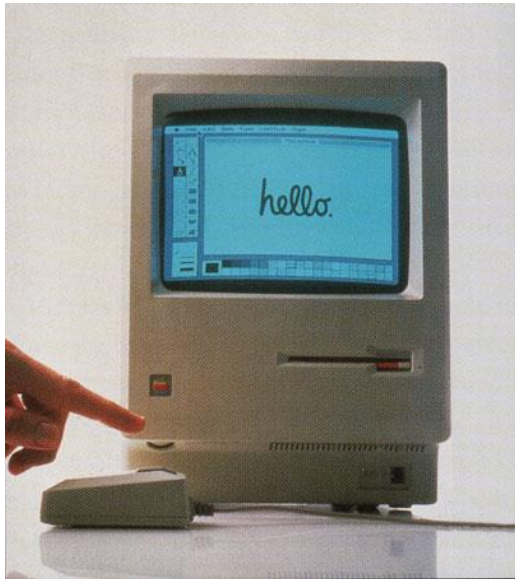
Unit 1. Digital Media Representation and Compression Methods

CS 3570
Shang-Hong Lai
CS Dept., NTHU

What is Multimedia?

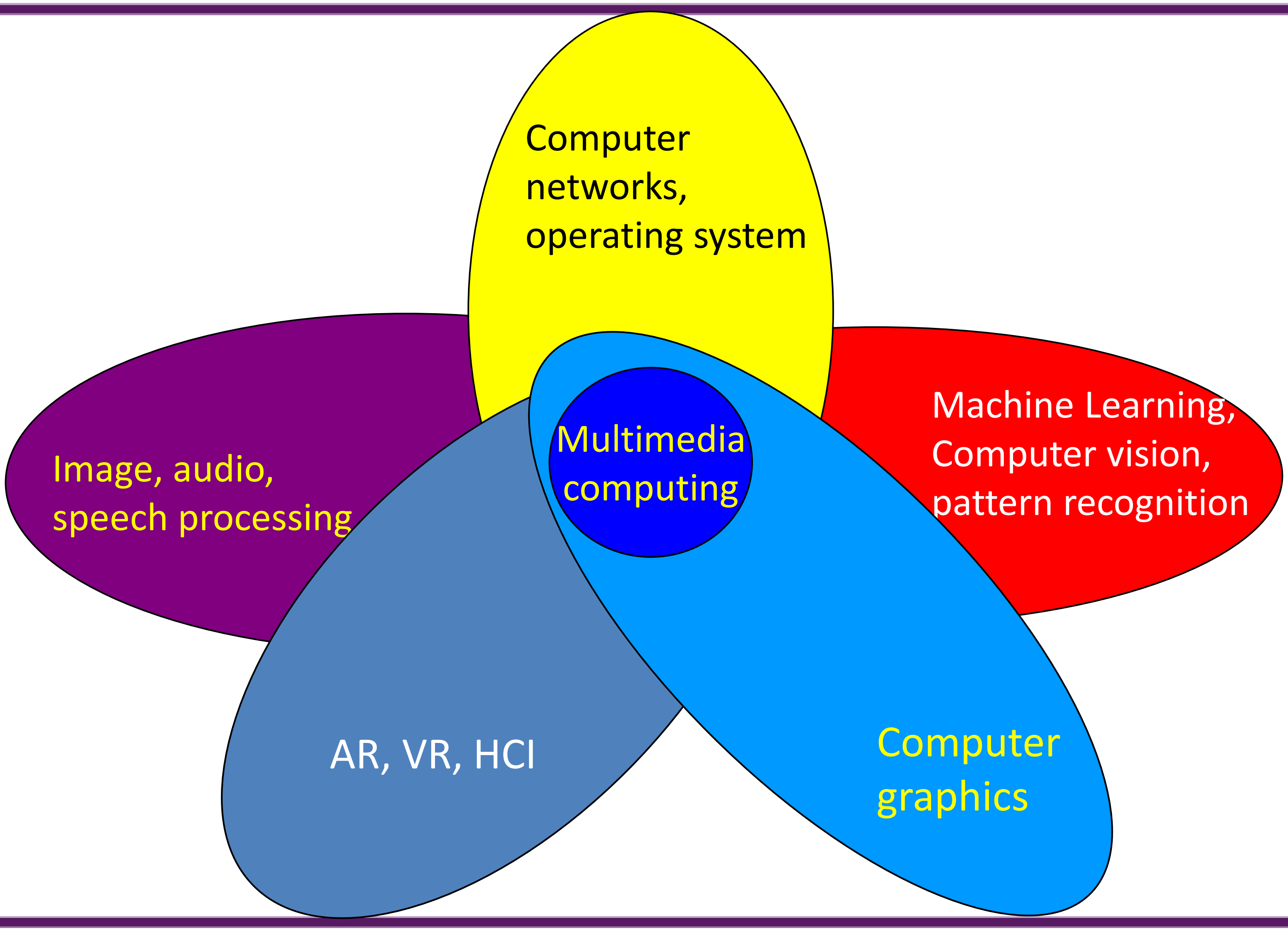
- “Multimedia” has no strict definition.
- In our context, multimedia indicates the computer technology (multimedia computing) for more efficient communication by using different media types:
  - Text
  - Audio and speech
  - Images
  - Graphics
  - Video



Introduction

- Digital media is multimedia driven by computers. You can see it, hear it, maybe even touch it, and certainly interact with it.
- We consider the topics in digital media work—choosing color modes, compressing files, identifying aliased frequencies, filtering, enhancing, transforming, creatively editing, and recognition.

Multimedia is Multidisciplinary


Computer networks, operating system
Image, audio, speech processing
Multimedia computing
Machine Learning, Computer vision, pattern recognition
AR, VR, HCI
Computer graphics

Multimedia Systems

- A Multimedia System is a system capable of processing multimedia data and applications.
- A multimedia system is characterized by the processing, storage, generation, manipulation and rendering of multimedia information.

Characteristics of a Multimedia System

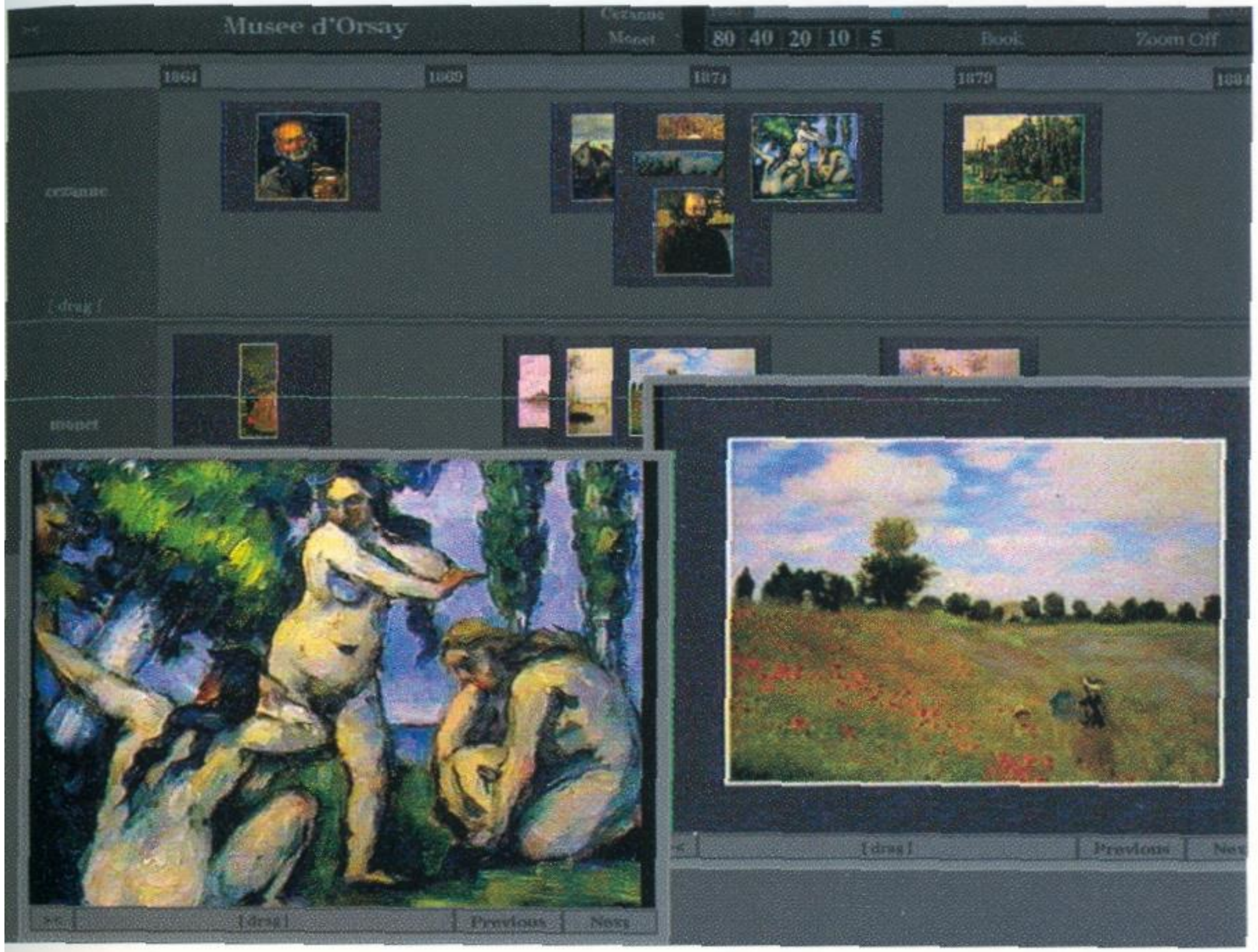
- A multimedia system has the following basic characteristics:
  - Multimedia systems must be computer controlled.
  - Multimedia systems are integrated.
  - The information they handle must be represented digitally.
  - The interface to the final presentation of media is usually interactive.

Multimedia Applications

- World Wide Web
- Hypermedia courseware
- Video conferencing
- Video-on-demand
- Interactive TV
- Home shopping
- Computer games
- Virtual reality
- Digital video editing and production systems
- …..

Example Multimedia System

Chronoscope: Musée d'Orsay



The *Chronoscope* application spreads out an artist's works on a timeline. Paintings by different artists of the same period can be studied side-by-side to explore the cross-fertilization of ideas. Based on impressionist paintings from 1848 to 1914, in the collection of the Musée d'Orsay in Paris, France.
Interface: Matthew Hodges.
Content: Musée d'Orsay, Paris, France.

Chronoscope in MIT's Project Athena

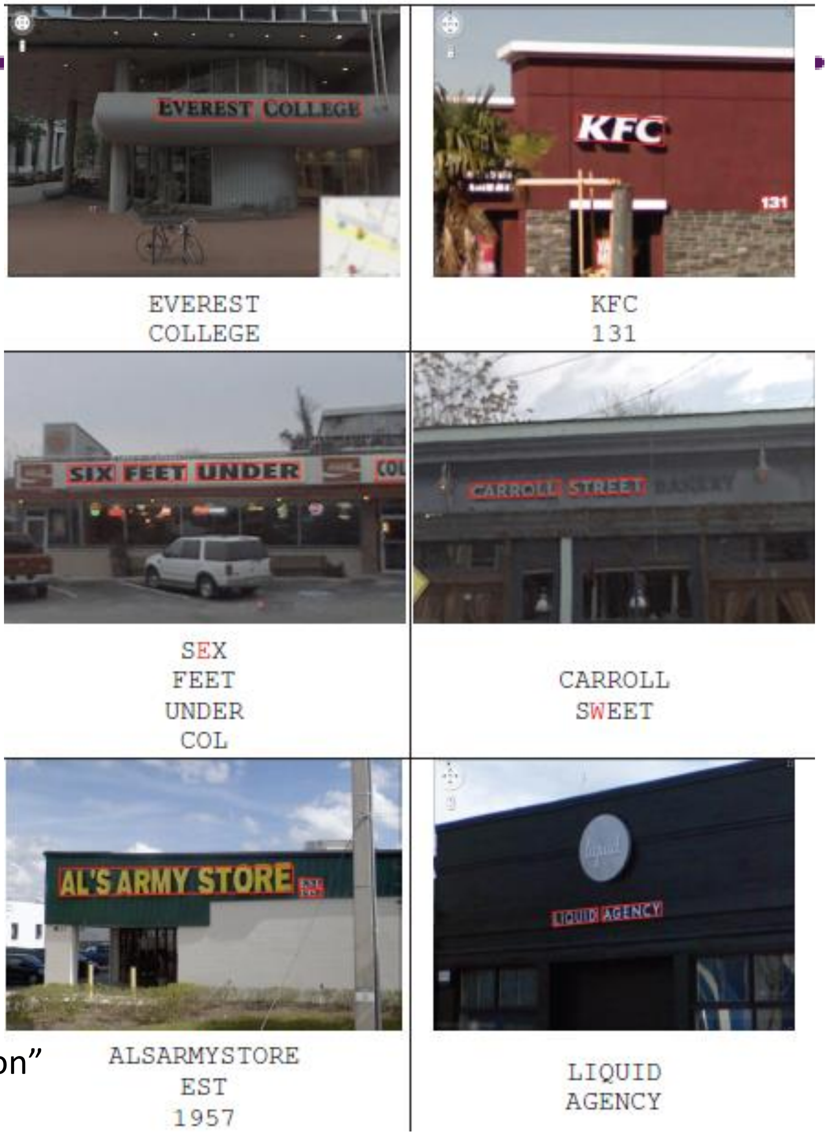
Example Multimedia System

Training in VR: flight simulation

A flight simulator used by the US Air Force (photo by Javier Garcia). The user sits in a physical cockpit while being surrounded by displays that show the environment.

Example-Text

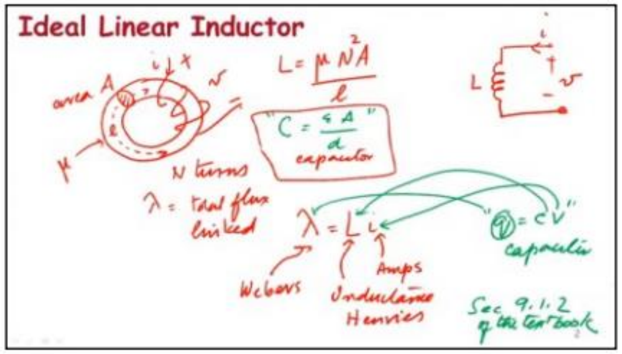
- Text detection
  - Detect text information from images and then recognize the text.



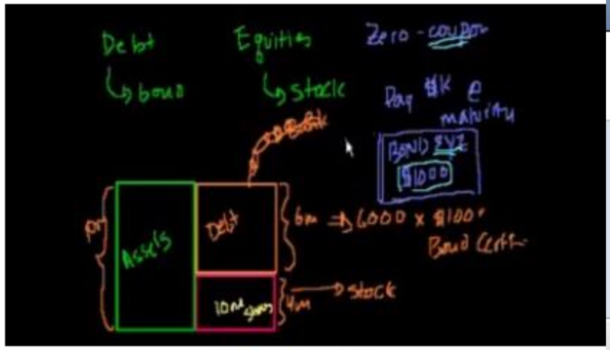
Images from papers "Real-Time Scene Text Localization and Recognition" and "Detecting Texts of Arbitrary Orientations in Natural Images".

Example-Text

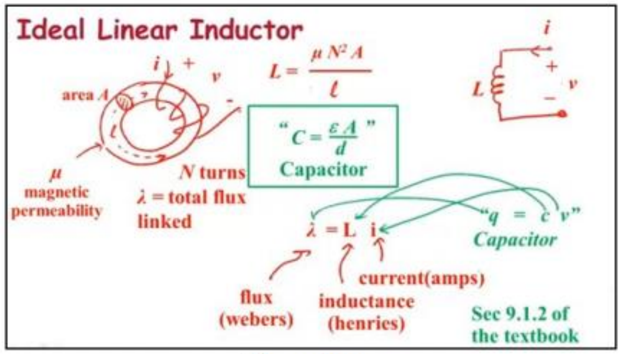
- Handwriting recognition



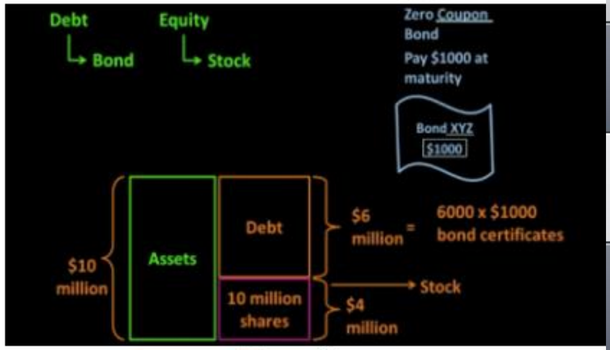
Handwriting



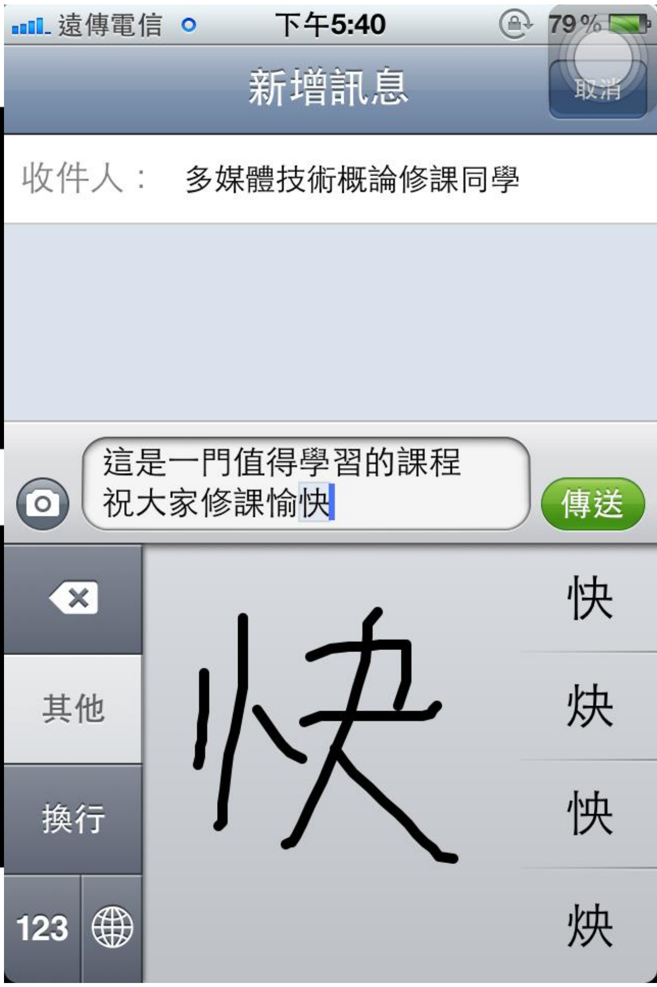
Handwriting



Typeface



Typeface


遠傳電信 下午5:40 79%

新增訊息 取消

收件人： 多媒體技術概論修課同學

這是一門值得學習的課程
祝大家修課愉快

Images from papers "TypeRighting: Combining the Benefits of Handwriting and Typeface in Online Educational Videos".

Audio

- Audio signals are continuous analog signals.
- Input: microphones and then digitized and stored CD Quality Audio requires 16-bit sampling at 44.1 KHz Even higher audiophile rates (e.g. 24-bit, 96 KHz)
- 1 Minute of Mono CD quality (uncompressed) audio requires 5 MB.
- 1 Minute of Stereo CD quality (uncompressed) audio requires 10 MB.
- Usually compressed (E.g. MP3, AAC, Flac, Ogg Vorbis).

Example-Audio and Speech

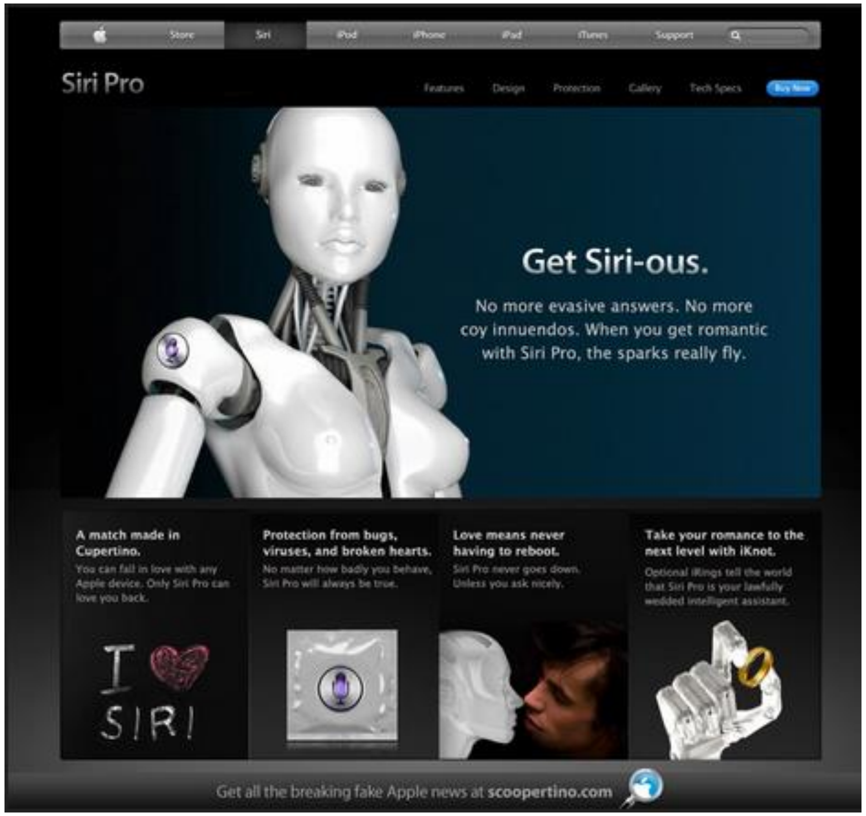
- Speech recognition

Sorry, I don't understand 'Can we be friends'.

Search the web

“ I love you ”

I hope you don't say that to those other mobile phones, Goran.



Siri Pro

Get Siri-ous.

No more evasive answers. No more coy innuendos. When you get romantic with Siri Pro, the sparks really fly.

A match made in Cupertino. You can fall in love with any Apple device. Only Siri Pro can love you back.

Protection from bugs, viruses, and broken hearts. No matter how badly you behave, Siri Pro will always be true.

Love means never having to reboot. Siri Pro never goes down. Unless you ask nicely.

Take your romance to the next level with iKnot. Optional iRings tell the world that Siri Pro is your lawfully wedded intelligent assistant.

Get all the breaking fake Apple news at scoopertino.com

- Siri was developed by integrating speech recognition and artificial intelligence techniques

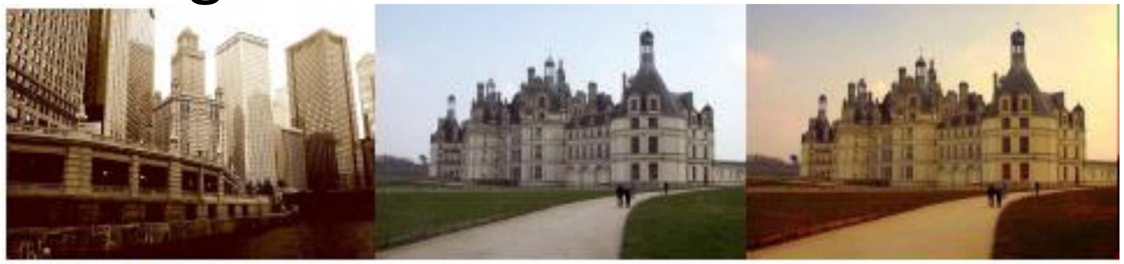
Images from google.

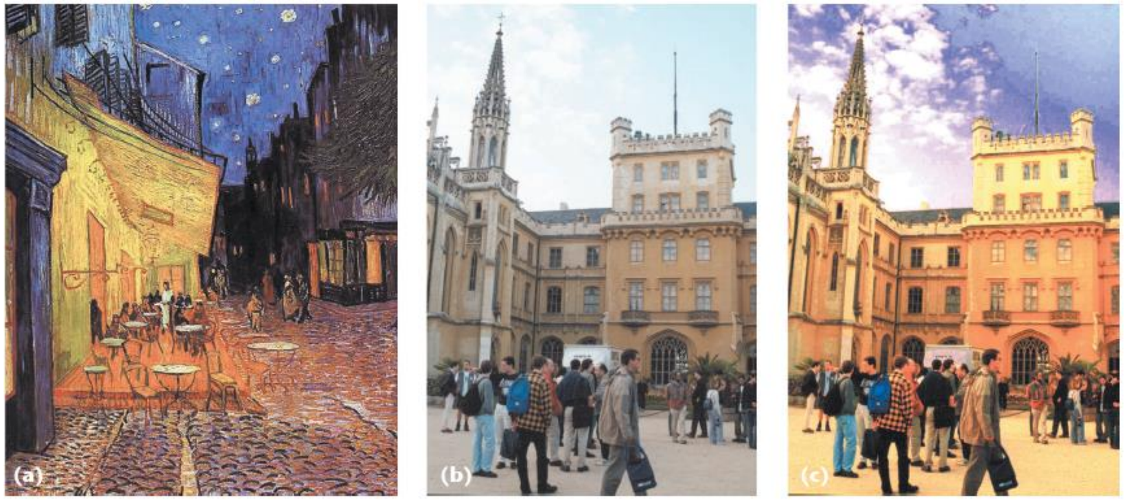
Images

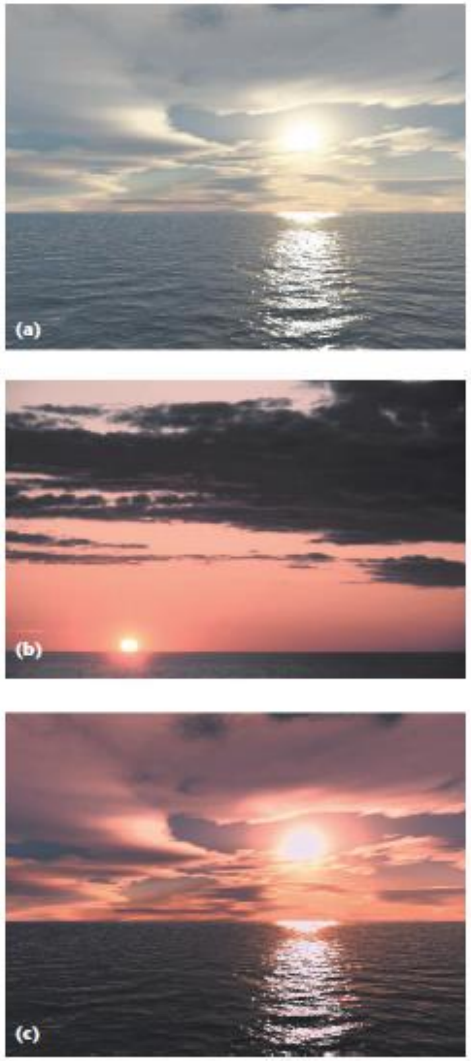
- Still pictures are represented as a bitmap (a grid of pixels).
- Input: digitally scanned pictures or directly acquired from a digital camera.
- Input: May also be generated by programs.
- Stored at 1 bit per pixel (Black and White), 8 Bits per pixel (Grey Scale, Colour Map) or 24 Bits per pixel (True Colour)
- Size: a 512x512 grayscale image takes up 1/4 MB, a 512x512 24-bit color image takes 3/4 MB with no compression.
- 10+ Megapixels digital camera takes 29MB uncompressed!
- Compression is commonly applied.

Example-Image

- Color transformation.
  - Change color information from images to images.



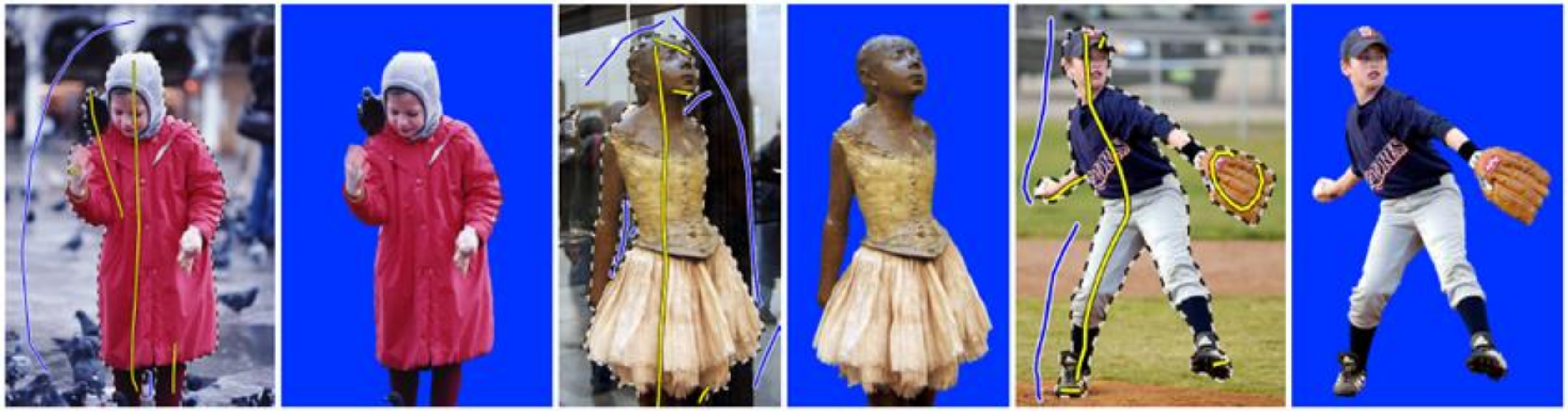


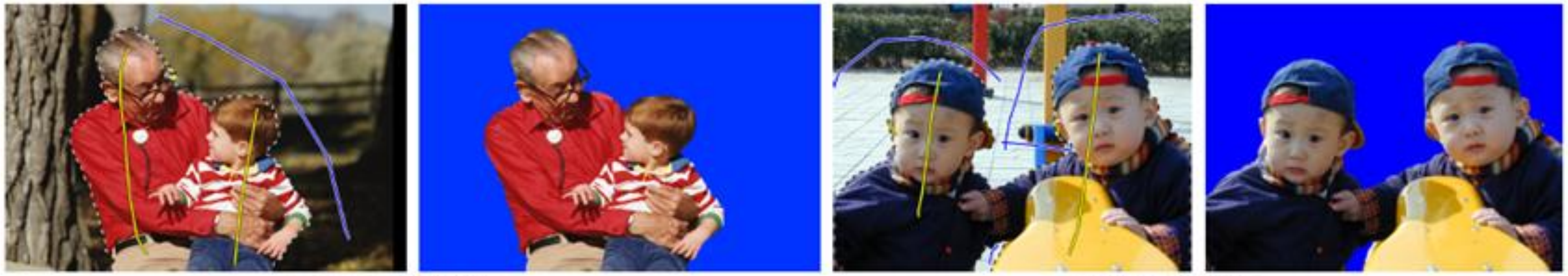


Images from papers "**Color Transfer between Images**".

Example-Image

- Foreground & background segmentation


(a) Girl (4/2/12) (b) Ballet (4/7/14) (c) Boy (6/2/13)


(c) Grandpa (4/2/11) (d) Twins (4/4/12)

Images from papers “**Lazy Snapping**”.

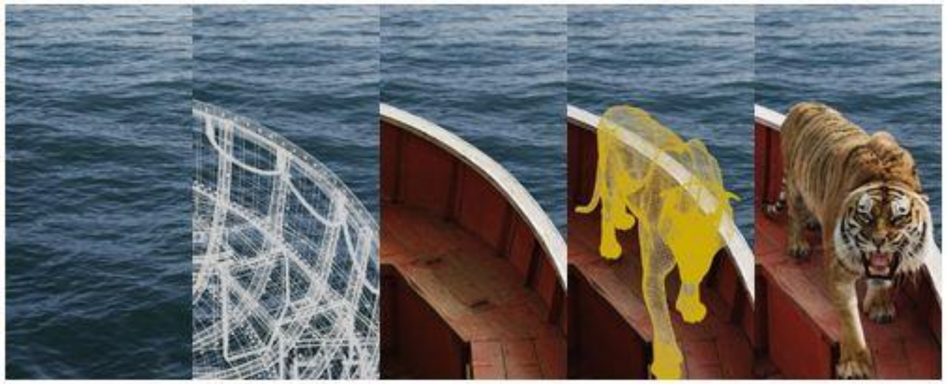
Graphics

- Format: constructed by the composition of primitive objects such as lines, polygons, circles, curves and arcs.
- Input: Graphics are usually generated by a graphics editor program (e.g. Illustrator) or automatically by a program (e.g. Postscript).
- Graphics are usually editable or revisable (unlike Images).
- Graphics input devices: keyboard (for text and cursor control), mouse, trackball or graphics tablet.
- Graphics usually store the primitive assembly

Example-Graphics

Images from NVIDIA

- Special effects in movies

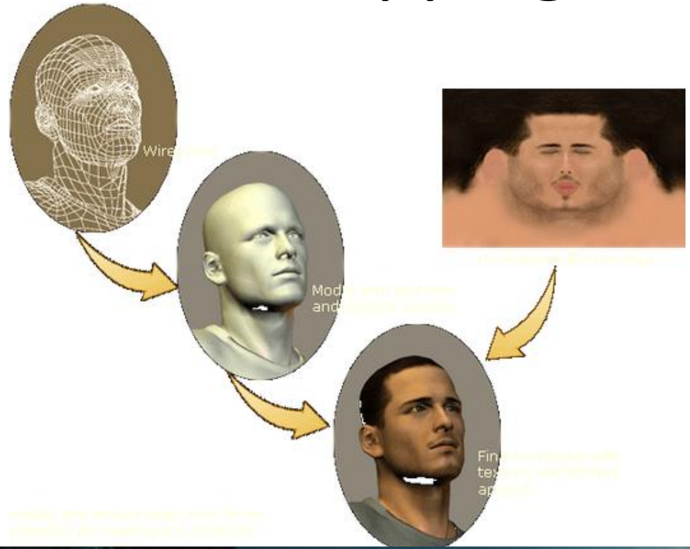

孟加拉虎 Richard Parker 的影像合成

為創造電影中的貓鼬島所採用的合成技術

利用 Rampage 繪製《少年 Pi 的奇幻漂流》中的天景

Example-Graphics

- Texture mapping and lighting



Some models in games.

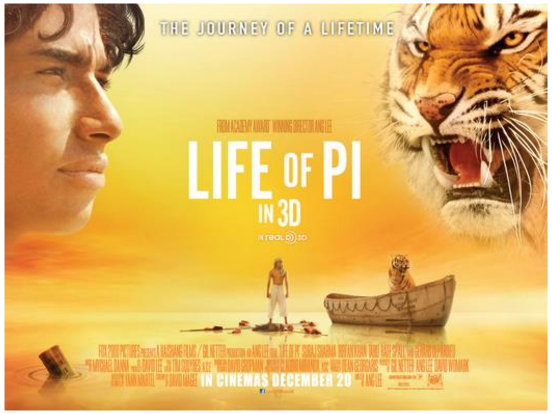
Images from google.

Video

- Input: Analog video is usually captured by a video camera and then digitized.
- Raw video can be regarded as being a series of single images.
- There are typically 25, 30 or 50 frames per second.
- E.g. A 512x512 size monochrome video images take 25x0:25 = 6.25MB for a second to store uncompressed.
- Typical PAL digital video (720x576 pixels per frame) takes about 1.24x25 = 31MB for a second to store uncompressed.
- HD video on Blu-ray (up to 1920 x1080 = 2 Megapixels per frame) $\sim$ 6.2 x 25 = 155MB/second to store uncompressed.
- Digital video clearly needs to be compressed for most times.

Example-Video

- Stereo videos and movies



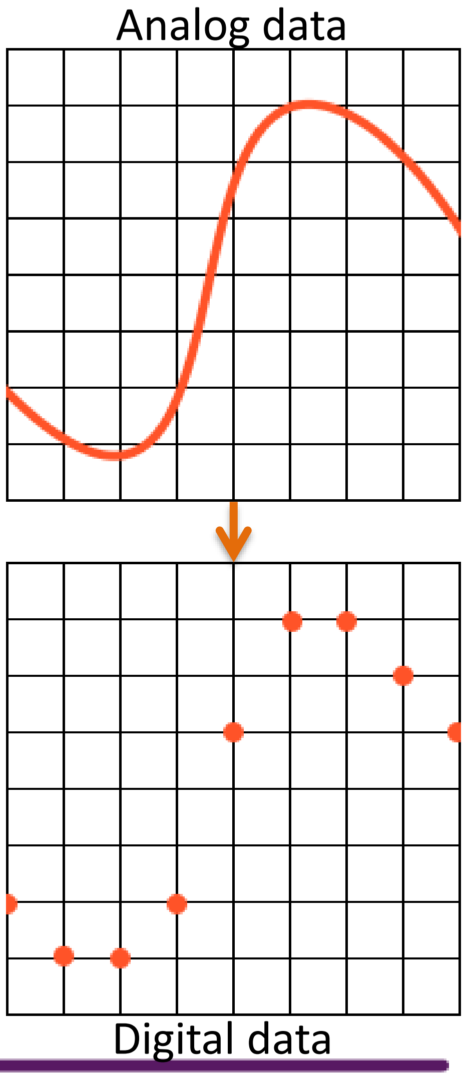
Images from google.

Example-Video

- ADAS

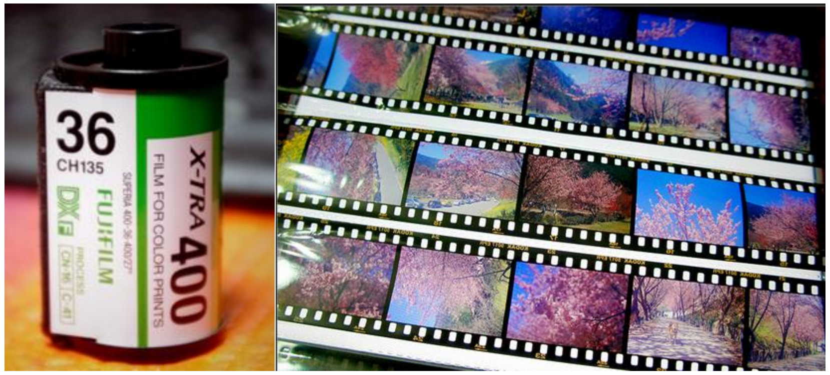
Analog and digital data

- Analog data
  - Takes on continuous values
  - E.g. voice, video
- Digital data
  - Takes on discrete values
  - E.g. text, integer
- Converting the continuous phenomena of images, sound, and motion into a discrete representation that can be handled by a computer is called ***analog-to-digital conversion***.



Analog data compared with digital data

- Analog data > digital data
  - More information -> more precise and better quality
- Digital data > Analog data
  - Size -> communicated more compactly
  - Reliably -> less affected by noise when transmitted



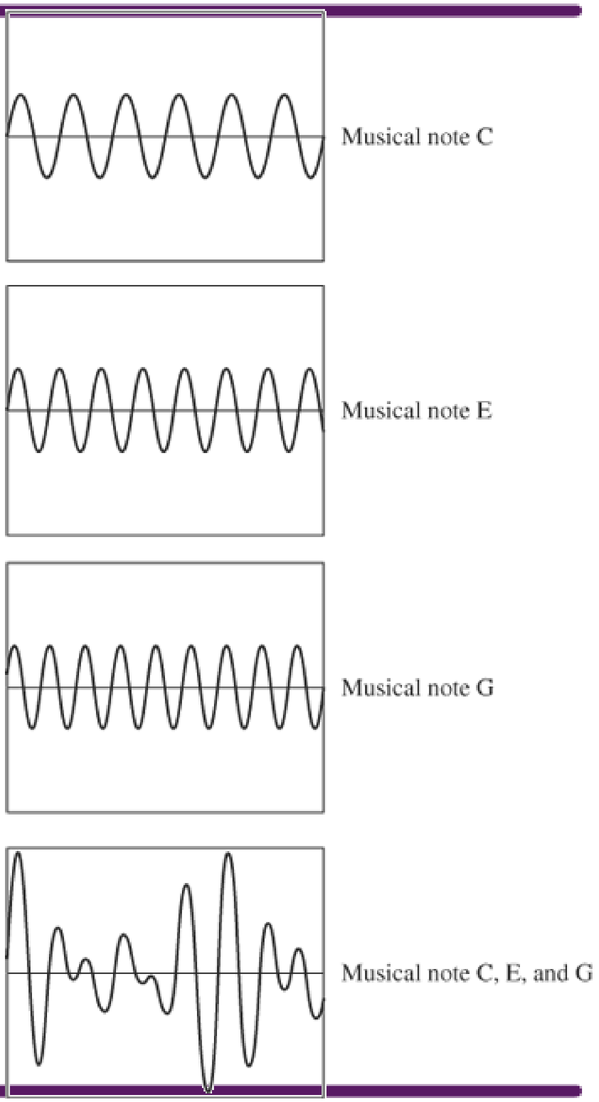
Analog **<->** ***Digital***

- **Computers** can **only** work with **digital data**.

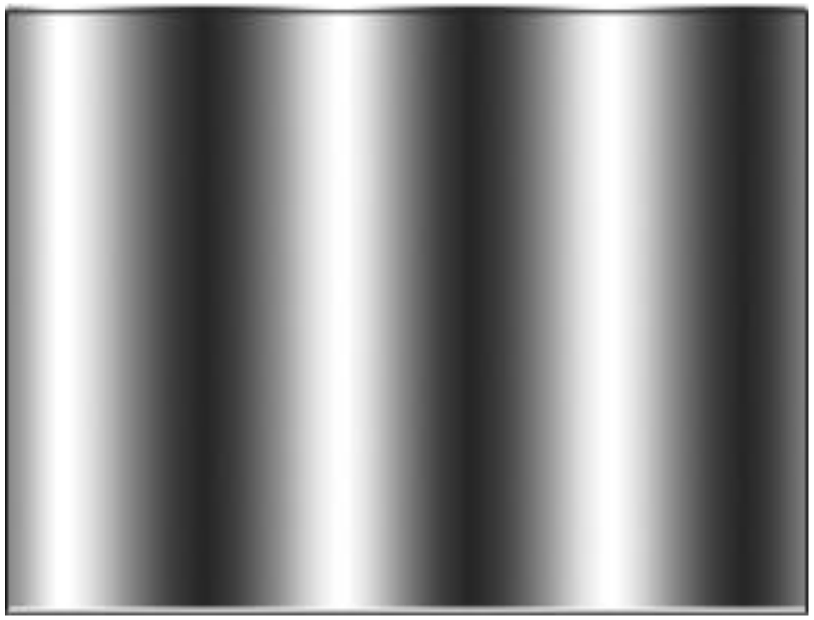
Images from google.

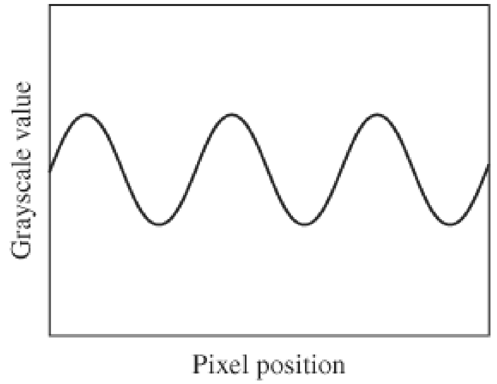
Fourier transform

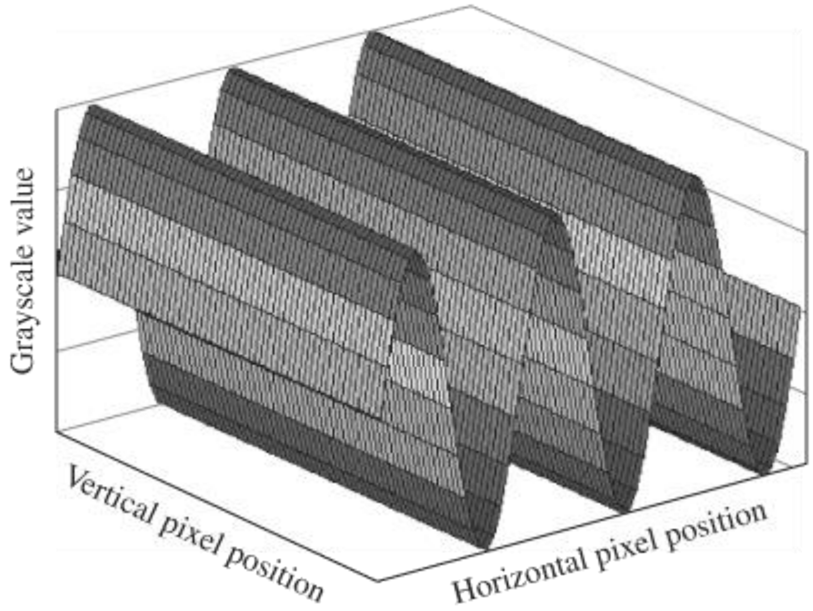
- Fourier analysis shows that any periodic signal can be decomposed into an infinite sum of sinusoidal waveforms.
- Fourier transform makes it possible to store a complex sound wave in digital form, determine the wave's frequency components, and filter out components that are not wanted.



An example of grayscale image





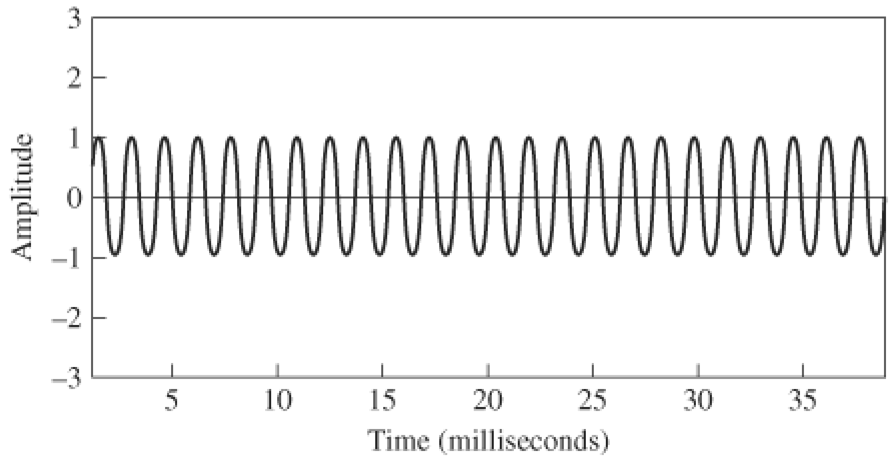


A/D Conversion

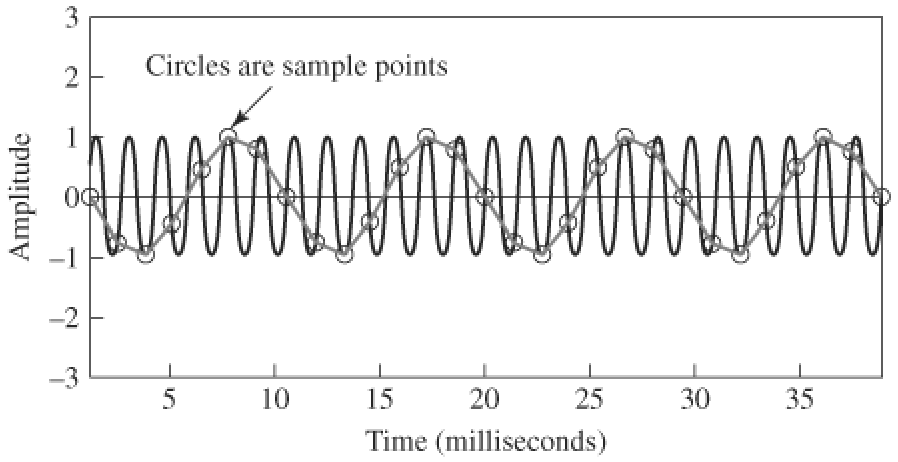
- Analog-to-digital conversion requires two steps: ***sampling*** and ***quantization***.
- The first step, sampling, chooses discrete points at which to measure a continuous phenomenon (***signal***).
- For images, the sample points are evenly separated in space. For sound, the sample points are evenly separated in time.
- The number of samples taken per unit time or unit space is called the ***sampling rate*** or, alternatively, the ***resolution***.
- The second step, quantization, requires that each sample be represented in a fixed number of bits, called the ***bit depth***. The bit depth limits the precision with which each sample can be represented.

Sampling and Aliasing

- ***Undersampling:*** sampling rate does not keep up with the rate of change in the signal.
- ***Aliasing*** in a digital signal arises from undersampling and results in the sampled discrete signal cannot reconstruct the original source signal.

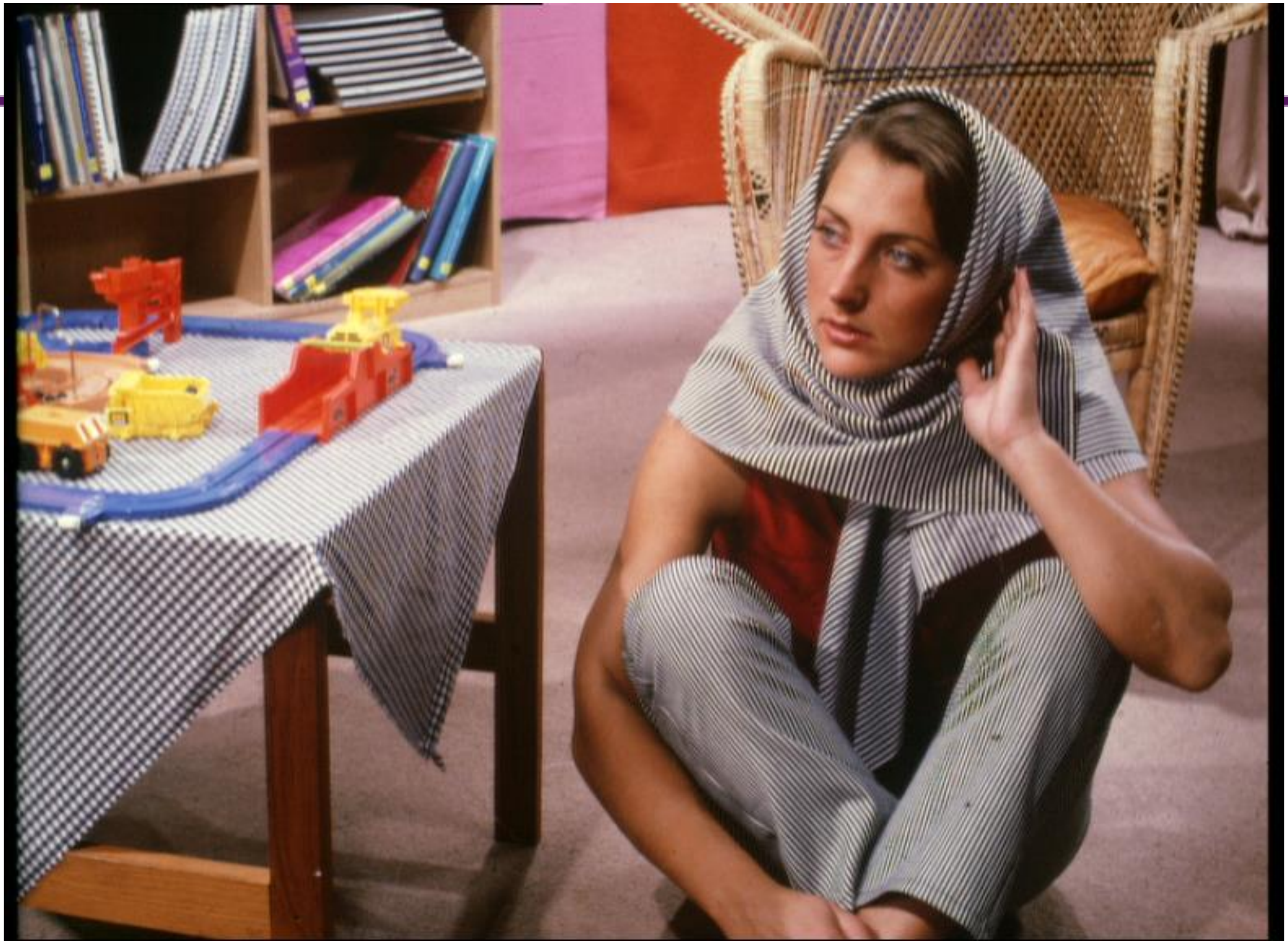


Audio wave at 637 Hz

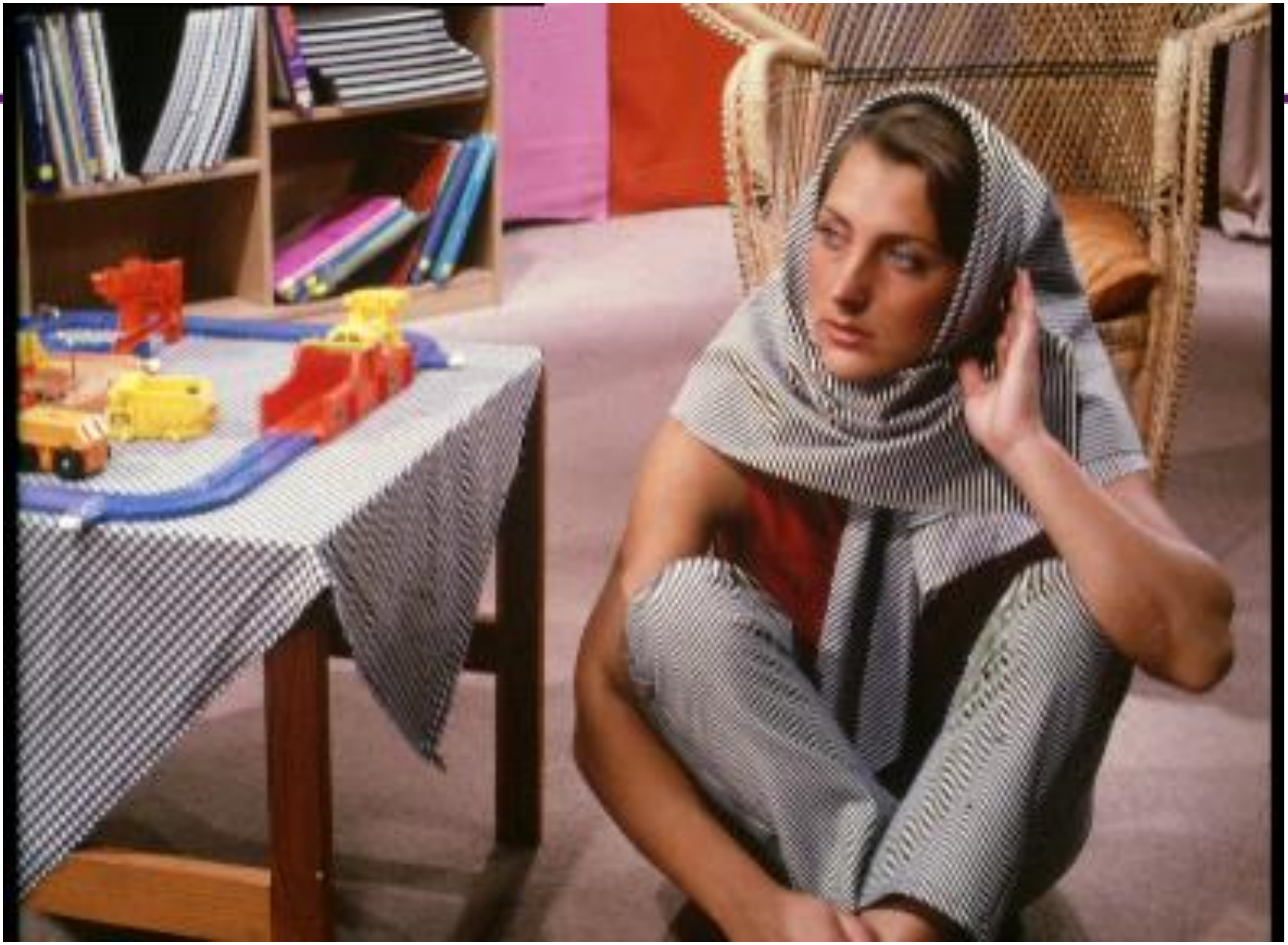


637 Hz audio wave sampled at 770 Hz

The image of Barbara



Aliasing due to sampling



Nyquist theorem

- The Nyquist theorem specifies the sampling rate needed for a given spatial or temporal frequency.
- To guarantee that no aliasing will occur, you must use a sampling rate that is greater than twice the maximal frequency in the signal being sampled.
- Let f be the frequency of a sine wave. Let r be the minimum sampling rate that can be used in the digitization process such that the resulting digitized wave is not aliased. Then the ***Nyquist frequency*** r is

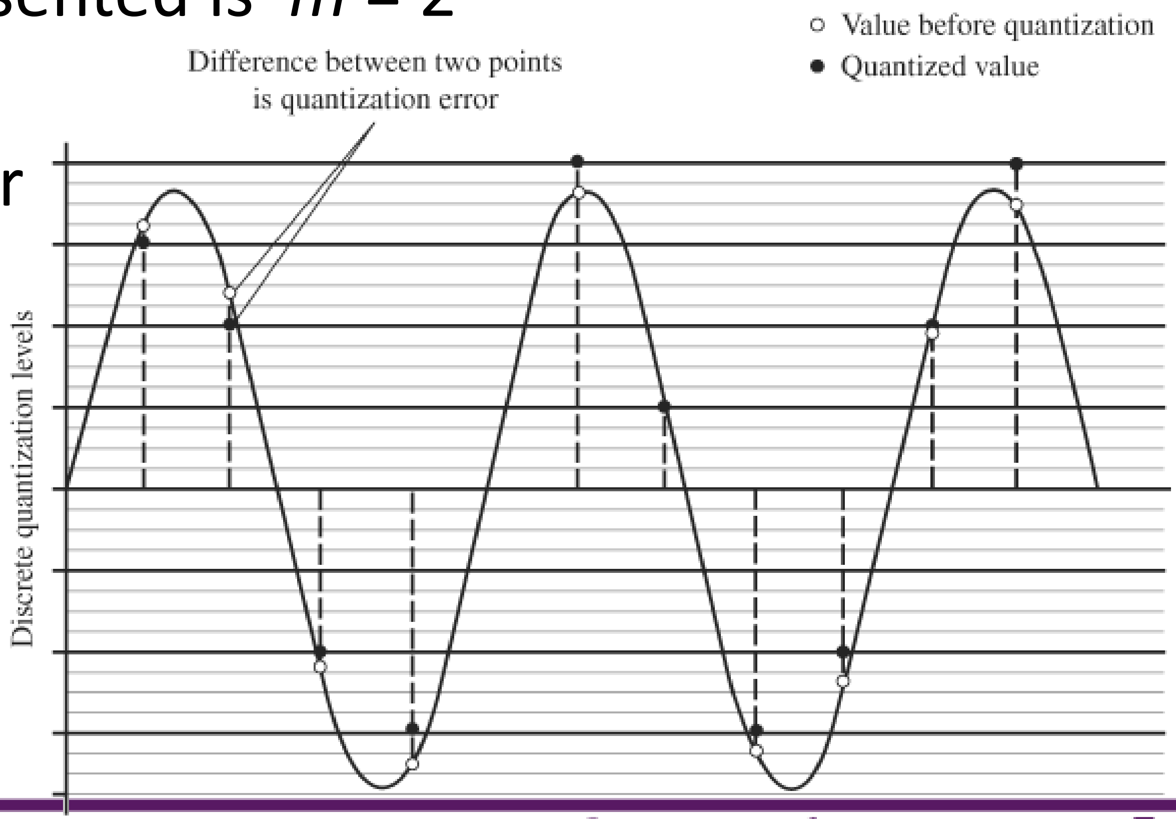
$$r = 2f$$

https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem

Quantization

- Let *n* be the number of bits used to quantize a digital sample. Then the maximum number of different values that can be represented is $m = 2^n$

- Quantization error


○ Value before quantization
● Quantized value
Difference between two points is quantization error
Discrete quantization levels

SNR and SQNR

- Signal-to-noise ratio (SNR) can generally be defined as the ratio of the meaningful content of a signal versus the associated noise.
- In analog data communication, SNR is defined as the ratio of the average power in the signal versus the power in the noise level.
- For a digitized signal, the signal-to-noise ratio is defined as the ratio of the maximum sample value versus the maximum quantization error. This can also be called ***signal-to-quantization-noise ratio*** (***SQNR***)

SQNR and Dynamic Range

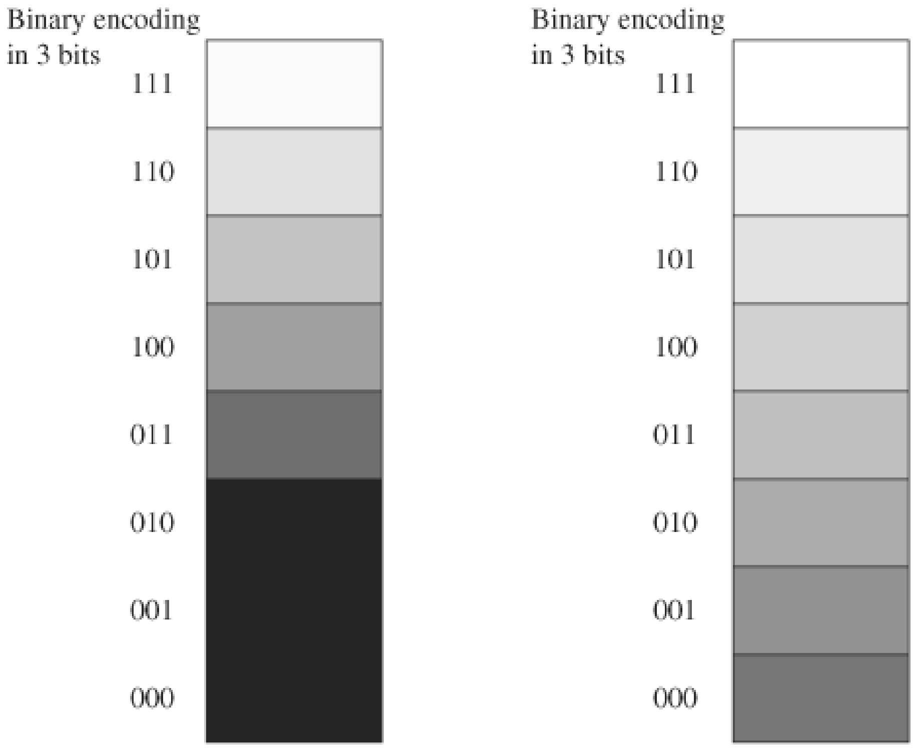
- Let *n* be the bit depth of a digitized media, Then the ***SQNR*** is

$$SQNR = 20 \log_{10}(2^n) \quad \text{(in dB)}$$

- Signal-to-quantization-noise ratio is directly related to ***dynamic range***.
- Dynamic range, informally defined, is the ratio of the largest-amplitude sound (or color, for digital images) and the smallest that can be represented with a given bit depth.

Dynamic Range

- With three bits, you have eight colors. You can spread these colors out over a wide range or a narrow range.
- In either case, the dynamic range is the same, dictated by the bit depth, which determines the maximum error possible (resulting from rounding to available colors) relative to the range of colors represented.



Compression

- Digital media files are usually very large, and they need to be made smaller—compressed
- Without compression
  - Won’t have storage capacity
  - Won’t be able to communicate them across networks without overly taxing the patience of the recipients
- Compression Rate
  - A file that is reduced by compression to half its original size **50% compression** or **compression rate is 2:1**
- Compression algorithms can be divided into two basic types: ***lossless compression*** and ***lossy compression***.

Types of compression

- Lossless compression
  - No information is lost between the compression and decompression steps
  - Reduces the file size to fewer bits
  - Method:
    - ✓Run-Length encoding (RLE)
    - ✓Entropy encoding
    - ✓Arithmetic encoding

Types of compression

- Lossy compression
  - Sacrifice some information which is not important to human perception
    - ✓ In image files: subtle changes in color that the eye cannot detect
    - ✓ In sound files: changes in frequencies that are imperceptible to the human ear
  - Method:
    - ✓ Transform encoding
    - ✓ Quantization

Types of compression

- **Dictionary-based** methods (e.g. LZW compression) use a look-up table of fixed-length codes.
- **Entropy compression** uses a statistical analysis of the frequency of symbols and achieves compression by encoding more frequently-occurring symbols with shorter code words.
- **Arithmetic encoding** is also based on statistical analysis, but encodes an entire file in a single code word rather than creating a separate code for each symbol.
- **Differential encoding** is a form of lossless compression that reduces file size by recording the difference between neighboring values.

Run-Length encoding (RLE)

- RLE is a simple example of lossless compression, being used in image compression.
- For example, BMP file optionally use RLE
- Instead of storing value of each pixel, to store number pairs (c,n).
  - c: grayscale value
  - n: how many consecutive pixels have that value

Run-Length encoding (RLE)

- Example 1
  - Data sequence:

    255 255 255 255 255 255 242 242 242 242 238 238 238 238 238 238 255 255 255 255

  - Encoding result:

    (255, 6), (242, 4), (238, 6), (255, 4)

- Example 2

  If one byte is used to represent each *n*, then the largest value for *n* is 255. If 1000 consecutive whites exist in the file.

  - Encoding result:

    (255, 255), (255, 255), (255, 255), (255, 235)

Choose the size of *n* is important

Run-Length encoding (RLE)

- Example 3
  - Data sequence:

    ```
    255 254 253 252 251 250
    ```

  - Encoding result:

    ```
    (255, 1), (244, 1), (253, 1), (252, 1), (251,1), (250,1)
    ```

  - RLE is not suitable for those sequences which have few repetitions.
- *Any lossless compression algorithm and for any length file, there will always exist at least one case where the algorithm does not reduce the size of an input file of that length.*

Entropy encoding

- Using fewer bits to encode symbols that occur more frequently, while using more bits for symbols that occur infrequently
- Shannon's equation, below, gives us a way judging whether our choice of number of bits for different symbols is close to optimal.
  - $H(S) = \eta = \sum_i p_i log_2(\frac{1}{p_i})$
  - S be a string of symbols and p_i be the frequency of the i^{th} symbol in the string.

Entropy encoding

- Applying ***Shannon's entropy equation***, you can determine an optimum value for the average number of bits needed to represent each symbol-instance in a string of symbols, based on how frequently each symbol appears
- Shannon proves that you can't do better than this optimum

Example of entropy encoding

- Example 1

  An image file that has 256 pixels, each pixel of a different color. Then the frequency of each color is 1/256. Thus, Shannon's equation reduces to

$$\sum_{0}^{255}\frac{1}{256}(log_2(\frac{1}{\frac{1}{256}})) = \sum_{0}^{255}\frac{1}{256}\left(log_2(256)\right) = \sum_{0}^{255}\frac{1}{256}(8) = 8$$

  This means that the average number of bits needed to encode each color is eight, which makes sense in light of the fact that $\log_2 256 = 8$

Entropy encoding

- Example 2

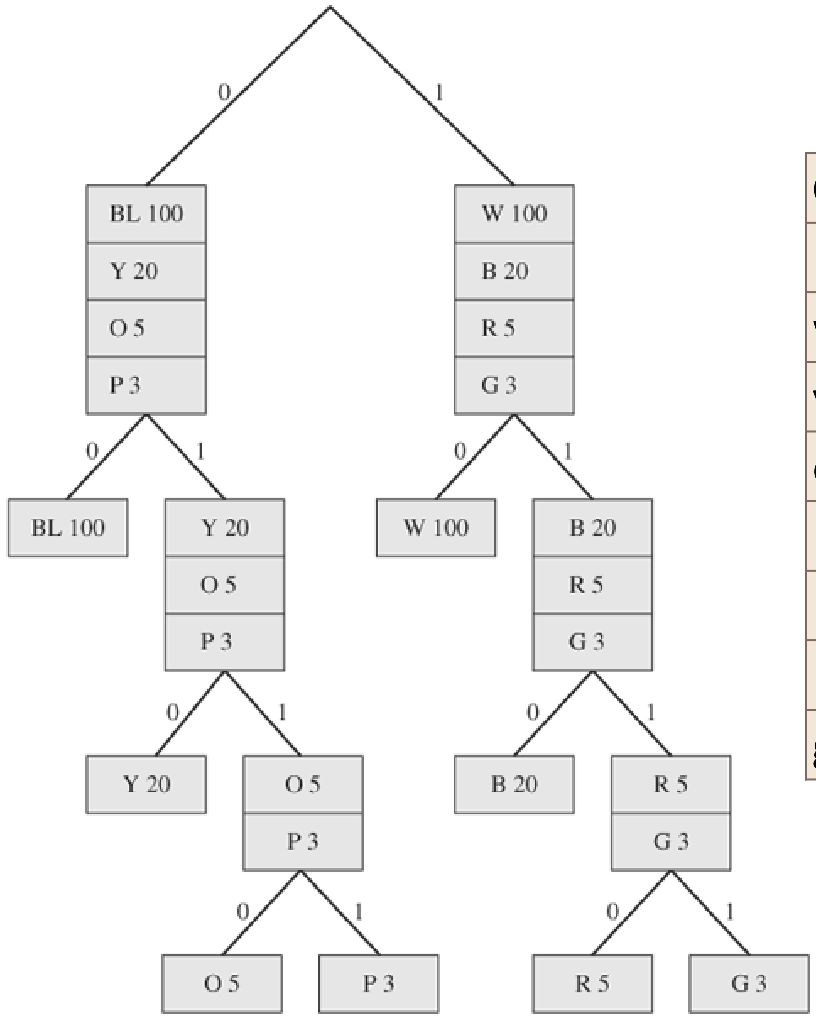
  An image file of 256 pixels and only eight colors

$$H(S) = \eta = \sum_i \underbrace{p_i}\,\underbrace{\log_2\left(\frac{1}{p_i}\right)}$$

Color	Frequency	Optimum Number of Bits to Encode This Color	Relative Frequency of the Color in the File	Product of Columns 3 and 4
black	100	1.356	0.391	0.530
white	100	1.356	0.391	0.530
yellow	20	3.678	0.078	0.287
orange	5	5.678	0.020	0.111
red	5	5.678	0.020	0.111
purple	3	6.415	0.012	0.075
blue	20	3.678	0.078	0.287
green	3	6.415	0.012	0.075

2.006

Example of Entropy Coding



Color	Frequency	Code
black	100	00
white	100	10
yellow	20	010
orange	5	0110
red	5	1110
purple	3	0111
blue	20	110
green	3	1111

Shannon-Fano algorithm

1. For a given list of symbols, develop a corresponding list of probabilities or frequency counts so that each symbol's relative frequency of occurrence is known.
2. Sort the lists of symbols according to frequency, with the most frequently occurring symbols at the left and the least common at the right.
3. Divide the list into two parts, with the total frequency counts of the left part being as close to the total of the right as possible.
4. The left part of the list is assigned to 0, and the right part is assigned to 1.
5. Recursively apply the steps 3 and 4 to each of the two halves, subdividing groups and adding bits to the codes until each symbol has become a corresponding code leaf on the tree.

What are the problems?

- Each symbol must be treated individually.
- Each symbol has its own code.
- Code must be represented in an integral number of bits.

Arithmetic Encoding

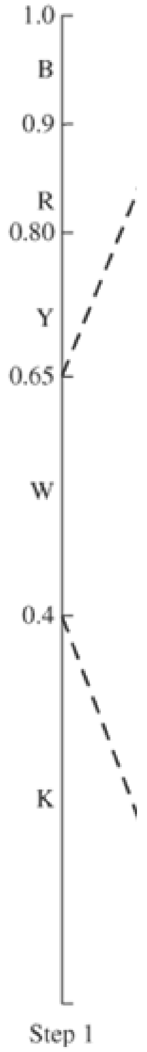
- Also beginning with a list of the symbols in the input file and their frequency of occurrence
- Example

  A file contains 100 pixels in five colors

Color	Frequency Out of Total Number of Pixels in File	Probability Interval Assigned to Symbol
black (K)	40/100 = 0.4	0 – 0.4
white (W)	25/100 = 0.25	0.4 – 0.65
yellow (Y)	15/100 = 0.15	0.65 – 0.8
red (R)	10/100 = 0.1	0.8 – 0.9
blue (B)	10/100 = 0.1	0.9 – 1.0

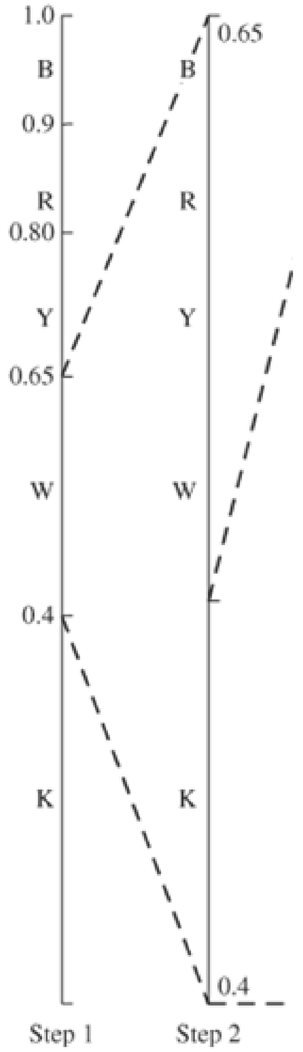
Arithmetic Encoding

- Step 1 – W
  interval : 0.4 – 0.65



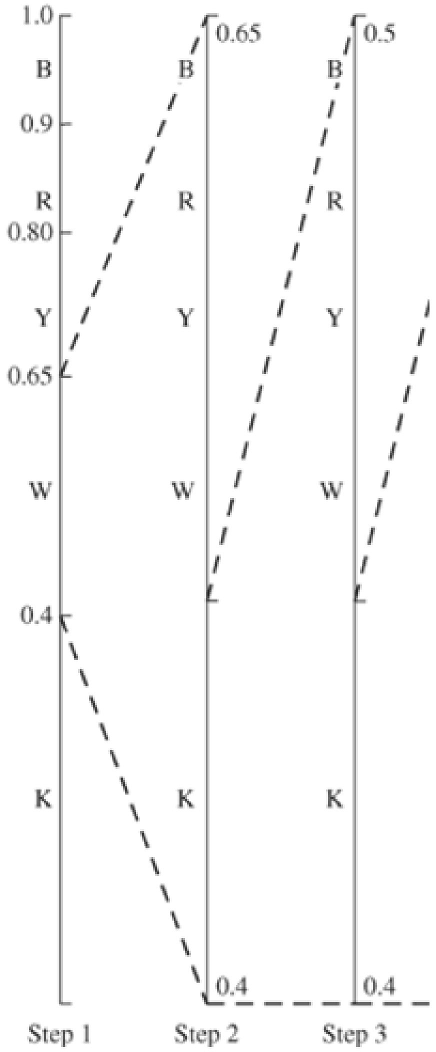
Arithmetic Encoding

- Step 1 – W
  interval : 0.4 – 0.65
- Step 2 – K
  0.4+0.25*0.4=0.5
  interval : 0.4 – 0.5



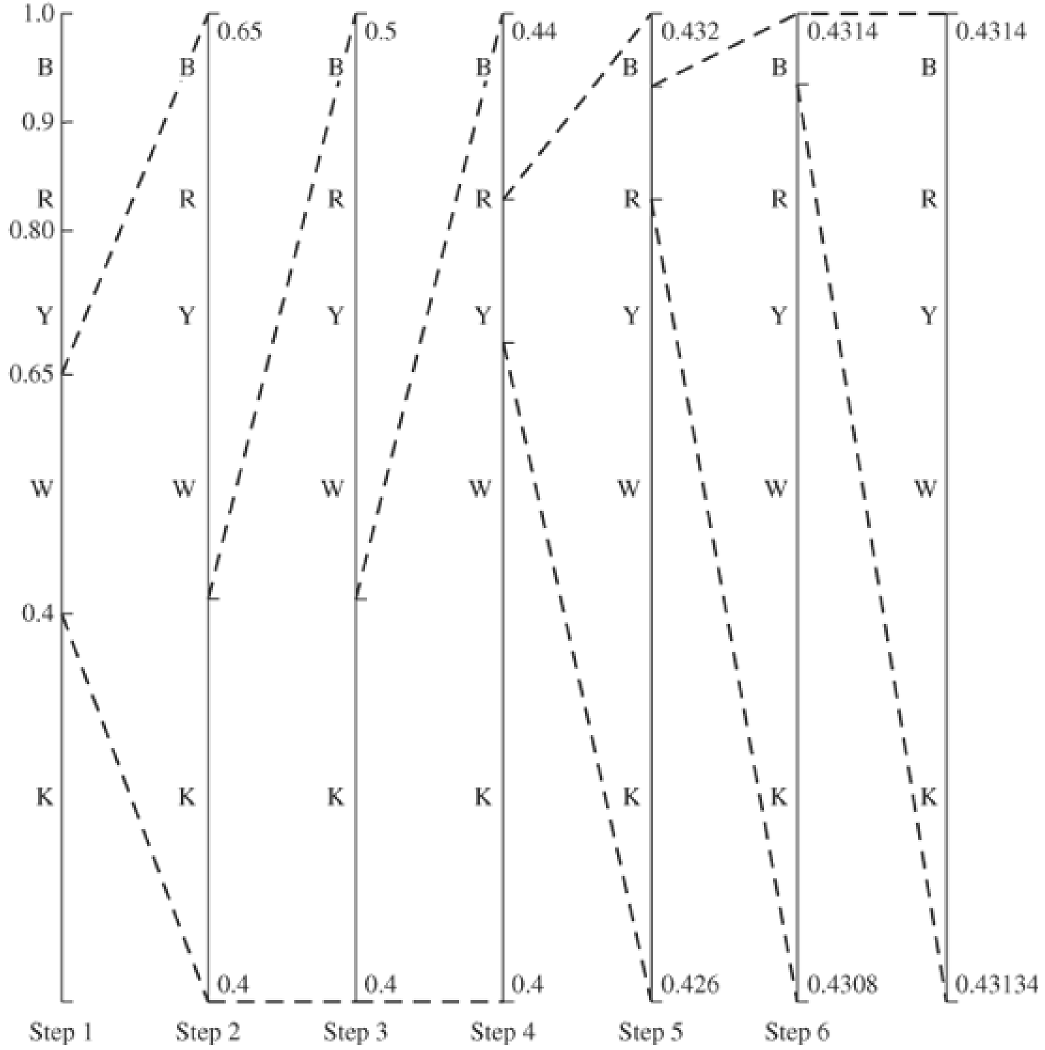
Arithmetic Encoding

- Step 1 – W
  interval : 0.4 – 0.65
- Step 2 – K
  0.4+0.25*0.4=0.5
  interval : 0.4 – 0.5
- Step 3 – K
  0.4+0.1*0.4=0.44
  interval: 0.4 – 0.44



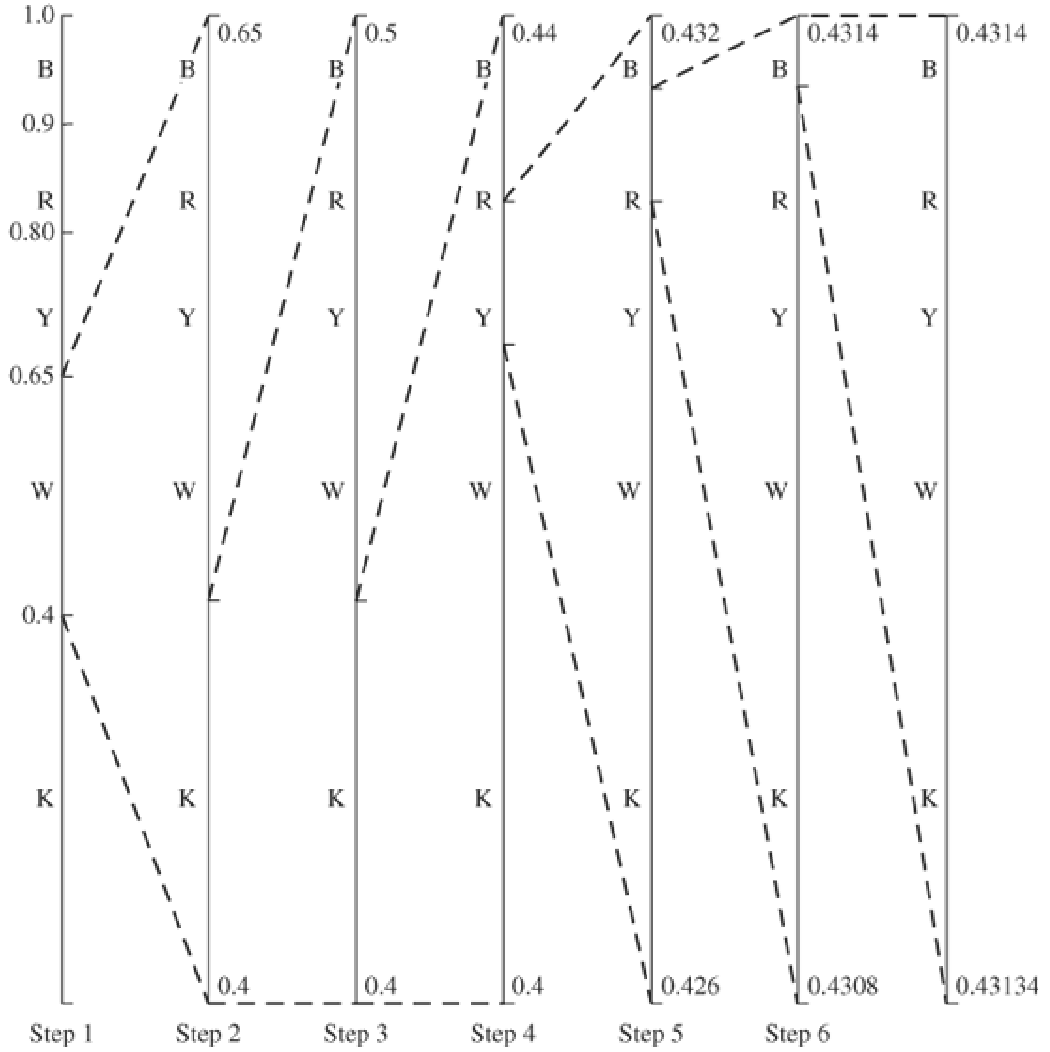
Arithmetic Encoding

- Step 1 – W
  interval : 0.4 – 0.65
- Step 2 – K
  0.4+0.25*0.4=0.5
  interval : 0.4 – 0.5
- Step 3 – K
  0.4+0.1*0.4=0.44
  interval: 0.4 – 0.44
- ……



Arithmetic Decoding

- Final encoding : 0.43137 (0.43134+0.4314)/2=0.43137
- 0.43137 fits in the interval assigned to W -> W
- Remove the scaling of W (0.43137-0.4)/0.25=0.12548 -> K
- Remove the scaling of K 0.12548/0.4=0.3137 -> K
- ……



Arithmetic Decoding

Floating Point Number f, Representing Code	Symbol Whose Probability Interval Surrounds f	Low Value for Symbol's Probability Interval	High Value for Symbol's Probability Interval	Size of Symbol's Probability Interval
0.43137	W	0.4	0.65	0.25
(0.43137 − 0.4)/(0.65 − 0.4) =0.12548	K	0	0.4	0.4
(0.12548 − 0)/(0.4 − 0) =0.3137	K	0	0.4	0.4
(0.3137 − 0)/(0.4 − 0) =0.78425	Y	0.65	0.8	0.15
(0.78425 − 0.65)/(0.8 − 0.65) =0.895	R	0.8	0.9	0.1
(0.895 − 0.8)/(0.9 − 0.8) =0.95	B	0.9	1.0	0.1

Transform Encoding

- Lossy methods are often based upon ***transform encoding***
- Most commonly used transforms in digital media
  - Discrete cosine transform (DCT)
  - Discrete Fourier transform (DFT)
- No information is lost in the DCT or DFT. When a transform is used, it becomes possible to discard redundant or irrelevant information in later steps, thus reducing the digital file size. This is the lossy part of the process.

1D DCT

- Any row of *M* pixels can be represented as a sum of the *M* weighted cosine functions evaluated at discrete points

A 1D image of eight pixels

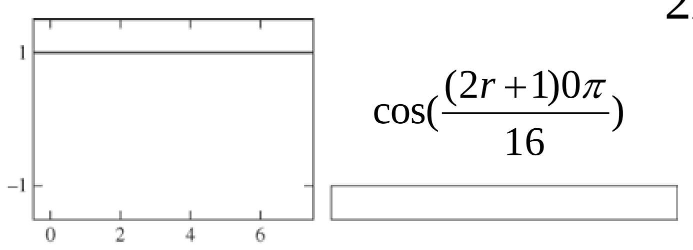
$$f(r) = \sum_{u=0}^{M-1} \frac{\sqrt{2}C(u)}{\sqrt{M}} F(u) \cos\left(\frac{(2r+1)u\pi}{2M}\right) \qquad for \quad 0 \le r \le M$$

$$where \quad C(u) = \frac{\sqrt{2}}{2} \qquad if \quad u = 0 \qquad otherwise \qquad C(u) = 1$$

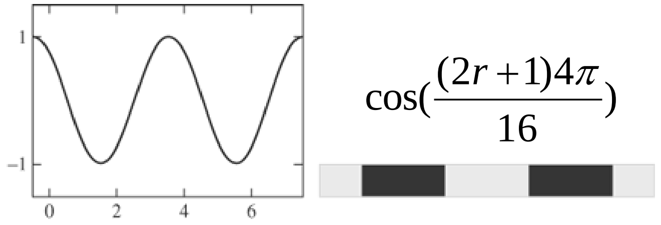
  - *f(r)* : a one-dimensional array
  - *F(u)*: one-dimensional array of coefficients

1D DCT

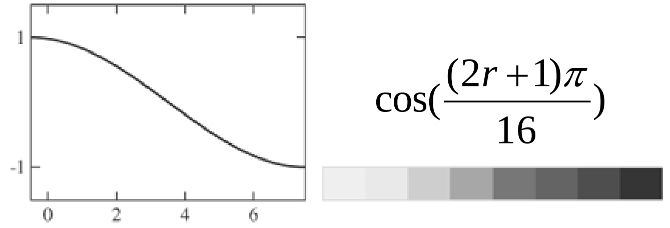
- Basis function: $\cos(\frac{(2r+1)u\pi}{2M})$, M=8



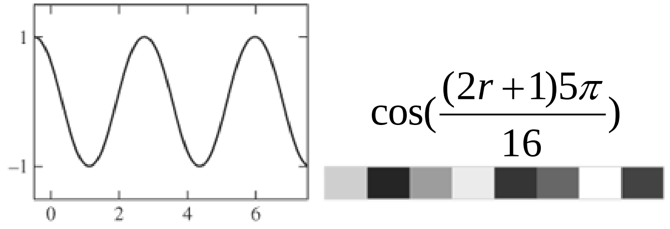
$\cos(\frac{(2r+1)0\pi}{16})$



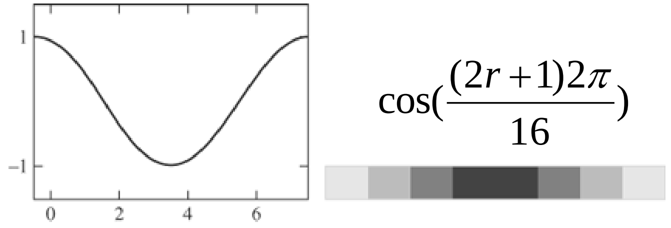
$\cos(\frac{(2r+1)4\pi}{16})$



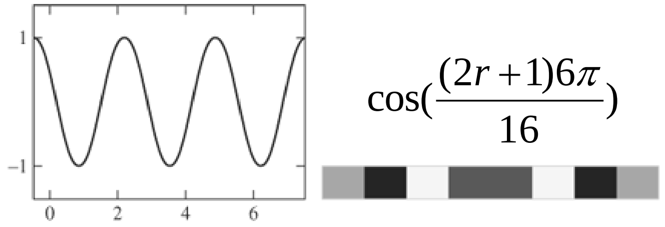
$\cos(\frac{(2r+1)\pi}{16})$



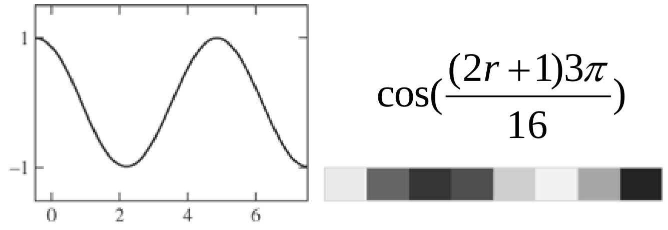
$\cos(\frac{(2r+1)5\pi}{16})$



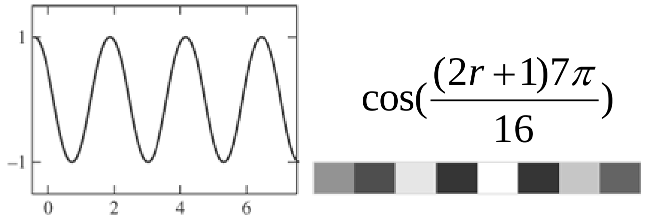
$\cos(\frac{(2r+1)2\pi}{16})$



$\cos(\frac{(2r+1)6\pi}{16})$



$\cos(\frac{(2r+1)3\pi}{16})$

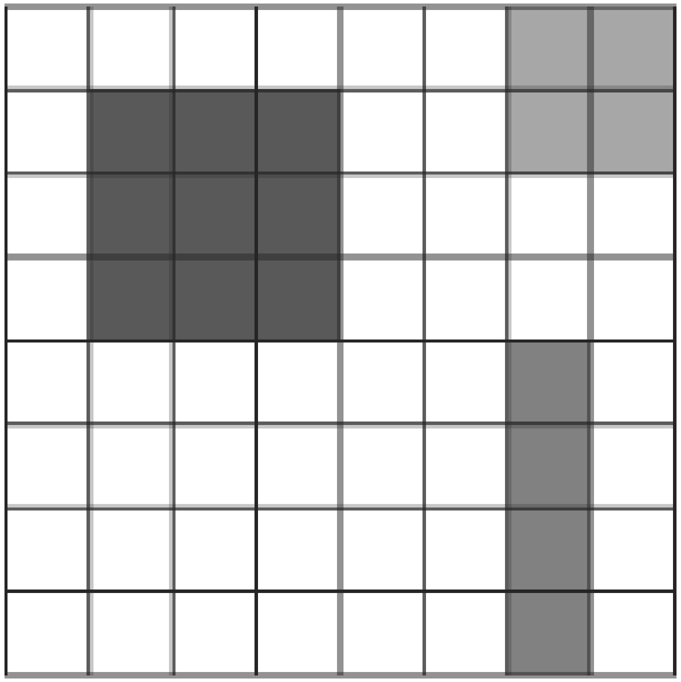


$\cos(\frac{(2r+1)7\pi}{16})$

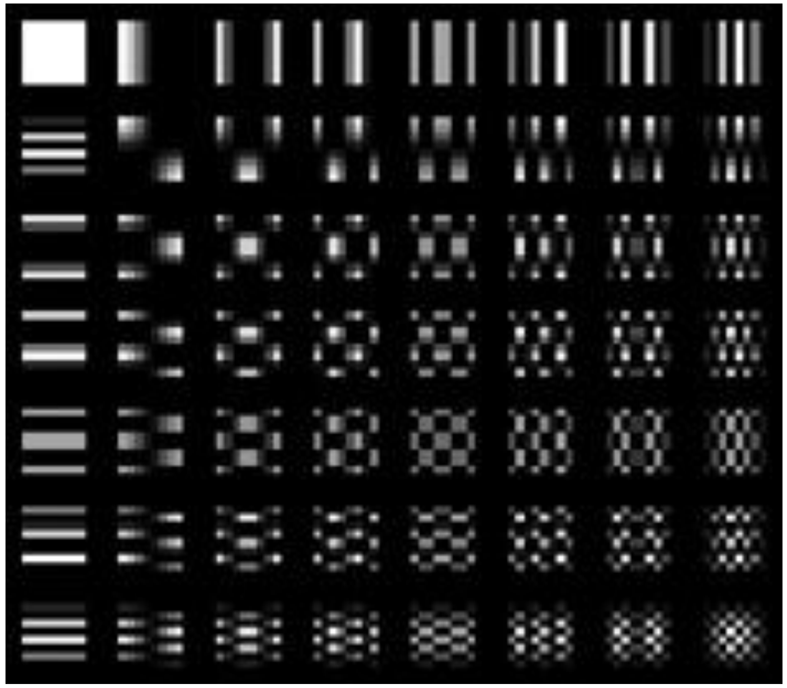
Transform Encoding

- 2D DCT

Origin image (8x8)



2D DCT Bases



Discrete Fourier Transform

- The ***discrete Fourier transform*** (**DFT**) operates on an array of *N* audio samples, returning cosine and sine coefficients that represent the audio data in the frequency domain.

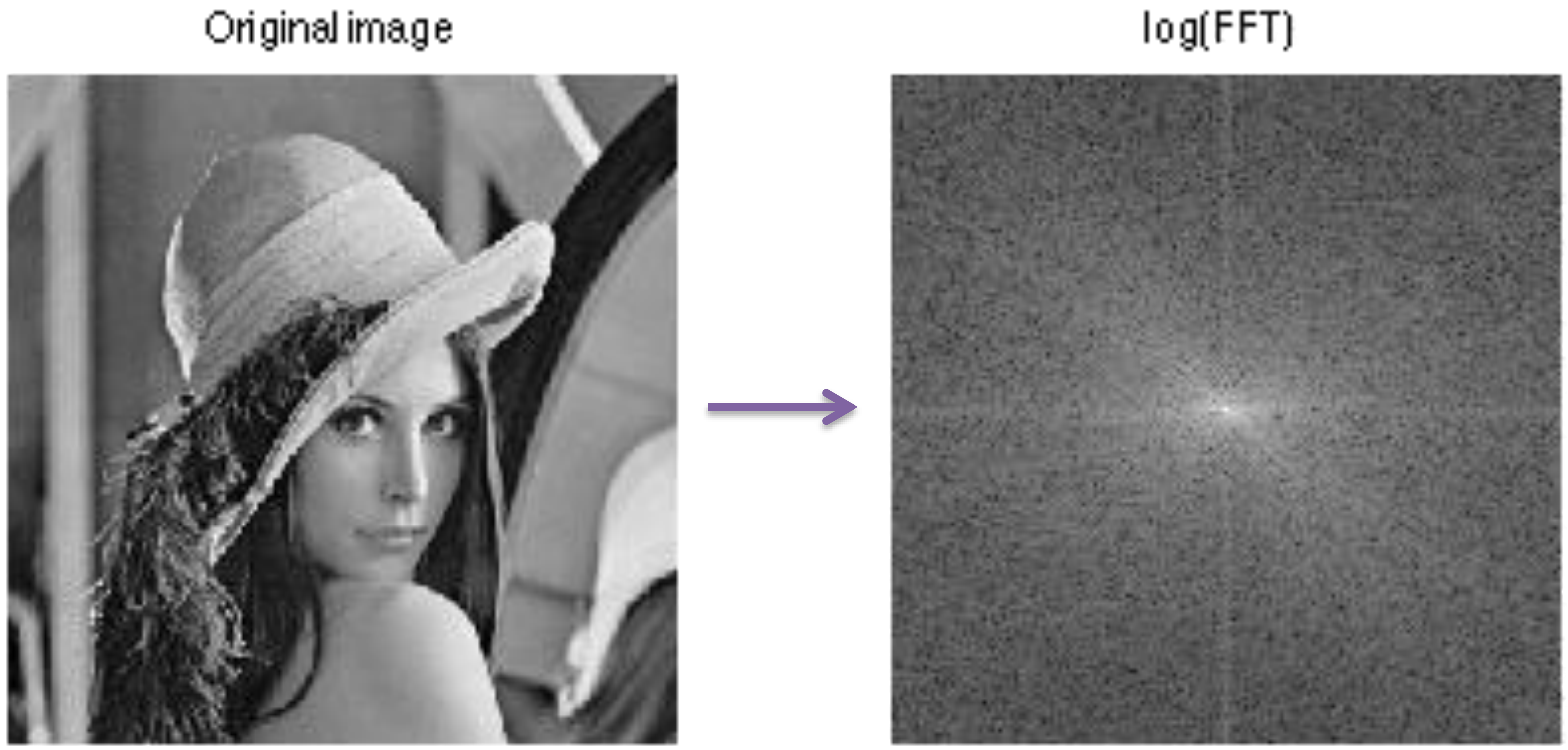
$$F_n = \frac{1}{N}\sum_{k=0}^{N-1} f_k \cos\left(\frac{2\pi nk}{N}\right) - i f_k \sin\left(\frac{2\pi nk}{N}\right) \qquad (4.8)$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} f_k e^{\frac{-i2\pi nk}{N}}$$

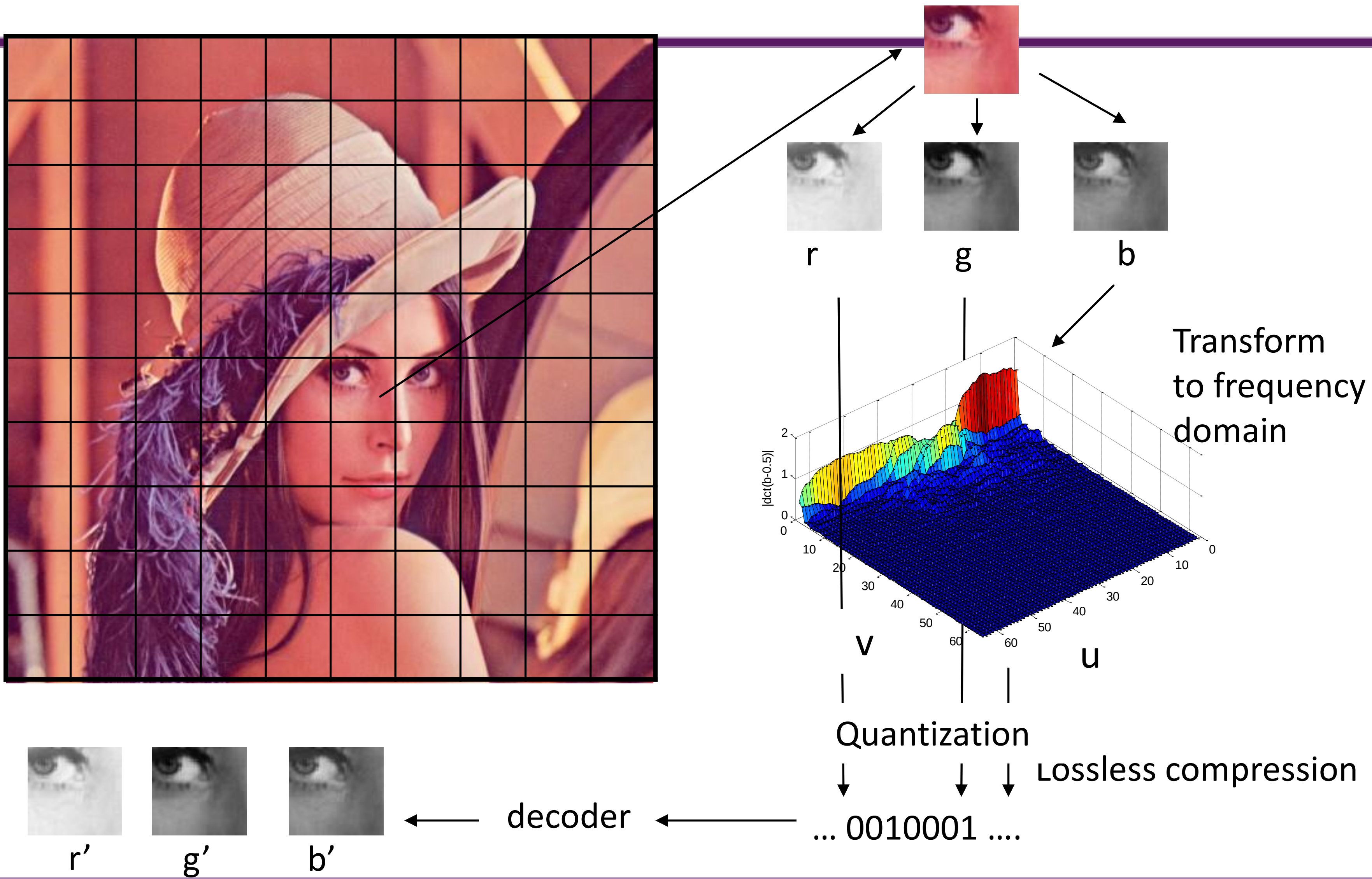
$$F(u) = \sum_{r=0}^{M-1} \frac{\sqrt{2}C(u)}{\sqrt{M}} f(r)\cos\left(\frac{(2r+1)u\pi}{2M}\right) \qquad \text{DCT}$$

Transform Encoding

- Discrete Fourier transform



“JPEG” Illustration



Summary

- Introduction to multimedia systems
- Analog to digital conversion
- Sampling and aliasing
- Nyquist Theorem
- Compression methods
  - Entropy coding
  - Arithmetic coding
  - Transform coding