 45_0 41_0 141_4 38.0 1256 0.884 43 46 39 43 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (1256)(75.93)(368.2) 0.5380E-01 (0.8397) 0.1248 0.7351E-03 -0.36/9E-02 -0.1733E-02 0.7275E-02 0.5761E-02 -0.5768E-04 0.5191E-02 0.3489E-03 0.3033E-01 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS i.021 0.4672E=02 0.7919E=02 0.5458E-01 0.2572E-01 0.1211 0.8420E-03 -0.2976E-02 -0.2448E-02 0.5419E-02 0.5834E-02 0.1807E-04 0.5101E-02 0.9550E-04 0.2855E-01 COMPONENT MOMENTS OF INCREASING 1.1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.2070E-01 0.3905 0.2304 0.22°5E-01 2.908 0.4705 0.1711E-02 -0.7356E-04 -0.3996E-01 0.6505E-01 -0.5106E-01 -0.4617E-01 -0.3223E-01 -0.4371E-01 0.5518 NUMBER OF OBJECT IN SCAN 75 AREA PERIM CIRCUL MARTIN DIAM PERIM AREA FERET DIAM MAX.CHOR EQ.DIAMETER -ARITY X Y X Y X Y 41.7 39.4 45.0 0.876 37.9 141.4 1221 45 44 42 38 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (1221.) (434.0)(188.7) (2.498)0.5290E-01 0.1269 0.8142E-03 -0.3489E-02 -0.2078E-02 0.7092E-02 0.5557E-02 -0.5470E-04 0.5229E-02 0.3343E-03 0.3121E-01 COMPONENT 1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1,017 0.35318-02 0.6381E-02 0.5354E-01 0.1840E-01 0.1237 0.8778E-03 -0.2948E-02 -0.2681E-02 0.5593E-02 0.7453E-05 0.5135E-02 0.56216-02 0.2969E-01 0.1222E-03 COMPONENT 1.1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.1679E=01 0.2438 0.3865 0.28505-01 2.874 0.4595 0.3720E-02 -0.1547E-02 -0.4908E-01 0.7608E-01

-0.2455 -0.1265 -0.5699E-01 -0.3727E-01 0.6099

NUMBER OF OBJECT IN SCAN 27 AREA PERIM CIRCUL MARTIN DIAM PERIM CAREA FERET DIAM MAX.CHORD EO, DIAMETER -ARITY X Y X Y X Y. 31_0 32_0 120_2 905 31 37 31 37 33.9 38.3 0.887 MOMENTS OF INCREASING COMPONENT 0. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (905.0) (413.1)(118.2)0.6803E-01 (-2.197) 0.1018 0.3414E-02 -0.2303E-02 -0.2441E-02 0.53.88-02 0.2359E-01 0.10486-01 0.99%6E-04 0.3520E-02 0.3007E-03 MOMENTS OF INCREASING COMPONENT 1. ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1_039 0.1298E-01 0.1364E-01 0.6729E-01 0.1300 0.9968E-01 0.4230E-02 -0.2057E-02 -0.2731E-02 0.2197E-02 0.2171E-01 0.9843E-02 0.5767E-03 0.3766E-02 -0.7171E-03 COMPONENT 1.1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 0.3867E-01 0.3662 0.3487 0.4725 0.2092E-01 -0.4471 -0.1556F-02 -0.2098E-02 0.3825E-01 0.3603E-01 0.2774F-02 0.1197E-02 0.6069 0.5591E-02 0.4305E-01NUMBER OF OBJECT IN SCAN 66 AREA PERIM CIRCUL MARTIN DIAM PERIM APEA FERET DIAM MAX.CHOR EQ.DIAMETER -ARITY X Y X ٧ X Y 33.5 0.894 36 37.5 33.0 30.3 117.7 881 31 36 31 COMPONENT 0. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS (881.0) (389.5)(230.8)0.6714E-01(-0.6317) 0.1024 -0.1787E-02 -0.2171E-02 0.29926-02 0.51438-02 0.1013E-01 -0.1021E-04 0.3464E-02 0.3353E-03 0.2380E-01 COMPONENT 1. MOMENTS OF INCREASING ORDER DOWNWARD AND INCREASING POWER OF Y ACROSS 1.037 0.7040E-02 0.9053E-02 0.6605E-01 0.1131 0_1006 -0.1413E-02 -0.2251E-02 0.17456-02 0.4205E-02 0.9479E-02 0.3734E-03 0.3625E-02 -0.5513E-03 0.2216E-01 COMPONENT 1.1. MOMENTS OF INCREASING ORDER DOWNPARD AND INCREASING POWER OF Y ACROSS 0.3065E-01 0.3419 0.3620 0:2138E-01 -0.4534 0.4400 -0.2054E-02 -0.2315E-02 0.30146-01 0:4937E-01 0_1471 0.3336E-01 0.1969E-01 0.5796E-02 0.4389

appearance, object area in picture points, centroid coordinates and angle of orientation measured in radians relative to the positive y axis.

Fig. 13. shows that all five objects are fairly similar with no 'wild' features. Also, the two scans produced boundaries with a considerable number of small but not insignificant discrepancies. Nevertheless, it is apparent that only a few of the shape factors calculated are needed for correct classification. One therefore infers scope for considerably more complex tasks to be performed. Low order moments appear to be excellent descriptors of general shape, whilst higher order values are more influenced by points distant from the centroid and may be interpreted as characterizing finer detail. Moments as high as tenth order have been calculated using the existing programme but as can be observed the number of significant decimal digits diminishes fairly rapidly with increasing order. Using moments up to fourth order, as shown, ten shape factors are available even for convex objects. Under most circumstances this level of analysis should be more than sufficient.

Where distortions are large it may be difficult to reach a satisfactory compromise between the value of AMIN and the order of moments calculated. Increasing the value of AMIN reduces the risk of bogus components being detected but if the number of components is reduced toodrastically correct classification may become impossible.

- 93 -

One's task for the purpose of this thesis has primarily been to develop and implement a method of shape analysis. This has been achieved. However, one of the design considerations was that either of the two parts of the analysis, namely, convex decomposition and invariant moment analysis could be used separately. This flexibility allows considerable further exploitation.

The precise method in which moments should be relativised is open to experimentation and it is doubtful if there is a single method which will be 'best' in all circumstances. It has previously been suggested that each non-convex component should be considered relative to its convex hull. Thus the ratio of corresponding moments of these two objects could be used as distortion tolerant parameters. Alternatively. if the shape of each convex component is considered important its shape describing moments may be made invariant in the normal manner i.e. by boundary This approach has been used so standardisation. far although the area, centroid coordinates and orientation of components have been relativised to the original scan object so that they also become shape descriptors.

By further processing it is possible to generate elongation invariant moments. This invariance may be achieved in addition to those

- 94 -

previously mentioned. The term elongation may be applied to the ratio of the moments M2O and MO2. This factor may be removed from moments by using the equation:-

Mpq(elongation invariant) = Mpq/(M20/M02)^{p-q/4}

Applying this transformation, both M2O and MO2 become $\sqrt{M2O.MO2}$. It is difficult to find an adjective to describe this term which has not been used for a different shape factor by another author. Its magnitude will be a minimum for a perfect circle and will be large for objects having long narrow projections along the x and y axes, e.g. a plus sign.

Factors which are ratios between components and convex hulls will always have magnitudes between zero and unity. This fact is particularly useful when considering numerical stability. It cannot be guaranteed that the set of moments currently produced will decrease in absolute value for increasing orders of x and y. Indeed, should any boundary point lie at a distance greater than or equal to one unit from the centroid coordinates (after standardisation) values will diverge with increasing order. If size standardisation is achieved by normalising the size of the object convex hull to unity and affine invariant moments

- 95 -

are used then it can be shown that higher order moments will eventually converge to zero. An affire transformation may be defined as a linear transformation plus a translation i.e. an affine invariant is invariant with respect to size, position, orientation and elongation. It is conjectured (the proof would be quite long) that the maximum distance of a boundary point of any object normalised in this manner would be $2/3^{0.75} \div 0.875$ units from the centroid.

The prospect of using weighting functions in moment calculation has previously been mentioned. Ideally, such weighting functions should give greater weight to points near the centroid in order to reduce the relative error magnitude. Two possible functions are the negative exponential and reciprocal. Both functions may be expressed as a series expansion e.g. for one dimension:-

 $\frac{f(x)}{1+x} = (1-x+x^2-x^3+\cdots) \cdot f(x)$

 $f(x) \cdot e^{-x} = (1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \cdots) \cdot f(x)$

The expansions are valid only for x < 1 hence normalisation along the lines suggested should first be implemented. These functions may then be calculated to any desired accuracy by summing the corresponding terms of the equivalent moment series expansion e.g.

 $f(x,y).e^{-x} = M00 - M10 + \frac{M20}{2!} - \frac{M30}{3!} + \cdots$

<u>-96-</u>

Similar expansions exist for other exponent terms which are simple expressions in x and y. Moments in negative powers of x and y may not be used since the centroid would then have infinite weight.

Unfortunately, the amount of computation required to evaluate high order moments is quite large. Since the number of boundary points involved should be much greater than the number of moment values most of this work will occur in calculating the initial simple moments. For each point and every moment calculated involving that point the following expression is evaluated:

$$\frac{(x+\frac{1}{2})^{p+1} - (x-\frac{1}{2})^{p+1}}{p+1} \cdot \frac{y^{q+1}}{q+1}$$

If all moments up to the Rth order are calculated then clearly this expression may be derived for every moment concerned from two sets of factors one involving x and the other y. If we assume that most computer time will be engaged in multiplication and division and consider either to be equivalent to one arithmetic operation then:-

> Operations to calculate x factors = 3.ROperations to calculate y factors = 2.ROperations to find moments from factors = $\frac{(R+1).(R+2)}{2}$

hence total number of operations per boundary point

 $= \frac{1}{2}R^2 + 6\frac{1}{2}R + 1$

- 97 -

When R < 13 the second term of this expression will be larger than the first. For practicle purposes one may assume that the amount of computation required is approximately proportional to the total number of moments evaluated (about 2 to $2\frac{1}{2}$ operations per point per moment).

The number of moments evaluated should vary with the size of the component involved and the type of classification for which shape factors are required. In the case of components which are only slightly larger than AMIN the most important factor will be that of existence. However, since existence is determined by evaluating component area the zeroeth order moment will always be It should be possible for a given available. analysis to establish thresholds above which the next higher order of moments will be calculated. Judicious use of such thresholds would greatly reduce the computation required for effective shape description.

Suggestions for Further Work.

In the introduction it was mentioned that several lifetimes of possible research became apparent whilst reviewing work done in image processing. As might be expected this amount has increased during the past year. This section is therefore restricted to an outline of further developments and application of one's current research.

One of the major obstacles encountered in image analysis is that of touching or overlapping objects. The former problem is frequently a function of the surface adhesion of individual objects whilst the latter is related to the fact that the objects are three-dimensional, only a two-dimensional projection being available for It is uncertain whether the former analysis. problem should be tackled by reducing surface adhesion, i.e. physical means, or by image processing. Certainly physical separation would make life much easier for the image processor but even where practicable a different method would be needed depending upon the type of objects involved. Software written to perform separation must rely upon estimated boundary statistics to guess 'most likely' positions for missing boundary points. Even when all the boundary points are available unless it is known that objects are touching but

- 99

do not overlap there exist cases in which even a human will be unable to uniquely determine the boundaries of individual objects.

If overlapping objects occur the situation becomes much more difficult. Where boundary statistics are well defined it should be possible to define a filter to detect the presence of an overlap. If the individual objects are convex (or possess only small concavities), i.e. less than AMIN, then overlaps may be detected by the simple application of convex decomposition. The number of level 1 concavities found should be twice the number of overlapping objects. Otherwise, if more than one overlap occurs within the same concavity, lower level components should establish the precise number. Unfortunately, a number of cases exist in which this method will be inaccurate, e.g. where an object is totally

included in an agglomerate. Also, if two objects are almost totally overlapped, any concavities found will be small and may therefore be erroneously ignored by the thresholū (AMIN).

The problem may be reduced to that of touching objects if complete three dimensional information is available. A number of methods for obtaining such information have been tried by various researchers. An interesting technique

- 100 ·

utilizes the phenomena of interference. A coherent light or ultra-sonic sound beam is directed onto the object. Part of the same beam is deflected from the source to the receiving medium where it interferes with light(or sound) reflected from the object. The phase relationship between the two beams supplies the third dimension or depth information. A similar method involves placing a grid between the object and light sensor. Again, reflected light interferes with light direct from the source at the grid. A moire fringe interference pattern comprising depth 'contours' is formed. This latter method may use normal incoherent light but grey level detail of the object will be lost. None of these methods enable total 360°. 3-D information to be collected and suffer from the snag that every point scanned must be in direct line of sight with both source and image sensor. To the author's knowledge no 3-dimensional scanning systems have yet been marketed commercially for image digitisation.

The concept of convex decomposition could readily be extended for use in three or more dimensions. Convex polygons would be replaced by convex bodies (polytopes), but the component connection tree would remain in the same form. Implementation in three-dimensions would of course require more computation although it, is envisaged that decomposition would involve the same basic steps.

- 101 -

Invariant moment analysis could also be applied to the components.

It would not be difficult to impose the constraint that any component accepted as convex should be star-shaped about its centroid. The relationship $r = f(\Theta)$ would then be single valued thereby allowing numerous methods of analysis. In particular, the Fourier descriptors mentioned by Zahn(35) could be used.

Analyses given as results required only a few shape factors for correct classification. It would be most valuable to study the precise relationships between individual factors and object shape. Affine invariant moments have been suggested but for any group of objects it should be possible to determine the linear correlation between each pair of moments and by suitable manipulation, e.g. using Eigenvalues or factor analysis, create a set of uncorrelated measurements.

CONCLUSIONS.

A method of shape analysis has been derived, implemented and illustrated. The technique of convex component decomposition does not appear to have been used before and has greater stability under distortion than alternative methods considered. Computation required is quite moderate.

Equations have been derived for the efficient machine generation of moments invariant with respect to size, position and orientation. When applied to the convex components a detailed object description is obtained.

One's primary aim has been to by-pass a stumbling block encountered by others, namely, analysis of objects with concavities. The method used capitalizes upon the existence of these concavities which are treated as important features. Classification becomes more presise for concave objects than for convex ones. Results are presented illustrating the discriminatory power of the shape factors derived.

- 103 -

Appendix A.

<u>Two-dimensional integral transforms.</u> ZW or discrete Laplace Transform.

This has the general form:-L $[f'(u,v)] = \sum_{x} \sum_{y} f(x,y) \cdot exp(-sux-tvy)$ where s and t are complex variables and \sum_{x} means $\sum_{x=0}^{x=N-1} \sum_{y} \text{ means } \sum_{y=0}^{y=N-1}$ for an N x N element picture. L[f'(u,v)] is the value of the transform at position -(u,v) in the transform plane. It may be observed that every point in the original picture is used to determine any point in the (u,v) plane, also the (u,v) plane is continuously defined even though only discrete points were known on the (x,y) plane.

Normally transformed values are calculated for a matrix of N x N values. This allows reconstruction of the original picture using an identical type of transformation with s and t substituted by their complex conjugates. The transformation is said to be separable since it may be considered as a two stage process, each involving one variable only:-

 $L[f'(u,v)] = \sum_{x} exp(-sux) \sum_{y} exp(-tvy). f(x,y)$

Separation allows a considerable saving in computation since each line of the picture may first be transformed and the final transform evaluated from these line transforms.

- A104 **-**

The procedure may be considered in matrix form as :-

$[Lf'] = [T] \cdot [f] \cdot [S]$

where all matrices are of size N x N and the v,yth element of [T], $t_{vy} = \exp(-t.v.y)$ and the u,xth element of [S], $s_{ux} = \exp(-s.u.x.)$

The discrete Laplace transform is often considered in terms of the complex variables^Z $\exp(s.u)$ and w = exp(t.v.). The transform equation becomes:-

$$f(z,w) = \sum_{x} \sum_{y} f(x,y) \cdot w \cdot z^{-y}$$

i.e. the transform is a complex polynomial in negative powers of $_z$ and $_w$, which when expanded becomes a rational polynomial.

Fourier Transform.

This may be considered as a special case of the Laplace transform where $s = t = 2\pi j$, $(j = \sqrt{-1})$. The transform equation may be written:-

$$F(u,v) = \sum_{x} \sum_{y} f(x,y) \cdot \exp(-2\pi j(ux+vy))$$

and in matrix form:-

[F] = [S].[f],[S]

Hadamard Transforms.

Let [H] be a Hadamard matrix, all coefficients being +1 or -1. Then:-

 $[H] \cdot [H]^{T} = N \cdot I$, where I is the identity matrix and all matrices are of order N. The simplest Hadamard matrix is $\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = H_2$. This may be used to generate all Hadamard matrices of order 2ⁿ by the equation

$$H_{2n} = \begin{bmatrix} H_n & H_n \\ H_n & -H_n \end{bmatrix}$$

The number of sign changes along a row of such a matrix is known as the sequency (cf. frequency). For symmetric Hadamard matrices of order $N = 2^{n}$.

$$F(u,v) = \frac{1}{N} \sum_{x} \sum_{y} f(x,y) \cdot (-1)^{b(x,y,u,v)}$$

where $b(x,y,u,v) = \sum_{i=0}^{i=N-1} (u_{i} x_{i} - v_{i} y_{i})$

and $u_i = sum$ of binary digits in u and the summation of b is performed in modulo 2.
Appendix B.

A Glossary of shape describing terms.

'oblongness'

'stringiness'

Largest ratio between length and width for all possible circumscribed rectangles.

2 times area/perimeter. If object of linelike parts of width w the $w \doteq stringiness.$

Convex envelope boundary. Vertex of convex hull.

Vertex of polygonal object boundary.. (also 'bay' and 'peninsula') closely adjacent pairs of curvature points with opposite curvature, produced artificially during quantisation. Number of curvature points (excluding inflections)/perimeter.

Number of occurrences of adjacent bends of opposite signs not classifiable as inflections.

Object having at least one internal point from which a line to any point on the boundary does not cross boundary elsewhere.

Sum of the second order moments about the principal axes i.e. M20-M02. M20-M02.

'convex hull'
'convex point'
'curvature point'
'inflection'

'curvaturerate'

'wiggliness'

'star-shaped'

'spread'

'slenderness'

- B107 -

'equivalent area diameter'

'equivalent perimeter

diameter'

'Martin's diameter'

'Feret's diameter'

'Maximum linear

diameter!

'diameter-to-length

ratio'

'general rotundity'

or 'roundness'

Diameter of circle with the same area as the object.

Diameter of circle with the same perimeter as the object. Length of the line through the particle profile, parallel to a fixed direction, which divides the particle profile into two equal areas.

Projected length of the particle profile with respect to a fixed direction.

The length along the major (longest) axis, also refered to as simply 'length'.

(length of a chord normal to the major axis)/(maximum linear diameter)

Average value of diameter-tolength ratio of a group of objects an equal number of measurements being made on each object.

Sources of definitions ...

Rosenfeld(1), Kaye(48), Zahn(26)

Appendix C.

List of periodicals and series' of publications frequently containing articles on image processing.

I.E.E.E. Transactions on: Audio and Electroacoustics, Biomedical engineering, Computers, Automatic control, Information theory and Systems, man and cybernetics. Journal of the association of computing machinery (J.ACM) Communications of the ACM (CACM).

Journal of the optical society of America.

Scientific American.

Journal of pattern recognition.

Advances in information processing, Plenum press.

Advances in computers, Academic press.

Machine intelligence, Edinburgh university press.

Appendix D.

Listing of computer subroutines.

			SUBROUTINE INITIAL	
,			COMMON IAMIN, SCAF, IOUT	
			COMMON/B1/NO, 108(100)	
			COMMON/BA/MAXX, IARC	
			EXTERNAL XYNSTACK	
	C		SET UP ERROR RECOVERY	
	-		IE(IENT.NE.0) 60 TO 3	
			IENT=1	
			READ(1,1) TAM, LCH, TMAX, MNC, TARC, NTHO, NST, NCORE, TAMTN, TOUT	•
		- 1	FORMAT(2010)	
			READ(1.14) NTADES -	
		14	FORMAT(1010)	
	C	•	TE TOUT OF O DEOT PARTICLE EDGES	
	•	•	IF(IOUT.GT.O) CALL UTPOP	
		19	CALL PREADS(IXX, IYY, MAXX, MAXY)	
			RFAD(1.99) NO. (IOB(JJ).JJ=1.NO)	
			NOP1	
		00	FORMAT(10010)	
•			REFERENT GT O) CALL PLOTST(MAXX,MAXY,SCAF)	
	6	•	ALLOCATE SPACE AND SET UP LENGTHS FOR PLEXES	
	•		CALL STACKS (NCORE, LARC+LARC, 90000, NST, LARC, 10)	
	Ĉ		CALL YVNSTACK AFTER ALLOCATING ARRAY SPACE	·
			CALL DIME (XYNSTACK, 5, TAM, LCH, IMAX, MNC, NTHO, 4	
			IS. TAN. ICH. TMAX. MNC. NTHO)	
		T	CALL ENDSTACKS	
		5	NTADESENTADESEA	
	0		TEST IF ANY PICTURES REMAIN TO BE PROCESSED	
	· •		TECNTADES ED A) GO TO 13	
			60 TO 40	
		42	TEATAIT OT AN CALL HTDEL	
		13	STAD	
			ENA	

EGMENT, LENGTH

150, NAME INITIAL

```
SUBROUTINE XYNSTACK(IV,ICHS,JNT,LCL,JPO,IAM,LCH,IMAX,
  1 MNC, NTHO)
   COMMON/BN/NCL
   COMMON/BA/MAXX, IARC
   DIMENSION IV(IAM), ICHS(LCH), JHO(NTHO), JNT(IMAX), LCL(MNC)
   STREAMLINED EDGE FOLLOWER UPDATED APL 1972 N. BISHOP
   CALL CLEARO(1,ICHS)
   CALL OPSTACK(JHOLE,NTHO)
   NP=1
   NHO,NCL:JX=0
   KMAX,JMAX=IMAX=4
   ILEJMAX
   Inll+2
  IMIN_JMAX-IL
   CALL PMOVE(JNT(IL+1), JNT(1), IMIN)
   1=1→1L
   IL¤KMAX-IMIN
   CALL prEAD8(JNT(IMIN+1),IL)
   JMAX=IL+IMIN
   J==1
 4 N=MOD(JNT(I),50000)
16 NT=I+N+N+1
   IF(NT-JMAX-1)8,13,18
18 IF(JX-MAXX)9,19,17
  IF(JNT(NT), EQ. JX+1) GO TO 13
 8
   1F(JX+1-MAXX) 14,19,18
19 NT=I+1
13 JNT(NT)=50000
   11=1
   19==1
   JNT(1), JNT(1-1)=100000
 1 IP¤Ip+2
   1=1+2
   JY=JNT(I=1)
   JYPL, JNT(I) = JNT(I) + JY
10 J=J+2
   JYPLL=JNT(J+1)
   1F(JVPL~JVPLL)2,3,5
 5 IF(JYPLL-JY)33,33,6
 6 IF(JNT(J+2)-JYPL)44,3,3
 2 IF(JVPL-JNT(J))11,11,7
   IF(JNT(I+1)-JYpLL)22,3,3
 7
 3 LCO=ISTACK(IV(TP+1), JYPL)=ISTACK(IV(IP), JY)
   ICHS(LCO)=ICHS(LCO)+1
   GO TO 1
99 K=0
   IF(JV_LT_40000) GO TO 12
   JX = JX + 1
   GO TO 4
22 K=1
12 CALL PMOVE(IV(1P+K),1V(1P+2+K),NP-IP-K)
   START 2 STACKS SETTING UP STACK HEADER
   FIRST & ITEMS ARE LABEL FOR CURRENT ARE CHAIN, ARE WHICH.
   MAY HELP IN LOCATING HOLES, NEXT CLOCKWISE ARC, JOIN TYPE,
   NEXT ANTI CLOCKWISE ARC, JOIN TYPE, UPPER OR LOWER ARC
   INDICATOR, FIRST X COORDINATE
   CALL OPSTACK(JP, IARC)
   CALL OPSTACK(LP, IARC)
   IV(IP+1)=INSTACK(LP,1,10,LP,0,0,0,JP,1+K,1,JX,0,0)
   IV(IP+K+K)=INSTACK(JP,1,10,LP,0,LP,1+K,0,0,-1,JX,0,0)
   NP=NP+2
   J=J-2
```

- D112

C C C

Ĉ

C

C .

	•	С	
•	151K FR. 03 GO TO 3		
	CALL INCOMPTIO 2. IV/IDAA		
	THE INDIVICESCATION DATA AND AND AND AND AND AND AND AND AND AN		
	TEAN OF THAT AND THAT AND A THAT AND AND A THAT AND		
	GO TO 3	•	
- 33	K=0		
	60 70 34		
44	Κ#1		
- 34	JB#IV(!P+K+K)		
	JAEINSTACK(IV(1P+1),3,2,JB,3+K)		
	CALL THETACK/ ID 5.2. LA . 34K)		
	JP=KSTAUK(JA)1)		,
•	LP=KSTACK(JB,1)		
	NP=NP=2 ·		
	CALL DHOVE (IV/ID+2+K).IV(JD+K).ND+ID+K)		
	THAIN NO INC. CLUCKET CONTRACTOR STRUCTURES IN THE STRUCTURE STRUCTURES IN THE STRUCTURES AND A STRUCTURES A		
÷	IF(JP.E4.LP) 00 10 35		
	£7=3		
	IF(KSTACK(JP,8)-KSTACK(LP,8)+K)38,38,36		
	- IDe IA		
-30			
		• .	•
	JP≖LP .		
38	CALL INSTRI(JB,1,JP)		
•••	JREVCTACV(JR. 17)		
	$\mathbf{v} = \mathbf{v} + $		
	16(28'NE'N) NO IO 20		,
	GO TO 10		
35	CALL INSTK1(JP,1,JX-1)	•	
• •	CALL PARTICLECICL, MNC. JHO, NTHO, NHO, JHOLE, JI	Paky	
	$e^{-e^{-e^{-e^{-e^{-e^{-e^{-e^{-e^{-e^{-$		
14	WRITE(2,15) JNT(NT)/JX/NT	•	
15	FORMAT(ERROR IN X SEQUENCE ,616)		
	FINISH SCAN AND RETURN	•	
	ENTRY COOPT/IFD)	·	
	DAINE BEFRIVIERZ	. •	
	WRITE(2/34) IER		
20	WRITE(2,39) JNT		
35	FORMAT(1018)		
-	TEXINAN IT ((TMAX+4)/403+40) CALL INTTAL		
	TRANANSEL CONNECTIONS		
	JUAYERIUAY		
	CALL PREADB(JNT(1), JMAX)		
	GO TO 20	•	
17	CONTINUE		
14		¹	•
	NA CO THILIN		
23	IF(ICHS(LCH-I+1),NE.U) GD TU 24		
24	WRITE(2,28) (ICHS(J),J=1,LCH-I+1)		
28	FORMAT(//24X, TOTAL CHORDS OF EACH LENGTH	INCREASING	FROM
	ST ONEF///449.401633		
	I DITE FELIDATIVAVES		
	KETURN		
	END .		
	•		

GMENT, LENGTH

Ĉ

642. NAME XYNSTACK

D113

			SUBROUTINE PARTICLE(LCL, MNC, JHO, NTHO, NHO, JHOLE, JP, K)
			CUMMUN/BN/NCL NTWEDOTON (BO/NTHO) (CL/MNC)
			SUPERSION JOURNARY CONDUCTED TEET TE ALUND OR HOLE
	L,		TRAN FO AN CO TO IS
•			17 (K, FJ, G) GU TU 43 NOLE FOUND
	C		NULE FOUND
	•		JHO(NHO)=JP
		•	KETURN TO THE TOP HELEN
	Ç		CLUMP FOUND LOOK FOR HOLES
		42	NGL=NGL+1
			IAMIN=100
			NPAR=200
			IHO=0
			CALL OPSTACK(LNCL,NPAR)
			IF(NHO_EQ.0) GO TO 47
			CALL INSTRACK(JP,5),3,0)
	,		JTHJP
	•	41	DO 48 K=1,NHO
		43	IF(JT.NE.KSTACK(JHO(K),2)) GO TO 48
			IHO=IHO+1
		1.	CALL ISTACK(JHOLE, JHO(K))
			CALL PMOVE(JHO(K+1), JHO(K), NHO+K)
			NHO=NHO-1
		•	IF(NHO.EQ.0) GO TO 42
			IF(K.LE.NHO) GO TO 43
		48	CONTINUE
			JT#KSTACK(JT,3)
	• .		IF(JT.NE 0) GO TO 41
		42	CALL INSTEI (KSTACK (JP, 5), 3, JP)
		47	NS.LCI/NCI)=INSTACK(LNCL+1+3+JP+IHO+1STACK(JHOLE)-IHO)
			CALL DARDAR(NS.JP)
			$\frac{1}{2} \frac{1}{2} \frac{1}$
			TCADWO
			NARCTARKING ZY
	•		NA DE VET VETARVING DI
			JUCT R-IJKJINGKUNJEJ
			LELSTACK(NS)
			JAMINAWI JAMINAWI
		54	LALL PARPAR(NO,KOTALK(JAVLE/M+K))
		21	
		39	GALL CLSTACK(NS)
	-		KETURN
			RND

GMENT, LENGTH 247.

NAME PARTICLE

```
SUBROUTINE PARPAR(NS;NT)
   COMMON/BI/NO, IOB(100)
   COMMON/BN/NCL
   COMMON IAMIN, SCAF, IOUT
   DIMENSION DIST(200), 1W(400), 1DW(400)
   IF(NCALL_NE.0) GO TO 90
   DO 18 I=1,200
48 DIST(1)=SQRT(FLOAT((I=2)*I+2))
   NCALL=NCALL+1
90 CONTINUE
   IAREA=0
   PERIM=0.
   P1=3.1415827
   RP1=1./P1
   RP14=4.*RP1
   JT, LS=NT
   IF(LSTACK(NS), EQ.3) JT, LS=KSTACK(NT,3)
   CALL INSTR1(KSTACK(JT,5),3,0)
   IF(IAMIN_LT.0) GO TO 7
   IYMIN=100000
   IYMAX, LMAX==IYMIN
   KP=11
   IF(JT.EQ.NT) GO TO 2
   IP=2+MOD(KSTACK(JT+6)+2)-1
   KP=0
   1P=1
 2 1P=-1P
   PERIM=PERIM+ISIGN(KSTACK(JT,KP),IP)
   IP#IP*(1=2*MOD(KSTACK(JT+4)+KSTACK(JT+6)+2))
   PERIM=PERIM+ISIGN(KSTACK(JT,11=KP), IP)
   IAREA=IAREA-ISUMSTK(JT)
   IYMIN=HINO(IYMIN,MINSTACK(JT))
   JT#KSTACK(JT,3)
   IP==IP
   PERIM=PERIM+IS:GN(KSTACK(JT,11-KP),IP)
   IP=IP+(1-2+MOD(KSTACK(JT+4)+KSTACK(JT+6)+2))
   PERIM=PERIM+ISIGN(KSTACK(JT,KP),1P)
   IAREA=IAREA+ISUMSTK(JT)
   IYMAX=MAXC(IVMAX,MAXSTACK(JT))
   JTHKSTACK(JT+3)
   IF(JT.NE.0) GO TO 2
   JT#LS
   IXMIN=KSTACK(NT,8)
   3XM=IXMIN-1
   IXMAX=KSTACK(NT/1)
   IFX=IXMAx=IXM
   IYM=TYMIN=1
   IYMM=IYM-1
   IFY=IYMAX-IYMIN
   CALL INSTACK(NS,LSTACK(NS)+1,5,IFX,IFY,IXMIN,IYMIN,IAREA)
   IF(IAREA_LT.IAMIN) GO TO 7
 6 CALL CLEARO(2, TW, IDW)
   PERIM=PERIM-2.
   LYEKSTACK (JT,14)
   M#KSTACK(JT,8)=IXM
   DO 3 K=11,LSTACK(JT)
   NY=KSTACK(JT+K)
   PERIM=PERIM+DIST(IABS(NY+LY)+1)
   IW(M) = IW(M) - NY
   M=M+1
   L=NY+IYM
   LYNNY
```

D115

			•				• •						
		-	1	6471	>+4				1 A.				
•		- 1 34 V - 1 7 e M	1	10~~~ 'V/17	. 3)								
		LY=K	STAC	KCJT.	(11)				• •				
		Mars	TACK	(T .	81-1X	M					• •		
		00 4	. K=1	1.15	TACKO	 					•		
•		NVer	стас	47.JT		•••							
		0501	M = 0 5	0744) T 2 T 6	1 4 8 5 4		V)+1)					
		- F G K J	87 6 1 1 - 1 1	17 M 1 4 4		1.00.							
		LW.⊓ M⇔N⊥]⊷30 .4	(តារ		•						,
	÷	t Mast	· •							•			
			¥ / _ + \/ I/										
1 - C		Levi Levi	•1Y0	em A La Z L	• •								
	4	100(DWCL	/*1. 						• . · ·		
		JTEK	STAC	KCJI	13)				•				
		JECJ	T.NE	-02		ר . י		• •			· · · .		
		PERI	M=PE	RIM+	2.	•							
		LARE	A . MA	REA	MAXX	MAXY	IWID	=0				• •	
		JARE	A=IA	REA/	2	•				•		· ,	
•		ASSI	GN 8	TO	LINK						•		
;		DO 5	1=1	+ I F Y	+1								
		IWID	= I W I	n+ID	W(1)			-					
. •	· .	HAXX	= M A X	0 (MA)	XX,IW	ID)					•		·
·		GO T	0 LI	NK									
	8	LARE	A⇒l.A	REA+	IWID	.•		÷.					
		1F(L	AREA	LT.	JAREA) GO	TO 5				•		5
		FRAC	=FLO	AT (L	ARFAm	JAREA)/FL	DATCI	WID)				
· · ·		JDW=	0						· · .				
		IF(I	.NE.	1) J	DW≖ID	W(I)			•	· •			
		XMAR	T=IW	10-1	DW+FR.	AC					•		•
		MART	X=NI	NTCX	MART)						· .		
		1224	GN 5	τO	LINK					•			
•	5	CONT	TNIE	• =			•	•					•
	-	ASSI	GN 9	70	LINK								
		DO 1		1.15	X			2					
• •		MAXY	ΞMAX	OFMA		(1))			•	•			
		60 T	0 17	NK						•			
	0	MAPF		DFA+	10(1)						•		
		TEIM		17	IADEA	ນີ ເດ	70 1	n			·	1	
		EDAC		1. 5. 7. 4. 1. 5. 7 2 M	ADFA.	IARFA	NZEL		46133				
								• / • • •					•
		104-	ALC:	• • •	NU_1U	(1)-1	14 6 7 m	1.5					
•			- N E -		1 NILLS	010 ///////////////////////////////////	W (1						
		TMAK	∓IW		100×2	RAG							
		MAKI	YENI										
		ASSI	GN 1	0 10	LINK								
	10	CUNI	TNUE							•			
		TPER	₽NIN	TCPE	RIMJ								
		CALL	INS	TACK	(NS/L	STACK	(NS)	*1,2,	IPER.	MARTA	#MART1	*****	MAXTJ
		ADIA	= SQR	T(RP	14+FL	OATCI	AREA	})	e Al an an an an			• •	
		PDIA	= P E R	TH+R	PI -					n de la suite Referencia		e.	
		A0P=	ADIA	/PDI	A								
		1F(1	OB (N	0).N	E.NCL) GO	TO 7			<i>e</i>			
	•	NO=N	0+1					•					
•		WRIT	E(2,	25)	NCL								
	25	FORM	AT (†	11,1	7X,1N	UMBER	ÛF	OBJEC	T IN	SCAN	,14)		
		WRIT	F(2,	21)									
	21	FORM	ATC'	01,1	7 x ,'A	REA P	ERIM	CIRC	UL MA	RTIN	DIAM F	ERIM 1	•
	- •	I AR	EA F	ERET	DIAM	MAX	CHOR	D LO	CATIO	N1/17	Х,'ЕQ.	DIAMET	ER!,
		21 mA	RITY	X		Ϋ́		•	X	Y -	X	Y	X*,
		 ξ •	γt	>		-				4			
	٩ ٩	WRIT	F(2+	20)	AD 1 A .	PDIA.	AOP	MART	.YMAR	T,PER	IM.IAR	EA, IFX	,1FY,
			MAY	Y 1 Y	MINII	YMIN	,						
	20	F09M	AT/1	58.2	F6.1.	F7.3.	F7.1	2 F 6 .	1,16.	415.2	16,15)		
	τV	PALL	1)(5) 1)(5)	TKIC	KSTAC	K(1 S -	5).7	LSY			-		
		1241		67 0) PAI:		7010	TINT	SCARN				
		1.1.1	001.	u i.v	, unte	5 FA8			vunr/				
	•	•				•	_ Ð	16	•		•••		

÷

٠

•-

•

• •

. .

AMIN=AMAX1(IARFA*.005,9.) CALL PARTMOM(NT,AMIN) 7 CALL INSTK1(KSTACK(LS,5),3,0) 77 CALL (LSTACK(LS) LS=KSTACK(LS,3) IF(LS.NE.0) GO TO 77 RETURN END

GMENT, LENGTH 764, NAME PARPAR

SUBROUTINE PREADS(IX,IY,MAXX,MAXY) READ(1,1) IX,IY,MAXX,MAXY \$ FORMAT(2014) RETURN END

GMENT, LENGTH 50, NAME PREADS

SUBROUTINE PREAD8(JNT,M) COMPACT CARD VERSION DIMENSION JNT(M),KNT(40) CALL CLEARO(1,JNT) I=0 3 READ(1.1.END=2) KNT 1 FORMAT(40A2) DU 5 K=1.40 5 CALL COPY(2,JNT(I+K),3,KNT(K).1) I=I+40 IF(M=I.GE.40) 40 TO 3 2 M=I RETURN END

GMENT, LENGTH

C

73, NAME PREAD8

	C	SUBROUTINE PLOTST(MAXX,MAXY,SCAF) Opens plotter sets scaling factor and creates Origin Memayo(Mayy,Mayy)
		N=MINO(MAXX,MAXY)
		SCAF=AHIN1(10,/FLOAT(M),6./FLOAT(N))
		CALL UTP2(FLOAT(MAXX)+SCAF,FLOAT(MAXY)+SCAF,0)
•	-	RETURN
		END
GMENT	LENG	TH A7. NAME PLOTST
•		
••••••••••••••••••••••••••••••••••••••		
	·· .	
		SUBRIUTINE PARTPEUTINT/SUAPJ
	U ·	ITANT
		(all thstk1(kstACK(HT,5),3,0))
	•	X=KSTACK(JT,8)+SCAF
		CALL UTP2(X,FLOAT(KSTACK(JT,11))*SCAF,1)
		X=X=SCAF
	1	S DO 1 K=11,LSTACK(JT)
•	٩	CALL UTP2(X,FLOAT(KSTACK(JI/K))*SCAF/2/
	. •	J =K\$ AUK(J +J] V=LC=+AUV(J++J+
•		REGIALROIJTI
		CA(1) HTD2(Y, FLOAT(KSTACK(JT,K=J)) + SCAF(2)
	2	X=X=SCAF
· .		JT#KSTACK(JT.3)
		IF(JT.NE.O) GO TO 3
		CALL UTP2(X+SCAF,FLOAT(KSTACK(NT,11))+SCAF,2)
		CALL INSTRICKSTACK(NT,5),3,NT)
а. - С С С С С С С С		RETURN
		END

GMENT, LENGTH

153, NAME PARTPLOT

									÷
		SUBROL	JTINE	PARTHO	YCJT, AI	MIN)		··· ·	•
C		GENERI	TES /	IND OUT	PUTS I	NVARIANI	MOMENTS D	F CONVEX	COMPONENTS
C		OF BOU	INDARY	IN ST	ACK JT				
		COMMON	1/50/5	;I,C†					
		INTEGR	R SET	(20) 15/10001	VSCA		1000		•
		DENI N Den n	10N N	15(1000) 00).vfA(27201.1	/FAC(20)	FACT(20)		
		TEINCA		()) GO	TO 99				
		P1=3.1	41582	7					
		P12=2+	Pi				•		
		P13=3*	PI						· ·
		FACT (1)=1						. ·
•		DO 98	1=2,2	10 17/7 1).			•		• .
	70	NCILIS	.) 4 8 A G A C A I 1	, (1 = 7) 41					
		NN=5		- * •		4			
		MM#15					. · · · · · · · · · · · · · · · · · · ·	•	•
		IS=100	1				•		
		NTRIAE	S(JT)						· ·
•		CALL I	NSTK1	(KSTACI	((NT,5));3;0)	0.1/		• .
Ç		HIND N		UF PU	LNIS UN	A BOOMDA	KΥ		
	*	NTOTEN	, T() T + I	STACK()	17)-10	•			لار
		NT=KST	ACKIN	17,3)			. •	•	
	•	IF(NT.	NE.0)	GO TO	5			•	
	,	NTRIAB	S (JT)	I					
	. •	CALL I	NSTK1	(KSTAC)	((NT,5)),3,NT)		•	
		JR#1	_ 0		•				••
		SET(1)	EU Inmeet	(NT.10)			M . Y F & C . 1 . F		CK (NT . 8)))
1.1.1		CALL M	LOMINU	CHOMAN	N . MM . X I	FAC,YFAC	FACTS		
		S¤SI							
•		C=C1				•		•	
		AREA=M	IOM (1)				•		
		XBAR=M	IOM (2)		- -		-		
		YBAREM	10M(5)						
		CFasos		A)	•				
		HOM(2)	= MON (2)+CF					
	•	MOM(3)	= MOM (3)*CF			,	•	
		CALL M	TUOMO	(MOM, SE	E T, M M, N	(N,JR)	•		•
		LMIN=1							. •
		SET(1)	21				· ·		
		35(1)= ¥€/1)=	10	•.					
		YS(1)=	NTOT-	1					
•	8	L= (MOD	(JR,2) +2=1)+	ISIGN	(1, JT)			
		CALL M	OMCON	(NS(LM)	(N),XS((LMIN),Y	S(LMIN),MO	M, NN, MM, Y	FAC,L,AMIN
	1	I.IS-LM	INY						
		LMINEL	MIN+L	,=1 ://0			ELATY		
		CALL M	1014 N V	(MUM)NN 109143/1	N F F M A K I	• A C # T P A C	FALIJ		•
		CFESOR	()////////////////////////////////////						
		X=MOM(2)+CF	-XBAR					
		Y=MOM (3)+CF	-YBAR					
		MOM(2)	=X + C +	Y * S			· · · · · · · · · · · · · · · · · · ·		
		MOM(3)	=Y+C-	X+S		. 			
		MOM(5)	AHOD	(MOM(5)	**************************************	(+PI3;P1	2)-PI		
		TEVI P		(MUM) SE	CE⊅MM∌N S	EN F J K J		i	· ·
		JRE.DA	जन्म 1	<u>an in c</u>	,		•		
		SET (JR)≖t			а		•	
			-						

•

•

.

.

· •

GO TO 8 6 IF(LMIN.EQ.0) RETURN 7 SET(JR)=SET(JR)-1 IF(SET(JR).GT.0) GO TO 8 JR=JR-1 GO TO 7 END

GMENT, LENGTH 352. NAME PARTMOM

Ĉ

SUBROUTINE MOMSET(JT, JJ, NHM, MOM, NN, MM, YFAC, MQ, XA) FINDS HOMENTS OF NNM POINT ARC FROM XA, YA TO XF, YF REAL MOM(MM), YFAC(NN) CALL CLEARO(1,MOM) NM=NNM M=MQ JeJJ X=XA NTEJT VA=KSTACK(NT+J+M) LEN=LSTACK(NT) 2 A=M D=SIGN(0.5,A) JE=11+MAX0(0,M)+(LEN-11) ND=MINO(NM, IABS(J-JE)) DO 1 1=1,ND J¤J+M CALL SMOMS (MOM, NN, MM, YFAC, X+D, X+D, FLOAT (KSTACK (NT, J))) 5 X=X+A NM=NM=ND IF(NM_EQ_0) GO TO 3 ManM X=X-A NTEKSTACK (NT.3) LEN=LSTACK(NT) J=10-MING(0,M)+(LEN-9) GO TO 2 3 XF=X-A RETURN END

GMENT, LENGTH 196, NAME

MOMSET

- D120 -

SUBROUTINE SHOMS (MOM, NN, MM, YFAC, XA, XC, Y) REAL MOM(MM), YFAC(NN) INTEGER P.Q Ĉ FINDS SIMPLE MOMENTS FROM UPPER OR LOWER BOUNDARY POINT XB#XA XD=XC LS=1 YP=1. DO 2 J=1.NN YP=YP+Y 2 YFAC(J)=VP/J DO 3 P=1,NN X={XB=XD}/P L.LS=LS+P-1 DO 4 Q=1,NN+1-P MOM(L)=MOM(L)+X+YFAC(Q) 4 L=L+Q+P XB#XB+XA 3 XD=XD*XC RETURN END GMENT, LENGTH 131. NAME SMONS SUBROUTINE MOMOUT (MOM, SET, MM, NN, JR) INTEGER SET(JR) REAL MOM (MM) J=O WR1TE(2,4) 4 FORMAT(! !) WRITE(2:1) SET 1 FORMAT(20X, 'COMPONENT', 13, 20(',', 11)) WRITE(2,5) 5 FORMAT(1+1,43X, MOMENTS OF INCREASING 1/20X/FORDER DOWNER 1 WARD AND INCREASING POWER OF Y ACROSS) DO 2 INT INN WRITE(2,3) (MOM(K),K=J+1,J+I) J=J+T 3 FORMAT(15X,8G12.4) 2 CONTINUE RETURN END

GMENT, LENGTH

85, NAME MOMOUT

- D121 ·

SUBROUTINE MCMINV(MOM;NN;MM;XFAC;YFAC;FACT) C C MOM CONTAINS SIMPLE MOMENTS, FACT(1) IS FACTORIAL 1-1 ALL OTHER ARRAYS ARE WORKING SPACE C C C OUTPUT IS INVARIANT MOMENTS BUT MOM(1) IS AREA. MOM(2) AND MOM(3) ARE SCALED CENTROID COORDINATES AND MON(5) IS THETA IN RANGE +PI TO -PI COMMON/SC/S,C DIMENSION CFAC(20), SFAC(20) REAL MOM(MM), XFAC(NN), YFAC(NN), FACT(NN) INTEGER P.Q AREA=MOM(1) SS,S=1./SORT(AREA) XP=S/AREA XBAR=MOM(2) + XP YBAR=MOM(3) + XP T,XP,YP=1. MAKE SIZE INVARIANT AND SET UP POSITION INVARIANCE FACTORS Ĉ L=Ô DO 1 1=1.NN S=S+SS XFAC(1)=XP YFAC(1)=YP XP=-XP+XBAR/T YP==YP*YBAR/T T=T+1. 1=1+1-1 DO 2 J=1.1 2 MOM(L+J)=MOM(L+J)*S/(FACT(I=J+1)*FACT(J)) CONTINUE MAKE POSITION INVARIANT С LL=NN+4 L=MM+1 DO 3 1=1,NN K¤LL ÷., LL=LL-1 DO 3 P=1,LL Q=K-P L,15,11=t-1 IKELL S¤Ô. DO 4 N=1,P IJHIK SSHO. DO 5 M=1,Q SS=SS+YFAC(M) + MOM(II) 11=I1-IJ 5 1J=1J-1 S=S+SS+XFAC(N) IK#IK-1 4 IS, IT=IS-IK 3 MOH(L)=S+FACT(p)+FACT(q) FIND THETA AND ITS SINE AND COSINE ¢ C2=0,5+0,5/SQRT(1.+(2.+MOM(5)/(MOM(4)-MOM(6)))++2) C = SQRT(C2)S=SQRT(1_-C2) AA=(C2+2.-1.)+MOM(5) BB#(MOM(6)-MOM(4))+S+C IF (ABS(BB+AA), LT, ABS(BB-AA)) GO TO 12 S==S 13 T=S S=C - D122 -C=+T

12	AM30-40M/71+C++3+	(3 + (MOM(8)+C+MOM(2+(2+2+(01)M0M+3+(2+(9)	
	AM03==M0M(7)+S++3	+(3 + (MOM(8)+S=MOM	1(9)+C)+S+MOM(10)+C+C)+C	
	TELAMON IT O OR	AM30 11.0.1 GO TO	13	
	THETA-ATAN2(S.C)			
	MAKE BOTATION			
	1=0			
	T.SF. (F=1		· . ·	
	DO 7 THI NN			
	SFAC(1)=SF			
	CFAC(1) = CF		•	
	DO 8 P=1,I			
	Q=I-p+1			
	SUM=0		. '	
	\$1=1.			
	DO 9 N=1.P	•	•	
	\$\$=0.			
•	K=L+N+Q	• •		
	DO 10 M=1,Q			
10	SS=SS+MOM(K-M)+SF/	AC(M)+CFAC(Q-M+1)		
	SUM=SUM+SS+SFAC(N))+CFAC(P-N+1)+SI		
9	\$1==31			
8	YFAC(P)=SUM+FACT(P) + FACT(Q)		
	DO 11 M=1,I		· · · ·	
11	MOM(L+M)=YFAC(M)		· · · · · · · · · · · · · · · · · · ·	4
	SF==SF*S/T			•
	CF#CF+C/T			
	T=T+1.			,
7	L=L+t			
	MOM(1)=AREA			
	MUM(2)=XBAR	•	•	•
	MOM(3)=YBAR			
	MOM(5)=THETA			
	KETURN	•		
	END			

IGMENT, LENGTH

C

607. NAME MOMINV

- D123 -

		SUBROUTINE MOMCON(NS,XS,YS,MOM,NN,MM,YFAC,LN,AMIN,IS)
C		FIND SIMPLE MOMENTS OF CONVEX HULL AND MARK CONCAVE SETS
		REAL MOM(MM)
		COMMON/BD/LB,LD,HC
		DIMENSION XS(IS), YS(IS), NS(IS), YFAC(NN)
		JT. NT = $NS(1)$
		$J_{\pm}J_{\pm}$
		M. ML=KSTACK(NT, 7)
		HCmLN
		CALL CONP(XS, YS, NS, JT, J, NINT(YS(1))+1, HC, IS, KN, H)
		CALL CLEARD(1,MOM)
C		CALCHLATE FIRST THE MOMENTS TO CLOSE BOUNDARY
•		
		$\mathbf{Y}\mathbf{P}\mathbf{a}\mathbf{Y}\mathbf{S}(1)$
•		YFRXC(KN)
		YF=YS(KN)
		ANEES
		A=0
		D#m0.5+H+HC
•		TE(NINT(XE+XP) EQ.0) GO TO 1
		AESTGN(1 .XF-XP)
		ANI = ARS (YFeXP)
		AMEAM/ANI
		TECHE, FO 1) CALL SMOMS(MOM,NN,MM,YEAC,XP+D,XP=D,YP)
		IF(LD_FQ_NS(KN)) ANL=ANL=1.
		TE(NINT(ANI), EQ. 0) GO TO 2
	4	CONTINUE
	•	DO 9 TET.NTNT(ANL)
		YP=YD+AM
		XPeXD+1
	0	CALL SHOWS (MOM, NN, MM, YEAC, XP+D, XP=D, YP)
	ź	CONTINUE
Ĉ	-	CALCULATE MOMENTS FOR BOUNDARY ITSELF
•		CALL MCONSET (NS, XS, YS, MOM, YFAC, NN, MM, LN, KN, NT, JL, ML, AMIN, IS)
	•	RFTURN
		END

GMENT, LENGTH 253, NAME

NAME MOMCON

- D124 -

C JLLLD, JUNCTIC JUNC JUNC JUNC JUNC JUNC JUNC JUNC JUN	r		SUBROUTINE CONP(XX, YY, LL, JT, J, NM, BMU, JTOT, NP, ML) SELECT CONVEX POINTS FROM NM POINTS STARTING AT	(NT,J+M)
DIAMON BD/ ED/ DIAMON BD/ ED/ DIAMON BD/ ED/ H=ML AMU=BMU+M NP.L=0 LM/N.LMAX=1 LEN=LSTACK(JT) X=KSTACK(JT,B)+J=11+M XMIN,XMAX=X C UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+AMXO(D,M)+(LEN=11) NDEMINO(NM,IABS(J=JE)) A=M DO 1 I=1.ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=Y+A NM=NM=ND X=A IF(NINT(YMAX=X),GE.O) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(YMIN=X).LE.O) GO TO 31 XMIN=X LMAX=L 31 CONTINUE IF(NM,EQ.O) GO TO 5 JT=KSTACK(JT.3) LEN=LS	L		COMMON/DN/ID.ID	
MHL MHL AMUBBNU+H NP,L=0 LMIN.LMAX=1 LENEISTACK(JT,B)+J=11+M XMIN.KMAX=X C UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+MAY0(0,H)*(LEN=11) NDEMINO(NM.IABS(J=JE)) A=M D0 1 1=1.ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=1 1 X=X+A NM=NM=ND M==M X=X=A IF(NINT(YMAX=X).GE.0) GO TO 30 XMAX=X LMAX=L S0 IF(NINT(YMAX=X).GE.0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JTAKSTACK(JT) J=10-MINO(IMAX,L(H)N) LD=LMAX+LMIN=LEN C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY.LL.JTOT.LB.LD.NP.AMU.BMU) LB=L8+1 C APPLY T1 AND T2 TO SEGMENT CD S2 CALLCONTRACT(XX,YY.LL.JTOT.LD+1.L.NP.AMU.BMU) 12 CONTINUE IF(FUR.EQ.0) GO TO 12 CALLCONTRACT(XX,YY.LL.JTOT.LD+1.L.NP.AMU.BMU) 12 CONTINUE IF(TH.EQ.ND COND END END			NIMENCION VY(ITOT).VY(ITOT).II((TOT)	
AMUE BMUJ*M NP.L=0 LMN.LMAX=1 LENELSTACK(JT) X=KSTACK(JT,B)+J=11+M XMIN.XMAX=X C UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+AKO(0,H)*(LEN=11) NDEMINO(NM,IABS(J=JE)) A=M D0 1 1=1+ND J=J+M L=L=1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A NM=NM-ND N==M X=X-A IF(NINT(XMAX=X),GE,0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMAX=X),GE,0) GO TO 31 XMIN=X LMAX=L 31 CONTINUE IF(NM_EQ_0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT,3) LEN=LSTACK(JT,3) LEN=LSTACK(JT,1) J=10=MINO(0,MAX_LLIN) LD=LMAX+LMIN=LB C ALLCONTRACT(XX,YY,LLJTOT,L,L,NP,AMU,BMU) LD=LMAX+LMIN=LB C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LLJTOT,LD+1,L,NP,AMU,BMU) LD=LMAXCT(XX,YY,LLJTOT,LD+1,L,NP,AMU,BMU) 15 CONTINUE IF(LD_EQ_1) GO TO 32 C ALLCONTRACT(XX,YY,LLJTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE FIFURM END LENCTU 407 MAME COND			Maki	
<pre>NPUBBOOK NPLEO LMIN.LMAX=1 LENESTACK(JT) X=KSTACK(JT,B)J=11+H XHIN.XHAX=X UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+MAYO(0,M)*(LEN-11) ND=MINO(MM,IABS(J-JE)) A=M DO 1 I=1,ND J=J+M L=L+1 Y(L)=KSTACK(JT,J) XX(L)=x LL(L)=L 1 X=X+A NM=NM=ND Y H==M X=X-A IF(WINT(XMAX=X),GE,0) GO TO 30 XMAXex LMAX=L UMAX=L 30 IF(WINT(XMIN=X),LE,0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM,EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTA</pre>			441-0411-13	
LMTN.LMAX=1 LEN=LSTACK(JT) X=KSTACK(JT,B)JJ=11+M XHIN.XMAX=X UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+MAYO(0,M)+(LEN=11) ND=MINO(NM.JABS(J-JE)) A=M DO 1 I=1.ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(1)=L 1 X=X+A NM=NM X=X-A IF(NINT(XMAX=X).GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NYNT(XMIN=X).LE.0) GO TO 31 XMAX=X LMAX=L 31 CONTINUE IF(MLE0.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=0-MINO(0,M)+(LEN=9) GO TO 4 5 LB=MINO(IMAX.LMIN) LD=LMAX+LMIN=LB C APPLY T1 AND T2 TO SEGMENT AB CALCONTRACT(XX,YY.LLJTOT.I.LB.NP,AMU.BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY.LLJTOT.LD+1,L,NP,AMU.BMU) 12 CONTINUE IF(LD.CO.LG 0) GO TO 5/2 CALLCONTRACT(XX,YY.LLJTOT.LD+1,L,NP,AMU.BMU) 12 CONTINUE IF(LD.CONTRACT(XX,YY.LLJTOT.LD+1,L,NP,AMU.BMU) 12 CONTINUE RETURN END			ND_L=AmU=BmU=M	
LENILSTACK(JT) LENILSTACK(JT, 8)-J-11+M XHIN,XHAX=X C UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+MAYO(0,M)+(LEN-11) ND=MINO(NM,IABS(J-JE)) A=M DO 1 I=1,ND J=J+M L=L-1 YY(L)=KSTACK(JT,J) XX(L)=X L(L)=L 1 X=X+A NM=NM-ND M=-M X=X-A IF(NINT(XMAX-X).GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMAX-X).LE.0) GO TO 31 XMIM=X LMIM=L 31 CONTINUE IF(NINT(XMIN-X).LE.0) GO TO 31 XMIM=X LMIM=L 31 CONTINUE IF(NINT(XMIN-X).LE.0) GO TO 31 ZMIM=X LMIM=L 32 CONTINUE IF(MAX-LMIN-LB C TEST IF AB EXISTS IF(LS:TACK(JT) LB=LSTACK(JT) J=10-MINO(0,M)+(LEN-9) GO TO 4 5 LB=MINO(IMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LS:C,1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALCONTRACT(XX,YY,LL,JTOT,ILB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE IF(LS:C, L) GO TO 12 C ALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE FND			NP)L=0 INTN INAV+4	
LEMELSIALKUJ, J, J-11+M X=KTACK(JJ, B)+J-11+M XHIW, XHAX=X C UNPACK THE POINTS AND LOCATE B AND D 4 JE=11+MAXO(0,M)+(LEN=11) ND=MINO(NM, IABS(J-JE)) A=M DO 1 I=1,ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A NM=NM-ND M=-M X=X-A IF(NINT(XMIN-X), GE,0) GO TO 30 XMAX=X LMAX=L IG T(NINT(XMIN-X), LE,0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(MN.EQ.0) GO TO 5 JT=KSTACK(JT,3) LFN=LSTACK(JT,3) LSTACK(JT,3)				
<pre>X=KSTACK(JT,0)+J=1+W XHN.XMA*xX C UMPACK THE POINTS AND LOCATE B AND D 4 JE=11+MAX0(0,M)+(LEN=11) NDDMINO(AM,IABS(J=JE)) A=M DO 1 I=1,ND J=J+M L=L+1 Y(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A MM=NM=ND Y==M X=X+A IF(NINT(XMAX=X).GE.0) GO TO 30 XMAX=X LMAX=L SO TF(NINT(XMIN=X).LE.0) GO TO 31 XMIN=X LMAX=L SI CONTINUE IF(NM,EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT,3) LSTACK</pre>			LENELSTACK(JI) Menoplakking BV, 1944.4M	
$ \begin{array}{llllllllllllllllllllllllllllllllllll$				
C UNDARK THE POINTS AND COCATE B AND C 4 $J = 11 + MAO (O(M) + (LEN-11)$ NDEMINO(NM, IABS(J-JE)) A=M DO 1 I=1,ND J=J+M L=L+1 Y (L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A MM=NM-ND M==M X=X-A IF(NINT(XMAX-X),GE.O) GO TO 30 XMAX=X LMAX=L SO IF(NINT(XMIN-X),LE.O) GO TO 31 XMIN=X LMAX=L SO IF(NM,EQ.O) GO TO 5 JFEKSTACK(JT,3) LFN=LSTACK(JT) J=10-MINO(0,M)+(LEN-9) GO TO 4 S LB=MINO(LMAX,LIAIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LS,EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,LB,NP,AMU,BMU) LB=LSA1 C APPLY T1 AND T2 TO SEGMENT CD S2 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END	•	•	XMIN, XMAXAX	$2 - \epsilon$
<pre>A JE 11 + MAYO(U, M) + (LEM-11) No MINO(NM, IABS(J-JE)) A=M DO 1 I=1.ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=Z LL(L)=L 1 X=X+A NM=NM-ND X=X+A NH=M X=X-A IF(NINT(YMAX=X),GE.O) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(YMAX=X),LE.O) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM,EQ.O) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT</pre>	C		UNPACK THE PUINTS AND LUCHIC D AND D	
MDEMINO(NM, LABS(J=JEJ) A=M DO 1 I=1,ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A NM=NM=ND X=X-A IF(NINT(XMAX=X),GE,O) GO TO 30 XMAX=X LABASEX SO TO 4 S LB=MINO(LMAX,L(1N) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB,C) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) LD=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 14 END LENCYU		. 4	JE#11+MAYO(U,M)*(LEN#11)	
A=M D0 1 1=1,ND J=J+M L=(+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A NM=NM=ND Y=A X=X-A IF(NINT(XMAX=X).GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMIN=X).LE.0) GO TO 31 XMIN=X LMAX=L 31 CONTINUE IF(NM_EQ.0) GO TO 5 JTEKSTACK(JT,3) LEN=LSTACK(JT,3) LSTACK(JT,		. •	NDEMINO(NM, LADS(J#JEJ)	
DO 1 I=1,ND J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A Nm=NM X=X-A IF(NINT(XMAX=X),GE.O) GO TO 30 XMAX=X LMAX=L MAX=L SO IF(NINT(XMIN=X),LE.O) GO TO 31 XMIN=X LMAX=L SI CONTINUE IF(NM.EQ.O) GO TO 5 JT=KSTACK(JT,3) LFM=LSTACK(JT) J=10=MINO(0,M)+(LEN=9) GO TO 4 5 LB=MINO(1MAX,LKIN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB_EO.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END				
<pre>J=J+M L=L+1 YY(L)=KSTACK(JT,J) XX(L)=x LL(L)=L 1 X=X+A NM=NM=ND %==M X=x-A IF(NINT(XMAX=X),GE,O) GO TO 30 XMAX=x LMAX=L 30 IF(NINT(XMIN=X),LE.O) GO TO 31 XMIN=x LMIN=L 31 CONTINUE IF(NM_EQ_O) GO TO 5 JT=KSTACK(JT,3) LEN=LS</pre>			DO 1 I=1,ND	
L=L+1 YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A Mm=Mm-ND X=X-A IF(NINT(XMAX=X),GE,O) GO TO 30 XMAX=X LMAX=L 30 TF(NINT(XMIN=X),LE.O) GO TO 31 XMIN=X LMMN=L 31 CONTINUE IF(NN,EQ,O) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10=MINO(0,M)+(LEN=9) GO TO 4 5 LB=MINO(LMAX,L(4IN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB_EQ,1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY)LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY)LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END	·		J = J + M	•
YY(L)=KSTACK(JT,J) XX(L)=X LL(L)=L 1 X=X+A NM=NM=ND M==M X=X-A IF(NINT(XMAX=X).GE.0) GO TO 30 XMAXeX LMAX=L 30 IF(NNT(XMIN=X).LE.0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NN.EQ.0) GO TO 5 JT=KSTACK(JT.3) LEN=LSTACK(JT.3) LEN=LSTACK(JT.3) LEN=LSTACK(JT.3) LEN=LSTACK(JT.3) LB=LSTACK(JT.3) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY.LL.JTOT.1.LB.NP.AMU.BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY.LL.JTOT.LB.LD.NP.=AMU.BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY.LL.JTOT.LD+1.L.NP.AMU.BMU) 12 CONTINUE RETURN END	-			
XX(L)=X LL(L)=L 1 X=X+A NM=NM-ND X=X-A IF(NINT(XMAX-X),GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMIN-X),LE.0) GO TO 31 XMJN=X LMIN=L 31 CONTINUE IF(NM,EQ.0) GO TO 5 JT=KSTACK(JT,3) LFN=LSTACK(JT) J=10=MIN0(0,M)+(LEN=9) GO TO 4 5 LB=MIN0(IMAX,L(4IN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB_EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LB_EQ.1) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END		•	YY(L)=KSTACK(JT/J)	
LL(L)=L 1 X=X+A NM=NM=ND N=M X=X-A IF(NINT(XMAX=X),GE.O) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMIN=X).LE.O) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM.EQ.O) GO TO 5 JT=KSTACK(JT.3) LFN=LSTACK(JT.3) LF		•	XX(L)=X	
<pre>1 X=X+A NH=NH=ND H==H X=X-A IF(NINT(XMAX-X).GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMIN-X).LE.0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT.3) LFN=LSTACK(JT) J=10-MIN0(0,M)+(LEN-9) GO TO 4 5 LB=MIN0(1MAX,L(4IN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENGTU</pre>				
<pre>NM=NM-ND M=-M X=X-A IF(NINT(XMAX-X),GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMIN-X),LE.0) GO TO 31 XMJN=X LMIN=L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT,3) LEN=LSTACK(JT,3) LEN=LSTACK(JT) J=10-MINO(0,M)+(LEN-9) GO TO 4 5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END</pre>		- 1	X=X+A	
<pre>M==M X=X-A IF(NINT(XMAX-X),GE.0) GO TO 30 XMAX=X LMAX=L 30 IF(NINT(XMIN-X),LE.0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10=MINO(0,M)+(LEN=9) GO TO 4 5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT.1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) END</pre>			NM=NM-ND /	
<pre>X=*-A IF (NINT (XMAX-X), GE, 0) GO TO 30 XMAX=X LMAX=L 30 IF (NINT (XMIN-X), LE.0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF (NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT,3) LEN=LSTACK(JT,3) LEN=LSTACK(JT,3) GO TO 4 5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF (LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT.1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF (LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LEWCYM</pre>		:	M¤≈M	
<pre>IF (NINT(XMAX=X),GE,O) GO TO 30 XMAX=X LMAX=L 30 IF (NINT(XMIN-X),LE.O) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF (NM.EQ.O) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10-MIN0(0,M)+(LEN-9) GO TO 4 5 LB=MIN0(LMAX,L(dIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF (LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF (LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END</pre>			Х≖Х⊶А	
<pre>XMAX=X LMAX=L 30 IF(NINT(XMIN-X).LE.0) GO TO 31 XMIN=X LMIN=X LMIN=X IF(NN=Eq_0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10-MINO(0,M)*(LEN-9) GO TO 4 5 LB=MINO(IMAX,L(4IN) LD=LMAX+I,MIN-LB C TEST IF AB EXISTS IF(LB_EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT.1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END</pre>		•	IF(NINT(XMAX-X),GE.O) GO TO 30	
LMAX=L 30 IF (NINT(XMIN-X).LE.O) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF (NM.EQ.O) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10-MINO(O,M)+(LEN-9) GO TO 4 5 LB=MINO(LMAX,L(4IN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF (LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF (LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END	•	·	XMAX=X	
<pre>30 IF(NINT(XMIN-X), LE.0) GO TO 31 XMIN=X LMIN=L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10-MINO(0,M)*(LEN=9) GO TO 4 5 LB=MINO(IMAX,LHIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END 4 ENCTH</pre>			LMAX=L	
<pre>XMJN=X LM1N=L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10=MINO(0,M)*(LEN=9) GO TO 4 5 LB=MINO(UMAX,LHIN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END</pre>		30	IF(NINT(XMIN-X).LE.O) GO TO 31	•
LMIN≈L 31 CONTINUE IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10-MINO(0,M)+(LEN-9) GO TO 4 5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END			XMIN=X	• •
<pre>31 CONTINUE IF (NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10=MINO(0,M)+(LEN=9) GO TO 4 5 LB=MINO(1MAX,LWIN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF (LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF (LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END</pre>			LMINAL	
<pre>IF(NM.EQ.0) GO TO 5 JT=KSTACK(JT,3) LEN=LSTACK(JT) J=10-MINO(0,M)+(LEN-9) GO TO 4 5 LB=MINO(IMAX,L4IN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END AENCYM</pre>		31	CONTINUE	
JT=KSTACK(JT,3) LFN=LSTACK(JT) J=10=MINO(0,M)+(LEN=9) GO TO 4 5 LB=MINO(1MAX,LHIN) LD=LMAX+LMIN=LB C TEST IF AB EXISTS IF(LB_EQ_1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END 4 ENCYM			IF(NM.EQ.O) GO TO 5	
LENELSTACK(JT) J=10-MINO(0,M)+(LEN-9) GO TO 4 5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT.1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END			JT=KSTACK(JT+3)	•
J=10-MINO(0,M)+(LEN-9) GO TO 4 S LB=MINO(LMAX,L41N) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB_EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD S2 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END		× 1.	LEN=LSTACK(JT)	
GO TO 4 5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT.1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END			J=10-M1N0(0,M)+(LEN-9)	
<pre>5 LB=MINO(LMAX,LMIN) LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCTH TOT, NAME CONP.</pre>			GO TO 4	
LD=LMAX+LMIN-LB C TEST IF AB EXISTS IF(LB.EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCTH JOZ, NAME CONP		5	LB=MINO(LMAX,LUIN)	
C TEST IF AB EXISTS IF (LB_EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF (LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCYH JOZ, NAME CONP			LD=LMAX+LMIN-LB	•
IF(LB_EQ.1) GO TO 32 C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCTH JOZ, NAME CONP	C		TEST IF AB EXISTS	· .
C APPLY T1 AND T2 TO SEGMENT AB CALLCONTRACT (XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT (XX,YY,LL,JTOT,LB,LD,NP,-AMU,BMU) C TEST IF EF EXISTS IF(LD,EQ,L) GO TO 12 CALLCONTRACT (XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCTH JOZ, NAME CONP			IF(LB_EQ_1) GO TO 32	
CALLCONTRACT(XX,YY,LL,JTOT,1,LB,NP,AMU,BMU) LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCTH JOZ, NAME CONP	Ĉ		APPLY T1 AND T2 TO SEGMENT AB	
LB=LB+1 C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENCTH JOZ, NAME CONP	•		CALLCONTRACT (XX, YY, LL, JTOT, 1, LB, NP, AMU, BMU)	
C APPLY T1 AND T2 TO SEGMENT CD 32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END AFMCTH JOZ, NAME CONP			LBala+1	
32 CALLCONTRACT(XX,YY,LL,JTOT,LB,LD,NP,=AMU,BMU) C TEST IF EF EXISTS IF(LD.EQ.L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENGTH JOZ, NAME CONP	C		APPLY T1 AND T2 TO SEGMENT CD	•
C TEST IF EF EXISTS IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END LENGTH JOZ, NAME CONP	•	32	CALLCONTPACT (XY, YY, LL, JTOT, LB, LD, NP, = AMU, BMU)	
IF(LD_EQ_L) GO TO 12 CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END	Ĉ.	JL	TEST IF EF EXISTS	
CALLCONTRACT(XX,YY,LL,JTOT,LD+1,L,NP,AMU,BMU) 12 CONTINUE RETURN END 15 CONP	•		TE(10 EQ 1) GO TO 12	• • •
12 CONTINUE RETURN END			CALLCONTRACT (XX, YY, LL, JTOT, LD+1, L, NP, AMU, BHU)	
RETURN END		4 2	CONTINUE	
END ENCTH JOZ NAME CONP	• .	16	DETION	•
LENGTH JOZ NAME COND			FNN	· · · · ·
A FNGTH JOZ NAME COND				
SENCTH TOT NAME COND				· · · · ·
		-	A TOT NAME COND	

:

GMENT

- D125 -

	SUBROUTINE CONTRACT(XX,YY,LL,JTOT,LS,LF,NP,AMU,BMU)
c .	CONTRACT POINTS FROM LS TO LF USING T1 AND T2
•	DIMENSION XX(JTOT), YY(JTOT), LL(JTOT)
	$\mathbf{TF}(1) = \mathbf{F}(2) = \mathbf{G}(1)$
	$\mathbf{M} = \mathbf{C} \mathbf{T} \mathbf{C} \mathbf{M} \mathbf{C} \mathbf{T} \mathbf{T} \mathbf{C} \mathbf{T} \mathbf{T} \mathbf{T} \mathbf{T} \mathbf{T} \mathbf{T} \mathbf{T} T$
	TD=NTNN(1,,,N(1))=N((+(1+)))
	DU C IFLS/LF TD-/NTNT/VV/TNA_TDN+MI
	$\mathbf{v} = (\mathbf{v} \in (\mathbf{A} \setminus (\mathbf{A})) = (\mathbf{a} \in ($
	$ = \left\{ \frac{1}{2} + \left\{ \frac{1}{2} + \frac{1}{$
	A ANT WONDOWN AN IN A ID A ITATA
	1 CALL MOVEP(XX, TTILL/L/JP/TIJIUI)
	IF(JP.1.E.2) 60 10 1
• .	KP,NP=NP+Z
•	DO 3 I=NP+1, JP
I	6 IF(NINT(:YY(NP=4)+(XX(I)=XX(NP))+YY(I)+(XX(NP)=XX(NP=4))=YY(NP)
	1(XX(I)=XX(NP+1)))+BMU))16;0;0
	NP=NP-1
	IF(NP.GE.KP) GO TO 6
٩.	6 NP=NP+1
	3 CALL MOVEP(XX, YY/LL, I, NP, 1, JTOT)
	RETURN
	1 NP=NP+JP
	CALL MOVEP(XX, YY, LL, LS, NP-JP+1, JP, JTOT)
	RETURN
	END CONTRACTOR OF A CONTRACTOR OF
LENG	TH 286, NAME CONTRACT

SUBROUTINE MOVEP(XX,YY,LL,I,J,K,L) DIMENSION XX(L),YY(L),LL(L) IF(I.E0.J) RETURN CALL PMOVE(XX(I),XX(J),K+K) CALL PMOVE(YY(I),YY(J),K+K) CALL PMOVE(LL(I),LL(J),K) RETURN END

GMENT, LENGTH

GMENT,

97, NAME MOVEP

```
SUBROUTINE MCONSET(NS,XS,YS,MOM,YFAC,NN,MM,L,K,NT,J,
  4 M.AMIN.TS)
   COMMON/BD/LB,LD,HC
   FINDS SIMPLE MOMENTS OF THE CONVEX HULL OF THE NM
   POINTS STARTING AT (NT, J+M) AND SUPPLIES DETAILS OF
   ANY CONCAVITIES
   DIMENSION XS(IS), YS(IS), NS(IS), YFAC(NN)
   REAL MOM(MM)
   HC=L
   L=O
   X=XS(1)
   Y=YS(1)
   N=NS(1)
   LEN=LSTACK(NT)
   ÐN=HC+Q.5+M
   DO 10 1=2,K
   XL¤X
   YA,YL=Y
   NLAN
   X=XS(1)
   Y=YS(I)
   N=NS(I)
   ¥=W
   D=DN
   IF(NL_LT_LB_OR_NL_GT_LD) D=+D
   AR=0.
   JLEJ
   NTL=NT
   ML=M
   JE.JEL=11+MAX0(0,M)+(LEN=11)
   IF(X.NE.XL) GO TO 21
   AMM, AM=0_
   GO TO 22
21 AMM, AM=(Y-YL)/ABS(X-XL)
22 NM, NML=N=NL
   FIND AREA BETWEEN LINE OF CONVEX HULL AND BOUNDARY
20 ND=MINO(NM, IABS(J-JE))
   DO 12 II=1,ND
   J=J+M
   AR¤AR+VL⇔KSTACK(NT,J)
12 YL=YL+AM
   IF(J.NE.JE) GO TO 23
   AR=-AR
   M≈⊨M
   AMawaM
   YL=YL+AM
   NT#KSTACK(NT+3)
   IF(NT.EQ.0) GO TO 23
   LEN=LSTACK(NT)
   J=10-MINO(0,H)+(LEN-9)
   JE=11+MAX0(0,M)+(LEN-11)
   NM=NM=ND
   IF(NM.NE.0) GO TO 20
   IF AREA LT AMIN ASSUME CONVEX HULL RUNS ALONG BOUNDARY
   ELSE USE CONVEX HULL LINE AND MARK CONCAVE SET
23 IF( ABS(AR).LT.AMIN) GO TO 13
   1=1+1
   NS(L)=NTL
   XS(L)=JL
   YS(L)=NML
   DO 14 II=1,IABS(NINT(X-XL))
   CALL SMOMS(MOM, NN, MM, YFAC, XL+D, XL-D, YA)
                           D127
```

C C C

C

Ĉ

	XL=XL+A
14	YA=YA+AMM
	GO TO 10
13	NM=N=ti
	DHKG+Mi+A.5
	TEVEN HE NOVEN AND I FOR KANDININIVELSIN FOR ON NMENKAS
	$= \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_$
• •	IF(LB.VE.1.AND.I.EV.2.AND.NINI(HC+1).EV.U) NII=2
16	ND=MINO(NM,IABS(JL=JEL))
•	IF(NII.EO.2) JL=JL+ML
	IF(ND_LT_NII) GO TO 17
	DO 15 II=NII,ND
	JL#JL+ML ·
	CALL SMOMS (MOM, NN, MM, VFAC, VL+D, XL-D, FLOAT (KSTACK (NTL, JL)))
15	
	YinYinA
47	
• •	
	JE(NM.EQ.U) GU IV IV
	NTL=KSTACK(NTL,S)
	LEN=I STACK (NTL)
	ML=+HL
	D==D
	Ar-A
	JL=10-MINO(0,ML)+(LEN-S)
	JEL=11+MAXO(0,ML)+(LEN=11)
	NTT=1
	60 YO 46
• •	60 TO
10	
	KC I UKN Pin
	END

GMENT, LENGTH 576, NAME MCONSET

- D128 -

Bibliography.

Rosenfeld. A.

	Academic Press (1969).
2.	Deutsch. S. A note on some statistics
·	concerning typewritten or printed material
	IRE Trans. Inf. Theory <u>IT3</u> , 147-148(1957).
3.	Simpson. R.S. et.Al. Compendium of redundancy
	removal processes, I.E.E.E. Trans. Aerospace
	and Electronic Systems <u>AES2</u> , 471-474(1966).
4.	Huffmann. D.A., A method for the construction
• .	of minimum redundancy codes. Proc. IRE. 40 pl098
	-1101 (1952).
5.	Schwartz. J.W. and Barker. R.C., Bit plane
÷ *	encoding: a technique for source encoding.
	I.E.E.E. Trans., Aerospace and Electronic
	Systems <u>AES2</u> , 385-392 (1966).
6.	Andrews. H., Transformations for the processing
	and coding of Images. J.P.L. Report (1969).
:	Edinburgh Univ. Press. pl41-150 (1971).
	W.K. Pratt. Hadamard transform image coding.
	Procc. I.E.E.E. <u>57</u> p 58-68 (1969).
7.	Ahmed. N. and Rao K.R., Additional properties
· · · · ·	of the complex BIFORE transform I.E.E.E. Trans.,
	Audio <u>AU19</u> p 253-254 (1971).
8.	Emden van. M.H. Optimal Data Compression.
те.	Machine Intelligence <u>6</u> Meltzer - Michie.
9.	Andrews. H.C., Multidimensional rotations in
	feature selection. I.E.E.E. Trans, Computers.
	<u>C20</u> p 1045-1051 (1971).

Picture Processing by computer.

1.

 Roberts. L.G., Picture coding using pseudo- random noise IRE. Trans. Inf. Theory. <u>IT8</u> p145-154 (1962). Gordon. R Herman., Reconstruction of pictures from their projections. C.A.C.M. <u>14</u> p 759-768 (1971). Fratt. W.K., Hadamard Transform Image Goding. Proc. IEEE <u>57</u> p58-68 (1969). Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>G18</u> p 733-740 (1969). Preston K. Jnr., The Callscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Go. (1968). Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report J2-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>021</u> p 179-186 (1972). 		
 Roberts. L.G., Picture coding using pseudo- random noise IRE. Trans. Inf. Theory. <u>IT8</u> p145-154 (1962). Gordon. R Herman., Reconstruction of pictures from their projections. C.A.C.M. <u>14</u> p 759-768 (1971). Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Callscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>021</u> p 179-186 (1972). 		
 random noise IRE. Trans. Inf. Theory. <u>IT8</u> p145-154 (1962). 11. Gordon. R Herman., Reconstruction of pictures from their projections. C.A.C.M. <u>14</u> p 759-768 (1971). 12. Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). 13. Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). 14. Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). 16. Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	10.	Roberts. L.G., Picture coding using pseudo-
 p145-154 (1962). 11. Gordon. R Herman., Reconstruction of pictures from their projections. C.A.C.M. <u>14</u> p 759-768 (1971). 12. Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). 13. Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). 14. Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). 16. Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		random noise IRE. Trans. Inf. Theory. 178
 Gordon, R Herman., Reconstruction of pictures from their projections. C.A.C.M. <u>14</u> p 759-768 (1971). Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). Sklansky, J., Thresholded convolution operations. JAC.M. p 161 (1970). Rosenfeld, A. and Pfaltz, J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay, M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>G18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea, D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		p145-154 (1962).
 pictures from their projections. C.A.C.M. <u>14</u> p 759-768 (1971). Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	11.	Gordon. R Herman., Reconstruction of
 14 p 759-768 (1971). Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). 13. Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). 14. Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). 16. Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		pictures from their projections. C.A.C.M.
 Pratt. W.K., Hadamard Transform Image Coding. Proc. IEEE <u>57</u> p58-68 (1969). Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>G18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		<u>14</u> p 759-768 (1971).
 Coding. Proc. IEEE <u>57</u> p58-68 (1969). 13. Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). 14. Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>G18</u> p 733-740 (1969). 16. Preston K. Jnr., The Collscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	12.	Pratt. W.K., Hadamard Transform Image
 Sklansky. J., Thresholded convolution operations. JAC.M. p 161 (1970). Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Collscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		Coding. Proc. IEEE <u>57</u> p58-68 (1969).
 operations. JAC.M. p 161 (1970). 14. Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>G18</u> p 733-740 (1969). 16. Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	13.	Sklansky. J., Thresholded convolution
 Rosenfeld. A. and Pfaltz. J.L., Sequential Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H., Improving biomedical image quelity by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		operations. JAC.M. p 161 (1970).
 Operations in digital picture processing. J.A.CM <u>13</u> p 471-494 (1966). 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). 16. Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	14.	Rosenfeld. A. and Pfaltz. J.L., Sequential
 J.A.CM <u>13</u> p 471-494 (1966). Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		Operations in digital picture processing.
 15. Golay. M.E., Hexagonal Parallel pattern transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). 16. Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	· · ·	J.A.CM <u>13</u> p 471-494 (1966).
 transformations. I.E.E.E. Trans., Computers. <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	15.	Golay. M.E., Hexagonal Parallel pattern
 <u>C18</u> p 733-740 (1969). Preston K. Jnr., The Cellscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		transformations. I.E.E.E. Trans., Computers.
 16. Preston K. Jnr., The Callscan System, Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		<u>C18</u> p 733-740 (1969).
 Proc. W.J.C.C. p 173-178 (1961) also, Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	16.	Preston K. Jnr., The Cellscan System,
 Scientific American <u>223</u> p 72-82 (1970). 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		Proc. W.J.C.C. p 173-178 (1961) also,
 17. Pingle. K., Visual Perception by computer. Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		Scientific American <u>223</u> p 72-82 (1970).
 Pictorial Fattern Recognition p 277-284. Thompson Book Co. (1968). 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	17.	Pingle. K., Visual Perception by computer.
 Thompson Book Co. (1968). 18. Seltzer. R.H.; Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		Pictorial Fattern Recognition p 277-284.
 18. Seltzer. R.H., Improving biomedical image quality by computers. JPL. Tech. Report 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		Thompson Book Co. (1968).
<pre>quality by computers. JPL. Tech. Report 32-1336.</pre> 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972).	18.	Seltzer. R.H., Improving biomedical image
 32-1336. 19. Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	• •	quality by computers. JPL. Tech. Report
 Barnea. D.I., A class of algorithms for fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 		32-1336.
 fast digital image registration. I.E.E.E. Trans., Computers. <u>C21</u> p 179-186 (1972). 	19.	Barnea. D.I., A class of algorithms for
Trans., Computers. <u>C21</u> p 179-186 (1972).	•	fast digital image registration. I.E.E.E.
		Trans., Computers. <u>C21</u> p 179-186 (1972).
	·	

è

Shanks. J.S., Two dimensional recursive filters. SWIEECO Tech. Papers Apl.(1969). Hall. E., A survey of pre-processing and feature extraction technieques for radiographic images I.E.E. Trans. Computers.

22.

23.

24.

20.

21.

I.E.E.E. Trans. Audio and Electroacoustics Special issues on digital filtering <u>AU16</u>Sept (1968) and fast Fourier transforms <u>AU15</u> June (1967).

Macon.N. and Kiefer. M., Computer manipulation of digitized pictures <u>AFIPS</u> (1971). Ledley. R.S., FIDAC Film Input to Digital Automatic Computer. Optical and Electrooptical Information Processing M.I.J. Press. Chap.33 (1965).

Klinger. A. et.Al., Computer analysis of chromosomespatterns: Feature encoding for Flexible decision making. L.E.E. Trans. Computers <u>C21</u> (1972). Zahn. C.T., 2-D pattern description and recognition via curvature points. S.L.A.C. Report 70 (1960).

Simon. J.C., Comparing two patterns independent of transformations and small distortions. Jour. Pattern Recognition <u>PR4</u> p 73-81 (1972).

26.

25.

Mucciardi. A. and Gose. E., A comparison of 7 techniques for choosing subsets of Pattern recognition properties. I.E.E.E Trans. Computers <u>C20</u> p 1023 (1971). Bishop. N.E., Computer Image Processing for particle analysis (SR.C. Progress Report) (1971).

Stevens. M.E., Selected pattern recognition projects in Europe. Pattern Recognition PRL p 103-118. (1968).

Strohl. G.E., A model for evaluating the effectiveness of resolution enhancement algorithms. Jour. Pattern Recognition <u>PR3</u> p 325-330 (1971).

Montanari. U., A note on minimal length polygonal approximation to a digitised contour. CACM <u>13</u> p 41-47 (1970). Sklansky. J., Minimum perimeter polygons of digitised silhouettes. I.E.E.E. Trans. Comp. C21 p 260-268 (1972).

Hawkins. J., Image Processing principles and techniques. Advances in Information Processing Vol. 3. Chap. 3. (1970). Plenum. Press.

Zahn. C.T., Fourier descriptors for plane closed curves I.E.E.E. Trans. Comp. <u>C21</u> p 269-281 (1972).

32.

28.

29.

30.

31.

33•-

34 ..

Blum. H., A transformation for extracting new descriptors of shape in Models for the perception of speech and visual form (MIT) p 362-380 (1967).

Rosenfeld. A., Computer representation of planar regions by their skeletons. CACM <u>10</u> p 119-125 (1967).

Rutovitz. D., Data Structures for operations on digital images. Pictorial Pattern Recognition p 105-133 (1968).

Freeman. H., Techniques for the computer analysis of chain-encoded arbitrary plane curves. Proc. Nat. Elec. Conf. <u>17</u> (1961). Philbrick O., Shape description with the medial axis transformation, in Pictorial Pattern Recognition. Thompson Book Co. p 395-407 (1968).

Montanari. U., A method for obtaining skeletons using quasi-Euclidean distance. JACM <u>15 p</u> 600-624 (1968).

Montanari. U., Continuous skeletons from digitised images. JACM <u>16</u> p 534-549 (1969). Hu. M.K., Visual pattern recognition by moment invariants Trans. I.R.E. <u>IT-8</u> p 179-182 (1962).

Walthen-Dunn., Models for the perception of speech and visual form M.I.T (1967)

40.

39.

36 .

37.

38.

42.

41.

43.

Ross. D.T., The AED free storage package, CACM 10 p 481 (1967).

Bishop. N.E., Implementation of variable dimension arrays in FORTRAN. (letter). Computer Bulletin <u>15</u> p376 (1971).

Bishop. N.E., Particle feature parameters by optical scanning. L.U.T. Chem. Eng. Jour. <u>6 p 34-37 (1971).</u>

Kaye. B.H., Some aspects of the efficiency of statistical methods of particle size analysis. Powder Technology, <u>2</u> p97-110 (1968)

46.

45.

48.