ADAPTIVE IMAGE DATA COMPRESSION

Oscar A. Zuniga, Robert M. Haralick
Virginia Polytechnic Institute and State University

Ronald L. Klein
Kansas University

Abstract

Image data compression can be achieved by a number of techniques such as DPCM and Transform Coding. Adaptive image compression can be done with any of these techniques. Adaptive image data compression is a procedure in which the number of bits allocated to each image block changes from block to block depending on the block complexity. This variable input bit rate must be converted to a constant output bit rate by a rate buffer.

In this paper we study some causal and non-causal approaches to adaptively allocating bits under the constraint of a fixed size buffer. We discuss the optimal non-causal approach whose performance is a least upper bound on any causal approach and we give some experimental results using a DPCM compression procedure.

I. Introduction

Many encoding schemes for achieving image compression have been used in the past, most popular among these DPCM and Transform Coding. To date most of these techniques have been utilized with fixed bit assignment procedures selected with respect to a particular encoding technique. However, we expect that the performance increase by adaptively encoding will be greater than that possible by experimentally "fine tuning" a particular kind of encoding scheme and using a fixed bit allocation procedure.

We assume that the image under consideration is partitioned into a set of equal-sized, non-overlapping blocks or subimages. The encoding of the image will then take place sequentially, block after block, and the encoding time for each block will be the same. We will also impose the constraint of a fixed number of bits per picture frame. The need for an adaptive bit allocation procedure arises for two reasons:

(1) The statistical characterizations of the image data are not known in advance.
(2) Some blocks of the image are more complex than others and require more encoded bits to maintain image quality.

Point (2) suggests that the bit rate generated by the allocation procedure should be variable, changing as the complexity of the blocks through the image changes. Best utilization of the channel, however, dictates that the channel capacity should be the desired long-range bit transmission rate. This implies that a buffer is needed to accept a variable rate input bit stream and which produces a constant rate bit stream to dump into the channel.

To ensure the long-range average bit transmission rate equals the channel capacity and at no time does the buffer overflow or underflow, a contoller is needed which given the constraints of buffer size and output bit stream rate, will allow more or less bits to be allocated to any given block depending on block complexity. For this to be possible, the controller must have knowledge of buffer state; i.e., how full the buffer is, and the complexity of future blocks. In this paper, we use the RMS error versus bit rate function of a block as a measure of its complexity.

In order to evaluate the results of the causal rate buffer constrained bit allocation procedure, we need to make a few comparisons. We expect that procedures can be ranked ordered by:

(1) Non-causal optimal bit allocation
(2) Non-causal optimal bit allocation with rate buffer constraints
(3) Causal adaptive bit allocation with rate buffer constraints
(4) Non-adaptive bit allocation

In this paper we discuss procedures for (1) through (3) and give some results on the comparisons between (1) and (2).

II. Causal Bit Allocation

In causal bit allocation, exact knowledge of error versus bit rate is not available for future blocks, but summary information of past blocks is available. Causal bit allocation then employs a model to estimate the future error versus bit rate function using the past information and the buffer constraints. Bit allocation then proceeds using these estimates. Blocks with high estimated complexity get more bits than blocks with low estimated complexity. First we describe a causal bit allocation procedure which does not use any rate buffer constraints and then we give a modification which uses the rate buffer constraints.

Let there be K blocks which must be allocated bits and let $P = \{P_1, \ldots, P_N\}$ be the N possible bit allocations which can be given to each block. Let e_p be the RMS error versus bit rate function for the present block, and let e_f be the average error per block we expect to make for future blocks after allocating bits. e_f and e_p will, therefore, map each element in the possible bit assignments set P into a real value representing the corresponding RMS error. For any number of bits b in the set P, $e_f(b)$ has the meaning of the average error made on a future block upon allocating an average of b bits to each future block.

Assume blocks 1 through t-1 are the past blocks. Let b^t be the bits allocated to the t^{th} block and B^t be the number of bits available to allocate for future blocks t to K. Then optimal bit allocation choose b^t to minimize

$$e_p^t(b^t) + (K-t)\, e_f^t\left(\frac{B^t - b^t}{K-t}\right) \tag{2.1}$$

After allocating b^t bits to the t^{th} block, there remains B^{t+1} bits where

$$B^{t+1} = B^t - b^t \tag{2.2}$$

The present error function can be used to update the expected future one. When $e_p^t(b^t) > e_f^t(b^t)$ the error that resulted from allocating b^t bits to the t^{th} block is worse than that expected for future blocks and

$$\frac{e_p^t(b^t) - e_f^t(b^t)}{e_f^t(b^t)}$$

is the relative amount of error more than expected. When this is greater than zero, it should tend to make the next estimate of expected future error larger. Hence a reasonable updating formula for e_f is:

$$e_f^{t+1} = \left[1 + \gamma \frac{e_p^t(b^t) - e_f^t(b^t)}{e_f^t(b^t)}\right] e_f^t \tag{2.3}$$

To take into account rate buffer constraints, we must not allocate a number of bits which makes the rate buffer over or underflow. Let r^t be the number of bits in the rate buffer just after block t-1 has been processed. Between time t-1 and t the rate buffer will dump c bits onto the channel and accept b^t bits from the t^{th} block. Hence,

$$r^{t+1} = r^t - c + b^t$$

The number r^{t+1} is constrained by not under or overflowing. If the buffer has R bits capacity,

$$0 \le r^{t+1} \le R$$

This implies

$$c - r^t \le b^t \le R + c - r^t \tag{2.4}$$

One possible rate buffer constrained bit allocation procedure is to choose b^t which minimizes (2.1) under the constraint of (2.4). Another is to minimize (2.1) with a penalty added for filling up the buffer. That is, assuming the buffer is initially half full and under the constraint of (2.4) minimize

$$e_p^t(b^t) + (K-t)\, e_f^t\left(\frac{B^t - b^t}{K-t}\right) + \frac{r^t - c + b^t - R/2}{R/2}\, \beta$$

The causal scheme is illustrated in Figure (2.1).

The buffer size problem can be stated as follows. Let R_o be the initial state of the buffer, R the buffer size, c the channel capacity and K the number of blocks per picture frame. In order to prevent overflowing or underflowing the following relation must be satisfied for every L, $1 \le L \le K$.

$$-R_o \le \sum_{n=1}^{L} b_n - Lc \le R - R_o \tag{2.5}$$

where b_n represents the bits allocated to the n^{th} block. A judicious choice for R_o is

$$R_o = R/2$$

therefore (2.5) becomes

$$\left|\sum_{n=1}^{L} b_n - L_c\right| \le R/2 \tag{2.6}$$

Letting the available number of bits per picture frame equal the amount that can be transmitted over the channel the following relation must be satisfied for every L, $1 \le L \le K$.

$$\sum_{n=1}^{L} b_n \le Kc \tag{2.7}$$

For large buffer sizes (R > 2 K c) only relation (2.7) impose a constraint on the bit allocation. For smaller buffer sizes both relations (2.6) and (2.7) constraint the bit allocation. We refer to these two situations as the non-buffer constrained and buffer constrained bit allocation respectively. This is illustrated in Figure (2.2).

Assuming that the bit allocation procedure allocates bits $b_1,\ldots,b_K$ and produces a fixed error for each block, we want to choose the buffer size R so that R is the smallest size buffer satisfying that for every L, $1 \le L \le K$

$$\text{Prob}\left\{\left|\sum_{n=1}^{L} b_n - Lc\right| \ge R/2\right\} \le P_o$$

where P_o is a given probability.

This size for R assures that by choosing a bit allocation that makes each block have the same error, the probability of buffer overflow is kept to less than probability P_o.

III. Non-Causal Bit Allocation

In the non-causal bit allocation problem the error versus bit rate functions for all blocks in the image are known before processing. Therefore, an optimal bit allocation over these blocks can take place which will minimize the total RMS error under the constraint of a fixed number of bits per picture frame.

The optimal non-causal bit allocation problem can be stated as follows. Let there be K blocks which must be allocated bits and let $P = \{P_1,\ldots,P_N\}$ be the set of N possible bit allocations which can be given to each block. The optimal non-causal bit allocation problem is then choosing $b_1,\ldots,b_K$ so that

$$\sum_{n=1}^{K} e_n(b_n) \tag{3.1}$$

is minimized under the constraint that

$$\sum_{n=1}^{K} b_n \le B \tag{3.2}$$

where B equals the number of bits that can be transmitted over the channel.

For the non-causal buffer constrained bit allocation problem, in addition to (3.2) the following relation constraints (3.1) for all L, $1 \leq L \leq K$

$$\left| \sum_{n=1}^{L} b_n - Lc \right| \leq R/2 \qquad (3.3)$$

where R is the buffer size and c the channel capactiy.

Appendix I provides a dynamic programming procedure that solves the non-causal bit allocation problem. Figure (3.1) illustrates the non-causal scheme.

IV. Experimental Results

A non-causal procedure using the dynamic programming bit allocation algorithm described in Appendix I was applied to two sample pictures. The encoding scheme used was an open and closed loop 2-D DPCM. The DPCM predictor was formed by an equally weighted average of the west, north-west, north and north-east previously DPCMed values and the five remaining neighbors coming from a low pass filtered version of the original picture. The quantizer used was a Max quantizer based on a Gaussian distribution A small amount of dither of about one quantization step was added before quantizing and substracted afterwards to help eliminate contouring effects.

Figure (4.1) shows two LANDSAT images that were used in this experiment. Both images consist of 100 x 100 picture elements and were quantized to 128 gray levels. These images were divided into 100 blocks of 10 x 10 picture elements before processing. The possible bit allocations which could be given to each block were chosen to be 1, 2, 3, 4, 5 or 6 bits per picture element.

We applied a non-causal DPCM compression procedure with buffer and no-buffer constraints to these two pictures using a compression ratio of 2.0 bits per picture element. For the buffer constrained case the buffer size was set to 5% of the total amount of bits transmitted over the channel per picture frame. There was not a significant degradation in the subjective quality of the images obtained using the buffer-constrained procedure as compared to that obtained using the non-buffer constrained procedure for the compression ratio used. The reconstructed images for the buffer constrained case are shown in Figure (4.2).

Figure (4.3) and (4.4) show plots of error versus bit rate with variance as a parameter and error versus variance for several bit rates. The RMS error between blocks in the original images and the corresponding blocks in the reconstructed images was used in both types of plots as well as the variances of the differences between the original blocks and the low-pass filtered versions which were better correlated with the errors than the variances of the original blocks. These curves were obtained by fitting the data with a 6^{th} degree polynomial using least squares. As expected for a fixed bit rate, blocks with lower complexity (variance) have associated smaller RMS errors than those with greater complexity. Also for a fixed block complexity more encoded bits result in a smaller RMS error.

Figure (4.5) shows plots of buffer state versus blocks encoded for the non-buffer constrained and buffer constrained bit allocation procedures. Shown also are the total RMS errors for each case and the RMS error corresponding to a fixed bit allocation procedure with the same compression ratio. As observed the non-causal bit allocation procedure with no-buffer constraints yields the smallest RMS errors and the fixed bit allocation procedure the largest.

V. Conclusions

The problem of Adaptive Image Data Compression has been discussed. Procedures for solving the causal non-buffer constrained and buffer constrained bit allocation problem have been suggested and experimental results for the optimal non-causal bit allocation procedure using a DPCM compression scheme were presented for the cases of non-buffer and buffer constrained bit allocation. The performance of the optimal non-causal approach is a least upper bound on any causal approach and provides us with a way of comparing the performance of different causal procedures.

It was experimentally found that the buffer constrained non-causal scheme performs well even for small size buffers.

Appendix I
Dynamic Programming Solution To The Optimal Non-Causal Bit Allocation Problem

First we describe a dynamic programming algorithm for solving the Optimal Non-Causal Bit Allocation problem with no buffer constraints and then provide a slight modification of it to include buffer constraints.

We assume there are K image blocks. Let $P = \{P_1, \ldots, P_N\}$ be the set of N possible bit allocations which may be made to any one block. Let e_n be the error versus bit rate function for the nth block. Let B be the total number of bits to be allocated to the K blocks. For the optimal non-causal bit allocation, we wish to find any

$$b_1^*, \ldots, b_K^* \in P \ , \ \sum_{k=1}^{K} b_k^* \leq B \ , \text{ satisfying}$$

$$\sum_{k=1}^{K} e_k(b_k^*) \leq \sum_{k=1}^{K} e_k(b_k) \text{ for every } b_1, \ldots, b_K \in P$$

and $\sum_{k=1}^{K} b_k \leq B$. A brute force procedure would successively go through all N^K possible values $b_1, \ldots, b_K$. Then for those satisfying the constraint $\sum_{k=1}^{K} b_k \leq B$, it would compute $\sum_{k=1}^{K} e_k(b_k)$ and remember the values $b_1^*, \ldots, b_K^*$ which gave the minimum. If we consider addition as the basic operation, such an inefficient procedure would take $2\,K\,N^K$ operations.

Fortunately, a more efficient procedure is available. It is a specialized version of Bellman's dynamic programming. To illustrate this technique, we need the following definition. For any T, $1 \leq T \leq B$ and for any M, $1 \leq M \leq K$ define $f_M(T)$ by

$$f_M(T) = \min_{b_1, \ldots, b_M \in P} \sum_{m=1}^{M} e_m(b_m)$$

$$\sum_{m=1}^{M} b_m \leq T$$

Then clearly

$$\sum_{k=1}^{K} e_k(b_k^*) = f_K(B)$$

Now notice that the f_M functions can be computed recursively since

$$f_M(T) = \min_{\substack{b_1,\ldots,b_M \varepsilon P \\ \sum_{m=1}^{M} b_m \leq T}} \sum_{m=1}^{M} e_m(b_m)$$

$$= \min_{b_M \varepsilon P} \min_{\substack{b_1,\ldots,b_{M-1} \varepsilon P \\ \sum_{m=1}^{M-1} b_m \leq T - b_M}} \{e_M(b_M) + \sum_{m=1}^{M-1} e_m(b_m)\}$$

$$= \min_{b_M \varepsilon P} \{e_M(b_m) + \min_{\substack{b_1,\ldots,b_{M-1} \varepsilon P \\ \sum_{m=1}^{M-1} b_m \leq T - b_M}} \sum_{m=1}^{M-1} e_m(b_m)\}$$

$$= \min_{b_M \varepsilon P} \{e_M(b_m) + f_{M-1}(T-b_M)\}$$

Computing $f_K(B)$ by this recursive procedure allows a more efficient calculation since it requires B N operations to compute the value of any f_M for all of its possible arguments. The values of the functions $f_1,\ldots,f_{K-1}$ have to be computed for all of their arguments and f_K only has to be computed for the argument B. This takes a total (K-1)B N + 1 operations.

These equations also suggest a quick way for determining the optimizing allocations $b_1^*,\ldots,b_K^*$. For each T, $1 \leq T \leq B$ and M, $1 \leq M \leq K$, define $P_M(T)$ to be the smallest element of P satisfying

$$f_M(T) = e_M(P_M(T)) + f_{M-1}(T - P_M(T))$$

Then

$$b_K^* = P_K(B)$$

and for $M < K$,

$$b_M^* = P_M(B - \sum_{m=M+1}^{K} b_m^*)$$

Buffer Constrained Optimal Bit Allocation Problem

When a compression procedure generates bits at a variable rate, a problem of excessively high channel capacity is created since the transmission channel must be able to handle the maximal rate that bits can be generated. This channel capacity problem can be alleviated by providing a buffer memory which dumps to the channel at a constant rate while receiving bits from the compressor at a variable rate. The size of the buffer memory and the capacity of the channel then define a bit allocation problem with side constraints that prohibit the buffer memory from overflowing.

The buffer constrained bit allocation is then described by:

- P - the set of N possible bit allocations which can be made to any block
- B - the set of K blocks to be compressed
- e_k - the distortion versus bit rate curve for the k^{th}
- B_k - the maximum number of bits that blocks 1 through k can generate without overflowing the buffer

The optimization problem is to find $b_1,\ldots,b_K \; \varepsilon \; P$ which minimizes

$$\sum_{k=1}^{K} e_k(b_k)$$

under the constraints

$$\sum_{m=1}^{M} b_m \leq B_M, \quad 1 \leq M \leq K$$

This problem is also a dynamic programming problem as can be shown by the following analysis. Define:

$$f_M(T) = \min_{b_1,\ldots,b_M \; \varepsilon \; P} \sum_{m=1}^{M} e_m(b_m)$$

$$\sum_{m=1}^{L} b_m \leq B_L \quad 1 \leq L \leq M$$

$$\sum_{m=1}^{M} b_m \leq T$$

Notice that these functions can be computed recursively, since:

$$f_M(t) = \min_{b_M \; \varepsilon \; P} \; \min_{b_1,\ldots,b_{M-1} \; \varepsilon \; P}$$

$$\{e_M(b_M) + \sum_{m=1}^{M-1} e_m(b_m)\}$$

$$\sum_{m=1}^{L} b_m \leq B_L \quad 1 \leq L \leq M$$

$$\sum_{m=1}^{M-1} b_m \leq T - b_M$$

$$f_M(T) = \min_{b_M \in P} \{e_M(b_M) + \min_{b_1,\ldots,b_{M-1} \in P} \sum_{m=1}^{M-1} e_m(b_m)\}$$

$$\sum_{m=1}^{L} b_m \leq B_L, \quad 1 \leq L \leq M-1$$

$$\sum_{m=1}^{M-1} b_m \leq \min\{B_M, T-b_M\}$$

$$= \min_{b_M \in P} \left\{ e_M(b_M) + f_{M-1}(\min\{B_M, T-b_M\}) \right\}$$

Let $p_m(T)$ be the best bit allocation for block m given that a total of T bits is available for blocks 1 through m. Then we define $p_1(T)$ to satisfy $f_1(T) = e_1(p_1(T))$ and $P_M(T)$ to satisfy $f_M(T) = e_M(p_M(T)) + f_{M-1}(\min\{B_{M-1}, T-p_M(T)\})$. The optimal allocation $b_1^*, \ldots, b_K^*$ for blocks 1 through K is then given by:

$$b_K^* = p_K(B_K) \quad \text{and}$$

$$b_M = p_M(B_K - \sum_{m=M+1}^{K} b_m^*), \quad 1 \leq M < K$$

References

[1] C. A. Andrews, J. M. Davies, and G. R. Schwarz, "Adaptive Data Compression," Proc. IEEE, vol. 55, pp. 267-277, March 1967.

[2] A. V. BalaKreshnan, R. L. Kutz, and R. A. Stampfl, "Adaptive data compression for video signals," NASA Goddard Space Flight Center, Rept. X-730-66-110.

[3] R. L. Kutz and J. A. Sciulli, "An adaptive image compression system and its performance in a noisy channel," presented at the Internatl. Information Theory Symp., 1967.

[4] L. D. Davisson, "Theory of adaptive data compression," in Advances in Communications Systems, New York: Academic Press, 1966, pp. 173-192.

[5] M. Tasto and P. A. Wintz, "Image Coding by Adaptive Block Quantization," IEEE Trans. Comm. Tech. vol. COM-19, pp. 957-972, 1971.

[6] A. Habibi and G. S. Robinson, "A survey of digital picture coding," Comput., pp. 22-34, May 1974.

[7] C. K. Chow, B. L. Deekshathln, and L. S. Loh, "Some computer experiments in picture processing for data compaction," Comput. Graphics and Image Processing, vol. 3, pp. 203-214, 1974.

[8] C. C. Cutler, "Differential PCM," U.S. Patent 2 605 361, July 29, 1952.

[9] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," IEEE Trans. Comput., vol. C-23, pp. 90-93, Jan. 1974.

[10] R. M. Haralick and N. Kattiyakulwanich, "A fast two dimensional Karhumen-Loeve transform," SPIE, vol. 66, pp. 144-158, 1975.

[11] A. Habibi and P. A. Wintz, "Hybrid Coding of Pictorial Data," IEEE Trans. Comm. Tech, vol. COM-22, no. 5, pp. 614-624, 1974.

[12] J. A. Rose, W. K. Pratt, G. S. Robinson, and A. Habibi, "Interframe Transform Coding and predictive coding methods," in 1975 Proc. ICC, IEEE Catalog 75 CH 0971-2GSCB, pp. 23.17-23.21.

[13] T. S. Huang and J. W. Woods, "Picture bandwidth compression by block quantization" presented at the 1969 Int. Symp. Information Theory, Ellenville, NY.

[14] A. Habibi and P. A. Wintz, "Image Coding by Linear Transformation and Block Quantization," IEEE Trans. Comm. Tech., vol. COM-19, pp. 50-62.

[15] J. Max, "Quantizing for minimum distortion," IRE Trans. Information Theory, vol. IT-6, pp. 7-12, March 1960.

[16] R. E. Bellman, Dynamic Programming, Princeton University Press, Princeton, NJ, 1957.

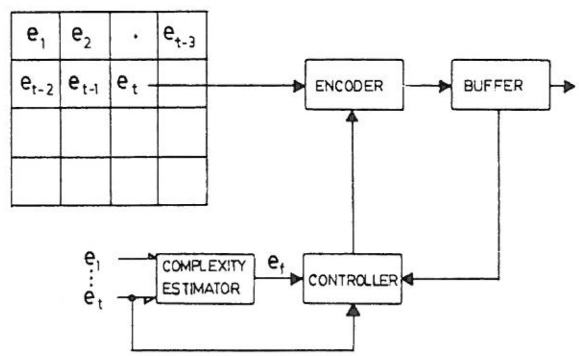


Figure 2.1 Causal Bit Allocation

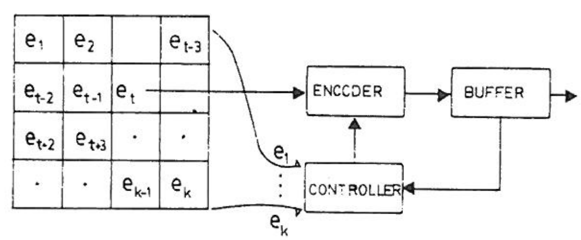
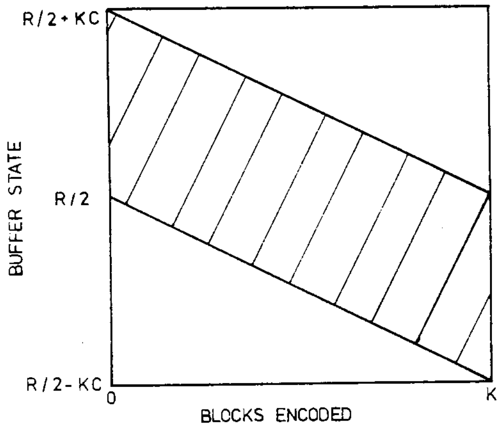


Figure 3.1 Non-Causal Bit Allocation

This research was supported by a grant from Wright Patterson Air Force Base.



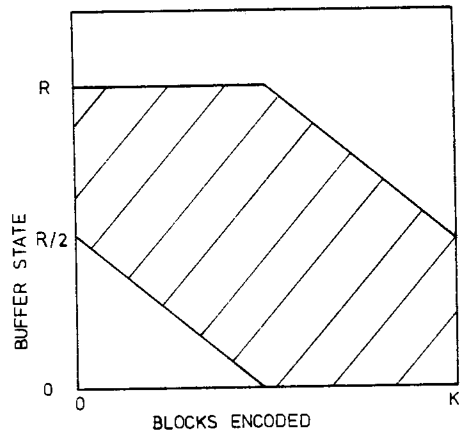


Figure 2.2 Possible region for the buffer state given R bits buffer size, K blocks per picture frame and channel capacity c. (a) Non-buffer constrained bit allocation (R>2Kc). (b) Buffer constrained bit allocation (R<2Kc).

Figure 4.1 Original images (a) First LANDSAT image (b) Second LANDSAT image.

Figure 4.2 Reconstructed images with buffer constrained bit allocation; 2 bits/pixel. (a) First LANDSAT image. (b) Second LANDSAT image.

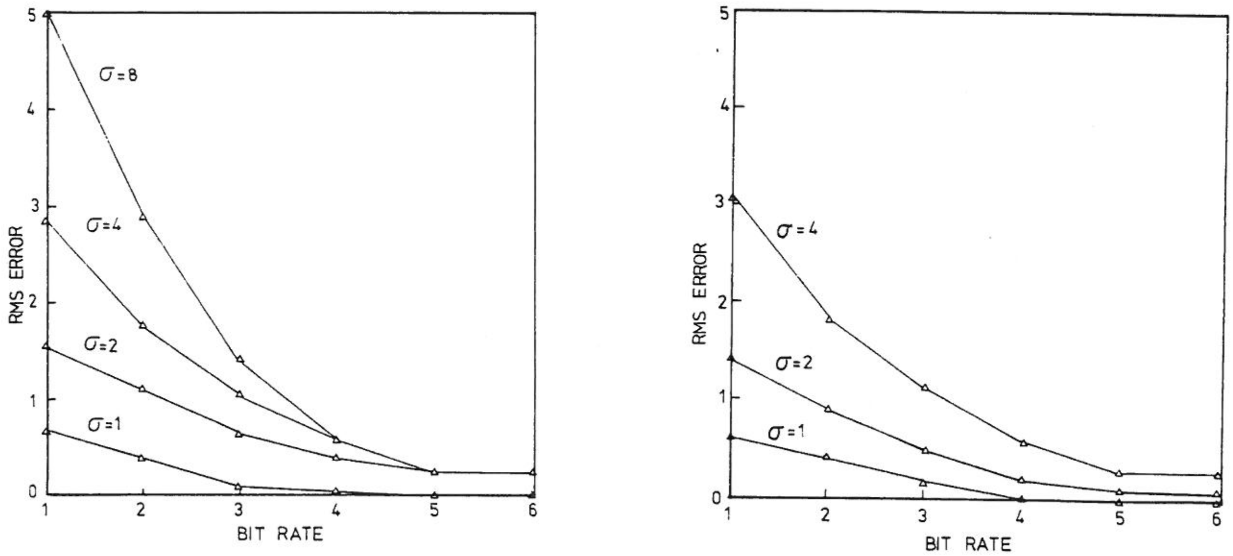


Figure 4.3 Block RMS error versus bit rate (a) First LANDSAT image (b) Second LANDSAT image

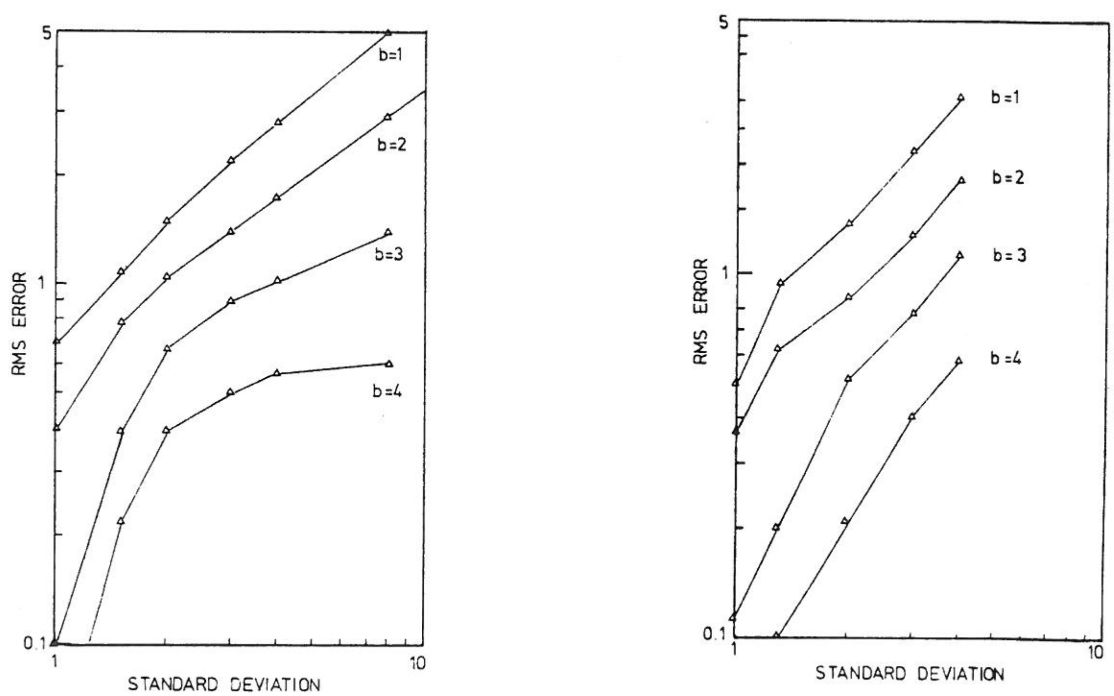
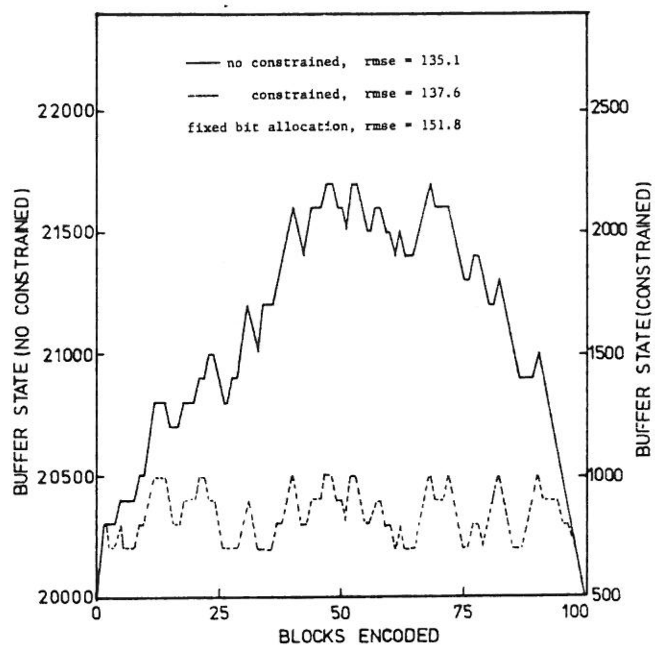


Figure 4.4 Block RMS error versus standard deviation (a) First LANDSAT image (b) Second LANDSAT image. The parameter b is the number of bits per pixel.



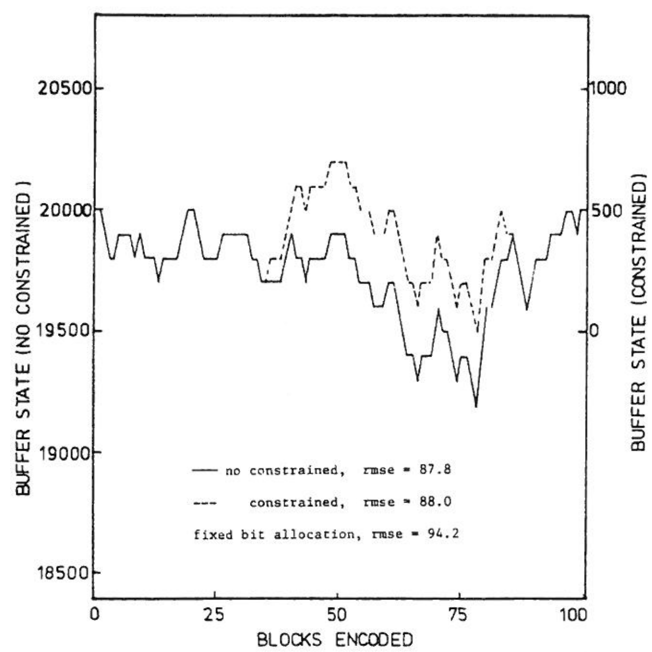


Figure 4.5 Buffer state versus number of blocks encoded.
(a) First LANDSAT image
(b) Second LANDSAT image