CoE4TN4
Image Processing

Chapter 8
Image Compression

McMaster
University

Image Compression

- Digital images: take huge amount of data
- Storage, processing and communications requirements might be impractical
- More efficient representation of digital images is necessary
- Image compression: reduces the amount of data required to represent a digital image by removing redundant data
- Theoretical work (1940 Shannon): information, its representation, transmission and compression
- Image compression is an enabling technology: multimedia, video conferencing, remote sensing, Fax, ….

McMaster University

Fundamentals

- Data compression: process of reducing the amount of data required to represent a given quantity of information
- Data and information are not the same.
- Data is used to convey information.
- Various amount of data may be used to represent the same information

Fundamentals

- Suppose that two representations have the same amount of information one using n1 units of data and the other one n2 units of data.
- Compression ratio: $C_R = n1/n2$
- Relative data redundancy of the first set is defined as:
- $R_D = 1 - 1/C_R$
- $n2 << n1$, C_R approaches infinity and R_D approaches one. This means that the data n1 has been highly redundant.
- $n2 >> n1$, C_R approaches zero and R_D approaches minus infinity. This means that the data n2 contains more data (expansion instead of compression).

McMaster University

Fundamentals

- A typical compression ratio: 10
- C_R=10, R_D=1-1/ 10=0.9
- 90% of the data in the first set is redundant

- Three forms of data redundancies exist in digital images
  1. Coding redundancy
  2. Spatial and temporal redundancy
  3. Irrelevant information

McMaster University

Coding redundancy

- The gray-level histogram of an image can provide a great deal of insight into the construction of codes to reduce the amount of data used to represent the image

$r_k \in [0,1]$ **Normalized gray levels**

$$p_r(r_k) = \frac{n_k}{n}$$ **Probability of k th gray-level**

r_k	$p_r(r_k)$	Code 1	$l_1(r_k)$	Code 2	$l_2(r_k)$
$r_{87} = 87$	0.25	01010111	8	01	2
$r_{128} = 128$	0.47	10000000	8	1	1
$r_{186} = 186$	0.25	11000100	8	000	3
$r_{255} = 255$	0.03	11111111	8	001	3
r_k for $k \neq 87, 128, 186, 255$	0	—	8	—	0

McMaster University

Coding redundancy

- Suppose that gray-level r_k is represented by $l(r_k)$ bits
- Average length of code words used for the image:

$$L_{avg} = \sum_{k=0}^{L-1} p_r(r_k) l(r_k)$$

Total bits=MNL$_{avg}$

Average length of a code is sometimes called the rate of that code

McMaster University

Coding redundancy

- If the same number of bits is used for all gray levels (natural m-bit binary code)

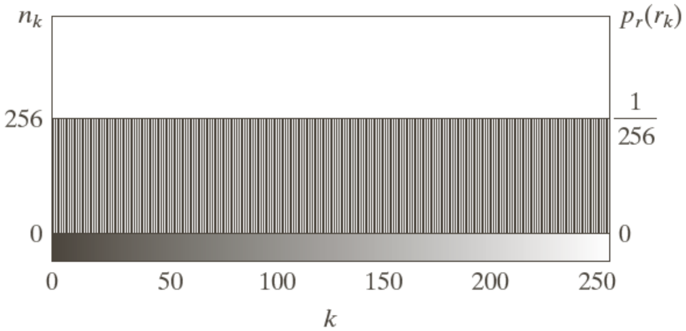
$$L_{avg} = m\sum_{k=0}^{L-1} p_r(r_k) = m$$

- If fewer bits are assigned to more probable gray-levels than to the less probable ones, compression can be achieved.
- This is called Variable Length Coding (VLC).

r_k	$p_r(r_k)$	**Code 1**	$l_1(r_k)$	**Code 2**	$l_2(r_k)$
$r_{87} = 87$	0.25	01010111	8	01	2
$r_{128} = 128$	0.47	10000000	8	1	1
$r_{186} = 186$	0.25	11000100	8	000	3
$r_{255} = 255$	0.03	11111111	8	001	3
r_k for $k \neq 87, 128, 186, 255$	0	—	8	—	0

McMaster University

Spatial and temporal redundancy



Spatial and temporal redundancy

- Codes designed based on histogram do not exploit the correlation between pixels.
- These correlations result from the structural or geometrical relationships between the objects in the image.
- The value of any pixel is highly correlated to its neighbors and the information carried by individual pixels is small.

Spatial and temporal redundancy

- How to reduce spatial and temporal redundancy?
- The image (which is a 2-D array) is transformed into a more efficient format.
- This process is called mapping or transformations.
- The transformation should reduce the redundancy.
- Example: the transform could be the difference between adjacent pixels
- The transform (or mapping) should be reversible. The original image should be recoverable from the transform.

Irrelevant Information

- The eye does not response with equal sensitivity to all visual information
- Certain information has less relative importance than other
- This information is said to be psychovisually redundant. It can be eliminated without significantly impairing the quality of image perception.
- Since the elimination of irrelevant information results in a loss of quantitative information, it is commonly referred to as quantization.

McMaster University

Measuring Image Information

- A random event E that occurs with probability P(E) is said to contain :

$$I(E) = \log \frac{1}{P(E)} = -\log P(E)$$

units of information

I(E) self-information

If P(E)=1 then I(E)=0,
no uncertainty is associated with the event, no information would be transformed by communicating that E has occurred.

McMaster University

Elements of Information Theory

- Information source generates a random sequence of symbols.
- The set of source symbols (source alphabet) A:

$$\{a_1, a_2, \ldots, a_J\}$$

Probability of the symbols of the source

$$z = \begin{bmatrix} P(a_1) \\ P(a_2) \\ .. \\ P(a_J) \end{bmatrix}$$

McMaster University

Elements of Information Theory

- Average information per source output $H(z) = -\sum_{j=1}^{J} P(a_j) \log P(a_j)$
- H: uncertainty or entropy of the source
- H is the average amount of information obtained by observing a single source output
- If the source symbols are equally probable, the entropy is maximized

McMaster University

Noiseless coding theorem

- What is the minimum average code word length that can be achieved?
- Noiseless coding theorem or Shannon's first theorem.
- The minimum average code word length that can be achieved is equal to the entropy of the source.
- One of the ways that we can get closer and closer to the entropy is to group the source outputs together and build n-tuples (nth-extension)

Noiseless coding theorem

Entropy: H=0.918 bits/symbol
Average code word length per symbol (first extension) = 1
Average code word length per symbol (second extension) = 1.89/2=0.945

α_i	Source Symbols	$P(\alpha_i)$ Eq. (8.3-14)	$I(\alpha_i)$ Eq. (8.3-1)	$l(\alpha_i)$ Eq. (8.3-16)	Code Word	Code Length
First Extension						
α_1	a_1	2/3	0.59	1	0	1
α_2	a_2	1/3	1.58	2	1	1
Second Extension						
α_1	a_1a_1	4/9	1.17	2	0	1
α_2	a_1a_2	2/9	2.17	3	10	2
α_3	a_2a_1	2/9	2.17	3	110	3
α_4	a_2a_2	1/9	3.17	4	111	3

Fidelity Criteria

- Since information may be lost during compression a means of quantifying the information loss is desirable.
- 2 type of criteria:
  1. Objective fidelity criteria
  2. Subjective fidelity criteria
- Objective fidelity criteria: the information loss is expressed as a function of input image (original image) and output image (compressed and decompressed)

Fidelity Criteria

- Root mean square (rms) error:

$$e_{rms} = \left[\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\hat{f}(x,y) - f(x,y) \right]^2 \right]^{1/2}$$

Mean square signal to noise ratio:

$$SNR_{ms} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left(\hat{f}(x,y) \right)^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\hat{f}(x,y) - f(x,y) \right]^2}$$

McMaster University

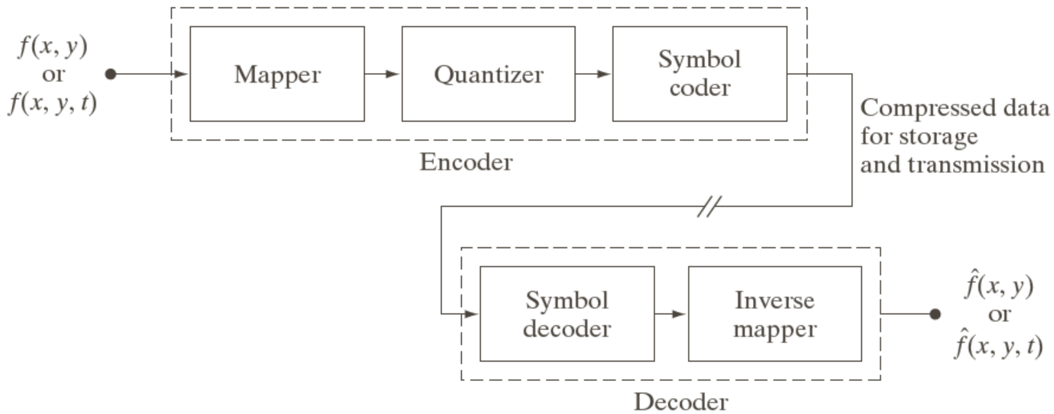
Fidelity Criteria

- Objective fidelity criteria: simple and convenient, sometimes misleading
- Subjective fidelity criteria: measuring image quality by the subjective evaluation of a group of viewers and averaging their evaluations.

Value	Rating	Description
1	Excellent	An image of extremely high quality, as good as you could desire.
2	Fine	An image of high quality, providing enjoyable viewing. Interference is not objectionable.
3	Passable	An image of acceptable quality. Interference is not objectionable.
4	Marginal	An image of poor quality; you wish you could improve it. Interference is somewhat objectionable.
5	Inferior	A very poor image, but you could watch it. Objectionable interference is definitely present.
6	Unusable	An image so bad that you could not watch it.

Image Compression Models

- A compression system consists of two distinct structure of blocks: encoder and decoder.
- Encoder: creates a set of symbols from the input image
- Decoder: reconstructs an output image $\hat{f}(x, y)$



Encoder

- Encoder: responsible for reducing or eliminating any coding, interpixel or pschovisual redundancies in the input image.
- Encoding can be modeled by three independent operations.
- Each operation is designed to reduce one of the three redundancies.

McMaster University

Encoder

- Mapper (transform): transforms the input data into a format designed to spatial and temporal redundancies.
  - This operation generally is reversible.
  - It may or may not reduce the amount of data required to represent the image.
- Quantizer: reduces the accuracy of the mapper’s output.
  - This stage removes the irrelevant information.
  - This operation is irreversible. It should be omitted when error-free (loss less) compression is desired.

$\rightarrow$ **Mapper** $\rightarrow$ **Quantizer** $\rightarrow$ **Symbol Encoder**

Encoder

- Symbol coder: creates a fixed or variable length code to represent the quantizer's output.
  - In most cases a VLC is used (shortest code words to the most frequently occurring values of the quantizer's output)
  - The operation is reversible.

Decoder

- Contains two components:
  - Symbol decoder
  - Inverse mapper
- They perform the inverse operations of symbol encoder and mapper.

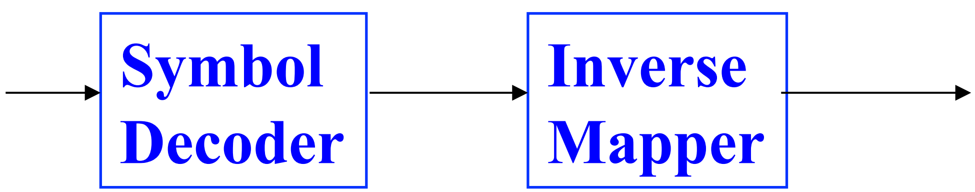


Image Formats, Compression Standards

Image Compression Standards, Formats, and Containers

- **Still Image**
  - **Binary**
    - CCITT Group 3
    - CCITT Group 4
    - JBIG (or JBIG1)
    - JBIG2
    - TIFF
  - **Continuous Tone**
    - JPEG
    - JPEG-LS
    - JPEG-2000
    - BMP
    - GIF
    - PDF
    - PNG
    - TIFF
- **Video**
  - DV
  - H.261
  - H.262
  - H.263
  - H.264
  - MPEG-1
  - MPEG-2
  - MPEG-4
  - MPEG-4 AVC
  - AVS
  - HDV
  - M-JPEG
  - QuickTime
  - VC-1 (or WMV9)

Name	Organization	Description
Bi-Level Still Images		
CCITT Group 3	ITU-T	Designed as a facsimile (FAX) method for transmitting binary documents over telephone lines. Supports 1-D and 2-D run-length [8.2.5] and Huffman [8.2.1] coding.
CCITT Group 4	ITU-T	A simplified and streamlined version of the CCITT Group 3 standard supporting 2-D run-length coding only.
JBIG *or* JBIG1	ISO/IEC/ ITU-T	A *Joint Bi-level Image Experts Group* standard for progressive, lossless compression of bi-level images. Continuous-tone images of up to 6 bits/pixel can be coded on a bit-plane basis [8.2.7]. Context sensitive arithmetic coding [8.2.3] is used and an initial low resolution version of the image can be gradually enhanced with additional compressed data.
JBIG2	ISO/IEC/ ITU-T	A follow-on to JBIG1 for bi-level images in desktop, Internet, and FAX applications. The compression method used is content based, with dictionary based methods [8.2.6] for text and halftone regions, and Huffman [8.2.1] or arithmetic coding [8.2.3] for other image content. It can be lossy or lossless.
Continuous-Tone Still Images		
JPEG	ISO/IEC/ ITU-T	A *Joint Photographic Experts Group* standard for images of photographic quality. Its lossy *baseline coding system* (most commonly implemented) uses quantized discrete cosine transforms (DCT) on 8 × 8 image blocks [8.2.8], Huffman [8.2.1], and run-length [8.2.5] coding. It is one of the most popular methods for compressing images on the Internet.
JPEG-LS	ISO/IEC/ ITU-T	A lossless to near-lossless standard for continuous tone images based on adaptive prediction [8.2.9], context modeling [8.2.3], and Golomb coding [8.2.2].
JPEG-2000	ISO/IEC/ ITU-T	A follow-on to JPEG for increased compression of photographic quality images. Arithmetic coding [8.2.3] and quantized discrete wavelet transforms (DWT) [8.2.10] are used. The compression can be lossy or lossless.

(*Continues*)

Name	Organization	Description
Video		
DV	IEC	*Digital Video*. A video standard tailored to home and semiprofessional video production applications and equipment—like electronic news gathering and camcorders. Frames are compressed independently for uncomplicated editing using a DCT-based approach [8.2.8] similar to JPEG.
H.261	ITU-T	A two-way videoconferencing standard for ISDN (*integrated services digital network*) lines. It supports non-interlaced 352 × 288 and 176 × 144 resolution images, called CIF (*Common Intermediate Format*) and QCIF (*Quarter CIF*), respectively. A DCT-based compression approach [8.2.8] similar to JPEG is used, with frame-to-frame prediction differencing [8.2.9] to reduce temporal redundancy. A block-based technique is used to compensate for motion between frames.
H.262	ITU-T	See MPEG-2 below.
H.263	ITU-T	An enhanced version of H.261 designed for ordinary telephone modems (i.e., 28.8 Kb/s) with additional resolutions: SQCIF (*Sub-Quarter* CIF 128 × 96), 4CIF (704 × 576), and 16CIF (1408 × 512).
H.264	ITU-T	An extension of H.261–H.263 for videoconferencing, Internet streaming, and television broadcasting. It supports prediction differences within frames [8.2.9], variable block size integer transforms (rather than the DCT), and context adaptive arithmetic coding [8.2.3].
MPEG-1	ISO/IEC	A *Motion Pictures Expert Group* standard for CD-ROM applications with non-interlaced video at up to 1.5 Mb/s. It is similar to H.261 but frame predictions can be based on the previous frame, next frame, or an interpolation of both. It is supported by almost all computers and DVD players.
MPEG-2	ISO/IEC	An extension of MPEG-1 designed for DVDs with transfer rates to 15 Mb/s. Supports interlaced video and HDTV. It is the most successful video standard to date.
MPEG-4	ISO/IEC	An extension of MPEG-2 that supports variable block sizes and prediction differencing [8.2.9] within frames.
MPEG-4 AVC	ISO/IEC	MPEG-4 Part 10 *Advanced Video Coding* (AVC). Identical to H.264 above.

McMaster University

Name	Organization	Description
Continuous-Tone Still Images		
BMP	Microsoft	*Windows Bitmap*. A file format used mainly for simple uncompressed images.
GIF	CompuServe	*Graphic Interchange Format*. A file format that uses lossless LZW coding [8.2.4] for 1- through 8-bit images. It is frequently used to make small animations and short low resolution films for the World Wide Web.
PDF	Adobe Systems	*Portable Document Format*. A format for representing 2-D documents in a device and resolution independent way. It can function as a container for JPEG, JPEG 2000, CCITT, and other compressed images. Some PDF versions have become ISO standards.
PNG	*World Wide Web Consortium* (W3C)	*Portable Network Graphics*. A file format that losslessly compresses full color images with transparency (up to 48 bits/pixel) by coding the difference between each pixel's value and a predicted value based on past pixels [8.2.9].
TIFF	Aldus	*Tagged Image File Format*. A flexible file format supporting a variety of image compression standards, including JPEG, JPEG-LS, JPEG-2000, JBIG2, and others.
Video		
AVS	MII	*Audio-Video Standard*. Similar to H.264 but uses exponential Golomb coding [8.2.2]. Developed in China.
HDV	Company consortium	*High Definition Video*. An extension of DV for HD television that uses MPEG-2 like compression, including temporal redundancy removal by prediction differencing [8.2.9].
M-JPEG	Various companies	*Motion JPEG*. A compression format in which each frame is compressed independently using JPEG.
Quick-Time	Apple Computer	A media container supporting DV, H.261, H.262, H.264, MPEG-1, MPEG-2, MPEG-4, and other video compression formats.
VC-1 WMV9	SMPTE Microsoft	The most used video format on the Internet. Adopted for HD and *Blu-ray* high-definition DVDs. It is similar to H.264/AVC, using an integer DCT with varying block sizes [8.2.8 and 8.2.9] and context dependent variable-length code tables [8.2.1]—but no predictions within frames.

McMaster University

Huffman coding

- Huffman coding is the most popular technique for VLC coding and removing the coding redundancies.
- The first step in Huffman coding is to create a series of source reductions by ordering the probabilities of the symbols and combining the lowest probabilities into a single symbol that replaces them in the next source reduction.
- The second step is to code each reduced source.

Huffman coding

A2	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
A6	0.3	00	0.3	00	0.3	00	0.3	00	0.4	1
A1	0.1	011	0.1	011	0.2	010	0.3	01		
A4	0.1	0100	0.1	0100	0.1	011				
A3	0.06	01010	0.1	0101						
A5	0.04	01011								

Lavg=0.4+0.3x2+0.1x3+0.1x4+0.06x5+0.04x5=2.2 bits/symbol
H=2.14 bits/symbol

Huffman coding

- Huffman code is an instantaneous, uniquely decodable, block code:
- Instantaneous: each code word in a string of code symbols can be decoded without referencing following symbols
- Block code: each source symbol is mapped into a fixed code word.
- Unique: each string of code symbols can be decoded in only one way.

Golomb Coding

- Coding of nonnegative integers with exponentially decaying probability
- n nonnegative integer to be coded, m parameter of the Golomb code

1. Form the unary code of quotient $\lfloor n/m \rfloor$ (unary code of an integer q is q 1s followed by a 0)
2. Let $k = \lceil log_2 m \rceil \quad c = 2^k - m \qquad r = n \quad mod \quad m$ and compute r'

$$r' = \begin{cases} r & \text{truncated to k-1 bits} & 0 \leq r < c \\ r + c & \text{truncated to k bits} & \text{otherwise} \end{cases}$$

3. Concatenate the results of steps 1 and 2

McMaster University

Golomb Coding

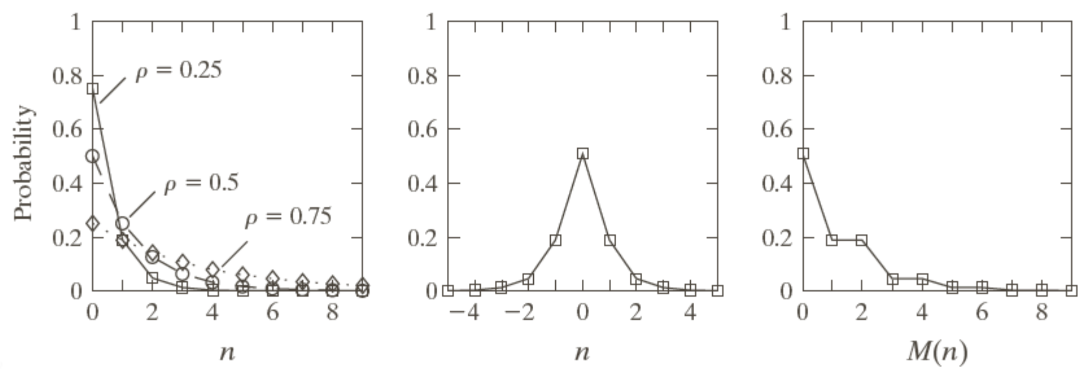
n	$G_1(n)$	$G_2(n)$	$G_4(n)$	$G_{\exp}^0(n)$
0	0	00	000	0
1	10	01	001	100
2	110	100	010	101
3	1110	101	011	11000
4	11110	1100	1000	11001
5	111110	1101	1001	11010
6	1111110	11100	1010	11011
7	11111110	11101	1011	1110000
8	111111110	111100	11000	1110001
9	1111111110	111101	11001	1110010

McMaster University

Golomb Coding

- Golomb codes are seldom used for coding of image intensities
- Intensity differences can be coded using Golomb codes after converting negative values to positive

$$M(n) = \begin{cases} 2n & n \geq 0 \\ 2|n| - 1 & n < 0 \end{cases}$$



a b c

FIGURE 8.10 (a) Three one-sided geometric distributions from Eq. (8.2-2); (b) a two-sided exponentially decaying distribution; and (c) a reordered version of (b) using Eq. (8.2-4).

McMaster University

Arithmetic Coding

- There is no one to one correspondence between source symbols and code words.
- An entire sequence of source symbols is assigned a single code word.
- Each source symbol is represented by an interval in [0,1). As the number of symbols increases the size of the interval reduces in accordance with the probability of occurrence of symbols.

Arithmetic Coding

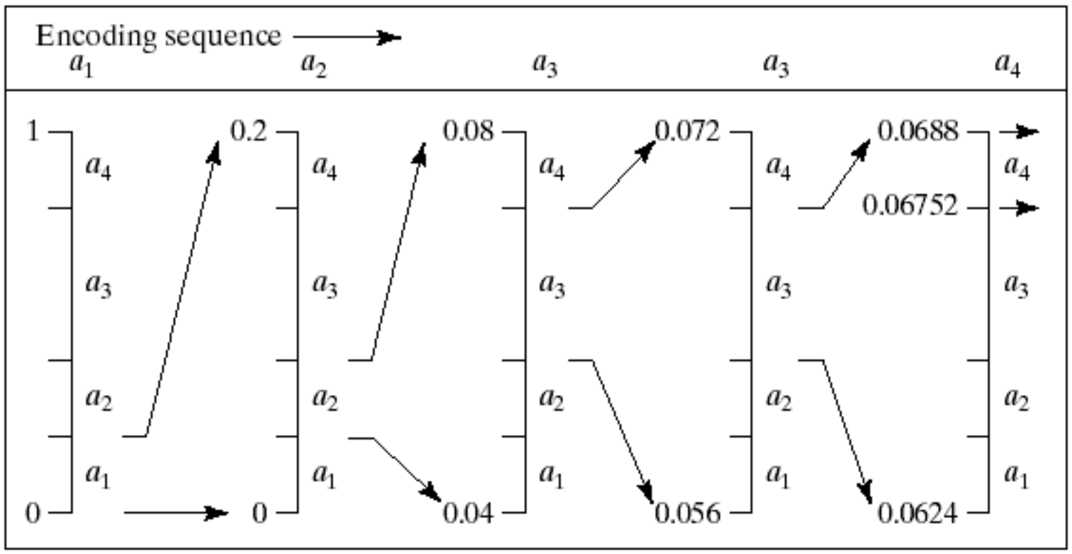
a1 0.2 [0,0.20

a2 0.2 [0.2,0.4)

a3 0.4 [0.4,0.8)

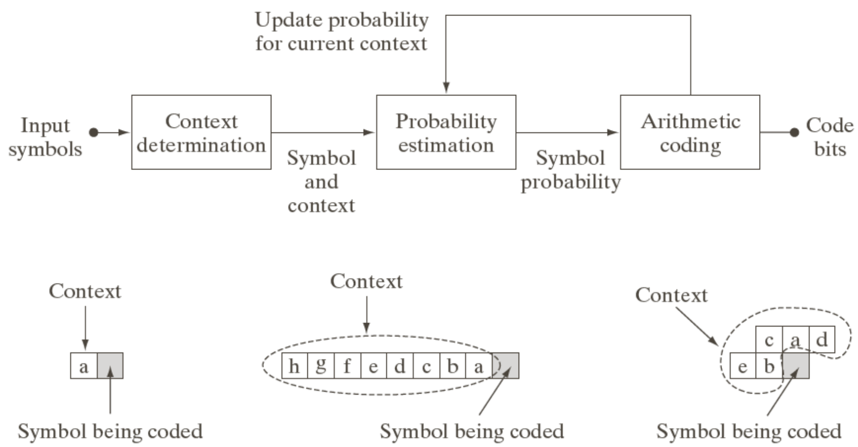
a4 0.2 [0.8,1.0)

Sequence to be coded a1 a2 a3 a3 a4



Arithmetic Coding

- Inaccurate probability models can lead to non-optimal results
- Solution: use an adaptive, context dependent probability model
- Adaptive: symbols probabilities are updated as the symbols are coded.
- Context dependent: probabilities are based on a predefined neighborhood of pixels around the symbol being coded.



LZW

- At the beginning of the coding process, a codebook or dictionary containing the source symbols to be coded is constructed.
- The encoder examines each symbol in the message and concatenates it with the symbols following it to find the longest sequence which is in the dictionary.
- If after concatenating, the sequence is not in the dictionary, the encoder adds the sequence as the new element to the dictionary.

LZW: Example

Encode “wabba_wabba_wabba_wabba_woo_woo_woo”

Output: 5 2 3 3 2 1 6 8 10 12 9 11 7 16 5 4 4 11 21 23 4

Dictionary:

1. _	6. wa	11. _w	16. ba_	21. oo
2. a	7. ab	12. wab	17. _wa	22. o_
3. b	8. bb	13. bba	18. abb	23. _wo
4. o	9. ba	14. a_w	19. ba_w	24. oo_
5. w	10. a_	15. wabb	20. wo	25. _woo

LZW

- LZW compression has been integrated in many imaging file formats including GIF, TIFF and PDF.
- Most practical applications require a strategy for handling dictionary overflow
- A simple solution is to flush or reinitialize the dictionary when it becomes full
- A more complex solution is to monitor compression performance and flush the dictionary when compression becomes poor or unacceptable.

Run Length Coding

- Run-length coding: coding the length of runs instead of coding individual values
  - Exp: 190 white pixels, followed by 30 black, followed by 210 white
  - Instead of coding 430 pixels individually, we code the sequence of 190,30, 210 along with an indication of the color of the first string

G3

- Group 3: includes two coding schemes:
  1. 1-D: coding of each line is performed independently of any other line
  2. 2-D: coding of one line is performed using line to line correlation.
- 1-D coding: is a run-length coding scheme in which each line is represented as alternative white and black runs from left to right.
- The first run is always a white run. So, if the first pixel is black, the white run has a length of zero.
- Runs of different lengths occur with different probabilities, therefore they are coded using VLC (variable length codes).
- CCITT uses Huffman coding

G3

- The number of possible length of runs is huge and it is not feasible to build a code book that large.
- The run length rl is expressed as:
  - $r_l = 64*m + t$ $t = 0, 1, .., 63$ $m = 1, 2, \ldots, 27$
- To represent r_l, we use the corresponding codes for m and t.
- The code for t are called terminating codes and for m make-up codes.
- If $r_l < 63$ only a terminating code is used
- Otherwise both a make-up code and a terminating code are used
- A unique EOL (end of line) codeword 000000000001 is used to terminate each line.

McMaster University

TABLE 8.14
CCITT terminating codes.

Run Length	White Code Word	Black Code Word	Run Length	White Code Word	Black Code Word
0	00110101	0000110111	32	00011011	000001101010
1	000111	010	33	00010010	000001101011
2	0111	11	34	00010011	000011010010
3	1000	10	35	00010100	000011010011
4	1011	011	36	00010101	000011010100
5	1100	0011	37	00010110	000011010101
6	1110	0010	38	00010111	000011010110
7	1111	00011	39	00101000	000011010111
8	10011	000101	40	00101001	000001101100
9	10100	000100	41	00101010	000001101101
10	00111	0000100	42	00101011	000011011010
11	01000	0000101	43	00101100	000011011011
12	001000	0000111	44	00101101	000001010100
13	000011	00000100	45	00000100	000001010101
14	110100	00000111	46	00000101	000001010110
15	110101	000011000	47	00001010	000001010111
16	101010	0000010111	48	00001011	000001100100
17	101011	0000011000	49	01010010	000001100101
18	0100111	0000001000	50	01010011	000001010010
19	0001100	00001100111	51	01010100	000001010011
20	0001000	00001101000	52	01010101	000000100100
21	0010111	00001101100	53	00100100	000000110111
22	0000011	00000110111	54	00100101	000000111000
23	0000100	00000101000	55	01011000	000000100111
24	0101000	00000010111	56	01011001	000000101000
25	0101011	00000011000	57	01011010	000001011000
26	0010011	000011001010	58	01011011	000001011001
27	0100100	000011001011	59	01001010	000000101011
28	0011000	000011001100	60	01001011	000000101100
29	00000010	000011001101	61	00110010	000001011010
30	00000011	000001101000	62	00110011	000001100110
31	00011010	000001101001	63	00110100	000001100111

McMaster University

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

Run Length	Code Word	Run Length	Code Word
1792	00000001000	2240	000000010110
1856	00000001100	2304	000000010111
1920	00000001101	2368	000000011100
1984	000000010010	2432	000000011101
2048	000000010011	2496	000000011110
2112	000000010100	2560	000000011111
2176	000000010101		

TABLE 8.15
CCITT makeup codes.

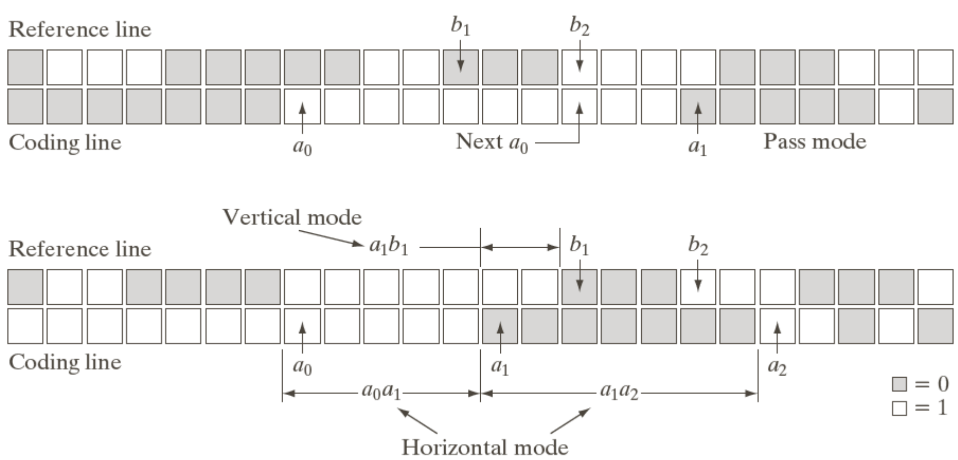
G3

- 2-D coding (modified READ=MR): rows of a facsimile image are heavily correlated. Therefore it would be easier to code the transition points with reference to the previous line.
- a0: the last pixel known to both encoder and decoder. At the beginning of encoding each line a0 refers to an imaginary white pixel to the left of actual pixel. While it is often a transition pixel it does not have to be.
- a1: the first transition point to the right of a0. It has an opposite color of a0.
- a2: the second transition pixel to the right of a0. Its color is opposite of a1.

McMaster University

G3

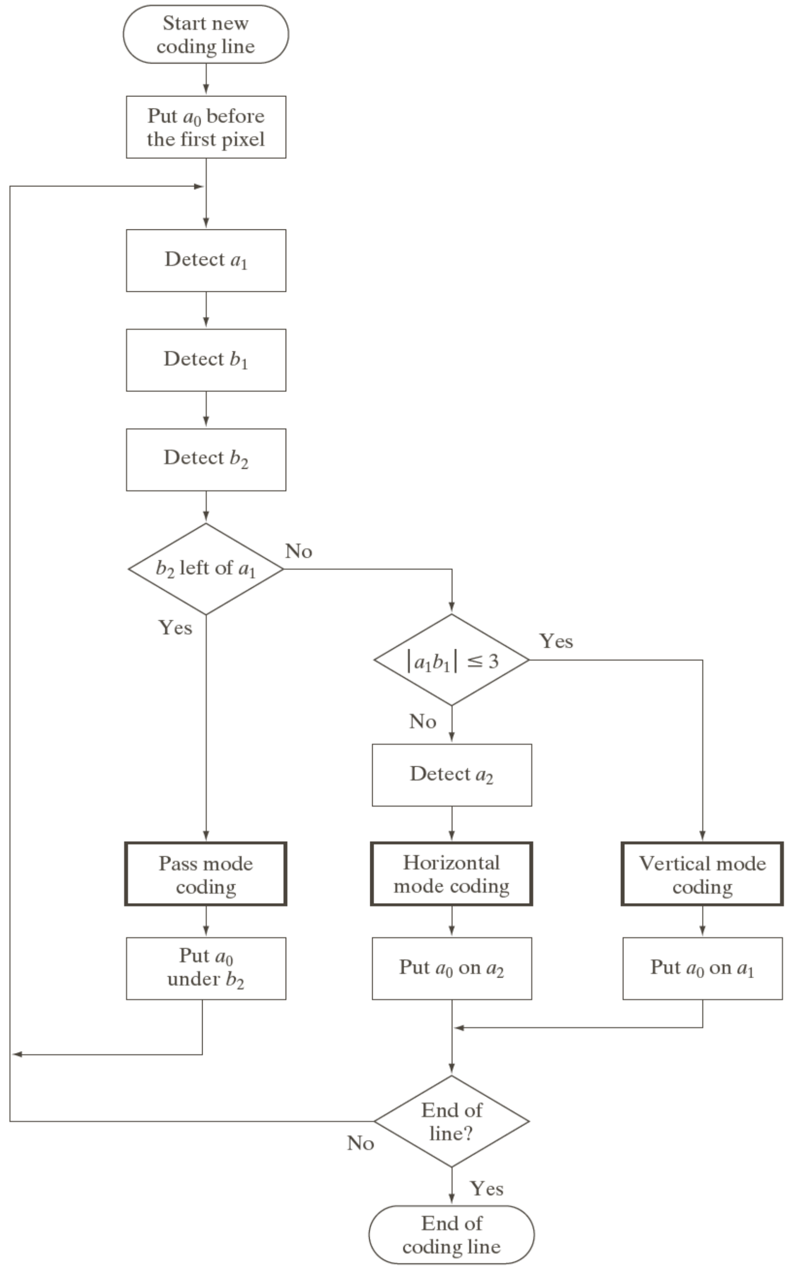
- b1: the first transition pixel on the line above currently being coded, to the right of a0 whose color is opposite of a0.
- b2: the first transition pixel to the right of b1 in the line above the current line



McMaster University

G3

- If b1 and b2 lie between a0 and a1 (pass mode): no transition until b2. Then a0 moves to b2 and the coding continues. Transmitter transmits code 0001.
- If a1 is detected before b2
  - If distance between a1 and b1 (number of pixels) is less than or equal to 3, we send the location of a1 with respect to b1, move a0 to a1 and the coding continues (vertical mode)
  - If the distance between a1 and b1 is larger than 3, we go back to 1-D run length coding and encode the distance between a0 and a1 and a1 and a2 (horizontal mode) using run-length coding. a0 is moved to a2 and the coding continues.



Start new coding line
Put a_0 before the first pixel
Detect a_1
Detect b_1
Detect b_2
b_2 left of a_1
No
Yes
$|a_1b_1| \leq 3$
Yes
No
Detect a_2
Pass mode coding
Horizontal mode coding
Vertical mode coding
Put a_0 under b_2
Put a_0 on a_2
Put a_0 on a_1
End of line?
No
Yes
End of coding line

Mode	Code Word
Pass	0001
Horizontal	$001 + M(a_0a_1) + M(a_1a_2)$
Vertical	
a_1 below b_1	1
a_1 one to the right of b_1	011
a_1 two to the right of b_1	000011
a_1 three to the right of b_1	0000011
a_1 one to the left of b_1	010
a_1 two to the left of b_1	000010
a_1 three to the left of b_1	0000010
Extension	0000001xxx

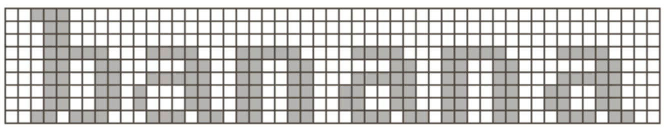
Symbol Coding

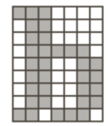
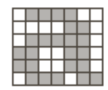
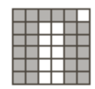
- Symbol or token coding: image is represented as a collection of frequently occurring sub-images called symbol
- Symbols are stored in the symbol dictionary
- Image is coded as a set of triplets

$$\{(x_1, y_1, t_1), (x_2, y_2, t_2), \cdots\}$$

- Storing repeated symbols once can compress images

Symbol Coding



Token	Symbol
0	
1	
2	

Triplet
(0, 2, 0)
(3, 10, 1)
(3, 18, 2)
(3, 26, 1)
(3, 34, 2)
(3, 42, 1)

Bit-plane coding

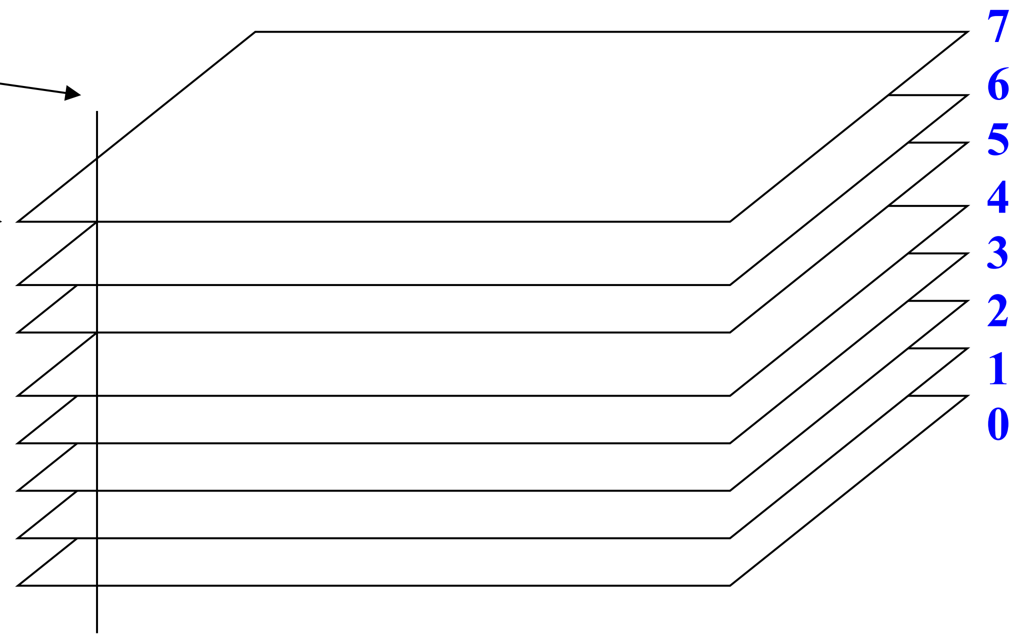
- This method is based on decomposing a multilevel (monochrome or color) image into a series of binary images and compressing each binary image.
- We review different approaches for decomposition and compression of binary images.

McMaster University

Bit plane decomposition

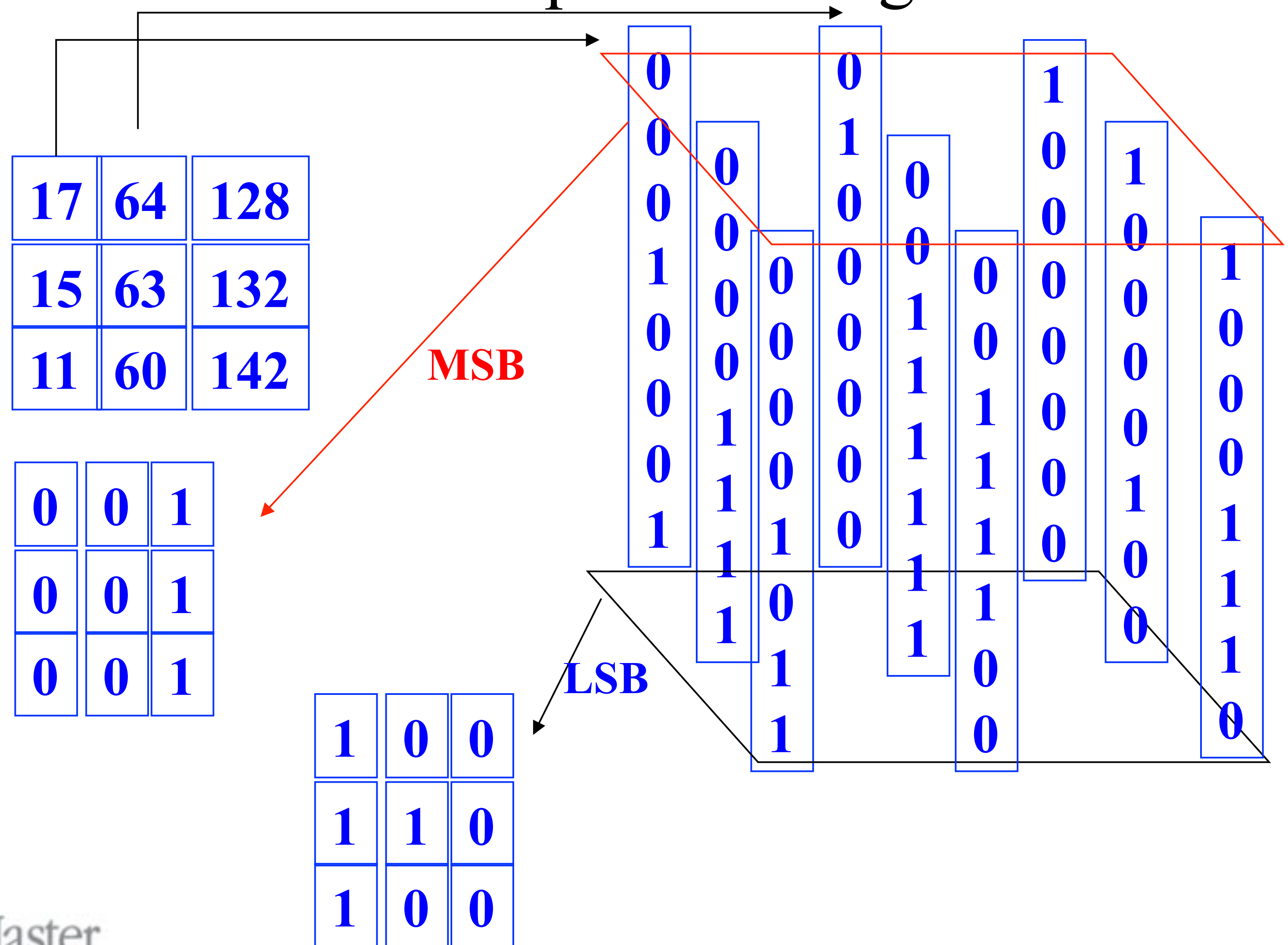
One 8-bit pixel value

Bit plane 7
(most significant)



Bit plane 0
(least significant)

Bit plane slicing



Bit-plane decomposition

- Disadvantage: a small change in gray level can have a significant impact in bit planes.
- Example: two adjacent pixels with values of 127 and 128.
  - 127: 01,111,111
  - 128: 10,000,000
  - There is a transition (0 to 1 or 1 to 0) in every bit plane at the location of these pixels.
  - This transition reduces the coding efficiency.

McMaster University

Bit-plane decomposition

- Alternative method: first represent the image by an m-bit Gray Code.
- Property of Gray code: successive code words differ only in one bit position.
- Small changes in the gray level are less likely to affect all bit planes.
- Example: two adjacent pixels with values of 127 and 128.
  - 127: 01,000,000
  - 128: 11,000,000
- To convert $a_{m-1}\, a_{m-2} \ldots\ldots a_0$ to Gray code:
- $g_{m-1} = a_{m-1}$ $\qquad$ $g_i = a_i \oplus a_{i+1}$

McMaster University

Compressing the bit planes

- A simple but effective method of compressing a binary image (or bit plane) is to use special code words to identify large areas of contiguous 1's or 0's.
- Constant Area Coding:
  - The image is divided into mxn blocks.
  - The blocks are classified as all white, all black or mixed.
  - The most probable category is assigned the one bit code word 0.
  - The other two categories are assigned the code words 10 and 11.
  - The mixed intensity category code is followed by the mn bit patterns of the block.

Compressing the bit planes

- Binary Tree based method: the binary image is decomposed successively into smaller and smaller subblocks.
- A white block is coded as a 0 and all other blocks are assigned a prefix of 1 and divided into smaller subblocks.
- If a subblock is white it is represented by 0 so the code will be 10.
- If the subblock is not white, the decomposing process is repeated until a predefined subblock size is reached and coded as either a 0 (if it is all white) or a 1 followed by the block bit pattern.

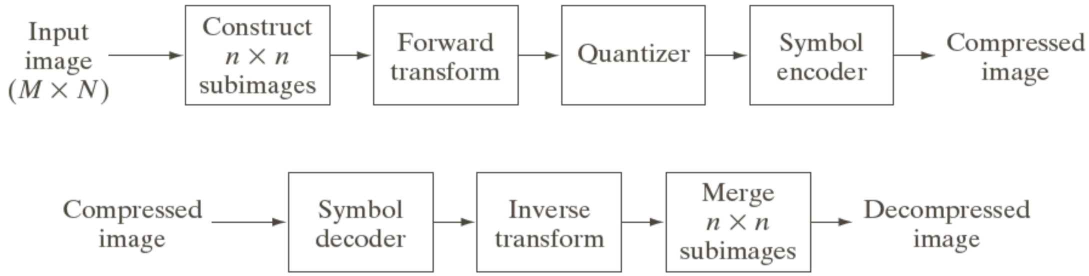
Compressing the bit planes

- One dimensional run-length coding:
- Each row of an image or bit plane is represented by a sequence of lengths that describe successive runs of black and white pixels.
- Example: 000011111 1000 -> 4 6 3
- The problem is to determine the value of the first run.
- The most common approaches for determining the value of first run:
    1. To specify the value of the first run in each row
    2. To assume that each row begins with a white run whose length may be zero

McMaster University

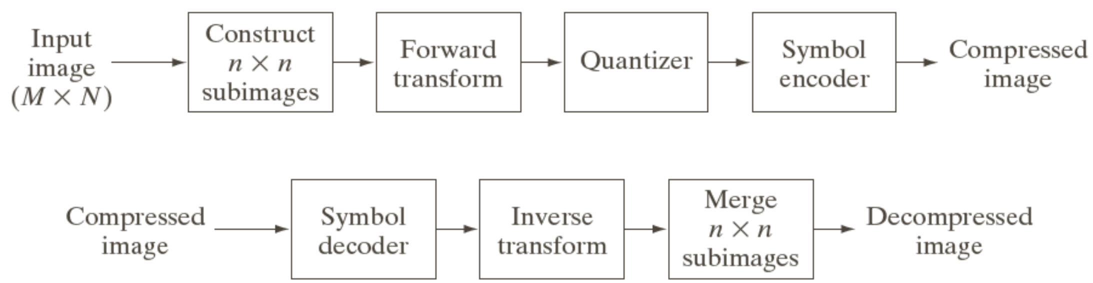
Transform Coding

- The modification is performed on the transform of an image
- A reversible, linear transform is used to map the image into a set of transform coefficients.
- The transform coefficients are then quantized and coded.
- A significant number of the coefficients have small magnitude and can be coarsely quantized (or discarded entirely) with little image distortion.



Transform Coding

- The goal of transformation process is to decorelate the pixels or to pack as much information as possible into the smallest number of coefficients.
- Note that compression is achieved during the quantization of the transformed coefficients not during transform.



McMaster University

Transform Coding

- The choice of particular transform depends on the amount of reconstruction error that can be tolerated and computational resources available.

$$T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{M-1} f(x,y)g(x,y,u,v)$$

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{M-1} T(u,v)h(x,y,u,v)$$

DFT

$$g(x,y,u,v) = \frac{1}{M^2}e^{-j2\pi(ux+vy)/M}$$

$$h(x,y,u,v) = e^{j2\pi(ux+vy)/M}$$

McMaster University

Discrete Cosine Transform) DCT

$$g(x, y, u, v) = h(x, y, u, v) =$$

$$\alpha(u)\alpha(v)\cos\left[\frac{(2x+1)u\pi}{2M}\right]\cos\left[\frac{(2y+1)v\pi}{2M}\right]$$

$$\alpha(u) = \begin{cases} \sqrt{\dfrac{1}{M}} f\ or u = 0 \\ \sqrt{\dfrac{2}{M}} f\ or u = 1,2,\ldots, M-1 \end{cases}$$

McMaster University

Walsh-Hadamard Transform (WHT)

$$g(x,y,u,v) = h(x,y,u,v) = \frac{1}{M}(-1)^{\sum_{i=0}^{m-1}[b_i(x)p_i(u)+b_i(y)p_i(v)]}$$

$$M = 2^m$$

$b_k(z)$ is the kth bit (from right to left) in the binary representation of z

$p_0(u)=b_{m-1}(u)$
$p_1(u)=b_{m-1}(u)+ b_{m-2}(u)$

$P_{m-1}(u)=b_1(u)+ b_0(u)$

McMaster University

| a | b | c |
| d | e | f |

FIGURE 8.24 Approximations of Fig. 8.9(a) using the (a) Fourier, (b) Walsh-Hadamard, and (c) cosine transforms, together with the corresponding scaled error images in (d)–(f).

McMaster University

Transform Coding

- Transformations that pack the most information into the fewest coefficients provide the best approximation, and consequently the smallest reconstruction error.
- Karhunen-Loeve Transform (KLT) is the optimal transformation in an information packing sense.
- It minimizes the mean-square-error for any input image and any number of retained coefficients.
- KLT is data dependent: obtaining the KLT basis images is computationally complex.

Transform Coding

- KLT is seldom used in practice, instead DCT whose bases images are fixed, normally is selected.
- DCT has been used in international standards for image compression.
- It has been implemented in a single IC.
- Subimage size: affects transform coding error and computational complexity.
  - Normally it is selected as a power of 2 (simplifies the computations)

Bit allocation

- Bit allocation: which transform coefficients are kept and what is the precision that is used to represent them.
- Retaining coefficients is equivalent to applying a mask to the coefficients. The mask is one where the coefficients are kept and zero where the coefficients are discarded.
- Precision of representation: number of levels of scalar quantizer used for a coefficient
- Retaining the coefficients:
  1. Maximum variance (zonal coding)
  2. Maximum magnitude (threshold coding)

McMaster University

Zonal Coding

- Uses the information theory concept of viewing information as uncertainty.
- Transform coefficients of maximum variance carry the most picture information and should be retained.
- How to calculate the variance:
  1. From the (N/n)(N/n) transformed subimage arrays
  2. An assumed image model
- Coefficients of maximum variance are located around the origin of the transform
- The retained coefficients must be quantized and coded.

Threshold coding

- In zonal coding a single, fixed mask is used for all subimages.
- Threshold coding is adaptive: the location of the transform coefficients retained for each subimage vary from one subimage to another one.
- For each subimage, the transform coefficients of largest magnitude make the most significant contribution to reconstructed subimage quality.
- When the mask is applied to the subimage for which it was derived, the resulting nxn array is reordered (zigzag scanning) and the 1-D sequence which contains long runs of zeros is run-length coded.

Threshold coding

- What is the threshold and how it is obtained?
  1. A single global threshold for all subimages
  2. A single threshold for each subimage
  3. The threshold can be varied as a function of the location of each coefficient within the subimage.

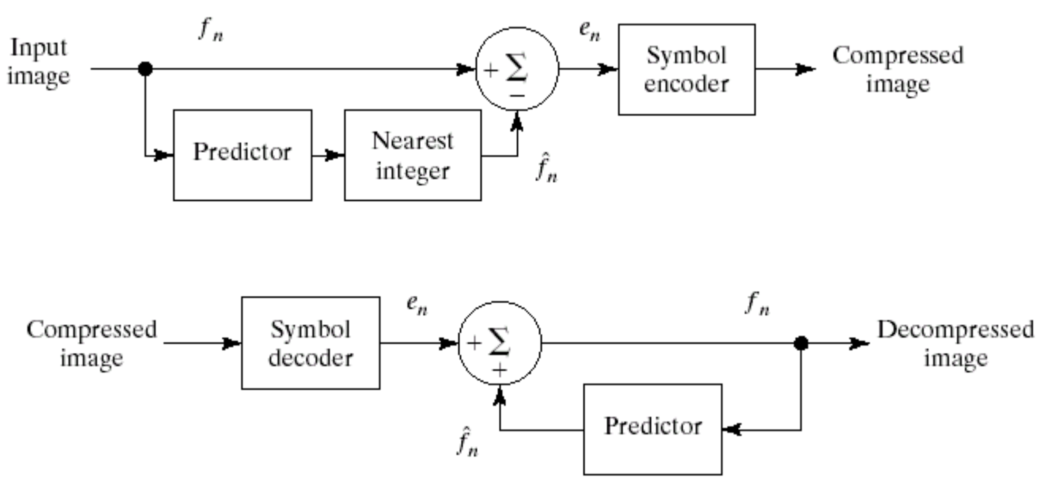
Lossless predictive coding

- In this technique there is no need to decompose the image into bit planes.
- This technique eliminates the interpixel redundancies of closely spaced pixels.
- How? By extracting and coding only the new information in each pixel
- New information: the difference between actual and predicted value of that pixel

McMaster University

Lossless predictive coding

- Predictors in encoder and decoder are the same.
- Various local, global and adaptive methods can be used to generate the prediction



Lossless predictive coding

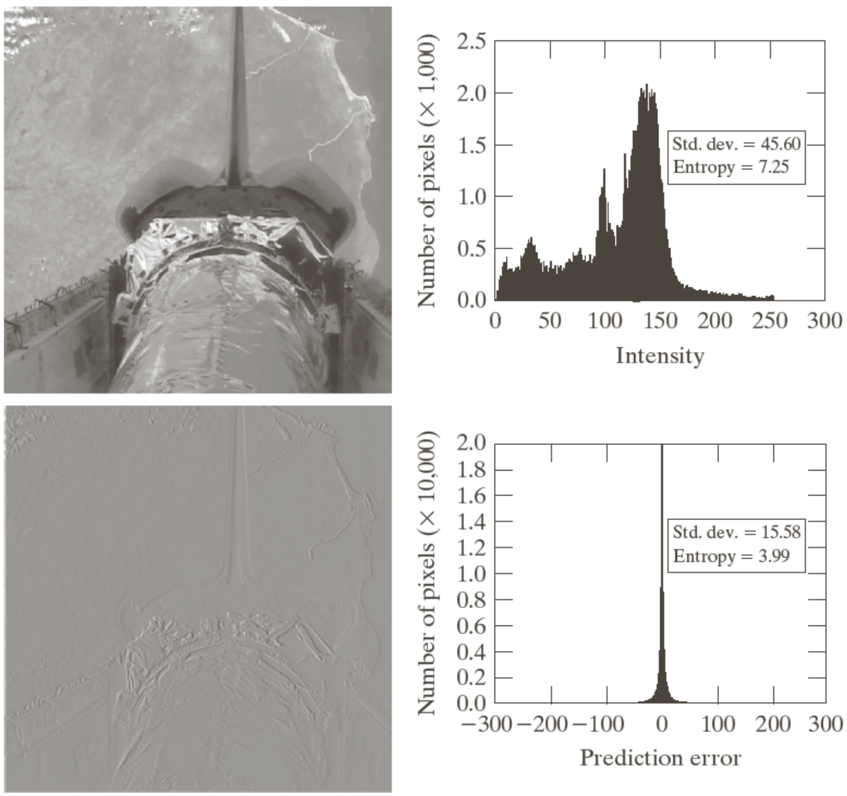
- Linear predictor is common:
- Previous pixels are used to estimate the value of the current pixel
- The previous pixels could be on the same row (column) with the current pixel (1-D prediction) or around the current pixel (2-D)

$$\hat{f}(n) = round\left[\sum_{i=1}^{m} \alpha_i f(n-i)\right]$$

McMaster University

Lossy compression

- Lossy compression: compromising the accuracy of the reconstructed image in exchange for increased compression
- Good reconstructed images even with a compression factor of 30:1.
- The principle difference between lossy and lossless compression is presence or absence of quantizer.



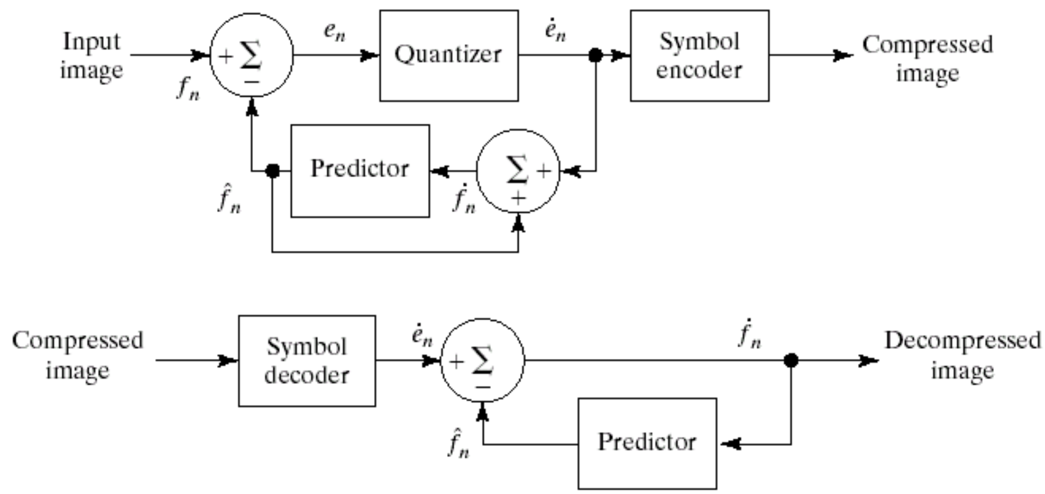
a b
c d

FIGURE 8.34
(a) A view of the Earth from an orbiting space shuttle. (b) The intensity histogram of (a). (c) The prediction error image resulting from Eq. (8.2-34). (d) A histogram of the prediction error. (Original image courtesy of NASA.)

McMaster University

Lossy predictive coding

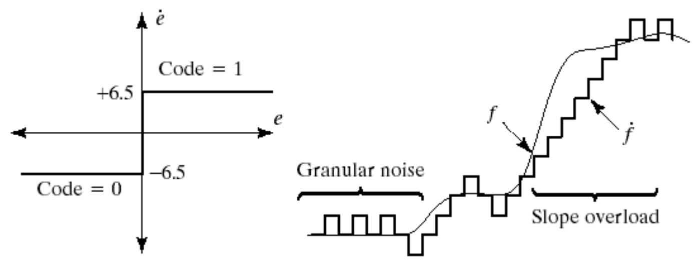
- The prediction at the decoder and encoder should be the same.
- This will prevent error built up at the decoder output



Delta modulator

$$\hat{f}_n = \alpha \dot{f}_{n-1}$$

$$\dot{e}_n = \begin{cases} +\delta \\ -\delta \end{cases}$$



a b
c

FIGURE 8.22 An example of delta modulation.

Input		Encoder				Decoder		Error
n	f	$\hat{f}$	e	$\dot{e}$	$\dot{f}$	$\hat{f}$	$\dot{f}$	$[f - \dot{f}]$
0	14	—	—	—	14.0	—	14.0	0.0
1	15	14.0	1.0	6.5	20.5	14.0	20.5	–5.5
2	14	20.5	–6.5	–6.5	14.0	20.5	14.0	0.0
3	15	14.0	1.0	6.5	20.5	14.0	20.5	–5.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
14	29	20.5	8.5	6.5	27.0	20.5	27.0	2.0
15	37	27.0	10.0	6.5	33.5	27.0	33.5	3.5
16	47	33.5	13.5	6.5	40.0	33.5	40.0	7.0
17	62	40.0	22.0	6.5	46.5	40.0	46.5	15.5
18	75	46.5	28.5	6.5	53.0	46.5	53.0	22.0
19	77	53.0	24.0	6.5	59.6	53.0	59.6	17.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

Optimal quantizer

$t=q(s)$

Quantizer is completely described by t_i and s_i the the first quadrant.

s_i: decision points

t_i: reconstruction levels

Design problem: select s_i and t_i for a particular optimization criterion and input probability density function $p(s)$

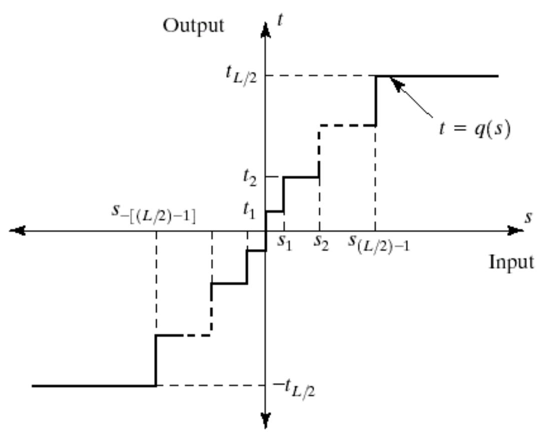


FIGURE 8.25 A typical quantization function.

Optimal quantizer

If mean square error is used and p(s) is assumed to be even then (Lloyd-Max quantizer):

$$\int_{s_{i-1}}^{s_i} (s - t_i) p(s) ds = 0$$

$$s_i = \begin{cases} 0 & i = 0 \\ \dfrac{t_i + t_{i-1}}{2} & i = 1,2,\ldots,L/2-1 \\ \infty & i = \dfrac{L}{2} \end{cases}$$

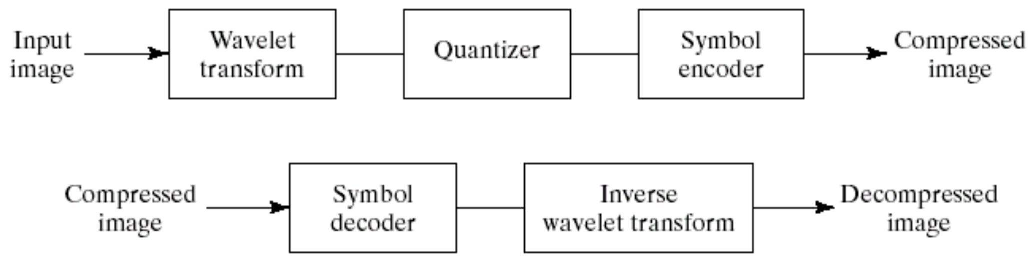
TABLE 8.10
Lloyd-Max quantizers for a Laplacian probability density function of unit variance.

Levels	2		4		8	
i	s_i	t_i	s_i	t_i	s_i	t_i
1	∞	0.707	1.102	0.395	0.504	0.222
2			∞	1.810	1.181	0.785
3					2.285	1.576
4					∞	2.994
θ	1.414		1.087		0.731	

McMaster University

Wavelet coding

- Image is wavelet transformed
- Many of the computed coefficients carry little visual information, they can be quantized.
- The quantization can exploit positional correlation across decomposition levels.

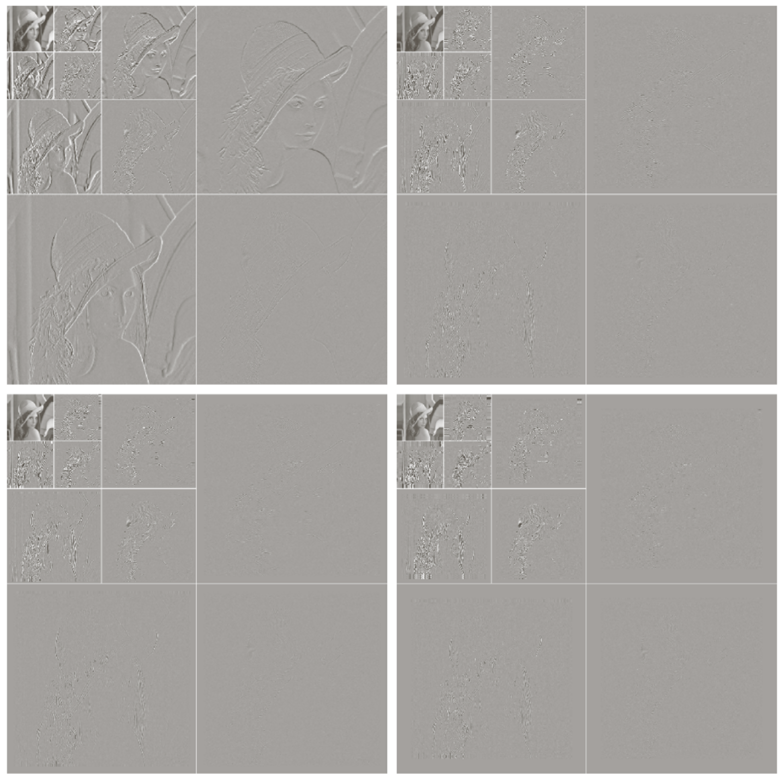


McMaster University

Wavelet coding

- Wavelet functions selected for wavelet compression, affect the performance of wavelet compression system
- The most widely used wavelet functions for compression are Daubechies wavelets.
- Another factor affecting wavelet coding computational complexity and reconstruction error is the number of levels.
- In many applications, the resolution of image and the scale of the lowest useful approximation normally determine the number of transform levels.

Wavelet coding



a b
c d

FIGURE 8.46 Three-scale wavelet transforms of Fig. 8.9(a) with respect to (a) Haar wavelets, (b) Daubechies wavelets, (c) symlets, and (d) Cohen-Daubechies Feauveau biorthogonal wavelets.

Wavelet coding

- Largest factor effecting wavelet compression and reconstruction error is coefficient quantization
- Effectiveness of the quantization can be improved significantly by
  1. introducing an enlarge quantization interval around zero called a dead zone
  2. adapting the size of quantization interval from scale to scale

JPEG 2000

- JPEG2000 extends JPEG to provide increased flexibility in compression and access to compressed data
- First step of encoding process is to DC level shift the samples of the image by subtracting 2^{Ssiz-1} (Ssiz is number of bits representing each pixel)
- After the image has been level shifted it is optionally divided into tiles
- Tiles are rectangular arrays of pixels
- One dimensional discrete wavelet transform of rows and columns of each tile is then computed
- For lossless compression, transform is based on a biorthogonal 5-3 coefficient scaling and wavelet vector

JPEG 2000

- A rounding procedure is defined for non-integer valued transform coefficients
- In lossy compression, a 9-7 coefficient scaling and wavelet vector is employed
- The transform produces four subbands, an approximation, horizontal, vertical and diagonal
- Transform might be repeated on the approximation component
- Coefficients in subbands are quantized
- The final steps of encoding are coefficient bit modeling, arithmetic coding, bit stream layering and packetizing.

JPEG 2000

- Coefficients of each transformed tile subbands are arranged into rectangular blocks called code blocks
- Code blocks are individually coded a bit plane at a time
- Each bit plane is processed in three passes: significant propagation, magnitude refinement, and cleanup
- Outputs are arithmetically coded and grouped with similar passes from other code blocks to form layers

Video compression standards

- Video compression standards can be grouped into two categories:
    1. video teleconferencing standards (H.261, H. 262. H.263)
    2. multimedia standards (MPEG1, MPEG2, MPEG4)
- Both groups are built around a hybrid block-based DPCM/ DCT coding scheme
- There are three basic types of encoded output frames

1. Frames that are coded without any reference to past frames (I frames)

Video compression standards

- Because I frames do not use temporal correlation, the compression rate is low
- I frames provide the highest degree of random access, ease of editing, and greatest resistance to errors

2. Predictive coded (P frames): are coded using motion compensation prediction from the last I or P frame, whichever happens to be closest.
   - Compression efficiency of P frames is substantially higher than I frames.

McMaster University

Video compression standards

3. Bidirectionally predictive coded (B frames): achieve a high level of compression by using motion compensation prediction from the most recent frame and the closest future frame.
   – By using both past and future for prediction, generally we can get better compression than if we only use prediction based on the past.

Video compression standards

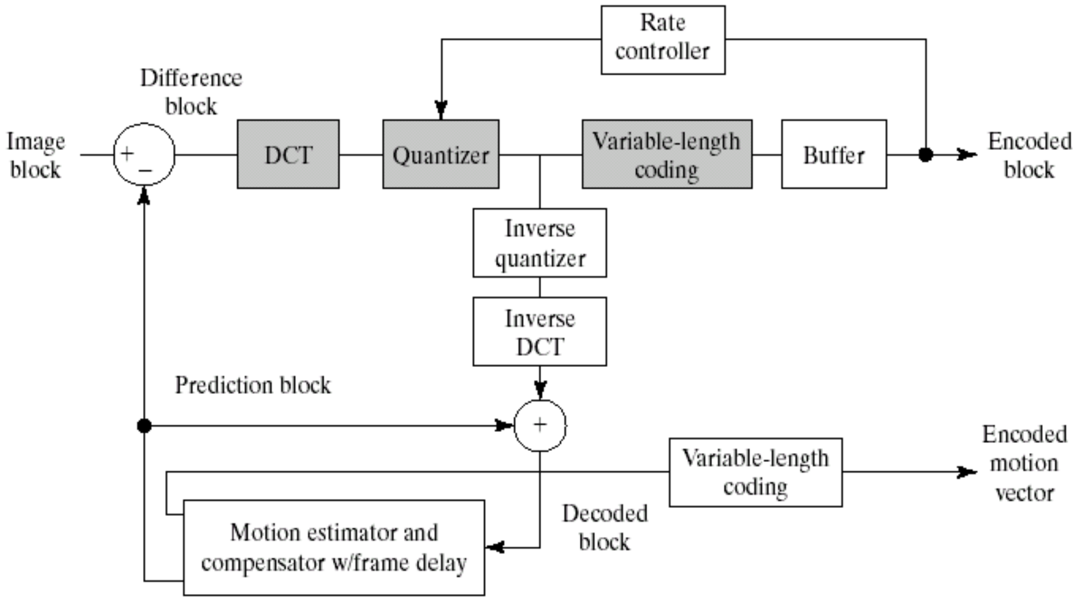


FIGURE 8.47 A basic DPCM/DCT encoder for motion compensated video compression.

Lossy compression

- In a lossless compression, the reconstructed signal is identical to the original signal
- Only a limited amount of compression can be obtained with lossless compression
- There is a floor (entropy of the source) below which we cannot drive the size of the compressed signal
- In some applications consequences of loss of information may prohibit us from loss of information (bank records, medical images)

McMaster University

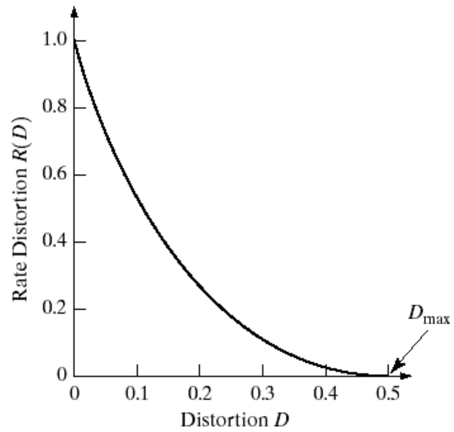
Lossy compression

- If the resources are limited and we do not require absolute integrity, we can improve the amount of compression by accepting certain degree of loss during compression
- Lossless compression: rate is the only performance measure
- In lossy compression rate by itself is not enough
- A distortion measure (difference between original and reconstructed data) is also required
- Goal in lossy compression: incur minimum amount of distortion while compressing to the lowest possible rate
- There is a tradeoff between minimizing rate and keeping distortion small

Source Coding Theorems

- What is the smallest rate, subject to a given distortion, at which the information of the source can be represented?
- For every source there is a Rate-Distortion (RD) function that gives the minimum rate achievable for a given distortion.
- RD function can be computed analytically for simple sources and distortion measures and iteratively for more complex situation.
- RD functions have typical form shown below.
- RD function is a lower bound, and the source coding theorem tells us that we can get as close as possible to this lower bound as we want.

FIGURE 8.10 The rate distortion function for a binary symmetric source.



Error free (lossless) compression

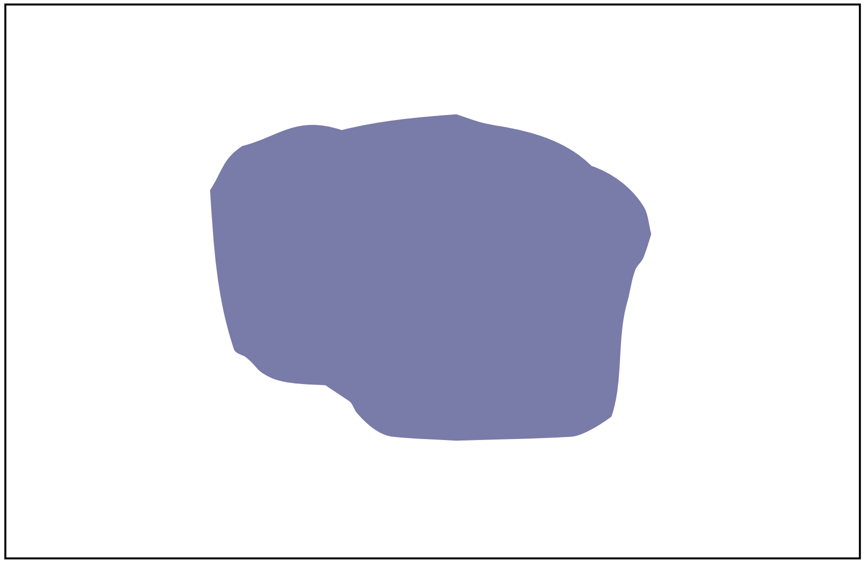
- Applications: archiving of medical or business documents where lossy compression is prohibited for legal reasons.
- Typical compression ratio: 2 to 10
- 2 operations are involved:
  1. Reduce the interpixel redundancies using a mapping or transformation
  2. Coding the representation to eliminate coding redundancies

- Variable Length Coding (VLC)
  – Assigns the shortest possible code words to the most probable gray levels.
  – The input to VLC could be the gray levels of an image or the output of a mapping operation.

McMaster University

Contour tracing and coding

- Different approaches for representing the contour of a binary image:
  1. Set of boundary points
  2. A single boundary point and a set of directions (directional contour tracking)
  3. Predictive differential quantization (PDQ)



PDQ

- Δ’: difference between the starting coordinate of the front contour on adjacent lines
- Δ’’: difference between front to back contour lengths (d1-d2)
- New start: a new contour
- Merge: old contour ends
- If Δ’’ is replace by Δ’’’ (difference between back contour coordinates of adjacent lines) the code is called double delta coding (DDC).

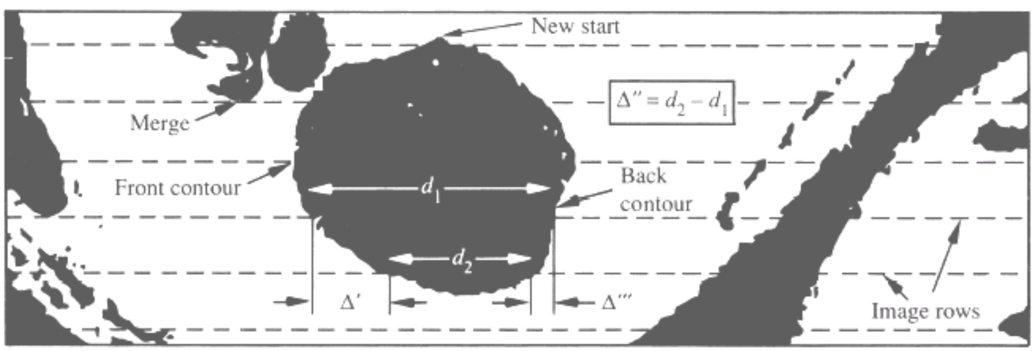


FIGURE 8.18 Parameters of the PDQ algorithm.

Image compression standards

- The standards have been developed under the supervision of ISO and CCITT.
- Binary image compression standards
- The most widely used standards for bilevel image compression are CCITT Group 3 and Group 4.
- They were originally developed for FAX coding.
- Group 3: includes two coding schemes:
    1. 1-D: coding of each line is performed independently of any other line
    2. 2-D: coding of one line is performed using line to line correlation.

G3

- 1-D coding: is a run-length coding scheme in which each line is represented as alternative white and black runs from left to right.
- The first run is always a white run. So, if the first pixel is black, the white run has a length of zero.
- Runs of different lengths occur with different probabilities, therefore they are coded using VLC.
- CCITT uses Huffman coding

G3

- The number of possible length of runs is huge and it is not feasible to build a code book that large.
- The run length rl is expressed as:
  - rl=64*m+t   t=0,1,..,63   m=1,2,…, 27
- To represent rl, we use the corresponding codes for m and t.
- The code for t are called terminating codes and for m make-up codes.
- A unique EOL (end of line) is used to terminate each line.

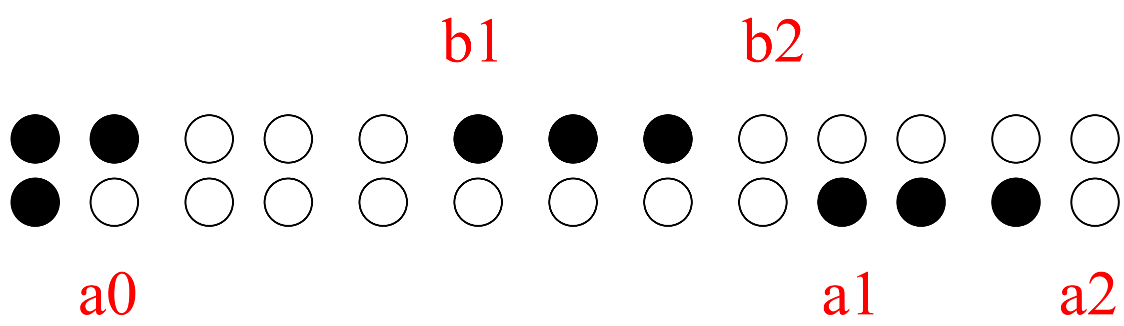
G3

- 2-D coding: rows of a facsimile image are heavily correlated. Therefore it would be easier to code the transition points with reference to the previous line.
- a0: the last pixel known to both encoder and decoder. At the beginning of encoding each line a0 refers to an imaginary white pixel to the left of actual pixel. While it is often a transition pixel it does not have to be.
- a1: the first transition point to the right of a0. It has an opposite color of a0.
- a2: the second transition pixel to the right of a0. Its color is opposite of a1.

McMaster University

G3

- b1: the first transition pixel on the line above currently being coded, to the right of a0 whose color is opposite of a0.
- b2: the first transition pixel to the right of b1 in the line above the current line



G3

- If b1 and b2 lie between a0 and a1 (pass mode): no transition until b2. Then a0 moves to b2 and the coding continues.
- If a1 is detected before b2
  - If distance between a1 and b1 (number of pixels) is less than or equal to 3, we send the location of a1 with respect to b1, move a0 to a1 and the coding continues (vertical mode)
  - If the distance between a1 and b1 is larger than 3, we go back to 1-D run length coding and send the distance between a0 and a1 and a1 and a2 (horizontal mode)

Still image compression

- JPEG: most popular and comprehensive continuous tone, still image compression standard.
- JPEG defines 3 different coding systems:
  1. Baseline coding (lossy, very common)
  2. Extended coding, for greater compression, higher precision or progressive reconstruction applications
  3. Lossless coding

- Baseline system (sequential baseline system)
- Input and output precision is limited to 8 bits.
- Quantized DCT values are restricted to 11 bits.

McMaster University

JPEG

- The image is divided into 8x8 blocks, which are processed from left to right and top to bottom.
- The 64 pixel values in each block are first level shifted by subtracting 128.
- The 2-D DCT of the block is then computed and quantized and reordered using the zigzag pattern.
- The DC coefficient (first DCT coefficient) is differentially coded relative to the DC coefficient of the previous block, and then Huffman coded.
- The rest of coefficients are coded using a variable length code that uses the value of the coefficients and number of zeros before the coefficient.

JPEG

- Since the number of values that the quantized coefficients can assume is large, a Huffman code for such an alphabet is unmanageable.
- The values are divided into ranges (categories) and the categories are Huffman coded.
- The elements in each category is specified by tacking on some extra bits.