N90-19411

1989

NASA/ASEE SUMMER FACULTY RESEARCH FELLOWSHIP PROGRAM

MARSHALL SPACE FLIGHT CENTER
THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

EVALUATION AND ANALYSIS OF THE ORBITAL MANEUVERING VEHICLE VIDEO SYSTEM

Prepared By:	Robert J. Moorhead II
Academic Rank:	Assistant Professor
University and Department:	Mississippi State University Electrical Engr. Department
NASA/MSFC:	
Laboratory:	Information and Electronic Systems
Division:	Computers and Communications
Branch:	Communications Systems
NASA Colleagues	H. Leann Thomas G. Daryl Craig
Date:	August 18, 1989
Contract No.:	The University of Alabama in Huntsville NGT-01-008-021

ABSTRACT

This report contains a summary of work accomplished in the summer of 1989 in association with the NASA/ASEE Summer Faculty Research Fellowship Program at Marshall Space Flight Center. The task involved study of the Orbital Maneuvering Vehicle (OMV) Video Compression Scheme. This included such activities as reviewing the expected scenes to be compressed by the flight vehicle, learning the error characteristics of the communication channel, monitoring the CLASS tests, and assisting in development of test procedures and interface hardware for the bit error rate lab being developed at MSFC to test the VCU/VRU.

Numerous comments and suggestions to the appropriate people have been made during the course of the fellowship period regarding the design and testing of the OMV Video System. Unfortunately from a technical point of view, the program appears at this point in time to be trouble from an expense prospective and is in fact in danger of being scaled back, if not cancelled altogether. This makes technical improvements prohibitive and cost-reduction measures necessary. Fortunately some cost-reduction possibilities and some significant technical improvements that should cost very little were identified.

ACKNOWLEDGEMENTS

There are many people I should thank. H. Leann Thomas supervised and directed the work, besides being a general source of information on other important matters (like where to eat). G. Daryl Craig made himself available for consultation and discussions, as well as participating in most of the meetings in which I was involved. J. Porter Clark was invaluable in keeping the computers operating properly and establishing communications between various machines and the printer. Dr. Wayne Smith, a fellow Fellow, taught me how to use the word processor programs. Dr. Frank Ingels of Mississippi State University assisted in many logistic and technical matters, both before and during my tenure here this summer. Bernd Seiler, Branch Chief, and Jim Atherton, Division Chief, made themselves available for any matters with which they could assist. Their assistance and direction is gratefully acknowledged.

TABLE OF CONTENTS

ORIGINAL PAGE IS
OF POOR QUALITY

NOMENCLATURE

CEI	Contract End Item
CLASS	Communications Link Analysis and Simulation System
CTF	Contrast Transfer Function
DPCM	Differential Pulse Code Modulation
dB	decibel
f/s	frames/sec
FOV	field of view
FWSI	Fairchild Weston Systems, Inc.
GCC	Ground Control Console
GSFC	Goddard Space Flight Center
JSC	Johnson Space Center
Kbps	Kilobits per second
MHz	Megahertz
OMV	Orbital Maneuvering Vehicle
PDA	Preliminary Design Audit
PDR	Preliminary Design Review
RS	Reed Solomon
RFI	Radio Frequency Interference
STel	Stanford Telecommunications, Inc.
TBD	to be decided
TDRS	Tracking and Data Relay Satellite
TDRSS	Tracking and Data Relay Satellite System
VCU	Video Compression Unit
VS	Video System
VRU	Video Reconstruction Unit
WSGT	White Sands Ground Terminal

1. INTRODUCTION

The Orbital Maneuvering Vehicle is an unmanned spacecraft which is scheduled to be launched in the early 1990's. Its purpose is to relocate satellites and other orbiting objects in space. One of its primary tasks is to reboost large observatories as their orbits gradually decay. The OMV Video System (VS) captures 5 frames of video data per second, compresses it by a factor of 5.5, and transmits it via TDRS to a GCC at JSC. The VS will be primarily used for remote-controlled docking with the orbiting object, since the final approach and rendezvous will be controlled by a ground-based pilot.

The OMV VS is crucial to a successful mission. However, it is highly constrained. The image quality must be sufficient for the pilot to precisely locate both the OMV and the target object. The data must be limited because of communication channel constraints. The hardware to compress the data is constrained by power and heat dissipation limitations on the OMV.

2. TASK DESCRIPTION

My task was described as follows:

1. Study OMV image processing technique using OMV documentation. List average bits/pixel at various points in the system, such as:

   a) after frame rate reduction from 30 frames/sec to 5 frames/sec.
   b) after 4 pixel:1 pixel averaging.
   c) after DPCM.
   d) after entropy encoding and Huffman runlength encoding.

2. Review scenes, digitized pixel histograms, etc. from scenes with Daryl Craig.

3. Review motion, spin, etc. of OMV and errors on channel.

4. Written report discussing the following points:

   a) expected attributes/disadvantages of OMV video compression technique.
   b) effects of the 5 frame/sec sampling rate versus motion of OMV.
   c) effects of 9.5445 MHz sampling of analog video voltage from CCD elements with bandwidth of 4.25 MHz.
   d) effects of elastic buffer/scalar feedback loop on picture quality and what feedback counts would optimize picture quality versus tendency for buffer to overflow.
   e) recommend other video compression techniques compatible with error channel characteristics and motion of OMV that could guarantee 486 Kbps data rate per camera. Compare these other techniques against OMV Video Compression Scheme relative to hardware complexity and to the factors a) through d) above.

5. Review CLASS test results/impact to OMV design.

6. Assist in development of test procedures for the bit error rate lab VCU/VRU based on channel characteristics from CLASS tests and OMV compression technique.

3. VIDEO SYSTEM DESCRIPTION

3.1 Overview

References [2-4] all have a system description to some extent. However, the system is constantly evolving. This description will emphasize the aspects that are presently under discussion or test.

The part of the video system that will be on the OMV consists of 2 redundant VCU's, 2 redundant zoom cameras, 2 redundant docking cameras, and 4 sets of docking lights. Although they are functionally not part of the video system, there are 6 sets of navigational lights which provide illumination. The ground-based portion of the video system consists of 2 redundant VRU's [2].

The primary function of the VCU's is the compression of television imagery to a bandwidth narrow enough for returning to the ground-based pilot via TDRSS at S-band [2]. The VCU's can accept video from one or two cameras simultaneously. The raw video data is compressed, RS error encoded, helically interleaved, and convolutionally encoded before being transmitted to the ground at 486 or 972 Kbps. The frame rate is fixed at 5 frames/sec. There are four compression modes from which the ground-based pilot can select (p. 1-67 of [2]):

Mode A: 2 cameras, each camera compresses 5 f/s to 486 Kbps for a total bit rate of 972 Kbps.
Mode B: 1 camera, compresses 5 f/s to 972 Kbps. The whole FOV is used.
Mode C: 1 camera, compresses 5 f/s to 486 Kbps.
Mode D: 1 camera, compresses 5 f/s to 972 Kbps. A constrained FOV is used.

In addition to the aforementioned 4 cameras, there are 2 camera interfaces designated for kit or payload cameras and 2 camera interfaces for cameras located on the Three Point Docking Mechanism. Thus the OMV had 8 camera locations and each of the 2 VCU's can read data from any of the 8 cameras.

3.2 Detailed Description

The nominal operating mode will be Mode A -- two compressed streams, each stream being 486 Kbps, interleaved for a total bit rate of 972 Kbps. In this mode, the 510x488 pixels obtained from each camera will be pixel-paired to give a 255(H) x 244(V) pixel array to be compressed. The cameras will normally be capturing 30 f/s of which 5 out of every 6 frames will simply be dropped. There is a camera mode in which 6 frames are averaged to provide a better video signal in low illumination situations. In Mode B vertical pixel pairing is performed. In Mode C, pixel pairing is performed in both dimensions, just like in Mode A. In fact, the only difference in Mode A and Mode C is that Mode A is two channel and Mode C is one channel. In Mode D only a 255x244 array of pixels -- centered in the FOV -- is used; no pixel pairing is performed. The ramifications that the nominal mode is a low resolution mode will be discussed in the ANALYSIS section below.

The video input from the cameras is standard RS-170A -- 525 lines/frame, 30 f/s, 2-to-1 interlaced. Since only 5 f/s are sent, in the nominal mode (Mode A) after pixel pairing, only a 5.5 to 1 compression ratio is needed. This is accomplished using DPCM and entropy coding techniques. The 5.5 to 1 compression yields on the order of 453,600 bits/sec, leaving room in the compressed stream for a (255,238) RS error correcting code scheme to be applied. In Modes C & D, the information for 255x244 pixels is also compacted into the same size code stream. In Mode B, there are twice as many pixels to compress per frame, but the average bits/pixel is the same. Mode B should be the easiest mode in which to achieve sufficient compression, since the compression rate scales with the square root of the area, not the area. In Mode A, the compressed video streams from each of the 2 channels are helically interleaved to depth 8 for error spreading. In the other 3 modes, 8 consecutive RS codewords are helically interleaved.

The VRU reconstructs the video and substitutes data from the previous frame for any current data that contains detectable, but uncorrectable, errors.

3.2.1 VCU

The VCU accepts analog RS-170A video from any 2 of the eight camera ports. The VCU provides a composite sync signal to the camera ports. The received signal is low-pass filtered. The latest specifications on the filter [15] indicate that the frequency response will be down 0 dB at 4.2 MHz, down 3 dB at 4.35 MHz, down 12 dB by 4.47 MHz, and eventually fall off by 45 dB . This information should be better documented. After the video signal is DC restored, it is normally routed to an 8-bit A/D converter. However, it may be routed to the bypass output if the VCU is in the bypass mode. The analog signal is sampled at 9.5445 MHz.

This a rather interesting value since it is not an integer multiple of the color subcarrier frequency or the cutoff frequency. It is sufficiently above the Nyquist rate so that sharp edges should not have ringing.

FWSI is still deciding on how they will handle the synchronization and buffering problem between the camera and the VCU. The two choices are one buffer, which is serially filled and emptied, or two buffers, one being filled while the other is being emptied. I think the two buffer approach makes more sense, but FWSI appears to be going with the one buffer approach. This means the actual compression process must occur faster which means more heat and power dissipation although it is a 75% duty cycle.

A compander circuit is used to push the coding error into the higher luminance ranges where it is not as easily detected by the eye (p. 309-310 of [9]).

3.2.1.1 Pixel Pairing

Pixel pairing (averaging two vertically adjacent pixels or averaging 4 adjacent pixels in a 2x2 area) is used to reduce the information input to the DPCM process. As previously indicated, 4 pixel pairing is used in Mode A and C, and 2 pixel pairing is used in Mode B. There is a better way to achieve this data reduction without having the negative effect on the DPCM process mentioned below [10]; this technique is discussed in the ANALYSIS section.

3.2.1.2 DPCM

DPCM is a good choice for a compression technique. Lossless coding using DPCM is generally able to achieve between 2:1 and 3:1 compression (p. 556 of [6]), so obtaining the 5.5:1 necessary for OMV should not be difficult since lossy coding is acceptable. The predictor that FWSI has chosen is consistent with what many others have determined to be the optimum 3 valued predictor ([7], [8], and p.322 of [9]). Namely, in the diagram below of 4 adjacent pixels,

$$\begin{array}{cc} C & B \\ A & X \end{array}$$

if X is the pixel value to be predicted, the predictor X' is:

$$X' = 3A/4 + 3B/4 - C/2$$

Note that A, B, and C are previous and adjacent pixel values, and $\hat{X}$ is the predicted value. Since the video signal is interlaced, to obtain the best prediction (i.e., have the highest correlation), A, B, and C should be in the same field as X. Modes A, B, and C all have vertical pixel pairing so this point is not applicable; however, in mode D, pixels C and B appear to be in the other field in the FWSI algorithm. If there is very little interfield motion, the compression reduction will be negligible, but then very little motion makes an even better case for interframe coding. (See SUGGESTIONS section for a discussion on interframe coding.)

3.2.1.2.1 Subframe Edges

There are some special cases in the FWSI DPCM technique. The first 3 special cases are basically a result of the subframe structure and the necessity of handling the leading edges of the subframes. They are:

(1) the first pixel of each subframe is a reference pixel and is PCM 8-bit coded, i.e.,

$\hat{X}=0$ (but normal correction mechanism is not used)

(2) the rest of the pixels on the first line of the subframe use only the pixel to the left as the predictor, i.e.,

$\hat{X}=A$

(3) the first pixel on the rest of the lines in the subframe uses only the pel above as the predictor, i.e.,

$\hat{X}=B$

3.2.1.2.2 Image Edges

The fourth caveat is an edge predictor circuit. Namely if |C-A| is much greater than |C-B|, then a horizontal edge is assumed to occur between the two lines and $\hat{X} = A$. Likewise, if |C-B| is much greater than |C-A|, then a vertical edge is assumed to occur between the two columns and $\hat{X} = B$. As seen by the results on the hex split screen test chart, this works great -- IF THE PICTURE CONTAINS NO NOISE. However, I question its value for a real scene. In fact, it appears NOT TO BE IN THE HARDWARE as documented 22 May 1989 for the timing audit conducted 14 April 1989 at FWSI.

3.2.1.3 Quantizer

The quantizer in the DPCM loop is used to control the rate at which compressed information is generated. A scalar, K, which establishes the bin width is determined by the bitrate controller. The K value is re-calculated every line pair. In the latest incarnation of the system [13], K can take on 16 different values ranging from 8 to 40. There are always 16 bins ranging in index from -8 to 7. The bin width, except for the 0 bin, is 2K wide. The 0 bin goes from -K/2 to K/2. For example, if K=8, any prediction error with a magnitude less than 4 falls into the 0 bin, any prediction error between 4 and 16 falls into the +1 bin, any prediction error between 16 and 32 falls into the +2 bin, and so forth. See Figure 1 (p. 2-35 of [2]). Representative values are indicated in Figure 2. Note that there is NOT a +8 bin, but there is a -8 bin. EVEN WITH K=40, THE SYSTEM IS NOT FAIL-SAFE. There are images -- antenna grid arrays and wire meshes -- that cannot be guaranteed to compress 5.5:1 with K=40. An obvious example, albeit slightly pedagogical, that is guaranteed to fail is a black and white checkerboard. Also, without the edge prediction circuit, the hex split screen test pattern would not compress sufficiently. THE SYSTEM NEEDS TO BE STRESSED. For the few test results that I have seen, the image content is so simplistic that the K value never moves into the higher values.

3.2.1.4 Entropy Coding

The quantized difference value is entropy coded (Huffman coded) in one of three forms. First, an attempt is made to send consecutive difference values via a runlength encoding of zero differences (i.e., succession of bin 0 values). The allowable runlengths are 10 to 74. Apparently FWSI found that little compression was gained by coding shorter or longer runlengths. If an appropriately long string of consecutive zero differences does not exist, then the case of 4 consecutive small differences (-1, 0, or 1 bin numbers) is tried. If that too fails, then the bin number is singly coded in a single Huffman code word.

The probability of getting runlengths of zero or four-datum groupings is enhanced by interleaving the pixel differences on adjacent video lines. For example, if X and Y are two consecutive lines of pixels,

```
... X(i-1)  X(i)  X(i+1) ...
... Y(i-1)  Y(i)  Y(i+1) ...
```

then the differences are examined in the order

..., X(i-1), Y(i-1), X(i), Y(i), X(i+1), Y(i+1), ...

Note that there are 65 codewords associated with runlength encoding, 81 codewords associated with 4 datum groupings, and 16 associated with single difference encoding. Four different codebooks are used, each codebook being associated with a group of four consecutive K values. See pages 2-38 to 2-42 of [2] for more details.

An unexplained anomaly exists in that THE FREQUENCY OF OCCURRENCE OF rlc=64 IS TWO ORDERS OF MAGNITUDE GREATER THAN ITS NEIGHBORS (p. 2-40 of [2]). It appears that FWSI was using some type of look ahead mechanism at some point in their code; IS THAT MECHANISM STILL IN THE CODE BEING RUN AT CLASS, BUT NOT IN THE HARDWARE?

3.2.1.5 Subframe Format

For error truncation purposes, each frame is divided into subframes. Each frame is 244 lines. A subframe can be 4, 10, or 20 lines, with 20 lines being the default. Each subframe is handled on a line-pair basis. Every subframe starts with a syncword (whose uniqueness is questionable [14]), a subframe I.D., and the reference pixel mentioned before. Every line pair includes the 4-bit scalar index and 2 lines of compressed video data. The assignment of size and value to some of these parameters -- sync word, subframe I.D., and scalar index -- is somewhat arbitrary, but the sizes and values NEED TO BE CLEARLY STATED. The subframe syncword (size and value) and one of the scalar values appear to have changed since C&DM PDR [2], for example. Note that all frames end with a 4 line subframe. Also there are 3 sync words -- subframe, RS, and Viterbi -- to keep up with.

3.2.1.6 Transmission Buffer and Bit Rate Controller

This is the part of the VCU that is still changing and is untested. What the Bit Rate Controller is supposed to do is try to maintain a constant bitrate per line-pair. The actual implementation is still evolving. The size of the output buffer is also changing. The latest guess from FWSI is that it is 32 kbits or 64 kbits.

3.2.1.7 Reed Solomon Encoder

Since the TDRSS will be affected by bursty RFI, the compressed data is RS encoded and helically interleaved to depth 8 for error detection, correction, and spreading. The RS format is such that every 255 bytes has 238 bytes of data, 16 bytes of error detection and correction code, and 1 sync byte.

3.2.2 VRU

Basically the VRU undoes what the VCU did. It de-interleaves the compressed data, performs error detection and correction, and decompresses the data. The one added complication is when an error is detected that cannot be corrected. This error may be detected as a result of an incorrect pixel count, an incorrect subframe I.D., or may come from the RS decoder. Independent of the source, the VRU simply retains the old data for that subframe rather than replace it with new, but known incorrect data. This is called subframe replacement.

3.2.3 Cameras

The cameras are a crucial part of the whole system. If they do not deliver a clear, clean, crisp, sharp video signal to the VCU, then the resulting image at the GCC will be degraded. The old computer paradigm holds true -- garbage in, garbage out. The key to obtaining a sharp, high resolution image is probably the CTF. Unfortunately, the only spec on the CTF is its value at the Nyquist frequency. It would be better if the roll-off was better specified, like the CTF at 90% and 110% of the Nyquist rate (see p. 2-62 of [2]).

4. STATUS

The original OMV proposal occurred in 1985; TRW is the prime contractor (p. 1-28 of [2]). TRW in turn subcontracted to FWSI to design and build the video system. FWSI in turn proposed that Cyclotomics be subcontracted to supply the RS coding/decoding once that function was added to the video system.

The other side of the organization chart is a result of the need to test the design and development activities. The present test activity is for the video return link. It was decided to do both static and dynamic tests at the CLASS facility at GSFC. GSFC has contracted with STel to integrate the test and model systems (e.g., to integrate the FWSI VCU/VRU code into CLASS), for setup and maintenance of the special configuration for the test, for system operation of the test, and to maintain the database. GSFC has contracted with LinCom to do model and analysis system development and to do special purpose analysis as required. LinCom generated the test plan and associated requirements and defined the special purpose models, analysis systems, and test points, i.e., the unique interfaces for OMV. Thus the organizational chart looks somewhat like this:

```
design/development <----------> test

                    MSFC
                   /    \
               TRW       GSFC
              /          /  \
          FWSI       STel  LinCom
         /
  Cyclotomics
```

A unit level PDR for the VCU/VRU was performed in 06/88 (Table 1.1-3, "Video Equipment Development Schedules" in [2].) Unfortunately, the cameras have yet to have even a PDR, although the C&DM PDR in 08/88 indicated that PDR for the cameras was scheduled for 02/89. The docking and navigational lights had their PDR in 12/88, with no further design reviews for the lights on the schedule in Table 1.1-3 of [2].

On June 20, 1989, GSFC reported good success with the tests of the RFI link using the FWSI VCU/VRU software.

On July 20, 1989, FWSI reported that they got the VCU/VRU hardware working. At this point all seemed well. However, soon after the GSFC presentation on June 20, I noticed that the bit rate was too low by about 6.4%. It turned out that the FWSI code had a bug in the bitrate controller section. Bytes were being reserved in the compressed data stream for the RS error correction bytes, but the bitrate was controlled as if the RS error correcting bytes would be appended. For example, the bit rate being obtained was 91,000 bits/frame instead of 97,200 bits/frame. 910 is 93.6% of 972. The RS code is a (255,238) code, i.e., for each 238 data bytes, 16 error correction bytes and 1 sync byte are added. 238 is 93.3% of 255. The amount of compressed data was being too heavily constrained; the FWSI compression code was forcing both the data and the check bytes into 91,000 bits instead of only the data into 91,000 bits. It is problems like this that make us leery of becoming too confident all is well.

The latest "schedule" for the video processing delay requires that the data from one frame of video be sent out every 200 msec. There is a long latency allowed (~100 msec) between when the first pel is camera-captured and the data corresponding to the last pixel is sent to TDRS, but this delay pales in comparison to the 3 second delay between the pilot sending a command from the GCC and the results being seen at the GCC.

The mechanism(s) for indicating subframe replacement rate and bit error rate at the GCC is still undecided. The need for 4 and 10 line subframes seems to be less obvious that it once was. With 20 line subframes, the knee in the margin curve (image quality vs. link margin) has been seen to be very sharp in the results from the CLASS tests [14].

Within the last few days there has been much discussion about cancelling or scaling back the OMV effort. As of 18 August 1989 OMV is still alive, but knowledgeable sources say it is likely it will be at least scaled back.

5. ANALYSIS

5.1 Video Data Quality

On page 1-41 and 1-42 of [2] is a specification of the Video Data Quality Requirement (CEI Paragraph 3.2.1.11.7.5). It is broken down into 4 parts: a) pixels/frame, b) FOV, c) frames/sec, and d) Video-Peak-Signal to RMS-Error Ratio. The paucity the requirements of parts a) and c) have been discussed above and will be discussed some more below, but they ARE specified. Part b) is probably best argued from a "we have to see the object" perspective. I have no problems with it. HOWEVER, part d) was left as a TBD. It was pulled out and discussed on page 1-74 and 1-75 of [2] as an issue which needed further study. The discussion there is generally to the point, but I take issue with the position that RMS error measures are meaningless -- they are less than perfect but much better than anything else. They are, in fact, least meaningful for noisy source images, which will occur on the OMV unless there are good lighting and cameras.

MSFC needs to make sure that the sequences being used by TRW, FWSI, and CLASS are as good an example of what OMV will see as possible. This means the noise content, the spatial sampling, and the dynamic range should match what the flight VCU will compress. A good test image would be obtained by adding 0.01 variance white noise to the present hex split screen test pattern. Rough calculations indicate it will NOT be sufficiently compressed since the white noise will defeat the edge predictor scheme. This, I claim, is a better approximation to scenes that will be encountered in space than the test pattern without noise.

5.2 NASCOM Induced Delay

Both a NASA report [1] and a TRW Quarterly Report (pp. 158-165 of [5]) address this concern: the 3 second delay between the time a pilot initiates a command and the result is displayed on a video monitor at the GCC. It is my understanding that this implies a 1.5 second delay in the forward link. I am told that this delay would be cut by 33% if the pilot was at WSGT or if the link between WSGT and JSC were terrestrial. Although the simulations indicate that this added delay would only reduce the probability of a successful first docking by 7% percent -- from 97% with a 2 second delay to 90% with a 3 second delay (p. 9 of [1]) -- that is a significant problem. Facets of the mission that will be negatively impacted include fuel consumption, mission time planning, and accuracy of actual docking attempt. This appears to be AN ADMINISTRATIVE PROBLEM that upper management NEEDS to address.

5.3 Bitrate Controller

The bitrate controller being used in the CLASS tests is one which FWSI had in their software simulator up to the end of last year. In January, 1989, FWSI proposed a new bitrate controller, which is supposedly the one they are implementing in the hardware they are building. The two controllers do differ; how much and is it significant are the questions. A timing audit [11] was performed on the FWSI VCU by TRW in April 1989. The bitrate controller board had a number of "possible problems". Few details were given about its operation (one block diagram at the level of PROMs, latches, and counters). More information must be forthcoming. Since the bitrate controller assures a fixed bit rate and makes sure the transmission buffer does not overflow, its correct operation is rather crucial.

I have a lot of questions about this board/scheme, primarily because I question the validity of the image test data being feed the VCU code at CLASS. The system simply has not been forced to do some real compression. It appears nobody has determined what will happen if the transmission buffer overflows. The way the FWSI VCU is designed, underflow should be preventable, but overflow is another issue; the scalar values simply do not go high enough. What will happen if the transmission buffer overflows?

The bitrate controller does seem to give a steady image quality over the whole picture. Unfortunately I was never cleared to look at the details of the FWSI code or certain documents. I suspect, however, they have methods to prevent the scalar value from oscillating or changing values wildly, since such methods and the needs for such methods have been well documented in the literature [8,12].

5.4 High Resolution

The preliminary specs I have seen on the cameras do indicate they are capable of capturing a good video signal. However, the VCU modes limit the resolution. The pixel pairing in Modes A, B, & C immediately half the spatial frequency in at least one, if not both, dimensions. I think it would have been better to have kept the input resolution into the VCU high and compressed more when necessary. Pixel pairing is like giving up before you start. There are better ways to get the same effect -- less input pixels.

One example is to compress the first field and use a smart interpolation/replication scheme for the second [7,10]. Note this reduces the input bit rate by a factor of 2, but keeps as high a spatial resolution as can be had with only one field. If a higher resolution picture of a stationary object is needed, both fields can be processed. Once the second field is processed, the data for it remains valid as long as the pixel values on the two adjacent lines, which are in the first field, do not change. However, once the object starts moving -- either from the camera moving or the object moving -- the motion is tracked by interpolating lines from the first field to produce the second field. Motion can be detected simply by examining the prediction error of the first field pixels.

Mode D is the only mode presently in which the pilot will be able to distinguish fine details; unfortunately the FOV is limited.

The present temporal resolution of 5 f/s seems sufficient for almost all possible OMV operations since NASA takes great pains to assure that everything happens in space at as deliberate a pace as possible.

5.5 Encryption/Decryption

Since the bits in the compressed data cannot be easily picked out and associated with a particular pixel, some encryption is being performed simply by the compression process. Changing the sync signals on every occurrence would add to the encryption process, otherwise the repetition could be picked out and the "code" begin to be broken. A compression technique that would even better encryption and has been shown to give 5-15% more compression on photographic quality images is arithmetic coding [7,10].

6. SUGGESTIONS

The first suggestion I was going to make was to have a system test of all the units. However, I understand that that idea has been already proposed and subsequently denied. I am especially concerned about the VCU box FWSI is building. The block diagrams from the April 1989 Timing Audit have MANY mistakes -- symbols/sec rates off by factors of 2, rounding and not rounding 9.5445 MHz to 10 MHz in the same figure (figure 5), D flip-flops that have input/output lines unlabelled, mislabelled, and multiply labelled, etc.. The document is so badly composed and so full of errors, I gave up trying to figure out what they were doing, much less whether it was good.

Once the Bit Error Rate Lab get setup, one of the first things that needs to be done is to simply run MANY, MANY (maybe 200) pictures through to check the bit rate controller and the transmission buffer parameters. Unfortunately the VCU/VRU scheme in the FWSI code differs appreciably -- the question is is it significantly -- from the VCU/VRU scheme being implemented in the hardware. This will taint anything that is discovered, but that may be the best that can be done.

There are some (minor) suggestions I have regarding the FWSI video compression scheme:

1) Quantizer Bin Representative Values: Rather than using the centroid of the distribution of values within the bin, a savings in hardware and pipeline delay can be had by using the smallest magnitude value in each quantization bin as the representative value [7,10]. This eliminates the need to clip the reconstructed pel values -- a PROM delay -- and also reduces the width of the adder output from 9 to 8 bits. Experimentally, the image degradation is usually unnoticeable. Note this is COST REDUCTION suggestion.

2) Gap Bridging: There is a technique known as gap bridging which should increase the length of the zero bin runs [7,12]. It has been shown to give 15-20% rate reduction, with little if any degradation in image quality. In fact it tends to eliminate noise spikes, thereby improving the image quality. Note this does indeed mean the SNR will go down since noise is being eliminated but measured as if it were added. I.e., the noise reduction will increase the difference between the original and the reconstructed image, thereby increasing the SNR. Gap

bridging does not, however, appear to be a necessary function since no test picture has really stressed the VCU yet.

3) Scalar Values (K): Although the bigger problem with the compression scheme lately has been underflow, there is a rather easy way to guarantee the VCU will never overflow. If the maximum value of K was increased from 40 to 64, the VCU would essentially have a Delta modulation mode, since all differences would fall into one of 3 bins: -1, 0, or 1. This would mean all the entropy encoding would be run length encoding or 4-datum encoding. The highest bits/pel average then would be 2.5, since some 4-datum groups require 10 bits. The easiest way to absolutely guarantee that all images could be compressed to 97,200 bits would be to have another codebook for K=64 in which the maximum bits/pel never exceeds 1.5. This is another of those "granularity vs. range" problems analogous to those encountered in designing floating point number formats.

My biggest suggestion is a big one -- use an interframe compression technique. The compression ratio should almost double [6]. The extra computation may be zero [10] and the extra memory will be one framebuffer. There are techniques to truncate the error propagation [10]. In short, it is almost a no-loss improvement. This is work I think should be done regardless of the planned missions for OMV; there is simply no reason to be using that much bandwidth to get that little video information to earth. Since I have already presented this suggestion elsewhere, I'll not expound too much on it here.

7. AREAS NEEDING FURTHER STUDY

7.1 Color

Processing full color images would do two things: 1) make the video "sexier", and 2) assist in object discrimination. Full color is tri-stimulus; therefore from capture to some point in the system three times the data will have to be handled. Ultimately, at the output buffer, the amount of compressed data should only increase by 10-30% over the compressed data from processing just the luminance information. The amount of additional computation depends on the scheme.

It is worth noting where the workstation and PC markets are going. Most PC companies have emphasized obtaining 256 colors, whereas most producers of scientific workstations have put more emphasis on spatial resolution. The pictures I am seeing from the OMV video test data appear not to have reached the limit of spatial resolution. That, I think, is a more important goal for space-based imaging. Color would increase the perceived resolution, but why not increase the real resolution first? The NEXT computer is an example of just how good images that have 2-bit (no pun intended) pixels, but lots of them, can look.

7.2 Increased Resolution

Just how much spatial resolution is needed is obviously mission dependent. If all OMV has to do is dock with the Hubble Space Telescope, the OMV Video System may have spatial resolution to spare. If, on the other hand, OMV will be used for remote inspection of antenna grid arrays or wire meshes, then the ultimate in image quality will be needed.

8. CONCLUSIONS

For what it is intended to do (dock with, transport, and reboost the Hubble Space Telescope), the OMV Video System appears adequate. I personally think a more aggressive compression technique should have been used -- which would have increased the signal power -- but it wasn't. I think the present scheme could have been made more flexible for minimum cost if that had been deemed important earlier. Probably the biggest deficiency in the whole video system is a Goddard problem -- the 3 second roundtrip from GCC to OMV and back.

Some of my concerns are a fear of the unknown. I never got clearance to look at certain documents that would have indicated whether or not certain requirements and specifications were being met. Other concerns are due to incomplete specifications. There are, however, some real potential problems.

Personally, I hope OMV flys. It would be nice to see on commercial TV (note the visual medium) something about which I know this much.

9. REFERENCES

[1] Richard Dabney & Gena Crook, "Results of POLY-2000 OMV Video Downlink Study," NASA MSFC, March 10, 1988.

[2] TRW, OMV PM12-PDR, Volume 4 - Avionics, Part 3 - Communications & Data Management, August 22, 1988.

[3] Frank Ingels, Glenn Parker, and Leann Thomas, "Simulated Performance Results of the OMV Video Compression Telemetry System," International Telemetry Conference, Oct. 30 - Nov. 2, 1989, San Diego, CA.

[4] Frank Ingels, Glenn Parker, and Leann Thomas, "OMV Video Compression Error Correction Coding System," Southeastcon 89, April 1989, Columbia, SC.

[5] TRW, OMV Quarterly Review, 18 April 1989.

[6] A.K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, 1989.

[7] D. Anatassiou, W.B. Pennebaker, J.L. Mitchell, "Gray Scale Image Coding for Freeze Frame Videoconferencing," IEEE Trans. on Comm., Vol. COM-34, No. 4, pp. 382-394, April 1986.

[8] H.G. Mussman, "Predictive Coding" chapter in the book Image Transmission Techniques, edited by W.K. Pratt, Academic Press, 1979.

[9] A.N. Netravali and B.G. Haskell, Digital Pictures: Representation and Compression, Plenum Press, 1988.

[10] R.J. Moorhead, J.S. Ma, and C.A. Gonzales, "Realtime Video Transmission over a Fast Packet-Switched Network," Digital Image Processing Applications, Ying-wei Lin, Ram Srinivasan, Editors, Proc. SPIE 1075, 118-123 (1989).

[11] TRW FWSI VCU Timing Audit Results, 22 May 1989.

[12] B.G. Haskell, P.L. Gordon, R.L. Schmidt, and J.V. Scattaglia, "Interframe Coding of 525-Line, Monochrome Television at 1.5 Mbits/s," IEEE Trans. on Comm., Vol. COM-25, No. 11, pp. 1339-1348, Nov. 1978.

[13] "The C.L.A.S.S. OMV DATACOMP Test System," Demonstration: March 27-30, 1989, NASA/GSFC.

[14] CLASS Results Meeting, June 20, 1989, NASA/MSFC.

[15] Frank Ingels, private conversation.

BIN	BIN BOUNDARY VALUES									
	GENERAL		K = 1		K = 8		K = 16		K = 32	
0	-K/2	K/2	-0.5	0.5	-4	4	-8	8	-16	16
1	2 K	K/2	2	0.5	16	4	32	8	64	16
-1	-2 K	-K/2	-2	-0.5	-16	-4	-32	-8	-64	-16
2	4 K	2 K	4	2	32	16	64	32	128	64
-2	-4 K	-2 K	-4	-2	-32	-16	-64	-32	-128	-64
3	6 K	4 K	6	4	48	32	96	64	192	128
-3	-6 K	-4 K	-6	-4	-48	-32	-96	-64	-192	-128
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
7	14 K	12 K	14	12	112	96	224	192	448	384
-7	-14 K	-12 K	-14	-12	-112	-96	-224	-192	-448	-384
-8	-16 K	-14 K	-16	-14	-128	-112	-256	-224	-512	-448

FIGURE 1: Quantizer Bin Boundaries

SCALAR	BIN NUMBER								
K	0	1	2	3	4	5	6	7	8
1	0	1	2	3	4	5	6	7	8
8	0	6	19	35	53	68	85	102	119
9	0	6	22	41	60	77	95	114	133
10	0	6	25	45	67	86	107	128	149
11	0	8	27	49	73	94	117	140	163
12	0	9	29	54	80	103	128	153	178
13	0	10	32	58	86	111	138	165	192
14	0	10	34	63	93	120	148	178	206
15	0	11	37	67	100	128	159	191	221
16	0	12	39	72	106	137	169	203	235
17	0	13	42	77	113	146	179	216	246
18	0	14	44	81	120	154	190	228	253
19	0	14	46	86	126	163	200	241	255
20	0	15	49	90	133	171	211	247	255
24	0	18	59	108	159	206	247	255	255
32	0	23	78	144	211	255	255	255	255

FIGURE 2: Quantization Bin Representative Values