ACKNOWLEDGMENT

The authors gratefully acknowledge the help of Ms. Shelly Rowe in preparing this paper.

REFERENCES

[1] A. Rosenfeld, R. A. Hummel, and S. W. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, pp. 420–433, June 1976.

[2] A. Rosenfeld, "Iterative methods in image analysis," in *Proc. IEEE Conf. Pattern Recognition and Image Processing*, Troy, NY, June 1977, pp. 14–18.

[3] S. W. Zucker, R. A. Hummel, and A. Rosenfeld, "An application of relaxation labeling to line and curve enhancement," *IEEE Trans. Comput.*, vol. C-26, pp. 394–403, April 1976.

[4] R. A. Hummel and A. Rosenfeld, "Relaxation processes for scene labeling," Computer Science Center, Univ. Maryland, College Park, Rep. TR-562, Aug. 1977.

Iterative Histogram Modification, 2

SHMUEL PELEG

***Abstract*—Histogram peaks can be sharpened using an iterative process in which large bins grow at the expense of nearby smaller bins. The modified histogram will consist of a few spikes corresponding to the peaks of the original histogram. The image corresponding to the modified histogram is often almost undistinguishable from the original image. The small number of different gray levels in that image can be used to facilitate approximating or segmenting it.**

I. INTRODUCTION

The histogram of an image is the discrete distribution function of the gray levels of the pixels in it. This correspondence describes a process for sharpening peaks on an image's histogram. It supplements preliminary work by Rosenfeld and Davis [1]. The process thins each peak on the original histogram into a spike. The image, corresponding to the modified histogram, has only a few gray levels. These gray levels correspond to the spikes in the modified histogram. The process can also generate a spike from the "shoulder" of a peak. Such shoulders are created by small peaks close to bigger ones; the process provides a cheap method of discovering such hidden peaks. The resulting image is a mapping of the original image into very few gray levels corresponding to the spikes found. This mapping provides an initial segmentation of the image, each segment corresponding to a spike in the histogram. Even though the modified image generally consists of very few gray levels, no deterioration in the image detail is seen. This should make possible efficient coding of the image without noticeable deterioration in its quality.

II. THE ALGORITHM

The algorithm described below operates on a one-dimensional histogram, but has a natural generalization to any number of dimensions. Thus it could be used to process three-dimensional color histograms or histograms based on additional pixel properties besides gray level. (This generalization was suggested by E. Riseman in a personal communication.)

Manuscript received December 23, 1977; revised March 6, 1978. This work was supported by the U.S. Army Night Vision Laboratory under Contract DAAG53-76C-0138 (ARPA Order 3206).

The author is with the Computer Science Center, University of Maryland, College Park, MD 20742.

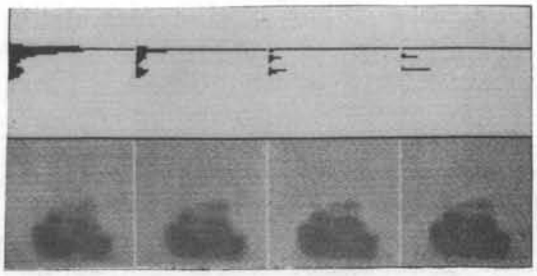

Fig. 1. Iterations 0, 1, 2, 4 of peak sharpening process.

Let B_i be the number of pixels having gray level i. For each histogram bin i, the neighboring $2r$ (an input parameter) bins $i \pm j$ on each side of i ($j = 1, 2, \cdots, r$) are examined. If B_i is greater than the average A of $B_{i+1}, \cdots, B_{i+r}$ (and similarly on the other side of i), we compute the ratio $X = (B_i - A)/B_i$, which specifies the fraction of pixels whose gray levels will be shifted towards i. Then the following gray-level changes are executed:

$$B_{i+r} \cdot X \text{ from } i + r \text{ to } i + r - 1;$$
$$B_{i+r-1} \cdot X \text{ from } i + r - 1 \text{ to } i + r - 2; \cdots;$$
$$B_{i+1} \cdot X \text{ from } i + 1 \text{ to } i.$$

The entire process is then iterated.

In order to minimize the changes in gray levels and to preserve their original order (i.e., to preserve "darker than" and "lighter than" relations), a "history" of pixel movement is kept. A matrix H is created in which element $H(\alpha, \beta)$ indicates the number of pixels currently at gray level α that had original gray level β. Initially,

$$H(i, j) = \begin{cases} 0, & i \neq j \\ \text{number of pixels with gray level } i, & i = j. \end{cases}$$

The algorithm performs gray-level changes on H only (*not* on the image). When transferring pixels from gray level α to gray level β, the pixels transferred first are those whose origin is closest to β. Finally, the image is transformed by changing $H(\alpha, \beta)$ pixels from gray level β to gray level α.

III. BANDWIDTH COMPRESSION

An immediate application of the algorithm described previously is to provide a segmentation of the image into a few gray levels. Simple images such as tanks (see Figs. 1, 6, 7) can be represented by three or four gray levels, thus reducing the number of bits per pixel from six to two. Figs. 2–5 show that even more complicated images can be represented by about eight distinct gray levels.

Efficient encoding schemes can be used to further improve compression. From the final histogram we can derive a Huffman coding [2] for the image. This coding gives about 1.4 bits per pixel for the tank images in Figs. 6 and 7 and 2.1 bits per pixel for Figs. 4 and 5. Run length coding can also be used, since the reduction in the number of gray levels favors longer runs.

IV. EXAMPLES

Fig. 1 shows the steps in the creation of spikes from the original histogram. Displayed are the original image and its histogram and the images produced after one, two, and four iterations. Figs. 2–7 each consist of an original image and, to its right, the images

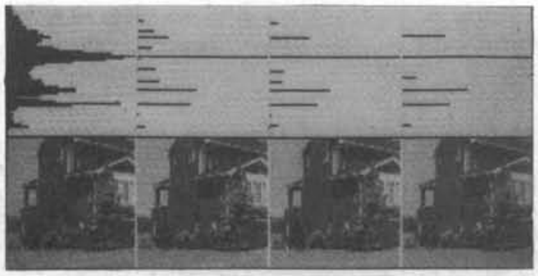

Fig. 2. Result of four iterations of peak sharpening process using neighborhood sizes of 2, 3, and 4. The image on the extreme left is the original.

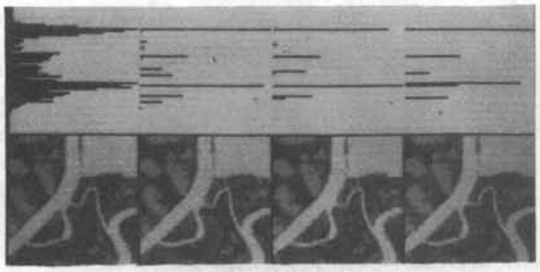

Fig. 3. Same as Fig. 2.

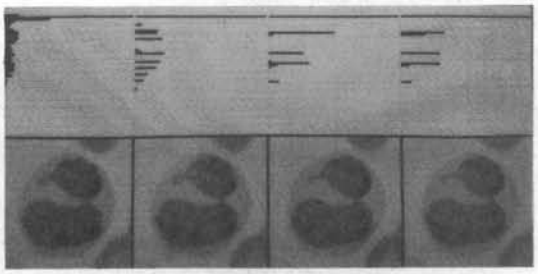

Fig. 4. Same as Fig. 2.

Fig. 5. Same as Fig. 2.

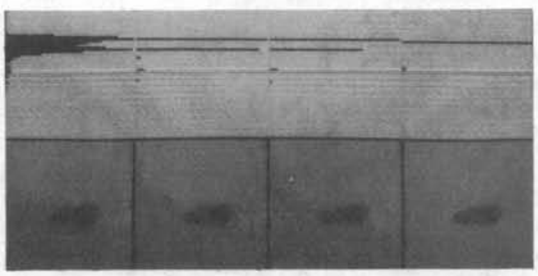

Fig. 6. Result of four iterations of peak sharpening process using neighborhood sizes of 2, 4, and 5. The image on the extreme left is the original.

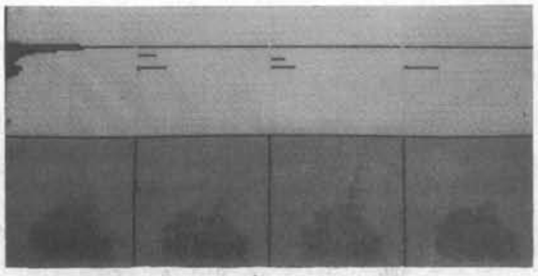

Fig. 7. Same as Fig. 2.

obtained after four iterations but with varying sizes of the neighborhood of each bin that was considered ($r = 2$, 3, and 4).

From Figs. 2–5 it can be seen that when a bin considers only a small neighborhood, high-frequency characteristics of the histogram are emphasized. Increasing the width of the neighborhood has a smoothing effect, and the high-frequency effects disappear. In Figs. 6 and 7 the target regions are successfully extracted.

V. Concluding Remarks

This correspondence has presented further results on histogram peak sharpening and its uses in image compression and segmentation. The method used is generally similar to that of Davis [1], but is improved in that it preserves the order of the gray levels, while producing sharper and better centered peaks. It also makes no use of Davis' preprocessing scheme of discarding points having high edge values. The examples show that this method should be useful for a wide variety of types of images.

Acknowledgment

The author gratefully acknowledges the help of Ms. Shelly Rowe in preparing this paper.

References

[1] A. Rosenfeld and L. S. Davis, "Iterative histogram modification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 300–302, Apr. 1978.
[2] D. Knuth, *The Art of Computer Programming*, vol. 3. Reading, MA: Addison-Wesley, 1975.

Feature-Ordering Criteria for Composite Classes Arising in Invariant Pattern Recognition

SARGUR N. SRIHARI, MEMBER, IEEE

***Abstract*—When decision rule invariance over complex transformations is desired and invariant features are absent, pattern classes are best represented as composite. An investigation of measurement-ordering criteria for discrimination between composite classes arising from continuous transformations is presented. Efficient methods of feature subset ordering based on the criteria are discussed with**

Manuscript received September 29, 1977; revised December 23, 1977. This work was supported in part by the Air Force Office of Scientific Research under Grant 74-2611. This paper is based on part of a dissertation submitted to the Department of Computer and Information Science, Ohio State University, Columbus, in partial fulfillment of the requirements for the Ph.D. degree.

The author is with the Computer Science Section, Department of Mathematics, Wayne State University, Detroit, MI 48202.