Building a Distance Function for Gestalt Grouping

ALBERT L. ZOBRIST, MEMBER, IEEE, AND WILLIAM B. THOMPSON, MEMBER, IEEE

***Abstract*—A central problem in the area of scene analysis is that of segmenting a scene into its natural objects. Current work emphasizes the semantic approach in which *a priori* knowledge of the shape of an object is used. Yet there is much to learn about more primitive cues for segmentation such as texture, color, and brightness. In the case of human perception, segmentation appears to be due to a multiplicity of cues which operate in a redundant fashion.**

This paper describes a method for combining multiple cues. To make the cues commensurable, each is regarded as a primitive distance function on pairs of regions of a scene. The total tendency of two regions to be grouped together or segmented apart is measured by a linear sum of the primitive distance functions. Computer-aided psychophysical experiments are described which test how closely the total distance function simulates human perception. The results also give guidance for the further implementation of primitive distance functions. The methodology is emphasized, but interesting results are also reported.

***Index Terms*—Clustering, Gestalt clustering, image processing, object identification, pattern recognition, psychophysics, robot vision, scene analysis, texture.**

INTRODUCTION

A CENTRAL problem in the area of scene analysis is that of segmenting a scene into regions which correspond to its natural objects. For example, if a robot is moving about in an environment containing such things as furniture, tools, or doors, then the robot must be able to segregate those things from their surroundings in order to act properly. We will refer to this visual process by the terms grouping or segmentation. In recent years, several efforts have been directed toward the study of physical cues for grouping and computer methods which are effective for a particular type of scene which involves a particular cue. Guzman [1], for example, has written a program which can determine the objects in a rectilinear line drawing of an overlapping jumble of blocks. The primary cues are the T and Y joins of the line drawing, where a T join indicates an edge of one block disappearing under another, and a Y join indicates a three-way corner of a single block [Fig. 1(a)]. Other such corner and edge cues give similar implications. The program then combines all of the implications to decide which faces form blocks. Zahn [2] uses the following type of algorithm for the Gestalt clustering of points. A connected graph is constructed which has the points as nodes and which has the following additional property; the sum of lengths of edges is minimal. Such a graph is called a minimal spanning tree (MST). Any edge which is unusually long compared to its neighbors is a separating edge [Fig. 1(b)]. Removal of a separating edge breaks the MST into parts which define natural clusters of points. The definition of separating edge can be varied to handle different types of grouping problems.

Techniques such as those of Guzman and Zahn work well only if they are applied to neatly defined artificial scenes. For a Guzman scene, the edges must be relatively straight and must meet in well-defined corners. In the real world of robot-video input, or of digitized photographs, problems of focus, resolution, brightness, or shadow cause edges to curve, fade, or even disappear entirely. The eventual necessity of dealing with real world scenes adds a level of complexity to the problem.

With regard to this problem we feel that basic research must now proceed along two lines. The first line of research involves the use of semantics or "world knowledge" to help in the segmentation of imperfect images. For example, if a robot sees most of a box, and it knows what a box should look like, then it can fill in missing lines to complete the box. This direction of research has appeared promising for some time. Some of the basic ideas can be found in an old paper by Roberts [3]. More recent work by Brice and Fennema [4], Waltz [5], Yakimovsky [6], and Harlow and Eisenbeis [7] have produced practical working systems for specialized problems.

The second line of basic research must explore the use of multiple or redundant physical cues for grouping. Pickett [8] points out that some object boundaries are not recognizable as sharp edges of contrasting brightness or color. For example, if a robot sees a box with a missing edge due to lack of brightness difference across that edge, then perhaps the edge will be discernable if there is a contrast in color or texture between the box and its background. Since we know that human vision employs such cues as brightness, contour, color, texture, stereopsis, and relative motion to give perceptual grouping, there is need to try computer simulations of each of these cues. Again, there has been some experimental work with single cues.

Manuscript received June 24, 1974; revised December 12, 1974. This research was sponsored by the U.S. Army Research Office—Durham under Contract DA-ARO-D-31-124-72G200, by the Joint Services Electronics Program through the Air Force Office of Scientific Research/AFSC under Contract F 44620-71-C-0067, and by the Advanced Research Projects Agency of the Department of Defense monitored by the Air Force Eastern Test Range under Contract F 08606-72-C-0008.

A. L. Zobrist is with the Department of Computer Science, University of Arizona, Tucson, Ariz. 85721, on leave from the Computer Science Program, University of Southern California, Los Angeles, Calif. 90007.

W. B. Thompson is with the Image Processing Institute, University of Southern California, Los Angeles, Calif. 90007.

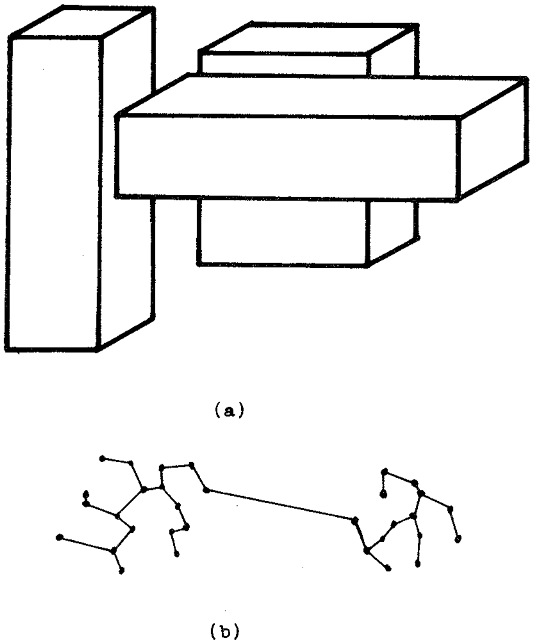


Fig. 1. Grouping problems solvable by (a) Guzman or (b) Zahn approaches.

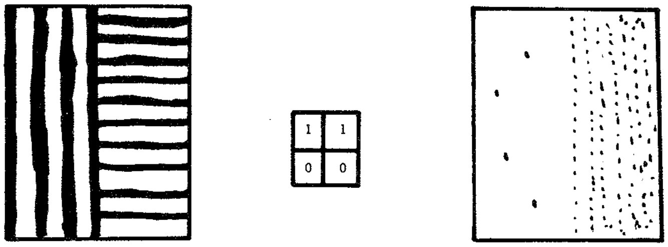


Fig. 2. Illustration of the Rosenfeld–Thurston technique.

Rosenfeld and Thurson [9], for example, apply diameter-limited operators (such as a small edge detector) to a digitized photograph. An area of high activity of one operator is interpreted as a possible Gestalt group. Fig. 2 shows this operation for a horizontal edge detector which searches for two ones above two zeros in a binary digitized image. This approach could employ a large variety of operators to assess a variety of textural differences. However little is known about mechanisms by which disagreements among the operators can be resolved or agreements can be amplified.

The problem most crucial to the second line of research is this; how can the computer make use of multiple or redundant cues to perform grouping? The situation is in curious contrast to classical pattern recognition, where much is known about classification for a given set of features, but little is known about the feature extraction process. Questions of the most basic sort are completely open here. Should simple procedures (such as that of Rosenfeld and Thurston) be applied sequentially or somehow be assessed in parallel? Which of the grouping cues is most effective? How important is context or other semantic information in the grouping process? The last question anticipates the eventual merging of the two lines of research we have delineated.

Our work begins with the following observation. If two parts of a scene are grouped, then they are close to each other in some sense. In the case of human vision this is a perceived closeness, due perhaps to similarity of texture, color, or brightness. To obtain computer grouping, we define a perceptual distance function which is the sum of components which can measure differences of color, texture, or brightness. This achieves the desired goal of integrating multiple cues for grouping into a single mechanism. This report describes the "building" of a distance function for Gestalt grouping which performs well on scenes which contain arrays of discrete figures, and, in a separate implementation, performs grouping on general image textures. Because the particular functions developed here may not be of direct use to others, emphasis is placed upon the methodology involved in building a distance function. Special consideration is given to the psychophysical testing of the distance function which is necessary to yield close simulation of human perceptual grouping.

II. PSYCHOPHYSICS OF GROUPING

Perceptual grouping appears to be the basis of human visual organization. Its importance to human vision has long been recognized by Gestalt psychologists [10]. It is responsible for the formation of wholes from parts, hence for the determination of the objects of a visual scene. Objects formed from parts in a scene may themselves to grouped, thus it appears that perceptual grouping yields a hierarchy of parts and subparts. Fig. 3 shows some of the physical correlates of grouping. In each case, simple differences account for a splitting of the field of vision into two areas.

Grouping is not only basic to the structuring of a scene but is usually a prerequisite for the recognition of objects in a scene. The ambiguous figure illusion illustrates this. For example, Fig. 4 can appear as a grinning man or as a cherub smoking a cigarette. The figure may be seen one way or the other, possibly in alternation, but not both ways at once. If recognition could proceed without grouping, then the figure could be recognized both ways at once even though only one way was grouped. Fig. 5 gives a more pertinent illustration. Both parts of the figure contain a numeral 4, but one is hard to recognize because it is not an object of our perception. The parts of the hidden 4 are distributed among three more primitive objects. The other numeral 4 is easily recognized despite a great deal of visual noise because it is an object of our perception. Both of these examples demonstrate that grouping is more primitive and more fundamental than recognition.

The determination of relational structure in the visual field is also due in part to grouping. For example, the leftmost target in a row of targets can be "seen" (that is, distinguished as leftmost) if and only if the row itself is an object of perception. Experiments with birds [11] have shown that grouping is essential for this type of perception.

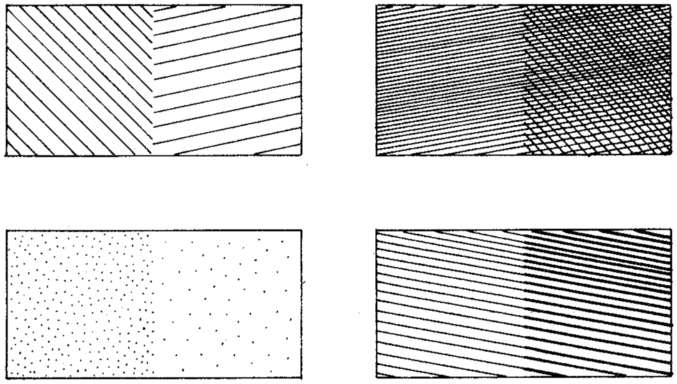


Fig. 3. Illustration of textural grouping.

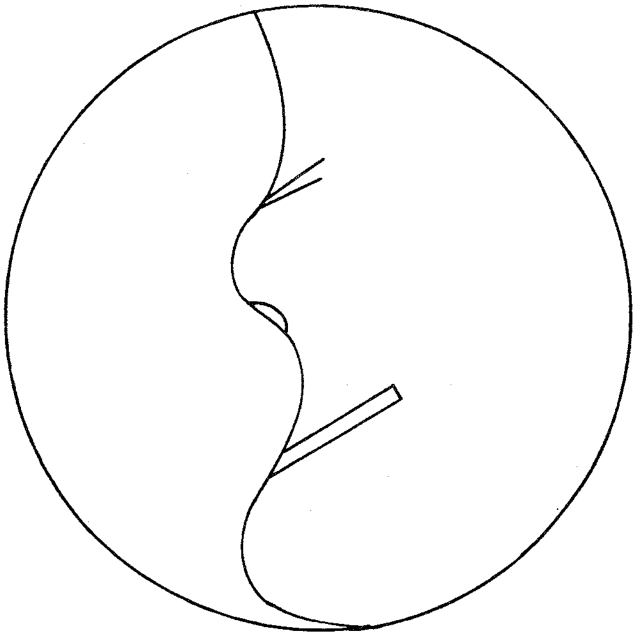


Fig. 4. Illustration of interplay between grouping and recognition.

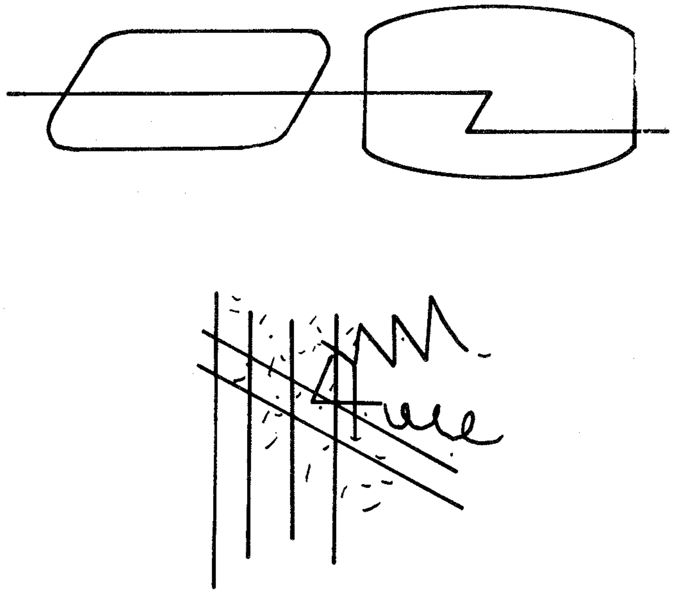


Fig. 5. Illustration that grouping is more primitive than recognition.

One complicating factor is that experience can affect grouping. In some sense, grouping is due to perceptual and cognitive factors, and the two are difficult to separate. Psychologists generally agree, however, that grouping is mainly a perceptual phenomenon which helps to organize the visual field. For example, Hebb [12] postulates a "primitive process" of grouping prior to the operation of his perceptual assemblies.

Though psychologists have an excellent qualitative view of perceptual grouping and its importance, attempts to deal with it in a quantitative fashion have been less successful. The basic problems are two-fold. First, it is hard to say precisely what causes grouping in a given scene aside from vague reports such as "the shapes are similar," or "the textures are the same." Second, grouping appears to be due to a multiplicity of cues which are hard to separate. Thus, experimentally, it has been difficult to give operational definitions to the factors of grouping. At the same time, it has been difficult to put forth any meaningful theory of grouping.

To illustrate these difficulties, consider Fig. 6. The Gestalt psychologists have postulated that similarly shaped objects are grouped [13]. Thus, three groups can be seen. Now suppose that in an experimental situation, the subject is instructed to report which of the adjacent pairs seem more like a single textural field. According to Gestaltists, the groups which tend to coalesce are those which have the most similarly shaped elements. But a useful theory must relate observables. An attempt by Beck [14] to give an operational definition of shape similarity yielded a surprising result. Though most people would segregate the slanted *T*'s from the upright *T*'s and angles, most would also report that on an individual basis, the upright *T* is more similar to the slanted *T* than it is to the angle. In general, Beck found that inspection of the individual figures did not consistently predict grouping of

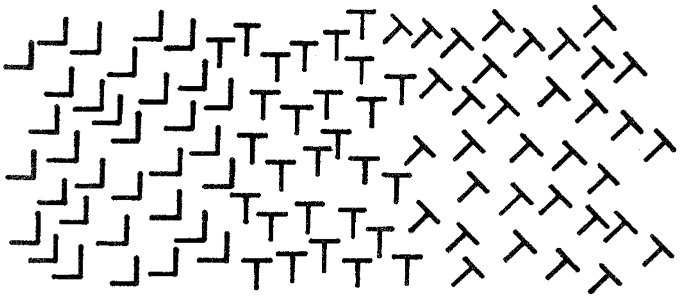


Fig. 6. Type of scene used in the Beck experiment.

arrays of those figures. Thus, it appears difficult to arrive at an operational definition of similarity (or similarity-for-grouping) which would admit the simple Gestalt theory.

The only alternative is to introduce new grouping principles such as "orientation" to explain the result of Beck's experiment. But each new factor moves us a step away from the clean theory proposed by the Gestaltists. It is in this sort of situation that computer modeling offers the best hope of giving an understandable explanation of mental phenomena. Instead of trying to explain perceptual grouping in terms of the fewest possible factors, one can assume an indeterminate number of factors but still have a simple model because the many factors can be organized into information processing routines which are easy to understand. Although the primary purpose of our work is to achieve by simulation human performance at a perceptual task, we also feel that our program is the embodiment of a simple information processing model of perceptual grouping.

III. BASIC ASSUMPTIONS AND METHODOLOGY

The following assumptions are made. First,. an indefinite number of elementary cues for grouping can be discovered and can easily be simulated on the computer. To make these commensurable, each may be regarded as a nonnegative distance function which gives smaller values to pairs of areas which are likely to be grouped and larger values to those that are not. The second assumption is that the total tendency to group is a positive linear combination of the elementary distance functions. Letting d_i, $i = 1,\cdots,n$ be the elementary distance functions

$$D = c_1d_1 + \cdots + c_nd_n$$

will be the distance function that measures the total tendency to group.

Considered in the light of human perception, these assumptions are rather natural ones to make. The first assumption eliminates the need to seek a single principle to explain all grouping phenomena. Dozens of cases similar to Fig. 3 can be presented, each possessing its own peculiar reason for the way it is perceived. The ultimate model of perceptual grouping may need dozens or hundreds of "principles" to explain the full range of cases. That model will be understandable if and only if there is an orderly methodology for exploring each of its parts despite the presence of the others. This requires that the interactions between the elementary grouping principles be of a relatively simple type. This simplicity is achieved here by taking a linear sum of the grouping principles. The coefficients c_i allow any particular cue to be more or less potent in the total sum. If the d_i are properly scaled, then the c_i are measures of the relative strengths of the cues. These assumptions constitute an information processing model of an aspect of human vision. The computer programs which will be described are embodiments of that model, and running them gives a test of the model.

The next task is to lay out an appropriate methodology for the development of the distance function D. The goal is a difficult one. The function D must yield a close simulation of an extremely complex and subtle visual phenomenon. This implies the need for a cycle of improvement and testing where the results of testing give data for further improvement. A single cycle of our program development can be outlined by the following steps.

Step 1: The programmer implements new elementary distance functions d_i or modifies existing ones. To the programmer, each d_i is a subroutine which measures differences between regions of an image. This modularity is well suited for corrective changes and additions indicated by Step 4.

Step 2: Experimental textures are arranged to form a set of psychophysical experiments. After a complete set is prepared, the experiments are conducted interactively, and while the human subject is responding, the machine records its response according to the distance function D.

Step 3: After a set of experiments is completed, the machine adjusts the coefficients c_i to obtain maximal agreement with the human subjects over all sets of experiments accumulated.

Step 4: Those experiments which still yield disagreement are used as guidance for the next cycle of development (Step 1). The key is to notice possible textural cues in the cases which disagree.

The details of each of these steps will eventually be made clear. The goal of this methodology is to converge rapidly to a good simulation of human perceptual grouping despite the dependence in Step 1 upon a programmer of unknown ability. This is a good accommodation to make, since the programming will probably fall into the "black art" category where all programmers should be considered to have unknown ability.

Since the computer is already being used as a simulation tool, the second step affords a spectacular opportunity to put the power of the computer to another use. The design, creation, and administration of effective psychophysical tests can be a difficult and time consuming job. More and more, psychologists are finding that the computer can be used as a powerful tool which enables them to perform experiments which would be too tedious without the computer's help. As a famous example, the dot patterns used in Julesz's experiments on binocular disparity [15] and texture perception [16] were created by computer. Here, if the development cycle is to be rapid, then some sort of computer-assisted psychophysical experimentation seems essential.

Since the distance function is a sum of terms, the ideal experiment would weigh the effect of one term against another. Hopefully, sets of experiments would test the terms in varying proportions to see if a single distance function D performed well in all cases. Fortunately, the previously mentioned Beck experiment can be used as an extremely delicate test of the tendency of one pair of areas to group against the tendency of another pair of areas to group. Although the Beck experiment was originally performed with hand-drafted textures, it can easily be modified so that the computer does almost all of the work involved in a rapid fashion.

IV. ARTIFICIAL TEXTURES

An initial experimental system was built for the purpose of trying out the methodology. The hardware components include an IBM 360/44 computer and an Adage AGT-10 computer and graphics display. Through the use of a link, the usual graphics functions are executable by Fortran programs running on the IBM machine. Fig. 7 outlines the flow of information between the basic software components. Only parts of the system are used at any one time, and it is useful to think of three modes of operation with executive control lodged in one of the boxes labeled A, B, or C.

In its first mode of operation, the system interacts with an experimenter to create a set of Beck experiments. Displays for two such experiments are shown in Fig. 8. The system first displays a menu of stick figures and the experimenter chooses three figures with a light pen. Using a set of buttons, the experimenter can rotate, shrink, or enlarge the figures. Another button push causes the figures to be repeated in a preset pattern to form three rectangular areas. A final button push stores the experiment in internal numerical form. It is also possible to invoke a pseudo-random number generator which generates random experiments from the menu. The stick figures are input to the system in numerical form by means of punched cards. Subject only to the limitation of the visual medium (a cathode-ray tube capable of displaying several hundred line segments) the experimenter can create dozens of experiments in a matter of minutes.

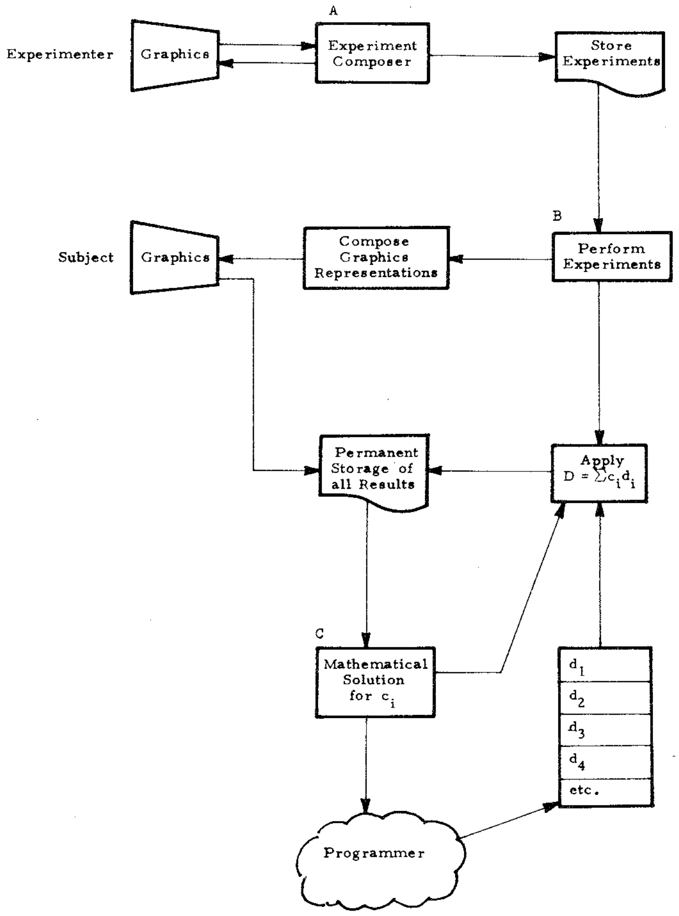


Fig. 7. System structure and flow of information.

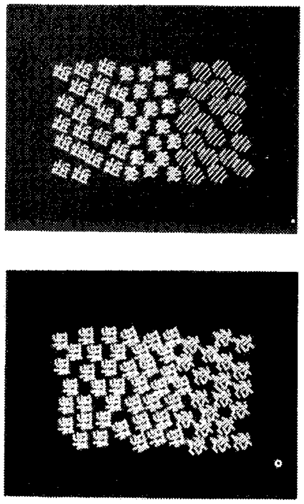


Fig. 8. Two sample displays for experiments using artificial textures.

Once a set of Beck experiments is stored in the system, the second mode of operation can be entered. A human subject is set in front of the graphics display and given a simple set of instructions. The subject interacts with the system by means of three buttons. He is told to press the center button to cause the scenes to be displayed in sequence. For each scene he is to press the right button to report the perception of more prominent division on the right and the left button for more prominent division on the left. It takes about two seconds for a scene to appear, and the subject usually responds within five seconds, so a set consisting of dozens of Beck experiments can be performed in a few minutes. The results are automatically stored in the computer. While the subject is thinking, the system calculates its response according to the distance function D, and the results of that calculation are also stored.

The means of solving for the coefficients c_i of the distance function will be taken up later. That solution process is designed to give help to the programmer (appropriately represented by a fuzzy cloud in Fig. 7) as he adds to or modifies the elementary distance functions d_i. The elementary distance functions are Fortran subroutines written by the programmer and compiled into the system. Appendix A of this report gives a description of the d_i developed for the artificial texture system.

Only two sets of experiments were performed with this system. The first set consisted of twelve Beck experiments which were produced interactively by one of the authors. A graduate student unconnected with the project served as the first subject. The initial distance function D used the seven elementary distance functions described in Appendix A. After solving for the coefficients c_i we obtained the surprising result that six of the c_i were zero. The only nonzero term was one which correlated the orientations of lines in the figures. In other words, areas with fine edges distributed over the same angles are grouped. The distance function agreed with the human subject on all of the twelve cases using only this term, a rather interesting result in itself.

The second set of experiments consisted of 47 randomly generated scenes. Since some of the scenes were ambiguous, three human subjects were used, and scenes which caused disagreement were discarded due to a constraint imposed by our method of solving for the coefficients. This left 32 scenes which were clearly grouped to one side or the other. After solving for a new set of coefficients c_i the results were poorer, but not poor enough to discourage us. These results are summarized and discussed in Section VII.

These two experiments convinced us that the methodology was workable and that an interesting degree of simulation of perceptual grouping was possible, so we decided to concentrate immediately on a system which would operate on naturally occurring image textures. There were several particular reasons for not continuing with the artificial texture system. The major reason was that a line drawing graphics display is too crude for phychophysical experimentation with texture. Only a certain number of lines could be drawn without flicker. Lines of different length had different brightnesses. Curves could not be drawn at all, and an approximation to a curve using short lines would be extremely bright. Second, the computer representation of the textures as line segments given by the coordinates of their end points was quite unsatisfactory so far as computation was concerned. We decided not to wrestle with these problems until the new and more interesting system was built.

V. NATURAL TEXTURES

The introduction of naturally occurring textures gives a tremendous increase to the dimensionality of the problem. Pickett [17] considers the varieties of textures to be like independent dimensions in an infinite-dimensional space. Practical scene analysis systems must be able to deal with this dimensionality, but in some orderly fashion starting with the most important dimensions. If a distance function D can be successfully implemented for natural textures, then some knowledge about this dimensionality will be gained.

The natural texture system is identical in principle to the previously described system. The graphics hardware is replaced by a Spectrovision SG-D 2219 color video display which is connected to a Hewlett–Packard 2100 computer. To obtain digitized images, the Hewlett–Packard computer is connected to an image digitizer which converts 35 mm photographic slides into 256 × 256 arrays with a 64 unit gray scale. The results are stored on magnetic tape. The Hewlett–Packard computer is connected to a teletypewriter adjacent to the Spectrovision display, so experiments can be composed or taken interactively as before. Fig. 9 shows several Beck experiments as they appear to a subject. The lower experiment is most interesting since it weighs a fine textural similarity on the left against a brick lattice textural similarity on the right. As the experiments are composed, they are also stored on magnetic tape, and the experiments are given to a subject by replaying the magnetic tape.

After a set of experiments is performed on the Aerojet and Hewlett–Packard equipment, the results are hand carried on magnetic tape to the IBM 360/44 computer for calculation of the distance function D and the coefficients c_i. The elementary distance functions d_i are modular routines on the 360/44 as before (see Appendix B). The results for this system are presented in Section VII.

This work has given one especially interesting result. Many of the "standard" textural measures used by other workers were coded as elementary distance functions here. Each one alone could handle a few specific types of cases, but could do little better than random over the range of cases we dealt with. A sum of such terms gave good results overall. In fact, Section VI will show that a sum of terms (as given by several nonzero coefficients c_i) is guaranteed to be superior to the use of any single term. Thus, the general ideas presented in this report would seem to be applicable to a variety of efforts in the area of texture analysis.

VI. RELATIVE STRENGTH OF CUES

Assuming that the d_i are scaled to roughly the same range, then the value of the coefficient c_i of a term is an indication of how important that term is in the calculation of grouping. Even if the d_i are not scaled, the values of the coefficients still determine the potency of the terms in the total sum. Recalling that Beck's original result was somewhat counterintuitive, one would expect that determining the values of these coefficients is a subtle and difficult task. As it turns out, there are mathematical methods for calculating optimal coefficients as experimentation proceeds.

Assume that a single Beck experiment is presented to the distance function routines and suppose that there are n elementary distance functions. Let d_{i1} be the ith measured distance between the two rectangular areas on the left and d_{i2} be the ith measured distance between the two rectangular areas on the right. If the human subject reports division on the left, then in order for the computer to agree

$$c_1(d_{11} - d_{12}) + \cdots + c_n(d_{n1} - d_{n2}) > 0.$$

If the human subject reports division on the right, then the inequality is reversed. The inequalities can always be regarded in the same sense if the proper sign is appended to the differences $(d_{i1} - d_{i2})$. Denote this ith signed difference by e_i.

Suppose now that m experiments are performed where m is larger than n. The result is m linear inequalities in n unknowns. Let e_{ij} be the ith signed difference for the jth experiment. Subtracting an artificial variable x_j from each equation converts it to an equality and the following equations can be written.

$$\begin{gathered} c_1e_{11} + \cdots + c_ne_{n1} - x_1 - y = b \\ \cdot \\ \cdot \\ \cdot \\ c_1e_{1m} + \cdots + c_ne_{nm} - x_m - y = b \\ c_1 + \cdots + c_n = 1 \\ c_i \geq 0 \qquad i = 1,\cdots,n \\ x_j \geq 0 \qquad j = 1,\cdots,m \\ y \geq 0 \end{gathered}$$

where b is an arbitrary positive constant. These equations are in standard form for solution by linear programming. The simplex algorithm can first be used to find whether a solution exists. If a solution exists, then there is usually a space of different solutions. Then setting

$$g_{\text{opt}} = y,$$

simplex can be used to find a solution in the space of solutions which maximizes the value of g_{opt}. This forces the weakest inequality in the original problem to be as strong as possible. In other words, the distance function will agree with the human subjects as strongly as possible over all cases.

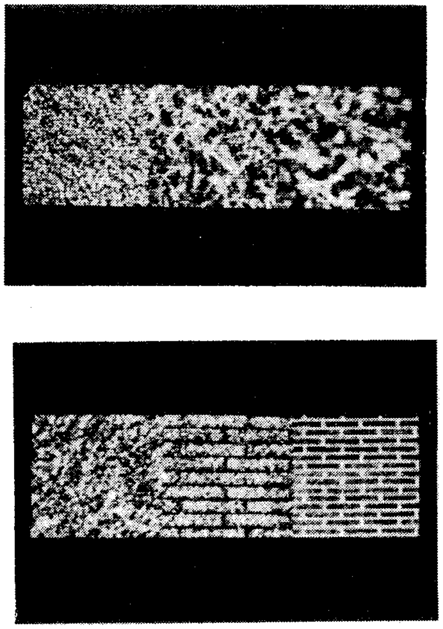


Fig. 9. Two sample displays for experiments using natural textures.

Each new Beck experiment adds a new linear inequality, so m increases as more experiments are performed. This poses no problem for the simplex algorithm since modern versions can handle literally thousands of variables and constraints. So long as a solution is maintained, the distance function achieves a perfect simulation of human perceptual grouping. A difficulty arises when simplex determines that no solution exists. To achieve as good a simulation as possible, one might try to satisfy as many of the linear inequalities as possible. This is quite compatible with our methodology since the remaining unsatisfied inequalities can be reviewed by the programmer as he modifies the programmed distance function. This can be done by showing the Beck displays to the programmer on the graphics or video display. If the programmer can detect some feature which has not been properly captured by the distance function, then he has some guidance for making the proper corrections.

Now consider the problem of satisfying as many of the linear inequalities as possible. Although mathematical techniques [18], [19] are known which are computationally feasible for small m, we have not found a practical way to find the optimal solution for large m, so the following approach is used. The inequalities shown above are modified by the addition of artificial variables $z_j \geq 0$ $j = 1,\cdots,m + 1$.

$$c_1e_{1j} + \cdots + c_ne_{nj} - x_j - y + z_j = b$$

$$c_1 + \cdots + c_n + z_{m+1} = 1$$

now the simplex algorithm is applied with

$$g_{\text{opt}} = -(z_1 + \cdots + z_m + Kz_{m+1})$$

in the hope that this will automatically force most of the z_j terms to zero. A large constant $K > m$ is included so that

z_{m+1} is guaranteed to be forced to zero. The addition of the new artificial variables guarantees that simplex will find a solution. If simplex finds a solution with $g_{\text{opt}} = -\Sigma z_j = 0$ then all of the original inequalities can be satisfied and a second application of simplex is performed with the z_j held constant and $g_{\text{opt}} = y$. Suppose that some of the artificial z_j are still nonzero. Deleting the inequalities having nonzero z_j results in a system of equations that can be solved without the use of the artificial variables z_j. However, it may be possible to delete a proper subset of the inequalities with nonzero z_j and still get this sort of solution. If the number of nonzero z_j is small, then it is relatively easy (by trying likely combinations) to find a locally optimal solution. That is, a set of inequalities with nonzero z_j which cannot be reduced to a proper subset.

Regardless of the mathematical techniques used, the following general statement can be made. The coefficients c_i are chosen which give some sort of best fit to the data, and the worst cases are given as feedback to help the programmer modify the distance function D. Although the values of the c_i are important to D, optimizing the c_i can only bring D to a level where it is limited by the quality of the programmed d_i's. We regard the mathematical solution for the c_i's not so much as an effort to obtain "good results" but more as a crucial part of our methodology. We would like to point out some of the considerations which led us to choose linear programming methods over the other available techniques. First, the degree to which a human subject perceives grouping cannot reliably be measured. If a subject initially perceives slight grouping, then that perception usually reinforces itself in the same way that a shadowy figure in a fog will take on a concrete form. Thus, the inequalities cannot easily be replaced by measures of the strength of grouping. Second, the use of a probabilistic measure of the percent of subjects which group to one side or another was not used because it was felt that, given the present state of the art, it was more important for a grouping program to be able to handle the cases which were obvious to human perception rather than cases which were ambiguous to human perception.

Usually, the simplex algorithm will set many of the c_i's to zero in the solution. This is usually an indication that the corresponding d_i's are defective (they could have the wrong sign) or wholly redundant. Once again this is useful information for the programmer. The previously mentioned mathematical methods [18], [19] could be used to force the maximal number of c_i to zero. This corresponds to the elimination of redundant cues and the number of nonzero c_i's could be called the intrinsic dimensionality of the solution. This probably should not be done on methodological grounds since one of the main goals of this work is the incorporation of redundant cues into a single mechanism. A closely related consideration is that of selecting experiments to discriminate among the d_i, which if done would tend to increase the dimensionality of the solution. These are but a few of the interesting mathematical issues that can be raised.

VII. RESULTS AND ASSESSMENT

The standard way to judge the success of a pattern recognition program is to measure its performance on test data which was not used in any way for training. Thus, the percentage of inequalities satisfied by optimizing the c_i's can only be regarded as a training score anyway. To obtain a test score, a "jack-knife" method was employed. After separating the data into equal parts, the program is trained on all of the parts but one, and then a test score is obtained on that one part. The training information is then erased and this procedure is repeated for each of the parts in turn, which gives the total test score. Ten percent jack-knife means that one-tenth of the data is tested at a time.

Table I shows the results for both grouping systems. The training score is obtained by optimizing the coefficients c_i over all of the data in the sample domain. Jack-knife testing was done with those samples missed in full set training removed. To give a fair comparison with standard work in pattern recognition, the results in parentheses count those removed samples as misses, although it should be noted that they might not have been missed in the jack-knife test.

The actual values shown in Table I are difficult to assess since there is no precedent for them in the literature. Perhaps it is most interesting to consider the results in the context of other scene analysis or robot-vision work, under the assumption that these are potential application areas. The "region growing" or "boundary melting" techniques operate by joining adjacent regions of a scene which are close in some sense. When the joining process is iterated to completion, the natural objects are supposed to be identified. A distance function which takes texture, color, and brightness into account could be used for this sort of process. For example, primitive regions which are close according to the distance function could be joined. More complex methods such as the "phagocyte heuristic" or the "weakness heuristic" (see [4]) can also be modified to use a difference of texture, color, and brightness, since they presently use a difference or brightness. For our experiments, the use of multiple cues was guaranteed to be superior to the use of a single cue, and it would be hoped that the same would be true for more complex applications.

With this in mind, the results in Table I can be appreciated somewhat. For a robot-vision system to obtain satisfactory results, the joining process has to be correct in almost 100 percent of the cases since the correct identification of objects is due to a sequence of decisions. The cues for the joining process can be divided into primitive cues such as color, texture, and brightness and "semantic" cues such as *a priori* knowledge of the shape of things. Thus, the results in Table I may be improved in any particular application which is able to make use of other types of information.

Though this paper has treated color, texture, and brightness in a symmetric fashion, there is one distinction that will eventually become important as practical systems

TABLE I
Test Results for Both Systems

Sample Domain	M	10% Jack-knife	50% Jack-knife	Training
Natural Textures:				
normalized	31	96.8%	93.5%	100%
un-normalized	38	84.6%	89.7%	100%
both	69	88.1% (85.5%)	91.0% (88.4%)	97.1%
Artificial Textures	32	85.2% (71.9%)	81.5% (68.8%)	84.4%

are built. Color and brightness can be measured at a point, but texture is a statistical property of an area of indefinite size. In a more comprehensive report on this research [20], one of the authors has developed methods for determining the minimum size over which a texture can be identified.

APPENDIX A

The textures are composed of arrays of stick figures as shown in Fig. 7. The stick figures are stored internally as a list of lines specified by the coordinates of their end points. Each of the distance functions was written to operate on a pair of stick figures, and it was assumed that such distance functions could simulate perceptual grouping of arrays of figures (even though this is against the spirit of the original Beck experiment). For a pair of stick figures, the seven elementary distance functions are as follows:

1) Distance between centers of gravity of the two figures.

2) Minimum distance between the two figures.

3) Touching distance: 0 if touching, 1000 if not touching.

4) Total length difference

$$\frac{|T_1 - T_2|}{T_1 + T_2} \times 1000$$

where T_i is the total length of lines in one of the stick figures.

5) Height-to-width ratio difference

$$\frac{|R_1 - R_2|}{R_1 + R_2} \times 1000$$

where R_i is the ratio of height to width in one of the stick figures.

6) Orientation difference: a cross-correlation between the angular distribution of lines in the figure.

7) Subtended area difference:

$$\frac{|S_1 - S_2|}{S_1 + S_2} \times 1000$$

where S_i is the area subtended by one of the figures relative to the center of gravity.

Table II shows the coefficients which resulted from training on the full set of data. Since all of the distance functions were scaled between 0 and 1000, these coefficients are an indication of which features were most crucial. Total length, which on a graphics display is actually a measure of brightness, had the highest coefficient. Orientation seems to be the most potent textural cue.

TABLE II
Coefficients Resulting From Full Set Training, Artificial Textures

	Feature	Coefficient
2.	Minimum length	.045
4.	Total length	.68
5.	Height-width ratio	.062
6.	Orientation	.13
7.	Subtended area	.083

APPENDIX B

There are a number of well-known natural image feature measures in the literature. By calculating a given measure for two regions, the absolute value of the difference can be considered to be a distance between those regions. A large number of features were computed from the spatial gray-level dependence matrices defined by the image [21]-[23], which have been used in a number of pattern classification applications [22],[24].

The spatial gray-level dependence matrix may be defined as

$$P_{ad} = P_{ad}(i,j)$$

where $P_{ad}(i,j)$ is the relative frequency of a pixel (picture element) of level i and a pixel of level j separated by distance d and an orientation a. The distance d is measured in pixels. The angle a is allowed to take on the value 0°, 45°, 90°, and 135°. By symmetry, $a = 180°$ is the same as $a = 0°$. All images were quantized to 16 levels of gray scale. Thus each P_{ad} is a 16×16 symmetric matrix.

Five textural features were computed from the gray-level dependence matrices:

$$T_1(a,d) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} i \cdot j \cdot p_{a,d}(i,j)$$

$$T_2(a,d) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} (i-j)^2 p_{a,d}(i,j)$$

$$T_3(a,d) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \frac{p_{a,d}(i,j)}{1 + (i-j)^2}$$

$$T_4(a,d) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} p_{a,d}(i,j) \log p_{a,d}(i,j)$$

$$T_5(a,d) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} |i-j| \, p_{a,d}(i,j)$$

where n is the number of gray levels.

Finally a set of measurements without explicit dependence on a was found:

$$\bar{M}_k(d) = d^{-1} \sum_{a} T_k(a,d)$$

TABLE III
COEFFICIENTS RESULTING FROM FULL SET TRAINING, NATURAL TEXTURES

Features Used	$\bar{M}_k$	R_k	V_k	D_k	
d = 1	.87				(i • j)
d = 2					
d = 1					$(i-j)^2$
d = 2	.092			.0069	
d = 1	.0026			.0030	$\frac{1}{1+(i-j)^2}$
d = 2				.018	
d = 1				.0021	p log P
d = 2					
d = 1					$\lvert i-j \rvert$
d = 2					

0.0	cross correlation
0.0	mean
0.0	variance

$$R_k(d) = \max_a T_k(a,d) - \min_a T_k(a,d)$$

$$V_k(d) = \tfrac{1}{4} \sum_a (T_k(a,d) - \bar{M}_k(d))^2.$$

Each of these functions was evaluated for $d = 1$ and $d = 2$. Thus, forty textural measurements were computed for each individual region. Each of the features was used to define a distance measure which represented the absolute value difference between the given feature evaluated for two different regions.

An additional set of measures was computed to assess gross orientation differences. These are a function of a pair of regions. Let $a = 0,1,2,3$ correspond to angles of 0°,45°,90°,135° respectively, and let d be the angular distance

$$d(a_l,a_m) = \begin{cases} 1 & \text{if } \lvert a_l - a_m \rvert = 3 \\ \lvert a_l - a_m \rvert & \text{otherwise.} \end{cases}$$

Then let

$$T_k(a,d,l) = T_k(a,d) \qquad \text{evaluated for region } l$$

$$R_k(d,l) = R_k(d) \qquad \text{evaluated for region } l$$

$$A_k(d,l,m) = d(a_l,a_m/\max_a T_k(a,d,l) = T_k(a_l,d,l),$$

$$\max_a T_k(a,d,m) = T_k(a_m,d,m)).$$

Then the gross orientation difference measures are given by

$$D_k(d,l,m) = \tfrac{1}{2}(R_k(d,l) + R_k(d,m)) * A_k(d,l,m).$$

Evaluating for $d = 1,2$ and $k = 1,\cdots,5$ gives ten more distance functions.

Three additional distance measures were used:

$$M_d(a,b) = \lvert \bar{\mu}_a - \bar{\mu}_b \rvert$$

$$V_d(a,b) = \lvert \sigma_a^2 - \sigma_b^2 \rvert$$

$$C_d(a,b) = 1/\max_{i,j} R_{ab}(i,j)$$

where

$\bar{\mu}_i$ = mean gray-level value of region i
σ_i^2 = variance of gray-level distribution of region i
R_{ab} is the cross-correlation function of regions a and b.

All told, 43 measurements were available. Table III shows the coefficients which resulted from training on the full set of data.

ACKNOWLEDGMENT

This work was made possible through the use of the Image Processing Laboratory at the University of Southern California under the directorship of Prof. W. Pratt and the University Computing Center under the directorship of R. Penne.

REFERENCES

[1] A. Guzman, "Decomposition of a visual scene into three dimensional bodies," in *Fall Joint Comput. Conf., AFIPS Conf. Proc.*, vol. 33. Washington, D. C.: Spartan, 1968, pp. 291–304.

[2] C. T. Zahn, "Graph-theoretical methods for detecting and describing Gestalt clusters," *IEEE Trans. Comput.*, vol. C-20, pp. 68–86, Jan. 1971.

[3] L. G. Roberts, "Machine perception of three-dimensional solids," in *Optical and Electro-Optical Information Processing*, J. T. Tippett *et al.*, Eds. Cambridge, Mass.: M.I.T. Press, 1965, pp. 159–197.

[4] C. R. Brice and C. L. Fennema, "Scene analysis using regions," *Artificial Intelligence*, vol. 1, pp. 205–226, 1970.

[5] D. L. Waltz, "Generating semantic description from drawing of scenes with shadows," Massachusetts Inst. Technol., Cambridge, M.I.T. AI Memo IR-271, 1972.

[6] Y. Yakimovsky, "Scene analysis using a semantic base for a region growing," Stanford Univ., Stanford, Calif., Stanford AI Lab. Memo AIM-209, June 1973.

[7] C. A. Harlow and S. A. Eisenbeis, "The analysis of radiographic textures," *IEEE Trans. Comput.*, vol. C-22, pp. 678–689, July 1973.

[8] R. M. Pickett, "Visual analysis of texture in the detection and recognition of objects," in *Picture Processing and Psychopictorics*, B. C. Lipkin and A. Rosenfeld, Eds. New York: Academic, 1970, pp. 298–308.

[9] A. Rosenfeld and M. Thurston, "Edge and curve detection for visual scene analysis," *IEEE Trans. Comput.*, vol. C-20, pp. 562–569, May 1971.

[10] W. Ellis, Ed., *A Sourcebook of Gestalt Psychology*. London, England: Routledge and Kegan Paul, 1938.

[11] M. Hertz, "Figural perception in the jaybird," in *A Sourcebook of Gestalt Psychology*, W. Ellis, Ed. London, England: Routledge and Kegan Paul, 1938.

[12] D. O. Hebb, *The Organization of Behavior*. New York: Wiley, 1949.

[13] M. Wertheimer, "Laws of organization in perceptual form" in *A Sourcebook of Gestalt Psychology*, W. Ellis, Ed. London, England: Routledge and Kegan Paul, 1938.

[14] J. Beck, "Effect of orientation and shape similarity on perceptual grouping," *Perception and Psychophysics*, vol. 1, pp. 300–302, 1966.

[15] B. Julesz, "Binocular depth perception without familiarity cues," *Science*, vol. 145, pp. 356–362, 1964.

[16] ——, "Visual pattern discrimination," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 84–92, Feb. 1962.

[17] R. M. Pickett, "The perception of a visual texture," *J. Experimental Psychology*, vol. 68, pp. 13–20, 1964.

[18] J. B. Rosen, "Minimal and basic solutions to singular linear systems," *J. Soc. Ind. Appl. Math.*, vol. 12, pp. 156–162, Mar. 1964.

[19] R. E. Warmack and R. C. Gonzalez, "An algorithm for the optimal solution of linear inequalities and its application to pattern recognition," *IEEE Trans. Comput.*, vol. C-22, pp. 1065–1075, Dec. 1973.

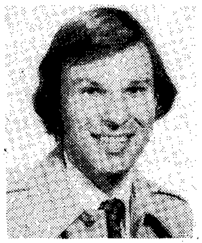
[20] W. B. Thompson, "The role of texture in computerized scene analysis," Comput. Sci. Program, Univ. Southern California, Los Angeles, Jan. 1975.

[21] A. Rosenfeld and E. Troy, "Visual texture analysis," Univ. Maryland, College Park, Tech. Rep. 70-116, June 1970.

[22] R. M. Haralick, K. Shanmugan, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, pp. 610–621, Nov. 1973.

[23] D. A. Ausherman, "Textural discrimination within digital imagery," Ph.D. dissertation, Univ. Missouri, Columbia, Dec. 1972.

[24] R. P. Kruger, W. B. Thompson, and A. F. Turner, "Computer diagnosis of pneumoconiosis," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-4, pp. 20–49, Jan. 1974.



Albert L. Zobrist (S'69–M'71) was born in Seattle, Wash., in February 1942. He received the B.S. degree in mathematics from the Massachusetts Institute of Technology, Cambridge, in 1964, and the M.S. degree in mathematics and the Ph.D. degree in computer science, both from the University of Wisconsin, Madison, in 1966 and 1970, respectively.

From 1966 to 1967 he was a member of the technical staff, Aerospace Corporation, El Segundo, Calif. Since 1970 he has served as Assistant Professor of Electrical Engineering and Computer Science at the University of Southern California, Los Angeles (presently on leave). From 1973 to 1974 he was also a member of the research staff of the University of Southern California Information Sciences Institute. He is presently Associate Professor of Computer Science at the University of Arizona, Tucson.

William B. Thompson (S'72–M'74) was born in Santa Monica, Calif., in August 1948. He received the Sc.B. degree in physics from Brown University, Providence, R.I., in 1970, and the M.S. and Ph.D. degrees in computer science from the University of Southern California, Los Angeles, in 1972 and 1975, respectively.

He is presently a Research Associate with the Image Processing Institute at the University of Southern California, Los Angeles. His interests are in the fields of artificial intelligence and data extraction from imagery, including scene and texture analysis.

Dr. Thompson is a member of Eta Kappa Nu and the Association for Computing Machinery.