MACHINE PERCEPTION AND DESCRIPTION OF PICTORIAL DATA*

Martin A. Fischler
Lockheed Palo Alto Research Laboratory
Palo Alto, California

Summary

This investigation of machine processing of pictorial data is based on the premise that people can recognize visual objects and describe them well enough so that other individuals can recognize the object from the description. Given a system of linguistic communication between a person and a digital computer, and given that the computer possesses adequate perceptual machinery, many currently refractory problems in pictorial data processing would be open to solution.

This paper describes a computer system which can perceive a limited class of graphical objects, create linguistic descriptions for the objects, and classify objects by comparison with a reference set of descriptions as might be produced in normal human communication.

Introduction

Over the past decade, most of the work in automatic pictorial pattern recognition has been based on partitioning the problem into a "feature" detection phase typically followed by a rather general decision-theoretic classification phase.[1, 2] The limited success of this approach and a fuller appreciation of the difficulties of the problem has recently caused many investigators to look elsewhere for a solution. Early papers and informal communications by Kirsch, Minsky, and Ledley suggested the use of linguistic methods for describing and processing pictures, A significant amount of work has now been done using the linguistic approach,[2"10] mostly from the standpoint of attempting to apply the tools and techniques of formal or mathematical linguistics.

It is this author's contention that the difficulty of the problem arises from the fact that whereas our mathematical tools are designed to work within a well defined structural framework or model, they offer very little guidance in helping us to obtain this model. The problem of obtaining a suitable representation for a picture is much more difficult than the problem of classifying the picture once a satisfactory representation is obtained. Thus, a study of the use of informal (linguistic) techniques for obtaining a suitable representation of a picture would appear more promising than an investigation of formal linguistic techniques for "solving" or manipulating such representations once obtained.

The overall objective of the effort to be described here is to permit a machine to replace or at least cognitively interact with a human in the generalized processing of pictorial data.

The main constituents of the approach are goal-directed perception and a descriptive hierarchy guided by and leading to a linguistic description of the pictorial object being processed.

♦This work was conducted under the Lockheed Independent Research Program.

The work presented in this paper follows in large part from the hypothesis that people can recognize visual objects and describe them well enough so that other individuals can recognize the objects from the description. If this hypothesis is valid, and if it is also possible to develop a vocabulary and a syntax which will permit similar communication between a person and a digital computer, then a whole range of currently refractory problems in machine perception and pictorial pattern recognition would be open to solution.

While the hypothesis given is reasonable, the following experiment was performed to verify the hypothesis and to illuminate the implications for man-machine communication.

An Experiment in Human Communication About Pictorial Objects

A set of four samples of each of 14 different hand-written Sanskrit characters (see Figure 1) was obtained, as were four subjects totally unfamiliar with the Sanskrit alphabet. The nature of the experiment was described to one of the subjects who was asked to prepare a written description for each character in one of the character sets (14 distinct items). These written descriptions were then used by the other three subjects to identify the characters in the three remaining (but randomly intermixed) character sets. Each subject, in turn, was given the description set for reference and then shown the characters one at a time. He was allowed all the time he wished in making each classification decision, but he could not ask any questions, nor change a decision once made, nor see any character other than the one he was currently working on. Figure 2 shows some of the descriptions which were generated during the experiment.

The outcome of the experiment can be summarized as follows: 11 of the descriptions produced were judged to be valid and unambiguous. The corresponding 33 characters in the test set were correctly classified by each of the subjects. Two of the descriptions were ambiguous. This situation was detected by all three subjects after viewing some portion of the test character set. The ambiguously described test characters were then lumped together as one class. One description contained an incorrect phrase. After seeing one or two occurrences of the incorrectly described character, the incorrect part of the description was discarded by all subjects and the character was correctly classified in its final occurrences. Some extraneous Sanskrit characters got mixed in with the test characters; these were discovered by all three subjects and rejected. Analysis of this experiment (as well as a number of other similar experiments which will not be discussed here) allowed the following conclusions to be drawn:

(l) The written descriptions passed on only a fraction of the information necessary to achieve the observed performance. The residual information was already possessed by the subjects in their "model-of-the-world." Thus, linguistic communication seems to operate by offering or suggesting modifications to an existing complex structure. If a machine is to be capable of interacting with a human in anything approaching a natural linguistic environment, the machine must be endowed with a fairly sophisticated model of the universe of discourse, and perhaps even a model of the human communication process itself; e.g., people frequently make erroneous as well as ambiguous or incomplete statements when attempting to transmit information.

(II) A linguistic description can only be a partial representation of the object or scene being described. The aspect of the object abstracted by the description will typically be guided by the type of questions to be asked of the description. Thus, in the preceding experiment, the descriptions were well suited to the task of classification, but it is obvious from Figures 1 and 2 that these descriptions are totally inadequate for the purpose of reconstructing the objects they describe. The implication here is that a description does not follow naturally from the object being described, but is actually an "artistic" creation, tailored to a specific need (or set of needs).

(III) Our original hypothesis has been verified. Surprisingly high classification performance can be achieved with very simple and seemingly crude descriptions. However, this efficiency of communication can only be accomplished when the communicants share a complex model of the universe of discourse and have similar visual and perceptual abilities.

Machine Perception

If a machine is to be capable of carrying on a conversation about pictorial data, it must not only be capable of perceiving such data, but it must also be cognizant of the manner in which humans perform this task. (See the discussion terminating the preceding section.) Let us now see what this implies.

We first note that there is a sharp distinction between "seeing" and "perceiving." "Seeing" is the passive reception of visual data; we might say that a T, V. camera "sees" the object it is focused on. " Perceiving" is an active creative process in which a visual scene is decomposed into meaningful units. This decomposition is a function of the stimulus pattern, the vocabulary and experience of the observer, and the "psychological set" of the observer.

In view of the complexities inherent in the perceptual process, a decision was made to initially restrict the universe of discourse for this investigation to line-type drawings. Thus, we would like the machine to be able to talk about lines, the way they are shaped, and their geometric relationships in a given scene.

The following paragraphs will sketch some of the details and theoretical considerations in designing a machine which can first see a graphical line-type object, and then perceive it in a goal-directed manner.

From a theoretical standpoint, we wish to generate a series of successively more structured representations which start with the raw input data and culminate in a representation of the object as a configuration of line segments with appropriate descriptive tags.

Level Ia: The Object

The lowest level (Ia) representation is the graphical object itself. In this investigation, this is a clear 35-mm film frame on which a line drawing is produced using a nylon-tipped pen.

Level Ib; The Sampled Contour

The next level (Ib) of representation is a set of points, critically selected from Ia, which retain sufficient information to permit subsequent processing without further recourse to level Ia (except in a very small percentage of cases). The Ib representation is obtained from Ia through the use of a computer-controlled flying-spot scanner. The scanner first operates in a search mode in an attempt to find an object (i.e., locate a significant black area) somewhere in the picture frame. When an object is found, the scanner enters a contour-following mode and determines if the object has a sufficiently long exterior contour to be considered significant. If the object is not deemed significant, the search mode is reentered and the search continues essentially where it left off. If the object is deemed significant, then 50 to 150 points are selected from its exterior contour. At present, the assumption is made that all objects are "solid" and no attempt is made to search for interior contours.

A number of interesting problems arise in the design of a contour follower which must operate in the presence of moderate amounts of noise without losing significant shape information. If a large scanning spot is used, critical shape information can be lost and background noise can fuse to produce spurious objects or false appendages to existing objects. On the other hand, a very small scanning spot can enter and get trapped inside a solid appearing object, typically requires more scanning time, and produces a greater volume of data, much of it redundant. Regardless of what the scanning-spot size is, certain logical problems still exist. For example, the spot can enter a cavity with a narrow inlet and then not be able to find this inlet again in order to escape.

The approach employed here was to use a fine scanning spot (capable of generating a raster 256 by 256 scan lines) to retain resolution, to use a sampling technique to speed up the scan and decrease redundant information acquisition, and to use "high-level" information feedback to counter both noise and logical problems.

Figure 3a shows the "square-sided-spiral" path taken by the follower as it looks for the next point on a contour; when such a point ("hit") is found, the spiral search pattern is repeated. If, after one hit, the first leg of the spiral produces a second hit, this second hit is ignored and the spiral path is altered, as shown in Figure 3b. This will permit the follower to enter a narrow well, or explore a small cavity. The follower is expected to return to its approximate starting location after traversing some maximum path length. If this does not occur, it is assumed that the follower is trapped; the scan parameters are automatically altered, and the follower is reset to make another attempt to find the contour. A fixed sequence of parameter sets is available to the follower. Each parameter set can alter the starting point for the follower, the effective resolution of the spot, and the criteria for determining a hit. Effective spot resolution is altered by having the spot repeat a traverse one or more times with some small lateral displacement. The intensity data returned from these parallel scans are made available to the decision procedure which determines if a hit was detected.

After the contour has been successfully acquired, a smoothing routine deletes duplicate points and inserts (by interpolation) points between contour points which have too wide a separation. Each contour point is stored in the "Contour Point List" (CPL) as a pair of (X, Y) coordinates, the sequence number of the point, and the gross direction (up, down, right, left) in which the follower was moving when the hit was detected.

Level Ic; Slope and Critical Point Determination

This level of representation is obtained by augmenting the information stored for each point in the CPL comprising the Ib representation. The slope, quantized into 15° intervals, between each point and its preceding neighbor is determined (by table lookup) and stored. Points at which a slope change of 60° or more occurs are flagged as critical points. Extremal points (horizontal and vertical) of the contour are flagged. Finally, referring to contour follower slope (see previous section) rather than interpoint slope, points about which a 180° change in follower slope occur are labeled Endpoints. The Endpoint label is tentative and can be deleted or shifted at higher levels of representation. Accurate placement of Endpoints according to the standard of human performance was found to be one of the more difficult tasks in acquiring the machine-perceived image.

Level Id: Decomposition Into Line Segments and Enclosures

The Id level of representation involves the assignment of labels to each of the points in the CPL which will identify that point as being part of a particular line segment, intersection, or enclosure. There are usually a number of distinct structurally valid decompositions or parsings of any graphical object. (Figure 4 illustrates this point with a number of examples.) Therefore, the processing routines associated with this task must be capable of producing many alternate decompositions from which a final choice can be made. This is accomplished by using a number of line-finding routines which operate independently of each other. These routines, using different criteria, search the CPL for point sequences which can be considered to be line segments. The net result is a list of candidate line segments, some of which overlap or even completely cover other candidate line segments.

The basic procedure for finding a candidate line segment is to first find a "seed" which can either be an Endpoint (as obtained for the Ic representation) or a pair of points separated by less than some threshold distance and having opposite contour slopes. Starting with the seed, pairs of points are searched for which are adjacent to those points already assigned to the growing line segment, and are separated by less than the threshold distance. The line segment is judged complete when no additional point pairs satisfying these criteria can be found.

Smoothing routines, driven by higher level requirements, select a covering set of line segments from the available candidates, adjust overlaps and fill gaps to obtain unique intersection points, and make the final decision as to which segments are to be considered enclosure boundaries. Endpoints and intersection points may receive minor adjustment, based on the slope information (critical point flags) obtained for the Ic representation. Redundant endpoints are deleted.

For example, if only the major characteristics of the graphical object are of interest, short line segments will be suppressed by choosing the smallest covering set from the line segment candidates. On the other hand, let us assume that the presence of even short line segments with isolated Endpoints is significant; in this case, the smoothing routines will choose a covering set of line segments containing the largest number of line segments grown from Endpoint seeds. The smoothing routines can also emphasize or eliminate enclosures, merge or break line segments, etc.

Level Ie: Labeling, and Construction of the line Segment Directory

At present, the Ie level is the highest level of representation and constitutes the perceived image. Line segments, enclosures, and intersections are assigned distinct labels (names). A directory is constructed listing each line sequent in this final stage, its type (i.e., part of an enclosure; a line segment having zero, one, or two Endpoints; or a line segment which is actually part of an extended intersection), and the names and locations of its vertex points in the CPL.

Algorithmic procedures are employed to determine which line segments go together to form the boundary of any given enclosure, and to determine which vertex points are actually part of the same intersection. This information is entered into the line Segment Directory.

Finally, global checks are made on the perceived image to determine if it is structurally valid. Thus, for example, the number of vertices (v), line segments (s), and enclosures (e) are counted and must satisfy Eulers formula ($v - s + e = 1$); the set of line segments defining the object must include all the points of the CPL and must form a closed contour; etc. Global checks are also set when individual smoothing algorithms are not able to successfully complete their task. When a global check is set, the perceptual image is considered to be invalid and a repeat pass is made through the complete perceptual machinery, using a new parameter set to control the contour follower.

Machine Description of Graphical Data

The preceding portions of this paper have covered the steps between seeing and perception. However, it was noted that the final form of the perceived image will depend on the parsing decisions and parameter settings called for by "higher level" expectations and requirements. We further noted that a linguistic description of a graphical object can only be a partial representation of the object and thus is typically created to satisfy some specific need. Both of these considerations suggest that perception and description can be most meaningfully discussed in connection with performance of some meaningful task. An experiment requiring machine classification of handprinted Arabic numerals will form the basis for the following discussion.

A set of reference descriptions, similar in form to those shown in Figure 2, was prepared for the ten numeric characters. The machine was required to produce a description of a test character presented to it and compare this description with the reference descriptions in order to classify the object.

A number of points must be emphasized here. This experiment involving handprinted numerals is simply a vehicle for investigating machine perception and description, not an attempt to produce a character reader. The machinery used here for both perception and description is much more general than is required for this classification task alone. The questions of interest in this experiment were how knowledge of the limited world of possible shapes could be used to direct perception, and what types of linguistic representations of the object would best serve as a data base for both the linguistic matching needed in the classification process, and also as a basis for operator machine communication to improve classification performance.

The Numeric Character Recognition Program (NCRP)

This program controls the perceptual machinery described earlier as well as the line Segment Analysis and Description Program (LSADP) to be described in the next section. Its Job is to classify graphical objects according to a set of canonical descriptions (given in Figure 5) and general information about the type of variations which might occur (given in Figure 6). It may be noted that these descriptions, similar in form to those in Figure 2, permit most of the characters to be classified even if they are rotated or mirror imaged (some confusion between the 2, 5, and the 6, 9 pairs will occur under a symmetric transformation). The three principal variations which are not covered at present are the closed-top-4, the open-loop-9, and the 5 with detached top. Procedures to cover these variations will be added in the near future.

NCRP begins its analysis of a character oy calling for a perceived image which suppresses small detail, but does capture the gross topology of the character. LSADP is then called to generate a description of the character.

If the resulting description shows that the character consists of a single enclosure, it is classified as either a "0" or an "8" depending on the number of lobes it has with respect to its diameter. *

If the character consists of two enclosures, with or without a connecting line between them, it is called an "8."

If the character consists of a single nonspiral line, then it is classified as a 1, 2, 3, 5, or 7 according to the criteria given for class ii objects in Figure 5. A spiral is classified as 6 (open loop). If the End points of a character tentatively classified as a "7" are very close together, the character will be reclassified as a "0."

If the character consists of a single enclosure and a single long line, it is classified as a "6" or "9" according to the criteria given for class III characters in Figure 5. However, if the single line segment was short, the perceptual and descriptive machinery is directed to reexamine the character after suppressing this line. The resulting image will then be classified as a "0" or "8," or possibly rejected.

If the character consists of three or more line segments (but no enclosures), its description is examined to determine if it can be classified as a "4." If this is not possible, then it is concluded that the character contains redundant strokes. The perceptual and descriptive machinery is directed to reexamine the character after suppressing all short length (or if necessary, even medium) line segments in order to find a valid classification.

If the character consists of a single enclosure and two line segments, the distance between the End points of these line segments is examined to determine whether they are sufficiently close together to constitute an improperly formed enclosure in the second loop of an "8." If the character cannot be classified as an 8, the perceptual and descriptive machinery is directed to reexamine the character suppressing any component (enclosure or line segment) which is small, or if necessary, medium in size. The result here will probably be a classification as a 2, 3, 6, 9, or reject.

If the character consists of some other combination of line segments and enclosures than those discussed so far, the perceptual and descriptive machinery is iteratively called to suppress or delete minor structures until the character is either classified or rejected.

The Numerical Character Recognition Program, as distinct from the perceptual and descriptive machinery which it directs, is a relatively simple program which took less than one man-week to write and check out. The 10 numeric canonical descriptions are its "model-of-the-world" and it tries to perceive any object it encounters as one of these entities. The information about possible variations is mostly used to help shape perception; however, for a few specific variations (at present) the final identity is deduced from an imperfectly perceived image.

The numeric descriptions and list of variations was obtained after a few design iterations using the linguistic output generated by the machine. This linguistic feedback was used to correct perceptual problems in the machine as well as shortcomings in the original numeric descriptions.

The current version of NCRP was tested on a set of 100 handprinted numerals generated by five different contributors. With the exception of one closed-top-4 and two open-loop-9 for which no descriptions were available to the machine, all the remaining characters were correctly classified. Many of the correctly classified characters were additionally presented to the machine in rotated or mirror imaged (the film frame flipped over) position, and with the exception of the 2, 5, and 6, 9 confusion, no additional errors resulted.

This experiment was not intendeu to assess NCRP's performance as a practical number classifier; the test set was too small, and the ink and film environment probably lead to more carefully generated characters than would result in a more natural situation. What the experiment did demonstrate was that a model based on a small number of "natural" English statements could permit classification of wide variety of graphic shapes by directing the machine to <u>perceive</u> these shapes in a canonical manner.

The line Analysis and Description Program

The perceived image, as currently constituted, is a configuration of line segments. Each line segment defined by a sequence of contour points in the CPL.

*The diameter of an object is the longest straight line which begins and terminates on the object. A lobe is a relative maximum in the normal distance from some line or axis.

Given the name of a line segment, the Line Segment Analysis and Description Program (LSADP) references the Line Segment Directory to find the location of the corresponding point sequence in the CPL. A series of analytic geometry routines are now called in which make measurements on both the named line segment and on the graphical object as a whole. In particular, the number of contour points assigned to the line segment is taken as a measure of its arc length, while the euclidean distance between its vertex points measures its extent. Shape information is obtained by constructing a straight line through the vertex points of the line segment and then computing the normal distance between each contour point assigned to the line segment and this constructed line. The resulting sequence of directed deviations is then processed to determine the number of maxima and the magnitude of the largest positive and negative deviations, the number of times the constructed line is crossed by actual line, and the relative positions of these crossings. Additional shape information is available from the interpoint slope data computed for the lc representation. Shape information is used to determine if the line segment is straight, slightly curved, or curved. If curved, the derived numerical data permit determination as to whether the curve is basically "C" shaped, "S" shaped, spiral or partially spiral shaped, or has some more complex configuration. Orientation is measured by the slope of the constructed line and the local slope of the actual line at its vertex points. Location of the line segment is determined by the location of its vertex points and the location of the line segments center of gravity. All descriptive evaluations are normalized relative to the overall size of the graphical object; all location references are with respect to the minimal rectangle enclosing the object.

LSADP is controlled by a decision tree which directs the analysis in accordance with intermediate findings. As the various attributes of the line segment are derived, these are stored as records in the "Object Description Table (ODT)," and a linguistic description is simultaneously constructed.

The linguistic description generated in the LSADP has some interesting characteristics. There are currently some 500 words and phrases which form the generator's vocabulary. As the line segment analysis proceeds, the linguistic phrases are assembled in a buffer area. Numeric values are converted into appropriate descriptive terms by means of translation tables. The specific table employed to translate a numeric quantity can vary depending upon the preceding analysis. Thus, for example, a line segment which forms the boundary of an enclosure must have approximately twice the arc length needed by a straight line to be given the attribute value "long." If there is too much ambiguity associated with a measurement, the corresponding descriptive phrase may be completely deleted, and under some conditions, a phrase previously entered into the buffer area may also be deleted.

The organization, phrasing, and information provided in the linguistic descriptions generated by USADP are based on experiments in which a subject attempted to reconstruct simple unseen line drawings based on the answers to questions he was permitted to ask.

It should be noted that LSADP does not correlate the descriptions it produces for each of the line segments of a given object. Thus, the collection of such descriptions must be modified to present an overall object description. A program which accomplished this is currently being prepared. This program, in addition to organizing the data generated by LSADP, must also change some of the assigned attributes. Thus, while LSADP might say that two slightly separated lines which leave the same vertex both have a vertical orientation, a program (or person) describing the situation would say that "one of the lines slopes slightly to the left of the vertical, the other line slopes slightly to the right." Similarly, if two line segments in the same object (especially straight lines) have almost identical lengths, then they should be given the same length attribute, even if some absolute criterion in LSADP attempts to assign them differently.

Figure 7 shows some perceived objects and the linguistic descriptions generated by LSADP for the corresponding line segments.

Concluding Comments

This paper sketches a research effort concerned with the design of a system capable of perceiving pictorial and graphical data, building a series of internal representations of these data, and then using these internal representations as the basis for automatic processing and "natural" or linguistic man-machine communication.

A detailed description of the specific algorithms and data structures employed is not presented in this paper. The design philosophy, as detailed previously, is summarized as follows:

(1) To the greatest extent possible, local decisions are avoided. Typically, a perceptual hypothesis is synthesized from as large a body of data as practical (the data are obtained from lower levels of representation) and then the "reasonableness" of this hypothesis is checked, either immediately or sometimes after a number of intervening steps, against the "model-of-the-world" that the system possesses. If the perceptual hypothesis is found to be untenable, then the system returns to some previous level of representation, modifies parameters and/or procedural steps, and once again attempts to construct a viable hypothesis. Typically, the successive attempts at hypothesis formation become more conservative; that is, loss of "resolution" is permitted after repeated failure.

(2) When a perceptual hypothesis fails, it is usually not necessary to determine the cause of the failure. The simplest way to handle a failure is to take a second look at the object from a slightly different orientation (e.g., starting the contour at a different location) with no change in the processing. If the processing step at which the failure occurred can be determined, then an alternate procedure might be used, or the original procedure might be repeated with a new set of parameters (chosen from a small selection of such sets), or the troublesome processing step might be omitted entirely. This is frequently possible with no final loss, or only minor loss, of detail. When the step or section in which the trouble occurred cannot be pinpointed, a return to one of the lowest levels of representation is made together with a fixed sequence of parameter and processing step changes.

(3) As implied, the complex* algorithms are not expected to be 100% effective. Typically, the more complex the algorithm, the more its function is overlapped by the preceding and following processing steps. Also, many of the complex algorithms are backed up by alternate procedures which can do essentially the same job.

(4) Perception, at each level of representation, is "directed" by higher level expectations. Thus, even though there are many structurally valid parsings or decompositions of the object at each level of representation, the particular parsing chosen is one which biases the representation toward some desired final canonical form. This procedure has two advantages. First, knowing what you are looking for and where to look is an effective way of combating noise. Second, the combinatorial growth of possible final representations is greatly restricted.

(5) The overall approach is procedure oriented rather than retrieval oriented. A gross change in representation (e.g., a change after detection of an error) is accomplished by altering some parameters or processing sequences, and then regenerating the representation starting at some lower level. This is preference to flagging or deleting the old data and adding the required change. Further, wherever practical, data are recomputed rather than stored and retrieved. The reason for both of the above techniques is to avoid the necessity of tracing through long chains of derived or deduced conclusions (stored as part of the representation) after a low-level error is detected.

Acknowledgments

The author acknowledges the many contributions made by O. Firschein to the investigation presented here. The work was performed under the Lockheed Independent Research Program. The support and encouragement of Dr. C. E. Duncan are greatly appreciated.

Cited References

References

1. M. Fischler, R. Mattson, O. Firschein, L. Healy, "An Approach to General Pattern Recognition," IRE Trans. on Information Theory. Vol. IT-8, Sep 1962

2. M. Minsky, "Steps Toward Artificial Intelligence," Proc. IRE. Vol. 49, Jan., 1961, pp. 8-30

3. R. Kirsch, "Computer Interpretation of English Text and Picture Patterns," IEEE Trans. on Electronic Computers, Vol. EC-13, No. 4, Aug 1964, pp. 363-376

*Those procedures which involve a large amount of data-dependent manipulations, and where both input and output to the algorithm are large data sets.

4. R. Ledley, L. Rotolo, T. Golab, J. Jacobsen, M. Ginsberg, and J. Wilson, "FIDAC: Film Input to Digital Automatic Computer and Associated Syntax-Directed Pattern Recognition Programming System," Optical and Electro-Optical Information Processing, Tippett. etal.. eds.. Chap. 33. MIT Press, 1965

5. R. Narasimhan, "Syntax-Directed Interpretation of Classes of Pictures," Comm. ACM, Vol. 9, No. 3, Mar 1966, pp. 166-173

6. T. Evans, "A Heuristic Program to Solve Geometric-Analogy Problems," AFIPS (SJCC) Proceedings, Vol. 25, 1964, pp. 327-338

7. A. Guzman, "Scene Analysis Using the Concept of a Model," AFCRL-67-0133, 1967

8. W. Miller and A. Shaw, "Linguistic Methods in Picture Processing, AFIPS (FJCC) Proceedings, Vol. 33, 1968, pp. 279-290

9. D. Londe, R. Simmons, "NAMER: A Pattern-Recognition System for Generating Sentences About Relations Between Line Drawings," ACM 20th National Conference, 1965, pp. 162-175

10. L. Lipkin, W. Watt, R. Kirsch, "The Analysis, Synthesis and Description of Biological Images," Ann. N.Y. Acad. Sci., Vol. 128, Jan 1966, pp. 984- 1012

Uncited References

R. Gregory, Eye and Brain, McGraw-Hill, 1966

G. Forsen, "Processing Visual Data With an Automaton Eye," Pictorial Pattern Recognition, Cheng, etal., eds., Thompson Book Co., 1968

B. Raphael, "SIR: A Computer Program for Semantic Information Retrieval," MIT Project MAC Report MAC-TR-2, Jun 1964

L. Coles, "Syntax Directed Interpretation of Natural Language," Ph.D. Thesis, Carnegie Institute of Technology, 1967

A. Newell, "Some Problems of Basic Organization in Problem Solving Programs," Self-Organizing Systems - 1962, Yovits, etal., eds., Spartan Books, 1962

T. Marill, A. Hartley, T. Evans, B. Bloom, D. Park, T. Hart and D. Dailey, "CYCLOPS-1: A Second-Generation Recognition System," AFIPS (FJCC) Proc. Vol. 24, 1963, pp. 27-33

J. Munson, R. Duda, P. Hart, "Experiments in the Recognition of Hand-Printed Text," AFIPS (FJCC) Proceedings, Vol. 33, 1968

R. Grimsdale et al., "A System for the Automatic Recognition of Patterns," Proc. Inst. Elect. Eng., Vol. 106 (Part B), 1959, pp. 201-211

L. Roberts, "Machine Perception of Three Dimensional Solids," MIT Lincoln Lab. Tech., Rept. 315, May 1963

E. Greanias et al., "The Recognition of Handwritten Numerals by Contour Analysis," IBM J. of Research and Development, Vol. 7, No. 1, Jan 1963, pp. 14- 20

H. Freeman, "On the Encoding of Arbitrary Geometric Configurations," IRE Transactions on Electronic Computers, Vol. EC-10, No. 2, Jun 1961, pp. 260-268

S. Amarel, "On the Mechanization of Creative Processes," IEEE Spectrum, Apr 1961, pp. 112-114

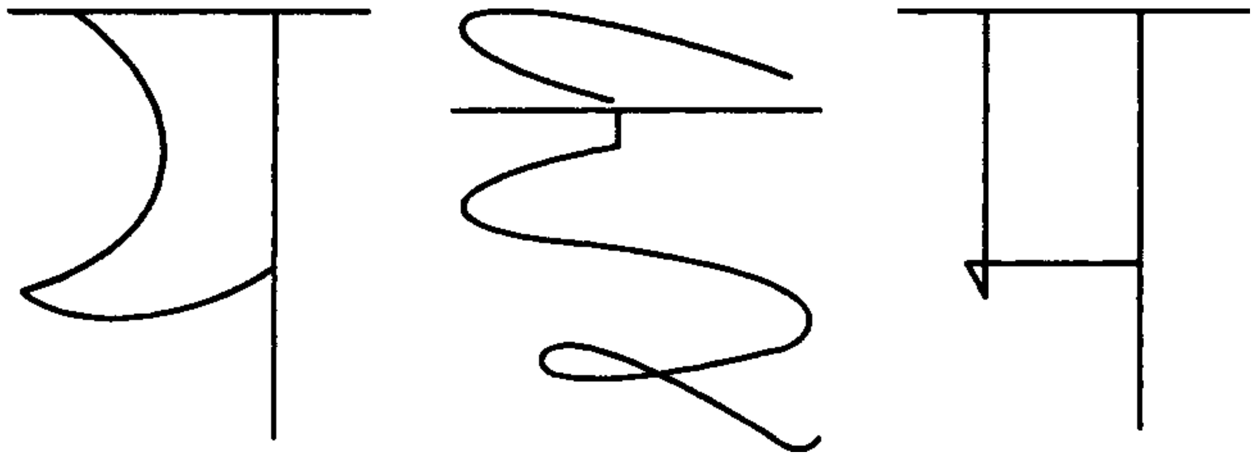


Figure 1 Sanskrit Characters

NOT FLAT TOP? / CURVED LINE AS TOP?

FLAT TOP? / LONG VERTICAL? / SORT OF A BACKWARD "P," CURVEY, EXCEPT FOR FLAT TOP?

FLAT TOP? / LONG VERTICAL? / LIKE BACKWARD "P" WITH STRAIGHT SIDES AND FLAT TOP?

FLAT TOP? / NO LONG VERTICAL? / VERTICAL ON RIGHT SIDE EXTENDS ABOUT HALF WAY DOWN?

FLAT TOP? / NO LONG VERTICAL? / VERY SHORT VERTICAL ATTACHED TO CENTER OF FLAT TOP?

NO FLAT TOP? / SLANT LINE ON TOP?

Figure 2 Descriptions To Permit Classification of Sanskrit Characters (Including Those Shown in Figure 1)

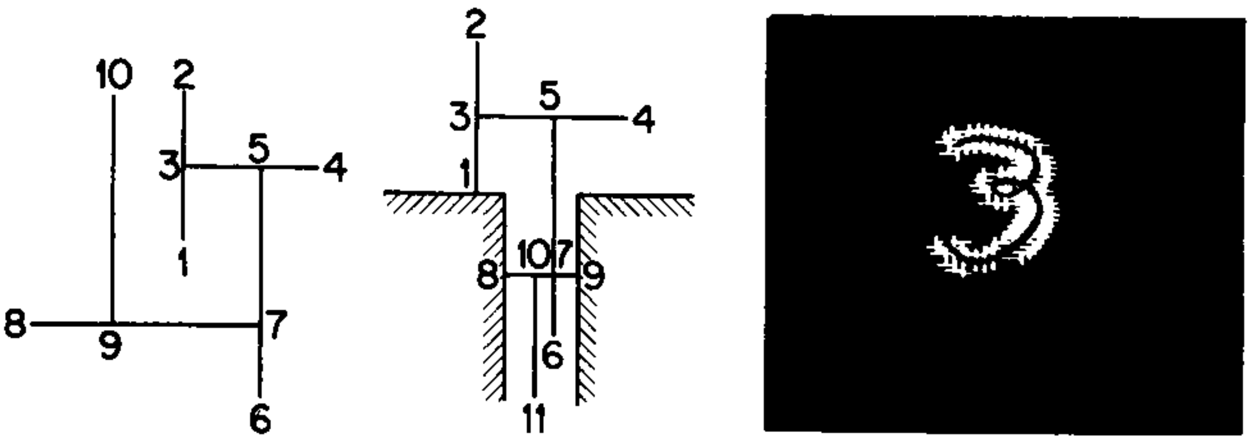


Figure 3 The Contour Follower. (A) Normal search pattern. (B) Search pattern when entering a narrow opening. (C) A photograph of the follower in operation

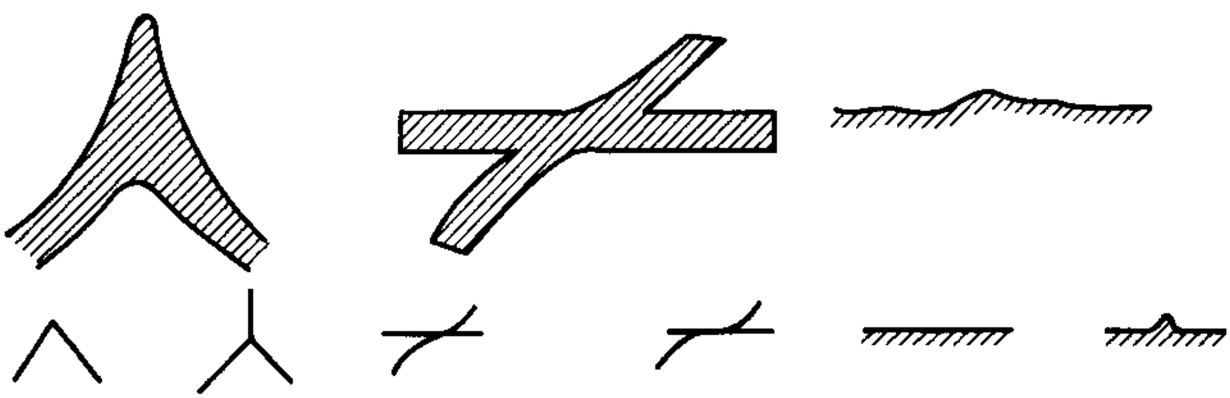


Figure 4 Examples of Line Drawings With More Than One Structurally Valid Decomposition

Class I

0 A single enclosure having two lobes (relative maxima) with respect to a diameter (longest straight line which begins and terminates on the object).

8 A single enclosure having four lobes with respect to a diameter; OR, two enclosures.

Class II

1, A single line, either straight or slightly curved.

22 A single "S" shaped line (one internal crossing with line between the Endpoints, and two lobes with respect to this line). Line between End-points has right diagonal orientation (slopes from upper left to lower right).

3 A single curved line having two lobes with respect to line between the Endpoints and no crossings with this line.

5 A single "S" shaped line. Line between Endpoints has a left diagonal orientation.

7_ A single "C" shaped line (a line having one lobe with respect to the line between its Endpoints).

Class III

6 A single enclosure below a single long line; OR, a single spiral shaped line.

9 A single enclosure above a single long line.

Class IV

4 Four line segments which intersect at a common vertex; OR, a "C" shaped curve and two straight or slightly curved lines meeting at a common vertex.

Figure 5 Canonical Descriptions for Numeric Characters

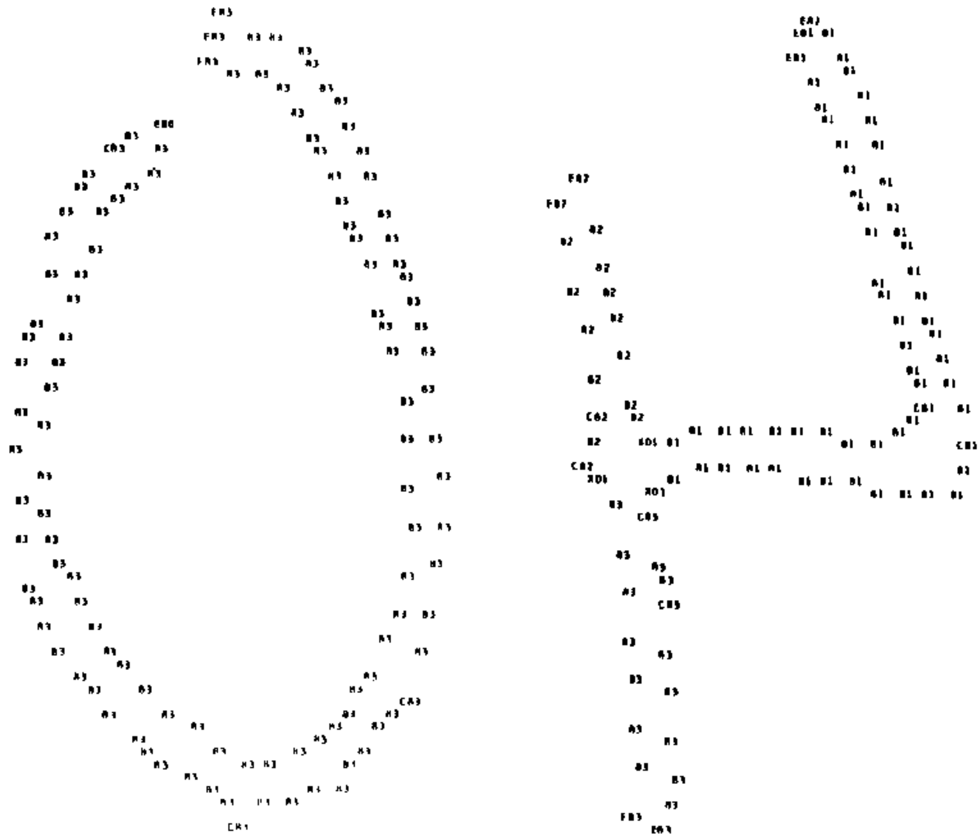
Classes I and III

An enclosure may be open. One or more small redundant line segments may be present.

Classes P. and IV

A redundant small enclosure may be added. One or more redundant line segments may be present.

Figure 6 Possible Numeric Character-Variations



The object being considered consists of the single line segment described below.. . Line segment 83 is a long curved line. It extends from vertex 83 (E, C,T) to vertex 80 (E, C, T). The diameter of this curve is 08 times the length of the line between the terminal points and has a vertical orienta-tion. The curve is open to the top. The The approach to vertex 80 is from the left. The approach to vertex 83 is from the right.

Line segment 81 is a long curved line. It extends from vertex 81 (E,C,T) to vertex 01 (X, L, C). The diameter of this curve is coincident with the line between the terminal points and has a left diagonal orientation. The curve is open to the left. The approach to vertex 81 is from below and to the right. The approach to vertex 01 is from the right. Line segment 82 is a long right diagonal line. It extends from vertex 82 (E, L,T) to vertex 01 (X, L, C). Line segment 83 is a long vertical line. It extends from vertex 01 (X,L,C) to vertex 83 (E,L,B).

Note: R = Right, E = Endpoint, C = Center, T = Top, B = Bottom, L = Left, X = Inter-section Point.

Figure 7 Computer Printout of Two Perceived Graphical Objects and Their Line Descrip-tions as Generated by LSADP