CMPT 365 Multimedia Systems

Mid-Term Review

Xiaochuan Chen
Spring 2017

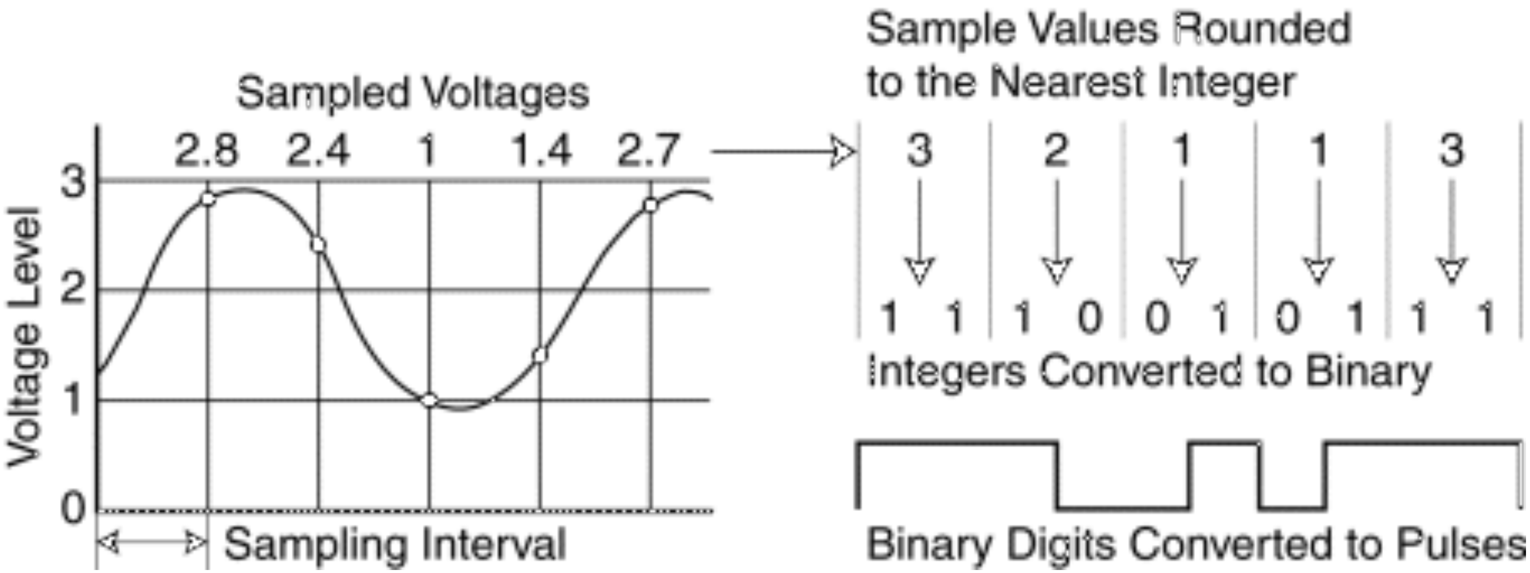
Adminstrative

- Mid-Term: Feb 22th, In Class, 50mins
- Still have a course on Monday Feb 20th!!!
- Pick up assignment: Today 4:30~5:30 with TA
- A2 will be released

Outline

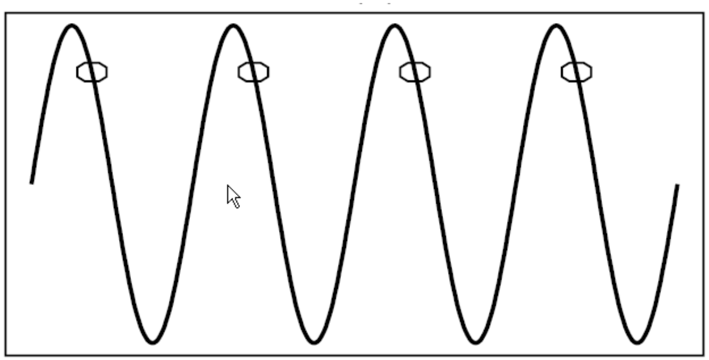
- Media Representation - Audio
- Media Representation - Image
- Media Representation - Video
- Lossless Compression

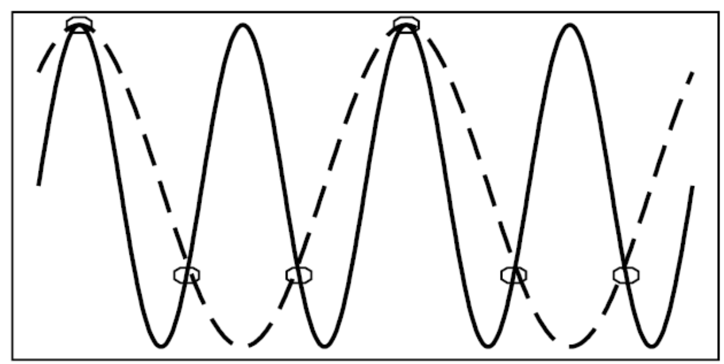
Quantization and Sampling



Sampling Rate cont'd

- For correct sampling we must use a sampling rate equal to at least *twice the maximum frequency* content in the signal. This rate is called the **Nyquist rate**.
- The relationship among the Sampling Frequency, True Frequency, and the Alias Frequency is as follows:
  - $f_{alias} = f_{sampling} - f_{true}$, **for** $f_{true} < f_{sampling} < 2 \times f_{true}$



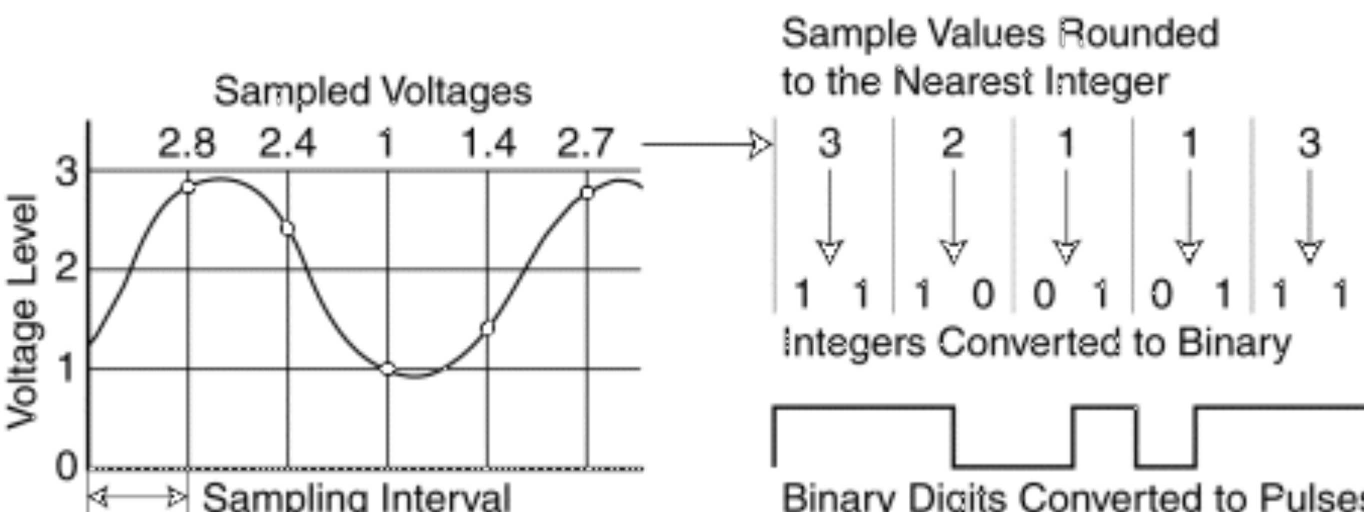


Sampling Rate cont'd

- **Nyquist frequency**: half of the Sampling rate
  - Since it would be impossible to recover frequencies higher than Nyquist frequency in any event, most systems have an **antialiasing filter** that restricts the frequency content in the input to the sampler to a range at or below Nyquist frequency.
- Sampling theory – Nyquist theorem
  - If a signal is **band-limited**, i.e., there is a lower limit $f1$ and an upper limit $f2$ of frequency components in the signal, then the sampling rate should be at least $2(f2 - f1)$.

Quantization Noise

- **Quantization noise**: the difference between the actual value of the analog signal, for the particular sampling time, and the nearest quantization interval value.
  - At most, this error can be as much as half of the interval.
- The quality of the quantization is characterized by the Signal to Quantization Noise Ratio (**SQNR**).



Signal to Noise Ratio (SNR)

- *Signal to Noise Ratio* (**SNR**): the ratio of the power of the correct signal and the noise
  - A common measure of the quality of the signal
  - The ratio can be huge and often non-linear
- So practically, SNR is usually measured in log-scale: decibels (**dB**), where 1 dB is 1/10 **Bel**. The SNR value, in units of dB, is defined in terms of base-10 logarithms of squared voltages, as follows:

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

Common sound levels

Table 6.1: Magnitude levels of common sounds, in decibels

Threshold of hearing	0
Rustle of leaves	10
Very quiet room	20
Average room	40
Conversation	60
Busy street	70
Loud radio	80
Train through station	90
Riveter	100
Threshold of discomfort	120
Threshold of pain	140
Damage to ear drum	160

Signal-to-Quantization Noise Ratio (SQNR) cont'd

- For a quantization accuracy of N bits per sample, the peak SQNR can be simply expressed:

$$SQNR = 20\log_{10}\frac{V_{signal}}{V_{quan\_noise}} = 20\log_{10}\frac{2^{N-1}}{\frac{1}{2}}$$

$$= 20 \times N \times \log 2 = 6.02\,N(\text{dB}) \qquad (6.3)$$

- $6.02N$ is the worst case.

Note: We map the maximum signal to $2^{N-1} - 1$ ($\simeq 2^{N-1}$) and the most negative signal to -2^{N-1}.

Dynamic range : the ratio of maximum to minimum absolute values of the signal: V_{max}/V_{min}. The max abs. value V_{max} gets mapped to $2^{N-1} - 1$; the min abs. value V_{min} gets mapped to 1. V_{min} is the smallest positive voltage that is not masked by noise. The most negative signal, $-V_{max}$, is mapped to -2^{N-1}.

Linear and Non-linear Quantization

- **Linear format**: samples are typically stored as uniformly quantized values.
- **Non-uniform quantization**: set up more finely-spaced levels where humans hear with the most acuity.
  - Weber's Law stated formally says that equally perceived differences have values proportional to absolute levels:

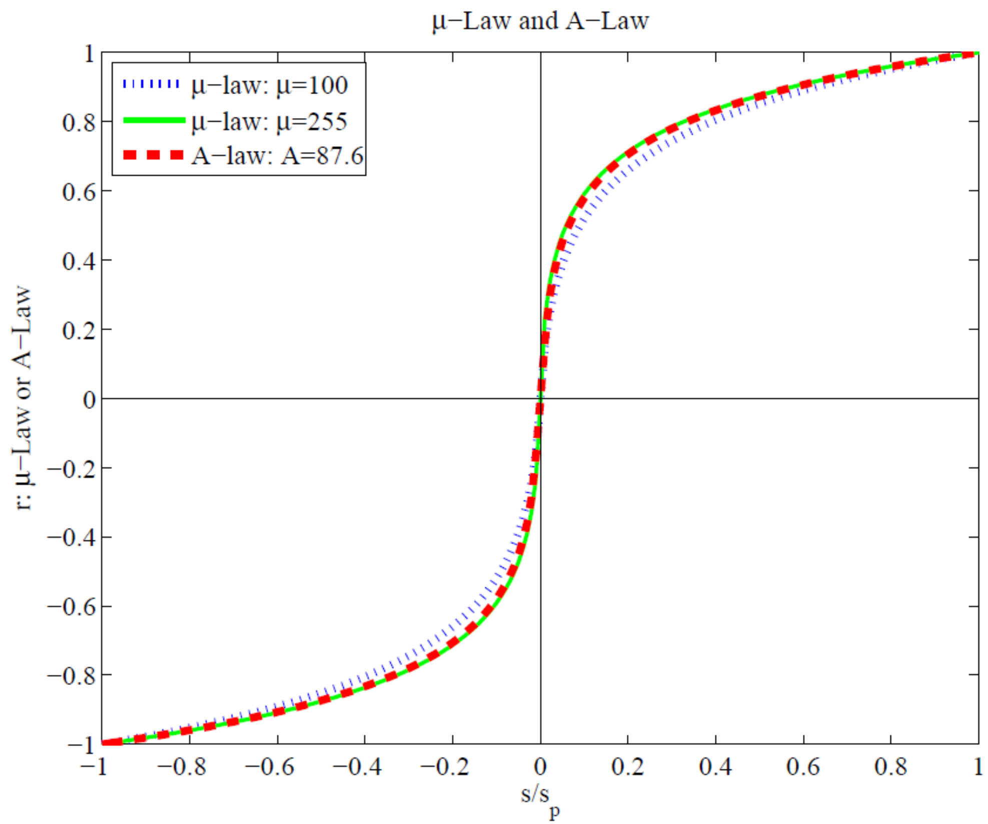
$$\Delta\text{Response} \propto \Delta\text{Stimulus}/\text{Stimulus} \qquad (6.5)$$

  - Inserting a constant of proportionality k, we have a differential equation that states:

$$dr = k\,(1/s)\,ds \qquad (6.6)$$

with response r and stimulus s.

Linear and Non-linear Quantization



- **Fig. 6.6**: Nonlinear transform for audio signals.

The parameter μ is set to $\mu = 100$ or $\mu = 255$; the parameter A for the A-law encoder is usually set to $A = 87.6$.

- The μ-law in audio is used to develop a nonuniform quantization rule for sound: uniform quantization of r gives finer resolution in s at the quiet end.

MIDI: Musical Instrument Digital Interface

- Use the sound card's defaults for sounds: ⇒ use a simple scripting language and hardware setup called **MIDI**.

- **MIDI Overview**
  - MIDI is a scripting language — it codes "events" that stand for the production of sounds. E.g., a MIDI event might include values for the pitch of a single note, its duration, and its volume.

MIDI Concepts

- MIDI **channels** are used to separate messages.

  (a) There are 16 channels numbered from 0 to 15. The channel forms the last 4 bits (the least significant bits) of the message.

  (b) Usually a channel is associated with a particular instrument: e.g., channel 1 is the piano, channel 10 is the drums, etc.

  (c) Nevertheless, one can switch instruments midstream, if desired, and associate another instrument with any channel.

MIDI Terminology

- Synthesizer:
  - was, and still can be, a stand-alone sound generator that can vary pitch, loudness, and tone color.
  - Units that generate sound are referred to as *tone modules* or sound modules.
- Sequencer:
  - started off as a special hardware device for storing and editing a *sequence* of musical events, in the form of MIDI data.
  - Now it is more often a software *music editor* on the computer.
- MIDI Keyboard:
  - produces no sound, instead generating sequences of MIDI in- structions, called *MIDI messages*
  - MIDI messages are rather like assembler code and usually consist of just a few bytes

6.2.2 Hardware Aspects of MIDI

- The MIDI hardware setup consists of a 31.25 kbps serial connection. Usually, MIDI-capable units are either Input devices or Output devices, not both.
- A traditional synthesizer is shown in Fig. 6.11:

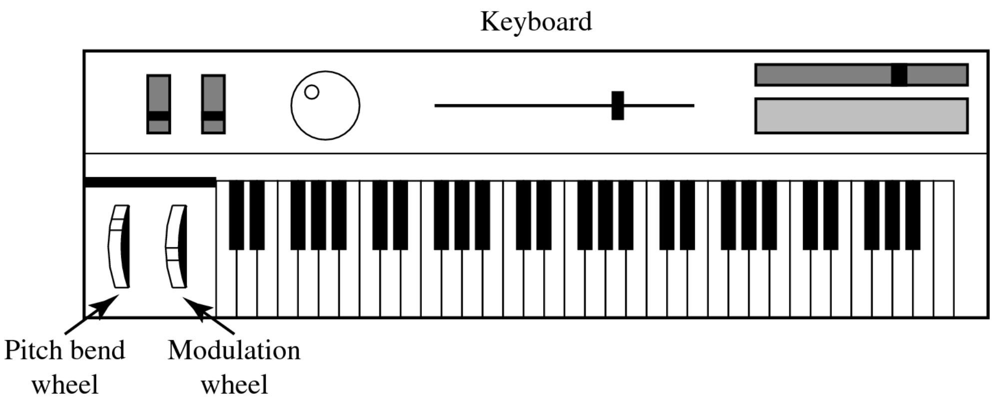


Fig. 6.11: A MIDI synthesizer

- The physical MIDI ports consist of 5-pin connectors for IN and OUT, as well as a third connector called THRU.
  a) MIDI communication is half-duplex.
  b) MIDI IN is the connector via which the device receives all MIDI data.
  c) MIDI OUT is the connector through which the device transmits all the MIDI data it generates itself.
  d) MIDI THRU is the connector by which the device echoes the data it receives from MIDI IN. Note that it is only the MIDI IN data that is echoed by MIDI THRU — all the data generated by the device itself is sent via MIDI OUT.

- A typical MIDI sequencer setup is shown in Fig. 6.12:

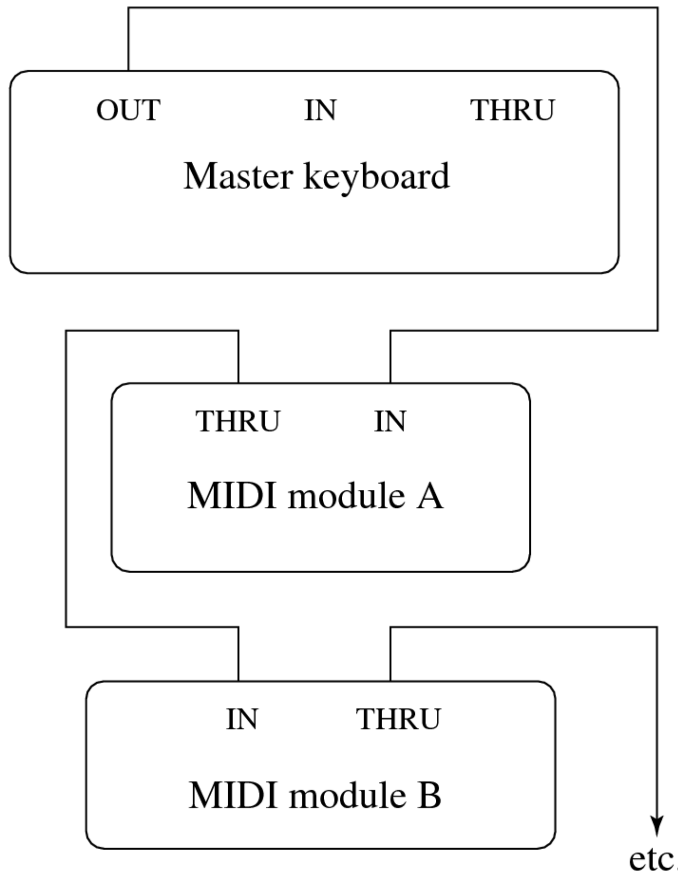


Fig. 6.12: A typical MIDI setup

❐ Table 6.3: MIDI voice messages

Voice Message	Status Byte	Data Byte1	Data Byte2
Note Off	&H8n	Key number	Note Off velocity
Note On	&H9n	Key number	Note On velocity
Poly. Key Pressure	&HAn	Key number	Amount
Control Change	&HBn	Controller num.	Controller value
Program Change	&HCn	Program number	None
Channel Pressure	&HDn	Pressure value	None
Pitch Bend	&HEn	MSB	LSB

- (** &H indicates hexadecimal, and 'n' in the status byte hex value stands for a channel number. All values are in 0..127 except Controller number, which is in 0..120)

Outline

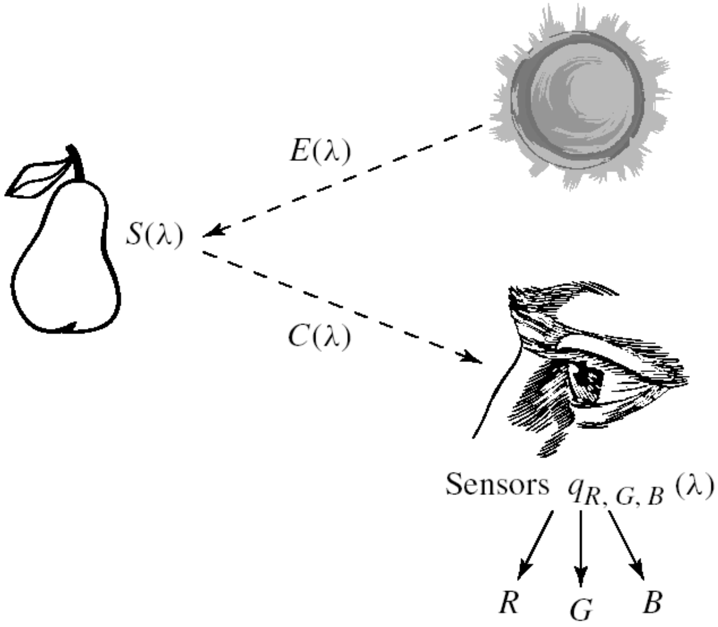
- Media Representation - Audio
- Media Representation - Image
- Media Representation - Video
- Lossless Compression

Color Formation

$$R = \int E(\lambda)\, S(\lambda)\, q_R(\lambda)\, d\lambda$$

$$G = \int E(\lambda)\, S(\lambda)\, q_G(\lambda)\, d\lambda$$

$$B = \int E(\lambda)\, S(\lambda)\, q_B(\lambda)\, d\lambda$$



4.1.6 Gamma Correction

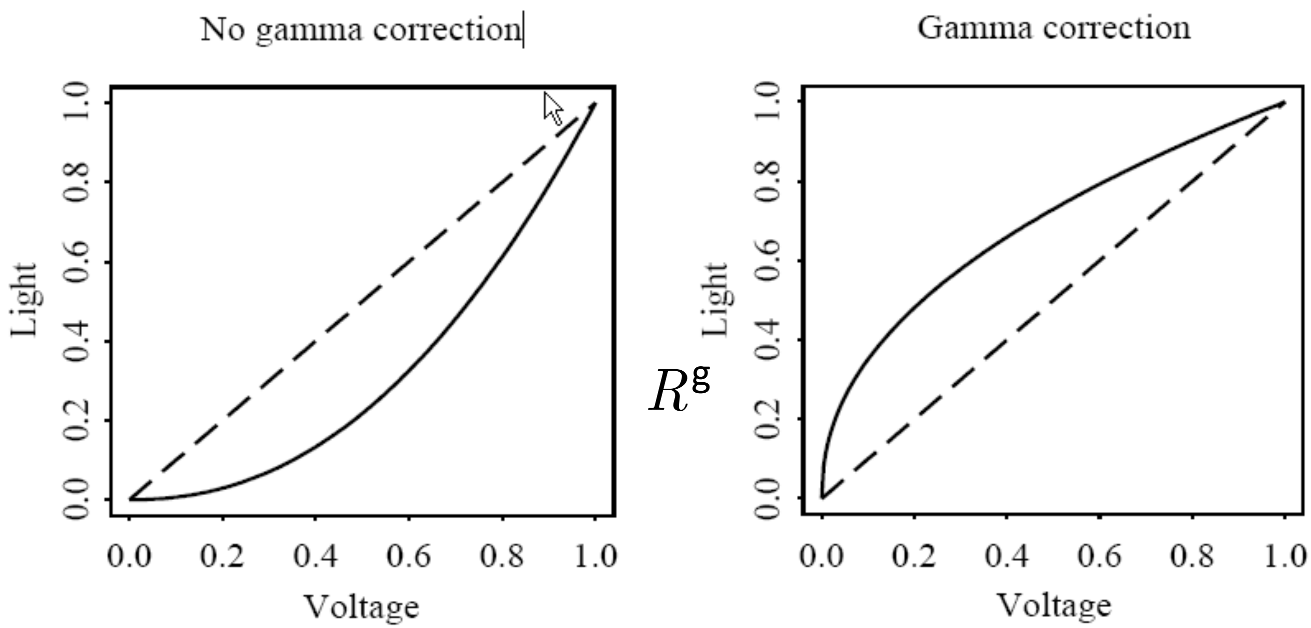
- The light emitted is in fact roughly proportional to the voltage *raised to a power*; this power is called **gamma**, with symbol γ.

(a) Thus, if the file value in the red channel is *R*, the screen emits light proportional to R^{γ}, with SPD equal to that of the red phosphor paint on the screen that is the target of the red channel electron gun. The value of gamma is around 2.2.

(b) It is customary to append a prime to signals that are **gamma-corrected** by raising to the power $(1/\gamma)$ before transmission. Thus we arrive at **linear signals**:

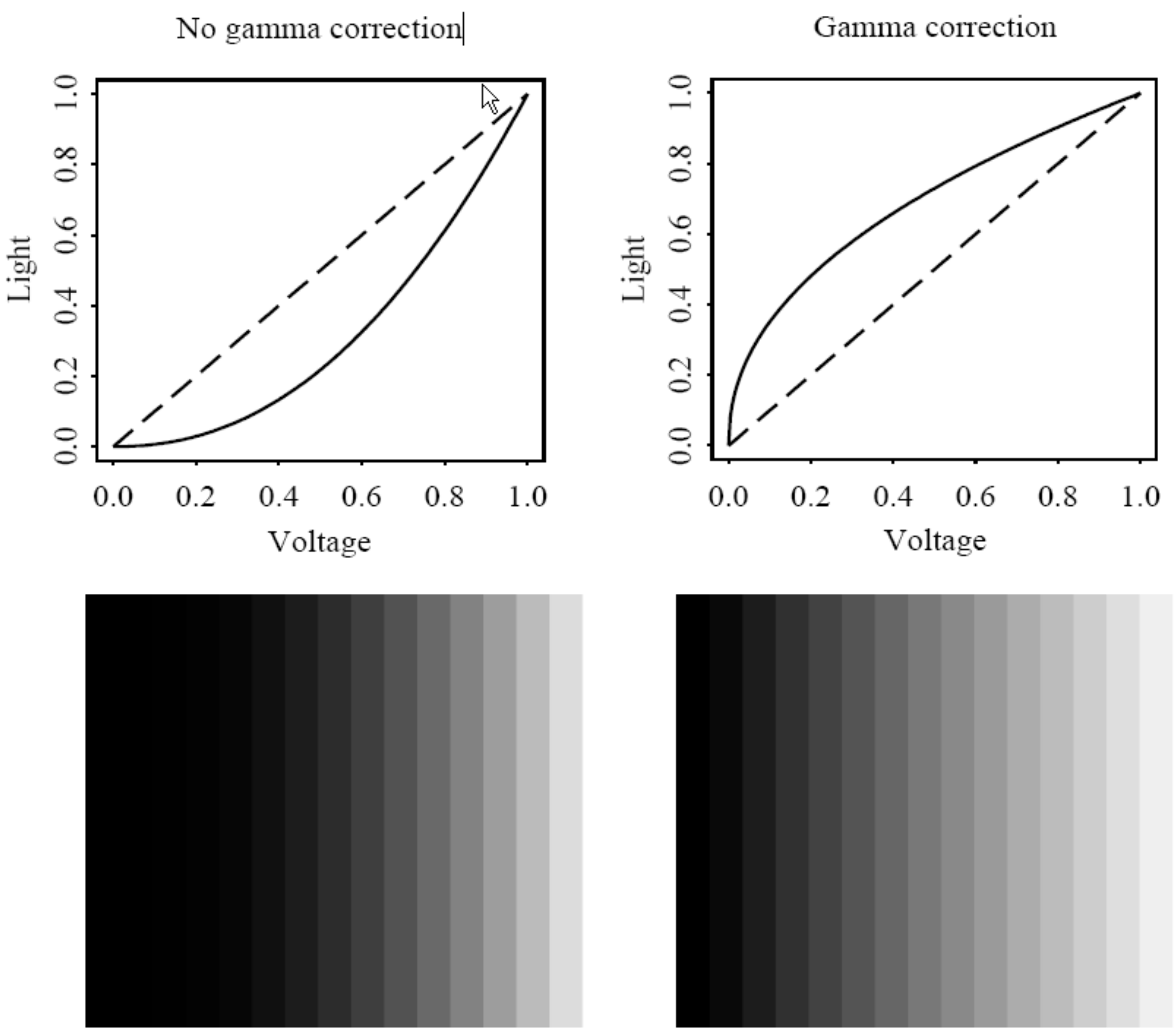
$$R \rightarrow R' = R^{1/\gamma} \Rightarrow (R')^{\gamma} \rightarrow R$$

Gamma Correction cont'd



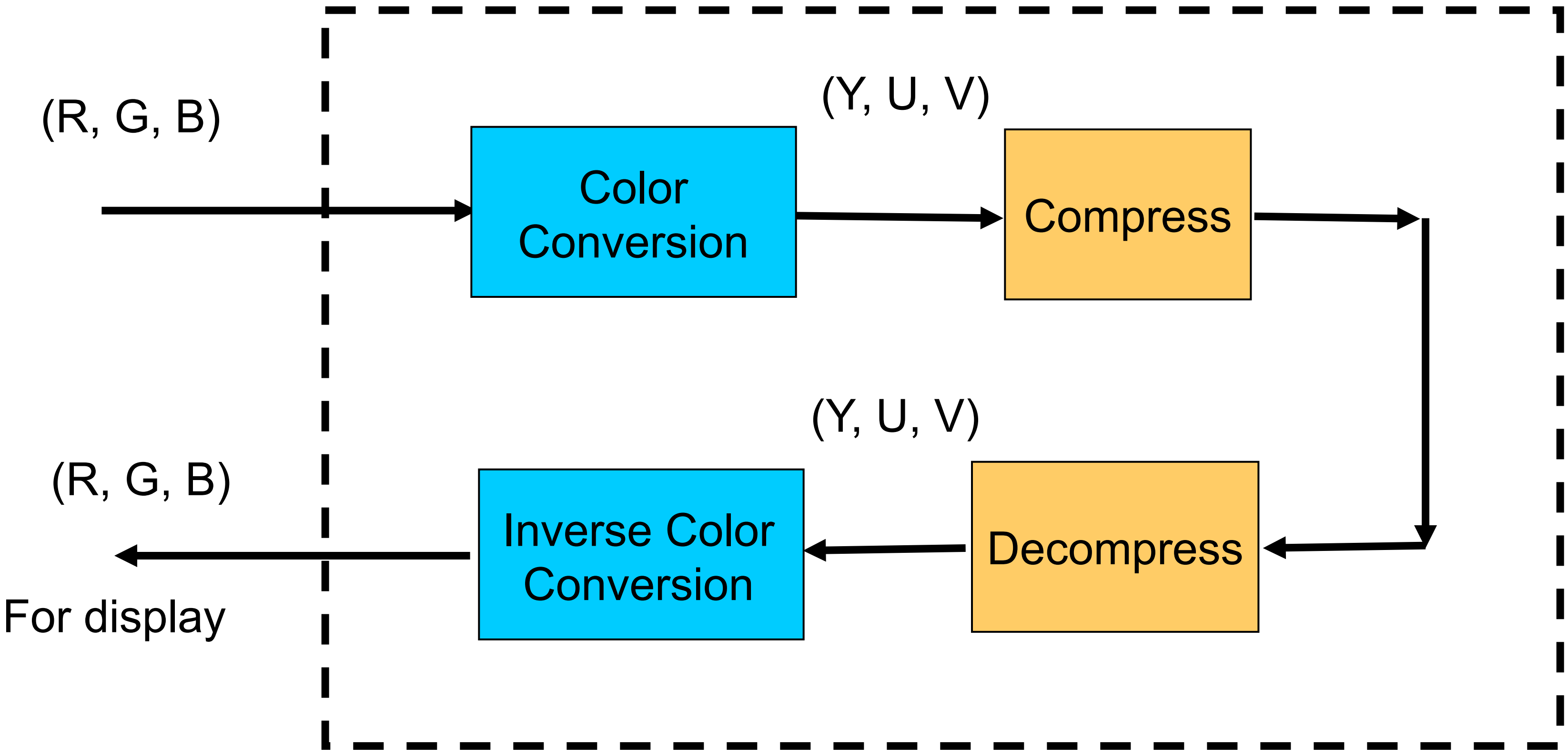
- Left: light output from CRT with no gamma-correction applied. -- Darker values are displayed too dark.
- Right: pre-correcting signals by applying the power law $R^{1/g}$
- Normalization (0-1) ?

Gamma Correction cont'd



Color Space: RGB→YUV

- Solution: convert to other spaces
- Why ? Display device, compression …



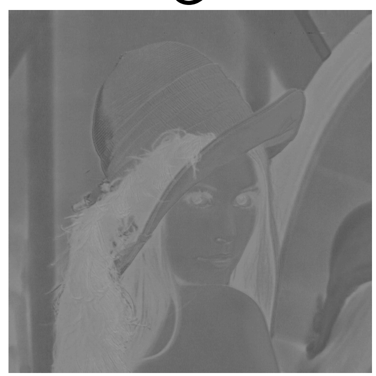
Color Space

R

G

B

Y



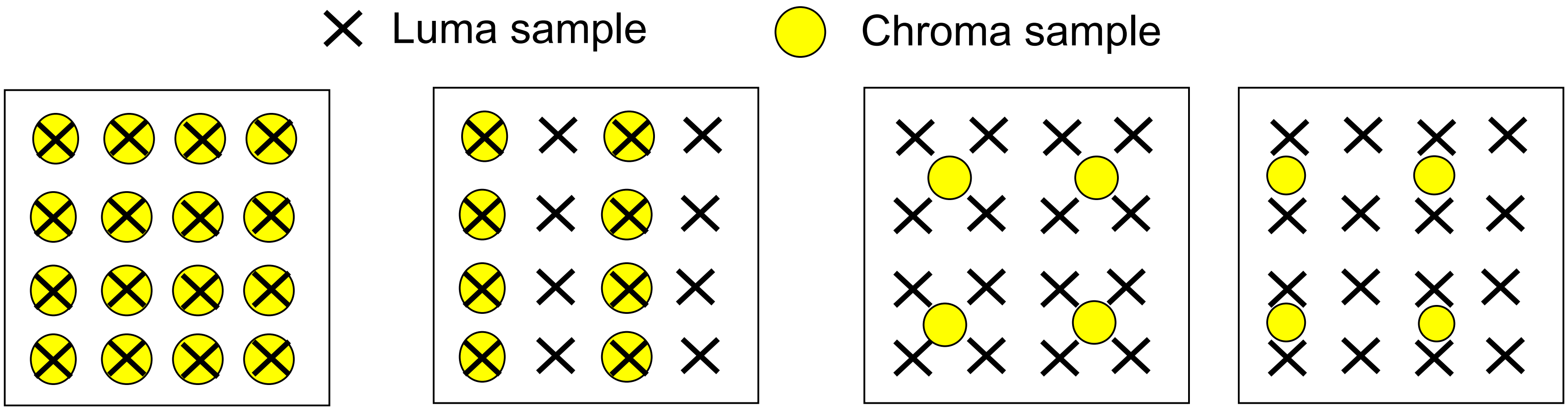
Cb

Cr

- Most information is in Y channel (brightness)
  - Cb and Cr are small → easier for compression
- Human eyes are not sensitive to color error
  - Don't need high resolution for color component

Color Space: Down-sampling

- Down-sampling color components to improve compression



MPEG-1 MPEG-2

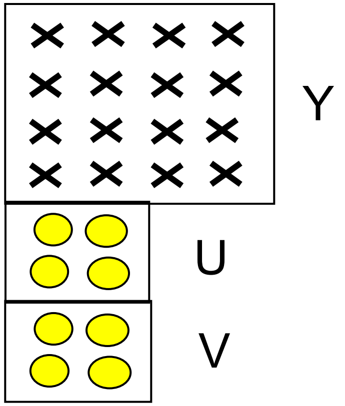
YUV 4:4:4
No downsampling
Of Chroma

YUV 4:2:2
- 2:1 horizontal downsampling of chroma components
- 2 chroma samples for every 4 luma samples

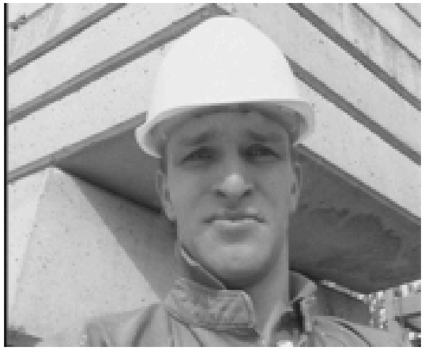
YUV 4:2:0
- 2:1 horizontal downsampling of chroma components
- 1 chroma sample for every 4 luma samples
- Widely used

Raw YUV Data File Format

- In YUV 4:2:0, number of U and V samples are 1/4 of the Y samples
- YUV samples are stored separately:
  Image: YYYY…..Y UU…U VV…V
  (row by row in each channel)
  Video: YUV of frame 1, YUV of frame 2, ……



- CIF (Common Intermediate format):
  - 352 x 288 pixels for Y, 176 x 144 pixels for U, V
- QCIF (Quarter CIF): 176 x 144 pixels for Y, 88 x 72 pixels for U, V
- CIF, and QCIF formats are widely used for video conference



Y: 176 x 144

U: 88 x 72

V: 88 x 72

Sample Matlab code: readyuv('foreman.qcif',176, 144, 1, 1);

Dithering

- **Rationale:** calculate square patterns of dots such that values from 0 to 255 correspond to patterns that are more and more filled at darker pixel values, for printing on a 1-bit printer.

- **Strategy:** Replace a pixel value by a larger pattern, say 2x 2 or 4 x 4, such that the number of printed dots approximates the varying-sized disks of ink used in analog, in **halftone printing** (e.g., for newspaper photos).

1. Half-tone printing is an analog process that uses smaller or larger filled circles of black ink to represent shading, for newspaper printing.
2. For example, if we use a 2 x 2 **dither matrix**

$$\begin{pmatrix} 0 & 2 \\ 3 & 1 \end{pmatrix}$$

Dithering cont'd

we can first re-map image values in 0..255 into the new range 0..4 by (integer) dividing by 256/5. Then, e.g., if the pixel value is 0 we print nothing, in a 2 x 2 area of printer output. But if the pixel value is 4 we print all four dots.

- The rule is:

  If the intensity is > the dither matrix entry then print an `on` dot at that entry location: replace each pixel by an *n* x *n* matrix of dots.

- Note that the image size may be much larger, for a dithered image, since replacing each pixel by a 4 x 4 array of dots, makes an image 16 times as large.

Ordered Dithering

- A clever trick can get around this problem. Suppose we wish to use a larger, 4 x 4 dither matrix, such as

$$\begin{pmatrix} 0 & 8 & 2 & 10 \\ 12 & 4 & 14 & 6 \\ 3 & 11 & 1 & 9 \\ 15 & 7 & 13 & 5 \end{pmatrix}$$

- An **ordered dither** consists of turning on the printer out-put bit for a pixel if the intensity level is greater than the particular matrix element just at that pixel position.

- Fig. 4 (a) shows a grayscale image of “Lena”. The ordered-dither version is shown as Fig. 4 (b), with a detail of Lena's right eye in Fig. 4 (c).

Dithering cont'd

- Algorithm for ordered dither, with *n x n* dither matrix, is as follows:

```
BEGIN
    for x = 0 to x_max                  // columns
        for y = 0 to y_max              // rows
            i = x mod n
            j = y mod n
            // I(x, y) is the input, O(x, y) is the output,
            //D is the dither matrix.
            if I(x, y) > D(i, j)
                O(x, y) = 1;
            else
                O(x, y) = 0;
END
```

Popular File Formats

- **8-bit GIF** : one of the most important formats because of its historical connection to the WWW and HTML markup language as the first image type recognized by net browsers.

- **JPEG**: currently the most important common file format.

Outline

- Media Representation - Audio
- Media Representation - Image
- Media Representation - Video
- Lossless Compression

Analog Video

- An analog signal *f*(*t*) samples a time-varying image
- Progressive scanning
  - traces through a complete picture (a frame) row-wise for each time interval.
- Interlaced scanning
  - Odd-numbered lines traced first, and then the even-numbered lines.
  - "odd" and "even" fields - two fields make up one frame
  - Widely used in traditional (non-digital) TV

NTSC Video

- **NTSC** (National Television System Committee) TV standard is mostly used in North America and Japan
  - YIQ color model
  - 4:3 **aspect ratio** (i.e., the ratio of picture width to its height)
  - 525 scan lines per frame at 30 frames per second (fps).
- Interlaced scanning, and each frame is divided into two fields, with 262.5 lines/field
  - horizontal sweep frequency is 525x29.97 = 15,734 lines/sec,
  - each line is swept out in 1/15,734 = 63.6 us
  - the horizontal retrace takes 10.9 sec, this leaves 52.7 sec for the active line signal during which image data is displayed

- PAL in Asia/Europe, SECAM in Europe
- All faded out (Canada, Aug 31, 2011)

Digital Video

- Why digital video ?
- Advantages
  - Stored on digital device or in memory
  - Faithful duplication in digital domain
    - Good or bad ?
  - Direct (random) access,
    - nonlinear video editing achievable as a simple, rather than a complex task
  - Ease of manipulation (noise removal, cut and paste, etc.)
  - Ease of encryption and better tolerance to channel noise
    - Multimedia communications
  - Integration to various multimedia applications

Analog Video Display Interfaces

Component video, Composite video, S-video, VGA

Entropy

- Suppose:
  - a data source generates output sequence from a set $\{A_1, A_2, \ldots, A_N\}$
  - $P(A_i)$: Probability of A_i
- First-Order Entropy (or simply Entropy):
  - the average self-information of the data set

$$H = \sum_i -P(A_i)\log_2 P(A_i)$$

- The first-order entropy represents the minimal number of bits needed to losslessly represent one output of the source.

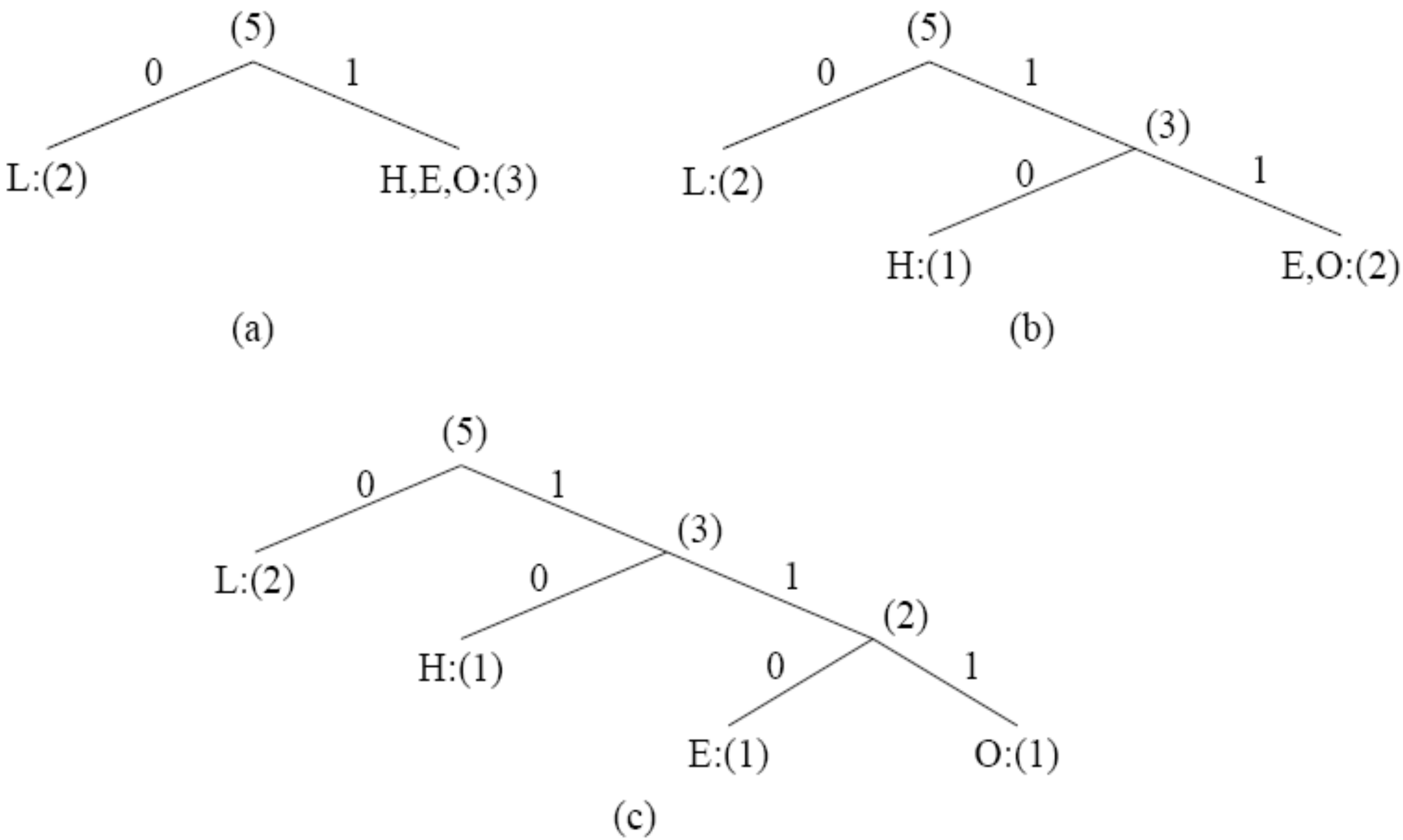
Shannon-Fano Coding

- **Shannon-Fano Algorithm** - a top-down approach
  - Sort the symbols according to the frequency count of their occurrences.
  - Recursively divide the symbols into two parts, each with approximately the same number of counts, until all parts contain only one symbol.
- **Example: coding of "HELLO"**

Symbol	H	E	L	O
Count	1	1	2	1

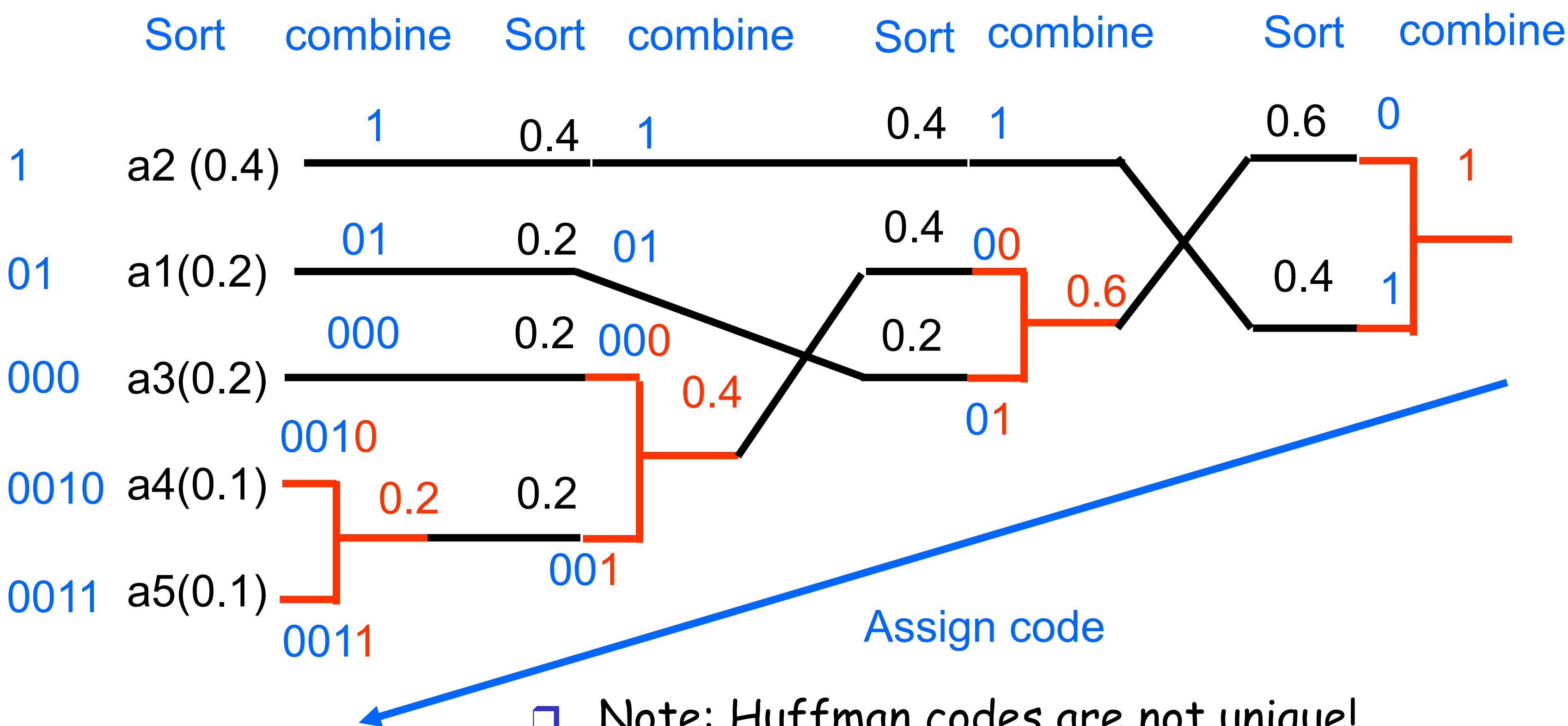
Frequency count of the symbols in "HELLO"

Coding Tree



Huffman Coding

- Source alphabet A = {a1, a2, a3, a4, a5}
- Probability distribution: {0.2, 0.4, 0.2, 0.1, 0.1}



- Note: Huffman codes are not unique!
  - Labels of two branches can be arbitrary.
  - Multiple sorting orders for tied probabilities

Exam Sample

- MIDI
  - What is MIDI?
  - How many I/O ports does MIDI support? What are they?
  - We have suddenly invented a new kind of music: "18-tonemusic", that requires a keyboard with 180 keys. How would we have to change the MIDI standard to be able to play this music?

Exam Sample

- Color Look up table
  - What is a color look-up table and how is it used to represent color?
  - Give an advantage and a disadvantage of this representation with respect to true color (24-bit) color
  - How do you convert from 24-bit color to an 8-bit color look up table representation?