DIGITAL IMAGE COMPRESSION

A Thesis presented by

Said Abdul-Amir

for the

DEGREE OF DOCTOR OF PHILOSOPHY

of the

COUNCIL FOR NATIONAL ACADEMIC AWARDS

June 1985

School of Electronic

and Electrical Engineering.

LEICESTER POLYTECHNIC

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisors, Dr. A.A. Hashim for his guidance advice and assistance throughout the research programme, and Dr. B. Rigg for his valuable help. The University of Technology, Baghdad provided the scholarship which allowed me to come to Leicester Polytechnic to study.

My wife has shown considerable patience and I thank her for her support during the research programme.

Also special thanks must be given to A.D. Clark for his valuable discussion and assistance, to S. Forrest and D. Raffe for their help and advice on the computer system and on programming, to J. Cazot and his wife for typing the thesis and drawing the illustrations and finally to the members of the Digital Signal Processing Research Group for their support.

DIGITAL IMAGE COMPRESSION

S.Amir, Dip. Ing.

ABSTRACT

Due to the rapid growth in information handling and transmission, there is a serious demand for more efficient data compression schemes.

Compression schemes address themselves to speech, visual and alphanumeric coded data. This thesis is concerned with the compression of visual data given in the form of still or moving pictures. Such data is highly correlated spatially and in the context domain.

A detailed study of some existing data compression systems is presented, in particular, the performance of DPCM was analysed by computer simulation, and the results examined both subjectively and objectively. The adaptive form of the prediction encoder is discussed and two new algorithms proposed, which increase the definition of the compressed image and reduce the overall mean square error.

Two novel systems are proposed for image compression. The first is a bit plane image coding system based on a hierarchic quadtree structure in a transmission domain, using the Hadamard transform as a kernel. Good compression has been achieved from this scheme, particularly for images with low detail.

The second scheme uses a learning automata to predict the probability distribution of the grey levels of an image related to its spatial context and position. An optimal reward/punishment function is proposed such that the automata converges to its steady state within 4000 iterations . Such a high speed of convergence together with Huffman coding results in efficient compression for images and is shown to be applicable to other types of data.

The performance and evaluation of all the proposed systems have been tested by computer simulation and the results presented both quantitavely and qualitatively. The advantages and disadvantages of each system are discussed and suggestions for improvement given.

CONTENTS PAGE

cont.

cont.

cont.

CHAPTER ONE

1 Image Data Compression Techniques :Introduction and Review

1.1 Introduction

With the continuing growth of modern communication technology, demands for image transmission and storage is increasing rapidly.

Advances in computer technology for mass storage and digital processing have paved the way for implementing advanced data compression techniques to improve the efficiency of transmission and storage of images.

Image data compression is concerned with minimization of the number of information carrying units used to represent an image. For digital image transmission and storage, the conventional methods is to use the pulse code modulation (PCM) technique. The continuous image is first sampled at Nyquist rate in the spatial domain to produce an N x N array of discrete samples. Sampling of a band-limited image signal is the simplest and most dramatic forms of data compression.

The samples thus obtained may have an infinite number of amplitude levels and hence may require infinite bandwidth for transmission. Therefore each image sample,

also called pel or pixel, must be represented by a finite number of levels 2^K (where K is the number of bits per sample) in order to transmit them over a digital channel. Normally the number of quantization levels in PCM is 64 or 128 corrosponding to 6 or 7 bits respectively [1,2]. Thus the PCM technique requires KN^2 bits per image. This needs a large bandwidth for image transmission, or large storage capacity in order to store the image for future retrieval and analysis. There is also degradation in subjective picture quality due to quantization errors which becomes perceptible when K is reduced to six or fewer bits per pixel. Therefore one needs an alternative approach to solve this problem, i.e. to keep the number of bits per pixel to a minimum, at the same time keeping quantization errors within tolerable limits.

Normally an image source is very highly correlated both spatially and temporally, there is a strong dependency among the values of individual picture elements (pixels). The dependency can be regarded as statistical redundancy. Taking the pixel correlation into consideration, will reduce the bits needed for representing each pixel. Measurement of second and third-order amplitude probabitity distribution and of auto-correlation functions carried by Schieiber [3] and Kretzmer [4] indicated an image entropy of two to three bits per sample. Moreover, if the images are to be viewed by human observers, then there is a psychovisual redundancy, because of the perceptual limitations of human vision.

Hence, the objective of any efficient method of image data compression is to remove the redundancy from the image without much degradation in subjective picture quality.

Picture quality does not depend on the compression method only, but also on the quantization strategy employed with this method.

Two important classes of compression schemes that make use of the statistical redundancy in the image are the linear transform coding and linear predictive coding, whereas the psychovisual redundancy can be exploited by using an appropriate quantization technique. There are other schemes, which lie more or less within the above classes.

The remaining material of this chapter contains five sections. The first four sections give a brief survey of transform coding, predictive coding, hybrid coding, and coding of non-stationary images. The final section gives the outline of the reserch direction of this thesis and underline the objective of this program of research.

1.2 Transform Coding Technique

Although the transform techniques were known a long time ago, they were first used in image coding in the late 1960's and early 1970's.

Initial concepts were based on Fourier transform

(FT) by Andrews and Pratt [5,7] and by Anderson and Huang [6], the Hadamard transform (HT) by Enomoto and Shibata [8], by Pratt et. al. [9] and by Wood and Huang [10], the Karhunen-Loeve transform (KLT) by Tasto and Wintz [11] and by Habibi and Wintz [12] and Haar transform by Haar [13]. Thereafter, a new tranform known as Slant transform, specialy designed for image coding was developed by Enomoto and Shibata [14] for data of vector lengths of four and eight. Pratt et.al. [15,16] developed a generalized Slant transform algorithm for larger sizes.

Implicit in all transform coding procedures, the image is divided into non-overlaping blocks called subimages, and the statistically dependent elements of each block are linearly transformed into a new set of desirably independent coefficients using some unitary transform matrices [17]. This transformation also results in compaction of the image energy into fewer coefficients [18], but, because of the orthogonality of the transform matrix, the block total energy in transform domain is equal to that in pel domain (spatial domain) [19].

The coefficients with low energy or minimum variance are discarded without seriously effecting the statistical information content of the output image

The remaining coefficients are quantized, coded and transmitted to the receiver. At the receiver the code words are decoded and inverse transformed to give the output image. The discarded coefficients are assumed to be zero at the receiver.

Fig 1.2.1 shows the block diagram of the transmitter and the receiver for transform techniques

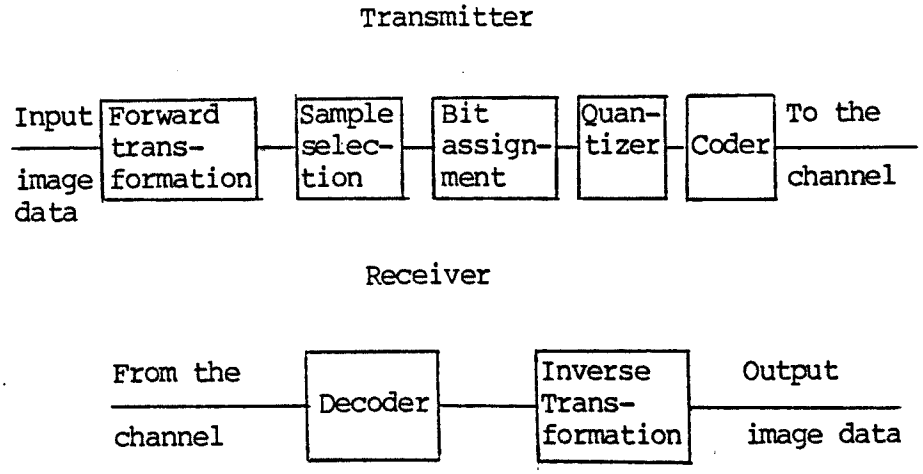


Fig 1.2.1 Block diagram of transform coding

Apart from quantization error, Wintz [18] and, Wintz and Kurtenbach [20] show that the mean-square error of the resultant image is the sum of the variances of the discarded coefficients.

The optimum transformation would be one that results in statistically independent coefficients and minimum mean-square error, but this requires knowledge of higher order statistics of the image. Although Shreiber [3] measured a few third order statistics, only first and second moment can be measured in detail. Furthermore, even if higher order statistics were known, the problem of determining a reversible transformation that results in independent coeffients remains unsolved.

1.2.1 Karhunen-Loeve Transform [KLT]

The best transform which is close to the optimum is the Karhunen-Loeve transform (KLT) [21], which was derived primarily for sampling an anologue signal. Brown [22] showed that for second order processes, using the expansion of Karhunen-Loeve, minimizes the sampling error.

However the discrete KL-transform as developed by Hotelling [23] can be used on an already digitized images to obtain uncorrelated coefficients or samples.

Although KL-transform has been known for some time, its use for the problem of information transmission was made much later [24,25]. Kramer and Mathews [24] applied KL transform for speech signals having assumed Gaussian distribution. Huang and Schultheis [25] developed an optimum block quantization algorithm for assigning binary digits to transform coefficients.

Performance results were obtained by Wintz and Kurtenbach [20] for stationary Gaussian Markoff models, by Pratt and Andrews [26] and by Habibi and Wintz [12] for pictorial data.

Although the KL-transform is optimum, its use in practice presents many problems. It requires statistical knowledge of the image source and does not posses a fast computational algorithm [27] and in many cases its covariance matrix is a singular [28]. Habibi and Wintz [12] have given a detailed discussion of the two - dimenssional KL-transform and the difficulties associated with

it. Jain [29,30] developed a fast KL- transform for finite first order Gauss-Markoff signals with known boundary values and he showed that the assumption of known boundary, replaces the non-periodic sine wave representation of KL-transform by a periodic sine wave and this leads to fast transform via sine or Fourier transform [31]. Also Haralic et al. [32] reported that under isotropicity condition, a fast KL-transform exists, which differs from the optimum KL by approximately 1%.

All these methods are only approximations and only valid for data from stationary Markoff processes with exponential correlation.

1.2.2 Discrete Fourier Transform (DFT)

Many simpler sub-optimum transforms have been developed whose performance are very close to that of KL-transform and are computationally easier. One of them is the discrete Fourier transform.

The development of the fast Fourier transform algorithms [33,34] has led to the investigation of the Fourier transform of image coding technique, even though, the discrete Fourier transform has long been used for signal analysis [35].

The concept of coding and transmitting of an image using two-dimentional Fourier transform was introduced by Andrews and Pratt [5,7] and Anderson and Huang

[6,36]. Andrews and Pratt [5,7] used the Fourier transform on complete images, where Anderson and Huang [6,36] divided the image in blocks of size 16 x 16 and used the Fourier transform on the blocks. The drawback of the Fourier transform is the complex arithmetic which it involves.

1.2.3 Hadamard Transform (HT)

Other sub-optimum transform is the Hadamard transform. This is the simplest to implement since the Hadamard matrix consists of $\pm$ 1's , and, therefore only additions and no multiplications are required. Bowyer [37] displayed the similarity between the Hadamard matrix and discrete Fourier transform.

Pratt et al [9] and Wood and Huang [10] recognized that the Hadamard transform could be utilized in place of Fourier transform with a considerable decrease in computational requirements. Pratt et. al. [9] transformed the entire picture as a unit. Wood and Huang used it on blocks of 4x4, 8x8, 16x16 and entire image. Habibi and Wintz [38] applied Hadamard transform on 16x16 and larger blocks.

1.2.4 Other Transforms

Other orthogonal transformation used in image coding and have performance close to KL-transform are Cosine transform [39,40], Haar transform [13,41], Slant transform [14,15,16,42] and the discrete linear basis (DLB) [43]. The DLB as developed by Haralic and Shanmugam [43], offers a good trade-off between the complex KL--transform and simple Hadamard transforms. This transform is very close to KL-transform for high compression rate and much better than Hadamard transform for all compression rates [43].

1.2.5 Block Size Consideration

Other consideration of the transform algorithms is the block or subpicture size (n) [44].

Large blocks size reduces the correlation between blocks (interblock correlation) and improves the mean-square error performance, since the number of correlations taken into account increases with n, [45]. However, for implementational simplicity as well as to adapt to the local changes in picture statistics and visual fidelity, a smaller size is desirable.

On the other hand, if n is too small, correlation between picture elements are not taken into account. Moreover, if adaptive coding is used, the overhead informa-

tion will increase, which results in a higher bit rate, when the size is reduced [46].

The block size therefore, should be chosen greater than but comparable to the interpixel correlation distance, which defined as the distance at which interpixel correlation becomes small enough to be assumed zero [47].

Claire, et al.[48] showed that, if the block size is chosen according to the conception above, then the block mean-square error performance should be comparable to the entire image transform.

However, most pictures contain significant correlation between pixels for only about 20 adjacent samples [49], although this number is strongly dependent on the amount of detail in the picture [12,Fig.2]. Pratt [45,Fig.23.2-4] has plotted the mean-square error of an image having a Markoff process covariance, as a function of block size for various transformations. The figure shows that the improvment is not significant as subpicture size exceeds 16x16. In fact using blocks of nxn = 8 x 8 does not significantly increase the error. Furthermore it shows that discrete cosine transform (DCT) has virtually the same energy compactness as the discrete KL-transform.

Tasto and Wintz [46,Fig.7] illustrated the relation between the bit rate with the block size, which indicates that the bit rate will increase as the size decreases.

The optimum size appears to be 16 for one dimensional blocks, and between 6 x 6 and 8 x 8 for two dimensional blocks [46]. However the subjective quality does not appear to improve with the size of the block beyond 4 x 4 [18].

Tasto and Wintz [50] showed that, even though a two dimensional block yields better performance than a one-dimensional block, the improvment is rather small - about 0.2 bit/pel. Sakrison and Algazi [51] analytical result based on rate distortion considerations, shows, that for a fixed distortion no more than a factor of 2 or 3 can be saved in the number of bits required for optimal two-dimensional as opposed to line by line processing.

On the other hand Saghri [52] showed that, at rates lower than 1 bit / pixel, the perfomance of an adaptive transform coding will improve with larger block size, and at higher bit rate the trend is reversed.

The block transform decorrelates only the pixels within the block and does not decorrelate pixels among the blocks. 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 blocks. If the interblock correlation is ignored, then the reconstructed images tend to take on a smooth-blurred appearance with edge lines (blocking effects) occuring between adjacent blocks, particularly at high compression factor.

Haralick et. al. [53] designed a transform method

for reducing the blocking-effect and called it " Annihilation transform ". They reported that this method has less blocking effect than the cosine transform at compression of 10.

1.2.6 Sample Selection in Transfomrm Domain

The next step in transform coding is the selection of the coefficient to be transmitted. One method is to evaluate the coefficients variances on a set of average pictures and then discard all coefficients whose variances is lower than a certain value. Such scheme is called " Zonal Sampling " and it is a non-adaptive technique.

In most scenes of interest the energy in transform domain tends to be clustered at the low spatial frequency coefficients [9] and most of the image energy is contained in a few samples [54]. Normally the high spatial frequency coefficients are discarded, which is equivalent to low pass filtering.

Coding degradation can be large if the image contains large amptitude of high frequency coefficients.

Landau and Slepian [28] found that for 4 x 4 Hadamard transform, very little degradation is seen for most pictures by discarding the coefficients 11,12,—,16. Haralic et. al. [43] selected the first four Hadamard and DLB transform coefficients with sequency (0,0), (1,0),

(0,1), (1,1), which have the most image energy.

Another sample selection method would be to first evaluate all n coefficients and then retain only those coefficients that exceed a preset threshold. This method is called " threshold sampling " [55] and it is an adaptive technique that retains only those coefficients that are large for a particular picture and block being processed. When threshold sampling is used, " book-keeping information" must be transmitted, which specifies the coefficients used.

1.2.7 Quantization of Transformed Coefficients

After selection the coefficients which have to be transmitted, each coefficient must be quantized and coded.

One criteria for designing the quantizer is based on minimizing the mean square error between the quantizer input and output [56-59].

Panter and Dite [56] considered signals whose probability density is an even function and that, is zero outside the interval (-v, v) which represent the range of the quantizer input. They found that for optimum performance, output levels (reconstruction levels) should be the midway between two adjacent decision levels.

Max [57] used differential calculus to derive the relation between input and output levels for Gaussian

signals.

Peaz and Glisson [58] used Max procedure to determine the optimum quantizer characteristics for Laplacian and Gamma distributed signals.

Reo [59] gave approximated solution for a more general case of probability distributions. Panter and Dite [56] and Max [57] found that if the probability density funtion of the input signal is uniform, then a uniform quantizer (uniform spaced output levels) is optimum. For other distributions the mean-square error can be decreased by using non uniform quantizer with small spacing in regions of higher probability and large spacing in regions of lower probability. All references listed above are dealing with single sample quantizer.

Quantization stratagies for minimizing the total meansquare error of block of samples (random variables) have been developed by Huang and Schultheiss [25], where a knowlege of the variances is essential for the optimization.

The mean square error is minimum if a vector X built from Gaussian random variables is transformed to uncorrelated vector Y by an orthogonal apertor and Y is quantized.

The best choice of the number of bits assigned to each variable is that the quantization error is the same for all variables. This is possible if the bit assignment is made propotional to the logarithms of their variances [60]. This technique is called " block quantization " and

it is significantly more efficient than using the same number of bits for all samples. Normally the number of bits assigned to each variable is not an integer, therefore a correcting process is necessary to assign an integral number of bits to each sample, which leads to deviation from optimality.

Mitrakos and Constantinides [61] presented a recursive procedure for optimum block quantization, based on dynamic programming, by means of which integer bit assignment constraints are easily met.

Image subjective quality can be improved by assigning more bits to coefficients with the larger variances and fewer to the coefficients with smaller variances [12].

The subjective quality can also be improved by using quantizers with characteristics different from that which minimize the mean square error.

Landau and Slepian [28] used Hadamard transformation with a 4x4 block. The number of quantization levels they assigned to each of the first ten coefficients (H1 to H10) was approximately proportional to the variance of that coefficient, whereas the last six coefficients (H11-H16) were dropped.

The first coefficient was quantized by a 64 - levels uniform quantizer. Coefficients H2 through H10 were quantized with quantizers having a companding characteristic given by a function of the form $y=k\ x$.

Tasto and Wintz [11] proposed an encoder using a

6 x 6 adaptive Karhunen-Loeve transformation. They first determined the number of coefficients that must be retained, then by trial and error they determined the quantizers characteristics for each coefficient, which result in best picture quality

Mounts et al. [62] described a systematic procedure for designing of optimum quantizer for Hadamard coefficients based on psychovisual criteria in the transform domain. Based on subjective tests, they evaluated the visibility funtion of quantization noise, then they developed a design procedure to minimize the mean-square subjective noise by replacing the coefficients visibility function in place of the probability density function in Max quantizer.

1.2.8 Adaptive Transform Coding

A nonadaptive algorithm is designed to be a fixed coding algorithm operating identically for all images and all image blocks. For a nonadaptive technique, the image to be coded is assumed to be a stationary source. Images for which the stationarity assumption is valid, a nonadaptive coder could be an optimal coder.

Usually, images have varying statistical structures, both from image to image as well as from region to region within an image. Breaking the image up into blocks of size n x n and calculating the mean and autocorrelation function within each block yields means and auto

correlation functions which change from one block to another. This fact reduces significantally the efficiency of nonadaptive coding and it is the prime reason for the development of adaptive techniques, which could substantially improve the coding performance.

It seems natural to adapt the quantization procedure from one block to another depending on the details of the image in that block. For example areas with high detail must be represented by more bits than areas with low detail where fewer binary digits can be used or, for areas with high detail, the number of coefficients to be transmitted must be higher than for areas with low detail. The adaptive procedure will therefore, depend upon the activity within the block chosen.

Charles and Wintz [63] suggested an adaptive KL transform coding. They defined eight categories according to the number of coefficient required to be transmitted (Table 1.2.8.1).

They have used a block of 6 x 6 data and by sequentially scanning each line of the block, they have converted it into a one dimentional vector $\underline{X}$ of m = 36 samples with zero mean, which makes the calculation simpler. Each vector $\underline{X}$ is transformed to a vector $\underline{Y}$ and the total enegy is calculated $E = \sum_{i=1}^{m} Y_i^2$.

Then find N such that $\sum_{i=1}^{N} Y_i^2 \geq (1-A(\underline{X}))E$, where $A(\underline{X})$ is a quality criterion for $\underline{X}$. Choose L from Table 1.2.8.1 such that $K_{L-1} < N \leq K_L$.

Table 1.2.8.1 Category Assignment for Constant A

Category (L)	(K_L) Number of Coefficients Transmitted for each Category
1	6
2	9
3	12
4	15
5	18
6	21
7	26
8	32

For example if N = 16, then L = 5 and the number of the coefficient transmitted is 18.

The overhead information is 3 bits per block for specifing the category. They reported a bit rate of 2.5 to 3 bit per picture element for A ($\underline{X}$) = constant = 0.01. For better subjective quality they used $A(\underline{X}) = \alpha\ (\bar{X}/Q)^n$ instead of a constant A and different category assignment table, where $\bar{X}$ is the average of a source vector $\underline{X}$, Q is the maximum grey level in the original picture and α and n are parameters. $\bar{X}/Q$ represents the normalized (average) brightness of a block. This system will allow more error in brighter areas than in darker areas. They

have claimed that this system produced good quality at 1.4 bit / pel for = 0.5 and n = 2.

Tasto and Wintz [11] proposed a different adaptive KL-Transform coding. Each block of 6x6 is converted to one dimention block of 36 samples as above and is classified into one of three categories. Category I: blocks containing a lot of detail, Category II: blocks containing little detail and darker than average and Category III: blocks containing little detail and lighter than average. For each class a covariance matrix and the corresponding set of eigen vectors are used to transform that particular class. Each class has its own quantization procedure for block quantization of transformed data. The overhead information required to be transmitted for specifying the class, to which the given block belongs, is 2 bits per block. A bandwidth reduction of 30 to 50 per cent have been reported over the non-adaptive KL transform method. In both schemes the same uniform quantizer, suggested by Hayes and Bobilin [64] was used for all the coeffiecients.

Since the adaptive KL transform is so complex, other deterministic transform like Fourier, Hadamard, discrete cosine transform (DTC) etc., have been tried along with some adaptation in either quantization or sample selection.

The simplest technique possibly, is to transform the blocks of data and then select only those coefficients which are larger than a certain threshold. The tra

nsform coefficients below the threshold are set equal to zero at the receiver.

The number and the location of the coefficients, above a fixed threshold, change from one block to another and, therefore, the system is adaptive. Dillard [65] reported a system using this adaptive threshold quantization technique with 4x4 Hadamard transform. The dc-term and the largest coefficients along with their adresses were transmitted. He reported a bit rate of 1.625 bit/pel without significant degradation by using the dc-term and the two largest coefficients.

Two dimentional Fourier transform on a block of 16x16 picture elements followed by an adaptive quantization has been reported by Anderson and Huang [6]. The standard deviation of the elements in each block was first measured. The number (L) of the coefficients to be coded from each block and the number of bits used for quantizing the magnitude and phase of these coefficients were made proportional to the standard diviation of the picture elements in that block. Then amplitude, phase and the position of the L transformed samples with the largest amplitude were transmitted. Good results were reported at 1.25 bits/pel.

Different adaptive transform coding methods use different measures for sample selection and quantization in the transform domain. Besides using the standard diviation of the block of picture elements, the most frequently used measures are the variances of the trans-

form coefficients, the sum of absolute values of the a-c. coefficients and the a-c. energy of the transformed coefficients. In a sense, all of them give a measure of the image activity and in fact the variances and the a-c. energy of transformed samples are a measure of the randomness of the image.

The sum of the absolute values of the a-c. coefficients in transform domain, refered to as the activity index [66], can be used to classify each block to one of the M possible classes. Each class would used a different sample selection and quantization procedure.

Claire [66] and Gimlett [67] have recommended the use of the activity index with four possible classes. They use a combination of zonal and threshold sampling for each class.

Chen and Smith [68] used fast computational discrete cosine algorithms to calculate block of 16x16 picture elements, followed by adaptive coding technique based on the image activity level. The transform blocks are sorted into four classes according to the level of image activity which is measured by the total a-c. energy in each block. Within each activity class, coding bits are allotted to individual transform elements according to the variance matrix of the transformed data. More bits are assigned to busy areas and higher level of activity are preferred over lower levels. They reported good picture quality at 1 bit/pel and satisfactory quality at 0.5 bit/pel for still pictures.

Different authors [69,70] have used the Cosine transform with adaptive quantization and good results have been reported compared with results obtained by using uniform or Max- quantizers [70].

Instead of using the sum of the magnitudes of the a-c. transform coefficients as a measure of image activity, the variances of the coefficients may be used to define spatial activity, which is then used for adaptive sampling and quantization of the coefficients.

Tescher et al. [71,72] have used the variances of the transformed coefficients as an activity index for adaptive bit assignment. They have taken a 256 x 256 picture and used a two-dimentional Fourier transform to obtain the complex Fourier coefficients. The complex coefficients are represented in term of their amplitude and phase and the two are treated separately. The variance of each amplitude components is estimated and bit assignment is made in proportion to the logarithm of its estimated variance.

In the same way, the phase components are processed but it has been found in practice that the phase components must be quantized more finely as the picture is more sensitive to phase degradation than amplitude degradation.

The phase component is quantized using one more bit than has been used for the corresponding amplitude component. The variance of the amplitude of individual coefficients is estimated using a predictor that combine

the variances of a number of adjacent quantized samples to predict the variance of a given coefficient.

This system requires prior knowledge of the estimated value of the first variance to start the process. An improvement of about 50% in reduction of the bit rate over the non-adaptive systems was reported. They used Fourier and Hadamard transformations with similar results.

A slightly different approach has been suggested by Tescher et. al.[73-75] to the problem of sample selection. They divided the image into blocks of 16x16 picture elements and transformed it using DCT or Slant transform. But instead of using usual scanning of line by line to convert the transform coefficients to one- dimensional set, they have used the scanning pattern shown in fig. 1.2.8.1

They argue that such scanning pattern gives a smoother decay in the size of the variance of the transformed coefficients and that strong correlation exists among adjacent coefficient variances, even though the transformed coefficients are uncorrelated.

Next the variances of the one-dimensional data sequence is estimated and bit assignment is made proportional to the logarithm of the estimated variances. When the variance of a coefficient is so small that the number of bits assigned to it falls below one bit, the processor stops and all remaining coefficients in that block are set to zero at the receiver.

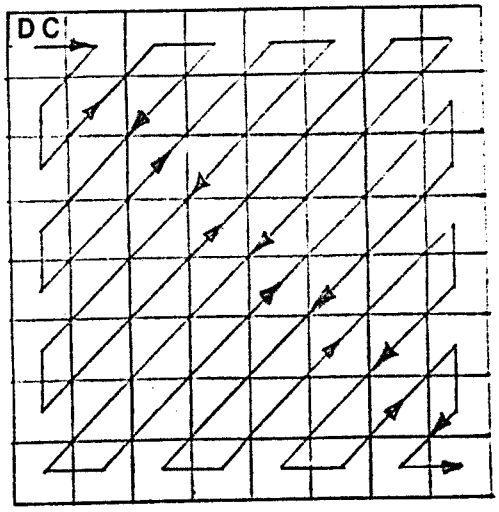


Fig. 1.2.8.1. Ordering of the samples in frequency domain of 8x8 block.

The estimate of the variance for the nth. transform coefficient σ_n^2 is

$$\sigma_n^2 = Al\sigma_{n-1}^2 + (1-Al)\hat{X}_{n-1}^2$$

where $\hat{X}_{n-1}$ is the quantized form of the (n-1)th. transformed sample and Al is a weighting factor which has been taken as Ø.75.

One procedure for estimating the first variance is [75] $\sigma_1^2 = (X_1^2 + X_2^2 + X_3^2 + X_4^2)/4$. where the X's are the transform coefficients.

In addition to these techniques, a number of adaptive methods have been devised that use a fixed number of coefficients and a fixed number of quantizers in each block. However a different normalizing constant is used to normalize the coefficients in that block prior to their quantization [76,77]. The normalizing constant

must also be transmitted for each block.

A comparative study between different adaptive schemes has been done by Nagan [78] using Cosine and Hadamard transforms.

The transform techniques mentioned above are not only used on still pictures, but also can be used on moving pictures, where not only the redundancy in the same frame (intraframe) is exploited but even so the redundancy in successive frames. This is called interframe transform coding. A study made by Roese [79] has shown that this approach yields transform coders whose performance greatly exceeds that of conventional intraframe coders. However, the main disadvantage of such coders is the requirement of excessive storage when the blocks are 16x16x16 (horizontal x vertical x temporal) or more. Therefore interframe transform coding uses smaller blocks. Mounts et al. [62] used blocks of 2x2x2 with Hadamard transform coder. Natarajan and Ahmend [80] applied Walsh-Hadamard transform (WHT) and the discrete Cosine transform (DCT) on 4x4x4 blocks achieving one bit/pel without motion degradation or other distortions.

1.3 Predictive Coding Techniques.

The next class of efficient coding techniques which makes use of the correlation between adjacent signal values is the predictive coding.

In basic predictive coding systems the sample to be coded is predicted (estimated) from previously coded information that has been transmitted. The difference signal (error) between the actual value and its estimate is then computed and quantized into a set of discrete amplitude levels. These levels are then represented as binary words of fixed or variable wordlength and transmitted. At the receiver, the code words are decoded and added to the receiver prediction to reproduce the reconstructed signal. Thus the predictive coder has three basic components: 1) Predictor, 2) Quantizer, 3) Code assigner. The most commonly used predictive coders used in speech and image compression are the differential pulse code modulation (DPCM) and delta modulation (DM).

1.3.1 The DPCM Coders.

DPCM system was first introduced by Cutler [81]. His invention is based on transmitting the quantized difference between successive sample values rather than the sample values themselves. Fig 1.3.1.1 shows the block diagram of a DPCM system.

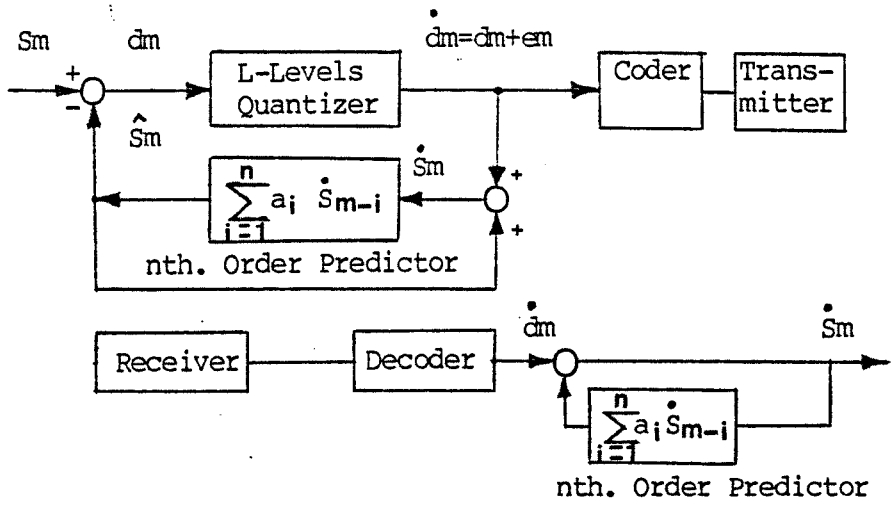


Fig 1.3.1.1 Block diagram Components of DPCM Compression.

Oliver [82] and Harrison [83] realized the importance of linear prediction in feedback communications systems and proposed that it be used to reduce the redundancy, and therefore, lower the required power in highly periodic signals such as television. Oliver [82] explained how linear prediction could be used to reduce the bandwidth required to transmit redundant signals. Harrison [83] actually built a signal processing system and applied it on image data and illustrated how redundancy could be removed from those signals using linear prediction. He extended the basics of DPCM system by forming the prediction from a linear combination of previous pixels along the same line and previous lines (two dimensional predictor). Later Elias [84] developed the theory of prediction coding which explained the use of linear prediction in PCM systems. Graham [85]

recognized that the theory of prediction could be incorporated into the system described by Cutler. He demonstrated by computer simulation the feasibility of using 3-bit DPCM for the television transmission of still black and white pictures.

O'Neal [86] has analysed DPCM for the transmission of video signals. He concluded that previous-sample feedback DPCM transmission system can provide a signal to-quantizing noise ratio approximately 15 db. higher than standard PCM, a signal encoded into DPCM is more vulnerable to noise in the transmission channel (bit error) than one encoded by PCM. He also demonstrated that, if the horizontal correlation is equal to the vertical correlation, then the improvement in signal to quantizing noise ratio, when vertical pixel is used in addition to horizontal pixel, is small and it is about 1.9 db.

However subjective evaluation indicates that the reproducing of vertical edges is significantly improved due to two-dimensional prediction [87].

The efficiency of DPCM image data compression depends on the order of the predictor n (number of previous pixels used by the predictor), the values of the predictor coefficients (ai), the quantization threshold and the number of quantization levels. The order of the predictor is determined by the data. In general, if a data sequence is modeled as an nth. order Markoff process, then an optimally designed nth. order predictor

will cause the resulting prediction error sequence to be uncorrelated [88]. Images are obviously not nth. order Markoff processes, but experience with image data has shown that it is possible to model the overall covariance statistics of images by third-order Markoff processes which leads to a third-order predictor [89]. Habibi [89] computed the prediction mean square error (MSE) using different numbers of previous picture elements. His results show that if the predictor coefficients are matched to the statistics of a picture, then for that picture, the MSE decreases significantly by using up to three picture elements, and further decreases are rather small by using more than three pixels.

If the prediction is well chosen, then the difference signal (dm) will be small in most part of the picture, thus the first order entropy H(dm) of the difference is in general substantially smaller than the entropy of the original input signal (Sm). Therefore a code with variable word length (for instance a Huffman code [90]) for coding the difference will reduce the bit rate compared to PCM.

The best DPCM-system design is that system which minimizes a measure of the overall error between the input and the output of the system. However this design procedure is prevented by the non-linear characteristic of the quantizer. Therefore the optimization problem is solved by designing the linear predictor ignoring the presence of the quantizer and then the quantizer is

designed to match the amplitude distribution of the difference signal. Bodycomb and Hadad [91] showed that if the quantizer is replaced outside the feedback loop, then the mean-square error (MSE) performance of the system for a Gaussian-Markoff process is not better than the performance of PCM-system. Geddes [92] shows that in such system, if the difference signal (prediction error) is quantized and transmitted directly, then the quantization error will accumulate in the integrating filter at the receiver producing gross streaking in the received picture. Therefore the quantizer has to be included in the prediction loop to make the transmitter and receiver identical, which means that the transmitter and receiver predictors will be operating from the same quantized predicted values, thus minimizing the error between them in the reconstruction process.

On the other hand replacing the quantizer in the loop will change the distribution of the prediction error (quantizer input) and makes the difference signal correlated and the system is no longer optimum. But if the number of quantizing levels is large, then including the quantizer has very little effect on the amplitude distribution of the difference signal. However, the system is optimum only for that input signal which the system is designed for.

The coefficients for optimum predictor are found by minimizing the mean-square error of the input image. This leads to a set of linear equations which can be

solved from knowledge of image autocorrelations. This method for designing a two-dimensional predictor could result in an unstable recursive filter [93]. This means, while the prediction error is minimized (ignoring the quantization effect), the reconstruction filter could be unstable causing any transmission error to be amplified greatly at the receiver. Pirsch [94] investigated the stability conditions for DPCM coder of multidimensional DPCM-systems and sufficient stability conditions have been derived both in the signal domain and transform domain.

The main impairments in DPCM-systems are the granular noise, slope overload and edge busyness [95], which are intrduced by the quantizer. The noise structure in slope overload has been investigated by Protonotarias [96]. Goldstein and Liu [97] who analysed the three types of error and approximated equations have been obtained for each of them for Gaussian input.

Almost commercial quality pictures have been obtained using 4 bit/pel DPCM coder [98,99], while acceptable quality can be produced using 3 bit/pel [95].

1.3.1.1 Adaptive DPCM Systems.

DPCM systems using a fixed optimized predictor generate a well behaved stationary differential signal if the original data is stationary. The stationary differential signal can be encoded optimally using a nonlinear quantizer matched to its statistics. However, when the input signal is non-stationary and the predictor parameters are fixed, a non-stationary difference signal is the result. Optimal encoding of the non-stationary difference signal then requires a variable quantizer which would change to accommodate the variation in the difference signal.

In designing an adaptive DPCM system one must either use a predictor with variable parameters such that the parameters would change with the variation of the input signal (adaptive prediction), thus generating a stationary difference signal, or one can use a fixed predictor with a variable quantizer (adaptive quantzation) to accommodate the resultant non-stationary difference signal, or using a combination of both.

1.3.1.1.1 DPCM Systems with Adaptive Quantizers

Let us assume a predictor which uses weighting of adjacent samples either in the same line or a combination of samples in the same line and the line just above it.

Since the input signal is non-stationary, the difference signal will also be non-stationary and therefore the quantizer must be nonlinear and adaptive to match the statistics of the difference signal. Ready and Spencer [100] have suggested an adaptive signal encoder called block-adaptive DPCM in which a block of M samples is stored and each sample is predicted using two dimensional fixed predictors. The errors are encoded by N possible quantizers. For each quantizer the mean-square error for the block is calculated at the transmitter. The quantizer which gives the smallest distortion is used and the quantized errors are transmitted. The system requires $(\log_2 N)/M$ binary digits per sample overhead information for receiver synchronisation.

In fact, they used fixed quantizer and N different pre-specified scaling factors to scale the difference signal before the quantization. The quantizer output must be rescaled by inverse factor prior to the predictor. They used a block of 16 samples with four possible scaling constants. They reported an improvement of 36% reduction in bit rate over a similar nonadaptive system. Brown [101] described DPCM system with quantizer whose quantizing range expands whenever the amplitude of the input signal exceeds a predetermined threshold. In this system the information that the quantizer has switched to the expanded range is communicated to the decoder by the transmission of an additional code word. This sliding -scale direct-feedback PCM coder makes it possible to

reduce the quantizing effects, such as the overload and edge busyness. The drawback of this system is that the uniform output bit rate of DPCM coding is interrupted by the insertion of the switching code words, thereby making buffering necessary. Musmann [102,103] and Lueder [104] proposed a DPCM coder in which the quantizer be switched as a function of previously reconstructed picture elements. This solution does not require switching information to be communicated to the decoder and the uniform output bit rate is maintained.

Cohen [105] and Kummerow [106,107] determined the improvement in the signal-to-quantizing noise ratio of video signals that can be achieved with a switched quantizer. They reported an improvement by 3-4 db. over the standard DPCM.

For better picture quality, quantizers should be designed on the basis of psychovisual criteria [108-113]. Adaptive systems using these quantizers have been also described [114-116].

Prasada et al. [114] have reported a procedure of reassigning the input levels of the quantizer to different representative output levels in such a way as to reduce the entropy of the quantized output. The visibility of the resultant quantization noise is kept below a certain specified threshold. They make use of the spatial masking effect defined as a reduction in the ability of a person to visually discriminate amplitude errors which occur at or in the neighbourhood of

significant spatial changes in the luminance. Another adaptive quantization process has been described by Prasada et al. [115] where the prediction error is multiplied by a constant which depends upon the local properties of the picture element surrounding the picture element being coded. The constant is chosen to make the visibility of the quantization noise approximately uniform throughout the picture. With a two dimensional predictor and this process, a good quality picture can be achieved by using 8 or 10 quantizer levels.

1.3.1.1.2 DPCM Systems with Adaptive Predictor.

Adaptive prediction systems for images were proposed first by Graham [85] and later by Connor et al. [87]. Both of these systems used intraframe switched predictors which based their prediction on previous pixels in the neighborhood of the pixel to be coded. Zschunke [117] has suggested an adaptive contour prediction technique. Here the information on contour directions derived from neighbouring picture elements is used to select a suitable prediction value for the actual sample. Zschunke used estimated of the contour direction from the previous picture elements for the contour adaptive DPCM system. The location of contour points and the contour gradients or area brightness levels are required to be transmitted in Zschunke system. He also used switched quantizer along with the adaptive prediction. He reported that a bit rate of 3 bits/pel gave acceptable results.

Another adaptive system was described by Dinstein and Garlow [118]. In this system each line is partitioned in M segments. The range of difference signal divided into three regions, fine, medium and coarse and 3, 6, and 10 bit quatizers respectively were used. Both horizontal and vertical predictions are applied to each segment. For each segment the number of bits required to code the quantizer outputs was calculated. The prediction that yields the smallest bit rate for the segment under

consideration is then used for transmission. One extra bit is transmitted with each segment to notify the receiver of the prediction which has been used. The reconstructed images at 3 bit/pel show no perceptible degradation.

1.3.2 Delta Modulation Systems.

Delta modulation is based primarly on the invention by De Jager [119,120]. It is a simple type of prediction quantization systems and is essentially a one-digit differential pulse code modulation system. Fig.1.3.2.1 is the block diagram of a delta modulator.

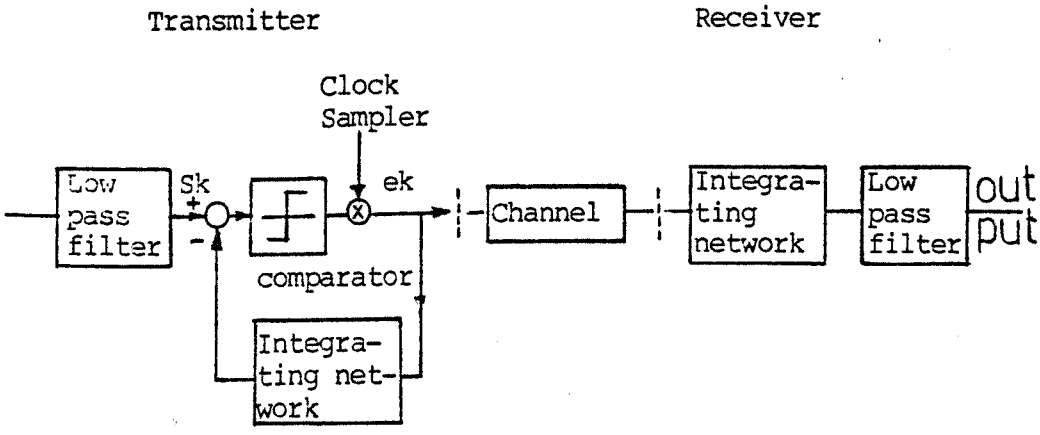


Fig. 1.3.2.1 Block Diagram of a Delta Modulator.

In this class of systems, the sampling rate is chosen to be many times the Nyquist rate for the input signal. As a result, adjacent samples become highly correlated.

The primary limitations of DM are slope overload, granularity noise and instability to channel errors. Slope overload occurs whenever there is a large jump or discontinuity in the signal to which the quantizer can respond in several delta steps. Granularity noise is the steplike nature of the output when the input signal is

almost constant.

The performance of the delta modulator can be considerably improved by making it adaptive. The adaptive Delta modulator (ADM) still remains a simple system even with the additional complexity of adaptation to the signal statistics. The adaptive strategy can be varied or controlled to suit the signal statistics and therefore substantial improvements are possible. In the ADM system shown in fig. 1.3.2.2 each sample is compared with its estimated value and the difference signal is either positive or negative and sampled at the clock rate to give ± 1. The output of the comparator is multiplied by a constant in the feedback loop and is used as an input to an integrator. The output of the integrator is an estimate of the next incoming signal value.

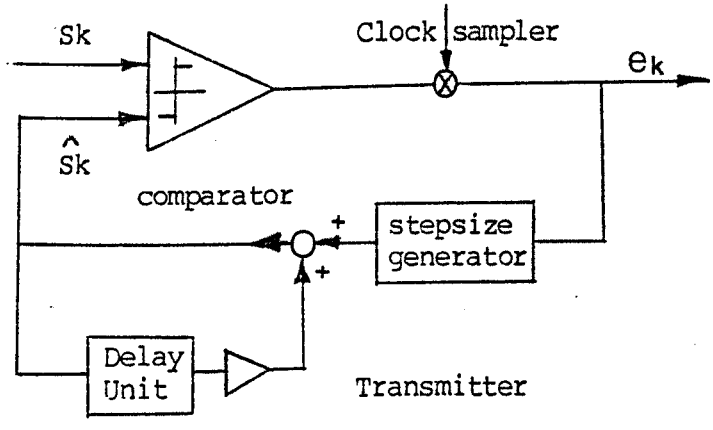


Fig. 1.3.2.2 Block diagram of an ADM Coder.

ADM systems have been designed both for speech [121] and video signals [122].

The most widely used approach is to change the step size of the system according to signal variation. Step size is increased if the polarity of the comparator output remains the same for many clock pulses indicating a high activity area. On the other hand, if the polarity alternates indicating low activity area, then the step size is kept the minimum.

1.3.2.1 Adaptive DM.

It was pointed out by Winkler [123,124] that the bit pattern of the output of coder can be used for detecting the presence of an edge. He made the step size adaptive depending upon the number of "ones" and "zeroes". Thus if the three pulses at the output are successively of the same polarity, then the step size with the third pulses is double that of the second pulse. If two successive pulses differ in polarity, the step size for the second pulse is half that of the first. The step size sequence is 1,1,2,4,8,....,etc, for continuous strings of Ø or 1, or half of the previous step size if there is a change in polarity.

Bosworth and Candy [125] carried out an extensive subjective testing programme in order to obtain a weighting sequence which is acceptable to the observer. They found that the sequence of 1,1,2,3,5,5...5, with sequence restarting at every change of code polarity was

the best. Since the first two steps are always 1,1 at the beginning, all the flat areas with non-zero slopes will be encoded using these step sizes. On an average, therefore, 90% of a picture will be coded with the smallest step size. An immediate return of the sequence to the start whenever a change of polarity occurs, avoids the use of large step size in flat areas following a step edge.

Jayant [126] made a modification of the Winkler weighting sequence by selecting multipliers P and Q for increasing and decreasing the step size respectively. He optimized the multipliers from the view point of the granular noise, the slope overload and the problem of stability. He concluded that P.Q=1 for reasons of stability and P=1.5 for optimum performance. Thus if the step size for the Kth. sample is $\triangle_K$ then

$$\triangle_K = \begin{cases} P\ \triangle_{K-1} & \text{if} \quad e_K = e_{K-1} \\ 1/P\ \triangle_{K-1} & \text{if} \quad e_K \neq e_{K-1} \end{cases}$$

Where e_K is the output of the comparator. Jayant [126] has claimed an improvement of 10 db. for his system over the nonadaptive DM for coding video signals.

A second approach, known as the Song delta modulator [127] uses the step size of the past sample to form a step size for the present sample. In this method the current step size is generated as

$$\Delta_K = \begin{cases} 2.D.e_{K-1} & \text{for} \quad |\Delta_{K-1}| < 2D \\ |\Delta_{K-1}|.(e_{K-1} + e_{K-2/2}) & \text{for} \quad |\Delta_{K-1}| \geq 2D \end{cases}$$

Where D is a constant which is the minimum step size of the system. In other words, if the previous step size is smaller than twice the minimum step size, then the current step size is 2D. On the other hand if the past step size is greater than 2D, then the new step size is 1.5 times or Ø.5 times Δ_{K-1} depending upon the history of the comparator output.

This weighting sequence was used for picture coding by Scheinberg and Schilling [128] and they have reported satisfactory results at 2 bit/pel. Schilling et al. [129] have modified the Song algorithm to improve the video coding at low bit rates. The algorithm employed is:

$$\Delta_{K+1} = \begin{cases} |\Delta_K|.(e_K + e_{K-1}/2) & \text{for } D \leq |\Delta_{K+1}| \leq 15D \\ 2D.e_K & \text{for } |\Delta_K| = D \\ 15D.e_K & \text{for } |\Delta_K| = 15D \end{cases}$$

Instead of a generalized system with P and Q multiplier, Habibi [13Ø] suggested a simple step size assignment technique based on the past three outputs of the comparator. For the eight possible combinations of the three output bits, he has assigned six step sizes of ±1, ±2 and ±4. Combinations of 11Ø and 1Ø1 are assigned

the same step size of +1, and similarly 010 and 001 are assigned -1. Performance of this scheme for coding of monochromatic signals is satisfactory at rates of 2-3 bits/pel.

1.3.2.2 Two-Dimensional ADM.

Picture encoding using ADM can be improved considerably as in the DCPM system, by performing an estimate of the current sample based on the adjacent samples in the same line and the line just above it. Lei et al. [122] have described a two-dimensional ADM and have concluded that estimates based on the previous horizontal and vertical elements, give the best results. They also proposed a system with look ahead facilities like the one suggested by Cutler [131] where they take the effect of the picture element right to the vertical element into account while estimating for the current sample.

1.4 Hybrid Coding

Each one of the two coding methods discussed previously (Transform and Predictive Coding) has advantages and disadvantages. The transform coding systems achieve superior coding performance at lower bit rates; they distribute the coding degradation in a manner less objectionable to a human viewer, and are less vulnerable to channel noise. The demerits of them are their complexity in term of both the storage of data and number of operations required. Although the use of large block sizes removes statistical redundancy quite effectively, it has two distinct disadvantages: 1) it requires storage of large amounts of data both at the transmitter and at the receiver, and consequently produces a delay in transmission, and 2) the accuracy with which different regions of the image need to be coded may vary widely within the block, and this makes adaptive coding (e.g. quantization) more difficult to accomplish.

On the other hand, DCPM systems, when designed to take advantage of spatial correlation of the data, achieve a better coding performance at higher bit rates, the equipment complexity and the delay to the coding operation is minimal, and the system does not require the large memory needed in the transform coding systems. The limitation of these systems are the sensitivity to picture statistics and the propagation of the channel

error on the transmitted picture. Thus the use of combinations of the two coding methods implies the advantages of both. This technique is called "Hybrid Coding".

The hybrid system as developed by Habibi [132] exploits the correlation of the data in the horizontal direction by taking a one-dimensional transform of each line of the picture, then a bank of DPCM coders are applied to each column of the transformed data. The DPCM coders quantize the signal in the transform domain, where they take advantage of the vertical correlation of the transformed data to reduce the coding error.

Another version of the hybrid processing is the application of small two-dimensional block or subimages of size NxN and the DPCM is used on coefficients of horizontally previous block.

The two-dimension transformation tends to decorrelate the samples in each block, whereas the DPCM exploits the interblock correlation. The transform coding may be any of the KL, Hadamard, Slant, Fourier or Cosine transforms.

Habibi [132] evaluated both theoretical and experimantal results, which indicate that a hybrid system employing a KL transformation produces a better result than the hybrid system using any other transformation. He also demonstrated the effect of channel error on the performance of the hybrid systems and found that the inherent propagation error of the DPCM systems can be

reduced considerably by using a value of Ø.9 for all prediction coefficients of the DPCM system. This is because Ø.9 is smaller than the optimum values of most of the prediction coefficients and this causes a shorter propagation of the channel error in the encoder. He achieved good picture quality at 2 bit/pel and acceptable quality at 1 bit/pel.

Ishii [133] showed that the bit rate performance of hybrid systems using one transform element for prediction, does not depend on the block size. He has obtained an acceptable picture quality at Ø.5 bit/pel. He also examined the effect of the channel error and found that the reconstructed image quality scarcely degrades even at noisy channel of $1Ø^{-4}$ error rate.

Rao et al. [134] applied hybrid techniques, where sample selection and variable bit allocation are adapted. They compared the hybrid process with the two-dimensional transform method in terms of bit rate, mean-square error, and computational complexity using block of 16x16.

Netravali et al. [135] used a small two-dimensional block and showed that, if transform coding other than the optimum (i.e. KL) is used, correlation is still present between coefficients of the same block, and therefore, a better predictor can be designed by using not only the corresponding transform coefficients of the previous block but, as well as coefficients of the present block which are available to the receiver. They

showed that such a system was 25% more efficient in terms of bits/pel for the same picture quality than predictor which used the corresponding coefficient of the previous block. Experimental and theoretical (Markoff process) performances have been investigated by Roese et al. [136] for transform coding using three-dimensional (interframe) blocks, and hybrid coding employing two-dimensional (intraframe) blocks followed by DPCM in temporal direction. They showed that the hybrid coder is quite efficient and does as well as a three-dimensional transform coder which uses four frames of storage. Adaptive interframe hybrid coding has also been implemented [137,138].

1.5 Coding of Non-stationary Image Signals

Most techniques of image bandwidth reduction incorporate a single source model for overall image signal behaviour. Wide-sense stationarity form the basic assumption of many transform or predictive coding techniques, where the wide-sense stationary parameters are the mean and autocorrelation function [139].

In general, picture signal is highly non-stationary and the local statistics vary considerably from region to region. Breaking the image up into blocks of size nxn and calculating the mean and autocorrelation function within each block, yields means and auto-

correlation functions which change from block to block. This fact reduces the efficiency of nonadaptive coding techniques significantly. Hunt [140] proposed the use of non-stationary statistical image model and discussed prospects of transforming a non-stationary image model into a stationary one. Stricklard [141] developed a transform for producing images with wide-sense stationarity.

Another approach for treating non-stationary image signal, is to consider the image signal as the output of many sources each tuned at certain type of statistics [142-144] and the sources are coded separately. Yan and Sakrison [142] considered a two components model in which the vertical edges (or the high frecuency components) are treated as one component and the rest (texture details) are treated as the other component. They argue that if the edge information is substracted from the picture signal, the rest of the signal appears to be close to a Gaussian process and, therefore can be efficiently coded by using one of the earlier mentioned coding techniques.

Mitrakos and Constantinides [144] presented a coding technique which provides a full control on the distortion in the reconstructed picture and on the transmission rate required. They partitioned the picture in two components. One represents the variation in the local background (C-component) and is entropy coded. The other which represents random variation in the texture (e

-component) has approximately a Gaussian distribution and is transform coded.

Compared to ordinary transform coding, for the same mean-square error this method offers several choices in terms of the subjective quality of the receiver image. However the complexity of the algorithm is increased substantially compared to other adaptive transform coding methods.

1.6 Data compression: The way forward.

Transform coding and predictive coding have been extensively researched by many authors, who have found that the performance can be improved by different adaptive algorithms. Although transform coding is considered to achieve the best compression and the distortion that arises from this technique is found to be visually less objectionable than that for predictive coding, the complexity of the computation makes it unattractive for the majority of applications. Predictive coding, though, gives high distortion and low immunity to channel noise, however, its simplicity and relative ease of implementation, make it the most popular coding technique. Both coding schemes, due to their inherent properties, are mainly used in picture and speech coding coding and are not universal algorithms in the sense, that they are not applicable to all kinds of data.

Moreover, they are not noiseless algorithms, which means, that the output data is always subject to error. Thus, there is a need to unify the approach to the compression of images, speech and other similar data. Such a scheme is desirable, in order to meet the requirement of modern data communication links. The objective of this work is to introduce a coding system, which satisfies the above requirements. The Thesis is divided into six chapters. After a brief review of image data compression in chapter one, a detailed study of DPCM and its performance from different aspects is discussed in chapter two. New adaptive approaches for improving the performance of DPCM systems are proposed and experimental results given in chapters three and four. A novel procedure using a hierarchic tree structured Hadamard transform is proposed in chapter five. This coding algorithm is flexible in the sense that it can be used for error-free coding if desired or for high compression whenever some noise is acceptable. An unified data compression strategy is introduced in chapter six, based on the learning automata and Huffman coding. This algorithm is shown to be universal and to satisfy the requirements of data links using visual, audio, alphabetic and numeric data.

CHAPTER TWO

2. Linear Predictive Quantizing System

2.1 Introduction

This chapter presents an analysis of differential pulse code modulation (DPCM). An optimum system for three pictures with different statistics was designed, and simulated on a computer; its performance was investigated for different types of predictor. The performance measurement is based on the mean square error, the density function and the spatial distribution of the differential signal. The design procedure is given in Appendix I.

The quantizer characteristics are determined to give, subjectively good picture quality and its mean square error performance is compared with that of a Max quantizer matched to the statistics of the differential signal.

Of practical value is the non-optimum (unmatched) systems. This system is also simulated and its performance subjectively and objectively is evaluated.

2.2 The Predictor

In DPCM image coding systems Fig.2.2.1, the predictor uses the statistical predictability between pixels to form an estimate of each pixel as a linear combination of previous pixels; where "previous pixels" is a term that has direct meaning in the context of top-to-bottom, left-to-right scanning, which imposes a specific sequence on pixel occurence. Fig.2.2.2 shows the elements location for interlaced case (T.V. signals) and non-interlaced case (facsimile signals).

The samples S1, S2, - - -,Sn need not be the most recently transmitted ones and they need not be in any particular order. They are simply n samples values which have been transmitted in the past. Fig.2.2.2 shows some sample values which can be used to form a reasonably good estimate of (So). Such an estimate would be

$$\hat{S}o = ao+a_1S1+a_2S2+a_3S3+a_4S4+a_5S5+ \text{---} + a_nSn \quad (2.2.1)$$

for convenience, the signal mean is subtracted from the signal, so the signal we deal with is zero mean. Therefore the d-c term is eq.(2.2.1) will become zero and the estimate is given as

$$\hat{S}o = a_1S1+a_2S2+a_3S3+a_4S4+a_5S5+ \text{---} + a_nSn \quad (2.2.2)$$

where a's are chosen to satisfy eq. (I-21b Appendix).

The number of previous pixels employed in the estimate operation, forms the predictor order. Predictor using one pixel is called "first order predictor". A "second order predictor" utilizes two pixels and an "nth. order predictor" would employ n previous pixels.

The location of the previous pixels used by the predictor, determines the predictor dimensionality. Prediction using only pixels along the same line as the pixel to be estimated is assigned as "one-dimentional prediction" whereas a "two-dimentional predictor" uses pixels from the current line as well as from the previous lines. Predictive coding using pixels from the current frame is called "intra-frame predictive coding".

It is possible to extend the technique of predictive coding to exploit also the correlation existing between pixels of succesive frames as in T.V. signals. This technique is called "Inter-frame predictive coding". For scenes with low detail and small motion, interframe prediction appears to be the best and can achive a low bit rate (1 bit/pixel) [145]. The drawback of interframe prediction is the large amount of frame storage required.

A standard DCPM coder, which utilizes the previous scanned pixel (S1) along an image line as the basis of its prediction of So, is often referred to as "previous-sample feedback system", and according to the definitions above, this is a one-dimensional, first order predictor. The estimate is given as

$$\hat{So} = a1S1 \qquad (2.2.3)$$

where a_1, according to eq. (I-21b Appendix) should be $R\emptyset1/R11 = R\emptyset1/\sigma^2$; the correlation between adjacent sample points divided by the mean square value of the input sequence. A second-order predictor would utilize the two previously scanned pixels along a line (S1 and S5) (one-dimensional predictor), or perhaps the previous pixel along the line (S1) and the nearest pixel from the previous line (S2) (two-dimentional predictor). A third-order predictor might employ (S1, S2, S3) (two - dimensional predictor) as the basis for its prediction. In this case

$$\hat{So} = a_1S1+a_2S2+a_3S3 \qquad (2.2.4)$$

The pixel measurements that should be employed for minimum coding error correspond to the pixel neighbours with the highest statistical correlation to the pixel to be estimated. Habibi [90] shows that the coding error reduction diminishes rapidly for more than third-order prediction system.

The difference between the actual value (So) and its estimate ($\hat{So}$) is called the prediction error (do).

$$do = So - \hat{So} \qquad (2.2.5)$$

The prediction error (do) is quantized to ($\dot{do}$)

and the quantized value is then added to the estimate value $\hat{So}$ to produce a quantized version ($\dot{So}$) of the input (So).

$$\dot{So} = \hat{So} + \dot{do} \qquad (2.2.6)$$

This sampled version is fedback to the predictor for estimating the next sample.

The difference between the prediction error and its quantized value is the quantization error (eo), by which the quantized version $\dot{So}$ differs from the input value (So), thus

$$eo = \dot{do} - do \qquad (2.2.7)$$

$$\dot{So} = So + eo \qquad (2.2.8)$$

It must be noted that the quantized prediction error ($\dot{do}$) is the quantity that is coded and transmitted. The DPCM systems achieve compression from the differencing step, since the prediction error will have a much smaller dynamic range than the original signal. The expression $10 \log_{10}\sigma^2/\sigma_d^2$ may be thought as the amount of redundancy removed from the signal, where σ^2 and σ_d^2 are the variances of the input signal (So) and the difference signal (do) respectively. The shape of the amplitude density of the prediction error (quantizer input) is of foremost importance in designing an optimum quantizer. Fig.2.2.3 shows a typical amptitude density function of the prediction error, which is normally approximated by Laplacian distribution.

2.3 Quantization of Prediction Error

Any analogue quantity that is to be transmitted over a digital channel must be represented as an integer number proportional to its amplitude. The conversion process between analogue samples and discrete-valued samples is called quantization which is always combined with the error called quantization error. In the predictive coding systems the analogue quantity, the prediction error, is fed to an analogue-to-digital converter (quantizer) to produce a digital format.

If the input to the quantizer in Fig.2.2.1 is do, then its output is $\dot{do} = do + eo$, where eo is the quantizing noise. Since the receiver forms the decoded output by adding do+eo to the estimate $\hat{So}$, the quantizing noise in the decoded output is also eo. Minimizing the quantization noise in the decoded output, therefore is equivalent to minimizing the RMS (root mean square) value of noise coming out of the quantizer.

In DPCM systems, the subjective effect of a video observer can be taken into account and the spectrum of the noise is shaped accordingly. It is known that the eye is more tolerant of noise located at black-white interfaces than in flat regions, where the perception is high. Interfaces are characterized by large values of difference signal and flat regions by small values. The

quantizer, therefore, must have a fine structure for low level difference signals (flat regions) and a coarse structure for large difference signals (interfaces). This can be accomplished through the use of a nonlinear quantizer, whose characteristics are depicted in Fig.2.3.1.

Different types of degradation can be seen due to improper design of the quantizer of a DPCM coder. These are referred to as granular noise, edge busyness and slope overload as shown in Fig. 2.3.2.

If the inner levels (for small magnitudes of difference signal) of the quantizer are too coarse, then the flat areas are coarsely quantized and have the appearance of random noise added to the picture (granular noise). On the other hand overload noise occurs when the signal to be quantized is outside the range of the quantizer. When the prediction error (d) is near a decision level (di), any fluctation (source noise and granular noise) makes the quantizer output oscillate between ($\dot{d}i$) and $\dot{d}i+1$ and may change from line to line or frame to frame, giving the appearance of a busy edge.

Quantizers can be designed on a statistical basis or by using certain psychovisual measures.

Experimental results indicate that for most typical pictures the probability density function of the difference signal is a two sided exponential [86]. For such a function it is possible to perform non-linear quantization by a companding operation [146], in which

the non-linear probability function p(d) of the error is first nonlinearly transformed to uniform density followed by uniform quantizer, then an inverse non-linear transformation is applied on the quantizer output.

2.4 Fidelity Measures

Techniques commonly employed for image data compression result in some degradation of the reconstructed image. A widely used measure of reconstructed image fidelity for N x N size images, is the average mean-square error defined as

$$e_{ms}^2 = \frac{1}{N \cdot N} \sum_{i=1}^{N} \sum_{j=1}^{N} E(S_{i,j} - \dot{S}_{i,j})^2 \qquad (2.4.1)$$

where $\{S_{i,j}\}$ and $\{\dot{S}_{i,j}\}$ represent the N x N original and reconstructed images, respectively. Experimentally, the average mean-square error is often estimated by the average sample mean-square error in the given image defined by

$$e_{ms}^2 = \frac{1}{N \cdot N} \sum_{i=1}^{N} \sum_{j=1}^{N} (S_{i,j} - \dot{S}_{i,j})^2 \qquad (2.4.2)$$

Therefore two definition of the term "signal to noise ratio" (SNR), that are used to the above error. These are defined as

(1) $SNR = 10 \log_{10}((\text{Peak value of original data})^2 / e_{ms}^2)$ db (2.4.3)

(2) $SNR = 10 \log_{10}(\sigma^2 / e_{ms}^2)$ db (2.4.4)

where σ^2 is the variance of the original data. Although the second definition of SNR is more widely used as a

measure of SNR in signal processing, the first definition is more commonly in image coding field.

Several visual fidelity measures for images, which have to be evaluated visually, have been suggested [147] such as weighted mean square error of contrast.

We will use in our evaluation the definition in eq. (2.4.3) for SNR measurement.

2.5 Experimental Procedure

The image world is classified in three classes, 1) images with large amount of detail; 2) images with moderate detail and 3) images with small detail. Each class is represented by a member, so a small ensemble of three images is formed. The images are scanned to form 256 lines then each line is sampled uniformly to form 256 picture elements. Thus, each image forms an array of 256 x 256 pixels. The intensity of each picture element is digitized to 7 binary digits (128 gray levels) and stored on a magnetic disk. The images are shown in Fig. 2.5.1, where picture A is low detailed, B is moderate detailed and picture C is high detailed.

For the pattern shown in Fig. 2.2.2b the correlations between So and Sj for normalized picture elements are listed in table 2.5.1.

In standard DPCM using the previous horizontal pixel S1, the first column has to be transmitted directly

as the known conditions upon which the predictions are based. For predictors using the previous vertical pixel S2, the first row has to be transmitted and for predictors using S1, S2 and S3, the first column and the first row have to be transmitted as the known condition. In the system to be considered, a column of zeros has been added before the first column for the predictor making use of the previous horizontal pixel S1, a row of zeros ahead of the first row for the predictor using S2 and a row and a column of zeros when using S1, S2 and S3 as the known conditions upon which the predictions are based (Fig. 2.5.2). Since the d.c. term does not contain any information, the mean value is subtracted from each picture element to obtain an array of zero mean. The value of prediction weighting coefficients Ai for optimum prediction are determined by calculating the covariance matrix and solving this matrix for Ai. i, e,

$$\begin{bmatrix} R11 & R12 & R13 & - & - & - & R1n \\ R21 & R22 & R23 & - & - & - & R2n \\ R31 & R32 & R33 & - & - & - & R3n \\ - & & & & & & \\ - & & & & & & \\ - & & & & & & \\ Rn1 & Rn2 & Rn3 & - & - & - & Rnn \end{bmatrix} \times \begin{bmatrix} A1 \\ A2 \\ A3 \\ - \\ - \\ - \\ An \end{bmatrix} = \begin{bmatrix} Ro1 \\ Ro2 \\ Ro3 \\ - \\ - \\ - \\ Ron \end{bmatrix} \qquad (2.5.1)$$

where Rij = E {Si.Sj} is the correlation between the elements Si and Sj with the equality Rij = Rji. The matrix above represents the simultaneous linear equations of eq (I-21b) in matrix form. Table 2.5.2 lists the weighting coefficients for various predictors. All predictors shown

in the table for all three images were simulated on the computer and the theoretical and actual MS values of the prediction error (without the quantizer) are calculated and listed also in the table 2.5.2. For all predictors the theoretical values agree well with the actual values as can be seen from the table.

In order to determine how effective are the predictors in removing the redundancy, the values of $20 \log_{10} \sigma / \sigma_d$ are computed, where σ^2 and σ_d^2 are the variances of the original image and the prediction error respectively. The table shows that the mean square prediction error decreases significantly with increasing order of predictor up to third order (two dimensional) and further decreases using more than three pixels, are rather small. The results agree well with Habibi's conclusion [89]. This because S1, S2 and S3 or S1, S2 and S4 provide almost all the information about So and once these pixels have been used, there is little advantage in using others. Comparing the MS value of predictors 3 and 4 with that of predictor 5, shows that the two-dimensional predictor is superior to predictors using only pixels on the same line or same column as the pixel to be coded. Figs. 2.5.3 and 2.5.4 show the histogram of the prediction errors resulting from first order predictor employing S1 and third order predictor employing S1, S2 and S3 respectively. In both cases, the density functions can be approximated reasonably well by a Laplacian function. The curves were found by rounding

the error samples to the nearest integer values in the range -127 to 127 and finding the number of samples for each value. Also using predictors higher than third order does not change significantly the shape of the amplitude distribution of the prediction error. The error histogram obtained using predictor 6 (third order) and 10 (seventh order) for scene A is shown in Fig. 2.5.5. The equivalent histogram for scene C is shown in Fig. 2.5.6.

To evalulate the effectiveness of predictors, it is important to base the evaluation not only on the prediction mean square error and the density funtion but also on the spatial distribution of the error. Fig. 2.5.7 shows the prediction error of scene A for a first order predictor using the horizontal pixel S1. This predictor estimates the horizontal edges very well, as can be seen at the top of the head and the face, but can not predict the vertical and inclined edges (see the left and the right side of the head and the shoulders). Here the black shape is a large negative error which occurs at white to black transitions and the white shape is a large positive error which occures at black to white transitions.

On the other hand, the first order predictor using the vertical pixel S2 predicts the vertical edges reasonably well but it produces a large error at horizontal and inclined edges (Fig. 2.5.8). Using higher order predictors which employ only pixels along the same line or the same column are not much better than first order predictors above as illustrated in Figs. 2.5.9 and

2.5.10.

The spatial distribution of the prediction error can be improved by using two-dimensional predictors. Fig. 2.5.11 shows the error of scene A for two-dimensional second order predictor using S1 and S2. It is easy to see that this predictor is better than the one - dimensional predictors. Further improvement can be achieved by involving S1, S2 and S3 (third order) in the prediction of SØ. Fig. 2.5.12 demonstrates the error spatial distribution of scene A of the third order predictor.

Here again, as in the MS error, the improvement in the prediction error by using more than a third order predictor is not significant, since S1, S2 and S3 provide all the information about So. Fig. 2.5.13 shows the prediction error of a seventh order predictor (predictor 10 table 2.5.2).

Figs. 2.5.14 and 2.5.15 illustrate the prediction of third and seventh order predictors respectively of scene B, and Figs. 2.5.16 and 2.5.17 of scene C.

The minimum value of prediction error coming out of the subtractor is set to Ø and the maximum value to 127. All values between the minimum and maximum are then linearly scaled and rounded to the nearest integer number and displayed on the monitor.

2.6 Determination of the Quantizer Characteristics

As mentioned earlier, the error in the reconstructed image (assuming an error free chanel) is the quantization error only. Most techniques use the MSE as their criterion and try to minimize it. Unfortunatly the MSE does not match with the visual impact on the human eye. There are many examples which show that a picture with large MSE can be excellent in quality or that a picture with small MSE is objectionable to human eyes. In fact, a picture of good quality and a picture with small MSE are two different things. Thus we will base the quantizer design on the picture quality, where the word "quality" means the subjective quality (i.e., the quality according to subjective judgment made by visual inspection). Nevertheless, smaller MSE is desirable provided that the reconstructed picture is of a good quality. Granular noise and slope overload are the most annoying errors introduced by the quantizer. Granular noise appeares as random noise added to the picture. Loss of resolution is a result of the slope overload. Since most important structures are located in highly varying areas, any loss of resolution would introduce very obvious degradation to the reconstructed picture. However, the noise like errors are much more tolerable. For the smooth areas, loss of resolution is hardly noticeable, but any noise - like errors will be very obvious.

Quantizer characteristics determined by Max procedure for minimum mean square quantizing error produces in general large overload-effects and degrades the edges. For small differences unnecessary close levels are assigned.

A better quantizer is that, whose levels are determined to reduce the edge degradation in highly varying areas and at the same time keeping the noise in the smooth area under the visual level. For a 3 bit, quantizer and third order optimum predictor this design starts by determining the two inner levels, adjusting them untill the noise in the smooth areas is just noticeable. Then the remaining levels are varied to reduce the degradation in highly varying areas. Table 2.6.1 lists the 3 bit quantizer levels along with the Max quantizer matched to the variance of the prediction errors for a Laplacian distribution for all three images of Fig. 2.5.1. The encoded pictures show less degradation than with Max quantizers. Even so the signal peak squared to quantizing noise ratios were better by 1-2 db.

Another way to see the quality of the reproduced images is by displaying the difference between the originals and the reproduced images. The error signal shows that these quantizers have produced less error than Max quantizers. Fig. 2.6.1 shows the original, the encoded image by Max quantizers along with the error signal and the reproduced image by the design quantizer along with its error signal for picture A. Figs. 2.6.2

and 2.6.3 show the same for scene B and C respectively.

Side by side comparision of the reconstructed images with the originals shows that the reconstructed images can be easily substituted for the original. A third order predictor with S1, S2 and S3 was used.

Also the 2 bit quantizers have produced reasonable pictures without recognisable noise except in scene C in which some noise was introduced in highly varying areas but was not noticable except by side to side comparision with the original at short viewing distance. Fig. 2.6.4 shows the reconstructed images with 2 bits quantizers, whose characteristics are listed in table 2.6.2. Here the signal to quantization noise level has been considerably improved compared with the Max quantizer. The improvement was 4.6 db for picture A, 3 db for picture B and 1.9 db for picture C. The quantizer thus designed are optiumum. Table 2.6.3 illustrates the resulting signal peak square to MS quantizing error (see eq. (2.4.3)) for three bits and two bits quantizers.

2.7 Unmatched Systems

The results considered so far are obtained by measuring the correlation among several points in a picture and then using these in designing the DPCM systems. That is, based on the statistics of the data, an optimum predictor and an optimum non-linear quantizer are

designed. Changing the input data in general degrades the performance of the DPCM system since the predictor and the quantizer will no longer be optimum.

Indeed, the performance of the unmatched systems is of practical value since in practice the DPCM system is designed based on some average statistics and will not be optimum for each particular picture.

To evaluate the performance of unmatched systems in encoding a picture, a second encoder must be designed which is optimum for encoding that picture. Then the difference in performance of the optimum and non-optimum will be a measure of the sensitivity of an encoder to unmatched statistics. We choose the system optimized for encoding picture B to encode the pictures A and C. The reason for choosing system B is, because picture B is of moderate detail and there is no high difference between its statistics and the statistics of picture A (low detail) or the statistics of picture C (high detail). The reconstruced images are shown in Fig. 2.7.1, which show no noticable degradation or noise. The signal to quantization noise ratio is degraded by 2.1 db for image A, where no change in the SNR of image C.

When system A is used to encode piture C, some degradation was noticable. This is because the dynamic range of the quantizer of system B is not large enough to accomadate the large difference signal. The signal to noise ratio was degraded by 3 db. On the other hand when system C is used to encode picture A, noise was just

visible in the smooth areas because the quantization levels are too coarse. The reduction in the signal to noise ratio was about 4 db.

2.8 Experimental Results

Three pictures of different stastics were coded using the DPCM system. Various orders of predictor have been designed to match the statistics of the input data. The prediction coefficients for optimum predictors are listed in table 2.5.2 and all the predictors for all the three pictures have been simulated on the computer. The performance of the predictors with respect to the prediction MSE is also tabulated on table 2.5.2. The density functions of the prediction error for first and third order for all pictures are shown in Figs. 2.5.3 and 2.5.4 and a comparison between third order and seventh-order predictors for pictures A and C is shown in Figs. 2.5.5 and 2.5.6. A display of the prediction error is shown in Figs. 2.5.14 - 2.5.17.

Quantizers are primarly designed for subjective quality of the reconstructed images and the quantization mean square error is considered as secondary outcome.

The practical case of the unmatched system is also simulated and the results are shown in Figs. 2.7.1.

2.9 Discussion and Conclusion

The three pictures used in the experiment can be considered as a representative of the image world. The statistics of the real time images will in general not differ substantially from these three pictures. The comparative performance of the differential PCM system for various predictors shows that the prediction error in respect of MS, shape of the density function and the spatial distribution is improved with the increase of the predictor order up to third and further improvement is rather small in using more pixels than three in the prediction. This is because, three pixels like S1, S2 and S3 provide all the information about So.

The values of the MSE listed in table 2.5.2 support this result. Figs. 2.5.3 and 2.5.4 show the shape of the density funtion of first and third order predictors respectively. The functions at zero level are more highly peaked for third order predictors than first order. Figs. 2.5.5 and 2.5.6 shows a comparison between the density functions of third and seventh order predictors, which shows that there is no gain in using a higher order than third. Figs. 2.5.7-2.5.17 display the prediction error for various predictors. It is easy to see the improvement in the appearence of the prediction error from first to second and third order predictor. Higher order predictors using pixels along the same line or on the same column are not much better than the first

order predictor, since using horizontal pixels only does not provide information about vertical edges and using vertical pixels does not provide the information about the horizontal edges. Two-dimensional predictors are preferable to one-dimensional predictors.

Even though the quantizers were designed for subjective quality, the quantization MSE has been reduced compared with the Max quantizers. For the unmatched statistics the performance of the DPCM does not deteriorate significantly when using the system which was optimized for a particular image, to encode a second image with similar statistics to the first. This was illustrated by using the system optimized for picture B to encode pictures A and C. However if the input statistics differ significantly from the statistics for which the system is designed, the reconstructed image will be degraded. The coding system optimized for picture A is used to encode picture C and slope overload-effects have been observed. When the system of picture C is used to encode picture A, granular noise occurred.

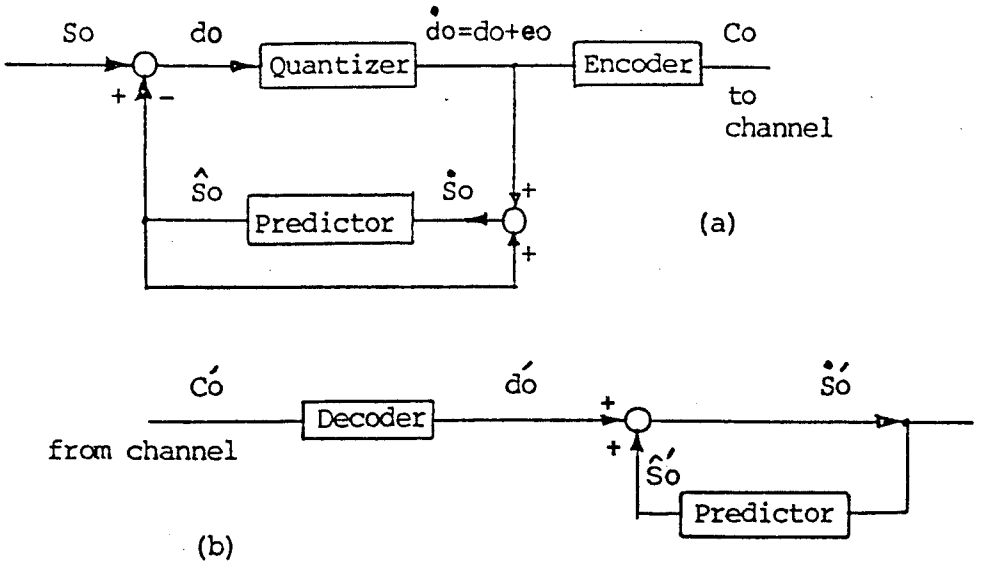
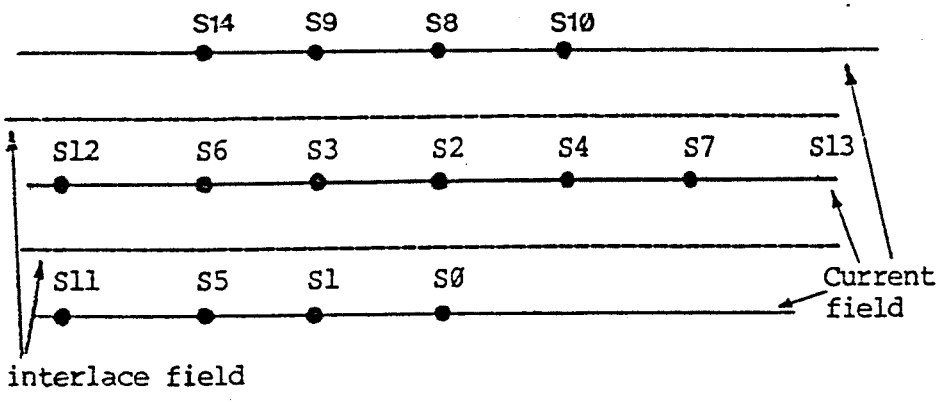
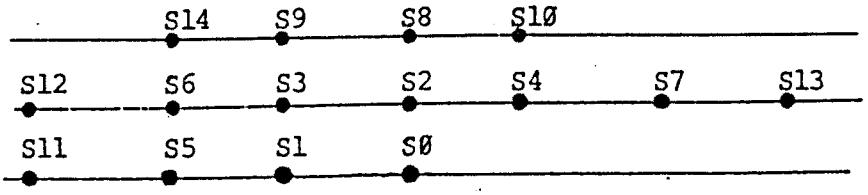


Fig.2.2.1 DPCM image coding system, (a) coder, (b) Decoder



(a)



(b)

Fig.2.2.2 Location of picture elements, (a) interlaced, (b) non-interlaced.

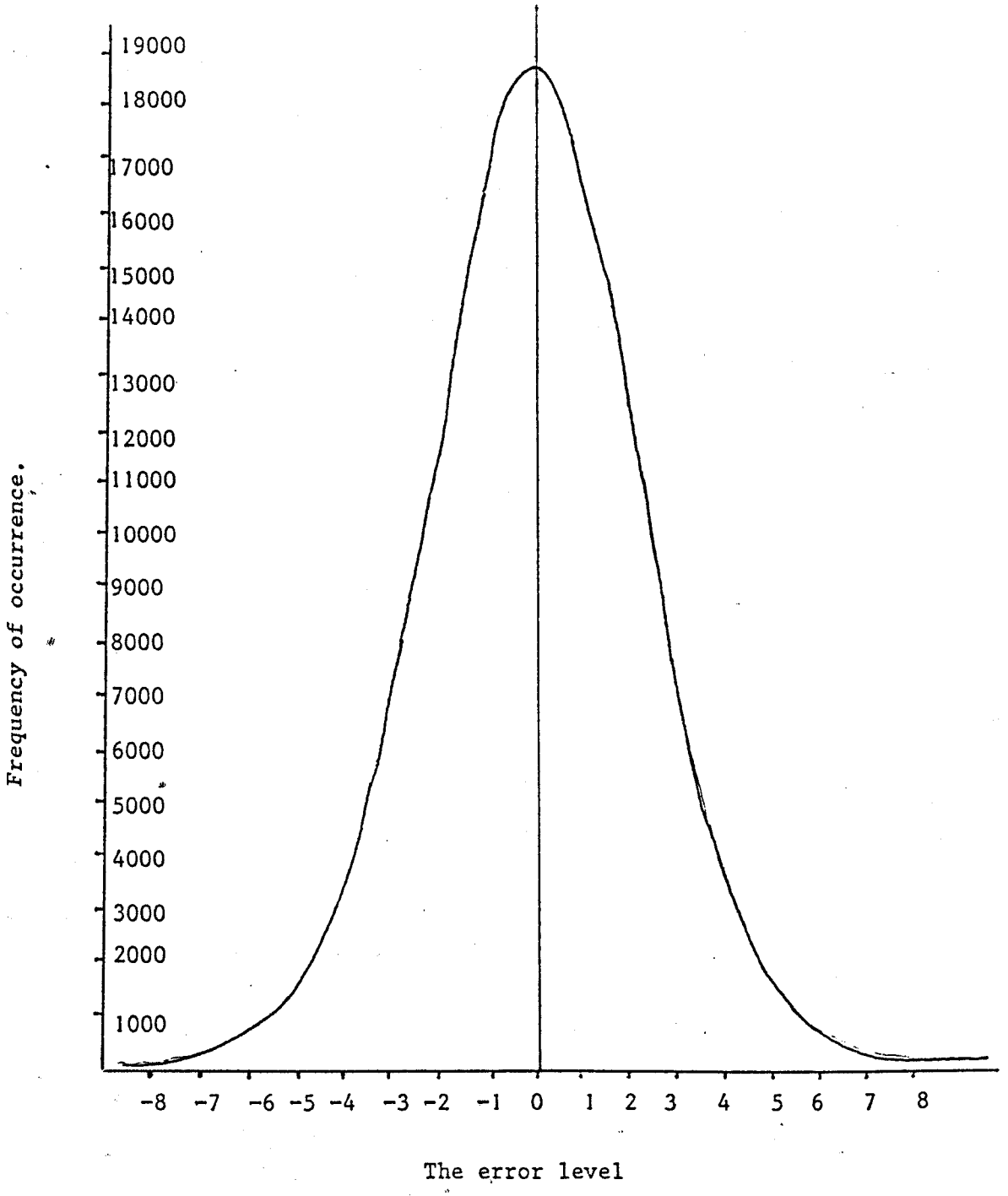


Fig.2.2.3 A typical histogram of differential signal (prediction error).

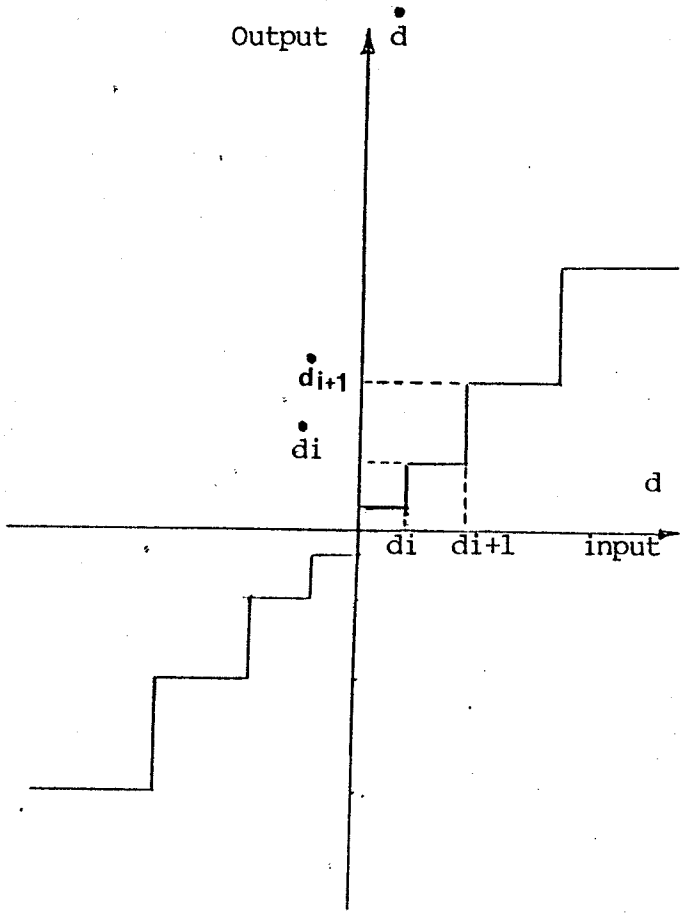


Fig. 2.3.1 Characteristics of a quantizer. d is the input and $\dot{d}$ is the output. {di} and {$\dot{d}_i$} are decision and quantization levels

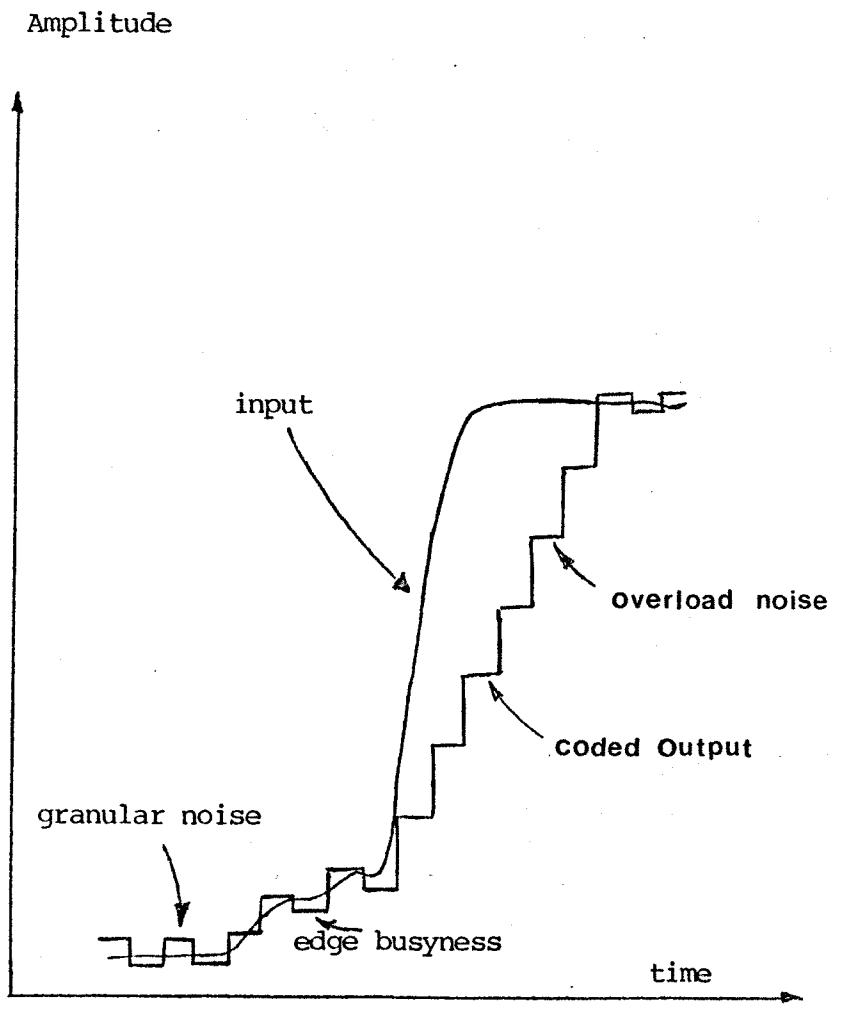
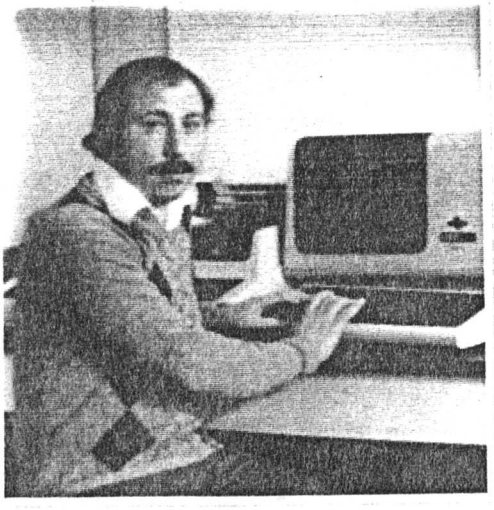


Fig. 2.3.2 Illustration of video waveform and and DPCM response showing granular noise, edge busyness, and slope overload

(a)



(b)

(c)

Fig.2.5.1 The original images of 256x256 pixels and 128 gray levels , (a) low detail, (b) moderate detail, (c) high detail.

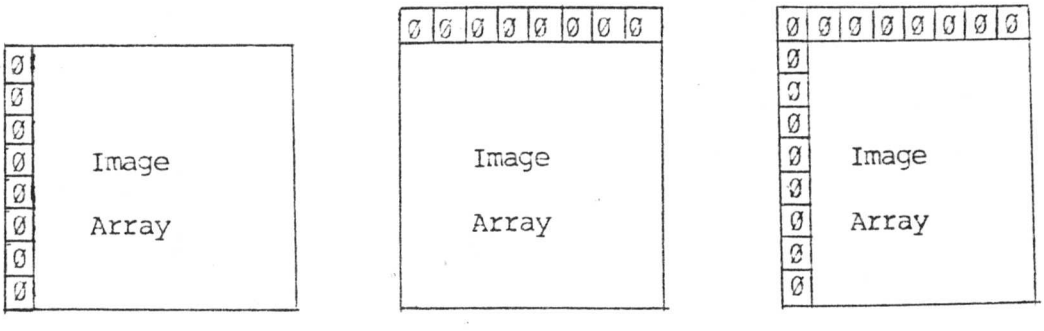


Fig.2.5.2 Adding zeroes as known conditions.

Table 2.5.1 Correlations between So and Sj for the pattern shown in Fig.2.2.2.

Picture A	Picture B	Picture C
RØ1 =Ø.98Ø3	RØ1 =Ø.9892	RØ1 =Ø.9837
RØ2 =Ø.9881	RØ2 =Ø.973Ø	RØ2 =Ø.9775
RØ3 =Ø.969Ø	RØ3 =Ø.9646	RØ3 =Ø.9644
RØ4 =Ø.9681	RØ4 =Ø.9649	RØ4 =Ø.9659
RØ5 =Ø.9531	RØ5 =Ø.97Ø9	RØ5 =Ø.9534
RØ6 =Ø.9432	RØ6 =Ø.9481	RØ6 =Ø.9366
RØ7 =Ø.9415	RØ7 =Ø.948Ø	RØ7 =Ø.9389
RØ8 =Ø.9695	RØ8 =Ø.93Ø2	RØ8 =Ø.9421
RØ9 =Ø.9514	RØ9 =Ø.9229	RØ9 =Ø.9312
RØ1Ø=Ø.9498	RØ1Ø=Ø.9241	RØ1Ø=Ø.9344
RØ11=Ø.9221	RØ11=Ø.9492	RØ11=Ø.9182
RØ12=Ø.9134	RØ12=Ø.9484	RØ12=Ø.9Ø35
RØ13=Ø.9112	RØ13=Ø.9287	RØ13=Ø.9Ø63
RØ14=Ø.9274	RØ14=Ø.9Ø94	RØ14=Ø.9Ø83

Table 2.5.2 Values of mean square prediction error, Redundancy removing factor $20\log\sigma/\sigma_d$ and the weighting coefficients for 10 predictors matched to the three pictures of Fig.2.5.1. σ^2=544.6 for picture A, σ^2=557.9 for picture B and σ^2=648.8 for picture C.

Predictor number	Sample used in prediction Fig.2.2.2	Scene	Mean square error σ^2 Theore-	Actual	$20\log\sigma/\sigma_d$	Prediction weighting coefficients a1	a2	a3	a4	a5	a6	a8
1	S1	A	9.568	9.644	17.50	0.993						
		B	12.158	12.218	16.60	0.991						
		C	22.771	22.908	14.50	0.985						
2	S2	A	10.386	10.484	17.10		0.992					
		B	25.566	25.611	13.20		0.977					
		C	25.544	26.676	13.80		0.980					
3	S1,S5	A	8.184	8.296	18.10	1.370				-0.381		
		B	10.865	10.971	17.10	1.311				-0.324		
		C	19.273	19.480	15.20	1.370				-0.393		
4	S2,S8	A	9.461	9.597	17.60		1.287					-0.299
		B	23.814	23.880	13.70		1.291					-0.322
		C	24.231	24.459	14.20		1.267					-0.927
5	S1,S2	A	5.188	5.252	20.10	0.520	0.477					
		B	9.206	9.253	17.80	0.708	0.291					
		C	14.117	14.221	16.60	0.545	0.454					
6	S1,S2,S3	A	2.241	2.371	23.60	0.883	0.872	-0.756				
		B	5.839	5.912	19.70	0.865	0.719	-0.657				
		C	9.508	9.678	18.26	0.804	0.765	-0.573				
7	S1,S2,S3 S4	A	2.211	2.334	23.68	0.881	0.780	-0.728	0.066			
		B	5.533	5.596	20.00	0.881	0.503	-0.550	0.165			
		C	9.333	9.479	18.60	0.795	0.513	-0.497	0.183			
8	S1,S2,S3 S4,S5	A	2.178	2.300	23.74	0.943	0.765	-0.706	0.061	-0.064		
		B	5.478	5.544	20.00	0.951	0.482	-0.522	0.161	-0.073		
		C	8.654	8.805	18.67	0.890	0.472	-0.448	0.183	-0.099		
9	S1,S2,S3 S4,S5,S6	A	2.164	2.288	23.76	0.928	0.794	-0.692	0.067	-0.062	-0.039	
		B	5.414	5.479	20.10	0.928	0.516	-0.497	0.172	-0.068	-0.053	
		C	8.458	8.607	18.77	0.864	0.534	-0.407	0.201	-0.092	-0.093	
10	S1,S2,S3 S4,S5,S6 S8	A	2.151	2.288	23.76	0.955	0.797	-0.728	0.068	-0.094	-0.038	-0.039
		B	5.236	5.344	20.15	0.945	0.504	-0.356	0.169	-0.034	-0.150	-0.049
		C	8.057	8.498	18.32	0.901	0.521	-0.315	0.200	-0.017	-0.185	-0.089

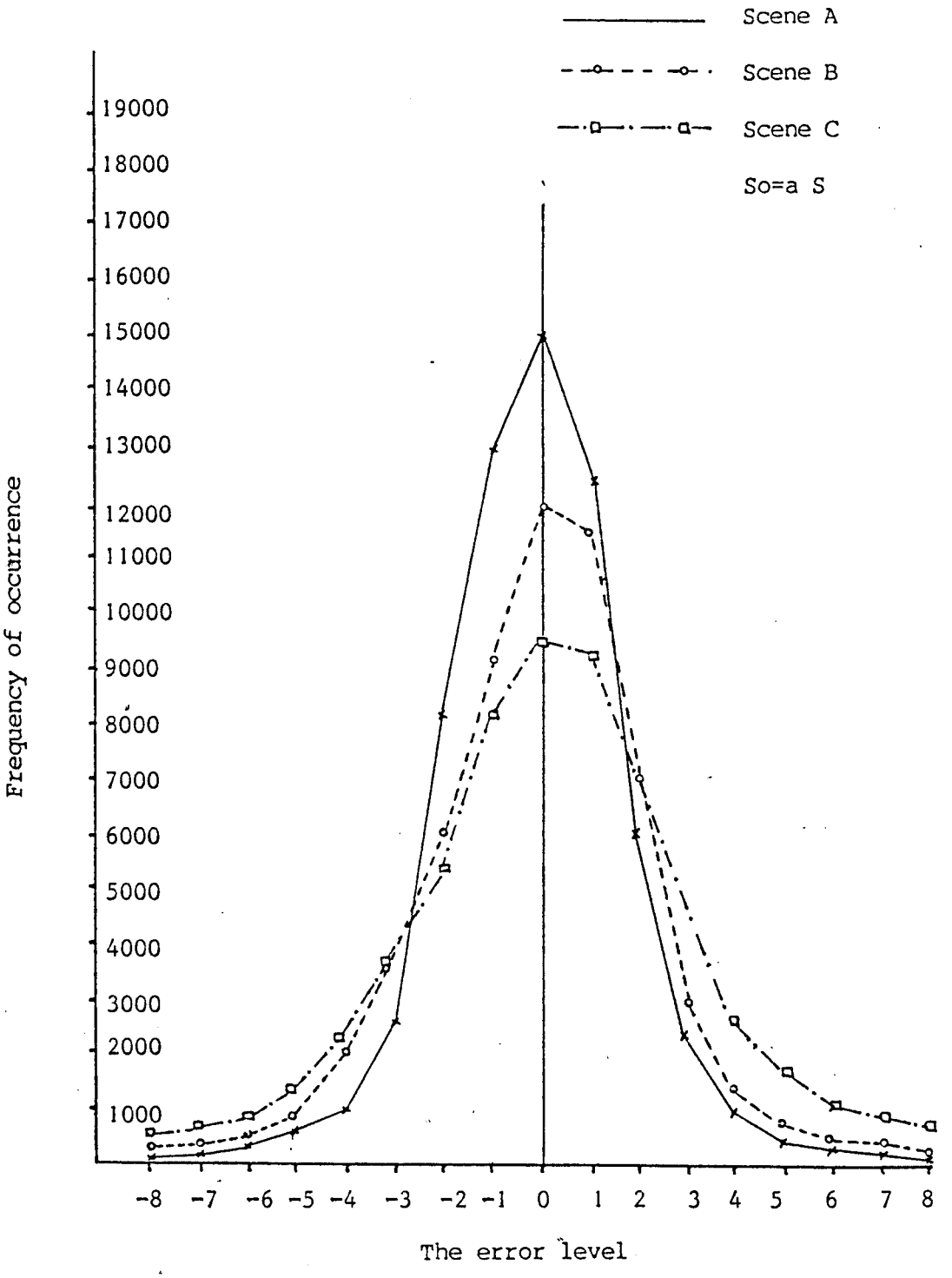


Fig.2.5.3 Histogram of prediction error (first order predictor).

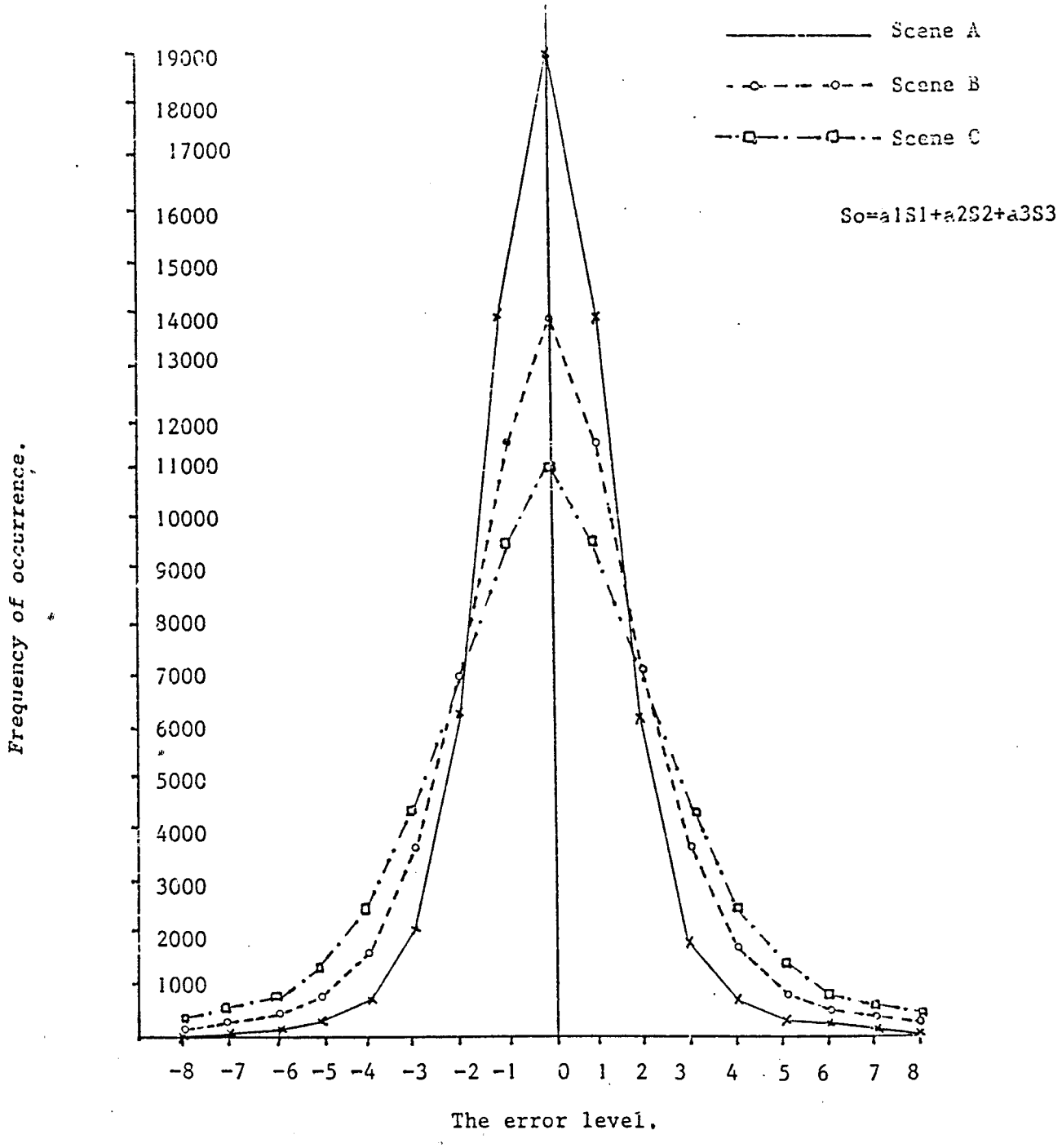


Fig.2.5.4 Histogram of prediction error. (third order predictor)

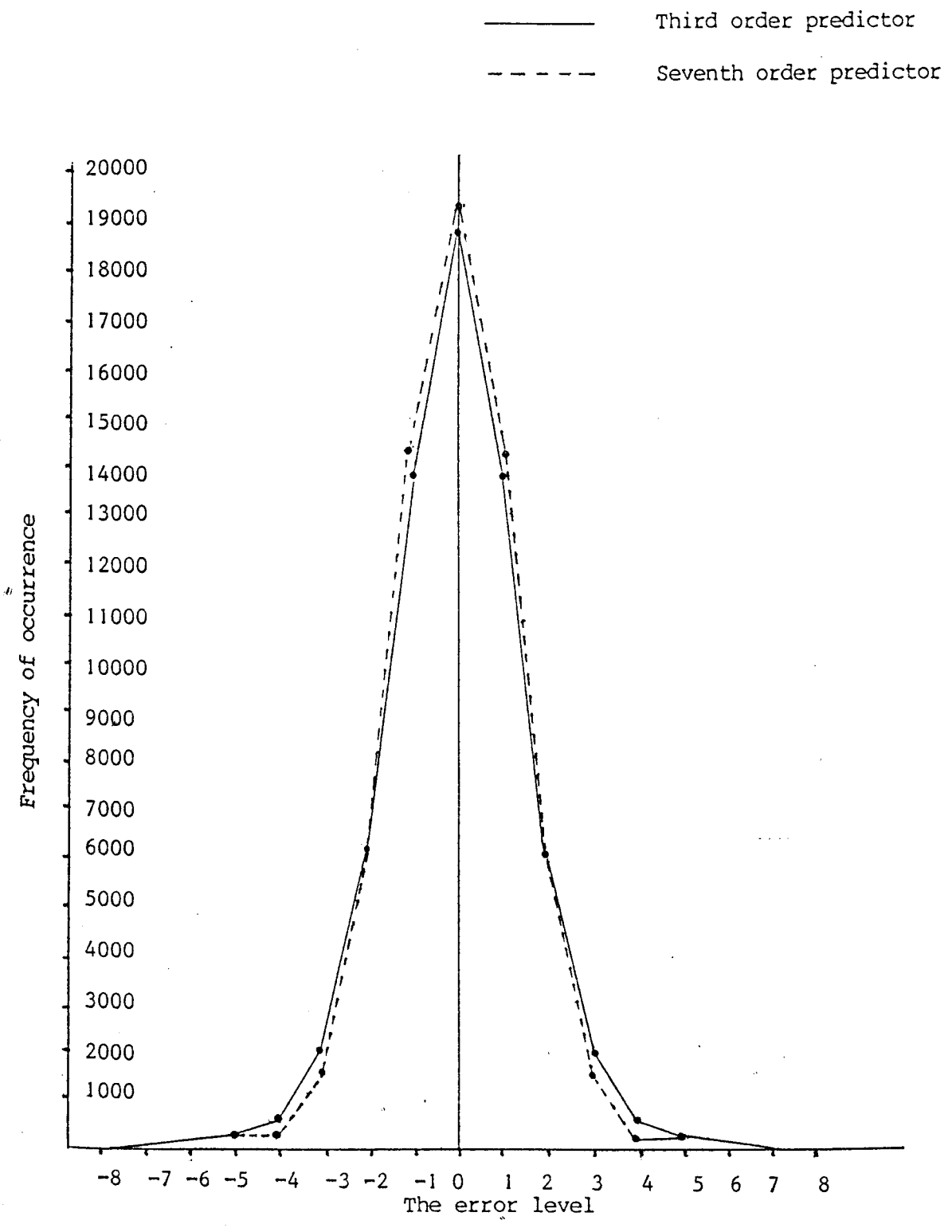


Fig.2.5.5 Histogram of prediction error of the third and seventh order predictors for scene A.

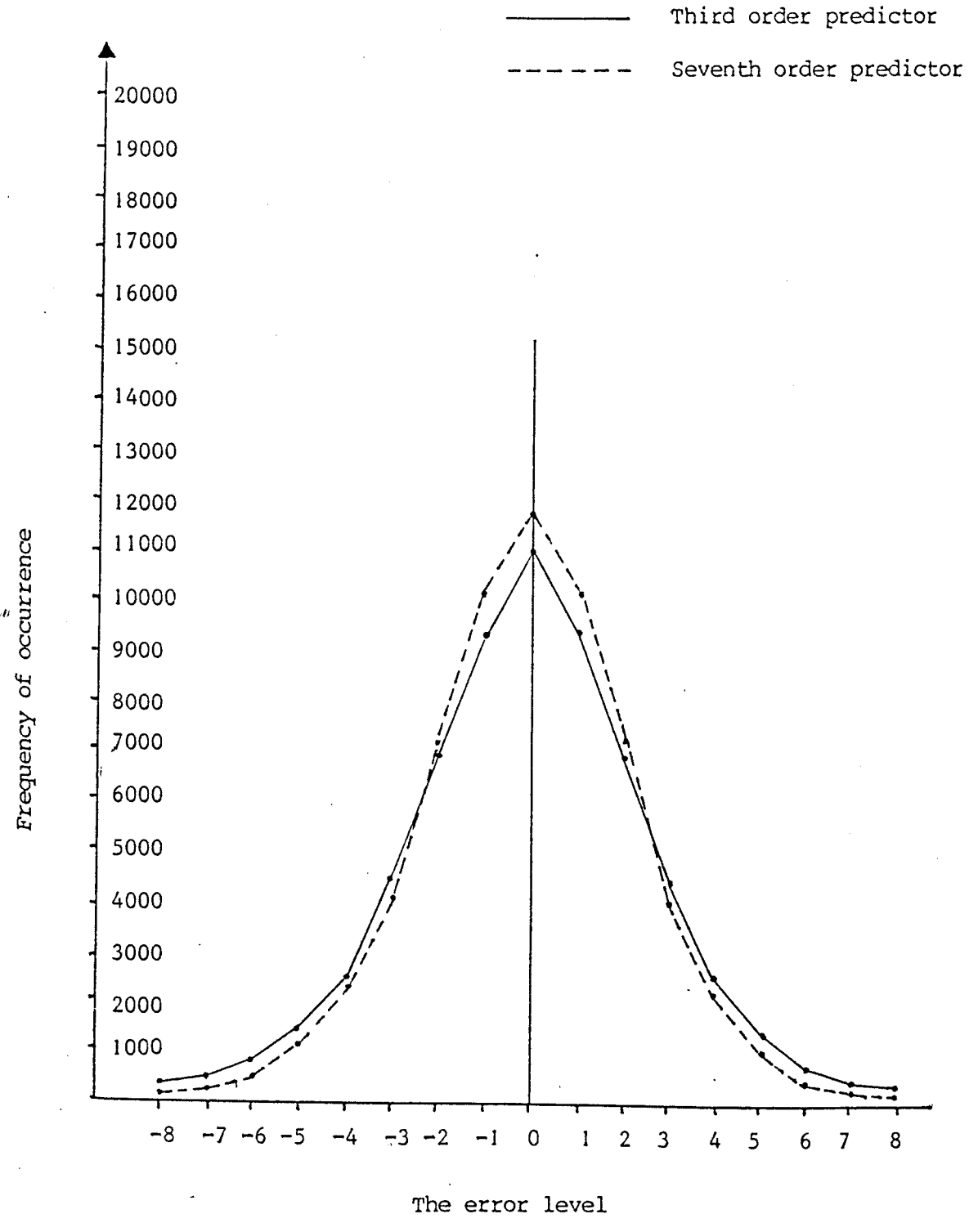


Fig.2.5.6 Histogram of prediction error of the third and seventh order predictors for scene C.

Fig.2.5.7 Prediction error of first order predictor using S1 (Scene A)

Fig.2.5.8 Prediction error of first order predictor using S2 (Scene A)

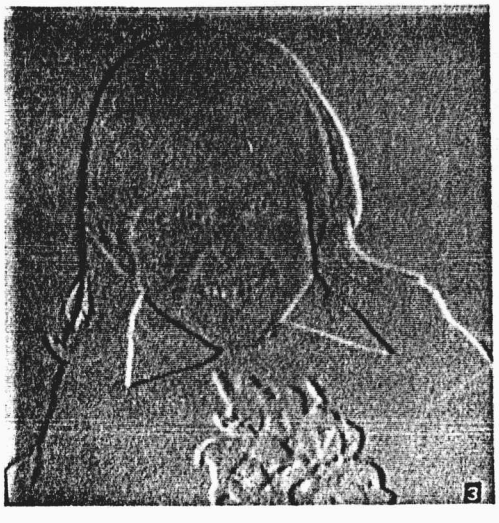


Fig.2.5.9 Prediction error of second order predictor using S1 and S5 (Scene A)

Fig.2.5.10 Prediction error of second order predictor using S2 and S8 (Scene A)

Fig.2.5.11 Prediction error of two dimensional second order predictor using S1 and S2 (Scene A)

Fig.2.5.12 Prediction error of two dimensional third order predictor using S1,S2 and S3 (Scene A)

Fig.2.5.13 Prediction error of two dimensional seventh order predictor using S1, S2, S3, S4, S5, S6 and S8 (Scene A).

Fig.2.5.14 Prediction error of third order predictor (Scene B)

Fig.2.5.15 Prediction error of seventh order predictor (Scene B)

Fig.2.5.16 Prediction error of third order predictor (Scene C)

Fig.2.5.17 Prediction error of seventh order predictor (Scene C)

Table 2.6.1 8-level quantizer characteristics, (a) of picture A, (b) of picture B and (c) of picture C.

	Determined quantizer		Max. quantizer	
	Decision levels	Recontructed levels	Decision levels	Recontructed levels
(a)	127.00	13.00	127.00	4.90
	10.00	7.00	3.78	2.66
	5.00	3.00	1.99	1.32
	2.00	1.00	0.84	0.37
	0.00		0.00	
(b)	127.00	22.00	127.00	7.60
	17.00	12.00	5.86	4.12
	9.00	6.00	3.10	2.10
	4.00	2.00	1.31	0.58
	0.00		0.00	
(c)	127.00	29.00	127.00	9.40
	23.00	17.00	7.25	5.10
	13.00	9.00	3.82	2.54
	6.00	3.00	1.63	0.71
	0.00		0.00	

Table 2.6.2 4-Level quantizer characteristics.

Picture A		Picture B		Picture C	
Decision level	Reconstructed level	Decision level	Reconstrucred level	Decision level	Reconstructed level
127.00	8.00	127.00	13.00	127.00	17.00
5.00	2.00	8.00	3.00	11.00	5.00
0.00		0.00		0.00	

Table 2.6.3 (Peak to peak)^2 to MS quantization noise (db) (see Eq.2.43).

	Picture A		Picture B		Picture C	
	Determined quantizer	Max. quantizer	Determined quantizer	Max. quantizer	Determined quantizer	Max. quantizer
3 bits	47.7	45.8	44.5	42.5	42.5	41.5
2bits	41.6	37.0	38.4	35.4	36.9	35.0

(a)

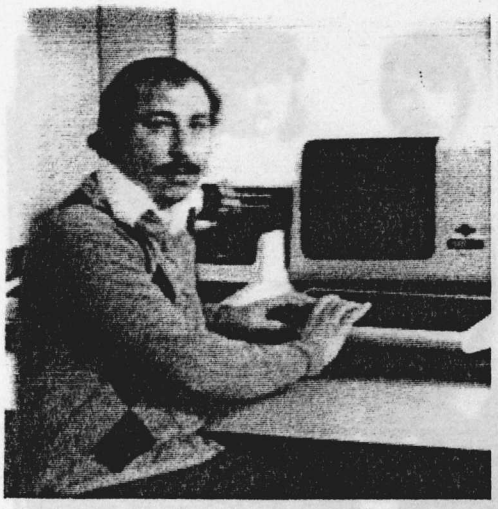
Fig.2.6.1 Third order predictor and 3 bit quantizer (a) original image, (b) and (c) error signal and reconstructed image using Max quantizer, (d) and (e) error signal and reconstructed image using subjective quantizer.

(b)

(c)

(d)

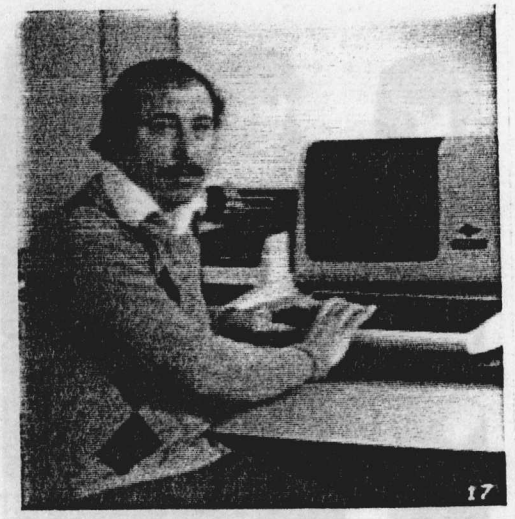
(e)



(a)

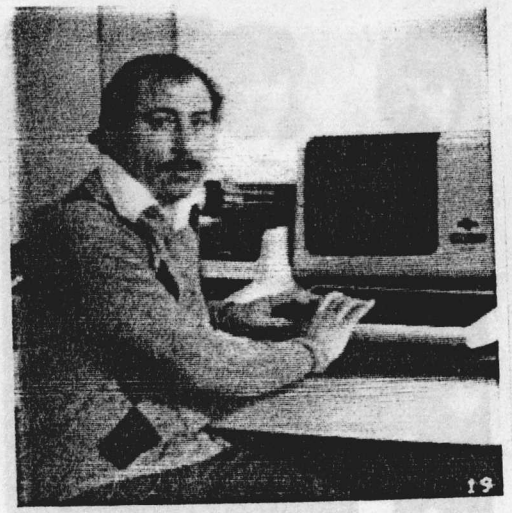
Fig.2.6.2 Third order predictor and 3 bit quantizer (a) original image, (b) and (c) error signal and reconstructed image using Max quantizer, (d) and (e) error signal and reconstructed image using subjective quantizer.

(b)



(c)

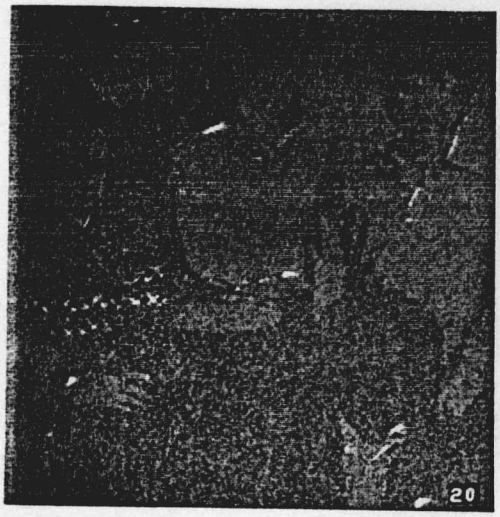
(d)



(e)

(a)

Fig.2.6.3 Third order predictor and 3 bit quantizer (a) original image, (b) and (c) error signal and reconstructed image using Max quantizer, (d) and (e) error signal and reconstructed image using subjective quantizer.



(b)

(c)

(d)

(e)

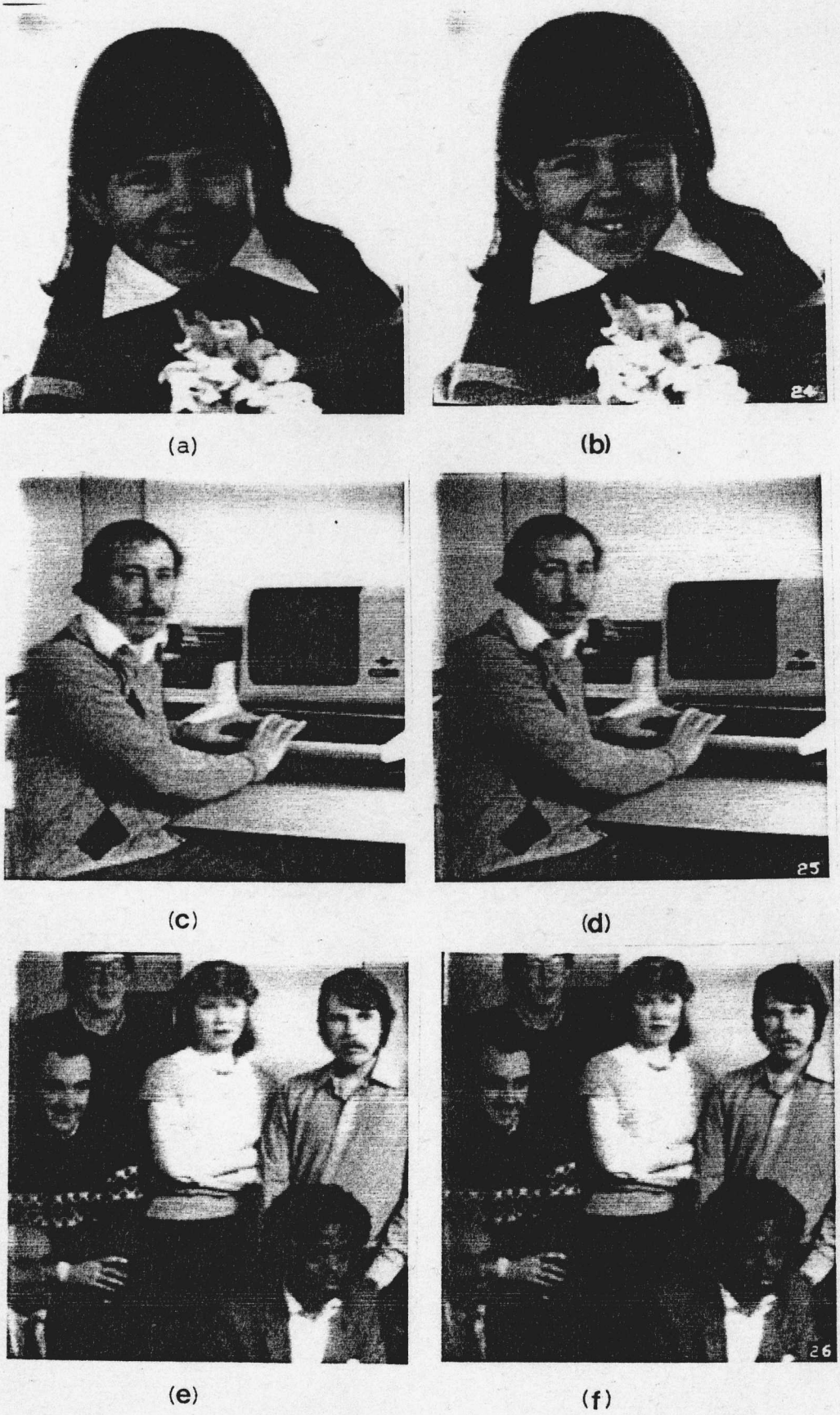


Fig.2.6.4 Third order predictor and 2 bit quantizer. (a), (c) and (e) original images, (b), (d) and (f) reconstructed images

(a) (b)

(c) (d)

Fig.2.7.1 Reconstructed images for unmatched system (a) and (c) original images, (b) and (c) the reconstructed images using the system optimized for coding the picture B.

CHAPTER THREE

3 Differential PCM with Asymmetrical Quantizer Characteristics

3.1 Introduction

With the aim of reducing granular noise and slope overload noise, a new coding system is proposed in this chapter. This system is adaptive quantization with backward estimation, in which the quantizer characteristics are shifted to one side or another, depending on the quantizer output given by the previous horizontal and vertical pixels. After presenting the principle of adaptive quantization and a brief review of some related systems, the motivation for the proposed system is demonstrated by comparing the response of the fixed quantization system with that of the adaptive quantizer for rectangular and sloped edges. Then the principal of the system is described.

Improvement in the quantization noise level for the proposed system is shown by computer simulation, and compared to the non-adaptive system performance.

3.2 Principle of the Adaptive Quantization Systems

It was illustrated in the last chapter that unmatched DPCM systems normaly degrade the reconstructed images. This is because, pictures usually have varying statistical structures, both from image to image and from region to region within an image. Even though a system may be designed to be optimally matched to the overall structural properties of a given image, it would often not be optimal for subimages due to local variations within the image.

Thus, one would expect to improve coding efficiency by adapting the coding strategy to satisfy the requirments above. If the predictor parameters are fixed, the quantizer must continually adapt to changes of signal statistics. This is called "adaptive quantization". The basic idea of adaptive quantization is to let the step size (or in general the quantizer levels and ranges) vary to match the variance of the input signal. An alternative point of view is to consider fixed quantizer characteristics preceded by a time varying gain which tends to keep constant the varying gain which tends to keep constant the variance of the quantizer input. In the first case the step size should follow increases and decreases of the variance of the input. In systems with nonuniform quantization, this would imply that the quantization levels and ranges would be scaled linearly to match the variance of the signal. In the case of

varying gain, which is applicable without modification to both uniform and nonuniform quantizers, the gain changes inversely with changes in the variance of the input to keep the variance of the quantizer input relatively constant. In either case, it is necessary to obtain an estimate of the time varying amplitude properties of the input signal for control of the quantizer.

There are two classes of adaptation, depending on the estimation technique. In one class, the amplitude or variance of the input is estimated from the input itself and the system is called "DPCM with forward adaptive quantization" Since the quantizer input (difference signal dn) is proportional to the input signal, it is reasonable to control the step size or the gain from dn or as depicted in Fig. 3.2.1, from the input Sn.

In the other class of adaptive quantizer, the step size or gain is adapted on the basis of the output of the quantizer $\dot{dn}$, or equivalently, on the basis of the output code words Cn (Fig. 3.2.2). This is called "DPCM with backward adaptive quantization".

In forward adaptation, it will be necessary to send side information about the gain or the step size to allow the receiver to decode the signal. In the backward adaptation, all necessary information is contained in the transmitted sequence.

Jayant [148] discussed a backward adaptive quantization system for speech and picture signals. For every new input sample, the system adapts the step size

based on the previous step size and previous quantizer output. In other words, the present step size is given by the previous step size multiplied by a factor, which depends on the code word magnitude of the quantizer output.

Zetterberg et al.[149] used the Jayant system to adapt the gain instead of step size. Here, for every new input sample, the system controls the gain, which is given by the last available gain muliplied by a factor that depends on the previous quantizer output. Zetterberg extended the algorithm to two dimension by applying the above rule both horizontally and vertically. He also described an adaptive system with forward gain estimation. In this system the image is divided into blocks and for each block the gain is estimated from the block elements. Here, overhead information about the gain must be transmitted. To reduce the overhead information the gain is classified into four intervals and to each interval, a prespecified gain is assigned, which required additional information of 2 bits per block.

Mussmann [103] presented a switched quantizer with forward estimation which does not require side information and has constant code word length. The system used not one quantizer as in standard DPCM but several different quantizers from which, for every new sample, one is chosen for quantizing the different signal. To explain this coding algorithm, let So in Fig. 2.2.2 of chapter two be the amplitude of the next picture element

to be quantized and coded. The previous elements S1, S2, S3, S4 and S5 are known to the transmitter and receiver, and each may take any of K possible values (K is the number of grey-levels). Any special combination of previous elements, say | S1-S4| will form what is called the "control quantity". To each value of the control quantity is assigned a quantizer, which is in general the best for quantizing the prediction error of So associated with that control quantity. Because the number of values that the control quantity can take, is large, the control quantity is divided into limited ranges and so the number of quantizers is reduced.

Another system proposed by Limb [150] is based on sign prediction. Here a nine level quantizer with a mid zero value is used, and the algorithm is based on the fact that the probability of having an outside level preceded by a level of the opposite sign, is small. Consequently, the sign of an outer level can be predicted fairly accurately by assuming that it has the same sign as the preceding sample. This means that the same code word is assigned to both outer levels. The polarity is extracted from the previous code word. If the prediction is wrong, then the next best level is assigned.

As mentioned earlier, the major problems in predictive coding for image signals are granular noise occurring in flat areas of intensity variations and slope overload noise occurring at outlines. In DPCM systems with a fixed quantizer, if the dynamic range of the

quantizer is made small, i.e. a quantizer with fine structure, then the granular noise is reduced but the slope overload is increased. If the dynamic range is large, the overload noise is decreased at the cost of increased granular noise. To reduce both kinds of noise, the system must be adaptive. Some adaptive quantization systems have been reviewed in the introduction.

A new coding system is investigated, in which the system switches between different quantizers, depending on the quantization levels occupied by the previous horizontal and vertical samples. To understand the motivation for developing the system, an overview of the response of fixed quantizer system to outlines is necessary. Then the system is described and its performance will be evaluated subjectively and objectively.

3.3 Response of Fixed Quantizer System to Outlines

We consider systems with a third order predictor employing the pixels S1, S2 and S3 of the pattern in Fig. 2.2.2 and assume that the next pixel is estimated by

$$\hat{S}o = S1 + (S2-S3)/2 \qquad (3.3.1)$$

and, moreover that a 3 bit quantizer with positive and negative maximum quantization levles equal to $\pm$ 20 is used. Fig. 3.3.1a shows the response of the fixed quantizer system to a spatial waveform with horizontal

and vertical edges, in which the intensity of all elements in the righthand lower portion bounded by the line is 0 and that of all other elements is 127. Here, we are concerned with the overload charateristic but not the quantization error, and hence we can assume that pixels whose prediction error may be coded by quantization levels other than the maximum are reproduced without noise. The figure shows that the response of the non-adaptive system to rectangular outlines results in high error and edge blurring. Fig. 3.3.2a shows the equivalent response for a sloping edge, which exhibits similar characteristics.

3.4 System Description

The system is edge adaptive and makes use of the fact that the probability of having an outer level followed by outer level of opposite sign is small. This fact can be exploited to assign more levels on one side than the other side. If an outer level of,say, positive sign is encountered, then, for coding the next pixel, a quantizer with, for example, six positive levels and two negative levels is used. On the other hand, if a negative outer level is produced, then a quantizer with six negative levels and two positive levels is employed in coding the next pixel. This will cause the error to decay rapidly and reduce the overload noise. The system

initially employs a symmetrical quantizer which is responsible for coding flat areas and, when an outer level is reconstructed, the system switches to the positive or negative asymmetric quantizer to code the following pixel, depending on the sign of the outer level. Fig. 3.4.1 shows some of the possibilities for arranging the quantizer characteristics. The asymmetric quantizer is called six of eight quantizer, which means that 6 levels of 8 are used to reduce the overload noise.

When the output level of the 6 of 8 mode is below (for positive quantizer) or above (for negative quantizer) a prespecified threshold, the system return to the normal mode for coding the next pixel.

The system can be extended to two dimensional adaptive quantization, where the quantizer output levels for both horizontal and vertical pixels have to be considered.

We define states as

Symmetric Quantizer level	State
positive extremum	+ 1
negative extremum	- 1
intermediate levels	Ø

Asymmetric quantizers	Positive	Negative
> threshold	+1	Ø
= threshold	+1	-1
< threshold	Ø	-1

nine combinations can result from the horizontal and vertical states, as shown in Fig. 3.4.2. These combinations are grouped into sets, each of which determines the quantizer that is to be used to code the pixel under consideration (So) as indicated in Fig. 3.4.2.

For example, if So is the pixel to be coded and the joint state of the horizontal and vertical pixels is (0,0), (-1,1) or (1,-1), then the system operates in nornal mode (symmetrical quantizer) to code the pixel So. If the joint state is (1,0), (0,1) or (1,1), then the system employs the positive quantizer. On the other hand, if the joint state is (-1,0), (0,-1) or (-1,-1) the negative quantizer is used to code the pixel So.

The system, as with all systems using backward estimation, does not require overhead information to be communicated to the receiver about the quantizer used. The information can be extracted from the transmitted sequence.

3.5 Consideration of the Quantizers

The determination of the quantizer characteristics is of major importance for successful operation of the system. We first consider the value of the maximum level (Lmax) in Fig. 3.4.1. The value of Lmax should be small in order to detect transitions as soon as possible. However, a small value of Lmax will increase the probability of an outside level being followed by an outside level of the opposite sign, which can lead to a large error. To explain this, let us assume that Lmax is equal to 3. Differences of about 3 will be recoded as transition and give rise to the system switching to the positive quantizer to code the following pixel. In this quantizer a large number of levels are assigned to positive differences and few levels to negative difference. Since in this case, the probability is high that the next pixel will create a large negative differential signal and only a few levels are avialable to code this signal, a large error may result. On the other hand, larger value of Lmax, mean that the levels are coarsely spaced for coding smooth areas and granular noise will be introduced. In fact, the value of Lmax, depends on the picture content, but a good choice proved to be Lmax $\cong$ 10% of the total signal range.

Some aspects must be considered when determining the levels of the asymmetrical quantizers; firstly not to shift all levels to one side but to leave few on the

other side for the case that the sign of the differential signal may change from one pixel to the next; secondly a reasonable number of levels must be assigned to code small differences for the case when the differential signal changes from large to small value and finally the remaining levels must cover a large range for coding large differences. Some of the possible assignments which satisfy the above requirements are shown in Fig. 3.4.1.

Following this strategy, the response of the adaptive system is shown in Figs. 3.3.1b and 3.3.2b, for the rectangular and sloped edges. It is easy to see that the error pattern decays more rapidly than that of non-adaptive system. The maximum level (Lmax) of the symmetric quantizer was set to 15 and the levels Ln = -Lp of the asymmetric quantizers were set to 40. The level assignment of Fig. 3.4.1b was used, in which three levels of each asymmetric quantizer cover the range of the symmetric quantizer. The threshold, which defines a transition was set equal to Lmax.

3.6 Simulation Results and Discussion

The proposed DPCM system was simulated on a computer for coding the pictures of Fig. 2.5.1 in chapter two, and its performance was investigated using the 8 of 8, 7 of 8 and 6 of 8 quantizers. Two possibilities of quantization levels assignment for the 6 of 8 quantizer,

and the signal to quantization noise performance are shown in table 3.6.1.

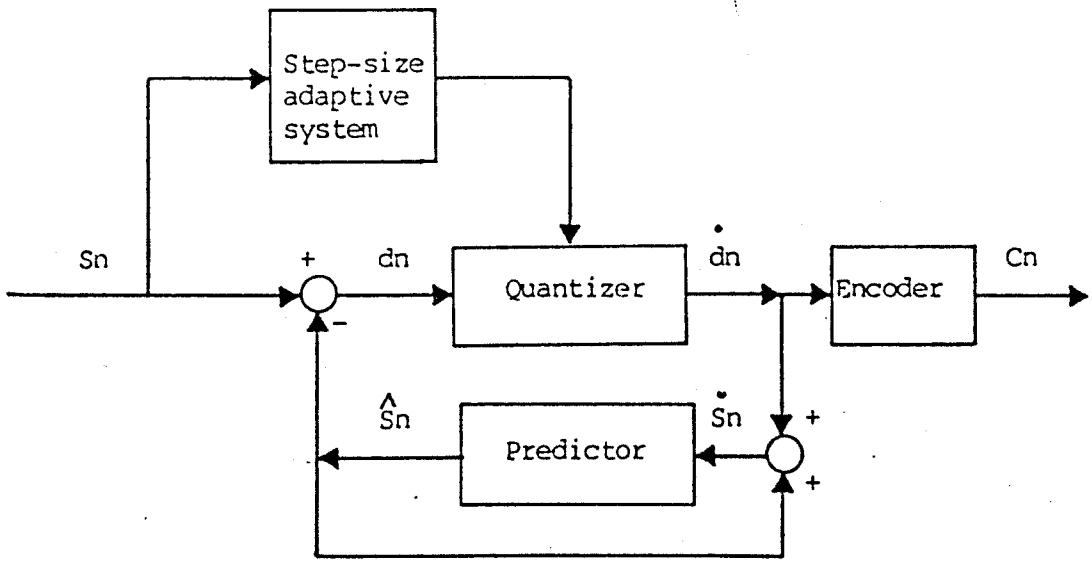
The last column represents the signal to quantization noise ratio and gain, against the fixed optimum system. for picture A, B and C respectively.

To evaluate the success of the system, its performance is compared with that of the fixed quantization system discussed in chapter two. The reconstructed images of the 8 of 8 and 7 of 8 adaptive systems show subjective improvement, though the mean square quantization error was not improved. Better subjective quality of the output images and objective improvement have been achieved with the system using the 6 of 8 quantizer (version I). Fig. 3.6.1 shows the reconstructed images obtained with the 6 of 8 adaptive quantizer and nonadaptive systems. The improvement in signal to noise ratio for picture (A) is about 1.5 db, which is mainly due to slope overload reduction and less granular noise. This is because the symmetrical quantizer, responsible for coding smooth areas has the same characteristics as the optimal quantizer designed for picture (A) in chapter two. However, the improvements of 2.3 db for picture (B) and 2.7 db for picture C are better than for picture (A). Since with the adaptive quantizer, larger differences can be coded and small differences more finely quantized than in the system with a fixed quantizer, the improvement is caused again by the reduction of slope overload and granular noise. Fig.

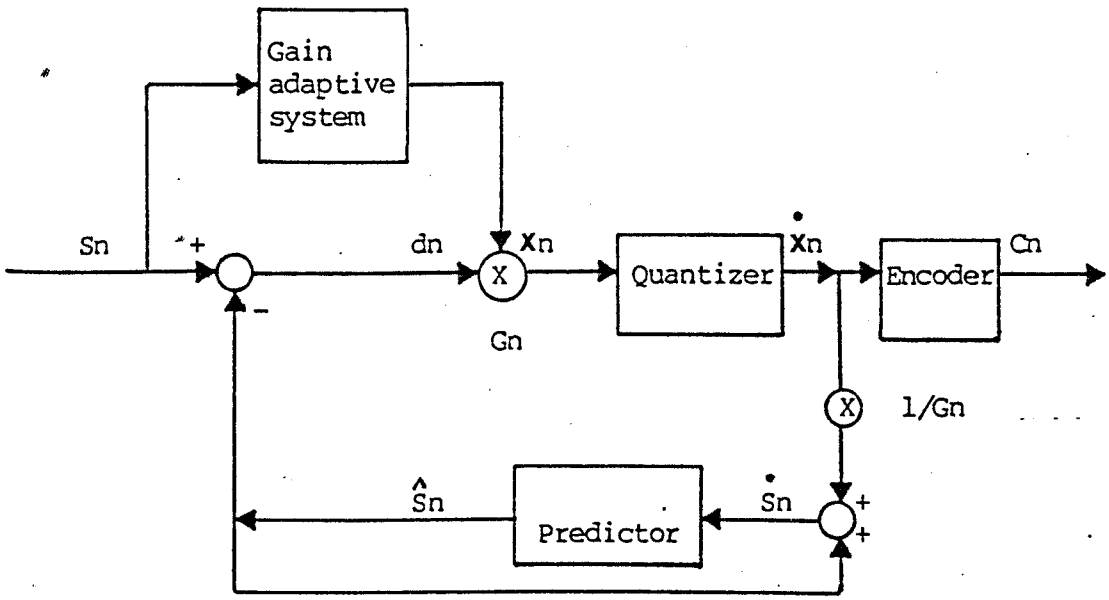
3.6.2 shows the reconstructed images of the adaptive and nonadaptive system for picture (B) and Fig. 3.6.3 for picture (C), where the threshold was set equal to Lmax. It is not difficult to see the inprovement in edge reproduction on all images.

The reconstructed images of the adaptive system using the version II quantizer showed the same quality as that of the version I quantizer. The signal to noise ratio has been slightly deteriorate. The system discussed so far, employs optimum prediction coefficients calculated for each image.

More important is to evaluate the performance of the system when using a non-optimum predictor. The system; was simulated for the cases where the prediction coefficients of picture (A), calculated in chapter two, are used to reconstruct picture (C) and the prediction coefficients of picture (C) are used to code picture (A). In contrast to the fixed system, the reconstructed images illustrated in Fig. 3.6.4. Show no noticable degradation over the whole image, which indicates that the system is still matched to the local variations, because of the adaptivity of the quantization. However, the signal to noise ratio was degraded, but it is still comparable with that of fixed optimum system.

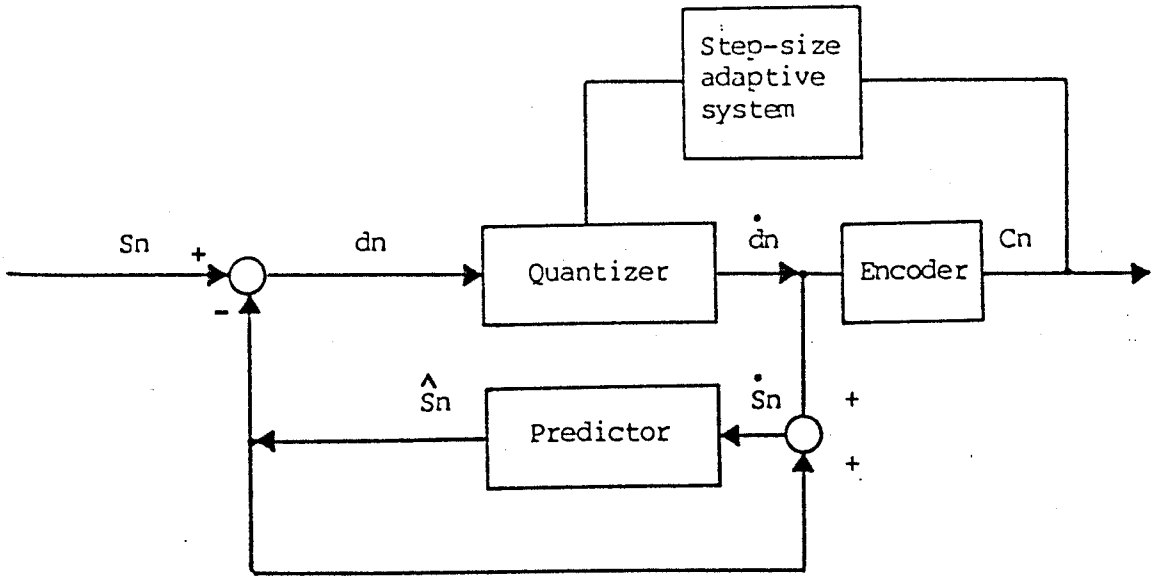


(a) DPCM system with adaptive step-size

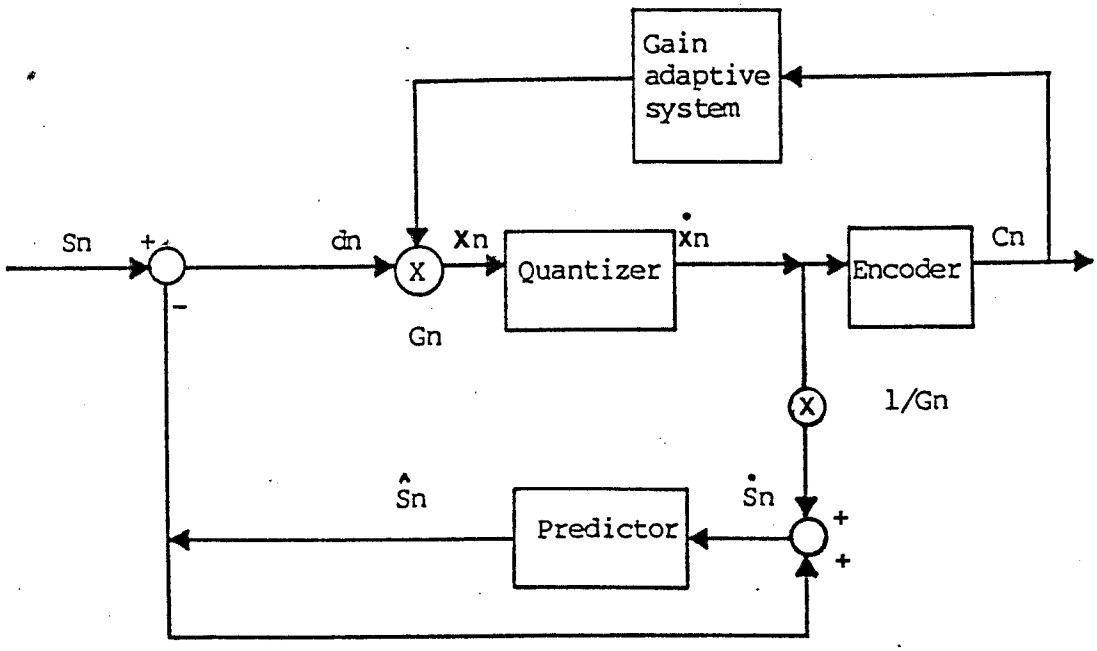


(b) DPCM System with adaptive gain Gn.

Fig 3.2.1. ADPCM System with forward adaptive quantization



(a) DPCM system with adaptive step-size



(b) DPCM System with adaptive gain Gn.

Fig 3.2.2. ADPCM System with backward adaptive quantization

127	127	127	127	127	127	127	127	127	
127	107	87	67	47	27	7	Ø	Ø	
127	97	67	37	7	Ø	Ø	Ø	Ø	
127	92	57	22	Ø	Ø	Ø	Ø	Ø	
127	89	51	13	Ø	Ø	Ø	Ø	Ø	(a)
127	88	49	1Ø	Ø	Ø	Ø	Ø	Ø	
127	87	47	7	Ø	Ø	Ø	Ø	Ø	
127	87	47	7	Ø	Ø	Ø	Ø	Ø	
127	87	47	7	Ø	Ø	Ø	Ø	Ø	

127	127	127	127	127	127	127	127	127	
127	112	82	42	2	Ø	Ø	Ø	Ø	
127	79	24	Ø	Ø	Ø	Ø	Ø	Ø	
127	63	Ø	Ø	Ø	Ø	Ø	Ø	Ø	
127	55	Ø	Ø	Ø	Ø	Ø	Ø	Ø	(b)
127	51	Ø	Ø	Ø	Ø	Ø	Ø	Ø	
127	49	Ø	Ø	Ø	Ø	Ø	Ø	Ø	
127	48	Ø	Ø	Ø	Ø	Ø	Ø	Ø	
127	47	Ø	Ø	Ø	Ø	Ø	Ø	Ø	

Fig.3.3.1 Response of the prediction system $\hat{S}o=S1+(S2-S3)/2$, (a) fixed quantization, (b) variable quantization.

				127	127	127	127	127	127	127	127	127
				127	127	107	87	67	47	27	7	0	0
			127	127	107	77	47	17	0	0	0	0	0
		127	127	107	77	42	7	0	0	0	0	0	0
	127	127	107	77	42	4	0	0	0	0	0	0	0
127	127	107	77	42	4	0	0	0	0	0	0	0	0
127	107	77	42	4	0	0	0	0	0	0	0	0	0

(a)

				127	127	127	127	127	127	127	127	127
				127	127	112	82	42	2	0	0	0	0
			127	127	112	64	4	0	0	0	0	0	0
		127	127	112	64	0	0	0	0	0	0	0	0
	127	127	112	64	0	0	0	0	0	0	0	0	0
127	127	112	64	0	0	0	0	0	0	0	0	0	0
127	112	64	0	0	0	0	0	0	0	0	0	0	0

(b)

Fig.3.3.2 Response of the prediction system to sloped edges $\hat{S}0=S1+(S2-S3)/2$, (a) fixed quantization, (b) variable quantization.

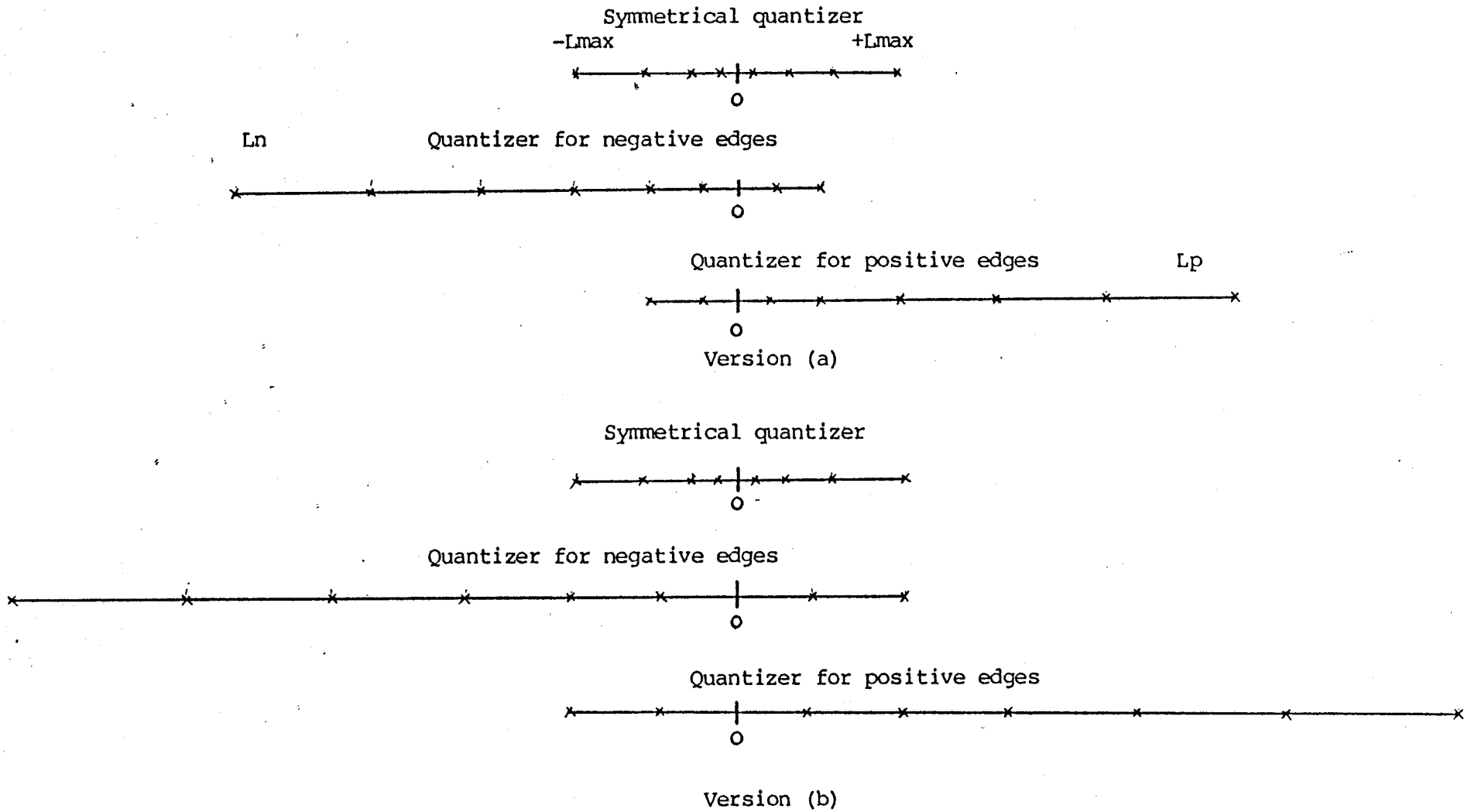


Fig.3.4.1 Possibilities of arranging the six of eight asymmetrical quantizer characteristics (Ln=-Lp) (a) 3 levels cover the range of the symmetrical quantizer, (b) 2 levels cover the range of the symmetrical quantizer.

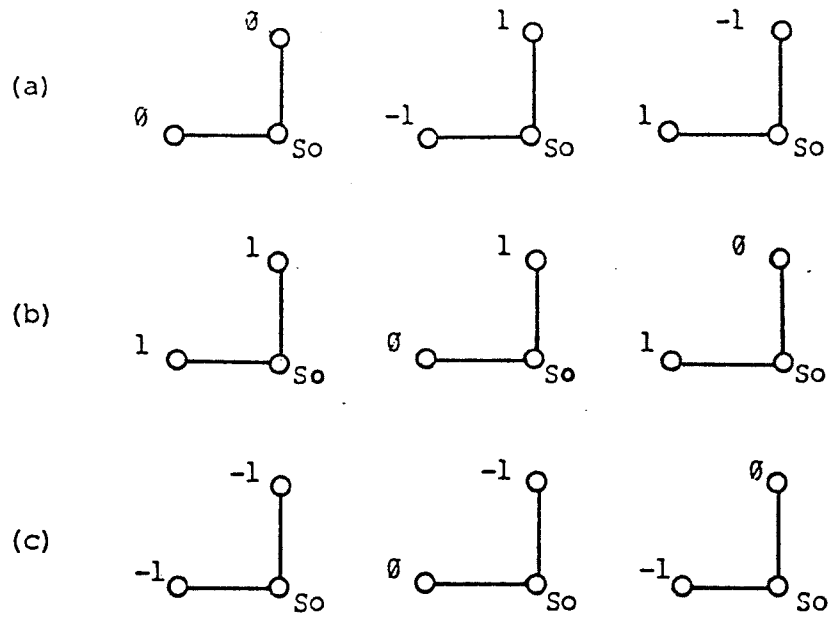


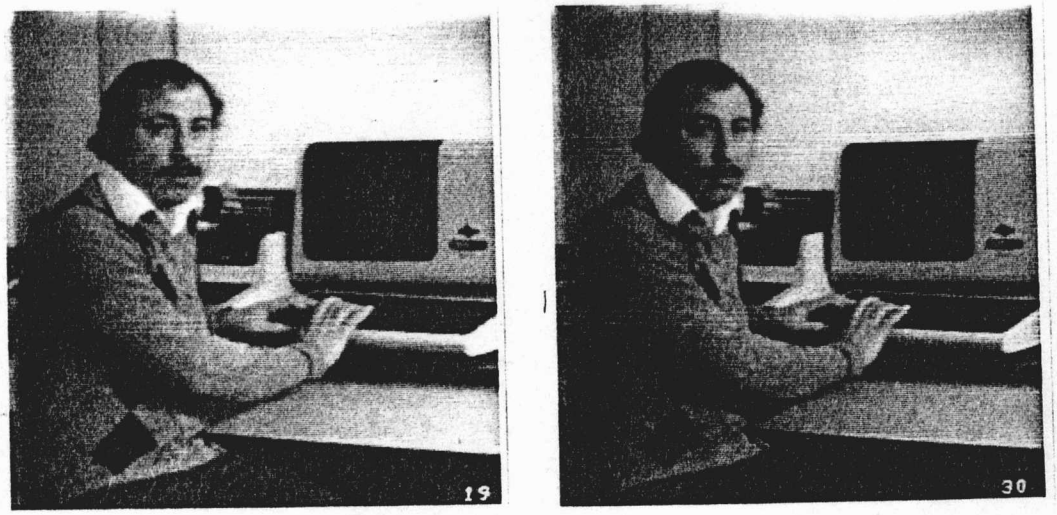
Fig.3.4.2 Joint states of the horizontal and vertical pixels. (a) transfer to symmetrical quantizer. (b) transfer to asymmetrical quantizer for positive edges, (c) transfer to asymmetrical quantizer for negative edges.

Table 3.6.1 6 of 8 quantizer characteristics and its signal to noise ratio.

Quantizers	Quantization levels								S/N
	Version I								
Symmetric	-13	-7	-3	-1	1	3	7	13	49.2, 1.5
Negative	-49	-33	-21	-13	-6	-2	2	6	46.8, 2.3
Positive	-6	-2	2	6	13	21	33	49	45.2, 2.7
	-13	-7	-3	-1	1	3	7	13	49.2, 1.5
	-64	-47	-33	-22	-13	-6	2	6	46.8, 2.1
	-6	-2	6	13	22	33	47	64	45.2, 2.3

(a) (b)

Fig.3.6.1 The reconstructed images, (a) using optimum non-adaptive system, (b) using adaptive quantization system with optimum predictor.



(a) (b)

Fig.3.6.2 The reconstructed images, (a) using optimum non-adaptive system, (b) using adaptive quantization system with optimum predictor.

(a) (b)

Fig.3.6.3 The reconstructed images, (a) using optimum non-adaptive system, (b) using adaptive quantization system with optimum predictor.

(a) (b)

Fig.3.6.4 The reconstructed images using adaptive quantization system with unmatched predictors.

CHAPTER FOUR

4 Adaptive Prediction System

4.1 Introduction

In this chapter a new adaptive linear predictor is presented to improve the prformance of the differential pulse code modulation (DPCM) applied to image compression.

In this system, each point is tested to see if it is an edge point, and accordingly the pixels employed in the prediction are updated to minimize the error between the predicted and the true value and thus reducing the quantization noise. The test is based on observing certain characteristics of already transmitted neighbouring elements. The procedure of the proposed predictor will be illustrated after introducing some of the adaptive predictors.

Performance measurement, based on the error histogram, the mean square of the prediction error, the quantization mean square error and the subjective quality of the reconstructed images, is demonstrated by computer simulation.

4.2 The Adaptive Predictor

Another approach for improving the performance of DPCM systems is by adapting the predictor to match the statistics of the input signals.

Analogous to adaptive quantizers, adaptive predictors can be classified as forward or backwards adaptive. For the forward adaptive predictors, the information is extracted from the input signal, while for backward adaptive predictors, the information is taken from the quantizer output.

Various adaptive prediction systems have been proposed in the literature, which either, continuously update the prediction coefficients or switch between several predictors. Gibson [151] described an adaptive predictor for speech signals, in which the prediction coefficients are continuously determined at the transmitter and the receiver, based on the past signal estimate and quantizer output.

Adaptive image predictors typically use switched prediction, where both transmitter and receiver have a bank of M possible predictors and adaption consists in switching to one of these predictors, such that each one of them will give small prediction error. Examples of this type of approach are the predictors used by Graham [85] and Conner [87] , in which either the previous line or the previous element is used for the prediction, and the switching is done by the surrounding line and element

differences as shown in Fig.4.2.1.

Another variation [152] in adaptive prediction is to use a weighed sum of several predictors, whose weights are switched from element to element depending on certain properties of already transmitted neighbouring pixels. As an example, assume that the pixel A (Fig.4.2.1) is already transmitted. The prediction error of A for each predictor in a set of predictors is then evaluated and the predictor that gives the least prediction error is used for the prediction of "X". The same calculation can be performed at the receiver and therefore, the predictor switching information does not need to be transmitted.

We present a new adaptive predictor, in which the prediction coefficients and the position of the pixels used for the prediction are fixed, but the pixel values are overweighted or underweighted, depending on the pixel under consideration, whether it is a point of black to white or of white to black transition. The detection of a transition and the pixel updating are based on already transmitted neighbouring elements, and therefore the system does not need overhead information.

4.3 System Procedure.

In this section we demonstrate on an example the predictibility of the edge points using the standard DPCM predictors. Then we describe the edge detection and

updating procedures with the improvement that may be achieved by using the adaptive system.

Fig.4.3.1 shows four possible edges which are most frequently encountered in real life pictures. The pixel values denoted by "Ø" represent black and those by "X" represent white intensities.

Let us use three pixels: the horizontal, vertical and diagonal for the prediction and assume that the prediction coefficients are 1, 1 and -1 respectively.

The predicted value for the nonadaptive predictor of the horizontal and vertical edge points P1-P8 in Figs.4.3.1a and 4.3.1b is close to the true value, which means a small prediction error. For example, the error of P2 is defined

$$e2=P2-(P1+P12-P11)$$

In Fig.4.3.1c the points P11-P27 can be predicted very well, whereas the errors associated with the points P1, P3, P6 and P7 have large positive values. The error of P3, for example, is given by

$$e3=P3-(P15+P17-P14)$$

Also the prediction of the points P4, P7 and PlØ produce a large positive error. Following the same procedure, the prediction of the points P5, P8 and PlØ in Fig.4.3.1c results in a large positive error, whereas the

prediction of the points P16, P2Ø, P24 and P27 will produce a large negative error.

The system predictibility may be improved (i.e. prediction error reduced), if the pixel to be coded, is examined to determine, whether it is an edge point and if the edge is increasing or decreasing grey level. When the pixel under consideration appears to be an edge point, then the pixel used for the prediction are altered so that the edge becomes smooth.

To explain this algorithm, let P1 in Fig.4.3.2 be the pixel to be coded, we calculate the following means:

(a)

$$M1=(P2+P5+P6)/3$$

$$M2=(P3+P4+P8+P9+P1Ø)/5 \qquad 4.3.1$$

or

(b)

$$M1=(P2+P5+P6+P7)/4$$

$$M2=(P3+P4+P8+P9+P1Ø+P11+P7)/7 \qquad 4.3.2$$

According to certain conditions we define state as:

	Condition	State
(1)	$\|M1-M2\|>LI$	
	$M1-M2<Ø$	decreasing grey level edge.
(2)	$\|M1-M2\|>LI$	
	$M1-M2>Ø$	increasing grey level edge.
(3)	$\|M1-M5\|\leq LI$	No edge.

Where LI is a positive threshold and | | means the absolute value.

For updating the pixels P2, P5 and P6 used for prediction of the pixel P1, we adopt one of the following two strategies:

Strategy I

(i) $|M1-M2|>LI$ $\quad \underline{P}i=\text{minimum } (Pi,M1)$

$M1-M2<\emptyset$

(ii) $|M1-M2|>LI$ $\quad \underline{P}i=\text{maximum } (Pi,M1)$

$M1-M2>\emptyset$

(iii) $|M1-M2|\leq LI$ $\quad \underline{P}i=Pi$

Strategy II

(i) $|M1-M2|>LI$ $\quad \bar{P}=\text{average value of pels } Pj, Pj<M1$

$M1-M2<\emptyset$ $\quad \underline{P}i=\text{minimum } (Pi,\bar{P})$

(ii) $|M1-M2|>LI$ $\quad \bar{P}=\text{average value of pels } Pj, Pj>M1$

$M1-M2>\emptyset$ $\quad \underline{P}i=\text{maximum } (Pi,\bar{P})$

(iii) $|M1-M2|\leq LI$ $\quad \underline{P}i=Pi$

where Pi and $\underline{Pi}$ are the old and new values respectively of the pixels P2, P5 and P6 shown in Fig.4.3.2. Pj are the pixels determining Ml.

Applying the procedure above using eq.4.3.1 for the mean values and strategy I, for example, the edge points P1-P8 in Figs.4.3.1a and 4.3.1b can be predicted with very small error.

The pixels P1, P3, P6 and P7 in Fig.4.3.1c are still subject to large prediction error, because these edges can not be detected with the above method. However the pixels P4, P7 and P1Ø can now be predicted leading to small error. Likewise, the pixels P5, P8, P1Ø, P2Ø, P24 and P27 in Fig.4.3.1d, which were not reliably predictable for the nonadaptive predictor, may be predicted accurately using the adaptive system. Even though pixel P6 can be detected as an edge point, the edge is not smooothed by the updating procedure and therefore, the error is large.

Nevertheless, when using the formula in eq.4.3.2, all the edge points mentioned above are proved to be good predictors, except the pixels P1 in Fig.4.3.1c and P6 in Fig.4.3.1d.

4.4 Simulation Results and Discussion.

The choice of the operator size i.e. the size of the area occupied by the pixels used for the edge detection, is of major importance. On one side, the size should be large to detect transition of any shape. On the other hand, when the size is large, more than one transition may be included, which may lead to the failure of the edge detection procedure. It seems from this logic, that the size in Fig.4.3.2 is a reasonable compromise.

The choice of the threshold (LI) is another factor in the system design. Small values of the threshold enable the system to detect the transition as efficiently as possible, however, small threshold values are sensitive to noise. A good choice of threshold (LI) proved to be about 5-15% of the dynamic range of the original signal.

The following four systems were simulated on the computer:

(1) M1 and M2 as in eq.4.3.1 with strategy I for the updating.

(2) M2 and M2 as in eq.4.3.1 with strategy II for the updating.

(3) M1 and M2 as in eq.4.3.2 with strategy I for the updating.

(4) M1 and M2 as in eq.4.3.2 with strategy II for the updating.

Each system was tested using different threshold values but the 8% threshold has proved to be the best. To reduce the propagation of quantization error, the prediction coefficients have been chosen slightly different from unity. The same prediction coefficients for all systems were used and equal to Ø.96, Ø.9 and -Ø.9 for horizontal, vertical and diagonal pixels respectivaly.

The aim in using the adaptive predictors is to reduce the mean error between the predicted and true values and thus reducing the quantization noise. Indeed all the systems have improved the mean square error. Systems (1) and (2) achieved a reduction of about 6% compared with that of nonadaptive predictors. The reduction achieved with systems (3) and (4) was about 9%. From the results obtained for the prediction mean square error and the density function, the updating strategy II has shown no improvement over the strategy I. Therefore wa shall discuss the results of system (1) and (3).

A comparison between the error density functions of third order nonadaptive and adaptive predictors for picture "A" of Fig.4.4.1 is shown in Fig.4.4.2. The figure shows that the density function of adaptive predictors is more highly peaked at zero level and less at the tails than that of the nonadaptive predictors. Equivalent results have been obtained for the pictures "B" and "C".

To evaluate the effect of the adaptive predictors

on the quantization noise, the signal peak squared to quantization mean square error ratio was determined. Table 4.4.1 shows the improvement in the signal to noise ratio when using adaptive instead of nonadaptive predictors.

Since the predictors are mainly designed to improve the predictibility of the edge points, the gain in the signal to noise ratio is expected to be higher for signals with more transition contents than for signals with few transitions. Indeed, the values listed in the table support this claim, where the gain for picture "C" is higher than for picture "B" and this is higher than that for picture "A". Furthermore, we recognize that the performance of system (3) is better than that of system (1). This was expected, since system (3) can better detect the edges.

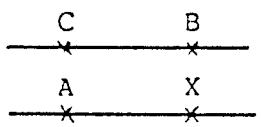
In section 4.3 we demonstrated the response of the adaptive systems to outlines without taking the quantizer effect in consideration. In fact the quantizer will degrade the system detection of the sloped and rectangular edges.

Consider the waveform in Fig.4.4.3 with rectangular edge where "Ø" and "X" represent black and white intensities respectively. Since the pixel P1 can not be detected, its estimated value will differ substantially from the true value, resulting in a large quantization error and the pixel is reconstructed as white intensity. This has the effect that P2 can not be detected and will

be wrongly reconstructed, which in turn hinders the detection of P3. Even more, the falsification of these pixels will prevent detecting P4 and P5 which leads to bad rendition of the pixels. The error extention depends on the value of the quantizer maximum level for a fixed threshold. The larger the maximum level, the sooner the edge can be detected. This is another reason that the gain for picture "C" is better than that for picture "B" and "A", since the maximum level of the quantizer for picture "C" is larger than that for "B" or "A". The 8 level quantizer designed in chapter two for each picture was used.

Although the improvement of the quantization noise was not significant, some subjective improvement, in particular in high detail areas, was noticeable, but unfortunately it was not as expected. Although the horizontal and vertical edges may be reproduced with small degradation by using nonadaptive predictors, the rendition of these edges has shown to be improved with the adaptive predictors.

Figs.4.4.4 and 4.4.5 show the reconstructed images of system (1) and (3) respectively, which show the superiority of system (3).



$$\hat{X} = \begin{cases} A, & \text{IF } |B-C| \leq |A-C| \\ B, & \text{otherwise} \end{cases}$$

Fig.4.2.1. Illustration of the adaptive predictor used by Graham [85] and Conner [87].

x	xP15	xP16	xP17	xP18		Xp15	xP11	oP1	oP5
x	xP11	xP12	xP13	xP14		xP16	xP12	oP2	oP6
o	oP1	oP2	oP3	oP4		xP17	xP13	oP3	oP7
o	oP5	oP6	oP7	oP8		xP18	xP14	oP4	oP8

(a) (b)

o	o	o	oP11	oP12	oP13	oP14
o	oP15	oP16	oP17	oP18	xP1	xP2
oP19	oP20	oP21	oP22	xP3	xP4	xP5
oP23	oP24	oP25	xP6	xP7	xP8	x
oP26	oP27	xP9	xP10	x	x	x

(c)

oP11	oP12	oP13	oP14	oP15	o	o
xP1	xP2	oP16	oP17	oP18	oP19	o
xP3	xP4	xP5	oP20	oP21	oP22	oP23
x	xP6	xP7	xP8	oP24	oP25	oP26
x	x	xP9	xP9	xP10	oP27	o

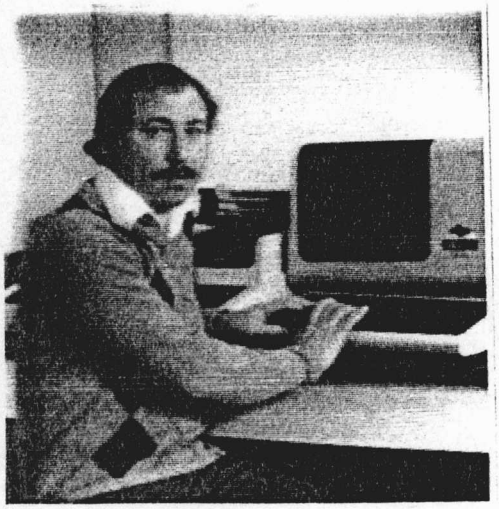
(d)

Fig.4.3.1 Four possible edges.

xP8	xP9	xP10	xP11
xP4	xP5	xP6	xP7
xP3	xP2	xP1	

Fig.4.3.2 The pixels used for detecting the edges, P1 is the pixel to be coded, P2-P11 are the previous pixels.

"A"



"B"

"C"

Fig.4.4.1 The original pictures.

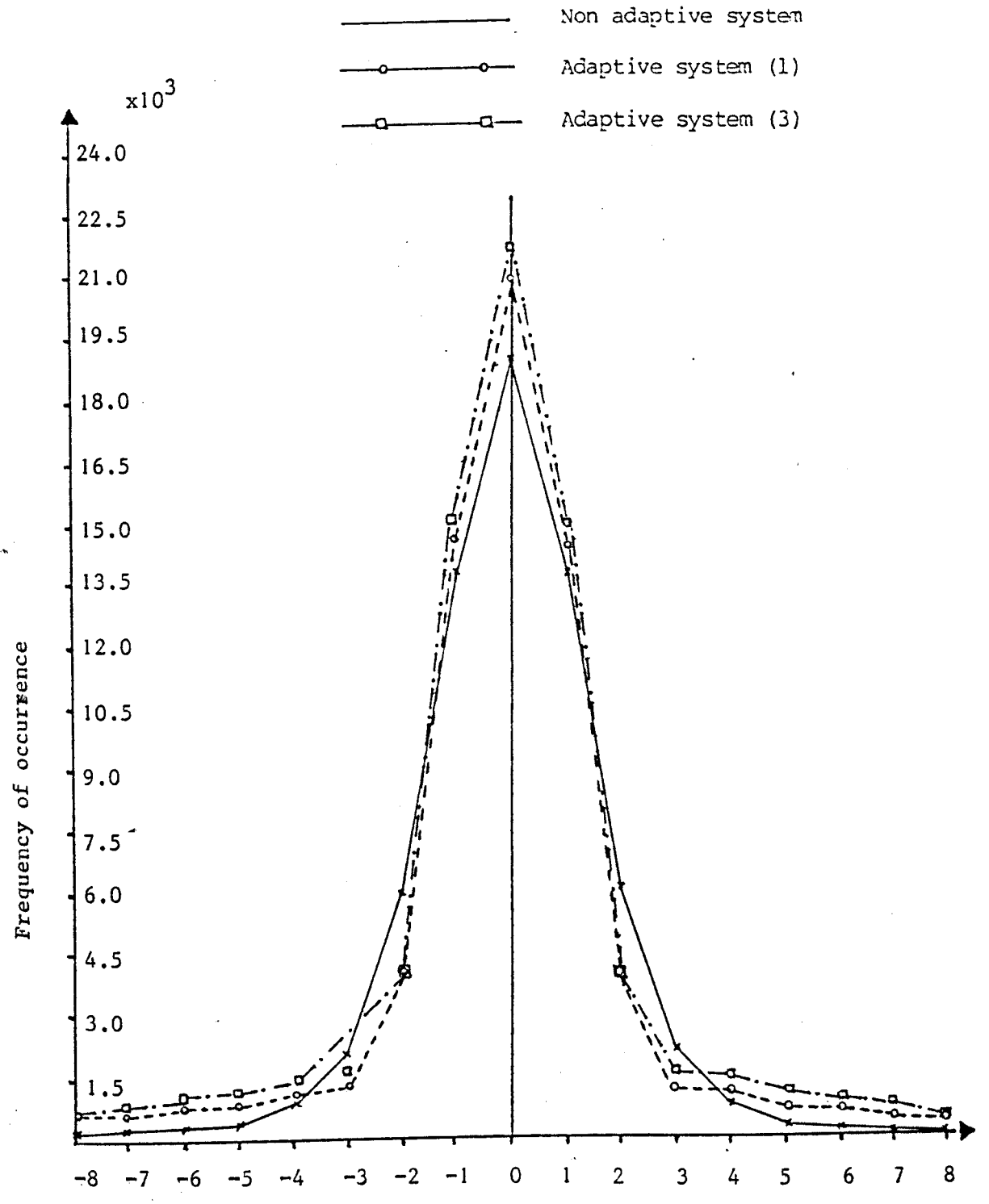


Fig.4.4.2 Histogram of the prediction error.

Table 4.4.1 The In the signal to noise ratio (SNR) of the adaptive predictors over the optimum non-adaptive predictors,

	Pictures		
	"A"	"B"	"C"
System (1)	Ø.5	Ø.9	1.1
System (3)	Ø.7	1.Ø	1.2

Ø	Ø	Ø	Ø	Ø
Ø	Ø	Ø	Ø	Ø
Ø	Ø	xP1	xP2	xP3
Ø	Ø	xP4	xP5	xP6
Ø	Ø	xP7	xP8	xP9

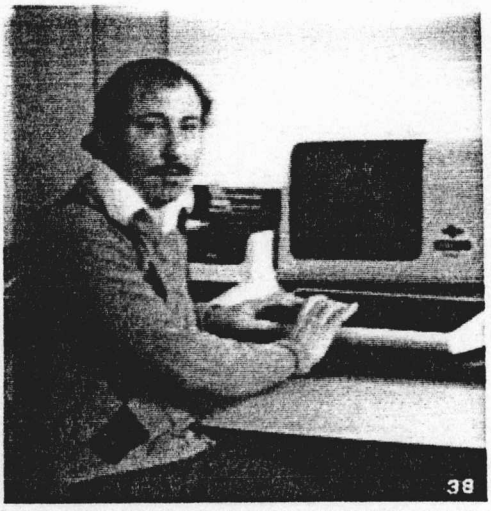
Fig.4.4.3 Waveform with rectangular edge, where "Ø" and "x" represent black and white intensities respectively.

(a) (b)

(c)

Fig.4.4.4 a-c. The recontructed images of the adaptive predictor (1).

(a)



(b)

(c)

Fig.4.4.5 a-c. The recontructed images of the adaptive predictor (3).

CHAPTER FIVE

5 The Hierarchic Hadamard Transform

5.1 Summary

A coding method based on a hierarchic quadtree structure in a transform domain is demonstrated. The n-bit picture is partitioned into L two level pictures called bit plane, so that the original picture is completely described by the set of L bit planes. Each bit plane of NxN elements is Hadamard transformed and the position and the sign of the largest coefficient above a given threshold is transmitted. If there is no coefficient larger than the threshold, then the NxN transformed data is inversed transformed to four arrays of (N/2xN/2) samples and the process repeated. At the receiver, the received coefficient information is assumed to relate to the maximum possible coefficient value and the transform is reconstructed. The information needed to be communicated to the receiver about each array consists of prefixed bit to indicate a further inverse transformation or a presence of a coefficient above the threshold. In the case of the presence of a coefficient above the threshold, an additional code word whose length depends on the array size is necessary to code the

coefficient position.

The algorithm was applied for two cases, (1) the bit planes are obtained with the L-bit pure binary code, (2) the bit planes are obtained with the L-bit Gray code.

5.2 Introduction

A bandwidth reduction of multilevel picture can be achieved by run-length coding the bit state transition in each bit plane. However, simulation tests indicate that the bandwidth reduction obtainable is usually much less than 2:1 [153,154].

Transform coding technique has been used with success in coding multilevel images to achieve high compression factor. However, its applicability to two levels fascimile picture has been proved to be limited [155]. Nevertheless its use in the compression of bit planes obtained from multilevel picture has not been examined. In this chapter, the feasibility of employing the transform coding technique on the bit planes for compressing multilevel images is investigated. The Hadamard transform and its fast algorithm are first reviewed. A heirarchic quadtree Hadamard transform coding is then introduced. Finally, a computer experiment performed on three picture samples is described and simulation results are discussed.

5.2.1 Discrete Linear Transformations and the Hadamard Transform

The one dimensional transform is related to one dimensional image array. For an image array f(x) of size N, the transformation in series form is expressed as [54]

$$F(u)=\sum_{x=0}^{N-1} f(x).h(u,x)$$

where F(u) is the transform of f(x), h(u,x) is the transform kernel, and u assumes values in the range Ø,1,.....,N-1.

The two dimensional transform of NxN image array f(x,y) results in an NxN transform array and is given by the equation

$$F(u,v)=\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y).h(x,y,u,v)$$

for u,v=Ø,1,.....,N-1

The kernel of the Hadamard transform is separable and symmetric i.e.

separable $h(x,y,u,v)=h(x,u).h(y,v)$

symmetric $h(x,y)=h(y,v)$

where h(x,y,u,v) is the two dimensional kernel and h(x,u)

is the one dimensional kernel. A separable two dimensional transformation can be computed in two steps. First, a one dimensional transform is taken along each column and next a second one is performed along each row. The one dimensional Hadamard kernel is defined as:

$$h(x,u)=1/N.(-1)^{\sum_{i=0}^{n-1} b_i(x).b_i(u)}$$

for $N=2^n$. The sumation in the exponent is performed in modulo 2 arithmetic and $b_i(z)$ is the k-th bit in the binary representation of z.

The separable and symmetric properties of the kernel leads to a matrix representation of the Hadamard transform i.e.

$$\underline{F}=\underline{H}\ \underline{f}\ \underline{H}$$

where $\underline{f}$ is an NxN matrix of the image array, $\underline{H}$ is an NxN symmetrical transform matrix generated from the one dimensional Hadamard kernel (h(x,u)), and $\underline{F}$ is an NxN matrix of the transformed image array. The Hadamard matrix of order 2 is given by:

$$\underline{H}(2)=1/\sqrt{2}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$$

where the matrix of order 4 is defined as

$$\underline{H}(4)=1/2 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

A more convenient definition exists for Hadamard matrix in a recurrence form. If $\underline{H}(N)$ represents a matrix of order N, then the matrix of order 2N ($\underline{H}(2N)$) is defined:

$$\underline{H}(2N)=1/\sqrt{2}\begin{bmatrix} \underline{H}(N) & \underline{H}(N) \\ \underline{H}(N) & -\underline{H}(N) \end{bmatrix}$$

The Hadamard matrix is a real symmetric unitry matrix possessing the following properties:

(i) $\underline{H}(N)=\underline{H}^{T}(N)$

(ii) $\underline{H}^{-1}(N)=\underline{H}^{T}(N)$

(iii) $\underline{H}(N).\underline{H}^{T}(N)=\underline{H}(N).\underline{H}^{-1}(N)=\underline{H}(N).\underline{H}(N)=\underline{I}$

where $\underline{H}^{T}(N)$ and $\underline{H}^{-1}(N)$ are the transpose and inverse of $\underline{H}(N)$ respectively, and $\underline{I}$ is the unity matrix.

The foregoing representation is called the "Natural ordered Hadamard matrix". Another representation exists for the Hadamard matrix in "Ordered form" in which the sequence of each row is larger than the preceding row, where "sequency" is defined as the number of sign changes along each row of Hadamard matrix.

5.2.2 The Fast Hadamard Transform

In common with the Fourier transform, a fast algorithm exists for the Hadamard transform [156]. This reduces the number of arithmetic operations for a two dimensional transform from the N^4 down to $2N^2\log_2(N)$, a saving of 99.7 percent for a 64x64 array. The operations needed are addition and substraction whereas the fast Fourier transform needs complex multiplication and addition.

The calculations are performed in stages illustrated by the simple example in Fig.5.2.2.1. The signal graph is used to pictorially represent the sequence of operations. Computation proceeds from left to right for the forward, or right to left for the inverse transform. The numerical value represented by a node on the graph is added (continuous line) or substracted (broken line) from the nodes to which it is connected.

The highly structured nature of the signal graph reflects the recurrence matrix definition of the Hadamard kernel. It also indicates that the computation should occur in a well structured way, which is the case for radix 2 transform, there will be n stages in signal graph for 2^n data elements. A radix 4 algorithm would have n/2 stages in the signal graph, each node corresponding to four arithmetic operations. This is useful for a two

dimensional transform, as the signal graph then reflects the symmetry of the square data array.

The fast algorithm for the natural ordered Hadamard transform may be derived from the recurrence relation. Considering a one dimensional transform of row vector $\underline{x}$

$\underline{X}=\underline{H}(n).\underline{x}$ where $\underline{x}$ has 2^n elements.

The matrices may be represented in partitional form as

$$\begin{bmatrix} \underline{X}a \\ \underline{X}b \end{bmatrix} = \begin{bmatrix} \underline{H}(n-1) & \underline{H}(n-1) \\ \underline{H}(n-1) & \underline{H}(n-1) \end{bmatrix} \begin{bmatrix} \underline{x}a \\ \underline{x}b \end{bmatrix}$$

where $\underline{X}a=\underline{H}(n-1).\underline{x}a+\underline{H}(n-1).\underline{x}b$

$= \underline{Y}a + \underline{Y}b$

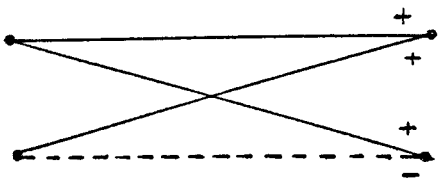
and $\underline{X}b=\underline{H}(n-1).\underline{x}a-\underline{H}(n-1).\underline{x}b$

$= \underline{Y}a - \underline{Y}b$

$\underline{Y}a$ and $\underline{Y}b$ are the Hadamard transforms of the two partitioned sections of $\underline{x}$.

Thus a transform of a sequence of length 2^n may be represented as a linear combination of the transforms of the two subsequences of length 2^{n-1}. The same will be true of each subsequence, the process continuing until eventually the subsequence length is one. This corresponds directly to a radix 2 for Hadamard transform, each reduction in subsequence length corresponding to a change in level on the signal graph for the algorithm.

An algorithm for computing the radix 2 transform can be simply derived from the signal graph. Fig.5.2.2.2 illustrates the terms used in the algorithm. The LEVEL refers to the stage in the signal graph, level "Ø" corresponding to the initial data. The nodes fall naturally into GROUPS, which increase in size as the transform proceds. A BUTTERFLY consists of a pair of result nodes derived from the same source nodes as in the figure below.



A computational "Butterfly"

Natural Hadamard transform; (radix 2, 2^n samples)

```
for level = 1 to n
  for group = 1 to max group
     for butterfly = 1 to max butterfly
        compute 1 pair of nodes
     continue
  continue
continue
end
```

The naturally ordered forward transform has

symmetric stages, and the forward algorithm can therefore be applied with no change.

Of particular interest for image processing, is the two dimensional transform. It may be implemented using radix 2 or radix 4 fast algorithm. A radix 4 transform, extends the concept of the radix 2 transform into two dimensions. Fig.5.2.2.3 shows the signal graph for this algorithm. Each node on the graph now corresponds to four arithmetic operations.

A partial transform [156] of two dimensional (fig.5.2.2.4) provides a hierarchy transforms of size 2^L (L=level), each of which is computed from the results of the previous 4 transforms which make up the same area.

5.2.3 The Hierarchic Hadamard Transform

The central concept is that the fast transform algorithm is ordered such that each stage of the fast algorithm can produce a set of sub-image transforms. Fig.5.2.3.1 shows a representation of the whole set of transforms for a 4x4 image. The hierarchic nature of the transform set can clearly be seen, and the relation to the quadtree structure is also evident.

It is thus possible to process the image in blocks from 1x1 to NxN pixels, and to extract a range of features (e.g. the largest coefficient in absolute value above a threshold) at any stage. A considerable degree of

freedom is available in the way in which this can be achieved. Processing may start from the pixel level and move to global transforms, begin by transforming the whole image and move towards the pixel level, or even start at an intermediate level and move either way. Clark [156], developed the hierarchic Hadamard transform and applied it to the extraction of features from grey level images.

5.3 Experimental Procedure

In constant word length PCM coding of an image, the code words may be conceptually organized into planes called bit planes. It has been found that in most natural images the most significant bit planes seldom change, while the least significant bit planes fluctuate almost randomly.

Each bit plane say of NxN pixels is transformed into Hadamard domain, and the absolute value of the transformed array are examined for a coefficient above the threshold. If there is one coefficient or more above the threshold, then the transformed array is coded using one bit to indicate the presence of a coefficient exceeding the threshold, $2\log_2(N)$ bits for the position of the largest coefficient and another bit for its sign. The value of the coefficient is set at the receiver equal to the maximum value that may result. In the case of no

coefficient above the threshold, a one bit code word is used to notify an inverse transform of the NxN array to 4x(N/2xN/2) arrays and each one is tested again.

To make easy the implementation of the transform, only integer calculations are performed. This is possible if the data in the forward transform are not scaled, while they are scaled by 1/R at each inverse transform stage, where R is the radix. In all systems, radix 4 fast Hadamard transform with the signal flow graph shown in Fig.5.2.2.3 was used. The pixel value af the bit plane is either "Ø" or "1". The value "Ø" is converted to "-1" for the transform, and therefore the maximum value possible of a coefficient is equal to the number of the pixels in the array.

The probability of having a coefficient with a maximum value, decreases with increasing array size. Thus, instead of applying the hierarchic transform discussed in section 5.2.3 to the entire bit plane of 256x256 elements, we divide it into subarrays of 32x32 and for each one the hierarchic transform is used.

The bits required to code a whole array of NxN are $2+2\log_2 N$, which means a compression of $N^2/(2+2\log_2 N)$. For all arrays larger than 2x2, a compression is possible. For the arrays of 2x2 no compression is possible because coding the individual pixels will result in the same bit number as with coding the whole array. Therefore, only the coefficient representing the mean is examined for the 2x2 arrays.

Finally we introduce the term "distortion factor" (K) or "error pixels" to indicate the maximum number of the pixel allowed to be reconstructed incorrectly. For example, an 8x8 array with say four error pixel (distortion factor K=4) means that the array may be reconstructed with four wrong pixels. In fact a high distortion factor corresponds to low threshold value and low distortion factor corresponds to high threshold. We will use the term "threshold" or "distortion factor" whenever it is convenient.

The relation between the threshold (TH) and the distortion factor (K) of an NxN array is defined:

$$TH=N^2-2.K$$

5.4 Simulation Results and Discussion.

The brightness of the pixels of the pictures shown in Fig.5.4.1 is represented by a 7 bit binary code words. The corresponding bit planes of pictures "A" and "B" are shown in Figs.5.4.2 and 5.4.3 respectively, where black corresponds to "Ø" and white corresponds to "1". Although the most significant bit plane (h) is very simple and suitable for the hierarchic quadtree coding, the complexity (the high frequency content) increases as the significance of the bit plane decreases. The least significant bit plane is very "noisy" and is unsuitable for information preserving coding. The compression may be

obtained in coding the three or four most significant bit planes, will be partly lost with the others.

The average bit rate of information preserving coding of picture "A" was 4.8 bit/pel, which means a saving of 2.2 bit/pel. In fact, the bit rate of the four most significant bit planes was 1.5 bit/pel. The bit rate of the first three least significant bits is therefore 3.3 bit/pel which is by Ø.3 bit/pel more than the direct coding of the pixels. The bit rate of information losless coding of picture "B" and "C" are 5.6 and 6.4 bit/pel respectively. The frequency occurance of the coded arrays of picture "A" for all 7 bit planes is listed in table 5.4.1.

A better compression may be achieved by using the threshold coding. The choice of the threshold depends on the bit plane and the array size. Since error produced from the most significant bits has noticeable effect on the reconstructed image quality, a high threshold is desirable to avoid the errors. On the other hand, a high threshold may reduce the compression. However, as the bit planes seldom change, high threshold or even errorless coding will achieve a reasonable compression.

For the least significant bit planes, which fluctuate randomly, a low threshold is required to achieve reasonable compression. Another point of view for getting better compression, is to use a high distortion factor for larger arrays, in particular for the least significant bit planes. This on the other hand may cause

that a number of adjacent pixels be incorrectly recontructed, which has more noticeable effect than randomly distributed pixels on the whole bit plane. Since errors in the least significant bit planes are randomly distributed and their effect on the quality of the reconstructed image is therefore small, a high distortion factor in general will hardly affect the image quality. Fig.5.4.4 shows the output images using the distortion factors listed in tables 5.4.2-5.4.4. No distortion of any kind was noticeable.

The bit rate of picture "A" is 3.7 bit/pel which means a compression ratio of 2. The bit rates of pictures "B" and "C" are 4.6 and 5.4 bit/pel. The threshold coding has improved the bit rate performance by nearly 1 bit/pel for all images, compared with the errorless coding.

If instead of pure binary coding, an 7 bit Gray code is used for representing the brightness of pels, the pattern of the bit planes change considerably. The bit planes obtained with the Gray coded version of pictures "A" and "B" are shown in Figs.5.4.7.and 5.4.8 respectively, where the areas of equal brightness are seen to be larger. The bit rate of error-less coding has been improved by 1 bit/pel compared with the equivalent case using the pure binary code. The distribution of the coded arrays of picture "A" is shown in table 5.4.5 . Also the threshold coding has reduced the bit rate by 1 bit/pel compared with the equivalent case of pure binary case. The output images are shown in Figs.5.4.7.

Thresholds (distortion factors) and the coded arrays are listed in tables 4.4.6-4.4.8. A compression ratio of 3 was achieved for picture "A". Table 5.4.9 summarise the obtained results.

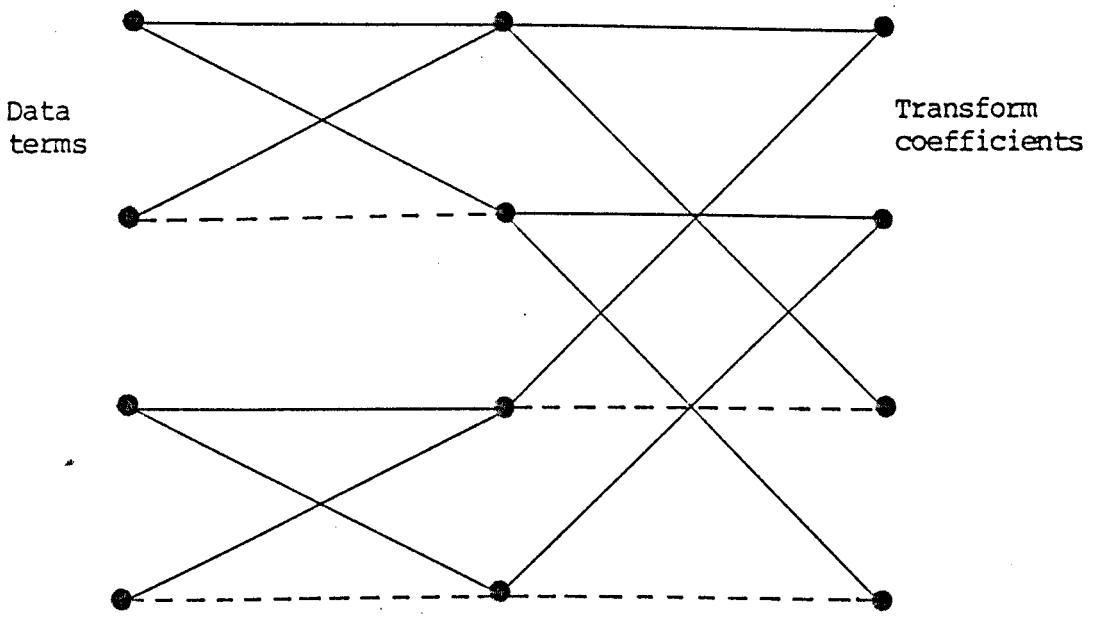


Fig.5.2.2.1 Signal graph for naturally ordered Hadamard transform.

addition ——→

substraction – – →

direction of forward travel

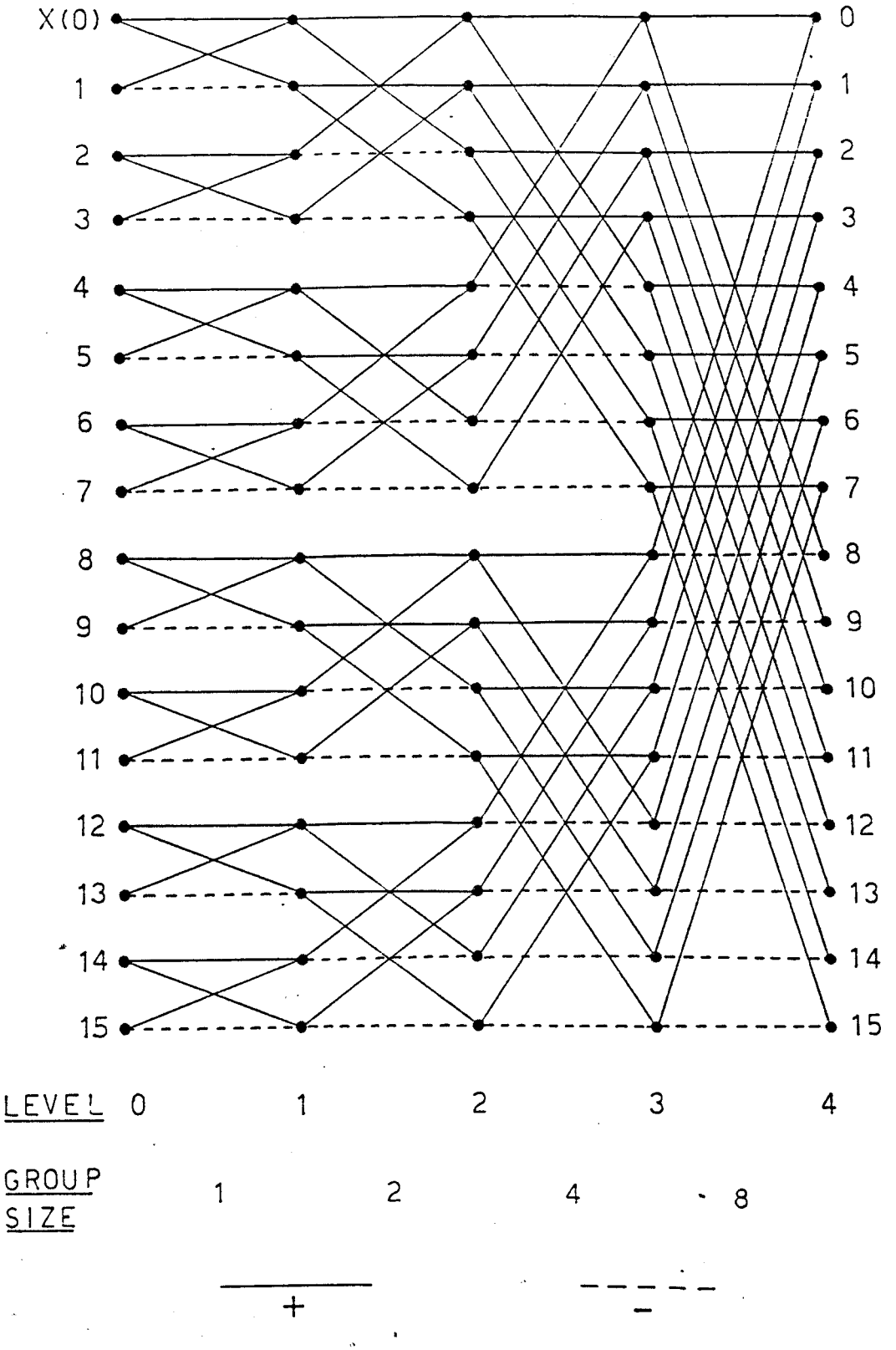


Fig.5.2.2.2 Signal flow graph for a 16 point naturally ordered Hadamard transform.

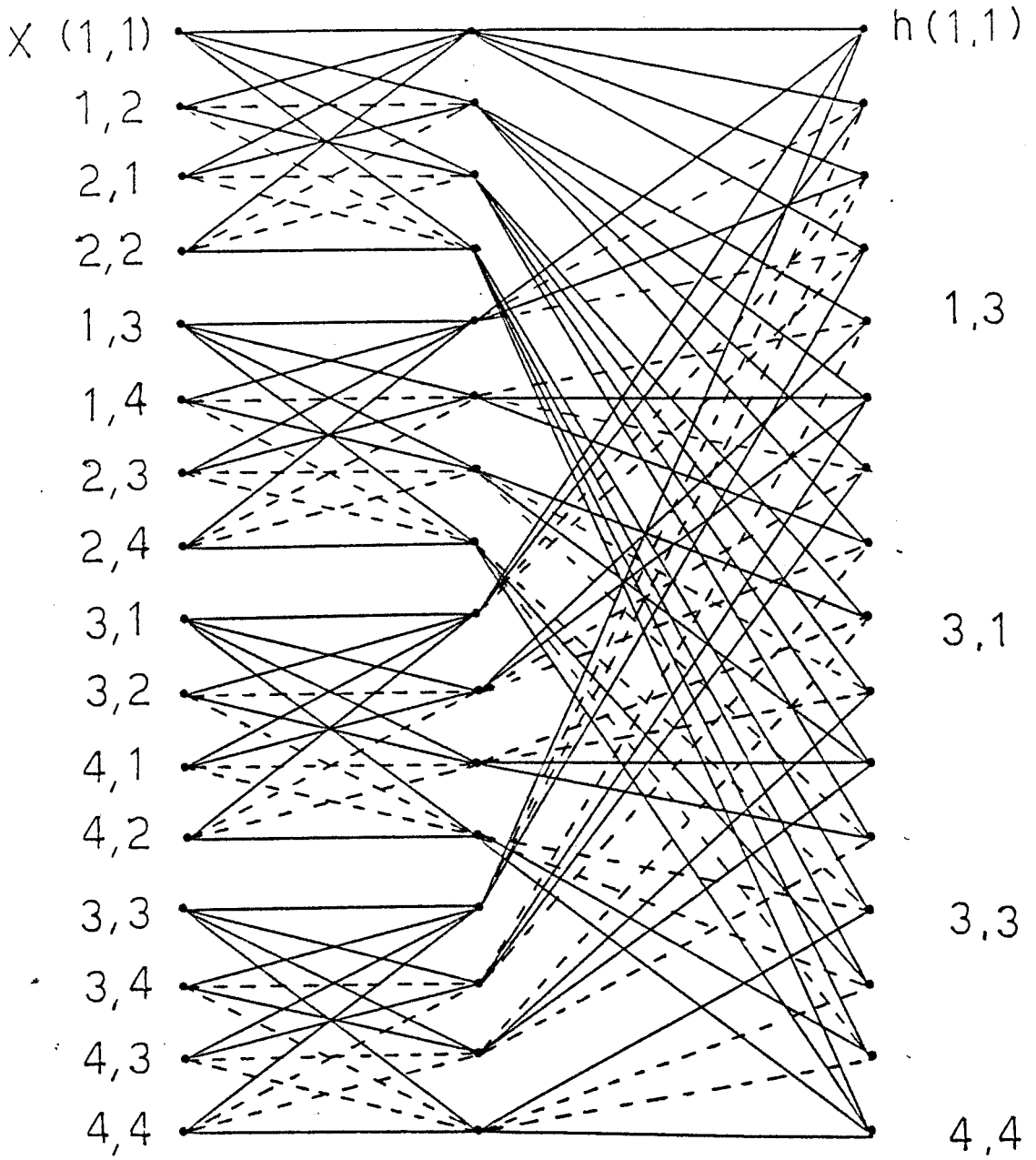


Fig.5.2.2.3 Signal flow graph for a (4x4) point naturally ordered radix 4 Hadamard transform.

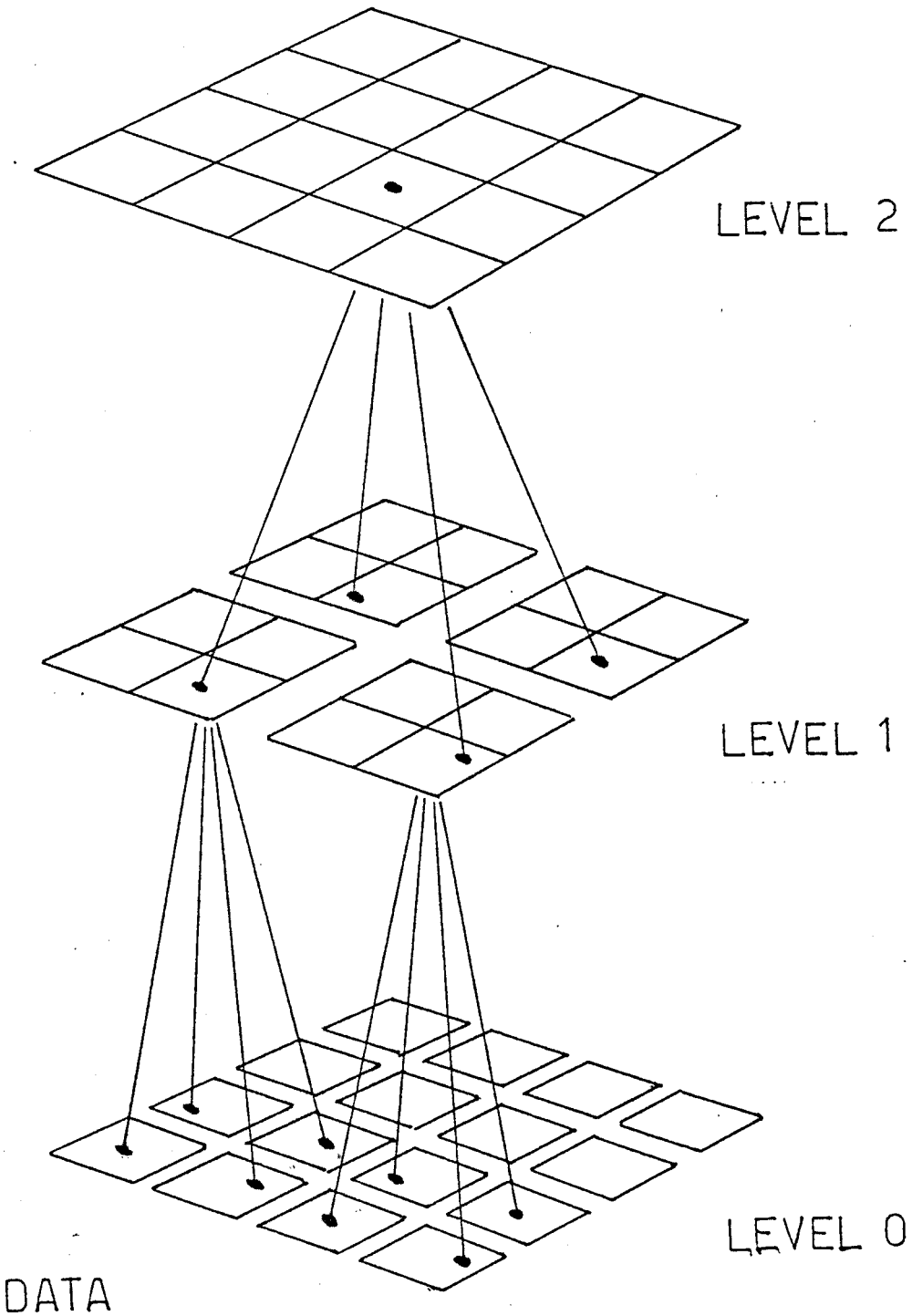
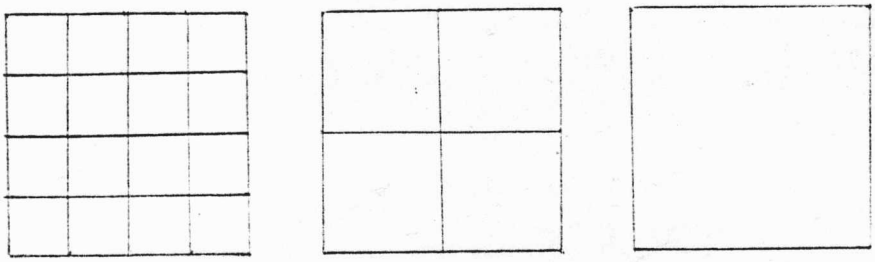


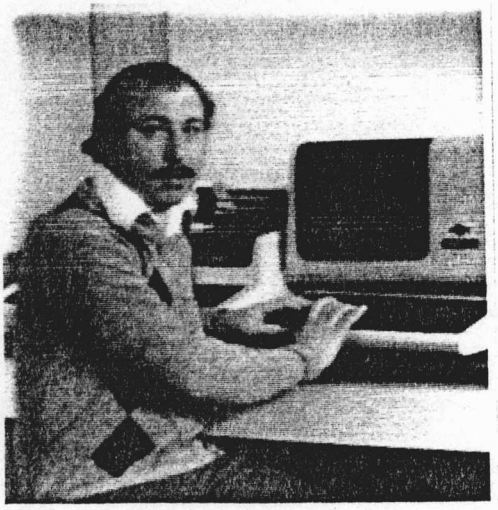
Fig.5.2.3.1 A Heirarchic set of Hadamard transforms



Data level Ø | 2x2 transforms level 1 | 4x4 transforms level 2

Figure 5.2.2.4 Two dimensional partial transform

"A"

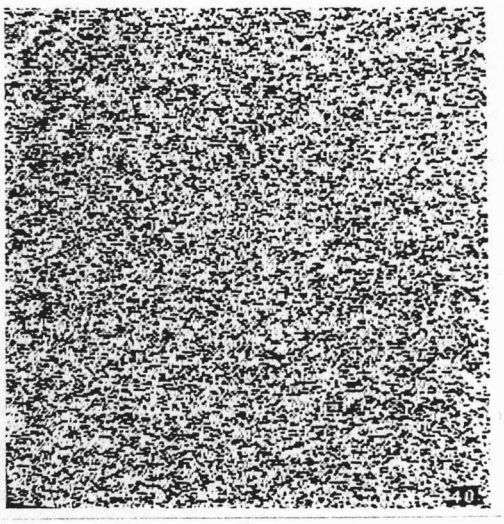


"B"

"C"

Fig. 5.4.1
Original images of
7 bit brightness and
256x256 elements.

(a)



(b)

(c)

(d)

cont.

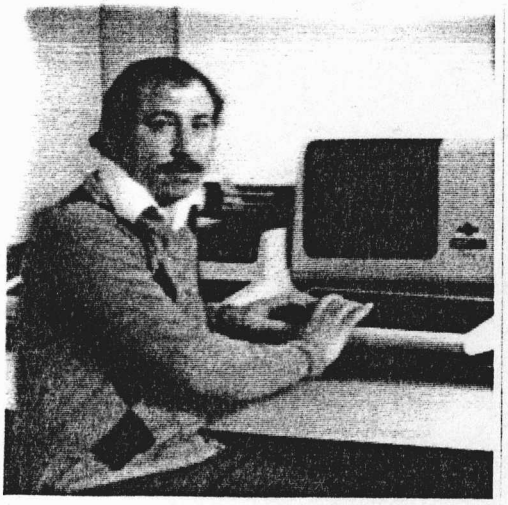
(e)

(f)

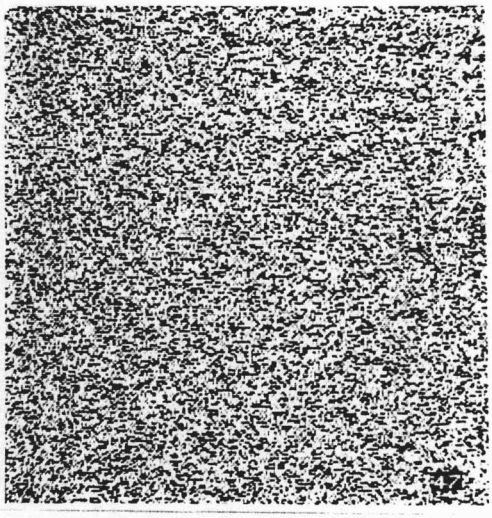
(g)

(h)

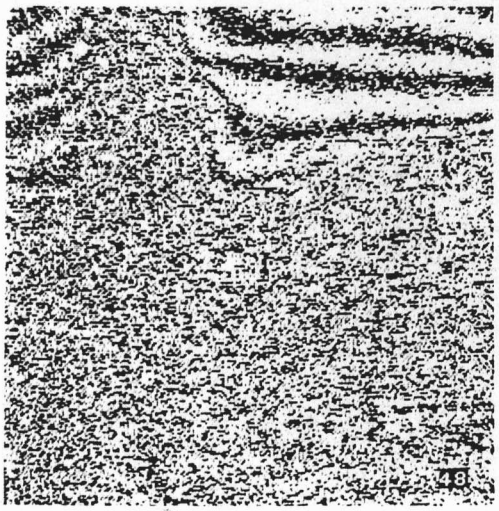
Fig.5.4.2 Bit representation of the image brightness. (pure binary code); (a) original image; (b) 1st bit plane (least significant bit); (c) 2nd. bit plane; (d) 3rd. bit plane; (e) 4th. bit plane; (f) 5th. bit plane; (g) 6th. bit plane; (h) 7th. bit plane (most significant.



(a)



(b)



(c)

(d)

cont.

(e)

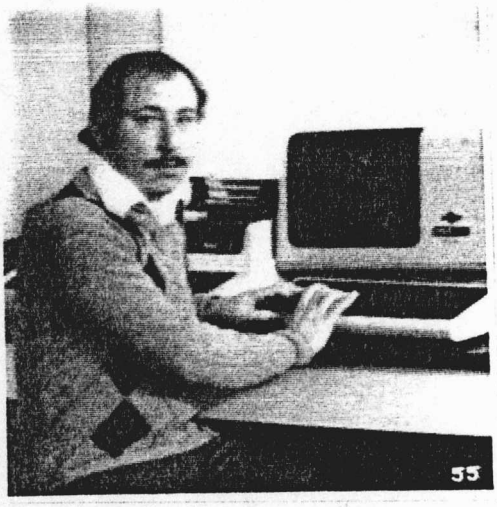
(f)

(g)

(h)

Fig.5.4.3 Bit representation of the image brightness (pure binary code); (a) original image; (b) 1st. bit plane; (c) 2nd. bit plane; (d) 3rd. bit plane; (e) 4th. bit plane; (f) 5th. bit plane; (g) 6th. bit plane; (h) 7th. bit plane (most significant.

(a)



(b)

(c)

Fig.5.4.4. Reconstructed images of the hierarchic Hadamard transform coding using pure binary code with distortion factors (a) as in table 5.4.2; (b) as in table 5.4.3; (c) as in table 5.4.4.

(a)

(b)

(c)

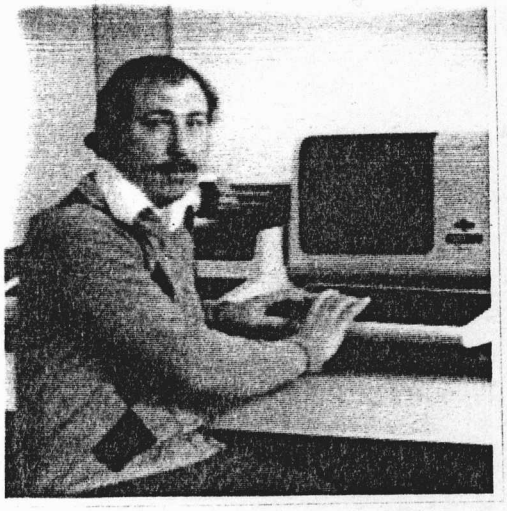
(d)

cont.

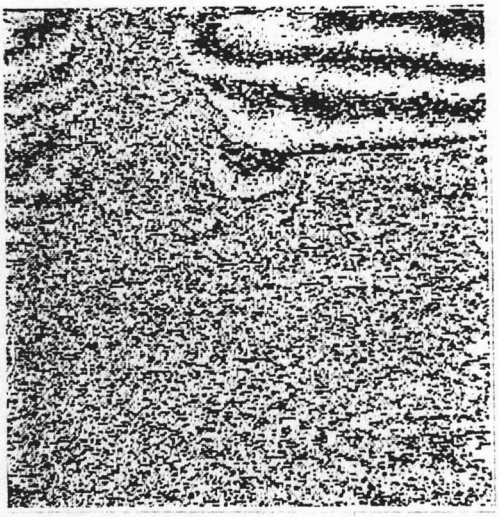
(e) (f)

(g) (h)

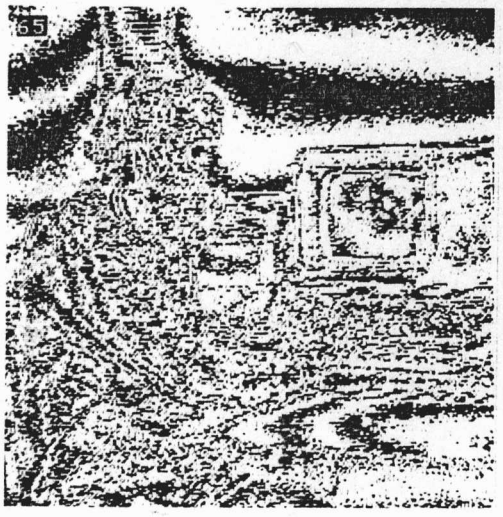
Fig.5.4.5 Bit representation of the image brightness (Gray code); (a) original image; (b) 1st. bit plane; (c) 2nd. bit plane; (d) 3rd. bit plane; (e) 4th. bit plane; (f) 5th. bit plane; (g) 6th. bit plane; (h) 7th. bit plane (most significant.)



(a)



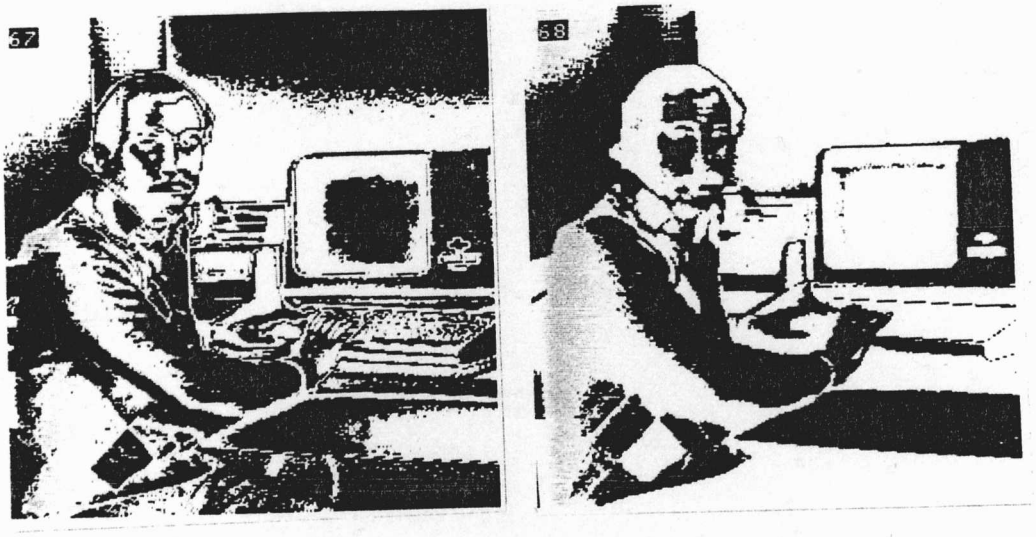
(b)



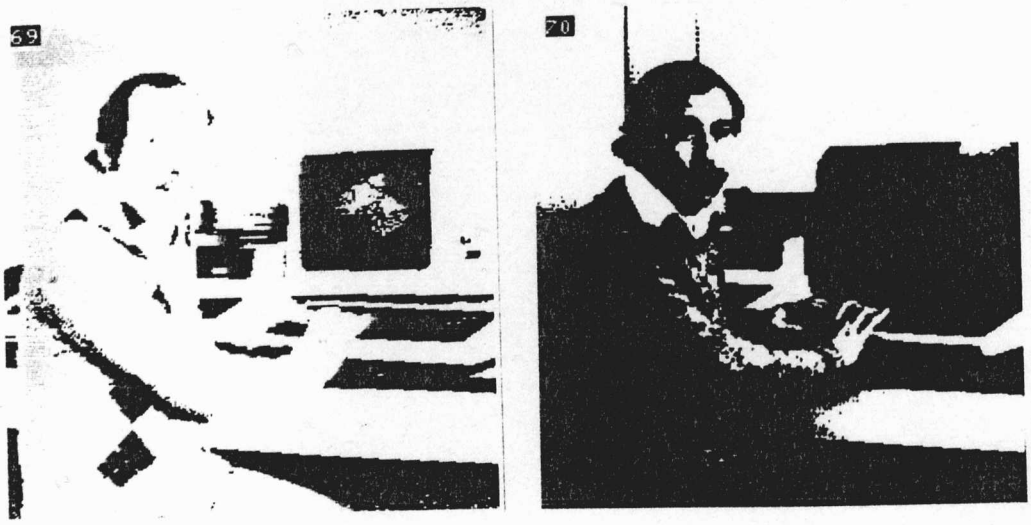
(c)

(d)

cont.



(e) (f)



(g) (h)

Fig.5.4.6 Bit representation of the image brightness. (Gray code); (a) original image; (b) 1st bit plane (least significant bit); (c) 2nd. bit plane; (d) 3rd. bit plane; (e) 4th. bit plane; (f) 5th. bit plane; (g) 6th. bit plane; (h) 7th. bit plane (most significant).

(a) (b)

(c)

Fig.5.4.7. Reconstructed images of the hierarchic Hadamard transform coding using Gray code with distortion factors (a) as in table 5.4.2; (b) as in table 5.4.3; (c) as in table 5.4.4.

Table 5.4.1 The frequency occurrence of coded arrays of picture "A" for losless information coding (Pure Binary Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	Ø	Ø
	16x16	Ø	Ø
(the least)	8x8	Ø	Ø
(significant)	4x4	Ø	3
	2x2	Ø	2318
2nd. bit plane	32x32	Ø	Ø
	16x16	Ø	Ø
	8x8	Ø	Ø
	4x4	Ø	84
	2x2	Ø	4229
3rd. bit plane	32x32	Ø	Ø
	16x16	Ø	2
	8x8	Ø	84
	4x4	Ø	677
	2x2	Ø	4681
4th. bit plane	32x32	Ø	1
	16x16	Ø	23
	8x8	Ø	139
	4x4	Ø	789
	2x2	Ø	4083
5th. bit plane	32x32	Ø	11
	16x16	Ø	47
	8x8	Ø	201
	4x4	Ø	716
	2x2	Ø	2362
6th. bit plane	32x32	Ø	18
	16x16	Ø	48
	8x8	Ø	206
	4x4	Ø	634
	2x2	Ø	1674
7th. bit plane	32x32	Ø	36
	16x16	Ø	51
(the most)	8x8	Ø	126
(significant)	4x4	Ø	262
	2x2	Ø	522

Table 5.4.2 The frequency occurrence of coded arrays of picture "A" for reduced information coding (Pure Binary Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	128	Ø
	16x16	64	Ø
(the least)	8x8	1Ø	Ø
(significant)	4x4	3	1411
	2x2	1	7Ø41
2nd. bit plane	32x32	128	Ø
	16x16	64	22
	8x8	1Ø	35
	4x4	2	528
	2x2	1	8443
3rd. bit plane	32x32	32	4
	16x16	8	16
	8x8	2	89
	4x4	1	727
	2x2	1	7365
4th. bit plane	32x32	Ø	1
	16x16	Ø	23
	8x8	Ø	139
	4x4	Ø	789
	2x2	Ø	4Ø83
5th. bit plane	32x32	Ø	11
	16x16	Ø	47
	8x8	Ø	2Ø1
	4x4	Ø	716
	2x2	Ø	2352
6th. bit plane	32x32	Ø	18
	16x16	Ø	48
	8x8	Ø	2Ø6
	4x4	Ø	634
	2x2	Ø	1674
7th. bit plane	32x32	Ø	36
	16x16	Ø	51
(the most)	8x8	Ø	126
(significant)	4x4	Ø	262
	2x2	Ø	522

Table 5.4.3 The frequency occurrence of coded arrays of picture "B" for reduced information coding (Pure Binary Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	256	0
	16x16	64	2
(the least)	8x8	10	4
(significant)	4x4	3	1385
	2x2	1	7003
2nd. bit plane	32x32	128	0
	16x16	64	3
	8x8	10	8
	4x4	3	1485
	2x2	1	6657
3rd. bit plane	32x32	64	0
	16x16	16	4
	8x8	4	62
	4x4	2	978
	2x2	1	7863
4th. bit plane	32x32	0	0
	16x16	0	11
	8x8	0	102
	4x4	0	741
	2x2	0	4707
5th. bit plane	32x32	0	5
	16x16	0	27
	8x8	0	176
	4x4	0	916
	2x2	0	3369
6th. bit plane	32x32	0	7
	16x16	0	46
	8x8	0	222
	4x4	0	869
	2x2	0	2431
7th. bit plane	32x32	0	25
	16x16	0	58
(the most)	8x8	0	144
(significant)	4x4	0	499
	2x2	0	1124

Table 5.4.4 The frequency occurrence of coded arrays of picture "C" for reduced information coding (Pure Binary Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	256	Ø
	16x16	96	4
(the least)	8x8	16	3
(significant)	4x4	3	1439
	2x2	1	6664
2nd. bit plane	32x32	128	Ø
	16x16	8Ø	Ø
	8x8	1Ø	Ø
	4x4	3	1455
	2x2	1	6813
3rd. bit plane	32x32	64	Ø
	16x16	16	Ø
	8x8	4	1Ø
	4x4	2	683
	2x2	1	9238
4th. bit plane	32x32	Ø	Ø
	16x16	Ø	1
	8x8	Ø	36
	4x4	Ø	517
	2x2	Ø	4779
5th. bit plane	32x32	Ø	1
	16x16	Ø	15
	8x8	Ø	131
	4x4	Ø	832
	2x2	Ø	4521
6th. bit plane	32x32	Ø	2
	16x16	Ø	18
	8x8	Ø	2Ø8
	4x4	Ø	1ØØ2
	2x2	Ø	3953
7th. bit plane	32x32	Ø	12
	16x16	Ø	67
(the most)	8x8	Ø	19Ø
(Significant)	4x4	Ø	682
	2x2	Ø	1886

Table 5.4.5 The frequency occurrence of coded arrays of picture "A" for losless information coding (Pure Gray Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	Ø	Ø
	16x16	Ø	Ø
(the least)	8x8	Ø	Ø
(significant)	4x4	Ø	71
	2x2	Ø	4363
2nd. bit plane	32x32	Ø	Ø
	16x16	Ø	1
	8x8	Ø	59
	4x4	Ø	656
	2x2	Ø	4974
3rd. bit plane	32x32	Ø	4
	16x16	Ø	45
	8x8	Ø	168
	4x4	Ø	687
	2x2	Ø	3189
4th. bit plane	32x32	Ø	8
	16x16	Ø	24
	8x8	Ø	145
	4x4	Ø	821
	2x2	Ø	3461
5th. bit plane	32x32	Ø	19
	16x16	Ø	53
	8x8	Ø	215
	4x4	Ø	553
	2x2	Ø	1459
6th. bit plane	32x32	Ø	19
	16x16	Ø	59
	8x8	Ø	210
	4x4	Ø	563
	2x2	Ø	1252
7th. bit plane	32x32	Ø	36
	16x16	Ø	51
(the most)	8x8	Ø	126
(significant)	4x4	Ø	262
	2x2	Ø	522

Table 5.4.6 The frequency occurrence of coded arrays of picture "A" for reduced information coding (Gray Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	128	0
	16x16	64	29
(the least)	8x8	10	35
(significant)	4x4	3	1487
	2x2	1	5538
2nd. bit plane	32x32	128	6
	16x16	64	80
	8x8	10	63
	4x4	2	515
	2x2	1	4751
3rd. bit plane	32x32	32	14
	16x16	8	41
	8x8	2	118
	4x4	1	690
	2x2	1	4076
4th. bit plane	32x32	0	8
	16x16	0	24
	8x8	0	145
	4x4	0	821
	2x2	0	3461
5th. bit plane	32x32	0	19
	16x16	0	53
	8x8	0	215
	4x4	0	553
	2x2	0	1459
6th. bit plane	32x32	0	19
	16x16	0	59
	8x8	0	210
	4x4	0	563
	2x2	0	1252
7th. bit plane	32x32	0	36
	16x16	0	51
(the most)	8x8	0	126
(significant)	4x4	0	262
	2x2	0	522

Table 5.4.7 The frequency occurrence of coded arrays of picture "B" for reduced information coding (Gray Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	256	Ø
	16x16	64	5
(the least)	8x8	1Ø	7
(significant)	4x4	3	1484
	2x2	1	6535
2nd. bit plane	32x32	128	Ø
	16x16	64	49
	8x8	1Ø	37
	4x4	3	1718
	2x2	1	3Ø63
3rd. bit plane	32x32	32	3
	16x16	8	7
	8x8	2	112
	4x4	1	1ØØ1
	2x2	1	6959
4th. bit plane	32x32	Ø	3
	16x16	Ø	26
	8x8	Ø	15Ø
	4x4	Ø	914
	2x2	Ø	3822
5th. bit plane	32x32	Ø	12
	16x16	Ø	47
	8x8	Ø	189
	4x4	Ø	823
	2x2	Ø	2179
6th. bit plane	32x32	Ø	1Ø
	16x16	Ø	64
	8x8	Ø	229
	4x4	Ø	752
	2x2	Ø	1682
7th. bit plane	32x32	Ø	25
	16x16	Ø	58
(the most)	8x8	Ø	144
(Significant)	4x4	Ø	499
	2x2	Ø	1124

Table 5.4.8 The frequency occurrence of coded arrays of picture "C" for reduced information coding (Gray Code).

	Array size	Distortion factor (K)	Frequency of coded arrays
1st. bit plane	32x32	256	Ø
	16x16	96	14
(the least)	8x8	16	8
(significant)	4x4	3	1398
	2x2	1	6288
2nd. bit plane	32x32	128	Ø
	16x16	8Ø	34
	8x8	1Ø	9
	4x4	3	1397
	2x2	1	5641
3rd. bit plane	32x32	64	2
	16x16	16	6
	8x8	4	56
	4x4	2	1164
	2x2	1	7035
4th. bit plane	32x32	Ø	Ø
	16x16	Ø	5
	8x8	Ø	117
	4x4	Ø	874
	2x2	Ø	4969
5th. bit plane	32x32	Ø	6
	16x16	Ø	41
	8x8	Ø	217
	4x4	Ø	963
	2x2	Ø	2755
6th. bit plane	32x32	Ø	4
	16x16	Ø	44
	8x8	Ø	273
	4x4	Ø	692
	2x2	Ø	2504
7th. bit plane	32x32	Ø	12
	16x16	Ø	67
(the most)	8x8	Ø	190
(significant)	4x4	Ø	682
	2x2	Ø	1886

Table 5.4.9 Bit rate of the different systems

Pictures	Code	Information Coding	Bit/pixel
A	Pure binary	error-less	4.8
B	Pure binary	error-less	5.6
C	Pure binary	error-less	6.4
A	Pure binary	reduced (as in table) (5.4.2)	3.7
B	Pure binary	reduced (as in table) (5.4.3)	4.3
C	Pure binary	reduced (as in table) (5.4.4)	5.1
A	Gray code	error-less	3.8
B	Gray code	error-less	4.7
C	Gray code	error-less	5.4
A	Gray code	reduced (as in table) (5.4.6)	2.65
B	Gray code	reduced (as in table) (5.4.7)	3.5
C	Gray code	reduced (as in table) (5.4.8)	4.1

CHAPTER SIX

6. Learning Automata and Data Compression

6.1 Introduction

In 1961 Tsetlin [157] presented a paper on the behavior of fixed structure automata operating in a random environment. Since that time, there have been many developments in the general area of the learning automata and the subject has been studied in great detail. A survey article which appeared in 1974 [158], covered most of the basic rasults in the field, re-examined some of the theoretical questions and suggested potential areas where the results could be applied.

In this paper, a novel approach to image data compression is proposed which uses a stochastic learning automata (SLA) to predict the conditional probability distribution of the adjacent pixels. These conditional probabilities are used to code the grey level values using a Huffman coder. The proposed system achieves a good compression without any degradation in the compressed image.

In Section Two of this chapter, the basic

concepts of stochastic learning automata and the currently used definitions are briefly outlined. The third section describes the strategy of the proposed learning automata compression system. In the final section, results of computer simulation are presented and conclusions are drawn.

6.2 The basic concepts.

Varsharskii et al. [159] extended the early work of Tseltin [157] to the case of stochastic automaton with variable structure and proposed both linear and non-linear schemes to update the structure of the automaton. A special article on the learning automata [160] contains recent work in the field and references [158,161] contain extensive bibliographies of contributions to both deterministic and stochastic learning automata models.

6.2.1 Automaton

The stochastic learning automaton (SLA) is defined as automaton that operates in a random environment and updates its action probabilities in accordance with the inputs received from the environment, so as to improve its performance in some specific sense. It may be

described by the quintuple $\{X,\underline{\Phi},\alpha,Fn,G\}$ where X is the input set, $\underline{\Phi}\{\phi 1,\phi 2,.....,\underline{\Phi}s\}$ is the interval state set, and $\alpha=\{\alpha_1,\alpha_2,.....,\alpha_r\}$ with $r\leq s$ is the output or action set, Fn and G are respectively the state transition and the output functions and n is the discrete time instant. In general, Fn and G are stochastic functions. If p(n) is an s-vector of state probabilities at time n, and U is an updating scheme which prescribes rules for changing p(n) at each stage or time instant n, then Fn can be replaced by p(n) and U so that an SLA may be described by the sixtuple $\{X,\underline{\Phi},\alpha,p(n),U,G\}$. In fixed structure automata Fn is described by stochastic transition matrices corresponding to each input $x\in X$. If these matrices contain only the elements Ø or 1, the automaton is a "fixed structure deterministic automaton". If the elements lie in the interval [Ø,1], the automaton is a "fixed structure stochastic automaton". In variable structure learning automata, the transition matrices corresponding to the various inputs are themselves updated as the automaton operates in its environment. In this case, the rule by which this updating is to be performed depending on the response of the environment has to be specified.

In a fixed structure automaton, the output mapping G is generally assumed to be deterministic. G partitions the state set $\underline{\Phi}$ into r subsets m_i (i=1,2,...,r), such that the elements of each subset m_k map into the same action α_k . When G is stochastic, a

unique action need not correspond to a given state ϕ.

Fig.6.2.1.1 illustrates a closed loop structure consisting of an SLA and a random environment. For each automaton action α_i at stage n, the environment responds with a random quantity $x(n)|\alpha_i$ which becomes the input to the automaton for the following stage. The environment is said to be stationary if the discrete time random processes $\{x(n)|\alpha_i$,n=1,2,.....} (i=1,2,....,r) are all stationary.

The automaton-environment combination is called i) a P-model if the environment's response (or automaton's input) is either $\emptyset$ or 1, ii) a Q-model if it takes a finite number of values in [$\emptyset$,1] and iii) an S-model if it lies in [$\emptyset$,1]. Schemes developed for P-models can in general be extended to Q and S models. Many probabilistic search procedures that have been reported [162] are similar to S-model schemes.

6.2.2 Environment

A random environment is defined by a finite set of inputs $\alpha=\{\alpha_1,\alpha_2,.....,\alpha_r\}$, an output set X={$\emptyset$,1} and a set $c=\{c_1,c_2,....,c_r\}$ of penalty probabilities. The ouput $x(n)=\emptyset$ at stage n is called a favorable response (success) and x(n)=1 an unfavorable response (failure). c_i is the probability of failure (penalty) when the input is α_i

$$c_i = Pr[x(n)=1 | \alpha(n)=\alpha_i] \qquad \alpha_i \in \alpha$$

In the simplest case the penalty probabilities are constant but unknown and the environment is said to be stationary.

6.2.3 Performance measure

To judge the effectiveness of the learning automaton, various performance measures can be set up. These in turn depend on the prior information available and the ultimate objective for which the automaton is designed. An obvious choice and one which has been used almost exclusively in the literature is the average expected penalty that the automaton receives from the environment. An alternative choice may be the probability with which the various actions are chosen in the limit. The entropy corresponding to the action probabilities has also been suggested [163] as a possible performance measure.

The average penalty is defined as:

$$M(n) = E[x(n) | P(n)] = \sum_{i=1}^{r} p_i(n) c_i$$
$$= \langle P(n), c \rangle$$

where $\langle , \rangle$ denotes the inner product. If a simplex Sp is

defined as:

$$Sp=\{P \mid \emptyset \leq p_i \leq 1, \sum_{i=1}^{r} p_i = 1\}$$

the action probabilities $P \in Sp$. If $p_i = 1/r$, the expected penalty may be expressed as

$$Mo = 1/r\left(\sum_{i=1}^{r} c_i\right)$$

if

$$c_\ell = \min_i \{c_i\}$$

the definitions of expediency, optimality and ε-optimality can be expressed in terms of c_ℓ and Mo.

if $\lim_{n \to \infty} E[M(n)] < Mo$

the learning automaton is said to be "expedient".

if $\lim_{n \to \infty} E[M(n)] = c_\ell$

it is said to be "optimal". The learning automaton is said to be "ε-optimal" if the parameters of the learning algorithm can be chosen that:

$$\lim_{n \to \infty} E[M(n)] \leq c_\ell + \varepsilon$$

for any $\varepsilon > \emptyset$.

A learning automaton is said to be "absolutely

expedient" if:

$$E[M(n+1)|P(n)]<M(n)$$

for all $P \xi Sp$ and $c_i \xi (\emptyset,1)$ $(i=1,2,...,r)$ [164].

Instead of the average penalty $M(n)$, as mentioned earlier, the entropy of the action probabilities can also be chosen as a performance measure. If $H(n)$ is defined as:

$$H(n)=-\sum_{i=1}^{r} p_i(n) \log p_i(n)$$

and $$p_i(\emptyset)=1/r \quad ;\Rightarrow \quad H(\emptyset)=\log r$$

The automaton, in this case, is said to learn if the entropy decreases. The entropy is obviously a minimum if the automaton chooses any action with probability one.

6.2.4 The Reinforcement Schemes

The reinforcement schemes are classified in linear and non-linear. The scheme is said to be linear if the updating procedure is a linear function of the probabilities $p_i(n)$ $(i=1,2,....,r)$. Otherwise is defined as non-linear. Here we give only the linear scheme presented by Narendra et al. [165], which can be regarded as the prototypes of all learning models.

$$p_i(n+1)=p_i(n)+a[1-p_i(n)]$$

$$p_{j\neq i}(n+1)=(1-a)p_j(n) \qquad \alpha(n)=\alpha_i \quad ,x(n)=\emptyset$$

$$p_i(n+1)=(1-b)p_i(n)$$

$$p_{j\neq i}(n+1)=(b/r-1)+(1-b)p_j(n) \qquad \alpha(n)=\alpha_i \quad ,x(n)=1$$

where $\emptyset<a<1$ and $\emptyset<b<1$ are constants called the reward and penalty parameters respectively. $x(n)=\emptyset$ describes a success where p_i is increased and p_j is decreased. $x(n)=1$ describes a failure where p_i is decreased and p_j is increased. If b=a, the scheme is called a linear reward -penalty (L_{R-P}) scheme, if $b=\emptyset$ it is a linear inaction (L_{R-I}).

The terms success and failure denote the response of the environment while reward, penalty and inaction refer to the corrections made to the automaton. For example, for the L_{R-P} model, a success from the environment results in a reward to the automaton and a failure in a penalty.

In the L_{R-P} scheme, when the automaton tries an action α_i and it results in a success, the probability $p_i(n)$ is increased and all $p_j(n)$ $(j\neq i)$ are decreased linearly. Similarly, if a failure results from action α_i, $p_i(n)$ is decreased while all $p_j(n)$ $(j\neq i)$ are increased. In an L_{R-I} scheme, while the probabilities are updated exactly as before for a success, all probabilities are left unchanged in the event of a failure. For an $L_{R-\varepsilon P}$ scheme the decrease in $p_i(n)$ due to

a failure is small compared to the increase in $p_i(n)$ due to a success.

6.3 Compression strategy

The objective is to compress a digital image of NxN pixels, each pixel quantized to L grey levels. The grey level value of the n-th. pixel, called pixel value, is denoted by g(n), such that g(n)=Ø,1,...L-1. The probability P(i)=P(g(n):g(n)=i), is called the pixel probability.

The compression strategy is based on the classical technique of variable length coding, which assigns code word lengths on the basis of fixed probability. Thus frequent pixel values are given shorter code words than less common pixel values. The variable length code is said to be optimum if the average code word length is equal to the entropy of the image. Huffman [90], suggested an algorithm for construction of an optimum code, which is optimal in the sense that no other prefixed code will achieve a lower average code word length. As the code is based on the knowledge of the pixel probabilities, the performance of the code is dependent on the accuracy with which the probabilities have been estimated, and on the adaption of the pixel probabilities with time or space.

An illustration of the strategy of the proposed

compression algorithm is shown in Fig.6.3.1. The image data is fed to stochastic learning automaton which provides an estimated pixel value from the probability distribution $(p_i : i=0,1,..L-1)$. The estimated value is fed to a stochastic environment which responds by an error function e(n) which is the difference between the actual and estimated value. The difference signal is fed to the automaton for updating the probability distribution.

The data compression system Fig.6.3.2 consists of a transmitter and a receiver. At the n-th. stage, both learning automata in the transmitter and receiver contain the state probability vectors $p_i(n-1) \mid g(n-1)$; $i=\{0,1,....,L-1)$; $g(n-1)=(0,1,....,L-1)$. The coder generates the appropriate binary codeword C(n) corresponding to the probability of the value g(n). The decoder at the receiver will decode the received codeword to g(n), according to the Huffman algorithm. Now both automatons provide an estimate $\hat{g}(n)$ of the value g(n) and calculate the difference $e(n)=g(n)-\hat{g}(n)$ for updating the probabilities according to reinforcement scheme U. $\hat{g}(n)$ is determined by a stochastic function G which employs a pseudo-random generator. Generators in transmitter and receiver are synchronised. Fig.6.3.3 shows the configuration of the automaton-environment used in compression systems.

6.4 Reinforcement scheme of the compression system.

Both automatons use identical updating schemes U for changing p (n-1):i=Ø,1,.....,L-1). Although various schemes have been studied, the most satisfactory performance was obtained by using a Q-model L_{R-P} scheme. This is described by:

$$P_i(n+1)=P_i(n)+B/C[C-ER]P_i(n)$$

$$P_{j\neq i}(n+1)=P_j(n)-B/C[C-ER][1-P_j(n)]$$

where $ER=|g(n)-\hat{g}(n)|$ and C and B are constants. Similar reinforcement scheme has been widely stated in the literature [158] and has shown to converge asymptotically and therefore is said to be "optimal" or "ε-optimal". Shapiro and Narendra [166] showed that the L_{R-I} scheme is optimaly convergent and Viswanathan et al. [167] showed that L_{R-P} is ε-optimal.

6.5 Results and Conclusion

The scheme presented in the previous section was tested using a 256x256 image with 16 grey levels. The first order entropy and the conditional entropy were computed and found to be 3.3 bit/pixel and 1.15 bit/pixel respectively. Initially the probabilities of the grey levels were set equally to 1/16 and the constant "C" to

3. "B" is calculated so that the initial probability reaches the maximum probability after 200-250 successive maximum reward iterations. "B" was found to be 0.005. The minimum probability was set equal to 0.003 and the maximum=1.0-15x0.003=0.955. With these parameters, the largest probability and its position of each probability vector was determined each 200 successive iteractions. The test showed that the position of the largest probability of each vector does not change. Fig.6.4.1 shows a plot of the largest probability versus trial number of one probability vector. Huffman codes have the disadvantages that the source statistics must be known a priori and that only stationary sources can be used. The probability distribution of the learning automaton provides a source of information which allows the Huffman algorithm to be used in a readly adaptive manner to overcome both of these difficulties.

Using the learning automaton probability distribution, the average Huffman wordlength was found to be 1.69 bit/pixel. The compression is without degradation of the image.

The system has been re-examined using unequal initial probabilities. In this case, it was found that the average bit rate has been improved by 0.2 bit per pixel.

The disadvantage of this system is that the number of grey levels must be low. For a large number of grey levels, the speed of convergence of the probability

distribution will be slowed down considerably which results in a poor compression ratio.

Conclusions.

It has been shown that a learning automaton may be used successfully in an image data compression system. The system uses a Q-model $L_{Q-\rho}$ updating scheme. Computer simulations have been used to demonstrate the nature of convergence and to compute the compression ratio.

Various linear and non-linear reinforcement schemes have been simulated which result in very poor compression ratios due to either the automaton failing to converge, or the convergence process was too slow for efficient compression. It should be noted that the compression ratio improves significantly with the increase of speed of convergence. The problem of speeding up the convergence process was closely examined by using the known S and Q-model schemes. An updating function was proposed using this principle to achieve the best compression ratio.

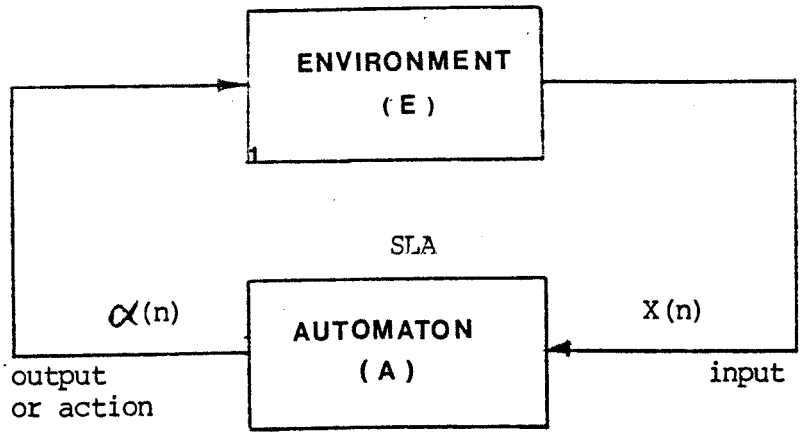


Fig.6.2.1.1 Automaton-environment feedback configuration

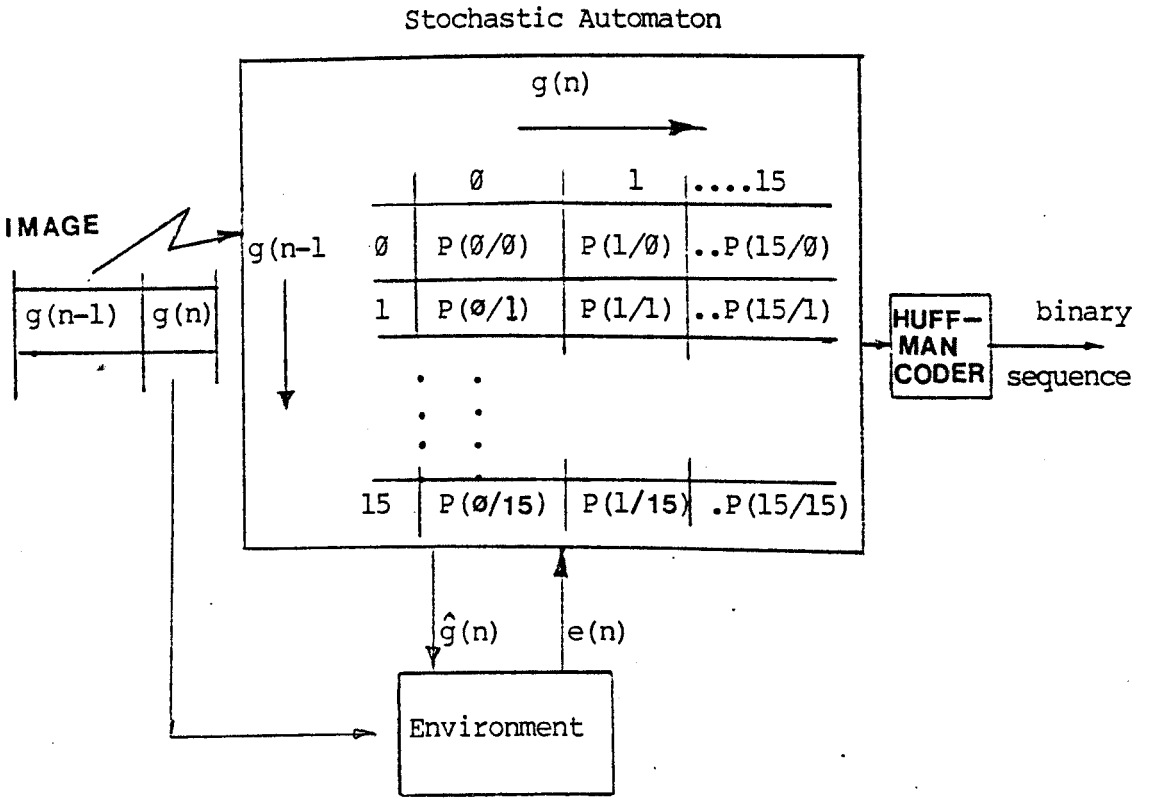


Fig.6.3.1 Illustration of compression algorithm.

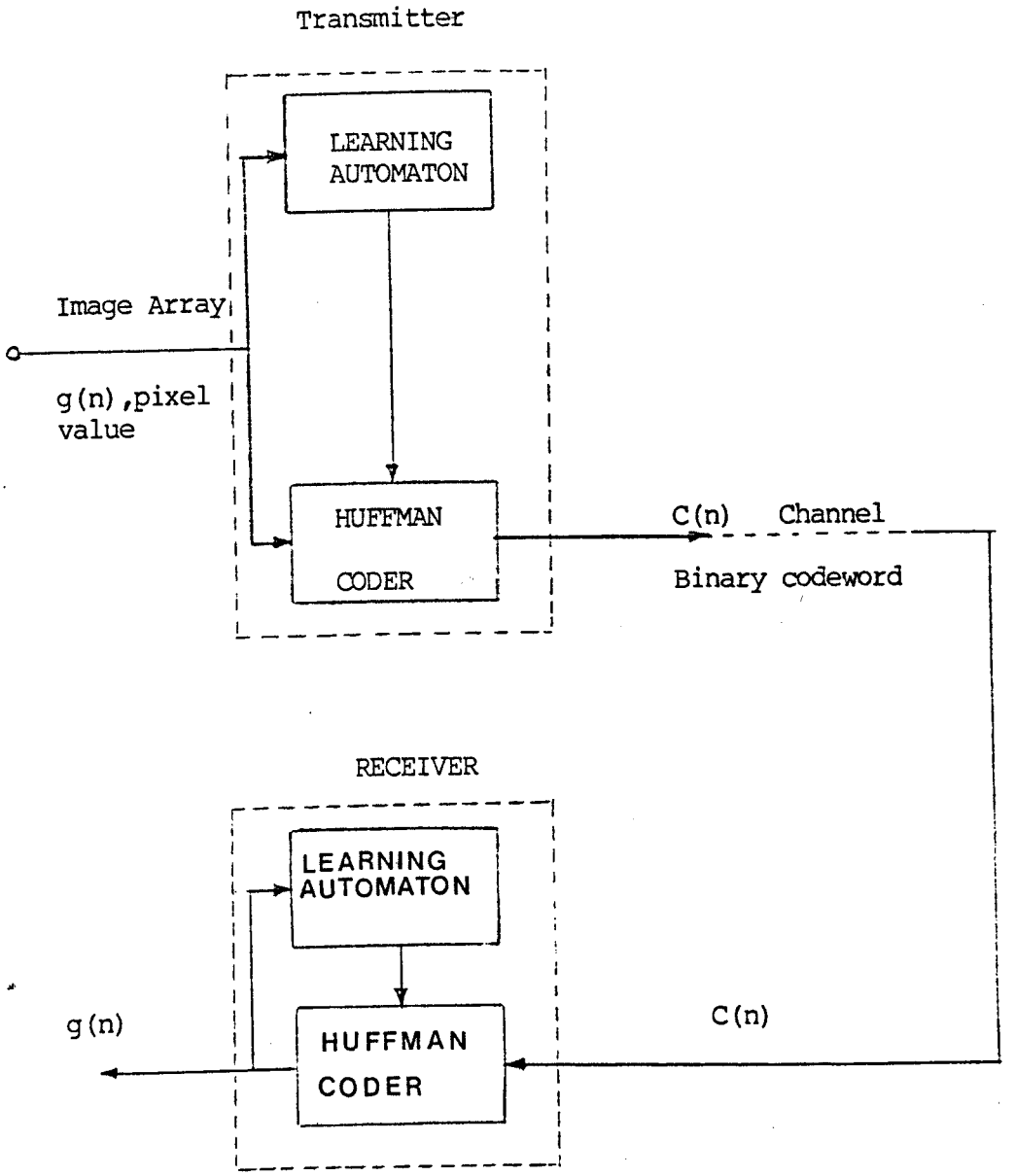


Fig.6.3.2 Data Compression System

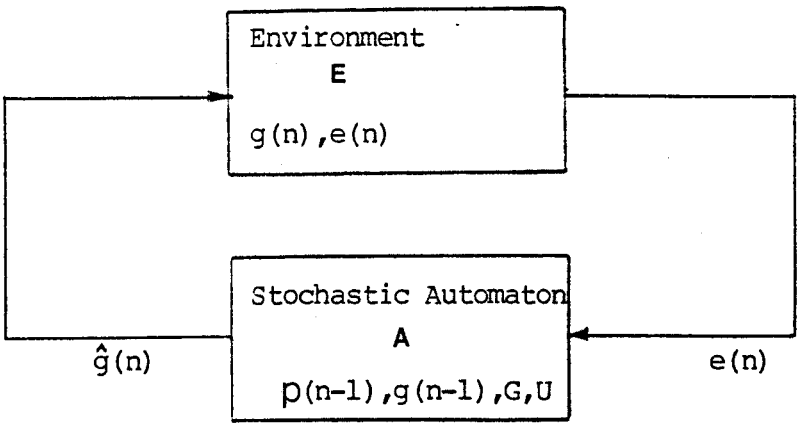


Fig.6.3.3 Illustration of the Automaton-environment combination of the compression system.

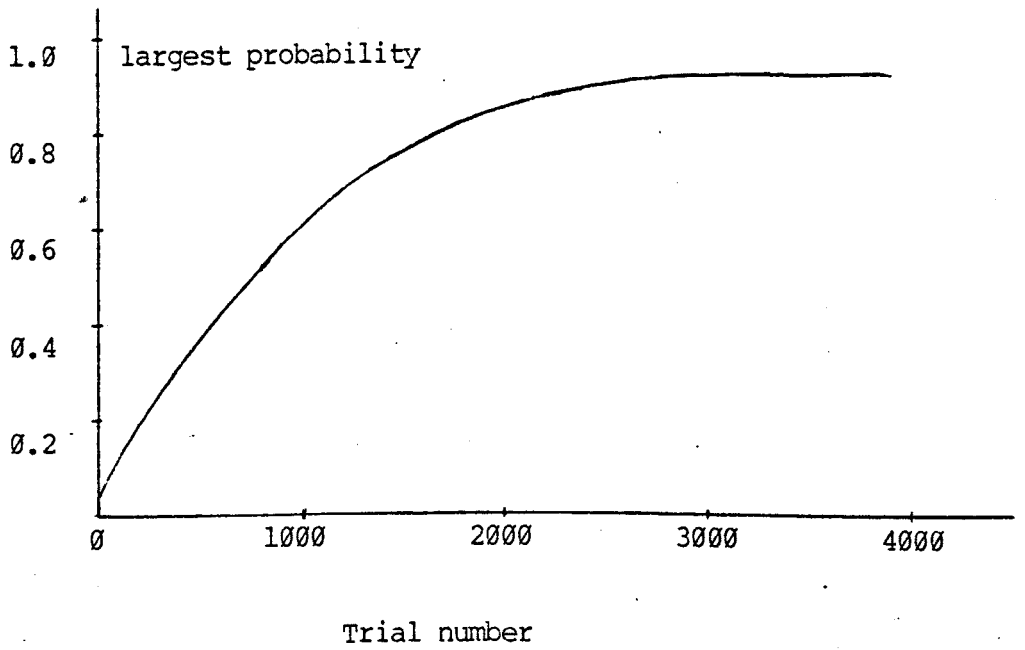


Fig.6.5.1 The largest probability

APPENDIX I

Design of linear predictive image coders

It is conceivable to design the system to minimize a measure of the overall error between the input and the output of the coding system. However, the analysis of such a system is inhibited by the non-linear characteristics of the quantizer. Therefore, the design procedure is to design the predictor ignoring the presence of the quantizer. Then, the quantizer is designed to match the amplitude distribution of the difference signal. On the other hand, placing the quantizer inside the feedback loop will alter the amplitude distribution of the difference signal and the system is no longer optimum. But, when the number of the quantization levels is large (≥ 8) the presence of the quantizer inside the loop has very little effect on the amplitude distribution of the difference signal and the system is nearly optimum. Since the predictor is designed by ignoring the quantizer, the input to the predictor is equal to the original input sequence.

Let {So} be a set of correlated signals with zero mean $E\{So\}=\emptyset$ and the variance σ^2. An n-th. order predictor estimates the next value So by $\hat{So}$ from a linear combination on the n-previously scanned sample values S1,S2,.....,Sn

$$\hat{So}=A1S1+A2S2+.....+AnSn= \quad AiSi \qquad (I-1)$$

If the mean of the signal is not equal to zero, i.e. $E\{So\}\neq\emptyset$, then a better estimate of So is possible if a constant term is also used [168]. Thus:

$$\hat{So}=Ao+A1S1+A2S2+.....+AnSn=Ao+\sum_{i=1}^{n} AiSi \qquad (I-2)$$

where the A's are prediction weighting constants.

We first design the predictor for the case $E\{So\}\neq\emptyset$, and from it we deduce the solution for the case $E\{So\}=\emptyset$. A difference signal or predictor is then defined as:

$$do=So-\hat{So}=So-(Ao+\sum_{i=1}^{n} AiSi) \qquad (I-3)$$

In most designs, the prediction weighting constants are chosen to minimize the mean square prediction error.

$$D=E\{do^2\}=E\{[So-(Ao+\sum_{i=1}^{n} AiSi)^2] \qquad (I-4)$$

The rational for this performance measure is that the measure is tractable, correlates reasonably well with subjective evaluation, and the quantizer error is directly proportional to the mean square-prediction error. Minimization of the mean-square prediction error can be performed by taking the partial derivative of D with respect to each weighting constant and setting the result to zero. The differentiation with respect to Ao results in:

$$\frac{\partial D}{\partial Ao}=\frac{\partial E\{[So-(Ao+\sum_{i=1}^{n} AiSi)]^2\}}{\partial Ao}=\emptyset \qquad (I-5)$$

$$\frac{\partial D}{\partial Ao}=-2E\{So-(Ao+\sum_{i=1}^{n} AiSi)\}=\emptyset \qquad (I-6)$$

which yields:

$$E\{So\}=E\{\hat{So}\}=E\{Ao+\sum_{i=1}^{n} AiSi\} \qquad (I-7)$$

$$E\{So\}=Ao+\sum_{i=1}^{n} AiE\{Si\} \qquad (I-8)$$

The differentiation with respect to each of the remaining weighting value Aj, j=1....n, is:

$$\frac{\partial D}{\partial Aj}=\frac{\partial E\{[So-(Ao+\sum_{i=1}^{n} AiSi)]^2\}}{\partial Aj}=\emptyset \qquad j=1,...,n \qquad (I-9)$$

$$\frac{\partial D}{\partial Aj}=-2E\{[So-(Ao+\sum_{i=1}^{n} AiSi)]Sj\}=\emptyset \qquad j=1,...,n \qquad (I-10)$$

which gives:

$$E\{SoSj\}=Ao\{Sj\}+\sum_{i=1}^{n} AiE\{SiSj\} \qquad j=1,...,n \qquad (I-11)$$

$$Roj=Ao\{Sj\}+\sum_{i=1}^{n} AiRij \qquad j=1....,n \qquad (I-12)$$

where

$$Roj=E\{SoSj\} \qquad j=1....,n$$

is the correlation between the variables So and Sj, and:

$$Rij=E\{SiSj\} \qquad i,j=1.....,n \qquad (I-13)$$

is the correlation between the variables Si and Sj. The optimum prediction coefficients Ao,A1,A2....,An, are the solution of the n+1 algebraic equations

$$E\{So\}=Ao+\sum_{i=1}^{n} Ai\{Si\} \qquad \text{(I-14a)}$$

and

$$Roj=Ao\{Sj\}+\sum_{i=1}^{n} AiRij \qquad j=1,....,n \qquad \text{(1-14b)}$$

The mean squared value of then prediction error is given by:

$$\sigma_d^2 = E\{(So-\hat{So})^2\} = E\{S-\hat{So})(So-\hat{So})\}$$

$$\sigma_d^2 = E\{(So-\hat{So})So\}-E\{(So-\hat{So})\hat{So}\} \qquad \text{(I-15)}$$

If the optimum predictor coefficients (obtained from Eq.2.2.14) are used, then the prediction error is uncorrelated with (orthogonal to) the past value (or any combination of the past values) of the predictor input [168], i.e.:

$$E\{(So-\hat{So})\hat{So}\}=\emptyset \qquad \text{(I-16)}$$

and therefore, the minimum mean-square prediction error is found to be:

$$\sigma_d^2 = E\{(So-\hat{So})So\} = E\{So^2\} - E\{So\hat{So}\} \qquad (I-17)$$

$$= \sigma^2 - (Ao + \sum_{i=1}^{n} AiRoi) \qquad (I-18)$$

where σ^2 is the variance of the input sequence {So}. The error sequence {do} is less correlated and has smaller variance than the signal sequence {So}. The use of linear prediction has produced a sequence {do} from which the sequence {So} can be reconstructed. The variance σ_d^2 of the error sequence {do} is less than the variance of the original sequence {So} by the amount shown in the parenthesis in Eq.2.2.18. As $n \rightarrow \infty$ then the sequence of the error samples can always be made completely uncorrelated. If {So} is an n-th. order Markov sequence, then only n previous samples are enough for forming the best estimate of So, so that the error sequence will be uncorrelated.

Now, if the input sequence {So} has zero mean, i.e.:

$$E\{So\} = \emptyset \qquad (I-19)$$

then all the variables S1,S2,.....,Sn will have zero mean:

$$E\{S_i\} = \emptyset \qquad i=1,2,.....,n \qquad (I-2\emptyset)$$

and Eq. I.14 results in:

$$Ao=\emptyset \qquad \text{(I-21a)}$$

and:

$$Roj=\sum_{i=1}^{n} AiRij \qquad j=1,2.....,n \qquad \text{(I-21b)}$$

The optimum values A1,....,An, are then found from the n linear equations of Eq.(I-21b). The minimum mean-square prediction error is given by:

$$\sigma_d^2 = \sigma^2 - \sum_{i=1}^{n} AiRoj \qquad \text{(I-22)}$$

REFERENCES

1 Cabrey R.L., "Video transmission over telephone cable pairs by pulse code modulation.", Proc. IRE, Vol. 48, Sept.1960, pp.1546-1561.

2 Harper L.H., "PCM picture transmissions.", IEEE Spectrum, Vol. 3, June 1966, pp. 146.

3 Schreiber W.F., "The measurement of third order probability distribution of television signals.", IRE Trans. Inform. Theory, Vol. IT-2, Sept.1956, pp.95-105.

4 Kretmer E.R., "Statistics of television signals.", Bell Systems T.J., July 1952, pp. 751-763.

5 Andrews H.C. and Pratt W.K., "Fourier transform coding of images.", Proc. Hawai Inter. Conf. Systems Sci., Western, Jan.1968, pp. 677-679.

6 Anderson G.B. and Huang T.S., "Piecewise Fourier transformation for picture bandwidth compression.", IEEE Trans. Commun. Tech., Vol. COM-19, No. 2, April 1971, pp. 133-140.

7 Pratt W.K. and Andrews H.C., "Application of Fourier Hadamard transformation to bandwidth compression.",

Presented at the 1969 Inter. Symp. on Picture Bandwidth Compression, pp. 515-554.

8 Enomoto H. and Shibata K., "Features of Hadamard transformed television signals.", 1965 Nat. Conf. IECE, Japan, paper No. 881.

9 Pratt W.K., Kane J. and Andrews, "Hadamard transform image coding.", Proc. IEEE, Vol. 57, No. 1, 1969, pp. 58-68.

10 Wood J.W. and Huang T.S., "Picture bandwidth compression by linear transformation and block quantization.", Presented at Symp. Picture Bandwidth Compression, 1969, pp. 555-573.

11 Tasto M. and Wintz P.A., "Image coding by adaptive block quantization.", IEEE Trans. Commun. Tech., Vol COM-19, 1971, pp. 957-972.

12 Habibi A. and Wintz P.A., "Image coding by linear transformation and block quantization.", IEEE Trans. Commun. Tech., Vol. COM-19, 1971, pp. 50-62.

13 Haar A., "Zur theorie des orthogonalen function systeme." Inaugural Dissertation, Math. Ann., Vol. 71, 1972, pp. 33-53.

14 Enomoto H. and Shibata K., "Orthogonal transform coding system for television signals.", J. of the Institute of TV Engineers of Japan, Vol. 24, No. 2, Feb. 1970, pp. 99-108.

15 Pratt W.K., Welch L.R. and Chen W.H., "Slant transforms for image coding.", Proc. Symp. on Application of Walsh Function, March 1972, pp. 229-234.

16 Pratt W.K., Chen W.H. and Welch L.R., "Slant transform image coding.", IEEE Trans. Commun. Tech., Vol. COM-22, 1974, pp. 1075-1093.

17 Pratt W.K., "Two dimensional unitary transforms.", Proc. of the Nato Advanced Study Institute on Digital Image Processing and Analysis, Bonas, France, 24-25 june 1976, pp. 1-21.

18 Wintz P.A., "Transform picture coding.", Proc. IEEE, Vol. 60, July 1972, pp. 809-820.

19 Andrews H.C., "Entropy consideration in frequency domain.", Proc. IEEE (Proceeding letters), Jan. 1968, pp. 113-114.

20 Wintz P.A. and Kurtenbach A.J., "Waveform error control in PCM telemetry.", IEEE Trans. Inform. Theory, Vol. IT-14, 1968, pp. 650-661.

21 Karhunen H., "Ueber die linieare methoden in der wahrscheinlichkeitrechnung.", Ann. Acad. Science Fenn, Ser. A.I. 37, Helsinki 1947.

22 Brown Jr.J.L., "Mean Square truncation error in series expansion of random functions.", J.SIAM, Vol. 8, March 196Ø, pp. 18-32.

23 Hotelling H., "Analysis of a complex of statistical variable into principal components.", J. Educ. Psychology, Vol. 24, 1933, pp. 417-441 and pp. 498-52Ø.

24 Kramer H.P. and Mathew M.V., "A linear coding for transmitting a set of correlated signals.", IRE Trans. Inform. Theory, Vol. 2, 1956, pp. 41-46.

25 Huang J.J.Y. and Schultheiss P.M., "Block quantization of correlated Gaussian random variables.", IEEE Trans. Commun. Systems, Vol. CS-11, Sept. 1963, pp. 289-296.

26 Pratt W.K. and Andrews H.C., "Transform image coding.", Electron. Sci. Lab., Univ. Southern California, Los Angeles, Rep. 387, March 197Ø.

27 Netravali R.N. and Limb J.O., "Picture coding : A review" Proc. IEEE, Vol. 68, No. 3, March 198Ø, pp. 366-4Ø6.

28 Landau H.J. and Slepian D., "Some experiments in picture processing for bandwidth reduction.", Bell Sys. Tech. J., Vol. 50, No. 5, 1971, pp. 1525-1540.

29 Jain A.K., "A fast Karhunen-Loeve transform for finite discrete images.", Proc. Natio. Electronic Conf., Chicago, Oct. 1974, pp. 323-328.

30 Jain A.K., "A fast Karhunen-Loeve transform for a class of random processes.", IEEE Trans. Commun., Sept. 1974, pp. 1023-1029.

31 Mieri A.Z. and Yudilevich E., "A pinned sine transform image coder.", IEEE Trans. Commun., Vol. COM-29, No. 3, Dec. 1981, pp. 1728-1735.

32 Haralick R.M., Griswold N. and Kattivakuwanich N., "A fast two-dimensional Karhunen-Loeve transform.", Proc. SPIE, Vol. 66, 1975, pp. 144-155.

33 Cooley J.W. and Tukey J.W., "An Algorithm for the machine calculation of complex Fourier series.", Mathematics of Computation, Vol. 19, No. 90, 1965, pp. 297-301.

34 Brigham E.O. and Morrow R.E., "The Fast Fourier Transform.", IEEE Spectrum, Vol. 4, No. 12, Dec. 1972, pp. 61-70.

35 Cooley J.W., Lewis P.A.W. and Welch P.D., "Historical notes on the fast Fourier transform.", Proc. IEEE, Vol. 55, Oct. 1967, pp. 1675-1677.

36 Anderson G.B. and Huang T.S., "Picture bandwidth compression by piecewise Fourier transform.", In Proc. Purdue Centennial Year Symp., Inform. Processing 1969, pp. 532-541.

37 Boyer D.E., "Walsh functions, Hadamard matrices and data compression.", Presented at 1971 Walsh Function Symp., pp. 33-37.

38 Habibi A. and Wintz P., "Linear transformation for encoding two dimensional sources.", TR-EE 70.2, Purdue University, June 1970.

39 Ahmed N., Natarajan T. and Rao K.R., "Discrete Cosine Transform.", IEEE Trans. Computer, Jan. 1974, pp. 90-93.

40 Chen W., Smith C.H. and Fralick S., "A fast computation algorithm for the discrete Cosine transform.", IEEE Trans. Commun., Vol. COM-25, Sept. 1977, pp. 1004-1009.

41 Lynch R.T. and Reis J.J., "Haar transform image coding.", 1976 National Telecommun. Conf. USA., 29 Nov.- 1 Dec. 1976, pp. 443/1-5.

42 Chen W.H., "Slant transform image coding.", USCEE Report 441. Technical report, Univ. of Southern California 1973.

43 Haralick R.M. and Shanmugam K., "Comparative study of a discrete linear basis for image data compression.", IEEE Trans. System, Man and Cybernetics, Vol. SMC-4, Jan 1974, pp. 16-27.

44 Tescher A.G., "Adaptive transform image coding.", IEEE Sixteeenth Asilomar Conf. on Circuits, Systems and Computer., 8-12 Nov. 1982, pp. 50-55.

45 Pratt W.K., "Digital image processing.", 1978.

46 Tasto M. and Wintz P.A., "A bound on the rate distortion function and applications to images.", IEEE Trans. Inform. Theory, Vol. IT-18, Jan. 1972, pp. 150-159.

47 Oppenheim A.V., "Application of digital signal processing.", 1978 , Chap. 4.

48 Claire E.J., Faber S.M. and Green R.R., "Practical techniques for transform data compression/image coding.", 1971 Proc. of Walsh Function Symp., 13-15 April 1971, pp. 2-5.

49 Rauch H., "Image compression by block coding and multiplexing technique for monochromatic pictures.",

Intr. Conf. on Video Data Recording, Southampton, England, 24-27 July 1979, pp. 287-294.

50 Tasto M. and Wintz P.A., "Picture bandwidth compression by adaptive block quantization.", Purdue Univ., Lafayette, In. Tech. Rep. TR-EE 70-14, June 1970.

51 Sakrison D.J. and Algazi V.R., "Compression of line by line and two dimensional encoding of random images.", IEEE Trans. Inform. Theory, Vol. IT-17, July 1971, pp. 386-398.

52 Saghri J.A., Tescher A.G. and Habibi A., "Block size consideration for adaptive image coding.", National Telesystems Conference, Nov. 1982, pp. E1-2-1-E1-2-4.

53 Haralick R.M., Groswold N.C. and Paul C.A., "An annihilation transform compression method for permuted images.", SPIE, Vol. 87, 1976, pp. 189-196.

54 Gonzalez R.C. and Wintz P., "Digital image processing.", 1977.

55 Rao K.R., Narasimhan M.A. and Revoluri, "Simulation of image data compression by transform threshold coding.", Proc. 7th. Annual Southeastern Symp. on System Theory, 20-21 March 1975, pp. 67-73.

56 Panter P.F. and Dite W., "Quantization distortion in pulse count modulation with non-uniform spacing of levels.", Proc. IRE, Vol. 39, Jan 1951, pp. 44-48.

57 Max J., "Quantization for minimum distortion.", IRE Trans. Inform. Theory, Vol IT-6, March 1960, pp. 7-12.

58 Peaz M.D. and Glisson T.H., "Minimum mean squared error quantization in speech PCM and DPCM systems.", IEEE Trans. Commun., Vol. COM-20, April 1972, pp. 225-230.

59 Roe G.M, "Quantization for minimum distortion.", IEEE Trans. Inform. Theory, Vol. IT-10, No. 4, 1964, pp. 175-176.

60 Habibi A. and Hershel R.S., "A unified representation of differential pulse code modulation (DPCM) and transform coding systems.", IEEE Trans. Commun., Vol. COM-22, May 1974, pp. 692-696.

61 Mitrakos D.K. and Constantinides A.G., "Optimum block quantization in signal processing.", IEE Proc., Vol. 130, No. 6, Oct. 1983, pp. 543-547.

62 Mounts F.W., Netravali A.N. and Prasada B., "Design of quantizers for real time Hadamard transform coding of pictures.", Bell Syst. Tech J., Vol. 56, No. 1, Jan. 1977, pp. 21-48.

63 Charles W. and Wintz P.A., "Picture bandwidth reduction for noisy channels.", Purdue University, Lafayette, Report TR-EE 70-30, Aug. 1971.

64 Hayes J.F. and Bobilin R., "Efficient waveform encoding.", School of Electrical Engineering, Purdue University, Tech. Rep. TR-EE 69-9, Feb.1969.

65 Dillard G.M., "Application of ranking techniques to data compression for image transmission.", NTC 75 Conference, Record 1, pp. 22-18 to 22-22.

66 Claire E.J., "Bandwidth reduction in image transmission.", In. Conf. Rec., 1972 IEEE Int. Conf. Communications, pp. 39-8 to 39-13.

67 Gimlett J.I., "Use of activity classes in adaptive transform image coding.", IEEE Trans. Commun., Vol. COM-23, 1975, pp. 785-786.

68 Chen W.H. and Smith C.H., "Adaptive coding of monochrome and colour images.", IEEE Trans. Commun., Vol. COM-25, 1977, pp. 1285-1292.

69 Kekre H.B. and Aleen S.A., "Adaptive linear quantization and recontruction scheme in the direct cosine transform domain.", Int. J. Electronics, Vol. 54, 1983, pp. 31-45.

70 Modestiono J.W., Farvardin N. and Ogrinc M.R., "Performance of block cosine image coding with adaptive quantization.", National Telesystems Conference, 1982, pp. E1.1.1-E1.1.6.

71 Tescher A.G., "The role of adaptive phase coding in two and three dimensional Fourier and Walsh image compression.", Applications of Walsh Functions and Sequency Theory, IEEE, New York 1974,pp. 26-65.

72 Tescher A.G., Andrews H.C. and Habibi A., "Adaptive phase coding in two and three dimensional Fourier and Walsh image compression.", 1974 Picture Coding Symp., Goslar, Germany, Aug. 26-28.

73 Parsons J.R. and Tescher A.G., "An investigation of MES contributions in transform image coding schemes.", Proc. SPIE, Vol. 66, 1975, pp. 196-206.

74 Cox A.V. and Tescher A.G., "Generalized adaptive transform coding.", 1976 Picture Coding Symp., Asilomar, Jan. 28-30.

75 Tescher A.G. and Cox A.V., "An adaptive transform algorithm.", 1976 Inter. Conf. Commun., 14-16 June, pp. 47-20 to 47-25.

76 Reader C., "Intraframe and interframe adaptive transform coding.", Proc. SPIE, Vol. 66, Aug. 1975, pp. 108-118.

77 Schaming W.B., "Digital image transform coding.", PE-622, Inter. Memorandum, RCA Corporation 1974.

78 Nagan K.N., "Adaptive transform coding of video signals.", IEE Proc., Vol. 129, Feb. 1982, pp. 28-40.

79 Rosse J.A., "Interframe coding of digital images using transform and hybrid transform/predictive techniques.", Tech. Rep. USCIPI 700, Univ. of Southern California, June 1976.

80 Natarajan T.R. and Ahmed N., "On interframe transform coding.", IEEE Trans. Commun., Vol. COM-23, 1977, pp. 1323-1329.

81 Cuttler C.C., "Differential quantization of communication signals.", Patent No. 2.605.351, July 29 1952, United States Patent Office.

82 Oliver B.N., "Efficient coding.", Bell Syst. Tech. J., July 1952, pp. 724-750.

83 Harrison C.W., "Experiment with linear prediction in television.", Bell Syst. Tech. J., July 1952, pp. 764-783.

84 Elias P., "Predictive coding.", IRE Trans. Inform. Theory, Vol. IT-1, March 1955, pp. 16-33.

85 Graham R.E., "Predictive quantizing of television signals.", IRE WESCON Record 2, Prat 4, 1958, pp. 147-156.

86 O'Neal Jr.J.B., "Predictive quantizing systems (Differential Pulse Code Modulation) for the transmission of television signals.", Bell Syst. Tech. J., May-June 1966, pp. 689-721.

87 Connor D.J., Pease R.F.W. and Scholes W.G., "Television coding using two dimensional spatial prediction.", Bell Syst. Tech. J., Vol. 50, 1971, pp. 1049-1061.

88 Habibi A. and Robinson G.S., "A survey of digital picture coding.", IEEE Trans. Computers, Vol. 7, 1974, pp. 22-24.

89 Habibi A., "Comparison of n.-th order DPCM encoder with linear transformation and block quantization techniques.", IEEE Trans. Commun. Technology, Vol. COM-19, Dec. 1971, pp. 948-956.

90 Huffman D.A., "A method for the construction of minimum redundancy codes.", Proc. IRE, Vol. 40, Sept. 1952, pp. 1098-1101.

91 Bodycomb J.V. and Hadad A.H., "Some properties of a predictive quantizing system.", IEEE Trans. Commun. Technology, Vol. COM-18, Oct. 1970, pp. 682-684.

92 Geddes W.K.E., "Picture processing by quantization of the time derivative.", BBC Eng. Div., Tech. Rep. T-114,1963.

93 Woods J.W., "Stability of DPCM coders for television.", IEEE Trans. Commun., Vol. COM-23, 1975, pp. 845-846.

94 Pirsch P., "Stability conditions for DPCM coders.", IEEE Trans. Commun., Vol. COM-30, 1982, pp. 1174-1184.

95 Millard J.B. and Maunsell H.I., "Digital encoding of the video signal.", Bell Syst. Tech. J., Vol. 50, 1971, pp. 459-479.

96 Protonotarias E.N., "Slope overload noise in differential pulse code modulation systems.", Bell Syst. Tech. J., Vol. 46, 1967, pp. 2119-2161.

97 Goldstein L.H. and Lie A., "Quantization noise in ADPCM systems.", IEEE Trans. Commun., Vol. COM-25, 1977, pp. 227-238.

98 Abbott R.P., "A differential pulse code modulation Codec for video telephone using four bits per sample.", IEEE

Trans. Commun. Technology, Vol. COM-19, 1971, pp. 907-912.

99 Limb J.O. and Mounts F.W., "Digital differential quantizer for television.", Bell Syst. Tech. J., Vol. 48, 1969, pp. 2583-2599.

100 Ready P.J. and Spencer D.J., "Block adaptive DPCM transmission of images.", NTC 75 Conference Rec., Vol. 2, pp. 22/10-22/17.

101 Brown E.P., "Sliding-scale operation of differential type PCM coders for television.", Bell Syst. Tech. J., Vol. 48, 1969, pp. 1537-1553.

102 Musmann H.G., "Codierung von video signalen.", Nachrichten Tech. Z., Vol. 24, 1971, pp. 114-116.

103 Musmann H.G., "A comparison of extended differential coding schemes for video signals.", Proc. 1974 IEEE Int. Zurich Seminar on Digital Commun., 1974, pp. C1(1)-C1(7).

104 Lueder R., "Adaptive Differenz-Pulscodemodulation fuer video signale.", Arch. Electron. and Uebertragungstech , Vol. 29, 1975, pp. 251-256.

105 Cohen F., "A switched quantizer for nonlinear coding of video signals.", Nachrichtentech. Z., Vol. 25, 1972,

pp.554-559.

106 Kummerow T., "Statistics for efficient linear and non-linear picture coding.", In. Proc. Int. Telemetering Conf., Vol. 8, 1972, pp. 149-161.

107 Kummerow T., "Ein DPCM system mit zweidimensionalem praediktor und gesteuertem quantisierer.", Tagungsbericht NTG-Fachtagung, "Signalverarbeitung", April 1973, pp. 425-439.

108 Candy J.C. and Bosworth R.H., "Method for designing differential quantizers based on subjective evaluation of edge busyness.", Bell Syst. Tech. J., Vol. 51, 1972, pp. 1495-1516.

109 Limb J.O. and Rubinstein C.B., "On the design of quantizers for DPCM coders: A functional relationship between probability and masking.", IEEE Trans. Commun., Vol. COM-26, 1978, pp. 573-578.

110 Rubinstein C.B. and Limb J.O., "On the design of quantizers for DPCM coders: Influence of the subjective testing methodology.", IEEE Trans. Commun., Vol. COM-26, 1978, pp. 565-572.

111 Sharma D. and Netravali A.N., "Design of quantizer for DPCM coding of picture signals.", IEEE Trans. Commun.,

Vol. COM-25, 1977, pp. 1267-1274.

112 Pirsch P., "Design of DPCM quantizers for video signals using subjective tests.", IEEE Trans. Commun., Vol. COM-29, 1981, pp. 990-1000.

113 Schaefer R., "Design of adaptive and non-adaptive quantizers using subjective criteria.", Signal Processing Vol. 5, 1983, pp. 333-345.

114 Prasada B.,Mounts F.W. and Netravali A., "Level reassignment: A technique for bit-rate reduction.", Bell Syst. Tech. J., Vol. 57, 1978, pp. 61-73.

115 Prasada B., Netravali A. and Kobran A., "Adaptive companding of picture signals in a predictive coder.", IEEE Trans. Commun., Vol. COM-26, pp. 161-164.

116 Netravali A. and Prasada B., "Adaptive quantization of picture using spatial masking.", Proc. IEEE, Vol. 65, 1977, pp. 536-548.

117 Zschunke W., "DPCM picture coding with adaptive predictor.", IEEE Trans. Commun., Vol. COM-25, 1977, pp. 1295-1302.

118 Dinstein I. and Garlow R., "A DPCM encoder for TV. signals using selective prediction and entropy coding.",

NTC 75 Conference Record, Vol. 2, pp. 22/26-22/30.

119 DeJager F.E., "Delta modulation, a method of DPCM transmission using a 1-unit code.", Philips Res. Rep., Dec. 19, pp. 442-466.

120 Schouten J.S., DeJager F.E. and Greefkes J.A., "Delta modulation, a new modulation system for telecommunications, Philips Tech. Rep., March 1952, pp. 237-245.

121 Jayant N.S., "Digital coding of speech waveforms PCM, DPCM and DM quantizers.", Proc. IEEE, Vol. 62, 1974, pp.611-632.

122 Lei T.R., Scheinberg N. and Schilling D.L., "Adaptive delta modulation systems for video encoding.", IEEE Trans. Commun., Vol. COM-25, 1977, pp. 1302-1314.

123 Winkler M.R., "High information delta modulation.", IEEE Int. Conv. Rec. Pt. 8, 1963, pp. 260-265.

124 Winkler M.R., "Pictorial transmission with HIDM.", IEEE Int. Conv. Rec. Pt.1, 1965, pp. 285-290.

125 Bosworth R.H. and Candy J.C., "A companded one bit coder for television transmission.", Bell Syst. Tech. J., Vol. 48, 1969, pp. 1459-1479.

126 Jayant N.S., "Adaptive delta modulation with one bit memory.", Bell Syst. Tech. J., Vol. 49, 1970, pp. 321-342.

127 Song C.L., Gardinick J. and Schilling D.L., "A variable step size robust delta modulator.", IEEE Trans. Commun., Vol COM-19, 1971, pp. 1033-1099.

128 Scheinberg N. and Schilling D.L., "Techniques for correcting transmission errors in video adaptive delta modulation channels.", IEEE Trans. Commun., Vol. COM-24, 1976, pp. 1064-1069.

129 Schilling D.L., Scheinberg N. and Garodnick J., "Video encoding using adaptive delta modulation.", IEEE Trans. Commun., Vol. COM-26, 1978, pp. 1682-1689.

130 Habibi A., "Delta modulation and DPCM coding of colour signals.", Proc. of International Telemetry Conference, Vol. 8, Los Angeles, Oct. 10-12 1972, pp. 333-343.

131 Cuttler C.C., "Delayed encoding stabiliser for adaptive coders.", IEEE Trans. Commun., Vol. COM-19, 1971, pp. 898-907.

132 Habibi A., "Hybrid coding of pictorial data.", IEEE Trans. Commun., Vol. COM-22, 1974, pp. 614-624.

133 Ishii M., "Picture bandwidth compression by DPCM in the Hadamard transform domain.", Fujitsu Scientific and Tech. J., Vol. 10, 1974, pp. 51-65.

134 Rao K.R., Narasimham M.A. and Gozinski W.J., "Processing image data by hybrid techniques.", IEEE Trans. Syst. Man. and Cybernetics, Vol. SMC-7, 1977, pp. 728-734.

135 Netravali A., Prasada B. and Mount F., "Some experiments in adaptive and predictive Hadamard transform coding of picture.", Bell Syst. Tech. J., Vol. 56, 1977, pp. 1531-1547.

136 Roese J.A., Pratt W.K. and Robinson G.S., "Interframe cosine transform image coding.", IEEE Trans. Commun., Vol. COM-25, 1977, pp. 1239.

137 Stuller J.A. and netravali A.N., "Transform domain motion estimation.", Bell Syst. Tech. J., 1979, pp. 1673-1702.

138 Netravali A.N. and Stuller J.A., "Motion compensated transform coding.", Bell Syst. Tech. J., 1979, pp. 1703-1718.

139 Lathi B.P., "An introduction to random signals and communication theory.", 1968.

140 Hunt B.R., "Non-stationary statistical image models and

their application to image data compression.", Computer Graphics and Image Processing, Vol. 12, 1980, pp. 173-180.

141 Stricklard R.N., "Adaptive data compression by transformations for generating stationary statistical image models.", Inter. Conf. on Electronic Processing, 26-28 July 1982, pp. 188-193.

142 Yan J.K. and Sakrison D.J., "Encoding of images based on a two-component source model.", IEEE Trans. Commun., Vol. COM-25, 1977, pp. 1315-1322.

143 Mitrakos D.K., "On a composite source model for image compression.", Tech. Rep., Digital Signal Processing Group, Dept. Of Electrical Engineering, Imperial College, 1981.

144 Mitrakos D.K. and Constantinides A.G., "Composite source coding technique for image bandwidth compression.", Digital Signal Processing Group, Dept. of Electrical Engineering, Imperial College.

145 Zetterberg L.H., Ericsson S. and Brusewitz H., "Interframe DPCM with adaptive quantization and entropy coding.", IEEE Trans. Commun., COM-32, 1984, pp. 457-462.

146 Smith B., "Instantaneous companding of quantized

signals.", B.S.T.J., May 1957, pp. 653-709.

147 Mannos J.L. and Sakrison D.J., "The effects of a visual fidelity criterion on the encoding of images.", IEEE Trans. Inform. Theory, Vol. IT-20, 1974, pp.525-536.

148 Jayant N.S., "Adaptive quantization with one word memory.", Bell Syst. Tech. J., Vol. 52, 1973, pp. 1119-1144.

149 Zettenberg L.H., Ericsson S. and Couturier C., "DPCM picture coding with two dimensional control of adaptive quantization.", IEEE Trans. Commun., COM-32, No.4, April 1984, pp. 457-462.

150 Limb J.O., "Adaptive encoding of picture signals.", In Picture Bandwidth Compression, edited by T.S. Huang and O.J. Tretiak, published by Gordon and Breach, 1972, pp. 341-382.

151 Gibson J.D., Jones S.K and Melsa J.L., "Sequentially adaptive prediction and coding of speech signals.", IEEE Trans. Commun., COM-22, No. 11, Nov. 1974, pp. 1789-1797.

152 Netravali A.N. and Rubinstein C.B., "Luminance adaptive coding of chrominance signals.", IEEE Trans. Commun., COM-27, April 1979, pp. 703-710.

153 Schwartz J.W. and Barker R.C., "Bit plane encoding: A technique for source encoding.", IEEE Trans. Aerospace Electron. Syst., Vol. AES-2, No. 4, July 1966, pp. 385-392.

154 Spencer D.R. and Huang T., "Bit plane encoding of continuous tone pictures.", Symposium on Computer Processing in Communications, Polytechnic Institute of Brooklyn, New York, April 1969.

155 Vuong S.T., "Linear transformation is bad for the coding of graphic images.", Proceeding of the 7th. Canadian Man-Computer Commun. Conference, 10-12 June 1981, Waterloo, Canada.

156 Clark A.D., "Image processing for three dimensional object identification.", Internal report, Leicester Polytechnic, 1983.

157 Tsetlin M.L., "On the behavior of finite automaton in random media.", Automation and Remote Control, Vol. 22, 1961, pp. 1345-1354.

158 Narendra K.S. and Thathachar M.A.L., "Learning automata : A survey.", IEEE Trans. Syst. Man and Cybernetics, SMC-4, 1974, pp. 323-334.

159 Varshavskii V.I. and Vorontsova I.P, "On the behavior of

stochastic automata with variable structure.", Automation and Remote Control, Vol. 24, 1963, pp. 327-333.

160 Narendra K.S., Special volume on learning automata., J. Cybern. and Inf. Sci., Vol. 1, 1977, pp. 2.

161 Lakshmivarahan S., "Bibliography on learning automata.", J. Cybern. and Inf. Sci., Vol. 2-3, 1977, pp. 189.

162 Devroye L.P., "Probabilistic search as a search solution procedure.", IEEE Trans. Syst. Man and Cybernetics, Vol. SMC-5, 1975, pp. 315-321.

163 Watanabe S., "Knowing and guessing.", John Wiley, 1969.

164 Lakshmivarhan S. and Thathachar M.A.L., "Absolutely expedient learning algorithms for stochastic .", IEEE Trans. Syst. Man and Cybernetics, Vol. SMC-3, 1973, pp. 281-286.

165 Narendra K.S. and Wheeler R.M., "An N-player sequential stochastic game with identical payoffs.", IEEE Trans. Syst. Man and Cybernetics, Vol. SMC-13, 1983, pp. 1154-1158.

166 Shapiro I.J. and Narendra K.S., "Use of stochastic automata for parameter self optimization with multimodal performance criteria.", IEEE Trans. Syst. Sci. Cybern.,

Vol. SCC-5, 1969, pp. 352-360.

167 Viswanathan R. and Narendra K.S., "A note on the linear reinforcement scheme for variable structure stochastic automata.", Becton Center, Yale University, New Haven, Conn., Tech. Rept. CT-42, July 1971.

168 Papoulis A., "Probability, random variables and stochastic processes.", Mc. Graw-Hill Inc, 1981.