THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 56, No. 5, May–June 1977
Printed in U.S.A.

Techniques for Coding Dithered Two-Level Pictures

By A. N. NETRAVALI, F. W. MOUNTS, and J.-D. BEYER

(Manuscript received October 14, 1976)

This paper considers several methods for the efficient coding of two-level pictures dithered to give the appearance of multiple amplitude levels. In the dithering technique, a multilevel image signal is compared with a position-dependent set of thresholds (called a dither matrix), and, if the image value exceeds the threshold, the two-level output signal is taken to be "white," otherwise it is taken to be "black." Spatial correlation present in the original image is not preserved in the two-level picture due to spatial variation of the value of the threshold; and, therefore, standard techniques for coding two-level pictures, such as run-length coding, lose their efficiency. We show how some of our recently developed techniques for coding two-level pictures can be modified to code two-level dithered images. Our computer simulations on a few representative two-level dithered pictures indicate that an entropy between 0.2 to 0.3 bit/pel is possible using our technique. A comparison with some recently proposed techniques by Judice indicates that those schemes result in about 10 to 60 percent higher entropy than our schemes.

I. INTRODUCTION

Techniques for representing the entire gray scale of a picture by only two levels have been receiving considerable attention[1–10] because many devices are limited to recording or displaying two-level signals. Although these techniques may differ in specific algorithms, they provide the subjective illusion of a wide range of gray shades by controlling the proportion of picture elements in a neighborhood that are in the "on" state.

One of the techniques (called "dither" and studied in detail by Limb,[2] Lippel and Kurland,[3] and Judice et al.[9–10]) consists of comparing the multilevel input image signal with a position-dependent set of thresholds and setting only those picture elements to "white" (or 1) where the image

PEL COLUMN ℓ	ℓ+1	ℓ+2	ℓ+3	
0	128	32	160	LINE m
192	64	224	96	LINE m + 1
48	176	16	144	LINE m + 2
240	112	208	80	LINE m + 3

Fig. 1—A 4 by 4 dither matrix to be used for images with 8-bit quantized samples.

input signal exceeds the threshold. A square matrix of threshold values (elements of a "dither matrix") is repeated as a regular array to provide a threshold pattern for the entire image. Subjective effects of gray shades are achieved by using a dither matrix. The 4 by 4 dither matrix used by Judice et al.[9] for an input image having gray levels between 0 and 255 is shown in Fig. 1. The values of the adjacent elements of the dither matrix were chosen to take advantage of the spatial low-pass filtering present in the human visual system and, at the same time, to reproduce edges accurately and to avoid objectionable patterns. When the input image intensity is compared with spatially varying thresholds, a large amount of spatial correlation present in the input image is suppressed. This loss of correlation in the dithered two-level image makes some of the standard methods of coding two-level signals, such as run-length coding,[11] inefficient.

Several modifications of the standard techniques are possible. Judice has considered two such modifications. In one scheme,[12] called bit interleaving, runs of picture elements corresponding to equal or near equal elements of the dither matrix are coded using standard techniques. The other scheme,[13] called pattern matching, assigns a code to two-dimensional bit patterns of the dithered image and relies for bit-rate reduction on the fact that all possible bit patterns do not occur with the same frequency.

In our schemes, the "state" of the coder when an element is to be coded is a function of the already transmitted values of surrounding elements and the dither threshold at the element to be coded. The value that minimizes the probability of prediction error conditioned on a state is the predicted value of the element to be coded. This is an extension of the predictive coding for two-level pictures discussed in Ref. 14. In one of our schemes, we code run-lengths of the prediction error. In our other schemes, we change the relative order of the picture elements along a scan line in such a way as to increase the average run-length of the black and/or white elements and then transmit the run-lengths. These schemes are described in detail in Section 2.3.

We have investigated the efficiency of our techniques by computer simulation on a few representative 4- by 5-inch pictures, scanned with an array of 512 by 512. Using run-length coding of prediction errors, it is possible to decrease the bit rate to about 0.22 to 0.42 bit/pel. "Good-bad" state ordering performs best among all the ordering schemes that we have considered, and it brings the bit rates down to between 0.20 to 0.30 bit/pel. As a comparison, results from simulation of the same pictures by Judice show a bit rate of between 0.22 to 0.48 bit/pel.

II. CODING ALGORITHMS

In this section, we describe our coding algorithms in detail and present results of our computer simulations. The pictures used for computer simulation are shown in Fig. 2. These pictures were 4 inches by 5 inches and were scanned with 512 samples along a line and 512 lines. Each picture element was digitized by a uniform PCM coder to an accuracy of eight bits (256 levels). The digitized signal was dithered by using the dither matrix shown in Fig. 1. The coding algorithms were applied to the dithered images. As a measure of performance, we used the sample first-order entropy of run-length statistics. We computed the average black and white run-lengths and the entropy of black and white runs using, for example, the formula

$$E_w = -\sum \frac{n_i}{N} \log_2 \frac{n_i}{N},$$

where E_w is the entropy of white run-lengths, n_i is the number of white runs of length i, and N is the total number of white runs. Using these, and eq. (1), we computed the entropy in bits/pel by:

$$E = \frac{E_w N_w + E_b N_b}{r_w N_w + r_b N_b} \tag{1}$$

$$= \frac{E_w + E_b}{r_w + r_b} \qquad (\text{when } N_w \cong N_b),$$

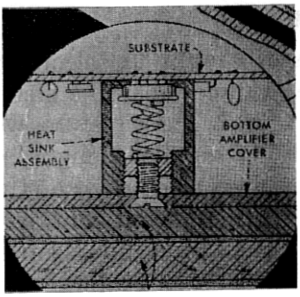
where

E_b is the entropy of the black run statistics (bits/run)
r_w is the average white run-length (pels/run)
r_b is the average black run-length (pels/run)
N_w, N_b are the number of white and black runs, respectively
E is the entropy in bits/pel.

2.1 Prediction algorithm

Consider a dithered picture element S_{ij} at position (i,j) where the dither matrix has value D_{ij}. To develop a predictor for S_{ij}, consider

(a)



(b)

(c)

Fig. 2—Dithered images used for computer simulation of coding algorithms. (a) Karen. (b) Engineering drawing. (c) House.

surrounding elements W, X, Y, Z as shown in Fig. 3. Note that elements W, X, Y, Z and the value of the dither matrix D_{ij} are known to the receiver when decoding the signal at point (i,j). The state associated with S_{ij} is defined as the five-tuple

$$Q = (D_{ij}, W, X, Y, Z). \tag{2}$$

Since we are using a 4 by 4 dither matrix, D_{ij} can have 16 different values. Each W, X, Y, Z can have two values, and, therefore, the number of possible states is 256. Let these be denoted by the set $\{Q_k\}$, $k = 1, \cdots, 256$.

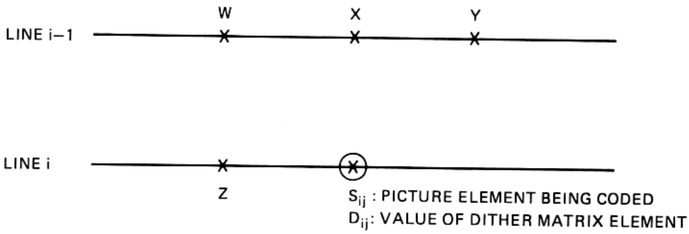


Fig. 3—Configuration for definition of state.

Then a predictor code book is given by:

$$C(Q_k) = \text{“0” if } P(S_{ij} = \text{“0”}/Q = Q_k) \geqq 0.5$$
$$= \text{“1”} \qquad \text{otherwise,} \tag{3}$$

where $C(\cdot)$ is the predictor code book and $P(\cdot|\cdot)$ is the experimentally determined conditional probability that a picture element has a value of "0" or "1" given the state Q_k. Thus, the predictor thus depends upon the previously transmitted values in a neighborhood and some partial information about the present picture element known both to the transmitter and the receiver in the form of the value of the dither matrix. The code book can be designed for each picture and transmitted before actual picture transmission,* or it can be taken to be an average for a class of pictures. A portion of the code book for each of the three pictures that we considered is shown in Table I, where we assign the state number, k, by computing formula (4), and show the number of occurrences of each state and the probability of error.

$$k = D_{ij} + 8W + 4X + 2Y + Z + 1. \tag{4}$$

We note that the probability of prediction being in error is always less than 0.5 due to our method of prediction. The code book defined by eq. (3) does vary from picture to picture. We examine the effects of such variation in a later section.

2.2 Run-length coding of prediction errors

In this technique, we code the run-lengths of the prediction errors along a scan line using the appropriate code book for each picture. The entropy of the run-length statistics is given in Table II. It varies between 0.22 to 0.42 bit/pel. To evaluate the effects of the variation of code book with respect to pictures, we used the code book of one picture for the prediction of another. The resulting entropies of the run-lengths of

* Only 256 bits are needed to transmit a code book; since there are $(512)^2$ picture elements per picture, transmission of one code book per picture corresponds to an additional 0.001 bit/pel.

Table I—State dependent predictors for different pictures. Also shown is the probability of prediction error and the number of elements in state. Only the first 54 states are shown for brevity.

State Definition						Picture: Karen			Picture: Engineering Drawing			Picture: House		
State Number	Dither Matrix (D_{ij})	W	X	Y	Z	Total Elements in State	Pre-diction	Probability of Prediction Error	Total Elements in State	Pre-diction	Probability of Prediction Error	Total Elements in State	Pre-diction	Probability of Prediction Error
1	0	0	0	0	0	4026	1	0.018	3197	1	0.041	6225	1	0.000
2	0	0	0	0	1	12	1	0.000	197	1	0.000	0	0	0.000
3	0	0	0	1	0	77	1	0.000	760	1	0.011	32	1	0.000
4	0	0	0	1	1	2	1	0.000	97	1	0.000	0	0	0.000
5	0	0	1	0	0	0	0	0.000	0	0	0.000	0	0	0.000
6	0	0	1	0	1	0	0	0.000	0	0	0.000	0	0	0.000
7	0	0	1	1	0	0	0	0.000	0	0	0.000	0	0	0.000
8	0	0	1	1	1	0	0	0.000	0	0	0.000	0	0	0.000
9	0	1	0	0	0	2590	1	0.000	1489	1	0.003	5665	1	0.000
10	0	1	0	0	1	43	1	0.000	429	1	0.000	0	0	0.000
11	0	1	0	1	0	6121	1	0.000	2342	1	0.002	4204	1	0.000
12	0	1	0	1	1	3258	1	0.000	7618	1	0.000	3	1	0.000
13	0	1	1	0	0	0	0	0.000	0	0	0.000	0	0	0.000
14	0	1	1	0	1	0	0	0.000	0	0	0.000	0	0	0.000
15	0	1	1	1	0	0	0	0.000	0	0	0.000	0	0	0.000
16	0	1	1	1	1	0	0	0.000	0	0	0.000	0	0	0.000
17	16	0	0	0	0	2938	1	0.279	3259	1	0.417	4217	1	0.047
18	16	0	0	0	1	4	1	0.000	108	1	0.000	0	0	0.000
19	16	0	0	1	0	61	1	0.000	785	1	0.135	10	1	0.000
20	16	0	0	1	1	0	0	0.000	32	1	0.000	0	0	0.000
21	16	0	1	0	0	0	0	0.000	2	0	0.000	0	0	0.000
22	16	0	1	0	1	0	0	0.000	0	0	0.000	0	0	0.000

23	16	0	1	1	0	0	0	0.000	0	0	0.000	0	0	0.000
24	16	0	1	1	1	1	1	0.000	0	0	0.000	0	0	0.000
25	16	1	0	0	0	2302	1	0.003	1335	1	0.061	5193	1	0.000
26	16	1	0	0	1	40	1	0.025	267	1	0.011	0	0	0.000
27	16	1	0	1	0	9185	1	0.006	5154	1	0.009	6964	1	0.000
28	16	1	0	1	1	1811	1	0.007	5442	1	0.001	0	0	0.000
29	16	1	1	0	0	0	0	0.000	0	0	0.000	0	0	0.000
30	16	1	1	0	1	0	0	0.000	0	0	0.000	0	0	0.000
31	16	1	1	1	0	3	1	0.000	0	0	0.000	0	0	0.000
32	16	1	1	1	1	39	1	0.000	0	0	0.000	0	0	0.000
33	32	0	0	0	0	4038	1	0.403	3096	0	0.193	6317	1	0.113
34	32	0	0	0	1	24	1	0.000	396	1	0.063	9	1	0.000
35	32	0	0	1	0	2562	1	0.002	1327	1	0.251	5641	1	0.000
36	32	0	0	1	1	70	1	0.000	587	1	0.027	24	1	0.000
37	32	0	1	0	0	0	0	0.000	0	0	0.000	0	0	0.000
38	32	0	1	0	1	0	0	0.000	1	1	0.000	0	0	0.000
39	32	0	1	1	0	0	0	0.000	0	0	0.000	0	0	0.000
40	32	0	1	1	1	2	1	0.000	0	0	0.000	0	0	0.000
41	32	1	0	0	0	33	1	0.091	333	1	0.339	42	1	0.000
42	32	1	0	0	1	47	1	0.021	513	1	0.226	16	1	0.000
43	32	1	0	1	0	1996	1	0.001	1187	1	0.127	2319	1	0.000
44	32	1	0	1	1	7090	1	0.000	8774	1	0.002	1888	1	0.000
45	32	1	1	0	0	0	0	0.000	0	0	0.000	0	0	0.000
46	32	1	1	0	1	0	0	0.000	0	0	0.000	0	0	0.000
47	32	1	1	1	0	15	1	0.000	1	1	0.000	0	0	0.000
48	32	1	1	1	1	379	1	0.000	41	1	0.000	0	0	0.000
49	48	0	0	0	0	2943	0	0.271	3024	0	0.082	4136	0	0.487
50	48	0	0	0	1	9	1	0.000	229	1	0.066	0	0	0.000
51	48	0	0	1	0	2274	1	0.047	1349	1	0.375	5146	1	0.008
52	48	0	0	1	1	6	1	0.000	265	1	0.049	0	0	0.000
53	48	0	1	0	0	0	0	0.000	0	0	0.000	0	0	0.000
54	48	0	1	0	1	0	0	0.000	0	0	0.000	0	0	0.000

prediction errors are also shown in Table II. As expected, when a code book is used that is not specifically matched to the picture, there is a loss of coding efficiency. However, by using the code book for the picture of Fig. 2a for all the pictures, there is a maximum loss of only 0.08 bit/pel. Thus, although better results in terms of entropy can be obtained by using a matched code book, it appears possible to use a general code book that will not degrade the performance significantly.

2.3 Ordering techniques

These ordering techniques are extensions of our techniques for two-level pictures.[15] In these techniques, we order either the elements or the prediction errors of the present line using a reference signal available to both the transmitter and the receiver—for example, the elements of the previous line.

To illustrate the technique, consider a memory containing 512 cells (equal to the number of elements per line). Suppose the cells of this memory are numbered from 1 to 512. If the first element of the previous line is white, then we put the prediction error for the first element of the present line in memory cell 1; if the first element of the previous line is black, then we put the prediction error for the first element of the present line in memory cell 512. We continue in this manner: the prediction error of the ith element of the present line is put in the unfilled memory cell of smallest index or of largest index depending on whether the ith element of the previous line is white or black. When the memory is filled, its cells are read in numerical order and the contents are run-length encoded. It is easy to see that the present line can be uniquely reconstructed from the knowledge of the run-lengths of the ordered line, since the ordering information is known to the receiver.

The efficiency of the simple ordering technique discussed above is given in Table II. It is seen that due to the process of dithering, much of the efficiency of ordering is lost. We overcome the effects of dithering by using an ordering technique based on the "goodness" of the state. We divide the states defined in eq. (2) into two groups. States that have a high probability of correct prediction are called "good" states and the remaining are called "bad" states. Our algorithms can be described as follows: we first evaluate the prediction error for a particular element of the present line, and then, if the state is "good," we put the prediction error in the unfilled memory cell with the smallest index; if the state is "bad," the prediction error is put in the unfilled memory cell with the largest index. Having ordered the prediction errors, we run-length code them as before.

It is easy to see that the line of picture data can be uniquely reconstructed from the coded run-lengths of the prediction error. The entropies obtained by this scheme are given in Table II. The criterion of

Table II—Entropy comparisons for different coding algorithms

	Algorithms			Entropy (Bit/Pel) Picture I "Karen"	Picture II "Engineering Drawing"	Picture III "House"
	Coding Algorithms					
I.	Run-length coding of prediction errors using code book of picture I			0.33	0.47	0.30
	using code book of picture II			0.38	0.42	0.44
	using code book of picture III			0.45	0.65	0.22
	Ordering Algorithms					
II.	Run-length coding of ordered elements 'X' with respect to elements 'Y'					
	(a) X: Samples of present line; Y: prediction of present sample			0.48	0.63	0.62
	(b) X: Prediction errors of present line; Y: samples of previous line			0.37	0.44	0.29
	(c) X: Prediction errors of present line; Y: good-bad states	Goodness threshold:	0.02	0.29	0.39	0.22
			0.05	0.26	0.30	0.20
			0.1	0.29	0.30	0.22
			0.2	0.31	0.40	0.22
	(d) X: Prediction errors of present line; Y: good-bad states; code book of picture I	Goodness threshold:	0.05	0.26	0.40	0.23
	Algorithms of Judice					
III.	One-dimensional bit interleaving			0.37	0.50	0.27
IV.	Two-dimensional bit interleaving			0.33	0.48	0.22
V.	Pattern matching			0.40	0.61	0.28

"goodness" of a state is determined by a threshold on the probability of error. Our simulations indicate that a threshold of 5-percent probability of error does better than thresholds of 2 percent, 10 percent, and 20 percent for all three pictures. The entropy is reduced to between 0.20 to 0.30 bit/pel. The advantage of ordering, obtained by comparing these entropies with those obtained from run-length coding of the prediction errors, is about 9 to 29 percent. We also considered the use of a prediction code book from Fig. 2 for all pictures. This resulted in a small increase in entropy over that obtained by using a matched code book.

2.4 Comparisons with the algorithms of Judice

We mentioned earlier that Judice has recently given two algorithms for coding of dithered two-level pictures. In one of them,[12] runs of picture elements corresponding to the same values of the dither matrix are run-length coded. He has discussed this scheme in one dimension (called one-dimensional bit interleaving) as well as two dimensions (called two-dimensional bit interleaving). The bit rates obtained by these two schemes are reproduced from Ref. 12 in Table II. A bit rate of 0.22 to 0.48 bit/pel is possible with these schemes. This is about 10- to 60-percent higher than the entropies obtainable from our "good-bad" state ordering schemes. The other scheme discussed by Judice et al.,[13] called pattern matching, assigns a code to two-dimensional bit patterns of the dithered images. The entropies obtained in this case, also reproduced in Table II from Ref. 13, are generally higher than those achieved by two-dimensional bit interleaving. Thus, our "good-bad" state ordering schemes perform more efficiently than the schemes proposed by Judice.

III. DISCUSSION AND SUMMARY

We have described schemes for efficient coding of dithered two-level signals. We started with the description of a predictor that depends upon already transmitted neighboring elements and the value of the dither matrix at the element being predicted. The predictor minimizes the probability of prediction being in error. We found that the run-length coding of the prediction errors brought the bit rate down to about 0.22 to 0.42 bit/pel for the three pictures we used for simulation. We then discussed several ordering algorithms in which the relative order of transmission of the picture elements in a scan line is changed to increase the average lengths of black and white runs. We found that the ordering scheme based on goodness of the state decreased the bit rate to 0.20 to 0.30 bit/pel. Finally, we compared our results with those obtained by Judice and found that his schemes gave about 10 to 60 percent higher entropy.

It should be mentioned that this is not a definitive coding system study. We have not considered many important factors crucial to the success of any coding system, such as run-length codes and their picture dependence, and the effect of transmission errors.

IV. ACKNOWLEDGMENTS

We would like to thank C. N. Judice of Bell Laboratories for many helpful discussions and for supplying us the picture data used in this paper.

REFERENCES

1. P. Stucki, "Generation of Grey Tones by Computer for Simulation of Visual Information Systems," IEEE Trans. Comput., *18* (July 1969), pp. 642–643.
2. J. O. Limb, "Design of Dither Waveforms for Quantized Visual Signals," B.S.T.J., *48*, No. 7 (September 1969), pp. 2555–2582.
3. B. Lippel and M. Kurland, "The Effect of Dither on Luminance Quantization of Pictures," IEEE Trans. Commun. Technol., *COM-19*, No. 6 (December 1971), pp. 879–888.
4. K. Knowlton and L. Harmon, "Computer-Produced Grey Scales," Computer Graphics and Image Processing, *1*, 1972, pp. 1–20.
5. B. E. Bayer, "An Optimum Method for Two-Level Rendition of Continuous-Tone Pictures," IEEE International Conference on Communications, Conference Record, 1973, pp. (26-11)–(26-15).
6. S. Arneman and M. Tasto, "Generating Halftone Pictures on Graphic Terminals Using Run-Length Coding," Computer Graphics and Image Processing, *2*, 1972, pp. 1–11.
7. J. F. Jarvis, C. N. Judice, and W. H. Ninke, "A Survey of Techniques for the Display of Continuous Tone Pictures on Bilevel Displays," Computer Graphics and Image Processing, *5*, No. 1 (March 1976), pp. 13–40.
8. J. F. Jarvis and C. S. Roberts, "A New Technique for the Bilevel Rendition of Continuous Tone Pictures," IEEE Trans. Commun., *COM-24* (August 1976), pp. 891–898.
9. C. N. Judice, J. F. Jarvis, and W. H. Ninke, "Using Ordered Dither to Display Continuous Tone Pictures on an AC Plasma Panel," Proc. Soc. Inform. Display, *15/4* (Fourth Quarter 1974), pp. 161–164.
10. C. N. Judice, "Dithervision—A Collection of Techniques for Displaying Continuous Tone Still-and-Animated Pictures on Bilevel Displays," International Symposium, Society for Information Display, 1975, pp. 32–33.
11. J. Capon, "A Probabilistic Model for Run-Length Coding of Pictures," IRE Trans. Inform. Theory, *IT-5* (December 1959), pp. 157–163.
12. C. N. Judice, "Data Reduction of Dither Coded Images by Bit Interleaving," Proc. Soc. Inform. Display, *17*, No. 2 (1976), pp. 91–99.
13. C. N. Judice, A. B. White, and R. L. Johnson, "Tranmission and Storage of a Dither Coded Image Using 2-D Pattern Matching," Proc. Soc. Inform. Display, *17*, No. 2 (1976), pp. 85–90.
14. J. S. Wholey, "The Coding of Pictorial Data," IRE Trans. Inform. Theory, *IT-7* (April 1961), pp. 99–104.
15. A. N. Netravali, F. W. Mounts, and E. G. Bowen, "Ordering Techniques for Coding of Two-Tone Facsimile Pictures," B.S.T.J., *55*, No. 9 (December, 1976), pp. 1539–1552.