Analysis and Synthesis of Human and Avian Categorizations of Fifteen Simple Polygons[1]

D. B. Mumford, R. J. Herrnstein, S. M. Kosslyn, and W. Vaughan, Jr.

Harvard University

Jan. 9, 1989

SECTION I: **INTRODUCTION**

This paper addresses the problem of how animals categorize visual stimuli according to their shapes, a problem that has proved to be surprisingly subtle. The difficulty may arise in part from the very richness of the stimulus domain. Not only are there any number of shapes, but there are any number of ways to represent a given shape, and some representations will be more useful than others for a particular purpose. This paper examines the performance of natural intelligences, both human and avian, in recognizing a small collection of arbitrary shapes, and compares this with the performance of mathematical algorithms devised for computer recognition of the shapes. The comparison across species may reveal differences in shape representation that reflect differences in the two species′ adaptation to their respective environments. Such findings may have implications for the design of machine algorithms, which could be modeled after the different species, as appropriate for accomplishing different tasks.

The experiments reported here focus on the perception and recognition of shape so that the appropriate stored representations can be compared with each other and with new stimuli. We draw inferences about the nature of shape representation by examining the patterns of confusions made by visual systems when given the task of differentiating among different shapes. The logic of the method is based on the principle that how shapes are represented determines which pairs of shapes look similar and which look different. For example, some schemes for the representation of shapes make mirror-image shapes unlike each other (e.g. any scheme with even crude coordinate localization of prominent parts would do so), but mirror-image confusions are common in human and pigeon performance, though perhaps not equally so (Vaughan & Greene, 1984). This observation underlies the central idea behind the present research: the pattern of confusions displayed by humans and pigeons in experimental situations should allow us to draw inferences about the ways shape is represented by the two species.

The focus of this study is on shape per se: the categorization of single, simple, uniformly colored figures on a contrasting uniform background. Qualities such as color and texture and motion are not varied in our stimuli. Moreover, we are interested here only in 2-dimensional shape, so shading and other depth cues from illumination, texture gradients, occlusion, etc. are excluded. What we are left with are just black and white silhouette images of simple shapes.

Why look at what might seem to be so restricted a world of visual stimuli? One reason is that the essence of vision, the characteristic qualities that make visual input distinctive, seems to be present in even this simple class of images. Another reason is that shape cannot be described easily in mathematical terms, whereas other visual qualities often can be. For example, there is a simple and satisfying way of describing color, at least to a first approximation (e.g., by three numbers representing the amounts of red, green and blue light that must be added to obtain the given color as seen by humans). No analogously straightforward description of 2-dimensional shape has yet been formulated.

To be precise, it is possible to define mathematically a **space** S of all 2-dimensional shapes: each point x of S stands for a particular shape A(x), and every shape in the plane (say with a smooth outline and a finite set of corners[2]) is equal to A(x) for exactly one point x in S. In S one may talk of points x and y being near or far: this corresponds to the shapes A(x) and A(y) being similar to each other or not[3]. Then the question arises: what is the simplest way to define, for each point x in S, a set of coordinates a(x), b(x), ..., so that two points x and y are close if and only if their coordinates are close. First of all, one needs an infinite set of coordinates, not three as with color. Thus one says that the space S is infinite-dimensional. Secondly, the space S apparently is not ´flat´ but has holes.[4] This means that if you find coordinates a(x), b(x),... as above, you cannot require that every possible set of values of the coordinates actually comes from a shape. Thirdly, it is not clear how to find an infinite set of coordinates a(x),..., even with this caveat.[5]

The principal case where confusabilities of shapes has been studied with human subjects is that of the confusions of the letters of the alphabet. Such studies were pioneered by the Gibsons (cf. Gibson, 1969) and continued by many others (e.g. Coffin, 1978; Holbrook, 1975; Keren & Baggen, 1985; Podgorny & Garner, 1979; Townsend, 1971; Wolford, 1975).

It seems fair to say that, despite all the work on letter confusions, there is as yet no good general-purpose theory of shape representation that can be used even to explain patterns of letter confusions for human subjects with a reasonably high degree of precision. The problem with the available theories may be with their formal structure as such, or may be with the particular choices made to make the theory precise. For example, the notion that shape can be appropriately described in terms of spatial relations among parts or "features" may be correct, but the appropriate relations and features may not yet have been identified.

Recently, Blough (1985) taught the roman upper case letters to pigeons and recorded their errors during learning. He compared his results with the human alphabet confusion data, finding substantial commonalities between the two data sets. It may seem odd to suggest that such inter-species comparisons may be illuminating. However, recent work on the pigeon´s extraordinary capacity for visual classification implies that the species possesses mechanisms of great power, comparable to those of the human system and well beyond anything yet achieved by artificial systems. The pigeon (and other birds) have a brain structure, the Wulst (see Karten, Hodos, Nauta & Revzin, 1973; Pasternak & Hodos, 1977), that may give it a special visual capacity as an adaptation to the visual way of life of this class of animals. With a relatively small nervous system especially adapted to visual perception, and its proven capacity to match at least some of the human ability to classify shape (see also Herrnstein, Loveland, & Cable, 1976), the pigeon may be an exceptionally useful model for the development of artificial visual processing systems.

Our working hypothesis is that the problem of shape representation is basically the same for the vision of a computer, a human, and a pigeon, in the broad sense that each visual system must solve the problem of extracting invariance from varying exemplars. Although our research has an inductive component -- the point of departure is a set of confusions for human or pigeon subjects -- it is also guided by theories of shape representation. We are particularly interested in whether specific representational schemes will prove useful for interpreting the confusion data, and whether the same scheme will apply to both human and avian data.

The main types of shape representations that have been proposed may be classified as follows:

I) **Geometric data structures**

a) **Two-dimensional structures**: These representations are two-dimensional arrays, whose elements are called pixels, which may simply record which points are in and which are out of a shape or may result from further processing such as filtering, thresholding, segmentation, or more abstractly "annotation" (which to say, the assigning of special labels to particular pixels)[6]. Comparison of this sort of representation with stored information is usually done via template matching. Most theorists argue that this representation is not appropriate for the end-product of the encoding process in humans (e.g. Lindsay & Norman, 1977), but nevertheless may be used in some tasks (cf. Lowe, 1987a and 1987b; Ullman, 1986).

b) **One-dimensional structures**: The boundary of the shape may be described by splines, chain-codes and "image files" (as in Kosslyn, 1980). They can be compared directly or indirectly via generation of an intrinsic image. The comparison algorithm is either some sort of template match or the use of grammars that generate all descriptions in each class to be recognized (Fu, 1982).

II) Combinatorial data structures

a) Feature lists: These representations are used in the classical approach, either specifying necessary and sufficient features (a la Aristotle) or more complex weighted sets. There is no organization among features in this scheme. The comparison algorithm here is a simple part-for-part match.

b) **Structured networks**: These representations make explicit the most salient parts of the shape, which are linked by a small class of universal relations (e.g. "attached to", "right of", "inside", etc.). The whole representation is a tree or a graph, often organized hierarchically, (cf. e.g. Palmer, 1977; Reed, 1974;

Reed & Johnsen, 1975; Marr, 1982). Comparison here involves graph matching, by either combinatorial algorithms or by relaxation.

III) **Distributed, non-locally interpretable data structures, Boltzmann machine, and other neural networks.** Here information is stored in the weights of the connections in a network, and matching is done by ´running´ the network, often finding the best (or "lowest energy") fit between a pattern of input and a pattern of activation in the network (as in Rumelhart & McClelland, 1986). If this model is correct, it may be impossible to infer the nature of the represented features or dimensions that underlie confusions; they may be accidents of the "weight space" formed by the network, with no particular semantically interpretable meaning.

In order to identify the representation scheme used by humans and pigeons, we need to have definite quantitative predictions of what kind of performance to expect if a visual system uses one or another scheme. To get these, we have implemented computer programs based on several of these schemes, implementing them in the ways which seemed most plausible in light of the data. Details will be given below in the sections devoted to each scheme.

In short, we chose one set of simple shapes and presented these as stimuli i) to pigeons in a learning experiment, ii) to humans in a discrimination experiment and iii) to computer programs for matching. In each case, we obtain a table of either errors, response times, or degrees of match. These tables are analyzed by the same techniques and directly compared with each other. Our goal is to clarify the similarities and differences in these patterns of data. Of course, computer programs to match two shapes typically have various parameters in them, as well as numerous variants, in which similar but not identical procedures are followed. Thus, one of the main parts of our analysis has been to explore the choice of parameters and variants in such a way as to find a computer program of the class in question that best predicts the human or pigeon data.

SECTION II: METHOD

Pigeon Experiment

Subjects. Six adult male White Carneaux pigeons, maintained at about 80% free-feeding body weights, served as subjects. None had worked previously in any experiment requiring complex visual-form discriminations.

Apparatus. The chamber in which the birds were tested was 30 cm long, 30 cm wide, and 33 cm high. Centered on the front wall was a 6.3 by 4.4 cm translucent panel, onto which slides could be back-projected and which, when pecked by a pigeon, sent a signal to the computer. On either side of this screen panel were standard pigeon keys, of which only the left one was used. Below the panel was the opening for a standard mixed grain hopper. This chamber was enclosed within a sound-attenuating plywood shell, with a fan that provided masking noise. A Kodak Carousel projector was mounted outside the shell, and projected onto the screen panel through a hole. The experiment was controlled by means of a PDP-8/e computer running SuperSKED software.

Stimuli. The stimuli used in this experiment consisted of 35mm slides of 15 solid black polygons on a white background, illustrated in Figure 1; all of the polygons have the same perimeter. They were constructed so as to fall, intuitively, into three classes of five polygons each. The first class consisted of nearly square shapes, some with corners removed and some with one edge partly deformed. The second class consisted of diamond, and modified diamond, shapes, whereas the third consisted of modified and/or truncated isosceles triangles. The size and orientation of each stimulus were not varied, though their location was (see below).

Insert Figure 1 about here

Procedure. At the beginning of each session, a response key to the left of the main screen key was illuminated red. A single response to that key produced three sec access to mixed grain. Four sec later the first slide was shown. Pecking responses to the screen were recorded for further analysis during the first 10 sec of presentation of a slide. Following that a variable-interval schedule of reinforcement with an average interval of 10 sec and a range of 1 to 35 sec (VI 10 sec) was in effect. At the completion of an interval in the VI, if the slide was positive (i.e., contained the positive polygon), a response within two sec of the preceding response (which could occur prior to the end of the VI) operated the food hopper for three sec. When the food cycle concluded, the projection lamp was turned off for four sec, then turned on with the same slide in place. After 10 sec (during which time responses were not recorded), a response within two sec of the preceding response again produced food for three sec. After this second reinforcement, the projector turned off, and four sec later a new slide was shown. If the bird failed to produce reinforcement 60 sec after timing out of the VI schedule, food was presented automatically, and events proceeded just as if the bird had

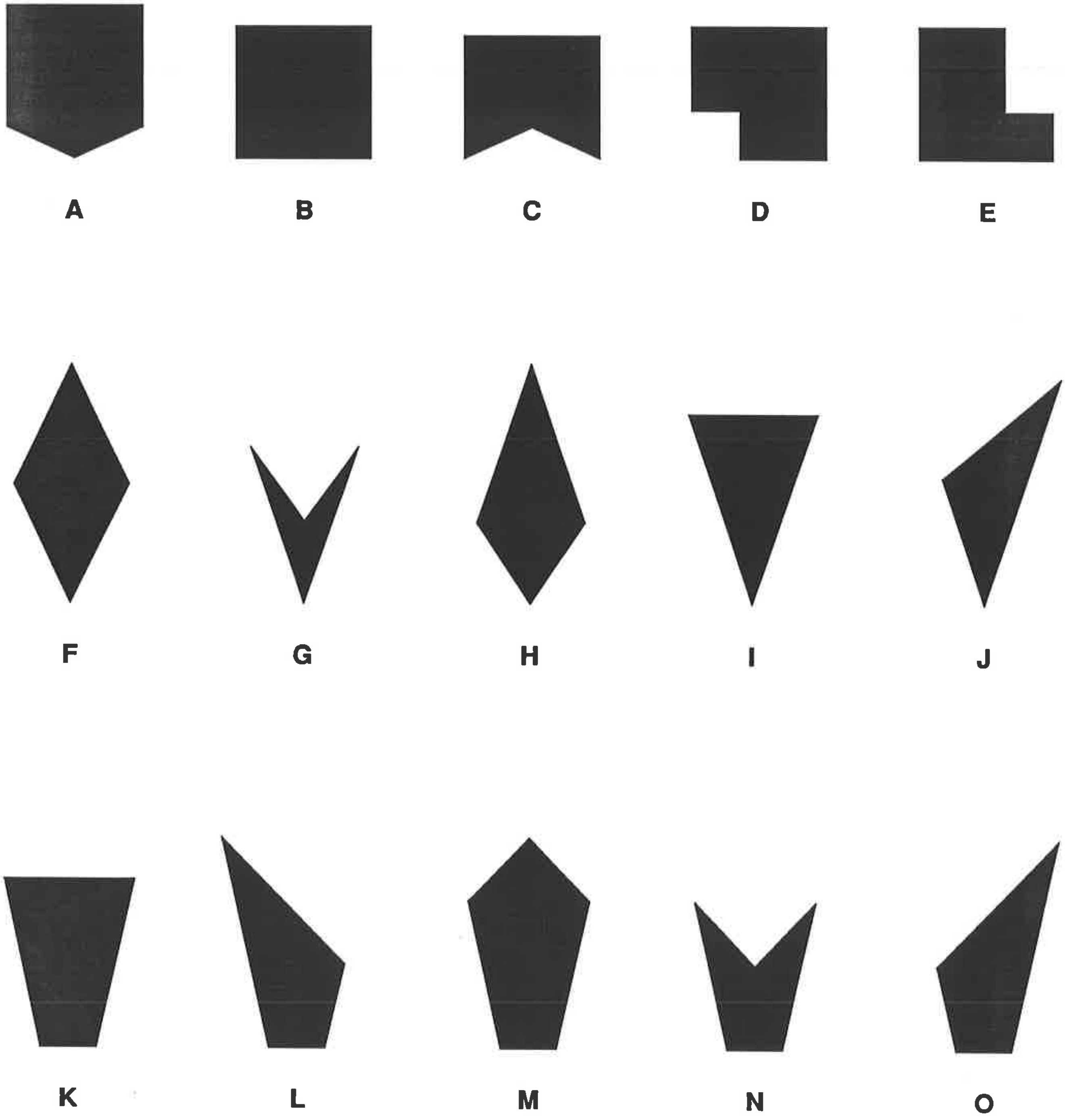


Figure 1

in fact produced the food. In the case of a negative slide (i.e., containing a negative polygon), responses were recorded during the first 10s of slide presentation. Next, a VI 10s was initiated, and, at the end of an interval in the VI, after five sec had elapsed without a response, the projector went off, and four sec later the next slide was shown. Thus, a negative slide could remain indefinitely, until a five sec interval elapsed between consecutive pecks. All analyses are based on pecking during the initial 10 sec of a slide´s presentation, prior to reinforcement or to the operation of the VI.

There were 75 slides in the slide tray, five copies of each of the 15 stimuli. The copies differed only in the precise placement of the polygon, which was either central, or slightly above, below, or to the left or right of, the center. The point of this variation was to prevent the pigeons from focusing on specific locations on the screen. The slides were shown in a different order in each session. The only constraint on the order of slides was that each set of 15 slides (e.g., slide sets 1 to 15, 16 to 30, and so on) contained one of each of the 15 classes (by "class" we mean the set of five exemplars of each polygon). It was thus possible for the same class to be repeated twice in a row, across the boundary between sets of 15 (e.g., the fifteenth slide might be from class 4, as well as the sixteenth slide). At the end of the first revolution in a session, the left red key was again illuminated, and a single peck at it produced three sec of food; four sec later the first stimulus was again shown. At this point the positive class could be different, depending on contingencies described below.

Each session comprised two revolutions of the slide tray, with the possibility that a different polygon could be positive during the two halves of the session. Starting with session 25, a new positive polygon was introduced as follows. During the first revolution of a session, the same polygon was positive as on the last revolution of the last session. On the second revolution, if the polygon had first been made positive on the previous session, it always remained positive. If it was first made positive two sessions previously, a new positive would be selected only if the ratio of average pecks to positives to average pecks to negatives exceeded 3. If it was first made positive three sessions previously, the criterial ratio was 2.5. If it was first made positive four sessions previously, the ratio was 2. If it was first made positive five sessions previously, the ratio was 1.5. If it was first made positive six sessions previously, a new positive polygon was selected, whatever the ratio of positive to negative pecks. In this way, pigeons that learned rapidly would advance through the set of positive polygons quickly. Slow learners would advance more slowly, but there was a minimum rate at which all pigeons were forced to advance.

The order of positive polygons was constrained so that if two polygons were consecutive in either direction for one pigeon (e.g., following polygon 7 as positive, polygon 4 was positive, or vice versa), they could not be consecutive in either order for any other pigeon (for no other pigeon did 4 follow 7 as positive, or 7 follow 4 as positive).

Human Experiment

Subjects. Twenty-eight Harvard University students (14 males, 14 females) volunteered to participate as paid subjects.

Stimuli. The same polygons were used for the human subjects as for the pigeons, with the only difference being that each was slightly displaced to the left of the center of the projection field of the slide projector.

Procedure. The subjects were asked to view a series of polygons back-projected through a translucent screen. The subjects sat in front of the screen, and the figures subtended an average of 5° degrees of visual angle. The subjects were shown a polygon and told that it was the standard for the next series of test trials. The task was to decide whether each polygon in the next 21 trials did or did not match the standard. If it did, the subjects were to press one response key; if it did not, they were to press the other key. Each of the other 14 polygons appeared once in a series and the standard appeared seven times; there was thus a two-to-one ratio of "different" to "same" trials. The subjects were not told this ratio and were asked to make their responses as quickly and accurately as possible.

In order to help the subjects learn the appearance of the standard stimulus, we asked them, first, to study it, and then to close their eyes and form a visual mental image of it. Following this, the subjects were to open their eyes and compare the mental image with the actual polygon in front of them. Disparities between the two were to be noted, and then subjects were to repeat the process, closing their eyes and forming an image again. A subject would repeat the process until he or she claimed to be able to form an accurate mental image of the standard polygon for the series. Following this were the 21 test trials. An Apple II computer presented a slide by opening a tachistoscopic shutter, which was left open until the subject responded by pressing either of the response keys, at which point the shutter closed. Four sec later a new test stimulus was presented, and so on through all 21 trials. A new standard polygon was then presented and the entire process repeated, until all 15 polygons had served as the standard.

Seven different orders of the standard polygons were prepared, and then these orders were reversed, creating a total of 14 orders. The seven orders of standards satisfied two constraints. First, each standard appeared equally often in the first and final third of the presentation orders. Second, for six of the orders, no consecutive pair of stimuli appeared more than once; in the seventh order, one pair that had been used in another ordering also appeared here. The order of test stimuli was random, with two constraints: a) each slide appeared first in a list exactly once, and b) no ordered pairs appeared more than once in the test sequences, excluding pairs that contained the standard. For each order of standards, the test polygons were presented in a given order once, and once again in the reverse order.

For each of the seven orders in which the standards were presented, two subjects responded "same" by pressing the key under their dominant hand and "different" by pressing that under their non-dominant hand, and vice versa for another two subjects. For a given order, the two subjects responding in a given way received the test trials in the opposite orders.

Subjects were tested individually in a single session typically lasting slightly more than an hour. No feedback was given for accuracy. Because subjects were almost always accurate (see below), response times were the measure of primary interest. The computer running the experiment also recorded responses and response latencies.

SECTION III: ANALYSIS

A) THREE PROCEDURES FOR ANALYZING CONFUSION MATRICES

The present experiments were designed to produce matrices whose rows and columns both correspond to the set of stimulus forms. Patterns of matrix entries represent the relative difficulty of discriminating among the stimuli, as reflected either in the pigeon error data or the human latency data. Matrix entries that are smaller (i.e., fewer errors or, for human subjects, shorter latencies) indicate that the shapes being discriminated are relatively different; entries that are larger indicate that they are relatively similar. The attraction of experiments of this kind is that such a "confusion matrix" contains a great deal of information about the set of stimuli as perceived by the subject, pigeon or human. Patterns in the matrices tell us about which dimensions are being employed by the information processor by showing which stimuli are closest, which stimuli are clustered, and so on.

The totality of all shapes, and in particular of all polygons, forms mathematically an infinite dimensional space (see Introduction); the 15 stimuli used in this experiment probe a few of the dimensions present here. The initial goal of the analysis of these matrices is not to find some specific statistic to prove or disprove any specific hypothesis about confusability, but to see what the data themselves are trying to say. We seek algorithms that can tell us something of the structure of an arbitrary matrix; afterwards compare the results of the analysis with the predictions of various theories of shape discrimination to discover whether the present data support these theories.

In analyses such as this, an important issue is reliability. There are no accepted tests for reliability of multi-dimensional scaling or clustering. Nonetheless, one must worry whether the conclusions are really strongly supported by the data or are artifacts of too much analysis built on too little data. We have two ways of establishing reliability of our analyses. One is the so-called ´bootstrap´, an idea due to Efron (Efron, 1979; Efron & Tibshirani, 1986). If the data are taken from a small set of experimental subjects, say N of them, one samples randomly from the full set of N subjects but **with replacement.** Since sampling is with replacement, when drawing N cases, some subjects will be chosen more than once and others, not at all. If the set of subjects was a fair sample of the population from which it was drawn, this procedure is one way to get additional samples. Whatever data analyses we have performed with the original N cases is then performed with the "new" bootstrapped samples (i.e., the one obtained by drawing N times with replacement), checking to see whether the results are like those obtained originally. The procedure is repeated several times, always sampling the original N subjects with replacement randomly.

In the second method for checking reliability, we change the criterion as to which responses were included. For example, with pigeon data, we vary the criteria used to select those trials whose data are averaged together; for human data, we vary the cut-off beyond which response latencies are discarded. Again, the analysis is repeated with matrices obtained for different response criteria. If the analysis is stable, similar results are obtained with various such criteria.

KYST. Multidimensional scaling is one method for extracting the underlying structure inherent in a ´confusion´ matrix. KYST (Kruskal, 1964) seeks a mapping of the stimulus set onto a configuration of points in a low-dimensional Euclidean space such that for any 4 stimuli A, B, C and D:

[A is closer to B than C is to D in Euclidean space]

if and only if

[the matrix entry AB is greater than the entry CD]

Put another way, one seeks a monotone rescaling of the data that converts the observed confusion matrix into the matrix of distances between suitable points in Euclidean space. This is likely to be too much to ask if the dimension of the Euclidean space is low (it puts too many constraints on the relative positions of the points, because the observed matrix reflects many influences including unsystematic error), so KYST is based on searching for a configuration of points that minimizes a measure of the misfit after monotone rescaling. This measure is called the ´stress.´ If the Euclidean space has large enough dimensionality, stress 0 can always be attained, but the result is not informative - the stimuli typically space themselves out over a sphere, adjusting their positions a bit to achieve the necessary inequalities. But if the configuration is forced down into a lower dimensional space, the results are often quite informative. A rule of thumb is to use at most n-space where

$$n < (\# \text{ of stimuli}) / 4.$$

In our case, almost perfect 3-dimensional configurations usually existed, and the 2-dimensional configurations were the most informative.[7]

ADDTREE. Each of the many clustering algorithms in the literature has its specific advantages and drawbacks (see Duda and Hart, 1973, Ch.6). However, we were attracted by the recent clustering algorithm introduced by Sattath and Tversky (1977), called ADDTREE, which seems to have worked especially well in analyzing other types of confusion matrices. To carry this out, we have written our own code, following as closely as possible the published description.

The idea is to examine each 4-tuple A,B,C,D of stimuli and ask whether both of the matrix entries AB and CD are greater than both of the entries AC or BD. This means that both pairs, AB and CD, are thought to be more similar than both pairs, AC and BD. We count up for each AB how many such pairs CD are found. This gives us a new matrix whose ABth entry is also large whenever A and B are considered similar. Finally, A and B are clustered if the entry in this new matrix for AB is larger than the entry for any other pair, AC, including A or any other pair, BC, including B. The stimulus set is collapsed by lumping the clusters together. A smaller confusion matrix is formed whose rows and columns correspond to these clusters:

for any pair of clusters, the new entry in the confusion matrix is computed by averaging the old matrix entries for all pairs AB, A in one cluster and B in another. Then the process is repeated iteratively until everything collapses to one cluster. The result is a tree with the stimuli as its leaves and the conjectured clusters are the subsets of the stimuli obtained by cutting one branch of the tree and considering all leaves on one side of the break. The strong point of the algorithm is that the complement of any cluster is also a cluster: all clusters are thought of as being formed in a context, relative to other stimuli present.

NEAREST NEIGHBORS. Finally, the most direct way to analyze a confusion matrix is to examine the largest entries of the matrix, the ´nearest neighbors.´ Because we only used 15 stimuli, it was practical to do this by inspection. For each matrix, we made a histogram of the entries. Usually, most entries are in the middle and small range, representing pairs of stimuli that have nothing very salient in common, and there is a small ´tail´ of large entries representing those pairs AB that are clearly very confusable or similar. We formed a graph by joining consecutively nearest neighbors starting with the nearest. We use double, single, and dotted lines as the matrix entry gets larger. Often, a subset appears as a clear cluster and then we make a check to see if it is a ´tight´ cluster: a tight cluster is a subset S of the stimuli such that for all A,B in S and C outside of S, the entry AB is always greater than AC.

For each of our experiments, we present the results from the three analyses in a single display: starting with the best 2-dimensional KYST plot, we superimpose the ADDTREE clusters that are most robust with respect to resampling the data by drawing circles around the clustered stimuli, then we draw links between the nearest neighbors as double, single or dotted lines. We have found that the result is a fairly compact and useful presentation of the data contained in the matrix.

B) PIGEON DISCRIMINATION

In analyzing the pigeon data, there were two specific problems. One was that the pigeons were far from being equal in their skills in this task: one (#4) learned each of the fifteen discrimination problems rapidly and well. Two others were moderately successful, and three came close to flunking the course. Inasmuch as the criterion for passing on to a new discrimination was a sliding one, even the final trials for each positive stimulus reflected widely varying levels of discrimination. Given the varying criteria for discrimination, the ´bootstrap´ is out of the question, so instead we split the data into #4 vs. the average of the five other birds in order to do a split-half reliability test.

The other problem is this: ideally, we sought a confusion matrix with comparable entries for every positive stimulus A and every distractor B. That is, we wanted a matrix in which the number in the ABth entry can be compared with the number in the CDth entry and the difference would reflect whether or not it was easier for the pigeon to reject B as a variant of A than to reject D as a variant of C. Unfortunately, each bird learned at a different rate, and the requirements of counterbalancing orders meant that each bird was given the positive stimuli in a different sequence. Hence, each subject may have had a different level of familiarity with the whole stimulus set when each particular polygon came up as the next positive. So, if bird b seeing positive polygon p took 8 sessions to meet the criterion, while bird b´ seeing the same polygon p took 3 sessions, how was one to average their error rates together? And worse, how could one compare this to the error rates when b and b´ saw a different polygon p´ as positive?

In order to get anywhere, we had to use a model of the learning process. One simple model of the error rate in acquiring a given discrimination (Mazur & Hastie, 1978) is that it follows hyperbolic curves:

$$\text{error rate} = \frac{\text{init}}{1 + \text{disc*time}}$$

where ´init´ is the initial error rate (e.g. 0.5 if there are two equivalent choices), and ´disc´ depends on how hard the discrimination is. According to this theory, the ratio of the error rates to two different distractors will approach $\text{disc}_1/\text{disc}_2$ as time gets large. It also seems reasonable to suppose that this ratio is relatively independent of the subjects´ native intelligence, which, for relatively simple tasks, would be expected to affect the rate of learning rather than the limit of discrimination. Therefore, we can hope that averaging these ratios over birds and over stages in the learning curve (after the initial period of learning) will give reasonable results. The only drawback in this approach is that the resulting matrix has entries such that the relative discriminability of distractors B and B´ in presence of positive A are comparable, but the discriminability of B as distractor to positive A vs. that of B´ as distractor to positive A´ is not computable.

Several confusion matrices have been derived from the data:

i) In the first one, the last trial for each positive stimulus for each bird was used (i.e., the last trial before a new positive stimulus was selected). For each such trial, the number of erroneous pecks to each negative stimulus was expressed as a percent of all erroneous pecks due to each negative stimulus. These percentages were then averaged over all such trials. This gives a 15x15 matrix, each of whose rows adds up to 1.0.

ii) In the next, the same procedure was followed except that all trials were used which met the following criterion[8] (also see Herrnstein, Loveland, & Cable, 1976): either rho > 0.9 or rho > 0.8 and the positive stimulus was the stimulus most responded to.[9] These so-called "good trials" included many which were not last trials and many last trials did not meet either criterion. A matrix X_{pgn} was derived by averaging the percent of error to each distractor over all good trials, which is reproduced in

printer:
greek
letter
'rho'
ρ

Table 1, and a histogram of the tabulated error rates is given in Figure 2.

Insert Table 1 and Figure 2 about here

----

iii) Another procedure was to use method (ii), but separately for the data from #4 vs the data from all other birds, on the possibility that #4 used quite different cues from the other birds.

iv) In order to obtain matrices in which entries across rows are comparable, one procedure is to assume that the full confusion matrix is approximately symmetric. Following the hyperbolic learning curve hypothesis, each row can be multiplied by a different scalar without changing the relative size of its entries (which is all the experiment determines). It can be proven[10] that there is a choice of such scalars, one for each row, such that if each row is multiplied by its scalar, then **the ith row sum equals the ith column sum.** Moreover, the resulting matrix is unique up to multiplication of each entry by the same scalar. We can perform this row normalization on all of the matrices described in i,ii, and iii.

One of the advantages of KYST is that it allows input confusion matrices in which distinct rows are noncomparable. This mode, called ´scale by rows,´ sets up independent rescaling for each row before fitting the matrix against the Euclidean distances of the

Table 1

$\underline{X}_{pgn}$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	*	14.9	8.6	10.1	9.3	4.7	4.7	4.6	7.2	4.8	6.7	4.7	10.3	4.9	4.2
B	22.8	*	9.9	10.7	9.4	2.8	2.1	3.1	8.3	5.6	5.5	3.2	11.2	2.4	3.0
C	10.6	11.3	*	14.1	10.9	3.2	2.5	3.5	11.4	8.7	3.9	3.4	10.3	3.3	2.9
D	9.7	12.4	11.3	*	15.2	4.0	3.1	3.6	8.5	6.9	4.0	3.9	9.1	4.5	3.7
E	10.3	13.1	10.8	12.4	*	4.5	4.3	5.9	7.9	5.3	4.8	4.0	8.0	4.7	3.9
F	4.3	3.4	2.5	2.0	2.3	*	7.9	13.8	4.1	5.4	12.7	13.3	5.2	9.2	13.8
G	2.9	2.8	2.2	2.7	3.2	11.2	*	12.3	3.2	5.9	11.3	11.1	3.4	16.2	11.5
H	3.3	3.7	2.7	3.0	2.7	12.9	8.9	*	5.3	7.5	11.5	11.9	3.1	10.4	13.0
I	8.8	9.9	8.3	9.3	8.6	4.9	2.9	4.4	*	10.6	7.6	5.8	10.7	3.6	4.5
J	7.5	6.9	6.3	5.8	5.8	6.4	5.2	5.9	14.8	*	6.0	6.2	11.1	4.1	7.8
K	6.1	5.1	3.4	3.7	3.6	12.2	5.4	12.2	4.9	4.0	*	14.3	4.9	8.5	11.7
L	3.9	3.7	3.0	2.4	4.2	14.1	5.8	15.2	3.5	3.4	13.7	*	3.2	9.5	14.4
M	12.9	11.3	7.6	8.4	6.5	6.2	3.0	6.0	8.1	6.6	7.5	5.9	*	5.2	4.8
N	4.1	4.9	4.2	4.3	9.2	7.3	25.2	7.3	4.2	4.1	6.7	7.6	3.6	*	7.2
O	4.2	4.9	5.5	4.2	4.2	9.8	4.0	11.4	8.3	10.4	11.0	10.7	5.2	6.2	*

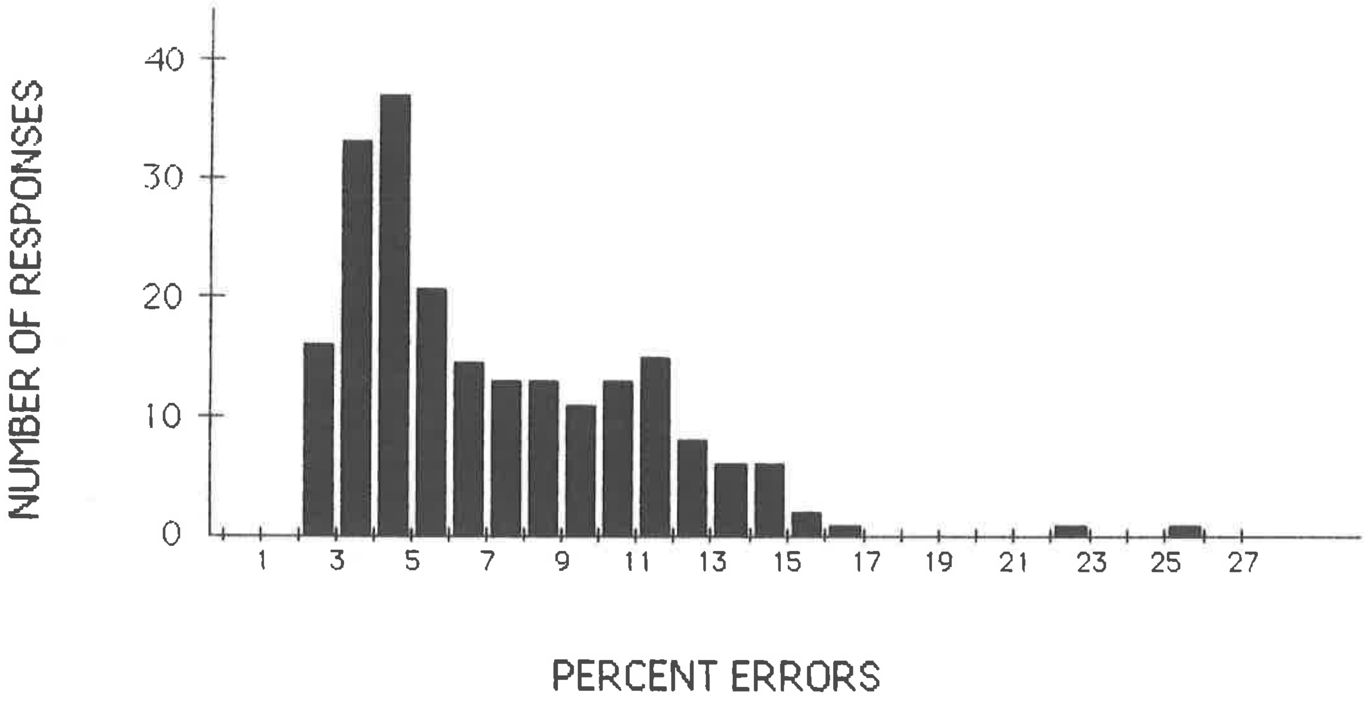

NUMBER OF RESPONSES
40
30
20
10
0
1
3
5
7
9
11
13
15
17
19
21
23
25
27
PERCENT ERRORS

Figure 2

configuration. We have used this option in finding scaling solutions for matrices of type i,ii and iii, as well as using the row normalization of iv and standard KYST. As would be expected, the stress is markedly lower with ´scale by rows,´ and given the uncertainties of the model, this seems the preferable way to do a scaling analysis of the pigeon data.

When the pigeon data were analyzed as in i, ii and iii, the correlations among the resulting matrices are presented in Table 2.

Insert Table 2 about here

KYST plots for each set of data were obtained by a) a preliminary search for a 2-dimensional configuration with independent monotone scaling on each row; and by b) taking this configuration as the starting point for a search with quadratic polynomial scaling by rows. The reason for this compound procedure (often used in scaling algorithms) is that monotone scaling by itself results in a nearly degenerate solution, but this is a good starting point for quadratic scaling (i.e., it gives the lowest stress found by all methods tried).

The rows were then normalized by multiplying by scalars so that the row sums and column sums were equal. Lack of symmetry in this normalized matrix indicates that a polygon A is distinguished from a polygon B more readily than B is distinguished from A. Asymmetry of this kind can be measured by taking the correlation of the normalized matrices with their transposes (the matrices formed by interchanging a matrix´s rows and columns). These correlations were 0.92 for last

Table 2

	last trials	all good trials	Subject 4 good trials	other birds good trials
last trials	-	0.95	0.86	0.90
all good trials		-	0.84	0.97
Subject 4 good trials			-	0.71
other birds good trials				-

trials, 0.87 for good trials, but only 0.79 for bird 4 alone and 0.82 for the other birds. This indicates either that there was more noise in the smaller data sets, or that asymmetric discriminations vary greatly from bird to bird and so cancel out in the merged data, or both. The resultant matrices were subjected to ADDTREE and were also evaluated in a nearest neighbor analysis.

The strongest pattern to emerge is that, no matter how analyzed, the birds seemed to cluster the polygons into four groups, as follows:

> (a) Five convex-skew quadrilaterals, F, H, K, L and O (see Figure 1), all with some acute angles, and without horizontal or vertical edges or right angles.
>
> (b) Two triangles, I and J.
>
> (c) Two concave figures, G and N, with ´crowns´ on top.
>
> (d) Six ´boxy´ figures, A, B, C, D, E, and M, either with horizontal and vertical edges and right angles, or with at least 5 sides and no acute angles.

This pattern emerged in **every** ADDTREE graph and in most of the nearest neighbor analyses. Thus cluster (a) is in fact a ´tight´ cluster in all data sets; (d) is nearly ´tight´, except for some close links to the polygon I in (b); and both pairs {I,J} and {G,N} are each others´ nearest neighbors in all data sets except that of Subject 4. Cluster (d) almost reproduces the intuitive (to humans) classification used to design the stimuli (see description in Methods section); three of the five figures in cluster (a) and both of the two figures in cluster (b) likewise fall into one of the design categories. Only cluster (c) violated the design classification, presumably because of the unanticipated influence of the crown on top, which only figures G and N shared.

´Good trials´ gave a KYST plot with the lowest stress, namely 0.127, and this plot seems to be the best summary overview of the pigeon errors; we have presented this solution in Figure 3. The four clusters above appear clearly in this KYST plot too. The most typical clusters appearing in most of the cluster analyses are illustrated in Figure 4.

Insert Figure 3 and 4 about here

Between clusters, one also finds quite consistent close links (i.e., large entries in the error matrices) as follows:

G,N with H,F and less so with L

I with M and less so with B,C

O with J

The most conspicuous exception to this picture is the data from Subject 4 alone. Although it is similar in broad outline, there is a suggestive pattern of differences. To begin with, Subject 4´s data do not seem to have a decent 2-dimensional KYST plot: all plots found have stress greater than 0.19 and each plot misses at least one major close pair (i.e. plots a pair with a large error entry too far apart).

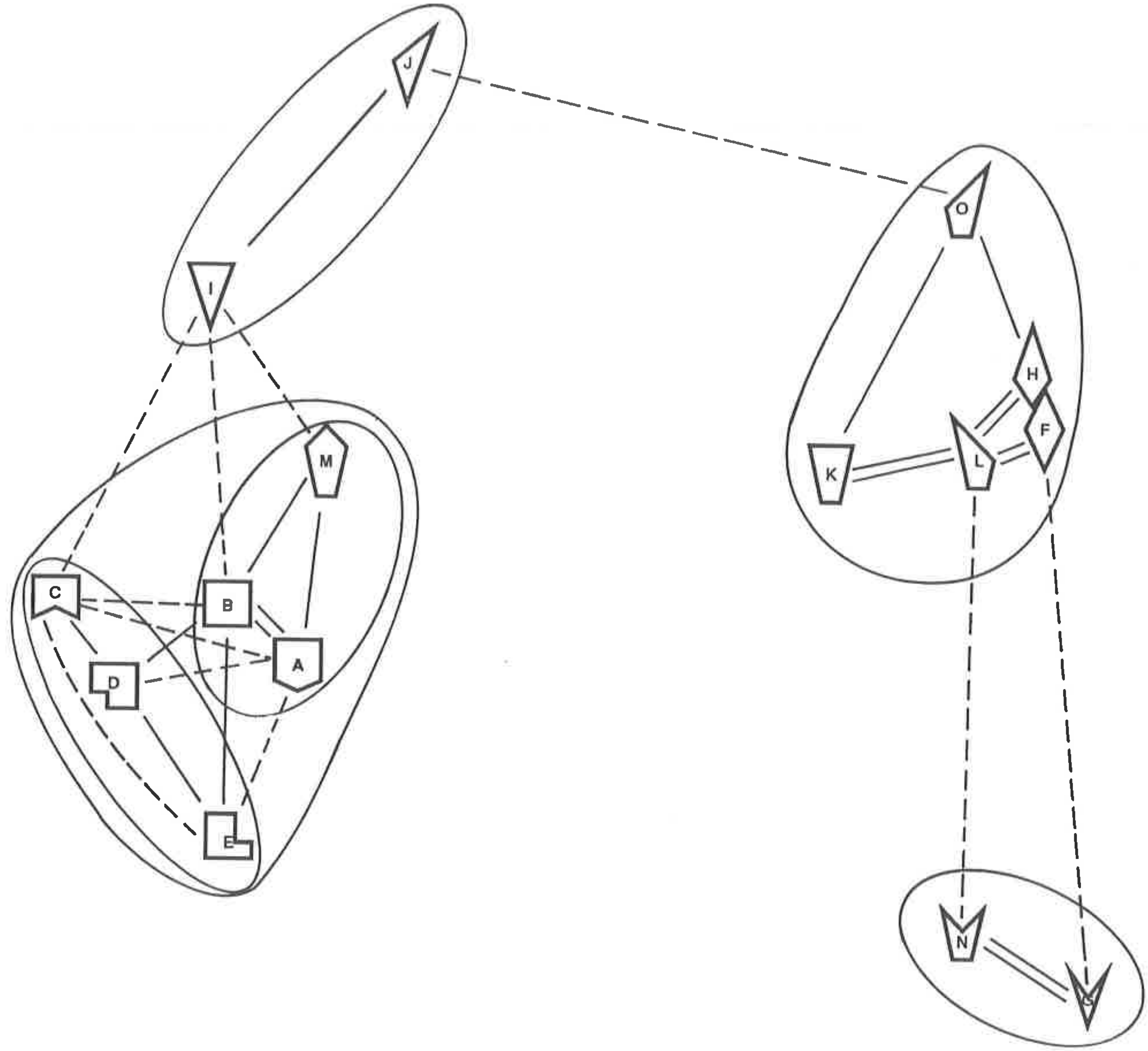


Figure 3

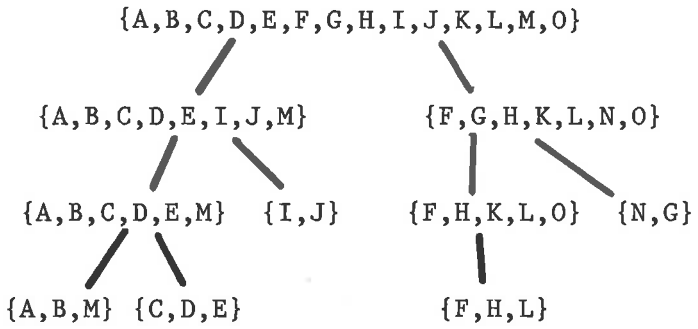

{A,B,C,D,E,F,G,H,I,J,K,L,M,O}
{A,B,C,D,E,I,J,M}
{F,G,H,K,L,N,O}
{A,B,C,D,E,M}
{I,J}
{F,H,K,L,O}
{N,G}
{A,B,M}
{C,D,E}
{F,H,L}

Figure 4

Moreover, the three clusters (a), (b), and (c) are not really separate, but rather are parts of a continuum like that in Figure 5.

Insert Figure 5 about here

Thus N is closer to H than G, and J is closer to O than I (compare Plot 1). Also {B,E} in cluster (d) is fairly close to {K,L} in cluster (a).

What does this mean? One interpretation is that Subject 4 was using a significantly more complicated set of features or dimensions with which to remember the different shapes. Instead of picking a few features that would do the trick most of the time, this bird had a more complete data structure to describe the shapes, one which does not admit a 2-dimensional representation and which drew on different characteristics of the shapes for different pair comparisons. The data for Pigeon 4, in its relatively fast and efficient learning cycle, reflect partial matches of one or another of the features being used for particular comparisons.

C) HUMAN DISCRIMINATION

The response times for each of the 28 subjects were arranged into a 15x15 matrix. Entry i,j in this matrix, in case i≠j, represented that subject´s time to judge whether the stimulus polygon j was different from the remembered polygon i or, in case i=j, represented the average time over 7 trials to judge whether the stimulus polygon i was the same as the remembered polygon i. To combine the matrices for

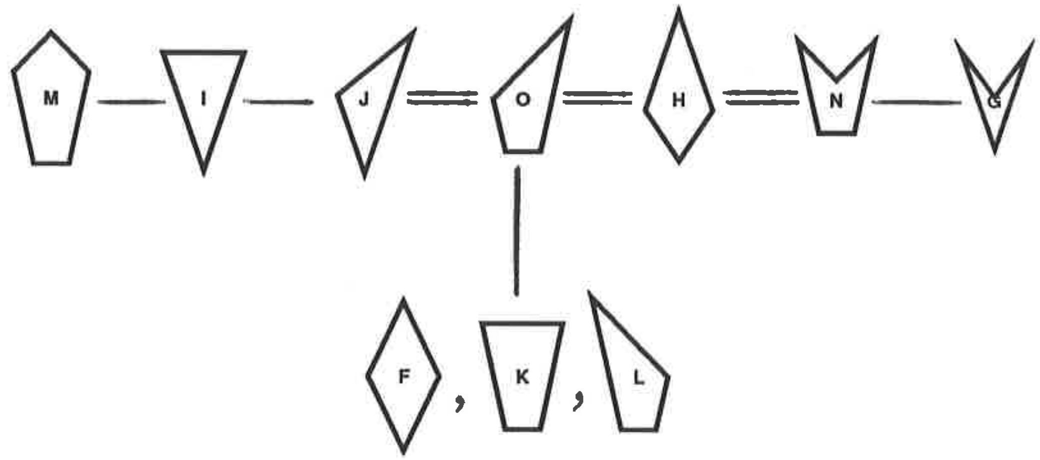


Figure 5

individual subjects into one matrix summarizing the data, two problems needed to be resolved: i) some people were on the average slower and some faster[11], and ii) occasional responses were far slower than the average (presumably due to the subject´s inattention). The first problem was handled by multiplying each subject´s matrix by the grand mean, for all subjects, over the subject´s own mean. To eliminate exceptionally slow outliers, response times falling in the upper 3% of the distribution were excluded from the analysis. This resulted in the elimination of all normalized times greater than 770 msec. For each individual pair i,j, with $i \neq j$, the remaining times that came from correct responses (i.e., polygon j was indeed different from polygon i) were then averaged, producing the i,j^th^ entry of the final matrix X_{hum}. The diagonal entries $(X_{hum})_{ii}$ are simply the average of all responses to stimulus polygon i with the same remembered polygon i, although these were not used in subsequent analysis. This matrix X_{hum} is reproduced in Table 3, and a histogram of the tabulated response times is shown in figure 6.

Insert Table 3 and figure 6 about here

To test the reliability of X_{hum}, we used the ´bootstrap´ procedure (see above). On average, if the 28 subjects are resampled, it turns out that the correlation of X_{hum} with the resampled matrix $X_{hum}´$ is 0.90. Matrix X_{hum} is not symmetric but, like all matrices, it can be decomposed into a sum of a symmetric and an anti-symmetric matrix:

Table 3

$\underline{X}_{hum}$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	445	436	450	413	414	402	412	418	417	410	488	406	401	465	390
B	455	439	429	481	418	408	454	431	414	432	431	415	419	406	466
C	539	454	489	425	492	419	449	423	401	426	416	482	434	464	462
D	451	477	489	481	584	413	404	387	405	450	402	410	415	415	418
E	426	469	469	529	478	417	449	429	457	459	407	422	420	422	457
F	410	445	424	408	441	445	423	554	462	424	435	469	444	464	429
G	383	370	413	410	379	390	440	437	438	429	403	414	415	448	379
H	426	386	433	383	414	595	410	487	467	424	434	464	477	445	430
I	427	393	474	413	432	468	453	550	451	482	453	456	481	415	445
J	393	404	428	428	417	487	439	458	456	500	433	462	436	401	539
K	439	415	392	445	400	444	471	434	451	414	463	431	503	427	444
L	432	452	437	433	442	491	432	517	475	526	460	512	435	437	539
M	439	440	493	404	411	491	419	467	467	430	468	416	462	453	417
N	418	412	452	443	413	431	529	441	424	438	465	452	464	498	410
O	427	439	429	463	453	438	451	489	455	519	452	519	458	436	511

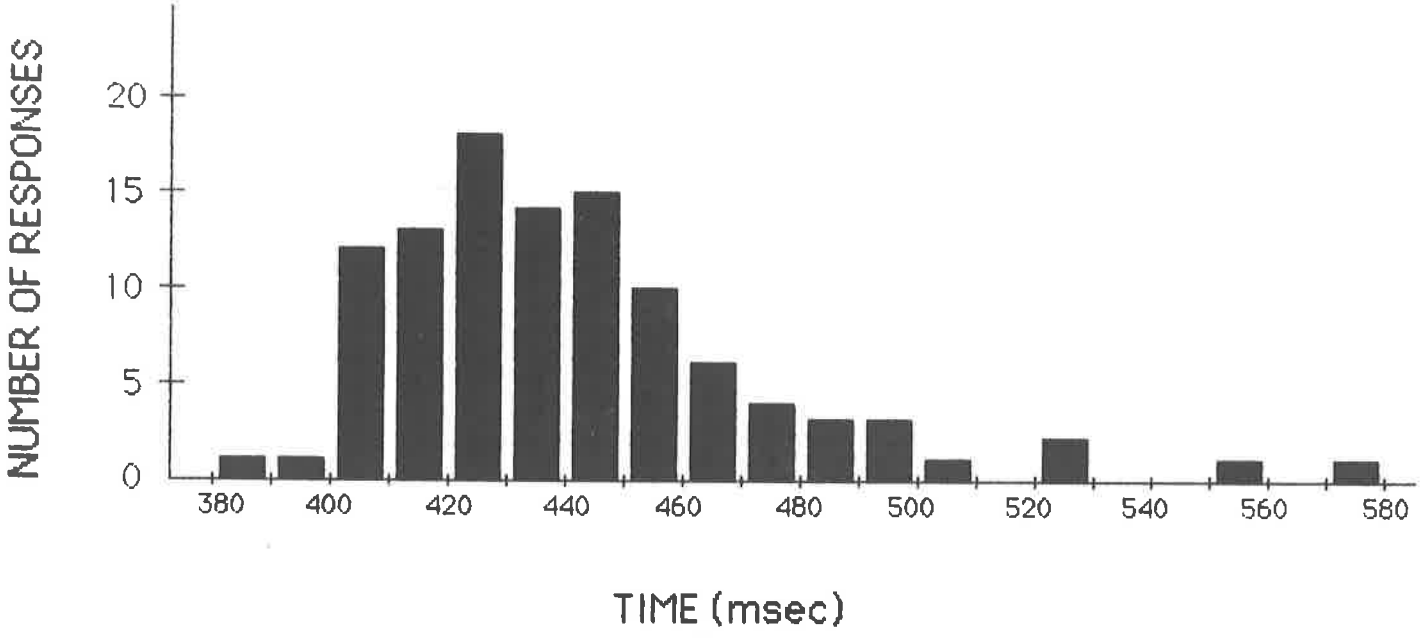

NUMBER OF RESPONSES
20
15
10
5
0
380
400
420
440
460
480
500
520
540
560
580
TIME (msec)

Figure 6

$$X_{hum} = (X_{hum})_{symm} + (X_{hum})_{anti},$$
$$(X_{hum})_{symm} = [X_{hum} + X^{t}_{hum}]/2,$$
$$(X_{hum})_{anti} = [X_{hum} - X^{t}_{hum}]/2,$$

(where X^{t}_{hum} is the matrix obtained from X_{hum} by interchanging rows and columns). The symmetric part of X_{hum}, i.e., $(X_{hum})_{symm}$, has an average correlation of 0.93 with the symmetric part of X_{hum}', and the anti-symmetric part of X_{hum}, i.e., $(X_{hum})_{anti}$, has an average correlation of 0.80 with the anti-symmetric part of X_{hum}'. We may therefore conclude that X_{hum}, and especially its symmetric component, are quite reliable, and that the anti-symmetric component also contains stable, though somewhat noisier, data.

The main analysis involved X_{hum} and ten variants of X_{hum} obtained by resampling X_{hum} by the bootstrap. We applied KYST, ADDTREE and the nearest neighbor analysis to the symmetric part, $(X_{hum})_{symm}$, of all eleven matrices. The ADDTREE clusterings were the most informative. The first result of note is that six of the variants gave clusterings completely identical to those for X_{hum} itself, and the others had only small variations. The clustering for X_{hum} is presented in Figure 7.

Insert Figure 7 about here

The only clusters that did not appear in 8 out of the 10 resamplings were the two highest level clusters {F,H,I,J,L,O} and {G,K,M,N}. Here, as for the pigeon data, the clusters echo, albeit imperfectly,

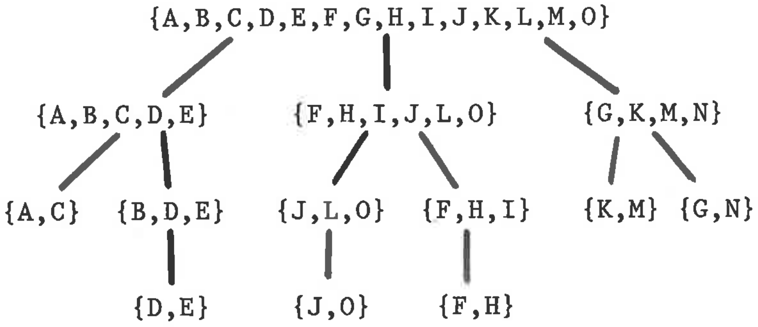

{A,B,C,D,E,F,G,H,I,J,K,L,M,O}
{A,B,C,D,E}
{F,H,I,J,L,O}
{G,K,M,N}
{A,C}
{B,D,E}
{J,L,O}
{F,H,I}
{K,M}
{G,N}
{D,E}
{J,O}
{F,H}

Figure 7

the intuitive classification used to design the stimuli: {A,B,C,D,E}, {F,G,H,I,J}, and {K,L,M,N,O}.

The derived clusters show an interesting combination of characteristics. Thus:

a) The clusters {J,O}, {F,H}, {G,N} and {K,M} are all cases that can be explained as ´template´ matches, that is, the two polygons can be nearly superimposed on each other and most of the vertices paired up so that the polygons nearly match and corresponding sides and vertices are close to each other.

b) The cluster {D,E}, however, cannot be explained by appeal to template matching because the notch in the two boxy figures is on opposite sides. Here it seems that the similarity is a case where the ´features´ one would naturally use to describe D and E are the same: both have only horizontal and vertical sides and right angles, both have six sides and both can be made by taking a rectangle and removing a notch. Only the "feature" of global orientation distinguishes the figures.

c) The clustering of L with {J,O} also does not reflect template matching. Here the cluster seems to include a mirror-image confusion of L and O, which seems totally natural to us, but which would not take place if one were using template matching.

d) The clustering of the triangle I with the diamonds {F,H}, especially with the bottom-heavy diamond H, may show that "sharp point" is a feature we use to encode shapes, regardless of orientation. All share a sharp point at the bottom, but two also have a point at the top.

e) Finally, A and C are clustered, perhaps as an example of opposites being confused. Both can be coded as squares with the bottom side dented, but in one case the dent is inward and in the other case, it is outward. Again, the "feature" appears to be somewhat abstract, not being bound to a particular direction or orientation.

Thus, it would appear that numerous features underlie the pattern of response times. Given this observation, it is not surprising that none of the two-dimensional KYST plots were really good representations of the matrix X_{hum} or its resampled variants, with stress values falling between 0.179 and 0.212. However, the analyses all showed similar patterns, with at most 2 or 3 polygons being shifted more than one-tenth of the width of the whole plot. We have reproduced in figure 8 the KYST plot and the ADDTREE clusters for the matrix derived from the full set of 28 subjects, omitting the unstable highest level clusters mentioned above. The stress for this plot is 0.186, indicating that it captures the relations in the matrix moderately well but not completely. Using several random initial conditions failed to produce a plot with lower stress.

Insert figure 8 about here

The double, single and dotted lines in Plot 2 represent the largest entries in the matrix X_{hum}. These links confirm closely the ADDTREE clusters and the two-dimensional geometry of the KYST plot. The double lines represent mean response times longer than 500 msec, the single lines response times from 475 to 500 msec and the dotted lines response times from 465 to 475 msec, at which point we begin to hit the main group in the histogram of response times (see Figure 6). The response times indicated by the links were reliably slow in almost all resamplings. The next slower group of response times, in the range 459 msec - 465 msec, is not marked in Plot 2. In this group, one finds a set of links between polygons which occur in almost all resamplings but which are not so intuitively natural:

C -- M
A -- K
C -- L

The anti-symmetric part of the matrix X_{hum} of response times is more difficult to understand. These comparisons reflect the degree to which it is easier to reject a polygon i as an instance of the remembered polygon j than to reject j as an instance of i. The simplest way to try to predict the non-symmetry in the matrix is to seek a linear ordering of the polygons such that i is to the left of j if and only entry $(X_{hum})_{ij}$ is greater than $(X_{hum})_{ji}$. Such a

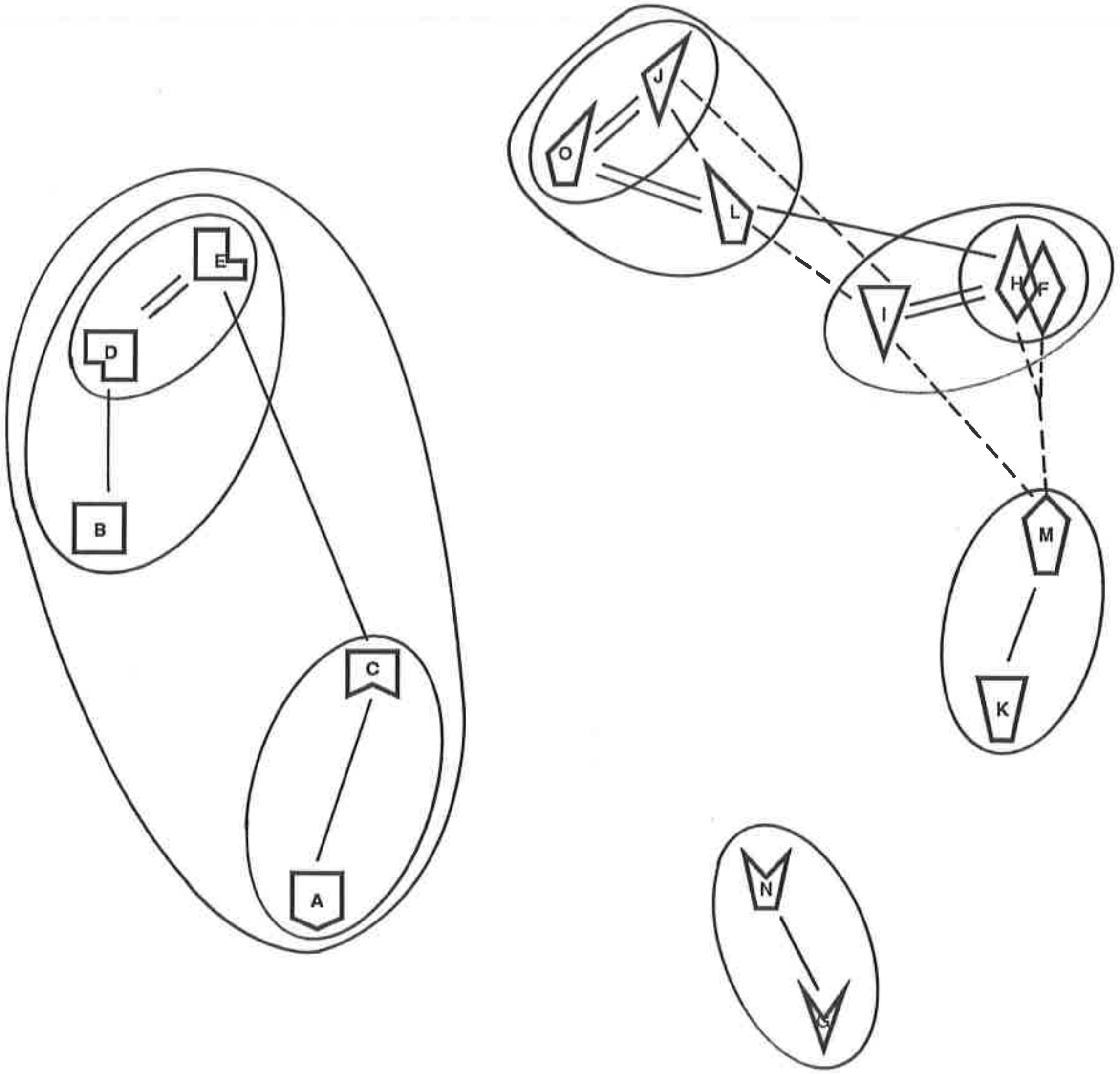


Figure 8

prediction follows, for instance, from Tversky´s theory of human similarity judgments (Tversky, 1977). The simplest way to derive such an ordering is to associate to polygon i the single number

$$\sum_j [(X_{hum})_{ij} - (X_{hum})_{ji}]$$

and to order the polygons using the magnitude of this sum. This number can be thought of as an overall distinguishability factor for each polygon. Doing this results in the ordering:

G , H , A , J , F , K , M , D , B , C , N , E , I , O , L.

This ordering was unstable when the bootstrap technique was used. However, the partial ordering of the polygons:

G , H , A , {B,C,D,F,J,K,M,N} , {E,I} , {L,O}

was much more reliable and makes some sense intuitively. Thus, G is the V-shaped crown which seems immediately distinguishable from all the other stimulus polygons, and O and L are the mirror-image skew quadrilaterals which are so alike that they are hard to keep in mind. But why is the bottom-heavy diamond H easier to distinguish from the other polygons than the triangle I?[12]

SECTION IV: COMPUTER SIMULATIONS

As described in the Introduction, an integral part of this investigation is a comparison of human and pigeon data with each other and with similarity measures derived from various algorithms in the computer vision literature. The classification of simple black and white silhouette shapes has been studied for various applications, including character recognition, automated blood cell and chromosome classification, circuit board inspection, etc. The most extensive work, however, has been done in character recognition, and here the most active area in recent years has been ´Kanji´, or Chinese ideogram recognition. Given the several thousand common Kanji, as opposed to 26 roman letters, the problem of Kanji recognition requires considerably more refined techniques than those for roman letters. Research on the recognition of Kanji has been actively pursued at several Japanese universities and industrial research laboratories. An excellent survey of the work can be found in Mori, Yamamoto and Yasuda (1984). Although there are many alternative algorithms in this and other papers, we have taken two algorithms which are characteristic of two of the major approaches to shape classification discussed in the Introduction: template matching with two-dimensional data structures and structural matching of combinatorial data structures. These two algorithms seemed to exemplify the basic principles of these two approaches, to be as powerful as any that we have seen and to fit easily the type of data in our experiment.

Although they are not the only potentially interesting algorithms, they provide good initial points of departure for evaluating the data.

A) TEMPLATE MATCHING

We follow the ideas of Maeda, Kurosawa, Asada, Sakai & Watanabe, (1982) in a template scheme that has, in fact, been implemented commercially by Toshiba. To make our terminology clear, suppose the two shapes to be compared are both represented by patterns of dots in a grid (e.g., as letters are represented on a computer bit-mapped screen). In other words, each shape is approximated by a set of points in a large square grid in which each individual grid point has coordinates given by a pair of integers i,j, and i and j are allowed to be any integer between 1 and some upper bound like 256. We then describe the two shapes S and T to be compared as **images** I(i,j) and J(i,j). I(i,j) is defined to be 1 at points (i,j) in the first shape S and to be 0 at points (i,j) outside S. J(i,j) is defined similarly using the second shape T. In straightforward template matching, one measures the "distance" from an image I to an image J either by what we may call the Euclidean distance between the multi-dimensional data I and the multi-dimensional data J:

$$\sum_{ij} [(I(i,j) - J(i,j))^2]$$

or by the correlation:

$$\sum_{ij} [I(i,j)*J(i,j)] \Big/ \sqrt{\sum_{ij} [I(i,j)^2]*\sum_{ij} [J(i,j)^2]}.$$ [13]

Maeda et al (1982) introduced two further ideas to make these template matching procedures more powerful. Both of these enhancements are motivated, in spirit if not in detail, by what is known physiologically about the initial levels of visual processing in many animals. The first is not merely to compare the raw images, but to "smooth" the images first. Smoothing is achieved by convolving I(i,j) and J(i,j) with a 2-dimensional Gaussian kernel, which has the same effect as viewing the shapes through a blurry lens. After blurring, the images are correlated (or the Euclidean distance is computed).

If images are blurred too much, all shapes look like amorphous blobs and the comparison is uninformative; if they are not blurred at all, the comparison may be unduly influenced by irrelevant noise and details (e.g. serifs on roman letters). Thus, the amount of blurring must be tuned to the application at hand, just as an animal´s attention may be focused on features of a particular size. If the right amount of blurring is chosen, smoothing can be an effective tool in increasing the power of template matching in making discriminations.

The second improvement consists of comparing not only the original images and their smoothings, but of comparing derived images computed from the partial derivatives of the images I and J. What do these derived images represent? The derivative in the i-direction, given numerically by:

$$I(i+1,j) - I(i-1,j)$$

will be zero away from edges, and near edges it will measure how vertical the edge is, having peaks on each side of the edge with opposite sign. The corresponding derivative in the j-direction is a similar horizontal edge detector. The neurophysiological analogue is the response of the so-called simple cells with one excitatory and one inhibitory band placed left-right or top-bottom within the receptive field. There is evidence that birds as will as mammals possess such specialized cells (Revzin, 1969).

Finally one may use higher derivatives. The most useful of these seems to be the Laplacian of the image, which is the sum of its second derivatives in the i- and j- directions. It is given numerically by:

$$I(i+1,j) + I(i,j+1) + I(i-1,j) + I(i,j-1) - 4*I(i,j).$$

What the Laplacian detects depends on how much the image has been blurred prior to computation. If the blurring is small, the Laplacian has peak response at the **corners**, being positive just inside convex corners and negative just outside concave corners. If the blurring is large, the Laplacian has peak positive response in the middle of the figure itself or the convex blobs out of which the figure is built, and has peak negative response in any "bays" (areas outside, but nearly surrounded by, the shape). The Laplacian of a blurred derived image is a computer simulation of the "center-surround" response of many retinal ganglion cells.

The basic idea of Maeda et al (1982) is to combine correlations or Euclidean distances of these various derived images with suitable weights to give a number that expresses how well not only the interiors of the shapes, but their edges, corners, centers and bays all match up when they are superimposed. The weights are selected **ad hoc** by what seems to make the categorizer work best: in character recognition, this meant optimizing performance on a training set, but in our case, we use the weights to make the model match observed behavior as well as possible.

Details of the algorithm can be found in Appendix I. The best fit for the pigeon data was a mixture with 70% coming from correlations of slightly blurred versions of the polygons and 30% from correlations of heavily blurred derivative images. This computer-generated correlation matrix has a correlation of 0.69 with the pigeon error rates after monotone rescaling, hence accounts for 47% of the variance. It has a correlation of only 0.56 with the human response time data after monotone rescaling, hence accounts for 32% of the variance. We denote this matrix by $X_{tem,pgn}$; it is reproduced in Table 4.

Insert Table 4 about here

The best fit for the human data used Euclidean distances and was a mixture of 70% distances between **heavily** blurred polygons and of 30% distances between **heavily** blurred derivatives images. This mixture has a correlation of 0.65 with the symmetrized human data after monotone rescaling, hence accounts for 42% of the variance of the data. It has a correlation of 0.65 with the symmetrized pigeon data after monotone rescaling too. The matrix, which we denote $X_{tem,hum}$, is presented in Table 5.

Insert Table 5 about here

Table 4

$\underline{X}_{tem,pqn}$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	*	.986	.896	.925	.867	.790	.623	.747	.820	.681	.857	.744	.811	.771	.734
B	.986	*	.931	.908	.885	.745	.612	.706	.778	.650	.804	.703	.756	.757	.699
C	.896	.931	*	.861	.822	.677	.596	.629	.690	.597	.665	.617	.634	.717	.618
D	.925	.908	.861	*	.810	.792	.677	.740	.827	.676	.827	.772	.789	.809	.715
E	.867	.885	.822	.810	*	.802	.646	.806	.763	.696	.786	.776	.771	.751	.718
F	.790	.745	.677	.792	.802	*	.717	.977	.885	.811	.898	.840	.952	.801	.840
G	.623	.612	.596	.677	.646	.717	*	.708	.836	.792	.760	.749	.713	.945	.749
H	.747	.706	.629	.740	.806	.977	.708	*	.836	.803	.856	.839	.915	.784	.839
I	.820	.778	.690	.827	.763	.885	.836	.836	*	.847	.966	.835	.925	.903	.835
J	.681	.650	.597	.676	.696	.811	.792	.803	.847	*	.826	.650	.797	.823	.965
K	.857	.804	.665	.827	.786	.898	.760	.856	.966	.826	*	.851	.965	.856	.851
L	.744	.703	.617	.772	.776	.840	.749	.839	.835	.650	.851	*	.830	.819	.679
M	.811	.756	.634	.789	.771	.952	.713	.915	.925	.797	.965	.830	*	.801	.828
N	.771	.757	.717	.809	.751	.801	.945	.784	.903	.823	.856	.819	.801	*	.818
O	.734	.699	.618	.715	.718	.840	.749	.839	.835	.965	.851	.679	.828	.818	*

Table 5

$\underline{X}_{tem,hum}$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	*	12	133	81	162	235	969	317	267	528	172	363	213	463	369
B	12	*	94	87	141	260	934	338	281	525	207	380	254	449	387
C	133	94	*	110	135	261	621	309	249	380	291	328	350	281	332
D	81	87	110	*	131	186	720	248	188	406	168	240	216	302	310
E	162	141	135	131	*	139	574	152	175	309	170	169	200	243	254
F	235	260	261	186	139	*	504	21	74	189	68	124	49	193	124
G	969	934	621	720	574	504	*	420	364	189	608	361	683	129	361
H	317	338	309	248	152	21	420	*	103	158	117	108	97	164	108
I	267	281	249	188	175	74	364	103	*	127	50	121	94	103	121
J	528	525	380	406	309	189	189	158	127	*	242	252	291	85	48
K	171	207	291	168	170	68	608	117	50	242	*	159	25	239	159
L	363	380	328	240	169	124	361	108	121	252	159	*	192	139	244
M	213	254	350	216	200	49	683	97	94	291	25	192	*	308	193
N	463	449	281	302	243	193	129	164	103	85	239	139	308	*	139
O	369	387	332	310	254	124	361	108	121	48	159	244	193	139	*

As in the analysis of the experimental matrices, we then obtained a KYST plot of the matrices $X_{tem,pgn}$ and $X_{tem,hum}$: these had stresses of 0.116 and 0.084 respectively, hence are good representations of the structure of $X_{tem,pgn}$ and $X_{tem,hum}$. Moreover, ADDTREE cluster analyses of $X_{tem,pgn}$ and $X_{tem,hum}$ resulted in quite similar clusters, illustrated in Figures 9 and 10.

Insert Figures 9 and 10 about here

The KYST plots, as well as these clusters and the closest neighbors, are depicted for the two matrices $X_{tem,pgn}$ and $X_{tem,hum}$ in figures 11 and 12 in the same display used for the pigeon and human experiments.

Insert figures 11 and 12 about here

What are the characteristics of these matrices? First, the clusters {A,B,C,D,E}, representing boxy shapes (i.e., shapes with horizontals, verticals and right angles), {G,I,J,O,N}, representing more or less triangular shapes and {F,H,K,L,M}, representing more or less the convex 4 and 5-sided figures, were present in each ADDTREE analysis. The main exception to this pattern is with the mirror-image shapes L and O: O is a close template match to the triangle J, hence is clustered with it, whereas L is not close to any triangle, hence is clustered with the ´nondescript´ quadrilaterals.

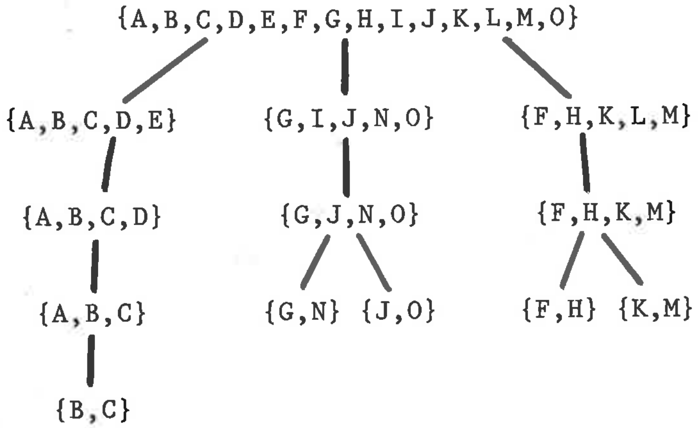

{A,B,C,D,E,F,G,H,I,J,K,L,M,O}
{A,B,C,D,E}
{G,I,J,N,O}
{F,H,K,L,M}
{A,B,C,D}
{G,J,N,O}
{F,H,K,M}
{A,B,C}
{G,N}
{J,O}
{F,H}
{K,M}
{B,C}

Figure 9

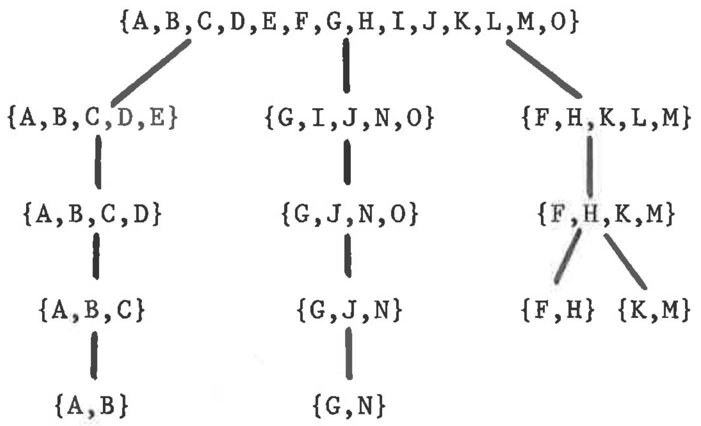

{A,B,C,D,E,F,G,H,I,J,K,L,M,O}
{A,B,C,D,E}
{G,I,J,N,O}
{F,H,K,L,M}
{A,B,C,D}
{G,J,N,O}
{F,H,K,M}
{A,B,C}
{G,J,N}
{F,H}
{K,M}
{A,B}
{G,N}

Figure 10

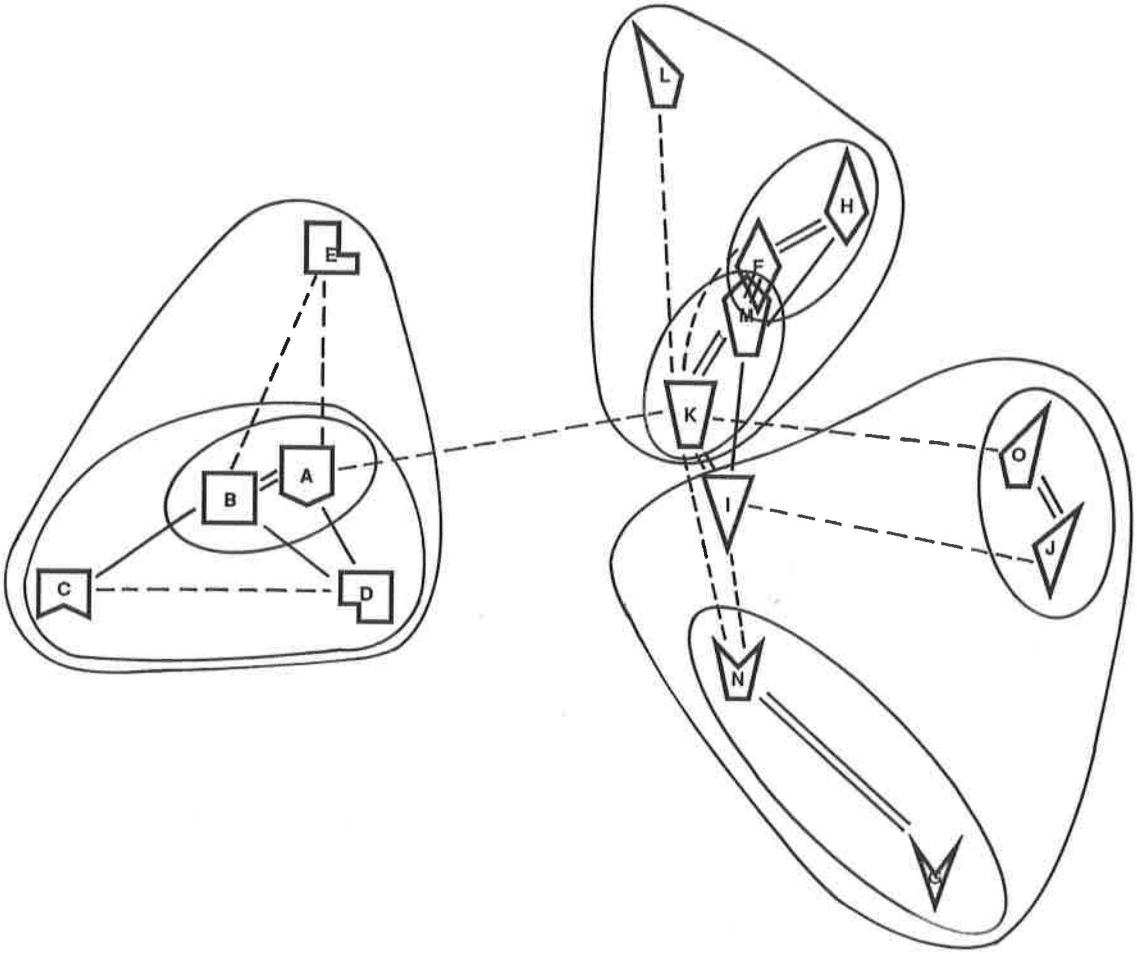


Figure 11

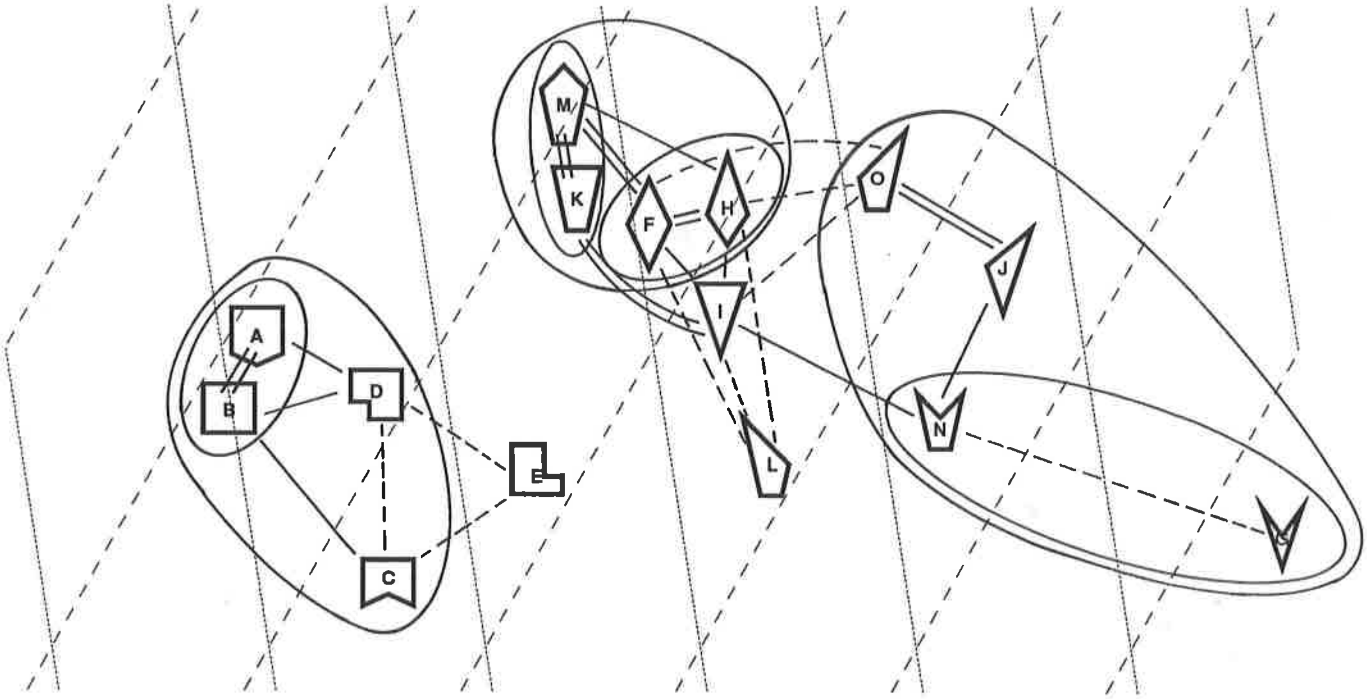


Figure 12

A striking difference between the two matrices, $X_{tem,pgn}$ and $X_{tem,hum}$, emerges, however, when looking at nearest neighbors. In the case of $X_{tem,pgn}$, nearest neighbor analysis defines a chain of polygons, each close in obvious template fashion:

bottom-heavy diamond H <---> symmetrical diamond F
<---> top-heavy coffin M
<---> broad- and flat-topped trapezoid K
<---> flat-topped triangle I.

Note that this seems natural with template matching, but is different from the clustering exhibited by the pigeon error rates: the pigeons clustered quadrilaterals H,F and K with the other quadrilaterals O and L, not with 3- and 5-sided figures like M and I. Similarly, from a template point of view the lopsided quadrilateral O, which stretches up to the right, is close to the lopsided triangle J, which stretches up to the right. But the pigeons clustered the two triangles J and I together and clustered the quadrilateral O with the other skew quadrilaterals. Thus, one aspect of the pigeon error rates that is not captured by template matching is the pigeons´ apparent use of the number of sides of polygons in clustering.

The above chain of nearest neighbors does not appear in $X_{tem,hum}$. In fact, close examination of the matrix reveals counter intuitive distance measures, such as the greater proximity of the crown shaped polygon N to the irregular triangle J than to the crown G with pointed bottom; and the greater distance of quadrilateral L from the trapezoid K with the same bottom half than from the triangle I. The most likely explanation may rest on the fact that the human data was matched best by using template matching at the highest blurring level. At this blurring, all the polygons are smooth blobs, with only slight irregularities to indicate their original sharp differences. What remains is their over-all brightness, which is determined by one thing: the area of the original. In fact, the areas in square pixels of the polygons is:

A:	690
B:	676
D:	595
M:	587
K:	562
C:	540
E:	532
F:	507
H:	465
I:	456
L,O:	441
J:	345
N:	324
G:	166

Tilting the KYST plot of $X_{tem,hum}$, one finds that one axis of this KYST plot is almost exactly given by the area of the polygon. In fact, if a ´boxiness´ feature is introduced by the following scale, then the KYST plot is almost perfectly represented by the two features, area and boxiness:

MOST BOXY

convex, right angles, horizontals+verticals

concave, right angles, horizontals+verticals

convex, horizontal edges, no angles < 45°

convex, no angles < 45°

convex, one angle < 45°

concave, horizontal edges

convex, two angles < 45°

concave, three angles < 45°

LEAST BOXY

In figure 12 the corresponding axes are drawn in. The axes are not perpendicular because these two attributes are partially correlated, due to the fact that the polygons were designed with equal perimeters. The template matching scheme yielding $X_{tem,hum}$ seems to capture that part of the human performance which, under time pressure, seizes on the most salient global large scale features of the polygons. Template matching on a finer scale does not seem to capture any more of the variance in the human response time data, and it totally fails to model other aspects of human performance, such as mirror-image confusion.

B) STRUCTURAL ANALYSIS

The idea behind structural analysis is to compare two figures, P and Q, by attempting to match the parts of P with the parts of Q, then quantifying the degree of match or nonmatch. The parts may be parts of its interior (e.g. the head and trunk are parts of the body) or parts of the boundary (e.g. a triangle has three edges and three vertices). For each pair of parts x of P and y of Q, one needs a measure d(x,y) of how different x and y are as isolated geometric objects.

In the simplest case, one looks at all matches of the parts of P with the parts of Q that preserve adjacency relations between parts. That is to say, if parts x_1 and x_2 of P are adjacent, then the corresponding parts y_1 and y_2 of Q should be adjacent, and vice versa. One assigns to each such match f a measure of how much it changes the parts of P to fit them to the parts of Q by adding up the differences d(x,f(x)) of corresponding parts. Call this the cost of the match. The difference between P and Q is then the least cost of all matches between P and Q. The basic ideas behind this way of measuring the similarity of two shapes can be found in Fischler & Elschlager (1973), Rosenfeld, Hummel, & Zucker (1976), and Ullman (1979). Although there is no neurophysiological evidence to date that animals use structural matching algorithms, there is considerable psychological evidence that humans use such procedures in some cases (Reed, 1974 and 1975; Palmer, 1977).

However, this procedure is complicated by the fact that figures do not usually come with an unambiguous set of parts. For instance, the classic demonstration of Attneave (1959) suggested that a cat silhouette is perceptually well approximated by a polygon with appropriate vertices, hence one is led to consider the **edges** of this polygon as the **parts** of the silhouette. But two people might look at the same silhouette of a cat and describe it as a polygon in different ways, one putting in more and the other putting in fewer edges. If we are to treat polygons as idealizations of general silhouettes, we must therefore allow matches between polygons with unequal number of edges. In any structural matching algorithm, one must allow for a refinement or restructuring of the parts of each figure before requiring a perfect part-for-part match between them.

In the case of polygons, we have introduced two ways of refining the parts of a polygon:

a) an edge of a polygon may be split by adding a corner somewhere in the middle. Thus, the edge is replaced by two edges and an extra corner is added.

b) a corner of a polygon may be refined by being truncated. Thus, a new, very short, edge appears where the corner was, and two new corners are added connecting the new edge to the old abutting edges.

If a polygon arises as an approximation to the shape of a figure with curved boundary, then these two refinement operations describe the simplest ways to refine the approximation with a polygon with one more corner. For a different approach to handling refinements, see A. Rosenfeld and K. Yamamoto (1982).

Details of the structural matching algorithm are presented in Appendix II. In brief, the algorithm that best matched pigeon error data defined the edges of a polygon to be its parts (allowing these to be refined as above in a match with another polygon). Differences of length and orientation of an edge are used to measure the difference between edges. An extra cost term was introduced if a concave part of one polygon was matched to a convex part of the other polygon. Finally, mirror reversing matches were excluded. The matrix of distances obtained from this algorithm will be called $X_{str,pgn}$, which is presented in Table 6.

Insert Table 6 about here

This matrix has a correlation of 0.67 with the symmetrized pigeon error rate, hence it accounts for 45% of the variance. It has a

Table 6

$\underline{X}_{str,pgn}$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	*	293	1205	824	1123	1584	1964	1338	728	1258	690	1251	1249	1809	1251
B	293	*	918	549	863	1959	2382	1884	1231	1625	534	1199	1063	1623	1199
C	1205	918	*	945	1748	2470	2740	2586	1654	2099	1476	2123	1992	2552	2123
D	824	549	945	*	1411	2316	2710	2292	1617	1921	975	1667	1469	2029	1479
E	1123	863	1748	1411	*	2610	2547	2509	1984	2420	1500	1851	1950	2093	2255
F	1584	1959	2470	2316	2610	*	1688	122	1222	561	1499	840	511	2053	840
G	1964	2382	2740	2710	2547	1688	*	1925	1317	1362	1573	1714	1838	335	1714
H	1338	1884	2586	2292	2509	122	1925	*	1616	732	1853	894	882	2238	894
I	728	1231	1654	1617	1984	1222	1317	1616	*	576	339	1071	996	1556	1071
J	1258	1625	2099	1921	2420	561	1362	732	576	*	854	1729	818	1462	360
K	690	534	1476	975	1500	1499	1573	1853	339	854	*	787	617	1177	787
L	1251	1199	2123	1667	1851	840	1714	894	1071	1729	787	*	637	1502	1396
M	1249	1063	1992	1469	1950	511	1838	882	996	818	617	637	*	1540	637
N	1809	1623	2552	2029	2093	2053	335	2238	1556	1462	1177	1502	1540	*	1502
O	1251	1199	2123	1479	2255	840	1714	894	1071	360	787	1396	637	1502	*

correlation of 0.57 with the symmetrized human response time matrix, hence it accounts for 33% of the variance.

In order to compare not only the fit with the pigeon data but to gain some insight into which aspects of the pigeon data this algorithm was, and was not, capturing, a KYST plot of $X_{str,pgn}$ was made. It had stress 0.124 and is reproduced in figure 13.

Insert figure 13 about here

Next, ADDTREE was applied to $X_{str,pgn}$ resulting in the clusters given in Figure 14.

Insert Figure 14 about here

The second structural matching algorithm was optimized to account for the human data. Details are again in Appendix II. In brief, this algorithm defined the parts of a polygon to be both its edges and corners. A match between two polygons must match the edges of the first polygon, suitably refined, to the edges of the second polygon, also refined, and it must match the corners to the corners. It measured differences using only lengths of edges and angles at corners, thus disregarding orientation. Mirror reversing matches were also allowed. This means that two polygons that differ by a rotation or reflection come out as having a perfect match. But the final cost was computed by adding the cost of the best match so obtained and a penalty for the net rotation or reflection used in this match. This

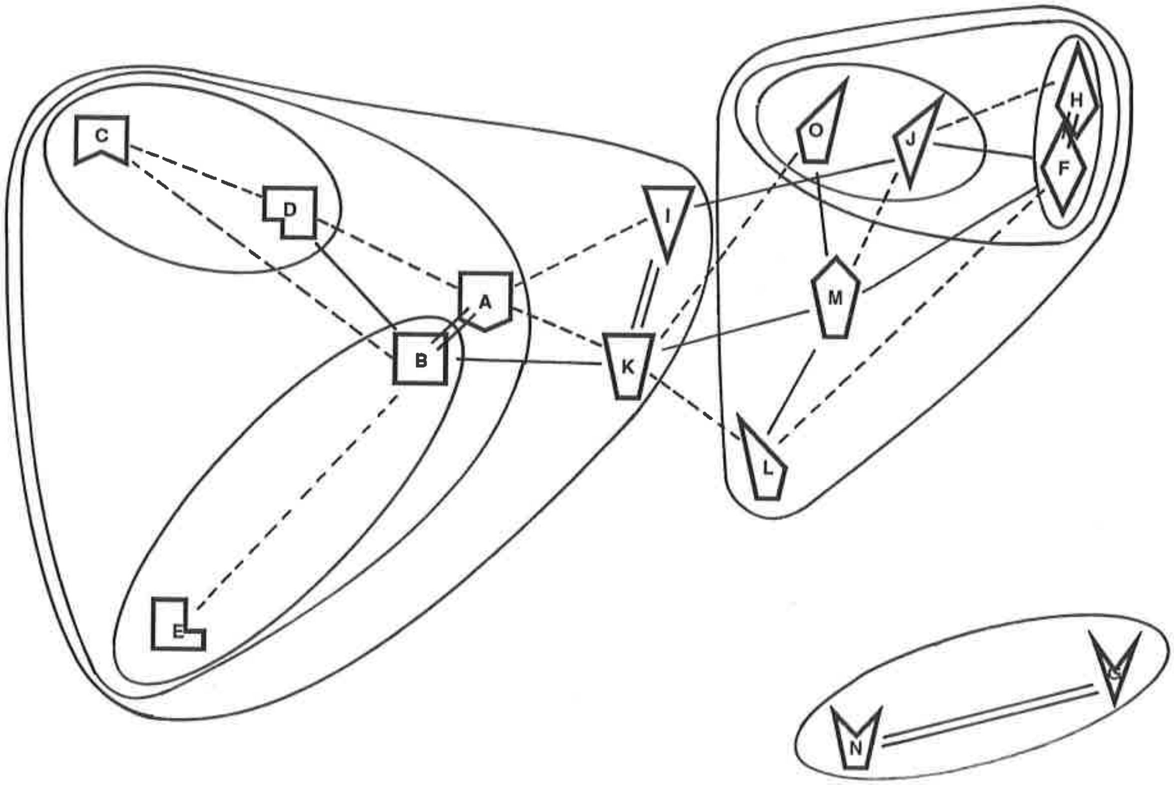


Figure 13

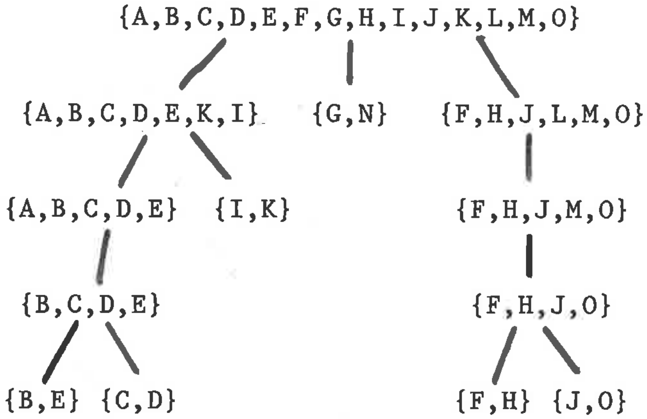

{A,B,C,D,E,F,G,H,I,J,K,L,M,O}
{A,B,C,D,E,K,I}
{G,N}
{F,H,J,L,M,O}
{A,B,C,D,E}
{I,K}
{F,H,J,M,O}
{B,C,D,E}
{F,H,J,O}
{B,E}
{C,D}
{F,H}
{J,O}

Figure 14

resulted in a matrix $X_{str,hum}$ of numbers representing the differences d(P,Q) of all pairs of our 15 polygons, presented in Table 7.

Insert Table 7 about here

The correlation of the human response time matrix and $X_{str,hum}$, after a monotone rescaling, was 0.80; or, equivalently, $X_{str,hum}$ rescaled accounts for 64% of the variance in the human response times.

As above, in order to compare not only the fit with the human data, but to gain some insight into particular aspects of the human data that are, and are not, captured by this algorithm, a KYST plot of $X_{str,hum}$ was made. It had stress 0.124 and is reproduced in figure 15.

Insert figure 15 about here

Next ADDTREE was applied to $X_{str,hum}$ resulting in the clusters presented in Figure 16.

Insert Figure 16 about here

A striking result here is that for the first time the set of non-convex polygons {C,D,E,G,N} shows up as a cluster for the matrix $X_{str,hum}$. As one might expect, the global property of being either convex or non-convex causes these two types of polygons to group together following structural matches, especially if these matches can

Table 7

$\underline{X}_{str,hum}$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	*	198	278	369	402	372	793	393	409	384	236	377	354	680	377
B	198	*	257	496	461	449	732	469	360	414	189	393	394	588	393
C	278	257	*	307	313	555	559	551	548	534	432	593	443	378	593
D	369	496	307	*	92	696	448	632	817	634	492	487	479	373	493
E	402	461	313	92	*	693	344	717	733	633	508	589	453	329	583
F	372	449	555	696	693	*	883	122	539	403	457	328	270	736	328
G	793	732	559	448	344	883	*	792	522	684	600	613	693	150	613
H	393	469	551	632	717	122	792	*	493	328	413	185	254	663	185
I	409	360	548	817	733	539	522	493	*	314	138	318	415	524	318
J	384	414	534	634	633	403	684	328	314	*	360	138	391	581	133
K	236	189	432	492	508	457	600	413	138	360	*	371	317	444	371
L	377	393	593	487	589	328	613	185	318	138	371	*	274	585	6
M	354	394	443	479	453	270	693	254	415	391	317	274	*	526	274
N	680	588	378	373	329	736	150	663	524	581	444	585	526	*	585
O	377	393	593	493	583	328	613	185	318	133	371	6	274	585	*

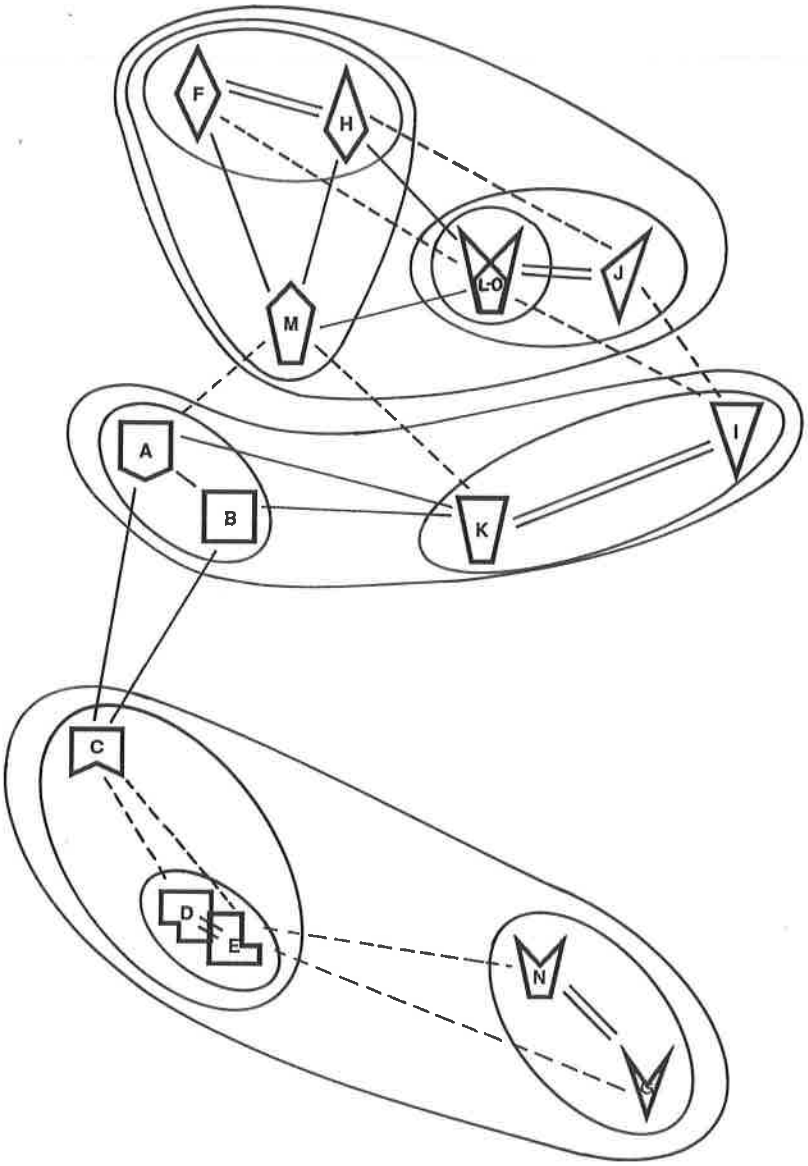


Figure 15

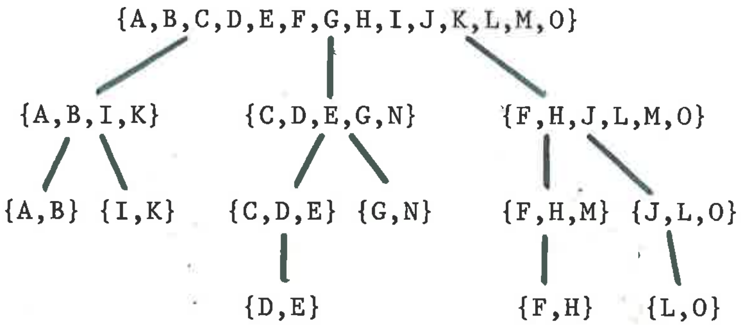

{A,B,C,D,E,F,G,H,I,J,K,L,M,O}
{A,B,I,K}
{C,D,E,G,N}
{F,H,J,L,M,O}
{A,B}
{I,K}
{C,D,E}
{G,N}
{F,H,M}
{J,L,O}
{D,E}
{F,H}
{L,O}

Figure 16

be made at any orientation with only a penalty for a difference in the overall orientation of the two polygons. The same phenomenon brings C and D even closer together in this algorithm: in this case a rotation of about 45o is needed to match them up. Note that the triple {J,L,O}, consisting of the mirror image quadrilaterals L and O with sharp tops and the triangle J with a sharp top similar to O, appears in the clusterings for $X_{str,hum}$. This triple is characteristic of the human confusions. Again, it shows up because mirror-image matchings with low cost are chosen by this algorithm with only a small penalty at the end for the flip. The form of the function d(P,Q) was chosen precisely to see if we could model mirror-image confusions in this way.

Aside from these differences, the two structural algorithms produce quite similar patterns: the top level grouping {F,H,J,L,M,O} consisting of the set of convex polygons with pointed tops shows up in both, and the top level grouping of polygons with flat tops also shows up in both, except that for $X_{str,pgn}$ it includes the non-convex ones and for $X_{str,hum}$ it does not.

The structural matching algorithms show considerable differences in detail from the template matchings. To give two examples, the top-heavy symmetrical triangle I is closer to the coffin-shape M than to the elongated asymmetrical triangle J in the template match $X_{tem,pgn}$ but vice versa for the $X_{str,pgn}$. And the trapezoid K is closer to the diamond F than to the square B for the template match $X_{tem,pgn}$, and vice versa for the structural one $X_{str,pgn}$. This illustrates how shapes may be close pixel by pixel without the edges lining up well, and how the edges may line up nicely yet the shapes be fairly far apart pixel by pixel.

SECTION V: CONCLUSIONS

To put the results in perspective, consider first Table 8, which contains the correlations of the following six matrices:

i) the normalized and symmetrized pigeon error rates,

ii) the symmetrized human response times,

iii-iv) the monotone functions of the two best-fitting computer models of the pigeon data,

v-vi) the monotone functions of the two best-fitting computer models of the human data.

Insert Table 8 about here

In Table 8, the decimal figures are the product-moment correlations and the percents below are their squares - the variance accounted for. These results show that a) the human data and human structural algorithm are similar; b) the two template algorithms are similar; c) the template and structural algorithms matched to pigeon data are fairly similar, and d) both of these are only moderately good predictors of the pigeon data.

The small entry in this table for the correlation between human and pigeon experimental data should not be directly compared with the other entries. This is because all the other entries are correlations between monotone functions of the computer generated data and the experimental data. Correlation after a monotone rescaling is

Table 8

	pigeon	human	temp-pgn	str-pgn	temp-hum	str-hum
pigeon	*	.44 (19%)	.69 (47%)	.53 (28%)	.67 (45%)	.41 (16%)
human		*	.43 (18%)	.65 (42%)	.44 (19%)	.80 (64%)
temp-pgn			*	.79 (62%)	.75 (57%)	.37 (14%)
str-pgn				*	.61 (38%)	.57 (33%)
temp-hum					*	.47 (22%)

naturally much better than without it. If a monotone rescaling of the human or pigeon data is done, we get better fits as follows:

A) correlation of 0.56 (variance accounted for 31%) between human data and monotone function of pigeon data.

B) correlation of 0.51 (variance accounted for 26%) between pigeon data and monotone function of human data.

A correlation was also calculated for the skew-symmetric parts of the human response time data and the normalized pigeon error rates. This gave a correlation of .04, indicating again that it is hard to make good sense of the asymmetry of these matrices.

Putting this story together, here are the main conclusions that our analysis leads us to make:

i) Pigeons are capable of forming abstract categories of shapes, such as those characterized by the number of vertices of a polygon, and, in doing so, of disregarding the orientation and proportions of the polygon as unimportant.

Clear evidence for this was the tight cluster formed by the five skew quadrilaterals F,H,K,L and O, as well as the large confusion between the very different triangles I and J. These clusters did not appear in any of the computer simulations. We may speculate that a form of ´shape constancy´ causes the pigeons to cluster these shapes. More precisely, note that a rectangular shape on the ground seen from the air is a skew quadrilateral, and hence it is possible to interpret all five skew quadrilaterals as aerial views of one and the same flat rectangle seen from above from different positions[14]. For ecological reasons, this might lead a bird to consider skew quadrilaterals as similar shapes.

The ability to form such abstract categories argues against the theory that an animal with as small a brain as a pigeon must be using simple discrimination routines, such as perceptrons (Minsky and Papert, 1969). Moreover, work in pattern recognition in the 1970′s, which was mostly based on perceptron-like discriminators, failed to find robust algorithms capable of categorizing real-world stimuli. Because pigeons are manifestly capable of such categorizing (viz. Herrnstein, 1984), this also argues against modelling pigeons by perceptrons or other such elementary discrimination algorithms.

ii) Pigeons also have the orientation of the polygon available to them, and, when the task demands it, will use this orientation as a clue.

The pigeons formed strong clusters out of the polygons {A,B,C,D,E}, presumably on the basis of the presence of horizontal and vertical lines, disregarding vertex number: the set includes figures with 4, 5 and 6 vertices. One could speculate that figures with vertical edges would not be clustered with the skew quadrilaterals because such a retinal image is more likely to arise from a naturally vertical object (such as a tree) seen from the ground than a flat rectangle viewed from above. Much evidence suggests that birds can pick and choose among various features in order to form the appropriate positive category (e.g., Hrycenko and Harwood, 1980; Herrnstein, 1984).

iii) The pigeon´s performance is strongly affected by the overall task; i.e., they paid attention to the aspects of the shapes that were most useful for distinguishing exactly among the members of the stimulus set.

In other words, the polygons were not viewed in isolation, but the features that most clearly distinguished each polygon from the rest came to govern response. Thus the categories were strongly affected by the **context** in which the categorization took place. For example, the stimulus set contained only two polygons G and N with concave tops, all the rest having flat or pointed convex tops. Therefore, these figures could be distinguished from the other 13 polygons on the basis of this feature alone. But the negative side of this is that G and N were strongly confused with each other because other features such as their bottoms or their areas were not so useful in most of the discriminations. Note, for instance, that the figures I and K, which were essentially G and N with tops filled in, were easily distinguished because they were classed with triangles and skew quadrilaterals, respectively. The overriding significance of the context was what determined whether the pigeon used number of vertices (a global property) or presence of vertical edges (a local property) in judging the stimulus. In fact, polygons similar by every computer matching algorithm, such as J and O, were easily distinguished when context led the pigeon to pay attention to aspects in which they clearly differed (here, vertex number).

This selection of discriminating features based on the whole stimulus set appears to be the fundamental reason why the pigeon error rates were not well modelled by any of the computer algorithms. In order to model their behavior better, one should probably look at **learning** algorithms, such as those used in connectionist models (Rumelhart and McClelland, 1986).

iv) Pigeons evidently vary in their ability to keep multiple aspects of the shapes ´in mind´ while making judgments and not to simplify the problem by focusing on only one or two main properties of the shapes.

Thus the data of bird 4, like the human data, and unlike the computer data from all algorithms used, did not have a good two-dimensional scaling plot. If its discrimination had been based on a small number of features, it is likely that its error rates would have such a plot.

v) Humans, in a speeded response time task, access fastest the large-scale features of shapes (e.g., their area) and do not use finer scale aspects (e.g., vertices with angles close to 180 degrees) nor global properties (e.g., the number of vertices).

This inference is supported by the best-fitting template match. The matrix of this algorithm was found by averaging 90% of template distances using the top blur (7th out of 7) and a mere 10% of blur 5.

The algorithm, as we saw, was essentially a two-feature algorithm, based on area and ´boxiness´ (see section IV). Moreover, all of the human clusters with more than 2 polygons contained polygons with differing numbers of vertices. For example, the triangle I was much harder to distinguish from the quadrilateral H than from the second triangle J, indicating that the three vertices I and J had in common did not distract humans from deciding they were different.

vi) Humans identify characteristic parts, or even the whole, of shapes by methods that, at first pass, disregard the orientation of the shape.

Evidence for this inference comes from the two-stage structure of the best-fitting structural match, wherein the shapes of parts and wholes were matched first disregarding the orientation, and then the differing orientations of whole figures were taken note of. Evidence also comes from the rather astonishing difficulty that humans had in distinguishing the top-heavy (downward pointing) triangle I from the bottom-heavy quadrilateral H, which it resembles only after a 180 degree flip. The same observation holds for E and F, which are box-like shapes with a bite taken out of them - but where the bite is on the bottom left for E and the top right for F. These were strongly confused, but their characteristic features line up only after a 180 degree rotation. Still another example is the mirror-image shapes L and O.

Everyone is familiar with the effort it takes to remember one of two mirror-image shapes for some task (e.g., Nickerson and Adams, 1971). This property places major constraints on theories of the data structure used at least under some circumstances by the human brain to store shapes: even crude template schemes would avoid this sort of difficulty. The structural matching scheme used for humans was, therefore, purposely crafted to yield close matches between mirror image shapes. Humans may note orientation in many recognition tasks, but, our results suggest, certain visual routines (to use Ullman´s [1984] idea) seem to proceed without regard to orientation.

vii) _People, like pigeons, are able to choose an effective algorithm from among many available, thus alternating between template comparison and a search for a single clear distinguishing feature if one is available._

Consider, for example, the trapezoid K versus the triangle I and the coffin shaped figure M. K is close to both M and I from a template standpoint. But I was rapidly distinguished from K by the strong feature given by the characteristic sharp angle at the bottom of I. M, however, has no such distinctive feature and was indeed hard for the humans to distinguish from K.

viii) _Neither pigeon nor human seems to use template or structural representations to the exclusion of the other. However the human performance is significantly better modelled by the structural matching algorithm._

Both schemes seem to be available to both species, with the task requirements leading the subject to select one over the other. In fact, each scheme seems to capture some details of the data for pigeons and humans not modelled by the other. For humans, the structural matching scheme seems to model the data significantly better than the template one, achieving a correlation of .80.

APPENDIX I: TEMPLATE MATCHING

In our implementation, we started from 64x64 pixel digitizations of the stimulus polygons. Each polygon was translated so that its centroid was at the center point (32,32), and then the functions that were 1 inside the polygons and 0 outside were convolved with seven Gaussian masks whose standard deviations were approximately 1,2,3,4,5,6 and 8 pixels. Including the unblurred polygon, this gives eight images for each polygon. For each polygon and each of these blurrings, three auxiliary images were computed next: one for the i-derivative, one for the j-derivative and one for the Laplacian. Finally, computing correlations and Euclidean distances, we took each pair of polygons and compared:

i) each blurred version of the polygons,
ii) the **gradients** of each blurred version (i.e., we took the i- and j-derivatives at each pixel, put these together into a long vector with 64x64x2 components, and compared these),
iii) the Laplacian of each blurred image.

The result was a set of 8x3x2 symmetric 15x15 matrices, each of which was a slightly different way to measure the template match of the 15 polygons. This came from the 8 possible blurring levels and the 3 derivative types and the two ways of comparing (correlation and Euclidean distance).

A reasonable way to find **one** best template match matrix seemed to be to choose three blurring levels, b0 for undifferentiated images, bl for the gradient and b2 for the Laplacian, and three weights w_0,w_1 and w_2 and form the combination:

$$w_0*C_{b0,0} + w_1*C_{b1,1} + w_2*C_{b2,2}.$$

where we have denoted by C_{bd} the matrix of correlations with blurring b and derivative type d (d = 0,1 or 2). The same thing can be done with Euclidean distances, which we denote by L_{bd}. To choose the best w´s and b´s, we used the regression of both the pigeon and human data against the template data and attempted to maximize the variance accounted for. We also had a choice as to whether to use ordinary regression or monotone regression at this point.[15] In view of the fact that there is no reason to expect a **linear** relation between template matching and pigeon or human error rates, we chose monotone regression. Note that this procedure involves matching 105 data values by a model with six free continuous parameters (w_0, w_1, w_2, b_1, b_2, b_3) and one binary parameter (correlation vs. Euclidean distance). But using monotone rather than linear regression means that the measure of goodness of match is lenient.

After some exploration, the mixture that best matched the pigeon data obtained from all good trials used correlations as follows:

$$X_{tem,pgn} = 0.7 * C_{3,0} + 0.1 * C_{6,1} + 0.2 * C_{6,2}.$$

$X_{tem,pgn}$ expresses a large contribution from the correlations between slightly blurred polygons themselves, added to a smaller contribution from the correlations of heavily blurred differentiated images. The mixture that best matched the human data used Euclidean distances as follows:

$$X_{tem,hum} = 0.7 * L_{7,0} + 0.2 * L_{7,1} + 0.1 * L_{5,2}.$$

$X_{tem,hum}$ expresses a large contribution from the Euclidean distances of **heavily** blurred polygons, plus smaller contributions from differentiated images emphasizing the edges and the vertices.

APPENDIX II STRUCTURAL MATCHING

We considered two ways to implement a structural matching algorithm between polygons: one defined the edges of a polygon to be its parts, and the other defined both the edges and the corners to be its parts (the ´part´ associated with a corner means a small neighborhood of the corner, showing the orientation of the two adjacent edges). In matching two polygons P and Q, the edges of P must be matched to the edges of Q and, if corners are also used, the corners of P to the corners of Q. Many different measures may be used to describe the difference of two edges or two corners, considered as fragments of a full geometric figure. We will describe below the measures used in our programs.

If only edges are taken as parts, each edge is considered adjacent to the previous and succeeding edges as the perimeter of the polygon is traversed. Thus, an allowable match would be given by starting at one pair of matching edges, proceeding around the both polygons in clockwise order and matching each successive edge as it is reached. One way to model the susceptibility of human perception to mirror image confusions is to also consider matches in which the edges of P, taken in clockwise order, are matched to the edges of Q, taken in counter-clockwise order.

If both edges and corners are taken as parts, an edge is considered adjacent to the two corners at its ends, and a corner is considered adjacent to the two edges abutting it. Thus, an allowable match between two polygons would be given by matching all edges and corners as the two polygons are circumambulated, as before. In both cases, this would not, however, allow for any matches between polygons with different numbers of vertices, as it would make them infinitely different and therefore we also allow the polygons being matched to be first ´refined´ by two procedures:

a) an edge of a polygon may be split by adding a corner somewhere in the middle. Thus the edge is replaced by two edges and an extra corner is added.

b) a corner of a polygon may be refined by being truncated. Thus a new very short edge appears where the corner was, and two new vertices are added connecting the new edge to the old abutting edges.

Putting these ideas together, we measure the difference between two polygons P and Q as follows: we first choose a refinement of P and a refinement of Q and then a perfect matching between these refinements. Each such match has a cost, and the difference between P and Q is the minimum cost over all refinements and all matchings. More precisely, a match is a sequence of matches of the parts of P and Q, each of which is one of 6 types:

a) a match of an edge of P with an edge of Q,

b) a match of a corner of P with a corner of Q,

c) a match of two abutting edges of P and their common corner with an edge of Q that is split into two edges and a corner in the refinement of P,

d) the same with the roles of P and Q reversed,

e) a match of an edge of P plus the two corners at its ends with a single corner of Q that is truncated to a short edge and two corners in the refinement of Q,

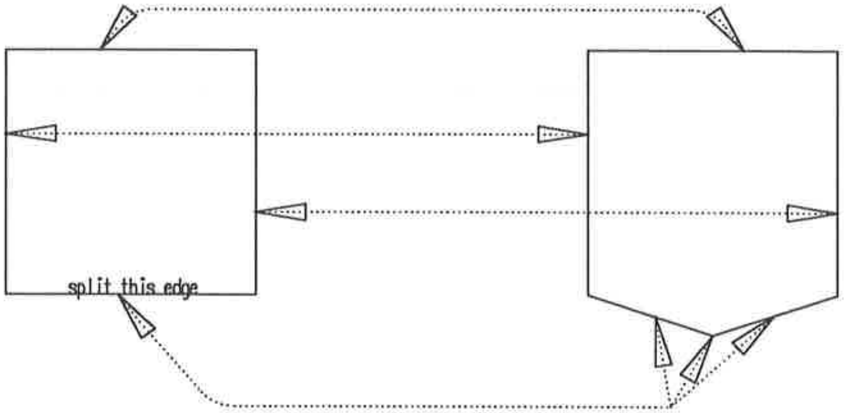
f) the same with the roles of P and Q reversed.

The sequence of matches must go once around P and once around Q, matching each edge and corner of each polygon exactly once. See Figure 17 for some examples.

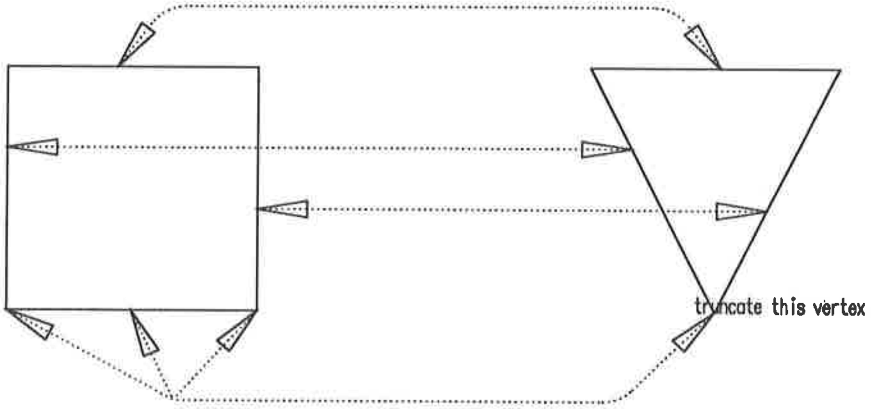
Insert Figure 17 about here

To compute the cost of a match, a function d(x,y) is needed. In the first type of algorithm only edges are used as parts. Taken out of the polygon containing it, an edge is just a directed line segment (the direction is given by the rule that the polygon is on the right when you traverse the edge from beginning to end). We measured the difference between two directed line segments by considering the difference in their lengths and their orientations.

The second algorithm used both edges and corners as parts. A corner, taken out of its polygon, is considered to be a vertex and two half lines radiating from this vertex, one to the left of the figure, one to the right. In these algorithms, we ignored the orientation of



Matching with refinement of bottom edge of left polygon



Matching with truncation of bottom vertex of right polygon

Figure 17

both the edges and corners and measured the difference between two directed line segments using only the difference in their lengths and the difference between two corners using only the difference in the internal angles of the polygon at these corners. This allows a perfect match between two polygons that differ only by a rotation or a reflection. Of course, vision systems must perceive the difference between the original and a rotated or reflected copy, so we added a second term to the cost of the best match to penalize for how much the match, overall, rotates or reflects P when matching it to Q.

We experimented with various choices of d(x,y) both for directed line segments and corners, as well as how to extend the definition of d to the cases where P or Q is refined. We also considered extra terms in the cost of a match f that matches a concave corner of P with a convex corner of Q or vice versa, or that matches parallel edges of P with non-parallel edges of Q or vice versa. Inasmuch as these experiments were driven by the goal of providing the best match to the pigeon and human data, we will not describe in detail all the variants that were tried, but only the best fitting structural matching algorithms.

In matching the pigeon data, the best fit was found using only edges as parts and measuring the difference of two directed line segments x and y by a formula that emphasizes the similarity of horizontal lines with each other and the similarity of vertical lines with each other. If we write x as a vector from the origin to (x_1,x_2) and y as a vector from the origin to (y_1,y_2), we then set:

$$d(x,y) = (x_1-y_1)^2/(|x_1|+|y_1|) + (x_2-y_2)^2/(|x_2|+|y_2|).$$

Note that d = 0 if and only if x = y (i.e., x and y have the same length and orientation). This formula was extended to allow refinements as follows. If an edge x was matched with a truncated corner of the other polygon, then the cost used was $C_{collapse}$*d(x,(0,0)), ($C_{collapse}$ is a constant). If two abutting edges x and x´ were matched with a split edge y of the other polygon, then the cost used was C_{split}*(d(x,u) + d(x´,v)), where C_{split} is a constant and the directed line segment y was split into two pieces u and v:

w = projection of the vector x-x´ onto the line through y,

u = 0.5 * (y + w), (vector addition of y and w),

v = 0.5 * (y - w), (vector subtraction of y and w).

Finally, if a concave corner a of P or Q is matched to a non-concave corner of Q or P, an extra penalty C_{ext} * d(x,(0,0)) is added, where x is the directed line segment spanning the ´mouth´ of the concavity of P or Q, i.e. from the corner before a to the corner after a.

Note that we have introduced three parameters $C_{collapse}$, C_{split} and C_{ext} into the algorithm, which weight the various components of the cost. However, after some experimentation, the best fit found was obtained by setting all three parameters to 1.0.

In matching the human data, the best fit was obtained by using edges and corners as parts, and measuring d from lengths and internal angles of the polygon without regard to orientation. Thus if x and y are edges,

$$d(x,y) = \mid \text{length}(x) - \text{length}(y) \mid .$$

If x and y are corners,

$d(x,y) = | \cos(\text{angle}(x)/2) - \cos(\text{angle}(y)/2) |$.

We chose cos(angle(x)/2), instead of using simply angle(x), because of evidence that human observers are more sensitive to angle variation near the angle 180o than near smaller angles (Foster, 1982). The extensions of these measures when P or Q is refined are shown in Figure 18.

Insert Figure 18 about here

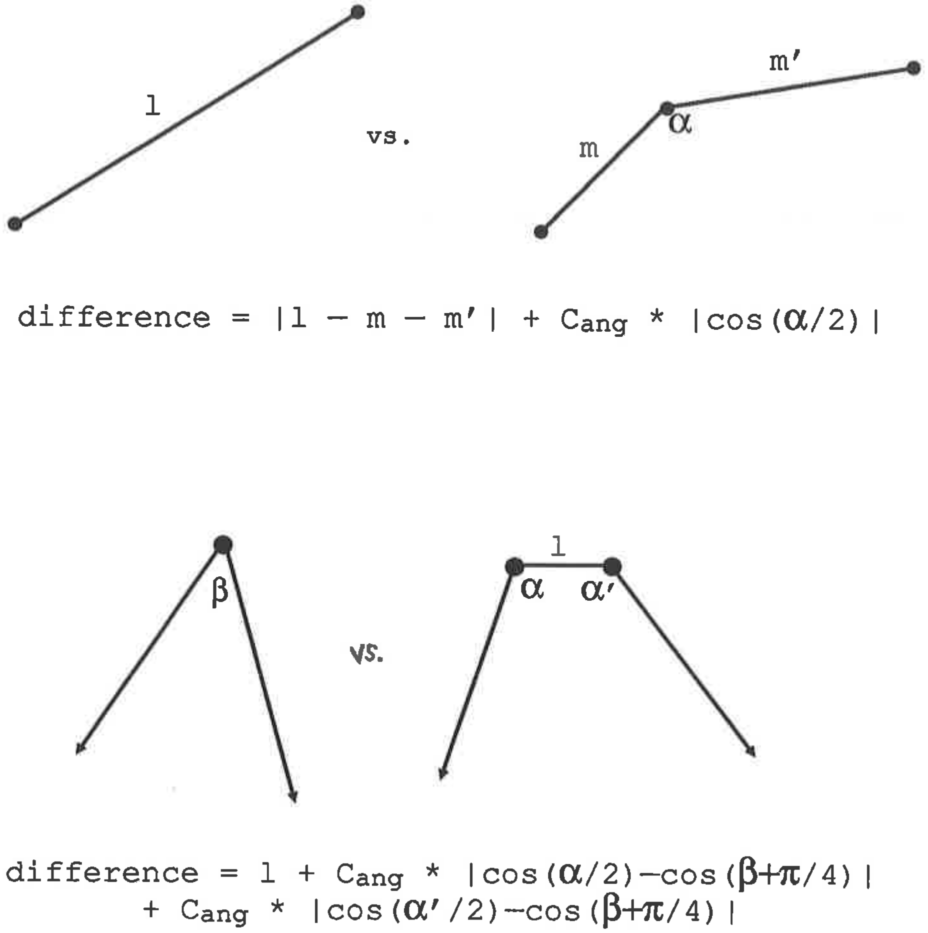
In this algorithm, the final measure of the difference of two polygons P and Q was computed as follows:

a) First find the match of least cost, allowing refinements and a rotation, measured by the d´s described above, between P and Q and between P and the mirror-image Q´ of Q (with respect to a vertical line). In the cost function, the penalties for angle discrepancies are multiplied by a constant, C_{angle}, before being added to the penalties for length discrepancies.

b) Second, find the angle of rotation that best fits the matches of the individual parts of P and Q in the best matches of a).

c) Third, combine this data via the formula:

$$d(P,Q) = \min \Big[(\text{cost of best match of } P,Q) + C_{rot} * (1.0 - \cos(\text{angle of this match})), \\ (\text{cost of best match of } P,Q') + C_{flip} + C_{rot} * (1.0 - \cos(\text{angle of this match})) \Big].$$



$$\text{difference} = |l - m - m'| + C_{ang} * |\cos(\alpha/2)|$$

$$\text{difference} = l + C_{ang} * |\cos(\alpha/2) - \cos(\beta+\pi/4)| + C_{ang} * |\cos(\alpha'/2) - \cos(\beta+\pi/4)|$$

Figure 18

Note that this whole procedure has three constants in it, which weight the various terms in the cost function: C_{angle}, C_{rot} and C_{flip}, although, of course, the form of the cost function involves a number of binary choices. After some experimentation, the best fit found was the choice of constants:

$C_{angle} = 3.6$

$C_{rot} = 5.2$

$C_{flip} = 0.5$

REFERENCES

Attneave, F. (1959). **Applications of Information Theory to Psychology,** New York: Holt, Rinehart & Winston.

Blough, D.S. (1982). Pigeon perception of letters of the alphabet, **Science, 218**, 397-398.

Coffin, S. (1978). Spatial frequency analysis of block letters does not predict experimental confusions. **Perception and Psychophysics, 23**, 69-74.

Daugman, J. (1987). Image analysis and compact coding by oriented 2D Gabor primitives, **SPIE Proc., 758.**

Duda, R. and Hart, P. (1973). **Pattern Classification and Scene Analysis**, Wiley.

Efron, B. (1979). Bootstrap methods, another look at the jackknife, **Ann. Statist., 7**, 1-26.

Efron, B. and Tibshirani, R. (1986). Bootstrap Methods for Standard Errors, Confidence Intervals, and other measures of Statistical Accuracy, **Statist. Sci., 1**, 54-77.

Fischler, M.A. & Elschlager, R.A. (1973). The representation and matching of pictorial structures, **IEEE Trans. on Computers, C-22,** 67-92.

Foster, D. (1982). Analysis of Discrete Internal Representations of Visual Pattern Stimuli. In Beck, J. (Ed.), **Organization and Representation in Perception**, Hillsdale, NJ: Lawrence Erlbaum, (pp.319-341).

Fu, K.S. (1982). **Syntactic Pattern Recognition and Applications.** Englewood Cliffs, NJ: Prentice-Hall.

Gibson, E.J. (1969). **Principles of Perceptual Learning and Development**, NY: Appleton-Century-Crofts.

Herrnstein, R.J. (1984). Objects, Categories and Discriminative Stimuli. In Roitblat, H.L., Bever, T.G. & Terrace, H.S. (Eds.), **Animal Cognition**, Hillsdale, NJ: Lawrence Erlbaum, (pp.233-261).

Herrnstein, R.J., Loveland, D.H. & Cable, C. (1976). Natural concepts in pigeons, **J. of Exp. Psych.: Animal Behavior Processes, 2**, 285-302.

Holbrook, M.B. (1975). A comparison of methods for measuring the inter-letter similarity between capital letters, **Perception & Psychophysics 17**, 532-536.

Hrycenko, O. & Harwood, D.W. (1980). Judgments of shape similarity in the Barbary Dove, **Animal Behavior, 28**, 586-592.

Karten, H.J., Hodos, W., Nauta, W.J.H. & Revzin A.M. (1973). Neural connections of the Visual Wulst of the avian telencephalon, **J. Comp. Neurology, 150**, 253-278.

Keren, G. & Baggen, S. (1981). Recognition models of alphanumeric characters, **Perception & Psychophysics, 29**, 234-246.

Kosslyn, S. (1980). **Image and Mind.** Cambridge MA: Harvard University Press.

Kruskal, J.B. (1964). Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis, **Psychometrica, 29**, 2-27.

Lindsay, P.H. & Norman, D.A. (1977). **Human Information Processing.** New York: Academic Press.

Lowe, D.G. (1987a). Three-dimensional object recognition from single two-dimensional images, **Artificial Intelligence, 31**, 355-395.

Lowe, D.G. (1987b). The viewpoint consistency constraint, **International Journal of Computer Vision,** 1, 57-72.

Maeda, K., Kurosawa, Y., Asada, H., Sakai, K. & Watanabe, S. (1982). Handprinted Kanji Recognition by Pattern Matching Method, **Proc. 6th Int. Conf. Pattern Recognition**, 789-792.

Marr, D. (1982). **Vision.** San Francisco, CA: W.H.Freeman.

Mazur, J.E. & Hastie, R. (1978). Learning as accumulation: a reexamination of the learning curve, **Psych. Bull. 85**, 1256-1274.

Miller, G.A. & Johnson-Laird, P.N. (1976). **Language and Perception.** Cambridge, MA: Harvard University Press.

Minsky, M. & Papert, S. (1969). **Perceptrons.** Cambridge, MA: MIT Press.

Mori, J.S., Yamamoto, K. & Yasuda, M. (1984). Research on Machine Recognition of Handprinted Characters, **IEEE Transactions on Pattern Analysis and Machine Intelligence, 6**, 386-405.

Nickerson, R.S. & Adams, M.J.,(1971). Long term memory for a common object, **Cognitive Psych., 11**, 287-307.

Palmer, S.E. (1977). Hierarchical structure in perceptual representations, **Cognitive Psychology, 9**, 441-474.

Pasternak, T. & Hodos, W. (1977). **J. Comp. Physiol. Psychol. 91**, 485-497.

Podgorny, P. & Garner, W.R. (1979). Reaction time as a measure of inter- and intraobject visual similarity, **Perception & Psychophysics, 26**, 37-52.

Reed, S.K. (1974). Structural descriptions and the limitations of visual images, **Memory and Cognition, 2**, 329-336.

Reed, S.K. & Johnsen, J.A. (1975). Detection of parts in patterns and images, **Memory and Cognition,** 3, 569-575.

Revzin, A.M., (1969). A specific visual projection area in the hyperstriatum of the pigeon, **Brain Research, 15**, 246-249.

Rosenfeld, A., Hummel, R. & Zucker, S. (1976). Scene labelling by relaxation operations, **IEEE Trans. Systems, Man & Cybernetics, 6,** 420-433.

Rosenfeld, A. & Yamamoto, K. (1982). Recognition of hand-printed Kanji characters by a relaxation method, **Proc. 6th Int. Conf. on Pattern Recognition**, 395-398.

Rumelhart, D.E. & McClelland, J.L. (Eds.) (1986). **Parallel and Distributed Processing, Vols. I and II.** Cambridge, MA: MIT Press.

Sattath, S. & Tversky, A. (1977). Additive similarity trees, **Psychometrica, 42**, 319-345.

Townsend, J.T. (1971). Theoretical analysis of an alphabetic confusion matrix, **Perception and Psychophysics, 9**, 40-50.

Tversky, A. (1977). Features of similarity, **Psych. Rev.,84**, 327-352.

Vaughan, W. & Greene, S.L. (1984). Pigeon visual memory capacity, **J. Exp. Psych.: Animal Behavioral Processes, 10**, 256-271.

Ullman, S. (1979). **The Interpretation of Visual Motion.** Cambridge, MA: MIT Press.

Ullman, S. (1984). Visual routines, **Cognition, 18**, 97-159.

Ullman, S. (1986). **An approach to object recognition: aligning pictorial descriptions.** MIT AI Lab Memo #931, December.

Wolford, G. (1975). Perturbation model for letter identification, **Psych. Rev. 82**, 184-199.

Figure Legends

Figure 1. Fifteen shapes used in the study, for discrimination by pigeons, recognition by people and classification by computer. The shapes have equal perimeters, and fall into three classes, as described in the text.

Figure 2. Histogram of error percentages of the entries in Table 1, for pigeons discriminating among the fifteen shapes.

Figure 3. KYST plot for pigeon learning error data, with clusters and nearest neighbors.

Figure 4. ADDTREE clusters for the matrix X_{pgn} of pigeon error rates.

Figure 5. Pattern of greatest confusions in responses of avian subject 4, exhibiting exceptionally fast learning.

Figure 6. Histogram of response times of the entries in Table 2, for people performing the recognition task with the fifteen shapes.

Figure 7. ADDTREE clusters for the matrix X_{hum} of human response times.

Figure 8. KYST plot for human response time data, with clusters and nearest neighbors.

Figure 9. ADDTREE clusters for the matrix $X_{tem,pgn}$.

Figure 10. ADDTREE clusters for the matrix $X_{tem,hum}$.

Figure 11. KYST plot for differences of polygons measured via template matches, parameters optimized to fit pigeon data.

Figure 12. KYST plot for differences of polygons measured via template matches, parameters optimized to fit human data.

Figure 13. KYST plot for differences of polygons measured via structural matches, parameters optimized to fit pigeon data.

Figure 14. ADDTREE clusters for the matrix $X_{str,pgn}$.

Figure 15. KYST plot for differences of polygons measured via structural matches, parameters optimized to fit human data.

Figure 16. ADDTREE clusters for the matrix $X_{str,hum}$.

Figure 17. Illustrations of the polygon matching scheme with refinement and truncation.

Figure 18. The expressions used to define the difference of two polygons when they are matched with refinement or truncation.

Endnotes

1. The authors gratefully acknowledge the support of the National Science Foundation grant IST- in the research and preparation of this paper. They would also like to acknowledge the capable assistance of Lynn Hilger in the human experiments and James Herrnstein in programming the template algorithm. Reprint requests may be addressed to any of the authors.

2. In mathematical jargon, we can, for instance, consider the set of open subsets in the plane R^2 which are the interior of their closures and whose boundary consists in a finite set of simple closed piecewise infinitely differentiable curves in R^2.

3. Again, we may make this precise by using a variant of the so-called Hausdorff metric. To compare two shapes A and B, we want to say that A and B are close if every point of A is near some point of B, every point of B is near some point of A, every point of the boundary of A is near some point of the boundary of B with approximately the same orientation and, finally, every point of the boundary of B is near some point of the boundary of A with approximately the same orientation. More precisely, we can measure the ´difference´ of points P on the boundary of A and Q on the boundary of B by the sum of the distance between P and Q and the angle between the tangent lines to A at P and to B at Q. Then the distance between two shapes A and B is defined to be the sum of the four numbers:

a) the maximum distance d of any point in A from the shape B (so that d equals 0 if A is a subset of B),

b) the maximum distance e of any point in B from the shape A,

c) the maximum difference f, as defined above, of any point on the boundary of A from the boundary of B,

d) the maximum difference g of any point on the boundary of B from the boundary of A.

Note that this is zero if and only if A is exactly the same shape as B. This definition makes S into a metric space. Thus the concept of a real-valued function on S being continuous or not can be defined.

4. One way to see this is to consider a particular family of shapes as follows: let A be the circle in the plane with center equal to the origin and with radius 2. For each point (x,y) in the plane, let B(x,y) be the circle with center (x,y) and with radius 1. Consider the family of crescent moon-like shapes:

$$A - B(x,y)$$

(the points in A but not in B(x,y)). Each shape A - B(x,y) determines a point P(x,y) in S. But note that if (x,y) is farther than 3 from the origin, then A and B(x,y) don´t overlap, so then A - B(x,y) equals A. Thus:

$$A - B(x,y) = A - B(u,v)$$

if **both** (x,y) and (u,v) are farther than 3 from the origin. P(x,y) therefore equals P(u,v) in this case. Now think of the family of points P(x,y) as defining a map of the x,y-plane into S. In this map, all points (x,y) farther than 3 from the origin are mapped to the same point (the shape A). This means that the points P(x,y) don´t form a flat plane inside S but instead roll up into a 2-dimensional sphere in S! In the space S, with the topology defined above, there does not seem to be any way to fill in this sphere with a 2-dimensional ball of further shapes: this is what we mean by saying that S is not a flat space.

5. The outputs of a set of "Gabor" filters (Daugman, 1987) are one possibility for coordinates, but these do not make topological properties of shapes, such as connectedness, explicit and it is not clear whether the nearness of these coordinates implies the similarity of the corresponding shapes.

6. The use of the term dimensions may be a bit misleading. Thus Marr´s 2 1/2-D sketch is a special kind of two-dimensional structure, namely one in which distance to the nearest surface and the slope of the surface at that point are recorded for all directions in a two-

dimensional array. Marr´s 3D model, on the other hand, is a special type of structured network.

7. It is important to realize that KYST does not guarantee that it finds the best possible configuration, because it works by ´hill-climbing´, but one can start it either with an initial guess derived from factor analysis, or with a random initial guess, or one can start with a high-dimensional fit and reduce the dimension one at a time. We have played with its options in each case, seeking what seemed to work best in terms of ´stress.´ All runs are based on the Fortran program, KYST-2, distributed by AT&T in the "2nd Multidimensional Scaling Program Package."

8. To compute rho, rank all slides in a trial by numbers of pecks. Then rho is the average rank of the slides that are positive, but scaled so that rho = 1.0 if all positive slides were ranked at the top, and rho = 0.0 if all positive slides were ranked at the bottom.

rho is Greek letter ρ

9. As noted earlier, rho is the probability of ranking a positive stimulus above a negative stimulus. See Herrnstein, Loveland, and Cable (1976) for further discussion of its characteristics.

10. This follows from the theory of finite Markov chains. Consider the confusion matrix A_{ij} with diagonal entries set to 0. Then A is the matrix of a Markov chain, and if every stimulus i can be confused with every stimulus j, A is certainly an irreducible Markov chain. If l_i is the stationary distribution of A, then the new matrix $B_{ij} = l_i * A_{ij}$ has row sums equal to its column sums.

11. The mean response time for different subjects varied from 334 msec to 600 msec and 10% of the responses from one subject were under 200 msec (all were correct).

12. Unfortunately, it is not clear how to perform a similar analysis on the pigeon data because of the difficulty, mentioned above, of comparing responses to different positive stimuli. In particular, the procedure used to normalize the rows made the rows and columns equal and so precludes an ordering such as that for the human data.

13. Note that these template measures are defined for any images I and J, not just those that arise from the silhouettes of two shapes S and T. The two formulas are closely related, one being computable from the other if the "size" of the images I and J -- the two terms in the denominator of the correlation -- are known. If the images I and J are silhouettes arising from figures S and T respectively, then the "size" of I and J are simply the areas of S and T and the Euclidean distance is simply the sum of the area of S-T plus the area of T-S, whereas the correlation is the ratio of the area of overlap to the geometric mean of the areas of S and T.

The basic difference between these two forms of template matching, at least for the case of silhouettes, is this: when the **scale** of the image is changed (e.g., both S and T are expanded), then the correlation remains unchanged, but the Euclidean distance between the two images is increased. Correlation should therefore be used if what is wanted is a measure of difference that disregards the absolute size of S and T (but not their relative size), whereas Euclidean distance should be used if larger size is supposed to make the differences between S and T more pronounced. Another way of describing the difference is to imagine putting a bump of fixed size on the side of a large and a small box. Then the Euclidean distance between the large box without the bump and the large box with the bump is just the area of the bump. This is the same as the distance between the small box without the bump and the small box with the bump. But the large box with and without the bump are more highly correlated with each other than the small box with and without the bump.

14. Mathematically, the point is that, given any flat rectangle on the ground and any skew quadrilateral on a screen, there will be viewing positions in the air or in front of the screen for which these two quadrilaterals generate the same retinal stimulus.

15. To fix notation, ordinary regression of data a_i against data b_i chooses a linear function l(x) which minimizes:

$$\sum_i \left[(a_i - l(b_i))^2 \right],$$

and monotone rescaling chooses a monotone increasing or decreasing function l to minimize the same sum. We also used a mixture, monotone cubic regression, where l is required to be a cubic polynomial which is monotone increasing or decreasing in the interval spanned by the smallest and largest b_i.