AN INTERLEAVING TECHNIQUE FOR BLOCK CODING OF BLACK-AND-WHITE FACSIMILE DATA

By

D. WEI-FANG HSU

Bachelor of Business Administration

National Chung Hsing University

Taiwan, R. O. C.

1981

Submitted to the Faculty of the Graduate College
of the Oklahoma State University in
Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE
December, 1991

AN INTERLEAVING TECHNIQUE FOR BLOCK CODING

OF BLACK-AND-WHITE FACSIMILE DATA

Thesis Approved:

Huizhu Lu
Thesis Advisor

J Chandler

D. E. Hedrick

Thomas C. Collins
Dean of the Graduate College

ACKNOWLEDGEMENTS

It is with great appreciation that the author acknowledges the support and assistance of those contributing to the completion of this thesis. Special thanks are extended to my advisor, Dr. Huizhu Lu, for her guidance, advisement and friendship throughout the duration of my graduate program. Many thanks are also extended to Dr. Hedrick and Dr. Chandler for advice and serving on my committee.

To the friends I met here, thanks for your caring and support. A sincere thanks goes to Mrs. Jenny Hong for taking care of my two years old daughter during my school time.

This thesis is dedicated to my family and my parents, Chun Doong and Yu-Yen-Chiao Doong, whose constant love and support has enable me to complete my educational goals. Lastly and most important; a special thanks and much love goes to my husband, Ker-Jen Hsu, for his love, patience, help, advice, much encouragement and putting up with me as I have finished.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

Figure		Page

CHAPTER I

INTRODUCTION

Motivation

Communication is changing. The need for digital storage and transmission of two-tone images such as engineering drawings, letters, newspaper pages, maps, finger print cards, and other documents has been increasing rapidly, especially with the advent of personal computers and modern telecommunications.

Image-oriented systems such as work stations and facsimile equipment use scanning techniques in order to convert images to electronic data. This is usually done on a line-by-line basis in order to provide a continuous stream of picture elements, or pixels for short (some authors prefer the slightly shorter term pels), which compose the image. The resulting image stream is then stored or transmitted, ultimately driving an image-reproduction or -display device.

In image systems, clarity depends on the fineness (resolution) of the scan. The standard measure used to quantify document scanning resolution is the number of pixels per inch represented in the scanned imagery. In digital document systems, this measure is referred to as points per inch (ppi). The minimum vertical resolution that is required to reproduce a legible page of text is approximately 150 ppi which is the standard printed text and office material scanning resolution. For applications involving documents with detailed line drawings, chemical or mathematical formulas, higher resolution (300 ppi) scanning may be required. TABLE I shows the number of bits required to store an 8 1/2 by 11 inch page scanned at various resolution [Green 89].

A standard A4 (8.5" X 11") letter sampled at a density of 150 ppi (points per inch) generates close to 2 million bits. Clearly, such a high volume of data per document is often uneconomical to store or transmit directly, so techniques of compression by encoding data before transmission are of great interest.

TABLE I

IMAGE SIZE IN PIXELS FOR AN 8 1/2 X 11 INCH PAGE

POINTS/INCH	PIXELS	NUMBER OF BITS
50	425 x 550	2.34×10^5
100	850 x 1100	9.35×10^5
150	1275 x 1650	2.10×10^6
200	1700 x 2200	3.74×10^6
300	2550 x 3300	8.42×10^6
400	3400 x 4400	1.50×10^7

Preliminary Literature Review

For efficient coding, there are always two fundamental concepts to converting information stored on paper to a bit-mapped electronic format. One is image-data compression, which is to reduce the amount of redundant data required either to transmit or to store an image. These redundancies can occur either horizontally, along the line (one-dimensional redundancies), or both horizontally and vertically, from line to line (two-dimensional redundancies). The other concept is the immunity to transmission errors, which always are caused by channel noise.

A large number of efficient coding methods for black-white images have been proposed and studied [Huang 77] [Yasuda 80] [Hunter 80] [Ronald 80] [Hou 83]. A brief summary of some techniques are given below, a detailed overview will be discussed in Chapter II. Figure 1 shows a convenient classification of compression algorithms for binary images.

Binary Data Compression Techniques

white block skipping	run-length coding	relative addressing techniques	other
*1-D	*fixed length	*PDQ	*Pattern recognition
*2-D	*Huffman	*RAC	
*quad-tree [Samet 80]	*MHC	*READ	
	*An & Bn		
	*THC		

Figure 1. Binary Data Compression Techniques

White block skipping coding

White block skipping (**WBS**) is a very simple but effective compression algorithm. Each sampled scan line is divided into N -pixel blocks. For a block containing all white pixels, the 1-bit codeword "0" is used. For a block containing at least one black pixel, $N+1$ -bit codeword was used, the first bit being "1" and the remaining N bits being the binary pattern of the N -pixel block (white = 0, black =1). Thus, if $N = 4$ and a block contains two white pixels followed by two black pixels, the block pattern is "0011" and the codeword for that block is "10011".

Run-length coding

In run-length coding (RLC), a run of consecutive pixels of the same color (i.e., black or white) is combined together and represented by a code word for transmission. Since runs of black and white pixels alternate, except for the first run of each line, the color of the run need not be transmitted. It is useful whenever large runs of 0s are expected. Such a situation occurs in printed documents, graphics, weather maps, and so on, where the probability of a white pixel is close to unity.

Prediction differential quantization and Relative address coding

Prediction differential quantization (PDQ) is an extension of run-length coding, where the correlation between scan lines is exploited. The method basically encodes the overlay information of a black run in successive scan lines. [Huang 72] [Usubuchi 80] [kobayashi 74]

Relative address coding (RAC) uses the same principle as the PDQ method and computes run-length difference by tracking either the last transition on the same line or the nearest transition on the previous line. [Wakahara 74] [Yamazaki 76] [Wakahara 76]

Pattern recognition coding

Pattern recognition coding techniques exploit macroscopic properties of images. It is based on the observation that in documents containing text, several patterns (e.g., characters) occur repeatedly and, therefore, compression efficiency can be improved by identifying these patterns and transmitting to the receiver only their identification codes. Incoming patterns are compared and matched to already transmitted patterns if a correct match is detected, only the position of the pattern and the identification code of matching pattern are transmitted. If a match has not been detected, the incoming pattern is both transmitted and added to the library as a new pattern.

Research Objectives

In the field of two-tone facsimile data compression a large number of techniques have been described. For those methods, which can achieve higher compression ratios, feature variable-rate output at the encoder and inevitably require variable-velocity scanners for synchronous transmission over digital channels. One penalty for those efficient source codings is an increased sensitivity to errors in the transmission channel: a single received bit error will corrupt the remainder of the image. These compression systems, therefore, require high quality channels for successful image transmission, and most compression schemes also require codebooks to encode and decode. Thus, these compression systems' applications are limited and implementations are expensive. Although CCITT (International Telegraph and Telephone Consultative Committee) has defined algorithms of one- and two-dimensional standards for data compression [CCITT 80], but the CCITT compression standards may not provide the optimum compression for specific types of imagery. For this reason, a lot of new researches and techniques are still being proposed for data compression of binary imagery.

This thesis describes a new coding method of two-tone image called Interleave Block Coding (IBC), which is inspired from white block skipping (WBS) in some ways. With this method, a binary image data is represented by a fixed-rate output which strongly immunizes against transmission errors, and the design and implementation of Interleave Block Coding is quite simple.

The compression method presented in this thesis yields intelligible images under quite noisy transmission conditions where the levels of errors may prove too great for variable-rate techniques. The constant output bit rate provides simple interfacing with fixed-rate equipment. The system is suitable where low implementation complexity has priority over high compression yields. The system, by virtue of being constant-rate, does not require continuous codeword synchronization as is necessary in variable-rate systems. The coding system can operate in either information-lossless (also called exact coding which allows perfect reproduction of the picture at the receiver in the absence of transmission error) or information-lossy (sometimes called approximate coding which makes an approximation of the picture) mode but a higher immunity may be achieved with lossy operation. In addition to immunizing transmission errors strongly, the coding method encodes block-pairs of binary image data over a field of blocks without the need for codebooks.

Organization of Thesis

The organization of this thesis is as follows. In Chapter I, an introduction and a preliminary literature review are included. In Chapter II, a detailed overview of binary data compression techniques is discussed. The Interleaved Block Coding algorithm and compression ratio are described in detail in Chapter III. The results of simulation studies assessing the objective and subjective performance of IBC are presented in Chapter IV, a Modified IBC is also introduced and simulated in this chapter. In Chapter V, a mathematical model for the probability of overflow occurring in a coding field is derived and discussed. Chapter VI will deal with a comparison of the effects of transmission errors for IBC, Modified IBC and WBS techniques. The suggestions for further works and conclusions constitute Chapter VII, the final chapter.

CHAPTER II

AN OVERVIEW OF BINARY DATA COMPRESSION TECHNIQUES

In this chapter we described several binary image coding techniques in some detail. Our purpose is not to present the current state of the art, for that changes almost on a daily basis. Rather, we wish to show some basic principles of these coding methods and how they are related or derived from. Finally, we will discussed their advantages and disadvantages in some outline.

White Block Skipping Coding

Most two-tone images consist of black symbols printed on a white background with the amount of white space far exceeding the amount of black. Therefore, if we skip the white space and transmit only the black, we can reduce the number of bits to be used. White block skipping (WBS) scheme is based on this idea [Coulon 74]. Each scan line is divided into blocks of N pixels. If the block contains all white pixels, it is coded by a 0. Otherwise, the code word has $(N+1)$bits, the first bit being 1, followed by the binary pattern of the block (See Figure 2).

```
  0       0      11110     0     11111     0     10001
0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1   Data
```

Figure 2. 1-D White Block Skipping Coding, N = 4

An adaptive WBS scheme improves the performance significantly by coding all white scan lines separately. A "0" is assigned to an all-white scan line. If a line contains at least one black pixel, a "1" precedes the regular WBS code for that line [Huang 75a]. The WBS method which was proposed and studied by de Coulon and Johnsen [Coulon 76] [Johnsen 76], can also be extended to two dimensions by considering MxN blocks of pixels. An all-white block is coded by a 0. Other blocks are coded by (MN+ 1) bits, whose first bit is 1 followed by the block bit pattern.

The WBS is a very simple and efficient compression algorithm, and it works well especially when an image contains large white space. At the other extreme, the largely-black images not can achieve the compression goal. In a practical compression scheme the system must be designed to cater for the worst-case image. The coding technique we discuss next does a better job in this problem.

Run-Length Coding

A binary image can be viewed as a sequence of alternating strings of 0s (white) and 1s (black). If a typical scan line within a digital image is examined, it is generally (a) either all white or (b) is represented by long string of white pixels interrupted by short strings of black pixels. One scheme for compressing the high degree of correlation is called run-length coding (RLC) [Capon 59] [Takagi 75]. In run-length coding the lengths of black and white runs on the scan line are coded and transmitted. The runs alternate in color as that as long as the first run in a line is guaranteed to be a white run no extra bits are needed to specify the color. A length of zero can be used if the line actually starts black. The example of one-dimensional fixed-length RLC is shown in Figure 3. Statistics of such run-length vary from document-to-document. In general, they are highly nonuniform. Variable length coding of run-length takes advantage of the nonuniform probability distribution of run-lengths. Moreover, since the statistics of 0's are usually quite different from these of 1's, different code tables are often used. Most of the variable length codes have been based on Huffman's procedure.

0 0 0 0 0 0 0 0	1 1 1	0 0 0 0 0	1 1 1 1	0 0 0 0 0 0 0	1	Data
8W	3B	5W	4B	7W	1B	Run length
1000	0011	0101	0100	0111	0001	RLC, fixed

Figure 3. Run-Length Coding, Fixed-Length 4-bit Codeword

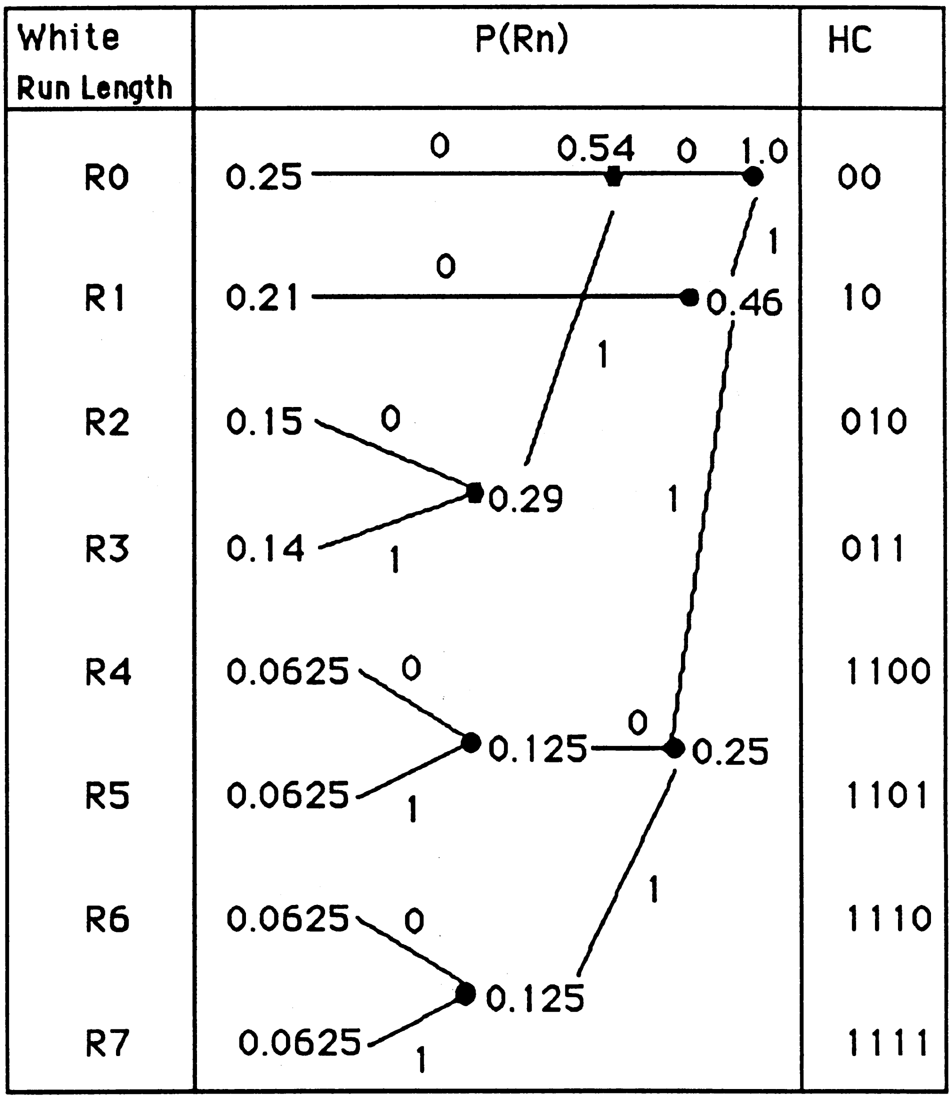
It is well know that for a given set of message probabilities, the most efficient code is a Huffman code [Huffman 52]. Each run length of black or white pixel is given a code based on the probability of that particular length. Figure 4 illustrates the tree structure associated with Huffman encoding for a simple example considering only eight statistically occurring white pixel run lengths within a scanned line. This type of tree-structured encoding yields a set of unique codes that can be decoded sequentially as the encoded bit stream is transmitted serially. The most frequently occurring run lengths will be represented by the shortest code words, and will be decoded more rapidly than less frequently occurring run lengths that are represented by longer code words.

Since most documents have more than a thousand pixels per line, code tables could, in principle, contain a large number of code words. However, to avoid a large code book, the truncated (THC) [CCITT no.11 76] and the modified Huffman code (MHC) [CCITT no. 7 76] are used. THC assigns separated Huffman code words for white and black runs up to lengths L_W and L_B, respectively. Typical values of these runs have been found to be $L_W = 47$, $L_B = 15$ [Musmann 77]. The modified Huffman code, which has been recommended by the CCITT (International Telegraph and Telephone Consultative Committee) as one-dimensional standard code for Group 3 facsimile transmission, uses $L_W = L_B = 63$. "Make up codes," incorporating code words of 9 bits or less, are used to represent lengths in multiples of 64 pixels. "Terminating codes" are used to represent run lengths of less than 64 pixels.

TABLE II shows the "make-up codes" used to represent multiples of 64 black or white pixel run lengths. TABLE III is an extended code table for larger paper widths up to A3 in size, which require up to 2560 pixels per line. TABLE IV shows the "terminating codes," used to represent pixel run lengths of less than 64 pixels. Black and white run lengths are encoded differently because the statistics of white run lengths differ significantly from the statistics of black run lengths in scanned imagery. As an example, a common run length is an entire line of white. If the document is scanned at 300 pixels/inch, a white run of 2544 pixels is indicated. This run length can be represented by a "make-up codes" for 2496 pixels (000000011110), a white run-length "terminating codes" for 48 pixels (00001011), and an end-of-line (EOL) code (000000000001).

Noteworthy among other forms of variable-length coding are the An and Bn codes [Huang 75b]. The An codes, also called Ln codes [Laemmel 51]; the Bn codes, also called Hn codes [Huang 74]; because these codes are not used practically, we do not described in detail in this section. The reader is referred to references for detail.

Run-length coding aimed at reducing redundancy, it reduces the storage requirement by factors of 5 to 15 for typical images but a single bit error will ruin the rest of the scan line. To decrease sensitivity to channel errors, it inevitably requires high quality channels. For Huffman code, Table 3 shows that short run lengths can be encoded with code words of more bits than the actual run length itself (e.g., a run length of 2 white pixels is represented as a 4-bit code word). If images are extremely "busy," with a significant amount of high frequency variation, it is possible to obtain "negative compression" using this algorithm. The algorithm also requires a codebook to represent black and white run length, and table look-up involves a large dictionary for its implementation.



Rn = n white pixels run length.
P(Rn) = the probability of occurrence of a bit string of length Rn.

Figure 4. Huffman Coding Tree Structure Example

TABLE II

MODIFIED HUFFMAN CODE MAKE-UP CODES

White Run length	Code Word	Black Run Length	Code Word
64	11011	64	0000001111
128	10010	128	000011001000
192	010111	192	000011001001
256	0110111	256	000001011011
320	00110110	320	000000110011
384	00110111	384	000000110100
448	01100100	448	000000110101
512	01100101	512	0000001101100
576	01101000	576	0000001101101
640	01100111	640	0000001001010
704	011001100	704	0000001001011
768	011001101	768	0000001001100
832	011010010	832	0000001001101
896	011010011	896	0000001110010
960	011010100	960	0000001110011
1024	011010101	1024	0000001110100
1088	011010110	1088	0000001110101
1152	011010111	1152	0000001110110
1216	011011000	1216	0000001110111
1280	011011001	1280	0000001010010
1344	011011010	1344	0000001010011
1408	011011011	1408	0000001010100
1472	010011000	1472	0000001010101
1536	010011001	1536	0000001011010
1600	010011010	1600	0000001011011
1664	011000	1664	0000001100100
1728	010011011	1728	0000001100101
EOL	000000000001	EOL	000000000001

TABLE III

EXTENDED MODIFIED HUFFMAN CODES

Run Length (black and white)	Makes-up Codes
1792	00000001000
1856	00000001100
1920	00000001101
1984	000000010010
2048	000000010011
2112	000000010100
2176	000000010101
2240	000000010110
2304	000000010111
2368	000000011100
2432	000000011101
2496	000000011110
2560	000000011111

TABLE IV

MODIFIED HUFFMAN CODE TERMINATING CODES

White Run Length	Code Word	Black Run Length	Code Word
0	00110101	0	0000110111
1	000111	1	010
2	0111	2	11
3	1000	3	10
4	1011	4	011
5	1100	5	0011
6	1110	6	0010
7	1111	7	00011
8	10011	8	000101
9	10100	9	000100
10	00111	10	0000100
11	01000	11	0000101
12	001000	12	0000111
13	000011	13	00000100
14	110100	14	00000111
15	110101	15	000011000
16	101010	16	0000010111
17	101011	17	0000011000
18	0100111	18	0000001000
19	0001100	19	00001100111
20	0001000	20	00001101000
21	0010111	21	00001101100
22	0000011	22	00000110111
23	0000100	23	00000101000
24	0101000	24	00000010111
25	0101011	25	00000011000
26	0010011	26	000011001010
27	0100100	27	000011001011
28	0011000	28	000011001100
29	00000010	29	000011001101
30	00000011	30	000001101000

TABLE IV (Continued)

White Run Length	Code Word	Black Run Length	Code Word
31	00011010	31	000001101001
32	00011011	32	000001101010
33	00010010	33	000001101011
34	00010011	34	000011010010
35	00010100	35	000011010011
36	00010101	36	000011010100
37	00010110	37	000011010101
38	00010111	38	000011010110
39	00101000	39	000011010111
40	00101001	40	000001101100
41	00101010	41	000001101011
42	00101011	42	000011011010
43	00101100	43	000011011011
44	00101101	44	000001010100
45	00000100	45	000001010101
46	00000101	46	000001010110
47	00001010	47	000001010111
48	00001011	48	000001100100
49	01010010	49	000001100101
50	01010011	50	000001010010
51	01010100	51	000001010011
52	01010101	52	000000100100
53	00100100	53	000000110111
54	00100101	54	000000111000
55	01011000	55	000000100111
56	01010101	56	000000101000
57	01011010	57	000001011000
58	01011011	58	000001011001
59	01001010	59	000000101011
60	01001011	60	000000101100
61	00110010	61	000001011010
62	00110011	62	000001100110
63	00110100	63	000001100111

Relative Addressing Techniques

It has been estimated that 50% of all transitions from white to black, or vice versa, on a scan line are likely to occur on the next adjacent scan line within a few pixel positions of the transition on the first line. Relative addressing coding (RAC) also known as two-dimensional encoding, take advantage of the vertical correlation between adjacent lines within a document .

The Modified Relative Element Address Designate (Modified READ) coding scheme [CCITT no.39 79] is a simplification of the original READ coding [CCITT no.42 78], which considered vertical reference coding out to plus or minus twenty-six pixels. The Modified READ only uses vertical reference coding out to plus or minus three pixels. The Modified READ algorithm has been recommended by CCITT as two-dimensional coding for Group 3, Group 4 facsimile apparatus. CCITT Group 3 compression utilizes a combination of one-dimensional run length encoding and two-dimensional coding based on bit reversals between adjacent lines. In these encoding modes, a given line is encoded using the MHC technique, and K-1 lines are encoded using Modified READ technique. The Group 3 standard recommends K values of 2 or 4. CCITT Group 4 compression utilizes only the two-dimensional coding based on bit reversals. Group 4, codes the first line using the MHC, while the remaining lines use the Modified READ algorithm.

To encode using the Modified READ algorithm, the relationship between a transition on the current line (coding line) and the previous line (reference line) is determined. It is a line-by-line scheme in which the position of each changing element on the present line (coding line) is coded with respect to either the position of a corresponding changing element on the previous line (reference line), which lies immediately above the present line, or with respect to the preceding changing element on the present line. After the present line has been coded, it becomes the reference line for the next line. Depending on the relative position of the changing element that is being coded, the coder operates in three modes, referring to Figure 5 [CCITT 80].

(A) Pass Mode

In this mode, the element of b2 lies horizontally to the left of a1,using the definitions, it occurs whenever the white or black runs on the reference line are not adjacent to corresponding white or black runs on the present line. The pass mode is represented by a single codeword.

(B) Vertical Mode

In this mode, the element a1 is sufficiently close to b1 and is therefore coded relative to the position of b1. It is used only if a1 is to the left or right of b1 by at most 3 pixels.

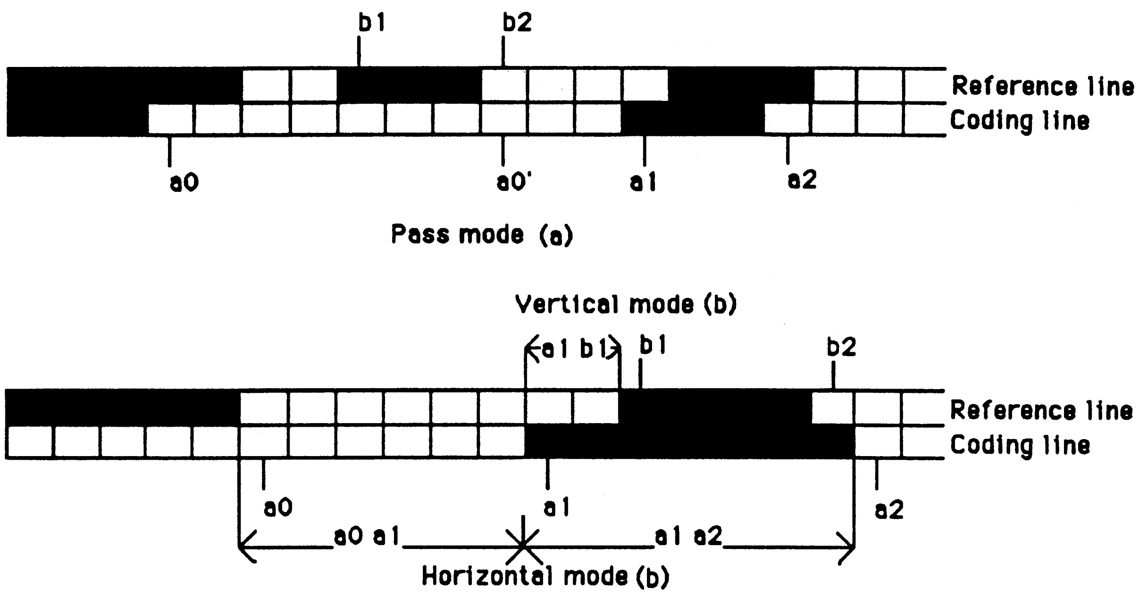
(C) Horizontal Mode

If a1 is not sufficiently close to b1, then its position must be coded by horizontal mode. Thus, the run-length a0a1 and a1a2 are coded using the concatenation of three codewords H, M(a0a1) and M(a1a2). Codeword H, taken to be 001, serves as a prefix or flag, and M(a0a1) and M(a1a2) are taken from the codes tables to represent the colors and values of the run-lengths a0a1 and a1a2. The Table is based on the modifies Huffman procedure shown in TABLE II, III and IV.

The one-dimensional and two-dimensional code words listed in TABLE V [Netravali 88]. A complete description of the one- and two-dimensional CCITT standards for compression is contained in [CCITT 85].

Two-dimensional schemes take advantage of correlation between successive scan lines to obtain better compression. At normal resolution the two-dimensional codes improve the compression by 10 to 30% over modified Huffman code. With two-dimensional codes, since a line coded requires a current reference line above it, a single channel error can cause significant degradation in the rest of the image, although these algorithms achieve high compression-ratio. In most systems, decompression speed is more critical than compression speed. As one would expect, the decompression speed of

Modified **READ** decreased, due to the computation necessary for decoding images, and this algorithm also need a codebook to look-up.



a0: The first changing element (whose color is different from that of the previous element along the same scan line) on the coding line.
a1: The next changing element on the coding line; by definition, it has opposite color to a0 and gets coded next.
a2: The changing element following a1 on the coding line.
b1: The changing element on the reference line to the right of a0 with the same color as a1.
b2: The changing element following b1 on the reference line.

Figure 5. Example of CCITT (a) Pass Mode (b) Vertical and Horizontal Mode

TABLE V

CCITT TWO-DIMENSIONAL CODES

Mode	Element to be coded		Notation	Code word
Pass	b1,b2		P	0001
Horizontal	a0a1,a1a2		H	001+M(a0a1)+M(a1a2) (see Note 1)
Vertical	a1 just under b1	a1b1=0	V(0)	1
	a1 to the right of b1	a1b1=1	VR(1)	011
		a1b1=2	VR(2)	000011
		a1b1=3	VR(3)	0000011
	a1 to the left of b1	a1b1=1	VL(1)	010
		a1b1=2	VL(2)	000010
		a1b1=3	VL(3)	0000010
Extension	2-D extension			0000001xxx
	1-D extension			000000001xxx
	End-of-line (EOL) code word			000000000001
	1-D coding of next line			EOL + '1'
	2-D coding of next line			EOL + '0'

Note 1. Code M( ) of the horizontal mode represents the code words in TABLE II, TABLE III and TABLE IV.

Pattern Recognition Coding

The idea of pattern recognition coding is one which a majority of researches of facsimile coding have thought about for many years. The first realization of this idea is Symbol-Matching Schemes which were first reported by Pratt et al. [Pratt 76, 80], and further implementation was studies by Silver [Silver 81, 84]. Pattern recognition technique was proposed by Johnsen et al. [Johnsen 82] for two level images. A pattern-matching scheme was checking whether incoming pattern is similar to an already transmitted pattern. The image source is a source of pattern, such as characters, line segments, and black regions. The document is examined from the leftmost to rightmost line by line. When a black pixel is located, the pattern isolator picks up the pattern. This pattern is either a symbol, being defined as a set of black pixels completely surrounded by white pixels, or a nonsymbol when no symbol can be extracted, and thus is a fraction of the black region. A template matching with existing library patterns is made in order to decide whether the incoming pattern is similar to an already transmitted pattern. To reduce the time-consuming template matching, a screening of the library patterns is performed, so that only possible matching patterns are considered. If a correct match is detected, information about the position of the pattern in the current document and its library identification must be coded and transmitted. If no match has occurred, the incoming pattern is transmitted and added to the pattern library which is empty at the beginning of the coding and is gradually built up by the incoming library patterns. A library update and management units adds patterns to and delete patterns from the library, and organizes them for the quickest possible match and most efficient coding. The schematic diagram is illustrated in Figure 6 [Yasuda 85].

According to Yasuda's result of simulation, it almost doubles the data compression ratio compared with the Modified READ coding. Although the scheme described above is very efficient, but this scheme is still in its infancy and the implementation is quite expensive and complex, since the transmitter included a pattern matching.

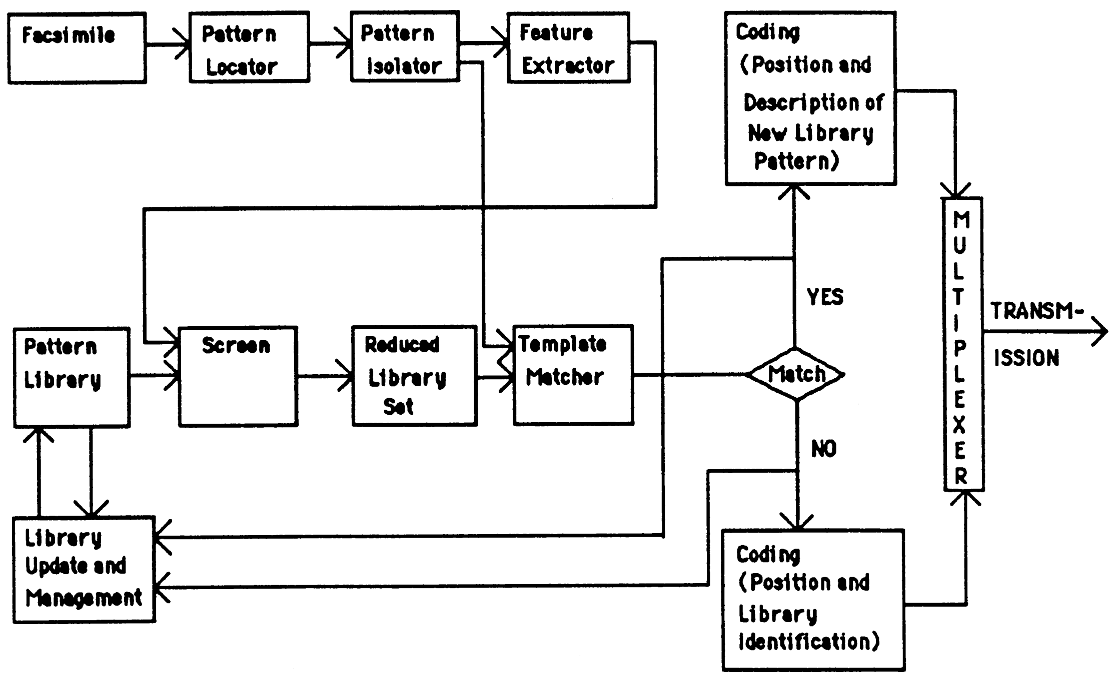


Figure 6. Block Diagram of Pattern Match

CHAPTER III

INTERLEAVED BLOCK CODING SCHEME

In a typical two-tone image, most part of the image is white, the amount of black is usually a very small fraction of the total area of the image. Therefore, it appears that if we rearranging the black element content to occupy the white spaces reduces the number of bits to be transmitted. A scheme based on this idea is described now. This scheme is similar to the **WBS** in some ways, for the **WBS**, we skip the white space and transmit only the black; for the **IBC**, we rearrange the black element content to occupy the white spaces. The method is very simple. Each scan line is divided into pairs of N -bit blocks and for each block-pair a fixed-length codeword is transmitted consisting of a three-bit header followed by an N -bit block pattern. An image is restored from a coding field which comprised a fixed number of block-pairs.

Definition of Block-Pair States

Based on the presence of the all-white, all-black and nonwhite block pattern on left or right side of block-pair, each block-pair is classified as being in one of nine states described below. These block-pair states are:

(a) White-White (WW), both left and right blocks contain all-white pixels.

(b) Black-White (BW), only the right block is all-white and the left block is all-black.

(c) Nonwhite-White (NW), only the right block is all-white and the left block is nonwhite, i.e., contains at least one black pixel but all-black block exclusive.

(d) White-Black (WB), only the left block is all-white and the right block is all-black.

(e) White-Nonwhite (WN), the left block is all-white and the right block is nonwhite.

(f) Nonwhite-Nonwhite (NN), both left and right blocks contain one or more black elements but all-black block pair is excluded.

(g) Nonwhite-Black (NB), the right block is all-black and the left block is nonwhite.

(h) Black-Nonwhite (BN), the left block is all-black and the right block is nonwhite.

(i) Black-Black (BB), both left and right blocks of the block-pair are all -black.

For data-compression, only the combinations of three header coding bits identify these nine states. Instead of increasing the header wordlength to four bits to identify the ninth state (BB), header word 111 is chosen to represent the nonwhite-black (NB) and the all-black (BB) block-pair states. The fact is that the combinations of header (111) and adjoined block patterns (all 1s) will never occur in the other eight-state arrangement. The nine states' header coding are shown below.

(a) Header word 000 represents WW.

(b) Header word 101 represents BW.

(c) Header word 100 represents NW.

(d) Header word 011 represents WB.

(e) Header word 010 represents WN.

(f) Header word 001 represents NN.

(g) Header word 111 represents NB.

(h) Header word 110 represents BN.

(i) Header word 111 also represents BB.

Coding Procedure

The coding procedure consists of two stages.

(A) Scanning the presence of the NN (nonwhite-nonwhite) block-pairs. The scan starts from the leftmost to rightmost block-pair of the field. When a NN block-pair is detected, its right nonwhite block pattern is stored into a first-in first-out (FIFO) buffer.

(B) Encoding block-pairs. The encoding stage commences after the scan stage is completed. From the leftmost block-pair of the field, each block-pair is encoded. A three-bit header word is assigned to be the state identification of each block-pair. This header is followed by an N -bit block pattern, where N is the number of pixels the block is assumed to comprise. The N -bit block pattern depends upon the state of the block-pair. For the block-pair of a **NW, WN, NB** or **BN** state, the N -bit block pattern is the nonwhite block of the pair. For the black-black (**BB**) state, the all-black left block pattern is transmitted. If the block-pair is **NN** state, since the right nonwhite block is fed into buffer the left nonwhite pattern is the only choice. Finally, if the current block-pair exhibits a all-white (**WW**), black-white (**BW**) or white-black (**WB**) the transmitted block pattern is taken from the FIFO buffer. If the buffer is empty an all-white pattern is transmitted respectively. In this coding method, right nonwhite elements of **NN** block-pairs in the field are carried out by the **WW, BW** and **WB** block-pairs. The example is shown in Figure 7.

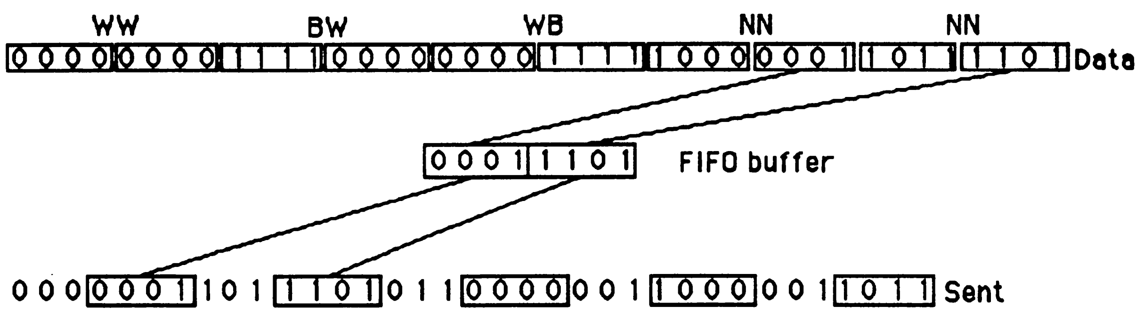


Figure 7. Example of IBC Coding Procedure, 4-bit Block

Decoding Procedure

At the receiver a reverse two-stages process is implemented.

(A) Examine the input header words for **WW**, **BW** and **WB** states. At the receiver, when a **WW**, **BW** or **WB** state is examined the adjoined N -bit block pattern is fed into a FIFO buffer.

(B) Decoding block-pairs. At the end of a scan for a field, the decoding stage starts. Here the block-pair pattern is reconstructed according to the header word and its adjoined block pattern. For a **WN** or **NW** block-pair, all white *N* -bit block is reinserted into left or right block. If block-pair is **BN** state, all-black *N* -bit block is inserted into the left block. For **NB** and **BB** block-pairs, have the same header (111); both states transmit left block pattern, so the all-black block is reinserted into the right block. If the block-pair is the **WW** state, then *2 N* -bit blocks for all-white are restored. For the **BW** and the **WB** states, the all-black and the all-white blocks are inserted into suitable position respectively. For a **NN** block-pair the right nonwhite elements are taken from the FIFO buffer. An example is given in Figure 8.

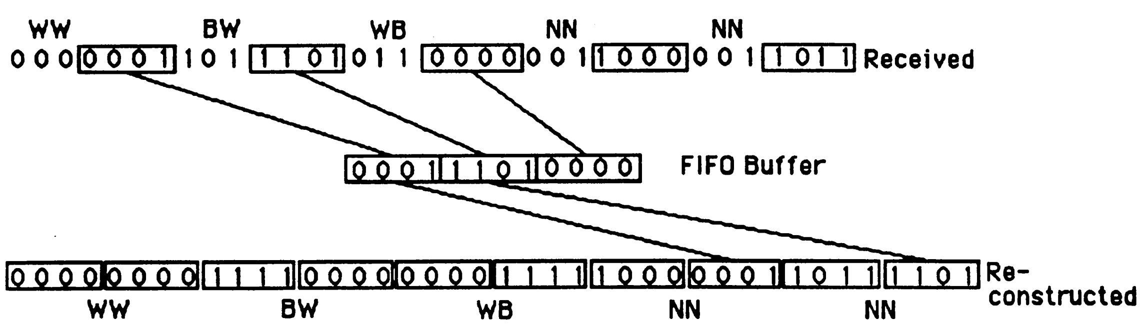
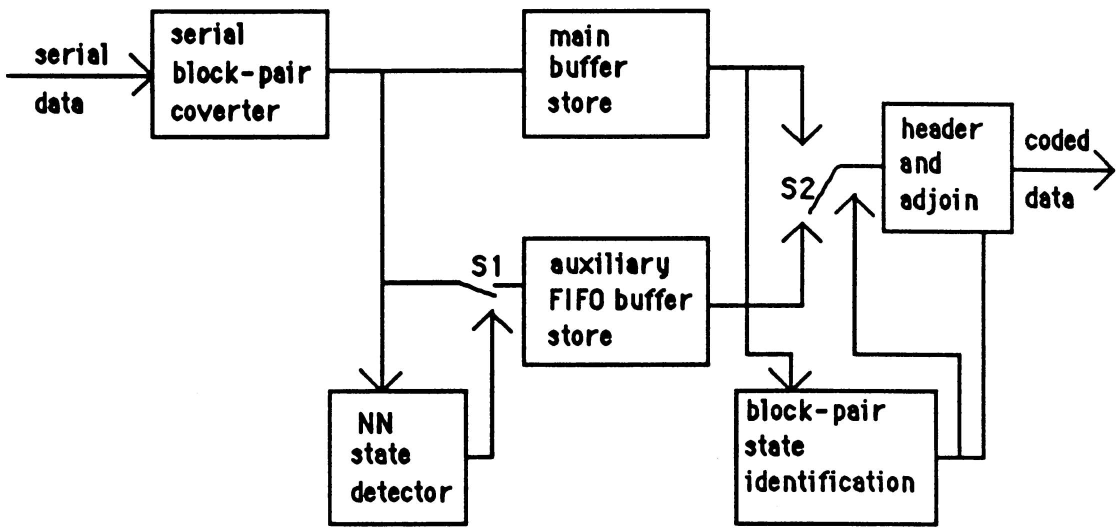


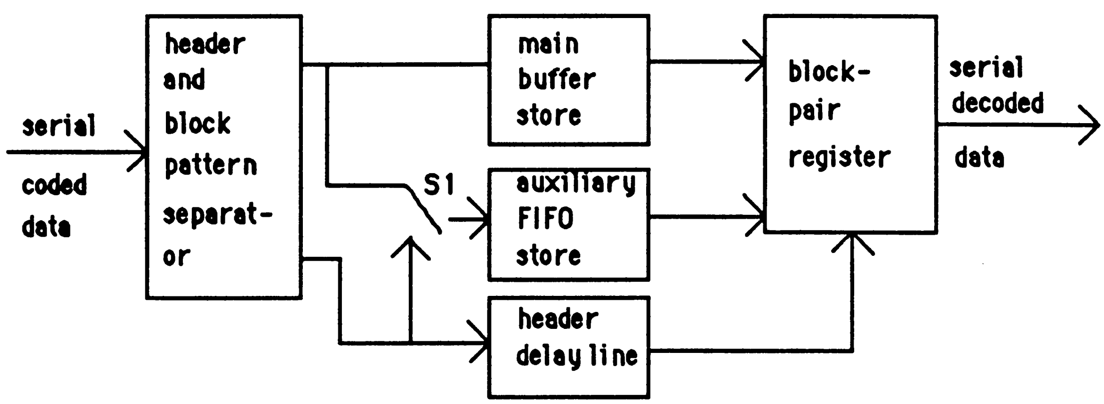
Figure 8. Example of IBC Decoding Procedure, 4-bit Block

Coding Diagram

In view of the interleaving nature of the compression technique, the method has been designated "Interleaved Block Coding" (IBC). Figure 9 contains the block diagram for instrumentation of this compression system. In the encoder the auxiliary FIFO buffer store holds the right nonwhite elements of all NN block-pairs in the field. The main buffer store implements the field delay. At the end of a encoded field the FIFO contents are inserted into the output, via S2, when **WW**, **BW** and **WB** patterns are detected. For each block-pair the header and block pattern are adjoined and released serially into the transmission channel. At the receiver, the header and block pattern components of the codeword are separated and for **WW**, **BW** and **WB** states the block pattern is fed into the FIFO buffer. At the end of a decoded field the contents of the later are reinserted into the NN block-pairs as identified by the delayed header words, and the reconstructed block-pair is fed to the recording unit serially.



(a)



(b)

Figure 9. Interleaved Block Coding Scheme (a) Coding Diagram (b) Decoding Diagram

Encoding Distortion

One impediment in this algorithm is the occurrence of the total number of **NN** block-pairs is over the total number of **WW, BW** and **WB** block-pairs, preventing the complete transmission of the contents of the FIFO buffer at the end of the field. This is called field "overflow", and the simplest strategy is not to transmit nonwhite blocks, their original right nonwhite blocks are replaced by the all-white pattern at the receiver. This process is termed "blanking" of excess nonwhite blocks. The resulting distortion takes the form of lost black elements. In the latter chapter we will discuss some techniques to improve this case where surplus nonwhite blocks are blanked and encoding distortion consists entirely of lost black elements.

Compression Ratio

The basic compression ratio, that can be obtained with the Interleave Block Coding method readily is seen to be:

$$\mathrm{CR} = \frac{2N}{N + 3} \tag{1}$$

$$= \frac{2}{1 + 3/N\,.} \tag{2}$$

Where N is the number of pixels in each block. The bit rate per block is defined as the reciprocal of the compression ratio. The result as a function of N is plotted in Figure 10. It is seen that beyond a value of N about 8 pixels the compression ratio is extremely flat, that means the compression ratio increases only gradually for increasing N and tends to two when the block size grows to infinity.

The compression ratio of the **IBC** is not very high, but in many instances the sacrifice in compression provides for ease of implementation decreasing the complexity of system. In the next chapter a block size of 8 pixels is used throughout the simulation. It is a good choice in terms of performance and efficiency of implementation by using commercial digital ICs and eight-bit microprocessors.

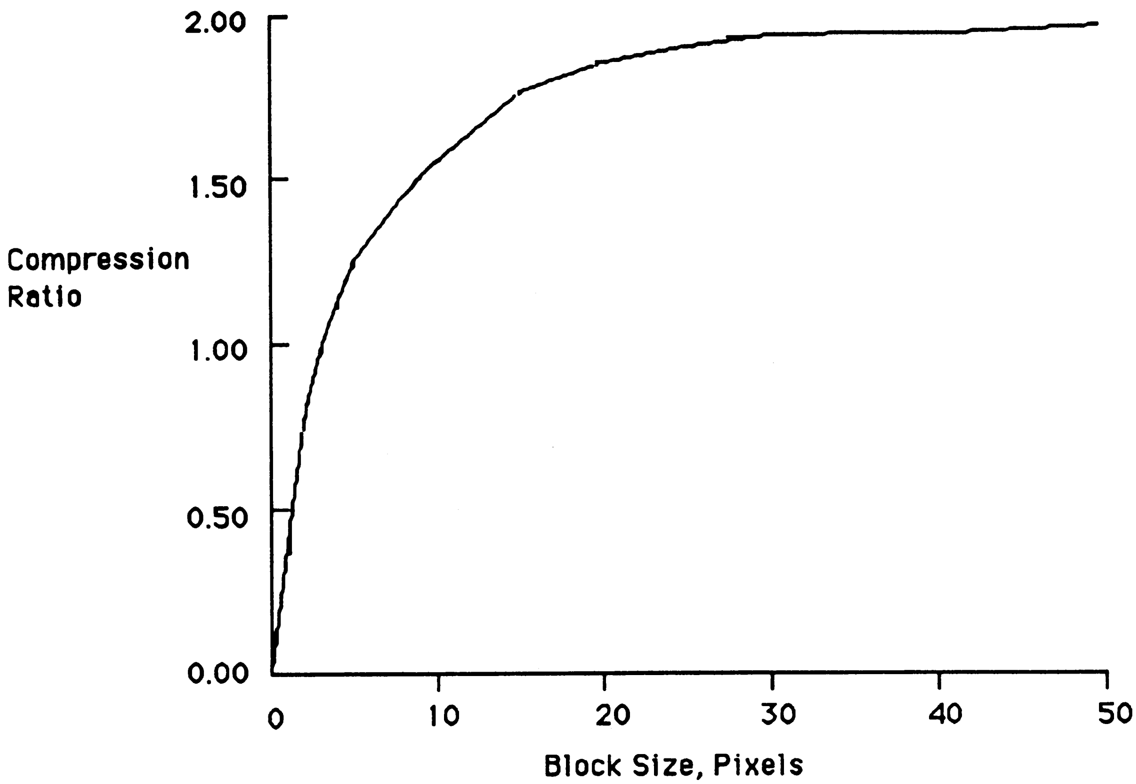


Figure 10. IBC Compression Ratio as a Function of Block Size

CHAPTER IV

SIMULATION STUDIES

This chapter shows the performance of the IBC using computer simulations with eight test images. The eight images are labeled as follows:

A1 Handwritten text

A2 Circuit diagram

A3 Financial statement

A4 Typewritten text

A5 Sample page from general paper

A6 Plotting figure

A7 Chinese text

A8 Trademark design.

The set of test images is shown in Figure 11-1 and -2, which is generally correspond to the set of eight **CCITT** reference documents, see Figure 12-1, -2.

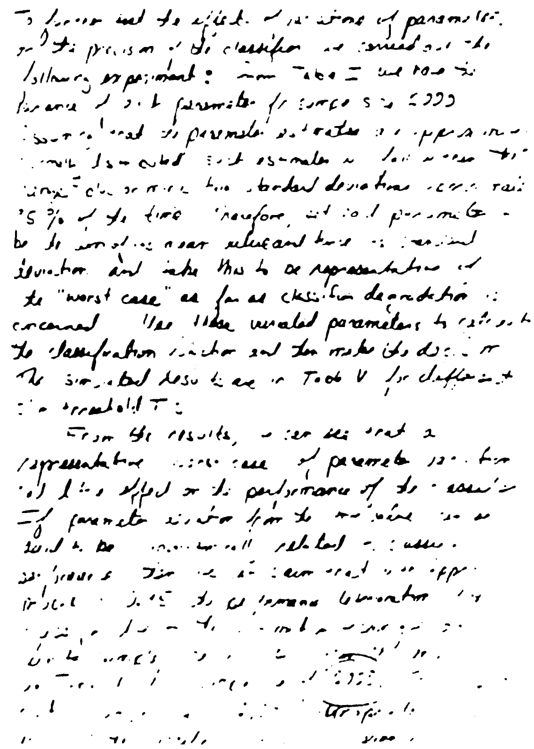
In this experimental simulation, the original documents are all 8 1/2 x 11 in size. For lack of the desired document scanners with appropriate resolution, in order to retain the probability distribution of 150 ppi resolution we have to sample portion of each document in 512 x 480 pixels, which is the limit of our handy equipment (MetraByte's MV2 Frame/Line Grabber), For the block size, we choose 8-bit block throughout the simulation therefore the compression ratio is fixed at 1.45, which corresponds to a bit rate of 0.69 (defined as the reciprocal of the compression ratio).

As mentioned earlier, when the total number of **NN** block-pairs is over the total number of **WW**, **BW** and **WB** block-pairs, there is encoding distortion. The measured probabilities of the block-pair states for the test images are shown in TABLE VI. It is clear that overflow-free, or lossless, coding is obtainable for all the test images when the field size is sufficiently large (like the whole image). Because for the whole image as a test field size, the probability $P_{(WW+WB+BW)}$ is sufficiently larger than the probability P_{NN}. In a practical implementation it is desirable to keep the field size as small as possible. Apart from increased amount of storage and hardware, an important reason is the effect of transmission errors.

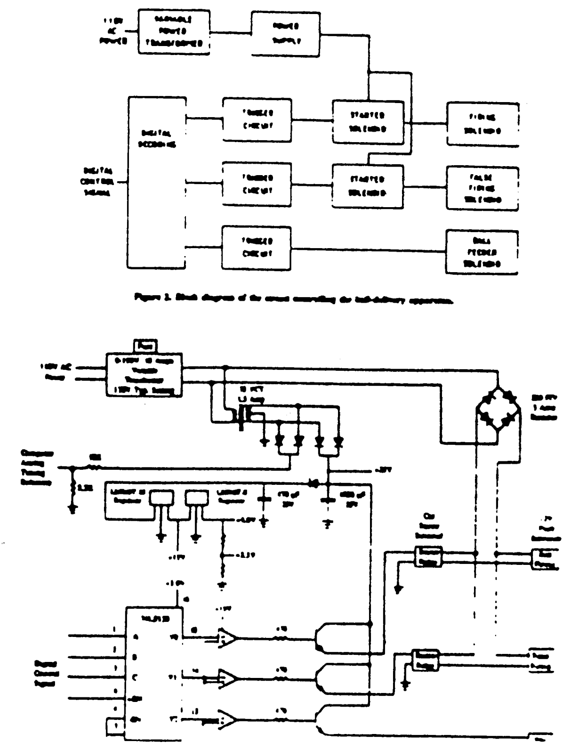
TABLE VI

STATISTICS OF TEST IMAGE FOR BLOCK SIZE OF 8 PIXELS

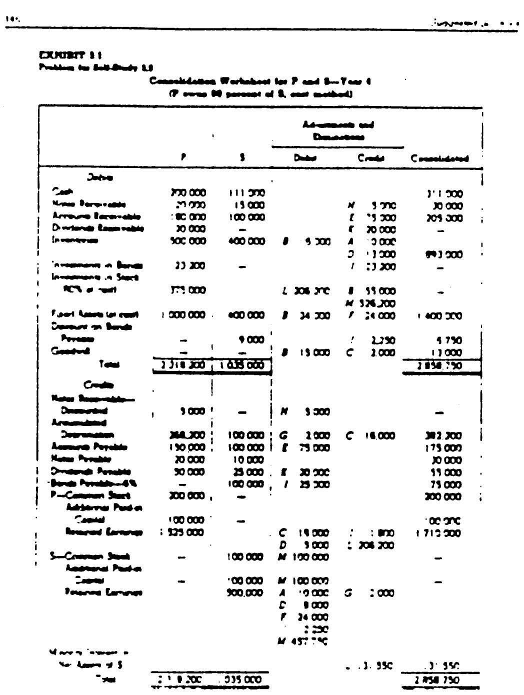
Image	P_{WW}	P_{WB}	P_{WN}	P_{BW}	P_{NW}	P_{NN}	P_{NB}	P_{BN}	P_{BB}	P_0	P_{255}
A1	0.6322	0.0148	0.0923	0.0025	0.0730	0.1324	0.0242	0.0148	0.0137	0.6870	0.0298
A2	0.7720	0.0238	0.0536	0.0023	0.0583	0.0418	0.0077	0.0102	0.0303	0.8118	0.0484
A3	0.7754	0.0182	0.0452	0.0008	0.0340	0.0738	0.0143	0.0120	0.0262	0.8075	0.0418
A4	0.6445	0.0154	0.0504	0.0008	0.0656	0.1688	0.0244	0.0140	0.0161	0.6778	0.0312
A5	0.7424	0.0227	0.0445	0.0008	0.0643	0.0940	0.0066	0.0082	0.0164	0.7764	0.0323
A6	0.7130	0.0228	0.1005	0.0003	0.0848	0.0453	0.0040	0.0026	0.0268	0.7747	0.0396
A7	0.4299	0.0312	0.1404	0.0044	0.1410	0.1361	0.0585	0.0433	0.0152	0.5179	0.0546
A8	0.3969	0.0079	0.0656	0.0033	0.0672	0.0939	0.0424	0.0309	0.2920	0.4353	0.3131



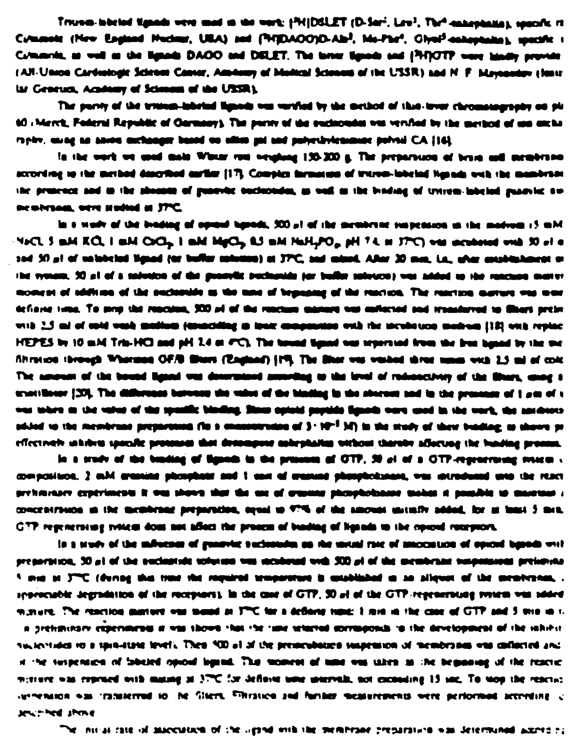
A1 - Handwritten Text



A2 - Circuit Diagram

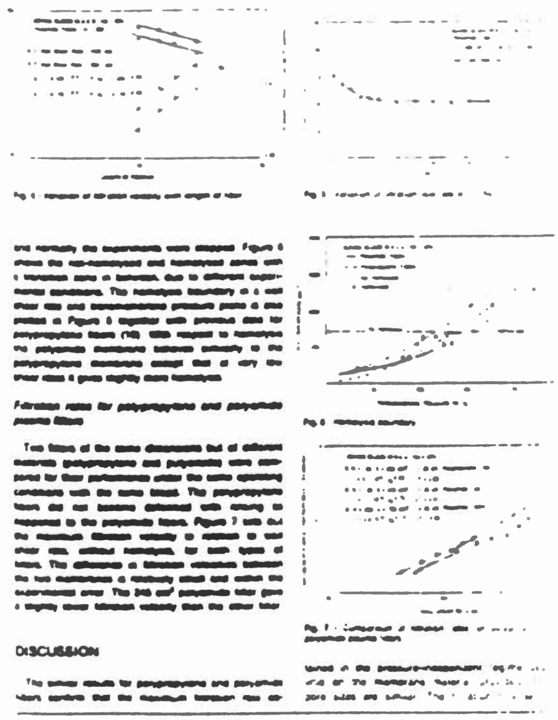


A3 - Financial Statement

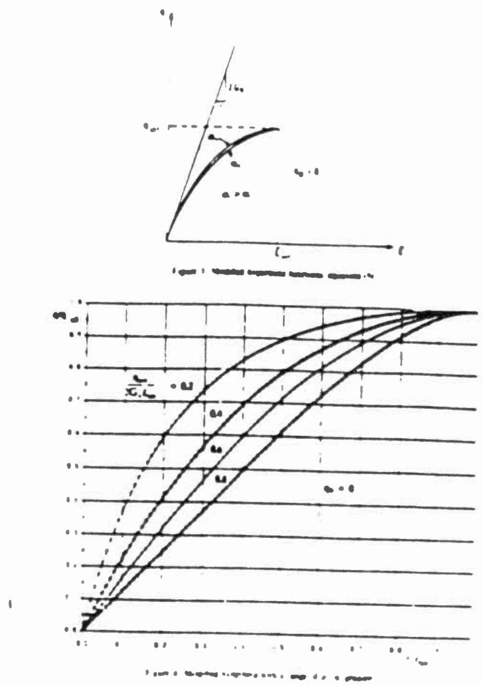


A4 - Typewritten Text

Figure 11-1. Original Test Images Used for IBC (A1 - A4)



A5 - Sample Page

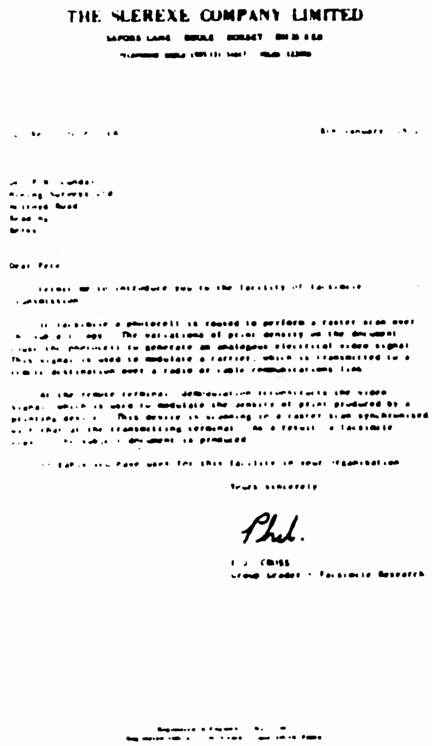


A6 - Plotting Figure

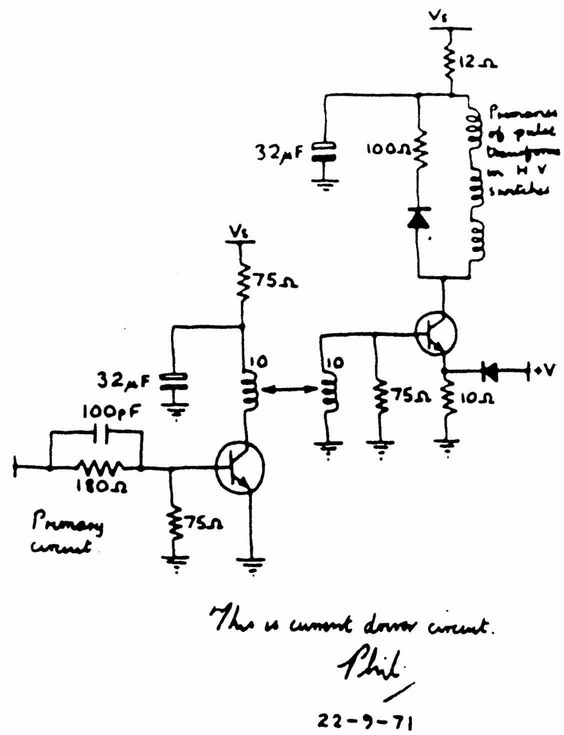
A7 - Chinese Text

A8 - Trademark Design

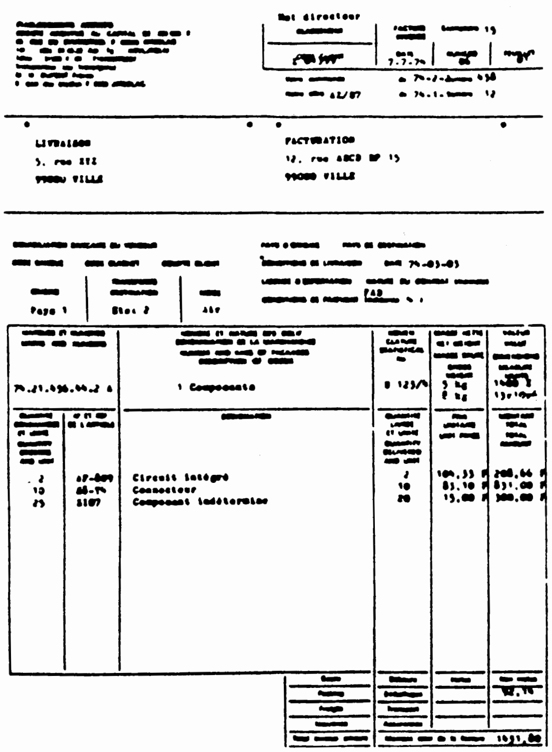
Figure 11-2. Original Test Images Used for IBC (A5 - A8)


THE SLEREXE COMPANY LIMITED

Document 1



Document 2

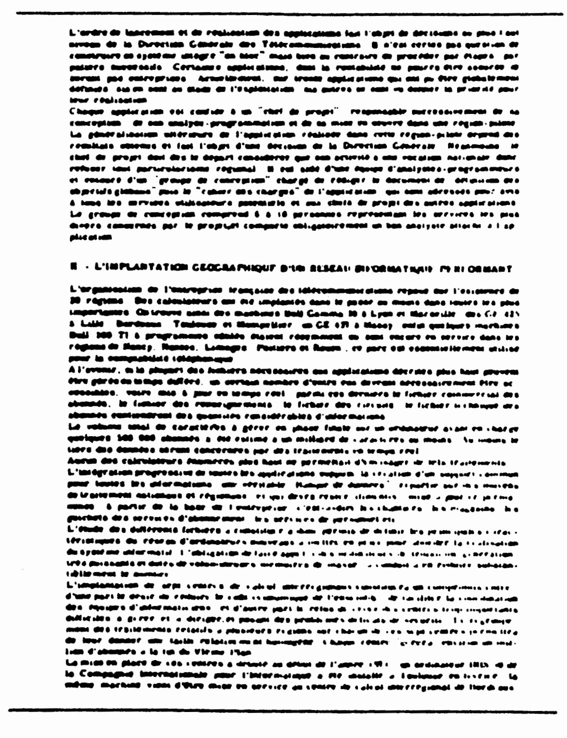

LIVRAISON

FACTURATION

Circuit intégré

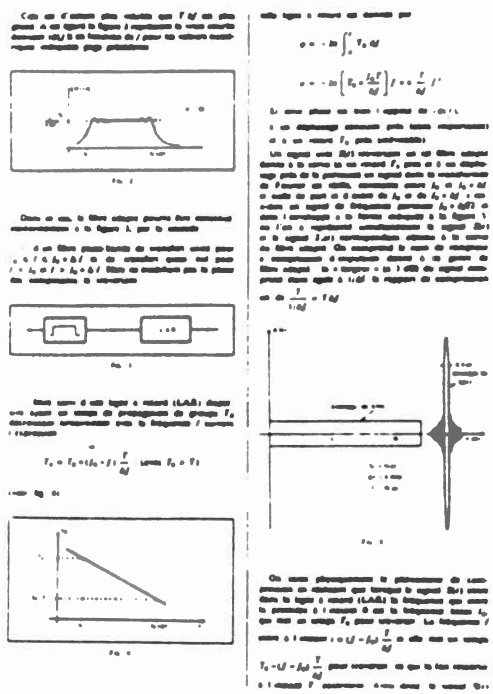
Connecteur

Document 3

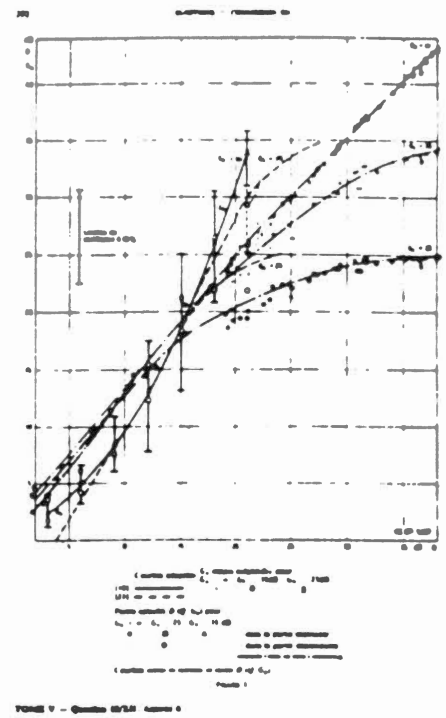


Document 4

Figure 12-1. CCITT Test Documents (Document 1 - 4)

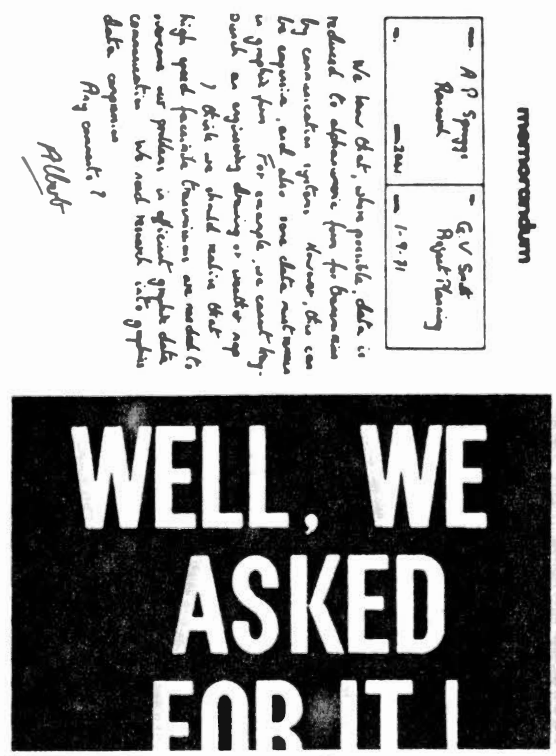


Document 5



Document 6

Document 7

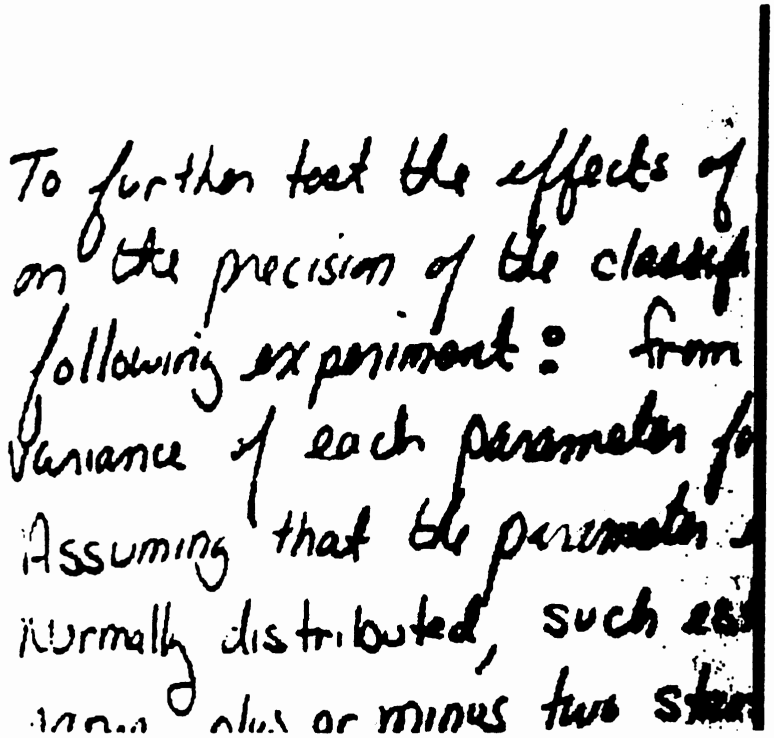


Document 8

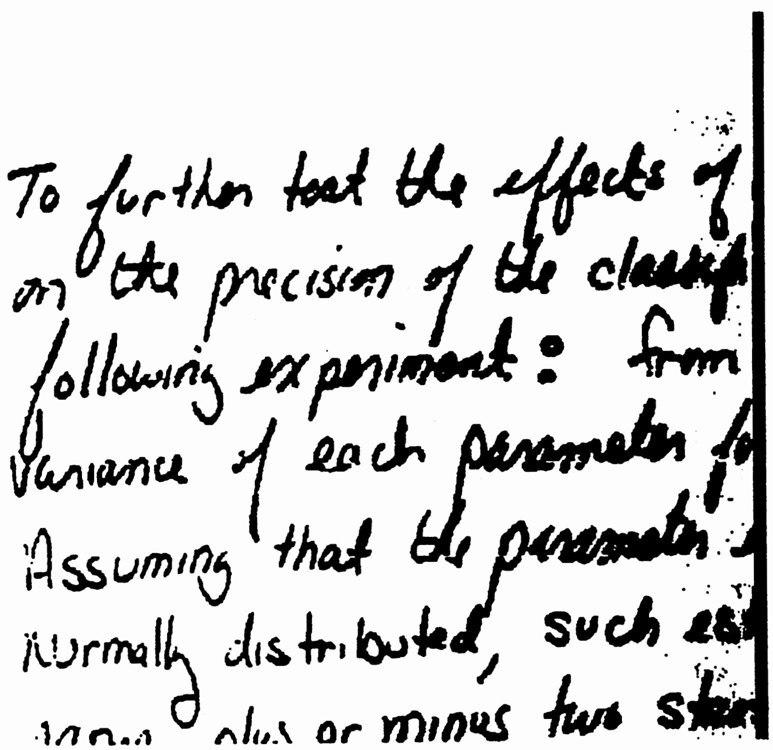
Figure 12-2. CCITT Test Documents (Document 5 - 8)

Interleaved Block Coding Method

Figure 13-1 to -8 show the original test sections of documents and simulation results for Interleaved Block Coding in field size F =160. Comparing with original documents, the reconstructed images are pretty good, especially when an image contains large white or large black spaces (see Fig. 13-2, -6 and -8). This is due to the number of **WW, BW** and **WB** block-pairs in the coding field being enough to transmit the NN block-pairs. The subjective effect of blanking of blocks in overflow fields may be examined in Fig. 13-4 and -7 where the images A4 and A7 are densely typed in English and Chinese texts, respectively. Inter-word and inter-letter white spaces of dense text are obviously by themselves unable to furnish enough **WW, BW** and **WB** pairs of blocks to convey the abundant right blocks of nonwhite-nonwhite block-pairs in the field. However, it is evident, that the intelligibility of characters is mainly unaffected by this kind of distortion. This is due to the fact that in the event of blanking, NN pairs will always have the right block only deleted, with the left block intact, and for a series of such NN block-pairs in an active region, left nonwhite blocks remain which are sufficient to preserve the form and legibility of characters. Thus, even in the worst situation when all the NN block-pairs in the field have their right blocks deleted, intelligibility is preserved largely due to retention of left nonwhite blocks. Generally, the loss of blocks is quite visible even though the percentage of such losses over the total number of blocks in the image may be very small. The human eye is quite sensitive to loss of black elements in the image. A modified **IBC** technique improves this coding method.

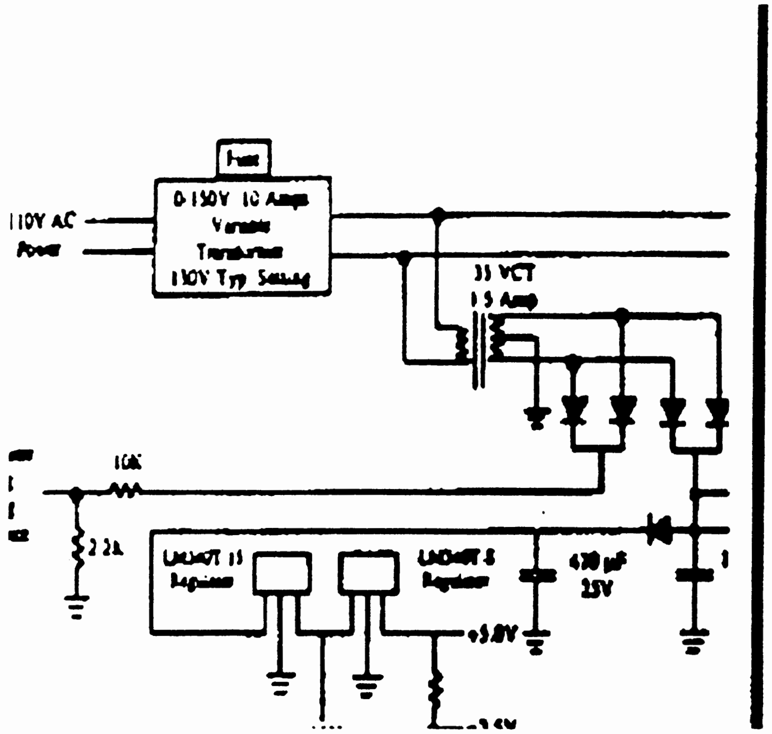


(a) Original

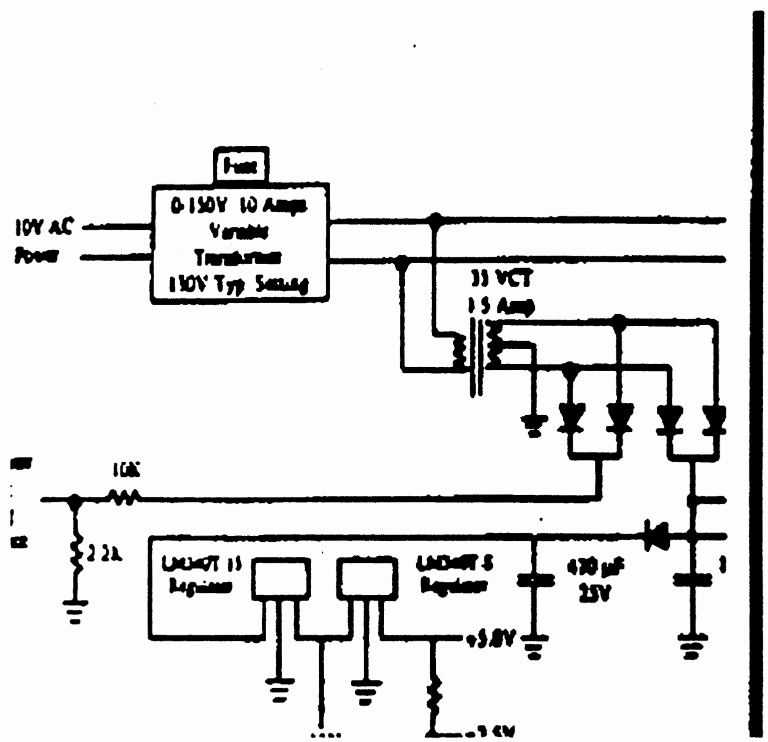


(b) Reconstructed

Figure 13-1. IBC Simulation Results - A1 Section, F = 160

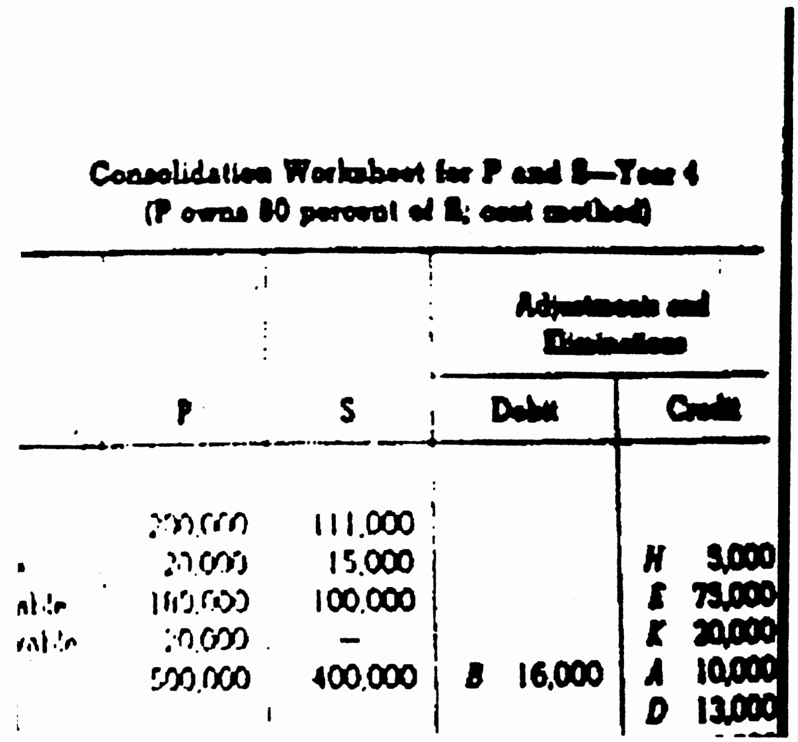


(a) Original

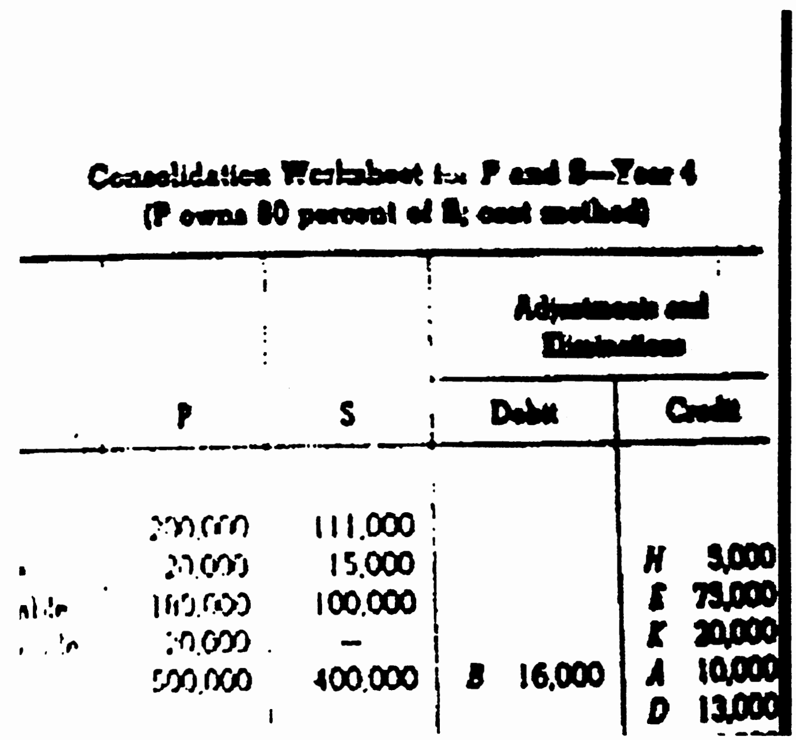


(b) Reconstructed

Figure 13-2. IBC Simulation Results - A2 Section, F = 160

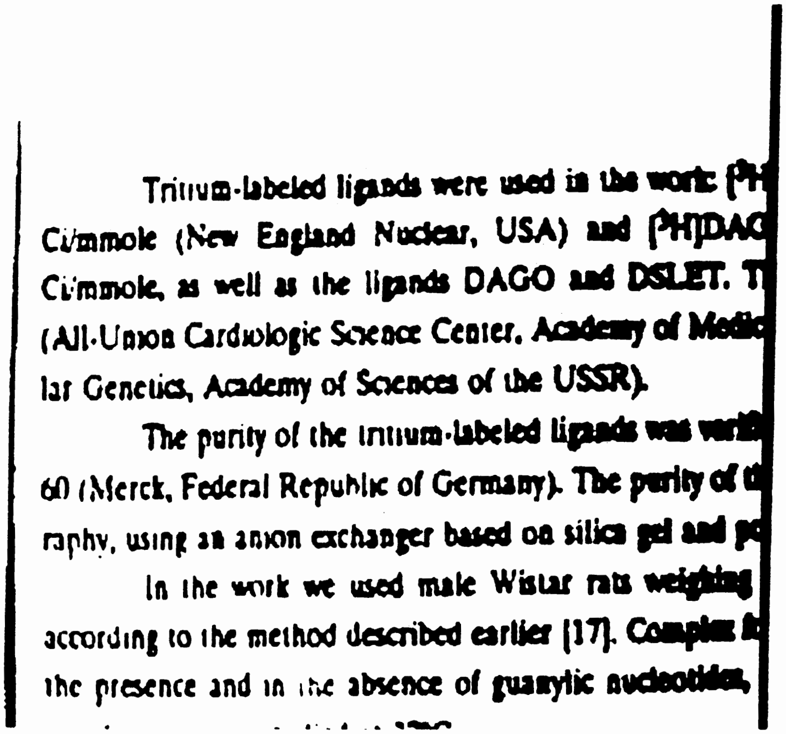


(a) Original



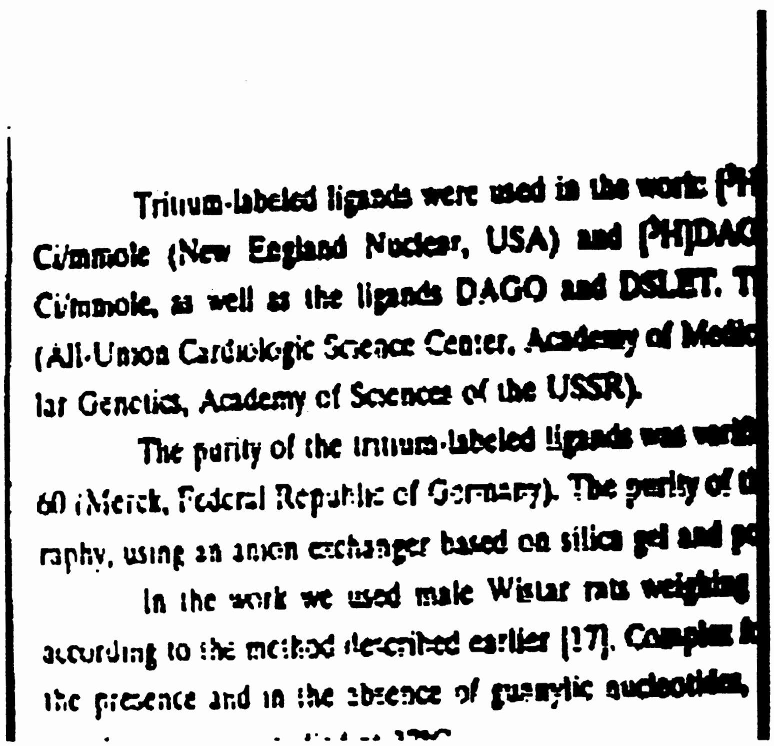
(b) Reconstructed

Figure 13-3. IBC Simulation Results - A3 Section, F = 160



Tritium-labeled ligands were used in the work: [^{3}H
Ci/mmole (New England Nuclear, USA) and [^{3}H]DAG
Ci/mmole, as well as the ligands DAGO and DSLET. T
(All-Union Cardiologic Science Center, Academy of Medic
lar Genetics, Academy of Sciences of the USSR).
The purity of the tritium-labeled ligands was verifi
60 (Merck, Federal Republic of Germany). The purity of th
raphy, using an anion exchanger based on silica gel and po
In the work we used male Wistar rats weighing
according to the method described earlier [17]. Complex fo
the presence and in the absence of guanylic nucleotides,

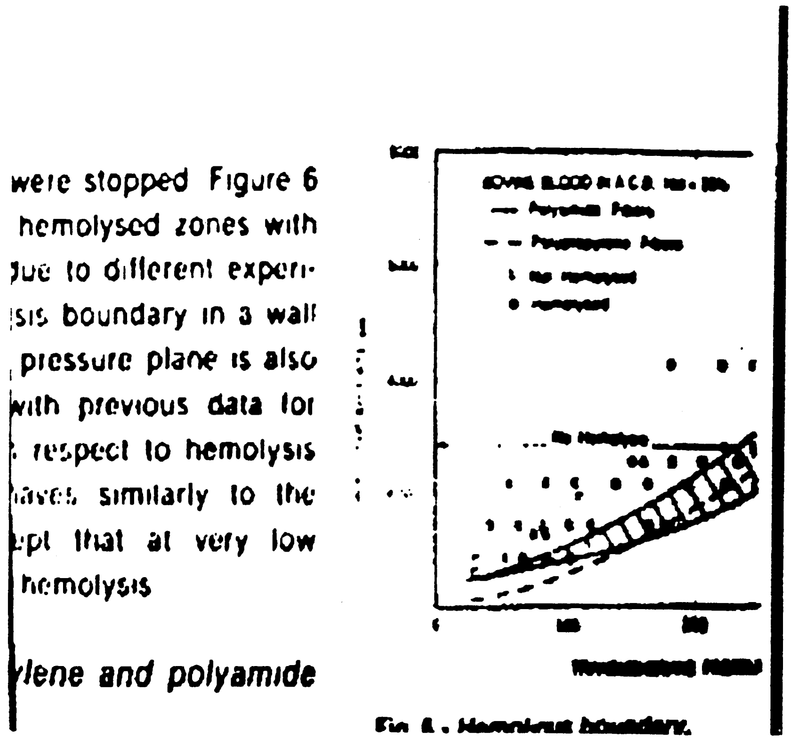
(a) Original



Tritium-labeled ligands were used in the work: [^{3}H
Ci/mmole (New England Nuclear, USA) and [^{3}H]DAG
Ci/mmole, as well as the ligands DAGO and DSLET. T
(All-Union Cardiologic Science Center, Academy of Medic
lar Genetics, Academy of Sciences of the USSR).
The purity of the tritium-labeled ligands was verifi
60 (Merck, Federal Republic of Germany). The purity of th
raphy, using an anion exchanger based on silica gel and po
In the work we used male Wistar rats weighing
according to the method described earlier [17]. Complex fo
the presence and in the absence of guanylic nucleotides,

(b) Reconstructed

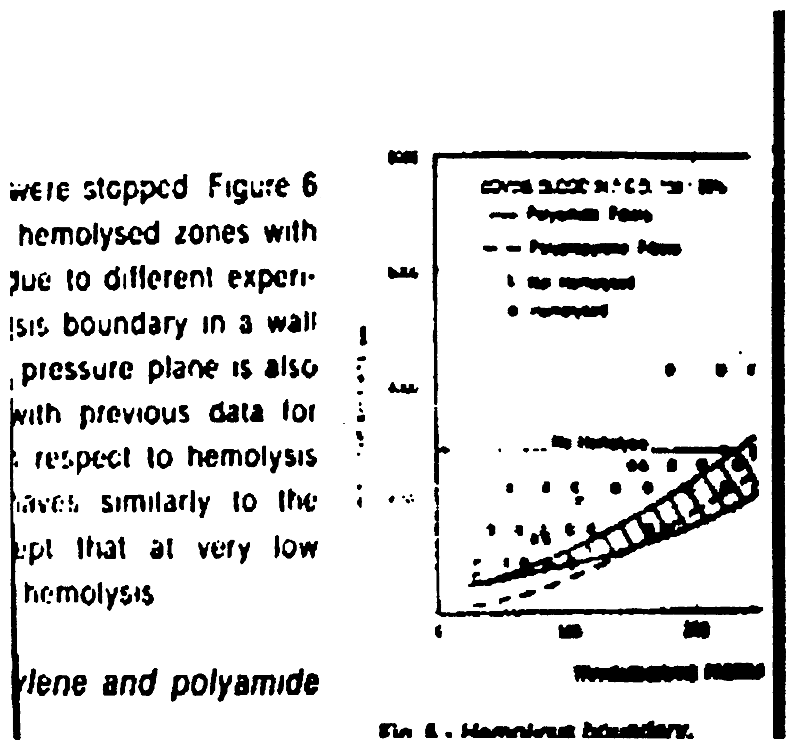
Figure 13-4. IBC Simulation Results - A4 Section, F = 160



were stopped Figure 6
hemolysed zones with
due to different experi-
sis boundary in a wall
pressure plane is also
with previous data for
respect to hemolysis
haves similarly to the
ept that at very low
hemolysis

lene and polyamide

(a) Original

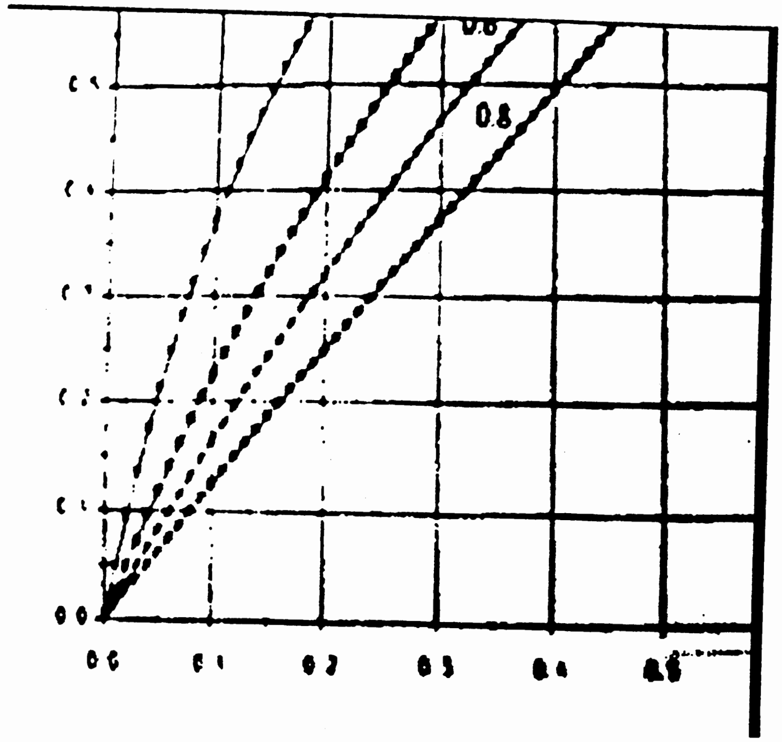


were stopped Figure 6
hemolysed zones with
due to different experi-
sis boundary in a wall
pressure plane is also
with previous data for
respect to hemolysis
haves similarly to the
ept that at very low
hemolysis

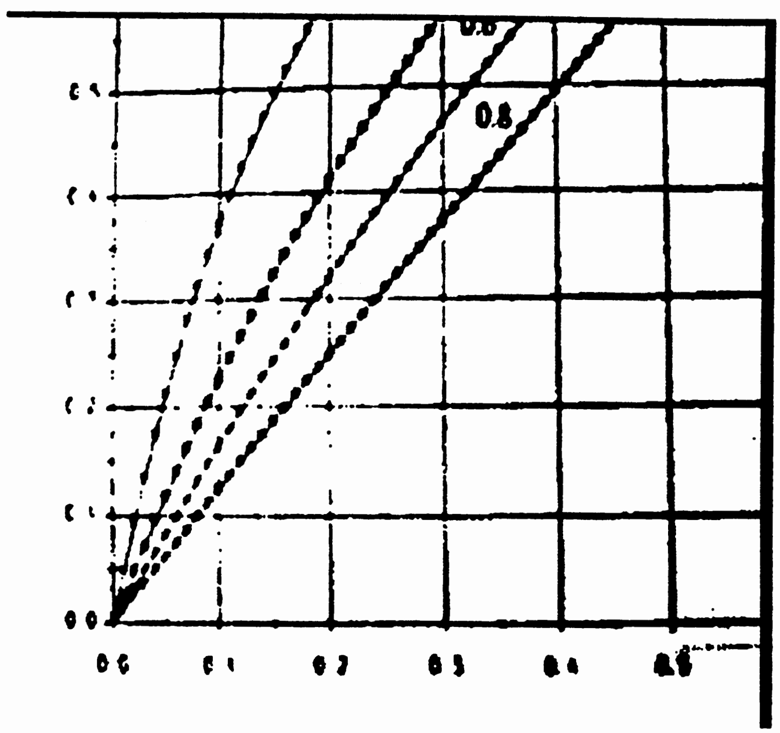
lene and polyamide

(b) Reconstructed

Figure 13-5. IBC Simulation Results - A5 Section, F = 160

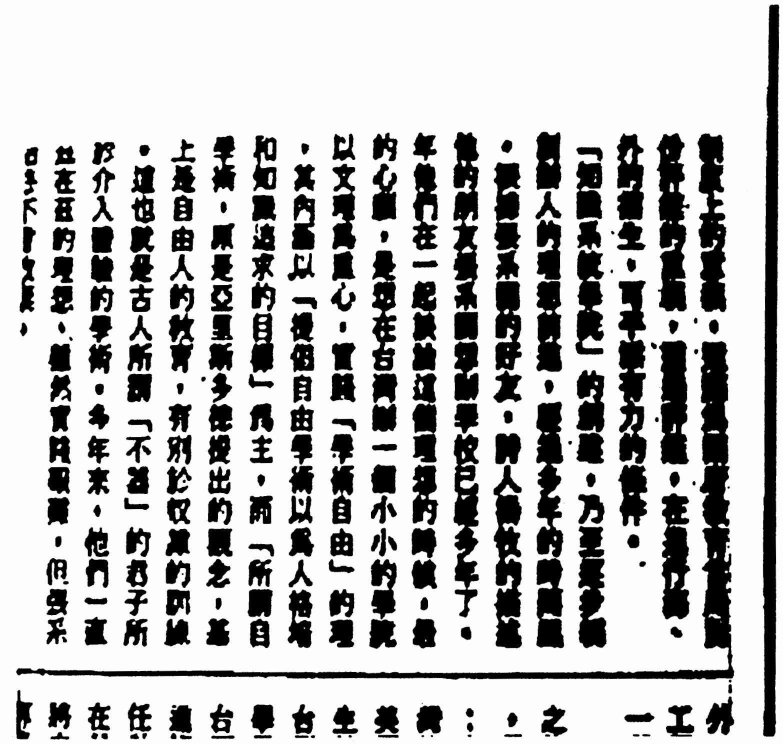


(a) Original

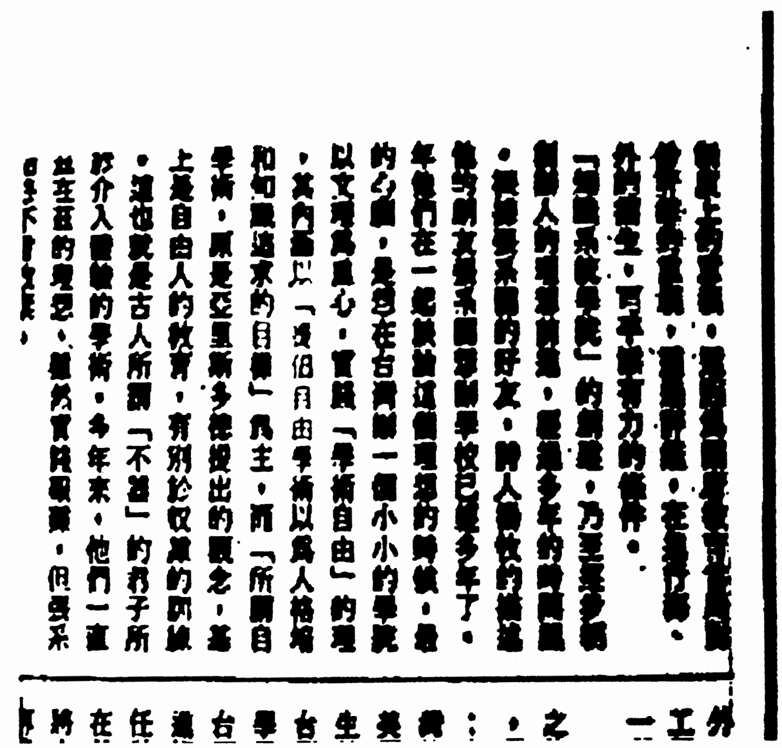


(b) Reconstructed

Figure 13-6. IBC Simulation Results - A6 Section, F = 160

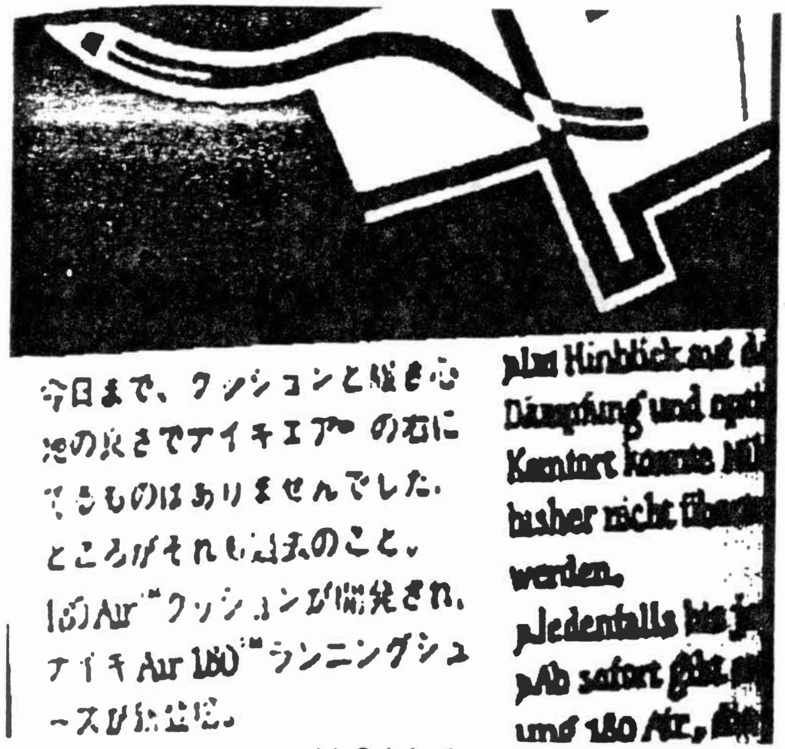


(a) Original

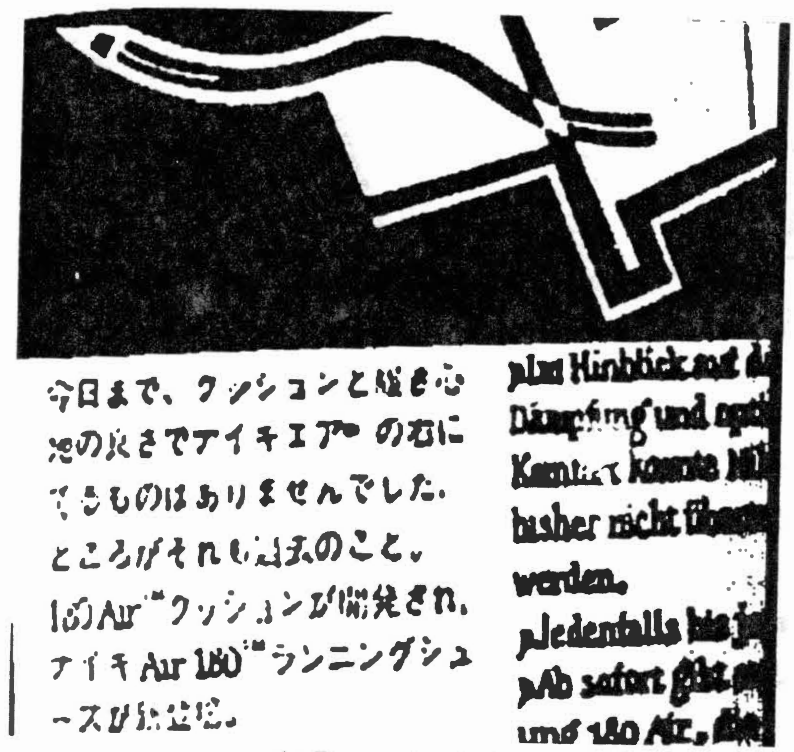


(b) Reconstructed

Figure 13-7. IBC Simulation Results - A7 Section, F = 160



(a) Original



(b) Reconstructed

Figure 13-8. IBC Simulation Results - A8 Section, F = 160

Modified Interleaved Block Coding Method

In order to reduce the amount of encoding distortion, a simple strategy may be adopted. Instead of omitting the last-occurring nonwhite blocks of the field, we use the color correlation of pixels in a block-pair source to reconstruct the image; i.e., for two side-by-side pixels, the latter is always the same color as the former. The procedure is as follow.

In the first step of the coding procedure, when a NN block-pair is detected, its even number pixels (2, 4, 6, 8, 10, 12, 14, 16 pixels in the simulation) are extracted as a N -bit block pattern and stored into the FIFO buffer. The encoding step for a NN block-pair state, the odd number pixels are the only choice as a transmitted block pattern since the even number pixels have been fed into the buffer . There is no change for all other states either in the coding procedure or in the decoding procedure. At the end of decoding step for a NN block-pair, the algorithm resotres the odd number-pixel block pattern which is transmitted by the pair itself, and restoreds even number-pixel block pattern which is taken from FIFO buffer according to their original odd or even numbers.

The procedures of the Modified IBC described above occur with the non-overflow condition. When the overflow condition happens, the even number-pixel block patterns in the FIFO buffer are not all transmitted by WW, BW and WB block-pairs; thus some NN block-pairs will only have odd number-pixel block pattern left at the end of the decoding procedure. Instead of substituting the all-white pattern at the receiver to lose black elements, we simply make a copy of the odd number-pixel block pattern as the even number-pixel block pattern, then reinsert these two patterns which have odd number and even number pixels respectively into their original positions. Figure 14 shows an example of Modified IBC under overflow condition.

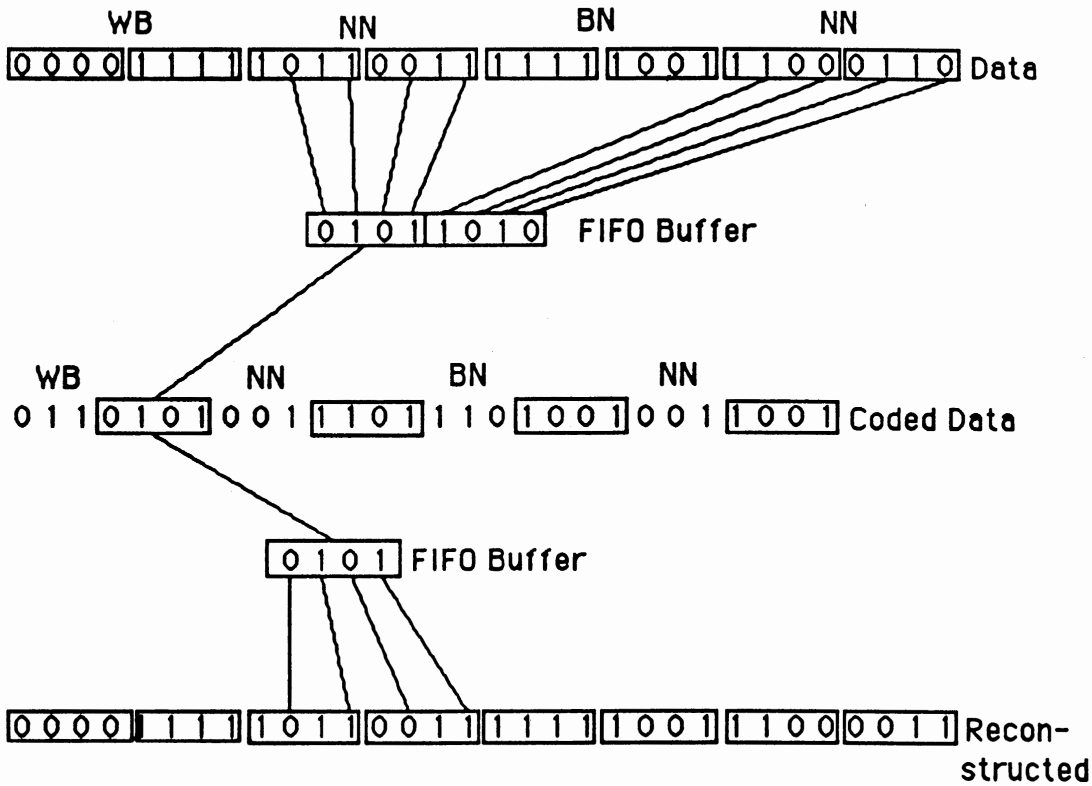
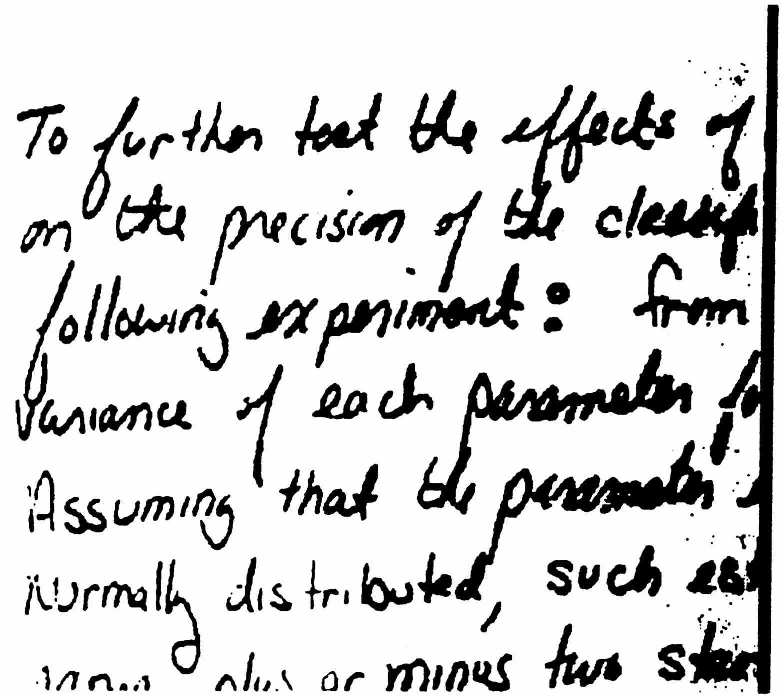
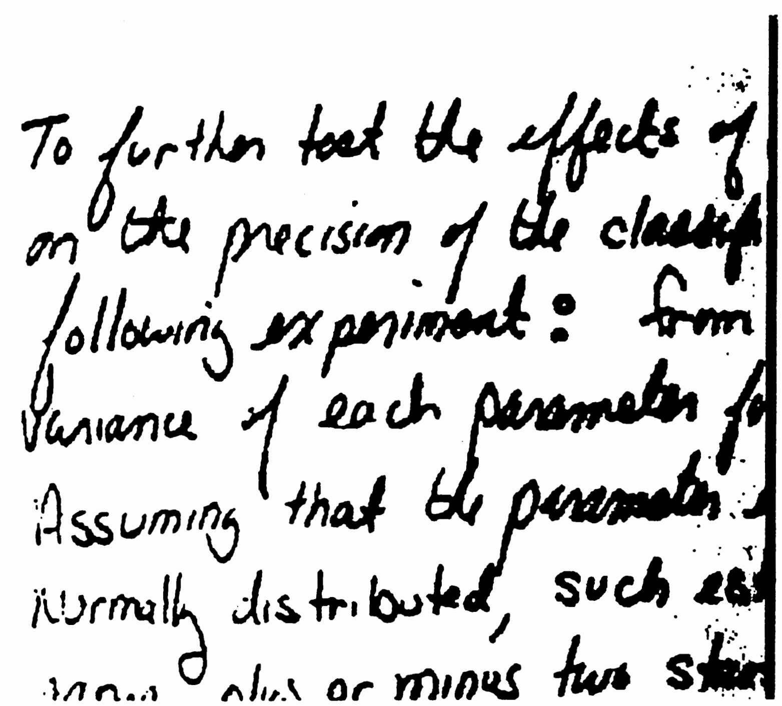


Figure 14. Example of Modified IBC, 4-bit Block, Overflow

The simulation results of the modified IBC with a field size, F=64, are represented in Figure 15-1(b) to -2(b). Comparing these with the results given in Figure 15-1(a) to -2(a) for the IBC, the improvement in the distortionless transmission is significant. This is expected because that instead of blanking the blocks, we take advantage of color correlation between two connected pixels within a document by duplicating the pixel. However, the modified IBC scheme exhibits a large improvement in distortionless and is still a simple, efficient coding method.

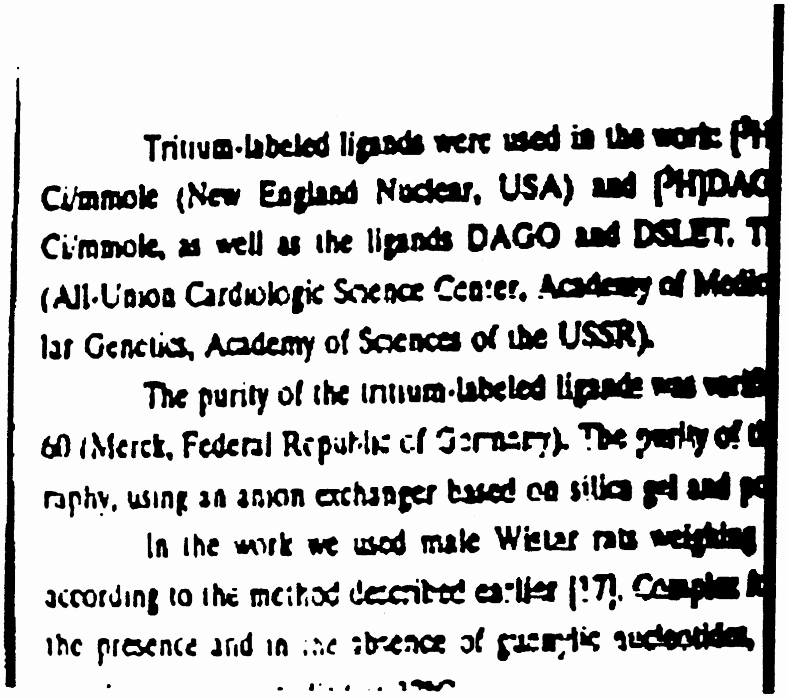


(a) IBC Technique



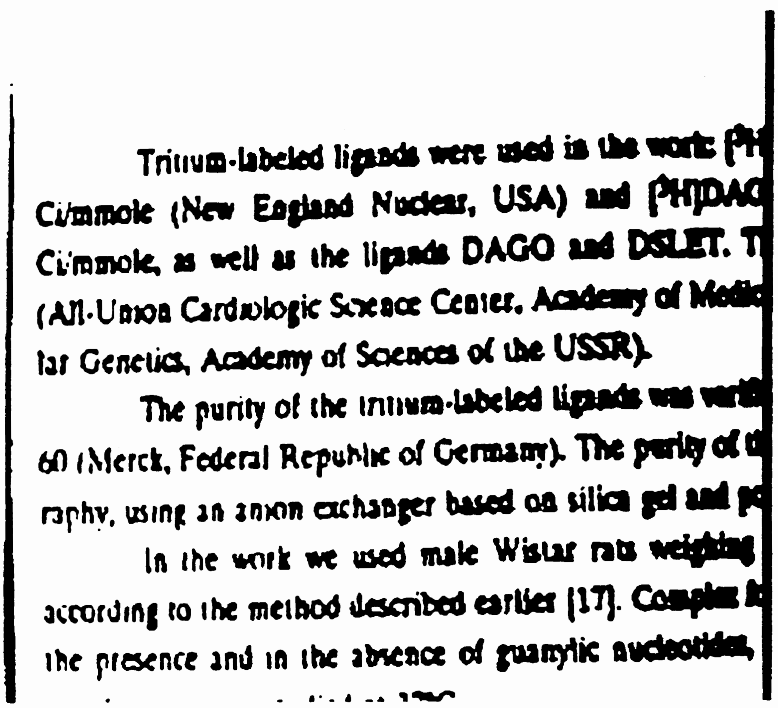
(b) Modified IBC Technique

Figure 15-1. IBC and Modified IBC Simulation Results-A1 Section, F=64



Tritium-labeled ligands were used in the work: [^{3}H
Ci/mmole (New England Nuclear, USA) and [^{3}H]DAG
Ci/mmole, as well as the ligands DAGO and DSLET. T
(All-Union Cardiologic Science Center, Academy of Medic
lar Genetics, Academy of Sciences of the USSR).
The purity of the tritium-labeled ligands was verif
60 (Merck, Federal Republic of Germany). The purity of t
raphy, using an anion exchanger based on silica gel and p
In the work we used male Wistar rats weighing
according to the method described earlier [17]. Complex f
the presence and in the absence of guanylic nucleotides,

(a) IBC Technique



Tritium-labeled ligands were used in the work: [^{3}H
Ci/mmole (New England Nuclear, USA) and [^{3}H]DAG
Ci/mmole, as well as the ligands DAGO and DSLET. T
(All-Union Cardiologic Science Center, Academy of Medic
lar Genetics, Academy of Sciences of the USSR).
The purity of the tritium-labeled ligands was verif
60 (Merck, Federal Republic of Germany). The purity of t
raphy, using an anion exchanger based on silica gel and p
In the work we used male Wistar rats weighing
according to the method described earlier [17]. Complex f
the presence and in the absence of guanylic nucleotides,

(b) Modified IBC Technique

Figure 15-2. IBC and Modified IBC Simulation Results-A4 Section, F=64

CHAPTER V

PROBABILITY OF FIELD OVERFLOW FOR A MEMORYLESS BLOCK SOURCE

It obviously is useful to have some idea of the likelihood of lossless performance based on some mathematical models, from a knowledge of the block statistics of the source. In this chapter the probability of overflow occurring in a coding field is determined as a function of the field size F. The image source is assumed to be an independent block source model in which the output can taken on one of 2^N states, where N is the number of pixels in the block. This source is therefore characterized by the set of probabilities

$$P_i \quad 0 \le i \le 2^N-1$$

where P_i is the probability of occurrence of the binary pattern i and

$$\sum_{i=0}^{2^N-1} P_i = 1$$

When a pair of blocks is taken the probabilities of block-pair states are given by

$$P\text{ (white-white)} = P_{WW} = (P_0)^2 \tag{3}$$

$$P\text{ (black-white)} = P_{BW} = P_{2^N-1} \times P_0 \tag{4}$$

$$P\text{ (nonwhite-white)} = P_{NW} = (1-P_0 - P_{2^N-1}) \times P_0 \tag{5}$$

$$P\text{ (white-black)} = P_{WB} = P_0 \times P_{2^N-1} \tag{6}$$

$$P\text{ (white-nonwhite)} = P_{WN} = P_0 \times (1-P_0 - P_{2^N-1}) \quad (7)$$

$$P\text{ (nonwhite-nonwhite)} = P_{NN} = (1-P_0 - P_{2^N-1})^2 \quad (8)$$

$$P\text{ (nonwhite-black)} = P_{NB} = (1-P_0 - P_{2^N-1}) \times P_{2^N-1} \quad (9)$$

$$P\text{ (black-nonwhite)} = P_{BN} = P_{2^N-1} \times (1-P_0 - P_{2^N-1}) \quad (10)$$

$$P\text{ (black-black)} = P_{BB} = (P_{2^N-1})^2 \quad (11)$$

and also

$$P_{WW} + P_{WB} + P_{NW} + P_{WB} + P_{WN} + P_{NN} + P_{NB} + P_{BN} + P_{BB} = 1 \quad (12)$$

For a finite coding field size F the probability P_F of a field overflowing can be obtained from the total probability of the field outcomes in which the number of **NN** states y is greater than the sum of **WW, BW** and **WB** states x .

Then

$$P_F = P(y > x)$$

$$= \sum_{i=1}^{F} \sum_{j=0}^{i-1} P(y = i, x = j) \quad \text{where } i+j \le F \quad (13)$$

The summation term can be evaluated using the multinomial theorem, given

$$P_F = \sum_{i=1}^{F} \sum_{j=0}^{i-1} \frac{F!}{i!\,j!\,(F-i-j)!} \times (P_{NN})^i (P_C)^j (1 - P_{NN} - P_C)^{(F-i-j)}$$

$$\text{where } P_C = P_{WW} + P_{BW} + P_{WB} \quad (14)$$

By substituting eqns. 8, 3, 4 and 6 into eqn. 14 the probability of overflow may be calculated from a knowledge of only the two probabilities P_0 and P_{2^N-1} which are of the N -bit all-white and all-black patterns, respectively.

Using the measured values of P_0 and P_{255} (all-white and all-black probability for 8-pixels blocks, respectively) for the test images, shown in TABLE VI, the results for P_F as calculated from eqn. 14 are plotted in Figure 16. TABLE VII shows the value of F when the probability of field overflow becomes less than 10^{-6}, a value which is probably sufficiently small to give lossless performance in a practical coding scheme. It is seen that the worst-case image A7 requires a field size of 100 block-pairs to achieve this value of P_F . At the other extreme, the largely-white image of A2 requires only 16 block-pairs to give the same value of probability. Of the images sampled at 150 ppi the largest field size of 60 block-pairs is demanded by the densely-typed image A4. In designing a compression system, the worst-case must be considered. For this set of test images, error-free coding for all images is therefore likely to be obtained using a field size of 100 block-pairs.

Comparing with the computed value F =49, which is sufficiently to give lossless performance, the reconstructed images of A1 in field sizes F =64 (see Fig. 13-1(b)) and F=160 (see Fig. 15-1(a)) are still shown the encoding distortion. The disparity between theoretical and experimental figures is evident and reflects the mismatch of the real image source from the statistically-stationary memoryless block model employed in the computations.

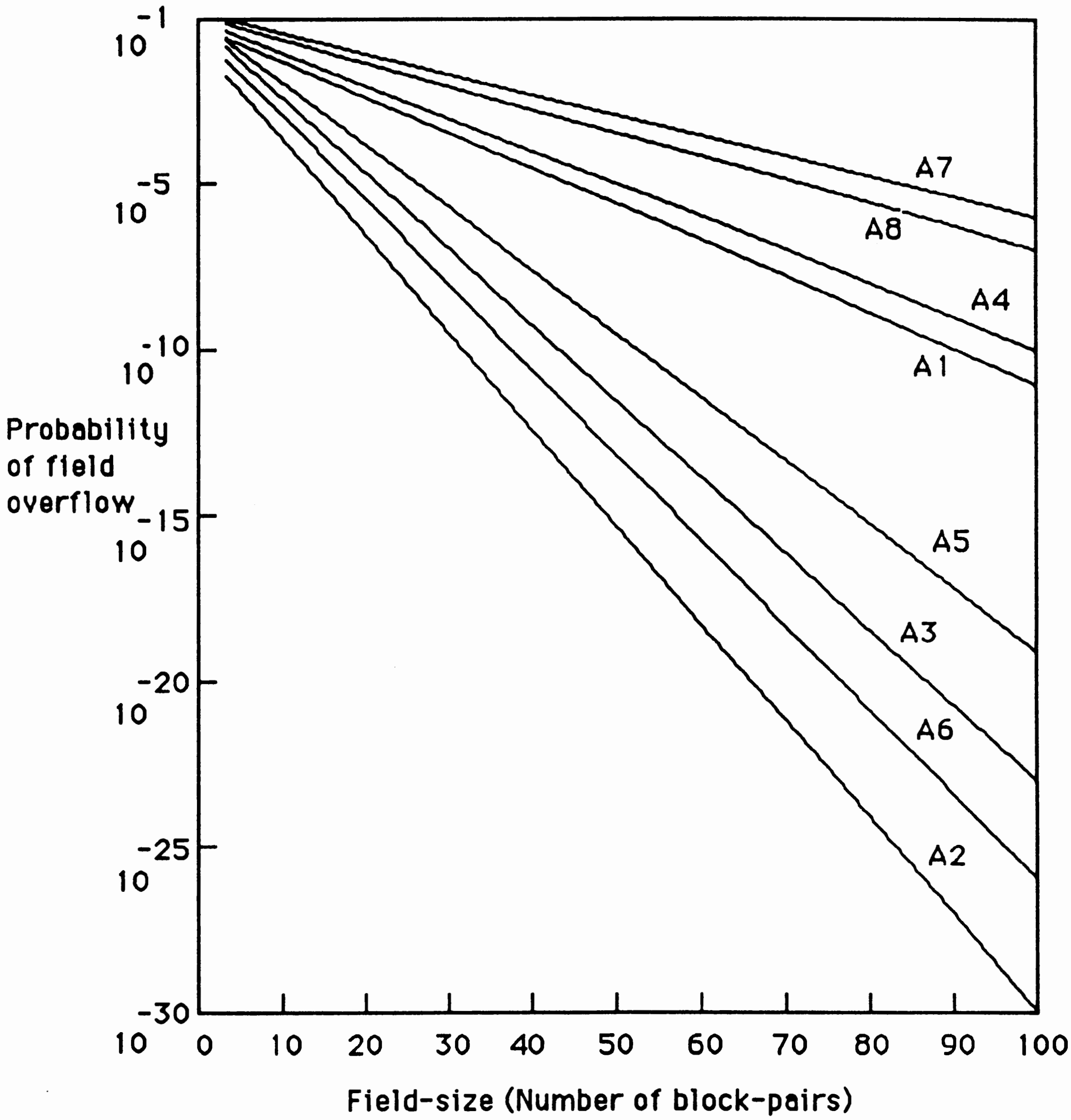


Figure 16. Probability of Overflow P_F Against Field Size F for Test Images Assuming a Memoryless Block Source

TABLE VII

FIELD SIZE AT WHICH P_F BECOMES LESS THAN 10^{-6}

IMAGE	F
A1-Handwritten text	49
A2-Circuit diagram	16
A3-Financial statement	23
A4-Typewritten text	60
A5-Sample page	27
A6-Plotting figure	20
A7-Chinese text	100
A8-Trademark design	88

CHAPTER VI

EFFECTS OF TRANSMISSION ERRORS

IBC and Modified IBC Techniques

The performance of the **IBC** method has hitherto been described on the assumption of error-free transmission conditions. In the event of transmission errors the effect depends on which type of bit have been corrupted. If the bits of the block pattern are incorrectly received, the reconstructed pattern shows out the errors on a pixel-by-pixel basis, i.e. a single corrupted bit produces a single reconstructed pixel error. The facsimile image can tolerate a substantial number of this type of errors without loss of intelligibility, especially if they are sufficiently isolated from one another. If the bits of the header word are incorrectly received, the corruptness of reconstructed image is more serious. A false report on the status of the current block-pair is the direct result. The most serious effect occurs when a **NN**, **WW**, **BW** or **WB** state header word is corrupted or generated. Then an incorrect block will be placed in the FIFO buffer, or a valid block may be omitted. Because the reinstatement of the FIFO buffer contents works on a sequential basis, the result can be the incorrect placement of all the blocks in the buffer. However, there is no total failure of the decoding process. **WW**, **BW**, **NW**, **WB**, **WN**, **NB**, **BN** and **BB** block-pairs may still be correctly reinserted. The left halves of nonwhite-nonwhite (**NN**) block-pairs will also be recovered without error for uncorrupted **NN** header states in the field. This aspect is of importance as it ensures that a large proportion of the image material is successfully reconstructed even if the sequential reinstatement of right blocks of nonwhite-nonwhite pairs has been adversely affected. Furthermore, as the proportion of header bits in the coded bit stream is small the IBC system is consequently able to withstand relatively large levels of transmission errors.

The effects of transmission errors were investigated via simulation by subjecting the coded output of the image A1 to controlled amounts of computer-generated random errors. Figure 17-1 to -3 show the reconstructed images for average error rates of 1 in 1000, 1 in 100 and 1 in 32, by using the IBC and the modified IBC respectively. From the experimental results, an excellent level of intelligibility is preserved at the rate of 1 in 1000. At the rate of 1 in 100 the text is totally readable and intelligibility is largely unimpaired. When the average error rate is increased to 1 in 32 the quality is degraded to some extent, but it is still possible to read and understand the text. Certain individual characters are unrecognizable but words and sentences can be made out from context. Anyway, from the studies of simulation and transmission errors, the modified IBC shows obviously the better results than the IBC.

Comparison with White Block Skipping Method

As mentioned in Chapter IV, the compression ratio of the IBC is not very high. TABLE VIII shows a comparison of the compression ratio achievable by the IBC and our original inspirational white block skipping (WBS) technique. In structured images like vertical densely-typed; e.g., Chinese and Japanese and largely-black documents, both methods achieve nearly ideal compression ratios. When a document contains large black portions like A8, the IBC even gets a little bit higher compression ratio than the WBS. However, it appears that for typical documents, the WBS is superior to the IBC in data compression.

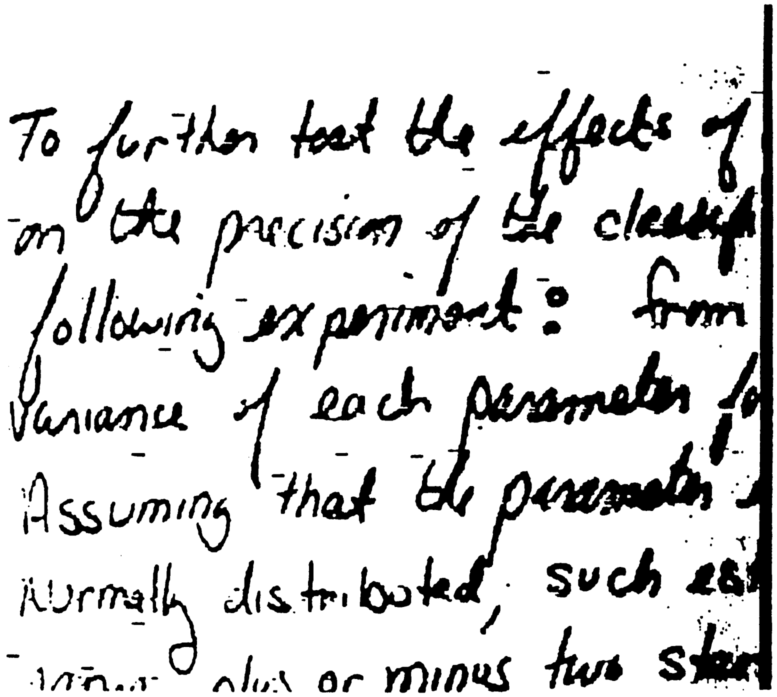
TABLE VIII

COMPRESSION RATIOS OF IBC AND WBS FOR 8-BIT BLOCK SIZE

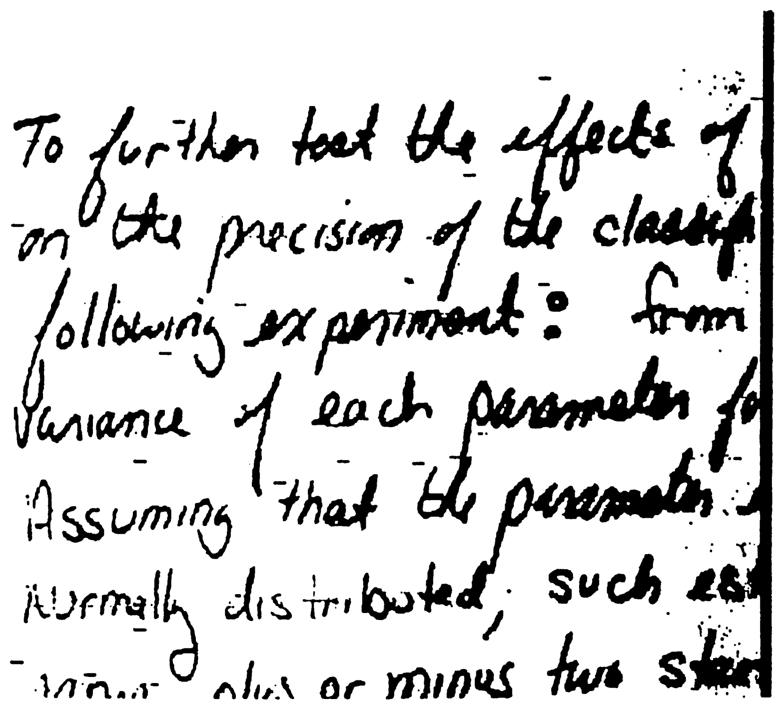
	A1	A2	A3	A4	A5	A6	A7	A8	Average
WBS	2.28	3.19	3.14	2.23	2.86	2.85	1.64	1.44	2.45
IBC	1.45	1.45	1.45	1.45	1.45	1.45	1.45	1.45	1.45

One penalty for efficient coding is increased sensitivity to channel errors. Figure 19 gives clear results of the effects of transmission errors for higher compression ratio techniques at a variable-rate output. For the WBS method, at the error rate of 1 in 1000, some words are recognizable, but sentences and context obviously cannot read from those words. At the rate of 1 in 100 and 1 in 32, the entire image is corrupt and the text is totally unreadable. When transmission errors occur, a higher the compression ratio makes the reconstructed image worse. For largely-white images such as A2 and A3, the WBS works especially well in data compression. A bit for all white block changing from 0 to 1 can cause loss of synchronization. Such an error causes the receiver to expect the wrong block pattern for the following block. As a result, the block pattern is reconstructed and located incorrectly; the error propagates to the subsequent blocks. Figure 19. illustrates the mismatch for A2 and A3 at the error rate 1 in 1000.

Compared to the WBS, the IBC has a lower compression ratio, but a higher image quality when channel noise occurs. As we expected, the fixed-rate output coding technique proves to have strong immunity to transmission errors.

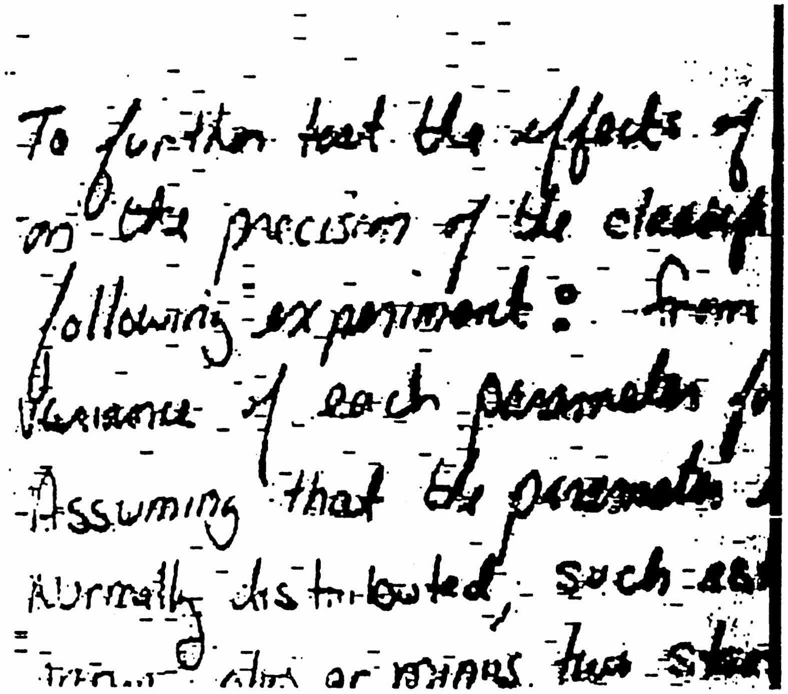


(a) IBC Technique-A1 Section

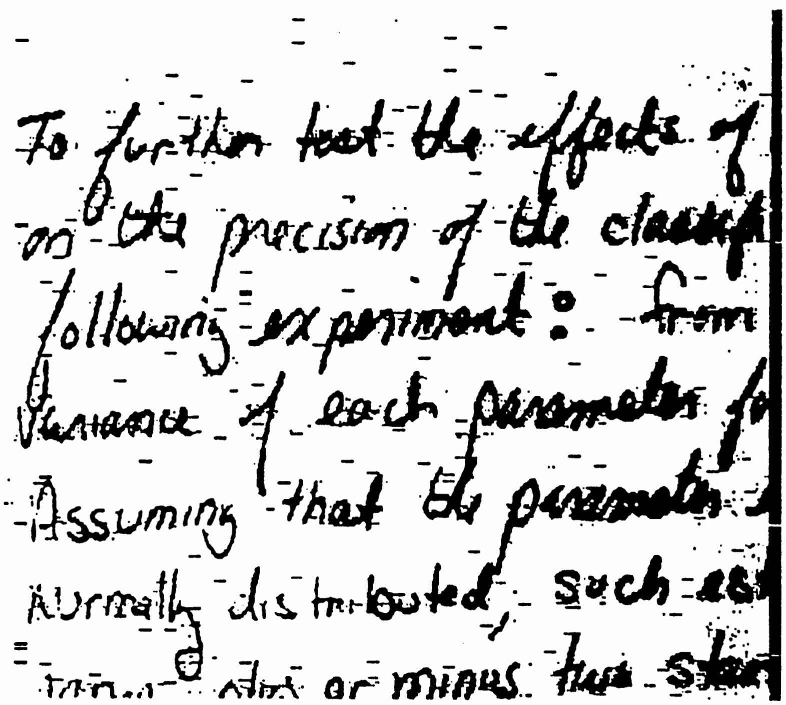


(b) Modified IBC Technique-A1 Section

Figure 17-1. Reconstructed Image in Presence of Random Transmission Errors for IBC and Modified IBC Techniques, F=64, Error Rate=1/1000

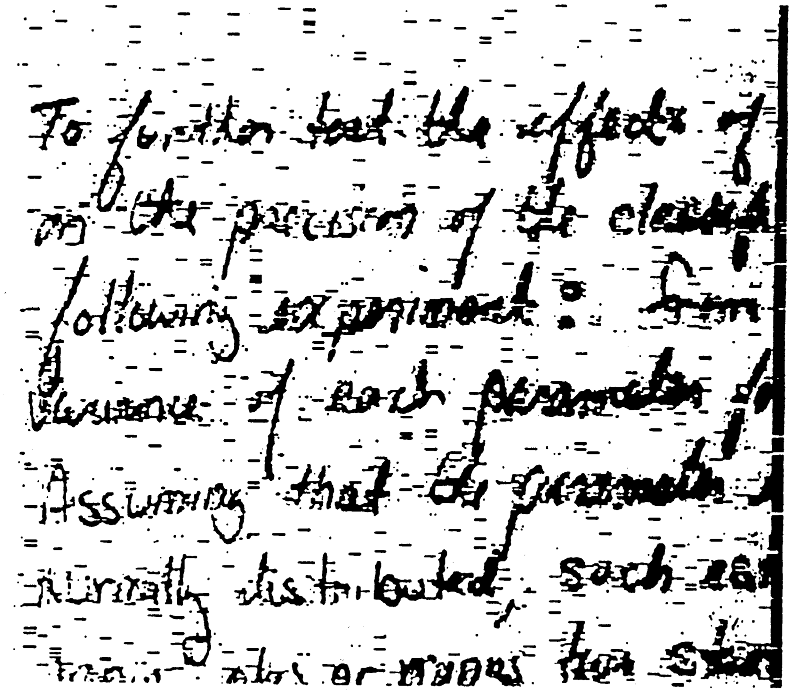


(a) IBC Technique-A1 Section

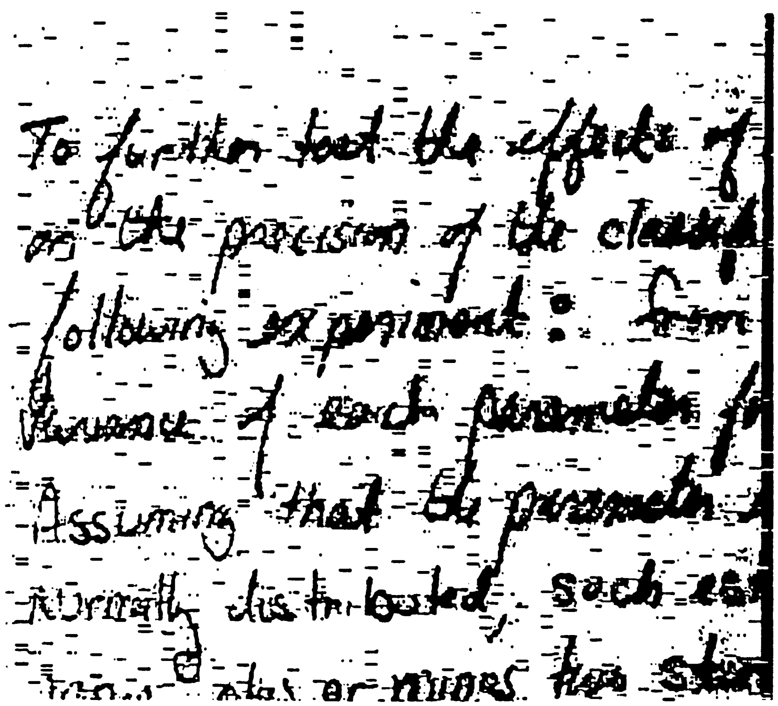


(b) Modified IBC Technique-A1 Section

Figure 17-2. Reconstructed Image in Presence of Random Transmission Errors for IBC and Modified IBC Techniques, F=64, Error Rate=1/100

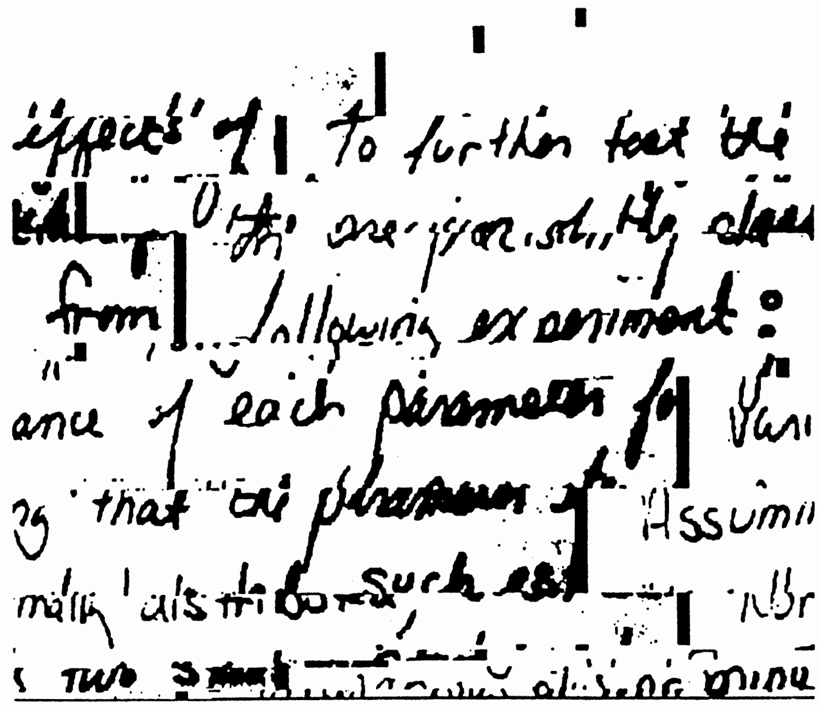


(a) IBC Technique-A1 Section

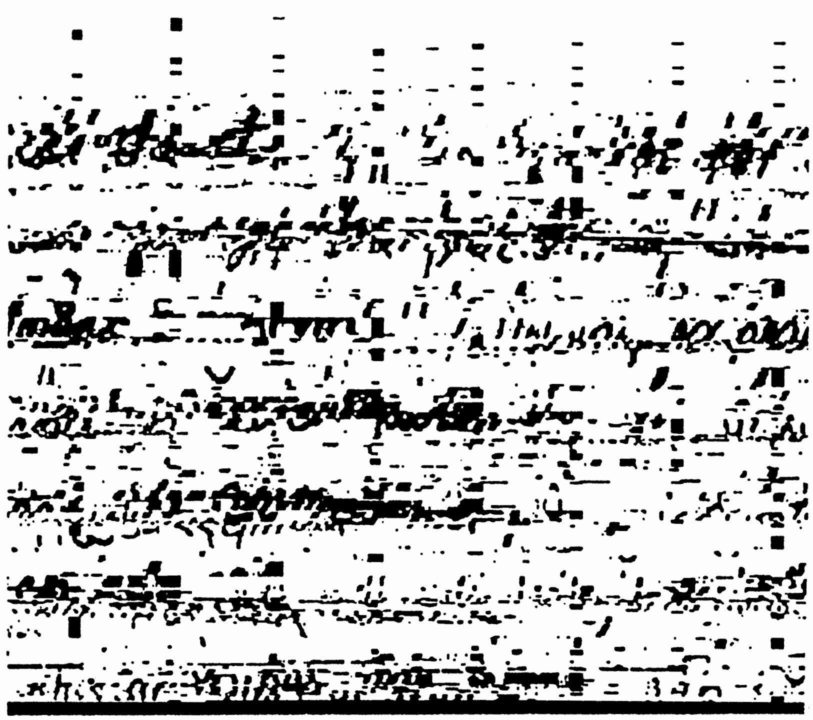


(b) Modified IBC Technique-A1 Section

Figure 17-3. Reconstructed Image in Presence of Random Transmission Errors for IBC and Modified IBC Techniques, F=64, Error Rate=1/32

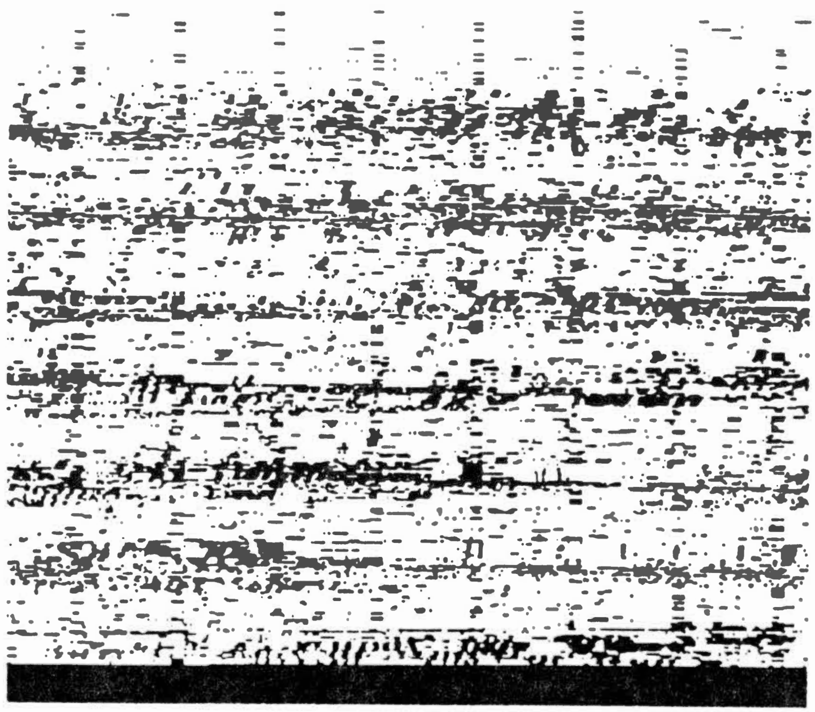


(a) WBS Technique-A1 Section, Error Rate=1/1000



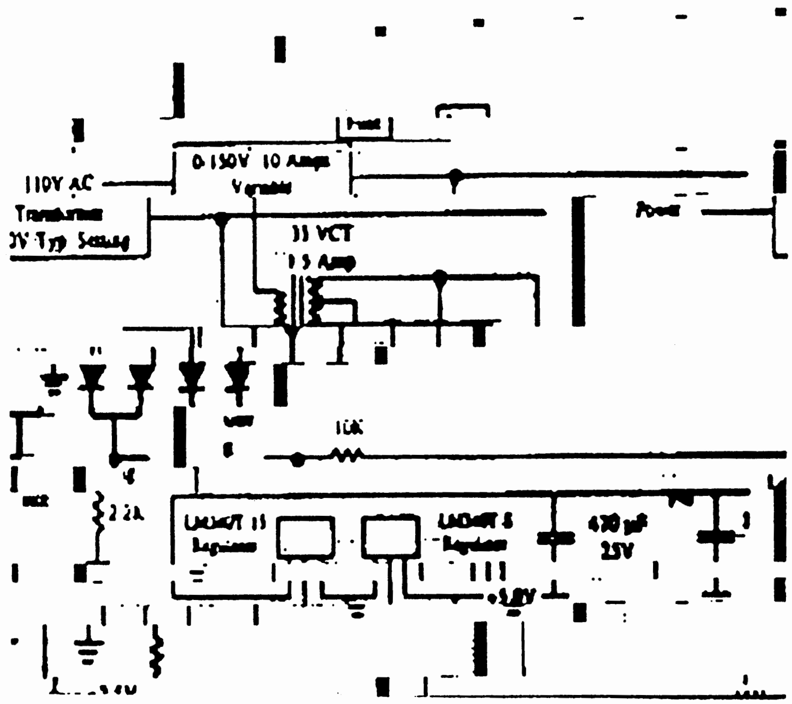
(b) WBS Technique-A1 Section, Error Rate=1/100

Figure 18. Reconstructed Image in Presence of Random Transmission Errors for WBS Technique, A1 Section

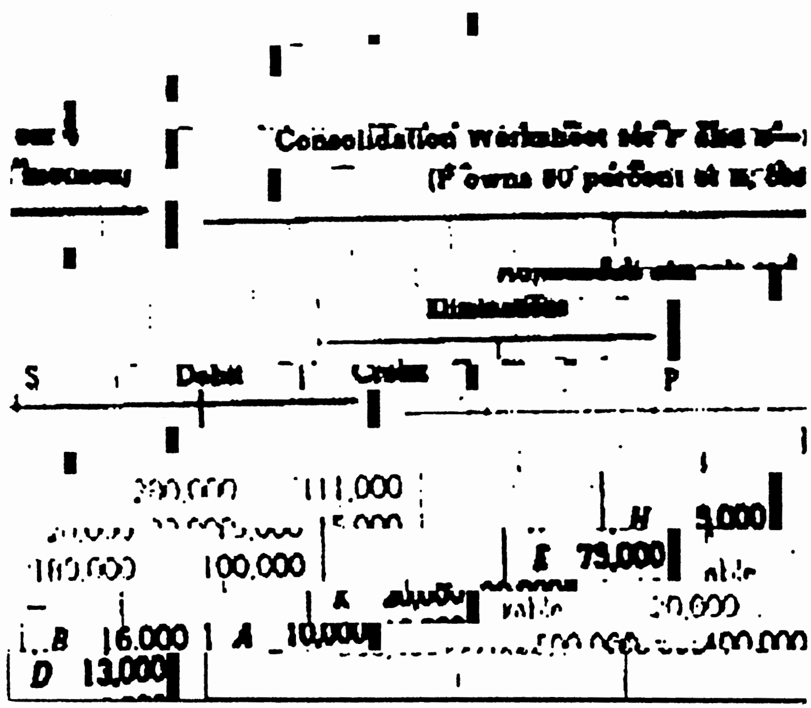


(c) **WBS** Technique-A1 Section, Error Rate=1/32

Figure 18. (continued)



(a) WBS Technique-A2 Section



(b) WBS Technique-A3 Section

Figure 19. Reconstructed Image in Presence of Random Transmission Errors for WBS Technique A2 and A3 Sections, Error Rate=1/1000

CHAPTER VII

CONCLUSIONS AND FUTURE RESEARCH SUGGESTIONS

A novel method of fixed-rate compression, has been presented in which the input image data is delayed and pairs of blocks are encoded in accordance with their white states. The Interleaved Block Coding (IBC) scheme ,which is similar to the white block skipping coding (WBS) in some ways but does better jobs in noise immunity, is very simple, efficient and is easy to implement. The Interleaved Block Coding method works well no matter when the image is largely-white or largely-black and the fixed-rate output exhibits a strong immunity to transmission errors. The method encodes block-pairs of binary image data without the need for codebooks. In the case where the distortion is due entirely to simple blanking of blocks, an extract procedure based on the color correlation between two connected pixels has been found to be effective for reducing the level of black element loss. Also, the results of simulation for 8-pixel blocks showed a high degree of intelligibility being maintained in the reconstructed image at the receiver, after exposure to random transmission errors whose average error rate are 1 in 1000 and 1 in 100, and at the rate of 1 in 32 the recovered image is largely readable. Being compared with other coding schemes, the compression ratio of the IBC is not very high, but as we said sometimes we are willing to sacrifice compression for ease of implementation and for decreasing the complexity of system. Thus, this coding scheme is preferred for low resolution (around 100 ppi) In the meanwhile, for some types of channels a more robust form of compression is demanded with lower levels of compression being an acceptable price for improved noise immunity. Examples include HF data links, aircraft communication channels and low-grade telephone circuits. In other words, the IBC has high image quality, high transmission errors immunity, low complexity and low compression capabilities. Also, in the IBC method, the 8 pixels block size is a good choice for implementation by using commercial digital ICs and eight-bit microprocessors.

The following suggestions are for future research.

(a) By employing a sufficient large coding field, a lossless form of coding is possible. In a practical implementation it is desirable to keep the field size as small as possible. Apart from increased amount of storage and hardware, an important reason is the effect of transmission errors. The results obtained in the Chapter V suggest the lossless coding can be achieved with smaller fields if the image is a true uncorrelated block source. Therefore, in order to attain lower permissible values of F for the coding scheme, a pre-processing operation of decorrelating blocks is the first priority in coding procedure. Interested readers are encouraged to do more research and experiments in decorrelating blocks before encoding.

(b) In the event of transmission errors the effect depends on which type of bit have been corrupted. If the bits of the blockpattern are incorrectly recieved, the reconstructed pattern shows out the errors on a pixel-by-pixel basis. If the header bits are corrupted, the block-pairs may be incorrectly reinserted image, especially when a **NN**, **NW**, **BW** or **WB** state header word is corrupted. For example, the **WW** headword is 000 and the **BB** headword is 001. If a **WW** header is incorrectly generated from 000 to 001, it will cause a valid block to be omitted (the result can be the incorrect placement of all the blocks in the buffer) and an incorrect block-pair to be reinserted (from all-white to all-black). So, to find out the best choice of the combinations of three bits to identify the nine states header is useful for avoiding serious corrupting of reconstructed images.

(c) Techniques which seek to minimize the effects of transmission errors of the **WBS** are also interesting. One method has been considered: restriction of the damage caused by errors to as small an area as possible. In order to reduce the effects of error propagation, a smaller coding field is advantageous. Hence, each coding field requires a codeword to represent the end of coding field since the WBS features a variable-rate output. The compression ratio of the method decreases with increasing EOF (end of field) codeword. Thus, the choice of coding field size and EOF codeword need to be discussed in more detail in order of keeping simplicity and efficiency of the WBS technique.

REFERENCES

[Arps 80] Arps., R. B. , "Bibliography on Binary Image Compression," Proc. IEEE Vol.68, no.7, pp. 922-924, July 1980.

[Capon 59] Capon, J., "A Probabilistic Model for Runlength Coding of Pictures," IRE Trans. on Inf. Theo., pp.157-636, Dec. 1959.

[CCITT no. 7 76] Graphic Sciences, Inc. et al., (no title, One-Dimensional Run Length Code Proposed for Standardization), CCITT Study XIV, temp. codum. no.7, Sept. 1976.

[CCITT no.11 76] Federal Republic of Germany, "Relations between Transmission Time, Transmission Bit Rate, Maximum Line Feed Speed, and Buffer Memory Size in Run Length Coded Facsimile Systems," CCITT Study temp. docum. no. 11, Sept. 1976.

[CCITT no. 42 78] CCITT Study Group XIV, Japan, "Proposal for Draft Recommendation of Two-Demensional Coding Scheme," Doc. 42-E, 1978.

[CCITT no.39 79] CCITT Study Group XIV - Temporary Document No.39-E, Kyoto 7-15, Nov. 1979.

[CCITT 80] CCITT "Standardization of Group 3 Facsimile Apparatus for Document Transmission" Geneva, 1980.

[CCITT 85] CCITT, "Terminal Equipment and Protocols for Telemation Services," Red Book Volume VII-Fascicle VII.3, VIII Plenary Session, 1985.

[Chamzas 89] Chamzas C. & Duttweiler D.L., "Encoding Facsimile Images for packet-switched networks," IEEE Journal on Selected Areas in Commun., Vol. 7, no. 5, pp. 857-863, June 1989.

[Coulon 74] Coulon, F. and Kunt, M., "An Alternative to Runlength Coding for Black-and-White Facsimile," Proc. 1974 International Zurich Seminar on Digital Communications, Zurich Switzerland, pp. C4.1-C4.4, March 1974.

[Coulon 76] Coulon, F. and Johnsen, O., "Adaptive Block Scheme for Source Coding of Black-and-White Facsimile," Electronics Letters, Vol. 12, no. 3, pp. 61-62, Feb. 1976.

[Dimento 90] Dimento L.J. & Berkovich S.Y., "The Compression Effects of the Binary Tree Overlapping Method on Digital Imagery," IEEE Trans. on Commun. Vol. 38, no. 8, pp. 1260-1265, August 1990.

[Gharavi 87] Gharavi H., "Conditional Run-Length and Variable-Length Coding of Digital Pictures," IEEE Trans. on Commun., Vol. COM-35, pp. 671-676, June 1987.

[Green 89] Green, W. B., "Binary Image Processing," in Digital Image Processing, pp. 476-553, New York Van Nostrand Reinhold,1989.

[Hou 83] Hou, H. S., "Digital Document Processing," pp. 116-139, New York, John Wiley and Sons, 1983.

[Huang 72] Huang, T. S., "Runlength Coding and Its Extensions," In Picture Bandwidth Compression, pp. 233-264, 1972.

[Huang 74] Huang, T. S. and Meyr, H., Rosdolsky, H. G., "Optimum Run-length Codes," IEEE trans. on Commun., pp. 826-837, June 1974.

[Huang 75a] Huang, T. S. and Hussain, A. B. S., "Facsimile Coding by

Skipping White," IEEE Trans. on Commun. COM-23, no.12, pp. 1452-1466, Dec. 1975.

[Huang 75b] Huang, T. S., "Easily Implementable Suboptimum Runlength Codes," Proc. ICC, San Francisco, 1975.

[Huang 77] Huang, T. S., "Coding of Two-Tone Images," IEEE Trans. on Commun. COM-12, no.11, pp. 1406-1424, Nov. 1977

[Huffman 52] Huffman, D. A., "A Method for the Construction of Minimum Redundancy Codes," Proc. IRE, Vol.40, pp. 1098-1101, Sept. 1952.

[Hunter 80] Hunter, R. & Robinson, A. H., "International Digital Facsimile Coding Standards," Proc. IEEE, Vol.68, no.7, pp. 854-867, July 1980.

[Jain 81] Jain A.K., "Image Data Compression: A Review," Proc. IEEE, Vol. 69, pp. 349-389, March 1981.

[Johnsen 76] Johnsen, O., "Etude de Strategies Adaptives Pour La Transmission d`images Facsimile' a' deux Niveaux," AGEN-Mittelungen, no. 20, June1976.

[Johnsen 82] Johnsen, O. and Segen, J., "A pattern Matching Technique for Facsimile Coding," in Papers of ICC'82, Philadelphia, PA, pp. 2G.2.1-2G.2.7, June 1982.

[Kobayashi 74] Kobayashi, H. and Bahl, L. R., "Image Data Compression Predictive Coding," IBM Journal Research and Development, pp. 164-179, 1974.

[Laemmel 51] Laemmel, A. E., "Coding Processes for Bandwidth Reduction in Picture Transmission," Report R-246-51, PIB-187. Microwave Research Institute, Polytechnic Institute of Brooklyn, New York, pp. 29, August 1951.

[Langdon 81] Langdon G. G. & Rissanen J., "Compression of Black-White Images with Arithmetic Coding," IEEE Trans. on Commun. COM-29, pp. 858-867, June 1981.

[Musmann 77] Musmann, H. G., "Comparison of Redundancy Reducing Codes for Facsimile Transmission of Documents," IEEE Trans. on Commun. COM-25, no.11, pp. 1425-1433, Nov. 1977.

[Netravali 80] Netravali A.N. & Limb J.O., "Picture Coding: A Review," Proc. IEEE, Vol. 68, pp. 366-406, March 1980.

[Netravali 88] Netravali, Arun N. and Haskell, Barry G. "Examples of Codec Designs," Digital pictures: Representation and Compression, Plenum Publishing Corporation, New York, pp. 505-550, 1988.

[Pratt 76] Pratt, W. K. et al., "Block Character Coding," Proc. SPIE Vol.87, Advances in Image Transmission Techniques, pp. 222-228, 1976.

[Pratt 80] Pratt, W. K. et al., "Combined Symbol Matching Data Compression System," Proc. IEEE, Vol.68, no.7, pp. 786-796, July 1980.

[Rose 89] Rose K.M. & Heiman A., "Enhancement of One-Dimensional Variable-Length DPCM Images Corrupted by Transmission Errors," IEEE Trans. on Commun., Vol. 37, no. 4, pp. 373-379, April 1989.

[Samet 80] Samet, H., "Region Representation : Quadtrees from Boundary Codes," Comm. ACM 23, pp. 163-176, March 1980.

[Silver 81] Silver, D. M., "Advanced Techniques for Data Compression of Digital Facsimile Pictures," M.Eng.Sc. thesis, Department of Electrical Engineering, University of Melbourne, 1981.

[Silver 84] Silver, D. M. and Johnson, D.A.H., "Facsimile Coding using

Symbol-Matching Techniques," Proc. IEE, Vol.131,Part F, no.2, pp. 125-129, April 1984.

[Takagi 75] Takagi, M. and Tsuda, T., "A Highly Efficient Run-Length Coding Scheme for Facsimile Transmission," Electron Commun. Japan 58, pp. 30, 1975.

[Usubuchi 80] Usubuchi, T. et al., "Adaptive Predictive Coding for Newspapers Facsimile," Proc. IEEE Vol. 68, no.7, pp. 807-813, July 1980.

[Wakahara 74] Wakahara, Y. et al., "Data Compression of Facsimile Signal by Relative Address Coding," Paper of Tech. Group IECE Japan, Vol. CS74-115, Nov. 1974.

[Wakahara 76] Wakahara, Y. et al., " Data Compression Factor of Relative Address Coding Scheme for Facsimile Signals," J. IIEE Jap., Vol.5, no.3, Oct. 1976.

[Yamazaki 76] Yamazaki, Y. et al., "Digital Facsimile Equipment 'Quick-FAX' using a New Redundancy Reduction Technique," presented at the NTC'76, pp. 6.2-1, Nov. 1976.

[Yasuda 80] Yasuda, Y., "Overview of Digital Facsimile Coding Technique in Japan," Proc. IEEE Vol.68, no.7, pp. 830-845, July1980.

[Yasuda 85] Yasuda, Y. et al., "Advances in FAX," Proc. IEEE,Vol. 73, no.4, pp. 706-729, April 1985.

VITA 2

D. Wei-Fang Hsu

Candidate for the Degree of

Master of Science

Thesis: AN INTERLEAVING TECHNIQUE FOR BLOCK CODING OF BLACK-AND-WHITE FACSIMILE DATA

Major Field: Computer Science

Biographical:

Personal Data: Born in Hsinchu, Taiwan, R.O.C., January 9th, 1960, daughter of Mr. Chun Doong and Mrs. Yu-Yen-Chiao Doong.

Education: Graduate from Hsinchu Girl High School, Taiwan, R.O.C., 1978; received a Bachelor of Business Administration with a major in Cooperation Economics from National Chung Hsing University, Taipei, Taiwan, R.O.C., in June, 1981; completed requirements for the Master of Science degree at Oklahoma State University in December, 1991.

Professional Experience:
System Support Analyst, Qume Corporation, Taiwan, Dept. of Quality Assurance (July 1984 - July 1986).