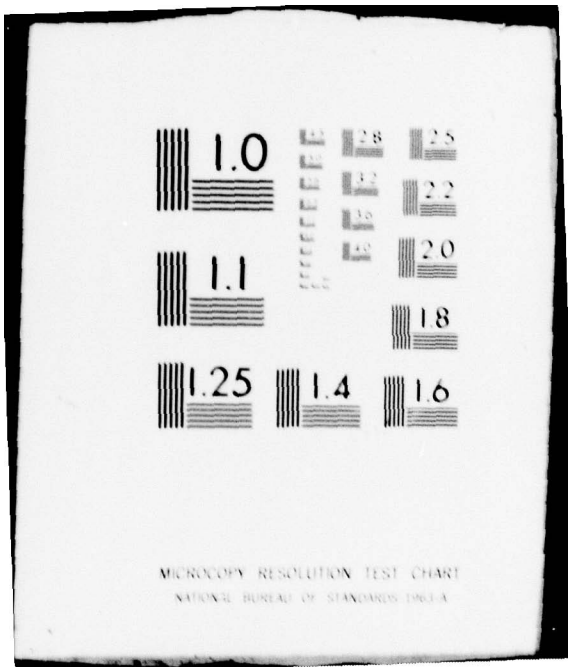
AD-A074 439 CATHOLIC UNIV OF AMERICA WASHINGTON D C DEPT OF ELEC--ETC F/G 9/4
CONCEPT DEVELOPMENT OF AUTOMATED IMAGE ANALYSIS. AUTOMATED CONT--ETC(U)
AUG 79 H F HARMUTH DAAK70-78-C-0147

UNCLASSIFIED ETL-0194 NL

| OF |
AD
A074439


1.0
1.1
1.25
1.4
1.6
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.8
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

ETL-0194

LEVEL II

CONCEPT DEVELOPMENT OF AUTOMATED IMAGE ANALYSIS

AD A074439

Automated Contour Recognition and Classification in Aerial Photography by Means of Angles and Curvature

Henning F. Harmuth
Department of Electrical Engineering
The Catholic University of America
Washington, D.C. 20064

DDC
REPRODUCED
SEP 27 1979
E

31 August 1979

Final Report for Contract DAAK-70-78-C-0147

Approved for Public Release; Distribution Unlimited.

Prepared for US Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

79 09 26 010

AD A074439
DDC FILE COPY

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ETL-0194	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) CONCEPT DEVELPMENT OF AUTOMATED IMAGE ANALYSIS		5. TYPE OF REPORT & PERIOD COVERED Contract Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Henning F. Harmuth		8. CONTRACT OR GRANT NUMBER(s) DAAK70-78-C-0147
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Electrical Engineering The Catholic University of America Washington, D.C. 20064		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Engineer Topographic Laboratories Fort Belvoir, Virginia 22060		12. REPORT DATE 31 August 1979 13. NUMBER OF PAGES 90
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Image Analysis; Contour Recognition; Feature Extraction;
Image Processing; Image Classification

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The automated contour recognition and classification in aerial photography requires a feature of contours that is invariant to shift, rotation and scaling. General polygons are characterized by their angles and their normalized sides. The normalization of the sides may be achieved by dividing either with the longest side or with the circumference of the polygon.

For a curved contour, the simplest invariant under shift and → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

rotation is the curvature. To make the curvature invariant to scaling, one may divide it by the largest curvature or multiply it with the length of the contour; the multiplication with the length of the contour is better because of the finite resolution of photographs and equipment. The principle of contour recognition and classification based on angles and curvature is worked out for practical displays that produce square patterns, such as liquid crystal or plasma tube displays. The mechanical and electronic design for experimental equipment is carried out based on a plasma tube display with 512 x 512 resolved points.

Accession For
NTIS GRA&I
DDC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist. | Avail and/or special
A

(19)

(18) ETL-0194

(6)

CONCEPT DEVELOPMENT OF AUTOMATED IMAGE ANALYSIS.

Automated Contour Recognition and Classification in Aerial Photography by Means of Angles and Curvature.

(10)

Henning F. Harmuth
Department of Electrical Engineering
The Catholic University of America
Washington, D.C. 20064

(12) 95 p.

(11)

31 August 1979

(9)

Final Report for Contract (15) DAAK-70-78-C-0147

Approved for Public Release; Distribution Unlimited.

Prepared for US Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

JCB

404 716

79 09 26 010

List of Contents

		Page
1.	Introduction	1
2.	Recognition of Polygons	3
3.	Contour Recognition	7
4.	Curvature in Finite Mathematics	16
5.	Continuous Curve, Discontinuous First Derivative	21
6.	Continuous First Derivative, Discontinuous Second Derivative	27
7.	Continuous Second Derivative	32
8.	Examples of Contours	41
9.	Summary of Theoretical Results	45
10.	Equipment for Contour Recognition	
	10.1 Block Diagram	48
	10.2 Mechanical Design	50
	10.3 Principle of Electronic Design	51
	10.4 Display Drive Control DDC	55
	10.5 Binary Function Generator BFG	57
	10.6 Format and Selector Switch Control	58
	10.7 Selector Switches SSW	60
	10.8 Power Driven PDR	60
	10.9 Fototube and Sensor Circuit SEN	62
11.	Conclusions and Recommendations	63
12.	Comments on Technology	65
13.	References	67
	Illustrations 1 - 40	68
	Blueprints (15 sheets in a roll supplied with a few selected copies of this report)	

1. INTRODUCTION

The problem of recognizing patterns in a two-dimensional image is an extremely general one, and we must break it up into more limited problems in order to make progress. We know from the transmission of information that we can divide the problem of reception of time signals - or pattern recognition with one variable - into two parts. First, we have the reception of one signal out of a small set of possible signals. This problem is encountered, e.g., in the reception of teletype signals. We know, due to the synchronization between transmitter and receiver, when a signal is received, and we also know that it must be one of the 32 possible teletype signals. The preferred way to decide which signal was actually transmitted is to multiply the received signal with samples of the 32 possible signals, and to integrate the 32 products. The sample signal producing the largest value of the integral is accepted to be the one which equals the transmitted signal. This process is usually referred to as signal detection by means of cross-correlation with sample signals.

Very different is the detection of the (time) patterns of telephone signals. Even though the number of telephone signals transmittable during any given time interval is finite, the number is so large that cross-correlation with sample functions becomes impractical. The ear uses a detection process that may be described as a sliding correlation based on the Fourier series.

We can apply the same classification to two-dimensional (spatial) patterns with the variables x and y rather than the one time variable t. A machine that can detect the characters of a typewriter - of which there are about 100 - needs to be "synchronized" to the location of the characters, and may then use cross-correlation with 100 sample characters. This process cannot be used to detect handwritten characters, or patterns on aerial photographs, since the number of possible "sample characters" would be enormous, even though it would be finite. We need something comparable to the detection of audio signals by the ears.

Let us look at Fig. 1. We recognize instantly several triangles at various locations with various rotations, sizes, and representations, but equal shape. With representations we mean that a triangle may be represented by a black line with three corners, by a black area with three edges, and in many other ways. Equal shape means that the triangles can be made to coincide by shifting, rotating, changing the size, and changing the representation. Alternately, two triangles with angles α_1, β_1, γ_1 and α_2, β_2, γ_2 - listed in both cases either clockwise or anti clockwise - have the same shape for $\alpha_1=\alpha_2$, $\beta_1=\beta_2$, $\gamma_1=\gamma_2$.

If we had used correlation with sample triangles to decide that Fig. 1 shows triangles, we would have to correlate triangles with various shapes, representations, x-shifts, y-shifts, rotations and sizes. This would require so much time that we still would not have come to a decision. Obviously, our brain does not use correlation to recognize a triangle.

How do we recognize a triangle? The word implies that the three angles are the important feature. The angles are independent of shift, rotation, size and representation. They are the invariants of the triangular shape, and we have just used them to define equal shape.

A check of the literature on human perception shows that experiments exist that support the claimed importance of the angles. The eye of the human newborn is attracted primarily by the angles of triangles presented to it, and not by the sides or the interior (Salapatek and Kessen 1966). A further clue that the eye recognizes patterns from their contours - and particularly from the change of the contour - rather than by cross-correlation with sample patterns is provided by the well-known sensitivity to contours and the insensitivity to average brightness. A contour in a photograph made fuzzy by motion or improper focusing is spotted immediately, even though it would have almost no effect on the integral of a cross-correlation with a sample pattern. On the other hand, the average brightness of a photograph has a huge effect on this integral, but the eye is almost insensitive to average brightness within a wide range.

2. RECOGNITION OF POLYGONS

Let us generalize the concept of invariants of shape. Figure 2a shows a polygon with six sides a ... f and six angles $\alpha \ldots \zeta$. Knowledge of the angles is no longer sufficient, we must also know the ratios of the length of the sides. In order to arrive at a unique set for these ratios, we look first for the longest side, which is f. We then form the ratios a/f, b/f, ..., g/f, 1. All ratios are 1 or smaller. Instead of the longest side one may also use the sum of all sides for normalization.

The longest side may be used to produce a unique listing of the ratios of the sides as well as of the angles. We start at the longest side f, and run in the mathematically positive sense (counter clockwise) around the polygon. The following listing of side ratios and angles is obtained:

$$1, \alpha, a/f, \beta, b/f, \gamma, \ldots, e/f, \zeta \qquad (1)$$

Note that the angles α, β, ... are always measured in the mathematically positive sense, starting from the side already listed and ending at the side to be listed next.

If two polygons produce the same list, we say they have the same shape.

A couple of refinements are still, needed. First, consider a polygon that is so large that we cannot see all of it at the same time. We will not know in which direction to proceed in order to run counter clockwise around it. A machine is in this position. It can observe only local features of a polygon, either a straight line or an angle, but it does not observe the whole polygon at once. Without knowing what will turn out to be running around the polygon in the mathematically positive or negative sense, one has no choice but to run around in both possible directions. This is shown in Fig. 2b. One obtains the listing

$$1, \overline{\zeta}, e/f, \overline{\varepsilon}, d/f, \overline{\delta}, \ldots, a/f, \overline{\alpha} \qquad (2)$$

The bars over the angles indicate the complement, e.g., $\bar{\alpha} = 2\pi - \alpha$.

If two polygons produce the same listing, they have the same shape. Actually, only one listing has to be produced by a shape recognizing machine, if both lists are stored for reference.

Let us replace the angles in Eqs. (1) and (2) by their complements:

$$1, \bar{\alpha}, a/f, \bar{\beta}, b/f, \bar{\gamma}, \ldots, e/f, \bar{\zeta} \qquad (3)$$

$$1, \zeta, e/f, \varepsilon, d/f, \delta, \ldots, a/f, \alpha$$

A comparison with Figs. 2c and d shows that these two listings define the mirror image of the shape of Fig. 2a. Hence, complementing the angles is the way to include mirror images into the definition of "equal shape".

The second required refinement is for polygons that have two or more sides of equal length, and which are at the same time longer than all the other sides. An example is shown in Fig. 3, where the two sides a and f are equally long. Starting with f and proceeding in the direction indicated in Fig. 3a we obtain the following listing:

$$1, \alpha, 1, \beta, b/f, \gamma, c/f, \delta, d/f, \varepsilon, e/f, \zeta \qquad (4)$$

However, one may start just as well with a to obtain a different listing:

$$1, \beta, b/f, \gamma, c/f, \delta, d/f, \varepsilon, e/f, \zeta, 1, \alpha \qquad (5)$$

Since Eqs. (4) and (5) differ by a cyclic shift only, one has to store only one listing for reference. If the machine makes a listing of a polygon that has to be identified, and more than one ratio 1 occurs for the sides, one must shift the listing cyclically so that each 1 comes to the head of the list, and compare each such list with the stored reference list.

Fig. 3b repeats Fig. 3a for running around the polygon clockwise. No additional problems are encountered.

When the shape of a polygon is to be recognized by our line extraction method, one must remember the starting point, since one would follow otherwise the closed line forever. Apart from this requirement, our method may be used in exactly the same way to recognize open ended, straight-line shapes, as shown in Figs. 4a and b. However, the listing cannot start with the longest line but with one of the lines at the ends:

$$a/b, \alpha, 1, \beta, c/b \tag{6}$$

$$c/b, \bar{\beta}, 1, \bar{\alpha}, a/b \tag{7}$$

If we take the complements of the angles, we obtain:

$$a/b, \bar{\alpha}, 1, \bar{\beta}, c/b \tag{8}$$

$$c/b, \beta, 1, \alpha, a/b \tag{9}$$

These listings define the mirror image shapes of Figs. 4c and d.

In order to extend the theory beyond straight-line structures we need a generalization of the concept of angle that is applicable to curved lines. Such a concept is the curvature of a line. The curvature is invariant to shifts and rotations, and by introducing a normalization factor we can make it invariant to size or scaling.

Refer to Fig. 5a for the definition of curvature. Two straight lines are drawn through the three points n-1, n and n+1. The angle between the lines at the point n is denoted $\Delta\varepsilon(n)$, and the distance between the points n and n+1 by $\Delta s(n)$. The limit

$$\lim_{\Delta s(n) \to 0} \Delta\varepsilon(n)/\Delta s(n) = d\varepsilon/ds \tag{10}$$

is called the curvature in differential mathematics.

It is obvious that the curvature is independent of any shift or rotation of the three points n-1, n and n+1. Scaling influences the curvature through $\Delta s(n)$, but a multiplication with a length characteristic for the contour will remove this influence.

The main difficulty we have to face is the fact that an aerial photograph or some automated analyzer do not have infinite resolution and thus cannot use the definition of Eq. (10). One must develop mathematical methods based on the mathematics of finite differences. To show why this must lead to results very different from the ones of differential mathematics, we investigate the concept of "radius of curvature".

In Fig. 5b a circle with radius $\rho(n)$ is drawn through the points n and n+1. The line from n-1 to n equals the tangent of the circle in point n. We can determine $\rho(n)$ and define a curvature $\varkappa(n) = 1/\rho(n)$ as shown in Fig. 5b.

Instead of defining a circle in this way, we may also draw a circle through all three points n-1, n and n+1 as shown in Fig. 5c. The radius $\rho(n)$ cannot be calculated exactly without solving a transcendental equation. This difficulty does not occur in differential mathematics, since the approximation $\sin\Delta\varepsilon \doteq \Delta\varepsilon$ becomes exact when $\Delta\varepsilon$ becomes a differential $d\varepsilon$. Furthermore, in differential mathematics we would have

$$\lim_{\Delta s \to 0} \tfrac{1}{2}\left[\Delta s(n-1) + \Delta s(n)\right] = ds, \tag{11}$$

and the relation

$$\rho = ds/d\varepsilon \tag{12}$$

would follow from Fig. 5c, which is just the inverse of Eq.(10). No such simple relation exists for finite differences $\Delta s(n)$ and $\Delta\varepsilon(n)$. If one wants to define a radius of curvature in finite mathematics, one would not want to use the one of Fig. 5c which requires the solution of a transcendental equation but prefer the one of Fig. 5b or - as will be shown later on - a still

other definition that is more useful than either the definition of Fig. 5b or c.

3. CONTOUR RECOGNITION

We base the following investigation on the premise that a pattern or a contour does not consist of nondenumerably infinite points with the infinitesimal area dxdy but of a finite number of points with a finite area. We choose this finite area to be squares with the sides Δx and $\Delta y = \Delta x$, having thus the area $\Delta x \Delta y$. An ellipse is shown in this representation in Fig. 6.

The reason for choosing squares for the elemental points is that liquid crystal displays produce such points (Harmuth 1977, Figs. 234-2 and 234-3). Plasma tube displays also come close to producing such points. A cathode ray display produces circular rather than square points, and these points are not inherently at such fixed positions as in liquid crystal and plasma tube displays.

Let the 34 x 26 elemental points in Fig. 6 represent a section of a liquid crystal or plasma tube display. A "scanning pattern" of four elemental points - as shown in the lower left of Fig. 6 - shall be illuminated. This scanning pattern can be moved to any location of the field of 34 x 26 elemental points in Fig. 6. In the case of a liquid crystal, the four points of the scanning pattern are light transparent while all the other points of the field of 34 x 26 points are opaque; in the case of a plasma tube display the scanning pattern consists of four glow discharge points while all the other points are dark.

Let a photo transparency be overlayed on the display. This transparency shall be opaque for the approximately elliptical area in Fig. 6 containing the crossed squares, and transparent for the other squares. In reality, it is only necessary that the contour of the elliptical area is noticeably less transparent than the area outside the ellipse.

The next step is to make the scanning pattern run around the contour of the ellipse in a way that can be automated. The

scanning pattern in Fig. 6 is moved to the right so that its point 0 coincides with the point $\xi=0$, $\eta=1$ of the display; the scanning pattern is then in the location $\xi=0$, $\eta=1$. None of the four illuminated points of the scanning pattern will be blocked by an opaque section of the transparency; an observer will see four bright points or the pattern denoted 0 in the upper left corner of Fig. 7. The number 0 indicates that none of the 2x2 illuminated points of the scanning pattern is blocked by an opaque section of the transparency.

The pattern 0 in Fig. 7 shows the command "scan". It means that the scanning pattern should be moved one step to the right; if the right edge of the field in Fig. 6 is reached, "scan" means that the scanning pattern is returned to the left edge, $\xi=0$, and moved one step up to $\eta=2$.

Moving the scanning pattern one step to the right from $\xi=0$, $\eta=1$ brings it to the point $\xi=1$, $\eta=1$. Again, the pattern is not blocked by any opaque section, the contour operator 0 of Fig. 7 and the command "scan" is obtained. This is repeated until the scanning pattern reaches the position $\xi=11$, $\eta=1$; the contour operator 1B of Fig. 7 is now obtained.

Contour operator 1B is shown with Arrow 1 pointing to the right and Arrow 2 pointing upward. This means that the scanning pattern should as the first choice be moved +1 in the direction of ξ. If this first choice is prohibited, the scanning pattern is moved as the second choice +1 in the direction of η. The only reason for not using the first choice is that the scanning pattern would be moved back to the point it had just previously been. A practical example will show the working of the contour operators better than many words.

Table 1 lists on the top left the values $\xi=11$, $\eta=1$ for which the contour operator 1B was observed. The point n=1 is recognized, and Fig. 7 shows that one has to advance +1 in the direction ξ if the contour operator 1B is observed. This brings the scanning pattern to $\xi=12$, $\eta=1$. The contour operator 2B of Fig. 7 is observed, the point n=2 is recognized, and one must advance +1 in direction ξ. The advancement by +1 in direction ξ and the contour operator 2B are repeated until the point

Table 1. Scanning of the contour of the ellipse of Fig.6.

location of scanning pattern		observed contour operator	recognized new point	command to advance		angle	line element	curvature
ξ	η		n	$\Delta\xi$	$\Delta\eta$	$\Delta\gamma(n)$	$\Delta s(n)/\Delta x$	$\frac{\Delta\gamma(n)}{\Delta s(n)/\Delta x}$
11	1	1B	1	+1		0	1	0
12	1	2B	2	+1		0	1	0
13	1	2B	3	+1		0	1	0
14	1	2B	4	+1		0	1	0
15	1	2B	5	+1		0	1	0
16	1	2B	6	+1		0	1	0
17	1	2B	7	+1		0	1	0
18	1	2B	8	+1		0	1	0
19	1	2B	9	+1		0	1	0
20	1	2B	10	+1		$\pi/4$	$(1+\sqrt{2})/2$	+0.6506
21	1	1C	-		+1	-	-	-
21	2	3B	11	+1		$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
22	2	2B	12	+1		0	1	0
23	2	2B	13	+1		0	$(1+\sqrt{2})/2$	0
24	2	1C	-		+1	-	-	-
24	3	3B	14	+1		$\pi/4$	$(1+\sqrt{2})/2$	+0.6506
.								
.								
.								
30	14	2C	26		+1	0	$(1+\sqrt{2})/2$	0
30	15	1D	-	-1		-	-	-
29	15	3C	27		(+1)	$\pi/4$	$(1+\sqrt{2})/2$	+0.6506
29	16	2C	28		+1	$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
29	17	1D	-	-1		-	-	-
28	17	3C	29		(+1)	$\pi/4$	$\sqrt{2}$	+0.5554
28	18	1D	-	-1		-	-	-
27	18	3C	30		(+1)	0	$\sqrt{2}$	0
27	19	1D	-	-1		-	-	-
26	19	3C	31		(+1)	0	$\sqrt{2}$	0

$\xi=21$, $\eta=1$ is reached. Now the contour operator 1C is observed, no new point is recognized, and the scanning pattern must be advanced +1 in the direction η. We leave it to the reader to verify the entries in the first six columns of Table 1 with the help of Figs. 6 and 7.

The entries (+1) in the column "command to advance, η" in Table 1 signify that the second choice rather than the first choice of advancement is used, which would be to advance +1 in direction ξ according to Fig. 7. One may readily verify that this first choice would return the scanning pattern to the point it had just been previously (from $\xi=29$, $\eta=15$ to $\xi=30$, $\eta=15$; from $\xi=28$, $\eta=17$ to $\xi=29$, $\eta=17$; etc.).

The three additional columns of Table 1 will be discussed later on in more detail. They show the angle $\Delta\gamma(n)$ between two lines drawn through the points n-1, n and n,n+1, furthermore the line element $\Delta s(n)/\Delta x$ and a curvature $\Delta\gamma(n)\Delta x/\Delta s(n)$.

Let us discuss how a machine can recognize the contour operators of Fig. 7. In principle, one could do just like our eyes do and use many - or at least four - receptors to recognize the operation. However, this leads to great problems in the implementation of equipment. A better solution is to make a sequence of four detections rather than four simultaneous detections. The reason is not so much to reduce the number of photo detectors and amplifying circuits from four to one, but to simplify the optical design.

Figure 8 shows again the scanning pattern with four bright points of Fig. 6 in the upper left corner. This pattern is now denoted as the two-dimensional Walsh function $wal(0,\xi) \times wal(0,\eta)$. In addition, the Walsh functions $wal(0,\xi)\ wal(1,\eta)$, $\widehat{wal}(1,\xi)\ wal(0,\eta)$ and $wal(1,\xi)\ \widehat{wal}(1,\eta)$ are shown. The white areas represent the value +1, the black areas the value -1. We produce the Walsh transform of the 16 contour operators with these four functions. In terms of differential mathematics we get the coefficients a(i,k)

$$a(i,k) = \int_0^2 \int_0^2 F(\xi,\eta)\ \widehat{wal}(i,\xi)\ \widehat{wal}(k,\eta)\ d\xi d\eta \qquad (13)$$

We have to insert each one of the 16 contour operators of Fig. 7 for $F(\xi,\eta)$. Ideally, the crossed squares of the operator are perfectly opaque and their brightness equals 0; the clear squares are fully transparent and their brightness may be set equal to +1. In reality, we would like to recognize the elliptical area in Fig. 6 if it is somewhat less transparent than the surrounding area, since this is what our eyes can accomplish. Hence, we assign the brightness q to the crossed, opaque squares of the operators in Fig. 7, and the brightness p to the clear squares:

p = brightness of transparent elemental point (14)

q = brightness of opaque or less transparent elemental point

$p/q = r$ = brightness ratio

$0 \leqq q < 1,\ 0 < p \leqq 1,\ p/q > 1$

In terms of finite mathematics we represent each operator of Fig. 7 by an array of four coefficients p and q; e.g., the operator 1A is represented as follows

$$\begin{bmatrix} p & p \\ p & q \end{bmatrix} \tag{15}$$

The four Walsh functions of Fig. 8 are represented by four arrays with the coefficients +1 and -1:

$$\begin{bmatrix} +1 & +1 \\ +1 & +1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 \\ +1 & +1 \end{bmatrix} \quad \begin{bmatrix} +1 & -1 \\ +1 & -1 \end{bmatrix} \quad \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix} \tag{16}$$

Multiplication of the array of Eq.(15) with the arrays of Eq.(16) and summation of the products yield the coefficient a(i,k):

$$a(0,0) = \sum \begin{bmatrix} p & p \\ p & q \end{bmatrix} \times \begin{bmatrix} +1 & +1 \\ +1 & +1 \end{bmatrix} = 3p + q = 3p(1+q/p) \tag{17}$$

$$a(0,1) = \sum \begin{bmatrix} p & p \\ p & q \end{bmatrix} \times \begin{bmatrix} -1 & -1 \\ +1 & +1 \end{bmatrix} = -p + q = -p(1-q/p)$$

$$a(1,0) = \sum \begin{bmatrix} p & p \\ p & q \end{bmatrix} \times \begin{bmatrix} +1 & -1 \\ +1 & -1 \end{bmatrix} = p - q = p(1-q/p)$$

$$a(1,1) = \sum \begin{bmatrix} p & p \\ p & q \end{bmatrix} \times \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix} = p - q = p(1-q/p)$$

The multiplication and summation process is the same as used for the reduction of two tensors of rank 2 to a constant.

The four coefficients of Eq.(17) are listed in Table 2 for the operator 1A. The listed coefficients for the other operators are obtained in the same way.

Since the absolute brightness of p and q is of less interest than their ratio r = p/q, we divide the coefficients a(0,1), a(1,0) and a(1,1) by a(0,0). The values in columns 6-8 of Table 2 are obtained. One may verify that R/3 is always smaller than Q or P. Hence, by setting threshholds at +R/6 and -R/6 one can divide the coefficients in columns 6-8 into three classes:

a) coefficients larger than +R/6 (18)
b) coefficients between +R/6 and -R/6
c) coefficients smaller than -R/6

$$R = \frac{1 - q/p}{1 + q/3p}$$

The numbers +1,0 and -1 are written in columns 9-11 of Table 2 for these three classes of coefficients. Each set of these "quantized" coefficients defines a contour operator, e.g., the set -1,0,0 defines the operator 2D. Hence, by using the four Walsh patterns of Fig. 8a and doing the processing according to Table 2 we need only one phototube with associated circuits but four observations to recognize any one of the 16 contour operators in Fig. 7 uniquely.

Since a liquid crystal or a plasma tube produces either "light" or "no light" at a point, but never "positive light" and "negative light", we still have to explain how we can implement

Table 2. Recognition of the 16 contour operators of Fig.7 by means of their Walsh coefficients a(i,k). $R=(1-q/p)/(1+q/3p)$, $Q=(1-q/p)/(1+q/p)$, $P=(1-q/p)/(1+3q/p)$.

1	2	3	4	5	6	7	8	9	10	11
contour operator	a(0,0)	a(0,1)	a(1,0)	a(1,1)	$\frac{a(0,1)}{a(0,0)}$	$\frac{a(1,0)}{a(0,0)}$	$\frac{a(1,1)}{a(0,0)}$	$\frac{a(0,1)}{a(0,0)}$ +R/6, -R/6	$\frac{a(1,0)}{a(0,0)}$ +R/6, -R/6	$\frac{a(1,1)}{a(0,0)}$ +R/6, -R/6
0	4p	0	0	0	0	0	0	0	0	0
1A	3p+q	-p+q	+p q	+p-q	-R/3	+R/3	+R/3	-1	+1	+1
1B	3p+q	+p-q	+p-q	-p+q	+R/3	+R/3	-R/3	+1	+1	-1
1C	3p+q	+p-q	-p+q	+p-q	+R/3	-R/3	+R/3	+1	-1	+1
1D	3p+q	-p+q	-p+q	-p+q	-R/3	-R/3	-R/3	-1	-1	-1
2A	2(p+q)	0	2(p-q)	0	0	+Q	0	0	+1	0
2B	2(p+q)	2(p-q)	0	0	+Q	0	0	+1	0	0
2C	2(p+q)	0	-2(p-q)	0	0	-Q	0	0	-1	0
2D	2(p+q)	-2(p+q)	0	0	-Q	0	0	-1	0	0
2E	2(p+q)	0	0	-2(p-q)	0	0	-Q	0	0	-1
2F	2(p+q)	0	0	2(p-q)	0	0	+Q	0	0	+1
3A	p+3q	p-q	p-q	p-q	+P	+P	+P	+1	+1	+1
3B	p+3q	p-q	-(p-q)	-(p-q)	+P	-P	-P	+1	-1	-1
3C	p+3q	-(p-q)	-(p-q)	p-q	-P	-P	+P	-1	-1	+1
3D	p+3q	-(p-q)	p-q	-(p-q)	-P	+P	-P	-1	+1	-1
4	4q	0	0	0	0	0	0	0	0	0

the Walsh functions practically. Actually, we do not need these functions but the coefficients a(0,0), a(0,1), a(1,0), and a(1,1) of Table 2. We may use the two-dimensional block functions $\widehat{blo}(i,\xi)\ \widehat{blo}(k,\eta)$ of Fig. 8b to obtain the Walsh coefficients.

Let the elemental point represented by the clear square of the function $\widehat{blo}(0,\xi)\ \widehat{blo}(0,\eta)$ be illuminated and the three elemental points represented by the black squares be dark. This implies that one square of a liquid crystal is transparent or one glow point of a plasma tube emits light. A photo transparency be overlayed; one of the sixteen patterns of Fig. 7 is thus overlayed on the pattern $\widehat{blo}(0,\xi)\ \widehat{blo}(0,\eta)$ of Fig. 8. The light passing through to the phototube will be proportionate to either p or q, depending on whether a transparent or an opaque square of the photo transparency overlays the bright square of $\widehat{blo}(0,\xi)\times \widehat{blo}(0,\eta)$. Whichever it is, the output voltage of the phototube represents the coefficient b(0,0), which can be written in analogy to Eq. (13):

$$b(i,k) = \int_0^2 \int_0^2 F(\xi,\eta)\ \widehat{blo}(i,\xi)\ \widehat{blo}(k,\eta)\ d\xi d\eta \qquad (19)$$

$$i = k = 0$$

Illumination of the liquid crystal or the plasma tube according to the other three functions of Fig. 8b produces the coefficients b(0,1), b(1,0) and b(1,1). A comparison of the Walsh and the block functions yields immediately the following relations:

$$a(0,0) = b(0,0) + b(0,1) + b(1,0) + b(1,1) \qquad (20)$$

$$a(0,1) = b(0,0) + b(0,1) - b(1,0) - b(1,1)$$

$$a(1,0) = b(0,0) - b(0,1) + b(1,0) - b(1,1)$$

$$a(1,1) = b(0,0) - b(0,1) - b(1,0) + b(1,1)$$

Hence, we have a practical way to obtain the coefficients a(0,0) to a(1,1) of Table 2. There is another way that does not require the block functions; it is based on interpreting the black areas of the Walsh functions as "no light emitted" and the clear areas as "light emitted."

Summing up, we have developed a deterministic method to find point-by-point the contour between areas of different light transmission of a photo transparency, if this transparency is placed on a liquid crystal or a plasma tube display. The method requires that one point of the display emits light, which passes through the transparency to the phototube. A sequence of four signals received by the phototube characterizes one of the 16 contour operators of Fig. 7, and this operator commands which quadruple of points should be illuminated next.

Let us end this section with an explanation how the arrows for the advancement were chosen in Fig. 7. The two arrows shown for the contour operator 1A assure that the opaque square will occur in the following operator too. For instance, if the operator 1A is recognized and the scanning pattern is advanced one step to the right, the next recognized pattern could be either 1D, 2D, 2E or 3D. If, on the other hand, the scanning pattern would have advanced one step to the left after recognition of the contour operator 1A, one might have lost the opaque point and obtained the contour pattern 0, which means the contour would have been lost and we would be back to scanning to find a contour.

One may readily verify that all arrows in Fig. 7 have been chosen so that the scanning pattern will never advance into an area that yields either the all transparent operator 0 or the all opaque operator 4. Putting it differently, the arrows are chosen so that the next contour operator will have at least one transparent square and at least one opaque square.

The choice of the advance in the directions of the two arrows is arbitrary. We have chosen the following order of preference: 1. advance right; 2. advance up; 3. advance left; 4. advance down. In other words, the arrow having the smaller angle with the positive x-axis got the number 1, the

other got the number 2. This rule also applies to the operators 2E and 2F.

4. CURVATURE IN FINITE MATHEMATICS

We have seen in Fig. 5a that one needs three adjacent points to define the angle $\Delta\varepsilon(n)$ and the curvature in point n. The three points n-1, n and n+1 could have any position relative to each other in Fig. 5, but the square pattern of Fig. 6 restrains this freedom greatly. Figure 9 shows three typical examples for the location of three adjacent points in the square field of Fig. 6. We must develop some concepts of finite mathematics on a square field before we can proceed.

In differential mathematics using cartesian coordinates the curvature is defined by the relation

$$\varkappa(x) = \frac{d\varepsilon}{ds} = \frac{y''}{(1+y'^2)^{3/2}} \tag{21}$$

This expression requires the first and second derivatives, and we will thus investigate the first and second derivatives of finite mathematics on a square field.

In finite mathematics we have at the point n of a curve a left difference quotient $\eta'_\ell(n)$ and a right difference quotient $\eta'_r(n)$.

$$\eta'_\ell(n) = \frac{\eta(n) - \eta(n-1)}{\xi(n) - \xi(n-1)} \tag{22}$$

$$\eta'_r(n) = \frac{\eta(n+1) - \eta(n-1)}{\xi(n+1) - \xi(n-1)} \tag{23}$$

Refer to Fig. 9a. The variable n determines a point along the line by counting from the arbitrarily chosen first point n=1. The equivalent in differential mathematics is the length of the curve from the arbitrarily chosen beginning to the point n. The cartesian coordinates of this point n are $\xi=j$ and $\eta=k$. The left difference quotient becomes

$$\eta'_{\ell}(n) = \frac{(k-1) - (k-1)}{j - (j-1)} = 0,$$

and the right difference quotient becomes

$$\eta'_{r}(n) = \frac{k - (k-1)}{(j+1) - j} = +1$$

The two quotients η'_{ℓ} and η'_{r} are shown in Fig. 9a to determine the tangent of the angles $\gamma_{\ell}(n)$ and $\gamma_{r}(n)$ with which straight lines through the points n-1, n and n, n+1 intersect the ξ-axis. The angle between the two lines is denoted $\Delta\gamma(n)$:

$$\Delta\gamma(n) = \gamma_{r}(n) - \gamma_{\ell}(n) \qquad (21)$$

Figures 9b and c give two more examples of three adjacent points, and the two straight lines with the angle $\Delta\gamma$ between them defined by three points. A closer study shows that there are only 40 possible sequences of three points on a square field. The first twenty are shown in Fig. 10, the other twenty in Fig. 11. These two figures differ only in the direction in which the numbering of the three adjacent points increases. A zero "0" is shown to indicate this direction of numbering.

Only eight different values of the left and the right difference quotients η'_{ℓ} and η'_{r} occur in Figs. 10 and 11. These values are listed in Tables 3 and 4. A complete listing of the quotients η'_{ℓ} and η'_{r} for the 40 configurations a - t and a' - t' of Figs. 10 and 11 is given in Table 5. Also shown is the second order difference quotient $\eta''(n)$:

$$\eta''(n) = \frac{\eta(n+1) - 2\eta(n) + \eta(n-1)}{\xi(n+1) - 2\xi(n) - \xi(n-1)} \qquad (24)$$

Finally the angle $\Delta\gamma(n)$ defined by the two straight lines through the three points - intersecting in the center point - are shown. One may see that η'_{ℓ}, η'_{r} and η'' uniquely define $\Delta\gamma$. Since the three difference quotients in turn are uniquely defined by

Table 3. The eight possible values of the left difference quotient.

$$\eta_l'(n) = \frac{\eta(n) - \eta(n-1)}{\xi(n) - \xi(n-1)}$$

$\eta(n)-\eta(n-1)$	-1	-1	-1	0	0	+1	+1	+1
$\xi(n)-\xi(n-1)$	-1	0	+1	-1	+1	-1	0	+1
$\eta_l'(n)$	+1	∞	-1	0	0	-1	∞	+1

Table 4. The eight possible values of the right difference quotient.

$$\eta_r'(n) = \frac{\eta(n+1)-\eta(n)}{\xi(n+1)-\xi(n)}$$

$\eta(n+1)-\eta(n)$	-1	-1	-1	0	0	+1	+1	+1
$\xi(n+1)-\xi(n)$	-1	0	+1	-1	+1	-1	0	+1
$\eta_r'(n)$	+1	∞	-1	0	0	-1	∞	+1

$\xi(n-1)$, $\xi(n)$, $\xi(n+1)$, $\eta(n-1)$, $\eta(n)$ and $\eta(n+1)$, which have been obtained in Table 1 by means of the contour operators, we can derive the values of $\Delta\gamma$ by means of the contour operators.

Let us show an example. For point n=10 in column 4 of Table 1 we find the values $\xi(9) = 19$, $\xi(10) = 20$, $\xi(11) = 21$, $\eta(9) = 1$, $\eta(10) = 1$ and $\eta(11) = 2$. It follows:

$$\eta'_{\ell}(10) = \frac{1-1}{20-19} = 0, \quad \eta'_{r}(10) = \frac{2-1}{21-20} = 1, \quad \eta''(10) = \frac{2-2+1}{21-40+19} = \infty$$

Table 5 shows this set of values for the sequence e in Fig. 10, and the angle $\Delta\gamma = \pi/4$ in correspondence with Table 1.

As a second example, consider the point n=11 of Table 1 with $\xi(10) = 20$, $\xi(11) = 21$, $\xi(12) = 22$, $\eta(10) = 1$, $\eta(11) = 2$ and $\eta(12) = 2$:

$$\eta'_{\ell}(11) = \frac{2-1}{21-20} = 1, \quad \eta'_{r}(11) = \frac{2-2}{22-21} = 0, \quad \eta''(11) = \frac{2-4+1}{22-42+20} = \infty$$

This set of values holds for the sequence h in Table 5 and Fig. 10. The angle $\Delta\gamma = -\pi/4$ is obtained in agreement with $\Delta\gamma(11)$ in Table 1.

The definition of Eq.(10) for the curvature contained the angle $\Delta\varepsilon(n)$ - which we denote now by $\Delta\gamma(n)$ for reasons to be seen later - and the length element $\Delta s(n)$. We will not use the length element in this form but for completeness sake we show in Fig. 12 that in finite mathematics on a square field there can be only three values for the normalized length element $\Delta s(n)/\Delta x$. The value of $\Delta s(n)/\Delta x$ is uniquely defined by the quotients $\eta'_{\ell}(n)$, $\eta'_{r}(n)$ and $\eta''(n)$, since one could list the value of $\Delta s(n)/\Delta x$ in addition to the value of $\Delta\gamma(n)$ in Table 5.

The straight forward way to arrive at a curvature as function of the point n would be to calculate the line element $\Delta s(n)/\Delta x$ and the curvature $\Delta\gamma(n)\Delta x/\Delta s(n)$ as shown in the last two columns of Table 1. It can be seen that this is not a satisfactory way since the curvature in Table 1 is mostly 0 and alternates between positive and negative values. One would have to average the curvature over many values of n to obtain a satisfactory result.

Table 5. The left and right difference quotients η_l', η_r', the second order difference quotients η'', and the angles $\Delta\gamma$ for the sequences of three points a - t in Fig.10 and a' - t' in Fig.11.

	η_l'	η_r'	η''	$\Delta\gamma$		η_l'	η_r'	η''	$\Delta\gamma$		η_l'	η_r'	η''	$\Delta\gamma$		η_l'	η_r'	η''	$\Delta\gamma$
a	0	0	0/0	0	k	-1	∞	0	$-\pi/4$	a'	0	0	0/0	0	k'	∞	+1	0	$-\pi/4$
b	∞	∞	0/0	0	l	+1	∞	0	$+\pi/4$	b'	∞	∞	0/0	0	l'	∞	+1	0	$-\pi/4$
c	+1	+1	0/0	0	m	-1	+1	∞	$+\pi/2$	c'	+1	+1	0/0	0	m'	+1	-1	∞	$-\pi/2$
d	-1	-1	0/0	0	n	+1	-1	∞	$+\pi/2$	d'	-1	-1	0/0	0	n'	-1	+1	∞	$+\pi/2$
e	0	+1	∞	$+\pi/4$	o	+1	-1	0	$+\pi/2$	e'	+1	0	∞	$-\pi/4$	o'	-1	+1	0	$-\pi/2$
f	0	-1	∞	$-\pi/4$	p	-1	+1	0	$-\pi/2$	f'	-1	0	∞	$+\pi/4$	p'	+1	-1	0	$+\pi/2$
g	-1	0	∞	$+\pi/4$	q	0	∞	-1	$+\pi/2$	g'	0	-1	∞	$-\pi/4$	q'	∞	0	-1	$-\pi/2$
h	+1	0	∞	$-\pi/4$	r	∞	0	-1	$-\pi/2$	h'	0	+1	∞	$+\pi/4$	r'	0	∞	-1	$-\pi/2$
i	∞	+1	0	$-\pi/4$	s	∞	0	+1	$+\pi/2$	i'	+1	∞	0	$+\pi/4$	s'	0	∞	+1	$-\pi/2$
j	∞	-1	0	$+\pi/4$	t	0	∞	+1	$-\pi/2$	j'	-1	∞	0	$-\pi/4$	t'	∞	0	+1	$+\pi/2$

The first reason for not following the straight forward way is thus that averaging is an unsatisfactory method. There is, however, an even more important reason. In aerial photography analysis we are mainly interested in man-made objects. The contours of such objects are primarily straight lines and circles. Hence, one will want a method that first tries to fit the contour obtained from the contour operators by straight lines; if this is impossible, a fit by circles should be tried next; and only if straight lines and circles fail will one admit a more general curvature.

For an illustration of the importance of this concept refer to Fig. 13, which shows an opaque rectangular area on the same square field as the ellipse of Fig. 6. Table 6 shows the characteristic values for the scanning of the contour of the rectangle in analogy to Table 1. If we simply average the values for the curvature in the last column, we would obtain zero curvature for the straight lines of the rectangle, but the corners at points n=1, 26, 39, 64 would be rounded.

5. CONTINUOUS CURVE, DISCONTINUOUS FIRST DERIVATIVE

We return to Table 1 but we disregard the values listed for the line element and the curvature. Using only the values of $\Delta\xi$ and $\Delta\eta$ as function of the variable n we can plot the 72 points n=1 to n=72 of Fig. 14a. This plot is already independent of any shifts in the directions ξ and η as well as from rotations. The independence from shifts is evident from the way $\Delta\xi$ and $\Delta\eta$ were obtained, while the independence from rotations is not so clear. We will demonstrate this independence later on, and it will become clear that the independence is obscured by the poor resolution of our ellipse. The measure for resolution is the number of points of the contour, which is 72 in Fig. 14a. For comparison, the eye can resolve about 0.1 mm. Hence, 75 points would imply an ellipse with a circumference of 7.5 mm. The eye would have trouble to decide whether such a small structure is an ellipse. As a result of this small number of points, our results will not

Table 6. Scanning the contour of the rectangle of Fig.13.

location of scanning pattern		observed contour operator	recognized new point	command to advance		angle	line element	curvature
ξ	η		n	$\Delta\xi$	$\Delta\eta$	$\Delta\gamma(n)$	$\Delta s(n)/\Delta x$	$\Delta\gamma(n)\Delta x/\Delta s(n$
5	1	1B	1	+1		$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
6	1	2B	2	+1		$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
7	1	1C	-		+1	-	-	-
7	2	3B	3	+1		$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
8	2	2B	4	+1		0	1	0
9	2	2B	5	+1		$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
10	2	1C	-		+1	-	-	-
10	3	3B	6	+1		$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
11	3	2B	7	+1		0	1	0
12	3	2B	8	+1		$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
13	3	1C	-		+1	-	-	-
13	4	3B	9	+1		$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
14	4	2B	10	+1		0	1	0
15	4	2B	11	+1		$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
16	4	1C	-		+1	-	-	-
.								
.								
.								
28	9	3B	24	+1		$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
29	9	2B	25	+1		0	1	0
30	9	2B	26	+1		$+\pi/2$	1	1.5708
31	9	1C	-		+1	-	-	-
31	10	2C	27		+1	$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
31	11	1D	-	-1		-	-	-
30	11	3C	28		(+1)	$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
30	12	2C	29		+1	0	1	0
30	13	2C	30		+1	$+\pi/4$	$(1+\sqrt{2})/2$	0.6506
30	14	1D	-	-1		-	-	-
29	14	3C	31		(+1)	$-\pi/4$	$(1+\sqrt{2})/2$	-0.6506
29	15	2C	32		+1	0	1	0

look very good, but the only way to improve them is to increase the resolution by increasing the number of points by an order of magnitude. This is no problem if the methods derived here are implemented by machines, but our drawings and numerical examples would become unmanageable with such a large number of points.

At this time, we want to turn to the problem of size or scaling. Consider an enlargement or a reduction of the "ellipse" represented by the 72 points in Fig. 14a. What would be enlarged or reduced would be the distances between the points, but their number would remain unchanged. On the other hand, it is quite evident from Fig. 6 that a larger ellipse would yield more points rather than larger distances between the same number of points. Such a problem does not occur in differential mathematics since all curves have there nondenumerably many points, and as a result there are no gaps or distances between the points.

In order to make Fig. 14a scalable, we have to replace the finite set of points by a nondenumerable set. This is readily done by connecting adjacent points by straight lines as shown in Fig. 14. Let us choose the coordinates of a point n arbitrarily to be ξ_0, η_0. The coordinates of the point n+1 are then $\xi_0 + \Delta\xi$, $\eta_0 + \Delta\eta$. The values $\Delta\xi$ and $\Delta\eta$ come for any pair of points n, n+1 from Tables 1 or 6. The equation of the line between the two points is

$$\eta = \frac{\Delta\eta}{\Delta\xi} (\xi - \xi_0) + \eta_0 \qquad (25)$$

Using this equation with arbitrary initial values ξ_0, η_0 for the point n and arbitrary direction of the ξ-axis - the η-axis being perpendicular and positive oriented - a computer can plot Fig. 14b using the values of n, $\Delta\xi$ and $\Delta\eta$ in Table 1. Figure 14b can be scaled in the usual meaning of this term since the gaps between the points have been filled by lines with nondenumerably many points. We have now a continuous curve with a discontinuous first derivative.

A human observer looking at Fig. 14b and knowing that it represents a contour on a square field will immediately smooth out the indentations at the points n=1, 14, 20, ... to obtain the smoothe contour of Fig. 15. This is still a continuous curve with discontinuous first derivative, but it is definitely better than Fig. 14b. Mathematically speaking, Fig. 15 is obtained by turning a straight line around every point in a positive sense. How can one automate this process with the values in Table 1?

We first rewrite Table 1 in a concentrated form. We do no longer need the "location of the scanning pattern" and the "observed contour point". We need n, $\Delta\xi$, $\Delta\eta$ and $\Delta\gamma$. Since the ellipse of Fig. 6 contains 72 points, the values of $\Delta\xi, \Delta\eta$ and $\Delta\gamma$ repeat after point 36. Hence, we list $\Delta\xi, \Delta\eta$ and $\Delta\gamma/2\pi$, n=1 ... 36 in Table 7. Furthermore, we list in column 5 the sum of $\Delta\gamma/2\pi$ taken from n=1 to n. We see that this sum increases at the points $n=n_c$ listed in column 6. A comparison with Fig. 15 shows that these are just the points needed for the construction of this curve. One may, of course, just as well say that the points n_c are the one distinguished by the positive values of $\Delta\gamma/2\pi$ in column 4.

Next we need the angles between the straight lines at the points $n=n_c$ in Fig. 15. Refer to Fig. 16. On top is shown a typical contour of an opaque area on a square field. The polygon approximation we want is shown by the dashed line through the points n=1, 2, 5, 7, 10, 11, 12. Figure 16b shows the contour points alone. The dashed-dotted line connecting all points corresponds to the approximation of Fig. 14b. The angles at the points n are denoted $\Delta\gamma(n)$ and their curve

$$\gamma(n) = \sum_{i=1}^{n} \Delta\gamma(i) \tag{26}$$

is plotted in Fig. 16c. The angles of the dashed lines with a certain reference line - typically the ξ-axis - in Fig. 16b are denoted $\varepsilon(n) = \varepsilon(n_c)$. The difference of the angles in a point n is denoted $\Delta\varepsilon(n)$. If we denote a certain <u>corner point</u> n_c with n_{c1} and the next larger one with n_{c2}, we obtain

$$\Delta\varepsilon(n_{c1}) = \varepsilon(n_{c2}) - \varepsilon(n_{c1}) \tag{27}$$

This relationship is quite evident in Fig. 16b for $n_{c1} = 5$ and $n_{c2} = 7$. The positive jumps of $\gamma(n)$ at the corner points

Table 7. Computation of the approximation by a continuous curve with positive turning tangent and discontinuous or continuous first derivative for the ellipse of Fig.6.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
n	$\Delta\xi$	$\Delta\eta$	$\frac{\Delta\gamma}{2\pi}$	$\sum\frac{\Delta\gamma}{2\pi}$	n_0	X	Y	$\Delta s(n_0)$	ε	$\Delta\varepsilon$	s	s*	$\Delta\hat{s}^*(n)$	$\varkappa^*(n)$
1	+1		+1/8	1/8	1	9	0	9	0	0.322	0	0	3.896	0.083
2	+1		0	1/8										
3	+1		0	1/8										
4	+1		0	1/8										
5	+1		0	1/8										
6	+1		0	1/8										
7	+1		0	1/8										
8	+1		0	1/8										
9	+1		0	1/8										
10	+1	+1	+1/8	2/8	10	3	1	3.162	0.322	0.322	9	11.09	3.896	0.083
11	+1		-1/8	1/8										
12	+1		0	1/8										
13	+1	+1	+1/8	2/8	13	2	1	2.236	0.464	0.142	12.16	14.98	2.755	0.052
14	+1		-1/8	1/8										
15	+1	+1	+1/8	2/8	15	4	4	5.657	0.785	0.321	14.40	17.47	2.755	0.118
16	+1	+1	0	2/8										
17	+1	+1	0	2/8										
18	+1	+1	0	2/8										
19		+1	+1/8	3/8	19	1	2	2.236	1.107	0.322	20.06	24.72	2.755	0.118
20	+1	+1	-1/8	2/8										
21		+1	+1/8	3/8	21	0	5	5	1.571	0.464	22.29	27.46	2.755	0.171
22		+1	0	3/8										
23		+1	0	3/8										
24		+1	0	3/8										
25		+1	0	3/8										
26	-1	+1	+1/8	4/8	26	-1	2	2.236	2.034	0.463	27.29	33.62	2.755	0.171
27		+1	-1/8	3/8										
28	-1	+1	+1/8	4/8	28	-4	4	5.657	2.356	0.322	29.53	36.38	2.755	0.118
29	-1	+1	0	4/8										
30	-1	+1	0	4/8										
31	-1	+1	0	4/8										
32	-1		+1/8	5/8	32	-2	1	2.236	2.678	0.322	35.18	43.35	2.755	0.118
33	-1	+1	-1/8	4/8										
34	-1		+1/8	5/8	34	-3	1	3.162	2.820	0.142	37.42	46.11	2.755	0.052
35	-1		0	5/8										
36	-1	+1	-1/8	4/8										
37	-1		+1/8	5/8	37	-9	0	9	3.142	0.322	40.58	50.00	3.896	0.083

n_c = 2, 5, 7, 10 is also evident.

The value of $\varepsilon(n)$ follows directly from an inspection of Fig. 17 as

$$\varepsilon(n) = \arctan Y/X \tag{28}$$

X is the number of steps $\Delta\xi$ in the direction ξ and Y is the number of steps $\Delta\eta$ in the direction η from the corner point $n=n_c=n_{c1}$ to the following corner point $n=n_{c2}$. The sums X and Y of $\Delta\xi$ and $\Delta\eta$ are listed in columns 6 and 7 of Table 7 for the ellipse of Fig. 6.

We see further from Fig. 17 that the distance $\Delta s(n_c)/\Delta x$ from one corner point to the next equals

$$\Delta s(n_c)/\Delta x = (X^2 + Y^2)^{1/2} \tag{29}$$

To simplify the notation we will generally not write the distance unit Δx, and thus simply write $\Delta s(n)$ for $\Delta s(n)/$

The values of $\Delta s(n)/\Delta x = \Delta s, \varepsilon(n)$ and $\Delta\varepsilon(n)$ are listed in columns 9-11 of Table 7.

We still need the total length of the contour of Fig. 15 from the point n=1 to any corner point $n=n_c$ = 10, 13, ... It follows from Eq.(29)

$$s(n_c) = \sum_{i=1}^{n=n_c-1} \Delta s(i) \tag{30}$$

n_c = 1, 10, 13, 15 ....

For n_c=1 the upper limit of this sum equals zero and there is thus no term $\Delta s(i)$ in the sum; hence, s(1)=0. For n_c=10 we have the upper limit n=9, and the only term in the sum is $\Delta s(1)$ = 9, which yields s(10) = 9. The values of $s(n_c)$ are listed in column 12 of Table 7. The largest value, s(37) = 40.58, is half the circumference of the approximated ellipse.

In order to scale all possible ellipses to the same reference size, we multiply their circumference with a factor that makes it 100 or rather $100\Delta x$. The number 100 is arbitrary,

and it is chosen because it is small enough to be manageable. For machine operation, one would choose a much larger number such as 1000 or 10000. It should be as large or larger as the largest number of points of a contour in a square field such as Figs. 6 and 13, in order to avoid a loss of information by scaling.

The normalized length $s^*(n_c)$ between the corner points follows from the relation

$$s^*(n_c) = (50/40.58)\ s(n_c) \tag{31}$$

and it is listed in column 13 of Table 7.

6. CONTINUOUS FIRST DERIVATIVE, DISCONTINUOUS SECOND DERIVATIVE

We have developed enough mathematics for the automated plotting of the approximation of an ellipse according to Fig. 15 from the observed contour operators of Table 1. We could now start discussion how such a plot could be recognized by a computer in the sense that the printout "ellipse with axis ratio p and circumference S recognized" is produced rather than the plot of Fig. 15. However, we will not do so yet for the following reason. We know from experience that the eye is very sensitive to breaks in a straight line, which means the angles in Fig. 15. Mathematically speaking, a discontinuity of the first derivative is readily spotted. This is not all. The eye is also sensitive to a discontinuity of the curvature. This is an experience gained by carefully drafting curves for publication. A jump of the tangent in a point, which is a discontinuity of the first derivative, is a glaring deficiency. A jump of the curvature obtained, e.g., by joining two circular sections with different radius but the same tangent at the junction, is less glaring but still noticeable. Hence, we will carry the theory from the continuous curve with discontinuous first derivative to a continuous curve with continuous first and second derivative before we turn to the recognition and classification of the curve. According to Eq.(21) one needs a continuous first

and second derivative to obtain a continuous curvature.

Let us observe at this time that the straight line approximation of Fig. 15 is already sufficient to recognize all contours consisting of straight lines, such as the rectangle in Fig. 13. To show this more clearly, we write Table 8 for the rectangle of Fig. 13 in analogy to Table 7 for the ellipse of Fig. 6. The interesting values are $\Delta\varepsilon$ and $s(n_c)$. With them, derived for all points n=1 ... 76 rather than only the part shown in Table 8, one can construct Fig. 18. The rotation of Fig. 13 is removed, and there are clearly four straight lines. The corners do not show one angle $\Delta\varepsilon=\pi/2$ but either two or three angles that sum to $\pi/2$:

$\Delta\varepsilon(26)=1.2490$, $\Delta\varepsilon(27)=0.3218$; $\Delta\varepsilon(26)+\Delta\varepsilon(27)=\pi/2$
$\Delta\varepsilon(76)=0.4636$, $\Delta\varepsilon(1)=0.7854$, $\Delta\varepsilon(2)=0.3218$; $\Delta\varepsilon(76)+\Delta\varepsilon(1)+\Delta\varepsilon(2)=\pi/2$

There is no particular problem for a machine to recognize a rectangle from the numbers in columns 11 and 12 of Table 8, particularly if the circumference of the rectangle contains an order of magnitude more points than the one in Fig. 13.

Let us return to the approximation of the ellipse in Fig. 15. If we used an order of magnitude more points we would increase the number of corner points where $\Delta\varepsilon$ is not zero, in contrast to the rectangle of Fig. 18. We would take this large number of corner points as a reason to decide that the contour does not consist of straight lines but is curved. Since man-made curved contours are primarily circles, we will use sections of circles to smooth the corners in Fig. 15. There are many ways of doing so; we chose the one shown in Fig. 19.

The three points n-1, n and n+1 are corner points in our terminology. The distance $\Delta s(n)$ is smaller than the distance $\Delta s(n-1)$. A circle is drawn from the center of the line between n and n+1 to the point on the line from n-1 to n having the distance $\Delta s(n)/2$ from n. The lines from n to n+1 and from n-1 to n are the tangents of the circle at the points where the lines meet the circle. This insures a continuous first

Table 8. Computation of the approximation by a continuous curve with positive turning tangent and discontinuous first derivative for the rectangle of Fig.3.

1	2	3	4	5	6	7	8	9	10	11	12	13
n	$\Delta\xi$	$\Delta\eta$	$\frac{\Delta\gamma}{2\pi}$	$\sum\frac{\Delta\gamma}{2\pi}$	n_c	X	Y	Δs	ε	$\Delta\varepsilon$	$s(n_c)$	$s^*(n_c)$
25		-1	0	6/8								
26	+1	-1	+1/8	7/8	76	1	-1		2.3562	0.4636	78.008	100
1	+1		+1 8	1/8	1	1	0	1	0	0.7854	0	0
2	+1	+1	+1/8	2/8	2	3	1	3.162	0.3218	0.3218	1	1.282
3	+1		-1/8	1/8								
4	+1		0	1/8								
5	+1	+1	+1/8	2/8	5	3	1	3.162	0.3218	0		
6	+1		-1/8	1/8								
7	+1		0	1/8								
8	+1	+1	+1/8	2/8	8	3	1	3.162	0.3218	0		
9	+1		-1/8	1/8								
10	+1		0	1/8								
11	+1	+1	+1/8	2/8	11	3	1	3.162	0.3218	0		
12	+1		-1/8	1/8								
13	+1		0	1/8								
14	+1	+1	+1/8	2/8	14	3	1	3.162	0.3218	0		
15	+1		-1/8	1/8								
16	+1		0	1/8								
17	+1	+1	+1/8	2/8	17	3	1	3.162	0.3218	0		
18	+1		-1/8	1/8								
19	+1		0	1/8								
20	+1	+1	+1/8	2/8	20	3	1	3.162	0.3218	0		
21	+1		-1/8	1/8								
22	+1		0	1/8								
23	+1	+1	+1/8	2/8	23	3	1	3.162	0.3218	0		
24	+1		-1/8	1/8								
25	+1		0	1/8								
26		+1	+1/4	3/8	26	0	0	1	1.5708	1.2490	26.296	33.709
27	-1	+1	+1/8	4/8	27	-1	3	3.162	1.8925	0.3218	27.294	34.991
28		+1	-1/8	3/8								
29		+1	0	3/8								
30	-1	+1	+1/8	4/8	30	-1	3	3.162	1.8925	0		
31		+1	-1/8	3/8								
32		+1	0	3/8								
33	-1	+1	+1/8	4/8	33	-1	3	3.162	1.8925	0		
34		+1	-1/8	3/8								
35		+1	0	3/8								
36	-1	+1	+1/8	4/8	36	-1	3	3.162	1.8925	0		
37		+1	-1/8	3/8								
38		+1	0	3/8								
39	-1		+1/4	5/8	39	-2	0	2	3.1416	1.249	39.944	51.077
40	-1		0	5/8								
41	-1	-1	+1/8	6/8	41	-3	-1	3.162	3.4633	0.3218	41.944	53.769

derivative. The radius $\rho(n)$ of the circle follows from Fig. 19:

$$\rho(n) = \frac{\Delta\hat{s}(n)}{2 \tan[\Delta\varepsilon(n)/2]} \qquad (32)$$

$$\Delta\hat{s}(n) = \Delta s(n) \text{ for } \Delta s(n) \leqq \Delta s(n-1)$$
$$\Delta\hat{s}(n) = \Delta s(n-1) \text{ for } \Delta s(n) \leqq \Delta s(n-1)$$

The curvature $\varkappa(n)$ is defined as the reciprocal of $\rho(n)$:

$$\varkappa(n) = 1/\rho(n) \qquad (33)$$

We could draw circles according to Fig. 19 to smooth the corners of the polygon of Fig. 15, but - apart from the labor involved - the approximation would become already so good that deviations would be difficult to recognize. Hence, we use a different representation from here on that not only is easier to plot but will conspicuously show deviations even at the last step when the second derivative is made continuous.

Figure 20 shows on top the sum $\sum\Delta\gamma(n)/2\pi$ according to column 5 of Table 7. The second row shows $\Delta\gamma(n)/2\pi$ according to column 4, and the third row shows $\Delta\varepsilon(n)$ according to column 11. The fourth row shows $\varkappa^*(n)=(40.58/50)\varkappa(n)$ according to Eqs.(32) and (33). The star indicates that the curvature holds for a curve normalized to the length $100\Delta x$ while the length of the curve in Fig. 15 is $2x40.58\Delta x = 81.16\Delta x$.

Refer to Table 7 for the computation of $\varkappa^*(n)$. First, $\Delta\hat{s}(n)$ is determined and multiplied by 50/40.58 to produce the normalized length $\Delta s^*(n_c)$ listed in column 14. The normalized curvature $\varkappa^*(n)$ follows then from Eqs.(32) and (33). Referring now to Fig. 19 we see that the curvature calculated for the point n at the distance $s^*(n)$ from the arbitrarily chosen beginning of the curve, also applies to the interval with distance $\pm\Delta\hat{s}^*(n)/2$ from the location $s^*(n)$. We may write for this interval:

$$s^*(n)-\Delta\hat{s}^*(n)/2 \leqq s^* \leqq s^*(n)+\Delta s^*(n)/2 \qquad (34)$$

Note that $s^*(n)$ means the normalized distance of the point n from the point 1 measured along the polygon of Fig. 15, while s^* means any distance from the point 1. The difference is that $s^*(n)$ is only defined for the corner points n_c, while there is no such restriction on s^*. Furthermore, $s^*(n)$ is a function of the independent variable n, while s^* is the new independent variable.

The intervals in which the values of $\varkappa^*(n)$ apply are shown by dashed lines in Fig. 20, row 4. For instance, the value $\varkappa^*(n)=0.083$ in the point $s^*(n_c)=11.09$ applies to the range from $11.09-3.896/2 = 9.14$ to $11.09 + 3.896/2 = 13.04$. This range is indicated in Fig. 20 by the dashed line from 9.14 to 13.04. The value of this line is given as $0.083q$ for the following reason. For a closed curve the total rotation of the tangent between a point and the same point after advancing around the curve equals 2π, as may be inferred from Fig. 21; for curves without continuous derivative one must not only start and stop at the same point but use the same tangent (right or left tangent) at the start and the stop. For the usual definition of curvature as $d\varepsilon/ds$ we obtain the "total curvature" over the length S of a closed curve

$$\int_0^S \frac{d\varepsilon}{ds}\, ds = 2\pi \qquad (35)$$

Our definition of curvature in Eqs.(32) and (33) must also yield the total curvature 2π for a closed curve. We have arbitrarily assigned the normalized length 100 or $100\Delta x$ to all our curves, which implies that we have a value for the curve, its derivatives, and its curvature at the points $s^*=0, 1, \ldots, 100$. Hence we show in Fig. 20, row 4, the value $0.083q$ for the curvature at the points $s^*=10, 11, 12, 13$. Correspondingly, the value $0.052q$ is assigned to $s^*=14, 15, 16$; etc. The value of q follows from the requirement that the total curvature must equal 2π over its total length $S=100\Delta x$. From Fig. 20 - which shows only half the length S/2 - follows:

$$2\big[2 \times 0.083 + 4 \times 0.083 + 3 \times 0.052 + 3 \times 0.118 + 3 \times 0.118 + 2 \times 0.171 + 2 \times 0.171 + 3 \times 0.118 + 3 \times 0.118 + 3 \times 0.052 + 2 \times 0.083\big]q = 2\pi$$

$$q = 1.02 \qquad (36)$$

Note that the curvature in the point 0 as well as in the point 100 has to be included, otherwise one would use the right tangent at point 0 but the left tangent at the point 100, which coincides with point 0.

7. CONTINUOUS SECOND DERIVATIVE

By rounding the corners in Fig. 15 by means of circles according to Fig. 19 we have made the curve and its first derivative continuous. The curvature - or the second derivative - are not yet continuous. One may readily see from Fig. 19 that the points on the circular section have the curvature $\kappa^*=1/\rho^*$, while the points beyond the circular section have the curvature $\kappa^*=0$.

In order to make the second derivative and the curvature continuous we must develop an approximation that avoids the discontinuity of the curvature of Fig. 19, but which still is based on Fig. 19. The reason is that the approximation of Fig. 19 will detect circular contours, just as the approximation by straight lines detected straight-line contours. The problem does not have a unique solution. We reduce the range of possible solutions by further requiring that the approximation should be easy to implement by computer.

Before proceeding, we must define the normalized length of the curve $s^*=s^*(n_c)$ in a way that is not changed every time the approximation is improved, but still identifies a point unambiguously. In Figs. 15 and 18 the length of the contour was measured along the solid line connecting the corner points n_c. The approximation of Fig. 19 rounded the corners and thus shortened the length of the contour, and any further improved approximation will change the length of the contour again. According to Fig. 22 we can use the distance s^* measured along the straight-

line approximation by connecting points s_1^*, s_2^*, ... on the straight lines with the centers P_1, P_2, ... of the circles of curvature in the points s_{1c}^*, s_{2c}^*, ... on the curved line. This requires, of course, that the centers P_1, P_2, ... are known before s_{1c}^*, s_{2c}^*, ... are known. This requirement is satisfied in Fig. 19, since the center of the circle smoothing the break at the point n is constructed solely with the help of the straight lines from n-1 to n and from n to n+1. Hence, we have the additional requirement that the points P_1, P_2, ... in Fig. 22 must be determined by the straight lines from n-1 to n and from n to n+1.

The approximation of Fig. 19 leaves a straight section with curvature 0 at either the longer one of the lines from n-1 to n and from n to n+1, or on both lines. We look first for a method to eliminate these straight sections. According to Fig. 23 we may do so by determining first a radius $\rho_1(n)=\rho(n)$ according to Fig. 19, and then define a variable value for $\rho(s)$ by the equation

$$\rho(s) = \rho_1(n) + H\left[s-s(n-1) - \Delta s(n-1)/2\right] \tag{37}$$

where H is a yet to be determined constant, and $s-s(n-1) - \Delta s(n-1)/2$ is the distance to the point s measured from the point $s(n-1) + \Delta s(n-1)/2$. For $s=s(n+1)$ one obtains $\rho(s)=\rho(n+1)=\rho_2(n)$ and a relation for H:

$$H = \frac{2\left[\rho_2(n) - \rho_1(n)\right]}{\Delta s(n-1) + \Delta s(n)} \tag{38}$$

If the angle δ is small compared with $\varepsilon(n)$ one obtains all the relations necessary to determine $\rho_1(n)$ and $\rho_2(n)$:

$$\rho_1(n) = \Delta s(n)/2 \tan\left[\Delta\varepsilon(n)/2 - \delta\right] \tag{39}$$
$$\rho_2(n) = \Delta s(n)/2 \tan\left[\Delta\varepsilon(n)/2 + \delta\right]$$
$$\delta = \frac{\Delta s(n-1) - \Delta s(n)}{\Delta s(n-1) + \Delta s(n)} \tan\left[\Delta\varepsilon(n)/2\right]$$

There are two drawbacks to this method. First, we had to make the assumption $\delta \ll \Delta\varepsilon(n)$ to avoid the solution of a transcendental equation, which introduces either an avoidable inaccuracy or calls for a lengthy solution of a transcendental equation by computer. A second drawback is even more severe. Let $\Delta\varepsilon(n)$ in Fig. 23 equal $\pi/2$. We will then evidently obtain $\rho_1(n) = \Delta s(n)/2$ and $\rho_2(n) = \Delta s(n-1)/2$. Hence, the shorter interval $\Delta s(n) < \Delta s(n-1)$ will get the larger radius $\rho_2(n) > \rho_1(n)$ and thus the smaller curvature $1/\rho_2(n) < 1/\rho_1(n)$ in its center. This is just the opposite of what we would want for a good approximation.

A second method to eliminate the straight sections left by the approximation according to Fig. 19 is shown in Fig. 24. First, the curvature radius and the center of the circle of curvature is determined for every corner point n, n+1, n+2, ... This yields $\rho(n)$, $\rho(n+1)$, $\rho(n+2)$,.... and the centers P(n), P(n+1), P(n+2), ... of the circles. These centers are connected by the dashed lines with length $\Delta d(n)$, $\Delta d(n+1)$, ... The center of the radius of curvature of an intermediate point, located, e.g., between n and n+1 and having the distance $q\Delta s(n)$ from n, would then be located on the line from P(n) to P(n+1) at the distance $q\Delta d(n)$ from P(n); the radius of curvature $\rho(n,q)$ would also change linearly with q from $\rho(n)$ to $\rho(n+1)$:

$$\rho(n,q) = (1-q)\,\rho(n) + q\,\rho(n+1) \qquad (40)$$

This method would yield a continuously varying curvature and thus a continuous second derivative. Its drawback is that the radius of curvature $\rho(n,q)$ could never be larger than either $\rho(n)$ or $\rho(n+1)$. A look at Fig. 15 shows that this means the curvature anywhere along the straight line between n=1 and n=10 could not be less than in the points n=1 or n=10. This feature makes this approach to a continuous second derivative undesirable.

A third extension of Fig. 19 is shown in Fig. 25. It is assumed that $\Delta s(n-1)$ is smaller than $\Delta s(n)$. The verticals in

the center of the line from point n-1 to point n, and at the distance $\Delta s(n-1)/2$ from point n on the line from point n to point n+1 yield the point P(n) and the radius $\rho(n)$. The vertical in the center of the line from point n to point n+1 yields then the point P'(n) and the radius $\rho'(n)$:

$$\rho'(n) = \rho(n) + \Delta s'(n)/\tan \Delta\varepsilon(n) \qquad (41)$$

A radius $\rho(n,d)$ at the point with distance d from the center of the line from point n-1 to point n, that varies continuously from $\rho(n)$ to $\rho'(n)$, may be defined in analogy to Eq.(40):

$$\rho(n,d) = (1-q)\ \rho(n) + q\rho'(n) \qquad (42)$$

$$q = \frac{d}{\Delta s(n-1) + \Delta s'(n)}$$

Substituting q and introducing the notation $\Delta\hat{s}(n)$ to include the case where $\Delta s(n-1) > \Delta s(n)$ rather than $\Delta s(n-1) < \Delta s(n)$ as in Fig. 25, one obtains:

$$\rho(n,d) = \rho(s) = \rho(n) + d\Delta s'(n)/[\Delta\hat{s}(n) + \Delta s'(n)]\ \tan \Delta\varepsilon(n)$$

$$\Delta s'(n) = |\Delta s(n) - \Delta s(n-1)|/2 \qquad (42)$$

$$\Delta\hat{s}(n) = \Delta s(n-1),\ s_0(n) = s(n) - \Delta\hat{s}(n)/2 \quad \text{for} \quad \Delta s(n-1) \leqq \Delta s(n)$$
$$= \Delta s(n),\ s_0(n) = s(n) + \Delta\hat{s}(n)/2 \quad \text{for} \quad \Delta s(n-1) \geqq \Delta s(n)$$

$$d = |s - s_0(n)|$$

It is readily apparent that the construction of Fig. 25 eliminates all flat spots along a polygon as shown in Fig. 15. The radius $\rho'(n)$ can be arbitrarily much larger than the radius $\rho(n)$, if $\Delta s(n)$ is arbitrarily much larger than $\Delta s(n-1)$. The curves constructed in this way will have a continuously varying curvature, except in the center of the lines between the points n-1 to n, n to n+1, etc. These points with a discontinuous second

derivative are a drawback, but in our finite mathematics one can readily overcome this drawback. Let us explain the whole process by means of an example.

Table 9 lists the values n, s^*, $\Delta\varepsilon$, $\Delta s^*(n)$, $\Delta s^*(n-1)$, $\Delta\hat{s}^*(n)$ and $\rho^*(n) = 1/\varkappa^*(n)$ taken from Table 7. Note that a star characterizes values applying to the normalized length 100 of the contour rather than the actual length 81.16 of the polygon shown by the solid line in Fig. 15:

$$\Delta s^*(n) = (100/81.16)\, \Delta s(n) \tag{46}$$

The values of s^* in Table 7 were only listed for the corner points n_c = 1, 10, 13, ... while a value of s^* is listed for every point n in Table 9. These intermediate values are produced by linear interpolation. Let $s^*(n_{c1})$ be the distance to the corner point n_{c1}, and $s^*(n_{c2})$ the distance to the next corner point n_{c2}. The distance $s^*(n)$ to a point n between n_{c1} and n_{c2} is defined by:

$$s^* = s^*(n) = s^*(n_{c1}) + \frac{s^*(n_{c2}) - s^*(n_{c1})}{n_{c2} - n_{c1}} (n-n_{c1}) \tag{44}$$

$$n_{c1} \leqq n \leqq n_{c2}$$

The value of $\rho^*(n)$ in Table 9 is the reciprocal of $\varkappa^*(n)$ in Table 7. The values of $\Delta s'^*(n)$ and s_0^* follow from Eq.(43). The distances d are divided into distances d_1 increasing with n and distances d_2 decreasing with n:

$$d_1^* = s^* - s_0^*, \quad d_2^* = s_0^* - s^* \tag{45}$$

The radius of curvature $\rho^*(n,d)$ follows from Eq.(43) and the curvature $\varkappa^*(n,d)$ is its reciprocal. We still have to normalize $\varkappa^*(n,d)$ in analogy to Eq.(36):

$$2q \sum_{n=1}^{36} \varkappa^*(n,d) = 5.87q = 2\pi \tag{47}$$

$$q/2\pi = 0.170$$

$$\varkappa_0^*/\pi = 0.170\ \varkappa^*(n,d)$$

Table 9. Computation of the approximation of the polygon of Fig.15 by a curve according to Fig.25 with curvature larger than zero in every point.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
n	s^*	$\Delta\varepsilon$	$\Delta s^*(n)$	$\Delta s^*(n-1)$	$\Delta\hat{s}^*(n)$	$\rho^*(n)$	$\Delta s'^*(n)$	s_o^*	d_1^*	d_2^*	$\rho^*(n,d)$	$\varkappa^*(n,d)$	$\varkappa_o^*/2\pi$
1	0	0.322	11.089	3.896	3.896	11.995	3.587	-1.95	1.95		14.80	0.0676	0.0115
2	1.23								3.18		16.57	0.0603	0.0103
3	2.49								4.41		18.34	0.0545	0.0093
4	3.70								5.65		20.12	0.0497	0.0085
5	4.93								6.88		21.89	0.0457	0.0078
6	6.16									6.88	21.89	0.0457	0.0078
7	7.39									5.65	20.12	0.0497	0.0085
8	8.62									4.42	18.35	0.0545	0.0093
9	9.86									3.18	16.57	0.0603	0.0103
10	11.09	0.322	3.896	11.089	3.896	11.995	3.597	13.04		1.95	14.80	0.0676	0.0115
11	12.38									0.66	12.94	0.0773	0.0132
12	13.69									2.67	22.58	0.0442	0.0075
13	14.98	0.142	2.755	3.896	2.755	19.369	0.571	16.36		1.38	21.03	0.0476	0.0081
14	16.36								0	0	13.93	0.0718	0.0122
15	17.74	0.321	6.970	2.755	2.755	8.482	2.108	16.36	1.38		10.27	0.0973	0.0166
16	19.48								3.12		12.53	0.0798	0.0136
17	21.23								4.87	4.85	14.79	0.0676	0.0115
18	22.97									3.11	12.52	0.0799	0.0136
19	24.72	0.322	2.755	6.970	2.755	8.482	2.108	26.08		1.36	10.23	0.0976	0.0166
20	26.08								0	0	7.16	0.1397	0.0238
21	27.46	0.464	6.161	2.755	2.755	5.831	1.703	26.08	1.38		6.88	0.1453	0.0248
22	28.70								2.62		7.83	0.1277	0.0218
23	29.93								3.85		8.77	0.1140	0.0194
24	31.16									3.84	8.77	0.1140	0.0194
25	32.39									2.61	7.83	0.1277	0.0218
26	33.62	0.463	2.755	6.161	2.755	5.831	1.703	35.00		1.38	6.88	0.1453	0.0248
27	35.00								0	0	7.16	0.1397	0.0238
28	36.38	0.322	6.970	2.755	2.755	8.482	2.108	35.00	1.38		10.23	0.0976	0.0166
29	38.12								3.12		12.52	0.0799	0.0136
30	39.87								4.87	4.86	14.79	0.0676	0.0115
31	41.61									3.12	12.53	0.0789	0.0136
32	43.35	0.322	2.755	6.970	2.755	8.482	2.108	44.73		1.38	10.27	0.0973	0.0166
33	44.73								0	0	13.93	0.0718	0.0122
34	46.11	0.142	3.896	2.755	2.755	19.369	0.571	44.73	1.38		21.03	0.0476	0.0081
35	47.40								2.67		22.58	0.0442	0.0075
36	48.71								0.66		12.94	0.0773	0.0132
37	50.00	0.322	11.089	3.896	3.896	11.995	3.597	48.05	1.95		14.80	0.0676	0.0115

The normalized curvature $\varkappa_0^*/2$ is listed in column 14 of Table 9. It is also shown by the dots in the lowest row of Fig. 20.

In order to obtain $\varkappa_0^*/2$ not as function of the points n but of the normalized distance s* along the curve for s* = 0, 1, 2, ..., 100 - or rather s* = 0, Δx, $2\Delta x$, ..., $100\Delta x$ - we must plot $\varkappa_0^*/2\pi$ for the integer values of s* rather than the integer values of n in Fig. 20. There are many ways to do so. The simplest one is to give $\varkappa_0^*(s^*)$ the same value as $\varkappa_0^*(n)$ has for the value of n closest to s*. The values of $\varkappa_0^*/2\pi$ plotted for s* = 0, 1, ... in Fig. 20 have been chosen in this way. This method of determination of $\varkappa_0^*(s^*)$ from $\varkappa_0^*(n)$ is shown in more detail in Fig. 26a.

A more sophisticated method that yields a <u>continuous</u> function $\varkappa_0^*(s^*)$ from the <u>samples</u> $\varkappa_0^*(n)$ is shown in Fig. 26b. The samples $\varkappa_0^*(n)$ for n = 1, 2, ..., m, ... are connected by the dashed line. This yields - except for a normalizing factor - $\varkappa_0^*(s^*)$ as a continuous function, and we have thus solved the problem to derive from the sampled function of Fig. 14a a continuous function with continuous first and second derivatives.

The normalizing factor for the continuous function $\varkappa_0^*(s^*)$ is, of course, zero. This causes no difficulty, since we will never use more than a finite number of samples of $\varkappa_0^*(s^*)$. The sample for s* = j in Fig. 26a follows readily from the formula for linear interpolation:

$$\varkappa_0^*(s^*) = \varkappa_0^*(m) + \frac{\varkappa_0^*(m+1) - \varkappa_0^*(m)}{s^*(m+1) - s^*(m)} \left[s^* - s^*(m) \right] \qquad (48)$$

One must be careful that another normalization is required after one makes the transition from $\varkappa_0^* = \varkappa_0^*(n)$ to $\varkappa_0^*(s^*)$. The ellipse of Fig. 6 has 72 points denoted n, but its normalized length equals $100\Delta x$, and there are thus 100 values $\varkappa_0^*(s^*)$ but only 36 values $\varkappa_0^*(n)$. Hence, we need another normalization coefficient q_0:

$$q_0 \sum_{s^*=0}^{99} \varkappa_0^*(s^*) = 2\pi \qquad (49)$$

The curvature with all normalizations is denoted $\varkappa_N$:

$$\varkappa_N = \varkappa_N(s^*) = q_0 \, \varkappa_0^*(s^*) \qquad (50)$$

Note that Eqs.(49) and (50) apply to the "step function" $\varkappa_0^*(s^*)$ of Fig. 26a, to the linearly interpolated function $\varkappa_0^*(s^*)$ of Fig. 26b, as well as to any function $\varkappa_0^*(s^*)$ derived from the values $\varkappa_0^*(n)$ by a higher order interpolation.

Either the step function $\varkappa_0^*(s^*)$ of Fig. 26a or the linearly interpolated function $\varkappa_0^*(s^*)$ of Fig. 26b can be used to recognize the shape of the contour in Fig. 6, independent of shift, rotation or scaling. If one looks for a contour of a particular given shape, one can cross-correlate $\varkappa_0^*(s^*)$ of the observed contour with $\varkappa_0^*(s^*)$ of the sample contours. This correlation process provides an averaging over all the observed values $\varkappa_0^*(s^*)$ and thus helps eliminate the influence of the square field in Fig. 6. In other cases one may want to classify the observed contour in a more general way than permitted by cross-correlation with sample functions. This will generally call for some averaging process to reduce further the influence of the square field in Fig. 6. We will show by means of an example how to proceed.

The integer values of s^* for one half of the observed ellipse are written in the first and sixth columns of Table 10. The values of $\varkappa_0^*/2\pi$ written for integer values of n but non-integer values of s^* in Table 9 are assigned to the integer values of s^* in Table 10 according to the staircase approximation of Fig. 26a. The average values $\overline{\varkappa}_0^*/2\pi = \overline{\varkappa}_0^*(s^*)/2\pi$ are obtained by averaging over the 11 values $\varkappa_0^*(s^*-5)/2\pi$ to $\varkappa_0^*(s^*+5)/2\pi$:

$$\overline{\varkappa}_0^*(s^*)/2\pi = \frac{1}{11} \sum_{i=-5}^{+5} \varkappa_0^*(s^*+i)/2\pi \qquad (51)$$

Summation of the 50 values of $\overline{\varkappa}_0^*(s^*)/2\pi$ for $s^* = 0 \ldots 49$ yields

Table 10. The curvature $\varkappa_0^*/2\pi$ for integer values of s^*, the curvature $\bar{\varkappa}_0^*/2\pi$ obtained by averaging over 11 values of $\varkappa_0^*/2\pi$, the averaged curvature normalized to $\bar{\varkappa}_N/2\pi$ according to Eq.(50), and the deviation $\bar{\varkappa}_N/2\pi-0.01$ of the averaged curvature from that of the circle; the values hold for the ellipse of Fig.6.

1	2	3	4	5	6	7	8	9	10
s^*	$\varkappa_0^*/2\pi$	$\bar{\varkappa}_0^*/2\pi$	$\bar{\varkappa}_N/2\pi$	$\bar{\varkappa}_N/2\pi-0.01$	s^*	$\varkappa_0^*/2\pi$	$\bar{\varkappa}_0^*/2\pi$	$\bar{\varkappa}_N/2\pi$	$\bar{\varkappa}_N/2\pi-0.01$
0	0.0115	0.0096	0.0068	-0.0032	25	0.0166	0.0180	0.0128	0.0028
1	0.0103	0.0092	0.0065	-0.0035	26	0.0238	0.0185	0.0132	0.0032
2	0.0093	0.0092	0.0065	-0.0035	27	0.0248	0.0195	0.0139	0.0039
3	0.0093	0.0093	0.0066	-0.0034	28	0.0248	0.0207	0.0147	0.0047
4	0.0085	0.0095	0.0068	-0.0032	29	0.0218	0.0217	0.0154	0.0054
5	0.0078	0.0092	0.0065	-0.0035	30	0.0194	0.0223	0.0159	0.0059
6	0.0078	0.0092	0.0065	-0.0035	31	0.0194	0.0223	0.0159	0.0059
7	0.0085	0.0095	0.0068	-0.0032	32	0.0218	0.0217	0.0154	0.0054
8	0.0085	0.0098	0.0070	-0.0030	33	0.0248	0.0207	0.0147	0.0047
9	0.0093	0.0096	0.0068	-0.0032	34	0.0248	0.0197	0.0140	0.0040
10	0.0103	0.0096	0.0068	-0.0032	35	0.0238	0.0187	0.0133	0.0033
11	0.0115	0.0100	0.0071	-0.0029	36	0.0166	0.0182	0.0129	0.0029
12	0.0132	0.0104	0.0074	-0.0026	37	0.0166	0.0177	0.0126	0.0026
13	0.0132	0.0111	0.0079	-0.0021	38	0.0136	0.0172	0.0122	0.0022
14	0.0075	0.0116	0.0083	-0.0017	39	0.0136	0.0164	0.0117	0.0017
15	0.0081	0.0120	0.0085	-0.0015	40	0.0115	0.0153	0.0109	0.0009
16	0.0122	0.0121	0.0086	-0.0014	41	0.0136	0.0139	0.0099	-0.0001
17	0.0122	0.0121	0.0086	-0.0014	42	0.0136	0.0130	0.0092	-0.0008
18	0.0166	0.0121	0.0086	-0.0014	43	0.0166	0.0122	0.0087	-0.0013
19	0.0136	0.0125	0.0089	-0.0011	44	0.0166	0.0122	0.0087	-0.0013
20	0.0135	0.0132	0.0095	-0.0005	45	0.0122	0.0120	0.0085	-0.0015
21	0.0115	0.0147	0.0105	0.0005	46	0.0081	0.0119	0.0085	-0.0015
22	0.0115	0.0159	0.0113	0.0013	47	0.0075	0.0115	0.0082	-0.0018
23	0.0136	0.0170	0.0121	0.0021	48	0.0075	0.0111	0.0079	-0.0021
24	0.0166	0.0175	0.0125	0.0025	49	0.0132	0.0104	0.0074	-0.0026

0.703. With the help of Eq.(49) one obtains:

$$q_0 \sum_{s^*=0}^{99} \bar{\varkappa}_0^* (s^*)/2\pi = 2 \times 0.703\ q_0 = 1 \qquad (52)$$

$$q_0 = 0.711$$

$$\bar{\varkappa}_N/2\pi = 0.711\ \bar{\varkappa}_0^*/2\pi$$

The values of $\bar{\varkappa}_N/2$ are listed in columns 4 and 9 of Table 10.

Instead of plotting $\bar{\varkappa}_N/2\pi$ we subtract first the curvature of a circle with length $s^* = 100$. From $2\pi R = 100$ follows the curvature $\varkappa_N = 1/R = 2\pi/100$, and $\varkappa_N/2\pi = 0.01$. This value 0.01 is subtracted in columns 5 and 10 of Table 10. The values $\bar{\varkappa}_N/2\pi - 0.01$ are plotted in Fig. 27a.

One may readily recognize a typical feature of the ellipse. For a short section of the length of the ellipse the curvature is much larger than that of a circle of equal length, while for a larger section the curvature is smaller than that of a circle. The magnitude of the positive peaks in Fig. 27a thus exceeds the magnitude of the negative peaks, but there are more negative peaks. The sum of all positive and negative values $\bar{\varkappa}_N/2 - 0.01$ is, of course, zero, since the total curvature of any closed contour equals 2π according to Fig. 21.

8. EXAMPLES OF CONTOURS

Let us consider a few examples. First, we rotate the ellipse of Fig. 6 into the position shown in Fig. 28. We then produce a table in analogy to Table 7, which yields the corner points n_c = 1, 9, 12, 14, 17, 20, 24, 31, 35, 37, 45, 48, 50, 53, 56, 60, 67, 71, as well as the other values corresponding to Table 7. The straight-line approximation of the ellipse is shown in Fig. 29. A comparison with Fig. 15 shows a similarity, but it is obvious that a much larger number of corner points would be needed to make Figs. 15 and 29 "approximately equal".

Figure 30 shows $\sum\Delta\gamma(i)$, $\Delta\gamma(n)$, $\Delta\varepsilon(n)$, $\varkappa^*(n)$ and $\varkappa_0^*/2\pi$ in

analogy to Fig. 20. Using the staircase approximation of Fig. 26a, just as it was done in Fig. 20, yields the normalized curvature averaged over 11 samples with the curvature of the circle subtracted out as shown in Fig. 27b. This Fig. 27b has been shifted relative to Fig. 27a so that the maxima at n=23 and n=59 in Fig. 27a line up with the maxima at n=31 and n=67 in Fig. 27b. The location of the maxima in terms of the length s* of the curve is $s_a^* = 30$ and $s_a^* = 80$ in Fig. 24a, and $s_b^* = 41$ and $s_b^* = 91$ in Fig. 27b. The difference $s_b^* - s_a^* = 11$ expresses the angle of rotation $\varphi(b)$ between the ellipses of Figs. 6 and 28. This relationship between $s_b^* - s_a^*$ and $\varphi(b)$ is shown in Fig. 31.

A much better way to obtain the angle $\varphi(b)$ is to cross-correlate the functions of Figs. 27a and b. The maximum of the cross-correlation function will occur for a certain shift between the scales s*, and this shift is used instead of the difference $s_b^* - s_a^*$ to determine $\varphi(b)$. Still better is to cross-correlate the functions $\varkappa_0^*/2\pi$ in Figs. 20 and 30.

Let us try next a larger ellipse as shown in Fig. 32. The number of points on its contour is 118 rather than 72 as in Figs. 6 and 28. The straight-line approximation in Fig. 33 looks much more like an ellipse than the one in Fig. 29. The functions in Fig. 34 are obtained in exactly the same way as the ones in Figs. 20 and 30, except that the linear interpolation of Fig. 26b rather than the staircase approximation of Fig. 26a is used for $\varkappa_0^*/2\pi$. The curvature averaged over 11 samples less the curvature of the circle is plotted in Fig. 27c. This plot is disappointing compared with the approximation of Fig. 33. However, one has to keep in mind that Fig. 27c uses the second derivative of a continuous curve approximating the ellipse, and the second derivative of a function is extremely sensitive to slight distortions of the function. We would need an order of magnitude more points to make the curvature of Fig. 27b look "smooth", that is independent of the square field on which the ellipse of Fig. 32 is plotted.

One can produce smoother looking curvatures by averaging over more than 11 values. The average of 25 values of $\varkappa_0^*/2\pi$

in Fig. 20 is used - with another minor modification - in Fig. 27d. This plot is much smoother than the one of Fig. 27a which uses the average of 11 values only. However, Fig. 27a is better since it shows that there are fewer positive than negative samples, and that they have a larger magnitude. This characteristic of an ellipse is lost in Fig. 27d.

As the next example we analyze the circular area of Fig. 35. The values obtained in analogy to Table 7 are listed in Table 11. Due to the symmetry of the circle we need to list only one quarter of the points on the contour - that is 76/4 = 19 - rather than one half the points as in the case of the ellipse. One may readily verify that the circular line of Fig. 36 yields the same values, because the contour operators of Fig. 7 do not distinguish between the contour of an area and a line. This is so because a line is essentially a narrow area in the mathematics on a square field. One can, of course, readily make a distinction by using contour operators with more than 2 x 2 points.

Figure 37 shows the straight-line approximation defined by columns 6, 7, 8 and 11 of Table 11. Plots of $\sum\Delta\gamma$, $\Delta\gamma$, $\Delta\varepsilon$ and $\varkappa^*(n)$ are shown in Fig. 38 in analogy to Figs. 20, 30 and 34.

Table 12 carries the approximation further in accordance with Table 9. The values of $\varkappa_0^*/2\pi$ are plotted at the bottom of Fig. 38. A linear approximation according to Fig. 26b is used to obtain $\varkappa_0^*/2$ for integer values of s*. These interpolated values are listed in column 2 of Table 13, which corresponds to Table 10. The sliding average of 11 samples of column 2 is listed in column 3. The sum of column 3 yields 0.3317, which means that $\overline{\varkappa}_0^*/2\pi$ must be multiplied by i/(4 x 0.3317) = 0.7537 to obtain $\overline{\varkappa}_N/2\pi$ in column 4. The normalized curvature $\overline{\varkappa}_N/2\pi$ less the normalized curvature 0.01 of the "ideal" circle is listed in column 5 and plotted in Fig. 39.

A comparison of Figs. 27 and 39 shows a satisfying difference between the ellipses and the circle. One should observe that the rounding-off errors of the computations in Tables 11 to 13 strongly affect the plot of Fig. 27. For instance, $\Delta\varepsilon$ is shown as 0.3218 for n=1 but as 0.3217 for n=15 in Table 11. Hence, the rounding

Table 11. Computation of the approximation by a continuous curve with positive turning tangent and discontinuous or continuous first derivative for the circles of Figs.35 and 36.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
n	$\Delta\xi$	$\Delta\eta$	$\frac{\Delta\gamma}{2\pi}$	$\sum\frac{\Delta\gamma}{2\pi}$	n_c	X	Y	$\Delta s(n_c)$	ε	$\Delta\varepsilon$	s	s^*	$\Delta\hat{s}^*(n)$	$\varkappa^*(n)$
1	+1		+1/8	1/8	1	5	0	5	0	0.3218	0	0	3.685	0.1027
2	+1		0	1/8										
3	+1		0	1/8										
4	+1		0	1/8										
5	+1		0	1/8										
6	+1	+1	+1/8	2/8	6	3	1	3.162	0.3218	0.3218	5	5.827	3.685	0.1027
7	+1		-1/8	1/8										
8	+1		0	1/8										
9	+1	+1	+1/8	2/8	9	2	1	2.236	0.4636	0.1418	8.162	9.511	2.606	0.0635
10	+1		-1/8	1/8										
11	+1	+1	+1/8	2/8	11	4	4	5.657	0.7854	0.3218	10.398	12.117	2.606	0.1452
12	+1	+1	0	2/8										
13	+1	+1	0	2/8										
14	+1	+1	0	2/8										
15		+1	+1/8	3/8	15	1	2	2.236	1.1071	0.3217	16.055	18.710	2.605	0.1452
16	+1	+1	-1/8	2/8										
17		+1	+1/8	3/8	17	1	3	3.162	1.2490	0.1419	18.291	21.315	2.605	0.0635
18		+1	0	3/8										
19	+1	+1	-1/8	2/8										
20		+1	+1/8	3/8	20	0	5	5	1.5708	0.3218	21.453	25.000	3.685	0.1027

affects the fourth digit, and - with two exceptions - the values in column 5 of Table 13 differ from zero in the fourth digit only.

9. SUMMARY OF THEORETICAL RESULTS

We have shown that contour operators or scanning patterns permit the detection of a contour where the shade of greyness of a photo transparency changes. This detection is deterministic and can readily be automated.

We have further shown that the curvature of a contour or a line normalized by its length is an invariant under shift, rotation and scaling. A continuous normalized curvature was derived from the finite samples taken with finite resolution Δx and Δy, or "on a square field" for short. This normalized curvature characterizes the shape of a contour or curve. The mathematical processes for its derivation were chosen so that they can be readily automated.

There are two areas in which the theory should be carried further.

(a) The straight line approximations of Figs. 33 and 37 appear substantially better than the normalized curvatures of Figs. 27c and 39. The reason is probably that the human eye smooths the polygons of Figs. 33 and 37 by looking at more than three adjacent points, while the theory developed so far only uses three (corner) points. The extension of the theory from three to more points is straightforward, and it would probably result in contour recognition and classification as good as done by the human eye.

(b) The contour operators of Fig. 7 with 2 x 2 points each are satisfactory for following a contour. They will not distinguish between an area and a line - as in Figs. 35 and 36, they will not bridge a gap in a line caused by photographic imperfection, and they will not recognize the branching or intersecting of lines - such as road intersections and crossings. Here again, the extension of the theory is straightforward to the use of contour operators with more than 2 x 2 points.

In order to make use of the normalized curvature for automated

Table 12. Computation of the approximation of the polygon of Fig.37 by a curve according to Fig.25 with curvature larger than zero in every point.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
n	s^*	$\Delta\varepsilon$	$\Delta s^*(n)$	$\Delta s^*(n-1)$	$\Delta \hat{s}^*(n)$	$\rho^*(n)$	$\Delta s'^*(n)$	s_0^*	d_1^*	d_2^*	$\rho^*(n,d)$	$\varkappa^*(n,d)$	$\varkappa_0^*/2\pi$
1	0	0.3218	5.827	3.685	3.685	9.737	1.071	-1.843	1.843		10.981	0.0911	0.0142
2	1.165								3.008		11.769	0.0850	0.0132
3	2.331								4.174		12.556	0.0796	0.0124
4	3.496									4.174	12.556	0.0796	0.0124
5	4.662									3.008	11.769	0.0850	0.0132
6	5.827	0.3218	3.684	5.827	3.685	9.737	1.071	7.670		1.843	10.981	0.0911	0.0142
7	7.055									0.615	10.152	0.0985	0.0153
8	8.283									2.531	18.787	0.0532	0.0083
9	9.511	0.1418	2.606	3.684	2.606	15.748	0.539	10.814		1.303	17.312	0.0578	0.0090
10	10.814								0	0	11.318	0.0884	0.0138
11	12.117	0.3218	6.593	2.606	2.606	6.887	1.994	10.814	1.303		8.581	0.1165	0.0182
12	13.765								2.951		10.724	0.0932	0.0145
13	15.414								4.600	4.600	12.870	0.0777	0.0121
14	17.062									2.951	10.726	0.0932	0.0145
15	18.710	0.3217	2.605	6.593	2.605	6.887	1.994	20.013		1.303	8.582	0.1165	0.0182
16	20.013								0	0	11.318	0.0884	0.0138
17	21.315	0.1419	3.685	2.605	2.605	15.748	0.539	20.013	1.302		17.310	0.0578	0.0090
18	22.543								2.530		18.784	0.0532	0.0083
19	23.772								0.614		10.152	0.0985	0.0153
20	25.000	0.3218	5.827	3.685	3.685	9.737	1.071	23.158	1.842		10.981	0.0911	0.0142

Table 13. The curvature $\varkappa_o^*/2\pi$ for integer values of s^*, the curvature $\bar{\varkappa}_o^*/2\pi$ obtained by averaging over 11 values of $\varkappa_o^*/2\pi$, the averaged curvature normalized to $\bar{\varkappa}_N/2\pi$ according to Eq.(50), and the deviation $\bar{\varkappa}_N/2\pi-0.01$ of the averaged curvature from that of the circle; values hold for the circles of Figs.35 and 36.

1	2	3	4	5
s^*	$\varkappa_o^*/2\pi$	$\bar{\varkappa}_o^*/2\pi$	$\bar{\varkappa}_N/2\pi$	$\bar{\varkappa}_N/2\pi-0.01$
0	0.0142	0.0125	0.0094	-0.0006
1	0.0133	0.0125	0.0094	-0.0006
2	0.0126	0.0130	0.0098	-0.0002
3	0.0124	0.0131	0.0099	-0.0001
4	0.0127	0.0129	0.0097	-0.0003
5	0.0134	0.0125	0.0094	-0.0006
6	0.0144	0.0125	0.0094	-0.0006
7	0.0153	0.0129	0.0097	-0.0003
8	0.0099	0.0133	0.0100	0.0000
9	0.0087	0.0134	0.0101	0.0001
10	0.0108	0.0134	0.0101	0.0001
11	0.0144	0.0134	0.0101	0.0001
12	0.0178	0.0134	0.0101	0.0001
13	0.0162	0.0135	0.0102	0.0002
14	0.0142	0.0142	0.0107	0.0007
15	0.0127	0.0147	0.0111	0.0011
16	0.0130	0.0146	0.0110	0.0010
17	0.0145	0.0141	0.0106	0.0006
18	0.0166	0.0134	0.0101	0.0001
19	0.0172	0.0133	0.0100	0.0000
20	0.0138	0.0133	0.0100	0.0000
21	0.0102	0.0134	0.0101	0.0001
22	0.0086	0.0134	0.0101	0.0001
23	0.0109	0.0132	0.0099	-0.0001
24	0.0151	0.0128	0.0096	-0.0004

contour recognition one must develop a library of normalized curvatures. One must be able to access them in an organized way, like one finds a book in a library by a hierarchy of more and more precise specifications of the subject. For instance, an ellipse would be found by working through the following hierarchy of categories:

1. straight line contour, yes or no
2. curvature of circle, yes or no
3. curvature periodic with half the length of the contour, yes or no
4. curvature equal to its mirror image at four points, yes or no

10. EQUIPMENT FOR CONTOUR RECOGNITION

10.1 Block Diagram

A block diagram of equipment for contour recognition by means of contour operators is shown in Fig. 40. The heart of the equipment is a plasma display tube with a resolution of 512 x 512 points. Such tubes are commercially available (Petty and Slottow 1971). The tube can produce patterns like the Walsh or block pulse patterns of Fig. 8 anywhere in its field of 512^2 points. Such a pattern with 8 x 8 points - rather than 2 x 2 points as in Fig. 8 - is shown in Fig. 40. The equipment is designed so that any pattern with n x m points, where n, m = 1, 2, 4, 8, 32, 64, can be produced. This makes the equipment good enough for the generalization mentioned in Section 9 under point b.

The plasma tube display requires 512 power drivers for the variable x (PDRx) and 512 more for the variable y (PDRy). The selector switches SSWx and SSWy determine which power drivers are activated. The format switches FSWx and FSWy determine the format of the scanning pattern in Fig. 6 and the contour operators in Fig. 7. As pointed out, the equipment is designed for the more general format n x m rather than 2 x 2. The binary function

generators BFGx and BFGy produce functions according to Fig. 8 but with up to 64 x 64 point elements instead of the 2 x 2 shown. A display drive control DDC produces the commands for the various circuits.

The basic idea for the use of the equipment of Fig. 40 is as follows: A transparency of an aerial photograph is placed over the plasma tube display. The display produces a set of functions according to Fig. 8 and scans step by step in the x- and y-directions. The light passing through the transparency is detected by a photo tube, the output voltage of which represents the coefficients b(i,k) of Eq.(19), or similar coefficients from which the Walsh coefficients a(i,k) can be derived as in Eq.(20). By digitizing the output voltage of the photo tube one may thus produce the Walsh coefficients and use them according to Table 2 to determine the observed contour operator. The observed contour operator then determines according to Fig. 7 in which direction the scanning pattern on the plasma tube display should be advanced. This yields the values of $\Delta\xi$ and $\Delta\eta$ in Table 1 or Table 6, from which everything else is derived (See, e.g., Tables 7, 8 and 11). Hence, in addition to the blocks shown in Fig. 40 we need a photo detector with digitizer, and a computer.

A refinement of Fig. 40 is not to place the transparency directly on the plasma tube display, but to project an image of the plasma tube display on the transparency by means of optical lenses. There are two reasons for it. First, the glow discharge inside the plasma tube is about two millimeters distant from the transparency, which reduces the resolution. A lens system can project the glow on the transparency so that it is in focus there and yields the best resolution. Second, the use of the lens system permits a reduction of the projected image of the plasma tube. The resolution of these tubes is about 2 (glowing) lines per millimeter. A reduction of 10:1 increases the resolution to 20 lines per millimeter, which is closer to the resolution of photo transparencies. A typical aerial photograph would require about 10 x 10 such reduced images of the plasma tube to be covered, which implies that the transparency must be moved

mechanically 10 steps in the x-direction and 10 steps in the y-direction to be fully analyzed. There is currently no way around this. The available plasma displays resolve 512^2 = 262,144 points, and the mechanical movement increases this number to $512^2 \times 10^2$ = 262,144,000 points. This is the order of magnitude needed to analyze aerial photographs. If and when electronic displays with that many resolved points become available, we can do away with the mechanical movement. The equipment described below does not have the mechanical stepping device required to make 10 steps in the x-direction and 10 steps in the y-direction, but it is designed so that this improvement can be added readily.

In the following sections we describe in more detail 15 sheets of blueprints that contain the mechanical design, electronic circuits and timing diagrams. Due to their size and specialized interest the blueprints are not supplied with all copies of this report but are added to a few selected copies only. Additional sets of the blueprints may be requested from the author.

10.2 Mechanical Design

The mechanical design is based on two enclosed telephone cabinets as shown in the blueprint Plasma Image Analyzer, Mechanical. On the left are the electronic circuits that drive the plasma tube display. On top of them is the plasma display itself. The light from the display is deflected by Mirror 1 into the cabinet on the right, and by Mirror 2 down through the lens assembly to the fresnel lens. The photo transparency is to be placed on top of the fresnel lens, and the image of the plasma tube display is focused onto it. In its current design, the image of the plasma display is reduced only slightly to fit the standard format of aerial photographs. To achieve a reduction 10:1 one must change the lens assembly and move it closer to the fresnel lens. The design makes this readily possible. Furthermore, a mechanical scan can easily be added by stepping the phototransparency ten steps in the direction vertical to the

paper plane (y-direction) and moving the projected image by means of mirrors ten steps from left to right in the paper plane (x-direction).

The fresnel lens collects most of the light passing through the transparency and throws it on a photo tube. The output voltage of this tube, after some processing and digitizing, is fed to a computer which produces the command what to do next for the Display Drive Control DDC in the left telephone cabinet.

The blueprints Mirror Mount 1, Mirror Mount 2 and Fresnel Lens Mount show details fo the design of these items. The adjustment screws with springs in the mirror mounts permit the fine adjustment of the optical axis.

10.3 Principle of Electronic Design

The block diagram shown in the blueprint Plasma Display Image Analyzer, Electronic begins with the computer shown in the center. Through the Computer Interface Circuit CIC one accesses the Display Drive Control DDC and the Fototube and Sensor Circuit SEN. The data flow is indicated by arrows. The circuit design is based on computer architecture principles. A number of lines such as Data Bus and Address Bus are connected to all circuits, and the address sent on the Address Bus determines which circuit will receive the data on the Data Bus. Table 14 lists the common lines. Table 15 explains the functions of CONTROL 1 and CONTROL 2 in more detail. Table 16 lists the address codes of the addressable circuits.

The general operation is as follows. The controlling computer sends an address on the ADDRESS BUS. CONTROL 2 enables all circuits to receive this address, but only the one with the correct address will be activated. For instance, the address 1111 will activate the format switch control FSC for x. If the computer sends now a number, e.g., 4=1000, on the DATA BUS, the format switch control FSC will close the first four format switches FSW. Generally, each operation requires two numbers from the control computer, the address and the data. We will see later that there is one exception: The gas discharge in the

Table 14. Common lines for all circuits of the Plasma Image Analyzer

	Name	Function
1	CLOCK 1	main clock pulses
2	CLOCK 2	main clock pulses shifted by half a clock period
3	CONTROL 1	enables data read-in
4	CONTROL 2	enables address read-in
5	ADDRESS BUS	transmits address
6	DATA BUS	transmits data
7	INHIBIT	inhibits data and address transfer while an instruction is operated on

Table 15. Function of CONTROL 1 and CONTROL 2 signals.

CONTROL 1
Enables data read-in. HIGH during data read-in, LOW otherwise. During HIGH data can be read serially into the activated shift register. The transition from HIGH to LOW transfers the data paralled from these registers to buffer storages.
CONTROL 2
Enables address read-in. HIGH during address read-in, LOW otherwise. During HIGH the address code is read into the shift registers of all address decoders. The pulses from CLOCK 1 are inhibited at all other times. Hence, the last address code transmitted is stored in all shift registers of the address decoders until the next address code arrives.

Table 16. Assignment of Addresses

address number		addressed circuit
0	0000	
1	0001	
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	general binary function circuit for y
7	0111	general binary function circuit for x
8	1000	voltage control for photo multiplier
9	1001	gas discharge control
10	1010	display drive control for y
11	1011	display drive control for x
12	1100	selector switch control for y
13	1101	selector switch control for x
14	1110	format switch control for y
15	1111	format switch control for x

display is switched on by the transmission of an address alone.

The circuit shown in the blueprint Plasma Image Analyzer, Flectronic contains the same circuits for the variables x and y. Table 16 shows the address numbers assigned to the various circuits. Only the addresses 6 to 15 are currently used, the others are a reserve for a future expansion of the equipment.

The Selector Switches SSW and the Power Drivers PDR are divided into 8 parts SSW1 - SSW8 and PDR1 - PDR8. The reason is that 512 output terminals are required for these circuits to drive the 512 lines or columns of the plasma tube display, and it is very difficult to accomodate that many terminals on fewer than eight printed circuit cards.

The various circuits shown in the blueprint Plasma Image Analyzer, Electronic are discussed in more detail in the following sections.

10.4 Display Drive Control DDC

Refer to the blueprint Display Drive Control DDC. It contains three sections. One produces the function wal(k,x), the second the function wal(m,y), and the third fires the plasma display. All three sections have to be addressed by a control computer via the ADDRESS BUS and CONTROL 2.

Let the address code 1011 be transmitted on the ADDRESS BUS. The address decoder 11 for the function wal(k,x) will be activated, which means that a HIGH voltage is obtained at terminal W of MUL3 if a HIGH voltage is applied to CONTROL 1. This enables the shift register SRE5, and the number on the DATA BUS is read into SRE5. The number is transferred parallel to the D-type flip-flop storage FDH2 at the end of the enable signal from MUL3. This number represents the normalized sequency k of wal(k,x). It is used to produce the spatial function wal(k,x) in the way to be discussed in the following Section 10.5.

The address code 1010 on the ADDRESS BUS will activate the address decoder 10 and produce the spatial function wal(m,x) in complete analogy to the generation of wal(k,x).

The address code 1000 on the ADDRESS BUS will activate the

...der 9. Every time this address ...
...y is fired twice; during the first dis...
...ires of the plasma display selected by the F...
be connected to a positive voltage, the other h...
voltage, while the polarity is reversed during t...
charge. Four output signals have to be produced ...
Power Drivers (TPD) to generate this double discha...
explanation of this circuit refer to the blueprint ...
for TPD, which shows the voltages at the points a, b, ...
function of time. The enable signal at point a is a p...
step function. It is differentiated to produce a posit...
pulse at point b, and the same pulse with reversed ampli...
point c. A timing circuit TIM1 produces a positive pulse ...
about 20 μs duration at point d; the duration of this pulse ...
be adjusted as indicated at the falling edge of the pulse i...
line d of the Time Diagram for TPD. A modulo 2 counter FFD p...
duces alternatingly a HIGH and a LOW voltage at point e every ...
a positive edge is produced by TIM1. The gate AND 2.1 passes t...
pulse (glitch) in line f of the Time Diagram for TPD; it must be ...
eliminated for a clean operation of the circuit.

The timer TIM2 produces a pulse of 2 μs duration at point g
for every negative edge at point f. This pulse is inverted at
point h, and differentiated at point i; the positive pulse at
point i becomes a negative pulse at point i. The action of the negative pulse
at the points c to i is shown in the blueprint Time Diagram for
TPD by dashed lines. The main feature is that the modulo 2 coun-
ter FFD turns off the gate AND 2.1 and thus prevents a second
cycle of the feedback loop to reach point i. Hence, the enable
voltage at point a produces the voltages at the point j and k
shown in the Time Diagram for TPD in the interval $0 < t < T$, and
no more.

The gate NOR2 produces negative trigger pulses at point ℓ,
which are transformed into rectangular pulses of 1 μs duration
at point m by the timer TIM3; the inverted pulses are produced

address decoder 9. Every time this address is received, the plasma display is fired twice; during the first discharge half of the wires of the plasma display selected by the Format Switches will be connected to a positive voltage, the other half to a negative voltage, while the polarity is reversed during the second discharge. Four output signals have to be produced by the Timer for Power Drivers (TPD) to generate this double discharge. For an explanation of this circuit refer to the blueprint Time Diagram for TPD, which shows the voltages at the points a, b, ..., $\bar{p}$ as function of time. The enable signal at point a is a positive step function. It is differentiated to produce a positive trigger pulse at point b, and the same pulse with reversed amplitude at point c. A timing circuit TIM1 produces a positive pulse of about 20 μs duration at point d; the duration of this pulse can be adjusted as indicated at the falling edge of the pulse in line d of the Time Diagram for TPD. A modulo 2 counter FFD produces alternatingly a HIGH and a LOW voltage at point e every time a positive edge is produced by TIM1. The gate AND 2.1 passes the voltage from point d whenever point e is HIGH. Note the short pulse (glitch) in line f of the Time Diagram for TPD; it must be eliminated for a clean operation of the circuit.

The timer TIM2 produces a pulse of 2 μs duration at point g for every negative edge at point f. This pulse is inverted at point h, and differentiated at point i; the positive pulse at point i becomes a negative pulse at point c, while the negative pulse at point i is suppressed. The action of the negative pulse at the points c to i is shown in the blueprint Time Diagram for TPD by dashed lines. The main feature is that the modulo 2 counter FFD turns off the gate AND 2.1 and thus prevents a second cycle of the feedback loop to reach point i. Hence, the enable voltage at point a produces the voltages at the point j and k shown in the Time Diagram for TPD in the interval $0 < t < T$, and no more.

The gate NOR2 produces negative trigger pulses at point ℓ, which are transformed into rectangular pulses of 1 μs duration at point m by the timer TIM3; the inverted pulses are produced

at point n. The signal with a glitch is produced at point o, the glitch is suppressed by an RC-filter, and clean signals are derived at the points p and $\bar{p}$. These two signals will be needed to turn on the power drivers of the plasma display. The signals e and $\bar{e}$ are used to produce first the Walsh functions -wal(k,x), +wal(m,y) and then their complements +wal(k,x), -wal(m,y). Note that the transients of e and $\bar{e}$ occur when the voltage at point p is high, and this prevents damaging surge currents during transients.

10.5 Binary Function Generator BFG

The Binary Function Generator BFG comes in two versions. The first accepts any string of binary numbers from the computer and stores them as function of either x or y. This permits the generation of any binary function but the process of loading is time-consuming. A second version produces only the Walsh functions, but it is hard wired and thus very fast.

For a discussion of the first version refer to the blueprint General Binary Function Circuit. Upon reception of the address code 6 for y and 7 for x the digits on the DATA BUS will be stored in the shift register SRE1 to SRE8. A total of 64 digits can be stored, but only 2 digits are needed to produce the four block functions of Fig. 8. For such a small number of digits the circuit works about as fast as the hard wired Walsh function circuit. The inverters INV3 and INV4 delay the clock pulse to make sure the voltage at terminal W has reached the HIGH state when the clock pulse reaches the shift registers SRE1 to SRE8. At the end of the enable pulse the digits in the shift registers are transferred in parallel to the D-type flip-flops FDH1 - FDH11.

The specialized circuit for Walsh functions only is shown in the blueprint Binary Function Generator for Walsh Functions BFG and Format Switches FSW. This circuit has been discussed in the literature (Harmuth 1977, p.150). The voltages at the input terminals x1 to x6 are either HIGH (1) or LOW (10). The states of the output terminals corresponding to the 2^6 combinations of voltages at the input terminals are shown in the

blueprint Time Diagram for Binary Function Generator for Walsh Functions.

The format switches FS are field effect transistor switches with a very high impedance in the open state. Hence, the output terminals ℓ of the format switches FS have three states: HIGH (+5V), LOW (0v) and FLOATING (no specified voltage but essentially no current). The purpose of the format switches is to select the format n x m of the functions used. All formats with n = 1, 2, 4, 8, 16, 32, 64 and m = 1, 2, 4, 8, 16, 32, 64 are possible; n and m can be selected independently.

10.6 Format and Selector Switch Control

Refer to the blueprint Selector Switch Control SSC and Format Switch Control FSC. The circuit diagram for the Format Switch Control starts out with an address decoder set for the code 15 = 1111 for the variable x and 14 = 1110 for the variable y. The correct address will enable the read-in from the DATA BUS to the shift register SRE5 when the line CONTROL 1 is HIGH. A six digit number is read in, and transferred in parallel to the D-type flip-flop storage FDH at the end of the enabling signal. The OR-gates make some or all of the output terminals n HIGH according to the listing of Table 17. The voltages H = +5V and L = 0 V open or close the format switches in the blueprint Binary Function Generator for Walsh Functions BFG and Format Switches FSW.

The Selector Switch Control SSC works very similarly. The address codes 13 = 1101 for x and 12 = 1101 for y permit the read-in from the DATA BUS. The data is transferred in parallel at the end of the enabling pulse to the storages FFD1 and FFD3. A total of 9 binary digits is stored, yielding 29 = 512 different output states. Both the outputs Q and their complements $\overline{Q}$ are used. The 2 x 9 voltages are fed to the Selector Switches. Each terminal Q or $\overline{Q}$ can be loaded with 16 mA, but only 512 x 10 pA = 5.12 mA are required to drive the switches. Hence, no overloading occurs despite the 512 single-pole-single-throw switches driven by each output terminal.

Table 17. Input data on DATA BUS and states of the output terminals n of the Format Switch Control FSC. The first digit of the input data arrives first and is stored in Q_F.

input data	state of output terminals n					
	1	2	3	4	5	6
000001	H	L	L	L	L	L
000010	H	H	L	L	L	L
000100	H	H	H	L	L	L
001000	H	H	H	H	L	L
010000	H	H	H	H	H	L
100000	H	H	H	H	H	H

10.7 Selector Switches SSW

Refer to the blueprint Principle of Selector Switches. It shows eight input terminals $\ell = 1 \ldots 8$, six sets of 64 switches each, and $2^6 = 64$ output terminals $i = 1 \ldots 64$. All switches are in the position 0 (right); when they are flipped to the left they are in the position 1. Let the eight voltages a, b, ... h be applied to the input terminals ℓ. The voltages at the output terminals for various positions of the six sets of switches SS1,i to SS6,i with $i = 1 \ldots 64$ are shown in Table 18. The output terminals that have no voltage assigned are FLOATING; they will permit practically no current to flow.

The implementation of this principle for up to 64 input voltages at the terminals $\ell = 1 \ldots 64$, nine sets of switches SS1,i to SS9,i and $2^9 = 512$ output terminals $i = 1 \ldots 512$ is shown in the blueprint Selector Switch Card SSC. Actually only 64 output terminals i are shown since one needs 8 such cards that are connected as shown in the blueprint Plasma Image Analyzer, Electronic.

10.8 Power Drivers PDR

Each selector switch card is connected to a power driver card. The circuit diagram of these cards is shown in the blueprint Power Driver Card PDR. Each card contains 64 power drivers PD, whose circuit is shown in the upper left corner.

The power drivers produce the three output states 0, +125 V (+125), and floating [F] in response to the three input states 0, +5 V (+5), and floating. The four transistors T3 - T6 must be high voltage types, such as 2N5059 or A5T5059. The rather large number of four transistors is chosen to save power, which is mandatory when working with the high voltages required by a plasma display.

The gates AND and OR prevent overloading of the transistors T5 and T6 during transients. For an explanation, let T6 be conducting and T5 by nonconducting (input voltage 0, output voltage 0). If the input voltage is now changed from 0 to +5 V, the transistor T5 becomes conducting and the transistor T6 nonconducting. During the transient, T5 as well as T6 may be

Table 18. Shifting of 8 input voltages a...h at the input terminals l=1...8 along the output terminals i=1...64 in the blueprint Principle of the Selector Switches by setting the six sets of switches SS1,i to SS6,i to the positions 0 (right) or 1 (left).

selector switches						output terminals i															
SS6,i	SS5,i	SS4,i	SS3,i	SS2,i	SS1,i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	0	0	0	0	0	a	b	c	d	e	f	g	h									
0	0	0	0	0	1		a	b	c	d	e	f	g	h								
0	0	0	0	1	0			a	b	c	d	e	f	g	h							
0	0	0	0	1	1				a	b	c	d	e	f	g	h						
0	0	0	1	0	0					a	b	c	d	e	f	g	h					
0	0	0	1	0	1						a	b	c	d	e	f	g	h				
0	0	0	1	1	0							a	b	c	d	e	f	g	h			
0	0	0	1	1	1								a	b	c	d	e	f	g	h		
0	0	1	0	0	0									a	b	c	d	e	f	g	h	
0	0	1	0	0	1										a	b	c	d	e	f	g	h
0	0	1	0	1	0											a	b	c	d	e	f	g
0	0	1	0	1	1												a	b	c	d	e	f
0	0	1	1	0	0													a	b	c	d	e
0	0	1	1	0	1														a	b	c	d
0	0	1	1	1	0															a	b	c

conducting, and the high voltage of 125 V could quickly cause thermal destruction. The gates OR and AND make both T5 and T6 nonconducting during any transient, and only after a certain guard time will one of the two transistors be made conducting. The AND-gate is controlled by the voltage on the line $\bar{p}$ of the Timer for Power Drivers (blueprints Display Drive Control DDC and Time Diagram for TPD), while the OR-gate is controlled by the voltage on line p.

10.9 Fototube and Sensor Circuit SEN

Refer to the blueprint Fototube and Sensor Circuit SEN. The current of the fototube is converted to a voltage and amplified to the 1 V level. Via the sample-and-hold circuit SHM the voltage is fed to the analog-digital-converter ADC, which feeds the digits to the control computer.

Two unusual circuits are required. A programmable, protected power supply adjusts the operating voltage of the fototube under computer control to the brightness of the plasma tube display. The reason for this is that the format of the displayed two-dimensional function may vary from m x n = 1 x 1 = 1 to m x n = 64 x 64 = 4112. The light emitted by the plasma display may thus vary by 4112:1, and the variation of the light received by the fototube is just as large. One can adjust the amplification of a fotomultiplier tube by changing its voltage. This is the purpose of the programmable power supply.

The address 8 = 1000 will enable the read-in of data from the DATA BUS to the shift register SRE2. At the end of the enabling pulse the data is transferred in parallel to the storages FFD1 and FFD2. The digital-to-analog converter DAC transforms the digits into a voltage which controls the output voltage of the power supply PSM.

A fototube is easily overloaded by accidental illumination from other sources than the plasma tube or by incorrect setting of the operating voltage delivered from the programmable power supply. A protection circuit is used that sets the operating voltage to zero if the fototube is saturated for about one second.

To do so the output voltage of OPA4 is integrated by OPA2. The integrator is reset to zero once a second via the timer TIM4. If during such an one second interval the integrated voltage exceeds about 10 V, a flip-flop FJK is triggered which switches the reference voltage of the digital-analog-converter via the switch ASW1 from -4.7 V to 0. At the same time, a glow lamp indicator "OUT" is turned on. The voltage for the fototube must be reset manually by means of the switch RESET "OUT".

11. CONCLUSION AND RECOMMENDATION

The contour recognition and classification based on angles and curvature probably simulates very closely the processes used by the human eye and brain. It is a much more sensitive method than classifications using correlation and mean square errors, which are based on integration, since it uses the second derivative of the contour which is much more sensitive to small changes than an integral over the area surrounded by the contour.

The equipment needed to implement the theory is well within the state of the art. No particular precision, as in Fourier optics processing, is needed, and the equipment lends itself readily to computer control in a feedback loop.

The following recommendations are made for the next steps of this program:

(a) Equipment according to the blueprints should be developed. The cost would be in the order of $100,000-, if the development is done by someone with previous experience with plasma tube displays and a scientific interest in the project, while about twice this amount will be required by a reliable organization with purely financial interest and no previous experience with plasma tubes. Before proceeding with the development a final determination should be made how versatile and complex one wants the equipment to be, since the development of the theory of contour recognition based on angles and curvature permits considerable simplifications compared with the equipment originally called for in this contract. After this possible final modification the design

should be frozen, since there are as many variations of the design as there are engineers and the perfect equipment is never built.

(b) There are three directions in which the theory should be expanded, but this advancement of theory is only justified if equipment is developed. The first direction is to work out a continuous second derivative of a contour which is based on more than three adjacent points; this appears to be at this time the primary shortcoming of the theory compared with a human observer. The second direction is to work out an intelligently organized "library" of typical contours encountered in aerial photography, which is needed to be able to classify the normalized curvature of an observed contour. The third direction is to advance from scanning patterns and contour operators with 2 x 2 points to more points, in order to solve the problem of road crossings and related problems.

12. COMMENTS ON TECHNOLOGY

An aerial photograph resolves about 10000^2 points. In order to implement the concepts in this report one needs an electro-optic converter that potentially permits this resolution, can operate with a computer in a feedback loop, and is sufficiently fast.

The cathode ray tube is the most widely used electro-optic converter. It is readily available for the resolution of about 1000^2 points. However, the existing diplays have been developed for applications with very different requirements and are thus not well suited for image processing. For instance, even though the writing speed of an electron beam is in the order of centimeter per nanosecond, the decay time of the phosphor is purposely made very long, typically in the order of milliseconds and longer. This makes such a converter as slow as a liquid crystal display in this application. A second inherent problem is the accuracy of the absolute location of the electron beam. A cathode ray tube may write 1000 lines on a TV monitor, but these lines may be shifted up or down by a few line widths without detrimental effect, as long as all the lines are shifted equally. This is not tolerable for image analysis, since one would not know exactly which point of an image is beeing analyzed.

The problem of absolute location of the cathode ray tube is overcome in the crossbar systems, the best known of which are liquid crystal displays and plasma displays. The crossbars, consisting of wires or conducting strips, are mechanically rigid and cannot shift due to voltage fluctuations. The main problem of crossbar systems is resolution.

The largest liquid crystal display known to the author resolves 32^2 points, and no work has been reported recently about an increase of this resolution. Furthermore, the liquid crystal display is slow, requiring upwards of 1 ms to advance from one

pattern to the next. Among its good features are the excellent definition of the patterns, low driving voltage, and next to no driving power.

Plasma displays are available with a resolution of 512^2 points. They can switch from one pattern to the next in about 10 µs. Hence, plasma displays are the first choice at this time. Their future development potential is, unfortunately, not good. For reasons of plasma physics, a display with 10000^2 points would have to be about 5^2 m^2 large. The high driving voltage of about 125 V is also undesirable. Hence, even though the equipment discussed in this report was designed for a plasma display, care was taken that most of the electronic circuits could be used with any other crossbar display too.

The crossbar devices with the best potential for future development appear to be those based on acousto-optical effects. There are a number of reasons for this. First, the area of a point - typically a light transmitting or reflecting spot - can be made in the order of 10^2 μm^2 as compared with 500^2 μm^2 for the plasma displays. This permits potentially displays for 10000^2 points to have manageable size. Second, the speed of such devices permits one to expect a transition time from one pattern to the next in 1 µs or less, which is faster than the plasma display can achieve. One should note that such acousto-optical scanners exist currently on paper only.

From the standpoint of technology one should thus develop image analysis with the help of the available plasma displays, but keep in mind that circuits and methods should be applicable to acousto-optical crossbar devices too. At the same time, image analysis provides a large application area for acousto-optical devices, and may thus help stimulate the development of acousto-optical crossbar scanners.

13. REFERENCES

Gurtler, R.W. and Maze, C. (1972). Liquid crystal displays. IEEE Spectrum, November, 25-29.

Harmuth, H.F. (1977). Sequency Theory - Foundations and Applications. Academic Press, New York.

Petty, W.D. and Slottow, H.G. (1971). Multiple states and variable intensity in the plasma display panel. IEEE Trans. Electron Devices ED-18, 654-658.

Salapatek, P. and Kessen, W. (1966). Visual scanning of triangles by the human newborn. J. Experimental Child Psychology 3, 155-167.

Tannas, L.E. and Goede, W.F. (1978). Flat-panel displays: a critique. IEEE Spectrum, July, 26-32.

Specialized Publications on Displays

1. Special Issue on Display Devices and Systems, IEEE Trans. Electron Devices ED-22, September 1975.
2. Special Issue on Displays and LEDs. IEEE Trans.Electron Devices ED-24, July 1977.
3. 1977 and 1978 Society for Information Display (SID) Digests of Technical Papers. Available from SID, 654 N. Sepulveda Blvd., Los Angeles, CA 90049.
4. Special Issue on Flat Panels and Large-Screen Displays, Proc. SID 17, no. 1, 1976.
5. Special Issue on Processing of Images for Bilevel Displays and the Generation of Pseudo-Gray Scale. Proc. SID 17, no. 2, 1976.

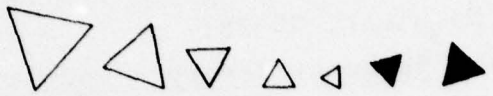


Fig. 1. Triangles with equal shape but various locations, sizes, rotations and representations.

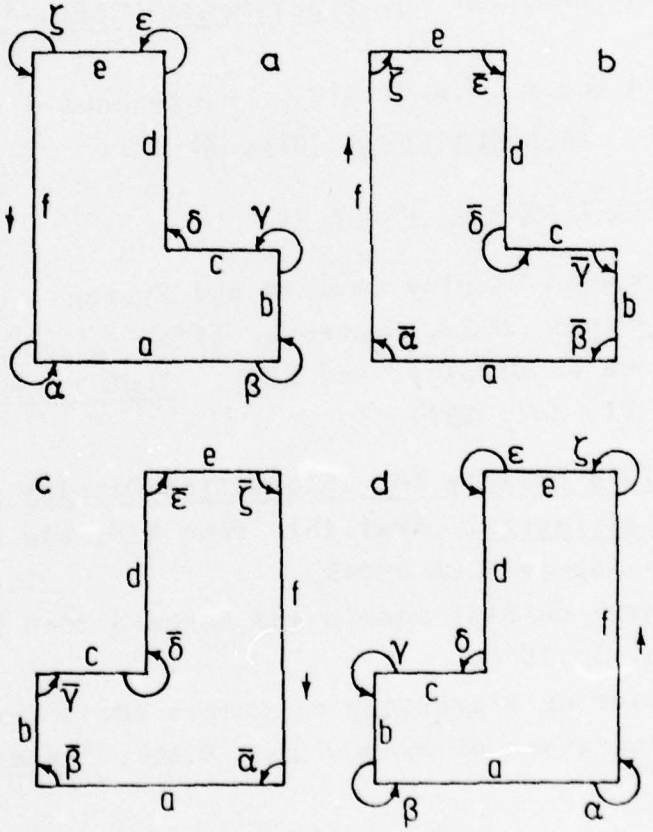


Fig. 2. Characterization of shape by a listing of relative lengths of sides and of angles. (a) counter clockwise listing; (b) clockwise listing; (c), (d) the same for the mirror image shape.

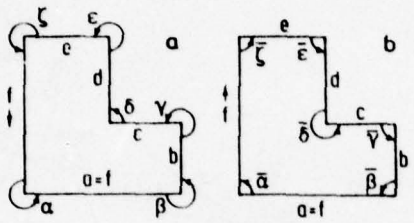


Fig. 3. Characterization of shape by a listing of relative lengths of sides and angles, if two sides are equally long and at the same time longer than all the other sides.

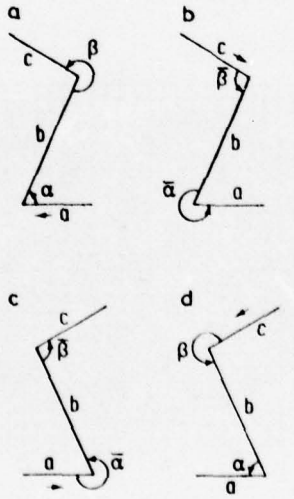


Fig. 4. Characterization of the shape of straight-line structures that are not closed.

whic are transformed into rectangular pulses of 1 µs duration
at point n by the timer TIM3; the inverted pulses are produced

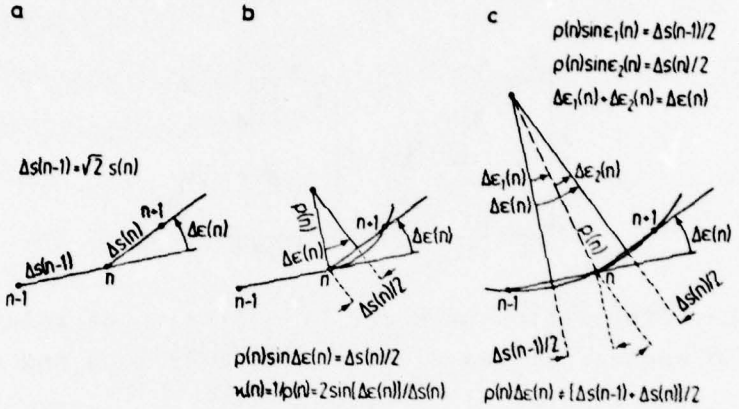


Fig. 5. Definition of curvature (a), and two possible definitions of a radius of curvature (b,c).

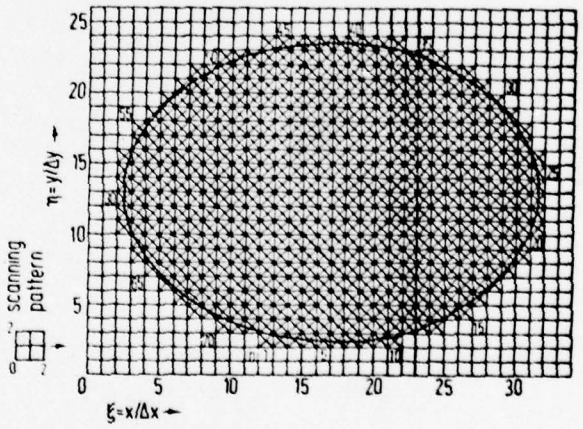


Fig. 6 An ellipse on a field with a finite number of elemental points or squares with a finite area ΔxΔy rather than in a field with nondenumerably many squares with infinitesimal area dxdy. The crossed squares represent the ellipse, the solid line with the more familiar shape of an ellipse is only shown as a visual aide. The 34 x 26 elemental points or squares represent a section of a liquid crystal or plasma tube display.

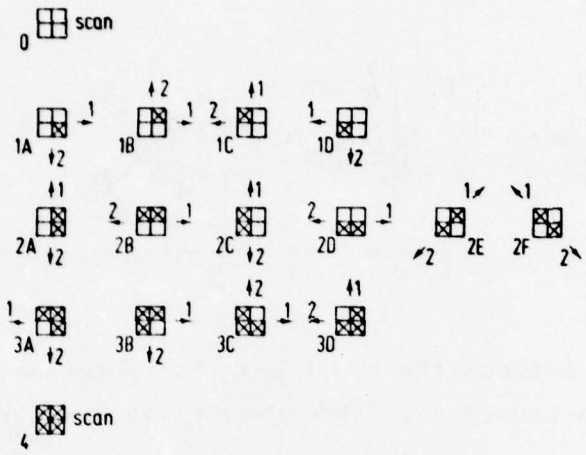


Fig. 7. The 16 possible contour operators produced by an illuminated scanning pattern with 2 x 2 points.

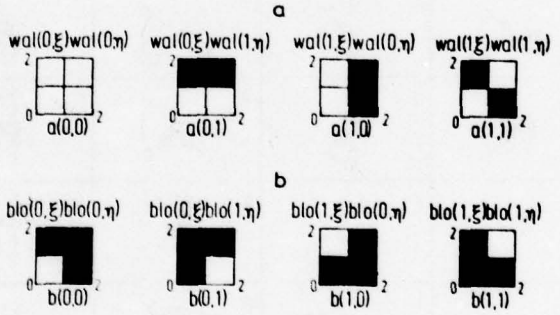


Fig. 8. Two-dimensional Walsh functions wal(i,ξ) wal (k,η) and block pulses or block functions blo(i,ξ) blo(k,η); white = illuminated = +1 for both systems of functions; black = -1 for the Walsh functions but black = opaque = 0 for the block functions.

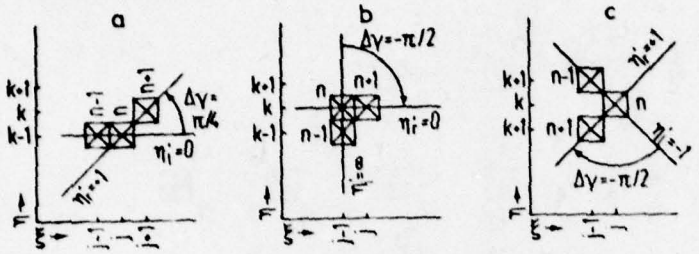


Fig. 9. Angles between the two lines that intersect at the center point of a sequence of three points on a square field.

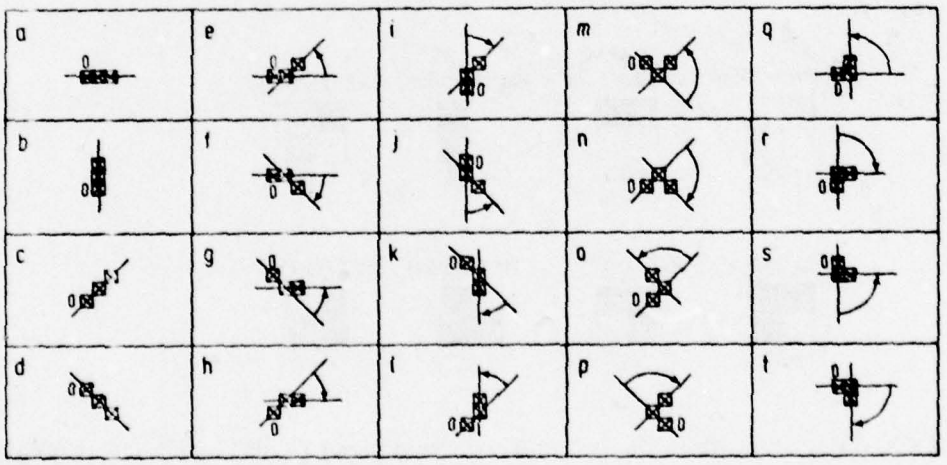


Fig. 10. The 20 possible sequences of three points of a contour on a square field if the numbering of the points begins at the point denoted 0.

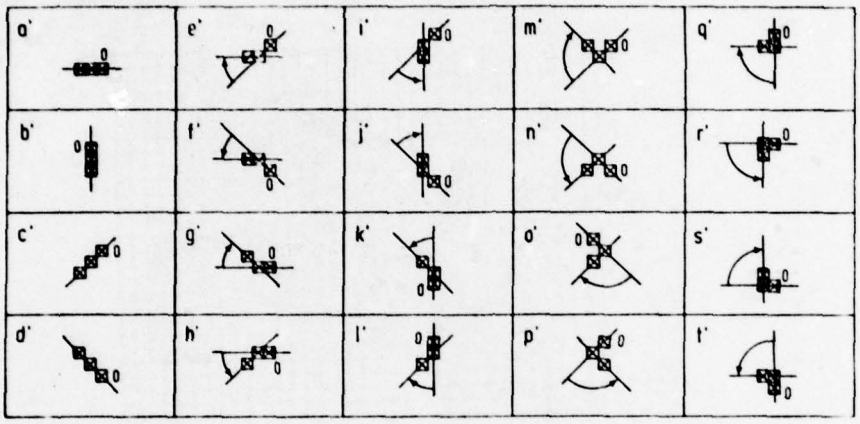


Fig. 11. The 20 sequences of three points complementing the 20 sequences of Fig. 10 by numbering the points in the opposite direction. The numbering starts with the point denoted 0.

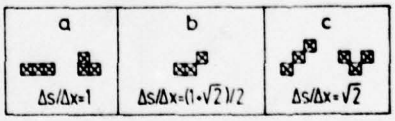


Fig. 12. The three possible values of the length element $\Delta s(n)/\Delta x$ in point n on a square field.

nonconducting. During the transient, T5 as well as T6 may be

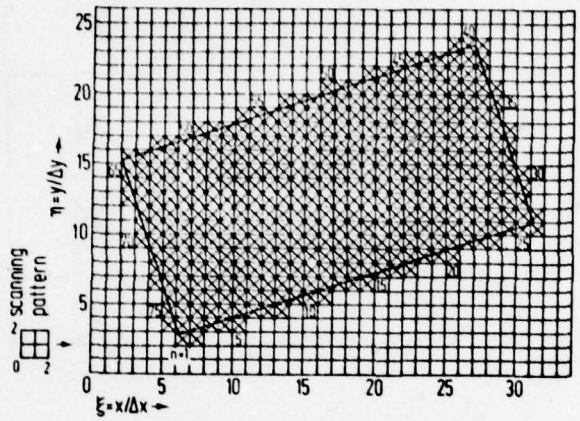


Fig. 13. A rectangle on a square field.

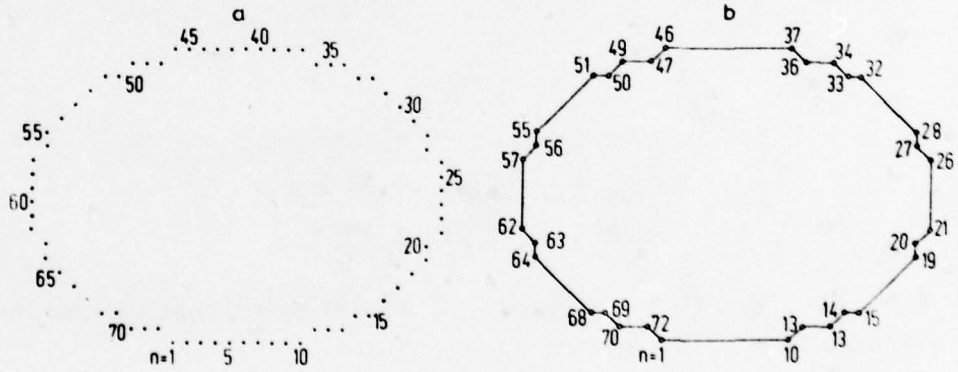


Fig. 14. Ellipse represented by a discontinuous set of a finite number of points (a) and by a continuous set of nondenumerably many points with a discontinuous first derivative.

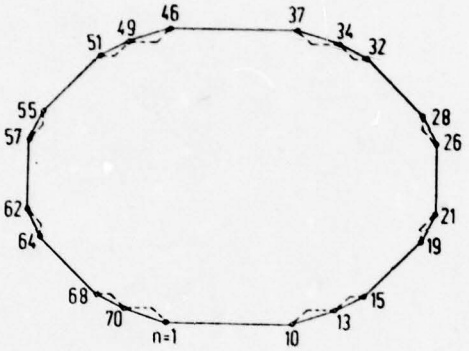


Fig. 15. Ellipse represented by straight lines between corner points. The angles at the corner points are always non-negative, which means either positive or zero.

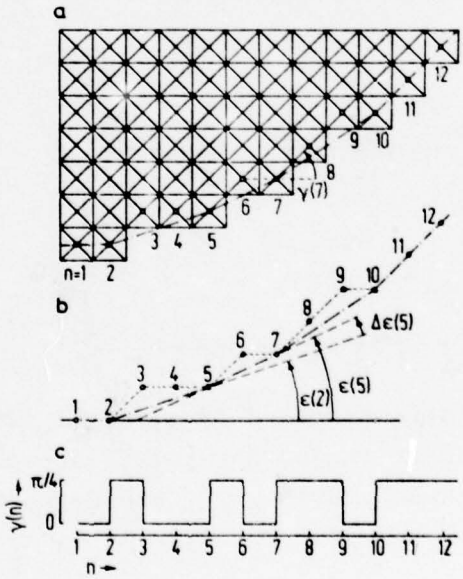


Fig. 16. Typical contour of an opaque area (a), definition of the angles $\varepsilon(n)$ and $\Delta\varepsilon(n)$ of the straight lines between corner points (b), and the characterization of the corner points by the positive steps of $\gamma(n)$ (c).

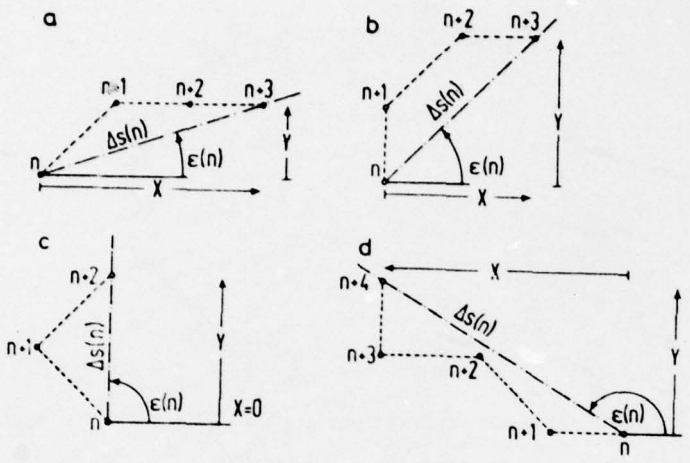


Fig. 17. Determination of ε(n) and Δs(n) from the values X and Y of a cartesian coordinate system.

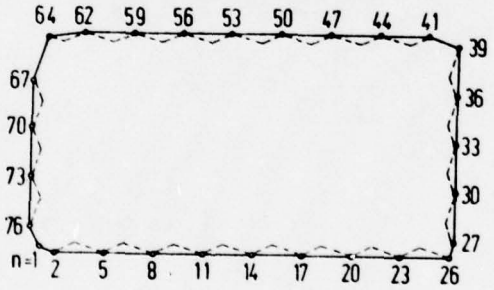


Fig. 18. Rectangle represented by straight lines between corner points. The angles at the corner points are always non-negative, but the corner points 5, 8, 11, ... are degenerated since Δε equals zero there.

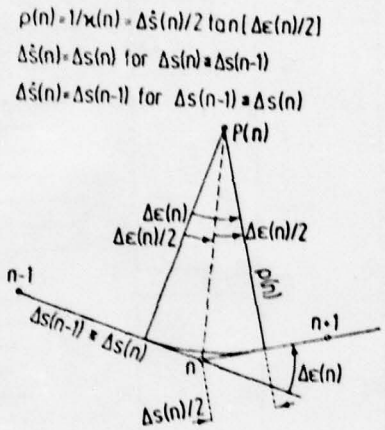


Fig. 19. Elimination of the discontinuity in a corner point n by means of a circle.

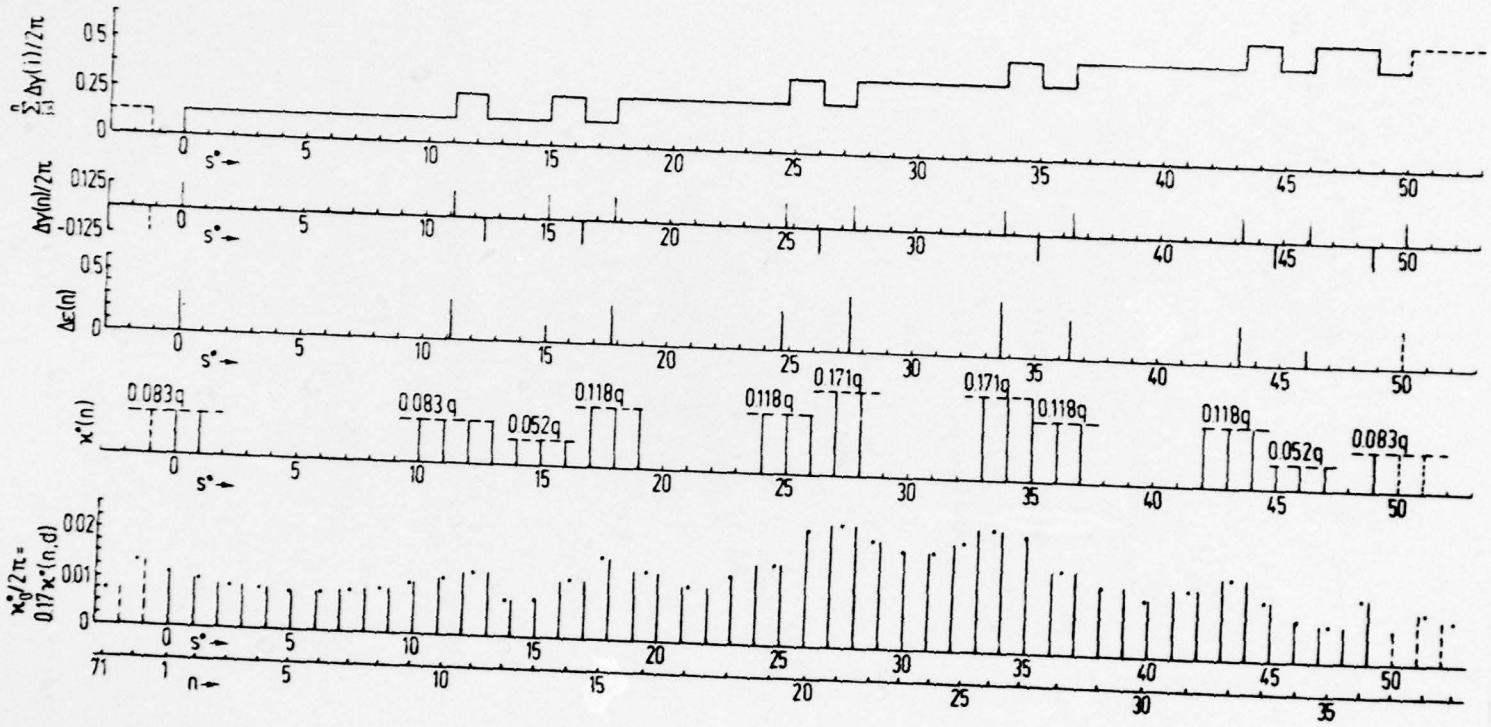


Fig. 20. Characteristic values of the ellipse of Fig. 6 as functions of the normalized curve length s* from the point n=1.

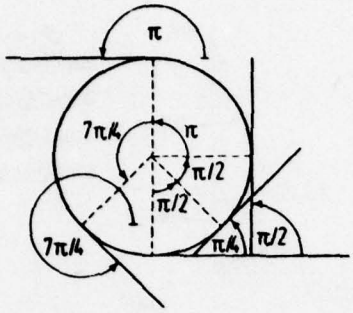


Fig. 21. Total rotation of the tangent around a closed curve must equal 2π, since in a description with polar coordinates the variable r would rotate 2π for a closed curve, and the tangent must rotate equally much to make the tangents at the beginning and end of the curve coincide.

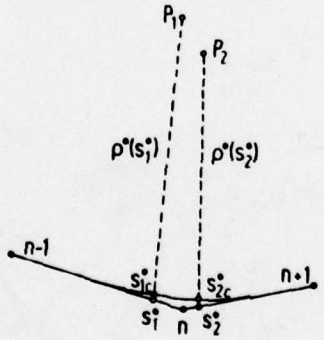


Fig. 22. Use of the normalized distances s_1^* and s_2^* measured along the straight solid lines in Fig. 15 to determine the corresponding points s_{1c}^* and s_{2c}^* on the curved line.

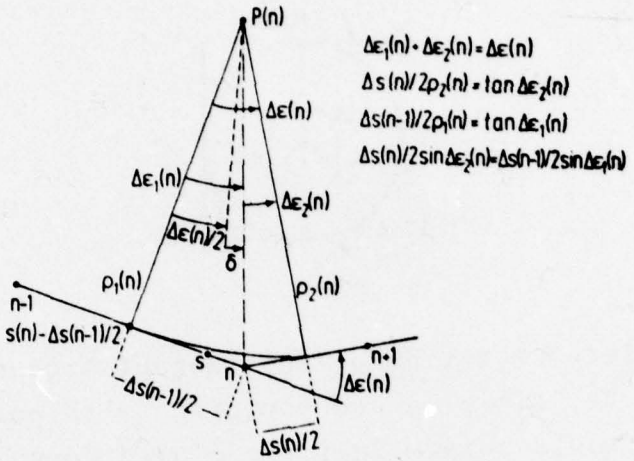


Fig. 23. First unsatisfactory method to generalize the circular section of Fig. 19 to eliminate all straight lines or "flat spots".

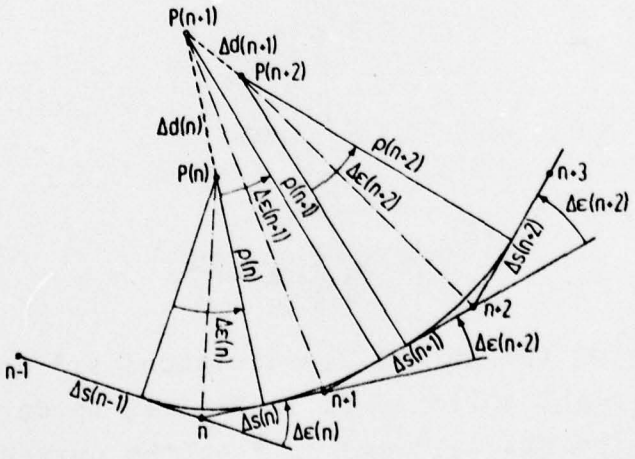


Fig. 24. Second unsatisfactory method to generalize Fig. 19 to eliminate all flat spots.

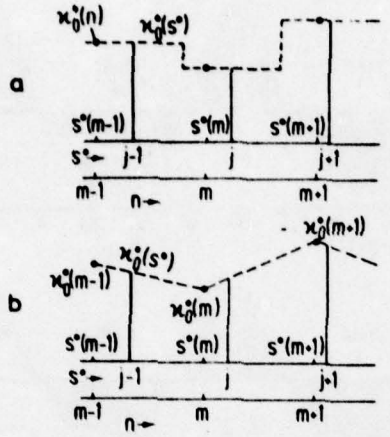


Fig. 25. Satisfactory generalization of Fig. 19 to eliminate flat spots.

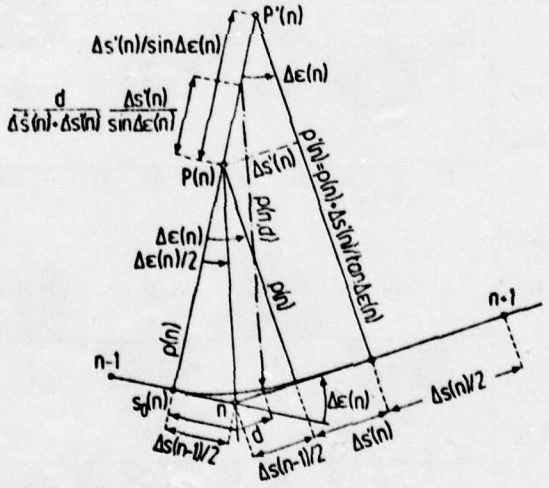


Fig. 26. Definition of $\kappa_0^*(s^*)$ for any value of s^* from $\kappa_0^*(n)$ defined for certain values of $s^* = s^*(n)$ only. (a) Step function approximation; (b) Linear interpolation approximation.

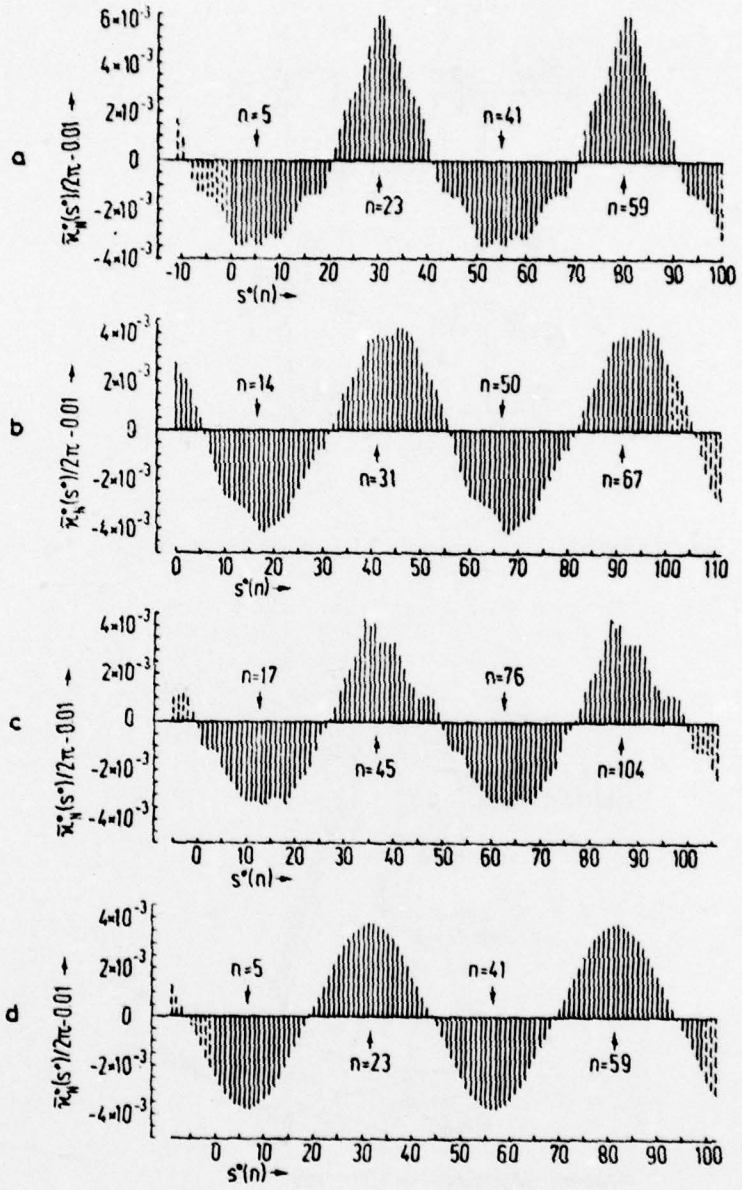


Fig. 27. Normalized curvatures of several ellipses, averaged over a number of samples, less the normalized curvature $2\pi/100$ of a circle with 100 points on its contour; (a) ellipse of Fig. 6, average of 11 samples; (b) ellipse of Fig. 28, average of 11 samples; (c) ellipse of Fig. 32, average of 11 samples; (d) ellipse of Fig. 6, average of 25 samples.

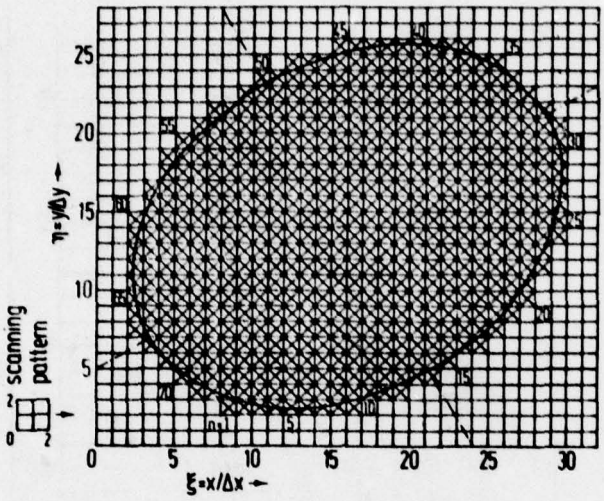


Fig. 28. The ellipse of Fig. 6 rotated through the angle $\varphi(b) = 30^{o}$.

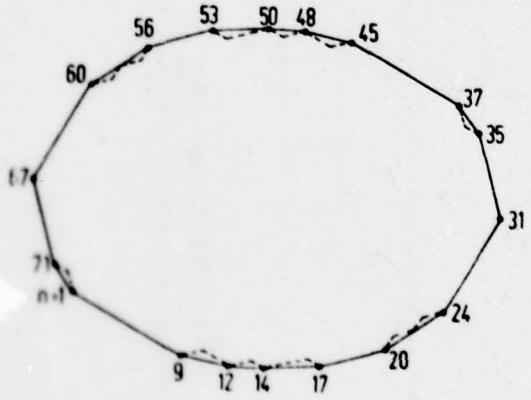


Fig. 29. The ellipse of Fig. 28 represented by straight lines between corner points.

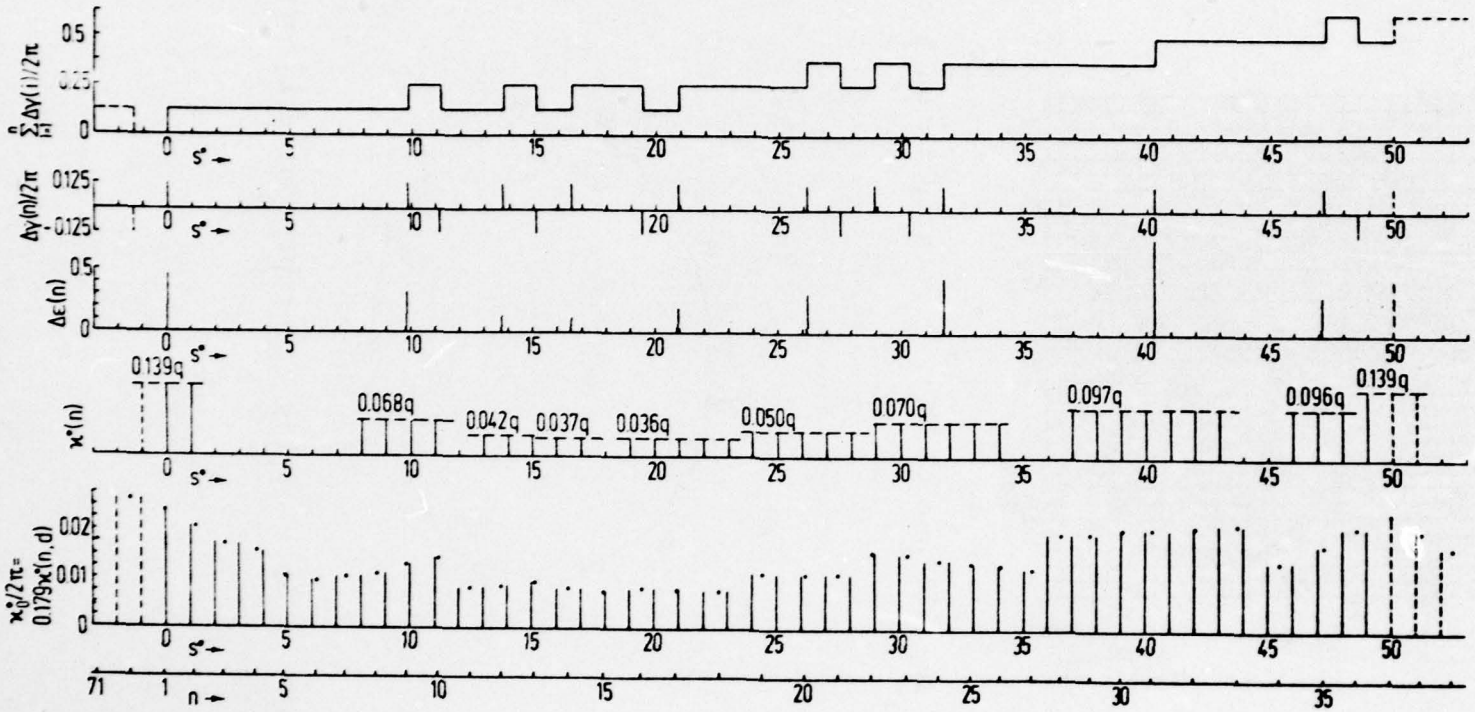


Fig. 30. Characteristic values of the ellipse of Fig. 28 as function of the normalized curve length s* from the point n=1.

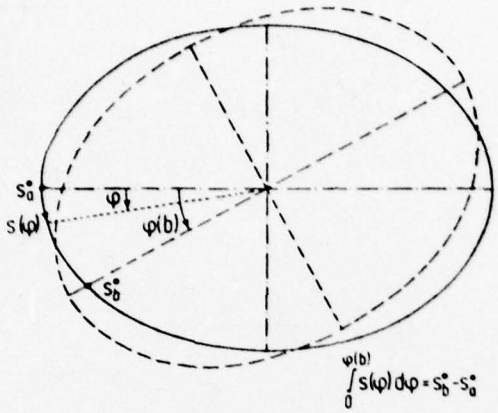


Fig. 31. Connection between the angle of rotation $\varphi(b)$ between two ellipses and the distance between their main axes measured along either one of the ellipses.

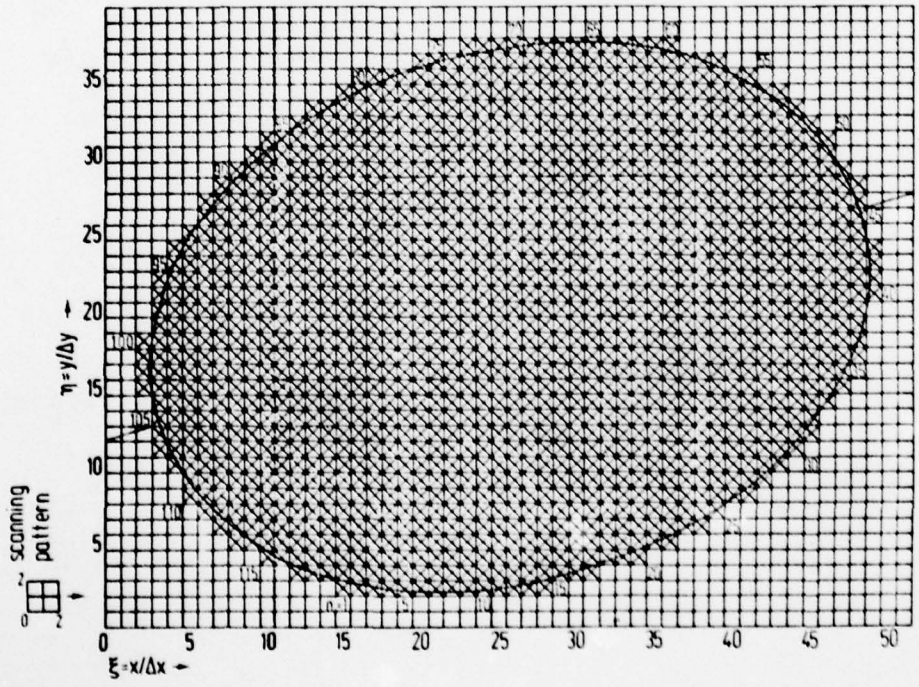


Fig. 32. An ellipse with 118 rather than 72 points on its contour in a square field.

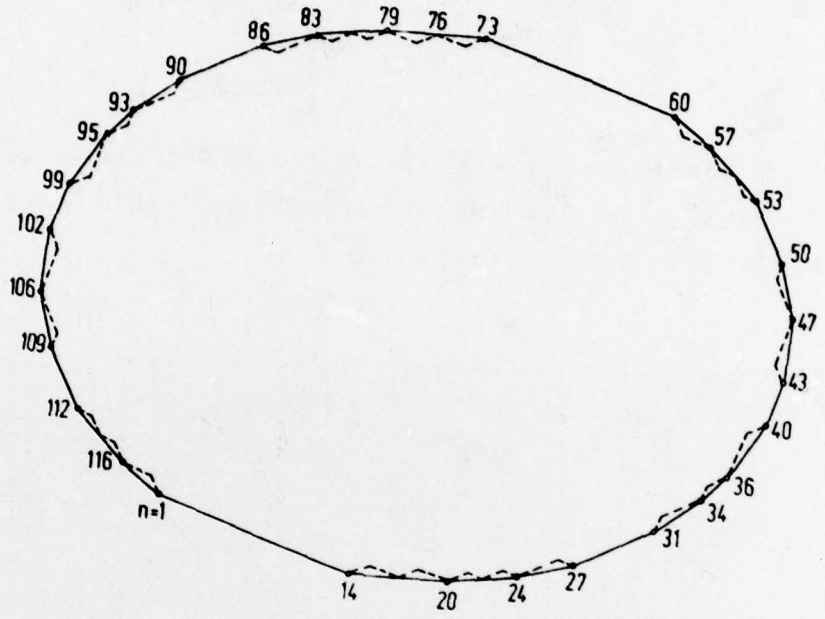


Fig. 35. The ellipse of Fig. 32 represented by straight lines between corner points.

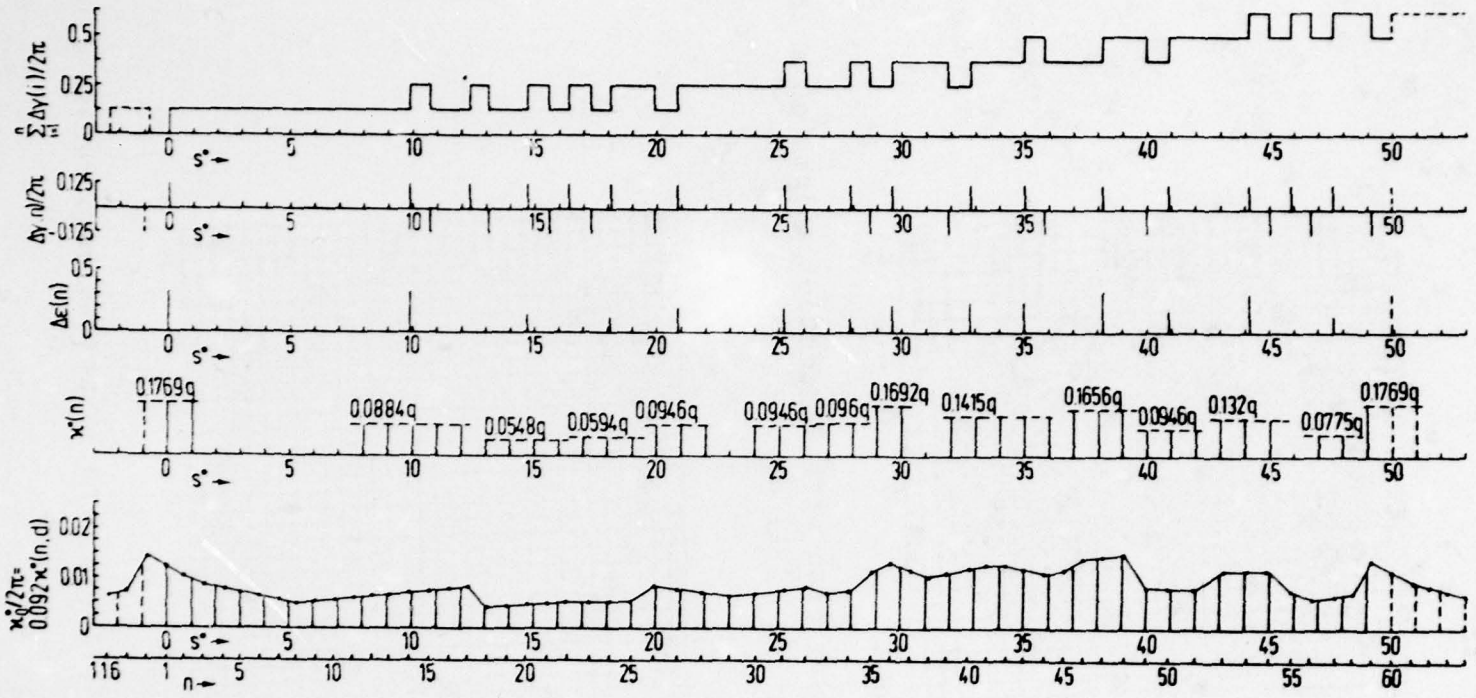


Fig. 34. Characteristic values of the ellipse of Fig. 32 as function of the normalized curve length s* from the point n=1. Note that the linear approximation of Fig. 26b is used for $\varkappa_0^*/2\pi$ rather than the staircase approximation of Fig. 26a as in Figs. 20 and 30.

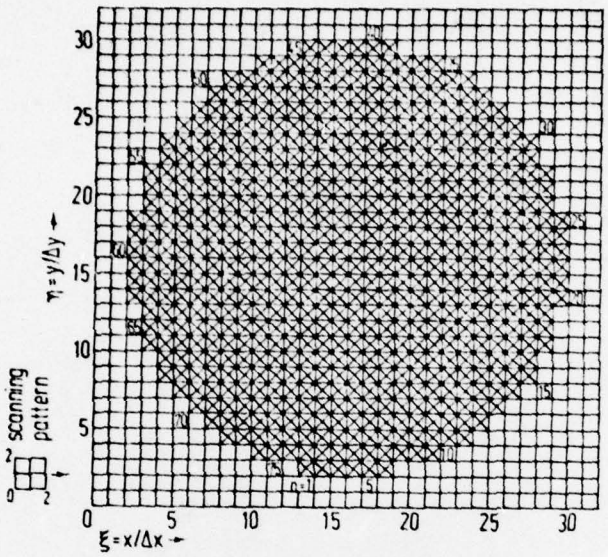


Fig. 35. A circular area with 76 contour points on a square field.

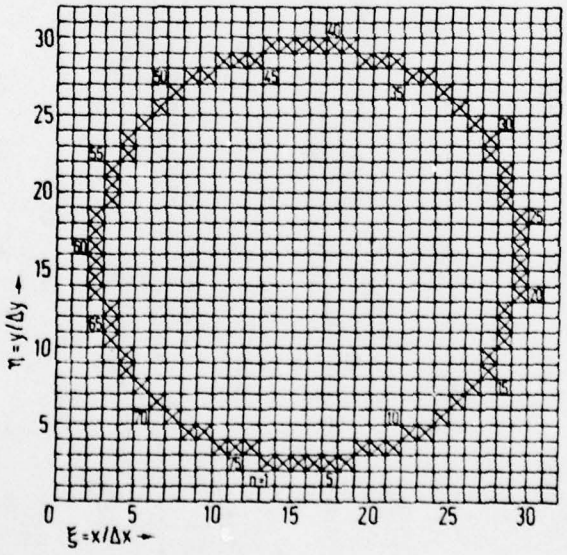


Fig. 36. A circular curve with 76 points on a square field.

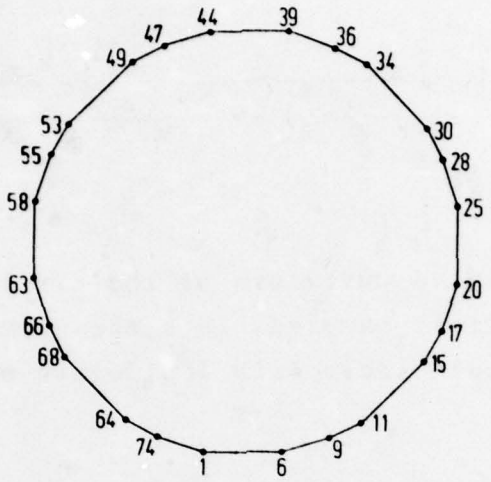


Fig. 37. The contour of the circular area of Fig. 36 or the circular curve of Fig. 36 represented by straight lines between corner points.

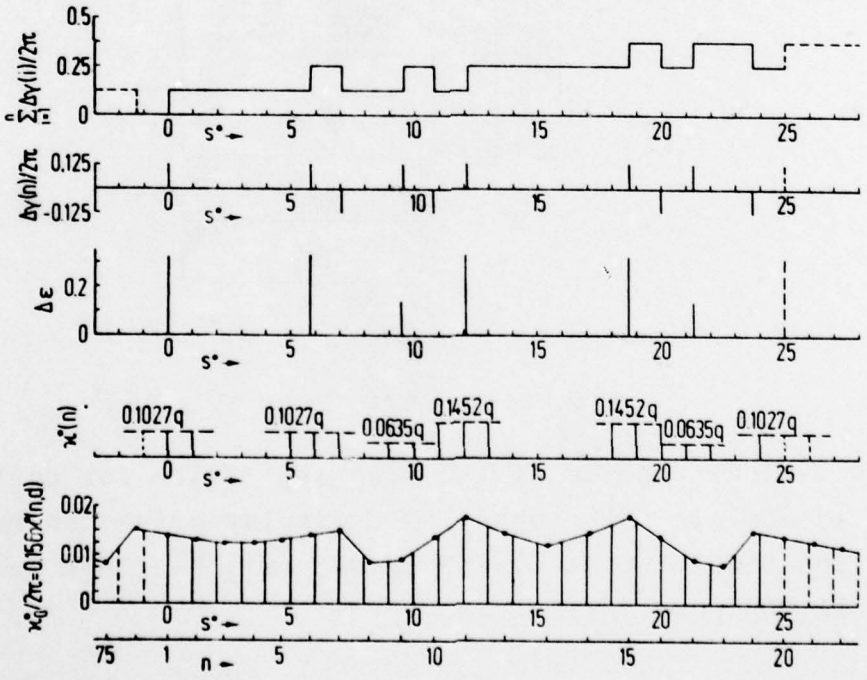


Fig. 38. Characteristic values of the circles of Figs. 36 and 37 as function of the normalized curve length s* from the point n=1. Note that the linear approximation of Fig. 26b is used for $\varkappa_0^*/2$ rather than the staircase approximation of Fig. 26a.

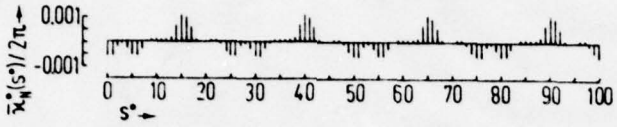


Fig. 39. Normalized curvature of the circles of Figs. 36 and 37, averaged over 11 samples, less the normalized curvature $2\pi/100$ of an ideal circle with 100 points on its contour.

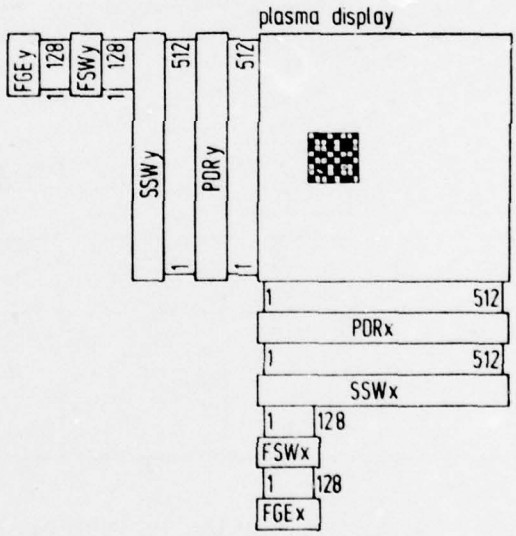


Fig. 40. Block diagram of the scanning device for contour recognition in aerial photographs. DDC display drive control, BFG binary function generator; FSW format switches; SSW selector switches; PDR power drivers.

END