Automated Reading of the Printed Page

Donald E. Troxel

This paper is primarily concerned with the automated reading of the printed page resulting in a sequence of character codes which can be further processed to make the information available to a blind person in the form of Grade II Braille, spelled speech, or synthesized speech. Heuristics are described for automatic threshold determination, font-size determination, line and character acquisition, contour tracing, and the recognition of punctuation and characters. As the output of the reading machine for the blind is to be absorbed directly by a human, the specifications for a page reader were that the speed should be approximately equal to normal speaking rates and that the reader should make somewhat fewer mistakes than a human reader would make. The actual speed achieved was approximately 75 words per minute, which does not quite meet the speed requirement. However, the error rate specification has been met with a measured error rate of .07 per cent.

A number of researchers have been concerned with the problem of sensory aids for the blind. There have been numerous studies of obvious auxiliary sensory inputs such as audition and taction, and even studies of the potential usefulness of electrotactile, kinesthetic, and olfactory stimulation. At another extreme there has been a substantial effort expended with the goal of alleviating specific problems of the blind—particularly those of mobility and reading.

Work on mobility has centered largely on the development of radar-like devices to sense information which would enhance mobility. Relatively little effort has been expended on the preprocessing of this information and its presentation to the blind.

The work concerned with reading aids has been distributed more evenly across the various facets of the total problem. It has included the optophone which accesses print via a small vertical slit, leaving the blind with a very complex recognition problem; a direct translation aid developed by Bliss and Linvill[1] which presents an enlarged virtual tactile image of the print; a small, relatively low performance character recognition machine developed by Mauch Laboratories[2] which utilizes spelled speech as its output; and a moderately complex page reader developed at M.I.T. which can communicate with the blind via spelled speech, Braille, or synthesized speech.

The earliest version of the M.I.T. system was first demonstrated in the spring of 1966. It was then capable of reading a single line of text, recognizing the characters and punctuation, and spelling out these symbols aurally. This experimental system was not intended to be a prototype of a practical reading aid but rather was a real-time research facility for the experimental investigation of human information requirements and human learning capabilities upon which prototype designs must be based. This experimental system has facilitated research on print scanning, character recognition, control by a blind reader, and auditory displays.[3,4,5,6,7]

One of the crudest features of the initial reading machine system was the opaque scanner, and work commenced immediately to develop an improved scanner and carriage control which would handle whole pages of print rather than individual lines. During the spring of 1968, funds were made available for the purchase of a PDP–9 computer which was to be used primarily for reading machine development. This computer has formed the nucleus of a second research version of a reading machine for the blind which, while still not a prototype, has considerably improved capabilities and represents a significant step towards the eventual realization of a practical machine.

System Description

The system diagram for the second version of the reading machine is shown in Figure 1. The transformation of a printed page into Braille, spelled speech, or synthesized speech has been demonstrated with this system.

The printed pages used thus far have been removed from a fourth grade reader entitled *Roads to Everywhere* published by Ginn and Co., Boston, 1964, and printed in 14-point Textype type. Only those pages have been used which do not contain pictures.

The opaque scanner is a flying spot scanner in which a dot of light positioned on the face of the cathode ray tube (CRT) is imaged on the page. The total light reflected from the page is sensed by photo-

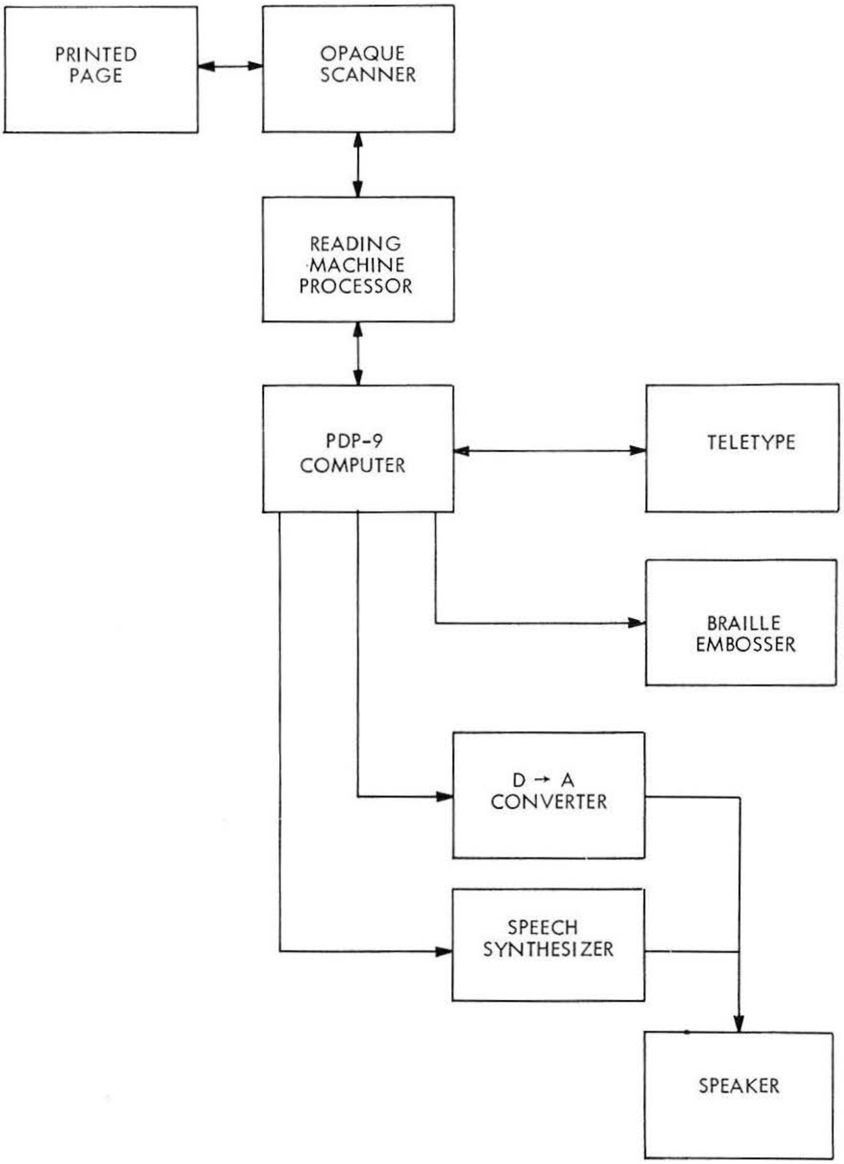


Figure 1. Reading machine system diagram.

multiplier tubes, the output of which is a function of the whiteness of the particular spot on the page. To achieve high resolution, the face of the CRT is imaged on to approximately one square inch of the page. The minimum necessary resolution has not yet been determined. Access to the whole area of the page is provided by a carriage which is integral to the opaque scanner and permits both horizontal and vertical motion of the page. Both the carriage and opaque scanner are operated by the reading machine processor (RMP) which in turn receives directives from the computer. This RMP in effect extends the instruction set of the computer and allows various complex functions to be initiated by the computer and to be performed while the computer is engaged in other calculations. One such directive which is often used is, "Given that the dot of light on the CRT is imaged on the contour of a character, find the next sequential contour point and return its coordinates to the computer." It was deemed necessary to provide these functions in the hardware rather than to perform them by computer program so that recognition rates could approximate human speaking rates.

The recognition of alphanumeric characters is a modification and extension of procedures developed by Clemens,[8] where a number or signature is computed from the exterior contour of a character, and the character is "recognized" by finding a match in a previously generated table of signatures and character codes. The sequence of operation of the page reading programs is indicated in Figure 2. The first step is page initialization where the threshold for the discrimination of black and white on the page is computed. The size of the typeface on the page is determined, and the left and upper boundaries of print on the page are located. Starting from the top of the page, the first line of print is acquired, and then the first character in this line is acquired. Then the exterior contour of the letter is traced; and if the character is deemed large enough, a signature is computed and the signature table is searched to yield the code for the character. Then an attempt is made to acquire the next character. Eventually all of the characters on that line will be processed and no such character will be available, resulting in an end-of-line determination. When this occurs, the program attempts to acquire the next line of print. If no such line is found, then the page is deemed finished.

If, after a character has been traced, it has been deemed small

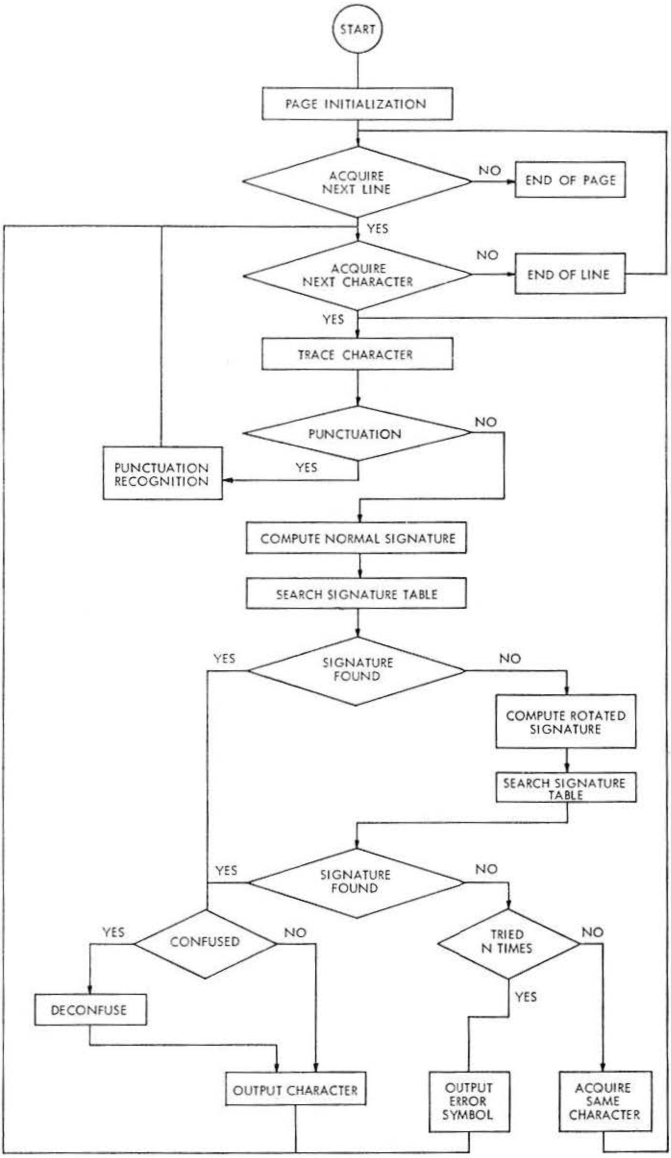


Figure 2. Page reader flow chart.

enough to be punctuation, control is transferred to a special routine which recognizes the punctuation mark and proceeds to acquire the next character.

As seen in the lower portion of Figure 2, it is possible that the first signature which has been computed is absent from the table. In this case an alternative method of computing the signature is used and the table is searched again. When a signature has been found in the table, it may or may not represent a unique recognition. This results from the fact that some shapes which correspond to two different characters result in the generation of identical signatures. Fortunately there are not very many occurrences of these confused signatures, and it has been possible to resolve these confusions by making auxiliary tests. If both signatures just generated do not exist in this signature table, then the character is reacquired and traced again. This procedure of trying again is applicable because it is an experimental fact that the statistics of signatures generated by successive tracings of the same character are similar to the statistics of signatures generated by successive tracings of like characters.

Page Initialization

Page initialization performs three main functions: the determination of an acceptable threshold for subsequent black-white decisions, the determination of the size of the typeface existing on the page, and the positioning of the carriage for subsequent acquisition of text. As the scanner can only access approximately one square inch of the page area, it is first necessary to locate the carriage so that this aperture contains some print. In the present system, this is accomplished simply by moving the aperture to the center of the page; then a number of A to D conversions are made over this central portion of the page, and the threshold is computed from a histogram of the results of these A to D conversions, as indicated in Figure 3. The two peaks of this histogram corresponding to the most common black and the most common white are found. Then the minimum of the central region of the histogram as bounded by these peaks, is determined. The threshold is then computed as the mid-point of a horizontal line drawn a fixed distance above this minimum and bounded by the interior sections of the histogram.

The font size is characterized by the x-height, which is the height of

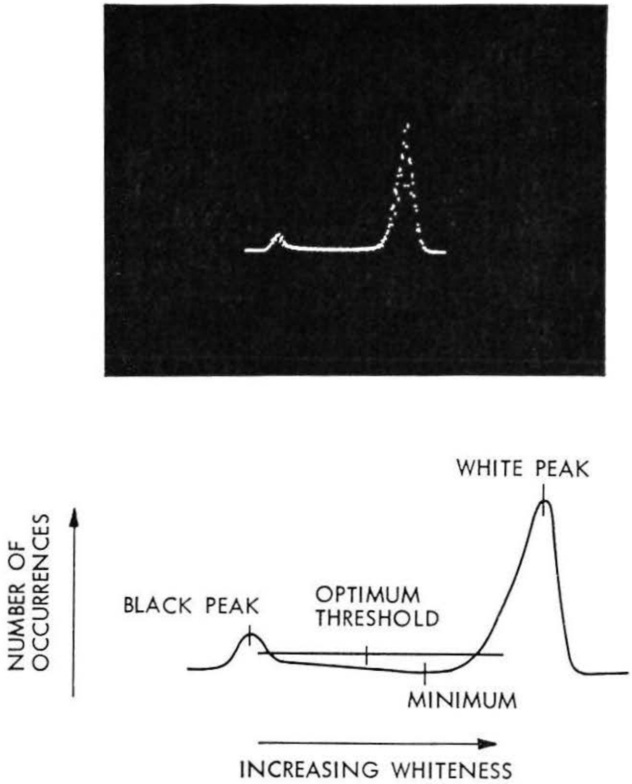


Figure 3. Threshold determination.

the lower-case x. This x-height is determined by tracing the contours of a number of characters and is calculated as the average height of the smallest characters, provided that there exists one character which is substantially taller than the smallest character. Care is taken, of course, that specks do not find themselves classified as characters.

The left print boundary is determined by moving the carriage so that the left edge of the aperture is guaranteed to be black (i.e., off the page). Then the aperture is moved right until a substantially all-white area is found. The motion to the right is continued until a substantial amount of black (the left edge of the print) is found. The aperture is then moved so that the print boundary is near the left edge of the aperture and the accuracy of this move is checked.

The top print boundary is found in a similar manner, but the top line of print is positioned in the center of the aperture.

Line Acquisition

Line acquisition makes extensive use of the horizontal histogram (HH) directive which is issued by the computer and executed by the RMP. When this directive is initiated, the RMP interrogates all points at a specified y value within the aperture and returns the total number of black points encountered. The heuristic used for the acquisition of a line involves the execution of the HH at selected y values. At first the search is coarse, that is, with a significant space between successive y values. When the number of black points encountered is greater than a fixed constant for four successive lines, the HH is executed every fourth y line (Fig. 4A); and the maximum number of black points is computed. When the number of black points resulting from a HH is less than one-eighth of the maximum encountered thus far, the base line of the line of text has been passed; and a fine search is performed but for increasing values of y until the number of black points is greater than one-half of the maximum previously encountered (Fig. 4B). Since this heuristic has been implemented, there has not been even a single instance of failure in the determination of the base line of a line of text, whether or not it contains letters with descenders.

For second and subsequent lines, the motion of the aperture of the scanner is effected to place the next line approximately in the center of the aperture, as well as returning the aperture to the left edge of the print. An end-of-page determination is made if no line can be found within the scanner aperture.

Character Acquisition

Character acquisition again makes use of another directive executed by the RMP. In this directive, a vertical scan is performed between two y values starting at a particular x value. If a black point is not encountered, the vertical scan is performed again but indexed to the right in x. This continues until a black point is encountered or the value of x equals that of the x mask. This scan is indicated in Figure 5. The values of the upper and lower y limits, the starting value of x and the x mask are transmitted to the RMP prior to the execution of the acquisition scan. Before the acquisition scan is initiated, the aperture is moved if necessary. The x mask is positioned so as to guarantee that if a character is found before reaching the x mask the character will be fully within the scanner aperture. The contour of the character is then

Figure 4. Line acquisition.

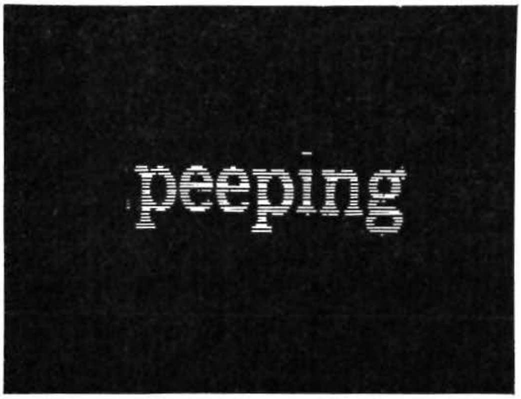


Figure 4a. Search every fourth line.

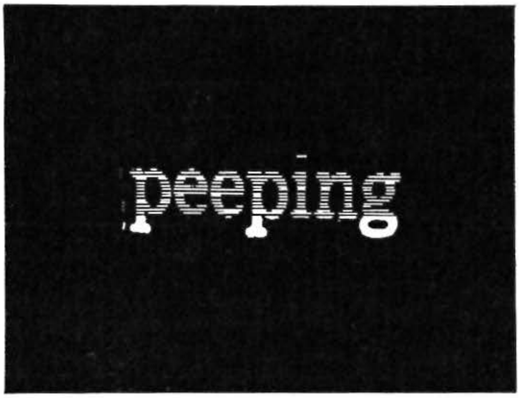
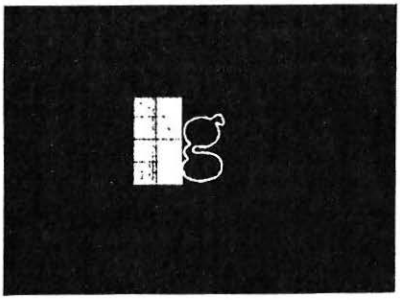
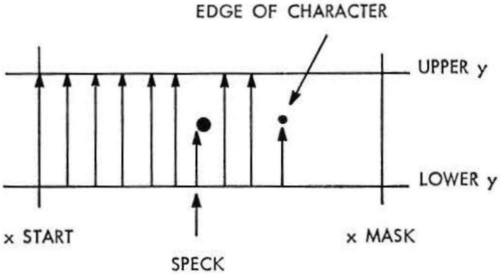


Figure 4b. Addition of fine search upward.

Figure 5. Acquisition scan.





traced. If the number of contour points is smaller than a fixed constant, it is deemed a speck; and the acquisition scan is restarted to the right of that speck. When large specks are encountered they are sometimes "recognized" as periods. The end-of-line determination is made if the acquisition scan has proceeded for a certain value of x with no more than specks being found. When the first character of a line has been found, the left edge of this character is compared to the left edge of the print on the page to see if a paragraph has been encountered. For succeeding characters, the distance between the left edge of the newly acquired character and the right edge of the previously acquired character is used to determine whether or not a space or spaces should be transmitted.

Contour Tracing

The fact that the exterior contour of a letter is traced has often been considered a key feature of the character recognition algorithm used. However, contour tracing has little to do with the character recognition algorithm, as it is primarily a means of defining the precise location of the character to be recognized. It is, however, a relatively fast technique for extracting the useful information from a fairly large area if one is confined to an optical transducer which can test only a single point at a time. A character in a 50 × 100 grid would require 5,000 tests if every point on the grid were to be examined. However, it would have only about 300 edge points, which would require approximately 600 tests.

There are many ways in which contour tracing can be implemented.[5] Perhaps the simplest digital method is the square trace (see Figure 6) which can be described by the following rule: If the last point tested was black (white), then make a 90° left (right) turn to determine which point will be tested next. This contour trace algorithm works well on fixed data. As a matter of choice, the points which are interrogated are along the edges of a letter. These, of course, are the very points which are on the borderline of being black or white. If there is even a small amount of noise present in the optical scanner, successive interrogations of the same point are likely to reveal different answers.

Because of the square trace algorithm's propensity to test the same point twice, it is possible to become trapped in black or white. A

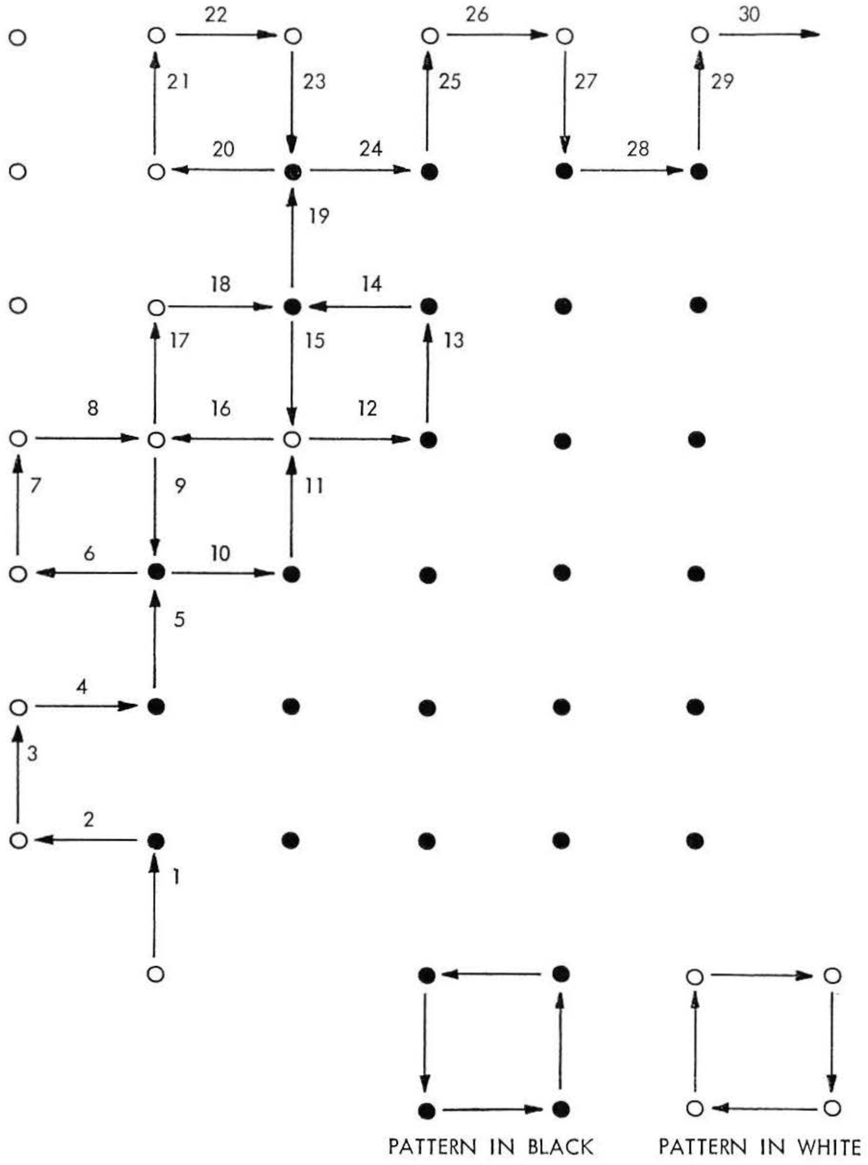


Figure 6. Square trace.

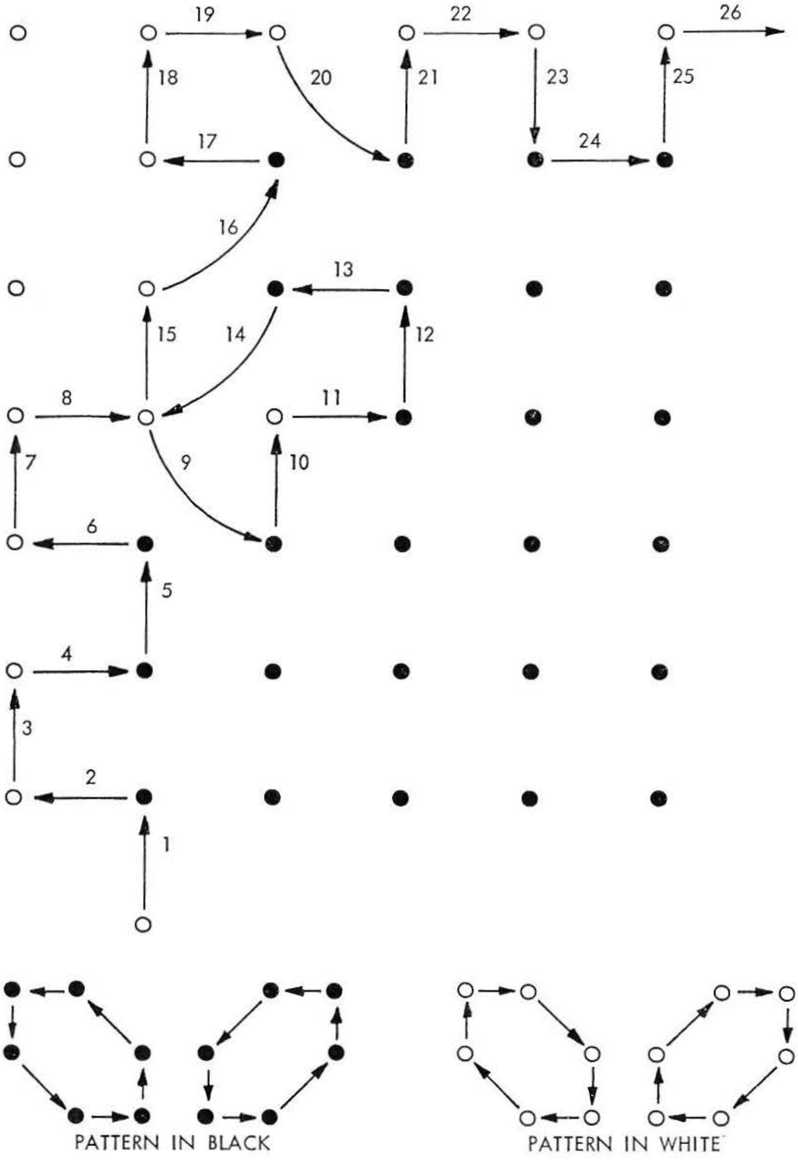


Figure 7. Modified square trace.

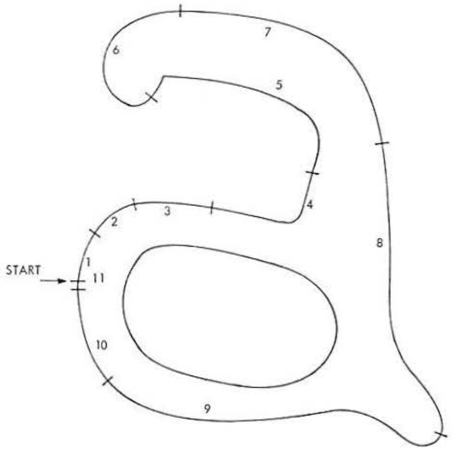


Figure 8. Segmented contour trace.

variation of the square trace algorithm, which is shown in Figure 7, eliminates the possibility of becoming trapped in black or white when the spatial quantization and grayness of the letter edges are such that the picture elements which are on the borderline between being black or white are less than two coordinate units wide. In general, this variation minimizes the square trace algorithm's tendency to test the same point twice by forcing diagonal moves when the three previous points were either all white or all black.

As is indicated in Figure 7, it is possible to become trapped in black or white even when using the second algorithm. The trapped condition is detected by the occurrence of six successive black (white) points and is termed a contour trace error. As the contour trace algorithm is realized in hardware by the RMP, the contour trace subroutine merely has to ask for the next contour point. Of course, the subroutine must determine when to terminate the tracing of a character. This is accomplished by a distance calculation of the present contour point relative to the initial contour point. The distance measure used is the maximum of the magnitude of the displacement in x or y. To minimize the time taken for the contour trace, this distance calculation is made only as often as is necessary to be sure that the original contour point has not been bypassed. Thus, the contour of a character is traced in segments as indicated in Figure 8, where the length of the perimeter of the next segment is determined by the distance measure of the present contour point to the initial contour point. Each time the distance calculation is made, a check is also performed to see if any contour trace errors have occurred on the previous segment; and if such is the case, the contour trace is restarted at the end of the previous segment. The shape whose contour has been traced is deemed a speck if the number of contour points is below a fixed threshold or if the initial distance calculation is less than a constant.

Punctuation Recognition

If the number of contour points of the shape as traced by the contour trace routine is smaller than a constant, then the character is considered to be a punctuation mark. Of course, this constant could easily be computed as a function of the typeface size. The allowable punctuation alphabet is shown in Figure 9, which also indicates the sequence of decisions which are performed.

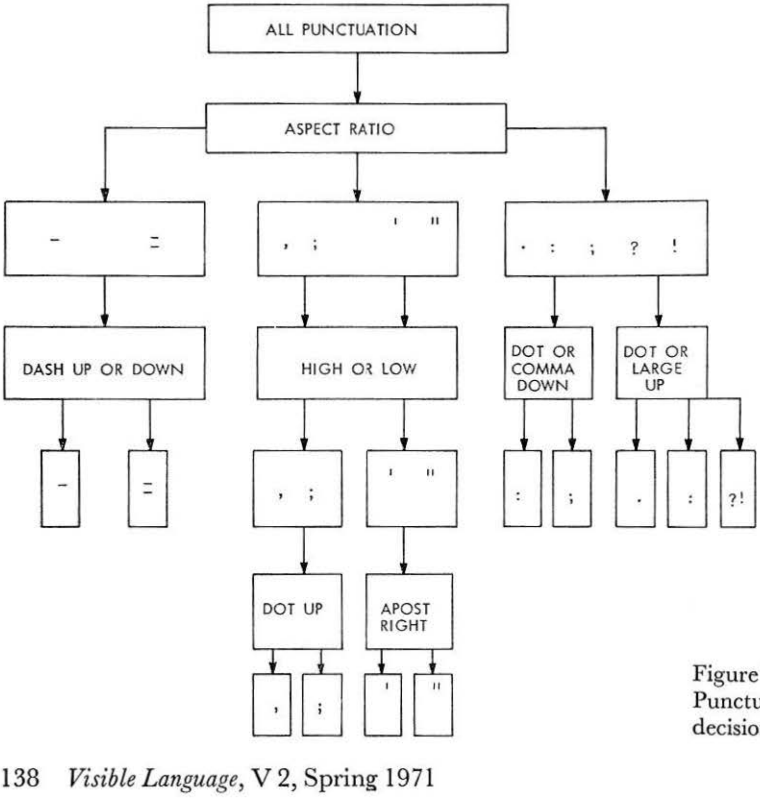


Figure 9. Punctuation decision tree.

The first decision is based upon the aspect ratio of the rectangle circumscribing the shape and divides it into three classes. A further discrimination between a dash and an equal sign is made by searching areas just above and below the shape for the existence of another dash-like character. It is conceivable that a shape which is not dash-like in its aspect ratio could be found, and this would be interpreted as an error. Those punctuation marks with the largest aspect ratios are further subdivided by their vertical position into two more classes, and these classes are resolved by searching appropriate areas for possible additional pieces of the punctuation mark. Punctuation marks which have aspect ratios near unity are similarly further classified by appropriate searches above and below the shape. The existence of a large shape above a dot is taken to indicate the presence of a question or exclamation mark, and these larger shapes are treated in exactly the same manner as alphanumeric characters.

Character Recognition

The basic scheme for character recognition is a modification and extension of the one that was used in the first M.I.T. reading machine system and is described in detail by Clemens[8] and Seitz.[5] After the contour of a letter has been traced, a characteristic number or signature is generated and a table of previously encountered signatures is searched for a match. The signature is computed by the method indicated in Figure 10.

As previously noted, there are two methods used to generate signatures—normal and rotated. In the normal case the signature is a concatenation of a code word and a coordinate word. For the rotated signature the same code word is used but the coordinate word is replaced by a rotated code word.

The code word is computed by finding local extrema for the horizontal and vertical directions if the excursions along the contour are greater than one-quarter of the letter height or width. That is, for example, an x maximum is not denoted unless one had to travel east for at least one-quarter of the letter width and then west for one-quarter of the letter width. With this scheme, minor features of the letter such as the droop in the upper left part of the "a" do not introduce extraneous extrema. The sequence of extrema can be coded with one bit per extremum if the first extremum recorded is an x minimum

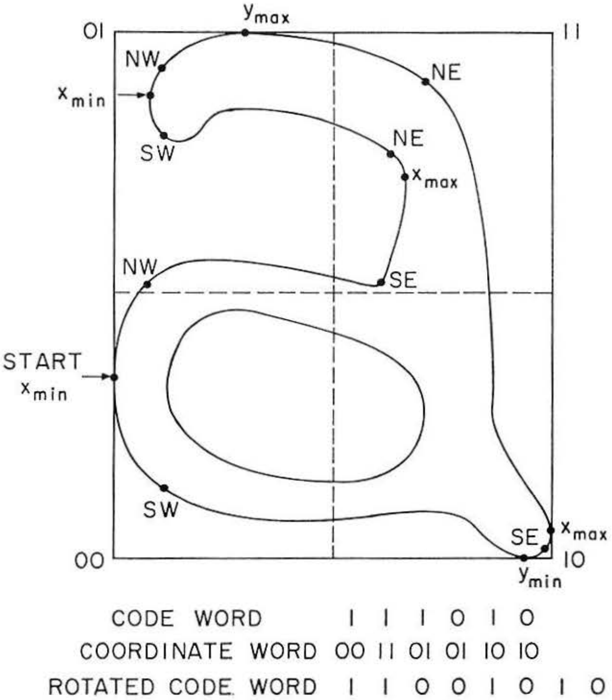


Figure 10. Signature generation.

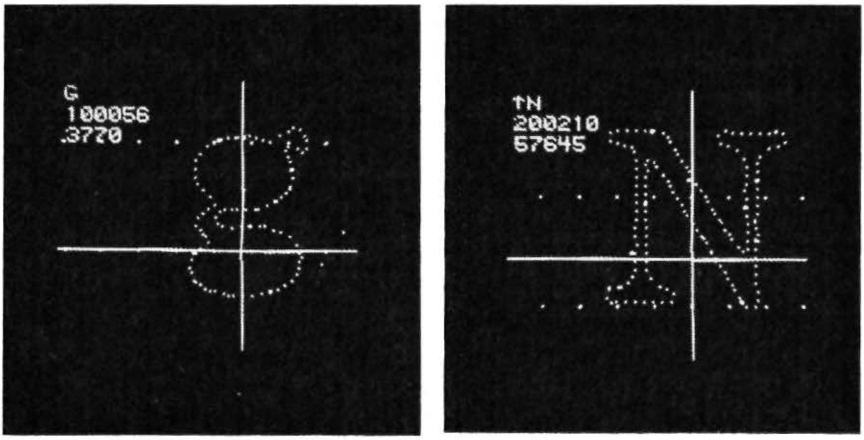


Figure 11. Training mode display.

and the contour trace is clockwise. The coordinate word is made up of a sequence of bit pairs specifying the quadrant in which a horizontal or vertical local extremum occurred. The rotated code word is computed in the same manner as the normal code word except that local extrema are determined with respect to x and y axes rotated by 45 degrees. These binary sequences, along with a small amount of information concerning the aspect ratio of the letter and its position relative to the y lines found by the line acquisition procedure, comprise the signature.

In order for characters to be recognized, the particular signature must have been encountered previously; that is, the machine must have been previously trained by a sighted operator. To enable the sighted operator to recognize the letters, a visual display (Fig. 11) is made. When the machine is unsuccessful in the search for a match for a particular signature, it can ask the trainer what the character is. Given this information, it can thus add to its experience. There is a certain amount of noise involved in the determination of black and white, and the presence of this noise actually simplifies the training procedure, as it appears that successive tracings of the same "a" have variations similar to tracings of different "a's." Thus a character repeat mode is provided in which one can lock on to a particular letter and build a repertoire of signatures for that particular letter.

Multiple Font Recognition

There are three aspects of the page reader programs which are peculiar to a specific font. These are the signature table, and the punctuation and deconfusion routines. Separate signature tables for any desired font can be generated easily by use of the training mode. The deconfusion routines which may be required must be programmed to resolve confusions generated during the training process. Punctuation recognition routines are probably quite universal as punctuation marks have small variations among most fonts. No provision has yet been made for truly multiple font recognition, i.e., when a single page contains a mixture of fonts.

Performance

Ten successive readings of the same page were made in order to test the accuracy of the page reading process. The page read consisted of

1087 characters, including letters, punctuation, spaces between words, end-of-line determinations, paragraph determinations, and an end-of-page determination. A total of twelve errors were made for a gross error rate of approximately 0.11 per cent. Four of the errors dealt with spaces, either missing or added. A program "bug" has been found and corrected so that this type of error no longer occurs. Two punctuation errors occurred in which a single quote was substituted for a double quote. There were four letter substitution errors and two letters not recognized. Virtually all of the mistakes concerned with letters were the result of broken or touching letters. If we discount the space related errors, the error rate was approximately 0.07 per cent. People are normally accustomed to reading text with considerably higher error rates.

The speed attained by the page reader is adequate for Braille or spelled speech output. It is, however, somewhat marginal for synthesized speech output. The average speed is 75.5 wpm. A large amount of the time spent in reading is "wasted" while the computer is awaiting completion of carriage motions. If we subtract the time spent waiting for completion of carriage motions, the reading speed would be 119 wpm.

Typeface Design

The following characteristics of typefaces are desirable to minimize the complexity of character recognition procedures as well as to facilitate poth lower error rates and higher recognition speeds. While the reliability of optical character recognition machines can be enhanced by special type styles, the type designer should not be expected to go to the extremes required by the magnetic characters on bank checks or OCR-A (Fig. 12).

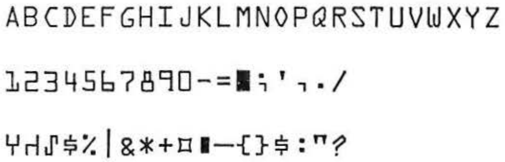


Figure 12. OCR-A.

Virtually all print reading machines must represent the image of each character as a two-dimensional array of black and white dots. The required resolution or number of picture elements (dots) per inch is a function of both the minimum stroke width of a character and the minimum distance between strokes (or between serifs). Higher resolution is expensive both in cost of the optical scanner and in processing or recognition time. In addition, some machines are sensitive to variations in stroke width, so it is desirable for a typeface to have a stroke width which is uniform (consistent with aesthetic considerations) and is wider than some minimum, such as approximately 1/50 inch. It is equally important that the minimum distance between strokes or between serifs be at least as great as the minimum stroke width. This minimum separation distance becomes critical between adjacent letters. Ligatures are acceptable as they can be treated as an additional character. Consistency is the key—if a letter pair always touches, it can be handled even though the complexity of the recognition procedure is increased slightly. Letter pairs that *sometimes* touch cause complications.

Another desirable characteristic is that characters be distinctly different. This statement may sound rather silly, but it is not uncommon for zero and capital Oh to be similar or for one and lower-case l to be identical. And it is desirable that this quality of distinctness be independent of vertical position. Common offenders in this regard are 9 and lower-case g, upper-case P and lower-case p, and upper-case Y and lower-case y. Virtually all automatic readers process each character separately and do not make use of contextual information as do human readers. Distinctness independent of vertical position facilities the task of reading skewed lines or unevenly spaced characters.

Most type styles designed expressly for automatic reading are sans-serif. This is not in itself a desirable goal, but it does make it easier to adhere to stroke width and character separation specifications.

---

1. J. G. Linvill and J. C. Bliss, "A Direct Translation Reading Aid for the Blind," *Proc. IEEE, 54* (1) (January 1966), 40–51.

2. G. C. Smith and H. A. Mauch, "Summary Report on the Development of a Reading Machine for the Blind," Mauch Laboratories, Inc., Dayton, Ohio, VA

Contract No. V1005M–1943 (June 1968). See also: G. C. Smith, "Type Reading Machines for the Blind," *The Journal of Typographic Research*, II (April 1968), 107–125.

3. D. E. Troxel, F. F. Lee, and S. J. Mason, "A Reading Machine for the Blind," QPR No. 89, RLE, M.I.T.

4. D. E. Troxel, F. F. Lee, and S. J. Mason, "A Reading Machine for the Blind," *Digest of the 7th International Conference on Medical and Biological Engineering*, 1967. Stockholm.

5. C. L. Seitz, *An Opaque Scanner for Reading Machine Research*, SM Thesis, M.I.T., 1967.

6. D. E. Troxel, "Page Position Selection for Use with a Reading Machine for the Blind," *Digest of the 8th Annual IEEE Symposium on Human Factors in Electronics*, 1967, Palo Alto.

7. D. E. Troxel and G. F. Pfister, "Optimum Threshold Level for the Reading Machine Opaque Scanner," QPR No. 86, RLE, M.I.T.

8. J. K. Clemens, *Optical Character Recognition for Reading Machine Applications*, Doctoral Thesis, M.I.T., 1965.

This work was supported principally by NIGMS Grants 5 POI GM15006–03 and GM14940–04, and in part by the Joint Services Electronics Program (Contract DA28–043–AMC–02536 (E)).

Cover Illustration: The Duke Parchment Palimpsest Fragment

According to Dr. William H. Willis, professor of Greek at Duke University and president of the American Society of Payrologists, the Duke Plato (*PKuk.* inv. G5) is significant in three respects: (1) it is the first example of a Plato text on parchment, (2) it is the earliest known Plato in codex form, rather than in a papyrus roll, and (3) it is the first known application of Roman uncial to parchment. It takes its place among only four surviving examples of first- or second-century parchment codices—bound books invented in Rome in the late first century.

It was close examination of the writing that convinced Dr. Willis that the parchment might be a rarity. Not an entire word was visible to the naked eye, but there was evidence of Greek uncial writing which was reserved for the most important texts in the most elegant editions. Ultra-violet light made visible the lines of the original uncial work by tracing the grooves etched by the writer's pen on both sides of the skin. Comparison of its "hand" showed it to be almost exactly like those in a papyrus roll containing part of the Iliad which came to light in 1888 with the mummy of a Graeco-Egyptian lady whose entombment is known to have been in the second century. The same hand is in another Homer papyrus at Oxford, which also is assigned a date in the early second century. And there are more than a dozen well-documented papyrus rolls written in Roman uncial, all datable to the first or second century, none of them later.

"In the history of manuscripts it is the period preceding the fourth century which remains essentially uncharted," Dr. Willis states, and it is on this early development that the Duke Plato throws light.