ANDY W – 365 CheatSheet 10/23/2025
Audio Signal - vibrates back and forth and produces a longitudinal pressure. Vibrates a magnet and create signal voltage on a coil.
Amplitude X Time = not linear, log scale
Threshold of hearing: minimum sound level human can hear
Digitization: conversion to a stream of numbers (to integers)
Bel decibel(dB) ratio (nonlinear) +formula+
Quantization = sampling in amplitude or voltage dimension
Convert voltage level to bits: 8 bits = 256 possible values (loosy)
Sampling in time = separate by time interval
Rate of sampling = sampling frequency = sampling rate
PCM – pulse coded modulation (analog signal is continuous)
? samples/Sec at ? bits [sampling rate in Hz, kHz...]
Reduce sampling rate = aliasing, data loss,signal distortion,chunky
Sampling rate = actual frequency = false signal detected
1.5 times the actual frequency = incorrect allignment
Nyquist Theorem(rate) = 2x the max frequency in signal
Signal limited = 2x[upper limit(max) – lower limit(mini)]
Ex: 8k samples/sec, 8 bits = 8k * 8 = 64kbps
Mp3 uses 96, 128, 160kbps
SNR = Signal to noise ratio (ratio of power of signal and the noise)
SNR - log scale(nonlinear) – in decibel (dB) +formula+
Quantization Noise = actual – quantized value = max error is half of the interval (quantized to nearest boundary)
SQNR = signal to quantization noise ratio = measure the quality of quantization (similar to SNR) Peak SQNR = worst case - 6.02N(dB) in N bits/sample. Max signal = 2^(N-1-1) Min signal = -2^(N-1)
AUDIO FORMAT = WAV + MIDI
WAV header = 36 bytes long + 8 data header
1-4 = "RIFF"(shows it's a riff file, each character is 1 byte long)
5-8 = file size in int, 9-12 = "WAVE". 13-16 = "fmt", 17-20 = 16
21-22=1(type of format, 1=PCM). 23-24=2 (number of channels)
25-28=sampleRate(int). 29-32=ByteRate(SR*Bit/Sp*channels)/8.
33-34=BlockAlign=4. 35-36=Bits/Sample. Rest is data subchunks:
37-44 = (4)"data" + (4) size of data section(subchunk2Size)
MIDI = Musical Instrument Digital Interface
MIDI codes "events" that stand for the production of sounds
A single MIDI port has 16 channels, 3 IO, input, output and through (5 pins each)
HAS = Human auditory System
Human hearing rate range = 20Hz to 20kHz
Ex: mini sp rate for music = 40kHz (nyquist)
Ex: speech = 300Hz-4kHz, telephone system = 8kHz
Threshold of feeling = minimum sound level a person can hear
Th... of audibility = mini sound makes physical discomfort or pain
Threshold is different for different frequencies
Critical Bands = perceive the sounds through 25 distinct critical bands. Bandwidth is larger in higher frequencies
Masking effect - one strong signal can hide another with masking threshold. Sound above threshold is called masker (will show)
Sound under will be hidden (masked sound, inaudible signal)
The masking effects in the frequency domain.
Masking threshold in time domain: pre (soft) sound will maksed.
Simultaneous masking: Two sounds occur simultaneously and one is masked by the other. Post (soft 200ms after loud will maksed).
HAS: Audio Filtering = Prior to sampling and AD (Analog-to-Digital) conversion, the audio signal is also usually filtered to remove unwanted frequencies. Band-pass filter that screens out lower and higher frequencies. Before analog to digital conversion.

==========IMAGE==========
Light = electromagnetic wave. Color = wavelength.
Visible light = 500nm-700nm. Blue=short, red=long wavelength.
Human visual sys can absorb diff wavelength wwith cones and rods
RGB color space [0:255] 8 bits/color. RGBA(alpha) transparency
RGB - color conversion - YUV – inverse color conversion - RGB.
YUV Y = Luminous(lot of info), UV = chrominance (blue red)
YCrCb space = UV shifted by 0.5
conversion metrics YUV = A*RGB. (floating point – rounding error)
A = [0.299,0.587,0.114; -0.299,-0.587,0.886; 0.701,-0.587,-0.114]
CgYCo = A*GBR. A = [1/2,-1/4,-1/4; 1/2,1/4,1/4; 0,-1/2,1/2;]
GBR = A*CgYCo. A = [1, 1, 0; -1, 1, -1; -1, 1, 1;] ----inverse
Lifting = elementary matrix row operation(graphs with arrows)
Down Sampling (color Space) – improve compression (YUV)
YUV = Y:U:V (int). Luma sample = Y. chroma sample = UV.
YUV = 4:4:4 = no downsampling of chroma – same ratio
YUV = 4:2:2 = 2:1 horizontal downsampling of chroma.
2 chroma for every 4 luma sample – for storage only
YUV = 4:2:0 = 2:1 horizontal downsampling (widelyUsed)
1 chroma for every 4 luma sample. -- MPEG-1 OR MPEG-2
YUV images stored separately: YYY..UU..VV..V
Videos are stored by frames = array of images
CIF = Common Intermediate format Y = 352 X 288, UV = 176 X 144
QCIF = Quarter CIF. Y = 176 X 144, UV = 88X72 pixels

Gamma correction
Display device vs. Human Vision System (non-linear)
= non linear operation to adject the brightness and contrast
If the file value is RGB, screen emits light is (RGB)^Y
With gamma correction, raising ot the power of (1/Y)
= inverse the gamma to display the image to get a linear signal
Without correction, darker value are too dark. Correction will normalized to [0,1]

Graphics/Image Data Types
Image = BMP, GIF, JPG, EPS, PNG... Sound = AIFF, AAC, MP3, M4A, MOV... Video: AVI, MOV, MP4, MPG...
1Bit image, 1pixel = 1bit, binary, no color, only luma Y.
8Bit Gray-level image – 1pixel=1byte. Gray Value, only Luma.
= bitmap = image data, a 2D array of values. Resolution is the number of pixels, more = higher resolution and better quality
GrayScale to Bi-level = to binary. To Dithering = dot pattern
Dithering = halftone printing = intentional addition of a small amount of low-level random noise. A technique that uses a pattern of dots of varying size, density, or spacing to create the illusion of continuous-tone images like photograph.
High resolution dithering will increase the number of pixels
= for printing gray level on a 1-bit printer
= calculate square patterns of dots
= such that each gray level from 0 to 255 (i.e., darkness), corresponds to a filled dot pattern, more filled dots represent darker grayscale.
= not true grayscale and Each pixel is either a black dot or nothing
= Replace a pixel value by a larger pattern, say 2x 2 or 4 x 4
= The number of printed dots approximates the grayscale
= Half-tone printing is an analog process that uses smaller or larger filled circles of black ink to represent shading.
Ex: a 2 x 2 dither matrix. First, invert pixel value by 255 – pixel value. This is more convenient for us to run the dithering algorithm
= remap 0-255 to 0-4 range. If the intensity is greater than the dither matrix entry, print a dot at that location.
= each pixel = 2x2 array of dots, 4 times larger
Ordered Dithering=uses a pre-set threshold map tiled across image
Ordered Dithering ≠ Dithering.
Algorithm: Double for loop for x/y to xmax/ymax {
i=x mod n; j=y mod n; if I(x,y)> D(i,j){O(x,y) = 1} else{ O(x,y) = = 0} }
= let I(x,y) = input
Colored Dithering – Can only print RGB, create illusion of more color and smoother transitions.
24-bit color image – 3 bytes for RGB, usually stored in 32bit for transparency.
8-bit color images – uses color index lookup table (LUTs) to store color info, each pixel stores an index (8 bits). Greate for storage.
64/128 bit images, encode invisible light, medical images, multispectral = more than 3 color images.
8 bit GIF - first image type recognized by net browsers. Supports interlacing = successive display of pixels in widely-spaced rows by a 4-pass display process. File format of GIF87 contains: GIF signature, screen descriptor, global color map, (n times of: )[image descriptor, local color map, raster area,] GIF terminator
= Screen Descriptor comprises a set of attributes that belong to every image in the file. Contains screen width, height, m, cr, 0, pixel, bakcground(color index of screen background, defined in global color map).m = 1, global color map follows descriptor, cr+1 = num of bits of color resoltuion, pixel+1 = num of bits/pixel in image.
= color map, size = 2^(pixel + 1).
= image descriptor = 2 bytes for image left, top, width and height. 1 bit for m(m=0, uses global color map, m=1, local color map follows, use "pixel"). 1 bit for i (i=0, image formatted in sequential order, i= 1, in Interlaced order), 3 bits of '0', and 2 bits for pixel, which is number of bits per pixel for this image.
JPEG - currently the most important common file format.
PNG - Support for up to 48 bits of color information(better than GIF)
TIFF = Tagged Image File Format, support for attachment of additional information, referred to as tags, good flexibility.

==========VIDEO==========