Digital Image Processing

Image Compression

Recap

Some definitions

- Compression ratio
- Fidelity criteria

Data Redundancy

- Coding
- Interpixel
- Psychovisual

Compression techniques

- Loss-less and Lossy
- Symmetric and Asymmetric

Variable length coding

- Huffman Coding
- Information theoretic analysis

    Entropy

Image Compression

Kraft's inequality

A uniquely decodable code with the codeword lengths $l_1, \ldots, l_N$ exists if and only if

$$\sum_{i=1}^{N} 2^{-l_i} \leq 1$$

Image Compression

Lower Bound

Given that we have a memoryless source X_j and that we code one symbol at a time with a prefix code. Then the mean codeword length $\bar{l}$ (which is equal to the rate) is bounded by

$$\bar{l} \geq -\sum_{i=1}^{L} p_i \cdot \log_2 p_i = H(X_j)$$

Shannon's Coding Theorem

$H(X_j)$ is called the *entropy* of the source.

Image Compression

$$
\begin{aligned}
H(X_j) - \bar{l} &= -\sum_{i=1}^{L} p_i \cdot \log p_i - \sum_{i=1}^{L} p_i \cdot l_i = \sum_{i=1}^{L} p_i \cdot (\log \frac{1}{p_i} - l_i) \\
&= \sum_{i=1}^{L} p_i \cdot (\log \frac{1}{p_i} - \log 2^{l_i}) = \sum_{i=1}^{L} p_i \cdot \log \frac{2^{-l_i}}{p_i} \\
&\le \frac{1}{\ln 2} \sum_{i=1}^{L} p_i \cdot (\frac{2^{-l_i}}{p_i} - 1) = \frac{1}{\ln 2} (\sum_{i=1}^{L} 2^{-l_i} - \sum_{i=1}^{L} p_i) \\
&\le \frac{1}{\ln 2} (1 - 1) = 0
\end{aligned}
$$

where we used the fact that $\ln x \le x - 1$ and Kraft's inequality.

Image Compression

Efficiency of Huffman Coding

$$H(z)/L(z)$$

Variants of Huffman Coding

- Higher order estimate of entropy
- Truncated Huffman Coding
- Dynamic or Adaptive Huffman Coding

Image Compression

Arithmetic Coding

Basic Idea:

a) Like Huffman coding requires prior knowledge of probabilities

b) Unlike Huffman coding, which assigns variable length codes to symbols arithmetic coding assigns codes to a variable group of symbols i.e. the message.

c) There is no one-to-one correspondence between the symbol and its corresponding code word.

d) The code word itself defines a real number within the half-open interval [0,1) and as more symbols are added, the interval is divided into smaller and smaller subintervals, based on the probabilities of the added symbols.

Image Compression

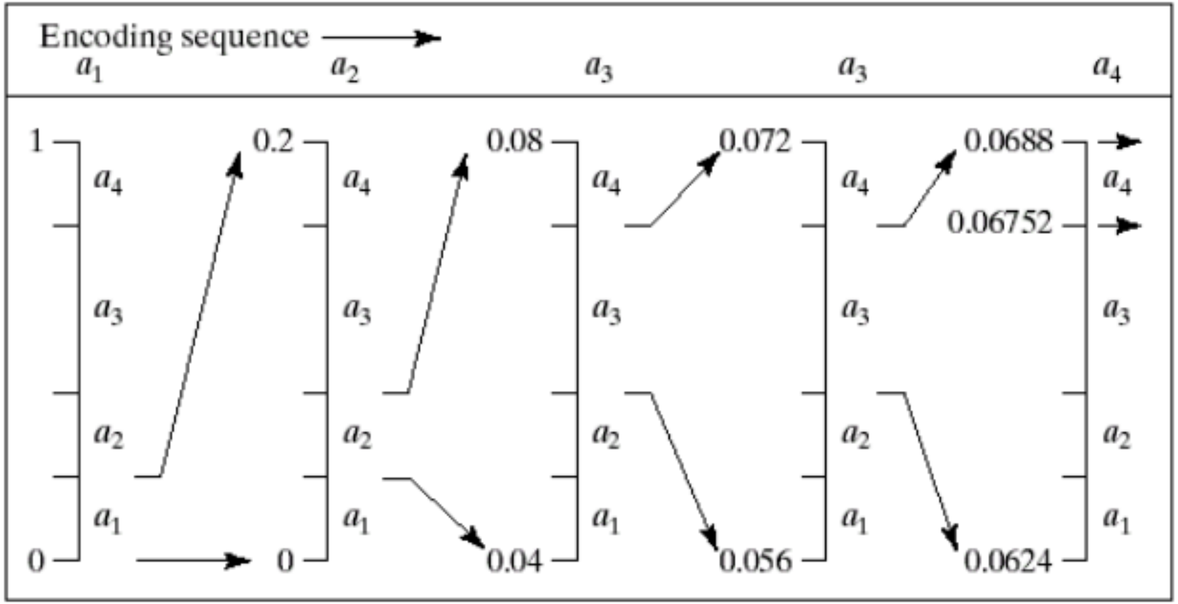
Arithmetic Coding

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)

End of message or length of message is known.

Source: Digital Image Processing, Gonzalez and Woods.

Arithmetic Coding



Final code 068

Source: Digital Image Processing, Gonzalez and Woods.

Arithmetic Decoding

Follows encoding procedure

- Code 068 may be converted to the real number 0.068, which falls in the first sub-interval [0,0.2) therefore first symbol is a1, and so on.

Image Compression

Dictionary based methods

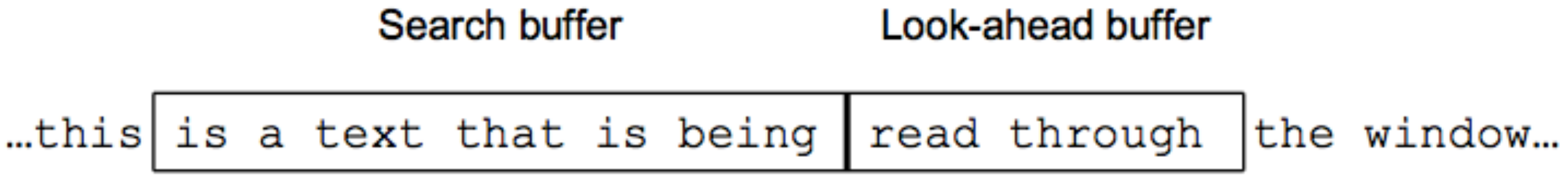
- Compressing multiple strings can be more efficient than compressing single symbols only (e.g. Huffman encoding).
- Strings of symbols are added to a dictionary. Later occurrences are referenced.
- Static dictionary: Entries are predefined and constant according to the application of the text
- Adaptive dictionary: Entries are taken from the text itself and created on-the-fly

Image Compression

Dictionary based methods: LZ77

By Lempel and Ziv in 1977 about lossless compression with an adaptive dictionary.

- Runs through the text in a sliding window
- Two buffers are used - search (history) buffer and a look ahead buffer.
- The search buffer is used as dictionary
- Sizes of these buffers are parameters of the design



Source: http://jens.jm-s.de/comp/LZ77-JensMueller.pdf

Image Compression

Dictionary based methods: LZ77

Encoding of the string:
abracadabrad

output tuple: (offset, length, symbol)

Search ⟷ | Look-ahead ⟷

	7	6	5	4	3	2	1							output
								a	b	r	a	c	ada...	(0,0,a)
							a	b	r	a	c	a	dab...	(0,0,b)
						a	b	r	a	c	a	d	abr...	(0,0,r)
					a	b	r	a	c	a	d	a	bra...	(3,1,c)
			a	b	r	a	c	a	d	a	b	r	ad	(2,1,d)
	a	b	r	a	c	a	d	a	b	r	a	d		(7,4,d)
...ac	a	d	a	b	r	a	d							

Source: http://jens.jm-s.de/comp/ LZ77-JensMueller.pdf

Image Compression

Dictionary based methods: LZ77

Decoding

input		7	6	5	4	3	2	1
(0,0,a)								a
(0,0,b)							a	b
(0,0,r)						a	b	r
(3,1,c)				a	b	r	a	c
(2,1,d)		a	b	r	a	c	a	d
(7,4,d)	abrac	a	d	a	b	r	a	d

Source: http://jens.jm-s.de/comp/
LZ77-JensMueller.pdf

Dictionary based methods: LZW

Extended by Welch (Lempel, Ziv and Welch)

This coding scheme has been adopted in a variety of imaging file formats, such as the graphic interchange format (GIF), tagged image file format (TIFF) and the portable document format (PDF).

Dictionary based methods: LZW

Extended by Welch (Lempel, Ziv and Welch)

- Unlike Huffman coding and arithmetic coding, this coding scheme does not require a priori knowledge of the probabilities of the source symbols.
- The coding is based on a “dictionary” or “codebook” containing the source symbols to be encoded. The coding starts with an initial dictionary, which is enlarged with the arrival of new symbol sequences.
- There is no need to transmit the dictionary from the encoder to the decoder. The decoder builds an identical dictionary during the decoding process

Image Compression

Dictionary based methods: LZW

Extended by Welch (Lempel, Ziv and Welch)

Example: 32 32 34 32 34 32 32 33 32 32 32 34

Consider a dictionary of size 256 locations (numbered 0 to 255) that contains entries corresponding to each pixel intensity value in the range 0-255.

Source: https://nptel.ac.in/courses/117/105/117105083/#

Image Compression

Dictionary based methods: LZW

Extended by Welch (Lempel, Ziv and Welch)

Currently Recognized Sequence	Pixel being processed	Encoded Output	Dictionary Location (Code word)	Dictionary Entry
	32			
32	32	32	256	32-32
32	34	32	257	32-34
34	32	34	258	34-32
32	34			
32-34	32	257	259	32-34-32
32	32			
32-32	33	256	260	32-33
33	32	33	261	33-32
32	32			
32-32	32	256	262	32-32-32
32	34			
32-34		257		

Source: https://nptel.ac.in/courses/117/105/117105083/#

Image Compression

Run Length Coding

Run: a string of the same symbol

Example

input: AAABBCCCCCCCCCAA
output: A3B2C9A2

compression ratio = 16/8 = 2

Image Compression

Predictive Coding

Basic premise: Current pixel is similar to the previous pixel (coherence)

Differential Coding

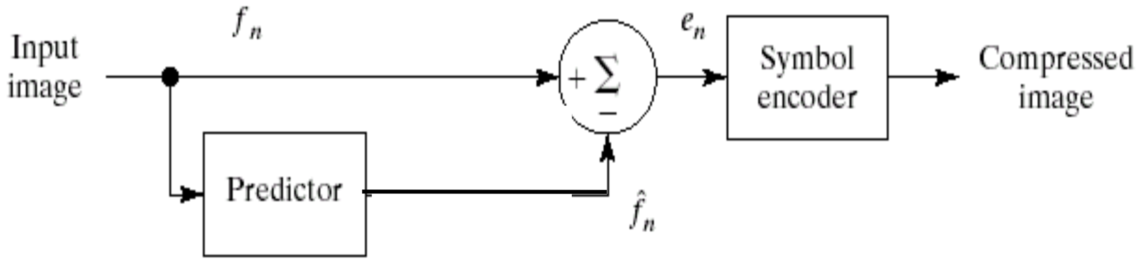
$$d(x,y) = I(x,y) - I(x-1,y)$$

$d(x,y)$ prediction error which is to be encoded.

Image Compression

Predictive Coding

Compression

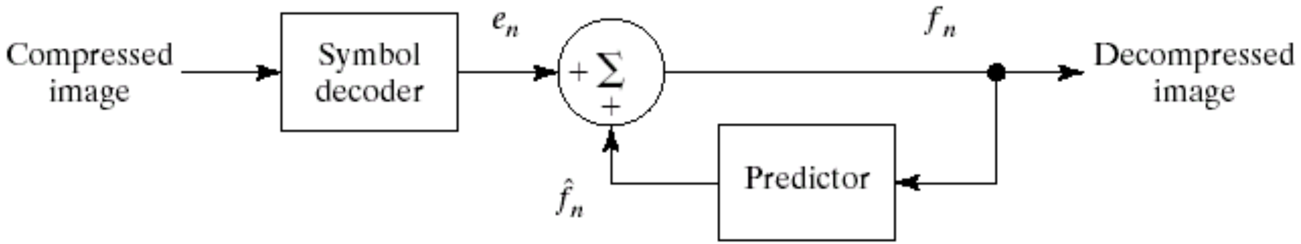


Source: Digital Image Processing, Gonzalez and Woods.

Image Compression

Predictive Coding

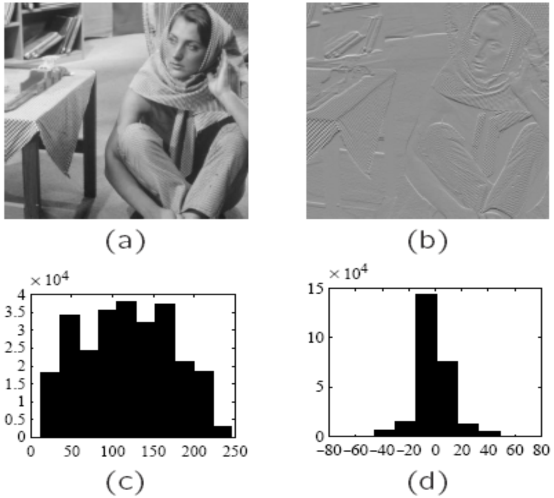
Decompression



Source: Digital Image Processing, Gonzalez and Woods.

Image Compression

Predictive Coding



Distributions for Original versus Derivative Images. (a,b): Original gray-level image and its partial derivative image; (c,d): Histograms for original and derivative images.