Computer Graphics
-- Images & Devices
or Electronic Imaging 101

Junjie Cao @ DLUT

Spring 2019

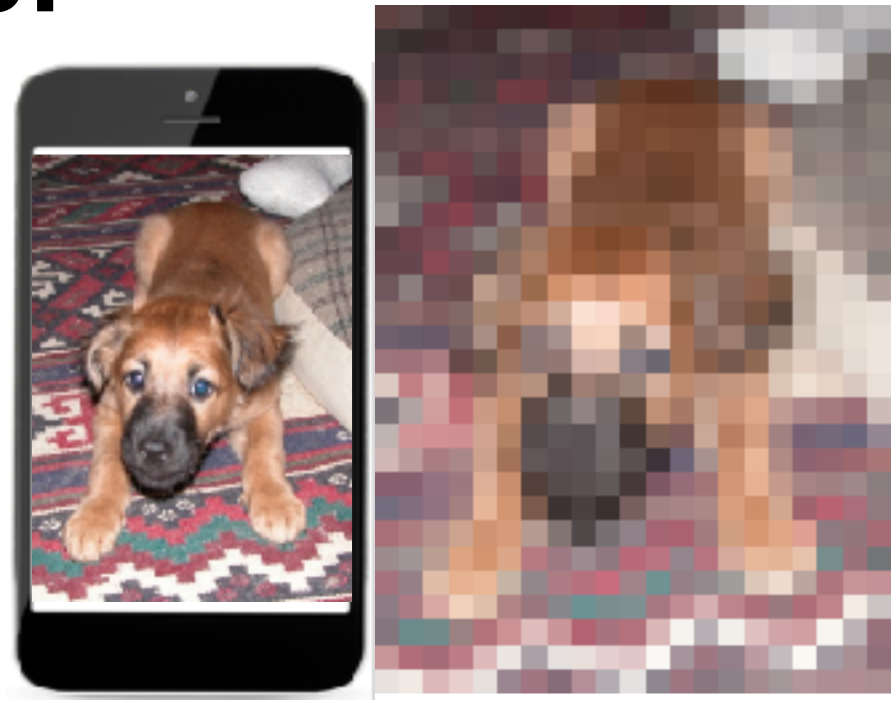
http://jjcao.github.io/ComputerGraphics/

What is an Image?

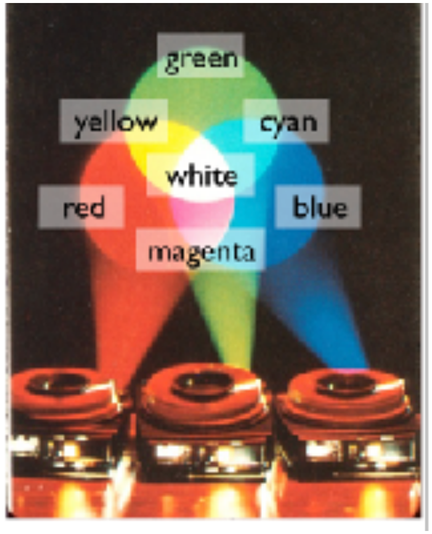
- A photographic print?
- This projection screen?
- Some numbers in RAM?

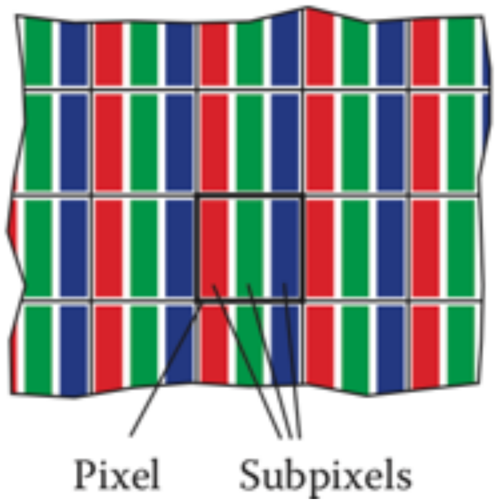
An image is:

- A 2D distribution of intensity or color
- A function defined on a two-dimensional plane



- **A raster image is a 2D rectilinear array of pixels**
  - Spatial resolution: w*h pixels
- ***Pixel* is short for "picture element."**





Raster Devices -- output

- raster image is ***device-independent* description** of image
- display device is a way of **approximating** ideal image.
- Displays
  - CRT
  - LCD
  - LED
  - Electrophoretic

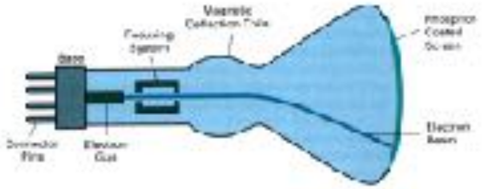
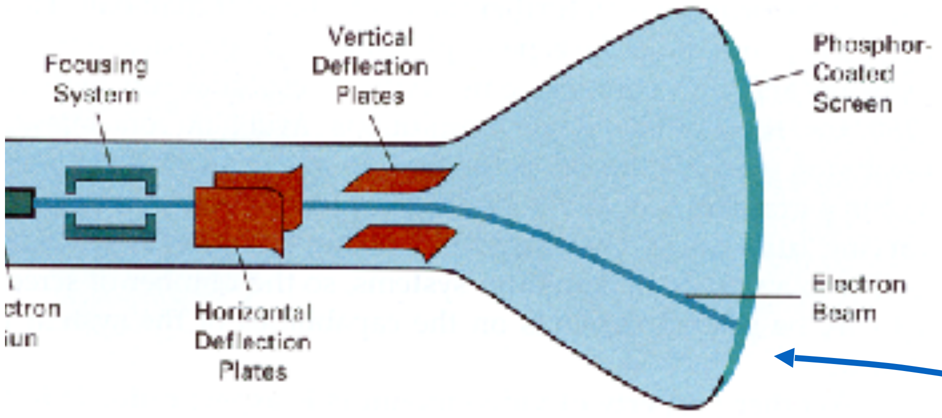


Image Display

- First widely used electronic display
  - Developed for TV in the 1920s–1930s
- Re-create continuous function from samples
  - Example: cathode ray tube



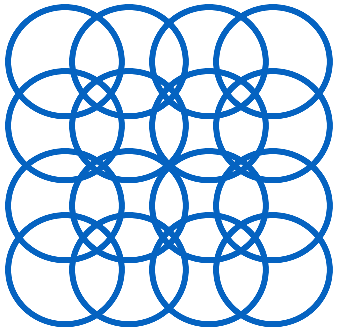
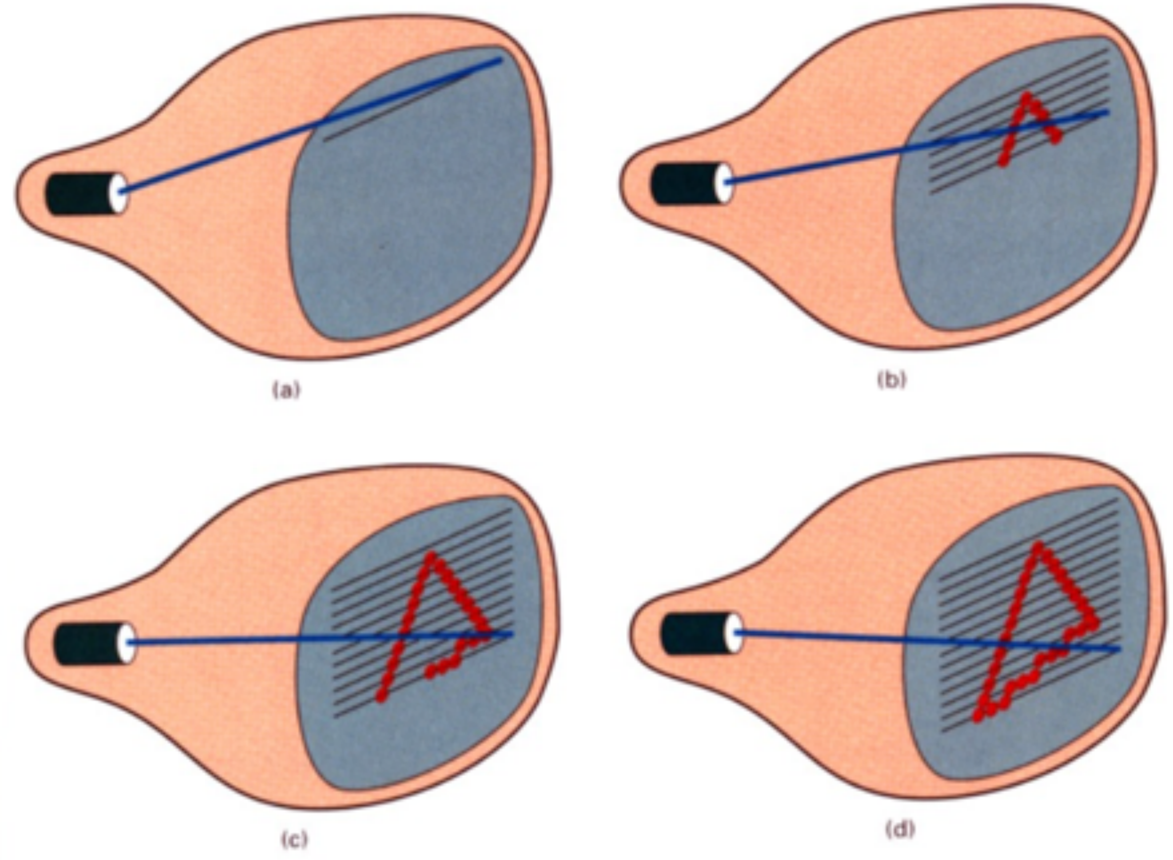


Image is reconstructed by displaying pixels with finite area (Gaussian)

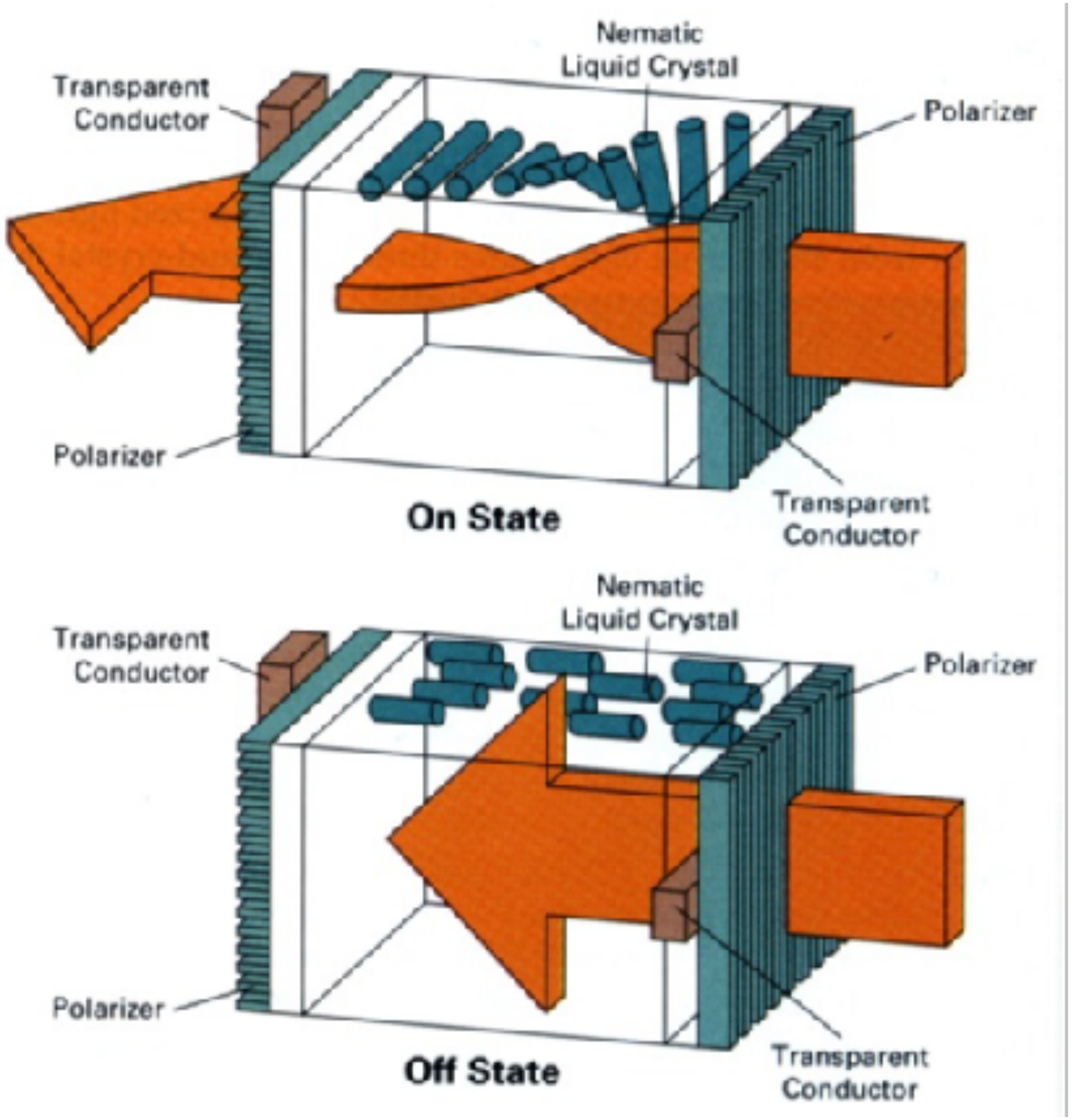
Raster CRT display

- Scan pattern fixed in display hardware
- Intensity modulated to produce image
- Originally for TV
  - (continuous analog signal)
- For computer, intensity determined by contents of framebuffer



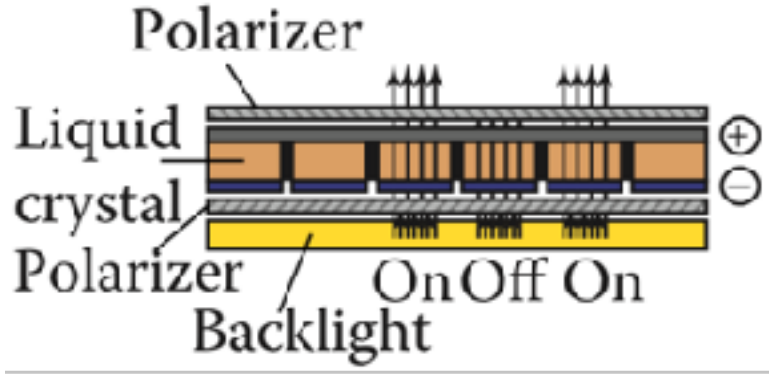
LCD

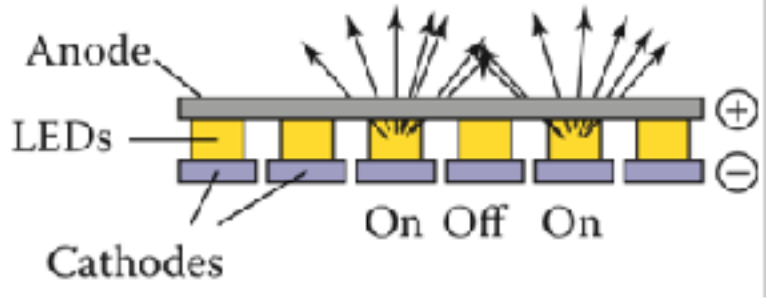
- Principle: block or transmit light by twisting its polarization
- Illumination from backlight (either fluorescent or LED)
- Intermediate intensity levels possible by partial twist
- Fundamentally raster technology
- Fixed format

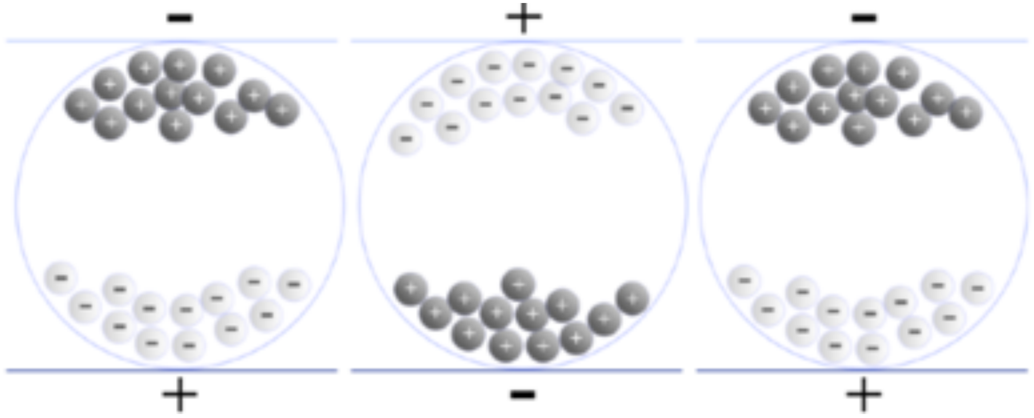


Raster Devices -- output

- Displays
  - Transmissive: liquid crystal display (LCD)
  - Emissive: light-emitting diode (LED) display
  - Electrophoretic



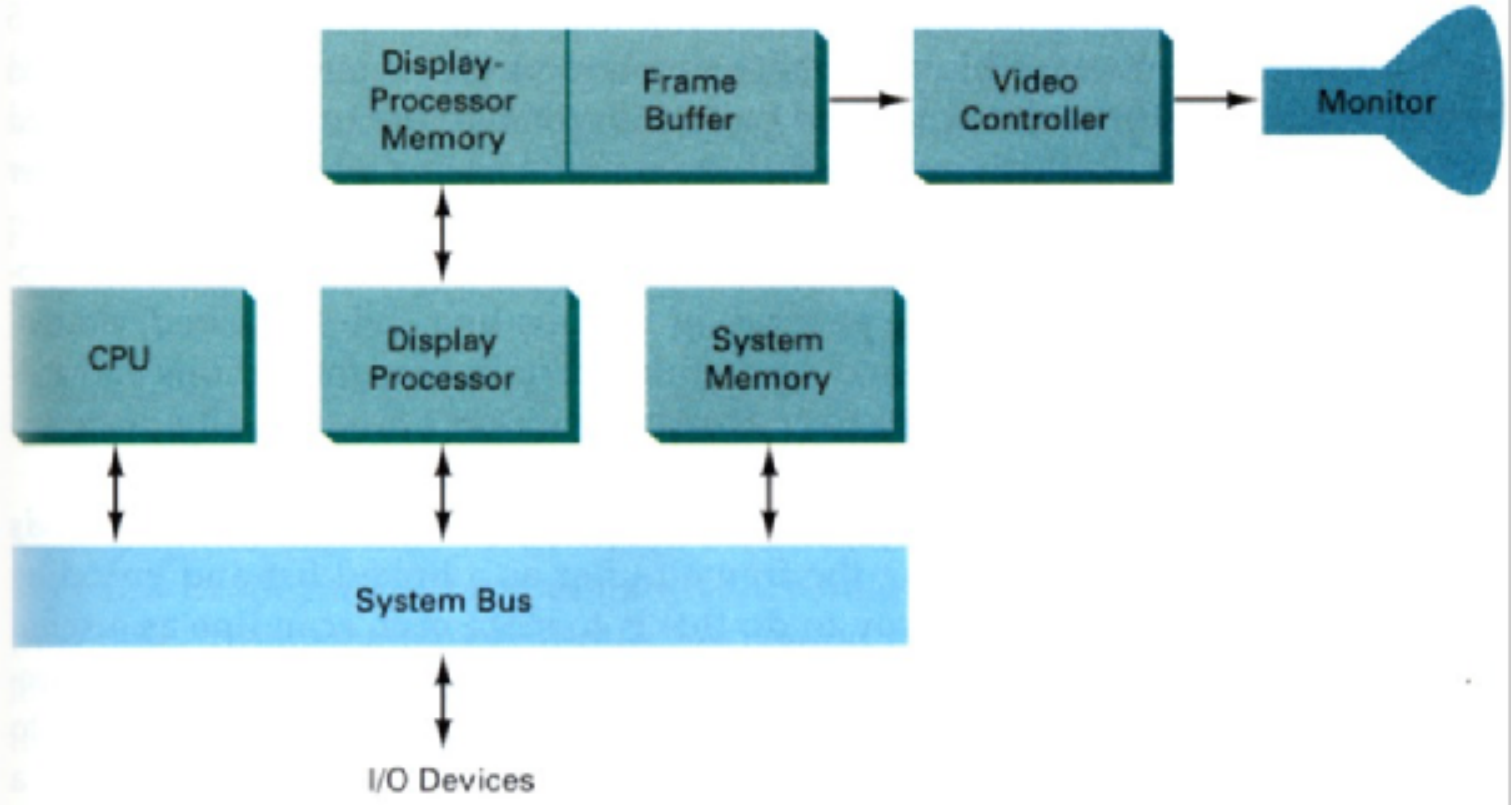




[Wikimedia Commons —Senarclens]

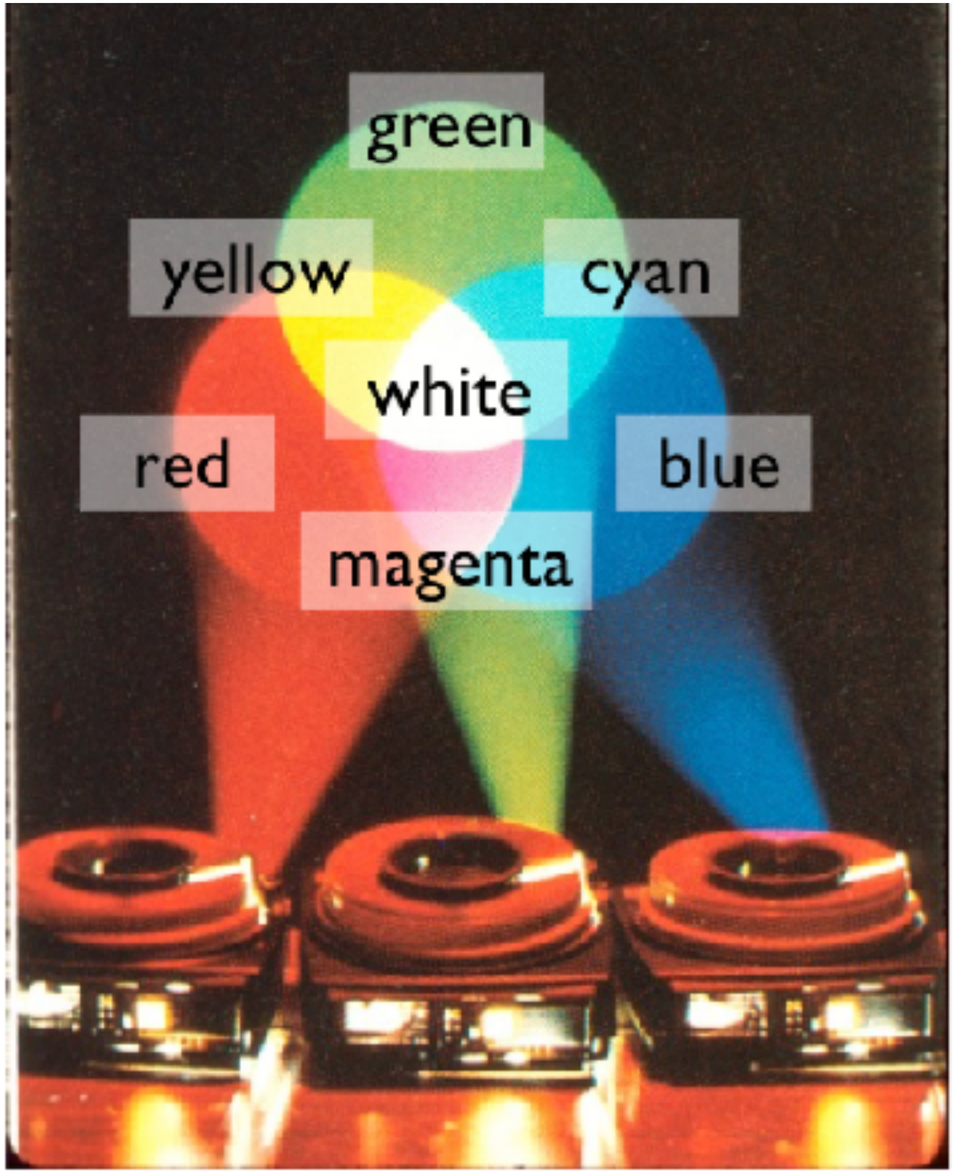
Raster display system

- Screen image defined by a 2D array in RAM
- In most (but not all) systems today, it's in a separate memory from the normal CPU memory
- The **framebuffer** is the memory area that maps to the screen



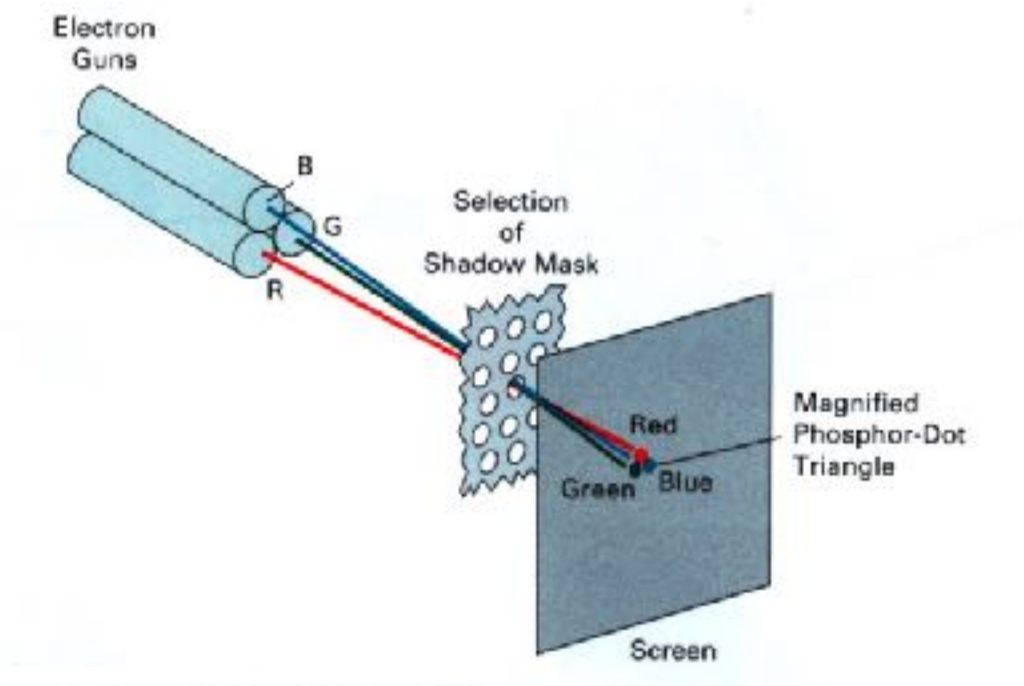
Color displays

- Operating principle: humans are trichromatic
  - match any color with blend of three
  - therefore, problem reduces to producing **3 images & blending**
  -
- **Additive** color
  – blend images by sum
  – e.g. overlapping projection
  – **R,G,B** make good primaries

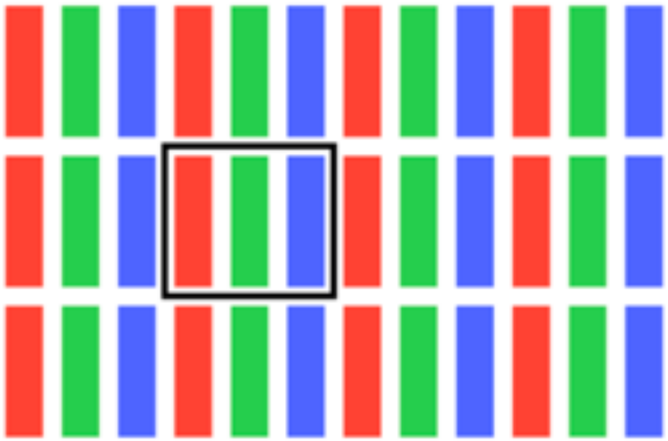


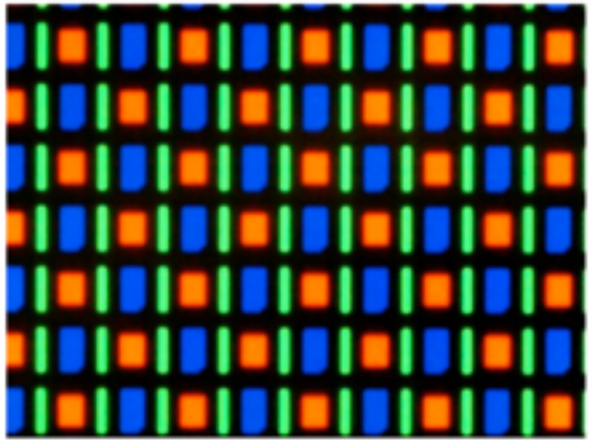
Color displays

- CRT: phosphor dot pattern to produce finely interleaved color images



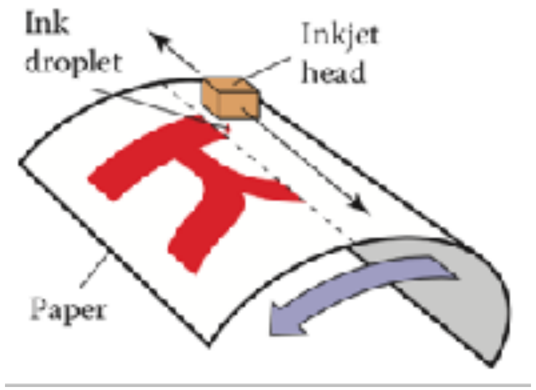
- LCD, LED: interleaved R,G,B pixels

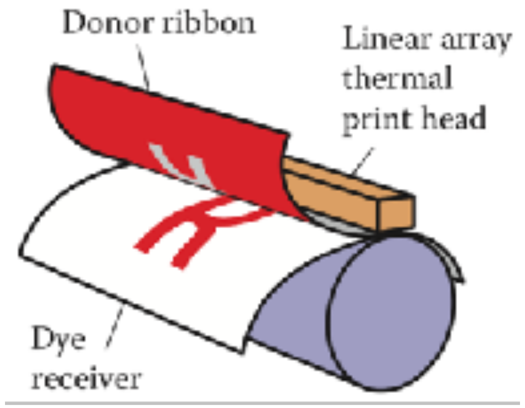




Raster Devices -- output

- Displays [resolution:1920 × 1200 pixels]
  - CRT
  - LCD
  - LED
  - Electrophoretic

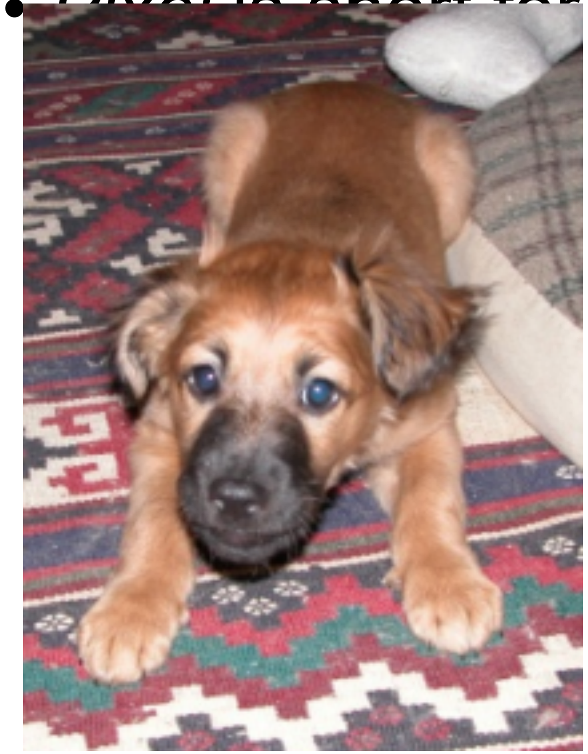




- Hardcopy
  - Binary: ink-jet printer [1200 *dots per inch* (dpi)]
  - Continuous tone: dye sublimation printer [300 *pixels per inch* (ppi)]

What is an Image?

- An image is a 2D rectilinear array of pixe
- *Pixel* is short for “picture element.”



Continuous image

Digital image

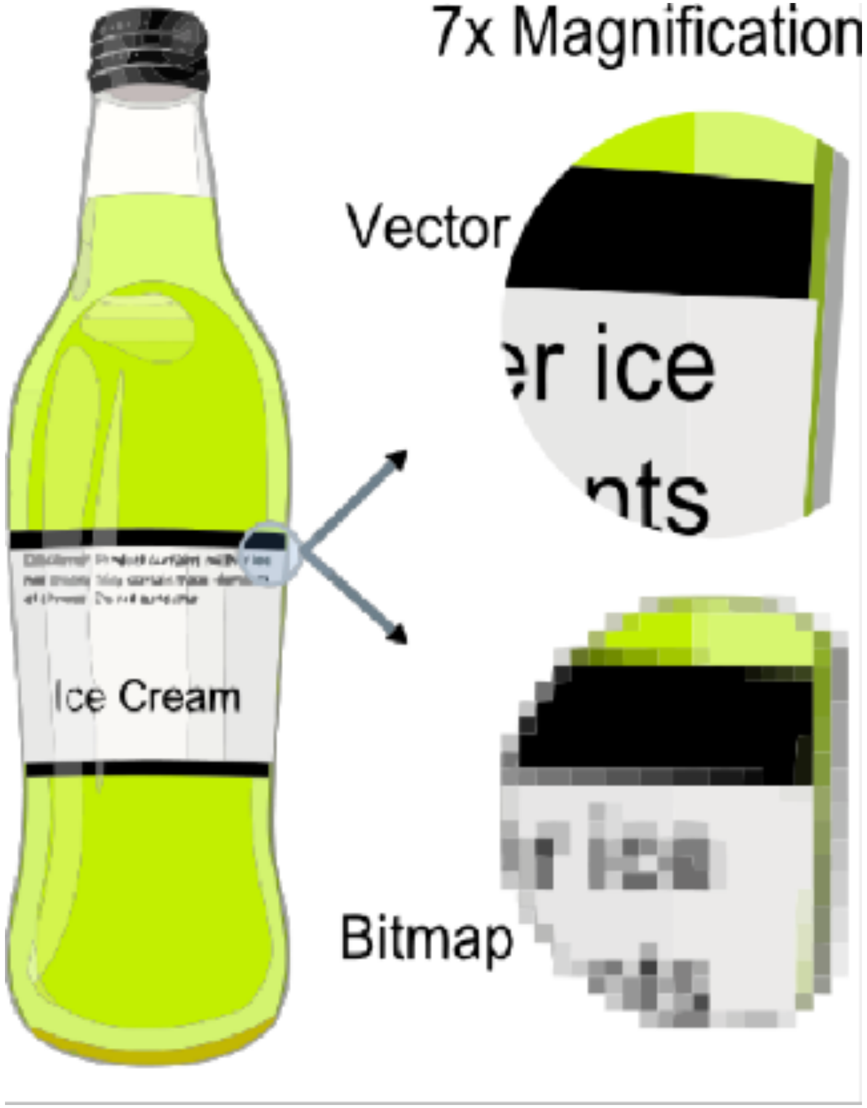
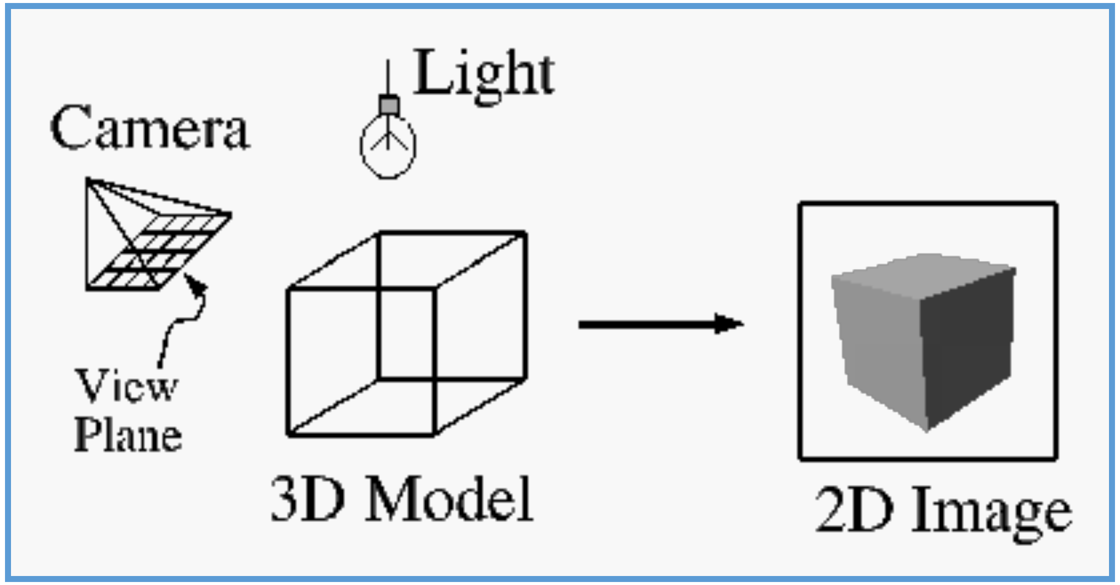


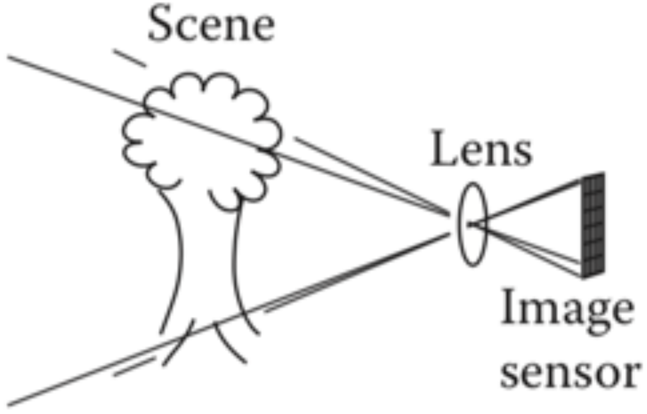
Image Acquisition

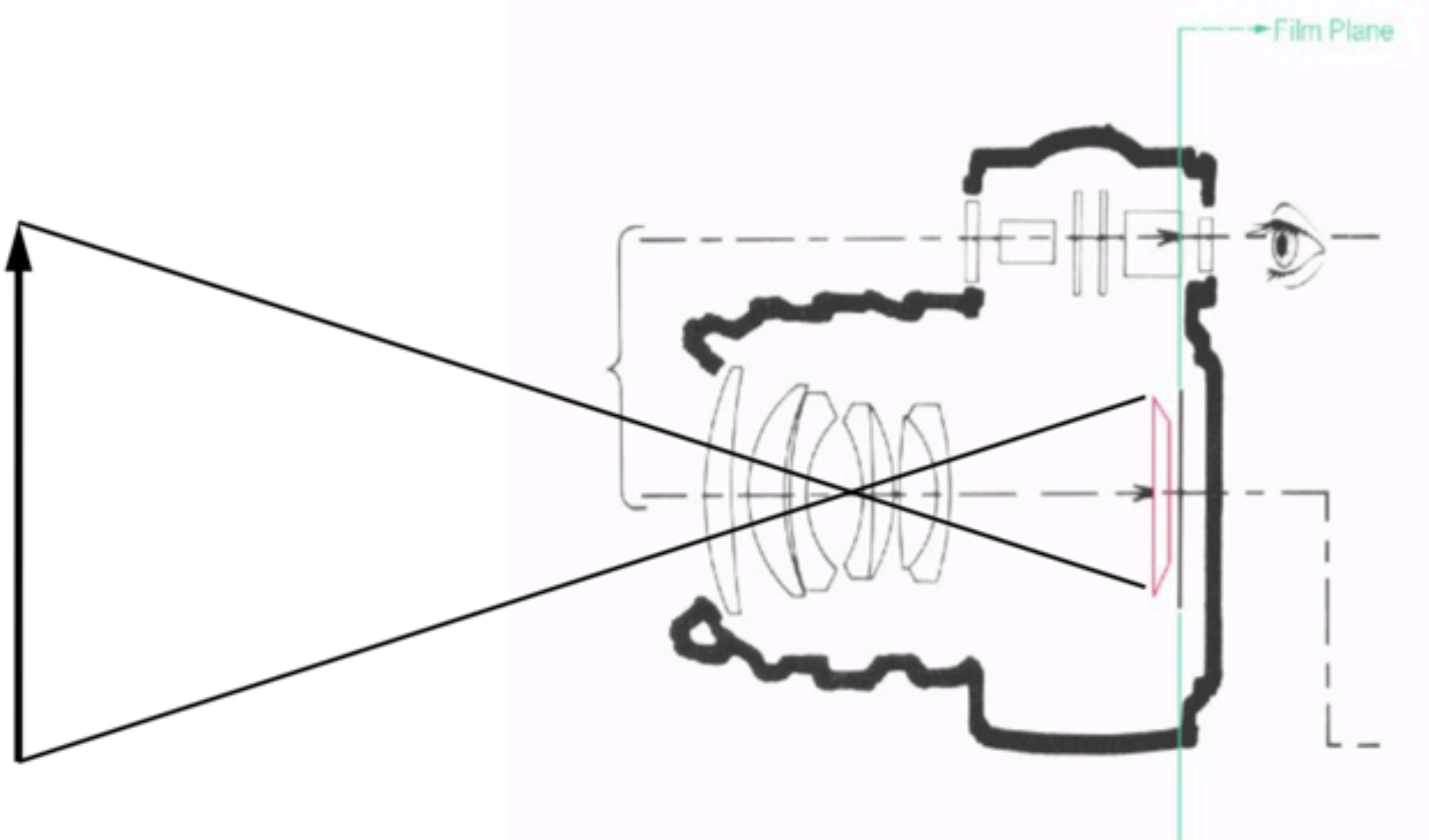
- Pixels are samples from continuous function
  - Photoreceptors in eye
  - CCD cells in digital camera
  - Rays in virtual camera



Raster Devices -- input

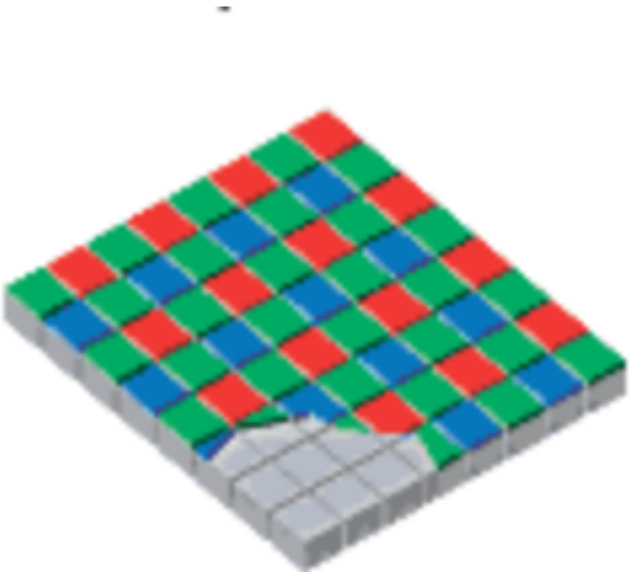
- 2D array sensor: digital camera
- 1D array sensor: flatb

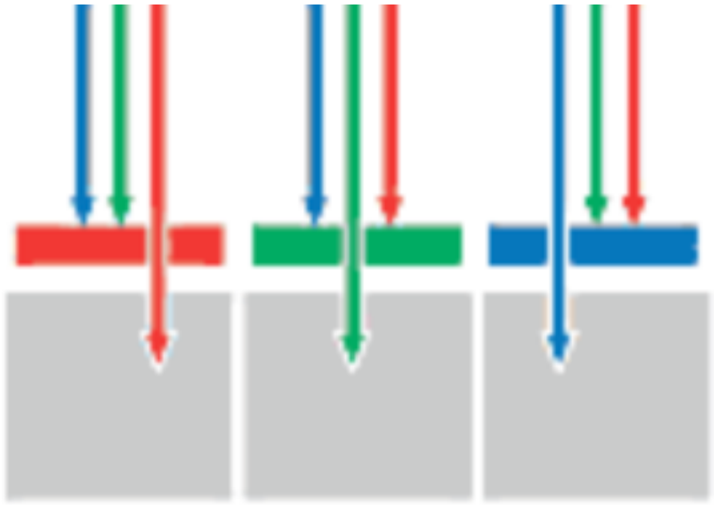




Digital camera

- Color typically captured using color mosaic
- *demosaicking*





Pixel Coordinates - Raster Image

the value x in a pixel, it means "**the value of the image in the vicinity of this grid point is x**."

- a pixel from a camera or scanner is a measurement of the average color of the image over some small area around the pixel.

- A display pixel, with its red, green, and blue subpixels, is designed so that the average color of the image over the face of the pixel is controlled by the corresponding pixel value in the raster image.

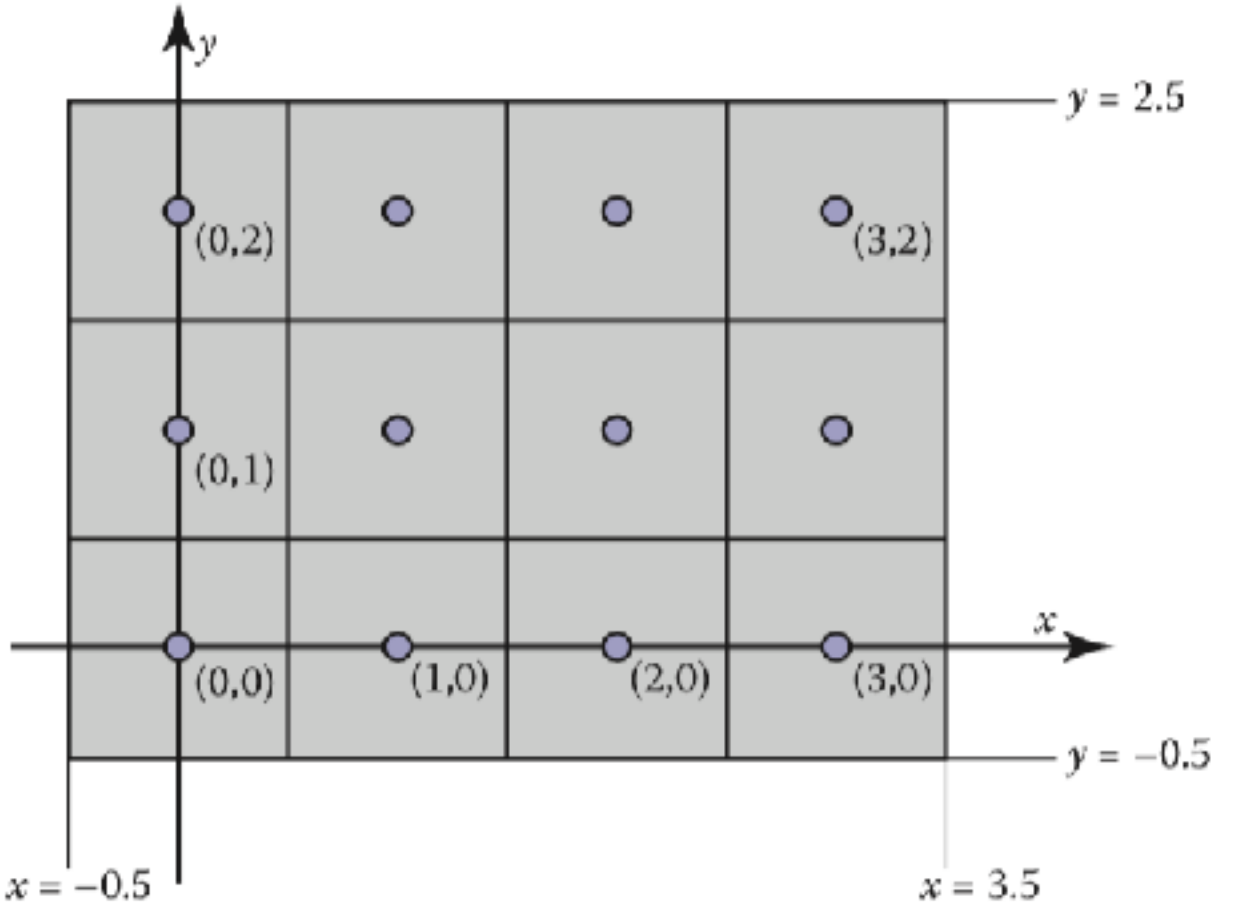


Image Resolution

- Spatial resolution
  - Image has only “Width” x “Height” pixels
- Temporal resolution
  - Monitor refreshes images at only “Rate” Hz
- Intensity resolution
  - Each pixel has only “Depth” bits for colors/intensities

Typical Resolutions

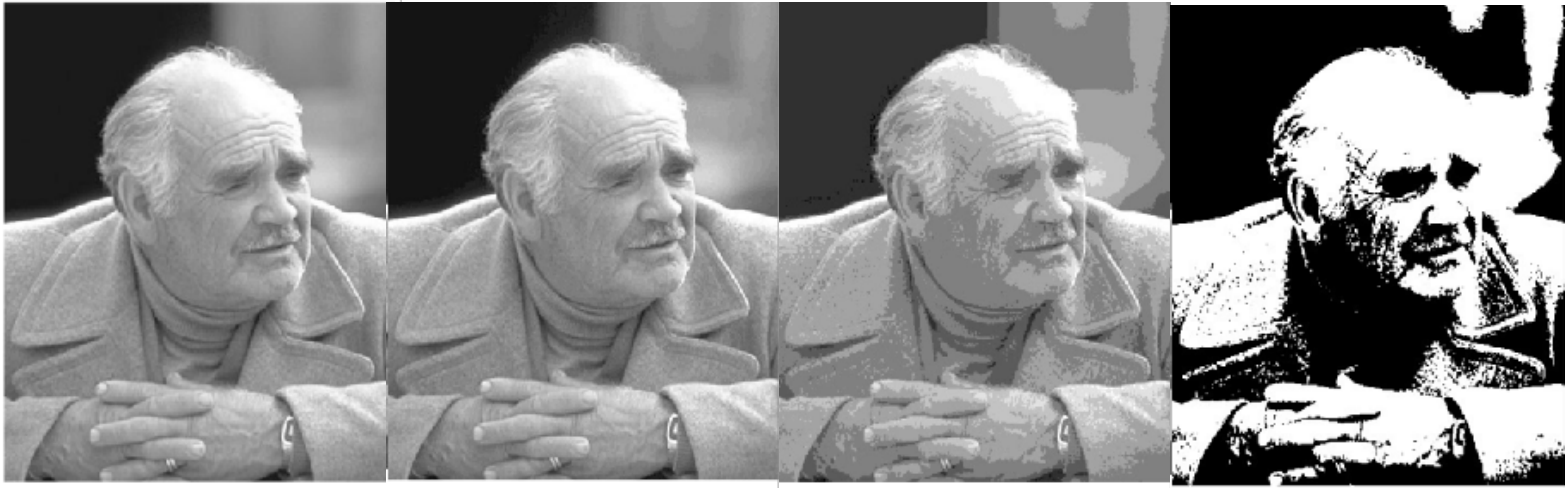
	Width x Height	Depth	Rate
NTSC	640 x 480	8	30
Workstation	1280 x 1024	24	75
Film	3000 x 2000	12	24
Laser Printer	6600 x 5100	1	-

Pixel Values (Framebuffer format)

- Binary: boolean per pixel (1bpp, i.e. 1-bit per pixel):
  - interp. = black and white; e.g. fax, text
- Grayscale: integer per pixel:
  - interp. = shades of gray; e.g. black-and-white print
  - precision: usually byte (8bpp); sometimes 10,12,or16bpp
- Color: 3 integers per pixel:
  - interp. = full range of displayable color; e.g. colorprint
  - precision: usually 3 bytes (24bpp); sometimes 16 (5+6+5) or 30 or 36 or 48bpp
- Floating point:
  - More abstract, because no output device has infinite range
  - **Provides *high dynamic range* (HDR)**
  - Represent real scenes independent of display
  - Becoming the standard intermediate format in graphics processor

Converting pixel precision

- Up is easy; down loses information—be careful



8 bpp (256 grays) | 5 bpp (32 grays) | 3 bpp (8 grays) | 1 bpp (2 grays)

Dithering

- When decreasing bpp, we quantize
  - Make choices consistently: banding
- Instead, be inconsistent—dither
  - Turn on some pixels but not others in gray regions
  - A way of trading spatial for tonal resolution
  - Choose pattern based on output device
    - laser, offset: clumped dots required (halftone)
    - inkjet, screen: dispersed dots can be used

Dithering methods

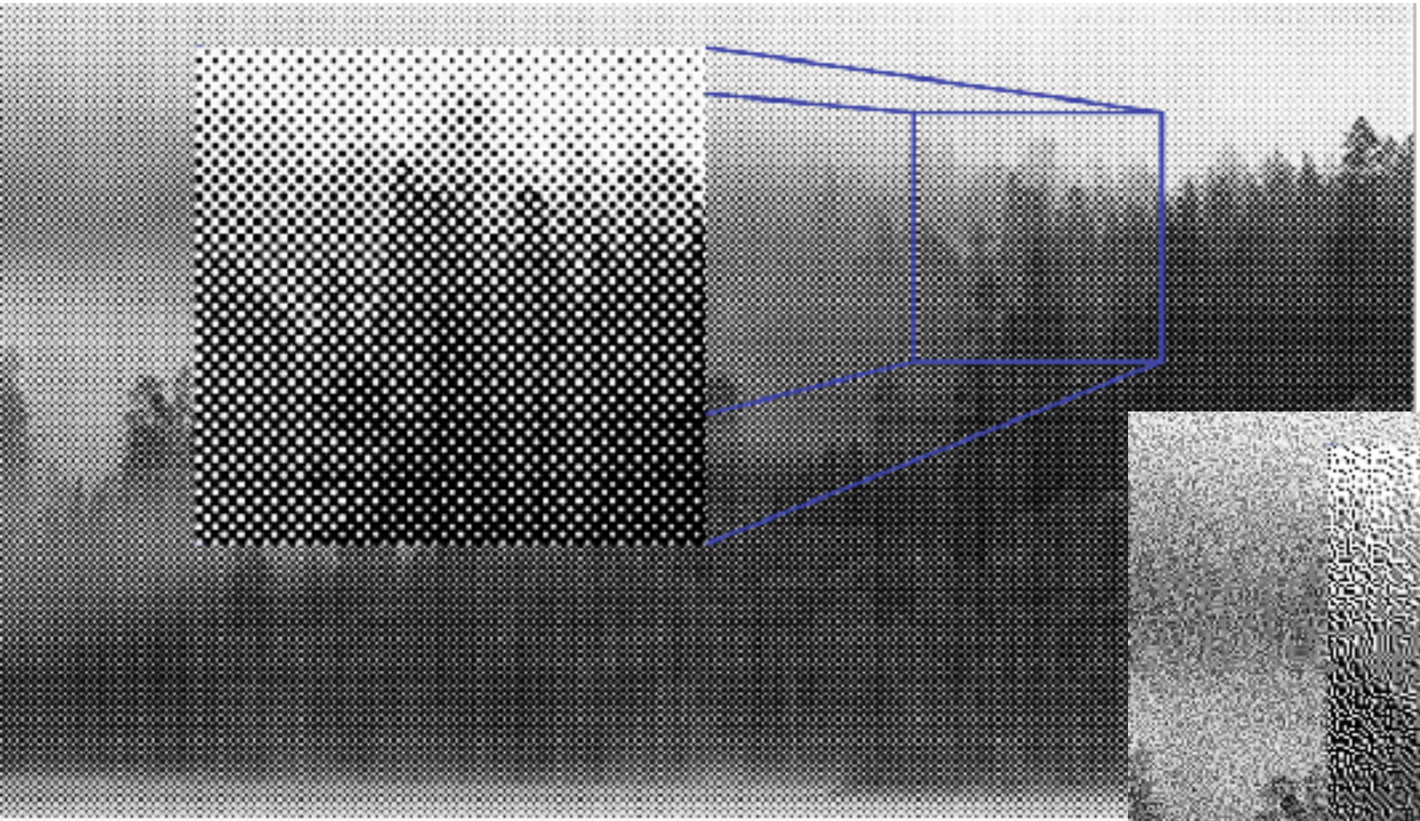
- Ordered dither
  - Based on traditional, optically produced halftones
  - Produces larger dots
- Diffusion dither
  - takes advantage of devices that can reproduce isolated dots
  - the modern winner for desktop printing

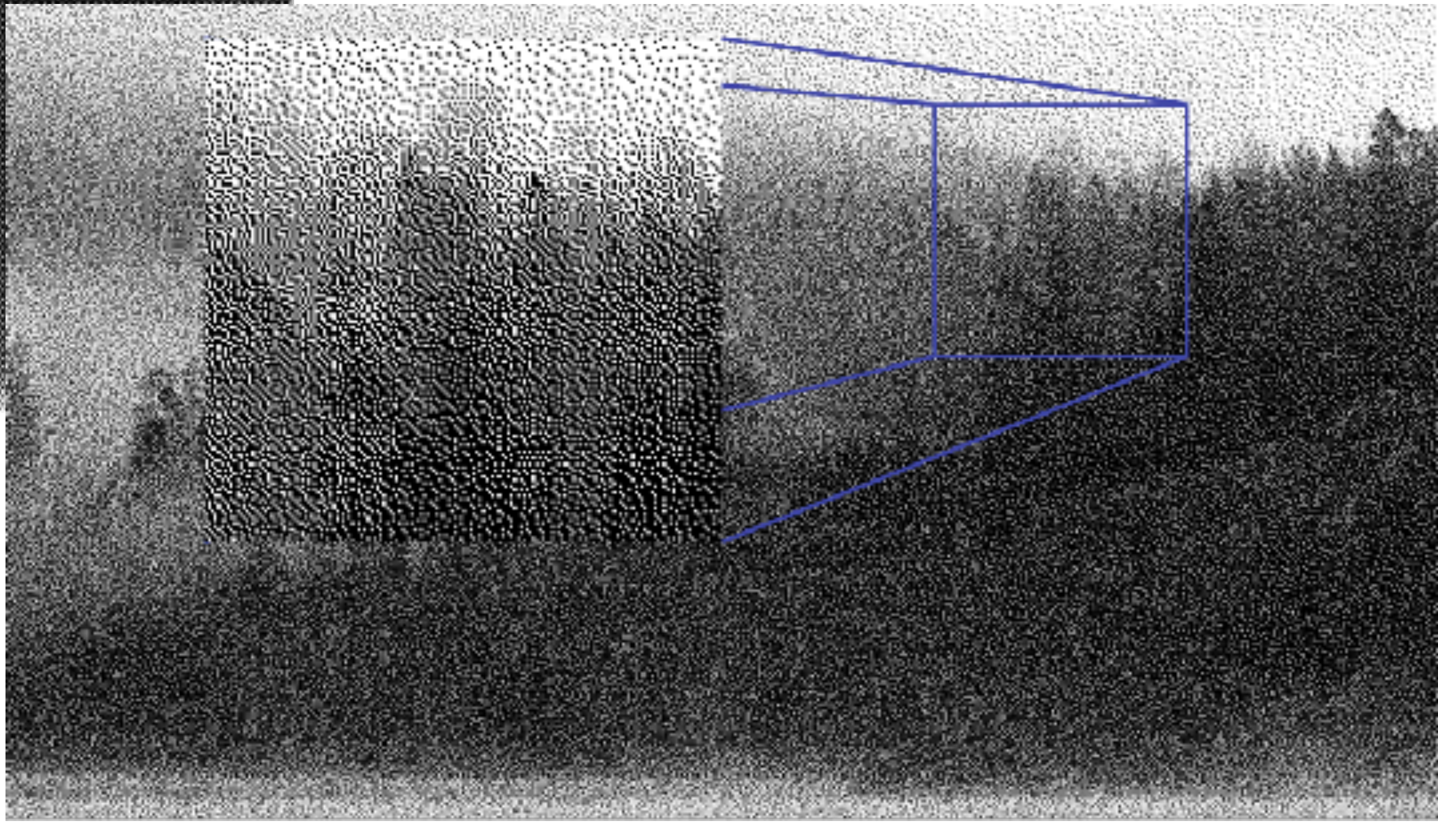
Ordered Dither example

Diffusion dither

- Produces scattered dots with the right local density

Ordered dither vs Diffusion dither





Intensity encoding in images

- What do the numbers in images (pixel values) mean?
  - they determine how bright that pixel is
  - bigger numbers are (usually) brighter
- ***Monitor: pixel value => intensity***
- ***Transfer function***: function that **maps input pixel value to luminance of displayed image**

$$I = f(n) \quad f : [0, N] \rightarrow [I_{\min}, I_{\max}]$$

- What determines this function?
  - physical constraints of device or medium
  - Desired visual characteristics

Constraints on transfer function

- Maximum displayable intensity, Imax
  - How much power can be channeled into a pixel?
    - LCD: backlight intensity,transmission efficiency(<10%)
    - projector: lamp power, efficiency of imager and optics

- Minimum displayable intensity, Imin
  - light emitted by the display in its “off” state
    - *e.g.* stray electron flux in CRT, polarizer quality in LCD

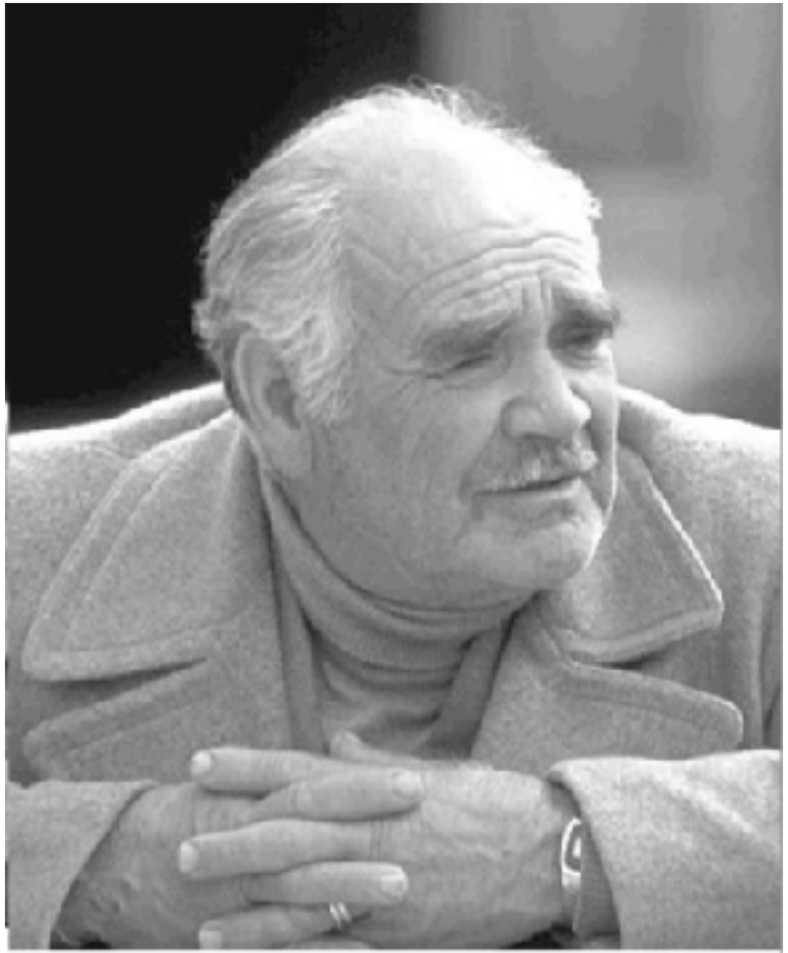
- Viewing flare, k: light reflected by the display
  - very important factor determining image contrast in practice
  - 5% of Imax is typical in a normal office environment [sRGB spec]
  - Much effort to make very black CRT and LCD screens

Dynamic range

- Dynamic range $R_d = I_{max} / I_{min}$, or $(I_{max} + k) / (I_{min} + k)$
  - determines the degree of image contrast that can be achieved
  - a major factor in image quality
- Ballpark values
  - Desktop display in typical conditions: 20:1
  - Photographic print: 30:1
  - Desktop display in good conditions: 100:1
  - High-end display under ideal conditions: 1000:1
  - Digital cinema projection: 1000:1
  - Photographic transparency (directly viewed): 1000:1
  - High dynamic range display: 10,000:1

Transfer function shape

- Real displays take **discrete input values**
- Desirable property: the change **from one pixel value to the next highest pixel value** should not produce a visible contrast
  - Otherwise smooth areas of images will show visible bands
- What contrasts are visible?
  - Rule of thumb: under good conditions we can notice a **2% change in intensity**
  - Therefore we generally need smaller quantization steps in the darker tones than in the lighter tones
  - Most efficient quantization is logarithmic



5 bpp (32 grays)

How many levels are needed?

- Depends on dynamic range
  - 2% steps are most efficient:

$$0 \mapsto I_{\text{min}}; 1 \mapsto 1.02 I_{\text{min}}; 2 \mapsto (1.02)^2 I_{\text{min}}; \ldots$$

  - log 1.02 is about 1/120, so 120 steps per decade of dynamic range
    - 240 for desktop display
    - 360 to print to film
    - 480 to drive HDR display
- If we want to use linear quantization (equal steps)
  - one step must be < 2% (1/50) of I_{min}
  - need to get from ~0 to $I_{min} \cdot R_d$ so need about 50 R_d levels
    - 1500 for a print; 5000 for desktop display; 500,000 for HDR display
- Moral: 8 bits is just barely enough for low-end applications
  - but only if we are careful about quantization

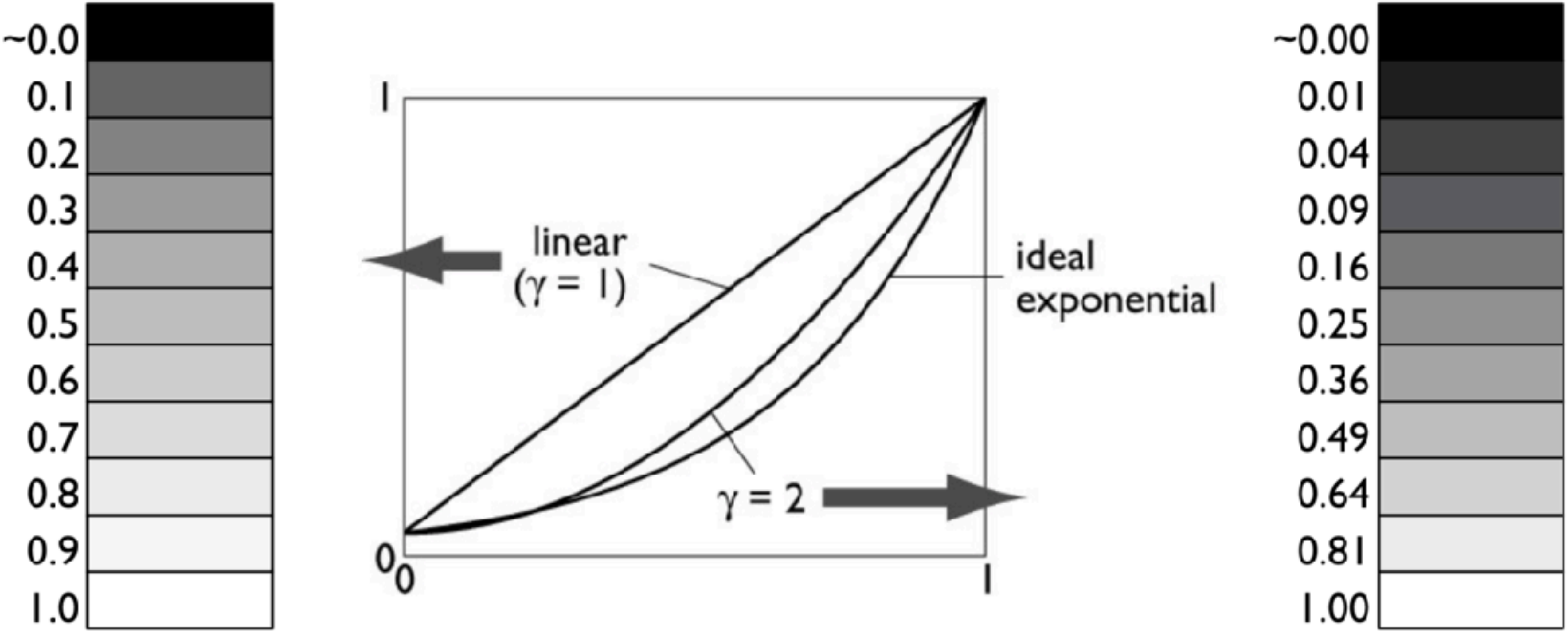
Intensity quantization in practice

- Option 1: linear quantization $I(n) = (n/N)\, I_{\max}$
  - pro: simple, convenient, amenable to arithmetic
  - con: requires more steps (wastes memory)
  - need 12 bits for any useful purpose; more than 16 for HDR
- Option 2: power-law quantization $I(n) = (n/N)^{\gamma}\, I_{\max}$
  - pro: fairly simple, approximates ideal exponential quantization
  - con: need to linearize before doing pixel arithmetic
  - con: need to agree on exponent
  - 8 bits are OK for many applications; 12 for more critical ones

Why gamma?

- Power-law quantization, or *gamma correction* is most popular
- Original reason: CRTs are like that
  - Intensity on screen is proportional to (roughly) voltage^2
- Continuing reason: inertia + memory savings
  - inertia: gamma correction is close enough to logarithmic that there’s no sense in changing
  - memory: gamma correction makes 8bits per pixel an acceptable option

Gamma quantization



- Close enough to ideal perceptually uniform exponential

Gamma correction

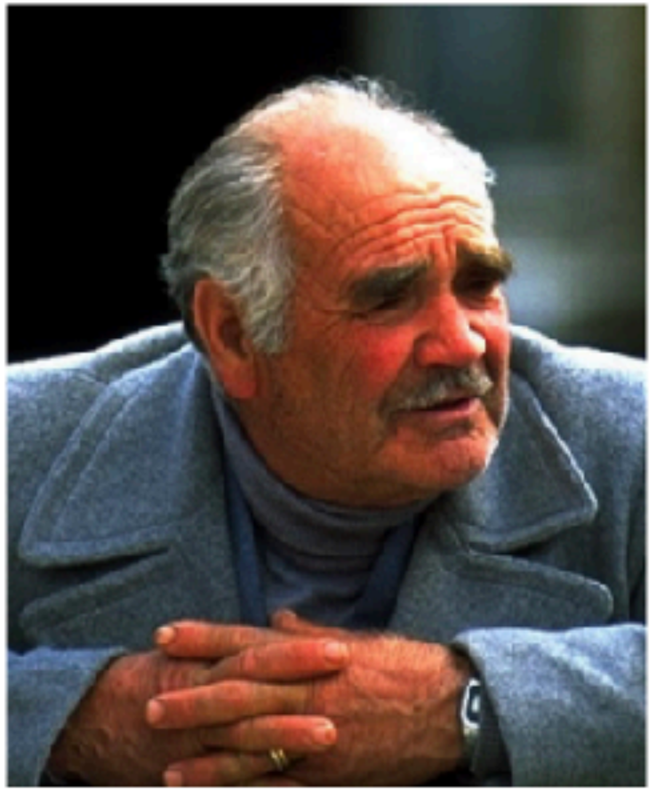
- Sometimes (often, in graphics) we have computed intensities *a* that we want to display linearly
- In the case of an ideal monitor with zero black level,

$$I(n) = (n/N)^{\gamma}$$

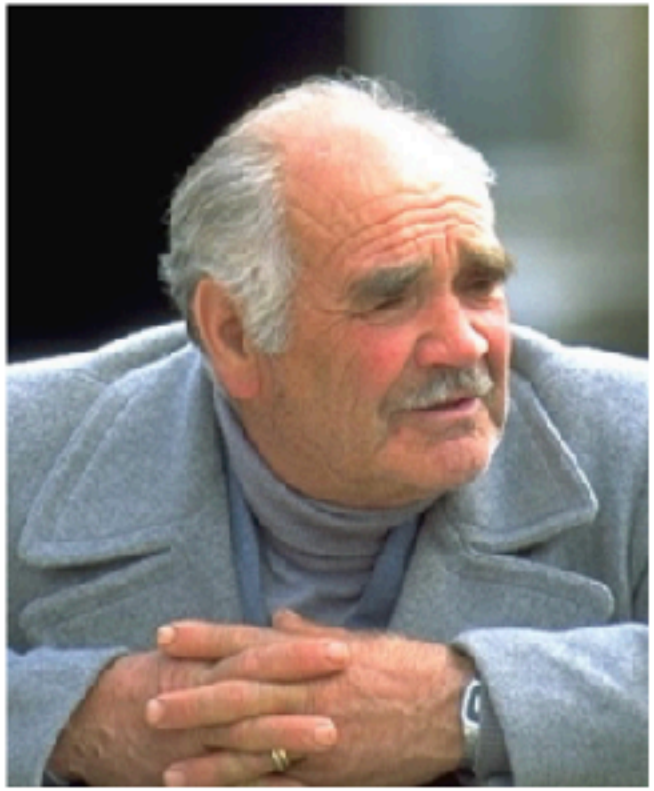
(where $N = 2^n - 1$ in *n* bits). Solving for *n*:

$$n(I) = N I^{\frac{1}{\gamma}}$$

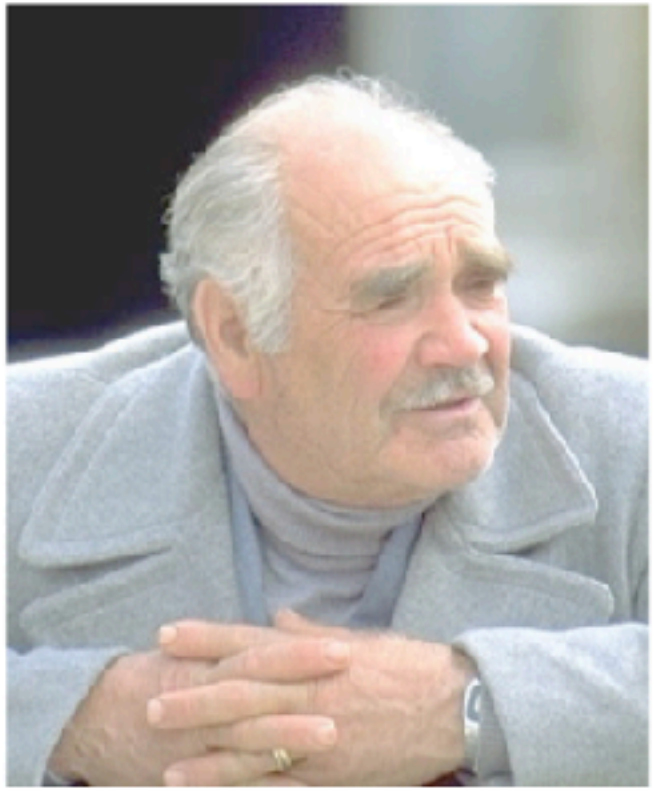
Gamma correction



corrected for γ lower than display



OK



corrected for γ higher than display

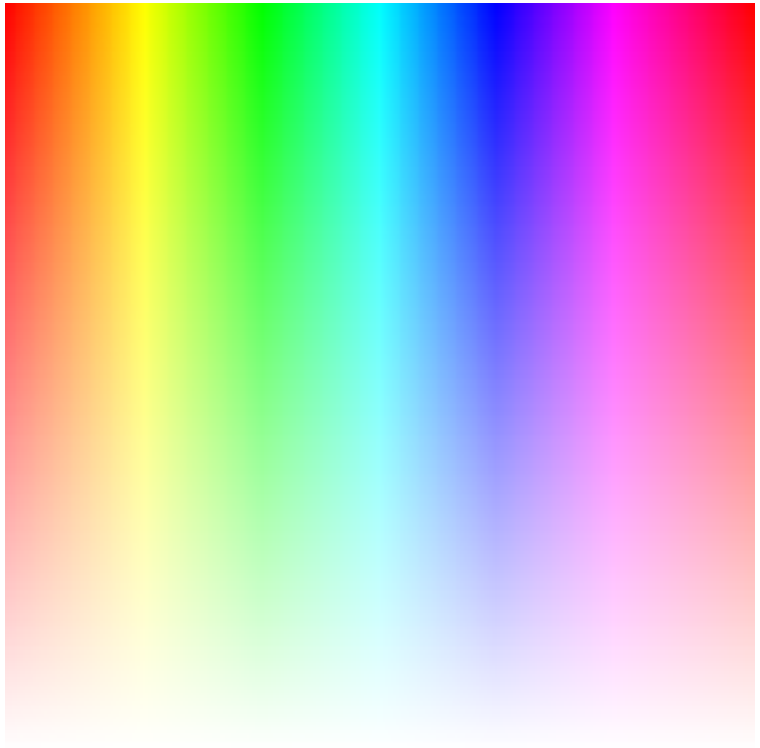
Datatypes for raster images

- For color or grayscale, sometimes add *alpha* channel
- Describes transparency of images
- More on this in a few lectures

Alpha Compositing

- A way to represent transparency
- The pixels of an image are blended linearly with the image below

$\alpha = 1$



$\alpha = 0$

$$\mathbf{c} = \alpha \mathbf{c}_f + (1 - \alpha)\mathbf{c}_b$$

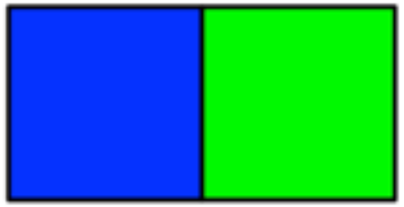
RGBA is very common, and you will use it often!

Converting pixel formats

Color to gray

- could take one channel (blue, say)
  - Leads to odd choices of gray value
- Combination of channels is better
  - But different colors contribute differently to lightness
  - Which is lighter, full blue or full green?
  - Good choice: gray = 0.2R+0.7G+0.1B
  - More on this in color,later on

Same pixel values



Same luminance?