INTRAFRAME AND INTERFRAME ADAPTIVE TRANSFORM CODING

Clifford Reader
Aeronutronic Ford Corporation
3939 Fabian Way, Palo Alto, California 94303

Abstract

This paper describes work performed as an extension of the techniques of transform coding with block quantization. These techniques have been very well defined theoretically with implementation of the predicted optimum results. The purpose of this work was to examine the techniques in practice and to define adaptive coding techniques which could be practically applied to the coding of still and moving pictures.

Introduction

The techniques of transform coding and block quantization for a single image are well established, the purpose of the work described in the first part of this paper was to examine in detail the application of these techniques in practice [1]. It is to be expected that there will be a variability of performance since any practical implementation will be rather removed from the theoretical optimum case with two dimensional Karhunen-Loeve sampling according to a known covariance and quantization based upon assumed statistics. In particular it is noted that the condition of stationarity of statistics will be violated and that for ease of implementation separable sub-optimum transformations - in this case the Slant - will be used over sub-blocks (16 x 16 pixels) of the image. A range of adaptive coding techniques is examined with practicality as well as coding efficiency being of importance.

The work is then continued into interframe coding. A coder is described which achieves a significant reduction in bit rate while not requiring the full frame memory of other interframe coders. The error performance of the coder is examined and static examples are presented. A discussion follows of the visual effect of the coder in real time together with the projected buffer requirements.

Intraframe Coding

Summary of Coding Procedure

The image is slant transformed in sub-blocks of 16 x 16 pixels. To perform quantization, the data is modelled as a Markov process and from this the variances of the transform domain samples are estimated. Coding bits are allocated in proportion to these variances and the samples are normalized in proportion to their estimated variances before being quantized according to the Max quantizer. Tuning of the process is possible by manipulation of the row and column correlations used to define the variances and with the constant of proportionality used in normalizing by the variances. The apparently arbitrary way in which tuning of the parameters influenced the coding error provided the motivation for this work.

Adaptive Coding 1 - The Amplitude Parameter

The constant of proportionality (C) in the normalization of the data is termed the amplitude parameter (AP). It was found that each sub-block of an image required a different AP for optimal coding (in mean squared error sense) and that although for most sub-blocks a good approximation could be made with one AP, in certain signficant cases a markedly different - higher - AP was required suggesting that an adaptive process was required. This was effected by making the constant C a function of the square root of the energy of the sub-block less the D.C. component, i.e., the A.C. energy. This is termed the amplitude factor AF. Thus C = AP x AF. The result was as follows. In non-adaptively coded images, the best result is obtained with an AP high enough to encode the problem sub-blocks and high frequency quantization noise then appears on the other sub-blocks. In the adaptive coding, the AF automatically compensates for the high energy problem sub-blocks such that one AP is more suitable for the whole image. This is seen as a freedom from noise in "flat" image areas and a slightly reduced mean square error. Figure 1.

Adaptive Coding 2 - Inter-Pixel Correlation

True horizontal and vertical correlations for natural images fall in the region of 0.91 to 0.96. Nevertheless, the effect was examined of using different correlations to define the model from which bit allocations and normalizing constants are derived. Results are summarized in Figure 2 where it can be seen that the lowest error is

APPLE-1037

encountered using far lower correlations than the real ones at the expense of increased sensitivity. What happens is that with the lower correlations, the available bits are spread futher into the transform domain and more samples are encoded. Figure 3. However each sample receives fewer bits and thus the position of the quantization levels (as fine tuned by the AP) is more critical. Correlations of 0.76 were selected as a compromise with the highly satisfactory results shown in Figure 4.

Figure 1

Comparison of Adaptive and Non-Adaptive Coding
Correlations (0.86; 0.86) 1.5 bits per Pixel



AP = 0.55 MSE = 0.000639



AP = 0.65 MSE = 0.000501



AP = 0.75 MSE = 0.000827

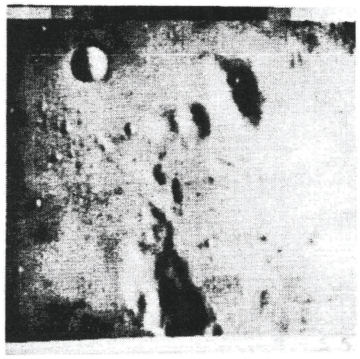
Square Root AC Energy
Adaptive Coding



AP = 0.85 MSE = 0.000724



AP = 0.85 MSE = 0.000588



AP = 0.85 MSE = 0.000818

Non-Adaptive Coding

Figure 2

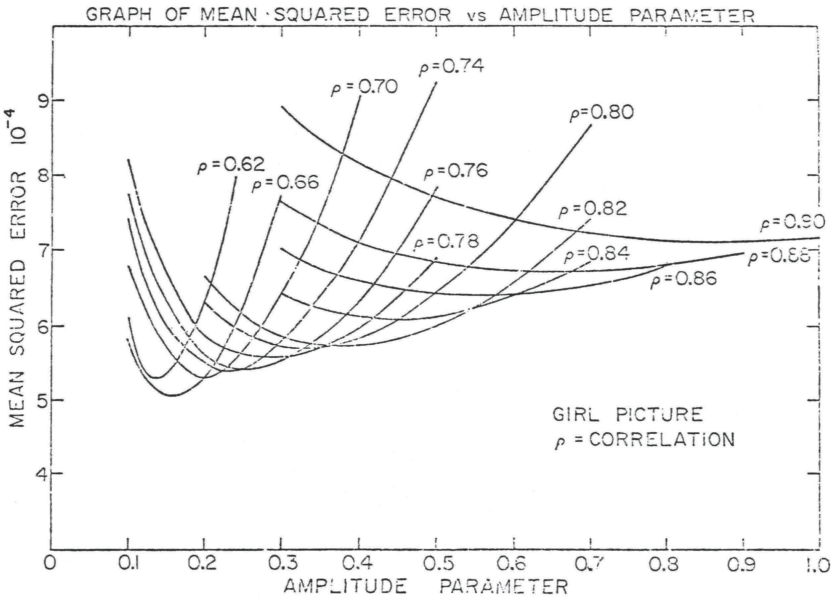
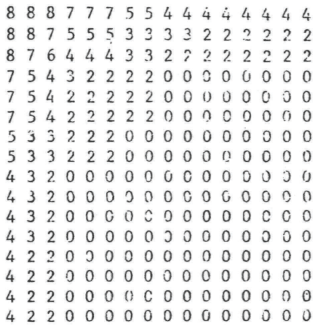
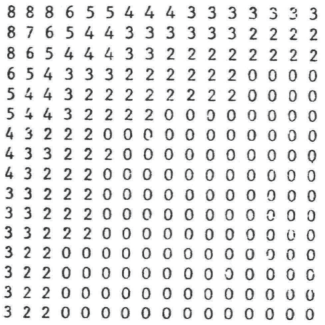


Figure 3
Bit Allocations



Correlations 0.86



Correlations 0.76

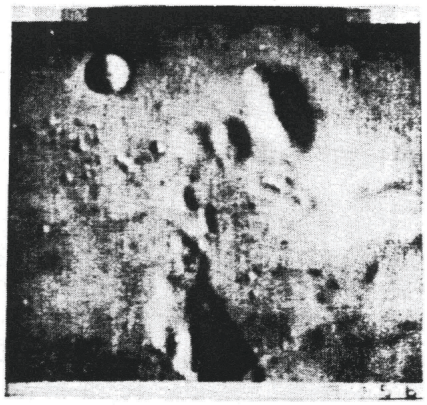
Figure 4
Coding with Square Root AC Energy Amplitude Factor
Correlations (0.76; 0.76)



AP = 0.25 MSE = 0.000544



AP = 0.30 MSE = 0.000446



AP = 0.35 MSE = 0.000749

1.5 Bits Per Pixel



AP = 0.25 MSE = 0.000863



AP = 0.3 MSE = 0.000757



AP = 0.35 MSE = 0.001023

1.0 Bits Per Pixel

Adaptive Coding 3 - Optimised Coding

The fine structure of the parameters required by each sub-block for optimum coding was examined by performing an experiment in which fifteen pairs of inter-pixel correlations (vertical and horizontal; Figure 5) were used to define bit allocations and normalizing constants which were applied in every combination (225) to every sub-block. Figure 6 shows sub-block maps in which the reference number (from Figure 5) is shown for the optimum combinations of bit allocations and normalizing constants. The resultant minimum mean squared error is shown in Figure 7 with the coded pictures shown in Figure 8. The pictures show good detail and edge rendition and it should be noted that even better coding could have resulted if correlations of less than 0.80 (type 1) had been allowed.

Figure 5

Correlation Reference No.	Correlations
1	0.80; 0.80
2	0.84; 0.84
3	0.88; 0.88
4	0.92; 0.92
5	0.96; 0.96
6	0.80; 0.92
7	0.80; 0.84
8	0.84; 0.92
9	0.88; 0.96
10	0.92; 0.96
11	0.92; 0.80
12	0.84; 0.80
13	0.92; 0.84
14	0.96; 0.88
15	0.96; 0.92

Figure 7

Sub-Block Map (Girl Image)

Optimum Mean Square Error

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	3	7	7	2	1	1	1	2	2
2	0	0	0	0	0	0	1	15	10	6	4	8	7	1	2	2	1
3	0	0	0	0	1	9	15	15	9	6	3	22	2	2	1	1
4	0	0	0	0	2	5	15	7	4	2	1	15	3	2	1	1
5	0	0	0	0	1	4	5	6	2	3	6	8	3	1	1	1
6	0	0	0	0	1	4	3	4	2	2	2	6	1	1	1	1
7	0	0	0	0	0	11	14	11	4	1	4	9	1	1	1	1
8	0	0	0	0	0	3	5	5	4	2	5	9	2	1	1	1
9	0	0	0	0	0	2	11	8	3	2	7	4	2	1	1	1
10	0	0	0	0	0	3	9	6	4	3	4	2	2	1	1	1
11	0	0	0	0	0	1	27	13	1	1	14	7	2	1	2	2
12	1	1	0	1	17	11	4	13	1	4	4	2	18	14	1	1
13	1	1	2	12	1	2	9	11	9	2	1	1	10	12	5	1
14	1	0	2	3	14	5	3	5	4	15	16	3	0	0	7	9
15	0	0	1	0	0	5	16	15	11	54	11	28	12	0	0	3
16	0	0	3	1	0	5	6	17	21	25	22	15	31	42	0	1

The Mean Square Error is coarsely approximated
1 ≡ 0.0001

Figure 6

Sub-Block Maps (Girl Image)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	1	1	1	1	1	7	1	8	1	1	12	1	1	1
2	1	1	1	1	1	1	7	1	1	(12)	12	8	1	1	1	1
3	1	1	1	1	1	(6)	(6)	1	1	6	(7)	3	1	1	1	
4	1	1	1	1	1	1	1	1	1	6	1	7	1	1	1	1
5	(3)	1	1	1	1	1	1	1	1	1	1	1	(11)	1	1	
6	11	1	1	1	1	(7)	1	1	1	1	1	1	(12)	1	1	
7	(11)	1	1	1	1	1	6	1	1	1	1	1	12	1	1	
8	(11)	1	1	1	1	1	6	1	12	1	1	1	1	1	1	
9	(13)	1	1	1	1	11	1	6	6	1	1	6	1	1	1	1
10	1	1	1	1	1	1	(6)	1	6	1	6	1	1	1	1	
11	(7)	1	1	1	1	1	7	1	(6)	1	(7)	1	1	1	1	
12	1	3	1	1	6	1	1	(12)	8	7	1	6	1	1	1	
13	12	3	1	(12)	1	1	1	1	1	1	(6)	6	1	(7)	1	
14	1	(12)	11	1	1	1	1	1	1	1	12	6	1	1	1	
15	1	1	1	1	(7)	1	1	(6)	1	1	6	7	(12)	1	1	
16	(6)	6	1	1	(6)	(7)	1	1	7	7	12	6	12	12	1	12

(i)

Sub-optimum Bit Allocations Types

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	1	1	1	1	1	10	10	10	1	1	3	1	1	1
2	1	1	1	1	1	1	5	1	1	(6)	3	6	15	1	1	1
3	1	1	1	1	1	1	1	(6)	8	8	6	(6)	4	1	1	1
4	1	1	1	1	1	1	(6)	1	1	1	1	3	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	(13)	1	1
6	1	1	1	1	1	(2)	1	1	1	1	1	1	1	(2)	1	1
7	1	1	1	1	1	1	8	(2)	(2)	1	1	8	1	3	1	1
8	1	1	1	1	1	5	6	3	6	1	1	1	1	1	1	1
9	1	1	1	1	1	15	1	6	8	1	1	1	1	1	1	1
10	1	1	1	1	1	1	6	(6)	1	1	1	1	1	1	1	1
11	(6)	1	1	1	1	1	6	6	1	1	(7)	1	1	1	1	1
12	4	10	1	1	9	6	1	(11)	5	8	(3)	8	8	1	1	1
13	10	4	1	(3)	1	1	1	1	1	1	1	6	1	(13)	1	1
14	5	(11)	11	11	1	1	1	1	1	1	8	1	1	1	1	1
15	1	1	11	1	(2)	1	1	(2)	1	4	8	10	(2)	1	1	1
16	(7)	1	1	1	(7)	(6)	1	1	9	3	3	8	5	5	1	15

(ii)

Sub-optimum companding Constant Types

Sub-blocks marked with a circle may be fairly well coded with bit allocations and companding constants of Type 1.

Adaptive Coding 4 - Zonal and Threshold Coding

The coding thus far described employs bit allocations in which those samples which are quantized fall inside a maximum variance zone. The coding error may be divided into the quantization error and the error resulting from those samples outside the zone which do not get coded at all. There will always be exceptional significant samples in the latter category and they could be coded by setting a magnitude threshold in the area outside the zone and run-length coding all samples exceeding such threshold. As an example, four bits were allocated to code each of such samples and, assuming that a line starting code were used, four bits were required to address each sample. The number of thresholded samples is shown by the sub-block maps of Figure 9 where it can be seen that a picture which has been zoned coded at 1.5 bits per pixel now requires 1.7 bits per pixel. In order to get back to 1.5 bits per pixel, the zonal coding

must be effected at 1.2 bits per pixel. The corresponding coded pictures are shown in Figure 10. It is considered that the result obtained at a total of 1.5 bits per pixel is not significantly improved and does not justify the work involved especially since channel error would be a problem.

Figure 8
Optimised Coding



Original



1.5 Bits Per Pixel
MSE = 0.000744



1.0 Bits Per Pixel
MSE = 0.000796

Figure 9

Sub-Block Maps (Girl Image)
The Number of Threshold Samples

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	5	15	7	1	0	0	0	1	1
2	0	0	0	0	0	0	20	19	14	6	11	11	0	1	3	0
3	0	0	0	0	0	10	19	33	25	12	4	21	0	0	0	0
4	0	0	0	0	2	8	21	8	6	2	0	17	3	2	0	1
5	0	0	0	0	0	5	11	11	0	3	14	10	4	0	0	0
6	0	0	0	0	0	7	1	1	0	2	1	10	0	0	0	1
7	0	0	0	0	0	14	23	18	4	1	2	13	0	0	0	0
8	0	0	0	0	0	2	7	8	7	0	6	10	1	0	0	0
9	0	0	0	0	0	1	10	9	3	2	9	0	1	0	0	0
10	0	0	0	0	0	3	11	9	6	4	0	0	0	0	0	0
11	0	0	0	0	0	0	29	26	0	0	23	7	0	0	0	0
12	0	0	0	0	22	23	9	20	0	5	3	3	28	31	0	1
13	0	0	2	21	1	0	12	17	11	1	1	0	18	21	6	0
14	0	0	0	6	25	9	1	2	8	22	30	0	0	0	2	20
15	0	0	0	0	0	6	30	22	21	45	19	41	19	0	0	2
16	0	0	2	0	0	7	13	22	28	29	39	20	55	37	0	0

(i)

1.7 bits per Pixel
Total Number of Threshold Samples = 6,008

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	7	21	14	8	0	0	0	1	4
2	0	0	0	0	0	1	36	26	18	14	23	20	0	1	3	2
3	0	0	0	0	0	18	28	47	34	18	10	43	2	0	0	0
4	0	0	0	0	2	14	31	20	11	5	1	26	6	2	0	1
5	0	0	0	0	1	12	18	21	8	6	22	23	7	0	1	0
6	0	0	0	0	0	16	4	6	6	5	2	18	0	0	1	1
7	0	0	0	0	0	27	35	35	18	4	9	25	0	0	0	0
8	0	0	0	0	0	7	9	14	14	2	15	15	1	0	0	0
9	0	0	0	0	0	2	24	21	6	3	20	0	1	0	0	0
10	0	0	0	0	0	7	18	18	8	5	0	1	1	0	0	0
11	0	0	0	0	0	4	39	39	0	0	38	18	0	1	0	2
12	1	0	0	0	32	34	12	33	0	11	8	6	43	43	0	2
13	0	0	5	38	1	0	23	29	27	5	2	8	32	33	16	0
14	1	0	0	11	40	19	4	13	17	40	47	4	0	0	13	31
15	0	0	3	0	0	20	44	34	36	70	33	59	36	0	0	5
16	0	0	10	1	0	20	25	37	48	47	55	29	75	60	0	0

(ii)

1.5 bits per Pixel
Total number of Threshold Samples = 10,532

Figure 10
Threshold Coding



Original



1.7 Bits MSE = 0.000533



1.5 Bits MSE = 0.000606

Adaptive Coding 5 - Zonal Coding

If the sub-block maps of Figures 6, 7 and 9 are studied, then it is seen that there is high correlation between them in terms of which sub-blocks display exceptional information. Two further maps are presented in Figure 11.

Figure 11

i

Sub-Block Map (Girl Image)

Sub-Optimum Companding Constant Types with Indication of High Amplitude Factor

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	1	1	1	1	1	[10]	[10]	[10]	1	1	3	1	1	1
2	1	1	1	1	1	1	(5)	1	1	6	3	6	15	1	1	1
3	1	1	1	1	1	1	7	6	8	8	6	[6]	4	1	1	1
4	1	1	1	1	1	1	6	1	1	1	1	[3]	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	13	1	1
6	1	1	1	1	1	2	1	1	1	1	1	1	1	2	1	1
7	1	1	1	1	1	1	[8]	2	2	1	1	8	1	3	1	1
8	1	1	1	1	1	[5]	6	3	6	1	1	1	1	1	1	1
9	1	1	1	1	1	15	1	6	8	1	1	1	1	1	1	1
10	1	1	1	1	1	1	6	6	1	1	1	1	1	1	1	1
11	6	1	1	1	1	1	[6]	[6]	1	1	7	1	1	1	1	1
12	4	10	1	1	[9]	6	1	11	[5]	8	3	8	[8]	1	1	1
13	10	4	1	3	1	1	1	1	1	1	1	6	1	13	1	1
14	5	11	11	11	1	1	1	1	1	1	[8]	10	1	1	1	1
15	1	1	11	1	2	1	1	2	1	(4)	8	(10)	2	1	1	1
16	7	1	1	1	7	6	1	1	(9)	[3]	[3]	8	(5)	(5)	1	15

◯ ≡ AF ⩾ 16
◯ ≡ AF ⩾ 9
□ ≡ AF ⩾ 6

ii

Sub-Block Map
AC Energy Amplitude Factors for Girl Image

0.1	0.2	0.1	0.0	0.1	0.5	0.1	9.0	13.	13.	1.1	0.2	1.5	1.2	0.2	0.3
0.1	0.2	0.1	0.0	0.0	0.9	16.	1.3	0.8	2.6	5.8	5.4	3.1	0.6	0.5	0.9
0.2	0.2	0.1	0.0	0.1	3.2	2.3	3.3	4.4	4.1	1.9	6.6	4.5	0.7	1.3	0.3
0.1	0.2	0.1	0.0	0.8	1.3	6.0	2.1	0.7	0.5	0.8	6.1	0.4	0.4	0.6	0.4
0.1	0.2	0.1	0.0	1.2	2.3	2.8	2.7	1.7	0.5	2.4	1.8	0.9	1.3	0.7	0.1
0.1	0.1	0.1	0.0	0.2	3.5	2.0	1.0	1.7	1.0	1.0	2.5	1.0	1.1	0.4	0.4
0.1	0.1	0.6	0.0	0.0	3.5	6.9	5.0	2.8	1.3	0.5	5.8	1.0	1.4	0.3	0.2
0.1	0.1	0.0	0.0	0.0	9.5	2.8	3.1	4.1	1.2	2.0	1.6	0.4	0.9	0.2	0.3
0.1	0.1	0.1	0.0	0.0	5.7	3.8	4.2	3.2	1.3	1.4	0.7	0.4	0.8	0.2	0.3
0.1	0.1	0.1	0.0	0.0	2.0	4.5	3.1	1.8	0.8	1.5	0.6	0.9	0.9	0.7	0.5
0.7	0.1	0.0	0.0	0.1	0.6	7.5	6.3	0.5	0.5	4.8	1.0	0.2	0.7	0.3	0.3
3.3	4.3	0.2	0.2	9.4	5.3	1.8	5.0	6.6	4.3	2.3	3.0	7.7	2.8	0.6	0.6
3.7	3.4	1.1	5.2	1.0	0.4	2.7	1.0	2.5	2.1	0.6	2.1	2.6	4.5	2.5	0.4
4.7	0.6	2.0	3.3	2.0	0.8	0.7	1.3	2.0	4.6	6.9	0.5	0.0	0.2	2.9	1.0
0.1	0.1	1.7	0.2	1.0	2.7	3.2	5.4	2.3	20.	2.1	22.	4.6	0.4	0.4	2.0
0.2	0.1	1.1	0.6	0.6	3.7	1.5	5.3	20.	8.1	6.3	1.0	20.	36.	0.1	4.2

Now it can be seen that certain sub-blocks require exceptional bit allocations and normalizing constants, have many significant samples without a normal bit allocation zone, produce a high mean squared error even when optimally coded and furthermore possess the highest AC energy. This last property allows easy identification of these sub-blocks and hence definition of an adaptive, variable zonal coding is possible. An example is presented to indicate the performance of such a scheme. The most difficult sub-blocks will require many bits in order to be coded satisfactorily, e.g. 3 bits per pixel then the relatively high energy sub-blocks might be coded well at 2 bits per pixel. The effect for these images is shown in Figure 12 where the bit rate for the remaining (background information) sub-blocks is shown calculated to yield an overall 1.5 bits per pixel.

Figure 12

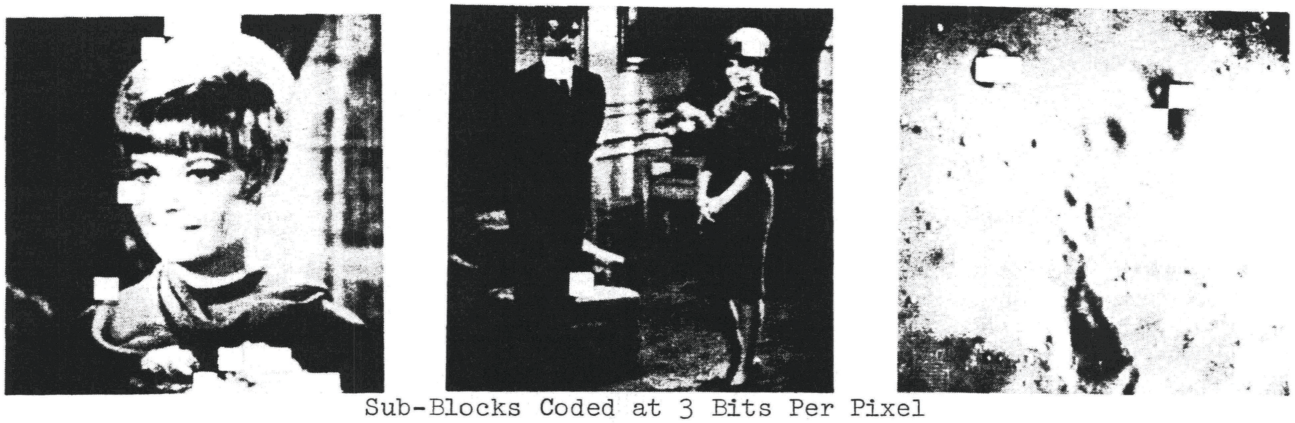
Image	Number of sub-blocks at 3.0 Bits per pixel	Number of sub-blocks at 2.0 Bits per pixel	Number of sub-blocks remaining	Rate of remaining sub-blocks Bits per pixel
Girl	11	127	118	0.82
Couple	3	120	133	1.02
Moon	4	71	181	1.27

The distribution of bits is shown in Figure 13 and it is immediately obvious that the AP is a very effective delineator of detail in the picture. Excellent quality would result from such a scheme with the only problem occurring when the discreteness of the sub-blocks causes part of a feature to be missed.

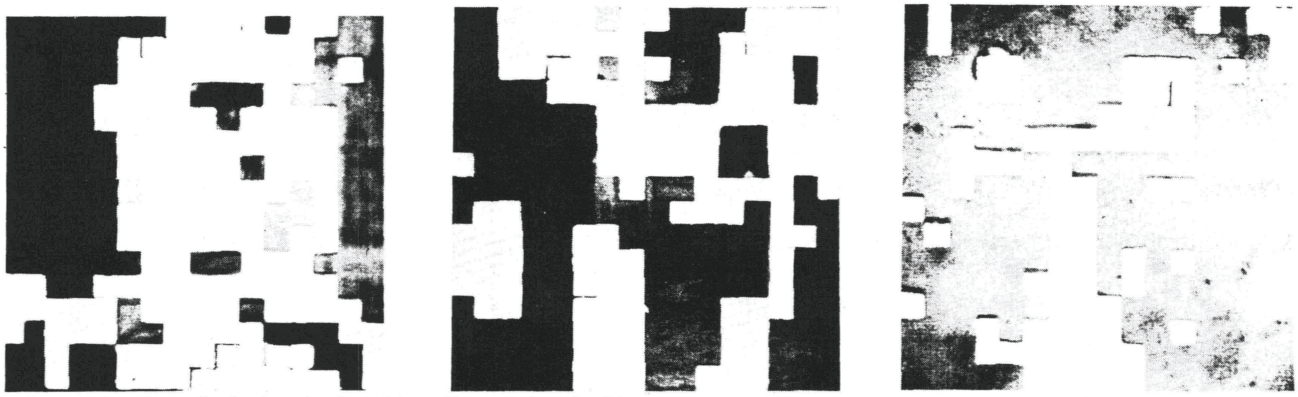
Conclusion - The Application of Statistical Coding

The results presented in previous sections show that most areas of the images are coded effectively by straightforward application of the statistical models. The exceptions, when they occur are significant because they involve prominent features.

Figure 13
Adaptive Zonal Coding



Sub-Blocks Coded at 3 Bits Per Pixel



Sub-Blocks Coded at 3 Bits Per Pixel Plus Those Coded at 2 Bits Per Pixel

Typically these are characterized by a sharp high contrast edge. Obviously, in a sub-block containing such features the statistics are not Markovian and not stationary. No manipulation of the maximum variance zone except to simply increase its size is effective in coding this data. As an example, the transform domain of the sub-block located on the diagonal edge of the girl's hairline (reference Figure 13) is shown in Figure 14.

Figure 14

Slant Transform domain of Sub-block (2,7)
Girl Image

6.080	-2.038	-0.173	-0.051	0.056	-0.099	0.005	0.002	-0.014	-0.008	0.025	0.002	0.008	-0.040	-0.001	-0.039
-2.951	-0.284	0.914	0.006	0.210	-0.040	-0.013	-0.016	-0.001	0.006	-0.021	-0.004	0.011	0.024	0.027	0.015
0.185	1.060	-0.049	-0.389	-0.009	-0.258	0.004	0.078	0.062	-0.093	0.019	0.039	-0.035	-0.014	-0.009	-0.038
0.135	0.001	-0.394	-0.055	0.161	-0.026	0.008	-0.072	-0.007	-0.011	-0.007	-0.006	-0.009	0.000	0.000	0.049
-0.269	0.418	-0.042	0.229	-0.016	0.072	0.070	-0.121	0.029	0.001	0.028	-0.124	0.003	-0.007	-0.013	-0.012
0.108	0.055	-0.543	0.090	-0.003	0.003	0.119	-0.064	-0.043	0.034	0.109	-0.022	0.012	-0.017	-0.018	0.031
0.123	-0.140	-0.049	0.144	-0.106	0.231	-0.070	0.058	0.065	-0.018	-0.001	0.020	0.001	-0.001	0.016	-0.008
0.037	-0.007	0.038	-0.021	-0.226	0.068	-0.045	0.074	-0.009	-0.057	-0.013	0.014	-0.005	0.007	0.015	-0.021
-0.018	0.029	0.084	-0.030	0.054	-0.053	-0.059	0.059	-0.037	0.020	-0.026	0.054	0.004	0.001	0.034	-0.001
0.032	-0.065	-0.047	-0.024	0.013	-0.020	-0.039	0.053	0.000	0.033	-0.061	0.036	-0.022	0.032	-0.026	0.011
0.099	-0.089	0.074	0.053	-0.030	0.063	0.012	-0.016	-0.088	-0.002	-0.005	0.002	0.020	0.031	-0.016	-0.001
0.015	-0.082	0.011	-0.046	-0.062	0.024	-0.003	0.027	0.026	0.041	0.002	0.014	0.003	-0.033	0.016	-0.026
-0.006	-0.095	-0.030	-0.001	0.026	0.020	0.060	0.003	0.009	-0.006	0.038	-0.016	0.022	-0.015	-0.020	-0.012
0.007	0.056	0.079	-0.023	-0.021	-0.044	-0.029	-0.030	0.019	-0.029	-0.005	-0.001	-0.002	0.003	0.017	0.019
0.001	0.028	-0.056	-0.007	-0.009	0.005	-0.001	-0.022	0.012	0.012	-0.007	-0.053	0.002	0.032	0.021	-0.029
-0.046	0.028	0.014	-0.010	0.037	-0.012	0.034	-0.017	0.020	-0.020	0.026	0.015	0.012	-0.035	-0.014	0.015

The significant samples have a distribution out along the leading diagonal, very unlike the maximum variance zone. In consequence, the adaptive zonal coding is recommended as best, particularly since bits are concentrated where desired - in the girl image, the girl herself is coded mostly with two bits and the insignificant background with only 0.82 bits. The moon picture, however, has the few outstanding features coded at 2 or 3 bits but the more noticeable background receives 1.27 bits.

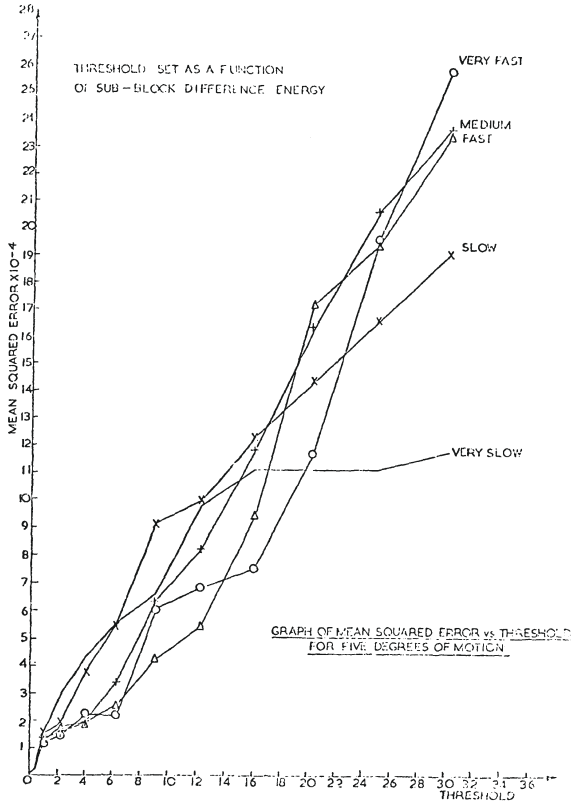
Lastly it is noted that the mathematical model approach to coding has now been abandoned in favor of zones based upon measured statistics (variances) an improved coding results since the requirements of the problem sub-blocks are built-in.

Interframe Coding

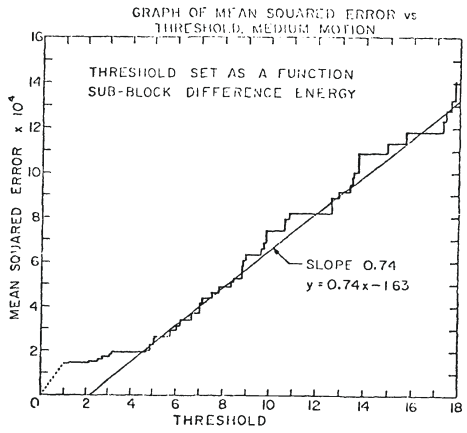
It is well known that considerable redundancy is present when a moving picture is transmitted. The adaptive coder described here was designed to take advantage of the limitations of human vision, essentially providing only sufficient information to yield the illusion of a complete high detail moving picture. The coder is specifically designed to minimize terminal cost in addition to minimizing transmission cost. In addition a two stage coding is applied in order to obtain flexibility in implementation. Ad application was picture telephone in which simple coding might be used locally with additional coding for long distance. The first stage of coding consists of the removal of spatial redundancy within single frames. This employs the techniques discussed earlier. The second stage of coding - interframe coding is thus presented with coded transform domain samples at a rate considerably less than the source rate. The bit rate reduction is to be achieved by the updating of only those portions of the frame in which movement has taken place since the previous frame. Since the interframe coder is presented with data which has been transformed in sub-blocks, it is necessary to decide whether a particular sub-block contains movement and if so, to update it. Detection of the sub-blocks which contain motion may be effected by examination of the sub-block frame difference energy with a threshold decision of whether to transmit a new sub-block or not. The effect is a remarkably linear one as indicated by the error curves of Figure 15.

Figure 15

i



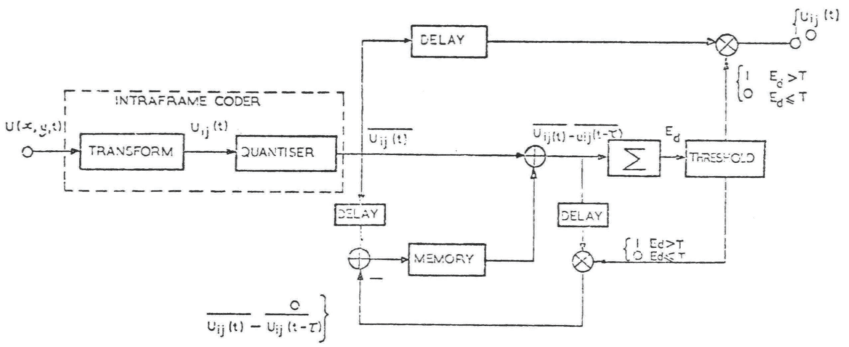
ii



Serious deviation occurs only for very slow motion where the discreteness of the update process becomes apparent. The block diagram for an implementation of the system is shown in Figure 16. Shown is the difference energy summing and thresholding, the memory and delays which compensate for the time to make the update decision. In a normal coder the memory would have to store a complete frame. The advantage of this coder is that compression has already been achieved by the intraframe coder and the same degree of compression applies to the size of the memory. A 4:1 compression (6 bits to 1.5 bits) is likely and this results in only a one quarter frame memory. From the curve of Figure 15 (ii) a threshold of 4 is selected as offering the best compromise between updating all

moving areas but not updating differences due to noise. The result is shown in Figure 17. Some blur is visible from the intraframe coding but the interframe coding is perfect. The result with a threshold of 9 is shown in Figure 18. Some updating errors are visible.

Figure 16



BLOCK DIAGRAM OF ADAPTIVE INTERFRAME CODER

Owing to the non-availability of data, it was not possible to test the coder dynamically - possibly a problem might be visibility of the sub-block updating structure. This could be coped with by overlapping sub-blocks or careful transmission of the DC and low frequency information. The buffering problems associated with the variable data rate produced by the coder were found to be similar to those encountered with a pixel cluster coding update scheme. In particular, in the case of buffer overload it is noted that an elegant reduction of data input may be effected by a further spatial compression (requantization or removal of transform domain samples) to take advantage of the limited spatial resolution of the eye to moving images.

Figure 17
Interframe Coding with Threshold of 4.0
Four Frame Sequence

Frame 2 | Frame 3 | Frame 4



1.16 bits per pixel
IMSE = 0.000052

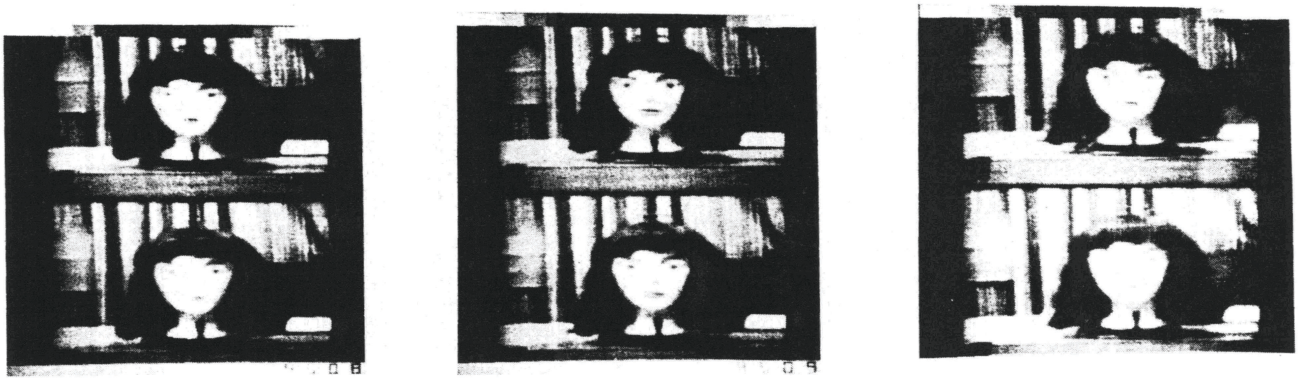
1.17 bits per pixel
IMSE = 0.000051

1.19 bits per pixel
IMSE = 0.000048

(i) Intraframe coding at 3.0 bits per pixel

Figure 17 (Continued)

Frame 2 | Frame 3 | Frame 4



0.54 bits per pixel
IMSE = 0.000049

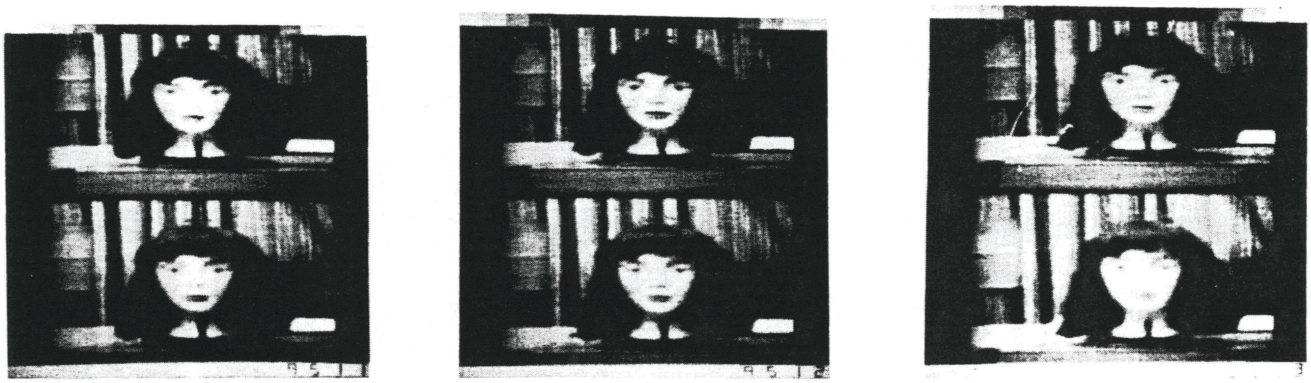
0.54 bits per pixel
IMSE = 0.000048

0.55 bits per pixel
IMSE = 0.000055

(ii) Intraframe coding at 1.5 bits per pixel

Figure 18
Interframe Coding with Threshold of 9.0
Four Frame Sequence
Intraframe Coding at 1.5 bits per pixel

Frame 2 | Frame 3 | Frame 4



0.5 bits per pixel
IMSE = 0.000063

0.52 bits per pixel
IMSE = 0.000074

0.52 bits per pixel
IMSE = 0.000092

References

1. "Orothogonal Transform Coding of Still and Moving Pictures", C. Reader, Ph.D. Thesis, Sussex University, England, 1974.