A SIMPLIFIED AND FAST VERSION OF THE HUECKEL OPERATOR FOR FINDING OPTIMAL EDGES IN PICTURES

L.Mero - 2. Vassy
Computer and Automation Institute,
Hungarian Academy of Sciences
Budapest, Hungary

Abstract

A simplified version of the Hueckel operator was worked out to preserve all the advantages of the original one with a running time considerably shorter. Its concept is based on the reinterpretation of the Hueckel operator as a template matching method. This reinterpretation renders some theoretical and computational simplifications possible. As a result, an optimal edge-shaped template is constructed by linear combination of two suitably chosen original ones. The parameters of this optimal template characterize the edge segment found in a picture area.

Introduction

Hueckel's operator for edge detection in pictures is one of the best tools which have been invented for this kind of tasks. In very noisy pictures its high effectiveness can be comparable only with the global optimization procedures e. g. by Montanari and Martelli (8), (9), but without their drawbacks in processing time and memory requirement. The Hueckel operator processes a picture block containing 52 points on PDP-10 in 10 msec, e. g. if it scanned for example a 2,56 by 256 point picture so that the blocks overlap by one thirds of each other, the processing time would be about 40 seconds. Of course, by means of sophisticated edge-following methods this time can be reduced significantly, thus the applicability of this operator is not beyond the possibilities to process a real picture. These 10 msec are, however, rather large to look for some modifications based on its basic idea to speed up the operation, even by possible simplifications which preserve the excellent properties of the original one.

In this paper we present a method aiming at this by reinterpreting the principle of the Hueckel operator and giving a modification of it based on our interpretation.

Hueckel's method in its original form

The original interpretation of the Hueckel operator can be summarized as follows:

a/ We seek for a function S on a D area in the input picture, which function has a constant b value on one side and another constant b+d at the other side of a certain line, the equation of which is

$$cx + sy = r;$$

b/ This function S is determined by the least squares method, i. e. we minimize the sum of the square of differences point by point inside of D between S and the input $F/x,y/$ gray level function;

c/ For the sake of easy computation we set S as an infinite Fourier series of some H_i base functions;

d/ The infinite set of these H_i functions is approximated by a finite set of its lowest frequency members $H_0...H_7$;

e/ We complete the minimization task in this new space. The parameters of S characterize the parameters of the line segment found in D.

Hueckel extended this method for finding not only edges but lines as well (6), but this extended version will not be treated here, as we are interested in edge detection only. The sketches for $H_0...H_7$ are seen in Figure 1; + and - marks denote the sign of the functions, the lines inside their area D denote the zero crossings. We also do not present other details, though some of them /e.g. the so-called "solution theorem"/ are even theoretically very attractive, too.

In the following we will prove that the Hueckel operator can be viewed as a kind of template matching method. Before this we are to briefly survey some approaches to template matching.

Template matching

Let be given two function, $f/x,y/$ and $g/x,y/$ on a G two-dimensional area, and let us seek $f/x,y/$ in $g/x,y/$, i. e. the area in G where g is identical to f. In this case we can compute the following convolution:

$$\int_{H<G} f/x,y/ \, g/x,y/ \, dx \, dy = f \otimes g$$

and the sought area H' is just where this convolution takes the maximum value in G. $f/x,y/$ is called template function or simply template (13).

If $f/x,y/$ is edge-shaped, it can be used to find edges in pictures. Rosenfeld (14), Rosenfeld and Thurston (15), (16)

and others (7), (10), (11), (12) have worked out several versions of this method and Investigated them empirically. For example, perhaps the simplest templates for two-dimensional pictures are those of Shirai (17) sketched in Fig. 2. The first of them shows a maximum if it lies exactly on a vertical and the second on a horizontal edge. For making the method less noise-sensitive, one can take into account larger areas, e. g. 5x5 or 8x8 etc.

Since Hubel and Wiesel (4) pointed out that the visual information processing system in mammals operates in a way very close to this method, at least for the local edge and spot extraction on the retina, template matching has a certain theoretical support. Moreover, just this physiological achievement led Deutsch to develop his template called SLEN /Short Line Extractor Neuron (1)/ used for numerical character recognition. For real-time processing of real pictures, however, these methods do not produce so effective results as other ones, being though ad hoc without *any* theoretical suggestions. In the case of edge detection, if the shape and orientation of the template are fixed, the probability of good matching is little because the output value of the convolution decreases very sharply either because of the increase of the angle between the template edge and the real one or the curvature difference of them (2). From this two main shortcomings follow. First, the system would not find certain, unluckily oriented edges even from the line-shaped ones. To surmount this difficulty, Holdermann and Kazmierczak (3) have presented a method with parallel application of several templates to the same picture area around a point. They took care also of noise elimination by averaging, and their output picture after this multiple template operation looked quite good in quality, but even their method could not avoid the following - second - difficulty: in the case of fixed templates the large majority of the required computer operations falls into the category of unsuccessful trials and the whole processing time may be too long, especially if the picture is not "angular" enough. The main reason of these shortcomings is obviously the rigidity of the templates applied. In contrast to e. g. Holdermann and Kazmierczak's eight templates, we would need only one of them; but that would be the best matching one. The question is how to find just this "optimal template*, i.e, how to solve the opposite problem to that of template matching: fitting the template to a fixed picture area and not the picture area to a/some fixed template/s/. Let D be this area of the picture, T the set of all possible templates, and $F/x,y/$ the input two-dimensional grey-level function, then we seek $T/x,y/\epsilon T$ so that

$$\int_D T/x,y/ \ F/x,y/ \ dx \ dy = T \otimes F = max. \quad /1/$$

In the next section we will point out that Hueckel's operator completes just this task.

The Hueckel operator as a kind of template matching method

We will first need to recall some of Hueckel's notations. Let the grey values of the input picture on a circle-shaped area D be $F/x,y/$, the sought two-dimensional step function be

$$S/x,y,c,s,r,b,d/ = \begin{cases} b & \text{if } cx + sy > r \\ b+d & \text{if } cx + sy < r \\ O & \text{otherwise.} \end{cases}$$

Let the base functions chosen in the Hilbert space of all the functions on D be $H_o/x,y/...H_7/x,y/$.

With these notations the functional equation to be solved is:

$$\int_D (F/x,y/ - S/x,y,c,s,r,d,b/)^2 \ dx \ dy = min. \quad /2/$$

The Fourier coefficients for F and S are:

$$f_i = \int_D F/x,y/ \ H_i/x,y/ \ dx \ dy \quad /3/$$

$$s_i/c,s,r,d,b/ =$$

$$= \int_D S/x,y,c,s,r,d,b/ \ H_i/x,y/ \ dx \ dy \quad /4/$$

$$\lambda/c,s,r,b,d/ =$$

$$= \sum_{i=0}^{7} (f_i - s_i/c,s,r,b,d/)^2 = min. \quad /5/$$

Hueckel's operator solves this functional equation[x]. Now we will prove that /5/ can be rewritten in the form of /1/ in the previous section. For this we need a lemma:

Lemma: Let

$$\phi_i \ /i=1,...N/ \text{ and } \psi_i/\underline{p}/ \quad /i=1,...N/ \text{ so}$$

that

$$\sum_{i=1}^{N} \psi_i/\underline{p}/ = const.$$

where $\underline{p}$ is a vector of arbitrary dimensions.
If

$$\mu/\underline{p}/ = \sum_{i=1}^{N} (\phi_i - \psi_i/\underline{p}/)^2 \quad \text{and}$$

$$\nu/\underline{p}/ = \sum_{i=1}^{N} \phi_i \psi_i/\underline{p}/$$

_____

[x]Here we neglect Hueckel's heuristic weighting coefficients as they have no relevance to the problem considered.

then if for some values of **p**

$\nu/\underline{p}/$ = min.   then for the **same**

values fof **p**

$\nu/\underline{p}/$ = **max.**

The proof is given in Appendix A.
According to this lemma, /5/ can be re-
written as

$$\lambda'/c,s,r,b,d/ = \sum_{i=0}^{7} s_i/c,s,r,b,d/ \, f_i = \text{max.}$$

Substituting f_i from /3/:

$$\lambda'/c,s,r,b,d/ =$$

$$= \sum_{i=0}^{7} s_i/c,s,r,b,d/ \cdot \int_D F/x,y/H_i/x,y/dxdy =$$

$$= \text{max.}$$

With some trivial transscriptions:

$$\lambda'/c,s,r,b,d/ =$$

$$= \int_D F/x,y/[\sum_{i=0}^{7} s_i/c,s,r,b,d/H_i/x,y/]dxdy =$$

$$= \text{max.}$$

The expression

$$T/x,y,c,s,r,b,d/ =$$

$$= \sum_{i=0}^{7} s_i/c,s,r,b,d/ \, H_i/x,y/$$

is a template by its role in /6/, as the
form of /6/ is identical with that of /l/.
T/x,y,c,s,r,b,d/ is the optimal template
to the F/x,y/ input function constructed
as a linear combination from the set of
original templates $H_0...H_7$.

The possibility of simplifying the Hueckel
operator

In Hueckel's interpretation the basic
aim, i. e. finding an edge, is reached by
fitting a two-dimensional function to the
input grey-level function in a picture
area. Consequently, a relatively large set
of both the base functions and edge para-
meters is required, because it is obvious
that no satisfying degree of fitting could
be approached by a remarkably simpler base
field and parameter vector /containing e.g.
two or three functions or values res-
pectively/. If we are, however, considering
the method as a template matching one, the
fitting problem becomes unimportant and its
requirements cannot be taken into account.

First, we may choose a two-dimensional
parameter vector /c,s/ instead of
/c,s,r,b,d/ as the grey values of the two
sides of the step function are irrelevant,
and the exact location of the edge inside
the D area can be computed in a very simple
way once we have determined the edge di-
rection. The direction is characterized
only by c and s, as in the equation of the
edge

   ex + sy = r

c and s are just the direction cosines to
the +x axis.

Second, if we need only the edge di-
rection, we may work with only two of
Hueckel's original templates H . ,.H7, be-
cause it seems well even intuitively /we
will prove it later, of course/ that by
the linear combination of H_2 and H- the
direction of the optimal template can be
determined. By the direction of a template
we mean the direction of the O-crossing
of a H_2 or H_3 type function.

Third, if we do not take care of the
mathematical conditions of the Fourier
function fitting in a Hilbert space – as
these conditions determined the choosing
of H ...H_7 in Hueckel's work /Chapter 2/ -
the 8nly criterion in choosing our two
templates can be the speed of computation
which is very crucial in real applications.
Thus we can use both constant-valued
template functions and a more convenient
shape for the D picture area. The con-
venience of this shape means also the
speed of computing c and s or any other
quantities characterizing the direction.

The method we have got to by the
above considerations will be set forth in
the following section. It is essentially
simpler and faster than that of Hueckel,
but as will be ssen from the examples pre-
sented later, it preserves all of its ad-
vantages in comparison to other methods.
The computation consists of two steps. The
first one is the determination of the edge
direction by constructing an optimal
template from two beginner ones by linear
combination. Just like in Hueckel's method,
a vast part of the extremizatlon problem
is solved analytically and the number of
the required arithmetical operations is
small. The second step is the determination
of the exact location of the edge in D by
its two border points. It is solved by some
simple logical rules and point reflections.

Description of our edge detection operator

The core of the method is based upon
the following elementary theorem:

Theorem: Let D denote the unit square, f.
be the function being +1 above the sub-
diagonal of D, and -1 below it , f_2 be the
function being +1 above the principial
diagonal of D and -1 below it, /see Fig. 3,
a, b/ and let f be a function being b+d
above a line e in D, and b below this line.
/See Fig. 3, c or d/. Then if a denotes
the angle between the subdiagonal of D and
e, the following equation will hold:

$$\frac{\int\limits_{I} f\ f_2 dxdy}{\int\limits_{I} f\ f_1 dxdy} = tg\ \alpha$$

The proof of the theorem will be presented in Appendix B. The odd thing about the theorem is that the expression at the left side does not depend either on b and d or on the real place of the line but its angle.

By this theorem we are given a very pleasant tool to determine an edge in a little part of the picture. The two integrals can be very easy and fast computed, and because of the integrations this method is of very low sensitivity for noises. /Later we shall elaborate further considerations about this./

In our work, we have chosen D to be a square consisting of 8x8 picture points. Having supposed that there was a straight edge in the picture at D, by using the above mentioned theorem we could determine its place, i. e. its two endpoints in D.

To begin with we transform the value of the tangent to a more treatable form. We determine from the value of tg a how many picture points are the difference between the right and the left end of a line in D, the angle of which with the subdiagona.1 is a. / It is enough to deal with lines lying in the quarter of the plane shown on Fig. 4, because if the edge lies in the other quarter, then by counting with jtg a|, after the whole process a reflection of the edge to the principial diagonal of D - i. e. its endpoints - gives the real edge./

Everything stated until now is valid both for pictures with two and for those of more grey levels. Now we shall show the determining process of the real edge at first only for pictures with 2 grey levels and after that we shall show how to generalize that idea for pictures with more grey levels.

In the case with 2 grey levels the sum of the 1's in D divided by 8 shows, roughly spoken, up to what a height D is filled with ones. This means e. g. that if the edge were horizontal, then the edge could be either the horizontal line at that height, or that reflected to the centre of D /for D could be filled with ones from its bottom or from its top/. But even if the edge were not horizontal, we could know what was the difference between its left and right endpoints' height, so we could determine the edge. If both the left and the right endpoints of this edge were inside of D then we had only to determine whether this was the real edge or we had still to reflect it to the centre of D.

If one of the endpoints of the above obtained edge were not in D, it could be only by 1, 2, or 3 points sticking out of D, because now already the edge we are working with lies in the plane quarter shown on Fig. 4. But in these cases it Is already very easy to determine the real edge.

For pictures with more grey levels we could use the same method if we knew b and d, because then the number of the points with greater grey level could be approximated by the following expression:

$$S' = \frac{S - 64b}{d}$$

where S denotes $\int\limits_{I} fdxdy$, i. e. the sum of the grey levels of all points of D. But now by using the notations of Fig. 5, let T_1, T_2, T_3 and T_4 denote the sum of the grey levels of f /i.e. of the picture/ in these 4 quarters of D, and $\frac{min\{T_i\}}{16}$ resp. $\frac{max\{T_i\}}{16}$ approximates well b resp. b+d. After this the above described method of finding the edge can be followed in the case with more grey levels as well.

By this method we have already found either the real edge in f or the same reflected to the centre of D. /See Fig.6/ To determine which of them is the real edge, the following table shows a method provided that up to this time we have supposed in the steps that D was filled by the ones /resp. by b+d's/ from its top. The case A means that we have found the real edge while B means that still we have to reflect our edge to the centre of D.

If	$tg\alpha \geqslant 0$	$T_1 \geqslant T_2$	then A
		$T_1 < T_2$	then B
	$tg\alpha < 0$	$T_3 \geqslant T_4$	then B
		$T_3 < T_4$	then A

This method was implemented by a program on the assembly language of a 16 bit microprogrammed minicomputer of 550 program instructions without any cycles. If there is an edge in D it needs 250-400 machine operations, i . e . about 500-800 usees. No floating point arithmetics are needed.

About the reliability of the method rough un.der-estim.ations show that supposing a noise with a binomial distribution with a mean value of 5 error points in a 8x8 square, the probability of the difference between the edge determined by this method and the real edge being not greater than 1 point anywhere, is over 95 %.

Finally we present three examples for the working of the method. Example 1 is the example shown in Hueckel's article (5)

which we have completed at the corners. The dotted line shows the edge found by Hueckel, the continuous one is the edge found by our method.

Appendix A

Proof of the lemma used on page 5:
We will prove that

$$\nabla_{\underline{p}} \mu /\underline{p}/ = -const. \nabla_{\underline{p}} \nu /\underline{p}/.$$

If it is true, then if

$\nabla_{\underline{p}} \mu /\underline{p}/ = 0$, then $\nabla_{\underline{p}} \nu /\underline{p}/ = 0$, and

$$sgn(\nabla^2_{\underline{p}} \mu /\underline{p}/) = -sgn(\nabla^2_{\underline{p}} \nu /\underline{p}/).$$

$$\nabla_{\underline{p}} \mu \underline{p} = \nabla_{\underline{p}} \sum_{i=1}^{N} (\phi_i - \psi_i(\underline{p}))^2 =$$

$$= \sum_{i=1}^{N} - 2(\phi_i - \psi_i(\underline{p})) \nabla_{\underline{p}} \psi_i \underline{p} =$$

$$= -2 \sum_{i=1}^{N} \phi_i \nabla_{\underline{p}} \psi_i(\underline{p}) - \sum_{i=1}^{N} 2\psi_i(\underline{p}) \nabla_{\underline{p}} \psi_i(\underline{p}) =$$

$$= -2\nabla_{\underline{p}} \nu(\underline{p}) - \nabla_{\underline{p}} (\sum_{i=1}^{N} \psi_i^2(\underline{p})) = -2\nabla_{\underline{p}} \nu(\underline{p})$$

Q.e.d.

Appendix B

The proof of the theorem:
Let's use the notations of Fig. 8. We shall deal only with the case if $45^o \leq \alpha < 90^o$, because all other cases can be handled symmetrically. First we show that in the case of the line on Fig. 8, the statement of the theorem is valid. Here the quotient of the two integrals is:

$$\frac{\int_I f_2 f}{\int_I f_1 f} = \frac{T_2 + T_1}{T_2 - T_1}$$

Considering the sides of D to be one, we have

$$y = tg\beta = tg(45^o - \alpha) = \frac{tg - 1}{1 + tg\alpha}$$

and so

$$\frac{T_2 + T_1}{T_2 - T_1} = \frac{\frac{(1 + y)m}{2}}{\frac{(1 - y)m}{2}} = \frac{1 + y}{1 - y} = tg\alpha$$

Consequently, if e' lay above the line examined just now, then because of the similarity of all relevant triangles /see Fig. 9/ the statement would be true for e' as well:

$$T_1' = k T_1, \quad T_2' = k T_2 ,$$

$$\frac{T_1' + T_2'}{T_2' - T_1'} = \frac{kT_1 + kT_2}{kT_2 - kT_1} = \frac{T_1 + T_2}{T_2 - T_1} = tg\alpha.$$

Now if e' lay between e and the centre of D, then using the notations of Fig. 10 on account of the similarity of the triangles T_1 and $T_2 UT_3 UX$ here and T_1 and T_2 on Fig. 10 we have

$$\frac{T_2 + T + X + T_1}{T_2 + T_3 + X - T_1} = tg\alpha.$$

So we have to show only that

$$(tg\alpha =) \frac{T_2 + T_3 + X + T_1}{T_2 + T_3 + X - T_1} =$$

$$= \frac{T_1 + T_2 - T_3}{T_2 + T_3 - T_1} (= \frac{\int_I ff_2}{\int_I ff_1})$$

Subtracting 1 from both side, taking their reciprocal and adding 1 again we have

$$\frac{T_2 + T_3 + X}{T_1} = \frac{T_2}{T_1 - T_3} \qquad i. e.$$

$$\frac{T_1}{T_2 + T_3 + X} = \frac{T_3}{T_3 + X} .$$

This means we have to show only that

$$\frac{DE}{EB} = \frac{FG}{FB}$$

And this is true for, because of the similarities of the triangles FJB and CAB, $\frac{FG}{FB} = \frac{CD}{CB}$; so it is enough to show that $\frac{DE}{EB} = \frac{CD}{CB}$, i. e. that $\frac{CD}{DE} = \frac{CB}{EB}$

Now for AD is the bisector of the angle CAE, we have:

$\frac{CO}{DE} = \frac{CA}{AE}$, and for the similarity of the triangles CAB and EHB

$$\frac{CB}{EB} = \frac{CA}{EH}$$

and for the triangle AEH is isoceles AE=EH, and the theorem is proved in this case, too. If e were below the centre of D, the same proof could be presented for negative d. /We have nowhere made use of the positive property of D./ So the whole theorem is proved.

References

(1) Deutsch, S.: Conjectures on mammalian neuron networks for visual pattern recognition. IEEE Trans. System Science and Cybernetics, SSC-2, pp. 81-85/1966/

(2) Hawkins, J.K.: Photographic techniques for extracting image shapes. Photo-

graphic Science and Engineering, 8, pp. 329-335 /1964/

(3) Holdermann, F.; Kazmierczak, H.: Pre-processing of grey-scale pictures. Computer Graphics and Image Processing, 1, pp. 66-80 *119121*

(4) Hubel, D.H.; Wiesel, T.N.: Receptive fields, binocular interaction, and functional architecture of the cat's visual cortex. J. Physiol., 160, pp. 106-154 /1962/

(5) Hueckel, M.H.: An operator which locates edges in digitized pictures. Journal of ACM, 18, pp. 113-125 /1971/

(6) Hueckel, M.H.: A local visual operator which recognized edges and lines. Journal of ACM, 20, pp. 634-647 /1973/

(7) Levine, M.D.: Feature extraction: a surveay. Proc. IEEE, 57, pp. 1391-1407 /1969/

(8) Martelli, A: Edge detection using heuristic search methods. Computer Graphics and Image Processing, 1., pp. 169-182 /1972/

(9) Montanari, U.: On the optimal detection of curves in noisy pictures. Comm. ACM, 14 pp. 335-345 /1971/

(10) McLeod, I.D.G.: Comments on "Techniques for edge detection". Proc. IEEE, 60, pp. 1344 */1972*/

(11) Nadler, M.: An analog-digital character recognition system. IEEE Trans, on Electronic Computers, EC-12, pp. 814-821 /1963/

(12) Nilsson, N.J.: Adaptive pattern recognition: a survey. 1966 Bionics Symp., Dayton, 1966

(13) Rosenfeld, A.: Picture processing by computer. Technical report 68-71, Univ. of Maryland, Computer Science Centre, College Park, Maryland /1968/

(14) Rosenfeld, A.: A nonlinear edge detection technique. Proc. IEEE, 58, pp. 814-816 */X970/*

(15) Rosenfeld, A.; Thurston, M. : Edge and curve detection for visual scene analysis. IEEE Trans, on Computers, C-20, pp. 562-569 /1971/

(16) Rosenfeld, A.; Thurston, M.; Lee, Y.: Edge and curve detection: further experiments, IEEE Trans, on Computers, C-21, pp. 677-715 /1972/

(17) Shirai, Y.: A step toward context-sensitive recognition of irregular objects. Computer Graphics and Image Processing, £, pp. 298-307 /1973/
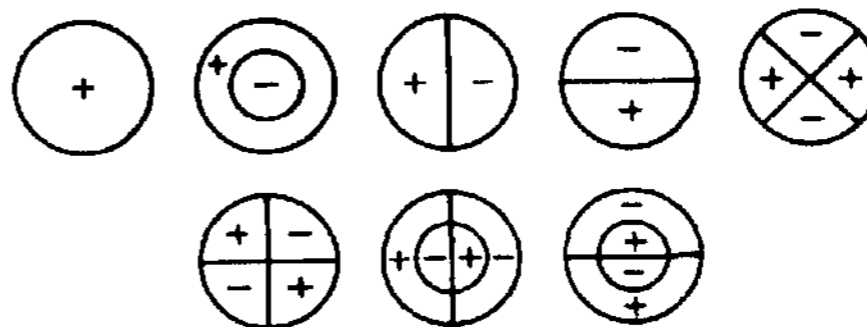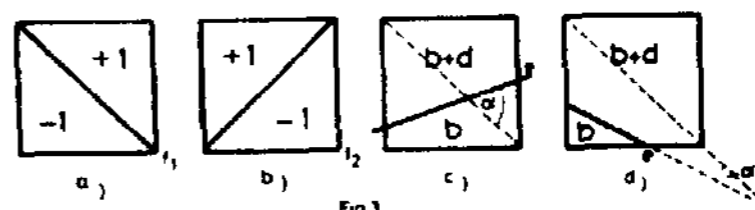
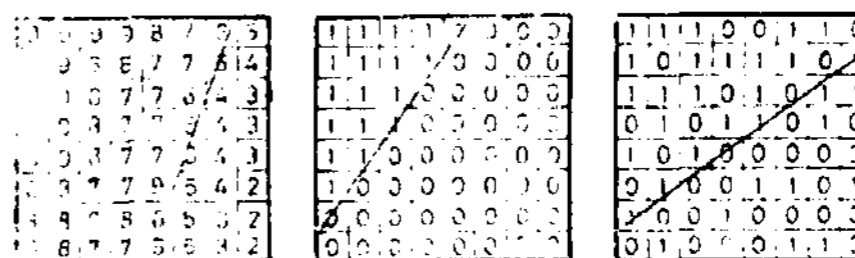Fig 1

Fig 2

Fig 3

Fig 4

Fig 5

Fig 6

Fig 8

Fig 9

Fig 10

ISOLATING AND IDENTIFYING OBJECTS IN LINE DRAWINGS

Edward L. Morofsky
Forest Management Institute
Environment Canada
Ottawa, Ontario K1A 0W2

Andrew K. C. Wong
Biotechnology Program
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

The analysis and representation of complex line drawings is treated and applied. A method for rating the complexity of drawings is developed and compared with the difficulties of human subjects in memorizing them. The problem of isolating and recognizing these memorized figures embedded in more extensive patterns is then investigated. Differing types of embedment are defined by their effect on the original components of the memorized figures. Degrees of embedment are calculated for each memorized figure and compared with experimental results. Predictions are made for human difficulties in recognizing a simple figure under various conditions of embedment.

Introduction

The recognition of a familiar figure embedded in a more extensive pattern is not only an interesting perceptual task, but more nearly recreates the conditions under which most human perception is accomplished. Thus, a critical test of any descriptive framework that hopes to simulate human performance is its ability to isolate and recognize figures embedded in a scene. For figures embedded in a complex drawing, isolation of the figure must precede identification. Isolating embedded figures varies in difficulty, for both human and automated systems, depending on the extent to which the familiar figure has been incorporated into the drawing. If the component parts of the familiar figure survive intact, isolation of the figure is straightforward. This is a simple elaboration of the figure without a change in its structure. When the components of the familiar figure are merged, wholly or in part, into new components isolation becomes more difficult, sometimes virtually impossible.

This paper attempts to employ the methodologies of the Pattern Perception System, PPS, to recognize familiar figures embedded in more complex line patterns. The PPS representations are used in an attempt to measure the relative complexity of patterns as indicated by the difficulties of human observers in memorizing their structures and to predict the observer's ability to recognize a familiar figure embedded in an unknown one. A more intensive introduction to PPS is available in references (1) and (2).

Analysis of a drawing

PPS analyzes line drawings into their constituent components or structural units. The analysis is not unique, but may be accomplished in several ways resulting in a different set of components. These components are intended to correspond to the way in which an observer might copy a drawing on paper in steps during which contact between paper and pencil is maintained for each component, with no backtracking or retracing. Of course, one must question whether the resulting components make possible a descriptive structure that allows efficient processing of descriptions and corresponds with human structuring inferred from psychological experiments. The specific psychological evidence that served as the motivating influence on the design of PPS is treated in (3). The present paper investigates the suitability of the descriptive structure for identifying embedded figures.

The input to PPS is a digitized array of 1 and 0 elements obtained from a line drawing by a modified form of the grid intersection method (4). The trace algorithm proceeds from element to element recording its path as a series of bearings and applies junction heuristics when alternative paths are present. Components are open or closed lines that together comprise the drawing and whose interrelationships form the basis for the representation and identification of the drawing. A junction occurs at any point where a choice is available to the trace algorithm, usually at the joining or intersection of two or more components. A free end is the termination point of a component where no junction is present. The types of junctions considered in PPS are the T, K, Y and X junctions. The drawing of Figure 1 has been segmented into five components A, B, C, D and E. A is a quadrilateral inscribed in the quadrilateral B and together making the four K-junctions 1-4. The component C makes two T-junctions (5 and 6) with B, and C terminates at both. D and E make T-junctions (7-8) with C and terminate at the free ends 9 and 10.


Figure 1
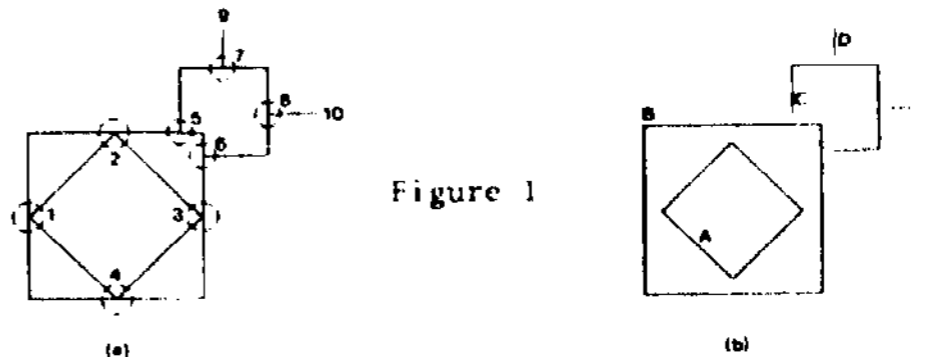
a) a line drawing with junctions 1-10 indicated.
b) drawing of a) segmented into components A-E.

Two main objectives commonly influence the decisions taken by the trace algorithm at junctions —continuity and closure. Continuity uses the various bearings available at the junction in deciding which components continue and in what direction and which terminate. This decision is based on minimization of bearing change between entering and leaving the junction of one or more of the components. On such a basis the decision at the various types of junctions are easily resolved as shown in Figure 5. In separating the intersecting squares into two components the
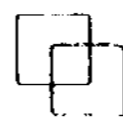
continuity criterion would be used, but for the squares touching at the corners continuity would yield one self-intersecting component. To

separate the squares the junction must be traced as
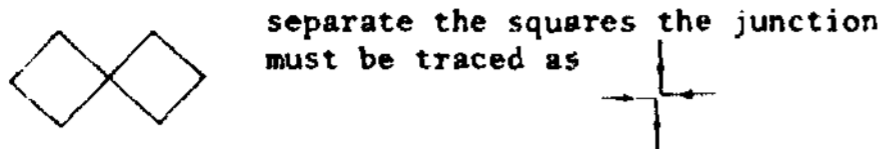
This strategy is termed closure because it seeks to close the component being traced without regard for bearing changes.

A potential factor that may occur in the segmentation of a drawing is the outer boundary, which is a closed line that bounds the entire drawing. Human subjects frequently select the outer boundary as a component (5), when one is present, especially if it also satisfies the continuous criterion at junctions or is itself a familiar shape such as a square or circle. These various possibilities are illustrated in the analyses of the drawing in Figure 2.
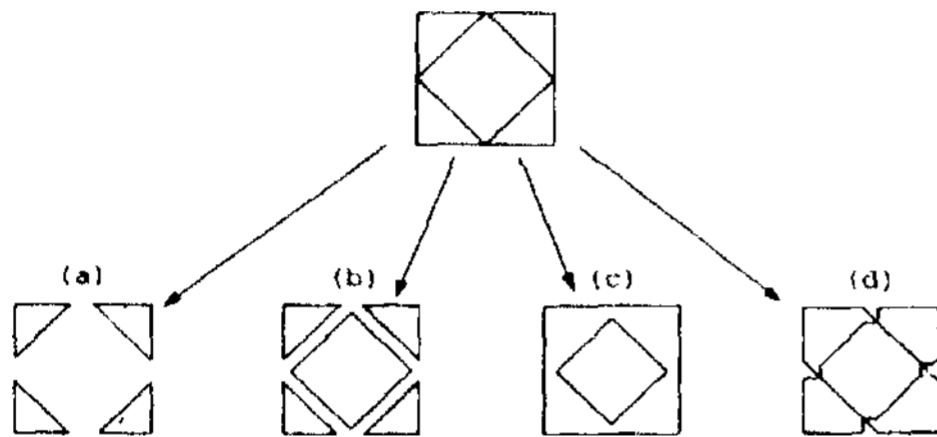
Figure 2. Decomposition of a figure by various tracing strategies.

Closure produces four triangles in (a) with large changes in bearing at the K-junctions. A variant of closure that produces closed shapes bounding every elemental area is known as the elemental area trace (b) and is simply the outer boundary trace with a changed criterion of choice at the junctions. It involves the retracing of some lines and corresponds to outlining every piece of a puzzle. The outer boundary trace produces a square inscribed within another in (c), while continuity will give either (c) or (d) since the total bearing change at the K-junctions is the same in both. Since the resulting components of (c) are more familiar or "simpler" figures than those of (d), it is the more likely analysis. The possible strategies at the various junction types are illustrated in Figure 3.
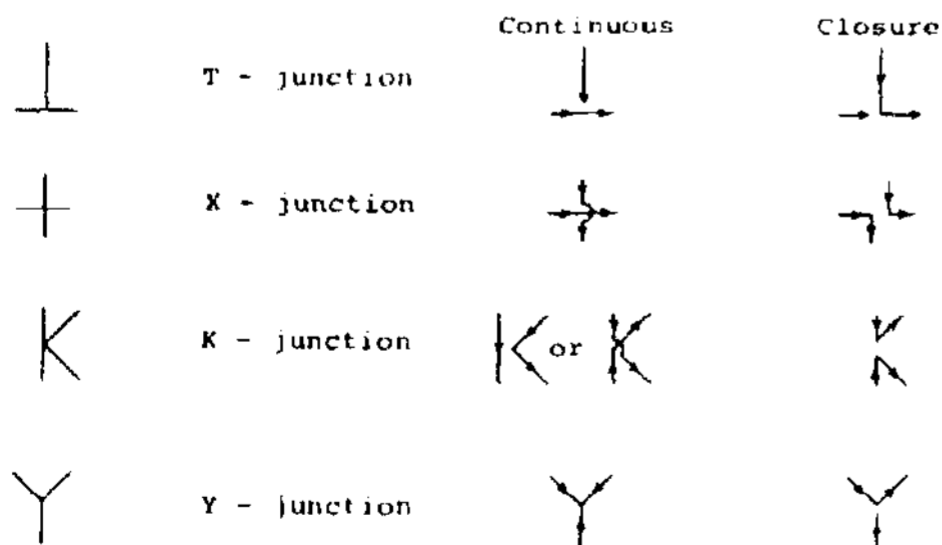

Figure 3. Results of tracing strategies at the four function types.

In PPS, the interrelations among the components form the structural description. The components isolated by PPS may be likened to the units used by human subjects when reproducing a pattern. Each stage in the reproduction corresponds to an additional component. The most primitive level of analysis is the constituent features of each component, i.e., straight lines and segments of varying curvature. The patterns of Figure 4 are first segmented into components and these are then subdivided into features. Each component can be represented generally by a feature code, which is merely the product of prime numbers that have been assigned to each primitive feature and other properties as follows.

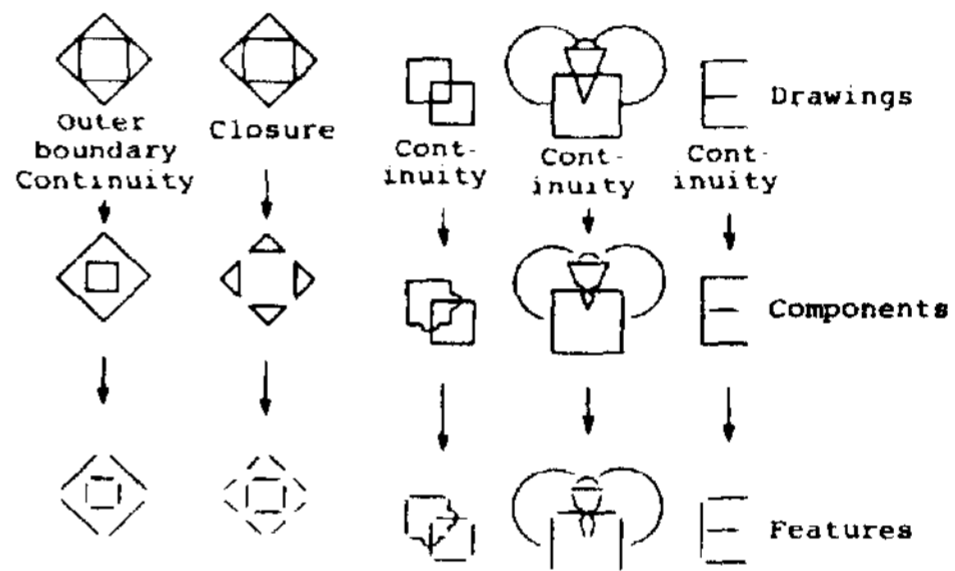|	
closed component	– 2
contains straight line	– 3
contains curved segment	– 5
convex curve	– 7
concave curve	– 11
component self-intersecting	– 13
contains an angle	– 17

Figure 4. Analysis of patterns into components and features as defined by PPS.

A square and a triangle would both have a feature code represented by 2 x 3 x 17 = 102 indicating that both are closed, contain straight lines and angles, but gives no information on the number of the respective features, their size or the sequence of their occurrence – thus, equivalent codes for the square and triangle. However, this code is quite sufficient to differentiate the following component from either of the above and to indicate the sole non-shared feature among them is the presence of a convex curve feature, since 714/102 = 7. Then, to test any component for the presence of a feature or property listed above simply divide the code by the primary number

2 x 3 x 7 x 17 = 714

corresponding to the feature and a zero residue indicates a positive result. The particular format employed is a product of the APL environment in which PPS is presently implemented, but the philosophy of having a dual level of description where the initial level reduces considerably the search space for the more exact description is a basic consideration of PPS,

Structural description of a drawing

Components resulting from the analysis are grouped into a hierarchy that represents the meaningful structural interrelationships among the components. This structure is flexible enough to allow additional components to be added or old ones removed and can represent a single figure or multiple figures comprising a scene. It also allows efficient processing of descriptions in such operations as matching patterns for correspondence. The construction of the description is treated in this section.

Hierarchical structures are composed of multiple levels each occupied by one or more components. Each component is designated by a hierarchical label (X Y), indicating the Yth component on the Xth hierarchical level. The components are grouped about the first level of the hierarchy and are assigned levels on the basis of their relationship to it. When the first level is occupied by one component it is referred to as the reference component or with multiple components, as the reference group. The simplicity criterion is introduced as an initial guide in choosing a reference. The criterion chooses as reference the component that when acting as reference results in the lowest average hierarchical level for the structural description. A drawing is analyzed into components A, B, C, D, E, F, and G. Figure 5 indicates two possible hierarchies. When structured with A as reference it has an average hierarchical level of 17/7, while the structure with C as reference has 18/7. The simplicity criterion indicates component A as the reference. Although often coinciding with human judgement, it is only
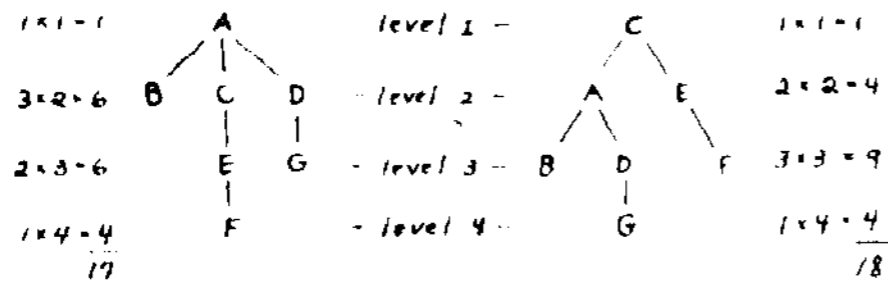

Figure 5

a guide that is often modified by factors such as the relative sizes and the shapes of the various components. Of course, one structure can be translated into other possible variants if need be. If structures are to be compared and classified, a consistent method of representation is necessary. Elaborations of a known pattern should not change the basic structure, nor should minor distortions, translations or rotations. These goals are accomplished in PPS. The drawing of Figure 6 illustrates the occurrence of a reference group, in this case the components A, B and C. They cannot be differentiated by the simplicity criterion and are symmetrically related one to the other.

The component or components acting as reference are assigned hierarchical level one. Components sharing a junction with level one components are assigned hierarchical level two and, in general, components are assigned a level greater by one than the lowest level component with which they share a junction. This is illustrated in the drawing of Figure 7 which is analyzed into components
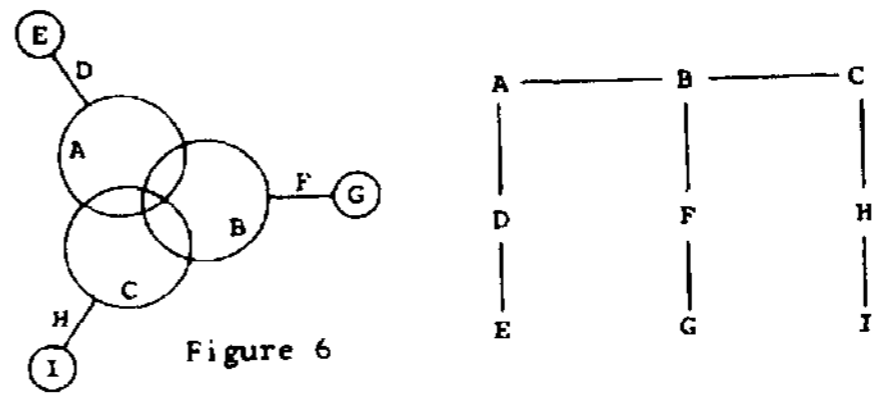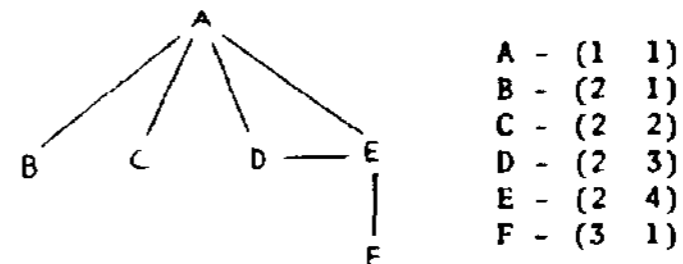

Figure 6

and grouped into a structure with hierarchical labels assigned.

A -	(1 1)
B -	(2 1)
C -	(2 2)
D -	(2 3)
E -	(2 4)
F -	(3 1)

Component A is the reference and all components sharing a junction with it are level two, component F is on level three.

The structural relations between components are then characterized. The basic relation is between a pair of components that share a junction. The components are referred to as a junction component pair of form (k Z) (m n), where k and m are hierarchical levels. Indices of the two components are given by l and n. The reference forms a junction component pair with each second level component in turn. Pairs may also be formed between components on the same hierarchical level. The relation between components indicates the type of junction and the relative relationship of the second member of the pair to the first or, alternatively, of the referent to the referee. The junction code factors which represent these relations are
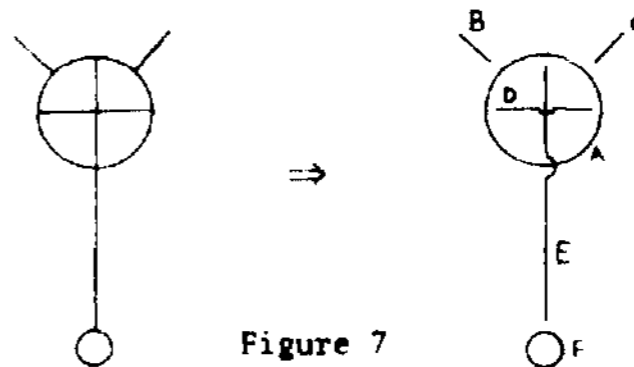

Figure 7

T-junction	- 2
Double junction	- 3
X-junction or intersection	- 5
Outside of a closed referee/negative side of an open referee	- 7
Inside of a closed referee/positive side of an open referee	- 11
K-junction	- 13
Y-junction	- 17

Thus, for the three component drawing shown in Figure 8 the junction component pairs are arranged
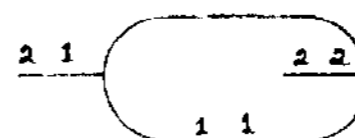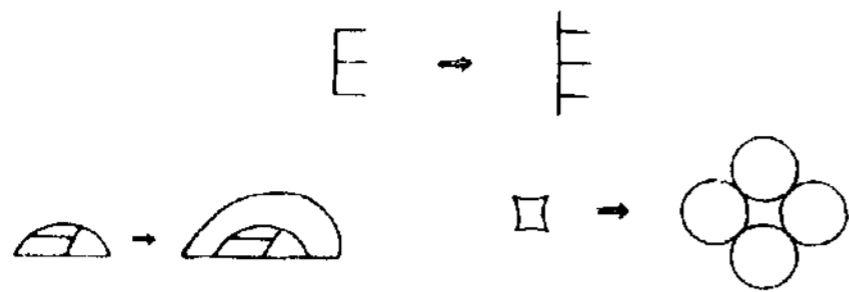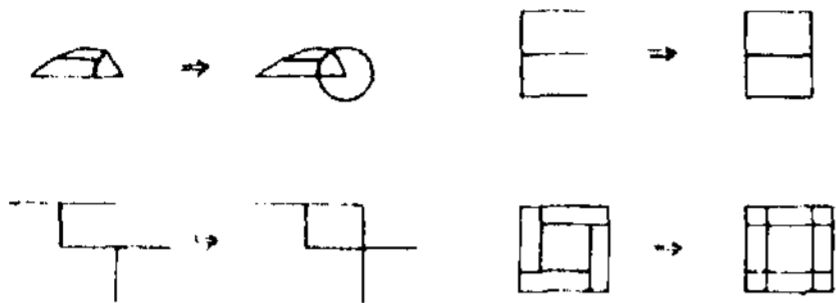

Figure 8

c) simple figures altered in such a way that a previous component(s) is wholly incorporated into a new component. Most effective if applied to the reference component.

The general applicability of these classifications to the human task should be readily apparent. It is assumed embedment of type (c) is most effective and (a) is relatively ineffective. This method is applied to further results of Djang to test its validity. The twelve complex figures shown in Figure 13 were presented to observers who

twelve complex figures and the ranking assigned to each embedment by the PPS classification. The classification indicates the relative difficulties quite well except for the complex figure POQ, which contains the simple figure KEZ. Again the familiar subgroups of this figure have resulted in a deviation from the classification assuming nonsense figures.

KOJ	QOW	POQ	KEH	GID	XUR	GAP	ZUB	CEF	GIN	ZIF	LAJ	simple figures
QEN	WEZ	KEZ	SIJ	BOF	MYD	HIV	DAB	OUQ	JYC	YOP	GAH	complex figures
2b	3b	5bIC	5b	4b	4b	4bIC	5b	4b	5bIC	5bIC	6b	difficulty rating
.87	.81	.72	.72	.53	.43	.30	.23	.22	.16	.17	.12	percentage of successful recognition

Table II. PPS rating of embedment difficulty and results of Djang experiment.

If applied to "clean" line drawings, rather than to the rather fuzzy dot patterns of Djang, the ratings should be more reliable. Figure 14 contains four figures with an embedded "E". It also shows the PPS continuity components and rates the patterns according to assumed difficulty in isolating the "E". For a different approach to these patterns, see [6]. The ranking certainly seems to correspond to human difficulties in this case.
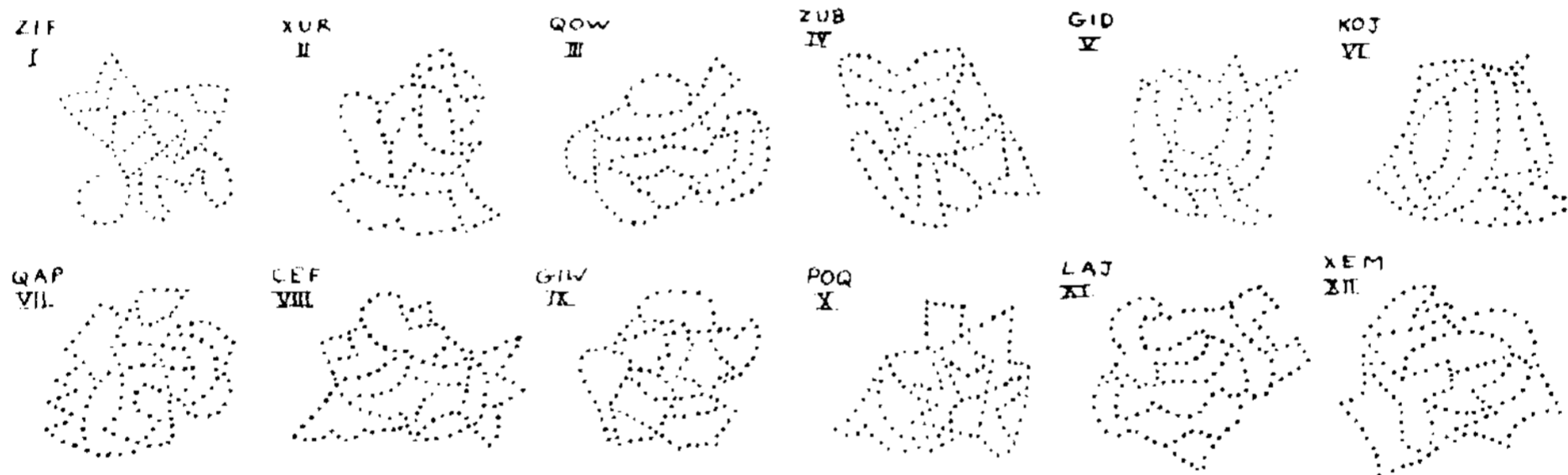
The type of matching illustrated in Figure 10

Figure 13. Twelve complex figures of Djang. Embedded in each is the corresponding simple figure from Figure 11.

memorized their structure. Each contains one of the formerly memorized simple figures of Figure 11 embedded within it. The occurrence of the simple figures as structural units in the reconstruction of the complex figures by the observers was noted and varied widely from figure to figure. These results prompted Djang to conclude that "... past experience plays an important role in our visual organization". The evidence indicated that the difficulty of recognizing embedded figures was related to complexity of both the masked-figure and the figure containing it, although the relation was not explicit enough to interpret the results of the experiment or predict the results of further experiments. Djang understood that the results depended on the "... relative ease of dissolving the different test-figures into the simple-figures and the parts making up the camouflage". Table II indicates the ability of observers to recognize the embedded figure in each of the

and defined in the previous section cannot be applied without modification when a figure is embedded by types (b) and (c) where some or all of the original components may have been incorporated into new components. At this point PPS, like the human observer, adopts a search heuristic in which the complex pattern is taken apart and reconstructed in a variety of ways. PPS tries the various trace strategies, yielding differing component sets, and attempts to spot a likely reference component from which to construct the embedded figure.

659

$D_{l-1} \approx D'_{l-1}$ and for all $d_l \in D_l$, there exists $d'_l \in D'_l$ such that

a) d_l [5] = d'_l [5]

b) there exists a row $d_{l-1} \in D_l$ and $d'_{l-1} \in D'_l$ such that

d_{l-1} [3 4] = d_l [1 2] and

d'_{l-1} [3 4] = d'_l [1 2]

c) d_{l-1} [6] = d'_{l-1} [6], or,

d_l [6] = d'_l [6]

Condition (a) requires equality of feature codes between corresponding rows of the descriptions, while condition (b) requires equivalence in linkage fields between levels $l-1$ and Z. Condition (c) is the junction equivalence for the two types of row elements. For topological equivalence condition (a) is relaxed to the presence of feature factors 2 and 13. For identity the hash coded features must be compared in each component for type, number, order and size. Clearly the lesser degrees of correspondence are least subject to transformation. Topologically equivalent descriptions may be rotated, stretched, bent, etc. and remain topologically equivalent as long as the junctions remain undisturbed. Figure 12 illustrates three types of correspondence that may occur between figures.
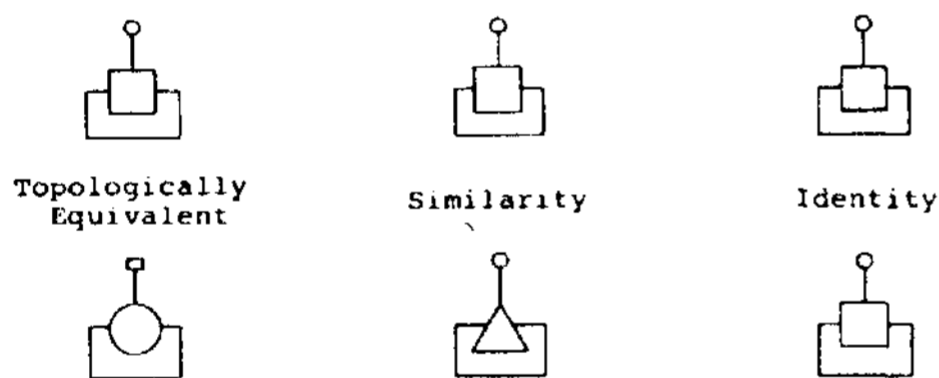

Figure 12. Three types of correspondence between figures.

The above matching algorithm is used as the basis of scene analysis and the isolation of embedded figures of type (a) as defined in the next section. The algorithm can be illustrated by reference to the matching components in the two descriptions of Fig. 10. Let the description of the pattern of Fig. 8 be D and that of Fig. 10 be D', then for l - 2,

a) d_2^3 [5] = $d_2'^5$ [5] = 3 and d_2^4 [5] = $d_2'^7$ [5] = 3

b) d_1^1 [3 4] = d_2^3 [1 2] = 2, 1

d_1^2 [3 4] = d_2^4 [1 2] = 2, 2

$d_1'^1$ [3 4] = $d_2'^5$ [1 2] = 2, 1

$d_1'^3$ [3 4] = $d_2'^7$ [1 2] = 2, 3

c) d_1^1 [6] = $d_1'^1$ [6] = 14

d_1^2 [6] = $d_1'^3$ [6] = 22

Embedded figures — their isolation and recognition

The description £ of a line drawing may be composed of one or more separable figures. A description containing multiple figures is referred to as a scene. The identification of the constituent figures of a scene is naturally a more difficult problem than identifying an isolated figure. It involves both the segmentation of the scene into separate patterns and their assignment to pattern classes. Normally this segmentation must be accomplished before classification can begin. However, PPS performs both the separation and classification simultaneously by using the analysis-by-synthesis approach. That is, an identifiable figure is synthesized or constructed from components contained in D' until a definite classification is made — this is the first step in the analysis of D into its constituent figures. Thus, the isolation is accomplished by a synthesis or clustering [6] of available components to construct pattern classes in the universe of PPS. This synthesis can begin at the reference component of other "nucleation" components that display the properties of a likely foundation.

While a scene implies the presence of multiple patterns, an embedded figure indicates a known figure "hidden" or camouflaged in a more complex pattern. This section investigates the ability of PPS to recognize embedded figures and compares this to the ability of humans in similar tasks. A method, based on PPS, is proposed for rating the difficulties involved in isolating embedded figures.

The ability of PPS to extract or identify a familiar embedded or simple figure within a complex figure depends on the type of embedding. These degrees of "embedment" are distinguished by their effects on the original components of the simple figure and may occur simultaneously. The difficulty of achieving the isolation and identification of an embedded figure in PPS is dependent on the survival of the components of the embedded figure. If the components remain the same after embedment, then isolation is relatively straightforward.

a) simple figure given additional components without affecting the integrity of the original components,
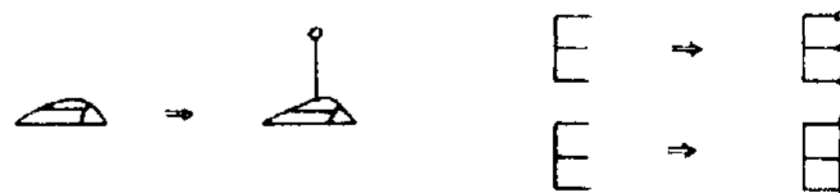
b) simple figures altered such that an original component is partially incorporated into two or more new components, more effective if the new component is smoother, simpler, etc. than the original.
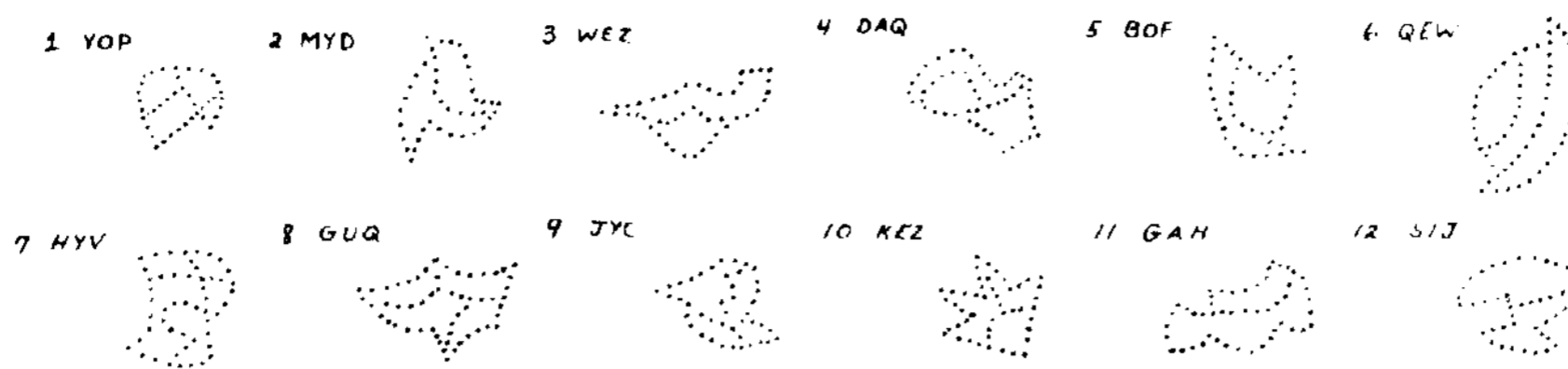
Figure 11. Twelve simple figures of Djang with attached nonsense names.

contribution to complexity is the number of features contained in the figure and the number of hierarchical levels needed to structure the figure. These measures have been combined into

$$\sum_{i=1}^{n} \text{number of features} \times i,$$ where n is the greatest

hierarchical level. Thus, features at the reference level contribute least to complexity. The actual PPS structures used with those given by Djang as the dominant organization used by observers, e.g., figures 8, 9, 10, 11, 12 were organized by giving the outer boundary and filling in the details. The type of intersection also contributes to the complexity by allowing alternate organizations, self-intersection, T-junctions of a component with itself and Y-junctions counting more heavily toward greater complexity. The third measure is the number of elemental areas or separate "pieces" of the pattern. These three measures are computed and combined as shown in Table 1. Remembering that the table is based on

greater than or equal to three, the above scheme or a similar one would be able to predict complexity in controlled situations such as presented in the next section.

Identification by comparison of descriptions

This section formalizes the comparison of two descriptions and the various degrees of correspondence that may exist between them. The next section will apply this method to embedded figures

In matching descriptions to determine their correspondence in terms of structure and features, several degrees of correspondence are utilized, ranging from identity to topological equivalence. Pattern matching in PPS begins at the reference component and proceeds to greater levels of the hierarchy. The following makes this more definite

D, the description of a pattern.

9	8	7	12	11	2	4	1	5	10	3	6	
JYC	GUQ	HYV	SIJ	GAH	MYD	DAQ	YOP	BOF	KEZ	WEZ	QEW	Name of simple figure
a	4.63	4.55	4.5	3.84	3.77	3.75	3.62	3.45	3.17	2.6	2.5	1.84	average no of trials to memorization
b	13	15	10	13	13	11	11	7	10	8	10	6	number of PPS features
c	21	18	11	14	14	11	12	8	10	20	11	4	no of features times hierarchical level
d	0	0	7	0	0	2	2	4	2	0	0	0	weighted intersections
e	21	18	18	14	14	13	14	12	12	20	11	4	row c + row d
f	3	4	5	2	2	2	3	4	2	3	2	3	elemental units
g	24	22	23	16	16	15	17	16	14	23	13	12	row e + row f

Table 1. Twelve simple figures of Figure 11 ranked according to PPS attributes.

one dominant organization among various used by the observers and the possible influence of familiar portions or whole figures, the results of Djang roughly correspond to the difficulties calculated. The highest and lowest scores go, respectively, to JYC and QEW with corresponding greatest and least difficulties. The other figures are partitioned into three groups with similar scores with the exception of KEZ. It is assumed that KEZ contains some familiar sub-figures which have effectively reduced its complexity and, indeed, if we imagine that one sees in KEZ the letter "M" sitting on a triangle — the score is reduced to 12 which fits nicely with its neighbours. Line drawings, rather than the dot patterns, would restrict the number of structures possible and clarify some junctions. It is thought that with the number of components

$\underline{D}_n$, a description with greatest hierarchical level of n.

d^i, the ith row of the description $\underline{D}$.

d^i_r, the ith row with $k^i = r$, where k is the hierarchical level of the referee.

A sub-description of $\underline{D}$, of l levels, is defined as $\underline{D}_l = \underline{D}[l;]$, where l represents the row indices of all $d^i_r \in \underline{D}$ such that $r \leq l$.

The similarity relation between a description $\underline{D}$ and another description $\underline{D}'$, $\underline{D} \approx \underline{D}'$, requires that at any level l of the match process, with $l \neq 1$ and $l \leq n$, $\underline{D}_l \approx \underline{D}'_l$. $\underline{D}_l \approx \underline{D}'_l$, if and only if,

661

in rows starting with the reference and including all components. The f and g for each row are the feature code and junction code. The junction code

	referees							
referees	1	1	2	1	f^1	g^1
referents	1	1	2	2	f^2	g^2
	2	1	1	1	f^3	g^3
	2	2	1	1	f^4	g^4

for the component pair (1 1) (2 1), or g^1, is 2 x 7 = 14, indicating a T-junction with the referent outside the referee. For the component pair (1 1) (2 2) the junction code, or g^2, is 2 x 11 = 22, indicating a T-junction with the referent on the inside of the referee. Filling in the f's and g's of the above array results in the following description,

$$\underline{D} \equiv \begin{array}{cccccc} 1 & 1 & 2 & 1 & 14 & 14 \\ 1 & 1 & 2 & 2 & 14 & 22 \\ 2 & 1 & 1 & 1 & 3 & 0 \\ 2 & 2 & 1 & 1 & 3 & 0 \end{array}$$

where the entries for g^3 and g^4 simply restate the fact that (2 1) and (2 2) have no higher level referents. Such components are called terminators An array consisting of each component acting in turn as referee with all of its referents listed, including feature and junction codes, forms the basis for the description of a drawing. When a referee has multiple referents, they are listed in the sequence they are encountered as the referee is traced, with the stipulation that closed components are traced in a clockwise manner. This is illustrated by the drawing of Figure 9. This
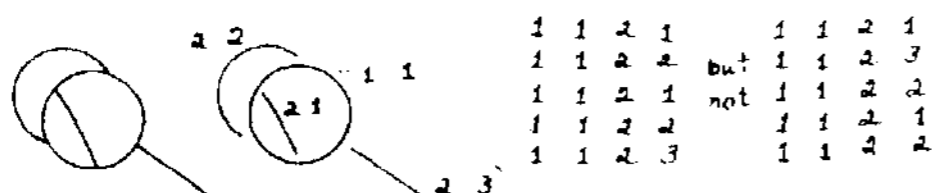

Figure 9.

simply establishes some rule of representation to standardize processing and simplify comparisons. The above example indicates that the description of a closed referee with multiple referents may begin with various component pairs while retaining the role of an equivalent description. PPS automatically sequences the referents of one of the descriptions when two drawings are being compared until a matching sequence occurs. This is necessary because component (2 2) of one drawing may correspond to component (2 3) of a matching drawing.

In addition to the feature code description of each component a more detailed description of every feature contained in the component, in sequence, is necessary. The feature code is simply an abridged version of this description. In the APL implementation of PPS, these descriptions are hash coded and the indices are stored in column seven of the array. This seven column array is known as the description, $\underline{D}$. The detailed feature description stored as hash code is required only when feature and junction code screenings have proved affirmative. Thus, general characteristics are matched and used to eliminate non-corresponding patterns before a detailed match is attempted. Only an identity match requires

inspection of the hash coded features, while for topological equivalence the feature and junction codes are sufficient.

In Figure 10 the drawing of Figure 8 is given additional components and the two descriptions are shown with the matching indicated. This is the basis of detecting embedded patterns or simply familiar subpatterns. PPS performs both the separation and classification simultaneously by using the analysis-by-synthesis approach. That is, an identifiable pattern is synthesized or constructed from components contained in the description until a definite correspondence is established. Non-matching components are simply passed over in the sequential search for matching components. This process is given a more formal basis in the following section.

The complexity of a pattern bears some relation to the number of individual features it contains, the number of components that comprise it and the number of different ways in which it can be organized. It is usually measured by observing the relative difficulties encountered by human subjects in memorizing and reproducing various patterns. The experiments by Djang (5) will be used to illustrate the application of PPS to interpretation of such measurements.

Attempts to quantify complexity are made difficult by the possibility that the figure, or part of it, corresponds more or less closely to a previously known or familiar figure. This usually reduces the complexity of a figure, although not always (5). The observer will often try to find something familiar in a novel pattern to facilitate structuring and different observers will see different things. The initial analysis and representation of PPS are based on a novel pattern assumption although the recognition of a known pattern class can cause a simplified restructuring of the representation. Experiments by Djang involved the presentation of simple figures which were memorized by the subjects and which subsequently occurred embedded in more complex figures.
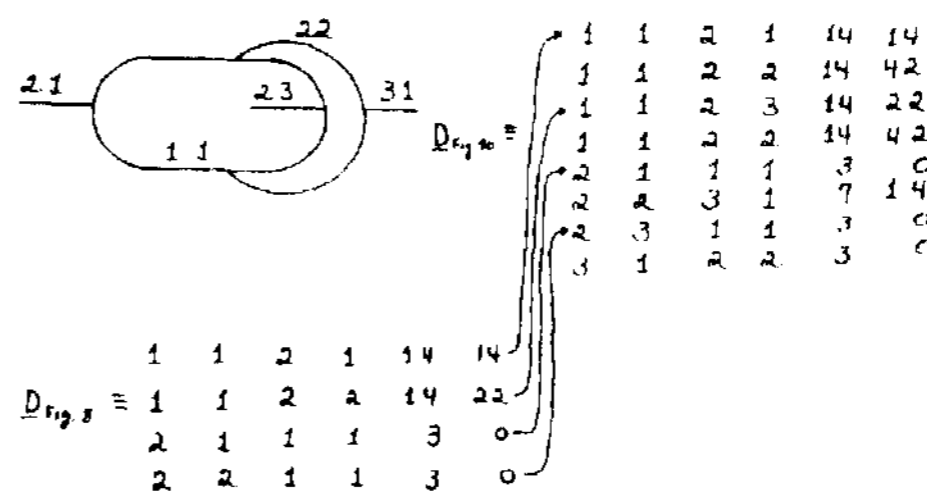

Figure 10. Description of Fig. 8 is matched with the description of an elaborated figure containing it. The matching components are indicated.

The twelve simple figures used by Djang, along with their associated nonsense names, are shown in Figure 11. Table 1 indicates the complexity of each figure as measured by the average number of trials needed to be able to accurately reproduce the figure. Let us see how a PPS view of complexity will rank these figures. The first
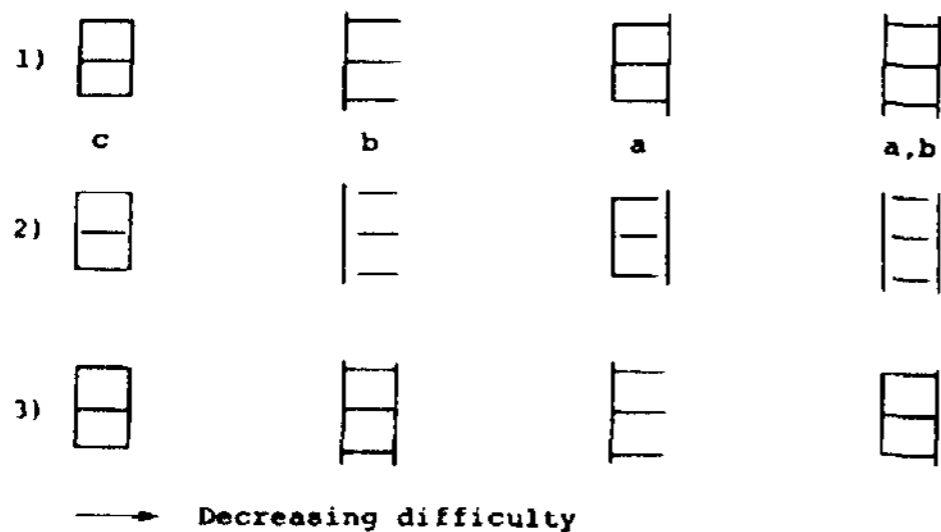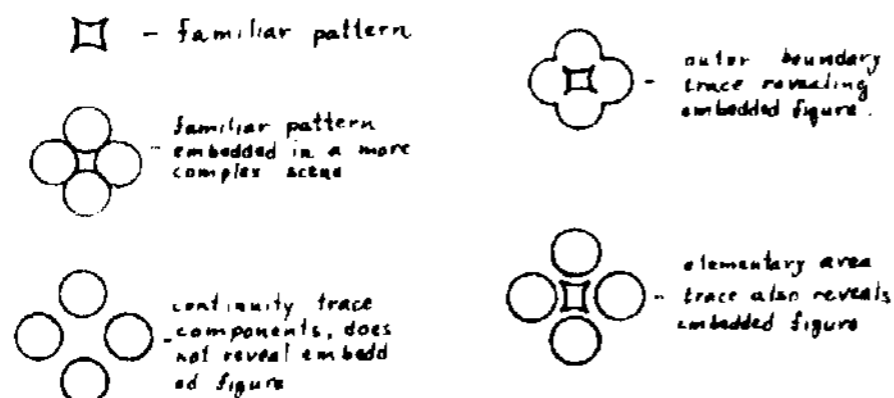
Figure 14. Row 1, four line drawings with an embedded "E", type of embedment shown under each. Row 2 gives the components resulting from a continuity trace. Row 3 shows the drawings arranged in decreasing order of difficulty based on the PPS ranking system.

The following illustrates the usefulness of multiple trace strategies.

A distinction is required between (1) perceiving an embedded figure and using it as a natural structural unit (as was intended in the experiments of Djang), and (2) deliberate search to isolate possible embedded figures. The fourth example under type (c) embedment will readily be admitted to contain the simpler figure, but perception of the embedded figure as a visual unit is highly unlikely. Performance of humans in such tasks is highly variable often depending on their initial organization of the pattern which then becomes difficult to alter.

References

1. Morofsky, E.L. and Wong, A.K.C. Computer perception of complex patterns. Second International Joint Conference on Artificial Intelligence, Advance Papers of the Conference, British Computer Society, 1971, 248-257.

2. Morofsky, E.L. Computer perception of complex line patterns. Doctoral dissertation, Carnegie – Mellon Univ., University Microfilms, 1972, No. 72-29, 862.

3. Morofsky, E.L. and Wong, A.K.C. Interactive on-line perception of line patterns. Behav. Res. Meth. and Instru., 1973, Vol.5(2),139-146.

4. Freeman, H. Computer processing of line-drawing images. Computing Surveys, Vol. 6, No. 1, March, 1974, 57-97.

5. Djang, S. The role of past experience in the visual perception of masked forms. Journal of Experimental Psychology, 1937, 20, 29-59.

6. Rosenfeld, A. and Lee, Y.H. A clustering heuristic for line-drawing analysis. IEEE Transactions on Computers, August, 1972, 904-11.

663