Lecture 7: Compression III

Audio Coding

Reading: Book Chapter 9.1-4, 9.7

Next 3 Weeks' Schedule

- Today
  - Audio compression

Moving to Media Distribution

- Wed.
  - Media Distribution (Review of IP networks)
- Mon., April 26
  - Media over IP
- Wed., April 28
  - No class, prepare for mid-term & work on hw3 and lab 2
- Mon., May 3
  - Mid-term
- Wed. May 5
  - Youtube + P2P VoD

RECAP

Redundancy in Media Data

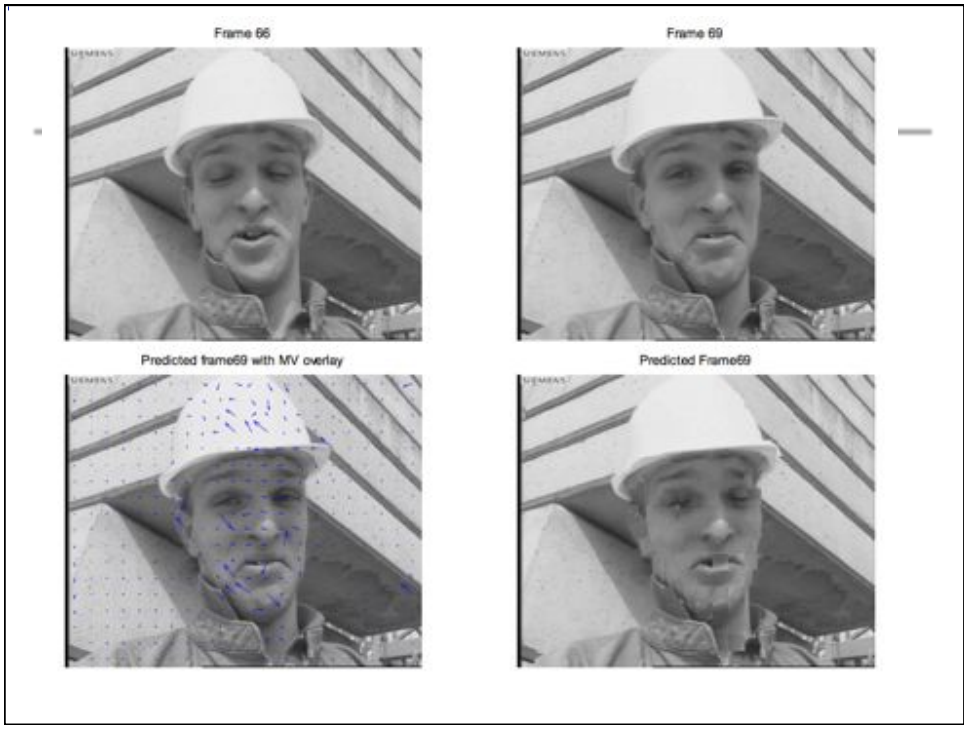
- Medias (speech, audio, image, video) are not random collection of signals, but exhibit a similar structure in local neighborhood
  - **Temporal redundancy**: current and next signals are very similar (smooth media: speech, audio, **video**)
  - **Spatial redundancy**: the pixels' intensities and colors in local regions are very similar
  - **Spectral redundancy**: When the data is mapped into the frequency domain, a few frequencies dominate over the others
- What is different with video?
  - We can apply JPEG to compress each video frame (Motion-JPEG)
  - But we can do more than that to achieve higher compression!

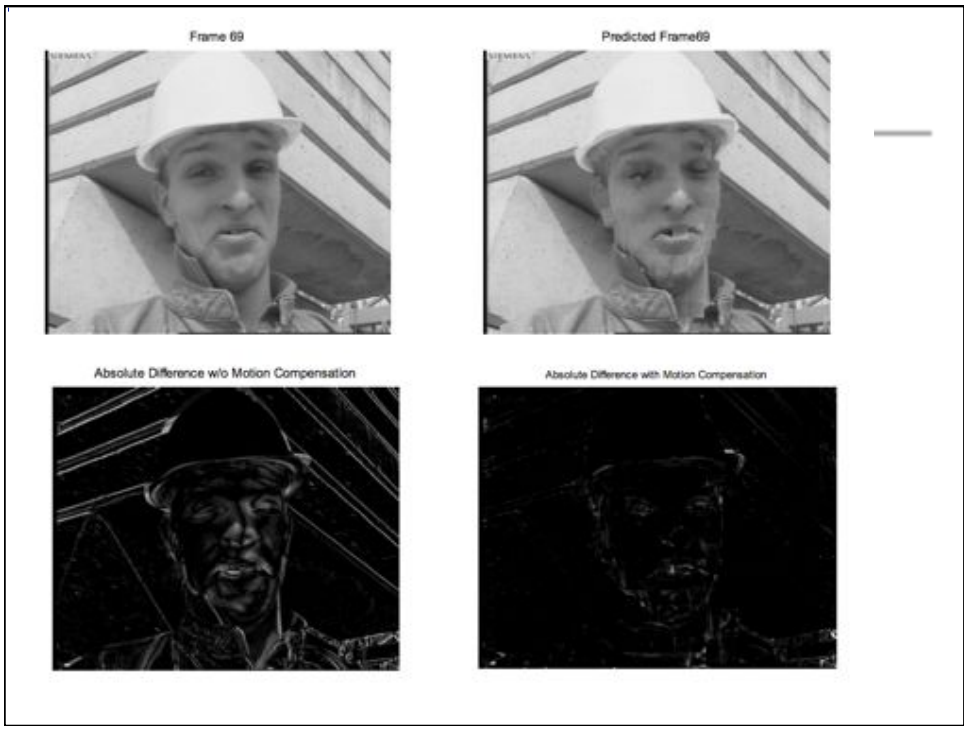
Motivation

- A video consists of a time-ordered sequence of frames, i.e., images.
- Adjacent frames are similar
- Changes are due to object or camera motion

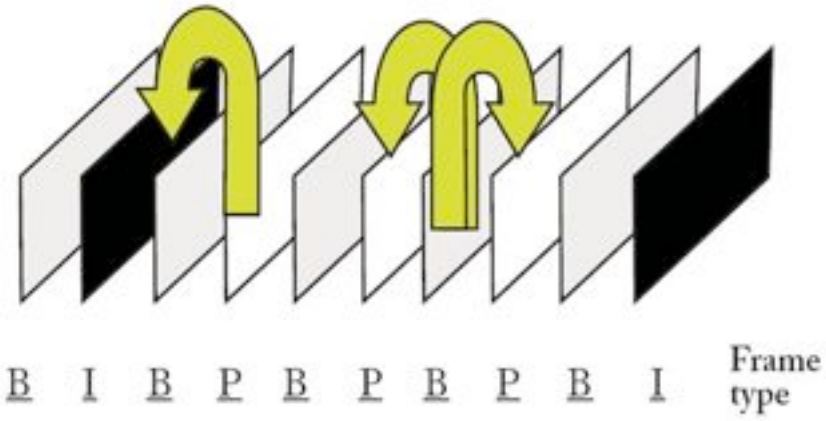
Key Concepts of Video Compression

- **Temporal Prediction**: (INTER mode)
  - Predict a new frame from a previous frame and only specify the prediction error
  - Prediction error will be coded using an image coding method (e.g., DCT-based JPEG)
  - Prediction errors have smaller energy than the original pixel values and can be coded with fewer bits
- **Motion-compensation** to improve prediction:
  - Use motion-compensated temporal prediction to account for object motion
- **INTRA frame coding**: (INTRA mode)
  - Those regions that cannot be predicted well are coded **directly** using DCT-based method
  - **Spatial prediction**: use spatial directional prediction to exploit spatial correlation (H. 264)
- **Work on each macroblock (MB) (16x16 pixels)** independently for reduced complexity
  - Motion compensation done at the MB level
  - DCT coding of error at the block level (8x8 pixels or smaller)
  - Block-based hybrid video coding


Frame 66
Frame 69
Predicted frame69 with MV overlay
Predicted Frame69


Frame 69
Predicted Frame69
Absolute Difference w/o Motion Compensation
Absolute Difference with Motion Compensation

Different Prediction Modes



Intra: coded directly;
Predictive: predicted from a previous frame;
Bidirectional: predicted from a previous frame and a following frame.
Can be done at frame or block levels

DCT on I frames/blocks

- For I-blocks, DCT is applied to original image values

DCT transform each 8x8 DCT block

↓

Quantize the DCT coefficients with properly chosen quantization matrices (different matrices for Y and C)

↓

The quantized DCT coefficients are zig-zag ordered and run-length coded

DCT on P frames/blocks

- First predict the current frame/block from the previous video frame
- DCT is applied to temporal prediction errors

For each macroblock (16x16), find the best matching block in a previous frame, and calculate the prediction errors

↓

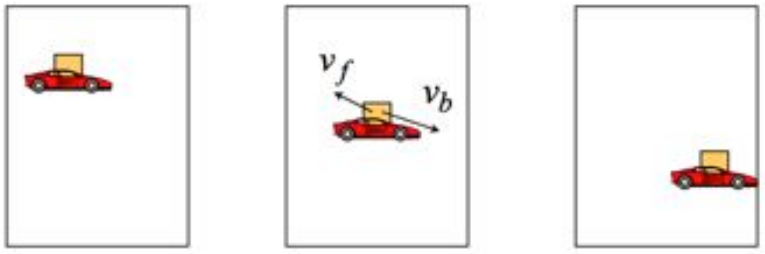
The prediction errors in each of the DCT blocks (8x8) are DCT transformed, quantized (according to specified QP), zig-zag scanned, and run-length coded

↓

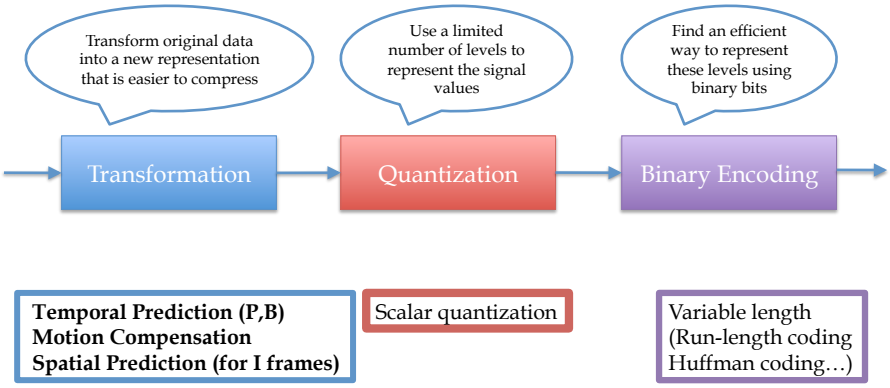
1 pair of motion vector (MV) also needs to be coded

DCT on B Frames/blocks

- Same as for the P-mode, except that a macroblock is predicted from both a previous picture frame and a following one
- 2 sets of MVs needed to be coded.



Summary: A Typical Video Compression System

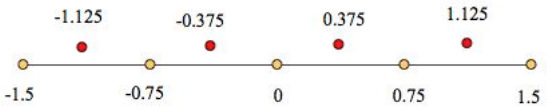


Today

- A bit more on **Quantization**
- Audio Coding/Compression

Quantization

- Map an amplitude value into a set of discrete value a→**kQ k=floor(a/Q) or ceil(a/Q)**
  - Instead of storing a, now store **k**
  - Represent **k** with R bits
- Example
  - A signal's DCT value is within the range of [-1.5, 1.5]
  - Let Q=3/4=0.75, divided into 4 levels



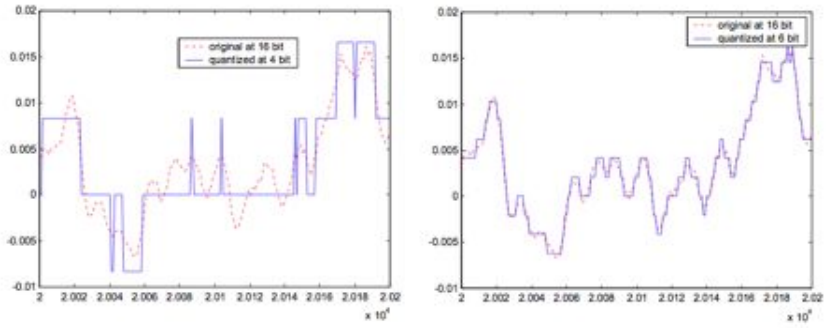
Yellow dots indicate the partition levels (boundaries between separate quantization intervals)
Red dots indicate the reconstruction levels (middle of each interval)

Uniform Quantization

- Applicable when the signal is in a finite range (f_{min}, f_{max})
- The entire data range is divided into L equal intervals of length Q (known as quantization interval orquantization step-size)
  - $Q=(f_{max}-f_{min})/L$
  - Any signal value **f** falls into Interval i is mapped to the middle value of this interval
  - We store/send only the index of quantized value **Q(f)=floor (f/Q)=k**
- De-quantization: f'= Q(f)*Q + Q/2;

How to determine Q

- Q depends on the dynamic range of the signal and perceptual sensitivity (and bit rate)
- $2^R=D/Q$
  - D is dynamic range, e.g. D=3 when the signal range is [-1.5, 1.5]
- For speech: *R = 8 bits;* For music*: R =16 bits;*



Matlab Demo

- Read and test quant_uniform.m

http://www.cs.ucsb.edu/~htzheng/teach/cs182/schedule/pdf/quant_uniform.m

- Adapted from Prof. yao wang's work

JPEG's Uniform Quantization

- After DCT, each AC is coded using uniform Quantization Q=16
- Each particular DC is coded using uniform Quantization, but Q depends on its location

- Except that the quantization is done slightly differently
  - **Q(f)=ceil(f/Q) f′ = Q(f)*Q**

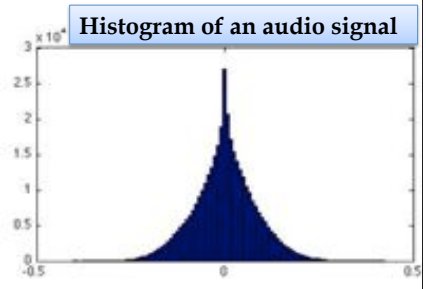
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Read Book Chapter 7, Section 4 For detailed quantization and DCT procedures

Non-Uniform Quantization

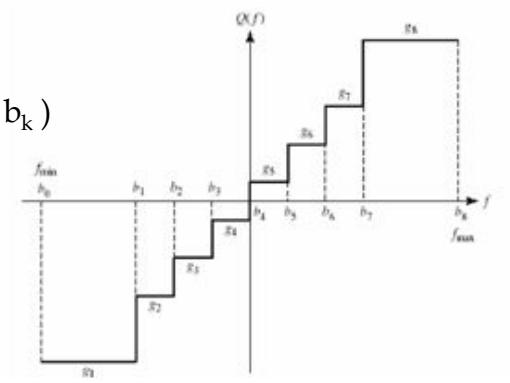
- Uniform quantization is good for uniformly distributed signal
- But real audio signals (speech and music) are more concentrated near zero
- Human ear is more sensitive to quantization errors at small values
- **Use smaller steps near zero!**

Histogram of an audio signal



Non-Uniform Quantization Procedure

- Quantization levels: L
- Partition values: b_k
- Partition regions :B= $[b_{k-1}, b_k)$
- Reconstruction values :g_k
  - in general, $g_k=(b_k-b_{k-1})/2$
- Quantized Index:
  - **Q(f)=k, if f $\in B_k$**
- Recovery: f′=g_k (given k)



A Particular Non-Uniform Quantization

- μ -law quantization
  - A companding algorithm that reduces the dynamic range of an audio signal
  - Reduce quantization error
- Given a signal x

$$F(x) = \text{sgn}(x)\frac{\ln(1+\mu|x|)}{\ln(1+\mu)} \qquad -1 \le x \le 1$$

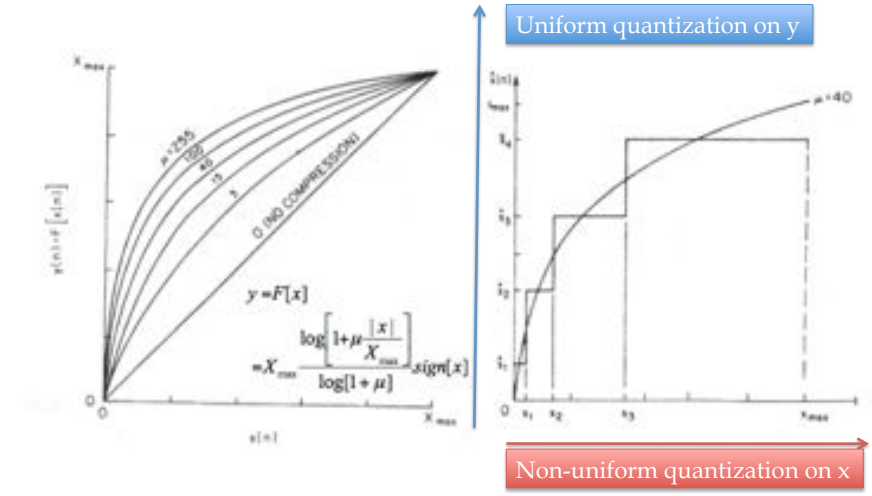
Quantize F(x) into k using a uniform quantizer

De-Quantize k to produce y

$$F^{-1}(y) = \text{sgn}(y)(1/\mu)((1+\mu)^{|y|} - 1) \qquad -1 \le y \le 1$$

μ = 255

Graphic View



μ -law transformation μ -law Quantization

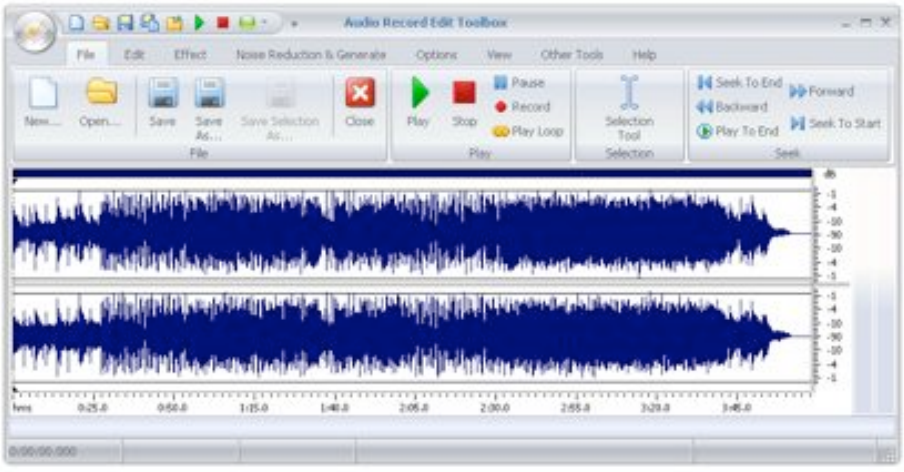
Matlab Demo

- Read and test mulaw_mulaw.m
  http://www.cs.ucsb.edu/~htzheng/teach/cs182/schedule/pdf/quant_mulaw.m
- Adapted from Prof. yao wang's work

AUDIO CODING

Image from www.acm.org/crossroads/xrds3-3/taskint.html

Audio as a Waveform



Speech vs. Audio

- **Speech coding**
  - Targeted for telephony applications
  - High rate waveform-based speech coder: for comfortable, natural sound, use simple **predictive coding** techniques (**4KHz bandwidth**, 8KHz sampling)
  - Low rate model-based speech coders: for intelligible speech, sufficient for communication purposes, use speech-production models (a filter driven by an excitation signal)
- **Audio coding**
  - For high quality production of music (including speech) in multiple channels
  - Music has a much wider bandwidth and multi-channels (~**20KHz bandwidth**, 44KHz sampling)
  - **Waveform**-based to retain the natural sound quality

    Need better compression mechanisms
  - Make extensive use of human hearing properties in **determining the quantization levels** in **different frequency bands**
    - Each frequency component is quantized with a step-size that depends on the hearing threshold
    - Don't code if the ear cannot hear it!

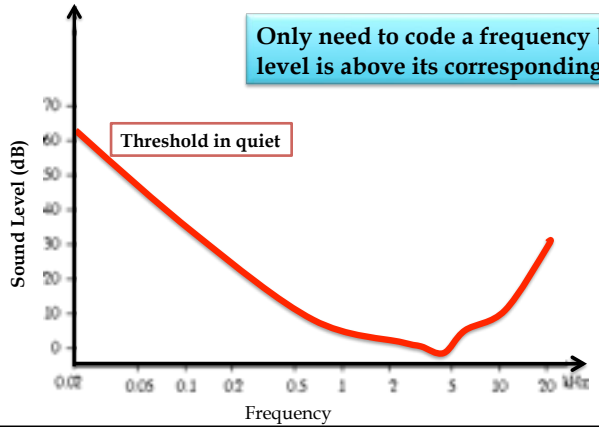
Your Ear as a Filterbank

- The auditory system can be roughly modeled as a filterbank, consisting of 25 overlapping bandpass filters, from **0 to 20 KHz**
- The ear cannot distinguish sounds within the same band that occur simultaneously.
- Each band is called a critical band
- The bandwidth of each critical band is about **100 Hz** for signals below 500 Hz, and increases linearly after **500 Hz** up to 5000 Hz
  - 1 bark = width of 1 critical band

$$\text{Bark} = \begin{cases} f/100, & f \le 500\text{Hz} \\ 9 + 4\log_2(f/1000), & f > 500\text{Hz} \end{cases}$$

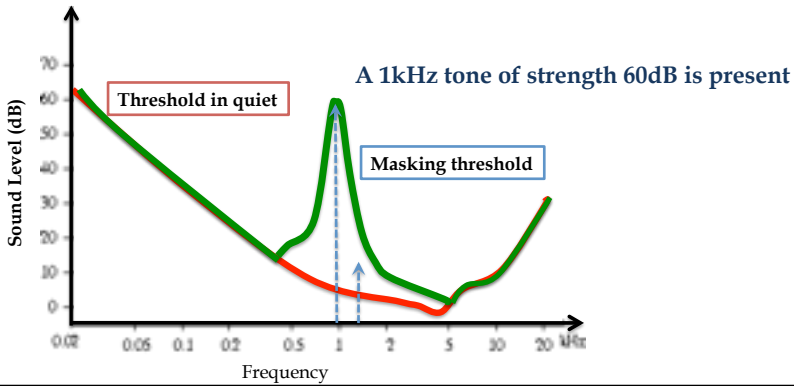
Threshold in Quiet

- Audible level at various frequencies:
  - The minimum sound level of an average ear with normal hearing can hear with no other sound present



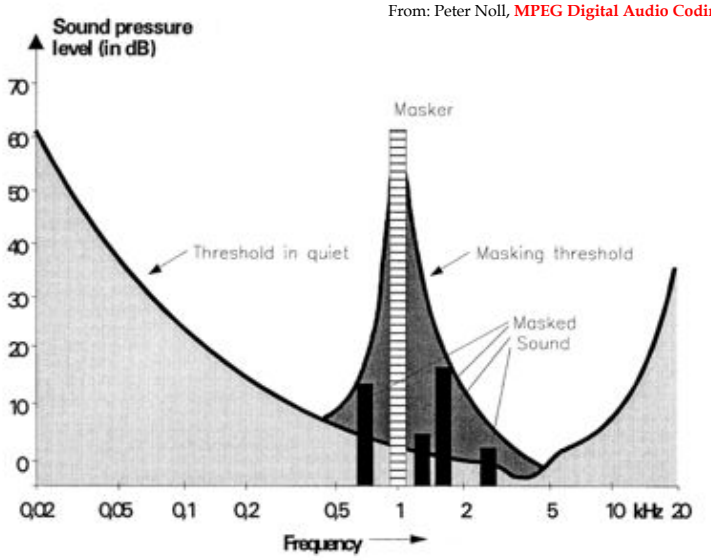
Frequency Masking

- When two sound frequencies are present in the signal **simultaneously,** the presence of one might hide the perception of the other
  - Also known as **simultaneous masking**
- A weak noise (the maskee) can be made inaudible by simultaneously occurring stronger signal (the masker), e.g, a pure tone; if masker and maskee are close enough to each other in frequency.



One more Example

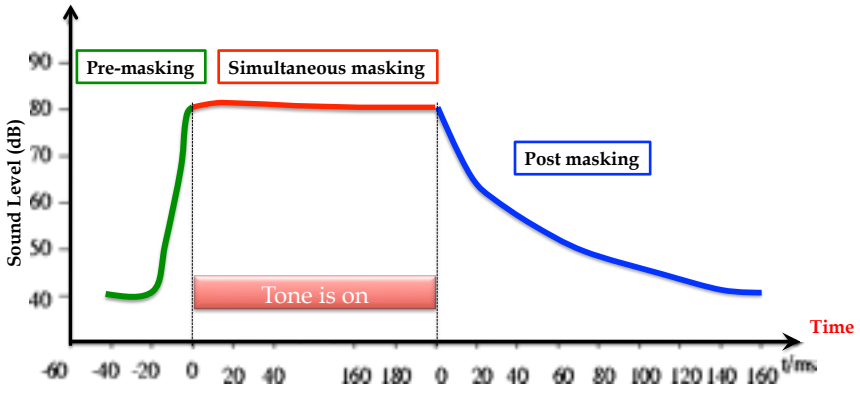
From: Peter Noll, MPEG Digital Audio Coding Standards



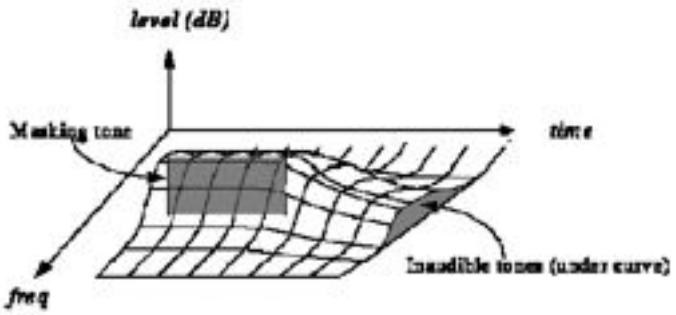
Temporal Masking

- If we hear a loud sound, then it stops, it takes a little while until we can hear a soft tone nearby

View from a particular critical band



Effect of Frequency and Temporal Masking



Perceptual Audio Coding

- **Decompose** a signal into separate frequency bands by using a filter bank
- **Analyze** signal energy in different bands and determine the total masking threshold of each band because of signals in other band/time
- **Quantize** samples in different bands with accuracy proportional to the masking level
  - Any signal **below the masking level does not need to be coded**
  - Signal above the masking level are quantized with a quantization step size according to masking level
  - Bits are assigned across bands so that each additional bit provides maximum reduction in perceived distortion.

Example

Band	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Level (db)	0	8	12	10	6	2	10	60	35	20	15	2	3	5	3	1

- Assume that If the level of the 8th band is 60dB, it gives a masking of 12 dB in the 7th band, 15dB in the 9th
- Level in 7th band is 10dB(<12dB), so ignore it
- Level in 9th band is 35dB (>15dB), so quantize, code and send it.

MPEG-1 Audio Coding

- **Sub-band filtering**: Divide the audio signal into 32 frequency subbands
- **Compute Masking**: Determine amount of masking for each band based on its frequency (threshold-in-quiet), and the energy of its neighboring band in frequency and time (frequency and temporal masking)
  - called the psychoacoustic model
- **Quantize or Ignore**:
  - If the energy in a band is below the masking threshold, don't encode it.
  - Otherwise, determine number of bits needed to represent the coefficient in this band such that the noise introduced by quantization is below the masking effect
  - (From quantization, 1 additional bit reduces the quantization nosie by 6 dB)
- **Format bitstream**:
  - insert proper headers, code the side information, e.g., quantization scale factors for different bands,
  - code the quantized coefficient indices, generally using variable length encoding, e.g. Huffman coding.

MPEG-1 Audio Layers

- **Layer 1**: DCT type filter with equal frequency spread per band. Psychoacoustic model only uses frequency masking.
- **Layer 2**: Same filter bank as layer 1. Psychoacoustic model uses a little bit of the temporal masking.
- **Layer 3 (MP3)**: Layer 1 filterbank followed by MDCT per band to obtain non-uniform frequency division similar to critical bands. Psychoacoustic model includes temporal masking effects, takes into account stereo redundancy, and uses Huffman coder.
  - because it provides good quality at an acceptable bit rate.
  - because the code for layer 3 is distributed freely.

Encoding/Decoding

Refer to the following article by Peter Noll

MPEG Digital Audio Coding Standards

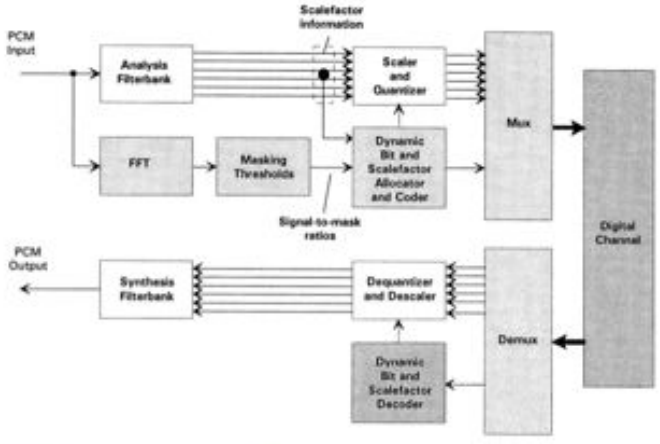
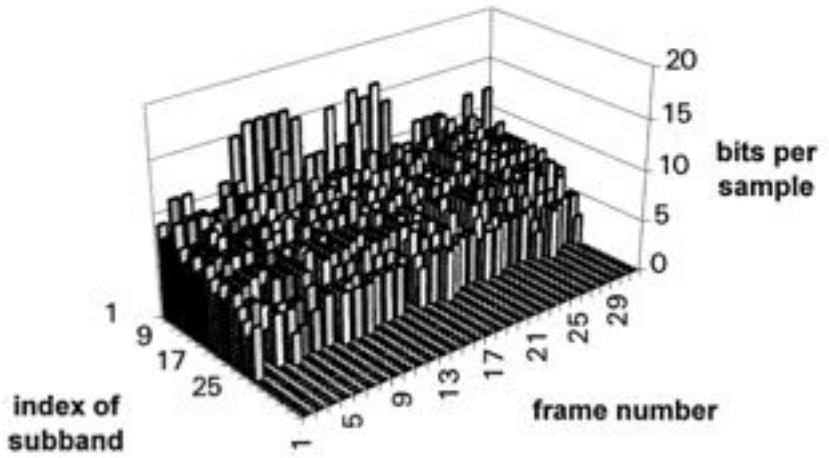


FIGURE 40.8: Structure of MPEG-1/Audio encoder and decoder, layers I and II.

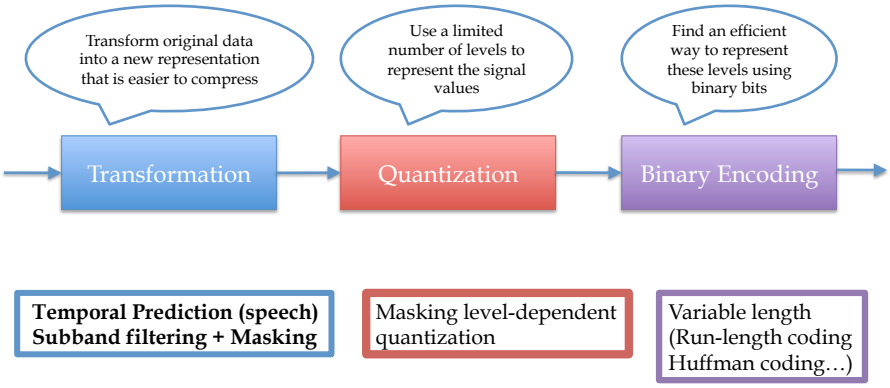
Dynamic Bit Allocation



Matlab MP3 Tools

- http://www.mathworks.com/matlabcentral/fileexchange/13852
- mp3read and mp3write
  - Behave like wavread, wavwrite but deal with mp3 files!
- Additional details:
  - http://labrosa.ee.columbia.edu/matlab/mp3read.html

Summary: A Typical Audio Compression System



What You Should Know

- Quantization
  - Uniform vs. Non-uniform quantization
  - Logarithm quantization (μ -law)
- The properties of the auditory system
  - Ear as a filterbank
  - Masking effects: threshold-in-quiet, frequency/ temporal masking
- Basic components in perceptual audio coding
  - Subband decomposition, bit allocation based on psychoacoustic model, quantization and coding
  - MPEG1 audio