Video Compression

INSTRUCTOR: YAN-TSUNG PENG

DEPT. OF COMPUTER SCIENCE, NCCU

CLASS 2

Math Background

- Video files are **large**, requiring efficient compression techniques to reduce storage and bandwidth while maintaining quality.
- **Linear Algebra and Probability** play a crucial role in achieving this.
- Linear Algebra in Video Compression
  - Transforms & Compression Algorithms
    - **Discrete Cosine Transform (DCT)** → Used in **JPEG, MPEG, H.264** to convert spatial data into frequency components.
  - Matrix Representations & Factorization
    - Videos are stored as **pixel matrices**, and linear algebra helps **optimize storage** by eliminating redundancy, using techniques like **Principal Component Analysis (PCA)** to achieve dimensionality reduction.
- Probability in Video Compression
  - **Entropy Encoding**
    - Used in **Huffman coding** & **Arithmetic coding** to assign shorter codes to more frequent data values, improving compression efficiency.

Array/Matrix Operations

Array Product

- Pixel-by-pixel basis

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$$

Matrix Product

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

Linear Operations

an image

- Consider a general operator: H

$$H\left[f(x, y)\right] = g(x, y)$$

- H: linear
  - satisfies the principles of **linearity**:
    - **Additivity:** The response to the sum of two inputs equals the sum of their individual responses.
    - **Homogeneity (Scaling):** Scaling the input scales the output by the same factor.

$$H\left[a_i f_i(x, y) + a_j f_j(x, y)\right] = a_i H\left[f_i(x, y)\right] + a_j H\left[f_j(x, y)\right]$$

f_z

$$= a_i g_i(x, y) + a_j g_j(x, y)$$

g_z

- Example: $\sum_x \left(3f(x) + 4g(x)\right)$: $\sum_x$ is a linear operation

$$= 3\sum_x f(x) + 4\sum_x g(x)$$

Nonlinear Operations

- H: nonlinear

$$H\left[a_i f_i(x,y) + a_j f_j(x,y)\right] \neq a_i H\left[f_i(x,y)\right] + a_j H\left[f_j(x,y)\right]$$

- Example:

$$f_1 = \begin{bmatrix} 0 & 2 \\ 2 & 3 \end{bmatrix} \quad \text{and} \quad f_2 = \begin{bmatrix} 6 & 5 \\ 4 & 7 \end{bmatrix}$$

$$\max(f_1 - f_2) \neq \max(f_1) - \max(f_2)$$

Linear versus Nonlinear Operations

- Linear $J = I \odot m + Z \odot (1 - m)$

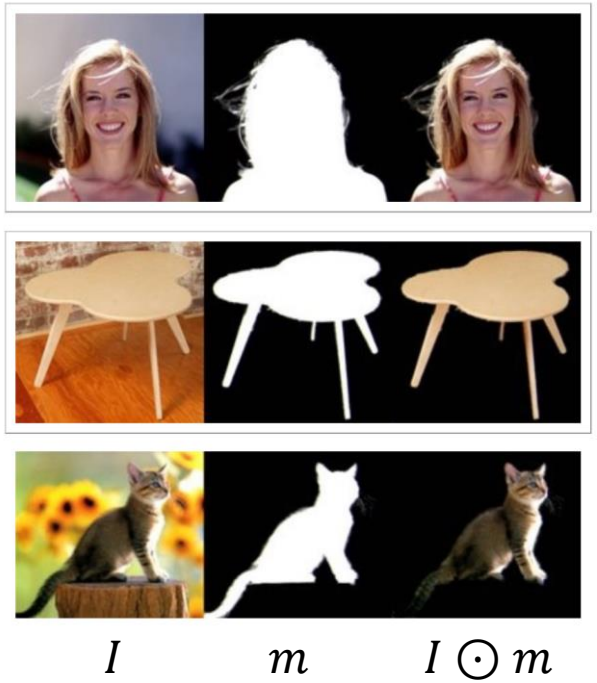
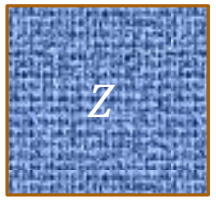


Image Matting



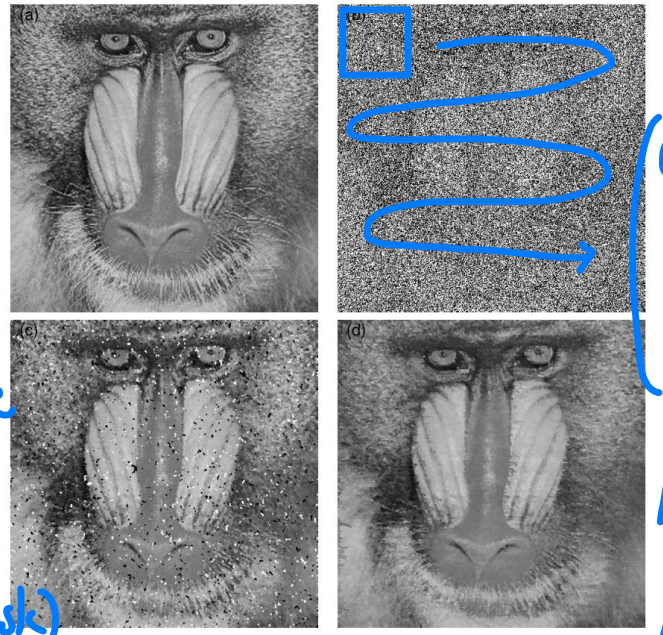
I: original image

Z: background image

m: alpha matte (transparency mask)

$\odot$: element-wise multiplication

- Nonlinear – median filtering



Pepper & Salt Noise Removal

sliding window

(ex. Replacing each pixel w/ the median value of its neighborhood)

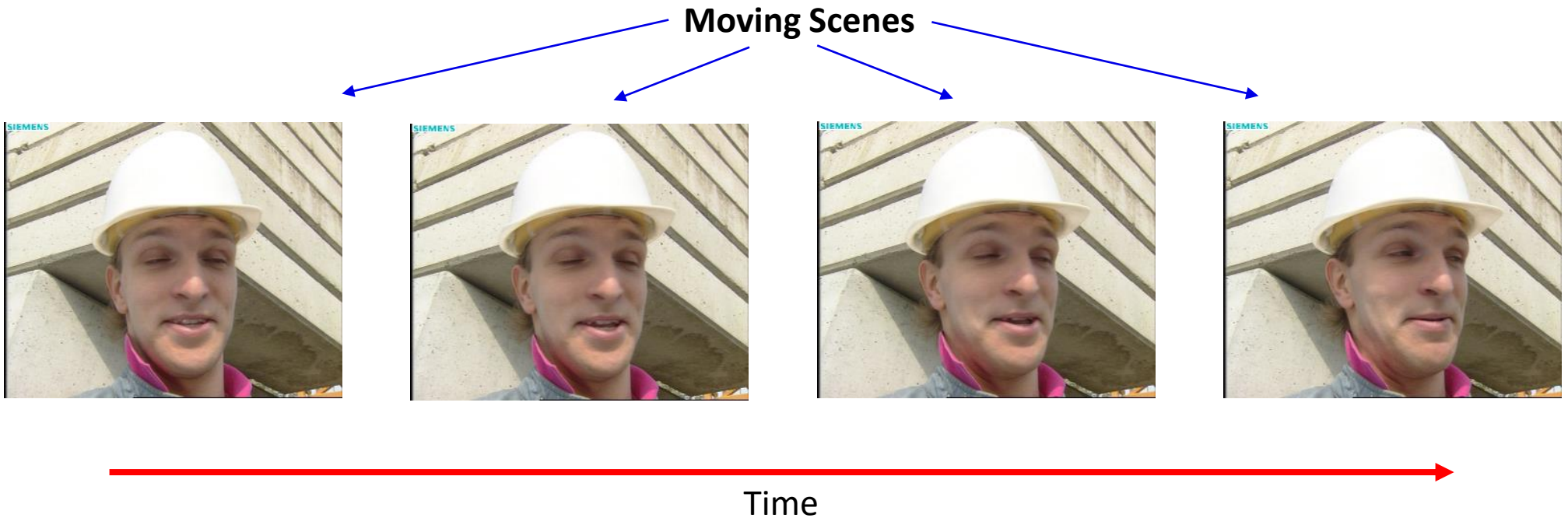
Median($f_1 + f_2$) ≠ Median(f_1) + Median(f_2)

filtered image

Introduction to Video Compression

What is a Video

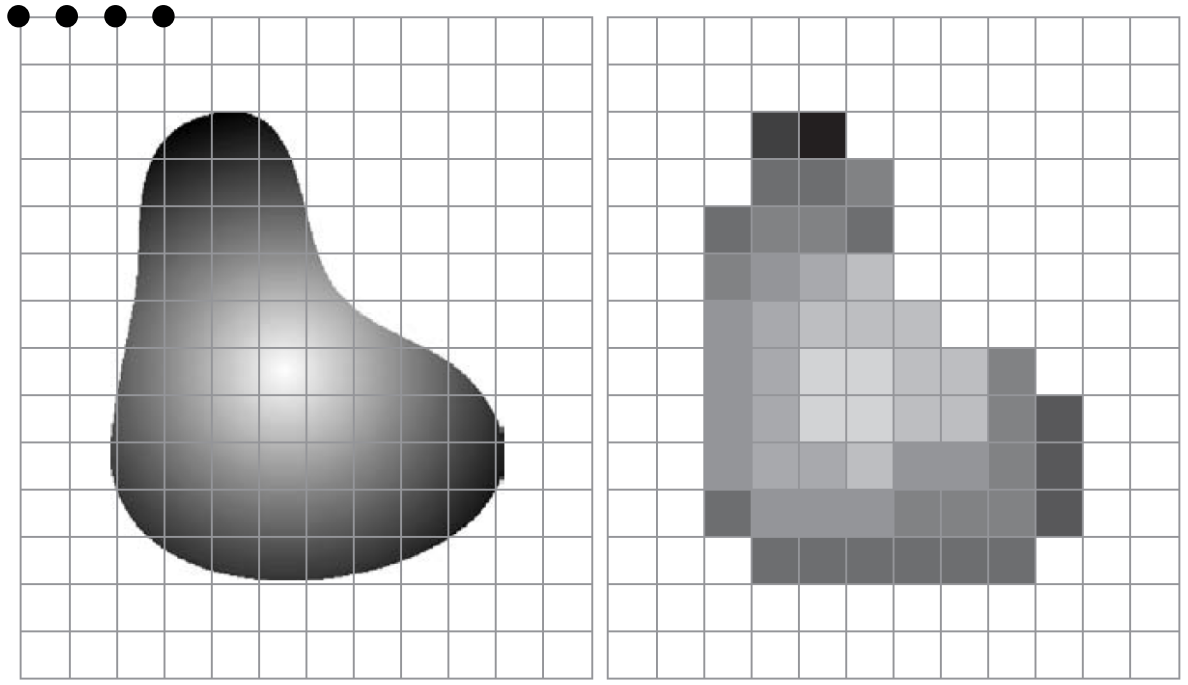
Definition: a video consists of continuous video frames (images) that are usually correlated both spatially and temporally



Digital Video

Sampling grid

Spatial sampling – Frame Resolution



Continuous frame

Sampling & Quantization

Temporal sampling – Frame Rate (frame per second, fps)

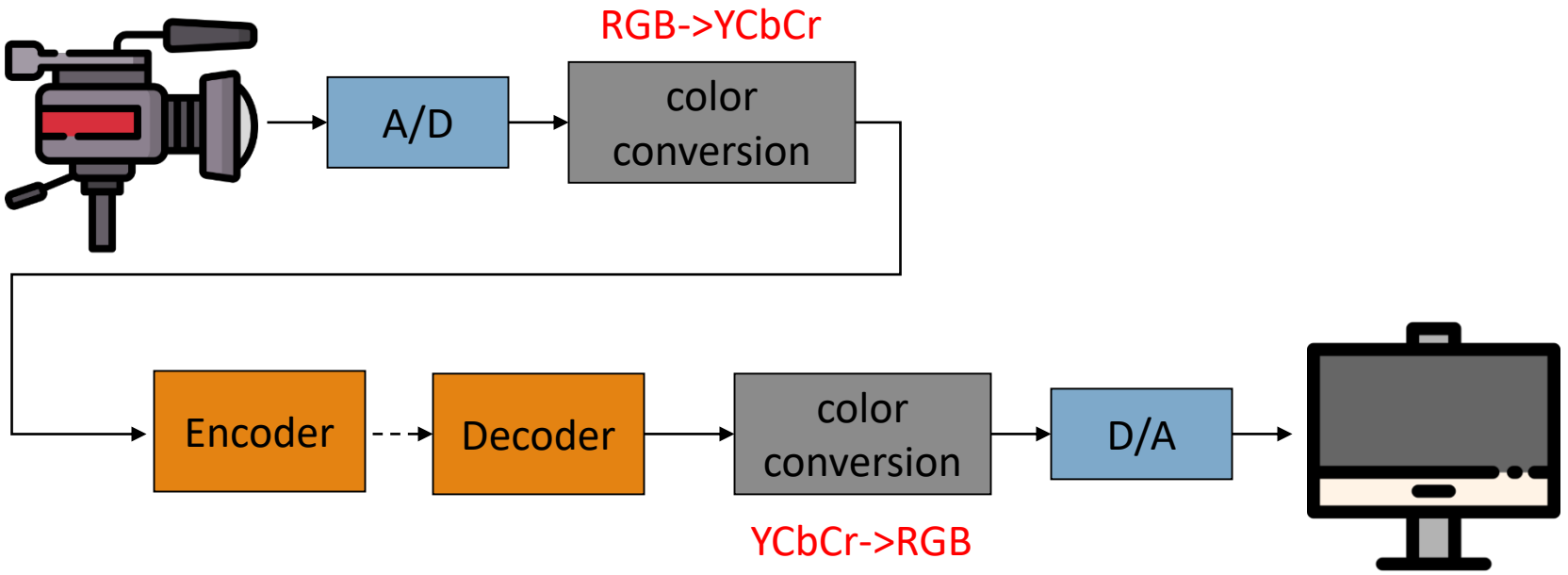
30 fps

15 fps

Spatial Resolution of Images/Videos

Image resolution	# of Sampling points	Analogue video equivalent
352x288	101376	VHS Video
704x576	405504	Broadcast television
1440x1152	1313280	High-definition television
1920x1080	2073600	Full HD (1080p)
3840x2160	8294400	Ultra HD (4K)
7680x4320	33177600	8K UHD
12K (11520x6480)	74649600	12K Digital Cinema

Video Systems



Analog Video Signals

Video format used in different countries worldwide

https://www.sony.com/electronics/support/articles/00006701

- The three primary analog television standards—**NTSC, PAL, and SECAM**—were widely used before the transition to digital broadcasting.

- National Television Systems Committee (NTSC)
  - Introduced in **North America** (1954)
  - **Analog color system** with **interlaced video**
  - **525 scan lines** (each frame has two interlaced fields)
  - **Frame rate: 29.97 fps**

- Phase Alternating Line (PAL)
  - Used in **Europe, Australia, and parts of Asia & Africa**.
  - **625 interlaced scan lines** for better resolution than NTSC
  - **Frame rate: 25 fps**

- SECAM (SEquential Color And Memory)
  - Used in **Eastern Europe, Russia, China, Pakistan, and parts of Africa**
  - **625 interlaced scan lines**, similar to PAL
  - **Frame rate: 25 fps**
  - Unique **color encoding system**, reducing transmission issues in long-dista

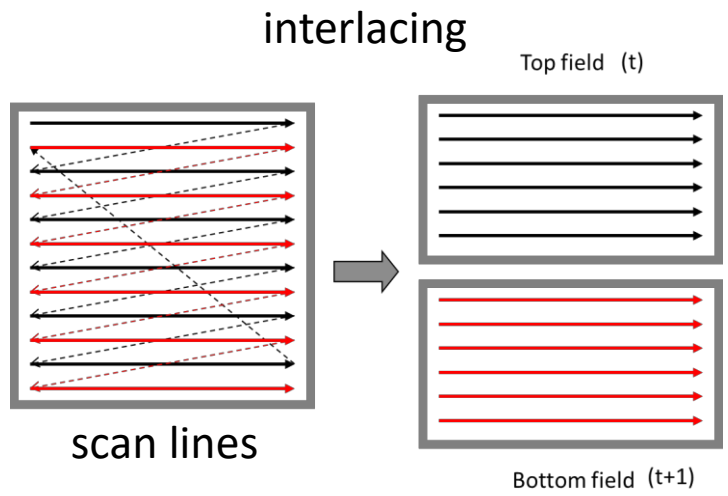
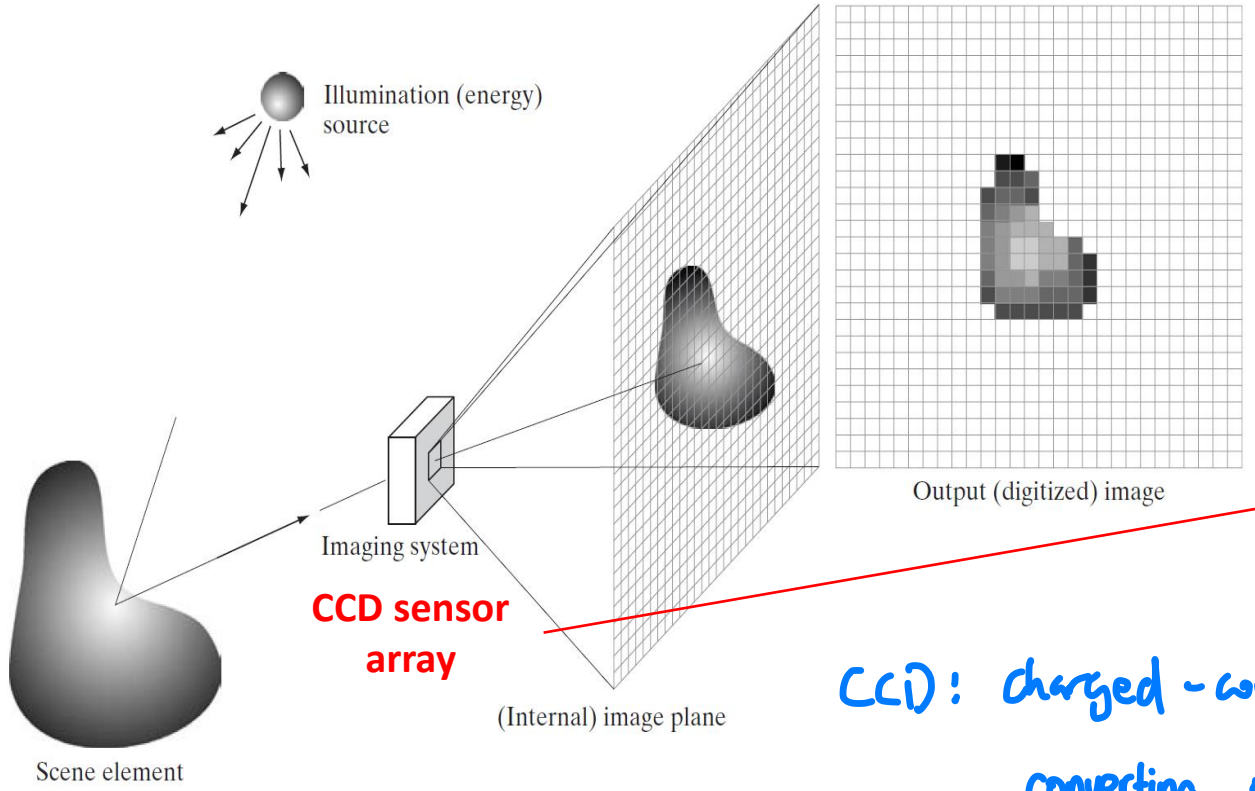
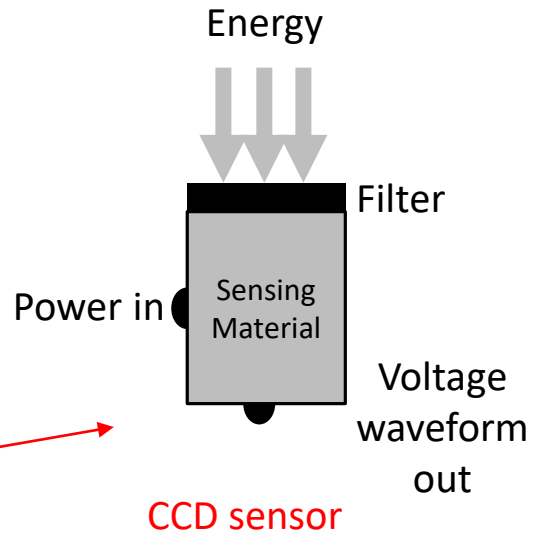


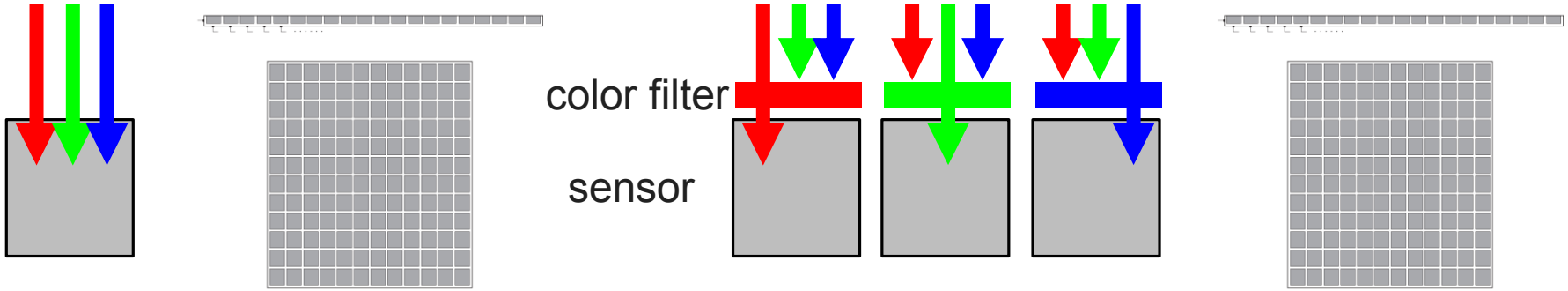
Image Formation Process





CCD: charged-coupled device, a light-sensitive IC converting photons to electrons to capture photos

CCD sensor



Monochrome sensor

Color sensor

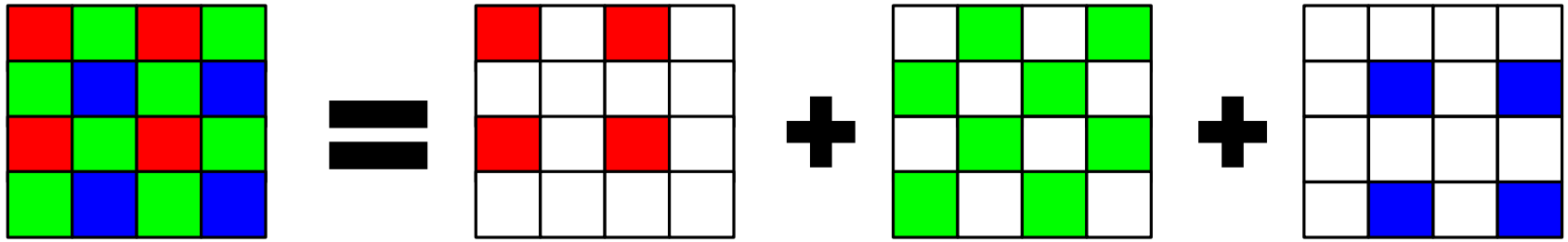
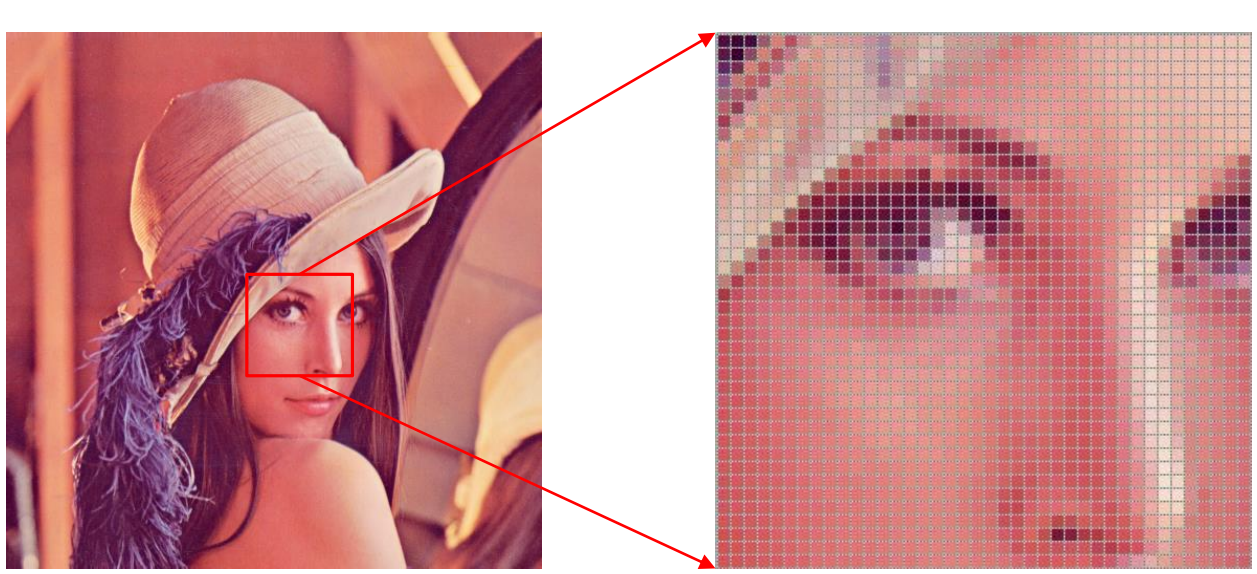
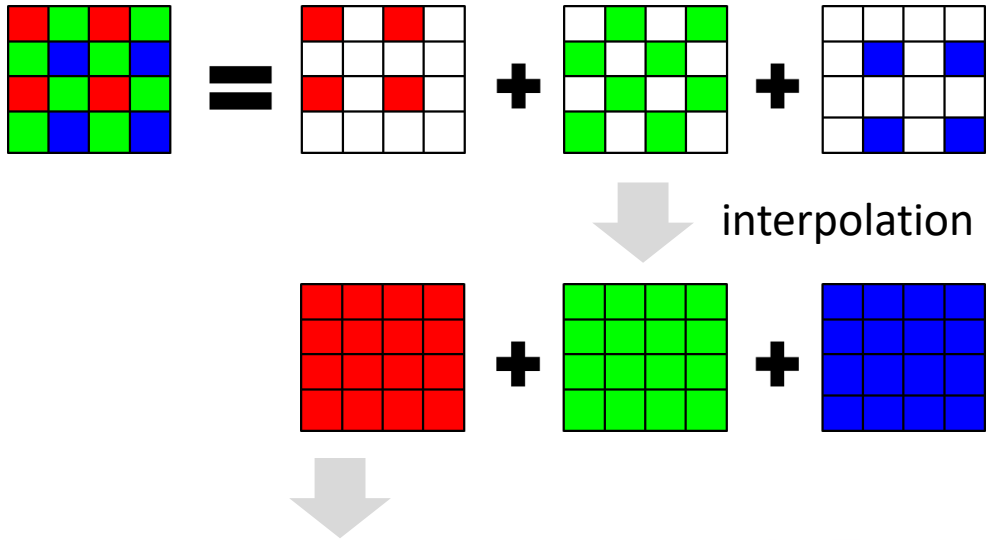
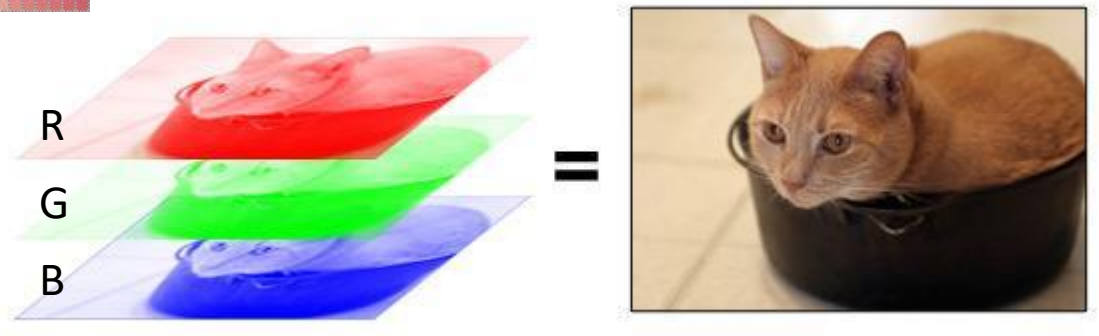


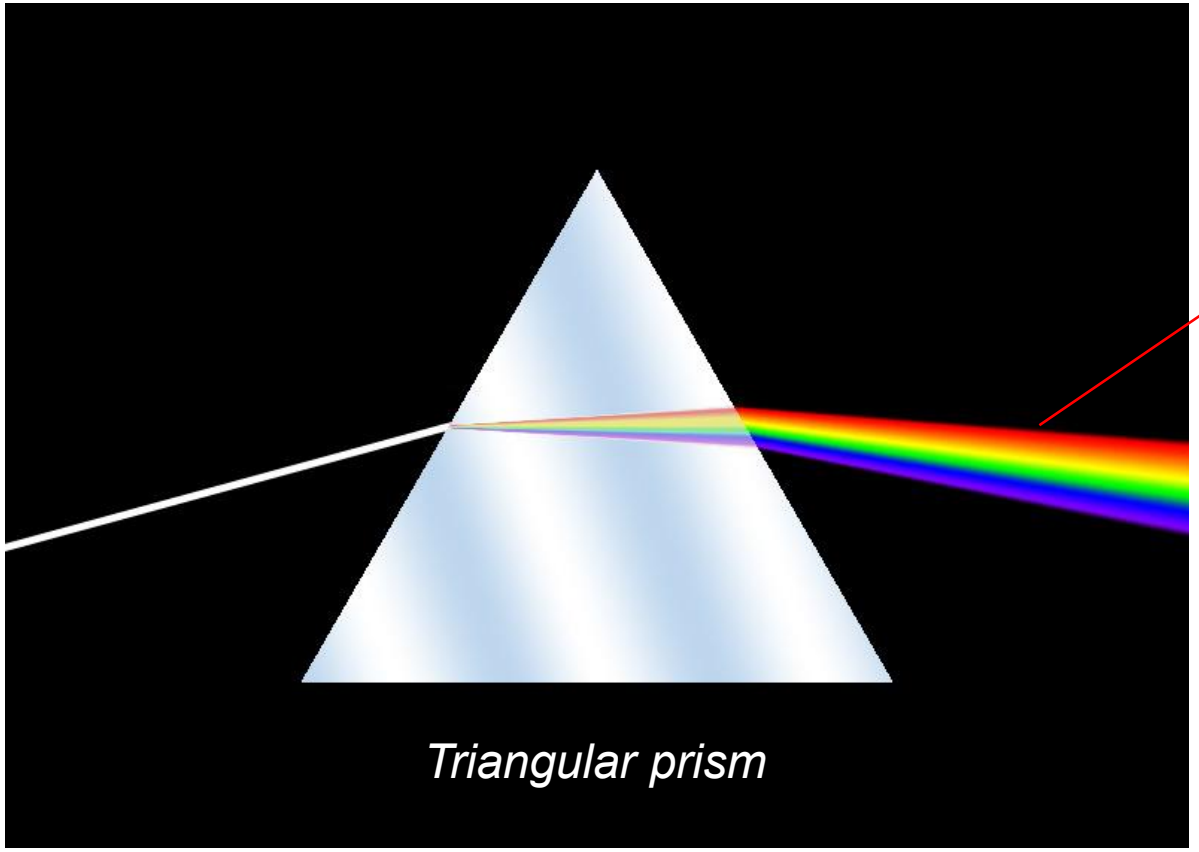
Image Pixel and RGB Color Space







Color



Spectral components of light

The prism deflects the light with different wavelengths at different angles

https://en.wikipedia.org/wiki/Dispersive_prism

Luminous Efficiency Function

Humans can perceive light with wavelength between 380nm and 750nm.

nm: nanometer = 10^{-9} meter

V(λ) describes the human eye's sensitivity to light at different wavelengths in daylight.

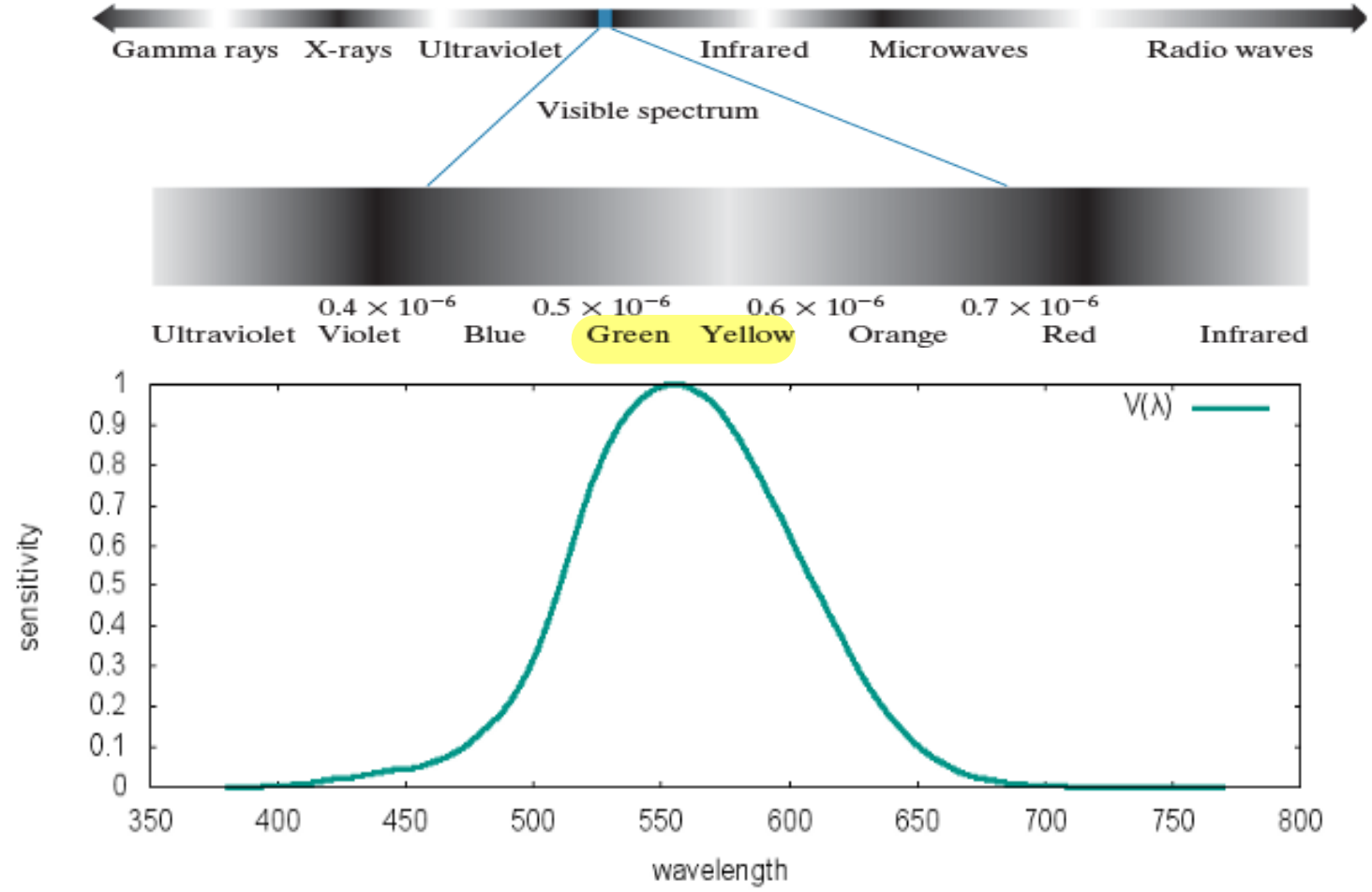


Figure is taken from https://medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a

Color Recreation

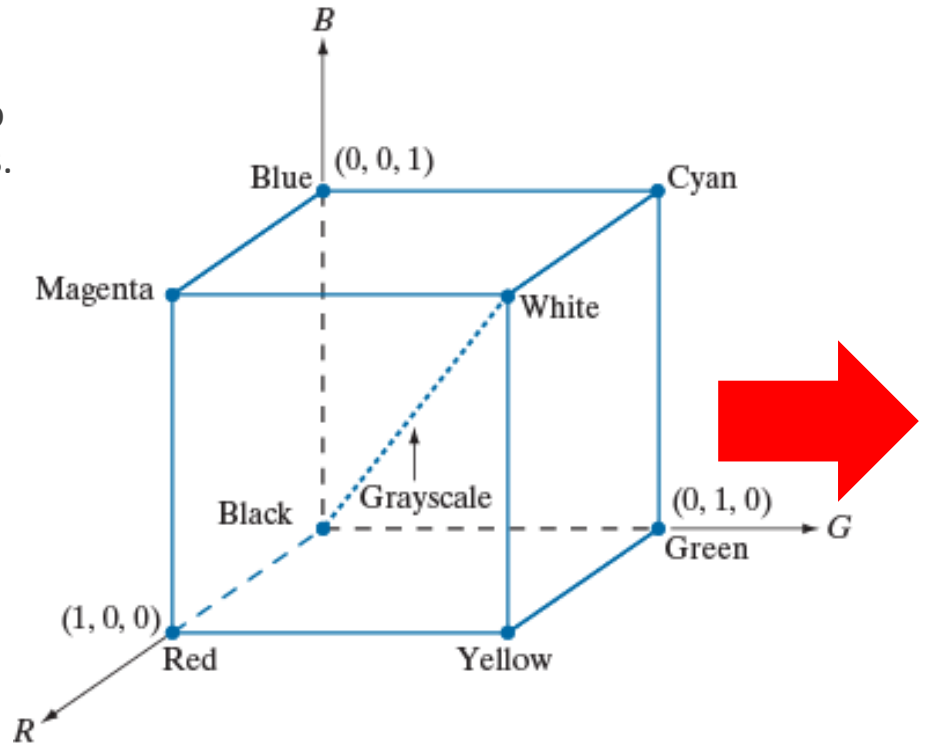
- Why can a display or projector show various colors?
  - Additive color mixing - combining different intensities of primary colors
    - Each pixel or light source in a display or projector adjusts the brightness of the three colors independently.
  - Projectors use separate light sources for each color or filter white light to achieve this.
- Does a projector need to emit light with distinct wavelengths for all the colors it shows?
  - No
  - It typically uses light sources for the three primary colors (red, green, and blue) with distinct wavelengths

Additivity of Color Mixing

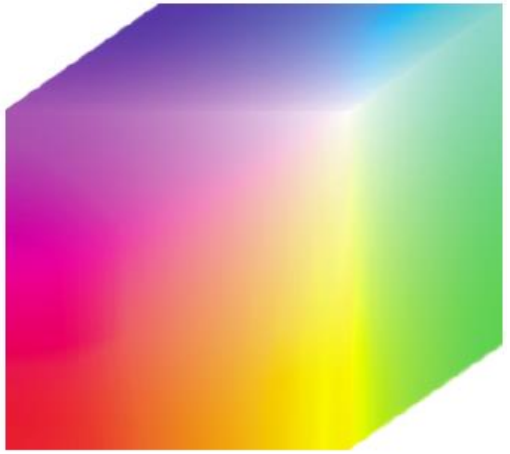
Simulate how a Projector works

Color Space - RGB

- Color space (a.k.a color model or color representation)
  - **RGB Color Space**
  - All visible colors can be broken down into **Red, Green, and Blue (RGB) components**.
  - This forms the basis for digital displays, where colors are created by mixing different intensities of **RGB light**.



RGB Color Cube



Human color perception is a three-dimensional phenomenon

Source: www.ImageProcessingPlace.com

Luminous Efficiency Function

V(λ) describes the human eye's sensitivity to light at different wavelengths in daylight.

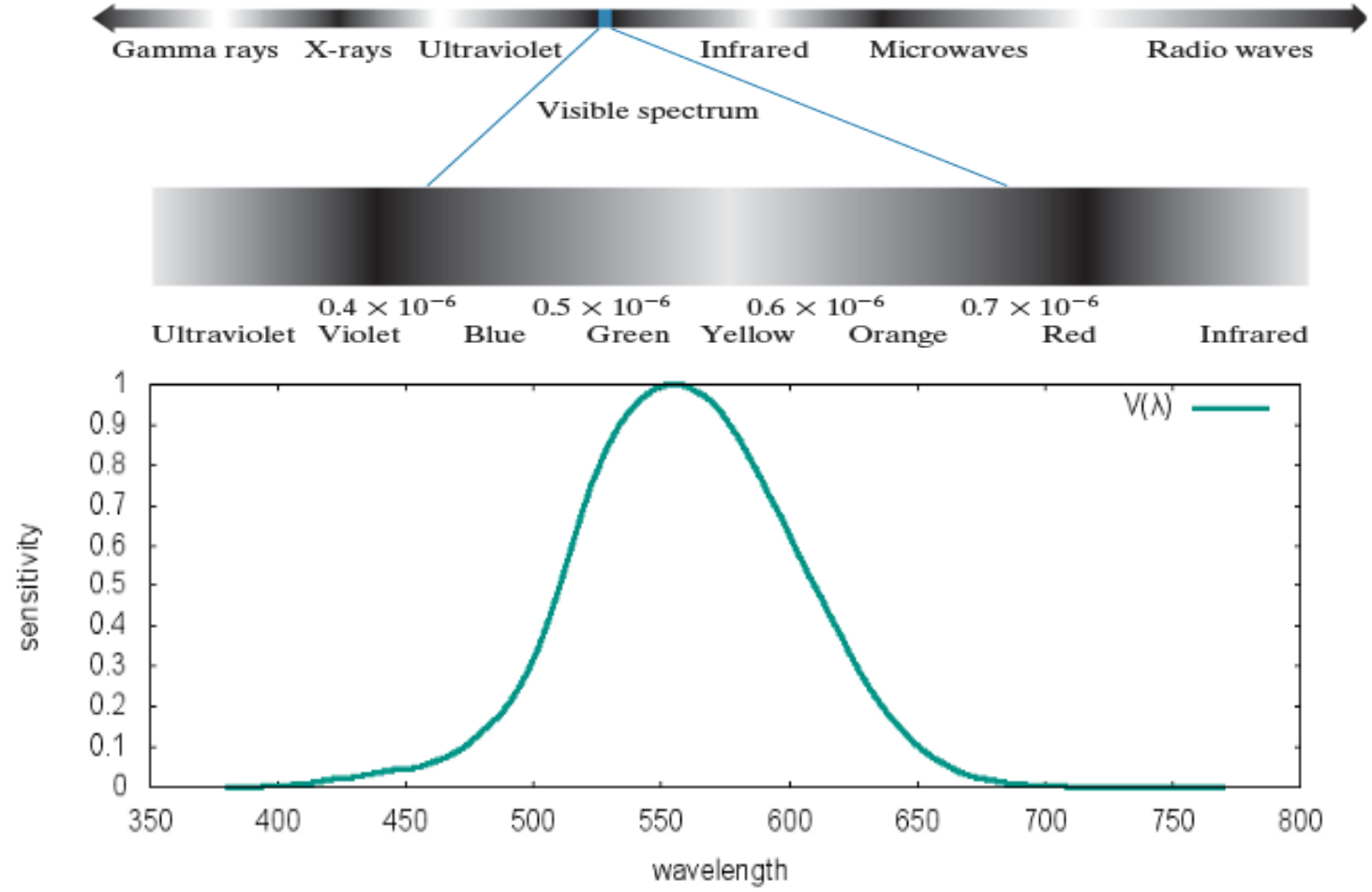


Figure is taken from https://medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a

International Commission on Illumination

Color Matching functions

- The **CIE RGB Color Matching Functions** define how the human eye perceives colors using three primary lights: **Red (R), Green (G), and Blue (B)**, each with a known single wavelength.
- Assume we have three lights, red, green, and blue ones with known single wavelengths.
- By adjusting the intensity of these primary lights, we can create **all visible colors** through additive color mixing.

Key Principles:

•Additive Color Mixing

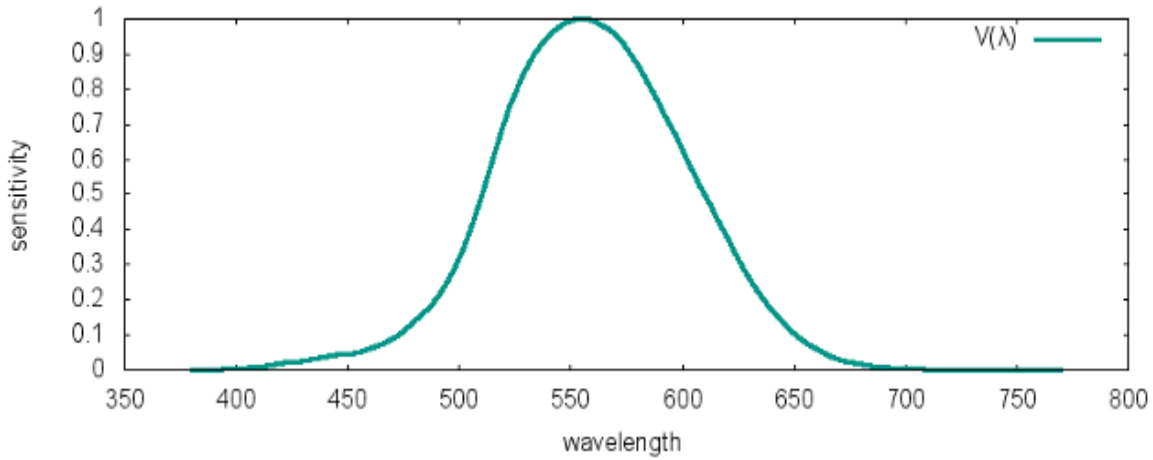
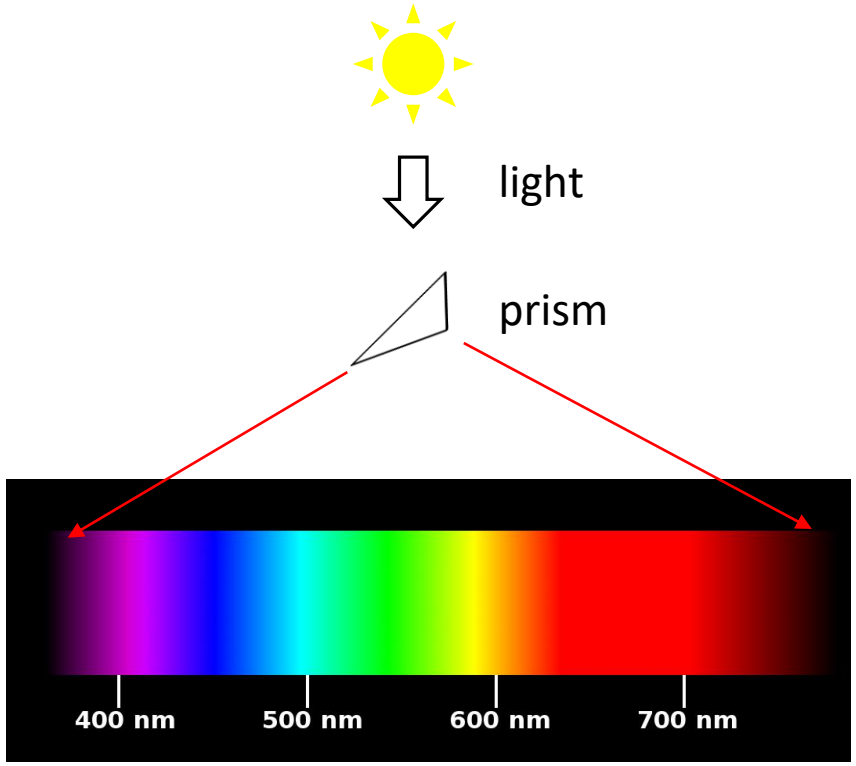


Figure is taken from https://medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a

Color Spectrum

A prism separates white light into the fundamental spectral colors.

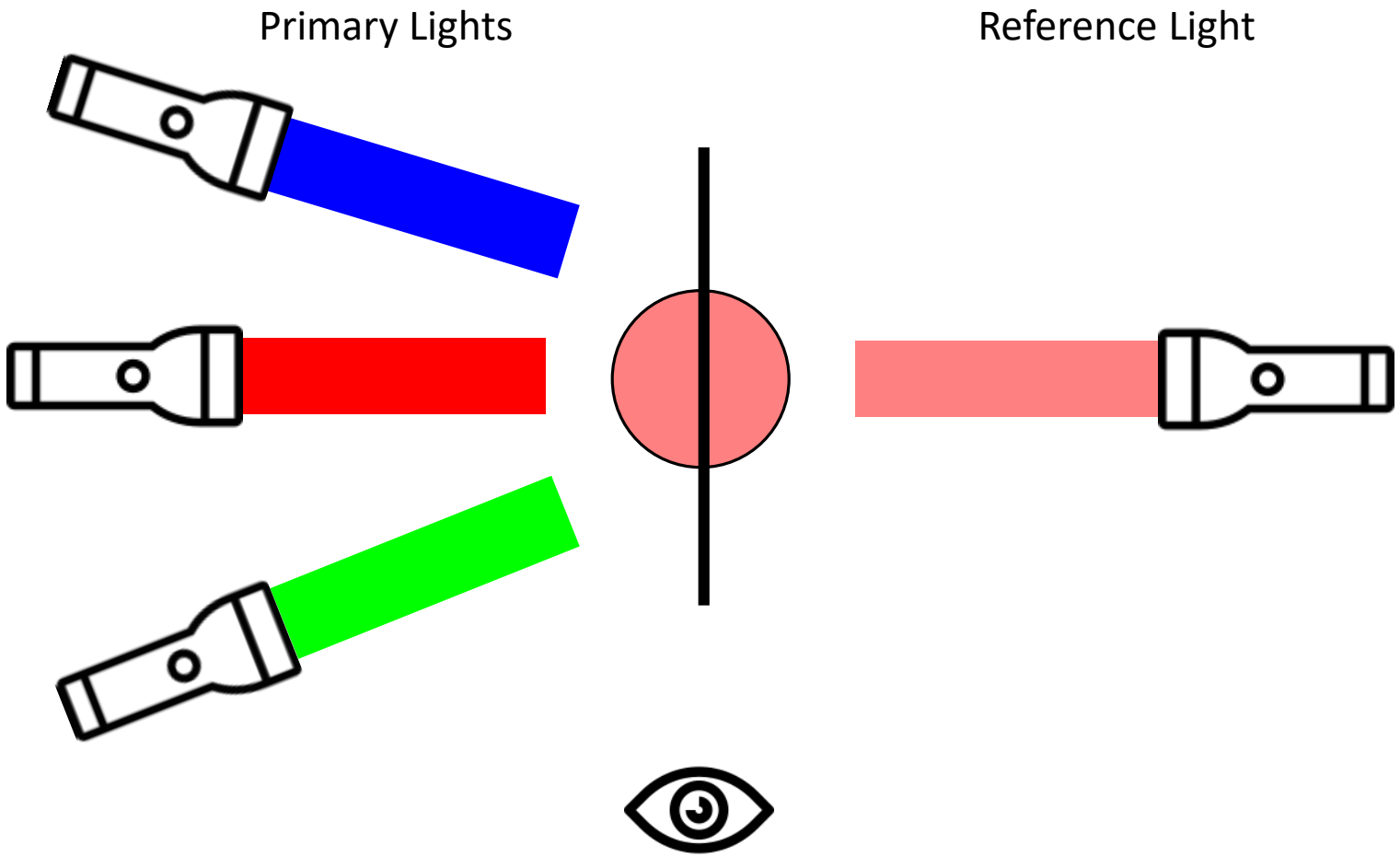


Can we recreate all the colors on the color spectrum using Red, Green, and Blue primary colors?

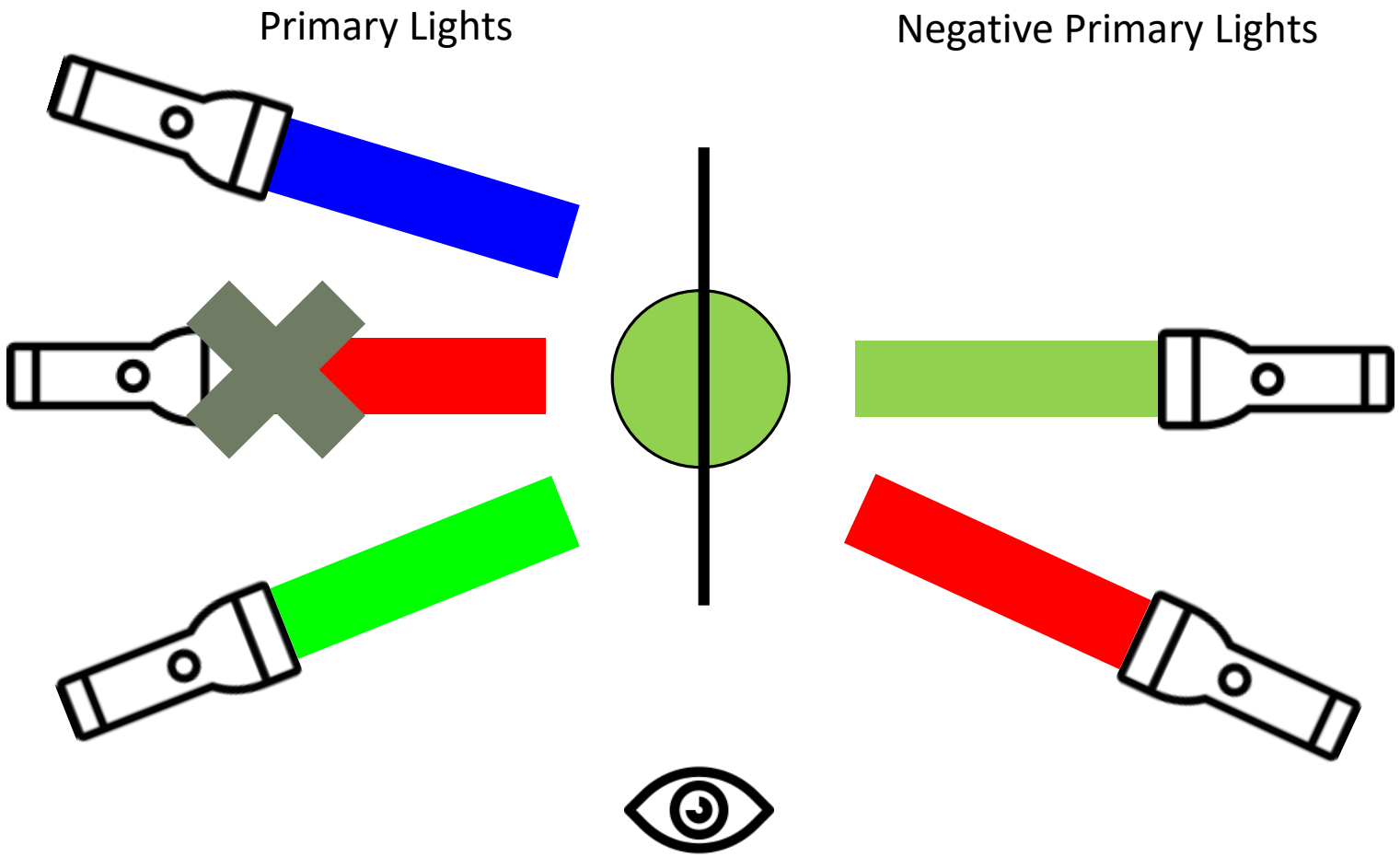
Ans : No

Image source: Christopher S. Baird.

Wright Guild Color Matching Experiments

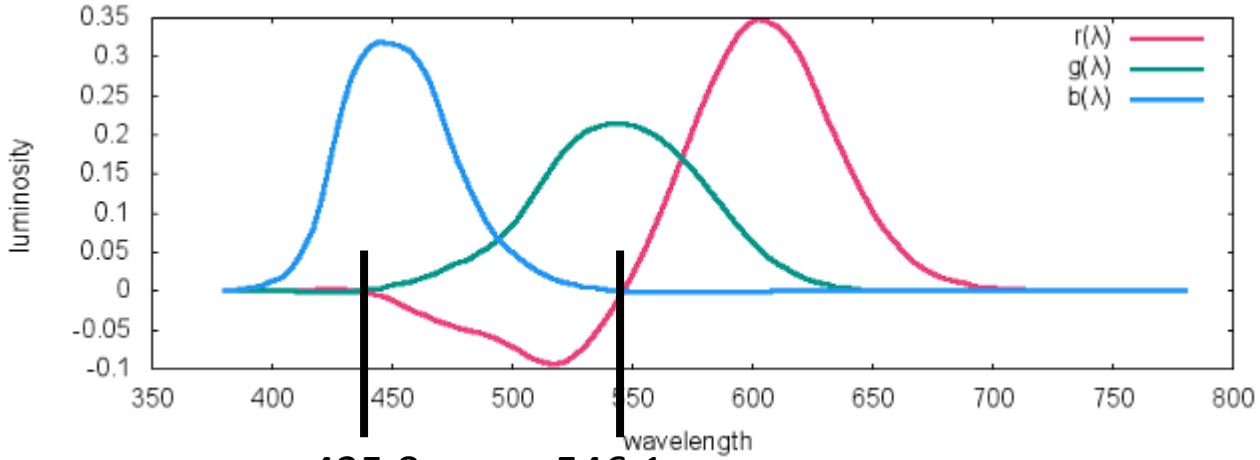


Wright Guild Color Matching Experiments



Color Matching System

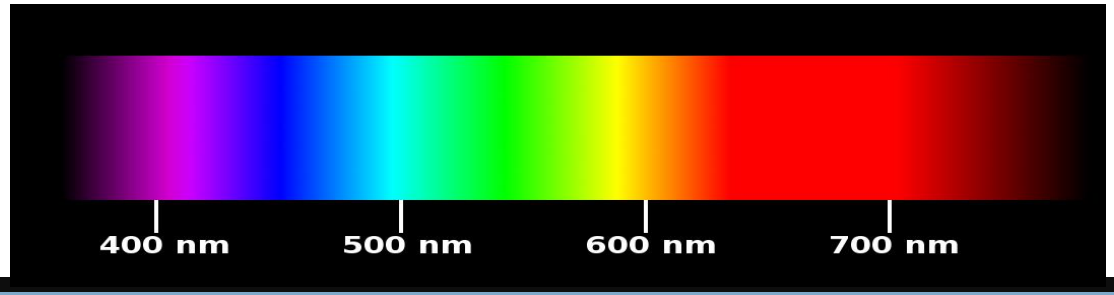
The CIE RGB color matching functions



435.8nm 546.1nm

Negative Primary Light

R(λ= 700nm), G(λ= 546.1nm) and B(λ= 435.8nm)



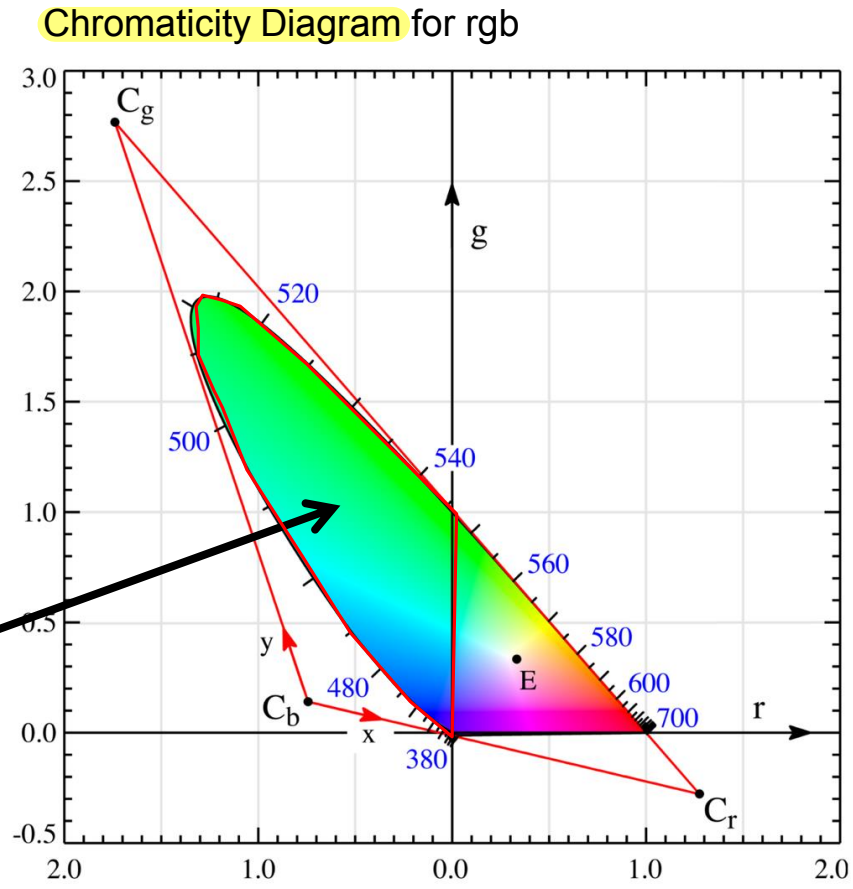
Color Space - RGB

- R, G, B, can be transformed into as trichromatic coefficients, r, g, b for plotting the chromaticity diagram

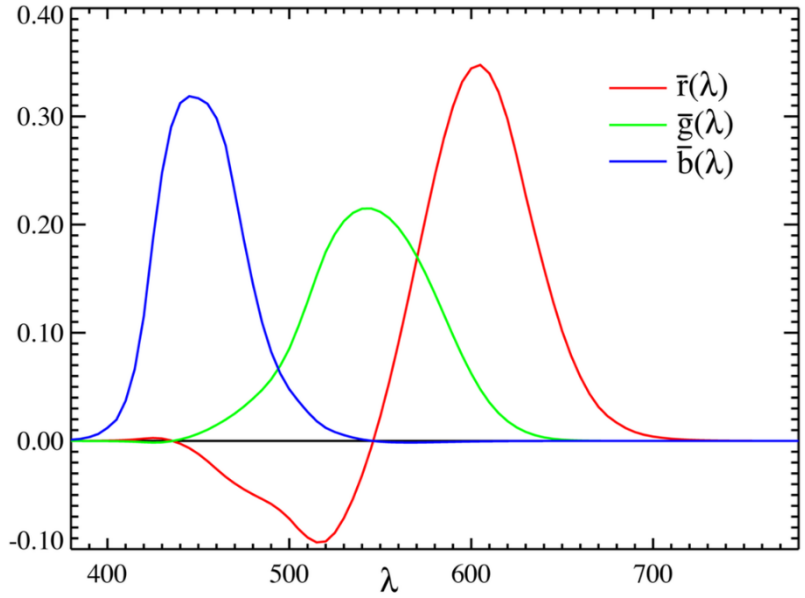
$$r = \frac{R}{R+G+B} \qquad g = \frac{G}{R+G+B} \qquad b = \frac{B}{R+G+B}$$

$$r + g + b = 1$$

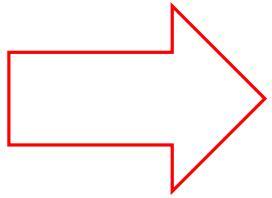
Negative tristimulus values

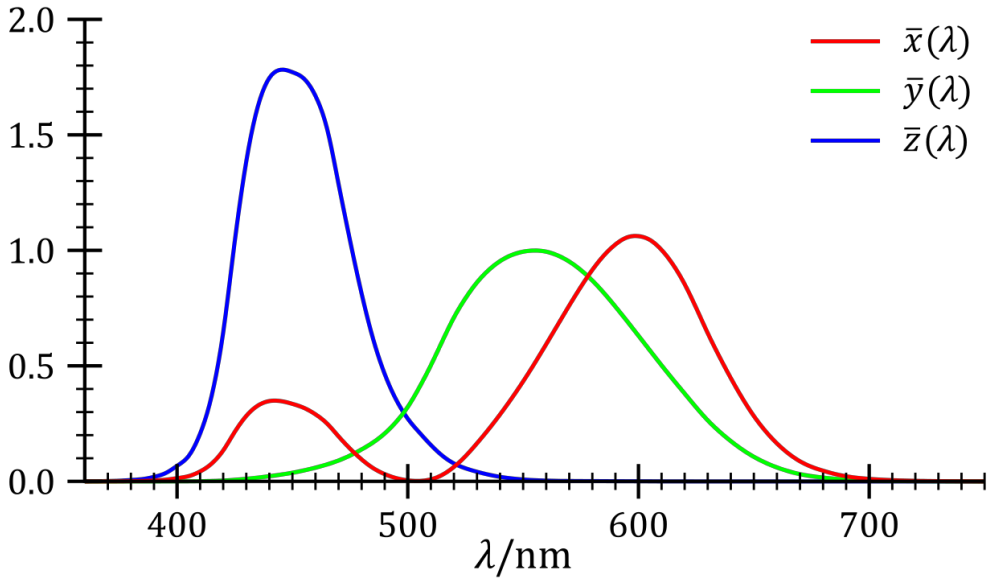


CIE RGB -> XYZ Color Matching Functions



The CIE RGB color matching functions





The CIE XYZ standard color matching functions

https://en.wikipedia.org/wiki/CIE_1931_color_space

Color Space - XYZ

- To avoid negative primary light effect, RGB is linearly transformed to XYZ color space.
- Compared to RGB, XYZ color space has mathematically convenient properties
- X, Y, Z are called “tristimulus values”
  - They are all positive
  - It encompasses all color sensations that are visible to the human eye
  - It can also be represented as trichromatic coefficients, x, y, z

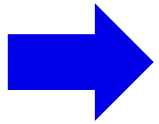
$$x = \frac{X}{X+Y+Z} \qquad y = \frac{Y}{X+Y+Z} \qquad z = \frac{Z}{X+Y+Z} \qquad x + y + z = 1$$

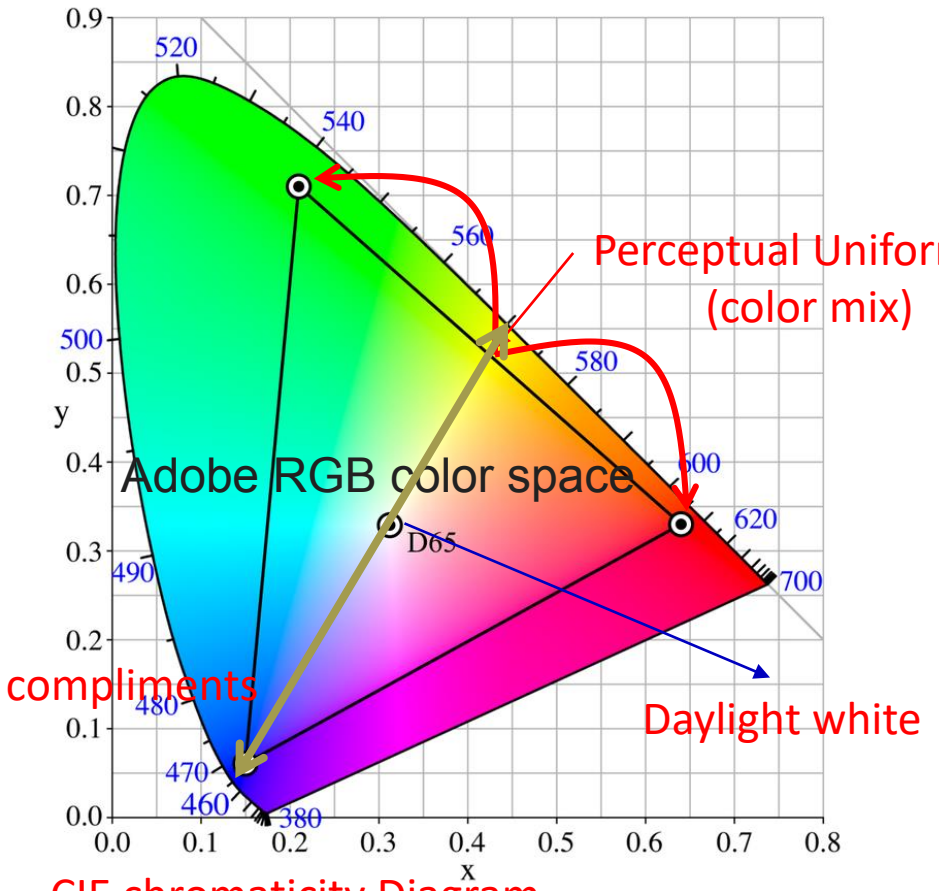
Source: www.ImageProcessingPlace.com

Color Spaces with RGB Primaries

Red: 700.0 nm
Green: 546.1 nm
Blue: 435.8 nm

- To link color primaries to human-visible color, it can be demonstrated using Adobe RGB color space by CIE 1931 color space chromaticity coordinates
- CIE 1931 Color Space & Chromaticity Diagram
  - **CIE 1931** is a **Color Matching System** that defines how a color can be **numerically specified and accurately reproduced**.
  - The **CIE color space** is visualized as a **horseshoe-shaped chromaticity diagram**, representing all colors perceptible to the human eye.
  - The **Adobe RGB color space** is a practical example of how **RGB color primaries** fit within the **CIE 1931 chromaticity diagram**.
  - This mapping helps standardize **color accuracy** in **displays, photography, and digital imaging**.





Color Spaces for Video Compression

- **RGB color space** is the most well-known and widely used for display purposes. However, it is **not efficient for video compression** due to its high redundancy and bandwidth requirements.
- **Y'CbCr color space** is commonly used for video coding because it separates **luminance (Y')** 亮度 from **chrominance (Cb, Cr)**, allowing for **efficient compression and storage**.
- Y'CbCr vs. YUV:
  - YUV is used for analog video signals.
  - Y'CbCr is designed for digital video processing and compression (e.g., JPEG, MPEG, H.264).
- Why Y'CbCr is preferred
  - Chrominance subsampling: Reduces data by encoding color information at a lower resolution while preserving detail in luminance.
  - Human visual perception: The human eye is more sensitive to brightness (Y') than color differences (Cb, Cr), making it possible to compress chroma without noticeable quality loss.

→ 色度, Cb: blue difference, Cr = red difference

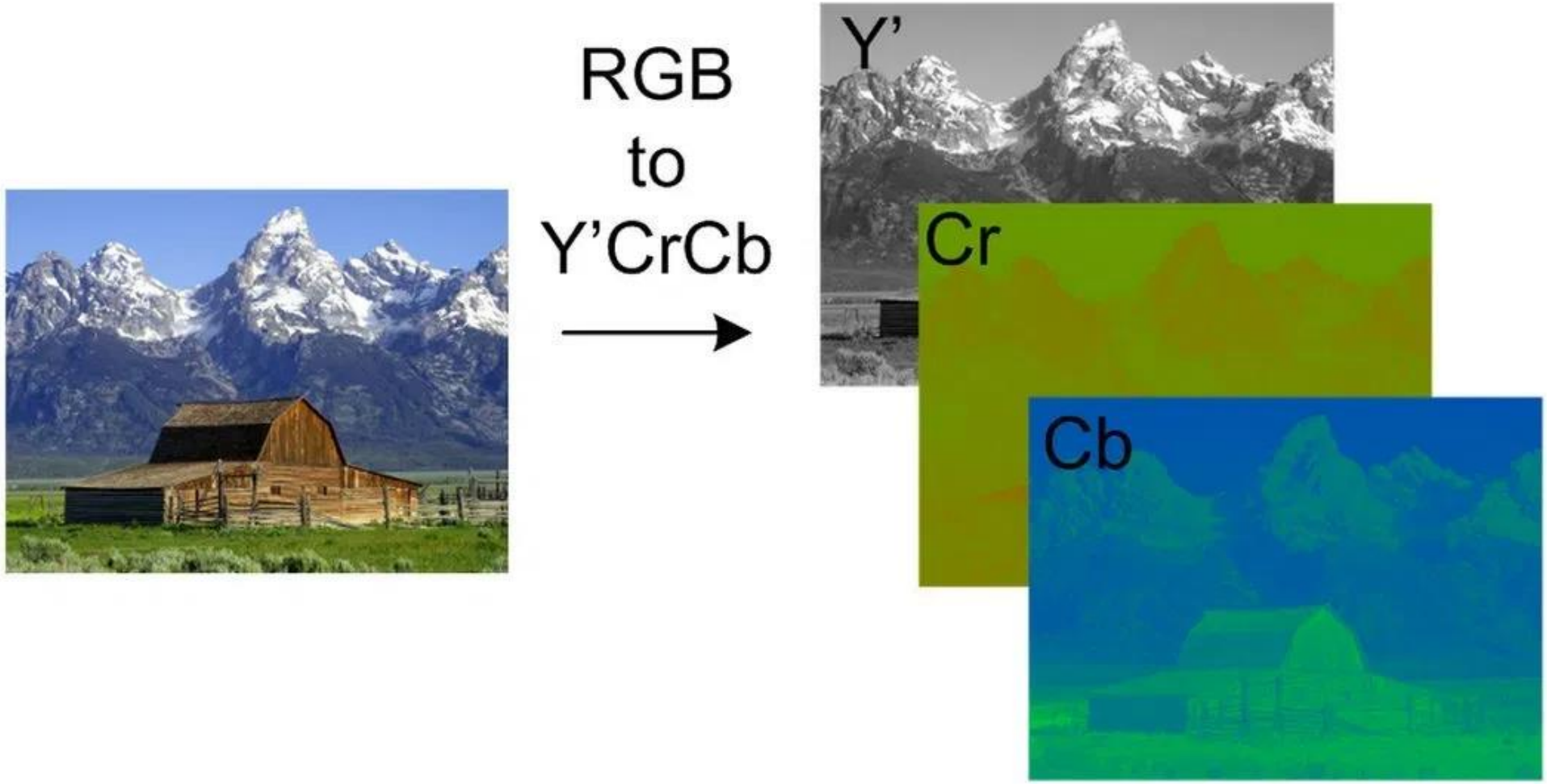


Figure is taken from https://www.hisour.com/ycbcr-color-spaces-26075/

Y’CbCr Color Spaces

- ITU-R BT.601 (CCIR 601)
  - For Digital Video
  - $16 \leq Y' \leq 235$, Cr & Cb range: 128 ± 112
  - $Y'_{601} = 0.299 \times R' + 0.587 \times G' + 0.114 \times B'$
  - $C_b = -0.168736 \times R' + 0.331264 \times G' + 0.5 \times B'$
  - $C_r = 0.5 \times R' + 0.418688 \times G' + 0.081312 \times B'$
- ITU-R BT.709
  - For HDTV Studio Video, Computer Graphics, etc.
  - $16 \leq Y' \leq 235$, Cr & Cb range: 128 ± 112
  - $Y'_{709} = 0.2125 \times R' + 0.7154 \times G' + 0.0721 \times B'$

Color Conversions

- RGB to BT-709 Y'CrCb (assuming 8-bit video)

$$\begin{bmatrix} E'_Y \\ E'_{PB} \\ E'_{PR} \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ -0.115 & -0.386 & 0.500 \\ 0.500 & -0.454 & -0.046 \end{bmatrix} \begin{bmatrix} E'_R \\ E'_G \\ E'_B \end{bmatrix}$$

$$E'_Y, E'_R, E'_G, E'_B \sim [0,1], \quad E'_{PB}, E'_{PR} \sim [-0.5,0.5]$$

$$\begin{cases} Y = 219 \cdot E'_Y + 16 \\ Cb = 224 \cdot E'_{PB} + 128 \\ Cr = 224 \cdot E'_{PR} + 128 \end{cases}$$

Ref. ISO/IEC 14496-2 Section 6.3.2

* Human vision is more sensitive to luminance (luma) than chrominance (chroma)

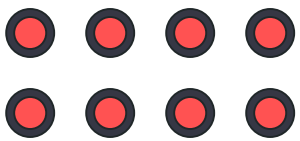
Chroma Subsampling

x:y:z

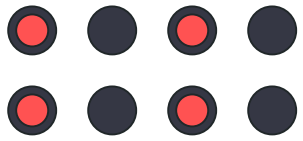
- x: the number of pixels for luma in each row
- y: the number of pixels in the top row have chroma
- z: the number of pixels in the bottom row have chroma

Subsampling is a technique used for (lossy) video compression

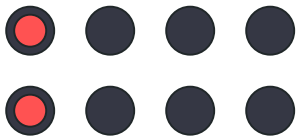
↳ describe how chroma info. is sampled relative to luma in a 2x2 pixel block



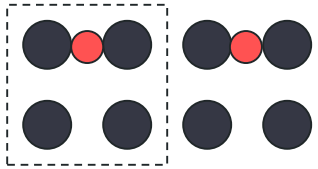
4:4:4



4:2:2



4:1:1



4:2:0

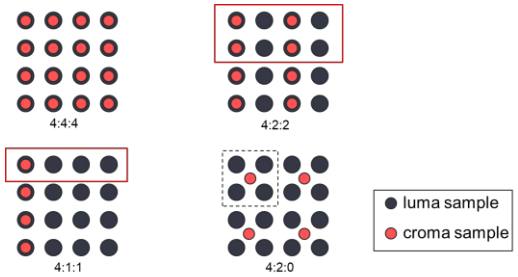
luma sample

chroma sample

Chroma Subsampling in Video Formats

- 4:4:4 (No Subsampling)
  - **Highest quality**, full chroma resolution.
    - Used in **HDCAM SR** for HDTV production.
      - Supports **10-bit 4:2:2 or 4:4:4** color depth.
- 4:2:2 (Professional-Grade Video)
  - **Reduces chroma by 50% horizontally** while preserving full vertical resolution.
  - Used in **high-end formats** like **AVC-Intra 100, Digital Betacam, Digital-S**.
- 4:1:1 (Lower Chroma Resolution)
  - **Color resolution is quartered horizontally** per row.
  - Found in **DVCPRO (NTSC/PAL), NTSC DV, and DVCAM**.
- 4:2:0 (Efficient Compression for Streaming & Broadcasting)
  - Halves chroma resolution both horizontally and vertically
  - Standard for **MPEG, H.26x codecs, VC-1**, and other digital video formats.

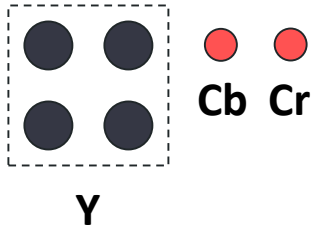
12 bpp



Ref: https://en.wikipedia.org/wiki/Chroma_subsampling#4:4:4

Chroma Subsampling Efficiency for 420

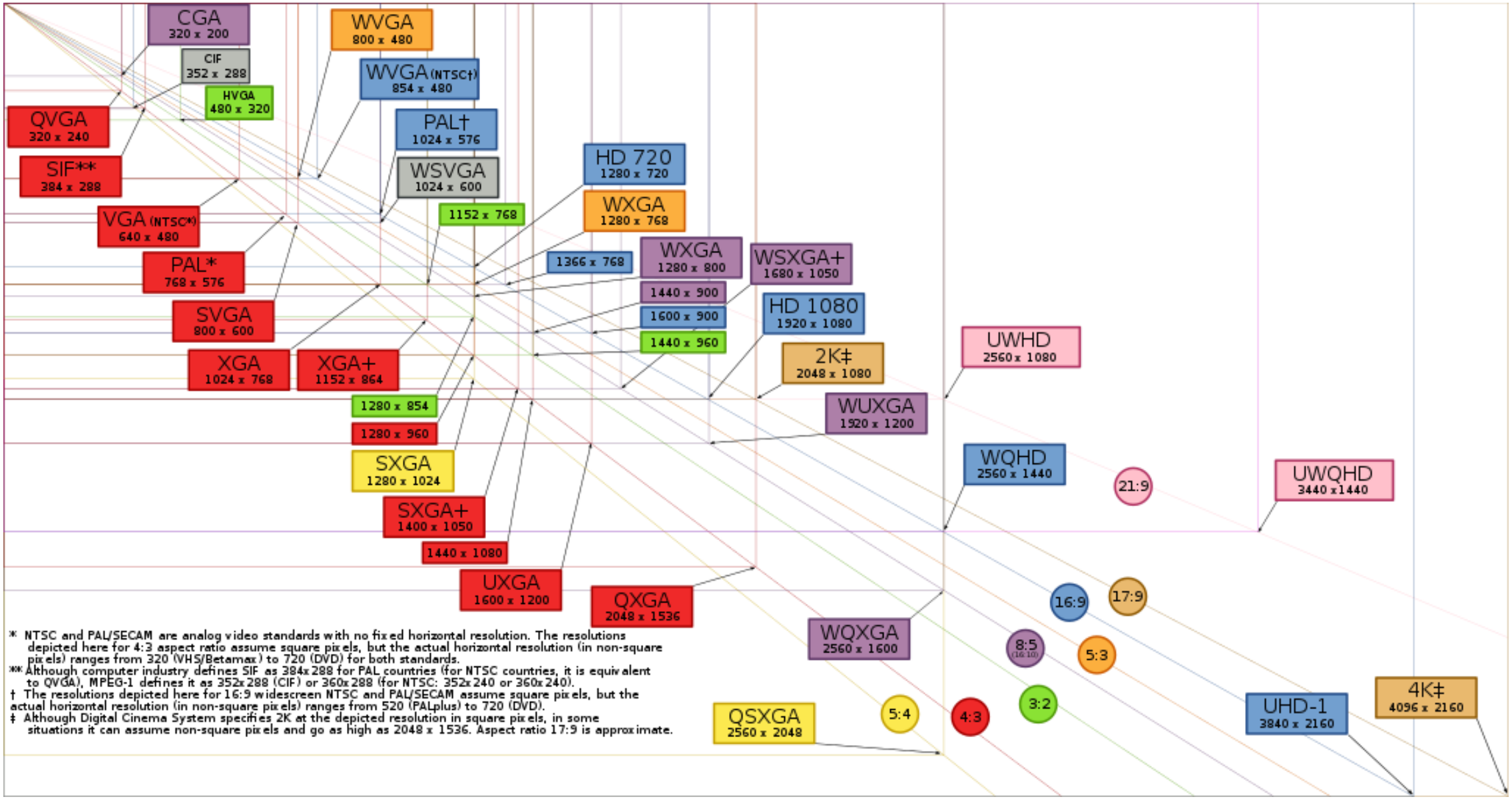
- The **Cb (blue-difference) and Cr (red-difference) components** can be stored at a lower resolution than **Y (luminance)** because the human eye is **more sensitive to brightness details than color variations**.
- 4:2:0 and Bit Depth
  - In **4:2:0 chroma subsampling**, chroma data is reduced both horizontally and vertically, leading to efficient compression.
  - It is sometimes referred to as **"12 bits per pixel (bpp)"** because each pixel, on average, receives **8 bits for Y** and **2 bits each for Cb and Cr**, spread across multiple pixels.



$$\frac{(8 \times 6)\ \text{bits}}{4\ \text{pixels}} = 12\ bpp$$

Old Video Formats

- CCIR-601 (NTSC): 720x480, interlaced, 4:2:2
- CCIR-601 (PAL): 720x576, interlaced, 4:2:2
- CIF : 352x288, progressive, 4:2:0
- QCIF: 176x144, progressive, 4:2:0
- SIF : 352x240, progressive, 4:2:0



REF: https://en.wikipedia.org/wiki/Display_resolution#/media/File:Vector_Video_Standards8.svg

Some Basic Definitions

- Intensity
  - Intensity refers to the rate at which radiant energy is transferred per unit area. In image science, we measure power over a specific range of the electromagnetic spectrum, typically focusing on power radiated from or incident upon a surface. Intensity is considered a linear-light measure, commonly expressed in units such as watts per square meter (W/m^2).
  - In a CRT monitor, the voltages applied control the intensities of the color components. However, this relationship is nonlinear—the voltage levels are not directly proportional to intensity.

cathode-ray tube

- Luminance
  - The CIE defines **luminance (Y)** as radiant power **weighted by a spectral sensitivity function** that aligns with human vision. While luminance is proportional to physical power—similar to intensity—its spectral composition is specifically related to human **brightness perception** rather than raw energy transfer.
  - Luminance can be computed as a weighted sum of the linear-light red, green, and blue (RGB) primary components. These weights reflect the varying sensitivities of the human eye to different colors, with green contributing the most, followed by red, and then blue.

Brightness is a perceptual quantity related to human visual sensitivity

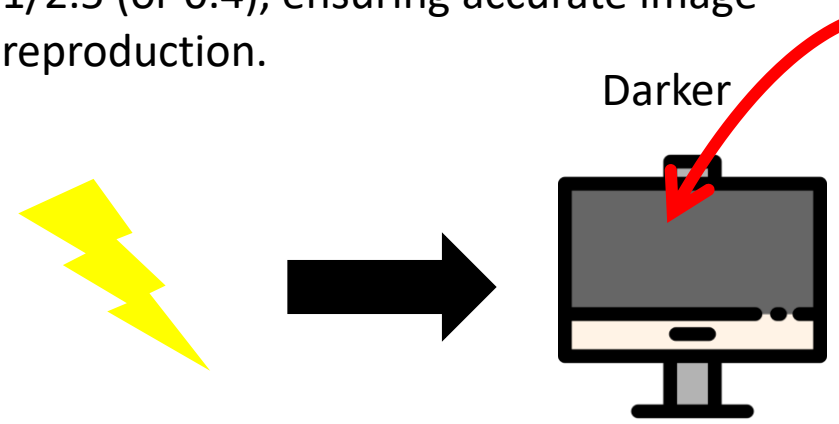
Some Basic Definitions

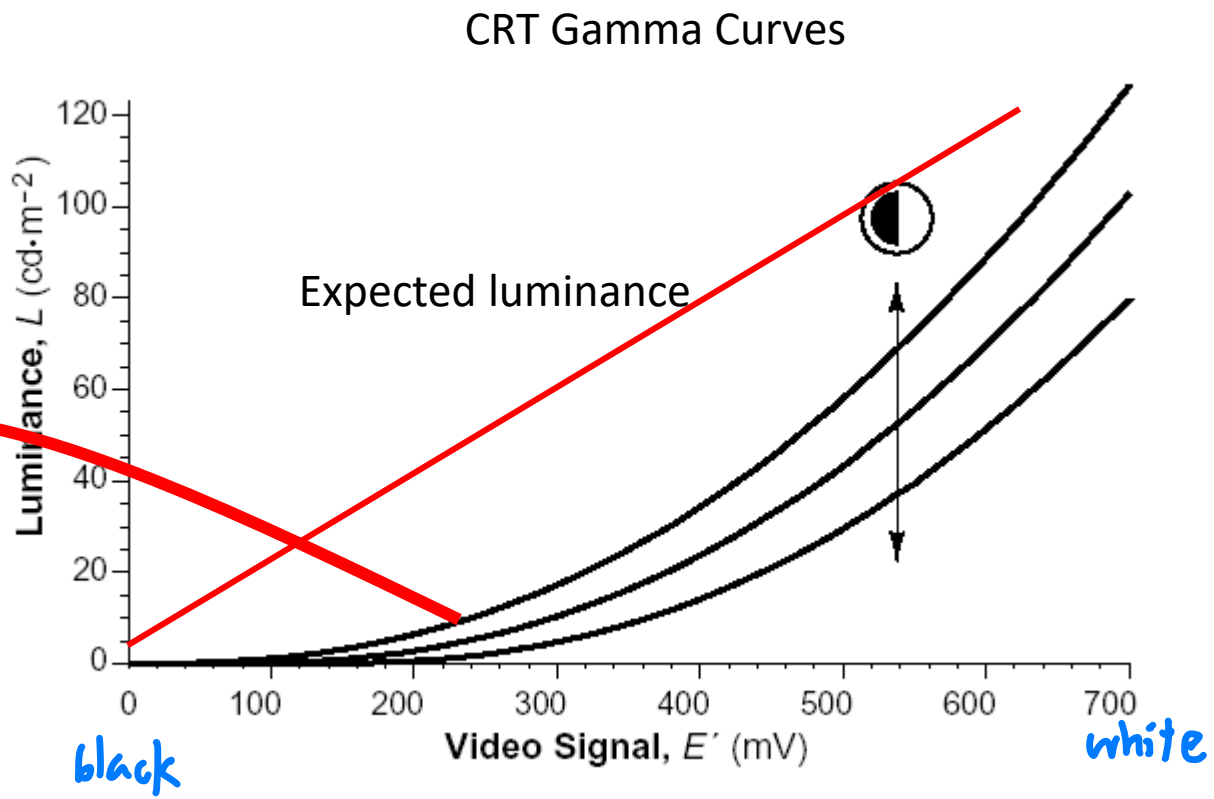
- Luma vs. Luminance
  - Luma (Y'): In video coding, it is computed as a weighted sum of nonlinear R'G'B' primaries (gamma-corrected).
  - Luminance (Y): The weighted sum of linear RGB components, representing actual light intensity.

- Gamma Correction
  - The intensity of light generated by a physical device is not usually a linear function of the applied signal.
  - **Gamma correction** is essential in image processing and display technologies to ensure accurate brightness and color reproduction, aligning with human visual perception.
  - Display devices like CRT monitors exhibit a nonlinear relationship between the input voltage and the resulting light intensity. Specifically, the light intensity (I) produced by a CRT is proportional to the applied voltage (V) raised to the power of approximately 2.5: $I \propto V^{2.5}$. This exponent is referred to as the device's gamma (γ)

Purpose of Gamma Correction

- To achieve accurate image reproduction, it's necessary to compensate for this nonlinearity.
- Gamma correction applies an inverse transformation to the input signal before it reaches the display
- It counteracts display nonlinearity by adjusting the signal with a gamma value of approximately 1/2.5 (or 0.4), ensuring accurate image reproduction.





This graph shows a video signal from 0 to 700 mV, where black is code 0 and white is code 255 in an 8-bit digital-to-analog converter on a framebuffer card.

encoding: $E = L^{1/\gamma}$ (usually $\gamma = 2.2$)

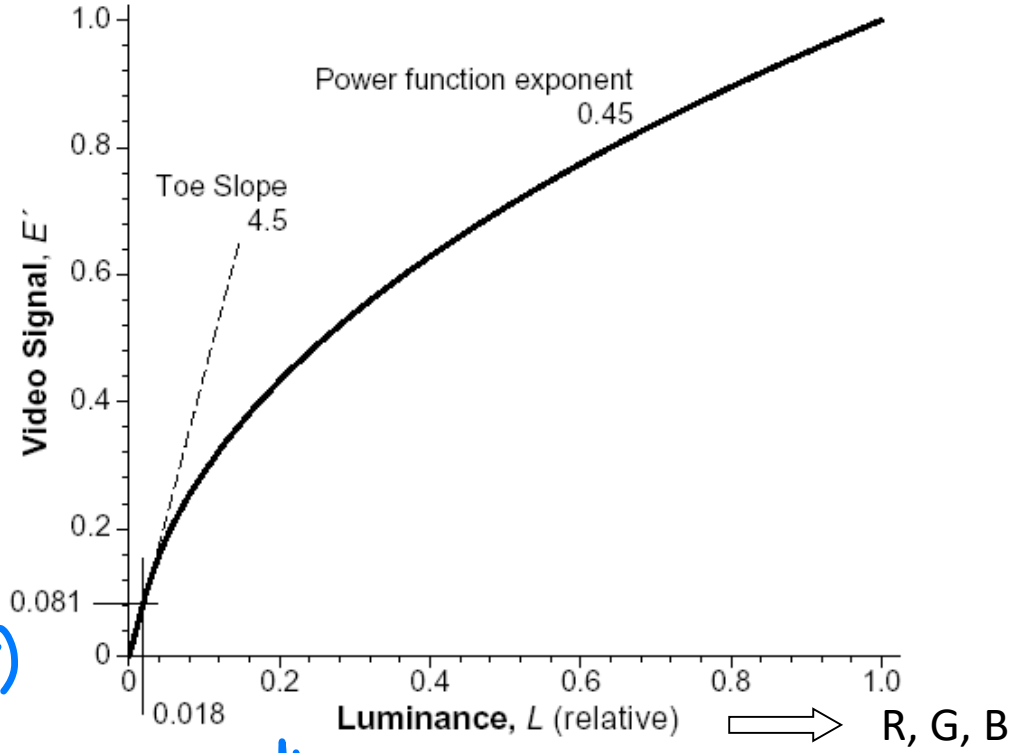
decoding: $L = E^{\gamma}$

Gamma Correction

In a video system, gamma correction transforms linear-light intensity into a nonlinear video signal, typically applied at the camera..

$0.4 \approx \frac{1}{2.2}$

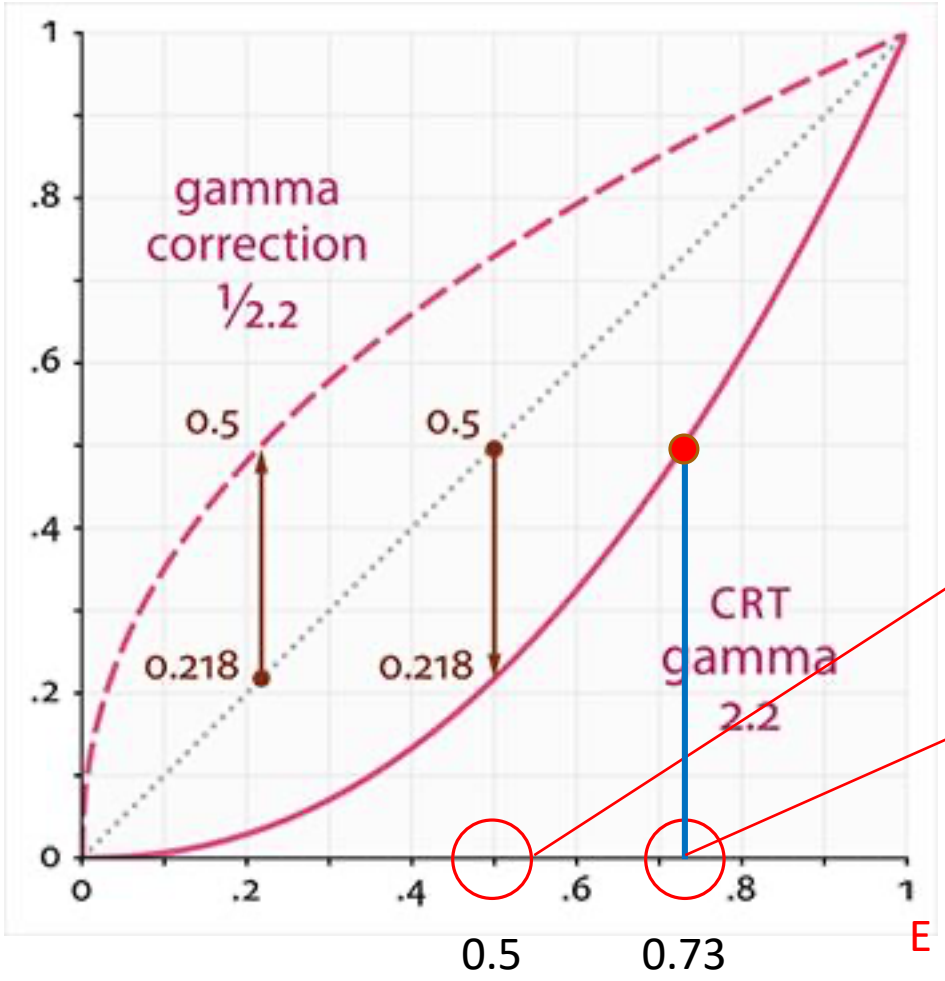
$$E'_{709} = \begin{cases} 4.5L, & L \le 0.018 \\ 1.099L^{0.45} - 0.099, & L > 0.018 \end{cases}$$



L: luminance (linear light intensity)

E': encoded video signal (gamma-corrected)

Luminance



I = 0.5

Expected Luminance

I = 0.218

Presented Luminance

I = 0.5

Presented Luminance

E= 0.5

Gamma correction

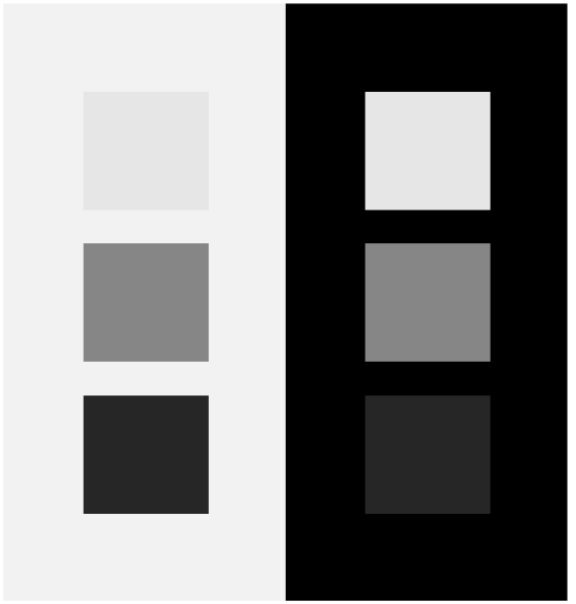
$E= 0.5^{\frac{1}{2.2}} = 0.73$

Surround Effect

- The typical gamma value is around **2.2**, but it varies based on display settings and viewing conditions:
  - **sRGB Standard**: Uses **gamma ~2.2**, common for computer monitors and web content.
  - **Broadcast TV (Rec. 709)**: Also around **2.2**, optimized for home viewing.
  - **Film Industry (Rec. 1886)**: Uses **gamma 2.4**, suited for dimly lit environments.
  - **Mac Systems (Pre-2009)**: Previously used **gamma 1.8** for better print matching.
  - **HDR Displays (PQ/ST 2084)**: Uses a more complex **perceptual quantizer** instead of a fixed gamma.
- For televisions viewed in dim environments, **simultaneous contrast** is considered to enhance the visual experience.
  - In dimly lit rooms, the eye perceives shadows more deeply (darker) and highlights more intensely (brighter).
  - To compensate, television standards like Rec. 1886 use a slightly higher gamma (~2.4) for a more natural appearance.
  - This ensures a better balance between dark and bright areas, preventing the image from looking washed out.

Surround Effect. The three gray squares surrounded by white are identical to the three gray squares surrounded by black, but the contrast of the black-surround series appears lower than that of the white-surround series.

– *LeRoy DeMarsh*

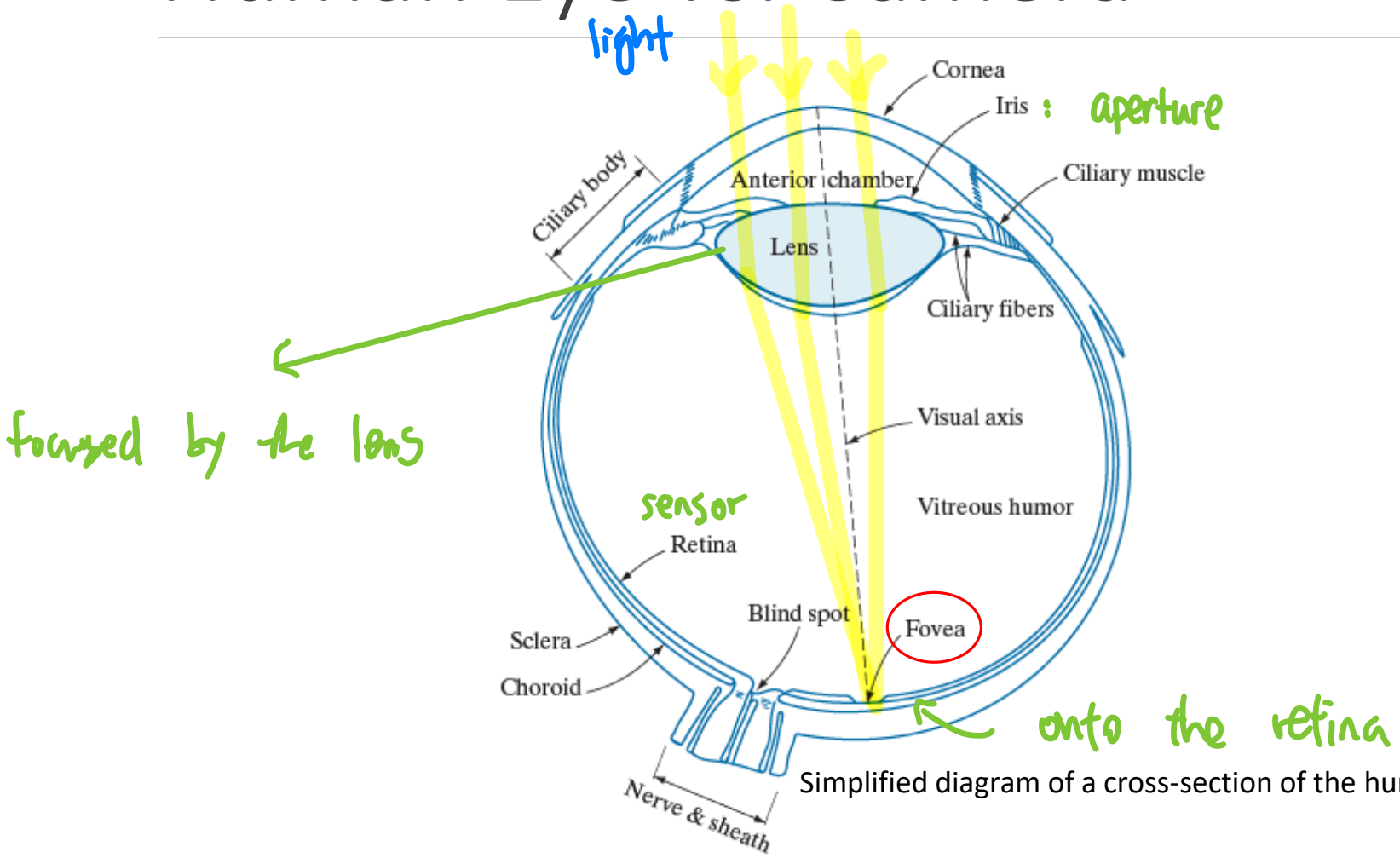


Ref: https://poynton.ca/PDFs/GammaFAQ.pdf

Human Visual Perception

- Human eyes can tolerate some types of information loss, allowing for efficient data compression
- Based on human visual perception characteristics
  - we can keep important and essential information intact but remove unnecessary information
- Compression Techniques
  - Lossless vs. lossy
    - Lossless: Used in legal/medical imaging where accuracy is critical (low compression ratio)
    - Losy: Multimedia data, such as audio, image, and video
      - some errors or loss are tolerable (and may not be noticed) – high compression ratio
      - taking advantage of human visual perception characteristics
- Constant bit rate (CBR) vs. variable bit rate coding(VBR)
  - Contents varying, VBR is more efficient in providing higher-quality contents but requires more complex processing
  - **CBR**: Maintains a fixed data rate, ensuring consistent performance

Human Eye vs. Camera



Simplified diagram of a cross-section of the human eye

Term	Corresp. To Camera
Iris/pupil	aperture 光圈
Retina	Sensor
Lens 晶状体	Focus

Source: www.ImageProcessingPlace.com

Human Visual System (HVS)

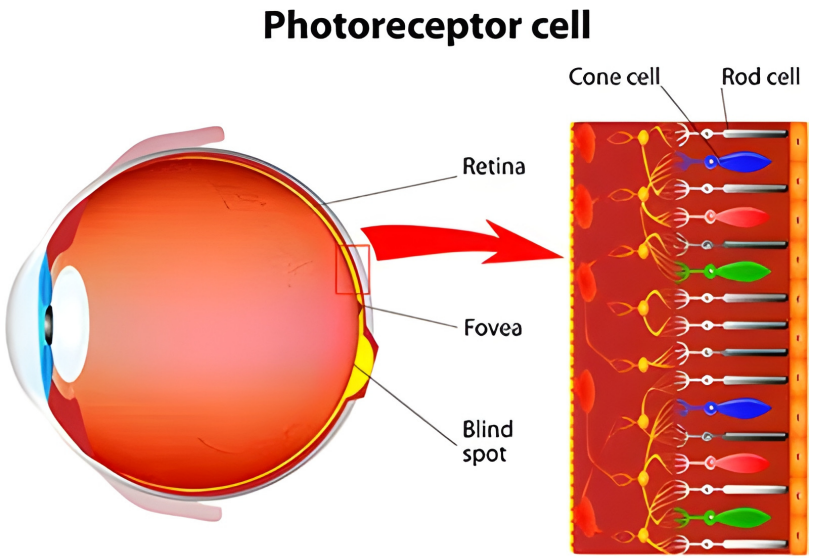
- ***Eye:*** The light is focused by the ***lens*** onto the ***retina.*** The iris, which works like the aperture in a camera, controls the amount of light that enters the eye.
- ***Retina****:* There are full of cone and rod cells on the retina. In the central region (the fovea), cone cells are concentrated, meaning the human eye perceives more color information of view in a small central region.
- ***Optic nerve****:* It carries and transmits electrical signals from the retina to the brain.
- ***Brain****:* It has neurons to process and interpret the electrical signals for humans to understand and recognize the view.

Retina: The Light-Sensitive Layer Full of Sensors

- light → photoreceptor cells on the retina → neural signals
- photopic vision : 明視覺
- scotopic vision : 暗視覺

- The **retina** contains specialized photoreceptor cells that convert light into neural signals, enabling vision.

- **Cone Cells**
  - **Function:** Responsible for **color vision** and **sharp detail**
    - Works Best: In **bright light** (photopic vision).
  - **Location:** Concentrated in the **fovea** (central retina).
  - **Types:**
    - **S-Cones** (Short) – Blue light
    - **M-Cones** (Medium) – Green light
    - **L-Cones** (Long) – Red light
- **Rod Cells**
  - **Function:** Enable **vision in low light** and **peripheral vision**.
  - **Location:** Spread throughout the **peripheral retina**, absent in the fovea.
  - **Sensitivity:** Highly sensitive to light, but **do not detect color**.
  - **Works Best:** In **dim light** (scotopic vision).



(視網膜的)中央凹

Fovea: The Center of Sharp Vision

黃斑部

- The **fovea** is a small pit in the **macula** of the retina, responsible for **sharp central vision** essential for tasks like reading, driving, and recognizing faces.
  - 🎯 **High Cone Density:** Packed with cone photoreceptors for **color vision** and **fine detail**, with no rod cells.
  - 🌈 **Color & Detail Perception:** Enables the **sharpest**, most vibrant vision in the center of our gaze.
  - 👁️ **Visual Acuity:** Light focuses on the fovea when looking directly at an object, providing **maximum clarity**.

- The fovea is the key to seeing vivid colors and fine details in our direct line of sight.

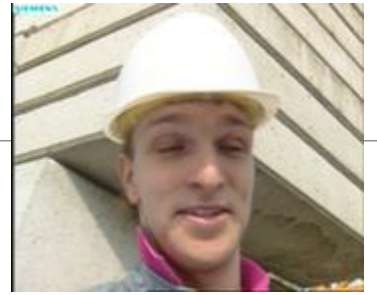
Human Visual System (HVS)

- From Features of the HVS

Feature	What we can do for compression
The HVS is more sensitive to luminance details than color	Reduce color information without noticeable quality loss.
The HVS is more sensitive to high contrast (large differences in luminance) than low contrast	Preserve large luminance changes (e.g., edges) to maintain image clarity.
The HVS is more sensitive to low spatial frequencies (gradual luminance changes) than high spatial frequencies (rapid changes over small areas)	Compress high spatial frequency data more while preserving edge details.

Y > Cb, Cr

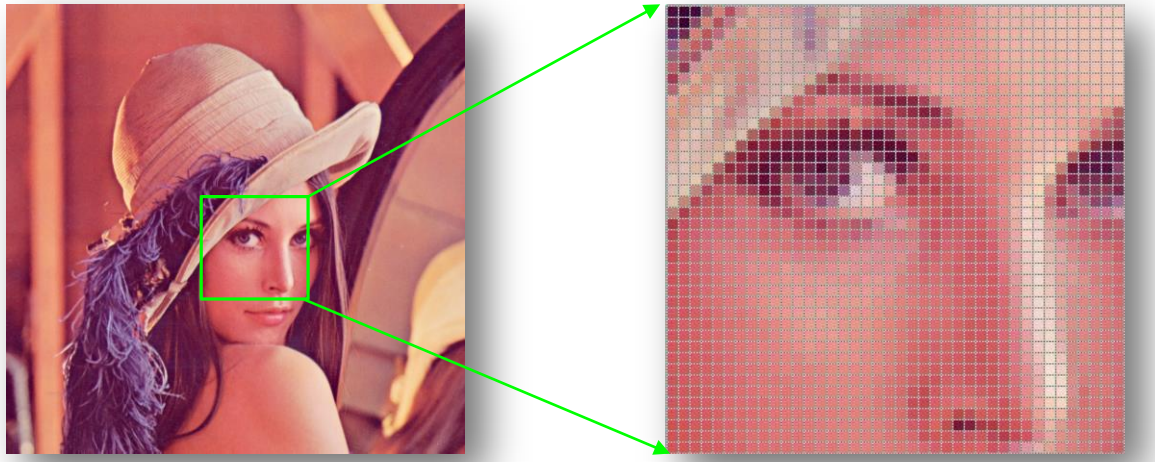
Human Visual System (HVS)



- From Features of the HVS

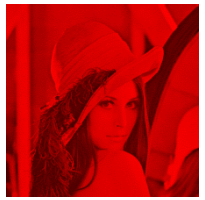
Feature	What we can do for compression
The HVS is more sensitive to image features that persist for a long duration	Avoid persistent artifacts, as they degrade perceived quality (e.g., key frames, intra-frame refresh)
The HVS perceives smooth motion when images are played at least 20 Hz	Maintain a frame rate of at least 20 Hz for smooth motion perception
HVS responses vary among individuals	Evaluate visual quality using multiple observers.

Overview of Image Compression



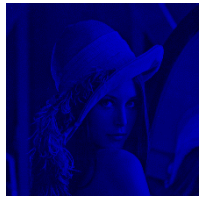
Unit: pixel

Color Image



Red plane

Green plane



Blue plane

Overview of Image Compression

If we store it in a JPEG format

Compression

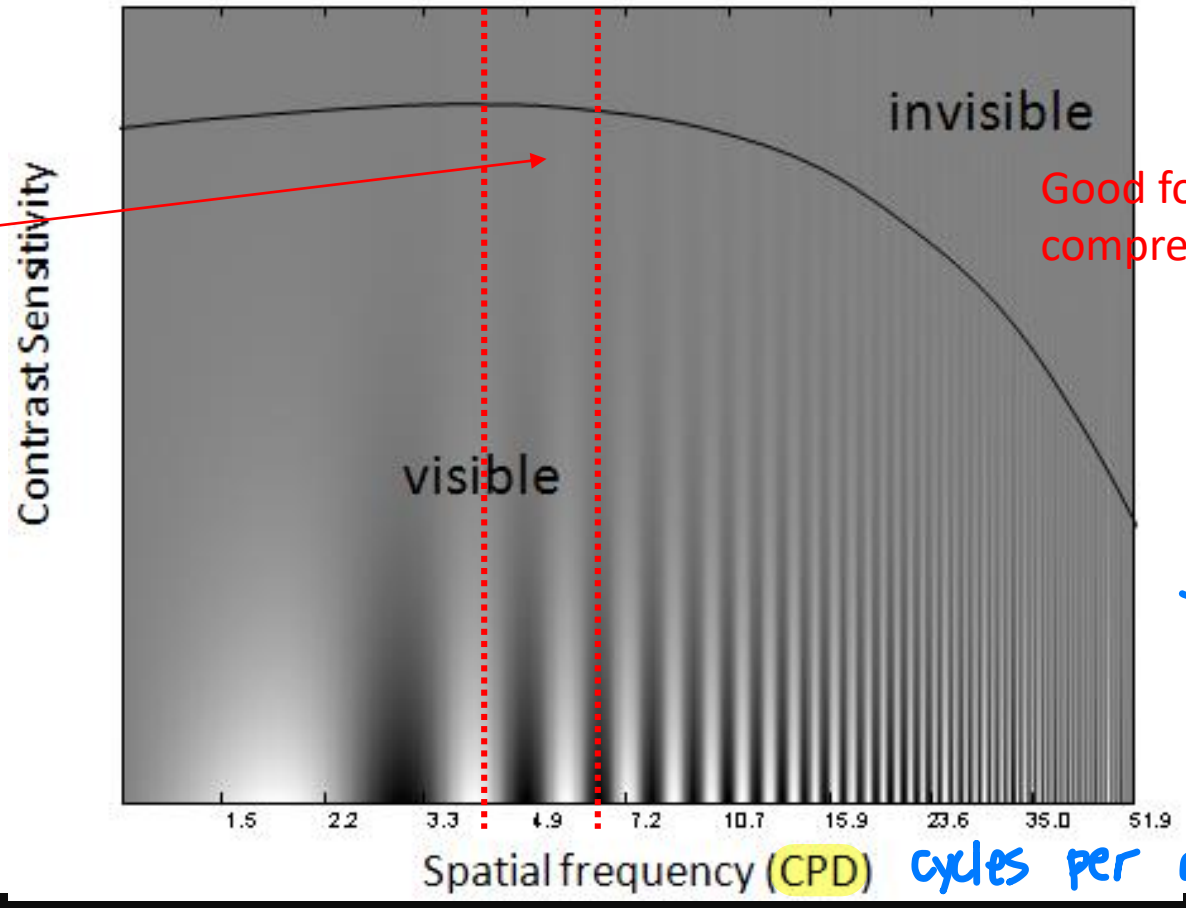
768x512 24-bit color image = 1.125 MByte

260 KB

Contrast Sensitivity vs Spatial Frequency

Contrast Sensitivity Function

- Describes how contrast sensitivity changes with spatial frequency.
- **Peak Sensitivity:** Highest at intermediate frequencies (4–6 cpd).
- **Decreasing Sensitivity:** Lower at both high and low spatial frequencies.
- **Visual Perception:** The human eye is less responsive to very fine (high-frequency) or coarse (low-frequency) patterns.



Good for compression

Contrast sensitivity function (CSF)

cycles per degree

Vanam, Rahul & Kerofsky, Louis & Reznik, Yuriy. (2014). Perceptual Pre-Processing Filter for Video on Demand Content Delivery., ICIP 2014. 10.1109/ICIP.2014.7025513.

1. What is Contrast Sensitivity?

Contrast sensitivity is how well your eyes can detect **differences in brightness** between objects and their background — especially for **faint patterns**.

- It measures **how little contrast** (light vs dark) is needed for you to **see a pattern**.
- Higher contrast sensitivity = you can see **fainter** patterns.
- It varies with **spatial frequency** — some patterns are easier to detect than others depending on their size.

---

2. What is cpd (Cycles Per Degree)?

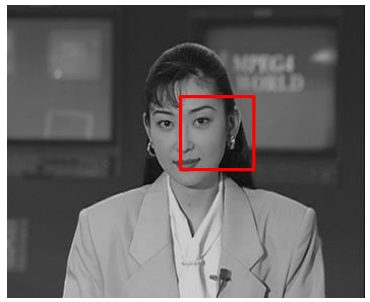
CPD = Cycles Per Degree of visual angle.

- It describes **spatial frequency**: how many **repeating patterns (cycles)** fit in one degree of your vision.
- One **cycle** = one light + one dark stripe.
- Higher CPD = **finer details** (more cycles squeezed into a small angle).
- Lower CPD = **coarser patterns** (big stripes).

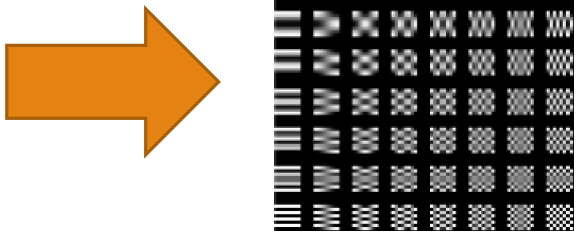
🔍 Example:

- A pattern with **2 cpd** = 2 light-dark stripe pairs per visual degree.
- A pattern with **20 cpd** = very fine stripes, harder to see.

Transformation from Spatial to Frequency Domains



Image



Spatial -> Frequency

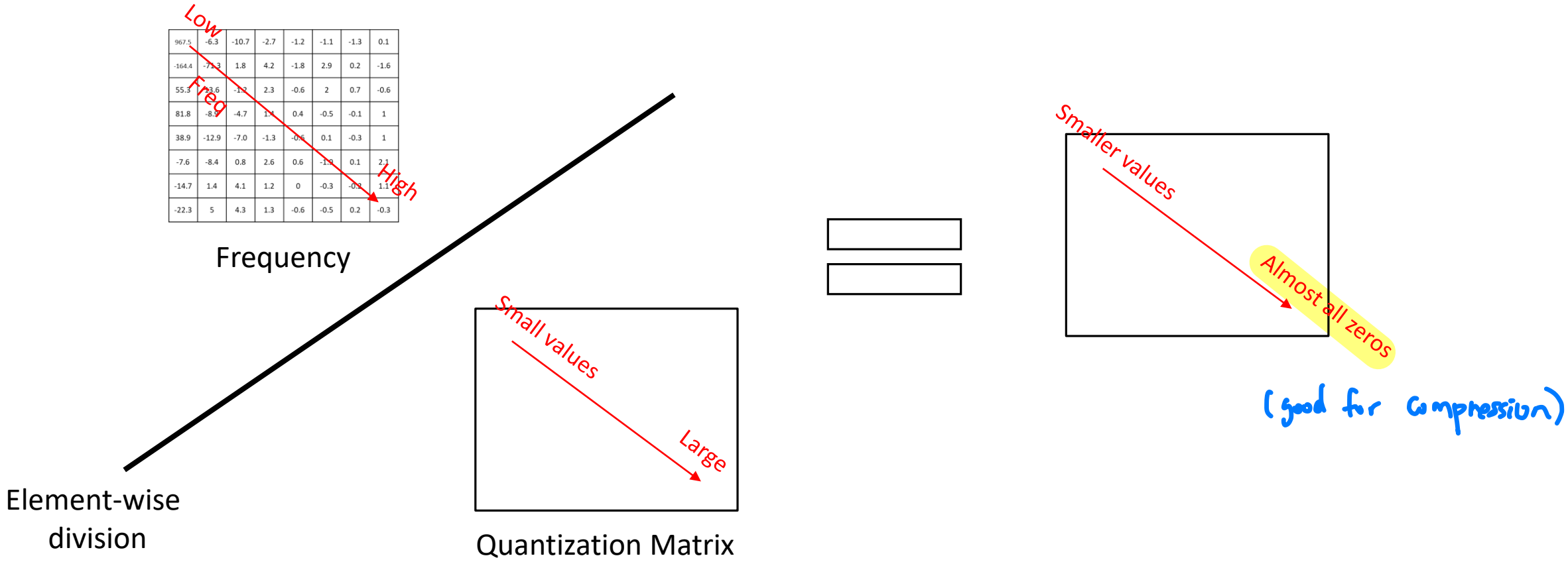
frequency coefficient

967.5	-6.3	-10.7	-2.7	-1.2	-1.1	-1.3	0.1
-164.4	-71.3	1.8	4.2	-1.8	2.9	0.2	-1.6
55.3	13.6	-1.2	2.3	-0.6	2	0.7	-0.6
81.8	-8.9	-4.7	1.4	0.4	-0.5	-0.1	1
38.9	-12.9	-7.0	-1.3	-0.6	0.1	-0.3	1
-7.6	-8.4	0.8	2.6	0.6	-1.9	0.1	2.1
-14.7	1.4	4.1	1.2	0	-0.3	-0.2	1.1
-22.3	5	4.3	1.3	-0.6	-0.5	0.2	-0.3

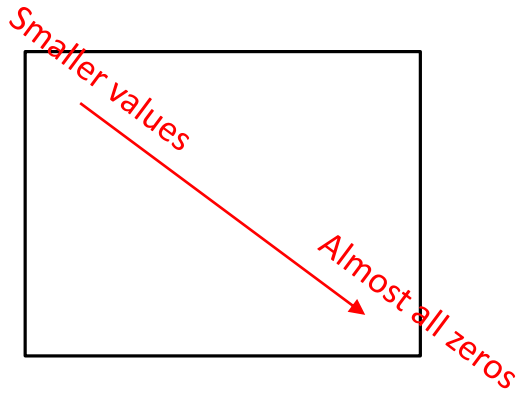
Frequency

e.g. Discrete Cosine Transform (DCT)

Lossy Compression: Quantization



Entropy Coding



Code	Freq
xxx	1/10
…	1/24
…	…

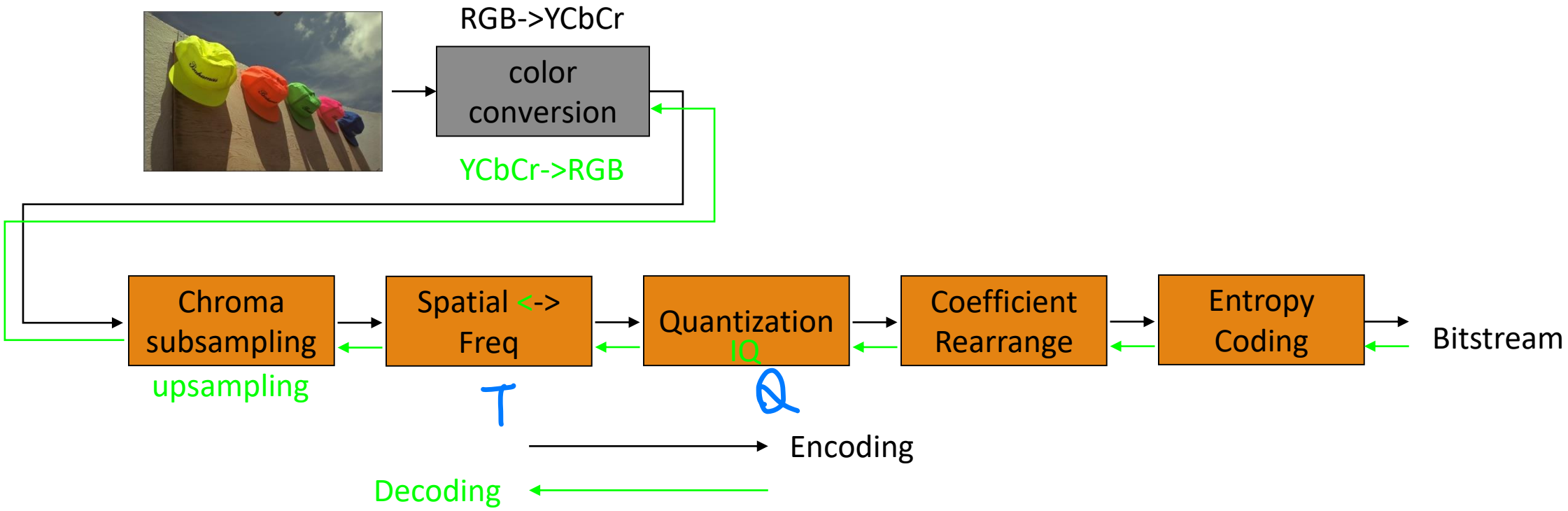
Bitstream

Re-arrange all the coefficients
(Run-length Coding)

18, 13, 0, 2, 1, 0, 0, …, 0

Huffman Coding

Image Encoding/Decoding Flowchart



Compression Artifacts

If we store it in a JPEG format




Image Size

260 KB — 41 KB — 18 KB — 12 KB

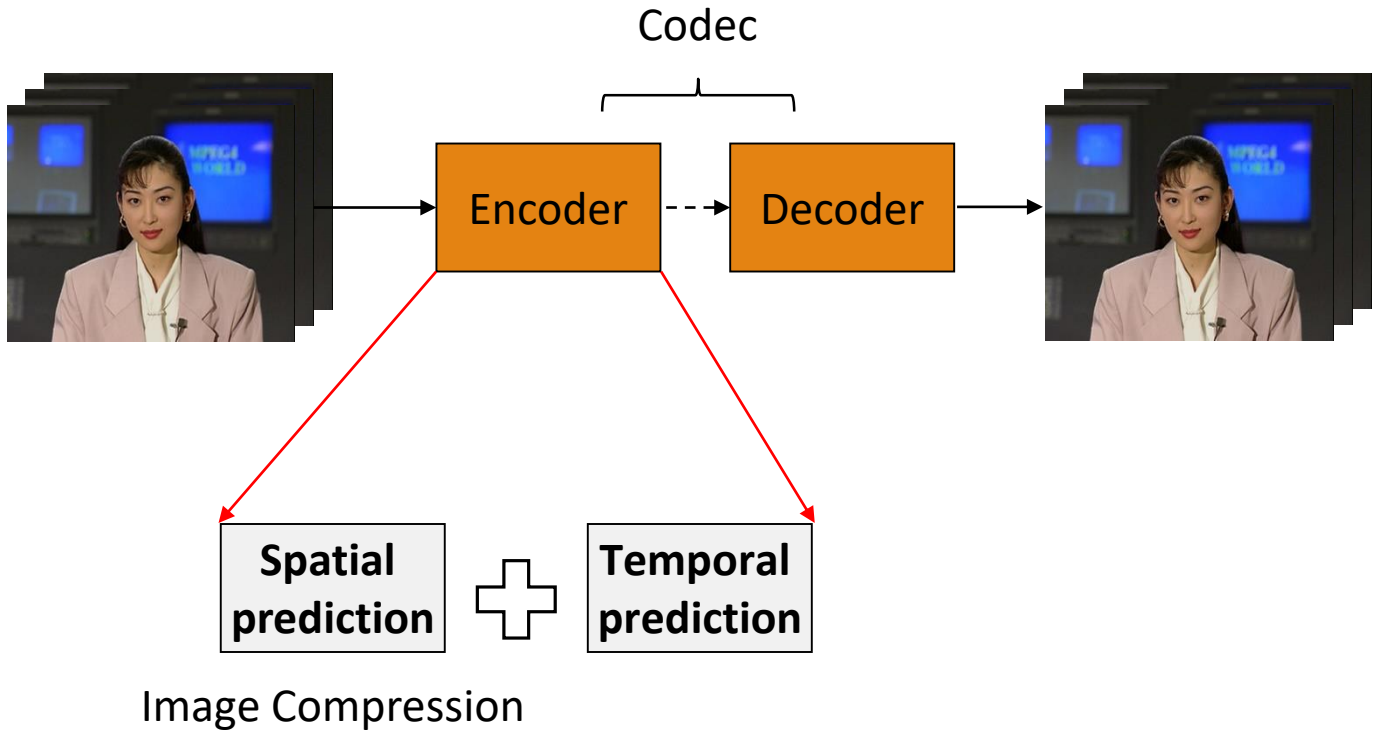
Large — Small

Quantizatoin

Weak — Strong

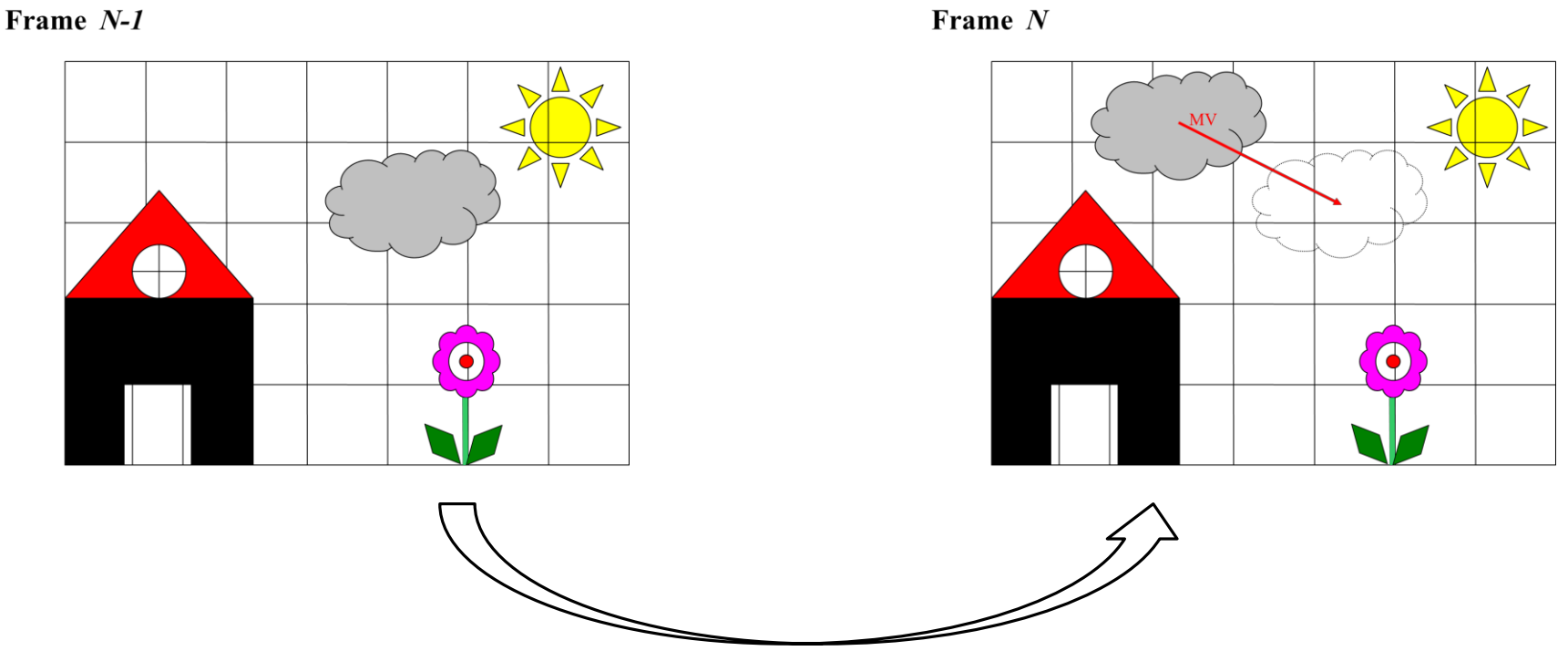
codec = Compression + decompression

Overview of Video Codec

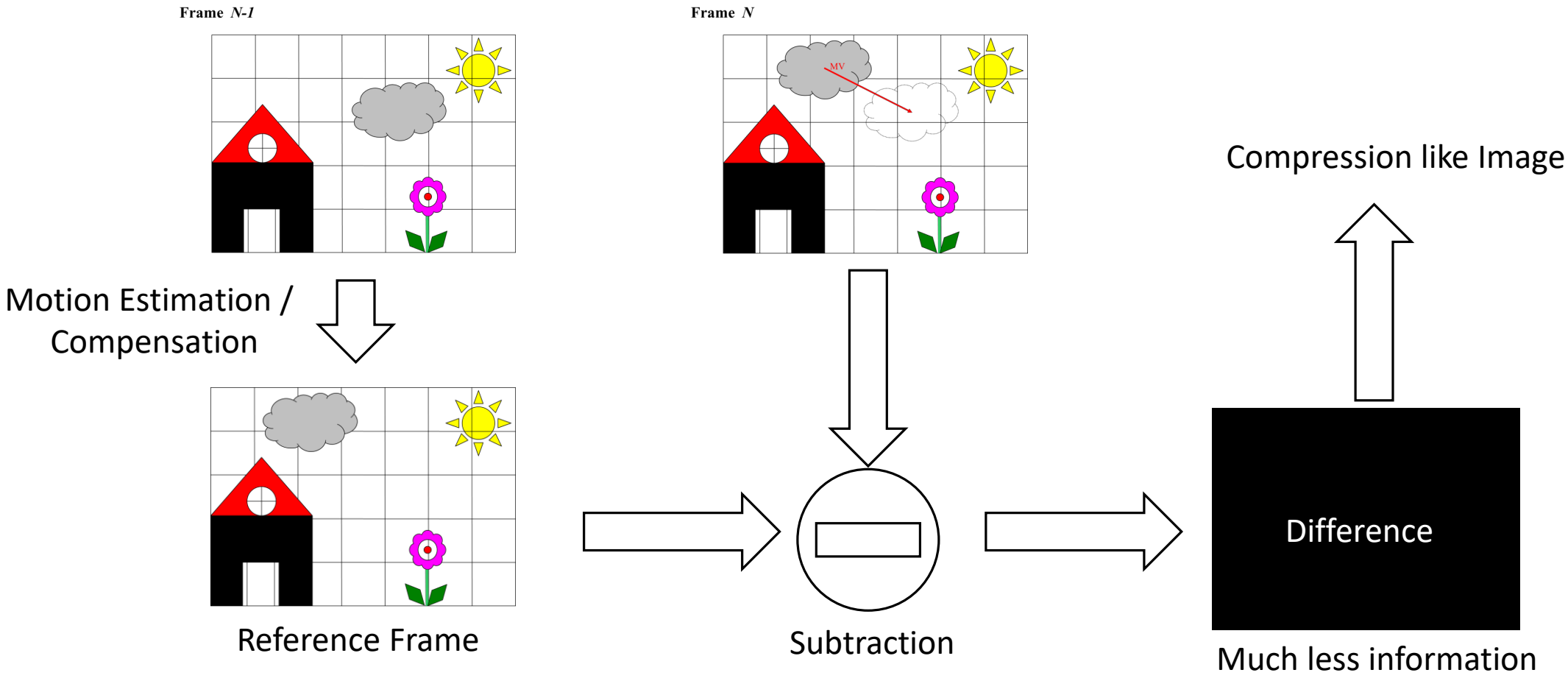


Temporal Prediction

Temporal Prediction: Focusing on changing parts across frames



Motion Estimation / Compensation



📖 1. Estimate (verb)

Oxford English Dictionary:

> *"Roughly calculate or judge the value, number, quantity, or extent of something."*

Merriam-Webster:

> *"To judge tentatively or approximately the value, worth, or significance of."*

🔧 In video compression:

- **"Estimate"** means trying to **guess how blocks in the current frame have moved** compared to the reference frame.
- The encoder **doesn't know for sure**, so it estimates motion **using algorithms** to find the best matching block from the previous frame.

🧠 Example:

> "Estimate where the cloud in Frame N came from by comparing it to Frame $N-1$."

📖 2. Compensate (verb)

Oxford English Dictionary:

> *"Offset the effect of something; make up for something undesirable."*

Merriam-Webster:

> *"To make an appropriate and usually counterbalancing payment or adjustment for..."*

🔧 In video compression:

- **"Compensate"** means **adjusting** the reference frame using the estimated motion, to account for the changes.
- You're trying to "offset" the difference caused by motion — by **rebuilding or predicting** the current frame using the past frame plus the motion vector.

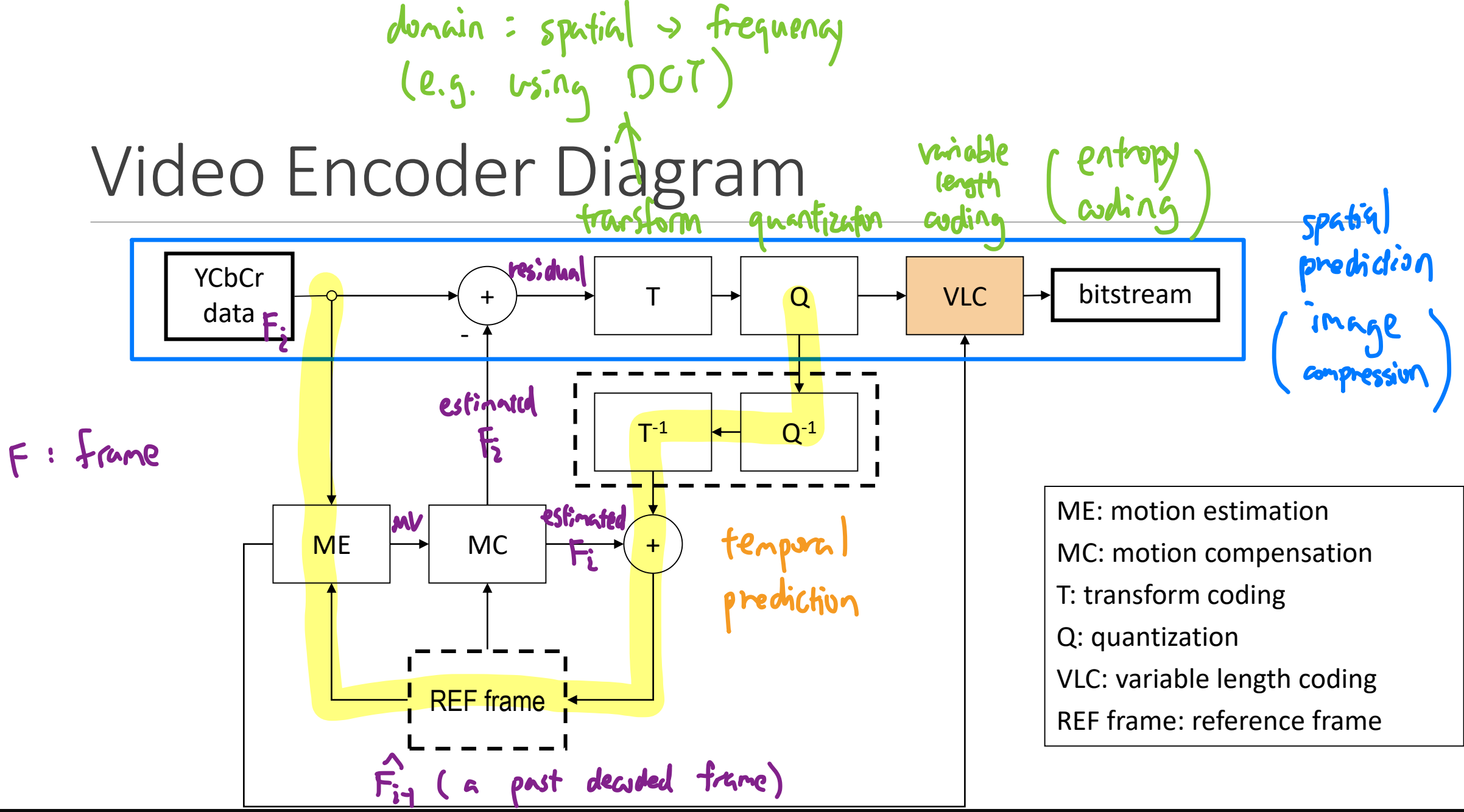
🧠 Example:

> "Compensate for the cloud's movement by shifting the cloud in the reference frame using the motion vector."

🧩 How They Work Together:

- **Estimate** motion → Find out **how** things have moved.
- **Compensate** motion → Use that info to **adjust** the old frame and predict the new one.

Video Encoder Diagram



So why do we transform, quantize it and then redo all this work? Why do we not just buffer it in the first place?

Excellent thinking, Kaiden — you're catching on to a subtle but very important point in video encoding design. Let's unpack it clearly:

❓ Why not just buffer the original frame directly after encoding?

Because the **decoder** (on the other end) doesn't receive the original frame — it only gets the **compressed bitstream**, which is made from the **quantized and transformed residual** + motion info. So...

> 📌 The encoder must **simulate the decoder's process** so both sides stay in sync.

🔁 That's why we "redo all the work":

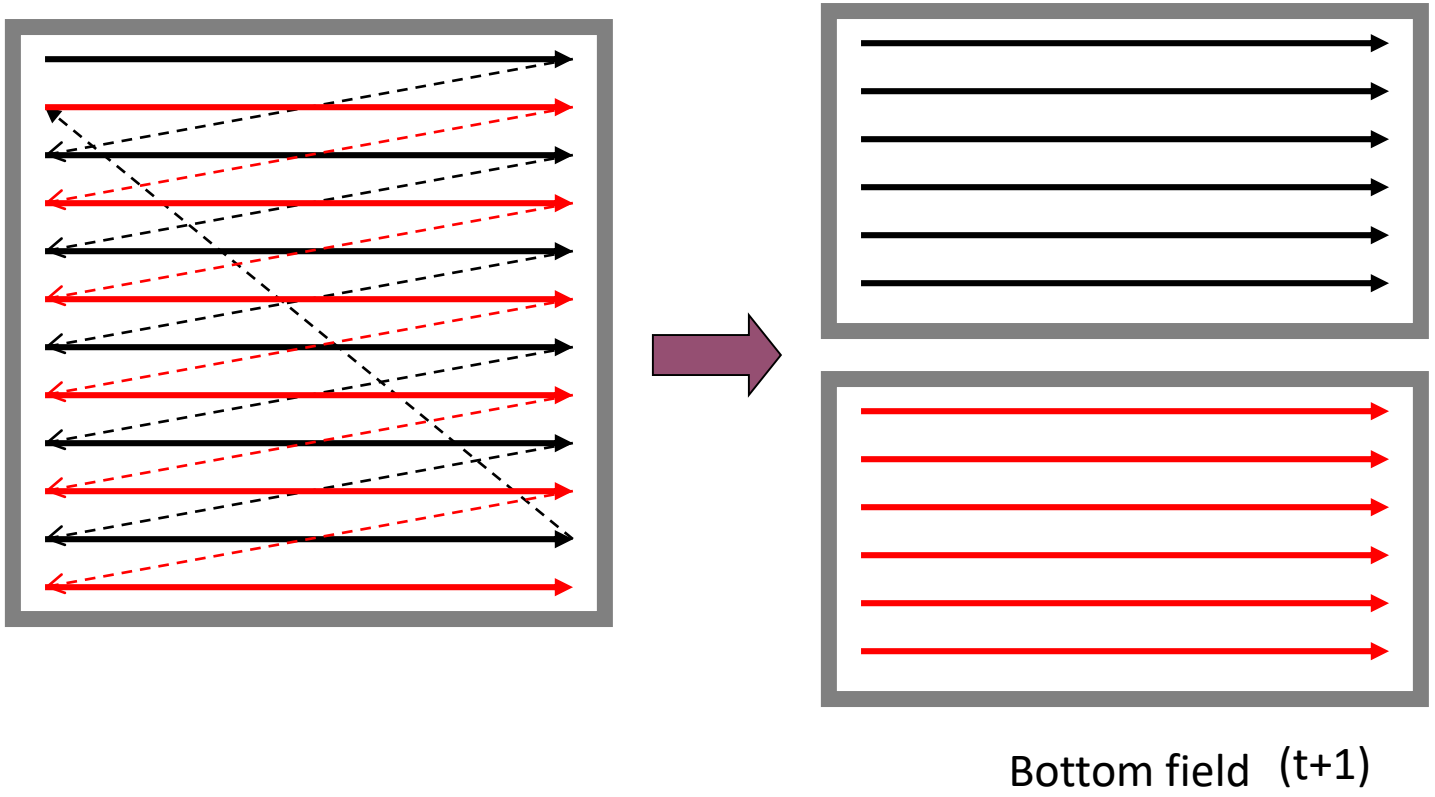
1. **Transform (T) + Quantize (Q)**:
   - These steps compress the residual (difference between actual frame and prediction).
   - **Lossy compression** happens here — details are thrown away to save space.
2. **Inverse Quantization (Q^{-1}) + Inverse Transform (T^{-1})**:
   - Reconstructs an **approximate version** of the residual.
   - **Added to motion-compensated reference** → gives **reconstructed frame**.
3. ✅ **Store this reconstructed frame** in the buffer — not the original one —
→ Because this is **exactly what the decoder will reconstruct**.

🎯 Summary:

- You can't buffer the original frame, because the decoder will never see it.
- You must buffer the **same decoded version** the decoder will use — based on quantized data.
- This ensures **predictive references stay aligned** between encoder and decoder.

Understanding Interlacing (not popular these days)

- Interlacing is a technique where each video frame is divided into two fields
  - one containing the odd-numbered lines and the other containing the even-numbered lines.
  - These fields are displayed in rapid succession, creating the illusion of a complete image.
- **Purpose:** Originally developed to improve motion portrayal without increasing bandwidth, interlacing reduces flicker and conserves bandwidth by transmitting half the frame at a time.



Interlacing Scan vs Progressive Scan

- Interlacing
  - With the same bandwidth, it can double the perceived frame rate of a video display
  - Broadcast Television: Widely used in traditional analog TV systems, such as NTSC, PAL, and SECAM.
  - Early HDTV Formats: Adopted in initial high-definition television standards to maintain compatibility with existing broadcasting infrastructure.
  - Physical Media: Employed in formats like DVDs and some Blu-ray discs to accommodate older display technologies.

- non-interlaced (or progressive scan)
  - Progressive scan displays each frame in its entirety, sequentially rendering all lines from top to bottom.
  - With the same bandwidth, it only updates the display half as often

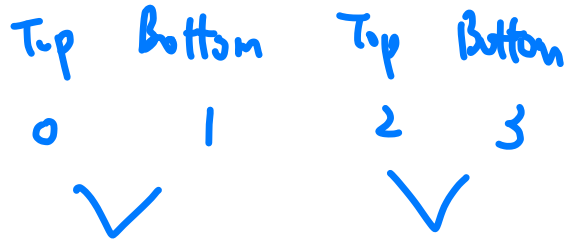
Interlacing Problem

- In progressive displays, **artifacts** can appear when displaying interlaced video, called **interlacing effects** or **combing**, especially during fast motion.
- Each interlaced video frame is composed of two fields captured at slightly different moments in time.
- When an object moves quickly, its position changes between the two fields. As a result, the object appears **disjointed or streaked** when viewed on a progressive display.
- The **combing effect** occurs when fast-moving objects appear with **horizontal jagged lines**, resembling a comb.

Need Deinterlacing

Figure is taken from https://en.wikipedia.org/wiki/Interlaced_video#/media/File:Interlaced_video_frame_(car_wheel).jpg

Deinterlacing - Weaving



- Field combination deinterlacing - Field Blending (Weaving)
  - Directly combining even and odd fields into one frame (Each field is captured at different times)
  - Naively, reducing the frame rate by half
  - Pros and Cons
    - **Pro: Fast and Simple** – No interpolation or complex processing required
    - **Con: Causing Interlacing Artifacts (Comb Effect)**
      - If motion exists between fields, fast-moving objects show **horizontal comb-like distortions** because the two fields were captured at different moments in time.
      - **Not Suitable for High Motion Videos**

Weaving

Interleaving even and odd fields into one frame

It may create jagged edges for moving scenes

Reference and figures: https://en.wikipedia.org/wiki/Deinterlacing

Deinterlacing – Frame Blending

- Instead of simply weaving the fields together (which can cause comb artifacts), **frame blending averages** pixel values from the odd and even fields.

$$P_{\text{blended}} = \frac{P_{\text{odd}} + P_{\text{even}}}{2}$$

- Create a **soft transition** between the two fields.
  - Pros
    - Minimizes the comb effect seen in weaving
    - Helps avoid sharp transitions between fields
  - Cons
    - Causes Blurry Output
    - Motion-heavy scenes may show "ghosts“
    - Blending does not reconstruct missing details but instead smooths over interlacing artifacts.

Blending (combined with a vertical resize)

Averaging even and odd fields to form one frame

It may generate ghosting artifacts for moving scenes

Deinterlacing - Half-sizing

- Field extension deinterlacing
  - extend one field to the entire screen to make a frame
  - To maintain correct proportions, the image is resized to half its original height
  - decrease the vertical resolution by half but maintain the original frame rate (field rate)
- Pros
  - Completely Removes Interlacing Artifacts
  - Good for Low-Detail Use Cases
- Cons
  - Loss of Vertical Resolution
  - Not Suitable for High-Quality Applications

Half-sizing

displays each interlaced field itself

Reference and figures: https://en.wikipedia.org/wiki/Deinterlacing

Deinterlacing - Line doubling

- **Duplicating Lines**
  - Instead of using both fields, **one field (odd or even) is retained**.
  - Each scanline (horizontal row of pixels) is **copied** and placed in the missing lines.
- Pros
  - Eliminates Interlacing Artifacts
  - Maintains Original Frame Size
- Cons
  - Since missing lines are duplicated, the vertical resolution is effectively halved
  - Soft or Blurry Appearance
  - **Jagged Edges (aliasing)** – Fine details and diagonal edges may appear rough.

Line doubling

doubles each interlaced field to fill up the entire frame.

Decreases vertical resolution and introduces visual anomalies

stationary objects can appear at different vertical locations when playing

Video Quality

- Subjective Quality Measurement
  - Double Stimulus continuous quality scale (DSCQS)

An assessor is presented with a pair of images or short video sequences, A and B, one after the other. Then, the assessor will be asked to grade the two using the form shown on the right hand side.

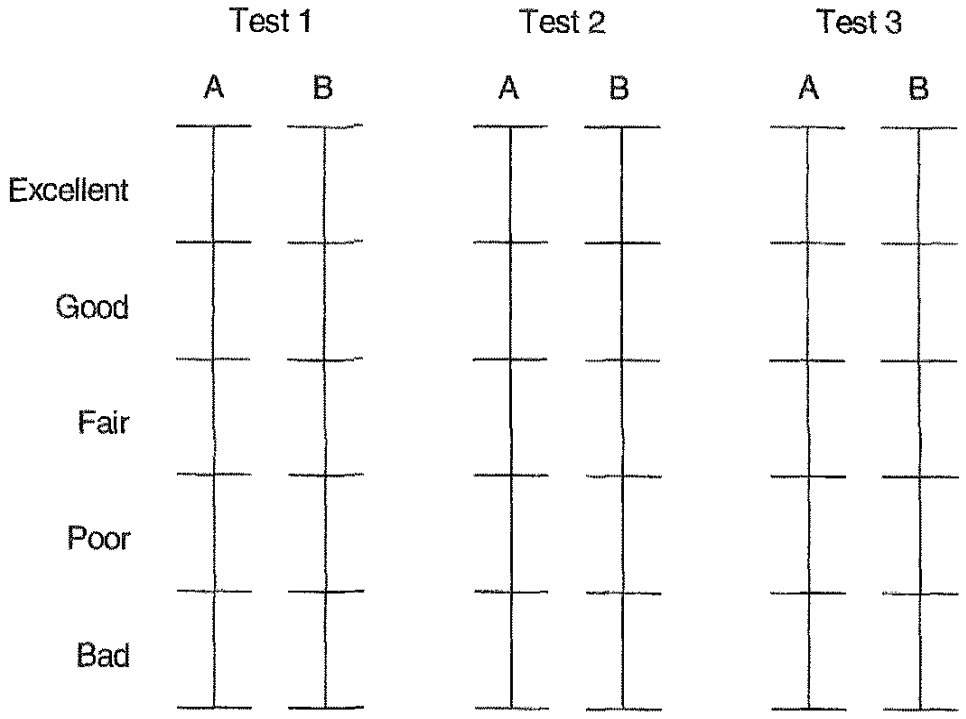


Figure 2.13 DSCQS rating form

REF: Video Codec Design: Developing Image and Video Compression Systems, Wiley

Video Quality

- Objective Quality Measurement
  - The most widely used metric
    - PSNR (Peak Signal-to-Noise Ratio)

The *Mean Square Error (MSE)* and the *Peak Signal to Noise Ratio (PSNR)* are the two most often used error metrics for comparing image compression quality.

$Let\ I_1$ and I_2 be two images. I_2 is a compressed version of I_1. $I_1, I_2 \in R^{M \times N}$

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N}$$

$$PSNR = 10 \log_{10} \frac{(2^n - 1)^2}{MSE}$$

↗ $2^n - 1$: maximal pixel value

(1. the larger PSNR the better
2. often expressed using dB)

REF: https://www.mathworks.com/help/vision/ref/psnr.html

🧮 Toy Example of PSNR Calculation

Let's say:

- You have two 2×2 grayscale images:
  - Original $I_1 = \begin{bmatrix} 100 & 100 \\ 100 & 100 \end{bmatrix}$
  - Compressed $I_2 = \begin{bmatrix} 98 & 102 \\ 99 & 101 \end{bmatrix}$
- 8-bit image → max value = 255 → $\text{MAX}_I = 255$

Step 1: Compute MSE

$$\text{MSE} = \frac{(100-98)^2 + (100-102)^2 + (100-99)^2 + (100-101)^2}{4} = \frac{4+4+1+1}{4} = \frac{10}{4} = 2.5$$

Step 2: Plug into PSNR

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{255^2}{2.5}\right) = 10 \cdot \log_{10}\left(\frac{65025}{2.5}\right) = 10 \cdot \log_{10}(26010) \approx 10 \cdot 4.414 = 44.14\,\text{dB}$$

✅ **Result**: PSNR ≈ **44.14 dB** — which is considered **very good quality**.

🔷 MSE (Mean Squared Error)

- **Unit: Pixel intensity squared**
- Example:
  - For an 8-bit grayscale image (pixel values from 0 to 255), the unit is **"gray levels²"**.
  - So if pixels differ by 5, the squared error is $5^2 = 25$ (unit: intensity²).
- It's a **unit-dependent** value based on pixel range.

🔷 PSNR (Peak Signal-to-Noise Ratio)

- **Unit: Decibels (dB)**
- It's a **logarithmic** measure, so it's **unitless** in content, but we write it in **dB** to show it's a **ratio**.

Summary:

Metric	Unit	Notes
MSE	intensity² (e.g., gray levels²)	Based on squared pixel differences
PSNR	dB (decibels)	Logarithmic measure of signal quality

PSNR Pitfalls

PSNR can be ineffective to assessing perceptual quality