Overview

Preview

- A two-hour standard definition (SD)television movie using 720×480×24 bit pixel arrays
- Display the frames sequentially at rates near 30 fps (frames per second)
- SD digital video data must be accessed at 31,104,000 bytes/sec
- A two-hour movie consists of $2.24 \times 10^{1}1$ bytes or 224 GB (gigabytes) of data
- Twenty-seven 8.5 GB dual-layer DVDs (assuming conventional 12 cm disks) are needed to store it.
- To put a two-hour movie on a **single DVD**, each frame must be compressed on average by a factor of **26.3**.
- The compression must be even higher for high definition (HD) television, where image resolutions reach $1920 \times 1080 \times 24$ bits/images

Overview

1. Fundamentals
   - Coding Redundancy
   - Spatial and Temporal Redundancy
   - Irrelevant Information
   - Measuring Image Information
   - Fidelity Criteria
   - Image Compression Models
   - Image Formats, Containers, and Compression Standards
2. Some Basic Compression Methods
   - Huffman Coding
3. Referred Books

Fundamentals

- **Data compression** refers to the process of **reducing the amount of data** required to represent a given **quantity of information**.
- **Data and Information**
- **Redundant data** - contain irrelevant or repeated information.
- **Relative data redundancy**,

$$R = 1 - \frac{1}{C}$$

Where **Compression Ratio**,

$$C = \frac{b}{b'}$$

, b = # bits uncompressed and b = # bits compressed

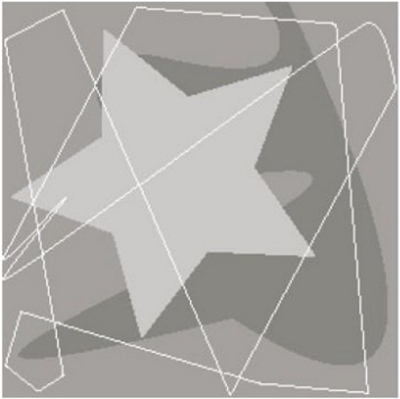
- **For Example** b = 10, b'=1, C = 10, R = 0.9, indicating that 90% of its data is redundant.

Fundamentals

- In image compression, try to preserve information, so that image can be reconstructed:
  - Exactly (loss-less compression)
  - Approximately (lossy compression)
- **Three** principal types of data redundancies -
  1. Coding redundancy
  2. Spatial and temporal redundancy
  3. Irrelevant information

Overview

Coding Redundancy



Coding redundancy: Only a few gray values are present; we can represent using short code words (with few bits)

Coding Redundancy

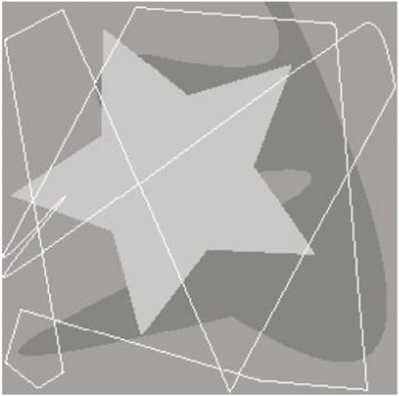
- Use short code words instead of long ones
- Can use variable length codes, so that **most common values** have **shortest codes**.
- $$p_r(r_k) = \frac{n_k}{MN}$$

  where $k = 0, 1, 2, ..., L-1$
- $l(r_k) \rightarrow$ the number of bits used to represent each value of r_k. The average number of bits required to represent each pixel is

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

Coding Redundancy

Computer generated $256 \times 256 \times 8$ bit images



Coding Redundancy

Computer generated $256 \times 256 \times 8$ bit images

TABLE 8.1
Example of variable-length coding.

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

$$L_{avg} = 2(0.25) + 1(0.47) + 3(0.25) + 3(0.03) = 1.81 bits$$

The total number of bits needed to represent the entire image is

$$MNL_{avg} = 256 \times 256 \times 1.81 = 118621$$

Coding Redundancy

Compression Ratio

$$C = \frac{256 \times 256 \times 8}{118621} \approx 4.42$$

Relative Data Redundancy

$$R = 1 - \frac{1}{4.42} = 0.774 = 77.4\%$$

Overview

Measuring Image Information

- **Information of an image** - a minimum amount of data that is sufficient to describe an image without losing information.
- **Information Theory:** a random event E with probability P(E) is said to contain

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

units of information.
- If P(E)=1 (that is, the event always occurs), I(E) = 0 and no information is attributed to it.
- If the base m logarithm is used, the measurement is said to be in **m-ary units**. If the base **2** is selected, the unit of information is the **bit**.

Note: P(E) = 0.5, I(E) = $-\log_2 2^{-1} = 1$. That is, 1 bit is the amount of information conveyed when one of two possible equally likely events occurs. A simple example is flipping a coin and communicating the result.

Measuring Image Information

- Given, we have a set of independent random events, drawn from a possible set of values

$$a_1, a_2, ..., a_j$$

- Well call these the "**source symbols**". Because they are statistically independent, the source itself is called a **zero-memory source**.
- The probabilities of generating these symbols are

$$P(a_1), P(a_2), ..., P(a_j)$$

- The information carried by a single symbol is $-\log(P(a_j))$
- Thus, the average information per symbol, called the **entropy** of the source, is

$$H = \sum_{j=1}^{J} P(a_j) \log(P(a_j))$$

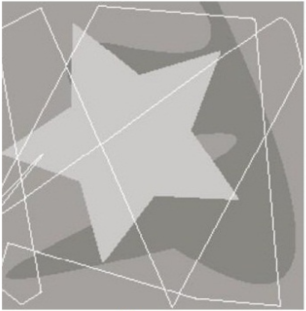
Measuring Image Information

- For image, the imaginary intensity sources entropy becomes

$$\tilde{H} = \sum_{k=0}^{L-1} p_r(r_k)\log(p_r(r_k))$$

- It is not possible to code the intensity values of the imaginary source (and thus the sample image)with fewer than $\tilde{H} bits/pixel$.

Measuring Image Information



(a)

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

$$\begin{aligned}\tilde{H} &= -[0.25 \log_2 0.25 + 0.47 \log_2 0.47 + 0.25 \log_2 0.25 + 0.03 \log_2 0.03] \\ &\approx -[0.25(-2) + 0.47(-1.09) + 0.25(-2) + 0.03(-5.06)] \\ &\approx 1.6614 \text{ bits/pixel}\end{aligned}$$

Overview

Image Compression Models

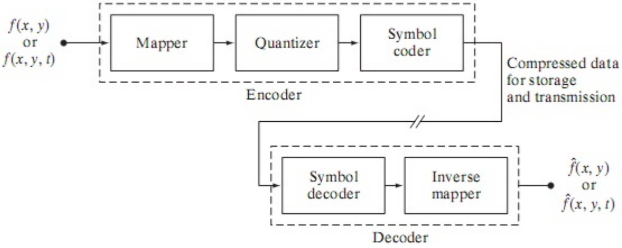


FIGURE 8.5 Functional block diagram of a general image compression system.

Image Compression Models

- An image compression system is composed of two distinct functional components:
  1. an encoder - `performs compression`
  2. a decoder - `performs decompression`
- A **codec** is a device or program that is capable of both encoding and decoding.
- **Video applications:** $f(x, y, t)$ - original image and $\hat{f}(x, y, t)$ - reconstructed image, t represent time.
- If $\hat{f}$ is an exact replica of f, the compression system is called **error free, lossless,or information preserving**.
- If not, the compression system is referred to as **lossy**.

Mapper

- Reduce **spatial and temporal** redundancy
- This operation generally is **reversible** [inverse mapper].
- Example: **Run-length coding**
- **In video application** - uses previous video frames to facilitate the removal of **temporal redundancy**.

Image Compression Models

Quantizer

- Reduces the accuracy of the mappers output in accordance with a **pre-established fidelity criterion**
- Keep **irrelevant information out** of the compressed representation.
- this operation is **irreversible**[omit: in error-free compression]
- In **video applications**, the **bit** rate of the encoded output is often measured (in bits/second) and used to adjust the operation of the quantizer so that a predetermined average output rate is maintained.

Symbol coder

- generates a **fixed- or variable-length code** to represent the quantizer output and maps the output in accordance with the code.
- This operation is **reversible** [symbol decoder].

Some Basic Compression Methods

Loss-less Compression Methods:

- Huffman Coding
- Golomb Coding
- Arithmetic Coding
- LZW Coding
- Run-Length Coding
- Symbol-Based Coding
- Bit-Plane Coding

Lossy Compression Methods:

- Block Transform Coding
- Predictive Coding
- Wavelet Coding
-

Overview

Huffman Coding

FIGURE 8.7
Huffman source reductions.

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				

Huffman Coding

Original source			Source reduction							
Symbol	Probability	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								

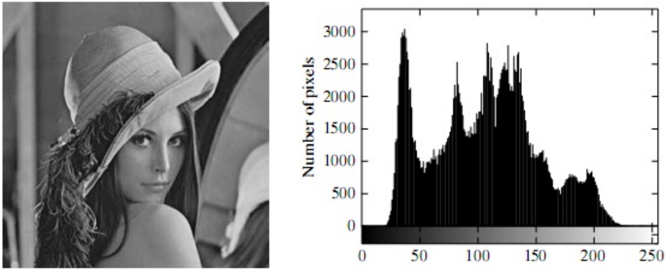
FIGURE 8.8 Huffman code assignment procedure.

The average length of this code is

$$
\begin{aligned}
L_{avg} &= (0.4)(1) + (0.3)(2) + (0.1)(3) + (0.1)(4) + (0.06)(5) + (0.04)(5) \\
&= 2.2 \quad bits/pixel
\end{aligned}
$$

and the entropy of the source is 2.14 bits/symbol .

Huffman Coding



a b

FIGURE 8.9 (a) A 512 × 512 8-bit image, and (b) its histogram.

$L_{avg} = 7.428 \quad bits/pixel$

$$C = \frac{8}{7.428} = 1.077$$

$$R = 1 - \frac{1}{1.077} = 0.0715$$

Thus 7.15% of the original 8-bit fixed-length intensity representation was removed as coding redundancy.

Huffman Coding

Several popular image compression standards, including the **JPEG and MPEG** standards **specify** default Huffman coding tables that have been precomputed based on experimental data.

Overview

Read

Gonzalez & Woods, Digital Image Processing - (3rd edition - March 30, 2007)

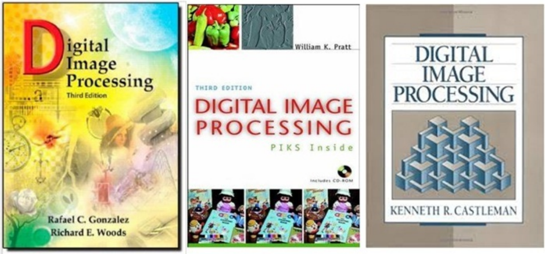
8.1 Fundamentals

8.2 Some Basic Compression Methods

8.2.1 Huffman Coding

Referred Books

- Rafael C. Gonzalez and Richard E. Woods
  Digital Image Processing
  Third Edition
- William K. Pratt
  Digital Image Processing
  Third Edition
- Kenneth R. Castleman
  Digital Image Processing



The End