Image Compression

EE 604

Govind Sharma

Department of Electrical Engineering
Indian Institute of Technology, Kanpur

3/4/2019

Outline

Image Compression

Images and videos in raw (uncompressed) format have very large data

1	Image/Video	Size
2	One page of text	16 kb
3	Standard 640 $\times$ 480, 24-bit colour image	7.3 Mb
4	FHD 1920 $\times$ 1080, 24-bit colour image	49.3 Mb
5	Animation	50 Mbps
6	Standard video	227.6 Mbps

Transmission and storage of such large data is not easy. So compression techniques are used.

Principles of Compression

Compression is achieved by exploiting one or more of the following

(a) Coding redundancy

(b) Spatial and Temporal redundancy

(c) Information irrelevant to human visual system

Types of compression

(a) Lossless compression: reversible,information preserving. Text and binary images.

(b) Lossy compression: irreversible. Most image and video compression methods.

(c) Nearly lossless compression: for medical images.

Goals of compression

(a) Given a data amount, achieve the best quality.

(b) Given allowed distortion, minimize the data amount.

Some distortion measures

(a) Mean absolute error, MAE$=\frac{1}{N}\sum_{i=0}^{N-1}|x[i]-y[i]|$.

(b) Mean squared error, MSE$=\frac{1}{N}\sum_{i=0}^{N-1}|x[i]-y[i]|^2$.

(c) Peak signal to noise ratio, PSNR$=10\log_{10}\left(\frac{A^2}{MSE}\right)$ where A is peak intensity value.

Some compression standards names

(a) For images: CCITT group 3, CCITT group-4, JBIG1, JBIG2, JPEG, JPEG-LS, JPEG-2000, BMP, GIF, PDF, PNG, TIFF etc.

(b) For video: DV, H.261, H.262, H.263, H.264, MPEG-1, MPEG-2, MPEG-4, MPEG-4 AVC, AVS, HDV, M-JPEG, VC-1, WMV9 etc.

Average length and entropy

Consider a source of N discrete symbols $X_n, \quad n = 1, 2 \ldots, N$ each taking one of M discrete values.

$$P(X_j = a_n) = p_n, \quad n = 1, 2, \ldots, M$$

If the number of bits used to represent a_n are equal to $l(n)$ then average number of bits per symbol is

$$L_{avg} = \sum_{n=1}^{M} p_n l(a_n)$$

Average length and entropy

Assuming symbols to be IID, the entropy is given by

$$H = -\sum_{n=1}^{M} p_n \log_2 p_n$$

Shannon's source coding theorem says that a code can be designed so that

$$-\log_2 p_n \leq l(a_n) \leq -\log_2 p_n + 1$$

Multiplying by p_n and summing we get that

$$H \leq L_{avg} \leq H + 1$$

Huffman code

Shannon's theorem gives a bound but does not show how to achieve this. Huffman coding algorithm gives a code that satisfies this bound. It is **instantaneous,uniquely decodable block code**.

Example Huffman code

Symbol	*Probability*	*Code*
a_2	0.4	1
a_6	0.3	00
a_1	0.1	011
a_4	0.1	0100
a_3	0.06	01010
a_5	0.04	01011

$L_{avg} = 2.2$ bits/symbol. While $log_2 6 = 2.585$ and $H = 2.14$ bits.

Example vector Huffman code

Symbol	*Probability*	*Code*
a	$2/3$	1
b	$1/6$	01
c	$1/6$	00

$L_{avg} = 4/3$ bits/symbol.

Example vector Huffman code

Symbol	*Probability*	*Code*
aa	4/9	1
ab	1/9	011
ac	1/9	0101
ba	2/9	0100
bb	1/36	00111
bc	1/36	00110
ca	1/9	000
cb	1/36	00101
cc	1/36	00100

$L_{avg} = 1.27$ bits/symbol.

Disadvantages of Huffman code

(a) For large alphabet size M requires large calculations.

(b) Requires knowledge of probabilities.

(c) Coding tables also needed to be sent.

(d) JPEG, MPEG etc have default Huffman coding tables.

Some other coding schemes

(a) Arithmetic coding.
(b) Golomb and Golomb-Rice coding.
(c) Lempel, Ziv, Welch (LZW) coding.
(d) Run length coding.
(e) Predictive coding

Run length coding

Identical consecutive symbol values are called runs. A run length pair indicates value of new intensity and run length. Most useful for binary images.

(a) 1-D run length coding: Assume that first intensity is always 1, run length may be zero. Alternate code words indicate runs of one and zeroes. A special code indicates end of line (EOL). Run lengths are Huffman coded and runs of one and zeroes may use different Huffman codes.

(b) 2-D run length codes: Uses relative addresses relative to previous line. For first line previous line is assumed to be all ones.

Bit plane coding, Symbol based coding

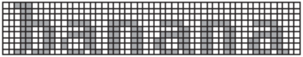
Bit plane coding: Images with more than two intensities can be coded using bit plane coding. An image with M bit intensity value can be considered as composed of M binary images by taking bits with same weightages. Intensities are generally Gray coded. BMP uses run length coding.

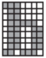
Symbol based coding: An image is represented as collection of frequently occurring sub-images called tokens or symbols. Each symbol is stored in a dictionary. An image is coded as sequence of triplets, first two codes give position information and third code gives location in dictionary.

FIGURE 8.19 (a) A 256-bit monochrome image. (b)–(h) The four most significant binary and Gray-coded bit planes of the image in (a).

a b c

FIGURE 8.17
(a) A bi-level document,
(b) symbol dictionary, and
(c) the triplets used to locate the symbols in the document.



Token	Symbol
0	
1	
2	

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

Predictive coding

Similar to DPCM.

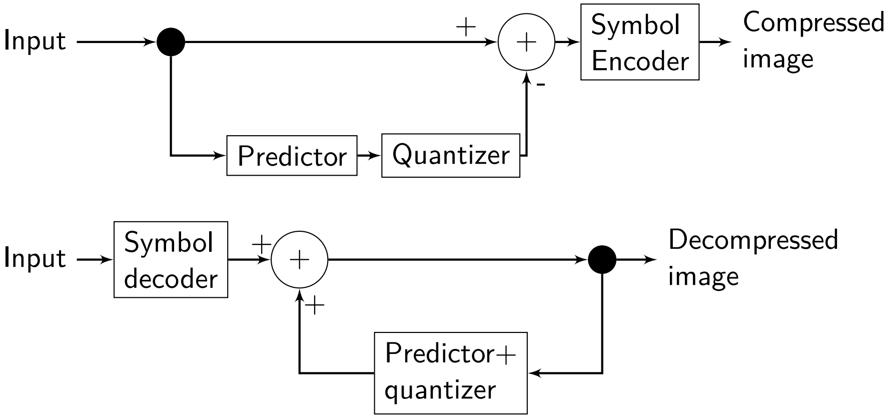
(a) Current pixel value can be predicted from neighboring pixels.

(b) Prediction error has nonuniform distribution near zero and can be specified with less number of bits.

(c) Prediction can be based on current line (1-D) prediction) or based on current and previous line (2-D prediction).

(d) In video coding motion compensated residuals are codded, giving large compression ratio.

(e) Prediction can be done using various criteria like linear minimum mean squared error (LMMSE) criteria or least squares criteria.


Input
+
+
Symbol
Encoder
Compressed
image
-
Predictor
Quantizer
Input
Symbol
decoder
+
+
+
Decompressed
image
Predictor+
quantizer

Transform coding

Lossless image compression has generally small values. For large compression we go for lossy compression. Block diagram of transform based compression is shown below:

Example using 2-D DCT

Image block-128

54	68	71	73	75	73	71	45
47	52	48	14	20	24	20	-8
20	-10	-5	-13	-14	-21	-20	-28
-13	-18	-18	-16	-23	-19	-27	-28
-24	-22	-22	-26	-24	-33	-20	-23
-29	-13	3	-24	-10	-42	-41	5
-16	26	26	-21	12	-31	-40	23
17	30	50	-5	4	12	10	5

Example using 2-D DCT

2-D DCT of the block

31.00	51.70	1.16	-24.58	-12.00	-25.75	11.96	23.28
113.57	6.97	-13.90	43.20	-6.09	35.59	-13.36	-13.00
195.58	10.13	-8.66	-2.93	-28.98	-7.93	0.87	9.55
35.87	-24.30	-15.57	-20.79	11.64	-19.10	-8.53	0.51
40.75	-20.55	-13.66	17.06	-14.25	22.38	-4.89	-11.36
7.19	-13.57	-7.59	-11.94	18.25	-16.26	-1.41	-3.50
-1.45	-13.32	-0.87	1.32	10.38	16.07	4.41	1.11
-6.77	-2.83	4.11	1.11	10.55	-2.73	-3.23	1.57

Low-low coefficients are bigger than high-high coefficients.

Example using 2-D DCT

Normalization matrix

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Example using 2-D DCT

$$QDCT[i,j] = round\left(\frac{DCT[i,j]}{N[i,j]}\right)$$

2	5	0	-2	0	-1	0	0
9	1	-1	2	0	1	0	0
14	1	-1	0	-1	0	0	0
3	-1	-1	-1	0	0	0	0
2	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Only 19 out of 64 remain non-zero.

Example using 2-D DCT

Zig-zag Scanning pattern

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

Example using 2-D DCT

Run-length symbol representation:

$$\begin{aligned} Code = \{&2, (0,5), (0,9), (0,14), (0,1), (1,-2), (0,-1), (0,1), \\ &(0,3), (0,2), (0,-1), (0,-1), (0,2), (1,-1), (2,-1), \\ &(0,-1), (4,-1), (0,-1), (0,1), EOB\} \end{aligned}$$

This is Huffman coded using coding tables. Compression of the order of 6 is obtained.

a b c
d e f

FIGURE 8.31 Approximations of Fig. 8.9(a) using the DCT and normalization array of Fig. 8.30(b): (a) **Z**, (b) 2**Z**, (c) 4**Z**, (d) 8**Z**, (e) 16**Z**, and (f) 32**Z**.

JPEG versus JPEG-2000

Pros of JPEG

(a) Low complexity
(b) Memory efficient
(c) Reasonable coding efficiency

Cons of JPEG

(a) Single resolution
(b) Single quality
(c) No target bit rate
(d) Blocking artifacts at low bit rate
(e) No lossless capability
(f) Poor error resilience
(g) No region of interest

JPEG versus JPEG-2000

JPEG-2000

(a) Improved coding efficiency
(b) Full quality scalability from lossless to lossy at different bit rates.
(c) Spatial scalabilility
(d) Improved error resilience
(e) Regions of interest can be coded with better quality
(f) More demanding in memory and computation resources

Scalability

(a) Same image may be accessed by users with different link speeds and display qualities. High resolution display with high speed link or small display with mobile radio link.

(b) We can have different versions for each desirable bit rate and quality. This is not efficient.

(c) Scalability: Asingle bit stream that can be accessed and decoded partially.

There are two kinds of scalabilities

(a) Quality scalability: Images having different qualities. Sizes may be same or different.

(b) Spatial scalability: Images have different sizes

Wavelet coding

The block diagram is same as before. Instead of DCT we now use wavelet transform. Spatial scalability is achieved by reconstructing only the approximate image. Quality scalability is achieved by decoding only the partial bit planes.

Wavelet coding

Commonly used wavelets are Daubechies 9/7 bi-orthogonal wavelet for lossy compression and LeGall 5/3 bi-orthogonal wavelet for lossless compression. LeGall 5/3 bi-orthogonal wavelet can be implemented using integer operations only.
Instead of square blocks standard allows for rectangular blocks called tiles. Each tile is coded separately.

Wavelet coding

(a) Quantization: Each sub-band may use different step size. It can be skipped for lossless coding.

(b) Entropy coding: Bit plane coding is used. Most significant bit plane is coded first. Acontext based arithmetic coding is used. Quality scalability is achieved by decoding some bit planes starting from MSB.

(c) Region of interest (ROI): Selected region is coded with high accuracy (e.g. face)

There are two kinds of scalabilities

(a) Quality scalability: Images having different qualities. Sizes may be same or different.

(b) Spatial scalability: Images have different sizes

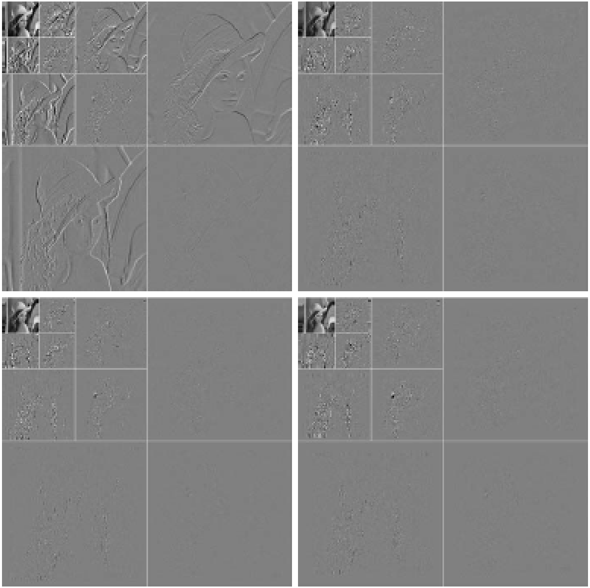


FIGURE 8.46 Three-scale wavelet transforms of Fig. 8.9(a) with respect to (a) Haar wavelets, (b) Daubechies wavelets, (c) symlets, and (d) Cohen-Daubechies-Feauveau biorthogonal wavelets.

(a) (b)

▲ *21. Reconstructed image "ski" after compression at 0.25 b/p by means of (a) JPEG and (b) JPEG 2000.*