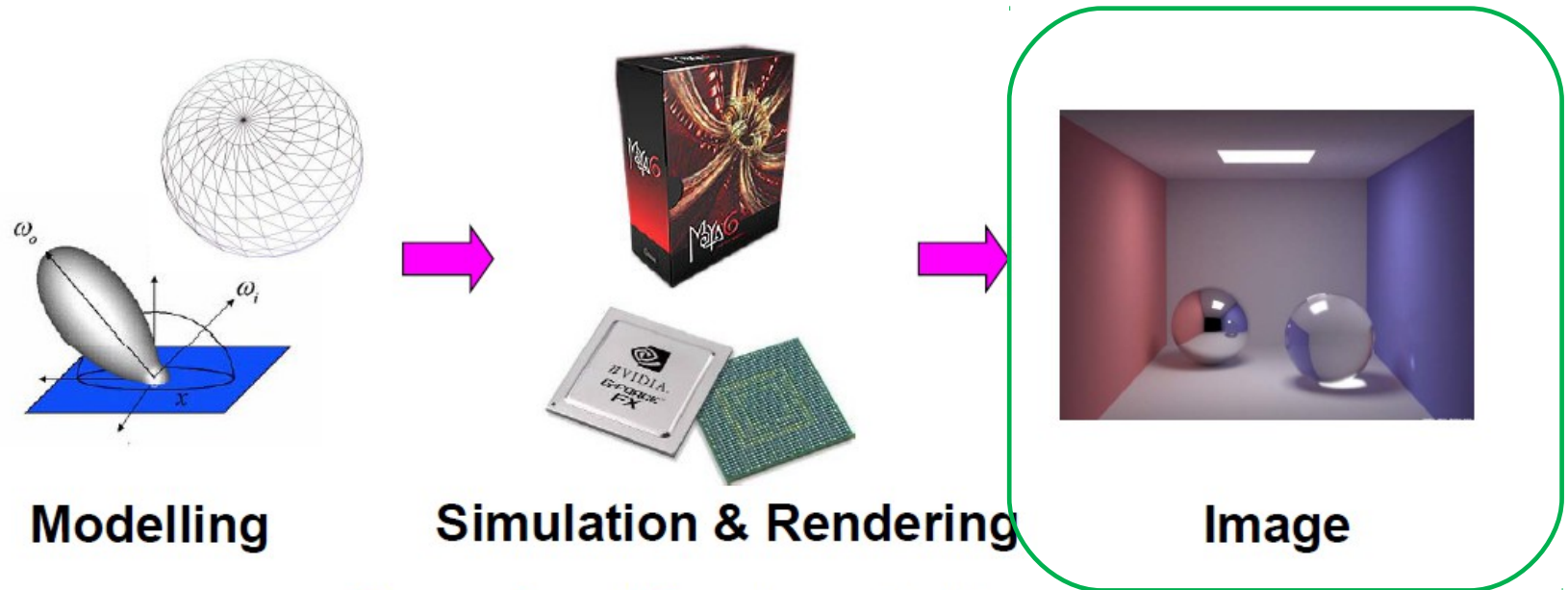


Images and Displays

Lecture 2

Introduction

- **Computer graphics:** The study of creating, manipulating, and using visual images in the computer.



Computer vision inverts the process
Image processing deals with images

What is an image?

- A photographic print?
- Pixels?
- Some numbers in RAM?

Google images

Web **Images** Videos News Maps Books

About 24,620,000,000 results

Any size
Large
Medium
Icon


Any color
Full color
Black and white
Transparent


Any type
Face
Photo
Clip art
Line drawing
Animated


Any time
Past 24 hours
Past week


Not filtered by license
Labeled for reuse with modification


You are seeing the basic version because we think your Internet connection is slow - [Switch to standard version](#)



nasa.gov
The **image** is, however, remarkably consistent and stable despite the fact ...
4096 × 4096 - 15262k - jpg



3rdbillion.net
fantastic **images** for facebook
1920 × 1080 - 587k - jpg

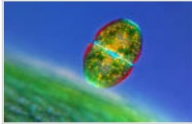

blog.gettyimages.com
Siberian-Tiger-Running-Through-Snow-Tom-Brakefield-Getty-
660 × 441 - 79k - jpg


hdwallpaper2013.com
Download Forest Background **Images** HD Wallpaper
1920 × 1080 - 450k - jpg


hdwallpaper2013.com
Beautiful Nature **Images** HD Wallpaper
1920 × 1080 - 497k - jpg


businessinsider.com
The Most Amazing Satellite **Images** Of The Year


3rdbillion.net
25ce2ed7b0e9a4ac68da2263cd8694cf.jpg (1920×1200) great **images** for facebook


arstechnica.com
... a z-stack of **images**, which is why there's such a good sense of depth

An image is:

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

$$I : \mathbb{R}^2 \rightarrow \dots$$

- Note: no mention of pixels yet
- To do graphics, must:
 - represent images—encode them numerically
 - Image files such as jpg, png, svg files
 - display images—realize them as actual intensity distributions

Representative display technologies

Computer displays

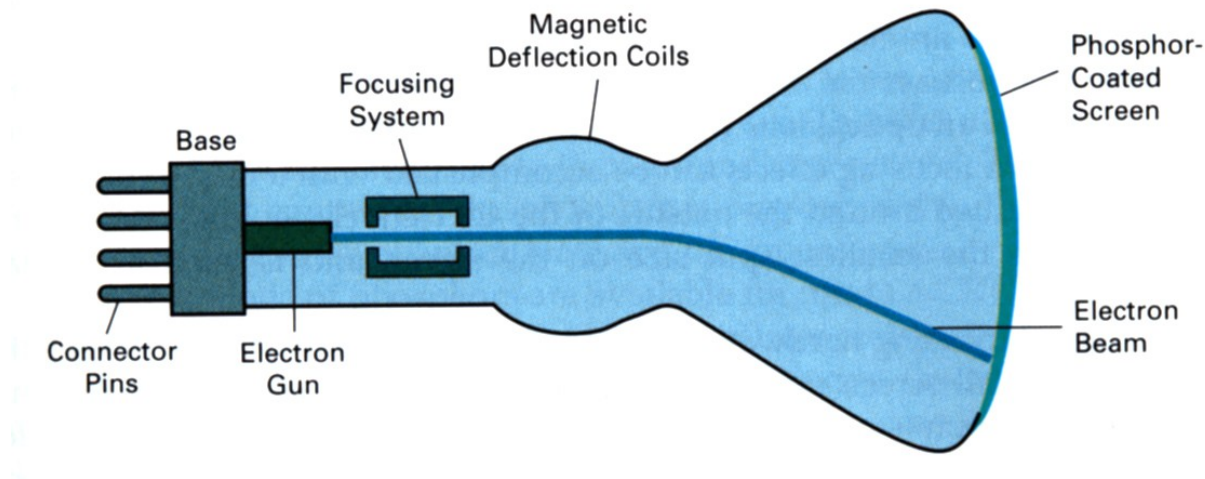
- Raster CRT display
- LCD display
- OLED display, DLP projector, ...

Printers

- Laser printer
- Inkjet printer

Cathode ray tube

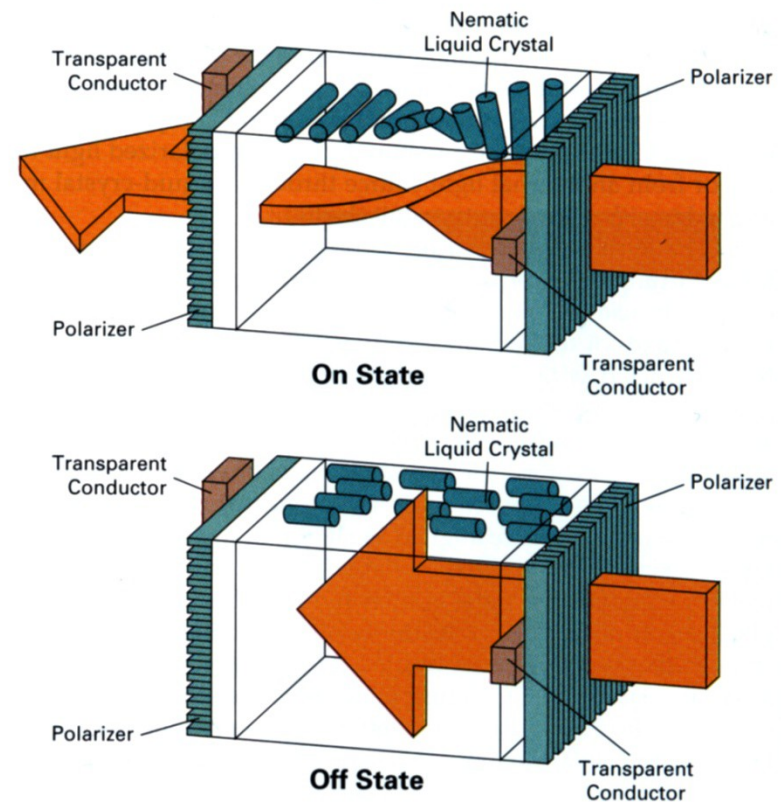
- First widely used electronic display
 - developed for TV in the 1920s-1930s



[H&B fig. 2-2]

LCD flat panel or projection display

- block or transmit light by twisting its polarization
- Intermediate intensity levels possible by partial twist

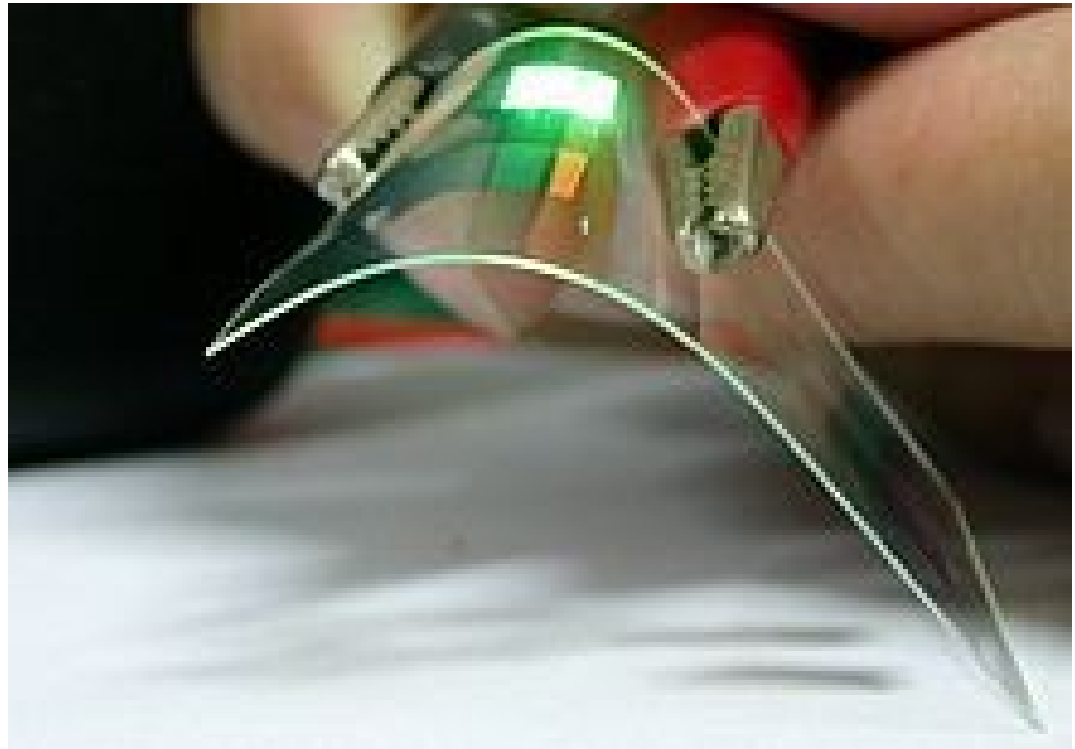


[H&B fig. 2-16]

OLED display

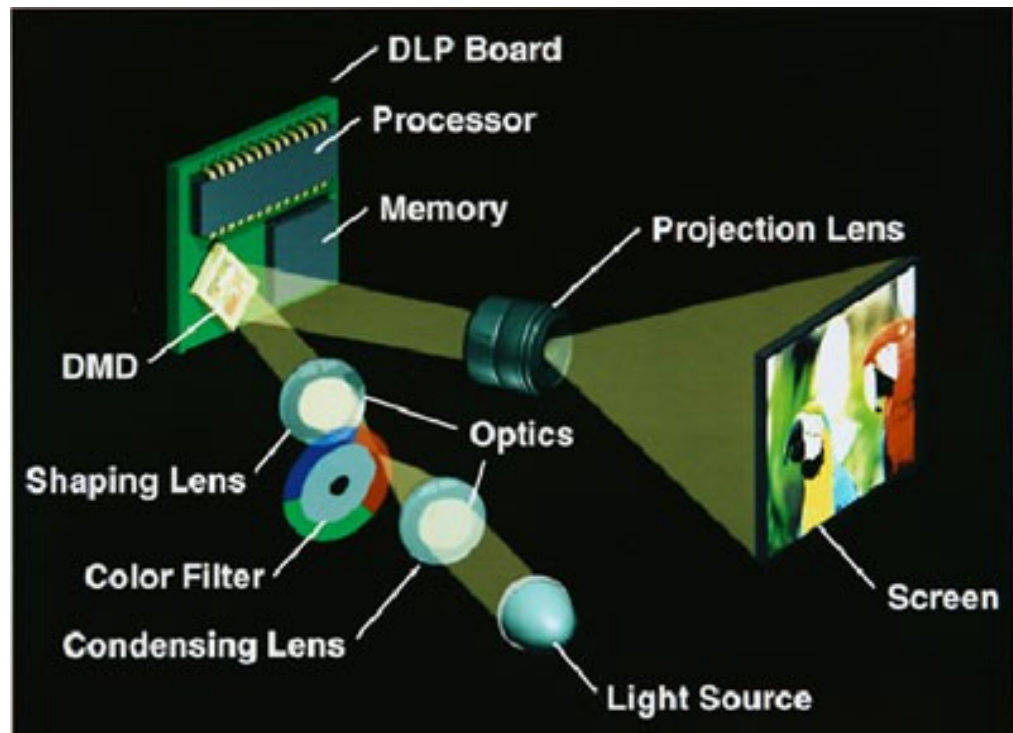
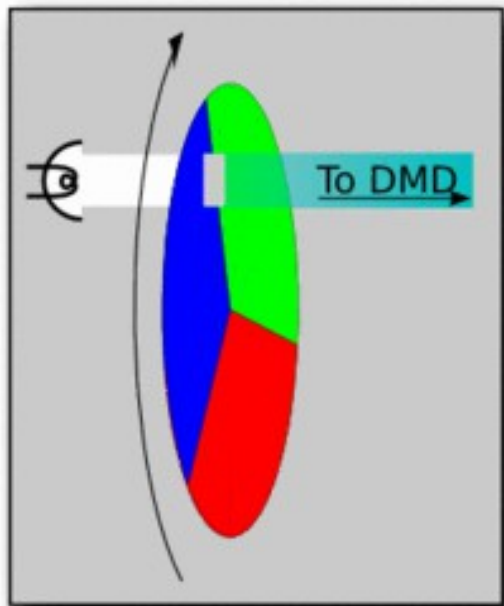
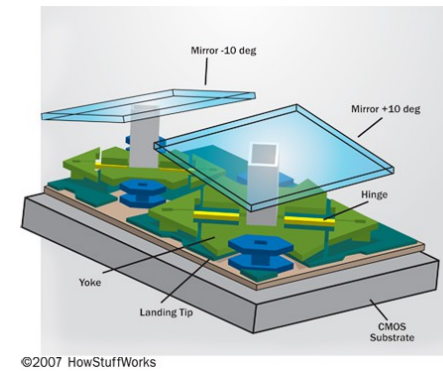
- An OLED is a light-emitting diode
- In low ambient light conditions such as a dark room, an OLED can achieve a much higher contrast ratio than an LCD

**A demonstration of a flexible OLED device
(from Wikipedia)**



DLP projector

- Digital light processing
- The image is created by microscopically small mirrors laid out in a matrix on a semiconductor chip, known as a Digital Micromirror Device (DMD)



Color displays (light emitting devices)

- Operating principle: humans are trichromatic
 - match any color with blend of three
 - therefore, problem reduces to producing 3 images and blending

- Additive color
(as opposed to subtractive color)

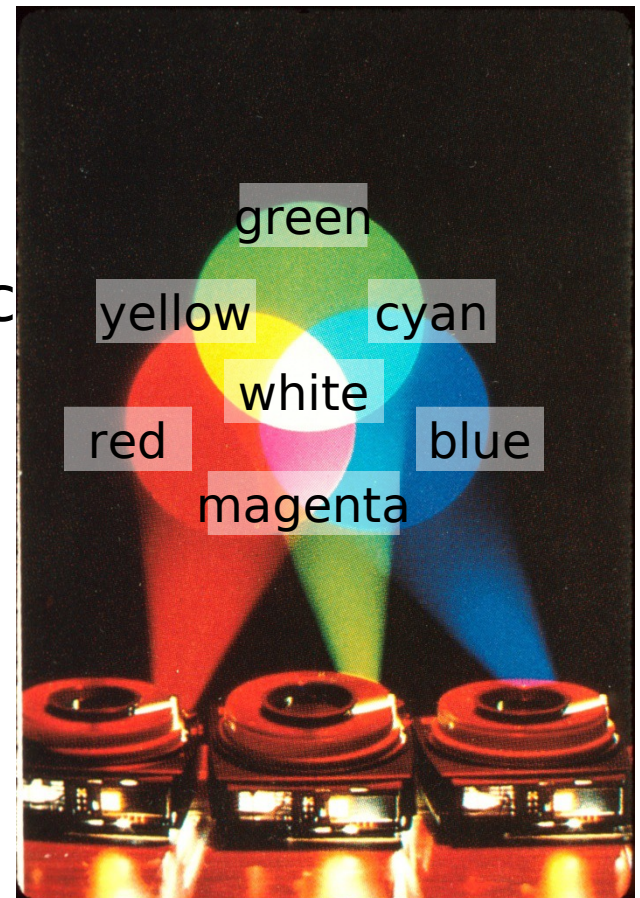
- blend images by sum
- e.g. overlapping projection
- R, G, B make good primaries

$$R=(1,0,0)$$

$$G=(0,1,0)$$

$$B=(0,0,1)$$

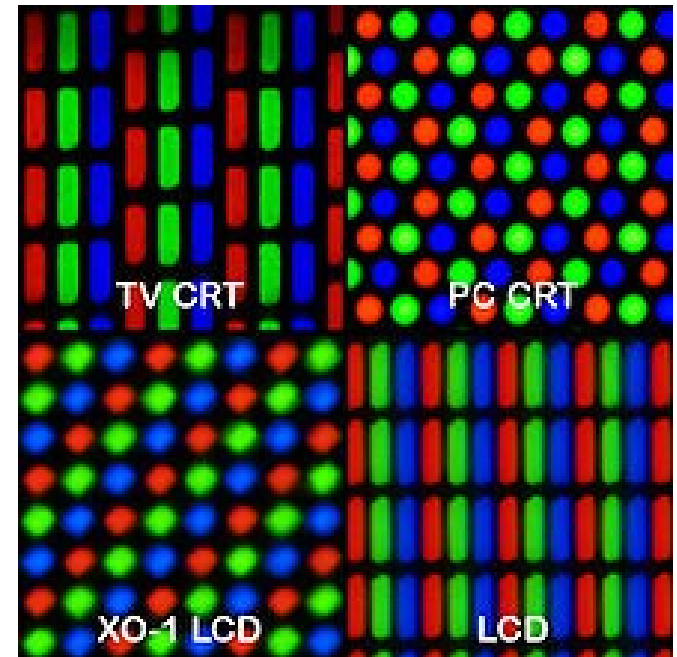
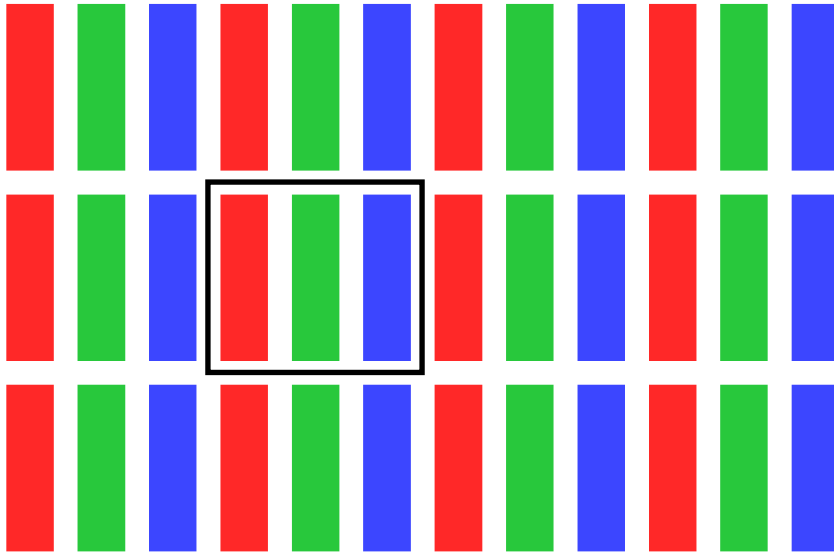
$$\text{Cyan}=G+B=(0,1,1)$$



[source unknown]

Color displays

- interleaved R,G,B pixels (except DLPs)



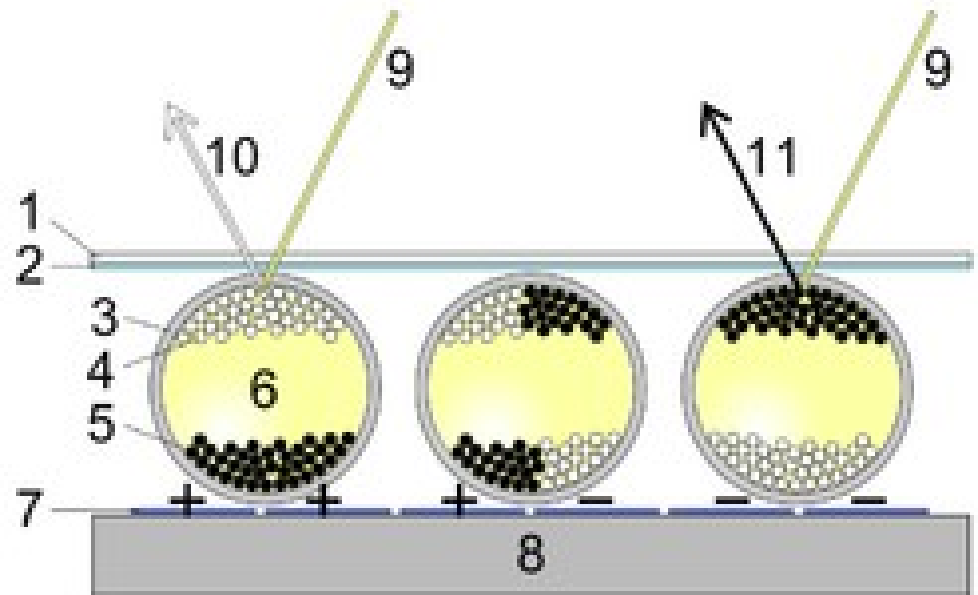
[H&B fig. 2-10]

Electronic paper (E Ink)

- In an extremely bright condition, E-ink outperforms all light-emitting displays

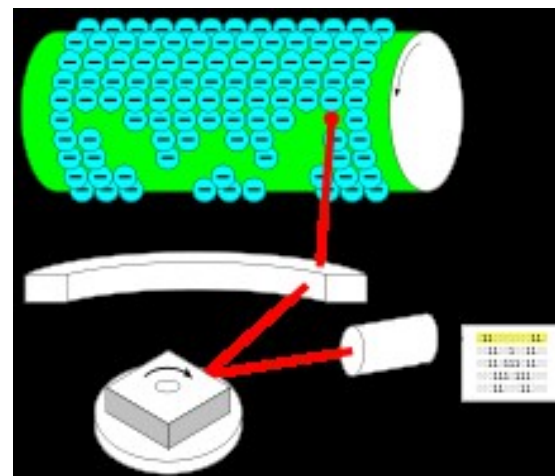
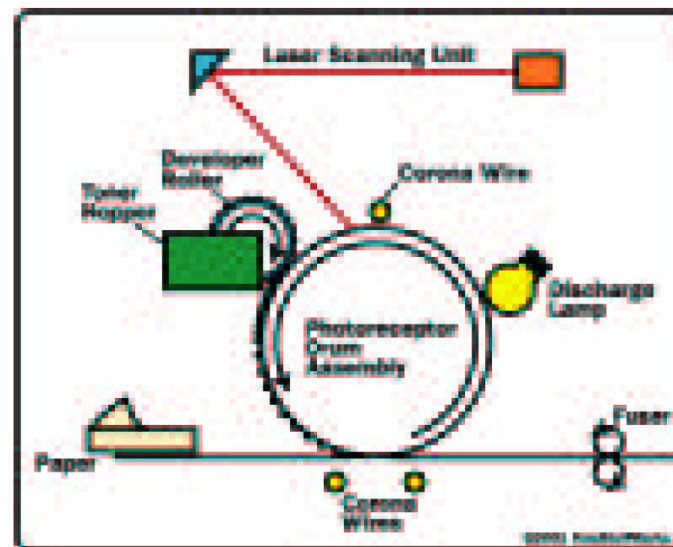


An e-paper display visible in the sunlight



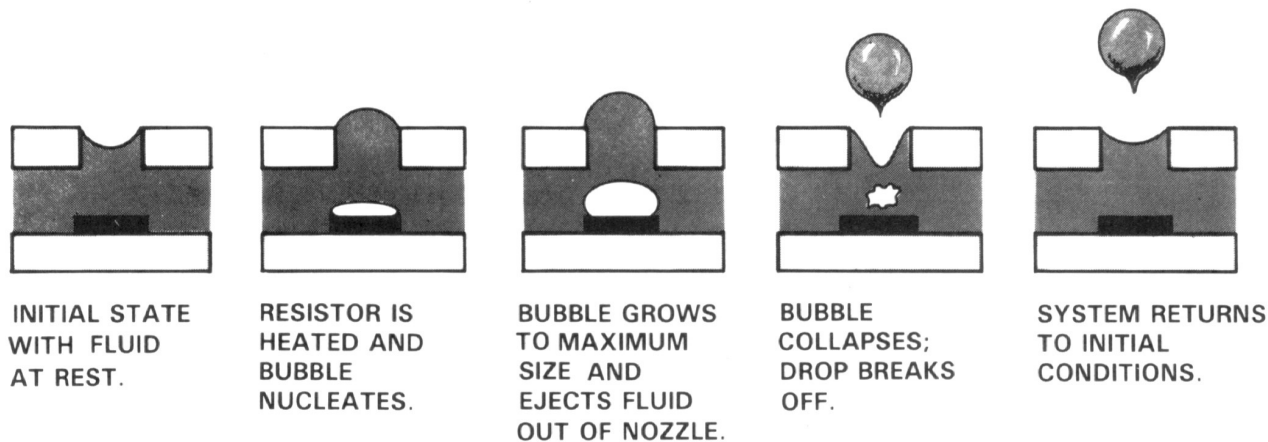
Laser printer

- Like a photocopier but with laser-scanned raster as source image
- Key characteristics
 - image is binary
 - resolution is high
 - very small, isolated dots are not possible



Inkjet printer

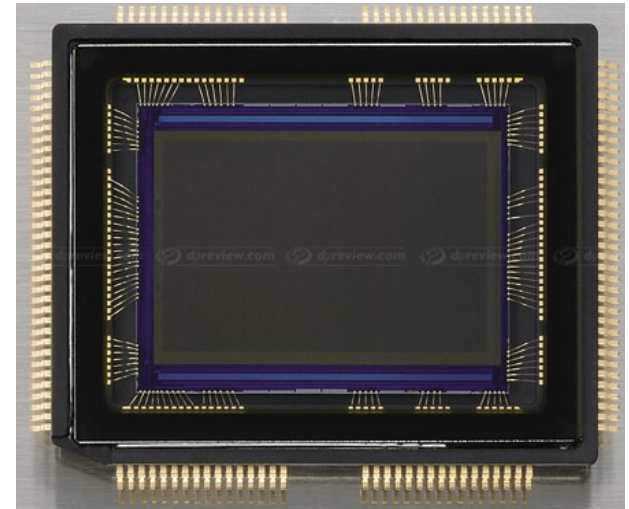
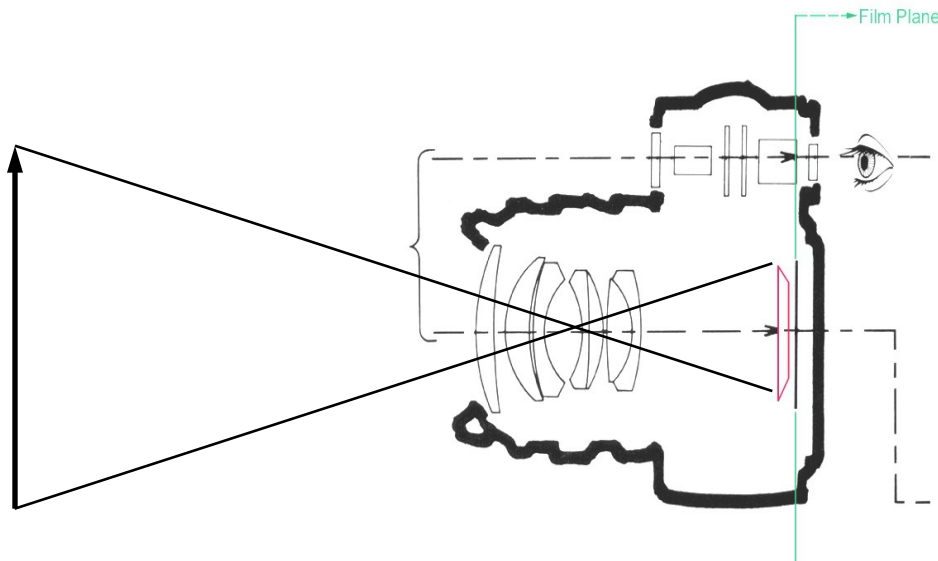
- Liquid ink sprayed in small drops
- Head with many jets scans across paper
- Key characteristics:
 - image is binary (drop or no drop; no partial drops)
 - isolated dots are reproduced well



[source unknown]

Digital camera

- A raster input device
- Image sensor contains 2D array of photosensors

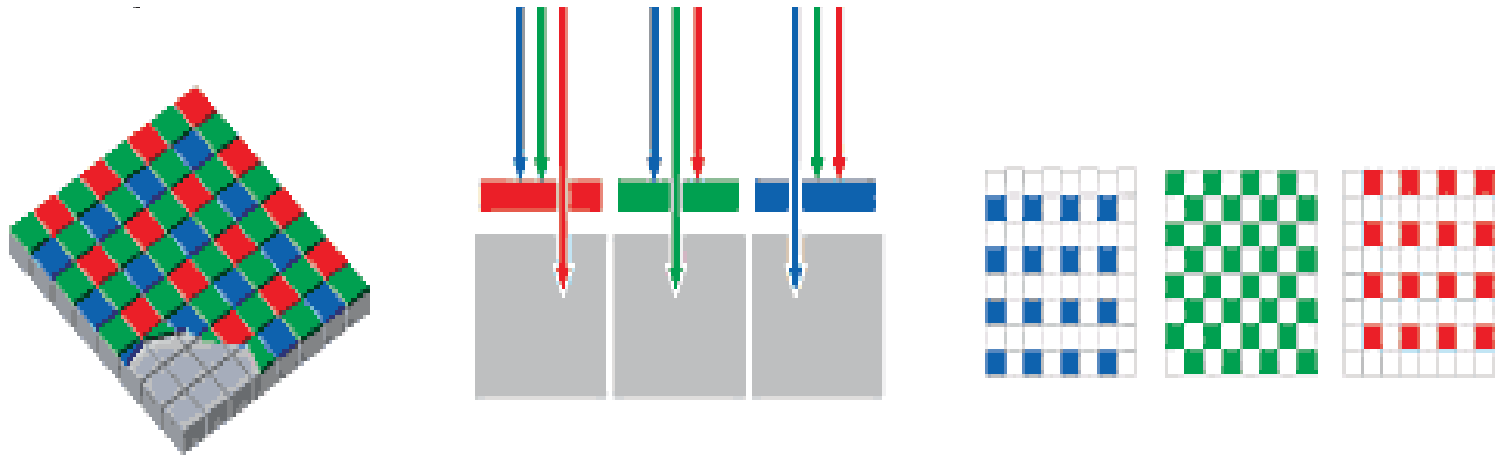


[CS 417 Spring 2002]

[dpreview.com]

Digital camera

- Color typically captured using color mosaic



[Foveon]

Raster image representation (Pixels as opposed to vectors)

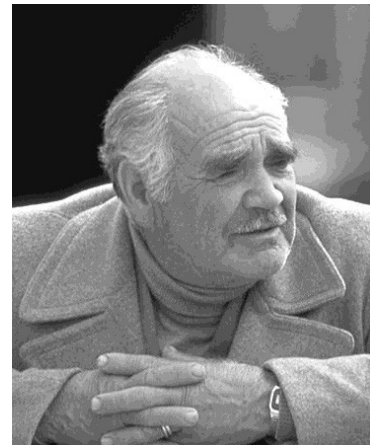
- All these devices suggest 2D arrays of numbers (pixels)
- Big advantage: represent arbitrary images
 - approximate arbitrary functions with increasing resolution
 - works because memory is cheap (brute force approach!)



[Philip Greenspun]

Datatypes for raster images

- Bitmaps: `boolean` per pixel (1 bpp):
 - black and white
- Grayscale: integer per pixel:
 - shades of gray; e.g. black-and-white print
 - precision: usually `byte` (8 bpp); sometimes 10, 12, or 16 bpp
 - E-ink: 4bpp (16 grey levels)
- Color: 3 integers per pixel:
 - full range of displayable color
 - precision: usually `byte[3]` (24 bpp)
 - sometimes 16 (5+6+5) or 30 or 36 or 48 bpp
 - indexed color (palletized color : usually 8bpp): a fading idea. GIF format



Datatypes for raster images

- Floating point:
 - more precision
 - provides *high dynamic range* (HDR)

HDR - 72bpp image

24bpp image

Higher
bits

Lower
bits

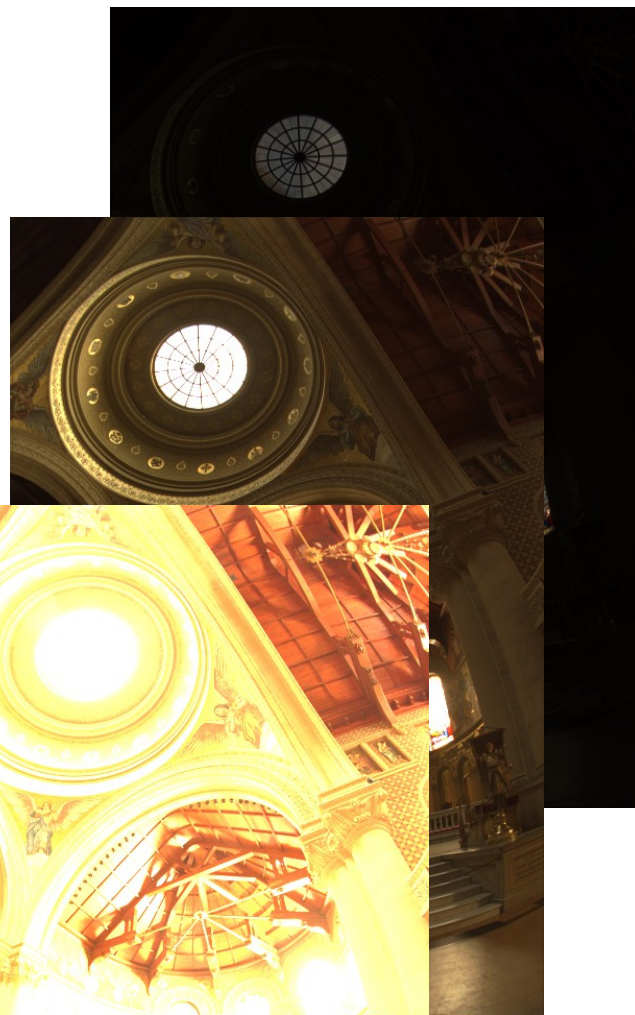
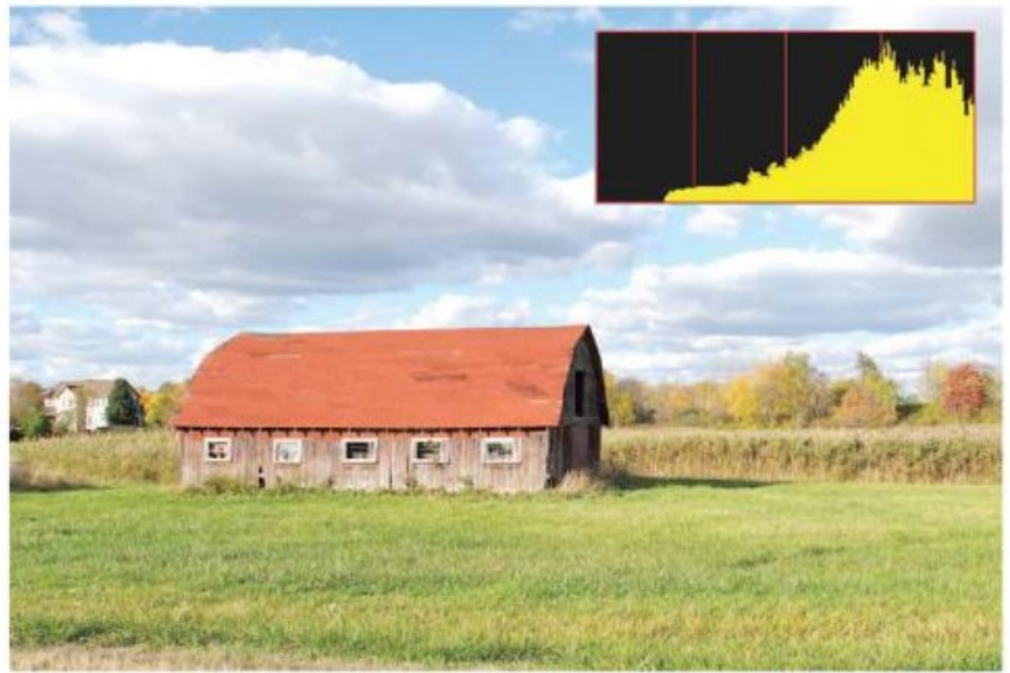
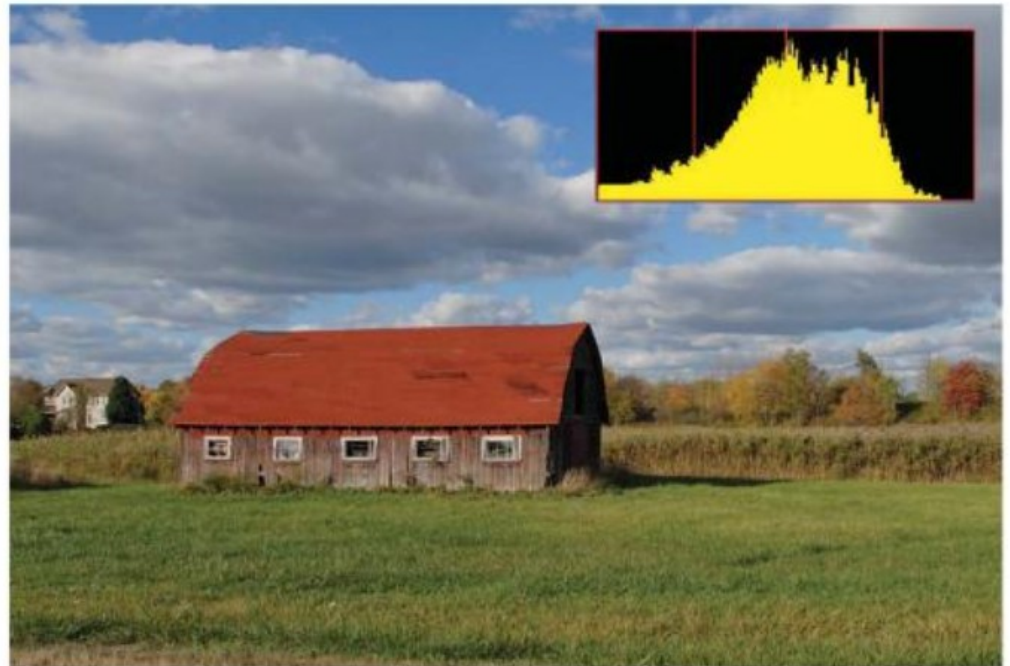


Figure 6.26
This histogram
reveals that the
image is over-
exposed.



HDR images are good for
post-processing
(exposure compensation)
- badly taken photos can
be recovered

Figure 6.27
A histogram for
a properly
exposed image
should look
like this.



exposure:
-8 stops



image: Paul Debevec

exposure:
+0 stops



image: Paul Debevec

exposure:
+6 stops



image: Paul Debevec

Datatypes for raster images

- Floating point:
 - more precision
 - provides *high dynamic range* (HDR)
 - represent real scenes independent of display
 - becoming the standard intermediate format in graphics processors

Datatypes for raster images

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



without
and
with
alpha



[Adobe Photoshop sample]

Storage requirements for images

- 1024x1024 image (1 megapixel)
 - bitmap: 128KB
 - grayscale 8bpp: 1MB
 - grayscale 16bpp: 2MB
 - color 24bpp: 3MB
 - floating-point HDR color: 12MB

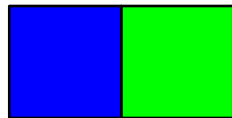
Image processing

- Color to gray
- Down pixel precision (16bpp → 4bpp)
 - Brute-force
 - Dithering
- Gamma correction

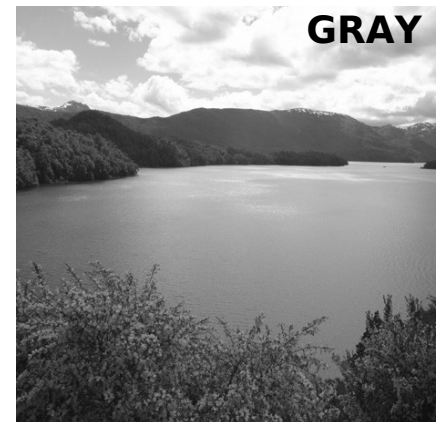
Converting pixel formats

- 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: $\text{gray} = 0.2 R + 0.7 G + 0.1 B$
 - more on this in color, later on

Same pixel values.

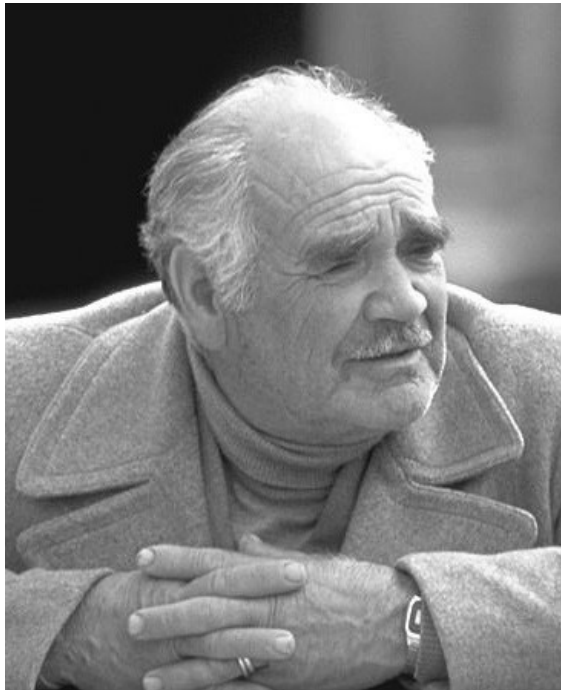


Same luminance?



Converting pixel precision

- Up is easy; down loses information—be careful

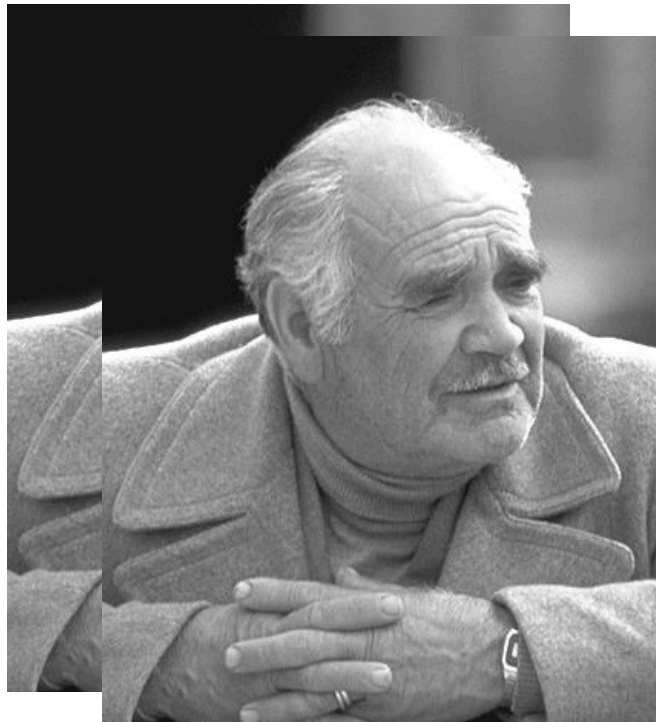


8 bpp (256
grays)

[photo: Philip
Greenspun]

Converting pixel precision

- Up is easy; down loses information—be careful



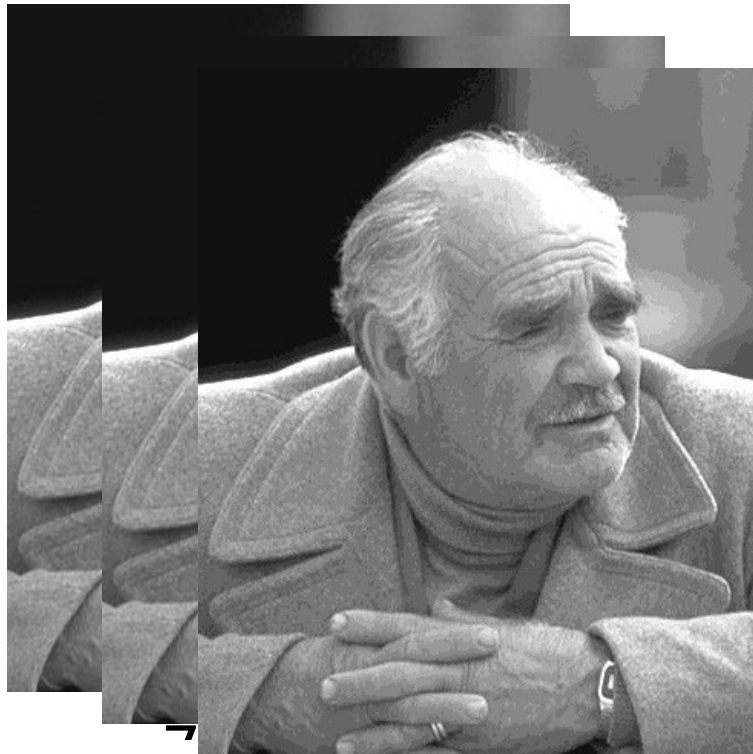
5 bpp (32 grays)

~ grays,

[photo: Philip
Greenspun]

Converting pixel precision

- Up is easy; down loses information—be careful

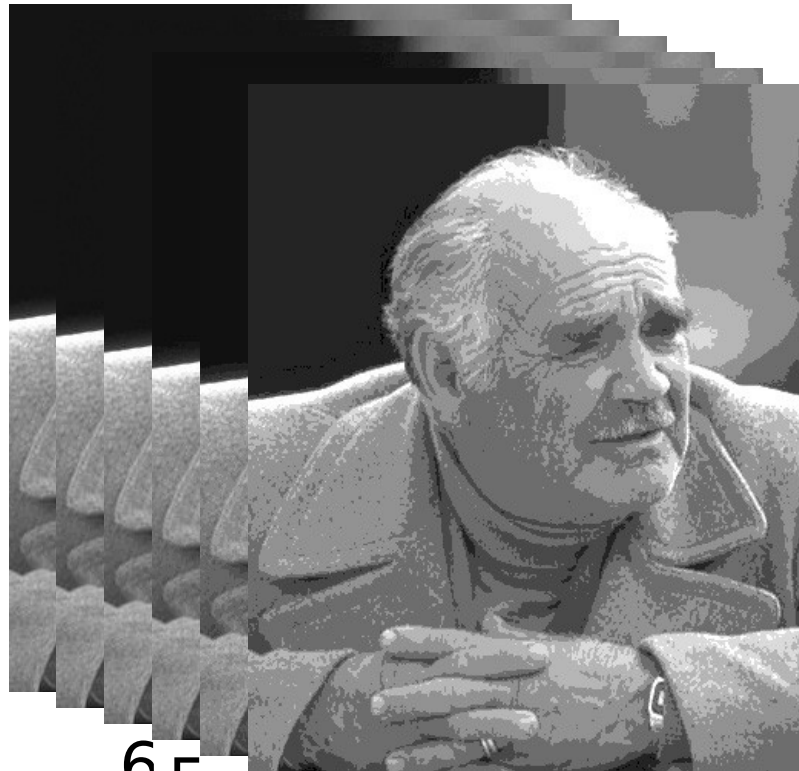


54 bpp (16 grays)

[photo: Philip
Greenspun]

Converting pixel precision

- Up is easy; down loses information—be careful

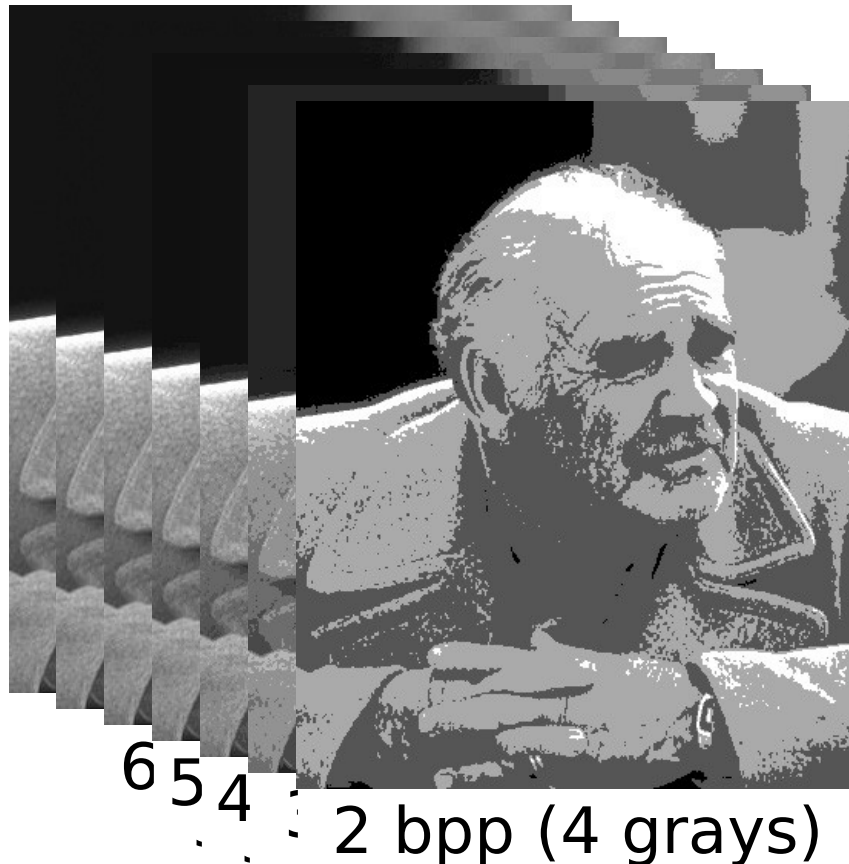


6 5 4 3 bpp (8 grays)

[photo: Philip
Greenspun]

Converting pixel precision

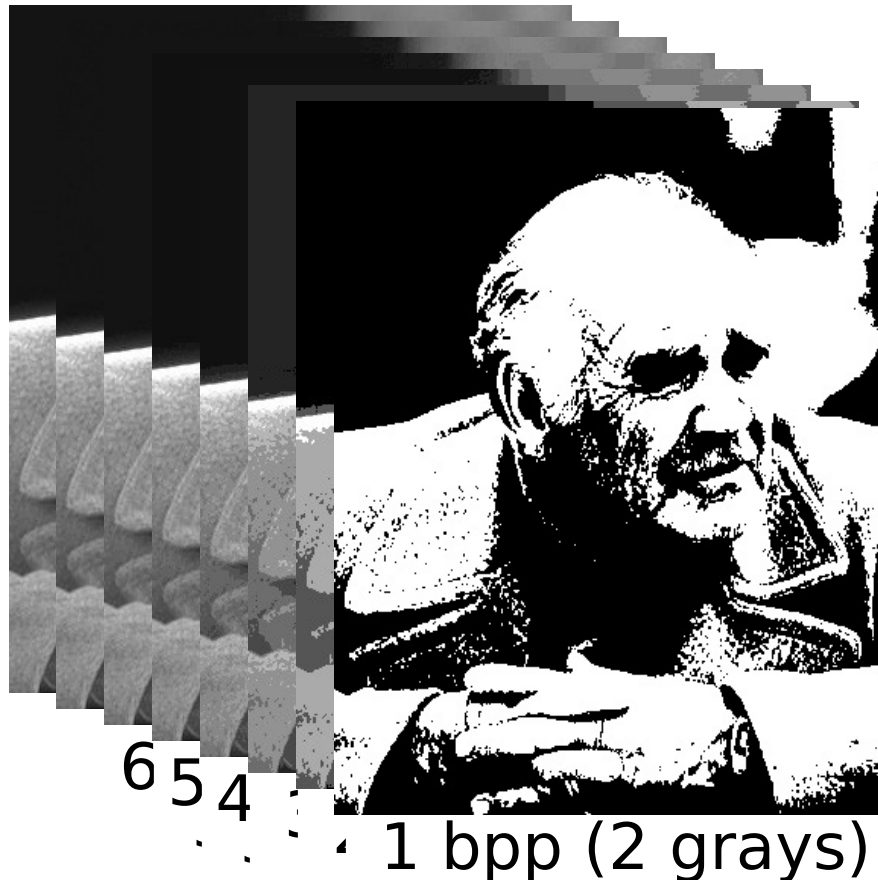
- Up is easy; down loses information—be careful



[photo: Philip Greenspun]

Converting pixel precision

- Up is easy; down loses information—be careful



Lacks
gradation

[photo: Philip
Greenspun]

Dithering

- When decreasing bpp, we quantize
- Make choices consistently:
banding artifacts
 - $I = 0$ if $I < 0.5$
 - $I = 1$ if $I \geq 0.5$
- Instead, be inconsistent—dither
 - turn on some pixels but not others in gray regions
 - a way of trading spatial for tonal resolution

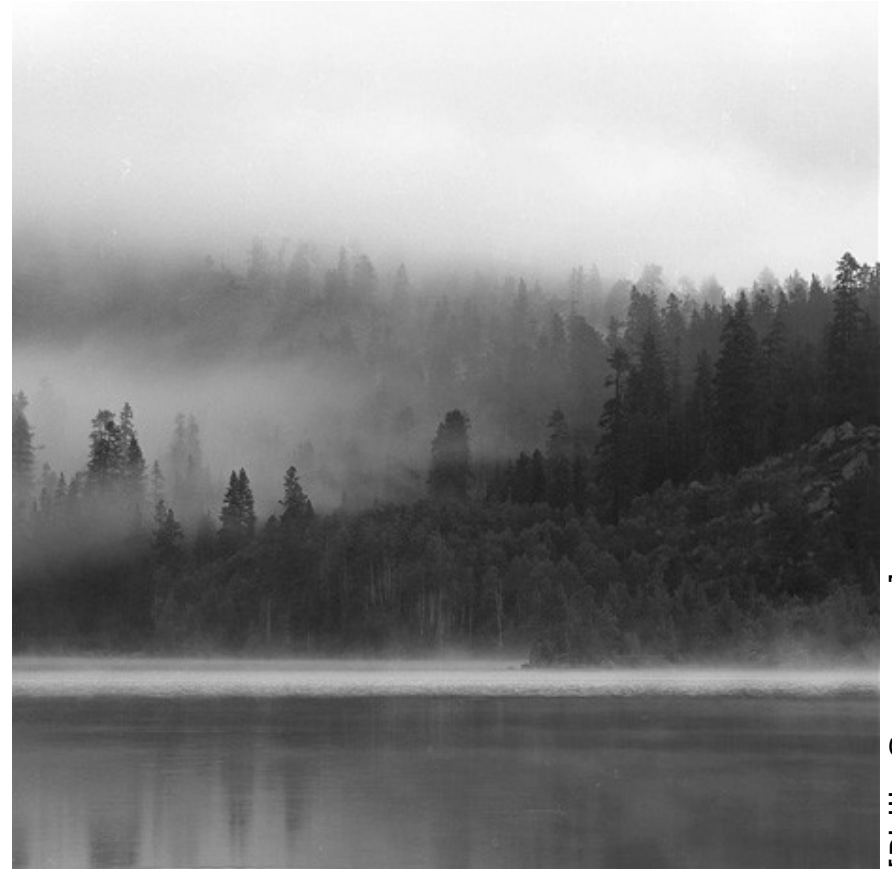


Laser printed image
WITHOUT dithering

[Philip Greenspun]

Dithering methods

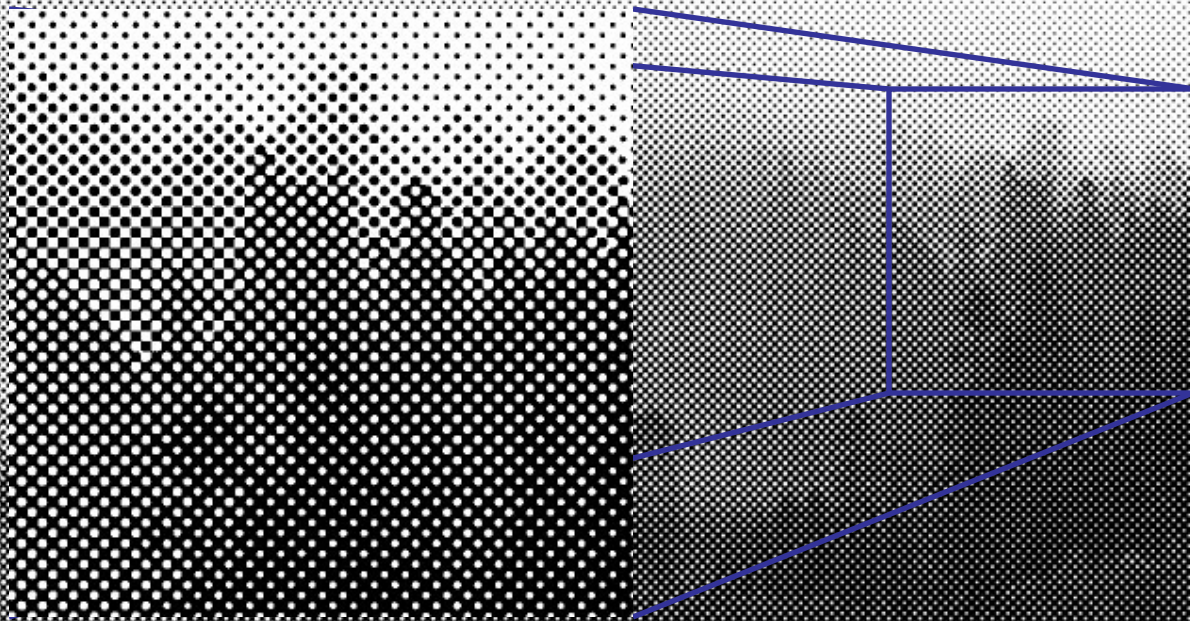
- Ordered dither
 - produces larger dots
 - laser printer
- Diffusion dither
 - takes advantage of devices that can reproduce isolated dots
 - inkjet, screen



[Philip Greenspun]

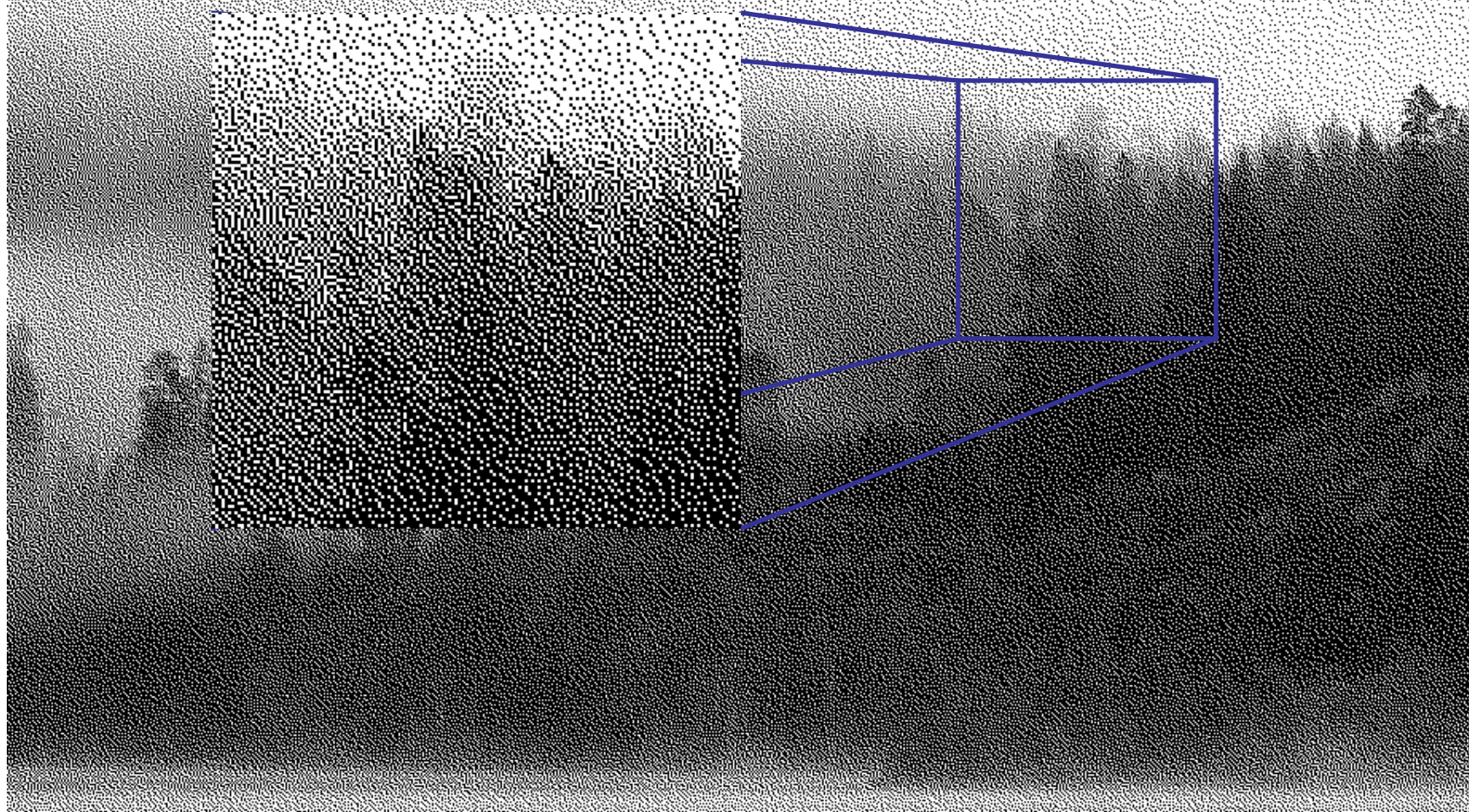
Ordered Dither example

- Produces regular grid of compact dots



Diffusion dither

- Produces scattered dots with the right local density

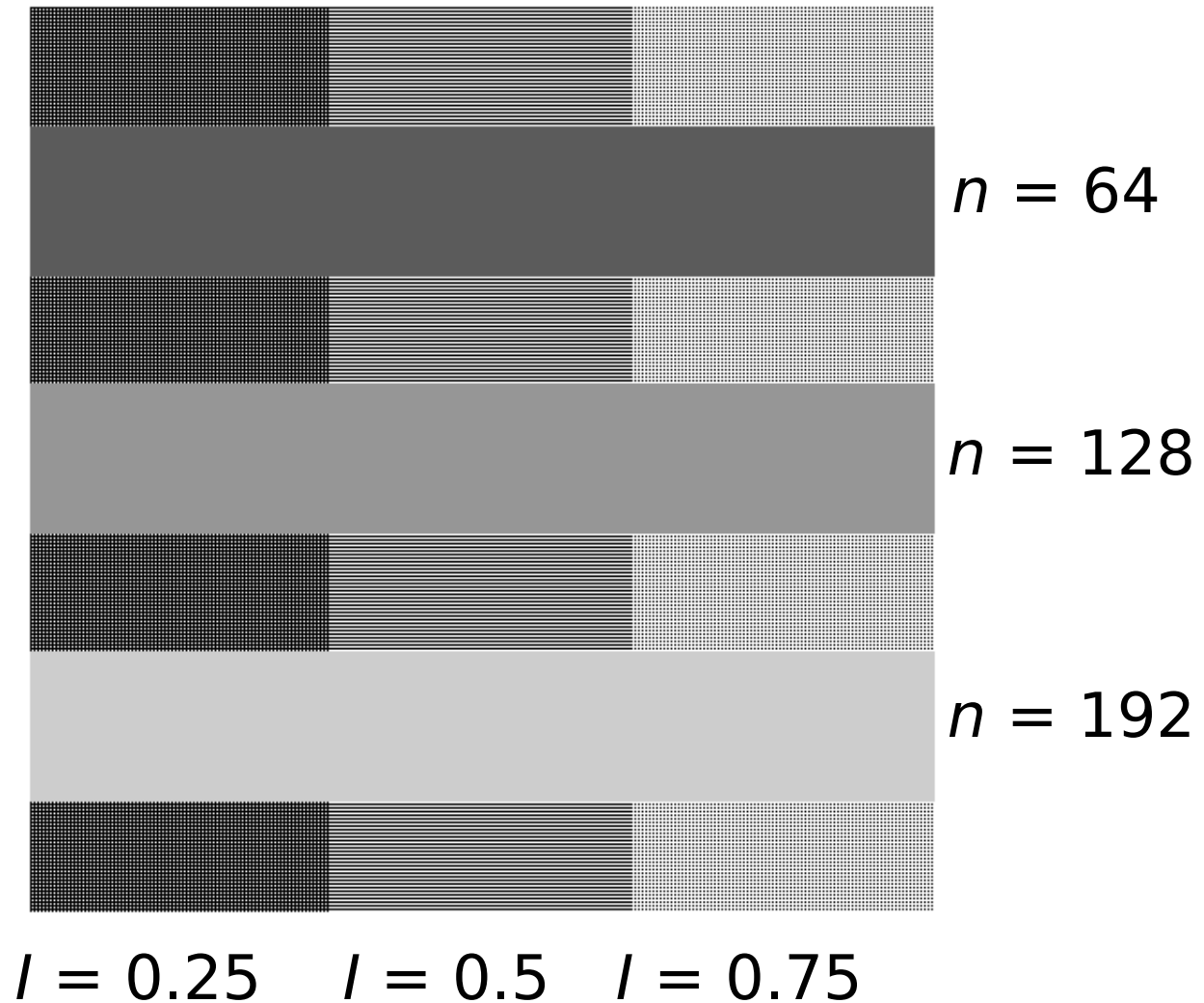


[photo: Philip Greenspun]

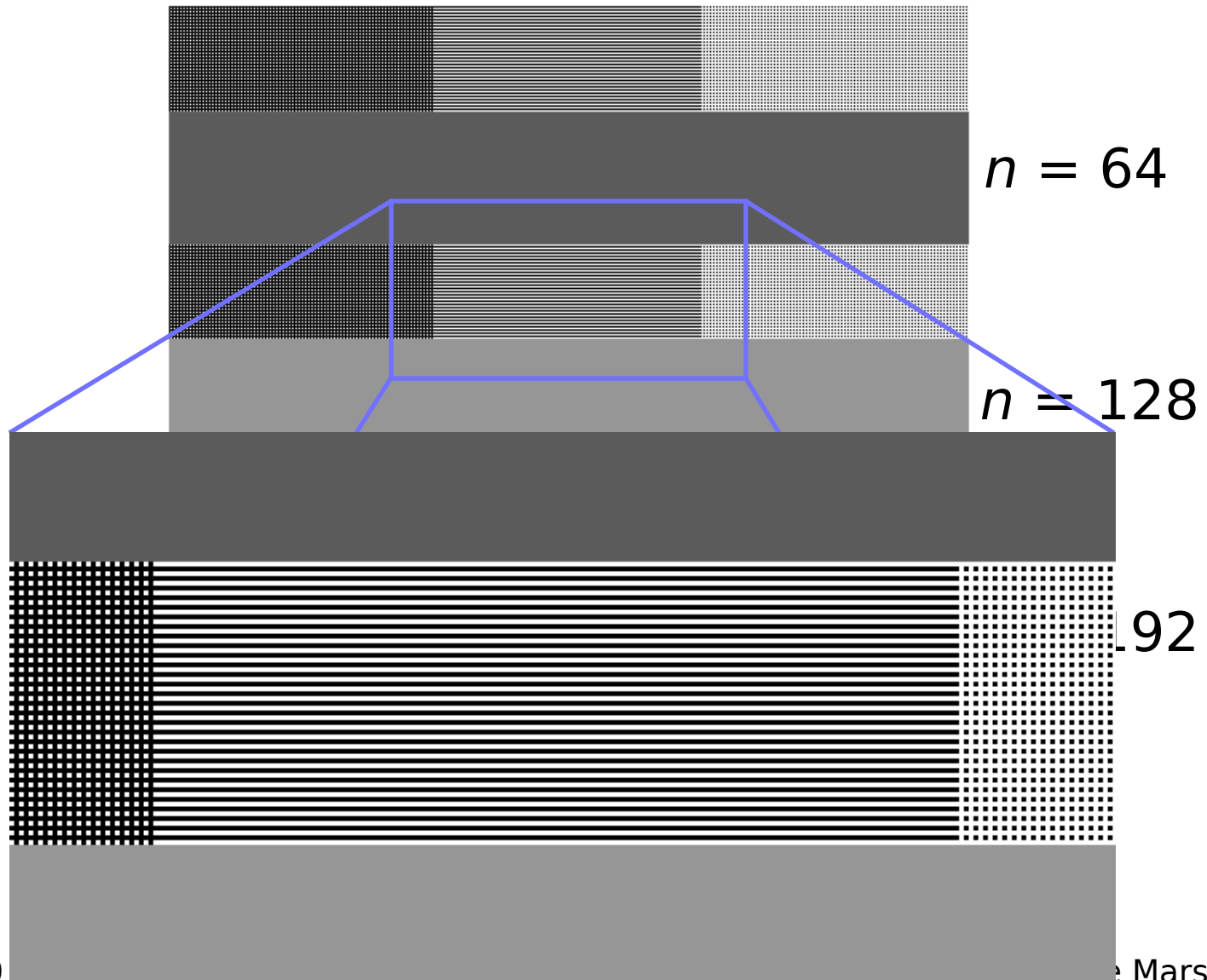
Pixel Values vs Light Intensity

- What do the numbers in images (pixel values) mean?
 - they determine how bright that pixel is
 - bigger numbers are (usually) brighter
- Is a pixel with value 200 twice as bright as another pixel with value 100?

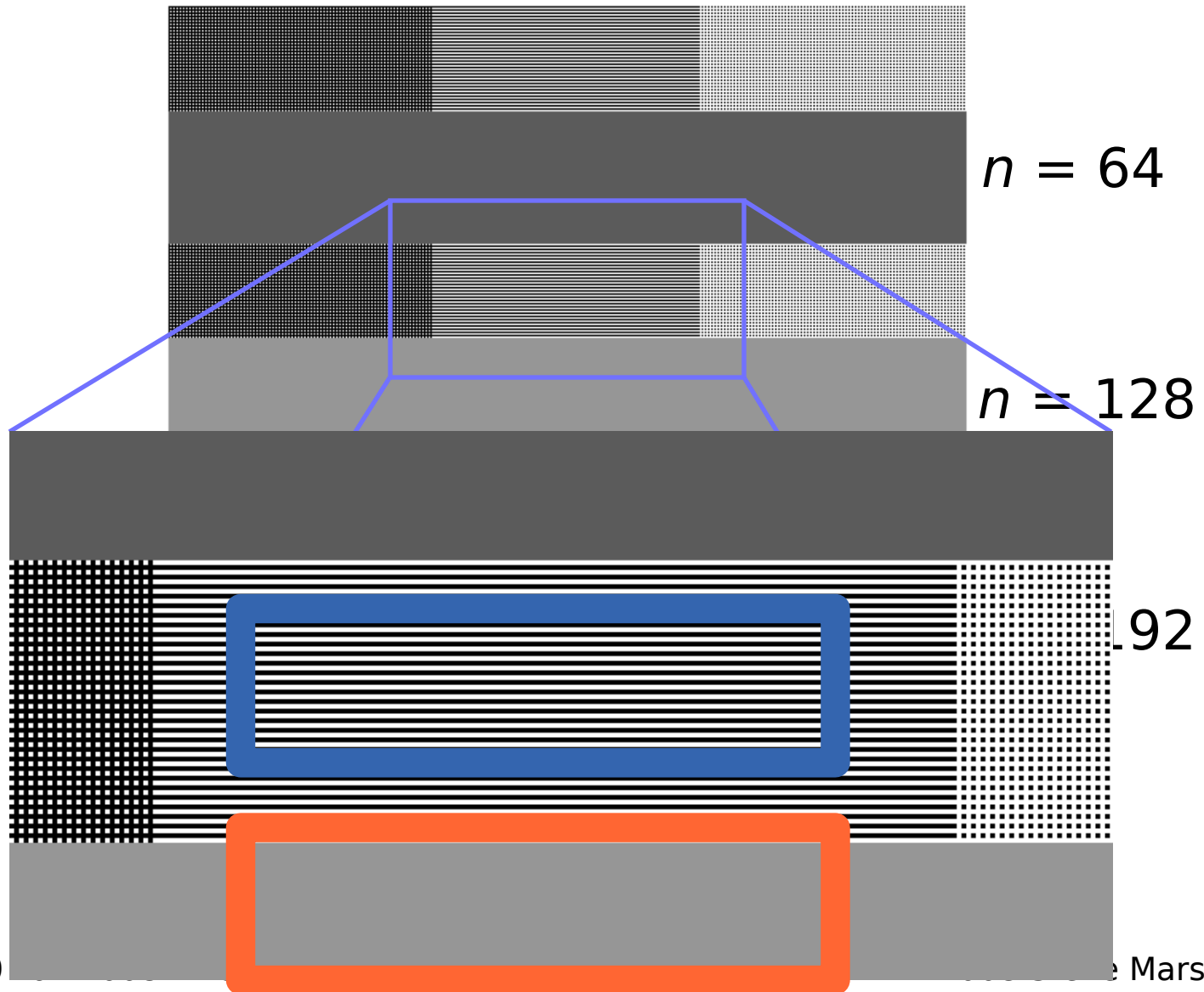
What this projector does



What this projector does



What this projector does



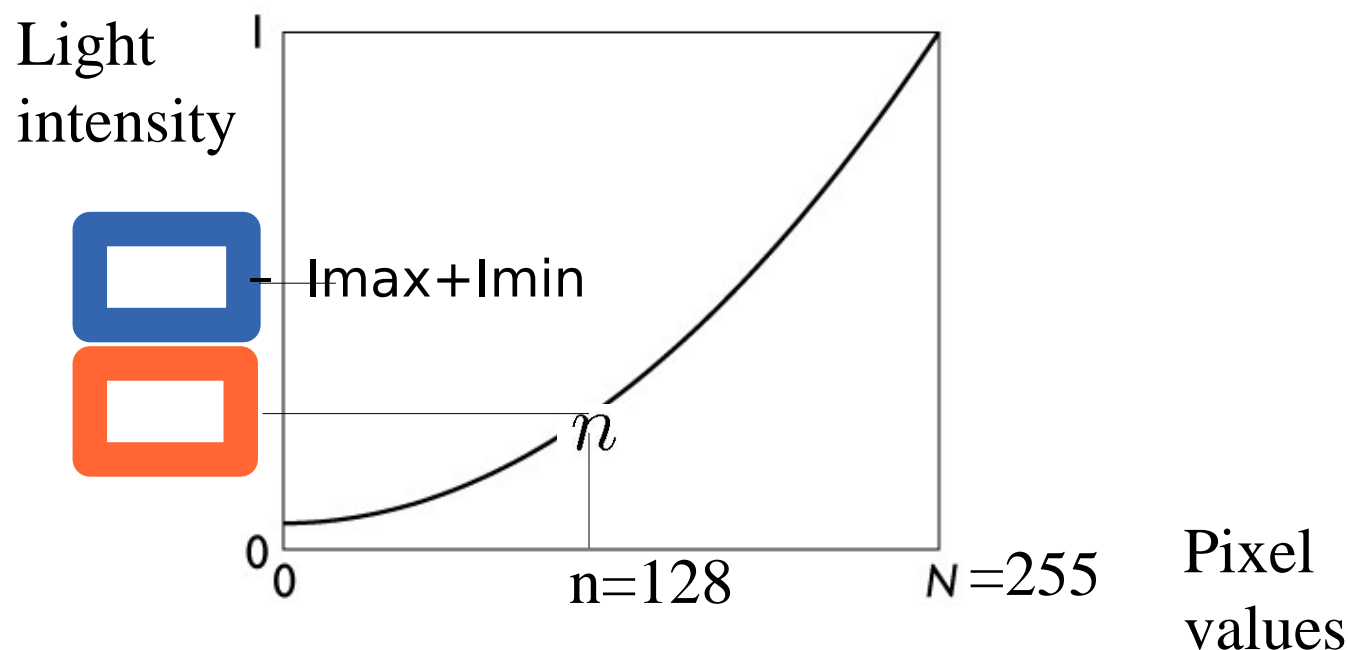
What this projector does

- Something like this if the blue boxed region look brighter than the red box (n=128:)



- Why?

- Boxed region: $I = 0.5 (I_{\max} + I_{\min})$
- $I(128) < 0.5 (I_{\max} + I_{\min})$

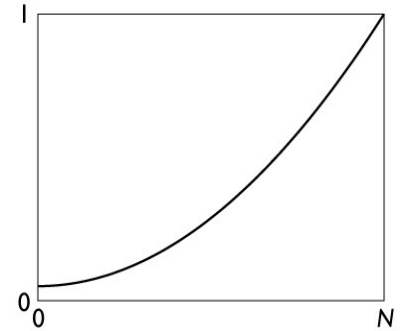


What do the numbers in images (pixel values) mean?

- they determine how bright that pixel is
 - bigger numbers are (usually) brighter
- *Transfer function*: function that maps input pixel value to luminance of displayed image

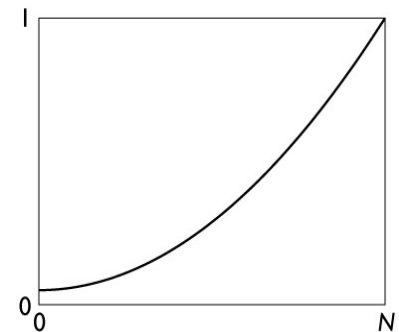
$$I = f(n) \quad f : [0, N] \rightarrow [I_{\min}, I_{\max}]$$

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



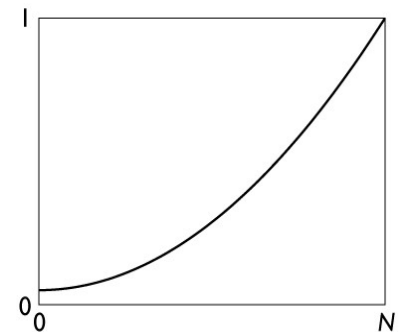
Constraints on transfer function (for light-emitting displays)

- Maximum displayable intensity, I_{\max}
 - 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, I_{\min}
 - 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

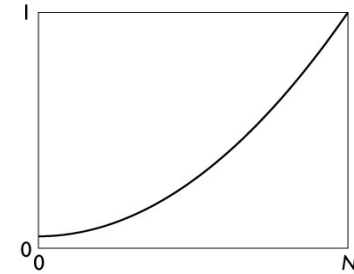
- very important factor determining image contrast in practice
 - 5% of I_{\max} is typical in a normal office environment [sRGB spec]
 - much effort to make very black CRT and LCD screens
 - all-black decor in movie theaters



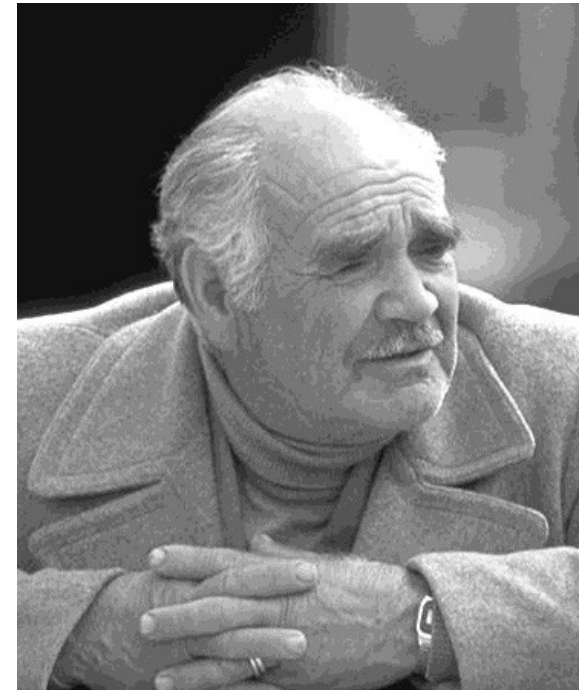
Dynamic range

- 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
 - Desktop display in a dark room: 1000:1 ~ inf:1
 - Photographic transparency (directly viewed): 1000:1
 - High dynamic range display in good conditions: 10,000:1 (very bright white)

Transfer function shape



- 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



an image with severe
banding

[Philip Greenspun]

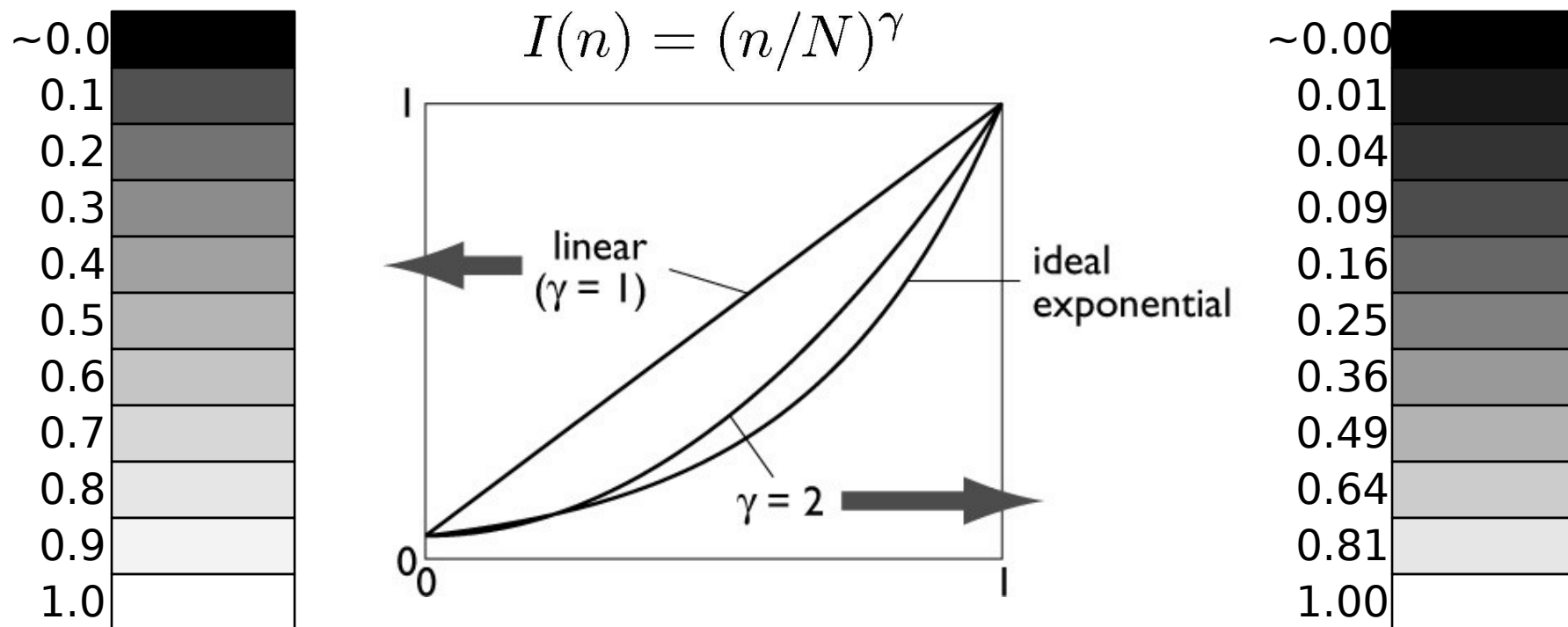
How many levels are needed?

- Ideal case : exponential quantization
 - Depends on dynamic range
 - 2% steps are most efficient:
$$0 \mapsto I_{\min}; 1 \mapsto 1.02I_{\min}; 2 \mapsto (1.02)^2 I_{\min}; \dots$$
 - $\log 1.02$ is about $1/120$, so 120 steps are needed for 10 times intensity
 - 240 steps are needed for 100:1 dynamic range
- Moral: 8 bits is just barely enough for low-end applications
 - even 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: Ideal exponential quantization $I(n) = (1.02)^n I_{\min}$
 - Too expensive
- Option 3: 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 (standard gamma: 2.2)
 - 8 bits are OK for many applications

Option3 : 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,

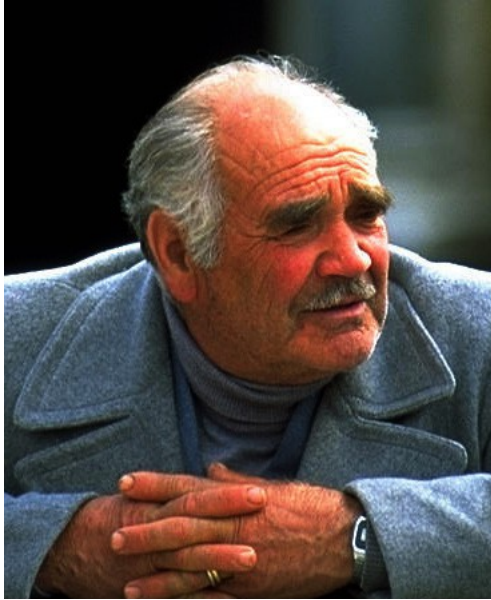
$$I(n) = (n/N)^\gamma$$

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

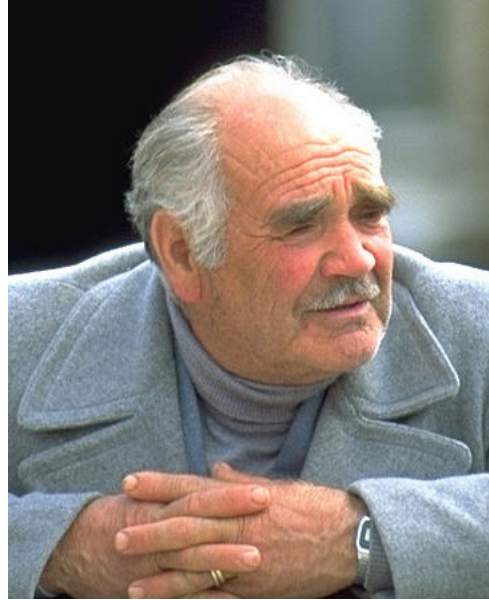
$$n = Na^{\frac{1}{\gamma}}$$

- This is the “gamma correction” recipe that has to be applied when computed values are converted to 8 bits for output
 - failing to do this (implicitly assuming gamma = 1) results in dark, oversaturated images

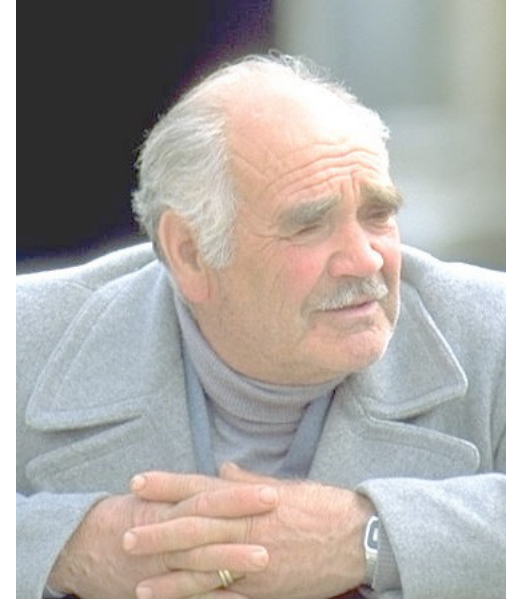
Gamma correction



corrected for
 γ lower than
display



OK



corrected for
 γ higher than
display

[Philip Greenspun]