## 6.4 Error-Free Compression

- medical or business documents, LANDSAT imagery, digital radiography
- compression ratio : normally 2~10
- equally applicable to both binary and gray-scale image
- two independent operations

  1) devising an alternative representation of image in which its interpixel redundancies are reduced

  2) coding the representation to eliminate coding redundancy

### 6.4.1 Variable-Length Coding

- coding redundancy : eliminated
  - $L_{avg} = \sum_{k=0}^{L-1} l(r_k) p_r(r_k)$ → minimization
  - higher prob. → shorter code length

    lower prob. → longer code length

1) Huffman Coding

- the most popular technique for removing coding redundance
- coding result : optimal

- procedure of Huffman coding
  - source merging

| Original source | | Source reduction | | | |
|---|---|---|---|---|---|
| Symbol | Probability | 1 | 2 | 3 | 4 |
| $a_2$ | 0.4 | 0.4 | 0.4 | 0.4 | 0.6 |
| $a_6$ | 0.3 | 0.3 | 0.3 | 0.3 | 0.4 |
| $a_1$ | 0.1 | 0.1 | 0.2 | 0.3 | |
| $a_4$ | 0.1 | 0.1 | 0.1 | | |
| $a_3$ | 0.06 | 0.1 | | | |
| $a_5$ | 0.04 | | | | |

- source splitting

| Original source | | | Source reduction | | | |
|---|---|---|---|---|---|---|
| **Sym.** | **Prob.** | **Code** | **1** | **2** | **3** | **4** |
| $a_2$ | 0.4 | 1 | 0.4 1 | 0.4 1 | 0.4 1 | 0.6 0 |
| $a_6$ | 0.3 | 00 | 0.3 00 | 0.3 00 | 0.3 00 | 0.4 1 |
| $a_1$ | 0.1 | 011 | 0.1 011 | 0.2 010 | 0.3 01 | |
| $a_4$ | 0.1 | 0100 | 0.1 0100 | 0.1 011 | | |
| $a_3$ | 0.06 | 01010 | 0.1 0101 | | | |
| $a_5$ | 0.04 | 01011 | | | | |

$$L_{avg} = (0.4)(1) + (0.3)(2) + (0.1)(3) + (0.1)(4) + (0.06)(5) + (0.04)(5)$$
$$= 2.2 \text{ bits/symbol}$$

**Table 6.5 Variable-Length Codes**

| *Source Symbol* | *Prob-ability* | *Binary Code* | *Huffman* | *Truncated Huffman* | *$B_2$-Code* | *Binary Shift* | *Huffman Shift* |
|---|---|---|---|---|---|---|---|
| *Block 1* | | | | | | | |
| $a_1$ | 0.2 | 00000 | 10 | 11 | C00 | 000 | 10 |
| $a_2$ | 0.1 | 00001 | 110 | 011 | C01 | 001 | 11 |
| $a_3$ | 0.1 | 00010 | 111 | 0000 | C10 | 010 | 110 |
| $a_4$ | 0.06 | 00011 | 0101 | 0101 | C11 | 011 | 100 |
| $a_5$ | 0.05 | 00100 | 00000 | 00010 | C00C00 | 100 | 101 |
| $a_6$ | 0.05 | 00101 | 00001 | 00011 | C00C01 | 101 | 1110 |
| $a_7$ | 0.05 | 00110 | 00010 | 00100 | C00C10 | 110 | 1111 |
| *Block 2* | | | | | | | |
| $a_8$ | 0.04 | 00111 | 00011 | 00101 | C00C11 | 111 000 | 00 10 |
| $a_9$ | 0.04 | 01000 | 00110 | 00110 | C01C00 | 111 001 | 00 11 |
| $a_{10}$ | 0.04 | 01001 | 00111 | 00111 | C01C01 | 111 010 | 00 110 |
| $a_{11}$ | 0.04 | 01010 | 00100 | 01000 | C01C10 | 111 011 | 00 100 |
| $a_{12}$ | 0.03 | 01011 | 01001 | 01001 | C01C11 | 111 100 | 00 101 |
| $a_{13}$ | 0.03 | 01100 | 01110 | 10 0000 | C10C00 | 111 101 | 00 1110 |
| $a_{14}$ | 0.03 | 01101 | 01111 | 10 0001 | C10C01 | 111 110 | 00 1111 |
| *Block 3* | | | | | | | |
| $a_{15}$ | 0.03 | 01110 | 01100 | 10 0010 | C10C10 | 111 111 000 | 00 00 10 |
| $a_{16}$ | 0.02 | 01111 | 010000 | 10 0011 | C10C11 | 111 111 001 | 00 00 11 |
| $a_{17}$ | 0.02 | 10000 | 010001 | 10 0100 | C11C00 | 111 111 010 | 00 00 110 |
| $a_{18}$ | 0.02 | 10001 | 001010 | 10 0101 | C11C01 | 111 111 011 | 00 00 100 |
| $a_{19}$ | 0.02 | 10010 | 001011 | 10 0110 | C11C10 | 111 111 100 | 00 00 101 |
| $a_{20}$ | 0.02 | 10011 | 011010 | 10 0111 | C11C11 | 111 111 101 | 00 00 1110 |
| $a_{21}$ | 0.01 | 10100 | 011011 | 10 1000 | C00C00C00 | 111 111 110 | 00 00 1111 |
| *Entropy* | 4.0 | | | | | | |
| *Average Length* | | 5.0 | 4.05 | 4.24 | 4.65 | 4.59 | 4.13 |

2) Arithmetic Coding

- Entire input sequence $S_m$ of length m symbols $\xrightarrow{assigned}$ a single arithmetic codeword

- Ex.

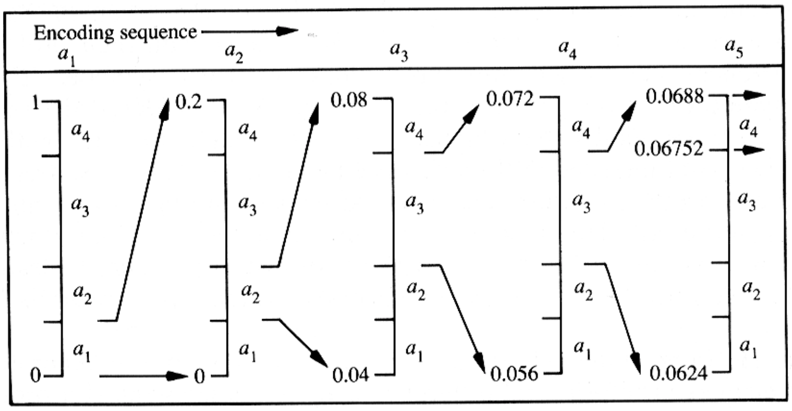



**Table 6.6 Arithmetic Coding Example**

| *Source Symbol* | *Probability* | *Initial Subinterval* |
|---|---|---|
| $a_1$ | 0.2 | [0.0, 0.2) |
| $a_2$ | 0.2 | [0.2, 0.4) |
| $a_3$ | 0.4 | [0.4, 0.8) |
| $a_4$ | 0.2 | [0.8, 1.0) |

- final message symbol → reserved as a special end-of-message indicator
- final subinterval [0.06752, 0.0688]
- any number within this subinterval (ex. 0.68) → can be used to represent the message

6.4.2 Bit-Plane Coding

- remove an image's interpixel redundancies
- procedure
  - multilevel image

    $\xrightarrow{\text{decomposing}}$ a series of binary images

    → compressing each binary images via one several well-known binary compression method

1) Bit-Plane decomposition

- M-bit gray-scale image
  - $a_{m-1}2^{m-1} + a_{m-2}2^{m-2} + \cdots + a_1 2^1 + a_0 2^0$
  - separate the m coeffs. of the polynomial into m 1-bit planes
  - disadvantage :

    small change in gray level

    → significant impact on the complexity of the bit planes

    ex.) gray level 127 (0111111)

    128 (1000000)

- Alternative decomposition method
  - m-bit gray coded image
  - reduce the effect of small gray-level variation
  - gray code

    $g_{m-1} = a_{m-1}$

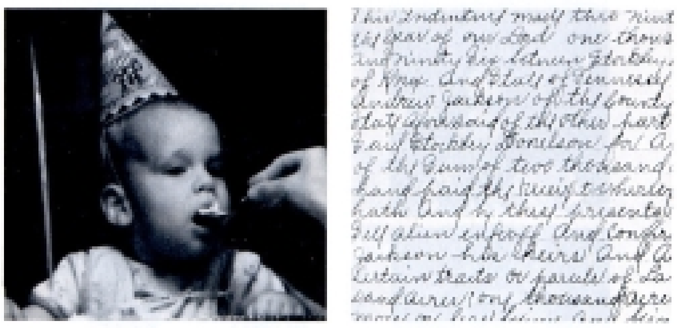
    $g_i = a_i \oplus a_{i+1} \qquad 0 \le i \le m-2$

    where $\oplus$ : XOR
  - successive code word : differ in only one bit position
  - small change in gray level : less likely to affect all m bit planes

    ex.) 127 : 11000000

    128 : 01000000

- Ex.



(a) (b)

*Bit 7*

*Bit 6*

*Bit 5*

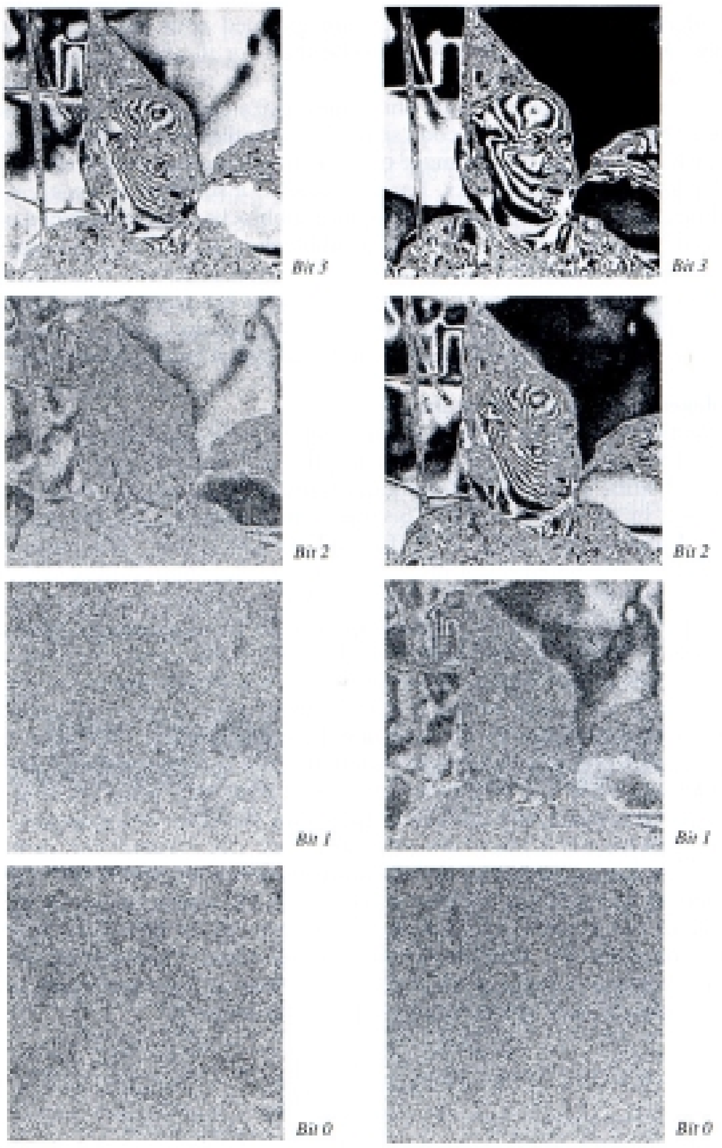
*Bit 4*

*Bit 7*

*Bit 6*

*Bit 5*

*Bit 4*




- high-order bit planes : far less complex, uniform area
- gray level bit planes : less complex than the corresponding binary bit planes

2) Constant area coding

- m× n pixels
  - all white : 0
  - all black : 10
  - mixed intensity : 11 → prefix followed by the mn-bit pattern of the block
- white block skipping (WBS)
  - white areas : 0
  - all other blocks : 1 → prefix

⇒ in the case that predominantly white text documents is compressed

- few solid block areas : grouped with the mixed intensity regions

  : text document 에서는 all black areas 는 거의 없다.

3) One-dimensional run-length coding

After runlength coding, variable length coding is performed to further compression.

Runlength entropy

4) Two-dimensional run-length coding ; omit

5) Contour Tracing and Coding ; omit

**Table 6.7 Error-Free Bit-Plane Coding Results for Fig. 6.14(a): $H \approx 6.82$ Bits/Pixel**

| Method | Bit-Plane Code Rate (Bits/Pixel) | | | | | | | | Code Rate | Compression Ratio |
|---|---|---|---|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| *Binary Bit-Plane Coding* | | | | | | | | | | |
| CBC (4 × 4) | 0.14 | 0.24 | 0.60 | 0.79 | 0.99 | — | — | — | 5.75 | 1.4: 1 |
| RLC | 0.09 | 0.19 | 0.51 | 0.68 | 0.87 | 1.00 | 1.00 | 1.00 | 5.33 | 1.5: 1 |
| PDQ | 0.07 | 0.18 | 0.79 | — | — | — | — | — | 6.04 | 1.3: 1 |
| DDC | 0.07 | 0.18 | 0.79 | — | — | — | — | — | 6.03 | 1.3: 1 |
| RAC | 0.06 | 0.15 | 0.62 | 0.91 | — | — | — | — | 5.17 | 1.4: 1 |
| *Gray Bit-Plane Coding* | | | | | | | | | | |
| CBC (4 × 4) | 0.14 | 0.18 | 0.48 | 0.40 | 0.61 | 0.98 | — | — | 4.80 | 1.7: 1 |
| RLC | 0.09 | 0.13 | 0.40 | 0.33 | 0.51 | 0.85 | 1.00 | 1.00 | 4.29 | 1.9: 1 |
| PDQ | 0.07 | 0.12 | 0.61 | 0.40 | 0.82 | — | — | — | 5.02 | 1.6: 1 |
| DDC | 0.07 | 0.11 | 0.61 | 0.40 | 0.81 | — | — | — | 5.00 | 1.6: 1 |
| RAC | 0.06 | 0.10 | 0.49 | 0.31 | 0.62 | — | — | — | 4.05 | 1.8: 1 |

**Table 6.8 Error-Free Binary Image Compression Results for Fig. 6.14(b): $H \approx$ 0.55 Bits/Pixel**

| | WBS (1 × 8) | WBS (4 × 4) | RLC | PDQ | DDC | RAC |
|---|---|---|---|---|---|---|
| *Code Rate (bits/pixel)* | 0.48 | 0.39 | 0.32 | 0.23 | 0.22 | 0.23 |
| *Compression Ratio* | 2.1: 1 | 2.6: 1 | 3.1: 1 | 4.4: 1 | 4.7: 1 | 4.4: 1 |