21

Reprinted from Photo-Opt. Instrum. Eng. Soc. Proc. 87:24-35 (1976)

ADAPTIVE HAAR TRANSFORM VIDEO BANDWIDTH REDUCTION SYSTEM FOR RPV's

J. J. Reis
R. T. Lynch
J. Butman*
Northrop Research and Technology Center
3401 West Broadway
Hawthorne, California 90250

Abstract

A practical video bandwidth compression system is described in detail. Video signals are Haar transformed in 2 dimensions and the resulting transform coefficients are adaptively filtered to achieve reduced transmission data rates to less than 1 bit/pel, while maintaining good picture quality.

Error detection and compensation is included into the transmission bit structure.

Introduction

The wideband video link of Remotely Piloted Vehicles (RPV's) is subject to both friendly and hostile interference. One way of minimizing the effects of jamming is to reduce the information bandwidth as much as possible before it is transmitted by a spread spectrum modem.

Northrop's approach for solving the RPV real time video bandwidth reduction problem is based upon a digital implementation of a 2-D fast rationalized Haar transform with a combination zonal and adaptive threshold compression algorithm. This concept is a natural outgrowth of Northrop's experience in designing and developing video processing systems based on the Haar transform. Figure 1 is a block diagram of the airborne portion of the proposed system. Input video, obtained from a TV camera, is preemphasized to reduce quantization noise before being digitized to 8 bits and transformed in two dimensions by using a fast rationalized version of the Haar transform. This transformation process maps a two-dimensional spatial picture of dimension 16 pels by 16 pels into a block of transform coefficients of dimension 16 x 16.

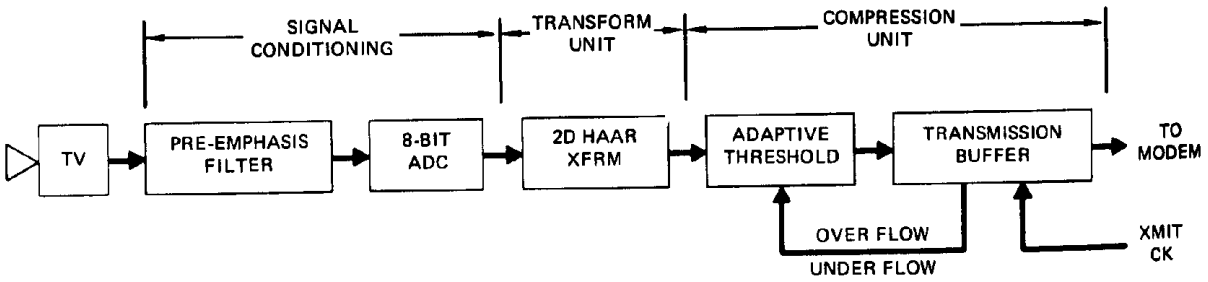


Figure 1. Block Diagram of Airborne Portion of the Northrop Real-Time Video Bandwidth Reduction System.

The purpose of the Haar transform is to transform a picture into a form where most of the picture energy is represented by a relatively small number of transform coefficients. These transform coefficients are then processed by a zonal filter and an adaptive threshold filter. The zonal filter always selects certain low order coefficients for transmission. The adaptive threshold filter selects from the remaining coefficients those coefficients whose amplitudes are greater than the threshold level. The number of coefficients to be selected by the adaptive threshold algorithm is determined by the transmission rate of the output modem and picture activity.

In addition to the spatial compression provided by the transformation and adaptive filtering, an additional 8:1 temporal compression is achieved by processing only 1/8 of each field as suggested by researchers at the Naval Undersea Center (NUC) in San Diego.[1,4] This 8:1 reduction is achieved by processing the picture in vertical stripes, 32 pels-wide by 240** lines long. Each successive field uses the next 32 pel wide stripe across the horizontal scan. A complete frame is sent after 8 fields. Because a frame store is not required, this method of obtaining temporal compression without introducing flicker in the reconstructed image permits a significant hardware reduction to be realized. Although this method produces temporal skewing of the reconstructed video data, under most operating conditions this artifacting should not be overly objectionable.

* Northrop Electronics Division, One Research Park, Palos Verdes Peninsula, California 90274
** 16 additional lines are taken up by the Vertical Interval

The airborne modem transmits the selected transform coefficients to the ground unit where they are inverse filtered and transformed and displayed on a TV monitor. Figure 2 shows the block diagram for the ground terminal of the real-time video bandwidth reduction system. Transform coefficients from the output of the receiver modem are accumulated by the input receiver buffer which arranges them into data blocks. The inverse compression unit takes these data blocks and expands them into transform coefficient blocks of dimension 16 x 16, filling in a zero for each coefficient discarded prior to transmission by the airborne compression unit. The 2-D inverse Haar transform re-maps these transform domain coefficients back into the spatial domain where they are transferred to the frame store.

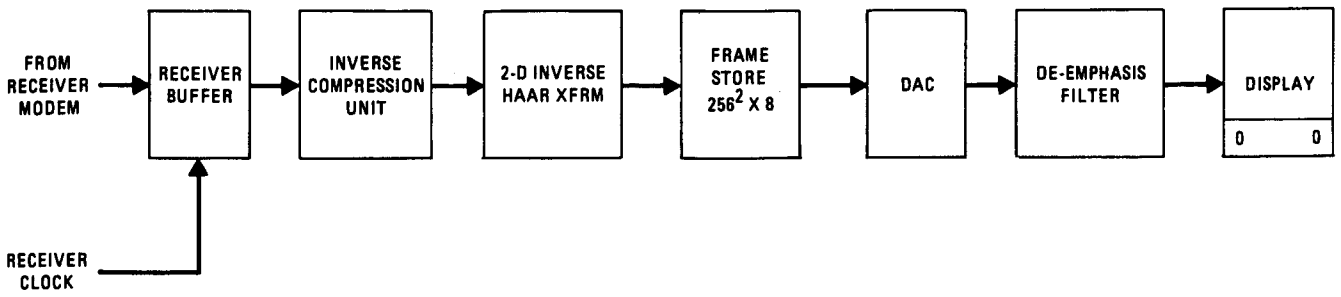


Figure 2. Block Diagram of Ground Terminal of the Northrop Real-Time Video Bandwidth Reduction System.

Airborne Subsystem

2-Dimensional Haar Transform Unit

The Northrop approach to developing a small, real time video bandwidth reduction system for RPV's utilizes a two-dimensional fast rationalized Haar transform to convert spatial domain video samples into transform domain coefficients. This transformation is given by equation 1 through 3.

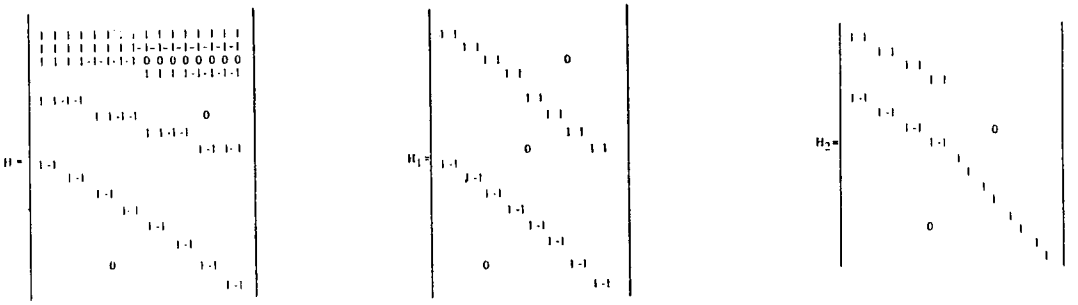
$$B = HAH^T \tag{1}$$

where
B = Square Matrix of Transform Coefficients
A = Square Matrix of Input Spatial Video Samples
H = One-Dimensional Rationalized Haar Transform
T = Transpose

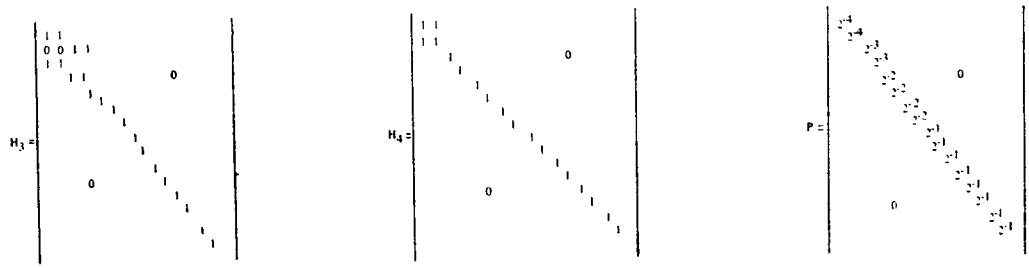
The one-dimensional Haar transform which is the basis for the 2-D transforms, has an extremely sparse matrix and a fast computational form as shown in Equation 2 and 3 .

$$H = H_4 H_3 H_2 H_1 \tag{2}$$

where H, H_1, H_2, H_3, H_4 and P are defined by equation set 3.



(3)



(3)

and the inverse transform is given by equation 4.

$$H^{-1} = H^{T}P = H_1^{T}H_2^{T}H_3^{T}H_4^{T}P \tag{4}$$

Figure 3 shows the hardware required to mechanize the one-dimensional fast rationalized Haar transform in pipeline form.[2] It is presented here to illustrate how the fast algorithm actually works and to show that transform coefficients are available in real time for processing by the compression algorithm before the entire 16 point transform has been completely computed.

Operation of the pipeline transform unit is as follows. Input video data from the ADC enters the 1-D transform unit as shown in Figure 3. The first video sample is stored into latch L_1. The second video sample is presented to the B input of the first adder, Σ_1, whereupon the sum and difference of these first two samples are computed. The sum is temporarily stored in the second stage data latch, L_2, while the difference is outputted as transform coefficient #9 (see equation 3, definition of H_1). The 3rd video sample is stored into data latch, L_1, and the 4th input data sample is presented to the B input of the first adder, Σ_1, whereupon the sum and difference of these two samples are computed. The difference is outputted as transform coefficient #10 and the sum is fed to the B input of the second stage adder, Σ_2, whereupon the sum and difference of this term and the term in latch L_2, are computed. The (stage 2) difference is outputted as transform coefficient #5 and the sum is stored in the third stage latch, L_3. This process continues in a similar manner for each of the input video samples until all of the transform coefficients are computed.

The pipeline structure of Figure 3 is capable of operating at sample rates in excess of 15 MHz, which for word lengths of 8 bits, represents a data throughput rate of 120 Mbps. Northrop has constructed three real-time video processors using this pipeline architecture. The first real-time processor[3] incorporates an adaptive threshold compression algorithm. A second unit, built in the Spring of 1975, is an 8-bit arithmetic precision 1-D processor with an adaptive N-largest* compression system. This system will be demonstrated at this SPIE Technical Symposium. The third system, built in late 1975 and now undergoing debug, is a 2-D 8 bit Haar processor incorporating various signal processing algorithms.

The input data rate for one identified application of an RPV bandwidth reduction system,[1] is (8 bits/sample) (4.8 M samples per sec) = 38.4 Mbps. The average data rate is further reduced to 4.8 Mbps by 8:1 frame rate reduction, which is achieved by processing only 1/8 of each horizontal line per field in

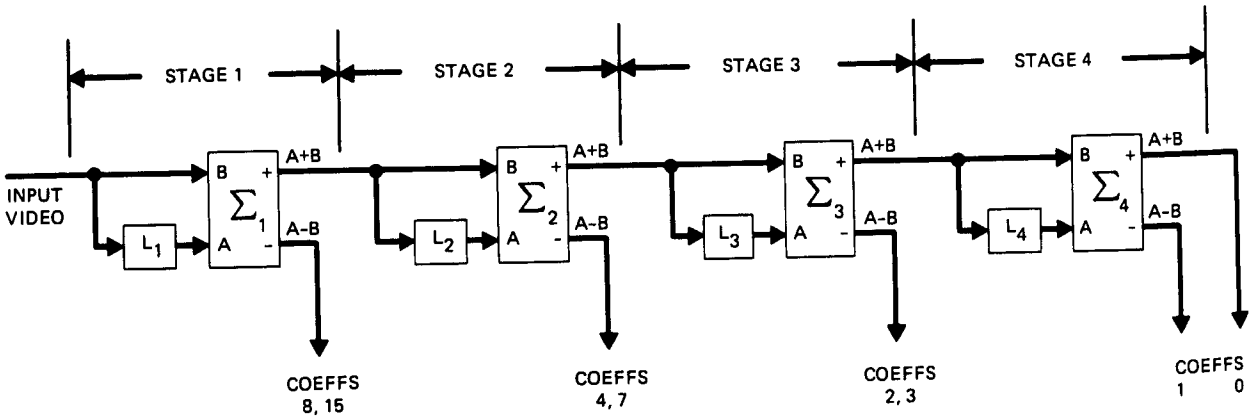
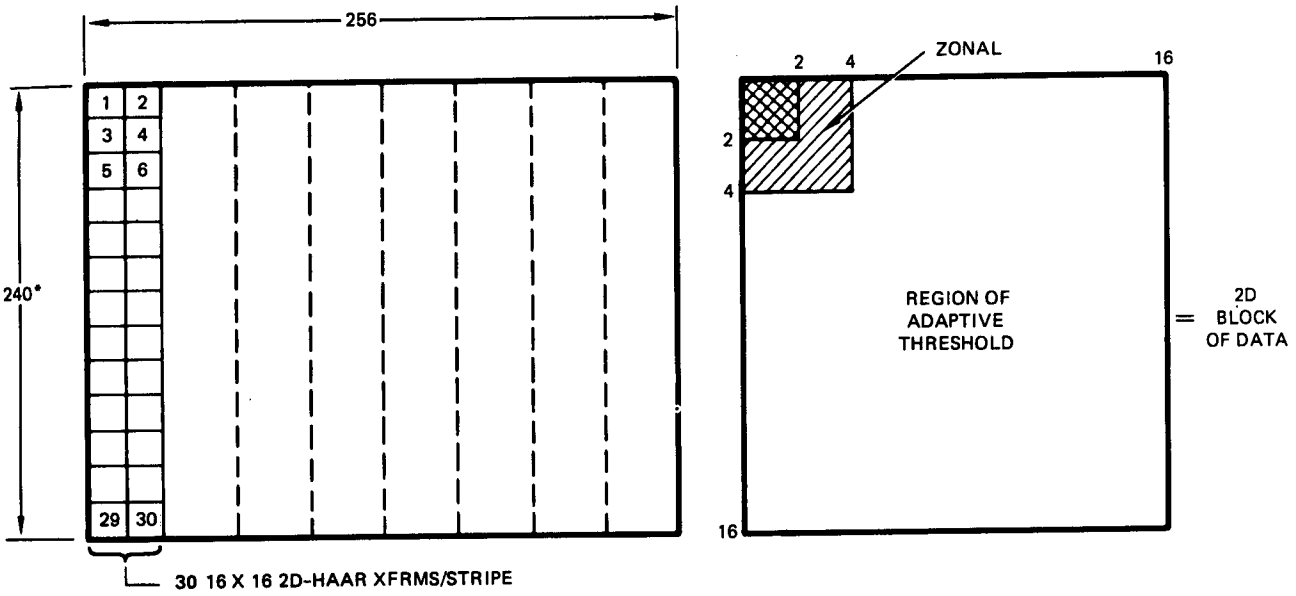


Figure 3. Block Diagram of Real-Time Fast Rationalized Haar Transform Unit (1-D).

* The largest "N" coefficients are selected for transmission where N, usually 1 or 2, is selected by front panel switches.

accordance with the concept developed by the researchers at NUC. The format used to obtain this frame rate reduction is shown in Figure 4. By reducing the data rate in this manner, the pipeline structure shown in Figure 3 can be telescoped into a more efficient architecture. This results in a considerable savings in hardware and power consumption. The new architecture is shown in Figure 5. This architecture is basically a highly specialized processor which consumes less than 5 watts of power and has been optimized to perform 2-D Haar transforms. Read-only memories (ROM's) control the operations of the adder/subtractor, data latches, shifters, and memory.

The 8:1 frame rate reduction requires 32 samples to be taken per horizontal line. Since Northrop uses a 16 point transform, two transforms per horizontal line must be computed as shown in Figure 4. The



*THE REMAINING HORIZONTAL LINES ARE TAKEN UP BY VERTICAL BLANKING

Figure 4. Scanning Format for Adaptive Compression.

processor of Figure 5 is programmed to take advantage of the fact that output transform coefficients are available in almost real time during the horizontal transformation process. This enables us to begin the computation of the vertical part of the 2-D transform before the horizontal transformation is completed. This feature of the hardware architecture allows us to reduce our storage requirements to 5 lines of 32 points each. Present methods of computing 2-D transformations of global transform (Walsh, Cosine, Fourier, etc.) require storage of 16 lines of data.

The details on how the vertical transforms are multiplexed between the horizontal transforms are relatively straightforward and will not be discussed here.

Hybrid Zonal/Adaptive Threshold Compression Unit

The compression unit is shown in Figure 6. The input video data from the Haar transform unit has been pre-whitened, digitized and transformed into coefficients in the Haar transform domain before entering the compression unit.

The compression unit consists of a cascade of operations. The transformed data is converted from 2's complement notation to magnitude and sign notation by an absolute value circuit. The converted data enters the threshold filter where only the largest coefficients are coded for transmission. The threshold level for a given stripe is the threshold value for the previous stripe plus a correction term which depends upon how full or empty the first-in-first-out (FIFO) buffer is. The correction term causes the total number of transmitted coefficients per stripe to be increased or decreased so that the FIFO will be (nearly) half full at the end of the next stripe. Provisions are provided to prevent buffer overflow or underflow by decreasing or increasing coefficients accordingly. The FIFO is large enough to store more than 1 1/4 full stripes at the highest data rate specified for the system (1.6 Mbps); therefore, buffer overflow will not be a problem.

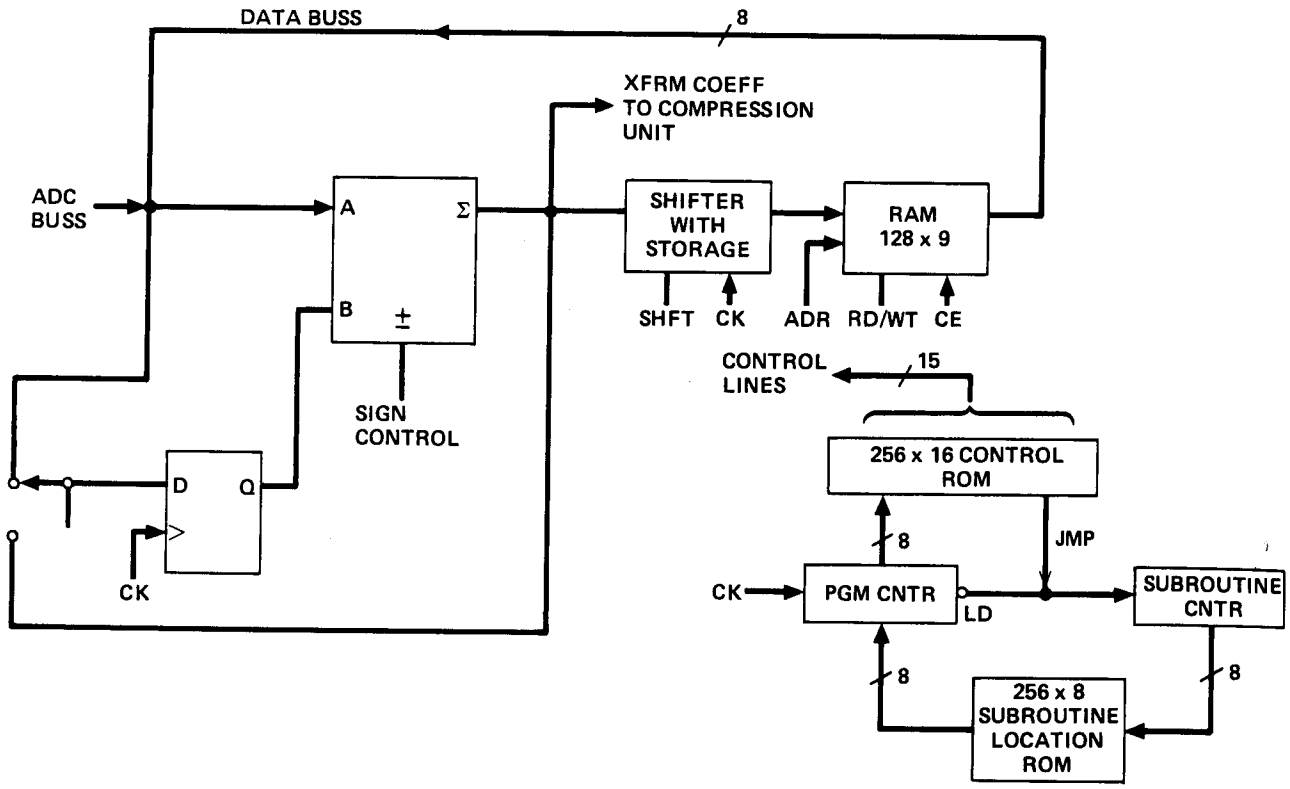


Figure 5. Block Diagram of 2-D Haar Transform Processor.

A fixed number of low frequency coefficients (zonal) in the upper left hand corner of the transform domain in Figure 4, are always sent and are not subject to thresholding. These zonal coefficients are almost always large and would almost always be selected for transmission by the adaptive threshold algorithm. It is therefore more efficient to encode these coefficients as a block rather than spend costly addressing bits to individually code them. The zonal coefficients are not companded. They are stored in order of arrival directly into the zonal burst buffer where they are accumulated for transmission as a continuous data block.

Transmission Format

The overall transmission format for the adaptive compression algorithms is shown in Figure 7. Two consecutive 16 point transforms are taken during each horizontal line of each stripe and 15 transforms are taken in the vertical direction for each stripe. There are 8 stripes per frame in accordance with the NUC temporal reduction concept. Figure 7 also shows that all the zonal coefficients for the entire stripe are sent as a fixed length contiguous block of data, followed by variable length data blocks composed of variable coefficients. Since most of the picture energy is contained in the low frequency zonal coefficients, we must make certain that channel errors do not cause them to be interpreted as variable coefficients.

To achieve this, the zonal coefficients are sent as a fixed block of ordered data to make them less sensitive to channel errors. The zonal coefficients are represented by a fixed number of bits and their relative position is always known with respect to frame and field sync. Channel errors can therefore only affect their amplitude, and not their location within the transform block. Because these low frequency terms are extremely important to the subjective quality of a video image, a parity bit is added to each zonal term. The format developed for error protection of the zonal coefficients is shown in Figure 8. The parity bit should be computed only over the 4 or 5 most significant bits, since errors in the lower bits are much less critical to the output picture quality. By including the parity bit, the majority of the transmission errors in the zonal coefficients can be detected. Although sufficient information is not available to correct detected errors in the zonal terms, errors can be compensated by replacement with the appropriate (possibly motion compensated) zonal coefficients from the previous frame. Error compensation for the zonal coefficient takes place in the ground terminal.

Non-zonal adaptive coefficients are transmitted in the format shown in Figure 9. Each companded coefficient is sandwiched between its 8 bit location address. This technique is used so the end-of-block sync word cannot be easily confused with the variable coefficient word. Note that each coefficient is absolutely addressed and each coefficient is represented by exactly 13 bits, including address, so by counting bits, address and amplitude information cannot be confused even when channel errors are present. The variable coefficients are transmitted in the order that the transform unit computes them so it is possible to

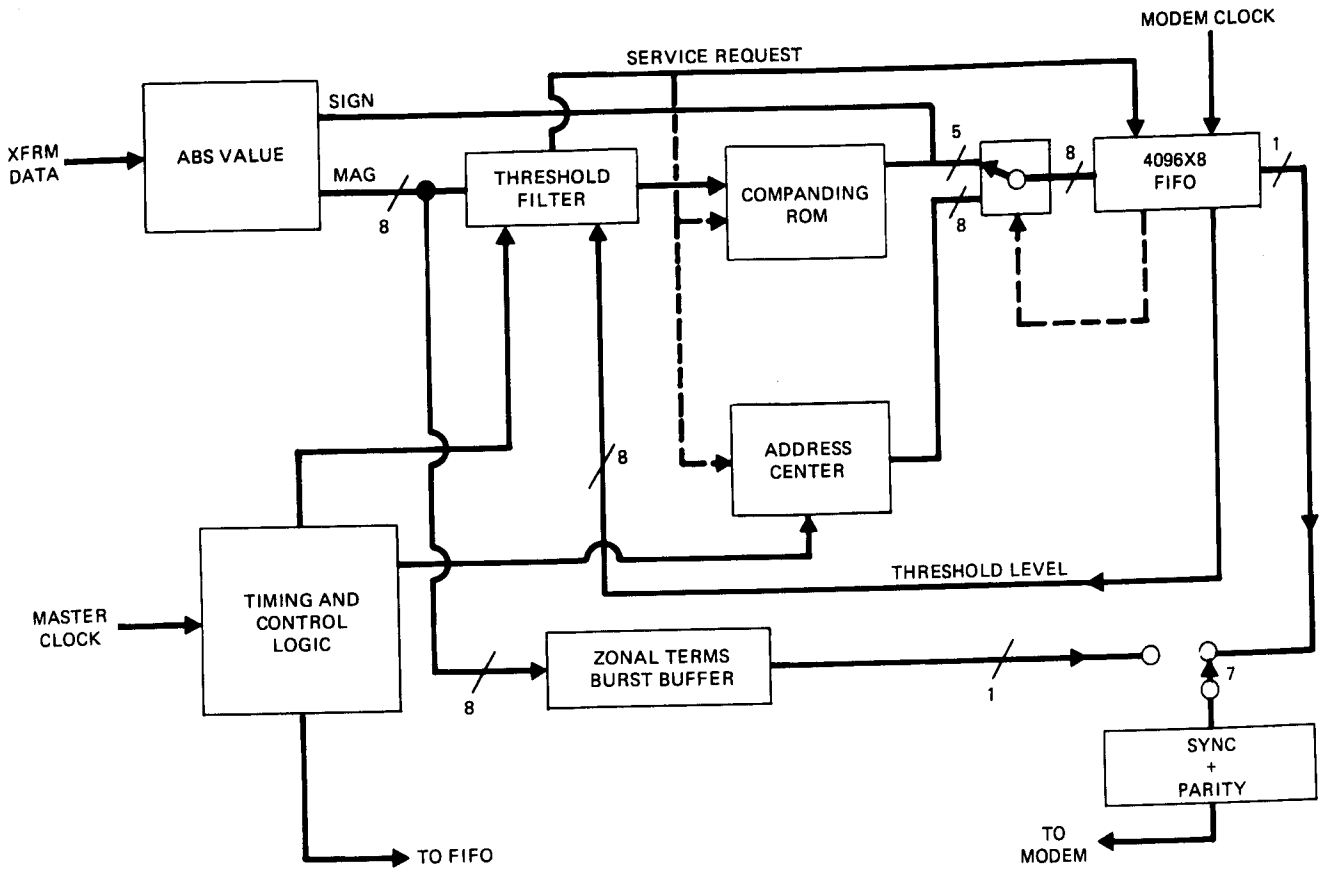


Figure 6. Block Diagram of Video Data Compression Unit.

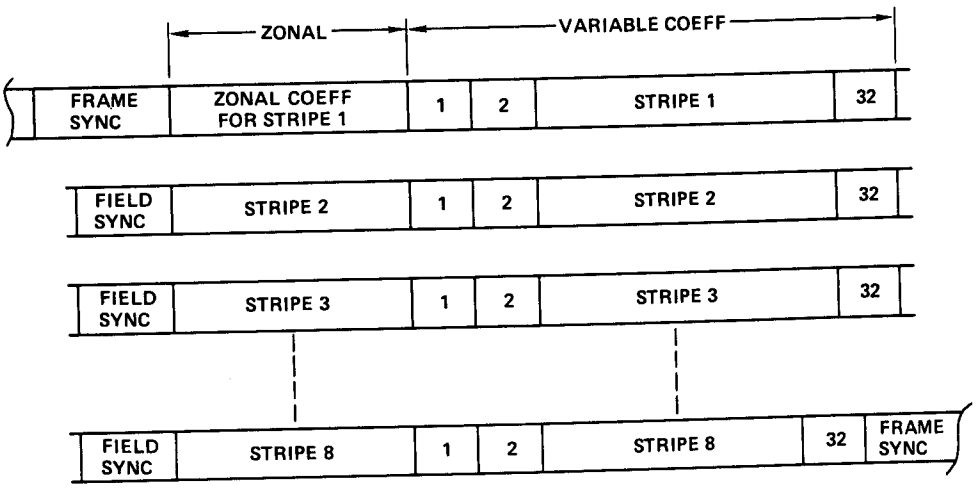


Figure 7. Transmission Format for Adaptive Compression Unit.

monitor the address of the coefficients being received on the ground to detect transmission errors in the address structure. Whenever errors are detected in the coefficients or address structure, these coefficients can be set to zero to minimize their effect on the reconstructed spatial picture. The presence of the onboard transmission buffer gives the Northrop system the opportunity to minimize the effects of burst errors on the received data by scrambling the order of the transmitted bits., i.e., spread out adjacent bits in time so that burst errors will not completely destroy several consecutive blocks of data.

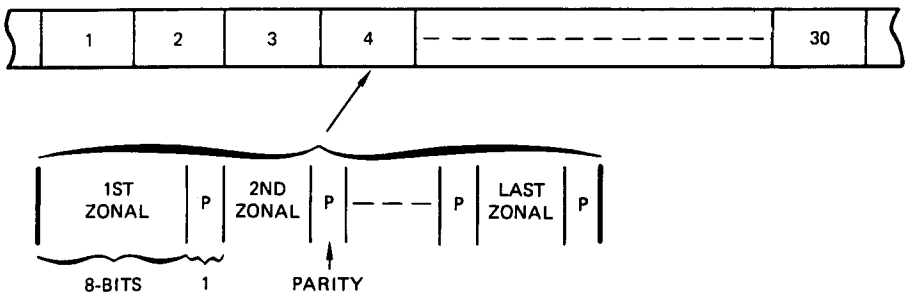


Figure 8. Format Detail for Zonal Coefficients.

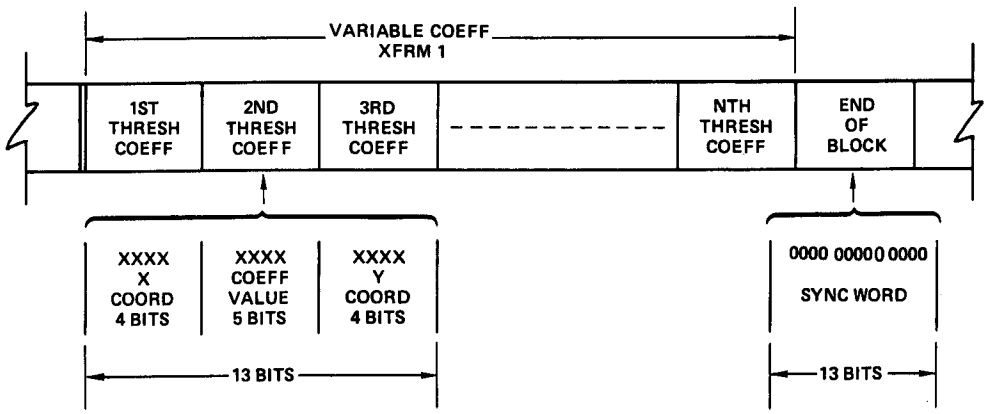


Figure 9. Format Details for Variable Coefficients.

Summary of Airborne Subsystem

The described 2-D Haar transform video bandwidth reduction system is designed to meet the size, weight and power constraints typical of real world RPV video communications links. This system also provides some inherent flexibility. For example, the frame rate can easily be reduced from the 7.5 frames/sec of the present design to 3.75 frames/sec by merely changing a few ROM's in the airborne unit so only one 16 point transform per line is computed instead of two per line. The frame rate can also be increased from 7.5 frames/sec to 12 frames/sec by taking three 16 point transforms per line instead of two per line. This modification of the transmission format can again be accomplished by changing a few ROM's in the airborne unit. In this manner it is possible to obtain almost any frame rate desired.

In designing the Northrop bandwidth reduction system, it was assumed that the input video has a continuous raster scan format. Though the basic Northrop video processor concept does not require that the input data have this format, the current design of the compression unit was influenced by the raster scan input. Northrop is aware that solid state sensors with snapshot capability are being developed and in fact has an IR&D program which has as its objectives the design and fabrication of visible and infrared CID and CCD sensor arrays with random access and non-destructive readout capability. The use of a snapshot mode sensor would eliminate the temporal skewing inherent in the stripe per field format.

Although the present design can operate with a raster scanned solid-state sensor array in a snapshot mode, when suitable random access sensors arrays become available, allowing greater flexibility in address techniques, the present bandwidth reduction system can be reconfigured to significantly reduce the size, weight and power consumption of the hardware.

The Northrop airborne subsystem provides significant flexibility to change data rates (compression ratios) on command from the ground terminal. In response to a (ground) command for a particular clock rate, the threshold in the compression unit will automatically increase or decrease as a function of how fast the transmission buffer empties. Thus, the adaptive compression scheme used in the airborne subsystem will respond to clock rates of almost any frequency by adaptively and automatically changing the

threshold to achieve the required transmission rate.

Ground Subsystem

A detailed block diagram for the ground terminal is shown in Figure 10. The ground terminal consists of an input receiver data buffer, error compensation circuitry and an inverse compressor which feeds the inverse 2-D Haar transform unit. The 2-D inverse transform unit converts transform domain coefficients back into spatial domain video samples which are stored in the Frame Store unit to provide flicker-free display of the temporally compressed video. The output of the Frame Store is fed through a digital-to-analog converter (DAC) and through the de-emphasis filter to the TV display.

The following sections give a more detailed description of the operation of each of the major functional blocks in the ground subsystem.

Receiver Data Buffer, Error Compensation and Inverse Compression Unit

This unit serves to accumulate data from the receiver modem into 16 x 16 transform coefficient blocks suitable for inverse 2-D transformation. Provisions are included to error-compensate the zonal coefficients by replacing any coefficient having a detected parity error with the corresponding coefficient from the previous frame. This requires storing the zonal coefficients from the previous frame. The required storage is less than 1000 eight bit words for a 2 x 2 block of zonal coefficients per transform. Variable coefficients are inverse companded and clocked into the variable coefficient burst buffer according to the address specified by that particular coefficient. Any coefficient not addressed is set to zero. When a complete 16 x 16 coefficient block has been accumulated, the coefficients are transferred to the 2-D transform unit for further processing.

Although it is possible to provide error compensation for the variable coefficients, we find that from a viewpoint of subjective picture quality, it is more important to compensate for errors in the zonal coefficients than for errors in the variable coefficients. This is especially true for global/local transforms, such as the Haar. Though it would ultimately be desirable to also include error compensation for the variable coefficients, this compensation scheme is still under development at Northrop and will not be discussed at this time.

Inverse 2-D Transform Unit

The hardware used to mechanize the 2-D inverse Haar transform is similar to that used in the forward transform. The only major difference between the hardware implementation of the airborne and ground subsystems is that the arithmetic precision in the ground subsystem will be increased from 8 bits to 16 bits to provide sufficient dynamic range to prevent channel errors from causing integer overflows. This provision will also give the flexibility needed for incorporating image enhancement and contour extraction functions, if so desired. Power consumption will be greater in the ground subsystem since higher speed devices will be required to compensate for the increased propagation delays associated with the 16 bit arithmetic. The mathematical basis for the design of the inverse transform is described in Equation 3. In simple terms, the inverse transform is obtained by running the forward transform algorithm backwards.

Although 16 bit arithmetic will be used to compute the inverse transform, the resulting spatial video will only have a precision of 8 bits. In this design, the 8 bits directly above the LSB are the bits of interest (the LSB is an arbitrary scale factor of 2), and bits greater than these should only occur due to channel errors. Any reconstructed brightness value greater than the allowed 8 bit dynamic range is clipped to full scale brightness. Similarly, any negative reconstructed picture brightness value will be clipped to zero, since the original input video is defined for positive brightness only.

Frame Store Unit

The transmission format for the video signal requires that only 1/8 of each field be transmitted to achieve an 8:1 frame rate reduction. A frame store is required as part of the ground terminal to assemble 8 successive fields to produce one frame which is then displayed on the TV monitor. This type of transmission and storage concept provides a flicker-free image to the viewer. The display is updated with new 1/8 frame stripes for each successive field of the TV camera. A semiconductor memory for picture storage is used in the ground subsystem. This memory is packaged onto 4 PC boards, 7 inches x 7.3 inches. Each PC card is a memory plane, organized as shown in Figure 11. Each memory plane reads or writes four 8-bit words simultaneously as controlled by a simple 4:1 multiplexing scheme. Data can be accessed or loaded either sequentially or randomly and individual memory refresh is incorporated into each plane.

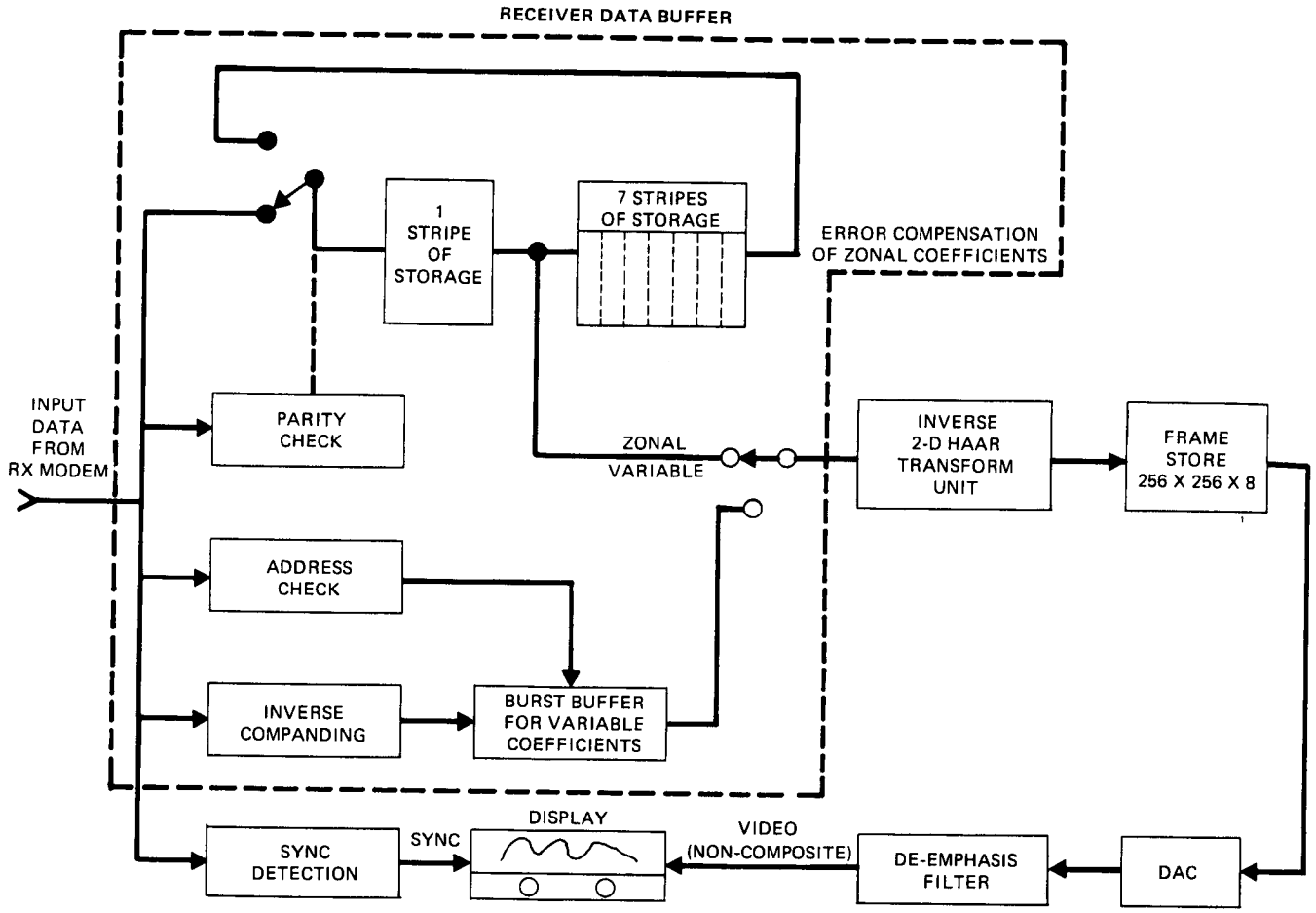


Figure 10. Block Diagram of Ground Terminal.

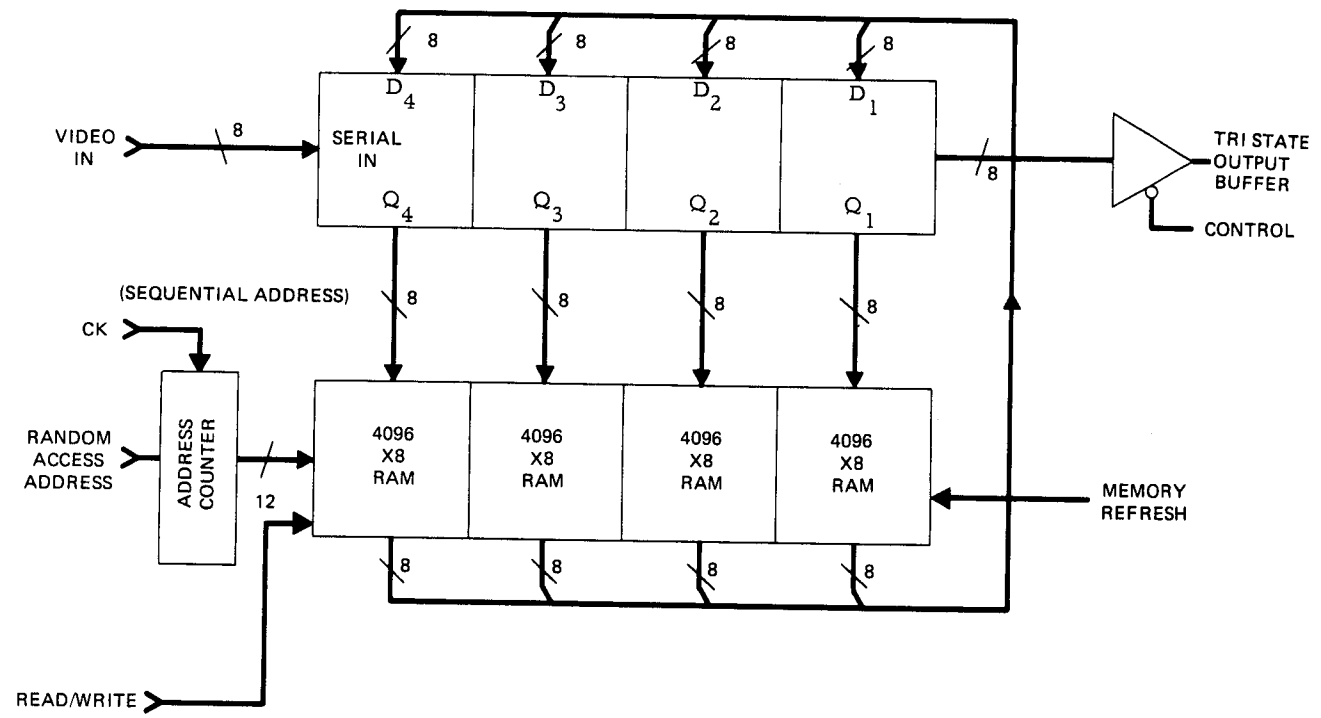


Figure 11. Organization of Memory Plane.

The output data from each memory plane drives a common tristate output buss so that the size of the memory can be easily expanded in increments of 16k, 8-bit words. The multiplexing scheme utilized in the design of the memory plane enables a frame store of size 256 pixel by 256 pixel by 8-bits/pixel to accept and deliver data at sample speeds in excess of 15 MHz.

DAC, De-emphasis, and Display

The DAC is a standard 10 bit 10 MHz commercial digital-to-analog converter. A 10 bit DAC is used because of its excellent linearity. This high quality DAC will provide an excellent quality 8-bit picture. The 2-LSB's are tied high to provide a slight black level which will help to produce a better quality image in the output display. The de-emphasis filter follows the DAC. Its function is to convert the frequency response of the reconstructed picture to the frequency response of the original picture. This filter will tend to remove some of the edge blockiness which tends to occur in highly compressed pictures (< 1 bit/pel). The bandwidth of the display should be greater than 10 MHz to ensure that the display does not degrade the high quality pictures obtained at the higher data rates.

Summary of Ground Hardware

The ground system contains a receiver data buffer unit, error compensation unit, inverse compression and Haar transform unit, frame store unit, DAC, de-emphasis circuitry and a high performance TV display. The basic differences between the airborne and ground subsystems are the error detection and compensation circuitry for the zonal terms which requires an additional storage of 1000 eight bit words and a frame store of 256^2 pels by 8 bits/pel to provide a flicker-free image on the TV display. The inverse Haar transform unit in the ground subsystem is implemented with 16 bit arithmetic to prevent integer overflows due to channel errors. A 10 bit 10 MHz DAC is used to provide the linearity required to obtain excellent quality 8 bit pictures.

Performance

The Northrop 2-D Haar video bandwidth reduction system has been designed to meet all of the performance, packaging and environmental specifications for some particular RPV applications.

As currently designed, the airborne subsystem will provide an 8 bit quality picture at the output of the 2-D Haar transform unit. The theoretical peak-to-peak signal to average RMS truncation error of the 2-D transform is 61.2 db and the average expected truncation error per transform coefficient is 0.19 quantization levels. This corresponds to an average roundoff error for the 2-D transform unit of 0.07% referenced to full scale signal, 2^8.

Figure 12 shows the mean squared error (MSE) performance of a computer simulation of the Haar Transform adaptive threshold compression system where MSE is defined by equation 5. Figure 13 shows some of the test pictures used in the simulation. It is Northrop's policy to always include the original picture whenever prints are made to minimize the effects of the developing process when a picture processing algorithm is being evaluated.

$$\text{MSE} = \frac{\sum_{X=0}^{N-1}\sum_{Y=0}^{N-1}\left[f(X,Y) - \hat{f}(X,Y)\right]^2}{\sum_{X=0}^{N-1}\sum_{Y=0}^{N-1}\left[f(X,Y)\right]^2} \cdot 100\% \qquad (5)$$

where

$f(X,Y)$ = original picture values

$\hat{f}(X,Y)$ = compressed picture values

N = number of horizontal and vertical samples.

Summary

A practical bandwidth compression system for RPV video links has been described. The system design is based on the fast 2-D Haar transform and an adaptive threshold compression technique to achieve a high level of system performance within the constraints imposed by current RPV requirements. A patent covering this system has been applied for.

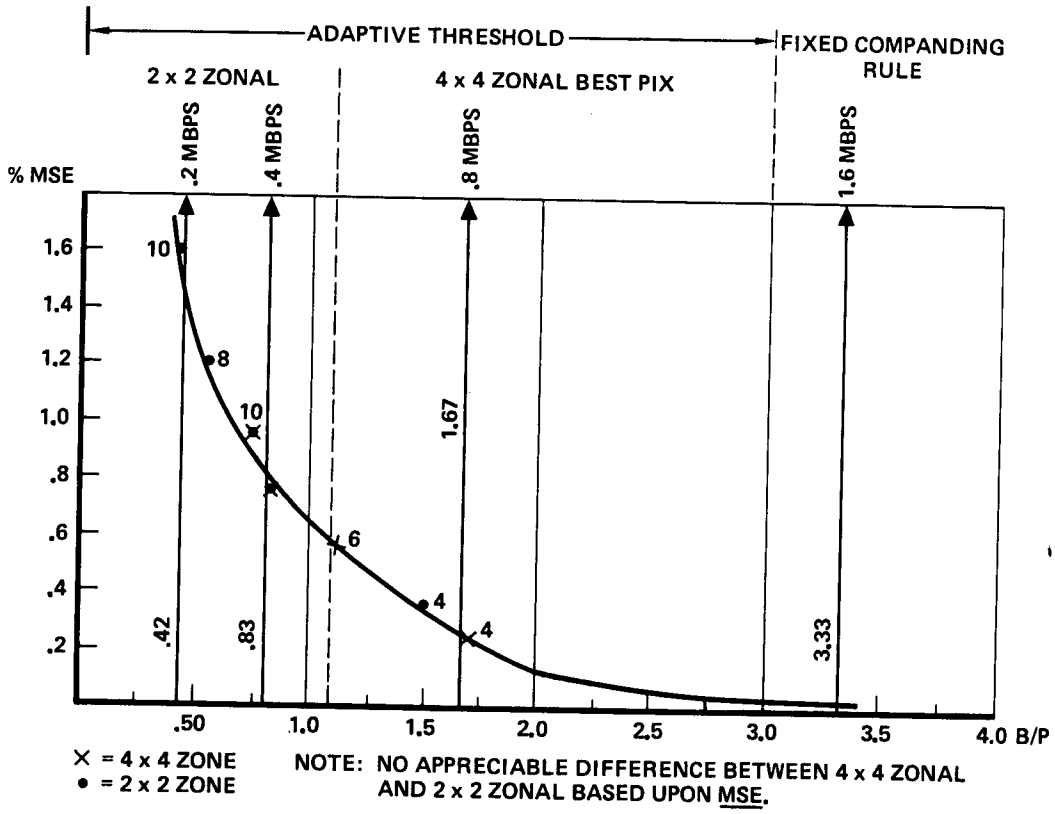


Figure 12. Mean Squared Error Performance of Adaptive Threshold Haar Compression System.

Acknowledgements

The authors wish to thank Mr. D. E. Miller of the Northrop Corporation for his assistance in preparing the demonstration of the 1-D real time Haar transform processor. We also wish to thank Mr. Donald J. Lynn and Mr. David Haas of the Space Image Processing Group at the Jet Propulsion Laboratory for their expert assistance in printing the pictures presented in this paper. The work reported in this paper was performed entirely under an on-going Northrop Corporation independent research and development (IR&D) program.

References

1. Naval Undersea Center (NUC) Solicitation No. N00123-76-R-0746, "Bandwidth Reduction System," February 1976.

2. J. J. Reis, R. T. Lynch, "Class of Transform Digital Processors for Compression of Multi-dimensional Data," Patent Application Serial No. 612,090.

3. R. T. Lynch, J. J. Reis, "Transformation of Television Picture, "Applications of Walsh Functions and Sequency Theory, H. Schieber, F. Sandy, IEEE New York (74CHD861-5 EMC) pp 140, 146, 1974.

4. R. W. Means, J. M. Speiser, H. J. Whitehouse, "Image Transmission via Spread Spectrum Techniques," ARPA Quarterly Technical Report ARPA QR3, Oct. 1, 1973 - Jan. 1, 1974, p 2, AD 780805.

Compressed (1.5 B/P) Original (8 B/P) Compressed (1.16 B/P) Original (8 B/P)

Compressed (1.7 B/P) Original (8 B/P) Compressed (1.1 B/P) Original (8 B/P)

Figure 13. Adaptive Threshold Compression of 2-D Haar Transform Pictures.

Relations Between Haar and Walsh/Hadamard Transforms

B. J. Fino

Abstract—Relations between the Haar and Walsh/Hadamard (W/H) transforms, which are proved, show that for some applications the Haar transform performs as well as, and faster than, the W/H transform. These relations yield a family of orthogonal transforms including the Haar and W/H transforms with a common fast algorithm.

Introduction

The Haar and Walsh/Hadamard (W/H) functions are formally defined by Alexits[1] and Fig. 1 shows the eight Haar and eight W/H functions of order 8. The W/H functions are ordered by their sequencies (number of sign changes). The square orthogonal matrices $[H_{2^n}]$ and $[W_{2^n}]$ of order 2^n, the rows of which are the 2^n Haar and 2^n W/H functions normalized by $1/\sqrt{2^n}$, are called the Haar and W/H matrices.[2]

In the following section, we wish to show that simple relations exist between Haar and W/H submatrices, which are obtained by partitioning the matrices $[H_{2^n}]$ and $[W_{2^n}]$, respectively, into $(n+1)$ rectangular matrices $[MH_{2^n{}^k}]$ and $[MW_{2^n{}^k}]$ of dimension $(2^n \times 2^{k-1})$. $[MH_{2^n{}^0}]$ is the first row H_0, $[MH2^{n^k}]$ is formed from the Haar functions of rank r, $2^{k-1} \leq r < 2^k$, for $k = 1, \cdots, n$, and similarly for the W/H submatrices $[MW_{2^n{}^k}]$. The submatrices of order 8 are shown in Fig. 1. We define the matrix operators $\mathcal{H}$ and $\mathcal{W}$ which express simply the recursive generations of the Haar and W/H submatrices. Then, using some properties of these operators, we prove by induction on n simple relations between the Haar and W/H submatrices.

Basic Relations

Alexits[1] has suggested that a matrix relation exists between the submatrices $[MH_{2^n{}^k}]$ and $[MW_{2^n{}^k}]$. We now prove that this relation is

$$[MW_{2^n{}^k}] = [S_{2^{k-1}}] \cdot [W_{2^{k-1}}] \cdot [MH_{2^n{}^k}], \quad k = 1, \cdots, n \tag{1}$$

in which $[W_{2^{k-1}}]$ is the ordered W/H matrix of order 2^{k-1}, and $[S_{2^{k-1}}]$ is the following permutation matrix of order 2^{k-1}:

$$[S_{2^{k-1}}] = \begin{bmatrix} 0 & & & 1 \\ & & \cdot & \\ & \cdot & & \\ 1 & & & 0 \end{bmatrix}.$$

If (1) holds, as $[W_{2^{k-1}}]$ and $[S_{2^{k-1}}]$ are symmetric and orthogonal, we have the converse relation:

$$[MH_{2^n{}^k}] = [W_{2^{k-1}}] \cdot [S_{2^{k-1}}] \cdot [MW_{2^n{}^k}]. \tag{2}$$

Recursive Definitions of the Haar and W/H Functions

We first define the matrix operators $\mathfrak{IC}$, $\mathfrak{W}$, and $\mathfrak{W}'$ which, applied to a $(m \times p)$ matrix $[M]$, generate the following $(2m \times 2p)$ matrices.

Operator $\mathfrak{IC}$:

$$[\mathfrak{IC}([M])] = [M] \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \left[\begin{array}{c|c} [M] & [0] \\ \hline [0] & [M] \end{array}\right]$$

where $\otimes$ denotes the Kronecker product.

Operators $\mathfrak{W}$ and $\mathfrak{W}'$: each row $(a_1, a_2, \cdots, a_m)$ of $[M]$ is replaced by two rows

$$\left|\begin{matrix} \mathfrak{L} \\ \mathfrak{L}' \end{matrix}\right. \text{for } \mathfrak{W} \quad \text{and} \quad \left|\begin{matrix} \mathfrak{L}' \\ \mathfrak{L} \end{matrix}\right. \text{for } \mathfrak{W}',$$

in which

$$\mathfrak{L} = (a_1, a_2, \cdots, a_m, \operatorname{sgn}(a_1 \times a_m) \times (a_1, a_2, \cdots, a_m))$$

and

$$\mathfrak{L}' = (a_1, a_2, \cdots, a_m, \operatorname{sgn}(-a_1 \times a_m) \times (a_1, a_2, \cdots, a_m)).$$

Using these operators it can be shown that:

$$[MH_{2^n{}^k}] = [\mathfrak{IC}([MH_{2^{n-1}{}^{k-1}}])], \quad k = 2, \cdots, n \tag{3}$$

$$[MW_{2^n{}^k}] = 1/\sqrt{2}[\mathfrak{W}([MW_{2^{n-1}{}^{k-1}}])], \quad k = 1, \cdots, n \tag{4}$$

and globally for the W/H matrices:

$$[W_{2^n}] = 1/\sqrt{2}[\mathfrak{W}([W_{2^{n-1}}])]. \tag{5}$$

These relations are alternative recursive definitions, given the matrices of order 2, for the Haar and W/H matrices and, from them, for the Haar and W/H functions. They are similar to the difference equations introduced by Harmuth.[3]

The following properties of the operators $\mathfrak{IC}$, $\mathfrak{W}$, and $\mathfrak{W}'$ will be used:

$$[S_{2m}] \cdot [\mathfrak{W}([M])] = [\mathfrak{W}'([S_m] \cdot [M])] \tag{6}$$

and

$$[S_{2m}] \cdot [\mathfrak{W}'([M])] = [\mathfrak{W}([S_m] \cdot [M])]. \tag{7}$$

Let $[M] = \{m_{ij}\}$ be an $(m \times p)$ matrix, $[N]$ a $(p \times q)$ matrix, and $[P] = (M)$. $[N] = \{p_{ij}\}$ of dimension $(m \times q)$. We say that the product matrix $[P]$ is sign invariant (contravariant) if for all i $(p_{i1} \times p_{iq})$ has same (opposite) sign as $(m_{i1} \times m_{ip})$. Then, if $[P]$ is sign invariant,

$$[\mathfrak{W}([M])] \cdot [\mathfrak{IC}([N])] = [\mathfrak{W}([P])]. \tag{8}$$

If $[P]$ is sign contravariant,

$$[\mathfrak{W}([M])] \cdot [\mathfrak{IC}([N])] = [\mathfrak{W}'([P])]. \tag{9}$$

[1] G. Alexits, *Convergence Problems of Orthogonal Series*. New York: Pergamon, 1961, pp. 46-62.
[2] H. C. Andrews, *Computer Techniques in Image Processing*. New York: Academic Press, 1970, pp. 73-90.
[3] H. F. Harmuth, *Transmission of Information by Orthogonal Functions*. New York: Springer, 1970, pp. 19-48.

Reprinted with permission from pages 844-848 of 3rd International Conference of Pattern Recognition Proceedings, Institute of Electrical and Electronics Engineers, New York, 1976

USE OF THE HAAR TRANSFORM AND SOME OF ITS PROPERTIES IN CHARACTER RECOGNITION

S. Wendling, G. Gagneux, and G. Stamon

Summary

Haar transform is defined and several of its properties are developed. A fast computing algorithm for Haar coefficients is given.

Haar power spectrum is introduced and it is shown that the obtained results can be generalized to any transform defined by an orthogonal matrix whose first row vector contains ones only. A character recognition experiment using both invariants within the power spectrum and features within the transform domain is described.

Introduction

Properties of global transforms such as Fourier transform or Hadamard transform and, for each of these a fast computation algorithm, account for the widespread use of these in picture (transmission, recognition,...) or speech processing.

This paper will introduce Haar transform which has over the former two (which are globally sensitive) the advantage of being both locally and globally sensitive. Once the definition of the Haar transform recalled, some of its properties will be established and an algorithm for fast and efficient computation of Haar coefficients developed. Then the Haar transform efficiency will be tested in a character recognition experiment.

Haar functions

Let E be the vector space of the numerical functions, defined within $[0,1]$, bounded, accepting on any point a limit on the left and on the right, and continuous to the right. With the usual inner product of $L^2([0,1])$, E is a pre-Hilbert space in which it is possible to define a complete orthonormal system (f_n), called Haar system, and defined in the following way :

1) $f_0(x) = 1 \quad \forall x \in [0,1]$

2) $\forall n > 0$, integer, let m be the greatest integer such as $2^m \leq n$ and $n = 2^m + k$

Then $f_n(x) = 2^{m/2}$ for $\frac{2k}{2^{m+1}} \leq x < \frac{2k+1}{2^{m+1}}$

$I_j = \left[\frac{j}{2^p} , \frac{j+1}{2^p}\right[$ with $j \in [0,1,2,\ldots,2^p-1]$

the f_k $(0 \leq k \leq 2^p-1)$ functions are constant.

$H(2^p)$, the $2^p \times 2^p$ matrix, called Haar matrix, is then obtained in the following way :

$H(2^p) = [h_{ij}]$ where h_{ij}, the generic term, located at the intersection of the i^{th} row and the j^{th} column, equals $f_i(I_j)$ (i.e. the value taken by f_i on the I_j interval ; see figure 1b).

The row vectors of $H(2^p)$ then establish an orthogonal system of $\mathbb{R}^{2^p}$, the orthogonality with the first row vector implying that

$$\sum_{j=0}^{j=2^p-1} h_{ij} = 0 \text{ for } i = 1,2,\ldots,2^p-1 \qquad (1)$$

Moreover, $H(2^p)$ is such as $H(2^p)\, H(2^p)^t = 2^p I_{2^p}$ where I_{2^p} is the $2^p \times 2^p$ identity matrix.

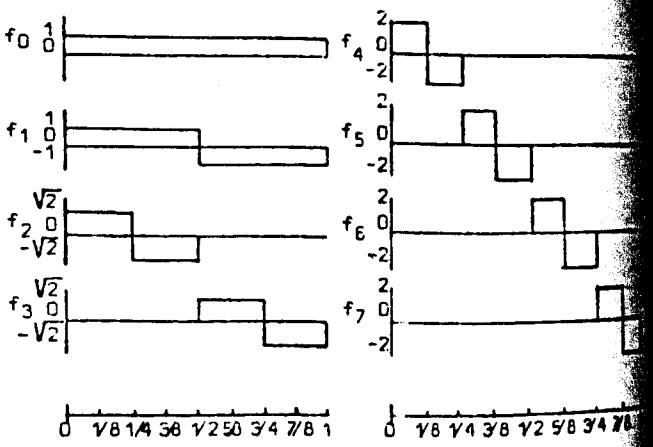


fig. 1a. Haar functions for n = 0,1,..... 7.

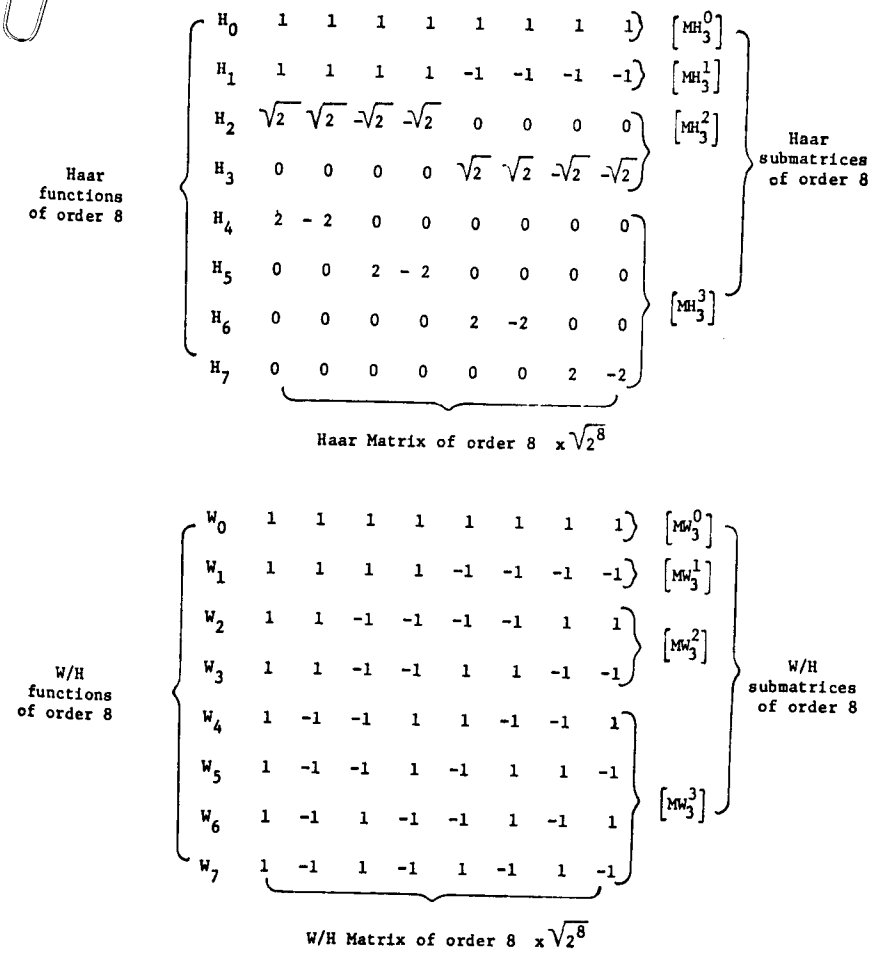


Fig. 1. Haar and W/H functions, matrices, and submatrices.

Proof of (1)

Assuming that (1) is true at the order $(n-1)$, we have:

$$[MW_{2^{n-1}}^{k}] = [S_{2^k}]\cdot[W_{2^{k-1}}] \cdot [MH_{2^{n-1}}^{k}], \quad k=1,\cdots,(n-1). \quad (10)$$

Applying $\mathcal{W}$ to both sides of (10), and using (4) for the left hand side and (7) for the other side we obtain:

$$\sqrt{2}[MW_{2^n}^{k+1}] = [S_{2^k}]\cdot[\mathcal{W}'([W_{2^{k-1}}] \cdot [MH_{2^{n-1}}^{k}])].$$

We can verify that the product of a matrix by a Haar submatrix $[MH_{2^n}^{k}]$ is sign contravariant; so, using (9), (3), and (5), we obtain (1) for $k=2,\cdots,n$. As (1) is obvious for $k=1$, this completes the proof of (1), and consequently (2).

Application to Transform Elements

The matrix relations (1) and (2) imply relations between the transform vectors $V_H(V_{H_0}, V_{H_1},\cdots, V_{H_{2^n-1}})$ and $V_W(V_{W_0}, V_{W_1},\cdots, V_{W_{2^n-1}})$ of a vector V. If we right-multiply (1) by V, we obtain:

$$\begin{pmatrix} V_{W_{2^k-1}} \\ \vdots \\ V_{W_{2^{k-1}}} \end{pmatrix} = [S_{2^k-1}]\cdot[W_{2^k-1}]\cdot\begin{pmatrix} V_{H_{2^k-1}} \\ \vdots \\ V_{H_{2^{k-1}}} \end{pmatrix}$$

$$k = 1,\cdots,n. \quad (11)$$

The converse relation is similarly obtained from (2).

Let a "zone" be the set of coefficients of the transform vectors which appear in these relations. We see that a zone in a transform vector determines the corresponding zone in the other transform vector. This property shows that, if we approximate V by the same subset of zones of the transform vectors V_H and V_W or, in particular, if we truncate these vectors at the end of a zone, we obtain the same approximate vector after inverse transforms.

We see that corresponding zones in (11) are related by an orthogonal transform. It follows by Parseval's theorem that the energies in corresponding zones of the transform vectors are identical.

The computations of the Haar and W/H transforms of a vector of order 2^n require, respectively, $2(2^n-1)$ and $n2^n$ elementary operations with a fast algorithm.[2] Thus when relations (1) and (2) can be used, the Haar transform performs as well as and faster than, the W/H transform (about 4 times faster for $2^n=256$). These results have been applied to image processing.[4]

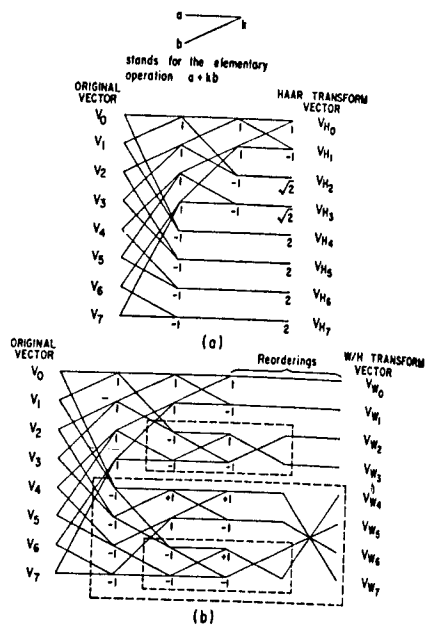


Fig. 2. (a) Haar transform fast algorithm. (b) Haar and W/H transforms and fast algorithm.

Application to Fast Algorithms

The diagram of a simple fast algorithm for the Haar transform of order 8 is shown in Fig. 2(a). By recursive use of relation (11), we derive from this diagram the diagram of a fast algorithm for the W/H transform of order 8 given in Fig. 2(b). The W/H transforms of lower orders are shown inside dotted lines and are followed by the reorderings of the matrices $[S]$.

Note that in the decomposition of a W/H transform into W/H transforms of lower orders, replacing some W/H transforms by corresponding Haar transforms yields a family of orthogonal transforms. Obviously the fast algorithm of Fig. 2(b) computes all the transform elements of all the transforms of this family including the Haar and W/H transforms. For each transform of this family the transform elements appear at appropriate nodes of Fig. 2(b).

Acknowledgment

The author wishes to thank Prof. V. R. Algazi for his advice and encouragement.

[4] B. J. Fino, "Etude expérimentale du codage d'images par transformations de Haar et de Hadamard complexe," *Ann. Télécommun.*, May–June 1972.

Haar transform

The Haar transform of an n periodic sequence, $X = \{x_0, x_1, \ldots, x_{n-1}\}$ is defined as

$$[T(X)] = 1/n\,[H][X] \qquad (2)$$

where $[X]$ is the vector representation of sequence X, $[H]$ the n x n Haar matrix and

$$[T(X)]^t = \{t_0(X), t_1(X), \ldots, t_{n-1}(X)\}$$

the coefficients of the Haar transform.

As $[H][H]^t = nI$, the inverse transform will be defined as $[X] = [H]^t\,[T(X)]$ (3)

Fast Haar transform

Using matrix factoring or matrix partitioning similar to those used for FFT (Fast Fourier Transform) and FHT (Fast Hadamard Transform) computation, fast and efficient algorithms for the Haar transform are obtained.

Given the particular structure of the $n \times n = 2^p \times 2^p$ Haar matrices (fig. 1b for $n = 2^3$), the total number of iterations needed to compute the transform coefficients is P. Each iteration reduces by half the number of values still to be computed. Thus, the first iteration gives the n/2 coefficients:

$$t_k(X) \qquad n/2 \leqslant k < n$$

the second iteration gives the n/4 coefficients

$$t_k(X) \quad ; \quad n/4 \leqslant k < n/2$$

the P^{th} iteration gives the 2 coefficients :

$$t_0(X) \text{ and } t_1(X)$$

For example, with n = 8, the signal flow graph of fig. 2 is obtained. Apart from the multipliers, the total number of arithmetic operations (additions or subtractions) required for computing the Haar coefficients is 8 + 4 + 2 = 14.

Generalization

The signal flow graph in fig. 2 can be straightforwardly generalized for $n = 2^P$. The following remarks can be made :

1. the total number of iterations is given by $P = \log_2 n$
2. the 1st iteration requires n arithmetic operations (additions or subtractions). At each iteration this number is divided by 2, which yields the total number of $n + n/2 + n/4 + \ldots + 2 = 2(n-1)$ operations. In the same way, the number of multiplications can be evaluated as $n/2 + n/4 + \ldots + 2 = (n-1)$. A single evaluation of (2) would have required n^2 operations.
3. Compared to Hadamard transform which can be evaluated in $n \log_2 n$ additions or subtractions, Haar transform requires nearly $(\log_2 n)/2$ less operations. As an example, with $n = 2^{10} = 1024$, FHT will need 10240 additions or subtractions whereas Haar transform will need only 2046, that is to say about five times less.

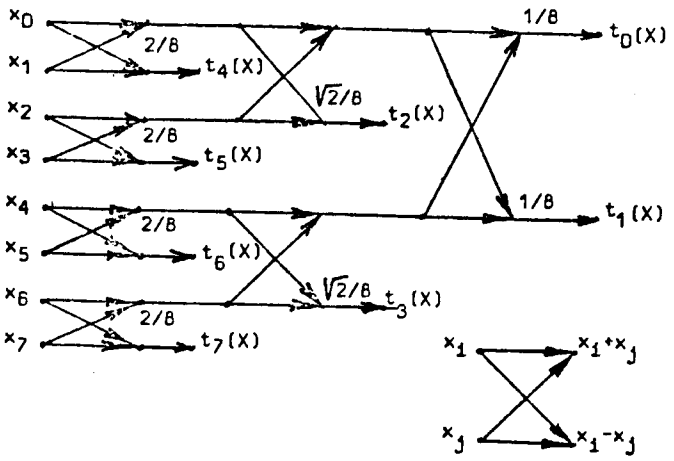


fig. 2. Signal flow graph illustrating the computation of the Haar transform coefficients for n = 8.

Parseval's formula (Power Spectrum)

The transpose of (2) yields :

$$[T(X)]^t = 1/n\,([H][X])^t = 1/n\,[X]^t[H]^t \qquad (4)$$

From (2) and (4) it follows that :

$$[T(X)]^t[T(X)] = 1/n^2\,[X]^t[H]^t[H][X]$$
$$[T(X)]^t[T(X)] = 1/n\,[X]^t[X]$$

or

$$\sum_{i=0}^{n-1} t_i^2(X) = 1/n \sum_{i=0}^{n-1} x_i^2 \qquad (5)$$

Thus, if $X = \{x_0, x_1, \ldots, x_{n-1}\}$ represents the sampled values of a signal, (5) shows that the power of this signal is conserved within the transform domain. The set $\{t_i^2(X)\}$, however, does not represent the individual spectral points as it is not invariant to the shift of the sampled data.

Spectrum development

Now a set of invariants is being looked for. The case where n = 8 is first considered.

Let X^l denote X shifted to the left by l positions.

$$X^l = \{x_l, x_{l+1}, \ldots, x_{l-2}, x_{l-1}\} \quad l = 1,2,\ldots,7$$

The Haar transform of X^l is :

$$[T(X^l)] = 1/8\,[H][X^l]$$
$$= 1/8\,[H][S^l][X] \quad \text{where } [S^l] \text{ is the l positions to the left translation matrix } (1 \leqslant l \leqslant 7)$$
$$= 1/64\,[H][S^l][H]^t[H][X]$$
$$[T(X^l)] = 1/8\,[A^l][T(X)] \quad \text{where } [A^l] = 1/8\,[H][S^l][H]^t$$

Now $[A^l]$ will always be orthogonal since

$$[A^l]^t[A^l] = 1/64\,[H^t]^t[S^l]^t[H^t][H][S^l][H^t]$$
$$= I_8 \text{ for } [S^l][S^l]^t = I_8$$

Moreover $[A^l]$ can always be written

$$A^l = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \tilde{A}^l & \\ 0 & & & \end{bmatrix}$$

where $[\tilde{A}^l]$ is a 7X7 matrix such as $[\tilde{A}^l][\tilde{A}^l]^t = I_7$

Indeed : $a^1_{oo} = 1$ since $\forall l$,

$$a^1_{oo} = 1/8 \; [1111111] \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$a^1_{oj} = 0$ for $1 \leq j \leq 7$ since in the Haar matrix, the sum of the elements in each row is zero (except in the first one)

$a^1_{io} = 0$ for $1 \leq i \leq 7$ since the row vectors of the Haar matrix are orthogonal to vector $[1111111]$ (except the first row) which is invariant to shifts.

So, $[A^1]$ is necessarily orthogonal.

It follows that :

$$(t_o(X^1))^2 = (t_o(X))^2$$

and $$\sum_{i=1}^{7} (t_i(X^1))^2 = \sum_{i=1}^{7} (t_i(X))^2 \qquad (6)$$
$$l=0,1,\ldots,7$$

Results (6), obtained by considering the particular case where $n=8$, can be straight-forwardly generalized for $n=2^p \times 2^\beta$ Haar matrices.
The following equations are then obtained :

$$(t_o(X^1))^2 = (t_o(X))^2$$

and $$\sum_{i=1}^{n-1} (t_i(X^1))^2 = \sum_{i=1}^{n-1} (t_i(X))^2 \qquad (7)$$
$$l=0,1,\ldots,(n-1)$$

Generalization

The results in (5) and (7) obtained for Haar matrices can be generalized for other orthogonal matrices whose first row vector consists of ones only. Moreover for the transformations defined by such matrices, there always exists at least two invariants in the power spectrum.

Indeed, let $[M]$ be a $n \times n$ matrix, whose generic term is m_{ij}, and verifying :

$$m_{oj} = 1 \quad o \leq j \leq n-1 \qquad (I)$$

$$[M]^t [M] = nI_n \qquad (II)$$

This second hypothesis implies in particular that

$$\sum_{j=0}^{n-1} m_{ij} = 0 \quad 1 \leq i \leq n-1 \qquad (8)$$

Let the transform of a vector X by $[M]$ be defined as :

$$[T(X)] = 1/n \, [M][X] \qquad (9)$$

(II) then necessarily implies that :

$$[T(X)]^t [T(X)] = 1/n^2 \, [X]^t [M]^t [M][X]$$

$$[T(X)]^t [T(X)] = 1/n \, [X]^t [X]$$

or else

$$\sum_{i=0}^{n-1} (t_i(X))^2 = 1/n \sum_{i=0}^{n-1} x_i^2 \qquad (10)$$

Under the condition of a l positions left shift of X ($1 \leq l \leq n-1$), (8) entails that :

$[T(X^1)] = 1/n \, [M][S^l][X]$ where $[S^l]$ is the l positions to the left translation matrix

$$[T(X^1)] = 1/n^2 [M][S^l][M][M][X]$$

$[T(X^1)] = 1/n \, [A^l][T(X)]$ where $[A^l] = 1/n[M][S^l][M]$

Given (I) and (II), it is easy to show that the matrix $[A^l]$ will always be written :

$$A^l = 1/n \begin{bmatrix} n & o & \cdots & \cdots & o \\ o & & & & \\ \vdots & & \tilde{A}^l & & \\ o & & & & \end{bmatrix}$$

where $[\tilde{A}^l]$ is such as $[\tilde{A}^l][\tilde{A}^l]^t = I_{n-1}$

Hence the following conclusion :

Under the hypotheses (I) and (II) there will always exist two invariants at least, viz. :

$$(t_o(X^1))^2 = (t_o(X))^2$$

$$\sum_{i=1}^{n-1} (t_i(X^1))^2 = \sum_{i=1}^{n-1} (t_i(X))^2 \qquad (11)$$
$$1 \leq l \leq n-1$$

Particular cases illustrating property (11) are given by the Haar, Fourier or Hadamard transforms, with however, additional properties for the last two, which result into the breaking of $(\tilde{A}^l)$ into orthogonal "block sub-matrices". This accounts for the fact that more than two invariants are obtained ($n/2 + 1$ for Fourier and $P + 1$ for Hadamard when $n = 2^p$).

Application of Haar transform

Given, on the one hand, the velocity of Haar transform computing and, on the other hand, the two invariants obtained in the power spectrum (7), it may seem interesting to apply it to character recognition.

The characters used are those given by MUNSON'S data basis, which consists of 49 X 3 alphabets, each corresponding to the 46 non-blank FORTRAN characters. The alphabets have been hand-written by 49 authors, each of these having produced three alphabets (so 3 X 46 characters each). The obtained images are binary and held in a 24 X 24 matrix. To obtain a $2^p \times 2^p$ matrix, each image has been transcribed into a 32 X 32 matrix by bounding it with zeroes so the earlier described algorithm could be used.

As for the experiment itself, repeated independently for each author, it was carried out in the following way :

Considering that the three hand-written alphabets for a given author were representative of his hand-writing characteristic features for each of the 46 characters were selected.

For a given character, two types of features were extracted :

. features within the power spectrum (2 invariants)

.. features within the Haar transform.

These characteristics obtained, each of the 138 characters (3 X 46) for one author was tried to be recognized. For this purpose, in a first clustering stage, the two invariants in the power spectrum were used, then in a second stage the transform features were used to complete the recognition process.

Features within the power spectrum

Feature extraction

For each of the i samples ($1 \leq i \leq 3$) of a given character, the two invariants defined in (7), which will be called P^i_{W0} and P^i_{W1}, were computed and the maximal and minimal values of these P^i_{W0} and P^i_{W1} ($1 \leq i \leq 3$) retained as representative for this character.

For the k characters in the alphabet ($1 \leq k \leq 46$) two values were twice obtained :

$$PWMIN0(k) = \inf_i(P^i_{W0}(k)) \text{ and } PWMAX0(k) = \sup_i(P^i_{W0}(k))$$
$$PWMIN1(k) = \inf_i(P^i_{W1}(k)) \text{ and } PWMAX1(k) = \sup_i(P^i_{W1}(k)) \quad (12)$$
$$1 \leq k \leq 46$$

That is to say that for each of the 3 X 2 invariants $P^i_{W0}(k)$ and $P^i_{W1}(k)$, an interval which includes the i ($1 \leq i \leq 3$) samples of the character k ($1 \leq k \leq 46$) is defined.

Use of features

Every time a character is to be recognized, its invariants (LPW0 and LPW1) will be computed, which will allow, by comparing them to the references defined in (12), to sort out of all the possible characters those for wich :

$$LPW0 \notin [PWMIN0(k), PWMAX0(k)]$$
$$LPW1 \notin [PWMIN1(k), PWMAX1(k)] \quad 1 \leq k \leq 46$$

So, starting with 46 classes, this first clustering stage will considerably reduce the number of possibilities to affect the still unknown character to a class. Moreover, if the samples used to establish the references defined in (12) are representative of the author's hand-writing the character to be recognized will necessarily belong to one of the remaining classes.

Figure 3 shows the average percentage of effectation possibilities eliminated by this first processing (only the 10 most representative authors have been represented).

For the whole of the 49 authors, the average of eliminated possibilities is 51.53%, i.e. starting with 46 possible classes, only 22 are still to be considered.

Features within the Haar transform

Feature extraction

When Haar transform is applied to a given image, 32 X 32 = 1024 coefficients are obtained. Obviously, the same method as that used for the power spectrum (definition for each of the coefficients of an interval containing all the possible values of this coefficient) cannot be used since 1024 X 2 = 2048 values would have to be conserved for each character. At most it would be possible for a given character to keep only the p intervals which differ most from those chosen for the other characters. However, this method has not been applied for two reasons : first because the coefficients obtained within the transform domain are not invariant to a shift of the input data so the intervals would be too large and second because the selection of references similar to those defined in (12) requires a great memory which was not avalaible for our application. Hence the following solution :

Let $t^i_j(k)$ be the j^{th} Haar transform coefficient of the i^{th} sample ($1 \leq i \leq n$; here n = 3) of the k^{th} character ($1 \leq k \leq 46$).

The average value of

$$t_j(k) = 1/n \sum_{i=1}^{n} t^i_j(k) \quad \begin{matrix} 1 \leq j \leq 1024 \\ 1 \leq k \leq 46 \end{matrix} \text{ is computed.}$$

That is, for character k, the value of the j^{th} coefficient is chosen equal to the mean of the values it takes in the set of character k samples.

Concurrently, $m_j = 1/46 \sum_{k=1}^{46} t_j(k)$ i.e. the mean of all the $t_j(k)$ in the alphabet is computed.

This being done, $d_j(k) = (t_j(k) - m_j)^2$ $1 \leq j \leq 1024$ is computed for each character, and the p values $t_j(k)$ for which $d_j(k)$ is maximal are taken as characteristic features of character k. That means that, for a given character, the coefficients that deviate most from the average of the whole alphabet are considered as characterizing this character. Note however that this method is far from being the most efficient since the $t_j(k)$ are obtained by a mean which does not convey the variations between one sample and another of the same character. Nevertheless, the aim of this experiment was not to find relevant characteristics within the transform domain.

Use of the features

For each character X to be recognized, its Haar transform and its two invariants are computed. Coefficients t_j and PW0 and PW1 are obtained. Then, using only PW0 and PW1, as above indicated, the 1st sorting is done. Going on with the n remaining characters, the p characteristics retained for the transform itself are used, and the distance between character X and each of the n classes is evaluated :

$d_k(X) = \sum_{i=1}^{P} (t_1 - r_1(k))^2$ $1 \leq k \leq n$ where l is the index of the i^{th} characteristic coefficient of class k, and $r_1(k)$ is the characteristic value retained for this coefficient in class k.
Character X is then effected to the class for which $d_k(X)$ is minimal.

Figure 4 shows the recognition percentages obtained by taking 1,2,...,15 features in Haar transform.

When 15 features are considered, the recognition average for all the authors above mentioned amounts to 97.75%, that is to say that among the 138 (3X46) tested characters for an individual author, only 3 are badly classed. This, if the preceeding remark about the choice of features in Haar transform in taken into account, leads one to surmising that a complete recognition can be achieved when more relevant characteristics are chosen in the transform domain.

	NUMBER OF THE AUTHOR								
	4	10	12	14	15	18	21	24	35	44
% of elimination	65.12	39.04	62.38	65.97	59.74	36.01	60.49	60.30	59.74	63.04

fig. 3 Average percentage of eliminated possibilities in the first phase of recognition.

Number of Features	NUMBER OF THE AUTHOR									
	4	10	12	14	15	18	21	24	35	44
1	49.28	29.71	52.17	44.93	44.20	31.16	44.20	39.86	42.75	54.35
2	64.49	54.35	78.99	66.67	66.67	63.77	70.29	63.77	77.54	65.94
3	73.19	66.67	84.06	71.01	80.43	79.71	83.33	81.16	86.96	80.43
4	79.71	72.46	90.58	77.54	87.68	83.33	92.03	88.41	92.03	85.51
5	82.51	81.16	92.03	78.26	92.75	87.68	93.48	92.03	93.48	92.03
6	87.68	82.61	95.65	81.16	95.65	89.86	94.93	92.75	94.20	92.75
7	89.86	86.96	97.10	86.96	97.10	92.03	97.83	95.65	98.55	92.03
8	91.30	88.41	97.10	89.86	97.83	93.48	98.55	96.38	99.28	95.65
9	92.03	92.75	97.10	91.30	97.10	95.65	99.28	97.10	99.28	96.38
10	94.20	94.93	97.83	92.75	96.38	97.10	100.00	96.38	100.00	96.38
11	94.93	94.20	97.83	94.20	94.93	97.10	100.00	95.65	100.00	98.55
12	95.65	92.75	97.83	94.20	96.38	97.10	100.00	95.65	100.00	98.55
13	95.65	94.20	98.55	96.38	97.10	95.65	100.00	97.10	100.00	99.28
14	97.10	93.48	98.55	97.10	97.10	96.38	99.28	98.55	99.28	96.55
15	96.38	94.93	98.55	97.10	97.10	96.38	100.00	98.55	99.28	99.28

fig. 4 Recognition percentages according to the number of features.

Conclusion

Haar orthonormal system and matrices were first introduced. Haar transform was then defined and some of its properties studied. A fast computation algorithm was thus obtained. After the particular case of Haar transform power spectrum was studied, it was demonstrated that, under certain conditions (orthogonal matrix whose 1st row vector consists of ones only), any transform satisfied Parseval's formula and possessed at least two invariants by translation. Finally, to bring to light the benefit of using invariants thus defined, a character recognition experiment was described.

References

1. WILLIAM K. PRATT, JULIUS KANE, C. ANDREWS — "Hadamard Transform Image Coding" IEEE, vol.57, N°1, January, 1969

2. NASIR AHMED, K.R. RAO, A.L.ABOUSSATTAR — "BIFORE or Hadamard Transform" IEEE, vol.Au-19, n° 3, September, 1971

3. PAUL P. WANG, ROBERT C. SHIAU — "Machine Recognition of printed Chinese Characters via Transformation Algorithms" Pattern Recognition, vol.5, pp.303-321, 1973

4. Laveen KANAL — "Patterns in Pattern Recognition 1968-1974" IEEE, vol.IT-20, n° 6, November, 1974

5. A. ROSENFELD — "Picture Processing by Computer" Academic Press, 1969

6. G. STAMON, S. WENDLING — "Etude Comparative de deux algorithmes de traitement global" A.F.C.E.T. (RABAT), 1975

7. H. ANDREWS — "Introduction to Mathematical Techniques in Pattern Recognition" New-York : Wiley, 1972