E123-0600

6 NOV 1978

ARPA OR8

AD A074606

DDC FILE COPY

LEVEL II

IMAGE TRANSMISSION VIA SPREAD SPECTRUM TECHNIQUES

A008563

E. H. Wrench
A. G. Weber

ARPA, GIDEP

OR8, E123-0600

ARPA Order-2303

ARPA Annual Technical Report, January 1977 - December 1977

1 January 1978

411

Prepared for
Defense Advanced Research Projects Agency
Order Number 2303
Code Number 3G10

Naval Ocean Systems Center
San Diego, California 92152

DDC RECEIVED OCT 4 1979 A

Approved for Public Release; Distribution Unlimited

393 159

NOV 0 1 1978

GOVERNMENT-INDUSTRY DATA EXCHANGE PROGRAM

GENERAL DOCUMENT SUMMARY SHEET

1 OF 1

Please Type All Information – See Instructions on Reverse

1. ACCESS NUMBER	2. COMPONENT/PART NAME PER GIOEP SUBJECT THESAURUS
E123-0600	Communications and Telemetry Equipment, Coder/ Decoder, Digital, Video, Receiver
3. APPLICATION	**4. MFR NOTIFICATION** / **5. DOCUMENT ISSUE (Month/Year)**
Engineering	☐ NOTIFIED ☒ NOT APPLICABLE — January 1978
6. ORIGINATOR'S DOCUMENT TITLE	**7. DOCUMENT TYPE**
Image Transmission Via Spread Spectrum Techniques	☒ GEN RPT ☐ NONSTD PART ☐ SPEC
8. ORIGINATOR'S DOCUMENT NUMBER	**9. ORIGINATOR'S PART NAME/IDENTIFICATION**
ARPA QR8	Image Bandwidth Reduction Program
10. DOCUMENT (SUPERSEDES) (SUPPLEMENTS) ACCESS NO.	**11. ENVIRONMENTAL EXPOSURE CODES**
None	N/A
12. MANUFACTURER	**13. MANUFACTURER PART NUMBER** / **14. INDUSTRY/GOVERNMENT STANDARD NUMBER**
RCA ----	None

15. OUTLINE, TABLE OF CONTENTS, SUMMARY, OR EQUIVALENT DESCRIPTION

This report summarizes the work performed on the Image Bandwidth Reduction Program for ARPA by the Naval Ocean Systems Center during calendar year 1977. Appendices A through I detail the work done, both in-house and under contract. Topics covered include Bandwidth Reduction System, Command Language Summary, Frame-Store Memory Executive Program, Derivation of the Fast Cosine Transform, Human Factors Involved With Viewing Compressed Images From Remote Piloted Vehicles, Coding Techniques For RPV and SAR Images, Image Coding Techniques, Stochastic Image Models and Hybrid Coding, and Operator Performance Evaluation Of Mini-RPV Video Image Bandwidth Reduction/Compression Techniques.

16. KEY WORDS FOR INDEXING	
Image Transmission; Spread Spectrum Techniques; Image Coding; Bandwidth Compression; Video Images	(Doc Des--M)
17. GIOEP REPRESENTATIVE	**18. PARTICIPANT ACTIVITY AND CODE**
Joseph Hirsch	USN, Naval Ocean Systems Ctr, San Diego, CA (Y3)

DD FORM 1 OCT 77 2000

REPRODUCTION OR DISPLAY OF THIS MATERIAL FOR SALES OR PUBLICITY PURPOSES IS PROHIBITED

411

INSTRUCTIONS FOR COMPLETING THE GIDEP GENERAL DOCUMENT SUMMARY SHEET

NOTE: Completion of a Summary Sheet by the participant is not mandatory for document acceptance into GIDEP. A Summary Sheet will be prepared by the GIDEP Operations Center for document submittals received.

PAGE 1 OF Enter the total number of summary sheet pages

BOX

1 Leave blank--entry will be completed by GIDEP Operations Center.

2 Enter standard nomenclature associated with GIDEP Subject Thesaurus selected from Section 12, Policies and Procedures Manual.

3 Indicate application which the device was used (e.g., ground, missile, shipboard, spacecraft, refer to P & P Manual, Section 13).

4 Device manufacturer must be notified of test results. Manufacturer approval of report is not required--include pertinent manufacturer correspondence with document submittal to GIDEP; check NOTIFIED entry. If document is for a nonstandard part or of a general nature and a manufacturer is not identified, check NOT APPLICABLE.

5 Enter month and year of document issue.

6 Enter complete document title exactly as it appears on originator document.

7 Identify document type by inserting letter X by appropriate descriptor.

8 Enter document number exactly as it appears on originator document.

9 Enter part name and identification as assigned by organization/agency originating the report; if not specified, enter N/A (Not Applicable).

10 Delete either SUPERSEDES or SUPPLEMENTS. If document supersedes/supplements an existing GIDEP document, enter GIDEP microfilm access number of appropriate document. If document neither supersedes nor supplements an existing GIDEP document, enter the word NONE.

11 Enter the single symbol coding for environmental exposure as defined in Subject Thesaurus, Section 12, P&P Manual (e.g., C – Salt Spray; V – Vibration; S – Shelf Life); if not specified, enter N/A (Not Applicable).

12 Enter manufacturer abbreviation and H-4 Code number listed in GIDEP Manufacturer List. If manufacturer is not listed, enter the phrase, SEE BOX 15; enter manufacturer's full name and division (if any) in Box 15. If more than one manufacturer, enter phrase, SEE BOX 15; enter additional manufacturers as appropriate. If manufacturer is not specified, enter N/A (Not Applicable).

13 Enter complete part number. Use open O for alpha letter O, and use Ø for numeric zero. If more than one part number, enter phrase, SEE BOX 15; enter additional part number(s). If a part number is not specified, enter four dashes (----).

14 Enter standard part number such as the 1N or 2N--diode and transistor designators. For GIDEP purposes, any military assigned number is considered as a government standard part number. If more than one standard number, enter phrase SEE BOX 15; enter additional standard number(s).

15 If subject matter in document can be categorized into more than one major subject category, enter additional subject categories in upper right-hand corner. Briefly summarize test results or material detailed in text of document. Include any pertinent details or comments required for proper interpretation of material presented (e.g., peculiar environmental capabilities, unique electrical characteristics that may be "state-of-the-art," or characteristics that restrict part usage in particular applications or any other details that may aid a prospective user of the part).

16 Enter appropriate words or phrases that enhance information retrieval on subject matter(s) contained in document. As a secondary data retrieval technique within each applicable Major Category (entry 2), the document is referenced in the computer data bank and Report Index according to each key word. Do not use abbreviations or words that are part of the subject category listed in Box 2. Key word phrases are limited to 60 total characters and blank spaces. Separate key words and/or phrases with commas.

17 Enter signature or name of GIDEP Representative.

18 Enter name, city, and state of participant activity or corporation and division submitting the documen: and GIDEP two-character code (e.g., X1).

IMAGE TRANSMISSION
VIA
SPREAD SPECTRUM TECHNIQUES

Investigated by
EH Wrench
AG Weber
Naval Ocean Systems Center
San Diego, California

Sponsored by
Defense Advanced Research Projects Agency
Order Number 2303
Code Number 3G10
Form Approved Budget Bureau No. 22-R0293

Accession For
NTIS GRA&I
DDC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist. | Avail and/or special
A

CONTENTS

INTRODUCTION. . . page 1

SUMMARY. . . 1

In-house work. . . 1
Contracts. . . 2

APPENDIX A: BANDWIDTH REDUCTION SYSTEM. . . 3

APPENDIX B: COMMAND LANGUAGE SUMMARY. . . 117

APPENDIX C: FRAME-STORE MEMORY EXECUTIVE PROGRAM. . . 121

APPENDIX D: DERIVATION OF THE FAST COSINE TRANSFORM. . . 125

APPENDIX E: STUDY OF HUMAN FACTORS INVOLVED WITH VIEWING COMPRESSED IMAGES FROM REMOTE PILOTED VEHICLES. . . 127

APPENDIX F: INVESTIGATION OF CODING TECHNIQUES FOR RPV AND SAR IMAGES. . . 175

APPENDIX G: STUDY OF IMAGE CODING TECHNIQUES. . . 250

APPENDIX H: STOCHASTIC IMAGE MODELS AND HYBRID CODING. . . 276

APPENDIX I: OPERATOR PERFORMANCE EVALUATION OF MINI-RPV VIDEO IMAGE BANDWIDTH REDUCTION/COMPRESSION TECHNIQUES. . . 361

INTRODUCTION

This report summarizes the work performed on the Image Bandwidth Reduction Program for ARPA by the Naval Ocean Systems Center* during ealendar year 1977. The summary briefly describes eaeh part of the program undertaken during this period. The appendiees describe in detail the work done.

SUMMARY

The major goal of the NOSC/ARPA Image Bandwidth Reduction Program was aceomplished in 1977. A eomplete flyable system was produced by RCA and delivered to Harris Corp for integration into an AQUILA remotely piloted vehiele.

The system that was delivered includes an airborne bandwidth-eompression board, a ground station deeoder, and a frame-store memory. The airborne unit is on one 5- by 9-inch (127- by 228-mm) board, dissipates 13.5 watts, and weighs 0.78 pound (0.353 kg). The unit contains four hybrid eireuits, a 32-point discrete cosine transform (DCT), a differential pulse code modulator (DPCM), an input-output interface, and a timing hybrid. The DCT and DPCM use CMOS/SOS LSI eireuits developed by RCA for the Air Foree on a previous program. These eireuits permitted RCA to design the very-high-speed low-power eireuits required for the airborne system. The airborne unit aecepts video at 60 fields per second and supplies a serial bit stream to the Harris Integrated Command and Navigation System (ICNS) for transmission by means of spread spectrum techniques. The data rates are 200, 400, 800, and 1600 kilobits per seeond. This corresponds to 0.4, 0.8, 1.6, and 3.2 bits per pixel, respectively.

The serial data stream reeeived at the ground with the Harris equipment is sent to the RCA ground station, where eireuits similar to the airborne unit perform inverse DPCM and DCT functions. The reconstrueted pixels are then stored in the frame-store memory and used to refresh the display monitor at the standard rate of 60 fields per second.

This equipment is currently at the Harris Corp in Florida and should be test flown in the AQUILA in early 1978. The final report from RCA is contained in appendix A.

Other work achieved during this period is listed below.

IN-HOUSE WORK

During this year, considerable effort has been expended to support the development of the bandwidth-compression flight hardware at RCA and to make the RCA system eompatible with the Harris modem and the AQUILA eamera. In addition, an extremely flexible eomputer-based simulation system was designed to NOSC speeifieations and delivered to NOSC by Dataware Corp. Two major pieees of software have now been written at NOSC that allow the simulation of video systems. (Actually, any system involving two-dimensional matriees ean be simulated.) The programs provide for on-line simulation of hardware systems for determination of processing algorithms, scaling, quantization levels, bloek size, etc. This software is described in appendices B and C.

---

* In 1977 the Naval Undersea Center and the Naval Electronics Laboratory Center were combined to form the Naval Ocean Systems Center.

There is considerable interest in finding computationally efficient techniques for the cosine transform. The derivation of one such algorithm is given in appendix D.

CONTRACTS

An image processing system for hardware simulation was developed by Dataware and delivered to NOSC. The system uses a PDP 11/04 computer as a controller for up to four frame-store memories of 256 pixels by 256 pixels, 8 bits per pixel each, and a television camera. The data from the camera can be transferred to any of the frame-store memories via the PDP 11 UNIBUS. The data in the frame-store memories are then available for processing either by the computer or by any device that is connected to the UNIBUS. This structure allows hardware to be simulated in the computer, then constructed and tied to the UNIBUS. The details of the hardware design can be worked out in simulation before the hardware is constructed, resulting in considerable savings in debug time on the hardware. Therefore, software routines can be replaced by hardware with relative ease, as it becomes available. Also the computer can be used to help debug the hardware, since a complete simulation of the hardware will already be running in the machine.

A study of target detection (appendix E) was conducted in which compressed video by Decision Science Corp was used to determine operator performance as a function of compression ratio and bit-error rate in the channel. The experiment involved the use of an oculometer to measure the eye movement of a subject during a target search. The results of the study indicate that the stripe mode of frame-rate reduction does not effect the eye motion of the operator during the task.

The University of Southern California (USC) has conducted several studies under contract to NOSC, including work on a new measure of image quality incorporating an eye model, a new optimum method of assigning bits for quantization, and the application of visual image compression techniques to radar images. Reports of this work are contained in appendixes F and G.

AK Jain and SH Wang, at the State University of New York, have completed a conceptual study of a hybrid cosine/DPCM system that uses an adaptive DPCM. The results indicate that the mean square error of a coded image is significantly reduced by using the technique. Since no additional information is transmitted for the adaptive system, the data rates remain unchanged. The authors also conducted some interesting theoretical work on a technique for determining the sinusoidal transform (having fast algorithms for calculation) that most closely approximates the Karhunen-Loeve transform when the statistics of the data are known. The results of this work are given in appendix H.

Northrop Corporation is under contract to investigate techniques for motion compensation of images for frame-to-frame processing. JA Roese's work at USC on interframe coding (summarized in appendix F) has shown that motion compensation can greatly increase the performance of his algorithms. Northrop is attempting to develop motion compensation techniques that can be implemented in real time with a minimum of hardware. This work is still in progress.

Also included in this report as appendix I is the final report for a Hughes Aircraft Company study of operator performance with the NOSC-developed 100-point cosine transform/DPCM compression system. This work was done some time ago but has not been reported previously.

The remainder of this report consists of appendices A through I.

APPENDIX A: BANDWIDTH REDUCTION SYSTEM

Final Report
Contract N00123-76-C-0746

July 1977

by
Advanced Technology Laboratories
Government Systems Division
RCA
Camden NJ 08102

17

PREFACE

The work described in this report was performed from June 1976 to May 1977 under U. S. Navy contract N00123-76-C-0746, entitled "Bandwidth Reduction System," by the Advanced Technology Laboratories of RCA Government Systems Division, Camden, New Jersey.

The contract was sponsored by the Defense Advanced Research Projects Agency; Col. H. Federhen was the cognizant officer. The contract was administered by the Naval Ocean Systems Center (San Diego) and the technical monitors were Dr. E. H. Wrench, Jr., and Dr. H. J. Whitehouse.

RCA personnel who contributed to the program and to the report include: J. J. Rudnick, G. M. Claffie, J. R. Richards, J. R. Barger, D. A. Bryan, R. B. Gordon, D. L. Landis, J. I. Fridgen, W. B. Schaming, and J. F. Schanne.

A portion of the work reported here was accomplished under RCA 1976 and 1977 IR&D projects R6-08 and R7-08 but is included here because of the additional insight provided into the design of the bandwidth reduction system developed under the contract. The IR&D projects covered design of the forward and inverse direct cosine transform circuits, design of the differential pulse code modulation circuits, development of the algorithm, and performance of the computer simulations.

Contract-supported tasks included the design of the analog circuitry, hybrid development and fabrication, fabrication of the airborne circuit boards, power conditioning, design of input and output digital circuitry, design and construction of the ground station, system integration, system testing, and field installation.

8

TABLE OF CONTENTS

Section		Page
I	INTRODUCTION	1-1
	A. General	1-1
	B. Equipment Function	1-3
	C. Summary of Tasks	1-3
	1. System Simulation and Analysis	1-4
	2. Design and Construction of Airborne Unit	1-4
	3. Design and Construction of Ground Station	1-4
	4. Integrated System Test	1-4
II	SYSTEM FUNCTIONS AND ARCHITECTURE	2-1
	A. Overall System	2-1
	1. TV Camera	2-1
	2. Airborne Unit	2-1
	3. Modem Equipment	2-2
	4. Ground Station	2-2
	B. Airborne Unit	2-3
	1. Synchronization and Video Signal Partitioning	2-3
	2. Input Signal Storage and A/D Conversion	2-4
	3. Discrete Cosine Transform Processing	2-6
	4. Differential Pulse Code Modulation Processing	2-7
	5. Output Signal Formatting	2-9
	C. Ground Station	2-9
	1. Input Signal Formatting	2-10
	2. Inverse Differential Pulse Code Modulation	2-12
	3. Inverse Discrete Cosine Transform	2-13
	4. Frame Store Memory	2-13
	5. Test Procedures	2-13
III	COMPRESSION ALGORITHM	3-1
	A. Horizontal Transform	3-1
	B. Coefficient Standardization	3-1
	C. DPCM Encoder	3-2
	D. Image Reconstruction	3-5
	E. Effects of Limited Precision	3-6
	F. Bit Assignment vs. Data Rate	3-12
	G. Quantizer Definition	3-13
	H. Simulator Output Sample	3-14

9

TABLE OF CONTENTS (Continued)

Section		Page
IV	SYSTEM TESTS	4-1
	A. Subjective Evaluation	4-1
	B. Digital Performance Verification	4-1
	C. Temperature Tests of Airborne Boards	4-1
	D. Waveforms	4-2
V	CMOS/SOS ARITHMETIC BUILDING BLOCKS	5-1
	A. 9 x 9 Bit Sign Magnitude Multiplier	5-1
	B. 9-Bit Adder	5-1
	C. Dual Register Retimer	5-2
VI	HYBRID CIRCUIT ASSEMBLIES	6-1
	A. General	6-1
	B. Input Interface/Master Timing and Output Interface Hybrid Circuit Assemblies	6-2
	C. DCT and DPCM Hybrid Circuits	6-3
VII	HARDWARE PACKAGING	7-1
	A. Airborne Unit	7-1
	B. Ground Station	7-5
	1. Decode Electronics Chassis	7-5
	2. Frame Store Memory	7-8
Appendix		
A	SYSTEM DESIGN	A-1
B	SYSTEM OPERATION	B-1

10

LIST OF ILLUSTRATIONS

Figure		Page
1-1	ICNS system diagram	1-2
2-1	Block diagram of airborne unit	2-4
2-2	Eneoder pipeline timing.	2-5
2-3	Block diagram of DCT processor	2-7
2-4	Block diagram of DPCM processor	2-8
2-5	Block diagram of ground station	2-10
2-6	Decoder pipeline timing	2-11
2-7	Block diagram of inverse diffcrential pulse code modulator	2-12
3-1	Coefficient variances.	3-3
3-2	Original and reeonstrueted images	3-4
3-3	Coefficient weighting which normalizes variances	3-4
3-4	DPCM encoder diagram	3-5
3-5	Forward and inverse DCT block diagram	3-7
3-6	Original and reconstrueted images	3-15
4-1	FSM output, de, no reset, no guard band	4-3
4-2	FSM output, positive pulse, no reset, no guard band	4-4
4-3	FSM output, negative pulse, no reset, no guard band	4-5
4-4	FSM output, ramp, no reset, no guard band	4-6
4-5	FSM output, sinusoid, no reset, no guard band	4-7
4-6	FSM output, dc, reset in, guard band out	4-8
4-7	FSM output, positive pulse, reset in, guard band out	4-9
4-8	FSM output, negative pulse, reset in, guard band out	4-10
4-9	FSM output, ramp, reset in, guard band out	4-11
4-10	FSM output, sinusoid, reset in, guard band out	4-12
4-11	FSM output, dc, reset in, guard band in	4-13
4-12	FSM output, positive pulse, reset in, guard band in	4-14
4-13	FSM output, negative pulse, reset in, guard band in	4-15
4-14	FSM output, ramp, reset in, guard band in	4-16
4-15	FSM output, sinusoid, reset in, guard band in	4-17
4-16	Linearity of analog input circuitry	4-18
4-17	Video output: Ramp Test Signal	4-19
5-1	General form of the butterfly for implementing the DCT proeessor	5-2
5-2	Average of maximum dynamic power for TCS-057 multipliers	5-3
5-3	Average propagation delay of longest path for TCS-057 multipliers	5-4
5-4	9-bit adder array	5-5

11

LIST OF ILLUSTRATIONS (Continued)

Figure		Page
5-5	Propagation delay of TCS-065 adder in 1's complement mode	5-6
5-6	Retimer registers (two per LSI array)	5-6
6-1	Output interface hybrid without cover	6-4
6-2	Master timing hybrid without cover	6-4
6-3	DCT hybrid assembly	6-6
6-4	DPCM hybrid assembly	6-6
7-1	Airborne unit	7-2
7-2	Large airborne circuit board	7-4
7-3	DCT hybrid temperature profile	7-6
7-4	Decode electronics chassis	7-7

LIST OF TABLES

Table		Page
2-1	Bits Transmitted	2-10
3-1	DCT Stage Growth for Flat Field Input of Unity Magnitude	3-8
3-2	DCT Stage Output for Maximum Fundamental Frequency	3-10
3-3	Scaling and Filter Definitions	3-11
3-4	DPCM Bit Assignments for Four Data Rates	3-12
3-5	Nonlinear Quantizer	3-13
3-6	Mean Square Error for Chemical Plant Images	3-14
6-1	Video Bandwidth Reduction System - Encoder Hybrid Data	6-2

Section I

INTRODUCTION

A. GENERAL

There has been a continuing effort to define and develop spread-spectrum image-transmission systems to provide antijam protection for a television link from small remotely piloted vehicles. The Naval Ocean Systems Center (NOSC), San Diego, CA (formerly NUC), has had a DARPA-sponsored program in this area for some time. In an early report,[1] the general system parameters to accomplish this were defined. In subsequent work, the bandwidth-compression system was defined to be a hybrid processor employing the cosine transform in combination with differential pulse code modulation (DPCM) of the frequency coefficients.[2,3] A number of techniques have been studied for implementation of this algorithm.

Early in 1976, the concepts had been developed sufficiently to undertake development of a video bandwidth-compression system for inclusion in the Integrated Communication and Navigation System (ICNS) for the Army's Aquila RPV. A contract for this was awarded to the Advanced Technology Laboratories of the RCA Government Systems Division, Camden, NJ. The equipment developed in that effort is described in this report.

Figure 1-1 shows how the RCA system fits into the Aquila electronics. The airborne unit accepts the output of a TV camera mounted in the nose of the aircraft, reduces the frame rate, digitizes the video, performs the bandwidth compression, and delivers a serial data stream to the ICNS modem, which multiplexes the video signal into its communication link with the ground. On the ground, the modem delivers the demultiplexed encoded signal to the video bandwidth-compression decoder, which restores the video to an analog signal suitable for driving a TV monitor.

---

[1] Image Transmission Via Spread Spectrum Technique, ARPA Quarterly Technical Report, Jan. 1974 - July 1974, Advanced Research Projects Agency, Order No. 2303, Code No. 3610.

[2] "Real Time TV Image Redundancy Reduction Using Transform Techniques," Robert W. Means, Harper J. Whitehouse, and Jeffrey Speiser, New Directions in Signal Processing in Communication and Control, edited by J. K. Skwyrzynski, NATO Advanced Study Institute Series, Series E: Applied Sciences - No. 12.

[3] "Television Encoding Using a Hybrid Discrete Cosine Transform and a Differential Pulse Code Modulator in Real Time," R. W. Means, H. J. Whitehouse, J. M. Speiser; 1974 National Telecommunication Record, pp. 61-74.

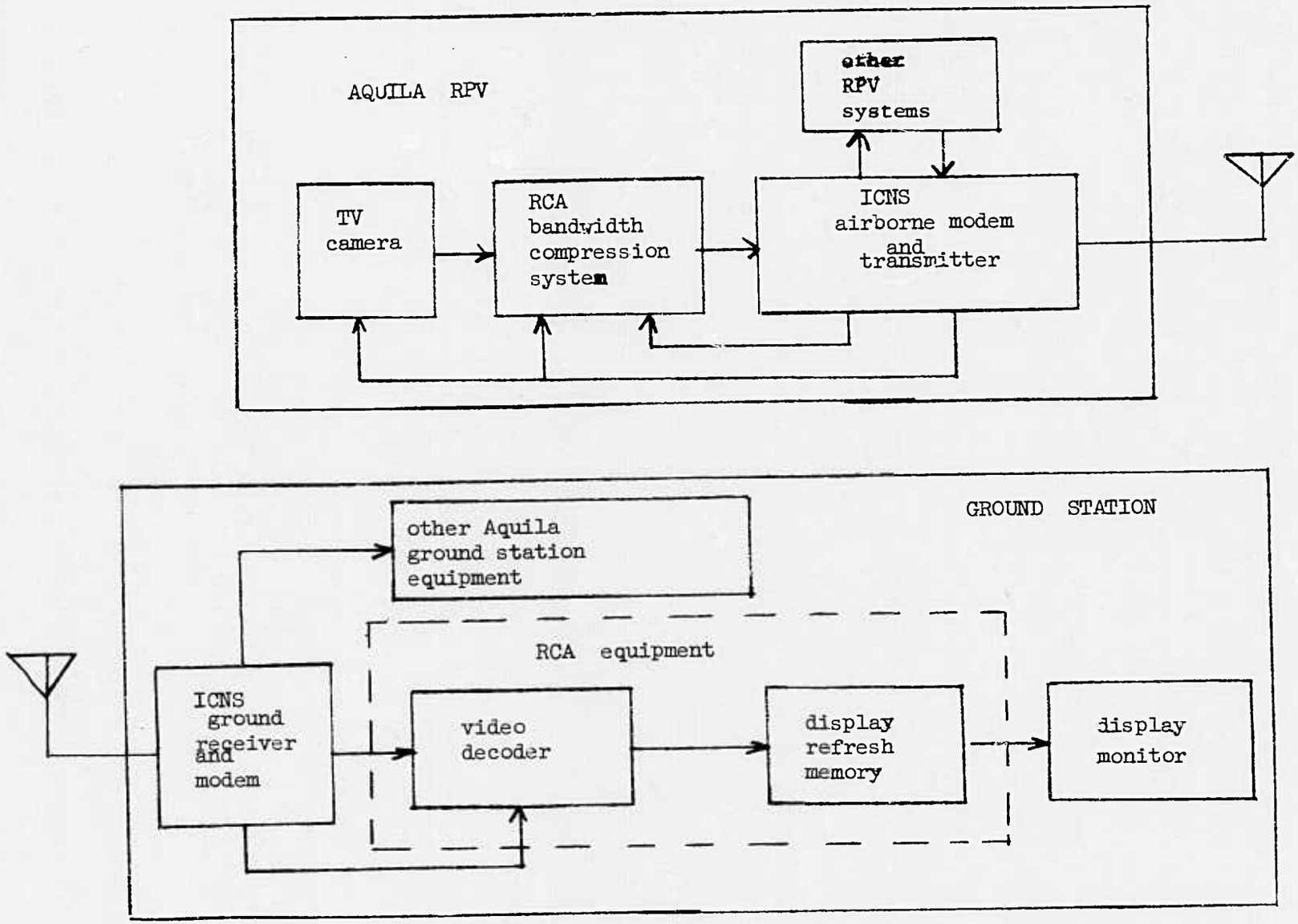


Fig. 1-1. ICNS system diagram.

Best
Available
Copy

The processing techniques to achieve bandwidth compression are: 1) the transformation, in the horizontal direction, of the video to its spatial frequency components by a discrete cosine transform (DCT); 2) the encoding of the transform coefficients in the vertical direction by a differential pulse code modulation (DPCM); and 3) selective quantization of the DPCM words. As a result of this processing, the bit rates for the transmission of the digitized TV are reduced to 3.2, 1.6, 0.8, and 0.4 bits per pixel (selectable). With the associated reduction in resolution to 256 x 256 pixels, and frame rate reduction to 7.5 frames/second, the overall data transmission rates are 1600, 800, 400, and 200 kb/s, corresponding to the four selectable data bit rates.

A particularly noteworthy aspect of the present design is its degree of miniaturization, achieved by a combination of silicon-on-sapphire LSI and advanced hybrid circuit/packaging. The resulting airborne encoder, embodying more than 112,000 equivalent transistors in four hybrid assemblies, is built on a board 5 inches x 9 inches, weighs 0.78 lb, and consumes 13.5 W.

Two identical, miniaturized airborne units configured for installation in the RPV were delivered. A single ground station system configured for installation in a truck transportable field shelter was also delivered.

B. EQUIPMENT FUNCTION

The airborne unit accepts an EIA monochrome television signal from a camera mounted in the RPV, digitizes the video signal, reduces the video frame rate, and uses complex digital data processing techniques to minimize the number of data bits required to supply a signal with an adequate information content to a digital communications modem. This data-compression operation is performed in real time by equipment that consumes 13.5 W of input power supplied by the RPV. Complex hybrid circuit packaging techniques have been used to hold the weight of the airborne unit to less than 0.78 lb and to confine its size to 9 inches x 5 inches.

The ground station accepts the compressed video data signal from the digital communications modem. Inverse processing operations are performed to reconstruct each digitized picture element from the encoded data stream. The decoded pixel data are input to a frame store memory for display on a TV monitor. The memory output data go through digital-to-analog conversion, and then are formatted to a standard EIA monochrome signal for use with a conventional monitor.

C. SUMMARY OF TASKS

The principal tasks of this project are summarized in the following paragraphs.

1. System Simulation and Analysis

The basic system architecture (e.g., discrete cosine transform differential pulse code modulation) and many of the system parameters (e.g., data rates) were specified by NOSC. However, a considerable number of design options were unresolved at the beginning of the program. In view of this, the digital portion of the RCA bandwidth compression hardware was emulated in software. Simulations were run by using this software to determine the effect of various system parameters on image quality. The simulation effort, which is summarized in Sec. III, provided the information required to proceed with the hardware design.

2. Design and Construction of Airborne Unit

Because the space and power available in the RPV are severely limited, the airborne unit was built within severe weight, size, and power consumption constraints. To meet these requirements, the circuitry had to be designed so as to use hybrid techniques to the fullest possible extent. The circuitry was first built as a breadboard using conventional DIP packaging. Extensive testing of the design was accomplished by using the breadboard to compress the video and then evaluating the quality of the decoded video. After the validity of the design was confirmed, a three-layer interconnecting board was designed, the circuits were built as hybrids, and the unit was assembled. Testing of the completed airborne board included waveform analysis, digital testing of the algorithm, and evaluation of compressed video.

3. Design and Construction of Ground Station

The RCA portion of the ICNS ground station includes: 1) a decoder unit to which the Harris modem delivers the encoded video received from the RPV, for restoration to quantized video; 2) a frame store memory, which receives the quantized pixels, reorganizes them into a standard video format, stores and updates a frame of video, performs digital-to-analog conversion and produces a composite video signal for display on a TV monitor, 3) a test fixture and associated circuits for operation of an airborne board into the decoder without connection to the Harris modem. The ground station equipment was built by using conventional techniques, i.e., DIP ICs and wire wrap, with no attempt to minimize size or power.

4. Integrated System Test

Tests were conducted 1) to verify that the digital portions of the system properly implemented the prescribed algorithms and signal-processing functions, 2) to verify proper operation of the analog video input and video output circuitry, and 3) to confirm that the characteristics of the output images were consistent with the predictions made by simulations performed by RCA and by NOSC.

Section II

SYSTEM FUNCTIONS AND ARCHITECTURE

A. OVERALL SYSTEM

The basic function of the overall system is to reduce the amount of data that is required to transmit television images of acceptable quality (i.e., with adequate information content) over a digital communications link from the remotely piloted vehicle to supporting ground equipment.

1. TV Camera

The camera, which is not supplied by RCA, outputs a monochrome video signal to the video signal encoder. This signal is in the composite RS170 EIA format with the exception that the time base is expanded slightly to accommodate a 65.0 μs time interval between successive horizontal line scans.

2. Airborne Unit

This module processes the video input signal to reduce the communication link bandwidth requirement. The principal functions of the airborne unit are:

a) Digitize 32 samples of the video input signal to 6 bits during the active portion of each horizontal line scan. These samples will be in one of the eight segments (stripes) that make up a frame 256 samples wide.

b) Reduce the video field rate by a factor of eight by selecting one vertical stripe, 32 pixels wide, during each video field from the camera. The stripes are sampled from left to right across the video frame such that each portion of the frame is updated once each eight camera fields. The even and the odd camera fields are processed identically (i.e., there is no interlace compensation).

c) Perform a discrete cosine transform (DCT) of the 32 horizontal pixels within a stripe to generate the equivalent 32 frequency coefficients.

d) Multiply each of the coefficients by a weighting factor to standardize the variance of the coefficients.

e) Perform a differential pulse code modulation (DPCM) operation on each of the frequency coefficients that are generated from successive 32 sample horizontal line segments. The output of this operation is the difference, for each of the 32 frequency coefficients, between the present and the preceding lines within a vertical stripe. The difference is quantized into a 6 bit code.

f) Assign word lengths to the 32 quantized DPCM frequency coefficients, format the encoded data, and output this data to the digital communications link modem equipment. The number of bits assigned to each coefficient is based on the importance of the coefficient to image quality and the data rate in use.

Four data rates are available. At the highest data rate (1600 kb/s), 104 bits are used to encode the frequency difference coefficients from 32 video signal samples. At the lower data rates, 52, 26, or 13 bits are transmitted for each of 32 video signal samples.

The digitizing to 256 x 256 samples for each compressed video frame and the frame rate reduction operations establish the basic pixel rate. The discrete cosine transform operation does not, by itself, reduce the number of bits required to represent thirty-two 6-bit pixels since thirty-two 9-bit frequency coefficients are generated. However, unlike the 32 pixel values, which have equal importance, the lower frequency components contain most of the image information. By appropriately quantizing the line to line difference in the amplitude of these frequency coefficients (in the DPCM and coefficient bit assignment operations) the number of bits that is required to transmit the sampled video signal is substantially reduced.

3. Modem Equipment

The Harris Corp. ICNS modem equipment provides synchronization signals for the camera and the airborne unit, and the data rate control signal to the airborne unit. The airborne modem equipment also accepts the encoded video signal from the airborne unit for transmission to the ground. The ground-based modem equipment, also part of the ICNS equipment, processes the received, encoded, video signal and applies this signal to the video bandwidth compressor ground station. The modem equipment also supplies synchronization signals and the data rate control signal to the ground station.

4. Ground Station

The ground station converts the encoded video input signal into a composite video signal that is compatible with a conventional monochrome TV display. Each of the ground station decoding functions is essentially the inverse of the encoding functions performed by the airborne unit, and is performed once during each output video horizontal line scan interval:

a) Format the encoded video signal to supply a set of 32 quantized frequency coefficient differences to the Inverse DPCM Processor.

b) Perform the inverse DPCM operation and deliver a set of approximated frequency coefficients to the inverse DCT processor. The number of coefficients and the accuracy of the coefficient magnitude approximations are functions of the data transmission rate.

c) Perform the inverse DCT operation to generate a group of thirty-two 6-bit decoded pixels. These pixels are an approximate reproduction of a group of 32 adjacent pixels of a horizontal line segment that was input to the DCT processor in the airborne unit.

A random access memory is used to store the 32 decoded pixels that are generated during each horizontal line scan interval. These pixels are written to this frame store memory during the horizontal blanking interval of the video output signal. All of the data for one of the eight vertical stripes are written to memory during each field of the video output signal.

The video data rate is expanded by a factor of eight by reading 256 pixels from the frame store memory during each horizontal line scan interval. This data is supplied to a digital-to-analog converter at a 4.8-MHz rate to recover the video signal. Synchronization signals are added to the video signal to construct a composite video signal that is compatible with a conventional TV monitor.

B. AIRBORNE UNIT

Figure 2-1 is an overall block diagram of the airborne unit that shows the functions that are required to transform the video input signal to a bandwidth-compressed serial bit stream. The functional groupings described are:

1) Synchronization and video signal partitioning
2) Input signal buffering and A/D conversion
3) DCT processing
4) DPCM processing
5) Output signal formatting.

1. Synchronization and Video Signal Partitioning

Synchronous operation is provided by the master timing control in response to a 9.6-MHz clock and synchronization signal supplied by the digital communications link modem. The 9.6-MHz clock is the timing reference for each of the signal-processing functions. The frame sync signal, which is supplied to synchronize the start of each compressed video frame, is also transmitted to the ground station to synchronize the inverse signal processing (decoding) operation.

A data-processing cycle is equal in length to the horizontal line scan time; the beginning of the cycle will be progressively displaced from the horizontal sync signal as vertical stripes 1 through 8 are processed. The frame sync signal will initiate the processing of the first 32-pixel horizontal line segment in the first vertical

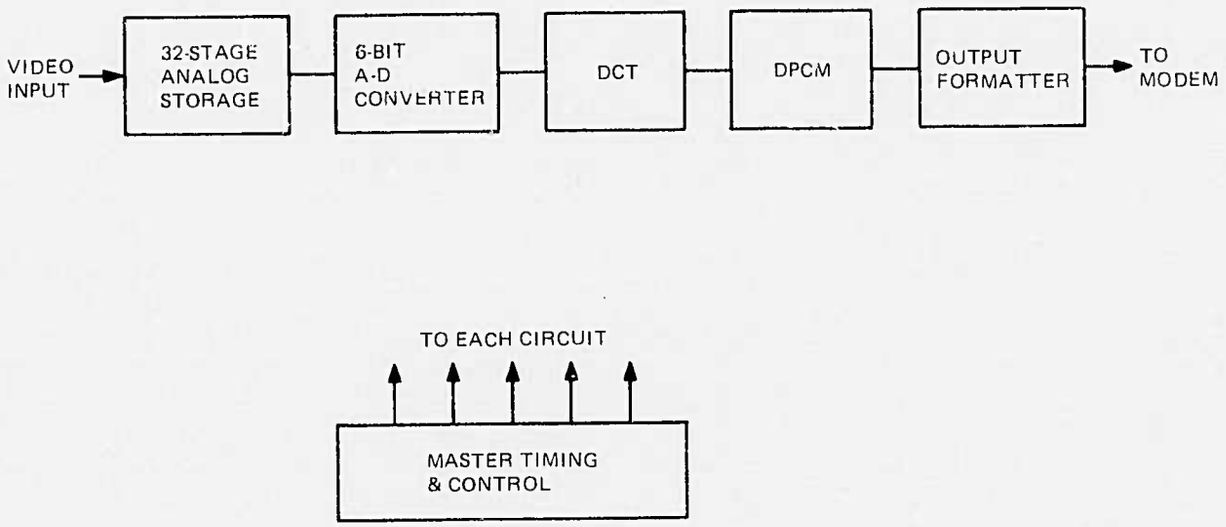


Fig. 2-1. Block diagram of airborne unit.

stripe. The processing operation will progress down the first stripe to digitize and process the first eighth of each horizontal line of the video input signal. The vertical sync pulse that occurs at the end of each video field displaces the start of the processing cycle such that stripes 2 through 8 are similarly digitized and processed prior to the next frame sync signal.

Each horizontal line segment is temporarily stored in a 32-stage analog delay line, digitized, and pipelined through the airborne unit in four sequential operations, each of which is allocated a time slot of up to one horizontal line scan interval (65 μs). The sequential operations are 1) input signal storage and A/D conversion, 2) discrete cosine transform (DCT) processing, 3) differential pulse code modulation (DPCM) processing, and 4) output signal formatting. Figure 2-2 shows the overall timing budget which introduces a delay through the airborne unit of four horizontal line scan intervals.

2. Input Signal Storage and A/D Conversion

The input interface electronics take 32 samples of the video input signal at a 4.8-MHz sample rate once during each horizontal line scan. This requires a 6.7 μs time interval that is displaced from the horizontal sync signal in accordance with the stripe that is being processed. The 32 samples are stored in a 32-stage analog signal

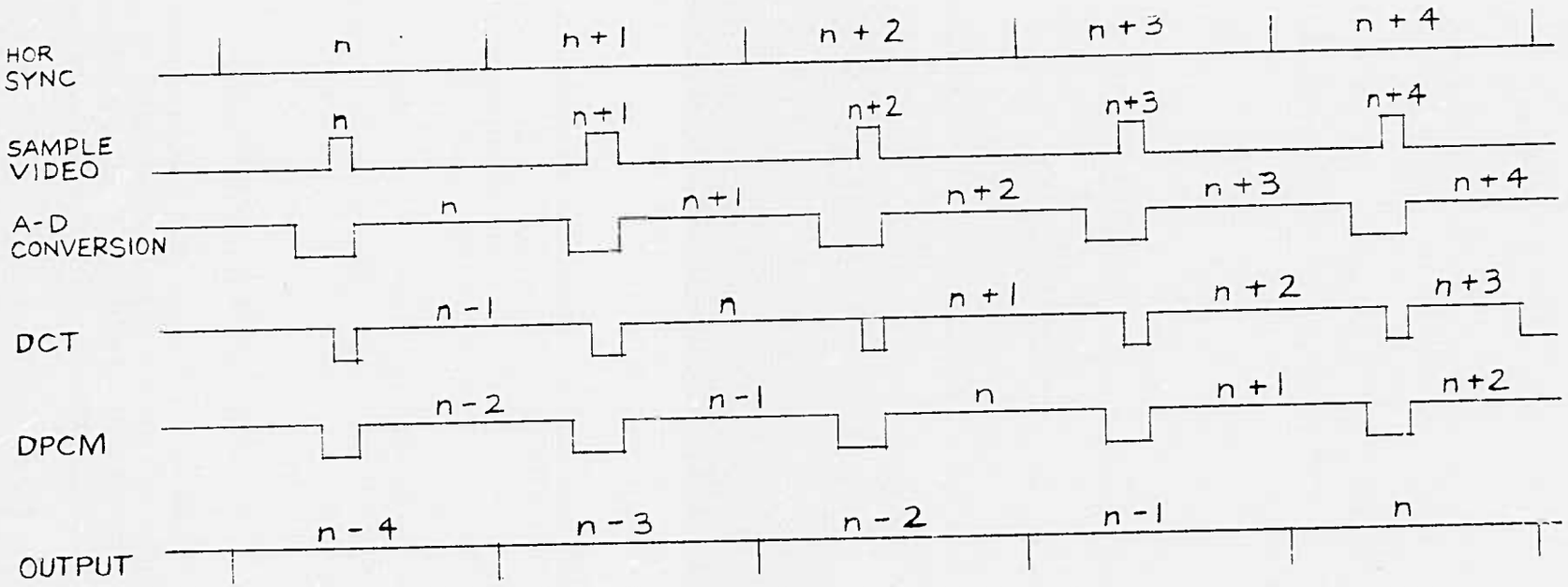


Fig. 2-2. Encoder pipeline timing.

delay line. Immediately after the aequisition proeess is eompleted, 6-bit A/D eonversion of the stored data eommenees. A relatively low rate delay line read elock (4.8 MHz divided by 7) is used to spread the conversion proeess over the remainder of the horizontal line scan interval. This time-base expansion permits the use of a small, low-power A/D eonverter. The 6-bit A/D converter output is stored in a one-word buffer in the DCT proeessor hybrid once eaeh read clock interval. The buffered word is ready by the DCT proeessor prior to completion of the next eonversion cyele.

The timing and control signals for the input interface electronies are supplied by the input interface/master timing hybrid.

3. Discrete Cosine Transform Processing

A simplified block diagram of the DCT processor is shown in Fig. 2-3. This processor performs the arithmetic operations required to eompute the 32 transform coefficients that eorrespond to a group of 32 adjaeent horizontal video signal samples. These samples are stored in the 64-word x 9-bit random aecess memory (RAM). The storage operation for a given horizontal line segment is aeeomplished by interrupting the DCT processing eycle for the immediately preceding horizontal line segment at the 32 points in time when each sample is available at the A/D eonverter output buffer.

The basic DCT operation, which is deseribed in detail in See. III of this report, has been factored into a 6-level, 32-node "butterfly diagram". The input nodes of the first level are the 32 video samples that are stored in the RAM. A value for each node of the seeond level is obtained by performing an AX + BY ealculation, where X and Y are video input values. The multiplying eoeffieients, A and B, are obtained from the two read-only memories. The two product terms are added or subtracted and the result is written to the RAM as a second-level node value. This process eontinues until all of the seeond-level node values are eomputed. The third-level node values are then computed from the second-level node values, and this process eontinues until the 32 sixth-level node values are eomputed. These are the 32 desired frequency coefficients.

The DCT processor has been designed to repeatedly perform a simple straightforward eomputation of (AX + BY) in order to minimize its complexity and therefore its parts count and its power consumption. At those nodes where the algorithm demands only addition or subtraction (i.e., A or B equals 1), the proeessor multiplies by unity rather than deviating from its normal proeedure. Beeause the DCT proeess is redueed, via eareful programming, to a repetition of the basie eomputation, the eomplexity of the circuitry and its attendant power eonsumption are minimized.

The requirements for small size and low power eonsumption, together with the requirement for a 4.8-MHz cloek rate to complete the proeessing eycle within a

Best
Available
Copy

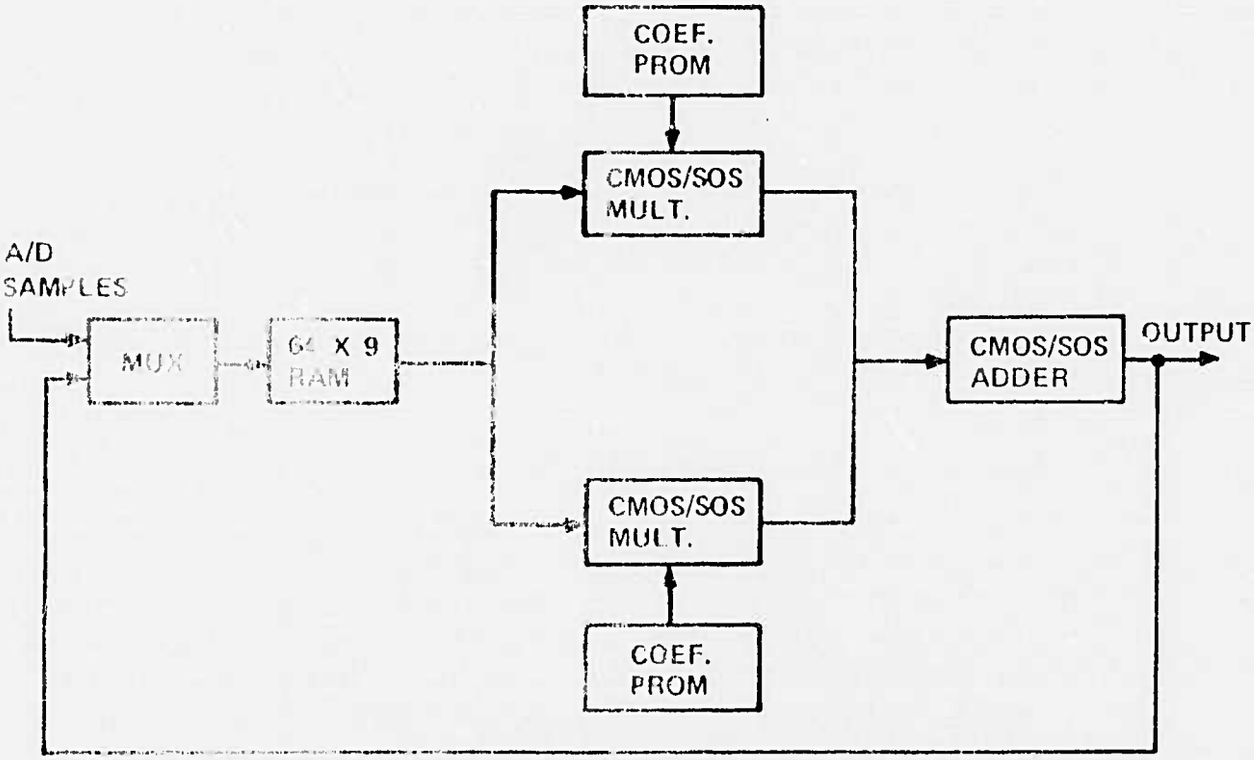


Fig. 2-3. Block diagram of DCT processor.

single horizontal line scan interval, made it necessary to use CMOS/SOS large scale integrated circuit multipliers, adders, and retiming registers. These specialized circuits had been developed by RCA prior to the start of this contract. Section V describes these integrated circuits.

The 32 frequency coefficients are stored in the DCT processor RAM at the end of the computation process. The final portion of the DCT cycle is used to multiply each of the frequency coefficients by normalizing constants that are stored in the DCT read only memories. The higher-order frequency coefficients are multiplied by progressively larger constants. This pre-emphasis normalizes the variance of these coefficients so that a single set of coefficient difference quantizer bounds may be used in the DPCM encoding ROM.

4. Differential Pulse Code Modulation Processing

A simplified block diagram of the DPCM processor is shown in Fig. 2-4. This processor outputs the quantized differences between each of the 32 frequency coefficients and the value of the corresponding coefficient that was computed for the immediately preceding horizontal line segment.

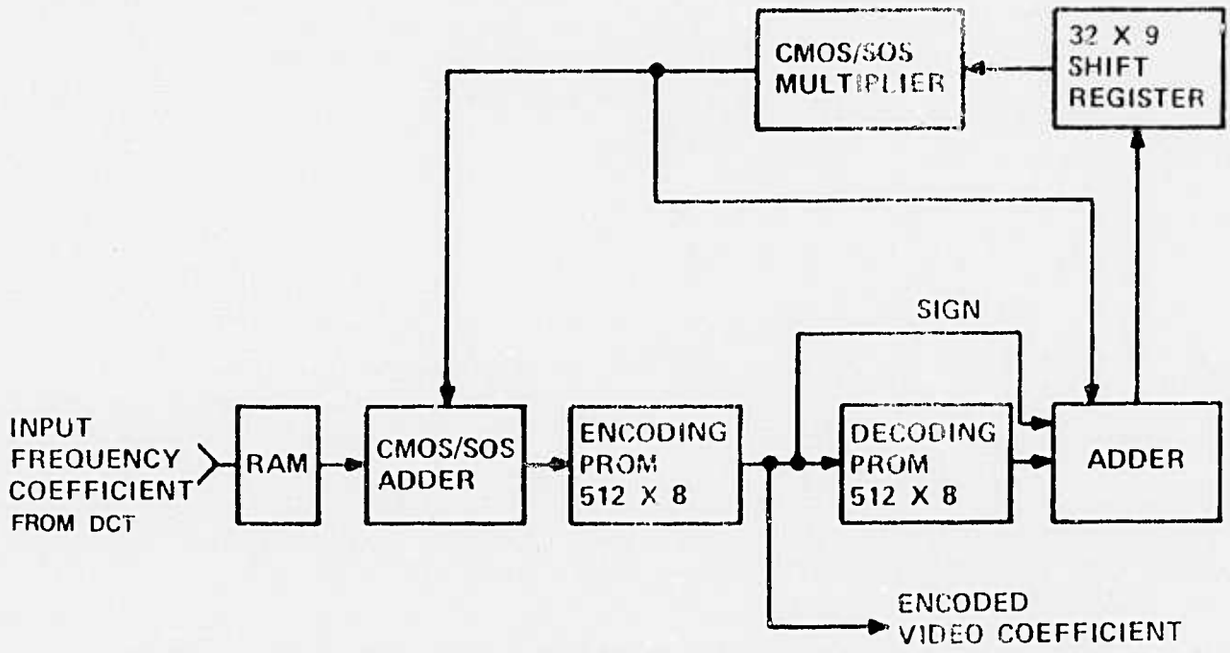


Fig. 2-4. Block diagram of DPCM processor.

The 32 frequency coefficients that are stored in the two DPCM processor RAMs are sequentially applied to the DPCM processor within a horizontal line scan interval. For illustrative purposes, consider the average dc value component which is the first frequency coefficient that is supplied to the CMOS/SOS adder array. The other input to the adder at this portion of the cycle is the reference value of the dc term. The difference is taken between each new coefficient value and the reference value of the corresponding coefficient from the previous line. This difference addresses an encoding ROM, whose output is a quantized, 6-bit approximation to the line-to-line difference. This output, the encoded video coefficient, is forwarded to the output formatting circuitry and is also applied as part of the address input to the decoding ROM within the DPCM processor. The remaining portion of the decoding ROM address is determined by the data rate being used to transmit the encoded video signal from the airborne unit to the ground station. The output of the decoding ROM is identical to the input of the inverse DPCM processor at the ground station when there are no data transmission errors. The decoded magnitude of the differences is applied to an adder. The other input to the adder is the reference value of the coefficient. The output of the adder is the reconstructed value of the coefficient. This output is applied to a shift register that sequentially stores the reconstructed values of the 32 frequency coefficients. The reconstructed values of the coefficients from the shift register, multiplied by an attenuation constant that is less than unity, become the reference values of the coefficients for the succeeding horizontal line segment.

The multiplying attenuation constant in the DPCM feedback path is determined by a set of jumper wires that may be changed in the field from the prewired value of 0.90625. This attenuation of the reconstructed value of each frequency coefficient is required to generate a builtin bias to assure that the inverse DPCM operation at the ground station will properly track the DPCM operation shortly after system startup and shortly after the occurrence of a bit error in the digital data transmission link.

5. Output Signal Formatting

The thirty-two 6-bit encoded video coefficients from the DPCM processor are accumulated within the output interface. The coefficients for an entire horizontal line segment are received and stored as computed, while simultaneously the accumulated data from the previous line segment is being formatted and output for transmission. The start of the accumulation processes is progressively time displaced from the horizontal sync signal, depending upon which of the eight stripes is being processed. However, the timing of the data formatting and output operation is the same for each stripe. This requires three section partitioning of the random access memory so that the variable time displacement between these two functions does not require that data be simultaneously written into and read from the same memory section.

The data-formatting operation is implemented by sequentially transferring the encoded video coefficients to a parallel input/serial output register. The register is advanced at the modem clock rate. The number of coefficients that are transmitted and the number of bits that are transmitted for each coefficient are determined by the modem data rate selected. The coefficient selection and bit allocation processes are controlled by a ROM that selects the memory address to be read and loads this data into the output register after the prescribed number of bits have been transmitted for the immediately preceding coefficients.

The total number of bits that are transmitted during a horizontal line scan interval at each data rate is shown in Table 2-1. The compression ratio shown is

TABLE 2-1. BITS TRANSMITTED

Data Rate (kb/s)	Number of Bits Transmitted	Compression Ratio
200	13	14.77:1
400	26	7.38:1
800	52	3.69:1
1600	104	1.85:1

the ratio of the number of bits that are used to represent the original set of 32 pixels within the noncompressed horizontal line segment (32 pixels x 6 bits/pixel = 192 bits) to the number of bits that are transmitted to the ground station. The assignment of bits among the coefficients is shown in Table 3-4.

C. GROUND STATION

Figure 2-5 is an overall block diagram of the ground station, showing the functions required to reconstruct a video presentation from the received modem output signal. The functional groupings described are:

1) Input signal formatting
2) Inverse DPCM processing
3) Inverse DCT processing
4) Frame store memory
5) Test procedures

Figure 2-6 shows the pipeline processing timing in the decoder.

1. Input Signal Formatting

The signal from the modem is in the form of a serial bit stream, with 13, 26, 52, or 104 bits per line, clocked by the modem clock signal of 200, 400, 800, or 1600 kb/s. The function of the input circuitry is to reformat the data into thirty-two 6-bit words per line, filling in zeros where the original data were not transmitted. This is done by a serial-to-parallel converter under the control of a PROM, which has four sets of instructions for the four data rates. A dual RAM buffer permits one line to be assembled while the preceding is being delivered to the inverse DPCM.

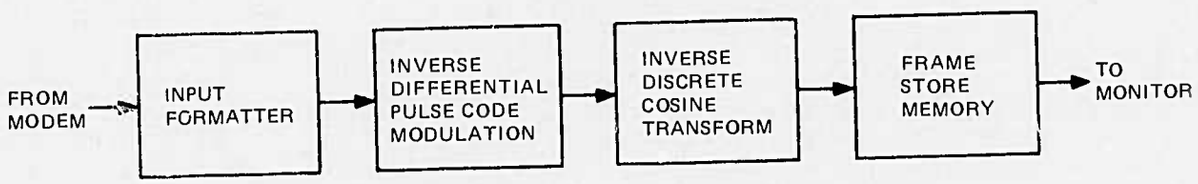


Fig. 2-5. Block diagram of ground station.

27

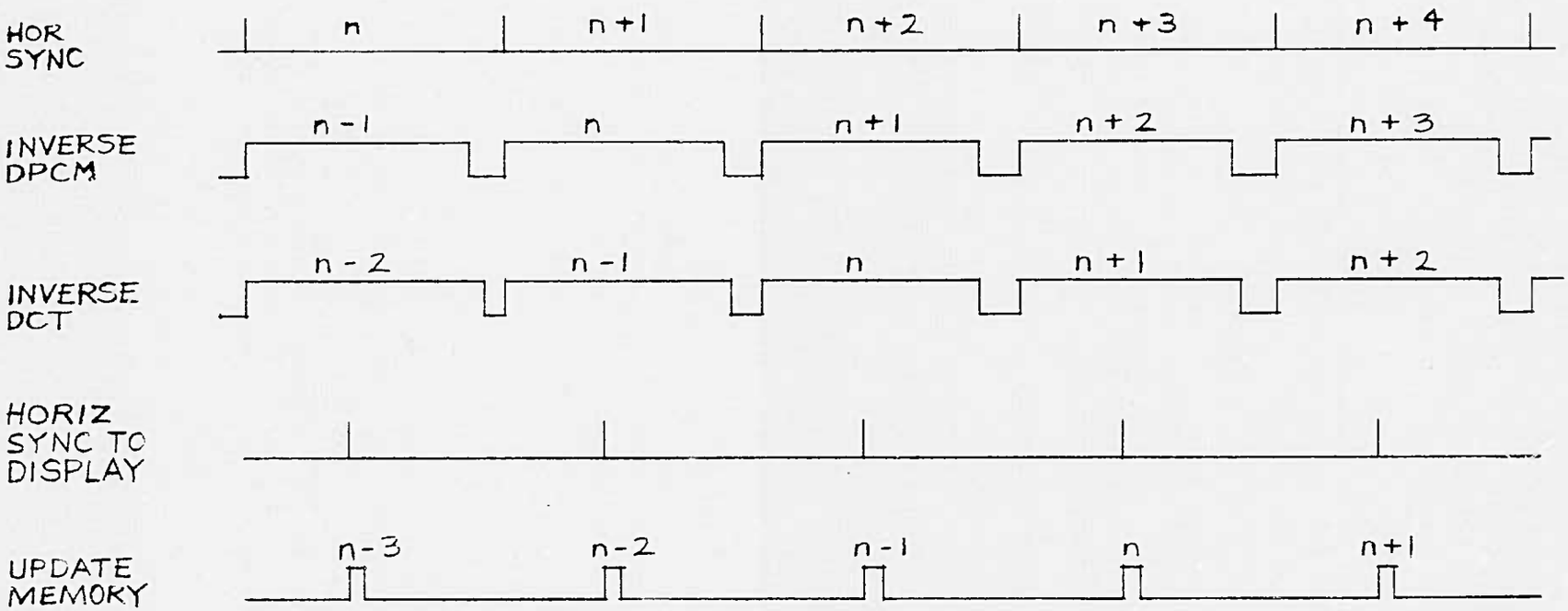


Fig. 2-6. Decoder pipeline timing.

2. Inverse Differential Pulse Code Modulation

A block diagram of the IDPCM is shown in Fig. 2-7. The words from the input circuits, which contain many nonsignificant bits because of the bits discarded in the bandwidth-compression process, are first applied to a decode PROM. This PROM is the inverse of that in the airborne encoder, which determined a quantized differential value after a comparison between the DCT output and the corresponding processed value from the previous line. The decode PROM, therefore, estimates a difference value from the differential word. A modifier PROM alters the decode values in accordance with the data rate. The decode difference is added to the previously accumulated coefficient value, after that value is modified by the attenuation factor. The resulting sum is sent to the inverse DCT and is also stored in a 32-stage shift register for comparison with the corresponding value in the next line.

Before the decoded coefficient value is released to the inverse DCT, it may be modified by the guard-band circuits. This circuit attempts to reduce noise in the picture by setting to zero all coefficient values below a minimum value, thereby eliminating the inherent noise of the DPCM process around low-level signals. The minimum boundary value is chosen differently for the various coefficients and data rates. The operation of the guard band is optional and may be disabled by a switch. When it is not in use, all the outputs of the IDPCM loop are routed directly to the inverse DCT circuit.

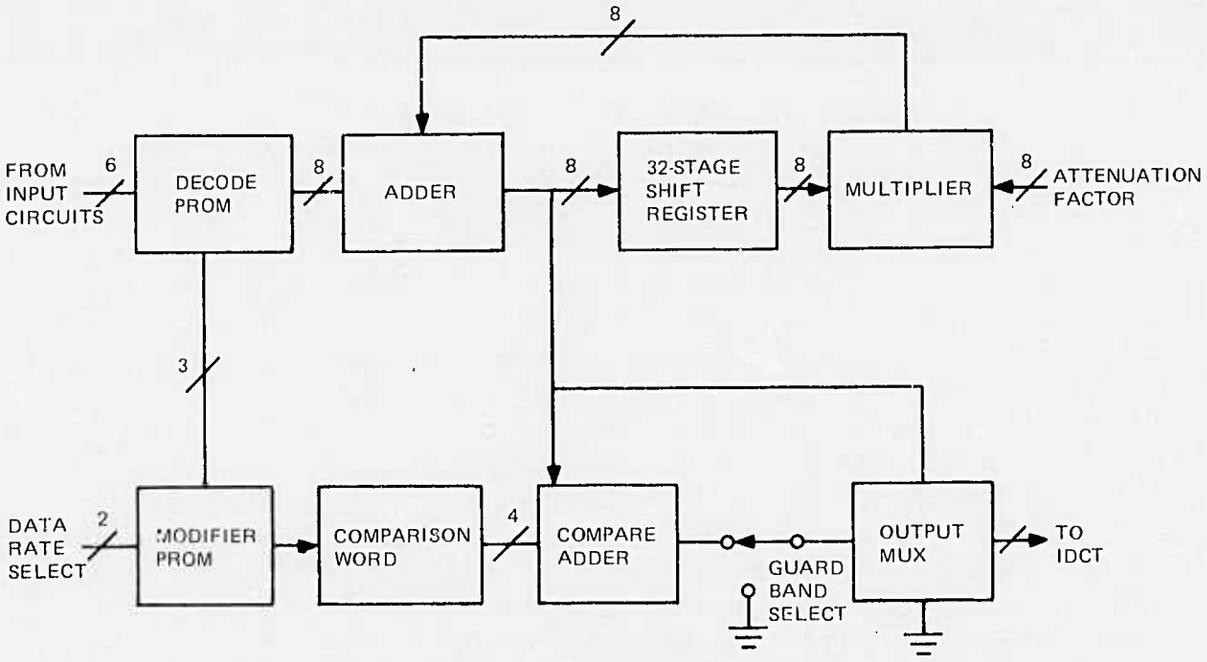


Fig. 2-7. Block diagram of inverse differential pulse code modulator.

29

3. Inverse Discrete Cosine Transform

The inverse DCT circuit converts the decoded coefficients to an approximation of the original video signal. The reconstructed signal can be only an approximation of the original because the quantizing in the DPCM process and the discarding of information in the bandwidth-reduction process prevent an exact reconstruction.

The IDCT circuit is very similar to the forward DCT. The IDCT operation, including inverse whitening, has been factored into a 6-level, 32-mode matrix operation. It is implemented in pipeline fashion by a RAM, two multipliers, an adder, storage ROMs, and a control ROM. CMOS/SOS LSI multipliers, adders, and retimers identical to those in the airborne unit are employed. After the inverse DCT process is complete, the reconstructed video is sent to the frame store memory.

4. Frame Store Memory

The frame store memory is a 32K x 12 NMOS RAM from the Electronic Memories and Magnetics Corporation (EMM). It stores the 6-bit video for a 256 x 256 picture. Thirty-two new 6-bit words are received from the IDCT during each horizontal line time and are used to update the memory during horizontal retrace. The memory also contains a D/A converter, a TV sync generator, and circuitry for assembling a composite video signal.

5. Test Procedures

Several test capabilities have been incorporated into the ground station. The frame store memory includes an EMM test card which can exercise the memory and check all its locations for proper operation. The control circuitry for the frame store memory has provision for insertion of a signal which mimics the output of the IDCT. It checks out frame store memory control circuitry and video circuitry as well as the memory itself. In the ground station is a generator of a digital data pattern which can be inserted at the input formatter. It checks out the entire ground station — input, IDPCM, IDCT, and FSM, and produces a known pattern on the monitor screen. The ground station also includes an airborne unit test capability. A test fixture is provided on which an airborne unit can be mounted, with appropriate voltages and control signals provided. A video camera output may then be fed into the airborne unit, be encoded, be decoded by the ground station, and the resulting video displayed on the monitor. A test function digital signal may also be injected into the airborne unit at the input to the DCT. The response of the digital circuitry to this test function has been emulated on a computer and the digital words at various points in the encoder-decoder have been tabulated. With the aid of a logic analyzer, the response of the circuits to the test function may be observed and compared with the computer printout.

Section III

COMPRESSION ALGORITHM

A. HORIZONTAL TRANSFORM

The horizontal transform which has been utilized is the discrete cosine transform defined by the equation:

$$F_k = \frac{1}{N} \sum_{n=o}^{N-1} f_n \cos \frac{2\pi\,(n+1/2)K}{2N}, \quad \text{where } K = 0, 1, \ldots N-1; \qquad (3\text{-}1)$$

and the corresponding inverse defined by

$$f_n = F_o + 2 \sum_{K=1}^{N-1} F_k \cos \frac{2\pi\,(n+1/2)K}{2N}, \text{ where } n = 0, 1, \ldots N-1. \qquad (3\text{-}2)$$

These equations were derived from the complex Fourier series of length 2N, assuming even symmetry about a point midway between the N and N+1st samples.* A phase shift has been built into the equation such that the samples lie in the center of the sampling interval. The matrix form of Eq. 3-1 can be factored into M+1 factors where $N = 2^M$. At first this appears to be one more factor than normally required for an FFT, but it should be remembered that the discrete cosine transform (DCT) of N points corresponds to an equivalent FFT of 2N points, which is inherent in the equations.

B. COEFFICIENT STANDARDIZATION

After the DCT coefficients are computed, their amplitudes must be adjusted in a known way prior to beginning the DPCM. The proper selection of this coefficient weighting is highly important.

Although in the DPCM it is actually coefficient differences that are quantized and transmitted, any scaling that is done at the DCT output is directly reflected in the differences. The power spectrum of any arbitrary image decreases with increasing

---

*There are a number of published algorithms for the discrete cosine transform involving both odd and even number of samples. One of the main differences is the definition of how the original function is to have its symmetry positioned.

frequency. This is directly related to the variance of the transform coefficients. Figure 3-1 shows the coefficient variances (V_i) plotted as a function of coefficient number for the 6-bit PCM scene shown in Fig. 3-2a.

Since the variance of the coefficients decreases with increasing frequency, the successive differences of coefficients also decrease with increasing frequency. If the coefficients are scaled such that each one has the same variance, then the coefficient differences taken line to line will also have nearly equal variance. This means that the difference distribution for each coefficient covers approximately the same dynamic range. Consequently, all coefficients can be processed using the same quantizer.

Figure 3-3 shows the coefficient weighting function (W_i) required to equalize the variances. The filter W_i is obtained from the coefficient variances by computing $(V_o/V_i)^{1/2}$ for i = o . . . N-1. The square root is involved since the weighting filter is applied to the coefficients in the frequency plane and the variance is a function of the square of the coefficients.

In all but the simplest of applications it is not practical to compute the coefficient variances in order to define the weighting filter. Certainly in the RPV application, which is a real-time process, the filter must be predefined. If the MTF of the system were limiting the power spectrum of the images, then the filter could readily be defined from a knowledge of the MTF. In general, however, the spectrum of an image will fall off faster (versus frequency) than the MTF of the system.

NOSC personnel suggested early in this program a filter which was proportional to the coefficient index. This appeared to perform reasonably well in the RCA simulations and has consequently been utilized in the system.

C. DPCM ENCODER

Figure 3-4 is a block diagram of the differential pulse code modulation (DPCM) encoder. It is in this system block that the bandwidth reduction takes place. Prior to the DPCM, the coefficients produced by the cosine transform process could be inverted to obtain the exact input. The DPCM process, however, introduces an approximation to the coefficients which produces the bandwidth reduction but at the same time prevents exact invertibility.

The DPCM operates as follows. Input coefficients are differenced with a fraction (a) of the corresponding coefficient from the previous lines. This difference is then approximated by one of 2^K levels in the quantizer. This quantized difference is then added to the fraction (a) of the previous line's coefficient and stored in memory for use in processing the next line.

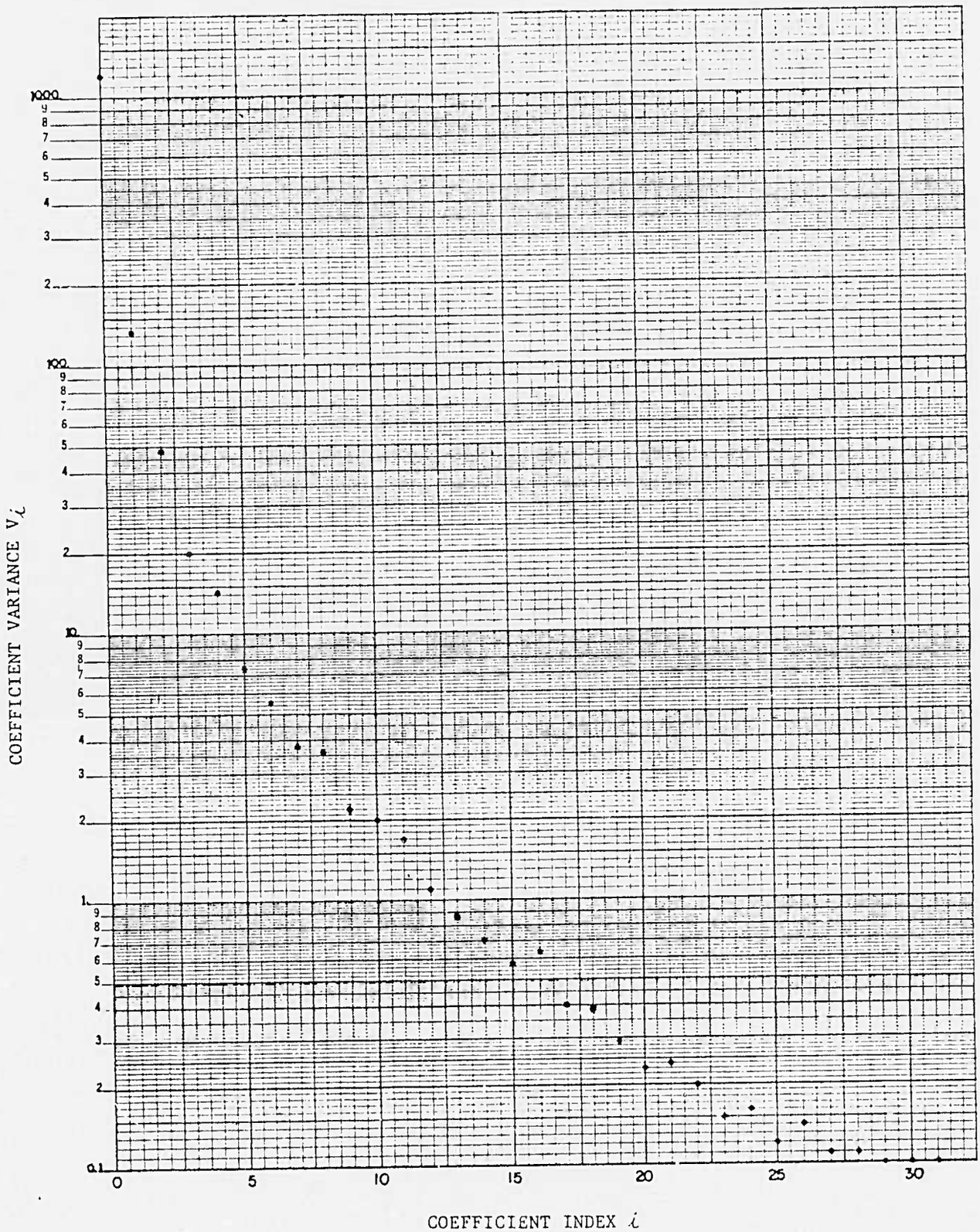


Fig. 3-1. Coefficient variances.

33

a. Original image.

b. Reconstructed image.

Fig. 3-2. Original and reconstructed images.

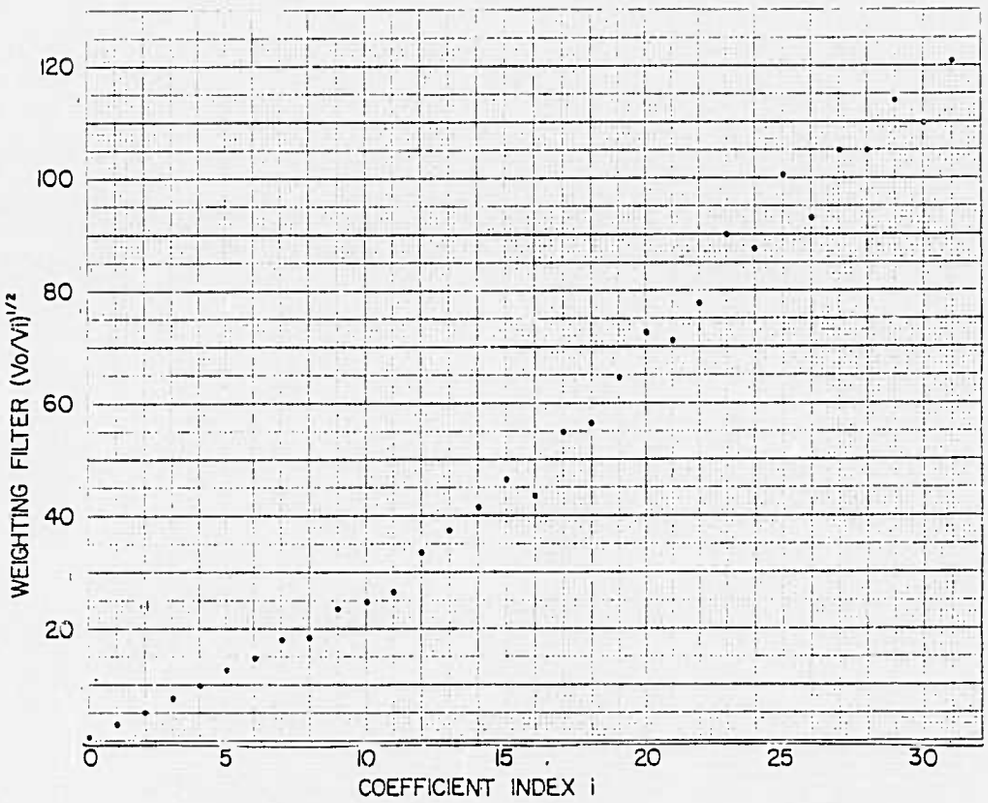


Fig. 3-3. Coefficient weighting which normalizes variances.

34

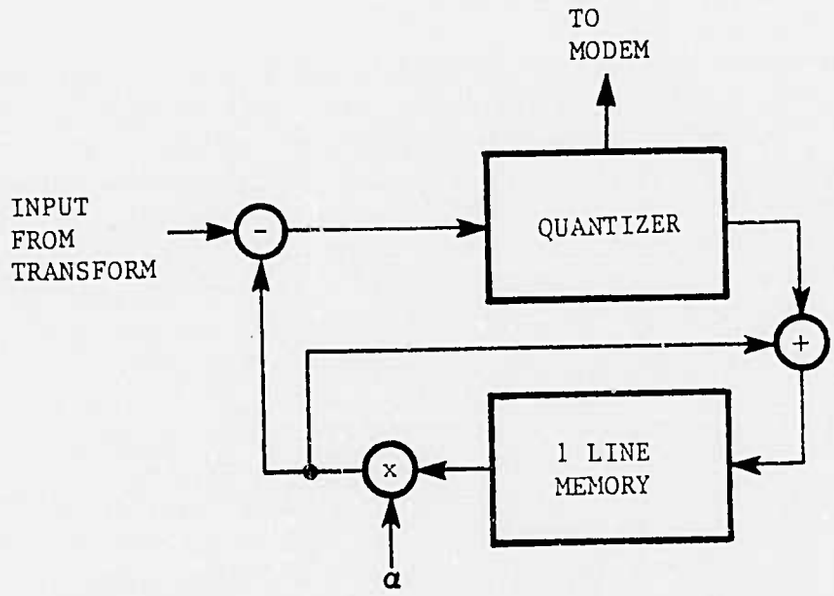


Fig. 3-4. DPCM encoder diagram.

Also, during the quantization process the K-bit number which defines which of the 2^K levels has been used is passed to the transmission link via a modem. By varying the value of K for successive coefficients, any desired average bit rate can be achieved. Typically, for 6-bit PCM input data, K would be chosen as five or six for dc and be decreased to as low as one for the higher frequency coefficients.

D. IMAGE RECONSTRUCTION

The process required at the receiver is the straightforward inverse of the encoder operation. The filter used to normalize the frequency coefficients before DPCM must be used to generate an inverse filter to be applied in the expander. Figure 3-2b is an example of an image encoded at approximately 3.2 bits/pixel using the weighting function in Fig. 3-3. The reason for using such a high bit rate in this example is to ensure that all coefficients are used in the process since at lower bit rates some of the higher-frequency coefficients are discarded. This seems reasonable since the purpose of this example is to show that indeed a weighting filter can be utilized to equalize the coefficient variances prior to quantization in the DPCM system. Bit assignments for this picture ranged from five bits at dc and low-frequency coefficients to one bit at the six highest frequencies. The single quantizer used is a nonlinear 5-bit quantizer with a maximum reconstruction level of ±33. The 1-bit encoder simply takes the most significant bit of the 5-bit quantizer. The fraction (α) in the DPCM loop for this test was set at 0.9.

E. EFFECTS OF LIMITED PRECISION

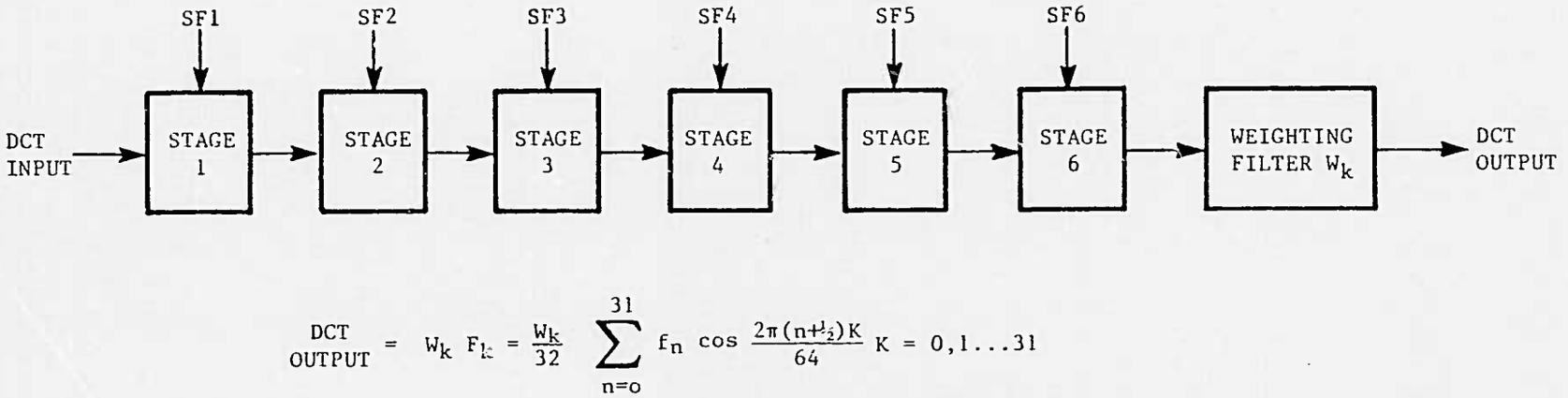
Most bandwidth-compression simulation is done in floating-point arithmetic. The hardware needed to implement floating point is larger and more complex than that required for integer arithmetic. For minimum equipment size and weight in an RPV, integer implementation is desirable. However, it then becomes necessary to determine the effects of limited precision arithmetic. Consequently, an investigation was undertaken to define the scaling required in the forward and inverse DCT such that not only is the mathematics satisfied, but the set of numbers presented to the DPCM hardware fill the quantizer dynamic range reasonably well. This must be accomplished without risking excessive overflow at any stage along the way.

In order to investigate this scaling problem, a software simulator was constructed for the forward and inverse transform portions of the system including the prewhitening (coefficient standardization) filter and its inverse. Initially, this was done in floating-point arithmetic but was later converted to integer with the multiplier hardware being emulated. The particular multiplier being used is the 8x8 CMOS/SOS TCS057. The CMOS/SOS adder TCS065 is also emulated in the sense that the divide-by-2 feature of the adder, which is controlled by an external input, was included in the simulation. It is this feature which is used to perform the scaling in the system. The principal task in this investigation was to determine which stages in the DCT should perform this divide-by-2 operation.

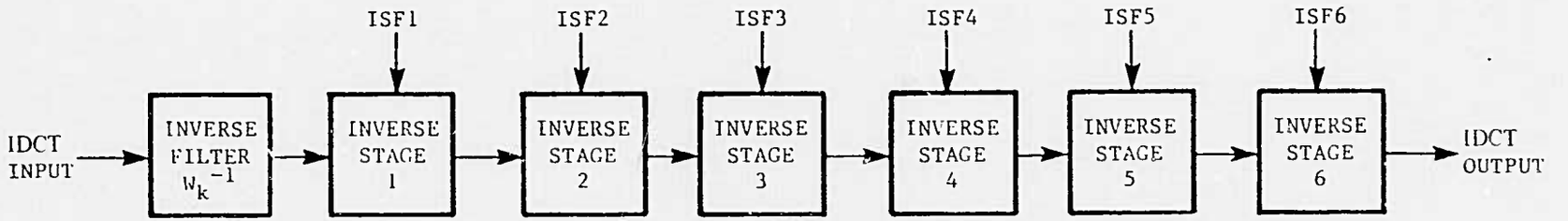
Figure 3-5 is a block diagram showing each of the stages in the forward and inverse DCT processing, including the frequency-dependent weighting function. The operations implied by each block labeled 'STAGE N' are all the computations needed to generate 32 outputs from 32 inputs, where each output is a linear combination of two inputs. Each stage has an external input (SF or ISF) which represents the divide-by-2 control to the adder. When SF (or ISF) is 1, the output of the stage is unchanged. If, however, SF (or ISF) is 1/2, the stage output is divided by 2.

The scaling which has to be accomplished in the forward chain is $W_k/32$, as shown in Fig. 3-5. When the maximum dc level of 63 is input to the system, the result is an output dc coefficient of 63. Since the DCT utilizes 8-bit plus sign hardware, it is advantageous to scale this up by a factor of four to preserve the maximum amount of significance at the output of stage 6 in the forward transform. Consequently, it is desirable to introduce a scale factor of 1/8 via the inputs SF1 through SF6 in Fig. 3-5 (rather than 1/32 as indicated in the upper equation).

Table 3-1 shows the output at each node of every stage in the forward and inverse transform with a flat field of unity magnitude and no scaling. In order to avoid overflow at any stage in the forward transform, the table suggests that the output of stages 3, 4 and 6 should be divided by 2. This provides the desired scaling of 1/8. A further check can be made by passing a signal through the system to produce the maximum fundamental frequency coefficient.



$$\text{DCT OUTPUT} = W_k F_k = \frac{W_k}{32} \sum_{n=o}^{31} f_n \cos \frac{2\pi(n+\frac{1}{2})K}{64} \quad K = 0,1\ldots31$$



$$\text{IDCT OUTPUT} = \frac{1}{2} f_n = \frac{1}{2}(W_o^{-1} F_o) + \sum_{K=1}^{31} (W_k^{-1} F_k) \cos \frac{2\pi(n+\frac{1}{2})k}{64} \quad n = 0,1\ldots31$$

Fig. 3-5. Forward and inverse DCT block diagram.

37

TABLE 3-1. DCT STAGE GROWTH FOR FLAT FIELD INPUT OF UNITY MAGNITUDE

Node	Stage Number											
	1	2	3	4	5	6	1^{-1}	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}
0	1	2	4	8	16	32	16	16	16	16	16	16
1	2										16	16
2												16
3	2										16	16
4												16
5	2										16	16
6												16
7	2										16	16
8				8				16				16
9	2										16	16
10												16
11	2										16	16
12												16
13	2										16	16
14												16
15	2	2								16	16	16
16												16
17	2	-4								-16	16	16
18												16
19	2	-4								-16	16	16
20												16
21	2	-4								-16	16	16
22												16
23	2	-4								-16	16	16
24												16
25	2	-4	-8						-16	-16	16	16
26												16
27	2	-4								-16	16	16
28												16
29	2	-4	-8						-16	-16	16	16
30												16
31	1			-16	16		16	-16			16	16
32												
33			-8						-16			
34												
35												
36												
37			-4						-16			
38												
39												

The maximum ac coefficient is achieved when the input function has a maximum amplitude wherever the basic function is positive and is zero wherever the basic function is negative. As an example, consider a square wave with a 50 percent duty cycle. From Fourier series theory it is known that this produces a fundamental amplitude of $2A/\pi$ where A is the square wave peak-to-peak value. When A equals 63, as in our 6-bit system, this results in a maximum fundamental frequency amplitude equal to $63*2/\pi$. When our processing gain of four to one is taken into account, this produces an expected output of $63*8/\pi$, which is two times greater than is actually obtained. The explanation for this is as follows. The Fourier series theory which predicts the fundamental amplitude of $2A/\pi$ is based upon the definition:

$$f(\theta) = \frac{1}{2} b_o + \sum_{n=1}^{\infty} b_n \cos(n\theta) + \sum_{n=1}^{\infty} a_n \sin(n\theta). \qquad (3\text{-}3)$$

However, the DCT equation which has been implemented was derived from the complex Fourier series definition:

$$f(\theta) = \sum_{n=-\infty}^{\infty} C_n e^{in\theta}. \qquad (3\text{-}4)$$

These coefficients are related by

$$C_n = \frac{a_n + ib_n}{2i}, \qquad (3\text{-}5)$$

which for an even symmetric function reduces to

$$C_n = \frac{1}{2} b_n. \qquad (3\text{-}6)$$

Since it is the coefficients C_n which the hardware is computing, the results are perfectly consistent with theory. Furthermore, with the scaling defined as 1/2 in stages 3, 4, and 6, it is guaranteed that in the 8-bit hardware with maximum ac signals there will be no overflow at the output of stage 6. This can be seen from Table 3-2 which shows the gain at each node for a unity-amplitude 50 percent duty cycle square wave into the system. Since all outputs of stage 6 are less than 32, there will be no overflow in 8-bit hardware when the scaling of 1/8 is included. Furthermore, by performing the scaling by dividing by 2 in stages 3, 4, and 6 as previously defined, no overflow will occur in any stage. To further specify this scaling, histograms of individual stage outputs were generated for three different input images and no overflow resulted.

TABLE 3-2. DCT STAGE OUTPUT FOR MAXIMUM FUNDAMENTAL FREQUENCY

Node	Stage Number											
	1	2	3	4	5	6	1^{-1}	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}
0	1	1	2	4	8	16	8	8	8	8	16	16
1	2	2		5.13	10.15	10.19	10.15	10.25		8	16	16
2			2.35						10.25			16
3	2	2	-1.57	-1.8	-3.3	-3.41	-3.3	-3.6	-2.04	8	16	16
4			.55	0					-3.6			16
5	2	2	2.77	1.2	1.87	2.06	1.87	2.4	2.4	8	16	16
6			2.61	0					10.45			16
7	2	2	-1.08	-1.0	-1.22	-1.48	-1.22	-2.04	-4.33	8	16	16
8			1.57	4				8	-2.41			16
9	2	2	2.35	-.20	.82	1.17	.82	.40	3.60	8	16	16
10			2.77	0					10.25			16
11	2	2	-.55	.67	-.53	-.97	-.53	1.34	2.03	8	16	16
12			2.41	0					9.66			16
13	2	2		-1.5	.30	.84	.30	-2.99		8	16	16
14				0								16
15	1	1		5.03	-.10	-.74	-.10	10.05		8	8	16
16	1	1			-1		-1			8	8	
17		-2				.67				-8		
18					1		1					
19		-2				-.62				-8		
20					-1		-1					
21		-2				.58				-8		
22					1		1					
23		-2	1			-.55			4	-8		
24			-1	1	-1		-1	2	-4			
25		-2	-4			.53			-8	-8		
26				-1	1		1	-2				
27		-2				-.52				-8		
28				1	-1		-1	2				
29		-2	-4			.51			-8	-8		
30				-1			1	-2				
31		1		-8	8	-.50	8	-8		8		
32												
33			-4						-8			
34												
35												
36												
37			-2						-8			
38												
39			.41						1.66			

40

The defined weighting filter, W_k, is unity at dc since the 8-bit dynamic range could be fully utilized at the stage 6 output. The ac coefficients are weighted by the function (K+2), where K is the coefficient index. This implies that there is a distinct possibility of overflow after the filter, but is rather unlikely since the typical image spectrum decreases as a function of frequency.

The inverse process must remove the filter as well as the 4-to-1 gain inherent in the forward process. In addition, there is another factor of 2 which is included in the inverse process to compensate for the fact that the matrix factorization which is implemented in the DCT produces 1/2 f_n at the output rather than f_n. (The IDCT produces 1/2 f_n because this allows the same factors to be used in forward and inverse process.) The resultant scaling which is done in the inverse operation is therefore:

$$\frac{2}{4\ (K+2)} \quad . \tag{3-7}$$

The factor (K+2) must be removed prior to the first inverse stage since only at this point is the data separated in frequency. The remaining factor of 1/2 which must be accounted for is left until the last stage in order to preserve a maximum amount of significance through the inverse system. A check on the scale factors for the forward and inverse DCT shows that their product is 2/N, which is correct for unity gain through the two processes.

The final scaling and filter selection are summarized in Table 3-3. This final scaling selection would probably be modified in a future system. The maximum ac coefficient obtainable is approximately 80 with the defined scaling. This is not preserving the maximum amount of significance in the system. To improve this situation implies two changes to the system. First the ability to scale individual nodes rather than all nodes at a given stage is required. This would allow scaling the ac coefficients differently than the dc. This would allow scaling only the dc in stage 6 and would provide one additional bit of significance in the ac coefficients. The second requirement is that the high-speed adders should saturate at 8 bits rather than wrap around as they currently do. This would allow coefficients to exceed 8 bits without introducing severe artifacts.

TABLE 3-3. SCALING AND FILTER DEFINITIONS

DCT	Filter	Filter^{-1}	IDCT
SF1 = 1	$W_o = 1$	$W_o = 1$	ISF1 = 1
SF2 = 1	$W_k = (K+2)$	$W_k^{-1} = 1/K+2$	ISF2 = 1
SF3 = 1/2			ISF3 = 1
SF4 = 1/2			ISF4 = 1
SF5 = 1			ISF5 = 1
SF6 = 1/2			ISF6 = 1/2

F. BIT ASSIGNMENT VS. DATA RATE

The horizontal line time in the ICNS system is 65 μs. At the modem clock rates of 1600, 800, 400, or 200 kb/s there are therefore 104, 52, 26 or 13 bits per line.

The bit assignments used to achieve this compression are given in Table 3-4.

TABLE 3-4. DPCM BIT ASSIGNMENTS FOR FOUR DATA RATES

Coefficient	Rate (kb/s)				Coefficient	Rate (kb/s)			
	1600	800	400	200		1600	800	400	200
1	5	4	3	2	17	3	2	0	0
2	5	3	2	2	18	3	1	0	0
3	5	3	2	1	19	3	1	0	0
4	5	3	2	1	20	3	1	0	0
5	5	3	2	1	21	3	1	0	0
6	5	3	2	1	22	3	1	0	0
7	5	3	2	1	23	2	1	0	0
8	5	3	2	1	24	2	0	0	0
9	4	3	2	1	25	2	0	0	0
10	4	3	1	1	26	2	0	0	0
11	4	3	1	1	27	1	0	0	0
12	4	2	1	0	28	1	0	0	0
13	4	2	1	0	29	1	0	0	0
14	4	2	1	0	30	1	0	0	0
15	4	2	1	0	31	1	0	0	0
16	4	2	1	0	32	1	0	0	0

42

G. QUANTIZER DEFINITION

A nonlinear quantizer is used to encode the difference signal in the DPCM processor. The maximum number of bits assigned to any coefficient is 5. Consequently a 5-bit quantizer was designed. One of two choices had to be made for the quantizer implementation. Either different quantizers had to be generated for the 5-, 4-, 3-, 2- and 1-bit encoders or the quantizer had to be collapsible in the sense that the high-order N bits of the 5-bit quantizer became the N-bit code word. The single quantizer approach was adopted in order to minimize the amount of storage required in the airborne unit. Table 3-5 defines the 5-bit nonlinear quantizer code words as well as the corresponding reconstruction levels for the 5-bit through 1-bit quantizers.

TABLE 3-5. NONLINEAR QUANTIZER

Quantizer Input	5-Bit Code	Ground Station Quantizer Values				
		5-Bit	4-Bit	3-Bit	2-Bit	1-Bit
0	00000	0	1	2	4	8
1	00001	1	1	2	4	8
2	00010	2	3	2	4	8
3	00011	3	3	2	4	8
4	00100	4	5	6	4	8
5	00101	5	5	6	4	8
6	00110	6	7	6	4	8
7	00111	7	7	6	4	8
8-9	01000	9	10	12	17	8
10-11	01001	11	10	12	17	8
12-13	01010	13	14	12	17	8
14-16	01011	15	14	12	17	8
17-19	01100	18	20	24	17	8
20-23	01101	22	20	24	17	8
24-28	01110	27	29	24	17	8
⩾29	01111	33	29	24	17	8
-1	10001	-1	-1	-2	-4	-8
-2	10010	-2	-3	-2	-4	-8
-3	10011	-3	-3	-2	-4	-8
-4	10100	-4	-5	-6	-4	-8
-5	10101	-5	-5	-6	-4	-8
-6	10110	-6	-7	-6	-4	-8
-7	10111	-7	-7	-6	-4	-8
-8 to -9	11000	-9	-10	-12	-17	-8
-10 to -11	11001	-11	-10	-12	-17	-8
-12 to -13	11010	-13	-14	-12	-17	-8
-14 to -16	11011	-15	-14	-12	-17	-8
-17 to -19	11100	-18	-20	-24	-17	-8
-20 to -23	11101	-22	-20	-24	-17	-8
-24 to -28	11110	-27	-29	-24	-17	-8
⩽-29	11111	-33	-29	-24	-17	-8

H. SIMULATOR OUTPUT SAMPLE

Figure 3-6 shows the type of results predieted from the bandwidth eompression hardware at the four data rates. The mean square error (MSE) for these images is given in Table 3-6. Very little difference is seen between the images simulated at the 1600 kb/s and 800 kb/s data rates. Degradation at the 400 kb/s rate is notieeable and at 200 kb/s is quite bad. The 200-kb imagery is, however, quite good when considered in terms of high jamming environment.

The simulator used to generate these images eomputes the forward and inverse DCT outputs by using the same six stages of eomputation defined for the hardware. The multiplier and adder are emulated in order to produee results eomparable to the hardware output. The results shown here appear to be quite close to those observed in the laboratory by freezing a frame. However, beeause the noise from frame to frame in an operating system is not stationary, the imagery looks somewhat worse than in single frames.

TABLE 3-6. MEAN SQUARE ERROR FOR IMAGES IN FIG. 3-6

Average Bit Rate (bits/pixel)	Data Rate (kb/s)	MSE (%)
6	3000	0
3.2	1600	0.33
1.6	800	0.53
0.8	400	1.12
0.4	200	2.32

44

ORIGINAL 6 BITS

3.2 BITS

1.6 BITS

0.8 BITS

0.4 BITS

Fig. 3-6. Original and reconstructed images.

Section IV

SYSTEM TESTS

System tests consisted of subjective video evaluation, digital performance verification, temperature tests, and examination of waveforms by oscilloscope.

A. SUBJECTIVE EVALUATION

The first test was a subjective evaluation of video through the system, viewed on a TV monitor. A precedent for the quality of video to be expected lay in previous experience with similar systems. The video was deemed acceptable by NOSC viewers.

B. DIGITAL PERFORMANCE VERIFICATION

A computer simulation of system performance was run and printed. The computer printout included a complete listing of the digital words at several points in the system, with a certain test function applied to the input of the forward DCT. With the aid of a logic state analyzer, the system performance was examined and found to duplicate exactly the computer simulation. Thus was the digital performance verified.

One observation regarding the digital test procedure should be made. The output of the forward and inverse DCT circuits is determinate for any 32-word input; the output of the forward and inverse DPCM circuits, however, is conditioned by both present and preceding inputs. In order to make the DPCM outputs predictable, the DPCM loops must be reset to zero. When the reset is removed, the loops build up toward their steady-state levels (with the test function applied to the forward DCT). During this recovery time the DPCM loop performance is predictable, and it is these values which have been determined by the computer simulation. The system operates for eight horizontal line intervals, after which there is another reset.

C. TEMPERATURE TESTS OF AIRBORNE BOARDS

The temperature tests of the airborne boards consisted of processing video and observing the resultant video from the system while the boards were heated and cooled. The boards were placed in a temperature controlled chamber and connected to the ground station by an adapter cable. The tests were conducted sufficiently slowly that the boards attained thermal equilibruim at the various test temperatures. Air

velocity aeross the boards in the chamber was 7.5 feet per seeond. This veloeity would provide heat transfer charaeteristics at 42.2°C equivalent to the design specifieation of 65°C at a veloeity of 18 ft/s. Tests, however, were conducted up to the 65°C temperature.

The performances of the two boards were similar but not identical. Board #1 produced, at 45°C, a picture equivalent to that at 20°C. At 50°C basis vectors began to appear in the output, manifested as fine vertical lines. At 55°C they were worse than at 50°C, but a further rise of the temperature to 65°C caused no additional degradation. On the low temperature side, at -20°C, the partitioning of the picture into 8 stripes was somewhat more noticeable than at +20°C. At -23°C, the picture disappeared entirely.

Board #2 produced, at 45°C, a picture equivalent to that at 20°C. At 50°C narrow dark vertical lines appeared in each of the 8 stripes. Raising the temperature to 65°C produced no further degradation. On the low side, at -23°C the picture was as good as that at +20°C. At -24°C noise and high frequency basis vectors appeared in the picture, rendering it useless.

D. WAVEFORMS

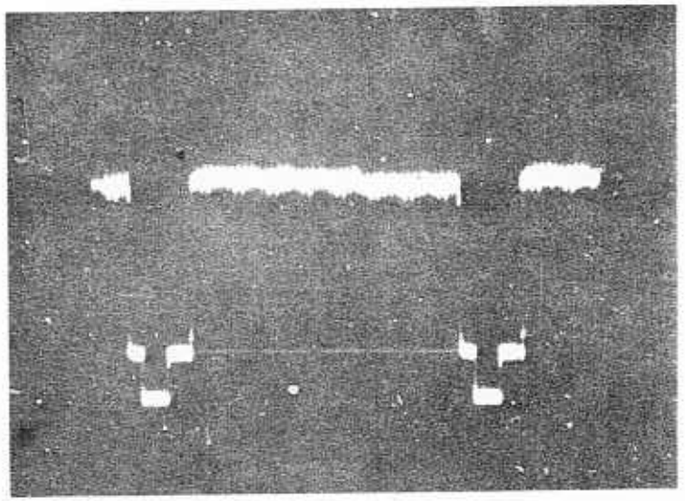
The third test, examination of waveforms, produced the oscilloscope traces in Fig. 4-1 through 4-17.

Figures 4-1 through 4-15 show the video output of the system for five kinds of input, with three sets of operating conditions, for each of the four data rates. The five inputs are dc, positive pulse, negative pulse, ramp, and sinusoid. The three sets of operating conditions are with neither vertical reset nor guard band, with vertieal reset but no guard band, and with both vertical reset and guard band.

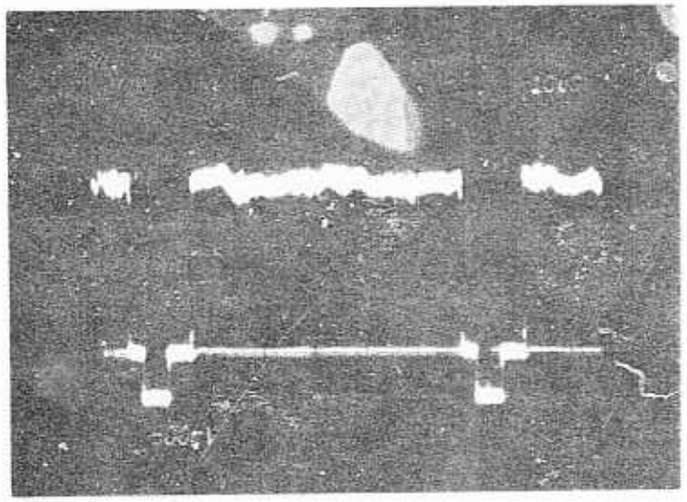
The pictures show slightly more than one horizontal line interval. The faint trace at the blanking level is attributable to the vertical retraee blanking. The dc introduced was slightly less than the positive saturation level. Negative saturation was slightly above the blanking level, as may be seen in the pictures of the ramps and sinusoids.

Figure 4-16 shows the linearity of the analog input circuitry (the input amplifier, the TAD analog delay device, sample-and-hold, and A/D converter). The output waveform was obtained from a D/A eonverter whose input was the 6-bit output of the A/D converter from the airborne board. Figure 4-16d shows the output waveform and also the output from the TAD analog delay over 32 samples, with a ramp input over the operating range of the circuits.

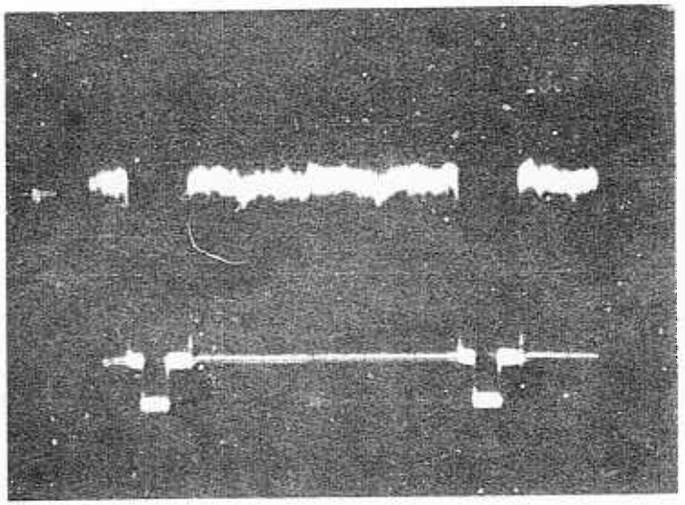
Figure 4-17 shows the video output with a ramp signal from the D/A converter. The ramp was generated in the frame store memory; it had not gone through the encoding-decoding process. Its purpose is to show the linearity and signal-to-noise ratio of the output circuits.



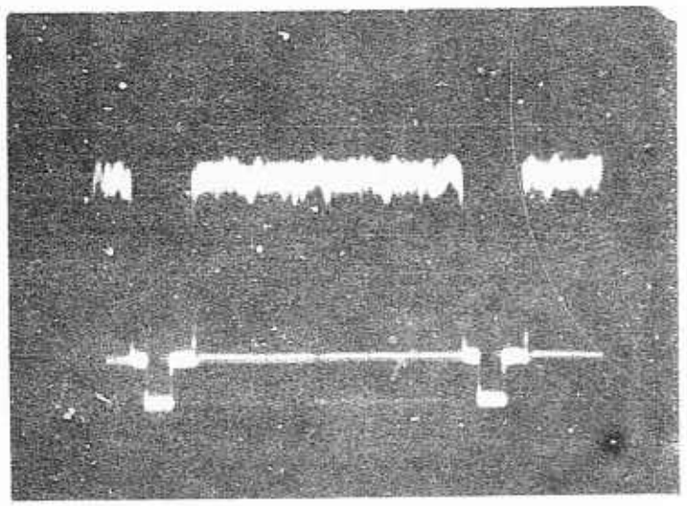
1600 kb/s



800 kb/s



400 kb/s



200 kb/s

Fig. 4-1. FSM output, dc, no reset, no guard band.

48

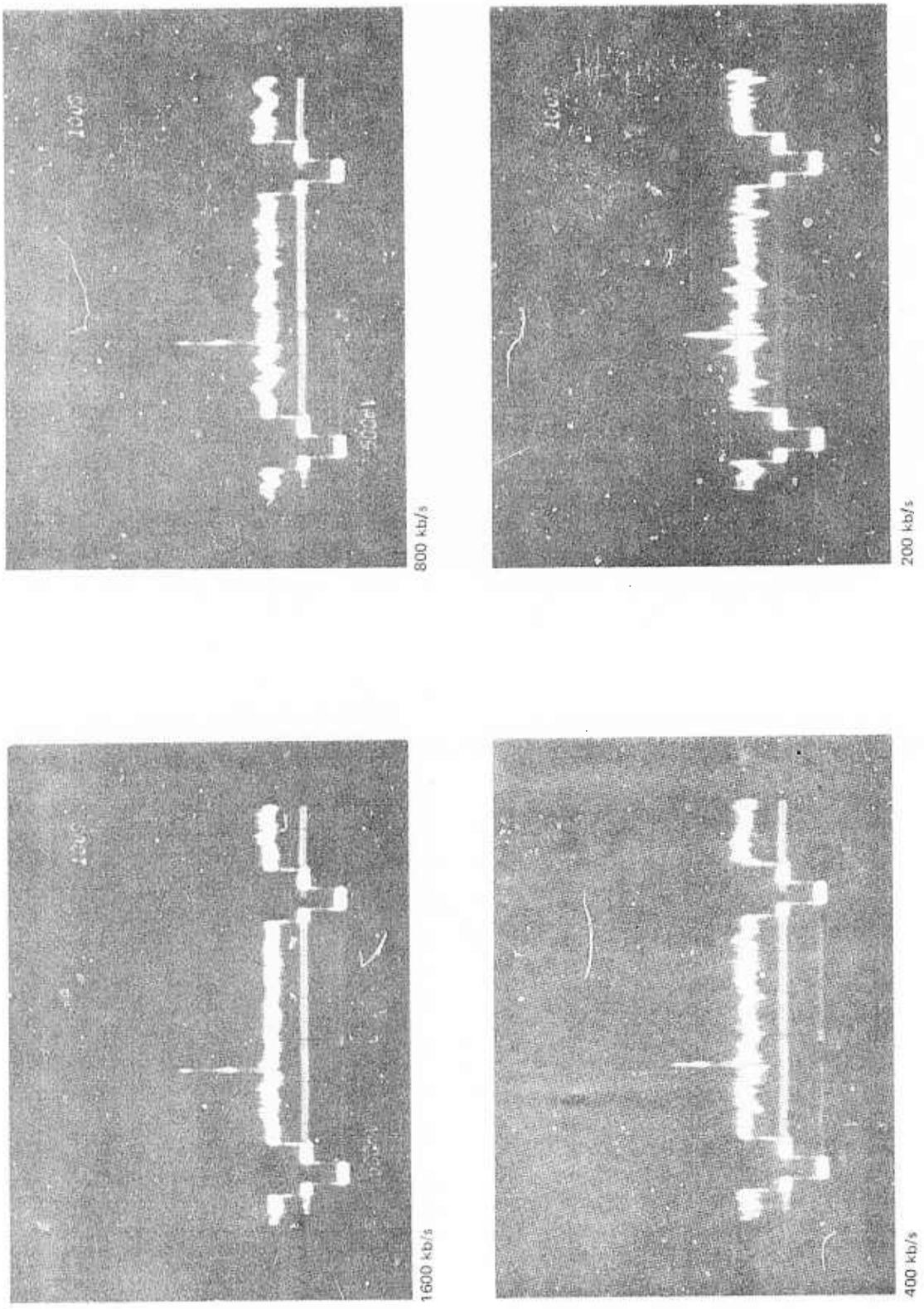
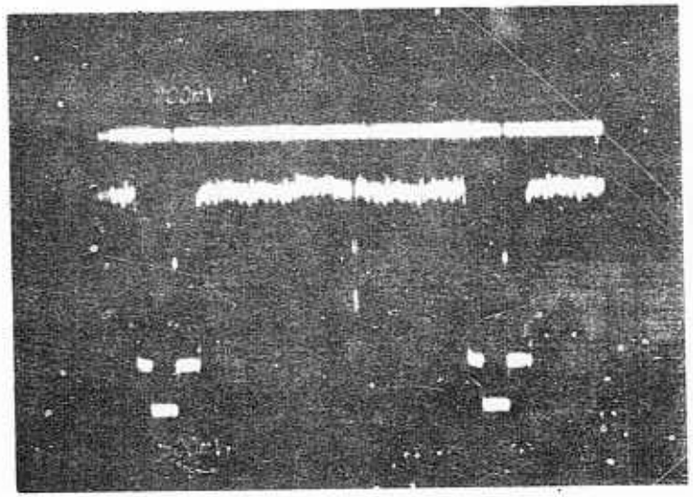
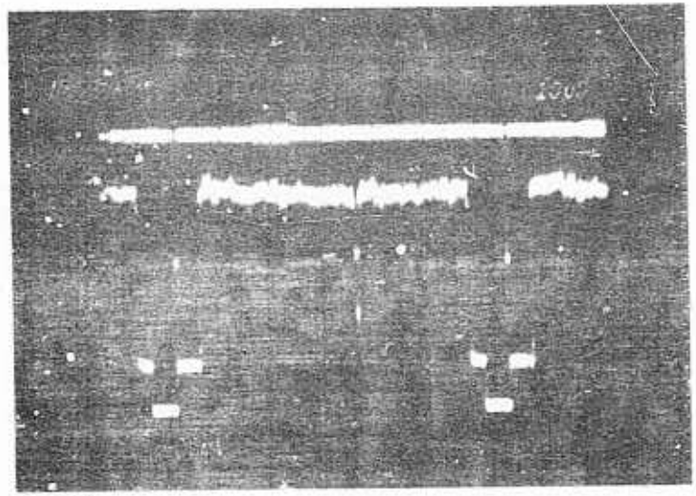


Fig. 4-2. FSM output, positive pulse, no reset, no guard band.

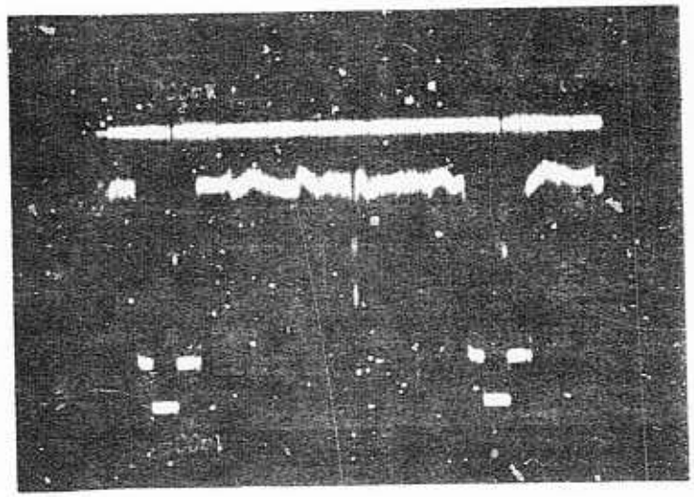
49



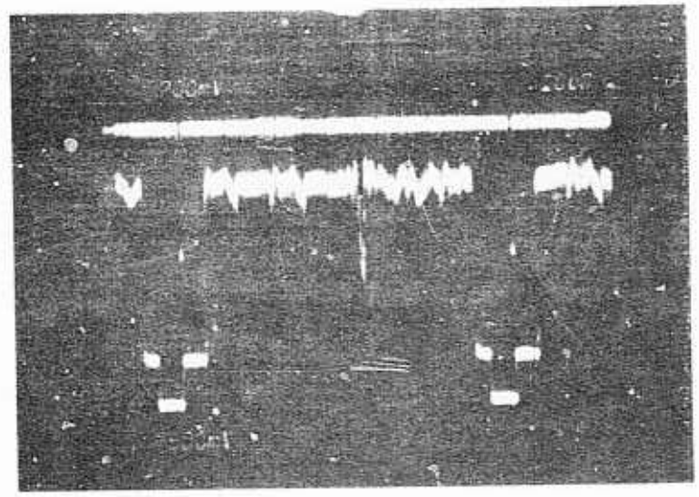
1600 kb/s



800 kb/s

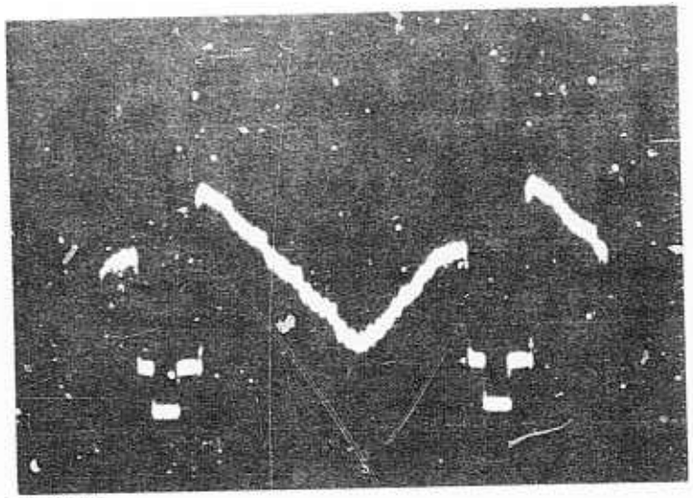


400 kb/s

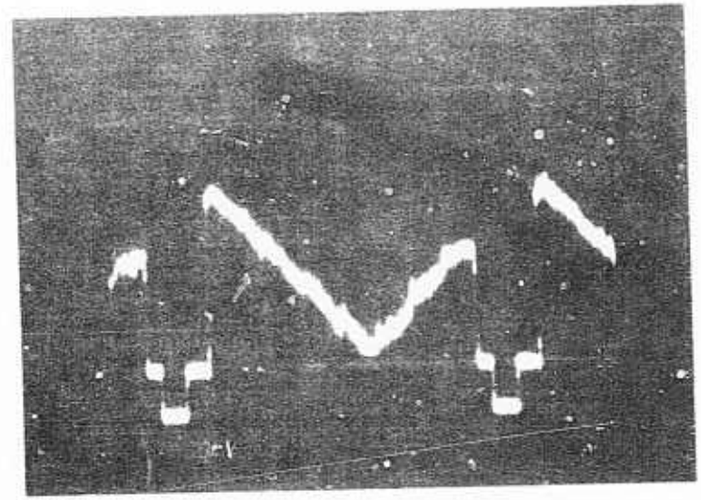


200 kb/s

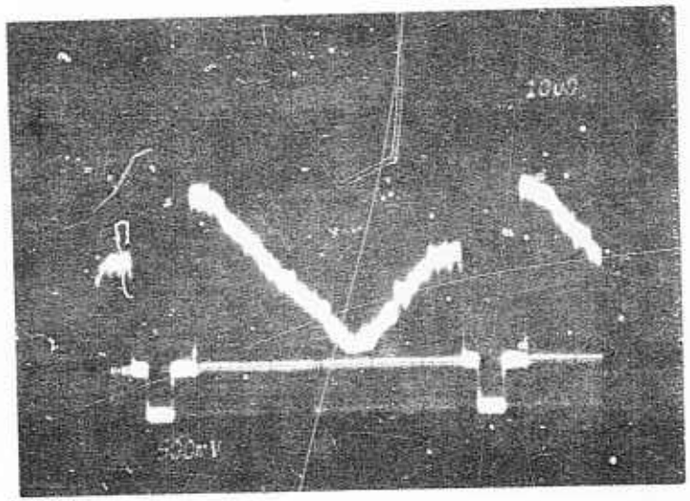
Fig. 4-3. FSM output, negative pulse, no reset, no guard band.



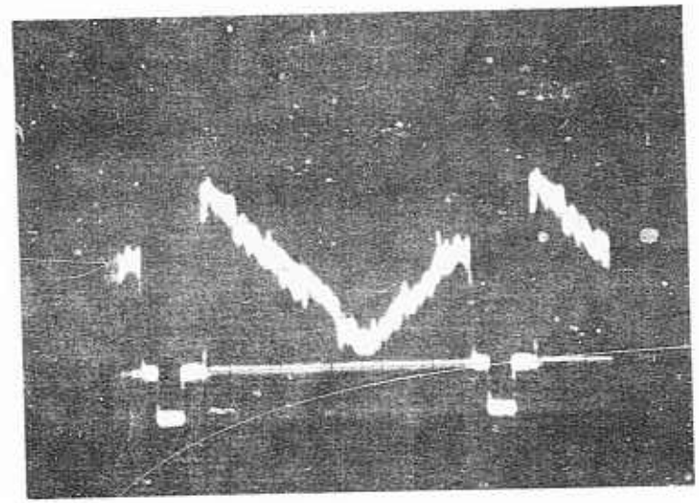
1600 kb/s



800 kb/s



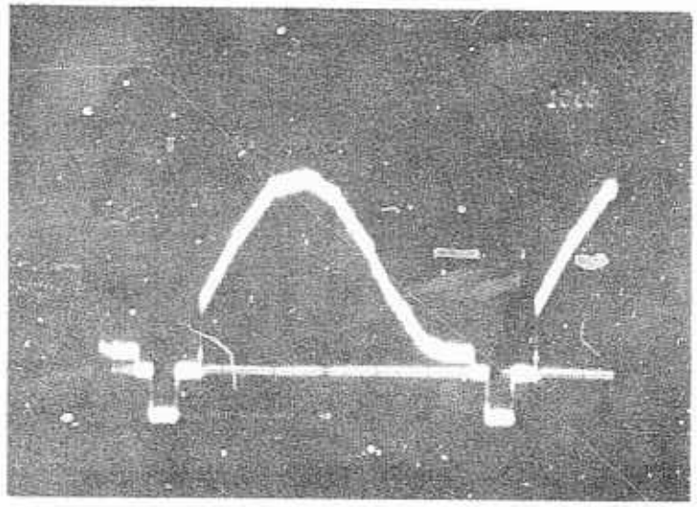
400 kb/s



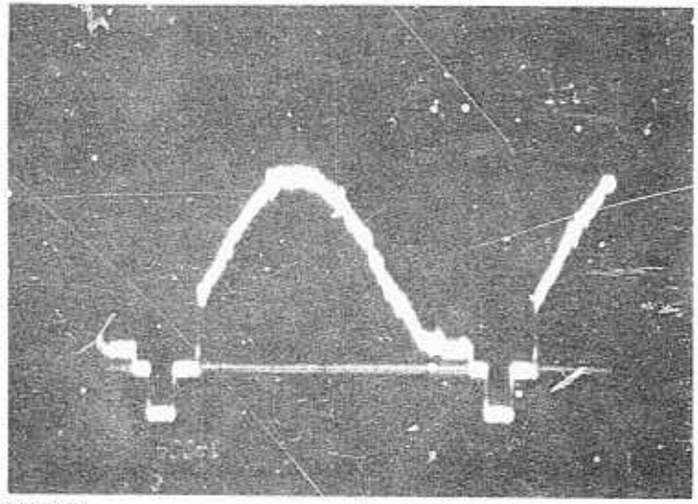
200 kb/s

Fig. 4-4. FSM output, ramp, no reset, no guard band.

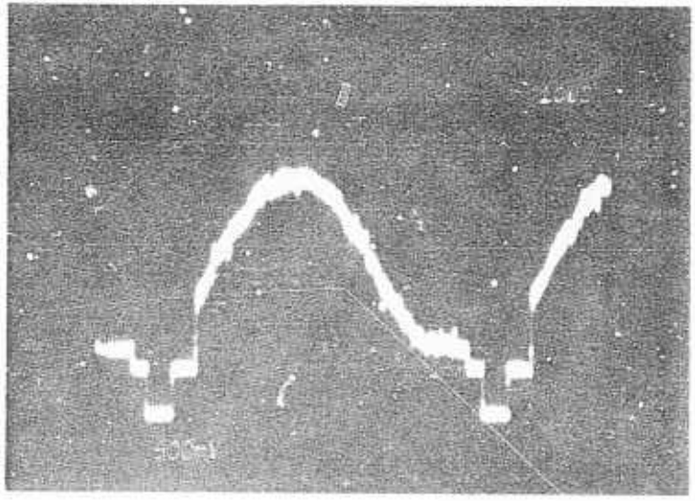
51



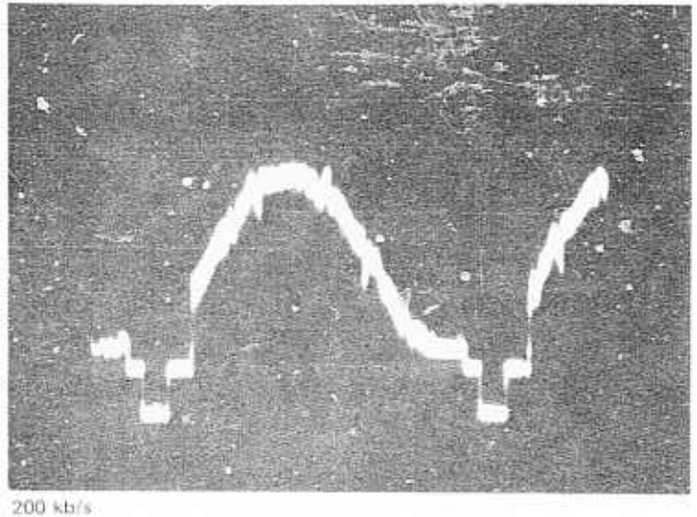
1600 kb/s



800 kb/s



400 kb/s



200 kb/s

Fig. 4-5. FSM output, sinusoid, no reset, no guard band.

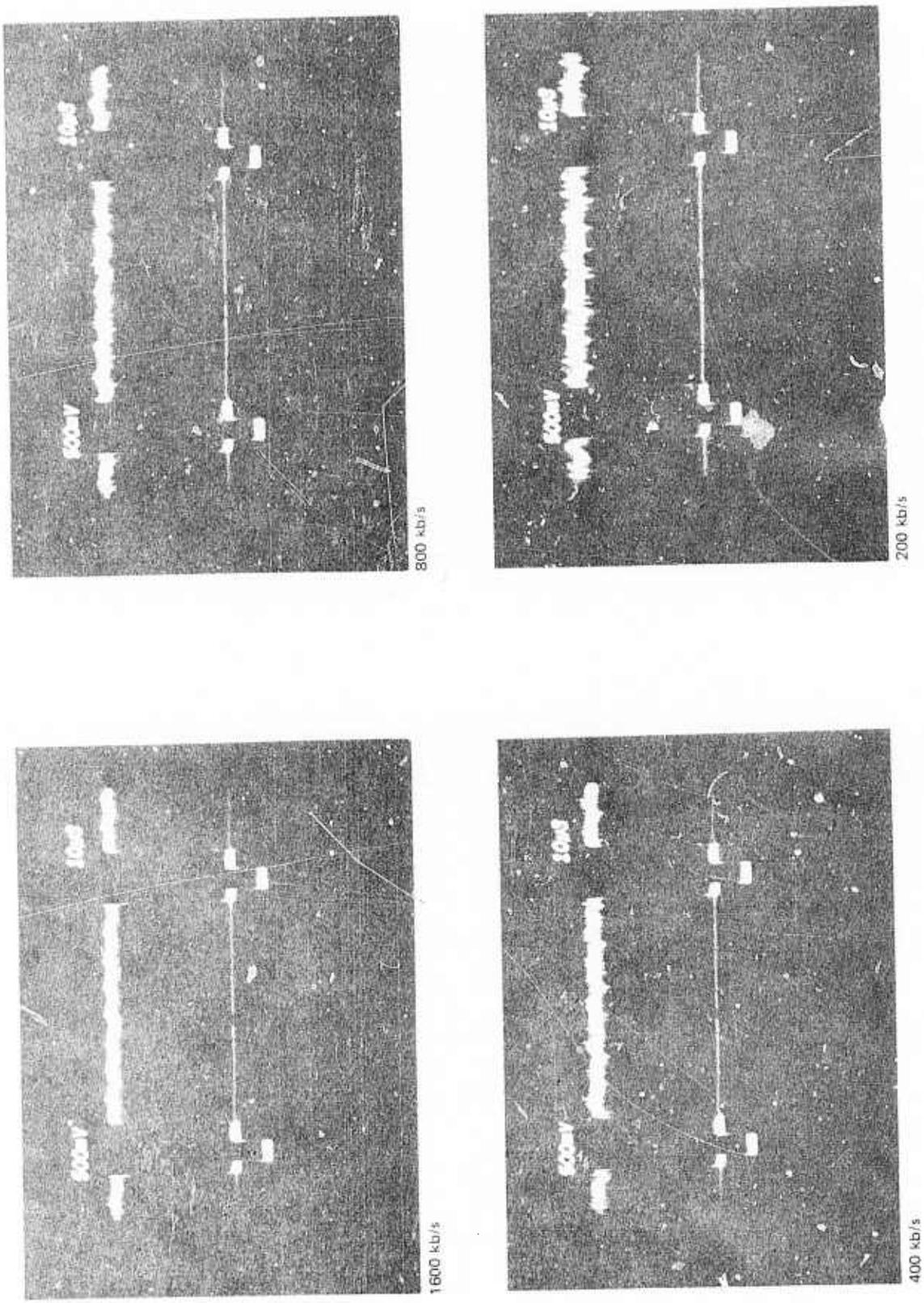


Fig. 4-6. FSM output, dc, reset in, guard band out.

53

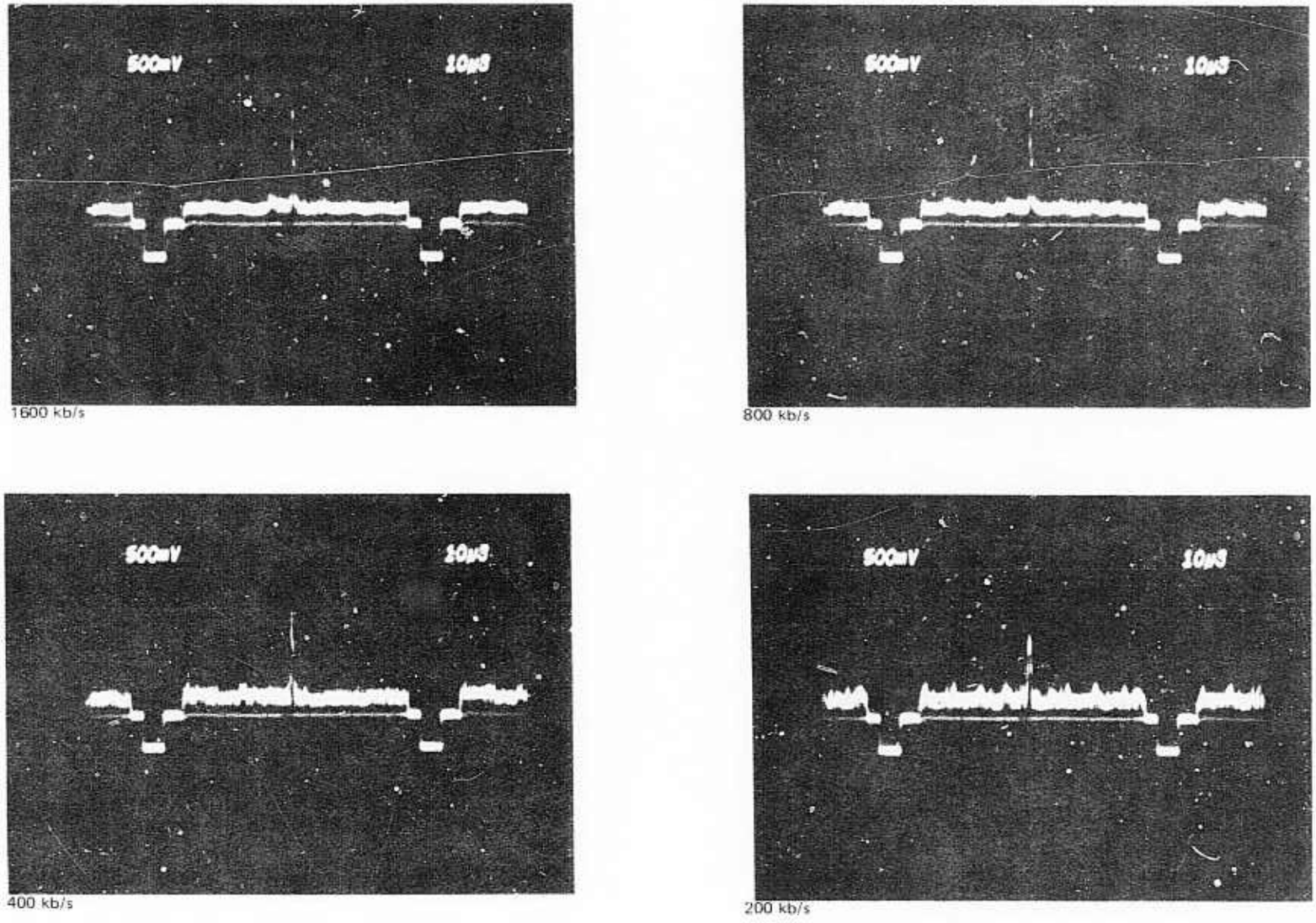
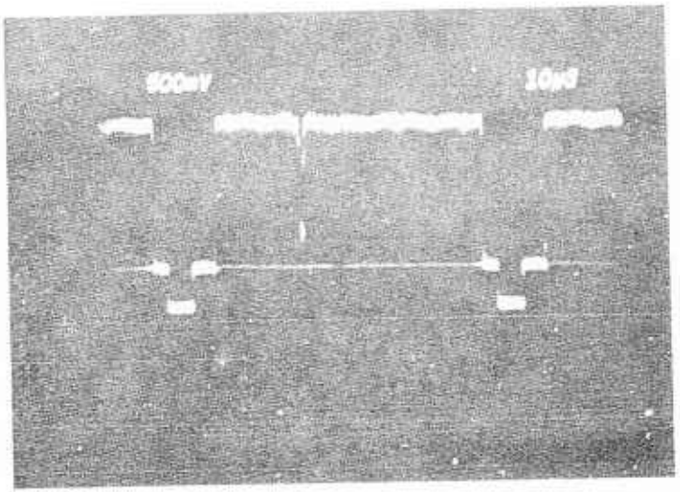
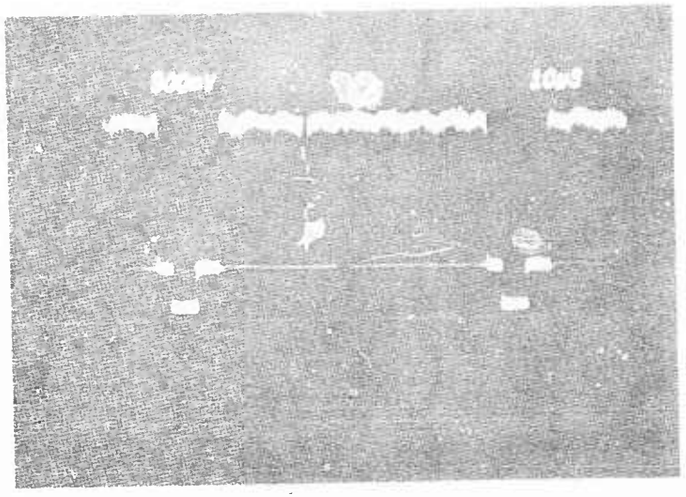


Fig. 4-7. FSM output, positive pulse, reset in, guard band out.

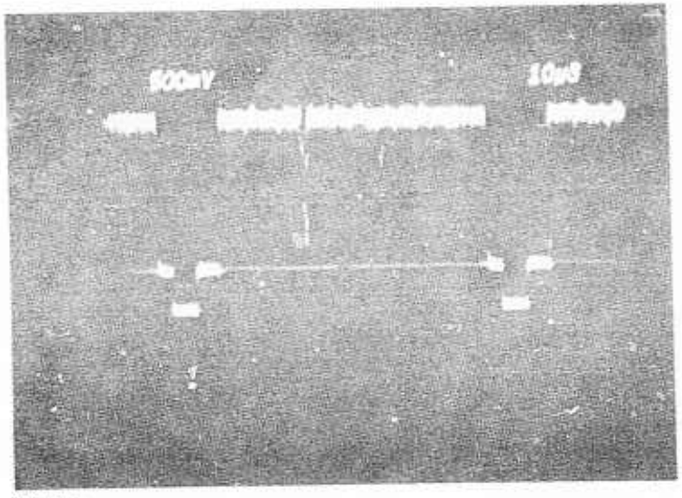
54



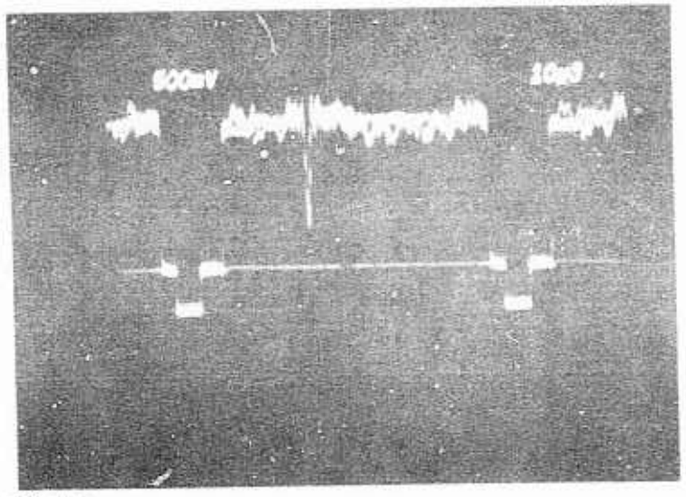
1600 kb/s



800 kb/s

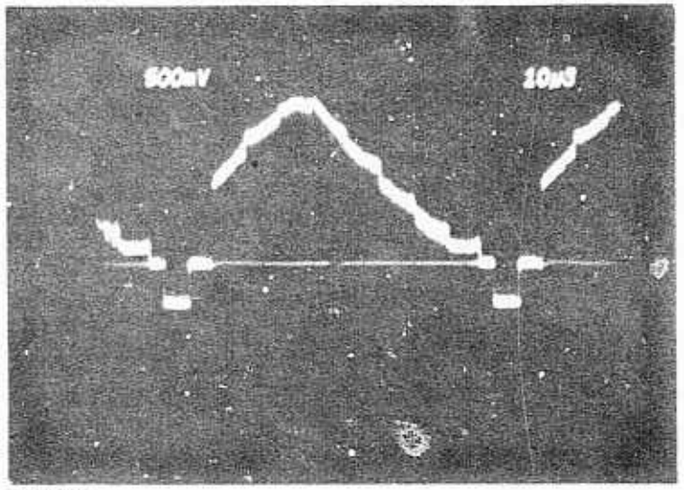


400 kb/s

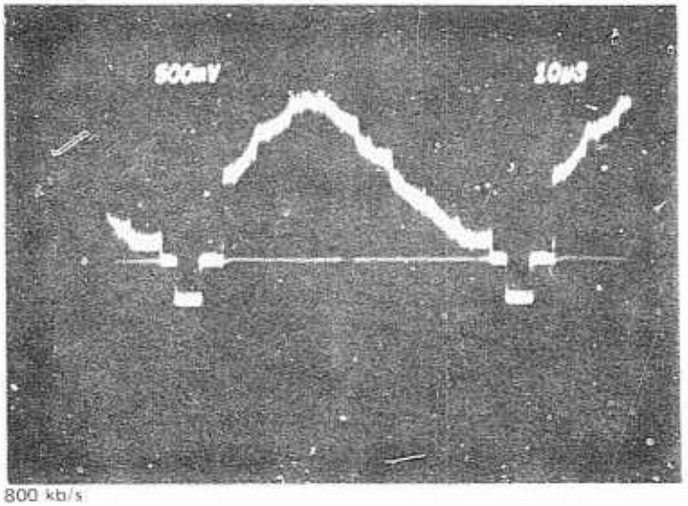


200 kb/s

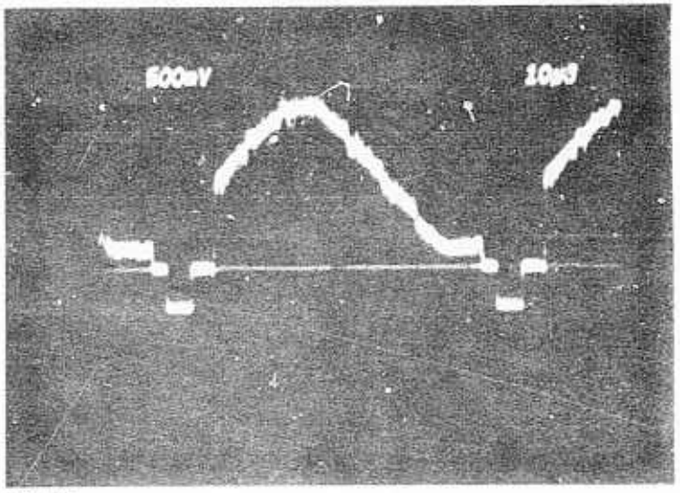
Fig. 4-8. FSM output, negative pulse, reset in, guard band out.



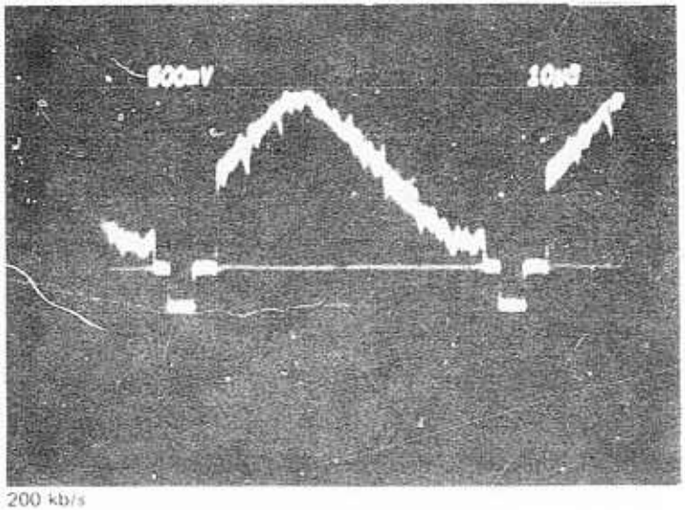
1600 kb/s



800 kb/s



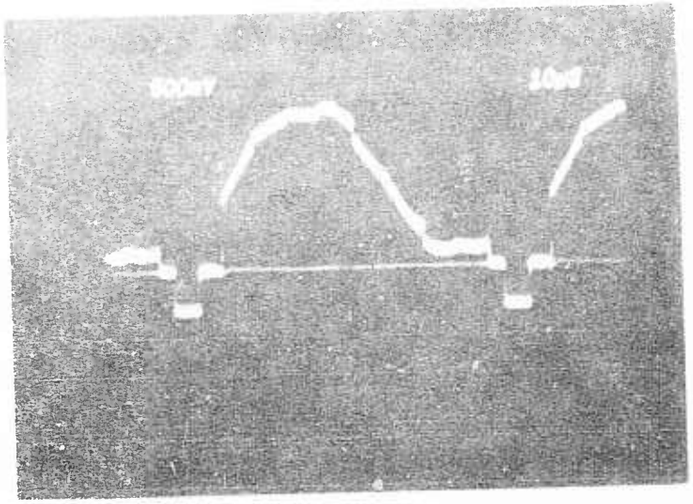
400 kb/s



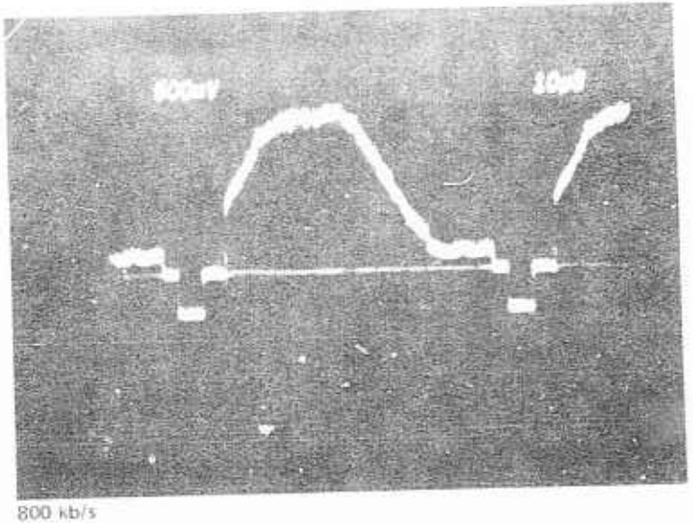
200 kb/s

Fig. 4-9. FSM output, ramp, reset in, guard band out.

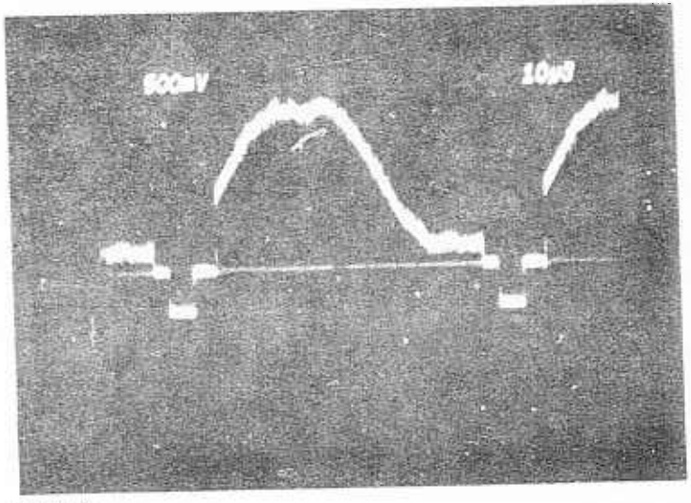
96



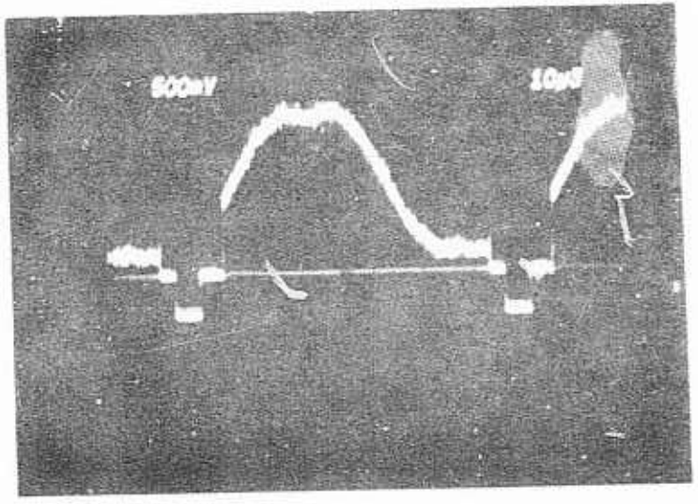
1600 kb/s



800 kb/s



400 kb/s



200 kb/s

Fig. 4-10. FSM output, sinusoid, reset in, guard band out.

67

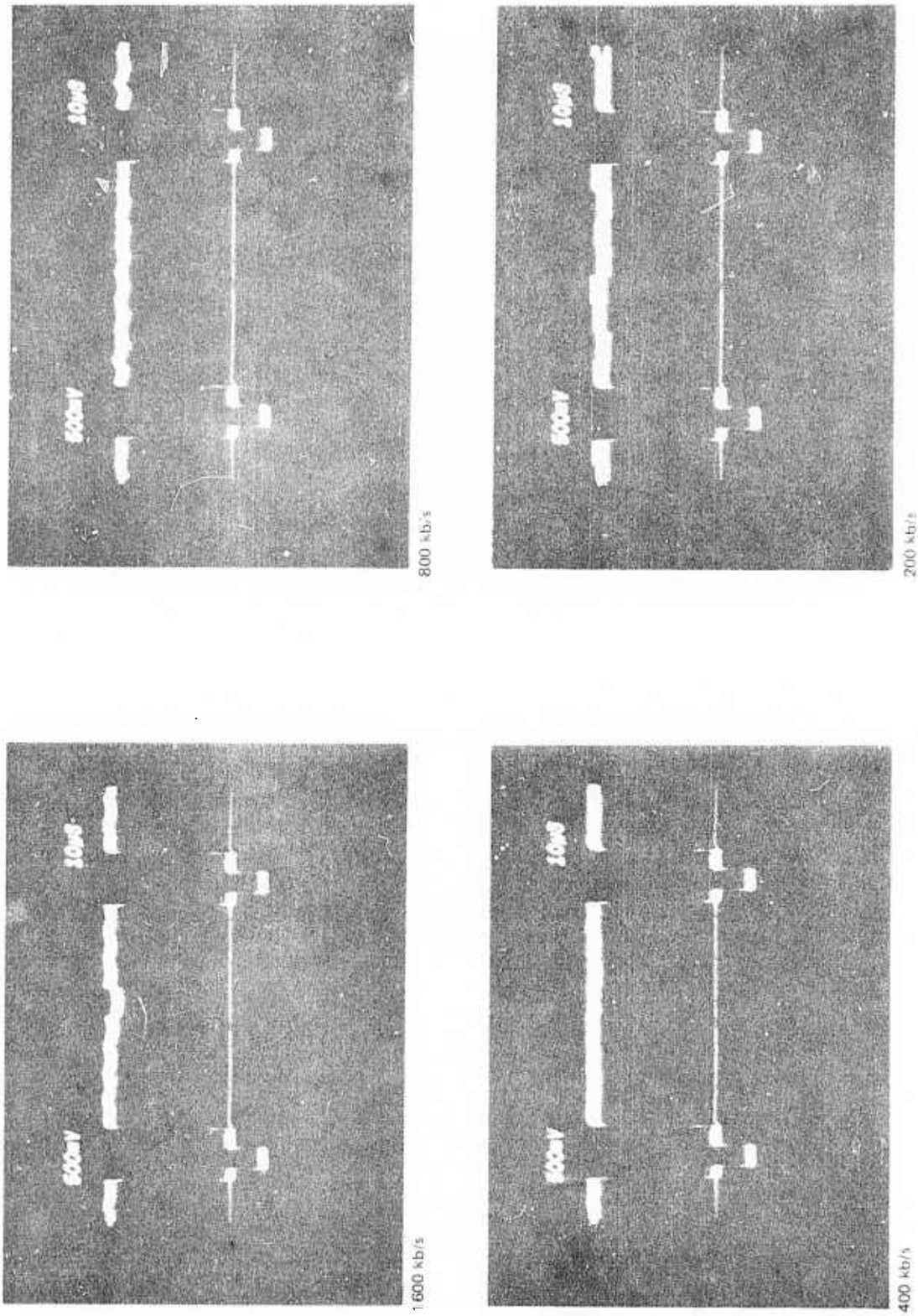
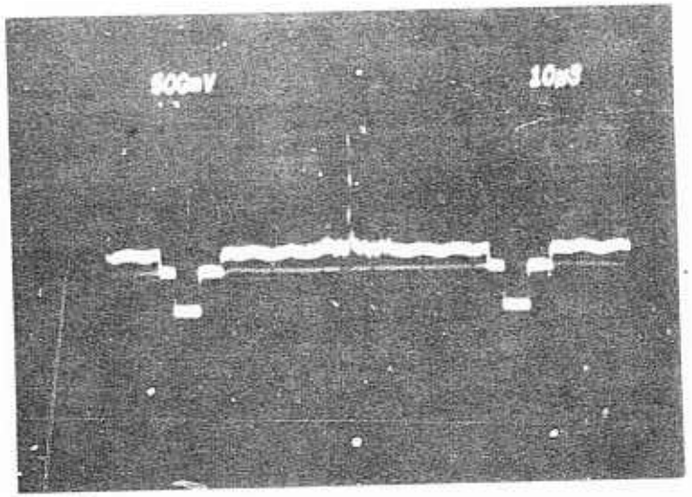
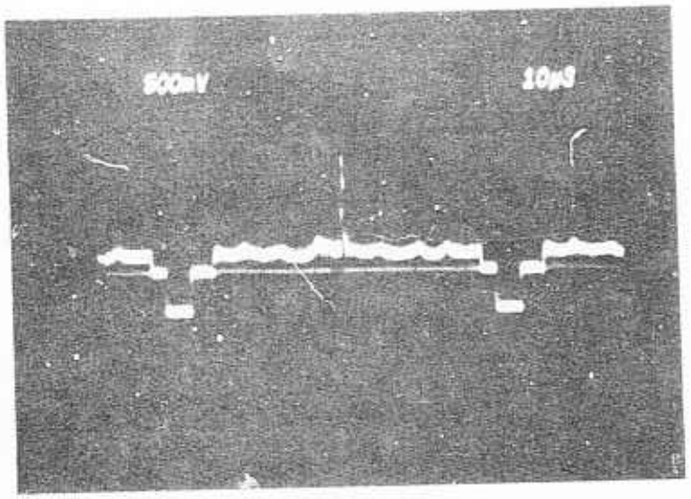


Fig. 4-11. FSM output, dc, reset in, guard band in.

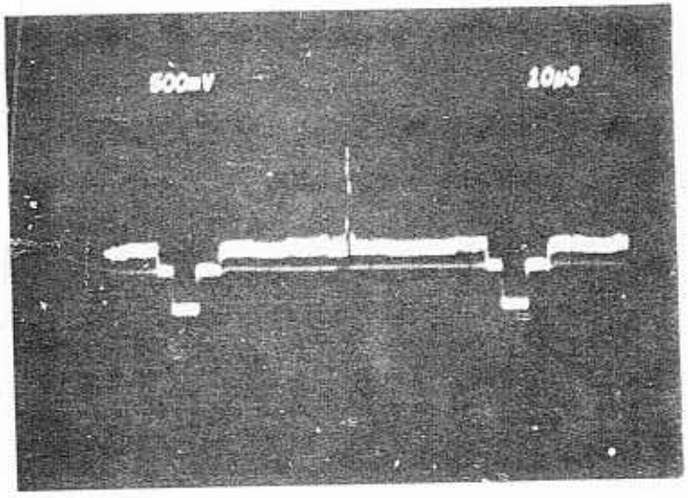
58



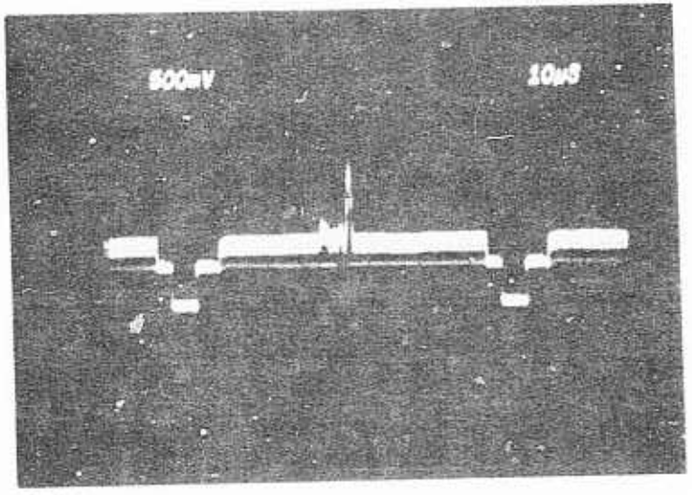
1600 kb/s



800 kb/s



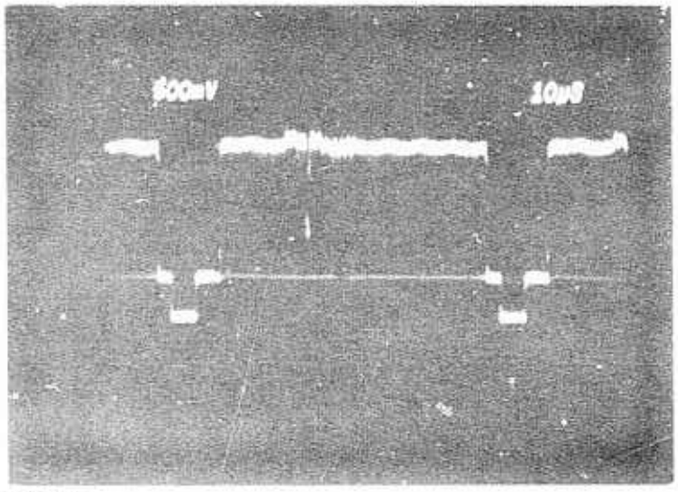
400 kb/s



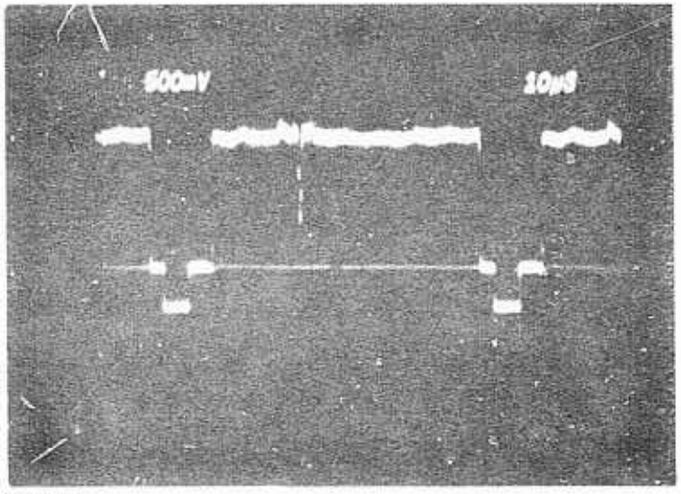
200 kb/s

Fig. 4-12. FSM output, positive pulse, reset in, guard band in.

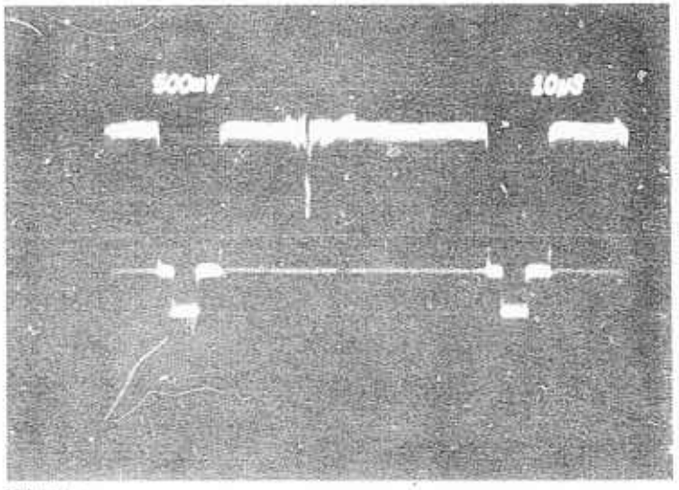
59



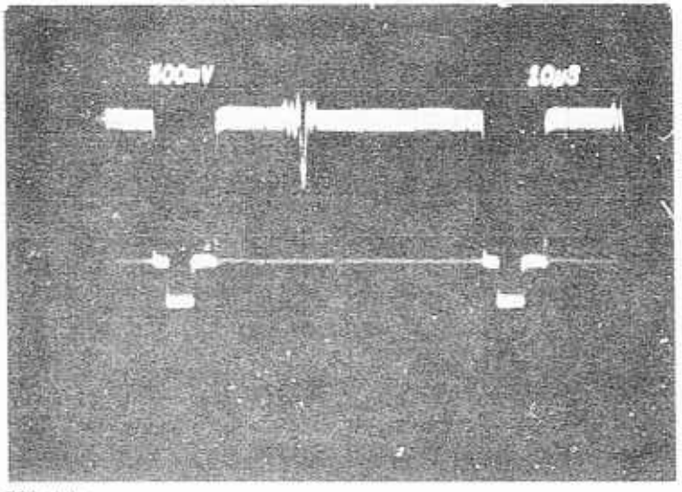
1600 kb/s



800 kb/s



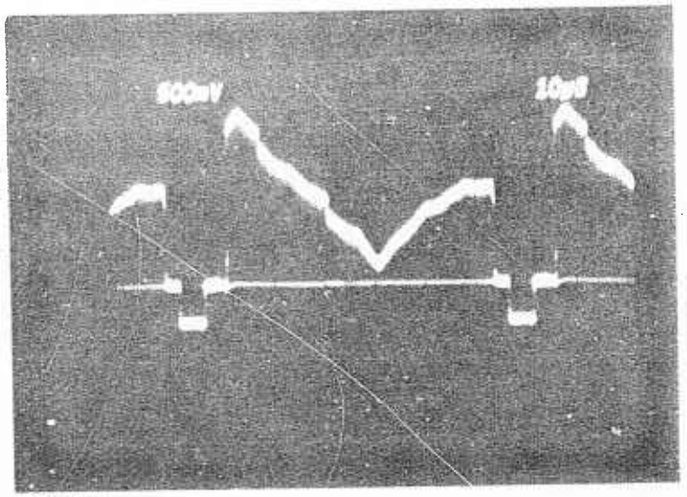
400 kb/s



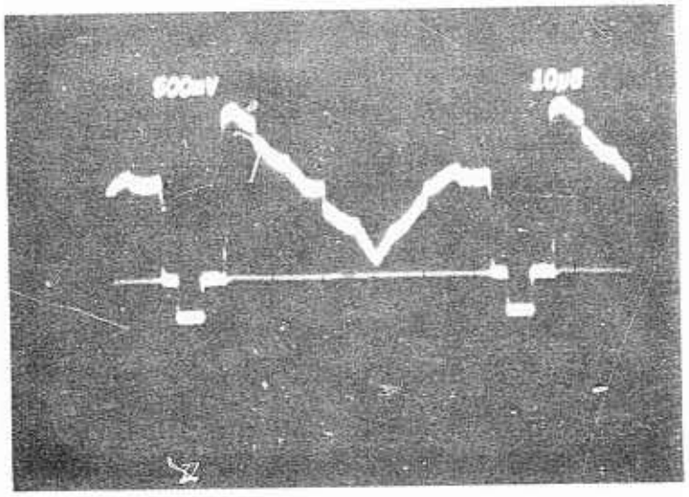
200 kb/s

Fig. 4-13. FSM output, negative pulse, reset in, guard band in.

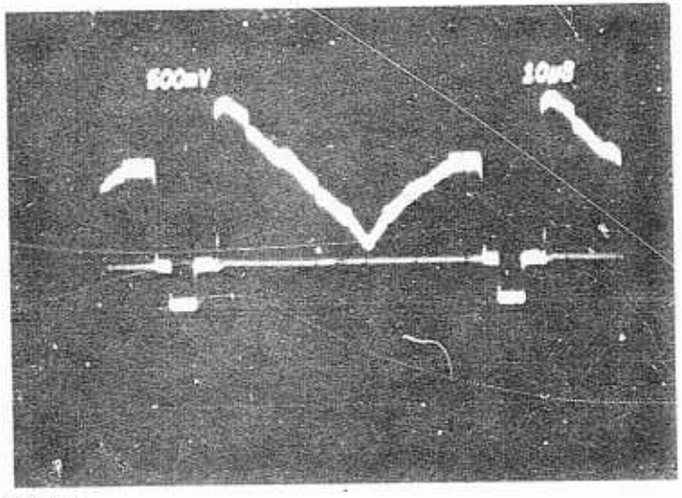
60



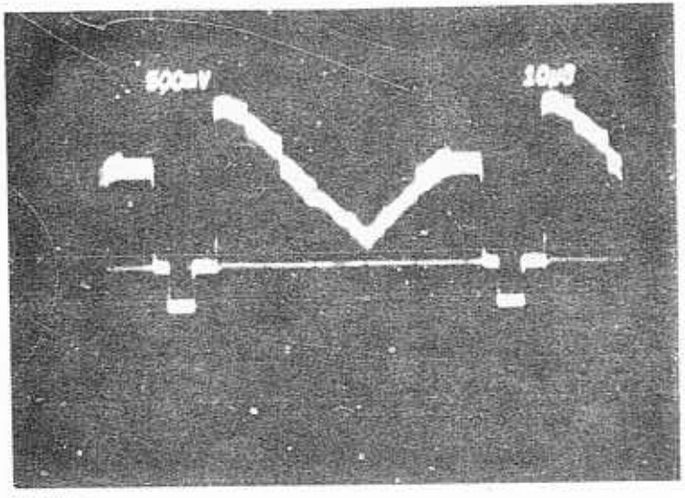
1600 kb/s



800 kb/s

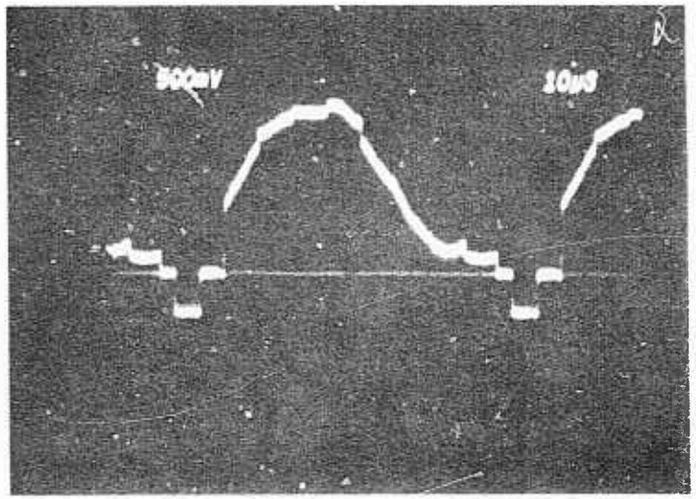


400 kb/s

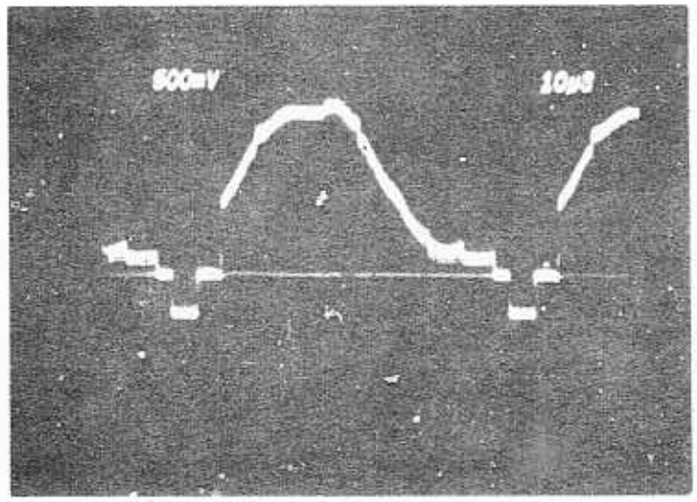


200 kb/s

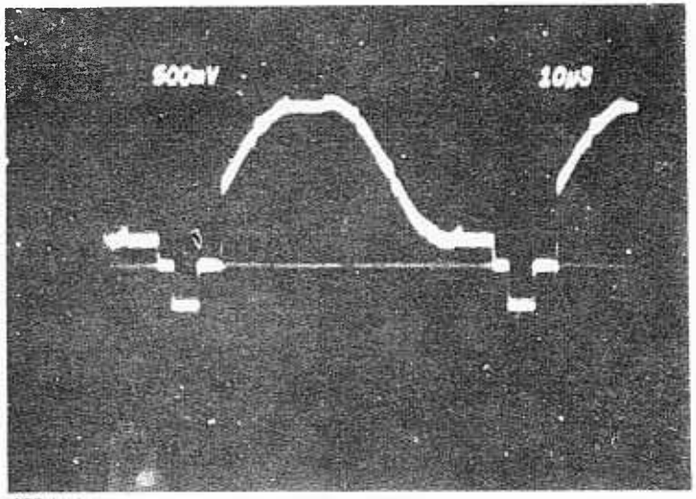
Fig. 4-14. FSM output, ramp, reset in, guard band in.



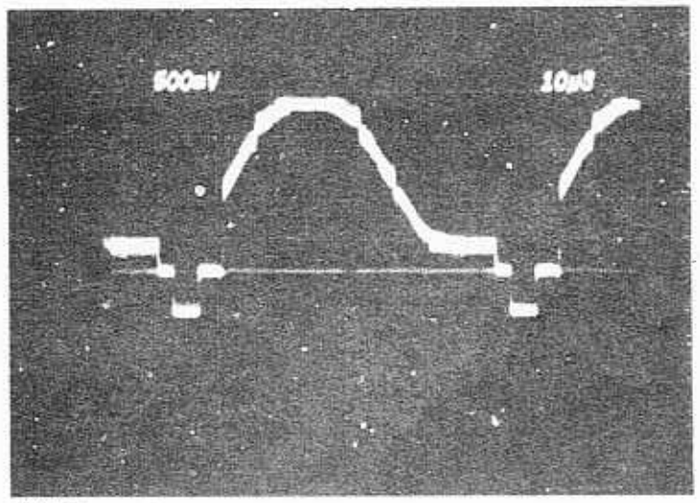
1600 kb/s



800 kb/s

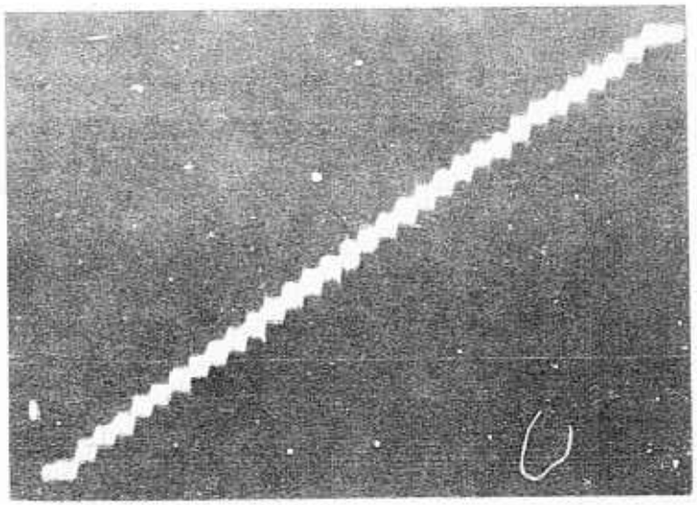


400 kb/s

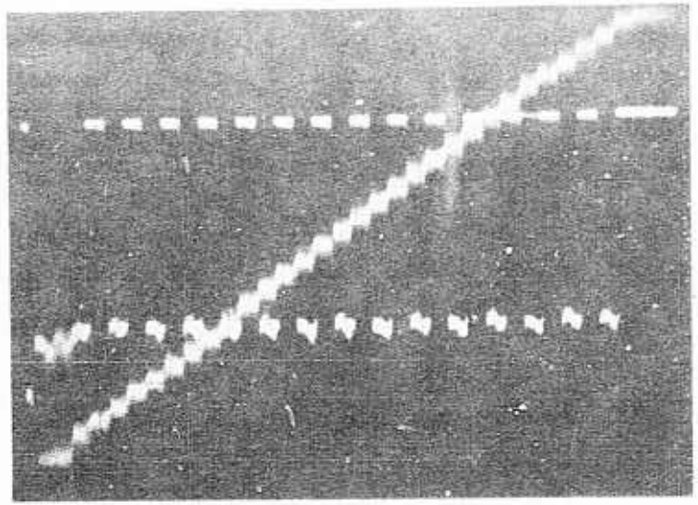


200 kb/s

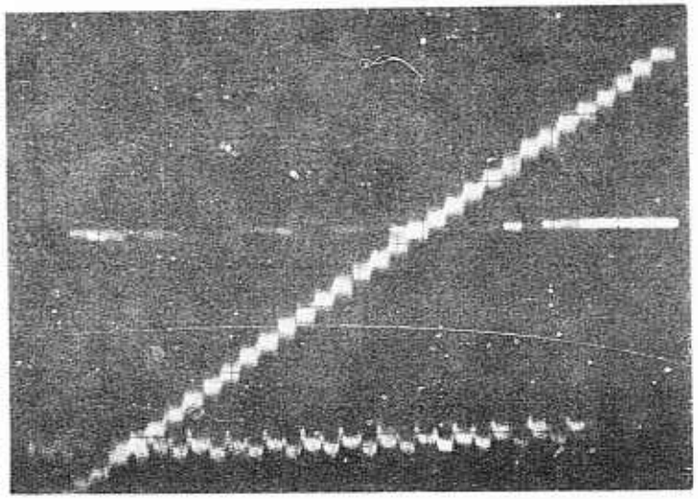
Fig. 4-15. FSM output, sinusoid, reset in, guard band in.



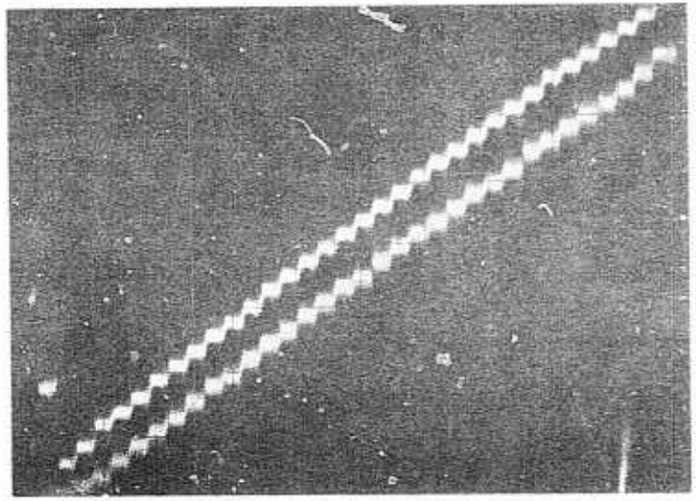
AIRBORNE A/D THROUGH D/A
FULL SCALE A/D OUTPUT



A/D THROUGH D/A
AND BIT 5 OF A/D



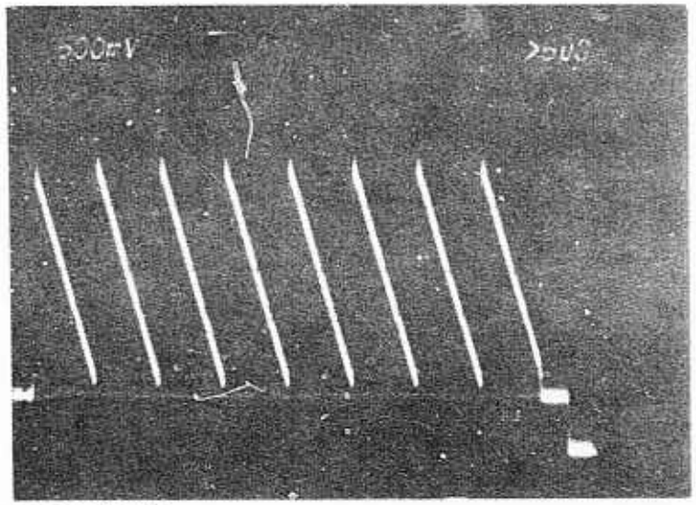
A/D THROUGH D/A
AND LSB OF A/D



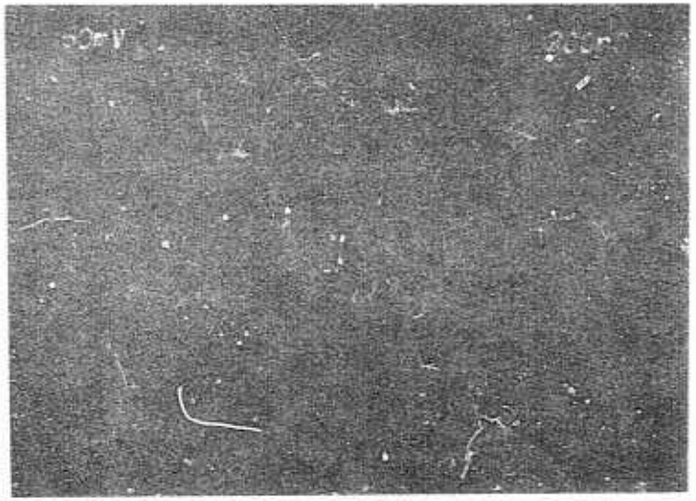
A/D THROUGH D/A (TOP)
TAD OUTPUT (BOTTOM)

Fig. 4-16. Linearity of analog input circuitry.

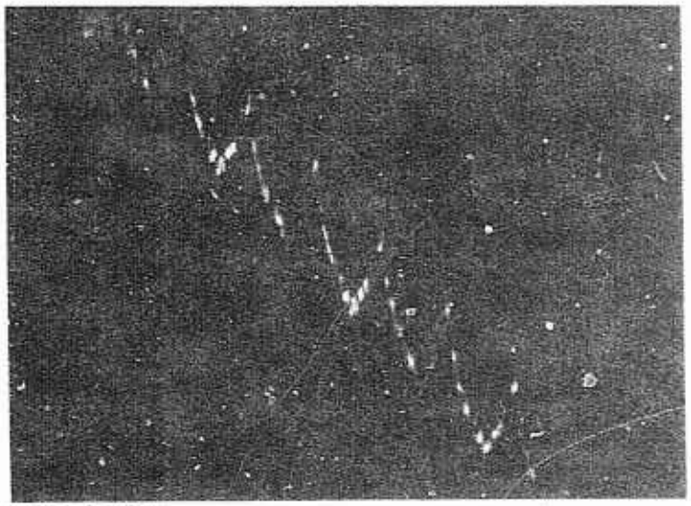
63



VIDEO OUTPUT:
RAMP TEST SIGNAL



VIDEO OUTPUT:
RAMP TEST SIGNAL EXPANDED



VIDEO OUTPUT:
RAMP TEST SIGNAL EXPANDED

Fig. 4-17. Video output: Ramp Test Signal

64

Section V

CMOS/SOS ARITHMETIC BUILDING BLOCKS

Two key factors enabled RCA to meet the system specifications of size, weight, and power for the airborne assembly. The first was the factorization of the direct cosine transform (DCT) into the general form of a butterfly (Fig. 5-1) that can be implemented by an arithmetic processor. The second factor was the availability of a set of CMOS/SOS arithmetic building blocks developed on a separate Air Force contract.* The specific CMOS/SOS building blocks (the 9 x 9 multiplier, 9-bit adder, and dual register retimer) are described in the following paragraphs.

A. 9 x 9 BIT SIGN MAGNITUDE MULTIPLIER

The asynchronous multiplier multiplies numbers A and B, each of which consists of 8 bits for magnitude plus a sign bit. Available at the output are all 16 bits of the product magnitude plus the output sign bit. A round-off control is available which if selected provides the output magnitude to be rounded to provide an 8-bit-plus-sign output. With R not selected, the full 16-bit product is available at the output. This option allows the multiplier to be used for both integer and fraction multiplications. The average dynamic power and average propagation delay of the longest path for a set of multiplier chips are shown in Fig. 5-2 and 5-3. An alternating all ones and all zeros input pattern was used to measure the maximum dynamic power dissipation. The average dynamic power is one-half of the maximum.

B. 9-BIT ADDER

Adder array TCS065, shown functionally in Fig. 5-4, adds numbers A and B to obtain sum C. Each of the inputs consists of 8 bits for magnitude and one bit for sign. A carry in (C_{in}) input is provided which is added into the least significant bit position. Either 1's complement or 2's complement representation for negative numbers can be handled. When used as a 1's complement adder, the end around carry is obtained from the F_{out} output. This end around carry is applied to the C_{in} input. When used as a 2's complement adder, the connection between F_{out} and C_{in} is not made. The output consists of 8 bits for magnitude, a sign bit, and an overflow indication. Whenever overflow occurs or when shifting is under external control, the output magnitude bits can be shifted one position to the right. The least significant bit is shifted out; the next least significant bit becomes the least significant bit, etc. The carry out of the last stage becomes the most significant bit. This feature is very useful in scaling the partial products at selected stages of the butterfly to eliminate overflow during calculation of the DCT. To prevent oscillations in the 1's complement arithmetic

*AFAL-TR-77-23, "Programmable FFT Linear FM Waveform Processor".

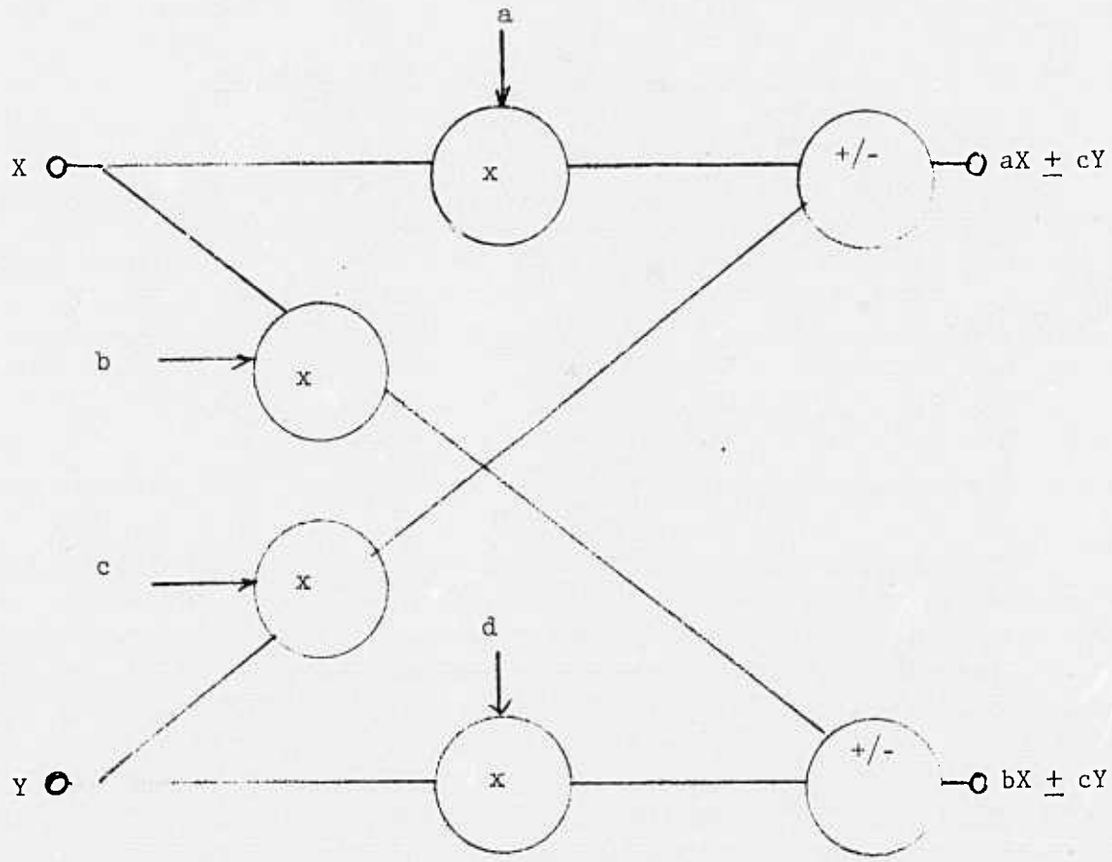


Fig. 5-1. General form of the butterfly for implementing the DCT processor.

used in the DCT, an all 1's condition with A and/or B negative is detected, which forces an end around carry giving a positive zero result. The average and maximum propagation delay for a set of adder chips is shown in Fig. 5-5.

C. DUAL REGISTER RETIMER

The dual register retimer shown functionally in Fig. 5-6 is used to change back and forth between sign magnitude numbers used in the multiplier and 1's complement numbers used in the adder. The retimer contains two independent sets of register circuits, each of which is capable of retiming 9 input bits. The retimers can be used simply to retime the input as well as to complement input signals.

When control input H is in the low state, outputs C_0 through C_7 are the same as inputs a_0 through a_7 except for the one clock period delay of the retimer. When H is in the high state, outputs C_0 through C_7 are the complements of inputs a_0 through a_7 except for the one clock period delay. For the 9th input bit, a_8, both a complemented and an uncomplemented output are provided.

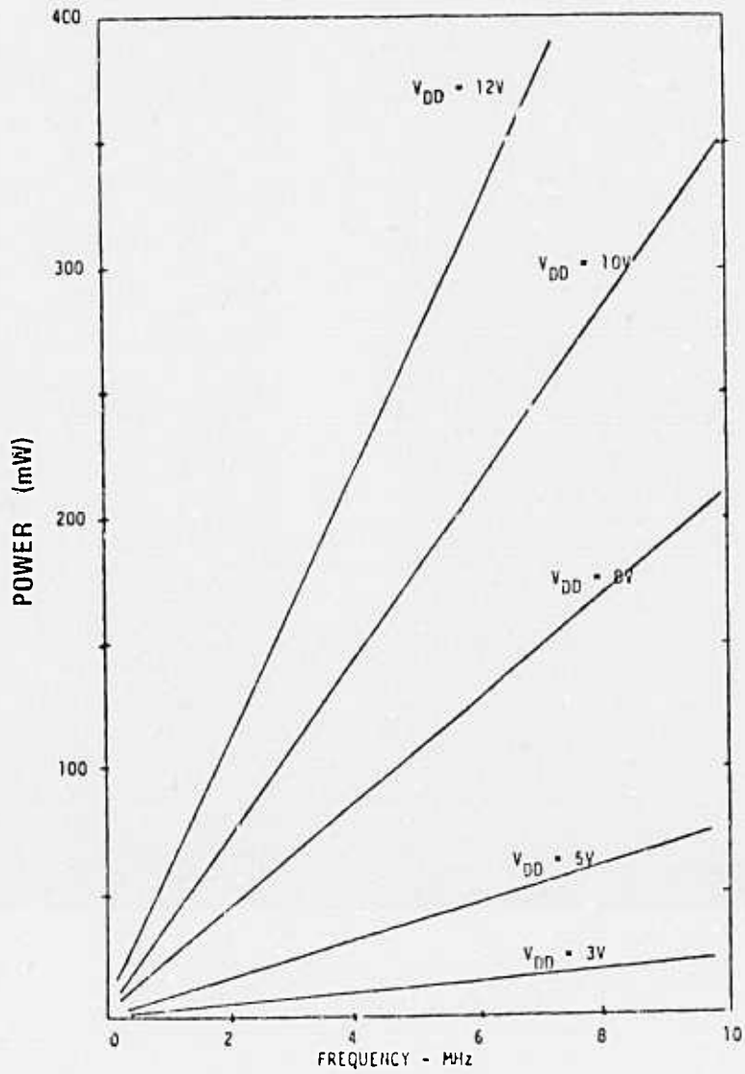


Fig. 5-2. Average of maximum dynamic power for TCS-057 multipliers.

In the DCT arithmetic unit the a_S input (sign bit) is used as the H input for the number being retimed in conversion from sign magnitude to 1's complement.

67

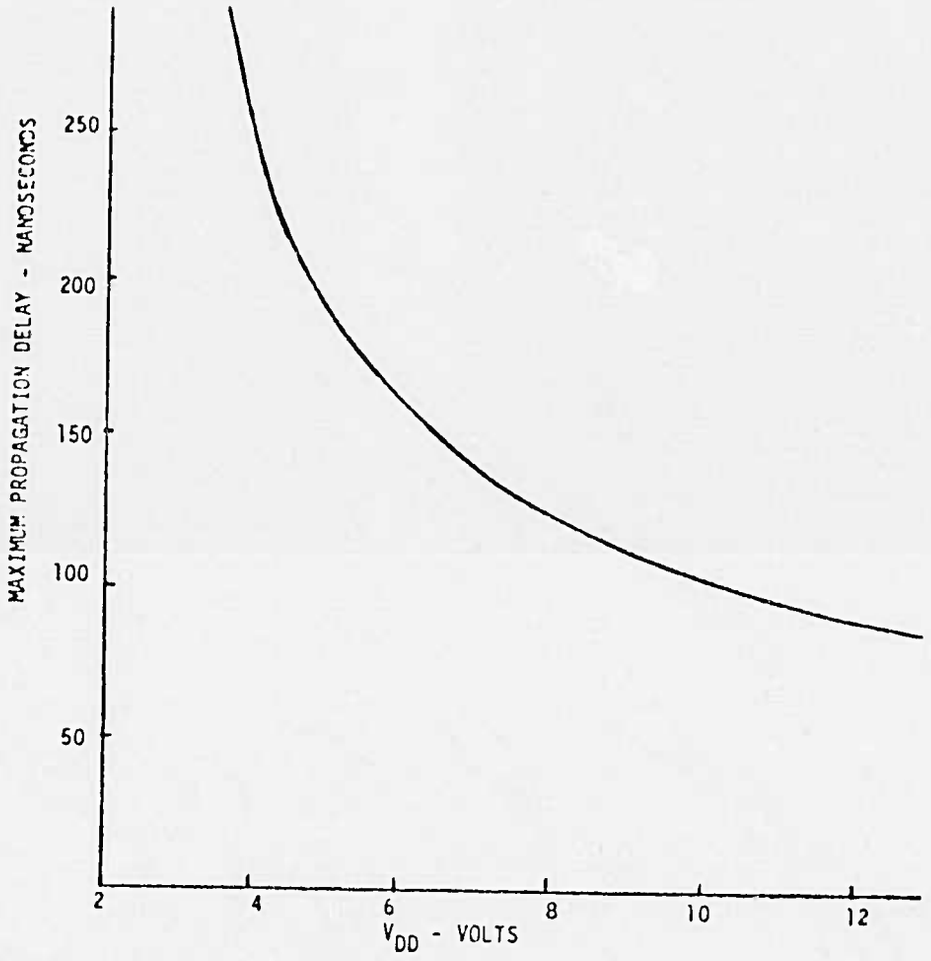


Fig. 5-3. Average propagation delay of longest path for TCS-057 multipliers.

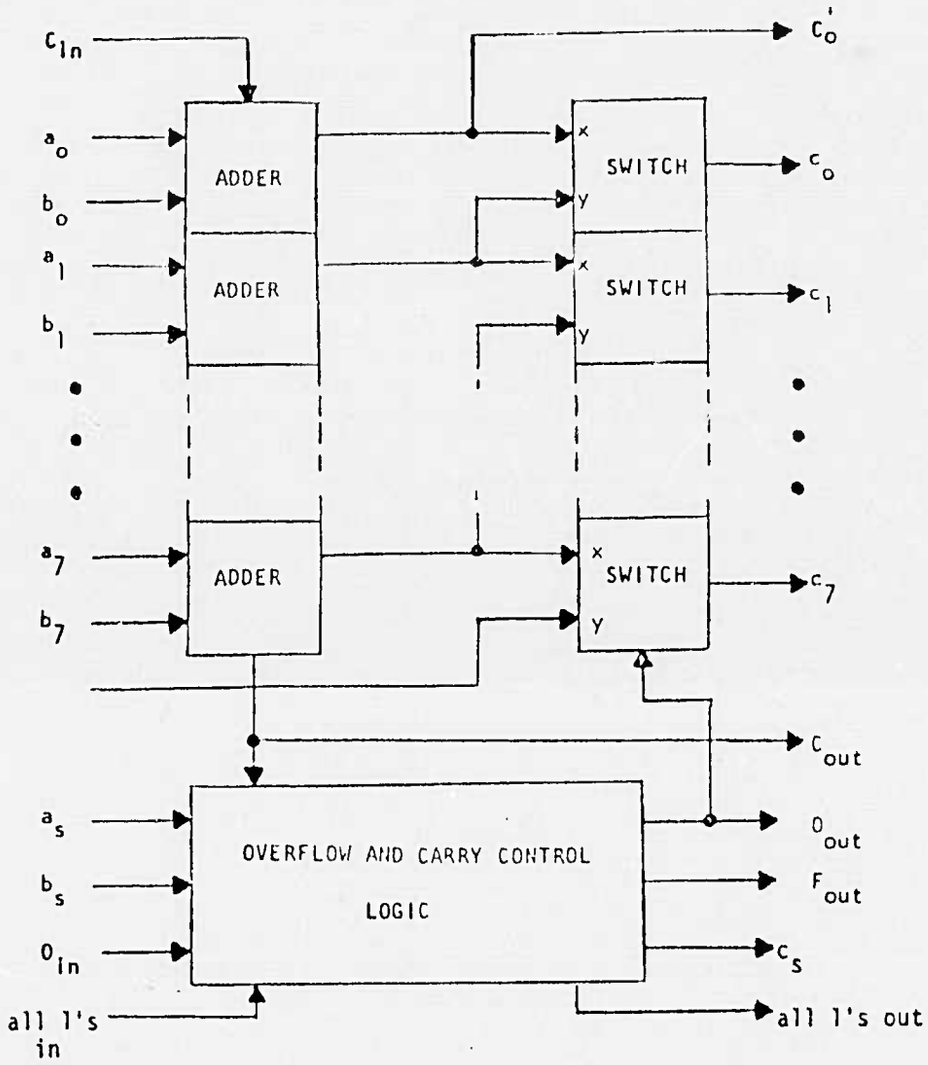


Fig. 5-4. 9-bit adder array.

69

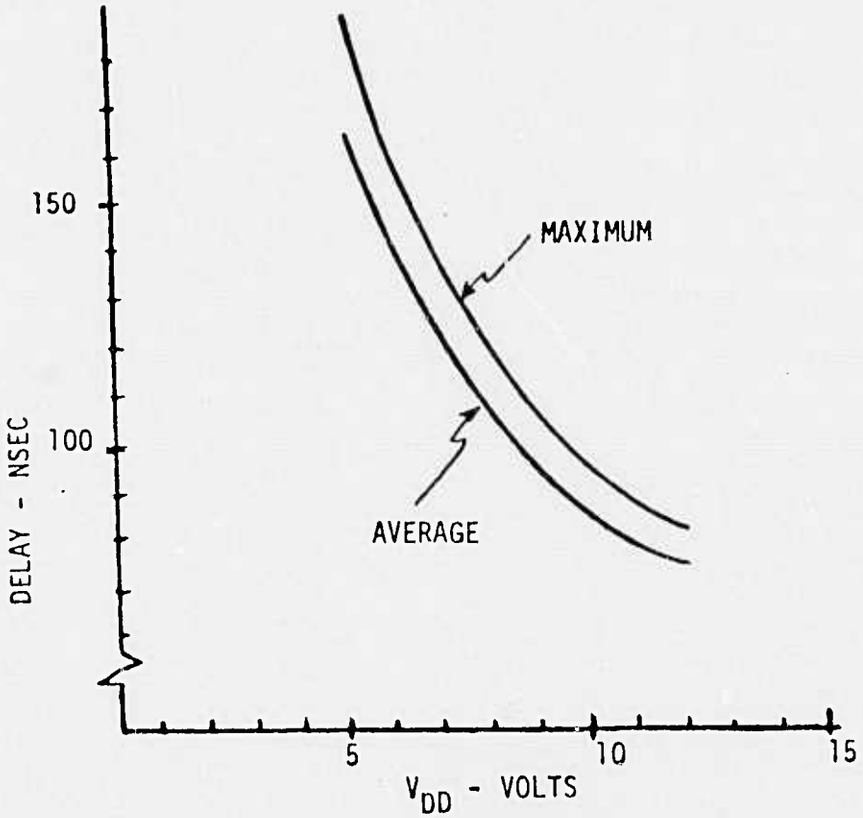


Fig. 5-5. Propagation delay of TCS-065 adder in 1's complement mode.

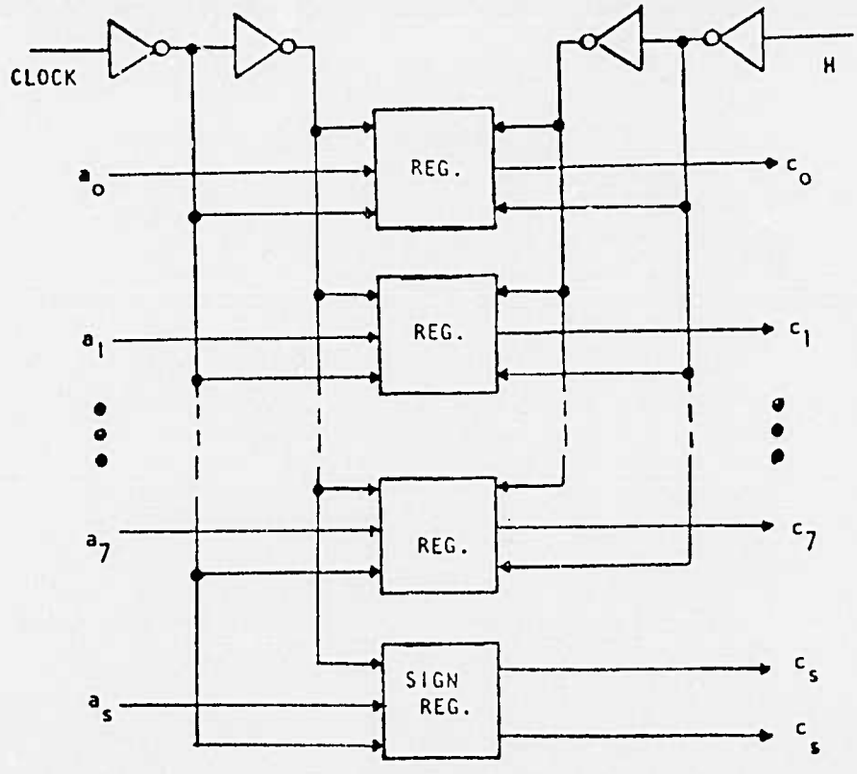


Fig. 5-6. Retimer registers (two per LSI array).

70

Section VI

HYBRID CIRCUIT ASSEMBLIES

A. GENERAL

Two approaches were used in the design, fabrication, and assembly of the four thick-film hybrid circuits used on the video encoder. Standard chip and wire bond with hermetic overall package techniques were used for the input interface/master timing and the output interface hybrids. This technique offered the most advantages for size and weight compression in these partitioned blocks because of the proliferation of common SSI IC chips used in the circuitry. Paragraph VI.B describes the common aspects of both of these hybrids as well as the unique individual differences.

The DCT and DPCM hybrids were designed to use leadless hermetic packages to hold the active semiconductor chips. This approach was particularly attractive because it afforded the opportunity to individually handle and test complex digital functions contained within each LHP prior to assembly on the hybrid substrate. It further facilitated the capability to program PROM chips, to burnin the CMOS/SOS LSI chips, to measure propagation delays of the LSI chips, and to functionally test the LSI chips in the breadboard system. Common and detailed information about the DCT and DPCM hybrids is given in paragraph VI.C.

Table 6-1 summarizes the pertinent statistical details about each of four hybrids in tabular form. Each of the tabularized items in the table is self-explanatory, with the possible exception of the difference between the number of functional and the number of actual interconnections. The number of functional interconnections refers to a wiring path connecting two components, or a component and an I/O, or a power bus lead on the hybrid. The larger number of actual interconnections is accounted for by the introduction of intermediate terminations within some of the functional interconnections, such as the wrap-around clips used for connections to pull-up resistors on the back of the DCT and DPCM hybrid substrates. It is also worthy to note that the equivalent number of transistors is high on the output interface and DPCM hybrids, largely because of the 1024-bit RAM chips used in their circuitry.

TABLE 6-1. VIDEO BANDWIDTH REDUCTION SYSTEM - ENCODER HYBRID DATA

	Hybrid			
	DCT	DPCM	Input Interface/ Master Timing	Output
Total No. of IC Chips	37	24	22	25
Chip Types				
CMOS-Bulk SSI	2	8	7	12
TTL (Schottky) SSI	4			
TTL (L.P. Schottky) SSI	10		14	7
TTL (L.P.) SSI			1	
TTL SSI	7	1		
CMOS/SOS Memory		4		6
CMOS/SOS LSI	9	6		
PMOS Register		2		
TTL Memory	1			
TTL - PROM	4	3		
No. of Chip Capacitors	17	9	3	
No. of Thick Film Resistors	45	26	2	9
No. of Chip Resistors	2		2	4
No. of Interconnections: Actual	1050	730		
No. of Interconnections: Functional	750	480	236	354
No. of I/O & TP Pins	52	53	46	46
No. of Interfacial Pins	27	22		
Equivalent No. of Transistors	29,916	41,940	1735	50,110
Planar Size (without leads) (inches)	4.20x3.00	3.00x3.00	1.25x1.25	1.25x1.25
No. of Interconnection Levels	4	3	4	4

B. INPUT INTERFACE/MASTER TIMING AND OUTPUT INTERFACE HYBRID CIRCUIT ASSEMBLIES

Both of these hybrids were designed to fit on a 1.12 by 1.12 inch square 96 percent alumina substrate 0.035 inch thick. The substrates in turn fit into a 1.25 x 1.25 inch, 46-lead Tekform hybrid case. Four levels of metalization were generated to provide the required interconnection pattern. A total of nine screen patterns were used to build the four-level interconnection patterns.

Each screened-on pattern was dried and fired immediately after being applied. The three insulating dielectric levels were screened and fired on the substrate twice to preclude the occurrence of pinhole short circuits. Each substrate intended for use was manually tested for continuity of nets and for the absence of short circuits before chip components were mounted.

Both hybrid pattern sets were generated from manual layouts, which were digitized on an Applicon interactive graphics terminal. The Applicon tape outputs were used to generate CalComp plotter review prints and the final Gerber plotter photographic artwork.

The silicon IC chips were bonded to their respective pads using Epotek H44 gold conductive epoxy. The sapphire IC chips, resistor chips, and capacitor chips were bonded in their locations using Epotek H61 non-conductive epoxy. Wire bonds were made between the chip connection pads and the substrate metalizations with ball and stitch thermocompression welds of 1.0-mil-diameter gold wire.

The completed substrates were epoxy bonded in their packages with Epotek H44. One-mil-diameter gold wire with ball and stitch thermocompression weld bonds was used to connect the substrate pads to the 46 I/O leads. Figures 6-1 and 6-2 show the output interface and input interface/master timing hybrids, respectively, in detail prior to their covers being sealed. Each hybrid was completely electrically tested at ambient, low (-30° C), and high (+65°) temperatures before being sealed.

A Tekform flat-lid cover was used to seal the hybrid packages with Sn 62 soft solder. The solder was used to tin the faying surfaces of the package and the cover and then reflowed to attain the seal. Hermeticity of each completed hybrid was assured by gross and fine leak testing. All units were also given final electrical tests before being assembled on the encoder printed-wiring board.

C. DCT AND DPCM HYBRID CIRCUITS

The DCT and DPCM thick-film hybrid assemblies utilize a design approach in which all active semiconductor chips are packaged in leadless hermetic packages (LHPs) designated for solder bonding to pads on a thick-film interconnection pattern screen-printed on a ceramic substrate. By adding soldered-on spring-clip input/output leads to this assembly, the mechanical configuration of the hybrid circuit is completed. The base substrates used for the DCT and DPCM hybrids were 0.040 inch thick, and their planar dimensions were 4.2 x 3.0 inches and 3.0 x 3.0 inches, respectively. In both hybrids, the thick-film resistors were screened on the back of the substrates. Interfacial connections between the front and the back surfaces of the substrates were made with spring clips soldered to pads on the edge of the substrates. Differences in the interconnectivity requirements and complexities between DCT and DPCM circuits required the use of four interconnectivity levels for the DCT and three for the DPCM.

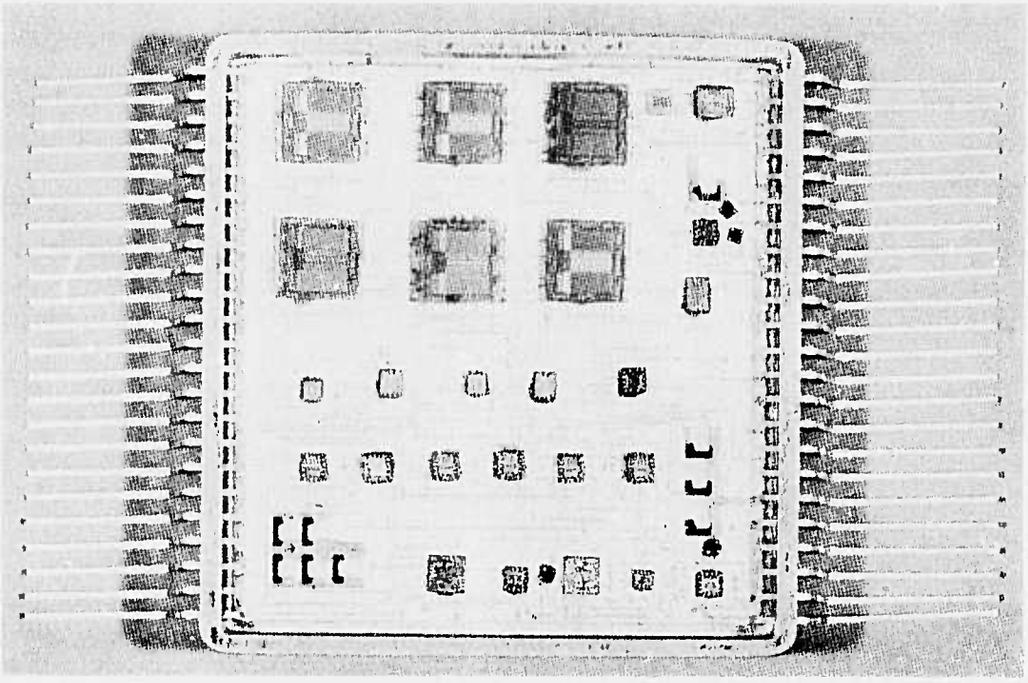


Fig. 6-1. Output interface hybrid without cover.

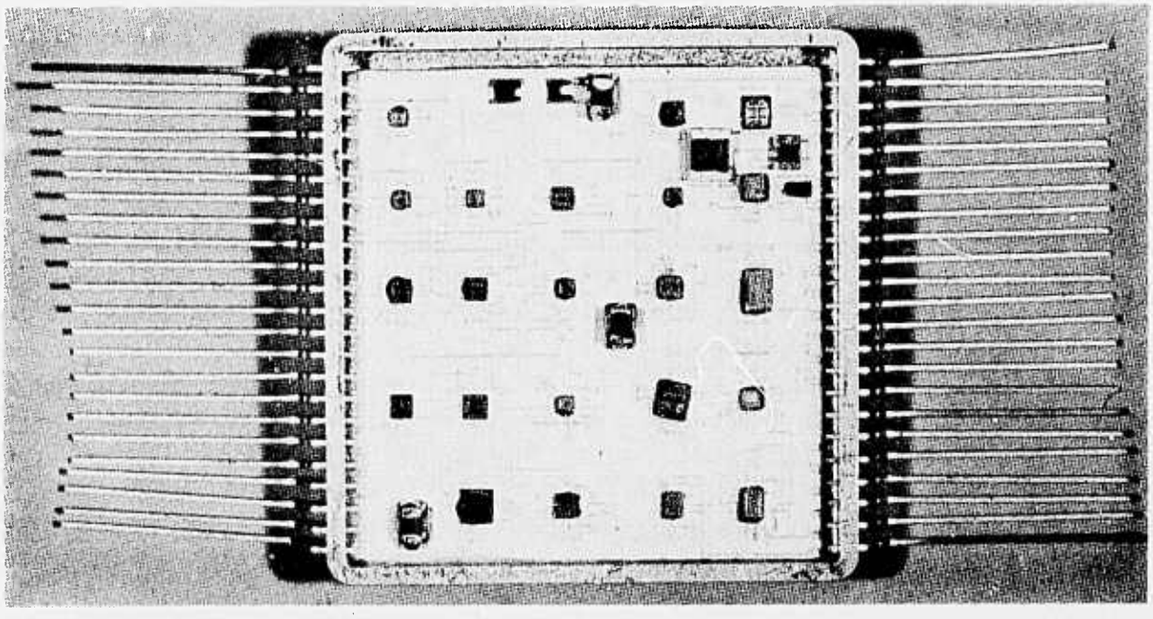


Fig. 6-2. Master timing hybrid without cover.

Assembly of the DCT and DPCM hybrids was completed by placing the LHPs, chip capacitors, lead frames, and wrap-around edge clips in their appropriate locations and reflowing the solder cream by vapor-phase immersion techniques. Figures 6-3 and 6-4 show the completed DCT and DPCM hybrid assemblies.

Chips that were packaged individually in the appropriately selected 28- or 48-pad LHPs included the RAMs, PROMs, adders, multipliers, and retimers. The SSI devices used in the DCT and DPCM circuits were packaged as subhybrids in 48-pad LHPs. The subhybrids were made using 10-mil-thick 96 percent alumina substrates with a 1-level DuPont 9791 gold metalization interconnection pattern screened on it. Crossovers were made using 1-mil gold wire stitches over areas on the substrate protected by DuPont 9841 bonded to the substrates with DuPont 5504 conductive epoxy. One-mil gold wires bonded between the chip pads and their substrate pads and between the substrate input/output pads and the package pads were thermocompression weld bonded. The covers on all the subhybrids were formed to compensate for substrate thickness in providing bond wire clearances. The covers, of gold-plated ASTM alloy F15, used on all the LHPs were hermetically sealed on the packages by reflowing gold-tin preforms attached to the covers.

Test pattern sequences were generated for all LHP components. These test patterns were used to test all of the LHP components on a Datatron automatic array tester prior to their assembly on the hybrid substrates. The CMOS/SOS LSI pads, retimers, multipliers, and adders were further burned-in under static electrical stress for a minimum of 100 hours at 135°C. Propagation delays were then measured through a critical path on these chips after burnin. The chips were then functionally tested in the breadboard system and assigned to like circuit locations for assembly on the hybrids.

A 40% oxygen environment was required to program the Intel PROM chips. This programming was accomplished by placing the unsealed PROM LHPs in fixtures adapting them to the Prolog PROM programmer inside a "clean box" filled with a 40% oxygen 60% nitrogen gas. The PROM programs were then copied from a master and verified. The LHP was sealed and the programming reverified.

Two repair procedures were accomplished on the LHP hybrids. On the DCT, an incorrectly programmed PROM was replaced. To implement this type of repair, the substrate was heated to 100°C, and localized heat was applied to the top of the LHP to be replaced. When the solder on the pads liquified, the LHP was removed with a knife blade. The LHP replacing the one removed was positioned in location and the solder was reflowed, using the same method as was used to remove the LHP.

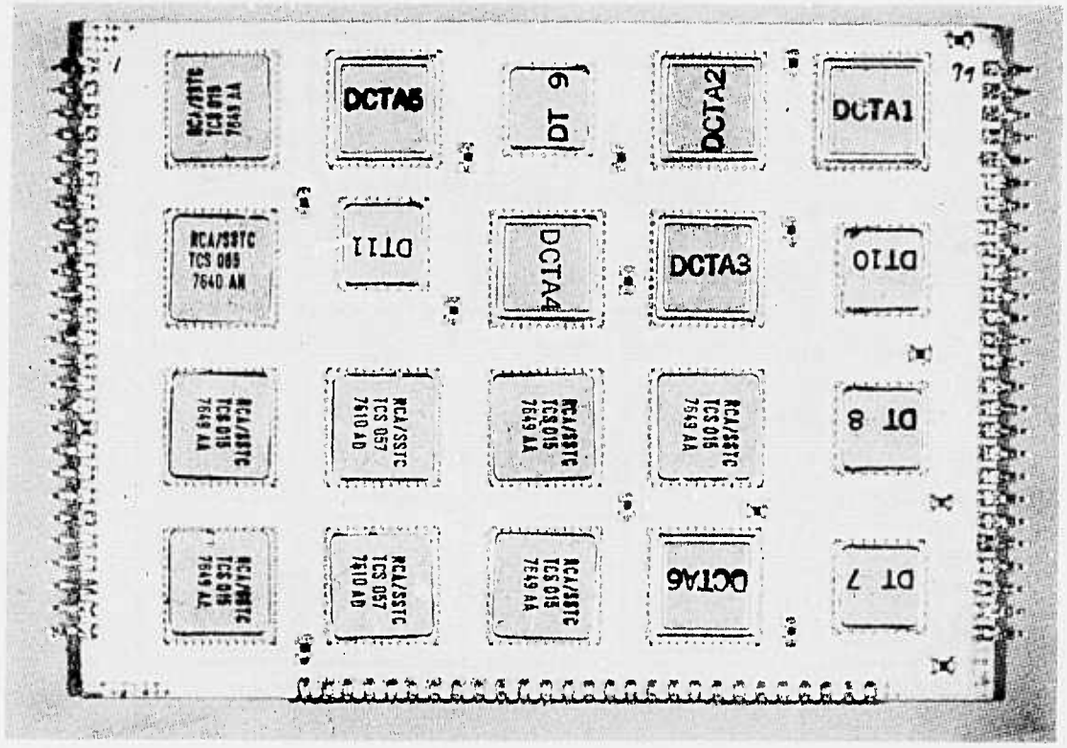


Fig. 6-3. DCT hybrid assembly.

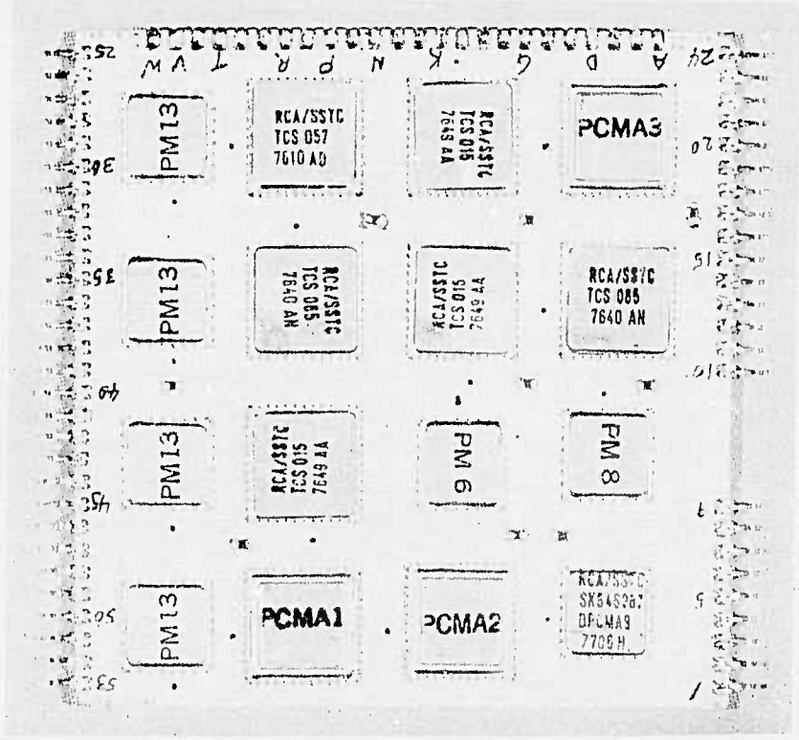


Fig. 6-4. DPCM hybrid assembly.

76

On the DPCM, excessive voltage drops were experienced in the 5 V dc distribution between the input leads and the LHPs where utilized. This voltage drop was corrected by the addition of wires from both the 5-V input lead and the ground lead to similar circuit locations at decoupling chip capacitors on the other side of the substrate. The location of these wires may be observed on the DPCM hybrid on the airborne assembly (Fig. 7-1). In any subsequent production of the DPCM hybrid, the need for the external wires could be eliminated by the use of a gold metalization pattern for power distribution in place of the presently used platinum-gold.

Section VII

HARDWARE PACKAGING

A. AIRBORNE UNIT

The airborne video bandwidth-reduction circuitry is packaged on two printed-wiring boards, mechanically fastened together by standoffs to form the configuration shown in Fig. 7-1. The uppermost, smaller of the boards is 5.0 inches by 2.0 inches and contains all of the power conditioning and filtering components required between the input power and the power required by the encoder board. Interconnections between the power conditioning board and video encoder board are provided through #20 AWG type E polytetrafluoroethylene insulated stranded copper wire solder-terminated in plated-through holes on the ends of the board. To minimize the size and envelope height of the power conditioning board, chokes L1 and L2 were custom wound and potted for use. Choke L1 was wound on a Magnetics Inc. toroidal core, using 100 turns of #22 AWG heavy armored polythermaleze wire to obtain an inductance of 0.8 mH. The wrapped core was dip-coated with GE type RTV-11 resin and then potted in a Robison Electronics Inc. potting cup, along with a #2-56 center located machine screw, using Emerson and Cummings Stycast #1090 potting resin. The choke is fastened to the board by its center-located machine screw, and its leads are terminated in plated-through holes in the printed wiring.

The smaller choke, partial reference designation L2, was wound on a Magnetics Inc. toroidal core using 24 turns of #22 AWG heavy armored polythermaleze wire to obtain an inductance of 20 μH. The wrapped core was dip coated with GE type RTV11 resin to provide a resilient flexible mechanical protection. Choke L2 is mechanically and electrically attached to the power conditioning board by its leads, which terminate in printed wiring after passing through plated-through holes.

The power conditioning printed-wiring board was made by using two-sided, 2-oz solder-plated copper-wiring patterns with plated-through holes on 0.063-inch-thick flame-retardant fiberglass epoxy laminate. Power inputs to the board flow from the four input pads, through the conditioning components, exiting through the nine pads on the right side of the board. Included among the nine pads are the common ground return for both the system and input power.

All of the remaining power conditioning components, regulator ICs, transistors, diodes, resistors and filter capacitors were mounted in conventional printed-wiring fashion. Appropriate spacer pads were used under all of the active TO-5 case type components. A Wakefield heat dissipator was used on the 2N4236 transistor to assure its safe operation at high temperature.

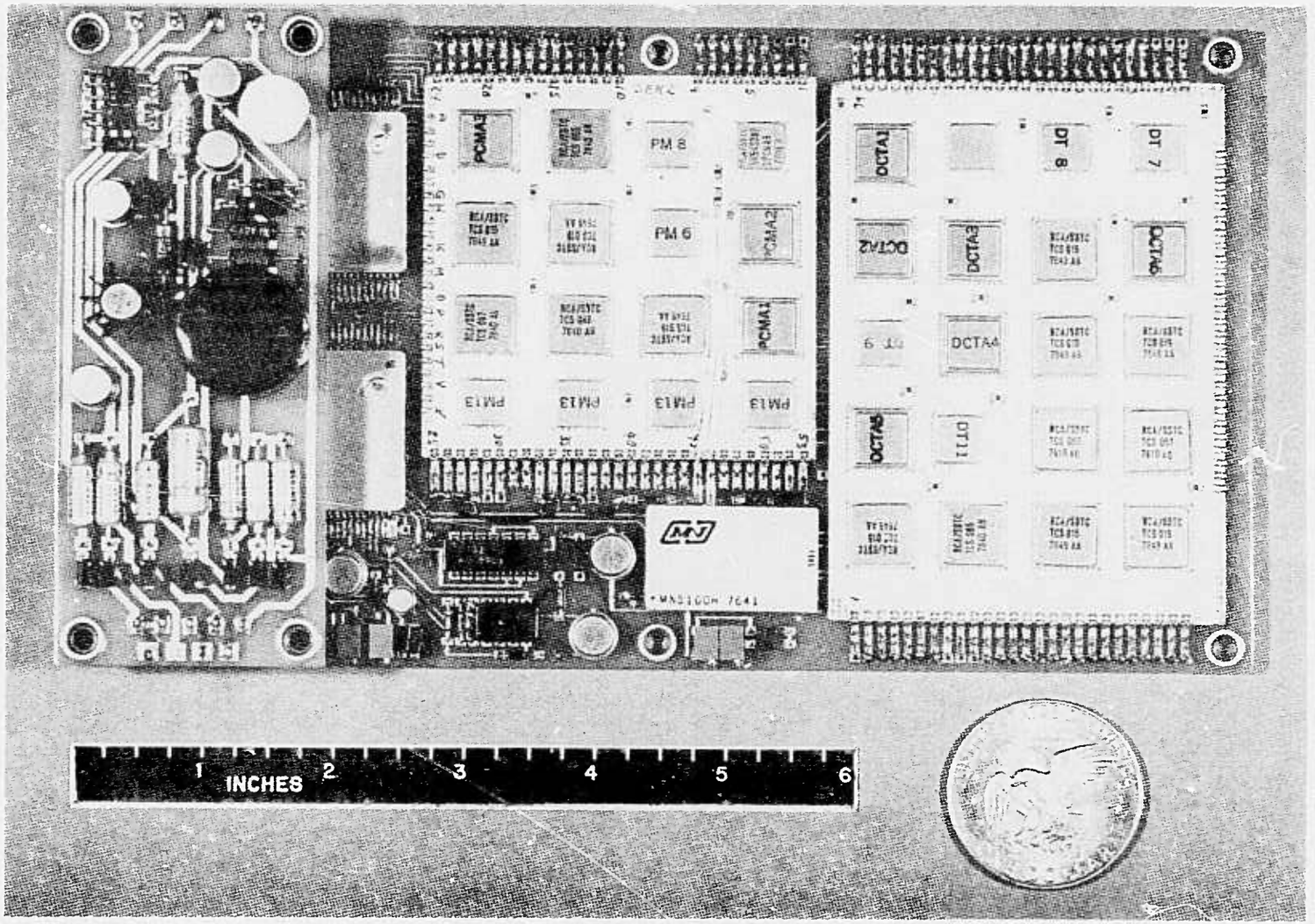


Fig. 7-1. Airborne unit.

The lower printed-wiring subassembly (Fig. 7-2a) contains all of the signal processing for the airborne video bandwidth-reduction encoding system. This circuitry was mechanically partitioned for arrangement on the board into the following electrical blocks: analog video signal processing; input interface/master timing; A/D converter; discrete cosine transform (DCT); differential pulse code modulation; output interface; and input/output buffering. The location of the components comprising each of these blocks is shown on Fig. 7-2b.

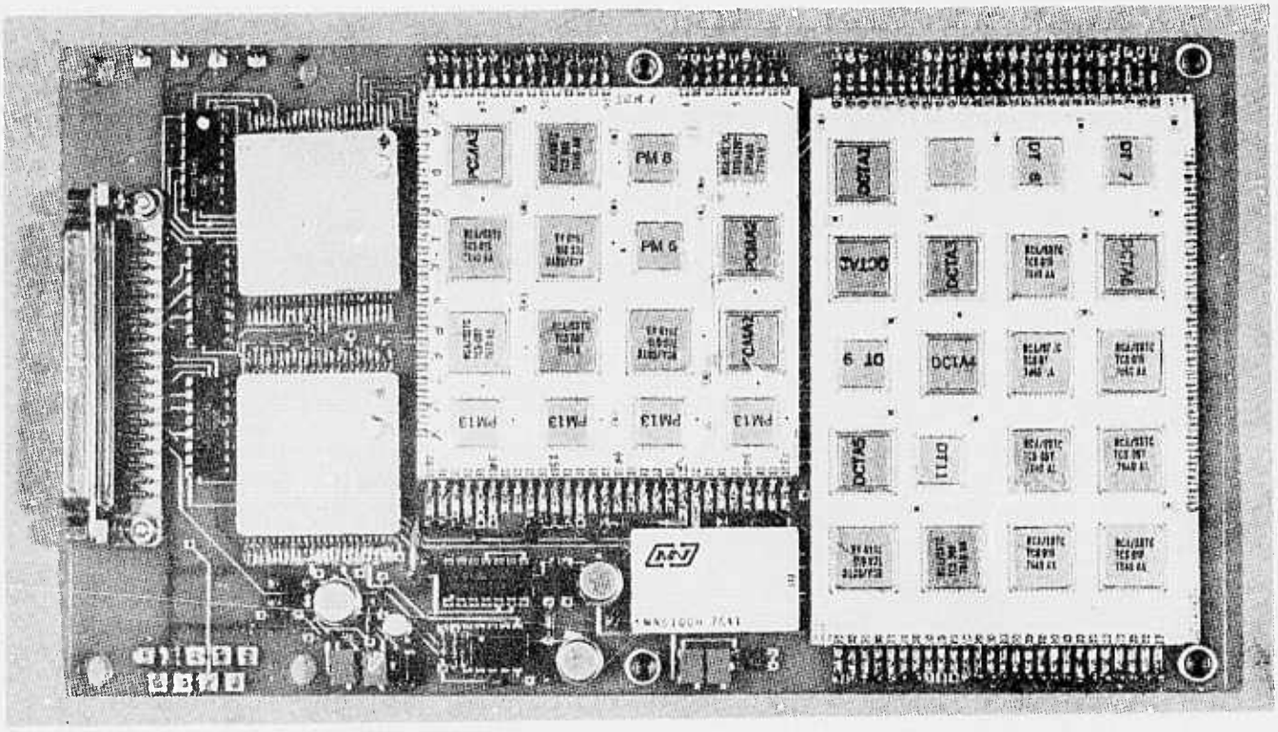
The video encoder printed wiring board is 0.063 inch thick and is 9.0 inches long by 5.0 inches wide. It was made of flame-retardant fiberglass epoxy laminate. It has three conductor layers of 2-oz copper interconnected with plated-through holes; the outer conductor surfaces were solder plated.

The four hybrid assemblies used on the video encoder board are surface mounted by their leads to wiring pads on the board to provide minimal risk of catastrophic damage to either the board or the hybrids if replacement is necessary. The leads on the hybrids are formed in a "stair step" configuration to provide stress relief and to provide space under the packages to preclude moisture, flux or other contaminant entrapment. The remaining components on the board were mounted in standard printed-wiring fashion. Spacer pads were used under all of the vertically mounted can components. Small terminal pins are used in a number of locations on the board for test points and to provide termination points for the wires extending from the power conditioning board.

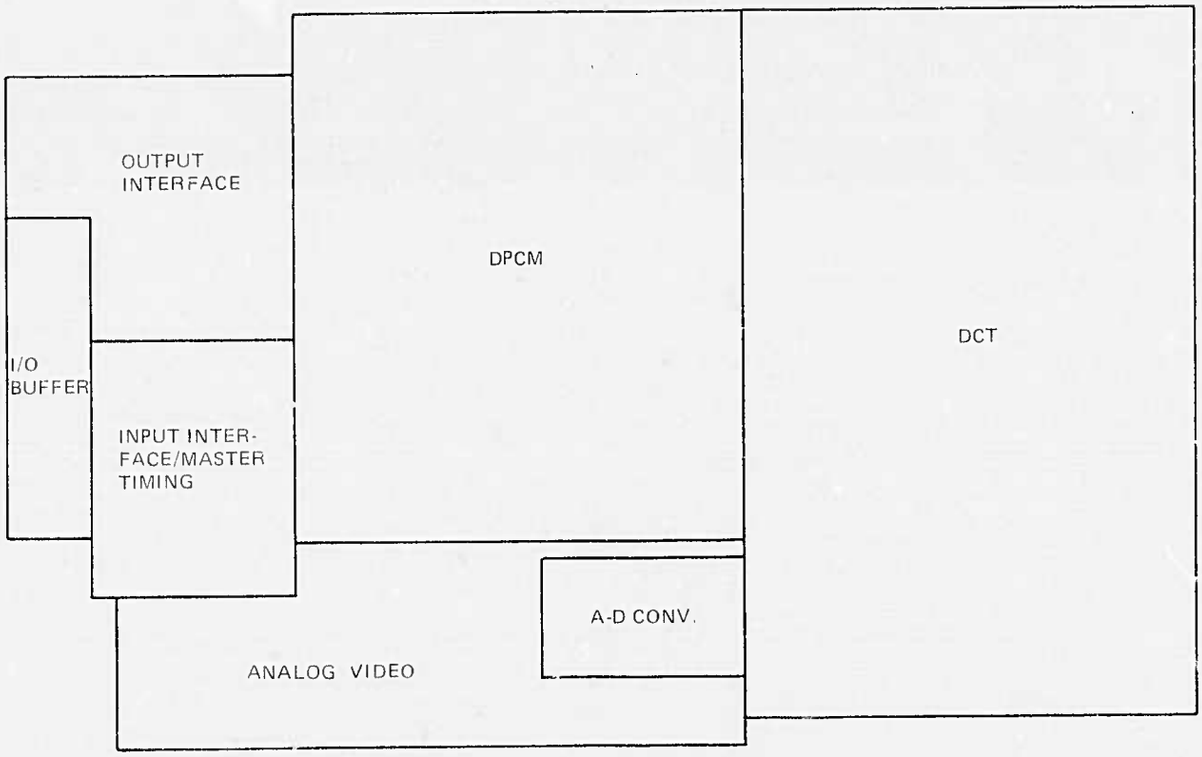
Standoffs are "roll-over" riveted in four locations on the board to provide 0.150 inch spacing from the assembly mounting surface. A similar mounting surface spacing is provided at the other four holes in the board by the nuts used to secure the standoff power conditioning board mountings to the video encoder board. All eight standoffs have 0.156 diameter holes to facilitate mounting in the aircraft.

The entire airborne video encoder assembly, except the connector and test point terminals, was coated with a moisture-fungus proofing (MFP) two-part polyurethane varnish, which conforms to the requirements of specification MIL-I-46058.

All of the components that were used on the airborne assembly were selected to minimally meet environmental temperature extremes from -29°C to +65°C, with an air velocity of 18 ft/s. The pacing component in achieving high-temperature performance was the DCT hybrid with a worst-case power dissipation of 8.5 W. This level of power was a major factor in selecting the planar size of the DCT hybrid. A smaller double-sided hybrid configuration was considered for use, but rejected in favor of the larger radiating area of a single-sided hybrid. Empirical temperature measurements were made on a number of locations on the DCT mounted on the airborne assembly in a natural-convection laboratory environment. The measurements were made by using Tektronix P6058 probe with a DM501 digital voltmeter in a TM503 manifold. Appropriate correction factors were applied to the measured values to



a. Circuit board.



b. Board layout.

Fig. 7-2. Large airborne circuit board.

81

compensate for the thermal gradient measurement effects and the probe heat-sinking effects. Figure 7-3 shows these corrected values in the locations on the DCT where they were measured. The ambient temperature was 27° C. Measured power dissipation during the tests was a total of 8.24 W, consisting of 5.51 W of 5-V power and 2.73 W of 10-V power.

The program goal of keeping the weight of the airborne unit less than 1.0 lb was readily attained by the final assembly measured weight of 0.78 lb.

B. GROUND STATION

The Video Bandwidth-Reduction System ground-station circuitry is packaged in two chassis designed for slide mounting in an EIA standard 19-inch-wide panel cabinet with a minimum depth of 23 inches. The circuitry in one chassis contains the decode electronics. The other chassis contains the frame store memory and video output circuit. Both units have self-contained power supplies operating from standard 115 V, 60 Hz ac input power. The ground-station interface specification defines in detail the complete mechanical and electrical interfaces with the two chassis. Paragraphs below provide more detailed descriptions of the decode electronics and frame store memory chassis.

1. Decode Electronics Chassis

The decode electronics chassis was designed as a modification of an Augat rack assembly with a back panel for thirteen 1/8-inch-thick boards. The modifications consisted of replacing the side plates with new sides to mount the power supplies, slides and I/O connectors and the addition of a functional control panel on the front surface. Figure 7-4 shows the decode electronics chassis mounted in a rack with all of the plugin boards and an airborne unit in test position. BNC connectors are provided on the panel for video in and sync signals out.

The right side panel provides mounting for two 15-V dc power supplies, one 5-V dc power supply, three over-voltage protectors, and the mounting plate and right slide. The input power connector is mounted on a wrap-around back section of the right side panel. The left side panel provides mounting for the left slide and its adapter plate, a +15 V dc to +10 V dc power supply circuit and the I/O connectors on its wrap-around back section.

The electronics contained on the plugin boards in the decode chassis were partitioned into the following functions shown from left to right in Fig. 7-4: airborne test fixture, test function generator, modem signal simulator, master timing and input interface, inverse DPCM and 2 boards with the inverse DCT. Hand wire-wrap was used to wire the boards. Axial lead and can type components were soldered either in adapter plug assemblies or between wire-wrap pins on the wiring side of the boards.

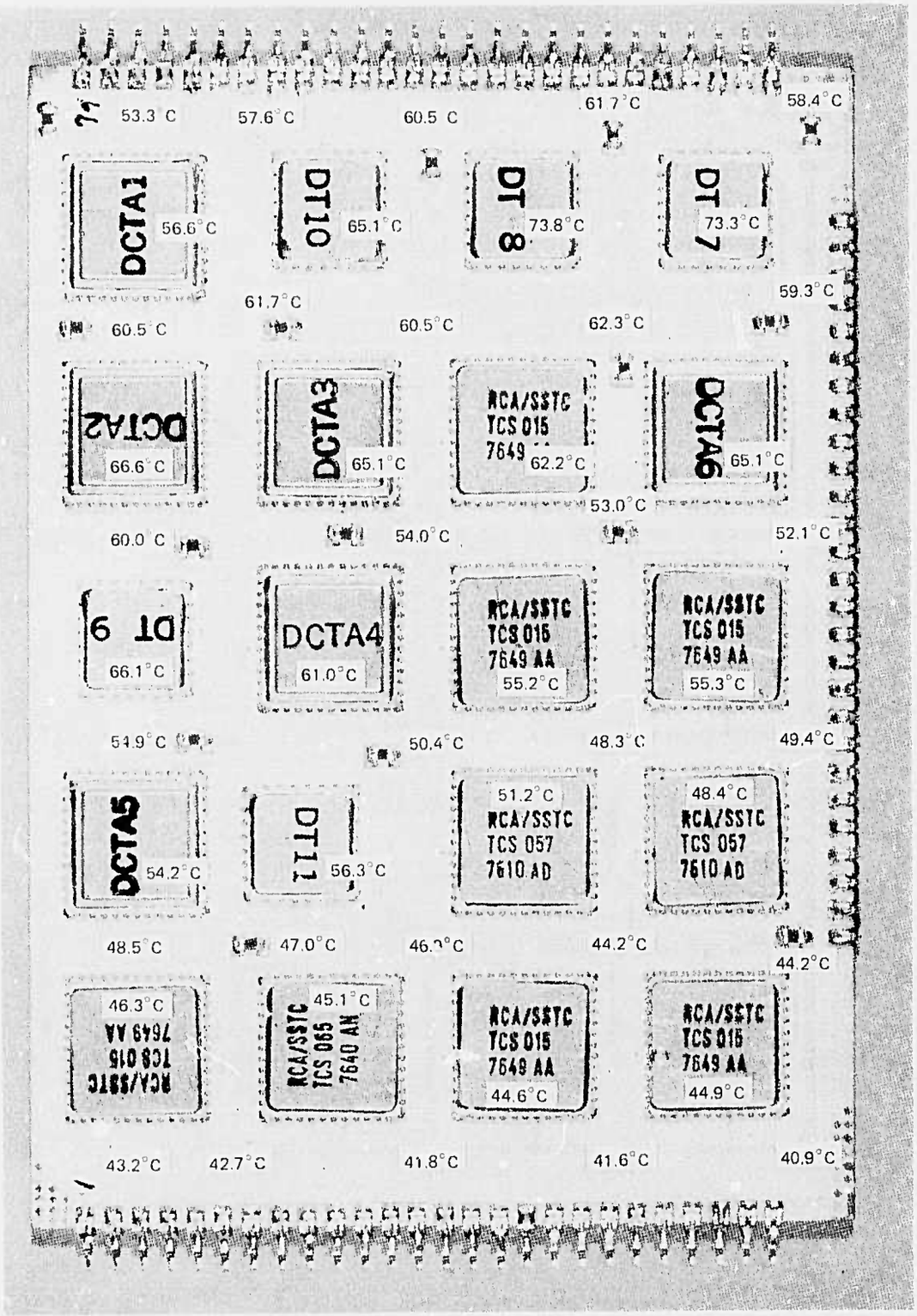


Fig. 7-3. DCT hybrid temperature profile.

83

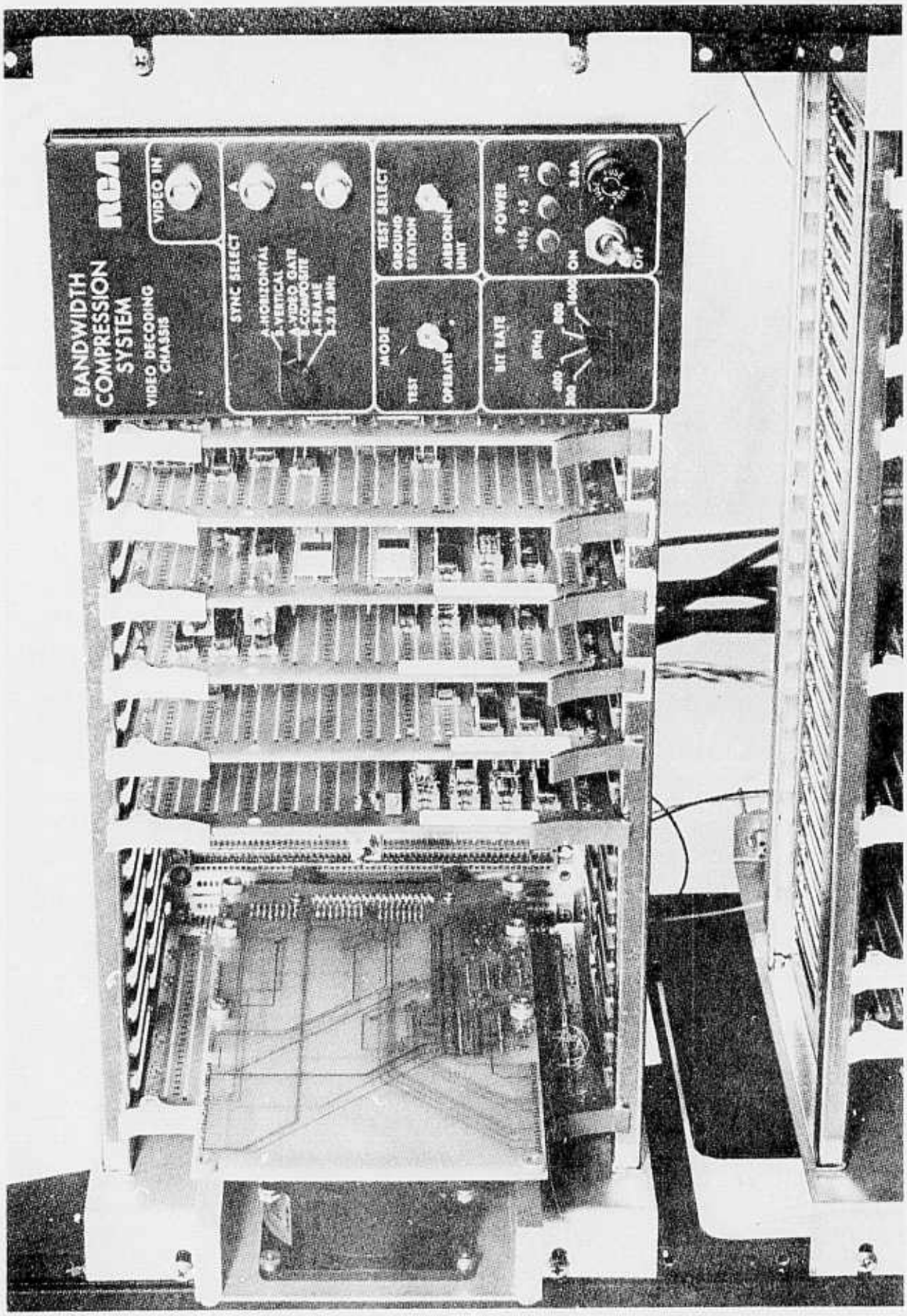


Fig. 7-4. Decode electronics chassis.

84

The airborne test fixture adapts the airborne board to a plugin board connector through a Cannon number DCM-37S connector and mounting rails designed to provide machine screw mounting faces for the center two standoffs on the airborne board. The mounting rails are also used to support a 115-V ac Rotron venturi fan arranged to provide forced air cooling normal to the DCT hybrid. To reduce the possibility of injury from shock several precautions were taken. First the 115 V ac power is applied to the backplane connector only when the decode electronics is in the airborne test mode of operation. The backplane of the connector has a clear plastic cover fastened over it stenciled with cautionary wording. The backplane connector is triple, keyed so that only the test fixture board and a special extender card will fit into it. The test fixture has a plastic cover with a caution note over the wire-wrap terminals that connect to the connector pins through which the 115 V ac passes. Electrical tape is used to cover the other (turret) end of these terminals on the component side of the board. An Augat extender card was modified for use in the airborne test slot position. The modifications consisted of 1) adding two notches in the edge connector to accommodate the triple connector keying and 2) bonding a strip of 0.010-inch-thick fiberglass epoxy with the caution warnings stenciled on it over the copper paths which carry the 110 V ac power.

Backplane interconnection wiring on the decode electronics chassis utilized wire-wrap connections for the most part. Stranded wires from the I/O connectors and coaxial wire connections from the TNC connectors were solder-terminated on the backplane wire-wrap pins.

2. Frame Store Memory

The frame store memory chassis was modified from a standard Electronic Memories and Magnetics Corporation chassis for a Microram 3400N-2 system. The modifications consisted of replacement of the I/O connector plate and connectors, addition of angle brackets and slides to the top of the chassis and replacement of the plastic front panel with a panel that covers the added height for the angle brackets and slides. A 115-V ac power supply capable of providing all of the required dc power used in the chassis was purchased with the memory. It occupies the right side of the unit as a bolted-in subassembly.

The electronics in the frame store memory are contained on three boards that plug into the left side of the chassis. These cards have planar dimensions of 11.75 inches by 15.40 inches with two 80-tab edge-card connectors on them. Two of the boards (a 32K x 12 bit Microram 3400 N-2 memory board and the memory self-test board) were used as supplied by the vendor. The third board contains memory control and video output circuitry. This board was assembled and wired by RCA, using a purchased universal logic card. Wiring was wire-wrapped on this board on the side on which the components were mounted. The memory and self-test boards use printed-wiring interconnections.

Backplane wiring in the chassis utilized wire-wrap techniques. Stranded wire connections from the 55-pin and the TNC I/O connectors were soldered to their respective wire-wrap termination pins on the backplane.

The modification adding the slides on angle brackets on top of the chassis was necessary because the chassis was too wide to permit their addition on the sides and still fit an EIA 19-inch-wide rack opening. The angle brackets used were 1.75 inches high. Adapter bars were provided with the slides to provide the correct vertical chassis position on the EIA rack mounting stripe.

Appendix A

SUBSYSTEM DESIGN

A. AIRBORNE UNIT

1. Analog Circuits

The function of the analog circuitry of the airborne unit is to sample the incoming video at the proper times, digitize the video to 6 bits, and make the digitized signal available to the DCT when required. Paragraph A.5 of this appendix describes the timing relationships. Suffice it to note here that the video must be sampled at 208-ns intervals (i.e., with a 4.8-MHz clock). Although analog-to-digital converters are available which can operate at this rate, they are, at present, too bulky and use too much power for use in the airborne board. Consequently, an analog storage system was adopted which permits the use of an A/D converter whose size and power demand are compatible with the restrictions on the airborne unit.

The video input (see Fig. A-1) is specified to be from 0 V (black) to 1 V positive (peak white). Blanking and sync are to be negative-going, with sync tips 0.3 to 0.4 V negative. The input amplifier inverts this signal, amplifies it by a factor of about 4, and shifts its level to about 2.5 V (white) to 6.5 V (dark). The potentiometers which control the gain and level set are located on the side of the airborne board, near the input connector, with the gain control nearer the connector. These controls may be adjusted, if necessary, by injecting a 0-to-1 V triangular or sawtooth signal (synchronized to the video gate) into the video input and observing the output of the tapped analog delay (TAD). Gain and bias are then set for the greatest signal that will pass through the TAD without limiting. Alternatively, if the overall system is operating, and the A/D input controls have already been adjusted, the system output with a normal camera input may be observed on a monitor and the controls adjusted for best picture out.

The white clipper, a 2N2432 transistor, is connected in an emitter-follower circuit with its emitter biased at 2.5 V. Any signal more positive than 2.5 V at the emitter will be passed by the transistor; any less positive than 2.5 V will be clipped. If a bright portion of scene content drives the input video more positive than 1 V, the signal at this point, having been inverted and level shifted, will go less positive than 2.5 V and be clipped. Such clipping is needed because the TAD, when driven to a voltage less positive than its acceptable input range, will remain saturated for some time after the overload has passed. In practice, the consequence of this is that a bright portion of a scene would have a trailing white tail. The clipper prevents this.

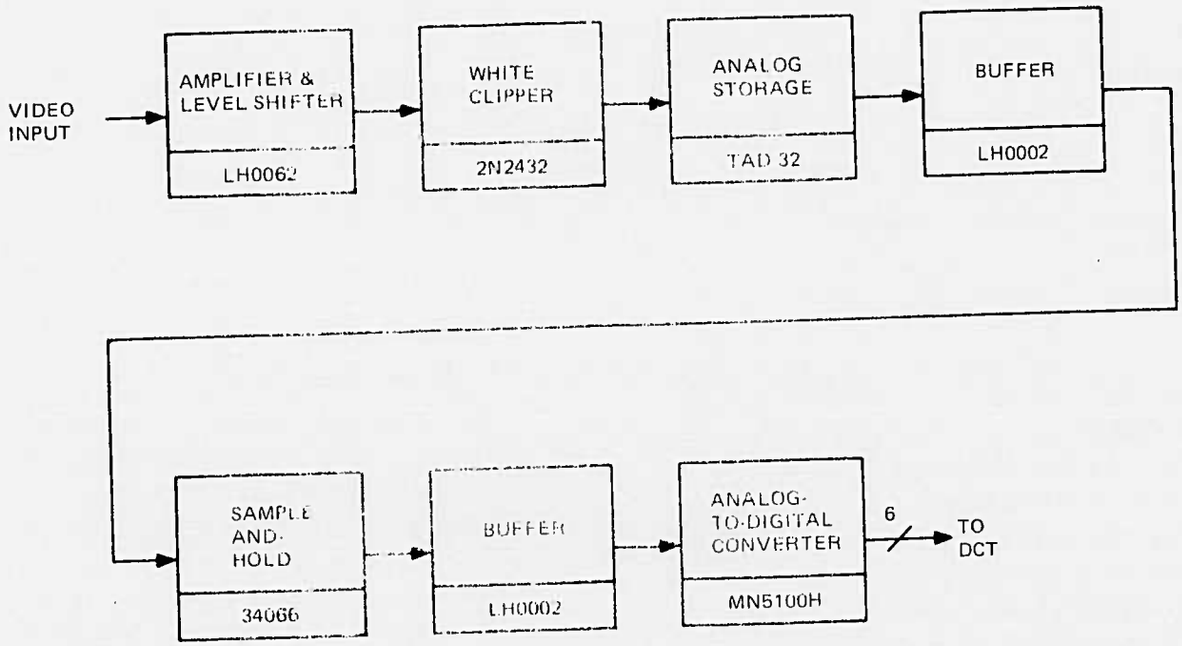


Fig. A-1. Block diagram of airborne analog circuits.

Analog storage is done in a TAD-32 tapped analog delay device from Reticon Corporation. The TAD-32 is normally mounted in a 40-pin package, but since space is limited on the airborne board, and the intermediate taps are not needed, the device was repackaged in an 8-pin package with only the feed-forward output tap available. The TAD is therefore operated as a 32-stage FIFO analog storage device.

The TAD clock runs at 4.8 MHz while the video is being sampled. Immediately after the 32nd sample is taken, the clock rate is reduced by a factor of 7, i.e., to one clock pulse for every 7 of the 4.8-MHz cycle times. The TAD clock is a square wave during signal acquisition; during readout the advance takes place in one 4.8-MHz interval, and the output is present for six 4.8-MHz intervals. This allows the output to settle prior to sampling. Following the TAD is an LH0002 buffer, having high input impedance, low output impedance, and nearly unity voltage gain.

The sample-and-hold circuit, consisting of a 34066 quad transmission gate with all four sections in parallel and a storage capacitor, holds the input to the A/D converter constant during conversion time. Following the sample-and-hold is another LH0002 buffer.

The MN5100H analog-to-digital converter is operated with a 4.8-MHz clock. It is capable of conversion to 8 bits, but its conversion cycle is terminated after 6 bits.

2. DCT

The DCT subsystem accepts thirty-two 6-bit time samples from the A/D converter and transforms them into 32 frequency coefficients, using the discrete cosine transformation (DCT) algorithm. The 32 frequency coefficients are transferred to the differential pulse code modulator (DPCM) subsystem.

Basic timing for the DCT is supplied by the master timing function and includes the line initiate pulse (DCTLIP), 4.8-MHz clock (48 CK1), and 3.2-MHz clock (32 CK1). DCT processing is initiated by DCTLIP and continues until halted by the internal DCT program.

a. DCT Subsystem Design

For purposes of discussion the DCT will be divided into the two subfunctions: timing and control, and processing as shown in Fig. A-2 and A-3.

The timing and control logic consists of TTL devices which generate the PROM addresses write enable (WREN) for the RAM and the five clocks (CL1-CL5) used by the DCT processor. Three timing signals are also generated to accomplish the transfer of data to the DPCM. The WRITE, CC1, CC2 and STOP levels from the PROM control the sequencing of the logic. External clocks and LIP complete the inputs to the timing and control logic.

DCT LIP resets the PROM address counter, which is then clocked continuously until the HALT bit appears in PROM 3. Clock for this counter is either 48 CK1 or 32 CK1, as determined by CC1 and CC2 control levels from PROM 4.

PROM 3 is programmed to provide five addresses for the TTL RAM, which is used for temporary storage in the DCT processor. The sixth RAM address bit is modified by the select level from PROM 3 as detailed later. Since all RAM addresses are for TTL logic, no level shifting is required.

PROM 4 provides a WRITE bit which causes data to be written into the RAM. SUB D level determines when the DCT processor performs a subtract operation rather than an add. A SCON D level determines the scaling of the adder results to avoid adder overflow conditions in the processor.

PROMs 1 and 2 provide the sine, cosine, and whitening coefficients to the processor. W1 and W2 words may also contain all ones or all zeros as required for the DCT algorithm. Level shifters are used to supply the 10-V logic levels for the DCT processor.

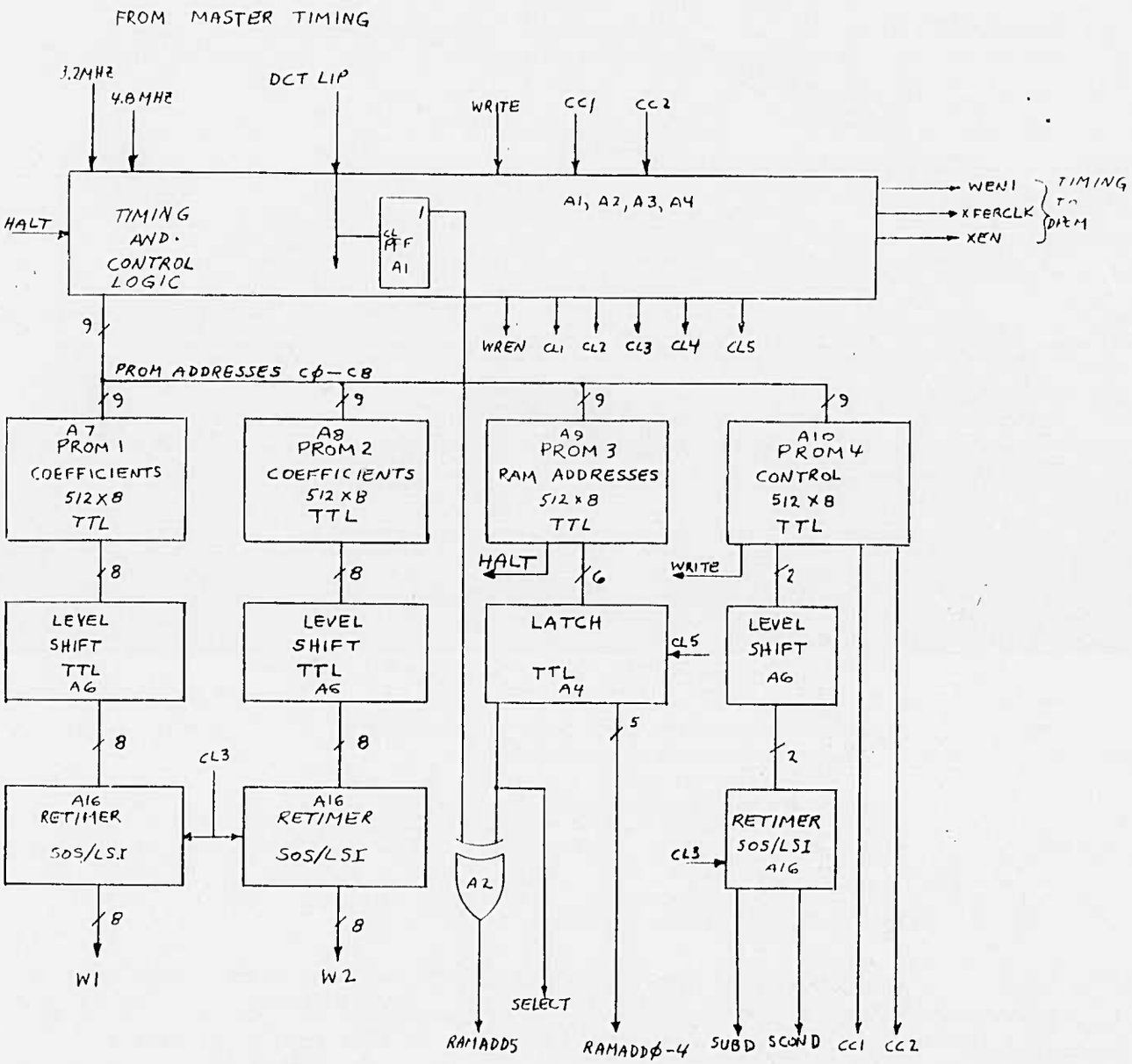


Fig. A-2. Block diagram of DCT hybrid timing and control.

90

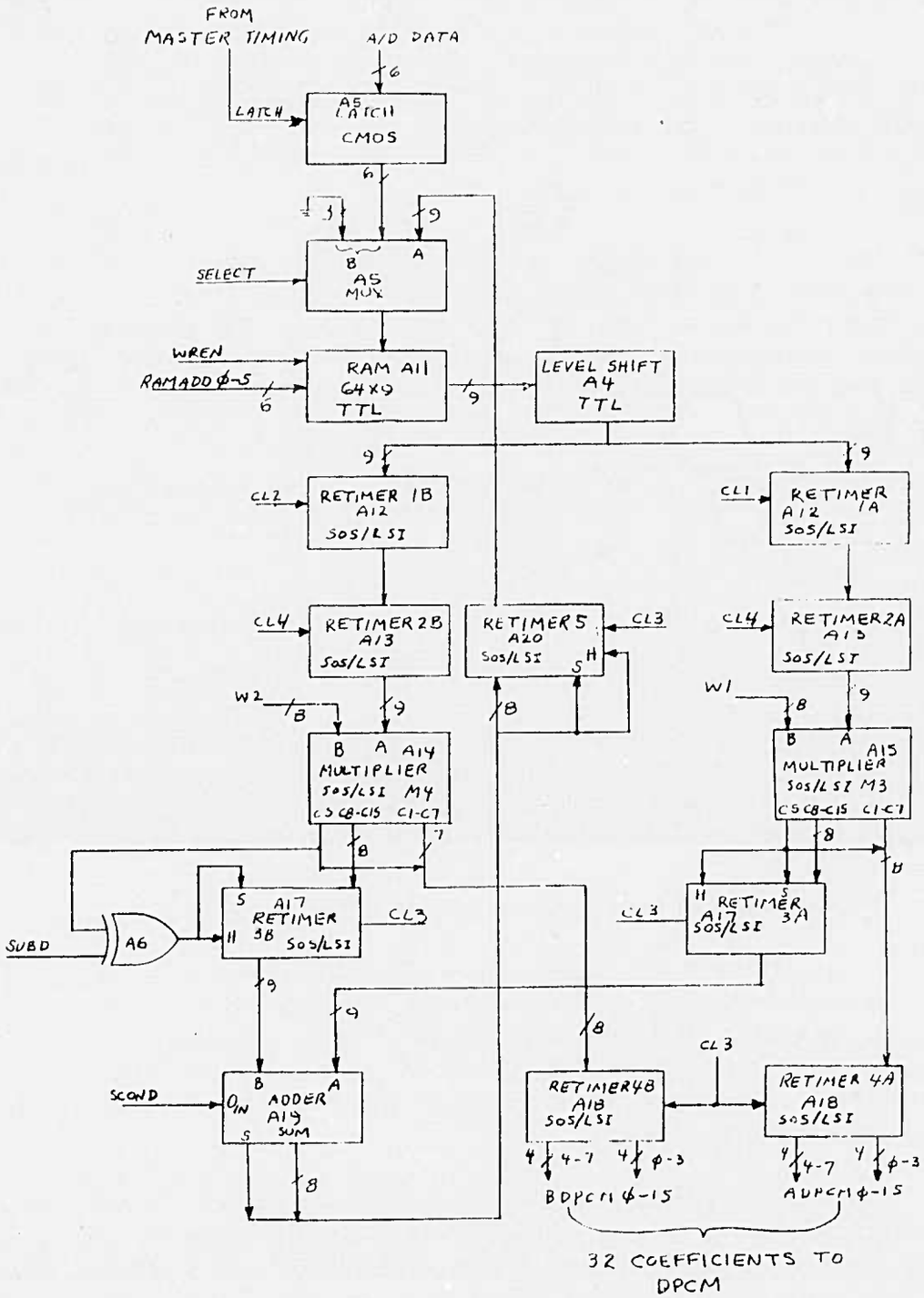


Fig. A-3. Block diagram of DCT hybrid processor.

91

The DCT proeessor uses SOS/LSI arithmetie funetions organized in a pipeline eonfiguration. Input A/D data are held in a lateh in sign magnitude form ready for use in the proeessor. A multiplexer (MUX) seleets either the A/D or the pipeline output to be written into the bipolar RAM, whieh operates with two types of eyeles. Thirty-two write word eyeles are spaeed at seleeted eloek periods throughout the line interval for writing new data samples. During the remaining eloek periods, data proeessing is performed and the memory operates in a READ/WRITE mode where a read oeeurs on the rising edge of the eloek and new data are written into the loeation just read at the falling edge of the eloek.

Data are organized into two pages in the RAM, one for new samples and one for aetive proeessing. The roles of two pages are reversed at eaeh LIP by aetion of the PFP in the eontrol and timing logie. At eaeh of the 32 data write times the Seleet level will eause the eurrent RAMADDS to be eomplemented, and new data will be written into the other data page. The nine RAM outputs are level shifted to 10-V levels for the LSI funetions.

The arithmetie portion of the pipeline performs the operation:

$$Y_1 = W1\ (N)A + W2\ (N)\ B$$

$$Y_2 = W1\ (N+1)A - W2\ (N+1)\ B$$

where W1, W2 are eonstants from the PROMs, as deseribed earlier.

Retimers 1A and 1B are eloeked on alternate cloek eyeles by CL1 and CL2 to lateh in the A and B words noted above. CL4 eloeks retimers 2A and 2B every time retimer 1B is eloeked. Thus A and B will be present at the inputs of multipliers M3 and M4 for two eloek times. W1 and W2 will ehange every eloek eyele so that a new produet is available at the basie eloek rate of 4.8 MHz. The multipliers are eonneeted to provide a truneated produet for the 9 most signifieant bits.

Retimers 3A and 3B hold the most signifieant half of eaeh produet for one eloek eyele at the adder input parts. The adder is conneeted for one's eomplement operation. Retimer 3A eonverts the produet from M3 into one's eomplement form. Retimer 3B in eonjunetion with the EXCLUSIVE-OR gate eonverts the produet from M4 into one's eomplement form and also eomplements the data if the SUB D eontrol level ealls for a subtraet operation.

In addition to performing the 1's eomplement addition, the adder is eonneeted so that the SCOND level will eause the output sum to be right adjusted at six preseleeted eloek times in the proeessing interval to avoid overflow of the data in the adder.

During the final processing before the next LIP, the W1 and W2 constants will be whitening coefficients, and valid data will be present in the least significant half of the M3 and M4 outputs. The 32 whitened coefficients are clocked into Retimers 4A and 4B in 1's complement form. Eight of the whitened coefficients are transferred as the negative of the actual value, which saves an additional pass through the pipeline. This difference of sign is easily accommodated in the inverse DCT of the ground station and requires no additional programming or hardware.

When the first whitened coefficient is ready for transfer to the DPCM, a transfer enable pulse (XEN) is generated. Transfer clock (XFERCLK) and write enable WEN1 complete the timing signals for the DPCM interface. Transfer takes place at a 1.6-MHz rate for each pair of coefficients, thereby giving an effective rate of 3.2 MHz. Note that the pipeline is also clocked at 3.2 MHz during this transfer.

At the end of the transfer interval a HALT bit is generated which stops the clocking of the address counter until the next LIP appears.

b. DCT Hybrid

The DCT function consists of 20 leadless packages on a ceramic substrate containing four levels of metalization.

TTL logic elements were used extensively in the timing and control logic functions due to the 4.8-MHz clocking speed. This logic is contained in sub-hybrids A1 through A6, which are partitioned as follows:

A1 - control flip-flops and clock gating

A2 - PROM address counter and control flip-flops

A3 - Dual one-shot with associated resistors and capacitors

A4 - Level shifters and RAM address latch

A5 - A/D input data latch and multiplexer (MUX)

A6 - Level shifters

Pull-up resistors for the level shifters are of the thick-film type and are located on the obverse side of the substrate to avoid additional subhybrid packages. Additional TTL packages are the four PROMs and one RAM. The remaining nine packages are SOS/LSI and use only the 10-V supply.

The preceding discussion has shown that 11 of the 20 packages are TTL using the 5-V supply. Five of these packages (A7, A8, A9, A10, A11) operate at the 700-mW level or higher. In addition, Schottky devices are contained in A2 and A3 and require adjacent bypass capacitors to eliminate noise problems. Thus the +5 V

bus and ground bus were considered critical and were given special consideration on the DCT. Eleven chip-type bypass capacitors of 0.1μ F each have been distributed along the +5 V bus. All buses were made wider and gold was used for the metalization in anticipation of the higher currents. The results have been very satisfactory, with maximum voltage drops less than 30 mV, and no noise problems have been encountered.

3. DPCM

The DPCM accepts 32 frequency coefficients from the DCT and computes the encoded video coefficients (EVC) which are transferred to the output interface. The DCT supplies a transfer clock (XFERCCK), Transfer Enable (XEN), and a Write Enable (WENI) to accomplish the data transfer to the DPCM. Master timing furnishes a line initiate pulse (DPCMLIP), and 600-kHz and 4.8-MHz clocks. Each EVC going to the output interface has an associated Encoded Video Address (EVA) and a Write Enable Pulse (WENZ). Two lines for data rate selection complete the output interface signals.

The DPCM hybrid circuit consists of a ceramic substrate on which are mounted 16 leadless integrated circuit packages.

Control and timing logic consists of three packages of which A4 is an SOS-LSI. Two packages, A1 and A2, contain discrete TTL logic chips which are connected to perform the random logic used in timing and control.

A third subhybrid package contains two TMS3112 shift registers chips to provide the 9 by 32 bit shift registers which store 32 data words from the previous line time.

Three packages are TTL programmable read only memories (PROM). A6 and A8 are Intel 3604-L6 chips and in a 512 x 8 bit format. The modifier PROM, A9, is a smaller 256 x 4 chip, 54S287.

The remaining nine packages are SOS-LSI functions of which A13, A14, A15, and A16 are the four RAM chips which store the 32 words from the DCT. A5 and A10 are parallel adders used for 1's complement addition or subtraction of operands of 8 bits plus sign.

A12 is a sign magnitude multiplier which multiplies the value from the previous line by the scale factor (SF) which is set to $(E8)_{16}$. Each operand is 8 bits plus sign. The 8 most significant bits plus sign of the multiplier output are used as the feedback term in the DPCM. The roundoff/truncate input, R, is connected to ground so that the output represents a truncated version of the product.

Paekages A7 and A11 eaeh eontain two retimer registers of 8 bits plus sign. As a useful feature, the "H" input of each register may be conneeted to provide eonversion from 1's eomplement to sign magnitude or the reverse as required by the implementation.

The DPCM requires three power supply voltages: +10, +5, and -12. Package A3 uses PMOS teehnology and is the only function requiring -12 V.

In the interest of low-power operation, most of the DPCM logic is powered from the 5-V bus. This ineludes TTL and SOS-LSI devices. However, the deviees interfaeing with the DCT operate at a 10-V level for two reasons. One reason is that level translation is simplified by doing the translation after the interfaee. The other reason is related to speed. Although the RAMs and eontrol logic operate with a 600-kHz eloek for most of the proeessing cycle, there is a time during the transfer across the DCT interfaee when 4.8 MHz is used — hence the higher speed requirement.

Since the DPCM uses a minimum of TTL logic, no problems were anticipated with voltage drops on the substrate buses. However, after the final substrate interconneetion tradeoffs were completed, the TTL deviees were loeated across the substrate from the +5-V input terminal. Subsequent testing after assembly showed that the drops in the +5-V and ground buses to these devices were exeessive. To avoid major artwork ehanges, two number 24 stranded wires were added on top of the substrate to eure the problem. Small dots of polyurethane strapping eompound were used to mechanieally secure the wires to the substrate.

4. Output Interfaee

The output interface reeeives 6-bit encoded video eoeffieients from the DPCM and transmits this data as a serial bit stream to the modem. Coefficients reeeived from the DPCM are synehronous with LIP and the 600-kHz eloek, while the serial data sent to the modem must be synehronous with horizontal sync and modem eloek. Figure A-4 presents the basic bloek diagram of the output interfaee.

In order to accommodate the timing variation of LIP with respeet to horizontal syne over the 8 stripes, a three-seetion RAM is employed to buffer the coefficients from the DPCM. Parallel data from the RAMs are eonverted to a serial bit stream and retimed to the modem eloek. The number of bits transmitted from eaeh eneoded video coefficient is determined by the data rate seleetion.

To reduee the power eonsumption of the output interfaee hybrid, extensive use is made of CMOS ICs in the low-speed portions of the eireuit. Low-power Sehottky TTL ICs are used only in those portions of the eireuit whieh will operate at the maximum 1.6-MHz eloek rate. The data buffer memory employs CMOS/SOS RAMs to fulfill the speed requirements yet maintain a low power consumption.

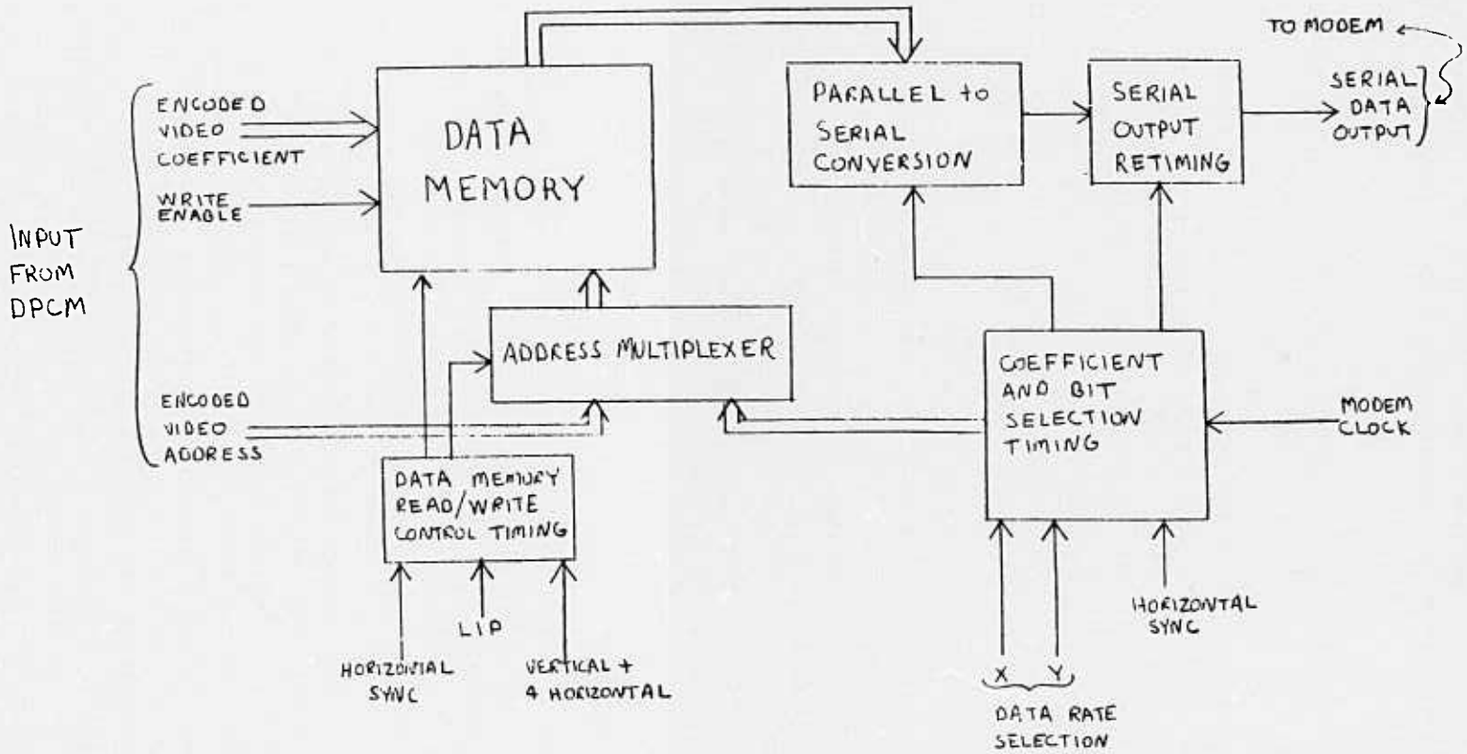


Fig. A-4. Block diagram of output interface.

Figure A-5, a timing diagram of the output circuit, shows that because of the processing delays in the output circuit, the first bit of a horizontal line is displaced to the second place, and all ensuing bits are similarly displaced. The ground station generates a horizontal sync which compensates for the displacement.

5. Master Timing Hybrid

The master timing hybrid accepts externally supplied 9.6-MHz clock, horizontal drive, vertical drive, and frame sync signals, and provides timing and control signals to the DCT, DPCM, and output interface hybrids and to the analog input and A/D conversion circuitry. The output signals are tabulated in Table A-1.

The basic signals which control the operation of the airborne bandwidth-compression system are a write start signal and a line initiate pulse (LIP). The occurrence of write start marks the beginning of sampling of input video by the TAD. After 32 samples are input, the start of processing by the A/D converter, DCT and DPCM is initiated by an LIP. The method by which the appropriate 32 samples are selected is discussed below.

TABLE A-1. OUTPUT SIGNALS

Signal	Destination
4.8-MHz DCT clock	DCT/DPCM hybrids
4.8-MHz A/D clock	A/D converter
3.2-MHz DCT clock	DCT hybrid
1 TAD clock	TAD 32
2 TAD clock	TAD 32
600 MHz DPCM clock	DPCM hybrid
Sample TAD output	CMOS analog switch
Start convert	A/D converter
Latch A/D data	DCT hybrid
DCT LIP	DCT hybrid
DPCM LIP	DPCM hybrid
Vertical + 4 horizontal	Output interface hybrid

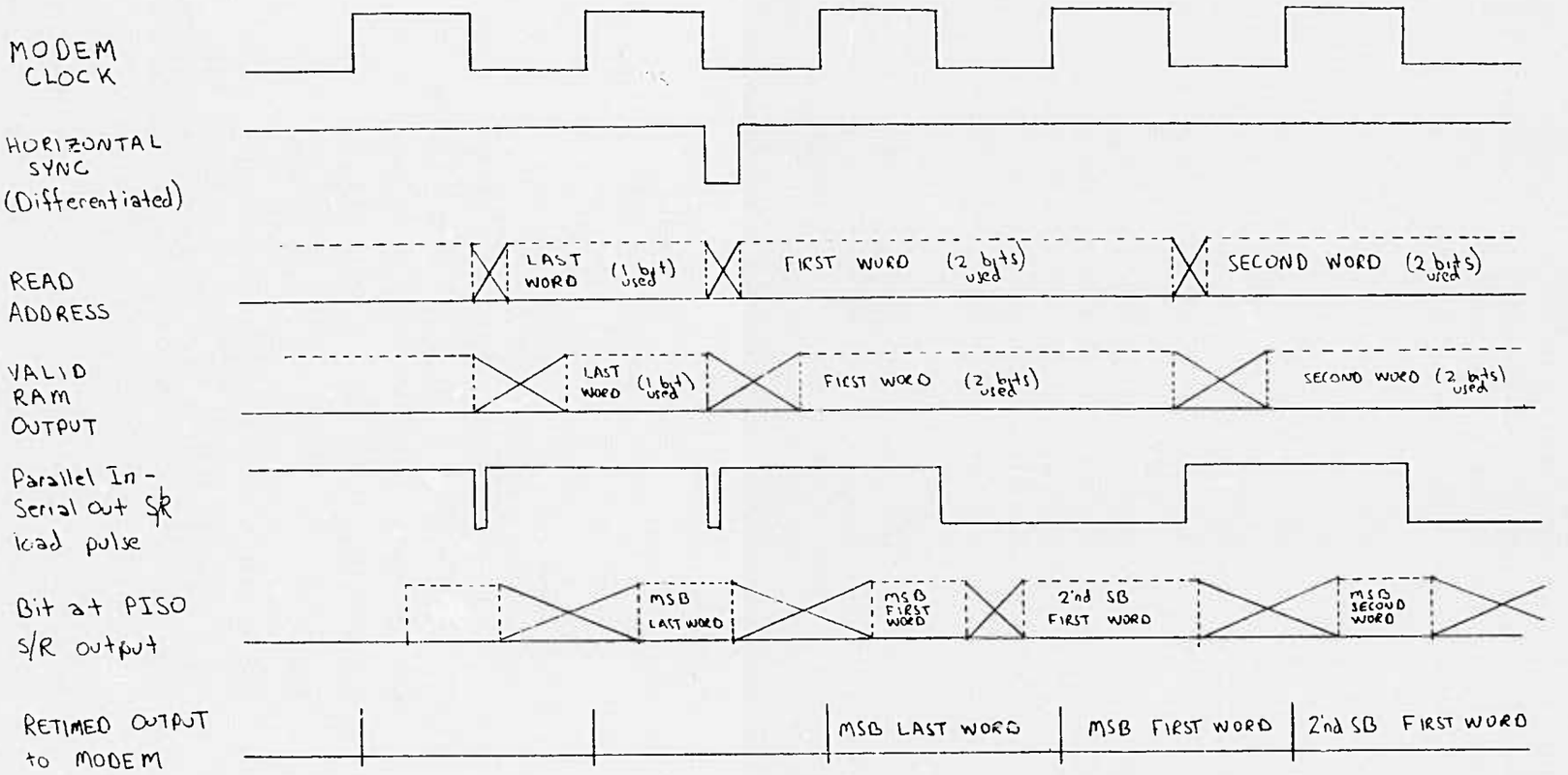


Fig. A-5. Output interface hybrid output timing.

98

a. Stripe Selection

The active stripe is determined by the state of the stripe counter (block 9 in Fig. A-6). Frame sync clears the stripe counter, marking the first (leftmost) vertical stripe. The stripe counter is advanced by vertical drive delayed by four horizontal pulses. This delay is provided by the horizontal line counter, block 8, to accommodate the system throughput delay.

b. Write Start/LIP Delay

After a given stripe is selected, an appropriate delay from the horizontal pulse is necessary to locate the window for sampling input video. This delay is the same for each line of video in a stripe.

No matter which stripe is active, initial delay must be provided from the horizontal pulse to reach active video after horizontal blanking. This blanking delay is provided by the 6-bit counter of block 1, the first stage of the 9-bit shift register of block 2, and the decode logic of block 4. The 6-bit counter is cleared by each horizontal pulse, which also sets up a "one" at the input of the 9-bit shift register and resets the decode logic (block 4). The 6-bit counter then proceeds to count 9.6-MHz clock pulses. At a count of 32, output Q6 of the counter rises, clocking a "one" to the Q1 output of the shift register. After 96 of the 9.6-MHz clock pulses, shift register Q2 output will go high. Assuming that the stripe counter indicates the first (leftmost) stripe, the Q2 output pulse propagates through the 8-to-1 MUX (block 5) and enables the decode logic (block 4). This logic introduces an additional delay of 8 of the 9.6-MHz clock pulses, for a total delay of 104 of the 9.6-MHz clock pulses (10.8 μs) from horizontal drive to stripe 1 write start. Stripe 1 LIP follows stripe 1 write start after an additional delay of 64 of the 9.6-MHz clock pulses (32 samples of video at 4.8 MHz).

As the stripe progresses from the first (leftmost) to the eighth (rightmost), more and more delay is called for between the horizontal pulse and the LIP. This is obtained by allowing the "one" set up by the horizontal pulse to propagate farther down the 9-bit shift register before being channeled to the decode logic by the 8-to-1 MUX.

c. Input Timing

The input timing is provided by the 7-bit shift register of block 6 and decode logic of block 7.

A write start pulse signals the switching of the composite 2-phase TAD clock to 4.8 MHz for sampling input video. When 32 samples have been input to the TAD 32, an LIP occurs. This sets up the proper bit pattern in the 7-bit shift register and resets the decode logic (block 7), switching the TAD clock to (4.8 ÷ 7) MHz for

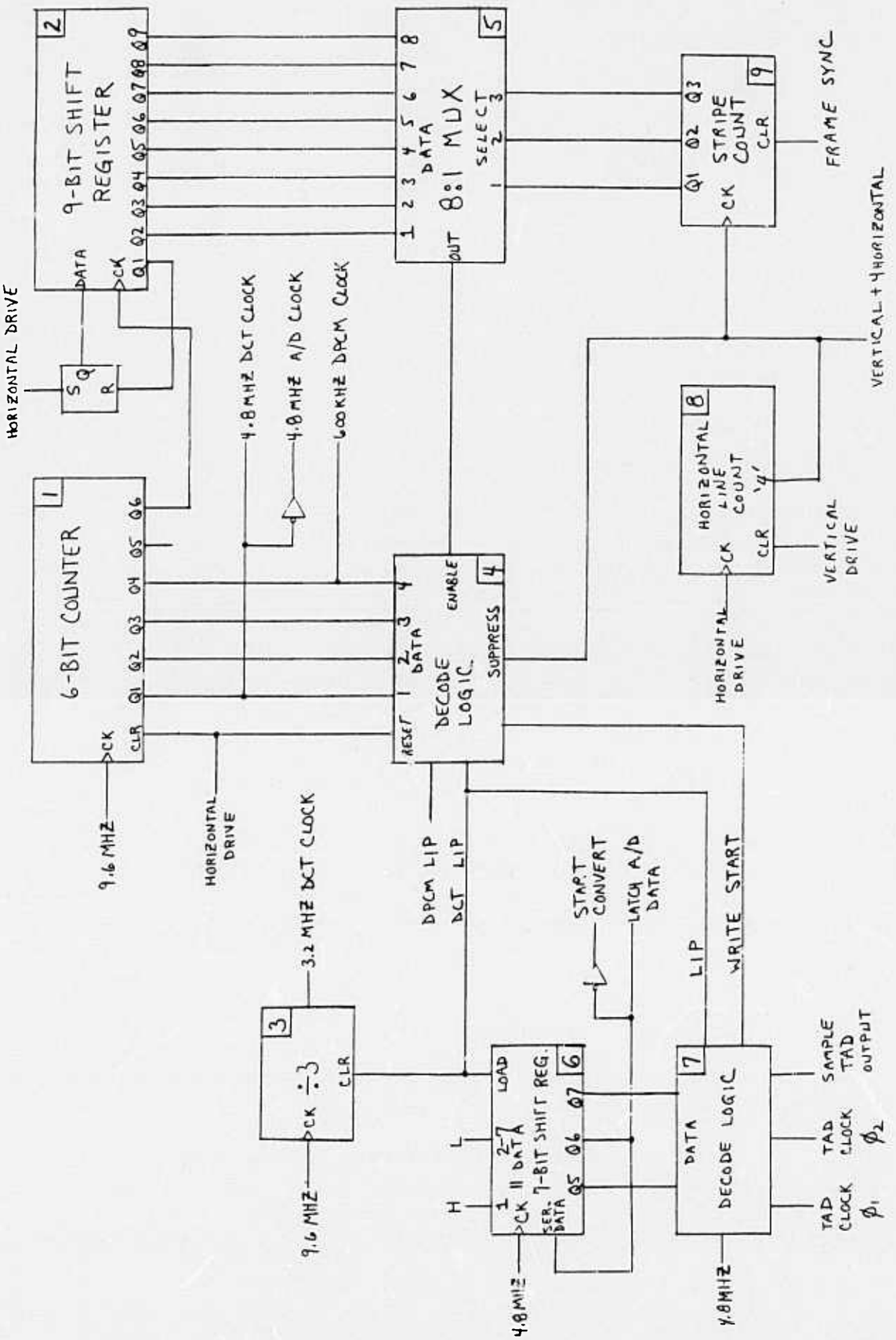


Fig. A-6. Block diagram of master timing hybrid.

100

reading out the video for A/D conversion. Other outputs of the 7-bit shift register are used to supply a pulse to sample the TAD output and to initiate and latch the result of each A/D conversion. The exact timing of these events is shown in Fig. A-7.

d. System Clocks

The 6-bit counter of block 1 supplies 4.8-MHz clocks to the DCT and DPCM processors and the 600-kHz DPCM clock. A separate ÷ 3 converter (block 3) supplies the 3.2-MHz DCT clock. The phasing of these clocks with respect to the LIP is important and is shown in detail in Fig. A-8.

e. Additional Details

Figure A-9 shows the details of events occurring near the time of stripe advance. After the vertical pulse, four horizontal pulses are counted by the horizontal line counter (block 8). At this time the "vertical + 4 horizontal" signal rises, advancing the stripe counter (block 9). In addition, the "vertical + 4 horizontal signal is used in the decode logic of block 4 to suppress the subsequent LIP. This is done to ensure that no two LIPs are closer than 65 μ s together, as would otherwise be the case when the stripe changed from eighth (rightmost) to first (leftmost).

The "vertical + 4 horizontal" signal is also sent to the output interface hybrid for reset purposes.

6. Airborne Unit Power

The following voltages and power levels are available to the airborne unit from the ICNS supplies:

Voltage	Power Available	Power Taken by Encoder
+15 V dc	9.25 W	6.45 W
+ 5 V dc	7.5 W	7.50 W
-5.2 V dc	0.1 W	0.05 W
-15 V dc	6.0 W	0.84 W

In addition to the voltages and currents available from the airframe, the encoder requires +10 and -12 V supplies, plus 2.5 more watts at 5 V. To fulfill these requirements, a power conditioning unit has been designed as a 2 x 5 inch PC board mounted on standoffs at the connector end of the airborne unit. This board provides bypass filtering for all the supplied voltages. To meet the +5 V dc power requirement

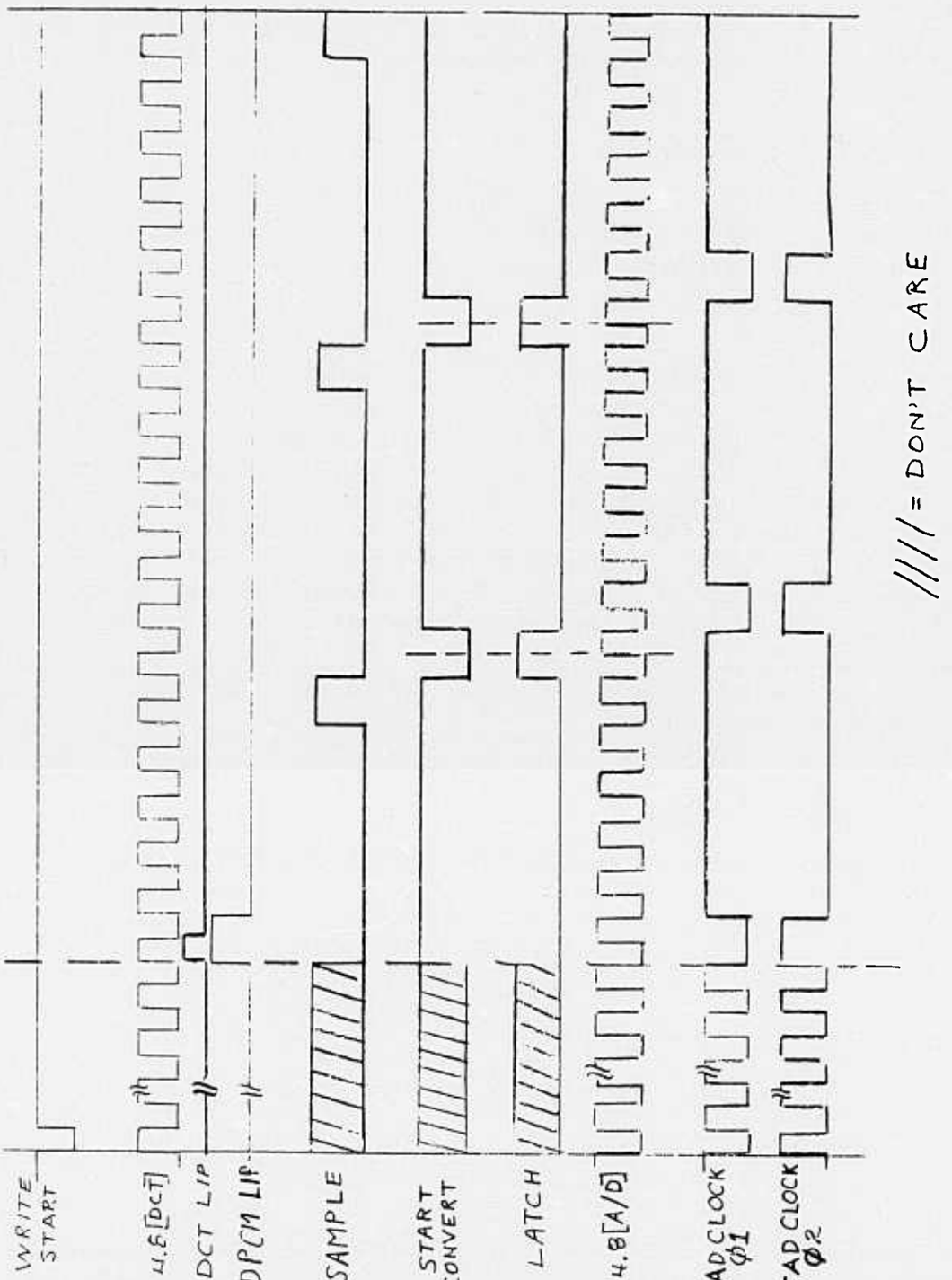


Fig. A-7. Input timing.

102

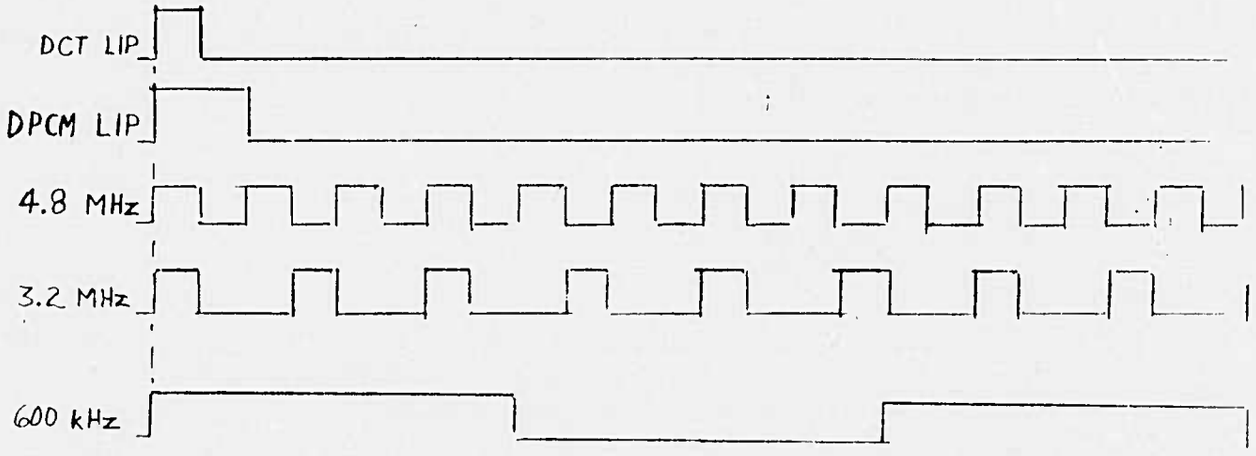


Fig. A-8. LIP/clock phasing.

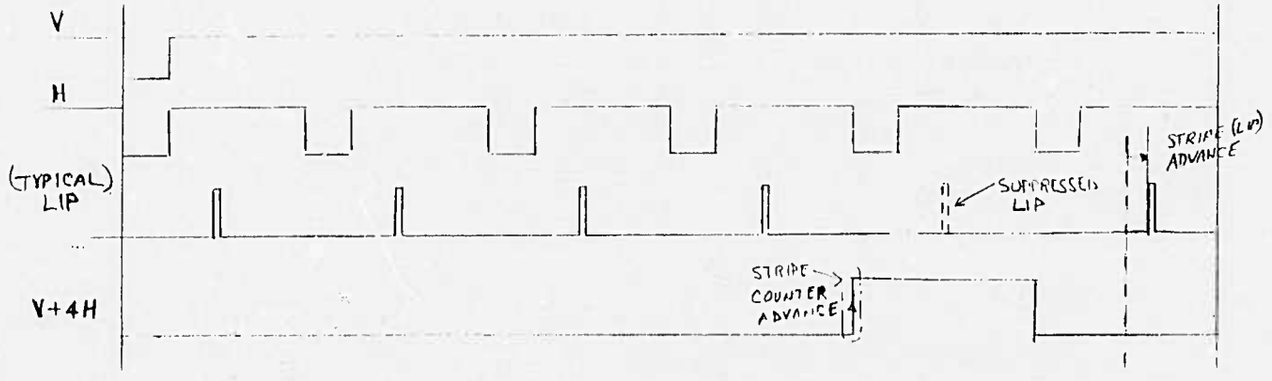


Fig. A-9. Stripe advance details.

in excess of the amount made available in the airframe, a switching regulator was designed to develop +5 V dc at 2.5 W from the +15 V dc source. The additional voltages required by the airborne unit are derived from available voltages using three-terminal IC regulators. The +15-V dc supply is used to generate +10 V dc and the -15 V dc supply to generate -12 V dc.

The total measured power drawn from the ICNS supplies is 14.85 W. Of this, about 1.5 W is dissipated in the power conditioning board, so the power required by the encoder itself is less than 13.5 W.

B. GROUND STATION

1. Input Interface

The input interface provides an inverse buffering operation relative to the airborne unit output interface. That is, it converts a serial bit stream from the modem into 6-bit encoded video coefficients for transmission to the inverse DPCM processor. Because the accumulation of serial input data extends over the complete 65-μs horizontal sync interval, it is necessary to employ two RAMS for data buffering, each performing a read or write function on alternate horizontal scans.

Figure A-10 is the block diagram of the input interface circuitry. Referring to the block diagram, Z1 through Z4 comprise the two-section data buffer RAM. Serial input data from the modem is clocked at the modem clock rate into Z10, which is organized as a serial-in, parallel-out 6-bit shift register. The six parallel shift register outputs are presented as data input to the RAM. Bit counter Z13 counts the number of bits transmitted from each coefficient, according to the stored program in input modifier prom Z16. When the proper number of bits have been received and clocked in Z19, a write strobe is generated by differentiating and gating of the terminal count of Z13 (gates Z11 and Z12). Flip-flop Z17-A controls the selection of which RAM receives the write strobe, and is toggled by LIP to alternate RAMs between reading and writing cycles. Addresses for writing data into the RAMs are generated by the word counter comprised of Z9 and Z15. This counter is reset by LIP at the beginning of each new line, along with bit counter Z13. Multiplexing of the write addresses and the corresponding read addresses generated by the inverse DPCM processor is performed by Z6-Z8, again under control of Z17A for proper read and write addressing of the data buffer RAMs.

2. Inverse DPCM

The inverse DPCM (IDPCM), Fig. A-11, is part of the ground station circuitry. It receives 6-bit parallel data words from the ground station input interface and provides output data words to the inverse discrete cosine transform (IDCT) function. Timing signals are supplied by the master timing logic of the ground station. Data rate select X and Y are supplied by the external modem interface.

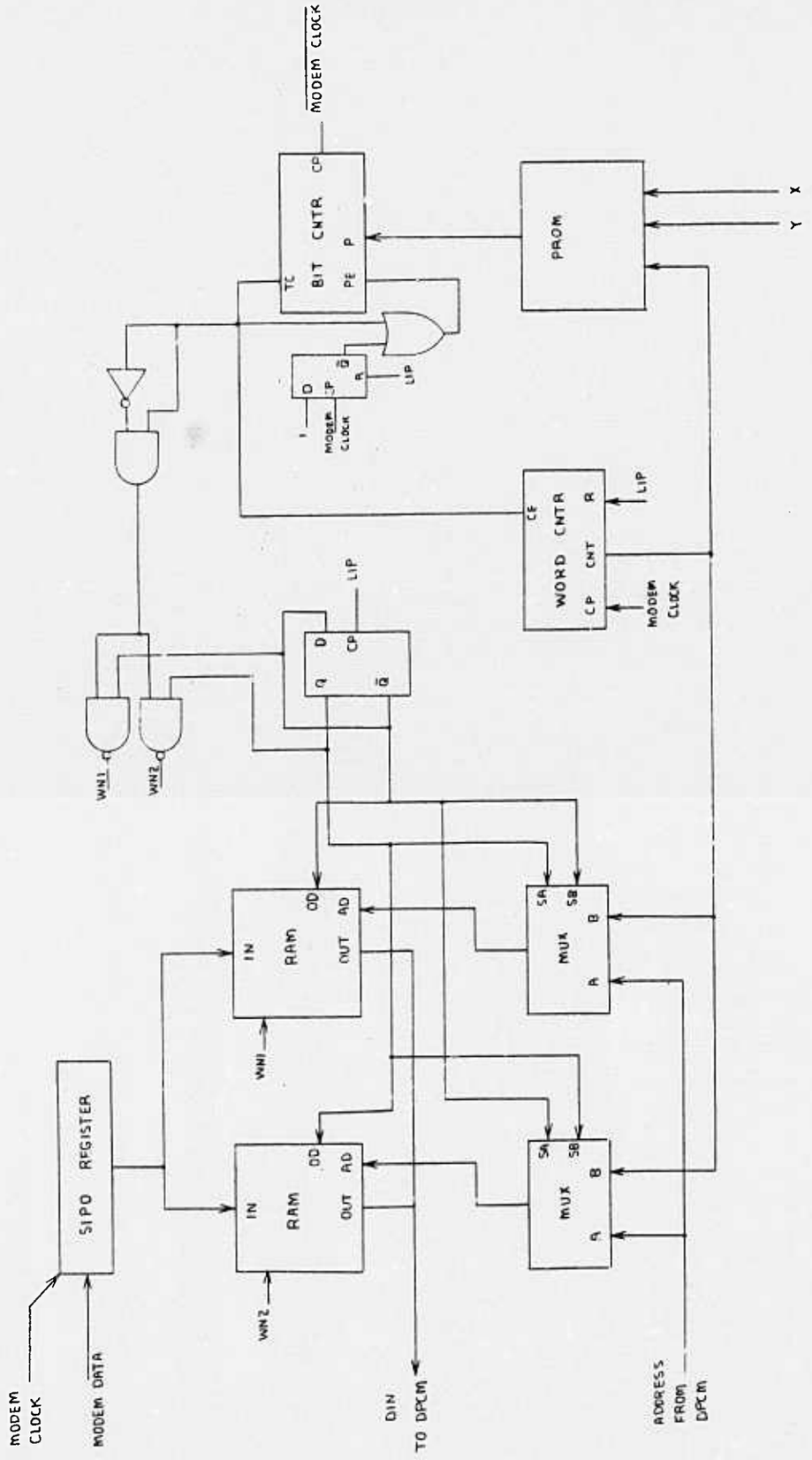


Fig. A-10. Ground station input interface.

105

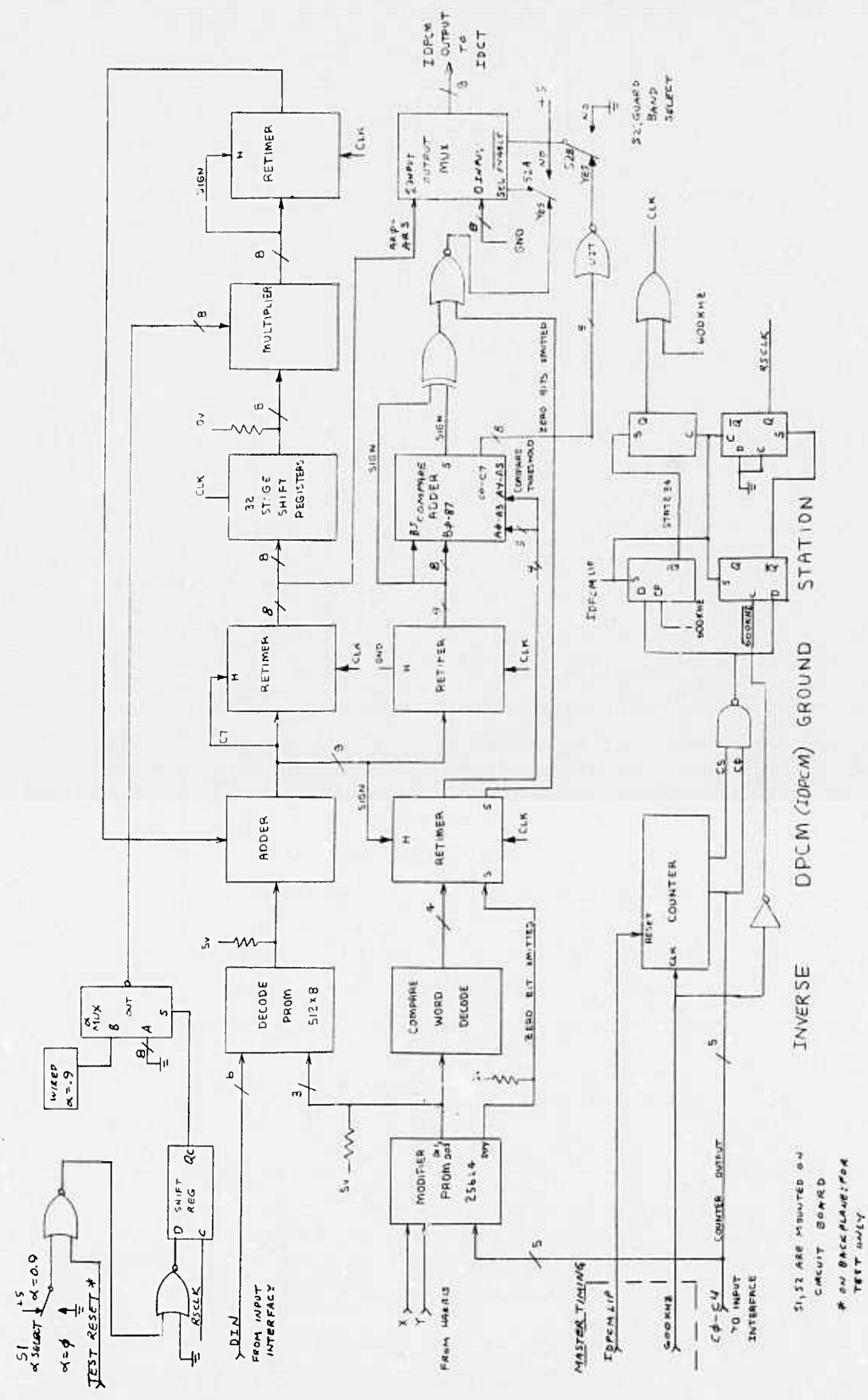


Fig. A-11. Inverse DPCM ground station.

106

Two toggle switches are mounted on the board so that the operator may control the IDPCM operating modes. A test input line is also provided so that the IDPCM may be reset with an appropriate timing waveform applied to the backplane.

Switch S1 controls the selection of attenuation constant a to be either zero or 0.90625. When S1 is in the a = zero position, the external test reset line noted above is inhibited. Switch S2 is provided to allow selection or removal of the guard band for the decoded data words. In normal system operation a = 0.90625 is selected and the guard band is enabled.

All logic, except the TMS3112 shift register, uses only the +5 V supply. The TMS3112 also requires -12 V, which is derived from the -15 V ground unit supply by a regulator chip on the board.

When connected in the normal operating mode, the IDPCM decodes the incoming data words from the input interface and forms reconstituted values of the frequency coefficients for delivery to the IDCT. Decoding is a function of magnitude and bit rate and is performed by the decode and modifier PROMs shown in the IDPCM block diagram. The main DPCM loop is completed by the path through the adder, the 32 stage shift register, and the multiplier. In normal mode the scale factor, a = 0.90625, is applied to one part of the multiplier. The decoded video coefficients are not used directly but are modified by a secondary loop which imposes a dead zone about the zero level. This loop consists of a compare word decode PROM, retimer, compare adder and associated gates. The EXCLUSIVE OR gate following the adder, in conjunction with the NOR gate, controls the SELECT input of the output MUX to force the output data to all zeros under certain predetermined conditions. NOR gate U27 ensures that the dead zone is symmetrical about the zero level by forcing all zeros via the ENABLE control input of the OUTPUT MUX. Under certain test conditions, a repetitive input waveform is used as an input to the airborne unit. Precalculated digital words will appear as IDPCM output words if the IDPCM can be reset. To do this, a is forced to zero for a given number of line intervals. At the end of the reset time, all shift-register stages will contain zeros, and computation can start from a known value. Without this reset, the shift registers will contain a random pattern of data bits which will alter the word patterns. By ensuring a fixed, known set of word values from the computations, the arithmetic and storage functions may be checked. Conditions for the electronic resetting require that S1 is in the normal (a = 0.90625) position and that an appropriate reset waveform is applied to the TEST RESET line.

3. Inverse DCT

The inverse DCT receives 8-bit modified frequency coefficients from the inverse DPCM and transforms them into 6-bit time sample words which are transferred to the frame store memory (FSM). A block diagram of the inverse DCT is shown in Fig. A-12 and A-13. Timing and control and pipeline processor functions are very similar to those of the DCT which has been discussed earlier. The only differences in block diagrams are in the frame store memory and inverse DPCM

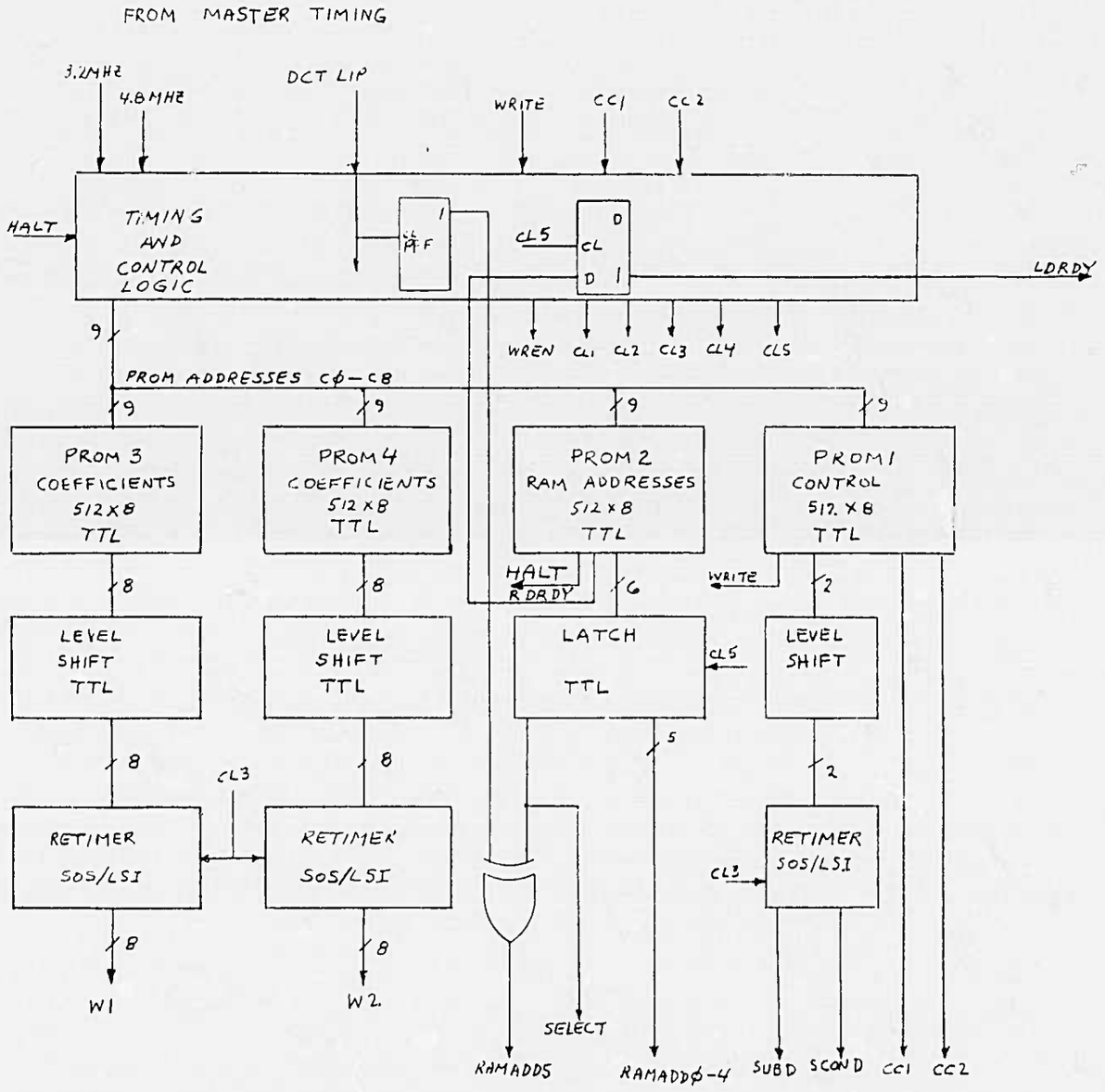


Fig. A-12. Block diagram of inverse DCT timing and control.

108

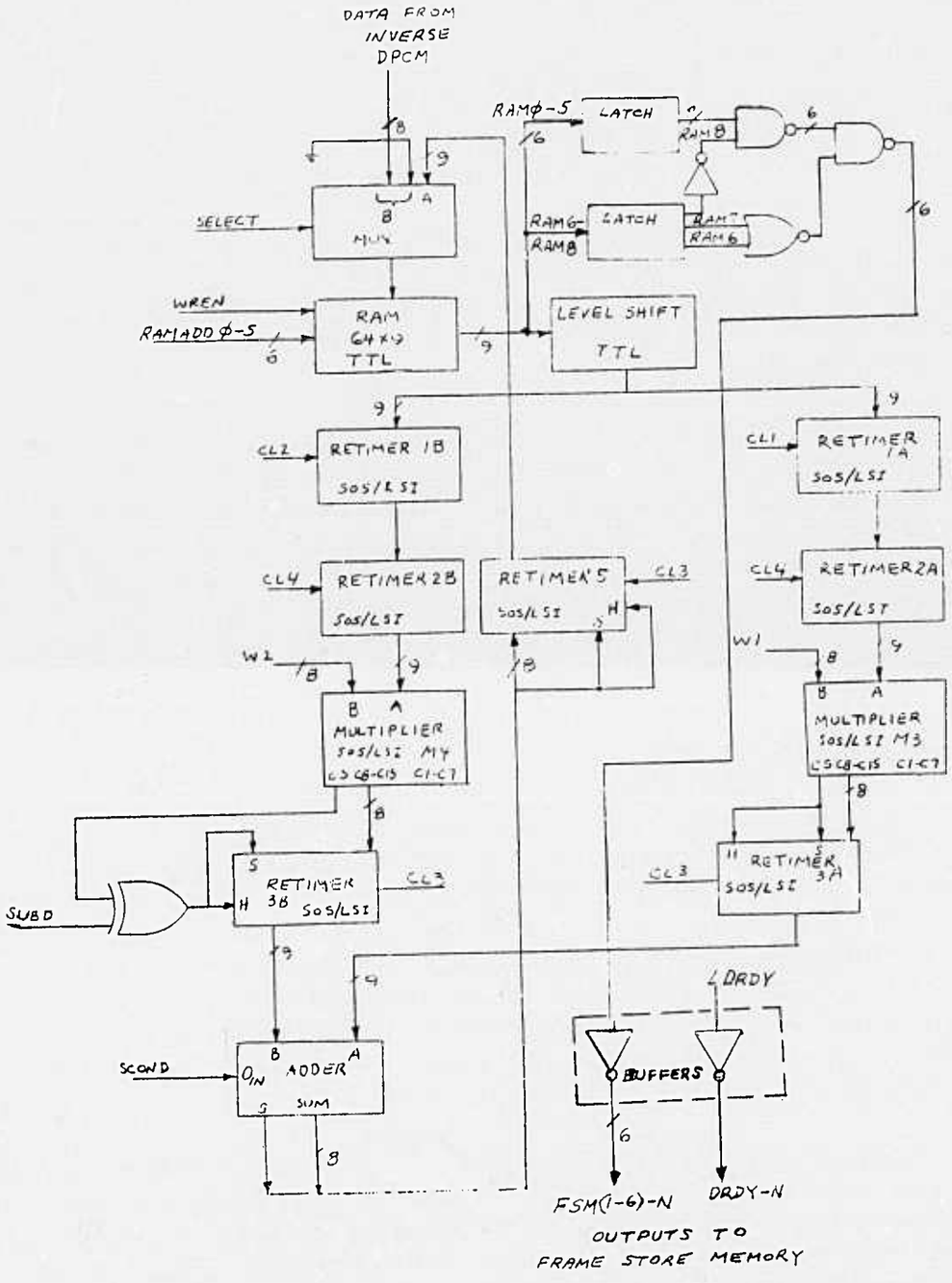


Fig. A-13. Block diagram of inverse DCT processor.

109

interfaces, which will be discussed in detail later. Although not affecting the block diagram, the programming of the four PROMs is different.

The inverse DCT function is packaged on two boards. Board A17 contains the timing logic and board A19 the inverse DCT pipeline processor.

The inverse DCT interface with the inverse DPCM is an 8-bit parallel bus. Thirty-two words are clocked into the inverse DCT at times determined by the programming of PROM 2 and the 4.8-MHz clock from the ground station master timing. Each word is loaded into the bipolar RAM at an address determined also by PROM 2.

The frame store memory interface consists of a 6-bit parallel transfer bus and a data ready (DRDY) line. Data transitions appear on FSM1-6 lines through the line-scan interval. Generation of the DRDY gate notifies the FSM that valid data will be on the FSM bus at the next 32 positive transitions of the 4.8-MHz clock occurring during the DRDY interval.

The source of FSM words is the latched output of the bipolar RAM. Two operations are performed on this output before it is sent to the FSM. First, any negative data word (RAM8=1) will be forced to all ones since it is sent to the FSM in complemented form. Secondly, when RAM6=1 or RAM7=1, any data word will be forced to all zeros, i.e., maximum magnitude. Since only positive numbers are present, they will be in sign magnitude format. The frame store memory bus and DRDY line are driven by inverting type line drivers capable of driving 10 ft of RG58/U coaxial cable.

4. Ground Station Master Timing

The ground station master timing circuitry is on board A13. Its purpose is to generate appropriate system timing signals from the input 9.6 MHz and the frame sync pulses. Because of delays inherent in the airborne output circuitry, the first bit of serial data in a frame occurs during the third modem clock period. To correct this in the ground station, the frame sync signal is delayed by two modem clock periods. It then synchronizes dividers which divide the 9.6-MHz clock to 4.8 MHz and 600 kHz, and dividers which in turn divide the 600 kHz by a factor of 39 to produce line initiate pulses. These line initiate pulses occur at horizontal line rates (65 μs) and control the start of inverse DPCM and inverse DCT operations.

Also located on board A13 is an input selection circuit. It selects, for the decoder input, either the signals from the exterior modem or from the ground station itself. All the ground station signals except modem data are generated on board A11. The ground station modem data may be either a test signal from a PROM on board A9 or modem data from an airborne unit under test in the ground station.

5. Frame Store Memory.

The Frame Store Memory is a 32K by 12 NMOS memory manufactured by Electronic Memories and Magnetics Corp. This memory stores the decoded output of the ground station and recalls it in proper timing relationship for display on the monitor. The memory is updated in increments of thirty-two 6-bit words, corresponding to one horizontal line of one vertical stripe. Such an update occurs once for each of the horizontal lines of video.

Figure A-14 is a block diagram of the control circuitry, which is located on a card in the memory enclosure. There are two TV sync ICs in the memory control. The first is triggered by frame sync after a delay. Its output can control a camera, with the delay being selected such that the camera's active video output will coincide with the sampling time of the airborne input circuitry. The second sync IC controls various counters in the control circuits.

Data is accepted from the DCT in a group of 32 words at any time while the memory is reading out to the monitor. The new information is placed in temporary storage until the memory is not busy during horizontal retrace time. It is also reorganized into 16 words, each 12 bits wide. During retrace it is written into the memory.

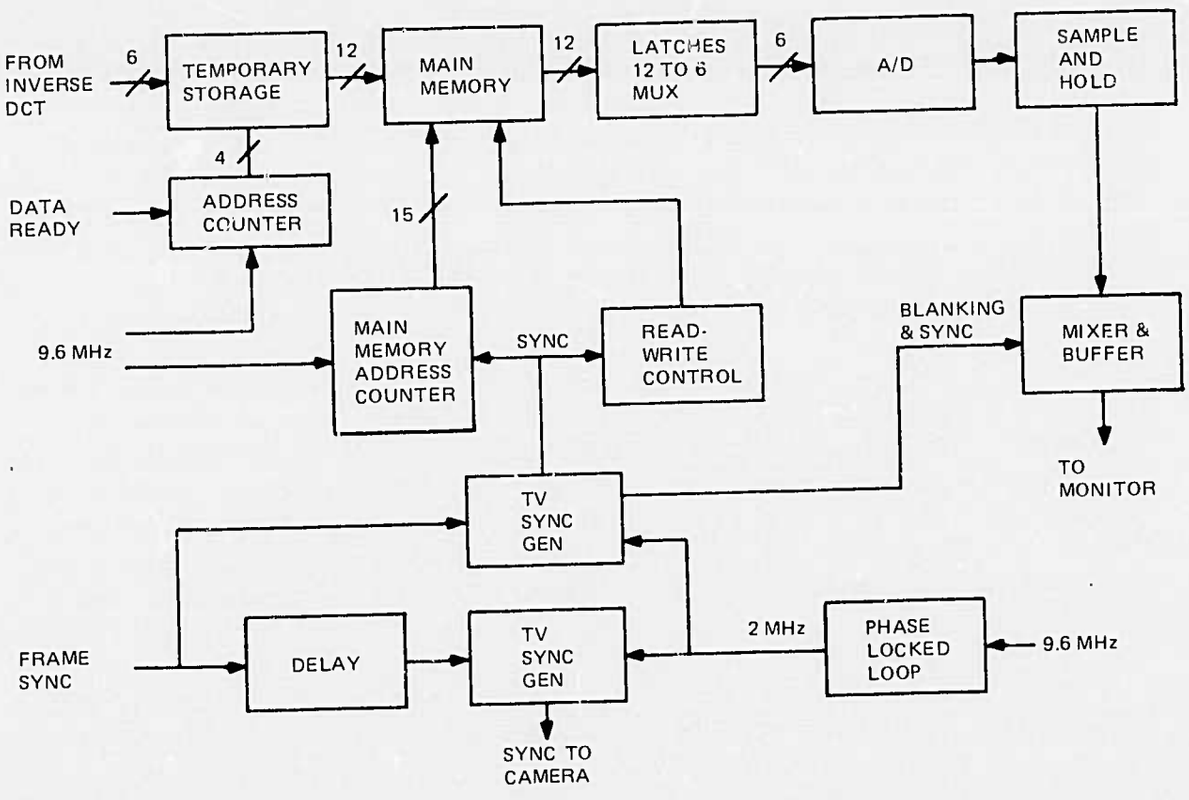


Fig. A-14. Frame store memory control.

111

The memory is read out at a 2.4-MHz rate during the time video is displayed on the monitor. The data is first latched, then multiplexed back from 12-bit-wide words at 2.4 MHz to 6-bit-wide words at 4.8 MHz. It then goes to a high-speed digital-to-analog converter. A high-speed sample- and -hold circuit following the converter eliminates glitches. A mixer circuit next combines the reconstructed video with blanking and sync pulses. The composite video signal is then buffered and sent to the output connectors.

6. Test Pattern Generator

On card A9 in the ground station are circuits for generating waveforms used in testing the system, a reset generator, and buffer devices for the sync signals to the BNC connectors on the front panel. The ground-station decoder and memory may be tested by introducing into the serial data input a known data series in lieu of the output of an airborne unit. A suitable input will produce on the monitor a pattern which can be recognized and from which the status of the system can be evaluated.

a. Serial Test Generator

The serial test generator produces a repetitive data stream 208 bits long at the modem clock rate. A different pattern is produced for each of the data rates. In the case of the 1600 kHz rate, the 208 bits comprise two alternating lines of serial data of 104 bits each; for 800 kHz, four lines of 52 bits each, etc. The bits were chosen by doing a computer simulation of the encoding of a ramp input and selecting a serial data pattern similar to that of the simulation. A DPCM system does not produce the same output each line for a fixed input. Rather, the DPCM values for a given coefficient tend to cluster about an average. For example, if a coefficient needed to be described by a DPCM value of 24, the actual DPCM words over several lines might be 22, 25, 24, 26, 23, 24, etc. Consequently, the picture produced in the monitor by the limited serial data stream bears little resemblance to a ramp function. Pictures of the presentation for each of the data rates are given in Appendix B.

A counter, clocked by the modem clock, supplies addresses to a PROM. The PROM is organized 256 X 8, but only 204 bit locations are used, because the 8-bit counter does not go through all possible states. A selector circuit extracts a 1-bit-wide data stream, depending on the bit rate. The PROM program for the 200 kHz rate produced a flat grey field on the monitor, so it is not used. Rather, the selector input for 200 kHz is kept high, producing a stream of all ones. The resulting pattern on the monitor consists of vertical bars, which are easier to evaluate than a flat grey field.

In the ground station test mode, either the aforementioned serial data (ground station test) or the output of the airborne unit in slot A-1 (airborne unit test)

is delivered to the deeoder. The output of the airborne unit may be the eneoding of video from a camera, of a test analog signal applied to the VIDEO IN port, or of a digital test funetion.

b. Digital Test Pattern Generator

The digital test pattern generator consists of a eounter eloeked by lateh pulses from the airborne unit and logie to deeode the funetion required for the digital testing. The eireuitry runs continuously, but only the first 32 bits following a slow eloek gate are used. The test funetion must drive the A/D eonverter in the airborne unit to saturation in both direetions. To do so requires a signal of 7 V amplitude, biased at about 6.5 V. The bias potentiometer may be adjusted while noting the output of the A/D eonverter to verify that it is being driven to saturation in both directions.

e. Vertieal Reset Generator

A reset generator on A9 produces for eaeh vertical sync, a pulse whose duration is between two and there horizontal line periods. This pulse is used to reset the DPCM loop at the beginning of each vertieal stripe. The shorter pulse, rather than the vertieal syne pulse itself, is used beeause the DPCM loop requires a time to recover from reset whieh is greater than the duration of vertical blanking. The reset, therefore, introduees shading at the top of the picture. Redueing the duration of the reset allows more time for reeovery before aetive video reeommences, and thereby minimizes the extent of the shading.

d. Test Reset Generator

The reset signal generator on board A11 generates an appropriately timed signal to reset the DPCM for digital system testing. This reset signal is also used to reset the inverse DPCM after being appropriately retimed on the inverse DPCM board (A15). The reset signal transition oecurs 4.5 600-kHz eloek pulses before LIP.

7. Modem Simulator

The modem simulator generator in the ground station provides loeally signals such as would be reeeived from the modem, i.e., 9.6 MHz, modem eloek, horizontal and vertieal sync, and data rate seleetion. These signals are used to simulate the modem when the ground station is operating independently of the modem, or when it is in the test mode. Figure A-15 is a bloek diagram of the modem generator. It consists of a 9.6-MHz erystal oseillator and appropriate divider and seleet eircuits. An H/2 delay eireuit delays the frame syne pulse to the eontrol panel. The frame syne pulse on the eontrol panel is intended primarily for synchronizing an SRL eamera.

The camera circuitry generates its internal horizontal sync such that frame sync falls midway during a horizontal line time. Delaying the frame sync by half a horizontal line interval causes the output video to be properly positioned with respect to airborne unit sampling time.

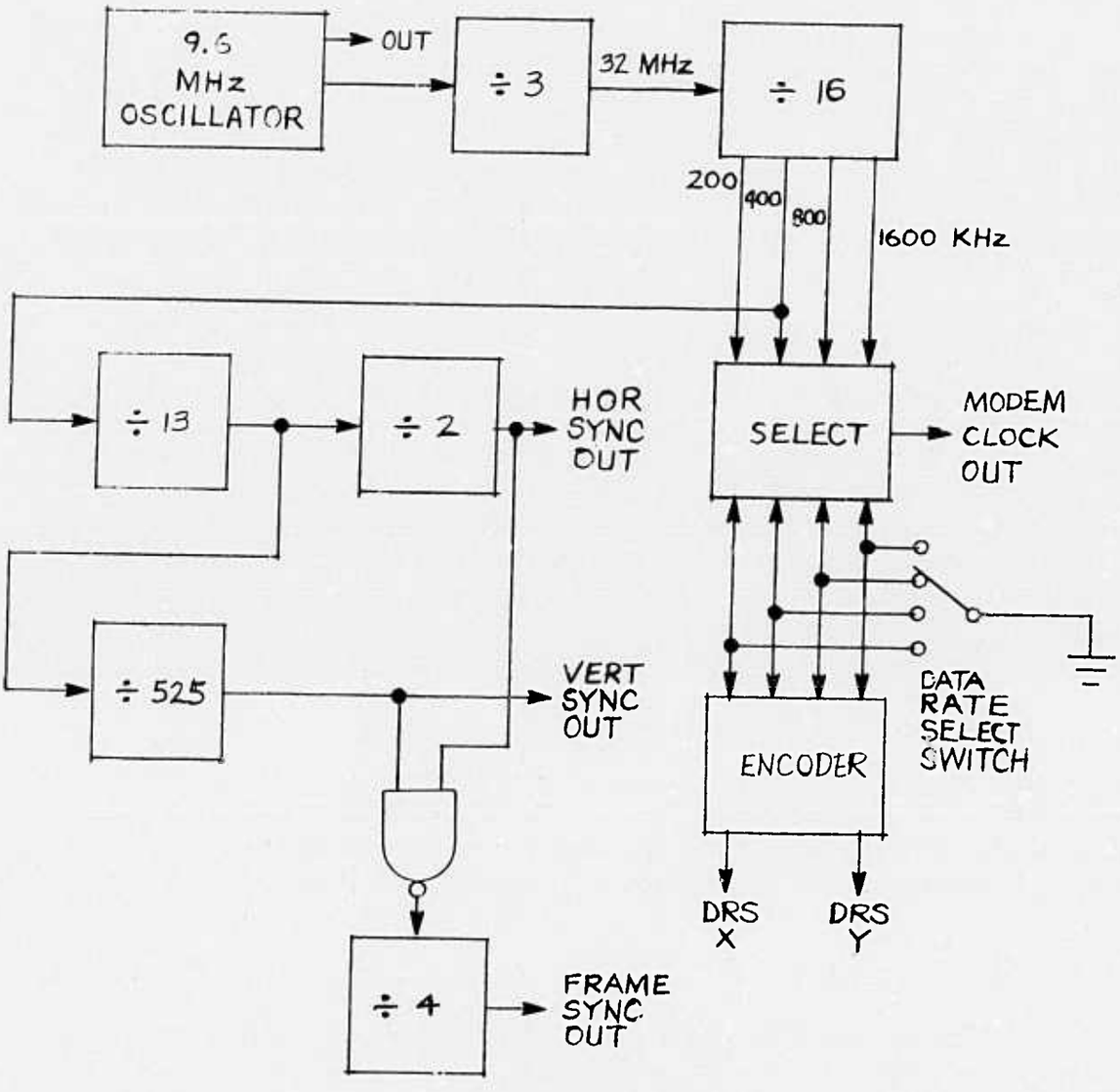


Fig. A-15. Modem generator.

114

8. Airborne Test Adapter

The airborne test adapter is a modified circuit board which fits into slot A-1 of the ground station. It holds an airborne unit for testing. A relay on the adapter withholds power from the airborne unit unless the ground station is in the TEST mode. A muffin fan cools the airborne unit's hybrids. Power for the fan is 115 V ac. Note that in the TEST mode this voltage is present on the rear connector and on the test adapter. The 115 V line has been insulated, but the user should be aware of its presence. Slot A-1 has two extra keys to prevent any other card from being inserted into this slot. Appropriate keyways are cut in the adapter and in the special extender boards which may be used in working on the airborne unit.

Located on the test adapter are a latch circuit and D/A converter. Latch pulses from the airborne unit are connected to the latch circuit. If connections are made from the output of the A/D converter on the airborne unit to the latch inputs, the D/A converter output will show how the input circuitry is sampling and digitizing the analog input. Connections to the output of the A/D converter may be made at the input to the DCT hybrid. Connections should be made as shown in the following listing:

Hybrid-Pin	to	Latch (SN74174) pin
3	-	3
2	-	4
5	-	6
4	-	11
6	-	13
7	-	14

9. Ground Station Power

DC power for the ground station is furnished by three Abbott power supplies, with outputs of +15 V, -15 V, and +5 V. The 115-V ac power for them is controlled by the power switch on the ground-station control panel. Across each output is an Abbot over-voltage protection module.

Several additional voltages required by various circuits are derived from the ±15 V. The airborne card needs -5 V which is derived by a resistor-zener combination on the airborne test adapter. The inverse DPCM needs -12 V, which is produced by a regulator on the inverse DPCM card. The inverse DCT needs +10 V, which is produced by a regulator mounted on the frame of the ground station.

Power for the memory and the memory control circuits comes from a supply furnished by EMM as an integral part of the memory assembly.

Appendix B

SYSTEM OPERATION

A. CONTROL PANEL

Figure B-1 shows the control panel of the ground station. At the lower right is the Power On-Off switch, which is in series with the 115-V power line coming into the ground station. A 3-A fuse adjacent to the switch is also in series with the ac line. Three indicator lamps monitor the outputs of the three dc power supplies in the decode nest.

The Mode switch controls the input to the decoder. In the Operate mode the signals reading the decoder are those from connectors J2 through J8. In the Operate mode the other control panel switches, except Power On-Off, have no effect on the decoder output. In the Test mode, signals generated within the ground station are applied to the decoder. In the Test mode the Test Select switch selects either a fixed test pattern (Test Ground Station) or the output of an airborne unit (Test Airborne Unit) as input to the decoder. The airborne unit is in this case mounted on a test card in slot A-1 of the ground station. In either of the Test Select options, the Bit Rate switch controls the serial data rate.

The Sync Select switch selects one of several combinations of synchronization signals to be available at BNC connectors A and B. These signals may be used to synchronize either a camera or a test generator to provide an analog input for an airborne unit under test. Such an analog input, applied to the Video In BNC connector, will be connected to the airborne unit in the test slot.

B. ADDITIONAL SYSTEM CONTROLS

Three additional switches and one potentiometer are located on boards in the ground station and may be operated without removing the boards from the nest.

On board A-9 the potentiometer adjusts the dc level of the test pattern which is injected into the airborne unit in conjunction with digital testing. This test pattern must drive the A/D converter over its range, and the potentiometer is used to adjust the signal to the proper level to do so. Also on board A-9 is the Reset Select switch, which selects reset signals applied to the inverse DPCM. When the switch is up (as in normal operation), the DPSM is reset at the beginning of each vertical stripe; when down, reset occurs every 16 horizontal lines.

Fig. B-1. Ground station control panel.

On board A-15 (the inverse DPCM board) are two switches. The upper switch, Guard Band On-Off, controls the guard band operation. When it is up, the guard band is not operating; when it is down, it is operating. In normal operation, this switch is down. The lower switch, Reset On-Off, controls the reset function. When it is up (as in normal operation), reset can take place, in the mode determined by the switch on A-9; when it is down, reset does not occur.

C. CONTROLS IN NORMAL OPERATION

In normal operation, i.e., in conjunction with an external modem, the only control panel switches of interest are the Power On-Off and Mode switches. Power On and mode Operate are required. The Reset On-Off and Guard Band On-Off may be in either position, as desired. The Reset Select should select vertical reset. On the memory chassis, power On, freeze Off, Test-Operate in Operate, and memory test card not in EX ON are required.

D. TEST CAPABILITIES

1. Ground Station Self-Test

In the ground station there is a 9.6-MHz oscillator, a set of dividers to provide horizontal, vertical and frame sync signals, a series data clock and a repetitive serial data signal. When the Mode switch is at Test and the Test Select switch is at Ground Station, the aforementioned signals are applied to the decoder inputs, at a clock rate determined by the setting of the Bit Rate switch. A pattern of eight vertical stripes is produced on the monitor. Figure B-2 shows the pattern produced at each of the four data rates. The appearance of these patterns guarantees that the decoder and memory are functioning properly.

These pictures show only a portion of the presentation. There are actually eight stripes across the screen. The display of these patterns in the ground station test mode verifies proper operation of the decoder and frame store memory.

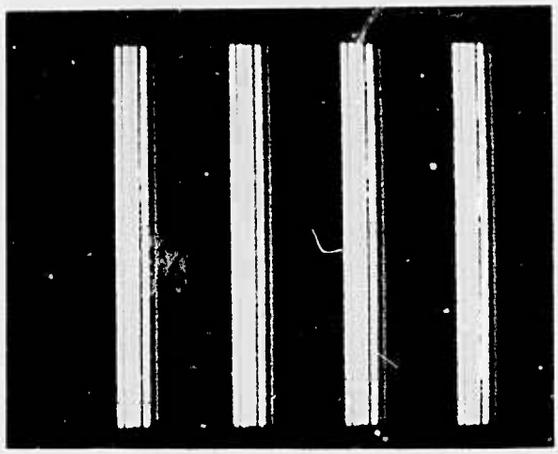
2. Airborne Unit Test

The ground station includes provision for testing airborne units. An airborne unit being tested mounts on a test adapter assembly which is plugged into position A1 of the decoder nest. When an airborne unit is so mounted and the ground station is set to Test Mode and Airborne Unit Test Select, power and appropriate timing signals are applied to the airborne unit and the serial data from the airborne unit is delivered to the input of the decoder. Analog signals at the Video In terminal on the control panel go into the video input of the airborne unit and are subsequently encoded. A test of the entire airborne-unit/ground-station system may be made by observing on the monitor both the video input to the airborne unit and the output of the ground station.

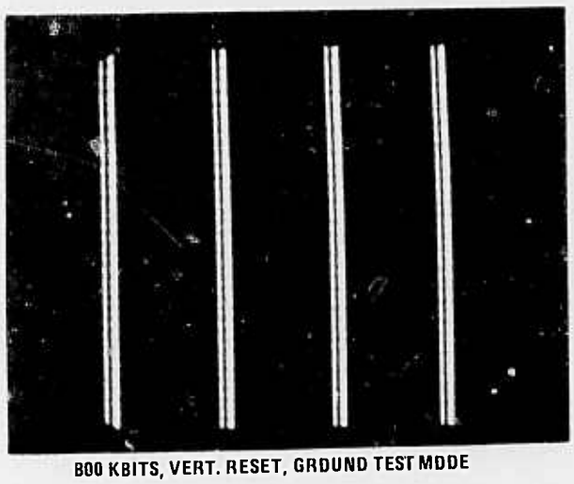
E. FRAME STORE MEMORY

On the control panel of the frame store memory chassis are an On-Off breaker switch and three toggle switches associated with the three power supplies, +15, -15, and +5 V. Each toggle switch, when pushed to the right, increases its associated voltage by 1 V in the case of the 15-V supplies and by 0.5 V in the case of the 5-V supply. When a switch is pushed to the left, a corresponding decrement occurs. These voltage-altering switches will ordinarily not be used.

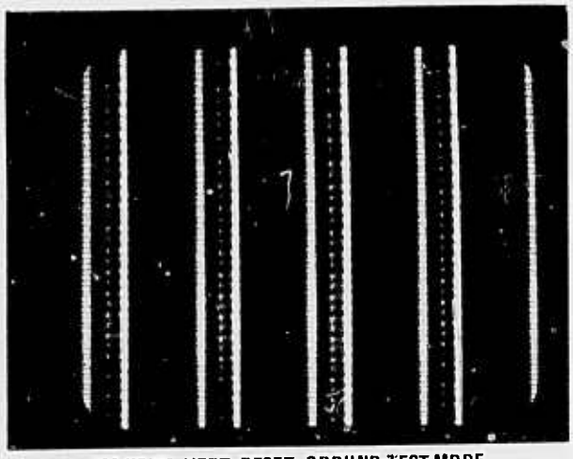
At the top of the memory chassis is a test card, which can exercise the memory's capabilities. The functions of its switches are described in the EMM manual. When it is in the EX ON mode, the usual functioning of the memory is inhibited. In normal operation, therefore, the test card must be either OFF or in DISP ON.



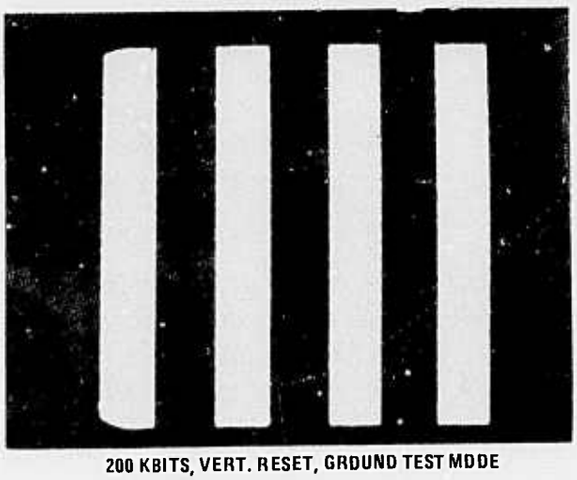
1600 KBITS, VERT. RESET, GROUND TEST MODE



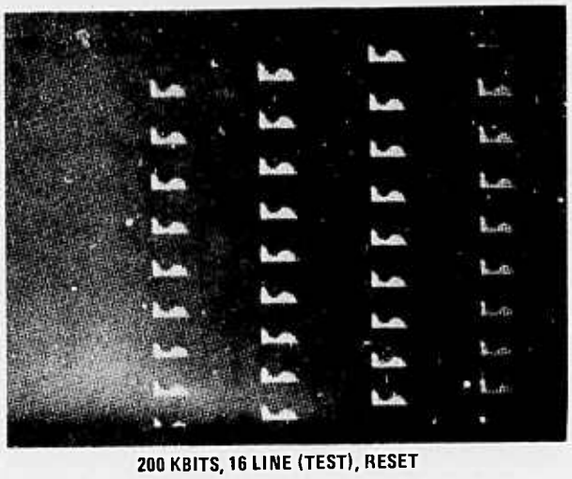
800 KBITS, VERT. RESET, GROUND TEST MODE



400 KBITS, VERT. RESET, GROUND TEST MODE



200 KBITS, VERT. RESET, GROUND TEST MODE



200 KBITS, 16 LINE (TEST), RESET

Fig. B-2. Ground station self-test displays.

On the memory control card, located at the bottom of the memory chassis, are two toggle switches and a BNC connector. These are accessible only by removal of the plastic front panel. The BNC connector provides video drive identical to that delivered to the TNC connectors on the rear of the chassis. When the toggle switch marked Test-Operate is in the Test position, the presentation on the monitor consists of 8 vertical stripes, each of which shows a graduation from white to black in 32 equal steps. In the Operate position, the monitor displays the input from the decode electronics. The other toggle switch is marked FREEZE-GO. In the GO position, operation is normal. In the FREEZE position, the memory write function is inhibited, and the memory reads out repetitively whatever it contains. In practice, with live video, this implies that the picture is frozen on the screen.

Also located on the memory control card are three potentiometers. The one on the left controls the output video amplitude, the one in the center controls the blanking pedestal amplitude, and the one on the right controls the sync pulse amplitude.

F. SYNCHRONIZATION SIGNALS

Three sets of synchronization signals are available at the BNC terminals of the control panel.

The first set consists of interlaced horizontal and vertical sync pulses derived from a TV sync generator IC located in the memory unit. These are so phased that a camera synchronized to them will provide active video at the proper time for processing by an airborne unit in the test adapter. These signals are not at the usual broadcast TV rates, however, but are at the reduced rates required by this system (i.e., 65 μs between horizontal sync pulses rather than the 63.5-μs RS170 interval). The vertical sync and composite blanking signals are scaled in the same ratio.

The second set of signals consists of video gate and composite blanking. Video gate is a pulse whose duration extends for 32 cycles of the 4.8-MHz clock. It coincides with the time when the analog input circuitry of the airborne unit is sampling the input video and storing the samples for processing. It is brought out for use in synchronizing a generator if test analog signals are required. The corresponding signal, composite blanking, is obtained from the TV sync generator IC which supplies the horizontal and vertical signals mentioned above. The composite blanking signal is not used at present.

The third set of signals, frame sync and 2.0 MHz, is provided to operate an SRL camera such as is used in the Aquila system. Frame sync is a 1.6-μs negative-going pulse, which occurs at the start of each 8th television field in the center of the first horizontal line. The 2.0 MHz is locked to the 9.6-MHz system clock via a phase-locked loop.

APPENDIX B: COMMAND LANGUAGE SUMMARY

21 December 1977

by
EH Wrench Jr
NOSC

The following is a descriptionof an interactive, on-line command language for manipulation of vector data images in particular. The program is implemented on a PDP 11/04 computer system that includes two frame-store memories, each capable of storing and displaying a 256 X 256 X 8 bit image. The data in the image are accessible to the computer via the PDP 11 UNIBUS. The memories can also be loaded directly from a TV camera attached to the UNIBUS or from mag tape or diskette. The system is also equipped with a card containing a hardware multiply, a cosine look-up table, and a bit swap register to increase the speed of a software fast Fourier transform (FFT) program. With this hardware, a 256-point cosine transform can be computed in approximately 250 ms and an entire image can be transformed in approximately 1 minute.

The command language consists of a large number of commands or "operators" that operate on the ACCUMULATOR, a 16-bit fixed-point vector whose length is variable from 1 to 256 points. The vector length is the "CONTEX" and is set by the operator. Operators that require two operands, such as add and subtract, always use the accumulator as one operand, and the results are stored in the accumulator. The second operand is specified in the command and can be one of eight working registers, A–H, or a line from either memory. A memory line is specified by typing M0 or M1 followed by "(n, a)" where n is the line number and a is the vector number within the line. For example, if the context is set to 32, then M1 (10, 3) will specify line 10 of memory 1, the third 32-point vector (points 64–95). If "a" is not specified, the first vector is assumed.

The program has the capability of repeating a set of commands a fixed number of times through the use of "DO" loops similar to those used in FORTRAN. Strings of commands can also be stored on disk and called as "canned" routines by use of a "RUN" command. The system is intended to be used for hardware signal processing simulation, and it therefore uses fixed-point integer arithmetic. All scaling and overflow correction (and detection) must be done by the user.

The operators currently available are described in detail in the next section.

SINGLE OPERAND INSTRUCTIONS

These commands are of the form "CMD".

CLR	Clears the Acc
ID	Loads each point in the Acc with its index number, ie, point zero = 0, point one = 1, etc.
SQ	Squares each point in the Acc
SQRT	Square root of each point in the accumulator
NEG	Negates the Acc
ABS	Absolute value of the Acc

LS	Circular shift left the Acc
RS	Circular shift right the Acc
INV	Calculates the reciprocal of the Acc
DCT	Calculates the even discrete cosine transform of the Acc
IDCT	Calculates the inverse cosine transform of the Acc
COS	Calculates the cosine of the Acc
SIN	Calculates the sine of the Acc
FFT	Calculates the Fourier transform of the Acc (real part) and the H reg (imaginary part)
IFFT	Calculates the inverse Fourier transform of the Acc (real part) and the H reg (imaginary part)
QUANT	Quantizes the Acc using one of eight tables stored in the file DXO: QUANT TBL (The contents of the H reg determine which table to use for each point in the Acc)
HALF k	Divides the Acc by 2 ** k

COMMANDS WITH TWO OPERANDS

These commands have the form "CMD X" where X can be one of the following: 1) An 8-bit integer number. 2) Register A, B, C. . . ., H. 3) M0 (n, a) or M1 (n, a), where n is the line number and a is an optional vector number within the line.

LOAD X	Loads the Acc with contents of X
STORE X	Stores the Acc in X
+ X	Adds X to the Acc
- X	Subtracts X from the Acc
* X	Multiplies the Acc by X
**X	Multiplies the Acc by X and saves the upper 16 bits

COMMANDS INVOLVING THE INDEX REGISTERS I-N

These commands operate on one of the index registers I-N. They have the form of. . . . CMD I . . . where I represents any of the 4 index registers.

SET I=n	Sets index reg I to value n
DEC I	Decrements register I
INC I	Increments register I
EVAL I	Prints contents of index register I

COMMANDS INVOLVING MEMORY

These commands are of the form "CMD M0" or "CMD M1".

ERASE Mn	Sets all points in the specified frame-store memory to zero

VIEW Mn	Displays memory Mn on the monitor
LDISC Mn(k)	Loads memory Mn with the kth image on DX1 (k=0, 1, 2)
SDISC Mn(k)	Stores memory Mn on DX1, the kth image (k=0, 1, 2)
DISPLAY Mn	Displays the Acc as an X, Y plot scaled to make the largest Acc value greater than 64 and less than 128
DISPLAY. Mn	Displays the Acc as above, but uses the scale from the previous DISPLAY
UP	Displays the Acc with the scale increased by 2
DOWN	Displays the Acc with the scale decreased by 2
TRANS-POSE Mn	Transposes the image in memory Mn
RECORD Mn	Takes samples from A/D converter 0 at a 10 kHz rate, and fills memory Mn
PLAY Mn	Outputs samples from memory Mn to the D/A converter 0 at a 10 kHz rate
HISTO Mn	Calculates the amplitude histogram of memory Mn and stores results in the ACC
SIGMA Mn	Calculates the variance and mean of each column in memory Mn (Acc contains variance, H reg contains mean)
MSE Mn(k)	Calculates the mean square error between Mn and the kth image stored on DX1, scaled by the image energy
MSE. Mn(k)	Calculates the mean square error as above, but scales by the max image energy

DO LOOPS

Strings of command are repeated by the "DO" command. The format for the command is "DO n CMD1 CMD2 . . . END". The list of commands CMD1, CMD2, . . . are executed "n" times. Multiple DO loops may be nested to 16 levels. For example, the command sequence " set I=0 DO 10 EVAL I DO 20 INC I END END" will print out 10 values of i (0, 20, 40, . . . , 200).

STORED COMMAND LISTS

Sequences of commands can be stored on disk and run without retyping the commands. This is done with the "LIST" command. The form is "LIST CMD1 CMD2 CMD3 . . . . After the list is entered, the "SAVE" command stores the list on disk. The system will ask for a 1-6 letter file name for the list, and it will be stored on DX0 with a file extension of . CMD. The stored list of commands may then be executed by entering the command "RUN filename".

APPENDIX C: FRAME-STORE MEMORY EXECUTIVE PROGRAM

by
AG Weber
NOSC

The following is a description of the frame-store memory (FSM) executive program implemented on the PDP 11/04 image processing system at NOSC.

The purpose of the FSM executive program is to facilitate bulk data manipulations both within the FSMs and between them and other mass-storage peripherals in the system. The simple architecture of the program also makes it advantageous to use it as a test program for developing new programs that involve using the FSMs.

The EXEC program is designed to be simple to learn and use. In most cases, all the operator needs to do is type in the command for the desired operation. The commands consist of two-letter mnemonics for the operation. For example: CI = clear image. Since the program looks only at the last two letters typed in, any input mistake can be corrected by simply proceeding to type in the correct letters. If the letters are not recognized as a valid command, the operator is informed of this and then prompted for a new command again. If the operator cannot remember the correct command, typing in a question will print out a list of the commands and their resulting operations. If more information is needed after the program recognizes a valid command, (eg, clear image in which FSM?), the program will ask for it.

The commands fall into three areas: (1) FSM commands that manipulate the data in one or both FSMs. (2) Transfer commands for moving data between the FSMs, the 9-track tape drive, and the floppy discs. (3) Tape commands for positioning the magnetic tape to the desired location.

Since there are currently two FSMs in the system, nearly all commands in the first two groups will ask the operator to select which of the two FSMs the operation is to be performed on. The following is a description of the commands available in the current version of EXEC.

FSM COMMANDS

CI (clear image):	Clears the selected FSM (0 or 1) to all zeros, which appears as totally black when viewed.
CM (copy FSM to FSM):	Copies the content of one FSM into the other. The operator is asked to enter the number of the destination FSM.
DM (disable FSM):	Disables the selected FSM from being read from or written to.
EM (enable FSM):	Enables the selected FSM for reading and writing.
GP (get picture):	Causes the selected FSM to be loaded from the camera. This will continue at a rate of about one picture per second until the operator presses the return key, at which point the last image captured will be left in the memory.

GS (gray scale): Puts a gray seale along either the top or the left side of the image, depending on whether the memory is in the stripe mode or line mode. The seale runs from 0 to 255 (blaek to white), thus covering all possible displayable intensities.

HG (histogram calculation): Calculates an intensity histogram of the selected FSM and plots the results in the other FSM. The plot is a graph of the number of pixels with a certain intensity versus the intensity.

MS (mean square error ealculation): Caleulates the mean square error in dB between the image in the selected FSM and a reference picture stored on floppy disc. The operator is asked to speeify which floppy image (0, 1, 2) is to be used as the reference.

MT (memory test): Tests the selected FSM for bad areas in the memory. Pixels whieh pass the test are left white, while ones whieh fail are left black on the sereen. The operator seleets how many times to try each loeation that seems to be bad before giving up and going on to the next.

QI (quantize image): Quantizes the selected image to a eertain number of diserete levels. The operator seleets the number of levels by entering the number of bits to save in eaeh pixel. For example, to quantize to 16 levels, enter 4.

SM (set FSM modes): This eommand sets the mode bits in the FSM eontrol-status register. The desired mode value is entered by the operator.

0 = stripe mode/viewed
1 = stripe mode/blanked
2 = line mode/viewed
3 = line mode/blanked

TI (transpose image): The selceted image is transposed in the memory. It is rotated around the diagonal from the upper left to the lower right.

VI (view image): Depending on the operator's response to the information requested, one, both, or neither of the FSMs is put in the viewing modes.

0 = view FSM 0
1 = view FSM 1
2 = view both
3 = vicw neither

TRANSFER COMMANDS

LD (load image from floppy dise): The selected FSM is loaded from the floppy in drive 1. The operator enters the number of the desired floppy image (0, 1, or 2).

LI (load image (line format)):	The selected FSM is loaded from the tape drive. The image file should contain 256 records, each with 256 bytes (one byte = one pixel). After the 256th record, there should be an end-of-file mark.
LF (load image fast (single record)):	The selected FSM is loaded from the tape drive. The image file should contain one record of 65 536 bytes (one byte per pixel) followed by an end-of-file mark.
LN (load image (NOSC format)):	The selected FSM is loaded from the tape drive. The image file should contain 256 records of 288 bytes each. (Nine bytes for eight pixels). This is compatible with UNIVAC 1100 tapes written in the 8-bit packed mode, where each 36-bit word contains 4 pixels in the packing method used by the COMTAL staff.
TD (tape-to-diskette transfer):	The selected floppy image is loaded from the tape drive.
UD (unload image to floppy disc):	The selected FSM is written onto the floppy image indicated by the operator. If the floppy in drive 1 is not initialized for images, the operator is informed and no write takes place. The operator has the option of canceling the operation or having the disc initialized and then written.
UI (unload image (line model)):	The selected FSM is written onto tape, followed by an end-of-file mark. For a description of the format, see LI.
UF (unload image fast (single record)):	The selected FSM is written onto tape, followed by an end-of-file mark. For a description of the format, see LF.
UN (unload image (NOSC format)):	The selected FSM is written onto tape, followed by an end-of-file mark. For a description of the format, see LN.

TAPE COMMANDS

DP (designate tape position):	Used to inform EXEC of the file number where the tape is currently positioned. (The file number must be octal.) Useful if entering EXEC when the tape is not in the rewound position.
EF (end of file):	Writes an end-of-file mark on tape.
MF (move past file marks):	Asks the operator for the number of file marks to cross. If number entered is positive, the move is forward; if negative, the move is backward. The number entered must be in octal.
PT (position tape):	Asks operator for the file number of the file to be found, then positions the tape at the beginning of that file. The file number must be in octal.
RP (read tape position):	Outputs the current tape position (file number in octal).
TR (tape rewind):	Rewinds tape to beginning of tape.

MISCELLANEOUS COMMANDS

EX (exit):	Exits EXEC and returns to RT-11 monitor.
OD (online debugging):	Jumps into the ODT program for doing debugging. To return to EXEC, issue ODT command 1000;G.
? (executive command summary):	Prints out a list of all the EXEC commands and a short description of each.

APPENDIX D: DERIVATION OF THE FAST COSINE TRANSFORM

by
FH Wrench
NOSC

$$C_K \equiv e^{-j2\pi\,(.5)\,k/2N} \sum_{n=0}^{2N-1} X_n e^{-j2\pi nk/2N} ,$$

where X_ns are the symmetrized data,

$X_0, X_1, X_2, \ldots, X_{N-1}, X_{N-1}, \ldots, X_1, X_0$. All Xs are real.

Note: $X_n = X_{2N-n-1}$.

$$C_K = e^{-j\pi k/2N} \left\{ \sum_{n=0}^{N-1} X_{2n}\, e^{-j2\pi nk/N} + \sum_{n=0}^{N-1} X_{2n+1}\, e^{-j2\pi(2n+1)k/2N} \right\},$$

but $X_{2n+1} = X_{2N-2n-2}$

$= X_{2\,(N-n-1)}$.

Let $s_n = X_{2n}$ for n=0, 1, 2. . .N–1.

Then $\{s_n\} = X_0, X_2, X_4 \ldots X_{N-2}, X_{N-1}, X_{n-3}, \ldots, X_3, X_1$.

Defining $\sum_{n=0}^{N-1} s_n\, e^{-j2\pi nk/N} = S_K$ the DFT of s_n ,

then $$C_k = e^{-j\pi k/2N} \left\{ S_k + \sum_{n=0}^{N-1} X_{2(N-n-1)}\, e^{-j2\pi(2n+1)k/2N} \right\} .$$

Let $n' = N-n-1$

or $n = N-n'-1$.

Then $$C_k = e^{-j\pi k/2N} \left\{ S_k + \sum_{n'=0}^{N-1} X_{2n'}\, e^{-j2\pi(2N-2n'-2+1)k/2N} \right\}$$

$$= e^{-j\pi k/2N} \left\{ S_k + e^{+j\pi k/N} \sum_{n'=0}^{N-1} s_{n'}\, e^{j2\pi n'k/N} \right\}$$

$$= e^{-j\pi k/2N}\ S_k + e^{+j\pi k/2N}\ S_k^{\star}$$

$$= 2\ \mathrm{Re}\left\{ e^{-j\pi k/2N}\ S_k \right\}.$$

Also,

$$C_{N-k} = 2\ \mathrm{Re}\left\{ e^{-j\pi(N-k)/2N} \sum_{n=0}^{N-1} s_n\ e^{-j2\pi n(N-k)/N} \right\}$$

$$= 2\mathrm{Re}\left\{ e^{-j\pi(N-k)/2N} \sum_{n=0}^{N-1} s_n\ e^{j2\pi nk/N} \right\}$$

$$= 2\mathrm{Im}\left\{ (e^{-j\pi k/2N})^{\star}\ S_k^{\star} \right\}.$$

Lemma: $\mathrm{Im}\,(A^{\star}B^{\star}) = -A_R B_I - A_I B_R = -\mathrm{Im}(AB)$,

then

$$C_{N-k} = -2\ \mathrm{Im}\left\{ e^{-j\pi k/2N}\ S_k \right\}.$$

Therefore the cosine transform of X_n $n=0, \ldots, N-1$ is

$$C_k = 2\ \mathrm{Re}\left\{ e^{-j\pi k/2N}\ S_k \right\} \quad k=0, 1, \ldots, N-1$$

$$C_{N-k} = -2\ \mathrm{Im}\left\{ e^{-j\pi k/2N} S_k \right\} \quad k=0, 1, \ldots, N-1,$$

where $$S_k = \sum_{n=0}^{N-1} s_n\ e^{-j2\pi nk/N},$$

$$s_n = X_{2n} \quad n=0, 1, \ldots, N/2-1,$$

and

$$s_{n+N/2} = X_{N-1-2n} \qquad N = N/2, \ldots, N-1.$$

APPENDIX E: STUDY OF HUMAN FACTORS INVOLVED WITH VIEWING COMPRESSED IMAGES FROM REMOTE PILOTED VEHICLES

Final Report
Contract N00953-77-C-0012

8 December 1977

by
ML Mout and RK Jellison
Decision Science, Inc
San Diego CA 92117

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

SUMMARY

INTRODUCTION

BACKGROUND

- Previous Work Summary
- Previous Work Conclusions

PRESENT EFFORT

- Discussion
- The Experiment
  - Filming
  - Editing and Recording
  - Target Selection
  - Scene Grouping
- Experimental Setup
  - Hardware
  - Software
- Subjects
- Subject Tasks
- Data Collection
- Measurements
- Analysis

TABLE OF CONTENTS (Continued)

Results

Subjects and Scenes

Bandwidth Reduction

Conclusions

Recommendation

Figures

Tables

REFERENCES

APPENDIX A--COMPUTER PROGRAM

APPENDIX B

B1--SUBJECT QUESTIONNAIRE

B2--INSTRUCTIONS TO SUBJECTS

LIST OF FIGURES AND TABLES

Figure 1	Sample Pictures
Figure 2	Bandwidth Reduction Experiment Setup
Figure 3	Computer Program Flow Chart
Figure 3A	TV Screen Coordinates
Figure 4	Bit Rate Reduction Effect
Table 1	Sample of Summary Printout
Table 2	Data Printout
Table 3	Scene and Run Order
Table 4	Analysis of Variance Tables, Subject Differences
Table 5	Subject Differences, Individual Effects
Table 6	Analysis of Variance Tables, Scene Differences
Table 7	Ordered Scene Group Effects
Table 8	Analysis of Variance Tables, Bandwidth Reduction
Table 9	Ordered Bandwidth Reduction Level Effects

SUMMARY

This report references a bandwidth reduction technique for real-time transmission of video images. An experiment was performed to evaluate the effect of this technique on the performance of the human observer. This technique separates the video image into eight vertical stripes. (One stripe is taken from each successive TV field). Each stripe is further compressed by means of transform techniques. Each transformed stripe is then transmitted sequentially, and the total picture is reconstructed at the receiving site. Each segment is updated as new information is received. The result is a mosaic of the scene with adjacent segments being updated consecutively, thus, generating a flickering effect as the segments are updated across the screen.

An experiment was designed to determine the degree to which such image distortion might degrade the performance of the human operator in a tactical reconnaissance situation. A series of helicopter flights were made over Camp Pendleton during Marine land-based exercises. During these flights, 16 mm color film was taken. The films represented a tactical situation as viewed from a slow, low flying aircraft. The films were edited and reduced to 26 short, separate scenes. These scenes were transformed, via the bandwidth reduction, and recorded on video tape for use in the experiment. In the experiment, these scenes were presented to military photointerpreters who were instructed to perform target

detection. The subjects were prebriefed regarding the scoring techniques and the experimental design. Speed and accuracy in target detection performance were emphasized. The subjects were monitored by an eye-tracking device to determine their targeting accuracy and their scanning techniques. This eye-tracking information was also used to detect any effects of the reduction techniques on scanning behavior. The target detection efficiency was measured in terms of time required to find the target, percent of correct targets, and number of false targets.

Analysis of the results of this experiment reveals a significant decrease in percent of correct targets under the reduction techniques. No other variables were significantly affected by the techniques.

In fact, the eye-movement measures and the time to find target measures are very consistent over all of the bit and error rates. The number of false targets was effected by the techniques but not significantly. The implication of the nonsignificance of the "time-to-find-target" measure becomes apparent when this result is taken in conjunction with the significance of the percent correct scores. If the target is easy to find, it is found quickly regardless of the data rate and bit error probability. If it is difficult to find and cannot be found in some nominal time, it is not found at all.

Significantly different eye-scan behavior and targeting was demonstrated by the subjects. The scene groups significantly affected the time to find the targets and percent correct.

Overall, the data rate and error probabilities examined do significantly degrade performance; thus, at a cost of reduction in target detection of 20% to 25% (using the original film 84% of the targets were detected, an average of 65% of the targets were detected with the 400 kbits per second data rate), a reduction in data rate to 400 kbits per second can be achieved. At this lower rate, 65% of the available targets are still detected.

INTRODUCTION

Reduction in the data rate required for transmission of images from a remote sensor is desirable for a number of reasons. These include reducing the possibility of enemy detection and hampering the use of electronic countermeasures. Many different types of bandwidth reduction techniques have been examined. Most of these involved various ways of reducing the amount of information transmitted for the construction of each picture element (pixel). These methods range from simply limiting the number of different levels of gray per pixel to calculating transforms of the picture and transmitting the coefficients which make the transform specific to the image in question. The particular method investigated included a different approach. Along with compressing the amount of information from a given picture, the image transmission is expanded over time. Thus, the viewed picture shows sectioning in accordance with the frame-by-frame transmission, these ranging from the present view back to the scene 7/60 of a second ago. Alternative time sequencing and image refreshment procedures are possible. An advantage of this technique is that it can be incorporated with other bandwidth compression techniques as referenced above; that is, the picture section being transmitted (the slice) is also compressed with respect to data transmitted per pixel. A sample of this compression can be seen in Figure 1.

The problem was to determine the degree to which these reduction processes might interfere with or degrade the performance of the human operator as he used this imagery in an operational setting and how this slicing effect interacts with other image degradation effects.

This will help to select the tradeoff between image quality and resistance to jamming.

BACKGROUND

Previous Work Summary

Decision Science, Inc. was previously under contract to Naval Undersea Center, San Diego, to perform similar work to that completed in this contract. This work was authorized under Contract No N66001-75-C-0124 and was completed in October, 1975. This work is reported in Reference 1.

Under this earlier contract an attempt was made to write a digital program to synthesize the slicing method. This attempt was not successful due to hardware limitation in the data transmission lines which restricted the response time to the video display. At that time, the hardware for the reduction technique was not yet operational, and an experiment to test the effects of the slicing was devised. This experiment utilized a static approach by modifying tactical and strategic reconnaissance photographs to reflect the slicing effect.

Military photointerpreter subjects viewed these modified photographs and were scored on their ability to find the targets. An eye-tracking device was utilized in this experiment, time to detect targets was scored, and percent of targets correctly detected was calculated. A number of the experimental methods and analysis techniques used in this experiment have been directly applicable or easily adaptable to the present work.

Previous Work Conclusions

The primary conclusion drawn from the previous experiment was that the slicing technique did not have any effect on the subjects' performance in this static situation. There was a significant difference in performance between subjects and between the different picture types presented, but this was as expected. In particular, the more complicated strategic pictures contributed to significantly poorer performance than the simpler tactical pictures.

The work under this contract led to the present contract. It was determined that since the transforms will be used in dynamic situations, an experiment should be conducted under dynamic simulated real-time conditions.

PRESENT EFFORT

Discussion

The object of the present effort was to expand the work from the previous contract to a more realistic presentation. To accomplish this, realistic, dynamic, tactical scenes were needed which could be presented to the subjects on a video display. To prevent bias of the experimental results, these scenes had to be presented in such a way as to produce consistent scene groups.

A presentation technique had to be devised along with scoring procedures and data recording. This involved hardware and software design, compatible with both experimental requirements and budget limitations.

The Experiment

Filming

The motion picture footage used in this experiment was filmed over Camp Pendleton Marine Base from a USMC UH-1 helicopter. Two helicopter photo missions were scheduled and flown in conjunction with this experiment, the first on 16 February 1977 and the second on 14 April 1977. Since permission for civilians to fly in Marine Corps helicopters was not received prior to the first mission, this flight was necessarily flown utilizing a Marine Corps photographer. Usable film footage returned from this mission was inadequte to support the experiment; therefore, a second mission was

requested. Proper authorization was received, and this mission was flown with a photographer from the Naval Undersea Center assisted by DSI personnel. Based on evaluation of the first mission's film, 1,000' AGL was selected as the filming altitude for this flight (at altitudes above 1,000' the targets were difficult to detect on the film, even before bandwidth reduction). A horizontal offset from the target of 1 to 2 km was used with camera angles of 90° and 20° to the line of flight. The helicopter speed during the filming was approximately 55 kts; the camera was mounted so it extended through the open helicopter side door into the airstream and at speeds above 55 kts. the wind caused unacceptable camera buffeting.

Editing and Recording

The film was processed and then edited to 26 short clips, each of approximately 8 to 30 seconds duration. These short scenes were then combined into a single film which the Naval Undersea Center processed by the bandwidth reduction technique and recorded on video cassettes.

Four reduction levels were chosen for the final experiment plus the original film. These levels are 1600 kbit/sec with 0 Bit Error Rate (BER), 1600 with 10^{-3} BER, 400 with 0 BER, and 400 with 10^{-3} BER. Figure 1 shows still photo examples of one scene at the two data rates, each at two bit error probabilities and the original.

Navy Personnel Research and Development Center and Decision Science, Inc. personnel then annotated and reedited these video cassettes onto 1/2 inch video tape. This last step was taken to facilitate scene selection since specific points on the tape can be more rapidly and accurately selected using the standard tape than is possible with current video cassette recorders.

Target Selection

Targets suitable for the experiment included vehicles (tanks, jeeps, and personnel carriers), buildings, and electrical transmission towers. Targets were subjectively ranked by level of difficulty of detection on a ten scale. This ranking was largely based upon the time to find the target in the original film. These rankings were used to determine the scene groupings as described below. The number of targets on each film clip varied from one to twelve. For the experiment only 20 scenes were required so 6 scenes were not used. The six scenes discarded were those with the most targets and those with targets that were too easily detectable.

Scene Grouping

The 20 scenes were grouped into five sets of four scenes each. This grouping was based on the total lengths of the four scenes and the number and difficulty level of the targets. Each scene group had a total length of approximately 116 seconds running time (range from 114 to 122) and approximately

9 targets total (8 to 11 range) with the overall difficulty levels of the targets as equal as possible.

Experimental Setup

Hardware

The hardware for the experimental setup involved a video tape machine, a video monitor, an eye tracking device, a PDP-12 computer, and an interface switcher between the computer and subject. Figure 2 shows a diagram of the physical setup and interconnection of these devices.

The first step in an experimental run was to start the video tape. When the scene initially appeared on the video monitor, the experimental observer turned the data switch "on" to signal the computer to start collecting data. At this time the computer monitored and processed the eye tracking data. The subject scanned the scene during the run, pressing the target identification button (subject response button) when he observed what he believed to be a target. At the same time the experimental observer monitored the subject's eye track on a separate video monitor (oculometer video display) linked to the eye tracking device. This monitor had a superimposed spot on the image indicating the subject's present look point. If the observer determined that the subject was scanning an actual target when the target identification button was pressed, the observer pressed the target verification button. The subject's target identification button caused a red light on the observer's control

panel to light (this button also emitted an audible click when pressed). The observer's verification button is silent so that the subject was not aware that it had been pressed. When the scene ended, the observer turned off the data "on" switch. The computer then turned off the green light (computer ready light) on the observer's control panel. The data from the completed run was then processed and the next run set up. When the computer was ready for the next run, it turned on the green (computer ready) light.

This experimental setup has some inherent drawbacks, the primary being that the data collection start and stop timer which is started and stopped by an observer actuated switch is subject to observer error; however, timing errors were determined to be in the range of only one second maximum and $\pm$ .5 seconds on the average. There are several advantages of this design. The primary advantage is that there is no need for a hardware link between the video tape and the computer. To solve the problems involved in a link of this kind would have been costly, time consuming, and beyond the scope of this experiment. The same advantages are basically true for the target verification scheme, although here the main drawbacks to computer controlled verification would have been software complications and computer memory restrictions.

Software

A computer program was written to monitor and process the eye track data and the targeting data during the

experiment. This program also summarizes the results of each run and performs initial subject eye track calibration. A complete listing of this program is shown in Appendix A. Figure 3 shows a generalized flowchart of the program logic.

The program intially reads the subject number, date, and time. A calibration routine which displayed the subject eye track on a CRT was then called. This routine allowed the computer operator to read in maximum and minimum x-y coordinate values in order to normalize the eye track for all subjects. Once the normalizing values were initialized, the program read in the scene and run numbers for the next run, turned on the observer computer ready light, and awaited the data "on" switch.

Once the data switch was switched "on," the program kept track of time, read the eye track data, normalized, smoothed, and processed this data, and wrote it on mag tape. The normalizing and smoothing algorithm normalized the eye movement data to 0 $\pm$ 512 and calculated a running average of every three eye track data points, ignoring values which exceed the calibration limits as these indicated look points outside of the video screen, eye blinks, or spurious noisc in the raw data.

During this process the program also checked to see if the target identification and verification buttons had been pressed. If they had been pressed, the time was saved along with the x,y look points at that time. All of this data was stored in computer memory during the data collection.

1460

At the end of each run a summary routine was called. This routine printed the total time, t, in one eighth seconds and the total amount of eye movement during the run.

The number of targets detected was also printed. For each target detected three data points were printed. These include the target detection time, time of verification by the observer, and three look points. These look points are the look point immediately before the detection, the look point at detection, and the point following detection. The use of this data is described in the measurements section.

Upon completing the summary, the computer awaited the next scene, run numbers, and the observer's run switch. Table 1 shows a sample print of the run summary.

Subjects

Ten Marine photointerpreters from Camp Pendleton were used as subjects in this experiment. The subjects' ages ranged from 24 to 38 years, averaging 30. All subjects were male and had a high school education or better (average education: 13 years). All subjects had some photointelligence experience, most had two or more years experience. All but two subjects had satisfactory vision (the eye track device does not track through glasses or contacts). Two subjects that had poor vision were tested on several sample runs. They could find the targets on these runs that were rated high on the difficulty scale with the same relative consistency as the subjects who did not wear glasses.

One or two subjects demonstrated erratic eye and head movements making it difficult to calibrate the eye tracking device; however, this did not effect their targeting performance. This effect does appear in increased eye movement measures.

Statistics on the subjects are summarized in the subject questionnaire (Appendix B).

Subject Tasks

Each subject was prebriefed about the experiment by being read the experiment description shown in Appendix B titled "Instructions to Subjects." The subjects then filled out a questionnaire (also shown in Appendix B). The subjects were then positioned in front of the eye tracking device which was calibrated to their own head position and eye movement.

Once the actual run began, the subjects were required to scan the video monitor; when a target appeared on the screen, the subjects pressed the subject response button in the hand-held box. A few times during the experiment there were uncertainties as to what order the targets were identified. When this occured, the uncertainties were resolved by questioning the subject at the end of the run.

Data Collection

The data was collected and normalized approximately every 45 ms and smoothed every 1/8 second (125 ms) according to the following algorithm:

If we let

x_i, y_i = original eye track at time i (every 45 ms or 24 per second)

$x_{max}, y_{max}, x_{min}, y_{min}$ = calibration values

$\hat{x}_i, \hat{y}_i$ = normalized values at time i

$\bar{x}_j, \bar{y}_j$ = smoothed values at time j (every 125 ms or 8 per second)

then

$$\hat{x}_i = \left[x_i - (x_{max} + x_{min})/2\right] \cdot \left[1024/(x_{max} - x_{min})\right]$$

$$\hat{y}_i = \left[y_i - (y_{max} + y_{min})/2\right] \cdot \left[1024/(y_{max} - y_{min})\right]$$

Now if $\hat{x}_i$ or $\hat{y}_i$ is greater than or less than ±512, they are set to the previous value between ±512. Next, for the running average smoothed values

$$\bar{x}_j = (\hat{x}_{i-1} + \hat{x}_i + \hat{x}_{i+1})/3 \qquad j = i/3$$

$$\bar{y}_j = (\hat{x}_{i-2} + \hat{x}_{i-1} + \hat{x}_i)/3 \qquad j = i/3$$

These smoothed values were calculated and stored on magnetic tape every one-eighth second. At this same time, a target number pointer was stored indicating whether or not a new target had been detected in the last time step.

Measurements

Several scoring measures were calculated for each run. The computer calculated the total and average eye movement per time for each run. The total eye movement is as follows:

$$EM = \sum_{j=1}^{n} \sqrt{(\bar{x}_j - \bar{x}_{j-1})^2 + (\bar{y}_j - \bar{y}_{j-1})^2}$$

where $\bar{x}_0, \bar{y}_0$ are the midpoints of the calibration values, $\frac{x_{max} + x_{min}}{2}$ and $\frac{y_{max} + y_{min}}{2}$. Here, n is the number of one-eighth second time steps in the run. The average eye movement per time is:

$$\overline{EM} = EM/n$$

The time of each target detection was recorded by the computer, and these times were compared to the actual time the target appeared on the screen to get a target detection late time. Each target that was detected by the subject was marked as either false or real. Real targets were verified by the observer during the run and also cross checked against the summary printout at the end of each run. Each run and each target was thoroughly screened and calibrated prior to running the experiment. This determined the target time on the screen and target location. Each look point has the x,y coordinates and a horizontal and vertical section number from one to three. The lower left section of the video screen was section 1,1 and the upper right was 3,3 (see Figure 3A). A sample printout is also shown in Table 1. In this example the subject (subject No. 5) identified three targets with a total eye movement of 18186.7 units and an average eye movement of 127.16 units per time step (these are normalized units, and the time step is one program cycle or approximately one eighth of a second). He identified the three targets at times 33, 78, and 99.

The first two targets were verified at times 44 and 88, respectively. The third target was not verified by the observer (verify time = 0). At the time of identification of the first target, the subject was looking at sections 2,3 and 2,2. The second target was identified while looking in 2,3 also. The third target had an associated look point in section 1,2.

The calibration runs on scene 16 showed three targets. The first appeared at time 25 in section 2,1 and moved to section 2,2 then back to 2,1. This data confirmed the observer's verification and resulted in a late time of eight time units (approximately one second). The second target appeared at time 70 and moved from section 2,3 to 2,2 to 2,1. A late time of eight units was also scored for this target. A third target appeared at time 90 in section 3,3 and moved through 3,2 to 3,1. The subject's identification time corresponded to the calibrated appearance time, but the look point was on the wrong side of the screen. This along with the observer's lack of verification resulted in a false target. Questioning of the subject regarding the description of this target confirmed that it was false.

The subject's targeting score on this run was one missed target, one false target, and two correct targets both with late times of eight time units. Scene 16 was part of scene group 5 which has 8 targets. Subject No. 5 identified five of the eight targets correctly for a percent

correct score of 62.5; he had only one false target and his average late time was 49.4 time units (about 6 seconds). These results are shown for all subjects and all runs in Table 2.

Each subject saw twenty scenes in five groups of four scenes each. For each scene group the number of correct targets, number of targets missed, and the number of false targets were the calculated scoring measures. Also, for each scene group the sum of the natural logarithm of eye movement for each scene was calculated along with the sum of the late times for the targets correctly identified. These made up the scoring measures for each scene group.

Analysis

Each subject viewed each of the five scene groups once. The data bit and error for each scene group varied for each subject. Table 3 shows the order and reduction level of the scenes for each subject. Each scene group was shown to two subjects under each of the levels.

Three aspects of the experiment were considered in the analysis. These were the differences between subject behavior, the differences between scene groups, and the effects of the bit and error rate.

Each of these effects were tested separately using analysis of variance on each of the following variables. For each group

Eye movement measure = sum of natural log of each eye movement for scene

Average late time = sum of target detection late time divided by the number of correct targets

Percent correct = number of targets correctly detected divided by total number of targets in the scene group

False targets = Number of false targets detected

For each scene the average eye movement was analyzed. Table 2 presents the raw data for these variables.

Results

Subject and Scenes

The initial analysis of variance discovered significant differences between subjects on the eye movement measure and the false targets. The scene groups demonstrated significant differences on the late time and percent correct measures. Tables 4 and 5 show these results for subjects' differences and Tables 6 and 7 show the results for the scenes. The analysis of variance tables (Tables 4 and 6) show the interim values used to calculate the test statistic for each variable. These are the sum of squares, the degrees of freedom, and the mean square. The resulting F value is compared to probability tables to find the significance levels of each variable shown in the probability column. The

individual effects tables (Tables 5 and 7) show the effect of each scene or subject on the specific variables and the subjects or scenes that caused the significant differences; for example, the subject effects on the eye movement variable show an average E.M. score of 39.32. Subject No. 7 had an added effect of .75 giving an average of 40.07. Subject No. 9 had an effect of -1.51 giving an average E.M. = 38.81 (= 39.32 - 1.51). Subject No. 9 had significantly lower eye movement than all but Subject No. 6. This is seen in the table where Subjects No. 6 and No. 9 are both underscored by the lower line indicating no significant difference at .05 level. Also, there were no significant differences between Subjects No. 1-8 and No. 10, which is indicated by all being underscored by the upper line. The significant effect of subjects on the eye movement measure is, therefore, due to the small E.M. measure of Subject No. 9. For the false targets measure, Subject No. 4 had significantly more false targets than all but Subjects No. 2 and No. 6.

Several interesting behavior trends may be noted from the subject effects. There is some interaction between the false targets and late time. Here, the subjects who were high in the false targets score (Subjects No. 4 and No. 2) were low on the late time. This indicates that the increase in false targets may be due to reacting too quickly before being sure that the targets are real. Subjects No. 1 and No. 3 who are high on late time scores also tend to be low

on the false targets again indicating the longer the subject waits to identify the target the less likely the target will be false.

The individual scene group differences showed significantly shorter late times for scene group 1 than group 5. Scene groups 3 and 1 have significantly higher percent correct than scene group 2.

Bandwidth Reduction

Four levels of scene modification and one normal segment (level N) were presented to each subject. The four bit and error rate levels consisted of a 1600 kbit/sec rate with 0 error rate (level A), 1600 kbit/sec with 10^{-3} error rate (level B), 400 kbit/sec rate with 0 error (level C), and 400 kbit/sec with 10^{-3} error (level D).

The analysis of subjects and scene groups identified significant effectson certain variables (late time and percent correct were affected by the scene groups, and the subjects affected eye movement and false targets). Inclusion of these effects in the data would have resulted in increased variability in the data which would have masked the effects of the Bandwidth Reduction Techniques. These effects were removed from the data by subtracting the subject or scene effect from data points on the variables where these effects were significant; for example, Subject No. 7 had eye movement scores an average of .75 higher than the other subjects. Subtracting .75 from each eye movement score of Subject No. 7 reduced his average to 39.32; likewise,

adding 1.51 to the eye movement scores of Subject No. 9 brought his average up to 39.32.

Tables 8 and 9 show the results of the analysis of the effects of varying bit and error rates. The only variable significantly affected by the bit and error rate was the percent correct score. Here, the C level (400 kbit-zero error) was significantly less than the no reduction level. No other significant differences were noted. Further investigation of the reduction levels shows that error rate change did not significantly affect the percent correct score (levels A and B and levels C and D are not significantly different). The main effect noted is the bit rate. A plot of the average effect of the bit rates reduction on percent correct is shown in Figure 4. Here, no reduction = 83.7% correct, 1600 kbit/sec. = 70.8%, and 400 kbit/sec. = 62.4%. Interpolating this data would indicate approximately 65% correct for the 800 kbit/sec. rate. Any extrapolation below 400 kbit/sec. would have to be based on some nonlinear trend to 0% correct at 0 kbit/sec.

Some trends indicated from the analysis are that false targets increase as the bit rate increases. This could be due to subjects picking out only the obvious targets in the degraded situations and tending to be less conservative with the improved picture quality. Also, the late times are very consistent under all bit rates. This indicates that if targets are found, the time to detect them is not affected by the bit rate level. No indication that the bit and error rate affected eye movement was found.

Conclusions

The primary conclusion drawn from this analysis is that the bit rate does effect target detection performance. A percent correct detection of over 80% can be expected from the original picture with a reduction of around 8% every time the bit rate is reduced by one half. The experimental results show a significant difference between the percent correct in the original scenes and the scenes with the 400kbit rate.

Several trends (non-significant) lead to tentative conclusions concerning the bandwidth reduction effects. The first of these regards the false targets; here more false targets were detected with increases in picture quality. This indicates that the subjects wanted to be more sure of a target when the picture was degraded. Another conclusion can possibly be drawn from the late time consistency. Even though the lower bit and error rates decreased detection efficiency, it did not effect time to detect the targets that were detected. This indicates that the difficult to find targets were never found in the degraded scenes whereas the easy targets were found quickly no matter what the mode of reduction.

Recommendation

This experiment was conducted in such a way as to simulate as nearly as possible the actual operational situation in which these bandwidth deduction techniques will be used. Subjective informal comments by the subjects confirmed that the simulation was realistic. The results and conclusions

drawn from this experiment should be applicable to any future use of these techniques.

The primary goal of this experiment was to measure and quantify the effects of this bandwidth reduction method. This has been done. Any recommendation as to the application of these techniques will have to be made by the eventual user. The main consideration is how much reduction in target detection confidence the user is willing to tolerate in order to have secure transmission of the data link. In other words what percentage of the available targets can nominally be correctly detected for the mission to be considered successful.

Several factors were not considered in this experiment which could increase the overall detection rate. The primary factor here would be training of the subjects on these particular kinds of target detection tasks. Familiarity both with the terrain and with specific types of targets should increase target detection for all reduction levels. Also control of the camera zoom and slew by the operator may increase the detection rate.

This study indicated that "easy" targets (on roads, moving, etc.) are detected regardless of bit rate.

The platform for the sensor simulated in this report is assumed to be an RPV. This RPV will be small, relatively quiet, and have a minimal radar image; thus, it will be difficult to detect by the targets which are being searched.

If these targets are unalerted to the possibility of reconnaissance in their area, they are likely to be "easy" targets.

Overall, the experiment shows that the subjects can perform the targeting task at all bit and error rates used, albeit with reduced efficiency at the lower bit rate.

REFERENCES

Ref. 1 Image Transmission Via Spread Spectrum Technique Appendix G. ARPA Annual Technical Report, 1976, Order No. 2303, Code #3G10

SAMPLE PICTURES

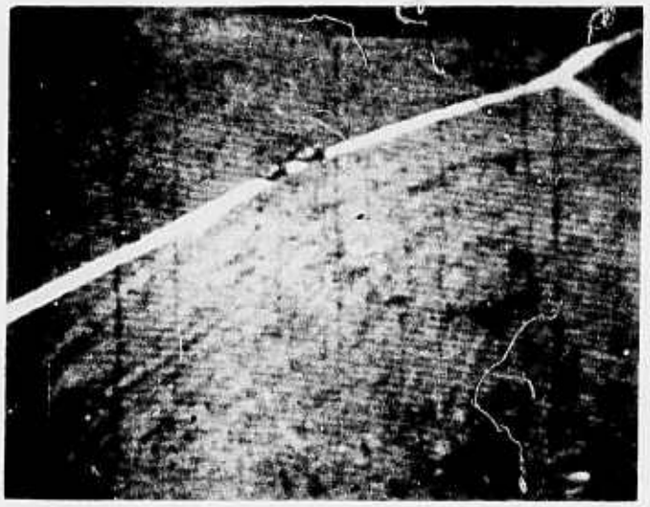


Figure 1A
Original Picture
(Videotaped from the film)

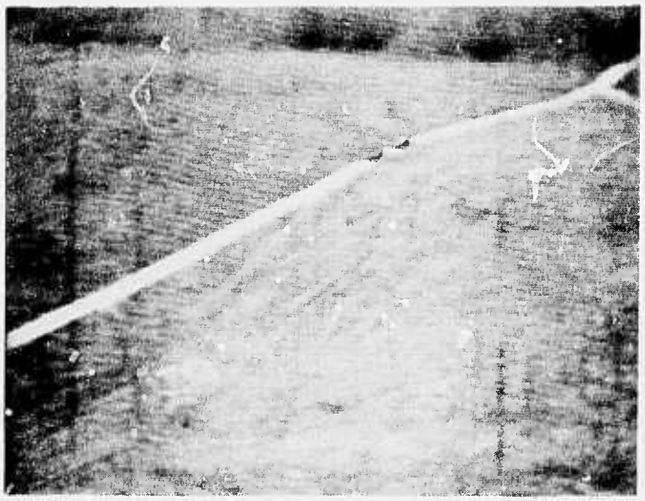


Figure 1B
1600 kbits/sec (0 BER)

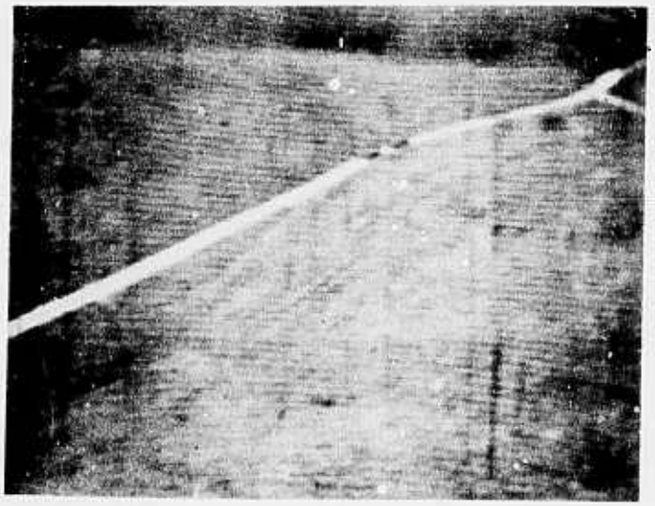


Figure 1C
1600 kbits/sec (10^{-3} BER)

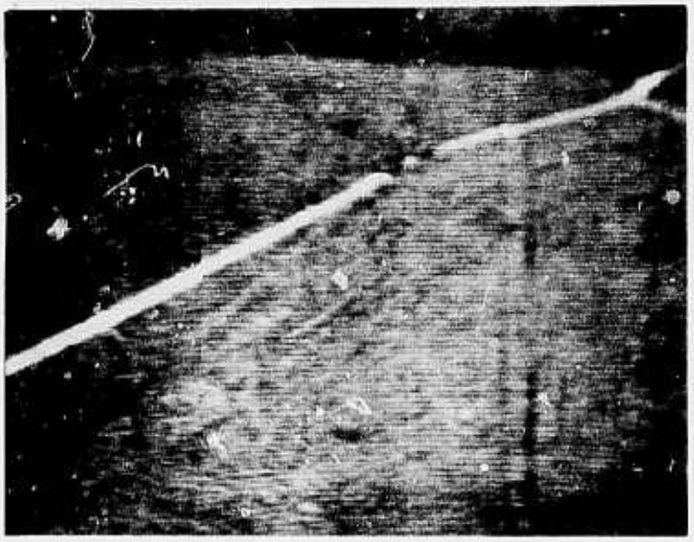


Figure 1D
400 kbits/sec (0 BER)

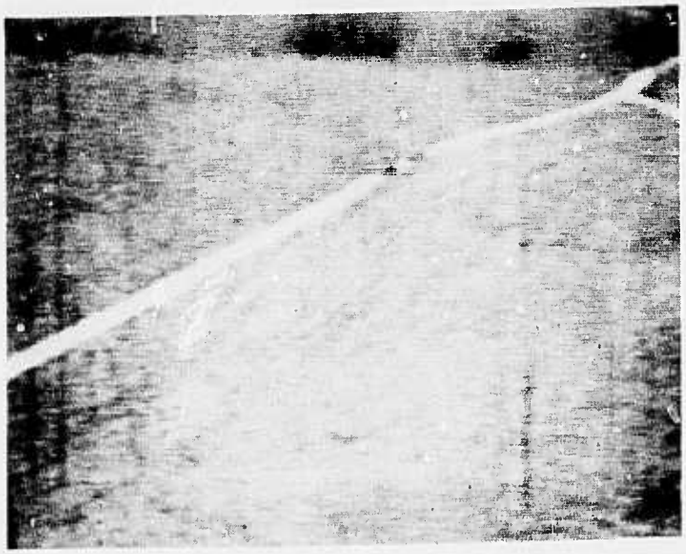


Figure 1E
400 kbits/sec (10^{-3} BER)

BANDWIDTH REDUCTION EXPERIMENT SET UP

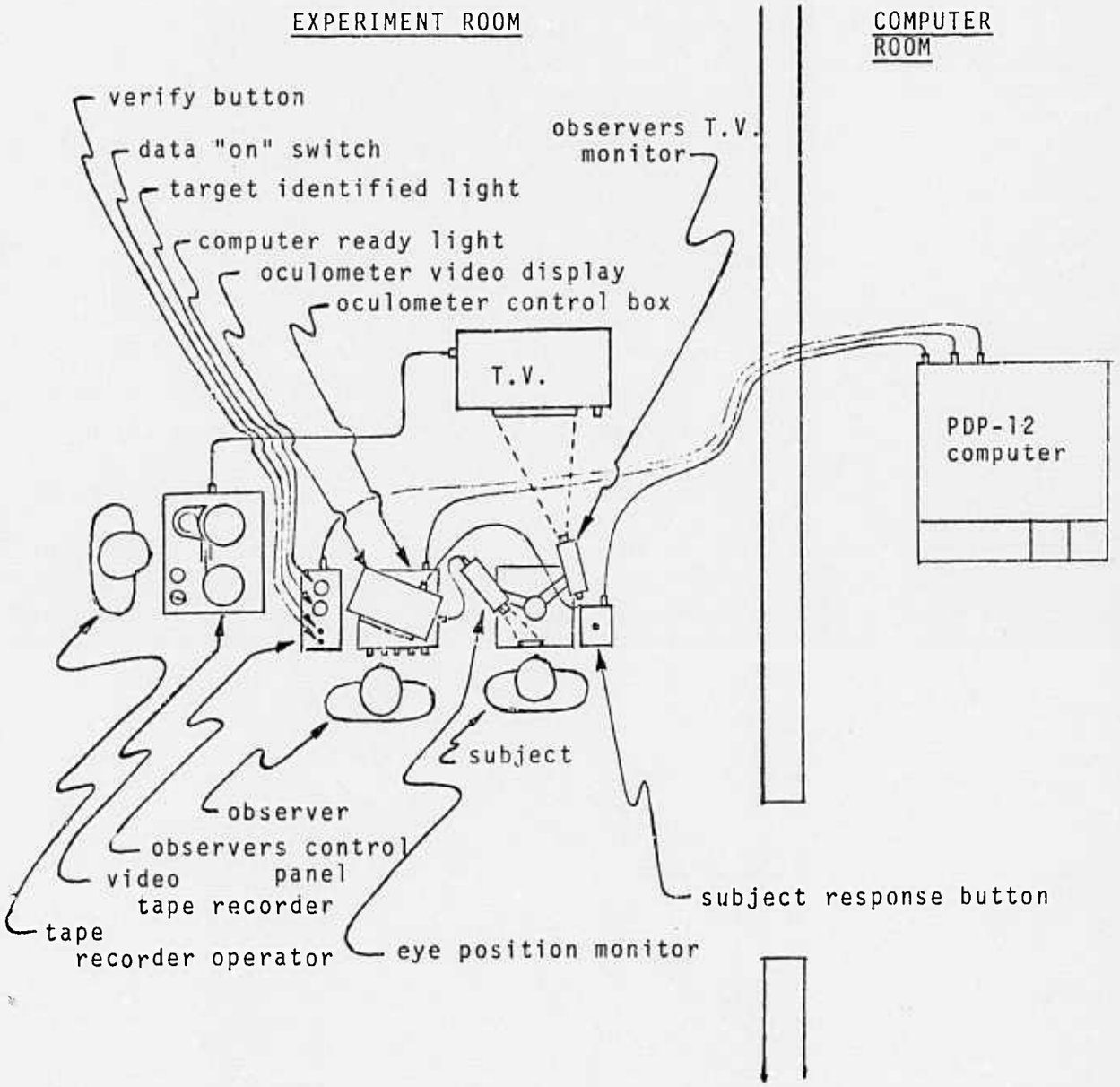


FIGURE 2

COMPUTER PROGRAM FLOWCHART

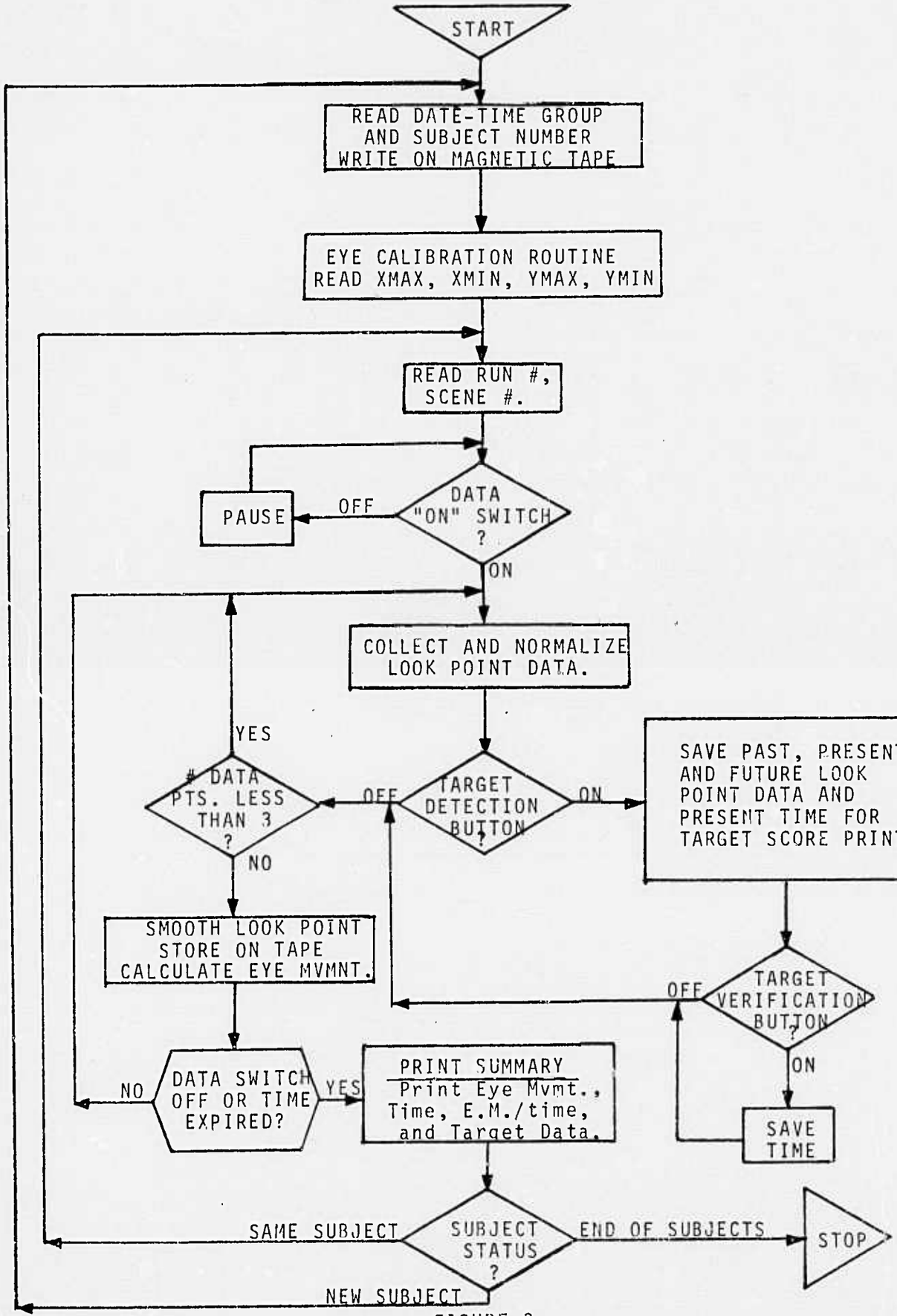


FIGURE 3

164

TV SCREEN SECTIONS

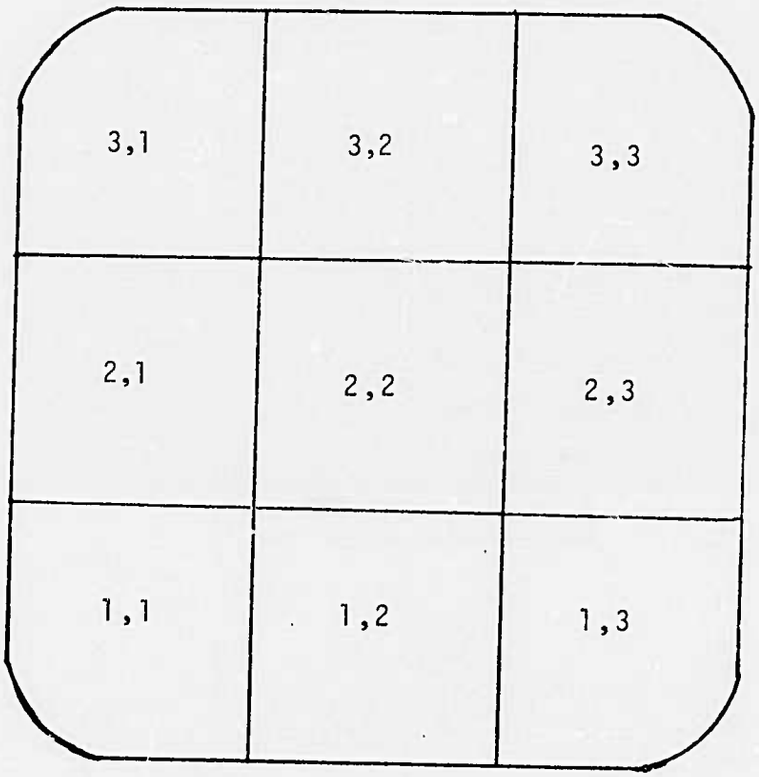


FIGURE 3A

BIT RATE REDUCTION EFFECT
ON PERCENT CORRECT SCORES

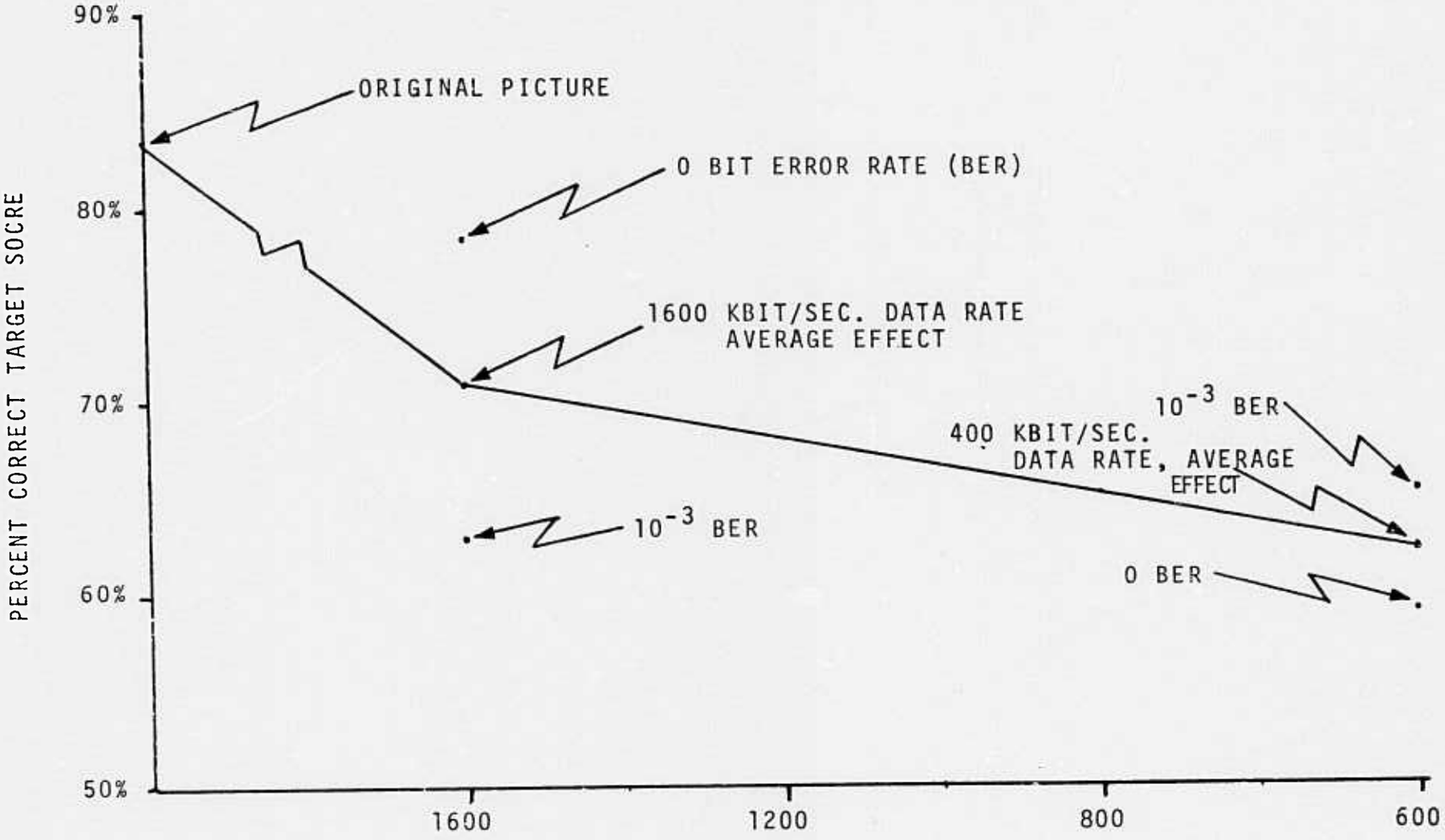


FIGURE 4

SAMPLE OF SUMMARY PRINT OUT

RUN = 10
SCENE = 16
TIME = 143
TARGETS I.D.ED= 3
EYE MOVEMENT = +0.181867E+05
EYE MVMNT./TIME = +0.127160E+03

TGT. NO.	I.D. TIME	VFY. TIME	X COORD.	Y COORD.	SECT. HOR.	VERT.
1	33	44	- 38	175	2	3
1	33	44	- 74	187	2	3
1	33	44	- 67	153	2	2
2	78	88	4	491	2	3
2	78	88	- 15	481	2	3
2	78	88	- 51	470	2	3
3	59	0	- 178	123	1	2
3	99	0	- 178	64	1	2
3	99	0	- 206	- 14	1	2

TYPE 0 FOR SAME SUBJECT , +1 FOR NEW SUBJECT AND -1 FOR END:0

TABLE 1

BANDWIDTH REDUCTION

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
1	1.	1.	39.6900	105141.	34.900000	.9091	.0909	0.
1	2.	2.	39.9400	102722.	33.666667	.6667	.3333	2.
1	4.	4.	40.5500	104714.	47.166667	.6667	.3333	0.
1	3.	3.	39.8700	93878.	27.500000	.6667	.3333	2.
1	5.	5.	40.2000	108689.	73.333333	.7500	.2500	0.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
2	2.	5.	39.4500	89887.	45.000000	.8750	.1250	4.
2	3.	1.	38.7300	75111.	26.333333	.8182	.1818	2.
2	5.	3.	39.2500	80053.	25.428571	.7778	.2222	3.
2	4.	2.	38.6800	76720.	49.250000	.4444	.5556	5.
2	1.	4.	39.5900	82104.	21.200000	.5556	.4444	5.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
3	3.	4.	39.7400	84028.	47.166667	.6667	.3333	1.
3	4.	5.	39.2000	97654.	31.285714	.8750	.1250	1.
3	1.	2.	38.8900	89299.	71.000000	.4444	.5556	1.
3	2.	3.	38.7000	69425.	47.166667	.6667	.3333	1.
3	5.	1.	38.9800	76729.	18.111111	.8182	.1818	0.

DATA PRINTOUT

Summary Values for Each Subject for Each Scene Group
Printed in Bit Rate Order - Original,
1600 - 0BER, 1600 - 10^{-3}BER, 400 - 0 BER,
400 - 10^{-3}BER

TABLE 2

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
4	4.	3.	39.7700	95754.	14.000000	.8889	.1111	6.
4	5.	4.	40.9500	113997.	18.714286	.7778	.2222	2.
4	2.	1.	38.3500	72285.	29.868889	.8182	.1818	8.
4	1.	5.	38.5400	77002.	13.000000	.5000	.5000	7.
4	3.	2.	39.9500	106934.	27.200000	.5556	.4444	8.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
5	5.	2.	38.7100	79024.	22.222222	1.0000	.0000	0.
5	1.	3.	39.0700	81160.	29.777778	1.0000	.0000	1.
5	3.	5.	39.7100	96045.	49.400000	.6250	.3750	1.
5	2.	4.	39.9600	89164.	46.857142	.7778	.2222	1.
5	4.	1.	39.4000	95676.	11.333333	.5455	.4545	0.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
6	1.	5.	38.9100	78808.	50.500000	.5000	.5000	7.
6	3.	3.	38.2500	64860.	32.111111	1.0000	.0000	5.
6	5.	1.	39.5200	90355.	10.714286	.6364	.3636	0.
6	2.	4.	39.5600	79652.	63.000000	.5556	.4444	4.
6	4.	2.	38.6200	73933.	35.250000	.4444	.5556	2.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
7	5.	5.	40.1100	91317.	34.500000	.8889	.1111	3.
7	2.	2.	39.1900	102817.	59.750000	.4444	.5556	1.
7	4.	4.	40.1200	113592.	45.600000	.6250	.3750	0.
7	1.	3.	40.7600	122306.	42.800000	.5556	.4444	4.
7	3.	1.	40.1700	113313.	21.142857	.6364	.3636	1.

DATA PRINTOUT
TABLE 2 (Continued)

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
8	4.	3.	39.6500	91658.	24.888889	1.0000	.0000	1.
8	1.	1.	38.6100	77888.	24.111111	.8182	.1818	5.
8	3.	4.	40.0700	92915.	18.571429	.7778	.2222	0.
8	5.	2.	38.9900	86654.	42.000000	.4444	.5556	1.
8	2.	5.	39.4500	91731.	77.833333	.7500	.2500	0.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
9	3.	2.	38.0600	67797.	24.000000	.8000	.2000	3.
9	5.	5.	38.1400	64131.	62.000000	.5000	.5000	0.
9	2.	3.	36.9100	48077.	33.200000	.5556	.4444	3.
9	4.	1.	38.0300	64484.	26.000000	.6364	.3636	3.
9	1.	4.	37.9300	54978.	33.600000	.5556	.4444	0.

SUBJ. NO.	RUN	SCENE GP.	SUM LOG E. M.	EYE MVMNT.	AVE. LATE TIME	PCNT. CORRECT	PCNT. MISS	NO. FALSE
10	2.	1.	39.5800	101348.	34.454545	1.0000	.0000	4.
10	4.	4.	40.4400	101193.	51.375000	.8889	.1111	2.
10	1.	2.	39.8300	108888.	30.000000	.3333	.6667	1.
10	3.	5.	39.7600	101673.	38.666667	.7500	.2500	0.
10	5.	3.	39.4900	83841.	48.142857	.7778	.2222	0.

DATA PRINTOUT

TABLE 2 (Continued)

021

SUBJECT NUMBER

RUN #	1			2			3			4			5			6			7			8			9			10					
	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*	SCENE	SGCR EONU EP	EFFECT*
1	4	1	No	23	4	D	18	2	B	7	5	C	26	3	A	16	5	No	2	3	C	4	1	A	3	4	D	18	2	B			
2	8	1	No	10	4	D	15	2	B	14	5	C	24	3	A	21	5	No	12	3	C	8	1	A	5	4	D	15	2	B			
3	17	1	No	5	4	D	13	2	B	16	5	C	12	3	A	14	5	No	24	3	C	17	1	A	10	4	D	13	2	B			
4	22	1	No	3	4	D	6	2	B	21	5	C	2	3	A	7	5	No	26	3	C	22	1	A	23	4	D	6	2	B			
5	6	2	A	7	5	No	2	3	C	22	1	B	23	4	C	3	4	C	6	2	A	21	5	D	26	3	B	22	1	No			
6	13	2	A	14	5	No	12	3	C	17	1	B	10	4	C	5	4	C	13	2	A	16	5	D	24	3	B	17	1	No			
7	15	2	A	16	5	No	24	3	C	8	1	B	5	4	C	10	4	C	15	2	A	14	5	D	12	3	B	8	1	No			
8	18	2	A	21	4	No	26	3	C	4	1	B	3	4	C	23	4	C	18	2	A	7	5	D	2	3	B	4	1	No			
9	2	3	C	22	1	A	23	4	No	18	2	D	7	5	B	2	3	A	4	1	D	3	4	B	18	2	No	21	5	C			
10	12	3	C	17	1	A	10	4	No	15	2	D	14	5	B	12	3	A	8	1	D	5	4	B	15	2	No	16	5	C			
11	24	3	C	8	1	A	5	4	No	13	2	D	16	5	B	24	3	A	17	1	D	10	4	B	13	2	No	14	5	C			
12	26	3	C	4	1	A	3	4	No	6	2	D	21	5	B	26	3	A	22	1	D	23	4	B	6	2	No	7	5	C			
13	23	4	B	6	2	C	7	5	A	26	3	No	22	1	D	6	2	D	21	5	B	2	3	No	4	1	C	3	4	A			
14	10	4	B	13	2	C	14	5	A	24	3	No	17	1	D	13	2	D	16	5	B	12	3	No	8	1	C	5	4	A			
15	5	4	B	15	2	C	16	5	A	12	3	No	8	1	D	2	2	D	14	5	B	24	3	No	17	1	C	4	4	A			
16	3	4	B	18	2	C	21	5	A	2	3	No	4	1	D	18	2	D	7	5	B	26	3	No	22	1	C	23	4	A			
17	7	5	D	2	3	B	22	1	D	3	4	A	18	2	No	1	1	B	4	4	No	6	2	C	21	5	A	26	3	D			
18	14	5	D	12	3	B	17	1	D	5	4	A	15	2	No	8	1	B	5	4	No	13	2	C	16	5	A	24	3	D			
19	21	5	D	24	3	B	8	1	D	10	4	A	13	2	No	17	1	B	10	4	No	15	2	C	14	5	A	12	3	D			
20	16	5	D	26	3	B	4	1	D	23	4	A	6	2	No	22	1	B	23	4	No	18	2	C	7	5	A	2	3	D			

*No-Original Picture; A-1600 kbit/sec, 0 BER; B-1600 kbit/sec, 10^{-3}BER; C-400 kbit/sec, 0 BER; D-400 kbit/sec, 10^{-3}BER.

SCENE AND RUN ORDER TABLE

TABLE 3

121

ANALYSIS OF VARIANCE TABLES, SUBJECT DIFFERENCES

Effect	Sum of Square	Degree of Freedom	Mean* Squares	F** Value	Probability of a larger F Value by Chance***
		Eye Movement Measure	$\left[\sum_{i=1}^{4} \ln (\text{E.M.})\right]$		
Subject	19.275	9	2.142	6.67	.00001
Residual	12.840	40	.321		
Total	32.115	49			
		Late Time	$\left[\sum_{i=1}^{\#\text{ correct}} \text{Late time}\right]/(\#\text{ correct})$		
Subject	2057.710	9	228.634	.845	.58006
Residual	10822.760	40	270.569		
Total	12880.469	49			
		% Correct	[# correct/(# correct + missed)]		
Subject	.167	9	.019	.547	.831
Residual	1.357	40	.034		
Total	1.525	49			
		# False			
Subject	142.980	9	15.887	5.40	.00008
Residual	117.600	40	2.940		
Total	260.580	49			

*Mean Square = Sum of Squares ÷ Degrees of Freedom
**F Value = Mean Square Subject ÷ Mean Square Residual
***Probability values less than .05 indicate significant differences on the referenced variable between at least two of the subjects. The individual effect are shown in Table 5.

TABLE 4

SUBJECT DIFFERENCES/INDIVIDUAL EFFECTS

ORDERED SUBJECT EFFECTS
(Differences above or below the average)

Variable	Ave.									
		Subj.7	Subj.1	Subj.10	Subj.4	Subj.5	Subj.8	Subj.2	Subj.3	Subj.6	Subj.9
Eye Movement	39.32	.75	.73	.50	.19	.050	.034	-.18	-.22	-.35	-1.51
Significant groups *											
		Subj.1	Subj.3	Subj.7	Subj.10	Subj.6	Subj.8	Subj.9	Subj.2	Subj.5	Subj.4
Late Time	36.5	6.81	6.44	4.26	4.03	1.81	.98	-.74	-3.06	-4.58	-15.94
		Subj.5	Subj.8	Subj.10	Subj.1	Subj.4	Subj.3	Subj.2	Subj.7	Subj.6	Subj.9
% Correct	70%	9%	6%	5%	3.3%	.9%	-.51%	-.51%	-7%	-7.2%	-9%
		Subj.4	Subj.2	Subj.6	Subj.7	Subj.9	Subj.8	Subj.10	Subj.3	Subj.1	Subj.5
# False Targets	2.22	3.98	1.58	1.38	-.42	-.42	-.82	-.82	-1.42	-1.42	-1.62
Significant groups *											

* Subjects effects underscored by the same line are not significant at the .05 level of significance.

TABLE 5

ANALYSIS OF VARIANCE TABLES, SCENE DIFFERENCES

Effect	Sum of Square	Degree of Freedom	Mean* Squares	F** Value	Probability of a larger F Value by Chance***
		Eye Movement			
Scenes	4.49	4	1.123	1.83	.14
Residual	27.62	45	.614		
Total	32.12	49			
		Late Time			
Scenes	3185.9	4	796.5	3.70	.011
Residual	9694.6	45	215.4		
Total	12880.5	49			
		% Correct			
Scenes	.324	4	.081	3.04	.027
Residual	1.2	45	.027		
Total	1.52	49			
		False Targets			
Scenes	7.08	4	1.77	.314	.867
Residual	253.5	45	5.63		
Total	260.6	49			

*Mean Square = Sum of Square ÷ Degree of Freedom
**F Value = Mean Square Scenes ÷ Mean Square Residual
***Probability values of .05 or less indicate significant differences between at least two scenes on the referenced variable. Individual effects are reported in Table 7.

TABLE 6

ORDERED SCENE GROUP EFFECTS

SCENE EFFECT (Above or below the average)

Variable	Average					
Eye Movement	39.32	Scene Gp.4 .571	Scene Grp.5 .027	Scene Gp. 3 -.148	Scene Gp.1 -.214	Scene Gp.2 -.234
Late Time	36.5	Scene Gp.5 11.0	Scene Gp.2 2.9	Scene Gp.4 2.8	Scene Gp.3 -4.0	Scene Gp.1 -12.8
Significant						
% Correct	70%	Scene Gp.3 8.9%	Scene Gp.1 6.4%	Scene Gp.5 .2%	Scene Gp.4 -1.5%	Scene Gp.2 -14.2%
Significant						
False Targets	2.22	Scene Gp.3 .38	Scene Gp.2 .18	Scene Gp.1 .08	Scene Gp.5 .08	Scene Gp.4 -.72

* Scene groups underscored by the same line are not different at the .05 level of significance.

TABLE 7

ANALYSIS OF VARIANCE TABLES, BANDWIDTH REDUCTION

Effect	Sum of Square	Degree of Freedom	Mean* Squares	F** Statistics	Probability of a larger F Value by Chance***
		Eye Movement			
Bandwidth	.114	4	.029	.1	.98
Residual	12.72	45	.283		
Total	12.84	49			
		Late Time			
Bandwidth	122.4	4	30.6	.1	.98
Residual	14744.1	45	327.6		
Total	14866.5	49			
		Percent Correct			
Bandwidth	.443	4	.111	2.89	.03
Residual	1.721	45	.038		
Total	2.164	49			
		False Targets			
Bandwidth	14.68	4	3.67	1.6	.19
Residual	102.92	45	2.29		
Total	117.60	49			

*Mean Square = Sum of Square ÷ Degrees of Freedom
**F Value = Mean Square Bandiwdth ÷ Mean Square Residual
***Probability values of .05 or smaller indicate significant differences between at least two bandwidth reduction levels on the references variable. The individual effects are reported in Table 9.

TABLE 8

ORDERED BANDWIDTH REDUCTION LEVEL EFFECTS
REDUCTION EFFECTS (Above or below the mean)

Variable	Average					
		Level D*	Level N*	Level B*	Level C*	Level A*
Eye Movement	39.32	.0576	-.0466	-.0004	-.035	-.068
		Level C	Level A	Level D	Level N	Level B
Late Time	34.3	2.82	.096	-.11	-.98	-1.82
		Level N	Level A	Level D	Level B	Level C
% Correct	70%	13.7%	8.4%	-4.4%	-6.9%	-10.3%
Signi cant						
		Level N	Level C	Level A	Level B	Level D
False Targets	2.22	.68	.58	-.12	-.52	-.62

* Level N = no bandwidth reduction
Level A = 1600 kbit/sec 0 BER
B = 1600 kbit/sec 10^{-3} BER
C = 400 kbit/sec 0 BER
D = 400 kbit/sec 10^{-3} BER

** Levels underscored by the same line are not significantly different.

TABLE 9

APPENDIX F: INVESTIGATION OF CODING TECHNIQUES FOR RPV AND SAR IMAGES

Final Report
Contract N00123-75-C-1246

30 June 1977

by
WK Pratt
Image Processing Institute
University of Southern California
Los Angeles CA 90007

FINAL REPORT

Investigation of Coding Techniques for RPV and SAR Images

30 June 1977

Submitted to:

Naval Ocean Systems Center
San Diego, California 92132

by

William K. Pratt
Image Processing Institute
University of Southern California
Los Angeles, California 90007

Contract No. N00123-75-C-1246

Abstract

This report describes research work performed by the University of Southern California, Image Processing Institute for the Naval Ocean Systems Center under Contract No. N00123-75-C-1246 during the period 1 April 1976 to 30 June 1977. During the contract period a new method of adaptive intraframe transform coding was developed and tested. Also, a new algorithm was discovered for block quantization bit assignment. The algorithm, which is applicable for intraframe and interframe transform image coding, provides a reduction in mean square coding error up to about 10%. Work has continued on the final design and evaluation of interframe transform image coding techniques for real time television signals. Efficient methods of adaptive and non-adaptive coding have been developed for image coding rates down to 0.1 bits/pixel/frame. Finally, preliminary investigations were begun on the feasibility of applying image coding methods developed for pictorial images to arrays of data obtained from radar.

Contents

1. Introduction

2. Prior Research Summary

3. Intraframe Transform Image Coding
   Appendix 3A
   Block Quantization Bit Assignment
   Appendix 3B
   Spatially Adaptive Block Quantization
   Transform Image Coding

4. Interframe Transform Image Coding
   Appendix 4A
   Interframe Cosine Transform Image Coding

5. Radar Image Coding

1. Introduction

This report summarizes research work performed by the University of Southern California Image Processing Institute for the Naval Ocean Systems Center under Contract No. N00123-75-C-1246 from 1 April 1976 to 30 June 1977. Research on this contract constitutes an ongoing effort begun in 1973. Section 2 provides a historical summary of past research.

Work on the present contract has concentrated in two areas: improvements in the coding performance of conventional transform and hybrid transform/DPCM image coders, and the application of these coding techniques to synthetic aperture radar (SAR) images and other non-pictorial image data arrays. Sections 3-6 describe research accomplishments during the contract period.

2. Prior Research Summary

The Image Processing Institute of the University of Southern California initiated a research project on 1 March 1973 under Contract No. N00123-73-C-1507 from the Naval Undersea Center to study various schemes of intraframe coding of video signals. The object of the project was to develop an efficient technique for transmitting video signals from remotely piloted vehicles (RPV) to ground stations, compatible with rather severe power and volume restrictions on the transmitter. The primary techniques that were explored for intraframe coding were transform, linear predictive, and hybrid transform/linear predictive techniques <1,2>. The hybrid coding system, which combines attractive features of transform and predictive coding methods, was studied in detail. The specific transform used was the discrete cosine transform <3>. Simulation studies were made on the hybrid cosine/DPCM system implemented to operate in real time through the use of charge coupled devices and conventional DPCM techniques <4>.

In December 1974 the Image Processing Institute embarked on a second phase of the study, under a separate contract (N00123-75-C-1192) from the Naval Undersea Center, to study interframe coding techniques combining spatial and temporal coding of image sequences. Coding systems based upon the two-dimensional cosine and Fourier transforms combined with DPCM in the temporal direction were investigated. Results were presented in terms of coding efficiency, storage requirements, computational complexity, and sensitivity to channel noise <5>. Performance of the hybrid interframe coders was investigated in the presence of channel noise. Results show that, for both coder implementations studied, minimal image degradation occurred for a channel error rate of one error per thousand bits or less. In the study, experiments were performed to determine the tradeoff between average bit rate per frame and the frame repetition rate for a fixed channel capacity. The measure of channel capacity used was the bit transfer rate, which is defined as the product of average pixel bit rate per frame and frame rate. The results of these experiments

show that reduced frame rates coupled with correspondingly higher average bit rates produce smaller mean square error for the individual frames coded. This indicates that reductions experienced in frame-to-frame correlations due to temporal subsampling are completely compensated by the increased number of bits available for coding. However, reduced frame rates tend to result in jerky subject motion. In order to simulate this effect on actual RPV data, a new 64-frame 512 x 512 data base was digitized and coded using the two-dimensional cosine transform combined with DPCM in the temporal direction. Computer simulations performed on this data base demonstrated high quality coded images with a 32:1 reduction in the average bit rate.

References

1. J.A. Roese, W.K. Pratt, G.S. Robinson, and A. Habibi, "Interframe Transform Coding and Predictive Coding Methods," Proceedings of the 1975 International Conference on Communications (ICC 75), Vol. II, Paper 23, pp. 16-18, June 1975.

2. A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Transactions on Communications, Vol. COM-22, No. 55, pp. 614-624, May 1974.

3. N. Ahmed, T. Natarajan, and K.R. Rao, "Discrete Cosine Transform," IEEE Transactions on Computers, Vol. C-23, No. 1, pp. 90-93, January 1974.

4. H.J. Whitehouse, R.W. Means, and E.H. Wrench, "Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices," Proceedings of the 19th Annual SPIE Symposium on Efficient Transmission of Pictorial Information, San Diego, August 1975.

5. J.A. Roese and G.S. Robinson, "Combined Spatial and Temporal Coding of Image Sequences," Proceedings of the 19th Annual SPIE Symposium on Efficient Transmission of Pictorial Information, San

Diego, August 1975.

3. Intraframe Transform Image Coding

Since the invention of the transform image coding concept in 1968 <1> there has been considerable research and development directed towards the analysis, improvement, and exploitation of the coding concept. A host of different unitary transforms have been investigated. The consensus among most researchers is that the cosine transform <2> provides the best theoretical and subjective performance among the class of transforms possessing fast computational algorithms.

Parallel investigations have also been made into improved means of quantization and coding. During this contract a new algorithm for transform coefficient quantizer bit assignment has been discovered <3>. On a theoretical basis the algorithm provides minimum mean square error quantization of transform coefficients. Experimental results indicate that the algorithm yields a reduction of about 5% to 8% as compared to conventional approaches. Appendix 3A contains a paper, submitted for publication, which details operation of the algorithm.

Another avenue of potential coding improvement is to adaptively adjust the quantizer scale according to image activity. If transform coefficients are expected to be of low magnitude, the quantizer scale should be shifted and compressed toward zero, and vice versa. Many different techniques have been proposed for sensing the state of transform coefficients and adjusting the quantizer scale. Appendix 3B describes a new technique that has been discovered during the course of this contract for adaptive transform image coding.

References

1. H.C. Andrews and W.K. Pratt, "Fourier Transform Coding of Images," Hawaii International Conference on System Science, January

1968, pp. 577-679.

2. N. Ahmed, T. Natarajan, and K.R. Rao, "On Image Processing and a Discrete Cosine Transform," IEEE Transactions on Computers, Vol. C-23, No. 1, January 1974, pp. 90-93.

3. W.K. Pratt, "Block Quantization Bit Assignment," submitted to IEEE Transactions on Information Theory, April 1977.

APPENDIX 3A

BLOCK QUANTIZATION BIT ASSIGNMENT

Submitted to:

IEEE Transactions on Information Theory

by

William K. Pratt
Image Processing Institute
University of Southern California
Los Angeles, California 90007

April 1977

Block Quantization Bit Assignment

1. Introduction

Block quantization <1> is a method for quantizing vectors and arrays of data samples in which each sample is separately quantized over a scale selected to minimize its mean square quantization error for a given number of quantization levels. The number of quantization levels assigned to each sample is chosen proportional to a code bit allocation subject to a constraint of a fixed sum of block code bits. This paper describes an algorithm for optimal bit assignment.

2. Scalar Quantization

Consider a scalar variable f which is a sample of a random process with known probability density p(f), variance σ_f^2, and mean η_f. The signal f is quantized to the j-th reconstruction level $\hat{f} = r_j$ where $j = 0,1,\ldots,J-1$ if $d_j \leq f < d_{j+1}$ where d_j represents the j-th decision level. Max <2> has found that the mean square reconstruction error $E\{(f-\hat{f})^2\}$ is minimized if the decision and reconstruction levels are chosen to satisfy the equations

$$d_j = \frac{r_{j+1} + r_j}{2} \tag{1a}$$

$$r_j = \frac{\int_{d_j}^{d_{j+1}} f\, p(f)\, df}{\int_{d_j}^{d_{j+1}} p(f)\, df} \tag{1b}$$

For most probability densities solution of eq.(1) can only be achieved by numerical calculation. With decision and reconstruction levels satisfying eq.(1) the residual mean square error becomes

$$\delta = E\{f^2\} - \sum_{j=0}^{J-1} r_j^2 \; P[d_j \le f < d_{j+1}] \qquad (2)$$

where P[.] denotes the occupancy probability of the j-th decision band as defined by the denominator of eq.(1b).

3. Vector Quantization

The mean square quantization error method for scalar variables can, in principle, be generalized to the quantization of data vectors. But, the optimal specification of decision regions and reconstruction vectors requires knowledge of the joint probability density of the vector components. Furthermore, computation can be arduous. A common sub-optimal compromise is to quantize and reconstruct each vector component separately, based upon its marginal density according to eq.(1), for a fixed total number of quantization levels for the entire data vector. The total quantization error of an N element vector is then

$$\delta = \sum_{n=1}^{N} \delta(n) \qquad (3)$$

From eq.(2) the quantization error of the n-th component can be written as

$$\delta(n) = E\{f^2(n)\} - F[J(n)] \qquad (4)$$

where J(n) denotes the number of quantization levels assigned to the n-th component for a given total number of levels

$$J = \sum_{n=1}^{N} J(n) \qquad (5)$$

In eq.(4) the error factor

$$F[J(n)] = \sum_{j=0}^{J(n)-1} r_j^2[J(n)] \; P\left[d_j[J(n)] \le f(n) < d_{j+1}[J(n)]\right] \qquad (6)$$

is written in a form to explicitly indicate the placement dependency

of reconstruction and decision levels upon both the component index (n) and component level assignment J(n). For digital transmission uniform length codes are usually employed, and the number of quantization levels is therefore restricted to be a binary number of the form

$$J(n) = 2^{B(n)} \tag{7}$$

where B(n) is the bit assignment of the n-th component, and

$$B = \sum_{n=1}^{N} B(n) \tag{8}$$

represents the total bit assignment. An optimization problem then arises. How should the levels or bits be assigned to minimize the total mean square error?

4. Log Variance Bit Assignment

Several bit assignment algorithms have been developed in which the number of code bits is assigned proportional to the logarithm of the component variance <3-5>. In the bit assignment algorithm of Ready and Wintz <4>, for zero mean random vectors, the error expression of eqs.(4) to (8) is approximated as

$$\delta(n) = \sigma^2(n) \exp\{-a\, B(n)\} \tag{9}$$

where $a = 1/2 \ln(10)$. Minimization of eq.(9) subject to the constraint of eq.(8) yields the relation

$$B(n) = \frac{B}{N} + 2 \log_{10}[\sigma^2(n)] - \frac{2}{N} \sum_{j=1}^{N} \log_{10}[\sigma^2(j)] \tag{10}$$

Equation (10) usually evaluates to non-integer values, and sometimes negative values. Hence, some roundoff adjustment procedure must be invoked to obtain integer bit assignments. The resulting bit assignment by this process is sub-optimal because of the roundoff and inaccuracy of the approximation of eq.(9) at low bit values B(n). Furthermore, because of the approximation of eq.(9), the log variance

bit assignment algorithm ignores the probability density structure of each sample.

5. Sequential Bit Assignment

Analytic minimization of the quantization error δ of eq.(4) as a function of the bit allocation B(n) is not feasible because of the complicated nonlinear relationship between δ and B(n). However, an optimal solution can be obtained by a sequential search procedure. With no quantization levels assigned for coding, the quantization error specified by eq.(4) is the total signal energy $E\{f^2(n)\}$ summed over all sample values. The first pair of quantization levels, corresponding to the first code bit clearly should be allocated to the vector component that results in the largest decrease in quantization error. Each additional set of quantization levels corresponding to each additional code bit should then be assigned to that component producing the greatest reduction in error. That is, assign the next bit to that component for which the rate of change in error

$$\Delta\delta(n) = F[2J(n)] - F[J(n)] \tag{11}$$

is greatest. For zero mean vectors

$$\Delta\delta(n) = \sigma^2(n)\,[\tilde{F}[2J(n)] - \tilde{F}[J(n)]] \tag{12}$$

where

$$\tilde{F}[J(n)] = \sum_{j=0}^{J(n)-1} \tilde{r}_j^2\,[J(n)]\; P\left[\tilde{d}_j[J(n)] \le \frac{f(n)}{\sigma(n)} < \tilde{d}_{j+1}[J(n)]\right] \tag{13}$$

is the error factor for unit variance random variables and the tilde sign (~) indicates reconstruction and decision levels for such variables.

6. Sequential Bit Assignment Algorithm

The following is a simple algorithm for sequential bit assignment for zero mean vectors and uniform length code words.

Step 1. Initial conditions:

B = total number of block code bits

N = block length

$\sigma^2(n)$ = component variance

$p\{f(n)\}$ = component probability density model

B_s = bit index (initially zero)

Step 2. Compute and store differential error factors

$$\tilde{D}[B(n)] \equiv \tilde{F}[2J(n)] - \tilde{F}[J(n)] \qquad (14)$$

where $J(n) = 2^{B(n)}$.

Step 3. Assign one bit to that component for which $\sigma^2(n)D[B(n)]$ is largest, increment $B(n)$ by one and increment B_s by one.

Step 4. If $B_s = B$, exit; otherwise go to step 3.

With this algorithm the differential error factors $D(\cdot)$ can be pre-computed and stored in a table of size N. The algorithm then proceeds through B sorting steps.

6. Extensions

The procedure for optimal bit assignment can be easily extended to non-uniform length component code words by evaluating the error reduction of eq.(11) for each additional level instead of each additional code bit. Huffman coding can then be applied to the coding of each component reconstruction level based upon its occupancy probability.

7. Evaluation

The mean square error coding performance of the log-variance and sequential bit assignment algorithms has been compared for one- and

two-dimensional Gaussian and Laplacian stochastic source models. It has been found that the sequential bit assignment algorithm provides a small (as much as 3%) reduction in mean square error for Gaussian sources and a somewhat larger (as much as 9%) error reduction for Laplacian sources. Measured squared error reductions of up to 8% have been achieved in image coding simulation tests using two-dimensional transform coding.

8. Summary and Conclusions

An optimal bit assignment procedure for block quantization bit assignment has been developed, and a simple, efficient algorithm has been presented for zero mean random vectors. The algorithm can easily be extended to non-zero mean random vectors, and to vectors of components with differing marginal densities.

9. Acknowledgements

Dr. Wen-hsiung Chen of Ford Aerospace and Communications Corporation in Palo Alto, California has provided valuable comments in the preparation of this paper. This research was supported in part by the Naval Ocean Systems Center, San Diego, California under Contract No. N00123-75-C-1192.

10. References

1. J.J.Y. Huang and P.M. Schultheiss, "Block Quantization of Correlated Gaussian Random Variables," IEEE Transactions on Communication Systems, Vol. CS-11, No. 3, September 1963, pp. 289-296.

2. P.J. Max, "Quantizing for Minimum Distortion," IRE Transactions on Information Theory, Vol. IT-6, No. 1, March 1960, pp. 7-12.

3. P.A. Wintz and A.J. Kurtenbach, "Waveform Error Control in PCM

Telemetry," IEEE Transactions on Information Theory, Vol. IT-14, No. 5, September 1968, pp. 650-661.

4. P.J. Ready and P.A. Wintz, "Multispectral Data Compression Through Transform Coding and Block Quantization," Purdue University, Laboratory for Applications of Remote Sensing, Information Note 050572, May 1972.

5. A. Habibi and P.A. Wintz, "Image Coding by Linear Transformation and Block Quantization," IEEE Transactions on Communication Technology, Vol. COM-19, No. 1, February 1971, pp. 50-63.

APPENDIX 3B

Spatially Adaptive Block Quantization Transform Image Coding

by

William K. Pratt
Image Processing Institute
University of Southern California
Los Angeles, California 90007

June 1977

196

Spatially Adaptive Transform Image Coding

1. Introduction

Transform image coding systems are now becoming widely accepted as a means of achieving bit rate or bandwidth compression for image transmission <1,2>. Most transform coders are designed on the basis of an a priori statistical model of the image source. Improved performance can be obtained by recourse to adaptive operation <3-5>. This paper introduces a new adaptive transform image coding concept based on an adaptive statistical model.

2. Basic Transform Image Coding Concept

The conventional design methodology for a basic, non-adaptive transform image system is described in figure 1. The operation is as follows:

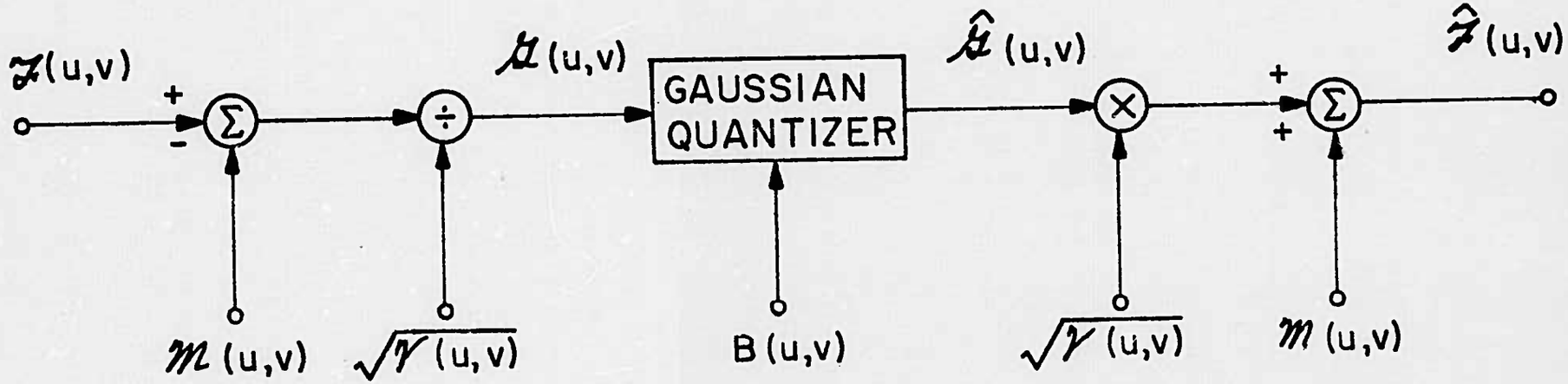
a. the image is subdivided into N x N pixel blocks F(j,k) and each block is transformed by a two-dimensional unitary transform.

b. each transform coefficient $\mathcal{F}(u,v)$ is scaled by subtraction of its estimated mean $\mathcal{M}(u,v)$ and division by its estimated standard deviation $\sqrt{\mathcal{V}(u,v)}$.

c. the scaled transform coefficients are then quantized and coded according to a mean square error process in which the number of code bits and quantization levels is chosen proportional to the transform coefficient estimated standard deviation.

d. at the receiver the decoded coefficients are rescaled and inverse transformed to reconstruct the pixel block.

A key step in this design procedure is the estimation of the transform coefficient mean and variance parameters. Normally, the image is



BASIC TRANSFORM COEFFICIENT QUANTIZATION

Figure 1.

195

8b1

modelled as a two-dimensional random process with known mean and covariance function, usually Markovian. The mean and variance of the transform coefficients are then found by a linear transformation relationship for a given unitary transform.

In practice natural images are rarely Markovian or stationary. Rather, the mean and spatial correlation changes significantly from block to block as a result of the global image structure. The spatially adaptive algorithm, described in the following sections, provides a simple means of adapting to the non-stationarity of an image field.

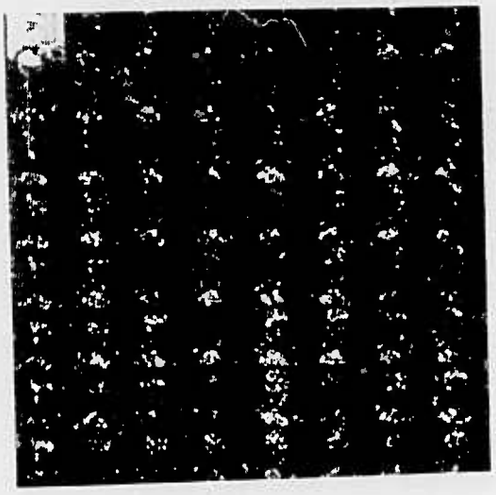
3. Transform Domain Block Structure

If a given image is a typical sample of a statistically stationary random process, then all of its transform blocks should be similar in structure. The inadequacy of the stationary model is demonstrated in the example of figure 2. Figure 2a contains a photograph of the 256 x 256 GIRL image. In figure 2b the logarithm of the magnitude of the transform coefficients are displayed for a cosine transform computed in 8 x 8 non-overlapping pixel blocks. It is observed that the amplitudes of the coefficients vary significantly from block-to-block. This characteristic is made apparent in the so-called Mandela <6> arrangement of figure 2c in which all coefficients of the same wave number index (u,v) are juxtaposed to the same block and placed according to their block index. The "d.c." coefficients corresponding to the constant valued basis function of the cosine transform form a low resolution (8 x 8 pixel average) version of the original image. If the original image is stationary, then coefficients of similar index should have about the same amplitude. This is clearly not the case. In the low wave number Mandela blocks the edge outline of the girl is clearly discernible.

4. Spatially Adaptive Transform Image Coding Algorithm

(a) Original

(b) Conventional transform display



(c) Mandela arrangement display

Figure 2. Transform domain representations of GIRL image. Cosine transform of 256 x 256 pixel image computed in 8 x 8 pixel blocks.

200

Figure 3 illustrates a simple method for exploiting the correlation between coefficients of the same wave number in spatially adjacent blocks. Each coefficient $\mathcal{F}(u,v)$ is scaled by an estimated mean $m_{m,n}(u,v)$ and standard deviation factor $[v_{m,n}(u,v)]^{\frac{1}{2}}$ obtained from a linear combination of previously reconstructed coefficients $\hat{\mathcal{F}}_{a,b}(u,v)$. Figure 4 indicates an estimate based on four neighboring coefficients in which the mean estimate

$$m_{m,n}(u,v) = \frac{1}{4}\left[\hat{\mathcal{F}}_{m-1,n-1}(u,v) + \hat{\mathcal{F}}_{m-1,n}(u,v) + \hat{\mathcal{F}}_{m-1,n+1}(u,v) + \hat{\mathcal{F}}_{m,n-1}(u,v)\right]$$

and the variance estimate

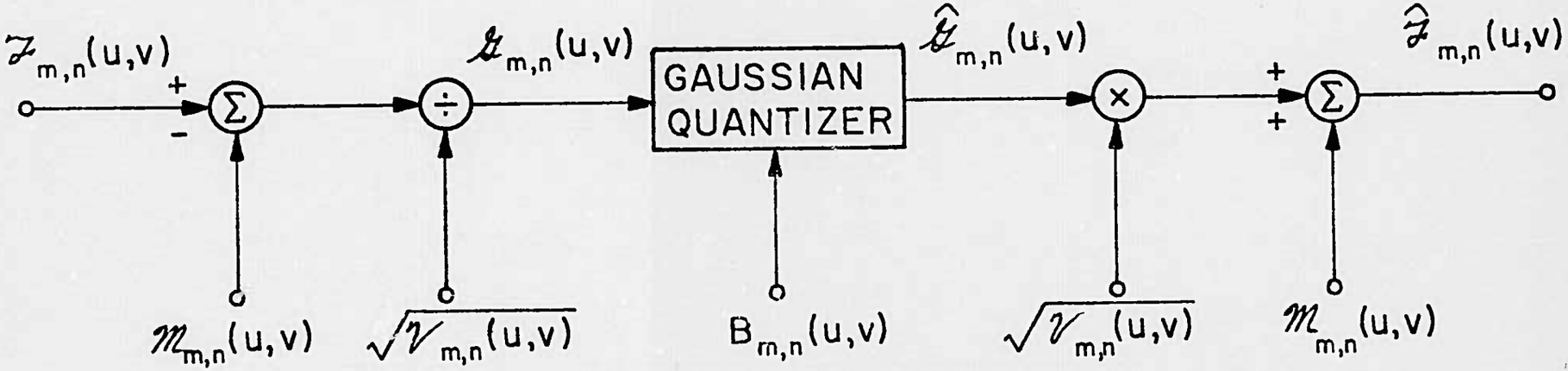
$$v_{m,n}(u,v) =$$

$$\frac{1}{4}\left[\left(\hat{\mathcal{F}}_{m-1,n-1}(u,v) - m_{m-1,n-1}(u,v)\right)^2 + \left(\hat{\mathcal{F}}_{m-1,n}(u,v) - m_{m-1,n}(u,v)\right)^2\right.$$

$$\left.+\left(\hat{\mathcal{F}}_{m-1,n+1}(u,v) - m_{m-1,n+1}(u,v)\right)^2 + \left(\hat{\mathcal{F}}_{m,n-1}(u,v) - m_{m,n-1}(u,v)\right)^2\right]$$

are obtained by simple averaging operations. Weighted linear averages can also be utilized if information is available as to the cross-correlation of neighboring coefficients. Generally, this information is not available. It should be observed that an estimate formed from neighboring coefficients, as indicated in figure 4, is similar to fourth order estimation in a linear predictive image coding system.

In the spatially adaptive coding system of figure 3 all the mean and variance coefficients of a block are estimated prior to the quantization step. The scaled coefficients are modelled as independent zero mean, unit variance Gaussian random variables. Each coefficient is quantized to $J_{m,n}(u,v)$ quantization levels using the Max <7> minimum mean square error quantization strategy. The number of quantization levels is chosen to be a binary number of the form



ADAPTIVE TRANSFORM COEFFICIENT QUANTIZATION

Figure 3.

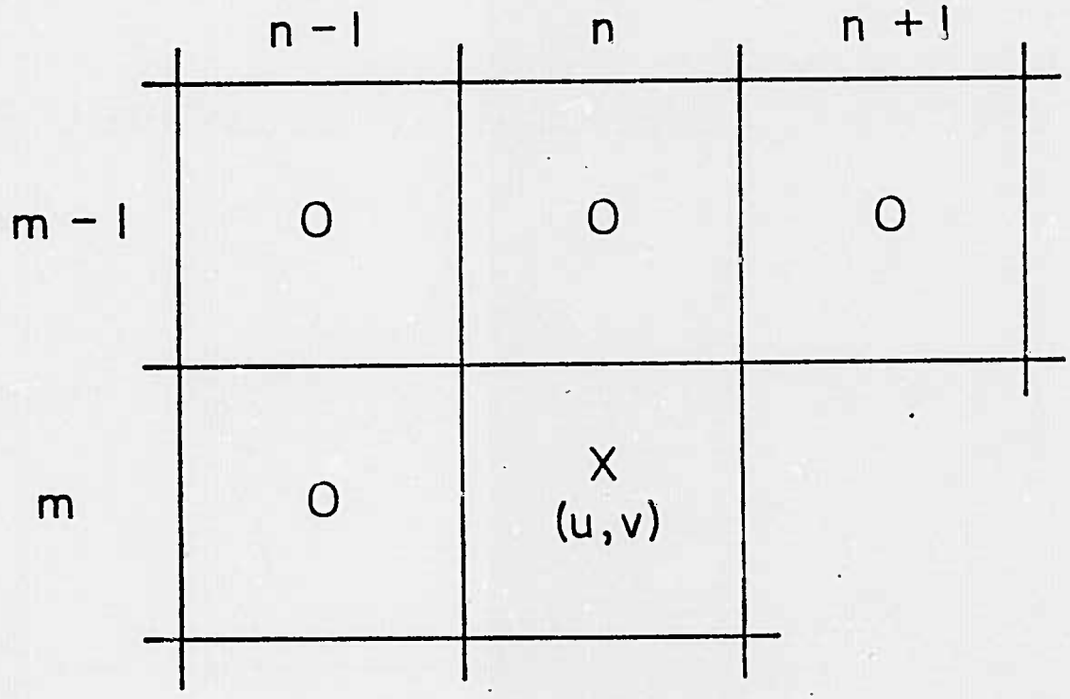


Figure 4.

$$J_{m,n}(u,v) = 2^{B_{m,n}(u,v)}$$

where $B_{m,n}(u,v)$ is the number of code bits assigned to the coefficient. The bit allocation is determined by a minimum mean square block quantization bit assignment algorithm <8>. It should be noted that the bit assignment will be variable from block-to-block.

The spatially adaptive algorithm possesses several advantages. It is relatively simple to implement requiring only a small amount of computation for the mean and variance estimates and only 2N lines of storage. Another, important feature is that the algorithm is self-adaptive. The mean and variance estimates are based on previously reconstructed coefficients at both the transmitter and receiver. Therefore, no accounting information need be transmitted. The algorithm, however, is sensitive to channel errors because of its feedback nature.

5. Experimental Results

A series of simulation experiments has been performed to evaluate the spatially adaptive transform image coding algorithm. Figure 5 contains an original 512 x 512 pixel image and reconstructions for cosine transform coding in blocks of 16 x 16 pixels for non-adaptive and adaptive operation. The adaptive algorithm results in better edge rendition and lower granularity in background areas.

6. Summary

A new algorithm for adaptive transform image coding has been developed. The algorithm utilizes the global correlation between transform coefficients of neighboring blocks to estimate the mean and variance of each transform coefficient. These statistical estimates are then used to optimally quantize the coefficients. Experimental results indicate a discernible improvement in image quality.

(a) Original

(b) Non-adaptive coding

(c) Adaptive coding

Figure 5. Examples of non-adaptive and adaptive block quantization transform image coding.

References

1. P.A. Wintz, "Transform Picture Coding," Proceedings IEEE, Vol. 60, No.7, July 1972, pp. 809-823.

2. W.K. Pratt, Digital Image Processing, Wiley-Interscience, New York, 1978.

3. H.C. Andrews and A.G. Tescher, "The Role of Adaptive Phase Coding in Two- and Three-Dimensional Fourier and Walsh Image Compression," Proceedings Walsh Function Symposium, Washington D.C., March 1974.

4. M. Tasto and P.A. Wintz, "Image Coding by Adaptive Block Quantization," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, December 1971, pp. 957-972.

5. W. Chen, "Adaptive Coding of Color Images Using Cosine Transform," Proceedings International Communications Conference, Philadelphia, June 1976, pp. 47-7 to 47-13.

6. J. Kajiya, "Group Representations and the Modeling of Visual Perception," Proceedings Symposium on Current Mathematical Problems in Image Science, Monterey, California, November 1976, pp. 67-70.

7. J. Max, "Quantizing for Minimum Distortion," IRE Transactions on Information Theory, Vol. IT-6, March 1960, pp. 7-12.

8. W.K. Pratt, "Block Quantization Bit Assignment," submitted for publication, April 1977.

4. Interframe Transform Image Coding

It has long been realized that the transform image coding concept can be exploited to reduce temporal redundancy between television frames as well as spatial redundancy within a frame. Exploitation, of the third dimension, however has been severely limited by the massive amount of image data that must be processed. During the contract several methods of interframe tranform coding have been investigated in detail. This research effort has culminated in a design study of two interframe transform coding techniques. The first, a baseline system, involves three-dimensional cosine transform coding in small blocks on the order of 16 x 16 x 16 pixels. The second system utilizes two-dimensional cosine transform coding in spatial blocks and DPCM coding between frames. Non-adaptive and adaptive implementations of these systems have been studied. Detailed research results are described in the USC report:

J.A. Roese
Interframe Coding of Digital Images Using Transform and Hybrid Transform/Predictive Techniques
University of Southern California
Image Processing Institute
USCIPI Report 700
June 1976

Appendix 4A contains a paper, to be published in IEEE Communications Transactions in November 1977, that summarizes the research effort.

APPENDIX 4A

Interframe Cosine Transform Image Coding

by

John A. Roese
Naval Ocean Systems Center
San Diego, California 92152

William K. Pratt
Image Processing Institute
University of Southern California
Los Angeles, California 90007

Guner S. Robinson
Northrop Research and Technology Center
Hawthorne, California 90250

April 1977

Interframe Cosine Transform Image Coding

Abstract

Two-dimensional transform coding and hybrid transform/DPCM coding techniques have been investigated extensively for image coding. This paper presents a theoretical and experimental extension of these techniques to the coding of sequences of correlated image frames. Two coding methods are analyzed: three-dimensional Cosine transform coding; and two-dimensional Cosine transform coding within an image frame combined with DPCM coding between frames. Theoretical performance estimates are developed for the coding of Markovian image sources. Simulation results are presented for transmission over error-free and binary symmetric channels.

1. Introduction

During the past decade there has been extensive research and development directed toward two-dimensional image coding systems based upon transform and linear predictive coding techniques <1-5>. These coding methods utilize the spatial redundancy within an image field to achieve efficient quantization and coding. In this paper the concept is extended to the coding of sequences of correlated image frames in order to exploit the temporal redundancy of television imagery.

2. Interframe Transform Coding

The basic concept of three-dimensional transform coding is illustrated in figure 1 <6>. A sequence of image frames $F(j,k,\ell)$ undergoes a three-dimensional transform in blocks of J x K x L pixels according to the general formula

$$\mathcal{F}(u,v,w) = \sum_{j=0}^{J-1}\sum_{k=0}^{K-1}\sum_{\ell=0}^{L-1} F(j,k,\ell)\,A(u,v,w;j,k,\ell) \qquad (1)$$

to produce a sequence of transform planes $\mathcal{F}(u,v,w)$ where $A(u,v,w;j,k,\ell)$ represents a unitary transform kernel, and (J,K,L) denote the row, column, and interplane indicies, respectively. The transform coefficients are then quantized and coded for transmission. Zonal sampling or zonal coding quantization and coding strategies are usually employed. In zonal sampling transform coefficients with the greatest expected energy are selected for transmission, and th remainder are discarded. Zonal coding entails the quantization of transform coefficients with the number of quantization levels chosen

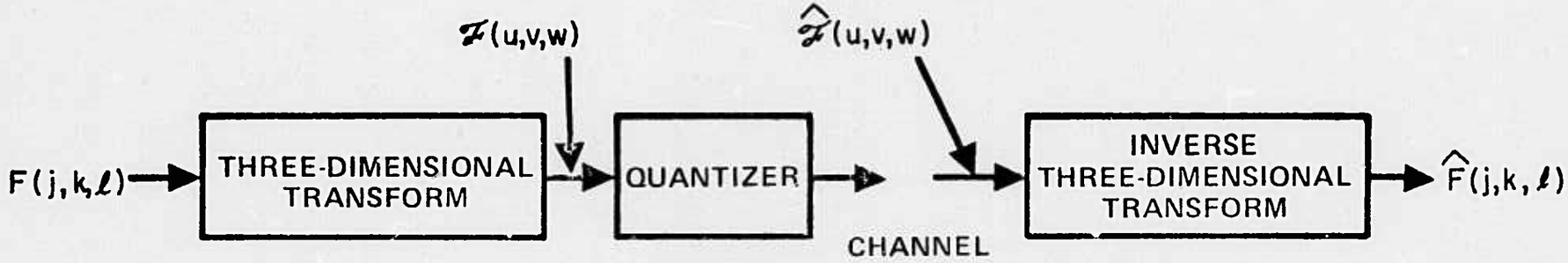


Figure 1. Three-dimensional transform coder.

to minimize the total mean square quantization error. At the receiving unit the quantized transform coefficients $\hat{\mathcal{F}}(u,v,w)$ are inverse transformed to produce the reconstructed frame sequence

$$\hat{F}(j,k,\ell) = \sum_{u=0}^{J-1}\sum_{v=0}^{K-1}\sum_{w=0}^{L-1}\hat{\mathcal{F}}(u,v,w)\, A^{-1}(u,v,w;j,k,\ell) \qquad (2)$$

where $A^{-1}(u,v,w;j,k,\ell)$ is the inverse unitary transform kernel.

Many different types of unitary transforms, including the Fourier, sine, cosine, Hadamard, slant, and Karhunen-Loeve transforms, have been investigated for transform image coding. One of the most attractive from the standpoint of coding performance and implementation simplicity is the discrete cosine transform defined by the separable kernel

$$A(u,v,w;j,k,\ell) = C(u,j)\, C(v,k)\, C(w,\ell) \qquad (3a)$$

where

$$C(u,j) = \frac{1}{\sqrt{J}} \qquad u = 0 \qquad (3b)$$

$$C(u,j) = \sqrt{\frac{2}{J}}\, \cos\left[\frac{(2j+1)u\pi}{2J}\right] \qquad u > 0 \qquad (3c)$$

It has been shown that the theoretical mean square error coding performance of the cosine transform is nearly equivalent to that obtained by the optimal Karhunen-Loeve transform for Markovian image sources <7>. Furthermore, the cosine transform can be efficiently computed by transversal filtering methods <8>. For these reasons the cosine transform has been adopted for the analysis of interframe coding techniques.

A bandwidth reduction is achieved in transform coding by quantizing and coding transform coefficients according to their expected energy based upon a statistical model of coefficient variances. The conventional model assumes pixels to be samples of a separable Markov process. The correlation matrix along each image column is given by

$$\underline{K}_{f_j} = \sigma_j^2 \begin{bmatrix} 1 & \rho_c & \rho_c^2 & \cdots & \rho_c^{J-1} \\ \rho_c & 1 & \rho_c & \cdots & \rho_c^{J-2} \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \rho_c^{J-1} & \cdot & \cdot & \cdots & 1 \end{bmatrix} \tag{4}$$

where σ_j^2 represents the pixel variance and ρ_c is the adjacent pixel correlation. Similar expressions exist for the row and temporal correlation matrices. As a result of the separability of the cosine transform the covariance matrices of the coefficients can be obtained from

$$\underline{K}_{f_u} = \underline{C}\,\underline{K}_{f_j}\underline{C}^T \tag{5a}$$

$$\underline{K}_{f_v} = \underline{C}\,\underline{K}_{f_k}\underline{C}^T \tag{5b}$$

$$\underline{K}_{f_w} = \underline{C}\,\underline{K}_{f_\ell}\underline{C}^T \tag{5c}$$

where $\underline{C}$ represents the cosine transform kernel C(j,u) in matrix form and $\underline{K}_{f_u}$ etc. are the transform coefficient column, row, and temporal covariance matrices. Column, row, and temporal variance vectors $\underline{v}_{f_u}$, etc. can then be obtained by extracting diagonal elements of the

corresponding covariance matrices. Finally, a three-dimensional variance array is formed by the product

$$V_{\mathcal{F}}(u,v,w) = v_{f_u}(u)\, v_{f_v}(v)\, v_{f_w}(w) \tag{6}$$

This array models the variance of each transform at coordinate (u,v,w).

For the zonal sampling form of transform coding the coefficient selection process can be conveniently represented by a three-dimensional transform domain selection operator $\mathcal{A}(u,v,w)$. The reconstructed image can then be expressed as

$$\hat{F}(j,k,\ell) = \sum_{u=0}^{J-1} \sum_{v=0}^{K-1} \sum_{w=0}^{L-1} \mathcal{F}(u,v,w)\, \mathcal{A}(u,v,w)\, A^{-1}(u,v,w;j,k,\ell) \tag{7}$$

where $\mathcal{A}(u,v,w)$ assumes the value of unity for selected transform coefficients and zero for discarded coefficients. Figure 2 illustrates coefficient selection. The resultant mean square coding error for zonal sampling is given by <9>

$$\mathcal{E} = E\{(F-\hat{F})^2\} = \sum_{u=0}^{J-1} \sum_{v=0}^{K-1} \sum_{w=0}^{L-1} E\{\mathcal{F}^2(u,v,w)\}\,[1-\mathcal{A}(u,v,w)] \tag{8}$$

The optimal coding strategy is to order the transform coefficients on the basis of largest variance as specified by eq.(6). Figure 3 contains an evaluation of the zonal sampling mean square error for an adjacent pixel correlation factor of $\rho = 0.96$ for a 32:1 sample reduction as a function of spatial block size and frame size. The curve for L = 1 represents the performance of a two-dimensional zonal sampling transform coder.

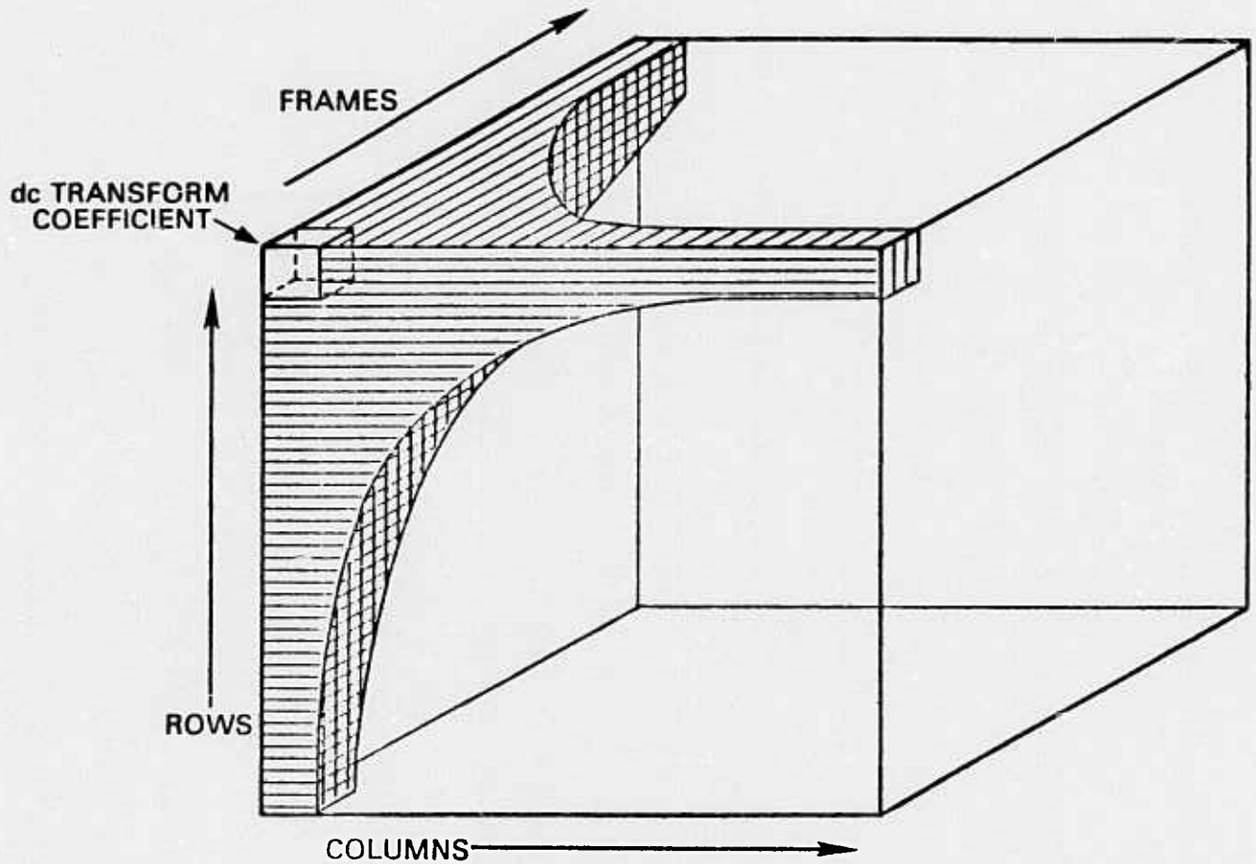


Figure 2. Three-dimensional array of selected transform coefficients.

212

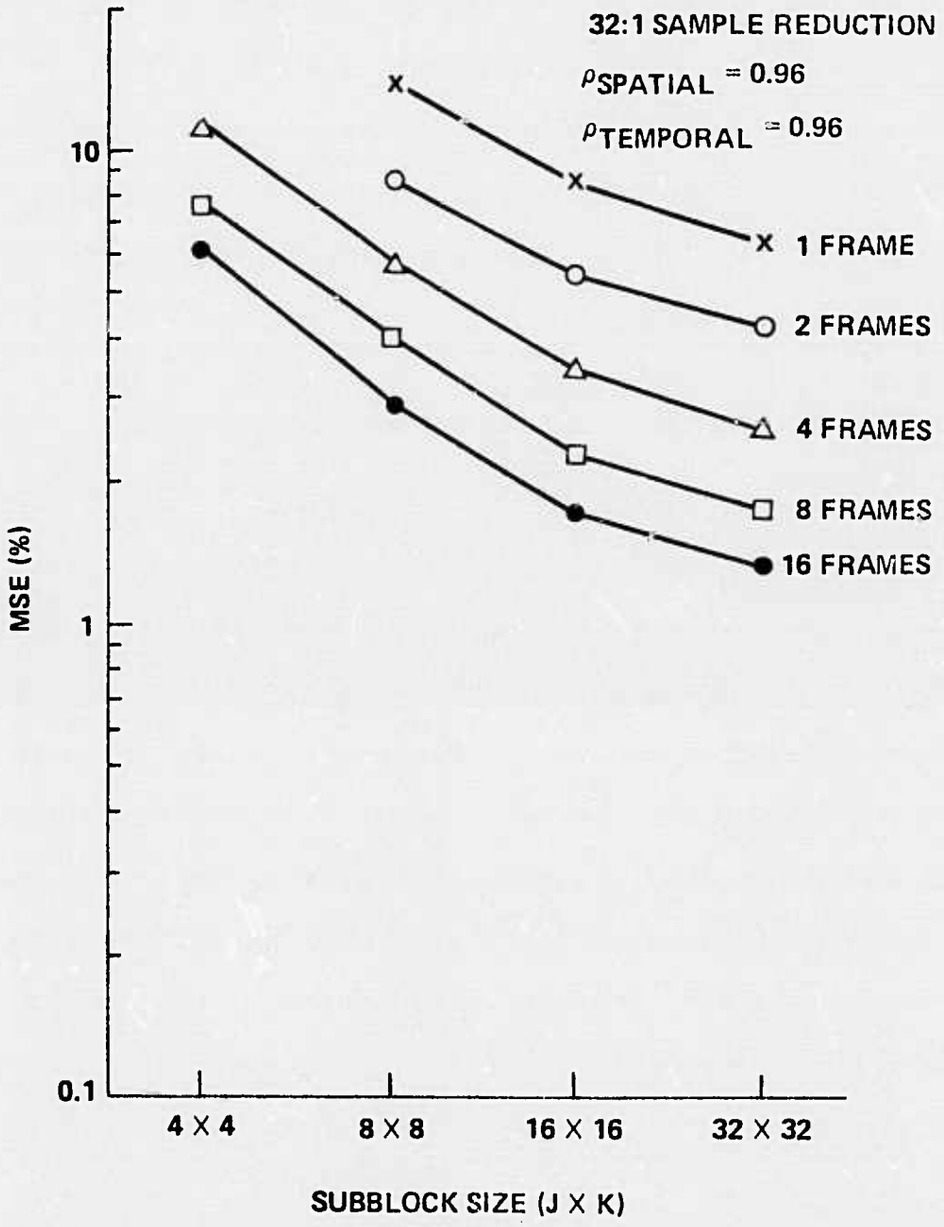


Figure 3. Theoretical performance evaluation for three-dimensional cosine transform coder with zonal sampling for Markov process data source.

In the zonal coding process the number of quantization levels assigned to each coefficient is commonly set proportional to the logarithm of the coefficient variance; the coefficient quantization and decision levels are nonlinearly spaced to minimize the mean square quantization error according to a probability density model for the coefficients <9>. Those coefficients receiving no quantization levels are discarded as in the zonal sampling process. The resulting mean square error expression for a constant length binary code of each coefficient b(u,v,w) is given by

$$\mathcal{E} = \sum_{u=0}^{J-1} \sum_{v=0}^{K-1} \sum_{w=0}^{L-1} \left[E\{\mathcal{F}^2(u,v,w)\} - \sum_{n=1}^{2^{b(u,v,w)}} R_n^2(u,v,w)\, P[R_n(u,v,w)] \right] \qquad (9)$$

where $R_n(u,v,w)$ represents the n-th quantization level of a coefficient, $P(\cdot)$ is the occupancy probability of the n-th quantization level, and b(u,v,w) denotes the number of bits assigned to a coefficient. The probability density of the "d.c." coefficient $\mathcal{F}(0,0,0)$ is usually modelled as a Rayleigh density, while the remaining "a.c." coefficients are modelled as Gaussian densities with variances specified by eq.(6). Figure 4 illustrates the mean square error zonal coding performance for a separable Markov process image source with $\rho = 0.96$ and a 32:1 bit rate reduction, i.e. coding from 8 to 0.25 bits/pixel. The relative efficiency between zonal sampling and zonal coding is evident from a comparison of figures 3 and 4.

3. Interframe Hybrid Transform/DPCM Coding

In 1974 Habibi introduced the concept of hybrid transform/DPCM image coding in which a unitary transform is taken along an image row

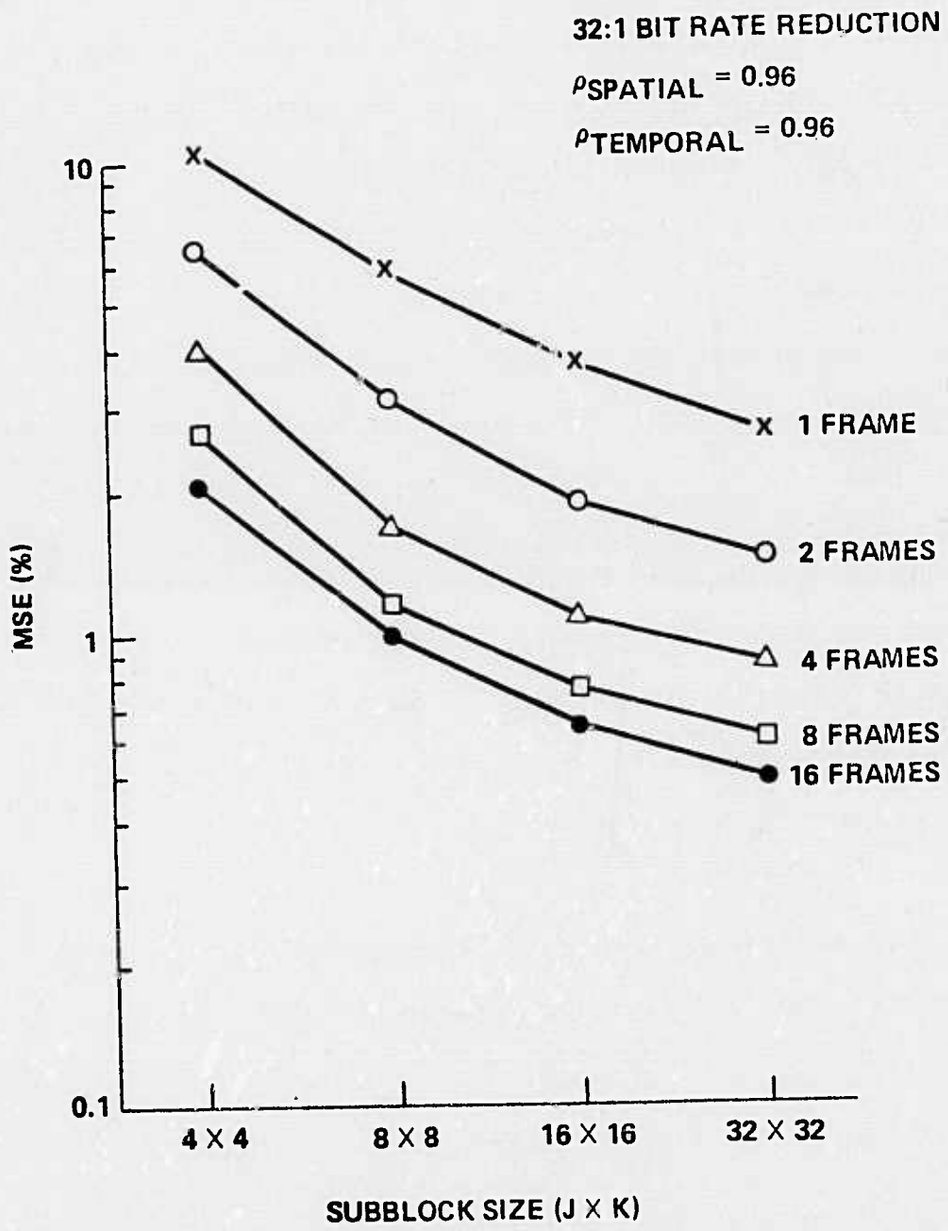


Figure 4. Theoretical performance evaluation for three-dimensional cosine transform coder with zonal coding for Markov process data source.

and differential pulse code modulation (DPCM) coding is performed on the transform coefficients along each column <10>. This concept can be easily extended to three-dimensional image sources <10-12>.

A block diagram of the basic interframe hybrid coder is shown in figure 5. In this coding system a two-dimensional unitary transform is performed on each partition or spatial subblock of the image. One of the bank of parallel DPCM linear predictive coders is then applied to each set of transform coefficients in the temporal direction. The resulting sequences of transform coefficient differences are quantized and coded for transmission. Image reconstruction occurs at the receiver where the transform coefficient differences are decoded and a replica of each transmitted image is reconstructed by a two-dimensional inverse transformation.

The three-dimensional array of elements obtained by a two-dimensional cosine transform in J x K pixel blocks may be expressed in general form as

$$\mathcal{H}(u,v,\ell) = \sum_{j=0}^{J-1} \sum_{k=0}^{K-1} F(j,k,\ell)\, C(u,j)\, C(v,k) \qquad (10)$$

where C(u,j) represents the cosine transform kernel. At each spatial frequency (u,v) the DPCM coder quantizes and codes the difference signal

$$\mathcal{D}(u,v,\ell) = \mathcal{H}(u,v,\ell) - \hat{\mathcal{H}}(u,v,\ell) \qquad (11)$$

where $\hat{\mathcal{H}}(u,v,\ell)$ denotes the predicted value of $\mathcal{H}(u,v,\ell)$. The difference signal $\mathcal{D}(u,v,\ell)$ is coded using zonal coding strategy similar to that of transform coding in which the number of code bits

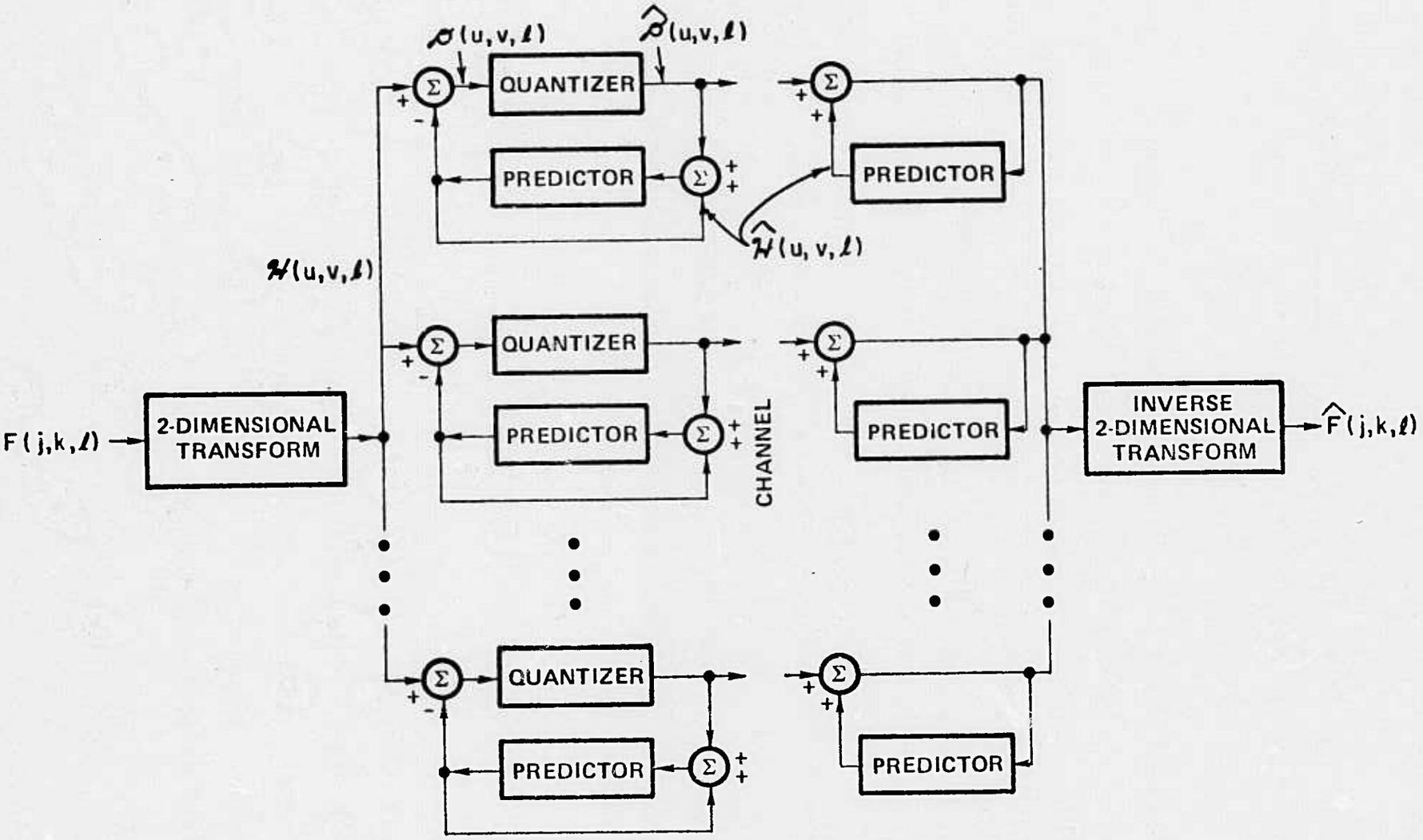


Figure 5. Hybrid two-dimensional transform/DPCM coder.

220

assigned to each difference signal is set proportional to the logarithm of its variance <4>

$$V_{\mathcal{K}}(u,v) = v_f(u)\, v_f(v)\, [1-\rho_T^2] \tag{12}$$

where ρ_T represents the temporal correlation factor. Quantization levels are usually set according to a Laplacian model of each difference signal. The mean square error expression is given by

$$\delta = \sum_{u=0}^{J-1} \sum_{v=0}^{K-1} \left[V_{\mathcal{K}}(u,v) - \sum_{n=1}^{2^{b(u,v)}} R_n^2(u,v)\, P(R_n(u,v)) \right] \tag{13}$$

where $R_n(u,v)$ represents the n-th reconstruction level of the quantized difference signal and $b(u,v)$ denotes the number of bits assigned at each spatial frequency coordinate. Figure 6 contains a plot of mean square error versus block size for a hybrid cosine transform/DPCM coder for a Markov process source with $\rho = 0.96$. A comparison of the theoretical performance of the hybrid and three-dimensional transform coders is shown in figure 7.

4. Adaptive Interframe Coding

Significant improvement in coding performance can be obtained by spatially adapting the hybrid/DPCM interframe coder to use calculated temporal statistical measures of the transform coefficient temporal difference signal $\mathcal{K}(u,v,\ell)$ in each sublock. The statistical measures utilized are the mean and correlation for each temporal sequence of transform coefficients defined as

$$\bar{\mathcal{K}}(u,v) = \frac{1}{J} \sum_{\ell=0}^{L-1} \mathcal{K}(u,v,\ell) \tag{14a}$$

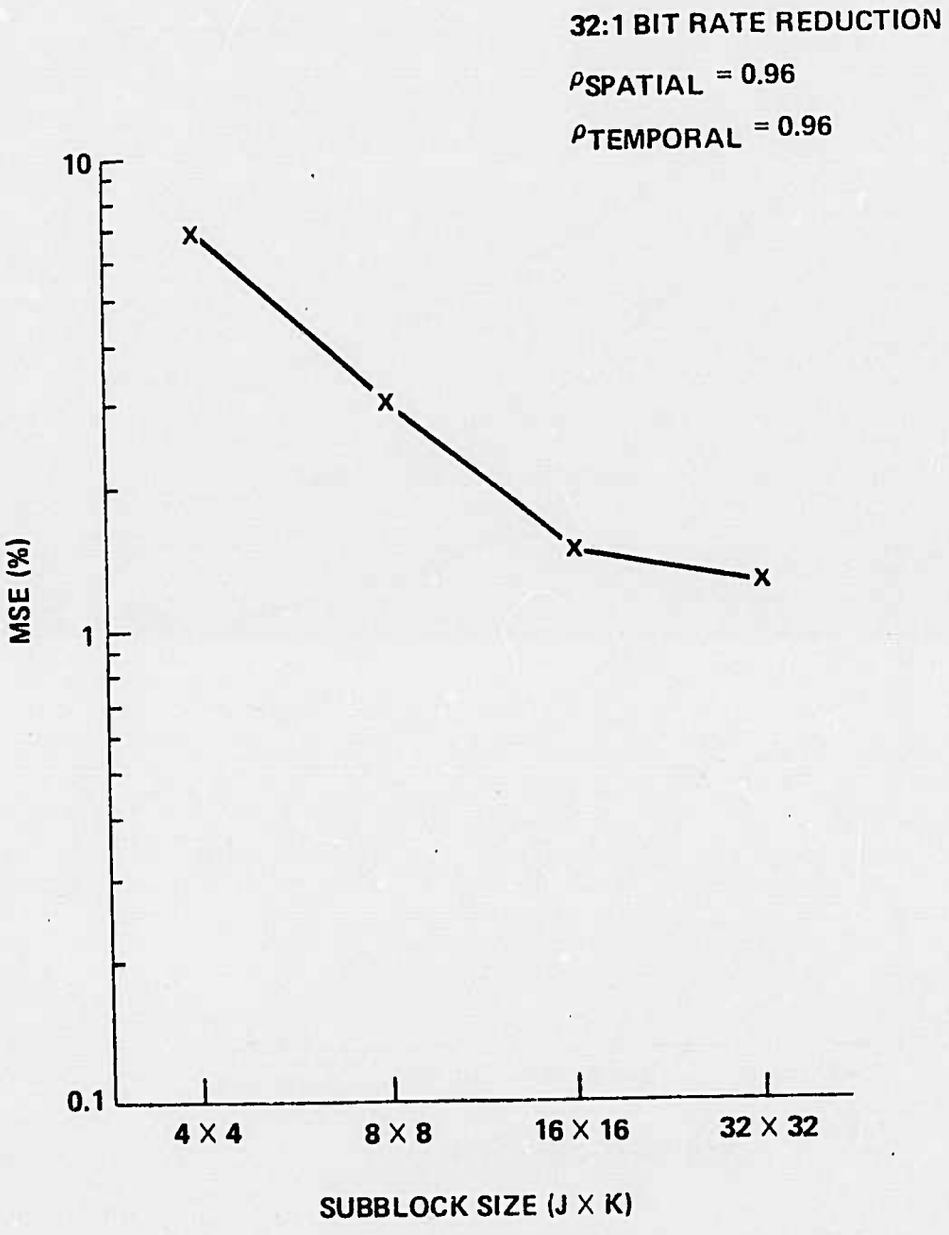


Figure 6. Theoretical performance evaluation for hybrid CCD coder with zonal coding for Markov process data source.

222

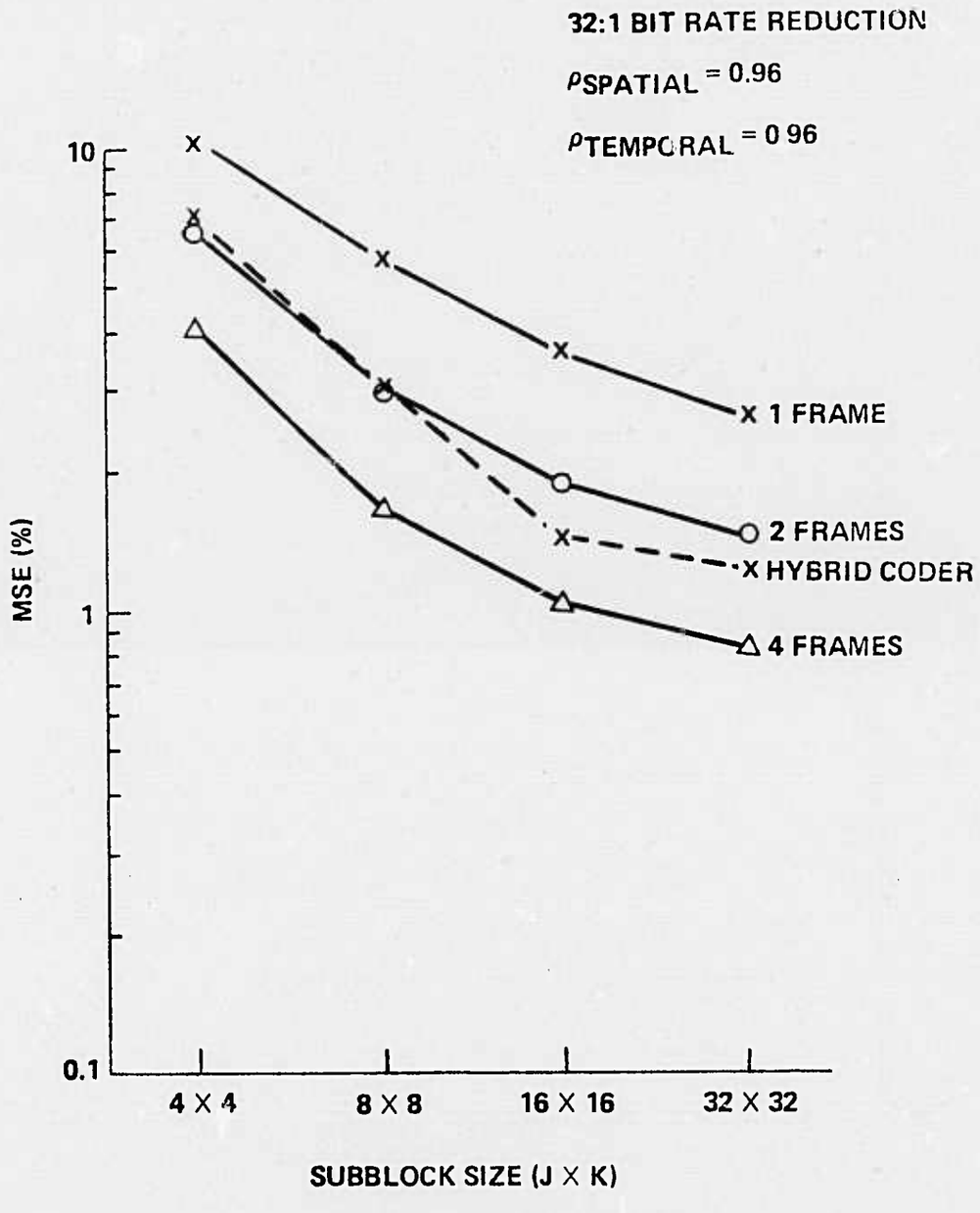


Figure 7. Theoretical performance comparison of three-dimensional cosine transform coder and hybrid two-dimensional transform/DPCM coder for Markov process data source.

and

$$\rho(u,v) = \frac{\frac{1}{L-1}\sum_{\ell=1}^{L-1} \mathcal{X}(u,v,\ell)\mathcal{X}(u,v,\ell-1)}{\frac{1}{L}\sum_{\ell=0}^{L-1} \mathcal{X}^2(u,v,\ell)} \qquad (14b)$$

The variance of the temporal coefficient difference is then estimated as

$$\sigma_D^2(u,v) = \frac{1}{L-1}\sum_{\ell=1}^{L-1} \tilde{\mathcal{X}}_D^2(u,v,\ell) - \left[\frac{1}{L-1}\sum_{\ell=1}^{L-1} \tilde{\mathcal{X}}_D(u,v,\ell)\right]^2 \qquad (15a)$$

where

$$\tilde{\mathcal{X}}_D(u,v,\ell) = \mathcal{X}(u,v,\ell) - \rho(u,v)\mathcal{X}(u,v,\ell-1) \qquad (15b)$$

for $\ell = 1,2,\ldots,L-1$. For each sequence of transform coefficients the temporal correlation $\rho(u,v)$ is the gain coefficient in the DPCM predictor feedback loop. The transform coefficient mean $\overline{\mathcal{X}}(u,v)$ provides biasing to achieve a zero-mean input sequence of transform coefficients. The computed transform coefficient difference variances $\sigma_D^2(u,v)$ are used to generate the two-dimensional subblock bit assignment. Thus, the resulting bit assignment, predictor feedback loop gain coefficients, and associated biasing and scaling factors are, in general, different for different subblocks. Although the subblock bit assignment may vary between subblocks, the total number of bits available for coding each subblock is constrained to be equal. Local adaptation to the measured statistics of each subblock will normally produce improved coding results when compared with

non-adaptive implementations. However, adaptation does result in increased coder complexity.

In the adaptive coding process described above the receiver must have available the three statistical parameters $\bar{x}(u,v)$, $\rho(u,v)$, and $\sigma_D^2(u,v)$ at each coefficient index (u,v). One approach is to form these statistics at the transmitter for L stored frames, and then quantize, code, and transmit each statistic to the receiver. The disadvantages of this approach are the data overhead and the effects of quantization error introduced in the transmission of each statistic. A better approach is to form the statistics jointly at the transmitter and receiver utilizing the coefficient feedback predicted value $\hat{x}(u,v,\ell)$ in eqs.(14) and (15). This scheme requires no channel overhead since the transmitter and receiver coefficient predictions are identical in the absence of channel errors. Furthermore, quantization error of the statistics is reduced by the DPCM feedback process. With either approach it is necessary to occasionally re-transmit the feedback prediction $\hat{x}(u,v,\ell)$ to correct for accumulated channel error effects.

The concept of temporal/spatial adaptation described above for the hybrid transform/DPCM interframe coding is not directly applicable to interframe three-dimensional transform coding since temporal averages are meaningless for a three dimensional coder. There are, however, a number of other means of adaptively computing subcube bit assignments for a three-dimensional transform coder based upon coefficient energy distribution, object motion, or edge structure

occuring within a subcube. Such methods have not been explored in this paper.

5. Experimental Results

An extensive set of computer simulations has been performed to experimentally evaluate the performance of interframe transform coders. The hybrid coders considered are parametric in many variables including choice of separable transforms, zonal sampling or zonal coding with various quantization strategies, presence or absence of channel error, spatial subblock size, and average pixel bit rate. In the evaluation of coder performance levels the normalized mean square error (NMSE) and signal-to-noise ratio criteria (SNR) are employed in conjunction with subjective visual evaluations. These error criteria are defined as

$$\mathrm{NMSE} = \frac{\sum_{j=0}^{J-1}\sum_{k=0}^{K-1}[F(j,k,\ell) - \hat{F}(j,k,\ell)]^2}{\sum_{j=0}^{J-1}\sum_{k=0}^{K-1}[F(j,k,\ell)]^2} \tag{16}$$

$$\mathrm{SNR} = -10 \log_{10}\left\{\frac{\frac{1}{JK}\sum_{j=0}^{J-1}\sum_{k=0}^{K-1}[F(j,k,\ell) - \hat{F}(j,k,\ell)]^2}{Y_M^2}\right\} \tag{17}$$

where F and $\hat{F}$ are the original and coded images, and Y_M represents the maximum luminance value of F, typically 255. The source data for the

computer simulations consists of two sets of 16 images digitized from sequential frames of a 24 frame-per-second motion picture. One sequence, called "Walter," contains images from a fixed position camera of a moving subject engaged in conversation. The other contains images of a chemical plant photographed from an airplane in a flyby trajectory. Each frame is digitized at a spatial resolution of 256 x 256 pixels with each pixel amplitude linearly quantized to 256 levels. Photographs of the sixteenth frame of each data sequence are presented in figure 8.

Figure 9 contains photographs of image reconstructions of the sixteenth frame of an image sequence for three-dimensional transform coding in 16 x 16 x 16 pixel cubes. Coding is performed at average rates of 1.0, 0.5, 0.25, and 0.1 bits/pixel/frame. Normalized mean square measurements are indicated in the figure. Subjectively, coding errors are not noticeable for coding down to 0.5 bits. Beyond this value some blotchiness becomes apparent.

Photographs of the image reconstruction of the sixteenth frame of the sequence for hybrid transform/DPCM coding system at average coding rates of 0.1, 0.25, 0.5, and 1.0 bits/pixel are shown in figure 10 and 11 for non-adaptive and adaptive coding, respectively. In these experiments the first of the sixteen frames is available at both the transmitter and receiver as an initial condition. The transform coefficient statistics in the adaptive coder simulation have been generated from the common feedback predicted coefficient $\hat{x}(u,v,t)$ available at both the transmitter and receiver, and therefore, there

(a) Walter

(b) chemical plant

Figure 8. Sixteenth frame of original sequences of test images.

(a) 1.0 bits/pixel/frame
NMSE = 0.163%

(b) 0.5 bits/pixel/frame
NMSE = 0.299%

(c) 0.25 bits/pixel/frame
NMSE = 0.491%

(d) 0.1 bits/pixel/frame
NMSE = 0.882%

Figure 9. Experimental coding performance of the three-dimensional cosine transform coder.

(a) 1.0 bits/pixel/frame
NMSE = 1.026%

(b) 0.5 bits/pixel/frame
NMSE = 1.227%

(c) 0.25 bits/pixel/frame
NMSE = 1.601%

(d) 0.1 bits/pixel/frame
NMSE = 2.634%

Figure 10. Experimental coding performance of the hybrid transform/DPCM coder without adaptive coding.

(a) 1.0 bits/pixel/frame
NMSE = 0.022%

(b) 0.5 bits/pixel/frame
NMSE = 0.067%

(c) 0.25 bits/pixel/frame
NMSE = 0.186%

(d) 0.1 bits/pixel/frame
NMSE = 0.625%

Figure 11. Experimental coding performance of the hybrid transform/DPCM coder with adaptive coding.

is no channel overhead for this form of adaptive coding. Visually, no image degradation can be seen for bit rates as low as 0.5 bits/pixel/frame. Some artifacting effects due to the 16 x 16 subblock partitioning of the images is apparent at the 0.25 bits/pixel/frame rate. Also, regions outlining the subject's head begin to show degradation at this bit rate because of head motion and the relatively few coefficients assigned to transmit high frequency coefficients. The observed image degradations are similar in nature, but more pronounced at the 0.1 bits/pixel/frame bit rate. The sixteen reconstructed frames produced by the coder have been displayed on a television monitor for real time subjective analysis. No error build-up has been perceived in the display.

Figure 12 illustrates NMSE and SNR measures as a function of frame number for the adaptive hybrid transform/DPCM coder implementation at average pixel bit rates from 0.1 to 1.0 bits/pixel/frame with 16 x 16 subblocks. These graphs indicate that, even at the lowest bit rate, stability in coder performance is achieved within the first eight frames. Perfomance stability occurs much earlier in the frame sequence at the higher bit rates. In these simulations the first frame is assumed available at the receiver, and consequently, zero NMSE is obtained for the first frame.

Performance of the adaptive hybrid transform/DPCM coder in the presence of noise has been investigated by computer simulation of a binary symmetric channel transmission system. The channel operates on each binary digit independently, changing each digit from 0 to 1 or

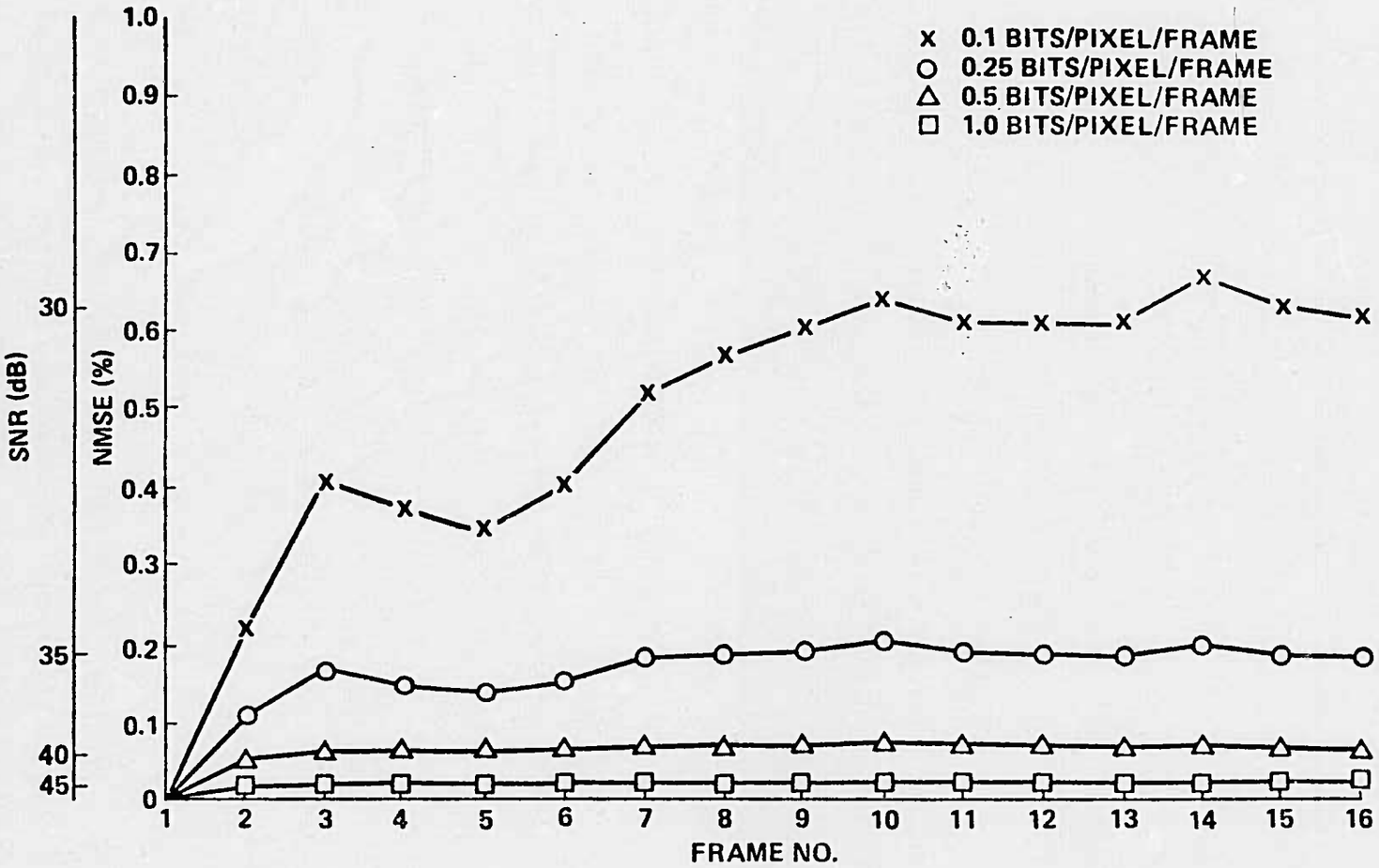


Figure 12. Coding performance as a function of frame number for the hybrid CCD coder.

from 1 to 0 with probability P and leaving the digit unchanged with probability 1-P. At the receiver the encoded picture is reconstructed from the string of binary digits, including errors, transmitted across the channel.

Photographs showing the visual effects of channel noise on the sixteenth frame of the data base are given in figure 13. The results illustrated are for average pixel bit rates of 1.0 and 0.25 bits/pixel, and P of 10^{-2} and 10^{-3}. Simulations have also been run with a channel error probability of 10^{-4}. Coding results obtained with $P = 10^{-4}$ are essentially indistinguishable from the case with $P = 0$, and therefore, are not included.

Figure 14 compares the coding results of intraframe and interframe transform/DPCM coders for the two image data bases. The average pixel bit rates used give subjectively equivalent image reconstructions although the nature of the image degradations differs for the two coders. For the "Walter" data base, exploitation of temporal correlation results in an 8:1 reduction average pixel bit rate. Comparison of coding performance using the "chemical plant" data base shows an average pixel bit rate improvement of 4:1. Since performance of the hybrid interframe coder is dependent on temporal correlation, reduced levels of performance are to be anticipated for image sequences distorted by the effects of camera motion.

6. Summary

(a) 1.0 bits/pixel, $P = 10^{-3}$
NMSE = 0.043%

(b) 1.0 bits/pixel, $P = 10^{-2}$
NMSE = 0.282%

(c) 0.25 bits/pixel, $P = 10^{-3}$
NMSE = 0.222%

(d) 0.25 bits/pixel, $P = 10^{-2}$
NMSE = 0.450%

Figure 13. Experimental coding performance of the hybrid transform/DPCM coder with adaptive coding and with channel error.

(a) 2.0 bits/pixel

(b) 0.8 bits/pixel

Hybrid Intraframe Coder

(c) 0.25 bits/pixel/frame

(d) 0.2 bits/pixel/frame

Hybrid Interframe Coder

Figure 14. Comparison of hybrid intraframe and hybrid interframe cosine transform/DPCM coders with adaptive coding.

Based on the theoretical and experimental results obtained, it has been demonstrated that exploitation of temporal as well as spatial correlations is a viable technique for coding digital image sequences. In addition theoretical performance levels predicted for the interframe transform and hybrid coders have been substantiated experimentally for two test sequences of image data. Supplementary studies of alternative operational modes and image motion compensation effects are described in reference <13>.

7. Acknowledgement

This research was supported by the Advanced Research Projects Agency of the Department of Defense and monitored by the Wright Patterson Air Force Base under Contract No. F-33615-76-C-1203 and by the Naval Undersea Center, San Diego, California, under Contract No. N00123-75-C-1192.

The authors wish to acknowledge the contributions of Dr. Ali Habibi of TRW Systems in the initial stages of research leading to this paper while Dr. Habibi was a staff member of the University of Southern California Image Processing Institute. Also, the authors are grateful to Dr. Clifford Reader of Ford Aerospace Corporation of Palo Alto California for his assistance in photographic recording of images.

8. References

1. A. Habibi and G.S. Robinson, "A Survey of Digital Picture

Coding," IEEE Computer, Vol. 7, No. 5, May 1974, pp. 22-34.

2. Special Issue on Redundancy Reduction, Proceedings IEEE, Vol. 55, No. 3, March 1967.

3. Special Issue on Digital Picture Processing, Proceedings IEEE, Vol. 6, No. 7, July 1972.

4. A. Habibi and P.A. Wintz, "Image Coding by Linear Transformation and Block Quantization," IEEE Transactions on Communication Technology, Vol. COM-19, No. , February 1971, pp. 50-62.

5. A. Habibi, "Comparison on n-th Order DPCM Encoder with Linear Transformations and Block Quantization Techniques," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, December 1971, pp. 948-956.

6. W.K. Pratt, W.H. Chen, and L.R. Welch, "Slant Transform Image Coding," IEEE Transactions on Communications, Vol. COM-22, No. 8, August 1974, pp. 1075-1093.

7. N. Ahmed, T. Natarajan, and K. R. Rao, "On Image Processing and a Discrete Cosine Transform," IEEE Transactions on Computers, Vol. C-23,No. 1, January 1974, pp. 90-93.

8. R. W. Means, H. J. Whitehouse, and J. M. Speiser, "Television Encoding Using a Hybrid Discrete Cosine Transform and a Differential Pulse Code Modulator in Real Time," Proceedings National Telecommunications Conference, San Diego California, December 1974, pp. 61-66.

9. W.K. Pratt, Digital Image Processing, Wiley, New York, 1977.

10. A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Transactions on Communications, Vol. COM-22, No. 5, May 1974, pp. 614-624.

11. J.A. Roese, et.al., "Interframe Transform Coding and Predictive Coding Methods," Proceedings 1975 International Communications Conference, San Francisco, Vol. 2, June 1975, pp. 17-21.

12. J.A. Roese and G.S. Robinson, "Combined Spatial and Temporal Coding of Digital Image Sequences," Proceedings SPIE, Vol. 66, August 1975, pp. 172-180.

13. J. A. Roese, "Interframe Coding of Digital Images Using Transform and Hybrid Transform/Predictive Techniques," Naval Undersea Center, TP 534, June 1976.

5. Radar Image Coding

There are many types of non-pictorial data which possess a high degree of spatial correlation. Examples include sonar signals recorded from multiple channels, forward-looking infrared scanners, and synthetic aperture radar (SAR). This spatial correlation can be exploited by two-dimensional transform coding techniques, originally developed for pictorial images, to achieve a substantial bandwidth compression for the non-pictorial data. This section presents preliminary results of an investigation of the application of image coding techniques to the transmission of radar imagery.

There are three basic forms of radar imagery: scanned narrow beam radar, strip synthetic aperture radar, and spotlight radar. References <1-4> provide background information of the operating principles of these radar systems.

5.1 Scanned Radar Image Coding

The simplest form of radar imaging, illustrated in figure 5-1, involves the measurement of the radar reflectivity of a scene by effectively scanning the scene with a narrow beamwidth antenna. The resultant images formed from the measured reflectivity appear different than optical images because of spatial resolution limitations imposed by the radar transmission frequency, and the inclusion of specular as well as diffuse reflection. This latter effect often results in an extremely large dynamic range image. From an image coding viewpoint the radar images exhibit a significant spatial correlation. Furthermore, multiple images are often formed at several different transmission frequencies. There is also significant correlation between the image arrays.

Scanned radar images can be treated in the same manner as visual images for purposes of data coding if proper attention is paid to the spatial and cross-frequency correlation factors, and to the wide

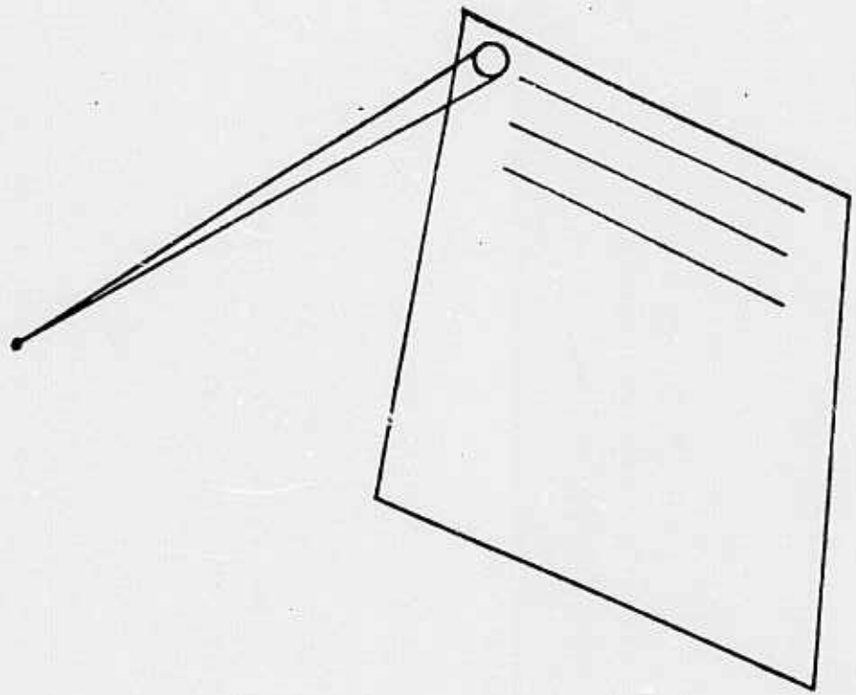
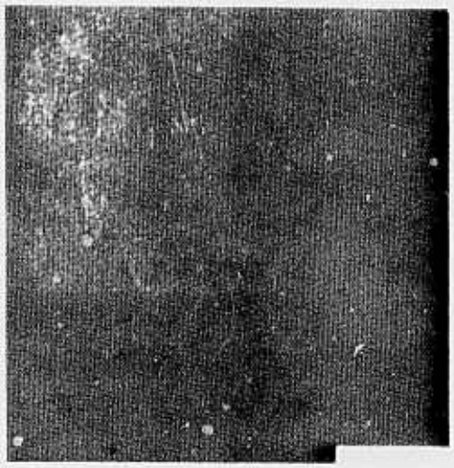


Figure 5-1. Scanned radar image.

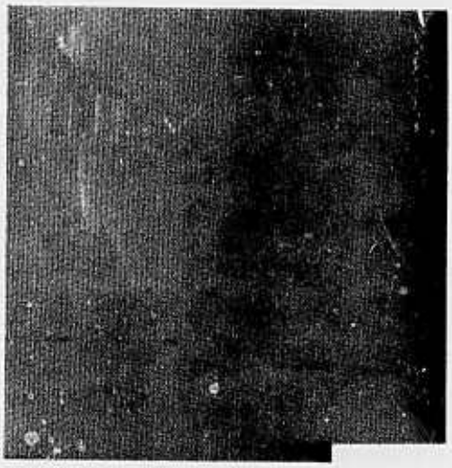
dynamic range of the data samples. Predictive, transform, and hybrid transform/predictive coding techniques should be well suited to the coding of this type of radar imagery. However, research is needed to determine optimal designs to match the coders to the radar signal characteristics, and then to verify the coding methods by computer simulation. Preliminary efforts are described below.

A set of scanned radar images obtained from a HOWLS (Hostile Weapons Location System) <1> radar has been obtained from Lincoln Laboratories for testing of image coding concepts. The set consists of sixty-four 160 x 160 pixel arrays each of which represents a radar image of the same physical area obtained over 64 swept transmission frequencies. Each pixel value is proportional to the magnitude of the complex (real and imaginary) components of a radar return. Pixel values were furnished quantized to 256 levels (8 bit binary code). Figure 5-2a contains a photograph of the first frame of the sequence. Subsequent frames are quite similar in appearance. Subjectively, the single frames appear to be quite noisy. The apparent noise is attributable to radar background clutter. Figure 5-2b shows the average of the 64 frames. The resultant image appears much less noisy, but its contrast is quite low. In figures 5-2c and 5-2d the first and the average frames have been rescaled over their minimum to maximum limits. Cube root scaling over the dynamic range is presented in figures 5-2e and 5-2f. Rescaling is seen to be quite effective in visual enhancement of the imagery.

Several examples of intraframe cosine transform coding are shown in figure 5-3 for coding at an average rate of 1.0 bit/pixel. In all the examples the image has been subdivided into 16 x 16 blocks and a zonal coding algorithm has been employed to code the cosine transform coefficients. Figures 5-3c and 5-3d show image reconstructions of the first and average frames for quantization based upon a Markov process correlation model of the original image assuming a separable correlation factor of $\rho = 0.7$. In figures 5-3e and 5-3f the correlation factor is set at $\rho = 0.9$. The coded pictures are



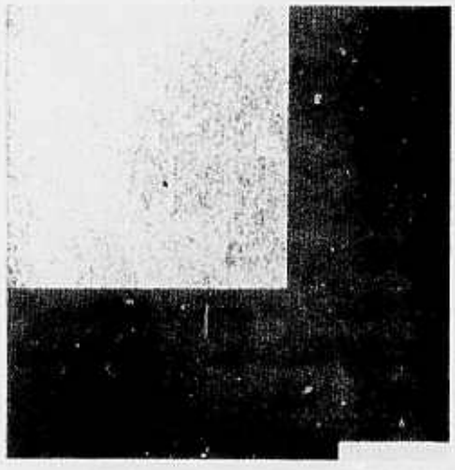
(a) frame 1



(b) average of 64 frames

(c) linear scaling of (a)

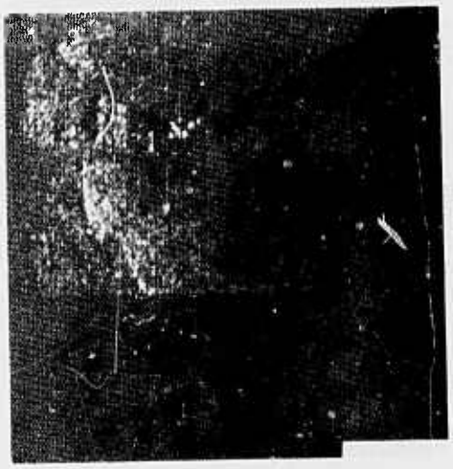
(d) linear scaling of (b)



(e) cube root scaling of (a)

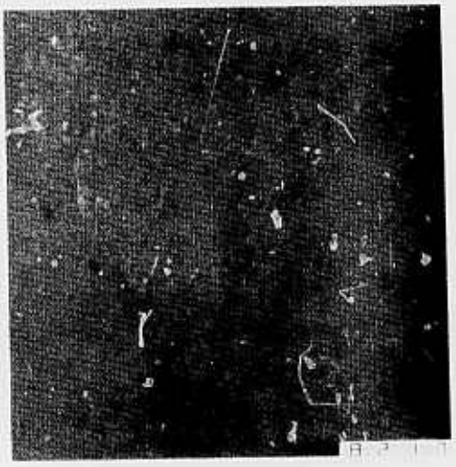
(f) cube root scaling of (b)

Figure 5-2. HOWLS radar images.



(a) original frame #1
linear scaled

(b) original 64 frame average
linear scaled



(c) cosine coding of (a)
1 bit/pixel, ρ = 0.7

(d) cosine coding of (b)
1 bit/pixel, ρ = 0.7

(e) cosine coding of (a)
1 bit/pixel, ρ = 0.9

(f) cosine coding of (b)
1 bit/pixel, ρ = 0.9

Figure 5-3. Cosine transform coding of HOWLS images.

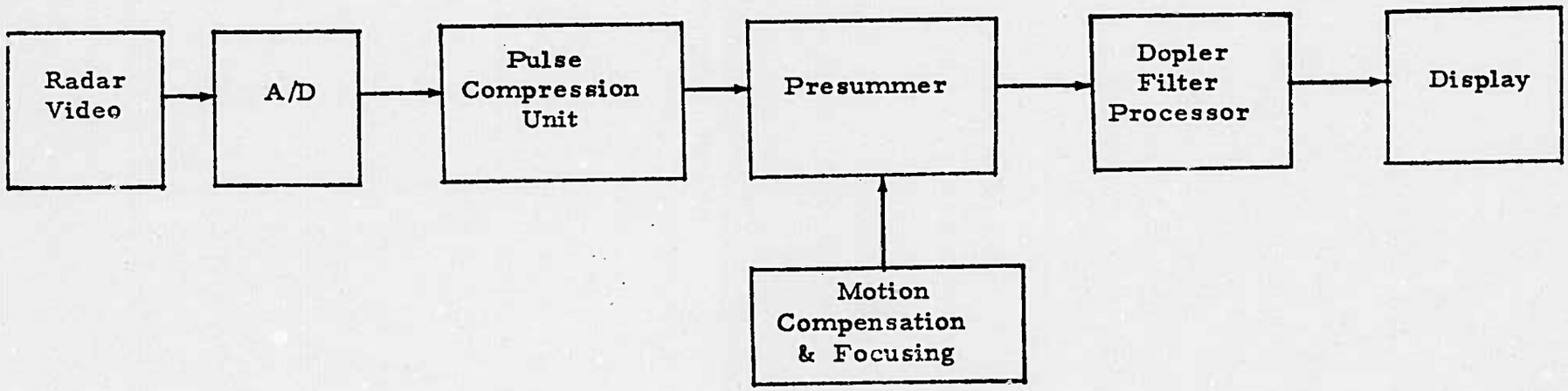
reasonable replicas of the originals. However, close inspection reveals some coding errors. At low correlation factors the specular reflection white peaks are preserved reasonably well, but the background contains some high spatial frequency artifacts. On the other hand, with a high correlation factor the background is much smoother, but the white peaks are attenuated. The basic problem is the wide dynamic range of the imagery. The point reflector targets are not well matched by the Markovian model. One possible solution is to detect the presence or absence of a point target in a spatial block and then adjust the correlation factor appropriately. Another approach is to utilize a spatially adaptive transform algorithm as described in Section 3. Both methods are scheduled for further investigation.

5.2 Strip Synthetic Array Radar Image Coding

In a synthetic array strip radar (SAR) system, reflected radar signals are received at different times as the radar platform moves along a flight path, which is usually normal to the viewed scene. A summation of signals takes place in a data processor. By using synthetic array techniques a single small antenna can have an effective beamwidth as narrow as that which could be obtained by a long linear array.

Optical processing has been the most common technique because of the efficiency with which Fourier transforms can be obtained with lenses. Real-time operation, however, requires all electronic processing. Electronic signal processing, and in particular, digital processing techniques make new modes of the SAR feasible: forward-looking (squint) strip mapping, doppler beam sharpened mapping, and spotlight mapping.

Figure 5-4 contains a block diagram of digital processing. The radar video signal is fed to an analog-to-digital converter (A/D), and then into the pulse compressor unit. The pre-summer is used to reduce



243

Figure 5-4. Digital processing of strip SAR

246

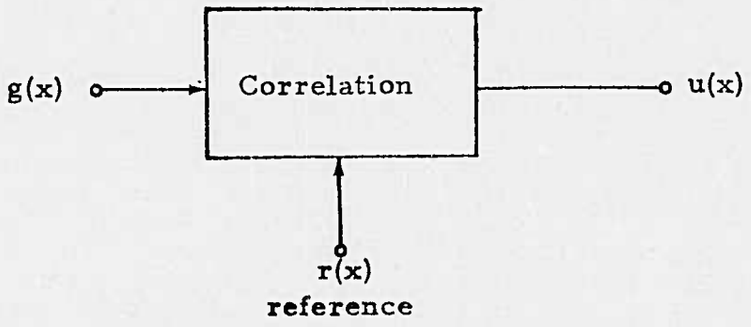
the data rate so that the doppler filter processor can be implemented in real time. Operation of each element is summarized below.

Pulse Compression Unit: The pulse compression unit (PCU) correlates the received pulse with the same waveform that was used to code the transmitted pulse. This code can either be linear frequency modulation (chirp) or a phase code (usually binary). Figure 5-5a illustrates the correlation operation. Alternatively, correlation can be obtained by the Fourier transform processing technique of figure 5-5b.

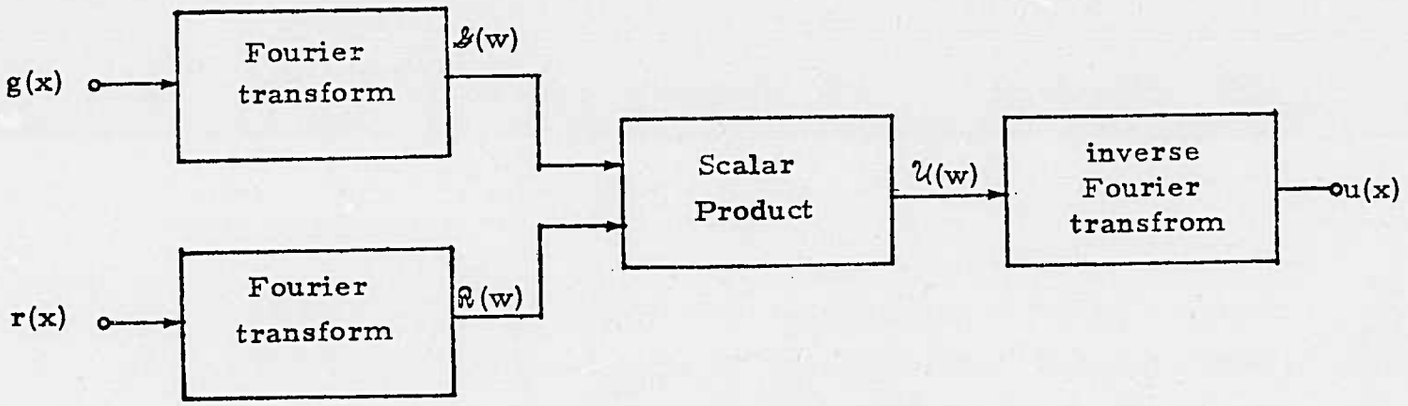
Pre-summer: The data rate of the PCU is often too fast for a digital arithmetic unit, so a pre-summer is used to sum a given number of received pulses. At this point the quadratic phase error is removed and any phase errors due to aircraft motion are compensated. This is the first step in azimuth beam compression.

Doppler Filter Processor: The doppler processor shown in figure 5-4 is the final step in azimuth compression. Each point on the ground in a different azimuth cell is associated with a different azimuth frequency, and this will appear in a different filter. Then filter outputs are displayed as an image array. The usefulness of a strip SAR radar for mapping and navigation is limited by its resolving ability in range and azimuth. Improvement in range and azimuth resolution can be achieved by pulse compression and synthetic aperture techniques. Digital processing of high-resolution radar data then requires a huge quantity of data processing. The advent of the FFT has made the concept of multi-stage frequency splitting feasible in order to minimize the arithmetic rate.

A very important factor in SAR is the correction of phase errors in order to achieve a coherent phase history. Since the synthetic array process is a vector summation of signals, it is very sensitive to phase errors introduced into the signals. Phase errors arise from equipment imperfections, processing approximations, atmospheric



(a) correlation method



(b) transform method

Figure 5-5. Correlation processing.

disturbances, but mainly nonlinear motion of the radar platform. In generating the synthetic antenna the signal-processing equipment assumes that the radar flies along a straight line at a constant speed. Since this is not possible because of weather conditions or intentional maneuver, it is often necessary to have additional equipment for motion compression.

There has been relatively little research devoted to the digital bandwidth compression of SAR strip radar images. The simplest approach, investigated by Steinberg, involves hard limiting of the SAR signal in the A/D converter to two binary levels <2>. This operation results in considerable image degradation, but the coded images are recognizable. Chen has developed a radar image coding method based upon dual mode interpolation <3> for images with strong targets in background clutter. This method takes advantage of the strong contrast between the target and background in SAR imagery and the fact that the target area is normally a small percentage of the total SAR image area. The compressor operates in two separate modes. In the target area, detected by the presence of large grey levels, all picture elements are transmitted, while in the background area a representative value is transmitted. The receiver performs linear interpolation to reconstruct the bandwidth compressed SAR imagery. Chen reports good quality SAR images using this method at a bit rate of 1.5 bits per pixel.

More sophisticated methods of image coding such as linear predictive coding could be applied to SAR radar formed images. Another approach, which offers significant implementation advantages, is to code the SAR image data in "phase history" form at the output of the scalar product unit of figure 5-4b. At this point in the chain, the radar data appears in a hybrid form: as the one-dimensional Fourier transform of the image in one spatial coordinate, and as a spatial domain representation in the orthogonal coordinate. As a result, the hybrid Fourier transform/DPCM image coding method can be applied directly at this point in the processing chain. Essentially

the hybrid coder reduces to a spatially adaptive DPCM coder since the data is already in Fourier transform form. Experience with optical images indicates that the phase history domain samples should be relatively de-correlated, and can, therefore, be efficiently coded by a switched DPCM coder which removes the remaining redundancy between SAR image lines. Research is required to determine the statistical characteristics of the phase history domain for typical radar imagery, and to assess the effects of DPCM quantization.

Figure 5-6 contains a photograph of an SAR image obtained from the University of Texas at Austin Applied Research Laboratories. This image is part of the FLAMR/FLAP data set generated for the Air Force Avionics Laboratory at Wright Patterson Air Force Base. The SAR image has been recorded in the phase plane before and after motion compensation, and in the space domain. Simulation studies of coding methods are now underway on this data.

5.3 Spotlight Radar Image Coding

In a spotlight radar imaging system the radar antenna is not fixed in a pointing direction, as in SAR radar, but rather the radar continuously illuminates a spot on a scene as the radar platform moves past the scene. An equivalent geometry can be obtained by imaging an object on a rotating platform with a static radar. In the spotlight imaging process the radar returns are spatially arranged in a polar coordinate system. This polar data is then interpolated over a rectangular grid, and an inverse two-dimensional Fourier transform is taken to produce a spatial domain image.

The spotlight radar signal processing methodology leads naturally to Fourier domain coding of its phase history representation in polar or Cartesian coordinate form. Coding simulation experiments will be undertaken when source data is available.

Figure 5-6. SAR image.

References

1. V.L. Lynn, "Howls Radar Development," AVP/GCP Joint Symposium - Avionics/Guidance and Control for RPV's, Florence, Italy, October 1976.

2. B.D. Steinberg, "Hard Limiting in Synthetic Aperture Signal Processing," IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-11, No. 4, July 1975, p. 556-561.

3. W.H. Chen, "A Dual Mode Nonlinear Interpolative Compressor for SAR Images," Proceedings of the 19th Annual SPIE Symposium on Efficient Transmission of Pictorial Information, San Diego, August 1975.

APPENDIX G: STUDY OF IMAGE CODING TECHNIQUES

Interim Report
Contract N66001-77-C 0327

31 January 1977

by
WK Pratt
Image Processing Institute
University of Southern California
Los Angeles CA 90007

Contents

ABSTRACT

1. Introduction
2. Synthetic Aperture Radar
3. FLAMR Radar
4. Statistical Properties of FLAMR Data
5. FLAMR Data Coding
6. Summary
7. References

ABSTRACT

This report describes research work performed by the University of Southern California, Image Processing Institute for the Naval Ocean Systems Center under Contract No. N66001-77-C-0327 during the period of 1 July 1977 to 31 January 1978. During the contract period strip type synthetic aperture radar image data was obtained in digital form. The data has been decoded from its tape and processing format to produce phase plane and spatial plane image records. Image statistics measured in the image plane reveal a stochastic structure similar to optical images.

1. Introduction

This interim report summarizes research work performed by the University of Southern California Image Processing Institute for the Naval Ocean Systems Center under Contract No. N66001-77-C-0327 from 1 July 1977 to 31 January 1978. The format of this interim report is an abbreviation of the final report to be issued upon completion of the contract.

The research study entitled, "Study of Image Coding Techniques," is concerned with assessment of the data compression potential of strip type synthetic aperture radar imagery both in the phase plane and in the image plane. The technical requirements of the contract are listed below:

1. Characterize the statistical properties (mean, variance, probability distribution, two-dimensional correlation, etc.) of SAR data, both image and phase plane, obtained from the University of Texas and data from two additional arrays, such data to be supplied by Naval Ocean Systems Center (NOSC).

2. Identify statistical models that describe the SAR data, that can be used to predict the performance of various bandwidth compression techniques.

3. Identify the nonlinear function such as contrast stretching or compression that will make the data statistics more closely matched to those required by known compression techniques.

4. Compare the performance of Fourier and Cosine transform compression techniques, both two-dimensional and hybrid, with the performance of the Karhunen-Loeve transform based upon the measured covariance function of the data. This should be done for both the University of Texas data and the additional array data referred to in Paragraph 1.

Progress has been made toward fulfilling these technical requirements. Strip type SAR imagery has been obtained from the FLAMR radar from the Applied Research Laboratories of the University of Texas at Austin. No additional SAR imagery has been supplied by NOSC. The Texas data has been successfully decoded from its magnetic form to produce data arrays at several points in the phase plane and at one point in the spatial plane. A computer processing algorithm has been developed to reconstruct an optical image of the spatial image data. Statistical analyses have been performed on the output array.

The following sections contain a summary description of synthetic aperture radar imaging, an outline of FLAMR data processing, a presentation of statistical measurements made on the FLAMR data, and an assessment of the data compression potential of FLAMR data.

2. Synthetic Aperture Radar

In a synthetic array strip radar (SAR) system, reflected radar signals are received at different times as the radar platform moves along a flight path, which is usually normal to the viewed scene. A summation of signals takes place in a data processor. By using

synthetic array techniques a single small antenna can have an effective beamwidth as narrow as that which could be obtained by a long linear array. Reference [1] contains a compilation of papers on the subject of SAR.

Figure 2-1 contains a block diagram of digital processing. The radar video signal is fed to an analog-to-digital converter (A/D), and then into the pulse compressor unit. The pre-summer is used to reduce the data rate so that the doppler filter processor can be implemented in real-time. Operation of each element is summarized below.

Pulse Compression Unit: The pulse compression unit (PCU) correlates the received pulse with the same waveform that was used to code the transmitted pulse. This code can either be linear frequency modulation (chirp) or a phase code (usually binary). Alternatively, correlation can be obtained by the Fourier transform processing technique.

Pre-summer: The data rate of the PCU is often too fast for a digital arithmetic unit, so a pre-summer is used to sum a given number of received pulses. At this point the quadratic phase error is removed and any phase errors due to aircraft motion are compensated. This is the first step in azimuth beam compression.

Doppler Filter Processor: The doppler processor shown in figure 2-1 is the final step in azimuth compression. Each point on the ground in a different azimuth cell is associated with a different azimuth frequency, and this will appear in a different filter. Then

Figure 2-1. Digital processing of strip SAR

filter outputs are displayed as an image array. The usefulness of a strip SAR radar for mapping and navigation is limited by its resolving ability in range and azimuth. Improvement in range and azimuth resolution can be achieved by pulse compression and synthetic aperture techniques. Digital processing of high-resolution radar data requires a huge quantity of data processing. The advent of the FFT has made the concept of multi-stage frequency splitting feasible in order to minimize the arithmetic rate.

A very important factor in SAR is the correction of phase errors in order to achieve a coherent phase history. Since the synthetic array process is a vector summation of signals, it is very sensitive to phase errors introduced into the signals. Phase errors arise from equipment imperfections, processing approximations, atmospheric disturbances, and the nonlinear motion of the radar platform. In generating the synthetic antenna the signal processing equipment assumes that the radar flies along a straight line at a constant speed. Since this is not possible because of weather conditions or intentional maneuver, it is often necessary to have additional equipment for motion compression.

There has been relatively little research devoted to the digital bandwidth compression of SAR strip radar images. The simplest approach, investigated by Steinberg, involves hard limiting of the SAR signal in the A/D converter to two binary levels [2]. This operation results in considerable image degradation, but the coded images are recognizable. Chen has developed a radar image coding method based

upon dual mode interpolation [3] for images with strong targets in background clutter. This method takes advantage of the strong contrast between the target and background in SAR imagery and the fact that the target area is normally a small percentage of the total SAR image area. The compressor operates in two separate modes. In the target area, detected by the presence of large grey levels, all picture elements are transmitted, while in the background area a representative value is transmitted. The receiver performs linear interpolation to reconstruct the bandwidth compressed SAR imagery. Chen reports good quality SAR images using this method at a bit rate of 1.5 bits per pixel.

More sophisticated methods of image coding such as linear predictive coding could be applied to SAR radar formed images. Another approach, which offers significant implementation advantages, is to code the SAR image data in "phase history" form. At this point in the chain, the radar data appears in hybrid form: as the one-dimensional Fourier transform of the image in one spatial coordinate, and as a spatial domain representation in the orthogonal coordinate. As a result, the hybrid Fourier transform/DPCM image coding method can be applied directly at this point in the processing chain. Essentially the hybrid coder reduces to a spatially adaptive DPCM coder since the data is already in Fourier transform form. Experience with optical images indicates that the phase history domain samples should be relatively de-correlated, and can, therefore, be efficiently coded by a switched DPCM coder which removes the remaining redundancy between SAR image lines.

3. FLAMR Radar Data Processing

The FLAMR radar is a forward angle looking synthetic aperture radar developed by Hughes Aircraft Company, Culver City, California. It has been equipped with a wideband instrumentation tape recorder for archival storage of signals at key points in the SAR signal processing chain. The Applied Research Laboratories at the University of Texas at Austin is the data processing contractor for conversion of the wideband tape data to computer compatible form, data analysis, and data dissemination. Details of the FLAMR data recording format and processing tasks are presented in reference [4].

Figure 3-1 contains a simplified block diagram of the FLAMR signal processing system. In operation, the radar return from the receiver enters a mixer which produces the quadrature "phase domain" signals I and Q which are then sampled, uniformly quantized, and binary coded with six bits. Next, pulse compression and motion compensation are performed. The final step in the processing is reconstruction of the spatial domain image from the phase plane data. This is accomplished by the system of figure 3-2. In the reconstruction processor, the I and Q signals are pre-summed. Then, a one-dimensional Fourier transform is taken over short sequences of the I and Q signals to produce a spatial domain representation. The logarithm of the magnitude of the real and imaginary sample components is then formed to produce a spatial domain image. At this point, the output is spatially redundant; containing the equivalent of four

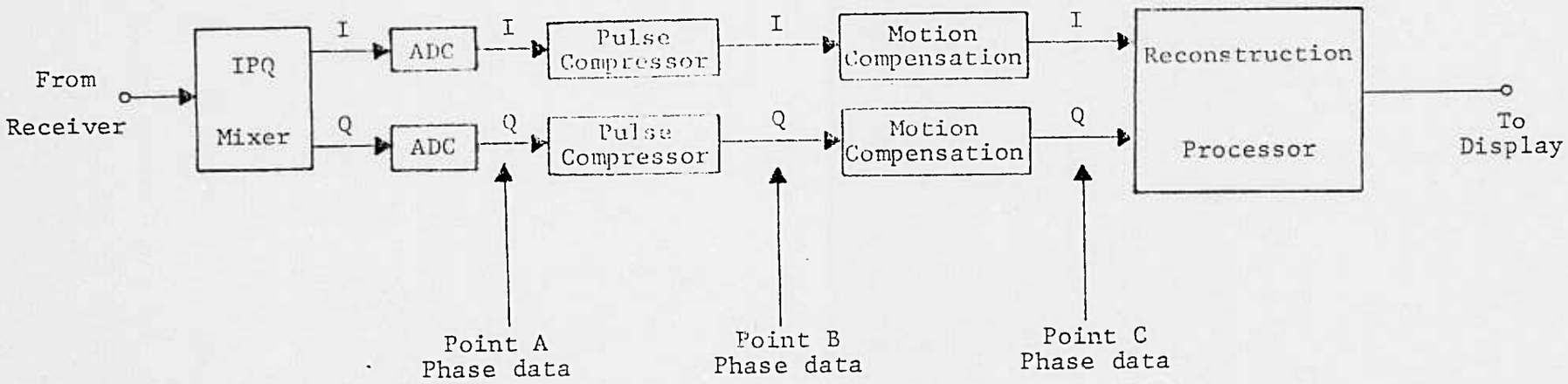


Figure 3-1. Simplified block diagram of FLAMR system

263

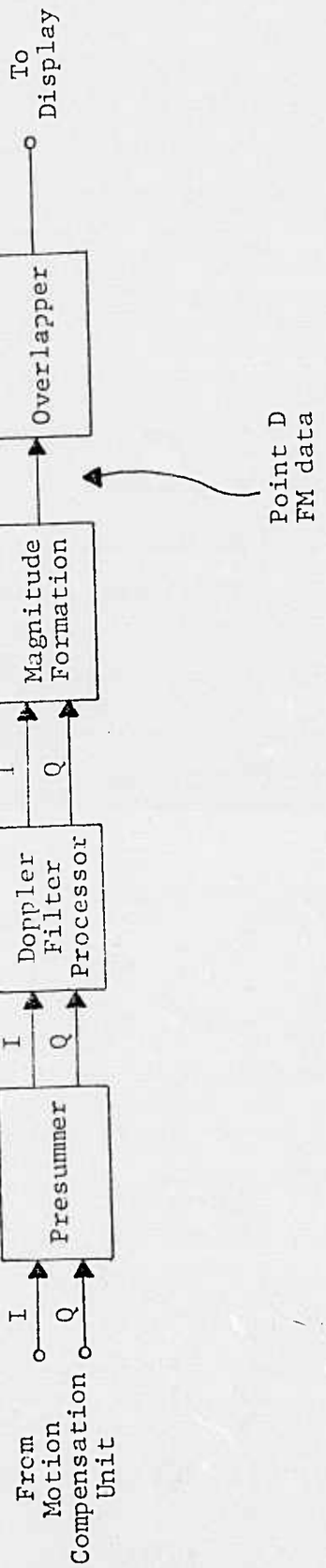


Figure 3-2. Block diagram of reconstruction processor

spatially multiplexed views of the same ground region. The redundancy is eliminated by a spatial averaging process in one-dimension, called overlapping. The final output is a data array that nominally represents the radar reflectance of a ground region on a point-by-point raster.

The FLAMR recording/processing system provides SAR data at four points in the system shown in figures 3-1 and 3-2 (labelled A, B, C, D). The sizes of the data arrays are indicated in figure 3-1; data formats are presented in figures 3-3 and 3-4.

Figure 3-5 contains two displays of the output image obtained directly from the FM data. The first is a direct eight-bit display and the second is a display after linear contrast stretching.

4. Statistical Properties of FLAMR Data

It is well known that the data compression performance of many types of image coding techniques such as statistical, predictive, interpolative, and transform coding can be accurately predicted from a stochastic characterization of the image source. Further, the stochastic characterization provides vital information in the design of image coders. Accordingly, statistical measurements have been made of the FLAMR imagery to develop a stochastic model of its imagery.

Any N x N pixel image source can be totally characterized by the joint probability density of order N^2. However, the establishment of this density, except for special cases such as the Gaussian density,

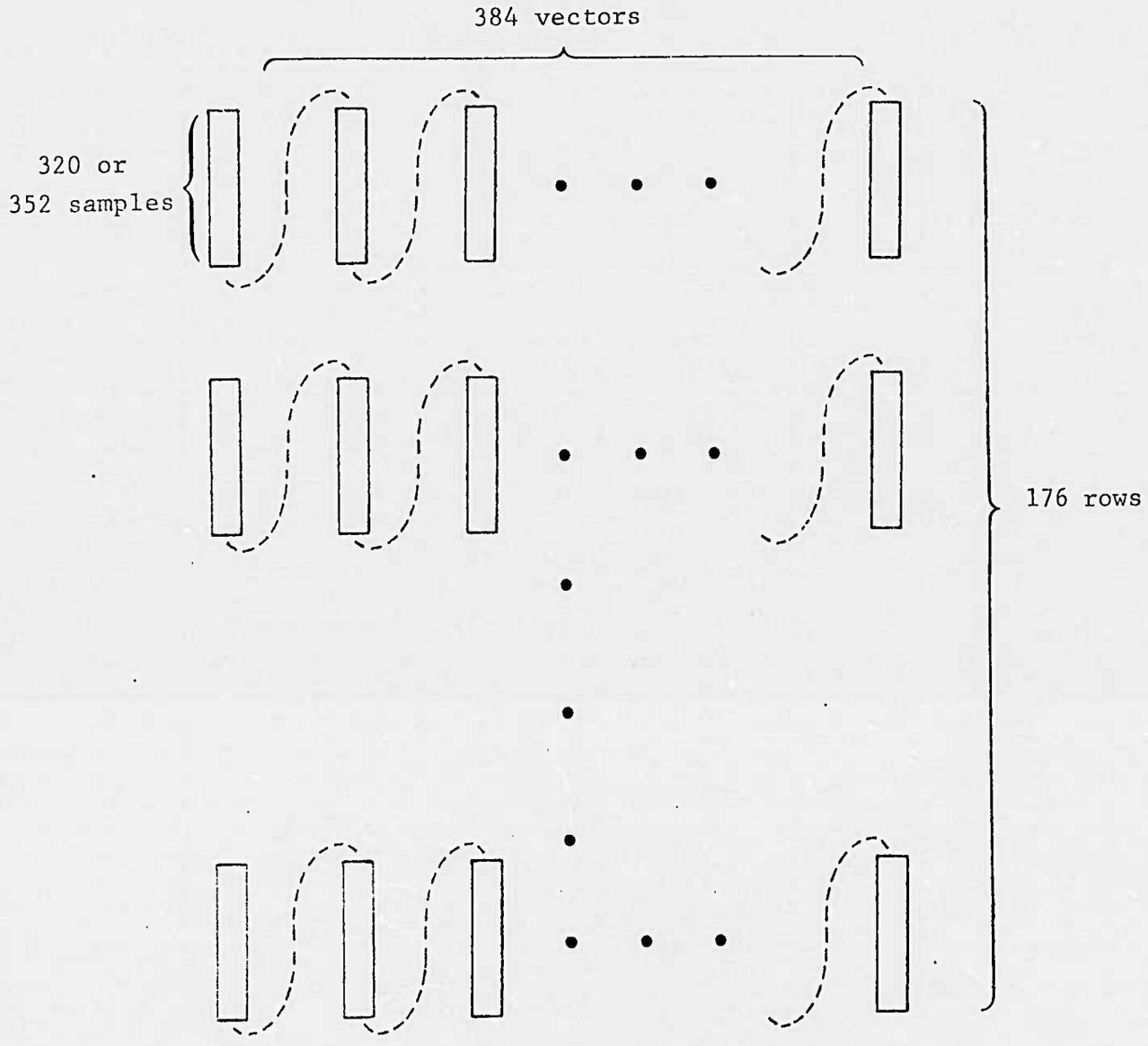


Figure 3-3. I-Q Phase Plane Motion Compensated Data.

2666

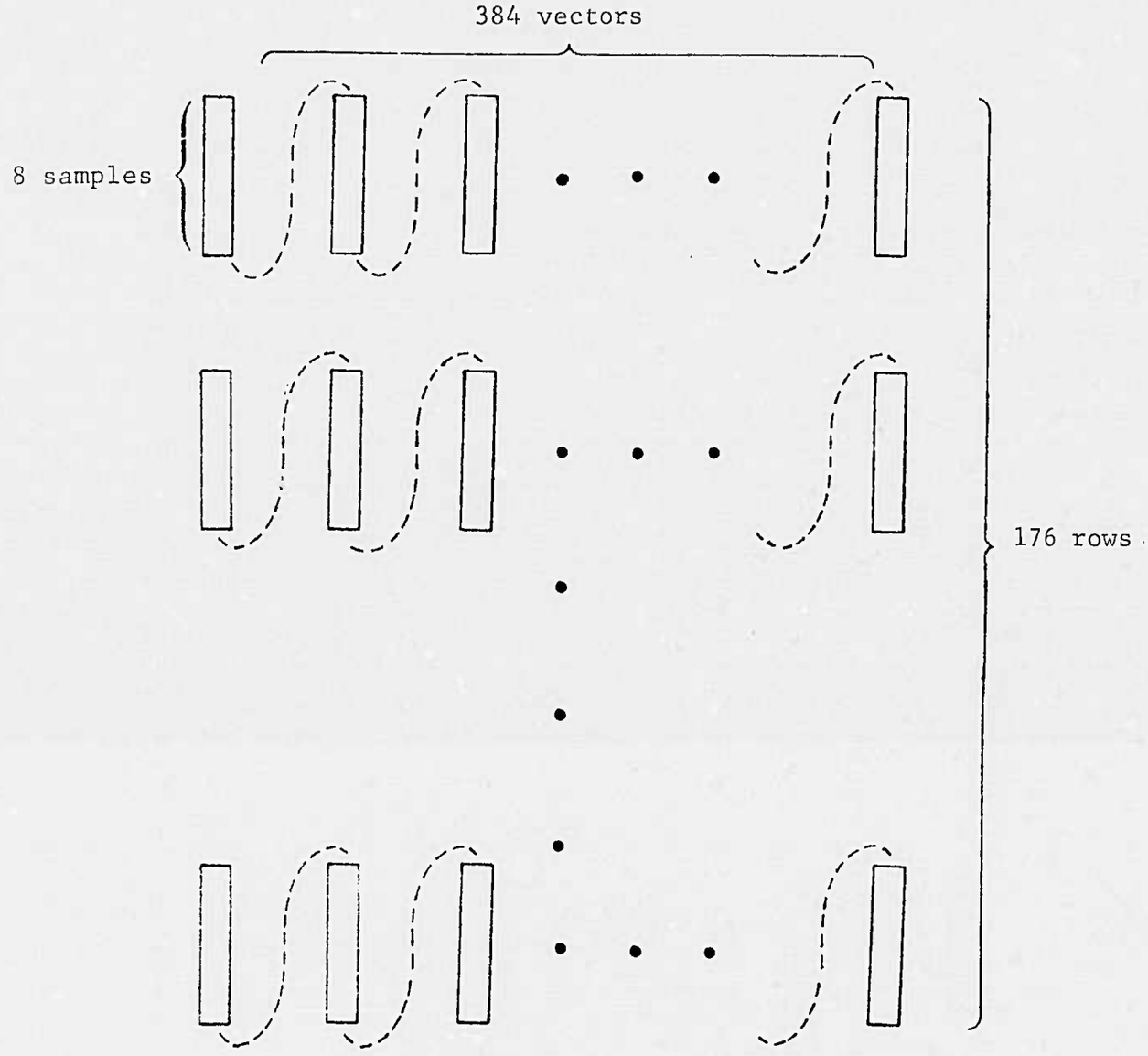


Figure 3-4. FM Data Format.

(a) original

(b) original after
contrast stretching

Figure 3-5. Displays of FLAMR FM data
after computer overlapping.

is infeasible because of its enormous dimensionality. It is common practice to measure only lower order moments and histograms, and then attempt to infer the stochastic source structure.

4.1 Image Plane Statistical Measurements

Several statistical measurements have been made on the FLAMR spatial domain output image. These include the first order histogram, before and after contrast stretching, and the correlation function on the stretched image.

Figure 4-1 contains a histogram plot of the FLAMR output image of 352 x 384 pixels. The histogram exhibits a spike at zero intensity resulting from low amplitude clipping in the FLAMR signal processing circuitry. The main portion of the histogram appears to be well-approximated by a Gaussian density. Figure 4-2 shows the histogram of the output image after linear contrast stretching.

To facilitate processing with existing statistical analysis software, the FLAMR output image was reduced in size to a 256 x 256 pixel array by truncating its bottom and right-hand side. The histogram of the resultant image is shown in Figure 4-3, and the mean and standard deviation are indicated on the figure.

Figures 4-4 and 4-5 contain plots of the normalized row and column measured covariane function of the 256 x 256 pixel image. Numerical values of the covariance function are listed in Table 4-1. The covariance drops to about 0.65 along rows and to about 0.53 along columns for nearest neighbors. These levels are somewhat lower than

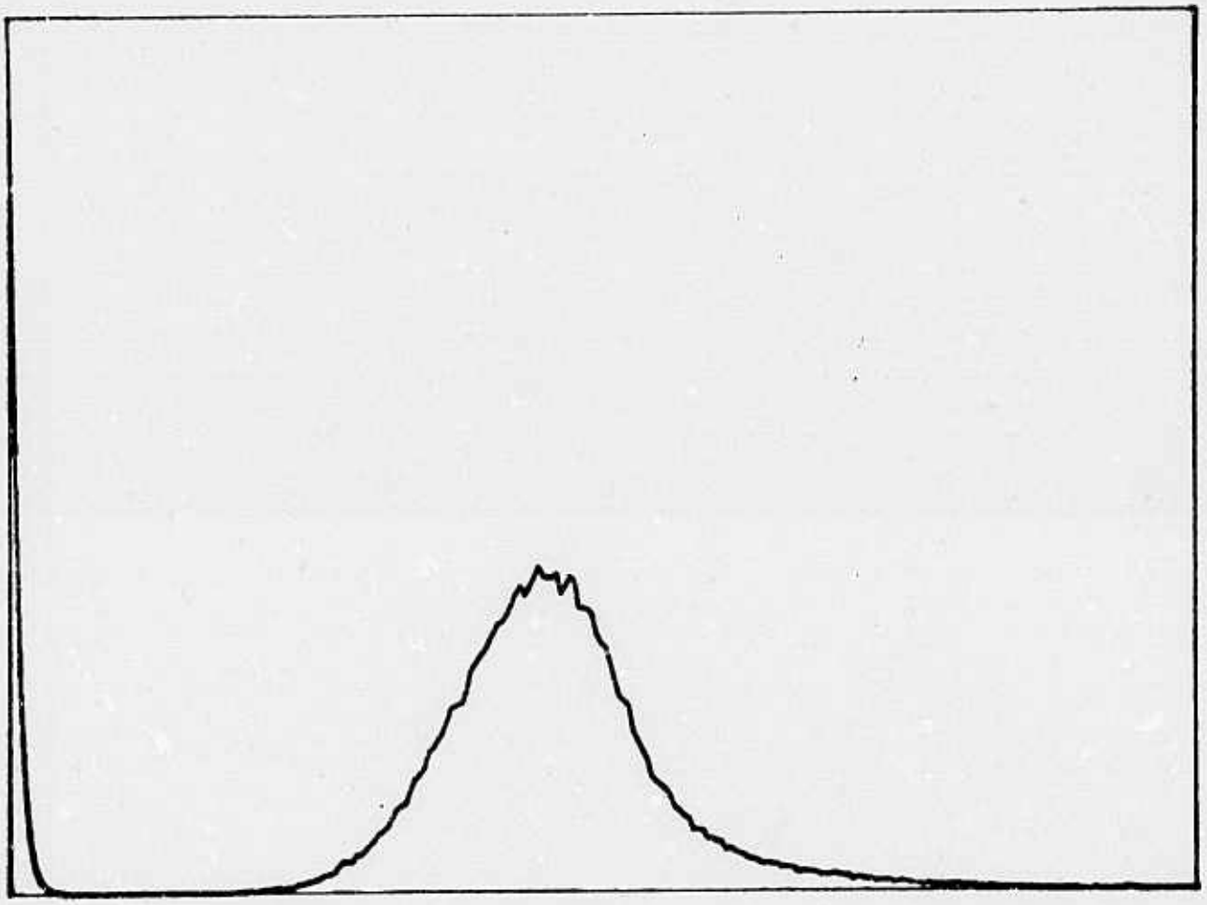


Figure 4-1. Full image spatial domain histogram 352 x 384 pixels.

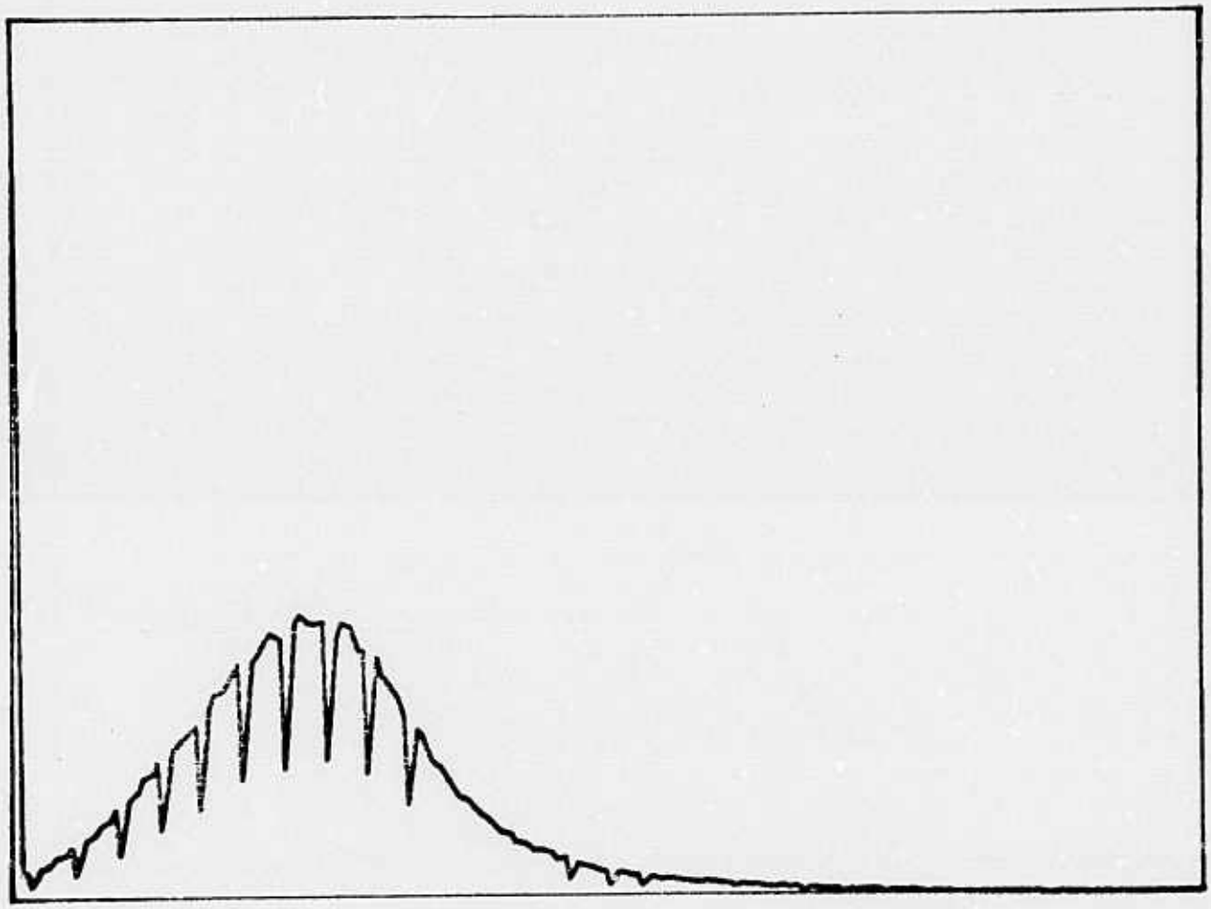


Figure 4-2. Full image spatial domain histogram after linear contrast stretching. 352 x 384 pixels.

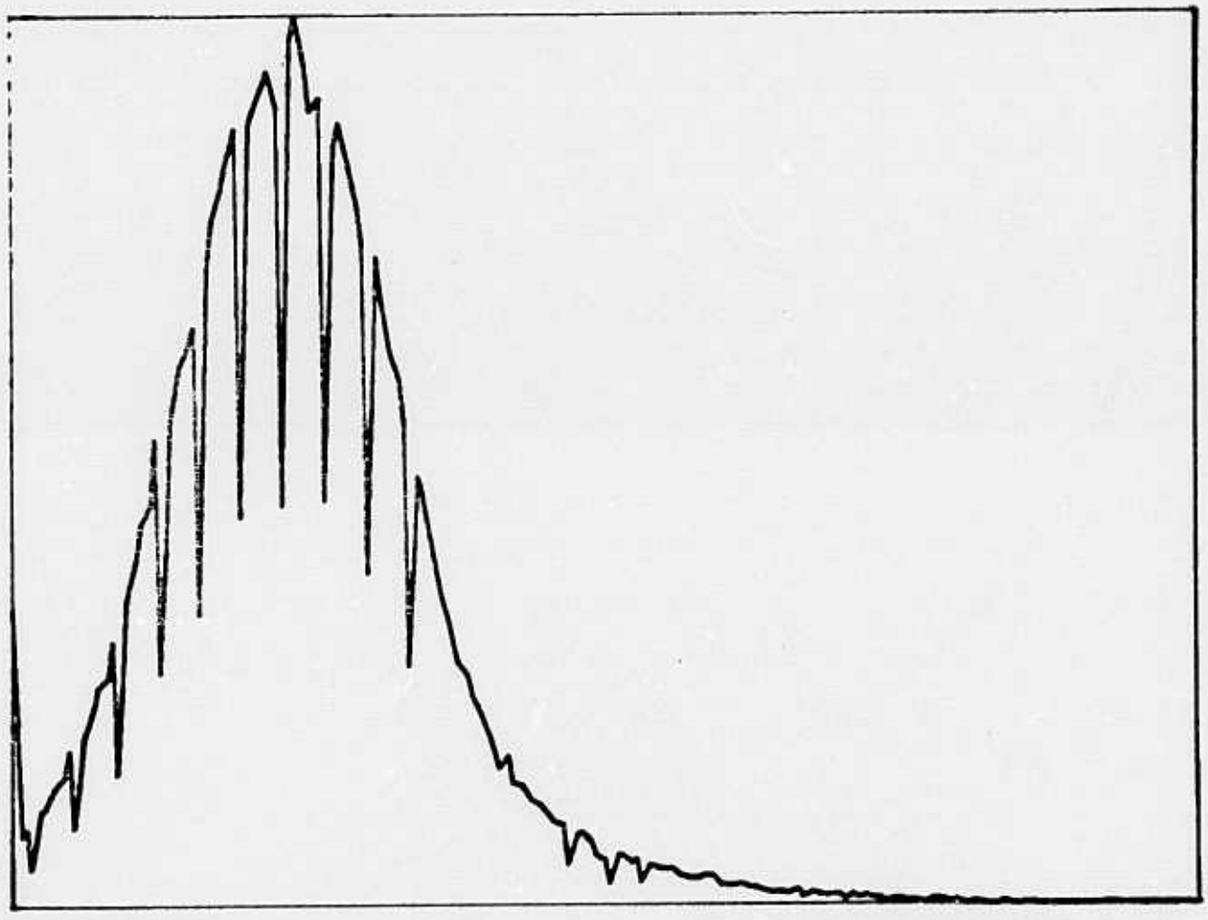


Figure 4-3. Reduced image spatial domain histogram after linear contrast stretching. 256 x 256 pixels. $\eta = 54.0$, $\sigma = 25.4$.

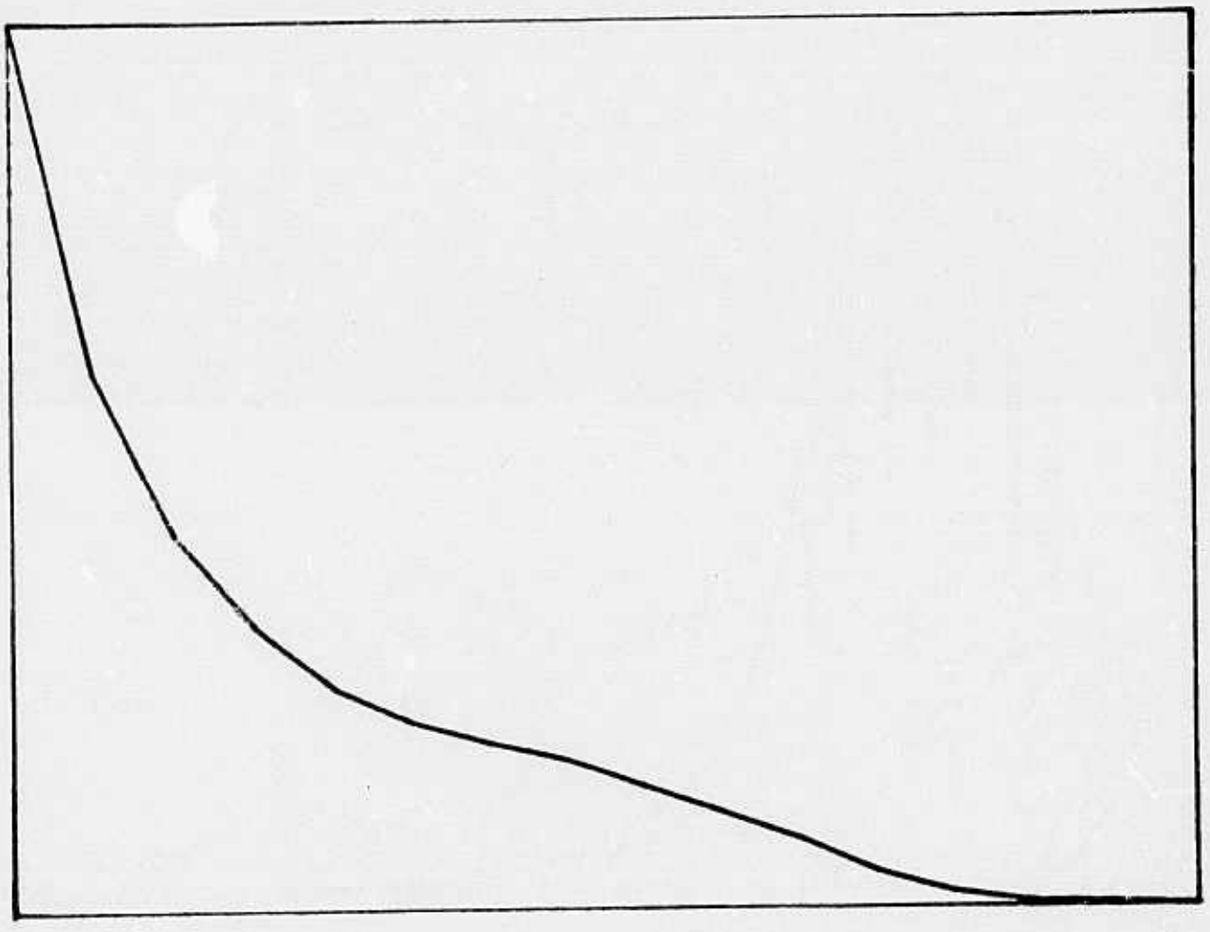


Figure 4-4. Reduced image row covariance function after linear contrast stretching. 256 x 256 pixels.

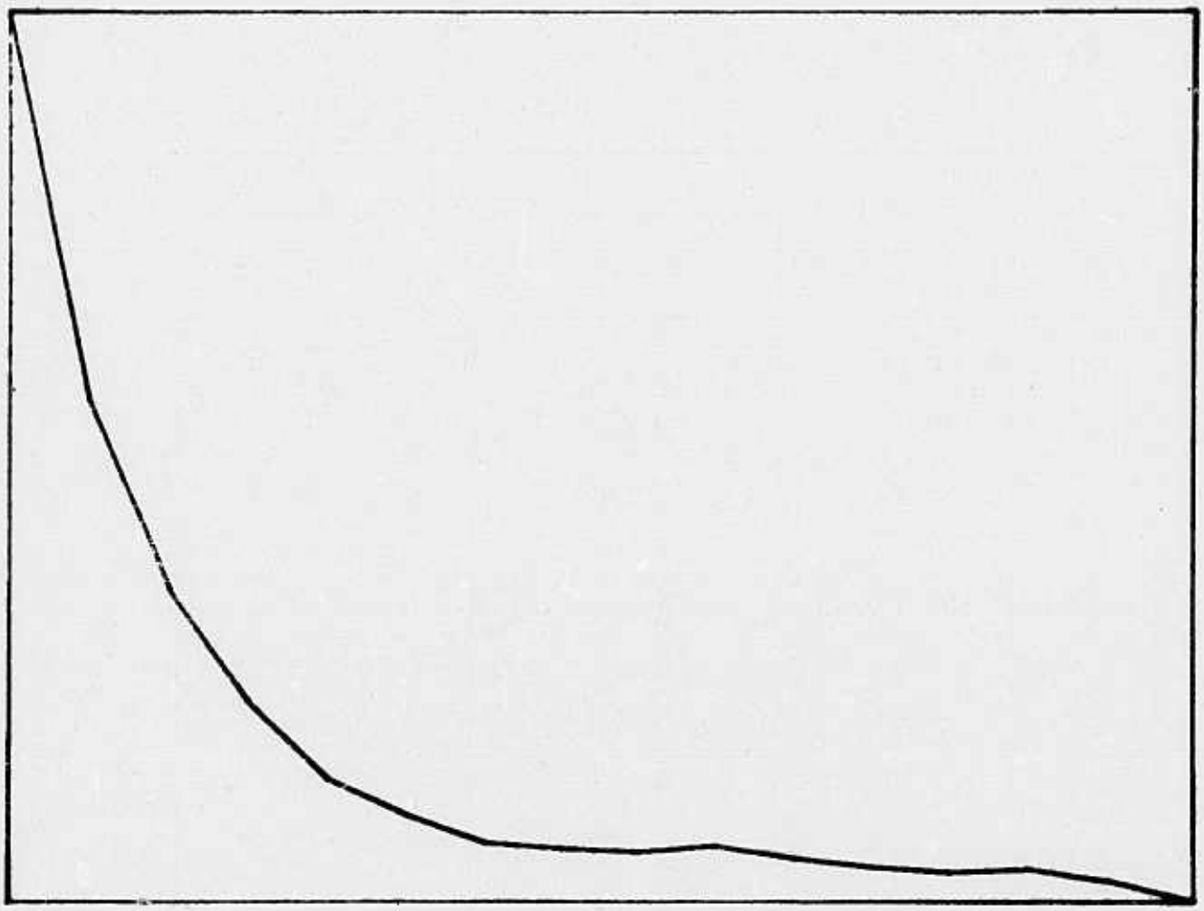


Figure 4-5. Reduced image column covariance function after linear contrast stretching. 256 x 256 pixels.

Table 4-1

Numerical values of row and column covariance functions for spatial domain test image.

log	row corr.	column corr.
0	1.000	1.000
1	0.649	0.527
2	0.449	0.341
3	0.356	0.250
4	0.304	0.198
5	0.270	0.162
6	0.250	0.145
7	0.238	0.132
8	0.227	0.129
9	0.212	0.128
10	0.197	0.130
11	0.183	0.124
12	0.164	0.120
13	0.152	0.117
14	0.145	0.119
15	0.145	0.113
16	0.143	0.103

the 0.80 to 0.95 values encountered with typical optical images. Finally, it should be noted that the covariance functions of figures 4-4 and 4-5 can be fit reasonably well by a simple first order Markov process.

5. FLAMR Data Coding

From the statistical information presented in the previous section, it is posible to establish performance bonds for data compression of FLAMR imagery. Such bounds can be developed for ideal coders and practical coding systems.

The ultimate performance bound for distortion-free coding of an image source can be obtained in terms of the source entropy. Image source entropy can be estimated from the source histograms. From the histogram of Figure 4-3, for the reduced output image with linear contrast stretching, the zeroth order entropy is found to be

$$H(x_0) = 5.93 \text{ bits/pixel}$$

over a scale of 8 bits/pixel. This means that statistical coding of individual pixels by a variable length Huffman code could lead to a savings as much as 2 bits/pixel. First order entropy measurements have been made from the joint row and column histograms of the test image. The row first order entropy is

$$H(x_0|x_1) = 5.45 \text{ bits/pixel}$$

while the column first order entropy is

$$H(x_0|x_1) = 5.59 \text{ bits/pixel}$$

Thus, coding of each pixel based upon knowledge of its previous

neighbor could lead to additional savings of about 0.5 bits/pixel.

6. Summary

The FLAMR data has been successfully decoded from the magnetic tapes supplied by the University of Texas. From this data a computer overlapping algorithm has been developed to produce a useful output image for data analysis. Histograms and moments have been calculated from the FLAMR test image in the spatial domain in order to form a statistical model of FLAMR imagery. From the statistical data presently obtained, it is clear that the FLAMR image is quite similar to an ordinary optical view of the same scene, except that it is somewhat noisier and exhibits less spatial correlation than the optical image. Entropy estimates have been obtained in the spatial domain on the basis of one- and two-dimensional histograms.

The remaining task in the study is to make statistical measurements of the phase domain signal representation of the image prior to doppler processing. In order to accomplish this task, it will be necessary to develop computer algorithms and software to perfrom the pre-summing, doppler filter processing, and magnitude formation. This effort is underway, but has been hindered by incomplete availability of details of the FLAMR processing system.

7. References

1. J.J. Kovaly, Synthetic Aperture Radar, Artech House, Inc., Dedham, Massachusetts, 1976.

2. B.D. Steinberg, "Hard Limiting in Synthetic Aperture Signal Processing," IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-11, No. 4, July 1975, pp. 556-561.

3. W.H. Chen, "A Dual Mode Nonlinear Interpolative Compressor for SAR Images," Proceedings of the 19th Annual SPIE Symposium on Efficient Transmission of Pictorial Information, San Diego, August 1975.

4. F.A. Collins and K.E. Graf, "FLAMR/FLAP Data: A Discussion of Origin and Formats," The University of Texas at Austin, Applied Research Laboratories ARL-TR-76-42, September 1976.

APPENDIX H: STOCHASTIC IMAGE MODELS AND HYBRID CODING

Final Report
Contract N00953-77-C-003 MJE

1 November 1976–31 October 1977

by
AK Jain and SH Wang
Department of Electrical Engineering
State University of New York at Buffalo
Bell Hall, Amherst Campus
Buffalo NY 14260

ACKNOWLEDGEMENT

The authors are grateful to Dr. Edwin Wrench of the Naval Ocean Systems Center for not only supporting this project but also for motivating several theoretical as well as computational experiments presented in this report. Special thanks are also due to Harper Whitehouse for his encouragement and enthusiasm for this project.

CONTRACT PERSONNEL

Anil K. Jain,	Project Director
Shenq H. Wang,	Graduate Student
Jaswant R. Jain,	Graduate Student
Joel Montione,	Programming Assistant
Rita M. Maedl,	Technical Typist

Research Papers Produced by Contract Related Work

1. A.K. Jain, "An Operator Factorization Method for Restoration of Blurred Images," IEEE Trans. Computers, vol. C-26, pp. 1061-1071, Nov. 1977.

2. A.K. Jain, "Fast Inversion of Banded Toeplitz Matrices by Circular Decompositions," Accepted for publication in IEEE Trans. Acous., Speech, and Signal Processing.

3. A.K. Jain,"A Sinusoidal Family of Unitary Transforms," Dept. of Electrical Engineering, SUNY Buffalo, Tech. Report, July 1977, (submitted for publication).

4. A.K. Jain and J.R. Jain, "Partial Differential Equations and Finite Difference Methods in Image Processing, Part II: Image Restoration", Accepted for publication in IEEE Trans. Auto. Control, see also "Application of Partial Differential Equation Models in Restoration of Images," Proc. IEEE Computer Conf. on Pattern Recognition and Image Processing, pp. 38-46, Rensselaer Poly. Ins., June 6-8, 1977.

5. A.K. Jain and S.H. Wang, "Application of Stochastic Models in Image Coding," International Picture Coding Symposium, Tokyo, Japan, Aug. 1977.

A.K. Jain and S.H. Wang jointly received two first prizes in the Image Coding Competition held at the Picture Coding Symposium for their entries (based on the results presented in this report) in 1 bit/pixel and 0.5 bit/pixel categories.

TABLE OF CONTENTS

	Page
ABSTRACT	1
CHAPTER ONE: INTRODUCTION AND SUMMARY	2
1. Introduction	2
2. Summary of Semicausal Models	3
3. Summary of Experiments	3
4. Summary of Results and Conclusions	4
CHAPTER TWO: SEMICAUSAL IMAGE MODELS AND HYBRID CODING	10
1. Hybrid Coding of Separable Covariance Image Fields	10
2. White Noise Driven Semicausal Representations(SC1 Model)	12
3. Minimum Variance Semicausal Representations(SC2 Model)	17
4. Semicausal Representations as Two Source Models for Finite Random Fields	18
5. Stochastic Decoupling of Semicausal Models by the Cosine Transform	21
6. Stochastic Decoupling of Stationary Semicausal Fields	26
CHAPTER THREE: HYBRID CODER DESIGN FOR SEMICAUSAL RANDOM FIELDS	32
1. DPCM Equation	32
2. Bit Allocation	33
3. Integer Bit Allocation	42
CHAPTER FOUR: ADAPTIVE HYBRID CODING	46
1. Adaptive Variance Estimation	46
2. Adaptive Classification	50
CHAPTER FIVE: EXPERIMENTAL RESULTS AND CONCLUSIONS	55
1. Nonadaptive Hybrid Coding	55
2. Adaptive Variance Estimation	58
3. Adaptive Classification	59
4. Conclusions	66
REFERENCES	80

ABSTRACT

This report presents analytic and computer simulation results of hybrid coding of two dimensional images. Adaptive as well as nonadaptive schemes which could be considered by NOSC for hardware integration in the existing implementations are presented. It is shown that a class of stochastic representations of images, called semicausal models, admit the existing hardware realization of hybrid coding utilizing the discrete Cosine transform (DCT) and a first order DPCM technique. Thus, several performance trade-offs in terms of improving signal to noise ratio, robustness with respect to channel errors and statistical variations in image, and reducing further, the system bandwidth requirements, are available without changing the basic hardware configuration.

KEY WORDS: Image Coding, Data Compression, Bandwidth Reduction, Random Fields, Image Models, Karhunen Loeve Transform, DPCM.

CHAPTER ONE

INTRODUCTION AND SUMMARY

1. Introduction

In this report we consider image coding techniques which combine transform and predictive coding techniques. Such techniques have been called (in the context of digital Image Process) Hybrid coding [1] or more generally, Hybrid processing. The basic idea behind hybrid processing is to first convert a two dimensional discrete random field (an array of random variables) into a sequence of one dimensional, independent, discrete random processes via a unitary transformation. (For practical reasons, one desires this unitary transformation to have a fast implementation algorithm). Then each of these one dimensional processes is processed independently by one dimensional techniques such as recursive filtering (for image restoration applications) or predictive coding (such as DPCM, for image transmission/storage applications).

Hybrid processing was first proposed by Jain and Angel [2,3] for recursive filtering of two and three dimensional images. In [2] it was shown that certain stochastic image models yield vector recursive equations which can be decoupled into a sequence of scalar recursive equations by a fast unitary transform. In [3], an interframe filtering problem was shown to be solved by first taking a two dimensional, fast unitary transform of each image frame and then solving decoupled, scalar, recursive (Kalman) filter equations for each point of the transformed sequence of frames. Hybrid coding was first proposed by Habibi [1]for two dimensional images modeled by a separable covariance function [see Eqn. (4) of Chapter Two]. Subsequently, Roese, Pratt and Robinson [4] extended this technique for interframe coding of images.

In this report we consider certain stochastic representation of images called Semicausal Models, which yield hybrid coding algorithms. Such models

do not necessarily require that the image covariance function be separable. Once the parameters of a semicausal model are identified, the entire hybrid coding scheme can be designed. Adaptive hybrid coding schemes can then be designed by simply adapting the semicausal model to changes in image statistics. Some practical adaptive hybrid coding methods which strike a balance between complexity and performance are studied.

2. Summary of Semicausal Models

Four semicausal models have been considered. The first model is based on separable covariance statistics [Eqn. 4 of Chapter Two] that represents a Markov-2 field. It is used to compare results with other models and with adaptive algorithms. The second model, called the SC1 model, is a white noise driven semicausal model that also represents a Markov-2 field whose covariance function is nonseparable. This model is also a finite difference approximation of a partial differential equation representing a random diffusion phenomenon [5,6]. The third model, called the SC2 model, is a minimum variance representation of a Markov-2 field which is asymptotically equivalent to the first model. The SC2 model is introduced primarily to show that by imposing certain boundary conditions on a representation of the separable covariance model, the KL transform of any column of the associated random field becomes the Cosine transform and simplifies some of the analytic computations. The fourth model is called the Actual Measurements Model. Here, for each element of the Cosine transformed column we define a first order Markov process whose parameters are actually measured.

3. Summary of Experiments

The various experiments performed can be summarized in the following categories.

1. Nonadaptive Hybrid Coding

2. Adaptive Variance Estimation Hybrid Coding

3. Adaptive Classification Hybrid Coding

All the experiments were performed by dividing a 256 x 256 image into sixteen image strips each of size 16 x 256. Each image strip was coded independently. Figures 1 and 2 show a typical (nonadaptive) hybrid coding scheme. Each image column is unitarily transformed and the rows of this transformed image are DPCM coded. The predictor, quantizer, and the unitary transform designs are dependent on the particular image model. In our models, the unitary transform is always the Cosine transform (either by our choice or by the charactertics of the model). The main reasons for restricting ourselves to this transform are two. First, the hardware developed by NOSC* for hybrid coding utilizes this transform, and second, for one dimensional vectors whose elements are highly correlated and are Markov, the Cosine transform is a very good (fast transform) approximation to its (slow) KL transform [7,8,9].

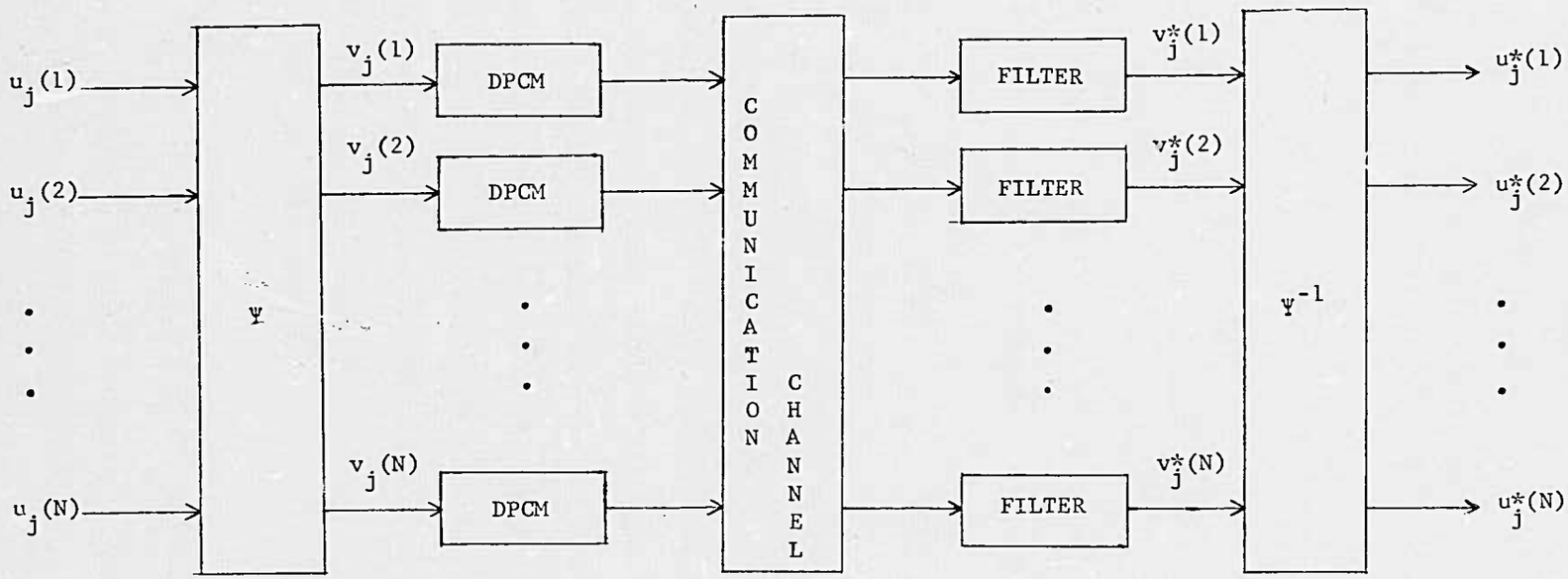
Figure 3 shows the adaptive variance estimation technique in hybrid coding. Here, the quantizer output is used to estimate the variance of the next quantizer input, in each DPCM channel. This variance is used to adjust the quantizer decision and reconstruction levels. Figure 4 shows the adaptive classification scheme. Here, each image column is classified into one of K classes and a different quantizer is used for each class. A larger number of quantization bits are allocated to image columns with high activity (measured by sample variance of that column) and fewer bits to columns with low activity. Each class of the image can have different model parameters.

4. Summary of Results and Conclusions

In summary, it was found that the performance of conventional nonadaptive hybrid coding can be improved upto 5.5 Db at low bit rates ($\simeq$ 1 bit/pixel) by

*Naval Ocean Systems Center

286



Ψ = One Dimensional Unitary Transform

FIGURE 1: A HYBRID CODING SYSTEM

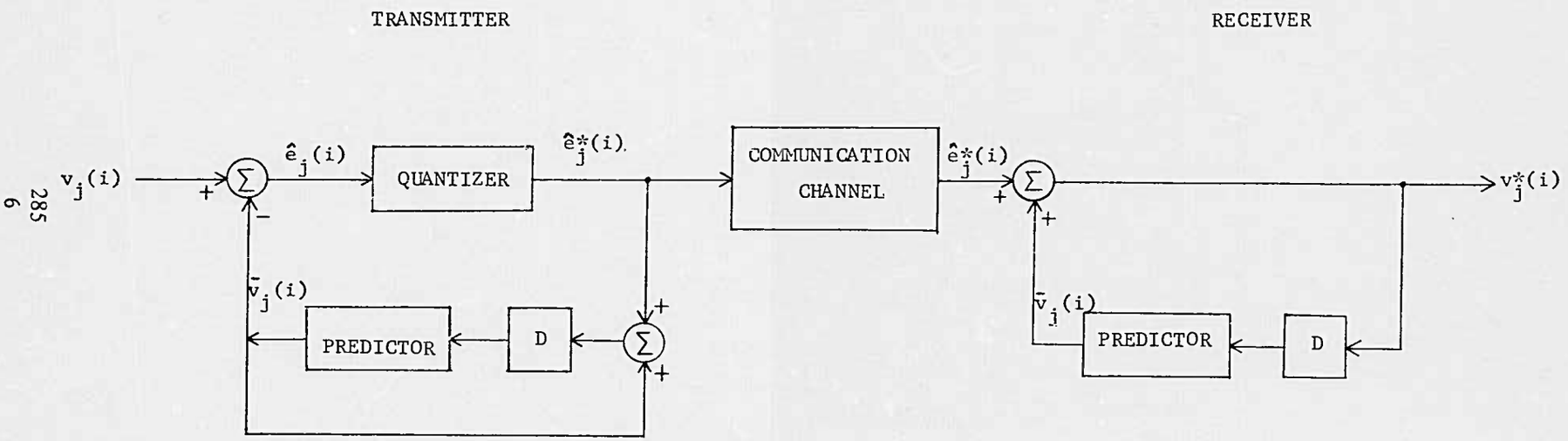


FIGURE 2 The i-th DPCM Channel

285 6

282

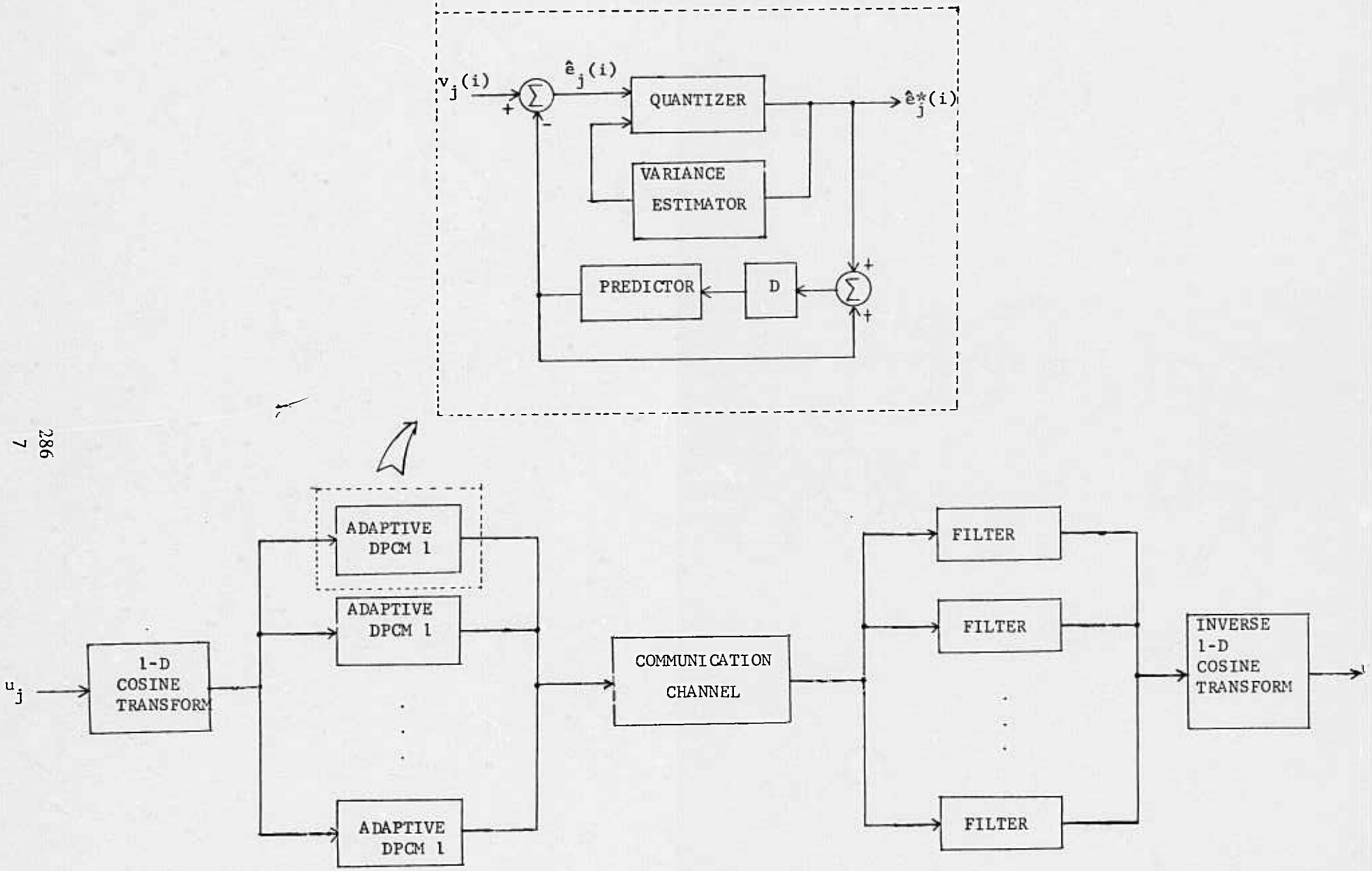
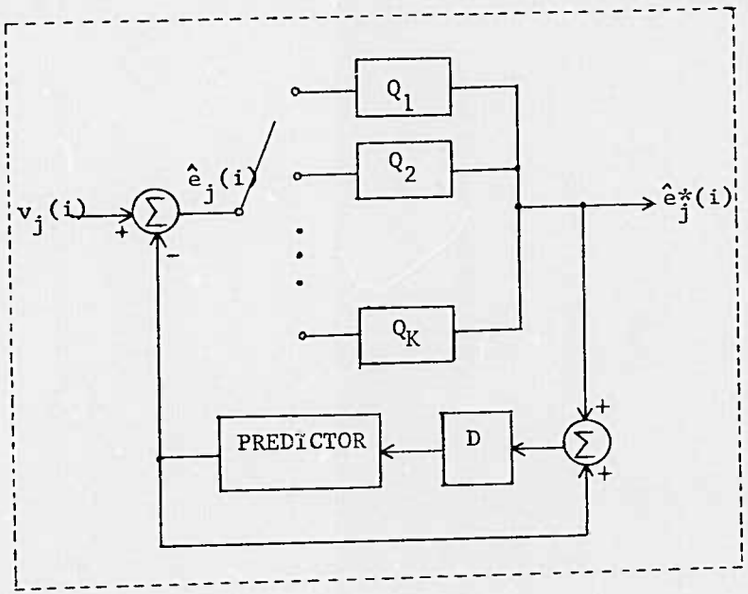


FIGURE 3 ADAPTIVE VARIANCE ESTIMATION (Hybrid Coding System 1)

6822

Q_k: indicates class k quantizer, $1 \le k \le K$



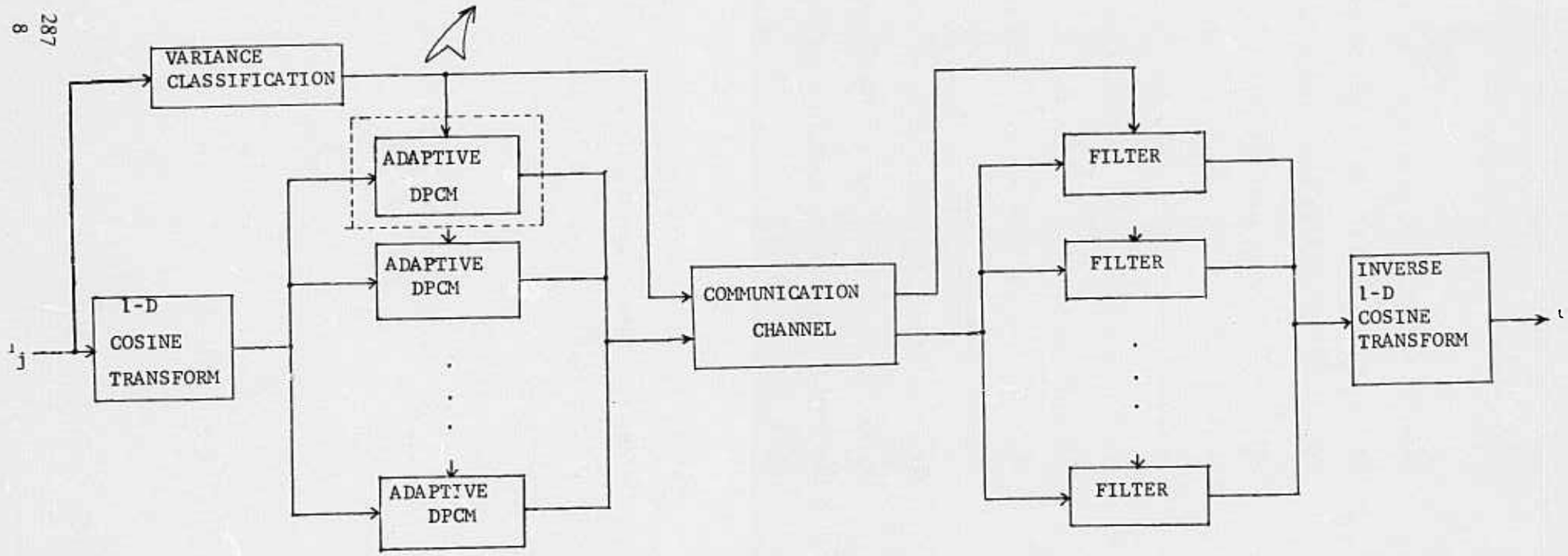


FIGURE 4 ADAPTIVE CLASSIFICATION (Hybrid Coding System 2)

290

improved statistical characterization of the image via more accurate models and/or via adapting the parameters of the chosen model to changes in image statistics. The actual measurements model applied in the adaptive classification coding scheme gave the best results. However, this model requires identification of a large number of parameters. In general, the SC1 model performed better than SC2 and separable covariance models by 2 to 3 Db. As expected from theoretical considerations the SC2 and separable covariance models performed almost identically.

The adaptive variance estimation method has minimal additional complexity over the nonadaptive scheme and is useful when image statistics deviates far from the model. It is most effective at relatively higher bit rates ($\geq$ 1 bit/pixel) and its performance improves as the size of the image column vector is increased. This scheme maintains a constant bit rate from one image column to the next.

The adaptive classification scheme is a variable bit rate (average bit rate from one column to the next can vary) technique which is robust (with respect to channel errors) and can yield varying degree of performances depending upon the coder complexity. It is relatively more effective at low bit rates ($\leq$ 1 bit/pixel) and its performance is degraded as the image column size increases.

Details of these results are given in Chapter Five.

CHAPTER TWO

SEMICAUSAL IMAGE MODELS AND HYBRID CODING

Let $\{u_{i,j}\}$ represent a zero mean, Gaussian, stationary random field whose covariance function is defined as

$$r(k,\ell) = Eu_{i,j}u_{i+k,j+\ell} \quad , \qquad \forall\, i,j \quad . \tag{1}$$

Consider an N x M segment of this random field where $1 \le i \le N$ and $1 \le j \le M$ and let u_j denote the jth column of the image, i.e.,

$$u_j = [u_{1,j}u_{2,j} \cdot \cdot \cdot \cdot \cdot u_{N,j}]^T \quad . \tag{2}$$

From (1) it follows that

$$Eu_j u_{j+\ell} = R_\ell \quad , \tag{3a}$$

where R_ℓ is an N x N Toeplitz matrix of elements

$$R_\ell(m,n) = r(m-n,\ell) \quad , \qquad 1 \le m,n \le N \quad . \tag{3b}$$

1. Hybrid Coding of Separable Covariance Image Fields

Consider the often used separable covariance model for images

$$r(k,\ell) = \sigma^2 \rho_v^{|k|} \rho_h^{|\ell|} \quad . \tag{4}$$

This gives

$$R_\ell = \sigma^2 \rho_h^{|\ell|} R_v \quad , \tag{5a}$$

where

$$R_v = \{\rho_v^{|m-n|}\} \quad . \tag{5b}$$

292

It is well known that R_v is the covariance of a zero mean, first order stationary Markov process whose one step correlation is ρ_v. Let Ψ denote the Karhunen Loeve (KL) transform of this process, i.e., Ψ is a unitary matrix that satisfies

$$\Psi R_v \Psi^T = \Lambda \quad , \qquad \Psi\Psi^T = I \quad , \tag{6}$$

where $\Lambda = \{\lambda_i\}$ is the diagonal matrix of eigenvalues of R_v. Define the transformation

$$v_j \overset{\Delta}{=} \Psi u_j \quad . \tag{7}$$

Using (5a), (6) and (7), one obtains

$$Ev_j v_{j+\ell}^T = \Psi R_\ell \Psi^T = \sigma^2 \rho_h^{|\ell|} \Lambda \quad , \tag{8a}$$

$$\text{or} \quad Ev_j(i)v_{j+\ell}(i+k) = \sigma^2 \rho_h^{|\ell|} \lambda_i \delta_{k,0} \quad . \tag{8b}$$

Eqn. (8b) implies that (i) the elements of the transformed column v_j are mutually uncorrelated and (ii) for each i, the sequence $\{v_j(i),\ 1 \le j \le M\}$ is a first order, stationary Markov process. Therefore, we can write

$$v_j(i) = \rho_h v_{j-1}(i) + e_j(i) \quad , \tag{9a}$$

where

$$Ee_j(i) = 0, \quad E\, e_j(i)e_{j+\ell}(i+k) = \sigma^2(1-\rho_h^2)\lambda_i \delta_{\ell,0}\delta_{k,0} \quad , \tag{9b}$$

and $\{v_j(i),\ 1 \le i \le N\}$ are the elements of the vector v_j. Eqns. (9) spell the so called Hybrid Coding method for the random field $\{u_{i,j}\}$. First, each column u_j is unitarily transformed (ideally by the KL transform) to obtain v_j and for each sequence $\{v_j(i),\ j = 1,2. \ . \ .\}$ for $1 \le i \le N$, an independent DPCM channel is used. This overall scheme is shown in Fig. 1 . Fig. 2 shows the

details of the ith DPCM channel and the reconstruction filter at the receiver. Often, a fast unitary transform is used to replace the KL transform to make the scheme more practical [1]. For video image data where $\rho_h \simeq \rho_v \simeq 0.95$, the Cosine transform performs very close to the KL transform defined in (6) [5-7].

From our foregoing discussion, if any two dimensional random field $\{u_{i,j}\}$ can be transformed via a unitary transformation to a sequence of independent one dimensional Markov processes (of any order), then a Hybrid Coding scheme for transmission of $\{u_{i,j}\}$ can be designed. In the sequel we will consider representations of random Gaussian fields which are Markov-2 and which yield the above mentioned stochastic decoupling via a fast unitary transform.

2. White Noise Driven Semicausal Representation (SC1 Model)

A semicausal image representation is defined as one which is causal in one of the image coordinates (say, j) and is noncausal in the other (say, i). Such representations are of the type [10]

$$u_{i,j} = \sum_{\forall k \neq 0} a_k u_{i-k,j} + \sum_{\ell \geq 1, \forall k} \sum a_{k,\ell} u_{i-k,j-\ell} + \varepsilon_{i,j}, \qquad \forall i,j \quad . \tag{10}$$

For white noise driven representations $\{\varepsilon_{i,j}\}$ is a white noise field. If we restrict ourselves to white noise driven semicausal models that are (discrete) Markov-2 random fields [10], then (10) must be of the form

$$u_{i,j} = \alpha_1 u_{i-1,j} + \alpha_{-1} u_{i+1,j} + \gamma u_{i,j-1} + \varepsilon_{i,j} \quad . \tag{11}$$

Further, if we assume that the image field is homogeneous along each of its individual coordinates, i.e.,

$$r(k,\ell) = r(|k|, |\ell|) \quad , \tag{12}$$

then it can be shown that [10]

$$\alpha_1 = \alpha_{-1} = \alpha \text{ (say)} \quad , \tag{13}$$

which gives

$$u_{i,j} = \alpha(u_{i-1,j} + u_{i+1,j}) + \gamma u_{i,j-1} + \epsilon_{i,j}, \quad \forall i,j \quad , \tag{14a}$$

where

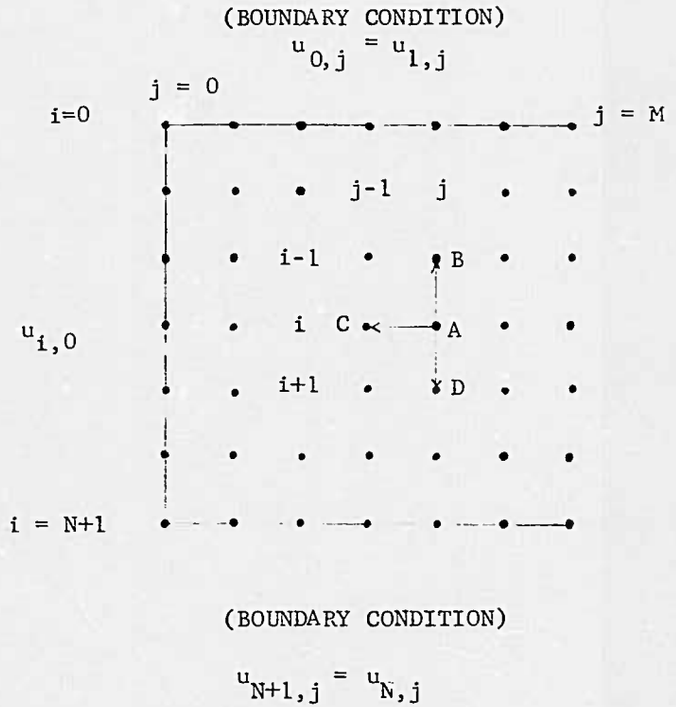
$$E\epsilon_{i,j}\epsilon_{k,\ell} = \beta^2 \delta_{i,k}\delta_{j,\ell} \quad , \tag{14b}$$

$$\alpha < \tfrac{1}{2}, \quad \text{and} \quad |2\alpha + \gamma| < 1 \ . \tag{14c}$$

Figure 5 shows the spatial structure of this model.
Assumption (12) is a reasonable one because often the orientation of the image is unknown and (12) makes the parameters of our model independent of 90° rotations. (Note that regardless of the image orientation, (14a) is causal in the "j" variable and is noncausal in the "i" variable). The model of (14a) also arises by a finite difference approximation of a parabolic partial differential equation called the diffusion equation [5]. Conditions of (14c) insure the stability of the finite difference scheme. In (14a), these conditions also guarantee that the two dimensional spectral density function of $\{u_{i,j}\}$ is positive. In the sequel we will call (14a) to be the SC1 (semicausal-1) model.

Figure 6a shows a plot of typical contours of the covariance function realized by the SC1 model for the GIRL image. This covariance function is nonseparable in the two dimensions and provides a better fit than the separable covariance function of (4), for various images. (for details see [5]). Figure 6b shows the contours of the actual (measured) covariance and the separable covariance functions for the same image. Here the parameters α and γ of (14a) and ρ_v, ρ_h of (4) were optimized to minimize the covariance function least squares error.

293



$$u_A = \alpha(u_B + u_D) + \gamma u_C + \epsilon_A$$

FIGURE 5: SPATIAL STRUCTURE OF SC1 MODEL

296

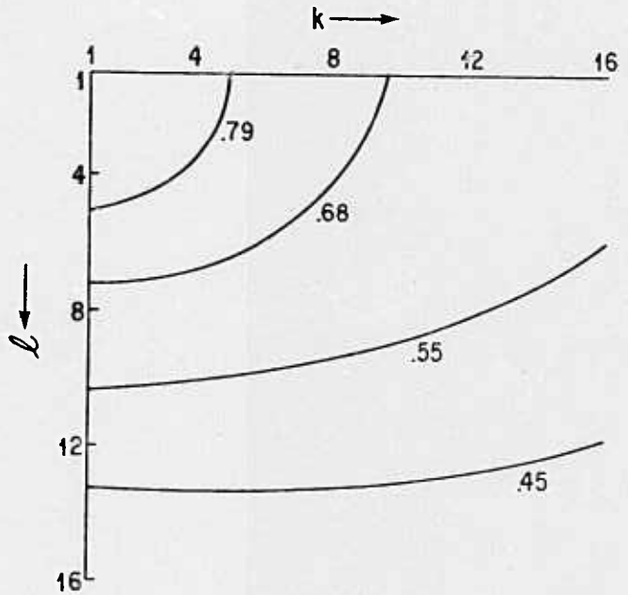


FIGURE 6a: COVARIANCE FUNCTION CONTOURS OF SC1 MODEL FOR GIRL IMAGE

297

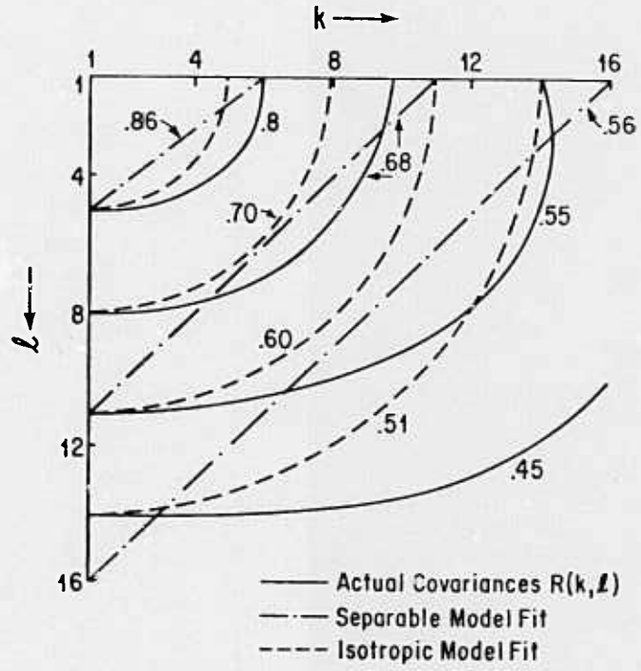


FIGURE 6b: COVARIANCE FUNCTION CONTOURS FOR GIRL IMAGE

3. Minimum Variance Semicausal Representations (SC2 Model)

As discussed in [10], for a minimum variance representation, the coefficients $\{a_k\}$ and $\{a_{k,\ell}\}$ in (10) are determined such that the variance of the residual $\epsilon_{i,j}$ is minimized for all i,j. The orthogonality condition for minimum variance requires

$$E u_{i-k,j-\ell}\epsilon_{i,j} = \beta^2 \delta_{k,0}\delta_{\ell,0} \quad , \quad \text{for } \ell \geq 0, \forall k \quad , \tag{15a}$$

$$E \epsilon_{i-k,j-\ell}\epsilon_{i,j} = -a_k \beta^2 \delta_{\ell,0} \quad , \quad \text{for } \forall \ell, k \quad , \tag{15b}$$

$$\text{where} \quad a_0 \overset{\Delta}{=} -1 \quad . \tag{15c}$$

From (15b) we see that $\{\epsilon_{i,j}\}$ is not a white noise field because it is correlated in the "i" variable. Since (15b) defines an autocovariance, we must have

$$a_k = a_{-k} = a_{|k|} \quad , \tag{15d}$$

i.e., $\{a_k\}$ is an even sequence. The separable covariance function of (4) has a semicausal realization given by [5,11]

$$u_{i,j} = \alpha\,(u_{i-1,j} + u_{i+1,j}) - \rho_h \alpha\,(u_{i-1,j-1} + u_{i+1,j-1}) + \rho_h u_{i,j-1} + \epsilon_{i,j} \quad , \tag{16}$$

where $\alpha = \rho_v/(1 + \rho_v^2)$,

and $\{\epsilon_{i,j}\}$ is a random field whose covariances are given by

$$E \epsilon_{i,j}\epsilon_{i-k,j-\ell} = \begin{cases} \beta^2 \delta_{\ell,0} \, , & k = 0 \\ -\alpha\, \beta^2 \delta_{\ell,0} \, , & k = \pm 1 \\ 0 \, , & \text{otherwise} \end{cases} \quad . \tag{17}$$

This representation will be called the SC2 Model. It can be shown that this model represents a Markov-2 field (whose covariance is the separable covariance function of (4)). There are other minimum variance semicausal representations which would yield Markov-2 fields [10]. However, here we will only be interested in the SC2 Model. Figure 7 shows the spatial structure of this model.

4. Semicausal Representations as Two Source Models for Finite Random Fields

Consider the given image to be an N x M segment of the infinite, stationary random field. Eqns. (14a) and (16), when written in an N x 1 vector notation are of the form

$$Qu_j = Pu_{j-1} + \varepsilon_j + b_j \quad , \tag{18}$$

where Q is a tridiagonal Toeplitz matrix defined by

$$q_{i,j} = \begin{cases} 1, & i = j \\ -\alpha, & |i-j| = 1 \\ 0, & \text{otherwise} \end{cases} \quad , \tag{19a}$$

and

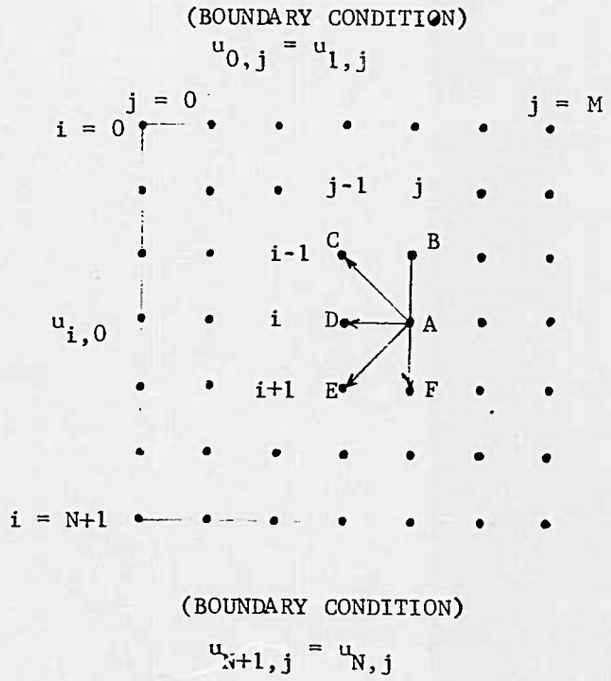
$$P = \begin{cases} \gamma I, & \text{for SC1} \\ \rho_h Q, & \text{for SC2} \end{cases} \quad , \tag{19b}$$

$$b_j = \begin{cases} \alpha[u_{0,j}, \overbrace{0, \ldots\ldots 0}^{\leftarrow N-2 \text{ terms} \rightarrow}, u_{N+1,j}]^T \quad , & \text{for SC1} \\ \alpha[b_{1,j}, \underbrace{0, \ldots\ldots 0}_{\leftarrow N-2 \text{ terms} \rightarrow}, b_{N,j}]^T \quad , & \text{for SC2} \end{cases} \quad , \tag{19c}$$

$$b_{1,j} = u_{0,j} - \rho_h u_{0,j-1} \quad ,$$

$$b_{N,j} = u_{N+1,j} - \rho_h u_{N+1,j-1} \quad . \tag{19d}$$

300



$$u_A = \alpha\,(u_B + u_F) - \alpha\,\rho_h(u_C + u_E) + \rho_h u_D + \epsilon_A$$

FIGURE 7: SPATIAL STRUCTURE OF SC2 MODEL

301

Thus, the N x 1 vector b_j contains only two nonzero entries which depend only on the boundary variables $\{u_{0,j}\}$ and $\{u_{N+1,j}\}$ of the jth image column. Note from (2), that the jth image column, u_j contains only N elements. From (14b) and (17), the covariance of ϵ_j is

$$R_\epsilon = E\epsilon_j\epsilon_k^T = \begin{cases} \beta^2 I\delta_{j,k} , & \text{for SC1} \\ \beta^2 Q\delta_{j,k} , & \text{for SC2} \end{cases} , \tag{19e}$$

where $\beta^2 = (1 - \rho_v^2)(1 - \rho_h^2) \Big/ (1 + \rho_v^2)$ for the SC2 model. (19f)

Eqn. (18) can be looked at, in general, as a two source model for the vector sequence $\{u_j\}$, whose statistics are specified by the statistics of the "source inputs", $\{\epsilon_j\}$ and $\{b_j\}$. We note that (18) may not always qualify to be a vector Markov model

$$Qu_j = Pu_{j-1} + f_j , \tag{20a}$$

by defining

$$f_j = \epsilon_j + b_j . \tag{20b}$$

This is because the sequence $\{f_j\}$ is not guaranteed to be white, even though $\{\epsilon_j\}$ is such. From the linearity of (18), the two source outputs can be identified by the solution of the equations

$$Qu_j^o = Pu_{j-1}^o + \epsilon_j , \tag{21a}$$

$$Qu_j^b = Pu_{j-1}^b + b_j , \tag{21b}$$

and the image field is the composition [5]

$$u_j = u_j^o + u_j^b . \tag{22}$$

302

The second source output u_j^b depends completely on the boundary variables. Hence, as the size of the image column gets larger, one would expect a smaller contribution of u_j^b in influencing the statistical properties of u_j. The N x M random field will be stationary when the statistics of the boundary variables are consistent with those of $\{u_{i,j}\}$. In other words, the case of stationary statistics is only a special case of (22). For minimum variance representations, the orthogonality condition (15a) implies that the two source outputs u_j^o and u_j^b are orthogonal. For stationary, white noise driven, semicausal models however, one does not have this condition. If one starts by specifying the boundary variables to be orthogonal to the stationary white noise field $\{\epsilon_{i,j}\}$, then the finite N x M field $\{u_{i,j}\}$ need not be stationary. Asymptotically, $(N,M\to\infty)$, Eqns. (21) and (22) will yield stationary random field $\{u_{i,j}\}$ (assuming Q is always nonsingular and (18) is a stable recursive relation).

5. Stochastic Decoupling of Semicausal Models by the Cosine Transform

For a large number of image fields, the adjacent elements of a column (or a row) are highly correlated and often the correlation parameter has a value around 0.95. Thus, for a finite N x M random field, it is reasonable to assume that the boundary variables that need to be specified in (18) satisfy the conditions

$$u_{0,j} = u_{1,j} \,,$$

$$u_{N+1,j} = u_{N,j} \,. \tag{23}$$

This means the two outermost boundary rows of the (N+2) x M field $\{u_{i,j};\ 0 \le i \le N+1,\ 1 \le j \le M\}$ are equal. This also means that (18) subject to (23) is no longer a stationary field on the (N+2) x M array of points. Using (23) in (18) and rearranging terms, we get a one source model (the second

source is deterministic, as defined by (23))

$$Q_c u_j = Pu_{j-1} + \epsilon_j \quad , \tag{24}$$

where Q_c is defined as

$$q_c(i,j) = \begin{cases} 1-\alpha, & i = j = 1 \text{ or } i = j = N \\ 1, & i = j, \ 2 \le i \le N-1 \\ -\alpha, & |i-j| = 1 \\ 0, & \text{otherwise} \end{cases} ,$$

i.e.,

$$Q_c = \begin{bmatrix} 1-\alpha & -\alpha & & & \\ -\alpha & 1 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 1 & -\alpha \\ & & & -\alpha & 1-\alpha \end{bmatrix} , \tag{25a}$$

and

$$P = \begin{cases} \gamma I, & \text{for SC1} \\ \rho_h Q_c, & \text{for SC2} . \end{cases} \tag{25b}$$

For SC1 model, $\{\epsilon_{i,j}\}$ is still a stationary white noise field. For SC2 model, (24) should satisfy the minimum variance conditions of (15). When applied to (24), this implies

$$E\epsilon_j \epsilon_k^T = \beta^2 Q_c \delta_{j,k} .$$

Hence, the oovariance of ϵ_j is now given as

$$R_\epsilon = E\epsilon_j \epsilon_k^T = \begin{cases} \beta^2 I \delta_{j,k} & \text{for SC1} \\ \beta^2 Q_c \delta_{j,k} & \text{for SC2} . \end{cases} \tag{26}$$

304

Proposition 1:

Asymptotically, as $N,M \to \infty$ the random fields of (24) and (18) are equivalent.

Proof:

The proof is obtained by inspection of (19a) and (25a). Since Q and Q_c differ only at the boundary elements of their diagonals and the vector b_j in (18) is nonzero only at the first and last elements, for infinite vectors (18) and (24) are identical. Also, since (24) is a stable recursive equation, solutions of (24) and (18) in the steady state would also be identical (regardless of initial condition u_0).

Proposition 2:

The KL transform of any vector u_j of the random field of (24) is the discrete Cosine transform (DCT) for SC1 as well as SC2 models.

Proof:

From (25b), first we see that the matrices P and Q_c commute. Hence, they have a common set of eigenvectors. Next, from [8,9] we use the result that the orthogonalized eigenvectors of the tridiagonal matrix Q_c are the rows of the DCT, which is defined as

$$\Psi_{i,j} = \begin{cases} \sqrt{\frac{1}{N}}, & i = 1, \quad 1 \le j \le N \\ \sqrt{\frac{2}{N}} \cos \frac{(2j-1)(i-1)\pi}{2N}, & 2 \le i \le N, \ 1 \le j \le N \end{cases} . \tag{27}$$

The eigenvalues of Q_c are given by

$$\lambda_{ci} = 1 - 2\alpha \cos\frac{(i-1)\pi}{N}, \qquad 1 \le i \le N . \tag{28}$$

305

The DCT matrix Ψ has a fast implementation algorithm similar to the fast Fourier transform (FFT) [7]. Also, it diagonalizes the matrices Q_c and P and satisfies the relations

$$\Psi Q_c \Psi^T = \Lambda_c , \qquad \Psi\Psi^T = I \tag{29a}$$

$$\Psi P \Psi^T = \Gamma \quad , \qquad \gamma_i = \begin{cases} \gamma & , \quad \text{for SC1} \\ \rho_h \lambda_{ci} , & \quad \text{for SC2} . \end{cases} \tag{29b}$$

The KL transform of a vector u_j is a unitary matrix Ψ such that the transformation

$$v_j = \Psi u_j \quad , \tag{30}$$

gives uncorrelated components to the vector v_j. This means Ψ must also satisfy

$$E[v_j v_j^T] = \Psi [E u_j u_j^T] \Psi^T$$

$$= \Psi R_j \Psi^T = \text{Diagonal} . \tag{31}$$

Multiplying both sides of (24) by Ψ and using (29a), (29b) and (30) we obtain

$$\Lambda_c v_j = \Gamma v_{j-1} + e_j , \tag{32a}$$

where $e_j \overset{\Delta}{=} \Psi \varepsilon_j$. (32b)

Since Λ_c and Γ are diagonal, (32a) is a set of decoupled equations

$$\lambda_{ci} v_j(i) = \gamma_i v_{j-1}(i) + e_j(i) \quad , \qquad 1 \le i \le N . \tag{33}$$

From (26) and (29a) and (32b) $\{e_j(i)\}$ is a white noise field which satisfies the equation

306

$$Ee_j(i)e_k(\ell) = \begin{cases} \beta^2\delta_{j,k}\delta_{i,\ell} , & \text{for SC1} \\ \beta^2\lambda_{ci}\delta_{j,k}\delta_{i,\ell}, & \text{for SC2} . \end{cases} \tag{34}$$

Hence, for each i, the sequence $\{v_j(i), \forall\ j\}$ is a first order Markov sequence. Moreover, if the initial variables $\{v_0(i), \quad 1 \le i \le N\}$ are uncorrelated then for $\forall\ j > 0, \{v_j(i), \quad 1 \le i \le N\}$ will be uncorrelated elements of the vector v_j. Also, for $u_0 = 0$ or $u_0 = u_\infty$ (i.e., steady state in j), the above condition will always be satisfied. For arbitrary initial conditions, say $u_0 = c$ where c is an arbitrary random vector, u_j will have a decomposition

$$u_j = u_j^o + u_j^c \quad , \tag{35a}$$

$$\text{where } Q_c u_j^o = P u_{j-1}^o + \epsilon_j \quad , \qquad u_0^o = 0 \quad , \tag{35b}$$

$$\text{and } Q_c u_j^c = P u_{j-1}^c \quad , \qquad u_0 = c \quad . \tag{35c}$$

The KL transform of u_j^o now will be the DCT. In a practical DPCM system, the statistics of the initial condition does not affect the ultimate system performance because the initial vector can always be quantized accurately and transmitted before starting the DPCM transmission.

The steady state covariance of u_j is obtained from (24) as

$$R_0 \overset{\Delta}{=} Eu_j u_j^T = \begin{cases} \beta^2(Q_c^2 - \gamma^2 I)^{-1} , & \text{for SC1} \\ \beta^2(1 - \rho_n^2)^{-1} Q_c^{-1} , & \text{for SC2} . \end{cases} \tag{36}$$

From (29a) and (36) it immediately follows that R_0 satisfies (31). Hence, the DCT is the KL transform of u_j and (33) defines a sequence of stochastically decoupled (independent) first order Markov processes.

6. Stochastic Decoupling of Stationary Semicausal Fields

Random fields for which (23) is not a reasonable assumption (i.e., one step correlation in the "j" direction is not close to 1) or when stationarity of the random field is an essential requirement of the model, we have to restrict ourselves to (18). The boundary variables (elements of b_j) are random variables which are samples from the stationary random field $\{u_{i,j}\}$ and cannot be specified independently. In that case, (18) admits the decomposition [see (21) - (22)]

$$Qu_j^o = Pu_{j-1}^o + \epsilon_j \quad , \quad u_0^o = 0 \quad , \quad j \geq 1 \quad , \tag{37a}$$

$$Qu_j^b = Pu_{j-1}^b + b_j \quad , \quad u_0^b = u_0 \quad , \quad j \geq 1 \quad , \tag{37b}$$

where

$$u_j = u_j^o + u_j^b \quad , \qquad j \geq 1 \quad , \tag{38}$$

and u_0 is the initial value of the sequence $\{u_j\}$.

Proposition 3:

The KL transform of any column vector u_j^o , $j \geq 1$ of the random field $\{u_{i,j}^o\}$ of (37a) is the Sine transform for SC1 as well as the SC2 models.

Proof:

The Sine transform Ψ is defined as [11]

$$\Psi_{i,j} = \sqrt{\frac{2}{N+1}} \sin \frac{ij\pi}{N+1} \quad , \qquad 1 \leq i,j \leq N \quad . \tag{39}$$

It is well known that Ψ is a unitary transform and contains the eigenvectors of any symmetric, tridiagonal, Toeplitz matrix e.g., Q, and satisfies

$$\Psi = \Psi^T \quad ,$$

$$\Psi Q \Psi = \Lambda = \text{Diagonal } \{\lambda_i\} \quad ,$$

$$\Psi\Psi = I \quad , \tag{40}$$

$$\lambda_i = 1 - 2\alpha \cos \frac{i\pi}{N+1} \quad , \quad 1 \le i \le N \quad .$$

Like the DCT, the Sine transform also has a fast algorithm for implementing a transformation of the type

$$v_j = \Psi u_j^o \quad , \quad e_j = \Psi \epsilon_j \quad . \tag{41}$$

Multiplying both sides of (37a) by Ψ and using (40), (41), (19b) and (19e) we get a sequence of decoupled equations

$$\lambda_i v_j(i) = \gamma_i v_{j-1}(i) + e_j(i) \quad , \quad v_0(i) = 0, \quad 1 \le i \le N, \quad j \ge 1 \quad , \tag{42a}$$

$$Ee_j(i)e_k(\ell) = \begin{cases} \beta^2 \delta_{i,\ell} \delta_{j,k} \quad , & \text{for SC1} \\ \beta^2 \lambda_i \delta_{i,\ell} \delta_{j,k} \quad , & \text{for SC2} \quad , \end{cases} \tag{42b}$$

$$\gamma_i = \begin{cases} \gamma \quad , & \text{for SC1} \\ \rho_h \lambda_i , & \text{for SC2} \quad . \end{cases} \tag{42c}$$

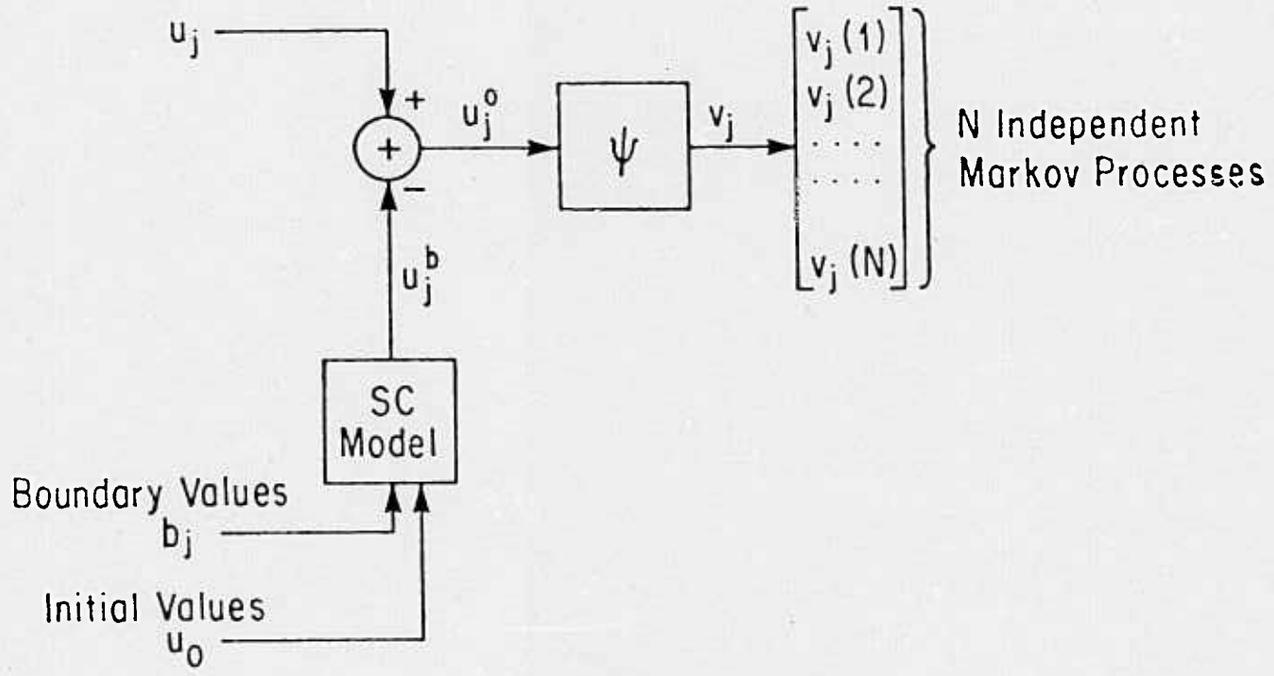
From (42a) and (42b) it is evident that $\{v_j(i)\}$ is a sequence of N, first order Markov processes and for any $j \ge 1$ the elements $\{v_j(i), \ 1 \le i \le N\}$ are uncorrelated. Hence, Ψ is the KL transform of u_j^o. Figure (8a) shows the realization of decomposition of (38) and stochastic decoupling of (42). Figures (8b) and (8c) show application of this decomposition for hybrid coding. The boundary variables are quantized first, from which the reconstructed vectors u_{bj}^* are calculated via (37b). The difference $u_j - u_{bj}^* = \tilde{u}_j^o$

is close to u_j^o (except for quantization effects of b_j^*) and is hybrid coded. The differential signal $\tilde{e}_j^*(i)$ and the boundary variables b_j^* are transmitted. The receiver simply solves (37c), (37b) and (38) to obtain u_j^*.

310

FIGURE 8a

REALIZATION OF SEMICAUSAL MODEL DECOMPOSITION



ψ = Fast Sine Transform

u_j = jth. Image Column

311

FIGURE 8b

FAST KL-DPCM (HYBRID) IMAGE DATA COMPRESSION VIA SEMI-CAUSAL P.D.E. MODELS

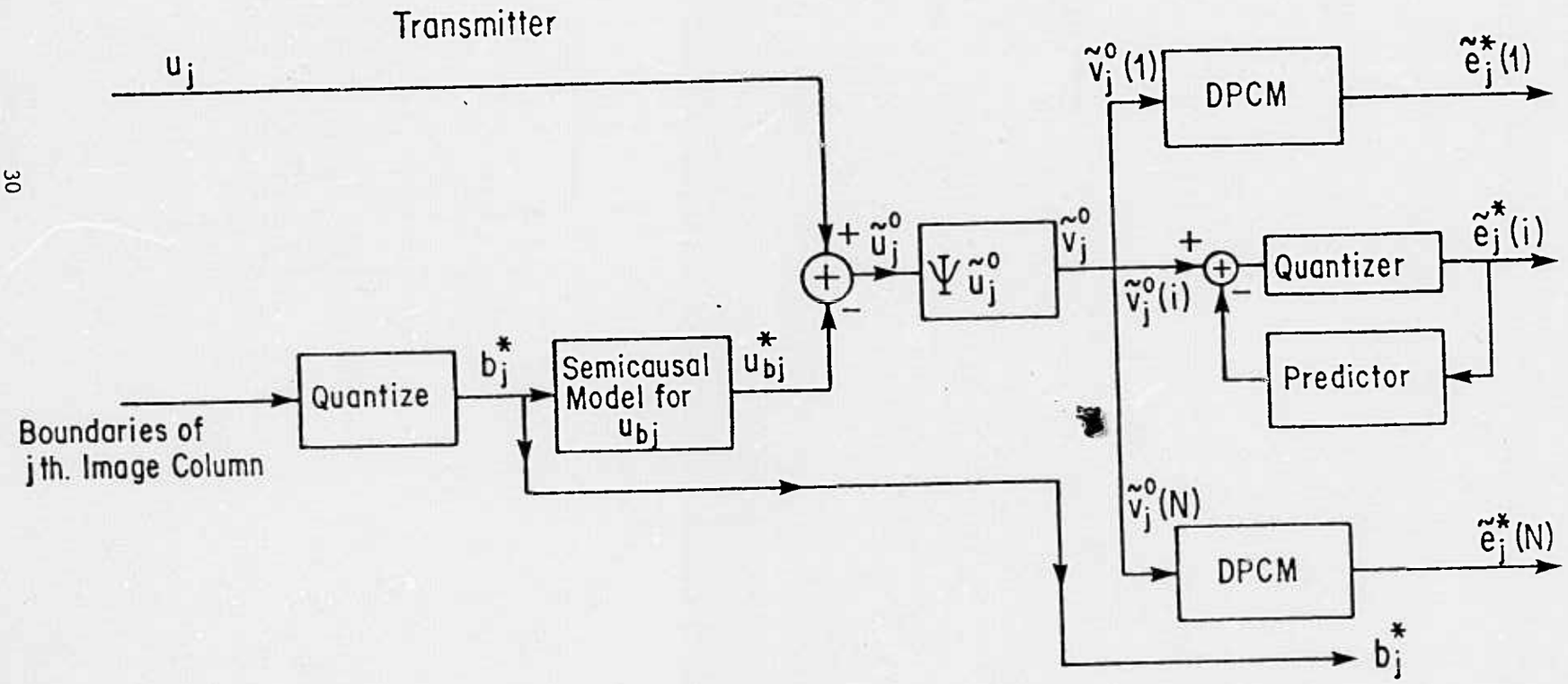
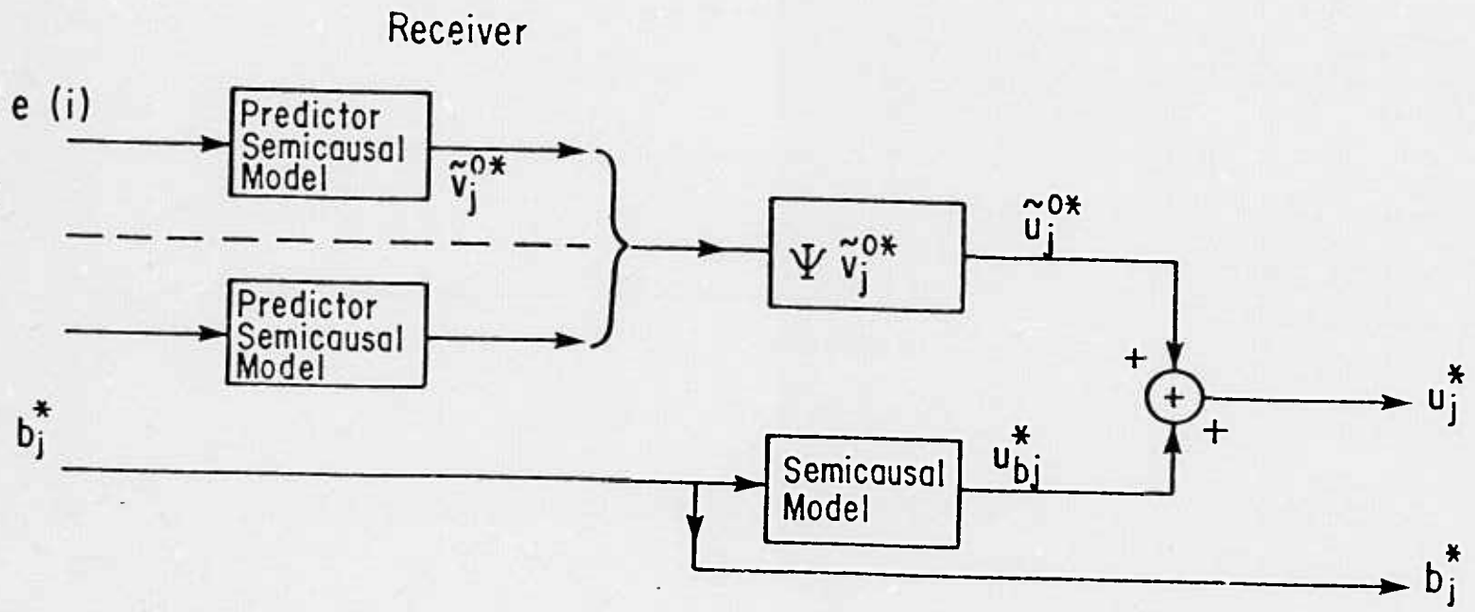


FIGURE 8c

FAST KL-DPCM (HYBRID) IMAGE DATA COMPRESSION VIA SEMI-CAUSAL P.D.E. MODELS



CHAPTER THREE

HYBRID CODER DESIGN FOR SEMICAUSAL RANDOM FIELDS

For the sake of simplicity of presentation, we will only consider the semicausal models of (2.18) which are decoupled by the Cosine transform. The sequence of decoupled equations (2.29b), (2.33) and (2.34) can be rewritten as

$$v_j(i) = \rho_i v_{j-1}(i) + e_j(i), \qquad 1 \le i \le N , \tag{1}$$

where

$$\rho_i \overset{\Delta}{=} \frac{\gamma_i}{\lambda_{ci}} = \begin{cases} \gamma/\lambda_{ci} , & \text{for SC1} \\ \rho_h , & \text{for SC2} , \end{cases} \tag{2a}$$

and $\{e_j(i)\}$ is a white noise field whose covariances are given by

$$Ee_j(i)e_k(\ell) \overset{\Delta}{=} \beta^2(i)\delta_{j,k}\delta_{i,\ell} ,$$

$$\beta^2(i) \overset{\Delta}{=} \begin{cases} \beta^2 / \lambda_{ci}^2 , & \text{for SC1} \\ \beta^2 / \lambda_{ci} , & \text{for SC2} . \end{cases} \tag{2b}$$

1. DPCM Equations

Since (1) is a sequence of N independent Markov processes, each of these sequences could be transmitted via an independent DPCM channel, as shown in Figures 1 and 2 . The equation at the transmitter and receiver are simply as follows.

Predictor: $\bar{v}_j(i) = \rho_i v^*_{j-1}(i)$ (3a)

Differential Signal: $\hat{e}_j(i) = v_j(i) - \bar{v}_j(i)$ (3b)

314

Quantizer Output: $e_j^*(i)$ (3c)

Reconstruction Filter: $v_j^*(i) = \rho_i v_{j-1}^*(i) + e_j^*(i)$ (3d)

The encoding scheme requires, first, to take the Cosine transform of each column vector u_j. The DCT is a fast transform and is also the KL transform of each column of the semicausal image fields characterized by (2.24). This is followed by N DPCM channels for predictive coding of successive Cosine transformed vectors. The receiver simply reconstructs the transformed vectors according to (3d) and performs the inverse Cosine transformation. To complete the design, we now need to specify the quantizers in the various DPCM channels.

2. Bit Allocation

We will assume all the quantizers are identical in their characteristics (i.e., in rate vs. distortion). Let

P = Average desired bit rate in bits/pixel of the image field.

$f(n)$ = Mean square distortion of a quantizer with 2^n levels for unit variance input random variables. This is a positive, monotonically decreasing convex function. We will assume $f(n)$ has a continuous first derivative.

$0 \le n_i$ = Number of bits per pixel, allocated to the ith DPCM channel.

From the definition of P, we have

$$\frac{1}{N} \sum_{i=1}^{N} n_i = P \quad , \quad n_i \ge 0 \quad . \tag{4}$$

The average mean square distortion in encoding of the image vector u_j is given by

$$D = \frac{1}{N} E\delta u_j^T \delta u_j \quad , \tag{5}$$

where $\delta u_j \overset{\Delta}{=} u_j - u_j^*$.

Since $v_j = \Psi u_j$ is a unitary transformation and $v_j(i)$ are coded via DPCM we have

$$\begin{aligned} \delta v_j(i) &= v_j(i) - v_j^*(i) \\ &= v_j(i) - \rho_i v_{j-1}^*(i) - e_j^*(i) \\ &= \hat{e}_j(i) - e_j^*(i) \\ &= \delta\hat{e}_j(i), \end{aligned} \tag{6}$$

i.e., the reconstruction error in $v_j(i)$ (assuming noiseless channel) equals the quantization error in the ith channel. This gives

$$\begin{aligned} D &= \frac{1}{N} E\delta v_j^T \delta v_j \\ &= \frac{1}{N} \sum_{i=1}^{N} E(\delta e_j(i))^2 \quad . \end{aligned} \tag{7}$$

Thus, the average distortion per pixel is the average quantizer distortion. If $\hat{\beta}_j^2(i)$ denotes the variance of the jth quantizer input in the ith DPCM channel, then

$$E(\delta\hat{e}_j(i))^2 = \hat{\beta}_j^2(i) f(n_i), \qquad \forall j \quad . \tag{8}$$

Using (1), (3a) and (7) in (3b) and simplifying we get

$$\begin{aligned} \hat{e}_j(i) &= v_j(i) - \bar{v}_j(i) \\ &= e_j(i) + \rho_i \delta\hat{e}_{j-1}(i) . \end{aligned} \tag{9}$$

316

Eqns. (8) and (9) together give a recursive relation for the variance of $\hat{e}_j(i)$ as

$$\hat{\beta}_j^2(i) = \beta^2(i) + \rho_i^2 \hat{\beta}_{j-1}^2(i) f(n_i) , \tag{10}$$

where we have used the fact that $e_j(i)$ is uncorrelated with the quantization error at step j-1. Eqn. (10) is a recursion in $\hat{\beta}_j^2(i)$ and solves to give

$$\hat{\beta}_j^2(i) = \frac{1 - [\rho_i^2 f(n_i)]^j}{1 - \rho_i^2 f(n_i)} \beta^2(i) . \tag{11a}$$

In steady state this gives

$$\hat{\beta}_j^2(i) \overset{\Delta}{=} \hat{\beta}^2(i) = \frac{\beta^2(i)}{1 - \rho_i^2 f(n_i)} . \tag{11b}$$

Hence, (8) becomes

$$E(\delta \hat{e}_j(i))^2 = \frac{\beta^2(i) f(n_i)}{1 - \rho_i^2 f(n_i)} , \tag{12}$$

giving

$$D = \frac{1}{N} \sum_{i=1}^{N} g_i(n_i) \beta^2(i) , \tag{13a}$$

where

$$g_i(x) = \frac{f(x)}{1 - \rho_i^2 f(x)} , \qquad x \geq 0 . \tag{13b}$$

The function $g_i(x)$ represents the mean square distortion of the ith DPCM loop of a Markov process whose prediction error variance is unity. Like the function f(x), for $|\rho_i| < 1$, each $g_i(x)$ is also a positive, monotonically decreasing, convex function with a continuous first derivative. (Proof of this statement is quite simple and is not given here).

The bit allocation problem is to minimize (13a) subject to the constraints of (4). This is equivalent to finding

$$D' = \min_{\substack{n_i \geq 0 \\ \lambda}} \left\{ \lambda P - \frac{1}{N} \sum \left[g_i(n_i)\beta^2(i) + \lambda n_i \right] \right\}$$

$$= \min_{\lambda} \left\{ \lambda P - \frac{1}{N} \sum_{i=1}^{N} \max_{n_i \geq 0} \left[g_i(n_i)\beta^2(i) + \lambda n_i \right] \right\} . \qquad (14)$$

Solution of constrained minimization problems of this type are well known in optimization theory. Recently, Segall [12] has considered a similar bit allocation problem [with $g_i(x) = g(x), \forall i$] for encoding of vector sources. The solution is given by

$$g_i'(n_i)\beta^2(i) + \lambda = 0 \quad , \qquad \lambda \leq -g_i'(0)\beta^2(i)$$

$$n_i = 0 \quad , \qquad \lambda < -g_i'(0)\beta^2(i) \quad . \qquad (15)$$

Denoting $h_i(x) = g_i'^{-1}(x)$, (16)

we can solve (15) for n_i as

$$n_i = \begin{cases} h_i\left(\dfrac{-\lambda}{\beta^2(i)}\right) , & \lambda < -g_i'(0)\beta^2(i) \\ 0 , & \lambda > -g_i'(0)\beta^2(i) \end{cases} , \qquad (17)$$

where λ is the root of the nonlinear equation

$$\sum_{i:\lambda \leq -g_i'(0)\beta^2(i)} h\left(-\frac{\lambda}{\beta^2(i)}\right) = NP \quad . \qquad (18)$$

The value of the minimum distortion is

$$D^* = \frac{1}{N}\left[\sum_{\lambda \le -g_i'(0)\beta^2(i)} g_i(n_i)\beta^2(i) + \sum_{\lambda > -g_i'(0)\beta^2(i)} \beta^2(i) \right] . \quad (9)$$

Now consider the following special cases.

1. $\hat{\beta}^2(i) \simeq \beta^2(i)$, i.e., the effect of quantization on the variance of the prediction error is ignored. This gives

$$g_i(n) \simeq f(n), \quad \forall i \ . \quad (20)$$

When the quantizer in the DPCM loop is the ideal optimal block encoder (Shannon quantizer) that achieves the rate distortion bound we have

$$f(n) = 2^{-2n} , \quad (21)$$

which gives

$$f'(x) = (-2\log_e 2)2^{-2x} ,$$

$$\text{or} \quad h(x) = f'^{-1}(x) = -\tfrac{1}{2}\log_2\left(\frac{x}{-2\log_e 2}\right) . \quad (22)$$

Defining

$$\theta = \lambda \Big/ (2\log_e 2) , \quad (23)$$

and using (21) in (17) to (19) we get

$$n_i = \begin{cases} \tfrac{1}{2}\log_2\left(\dfrac{\beta^2(i)}{\theta}\right) , & 0 < \theta \le \beta^2(i) \\ 0 , & \theta > \beta^2(i) , \end{cases} \quad (24a)$$

where θ is the solution of

$$\sum_{i=1}^{N} \max\left[0, \tfrac{1}{2}\log_2 \frac{\beta^2(i)}{\theta}\right] = NP , \quad (24b)$$

and the minimized distortion is

$$D^* = \frac{1}{N} \sum_{i=0}^{N} \min[\beta^2(i), \theta] \quad . \tag{24c}$$

Equations (24) are valid for both SC1 and SC2 models, where $\beta^2(i)$ is defined by (2b).

2. Consider the case when $f(\cdot)$ and $g_i(\cdot)$ are given by (21) and (13b) respectively. Then

$$g_i'(x) = \frac{f'(x)}{(1 - \rho_i^2 f(x))^2} \tag{25}$$

$$= \frac{(-2\log_e 2)2^{-2x}}{(1 - \rho_i^2 2^{-2x})^2} \quad .$$

From this equation one can obtain a unique inverse (note that $g_i'(x)$ must be nonpositive for all $x \geq 0$), as

$$g_i'^{-1}(x) \overset{\Delta}{=} h_i(x) = \begin{cases} \frac{1}{2}\log_2\left[\dfrac{-\log_e 2 + \rho_i^2 x + \sqrt{(\log_e 2)^2 - (2\log_e 2)\rho_i^2 x}}{\rho_i^4 x} \right], & \rho_i \neq 0 \\ \frac{1}{2}\log_2 \left(\dfrac{-x}{2\log_e 2} \right), & \rho_i = 0 \end{cases} \quad . \tag{26}$$

For SC1 model, defining

$$\theta = \frac{\lambda}{(2\log_e 2)} \quad , \tag{27}$$

and using (26) in (17), (18) and (19) we get

$$n_i = \begin{cases} \frac{1}{2}\log_2 \dfrac{\beta^2(i)}{\theta} w(i,\theta) & , \quad 0 < \theta \leq \dfrac{\beta^2(i)}{(1 - \rho_i)^2} \\ 0 & , \quad \theta > \dfrac{\beta^2(i)}{(1 - \rho_i^2)^2} \end{cases} , \tag{28a}$$

where

$$w(i,\theta) \triangleq \begin{cases} 1, & \rho_i = 0 \\ \dfrac{1}{2\rho_i^4}\left[1 + \dfrac{2\rho_i^2\theta}{\beta^2(i)} - \left(1 + \dfrac{4\rho_i^2\theta}{\beta^2(i)}\right)^{\frac{1}{2}}\right], & \rho_i \neq 0 , \end{cases} \tag{28b}$$

θ is the solution of

$$\sum_{i:\, 0<\theta\leq \frac{\beta^2(i)}{(1-\rho_i^2)^2}} \tfrac{1}{2}\log_2 \frac{\beta^2(i)w(i,\theta)}{\theta} = NP , \tag{28c}$$

and the minimum distortion is

$$D^* = \frac{1}{N}\sum_{i=1}^{N} \min[\beta^2(i),\ \theta/w(i,\theta)] . \tag{28d}$$

In case of SC2 model, substitution of $\rho_i = \rho_h$ = constant, in Eqns. (28) above give the desired bit allocation and distortion formulas. The bit allocations are obtained by first solving (28c) iteratively (by Newton's method, for example) to obtain θ, for a fixed rate P. Given θ, (28a) and (28b) yield the desired allocations $\{n_i\}$ and (28d) gives the corresponding distortion.

For practical quantizers similar formulas could be obtained if their distortion-rate functions $f(\cdot)$ and their inverse functions $h_i(\cdot)$ are known in analytic forms. Table I , gives approximate models of $f(\cdot)$ for various practical quantizers. If $g_i(\cdot)$ is assumed to be nearly equal to $f(\cdot)$, then $h_i(x) = f'^{-1}(x) \triangleq h(x)$. Table II , gives approximate analytic models for the $h(\cdot)$ functions for these quantizers. These models were determined by Wang [15] by a piecewise exponential fit of the distortion-rate curves $f(x)$ and its derivative $f'(x)$.

TABLE I: Analytic Models for Some Often Used Quantizers For Unity Variance Probability Density Functions

QUANTIZER	PROBABILITY DENSITY	MEAN SQUARE ERROR MODEL, f(n) n = number of bits 2^n = number of levels
Shannon (Lower Bound)	Arbitrary Unity Variance	$f(n) = 2^{-2n}$, $n \geq 0$
Lloyd-Max Optimum Mean Square	Gaussain $\mathcal{N}(0,1)$ $\frac{1}{\sqrt{2\pi}} \exp(-x^2/2)$	$f(n) = \begin{cases} 2^{-1.5047n}, & 0 \leq 2^n < 5 \\ 1.5253\ 2^{-1.8274n}, & 5 \leq 2^n \leq 36 \\ 2.2573\ 2^{-1.9626}, & 36 \leq 2^n \leq 512 \end{cases}$
Lloyd-Max Optimum Mean Square	Laplacian Unity Variance $\sqrt{2} \exp(-\sqrt{2}\lvert x \rvert)$	$f(n) = \begin{cases} 2^{-1.1711n}, & 0 \leq 2^n < 5 \\ 2.0851\ 2^{-1.7645n}, & 5 \leq 2^n < 36 \\ 3.6308\ 2^{-1.9572n}, & 36 \leq 2^n \leq 512 \end{cases}$
Optimum Uniform	Gaussain $\mathcal{N}(0,1)$	$f(n) = \begin{cases} 2^{-1.5012n}, & 0 \leq 2^n < 5 \\ 1.2477\ 2^{-1.6883n}, & 5 \leq 2^n < 36 \\ 1.5414\ 2^{-1.7562n}, & 36 \leq 2^n \leq 512 \end{cases}$
Optimum Uniform	Laplacian Unity Variance	$f(n) = \begin{cases} 2^{-1.1619n}, & 0 \leq 2^n < 5 \\ 1.4156\ 2^{-1.4518n}, & 5 \leq 2^n < 36 \\ 2.1969\ 2^{-1.5944n}, & 36 \leq 2^n \leq 512 \end{cases}$

322

TABLE II : Formulas for h(x) for different Quantizers

Max Quantizer, Gaussain Density

$$h(x) = \begin{cases} 0.04034 - 0.6646\log_2(-x), & -1.043 \le x < -0.102 \\ 0.52 - 0.5472\log_2(-x), & -0.102 \le x < -0.0027 \\ 0.8247 - 0.5192\log_2(-x), & -0.0027 \le x \le -0.000015 \end{cases}$$

Optimum Uniform Quantizer, Gaussain Density

$$h(x) = \begin{cases} 0.0382 - 0.6661\log_2(-x), & -1.041 \le x < 0.0965 \\ 0.3235 - 0.5923\log_2(-x), & -0.0965 \le x < 0.0034 \\ 0.517 - 0.5694\log_2(-x), & -0.0034 \le x \le 3.28 \times 10^{-5} \end{cases}$$

Max Quantizer, Laplace Density

$$h(x) = \begin{cases} -0.2569 - 0.8539\log_2(-x), & -0.811745 \le x < -0.1490 \\ 2.085 - 0.5667\log_2(-x), & -0.1490 \le x < -0.0044 \\ 1.175 - 0.5109\log_2(-x), & -0.0044 \le x \le -2.45 \times 10^{-5} \end{cases}$$

Optimum Uniform Quantizer, Laplace Density

$$h(x) = \begin{cases} -0.2688 \; -0.8607\log_2(-x), & -0.8054 \le x < 0.1377 \\ 0.3516 - -0.6888\log_2(-x), & -0.1377 \le x < -0.008 \\ 0.8026 - 0.6272\log_2(-x), & -0.008 \le x < -1.16 \times 10^{-4} \end{cases}$$

Shannon Quantizer

$$h(x) = \tfrac{1}{2}\log_2(-x \; 2\log_e^2) \qquad \text{all } x \le 0$$

323

3. Integer Bit Allocation

In the above, we assumed n_i to be a real number. In many practical situations n_i is required to be an integer. An approximate integer bit allocation can be obtained via the foregoing method by rounding n_i to the nearest integer. The actual bit rate then would be

$$P_a = \frac{1}{N} \sum_{i=1}^{N} [n_i] \quad ,$$

where $[x]$ indicates the nearst integer to x. To obtain the exact solution, the minimization of (13a) is subject to the constraint that $\{n_i\}$ be a set of nonnegative integers. The solution is obtained via an iterative algorithm due to Fox [13] as follows.

Let n denote a vector of elements $\{n_i, \quad 1 \le i \le N\}$.

1. Start with the allocation $\underline{n}^0 = \underline{0}$.

2. Set $\ell = 1$.

3. $\underline{n}^{\ell} \leftarrow \underline{n}^{\ell} + e_i$, where e_i is the ith unit vector and is any index for which

$$\Delta_{i,\ell} = \beta^2(i) g_i(n_i^{\ell-1}) - \beta^2(i) g_i(n_i^{\ell-1} + 1)$$

is maximum. The quantity $\Delta_{i,\ell}$ is called the marginal return of increasing the ith allocation in ℓth step by one bit.

4. If $\sum_{i=1}^{N} n_i^{\ell} \ge PN$, terminate; otherwise $\ell \rightarrow \ell + 1$ and go to step 3.

This algorithm simply means, the marginal returns $\Delta_{i,\ell}$ are calculated for $1 \le i \le N$ and $\ell = 1,2, \ldots$, and are arranged in a decreasing order until all the bits are exhuasted. Tables III to V show the bit allocations for the SC1, SC2 and separable covariance models obtained via real bit allocation and integer bit allocation algorithms described above.

TABLE III: BIT ALLOCATION PATTERNS BASED ON EQNS. (28), SC2 MODEL

Desired Average Bit Rate	2			1.5			1			0.5		
Type of Bit Allocation / DPCM Channel No.	n	[n]	n*	n	[n]	n*	n	[n]	n*	n	[n]	n*
1	5.97	6	6	5.30	5	5	4.45	4	4	3.19	3	3
2	3.99	4	4	3.33	3	3	2.51	3	2	1.39	1	1
3	3.05	3	3	2.40	2	2	1.65	2	1	0.77	1	1
4	2.50	3	3	1.89	2	2	1.21	1	1	0.51	1	1
5	2.13	2	2	1.55	2	1	0.95	1	1	0.38	0	1
6	1.87	2	2	1.32	1	1	0.78	1	1	0.29	0	1
7	1.67	2	2	1.15	1	1	0.66	1	1	0.24	0	0
8	1.51	2	2	1.02	1	1	0.57	1	1	0.20	0	0
9	1.39	1	1	0.92	1	1	0.51	1	1	0.17	0	0
10	1.29	1	1	0.85	1	1	0.45	0	1	0.15	0	0
11	1.21	1	1	0.79	1	1	0.43	0	1	0.14	0	0
12	1.15	1	1	0.75	1	1	0.40	0	1	0.12	0	0
13	1.11	1	1	0.71	1	1	0.38	0	0	0.12	0	0
14	1.07	1	1	0.69	1	1	0.36	0	0	0.11	0	0
15	1.05	1	1	0.67	1	1	0.35	0	0	0.10	0	0
16	1.04	1	1	0.66	1	1	0.35	0	0	0.10	0	0
Average Bit Rate	2.00	2	2	1.50	1.56	1.5	1.00	0.94	1	0.50	0.38	0.5

where n: real bit allocation

[n]: indicates the nearest integer of n

n*: integer bit calculated via integer bit allocation algorithm

322

325

TABLE IV: BIT ALLOCATION PATTERNS BASED ON EQNS. (28), SCI MODEL

Desired Average Bit Rate	2			1.5			1			0.5		
Type of Bit Allocation / DPCM Channel No.	n	[n]	n*	n	[n]	n*	n	[n]	n*	n	[n]	n*
1	4.27	4	4	3.77	4	4	3.15	3	3	2.16	2	2
2	4.12	4	4	3.62	4	4	3.00	3	3	2.01	2	2
3	3.74	4	4	3.24	3	3	2.62	3	3	1.63	2	2
4	3.28	3	3	2.78	3	3	2.16	2	2	1.17	1	1
5	2.83	3	3	2.33	2	2	1.70	2	2	0.72	1	1
6	2.42	2	2	1.92	2	2	1.30	1	1	0.31	0	0
7	2.06	2	2	1.56	2	2	0.94	1	1	0	0	0
8	1.75	2	2	1.25	1	1	0.63	1	1	0	0	0
9	1.49	1	1	0.99	1	1	0.37	0	0	0	0	0
10	1.27	1	1	0.77	1	1	0.14	0	0	0	0	0
11	1.08	1	1	0.58	1	1	0	0	0	0	0	0
12	0.93	1	1	0.43	0	0	0	0	0	0	0	0
13	0.81	1	1	0.31	0	0	0	0	0	0	0	0
14	0.71	1	1	0.21	0	0	0	0	0	0	0	0
15	0.65	1	1	0.15	0	0	0	0	0	0	0	0
16	0.61	1	1	0.11	0	0	0	0	0	0	0	0
Average Bit Rate	2.00	2	2	0.15	1.5	1.5	1.0	1.0	1.0	0.50	0.5	0.5

where n: real bit allocation

[n]: indicates the nearest integer of n

n*: integer bit calculated via integer bit allocation algorithm

323

326

TABLE V: BIT ALLOCATION PATTERNS BASED ON EQNS. (24), SEPARABLE COVARIANCE MODEL

Desired Average Bit Rate	2			1.5			1			0.5		
Type of Bit Allocation / DPCM Channel No.	n	[n]	n*	n	[n]	n*	n	[n]	n*	n	[n]	n*
1	5.39	5	5	4.89	5	5	4.38	4	4	3.44	3	4
2	4.06	4	4	3.56	4	4	3.04	3	3	2.10	2	2
3	3.26	3	3	2.76	3	3	2.25	2	2	1.31	1	1
4	2.70	3	3	2.20	2	2	1.68	2	2	0.74	1	1
5	2.31	2	2	1.81	2	2	1.30	1	1	0.36	0	0
6	2.01	2	2	1.51	2	1	1.00	1	1	0.06	0	0
7	1.78	2	2	1.28	1	1	0.76	1	1	0	0	0
8	1.59	2	2	1.09	1	1	0.57	1	1	0	0	0
9	1.43	1	2	0.93	1	1	0.42	0	1	0	0	0
10	1.31	1	1	0.81	1	1	0.29	0	0	0	0	0
11	1.20	1	1	0.70	1	1	0.18	0	0	0	0	0
12	1.11	1	1	0.62	1	1	0.10	0	0	0	0	0
13	1.05	1	1	0.55	1	1	0.03	0	0	0	0	0
14	1.00	1	1	0.50	0	0	0	0	0	0	0	0
15	0.96	1	1	0.46	0	0	0	0	0	0	0	0
16	0.94	1	1	0.44	0	0	0	0	0	0	0	0
Average Bit Rate	2.10	1.94	2	1.51	1.57	1.5	1.00	0.94	1	0.50	0.44	0.5

where n: real bit allocation

[n]: indicates the nearest integer of n

n*: integer bit calculated via integer bit allocation algorithm

324

327

CHAPTER FOUR

ADAPTIVE HYBRID CODING

The coding scheme of the previous section can be adapted to image fields whose spatial statistics vary slowly by updating the various coefficients of the image model with variations of the statistics. A complete update of the image model parameters could often increase the complexity of the coder to make it impractical. In the sequel we consider two adaptive schemes which offer reasonable improvement in coder performance with only a marginal increase in its complexity.

1. Adaptive Variance Estimation (Adaptive Hybrid Coding 1)

For a fixed predictor in the feedback loop of a DPCM channel, the variance of the prediction error will fluctuate with changes in image statistics. In this scheme we update the variance of the prediction error at each step j. This updated variance is used to adjust the spacing of the qunatizer levels in each DPCM channel. For mean square error criterion, the decision and reconstruction levels of a quantizer are directly proportional to the standard deviation of the input random variables. Hence, the above mentioned adaptation can be achieved by simply normalizing the prediction error by its updated standard deviation. The quantizer levels are then fixed and correspond to unit variance random variables. Figure 9 , shows this scheme.

Let

$$\hat{\sigma}_j^2(i) \triangleq \text{variance of } \hat{e}_j(i), \text{ the prediction error at step j of the ith DPCM loop} = E[\hat{e}_j(i)]^2 , \qquad (1a)$$

328

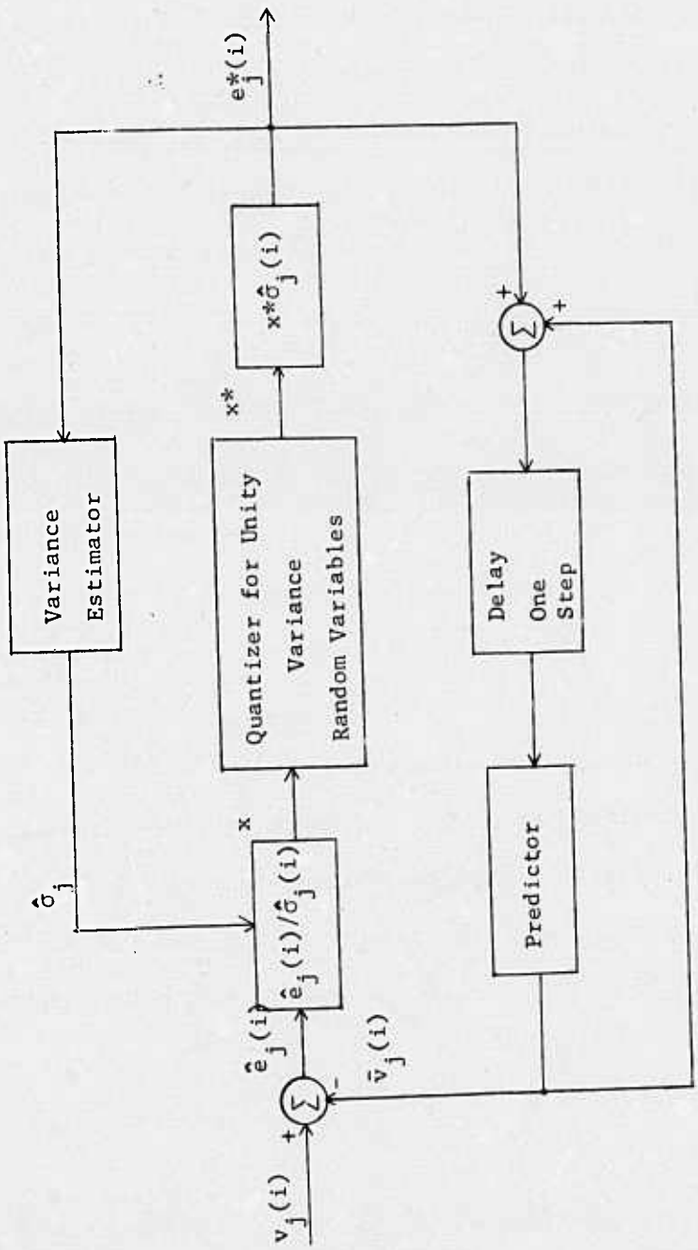


FIGURE 9: ADAPTIVE VARIANCE ESTIMATION

329

$$\sigma^{*2}_j(i) \triangleq \text{variance of the quantized values} = E[e^*_j(i)]^2 \,. \qquad (1b)$$

Since the quantized variables $e^*_j(i)$ are available both at the receiver and the transmitter, it is easy to estimate $\sigma^{*2}_j(i)$. Castellino, et. al [16] have suggested a formula for an estimate of $\sigma^{*2}_j(i)$ (for single channel DPCM system) as

$$\sigma^{*2}_j(i) = \frac{1-\gamma}{\gamma} \sum_{m=1}^{j} \gamma_m e^{*2}_{j-m+1}(i) + \gamma^j \sigma^{*2}_1(i), \qquad 0 < \gamma < 1 \,. \qquad (2)$$

This has been called an exponential average variance estimator. A more convenient form of (2) is the recursion

$$\sigma^{*2}_{j+1}(i) = (1-\gamma) e^{*2}_j(i) + \gamma \sigma^{*2}_j(i), \qquad j = 1, 2, \ldots \qquad (3)$$

For small quantization errors one may take σ^*_j to be equal to $\hat{\sigma}_j$. For Lloyd-Max quantizers [17] (or approximations thereof) a more accurate estimate of $\hat{\sigma}_j$ is possible. Since, the variance of a Lloyd-Max quantizer input equals the sum of the variances of the quantizer output and the quantization error we have

$$\hat{\sigma}^2_j(i) = \sigma^{*2}_j(i) + f(n_i)\hat{\sigma}^2_j(i) \,,$$

$$\text{or} \quad \hat{\sigma}^2_j(i) = \frac{\sigma^{*2}_j(i)}{1-f(n_i)} \,, \qquad n_i \geq 1 \,, \qquad (4)$$

where n_i is the number of bits allocated to the ith channel. Using (4) in (3) we get

$$\hat{\sigma}^2_{j+1}(i) = \left(\frac{1-\gamma}{1-f(n_i)}\right) e^{*2}_j(i) + \gamma \hat{\sigma}^2_j(i), \qquad n_i > 1, \quad 0 < \gamma < 1 \,. \qquad (5)$$

The above estimates become poor for DPCM channels which are assigned small number of bits ($n_i \simeq 1$). For these channels $\hat{\sigma}_j(i)$ could be estimated by some sort of extrapolation of the already estimated $\hat{\sigma}_j(i)$. For example, for the SC1 model, we know

$$\hat{\sigma}_j^2(i) \simeq E[\hat{e}_j^2(i)] = \frac{\beta^2}{[1-\rho_i^2 f(n_i)]\lambda_{ci}^2} \quad , \quad i \in \mathcal{L} \ , \tag{6}$$

where $\mathcal{L}$ is the set of , say m, channels for which $n_i > 1$. From (6), an estimate of β^2 at step j, denoted by $\tilde{\beta}_j^2$ is given by

$$\tilde{\beta}_j^2 = \frac{1}{m} \sum_{i \in \mathcal{L}} \hat{\sigma}_j^2(i)\lambda_{ci}^2[1-\rho_i^2 f(n_i)] \ . \tag{7}$$

This gives the extrapolated estimates as

$$\hat{\sigma}_j^2(i) = \frac{\tilde{\beta}_j^2}{\lambda_{ci}^2[1-\rho_i^2 f(n_i)]} \quad , \quad i \notin \mathcal{L} \ . \tag{8}$$

For SC2 model, the equations corresponding to (7) and (8) become

$$\tilde{\beta}_j^2 = \frac{1}{m} \sum_{i \in \mathcal{L}} \hat{\sigma}_j^2(i)\lambda_{ci}[1-\rho_h^2 f(n_i)] \ , \tag{9}$$

$$\hat{\sigma}_j^2(i) = \frac{\tilde{\beta}_j^2}{\lambda_{ci}[1-\rho_h^2 f(n_i)]} \quad , \quad i \notin \mathcal{L} \ . \tag{10}$$

The adaptive variance estimator only needs to solve Eqns. (5), (7) and (8) for SC1 model and (5), (9) and (10) for SC2 model. We finally note that this adaptive scheme maintains the same constant data rate in each DPCM channel as in the nonadaptive scheme of the previous section.

2. Adaptive Classification

In this method each image column is classified as belonging to one of K pre-determined classes. Each of these classes is determined according to the activity in that image column. The activity in an image column is measured by the variance of that column. For a given class of images the probability distribution function of the variances of all the columns can be pre-determined. (Note that this probability distribution function could depend on the size of the image column used in hybrid coding).

In the DPCM loop, the transformed signal in each DPCM channel is processed as before except that the differential signal is quantized according to the classification of that image column. Thus, the quantizer in the DPCM loop is adapted to the image activity in each column. The number of quantization levels (or bits) depend on the variance of a class so that columns of high dynamic activity are assigned more bits than those of low activity. In this way quantizing bits are employed more efficiently. However, the classification information for each column needs to be transmitted. This can be done by sending an extra $\log_2 K$ bits per column or $\frac{1}{N} \log_2 K$ bits/pixel. Experimentally it was found that the variance of successive columns, denoted by $\{\xi_j\}$ (say), are highly correlated. Instead of transmitting the classification map, a DPCM loop for transmission of $\{\xi_j\}$ turns out to be more robust. The reconstructed values $\{\xi_j^*\}$ are used for classification at the receiver as well as the transmitter. A two bit quantizer was found to give very accurate reproduction of $\{\xi_j\}$.

Let $p(\xi)$ = probability density of the variances ξ of a column.

Suppose the K classes have been predetermined and let (t_k, t_{k+1}), $1 \le k \le K$ denote the decision boundaries for the kth class. Since $\xi \ge 0$, $t_0 = 0$, $t_{K+1} = \infty$. Then

$$p_k = \int_{t_k}^{t_{k+1}} p(\xi)d\xi = \text{probability of the kth class} , \tag{11}$$

$$\bar{\xi}_k = \frac{1}{p_k} \int_{t_k}^{t_{k+1}} \xi p(\xi)d\xi = \text{variance of the kth class} . \tag{12}$$

First assume that the thresholds $\{t_k\}$ have been somehow predetermined and p_k and $\bar{\xi}_k$ are known. Also, we will assume that for each class k, the image model parameters have been pre-determined and are known. To this end, let us denote by $\rho_{i,k}$, $\beta^2(i,k)$, $n_{i,k}$ etc., the quantities for the ith DPCM channel and kth class corresponding to the definitions of ρ_i, $\beta^2(i)$, n_i, etc., respectively. Then, following the development of the foregoing chapters, we can write the expected average distortion for any image column (in steady state DPCM) as

$$D = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} \beta^2(i,k) g_{i,k}(n_{i,k}) p_k , \tag{13}$$

where $n_{i,k} \ge 0$ are such that the expected average bit rate must be constant, i.e.,

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} n_{i,k} p_k = P . \tag{14}$$

The bit allocation problem associated with (13) and (14) can be solved along the same lines as in Chapter Three. As an example, consider the case when

$$g_{i,k}(x) \simeq f(x) = 2^{-2x} ,$$

then one obtains

$$n_{i,k} = \max\left[0, \tfrac{1}{2}\log_2 \frac{\beta^2(i,k)}{\theta} \right] , \tag{15a}$$

where θ is the solution of

$$\sum_{k=1}^{K} \sum_{i=1}^{N} \max\left[0, \tfrac{1}{2}\log_2 \frac{\beta^2(i,k)}{\theta} \right] p_k = NP , \tag{15b}$$

and the minimum achievable distortion is

$$D = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} \min[\theta, \beta^2(i,k)] p_k . \tag{15c}$$

Figure 10 shows a typical bit allocation patterns for four class problem with equal probabilities ($p_k = 0.25$) for each class. This method of adaptive hybrid coding requires, (i) measurement of variance of each column and its transmission by an extra DPCM channel, (ii) classification of each column to one of K classes based on thresholds $\{t_k\}$, and (iii) storing of K bit allocation tables, one for each class.

For equal probabilities p_k, the thresholds $\{t_k\}$ are easily determined. These are simply the points where the probability distribution function takes the values $\frac{k}{K}$, $0 \le k \le K$. On the other hand, the optimum threshold $\{t_k^*\}$ would be such that the distortion D would be minimum. Hence, the optimum bit allocation and optimum threhold selection problem is to minimize with respect to $\{t_k, n_{i,k}\}$, the distortion

$$D = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} \beta_0^2(i,k) g_{i,k}(n_{i,k}) \bar{\xi}_k p_k , \tag{16a}$$

subject to

334

$k \rightarrow$	I	II	III	IV
	5	5	5	5
$n_{i,k}$	4	4	3	1
	4	3	2	1
↓	4	3	2	1
	4	3	2	1
	3	3	2	1
	3	2	2	1
	3	2	1	1
	3	2	1	1
	3	2	1	1
	2	2	1	1
	2	2	1	0
	2	1	1	1
	2	1	1	0
	2	1	1	0
	2	1	1	0

AVERAGE BIT RATE= .20156E+01

$k \rightarrow$	I	II	III	IV
	4	5	5	4
$n_{i,k}$	4	3	2	1
	4	3	2	0
↓	3	3	2	0
	3	2	1	0
	3	2	1	0
	3	2	1	0
	2	2	1	0
	2	2	1	0
	2	1	1	0
	2	1	1	0
	2	1	1	0
	1	1	1	0
	1	1	1	0
	1	1	1	0
	1	1	1	0

AVERAGE BIT RATE= .15156E+01

$k \rightarrow$	I	II	III	IV
	4	4	4	3
$n_{i,k}$	3	3	2	0
	3	2	1	0
↓	3	2	1	0
	2	2	1	0
	2	2	1	0
	2	1	0	0
	2	1	0	0
	1	1	0	0
	1	1	0	0
	1	1	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0

AVERAGE BIT RATE= .96875E+00

$k \rightarrow$	I	II	III	IV
	3	3	3	3
$n_{i,k}$	2	2	1	0
	2	1	0	0
↓	2	1	0	0
	2	1	0	0
	1	1	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0

AVERAGE BIT RATE= .50000E+00

FIGURE 10: BIT ALLOCATIONS FOR ADAPTIVE CLASSIFICATION
(K = 4, Actual Measurements Model)

335

$$\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{N} n_{i,k}P_k = P \quad , \tag{16b}$$

where $\bar{\xi}_k = \bar{\xi}(t_k, t_{k+1})$, $P_k = P_k(t_k, t_{k+1})$, satisfy (11) and (12), and

$$\beta_0^2(i,k) = \beta^2(i,k)/\bar{\xi}_k \quad , \tag{17}$$

i.e, $\beta_0^2(i,k)$ is the prediction error variance of a unit variance Markov process representing the ith channel for kth class. If we assume that $P(\xi)$ is defined over a finite interval $t_1 \le \xi \le t_{K+1} < \infty$ and we can approximate (11) as

$$P_k \simeq (t_{k+1} - t_k)\, p\left(\frac{t_{k+1} + t_k}{2}\right) \quad , \tag{18}$$

then the above optimization problem can be shown to be solved by the nonlinear equations

$$n_{i,k} = \begin{cases} h_{i,k}\left(\dfrac{-\lambda}{\beta_0^2(i,k)\bar{\xi}_k}\right) & , \quad \lambda \le -g'_{i,k}(0)\beta_0^2(i,k)\bar{\xi}_k \\ 0 & , \quad \lambda > -g'_{i,k}(0)\beta_0^2(i,k)\bar{\xi}_k \quad , \end{cases} \tag{19a}$$

where λ is the solution of (16b) and (19a). The thresholds $\{t_k\}$ must satisfy

$$t_k = \frac{\sum_{i=1}^{N}(n_{i,k} - n_{i,k-1})}{\sum_{i=1}^{N}\left[\beta_0^2(i,k-1)\bar{\xi}_{k-1}g_{i,k-1}(n_{i,k-1}) - \beta_0^2(i,k)\bar{\xi}_k g_{i,k}(n_{i,k})\right]} \quad , \; 1 \le k \le K \, , \tag{19b}$$

where $t_1 = 0$, t_{k+1} = given. If $\{t_k\}$ are known, then

$$\bar{\xi}_k = \tfrac{1}{2}(t_{k+1} + t_k) \, . \tag{19c}$$

A simultaneous solution of (16b) and (19a) to (19c) will yield the desired result.

336

CHAPTER FIVE

EXPERIMENTAL RESULTS AND CONCLUSIONS

Several computer experiments have been performed to simulate the hybrid coding schemes discussed in the previous chapters. Table VI shows the parameters of the various image models used here. In all the experiments, the Cosine transform based semicausal models were used. A 256 x 256 girl image was divided into 16 strips, each of size 16 x 256. Each strip was hybrid coded independently. This way only a 16 step Cosine transform is needed which makes the scheme feasible for a real time implementation [18]. In all the performance measurements, the signal to noise ratio (SNR) is defined as

$$\text{SNR} = 20\log_{10}\left(\frac{\text{Peak to Peak value of the Signal}}{\text{r.m.s. error}}\right). \qquad (1)$$

For an eight bit original data this becomes

$$\text{SNR} = 20\log_{10}\frac{255}{\text{r.m.s. error}}. \qquad (2)$$

The root mean square error is measured over the entire 256 x 256 image and is defined as

$$e_{rms} = \left[\frac{1}{(256)^2}\sum_{i=1}^{256}\sum_{j=1}^{256}(u_{i,j} - u^*_{i,j})^2\right]^{\frac{1}{2}}. \qquad (3)$$

In each DPCM channel a compandor is used as an approximation to the Max quantizer [15,19].

1. Nonadaptive Hybrid Coding

Figure 11 shows the SNR vs. bit rate for the SC1, SC2 and the separable covariance image models for the 256 x 256 girl image. As expected, the performance of SC2 and separable covariance models are very close. The SC1

TABLE VI: PARAMETERS OF IMAGE MODELS

Separable Covariance Model

$$\rho_h = \rho_v = 0.95$$

$$\beta^2(i) = (1-\rho_h^2)/\lambda_i$$

$$\lambda_i = [\Psi R \Psi]_{i,i}$$

Semicausal Model SC1

$$\alpha = .4275, \quad \gamma = 0.1415, \quad \beta^2 = 0.0198$$

$$\lambda_{ci} = 1 - 2\alpha \cos \frac{(i-1)\pi}{N}, \quad 1 \le i \le N$$

$$\beta^2(i) = \beta^2/\lambda_{ci}^2$$

$$\rho_i = \gamma/\lambda_{ci}$$

Semicausal Model SC2

$$\rho_h = \rho_v = 0.95, \quad \alpha = \rho_v/(1+\rho_v^2)$$

$$\lambda_{ci} = 1 - 2\alpha \cos \frac{(i-1)\pi}{N}, \quad 1 \le i \le N$$

$$\beta^2 = (1-\rho_h^2)(1-\rho_v^2)/(1+\rho_v^2)$$

$$\beta^2(i) = \beta^2/\lambda_{ci}$$

$$\rho_i = \rho_h$$

Actual Measurement Model

$$v_j(i) = \rho_i v_{j-1}(i) + e_j(i)$$

$$\rho_i = E v_j(i) v_{j-1}(i) \simeq \frac{1}{16M} \sum_{j=2}^{M} \sum_{k=i}^{16} v_j^k(i) v_{j-1}^k(i)$$

$$\beta^2(i) = E e_j^2(i) = (1-\rho_i^2) E[v_j(i)]^2 \simeq \frac{(1-\rho_i^2)}{16M} \sum_{j=1}^{M} \sum_{k=1}^{16} (v_j^k(i))^2$$

where $v_j^k(i)$ = ith element of the jth column of the kth 16 x M size image strip, (k = 1, . . . 16 here)

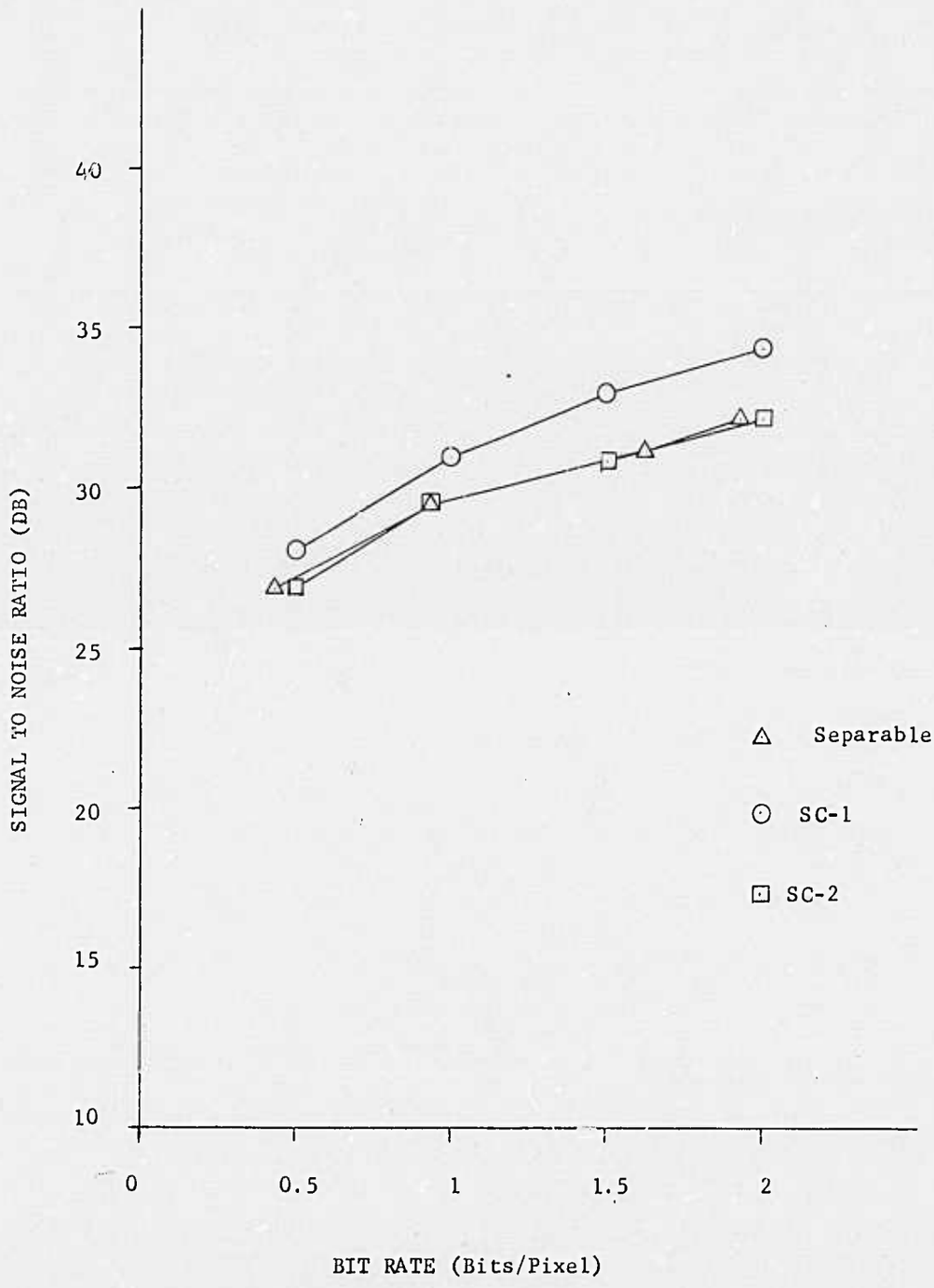


FIGURE 11: Nonadaptive Hybrid Coding

339

model performs better than the other two by about 1.5 to 2 Db, in general, above rates of 1 bit/pixel.

2. Adaptive Variance Estimation

Figure 12, shows the SNR vs. rate curves for the various models for this case. In these experiments, the variance estimates were updated only for channels for which $n_i \geq 2$ bits/pixel.

The performance of SC2 and separable covariance model improves by about 1.5 Db at 1 bit rate and about 4 DB at 2 bit rate. The reason for larger improvement at 2 bit rate is that the variance estimates (based on quantized data) are more accurate since many more channels have $n_i \geq 2$ bits/pixel. The performance improvement for SC1 model is less (only 0.5 DB at 1 bit rate and 1.5 Db at 2 bit rate). This could be because the SC1 model performance was already better (than SC2) to start with and the margin of improvement by updating prediction error variance is small. As the size of the image column (N) is increased, the performance of the adaptive variance estimator was found to improve. This is, again, because the number of channels (for a fixed average rate) with bit allocation $n_i \geq 2$ would increase giving a better estimate of the variances (as well as of the extrapolated values).

3. Adaptive Classification

Two different experiments utilizing this method, were performed. In the first experiment , a probability distribution function of the column varainces was determined from the histogram of their sample variances. Classification thresholds $\{t_k\}$ were determined for a K = 4 case assuming equal probability for each class. Then each column of the image was classified. The acutal measurements model for the Cosine transformed vectors was assumed, as

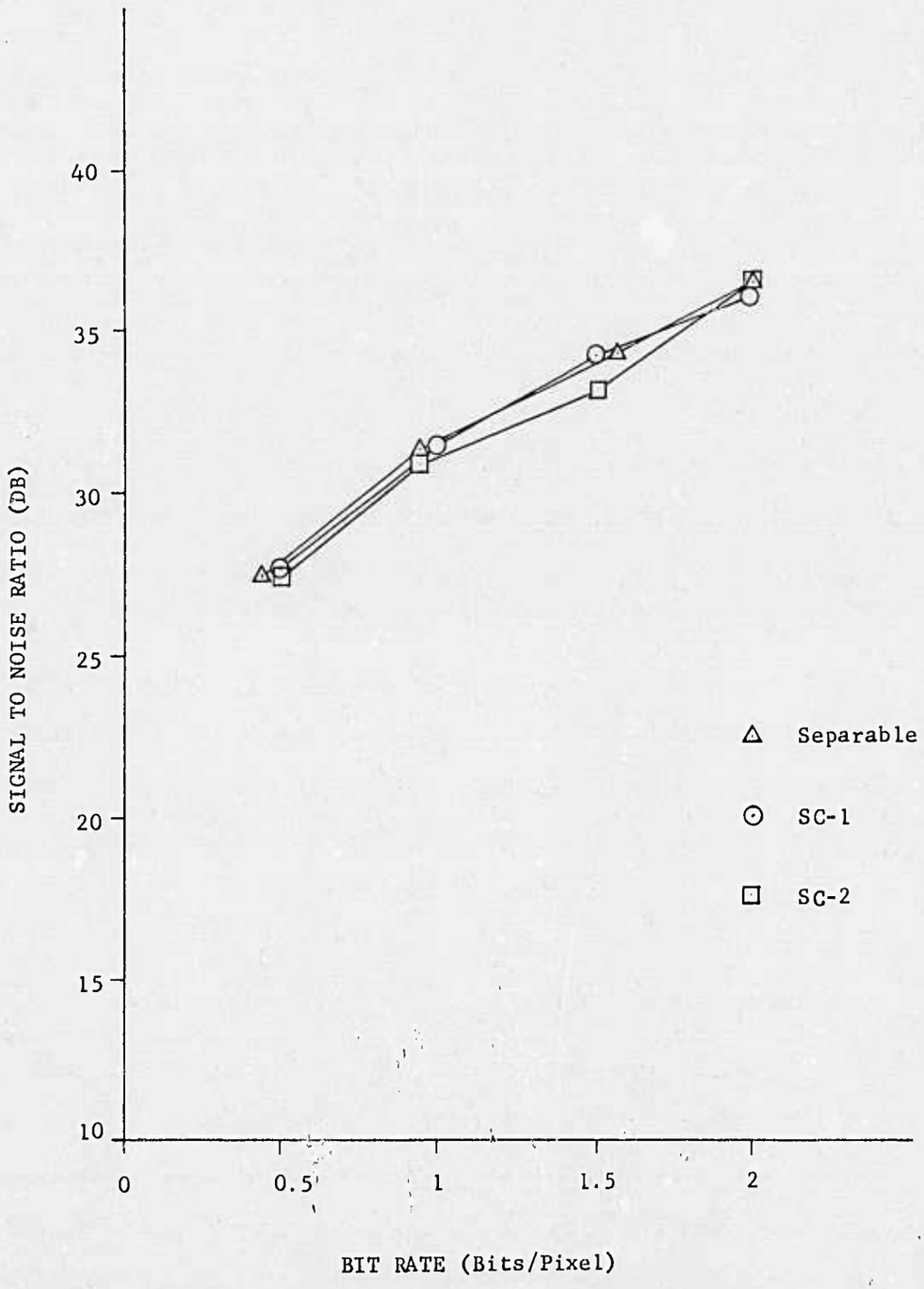


FIGURE 12: Adaptive Variance Estimator

341

$$v_j(i,k) = \rho_{i,k} v_{j-1}(i,k) + e_j(i,k) , \quad 1 \le i \le N , \quad k = 1, ..4,$$

with $Ee_j^2(i,k) \overset{\Delta}{=} \beta^2(i,k)$

$$\overset{\Delta}{=} (1-\rho_{i,k}^2) E[v_j(i,k)]^2 .$$

The parameters $\rho_{i,k}$ and $\beta^2(i,k)$ were measured for each class and bit allocation was done according to the method described in the previous chapter [see Fig. 10]. Figure 4 shows the adaptive classification hybrid coding scheme. Figure 13 shows the 16 x 32 classification map of 256 x 32 portion of the image with 16 element columns. (Recall that the given 256 x 256 image was divided into sixteen 16 x 256, rectangular subimages).

This method gave a large improvement (see Fig. 14) in SNR over the previous two schemes (about 5.5 Db at 1 bit rate and 8 Db at 2 bit rate over non-adaptive). However, the above procedure is somewhat impractical for online transmission (could be acceptable for storage) because it requires identification of the 2NK = 8N parameters, $\rho_{i,k}$ and $\beta^2(i,k)$, for each image. These parameters are than used to determine the bit allocations for various classifications. Assuming eight bits are required to transmit each of these parameters (this is actually a pessimistic estimate), the additional overhead for their transmission is only $\frac{8 \times 16 \times 8}{256 \times 256} \simeq \frac{1}{32}$ bit/pixel. The overhead for transmission of the column variances is only 2 bits per column or $\frac{1}{8}$ bit/pixel. The use of a 2 bit DPCM scheme allows for many more classes than four (because experimentally 2 bit DPCM gives very accurate reproduction of the variances) and for a four class case, this scheme is very robust compared to transmission of the classification map using 2 bits/column. This overhead could be reduced by using a 1 bit DPCM channel for the variances (for K = 4, it would be sufficient). Thus, the total overhead is always less than 0.13 bit/pixel. However, the complexity of the scheme at the

CLASSIFICATION MAP OF THE LEFT-QUARTER PORTION 16 BY 32 TRANSFORMED SUB-COLUMNS

```
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.4.4.4.4.4.3.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.4.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.4.4.4.4.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.4.4.4.4.4.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.
4.3.4.4.3.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.4.4.4.4.4.4.4.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.
4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.
4.4.4.4.4.4.4.4.3.4.4.4.4.3.3.4.3.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.
1.2.2.2.2.3.3.3.3.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.4.4.4.4.4.4.4.4.
1.2.2.2.2.2.2.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.2.2.3.
2.1.1.1.1.1.1.1.1.1.2.3.3.3.3.3.2.2.2.3.3.2.2.2.1.1.1.1.1.1.1.2.2.
1.1.1.1.1.1.1.1.1.1.1.2.2.3.3.4.4.4.4.4.3.3.3.2.2.2.2.2.2.2.2.2.2.
4.3.3.3.3.3.3.3.4.4.4.4.4.4.4.4.4.4.4.4.4.3.3.3.3.3.3.3.3.3.4.4.
2.2.2.2.3.3.3.3.3.3.3.3.3.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.3.4.3.3.3.
```

1 SPECIFIES THE HIGHEST ACTIVITY SUB-COLUMNS
4 SPECIFIES THE LOWEST ACTIVITY SUB-COLUMNS

FIGURE 13: CLASSIFICATION OF COLUMNS OF THE 256 x 256 PORTION OF THE IMAGE

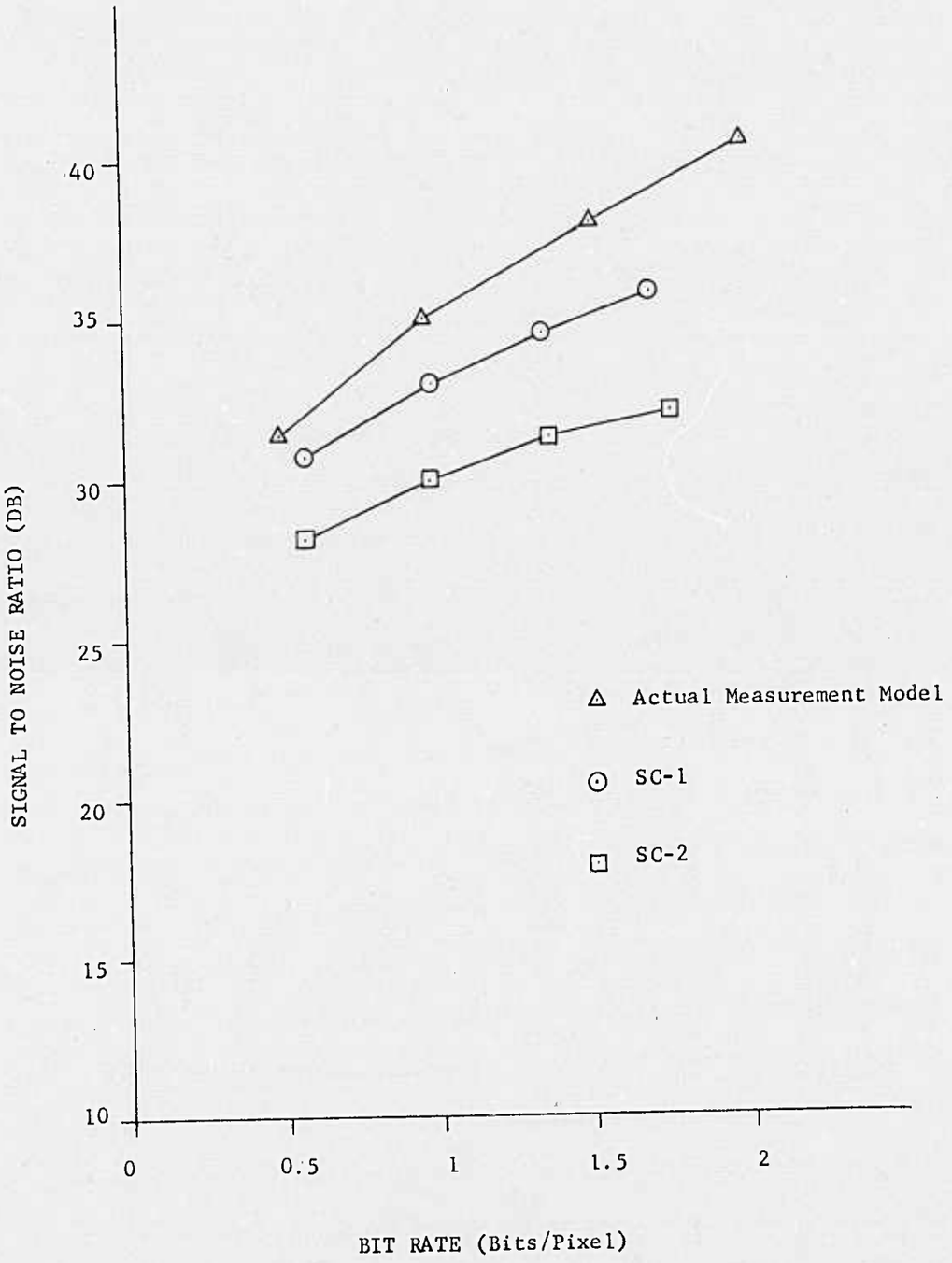


FIGURE 14: Adaptive Classification Hybrid Coding

344

transmitter is increased substantially.

The complexity of the foregoing method can be reduced considerably by using models for $\rho_{i,k}$ and $\beta^2(i,k)$ according to SC1 or SC2 representations. In our experiments here we assumed

$$\rho_{i,k} = \rho_i, \quad \forall\, k \quad , \qquad \rho_i = \begin{cases} \gamma/\lambda_{ci}\,, & \text{for SC1} \\ \rho_h\,, & \text{for SC2}\,, \end{cases}$$

$$\beta^2(i,k) = \beta_0^2(i)\sigma_k^2, \quad \forall\, k, \qquad \beta_0^2(i) = \begin{cases} \beta_0^2/\lambda_{ci}^2\,, & \text{for SC1} \\ \beta_0^2/\lambda_{ci}\,, & \text{for SC2}\,, \end{cases}$$

where σ_k^2 is the variance of the columns belonging to the kth class, $\beta_0^2(i)$ represents the prediction error variance of the ith, unit variance Markov process. Compared to the nonadaptive case, the only additional computation this model requires is the measurement of the sample variance of an image column and classifying it. Figures 15 and 16 show the bit allocations for the SC1 and SC2 models used for adaptive classification hybrid coding. Since the first DPCM channel in the Cosine transform represents the d.c. value (or sample mean of the columns), the number of bits allocated to this channel (for a given bit rate and class) were kept the same for all different models. The bit allocation algorithm of previous chapters was thus used only for $i = 2, \ldots N$ in the case of SC1 and SC2 models. Figure 14 shows the performance of these models when used for adaptive classification. Generally, the performance of the SC1 model is superior to that of the SC2. It is better than nonadaptive (separable covariance model) hybrid coding by about 3 Db, but is worse than the previous adaptive classification experiment by 2 to 3 Db. This, of course, is the price the designer has to pay for a simplified system. Finally, note that this scheme will produce variable bit rate form one image column to the next because of a possible change in classification.

k→	I	II	III	IV
$n_{i,k}$ ↓	5	5	5	5
	5	4	3	2
	5	4	2	1
	5	3	2	1
	4	3	2	0
	4	2	1	0
	3	2	1	0
	3	2	0	0
	3	1	0	0
	3	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	0	0	0

AVERAGE BIT RATE= .1703125000E+01

k→	I	II	III	IV
$n_{i,k}$ ↓	4	5	5	4
	5	4	2	1
	5	3	2	1
	4	3	2	0
	4	2	1	0
	3	2	1	0
	3	1	0	0
	3	1	0	0
	2	1	0	0
	2	1	0	0
	2	0	0	0
	2	0	0	0
	2	0	0	0
	2	0	0	0
	1	0	0	0
	1	0	0	0

AVERAGE BIT RATE= .1359375000E+01

k→	I	II	III	IV
$n_{i,k}$ ↓	4	4	4	3
	4	3	2	0
	4	3	1	0
	4	2	1	0
	3	2	1	0
	3	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0

AVERAGE BIT RATE= .1000000000E+01

k→	I	II	III	IV
$n_{i,k}$ ↓	3	3	3	3
	3	2	1	0
	3	2	1	0
	3	1	0	0
	2	1	0	0
	2	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0

AVERAGE BIT RATE= .5781250000E+00

FIGURE 15: Bit Allocation for Adaptive Classification, K = 4, SC1 Model

346

k	I	II	III	IV
$n_{i,k}$ ↓	5	5	5	5
	6	4	3	2
	5	3	2	1
	4	3	2	0
	4	2	1	0
	3	2	1	0
	3	2	1	0
	3	2	0	0
	3	1	0	0
	3	1	0	0
	3	1	0	0
	3	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0

AVERAGE BIT RATE= .1671875000E+01

k→	I	II	III	IV
$n_{i,k}$ ↓	4	5	5	4
	5	4	3	1
	4	3	2	0
	4	2	1	0
	3	2	1	0
	3	2	0	0
	3	1	0	0
	3	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	0	0	0
	2	0	0	0

AVERAGE BIT RATE= .1375000000E+01

k→	I	II	III	IV
	4	4	4	3
$n_{i,k}$ ↓	5	3	2	1
	4	2	1	0
	3	2	1	0
	3	1	0	0
	2	1	0	0
	2	1	0	0
	2	1	0	0
	2	0	0	0
	2	0	0	0
	2	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0

AVERAGE BIT RATE= .9843750000E+00

k→	I	II	III	IV
	3	3	3	3
$n_{i,k}$ ↓	4	2	1	0
	3	1	0	0
	3	1	0	0
	2	0	0	0
	2	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	1	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0

AVERAGE BIT RATE= .5781250000E+00

FIGURE 16: Bit Allocations for Adaptive Classification, K = 4, SC2 Model

347

Figure 17 compares the conventional nonadaptive scheme with the various adaptive hybrid coding methods using the SC1 model. Tables VII to XII give quantitative comparisons of the above mentioned results on a model by model basis. Figures 18 to 20 show the pictures at approximate bit rates of 1, 0.5 and 2 bits/pixel encoded by the above mentioned methods. Figures 21 and 22 show the effect of channel errors on encoding schemes. Generally, the adaptive classification is quite robust. The nonadaptive and adaptive variance schemes can be made less sensitive to channel errors by reducing somewhat the horizontal correlation parameter. The SC1 model adaptive variance estimation seems less sensitive than SC2 model. Figures 23 and 24 show the encoded images in the presence of channel errors. Figure 25 compares the original, the nonadaptive separable model hybrid coded, and the adaptive classified (based on actual measured statistics model) hybrid coded images at 1 bit/pixel.

Finally, observe that as an image column size is increased, the ambiguity in classifying it into different classes will also go up because the activity in a large image column is not likely to be uniform, so as to be represented by its variance.

4. Conclusions

Based on the above experiments and the theory discussed in the previous chapters, the following conclusions are made.

(i) Depending on the computational complexity, adaptive hybrid coding schemes can improve the signal to noise ratio over the standard nonadaptive hybrid coding scheme (based on separable covariance model) by 2 to 5 Db at bit rates around 1 bit/pixel.

(ii) The adaptive variance estimation technique is useful when it is desired to maintain the performance of a hybrid coder, designed for a nominal statistics, in the face of changing statistics (e.g, the correlation parameter). If the image data statistics is close to

the design parameters, then the improvement over nonadaptive scheme is nominal (1 to 1.5 Db at 1 bit/pixel). This techniques is effective at relatively higher bit rates ≥ 1 bit/pixel and for relatively larger image column sizes. It should be considered not so much for achieving low transmission rates as for achieving higher SNR at moderate rates. The complexity over a nonadaptive scheme is marginal, and it can be hardware implemented very easily. Finally, it maintains a constant bit rate from one image column to the next and its performance in the presence of channel errors is no worse than the nonadaptive schemes.

(iii) The adaptive classification scheme is well suited for designing low bit rate transmission/storage systems. It is robust and permits a wide range (3 to 6 Db) of performance vs. complexity trade-offs. Reasonable results could be expected at ½ bit/pixel (see Figure 19).

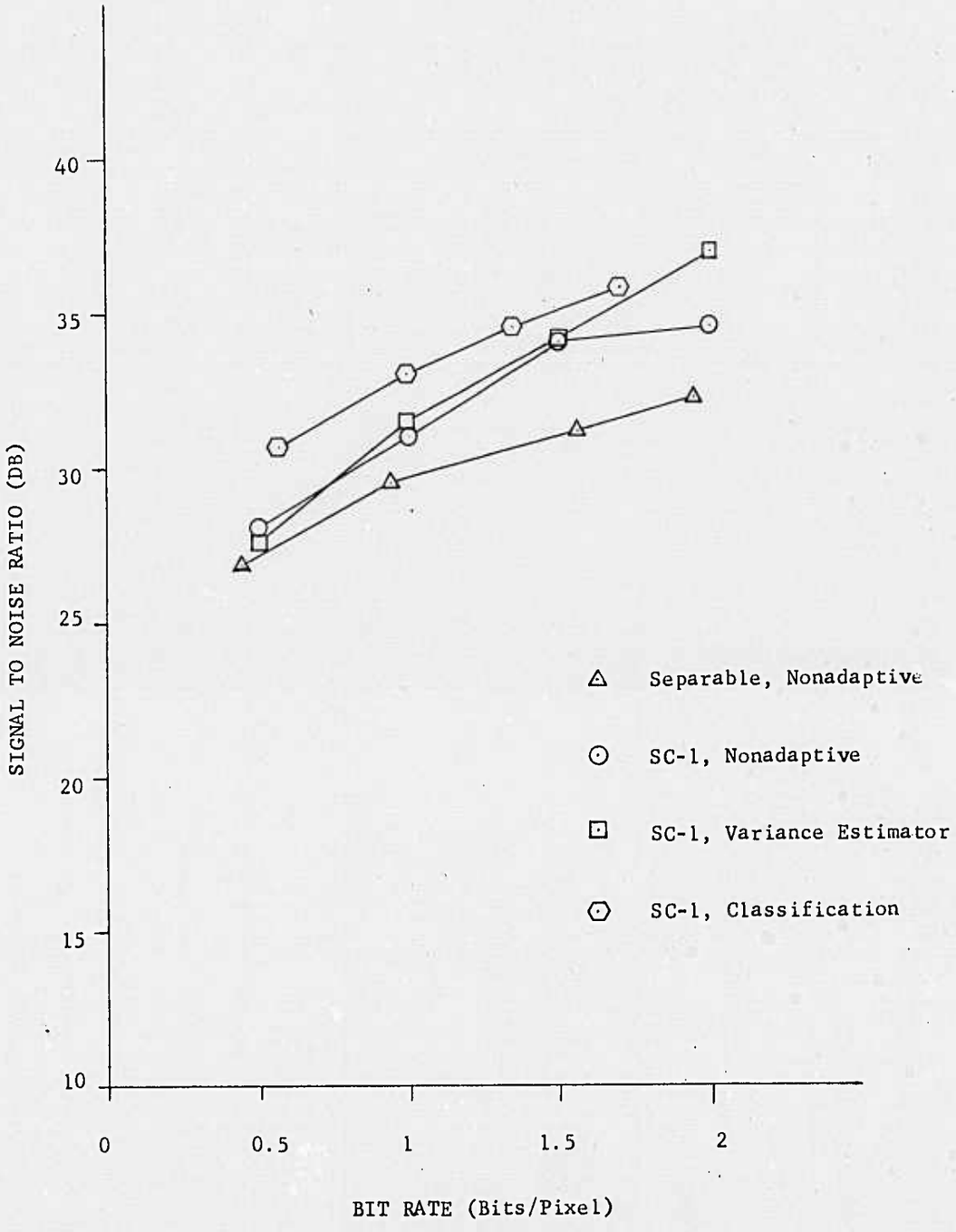


FIGURE 17: Comparison of Nonadaptive and Adaptive Hybrid Coding Schemes, SC1 Model

350

TABLE VII- 256 x 256 GIRL IMAGE, SEPARABLE MODEL, HYBRID COING

DESIRED AVERAGE BIT RATE	2		1.5		1		0.5	
METHOD	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE
NONADAPTIVE SEPARABLE COVARIANCE MODEL	32.26	1.94	31.26	1.56	29.57	0.94	26.91	0.44
ADAPTIVE VARIANCE ESTIMATION SEPARABLE COVARIANCE MODEL	36.24	1.94	34.33	1.56	31.31	0.94	27.47	0.44
ADAPTIVE CLASSIFICATION ACTUAL MEASUREMENTS MODEL	40.59	2.02	38.07	1.52	35.01	0.97	31.42	0.5

TABLE VIII- 256 x 256 GIRL IMAGE, SEMICAUSAL MODEL SC1, HYBRID CODING

DESIRED AVERAGE BIT RATE	2		1.5		1		0.5	
METHOD	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE
NONADAPTIVE	34.55	2	33.07	1.5	31.05	1	28.13	0.5
ADAPTIVE VARIANCE ESTIMATION	35.97	2	34.25	1.5	31.48	1	27.73	0.5
ADAPTIVE CLASSIFICATION	35 80	1.70	34.55	1.36	33.07	1	30.74	0.5

TABLE IX - 256 x 256 GIRL IMAGE, SEMICAUSAL MODEL SC2, HYBRID CODING

DESIRED AVERAGE BIT RATE	2		1.5		1		0.5	
METHOD	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACUTAL RATE	SNR	ACTUAL RATE
NONADAPTIVE	32.30	2	30.91	1.5	29.53	0.94	27.03	0.5
ADAPTIVE VARIANCE ESTIMATION	36.25	2	33.17	1.5	31.00	0.94	27.48	0.5
ADAPTIVE CLASSIFICATION	32.05	1.67	31.31	1.38	29.97	0.98	28.20	0.5

TABLE X - 256 x 256 GIRL IMAGE, SEPARABLE MODEL, HYBRID CODING WITH CHANNEL ERRORS

BIT ERROR PROBABILITY Pe	0		0.0001		0.001	
METHOD	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE
NONADAPTIVE SEPARABLE COVARIANCE MODEL	29.57	0.94	29.38	0.94	27.99	0.94
ADAPTIVE VARIANCE ESTIMATION SEPARABLE COVARIANCE MODEL	31.31	0.94	28.17	0.94	11.77	0.94
ADAPTIVE CLASSIFICATION ACTUAL MEASUREMENTS MODEL	35.01	0.97	35.00	0.97	30.88	0.97

TABLE XI- 256 x 256 GIRL IMAGE, SEMICAUSAL MODEL SC1, HYBRID CODING WITH CHANNEL ERRORS

BIT ERROR PROBABILITY Pe	0		0.0001		0.001	
METHOD	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE
NONADAPTIVE	31.05	1	30.97	1	29.60	1
ADAPTIVE VARIANCE ESTIMATION	31.48	1	23.45	1	21.60	1
ADAPTIVE CLASSIFICATION	33.07	1	32.95	1	30.24	1

TABLE XII- 256 x 256 GIRL IMAGE, SEMICAUSAL MODEL SC2, HYBRID CODING WITH CHANNEL ERRORS

BIT ERROR PROBABILITY Pe	0		0.0001		0.001	
METHOD	SNR	ACTUAL RATE	SNR	ACTUAL RATE	SNR	ACTUAL RATE
NONADAPTIVE	29.53	0.94	29.31	0.94	25.17	0.94
ADAPTIVE VARIANCE ESTIMATION	31.00	0.94	29.95	0.94	14.33	0.94
ADAPTIVE CLASSIFICATION	29.97	0.98	29.97	0.98	28.97	0.98

a) SC1 Model, Nonadaptive

b) SC1 Model,Adaptive Variance Estimation

c) SC1 Model, Adaptive Classification

d) SC2 Model, Nonadaptive

e) SC2 Model,Adaptive Variance Estimation

f) SC2 Model, Adaptive Classification

FIGURE 18: HYBRID ENCODED 256 x 256 GIRL IMAGE AT 1 BIT/PIXEL APPROXIMATELY

a) SC1 Model,Nonadaptive

d)SC2 Model, Nonadaptive

b) SC1 Model, Adaptive Variance Estimation

e)SC2 Model, Adaptive Variance Estimation

c) SC1 Model, Adaptive Classification

f)SC2 Model, Adaptive Classification

FIGURE 19: HYBRID ENCODED 256 x 256 GIRL IMAGE AT 0.5 BIT/PIXEL APPROXIMATELY

a)SC1 Model, Nonadaptive(2bits/pixel)

d)SC2 Model,Nonadaptive(2bits/pixel)

b)SC1 Model,Adaptive Variance Est.(2bits/pixel)

e)SC2 Model,Adaptive Variance Est.(2bits/pixel)

c)SC1 Model,Adaptive Classification(1.7bits/pixel)

f)SC2 Model,Adaptive Classification(1.67bits/pixel)

74

FIGURE 20: HYBRID ENCODED 256 x 256 GIRL IMAGE AT 1.7 to 2 BITS/PIXEL APPROXIMATELY

AVERAGE BIT RATE $\simeq$ 1

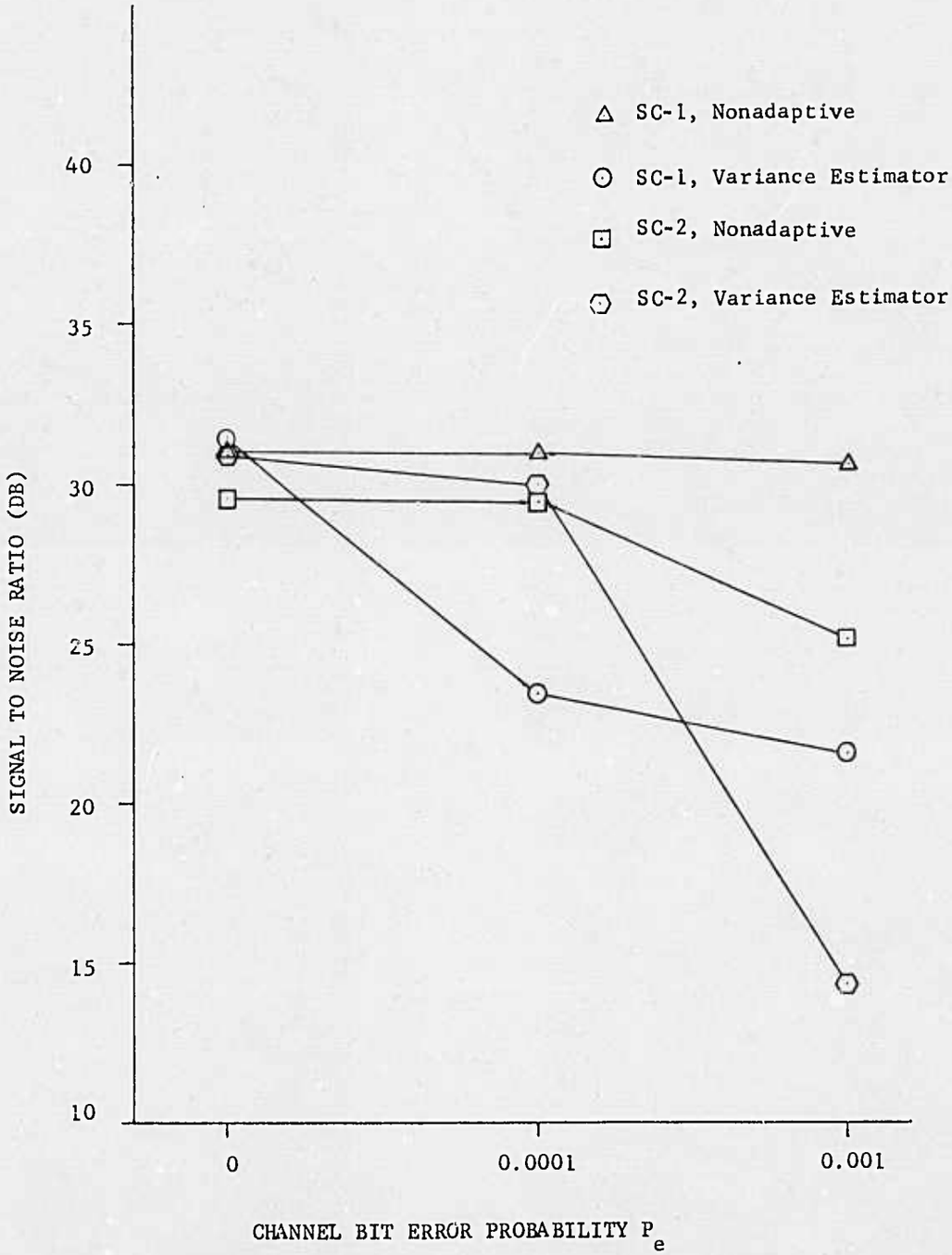


FIGURE 21: Effect of Channel Errors

CLASSIFICATION WITH DESIRED AVERAGE BIT RATE = 1

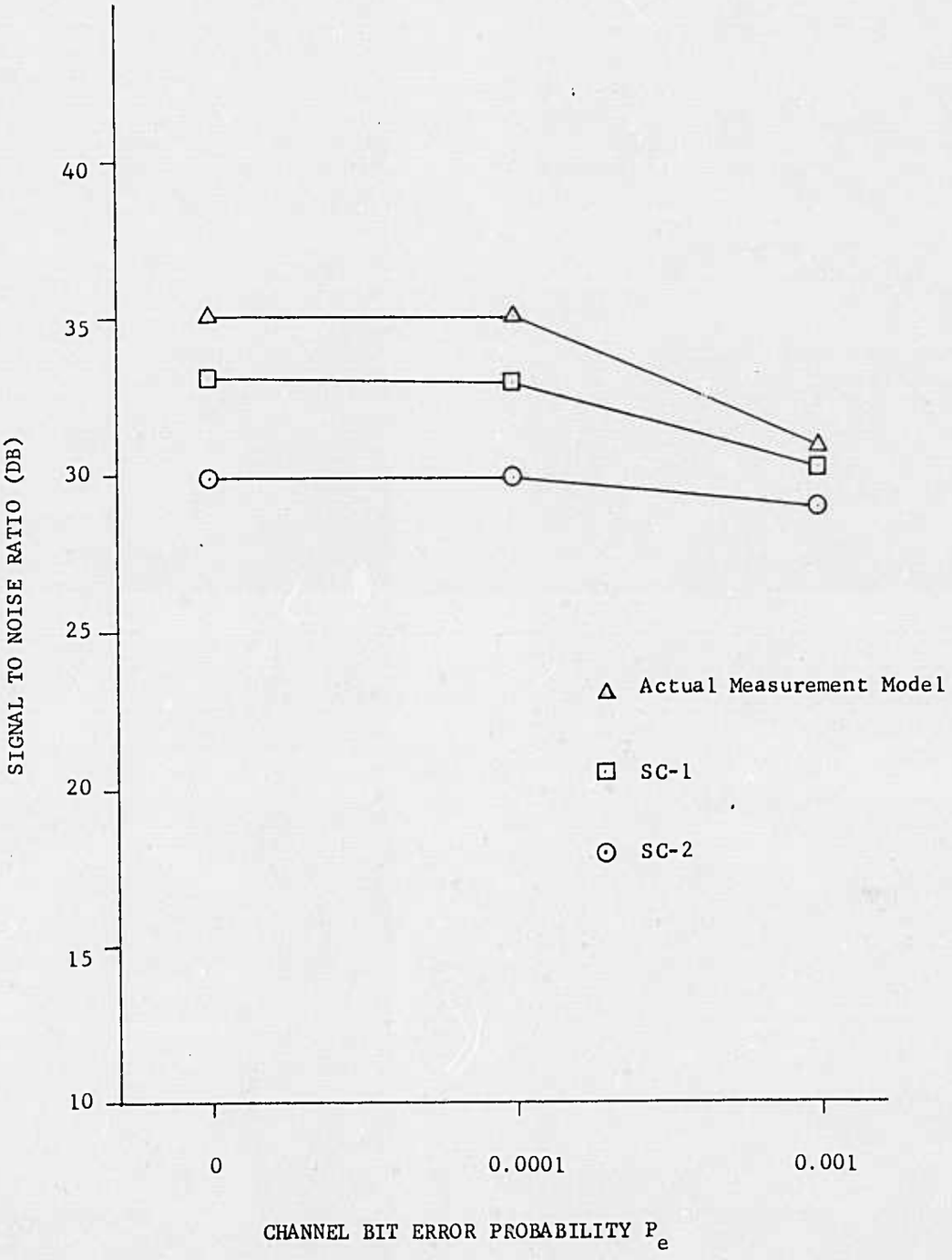


FIGURE 22: Effect of Channel Errors

358

a) SC1 Model, Nonadaptive

d) SC2 Model, Nonadaptive

b) SC1 Model, Adaptive Variance Estimation

e) SC2 Model, Adaptive Variance Estimation

c) SC1 Model, Adaptive Classification

f) SC2 Model, Adaptive Classification

FIG. 23: HYBRID ENCODED 256 × 256 GIRL IMAGE WITH CHANNEL ERROR P_e = 0.0001 AT 1BIT/PIXEL APPROX.

a) SC1 Model, Nonadaptive

b) SC2 Model, Nonadaptive

b) SC1 Model, Adaptive Variance Estimation

c) SC2 Model, Adaptive Variance Estimation

c) SC1 Model, Adaptive Classification

f) SC2 Model, Adaptive Classification

FIG. 24: HYBRID ENCODED 256 x 256 GIRL IMAGE WITH CHANNEL ERROR P =0.001 AT 1 BIT/PIXEL APPROX.

a) Original

b) Hybrid Coding(Separable Model, Cosine/DPCM)

c) Adaptive Classification (Actual Measurements Model)

FIGURE 25: NONADAPTIVE VS. ADAPTIVE HBYRID CODING AT 1 BIT/PIXEL APPROXIMATELY

REFERENCES

1. A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Transaction on Comm., vol. COM-22, No. 5, May 1974, pp. 614-624. Also see USCEE Report 444, pp. 34-40, March 1973.

2. A.K. Jain and E. Angel, "Image Restoration, Modeling, and Reduction of Dimensionality," Tech. Report 72-50, USC-113P19-31, Sept. 1972. Also see USCEE Report 444, March 1973, pp. 66-76, and IEEE Trans. Computers, vol. C-23, pp. 470-476, May 1974.

3. E. Angel and A.K. Jain, "Filtering of Multidimensional Diffusion Processes," 6th Asilomar Conf. Circuits and Syst., Nov. 1972. Also see "Frame to Frame Restoration of Diffusion Images," (to appear in IEEE Trans. Auto. Control)

4. J.A. Roese, W.K. Pratt, and G.S. Robinson, "Interframe Cosine Transform Image Coding," IEEE Trans. Commun., vol. COM-25, pp. 1329-1339, Nov. 1977.

5. A.K. Jain, "Partial Differential Equations and Finite-Difference Methods in Image Processing, Part I: Image Representation," J. Optimization Theory and Appl., vol. 23, pp. 65-91, Sept. 1977.

6. A.K. Jain and J.R. Jain, "Partial Differential Equations and Finite-Difference Methods in Image Processing, Part II: Image Restoration," (to appear in IEEE Trans. Auto. Control)

7. N. Ahmed, T. Natarajan and K.R. Rao, "Discrete Cosine Transform," IEEE Transactions on Computers, pp. 90-93, January 1974.

8. A.K. Jain, "Some New Techniques in Image Processing," IMAGE SCIENCE MATH., Proceedings ONR Symposium on Current Problems in Image Science, (1967) (C.O. Wilde & E.Barrett, Editors), West. Periodicals Co., No. Hollywood, CA, 1977.

9. A.K. Jain, "A Sinusoidal Family of Unitary Transforms," Dept. of EE, SUNY Buffalo, Tech. Report, (submitted for publication).

10. A.K. Jain, "Discrete Random Fields and Image Models," (to appear).

11. A.K. Jain, "A Semicausal Model for Recursive Filtering of Two-Dimensional Images," IEEE Trans. on Computers, vol. C-26, No. 4, pp. 343-350, April 1977.

12. A. Segall, "Bit Allocation and Encoding for Vector Sources," IEEE Trans. on Information Theory, vol. IT-22, No. 2, pp. 162-169, March 1976.

13. B. Fox, "Discrete Optimization via Marginal Analysis," Management Science, vol. 13, pp. 201-216, Nov. 1966.

14. A.K. Jain, "A Fast Karhunen Loeve Transform for a Class of Random Processes," IEEE Trans. Commun., vol. COM-24, pp. 1023-1029, Sept. 1976.

15. S.H. Wang, "Analytic Models for Characteristics of Practical Quantizers," Dept. of Electrical Engineering, SUNY Buffalo, Tech. Report, (to appear).

16. P. Castellino, G. Madena, L. Nebbia, and C. Scngliala, "Bit Rate Reduction by Automatic Adaptation of Quantizer Step-Size in DPCM Systems," In Proc. 1974 Int. Zurich Seminar Digital Communications, pp. B6(1)-B6(6), 1974.

17. J. Max, "Quantizing for Minimum Distortion," IRE Trans. on Information Theory, vol. IT-6, pp. 7-12, March 1960.

18. R.W. Means, E.H. Wrench and J. J. Whitehouse, "Image Transmission via Spread Spectrum Techniques," ARPA Quarterly Technical Report, ARPA-QR6, Naval Undersea Center, San Diego, California 92132, Jan.-Dec. 1975.

19. V.R. Algazi, "Useful Approximations to Optimum Quantization," IEEE Trans. on Communication Technology, vol. COM-14, pp. 297-301, June 1966.

APPENDIX I: OPERATOR PERFORMANCE EVALUATION OF MINI-RPV VIDEO IMAGE BANDWIDTH REDUCTION/COMPRESSION TECHNIQUES

Final Report for period June 1975–February 1976
Contract N66001-75-C-0228

June 1976

by
M L Hershberger
Display Systems and Human Factors Department
Hughes Aircraft Company
Culver City CA 90230

364

PREFACE

The research covered herein was initiated by the Naval Undersea Center, San Diego, California as part of research and development for the Image Transmission via Spread Spectrum Techniques project sponsored by the Defense Advanced Research Projects Agency. The research was concerned with man-in-the-loop evaluation of video bandwidth reduction and compression techniques for mini-RPVs. The technical coordinators for the Naval Undersea Center were Dr. Robert W. Means and Dr. Edward H. Wrench, Code 608. The research was conducted by the Display Systems and Human Factors Department of Hughes Aircraft Company, Culver City, California, under Navy contract N66001-75-C-0228. Mr. M. L. Hershberger of Hughes Aircraft Company was Project Manager.

Special acknowledgement is gratefully made of the following contributions to the performance of the research. Mr. Harper J. Whitehouse of the Naval Undersea Center, Code 6003, was instrumental in supporting the need for this research and defining the research area. Mr. R. L. Andrews of Hughes Aircraft Company interfaced the Naval Undersea Center hybrid cosine/DPCM transform system with the Hughes RPV simulator and made the two equipments work together to conduct the research. Dr. L. A. Scanlan of Hughes Aircraft Company made major contributions to the research approaches used to conduct the two studies performed. Mr. C. E. Dickson collected and analyzed the data for the two studies. The author is particularly indebted to Dr. Robert Means, Dr. Edward Wrench, and other members of the Naval Undersea Center who developed the hybrid cosine/DPCM transform equipment which was essential to this effort.

CONTENTS

Section		Page
1	INTRODUCTION AND SUMMARY	1
	Introduction	1
	Summary	3
2	LABORATORY EQUIPMENT	5
	Cosine/DPCM transform equipment	5
	RPV simulator	8
	Television scanner	8
	Digital refresh memory	10
	Digital computer	10
	Operator's control console	13
	Experimenter's control console	14
3	BANDWIDTH COMPRESSION STUDY	17
	Introduction	17
	Research parameters	18
	Research design	19
	Operators	19
	Targets	19
	Study procedure	21
	Performance measure	22
	Results and discussion	22
	Conclusions and recommendations	25
4	FRAME RATE STUDY	29
	Introduction	29
	Research parameter	30
	Research design	30
	Operators	31
	Target scene	31
	The tracking task	32
	Laboratory equipment	33
	Study procedure	34
	Performance measures	35
	Results and discussion	35
	Designation time	35
	Designation error	37
	RMS tracking error	38
	Conclusions and recommendations	39
	REFERENCES	41
	APPENDIX A	43

3606

FIGURES

Figure		Page
1.	Hybrid cosine transform DPCM block diagram	7
2.	Hughes RPV simulator	9
3.	Cosine/DPCM equipment, RPV simulator interface block diagram	11
4.	Functional diagram of mini-RPV simulator	13
5.	Operator's control console	14
6.	Experimenter's control console	15
7.	Research design used in bandwidth compression study	20
8.	Effects of bandwidth compression on operator tactical target recognition performance	23
9.	Effects of bit error rate noise jamming on operator tactical target recognition performance	24
10.	Estimated operator performance/bandwidth compression function	25
11.	Effect of video frame rate on operator target recognition performance	30
12.	Research design used in frame rate study	31
13.	Tank target used in the study	32
14.	Hand control shaping function	33
15.	Effects of frame rate on target designation time	36
16.	Effects of frame rate on precision target designation accuracy	37
17.	Frame rate effects on target tracking accuracy	38

SECTION I

INTRODUCTION AND SUMMARY

INTRODUCTION

Remotely Piloted Vehicles (RPV's) may be used for a variety of missions in the future. It is unlikely that such drones can be operated completely automatically; therefore, sensor and control signal data must be transmitted via data link to a ground controller. Control signals from the ground controller will be data linked to the RPV. It is obvious that for the mission to be successful and for the RPV concept to be feasible, the data link must remain secure (unjammed). The lower the bandwidth requirement of the data link, the greater the possibility of maintaining its security. The sensor data from the RPV to the controller have much higher bandwidth requirements than flight data and control signals; therefore, the critical bandwidth limitation is for the sensor.

Operational experience gathered in the Far East has demonstrated the effect that jamming can have on communications links and has pointed out a real need for developing data link systems that are more immune to electronically hostile environments. Since many RPV missions are dependent upon the successful operation of wide-band video data links to achieve mission success, the quality of the information transmitted and displayed to the operator is a crucial factor in the determination of RPV system effectiveness. Reconnaissance and targeting RPV missions planned for Army mini-RPV's require the use of forward-looking sensors (like TV and FLIR) that are carried on-board for the purpose of relaying information to the remote operator who must find targets and point a laser designator on the basis of this information. The problem is to determine methods for decreasing the susceptibility of the RPV system to jamming while maintaining acceptable operator performance.

A promising method for improving data link performance is the reduction of video data link bandwidth from that required for conventional TV systems while transmitting sufficient data to enable the operator to perform his tasks. The bandwidth limitation problem can be approached in two ways: bandwidth reduction and bandwidth compression. A variety of technqiues are available

for reducing and compressing the bandwidth of the transmitted sensor image. These include frame rate reduction, reduction in video resolution, and image compression techniques in the Cosine, Fourier, or Hadamard domains. These techniques are designed to reduce the bandwidth of the transmitted imagery while maintaining sufficient quality to permit satisfactory mission performance by the operator.

The Naval Undersea Center (NUC) has, under sponsorship of the Defense Advanced Research Projects Agency, been working on the development of a spread spectrum image transmission system to provide increased anti-jam protection to a television link from small remotely piloted vehicles (mini-RPV's). The system selected for development uses transform encoding and frame rate control to remove redundancy in television imagery.

In early 1975, a hybrid discrete cosine transform system with differential pulse code modulation and frame rate control was built by NUC. This first system has 100 by 100 element resolution with 5, 2, 1, and 0.5 bits per picture element transform compression and frame rates of 5, 2.5, and 1.25 frames per second. Electronics to simulate bit error rate jamming were also built into the equipment.

The 100 by 100 element resolution cosine/DPCM transform system was designed and developed to satisfy two major objectives: 1) to demonstrate the feasibility of small light weight, low power transform technology using charge transfer devices for use in mini-RPV's and 2) to provide a vehicle which could be used to evaluate the effects of bandwidth reduction and compression on RPV operator performance. The research described in this report was concerned with the second objective.

A channel bandwidth of 20 M bits per second has been planned for the mini-RPV video bandwidth. This channel bandwidth can be achieved through the judicious selection of video resolution, frame rate, and data compression which are the principle parameters that determine video channel bandwidth. There are many combinations of these parameters that will result in the desired 20 M bits per second channel bandwidth, permitting design trade-offs. Mini-RPV system designers have selected a 256 by 256 element video resolution for reconnaissance and targeting missions based on previous tactical target recognition research. There are, however, limited data to

guide the designer in the selection of bandwidth compression for tactical target recognition and frame rate for targeting that will result in the desired channel bandwidth without degrading operator performance beyond acceptable limits. Since a major criterion for selecting mini-RPV compression level and frame rate is the performance of the RPV ground controller, operator performance data as a function of bandwidth compression and video frame rate were needed. The research described in this report was conducted to satisfy this need.

SUMMARY

Two man-in-the-loop laboratory research studies were conducted during the research program. In the first study, the effects of bandwidth compression and bit error rate jamming on operator tactical target recognition performance were investigated. The NUC developed cosine/DPCM transform system interfaced with a Hughes RPV simulator was used in the first study. The effects of frame rate on operator tactical target designation and tracking were investigated in the second study. The research methodology employed and the results of the two studies are summarized below.

The bandwidth compression study investigated the nine combinations of 5, 2, and 1 bits per pixel compression and zero, 10^{-3}, and 10^{-2} bit error rate jamming. In addition, a special uncompressed video condition was investigated as a baseline comparison. The Naval Undersea Center's hybrid cosine/DPCM transform system was interfaced with the Hughes RPV simulator to provide the facility for conducting the study. The operators' task was the recognition of tactical vehicle size targets. The number of sensor resolution elements across the height of the tactical targets was the performance measure used to assess the effects of bandwidth compression and bit error rate jamming on operator target recognition performance.

The results of the study showed there was no degradation of operator performance, compared to the baseline uncompressed video condition, until 6:1 compression (1 bit per pixel) was reached. Bit error rate jamming had no effect on operator tactical target recognition performance. The number of resolution elements required for target recognition at the uncompressed video condition and the 5 and 2 bits per pixel compressed video conditions

agreed well with other research findings (Johnson, 1958). It was concluded that mini-RPV's could operate at 10^{-2} bit error rate jamming levels with 1.5 bits per pixel (4:1 compression) without any degradation of operator performance.

The frame rate study investigated 0.94, 1.88, 3.75, 7.5, 15 and 30 frame per second frame rates with a simulated 3-axis stabilized sensor pointing mini-RPV system. The Hughes RPV simulator without the NUC cosine/DPCM transform system was used to conduct the study. The operator's task was to designate a defined aimpoint on a Stalin tank and track the tank aimpoint for 30 seconds. An RPV flying at a 762-meter altitude, a speed of 50 meters per second, and a ground range-to-target of 3000 meters was simulated to provide a tangent function tracking task. The target designation time, target designation error, and RMS tracking error were measured.

The general effects of frame rate were found to be the same for all three performance measures. Operator performance improved rapidly as frame rate increased from 0.94 to 3.75 frames per second. There was a slight but nonstatistically reliable improvement in operator performance as frame rate increased from 3.75 to 15 frames per second. Between 15 and 30 frames per second, operator performance was constant. The results thus indicate that video frame rate can be reduced to 3.75 frames per second with little if any degradation of operator target designation and tracking performance. Design of mini-RPV's with a 3.75 frames per second frame rate will provide an 8:1 bandwidth reduction compared to the standard 30 frame per second video frame rate.

SECTION 2

LABORATORY EQUIPMENT

The laboratory studies required the controlled variation of bandwidth compression, bit error rate jamming, and video frame rate in the context of an RPV mission task setting to measure operator target recognition, designation, and tracking performance. The Naval Undersea Center's 100 by 100 element resolution hybrid discrete cosine transform with differential pulse code modulation and bit error rate jamming simulation provided the means to vary bandwidth compression and bit error rate jamming. Hughes Aircraft Company's RPV simulator was used to vary video frame rate, simulate RPV vehicle dynamics, provide television video, provide an operator's console for display of sensor video and control of sensor pointing, and provide performance scoring and data recording functions. The cosine/DPCM transform equipment was interfaced with the RPV simulator in order to conduct the study of bandwidth compression and bit error rate jamming. The RPV simulator without the cosine/DPCM equipment was used to conduct the study of video frame rate effects on operator target designation and tracking. The cosine/DPCM transform and RPV simulator equipment is described in this section.

COSINE/DPCM TRANSFORM EQUIPMENT

The TV bandwidth compression system used for the study of bandwidth compression was developed at the Naval Undersea Center under DARPA support. The system accepts video at 5 frames per second in a 100 by 100 picture element format. Video is compressed using a discrete cosine transform in the horizontal direction and differential pulse code modulation (DPCM) in the vertical direction. There are four output data rates — 25, 50, 100 and 250 K bits per second, corresponding to 0.5, 1, 2, and 5 bits per picture element, respectively.

The compression technique, referred to as a hybrid cosine transform DPCM, was implemented using bucket brigade transversal filters to perform the cosine transform. A chirp Z algorithm was used which decomposed the cosine transform into a pre-multiplication by a discrete

complex chirp, convolution with a complex chirp, and post-multiplication by a complex chirp. The exact form is given in the following equation:

$$G_n = e^{-j\pi n^2} \sum_{k=0}^{N} \hat{g}_n \; e^{-j\pi k^2} \; e^{j\pi(k-n)^2} ,$$

where G_n is the transform coefficient and $\hat{g}_n$ represents the data points with $\hat{g}_o = 1/2\, t_o$ and $\hat{g}_n = g_n$ for $n \neq 0$. The advantage of this algorithm is that the reference function for the convolution is fixed and can therefore be implemented using a transversal filter with fixed tap weights. In this case, a 199 point bucket brigade transversal filter was used to implement a 100 point discrete cosine transform.

With this system, an entire line of video (100 picture elements) was transformed in a single block. The 100 cosine coefficients were digitized and fed into a differential pulse code modulator. A block diagram of the system is shown in Figure 1. The DPCM stores a reference of the coefficients from the previous line, and the differences between the reference and the coefficients from the current line are obtained. These 100 differences (one for each of the 100 coefficients) are quantized with a variable number of bits depending on their importance to the picture. The differences of the low frequency coefficients are quantized with a large number of bits (typically 5), and the high frequency coefficient differences receive fewer bits (zero bits may be assigned at high compression rates). The total number of bits used to quantize all coefficient differences was determined by the desired bandwidth compression ratio. The quantized coefficient differences were transmitted by the system. At the receiver, the differences were integrated to reconstruct the actual cosine transform coefficients. These coefficients were fed into an inverse cosine transform (the cosine transform is its own inverse) to reconstruct the picture elements. These picture elements were stored in a frame store memory so that a flicker free, 60 field per second display was obtained.

The system had the capability of introducing errors in the transmitted bit stream to simulate channel errors. A bit error rate generator produced random errors that simulated a binary symmetric channel. The probability of a bit error was programmable by switches. A maximal

D3906/3425

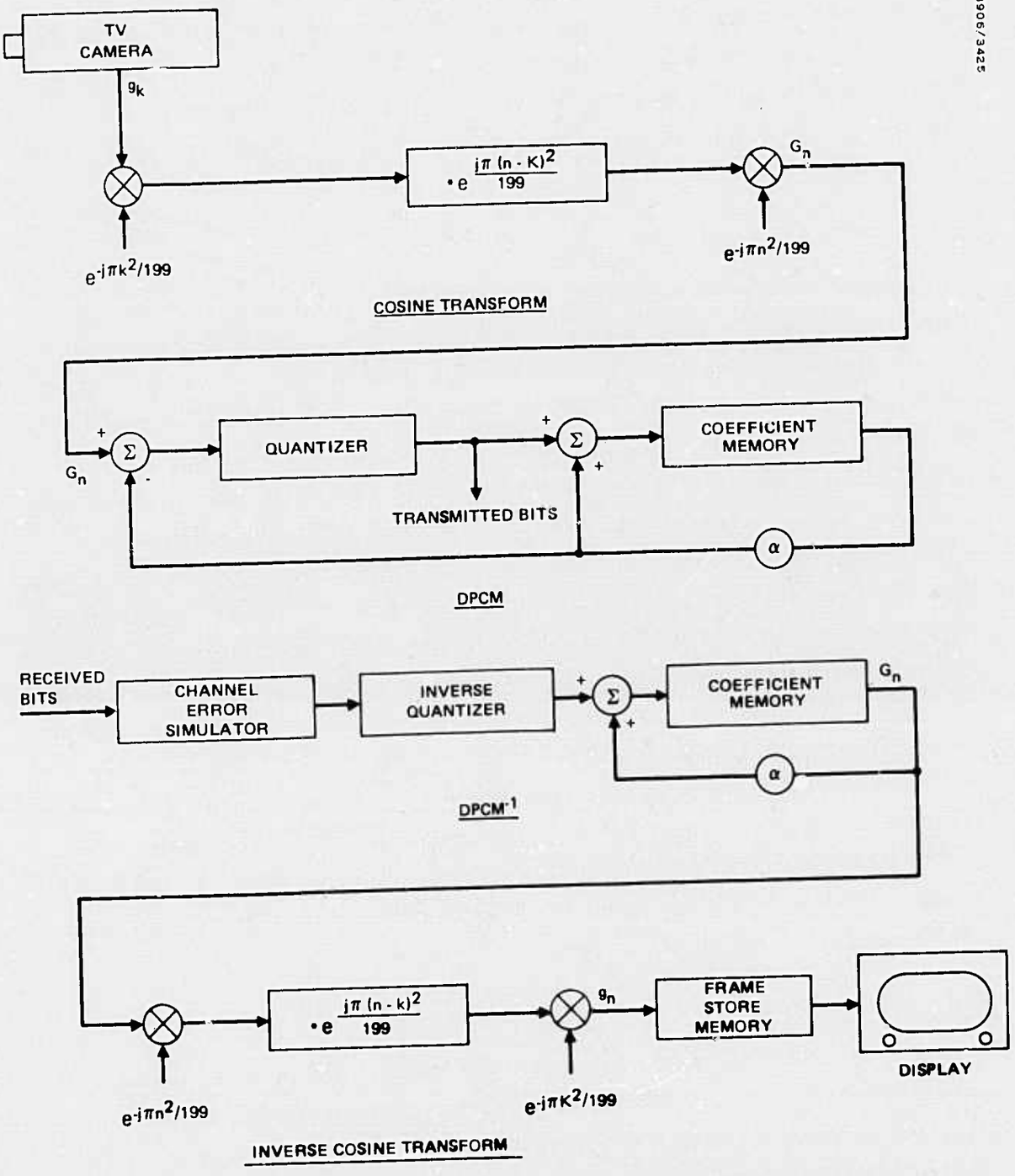


Figure 1. Hybrid cosine transform DPCM block diagram.

374

length shift register sequence (2^{31}) was used to generate a 12 bit uniformly distributed random variable. The value of this random variable was compared to the number set in the switches. An error was then generated every time the value of the random variable exceeded the value in the switches. The probability of a bit error was therefore random and equal to the number set in the switches (0, 1, ... 4095) divided by 4096. This allowed any probability of bit error between 0 and 1/2 in steps of 2.4×10^{-4} to be simulated.

RPV SIMULATOR

The Hughes RPV simulator is depicted in block diagram form in Figure 2. Video is simulated in a television-type format at a selected frame rate. It is digitized at the spatial and intensity resolution desired and either transmitted in the spatial domain or transformed and transmitted. Jamming noise is introduced to randomly change the bits at the desired error rate. The digital video is then buffered (decoded back to the spatial domain if initially transformed) prior to loading into a refresh memory. The refresh memory is required to convert low frame rate formats to a flicker-free standard television format. Output video is D/A converted and mixed with display sensor and cursor video and presented on a television monitor. All necessary parameters are varied from a central control console. A digital computer is used to provide the flight and sensor dynamics and provides experimental control and scoring.

The principal parts of the simulator that were used in the program include: a variable scan rate television camera, an A/D converter, a 1.5 million bit refresh memory, a D/A converter, various electronics for synchronization and mode control, a Sigma 5 digital computer, an operator's console, and an experimenter's console.

Television Scanner

The dynamic television scanner consists of a high quality variable scan rate closed-circuit television camera, automated 20:1 zoom lens, rotating/translating film transport, and film illumination source. The zoom lens is computer controlled and provides simulation of an RPV closing on a target as well as an optical zoom function of an RPV sensor. The 3-axis film transport, which takes 9- by 9-inch film, provides manual

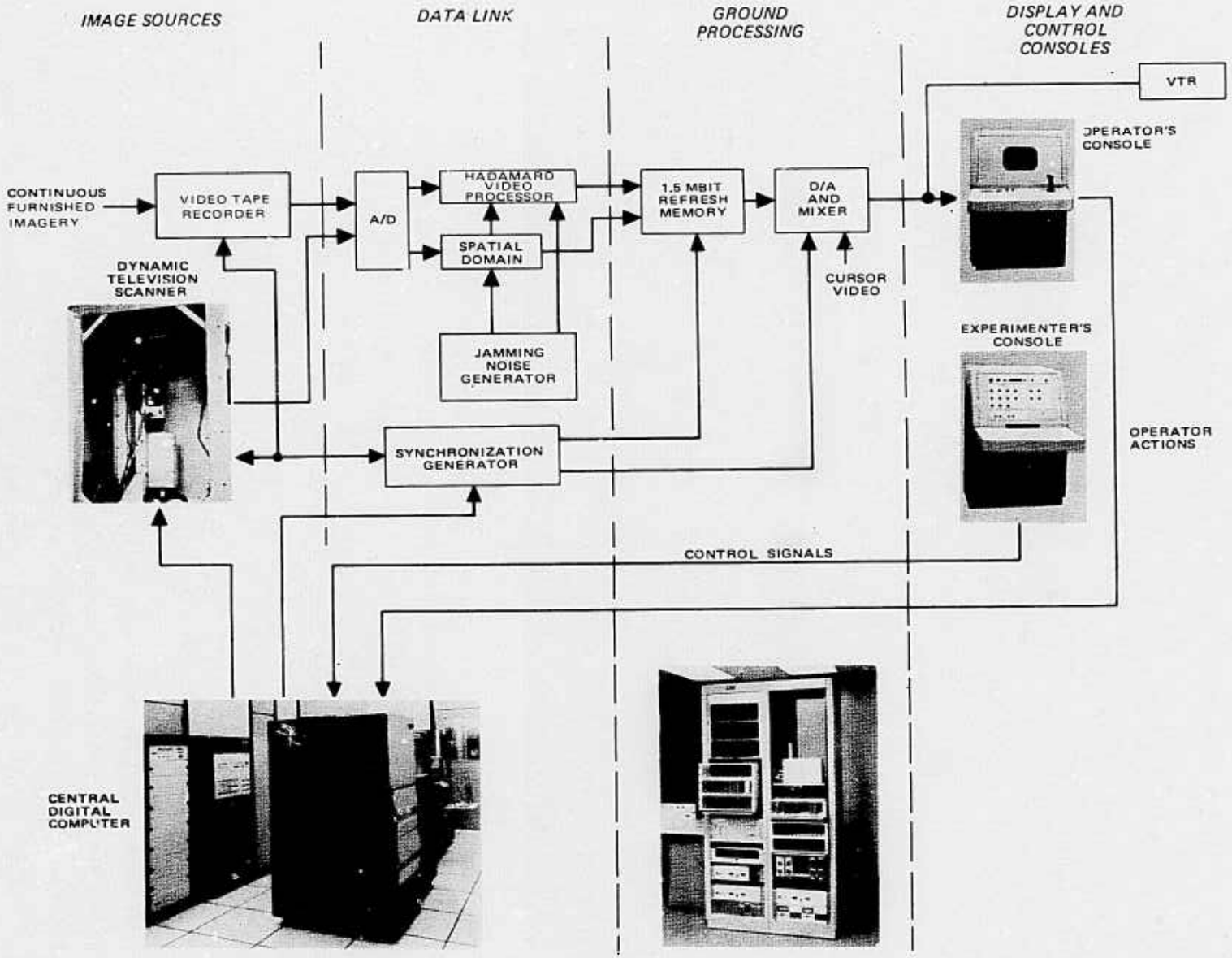


Figure 2. Hughes RPV simulator.

Figure 2. Hughes RPV simulator.

(experimenter controlled) or computer-controlled simulation of sensor pointing. Illumination is provided by continuous bulb or if frame rates below 30 frames per second are simulated by a strobe light.

Digital Refresh Memory

In an actual RPV mechanization, low scan rate video would be received on the ground and displayed to the RPV operator. It is unacceptable to provide a display which is refreshed at a rate slower than 40 to 50 Hz, since the flicker would cause operator fatigue and degrade operator performance. Therefore, the input video must be stored and used to refresh the display at a flicker-free rate. The closed-circuit television camera is scanned at the desired frame rate (0.1 to 30 Hz) and utilizes a 1.5 M bit digital scan converter memory to provide the display refresh function. The refresh memory, implemented with MOS RAMS, consists of 32,768 words of 48 bits each. The digitized video is received from the simulated data link in parallel digital words of up to 6 bits each (depending on the gray shade encoding selected). Eight words are accumulated in a 48-bit buffer and loaded simultaneously into the refresh memory. Readout is accomplished in the same format, but always at a 30 frame per second, 60 fields per second rate.

The digitized video is read out of the refresh memory, reformatted, D/A converted, and mixed with composite sync pulses to drive the displays. A positionable cursor is also provided, generated by comparing hand control position signals with display position addresses. This symbol video is mixed with the sensor video and displayed. As illustrated in Figure 2, the cosine/DPCM equipment with bit error rate jamming simulation was interfaced between the RPV simulator television scanner and the 1.5 M bit digital refresh memory. A block diagram of the interface is shown in Figure 3.

Digital Computer

Automatic simulator functions and vehicle/sensor dynamics computations for the RPV simulator are performed by a Xerox Sigma 5 digital computer. The computer software written for the RPV simulator performs the following functions: vehicle kinematics and sensor dynamics, scoring, data recording, and experimental conditions set-up.

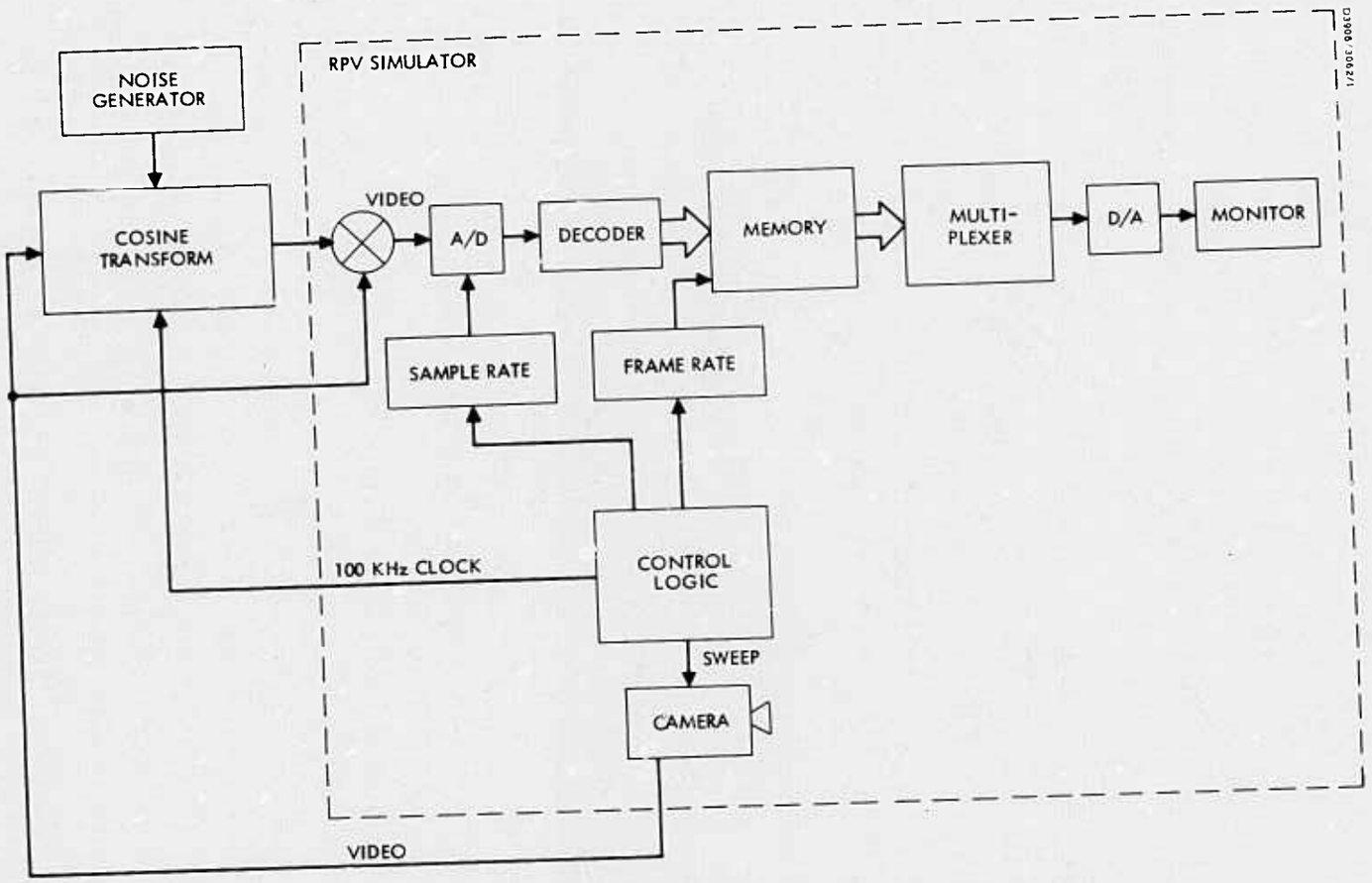


Figure 3. Cosine/DPCM equipment, RPV simulator interface block diagram.

Vehicle Kinematics and Sensor Dynamics

Vehicle kinematics and sensor dynamics are computed from a model which has been designed to provide typical mini-RPV performance characteristics. The basic computation loop operates at a rate of 20 Hz. The RPV is assumed to be flying at constant velocity and altitude over the ground. The sensor is pointing toward the ground at an initial depression angle. The sensor pointing angle can be driven in azimuth and elevation by a slew rate hand control in which the slew rate is proportional to the square of the hand control position.

Performance Scoring and Data Recording

During a trial run, the computer constantly measures the RMS tracking error between the actual target location and the pointing angle of the sensor. At the end of each run the subject number, test conditions, trial time, designation error, and tracking error are printed out.

Experimental Conditions Set-up

The experimenter has the ability to control a large number of experimental variables from his console. This same control can be exercised by the computer when the master control mode is set to "automatic". By using this capability, the experimenter can pre-define a series of experimental conditions which are selectable. As the experiment progresses, the experimenter can select each new set of parameter values by setting digi-switches to the next test number. The computer then automatically sets all of the variables to the desired values.

Taken together, these functional capabilities are implemented as shown in the block diagram of Figure 4. At the beginning of each series of trials, data describing initial conditions and test conditions are put onto the "Master Data Base". Data corresponding to that selected by the experimenter (initial conditions and experimental parameters) are transferred from the Master Data Base to the Working Data Base. The real time computation processes are then initiated. Once each 50 milliseconds, analog and discrete control signals are read from the consoles, vehicle and sensor dynamics are computed, and appropriate command values are output to the servos in the simulator. At the end of each trial, time and error data are printed out.

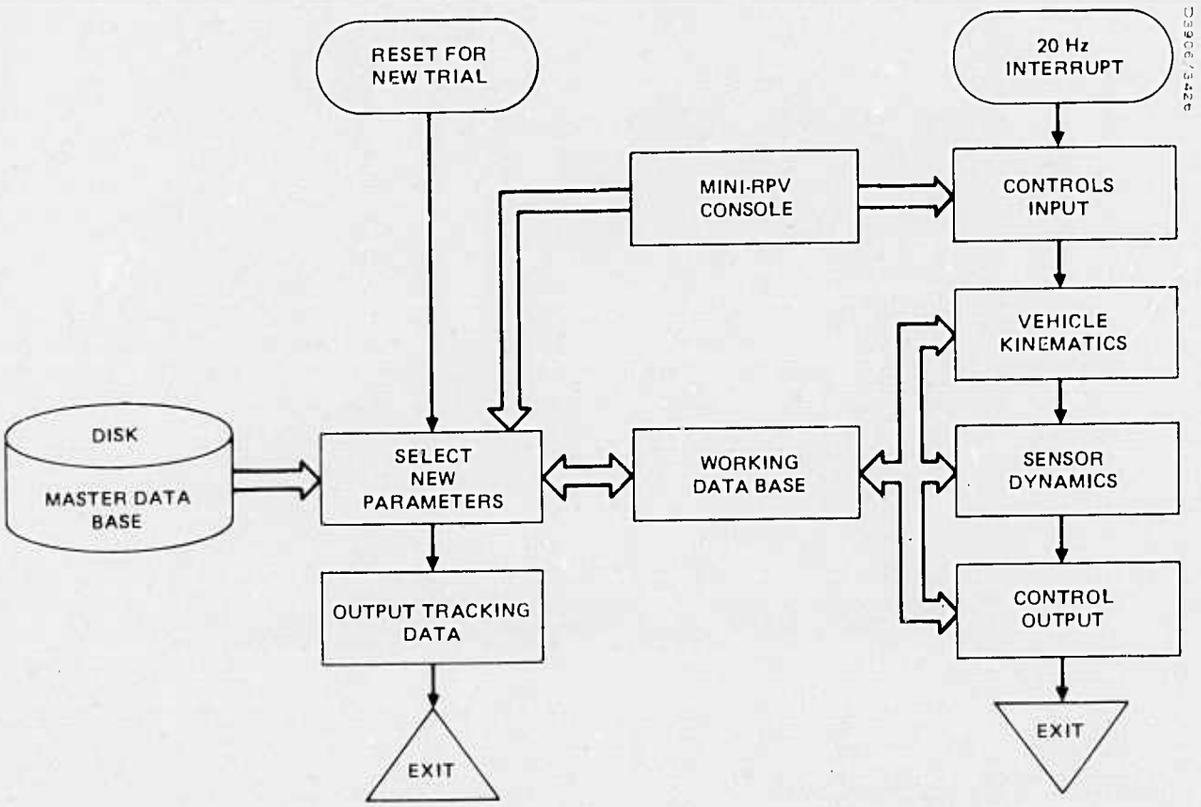


Figure 4. Functional diagram of mini-RPV simulator.

Operator's Control Console

An operator can control simulated RPV vehicle steering in a vehicle steering mode by introducing pitch rate and/or turn commands with a proportional, two-axis displacement hand control. In a ground stabilized sensor steering mode, displacement of the hand control introduces proportional vertical and/or horizontal rate commands to control the pointing angles of a simulated 3-axis gimballed, ground-stabilized sensor. The same hand control may be used in other modes to position a cursor on the display for various designation or tracking tasks. Manual zoom, lock-on, cursor call-up, and freeze commands are also activated by switches on the hand control.

A 14-inch diagonal television monitor is used for the video display. The number of raster lines displayed equals the number scanned on the sensor simulator. The operator's control console is shown in Figure 5.

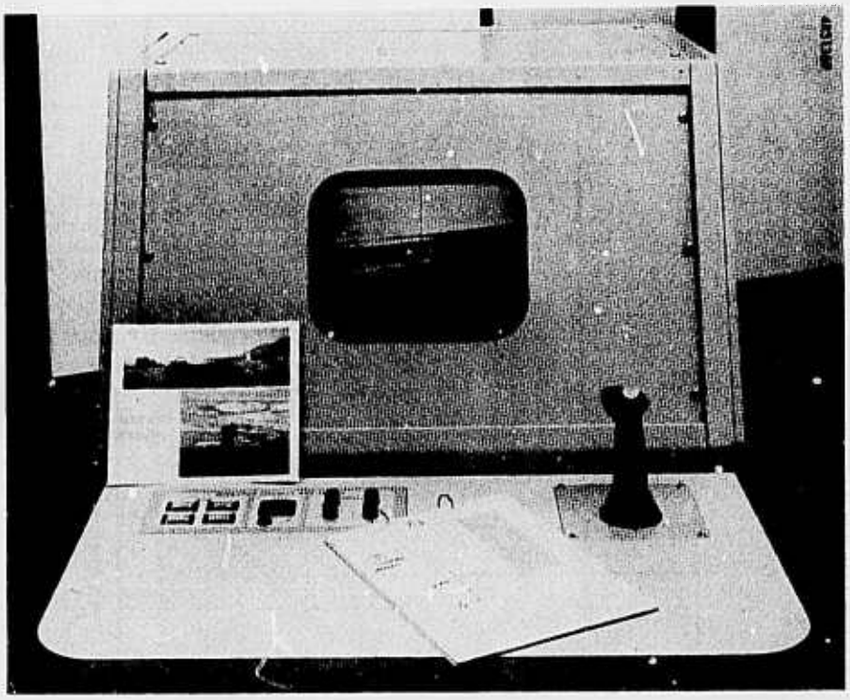


Figure 5. Operator's control console.

Experimenter's Control Console

The controls for the RPV simulator not related to operator functions are on the experimenter's console. They provide for mode and parameter selection and system adjustments. Some of the controls command the computer to initiate automatic operations and adjustments for formal studies. Others are used manually for study checkout and pilot studies. The functions performed by these controls are as follows: simulator mode control, subject code insertion, test and initial condition selection, test run control, parameter selection in the manual mode, manual scoring, data printout control, and cursor positioning and TV scanner adjustment in the check mode. The experimenter's console, shown in Figure 6, is positioned near the operator's console so that the experimenter can monitor the display and operator actions.

The Hughes RPV simulator interfaced with the Naval Undersea Center's cosine/DPCM transform unit provided the vehicle for conducting the two laboratory research investigations that are described in the following sections of this report.

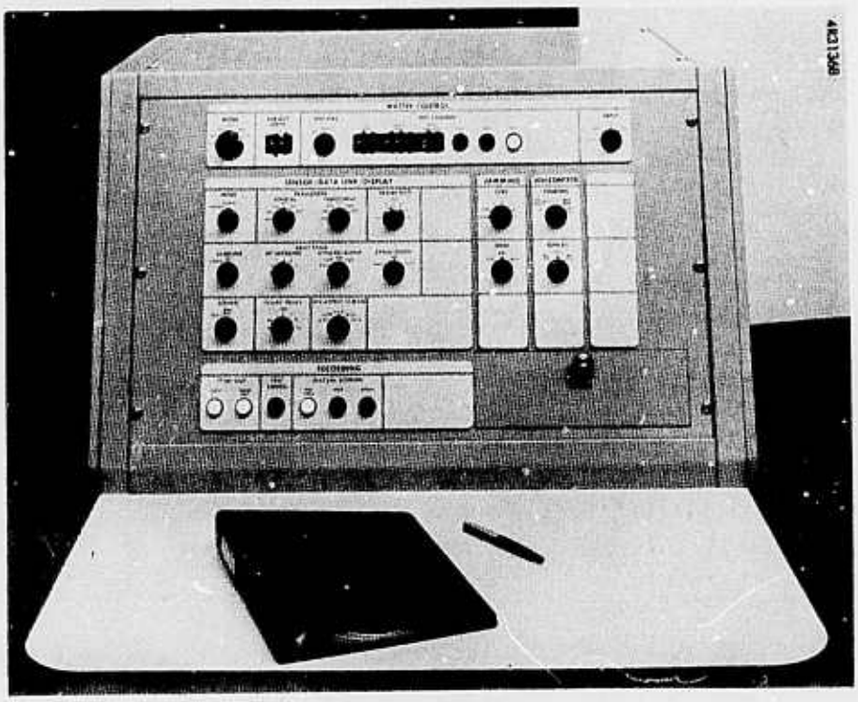


Figure 6. Experimenter's control console.

382

SECTION 3

BANDWIDTH COMPRESSION STUDY

INTRODUCTION

Anti-jam protection of wideband video data links is one of the most dificult problems confronting RPV system designers and can best be achieved by RF bandwidth reduction. Bandwidth compression is one means of reducing video bandwidth by reducing spatial redundancies that exist within a frame of video data.

Bandwidth compression techniques can be categorized into techniques that: 1) process time domain video samples directly to achieve compression and 2) techniques that employ transform domain sampling and truncation of the resulting coefficients. Differential pulse code modulation and delta modulation are well known time domain compression techniques. These techniques are generally limited to compression ratios on the order of 2:1. The transform coding techniques offer much greater potential for bandwidth compression. Transform coding techniques are generally assumed capable of achieving compression ratios as high as 6:1 without serious image degradation. Transform coding systems can be either one- or two-dimensional; examples of transform coding algorithms are the cosine, Haar, Hadamard, and Fourier transforms.

The Naval Undersea Center's transform system is a hybrid system which uses one-dimensional cosine transform coding with differential pulse code modulation. This hybrid system avoids the requirement for a block memory inherent to two-dimensional video transform systems.

Video bandwidth compression techniques have been explored extensively over the past few years. This work has included theoretical studies and computer simulation studies. Evaluation of images produced by simulations of bandwidth compression techniques has been largely qualitative in which an original uncompressed image is subjectively compared to the same image that has undergone bandwidth compression. Little has been done in the way of quantitatively assessing the effects of bandwidth compression on operator task performance. Therefore, one cannot specify the level of bandwidth compression that can be tolerated without degrading operator task

performance below acceptable performance limits. In the case of mini-RPV's, the operator task requirement is recognition of tactical targets (tanks, trucks, SAM sites, and the like) during surveillance and targeting missions.

This laboratory man-in-the-loop study was conducted to investigate the effects of bandwidth compression and bit error rate jamming on operator tactical target recognition performance. The hybrid cosine/DPCM transform system developed by the Naval Undersea Center was used to achieve and vary bandwidth compression and bit error rate jamming. Although the NUC system was limited to 100 by 100 resolution elements, the study was conducted in such a manner so that the results would be independent of the relatively low 100 by 100 elements of the NUC equipment.

RESEARCH PARAMETERS

Bandwidth compression and bit error rate jamming were variable parameters in the study. Compression levels of 1, 2, and 5 bits per picture element (compression ratios of 6:1, 3:1, and 1.2:1, respectively) were investigated. The NUC equipment was capable of 0.5 bit per picture element (pixel) compression; however, the image quality at this level of compression was deemed too poor to include in the study. Bit error rate (BER) jamming levels of zero, 10^{-3}, and 10^{-2} bit errors per sample were investigated in combination with each of the three compression ratios.

In addition to the nine combinations of bandwidth compression and jamming, a special no compression/no jamming condition was investigated. The NUC cosine/DPCM equipment was bypassed for the special condition and the Hughes RPV simulator was set up in a digital mode with 256 by 256 element resolution and 5-bit gray shade encoding. This special condition was included in the study as a baseline against which the 5 bits per pixel compression/zero jamming condition could be compared. The 5 bits per pixel compression condition should result in minimum degradation of operator target recognition performance compared to uncompressed digital video with 5 bits gray shade encoding.

The difference in resolution between the 100 by 100 resolution/5 bits per pixel compression condition and the 256 by 256 resolution/5 bits gray level encoding uncompressed condition was corrected for by using different viewing distances so that picture element visual subtense was equivalent and by using resolution elements per target height as the performance measure.

RESEARCH DESIGN

A 3 x 3 within-subjects factorial design was used to present all combinations of bandwidth compression and bit error rate jamming to nine operators (subjects). Compression was blocked into three groups of six trials each to avoid unrealistic changes in compression ratio from trial to trial. The order of block presentation was counterbalanced across the nine operators. Within a block of six trials, each bit error rate occurred twice, and across the 18 trials that each operator received all six possible orders of the three bit error rates were seen. Further, across the entire design each target occurred equally often in each of the nine combinations of conditions which completely balanced targets, operators, and conditions. The resulting design is shown in Figure 7.

The special 256 by 256 element resolution/5 bit gray level encoding condition was run separately after all the operators had been presented the nine compression/jamming conditions. This was necessary because of the fairly time consuming equipment set-up required to change from cosine/DPCM bandwidth compression to no bandwidth compression. Each operator received two special condition trials on two of the 18 targets. Thus the 18 targets and nine operators were equally represented in the nine compression/jamming conditions and the special condition.

OPERATORS

Nine Hughes engineering personnel served as operators (subjects) in the study. All the operators had served as subjects in previous imaging sensor studies at Hughes.

TARGETS

Tactical, vehicular, HO gauge, target models photographed against a plain background were used in the study. Since the operators' task was recognition of tactical targets, it was deemed desirable to eliminate the variance that would result from background clutter and visual search. The targets, therefore, were presented in a plain background in the center of the operators' display. Nine vehicle targets photographed at two aspect angles provided the 18 target images used in the study. The nine target types were a jeep, a staff car, a half-truck, an open truck, a tow truck, a 5-ton truck, a tank, an APC, and a crane. The two aspect angles were

Operators	Targets																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
O_1	A1	A2	A3	A2	A1	A3	B2	B3	B1	B3	B2	B1	C3	C1	C2	C1	C3	C2
O_2	A2	A3	A1	A3	A2	A1	C3	C1	C2	C1	C3	C2	B1	B2	B3	B2	B1	B3
O_3	A3	A1	A2	A1	A3	A2	B1	B2	B3	B2	B1	B3	C2	C3	C1	C3	C2	C1
O_4	B1	B2	B3	B2	B1	B3	C2	C3	C1	C3	C2	C1	A3	A1	A2	A1	A3	A2
O_5	B2	B3	B1	B3	B2	B1	A3	A1	A2	A1	A3	A2	C1	C2	C3	C2	C1	C3
O_6	B3	B1	B2	B1	B3	B2	C1	C2	C3	C2	C1	C3	A2	A3	A1	A3	A2	A1
O_7	C1	C2	C3	C2	C1	C3	A2	A3	A1	A3	A2	A1	B3	B1	B2	B1	B3	B2
O_8	C2	C3	C1	C3	C2	C1	B3	B1	B2	B1	B3	B2	A1	A2	A3	A2	A1	A3
O_9	C3	C1	C2	C1	C3	C2	A1	A2	A3	A2	A1	A3	B2	B3	B1	B3	B2	B1

Key:

Compression, bits/pixel

A = 5

B = 2

C = 1

Bit Error Rate Jamming

1 = 0

2 = 10^{-3}

3 = 10^{-2}

Figure 7. Research design used in bandwidth compression study.

386

a broadside and a 45-degree quartering angle. The size of the targets was controlled during the photography so that target height was the same for all the targets. This was done so that the operators would not learn to guess any target type because of its overall physical size. Height-to-length target ratios, of course, were not altered.

STUDY PROCEDURE

The operators' task was to recognize a single target (state the type of target, i.e., jeep, tank, APC, 5-ton truck, etc.) presented in the center of the television monitor. A trial began with the display blanked. The experimenter unblanked the display, and the operator saw one of the 18 target images at a size such that 5 TV lines were across the height of the target. The operator was required to state which of the nine target types was being presented, if possible. After the operator made a response, the display was blanked and target size was increased (the experimenter increased the focal length of the zoom lens via a 10-turn potentiometer) so that 7 TV lines were across the height of the target. The display was then unblanked and again the operator was asked to state which of the nine targets was being presented. This procedure was repeated at steps of 9, 11, 14, 17, 21, 25, 30, 35, 40, 45, 50, and 55 TV line across the height of the target. Each target presentation was timed and a response was requested if the operator had not made a response after 20 seconds.

Prior to initiation of formal data collection trials, standardized written instructions were given to the operators to read; additional clarifications to the instructions were provided as needed. The operators were then trained to recognize the tactical target types using a two-choice paradigm. Differentiating between similar targets was emphasized. The operators were told to ignore target size (height) as a cue, because the targets would be presented at the same size on the display. They were also told that the targets could occur in any orientation between 90 and 270 degrees (90 degrees being a broadside view, heading to the operator's right). Data collection began immediately following the training. A rest pause was given midway in the single session which lasted approximately 70 minutes.

As stated previously, data collection for the nine bandwidth compression/jamming conditions was conducted first for all nine operators. After this major part of the study had been completed, the cosine/DPCM system was

disconnected from the RPV simulator and the special 256 by 256 element resolution/5 bit gray level encoding condition was set up. Each of the nine operators was then brought back, and each received two trials. The procedure for the special condition was the same as described above except target order was reversed and the step increments in TV lines on the targets were smaller. The steps were 4, 5, 6, 7, 9, 11, 13, 15, 18, 21, 24, 28, 32 and 36 TV lines. During both the training and data collection trials, HO gauge models of the nine targets were situated on the display console desk surface for use by the operators.

In the study, a picture element subtended 3.6 arcminutes for both the 100 by 100 element resolution cosine/DPCM conditions and the 256 by 256 element resolution/5 bit gray level encoding condition. Viewing distances of 64 and 25 inches were used to maintain the constant 3.6 arcminutes picture element visual subtense for the 100 by 100 and 256 by 256 element resolutions.

PERFORMANCE MEASURE

The number of TV lines across the height of the targets at recognition was used to compare operator target recognition performance differences caused by bandwidth compression ratios and levels of bit error rate jamming. Target recognition was defined as the first of three consecutive correct target recognition responses. Thus, the number of TV lines at which a correct recognition response occurred and was followed by two consecutive correct recognition responses at the next two TV line step increments was used as the performance measure for a trial.

RESULTS AND DISCUSSION

The TV lines across target at recognition data were analyzed for the reliability of bandwidth compression and bit error rate jamming effects with analysis of variance. Two analyses of variance were computed – one to test the effects of the three compression levels and the three jamming levels, and one to test the three compression levels and the special no compression condition at zero jamming. Analyses of variance summary tables are contained in the appendix to this report.

In both analyses, the effects of bandwidth compression (bits per picture element) were found to be statistically reliable at the 0.001 probability level. Thus the chance that the obtained differences among the levels of compression investigated could have occurred by chance was less than one out of one thousand. Bit error rate jamming had no effect on the operators' target recognition performance. The probability that the differences obtained among the zero, 10^{-3}, and 10^{-2} bit error rate jamming levels could occur by chance was greater than 0.25.

Figure 8 shows the number of TV lines across the height of tactical targets at recognition for the 1, 2, and 5 bits per pixel compressions and the special no compression condition. It is obvious from Figure 8 that there were no performance differences among 2 bits per pixel, 5 bits per pixel, and the special condition. The 1 bit per pixel compression condition, however, required appreciably more TV lines across targets than did the other three conditions. The 2 and 3 bits per pixel compression condition and special no compression condition required 8 to 9 TV lines across targets

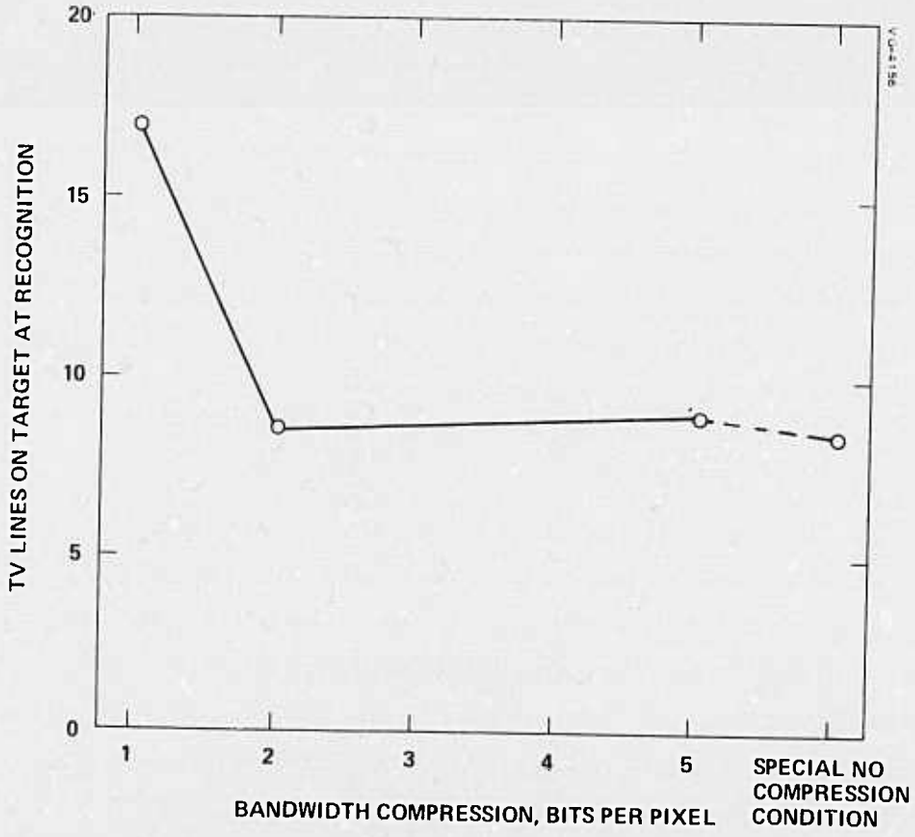


Figure 8. Effects of bandwidth compression on operator tactical target recognition performance.

for recognition to occur; the 1 bit per pixel condition required 17 TV lines — a 100 percent difference. A Newman-Keuls simultaneous significance test for multiple contrasts verified that the performance differences among the 2 and 5 bit per pixel conditions and the special condition were not statistically reliable ($p > 0.05$), and that the 1 bit per pixel condition was significantly poorer than the other three conditions ($p < 0.01$). The results of the study thus clearly indicate that a severe performance penalty is paid going from 3:1 to 6:1 compression.

Figure 9 shows the lack of effect that bit error rate (BER) jamming had on operator tactical target recognition performance. There was a slight increase in the number of TV lines (1.5 TV lines) required for target recognition to occur at 10^{-2} BER jamming. This increase was not, however, a statistically reliable effect; hence, it cannot be considered a real effect. It can, therefore, be stated that jamming effects on video data transmitted in the digital transform domain at levels as high as 10^{-2} bit error rate will not cause any degradation of an RPV operator's ability to recognize tactical targets.

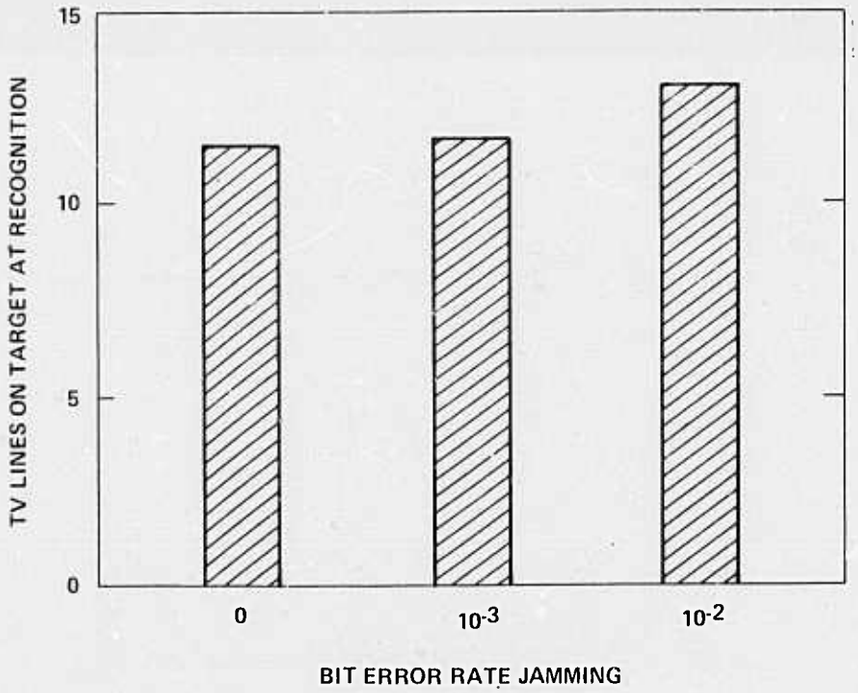


Figure 9. Effects of bit error rate noise jamming on operator tactical target recognition performance.

390

CONCLUSIONS AND RECOMMENDATIONS

The results of this study showed no operator performance decrement as bandwidth compression was increased from 1:1 to 3:1 and a major decrement between 3:1 and 6:1 compression. Since, human behavior is normally best described by a smooth linear or second-order function, compressions greater than 3:1 but less than 6:1 should result in degraded but acceptable performance. Figure 10 depicts our estimated operator performance function for bandwidth compression. Thus 4:1 compression would only cause an increase of 1 TV line over 3:1 compression, and 5:1 compression would cause an increase of 4 TV lines over 3:1 compression. From this, one could recommend a 4:1 (1.5 bits per pixel) compression for mini-RPV systems that would not result in any appreciable degradation of operator tactical target recognition performance compared to uncompressed sensor video.

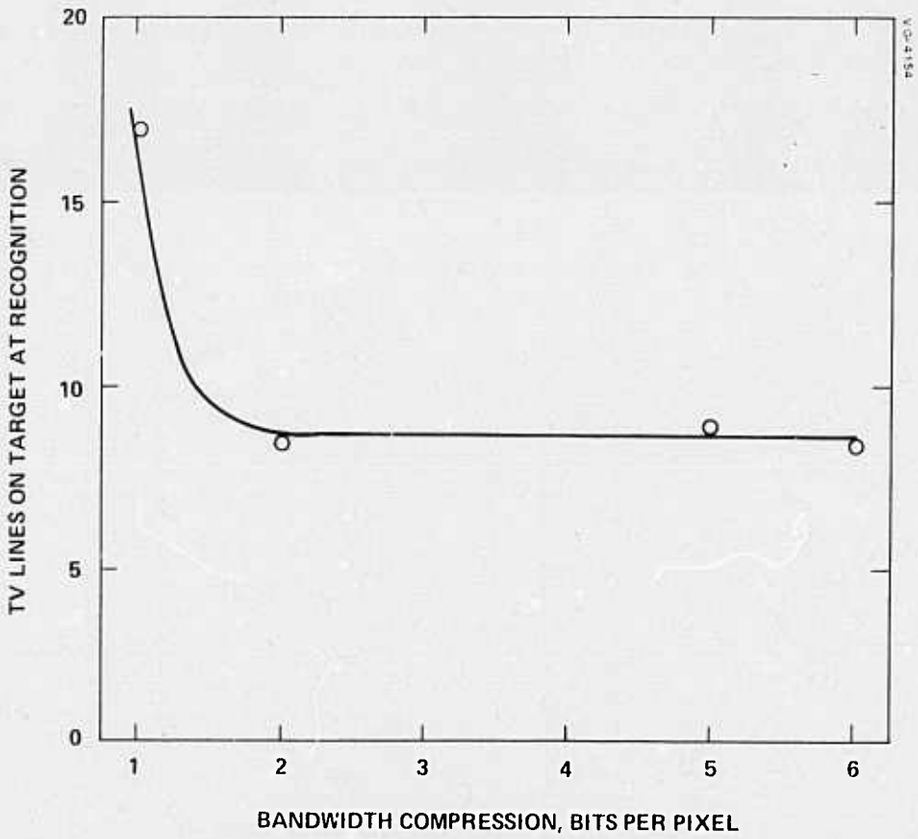


Figure 10. Estimated operator performance/ bandwidth compression function.

391

The results of the study showed that there was no difference between uncompressed TV video and 3:1 compressed video and that 8 to 9 TV lines across the height of tactical targets were required for recognition at compressions from 1:1 to 3:1. This latter finding is in close agreement with Johnson's (1958) work on required resolution per minimum target dimension. Johnson used tactical targets in a plain background and found that on average 8 TV lines were required to recognize tactical targets. It can therefore be concluded that the results of the bandwidth compression study are reliable for use in the specification of system design requirements for mini-RPVs.

Required resolution per minimum target dimension has been shown to be a major determinant of operator tactical target recognition performance. In RPVs, resolution lines across targets will be determined by the resolution of the sensor, the field of view of the sensor, and the range to the target. Given a performance criterion for the required number of resolution lines across targets (e.g., 8 TV lines at 3:1 compression) sensor resolution, sensor field of view, and range-to-target can be traded off. Similarly, the performance criterion can be changed (e.g., 18 TV lines) to accommodate a 6:1 compression system and tradeoffs among resolution, field of view, and range-to-target can be made.

Table 1 gives the number of TV lines that would be on a 3-meter high tank at sensor resolutions of 240 and 480 active TV lines, sensor fields of view of 0.5, 1, 2, 5, and 10 degrees, and ranges-to-target of 1, 2, and 5 kilometers. The areas outlined in bold lines indicate the conditions which would result in operator target recognition with a 4:1 compression system (1.5 bits per pixel) where 8 to 9 TV lines across the target's height are required for recognition. If a 6:1 compression system (1 bit per pixel) were used, 17 or more TV lines are required for target recognition. The sensor resolution, sensor field of view, and range to target conditions that satisfy this criterion are indicated by the area outlined in bold outlining in Table 2.

For example, if a 256 by 256 resolution sensor is to be used, the desired range-to-target at recognition is 5 kilometers, and 1.5 bits per pixel provides sufficient anti-jam margin, a field of view of 1.0 degree or smaller is required. If 1 bit per pixel compression with a 256 by 256 resolution system is required to obtain the required anti-jam margin, a 0.5 degree or smaller field of view will be required to recognize targets at a 5 kilometer range.

TABLE 1. NUMBER OF TV LINES ACROSS 3-METER HIGH TANK

Range-To-Target, Kilometers	Resolution, Active Elements									
	240					480				
	Field of View, Degrees					Field of View, Degrees				
	0.5	1	2	5	10	0.5	1	2	5	10
1	84	42	21	8.4	4.2	168	84	42	17	8.4
2	42	21	10.5	4.2	2.1	84	42	21	8.4	4.2
5	17	8.4	4.2	1.7	0.84	34	17	8.4	3.4	1.7

TABLE 2. NUMBER OF TV LINES ACROSS 3-METER HIGH TANK

Range-To-Target, Kilometers	Resolution, Active Elements									
	240					480				
	Field of View, Degrees					Field of View, Degrees				
	0.5	1	2	5	10	0.5	1	2	5	10
1	84	42	21	8.4	4.2	168	84	42	17	8.4
2	42	21	10.5	4.2	2.1	84	42	21	8.4	4.2
5	17	8.4	4.2	1.7	0.84	34	17	8.4	3.4	1.7

From the results of the bandwidth compression study and the type of data given in Tables 1 and 2, the mini-RPV system designer knows the design latitude he has to work with and can make realistic tradeoffs among sensor resolution, sensor field of view, and bandwidth compression to achieve the required operator performance at the desired range-to-target.

SECTION 4

FRAME RATE STUDY

INTRODUCTION

Frame rate reduction offers large potential for reducing video bandwidth. Whereas a bandwidth compression of 4:1 was found to be acceptable in the bandwidth compression study for the mini-RPV tactical target surveillance mission, bandwidth reduction of 8:1 or greater may be achievable with frame rate reduction.

Reduction of frame rate below the standard 30 frames per second TV rate may affect the RPV operator's target recognition performance and his ability to point the sensor for target designation and tracking. Both operator target recognition and designation/tracking performance could be degraded at low frame rates because of the increased time delay between successive frames of video. Some limited work has been performed in an attempt to understand the effects of frame rate reduction on target recognition and target designation performance.

A study conducted by the Air Force Aeromedical Laboratory (Self and Heckart, 1973) investigated the effects of 1, 3, 8, and 24 frames per second frame rates on RPV operator target acquisition performance. Although the operators' task required control of sensor pointing, the control update rate was always 30 frames per second. Thus, only the effect of frame rate on operator target recognition performance was assessed. The results of the Self and Heckart (1973) study, shown in Figure 11, indicate a slight trend for operator performance (range-to-target at acquisition) to improve as frame rate increased from 1 to 8 frames per second. The differences among the four frame rates were not, however, statistically reliable. It was therefore concluded that frame rates as low as 1 frame per second do not degrade RPV operator target recognition performance.

Hillman (1967) in a literature review of Human Factors Considerations in Real-Time Airborne Target Acquisition reviewed four papers which investigated the effects of frame rate on target designation performance. The findings of the four investigations indicate the lowest acceptable frame rate to be

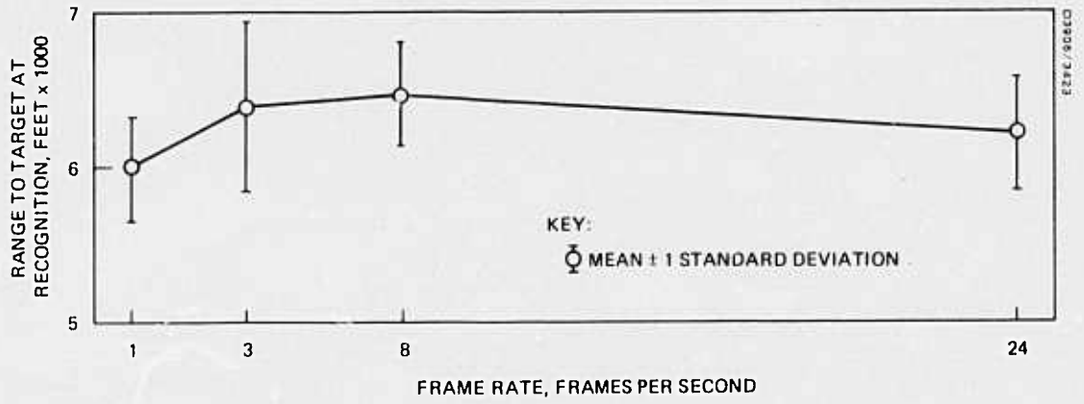


Figure 11. Effect of video frame rate on operator target recognition performance.

between 2 and 7.5 frames per second. These findings indicate a bandwidth reduction potential somewhere between 4:1 and 15:1.

This past research indicates that frame rates as low as 1-frame per second have little effect on operator target recognition performance but for target designation the lower bound for acceptable operator performance is between 2 and 7.5 frames per second. Since the past target designation research was not directed to the mini-RPV mission and because of the discrepancies in the lowest acceptable frame rate among the different investigators, this study was conducted to determine the effects of frame rate on operator target designation and tracking performance for the mini-RPV targeting mission.

RESEARCH PARAMETER

Video frame rate was the single parameter investigated in this study. Six values of frame rate – 0.94, 1.88, 3.75, 7.5, 15.0, and 30.0 frames per second – were investigated.

RESEARCH DESIGN

A within-subjects design was used with six subjects (operators). Each operator received all six frame rates in which frame rates were counterbalanced across operators using a latin-square technique. Five replications of each frame rate/operator combination were run. The first replication served as a practice/warm-up trial and was not used in the data analyses.

Thus, each operator received six blocks (frame rates) of five trials (replications) in which the last four replications in each block provided the data to determine the effects of frame rate on target designation and target tracking performance. Figure 12 shows the research design used in the study.

OPERATORS

The six operators used in the study were engineering personnel within the Display Systems and Human Factors Department of Hughes Aircraft Company. These personnel were selected based on their past experience and demonstrated performance in target designation and tracking tasks.

TARGET SCENE

A single target scene, Stalin tank in a plain background, was used in the study. The tank was scaled to represent a 3-kilometer slant range with a 2-degree sensor field of view. The tracking point was the centroid of the tank. The physical location of the tank centroid was pointed out to the six operators. The tank, as it appeared to the operators during the trials, is shown in Figure 13.

OPERATORS	FRAME RATE, FRAMES PER SECOND					
1	0.94*	30	1.88	15	3.75	7.5
2	1.88	0.94	3.75	30	7.5	15
3	3.75	1.88	7.5	0.94	15	30
4	7.5	3.75	15	1.88	30	0.94
5	15	7.5	30	3.75	0.94	1.88
6	30	15	0.94	7.5	1.88	3.75

*Each cell represents a block of five trials.

Figure 12. Research design used in frame rate study.

THE TRACKING TASK

Figure 13. Tank target used in the study.

At the start of each trial, the tank target was positioned 41 meters from the center of the display in a random direction and drifting at 50 meters per second. The drift rate was due to the simulated vehicle/target geometry in which the RPV was flying at a 762-meter altitude, a speed of 50 meters per second, and a ground range-to-target of 3000 meters. The sensor field of view was 2 degrees. These mission parameters were obtained from an Army Electornics Command, Fort Monmouth, New Jersey letter on mini-RPV mission scenarios from A. Zarin, Chief Data Transmission, Process and Display Technical Area, dated 23 July 1975.

At the start of a trial, the 2-degree field of view represented a ground coverage of 108 meters. The 41-meter initial target offset position represented 38 percent of the field of view or a displacement of 3.8 inches from the center of the 14-inch diagonal display. The forcing function the operators were required to track was a tangent function in which target motion rate increased with time (as range-to-target decreased).

The operator's task was to position the target aimpoint on a set of fixed crosshairs on the display as quickly as possible, depress a pushbutton (for measurement of designation accuracy), and track the target (maintain minimum displacement between the target aimpoint and the crosshairs) for 35 seconds.

A two-axis position displacement hand control was used by the operators to input sensor rate commands to a simulated 3-axis stabilized sensor. Vertical stick deflection commanded sensor elevation rate slewing; horizontal stick deflection commanded sensor azimuth rate slewing. Hand control pitch rate commands appeared as vertical picture translation rates on the TV monitor, and hand control azimuth rate commands appeared as horizontal picture translation rates. Figure 14 shows sensor slew rate per amount of hand control deflection. This X^2 function was the same for all four quadrants of hand control deflection.

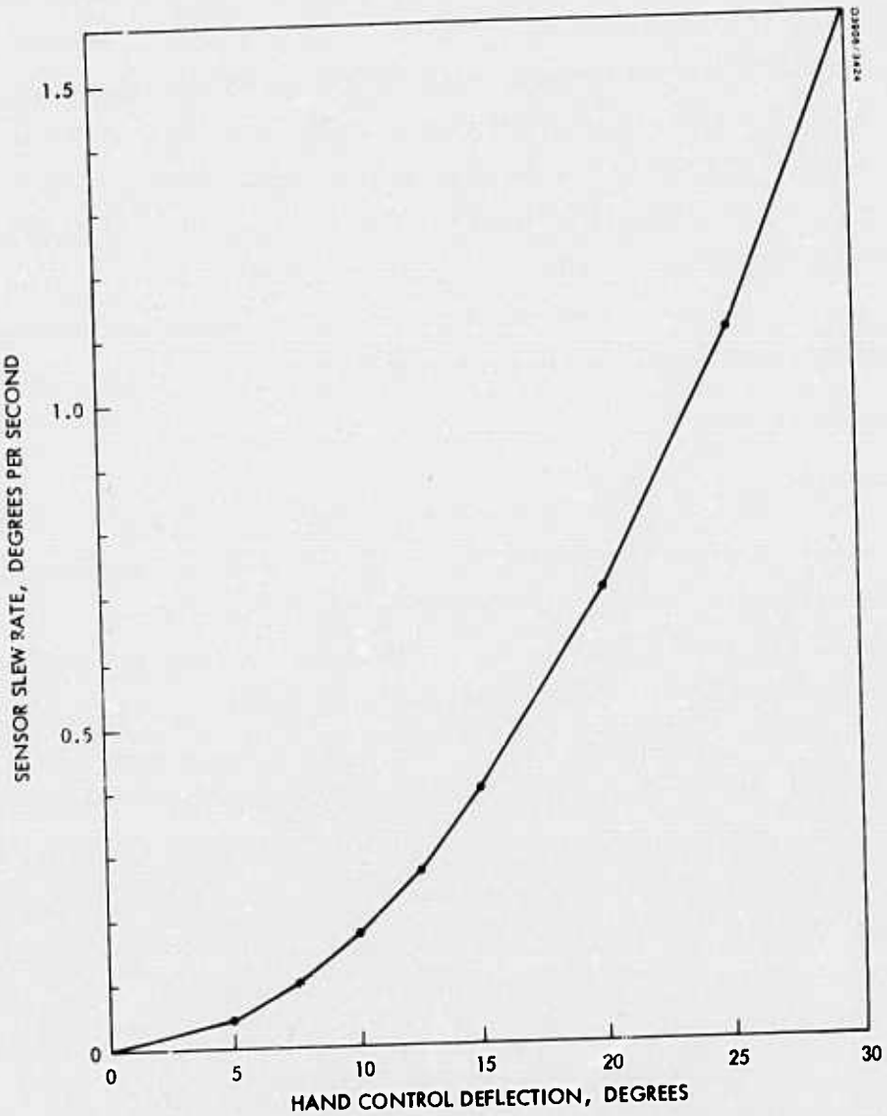


Figure 14. Hand control shaping function.

LABORATORY EQUIPMENT

The Hughes RPV simulation facility, described in Section 2, was used in conjunction with a general purpose RPV simulation computer program. This facility consists of a Xerox Sigma 5 digital computer and associated software, a television scanner and transport mechanism, a digital refresh memory, an experimenter's control console, an operator's console, and associated interface electronics. This system provided realistic control of flight dynamics within the limits of the programmable vehicle characteristics and the means of varying frame rates and other transmission variables. The reduced frame rates were produced by updating the digital refresh memory

at the reduced rate while flicker free television was provided by reading out of the memory at a standard 60 Hz rate. The desired experimental conditions were selected from the experimenter's control console and then computer generated. The target was displayed through use of the television scanner and was driven by the transport platform servos. The target motion drive signals were from the computer. The operator's control actions were relayed by the computer to the transport platform. The target aimpoint location was stored in the computer so that it could determine operator target designation and tracking error and print them out at the end of each trial.

STUDY PROCEDURE

When each operator entered the laboratory, he was given a set of standardized written instructions to read. Any requested clarifications to the instructions were then given. Prior to formal data collection, extended training was given to familiarize the operators with the tracking task using the various frame rates. Each operator received training on all frame rates. Initially the operators received the 30 frame per second frame rate and were allowed to learn the sense of the hand control (relationship between hand control displacement and picture motion). The operators could choose between a fly-to sense (move hand control towards the target) and a conventional flight control sense (move hand control toward target in azimuth and away from target in elevation to bring target towards the center of the display). After the operators had selected and learned the hand control sense, extensive training was given at each of the six frame rates.

Each trial started with the display blanked. When the test conductor had selected the appropriate test conditions, he depressed the computer "RUN" pushbutton. The computer selected the required conditions, unblanked the display, and started the trial. The operator saw the target displayed 41 meters (3.8 inches) from the crosshair in the center of the display. The operator positioned the target aimpoint (slewed the sensor) until the target aimpoint was positioned on the crosshair and depressed a pushbutton on the hand control. During this part of the trial, the target moved at a fixed 50 meter per second rate. When the operator depressed the pushbutton, designation error was measured by the computer and the simulation of range closure to the target was initiated, resulting in the tangent function tracking

rate. The operator was then required to track the target aimpoint for 35 seconds. At the end of 35 seconds, the trial was terminated and RMS error over the last 30 seconds of the trial was computed.

The use of the 50 meter per second fixed rate at the beginning of each trial and the introduction of range closure with increased tracking rate in the latter part of each trial was done because of the desire to measure both single target designation error and target tracking error. It was not feasible to initiate range closure at the start of the trials, because the time required to make the single designation at the low frame rates would have resulted in the tracking rate going to infinity before the 35 second tracking task started.

PERFORMANCE MEASURES

Time to target designation, designation error, and RMS tracking error were measured on each trial. Time, in seconds, was measured from the start of a trial till the operator positioned the target aimpoint on the display crosshairs and depressed the designation pushbutton on the hand control. Radial designation error in milliradians was measured at the time the operators depressed the pushbutton on the hand control. Radial RMS tracking error in milliradians was measured during the last 30 seconds of the 35 second tracking run. The tracking run started as soon as the operators had made their designation and depressed the pushbutton on the hand control.

RESULTS AND DISCUSSION

The time and error data resulting from the study were plotted as a function of frame rate and subjected to analyses of variance to test for the reliability of the effects of frame rate on the performance measures. The probability that the obtained differences among frame rates could have occurred by chance was less than 0.001 for designation time and designation accuracy and less than 0.01 for RMS tracking error.

Designation Time

The effect of frame rate on target designation time is shown in Figure 15. Designation time ranged from 16 seconds at the 30 frames per second frame rate to 61 seconds at the 0.94 frame per second frame rate – nearly a four times increase in the time required to make a single precision designation

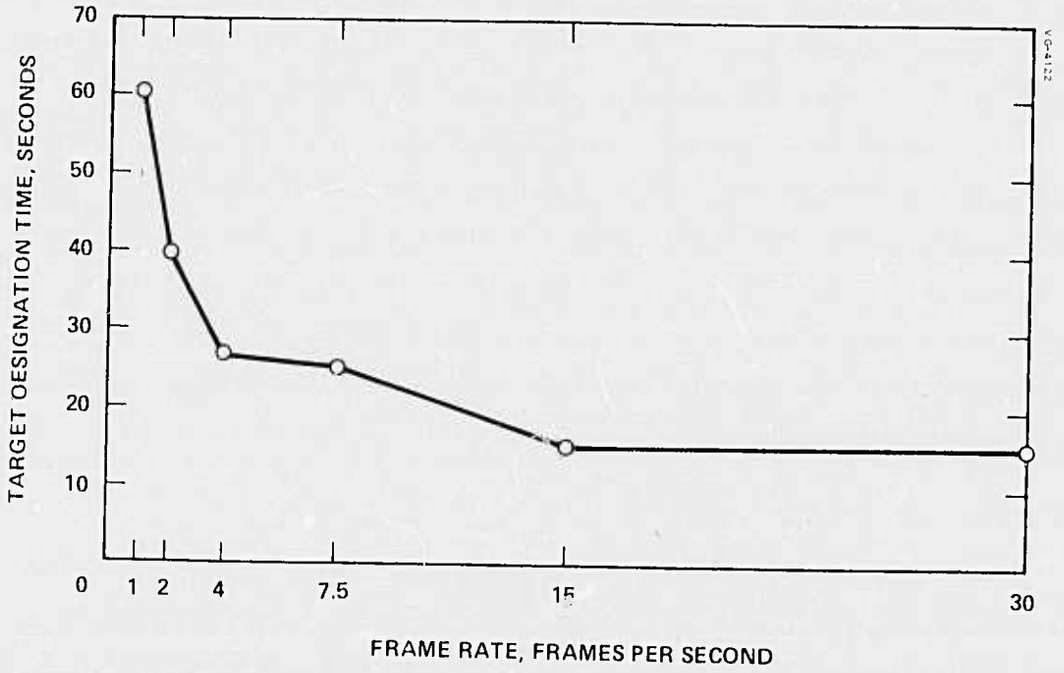


Figure 15. Effects of frame rate on target designation time.

of the tank target. The largest reduction in time as frame rate increased was between 0.94 and 3.75 frames per second (33 seconds). Between 3.75 and 15 frames per second, the reduction in designation time was more gradual (13 seconds). There was no difference between 15 and 30 frames per second frame rates.

This study was principally concerned with the effects of frame rate on precision target designation and tracking. The instructions to the operators stressed accuracy – not time. Therefore the time data are not necessarily indicative of the time it would take an operator to make a precision target designation in the real-world situation. The time data are indicative of the task difficulty with the various frame rates and thus provide an index of the relative time required to make a precision target designation for frame rates from 0.94 to 30 frames per second. Clearly, 0.94 and 1.88 frame per second frame rates are extremely difficult to use, 3.75 and 7.5 frames per second frame rates are of moderate difficulty, and 15 and 30 frames per second frame rates can be used with relative ease.

Designation Error

Figure 16 shows the effect of frame rate on radial designation error. Designation error ranged from 4.46 milliradians at the 0.94 frame per second frame rate to 0.82 milliradian at the 30 frames per second frame rate — a greater than 5 to 1 difference. It is obvious from Figure 16 that the largest improvement in designation error occurred as frame rate increased from 0.94 to 1.88 frames per second (from 4.56 to 1.56 milliradians designation error). There was a more gradual improvement in designation error as frame rate increased from 1.88 to 7.5 frames per second (1.56 to 0.88 milliradians designation error). Target designation error was essentially constant from 7.5 to 30 frames per second (0.88 to 0.82 milliradian designation error). A Newman-Keuls simultaneous test for multiple contrasts was used to test for the reliability of differences among each of the pairs of frame rates. The results of this test showed that the 0.94 frame per second frame rate produced significantly ($p < 0.01$) greater designation error than the other five frame rates. Differences among the 1.88, 3.75, 7.5, 15, and 30 frame per second frame rates, however, were not statistically reliable. One can therefore conclude that any frame rate of 1.88 frames per second or greater will result in equivalent operator precision target designation performance. With a larger data sample and more operator training, it might be possible to tease out reliable differences between frame rates from 1.88 to 7.5 frames per second.

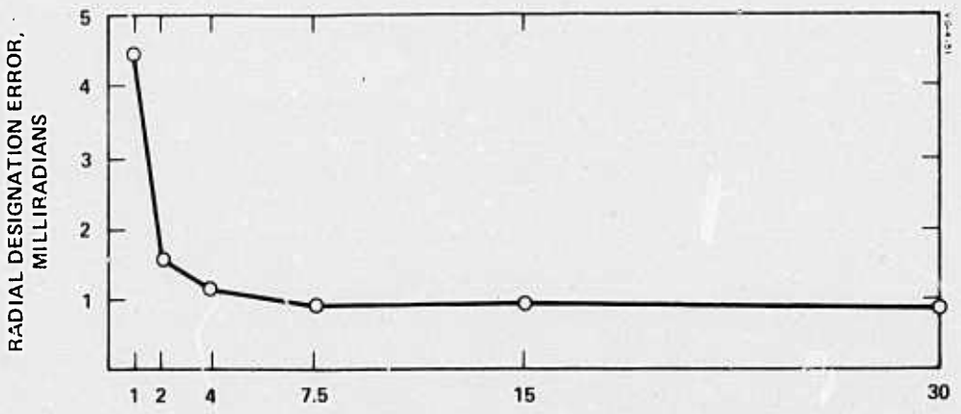


Figure 16. Effects of frame rate on precision target designation accuracy.

402

RMS Tracking Error

The effect of frame rate on RMS tracking error, as shown in Figure 17, reveals the same general results that were obtained for designation error. Tracking error decreased rapidly as frame rate increased from 0.94 to 3.75 frames per second (10.4 to 1.5 milliradians), a small decrease in tracking error was observed as frame rate increased from 3.75 to 15 frames per second (1.5 to 0.77 milliradians), and tracking error remained essentially constant between 15 and 30 frames per second (0.77 to 0.70 milliradian).

Although there was a two-to-one decrease in tracking error as frame rate increased from 3.75 to 15 frames per second, this difference was not statistically reliable ($p > 0.05$). Thus a major improvement in tracking accuracy occurred as frame rate increased from 0.94 to 3.75 frames per second; increasing frame rate greater than 3.75 frames per second resulted in little if any improvement in operator tracking performance.

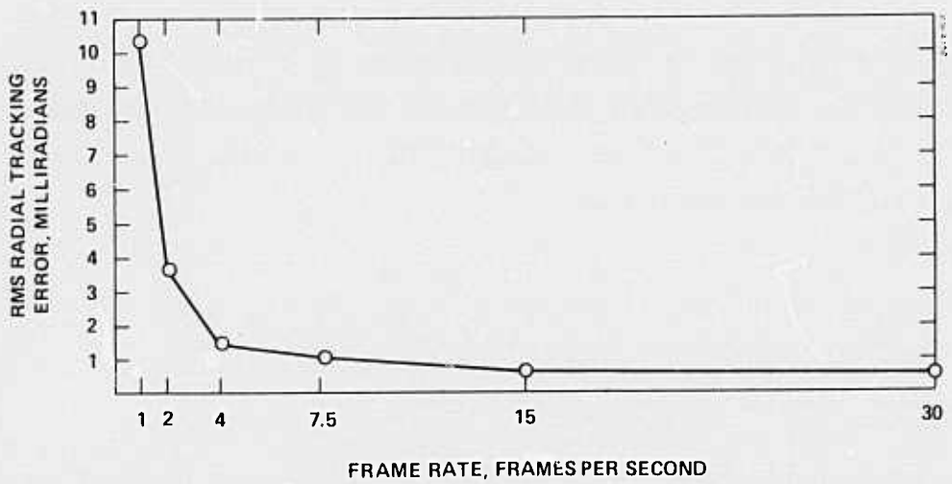


Figure 17. Frame rate effects on target tracking accuracy.

403

CONCLUSIONS AND RECOMMENDATIONS

The three performance measures; designation time, designation error, and tracking error; all showed the same general effect of frame rate on operator performance. Performance improved rapidly as frame rate increased from 0.94 to 3.75 frames per second. Increasing frame rate from 3.75 to 15 frames per second provided a slight improvement in operator performance. Increasing frame rate beyond 15 frames per second did not improve the operator's performance for the mini-RPV targeting task.

Based on the small performance differences among the 3.75, 7.5, 15, and 30 frames per second frame rates and the fact that the differences obtained among these frame rates were not statistically reliable, it can be concluded that there is little, if any, performance advantage to recommend a frame rate greater than 3.75 frames per second for mini-RPV applications. The findings of this study, therefore, indicate that an 8:1 bandwidth reduction can be achieved through frame rate reduction compared to a standard 30 frame per second video frame rate.

In this study, 3-axis stabilized sensor pointing was simulated. There has been some consideration by mini-RPV designers of a caged (unstabilized) sensor. The danger of using an unstabilized sensor is, of course, atmospheric turbulence which can cause large, unpredictable movement of the displayed sensor image. The results of this study are applicable to a caged sensor mini-RPV, assuming that atmospheric turbulence does not cause large mini-RPV disturbances. This conclusion is based on the results of a Hughes funded study which showed there was no difference in operator's designation accuracy with 3-axis stabilized and caged sensor modes with frame rates from 1 to 30 frames per second for a BGM-34 class of RPV with an attitude hold autopilot flying in low turbulence weather. The operator's ability to designate and track targets with a caged sensor in moderate and large turbulent weather has not, to our knowledge, been investigated at any frame rate. The issue of caged sensor and reduced frame rate in a turbulent environment should be investigated before mini-RPV designers select a caged sensor reduced frame rate design. We would not recommend generalizing from the results of the present study or the Hughes funded study for a caged sensor design because turbulence was, in essence, not a factor.

REFERENCES

Hillman, B. Human Factors Considerations in Real-Time Target Acquisition, in Air-to-Surface Missile Technology, 1975-1980, Vol. IV. Supplementary Reports, Institute for Defense Analyses, Science and Technology Division, IDA Log. No. HQ 67-7197, December 1967.

Johnson, J. Analysis of Image Forming Systems, in Image Intensifier Symposium, Fort Belvoir, Virginia, October 6 and 7, 1958. (AD 220160).

Self, H.C. and Heckart, S.A. TV Target Acquisition at Various Frame Rates. Technical Report AMRL-TR-73-111, Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Dayton, Ohio, September 1973.

APPENDIX A

ANALYSIS OF VARIANCE SUMMARY TABLES

406

TABLE A-1. ANALYSIS OF VARIANCE FOR BANDWIDTH COMPRESSION STUDY

SOURCE OF VARIANCE	ERROR TERM	F-RATIO	SUM OF SQUARES	DEGREES OF FREEDOM	MEAN SQUARE	PROBABILITY
Bandwidth Compression (B)	BS	11.6706	1748.677	2	874.3384	<0.001
Noise Jamming (N)	NS	0.8106	72.14322	2	36.07161	>0.25
Trials (T)	TS	12.0515	495.4209	1	496.4209	<0.001
Subjects (S)			734.7842	8	91.84802	
BN	BNS	0.3034	90.59970	4	22.64992	>0.25
BT	BTS	0.3528	110.1897	2	55.09485	>0.25
NT	NTS	0.3032	27.10376	2	13.55188	>0.25
BS			1198.687	16	74.91791	
NS			711.9753	16	44.49846	
TS			328.8699	8	41.10873	
BNT	BNTS	0.4490	153.8806	4	38.47015	>0.25
BNS			2388.854	32	74.65167	
BTS			2498.328	16	156.1455	
NTS			715.1113	16	44.69446	
BNTS			2741.635	32	85.67609	

407

TABLE A-2. ANALYSIS OF VARIANCE FOR BANDWIDTH COMPRESSION STUDY — NO NOISE JAMMING CONDITIONS

SOURCE OF VARIANCE	ERROR TERM	F-RATIO	SUM OF SQUARES	DEGREES OF FREEDOM	MEAN SQUARE	PROBABILITY
Bandwidth Compression (B)	BS	6.2647	956.3608	3	318.7869	<0.001
Trials	TS	4.8832	151.3761	1	151.3761	0.05 < 0.10
Subjects (S)			160.4351	8	20.05438	
BT	BTS	1.1182	286.6829	3	95.56095	>0.25
BS			1221.267	24	50.88611	
TS			247.9934	8	30.99918	
BTS			2050.992	24	85.45801	

405

408

TABLE A-3. ANALYSIS OF VARIANCE FOR TARGET DESIGNATION TIME

SOURCE OF VARIANCE	ERROR TERM	F-RATIO	SUM OF SQUARES	DEGREES OF FREEDOM	MEAN SQUARE	PROBABILITY
Frame Rate (F)	FS	7.8664	36309.71	5	7261.941	<0.001
Subjects (S)			34720.56	5	6944.109	
Trials (T)	ST	1.4417	1184.975	3	394.9917	>0.25
FS			23078.85	25	923.1538	
FT	FST	0.7467	4588.121	15	305.8745	>0.25
ST			4109.543	15	273.9695	
FST			30720.95	75	409.6125	

406
404

TABLE A-4. ANALYSIS OF VARIANCE FOR TARGET DESIGNATION ERROR

SOURCE OF VARIANCE	ERROR TERM	F-RATIO	SUM OF SQUARES	DEGREES OF FREEDOM	MEAN SQUARE	PROBABILITY
Frame Rate (F)	FS	11.1612	239.0675	5	47.81351	<0.001
Subject(S)			58.14612	5	11.62922	
Trials (T)	ST	1.6736	14.80952	3	4.936505	>0.25
FS			107.0974	25	4.283895	
FT	FST	0.8604	42.16078	15	2.810719	>0.25
ST			44.24409	15	2.949606	
FST			245.0066	75	3.266755	

407

H10

TABLE A-5. ANALYSIS OF VARIANCE FOR RMS TARGET TRACKING ERROR

SOURCE OF VARIANCE	ERROR TERM	F-RATIO	SUM OF SQUARES	DEGREES OF FREEDOM	MEAN SQUARE	PROBABILITY
Frame Rate (F)	FS	5.3103	1718.426	5	343.6851	0.001 < 0.01
Subjects (S)			516.5952	5	103.3190	
Trials (T)	ST	1.9787	189.6862	3	63.22873	0.10 < 0.20
FS			1618.020	25	64.72079	
FT	FST	2.4076	920.5735	15	61.37155	0.001 < 0.01
ST			479.329	15	31.95526	
FST			1911.796	75	25.49060	

114