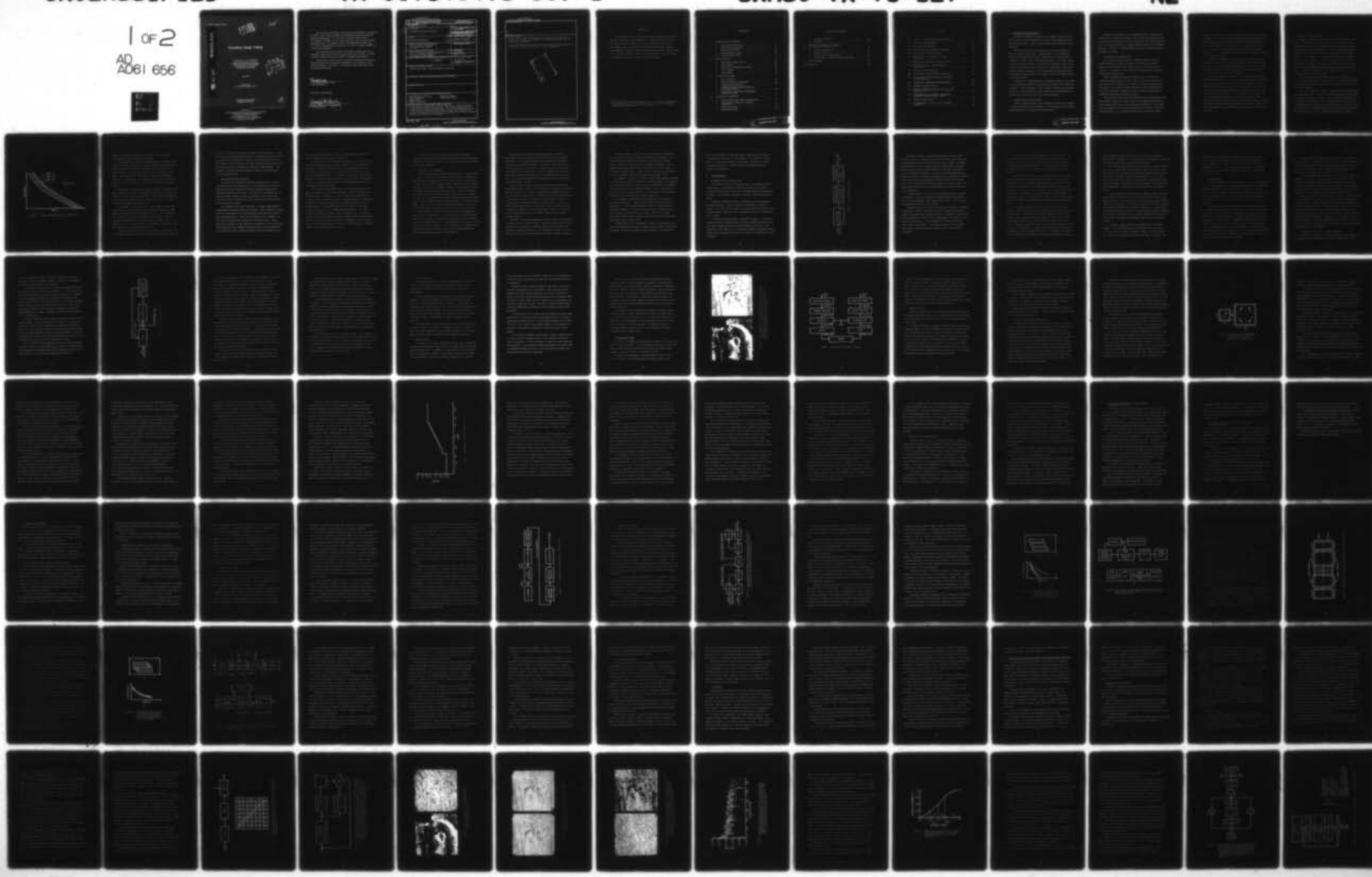
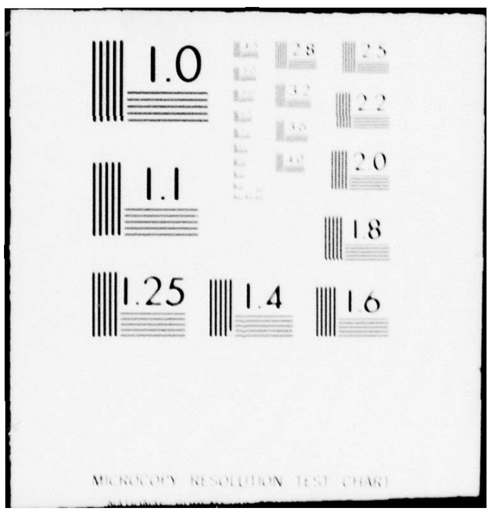
AD-A061 656 AEROSPACE CORP EL SEGUNDO CALIF ENGINEERING SCIENCE --ETC F/G 9/4
TRANSFORM IMAGE CODING.(U)
MAY 78 A G TESCHER F04701-77-C-0078
UNCLASSIFIED TR-0078(3478-30)-1 SAMSO-TR-78-127 NL

1 OF 2
AD
A061 656




1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART

REPORT SAMSO-TR-78-127

12

Transform Image Coding

Prepared by A. G. TESCHER
Electronics and Optics Division
Engineering Science Operations
The Aerospace Corporation
El Segundo, Calif. 90245

May 1978

Final Report
(1 April-30 November 1977)

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED

THE AEROSPACE CORPORATION

Prepared for
SPACE AND MISSILE SYSTEMS ORGANIZATION
AIR FORCE SYSTEMS COMMAND
Los Angeles Air Force Station
P.O. Box 92960, Worldway Postal Center
Los Angeles, Calif. 90009

This report was submitted by The Aerospace Corporation, El Segundo, CA 90245, under Contract F04701-77-C-0078 with the Space and Missile Systems Organization, P. O. Box 92960, Worldway Postal Center, Los Angeles, CA 90009. It was reviewed and approved for The Aerospace Corporation by A. J. Boardman, Electronics and Optics Division, and J. R. Henry, Satellite Systems Division. Lt Col D. A. Widen, SAMSO/YD, was the project engineer.

This report has been reviewed by the Office of Information and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and approved for publication. Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the interchange and stimulation of ideas.

Donald A Widen

D. A. Widen, Lt Col, USAF
Project Engineer

FOR THE COMMANDER

Joseph J. McGlinchey

J. McGlinchey, Col, USAF
Deputy for Defense Meteorological
Satellite System

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

18 19

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: SAMSO-TR-78-127
2. GOVT ACCESSION NO.:
3. RECIPIENT'S CATALOG NUMBER: 9
4. TITLE (and Subtitle): 6 TRANSFORM IMAGE CODING.
5. TYPE OF REPORT & PERIOD COVERED: Final Report, 1 April - 30 November 1977.
6. PERFORMING ORG. REPORT NUMBER: 14 TR-0078(3478-30)-1
7. AUTHOR(s): 10 Andrew G. Tescher
8. CONTRACT OR GRANT NUMBER(s): 15 F04701-77-C-0078
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The Aerospace Corporation, El Segundo, Calif. 90245
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: Space and Missile Systems Organization, Air Force Systems Command, Los Angeles, Calif. 90009
12. REPORT DATE: 11 May 1978
13. NUMBER OF PAGES: 112
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 12 112p.
15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Communication technology
Data compression
Image coding
Image compression
Image processing
Transform coding

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In recent years, considerable research effort has been done to optimize various digital communication channels. The significant growth of both commercial and military communication systems has emphasized the need for efficient data compression procedures. In the case of image transmission, the requirement is particularly demanding because of the significant amount of information to be transmitted. → next page

DD FORM 1473 (FACSIMILE)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

404 068

LB

19. KEY WORDS (Continued)

20. ABSTRACT (Continued)

In this report, image compression procedures based on transform techniques are reviewed. The emphasis is on adaptive and, especially, on rate adaptive algorithms.

The discussion includes an historical review, theoretical development, and illustrative examples of the transform image coding field.

PREFACE

The assistance of several individuals in the preparation of this report* is greatly appreciated. The suggestions of Professor William K. Pratt of the Image Processing Institute, University of Southern California, were most helpful. Ms. Carolyn Budelier provided expert editorial assistance. Ms. Peggy Ritter typed the earlier versions of the manuscript. The encouragement from The Aerospace Corporation and, in particular, from Mr. Charles J. Leontis to undertake this effort has been most welcome.

*This report, in somewhat different format, is concurrently being published by the Academic Press, Inc., New York, N.Y., as Chapter 4 of Image Transmission Techniques (W. K. Pratt, editor).

CONTENTS

A. INTRODUCTORY CONCEPTS 7

1. Image Coding Motivations 7
2. Theoretical Backgrounds 8
3. Practical Constraints 10
4. Implementation Concepts 13
5. Communication Models 15

B. IMAGE MODELS 18

1. Generalized Imaging System 18
2. Video Models 23
3. Stationary and Nonstationary Models 24
4. Vector Models 29
5. Image Quality 29

C. TRANSFORM CODING 31

1. Basic Concepts 34
2. Review of Classical Techniques 35
3. Image Quality Degradation 48
4. Tradeoff Between Imaging Design and Compression System 50
5. Fundamental Limitations of Nonadaptive Transform Coding 51

D. ADAPTIVITY CONCEPTS 53

1. Nonstationary Image Model Considerations 53
2. Communication Model as a Practical Constraint 54
3. Types of Adaptivity 55
4. Buffering Concepts 59

CONTENTS (Continued)

5. Learning Procedures 73

6. Classifiers 74

E. IMPLEMENTATIONS OF ADAPTIVE TRANSFORM CODING SYSTEMS 77

1. Memory Architectures 78

2. Specific Designs 79

3. Channel Error Considerations 98

4. Applicability of Concepts to Other Techniques 99

F. CONCLUSIONS 100

REFERENCES 105

FIGURES

1. "Typical" Distortion (D) vs Rate (R) Curves 11

2. Generalized Imaging System 19

3. Recursive Image Modeling 26

4. Demonstration of Nonadaptive Processing 32

5. Schematics of Transform Coding 33

6. Demonstration of Symmetry for Cosine Transform 37

7. Quantizer Demonstration (Tescher, 1973) 43

8. The NASA/Ames Compression System (Knauer, 1975) 58

9. Variable Rate Coding System 60

10. Variable Rate Coding through Prebuffering 63

11. Demonstration of Image Quality Discontinuity with Prebuffering 66

12. Variable Rate Coding through Postbuffering 68

13. Adaptive Transform Coder 83

14. Subblock Classification Based on Energy (Chen and Smith, 1976) 90

15. Flow Diagram of an Absolute Classification Procedure (Cox and Tescher, 1976) 93

16. Buffer Feedback Logic 94

17. Demonstration of "Typical" Buffer Feedback Behavior 96

A. INTRODUCTORY CONCEPTS

Transform coding was invented in the late 1960s. It has been close to ten years since various formulations of transform coding algorithms have been utilized; however, practical implementations only recently have been considered.

The primary purpose of this chapter is to review the progress to date on transform image coding. The discussion, in part, attempts to present the progress in the transform coding through an historical review. In addition, a critical review of this field is provided. A transform coding algorithm requires a relatively complicated implementation. Thus, impact on hardware design also must be considered. In addition to reviewing the field as it exists, an attempt is made to extrapolate into the future regarding potential implementations. By necessity, the opinion presented here is a personal hypothesis.

Because of the complicated nature of transform coding, it is necessary to demonstrate that potential implementations are likely to be of significant advantage. In this introductory section, theoretical background appropriate to transform coding is reviewed. The formalism is also developed which permits the analysis and critical review of this technology.

1. IMAGE CODING MOTIVATIONS

When a complex procedure such as transform technique is considered for data compression and, specifically, for image coding, it is important to trade off hardware complexity with potential benefits.

Transform techniques are more complex than other conventional and classical data compression algorithms. One should expect excellent performance given the complexity of transform algorithms. It will be demonstrated that, for nonadaptive transform technique implementations, only small future advances are expected. Further additional advances are expected primarily in adaptive techniques.

2. THEORETICAL BACKGROUNDS

Since several excellent reviews on transform coding concepts are available (Huang, 1972; Huang and Tretak, 1972; Huang, et al, 1971; Pearson, 1975; Schreiber, 1967; Wintz, 1972), only a short discussion is given here. In particular, Wintz provides concise explanation on the relevant theoretical backgrounds.

First, a definition of transform coding is given. The data compression technique transform or block coding is a procedure where a set of source elements is coded as a unit. The term "transform" indicates that the original set of elements is first processed by an invertible transformation prior to encoding. Motivation for this procedure is provided by the statistical characterization of typical pictures.

Several recent papers have analyzed the theoretical concepts associated with transform coding (Jain, 1976). However, the original paper by Huang and Schultheiss (1963) is still an excellent basic reference for a description of the advantages of certain transformations prior to encoding. The authors demonstrate that, for a correlated Gaussian source, optimum encoding consists of two steps. These are the decorrelation process followed by the application of a memoryless quantizer on the output of the transformation. If the Gaussian assumption can be extended to image sources, then the optimum encoding should always first involve a decorrelation procedure. For Gaussian sources, decorrelation implies statistical independence (Papoulis, 1965). Consequently, the decorrelated samples provide information only about themselves.

The optimum transform for decorrelation has also been discussed extensively in the literature, and it is usually identified as the Karhunen-Loeve (K-L) or Hoteling transformation (Habibi and Wintz, 1971). Thus, the early concepts indicated the desirability of a decorrelating transform and provided motivation for transform coding.

Other coding aspects can be related to the relationship between compression rate and distortion. Again, the available literature is quite extensive. The primary and obvious consideration is that transform coding introduces a distortion which is a function of the desired compression rate. The understanding of the tradeoff between rate (R) and distortion (D) will lead to practical procedures, i.e., for an at least locally stationary procedure, the relationship between rate and distortion leads to useful algorithms. It is demonstrated that the rate-versus-distortion formalism is fundamental in the proper design of a highly adaptive coding system. The appropriate formalism is developed as needed for the appropriate adaptive techniques.

3. PRACTICAL CONSTRAINTS

In order to establish a framework within which several different and possibly competing designs can be compared, it is necessary to define certain evaluation criteria. It is attempted here to present a somewhat simplified but practical demonstration of various theoretical information concepts. The concepts are presented as subsequent aids for tradeoff analyses of various designs. The following three primary quantities or parameters are considered: cost (C), distortion (D), and rate (R). For example, in Figure 1 several curves are shown which are representative of the typical functional relationships among the three parameters. Of these quantities, only the rate is well defined; it will always refer to the appropriate channel operation rate.

The term "distortion" is somewhat ambiguous. Frequently, this parameter is defined as the mean square error. In subsequent discussions in this chapter, the distortion parameter provides the mechanism by which the rate is controlled.

The "cost" is difficult to define. It can be defined in several different, although related, ways. Cost may imply financial investment in the design and hardware complexity. In many cases, an adaptive technique by itself may not be too complicated. However, the required channel protection necessitates added complexity which must be factored into the overall design. Even without the clear definition of the distortion and the cost parameters, the behavior shown in Figure 1 schematically indicates the potential

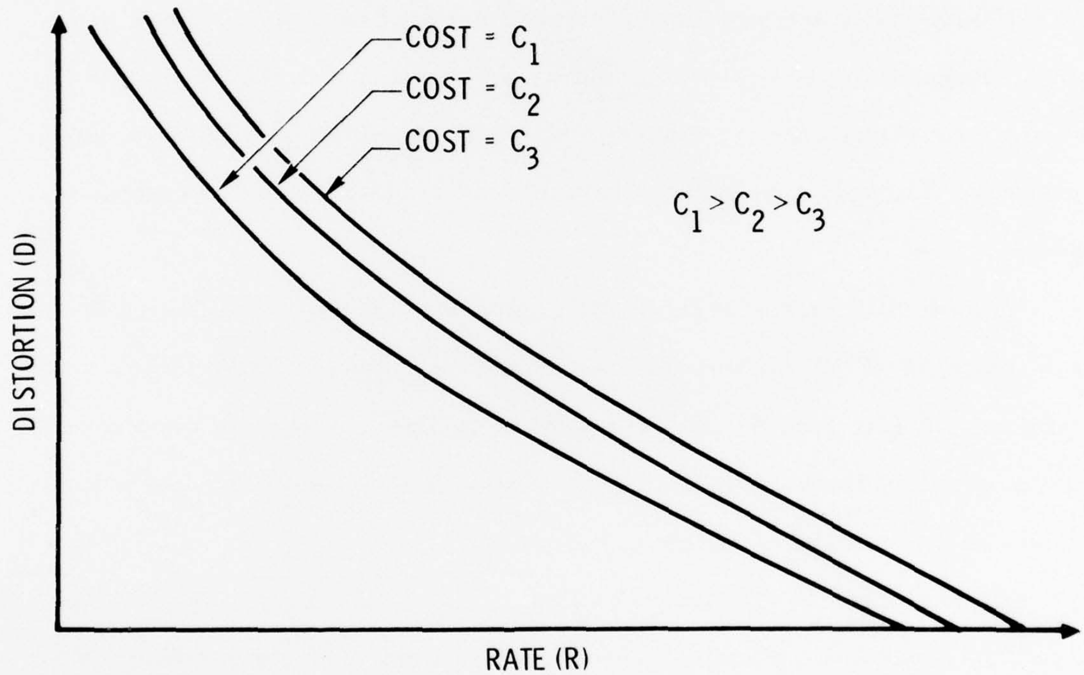


Figure 1. "Typical" Distortion (D) vs Rate (R) Curves

tradeoff among the three fundamental parameters: channel capacity, image quality, and the complexity of the design.

To be meaningful, the distortion measure should be related to image quality. In general, reasonable correlation exists between distortion and what appears to be a perceived image quality measure. Based on the general behavior of the presented curves, several conclusions may be drawn. Regardless of cost, with coding at a fixed rate, the attainable distortion or, equivalently, image quality is restricted to a certain range. In other words, only limited gain is achieved by going to more complex designs.

Equivalently, for a desired image quality and rate, the design may be rather expensive. It suffices to say that the cost aspects are complex and numerous. A fair portion of cost analysis to date is either of questionable value because of the rapidly changing hardware technology or irrelevant because of the context in which it was made.

For example, in transform coding, it is necessary to reformat the image into blocks, inasmuch as the normal ordering of picture elements (pixels or pels) follows the conventional raster scan. The input requirement by the transform coder necessitates a block ordering. This consideration is unimportant for a video type implementation. However, the appropriate memory consideration for reformatting becomes a major cost factor in the case of large size image formats.

Technology advances in terms of special purpose analog devices and general digital equipment have created a significant change in the attitude of designers as to what is reasonable to compute in a practical device. It was only a few years ago that implementation of a sinusoidal type of transform for a practical device was totally unacceptable, primarily on the basis of cost. But, based on current technology, the same computation may represent only a fraction of the cost associated with complicated coding equipment. The complexity associated with reformatting is probably more demanding for large size image formats.

4. IMPLEMENTATION CONCEPTS

Up to this point, the design constraints introduced were in terms of basic parameters of the coding process. It is important also to consider building blocks of the coding device from the standpoint of both general availability and hardware complexity. In general, the following device considerations apply: memory, arithmetic operations, control units, and auxiliary operations. Superficially, these concepts also would be utilized for the design of a computer.

The similarity is not really surprising. An image coding algorithm implemented in the digital domain is functionally a special purpose computer. However, fundamental laws of information theory also explicitly apply for a data compression device. The previous schematic curves of Figure 1 are the result of the appropriate constraints of information theory. Thus, despite the willingness to apply high-speed arithmetical operations, the potential performance gain in additional compression may not be too significant. On the other hand, basic concepts associated with the compression device can be functionally described in terms of computer terminology such as memory management and arithmetic operations.

Another general concept of significant importance is the relative complexity between the compressor and the expander. For most implementations, the compressor is likely to be more complicated than the decoder. The reasons are fundamental. The decoder is slaved to the coder; it can make no independent decisions. Conversely, the coding device has not only the same information available to the decoder, it has access to the original source. Consequently, it has the option of making decisions based on the availability of the original image.

The distinction in complexity between coder and decoder is not particularly significant for nonadaptive algorithms. It becomes rather important for adaptive procedures. For more complex and sophisticated compression schemes, the coder is likely to be significantly more complex than the decoder. Although this observation is useful for the design of image coding systems, it is unfortunate that, for most difficult image coding problems, the coder is more constrained than the decoder. The usual constraints are hardware utilization, power consumption, etc. Two obvious examples are satellite applications such as Landsat and the remotely piloted vehicle. For these applications, the coder complexity is severely limited. Conversely, the decoding process can allow for relatively greater complexity (Habibi, 1975a; Habibi and Samulon, 1975).

In terms of an overview, an attempt is made in this chapter to categorize various actual and potential designs via the introduced concepts. Specifically, memory, arithmetical operation, and complexity of controller functions are considered.

5. COMMUNICATION MODELS

The various concepts are introduced as they relate to image compression. It is important to realize that the concepts must be consistent with the fundamental constraints of the communication system. Image coding is a source coding procedure (Gallager, 1968). However, the source coding procedure also must be acceptable to a realistic communications environment (Shannon, 1948; Shannon and Weaver, 1949). These considerations are likely to become important for specific implementations. Virtually in all cases, the compressed data will be transmitted through a fixed capacity channel. For a nonadaptive data compression system, no unusual problem exists. However, for a variable rate compression system, a major constraint is introduced. In Section D, this problem is analyzed in detail.

Another problem area involves real-time image coding application jointly with voice transmission. An obvious example is the picture/phone type application. Here, the constraint is that not only the communication system must operate in real time, but unreasonable delays relative to voice transmission are not acceptable. This consideration is important whenever the expected response from the receiver is near instantaneous after the source completes the generation of its information.

Although transform coding techniques applicable to interframe applications are few, the previously introduced concepts imply practical constraints. In picture/phone application, three-dimensional transform coding cannot be applied to more than a few frames. If the coder simultaneously operates on several frames, an inherent delay in transmission with respect to voice will develop. The same constraint does not exist in a television broadcast application where a constant time delay is unimportant. For variable rate techniques, the required buffering develops into a valid and realistic constraint.

Channel noise is a subject that also deserves comment within this introductory section. However, it is by design (based perhaps on a personal bias of this author) that the source coding problem is handled separately from the channel coding problem. The latter problem is to maintain reliable communication through a noisy channel. Noise immunity or the lack of it in certain source encoding procedures is of some concern. But it is a rather unreasonable approach to require that a source encoding procedure be useful and operational in a noisy environment. However, in an overall communication design problem, the channel coding problem should be carefully analyzed.

Basically, the following philosophy is offered. First, one should design an optimum source encoding procedure. Next, the developed design should be appended or modified to be consistent with the need for channel error protection or detection procedures. This philosophy is likely to be reasonable for most source coding, particularly for transform coding procedures.

Transform coding techniques are more complex than most classical source coding procedures operating on individual picture elements. Thus, although additional hardware is involved, the incorporation of channel coding increases the overall hardware complexity only slightly. Several studies analyzed the inherent noise immunity of several source coding techniques including transform coding. These considerations are basically limited to nonadaptive, fixed rate techniques (Pratt, et al., 1974).

For adaptive, variable rate techniques, additional considerations should be followed when an actual communication system is designed. One approach is to eliminate bit errors via the appropriate channel coding implementation. The second approach only considers catastrophic errors. Again, the primary relevance is to adaptive systems. A catastrophic error may prevent the decoding of all subsequently transmitted information.

The avoidance of catastrophic errors can be achieved through synchronization procedures. This procedure periodically reinitializes the transmission. If the anticipated bit error rate is low, this procedure is perfectly acceptable. This synchronization loss due to catastrophic error can only affect a limited section of the image. Except for occasional comments relating to specific designs, the remaining portion of this chapter ignores the channel coding problem.

An additional reason for delegating the channel error sensitivity problem to another field is the constant research for more efficient compression techniques. Since these techniques will be the adaptive and variable rate types, it is unrealistic to process the output of these algorithms through unprotected noisy channels. So, consequently, once a decision is made to utilize a variable rate technique, it is necessary to eliminate or at least minimize channel errors. Failure to do so prevents the utilization of variable rate techniques.

B. IMAGE MODELS

1. GENERALIZED IMAGING SYSTEM

The output to the source encoding procedure is an analog function for image coding. At the destination, the image is viewed in analog form. Transform coding similar to other data compression systems utilizes digital techniques. Yet, the basic information, the image, in its original form is analog.

In general, the digitized image is considered as the original input to the coding procedure. Thus, the original analog nature of the image is ignored. As it will be shown in this section, the compression-introduced degradations may be of the same order of magnitude as the errors introduced by the imaging.

A generalized imaging system is illustrated in Figure 2. It should be emphasized that the imaging system introduces an additional problem in the sense that it superimposes its own characteristics on the source. It is stated that, in the image, it is not the natural redundancy associated with the original scene, but rather the impact of the particular imaging system which dominates.

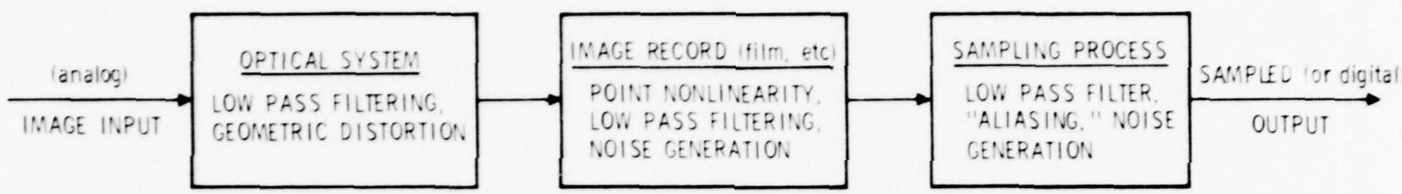


Figure 2. Generalized Imaging System

The image redundancy is quantified through its correlation model. Transform as well as other data compression techniques achieve the appropriate bandwidth reduction because the source (e.g., image) is correlated. A reasonable question may attempt to separate the correlation properties between the scene and its sampled equivalent. Reasonable assumption can be made about scene correlation. It can be easily demonstrated that the effective correlation in the sampled digital image is to a high degree induced by the very process which produced the digital equivalent of the original scene. While this statement may sound reasonably noncontroversial, it is fundamentally different for the two-dimensional case, unlike, for example, voice compression.

It is appropriate to review the steps through which image correlation is introduced. A primary observation is that all image forming systems act as two-dimensional low pass filters (Goodman, 1968). However, these low pass filters cannot have sharp cutoffs.

Although the basic theory has been available for some time, the proper consideration of this low pass filtering effect, including the permissible classes of low pass filters, has not been made. According to Lukosz (1962), the low pass filter associated with an imaging system is constrained according to a specific function. This constraint is the result of the physical nature of the imaging system. The impulse response of the appropriate linear system which represents the imaging system is restricted to be nonnegative.

The same nonnegative bound is applicable for image sampling. The sampling is performed by measuring the local intensity in the image. An aperture is placed over the appropriate location. The aperture in that local region collects the intensity which provides the local sample value. This aperture function is also related to and constrained by the Lukosz bound.

Additional differences between one-dimensional (such as voice) and image compression techniques should be emphasized. For a voice compression system, the sampling principle is readily applicable. The appropriate sharp low pass filter is available followed by the sampling process with a narrow aperture function. The latter closely approximates the Dirac delta function. Similarly, for voice reconstruction, the sampled data can be processed again by an essentially perfect low pass filter resulting within the given bandwidth in an essentially perfect reconstruction of the original voice.

For an image compression system, perfect reconstruction is not possible. The primary difficulty is the unavailability of the required perfect low pass filter. The perfect low pass filter implies an impulse response which is analytically the sinc function (Goodman, 1968). The sinc function has negative sidelobes which are not realizable according to the Lukosz bound. Although consideration of the imaging system as related to an image compression configuration is important from the philosophical standpoint, significant practical considerations also apply. Virtually all components of the conversion process from the analog-to-digital domain and, similarly, from the digital-to-analog domain involve the concept of realizability relating to low pass filters. The overall effect is of considerable importance in the design of specific image compression algorithms.

The following simple case serves as an example. A transmitted image may contain an appreciable amount of high frequency information. Unless the display device is capable of reconstructing the high fidelity information, it should be transmitted. Again referring to Figure 2, it may be noted that the generation of the digitized equivalent of the original scene involves nonlinear mappings, linear filtering, and noise addition. These effects are superimposed and, consequently, become an integral part of the digital image.

*In general, most coding systems do not explicitly consider the distor*tions which are introduced by the image formation process and analog-to-digital conversion. The so-called "eye" model indirectly includes indicated previous characteristics. The simple Roberts (1962) technique successfully incorporated the insensitivity of the human observer into a coding scheme. This technique (similar to a class of so-called dither techniques) adds noise to the image. Although the imagery is further degraded, the elimination of the noticeable and unpleasant contouring results in improved subjective viewing.

A transform image coding system introduces two types of degradations: a low pass filtering effect and additive noise introduction. The first effect is peculiar to transform coding. The second effect is common to most noninformation-preserving data compression, and is the result of the requantization performed by the coder. If the noise prior to coding is comparable to what is introduced by requantization, the visible degradation is acceptable. The reader is urged to consult the classical references on image formation and its relevance to information theory (e.g., Cornsweet, 1970; Fellgett and Linfoot, 1955; Huang, 1971; Levi, 1970).

2. VIDEO MODELS

The foregoing subsection provides a short review of generalized imaging systems. Additional specific considerations apply to video. The generalized system illustrated in Figure 2 includes video. However, for television-related compression problems, the quality of the input to the compressor is well regulated by convention. The noise content and low pass filtering associated with video recording and display are easily modeled and are available.

Consequently, and with some justification, researchers of the video compression field usually ignore the imaging aspect of video communication. These researchers are working in an industry well constrained by conventions, and the industry is not likely to change its procedures without strong overriding reasons (Fink, 1955). In addition to low pass filtering and noise addition in video recording, color and the statistics associated with interframe introduce additional modeling considerations. For the latter categories, insufficient research has been performed to date from the standpoint of practical implementation. The problem of color interframe coding has not been considered for transform techniques.

Transform coding is attractive for video systems. One reason for this is the relatively low quality of video. Here, the reference to quality is made in the relative sense. The comparison is with optical projection or with film-based systems. Thus, quality limitation of the video system permits further degradations to be introduced by the coder.

The other singular aspect of the video system is the fairly small format image size. The typical order of magnitude of 500×500 is not likely to change. Consequently, the appropriate memory requirements, such as buffering, are relatively straightforward. In contrast, a potentially high quality transmission system of still photographs would involve image sizes several orders of magnitude larger than current formats for video.

Within the framework of a critical overview, some comments pertaining to video systems are appropriate. The general assumption should not be made that what is acceptable to qualify for video is equally acceptable for higher quality display systems. Another observation relates to the development of new television cameras. Specific reference is made to charge-coupled device (CCD) type imaging, where the quality of recorded information may significantly improve. Consequently, if video systems improve with the availability of improved equipment such as solid state cameras, the permissible amount of degradation introduced by the image compression system will have to be decreased.

3. STATIONARY AND NONSTATIONARY MODELS

Following the description of the physical model of image formation, it is important to review the appropriate available mathematical models.

The common statistical models are limited, although they are useful for the analysis of image compression systems. However, these models are insufficient to achieve maximum compression.

A few general comments relative to modeling are appropriate prior to the analysis of the actual mathematical structures. The implementation of a mathematical model for an image compression system is the mechanism by which the a priori information is incorporated in the image coding process (Jones, 1976). By a mathematical model, one refers to information equally available to both the transmitter and the receiver. Most current mathematical models are based on stationary statistics as evidenced by implementations. In contrast, nonstationary statistics actually imply the lack of a useful model.

Generally, modeling refers to a parameterization procedure within the coding technique. In terms of nonstationary modeling, a "learning" procedure can be utilized. The learning function introduces a hierarchy of procedures. This learning procedure may be based on the original source in which case the appropriate model also must be transmitted. The model could also be derived from information already transmitted to the receiver. In this case, the model generation is without any overhead.

The conceptual schematic diagram of this type of modeling approach is shown in Figure 3. An essentially recursive approach to image modeling is introduced. In subsequent sections this concept is placed on a more concrete basis and related to specific algorithm development.

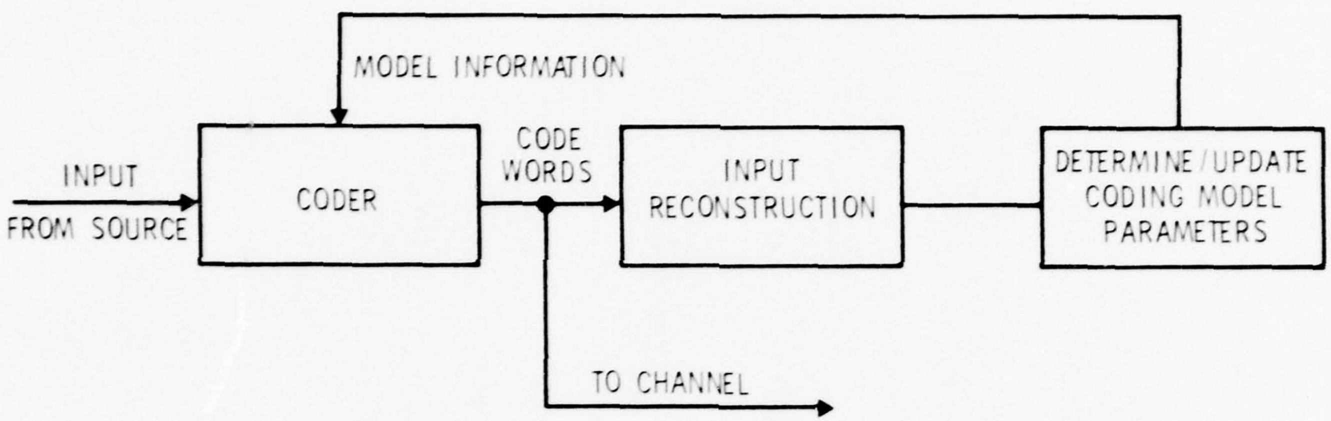


Figure 3. Recursive Image Modeling

The conventional mathematical model for image coding is the so-called Markov model (Habibi and Wintz, 1976). In this case, the image is describable by a correlation function which is the product of exponentials. This model is restricted to one or two parameters describing the two orthogonal directions in the image plane. An alternative form assumes rotational symmetry for the correlation. Despite its simplicity, the Markov model has achieved reasonable success for many implementations. In fact, in recent times the Markov model has been extended to the time variable interframe coding (Roese and Pratt, 1976). The popularity of the Markov model is the fact that it lends itself to convenient analysis. In other words, making the assumption that the image is a Markov process permits a certain amount of performance evaluation.

However, the simple Markov model has two basic limitations. (1) It assumes that the image correlation and equivalent power spectrum can be well modeled by the appropriate expressions. But why should a separable correlation model be pertinent to an image? The particular symmetry may be introduced through the process of converting the image into the digital form. (2) The Markov correlation model does not take into account the probable uncorrelated component that is representative of image noise. In fact, the basic limitation of the Markov model is not its relative inaccuracy but that it ignores image noise effects.

The assumption of the Markov model validity is reasonable for the original analog scene. The additional noise component in the digitized image, particularly at higher spatial frequencies, significantly degrades the model.

Consideration of noise in image coding suggests that an efficient coding procedure is likely to utilize a prefilter prior to coding. Of course, a filter prior to the compression process alters the actual model. This author is of the opinion that efficient compression techniques are adaptive. Therefore, utilization of the simple Markov model, while useful for analysis purposes, represents a basic limitation on the compression system.

Image nonstationarity can be demonstrated by numerous examples. Two specific cases are of particular interest: the nonstationarity within the original scene and the nonstationarity introduced and superimposed by the imaging process. The first case is almost obvious. One has only to look at any scene within his own surroundings and simply observe the variations in the local structure. The second case, the effect of imaging process, is somewhat more subtle. The primary concept here is that of an image typically inputed into an image coding system as a two-dimensional image. This two-dimensional representation is generated from the original three-dimensional scene.

Even without attempting to cover image formation in great detail, it is obvious that, in general, only part of the scene is sufficiently in focus while other parts are out of focus. It is inefficient to represent the entire scene, including the in and out of focus regions, by the same statistical model. Yet, no *a priori* information is available that would indicate to the decoder which parts of the scene are in and out of focus.

4. VECTOR MODELS

The image nonstationarity consideration suggests that, rather than modeling individual pixels, one could model groups of pixels. One may consider a set of picture elements as an element of an ensemble of observations.

In image coding similar to estimation theory, one utilizes two sets of quantities: the random variables (picture elements) and the relevant parameters associated with the random variables. In image coding, the parameterization is often ignored. For a more advanced modeling approach, a two-level processing is proposed. The coding of picture elements is performed as before, but one does not have the exact information about the relevant parameters. Consequently, a parameter estimation is also required.

The nonstationarity concept through a vector model is somewhat unconventional. However, a vector model is natural for transform coding. In its more advanced utilization, parameterization results in a more efficient data compression which also includes highly nonstationary techniques.

5. IMAGE QUALITY

The efficiency of an image compression technique can be quantitatively evaluated by comparing the change in the image quality between the original and the decoded image. Unfortunately, image quality is not well defined in terms of mathematical formalism. The problem of quantifying the effects of the compression, as well as any other processing step, is common to several techniques and not to transform coding alone. It is necessary to consider the image quality problem; otherwise, the discussed results are not meaningful.

Researchers have generally utilized the mean square error as the qunatity representative of the image quality. While mean square error techniques have been used successfully in many areas of optimization problems, the same evaluation procedures have been of limited use in the application of image coding. The mean square error is appropriate to the analysis of a stationary process. However, the concept of nonstationarity reduces the usefulness of the classical quality measure in image coding application.

The mean square error measure has been modified for image coding applications, as well as for general image quality analysis, to include an essentially prewhitening filter. The improvement although real is limited. In fact, the "eye" model in image coding modifies the mean square error formalism by providing a shaping factor for the image power spectral density.

It is necessary to quantify the efficiency, at least in a relative sense, of various transform coding procedures. One could also compare the various techniques in a common simulation. However, in most cases this is not practical. The comparison of different techniques at a single facility is a significant effort; furthermore, many compression techniques have been designed for particular image types.

However, there are some future directions specifically for image coding application. The mean square error quantity is somewhat superficial, because it is a global measure and does not indicate the distribution of the error within the image. Visual demonstration of the error distribution (here, the error is the pixel-by-pixel difference between the original and the decoded image) can yield a meaningful evaluation measure. It is reasonable to assume that the residual error between the processed and the original image should have small spatial structure and should be uncorrelated with the image.

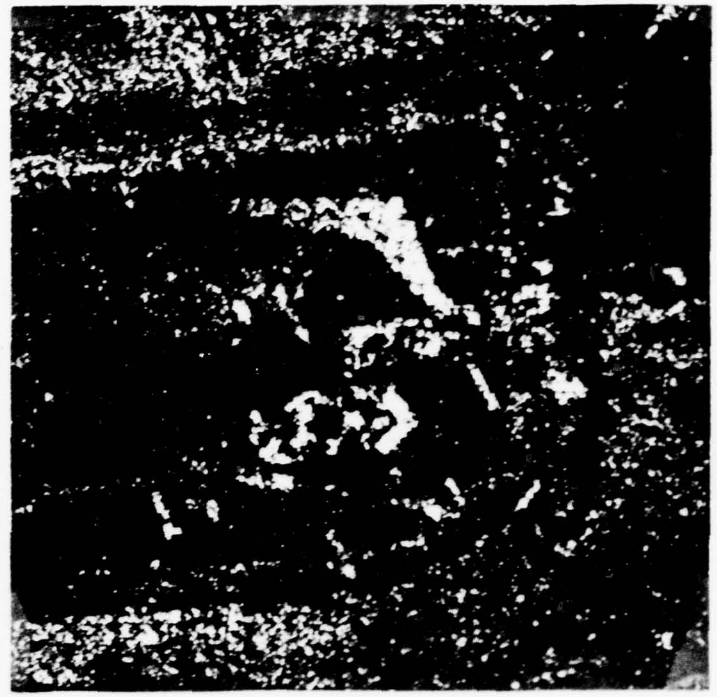
Consequently, a successful evaluation measure, which also suggests adaptive techniques, should result not only in a small mean square error for a "good" decoded image but also an error image that would appear as white noise. Visual presentation of the appropriate error image is indicative of the benefits of adaptive techniques. As shown in Figure 4, a nonadaptive technique generates a local error structure that is highly correlated with the original image.

C. TRANSFORM CODING

The basic block diagram representation of a transform coding-decoding system is shown in Figure 5. The coding system consists of a reformatting memory followed by the transformation and, finally, the actual coding process. The receiver is the mirror image of decoder.

Because of the availability of published material (Tescher, 1975, 1976, 1977; Wintz, 1972), including tutorial articles, this section is

(a)



(b)

Figure 4. Demonstration of Nonadaptive Processing. [(a) The original "site" image. (b) The error image associated with low-pass filtering in the cosine transform domain. The actual image represents thirty-fold scaling of the absolute difference between the original and the processed versions.]

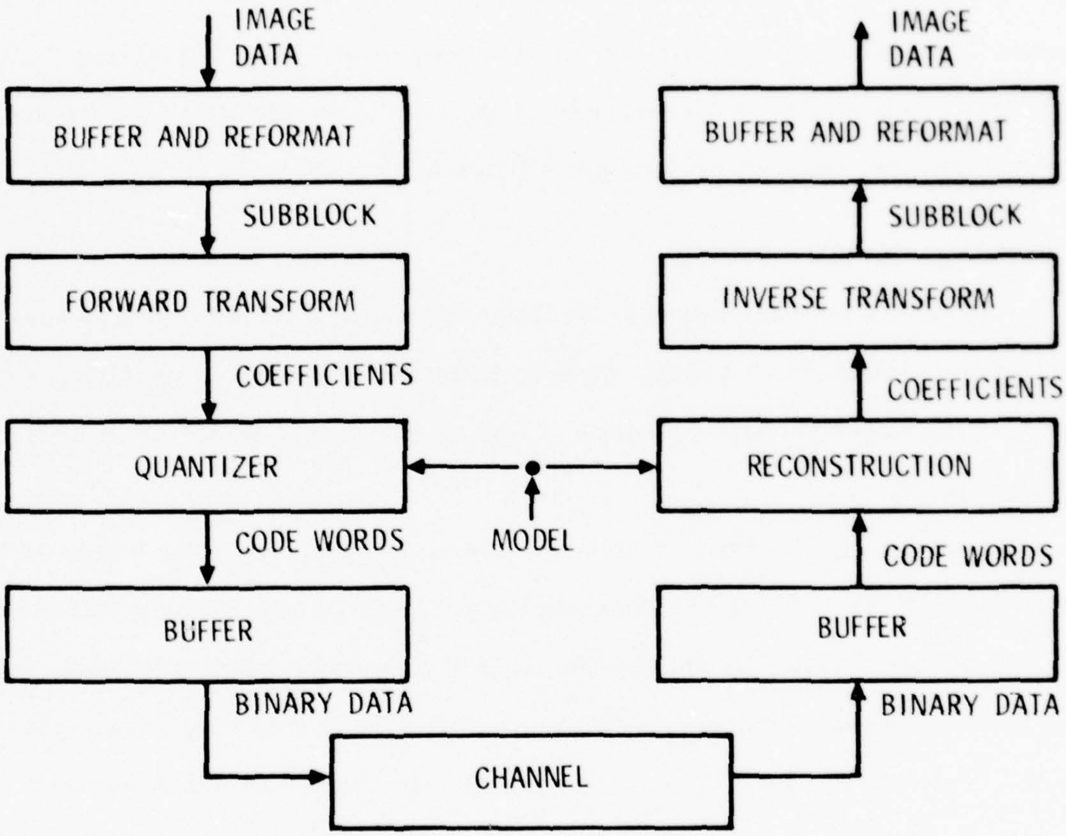


Figure 5. Schematics of Transform Coding

limited to a critical review rather than a comprehensive discussion of all apsects of transform coding. The primary subjects addressed are the actual transform and the coding of the transform coefficients. Unfortunately, most of the research work has directly related to transform algorithms and not to what should be done once the transformation is accomplished. In the opinion of this author, the emphasis on the transform algorithm, essentially at the expense of the coding procedures, has resulted in a suboptimum level of progress for transform coding.

1. BASIC CONCEPTS

Numerous formal papers in the literature demonstrate the principal advantages of transform coding. Here, an attempt is made to justify transform coding based on first principle. However, the mathematics is kept rather simple.

The basic justification of transform coding was offered by Huang and Schultheiss (1963). They demonstrated that the optimum coding procedure for a correlated Gaussian source consists of two steps. Decorrelation results in an independent source which is optimally coded by a memoryless coder. This basic concept, and the appropriate mathematical formalism, is sufficient to introduce the K-L transform. The source uncorrelation produces uncorrelated quantities that form the input to the coder. Early work in transform coding resulted in actual utilization of the K-L transform and provided impressive results (Habibi and Wintz, 1971).

Another observation is that the Fourier transform "tends to approach" the K-L transform with increasing transform size. Consequently, the Fourier transform can successfully approximate the K-L transform for reasonably large transforms.

Thus, the basic transform coding philosophy requires an uncorrelating transform. Furthermore, the Fourier transform is a close approximation to the actual K-L transform. Another important concept is adaptivity. For this case, no general theoretical solution is available.

Combination of the decorrelation transform with adaptive coding is a powerful method of data compression. The discussion of specific algorithms is deferred to subsequent sections.

2. REVIEW OF CLASSICAL TECHNIQUES

It is appropriate to provide a short historical review of transform coding. Early research on transform coding primarily concentrated on the transform algorithms. The Fourier transform was found to be a good approximation to the K-L transform. However, the early problem with Fourier techniques involved another classical behavior of Fourier analysis which has been referred to as the Gibbs phenomenon (Bracewell, 1965). This term really refers to a particular behavior of the Fourier transform as a method for the approximation of functions with discontinuities. The Fourier transform, when used for approximation (interpolation), replaces the discontinuity with the average value of the neighborhood of the discontinuity. The discretized implementation of the Fourier transform creates difficulties related to the Gibbs phenomenon.

The Fourier integral is defined over the entire real axis. However, the discrete Fourier series is defined over a finite interval. The additional property of the Fourier series is that it assumes the function it approximates is a periodic function. The basic period corresponds to the finite interval over which the Fourier series is defined. The practical implication is that the numerical Fourier transform assumes the first and the last points of the basic interval are neighbors. The Gibbs phenomenon is applicable to the Fourier series. Thus, the discontinuity between beginning and end of the basic period is approximated by the Fourier series with the appropriate average values.

In the practical sense, the discontinuity problem of the discrete Fourier transform in early applications of transform coding has been significant. The Fourier transform approximation when used over small blocks produces undesirable blocking effects. Several attempts have been made to minimize this blocking problem (Anderson and Huang, 1971).

In recent years, it was realized that significant improvement can be obtained in eliminating the blocking problem by introducing forced symmetry. The basic solution is rather simple. The original subblock is replaced by its symmetrized version as shown in Figure 6. The Fourier transform is applied to this new larger subblock. Since only the even terms are nonzero, the number of nonzero output elements is identical to the number of input elements. Moreover, this larger subblock has no discontinuities between the appropriate boundary points. Consequently, the

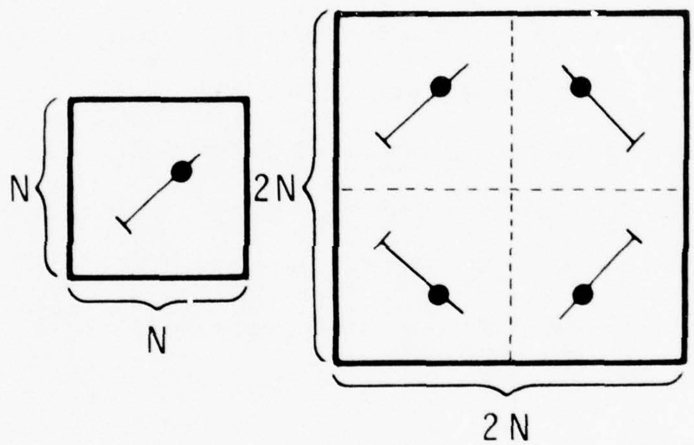


Figure 6. Demonstration of Symmetry for Cosine Transform. (The larger block is the "symmetrized" version of the original block.)

modified block structure does not eliminate the Gibbs Phenomenon but rather the original discontinuity that creates the Gibbs phenomenon.

It is surprising that this symmetrized approach to Fourier coding only recently was discovered. It has been demonstrated that the symmetrized Fourier transform, which has been designated as the cosine transform in the literature, closely approaches the K-L transform for many practical applications (Ahmed, et al., 1974). Recent studies further demonstrates that the cosine transform is virtually identical to the K-L transform for numerous practical conditions (Jain, 1976).

The cosine transform, unlike the K-L transform, can be implemented numerically through a "fast" transform. More importantly, the cosine transform virtually approaches the K-L transform performance with neither utilization nor knowledge of the source correlation. For the K-L transform, the source covariance model must be available to derive the actual transform matrix.

The cosine transform is a strictly deterministic transform. Conversely, the K-L transform is a class of transformation, and, for each application, it is a function of the appropriate covariance matrix. It is a significant practical benefit that the single deterministic transformation closely approaches in performance the entire class of theoretical optimum transformation.

The considerable amount of past research effort involving the various types of transforms appears to be of somewhat questionable value. Early techniques included utilization of several transforms including the slant, Hadamard, and a large class of transforms which can be derived from the first two transforms (Dail, 1976; Pratt, et al., 1974; Andrews, 1975).

The transform research may be summarized in two conclusions: (1) No deterministic transform has equaled the performance of the cosine transform, and (2) no theoretical justification was offered in the first place as to why the "other" transforms should be beneficial.

When this author attempted, at an earlier date, to produce a justification for the Walsh functions, the only reason he could find was that these functions were similar to the Fourier basic functions (Tescher, 1973). Thus, the Hadamard/Walsh techniques succeed as essentially ad hoc approximations to the Fourier/cosine techniques. A further fundamental observation is that, unlike the Fourier transform, the various other transforms will not asymptotically approach the K-L transform performance with increasing transform sizes.

Two other considerations of transform algorithms are the appropriate computability and the size. Under computability, the complexity of the potential device for implementation of the transform is defined. Fortunately, other than the general K-L transform, most of the useful transformations are implementable via fast algorithms (Cooley and Tukey, 1965). The suboptimal Hadamard and Haar can be implemented without multiplication. For specialized applications, a computationally more efficient transform, even at the expense of performance loss, may be preferred. For example, a specific tradeoff may involve the Haar transform which has a fast algorithm without any multiplication. On the other hand, various hybrid implementations for the cosine transform, e.g., through CCD technology, minimize the advantages of the suboptimal transform (Whitehouse, et al., 1975).

Transform size is an important practical consideration. The argument is often made that no benefit is obtained by choosing transforms larger than the image correlation distance, assuming the latter quantity is available. This approach is artificial and ignores the fact that the transform uncorrelates only the pixels within the subblock. It will not uncorrelate the pixels among subblocks. A typical subblock consists of 8×8 or 16×16 pixels. The appropriate reasoning is that the image correlation is not likely to exceed 8 or 16 pixels, respectively.

Although this reasoning has not been challenged, it is not valid. Even if all pixels within the transform block become decorrelated via the transformation, the pixels on the border remain correlated with respect to pixels on the borders of adjacent subblocks. Consequently, the argument against using large subblocks to achieve improved image decorrelation is not proper as it is based on the correlation distance. Recent work with relatively large size (256×256) transforms demonstrated significant performance gains over small size implementations (Tescher, 1973; Tescher and Andrews, 1974). However, for practical reasons it is advantageous not to exceed 16×16 or 32×32 size.

An additional argument against the larger size is the concept of adaptivity. It is still an unresolved question how to optimize the transform size both to achieve maximum decorrelation and yet include adaptivity consideration dependent on local image structure. These two concepts are contradictory and the solution is not at all obvious. To obtain maximum decorrelation, increasing the size is beneficial. Conversely, to adapt to the local image structure, a smaller block size is preferred.

Although it was earlier argued by Wintz (1972) that utilizing smaller blocks permits implementation of adaptive models, his suggestions have not been followed. The concept of adaptivity favors small subblocks. The same concept introduces consideration of overhead information. In general, the overhead associated with an adaptive transform algorithm is likely to become more important with decreasing transform block sizes.

In summary, three primary considerations relate to transform size. Larger sizes minimize block to block correlation. Adaptive procedures are likely to favor smaller sizes. Finally, the overhead information per subblock should not exceed a reasonable fraction of the available bandwidth.

Another potential study relates to the (spatial) shift variance impact of the transform. In general, transform coding is a shift variant procedure. The degree of shift variance is a function of the transform size as well as the local image origin. Consideration of the shift variance problem has been generally ignored.

An example of a simple shift variant filtering is the removal of the visually undesirable blocking introduced at extreme compression rates. The required filter will only smooth block boundaries without much processing within blocks.

The general effect of quantization of the source is a primary consideration for all source coding procedures. Quantization is the non-invertible mapping from the analog source to its discretized equivalent. Thus, the originally continuous parameters will be represented by integers. In general, the quantizer is the major error source in image coding. In most cases, quantization error is the only distortion present in the decoded image. All other error sources, such as numerical roundoff, are negligible in most instances. If not, they can be minimized or even eliminated by increasing register sizes and by using integer arithmetic. However, the quantization is a fundamental distortion and in most cases it is unavoidable.

As shown in Figure 7, the quantizer is a mapping from the continuous variable domain of transform coefficients into the domain of integers. These integers become the code words that are transmitted through the channel (O'Neal, 1971). An alternative superior approach is a two-phase coding technique in which the integers are the secondary input into an entropy coding processor. The output of this coder generates the final code words to be transmitted through the channel. The two-phase coding approach, except for Tasto and Wintz (1971), has not been pursued. Practical difficulty with entropy coding is that it is an open loop procedure.

The quantizer output is an integer which is also the code word for one-step coding. The relevant optimization procedure is the minimization of mean square error between original and quantized coefficients. Since all practical transform coding systems utilize unitary transforms, the mean

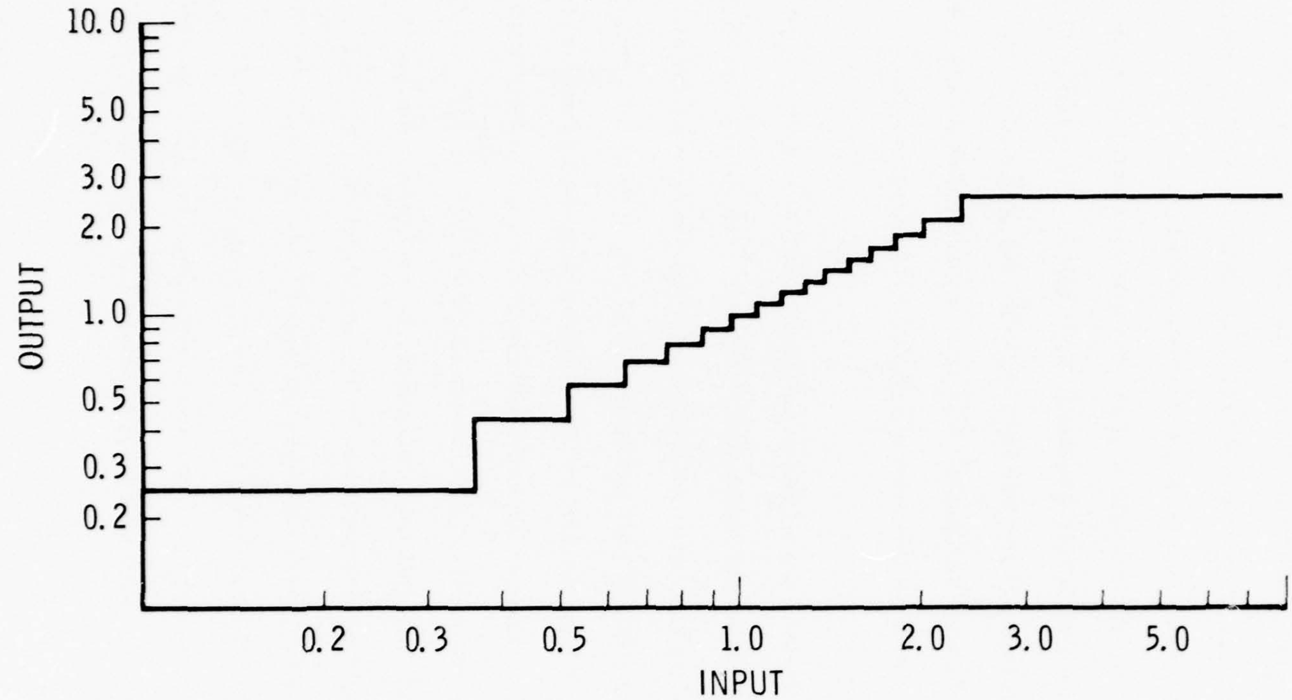


Figure 7. Quantizer Demonstration (Tescher, 1973)

square error is preserved under the transformation. Thus, quantizer optimization in the transform domain guarantees optimum spatial domain performance in the minimum mean square error sense.

For a given transform domain model, the quantization parameters are easily obtained via the Max (1960) quantizer algorithm. Knowledge of the probability density function of the random variable for a specified number of quantization bins permits computation of the required thresholds and reconstruction levels through the Max algorithm or approximations (Panther and Dite, 1951).

The actual coefficient quantization is performed in two steps: (1) The coefficient is normalized by its estimated variance, and (2) the normalized variable is processed by the optimum quantizer based on the modeled probability density function of unit variance. The number of bits for a quantized coefficient is determined by relating the assumed prequantized variance to distortion. The result can be derived through several different ways including formal rate distortion theory (Habibi, 1975b). To maintain equal distortion for each quantized coefficient, the required number of quantization bins should be proportional to the standard deviation of the quantity to be quantized. Thus, the appropriate bit assignment procedure follows directly from this argument. The coefficients are generally assumed to be Gaussian random variables. In recent experiments, the exponential density function has been found more appropriate (Tescher and Cox, 1976).

It should be noted that the Max quantizer minimizes the global error between input and output, usually mean square error. A more difficult problem is accurate determination of the prequantizer normalization factor. This problem is the determination of the transform domain model. The basic question is how to normalize each of the transform coefficients prior to quantization.

Most investigators assume the Markov correlation model for the image from which the generalized power spectral density is derived. This model is convenient and only requires one or two parameters. For the latter case, the image correlation in orthogonal directions is assumed to be different. The required normalization factors also can be obtained directly from a class of images through a learning procedure in which the appropriate transform coefficients of one or several images are averaged in the root mean square sense. This approach is found to be superior to the Markov model and it produces the required normalization parameters directly.

A disadvantage of the second approach is that transmission of the normalization parameters may significantly tax the available bandwidth. For a separate training set, a mismatch may develop between the training set and the image to be coded. The logistics of determining the normalization parameters for different image classes may become substantial.

An alternative technique by Tescher (1973) requires no a priori development of normalization parameters. The necessary parameters are obtained recursively from previously quantized transform coefficients and may be determined in real time. Although the early work was promising, it has not been pursued by other researchers primarily because of its complexity. The implementation was over large transforms. It was also an open loop procedure; thus, the actual compression rate could not be specified in advance. These early problems have been resolved and the procedure is discussed in Section C.4.

Since the recursive model is attractive for image compression, its theoretical advantages should be discussed. The previous two quantization models (the Markov and the training set model) are utilized in deterministic fashion. In contrast, the recursive model considers the normalization parameters to be a set of highly correlated random variables. The recursive model generates the normalization parameters from the already decoded values. Thus, the recursive approach to transform image coding allows an improved utilization of the transform domain without any overhead and prior model assumptions.

Several additional considerations should be mentioned. Clearly, the mathematical complexity has increased. Both the estimation procedure and the determination of the number of bits for each coefficient require real-time computation. The same hardware must also be included in the decoder since it performs the same recursive computation. The results of the computation in the decoder must be identical to that of the coder; otherwise, synchronization loss may develop. This requirement implies not only performance of the identical estimator procedures but also implementation of the appropriate algebraic steps in the identical order accuracy.

The potential problem is that the roundoff error may produce different bit assignments between the coder and decoder. This technique, like all variable rate adaptive techniques, is sensitive to channel errors. Thus, potential propagation of catastrophic errors must be avoided by periodic resynchronization.

Thus far in the discussion, no attempt has been made to specify the transformation. However, one could argue that the quantization model and its subsequent utilization are at least as important as the transform itself. The various quantization models are equally applicable to all transformations.

The importance of the proper quantization procedure cannot be overemphasized. The cosine transform decorrelates most practical types of imagery. Both theoretically and practically, the cosine transform closely approximates the K-L transform performance for reasonable block sizes of 8×8 pixels or greater. Thus, little, if any, additional gain can be obtained by the optimization of the transform itself. The real gain is in the modeling and the appropriate quantization strategy (Reader, 1975; Reis, et al., 1976). Entropy coding and/or carefully designed quantization strategy are more important than optimization of transforms (Huffman, 1952). A practical difficulty with conventional entropy coding is that it does not guarantee a specific rate. It may not even achieve bandwidth compression. The generalized adaptive coding strategy is also applicable to entropy coding. Conventional open loop entropy coding can be converted into a closed loop system through the generalized adaptive model.

For completeness, the so-called threshold coding should be mentioned (Anderson and Huang, 1971). Here, coefficients are quantized which exceed a fixed threshold. Threshold coding utilizes all the undesirable features of the various quantization models. It is open loop, highly error sensitive. Its coefficient modeling is poor since each coefficient is processed by the same quantizer if it is above the threshold. Specification of the coefficients to be transmitted requires considerable overhead. Threshold coding approximates entropy coding in complexity without benefit of the latter technique.

3. IMAGE QUALITY DEGRADATION

For image coding, in general, and transform coding, in particular, image quality measures which quantitatively correlate well with the perceived image quality and degradation as introduced by the coder, are unavailable. This is true even though conventional measures, such as the mean square error and its derivatives (e.g., the weighted mean square error), are helpful for comparison purposes. However, these measures are limited, in an absolute sense, for comparison between the original and processed images. Even without useful quantitative measure, degradations associated with transform coding can be discussed.

Three classes of degradations can be identified. (1) The coefficients are replaced by their quantized equivalent. (2) Some of the coefficients are replaced by zero. In effect, a great deal of the bandwidth compression is the result of this low pass filtering. (3) The transform coding procedure is not shift invariant. The processing is performed over adjacent blocks. If a new global origin is chosen for the process, the results are different.

Visually, these effects can be separated in most instances. Coefficient quantization introduces apparent additional image noise. Replacement of coefficients by zero in the decoder is a low pass filtering effect. If observable, the image will appear exactly as such. The decoded image compared with the original image may loose some of its details. Low pass filtering is rather noticeable at low data rates. Although this effect is transform dependent even for the cosine transform at low rates, the visual blocking at the transform boundaries is clearly undesirable. The term "low bit rate" indicates that only a small fraction of coefficients, say 10%, will be transmitted. The same problem is further magnified with suboptimal transforms.

For the Fourier transform, the Gibbs phenomenon is discussed earlier in the chapter. The cosine transform is less sensitive to this blocking effect. However, in the limit, it also introduces blocking.

Although reliable mathematical measures are not available, techniques that indicate image quality degradation in the decoded image compared with the original are available. The "error image" is usually helpful. The absolute value of the difference between the original image and the decoded image defines this quantity. For display purposes, the error image is usually scaled by a factor of from 10 to 40. If the information loss is small, the information present in the error image should also be small. For a high quality coding system, the error image will appear to be white noise.

4. TRADEOFF BETWEEN IMAGING DESIGN AND COMPRESSION SYSTEM

In most cases, the one who implements and the one who designs an image compression system utilize the output of some imaging device without any interaction or consideration of its original design. Historically, this procedure has been common. Television, as an example, was designed primarily for a particular image quality operation. The developer of the video compression algorithm cannot interact with the design of the television system. In principle, one would desire an interaction between the two.

Transform coding algorithm achieves its bandwidth reduction, in part, by not transmitting the entire transform plane. Thus, a nonadaptive transform algorithm particularly at low rates, acts as a low pass filter. A general imaging system also acts as a low pass filter. Thus, the required low pass filtering of the compression procedure could be achieved equally well in the analog domain. If the low pass filtering is performed by the imaging device, the required sampling rate also could be reduced. The equivalent rate reduction is achieved by reducing the number of picture elements representing the original analog image source.

The analog low pass filtering is not equivalent to low pass filtering performed by the transform algorithm. Unlike the transform coding algorithm, the imaging device is approximately a spatially invariant filter.

These comments represent an oversimplification. In principle, the algorithm input should be projected back into the analog domain. The digital image format is an intermediate step. Consequently, various aspects of the transform coding algorithms could be performed in the imaging device itself. Similarly, if the imaging device as a low pass filter is mismatched to the sampling procedure, e.g., in the case of oversampling, the apparent successful bandwidth compression is the result of low quality image generation in the digital domain and not necessarily evidence of efficient coding.

5. FUNDAMENTAL LIMITATIONS OF NONADAPTIVE TRANSFORM CODING

Various degradations associated with transform coding are described in previous sections. For a nonadaptive technique with a specified rate, the degradations are essentially unavoidable. Other than changing the bit rate, these degradations are deterministic. A nonadaptive algorithm is designed to be a fixed coding algorithm operating identically for all images and all image regions.

The question is: What is optimality? For a nonadaptive technique, the image to be coded is assumed to be a stationary source. Were the stationarity assumption valid, a nonadaptive coder could be an optimal image coder. In general, the stationarity assumption is not a good one.

Images usually have different statistical structures, both from image to image as well as within an image. Image nonstationarity results not only from scene nonstationarity but also from the imaging process itself. An imaging device projects a three-dimensional scene onto a two-dimensional plane. Consequently, those parts of the image in focus require high fidelity and, equivalently, a substantial portion of the available bandwidth.

Those segments that are out of focus (say, the background) appear to be blurred, thus requiring only a small fraction of the bandwidth.

This qualitative discussion suggests that the stationarity model for image coding might be a severe design limitation. In his review, Wintz (1972) indicated that nonadaptive transform coding is of little, if any, benefit over other conventional but simpler image coding techniques (Habibi, 1971). The real gain is in adaptivity. Unfortunately, research advances obtained to date are limited.

In the next section, primary concentration is on nonstationary models, algorithm implementations, and relevant concepts.

D. ADAPTIVITY CONCEPTS

Several years ago, Wintz (1972) urged the implementation of adaptive techniques. In this review, essentially a half decade later, one would be pleased to discuss the numerous algorithms and implementations dealing with adaptivity. Unfortunately, this situation has not developed. Most techniques have followed primarily nonadaptive procedures.

The original suggestions for adaptivity in transform coding are valid today. In this section, the principles of adaptivity are reviewed. Furthermore, various implementation methods are discussed. Unfortunately, the available literature on which to base this analysis is rather limited, and the following discussion is heavily based on the author's own research.

1. NONSTATIONARY IMAGE MODEL CONSIDERATIONS

Here, the assumption is made that imagery as a source should be modeled as a nonstationary representation. Consequently, coding procedures, which to a large extent freeze the algorithm, are not appropriate. In the following discussions, the image coding model is part of the information to be utilized by the decoder. This fact does not necessarily indicate that new or additional modeling information must be transmitted to the decoder. For an efficient system, the modeling information can and should be derivable from previously decoded information.

The basic assumption is made that the image coding procedure is the transmission of fluctuation information about a preassigned model. However, this model is part of the information to be transmitted. Ideally, the model would permit local changes through parameterization. More importantly, it would assist in meaningful redistribution of the available bandwidth among partitions of the image.

However, before proceeding with various adaptive algorithms, it is necessary to review the impact of a realistic communication system model as a constraint.

2. COMMUNICATION MODEL AS A PRACTICAL CONSTRAINT

Although image coding is applicable for nonreal-time applications such as storing pictorial information on magnetic tape, the primary application is real-time image transmission. Thus, the image coder must be consistent with the constraints of a realistic communication model. The appropriate limitation is straightforward.

With increasing adaptivity, the coder output will fluctuate in rate if the adaptivity permits a variable rate compressor. The relevant problem is how to interface the variable rate compressor with the fixed rate channel. For a variable rate compressor, communications model constraints become a necessary consideration.

In principle, the solution is simple. The compressor (source encoder) must be interfaced with the communication channel through a rate equalizing buffer. This buffer permits the deviation in bits from the average rate as required. However, the practical solution is somewhat involved.

Another practical consideration of a communication channel relates to channel errors. The various applicable channel coding techniques can be classified into two groups: In one case, through algebraic coding, channel errors are essentially eliminated. In the other case, a fixed, nonnegligible rate of channel errors, including catastrophic errors, is permitted. However, periodic reinitialization of the algorithm is performed.

In general, it is difficult to evaluate the impact of channel errors for adaptive procedures. Erroneous bits may be significantly important, particularly for overhead information. On the other hand, erroneous bits that result only in incorrect reconstruction of a single transform coefficient may result in relatively minor image degradation. Consequently, the primary impact of a communication system is the need to interface the inherently fixed rate transmission system with a locally fluctuating rate source encoding system. The appropriate problem is to design a global fixed rate system that is highly rate adaptive locally.

3. TYPES OF ADAPTIVITY

Adaptivity procedures can be classified into two broad categories. For the first, the source coder rate is constant. However, various other coder parameters are changing. For the second, in addition to other parameters, the local compression rate is also variable. The second adaptivity category involves implementation difficulty because of the varying rate.

In addition to classification according to rate, other considerations are appropriate. For a highly adaptive system, the degree of adaptivity or the number of different ways the coder can operate may be either a large number or essentially a continuous parameter. In this case, the overhead information needed by the decoder to determine the operation mode should be decodable from previously decoded data. Otherwise, the transmitted overhead would require a greater fraction of the available bandwidth.

A somewhat simplified adaptivity procedure may allow a few operational modes, say four, and within each class the appropriate coding algorithm may be considered to be a nonadaptive procedure. This type of approach has several advantages despite its limitations. Basically, for a small number of algorithm modes, one designs several independent coding algorithms corresponding to the different classes. Thus, other than determining within which class the algorithm currently operates, the decoder is essentially nonadaptive.

The appropriate modeling is similar to a Markov chain. For each subblock, the source corresponds to one of the states of the Markov chain, and the coder processes the data according to the same state classification. For the four-class example, two bits must be allocated for overhead per subblock. Consequently, the available bandwidth is primarily utilized for the transmission of transform coefficients for any reasonable size transform block, say 8×8 or larger.

Before proceeding, a short overview of a specific transform coding system that has been developed is in order. A system built at NASA by Knauer (1975) and his coworkers is the three-dimensional implementation of the relatively simple algorithm originally proposed by Landau and Slepian (1971). The NASA model is an adaptive interframe coder utilizing the $4 \times 4 \times 4$ Hadamard matrices. The system operates at a constant rate. The appropriate normalization model (i.e., the quantizer) varies according to local activity. The implementation is straightforward since the rate is fixed. A frame storage is required for four frames. The appropriate transform is three-dimensional Hadamard. Two-bit overhead indicates the classification information to the decoder. The system operates at one bit per pixel with excellent image quality compared with the original video image. The actual implementation of the algorithm has been performed at regular TV rates.

In Figure 8, a schematic description of this algorithm is shown. The motivation to assign different classes is based on temporal activity in the image. Thus, the amount of image motion within four frames determines the allocation of the available bits in the temporal direction. Since the overall rate is fixed, the total number of bits for each $4 \times 4 \times 4$ transform block is constant.

The discussed algorithm at one bit per pixel is probably close to the lowest rate at which the system could operate with acceptable image quality. The basic system limitation is that each three-dimensional transform block utilizes an identical fraction of the available bandwidth. For an interframe coding problem, the local variability, both spatially and temporally, is likely to be significant. The current NASA concept is the only existing real-time transform coding system either for intraframe or interframe coding.

The NASA design incorporates several diagnostic modes, which also serve an educational purpose by demonstrating various motion types within an image. It is also helpful to design quantizer parameters associated with various classes. Other than the requirement for the four-frame memory, the system design is straightforward. Implementation through the Hadamard matrices eliminates multiplications.

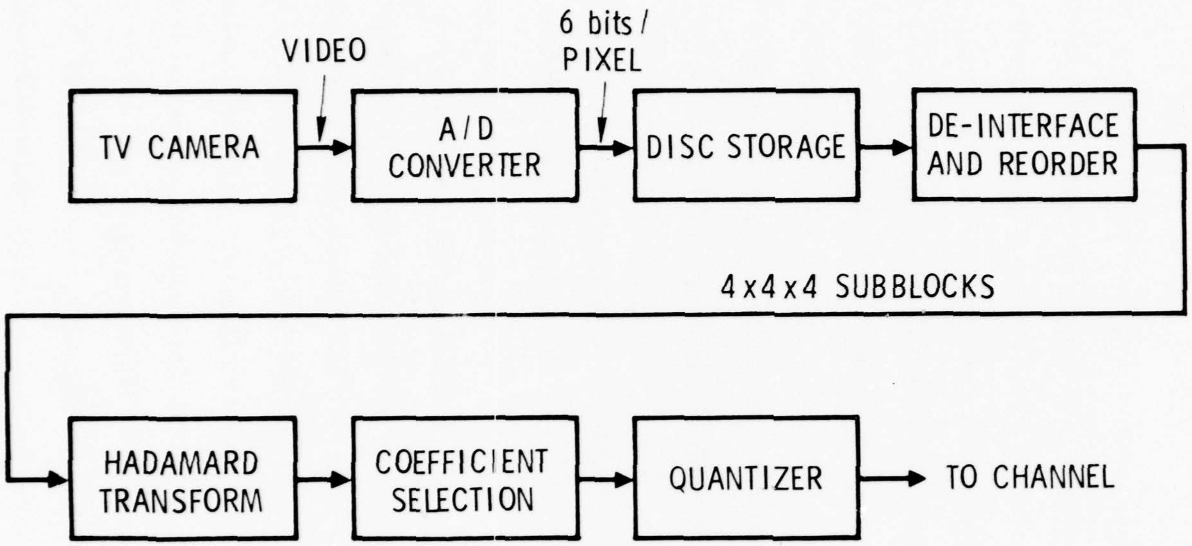


Figure 8. The NASA/Ames Compression System (Knauer, 1975)

4. BUFFERING CONCEPTS

Variable rate source coding techniques require rate equalizer buffers. This requirement is easy to demonstrate. Since the communication channel operates at a fixed rate, the source coder rate fluctuation must be absorbed by the additional buffer. The buffering design is important.

The general model of a variable rate system is shown in Figure 9. The rate equalizing buffer is implemented between the source coder and the channel. The important consideration is how the source coding system controls gross parameters of the decoder. In a realistic system, the source coder cannot operate at variable rates without some externally controlled mechanism. An open loop source coding system is not acceptable. This statement is consistent with the assumed philosophy of nonstationarity. Even for a variable rate system, image nonstationarity may be more extreme than what the rate equalizer buffer could allow without an additional controlling mechanism.

For a transform coding procedure (Figure 5) a "prebuffer" is required. This buffer performs the image reformatting into subblocks. For a variable rate system, the additional buffer is required to accommodate the source coder fluctuations. The basic problem is to determine how to control the overall coding parameters in terms of utilization of the various buffers. Two general solutions are available.

The first approach uses the reformatting buffer to control the coding parameters. The second approach uses the rate equalizer buffer in a feedback loop. The two techniques are not equivalent. The first technique is

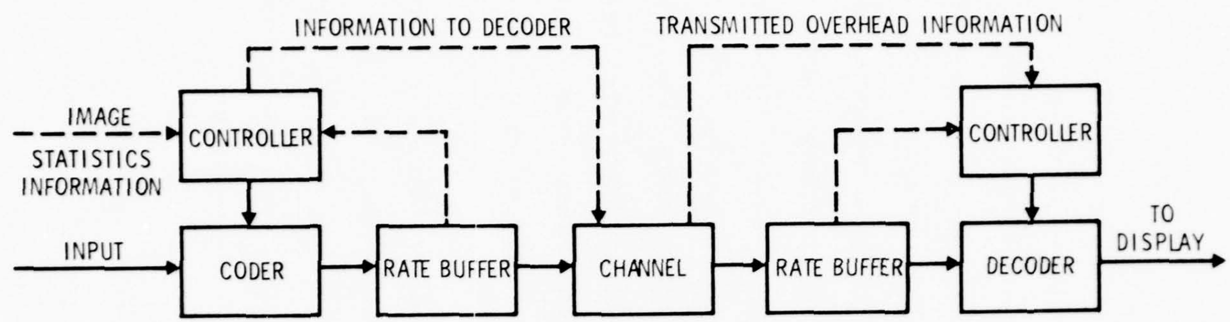


Figure 9. Variable Rate Coding System. (The broken lines indicate potential information flow for the coding model transmission. Specific implementations are considered in Figures 10 and 12.)

logically easy to understand; however, it is less efficient. Here, the reformatting buffer serves as a control mechanism. Although general designs are deferred to the Section D. 5, the principle of buffer control is introduced here.

One assumes that the adaptivity procedure does not refer to the entire image but is restricted to only that portion of the image which resides, at any one time, in the reformatting buffer. Consequently, each image segment corresponding to the prebuffer is transmitted at a fixed rate corresponding to the required channel rate.

Larger reformatting buffers result in more adaptive systems. For conventional raster type image structure, the transform blocks are $n \times n$ subblocks. Thus, the minimum prebuffer size is n lines. For a 512×512 image, with 16×16 transform subblocks, the adaptivity is implemented over 32 subblocks. The result is a significant redistribution of the available bandwidth over a reasonable image size. If each image segment corresponding to the prebuffer is coded at the channel rate, the coding system operates at the appropriate channel rate.

The logical operational approach is to code the entire prebuffer and to place the output into the "postbuffer." By design, the postbuffer is filled if the ratio of memory sizes is equal to the appropriate ratio associated with the image compression. The compressed data may be transmitted through the channel at its fixed rate. A short delay will develop corresponding to coding the image segment that fills the prebuffer. For a practical implementation, two memories could be used. While compressed data is transmitted from the equalizer buffer, another compressor operates on another identical prebuffer. The delay is also a result of the finite time required for classification. The reformatting operation time can be minimized. Each line segment can be placed in the proper location of the prebuffer in real time. Thus, after the last segment is available, the coder can immediately begin its operation.

However, the controller must look at the entire image region residing in the reformatting buffer in order to arrive at the correct control parameter. Hence, the engineering design must take into consideration two types of delays. One delay is associated with image line reformatting into subblocks. The additional delays are required by the controller. Thus, the controller should be fast, because during the operation additional buffering arrangement must be made to accept incoming data.

Increasing the region over which the algorithm is adaptive can be accomplished only by increasing the reformatting buffer. For example, to double the adaptivity area, the prebuffer size must also be doubled. It should be noted that the prebuffer contains the image in full resolution. An increase in prebuffer size refers to the original image representation. The smallest prebuffer size is determined by the minimum needed for reformatting. It is n lines for an $n \times n$ transform coder.

The general concept of prebuffering may be understood by referring to Figure 10. Several nonoverlapping image segments, which reside one at a time in the prebuffer, are shown with the appropriate D vs R figures. For each image segment, a distortion measure is associated with the

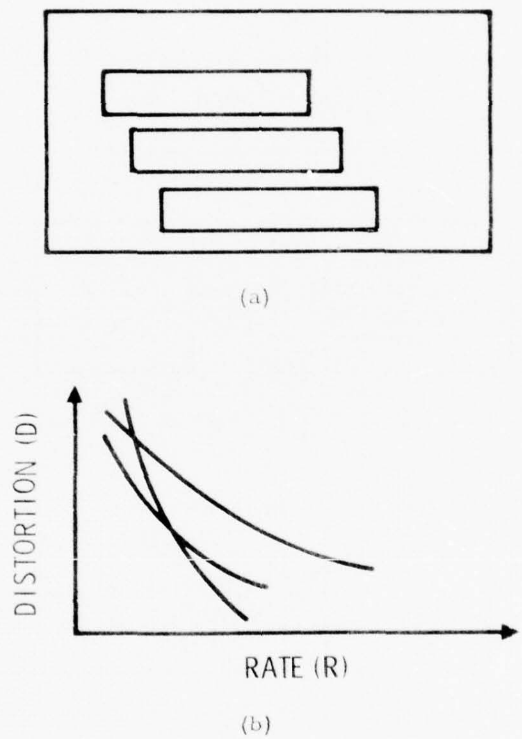
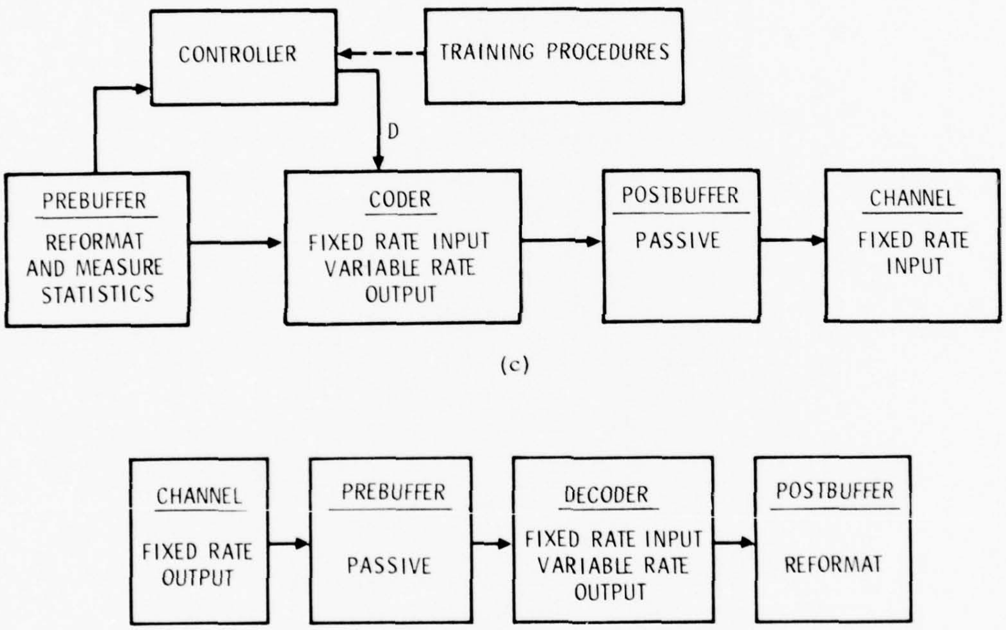


Figure 10. Variable Rate Coding through Prebuffering. [The several nonoverlapping image segments of (a) may be modeled by the several D vs R curves in (b).]



(d)

Figure 10. Variable Rate Coding through Prebuffering (Continued). [The coding system is shown in (c). The decoding system is shown in (d).]

compression rate. The distortion parameter may be considered as a global parameter for that image region. This parameter controls the algorithm to yield the required rate for the particular segment.

For a nonstationary model, the appropriate D vs R curves are different for various regions. The controller must determine for the image segment residing in the prebuffer the corrected D parameter which yields the required global rate. Specific designs are considered in Section D.5.

A potentially negative aspect of prebuffer utilization is image quality discontinuity between adjacent image segments. This statement is more understandable through a specific simple example. A small area (A), corresponding to a transform block, is present in two different image regions which occupied the reformatting buffer at different times. By assumption, one region is "busy" and the other one is "quiet." In the busy region, fewer bits will be available for A than in the quiet image region. If the same subblocks are also adjacent to each other, the apparent image quality associated with them will be significantly different, because, in one case, the coder allocated more bits to the same subblock than it did in the other case. This example is illustrated in Figure 11.

The prebuffer segments the image into nonoverlapping regions corresponding to the reformatting buffer size or its multiple. It performs a rate controlling function for each nonoverlapping region. A controller design through a reformatting buffer is inflexible. This system is heavily image format dependent. Furthermore, it is not easily adaptable to an image coding application for varying image sizes.

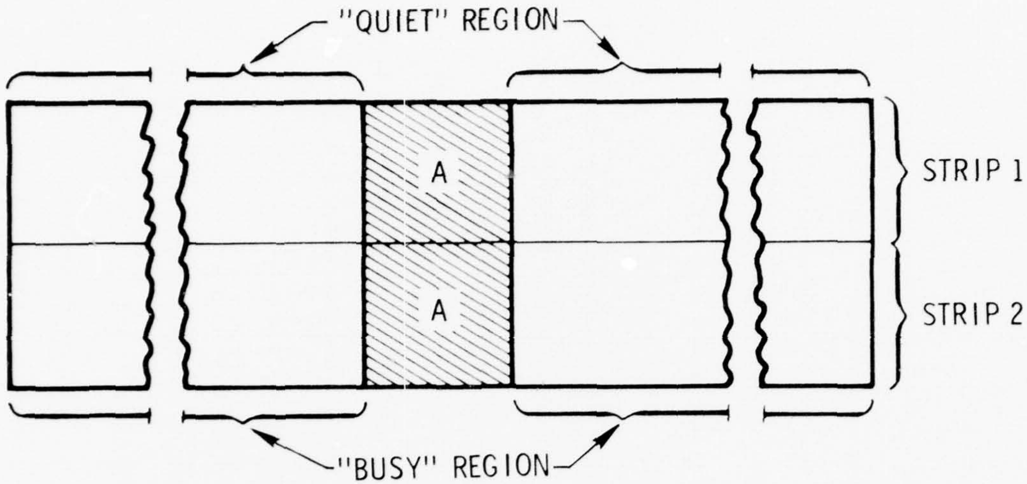


Figure 11. Demonstration of Image Quality Discontinuity with Prebuffering

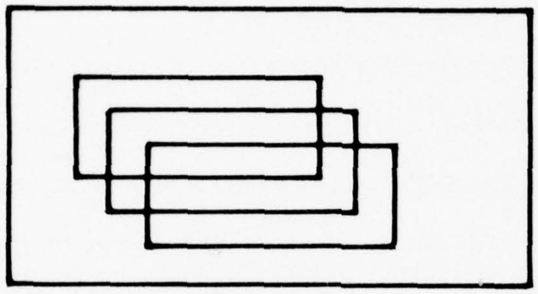
The prebuffering concept eliminates the difficulties associated with an open loop operation. However, difficulties and disadvantages remain with a prebuffer utilization. Another buffering approach, postbuffering, eliminates most of the indicated disadvantages.

In Figure 12, a postbuffer control communication system utilizing adaptive transform coding is illustrated. It should be noted that, although advantages are highly applicable to transform coding, this approach to adaptive coding is general. Thus, the same concept is equally applicable to transform and other coding procedures where a "feedback" is implementable. Utilization of a rate equalizer buffer with feedback is not new. However, the mode of feedback mechanism implementation is novel.

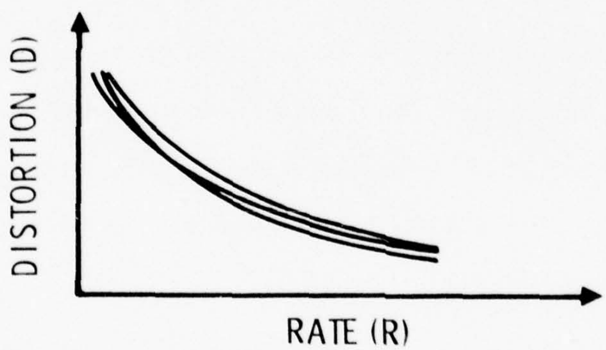
The actual solution involves modification of the distortion parameter based on an image region that previously resided in the equalizer buffer. The earlier introduced representation of the D vs R curve will be utilized. The controller constantly adjusts the appropriate distortion parameter to yield the required rate.

The basic problem is how to deal with nonstationarity for overlapping image regions. For each region, one must approximate the D vs R curve. Subsequently, the correct distortion parameter to yield the appropriate rate is determined. By design, adjacent image regions significantly overlap. For example, two adjacent image regions (say, A and B) contain a common area exceeding 95 percent for each segment.

For this example, the appropriate D vs R curves will only vary a small amount between the two segments. This result is expected since, except for a 5% fraction of each segment, the two are identical.



(a)



(b)

Figure 12. Variable Rate Coding through Postbuffering. [The greatly overlapping image segments of (a) result in almost identical D vs R curves in (b).]

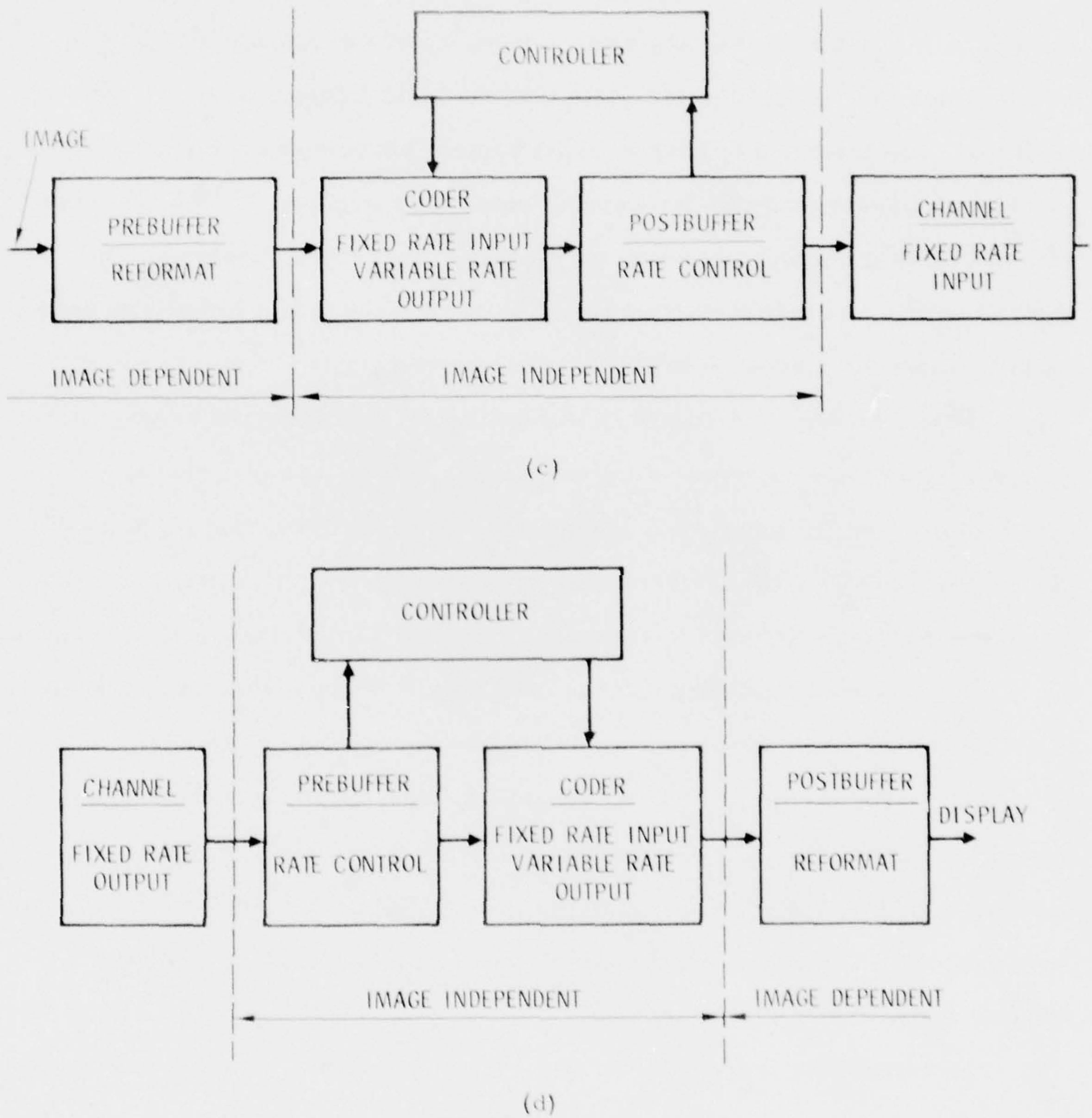


Figure 12. Variable Rate Coding through Postbuffering (Continued). [The coding system is shown in (c). The decoding system is shown in (d).]

One may assume that the appropriate distortion parameter is available for segment A. Within that segment, utilization of the appropriate D parameter yields the required rate. Note that the local fluctuation is not constrained. One attempts to have a global control parameters for segment A, which on the average yields the correct rate. For segment B, it is reasonable to utilize the same distortion parameter. Therefore, by design, the controller chooses that distortion term for segment B. The coder will process the input for a small fraction of that segment.

At the conclusion of coding of the last small segment, the average rate over the past image segment will differ from the desired rate. Thus, the rate equalizer buffer acquires a small bias. Consequently, the distortion parameter must be modified to adjust the rate properly. Furthermore, the coder must also be adjusted to counteract the bias in the rate equalizer buffer.

Operationally, one does not have the D vs R curve. However, its local history is obtained in terms of samples on the curve. Through those samples, the coder, based on the past, can estimate the desired D value by linear interpolation utilizing the two previous elements on the D vs R curve. The estimation could also be extended to the differential of this curve. The given approach could be implemented by methods of Kalman filtering; however, this has not yet been attempted.

The estimation procedure, in general, requires a local estimate of both the distortion parameter and its change of rate with respect to R. Since, for a practical communication system the differential will have the same sign, a simplified estimator could be utilized.

The indicated conceptual procedure is a buffering technique that utilizes the important parameters, i.e., the buffer size and its projections into the image. This technique is stable because variations in buffer status are immediately fed back into the control operation.

Before specific implementations are considered, the general usefulness of the discussed approach can be compared with the controlling operation through prebuffering. The coding mechanism via postbuffering is highly flexible. Other than the reformatting buffer, the system is completely format independent. The reformatting buffer performs no controlling operation. Consequently, the coder can be utilized for different image formats by modifying the reformatter.

This consideration may be important if a general adaptive transform coder is to be developed. Since this system will be of considerable complexity, it would be desirable to use the same design for various applications including different image types and formats.

The coder operates with one block of data at one time; therefore, the postbuffering concept can be extended to include several multiplexed coders that utilize a single postbuffer. The postbuffer controls each coder. Each transform block is treated independently by the coder.

Similarly, the coder may operate on subblocks in a nonsequential order. The rate fluctuation from block to block will, of course, differ for an alternate ordering; however, the coding efficiency should remain the same.

This consideration is not entirely artificial. Transform coding requires a considerable number of arithmetical operations. At video rates, this requirement may be quite demanding. In order to reduce the effective computation rate, several coders may multiplex their output into a single rate buffer.

Another observation is that, unlike the case of prebuffer control, no discontinuity in image quality among different image regions develops as a result of adaptive coding. The distortion parameter D is updated continuously. Two similar adjacent subblocks are not likely to be coded differently because of discontinuity in the D parameter. To minimize the problem of image quality discontinuity, the postbuffer should be sufficiently large, say, two or more, image strips.

Memory utilization is improved through postbuffer control compared with prebuffering. The postbuffer control operates through the compressed image, unlike the prebuffer control. To improve adaptivity with postbuffer control is less demanding, since the necessary buffer increase is in terms of a compressed image.

Another advantage of postbuffer control is parallel controller operation. By design, this function may be simultaneously performed with the coding operation. Consequently, the requirement for additional buffering to permit the controller to determine the appropriate distortion parameter is not necessary.

The controller updates the distortion parameter in parallel significantly less frequently than the actual arithematic operation rate required for the coder. Thus, the computational requirement for the controlling system is limited. It should also be noted that the buffer control is strictly based on buffer behavior, and the required information is based on previously transmitted data. Since the decoder performs the same function, the entire control operation can be performed with no overhead.

5. LEARNING PROCEDURES

Compression algorithms with separate adaptivity classes require pretraining prior to actual coding. For these algorithms, a parameterization is required in order to relate to local image structure. In order to perform the adaptive coding, it is necessary to develop techniques which meaningfully assign the mode of the image coder into one of available classes.

In general, it is necessary to develop training procedures that yield the appropriate coding classes. Thus, prior to the actual coding, a training procedure must be implemented. The result of this training of the coder must also be available to the decoder as well. It requires considerable overhead to transmit this information. Consequently, the training procedure is not likely to be performed frequently.

Alternatively, several types of training images may be available to both coder and decoder. When coding a specific image prior to transmission of the compressed image, the required parameters are specified by indicating the appropriate training image.

Performance variation of an adaptive technique based on classification procedures, which, in turn, are based on "typical" image identification, makes evaluation of the adaptive algorithm difficult. First, one must evaluate the capability and performance of the algorithm. Second, the algorithm sensitivity to the specification of typical image sets should be considered. Thus, an efficient compression algorithm can become suboptimal. The implementation can be significantly degraded if a mismatch develops between the image to be coded and the image which was used for training.

Adaptive techniques with a fixed number of operating modes are attractive. The overall system complexity must include the training procedures and, consequently, it becomes rather involved. An equivalent adaptive technique that eliminates the training procedure is discussed in Section 4.5. Although the source coder complexity increases, the overall system complexity is reasonable. Besides a higher degree of coder adaptivity, the need for training procedures is eliminated.

6. CLASSIFIERS

In this section, the discussion is limited to adaptive techniques which utilize a finite number of classes. For each class, a different coding mode is utilized. The conceptual difference should be noted between a classification process, which is a response to local image structure fluctuation over subblock size, and buffer control. The latter function operates over much larger image regions. It introduces a small perturbation in the coding process to achieve the necessary fixed rate output averaged over many subblocks.

To perform the classification, the coding algorithm performs some measurement and, through an appropriate deterministic procedure, each subblock is assigned to one of the finite number of classes. The considerable amount of research in pattern recognition performed to date has had little or no impact on transform image coding algorithms.

Two types of simple classifiers have been developed. Only one has been used extensively. The first classification process has utilized the subblock energy as primary input to the classifier algorithm. Tasto and Wintz (1971) as well as Chen and Smith (1976) utilized this local energy concept to perform the classification process. In both cases, the local AC energy (e.g., pixel variance subblocks) is deterministically mapped onto a small set of integers representing the finite number of classes.

The energy concept appears reasonable. It is indicative of subblock business. For a unitary transformation, the energy is an invariant parameter. However, energy in the transform domain is primarily represented by a few transform low order coefficients only. Consequently, an energy-based classification procedure does not properly assign the available bandwidth among the various classes, since the relevant decision is magnitude of a few low order coefficients.

A recently proposed classification procedure utilizes the sum of the logarithms of coefficient magnitudes (Melzer, 1977). This quantity can be interpreted to be image entropy. Thus, the classification decision is made on local entropy.

It should be noted that zero order entropy in the transform domain closely approximates image entropy, since the coefficients are nearly decorrelated. For this second procedure, the classifier specifies classes in accordance with local entropy.

Classification includes bit allocation as well. The classifier specifies two sets of matrices, representing the normalization factors for coefficients and the appropriate bit assignment. The latter two quantities can be determined independently through training procedures. For an "n" class classifier, normalization matrices and n bit assignment matrices are required. Alternatively, first the normalization matrices are obtained. The bit assignment matrices are derived from the normalization matrices according to the standard rule (Habibi and Wintz, 1971). The distortion parameter (D) is determined such that the required number of bits is achieved. Specific implementations are considered in Section E.2.

Although the concepts discussed are independently straightforward, the various quantities must be related to each other. The classification must be jointly considered with different buffering procedures.

Classification with subblock energy can be performed prior to transformation. Consequently, it can be performed as part of the controlling function for buffer control using the prebuffering concept.

For classification in the transform domain using local entropy, the same procedure becomes rather involved with the prebuffering concept. The primary problem is to perform buffer control. It is necessary to operate simultaneously on all blocks residing in the prebuffer. Before the controlling function can be implemented, all subblocks must be transformed. Therefore, the requirement develops for another large buffer in which the transform blocks of the prebuffer must be stored.

However, if rate control is achieved by the postbuffering technique, the requirement for the additional large buffer does not exist. Postbuffer control can be equally well implemented through spatial and transform domain classification. For the present discussion, the emphasis is not on arithmetic complexity, but rather on required buffer size.

E. IMPLEMENTATIONS OF ADAPTIVE TRANSFORM CODING SYSTEMS

Previous subsections reviewed basic concepts relevant to transform image coding. Specifically, the required concepts for adaptive transform coding were considered. It should be reemphasized that adaptivity is necessary to justify the additional complexity of transform coding. However, to date, only a limited number of transform coding systems have been studied. In particular, no variable rate transform coding design has been developed to the hardware stage. However, several computer simulated systems have been studied.

In this section, these designs are reviewed. Specifically, fully adaptive transform coding procedures with a rate equalizer buffer are discussed. The emphasis is on implementation concepts. The discussion, here, relates to transform techniques. However, it should be pointed out that the same concepts are general and are also applicable to other coding techniques.

Prior to the discussion of mathematical models for adaptive transform coding, specific definitions of image segmentation are required.

Four image region types are considered. The entire image is referred to as the "frame." The smallest grouping of pixels used is a "block." The block is the basic input to the transform coding algorithm. It corresponds to the pixel set over which the transformation is performed. The "strip" refers to a set of blocks, which, in general, corresponds to n image lines for n × n block size. Transform coding can be implemented with nonsquare blocks. However, to simplify the discussion, only square blocks are considered. The strip also corresponds to the minimum number of sub-blocks required to perform the reformatting function.

The "region" corresponds to a relatively small image area over which the adaptive coder operates without additional control information from the buffer control algorithm. The region size can be a block or several blocks.

These image subelements are utilized for adaptivity, adaptivity control, and local transformation.

1. MEMORY ARCHITECTURES

A transform coding algorithm operates over transform blocks. Thus, the conventional image raster format requires reformatting into blocks. For variable rate adaptivity, in addition to a reformatting buffer, a rate equalizer buffer is included.

Memory architecture associated with the indicated buffers is simple. For application with large size images, these buffers may become important. Image compression with large formats also require large size buffers. While operation complexity remains simple, the large buffers require consideration of expense and reliability.

Reliability is particularly important for the equalizer buffer that contains code words. An error within the rate equalizer buffer is equivalent to channel error.

The buffer that performs the reformatting function, although not truly random access, requires that memory input and output addressing be performed differently. The rate equalizer buffer may be considered schematically as a two-sided memory. At the forward side, the compressor inputs code words. From the other side, the channel, at fixed bit rate, removes code words.

Depending on adaptivity implementation, a control function is associated with one of the two buffers. The information required for adaptivity processing is external to the buffers. For example, the prebuffering concept for adaptivity control requires block energy, which can be determined on input to the memory. Consequently, no memory access is required after the appropriate image area is placed into this buffer.

Similarly, for postbuffer control no requirement exists to access data in the buffer. The buffer control requires the buffer fullness information which, again, is determined externally to the buffer. The buffer fullness is obtained through monitoring the number of bits into the buffer and into the channel. The difference between the two is the buffer fullness required for adaptivity control.

2. SPECIFIC DESIGNS

In this section, several approaches to adaptive transform coding are discussed. Through these techniques, actual implementation is related to theoretical concepts previously developed.

First, discussion is on open loop techniques. An open loop transform coding procedure is a fully adaptive algorithm with no requirement to constrain the bit fluctuation within an equalizer buffer. Consequently, no deterministic procedure maintains the algorithm at a specified bit rate.

Two techniques have been developed previously. Since these techniques cannot easily operate in a standard communications environment that requires fixed rate throughput, they are mainly of interest for historical perspective. However, these techniques also serve as a motivation to develop the required modifications which would permit them to operate at fixed rate.

The first open loop technique, the algorithm of Tasto and Wintz (1971), utilized the K-L transform. Subblocks were partitioned (classified) according to energy. Four classes, each requiring a two-bit overhead, were used. The implemented quantization strategies included a combination of a uniform quantizer with Huffman coding. The paper by Tasto and Wintz strongly motivated this author to study the practical implementations of Huffman coding in the transform domain. Algorithm development has not considered what was earlier referred to as training procedures. Both training and coding were performed on the same image. Consequently, the problem associated with maintaining channel rate from one image to another and the mismatch problem were avoided. The results demonstrated that adaptivity is a viable concept. The Tasto and Wintz paper was the first systematic study of an adaptive transform coding procedure.

Another open loop technique was developed by Tescher (1973). The adaptive coding procedure eliminated the need for classification. The algorithm utilized large transform blocks (256×256). The new concept was to derive recursively, based on previously decoded information, both the normalization terms and the bit assignment.

Superficially, this technique is similar to conventional, two-dimensional differential pulse-code modulation (DPCM). However, unlike DPCM, rather than the actual quantity, its variance is estimated. Thus, the appropriate estimation procedure obtains the standard deviation of the transform coefficient to be coded. Since only previously coded values are used in the estimation, the procedure is fully decodable. The decoder repeats the same steps of the coding process.

The described technique is fully adaptive. It adapts to the transform domain structure. The compression rate is determined on actual image activity. The developed technique is "self-truncating." When the predicted number of bits for a coefficient falls below unity, no further information is transmitted. In that fashion, at that point, and for that appropriate image line in the transform domain, the coding is terminated. This open loop technique for large transform sizes has been successful for various applications including monochrome color and interframe coding.

The two fundamental new concepts established deserve to be emphasized. The first result is that a successful adaptive transform coding may be developed without a priori assumed fixed model, such as the Markov model. More importantly, a procedure has been developed which permits real-time model identification through which the appropriate bit assignment procedure can also be implemented. Through the second concept, an algorithm was developed that utilizes the available bandwidth according to need. Unfortunately, the discussed algorithm is impractical. The utilization of large transform sizes and open loop techniques is inappropriate. The next step was to develop an implementation which is fully adaptive, yet utilizes small transforms. The author extended the early adaptive concepts to more conventional transform sizes.

Basic features of the algorithm are presented in Figure 13. A recursive procedure is utilized. Starting with the largest coefficient, the algorithm recursively predicts the next coefficient standard deviation from which the appropriate number of bits is determined. A simple procedure maps the two-dimensional block into one dimension. As illustrated, the square region is mapped in a zig-zag fashion to yield a one-dimensional function. The subdiagonal elements have approximately the same relative importance. Thus, the mapping results in a monotonically decreasing set of elements. Several examples are shown in Figure 13. Here, the absolute coefficient values on a logarithmic scale are given.

The demonstrated procedure is fully adaptive. The normalization coefficients, the bit assignment, as well as the total number of bits per block, are adaptively determined. The algorithm is "self-terminating." When the predicted number of bits for a coefficient falls below a fixed value, say two, the coding is terminated for that block. Similarly, the receiver duplicates the recursive procedure; thus, the process is decodable.

The outlined procedure appears to be promising. However, its undesirable feature is the open loop operation. The coding algorithm properly

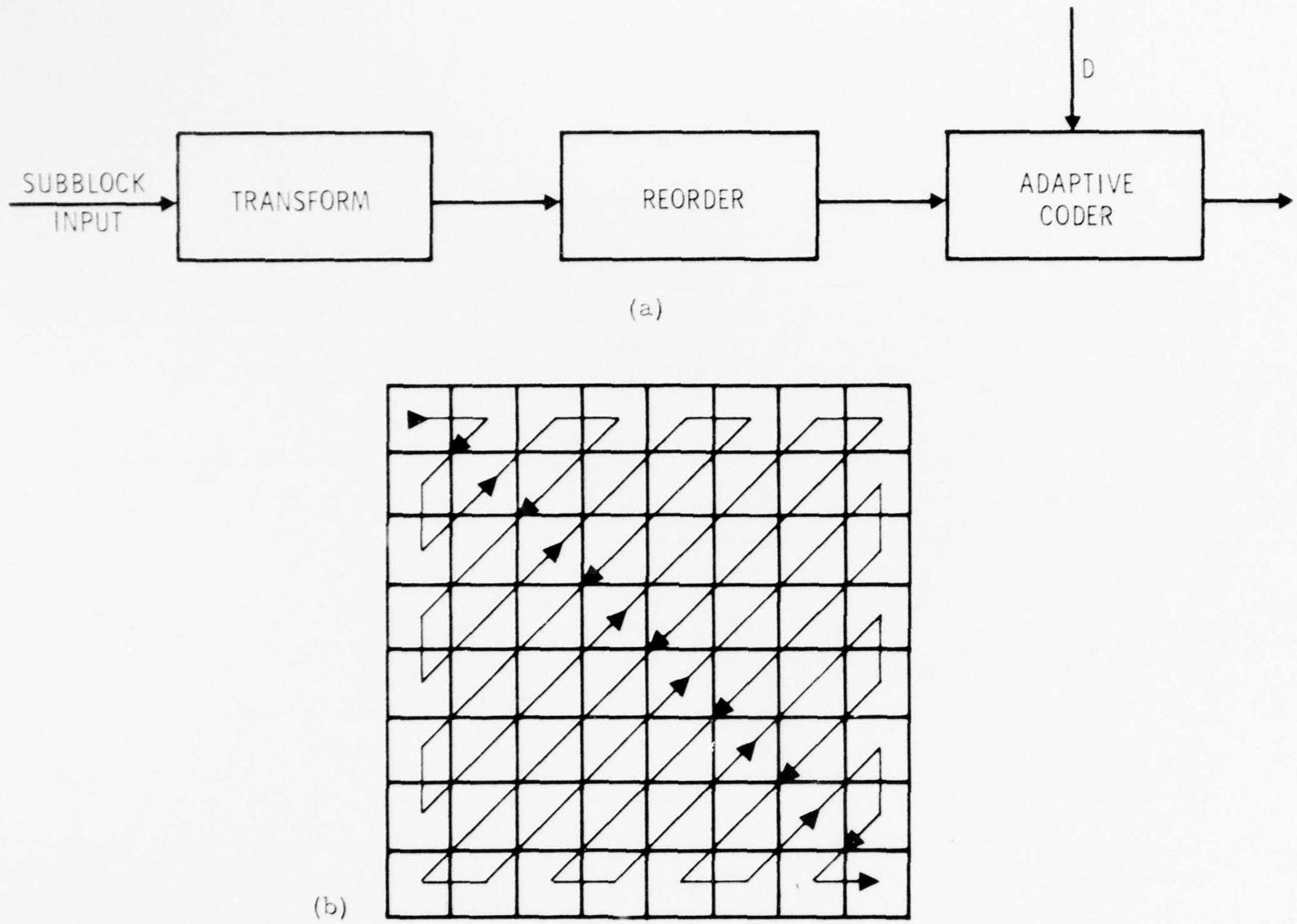
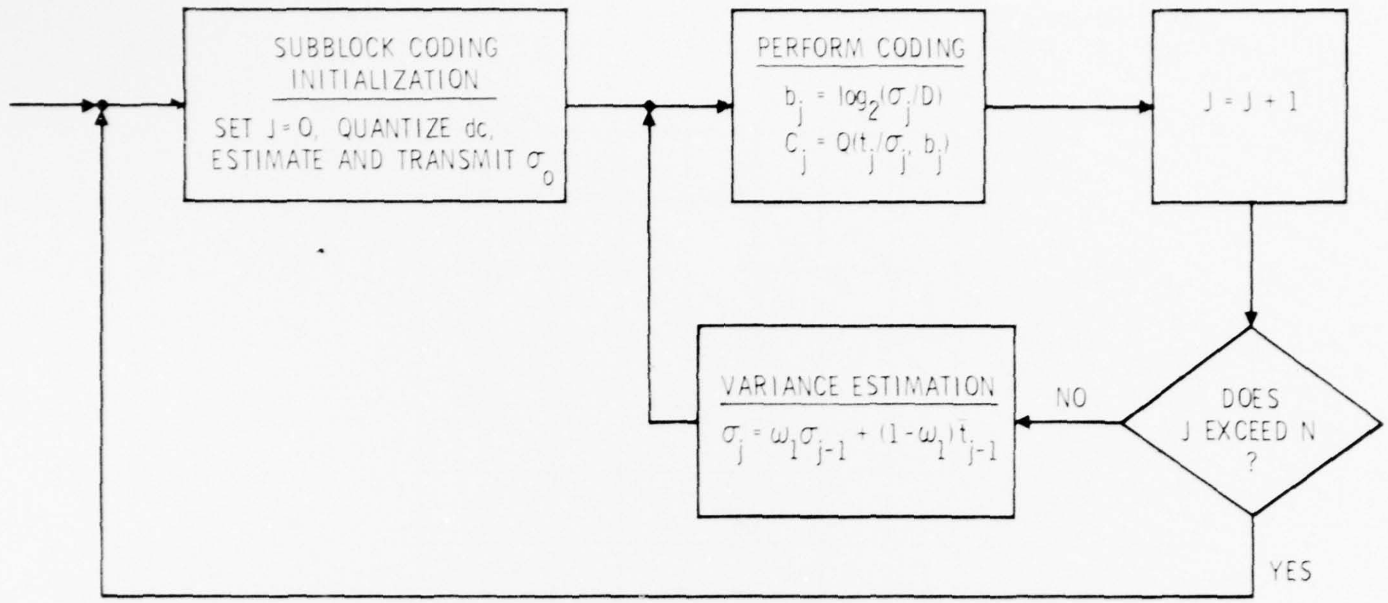


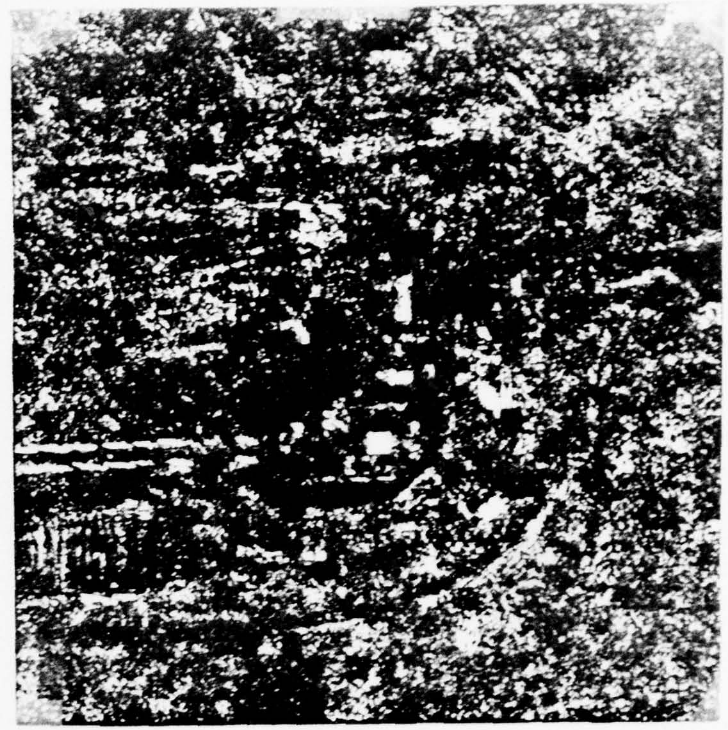
Figure 13. Adaptive Transform Coder. [(a) The basic concept. (b) The logic of the coefficient reordering. The dc term is assumed to be in the upper left corner.]



(c)

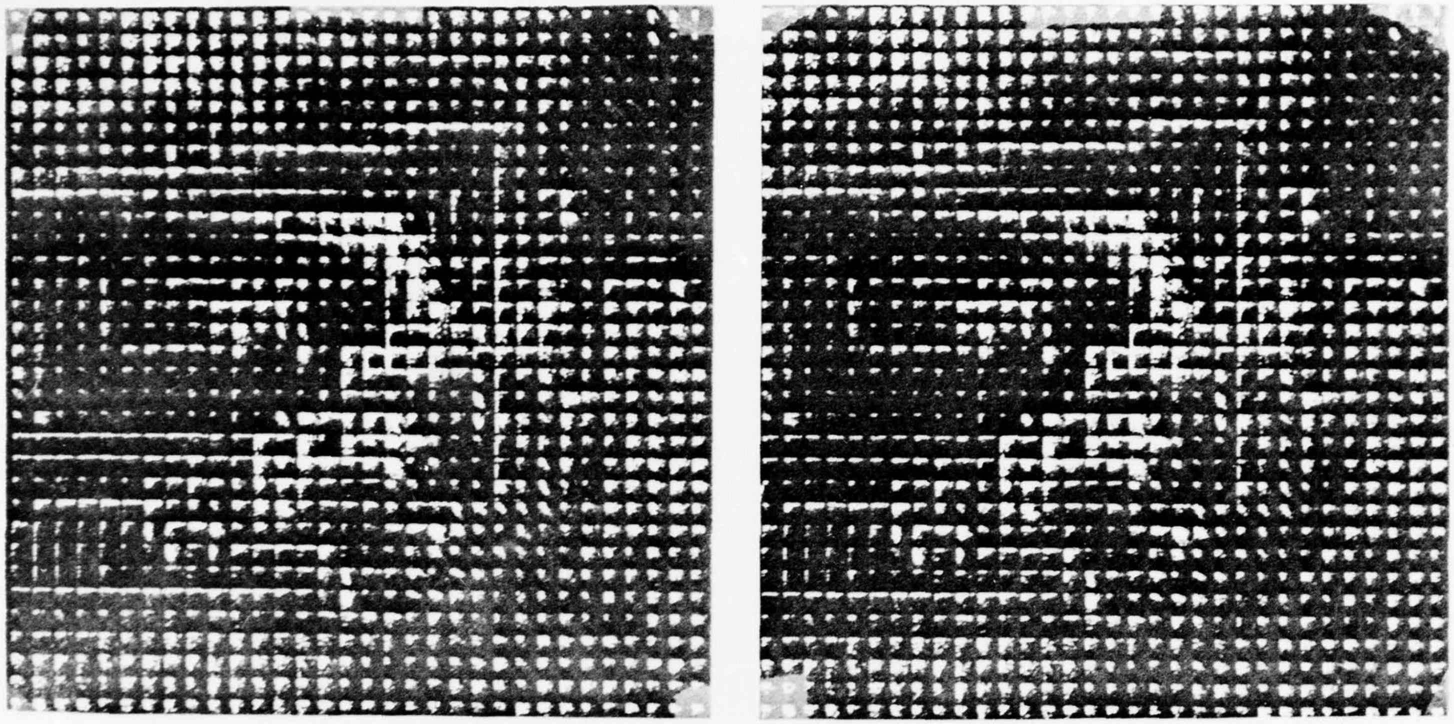
Figure 13. Adaptive Transform Coder (Continued). [(c) The adaptive coding obtained through estimating the variance of the next coefficient to be coded, Tj. Other needed parameters are j, the coefficient index; D, the externally specified distortion parameters; N, the number of coefficients in a sub-block; bj, the dynamically determined number of bits for the j-th coefficient tj; tj, the reconstructed coefficient; Q, the quantizer; and w_1, the estimator control (usually the chosen value is 3/4).]

(d)



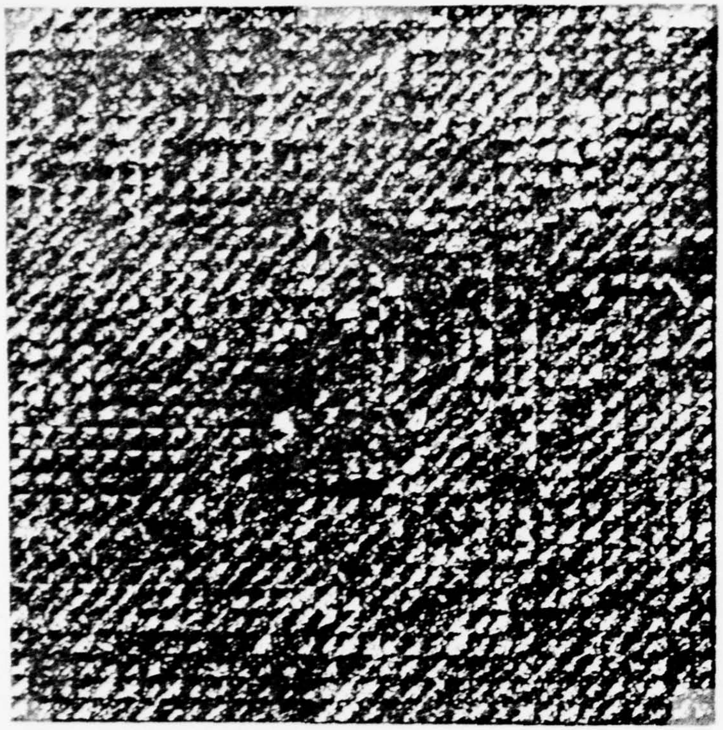
(e)

Figure 13. Adaptive Transform Coder (Continued). [(d) The decoded "site" (2 bpp). (e) The absolute error image (30-fold magnification).]

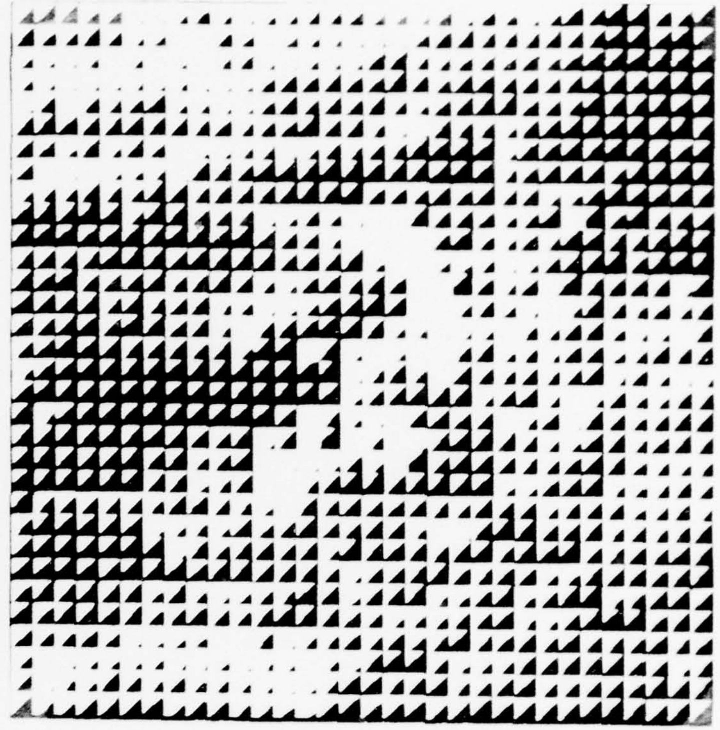


(f) (g)

Figure 13. Adaptive Transform Coder (Continued). [(f) The transform domain display (coefficient magnitudes). (g) The decoded version of (f).]

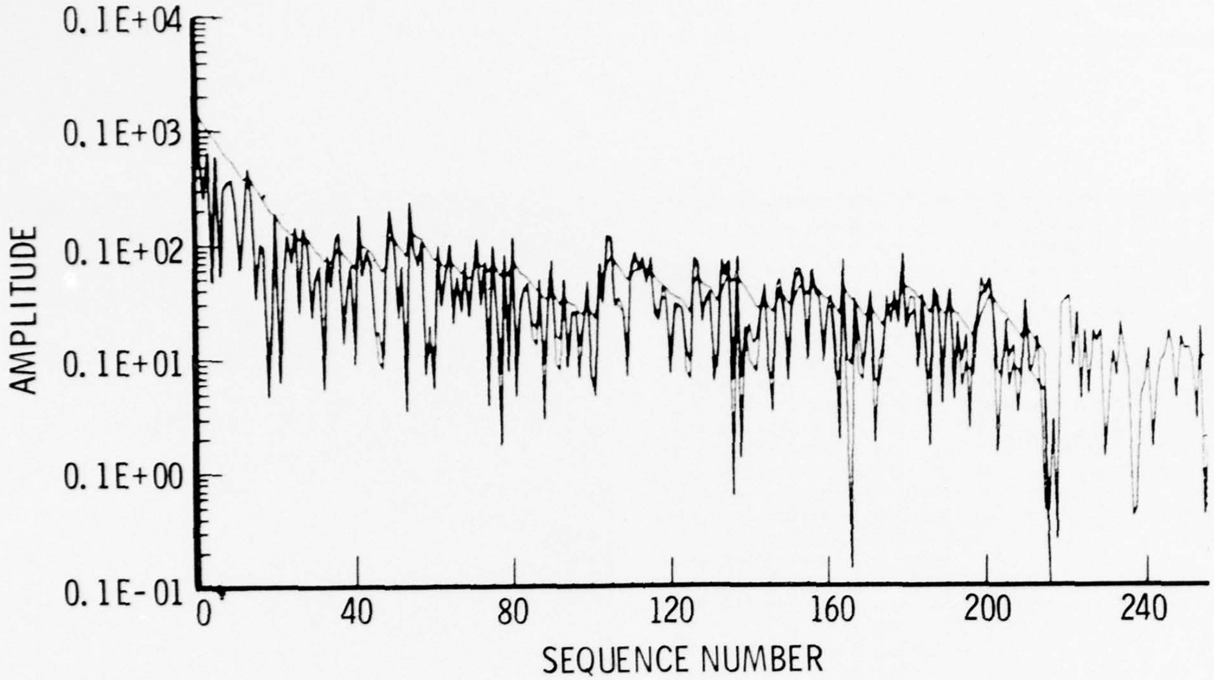


(h)



(i)

Figure 13. Adaptive Transform Coder (Continued). [(h) The transform domain error. (i) The binary display indicating which coefficients are transmitted. The white value indicates that the coefficient has been transmitted. Note the correlation between the number of transmitted coefficients and the local image activity.]



(j)

Figure 13. Adaptive Transform Coder (Continued). [(j) The estimating procedure for a typical subblock. The smooth curve is the variance estimate derived from the previously transmitted values. The decoded values are also shown; however, they are almost indistinguishable from the original.]

allocates bits according to the relative importance (e.g., image entropy) of subblocks. However, channel rate is not considered.

The next logical step is to convert the open loop technique to a closed loop procedure. Through the utilization of the previously discussed general rate control approach, the solution is straightforward. But first, techniques based on classification procedures should be discussed.

Chen and Smith (1976) developed a closed loop adaptive procedure which is a simplification of Tasto's technique. The Chen-Smith coder utilized the cosine transform. Based on subblock energy, a four-class partitioning is performed. Prebuffer control maintains the required bit rate.

The concept is as follows. The coefficient energy of subblocks within the reformatting buffer is measured and classified. The four-class classifier requires three thresholds in subblock energy. For each class, subblocks are given a predetermined number of bits. The distribution function for subblock energy is determined. For the proper thresholds, the product of the given class bit rate and the integral of the distribution function summed over the four classes yields the required rate, as shown in Figure 14. Consequently, the number of bits for the various classes permits straightforward computation of the required thresholds. The necessary computation, although simple, must be performed for each data set that resides in the reformatting buffer. For a real-time system, the computation requires additional buffering.

In their paper, Chen and Smith (1976) effectively argued the benefits of adaptive coding. Through various experiments, they identified a "typical"

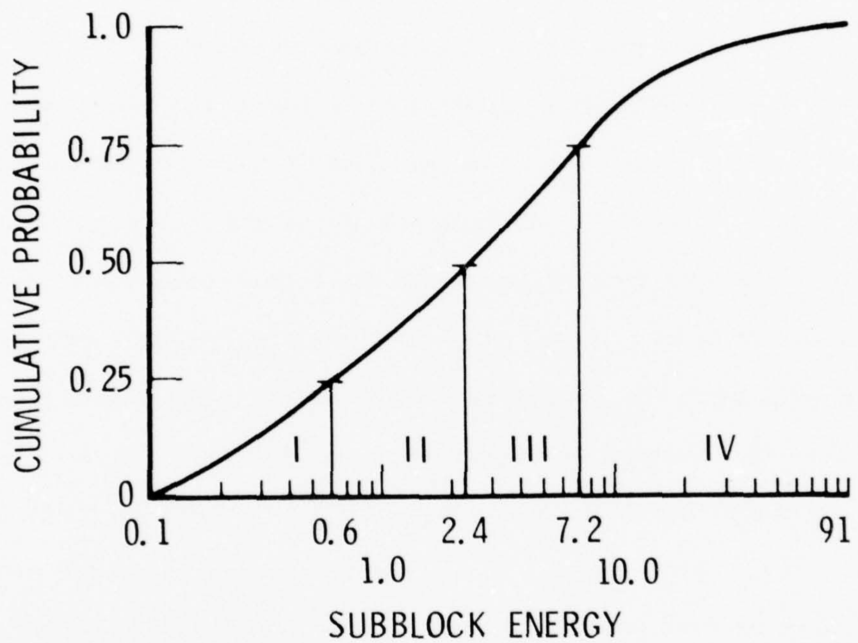


Figure 14. Subblock Classification Based on Energy (Chen and Smith, 1976). (The four indicated regions partition the available bandwidth.)

probability distribution for subblock activity. They observed, consistent with this author's experience, that, for many image types, the indicated probability density is an approximately exponentially decreasing function of energy.

This observation indicates that, in general, most subblocks are relatively quiet. Only a small number of subblocks contain significant activity. Consequently, a strong motivation for adaptivity is available. The coding algorithm at relatively low rates is acceptable for a large fraction of image subblocks. A small additional bandwidth is required to accommodate the small number of active subblocks. The coding experiment of Chen and Smith included monochrome as well as color images.

Unlike Tasto and Wintz (1971), Chen and Smith did not consider entropy coding of the transform domain. Tasto and Wintz utilized the appropriate K-L transform for each class. In contrast, the implementation by Chen and Smith utilized the cosine transform for all classes. The probable benefit of a specific transform for each class is marginal.

The techniques of both Chen and Tasto share a common limitation. The classifier uses floating thresholds to arrive at the fixed rate. Consequently, the energy classification is adaptive for each subblock. How a particular subblock is classified depends on the subblock energy distribution for the image region which is utilized for threshold determination.

Cox and Tescher (1976) developed an absolute classification procedure of the closed loop format. This classifier utilizes fixed preassigned thresholds.

Thus, each subblock is classified according to its absolute energy without regard to the distribution of neighbor subblocks.

To maintain a fixed average channel rate, bit assignment matrices are recalculated for each image strip. Figure 15 indicates the appropriate constraints. Determination of the D parameter is made such that summation over all bit assignment matrices results in the total allowed number of bits for that image segment associated with the adaptivity computation. One could argue the superiority of this technique: an image subblock is paired with the appropriate normalization matrix regardless of the distribution of other subblocks in the adaptivity region.

A major criticism of previous techniques is that classification is based on a single parameter which can provide only limited information on image structure. Subblock entropy utilization should yield superior classification. The problem still remains how to model the normalization matrix for a class.

The next logical approach is to implement the recursive fully adaptive technique over small subblocks with postbuffer feedback control. The result is a self-consistent transform coding procedure. This technique, developed by this author, is discussed next.

Based on the foregoing discussion, the combination of general rate control and the open loop adaptive transform coding is relatively simple. The procedure is to develop a controller that specifies to the previously open loop technique, the D parameter.

In Figure 16, a flow chart is given for the appropriate logic. The same figure lists the parameters necessary to implement the algroithm.

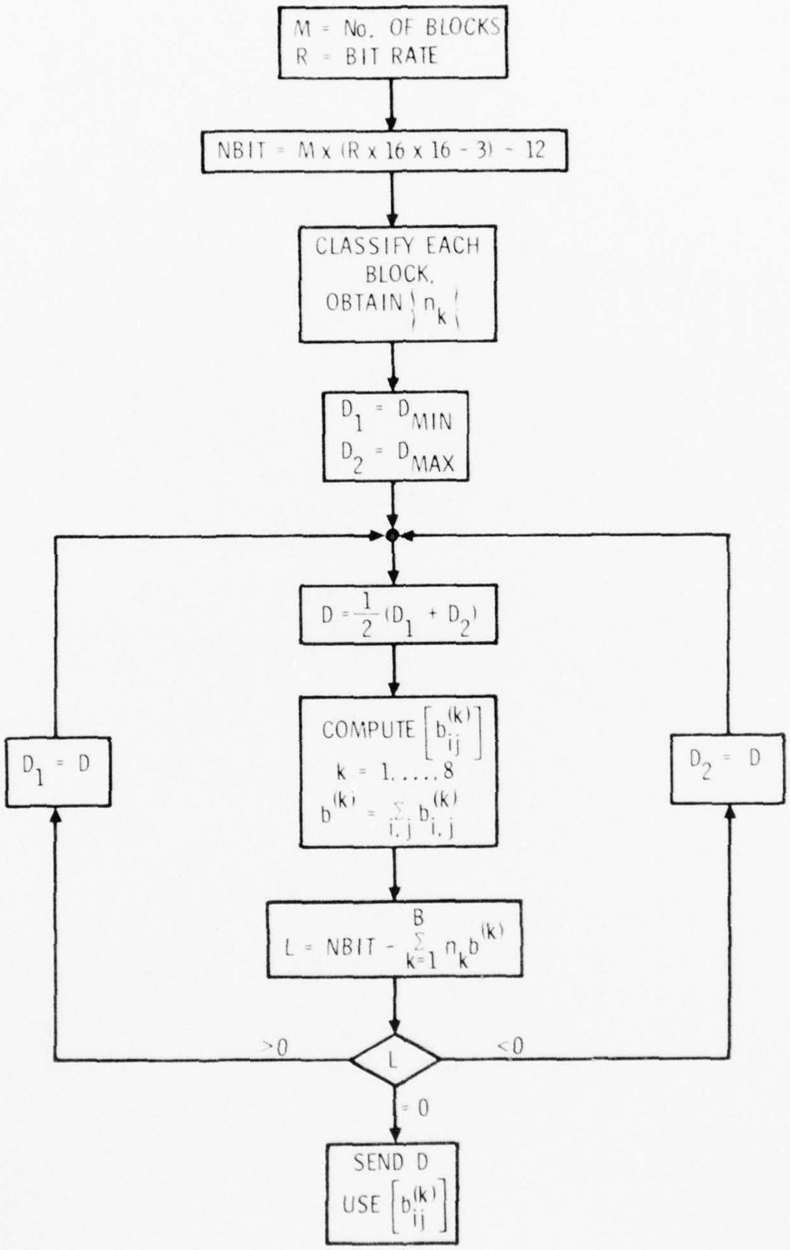
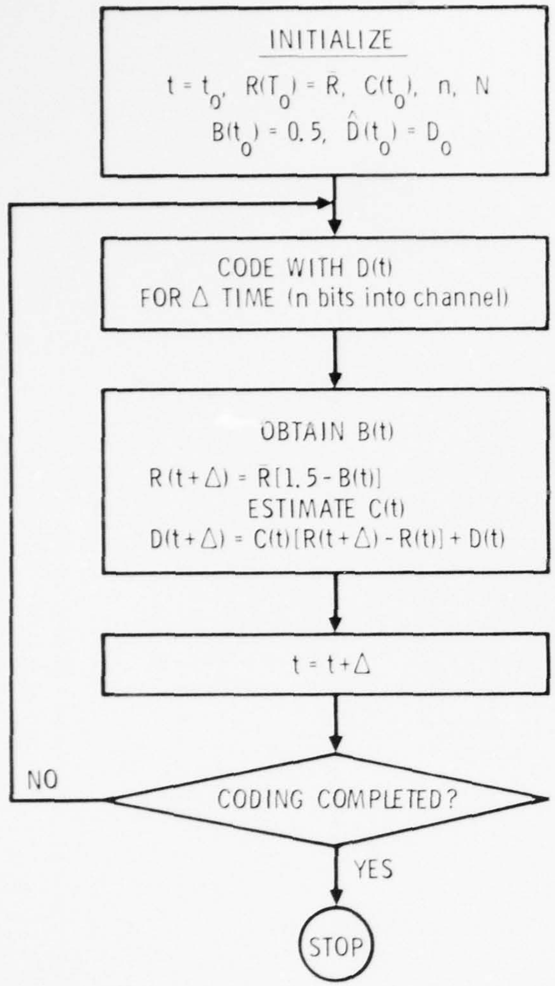


Figure 15. Flow Diagram of an Absolute Classification Procedure (Cox and Tescher, 1976). [The following definitions apply: n_k is the number of subblocks classified in the k-th class according to absolute energy. The algorithm determines dynamically $b_{i,j}{}^{(k)}$, the number of bits for coefficient i, and j in class k.]



(a)

(b)

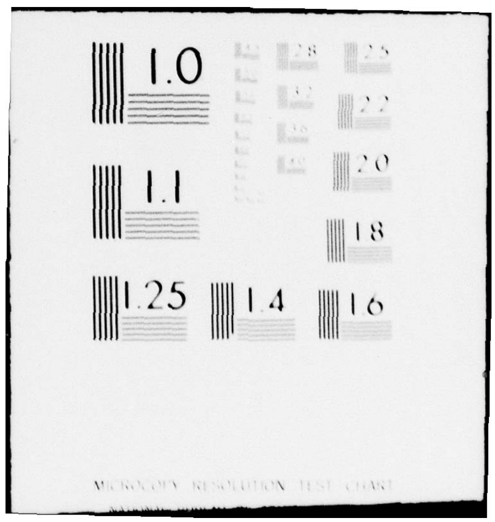
$\bar{R}$	CHANNEL OPERATION RATE
R (t)	REQUIRED CURRENT RATE
DR CURVE	D vs R CURVE
C (t)	SLOPE OF DR CURVE AT TIME t
$\hat{D}$ (t)	ESTIMATE OF D AT TIME t
N	BUFFER SIZE IN BITS
Δ	CYCLE TIME
n	NUMBER OF BITS TRANSMITTED DURING Δ
B (t)	BUFFER STATUS: NUMBER OF BITS WITHIN BUFFER NORMALIZED BY BUFFER SIZE (at nominal rate B = 1/2)

Figure 16. Buffer Feedback Logic. [(a) The basic system. (b) Parameter definitions.]

AD-A061 656 AEROSPACE CORP EL SEGUNDO CALIF ENGINEERING SCIENCE --ETC F/G 9/4
TRANSFORM IMAGE CODING.(U)
MAY 78 A G TESCHER F04701-77-C-0078
UNCLASSIFIED TR-0078(3478-30)-1 SAMSO-TR-78-127 NL

2 OF 2
AD A061 656

END
DATE FILMED
2-79
DDC


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART

The initial condition must be separately specified. For the first subblock to be coded, an assumed distortion (D_o) is specified which corresponds to the desired average rate. The indicated algorithm is stable. Even for unreasonable initial conditions, the controller rapidly stabilizes at an effectively steady state condition.

Conceptually, the controller attempts to maintain the buffer bias (B) within the available buffer. Consequently, the maximum deviation from a specified rate is negligible for image sizes much larger than the postbuffer size.

The postbuffer control size has a limited impact on performance, although it is small. Computer simulations have indicated a small image quality degradation in terms of mean square error as the postbuffer size was reduced from two-image strips to one-fourth of an image strip.

As discussed in Section D.4, this technique can utilize parallel computation, and it is flexible in image format. Various examples, including buffer performance behavior, are shown in Figure 17. It should be noted that the various parameters behave reasonably, and no sign of an unstable buffer behavior develops. The intermediate controller parameters are also shown. They are the "desired" local rate and the estimated and required D parameters. The estimated parameters represent quantities averaged over large image segments. These parameters change only slowly. The instantaneous rate associated with individual subblocks fluctuates significantly. At two bits per pixel, for example, the instantaneous rate variation is likely to exceed a factor of four.

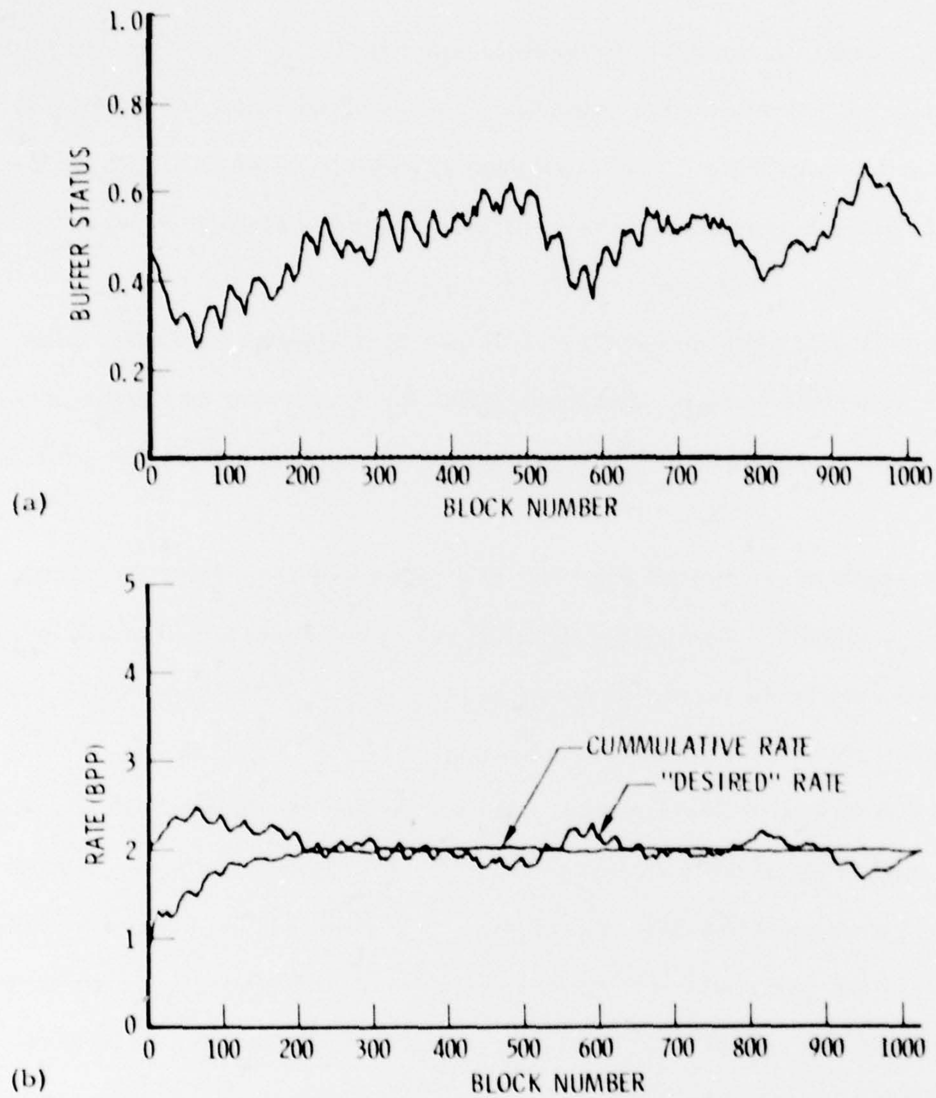
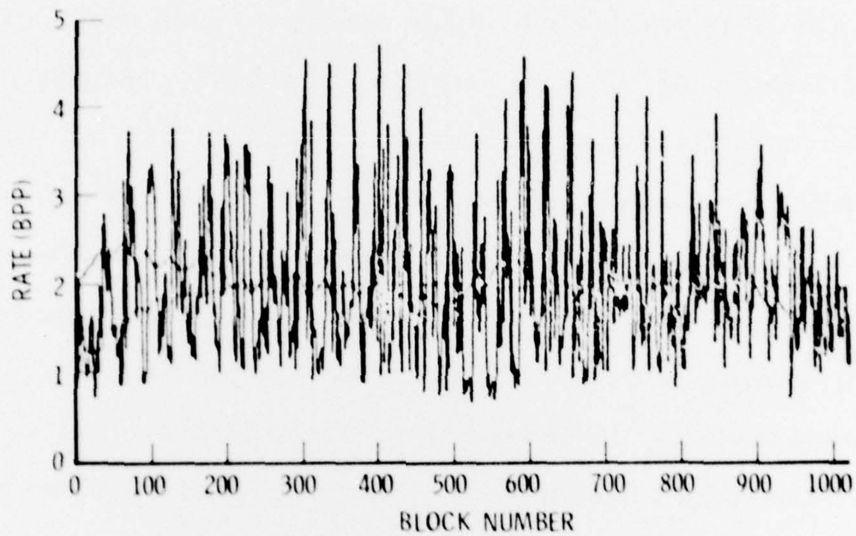
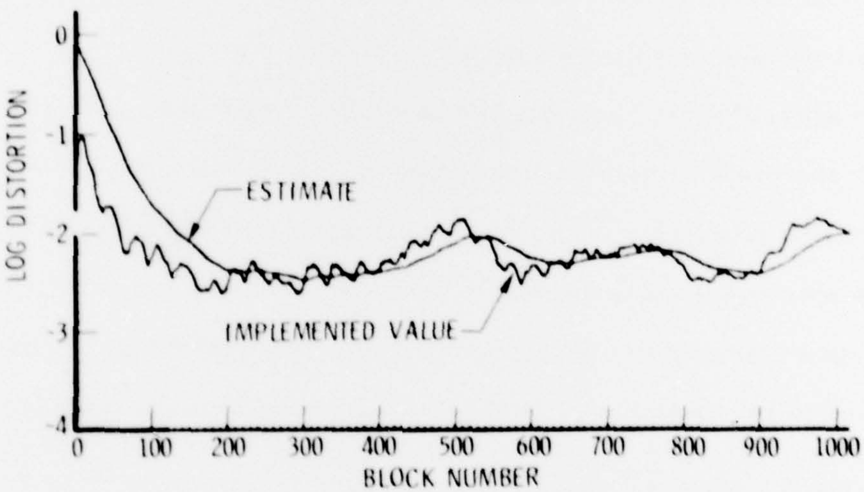


Figure 17. Demonstration of "Typical" Buffer Feedback Behavior. [(a) Normalized buffer status. Here, underflow and overflow would occur at 0 and 1, respectively. (b) The cumulative and "desired" rates; the second parameter is modified to compensate for the developing buffer bias.]



(c)



(d)

Figure 17. Demonstration of "Typical" Buffer Feedback Behavior (Continued). [(c) Superimposed on the curves in (b) is the actual local subblock rate. (d) The two distortion curves represent the local estimate and the specific value needed to produce the desired rate in (b).]

The described procedure is highly adaptive to local image structure and is self-terminating. Yet, the procedure behaves in a manner consistent with a communication channel for practical operation. For a lower average bit rate, while more image subblocks operate at a lower instantaneous rate, the coder still assigns a relatively large number of bits to a selected small set.

This technique is particularly beneficial for high quality imagery with high signal-to-noise properties. For this case, the various image regions are dispersed in a local image structure. The algorithm requires minimal overhead. The coder adapts to each subblock independently. Since possible combinations for different subblocks are essentially infinite, this procedure effectively utilizes an infinite number of classes.

The algorithm can be extended to utilize a two-dimensional predictor. However, the prediction mechanism has to be more complex to follow each transform line of rapidly decaying transform coefficient amplitudes.

A major practical advantage of the self-adaptive transform coding algorithm is its capability to operate without any prior training. Thus, the same algorithm can be equally applicable for significantly different image types.

3. CHANNEL ERROR CONSIDERATIONS

While consideration of the channel error problem is important, it is a separate problem. Except for Chen and Smith (1976), studies of adaptive transform techniques have not considered channel correction procedures. Chen and Smith performed several experiments to demonstrate, not unexpectedly, that additional channel consideration can effectively minimize the channel error problem.

Channel error sensitivity usually is directly proportional to the degree of adaptivity. In principle, a high sensitivity to channel noise may be taken advantage of in the self-adaptive transform coding algorithm. This can be accomplished as follows. A single channel error causes synchronization loss. Thus, subsequent transform values yield erroneous transform variance estimates. Since the estimated values immediately become obviously unrealistic, it is relatively easy to identify an error. A simple algorithm may monitor the estimated variance or the predicted bit assignment to check if the algorithm is working properly.

For a typical subblock, the estimated bit assignment values over small transform domain regions are constant or decreasing; only occasionally does a small increase occur in the allocated number of bits for the next coefficient. Consequently, if the predicted number of bits begins to increase, a channel error is indicated. Once the error is detected, the decoder may simply ignore the remaining part of the subblock. Since the erroneous bit is closely localized, one may attempt to correct the error by trial and error and recover the rest of the subblock. However, this is a conceptual approach yet to be implemented.

4. APPLICABILITY OF CONCEPTS TO OTHER TECHNIQUES

While it is not appropriate for this author to consider image coding procedures other than transform techniques, it is worthwhile to point out that some of the discussed concepts of this section are directly applicable with little or no modifications to other coding techniques (Tescher and Cox, 1977) (e.g., the self-adaptive procedure is applicable to the hybrid transform/DPCM coding). The generalized buffer feedback procedure is applicable to all image coding types as well as to general data compression procedures. In particular, the same feedback logic is useful for entropy coding as well as to procedures where entropy coding is utilized jointly with another coding technique.

F. CONCLUSIONS

This report has attempted to demonstrate that transform coding is a valuable and efficient technique for image compression. However, various considerations are necessary before the transform coding procedure can be used in a practical environment. This report reviews adaptivity concepts followed by considerations of variable rate algorithms with the constraint of the fixed rate channel.

Several solutions are identified. Some of these solutions are also applicable to other data compression algorithms. Adaptive transform coding procedures are promising. These techniques can and should be implemented through a feedback mechanism to control the rate. The necessary hardware will likely exceed what is needed for a nonadaptive technique. However, the benefits will more than offset the hardware penalty.

The required technology is in existence. The discussed channel error correction procedure, based on the detection of catastrophic failure location, is an interesting problem that would probably be worthwhile to pursue.

It should be again emphasized that the most promising problem areas for improved efficiency are not in the development of new transform algorithms but rather in the procedures that follow the transformation.

Several other considerations associated with transform image coding need to be reviewed. Although not explicitly stated, transform coding is a statistical procedure based on the decorrelation property of the transformation. An alternative philosophy based on an approximation theory has been proposed recently. Image regions are considered to be two-dimensional segments that the data compression algorithm approximates by some known function set.

Therefore, one could claim that an approximation with sinusoidal functions may be suboptimal. The relevant question is whether other functional approximations of image segments may converge faster than the sinusoidal set. This question is discussed with reference to the singular value decomposition technique of image segments which, in the least square sense, is optimum (Albert, 1972).

Unfortunately, what may be optimum or attractive in approximation theory does not always result in an efficient coding system. The least squares approach requires the transmission of the transformation parameters as well as of the appropriate coefficients. The required bandwidth is likely to become considerable. For a specific type of imagery, such as artificially generated scenes, an approximation theory may be appropriate (Andrews and Patterson, 1976; McCaughey, 1976). However, most imagery can be characterized only in a statistical sense. Therefore, an approximation theory will not yield improvements over statistical transform techniques.

For completeness, analog transform implementation should be mentioned. The CCD technology allows analog implementation of various transform types (Buss, et al., 1975). This interesting development is beneficial to data processing applications involving large size transforms. For realistic transform coding algorithms, the benefits are marginal, since large transform sizes and the associated high number of arithematic operations are not justified. For specialized implementations with stringent power and weight requirements, an analog approach might be appropriate.

Another area which needs further study and could be potentially useful for transform coding is a better understanding of image quality criteria. The self-adaptive technique approximately performs classification according to transform domain entropy, which is also a measure of image entropy. For large bandwidth compression, transform block classification according to some generalized criteria may be valuable even at the expense of increased distortion. However, it must be remembered that both theoretical and practical limits for an image compression algorithm with negligible distortion are determined by the entropy measure. A highly adaptive transform coding technique closely approximates the actual image entropy.

Future algorithm developments in compression rate reduction are not likely to be dramatic. Most improvements may be accomplished by the implementation of techniques discussed in this report. Considerable effort is still required to ensure that the various designs are implemented through practical hardware.

In this chapter, the primary algorithm components that are necessary for implementation are identified. In addition, it was attempted to interface, at least conceptually, some of these components. What are these components? For transform coding, a transform algorithm is needed. Prior to the transform, a reformatting buffer is required. An important new component was the buffer control algorithm.

Hardware implementation for the discussed techniques could benefit from a possible design that is based on independent components. These components can be independently developed and appropriately interfaced for a particular transform coding algorithm. This modular design, in addition to transform coding, is also applicable to other data compression techniques such as entropy coding through a buffer feedback mechanism.

This author believes that a well-designed adaptive transform coding algorithm represents the most efficient image compression technique for high quality data. Although conceptual design configurations discussed in this chapter are promising, they are still only in the simulation stage and the actual hardware design is yet to be pursued.

REFERENCES

Ahmed, N., Natarajan, T., and Rao, K. (1974), "Discrete Cosine Transform," IEEE Trans. Comput., C-23, 90-93.

Albert, A. (1972), Regression and the Moore-Penrose Pseudoinverse, Academic Press, New York.

Anderson, G. B. and Huang, T. S. (1971), "Piecewise Fourier Transformation for Picture Bandwidth Compression," IEEE Trans. Commun. Technol., COM-19, 133-140.

Andrews, H. C. (1975), "Two-Dimensional Transforms," in Picture Processing and Digital Filtering (Topics in Applied Physics, T. S. Huang, ed.), Vol. 6, Springer-Verlag, New York.

Andrews, H. C. and Patterson, C. L., III (1976), "Singular Value Decompositions and Digital Image Processing," IEEE Trans. Acoust., Speech, and Signal Process., ASSP-24, 26-53.

Bracewell, R. (1965), The Fourier Transform and Its Applications, McGraw-Hill, New York.

Buss, D. D., Brodersen, R. W., Hewes, C. R., and Teasch, A. F., Jr. (1975), "Technology of Charge-Coupled Devices for Video Bandwidth Reduction." In Tescher (1975), 48-56.

Chen, W. and Smith, C. H. (1976), "Adaptive Coding of Color Images Using Cosine Transform," ICCC76, IEEE Catalog Number 76CH1085-0 CSCB, 47-7 through 47-13.

Cooley, J. W. and Tukey, J. W. (1965), "An Algorithm for the Machine Calculation of Complex Fourier Series," J. Math. Computation, 19, 297-301.

Cornsweet, T. N. (1970), Visual Perception, Academic Press, New York.

Cox, R. V. and Tescher, A. G. (1976), "Channel Rate Equalization Techniques for Adaptive Transform Coders." In Tescher (1976), 239-246.

Dail, J. E. (1976), "Polynomial Transform Coding of Image Signals," ICC76, IEEE Catalog Number 76CH1084-0, CSCB, 47-14 through 47-19.

Davisson, L. D. (1972), "Rate Distortion Theory and Applications," Proc. IEEE, 60, 800-808.

Fellgett, P. N. and Linfoot, E. H. (1955), "On the Assessment of Optical Images," Philos. Trans. R. Soc., London, Series A, 247, 369-407.

Fink, D. G. (1955), Color Television Standards, N.T.S.C., McGraw-Hill, New York.

Gallager, R. G. (1968), Information Theory and Reliable Communications, John Wiley and Sons, New York.

Goodman, J. W. (1968), Introduction to Fourier Optics, McGraw-Hill, New York.

Habibi, A. (1971), "Comparison of n-th Order DPCM Encoder with Linear Transformations and Block Quantization Techniques," IEEE Trans. Commun. Technol., COM-19, 948-956.

Habibi, A. (1975a), "Study of On-Board Compression of Earth Resource Date" Tech. Doc. Rep. No. 26566, TRW Systems, Redondo Beach, Calif.

Habibi, A. (1975b), "A Note on the Performance of Memoryless Quantizers," NTC 75, IEEE Catalog Number 75CH 1015-7 CSCB, 38-16 through 38-21.

Habibi, A. and Samulon, A. S. (1975), "Bandwidth Compression of Multispectral Data." In Tescher (1975), 23-35.

Habibi, A. and Wintz, P. (1971), "Image Coding by Linear Transformations and Block Quantizations," IEEE Trans. Commun. Technol., COM-19, 957-972.

Huang, J. J. Y. and Schultheiss, P. M. (1963), "Block Quantization of Correlated Gaussian Random Variables," IEEE Trans. Commun. Syst., CS-11, 289-296.

Huang, T. S. (1971), "Digital Holography," Proc. IEEE, 59, 1335-1346.

Huang, T. S. (1972), "Bandwidth Compression of Optical Images," Progress in Optics (E. Wolf, ed.), Vol. X, 1-44, North-Holland, Amsterdam.

Huang, T. S., Schreiber, W. F., and Tretiak, O. J. (1971), "Image Processing," Proc. IEEE, 59, 1586-1609.

Huang, T. S. and Tretak, O. J., editors (1972), Picture Bandwidth Compression, Gordon and Breach, New York.

Huffman, D. A. (1952), "A Method for the Construction of Minimum Redundancy Codes," Proc. IRE, 40, 1098-1101.

Jain, A. K. (1976), "Some New Techniques in Image Processing." In Wilde (1976), 201-223.

Jones, H. W., Jr. (1976), "A Real-Time Adaptive Hadamard Transform Video Compression." In Tescher (1976), 2-9.

Knauer, S. C. (1975), "Real-Time Video Compression Algorithm for Hadamard Transform Processing," In Tescher (1975), 58-69.

Landau, H. J. and Slepian, D. (1971), "Some Computer Experiments in Picture Processing for Bandwidth Reduction," Bell System Tech. J., 50, 1525-1540.

Levi, L. (1970), "On Image Evaluation and Enhancement," Optica Acta, 17, 59-76.

Lukosz, W. (1962), "Ubertragung Nicht-Negativer Signale Duzch Linear Filter," Optica Acta, 9, 335-364.

Max, J. (1960), "Quantizing for Linear Distortions," IRE Trans. Info. Theory, IT-6, 7-12.

McCaughey, D. G. (1976), "Variable Knot Splines as an Image Analysis Technique," In Wilde (1976), 168-175.

Melzer, S. (1977), "Image Transform Coding with Generalized Correlation Models." In "Sensor Data Characteristics and Data Compression Projects of Future Military Satellite Systems" (R. G. Nishinaga, Ed.), Tech. Doc. Rep. No. TOR-0077(2060)-1, App. C, The Aerospace Corporation, El Segundo, Calif.

O'Neal, J. B. (1971), "Entropy Coding in Speech and Television Differential PCM Systems, "IEEE Trans. Inf. Theory, 66-71.

Panther, P. F. and Dite, W. (1951), "Quantization Distortion in Pulse-Code Modulation with Nonuniform Spacing of Levels," Proc. IRE, 39, Papoulis, A. (1965), Probability Random Variables and Stochastic Processes, McGraw-Hill, New York.

Pearson, D. E. (1975), Transmission and Display of Pictorial Information, John Wiley and Sons, New York.

Pratt, W., Chen, W., and Welch, L. (1974), "Slant Transform Image Coding," IEEE Trans. Commun. Technol., COM-22, 1075-1093.

Reader, C. (1975), "Intraframe and Interframe Adaptive Transform Coding," In Tescher (1975), 108-118.

Reis, J. J., Lynch, R. T., Butman, J. (1976), "Adaptive Haar Transform Video Bandwidth Reduction System for RPV's." In Tescher (1976), 24-35.

Roberts, L. G. (1962), "Picture Coding Using Pseudo-Random Noise," IRE Trans. Info. Theory, IT-8, 145-158.

Roese, J. A. and Pratt, W. K., "Theoretical Performance Models for Interframe Transform and Hybrid Transform/DPCM Coders." In Tescher (1976), 172-179.

Schreiber, W. F. (1967), "Picture Coding," Proc. IEEE, 55, 320-330.

Shannon, C. E. (1948), "A Mathematical Theory of Communications," Bell System Tech. J., 27, 379-423 and 623-656.

Shannon, C. E. and Weaver, W. (1949), The Mathematical Theory of Communications, University of Illinois Press, Ill.

Tasto, M. and Wintz, P. (1971), "Image Coding by Adaptive Block Quantizations," IEEE Trans. Commun. Technol., COM-19, 957-972.

Tescher, A. G. (1973), "The Role of Phase in Adaptive Image Coding," PhD Dissertation, U. of Southern California, USCIPT Report 510.

Tescher, A. G. (1975), editor, Efficient Transmission of Pictorial Information, SPIE Proc. 66, Bellingham, Washington.

Tescher, A. G. (1976), editor, Advances in Image Transmission Techniques, SPIE Proc. 87, Bellingham, Washington.

Tescher, A. G. and Andrews, H. C. (1974), "The Role of Adaptive Phase Coding in Two- and Three-Dimensional Fourier and Walsh Image Compression," in Applications of Walsh Functions and Sequency Theory (H. Schreiber and G. F. Sandy, editors), IEEE Cat. No. 74CH0861-SEMC.

Tescher, A. G. and Cox, R. V. (1976), "An Adaptive Transform Coding Algorithm," ICC76, IEEE Catalog Number 76CH1085-0 CSCB, 47-20 through 47-25.

Tescher, A. G. and Cox, R. V. (1977), "Image Coding: Variable Rate DPCM through Fixed Rate Channel," In Tescher (1977).

Whitehouse, H. J., Means, R. W., and Wrench, E. H. (1975), "Real-Time Television Image Bandwidth Reduction using Charge Transfer Devices." In Tescher (1975), 36-47.

Wilde, O. C. and Barrett, E., editors (1976), Image Science Mathematics, Western Periodicals Company, North Hollywood, Calif.

Wintz, P. A. (1972), "Transform Picture Coding," Proc. IEEE, 60, 809-820.