COMP 102: Computers and Computing

Lecture 16: Multi-media data representation

Instructor: Kaleem Siddiqi (siddiqi@cim.mcgill.ca)

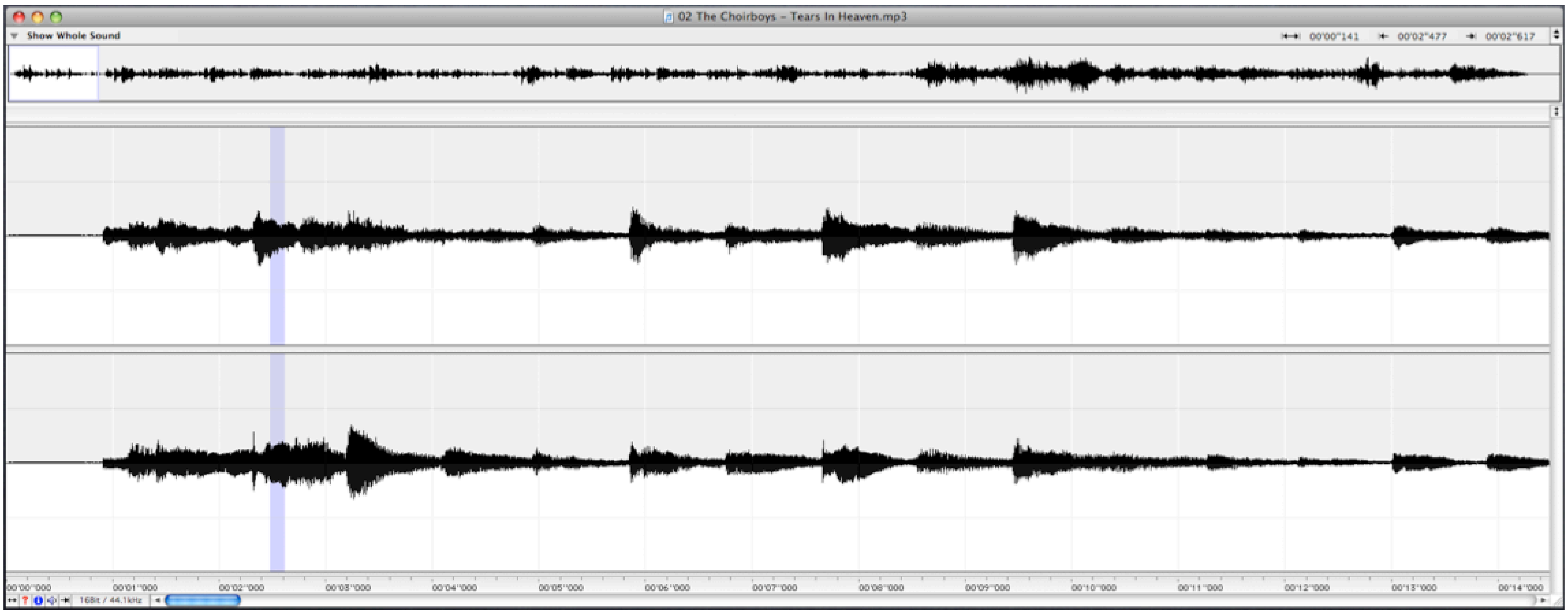
Class web page: www.cim.mcgill.ca/~siddiqi/102.html

Quick recap of data compression

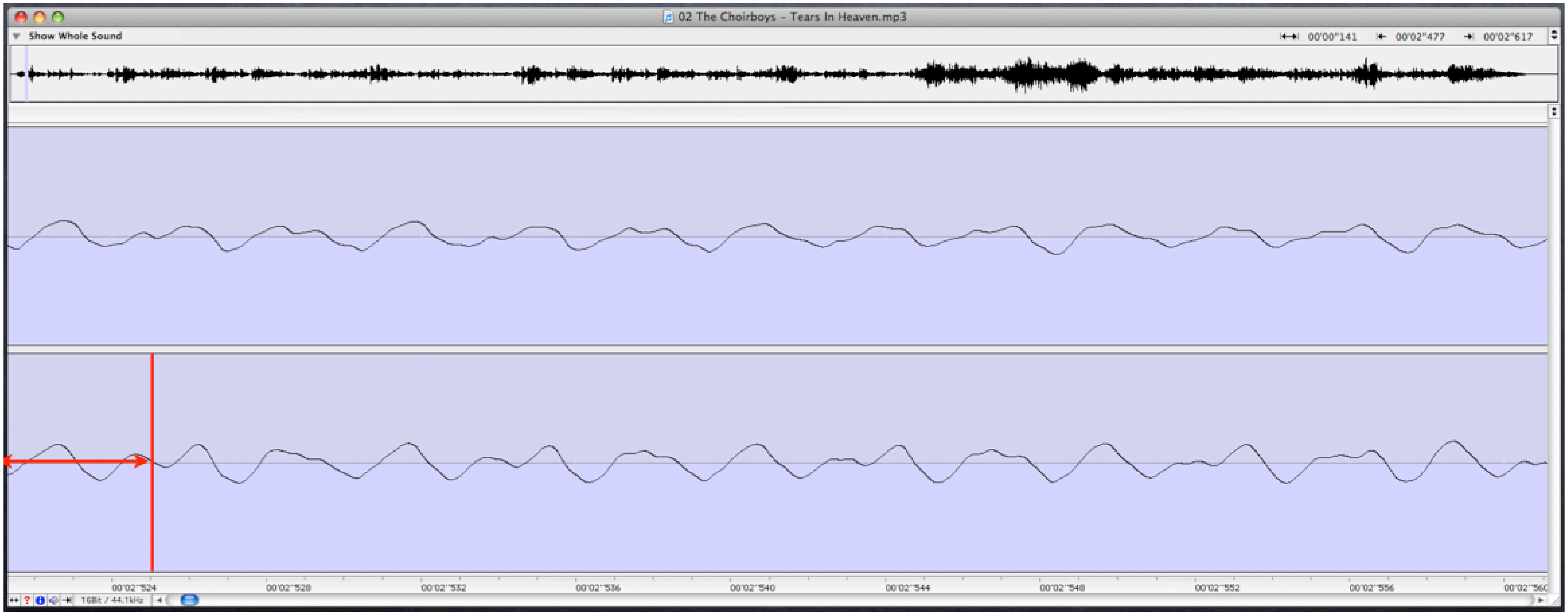
- Previous class: Basic ideas in data compression
  - Alphabet = set of symbols to encode.
  - Code = mapping from alphabet to binary symbols.
  - Prefix code = code where no codeword is a prefix of another.
  - Huffman code = most used symbols get shorter code.
  - Lossless vs Lossy compression.

- Today: Data representation for sound, images and video.

Sound formats



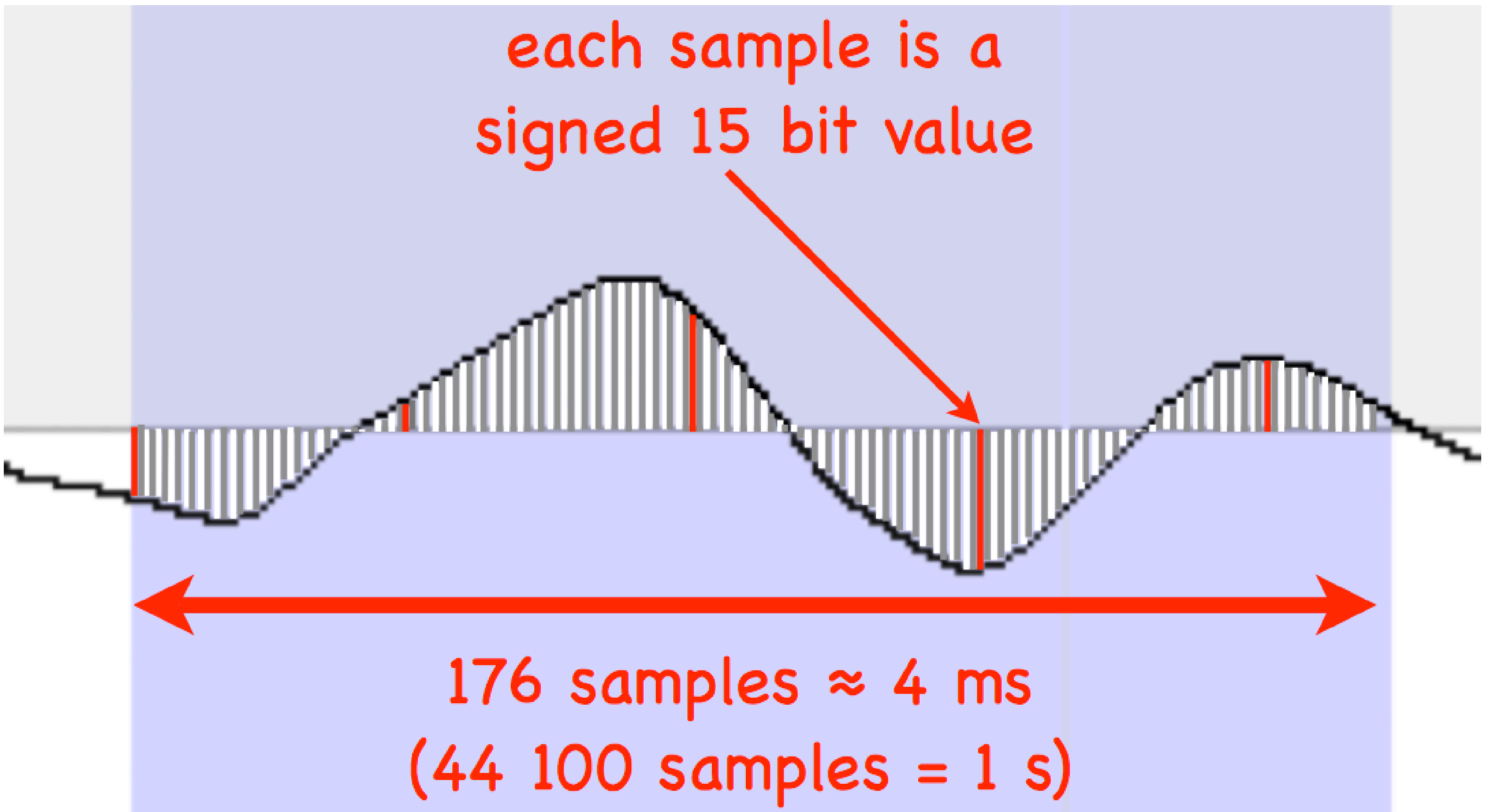
Same signal in more detail



Digitizing sound

- A digital audio file represents a sequence of measurements of air pressure as a function of time.
  - Average atmospheric air pressure is called 1 atmosphere.
- Audio files represent the difference between the air pressure at a point in space and the constant atmospheric pressure.
  - That point in space corresponds to the position of the microphone.
  - The ambient atmospheric pressure is the mean value of the measurements.
- High quality audio uses 16 bits (i.e. 2^{16} levels) to partition the range of possible pressures into equal units.
- This represents about $\pm 2^{15}$ levels above and below atmospheric pressure. Can you explain why?

AIFF sound format



AIFF sound format

- Raw high quality audio is typically sampled at rate of 44,000 pressure measurements per second.
  - Remember each sample requires 16 bits (= 2 Bytes) to store.
  - So we need 88,000 Bytes to store one second of sound.
  - Need 176,000 Bytes/s (or 176 KB/s) for stereo sound.
- CDs store all of this data without any compression.
  - 1 CD has ~ 700 MB. That corresponds to ~75mins of music.

Choosing a sampling rate

- Why use a sampling rate of 44,000 measurements per second?

  The ear is not sensitive to variations that are faster than this.

- Think of the ear drum as a mechanical device. Naively you can think of this limit on sensitivity as a mechanical limit on how fast the ear drum can vibrate to follow changes in the air pressure.

- If we use fewer samples, the sound quality is not as good.

Need a way to compress all this data!

Differential coding

- Audio files are compressible because the pressure often changes slowly from one time sample to the next.
- We could store the original sequence of measurements:

$X_1 , X_2 , ... , X_n$

- But we are assuming that X_j and X_{j+1} tend to have nearly the same value for all j .
- One strategy for encoding such a file is to store the <u>differences</u>:

$X_1 , X_2 - X_1 , X_3 - X_2 , ... , X_{j+1} - X_j , ... , X_n - X_{n-1}$

Why does differential coding work?

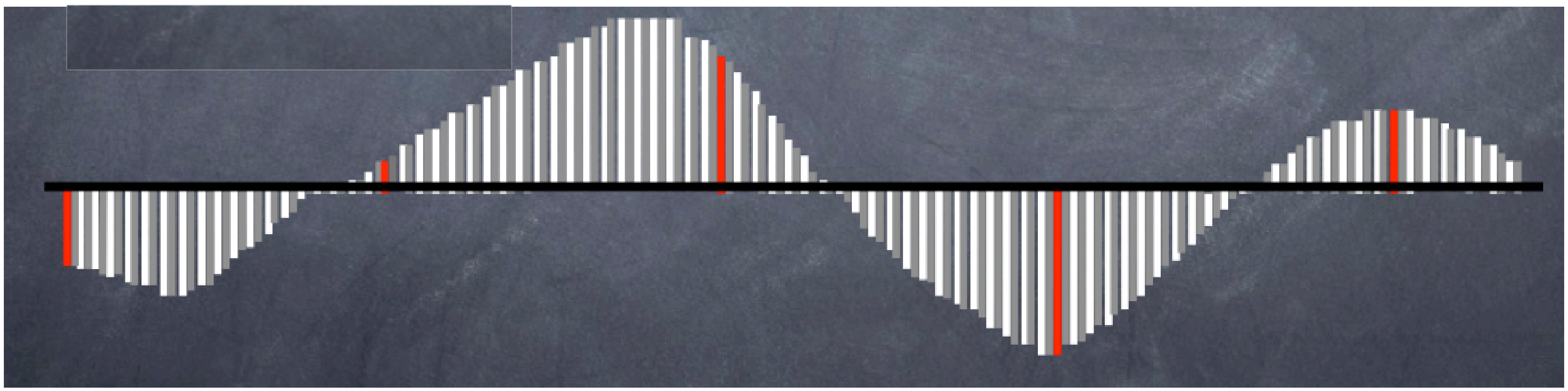
- In the original sequence, there are 2^{16} sound levels.
- But there are twice as many possible differences.
  - Max possible difference is 2^{16}. Min possible difference is -2^{16}.
  - So the alphabet of possible symbols is twice as large.
- However the differences change less, and tend to be very small (e.g. near "0").
- Differential coding idea: use short codewords for values near "0", and long codewords for large differences.

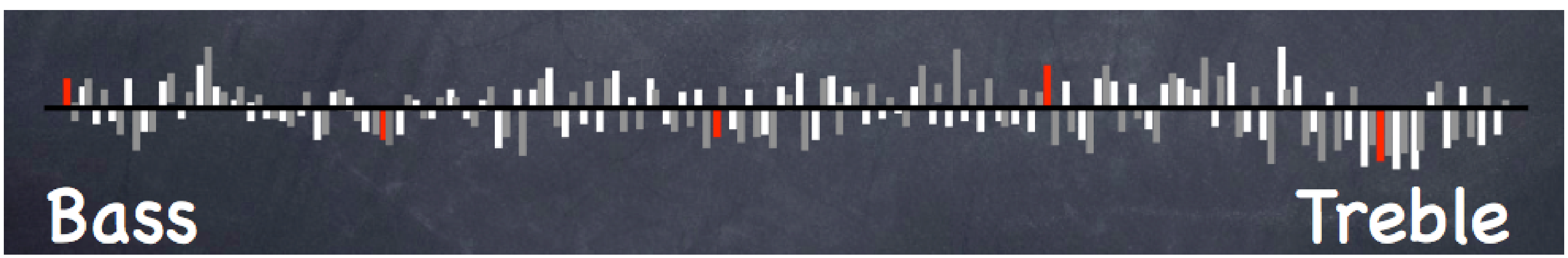
MP3 sound format

- MP3 = MPEG-1 Audio Layer 3.
- This is a lossy encoding for digital audio data.
  - "lossy" = do not preserve all 44,000 samples/sec exactly
- Can you tell when you listen? Hopefully not!
  - But: Compression works by reducing accuracy in ranges of the audio spectrum in which most people are not very sensitive.

MP3 frequency encoding

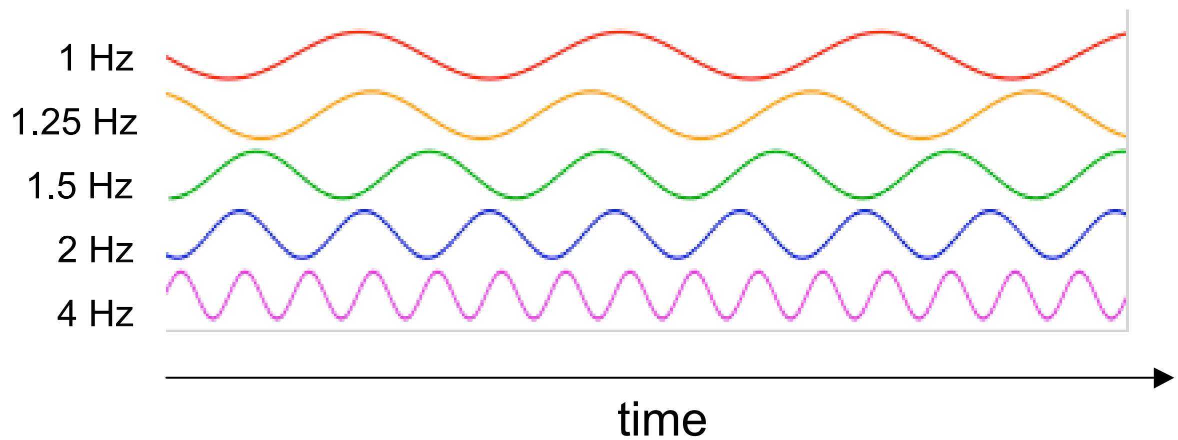
- Based on the Fourier transform.
- 576 samples of amplitude/time are converted to a representation based on distinct frequencies.





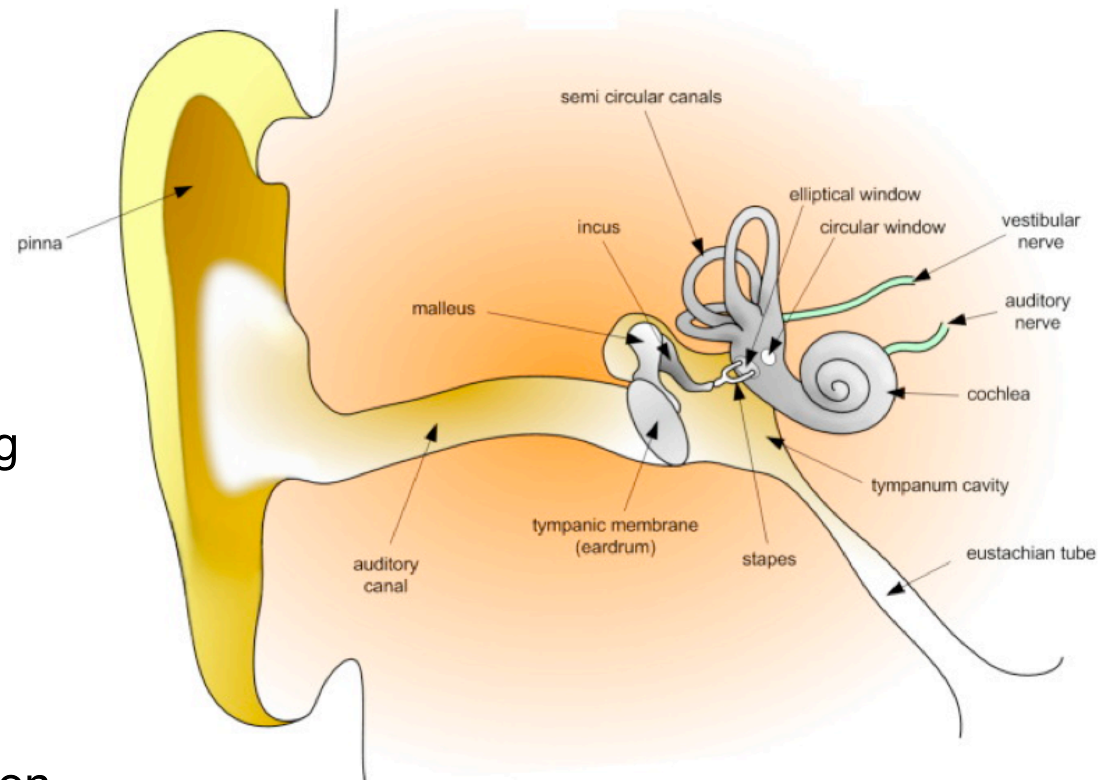
What are frequencies?

- **Definition**: The frequency is the number of times a specific event occurs per unit of time.
  - E.g. Heart rate of 80 beats per minute.
- Fourier showed that any sound wave can be decomposed into a combination of sinusoidal waves of different frequencies.



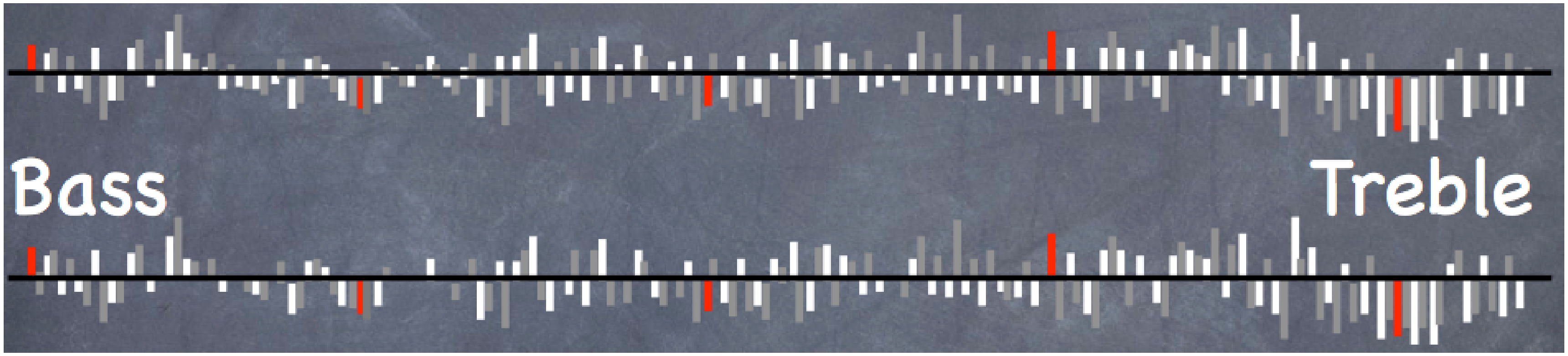
Understanding the human ear

- In human ears, the cochlea is mechanically performing a process analog to the Fourier Transform.
- The eardrum vibrates back and forth according to the wave-like representation of the sound.
- The frequency information stimulates a specific area in the cochlea.



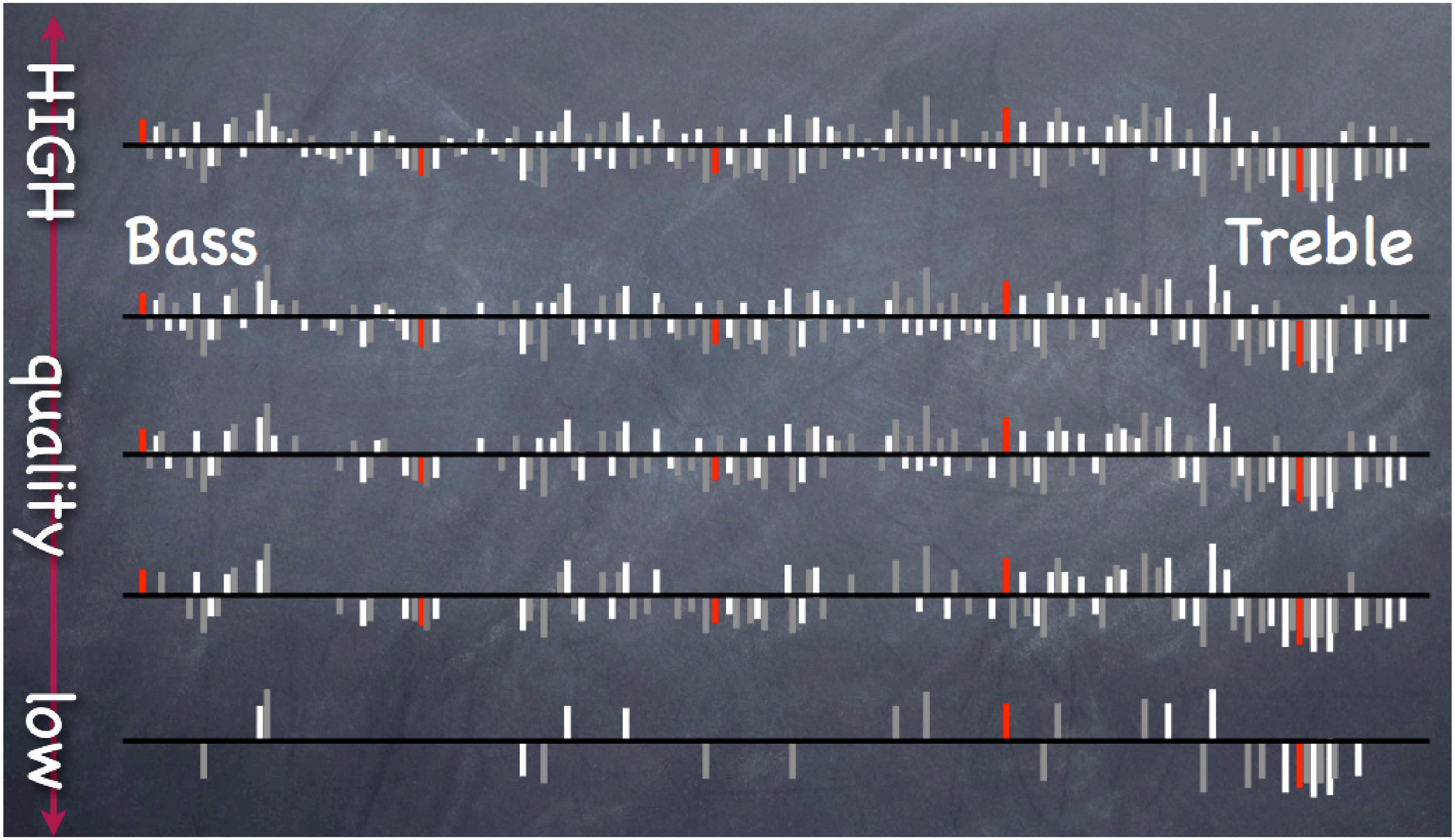
MP3 lossy encoding

- Frequencies with small coefficients are removed.



- Waveform reconstructed is close to the original.

More compression = lower quality



Lossless Image Encoding

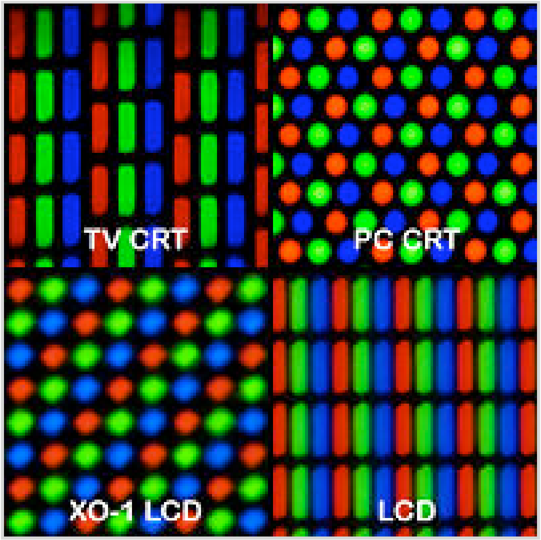
Digital images:

- Each image location is called a **pixel**.
- A **group of bits** represents the image content at each pixel.
- In **black&white** images, only need 1 bit for each pixel.
- In **colour** images, the number of bits for each pixel controls the number of colours available.
  - 1 bit = 2 colours
  - 2 bits = 4 colours
  - 8 bits = 256 colours
  - 16 bits = 65,536 colours (usually 5 bits each for red, blue, green + 1 transparency bit)

Displaying pixels

- Some displays are only able to display three basic colours, and not at the same site.

- In that case, a set of subpixels is used to give the impression of the correct colour when seen at a distance.

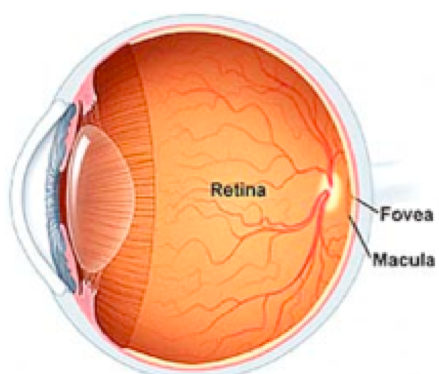


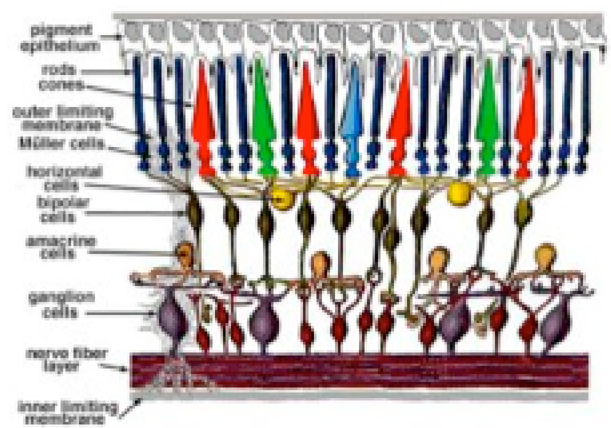
Pixel geometry of various CRT and LCD displays.

http://en.wikipedia.org/wiki/Pixel

Human visual perception

- Animal eyes focus light on the retina where an image of the environment is produced.
- This image is analyzed according to 3 types of colour sensitive cones, mostly triggered near the red, green and blue bands.
- A perceived colour is a triplet (x,y,z) of excitations of the 3 types of cones.
- Two combinations of colours yielding the same triplet (x,y,z) are indistinguishable.





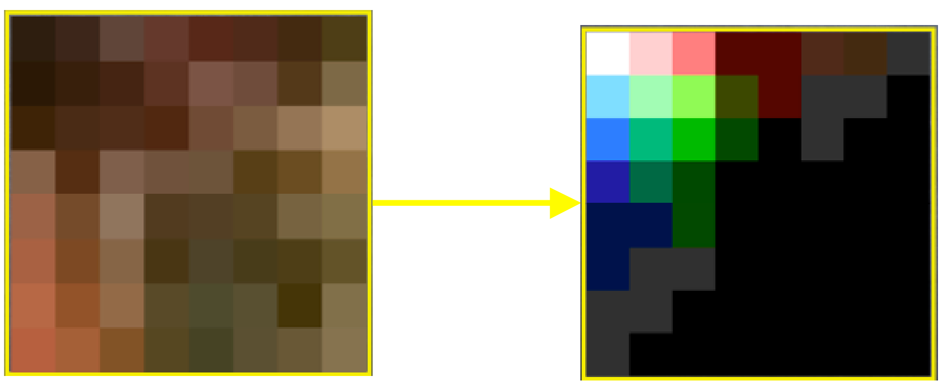
TIFF image format

- Image is separated into pixels.
- Consider an 8x8 sub-image.

- Each individual pixel uses
  24 bits (=3 Bytes):
  - 8 for red, 8 for blue, 8 for green
- Total size of the image =
  number of pixels * 3 Bytes (no compression)

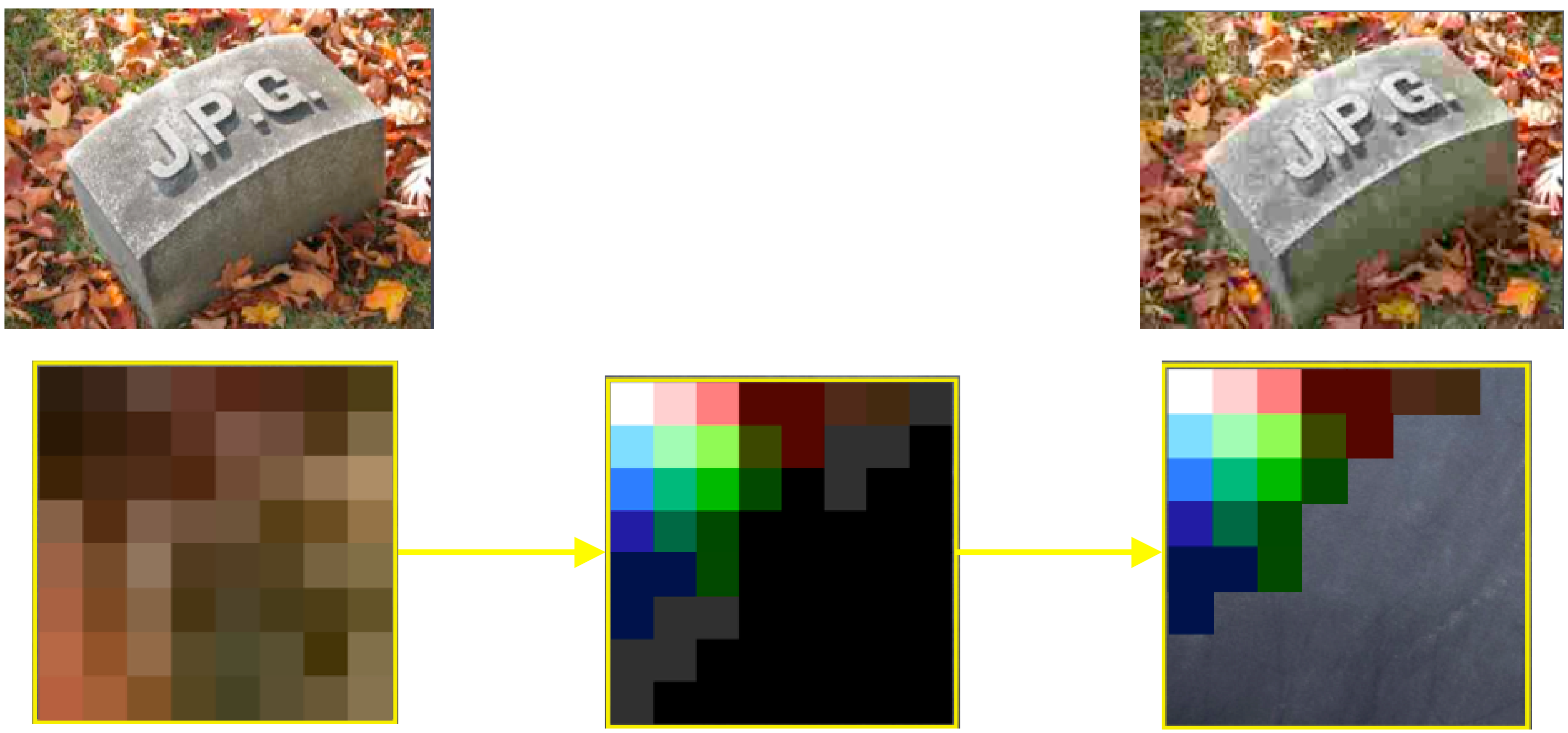
JPEG

- Break image into 8x8 (non-overlapping) sub-blocks.
- Use a transformation similar to Fourier transform (as in MP3 encoding), applied to each sub-block.
- Encode the coefficients of that transformation (similar to differential coding).
  - Colours are used for abstract data. Dark means small change from the mean, bright means large change.



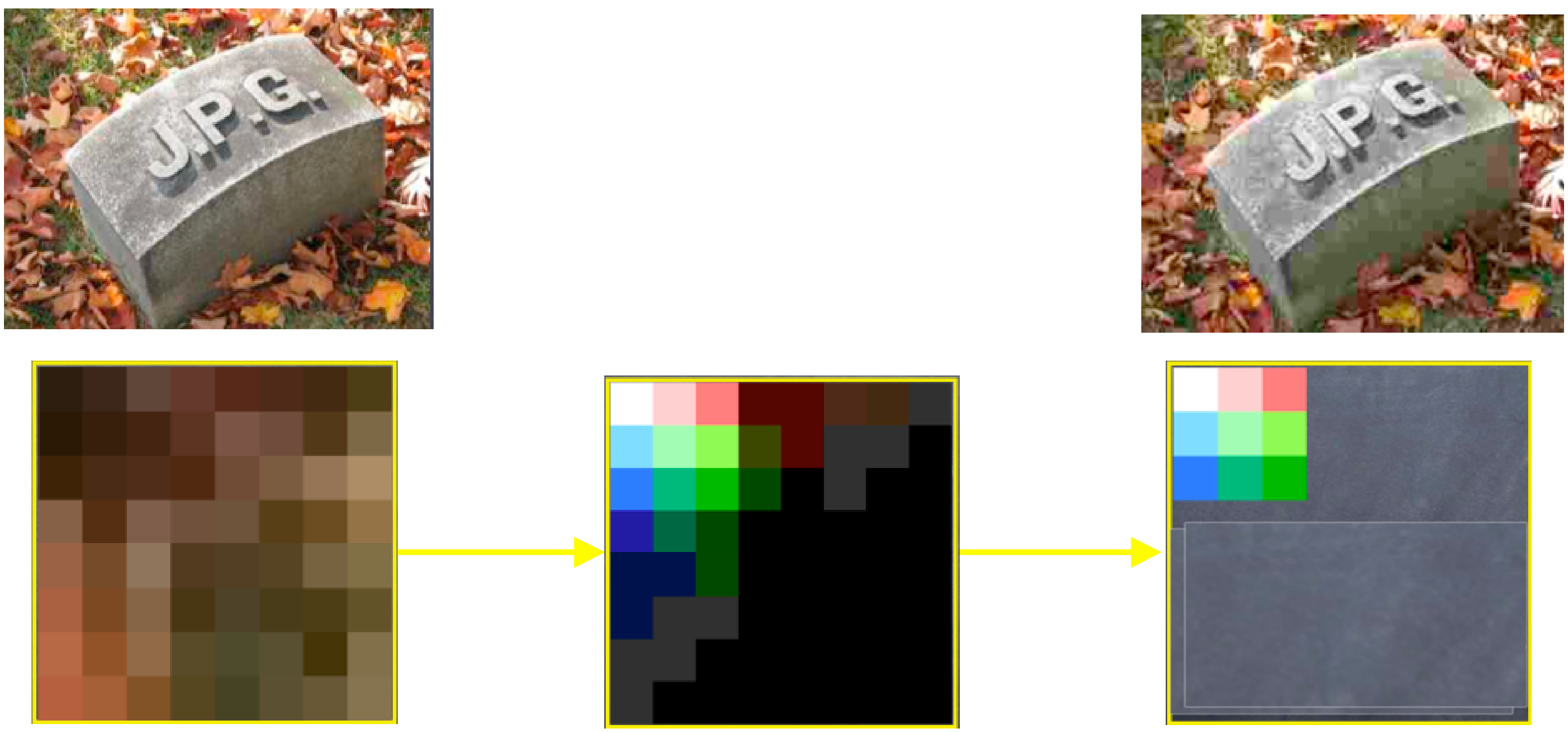
JPEG

- If no data is removed, the resulting image is nearly identical to the original.
- If all data <u>really close to zero</u> is removed, the resulting image is only slightly different from the original.



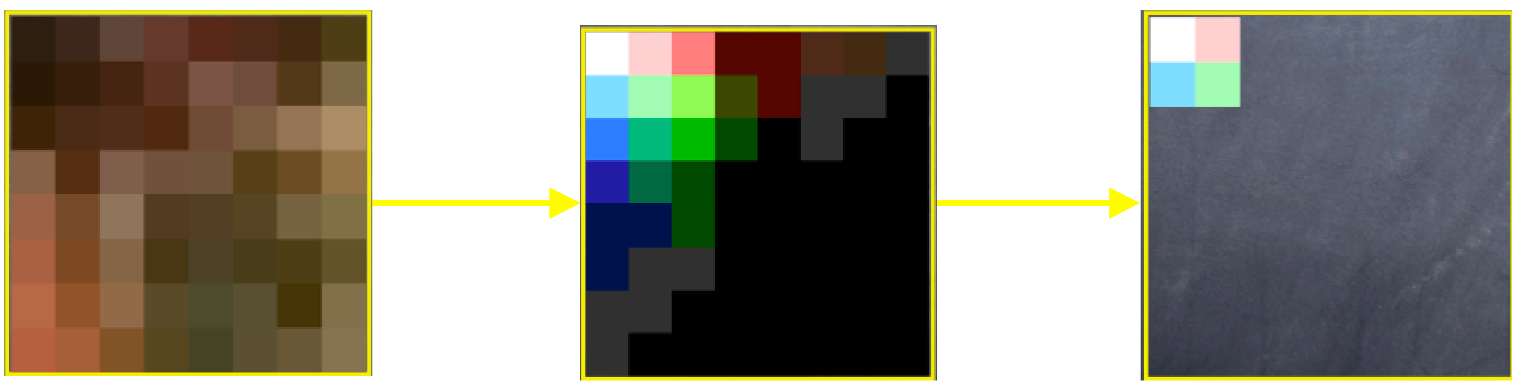
JPEG

- If all data <u>of small magnitude</u> is removed, the resulting image is still very similar to the original.



JPEG

- If only data <u>of large magnitude</u> is left, the resulting image is similar, but quite different from the original. Most details are wiped out.



Movie Representation

RAW movie format

- 720×576 pixels per frame
- 24 bits (colour) per pixel
- 30 frames per second
- 30 x 3 x 720 x 576 = 37 MB/s = 135 GB/hour
- typically 200 GB per movie !!! (= 50 DVDs)

MPEG2 movie format

- Each image is encoded with JPEG or similar format.
- Sound is encoded with MP3 or similar format.
- Most frames use only small amount of info to construct from previous frames (another version of differential coding).
  - Assume fixed background images, encode only the moving pixels.
    - Saving is about 96%.
  - Assume a traveling background, then only need to encode the direction and speed of the travel.
  - A complete frame is displayed occasionally to ensure the fixed part (or traveling part) has not substantially changed.

Compression in natural language

- Most languages, including English, have a lot of data compression built in.

  E.g. **Joelle picked up a book. She placed it on a table.**

- The use of pronouns is an instance of compression in natural language.

- Is this **lossless** or **lossy** compression?

Take-home message

- Basic methods for encoding sound (AIFF, MP3), images (TIFF, JPEG) and movies (MPEG).
- Lossless encoding vs lossy encoding.
- The basic idea of differential encoding, and how it applies to sound, images and video.

Comments

- Material for these slides was taken from:
  - *http://www.cim.mcgill.ca/~langer/423/lecture22.pdf*
  - *http://crypto.cs.McGill.ca/~crepeau/COMP102/Bits%20and%20Bytes3.pdf*