COMPRESSION AND ERROR CORRECTION FOR TV

R. B. Blizard, W. A. Stevens, J. L. McKinney, H. M. Gates, W. B. Anthony, D. L. Manion, and C. C. Korgel

Martin Marietta
Denver, Colorado

COMPRESSION AND ERROR CORRECTION FOR TV

The objective of this study was to determine the feasibility of digital transmission of real-time, standard format TV, together with voice and all the other data that must be returned from the Apollo spacecraft. The principal benefit offered by a digital system is a reduction in the RF power required to transmit a good picture and reliable data. To achieve this improvement it is necessary to use two techniques, always considered together, made possible by converting to an all-digital system: error-correcting codes and data compression. Data compression makes each transmitted bit more important so that fewer errors can be tolerated, and error correction gives more improvement at low error rates.

Two reports were written:

"Potential Applications of Digital Techniques to Apollo Unified S-Band Communications System. Final Report." Martin Marietta Corporation MCR-70-34 February 1970.

"Study of Potential Applications of Digital Techniques to the Apollo Unified S-Band Communications System Phase 2 - Final Report". Martin Marietta Corporation MCR-70-419, November 1970.

COMPRESSION AND ERROR CORRECTION FOR TV

MARTIN MARIETTA
DENVER DIVISION

<u>OBJECTIVES:</u>

Good Picture.

Low SNR Requirement.

Reasonable Bandwidth.

SIMULTANEOUS STUDY OF COMPRESSION AND CODING.

ANALYSIS OF CODE PERFORMANCE.

SIMULATION OF COMPRESSION AND CODING.

HARDWARE DESIGN.

BREADBOARD.

Contract NAS9-9852 - Study of Potential Applications of Digital Techniques to the Apollo Unified S-Band Communication System.

July 1969-November 1970.

DEN 066232-02(2-69)

COMPRESSION ALGORITHMS

The delta modulators, though easy to implement at the source, still will require some memory for buffering data during frame and line retrace, and for the multiplexing of other data. The predictors and interpolators make more efficient use of available memory.

The statistics of pictures is not favorable for delta modulators compared to algorithms that have a large buffering capability. Pictures typically have broad dull regions with patches of detail. The eye focuses on the detail and demands that it be reproduced sharply, if not with perfect brightness fidelity. It is impossible for a system without a buffer to encode this type of source efficiently, and therefore concentration on polynomial algorithms began early in the study.

We recommended the zero order interpolator (ZOI) algorithm, but results do not show much difference between the ZOI and the zero order predictor (ZOP). Both algorithms are fairly simple to implement, and our design studies show that operation at 5×10^6 pixels/sec is feasible (pixel: picture element). The first order interpolator (FOI) gives pictures without contouring, but it is more difficult to implement because it requires taking the quotient of two numbers.

COMPRESSION ALGORITHMS

MARTIN MARIETTA
DENVER DIVISION

I. DELTA MODULATION

II. POLYNOMIAL - Zero and First Order Predictors and Interpolators.

(ZOP, ZOI, FOP, FOI)

FOI - Best Pictures, No Contouring.

Requires Division.

$\left.\begin{matrix}\text{ZOP}\\ \text{ZOI}\end{matrix}\right\}$ Good Pictures, Simple Computation

Require Memory Buffer. Buffer Permits Use of Retrace Time.

DEN 066232-02(2-69)

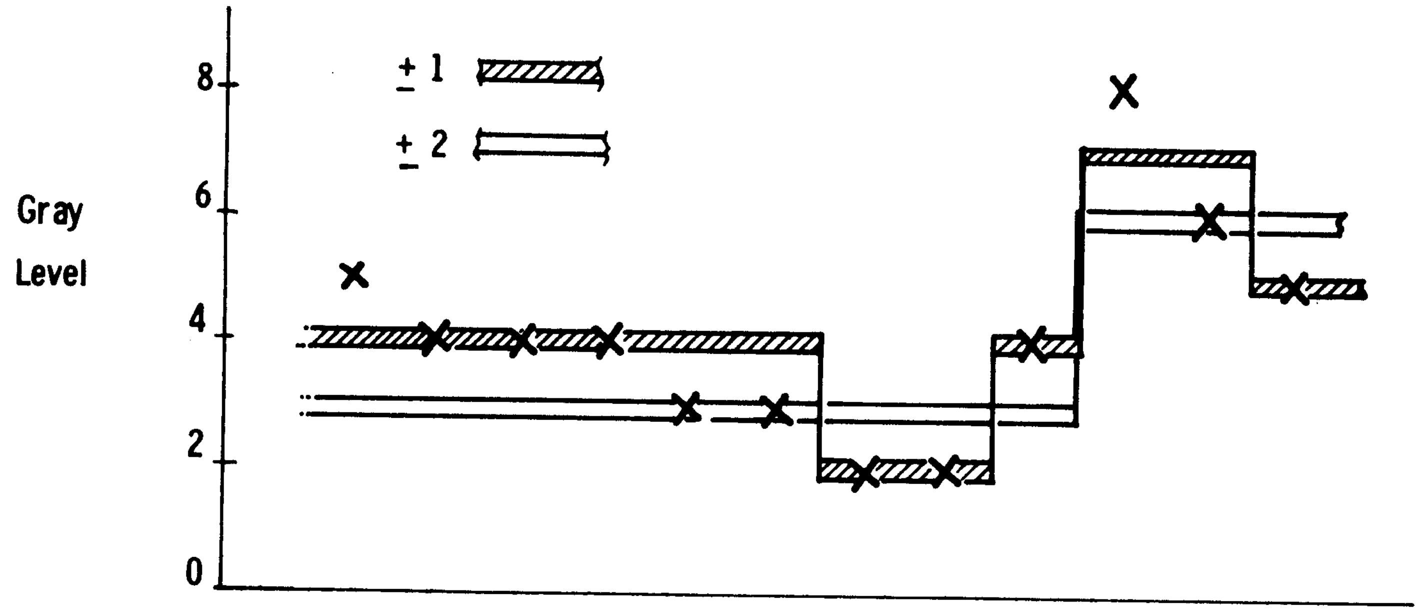
ZOI COMPRESSION VARIABLE TOLERANCE

The game is to draw horizontal lines that approximate, within a given tolerance, the actual sampled and quantized data points, represented by X's.

With a tolerance of ± 1, the transmitted message would be "gray level 4, 6 times; gray level 2, 2 times; gray level 4, 1 time; gray level 7, 2 times; gray level 5, 1 time." Relaxing the tolerance to ± 2 permits a much briefer message "gray level 3, 9 times; gray level 6, 3 times."

The average bit rate of the message is made to fit the channel rate by adjusting the tolerance.

ZOI COMPRESSION, VARIABLE TOLERANCE



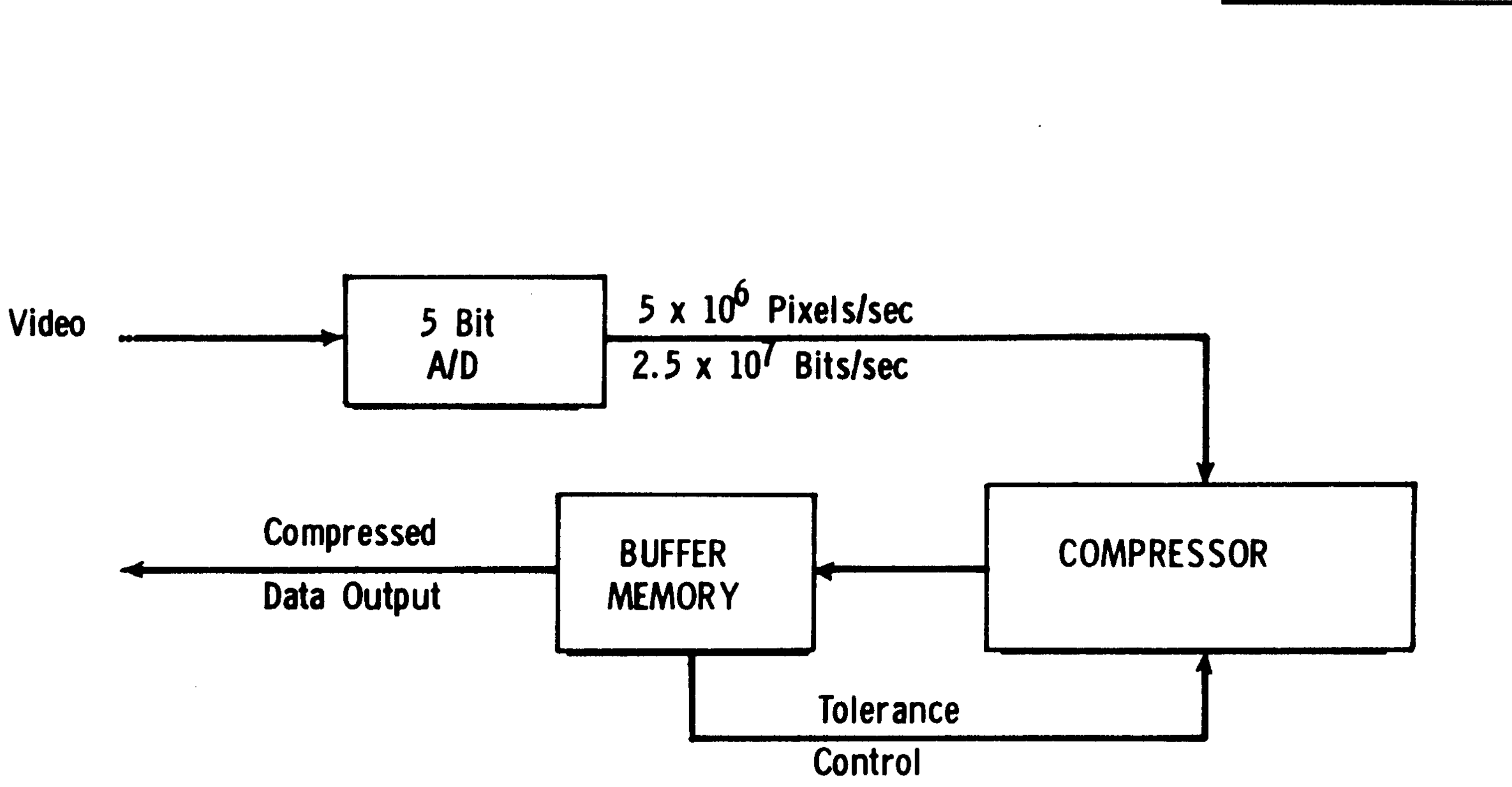
DEN 066232-02 (8-68)

COMPRESSOR BLOCK DIAGRAM

The video signal is sampled 5 times each microsecond and digitized to 5 bits. The compressor converts these data to gray-level and run-length form. The amount of data stored in the buffer is controlled by adjusting the compression tolerance. As the buffer nears overflow, the tolerance is increased in steps to $\pm$ 8 and finally the run-length is forced to a large value. When the buffer contents decrease, the tolerance is reduced, and as a final measure to prevent underflow, the compressor is forced to transmit every pixel.

The average number of bits transmitted per pixel is 1.8.

COMPRESSOR BLOCK DIAGRAM



DEN 066232-02 (8-68)

DATA CODING

The transmitted information is encoded into words of variable length. The first five bits (in the selected version) indicate one of 32 possible gray levels and the remaining bits indicate how many successive picture elements or "pixels" are to be given that gray level. This number is called the run length, and the code for the run length is of variable length, three bits for 1 through 4, four bits for 5 through 8, and so forth up to length 20. The last bit of each word is a 0, called a comma, to indicate to the reconstructing system that the word is terminated.

With such a system, unless precautions are taken, a single bit in error can upset a large part of the picture that is being transmitted. An error in the gray level code will persist for a run length. An error in the run length code will displace all subsequent pixels. An error in a comma, or an error that inserts a comma where none should be, completely upsets reconstruction of the line from that point on. To limit propagation of the effects of errors, a special code is used at the end of each line that cannot be mistaken for any segment of ordinary code words. Therefore no matter how badly the timing of a line is upset, the reconstruction system will recover on the first end-of-line code that it receives correctly. Other special orders signify the ends of field and frame.

DATA CODING

MARTIN MARIETTA
DENVER DIVISION

GRAY LEVEL CODE -- 5 BIT BINARY

RUN LENGTH CODE -- VARIABLE LENGTH WORD:

1	000	5	0010
2	010	11	10110
3	100	14	011110
4	110	20	1111110

END OF LINE	11111111111100
END OF FIELD	11111111111101
END OF FRAME	11111111111110

DEN 066232-02(2-69)

SIMULATION OF A SINGLE TV FRAME

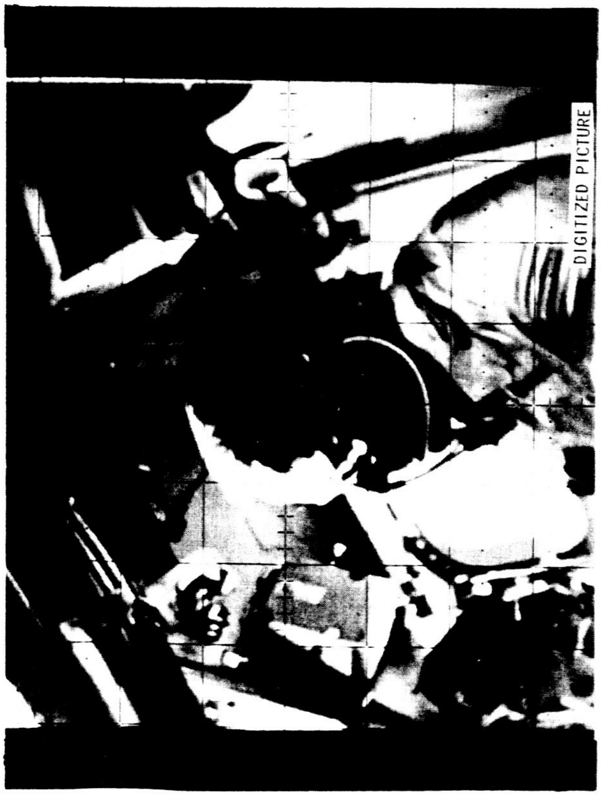
Slide 6 shows the original digitized picture with 64 gray levels and 512 x 512 pixels.

Slide 7 shows the same pattern with half the pixels ignored (in a checkerboard pattern) and the remaining pixels transmitted with an average of 1.8 bits per remaining pixel. Slides 3 and 9 show the effect of the kind of noise that would be seen in a channel using a convolutional code with constraint length 5 and rate ½ at signal-to-noise ratios per bit (E_b/N_o) of 3.0 and 4.0 dB.

It is obvious that the end of line codes are effective in getting reconstruction back on the track. The frequency of these special codes is a parameter of the system that may be adjusted for optimum performance. If middle-of-line codes were also used, the error streaks would only be half as long, but signal energy would be diverted from the main data and there would be more errors. Similarly, signal energy could be saved by inserting the special code words only after every other line, thus making half the streaks extend into the next line. The choice of this parameter is not critical, but the optimum seems to be near 1 per line, and this is a convenient arrangement.


COMPRESSED PICTURE


E_b/N_o = 4.0dB


DIGITIZED PICTURE


E_b/N_o = 3.0dB

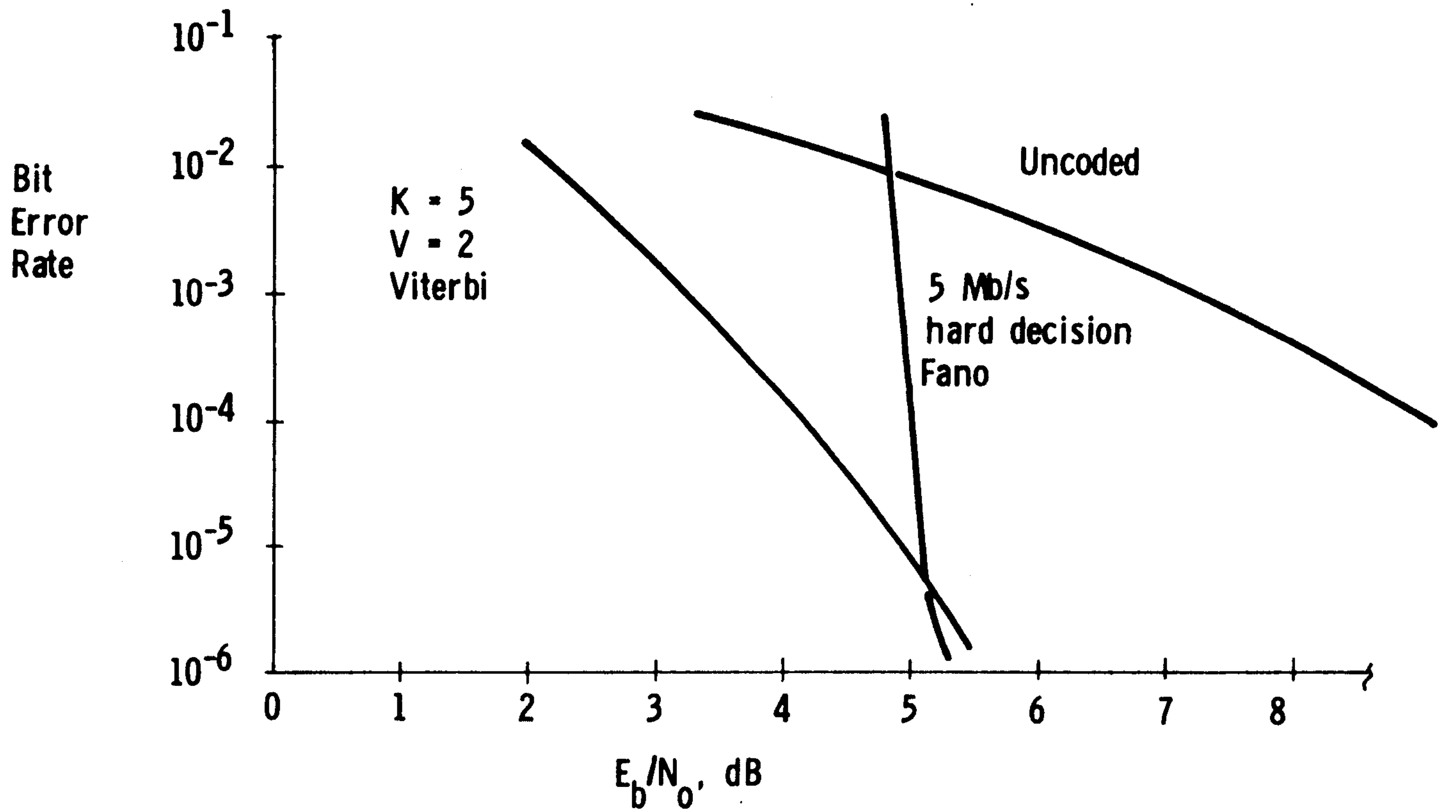
CODING COMPARISON

Two types of convolutional decoders were seriously considered for this application: a hard-decision Fano decoder like the one that the Codex Corporation had built for the Army, and a Viterbi decoder for constraint length 5 and rate ½. As the slide shows, their signal requirements are nearly identical for an error rate equal to 10^{-5} (a good error rate for compressed pictures), but the Viterbi gives better performance at lower signal-to-noise ratios.

We designed a Viterbi decoder that we were confident would run at 7.5 Mb/s, the speed needed for TV. A parts count showed that it needed about the same number of integrated circuits as the Codex machine but did not need a core memory as did the latter. Furthermore, the speed of the Codex machine (5 Mb/s) was a little lower than we wanted. For these reasons we recommended the Viterbi decoder.

A reevaluation at this time should take account of recent developments including our new algorithm.

CODING COMPARISON



MARTIN MARIETTA DENVER DIVISION

DEN 066232-02 (8-68)

NEW DECODING ALGORITHM

As part of the contract we originated, developed, and demonstrated a new decoding algorithm that can operate at very low signal-to-noise ratios and is practical at high as well as low data rates. It uses a systematic code and solves the decoding equations by successive approximations. Large constraint lengths are practical -- current work is with a length of 220, and we have used lengths as high as 958. Analysis shows that, in the limit of large constraint length and large band expansion, channel capacity can be approached. Unlike sequential decoding, the new algorithm is self-starting, and it is not necessary to insert periodic sequences of known bits to assure recovery after decoding failure.

To demonstrate the feasibility of operation at high speed, we built a breadboard that operates at 11 Mb/s with TTL MSI.

Since the completion of the contract, we have continued to develop the algorithm and have achieved substantial improvements for rate 1/2 codes. The slide shows current performance results for a code of constraint length 220 for 5, 10, and 20 iterations of the decoding process. The lower bound on error probability is based on the minimum code word weight, and it could be reduced at the cost of some increase in the amount of computation that the decoder must perform. With the present code, 20 iterations requires slightly more arithmetic operations than does the Viterbi algorithm with a constraint length of 9.

The recent improvements have not been applied to rate 1/3 codes, but we speculate that it should be quite feasible to operate at $E_b/N_o = 1.0$ dB.

NEW ALGORITHM. K = 220, V = 2.

MARTIN MARIETTA
DENVER DIVISION