INTRODUCTION

To an increasing degree, equipment is available which is capable of converting photographic information into machine readable form or converting computer files into a visual image more or less resembling a half-tone picture, thus rendering photographic and other pictorial information available as data for processing by digital computers. Three major directions of effort have ensued. Necessary utility routines have been developed for managing I/O, file manipulation, and the implementation of languages to facilitate programming effort; analytical work toward character and pattern recognition, and toward parametric characterization of pictures has led to algorithms for accomplishing various sorts of mensuration and analysis; and, finally, algorithms have begun to emerge which are designed to change the appearance of a picture by modifying the file which represents it. A brief list of references to this work is given at the end of the paper, including an excellent review of the field by Lipkin and Rosenfeld.

It is the third type of processing which has motivated the work described here, with the belief that picture modification in an on-line, interactive environment will soon be an important tool to experimental psychology and a useful training device for photo-interpreters, medical technicians, and others. In this context there are two types of manipulation: global manipulation of an entire file, for example, for the purpose of enhancing or degrading picture quality, and local manipulation in which one or more elements of a picture are modified by enlargement, rotation, translation, erasure, or the like.

REPRESENTATION OF IMAGE AND NOTATION

The image to be processed is represented in a file by a matrix, $z$, of positive integers. The matrix elements, $z_{ij}$, are the (digitized) average gray levels or transmittance values over a square area of the image, and each is associated with the midpoint of its respective square. In the discussion we define the image as a function of three variables, $y_i, x_j, z_{ij}$. The $y_i, x_j$ are coordinates of midpoints of grid squares of the $i$th row and $j$th column of the $M$ by $N$ matrix; they normally take on the values $y_i = .5, 1.5, 2.5, ...; i = 1, ..., M$ and $x_j = .5, 1.5, 2.5, ..., j = 1, ..., N$; i.e., $y_i = i-.5, x_j = j-.5$. The upper left hand corner square of the image is represented by $y_1, x_1, z_{11}$.

An object or area within the image is defined by one or more connected boundaries. A list of $y_i, x_j$ pairs outlining the object represents a boundary. The convention is followed of recording the coordinate pairs sequentially as though one were “walking around” the object, keeping it to the left. A restriction is that each boundary is continuous and closed in the sense that the absolute difference between successive $y$ (and $x$) values in the list, including the first and the last, never exceeds 1; that is, no “jumping over elements” in either the horizontal or vertical direction is implied by the list. If, in scanning the boundary, it is found that the restriction does not hold, the “missing” coordinate pairs are linearly interpolated and inserted in the list.

Boundary points are considered to be inside, or a part of, the represented object. Where the area to be manipulated lies on or touches the frame of the image, the coordinate pairs of the frame itself are taken as the boundary.

DELINEATION OF OBJECTS

Manipulation of local areas or objects in a picture requires some technique of defining the domains of operation. In our programs, a local area was represented as a set of contiguous horizontal strips, the end points of which comprise its boundary. In the illustrations which follow, a boundary is denoted as a list of
points \( B_k = (y_{ik}, x_{jk}) \), where \( k = 1, 2, \ldots, L \). This list is scanned and then sorted in ascending order of \( y \). For each \( y \), sets of left and right \( x \)-values are recorded in ascending order. The strips, not necessarily contiguous, which these sets define thus comprise the area, row by row.

To illustrate, assume a square area having an even number of elements, for example, 16 as in Figure 1a, with the following boundary:

\[
\begin{array}{cccc}
  k & y_{ik} & x_{ik} & k & y_{ik} & x_{ik} \\
  1 & .5 & 3.5 & 7 & 3.5 & .5 \\
  2 & .5 & 2.5 & 8 & 3.5 & 1.5 \\
  3 & .5 & 1.5 & 9 & 3.5 & 2.5 \\
  4 & .5 & .5 & 10 & 3.5 & 3.5 \\
  5 & 1.5 & .5 & 11 & 2.5 & 3.5 \\
  6 & 2.5 & .5 & 12 & 1.5 & 3.5 \\
\end{array}
\]

This list, when sorted as described, yields:

\[
\begin{array}{cccc}
  y-values & \text{Left and right } x \text{-values} \\
  .5 & .5, 1.5 & 2.5, 3.5 \\
  1.5 & .5, 3.5 \\
  2.5 & .5, 3.5 \\
  3.5 & .5, 1.5 & 2.5, 3.5 \\
\end{array}
\]

Steps are taken in the initial scanning of the original boundary list to ensure that each section of a horizontal strip contains an even number of \( x \)-values. When an odd number of \( x \)-values appear for a section of a given \( y \), as, for example, when a horizontal boundary contains an odd number of elements or when a vertex is encountered, it is necessary to replicate one of the \( x \)-values. Suppose, for instance, that the object given in Figure 1b were to be represented. In the scanning process the pairs representing \( B_6, B_9, \) and \( B_{13} \) would be replicated and the strip representations after sorting would appear as:

\[
\begin{array}{cccc}
  y-values & \text{Left and right } x \text{-values} \\
  .5 & .5, .5 & 1.5, 2.5 & 3.5, 4.5 \\
  1.5 & .5, 4.5 \\
  2.5 & .5, 4.5 \\
  3.5 & .5, 1.5 & 2.5, 3.5 & 4.5, 4.5 \\
  4.5 & .5, .5 \\
\end{array}
\]

Obviously, an object may have more than one boundary (e.g., an annulus); and a boundary can have singular points (e.g., a figure eight). When more than one boundary defines the area, the boundary lists are combined before sorting. In this manner, shapes other than simple closed curves, such as disjoint areas, can be manipulated in one operation. Once the areas are defined as strips, the elements contained therein can be translated with or without rotation, subjected to gray level transformations, replaced by other contents as in hole filling, and, by appropriate horizontal and vertical operations, contracted and expanded.

**TRANSLATION AND ROTATION**

Let \( i_t \) and \( j_t \) denote the indices of some pivot or anchor point \((y_{it}, x_{it})\), and let \( i_r \) and \( j_r \) represent the indices of the point into which the pivot point is to be mapped. Since the indices are here measures of distances, translation of an area of a picture into another position such that any \((y_i, x_j)\) of the object maps into \((y_{i'}, x_{j'})\) is a trivial operation in that all the new indices for the \( z_{ij} \) values are given by

\[
i' = i + (i_t - i), \quad j' = j + (j_t - j),
\]

and the new coordinate pairs by \((y_{i'}, x_{j'})\). When translation is combined with rotation about the pivot through a counterclockwise angle, \( \alpha \), the new coordi-
nates, which are given by
\[ y' = y_i + (i - i_r) \cos \alpha - (j - j_r) \sin \alpha \]
\[ x' = x_j + (j - j_r) \cos \alpha + (i - i_r) \sin \alpha \]
will not always be at the midpoints of the new cells (except in the case of angles which are multiples of 90 degrees). The fact that there is no longer a one to one correspondence between the original cells and the new cells requires a procedure for assigning \( z_{ij} \) values. Assignment of the gray level \( z_{ij} \) to the new cell into which the midpoint \( y_i, x_j \) falls would result in overwriting certain cells and leaving others empty. The “tilting” of each square element within the area rotated suggests a weighted assignment of the gray level on the basis of the area covered to the (possibly) six cells overlaid, as shown in Figure 2a. Such an allocation of gray levels might be called a “girdle” technique, since the orientation of the overlying square is constrained. In its place we implemented an approximation, the “chemise”.

The chemise approximation allocates gray levels to underlying squares as if each rotated square had been allowed to “settle” about its center into the horizontal orientation shown in Figure 2b. Weights are assigned to the covered cells, never exceeding four in number, on the basis of the area covered. The indices and weights resulting are as follows:

- \( i' = \text{integral part of } y' + 0.5 \)
- \( j' = \text{integral part of } x' + 0.5 \)
- \( p = \text{fractional part of } y' + 0.5 \)
- \( q = \text{fractional part of } x' + 0.5 \)

\[ w_{i',j'} = (1 - p)(1 - q) \]
\[ w_{i',j'+1} = (1 - p)q \]
\[ w_{i'+1,j'} = p(1 - q) \]
\[ w_{i'+1,j'+1} = pq \]

These weights determine the proportionate parts of the \( z_{ij} \) value assigned to each of the four cells.

The illustrations which follow were obtained with the chemise approximation. It appears to be adequate for visual purposes.

Picture 1, for example, is computer generated. The original, which is not reproduced, consisted of thirty horizontal bars of equal width, with gray levels ranging from 0 to 29. (The actual matrix is 400 by 400.) The portion of the original bars contained in a butterfly outline were rotated 20 degrees counterclockwise with the upper right wing tip as an anchor. Though the reproduction, which was obtained from a local vendor who scanned the computer file with off-the-shelf equipment, suffers from cyclical noise and inability to show all thirty gray levels, the appearance of straight lines has been retained and it is doubtful that quality could have been enhanced by increased precision in the algorithm.

**HOLE FILLING AFTER ERASURE**

Suppose that an object is to be erased, that is, that the gray levels within its boundary \( B_k, k = 1, 2, ..., L \), are to be set to zero. Frequently the white area thus created will be visually distracting or unesthetic. The hole filling problem is that of associating a corresponding gray level \( z \), which is in some sense consistent with the gray levels \( z_k \) of the \( B_k \), with each point \((x, y)\) of a given set of points interior to the boundary.
The boundary, $B$, may be taken to be a closed curve and the set of $(x, y)$ to be the interior of $B$, though we will also provide for situations in which $B$ "runs off the picture" by allowing for missing gray levels along the edge. The statement that the gray levels, $z$, in the interior are consistent with those of $B$ is here interpreted to mean that a given implementation of a hole filling technique can yield a result which is subjectively acceptable in the sense that neither $B$ nor the hole are visually apparent.

Consider the stencil consisting of a given point $(x, y)$ interior to $B$ and the corresponding boundary points $(x_1, y_1), (x_2, y_2), (x, y_1)$ and $(x, y_2)$, as shown on the diagram, with associated gray levels $z, z_1, z_2, z_3$ and $z_4$, respectively, where $z$ is unknown (and, indeed, undefined).

Geometrically, the diagram may be interpreted either as the projection on the $x$-$y$ plane of the five points whose coordinates are given, or as a two dimensional array having a $z$-value associated with each point. Algebraically, we are going to determine a number $z = z(x, y, x_1, x_2, y_1, y_2, z_1, z_2, z_3, z_4)$ such that

1. $z$ is a linear combination of $z_1, z_2, z_3,$ and $z_4$;
2. $\lim_{x \to x_i} z = z_{i-\delta}$, $i = 1, 2$;
3. The horizontal weights (coefficients of $z_1$ and $z_2$) and vertical weights (coefficients of $z_3$ and $z_4$) will be directly proportional to two non-negative horizontal and vertical control parameters, $L$ and $M$;
4. The horizontal and vertical weights will decrease monotonically with $|x - x_i|$ and $|y - y_i|$, where $x_1 \leq x \leq x_2$ and $y_1 \leq y \leq y_2$.

In what follows we will first develop an interpolation formula for the stencil shown. Then we will develop a mixed interpolation—extrapolation formula for use when a point of $B$ is missing, thus illustrating the handling of exceptional stencils.

To form the linear combination we first interpolate in the horizontal direction by forming

$$\frac{x-x_1}{x_2-x_1}z_2 - \frac{x-x_2}{x_2-x_1}z_1,$$

that is,

$$\frac{1}{x_2-x_1}[(x-x_1)z_2 + (x-x_2)z_1].$$

Similarly, the vertical interpolant is taken to be

$$\frac{1}{y_2-y_1}[(y-y_1)z_4 + (y-y_2)z_3].$$

The weights for these interpolants are defined as

$$\frac{L(y_2-y_1)}{M(x_2-x_1)+L(y_2-y_1)} \quad \text{and} \quad \frac{M(x_2-x_1)}{M(x_2-x_1)+L(y_2-y_1)},$$

so that conditions (3) and (4) are met, and the linear combination is

$$z = \frac{1}{M(x_2-x_1)+L(y_2-y_1)} \left[ L\frac{y_2-y_1}{x_2-x_1}[(x-x_1)z_2 + (x-x_2)z_1] + M\frac{x_2-x_1}{y_2-y_1}[(y-y_1)z_4 + (y-y_2)z_3] \right]. \quad (1)$$

That condition (2) holds is a simple exercise.

Notice that changes in vertical or horizontal patterns can be achieved by adjusting $L$ or $M$, and also there would be no loss of generality were we to set $M = 1 - L$. Furthermore, it is simple to introduce a "gain" parameter $K(x, y)$ as a multiplier of the right-hand side of the Equation (1). These parameters, $L$, $M$ and $K$ when used in conjunction with such unit operations as rotation, patching, superposition (or multiple-exposure), and smoothing would seem to be adequate for most practical purposes.

Suppose $(x_2, y, z_2)$ is missing. Consider the stencil

$$(x_1, y, z_1) \rightarrow (x, y, z) \rightarrow (x_1, y_1, z_3) \rightarrow (x_1, y_1, z_4)$$

where, $z_3$ and $z_4$ are assumed known. For the horizontal interpolation we wish to form

$$z_1 + m(x-x_1),$$

where $m$ is an appropriately weighted average of the
The weights are taken as

\[
\frac{y - y_1}{y_3 - y_1} \quad \text{and} \quad \frac{y_3 - y}{y_3 - y_1},
\]

respectively, yielding

\[
z_1 + m(x-x_1) = z_1 + \left[ \frac{(y - y_1)(z_4 - z_3)}{(y_3 - y_1)(x-x_1)} \right] (x-x_1)
\]

\[
+ \left[ \frac{(y_3 - y)(z_4 - z_3)}{(y_3 - y_1)(x-x_1)} \right] (x-x_1)
\]

\[
= z_1 + \frac{(y - y_1)(z_4 - z_3) + (y_3 - y)(z_3 - z_3)}{y_3 - y_1}
\]

Once again we interpolate in the vertical direction by forming the linear combination

\[
\frac{y - y_1}{y_3 - y_1} \quad \frac{x-x_1}{y_3 - y_1}
\]

which reduces to

\[
\frac{1}{y_3 - y_1} [(y - y_1)z_4 + (y_3 - y)z_3].
\]

The weights for the horizontal and vertical interpolants are taken to be

\[
\frac{L(y_3 - y_1)}{M(x-x_1) + L(y_3 - y_1)} \quad \text{and} \quad \frac{M(x-x_1)}{M(x-x_1) + L(y_3 - y_1)}
\]

respectively. Thus, we have obtained

\[
z = \frac{1}{M(x-x_1) + L(y_3 - y_1)} \left\{ L[(y_3 - y_1)z_3 + (y - y_1)(z_4 - z_3) + (y_3 - y)(z_3 - z_3)]
\right.
\]

\[
+ \frac{M(x-x_1)}{y_3 - y_1} [(y - y_1)z_4 + (y_3 - y)z_4] \right\}
\]

as an interpolant analogous to Equation (1). The corresponding formulas for other stencils having a single missing boundary point can be written in a similar manner.

RESULTS

The pictures, except for computer generated Picture 1, are the results of processing a file on 7-channel magnetic tape. A microphotograph of brain tissue was
scanned and sixteen gray levels recorded on the tape. (A reproduction of the original photograph from which this tape was made can be found on page 8 of Reference 1.) The tape was processed on an IBM 360/40 (with FORTRAN under Version 13 of OS) and the resulting gray level values were output on 9-channel tape. The output tapes were read and recorded on film by a local vendor. Picture 2 is the recording of the original file. It consists of 1100 rows of 750 elements of the scanned photograph.

The local areas within the picture to be manipulated were outlined manually, and coordinates, $B_k$, of their boundary points were punched on cards. Picture 3 is a reproduction of Picture 2 with the two objects referred to in this discussion outlined.

Picture 4 is a reproduction after operations on a small section (300 rows) of the original image. Object 2 has been rotated counterclockwise 30 degrees about the indicated pivot point, $P$.

Picture 5 is a reproduction after several operations. Object 2 has been rotated 180° and translated to a position in the lower half of the picture. Object 1, has been rotated 60° and moved to the right. The areas formerly occupied by both objects have been filled in by the hole-filling technique described.

Owing to the relatively poor quality of reproduction of the pictures that have thus far been generated, it is not possible to speculate on the extent to which more precise algorithms would be required for any given implementation. In the geometric butterfly of Picture 1, the introduction of cyclical noise of both horizontal
and vertical nature makes it difficult to determine the extent to which staircasing was avoided by the weighting scheme. Yet in Picture 4, the detail in the original picture has been retained even to the fine, well-defined curved line on the right of the object. No staircasing, haloing, or other distortion in the surrounding area is observed.

CONCLUSION

Up to this point in time the literature demonstrating the effect of local manipulations of pictures has been limited, and comparative evaluation of our results is not possible. It has been demonstrated here that one can insert or delete items from a picture or change their appearance, position, and orientation using standard programming techniques and modest computing equipment, and that the modified pictures will be of adequate quality to support a variety of experiments. This paper treats the subject from a technique-oriented standpoint and establishes the validity of the techniques.

REFERENCES

1 B S LIPKIN A ROSENFELD Editors
   *Picture processing and psychopictorics*
2 A ROSENFELD
   *Picture processing by computer*
3 I H BARKDOLL B L McGLAMERY
   *An on-line image processing system*
   Proceedings of the ACM 23rd National Conference
   Washington DC August 1968 pp 705-716
4 R NATHAN
   *Digital video-data handling*
   NASA Tech Report No 32-877 Jet Propulsion Laboratory
   Pasadena California January 5 1966
5 R H SELZER
   *Digital computer processing of x-ray photographs*
   NASA Tech Report No 32-1028 Jet Propulsion Laboratory
   Pasadena California November 15 1966

From the collection of the Computer History Museum (www.computerhistory.org)