Chapter 3. Multimedia Data Representations

3.1. Basics of Digital Audio

Reference: Chapter 3 of Steinmetz and Nahrstedt

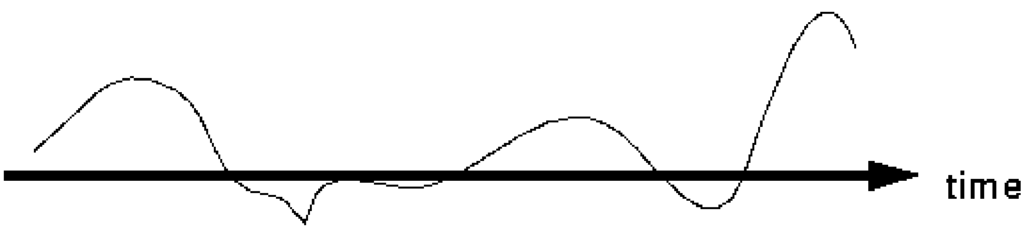
3.1.1 Digitization of Sound

Facts about Sound

- Sound is a continuous wave that travels through the air
- The wave is made up of pressure differences. Sound is detected by measuring the pressure level at a location.
- Sound waves have normal wave properties (reflection, refraction, diffraction, etc.).

Digitization in General

- Microphones, video cameras produce *analog signals* (continuous-valued voltages)



- To get audio or video into a computer, we must *digitize* it (convert it into a stream of numbers)

  So, we have to understand *discrete sampling* (both time and voltage)

- *Sampling* -- divide the horizontal axis (the time dimension) into discrete pieces. Uniform sampling is ubiquitous.

  Quantization -- divide the vertical axis (signal strength) into pieces. Sometimes, a non-linear function is applied.

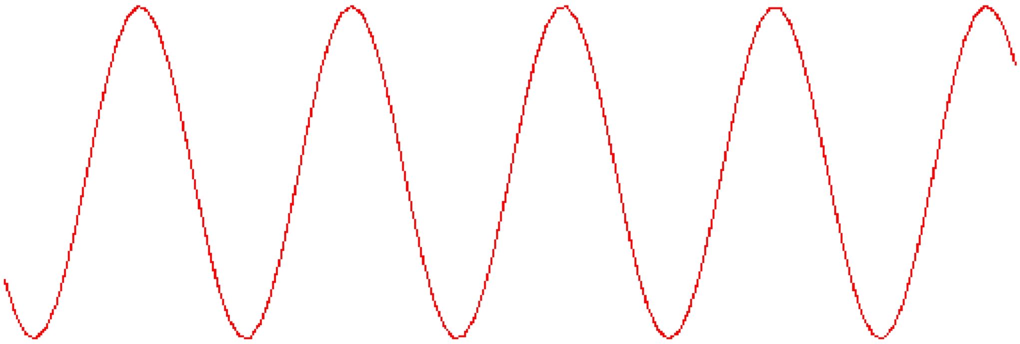
  - 8 bit quantization divides the vertical axis into 256 levels. 16 bit gives you 65536 levels.

Digitizing Audio

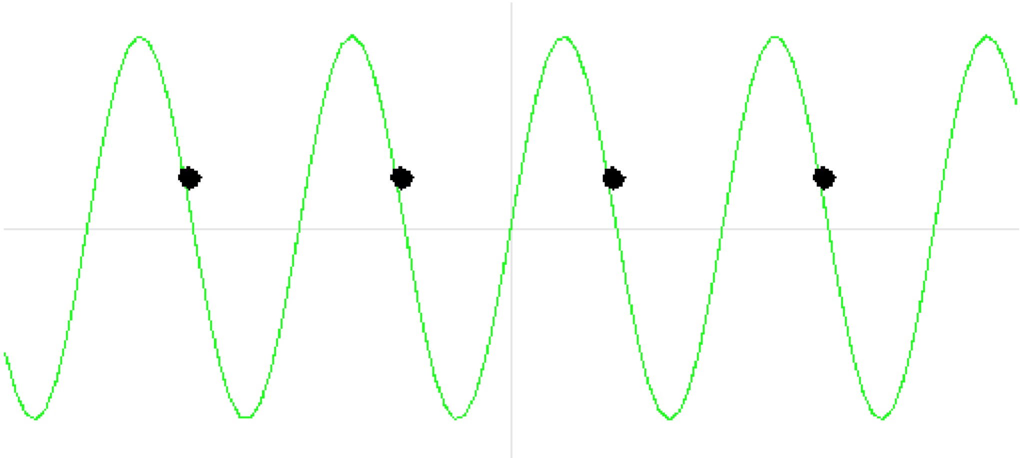
- Questions for producing digital audio (Analog-to-Digital Conversion):
    1. How often do you need to sample the signal?
    2. How good is the signal?
    3. How is audio data formatted?

Nyquist Theorem

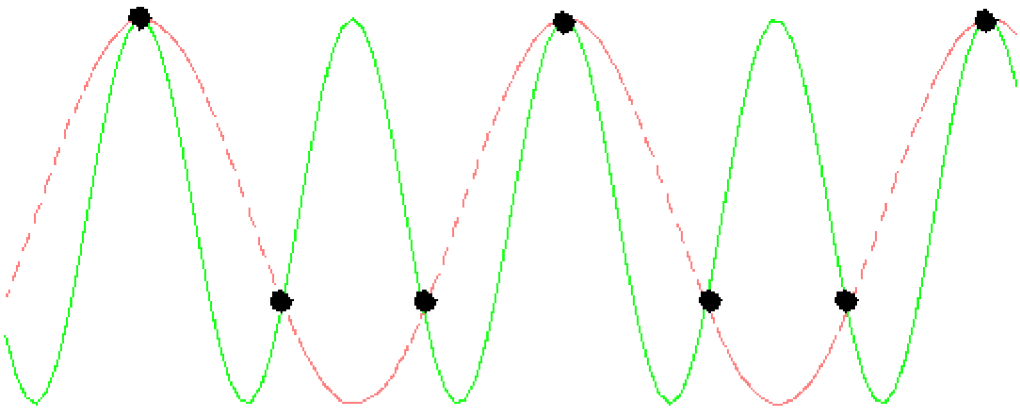
- Suppose we are sampling a sine wave. How often do we need to sample it to figure out its frequency?



- If we sample at 1 time per cycle, we can think it's a constant



- If we sample at 1.5 times per cycle, we can think it's a lower frequency sine wave --> *Alias*



- **Nyquist rate** -- For good digitization, the sampling rate should be at least twice of the maximum frequency.

Signal to Noise Ratio (SNR)

- In any analog system, some of the voltage is what you want to measure (*signal*), and some of it is random fluctuations (*noise*).
- Ratio of the power of the two is called the *signal to noise ratio* (*SNR*). SNR is a measure of the quality of the signal.
- SNR is usually measured in decibels (*dB*).

$$SNR = 10 \log \frac{V_{signal}^2}{V_{noise}^2} = 20 \log \frac{V_{signal}}{V_{noise}}$$

Signal to Quantization Noise Ratio (SQNR)

- The precision of the digital audio sample is determined by the number of bits per sample, typically 8 or 16 bits.

  The quality of the quantization can be measured by the Signal to Quantization Noise Ratio (SQNR).

- The *quantization error* (or *quantization noise*) is the difference between the actual value of the analog signal at the sampling time and the nearest quantization interval value.

  The largest (worst) quantization error is half of the interval.

- Given N to be the number of bits per sample, the range of the digital signal is - 2 exp (N-1) to 2 exp (N-1).

$$SQNR = 20 \log \frac{V_{signal}}{V_{quan-noise}} = 20 \log \frac{2^{N-1}}{\frac{1}{2}} = N \times 20 \log 2 = 6.02N \ (dB)$$

  In other words, each bit adds about 6 dB of resolution, so 16 bits enable a maximum SQNR = 96 dB.

  (** The above is for the worst case. Assume the input signal is sinusoidal, and the quantization error is statistically independent and its magnitude is uniformly distributed between 0 and half of the interval,
  SQNR = 6.02N + 1.76. [Pohlmann95, p. 37])

Linear and Non-linear Quantization

- Samples are typically stored as raw numbers (*linear format*), or as logarithms (*u-law* (or *A-law* in Europe)).
  - Logarithmic quantization approximates *perceptual non-uniformity*.

Typical Audio Formats

- Popular audio file formats include .au (Unix workstations), .aiff (MAC, SGI), .wav (PC, DEC workstations)

- A simple and widely used audio compression method is Adaptive Delta Pulse Code Modulation (ADPCM). Based on past samples, it predicts the next sample and encodes the difference between the actual value and the predicted value.

Audio Quality vs. Data Rate

Quality	Sample Rate (KHz)	Bits per Sample	Mono/ Stereo	Data Rate (Uncompressed)	Frequency Band
Telephone	8	8	Mono	8 KBytes/sec	200-3,400 Hz
AM Radio	11.025	8	Mono	11.0 KBytes/sec	
FM Radio	22.050	16	Stereo	88.2 KBytes/sec	
CD	44.1	16	Stereo	176.4 KBytes/sec	20-20,000 Hz
DAT	48	16	Stereo	192.0 KBytes/sec	20-20,000 Hz

- Telephone uses *u-law* encoding, others use linear. So the dynamic range of digital telephone signals is effectively 13 bits rather than 8 bits.
- CD quality stereo sound --> 10.6 MB / min.

Synthetic Sounds

- FM (Frequency Modulation) Synthesis -- used in low-end Sound Blaster cards, OPL-4 chip
- Wavetable synthesis -- wavetable generated from sound waves of real instruments
  - FM Synthesis is good for creating new sounds. Wavetables can store sounds of existing instruments nicely.
  - The wavetables are stored in memory on the sound card and they can be manipulated by software.
  - To save memory space, a variety of special techniques, such as sample looping, pitch shifting, mathematical interpolation, and polyphonic digital filtering can be applied.

Further Exploration

CD audio file formats

3.1.2 Introduction to MIDI (Musical Instrument Digital Interface)

Definition of MIDI: a protocol that enables computer, synthesizers, keyboards, and other musical device to communicate with each other.

1. Terminologies:

Synthesizer:

- It is a sound generator (various pitch, loudness, tone color).
- A good (musician's) synthesizer often has a microprocessor, keyboard, control panels, memory, etc.

Sequencer:

- It can be a stand-alone unit or a software program for a personal computer. (It used to be a storage server for MIDI data. Nowadays it is more a software *music editor* on the computer.)
- It has one or more MIDI INs and MIDI OUTs.

Track:

- Track in sequencer is used to organize the recordings.
- Tracks can be turned on or off on recording or playing back.

Channel:

- MIDI channels are used to separate information in a MIDI system.
- There are 16 MIDI channels in one cable.
- Channel numbers are coded into each MIDI message.

Timbre:

- The quality of the sound, e.g., flute sound, cello sound, etc.
- Multitimbral -- capable of playing many different sounds at the same time (e.g., piano, brass, drums, etc.)

Pitch:

- musical note that the instrument plays

Voice:

- Voice is the portion of the synthesizer that produces sound.
- Synthesizers can have many (16, 20, 24, 32, 64, etc.) voices.
- Each voice works independently and simultaneously to produce sounds of different timbre and pitch.

Patch:

- the control settings that define a particular timbre.

2. Hardware Aspects of MIDI

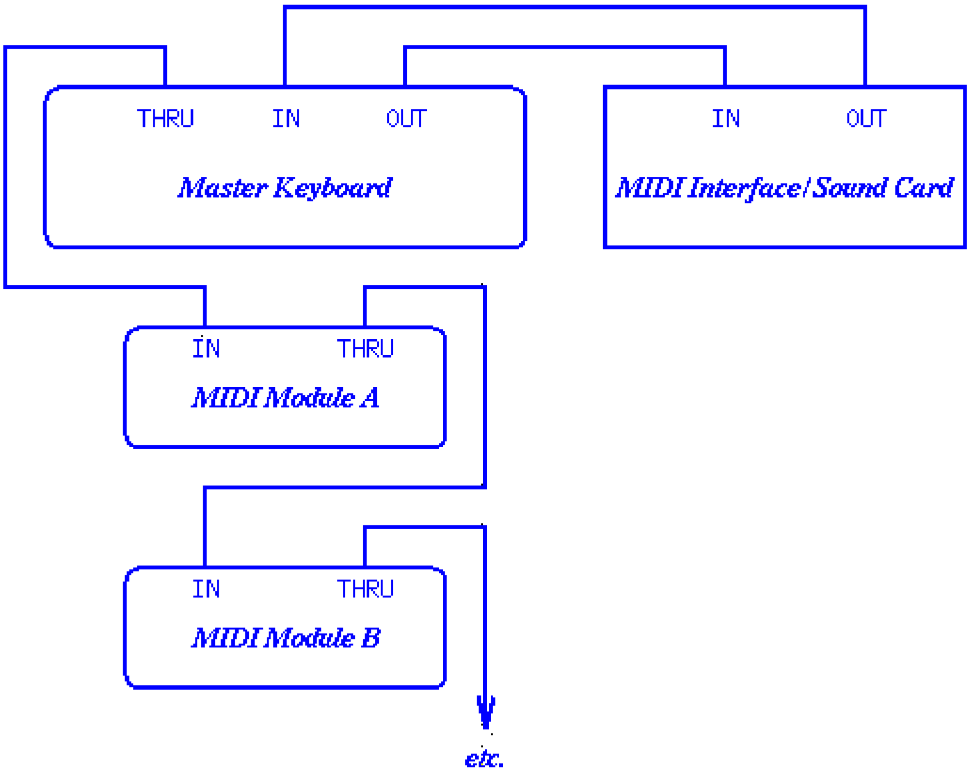
MIDI connectors:

-- three 5-pin ports found on the back of every MIDI unit

- MIDI IN: the connector via which the device receives all MIDI data.
- MIDI OUT: the connector through which the device transmits all the MIDI data it generates itself.
- MIDI THROUGH: the connector by which the device echoes the data receives from MIDI IN.

Note: It is only the MIDI IN data that is echoed by MIDI through. All the data generated by device itself is sent through MIDI OUT.

A Typical MIDI Sequencer Setup:



- MIDI OUT of synthesizer is connected to MIDI IN of sequencer.
- MIDI OUT of sequencer is connected to MIDI IN of synthesizer and "through" to each of the additional sound modules.
- During recording, the keyboard-equipped synthesizer is used to send MIDI message to the sequencer, which records them.
- During play back: messages are send out from the sequencer to the sound modules and the synthesizer which will play back the music.

3. MIDI Messages

-- MIDI messages are used by MIDI devices to communicate with each other.

Structure of MIDI messages:

- MIDI message includes a status byte and up to two data bytes.
- Status byte
  - The most significant bit of status byte is set to 1.
  - The 4 low-order bits identify which channel it belongs to (four bits produce 16 possible channels).
  - The 3 remaining bits identify the message.
- The most significant bit of data byte is set to 0.

Classification of MIDI messages:

```
                                                ----- voice messages
                    ---- channel messages -----|
                   |                            ----- mode messages
                   |
MIDI messages ----|
                   |                            ---- common messages
                    ----- system messages -----|---- real-time messages
                                                ---- exclusive messages
```

A. Channel messages:

-- messages that are transmitted on individual channels rather that globally to all devices in the MIDI network.

A.1. Channel voice messages:

- Instruct the receiving instrument to assign particular sounds to its voice
- Turn notes on and off
- Alter the sound of the currently active note or notes

Voice Message	Status Byte	Data Byte1	Data Byte2
Note off	&H8x	Key number	Note Off velocity
Note on	&H9x	Key number	Note on velocity
Polyphonic Key Pressure	&HAx	Key number	Amount of pressure
Control Change	&HBx	Controller number	Controller value
Program Change	&HCx	Program number	None
Channel Pressure	&HDx	Pressure value	None
Pitch Bend	&HEx	MSB	LSB

Notes: `x' in status byte hex value stands for a channel number.

Example: a Note On message is followed by two bytes, one to identify the note, and on to specify the velocity.
To play note number 80 with maximum velocity on channel 13, the MIDI device would send these three hexadecimal byte values: &H9C &H50 &H7F

A.2. Channel mode messages: -- Channel mode messages are a special case of the Control Change message (&HBx or 1011nnnn). The difference between a Control message and a Channel Mode message, which share the same status byte value, is in the first data byte. Data byte values 121 through 127 have been reserved in the Control Change message for the channel mode messages.

- Channel mode messages determine how an instrument will process MIDI voice messages.

1st Data Byte	Description	Meaning of 2nd Data Byte
&H79	Reset all controllers	None; set to 0
&H7A	Local control	0 = off; 127 = on
&H7B	All notes off	None; set to 0
&H7C	Omni mode off	None; set to 0
&H7D	Omni mode on	None; set to 0
&H7E	Mono mode on (Poly mode off)	**
&H7F	Poly mode on (Mono mode off)	None; set to 0

** if value = 0 then the number of channels used is determined by the receiver; all other values set a specific number of channels, beginning with the current basic channel.

B. System Messages:

- System messages carry information that is not channel specific, such as timing signal for synchronization, positioning information in pre-recorded MIDI sequences, and detailed setup information for the destination device.

B.1. System real-time messages:

- messages related to synchronization

System Real-Time Message	Status Byte
Timing Clock	&HF8

Start Sequence	&HFA
Continue Sequence	&HFB
Stop Sequence	&HFC
Active Sensing	&HFE
System Reset	&HFF

B.2. System common messages:

- contain the following unrelated messages

System Common Message	Status Byte	Number of Data Bytes
MIDI Timing Code	&HF1	1
Song Position Pointer	&HF2	2
Song Select	&HF3	1
Tune Request	&HF6	None

B.3. System exclusive message:

- (a) Messages related to things that cannot be standardized, (b) addition to the original MIDI specification.
- It is just a stream of bytes, all with their high bits set to 0, bracketed by a pair of system exclusive start and end messages (&HF0 and &HF7).

4. General MIDI

- MIDI + Instrument Patch Map + Percussion Key Map --> a piece of MIDI music sounds the same anywhere it is played
  - Instrument patch map is a standard program list consisting of 128 patch types.
  - Percussion map specifies 47 percussion sounds.
  - Key-based percussion is always transmitted on MIDI channel 10.

- Requirements for General MIDI Compatibility:
  - Support all 16 channels.
  - Each channel can play a different instrument/program (multitimbral).
  - Each channel can play many voices (polyphony).
  - Minimum of 24 fully dynamically allocated voices.

Appendix

A1. General MIDI Instrument Patch Map

A2. General MIDI Percussion Key Map

Further Exploration

Try some good sources for locating internet sound/music materials at

A tutorial on MIDI and wavetable music synthesis

YAHOO's Multimedia:Sound Page

Last Updated: ***5/18/98***

3.2. Graphic/Image File Formats

This section introduces some of the most common graphics and image file formats. Some of them are restricted to particular hardware/operating system platforms, others are *cross-platform* independent formats. While not all formats are cross-platform, there are conversion applications that will recognize and translate formats from other systems.

Most image formats incorporate some variation of a *compression* technique due to the large storage size of image files. Compression techniques can be classified into either **lossless** or **lossy**. We will study various video and audio compression techniques in Chapter 4.

3.2.1. Graphic/Image Data Structures

- **Pixels** -- picture elements in digital images
- **Image Resolution** -- number of pixels in a digital image
  (Higher resolution always yields better quality.)
- **Bit-Map** -- a representation for the graphic/image data in the same manner as they are stored in video memory.

Monochrome Image

Example of a Monochrome Bit-map Image

- Each pixel is stored as a single bit (0 or 1)
- A 640 x 480 monochrome image requires 37.5 KB of storage.
- *Dithering* is often used for displaying monochrome images

Gray-scale Images

Example of a Gray-scale Bit-map Image

- Each pixel is usually stored as a byte (value between 0 to 255)
- A 640 x 480 grayscale image requires over 300 KB of storage.

24-bit Color Images

Example of 24-Bit Color Image

- Each pixel is represented by three bytes (e.g., RGB)
- Supports 256 x 256 x 256 possible combined colors (16,777,216)
- A 640 x 480 24-bit color image would require 921.6 KB of storage
- Many 24-bit color images are stored as 32-bit images, the extra byte of data for each pixel is used to store an *alpha* value representing special effect information

8-bit Color Images

Example of 8-Bit Color Image

- One byte for each pixel
- Supports 256 out of the millions colors possible, acceptable color quality
- Requires Color Look-Up Tables (LUTs)
- A 640 x 480 8-bit color image requires 307.2 KB of storage (the same as 8-bit grayscale)

3.2.2. Standard System Independent Formats

The following brief format descriptions are the most commonly used formats. Follow some of the document links for more descriptions.

GIF (GIF87a, GIF89a)

- Graphics Interchange Format (GIF) devised by the UNISYS Corp. and Compuserve, initially for transmitting graphical images over phone lines via modems
- Uses the Lempel-Ziv Welch algorithm (a form of Huffman Coding), modified slightly for image scan line packets (line grouping of pixels)
- Limited to only 8-bit (256) color images, suitable for images with few distinctive colors (e.g., graphics drawing)
- Supports *interlacing*
- GIF89a supports simple *animation*
  (Graphics Control Extension has control over *delay time*, *transparent index*, etc. Software such as Coral Draw will allow access and editing of GIF images.)

JPEG

- A standard for photographic image compression created by the Joint Photographics Experts

Group

- Takes advantage of limitations in the human vision system to achieve high rates of compression
- Lossy compression which allows user to set the desired level of quality/compression
- Detailed discussions in Chapter 4.

TIFF

- Tagged Image File Format (TIFF), stores many different types of images (e.g., monochrome, grayscale, 8-bit & 24-bit RGB, etc.) --> tagged
- Developed by the Aldus Corp. in the 1980's and later supported by the Microsoft
- TIFF is a lossless format (when not utilizing the new JPEG tag which allows for JPEG compression)
- It does not provide any major advantages over JPEG and is not as user-controllable it appears to be declining in popularity

Graphics Animation Files

- FLC -- main animation or moving picture file format, originally created by Animation Pro
- FLI -- similar to FLC
- GL -- better quality moving pictures, usually large file sizes

Postscript/Encapsulated Postscript

- A typesetting language which includes text as well as vector/structured graphics and bit-mapped images
- Used in several popular graphics programs (Illustrator, FreeHand)
- Does not provide compression, files are often large

3.2.3. System Dependent Formats

Many graphical/imaging applications create their own file format particular to the systems they are executed upon. The following are a few popular system dependent formats:

Microsoft Windows: BMP

- A system standard graphics file format for Microsoft Windows
- Used in PC Paintbrush and other programs
- It is capable of storing 24-bit bitmap images

Macintosh: PAINT and PICT

- PAINT was originally used in MacPaint program, initially only for 1-bit monochrome images.
- PICT format is used in MacDraw (a vector based drawing program) for storing structured graphics

X-windows: XBM

- Primary graphics format for the X Window system
- Supports 24-bit color bitmap
- Many public domain graphic editors, e.g., *xv*
- Used in X Windows for storing icons, pixmaps, backdrops, etc.

Further Exploration

- A new graphics file format -- PNG (Portable Network Graphics)
- The PNG Specification by W3C
- Animated illustration of the PNG 2-D interlacing

Last Updated: ***5/18/98***

Top | Chap3"../../../../index.html"> CMPT 365 Home Page | CS

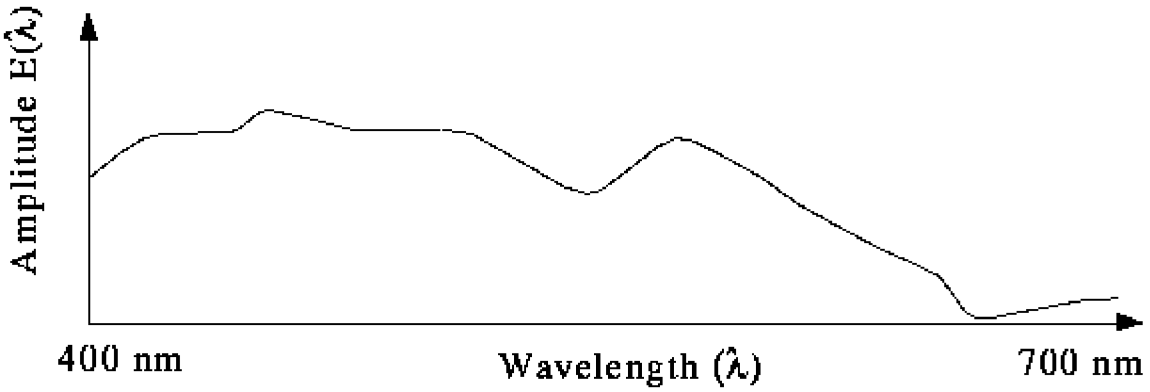
3.3. Color in Image and Video

3.3.1. Basics of Color

Light and Spectra

- Visible light is an electromagnetic wave in the 400 nm - 700 nm range.

  Most light we see is not one wavelength, it's a combination of many wavelengths.



- The profile above is called a *spectrum*.

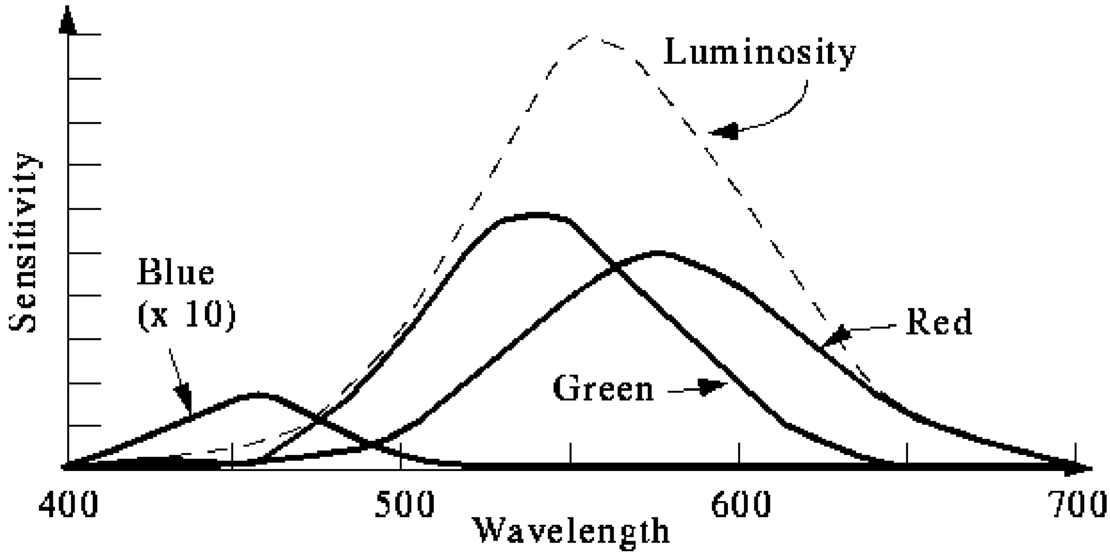
The Human Retina

- The eye is basically just a camera

  Each neuron is either a *rod* or a *cone*. Rods are not sensitive to color.

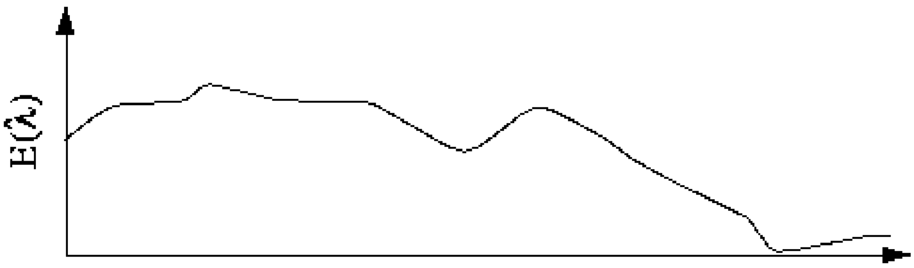
Cones and Perception

- Cones come in 3 types: red, green and blue. Each responds differently to various frequencies of light. The following figure shows the spectral-response functions of the cones and the luminous-efficiency function of the human eye.



- The color signal to the brain comes from the response of the 3 cones to the spectra being observed. That is, the signal consists of 3 numbers:

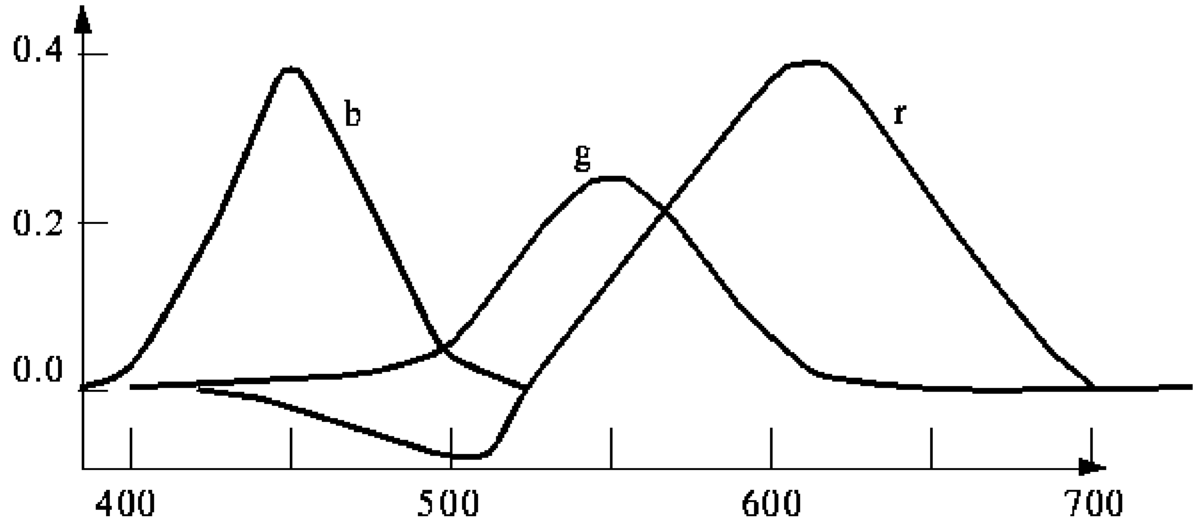
$$R = \int E(\lambda)\, S_R(\lambda)\, d\lambda$$
$$G = \int E(\lambda)\, S_G(\lambda)\, d\lambda$$
$$B = \int E(\lambda)\, S_B(\lambda)\, d\lambda$$



where *E* is the light and *S* are the sensitivity functions.

- A color can be specified as the sum of three colors. So colors form a 3 dimensional vector space.

- The following figure shows the amounts of three primaries needed to match all the wavelengths of the visible spectrum.

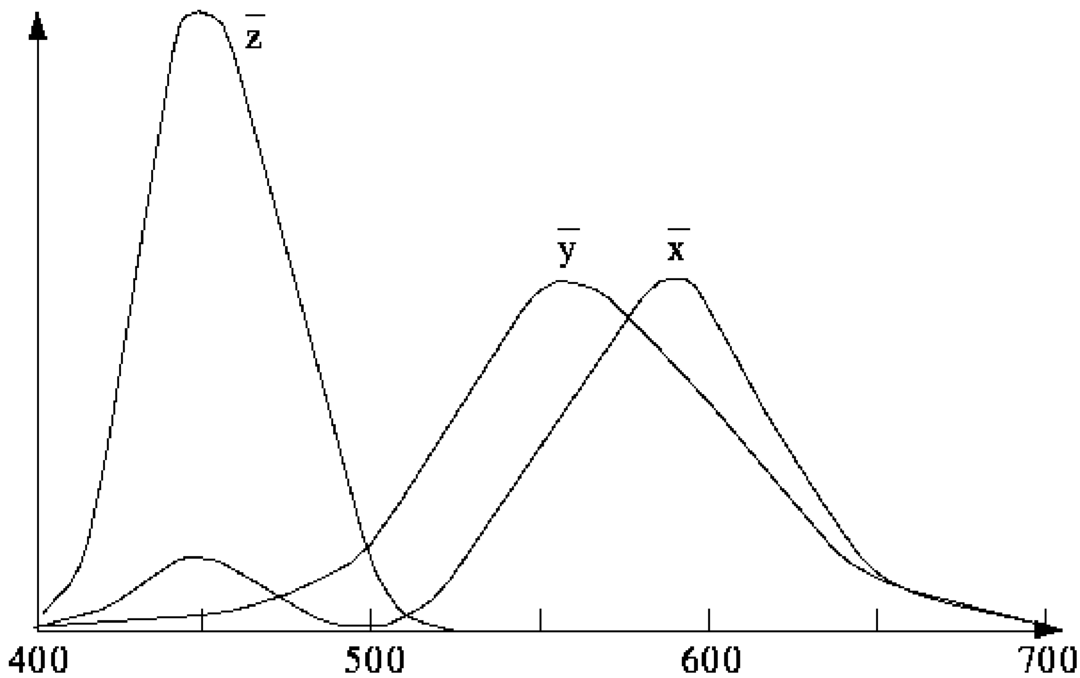


- The negative value indicates that some colors cannot be exactly produced by adding up the primaries.

CIE Chromaticity Diagram

- Q: Does a set of primaries exist that span the space with only positive coefficients?
- A: Yes, but no pure colors.

  In 1931, the CIE defined three standard primaries **(X, Y, Z)**. The **Y** primary was intentionally chosen to be identical to the luminous-efficiency function of human eyes.



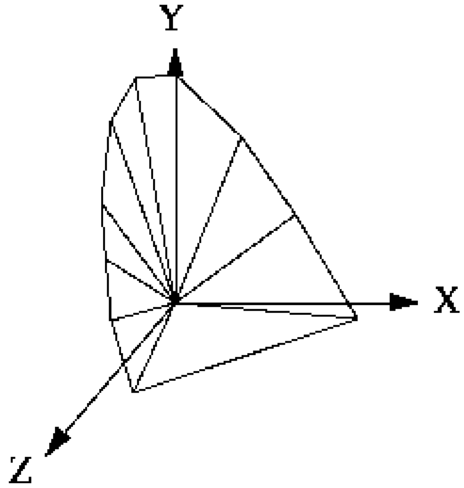
- The above figure shows the amounts of X, Y, Z needed to exactly reproduce any visible color.

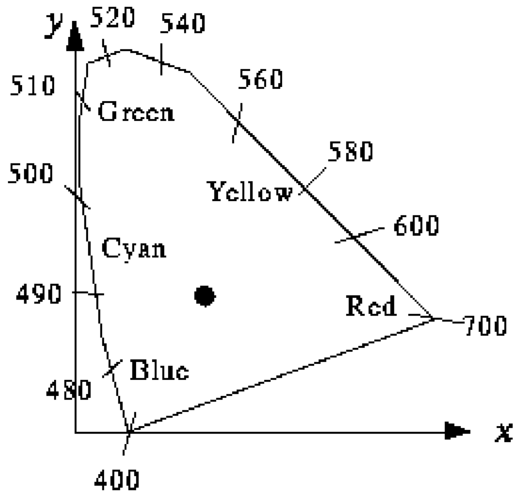
$$X = \int E(\lambda)\, \bar{x}(\lambda)\, d\lambda$$

$$Y = \int E(\lambda)\,\bar{y}(\lambda)\,d\lambda$$

$$Z = \int E(\lambda)\,\bar{z}(\lambda)\,d\lambda$$

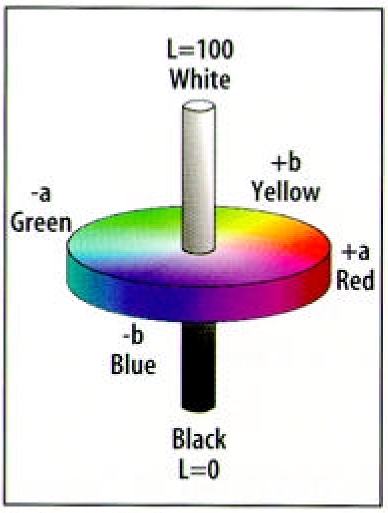
- All visible colors are in a "horseshoe" shaped cone in the X-Y-Z space. Consider the plane $X+Y+Z=1$ and project it onto the X-Y plane, we get the *CIE chromaticity diagram* as below.





- The edges represent the "pure" colors (sine waves at the appropriate frequency)
- White (a blackbody radiating at 6447 kelvin) is at the "dot"
- When added, any two colors (points on the CIE diagram) produce a point on the line between them.
- Q: how can we find a color's complement on the CIE diagram?

L*a*b (Lab) Color Model



Lab model

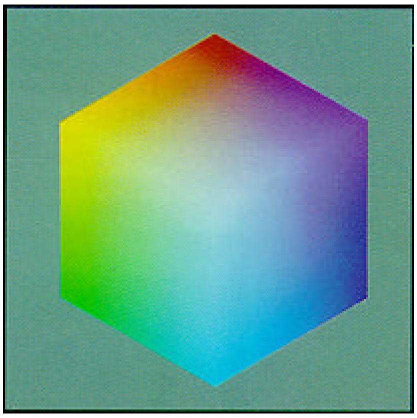
- A refined CIE model, named CIE L*a*b in 1976
- Luminance: L
  Chrominance: a -- ranges from green to red, b -- ranges from blue to yellow
- Used by *Photoshop*

3.3.2. Color Models in Images

- A color image is a 2-D array of (R,G,B) integer triplets. These triplets encode how much the corresponding phosphor should be excited in devices such as a monitor.

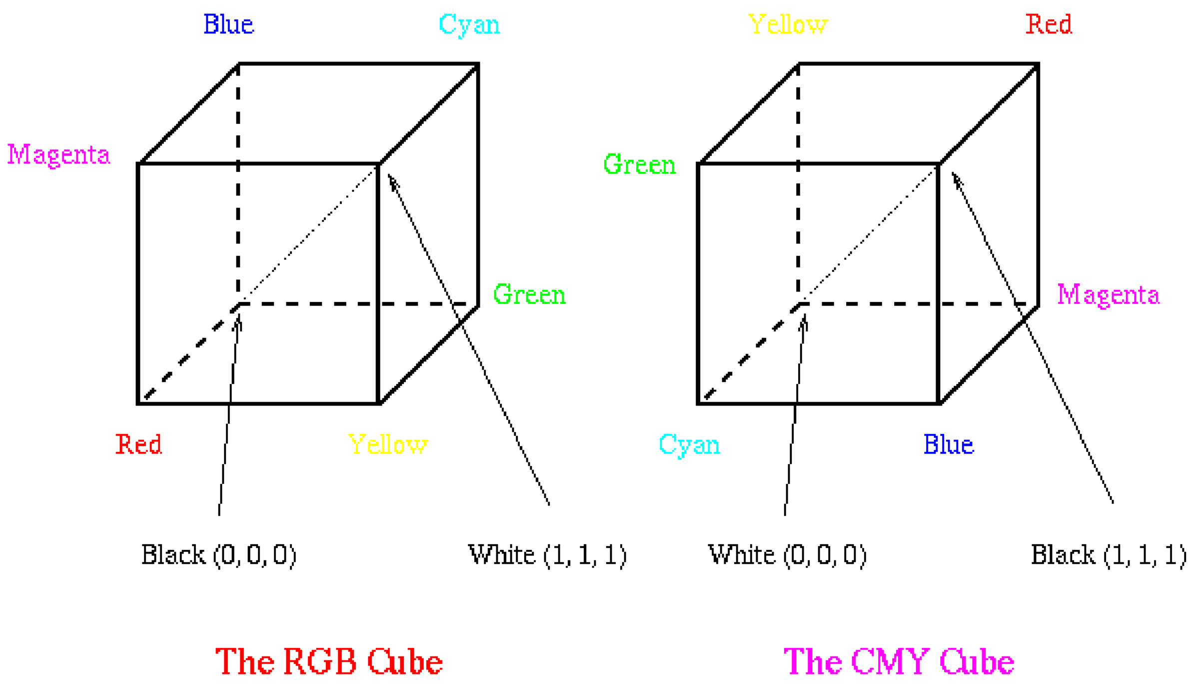
RGB Color Model for CRT Displays

- CRT displays have three phosphors (RGB) which produce a combination of wavelengths when excited with electrons.



CMY Color Model

- Cyan, Magenta, and Yellow (CMY) are complementary colors of RGB. They can be used as *Subtractive Primaries.*
- CMY model is mostly used in printing devices where the color pigments on the paper absorb certain colors (e.g., no red light reflected from cyan ink).



The RGB and CMY Cubes

Conversion between RGB and CMY:

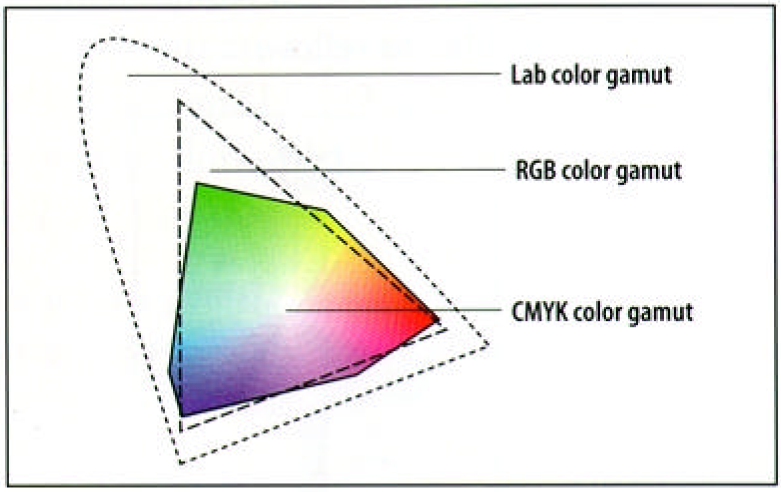
-- e.g., convert **White** from (1, 1, 1) in RGB to (0, 0, 0) in CMY.

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$

- Sometimes, an alternative CMYK model (K stands for *Black*) is used in color printing (e.g., to produce darker black than simply mixing CMY).
  - K := min (C, M, Y), C := C - K, M := M - K, Y := Y - K.

Comparison of Three Color Gamuts



- The *gamut* of colors is all colors that can be reproduced using the three primaries
- The Lab gamut covers all colors in visible spectrum
- The RGB gamut is smaller, hence certain visible colors (e.g. pure yellow, pure cyan) cannot be seen on monitors
- The CMYK gamut is the smallest (but not a straight subset of the RGB gamut)

3.3.3. Color Models in Video

- YIQ and YUV are the two commonly used color models in video

YUV Color Model

- Initially, for PAL analog video, it is now also used in CCIR 601 standard for digital video
- Y (luminance) is the CIE Y primary.

  $Y = 0.299R + 0.587G + 0.114B$

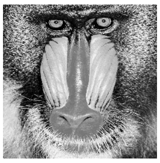
- *Chrominance* is defined as the difference between a color and a reference white at the same luminance. It can be represented by U and V -- the *color differences*.

  $U = B - Y$
  $V = R - Y$

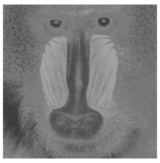
  - If b/w image, then U = V = 0. --> No chrominance!
  - ** In actual PAL implementation:

    $U = 0.492\,(B - Y)$
    $V = 0.877\,(R - Y)$

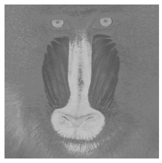
- Sample YUV Decomposition:



Y



U



- Eye is most sensitive to Y. In PAL, 5 (or 5.5) MHz is allocated to Y, 1.3 MHz to U and V.

YCbCr Color Model

- The YCbCr model is closely related to the YUV, it is a scaled and shifted YUV.

  Cb = ((B - Y)/ 2) + 0.5
  Cr = ((R - Y) / 1.6) + 0.5

- The chrominance values in YCbCr are always in the range of 0 to 1.

- YCbCr is used in JPEG and MPEG.

YIQ Color Model

- YIQ is used in NTSC color TV broadcasting, it is downward compatible with B/W TV where only Y is used.

- Although U and V nicely define the color differences, they do not align with the desired human perceptual color sensitivities. In NTSC, I and Q are used instead.

  I is the orange-blue axis, Q is the purple-green axis.
  I and Q axes are scaled and rotated R - Y and B - Y (by 33 degrees clockwise).

  I = 0.877(R - Y) cos 33 - 0.492(B - Y) sin 33
  Q = 0.877(R - Y) sin 33 + 0.492(B - Y) cos 33

  Namely,

  I = 0.74(R - Y) - 0.27(B - Y) = 0.596R - 0.275G - 0.321B
  Q = 0.48(R - Y) + 0.41(B - Y) = 0.212R - 0.523G + 0.311B

- The YIQ transform:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

- Eye is most sensitive to Y, next to I, next to Q. In NTSC broadcast TV, 4.2 MHz is allocated to Y, 1.5 MHz to I, 0.55 MHz to Q.

Summary

- Color images are encoded as triplets of values.
- RGB is an additive color model that is used for light-emitting devices, e.g., CRT displays
  CMY is a subtractive model that is used often for printers
- Two common color models in imaging are RGB and CMY, two common color models in video are YUV and YIQ.
- YUV uses properties of the human eye to prioritize information. Y is the black and white (luminance) image, U and V are the color difference (chrominance) images. YIQ uses similar idea.
- Besides the hardware-oriented color models (i.e., RGB, CMY, YUV, YIQ), HSB (Hue, Saturation, and Brightness) and HLS (Hue, Lightness, and Saturation) are also commonly used.

Further Exploration

sRGB -- a proposed Device Independent Color Space for the Internet

---

Last Updated: ***5/26/98***

Top | Chap 3 | CMPT 365 Home Page | CS

3.4. Basics of Video

Reference: Chapter 5 of Steinmetz and Nahrstedt

- Analog video is represented as a continuous (time varying) signal.
- Digital video is represented as a sequence of digital images.

3.4.1 Types of Color Video Signals

- **Component video** -- each primary is sent as a separate video signal.
  - The primaries can either be RGB or a luminance-chrominance transformation of them (e.g., YIQ, YUV).
  - Best color reproduction
  - Requires more bandwidth and good synchronization of the three components
- **Composite video** -- color (chrominance) and luminance signals are mixed into a single carrier wave. Some interference between the two signals is inevitable.
- **S-Video** (Separated video, e.g., in S-VHS) -- a compromise between component analog video and the composite video. It uses two lines, one for luminance and another for composite chrominance signal.

3.4.2 Analog Video

The following figures are from A.M. Tekalp, "Digital video processing", Prentice Hall PTR, 1995.

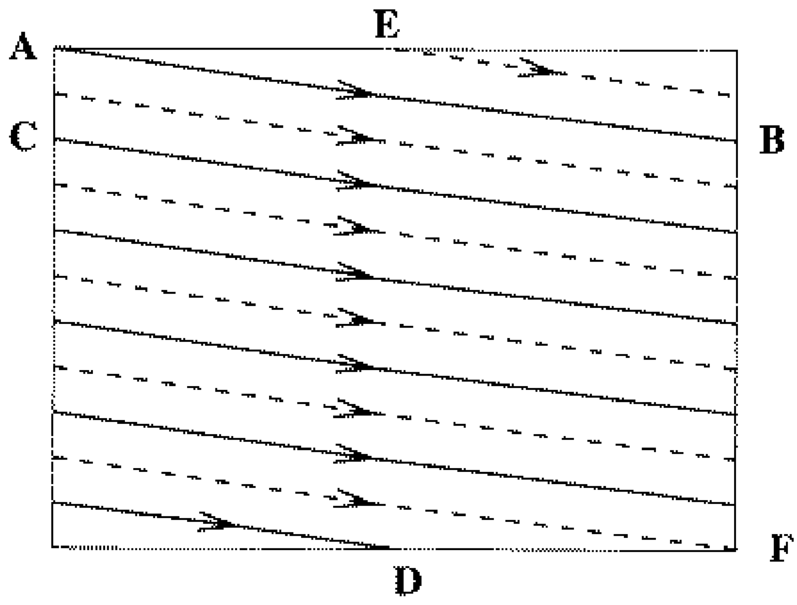


Figure 1.1: Scanning raster.

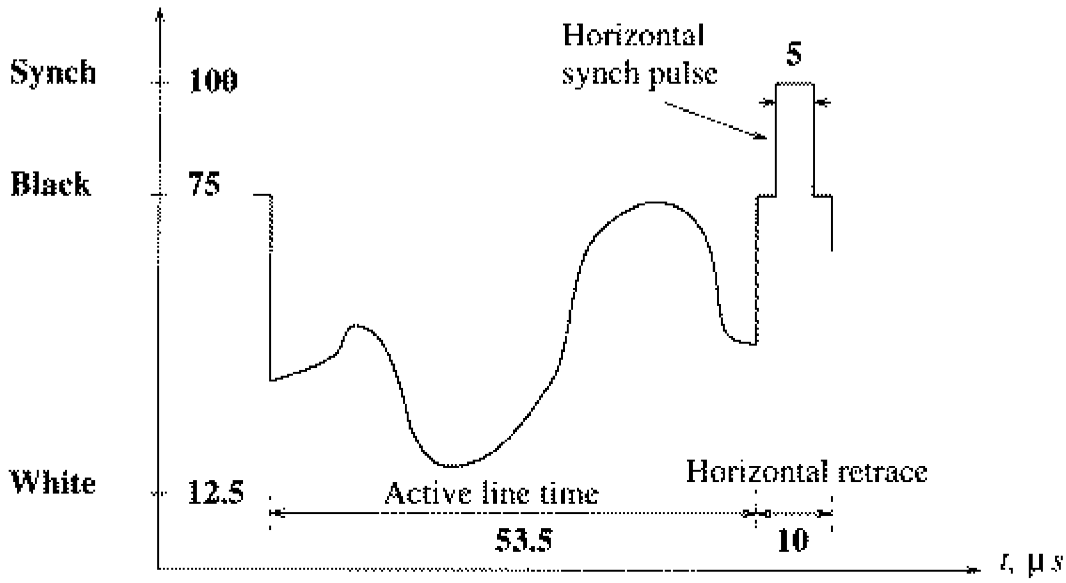


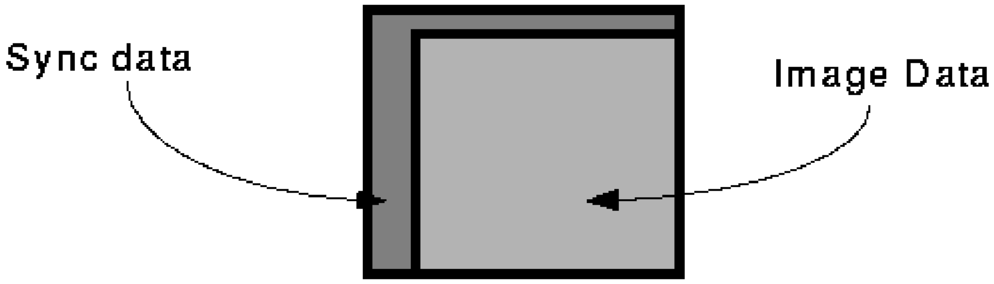
Figure 1.2: Video signal for one full line.

NTSC Video

- 525 scan lines per frame, 30 frames per second (or be exact, 29.97 fps, 33.37 msec/frame)
- Interlaced, each frame is divided into 2 fields, 262.5 lines/field
- 20 lines reserved for control information at the beginning of each field
    - So a maximum of 485 lines of visible data

- Laserdisc and S-VHS have actual resolution of ~420 lines
- Ordinary TV -- ~320 lines

- Each line takes 63.5 microseconds to scan. Horizontal retrace takes 10 microseconds (with 5 microseconds horizontal synch pulse embedded), so the active line time is 53.5 microseconds.



Digital Video Rasters

- Color representation:
  - NTSC uses YIQ color model.
  - composite = Y + I *cos*(Fsc t) + Q *sin*(Fsc t), where Fsc is the frequency of color subcarrier

PAL Video

- 625 scan lines per frame, 25 frames per second (40 msec/frame)

- Interlaced, each frame is divided into 2 fields, 312.5 lines/field

- Uses YUV color model

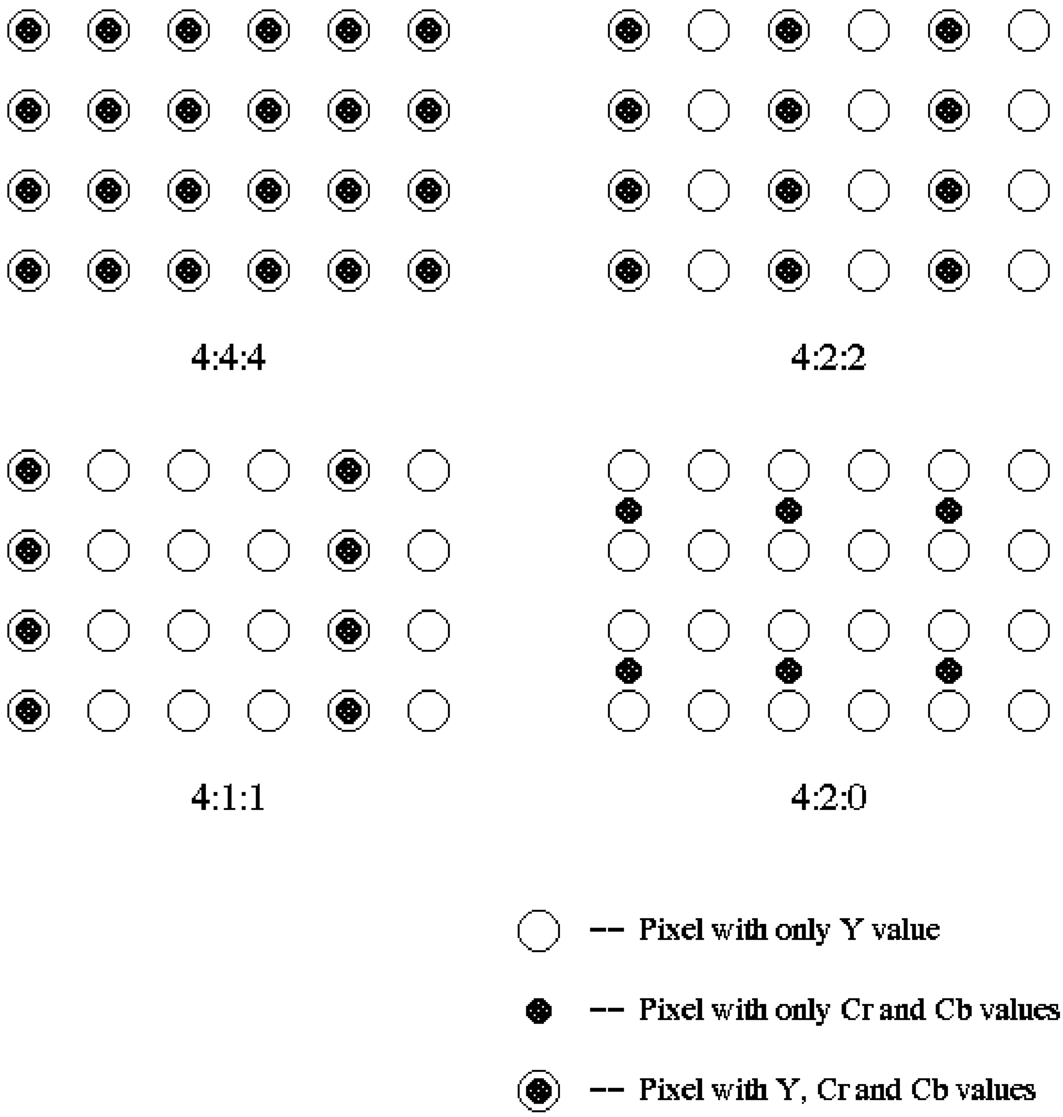
3.4.3 Digital Video

- **Advantages:**
  - Direct random access --> good for nonlinear video editing
  - No problem for repeated recording
  - No need for blanking and sync pulse

- Almost all digital video uses component video

Chroma Subsampling

- How to decimate for chrominance?



- 4:4:4 --> No chroma subsampling, each pixel has Y, Cr and Cb values.

  4:2:2 --> Horizontally subsample Cr, Cb signals by a factor of 2.

  4:1:1 --> Horizontally subsampled by a factor of 4.

  4:2:0 --> Subsampled in both the horizontal and vertical dimensions by a factor of 2. Theoretically, the chroma pixel is positioned between the rows and columns as shown in the figure.

- 4:1:1 and 4:2:0 are mostly used in JPEG and MPEG (see Chapter 4).

CCIR Standards for Digital Video

(CCIR -- Consultative Committee for International Radio)

	CCIR 601 525/60 NTSC	CCIR 601 625/50 PAL/SECAM	CIF	QCIF
Luminance resolution	720 x 485	720 x 576	352 x 288	176 x 144

Luminance resolution	720 x 485	720 x 576	352 x 288	176 x 144
Chrominance resolut.	360 x 485	360 x 576	176 x 144	88 x 72
Color Subsampling	4:2:2	4:2:2	4:2:0	4:2:0
Fields/sec	60	50	30	30
Interlacing	Yes	Yes	No	No

- CCIR 601 uses interlaced scan, so each field only has half as much vertical resolution (e.g., 243 lines in NTSC). The CCIR 601 (NTSC) data rate is ~165 Mbps.
- CIF (Common Intermediate Format) -- an acceptable temporary standard
  - Approximately the VHS quality
  - Uses progressive (non-interlaced) scan
  - Uses NTSC frame rate, and half the active lines of PAL signals --> To play on existing TVs, PAL systems need to do frame rate conversion, and NTSC systems need to do line-number conversion.
- QCIF -- Quarter-CIF

ATSC Digital Television Standard

(ATSC -- Advanced Television Systems Committee) The **ATSC Digital Television Standard** was recommended to be adopted as the Advanced TV broadcasting standard by the FCC Advisory Committee on Advanced Television Service on November 28, 1995. It covers the standard for **HDTV** (High Definition TV).

Video Format

The video scanning formats supported by the ATSC Digital Television Standard are shown in the following table.

Vertical Lines	Horizontal Pixels	Aspect Ratio	Picture Rate
1080	1920	16:9	60I 30P 24P
720	1280	16:9	60P 30P 24P
480	704	16:9 & 4:3	60I 60P 30P 24P
480	640	4:3	60I 60P 30P 24P

- The aspect ratio for HDTV is 16:9 as opposed to 4:3 in NTSC, PAL, and SECAM. (A 33% increase in horizontal dimension.)
- In the picture rate column, the "I" means interlaced scan, and the "P" means progressive (non-interlaced) scan.
- Both NTSC rates and integer rates are supported (i.e., 60.00, 59.94, 30.00, 29.97, 24.00, and 23.98).
- At 1920 x 1080, 60I (which CBS and NBC have selected), there will be 1920 x 1080 x 30 = 62.2 millions pixels per second. Considering 4:2:2 chroma subsampling, each pixel needs 16 bits to represent, the bit rate is 62.2 x 16 = 995 Mb/sec.

Further Exploration

Digital TV