Understanding color & the in-camera image processing pipeline for computer vision

Dr. Michael S. Brown

Canada Research Chair Professor

York University - Toronto

ICCV 2019 Tutorial – Seoul, Korea

Tutorial schedule

- Part 1 (General)
  - Motivation
  - Review of color & color spaces
  - Overview of in-camera imaging pipeline

1.30pm – 3.30pm

Break
3.30pm – 4.30pm

- Part 2 (Imaging and Computer Vision)
  - Misconceptions in the computer vision community regarding color
  - Recent work on color and cameras
  - Concluding remarks

4.30pm – 6.00pm

Tutorial schedule

- Part 1 (General)
  - Motivation
  - Review of color & color spaces
  - Overview of in-camera imaging pipeline

Beginner audience
Covers color and camera.
General knowledge topic.
Not specific to any papers.

- Part 2 (Imaging and Computer Vision)
  - Misconceptions in the computer vision community regarding color
  - Recent work on color and cameras
  - Concluding remarks

Intermediate audience
Covers recent research on camera pipelines -- assumes you have a background in computer vision.
(Note: Includes papers from my research lab)

Part 1: Motivation for this tutorial?

Shifting landscape of cameras

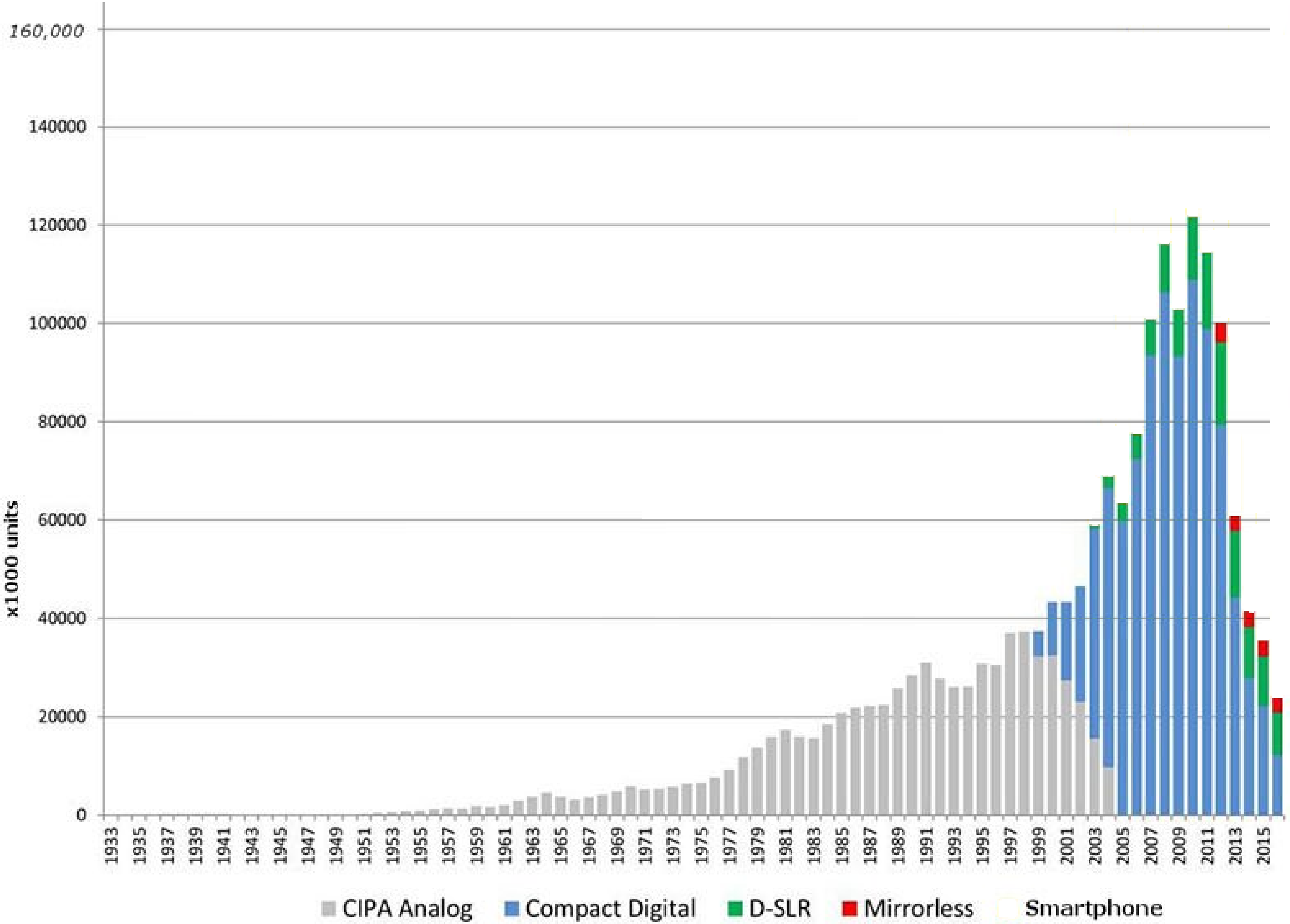
Film

Point-and-shoot

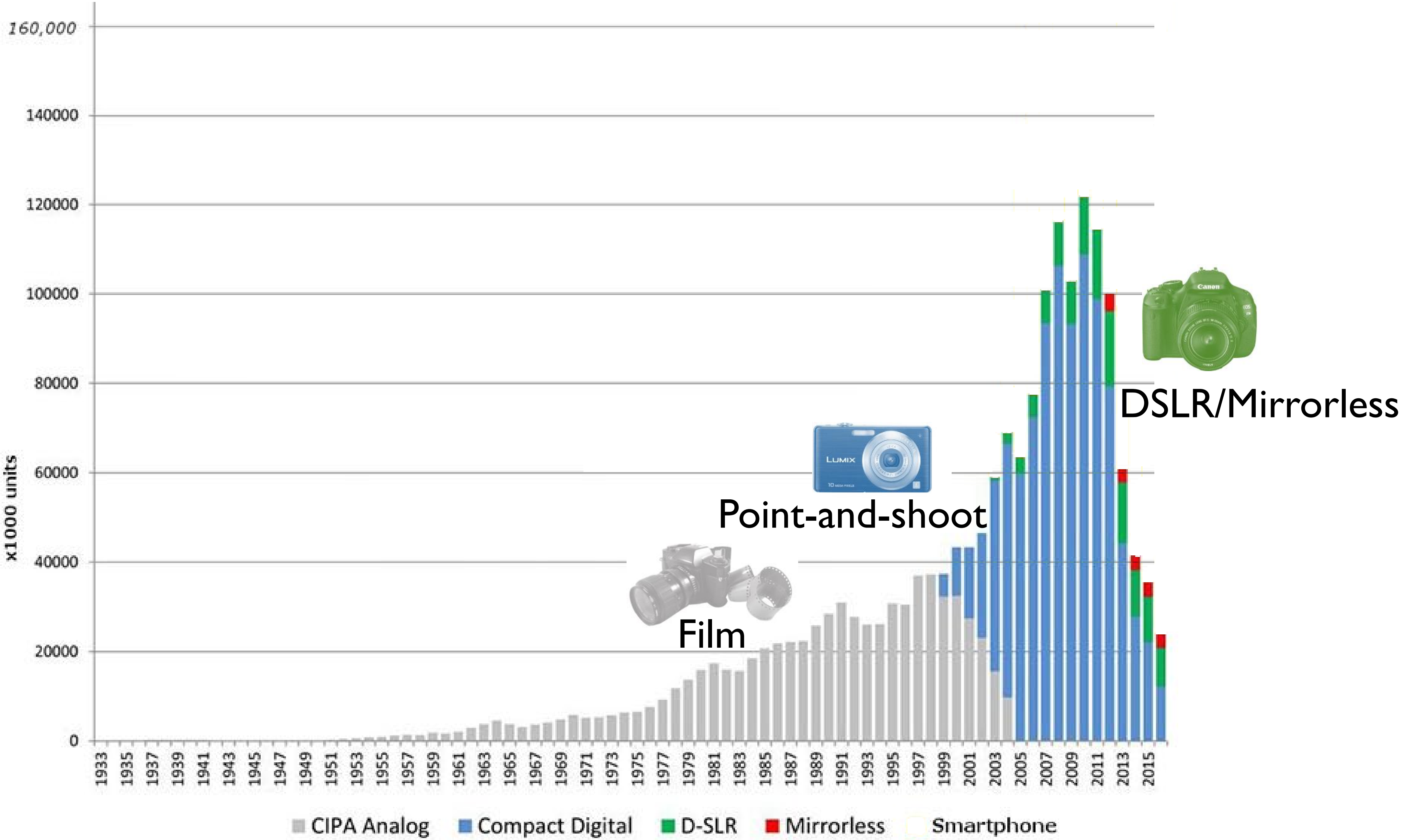
DSLR/Mirrorless

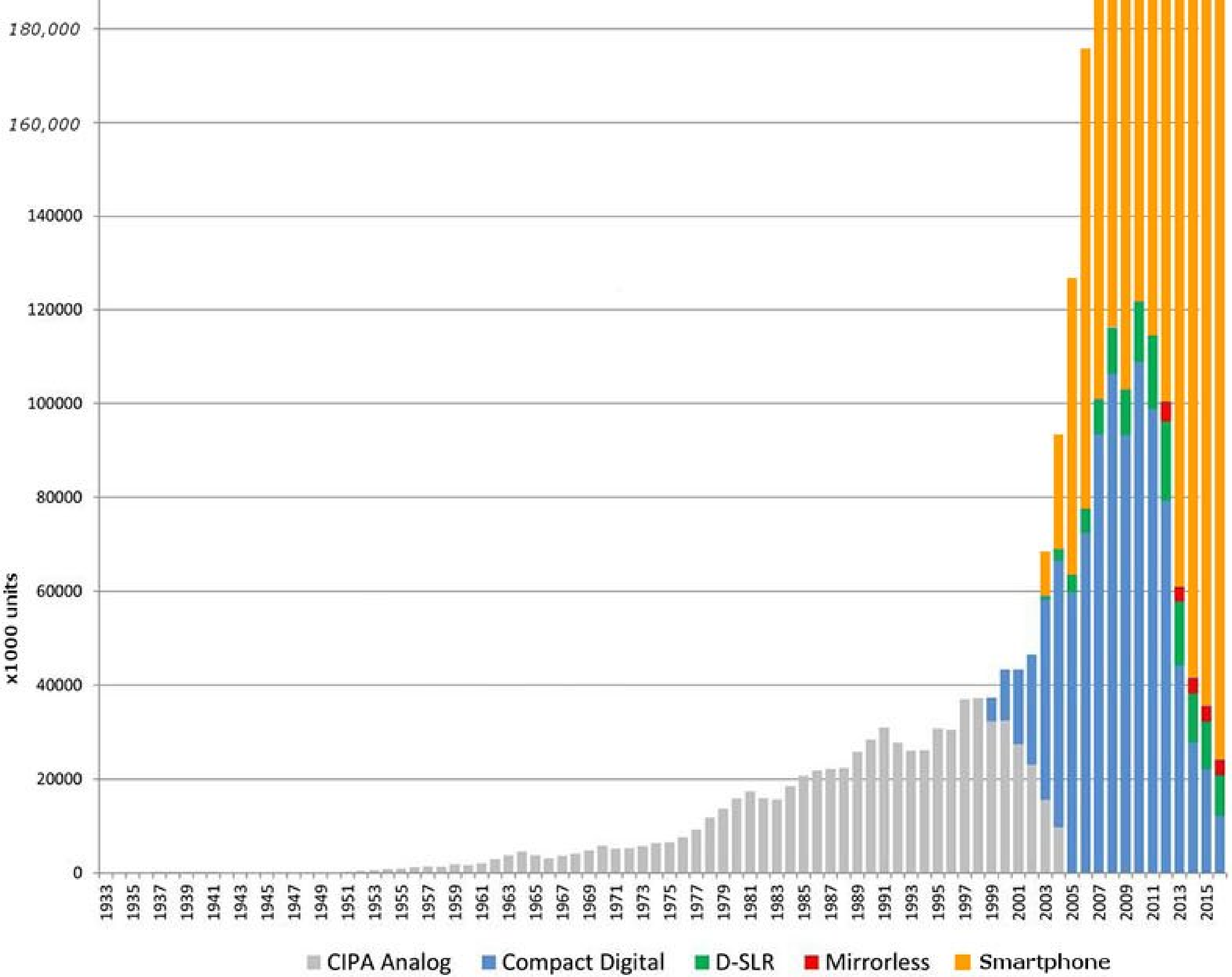
Smartphone

Units of each device sold 1933 - 2016

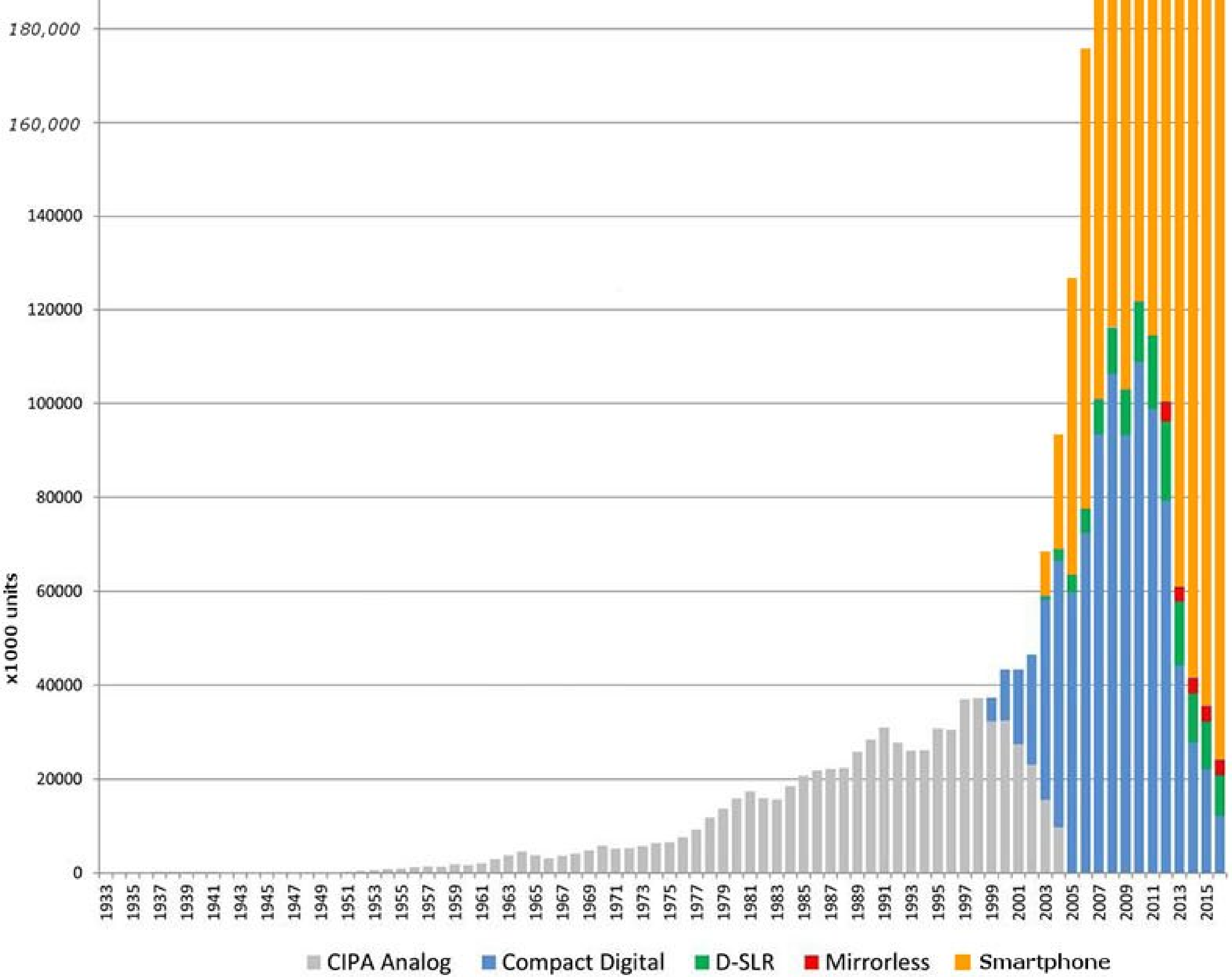


Units of each device sold 1933 - 2016




180,000
160,000
140000
120000
100000
80000
60000
40000
20000
0
x1000 units
1933 1935 1937 1939 1941 1943 1945 1947 1949 1951 1953 1955 1957 1959 1961 1963 1965 1967 1969 1971 1973 1975 1977 1979 1981 1983 1985 1987 1989 1991 1993 1995 1997 1999 2001 2003 2005 2007 2009 2011 2013 2015
CIPA Analog
Compact Digital
D-SLR
Mirrorless
Smartphone

Smartphone



180,000
160,000
140000
120000
100000
80000
60000
40000
20000
0
x1000 units
1933
1935
1937
1939
1941
1943
1945
1947
1949
1951
1953
1955
1957
1959
1961
1963
1965
1967
1969
1971
1973
1975
1977
1979
1981
1983
1985
1987
1989
1991
1993
1995
1997
1999
2001
2003
2005
2007
2009
2011
2013
2015
CIPA Analog
Compact Digital
D-SLR
Mirrorless
Smartphone

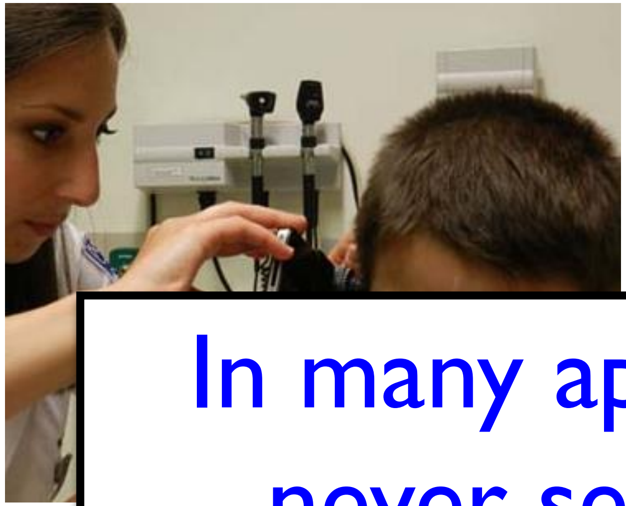
Smartphone

Photography = smartphone camera

Imaging is at our finger tips

Not always a good thing . . .

And useful for purposes beyond photography



In many applications, a human will never see the captured image.



Scientist’s view of photography

Photo by Uwe Hermann

Scientist's view of photography

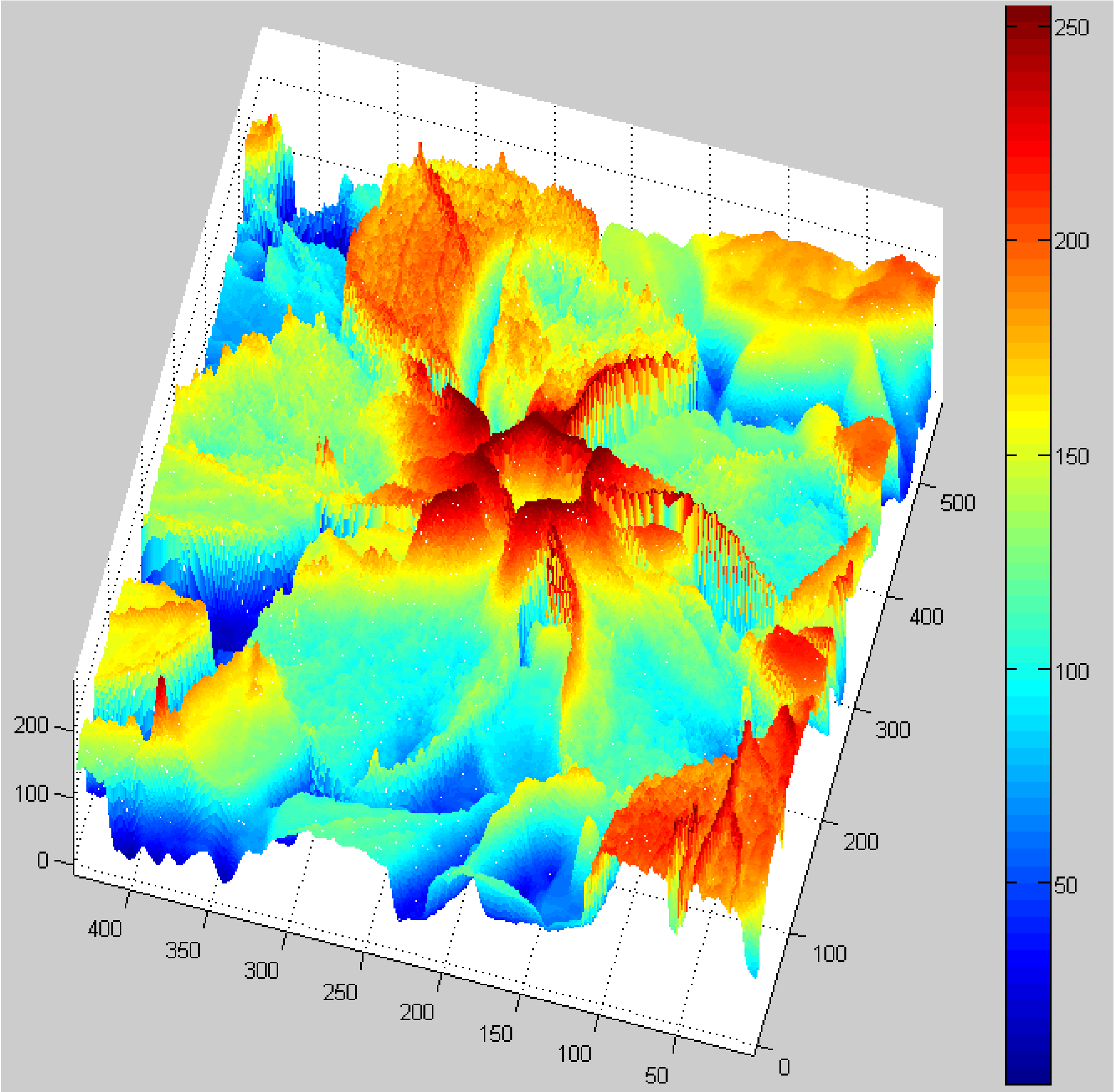
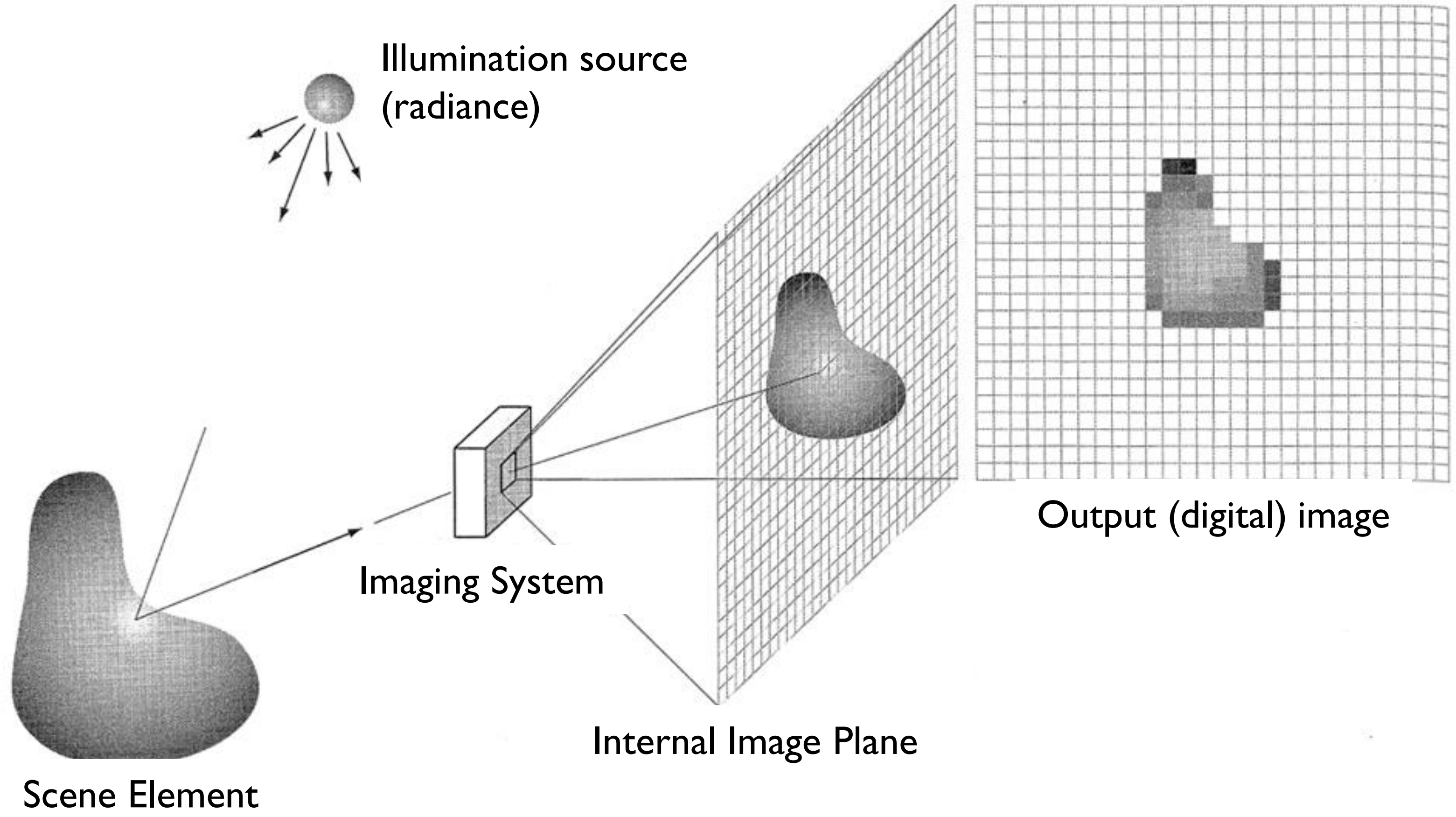


Photo by Uwe Hermann

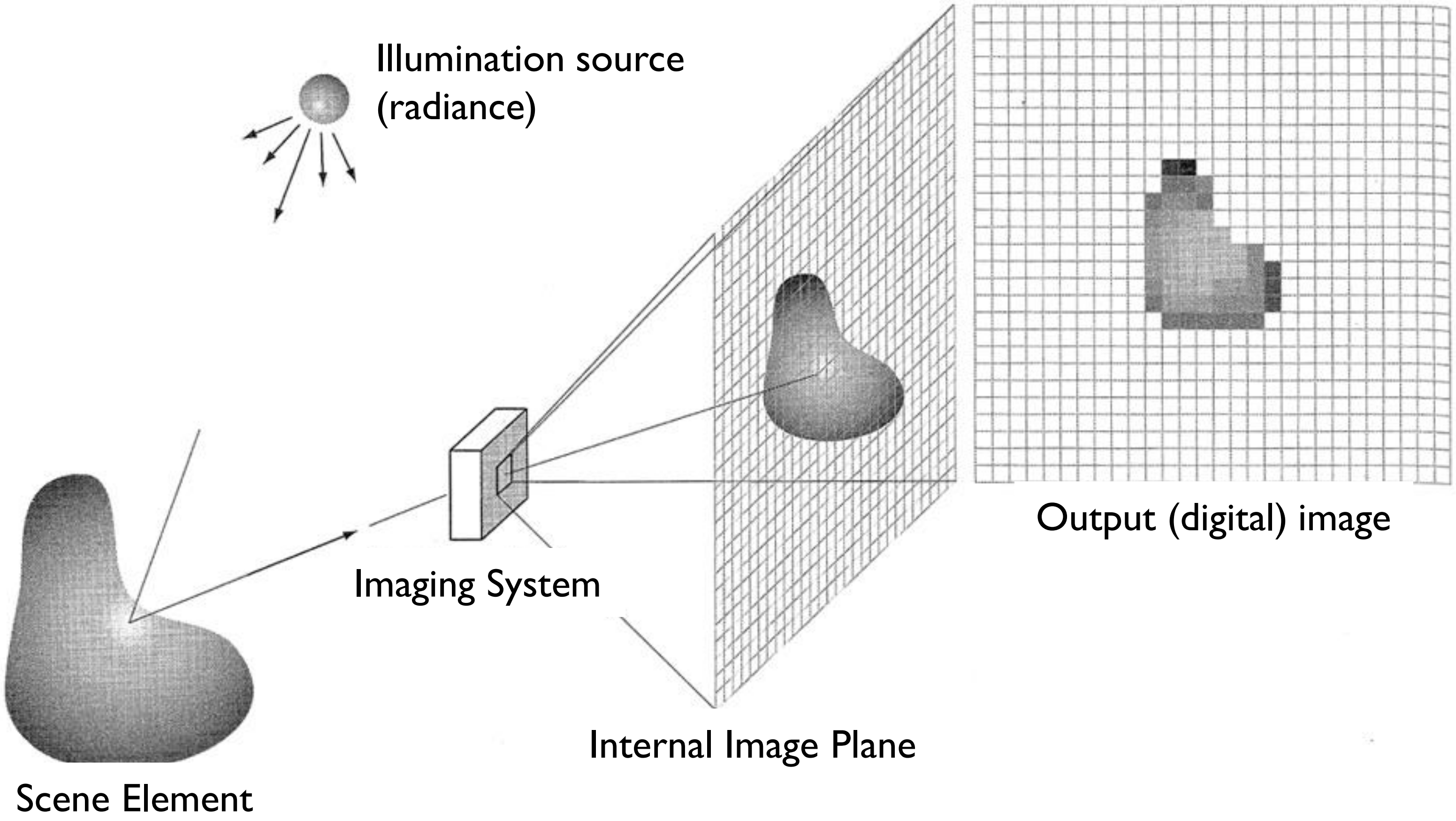
Camera = light-measuring device



Simple models of a camera assumes an image is a "quantitative measurement" of scene radiance.

Figure from Digital Image Processing, Gonzales/Woods

Image = radiant energy measurement



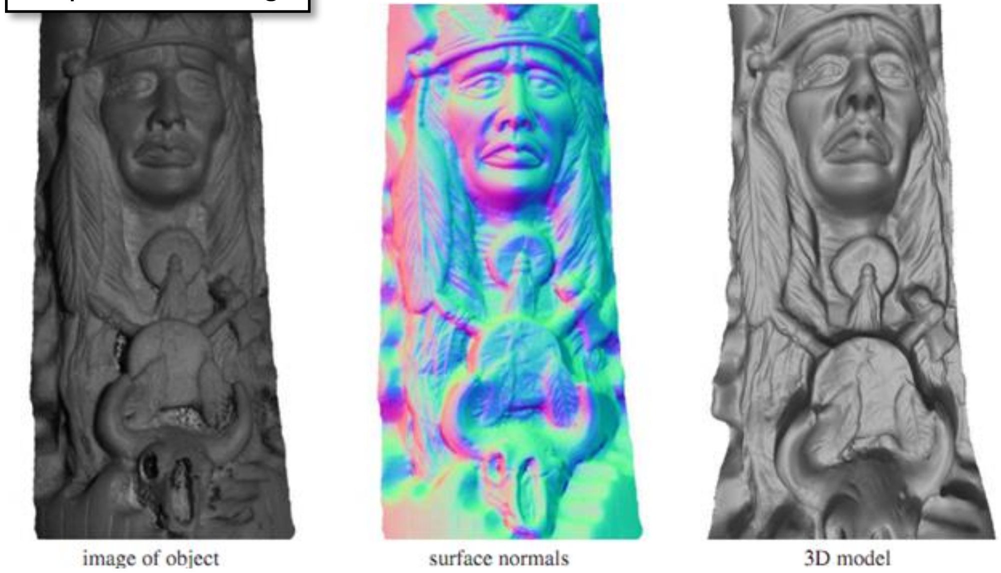
Simple models of a camera assumes an image is a "quantitative measurement" of scene radiance.

Figure from Digital Image Processing, Gonzales/Woods

This assumption is made often in computer vision

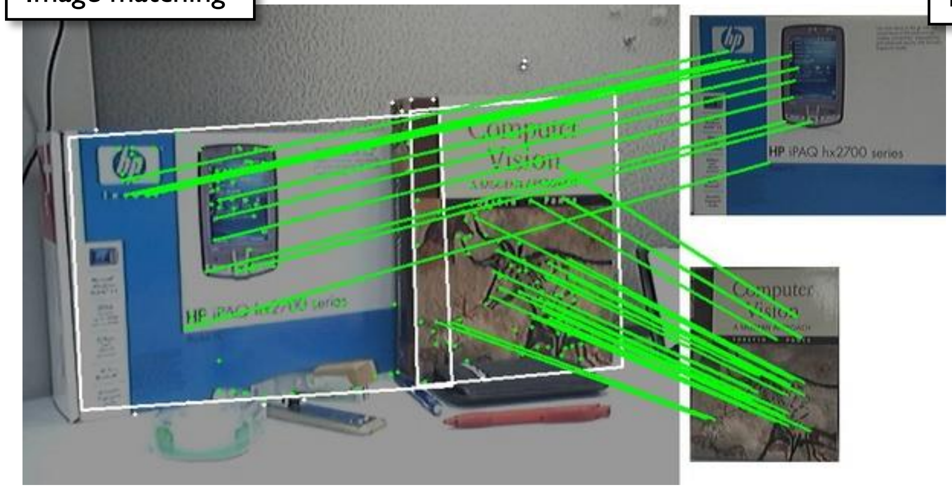
- Shape from shading
- HDR imaging
- Image matching
- Color constancy
- Applications relying on color
- Image delubrring
- Etc . . .

Shape-from-shading



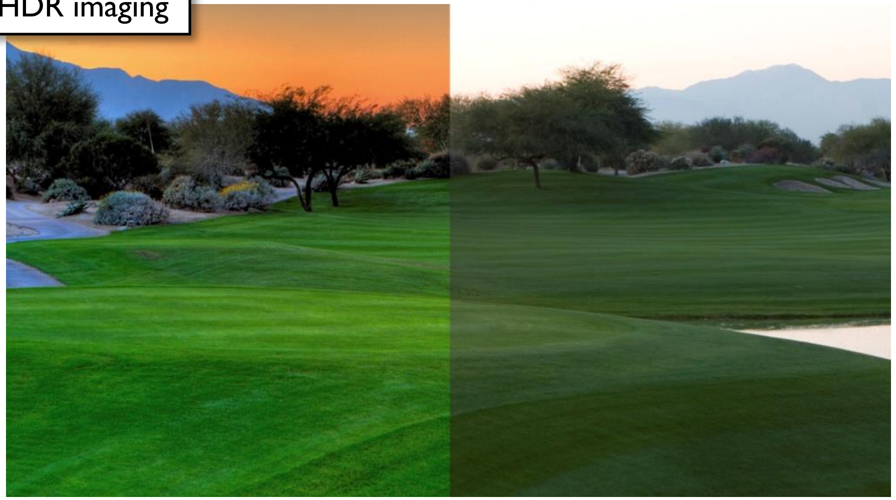
From Lu et al, CVPR'10

Image matching



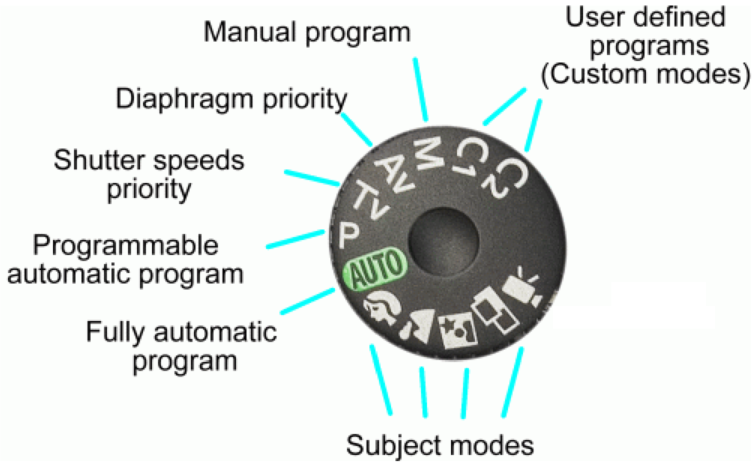
From Jon Mooser, CGIT Lab, USC

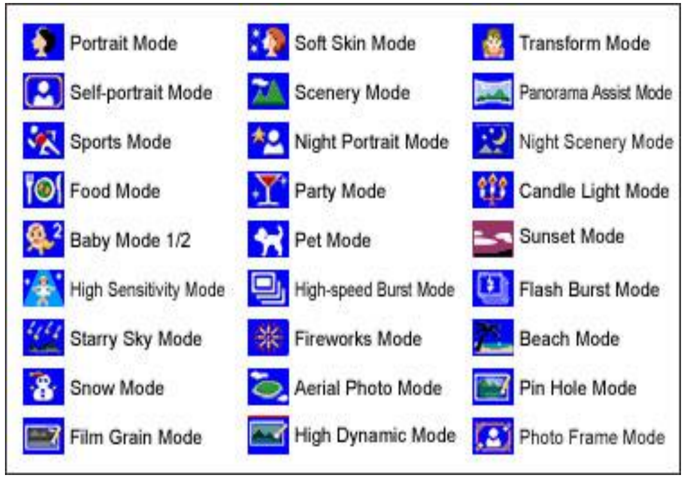
HDR imaging



From O'Reilly's digital media forum

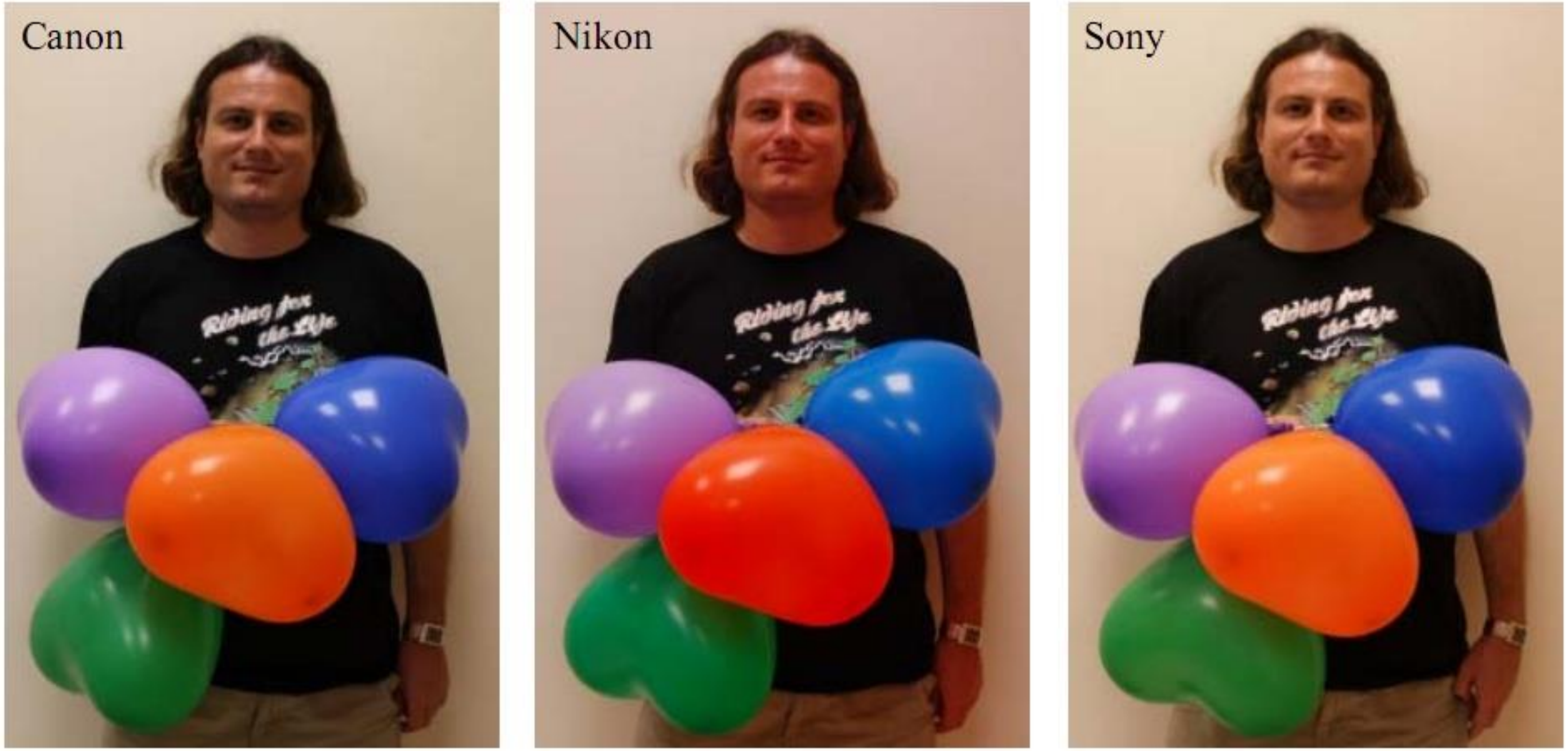
Camera = light-measuring device?





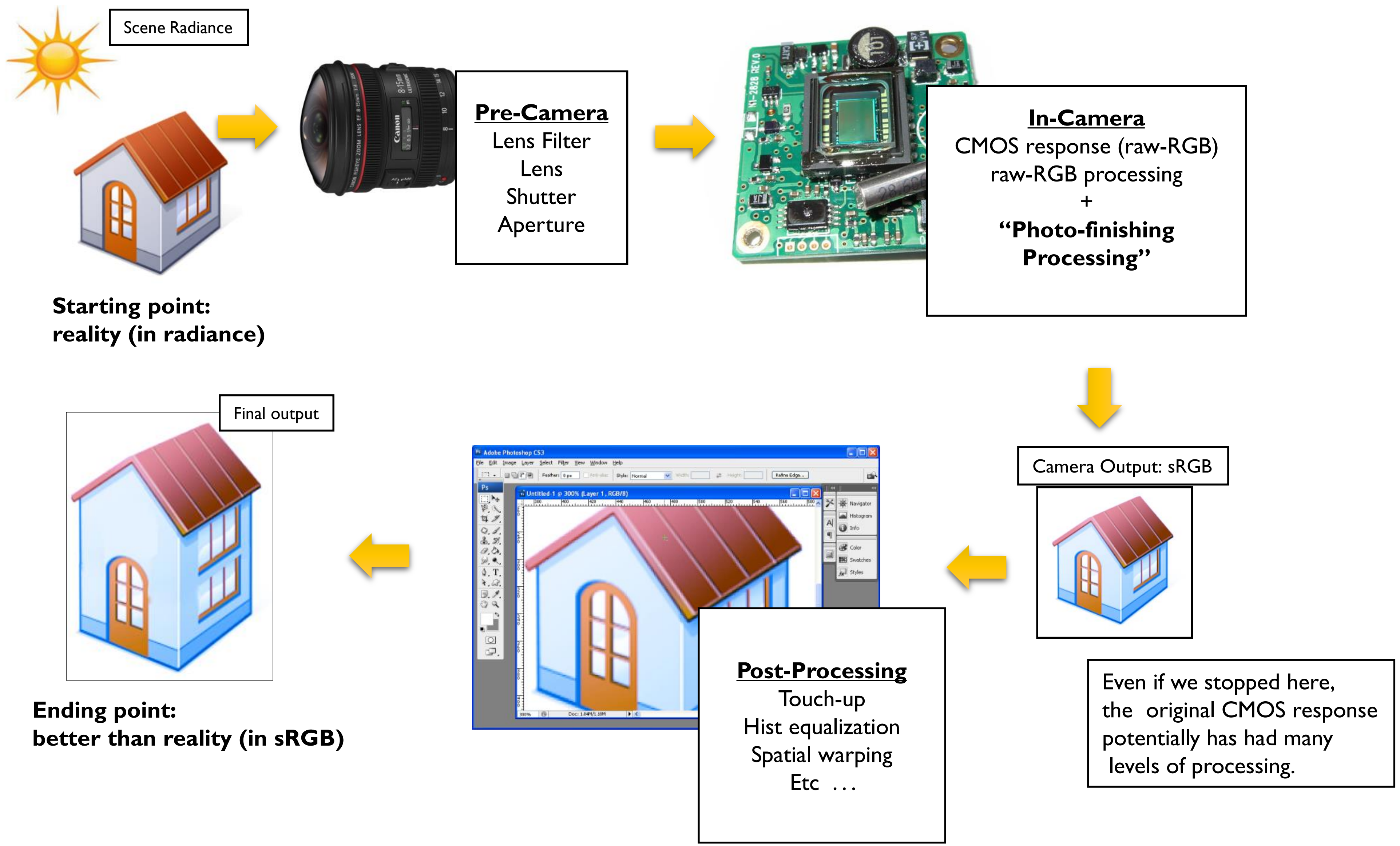
Camera pipeline photo-finishing routines

“Secret recipe” of a camera



Photographs taken from three different cameras with the same aperture, shutter speed, white-balance , ISO, and picture style.

Modern photography pipeline



Digital cameras

- Digital cameras ***are not designed to be*** light-measuring devices
- **They are designed to produce visually pleasing photographs**
- There is a great deal of processing (photo-finishing) applied in the camera hardware

The goal of this tutorial is to discuss common processing steps that take place onboard consumer cameras

Tutorial schedule

- Part 1 (General)
  - ~~Motivation~~
  - Review of color & color spaces
  - Overview of in-camera imaging pipeline

1.30pm – 3.30pm

Break
3.30pm – 4.30pm

- Part 2 (Imaging and Computer Vision)
  - Misconceptions in the computer vision community regarding color
  - Recent work on color and cameras
  - Concluding remarks

4.30pm – 6.00pm

Part 1: "Crash Course" on Color & Color Spaces

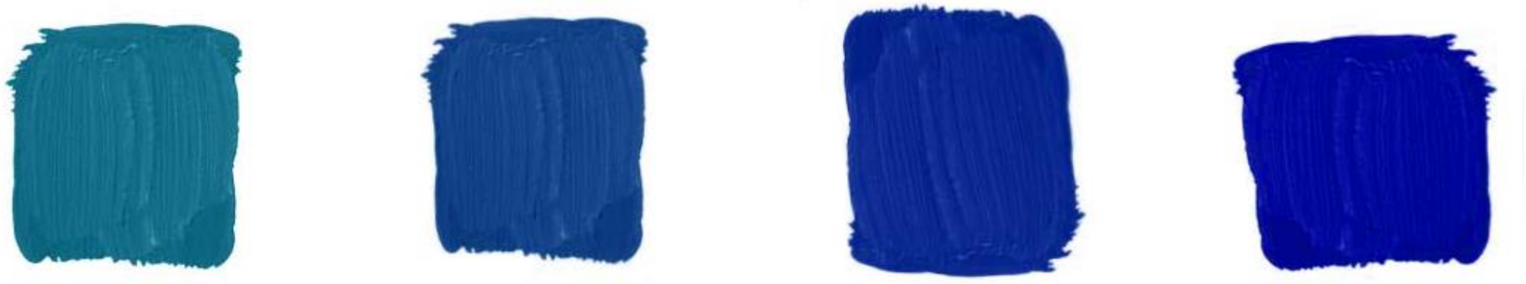
Color

Def *Color* (noun): The property possessed by an object of producing different sensations on the eye as a result of the way it reflects or emits light.

Oxford Dictionary

Color is perceptual

- **Color is not** a primary *physical* property on an object
- Red, Green, Blue, Pink, Orange, Atomic Tangerine, Baby Pink, etc. . .
  - These are words we assign to human color sensations



Which is the "true blue"?

http://en.wikipedia.org/wiki/List_of_colors

Subjective terms to describe color

Hue
Name of the color
(yellow, red, blue, green, . . .)

Value/Lightness/Brightness
How light or dark a color is.

Saturation/Chroma/Color Purity
How "strong" or "pure" a color is.

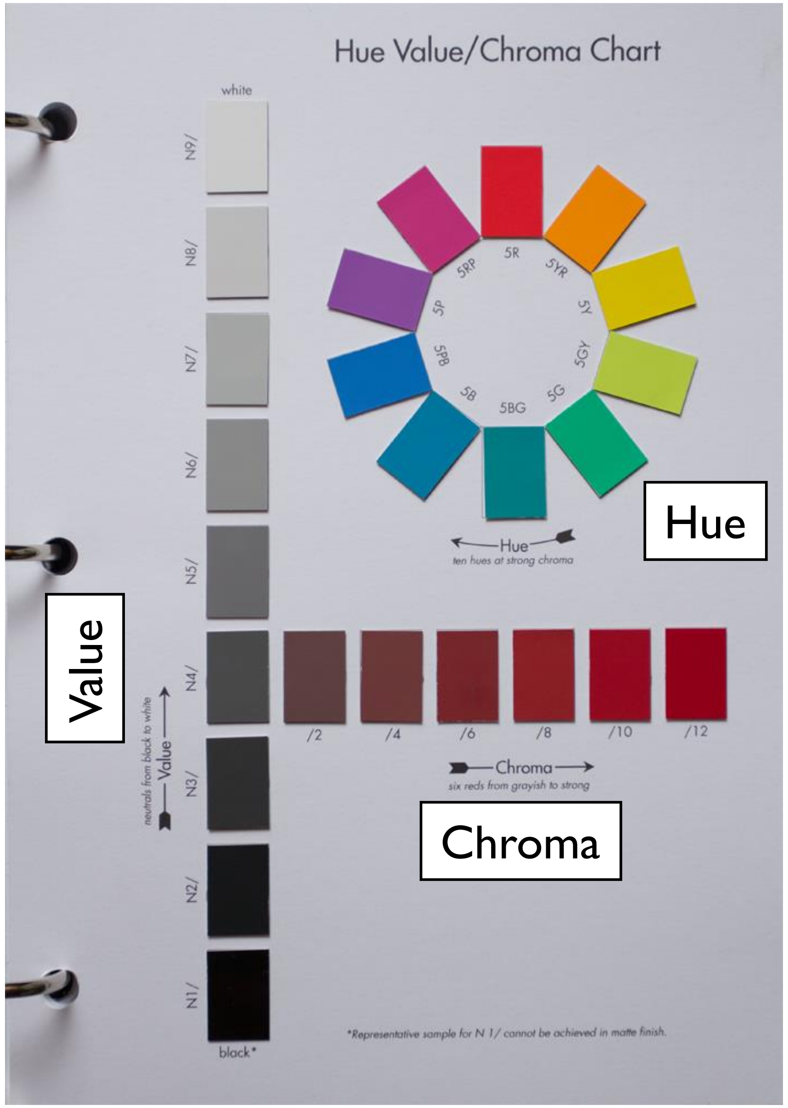
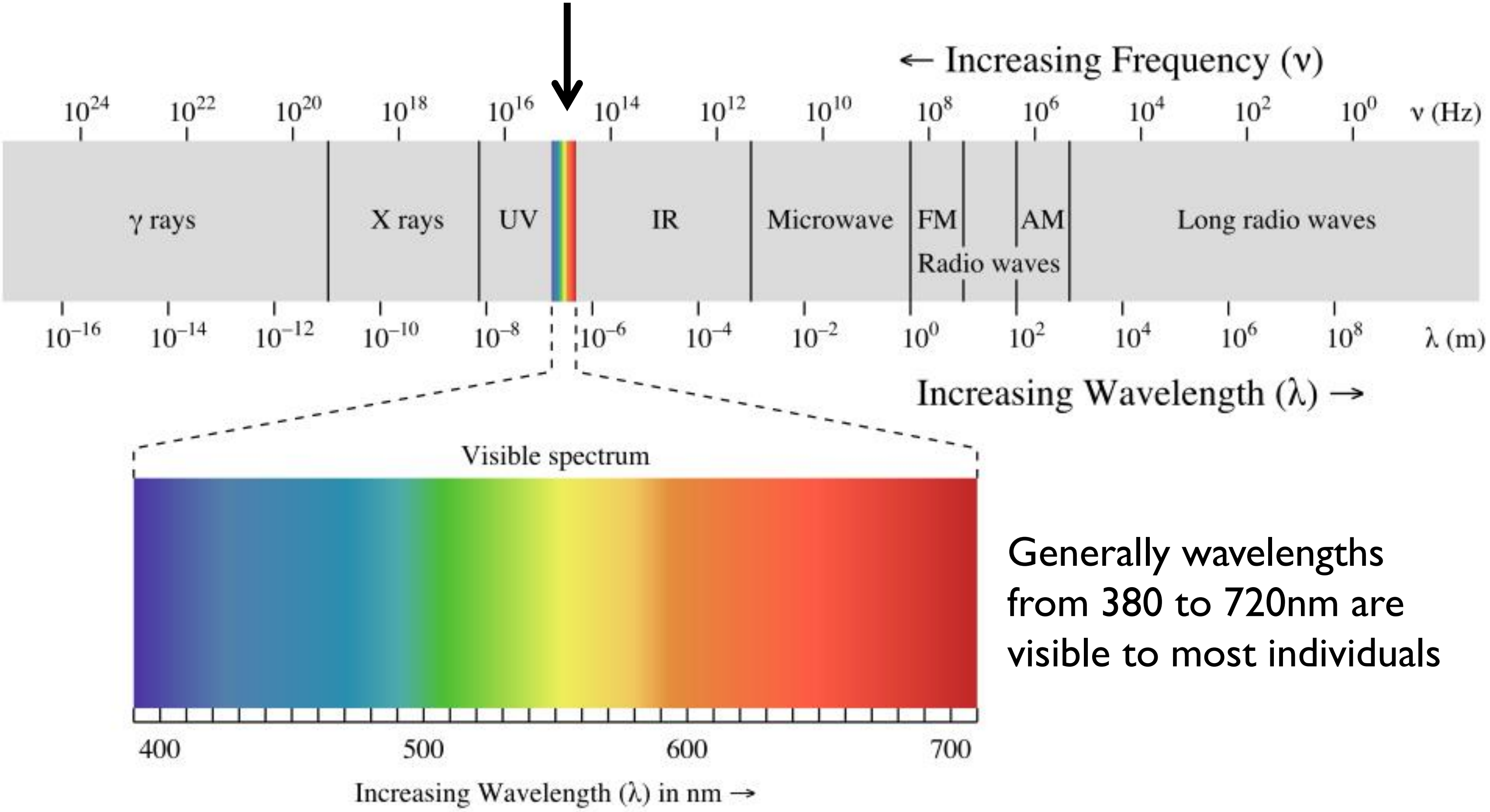


Image from Benjamin Salley
A page from a Munsell Student Color Set

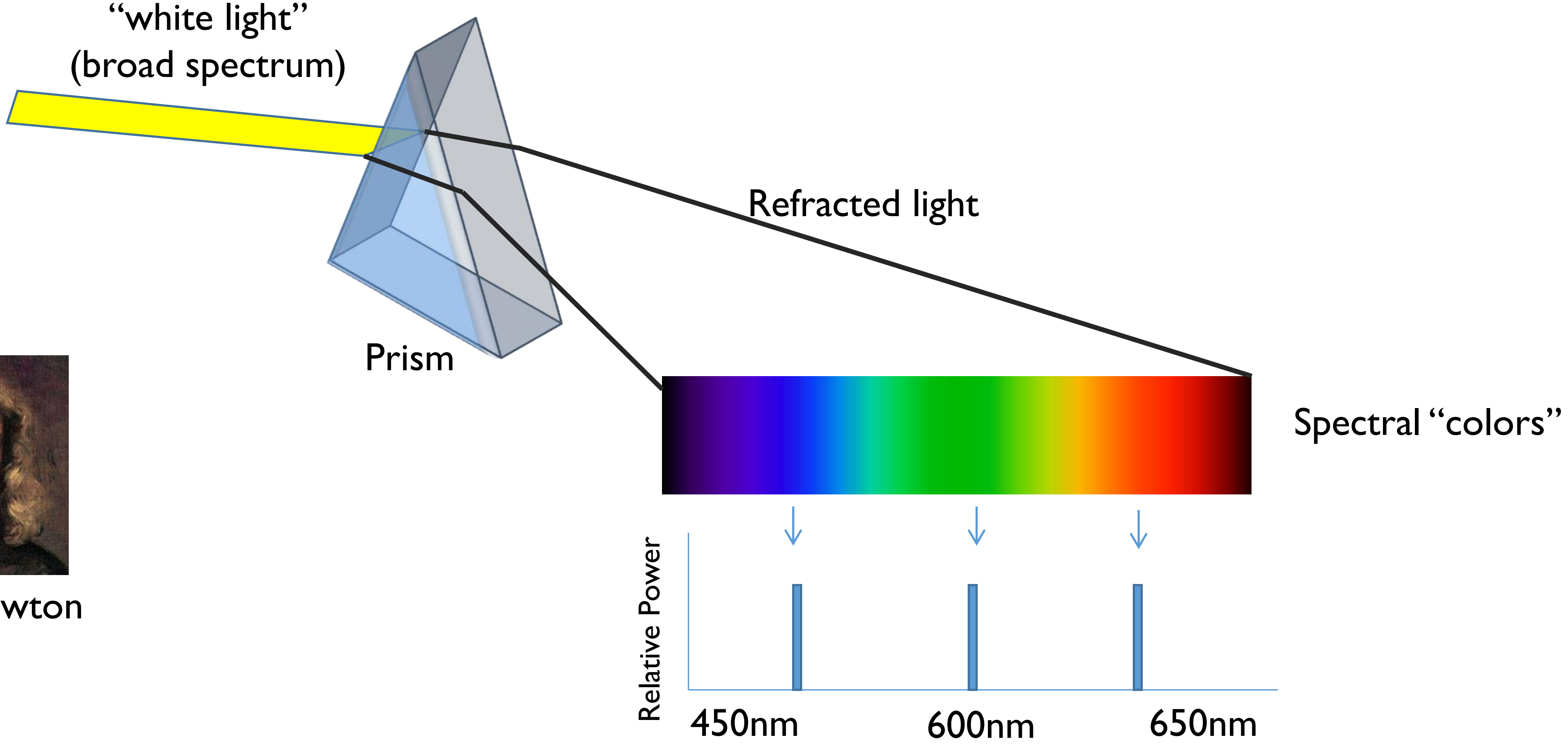
Where do "color sensations" come from?

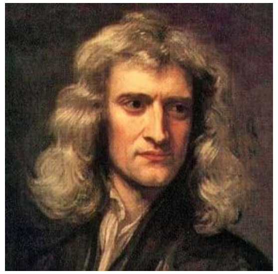
A *very* small range of electromagnetic radiation



Generally wavelengths from 380 to 720nm are visible to most individuals

White light through a prism

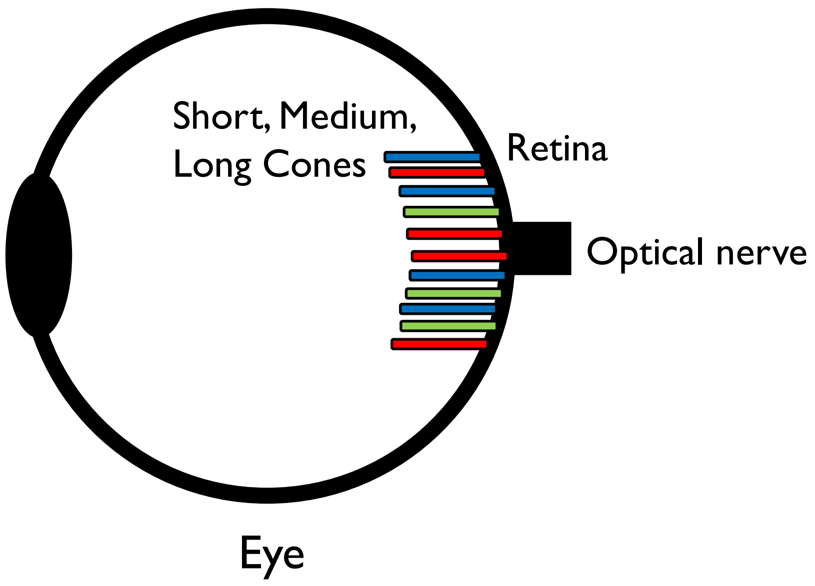


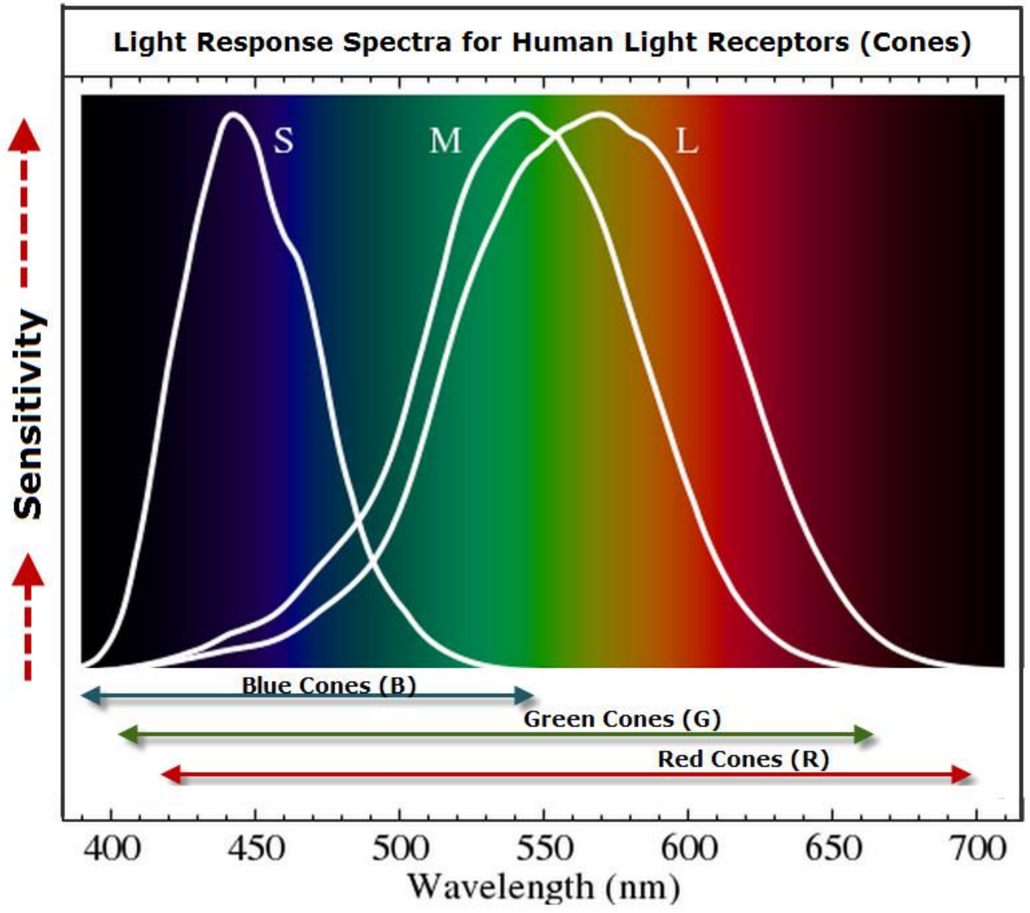

Isaac Newton

Light is separated into "monochromatic" light at different wave lengths.

Biology of color sensations

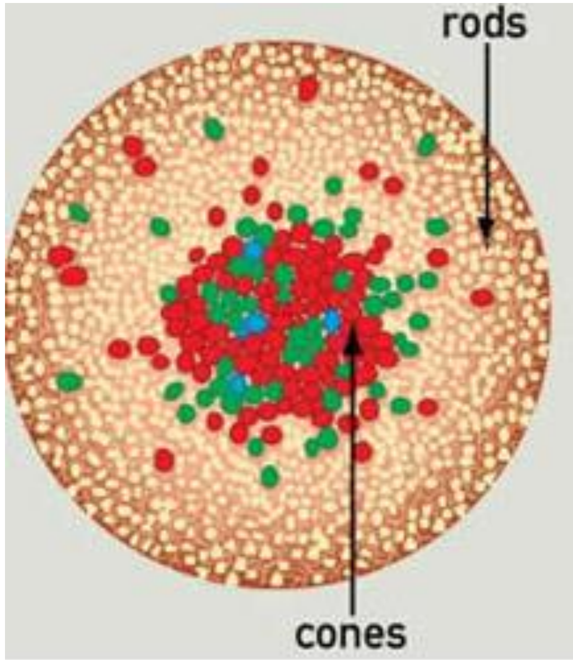
- Our eye has three receptors (cone cells) that respond to visible light and give the sensation of color

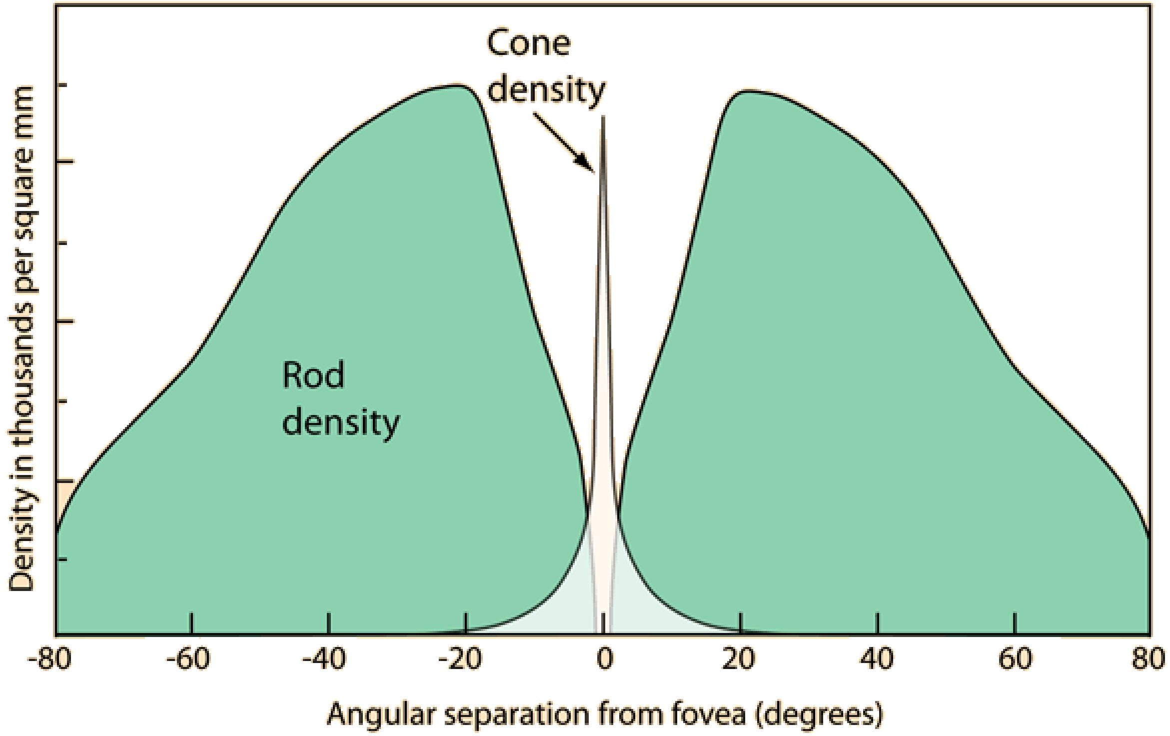




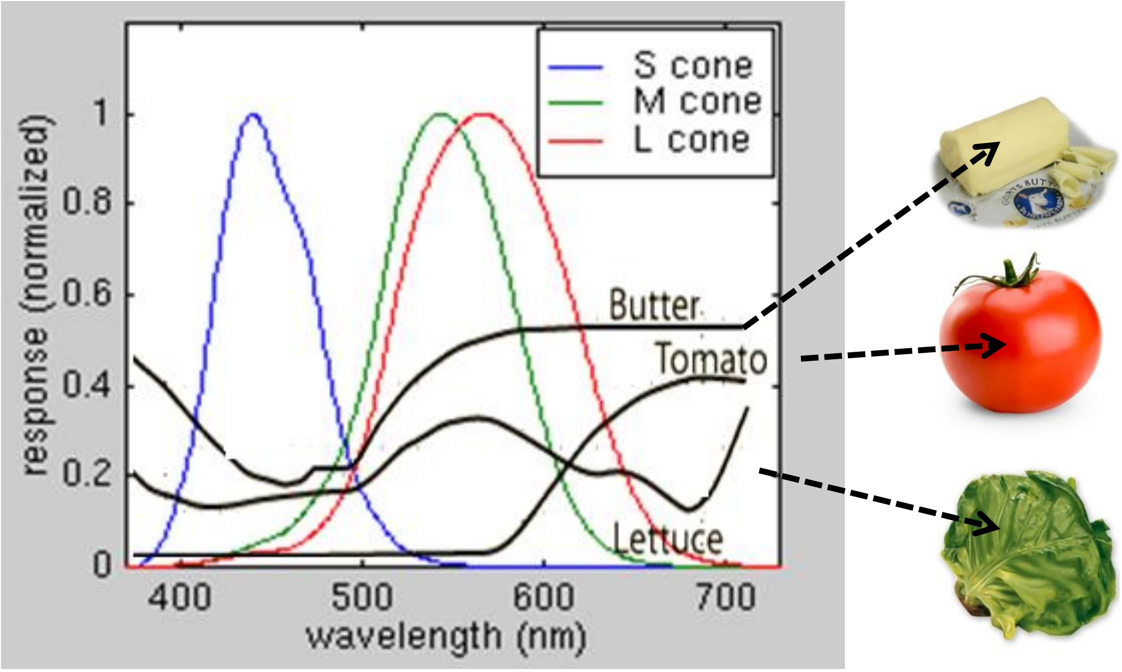
Cones and rods

- We have additional light sensitive cells called *rods* that are not responsible for color. Rods are used in low-light vision.
- Cone cells are most concentrated around the fovea of the eye





Spectral power distribution (SPD)

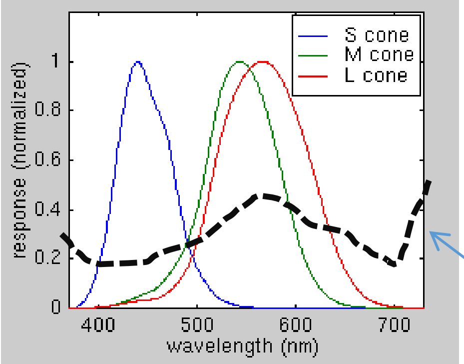


We rarely see monochromatic light in real world scenes. Instead, objects reflect a wide range of wavelengths. This can be described by a ***spectral power distribution*** (SPD) shown above. The SPD plot shows the relative amount of each wavelength reflected over the visible spectrum.

SPD relation to color is not unique

- Due to the accumulation effect of the cones, two different SPDs can be perceived as the same color (such SPDs are called "metamers").

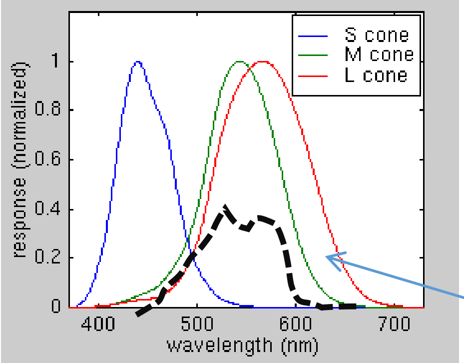
Lettuce SPD



Lettuce SPD stimulating S=0.2, M=0.8, L=0.8

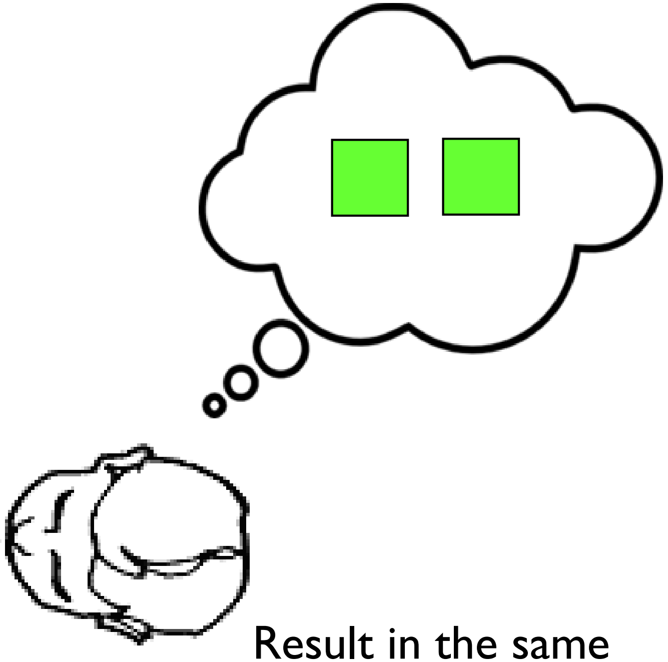
SPD of "real lettuce"

Green Ink SPD



Green ink SPD stimulating S=0.2, M=0.8, L=0.8

SPD of ink in a "picture of lettuce"



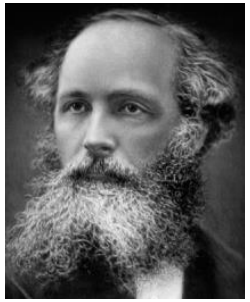
Result in the same color "sensation".

Tristimulus color theory

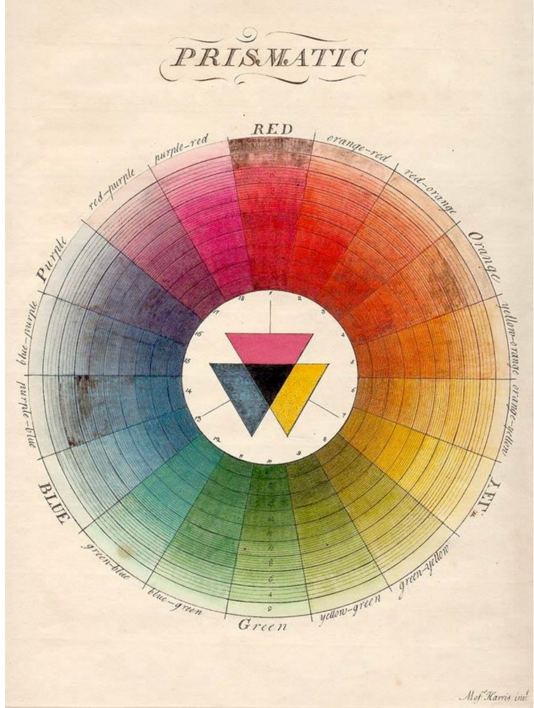
- Before the biology of cone cells was understood, it was empirically known that only three distinct colors (primaries) could be mixed to produce other colors
- Thomas Young (1803), Johann Wolfgang von Goethe (1810), Hermann Grassman (1853), James Maxwell (1856) all explored the theory of trichromacy for human vision


Thomas Young


Hermann Grassman

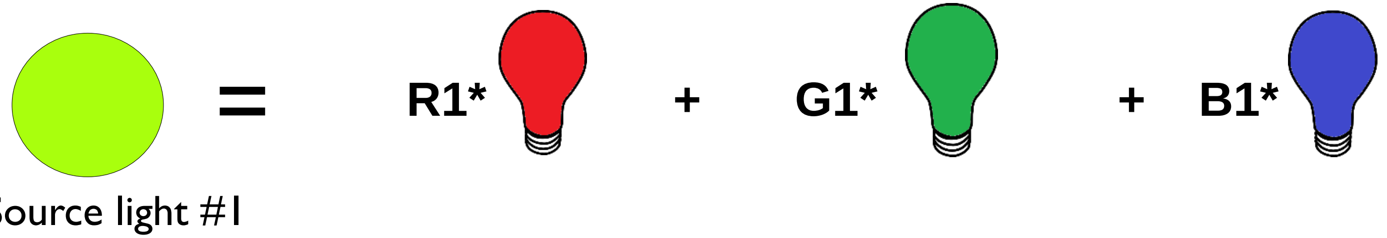

James Maxwell


von Goethe

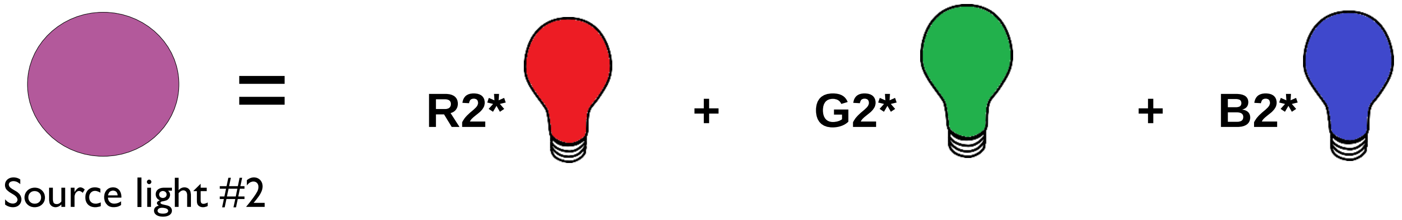


Tristimulus color theory

Grassman's Law states that a source color can be matched by a **linear** combination of three independent "primaries".



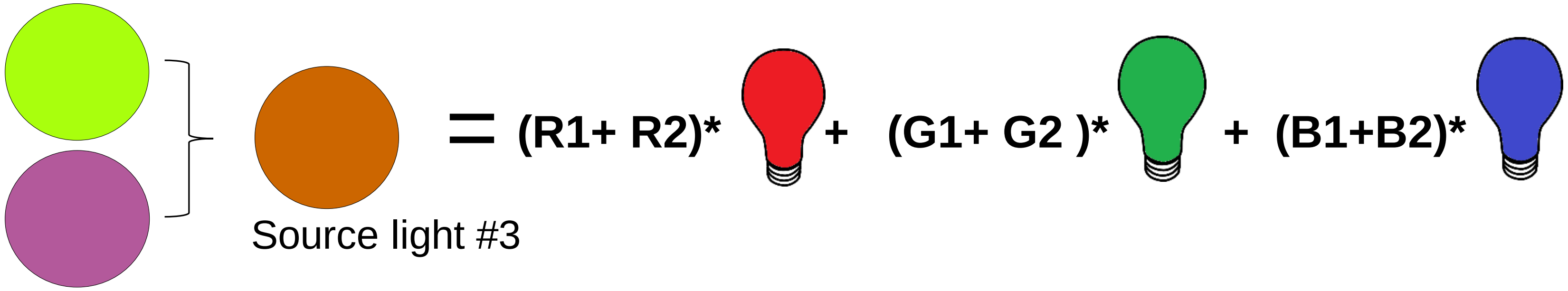
Three lights (shown as lightbulbs) serve as primaries. Each light has intensity, or weights, R1, G1, B1 to match the source light #1 perceived color.



Same three primaries and the weights (R2, G2, B2) of each primary needed to match the source light #2 perceived color

If we combined source lights 1 & 2 to get a new source light 3

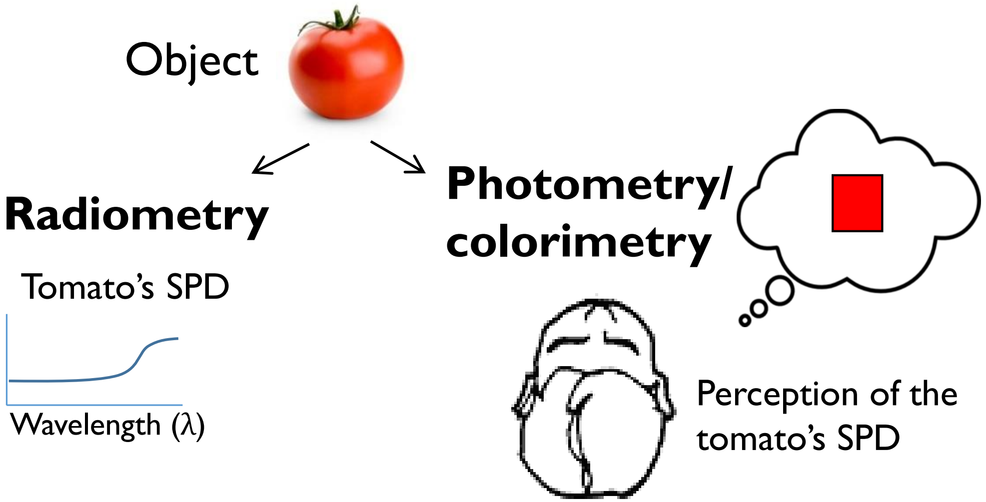
The amount of each primary needed to match the new source light #3 is the sum of the weights that matched lights sources #1 & #2.



This may seem obvious now, but discovering that light obeys the laws of linear algebra was a huge and useful discovery.

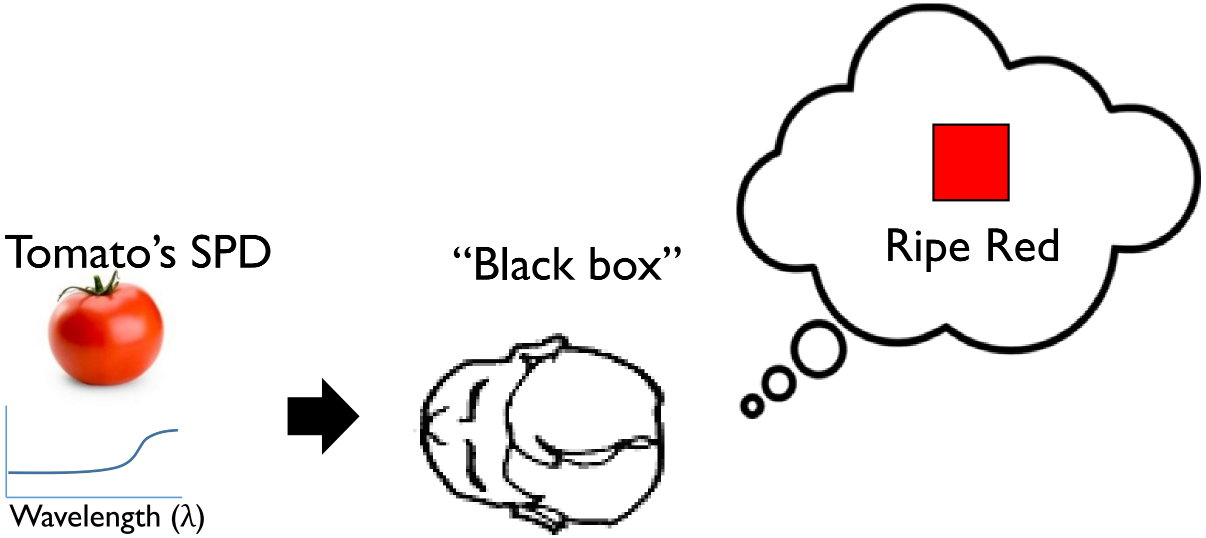
Radiometry vs. photometry

- **Radiometry**
  - Quantitative measurements of radiant energy
  - Often shown as spectral power distributions (SPD)
  - Measures either light coming from a source (radiance) or light falling on a surface (irradiance)
- **Photometry/ colorimetry**
  - Quantitative measurement of **perceived** radiant energy based on human's sensitivity to light
  - Perceived in terms of "brightness" (photometry) and color (colorimetry)

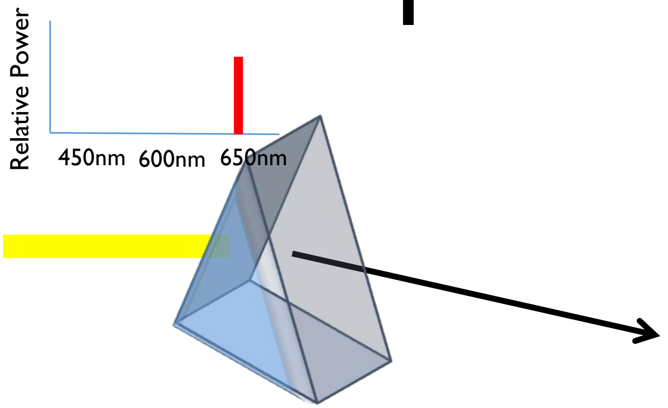


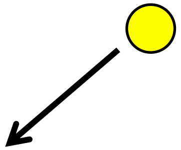
Quantifying color

- We still need a way to quantify color & brightness
- SPDs go through a "black box" (human visual system) and are perceived as color
- The only way to quantify the "black box" is to perform a human study

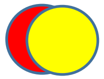


Experiments for photometry





Reference bright light
with fixed radiant power.



(Alternating between source and reference @ 17Hz)

Chromatic **source** light at a particular wavelength and adjustable radiant power.

Alternate between the source light and reference light 17 times per second (17 hz). A flicker will be noticeable unless the two lights have the same perceived "brightness".

+

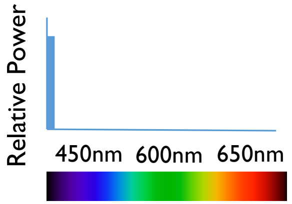
Viewer gradually increases source radiant power

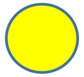
The viewer adjusts the radiant power of the chromatic light until the flicker disappears (i.e. the lights fuse into a constant color). The amount of radiant power needed for this fusion to happen is recorded.

Repeat this flicker fusion test for each wave-length in the source light. This allows method can be used to determine the perceived "brightness" of each wavelength.

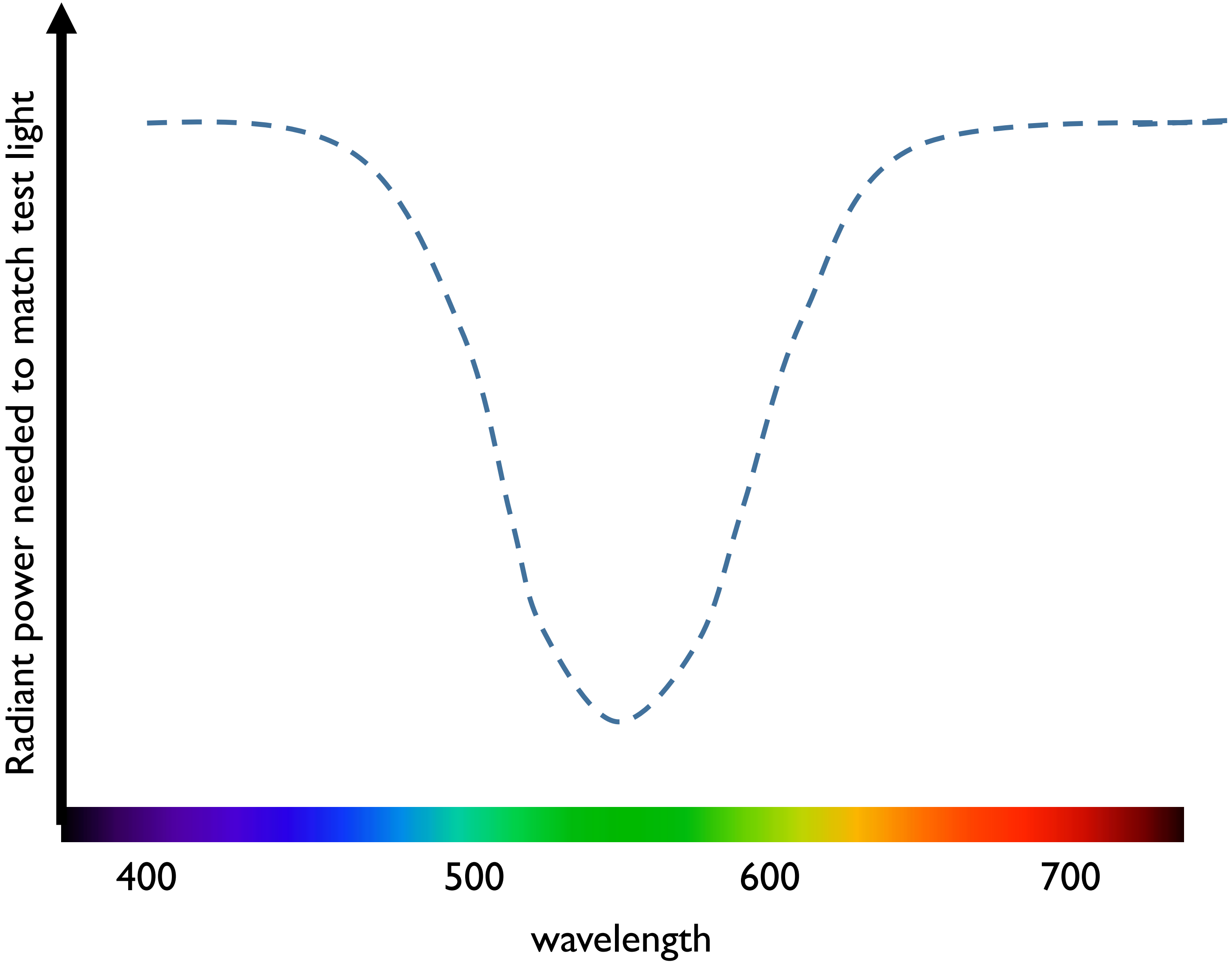
The "flicker photometry" experiment for photopic sensitivity.

Result of the flicker experiments





Reference light



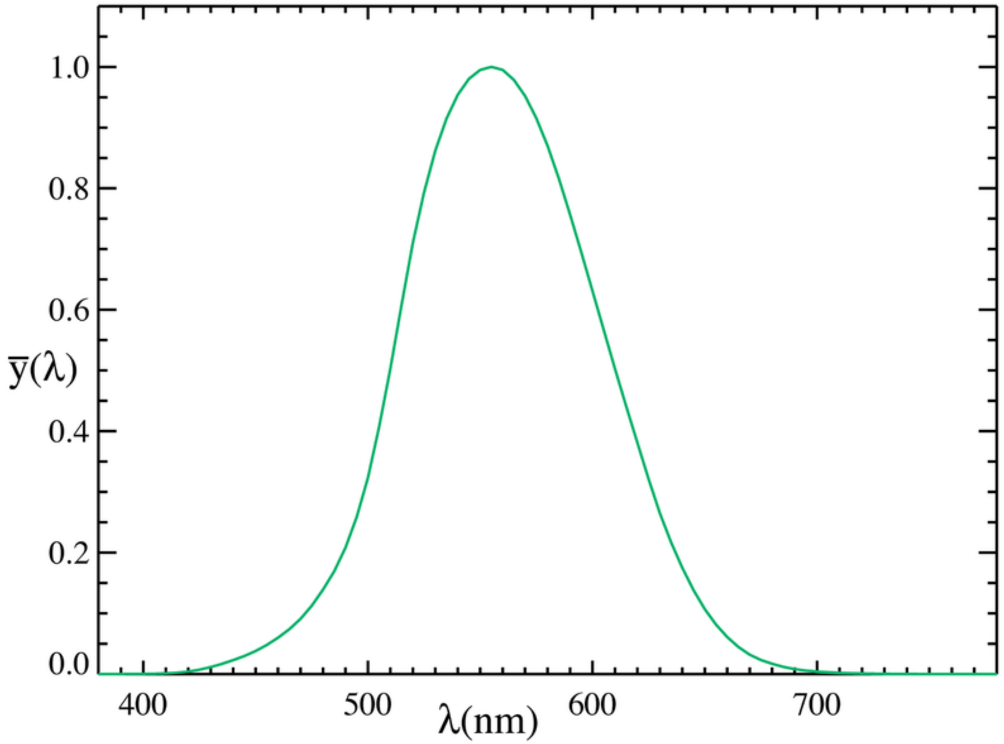
Amount of radiant power need for each wavelength
to make the reference light.

You need a lot more 400nm light to match the reference than you do the 550nm.

This means you perceive 550nm brighter than 400nm.

CIE (1924) Photopic luminosity function

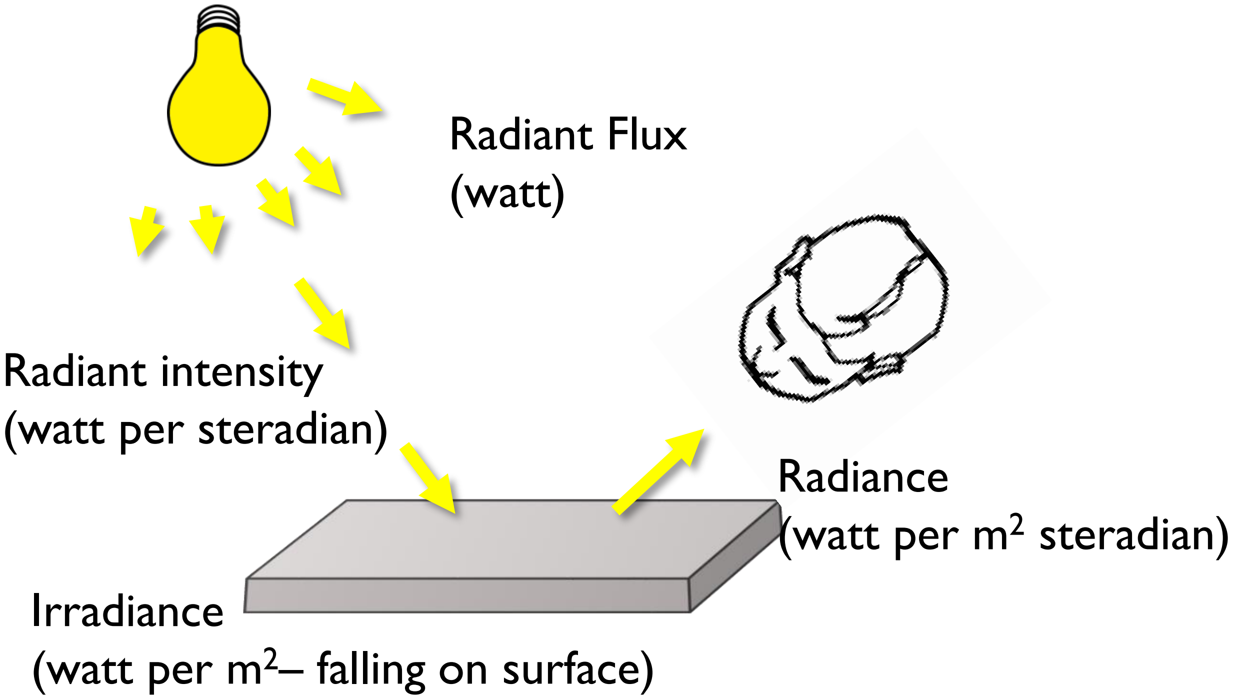
If we invert the curve on the pervious slide, we get the luminosity function.



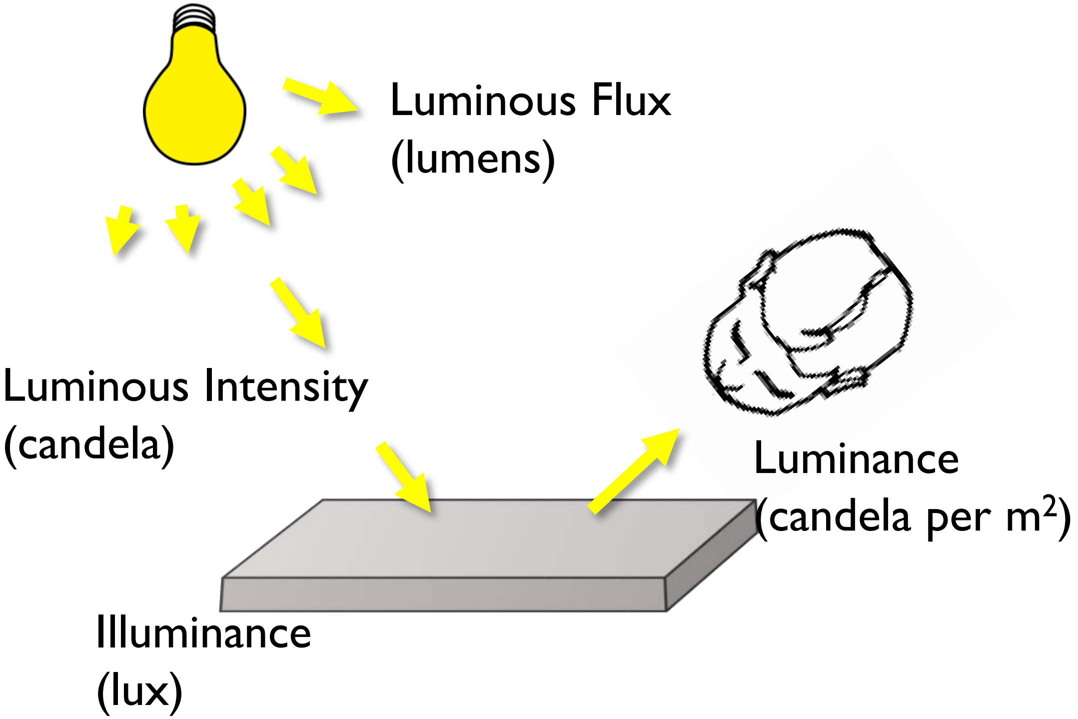
The Luminosity Function (written as $\bar{y}(\lambda)$ or $V(\lambda)$) shows the eye's sensitivity to radiant energy into luminous energy (or perceived radiant energy) based on human experiments (flicker fusion test).

International Commission on Illumination (CIE comes from the French name *Commission internationale e l'éclairage*) was a body established in 1913 as an authority on light, illumination and color . . CIE is still active today -- http://www.cie.co.at

Radiometric vs. photometric units



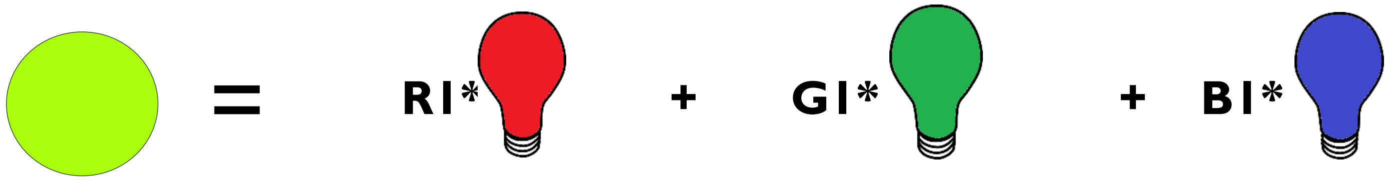
Radiometric values



Photometric values
(Radiometric values weighted by the Luminosity Function)

Colorimetry

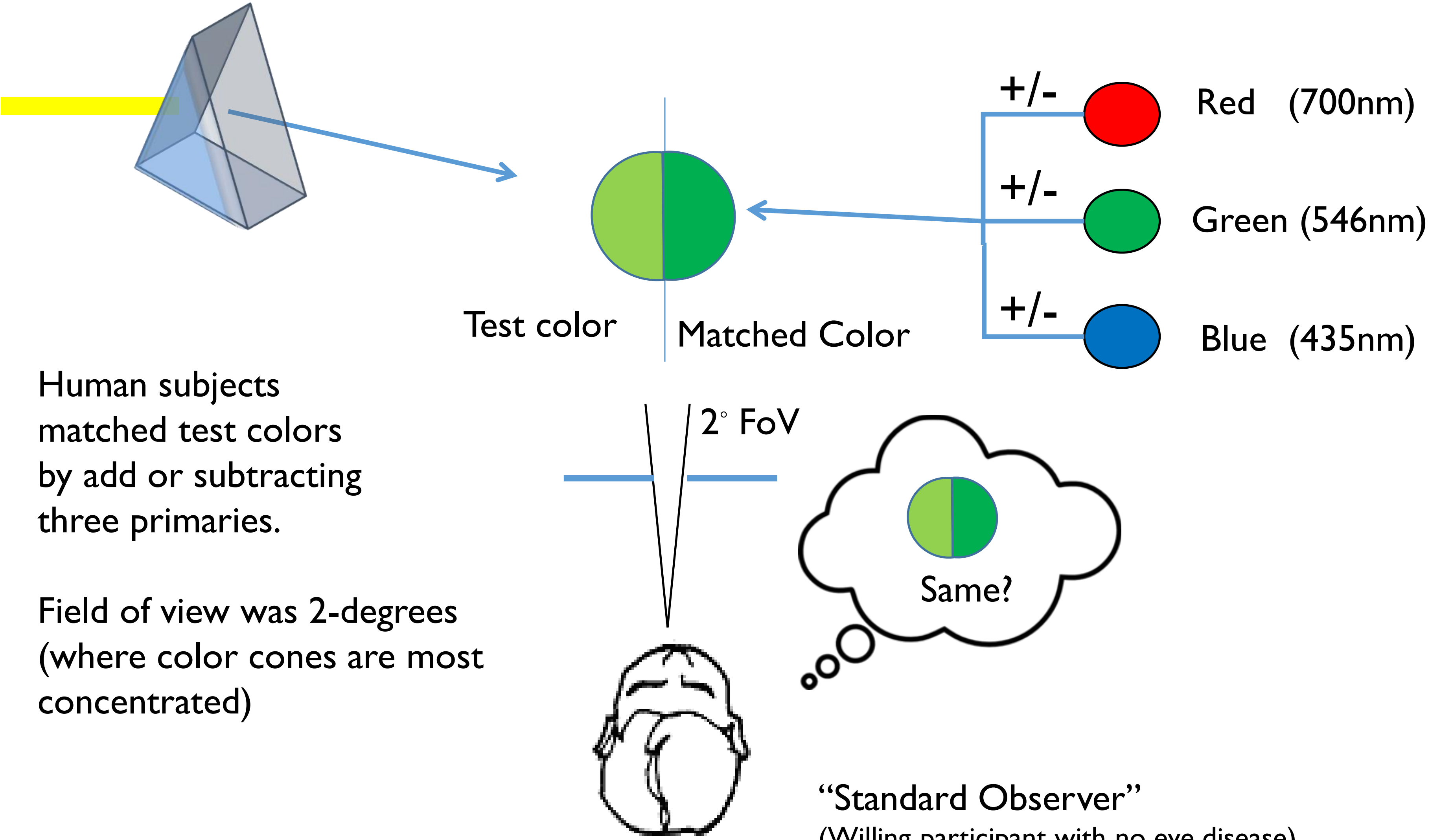
- Based on tristimulus color theory, colorimetry attempts to quantify all visible colors in terms of a standard set of primaries



Target color

Three fixed primary lights

CIE RGB color matching



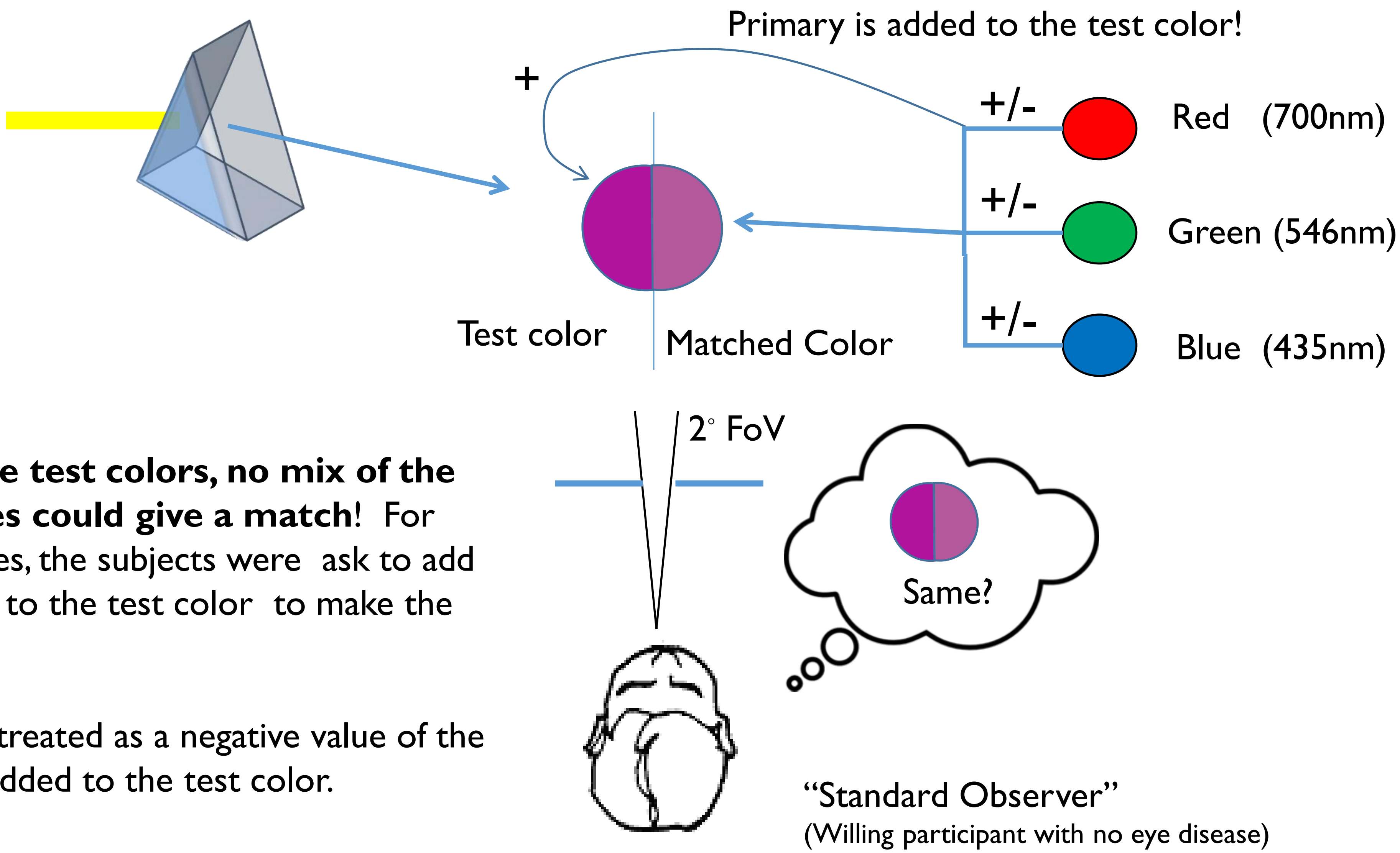
Human subjects
matched test colors
by add or subtracting
three primaries.

Field of view was 2-degrees
(where color cones are most
concentrated)

"Standard Observer"
(Willing participant with no eye disease)

Experiments carried out by
W. David Wright (Imperial College) and John Guild (National Physical Laboratory, London) – Late 1920s

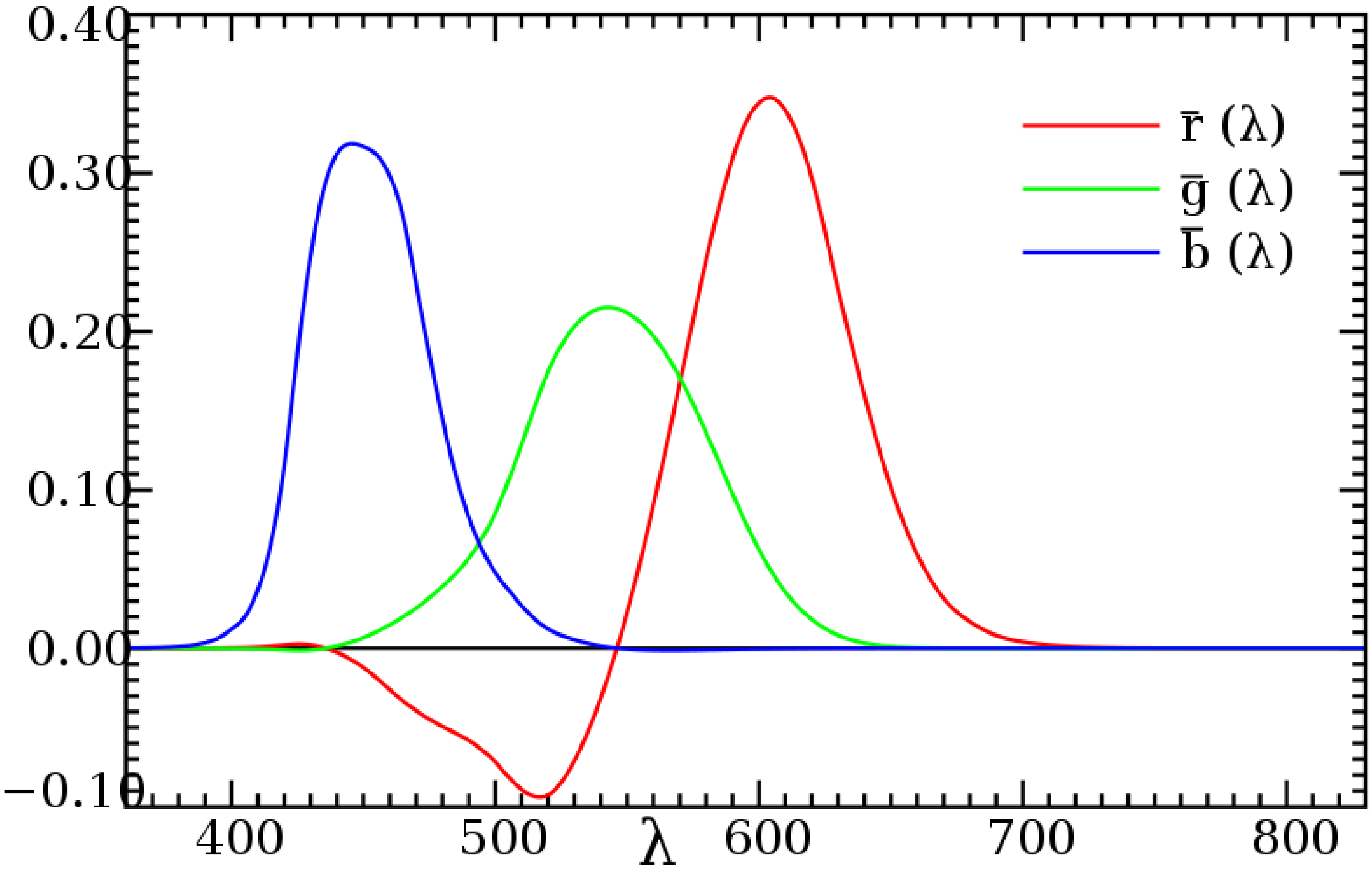
CIE RGB color matching



For some test colors, no mix of the primaries could give a match! For these cases, the subjects were ask to add primaries to the test color to make the match.

This was treated as a negative value of the primary added to the test color.

CIE RGB results

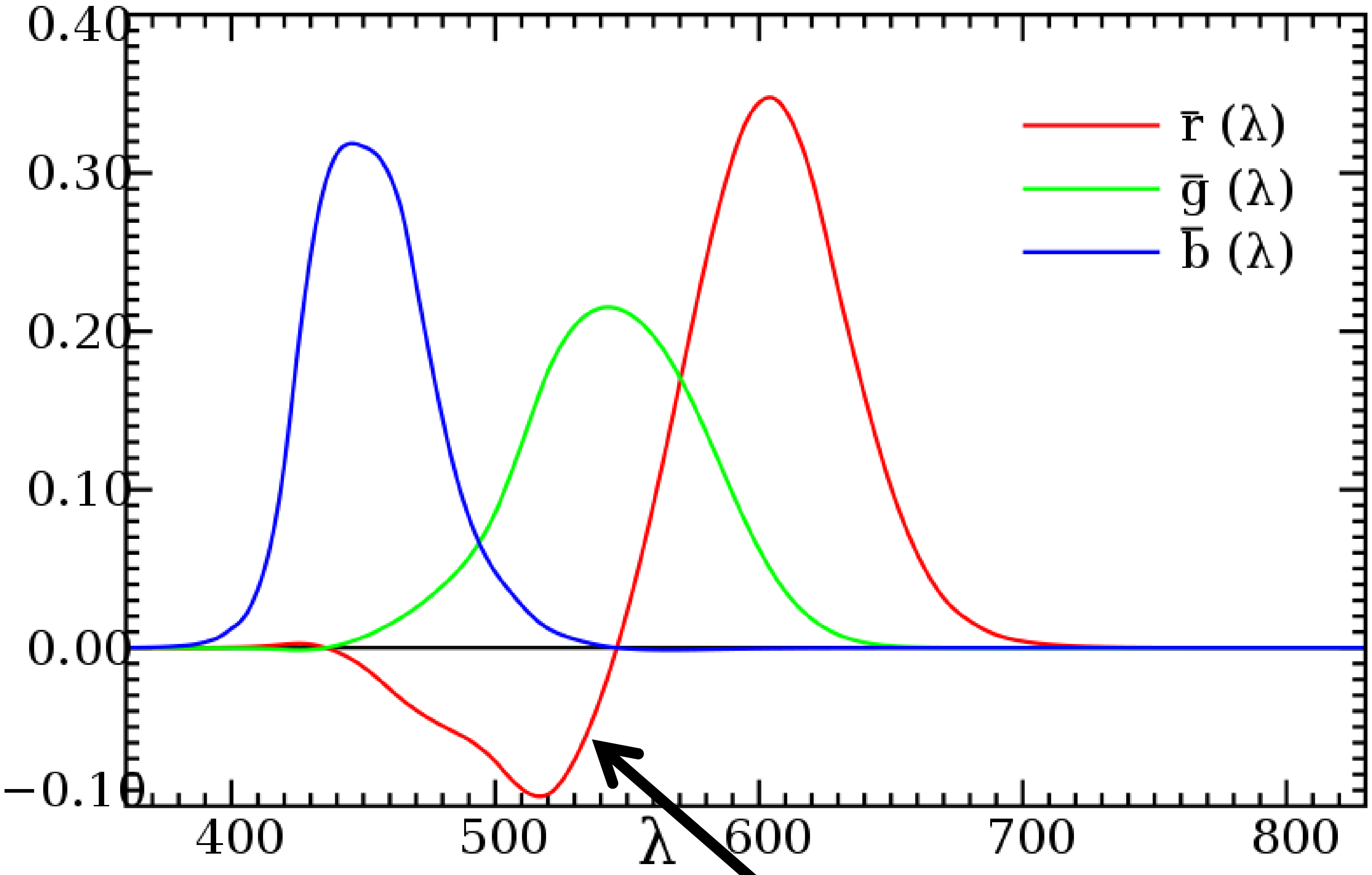


CIE RGB 2-degree Standard Observer
(based on Wright/Guild's data)

Plots are of the mixing coefficients of each primary needed to produce the corresponding monochromatic light at that wavelength.

Note that these functions have been scaled such that area of each curve is equal.

CIE RGB results



Negative values -- the three primaries used did not span the full range of perceptual colors.

CIE 1931 XYZ

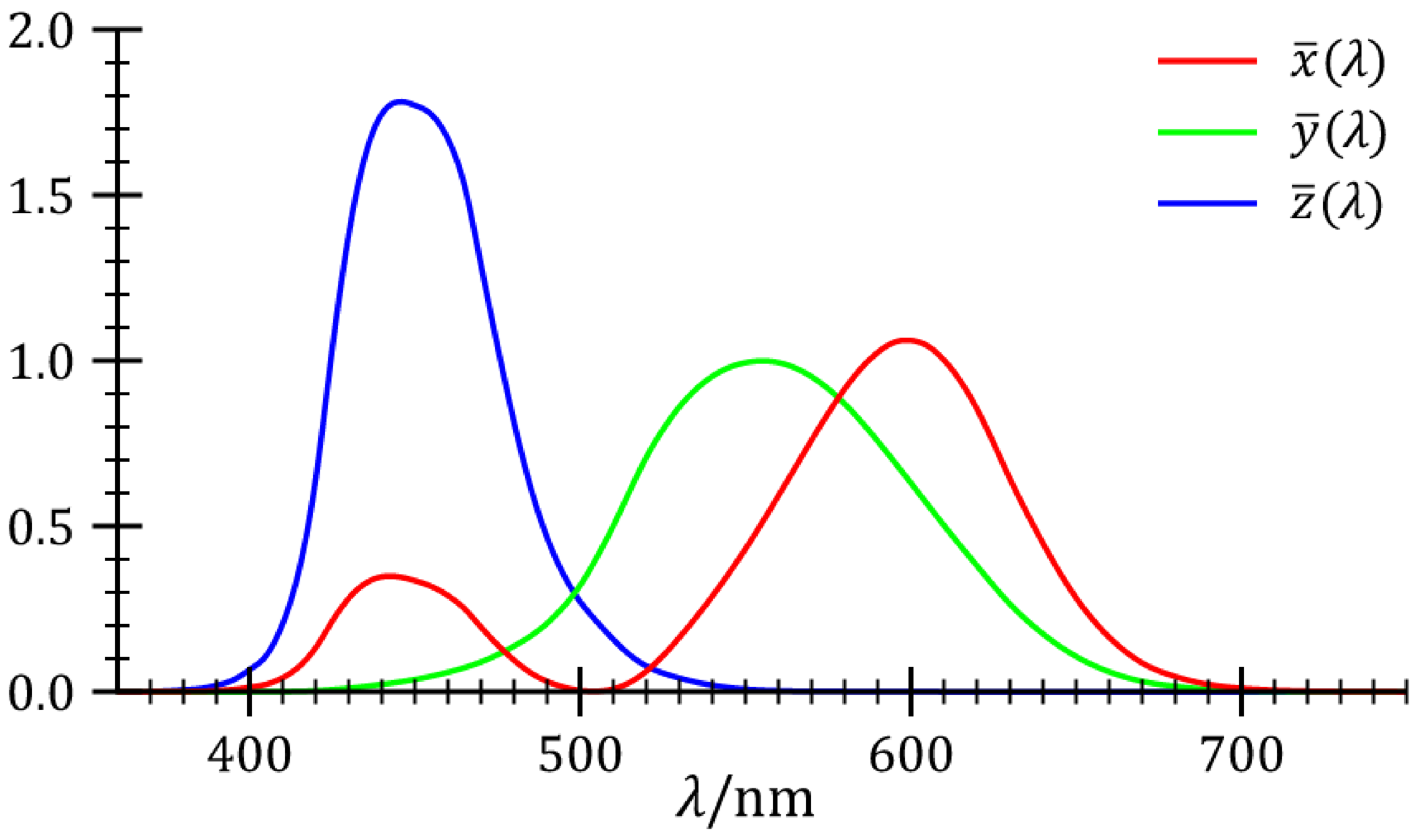
- In 1931, the CIE met and approved defining a new canonical basis, termed XYZ that would be derived from Wright-Guild's CIE RGB data
- Properties desired in this conversion:
  - White point defined at X=1/3, Y=1/3, Z=1/3
  - Y would be the luminosity function ($V(\lambda)$)
  - Quite a bit of freedom in selecting these XYZ basis
  - In the end, the adopted transform was:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4887180 & 0.3106803 & 0.2006017 \\ 0.1762044 & 0.8129847 & 0.0108109 \\ 0.0000000 & 0.0102048 & 0.9897952 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

CIE 1931 RGB

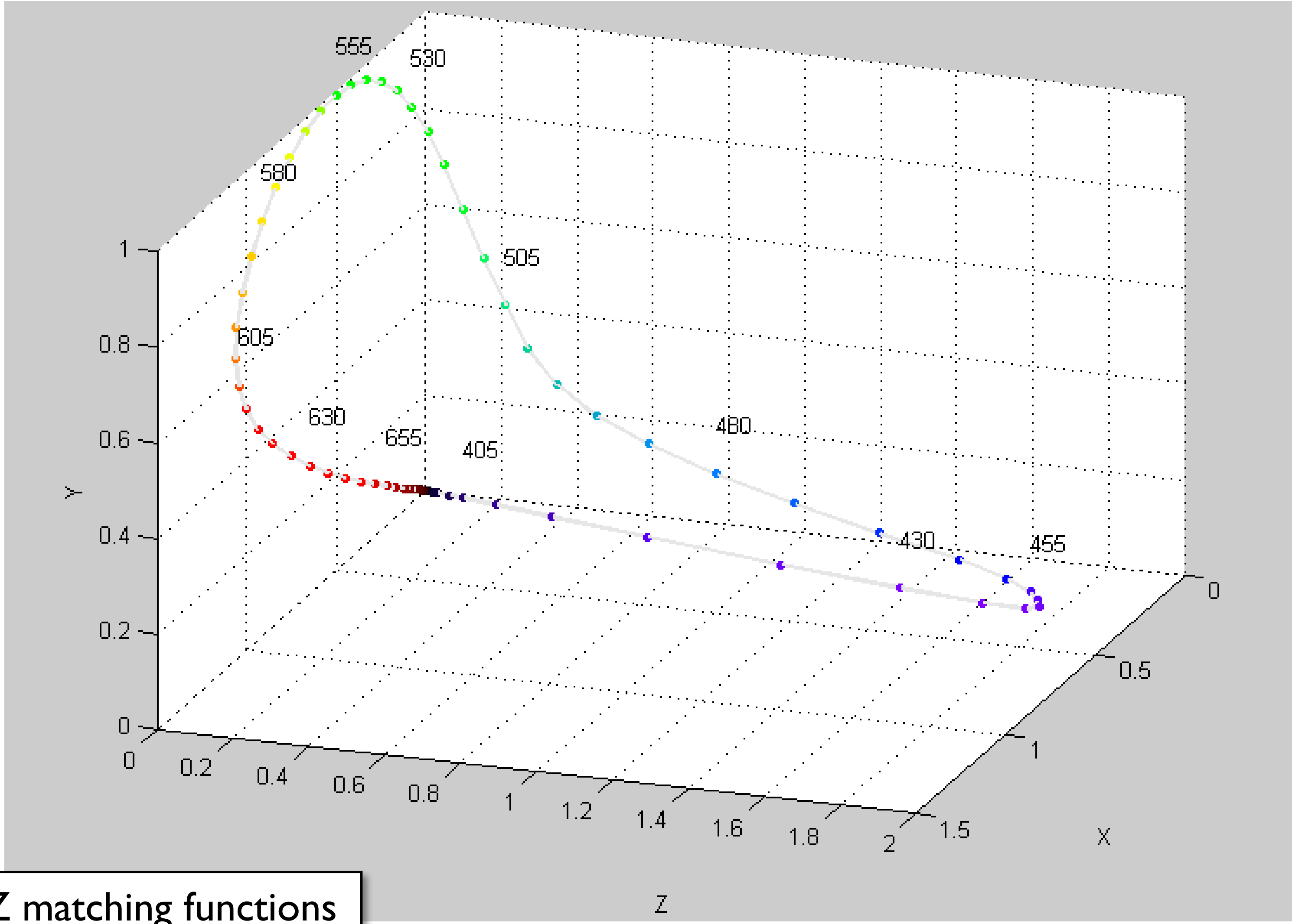
Nice article see: Fairman et al "How the CIE 1931 Color-Matching Functions Were Derived from Wright–Guild Data", Color Research & Application, 1997

CIE 1931 XYZ



This shows the mixing coefficients $\overline{x}(\lambda)$, $\overline{y}(\lambda)$, $\overline{z}(\lambda)$ for the CIE 1931 2-degree standard observer XYZ basis computed from the CIE RGB data. Coefficients are all now positive. Note that the basis XYZ are not physical SPD like in CIE RGB, but linear combinations defined by the matrix on the previous slide.

CIE XYZ 3D plot



3D plot of the CIE XYZ matching functions against the XYZ axis. Note that scaling of this plot is not uniform.

Using CIE 1931 XYZ functions

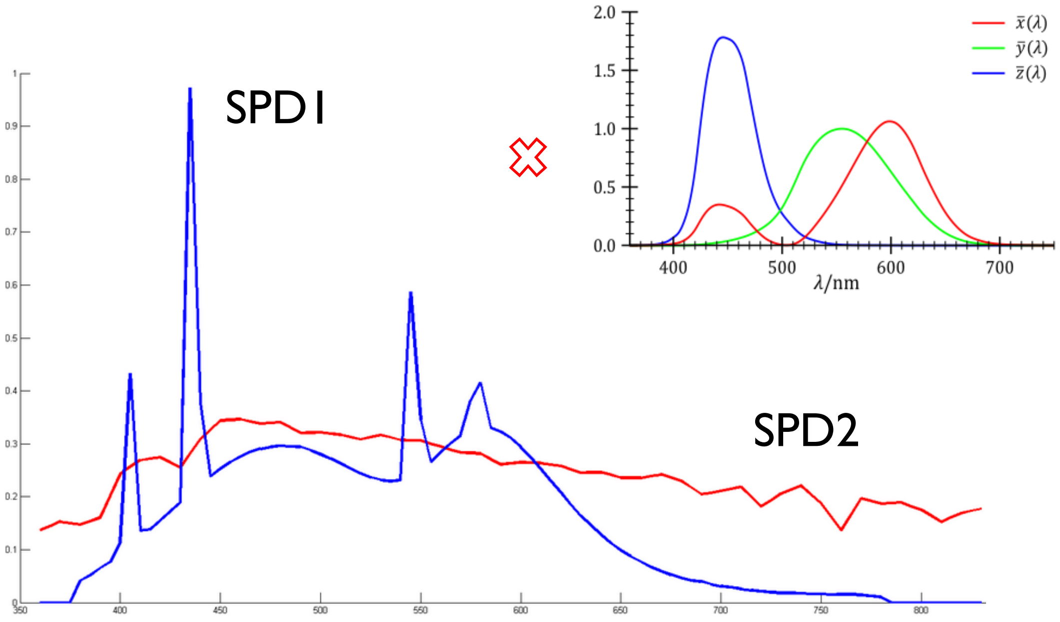
- **We now have a canonical color space to describe SPDs**
- Given an SPD, I(λ), we can compute its mapping to the CIE XYZ space

$$X = \int_{380}^{780} I(\lambda)\bar{x}(\lambda)\, d\lambda \qquad Y = \int_{380}^{780} I(\lambda)\bar{y}\,(\lambda)\, d\lambda \qquad Z = \int_{380}^{780} I(\lambda)\bar{z}(\lambda)\, d\lambda$$

- Given two SPDs, if their CIE XYZ values are equal, then they are considered the same perceived color, i.e.
  - $I_1(\lambda), I_2(\lambda) \rightarrow (X_1, Y_1, Z_1) = (X_2, Y_2, Z_2)$ [perceived as the same color]

- So . . we can quantitatively describe color!

SPD to CIE XYZ example



Two SPDs

CIE XYZ Values

SPD1
X=0.2841
Y=0.2989
Z=0.3254

SPD2
X=0.2841
Y=0.2989
Z=0.3254

From their CIE XYZ mappings, we can determine that these two SPDs will be *perceived* as the same color (even without needing to see the color!)

Thanks CIE XYZ!

Radiometric **Colorimetric**

CIE XYZ gives a way to go from radiometric to colorimetric.
Imbedded is also the photometric measurement in the Y value.

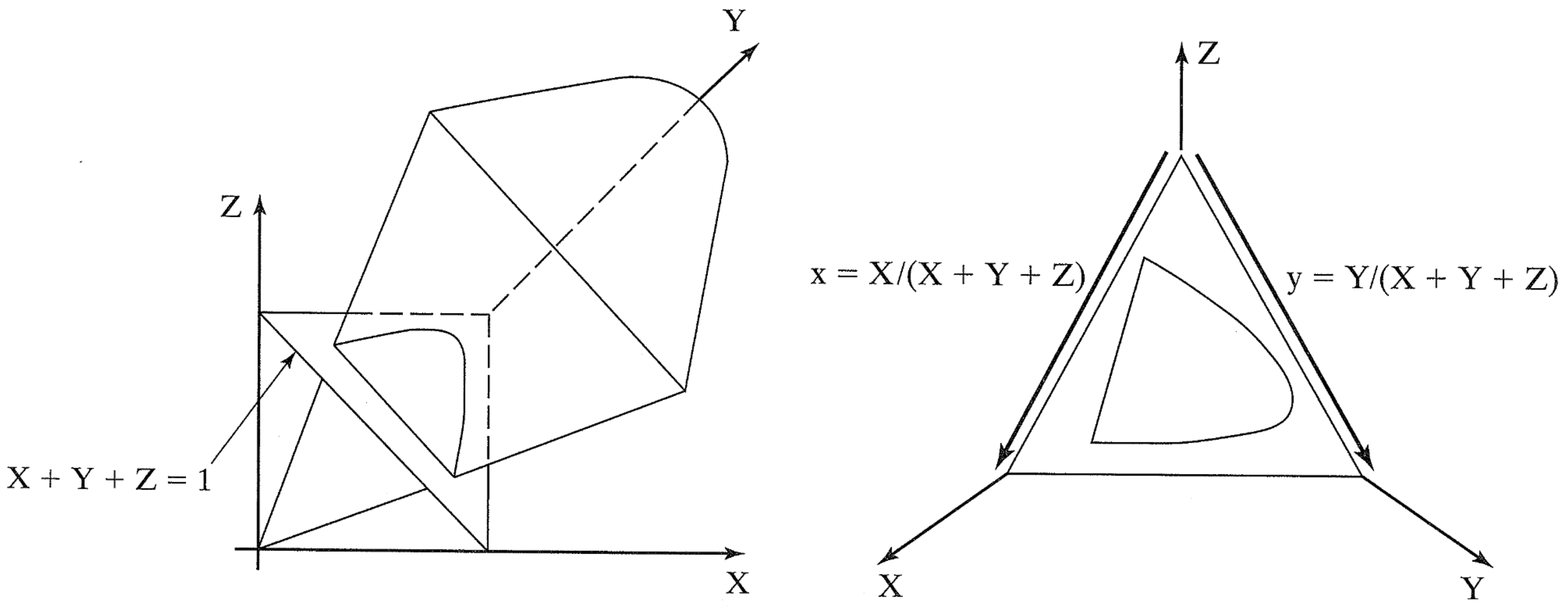
Usefulness of CIE 1931 XYZ

- CIE XYZ space is also considered "device independent" – the XYZ values are not specific to any device
- Electronic devices (e.g. cameras, flatbed, scanners, printers, displays) can compute mappings of their device specific values to the corresponding CIE XYZ values.
- This provides a canonical space to match between devices (at least in theory).

Luminance-chromaticity space (CIE xyY)

- CIE XYZ describes a color in terms of linear combination of three primaries (XYZ)
- Sometimes it is useful to discuss color in terms of luminance (perceived brightness) and chromaticity (we can think of as the hue-saturation combined)
- CIE xyY space is used for this purpose

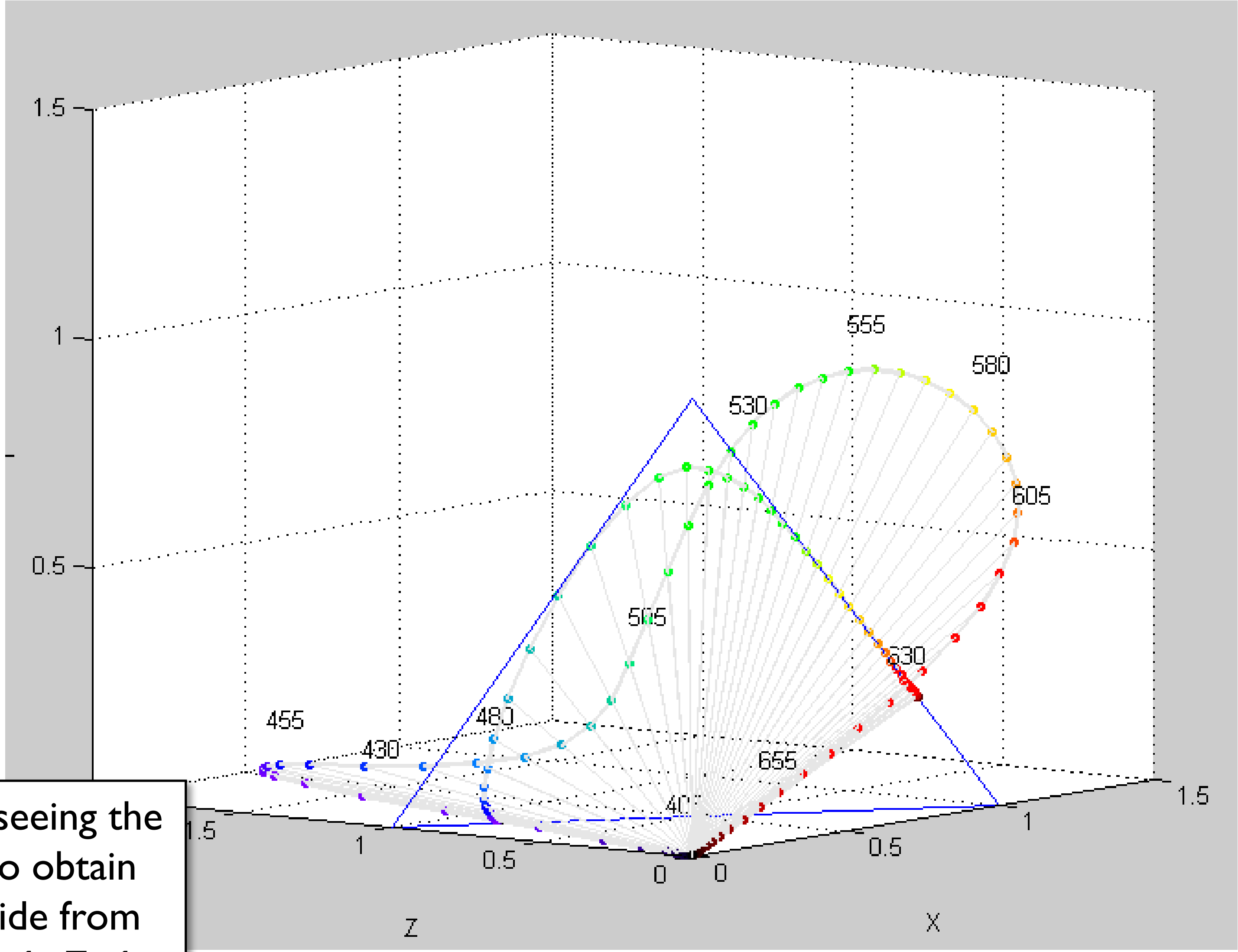
Deriving CIE xyY



Project the CIE XYZ values onto the X=1, Y=1, Z=1 plane.

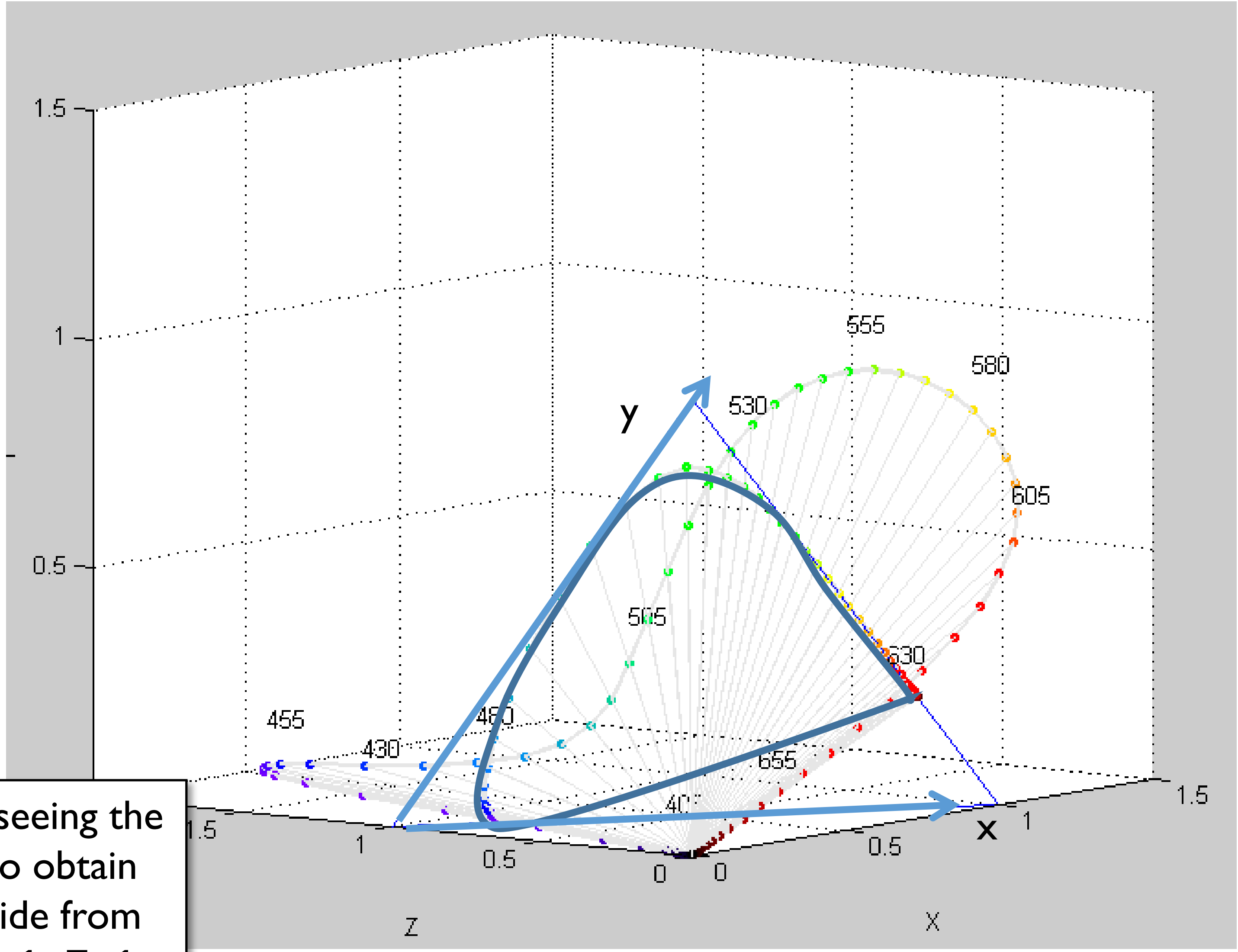
Figure from Ponce and Forsyth

Projection plot



A bit hard to visualize after seeing the CIE XYZ 3D plot, but we do obtain the shape on the previous slide from the projection onto X=1, Y=1, Z=1

Projection plot



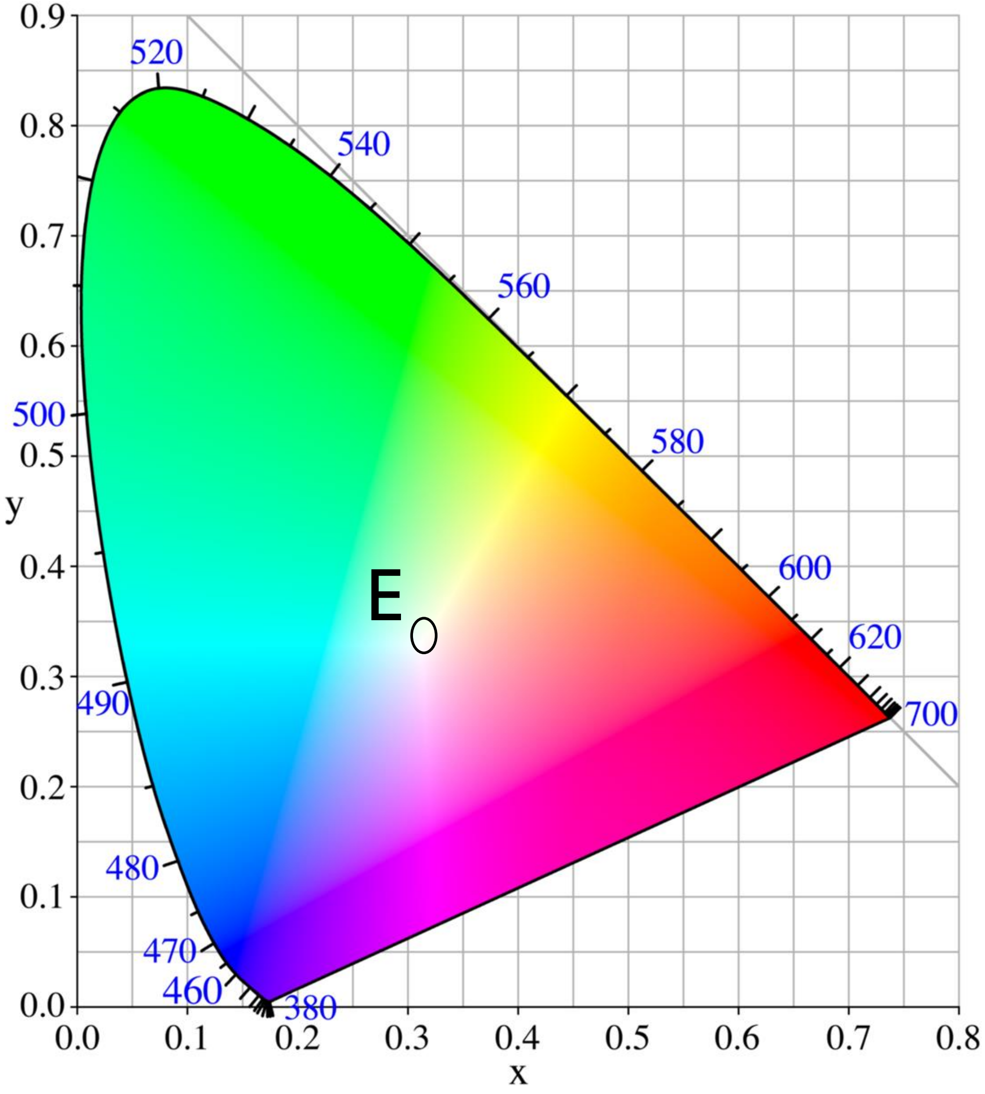
A bit hard to visualize after seeing the CIE XYZ 3D plot, but we do obtain the shape on the previous slide from the projection onto X=1, Y=1, Z=1

CIE xy chromaticity diagram

This gives us the familiar horseshoe shape of visible colors as a 2D plot. Note the axis are x & y.

Point "E" represents where X=Y=Z have equal energy (X=0.33, Y=0.33, Z=0.33)

CIE XYZ "white point"



In the 1930s, CIE had a bad habit of over using the variables X, Y. Note that x, y are chromaticity coordinates, $\bar{x}, \bar{y}$ (with the bar above) are the matching functions, and X, Y are the imaginary SPDs of CIE XYZ.

Fast forward 80+ years

- CIE 1931 XYZ, CIE 1931 xyY (2-degree standard observer) color spaces have stood the test of time
- Many other studies have followed (most notably - CIE 1965 XYZ 10-degree standard observer), . . .
- But in the literature (and in this tutorial) you'll find CIE 1931 XYZ color space remains the preferred standard

What is perhaps most amazing?

- 80+ years of CIE XYZ and it is all based on the experiments by the "standard observers"

- How many standard observers were used? 100, 500, 1000?

A Standard Observer

CIE XYZ is based on 17 standard observers

10 by Wright, 7 by Guild

“The Standard Observers”

A caution on CIE xy chromaticity

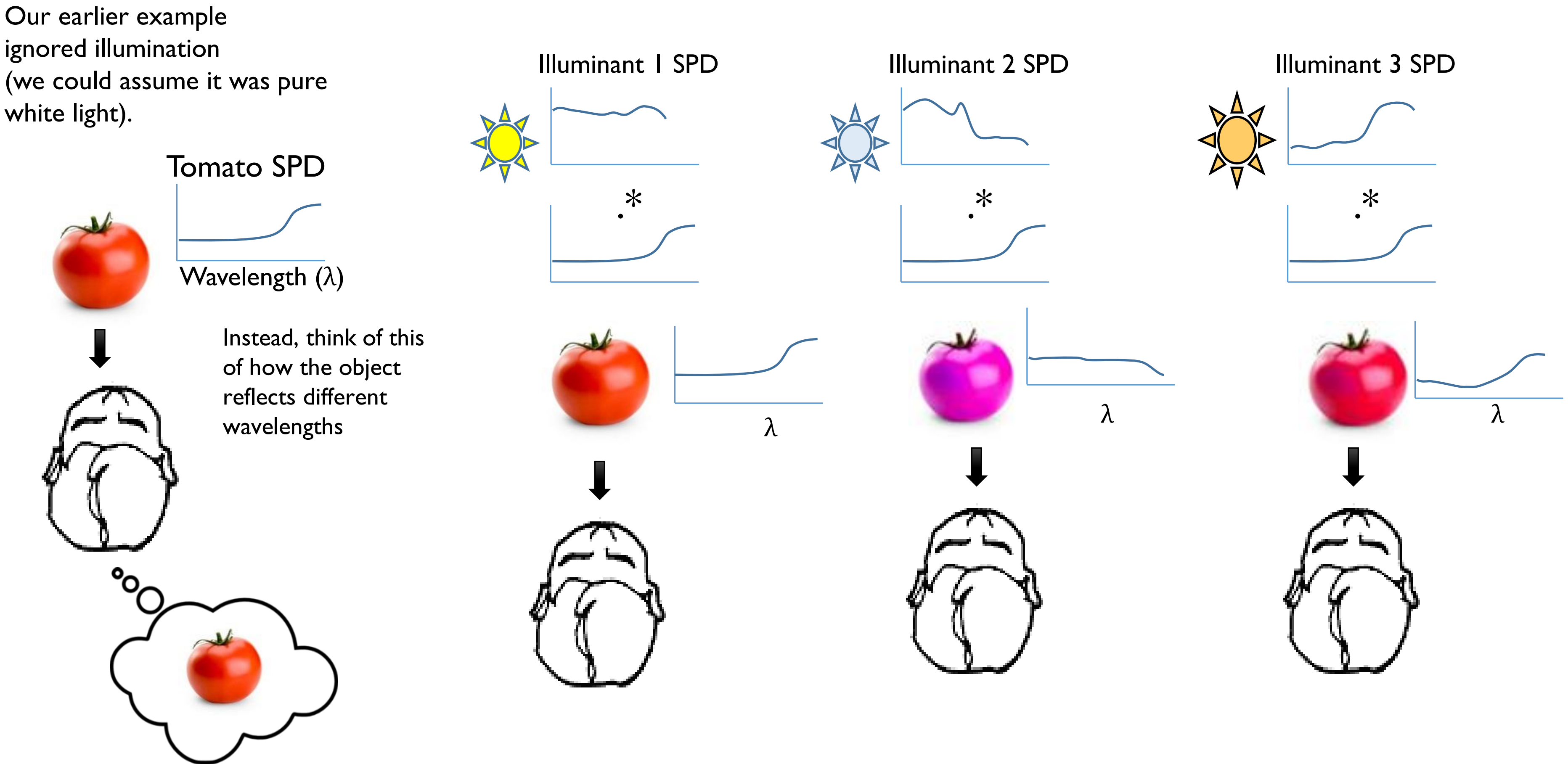
From Mark D. Fairchild book: "Color Appearance Models"

"*The use of chromaticity diagrams should be avoided in most circumstances, particularly when the phenomena being investigated are highly dependent on the three-dimensional nature of color. For example, the display and comparison of the color gamuts of imaging devices in chromaticity diagrams is misleading to the point of being almost completely erroneous.*"

Are we done with color?

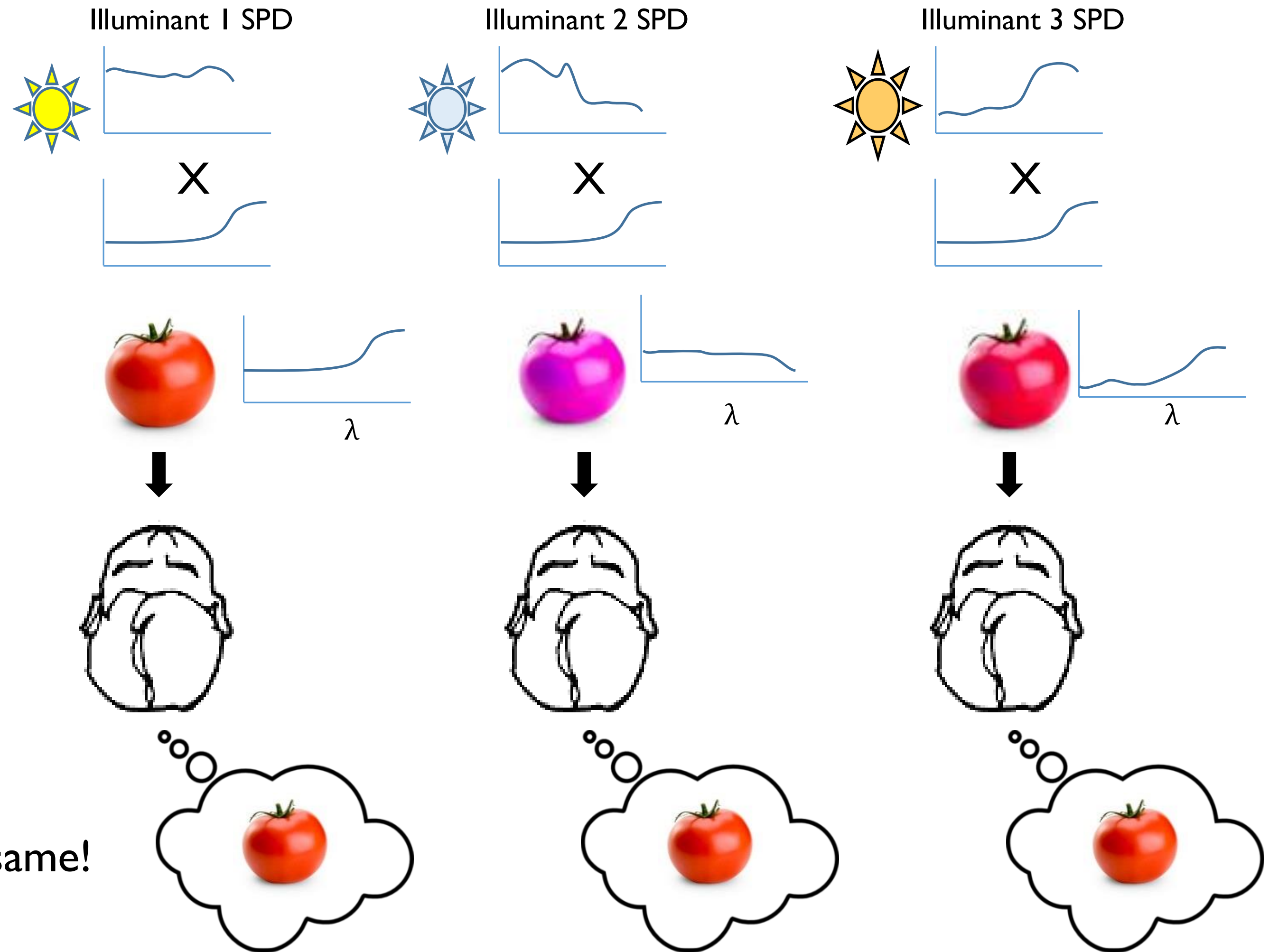
An object's SPD

- In a real scene, an object's SPD is a combination of the its reflectance properties **and** scene illumination



Color constancy

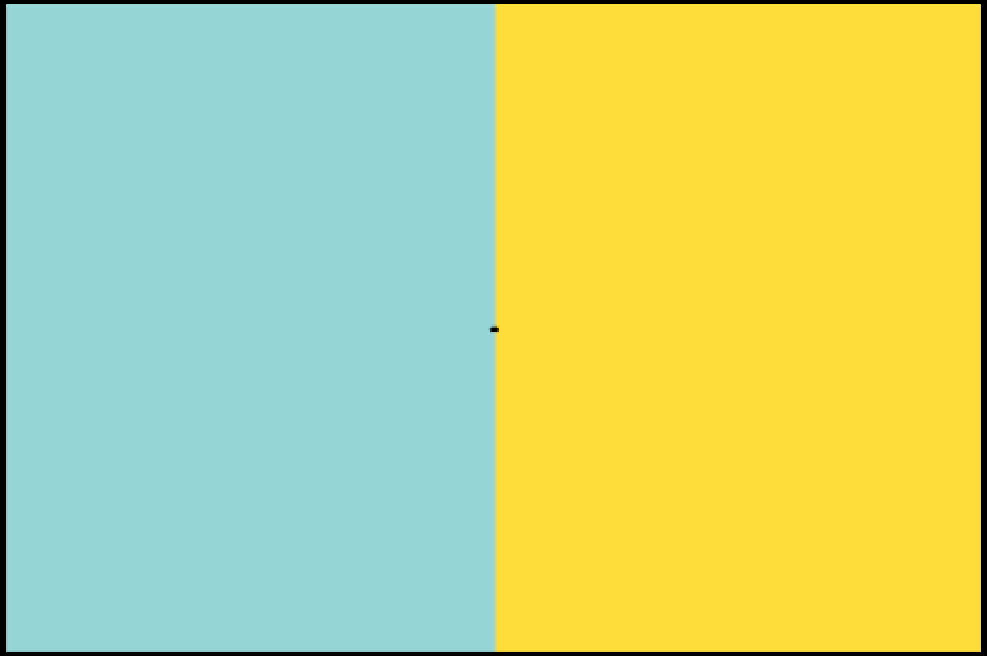
- Our visual system is able to compensate for the illumination



Looks the same!

Chromatic adaptation example

Chromatic adaptation example



Color constancy/chromatic adaptation

- Color constancy (also called *chromatic adaptation*) is the ability of the human visual system to adapt to scene illumination
- This ability is not perfect, but it works fairly well
- Image sensors do not have this ability (it must be performed as a processing step, i.e. "white balance")

Note: Our eyes do not adjust to the illumination in the photograph -- we adjust to the viewing conditions of the scene we are viewing the photograph!

Color constancy and illuminants

- To understand color constancy, we have to consider SPDs of different illuminants

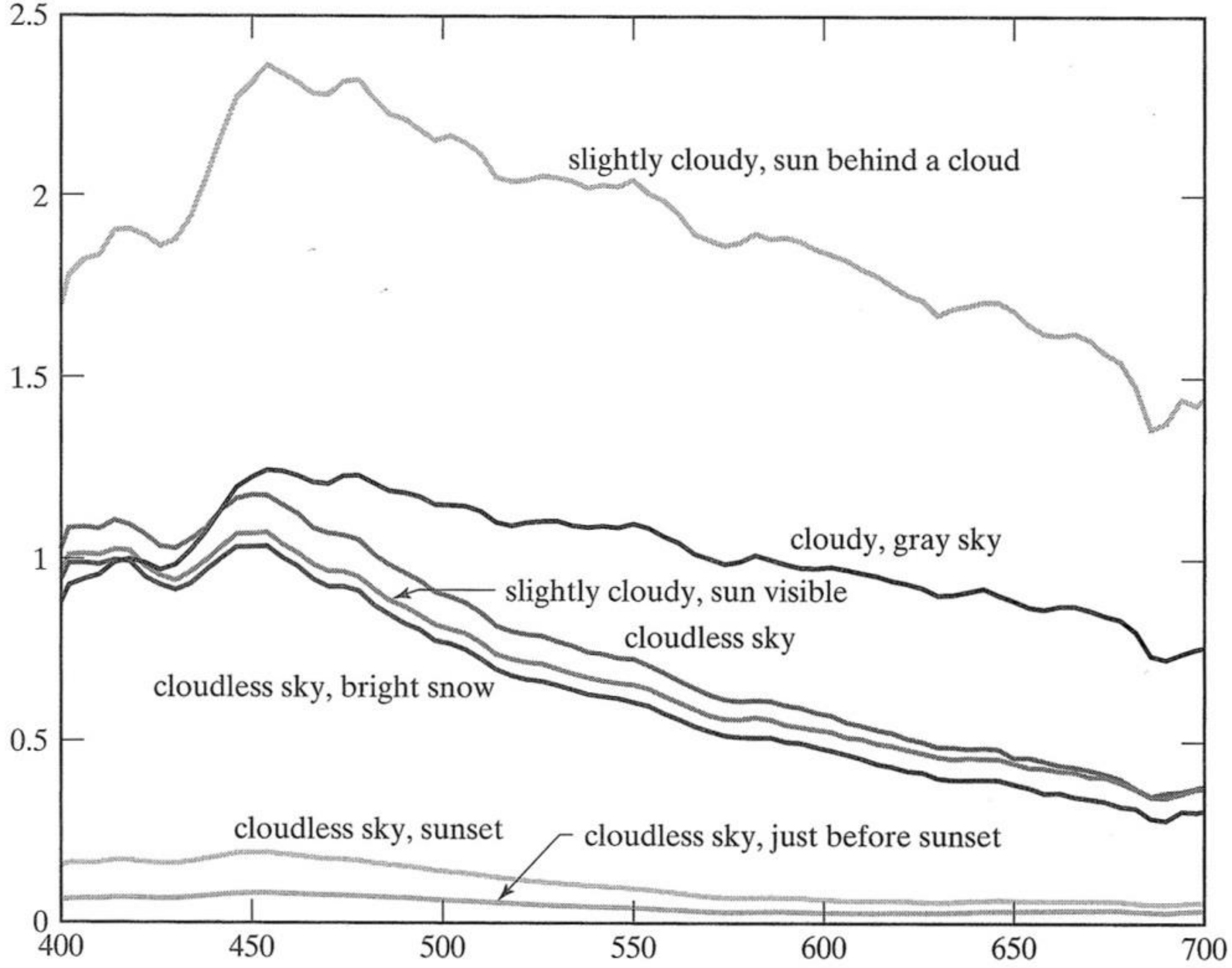
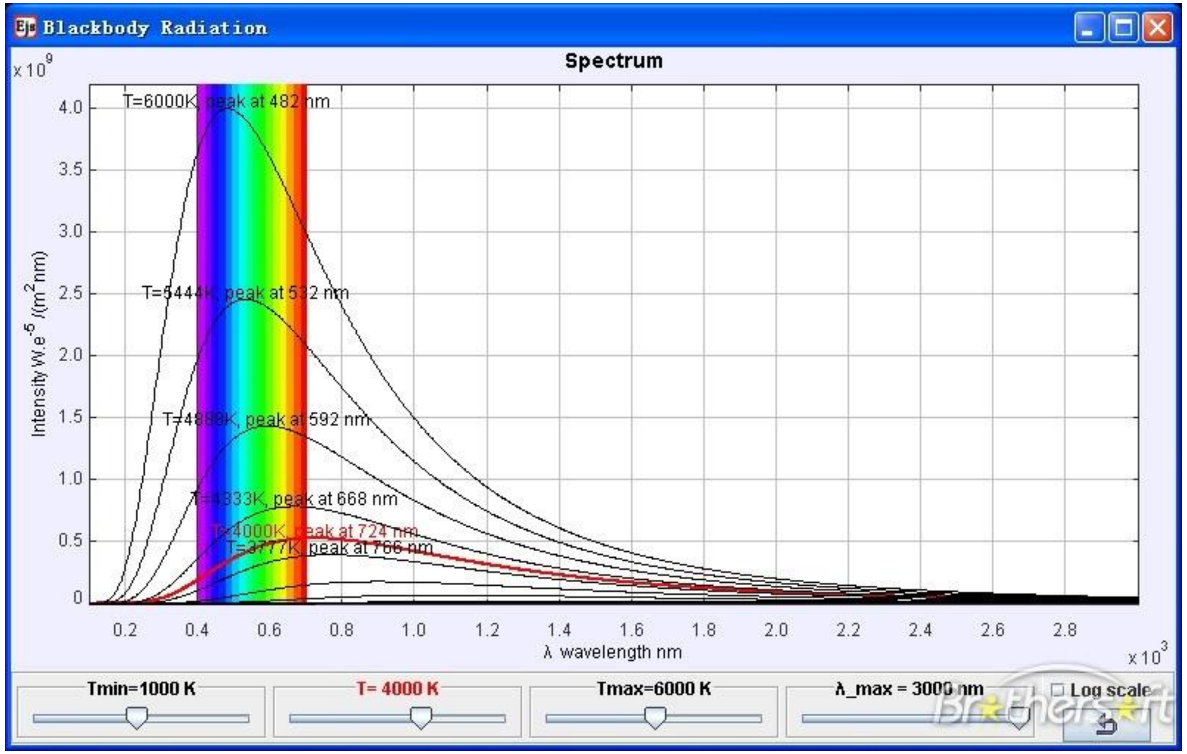


Figure from Ponce and Forsyth

Color temperature

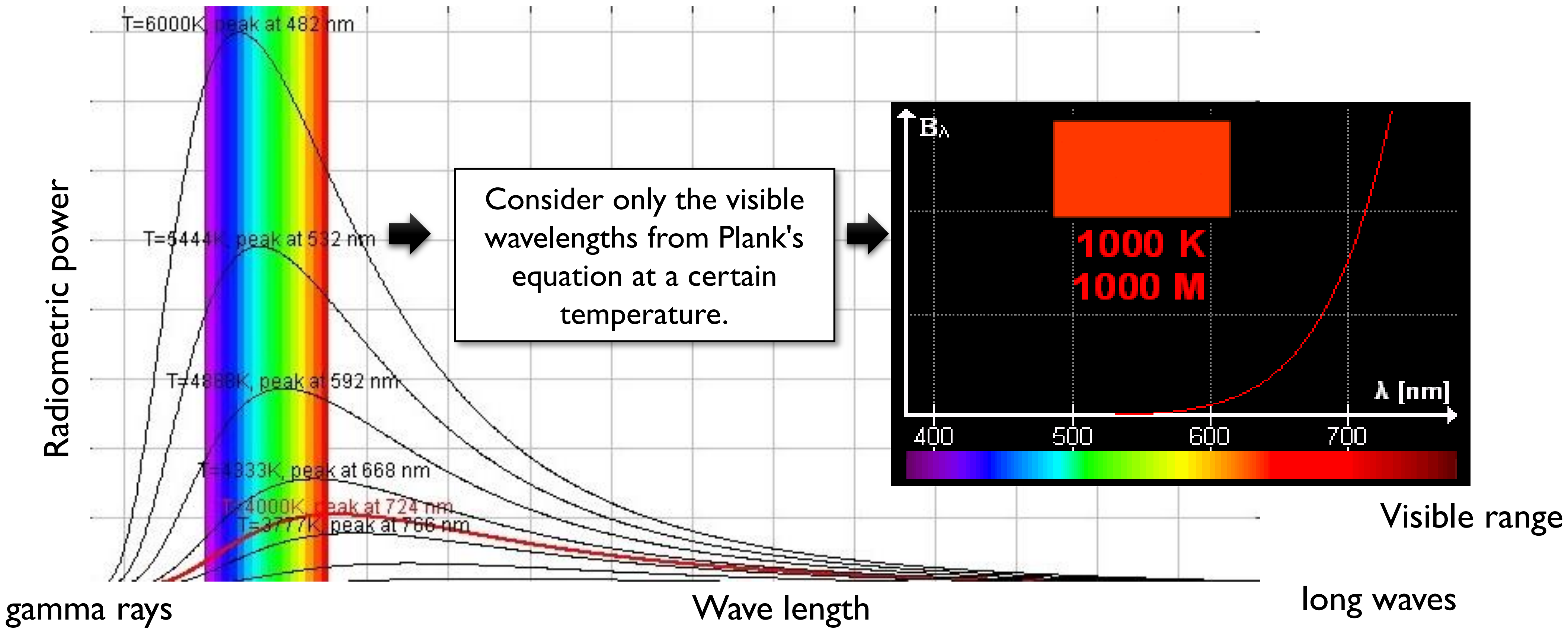
- Illuminants are often described by their "color temperature"
- This mapping is based on theoretical "blackbody radiators" that produce SPDs for a given temperature -- expressed in Kelvin (K)
- We map light sources (both real and synthetic) to their closest color temperature (esp in Photography/Video production)



$$B_\lambda(\lambda, T) = \frac{2hc^2}{\lambda^5} \frac{1}{e^{\frac{hc}{\lambda k_B T}} - 1}$$

Plank's law
Spectral density of electromagnetic radiation emitted by a blackbody radiator at a given temperature T.

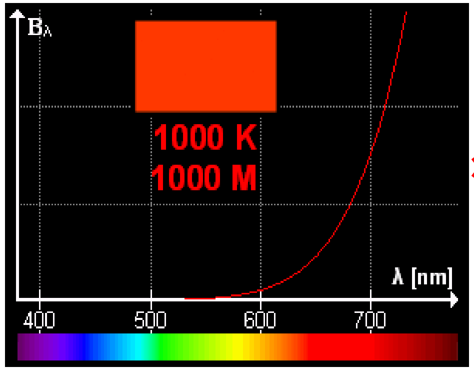
Visible range of a black body radiator

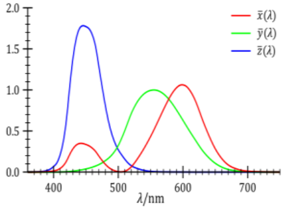


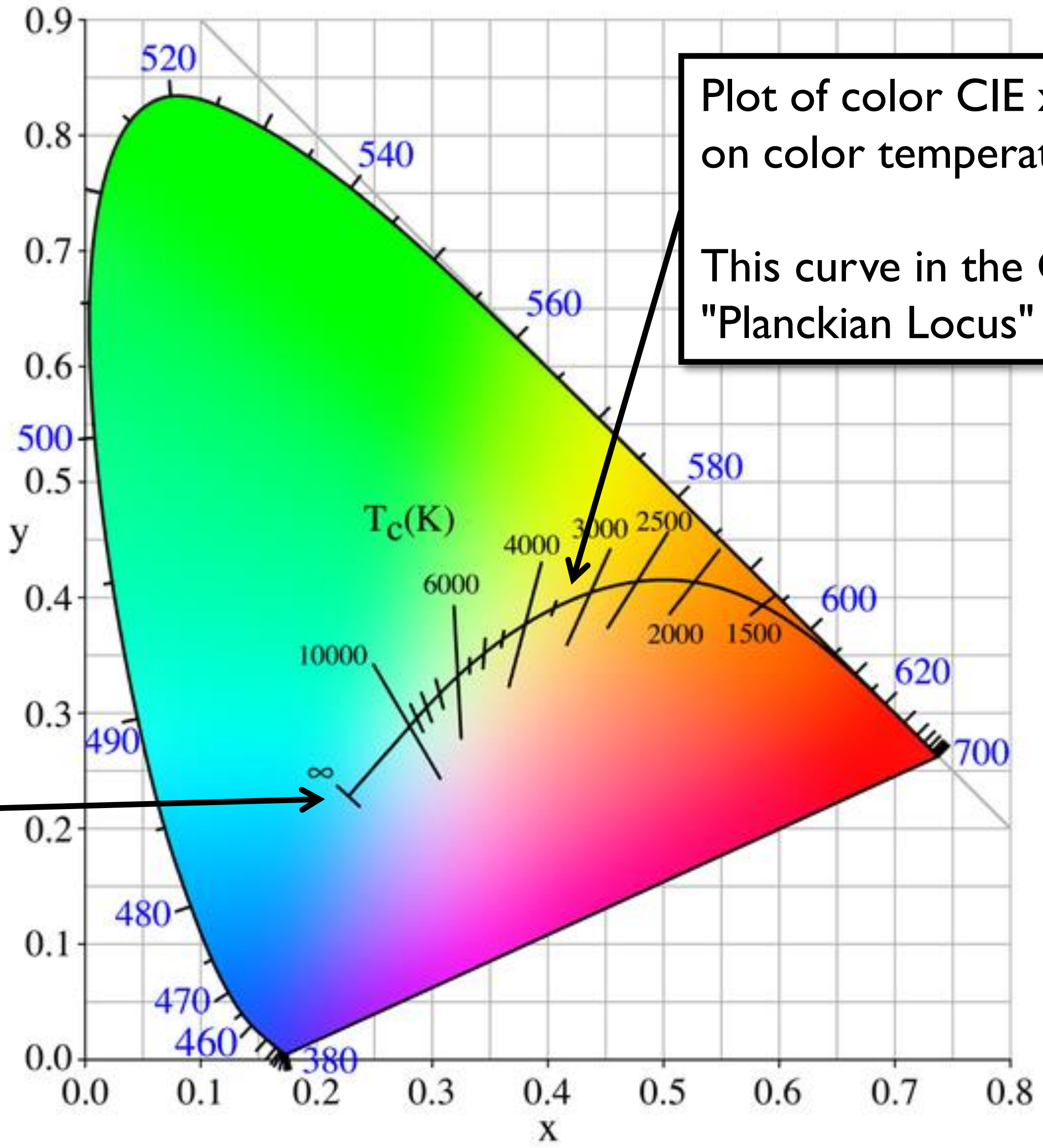
Black body radiator SPD for different color temperatures

Animation credit: Dariusz Kowalczyk

Plot visible SPDs in CIE xy chromaticity





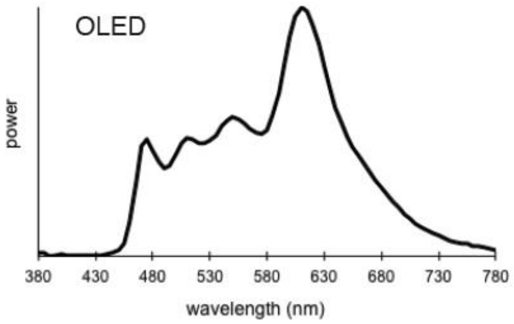


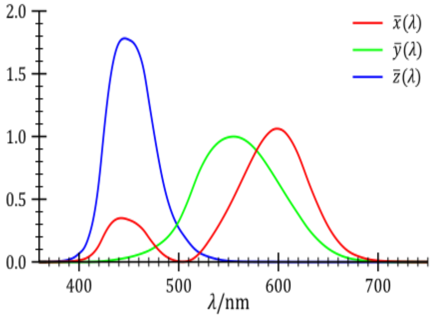
Plot of color CIE xy locations of SPDs based on color temperature.

This curve in the CIE xy plot of the "Planckian Locus" of color temperatures.

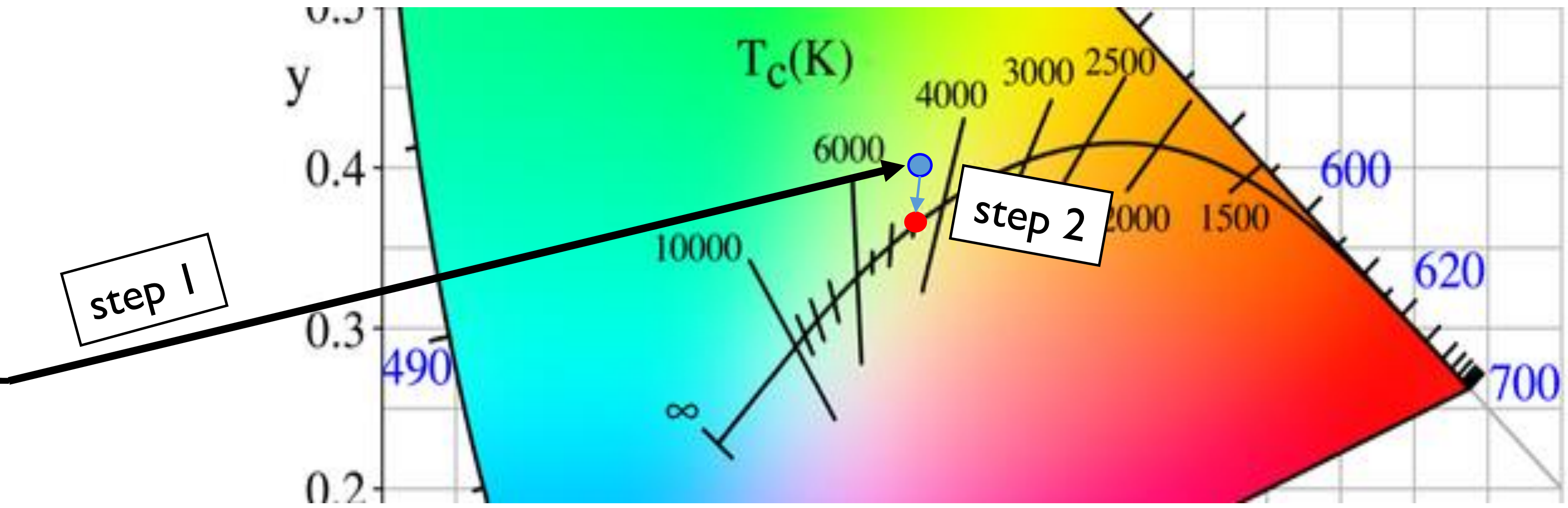
Color temperature of an SPD example

SPD of a light source





CIE 1931 mapping functions

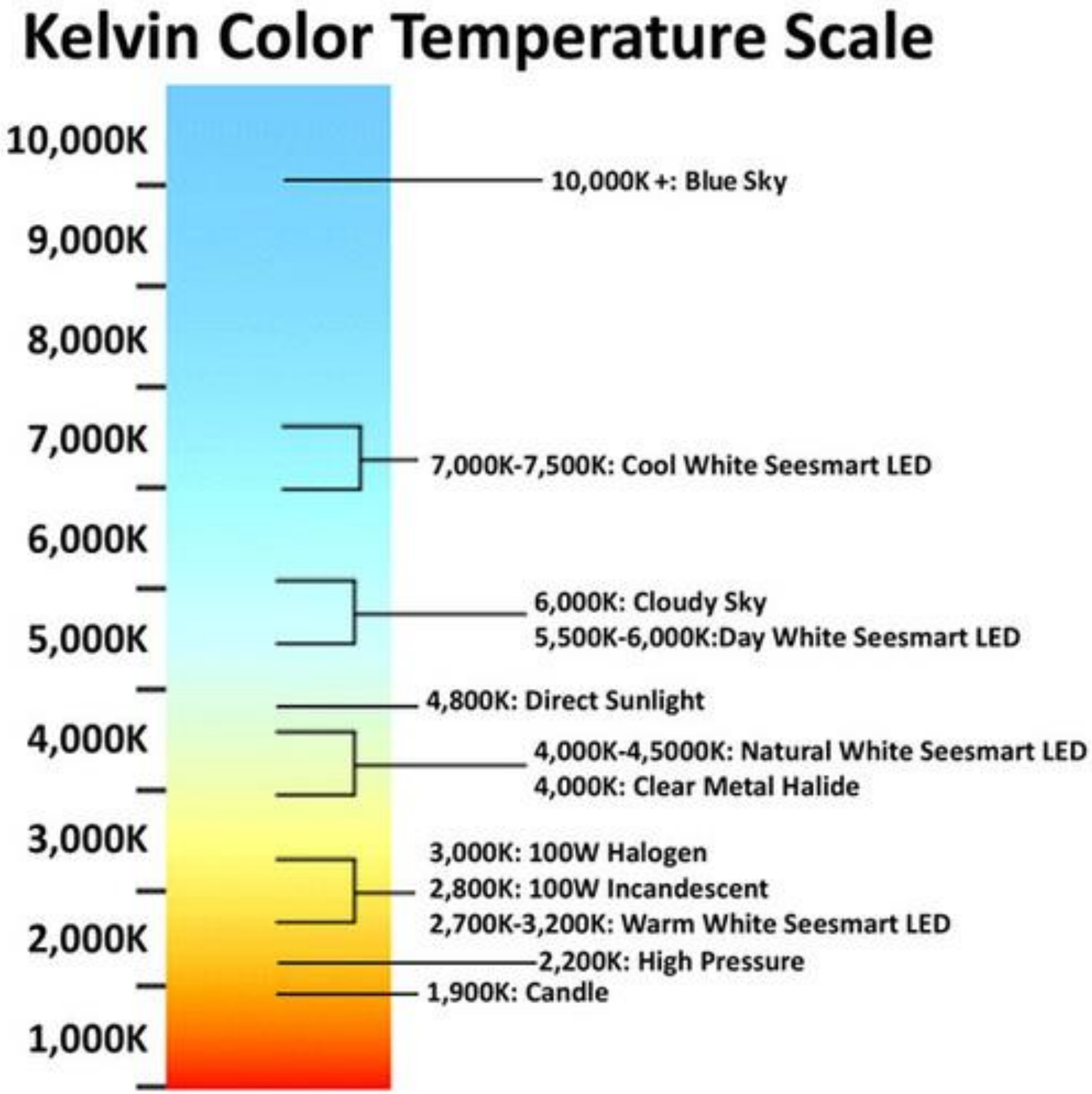


(1) Find the light sources SPD mapping to CIE XYZ using the CIE 1931 mapping functions.
(2) Project the CIE xyY value to the Planckian locus line.

Where it falls is the Correlated Color Temperature (CCT) of this light source. So, this example the OLED light source is roughly 4500K.

While we often say "color temperature", we should say "correlated color temperature". The concept is not related to the physical temperature of the light source, but its *correlation* with the black body radiator's color temperature.

Color temperature



Typical description of color temperature used in photography & lighting sources.

From B&H Photo

Man made illuminants SPDs

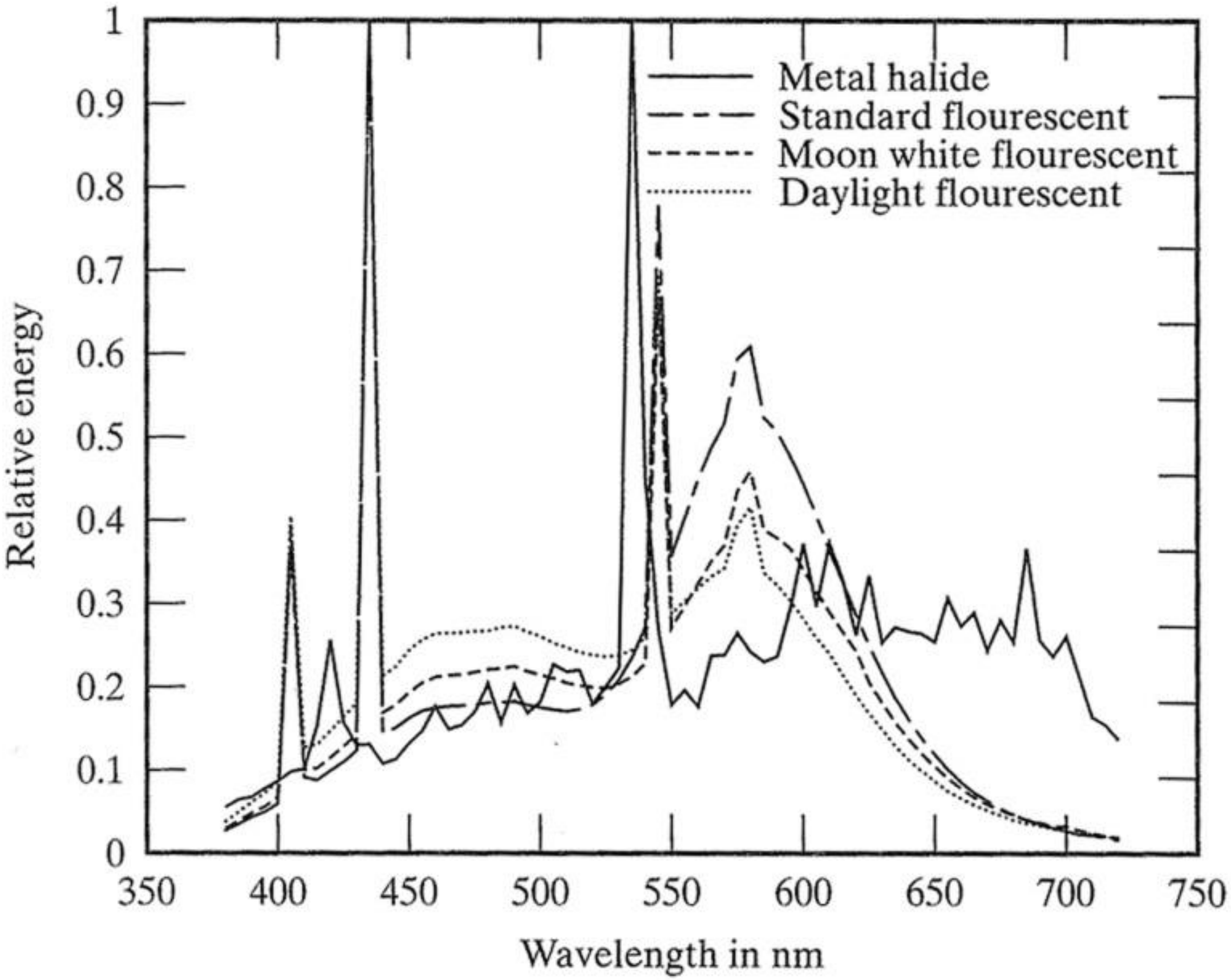


Figure from Ponce and Forsyth

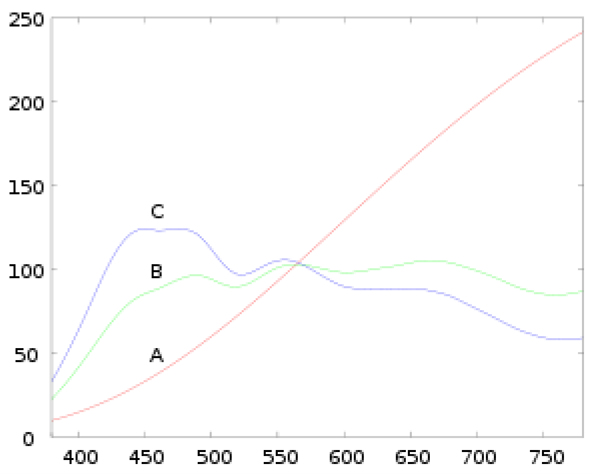
Lighting industry uses color temperature

Usage of color temperature in these ads relate to the perceived color of the bulb's light. The heat output of a typical LED bulb is between 60C-100C (~333-373K).

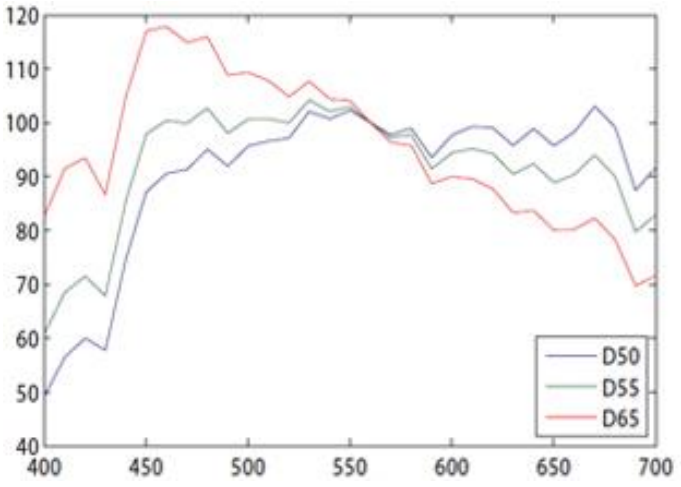
CIE standard illuminants

- CIE established several "synthetic" SPDs that serve as proxies for common real illuminants
- Illuminant A
  - tungsten-filament lighting (i.e. a standard light-bulb)
- Illuminant B
  - noon sunlight
- Illuminant C
  - average daylight
- Illuminant D series
  - represent natural daylight at various color temps (5000K, 5500K, 6500K), generally denoted as D50, D55, D65
- Illuminant E
  - idea equal-energy illuminant with constant SPD
  - does not represent any real light source, but similar to D55
- Illuminant F series
  - emulates a variety of fluorescents lamps (12 in total)

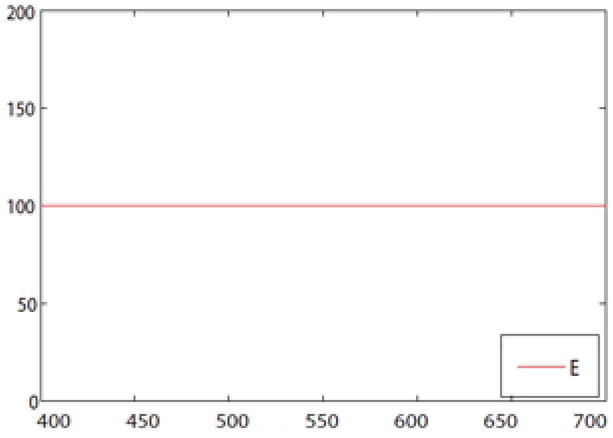
CIE standard illuminants



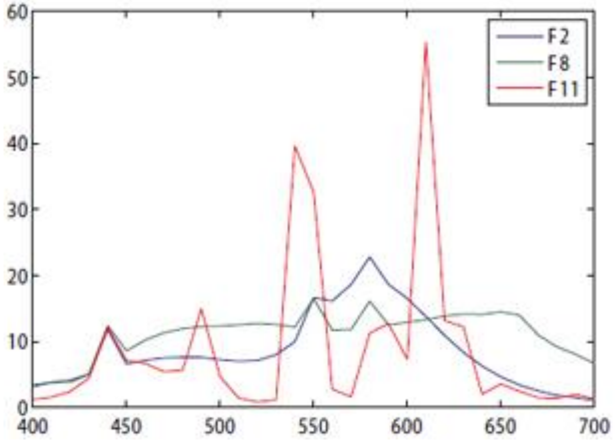
SPDs for CIE standard illuminant A, B, C



SPDs for CIE standard illuminant D50, D55, D65



SPDs for CIE standard illuminant E



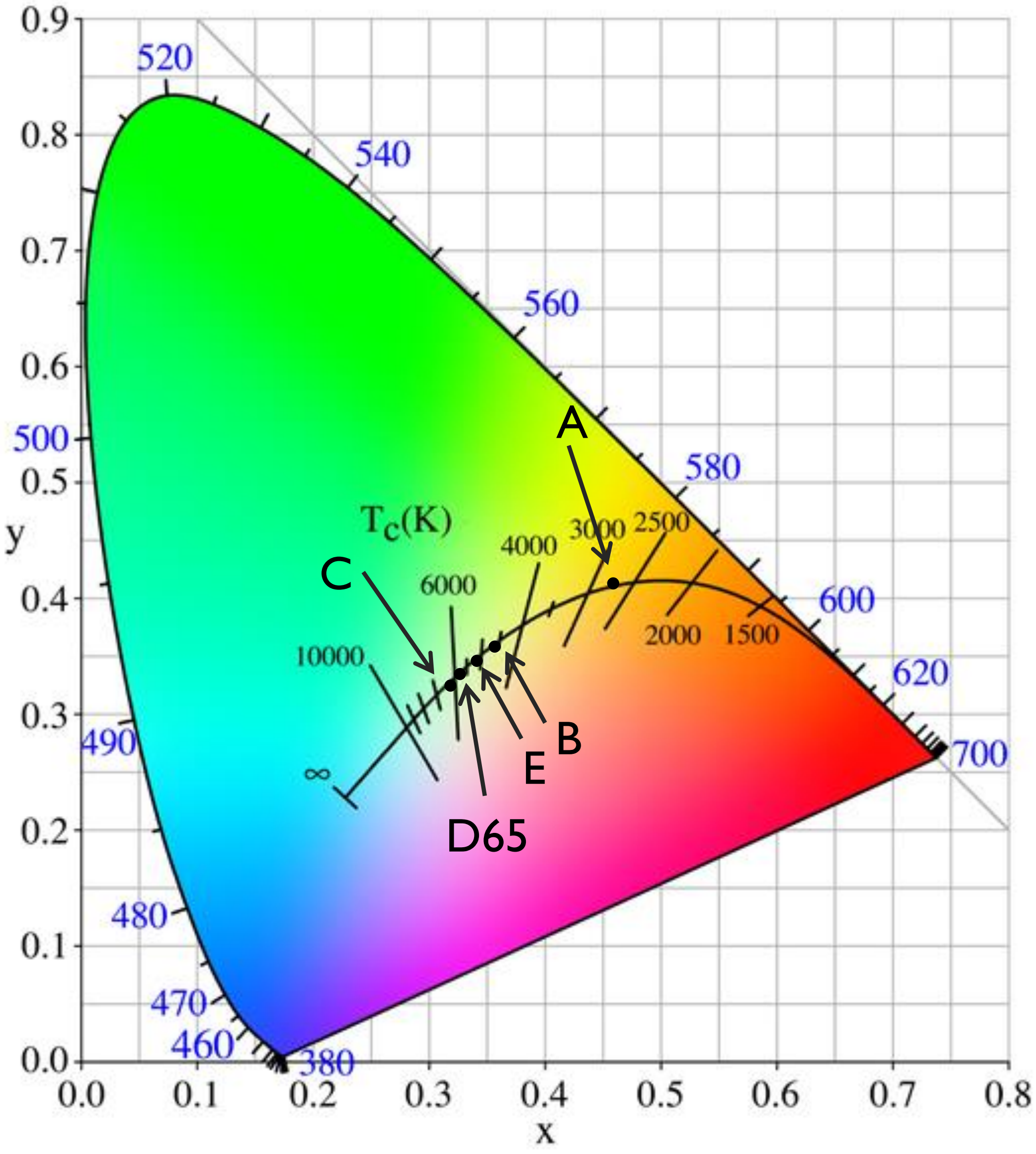
SPDs for CIE standard illuminants F2, F8, F11

D, E, and F series images from http://www.image-engineering.de

White point

- A white point is a CIE XYZ or CIE xyY value of an ideal "white target" or "white reference"
- This is essentially an illuminants SPD in terms of CIE XYZ/CIE xyY
  - We can assume the white reference is reflecting the illuminant
- The idea of chromatic adaptation is to make white points the same between scenes

White points in CIE xy chromaticity



CIE Illuminants

A, B, C, D65, E in terms of CIE xy

CIE	**x**	**,**	**y**
A	0.44757	,	0.40745
B	0.34842	,	0.35161
C	0.31006	,	0.31616
D65	0.31271	,	0.32902
E	0.33333	,	0.33333

Color constancy (at its simplest)


Johannes von Kries

- (Johannes) *Von Kries* transform
- Compensate for each channel corresponding to the L, M, S cone response

$$\begin{bmatrix} L_2 \\ M_2 \\ S_2 \end{bmatrix} = \begin{bmatrix} 1/L_{1w} & 0 & 0 \\ 0 & 1/M_{1w} & 0 \\ 0 & 0 & 1/S_{1w} \end{bmatrix} \begin{bmatrix} L_1 \\ M_1 \\ S_1 \end{bmatrix}$$

L_2, M_2, S_2 is the new LMS response with the illuminant divided "out". In this case white is equal to [1,1,1]

L_{1w}, M_{1w}, S_{1w} is the LMS response to "white" under this illuminant

L_1, M_1, S_1 are the input LMS space under an illuminant.

Illuminant to illuminant mapping

- More appropriate would be to map to another illuminant's LMS response (e.g. in the desired viewing condition)
- $(LMS)_1$ under an illuminant with white-response (L_{1w}, M_{1w}, S_{1w})
- $(LMS)_2$ under an illuminant with white-response (L_{2w}, M_{2w}, S_{2w})

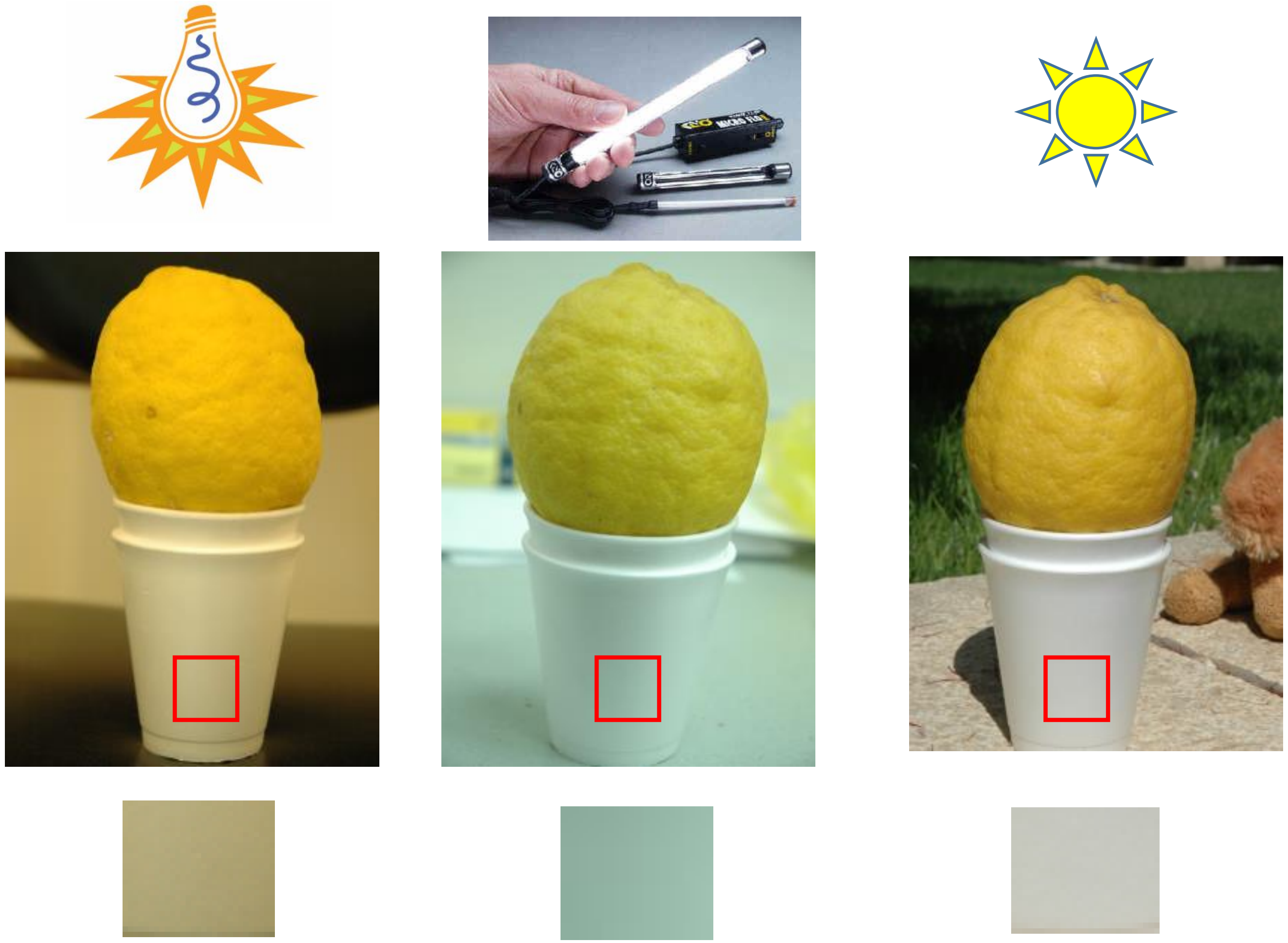
$$\begin{bmatrix} L_2 \\ M_2 \\ S_2 \end{bmatrix} = \begin{bmatrix} L_{2w}/L_{1w} & 0 & 0 \\ 0 & M_{2w}/M_{1w} & 0 \\ 0 & 0 & S_{2w}/S_{1w} \end{bmatrix} \begin{bmatrix} L_1 \\ M_1 \\ S_1 \end{bmatrix}$$

L_2, M_2, S_2 is the new LMS response with the illuminant divided "out" and scaled to LMS_2 illuminant

L_{1w}, M_{1w}, S_{1w} is the LMS response to "white" the input illuminant, L_{2w}, M_{2w}, S_{2w} response to "white" of output illuminant

L_1, M_1, S_1 are the input LMS space under an illuminant.

Example

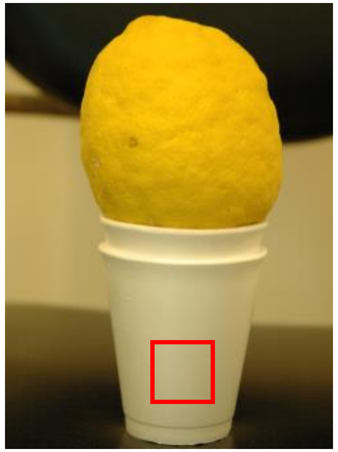


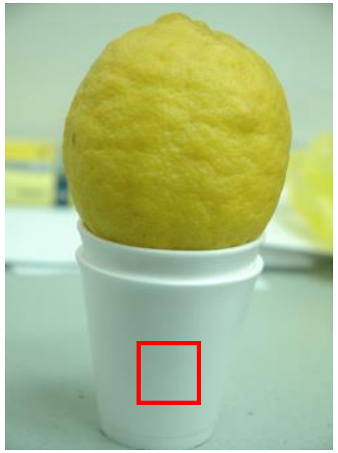
Simulation of different "white points" by photographing a "white" object under different illumination.

Images courtesy of Sharon Albert (Weizmann Institute)

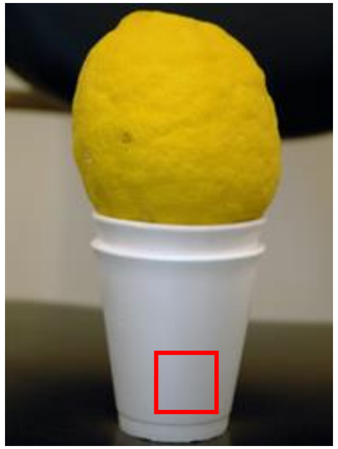
Example

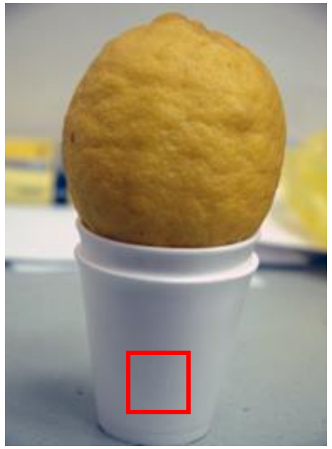
Input

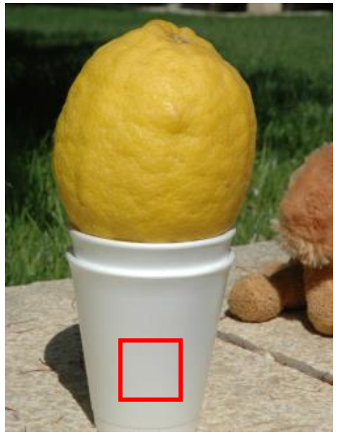




Adapted to
“target”
illuminant





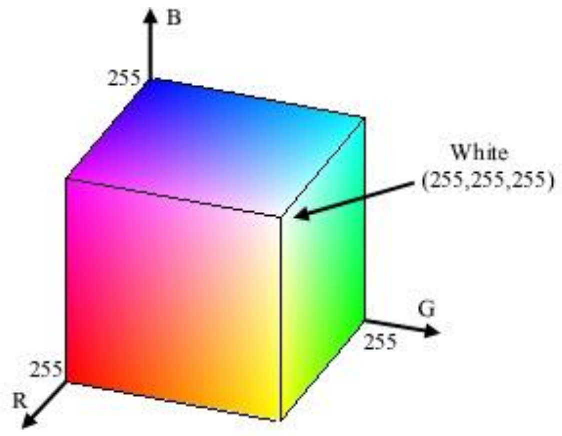


Before After Before After Target Illumination

Now we are finally done with color?
Almost (really) . . .

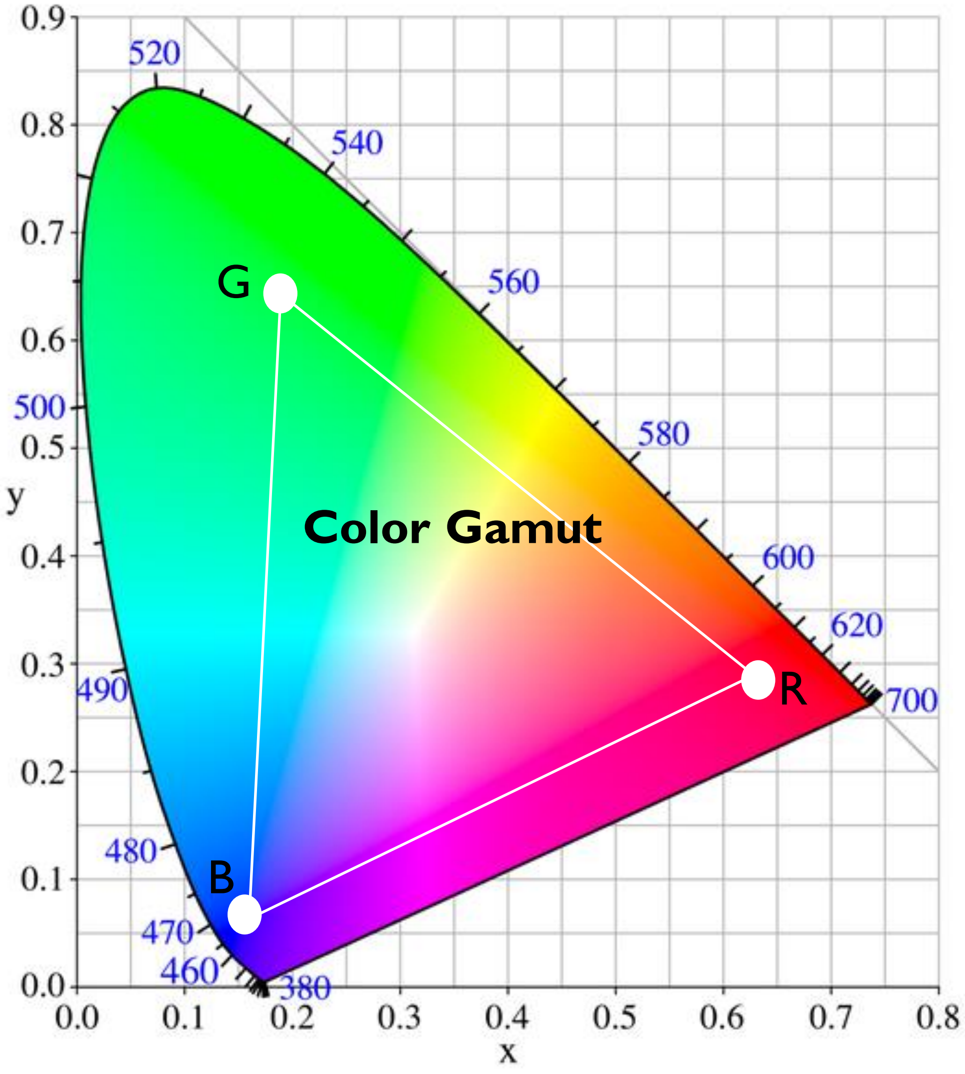
CIE XYZ and RGB

- While CIE XYZ is a canonical color space, images/devices rarely work directly with XYZ
- XYZ are not real primaries
- RGB primaries dominate the industry
- We are all familiar with the RGB color cube



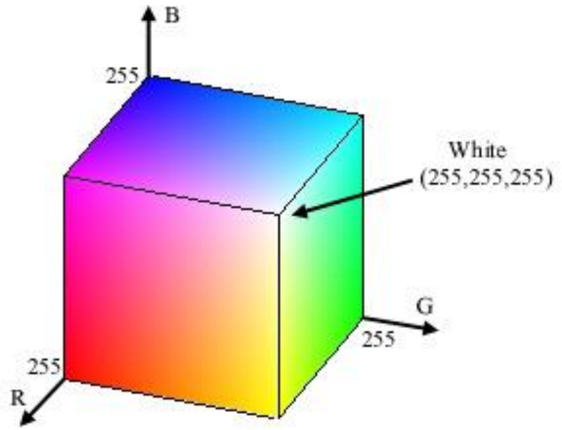
But by now, you should realize that Red, Green, Blue have no quantitative meaning. We need to know their corresponding SPDs or CIE XYZ values

Device specific RGB values

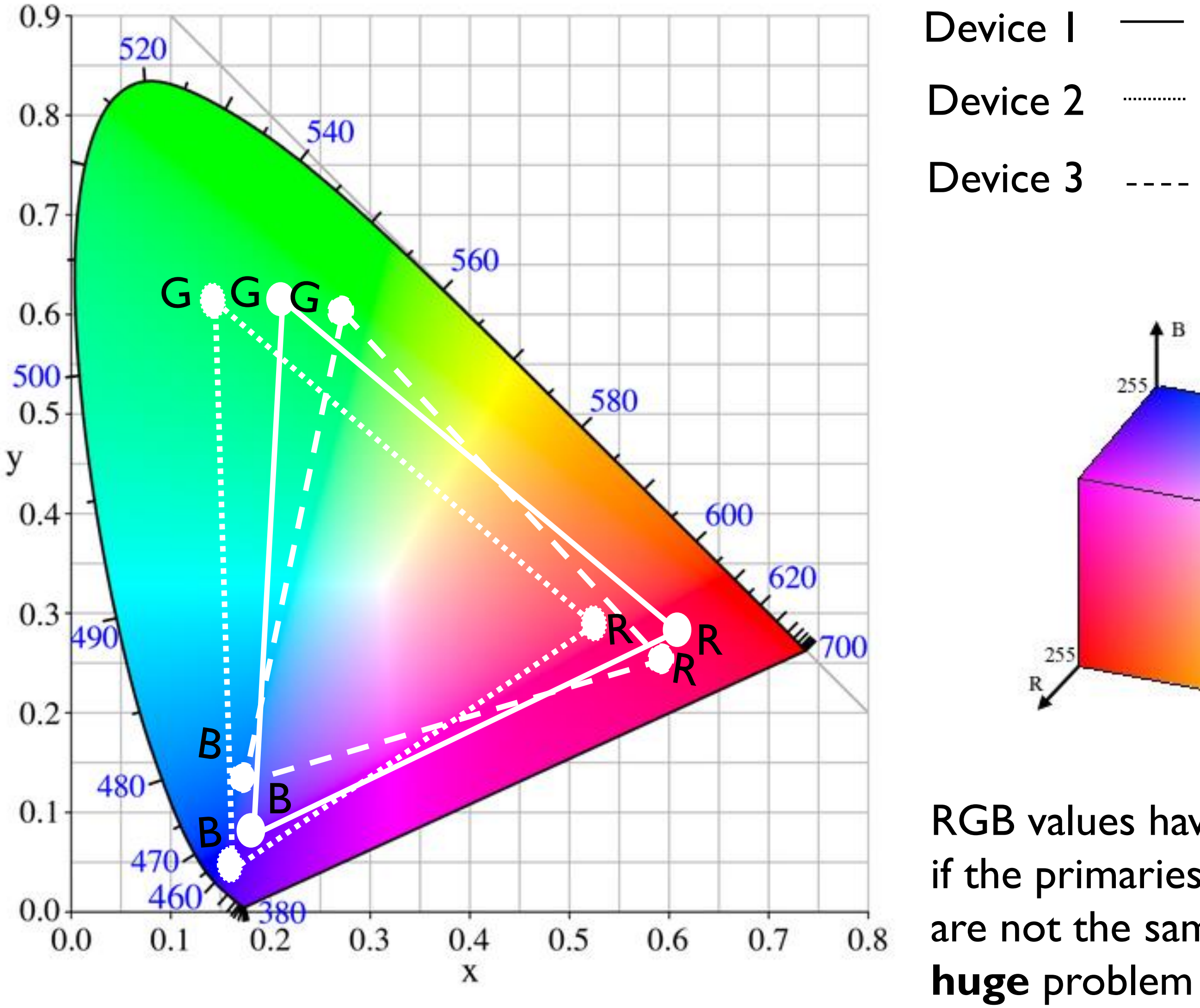


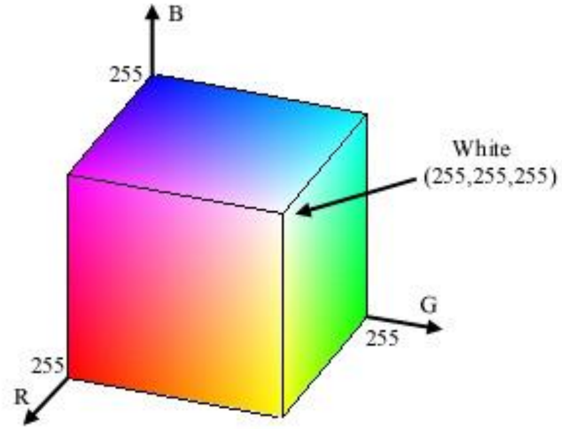
The RGB values span a subspace of CIE-XYZ to define the devices gamut.

If you have RGB values, they are specific to a particular device .



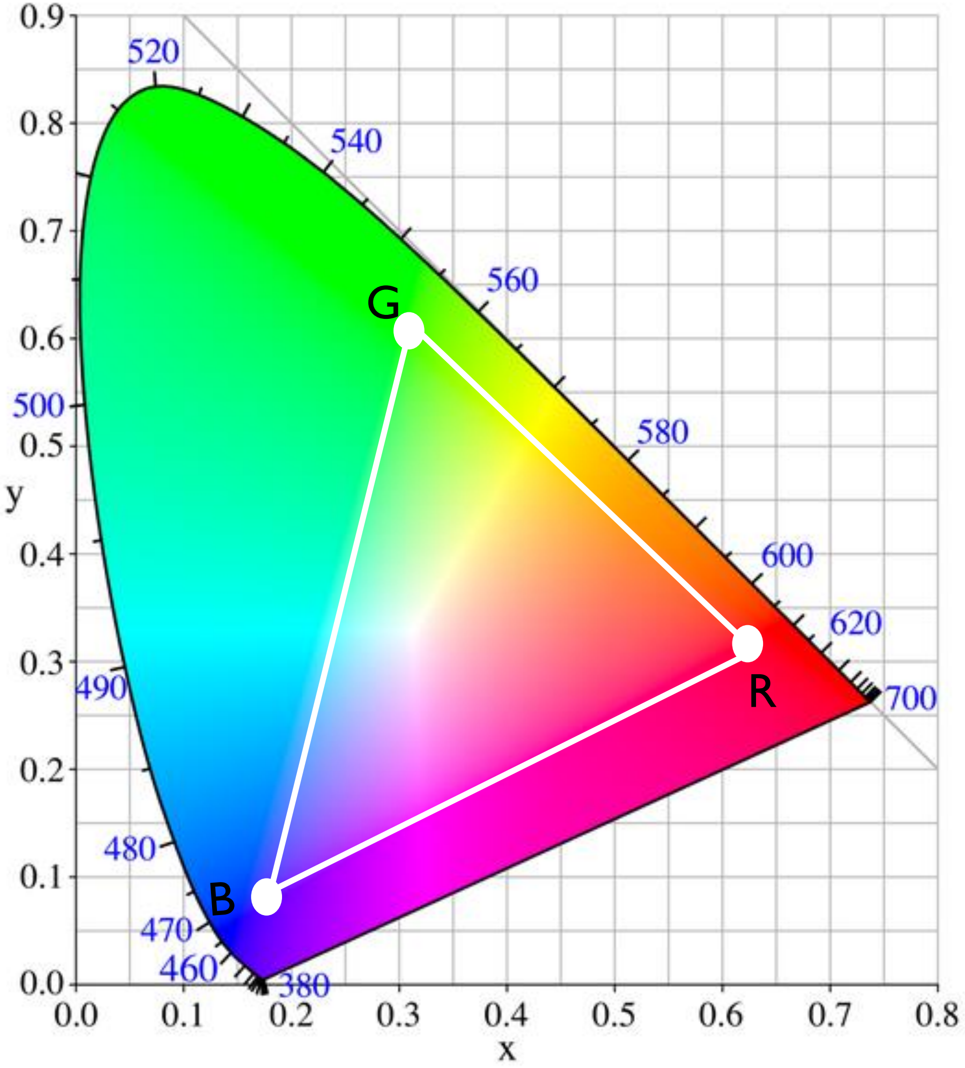
Trouble with RGB





RGB values have no meaning if the primaries between devices are not the same! This is a **huge** problem for color reproduction from one device to the next.

Standard RGB (sRGB) – Rec.709



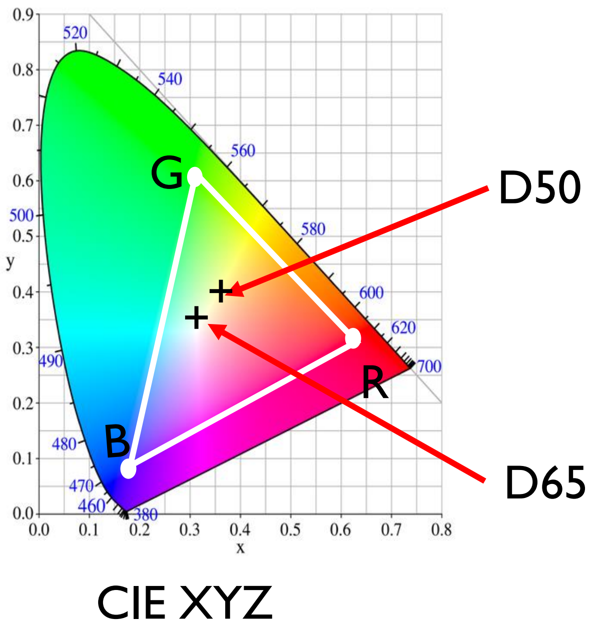
In 1996, Microsoft and HP defined a set of "standard" RGB primaries.
R=CIE xyY (0.64, 0.33, 0.2126)
G=CIE xyY (0.30, 0.60, 0.7153)
B=CIE xyY (0.15, 0.06, 0.0721)

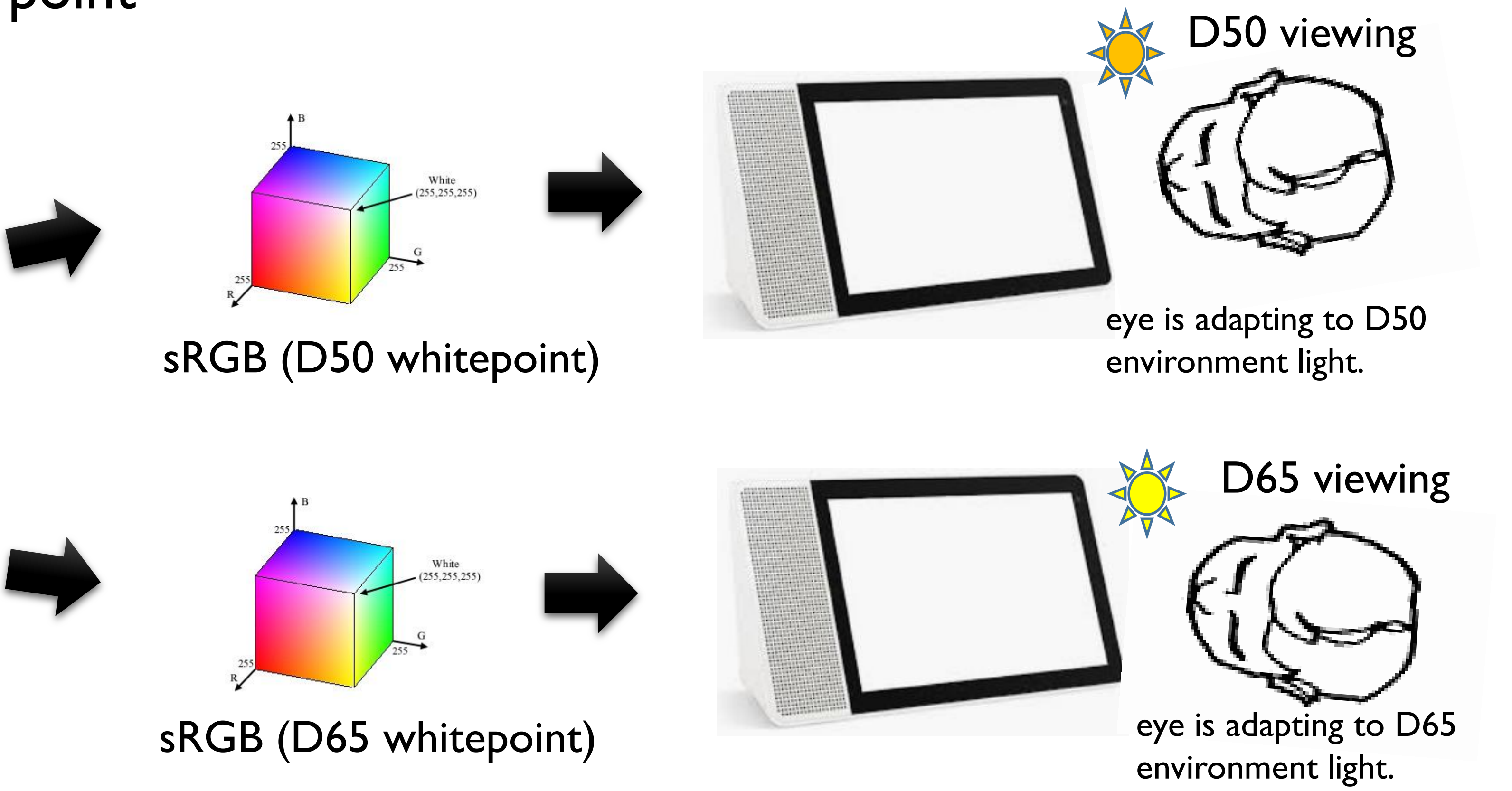
This was considered an RGB space achievable by most devices at the time.

White point was set to the D65 illuminant. **This is an important thing to note.** It means sRGB has built in the assumed viewing condition (6500K daylight).

sRGB's white point

- When we map from CIE XYZ to a color space, we need to specify the white point, --i.e., what is the CIE XYZ of "white" where we will view our image.
- This is to match the assumed viewing condition of my device
  - While in my sRGB color space, the white-point is r=g=b, it was transform from CIE XYZ at a particular white-point





The positions of the white-point locations are exaggerated here.

CIE XYZ to sRGB conversion

Matrix conversion:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.2404542 & -1.5371385 & -0.4985314 \\ -0.9692660 & 1.8760108 & 0.0415560 \\ 0.0556434 & -0.2040259 & 1.0572252 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

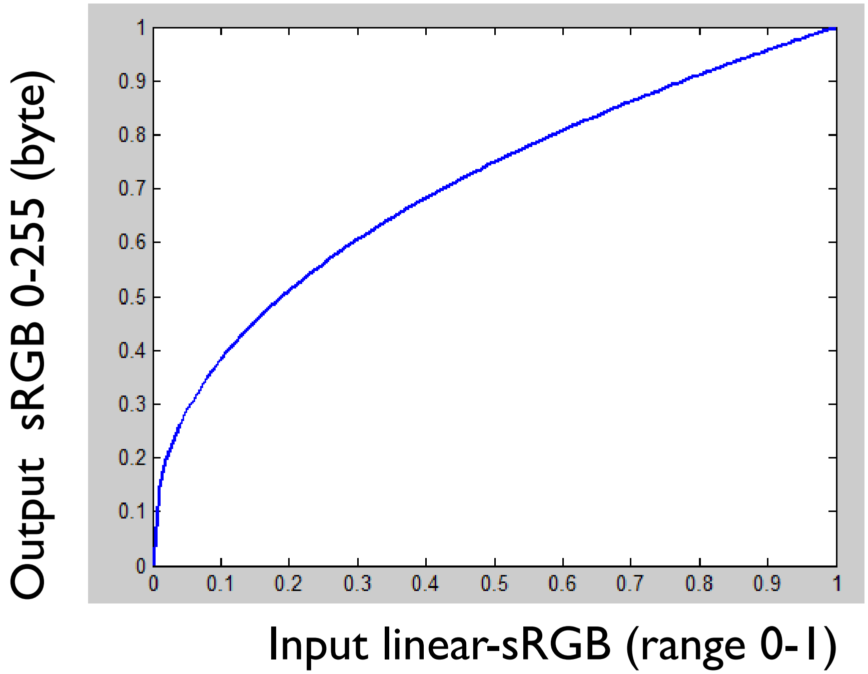
Linearized sRGB (D65)

CIE XYZ

- D65 is taken as the white-point
- This is the linear-sRGB space
- sRGB also specifies a gamma correction of the values
- The CIE refers this as the Recommendation 709 color space – or Rec.709

* Thanks to Boris Venet for finding an error in the matrix above in earlier online versions of these slides posted before March 2020.

sRGB gamma curve



This is a close approximation of the actual sRGB gamma

Actual formula is a bit complicated, but effectively this is gamma ($I' = 255*I^{(1/2.2)}$), where I' is the output intensity and I is the linear sRGB ranged 0-1, with a small linear transfer for linearized sRGB values close to 0 (not shown in this plot).

Stevens' power law

- Physical stimulus vs. perceptual sensation
- Stevens' Power Law

$$S = kI^{a}$$

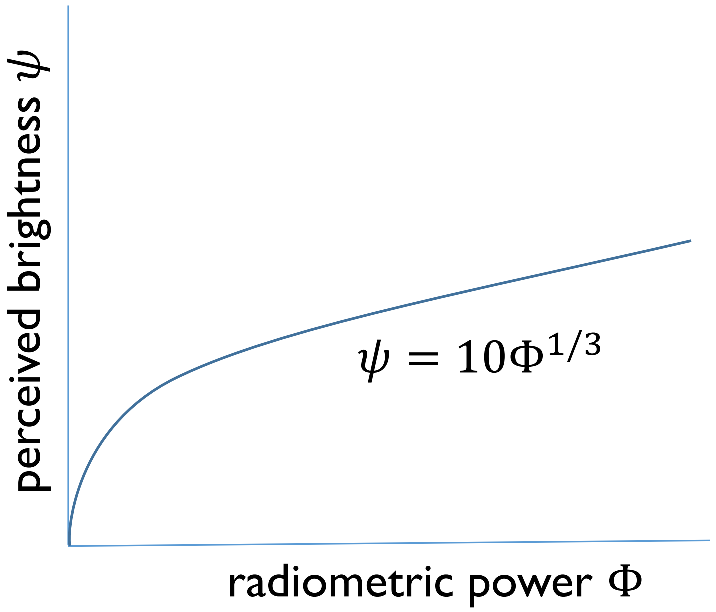
Human sensation

Constant

Stimulus intensity

power exponent

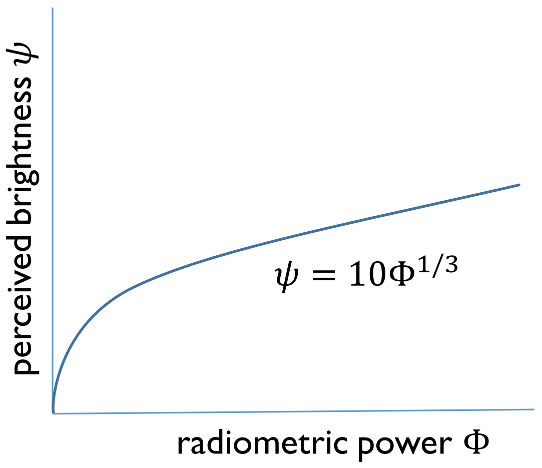
Dr. Stanley Stevens showed that most human sensations follow a power-law relationship between stimuli and sensation.

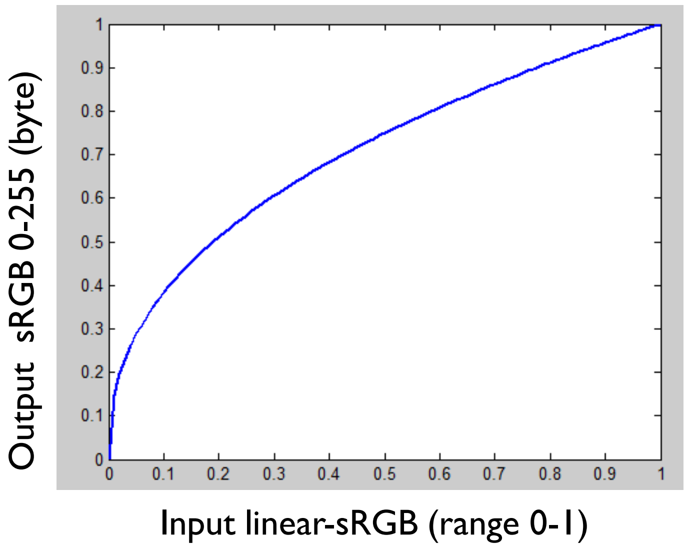


Stevens' model stated that human perception to brightness followed a cube-root power-law.

sRGB gamma

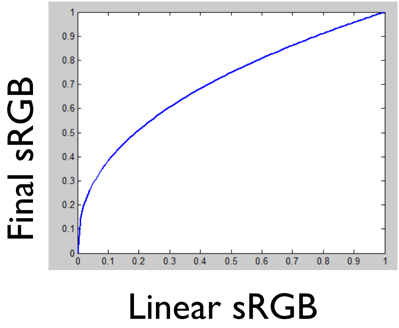
- The sRGB gamma encoding is related to the Steven's power-law
- The sRGB gamma is approximately a $\sqrt{3}$ power-law





Before (linear sRGB) & after (sRGB)

Linear sRGB



Final sRGB

An additional fun fact

- Physical stimulus vs. **human** sensations
- Stevens' Power Law

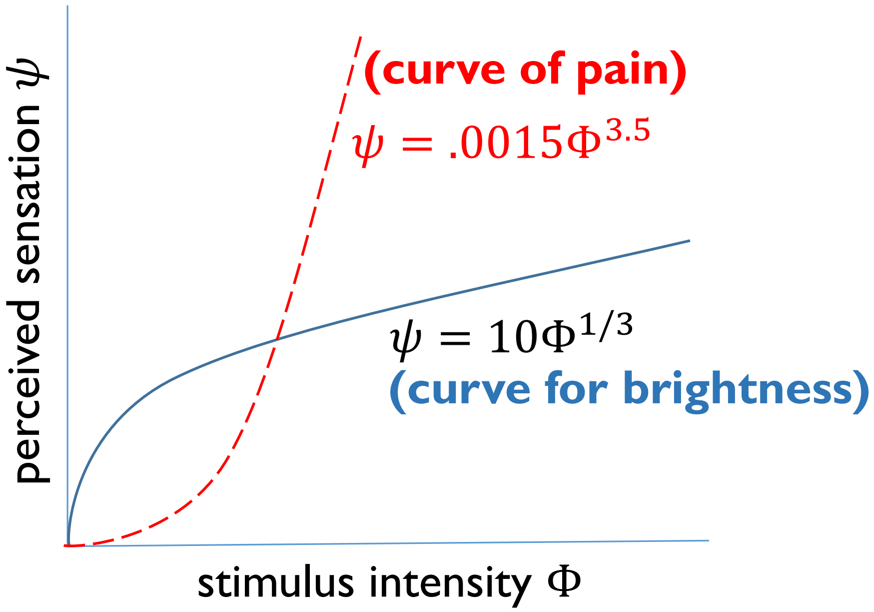
$$S = kI^{a}$$

Human sensation

Constant

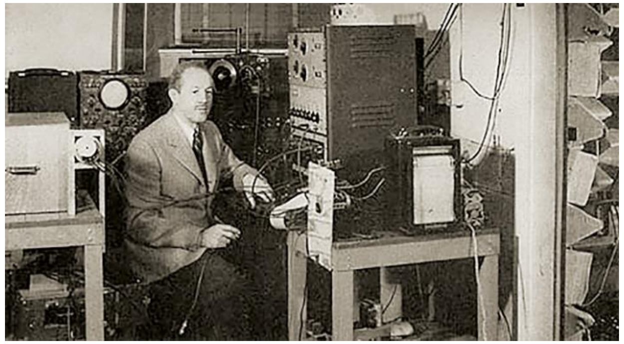
Stimulus intensity

power exponent

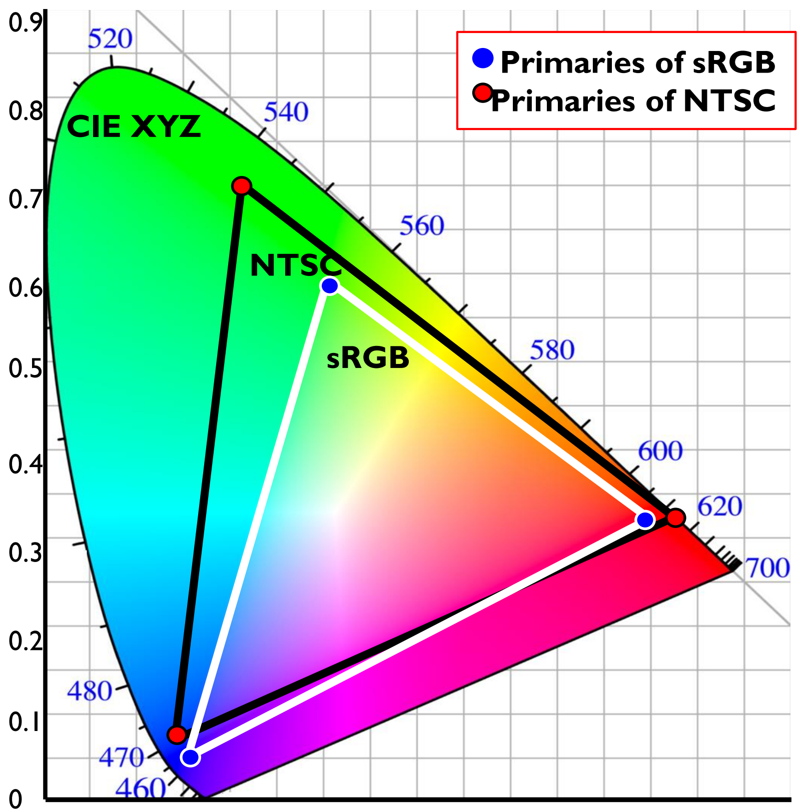


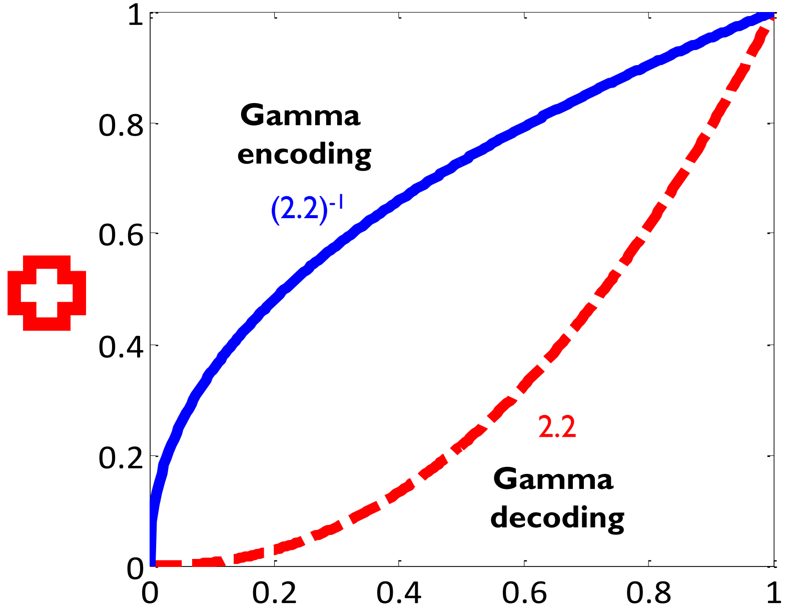
Dr. Stanley Stevens introduced showed that most human sensations follow a power-law relationship between stimuli and sensation.

Stevens also did experiment on the **pain sensation** of electrical shock! Turns out our sensitivity is the opposite than with radiometric power to brightness.



Standardization is not new - NTSC/PAL





Both NTSC and sRGB used gamma encodings.

CIE XYZ $\leftrightarrow$ NTSC/sRGB
(know your color space!)

It is important to known which color space your image is in.

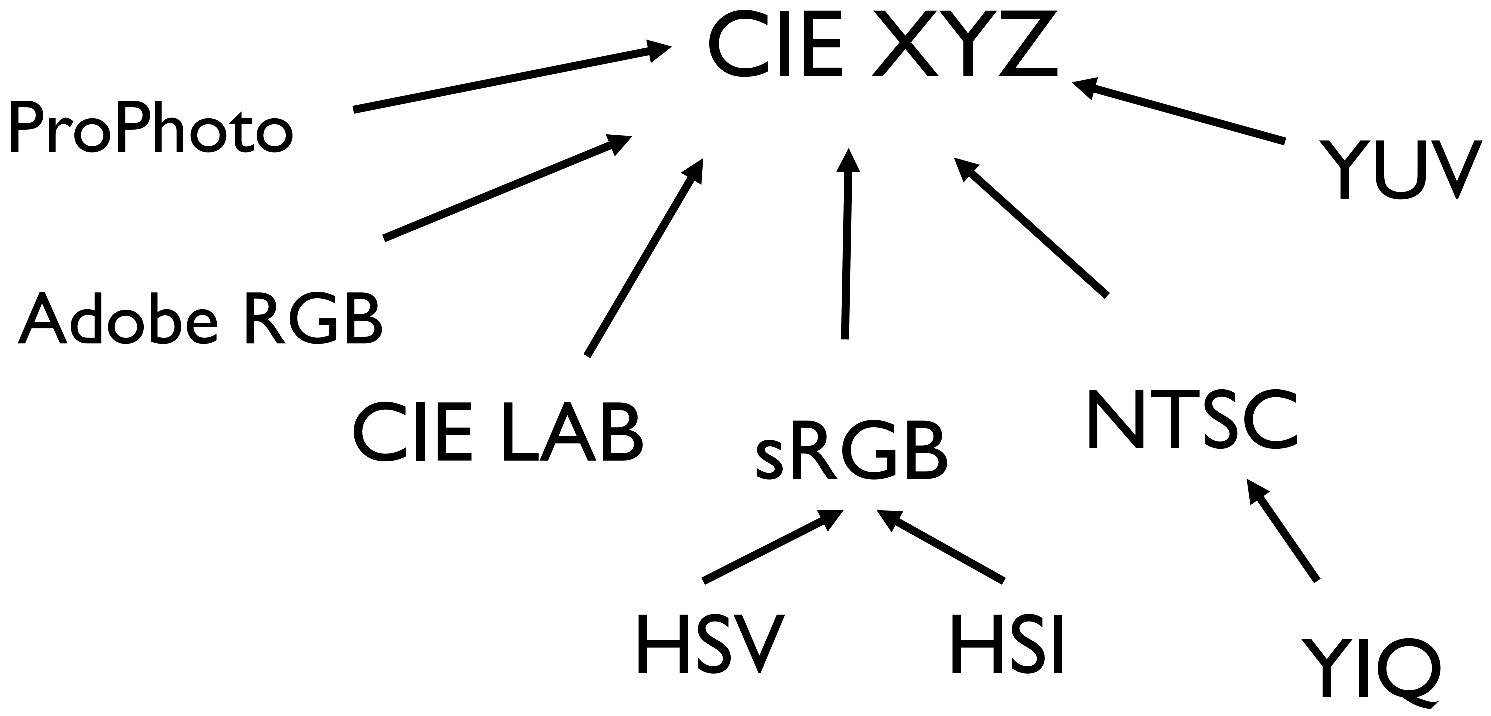
Linear-sRGB back to XYZ

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

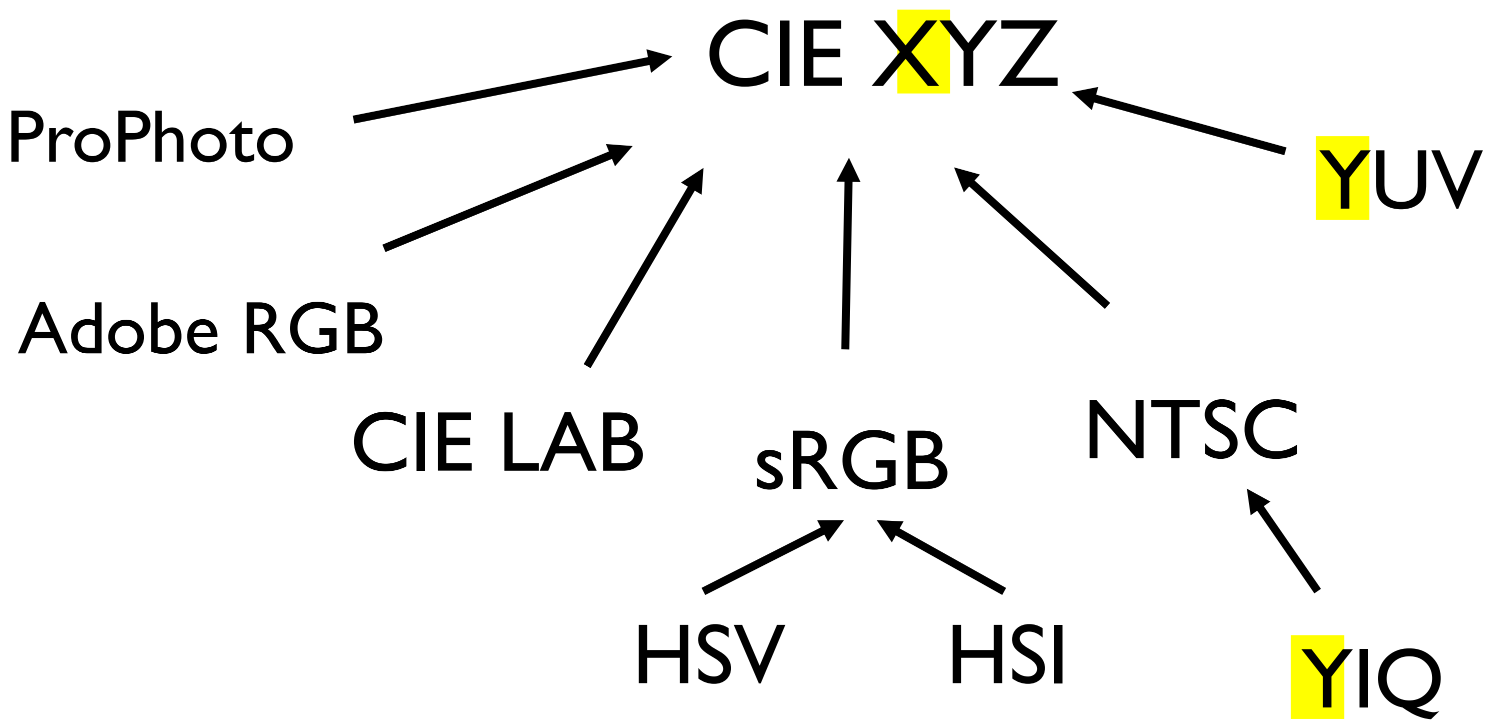
Linear-NTSC back to XYZ

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.6071 & 0.1736 & 0.1995 \\ 0.2990 & 0.5870 & 0.1140 \\ 0.0000 & 0.0661 & 1.1115 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

CIE XYZ: The mother of color spaces
grand



CIE XYZ: The mother of color spaces
grand



Be careful, not all Y's are the same!

Other common color spaces

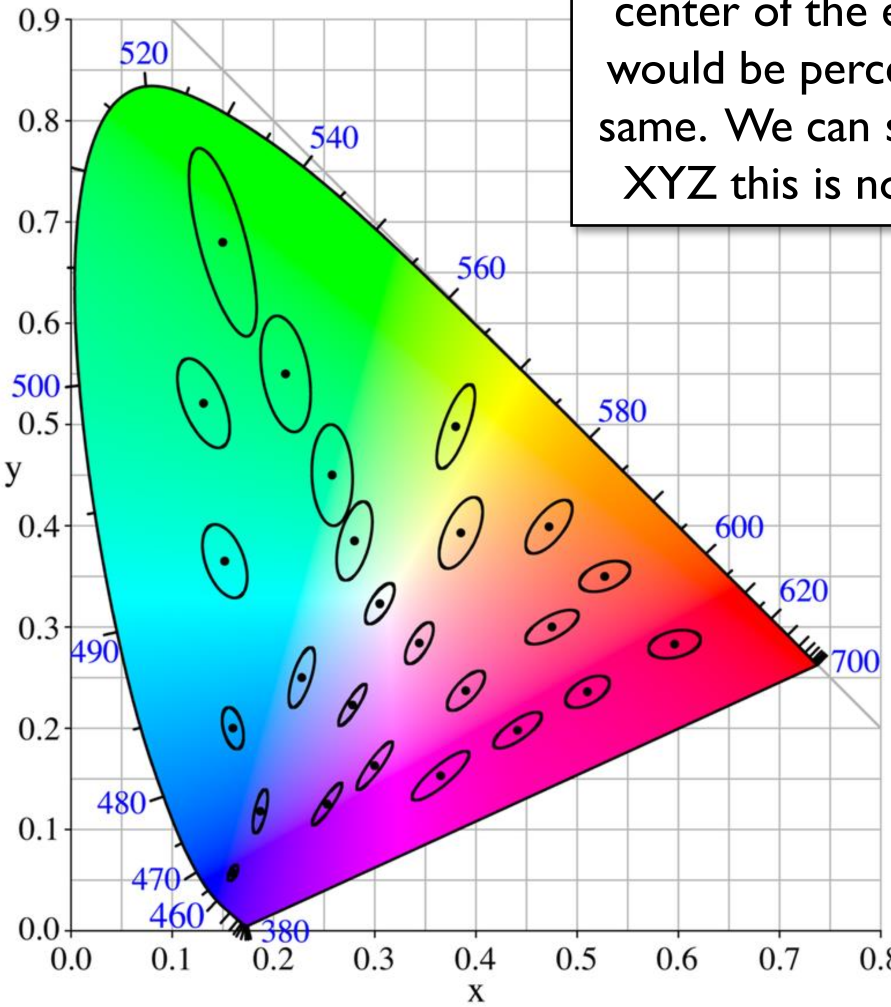
This tutorial does not go into the details of the mathematical transformations to other color spaces (we'd need another tutorial for that). You can find the transforms online.

The goal here is to explain the rationale behind each transform so you understand why the other color spaces are introduced.

CIE LAB space

- CIE LAB space (also written as CIE L*a*b*) was introduced as a perceptually uniform color space
- **Why?**
  - CIE XYZ provides a means to map between a physical SPD (radiometric measurement) to a colorimetric measurement (perceptual)
  - However, a uniform change in CIE XYZ space does result in an uniform change in perceived color difference (see diagram)
- CIE Lab transforms CIE to a new space where color (and brightness) differences are more uniform.

The ellipses shows the range of colors (around the center of the ellipse) that would be perceived as the same. We can see that CIE XYZ this is not uniform.



David MacAdam performed experiments into color perception. This plot is known as the MacAdam ellipses.

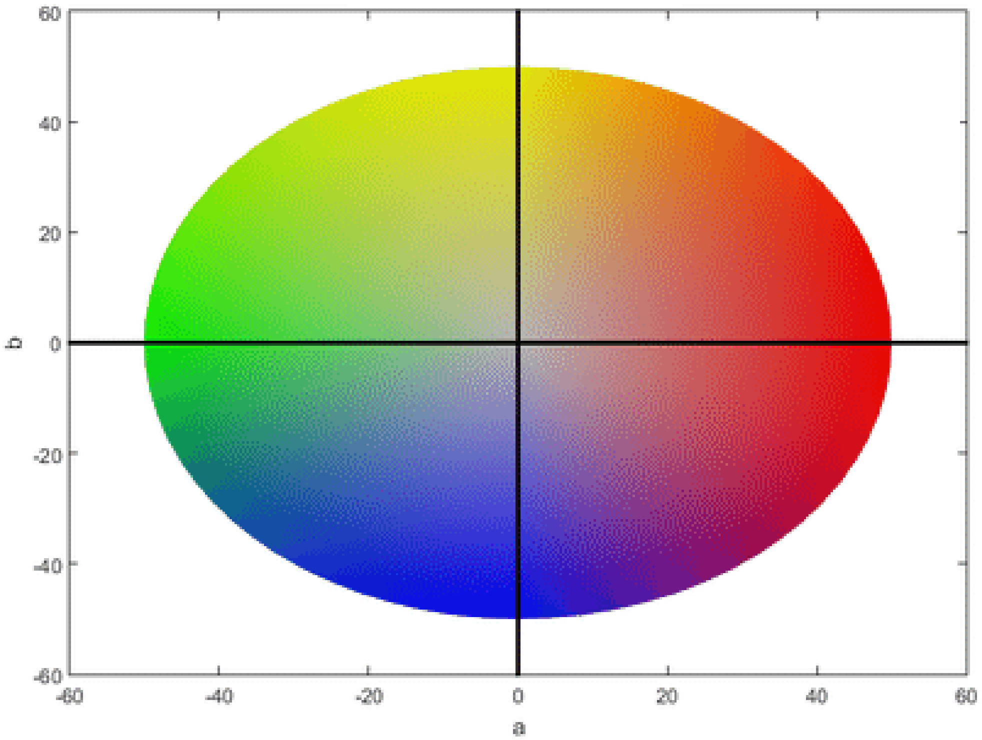
CIE 1976 LAB

- Considering the MacAdam experiments and the Steven's power-law, CIE LAB was derived in 1976 by applying various transformations to the CIE XYZ values that result in the following:
- L* represents a perceptual brightness measure between 0-100
  - L* is a non-linear transformation of the Y component of CIE XYZ.
  - L is approximately a cube root of Y (directly from Steven's power law)
- a* and b* (often range ±100)
  - Both have similar non-linear transformations applied, and represent approximately:
  - a* values lying along colors related to red and green
  - b* values lying along colors related to yellow and blue
  - a*=b*=0 represents neutral grey colors

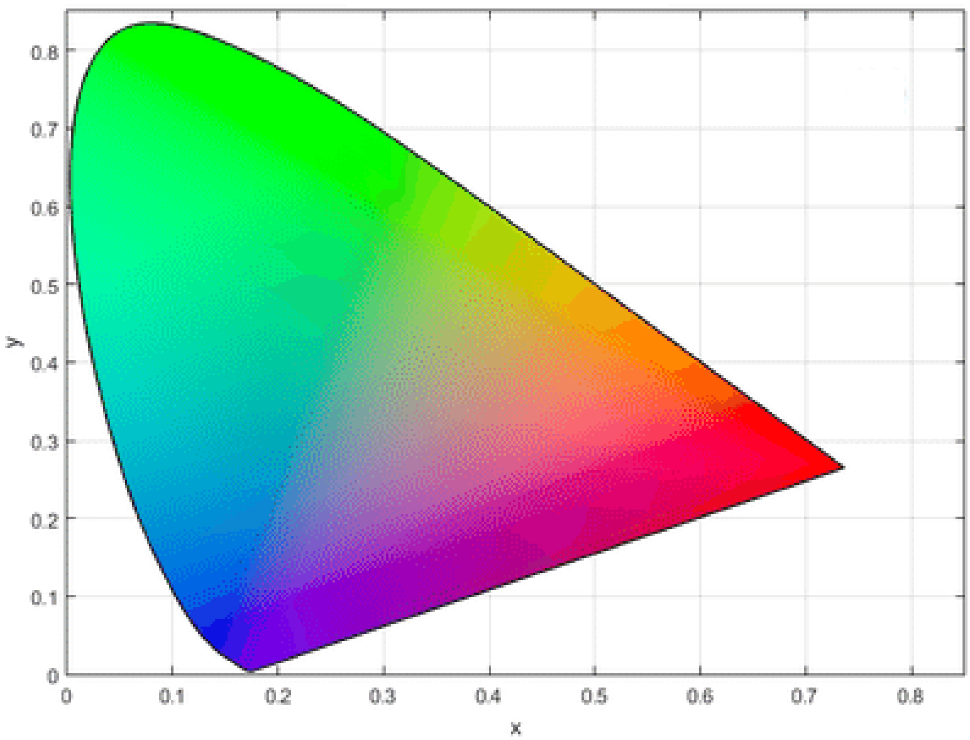
  NOTE: CIE LAB requires the white-point to be specified for the transformation.
  The default white-point is D65.

CIE LAB

CIE-L*ab space



CIE-xyY space



Chromaticity comparison's between CIE LAB and CIE XYX

Image from Bagdasar et al ICSTCC'17

Y'UV, Y'IQ, Y'CrCb

- These spaces are color decompositions that separate the RGB space into a "brightness-like" component and chrominance (color) components.
- The Y in these color spaces are *not defined* on linear-sRGB or linear-NTSC
- They are defined on the ***gamma encoded*** sRGB and NTSC color spaces
- These Y are referred to as "Luma", not Luminance
- It should be written as Y' but they are typically written as only Y

$$\begin{bmatrix} Y' \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ -0.09991 & -0.33609 & 0.436 \\ 0.615 & -0.55861 & -0.05639 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Gamma encoded (nonlinear) sRGB values

* Treating Y from YUV, YIQ, and YCrCb as CIE Y luminance is a common mistakes in the computer vision community.

Color error metric – CIE 2000 Delta E (ΔE)

- The CIE defined a color error metric in 2000 based on the CIE LAB space. This returns a color error between 0-100.
- You will see this referred to as CIEDE2000, CIEDE, ΔE, Delta E, DE, ..
- Delta E 2000 interpretation:

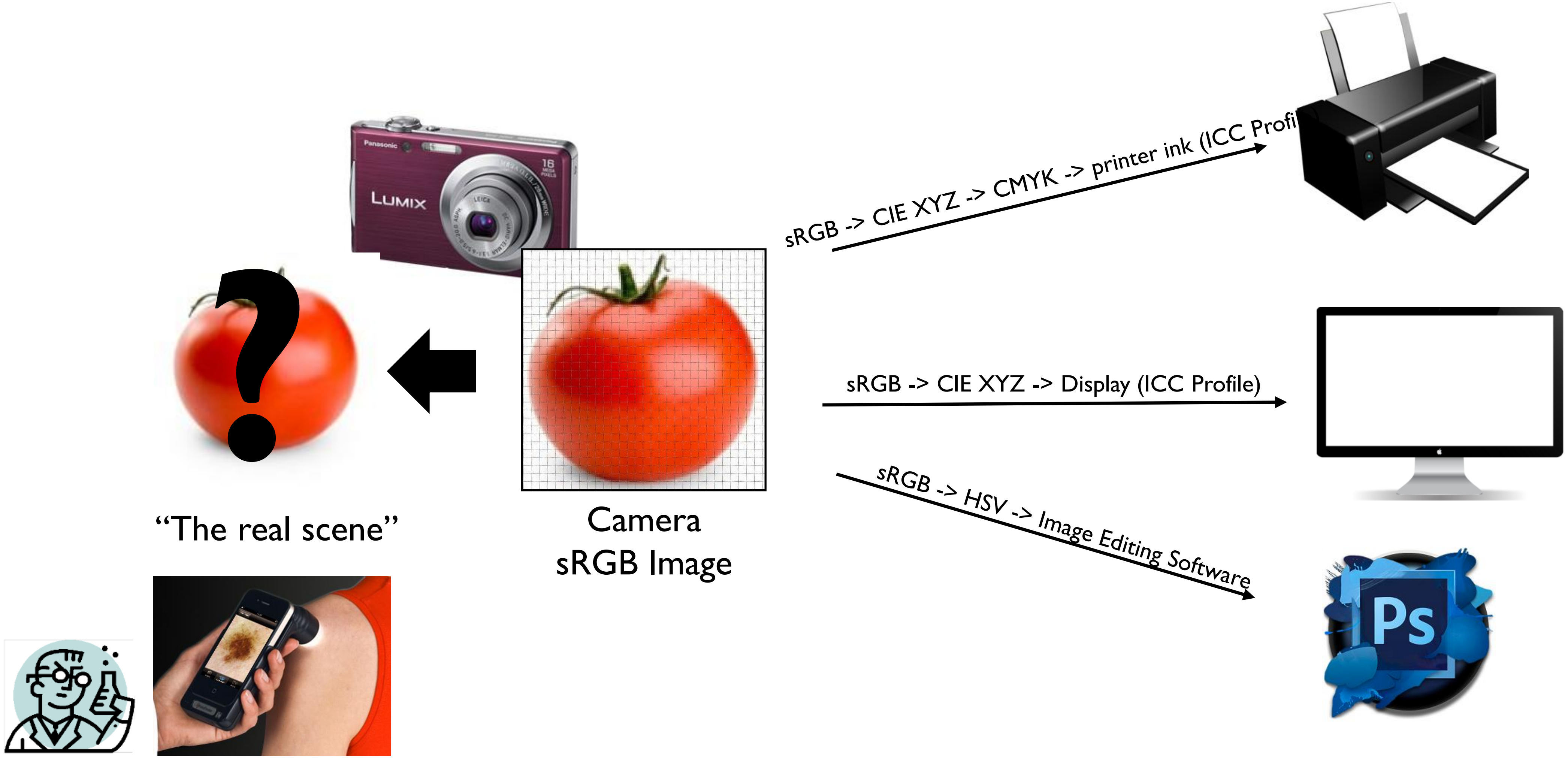
Delta E	Perception
<= 1.0	Not perceptible by human eyes.
1 - 2	Perceptible through close observation.
2 - 10	Perceptible at a glance.
11 - 49	Colors are more similar than opposite
100	Colors are exact opposite

In general, DE of 2 or less is considered to be very good. It means a standard observer could not tell that two colors are different unless they observed them very closely.

Table from
https://zschuessler.github.io/DeltaE/learn/

Congratulations!

Standard color spaces are great



Tutorial schedule

- Part 1 (General)
  - ~~Motivation~~
  - ~~Review of color & color spaces~~
  - Overview of in-camera imaging pipeline

1.30pm – 3.30pm

Break
3.30pm – 4.30pm

- Part 2 (Imaging and Computer Vision)
  - Misconceptions in the computer vision community regarding color
  - Recent work on color and cameras
  - Concluding remarks

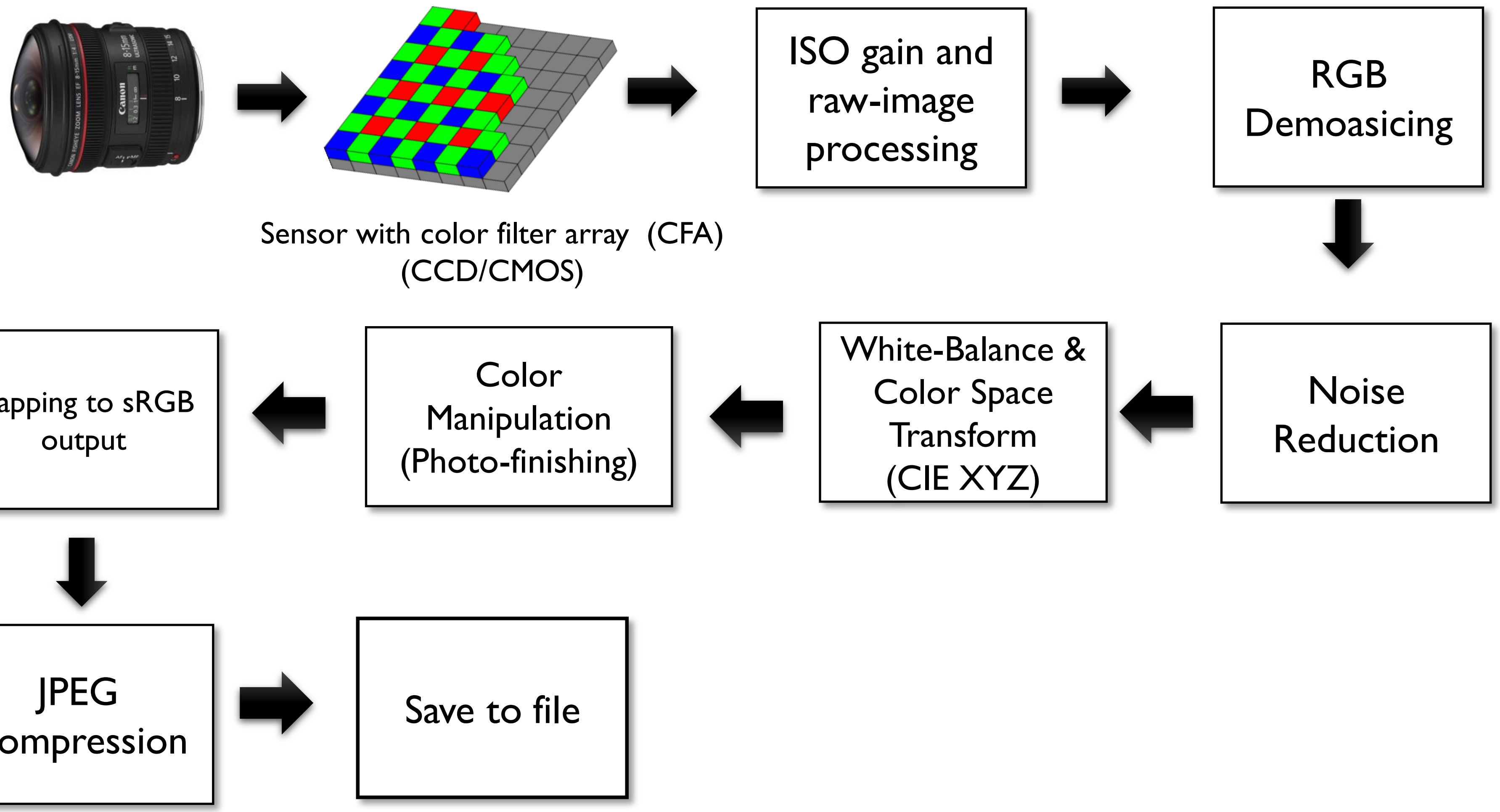
4.30pm – 6.00pm

Part 1: Overview of the Camera Imaging Pipeline

Integrated signal processor (ISP)

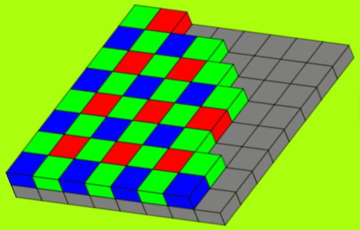
- You will hear the term "ISP" associated with camera pipelines
- An ISP is dedicated hardware used to process the sensor image to produce the final output (JPEG image) that is saved on your device
- The ISP is usually integrated as part of a system on a chip (SoC) that has other modules
- Companies such as Qualcomm, HiSilicon, Intel (and more) sell ISP chips
    - An ISP can be customized by the customer (Samsung, Huawei, LG, Apple, etc)
- Note that it is also possible to perform operations common on an ISP on your device's CPU and GPU

A typical color imaging pipeline



NOTE: This diagram represents the steps applied on a typical consumer camera pipeline. ISPs may apply these steps in a different order or combine them in various ways. A modern camera ISP will undoubtedly be more complex, but will almost certainly implement these steps in some manner.

A typical color imaging pipeline



Sensor with color filter array (CFA)
(CCD/CMOS)

→ ISO gain and raw-image processing

→ RGB Demoasicing

→ Noise Reduction

→ White-Balance & Color Space Transform (CIE XYZ)

→ Color Manipulation (Photo-finishing)

→ Mapping to sRGB output

→ JPEG Compression

→ Save to file

Camera sensor

CMOS sensor

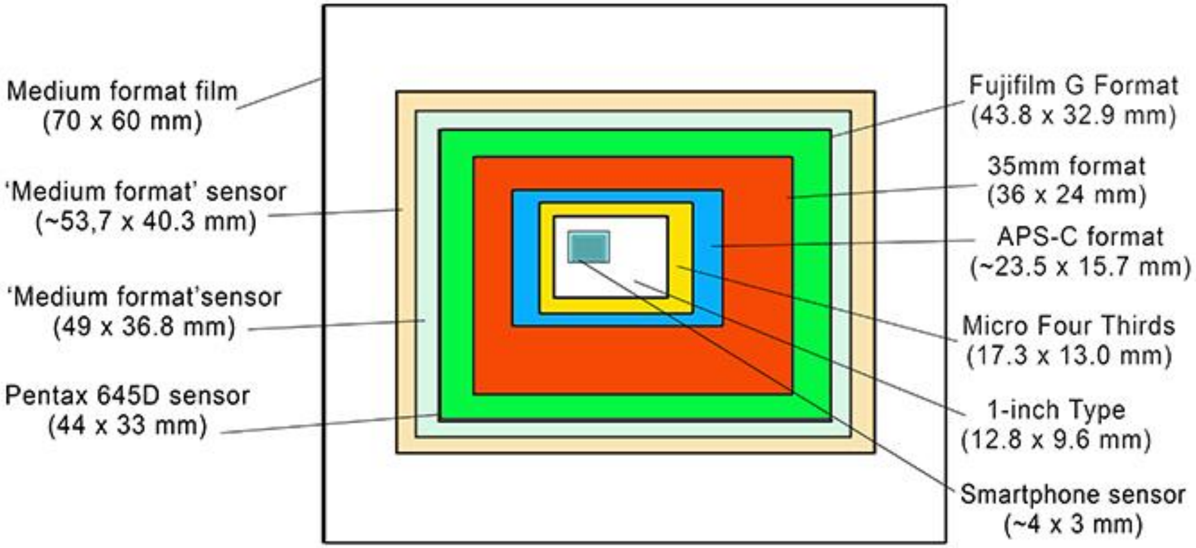
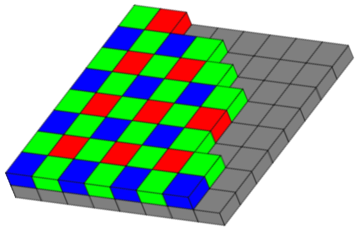


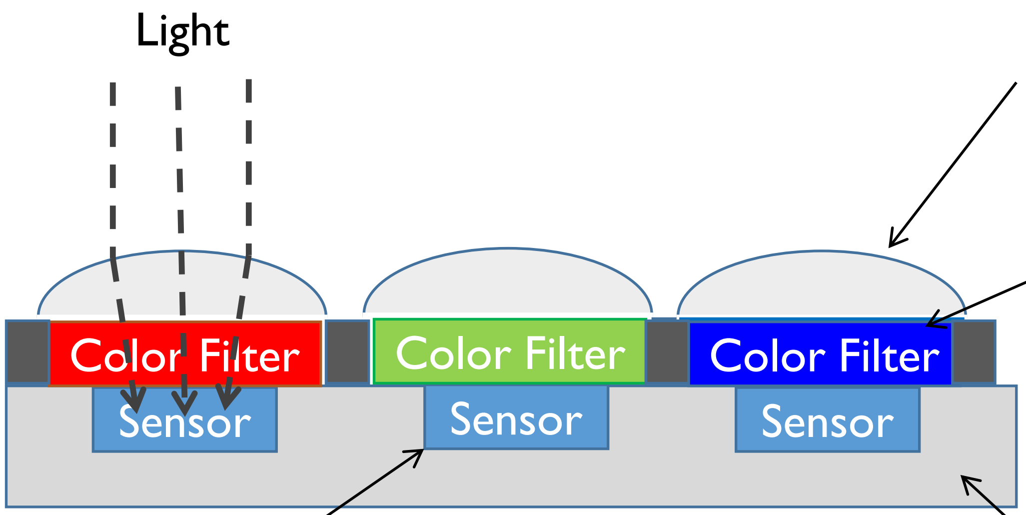
Figure from Photo Review website.

Almost all consumer camera sensors are based on complementary metal-oxide-semiconductor (CMOS) technology.
We generally describe sensors in terms of number of pixels and size. The larger the sensor, the better the noise performance as more light can fall on each pixel. Smart phones have small sensors!

Camera sensor RGB values



Color filter array
or "Bayer" pattern.



Micro-lenses are placed over the diode to help increase light collection on the sensor

Color filters place over the sensor. This forms a Color Filter Array (CFA) also called a "Bayer Pattern" after inventor Bryce Bayer.

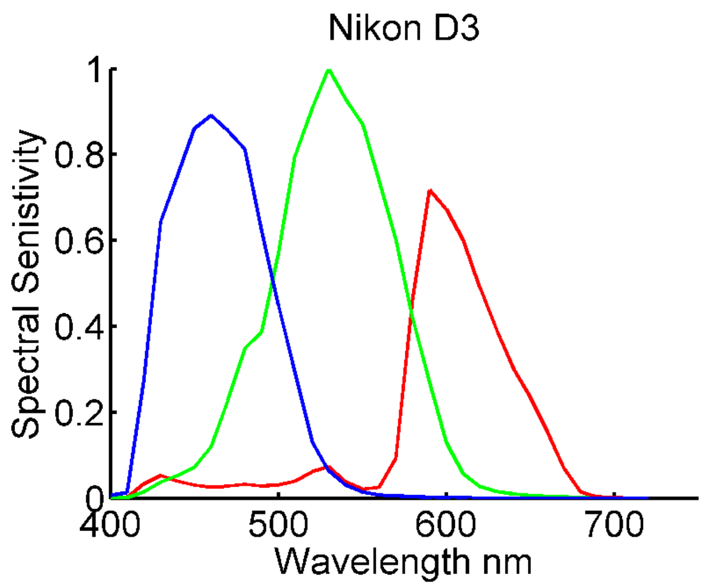
Photodiode
Photons hit the diode and force out electrons. This design is similar to a solar cell!

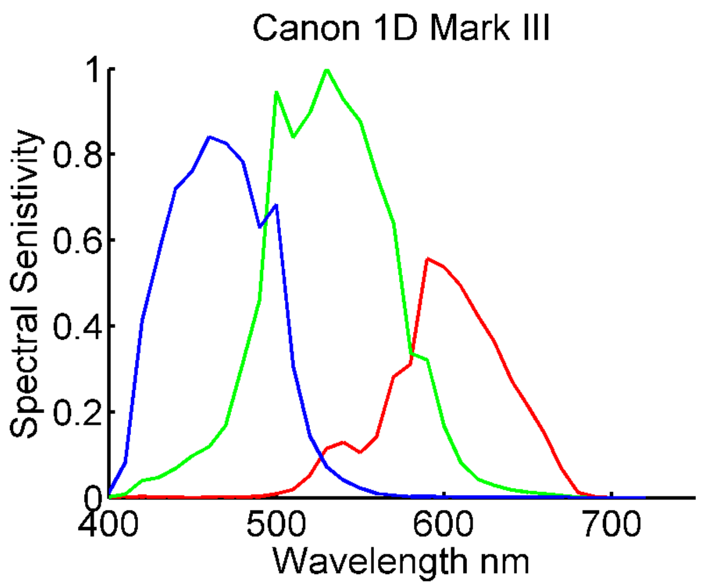
Silicon/Circuitry

Bryce Bayer

Camera RGB sensitivity

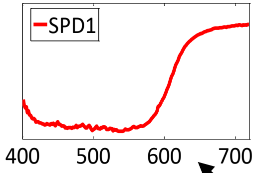
- The color filter array (CFA) on the camera filters the light into three *sensor-specific* RGB primaries

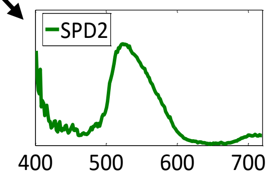




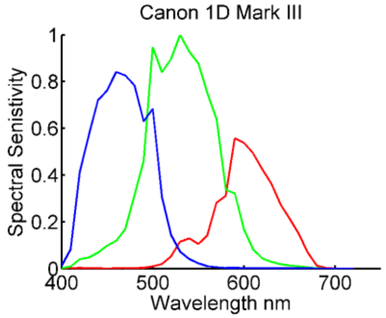
Plotted from camera sensitivity database by Dr. Jinwei Gu from Rochester Institute of Technology (RIT). Dr. Gu is now at SenseTime (USA). http://www.cis.rit.edu/jwgu/research/camspec/

Sensor raw-RGB image





Remember: physical world is measured by radiometric spectral power distributions.



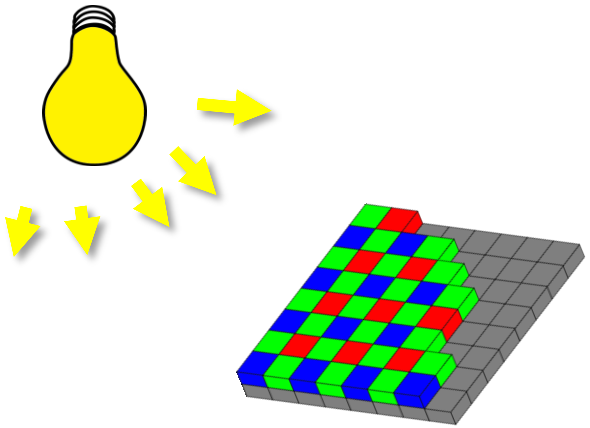
Color Matching Functions
CIE XYZ

Your camera sensor RGB filter is sensitive to different regions of the incoming SPD.

raw-RGB represents the physical world's SPD "projected" onto the sensor's spectral filters.

Sensors are linear to irradiance

- Camera sensors are decent light measuring devices
- If you double the amount of light hitting a sensor's pixel, the digital value output of that pixel will double

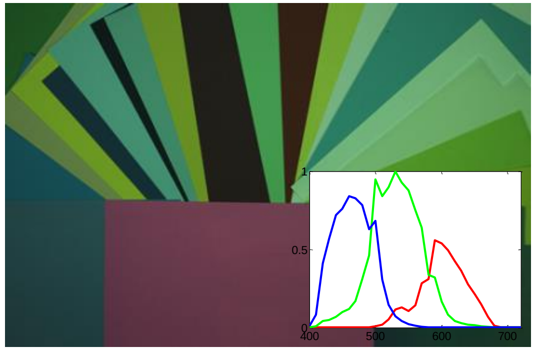


Sensor output is linear with respect to irradiance falling over the sensor over a certain amount of time.

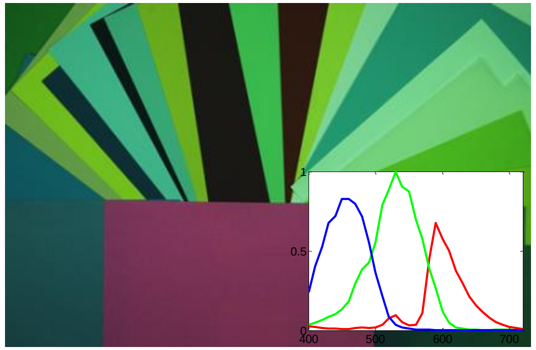
$$I = i * t$$

Digital value I is a linear function of irradiate i and exposure t.

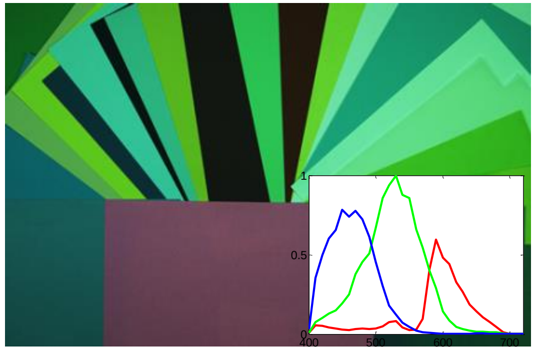
IMPORTANT: raw-RGB sensor images are <u>not</u> in a standard color space



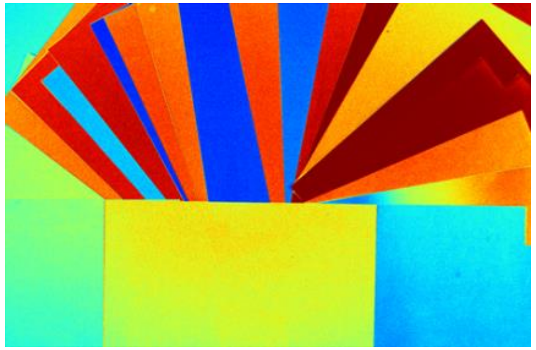
Canon 1D



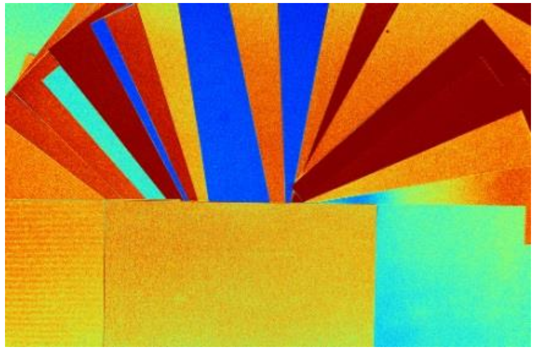
Nikon D40



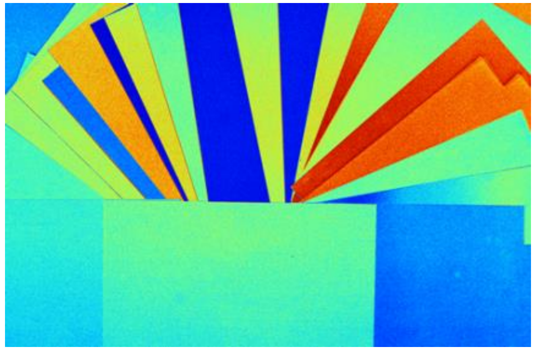
Sony $\alpha57$



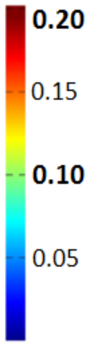
$||$Canon 1D – Nikon D40$||_2$



$||$Canon 1D - Sony $\alpha57||_2$



$||$Nikon D40 - Sony $\alpha57||_2$



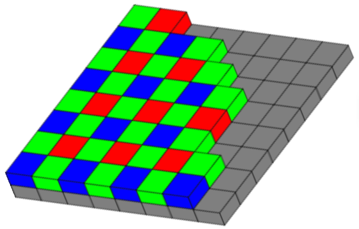
Color plots show L2 distance between the raw-RGB values with different cameras.

Displaying raw-RGB images

- Inserting a raw-RGB image in your slides, research paper, etc will result in strange colors.
- Why? Our devices (computers, printers, etc) expect the image to be in a standard color space like sRGB.

This is a raw-RGB image. Why does it look bad?
Because the RGB values are not sRGB values.
Knowing your color space is important!

A typical color imaging pipeline



Sensor with color filter array (CFA)
(CCD/CMOS)

ISO gain and raw-image processing

RGB Demoasicing

Noise Reduction

White-Balance & Color Space Transform (CIE XYZ)

Color Manipulation (Photo-finishing)

Mapping to sRGB output

JPEG Compression

Save to file

ISO signal amplification (gain)

- Imaging sensor signal is amplified and digitized
- Amplification to assist A/D conversion
  - Need to get the voltage to the range required to the desired digital output
- This gain is used to accommodate camera ISO settings
  - Gain to signal applied on sensor
  - Note – gaining the signal also gains image noise

Different ISO settings (note: the exposure will be shorter for higher ISO)

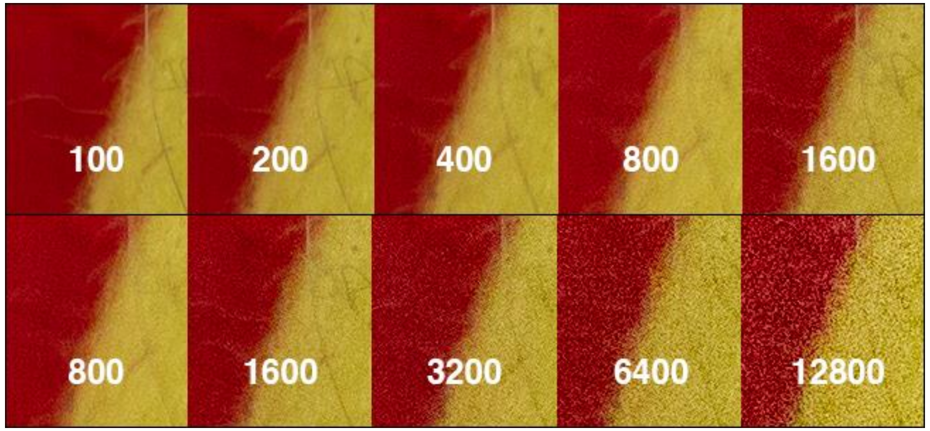


Image: Harry Guinness

ISO gain and raw-image processing

Pixel "intensity"

- We often talk about a pixel's intensity, however, a pixel's numerical value has no *unit*

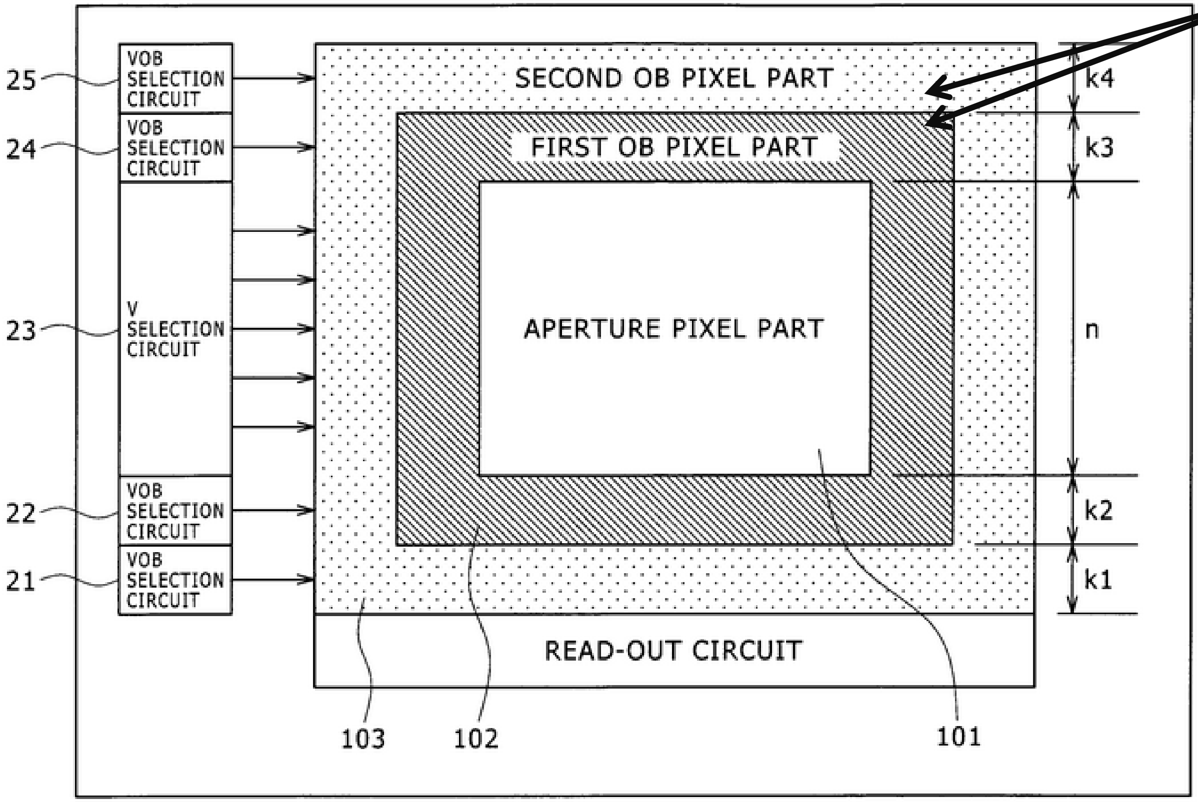
- The digital value of a pixel is based on several factors
  - Exposure (which is a function of both shutter speed and exposure)
  - Gain (ISO setting on the camera)
  - Camera hardware that digitizes the signal

- We typically rely on the ***relative*** digital values in the image and not the absolute digital values

Black light subtraction

- Sensor values for pixels with "no light" should be zero
- This is not the case due to sensor noise
  - The black level often changes as the sensor heats up
- This can be corrected by capturing a set of pixels that do not see light
- Place a dark-shield around sensor
- Subtract the level from the "black" pixels

Optical black (OB)



Sensor area(s) capturing "optical black"

Black light capturing areas (likely exaggerated) from Sony US Patent US8227734B2 (Filed 2008) .

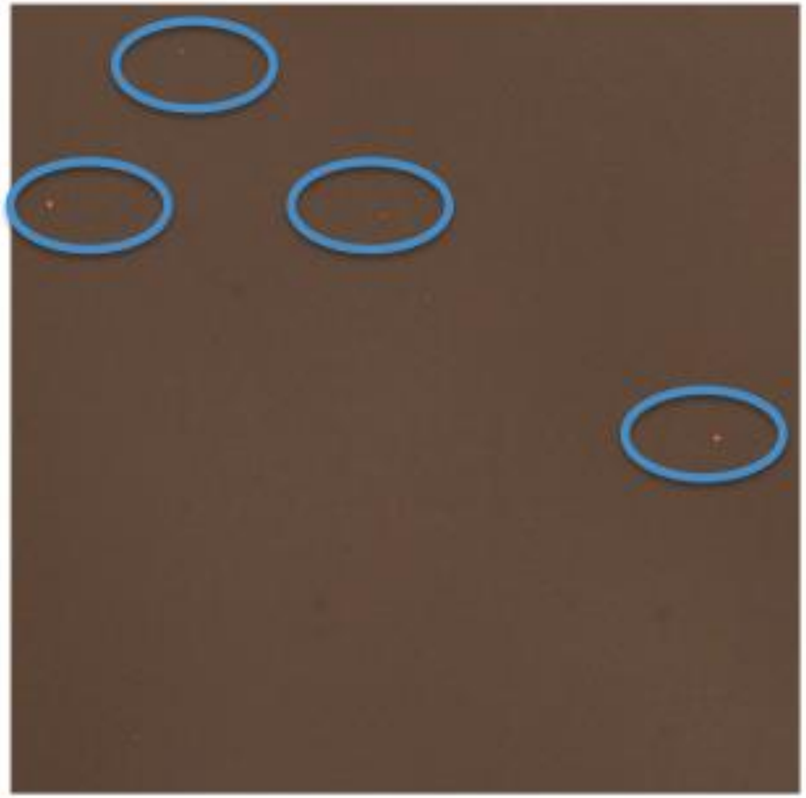
ISO gain and raw-image processing

Defective pixel mask

- CMOS have pixels that are defective
- Dead pixel masks are pre-calibrated at the factory
  - Using "dark current" calibration
  - Take an image with no light
  - Record locations reporting values to make "mask"
- Bad pixels in the mask are interpolated

ISO gain and raw-image processing

Defective pixel mask example



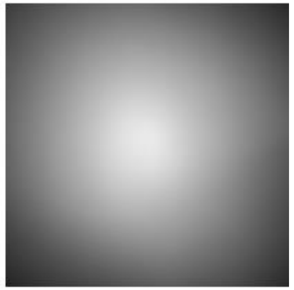
Identifying "dead pixels"

After interpolation

ISO gain and raw-image processing

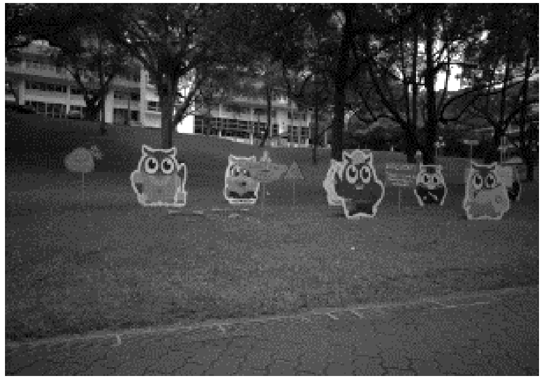
Image courtesy of Lu Zheng and Moshe Ben-Ezra

Flat-field correction

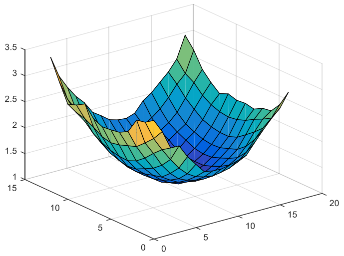


Uniform light falling on the sensor may not appear uniform in the raw-RGB image. This can be caused by the lens, sensor position in the camera housing, etc.

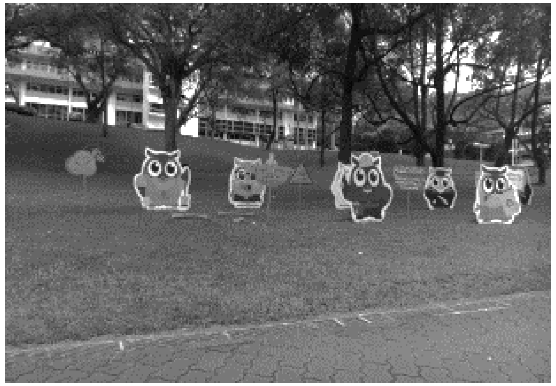
We want to correct this problem such that we get a "flat" output.



Before correction



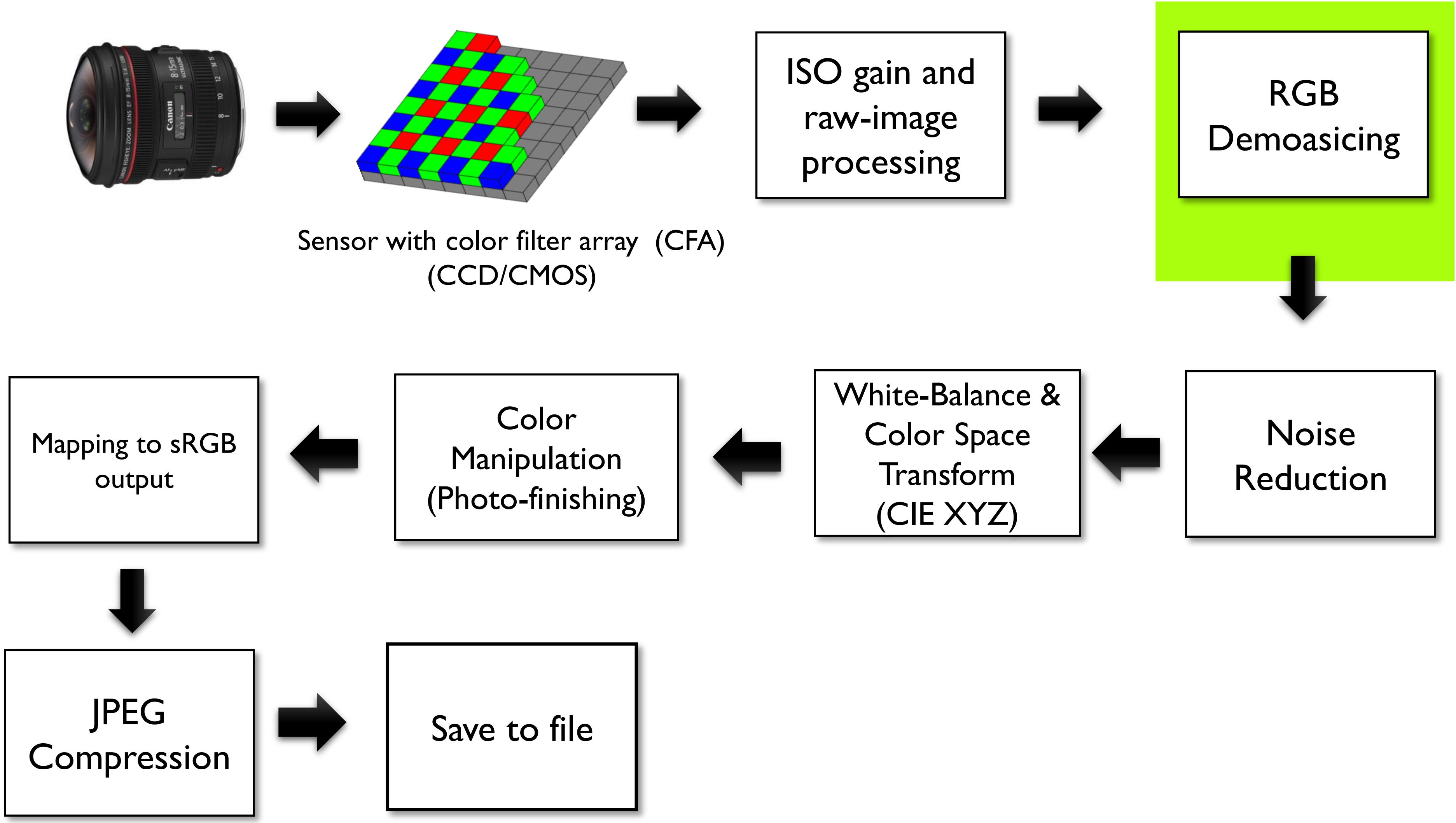
Apply a correction gain over the sensor values.



After correction

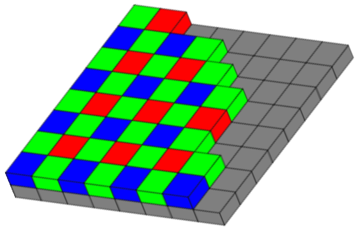
This operation can also be called "lens shading" correction.

A typical color imaging pipeline

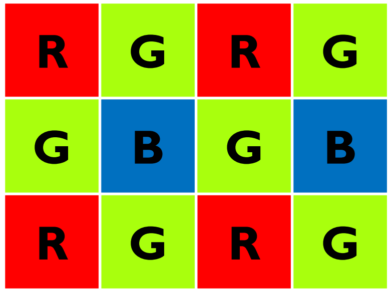


CFA/Bayer pattern demosaicing

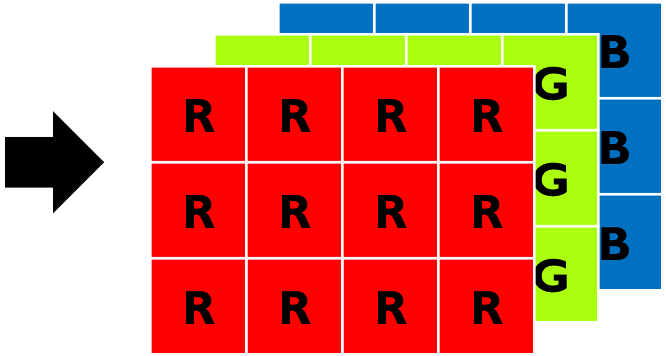
- Color filter array/Bayer pattern placed over pixel sensors
- We want an RGB value at each pixel, so we need to perform interpolation



Sensor with color filter array (CMOS)



Sensor RGB layout



Desired output with RGB per pixel.

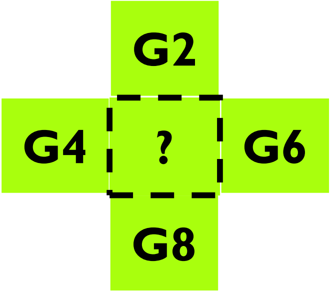
Simple interpolation

This is a zoomed up version of the Bayer pattern.

B1	G2	B3
G4	R5	G6
B7	G8	B9

At location R5, we have a red pixel value, but no Green or Blue pixel.

We need to estimate the G5 & B5 values at location R5.



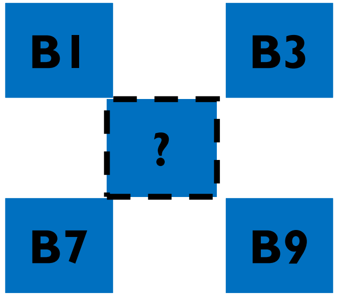
R5

G5 ?

B5 ?

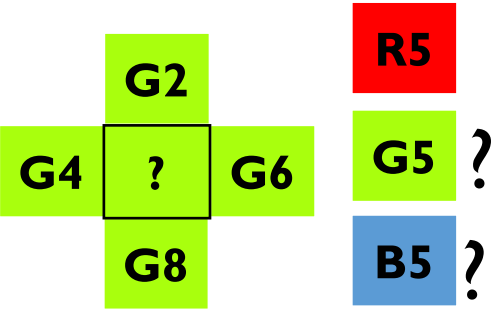
$$G5 = \frac{G2 + G4 + G6 + G8}{4}$$

$$B5 = \frac{B1 + B3 + B7 + B9}{4}$$



Simple "edge aware" interpolation

B1	G2	B3
G4	R5	G6
B7	G8	B9



If (|G2–G8) && |(G4-G8)| both < Thres):

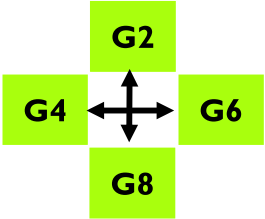
$$G5 = \frac{G2 + G4 + G6 + G8}{4}$$

elseif (|G2-G8| > Thres):

$$G5 = \frac{G4 + G6}{2}$$

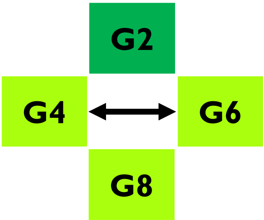
else:

$$G5 = \frac{G2 + G8}{2}$$



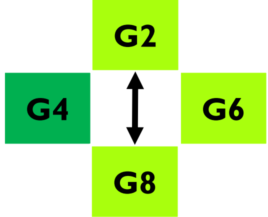
Case 1

All about the same.



Case 2

G2 and G8 differ – ignore them in the interpolation



Case 3

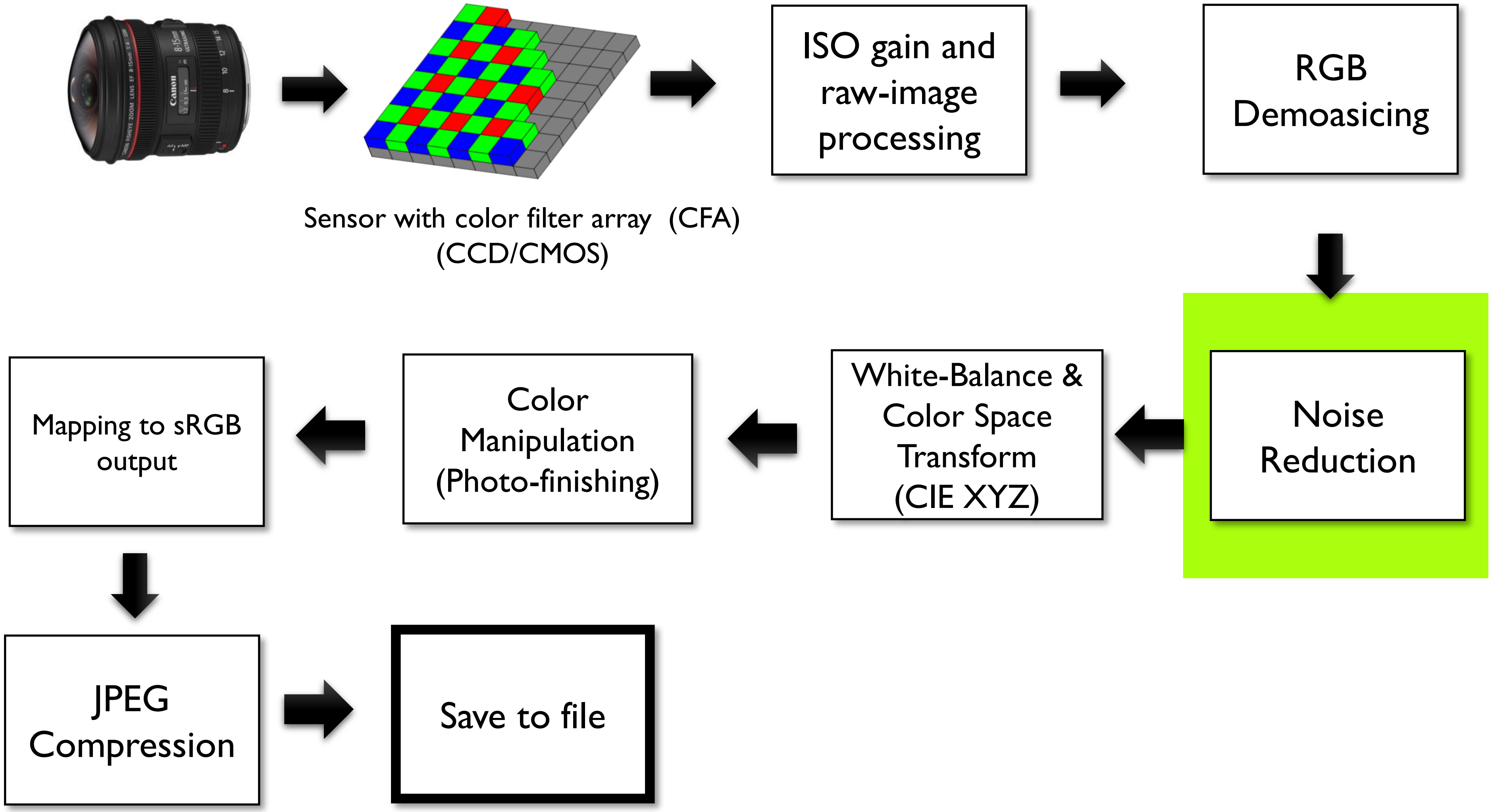
G2 and G8 differ – ignore them in the interpolation

Do this procedure also for the blue pixel, B5.

Demosaicing in practice

- The prior examples are illustrative algorithms only
- Camera IPSs use more complex and proprietary algorithms.
- Demosaicing can be combined with additional processing
  - Highlight clipping
  - Sharpening
  - Noise reduction

A typical color imaging pipeline



* Note that steps can be optional (e.g. noise reduction) or applied in slightly different order.

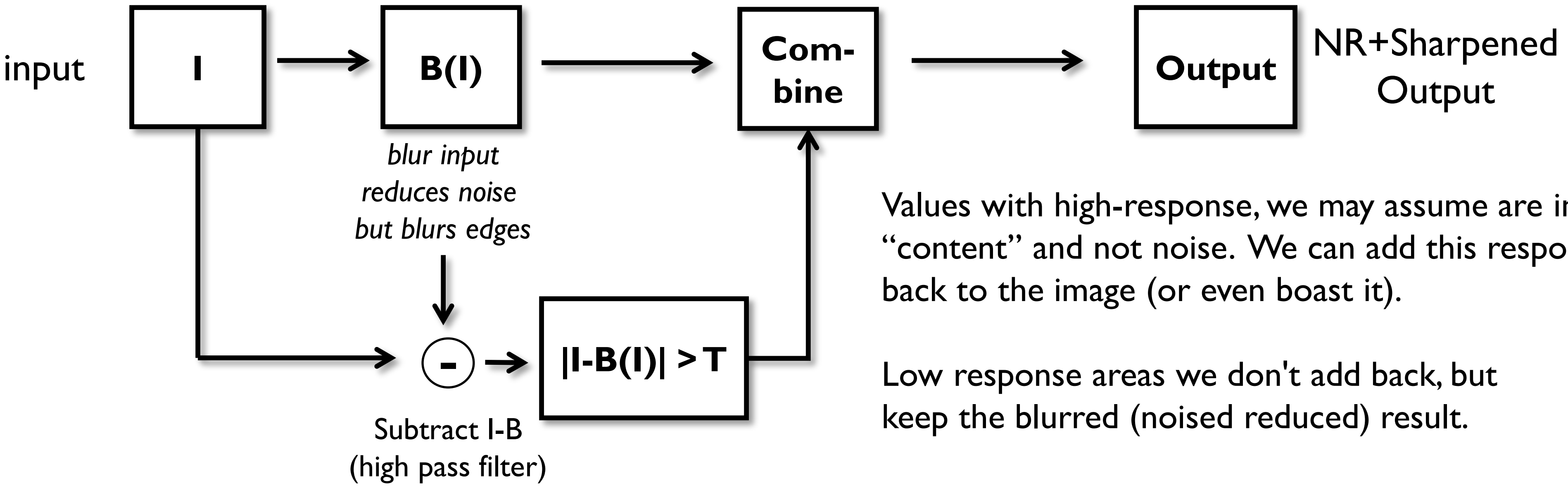
Noise reduction (NR)

- All sensors inherently have noise
- Most cameras apply additional NR after A/D conversion
- A simple method is described in the next slide
- For high-end cameras, it is likely that cameras apply different strategies depending on the ISO settings, e.g. high ISO will result in more noise, so a more aggressive NR could be used
- Smartphone cameras, because the sensor is small, apply aggressive noise reduction.

A simple noise reduction approach

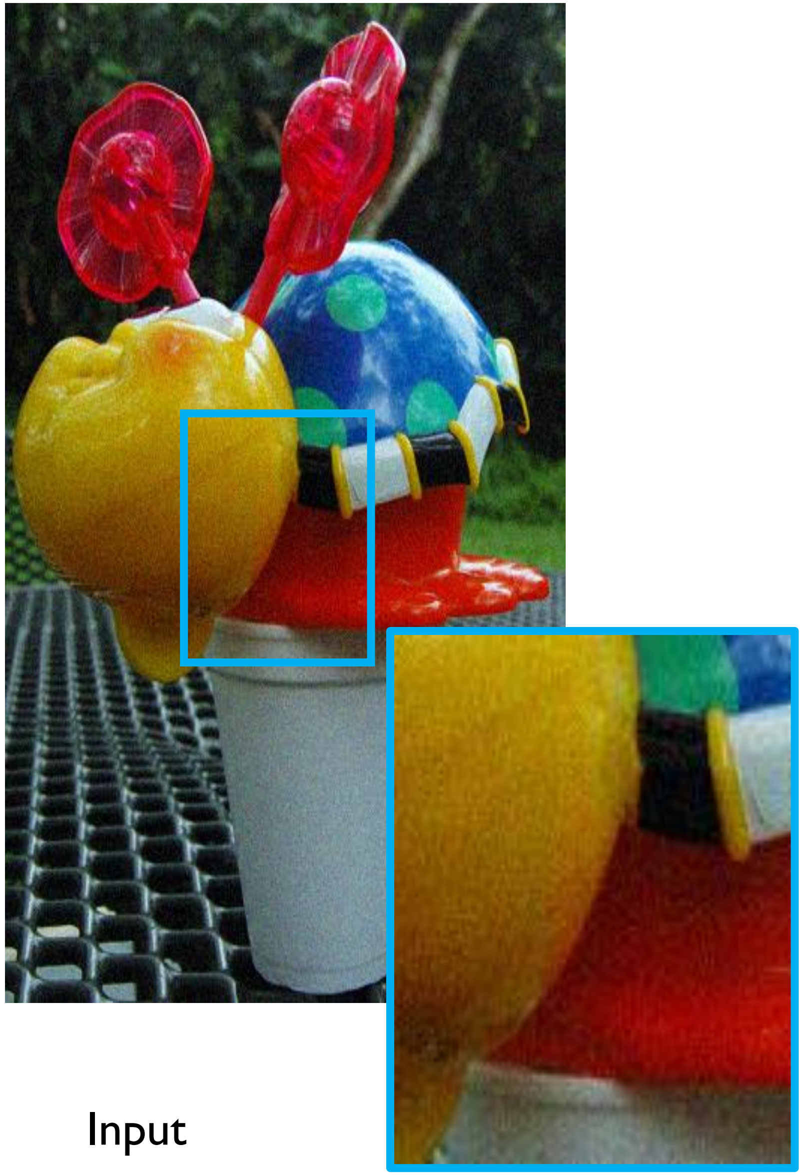
- Blur the image based on the ISO setting (higher ISO = more blur)
- Blurring will reduce noise, but also remove detail.
- Add image detail back for regions that have a high signal. We can even boost some parts of the signal to enhance detail (i.e. "sharpening")

Sketch of the procedure here

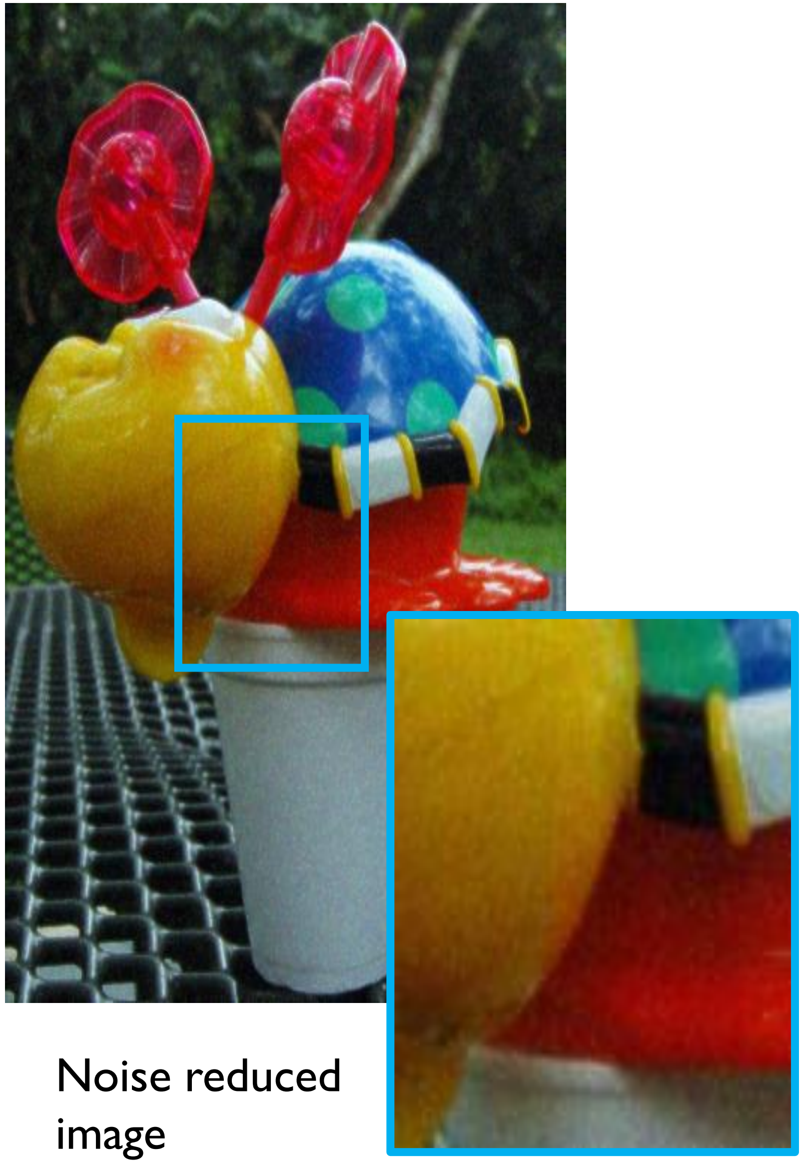


Values with high-response, we may assume are image “content” and not noise. We can add this response back to the image (or even boast it).

Low response areas we don't add back, but keep the blurred (noised reduced) result.

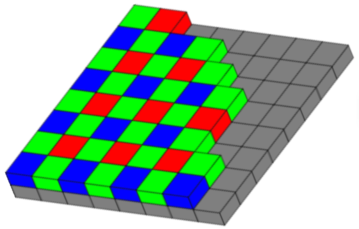


Input



Noise reduced image

A typical color imaging pipeline



Sensor with color filter array (CFA)
(CCD/CMOS)

ISO gain and raw-image processing

RGB Demoasicing

Noise Reduction

White-Balance & Color Space Transform (CIE XYZ)

Color Manipulation (Photo-finishing)

Mapping to sRGB output

JPEG Compression

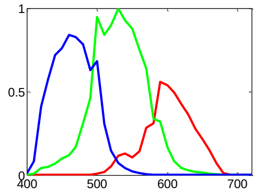
Save to file

Color mapping/colorimetric stage

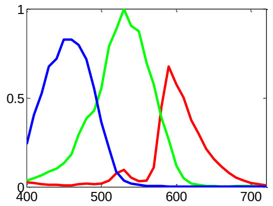
- This step in the IPS converts the sensor raw-RGB values to a device independent color space

Camera sensors have their own spectral response.

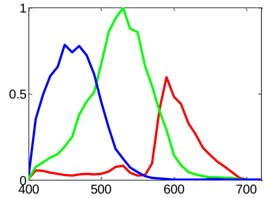
We need to map it into a standard response (CIE XYZ).



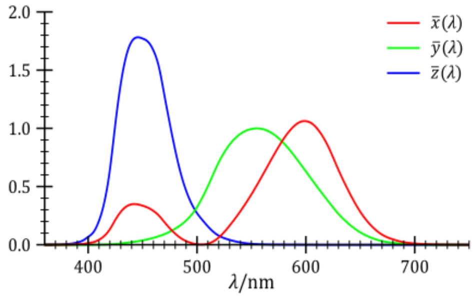
Canon 1D



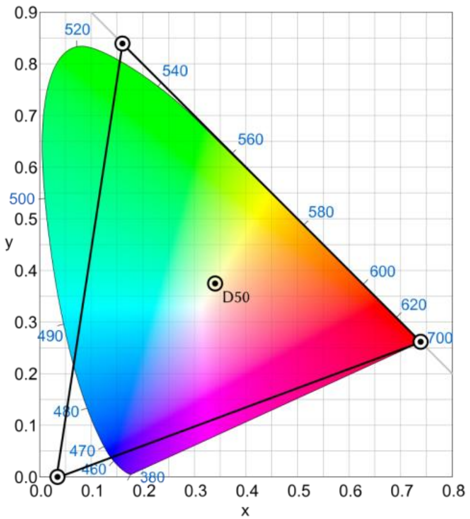
Nikon D40



Sony α57



CIE XYZ



ProPhoto RGB

We will use CIE XYZ in this tutorial, but most cameras use a related space called ProPhoto.

White-Balance & Color Space Transform (CIE XYZ)

Two step procedure

(1) apply a white-balance correction to the raw-RGB values

(2) map the white-balanced raw-RGB values to CIE XYZ

White balance

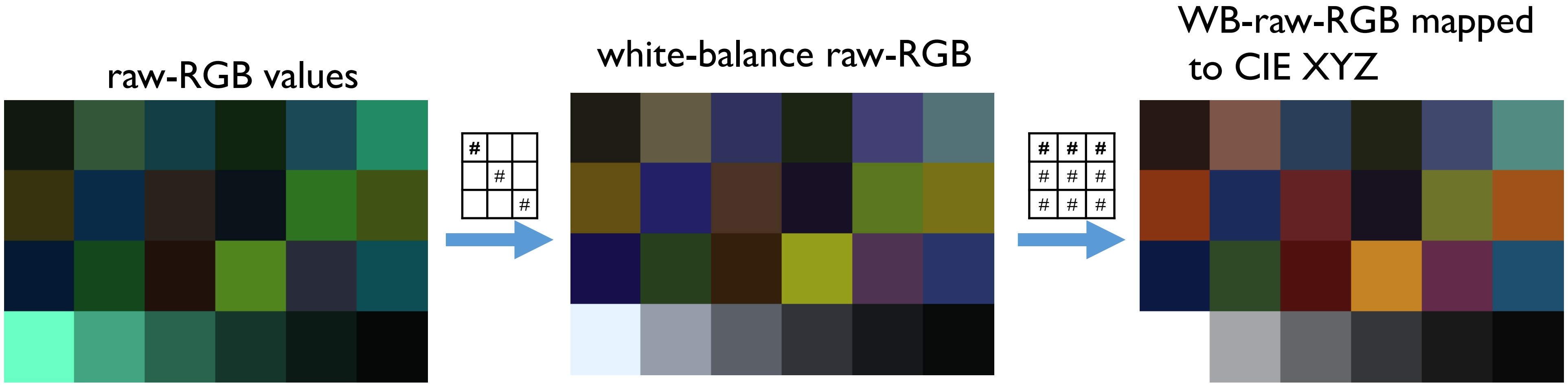
$$\begin{bmatrix} \# & & \\ & \# & \\ & & \# \end{bmatrix}$$

3x3 diagonal matrix

Color space transform (CST)

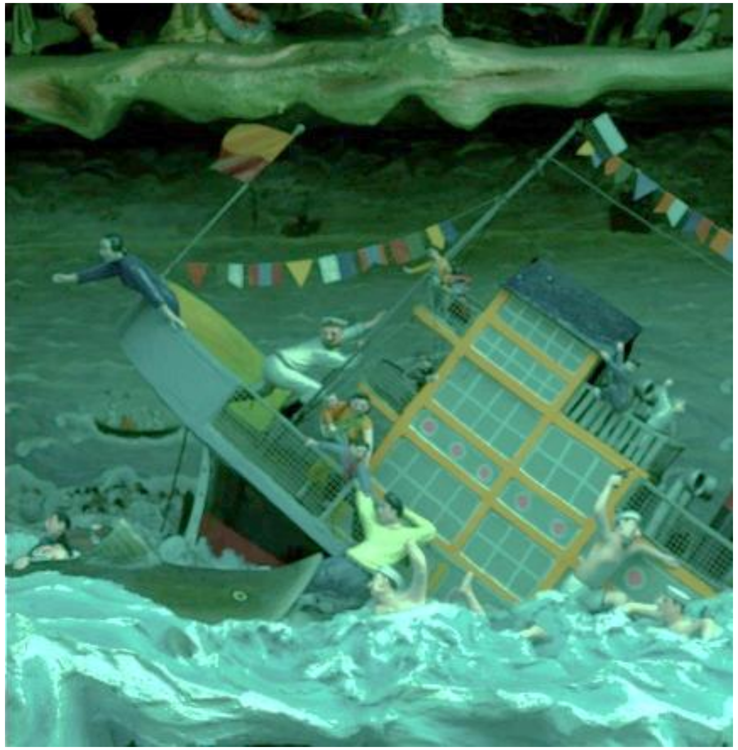
$$\begin{bmatrix} \# & \# & \# \\ \# & \# & \# \\ \# & \# & \# \end{bmatrix}$$

3x3 full matrix (or polynomial function)



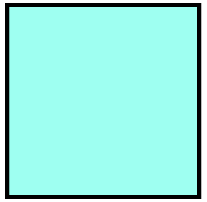
White-Balance & Color Space Transform (CIE XYZ)

How does white balance (WB) work?

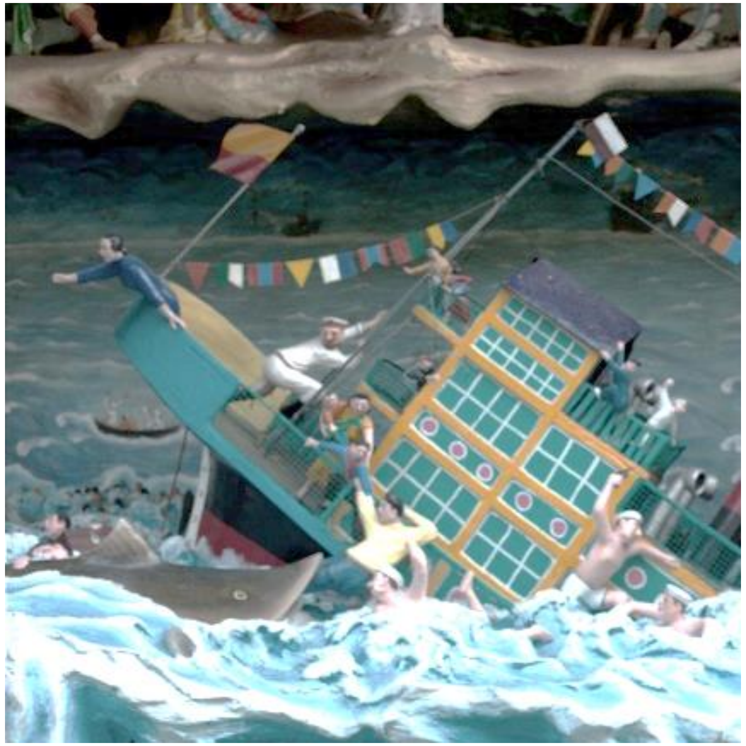


raw-RGB sensor image
(pre-white-balance correction)

Sensor's response to illumination (ℓ)



$$\begin{bmatrix} \ell_r \\ \ell_g \\ \ell_b \end{bmatrix} = \begin{bmatrix} 0.2 \\ 0.8 \\ 0.8 \end{bmatrix}$$



"White-balanced" raw-RGB image

White-balance diagonal matrix

$$\begin{bmatrix} r_{wb} \\ g_{wb} \\ b_{wb} \end{bmatrix} = \begin{bmatrix} 1/\ell_r & 0 & 0 \\ 0 & 1/\ell_g & 0 \\ 0 & 0 & 1/\ell_b \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix}$$

White-Balance & Color Space Transform (CIE XYZ)

White balance
(computational color constancy)

- **The challenging part for white-balance is determining the proper white-balance setting!**
- Users can manually set the white balance
  - Camera specific white-balance matrices for common illuminations
  - These can be manually selected by the user
- Otherwise auto white balance (AWB) is performed
  - In computer vision, we often refer to AWB as "illumination estimation"
  - Since the hard part is trying to determine what the illumination in the scene is.

White-Balance & Color Space Transform (CIE XYZ)

WB manual settings

WB SETTINGS	COLOR TEMPERATURE	LIGHT SOURCES
	10000 - 15000 K	Clear Blue Sky
	6500 - 8000 K	Cloudy Sky / Shade
	6000 - 7000 K	Noon Sunlight
	5500 - 6500 K	Average Daylight
	5000 - 5500 K	Electronic Flash
	4000 - 5000 K	Fluorescent Light
	3000 - 4000 K	Early AM / Late PM
	2500 - 3000 K	Domestic Lightning
	1000 - 2000 K	Candle Flame

Cameras can pre-calibrate their sensor's response for common illuminations.
Typical mapping of WB icons to related color temperate.

Image from ExposureGuide.com

Examples of manual WB matrices

Nikon D7000

Sunny

$$\begin{bmatrix} 2.0273 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.3906 \end{bmatrix}$$

Incandescent

$$\begin{bmatrix} 1.3047 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 2.2148 \end{bmatrix}$$

Shade

$$\begin{bmatrix} 2.4922 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.1367 \end{bmatrix}$$

Canon 1D

Daylight

$$\begin{bmatrix} 2.0938 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.5020 \end{bmatrix}$$

Tungsten

$$\begin{bmatrix} 1.4511 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 2.3487 \end{bmatrix}$$

Shade

$$\begin{bmatrix} 2.4628 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.2275 \end{bmatrix}$$

Sony A57K

Daylight

$$\begin{bmatrix} 2.6836 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.5586 \end{bmatrix}$$

Tungsten

$$\begin{bmatrix} 1.6523 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 2.7422 \end{bmatrix}$$

Shade

$$\begin{bmatrix} 3.1953 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.2891 \end{bmatrix}$$

White-Balance & Color Space Transform (CIE XYZ)

Pre-calibrated white-balance matrices for different brands of cameras.

Auto white balance (AWB)

- If manual white balance is not used, then an AWB algorithm is performed
- AWB needs to determine the sensor's raw-RGB response to the scene illumination from an arbitrary image
- This is surprisingly hard and AWB still fails from time to time (see next slide)

AWB is not easy

raw-RGB input image before white-balance

Given an arbitrary raw-RGB image, determine what is the camera's response to the illumination.

The idea is that something that is *white** is a natural reflector of the scene's illuminations SPD.

So, if we can identify what is "white" in the raw-RGB image, we are observing the sensor's RGB response to the illumination.

* It doesn’t have to be "white", but grey – sometimes we call these scene points "achromatic" or "neutral" regions.

AWB: "Gray world" algorithm

- This methods assumes that the average reflectance of a scene is achromatic (i.e. gray)
  - Gray is just the white point not at its brightest, so it serves as an estimate of the illuminant
  - This means that image average should have equal energy, i.e. R=G=B
- Based on this assumption, the algorithm adjusts the input average to be gray as follows:

First, estimate the average response:

$$R_{avg} = \frac{1}{Nr}\sum Rsensor(\mathrm{r}) \quad G_{avg} = \frac{1}{Ng}\sum Gsensor(\mathrm{g}) \quad B_{avg} = \frac{1}{Nb}\sum Bsensor(\mathrm{b})$$

r = red pixels values, g=green pixels values, b =blue pixels values
Nr = # of red pixels, Ng = # of green pixels, Nb = # blue pixels

Note: # of pixel per channel may be different if white balance is applied to the RAW image before demosaicing. Some pipelines may also transform into another colorspace, e.g. LMS, to perform the white-balance procedure.

AWB: "Gray world" algorithm

- Based on the image average R/G/B value, white balance can be expressed as a matrix as:

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} G_{avg}/R_{avg} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & G_{avg}/B_{avg} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

White-balanced sensor RGB

Sensor RGB

Matrix scales each channel by its average and then normalizes to the green channel average.

White-Balance & Color Space Transform (CIE XYZ)

AWB: "White patch" algorithm

- This methods assumes that "highlights" (bright spots) represent specular reflections of the illuminant
  - This means that maximum R, G, B values are a good estimate of the white point
- Based on this assumption, the algorithm works as follows:

$$R_{max} = max(Rsensor(\mathrm{r})) \qquad G_{max} = max(\, Gsensor(\mathrm{g})) \qquad B_{max} = max(\, Bsensor(\mathrm{b}))$$

r = red pixels values, g=green pixels values, b =blue pixels values

White-Balance & Color Space Transform (CIE XYZ)

AWB: "White patch" algorithm

- Based on RGB max, white balance can be expressed as a matrix as:

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} G_{max}/R_{max} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & G_{max}/B_{max} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

White-balanced sensor raw-RGB

Sensor raw-RGB

Matrix scales each channel by its maximum value and then normalizes to the green channel's maximum.

White-Balance & Color Space Transform (CIE XYZ)

AWB example

Input

Gray World

White Patch

White-Balance &
Color Space
Transform
(CIE XYZ)

Better AWB methods

- Gray world and white patch are *<u>very basic</u>* algorithms
  - These both tend to fail when the image is dominated by large regions of a single color (e.g. a sky image)
- **There are many AWB methods in the literature**
- Camera's often use their own proprietary white-balanced
- Note – they may not use the exact scene illumination, but a slightly different result to leave a small color cast in the image for aesthetic reasons.

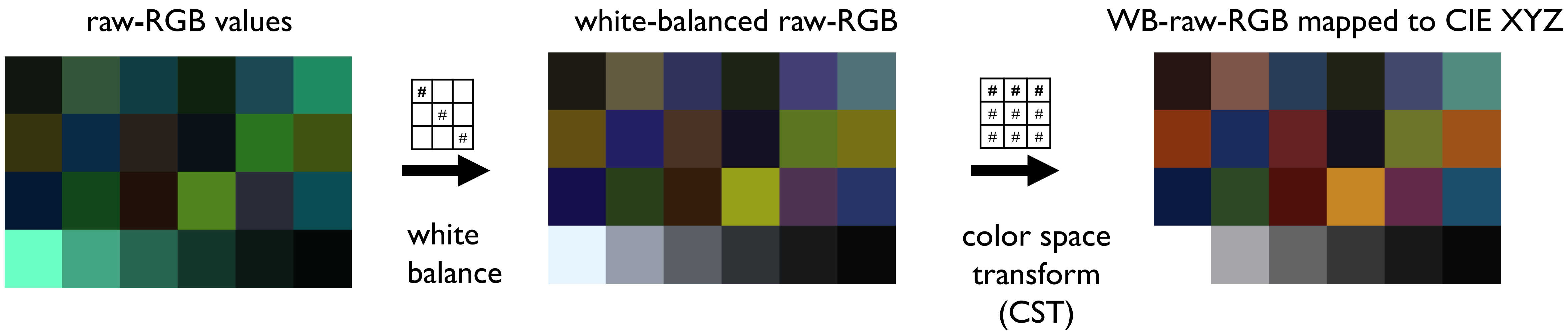
White-Balance & Color Space Transform (CIE XYZ)

Color space transform – part 2

- Process used on cameras involves interpolation from factory presets

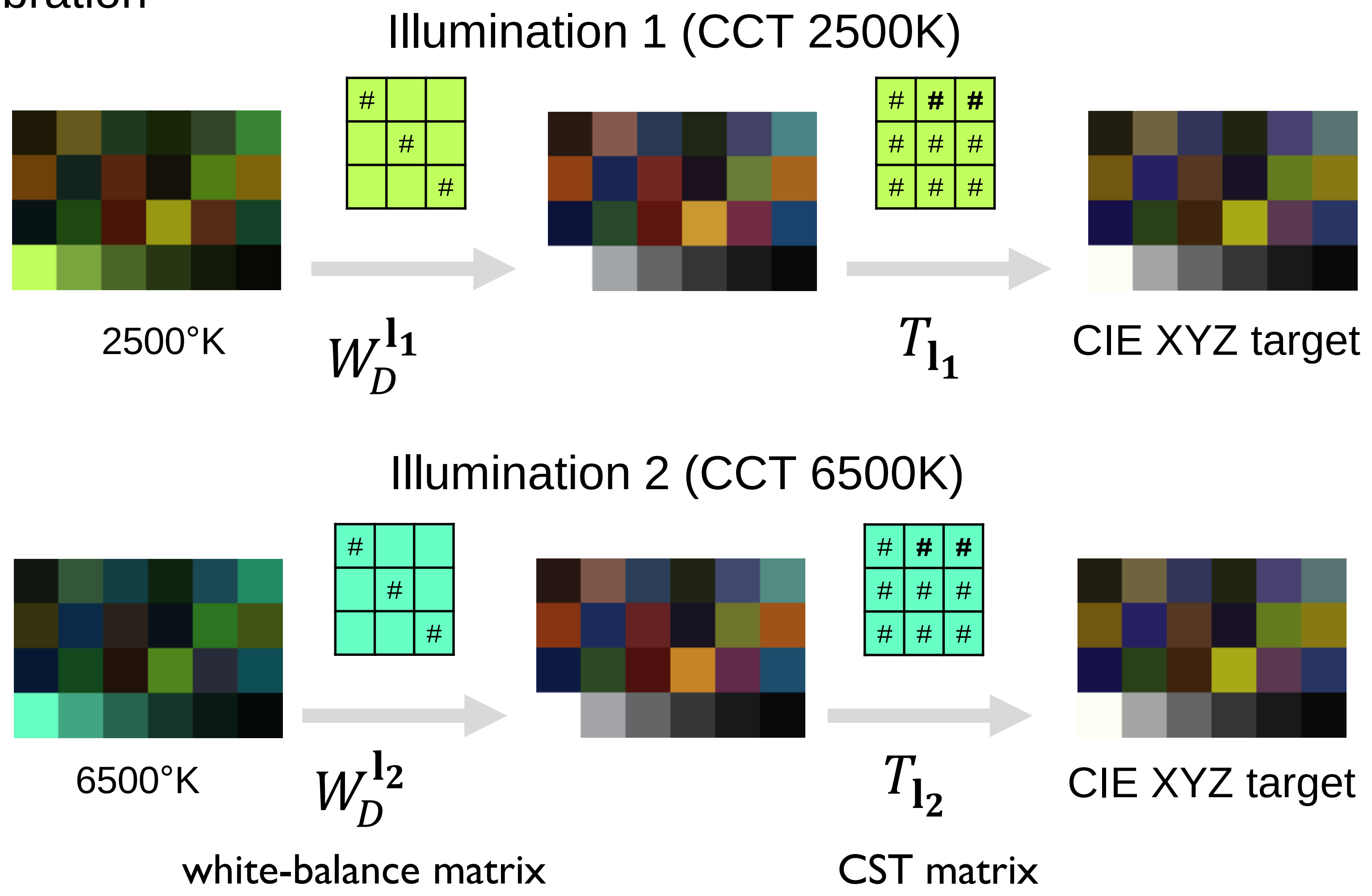
- **The need for interpolation is relate to white-balance only approximating true color constancy**

Color space transform is applied after the white balance. In fact, the matrix we use to perform the CST is based on the white-balance CCT.



Color space transform (1/3)

Factory pre-calibration

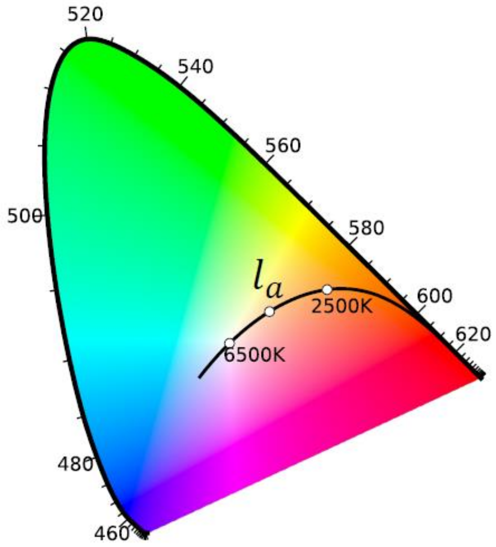


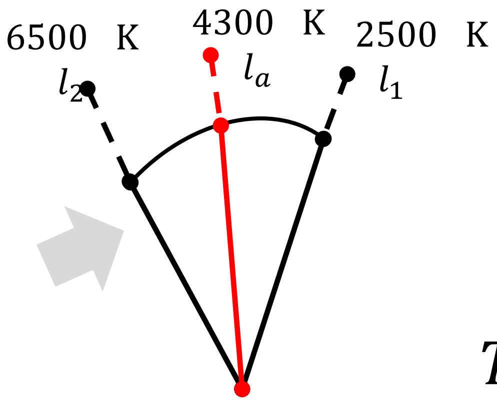
White-Balance & Color Space Transform (CIE XYZ)

CST matrices ($T_{\mathbf{l}_1}$ and $T_{\mathbf{l}_2}$) are calibrated for two different illuminations (l1 and l2). Depending on the temperature of the white-balance, we use the corresponding CST.

Color space transform (2/3)

Interpolation process





$$g = \frac{\mathrm{CCT}_{\mathbf{l_a}}^{-1} - \mathrm{CCT}_{\mathbf{l_2}}^{-1}}{\mathrm{CCT}_{\mathbf{l_1}}^{-1} - \mathrm{CCT}_{\mathbf{l_2}}^{-1}}$$

$$T_{\mathbf{l_a}} = gT_{\mathbf{l_1}} + (1-g)T_{\mathbf{l_2}}$$

New input (4300°K) $\Phi_{\mathbf{cam}}^{\mathbf{l}_a}$

$W_D^{\mathbf{l_a}}$

$T_{\mathbf{l_a}}$

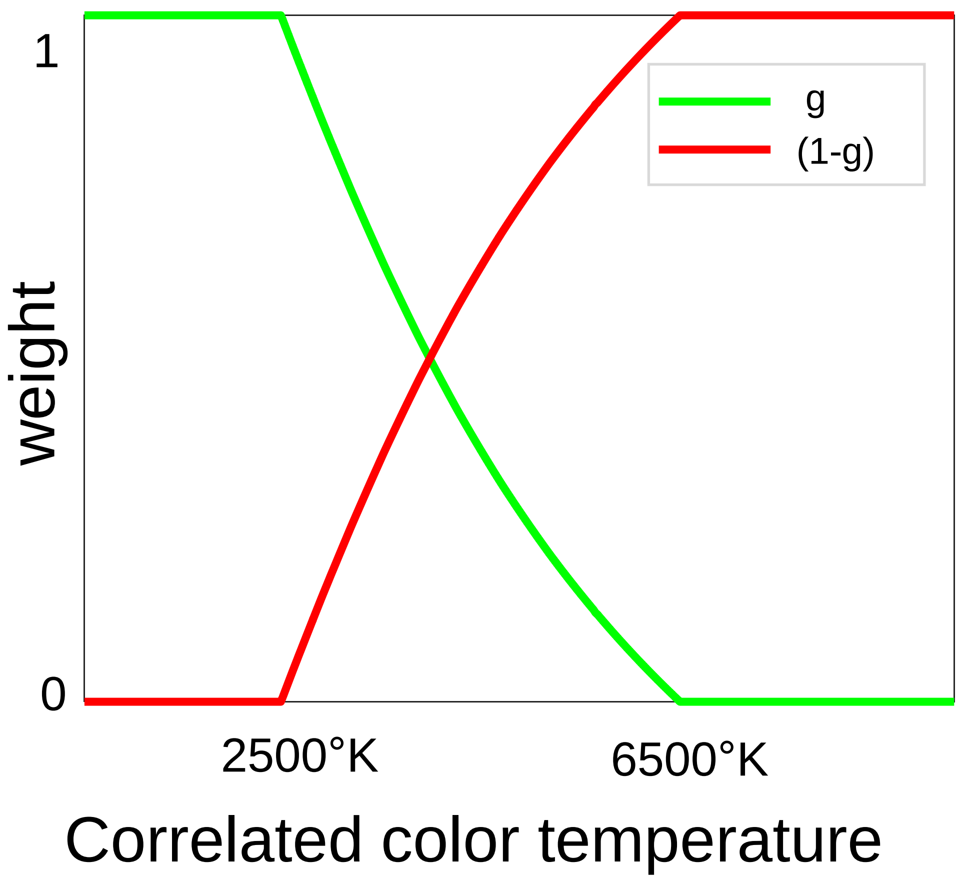
Given a new illumination (la) and its estimated correlated color temperature (CCT), we construct a CST matrix by blending the two factory pre-calibrated matrices.

Color space transform (3/3)

Weighting functions

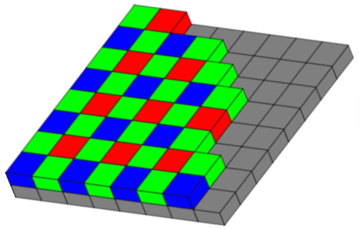
$$g = \frac{\mathrm{CCT}_{\mathbf{l_a}}^{-1} - \mathrm{CCT}_{\mathbf{l_2}}^{-1}}{\mathrm{CCT}_{\mathbf{l_1}}^{-1} - \mathrm{CCT}_{\mathbf{l_2}}^{-1}}$$

$$T_{\mathbf{l_a}} = gT_{\mathbf{l_1}} + (1-g)T_{\mathbf{l_2}}$$



White-Balance & Color Space Transform (CIE XYZ)

Typical color imaging pipeline



Sensor with color filter array (CFA)
(CCD/CMOS)

ISO gain and raw-image processing

RGB Demoasicing

Noise Reduction

White-Balance & Color Space Transform (CIE XYZ)

Color Manipulation (Photo-finishing)

Mapping to sRGB output

JPEG Compression

Save to file

Color manipulation

- This is the stage were a camera applies its "secret sauce" to make the images look good
- This procedure can be called by many names:
  - Color manipulation
  - Photo-finishing
  - Color rendering or selective color rendering
  - Yuv processing engine
- DSLR will often allow the user to select various photo-finishing styles
- Smartphones often compute this per-image
- Photo-finishing may also tied to geographical regions!

DSLR "picture" styles

From Canon’s user manual

Standard

Glowing prints with crisp finishes.
It is the basic color of EOS DIGITAL.

Portrait

For transparent, healthy skin for women and children

Landscape

Crisp and impressive reproduction of blue skies and green trees in deep, vivid color

Neutral

Subjects are recorded in rich detail, giving the greatest latitude for image processing

Faithful

Accurate recording of the subject’s color, close to the actual image seen with the naked eye

Monochrome

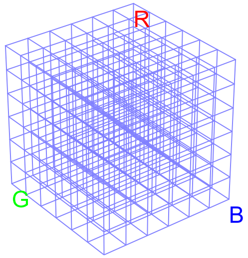
Filter work and sepia tone with the freedom of digital monochrome

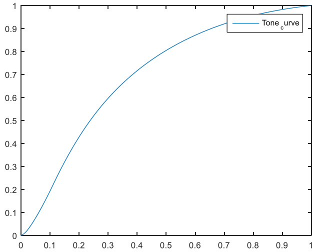
Picture styles

Example of four different picture styles from Nikon
This image is the **same** raw-RGB image processed in four different ways.

Color
Manipulation
(Photo-finishing)

Nonlinear color manipulation





3D Look up table (LUT)

1D Tone Curve

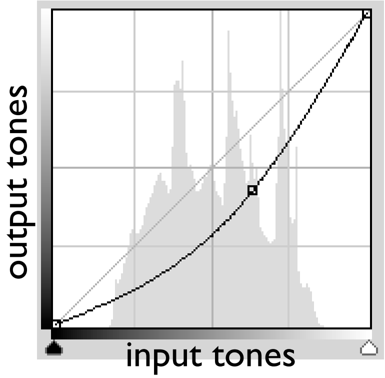
Color manipulation can be implemented using a 3D look up table (LUT) and a 1D LUT tone-curve.
The 3D LUT table acts like a 3D function: $f(X, Y, Z) \rightarrow X', Y', Z'$
The 1D LUT table is applied per channel: $g(X) \rightarrow X', g(Y) \rightarrow Y', g(Z) \rightarrow Z'$

The 3D and 1D LUT can change based on picture style.

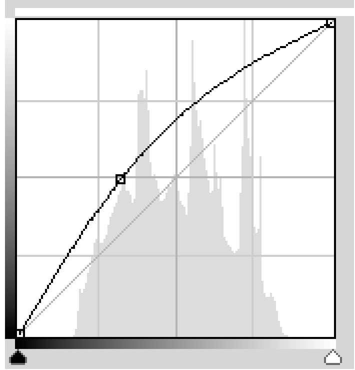
Color Manipulation (Photo-finishing)

Global tone map example (1D LUT)

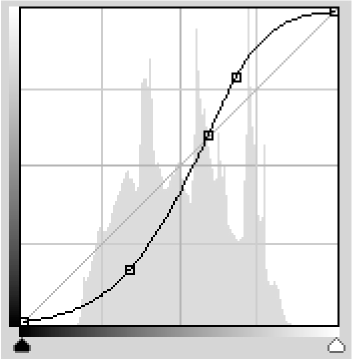
Input



Darkening the image

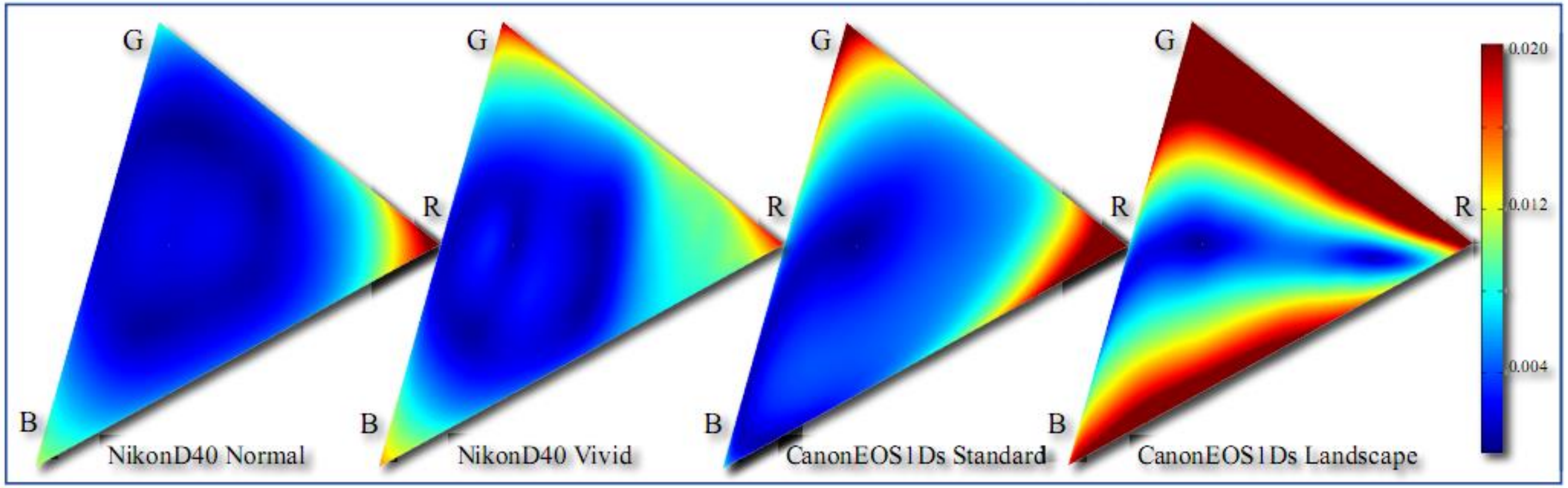


Brightening the image

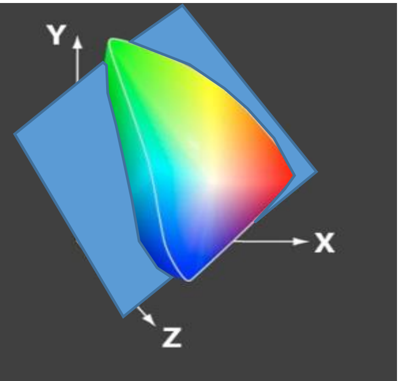


Enhancing contrast (called an S-curve)

3D LUT color manipulation visualization



Visualization as a **displacement map** of a *slice*
of the 3D LUT mapping, warping an input and output value

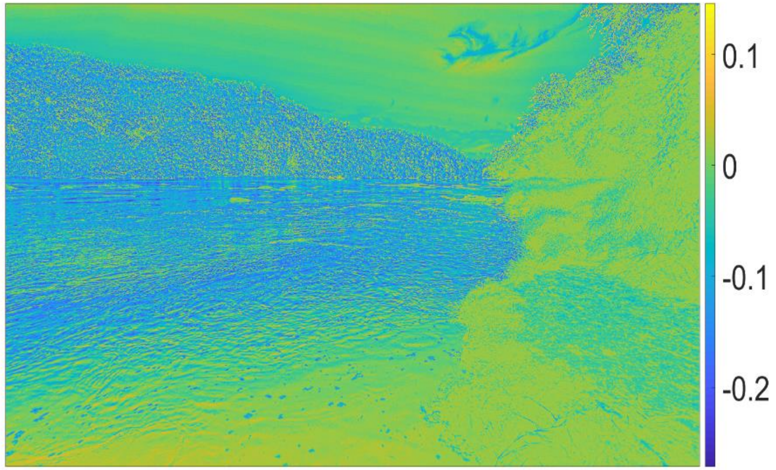


Local tone mapping (LTM)

Global tone-mapping
Camera mode - Manual

Local tone-mapping
Camera mode - Auto

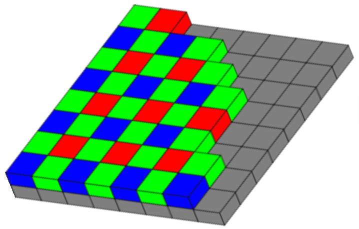
NOTE: On many cameras, esp smartphones, a local tone map is applied as part of the photo-finishing. This helps bring out highlights in the image.



Difference map between image before and after LTM

Color Manipulation (Photo-finishing)

Typical color imaging pipeline



Sensor with color filter array (CFA)
(CCD/CMOS)

→ ISO gain and raw-image processing → RGB Demoasicing → Noise Reduction → White-Balance & Color Space Transform (CIE XYZ) → Color Manipulation (Photo-finishing) → Mapping to sRGB output → JPEG Compression → Save to file

Final sRGB conversion

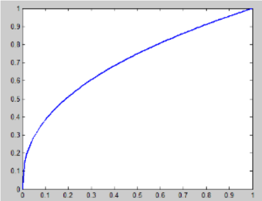
- Map from *photo-finished* CIE XYZ image to sRGB
- Apply the sRGB $(2.2)^{-1}$ gamma encoding

Photo-finished CIE XYZ

Covert to linear sRGB

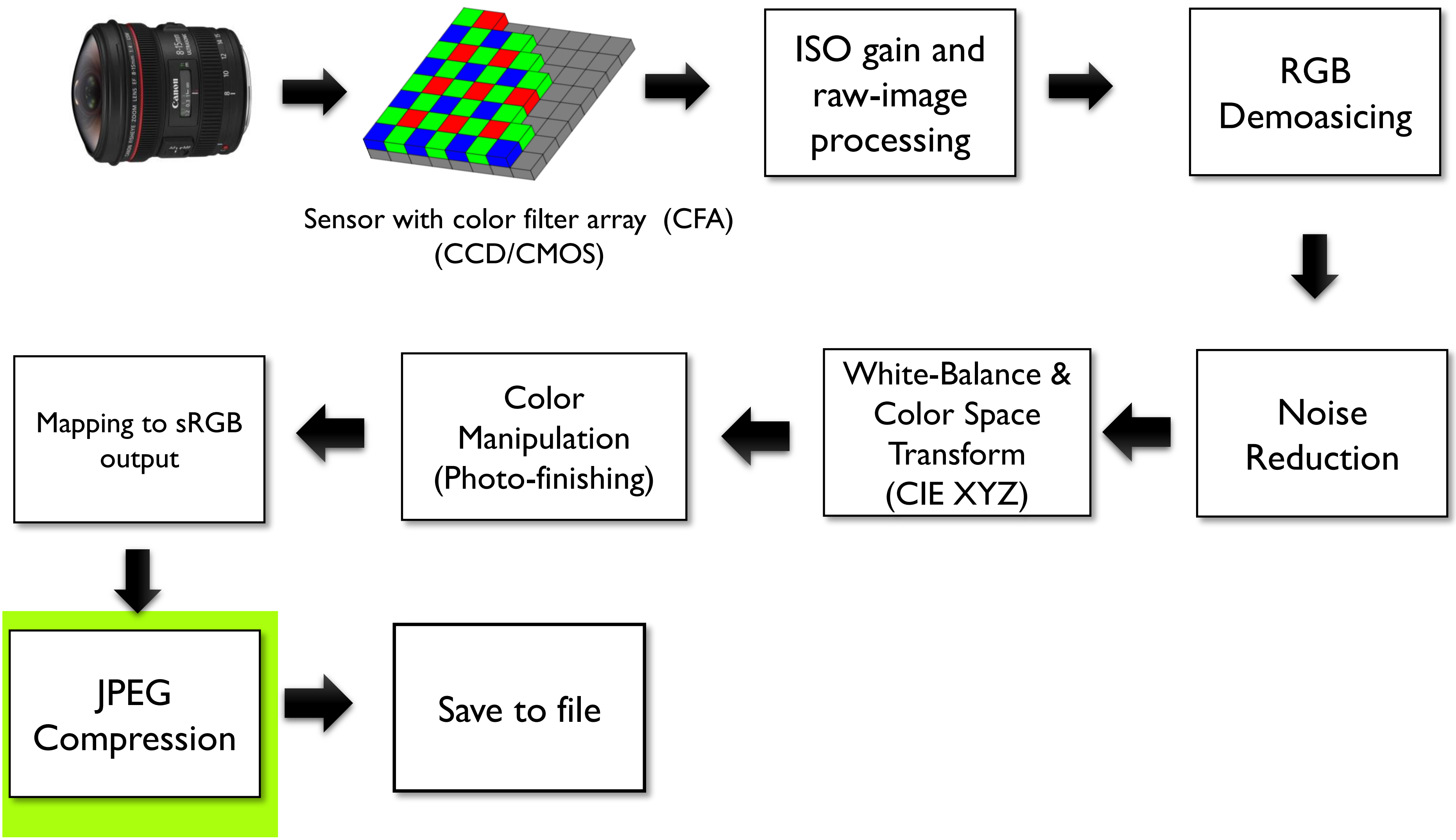
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.2404542 & -1.5371385 & 0.4985314 \\ -0.9692660 & 1.8760108 & 0.0415560 \\ 0.0556434 & -0.2040259 & 1.0572252 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Apply sRGB gamma

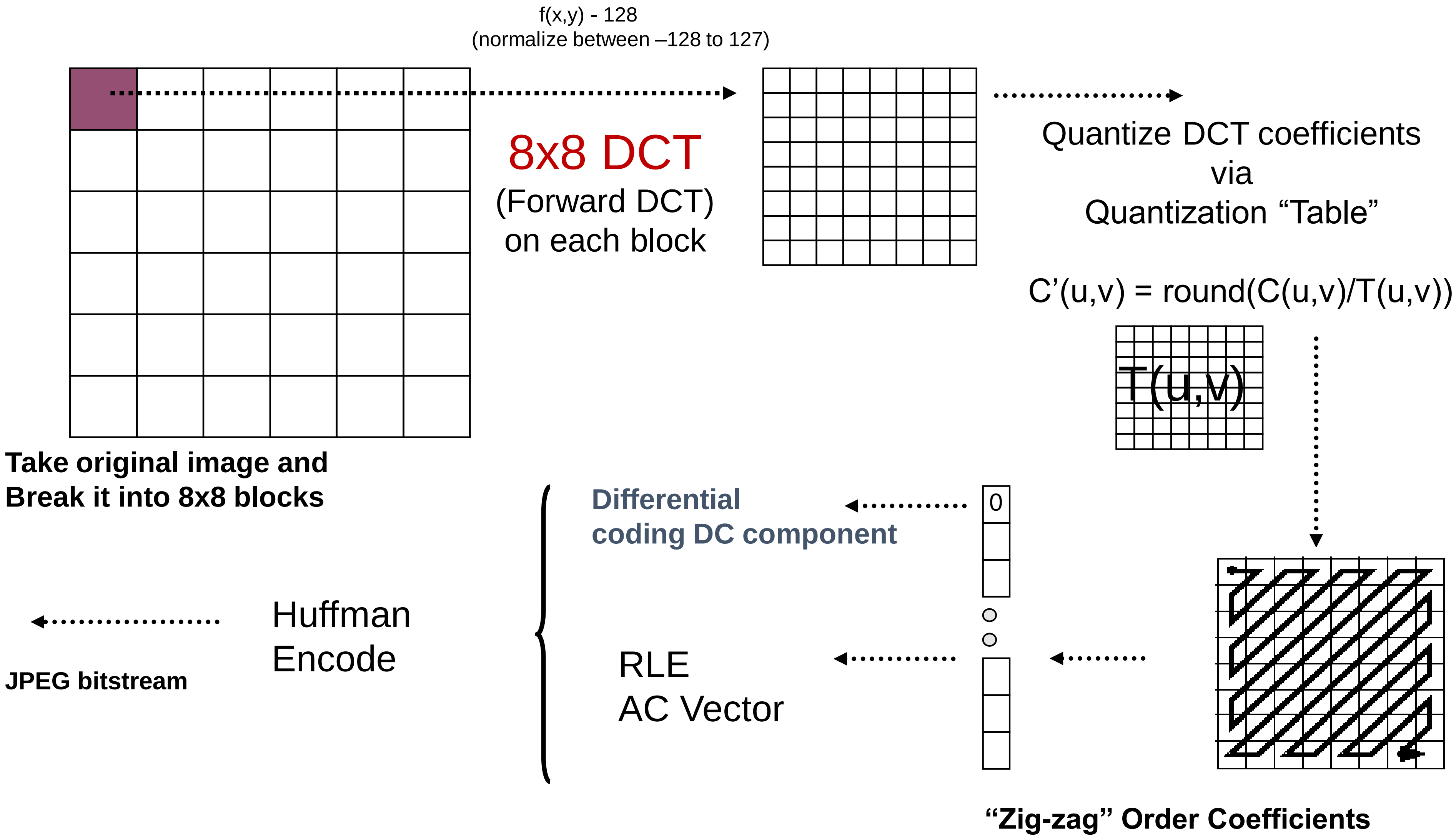


Map to sRGB output

Typical color imaging pipeline



JPEG compression scheme



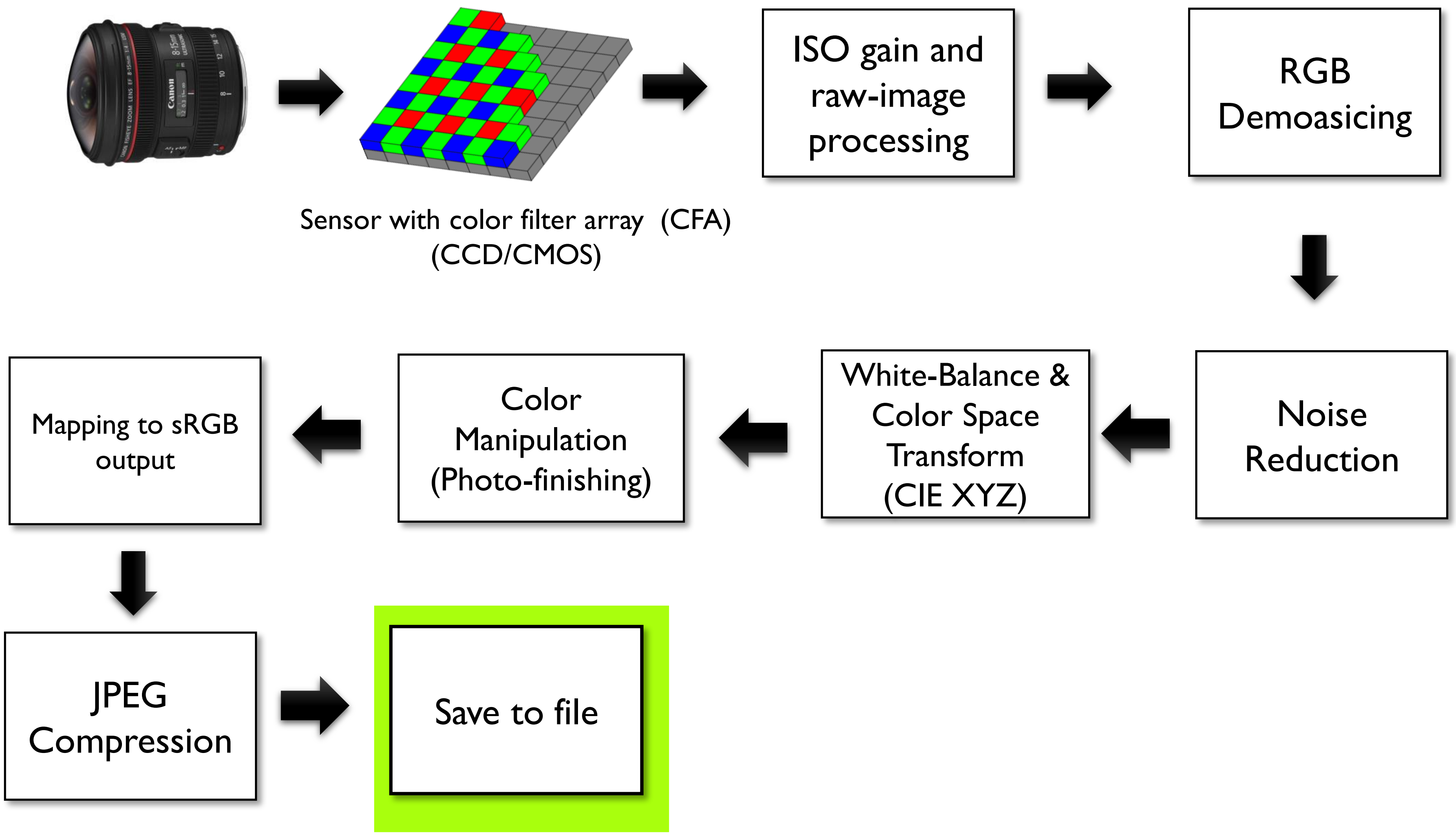
JPEG applies almost every compression trick known.
1) Transform coding, 2) psychovisual (loss), 3) Run-length-encoding (RLE), 4) Difference coding, and Huffman.

JPEG quality

- The amount of quantization applied on the DCT coefficients amounts to a “quality” factor
  - More quantization = better compression (smaller file size)
  - More quantization = lower quality
- Cameras generally allow a range that you can select

Image from nphotomag.com

Save to storage and we are done!



Exif metadata

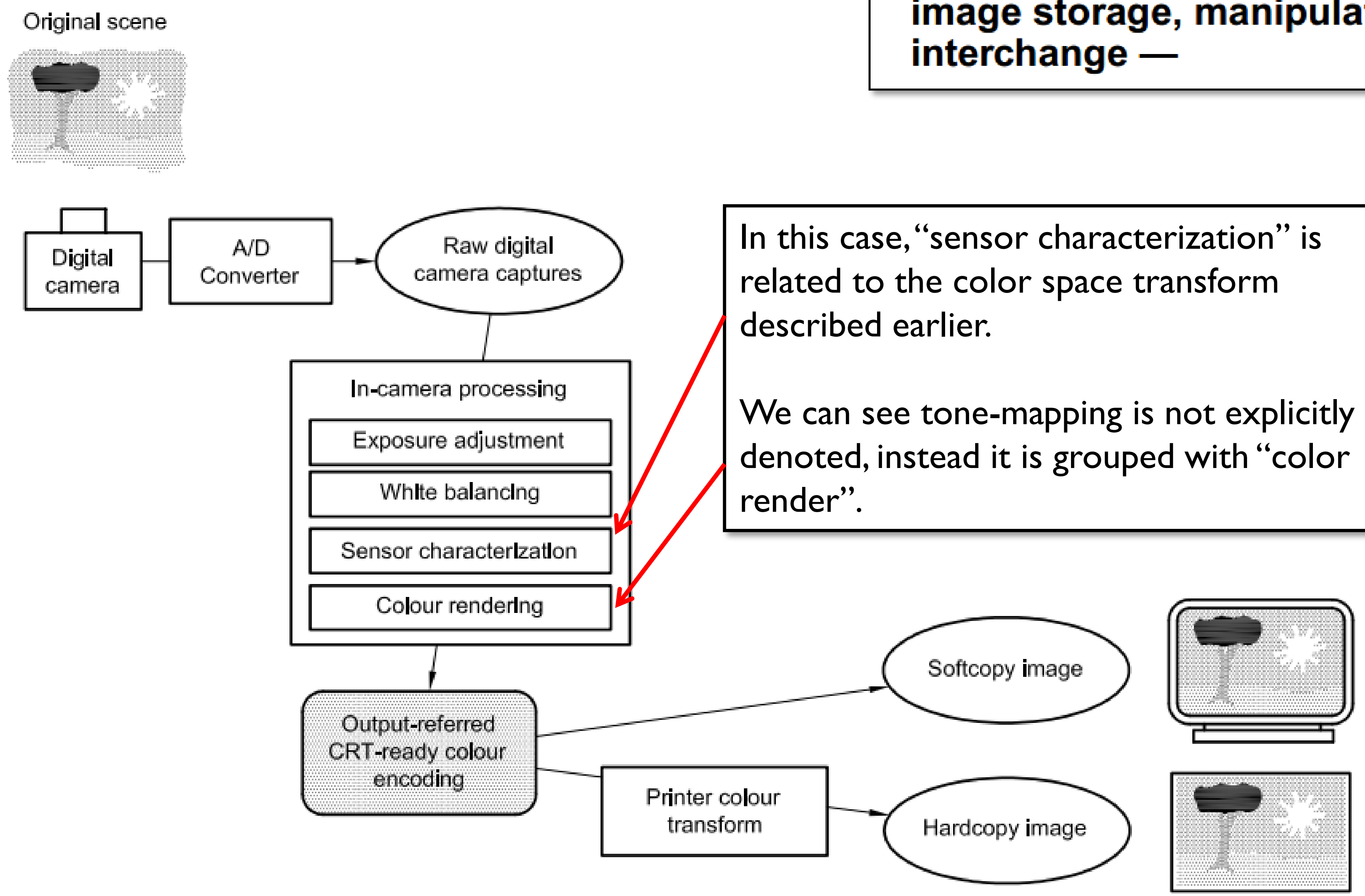
- Exchangeable image file format (Exif)
- Created by the Japan Electronics and Information Technology Industries Association (JEITA)
- Associates meta data with images
  - Date/time
  - Camera settings (basic)
    - Image size, aperture, shutter speed, focal length, ISO speed, metering mode (how exposure was estimated)
  - Additional info (from in some Exif files)
    - White-balance settings, even matrix coefficients of white-balnace
    - Picture style (e.g. landscape, vivid, standard, portrait)
    - Output color space (e.g. sRGB, Adobe RGB, RAW)
    - GPS info
    - More . . .

ICC and color profiles

- International Color Consortium (ICC)
  - In charge of developing several ISO standards for color management
- Promote the use of ICC profiles
- ICC profiles are intended for device manufacturers to describe how their respective color spaces (e.g. sensor RGB) map to canonical color spaces called Profile Connection Spaces (PCS)
- PCS are similar to linking all devices to CIE XYZ, but are more flexible allowing for additional spaces to be defined (beyond CIE XYZ)

From the ICC-ISO 22028

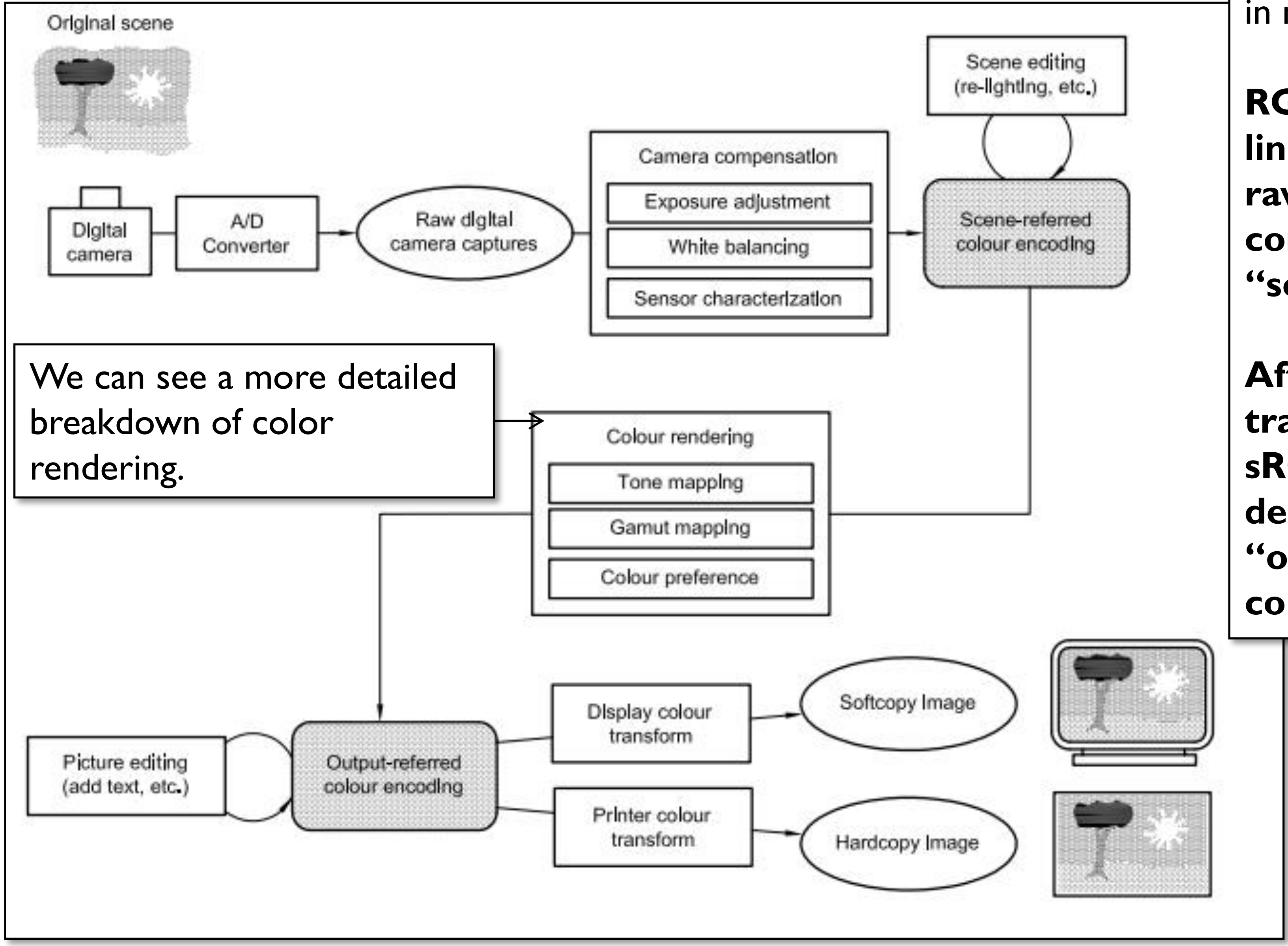
Photography and graphic technology — Extended colour encodings for digital image storage, manipulation and interchange —



In this case, "sensor characterization" is related to the color space transform described earlier.

We can see tone-mapping is not explicitly denoted, instead it is grouped with "color render".

From the ICC-ISO 22028



We can see a more detailed breakdown of color rendering.

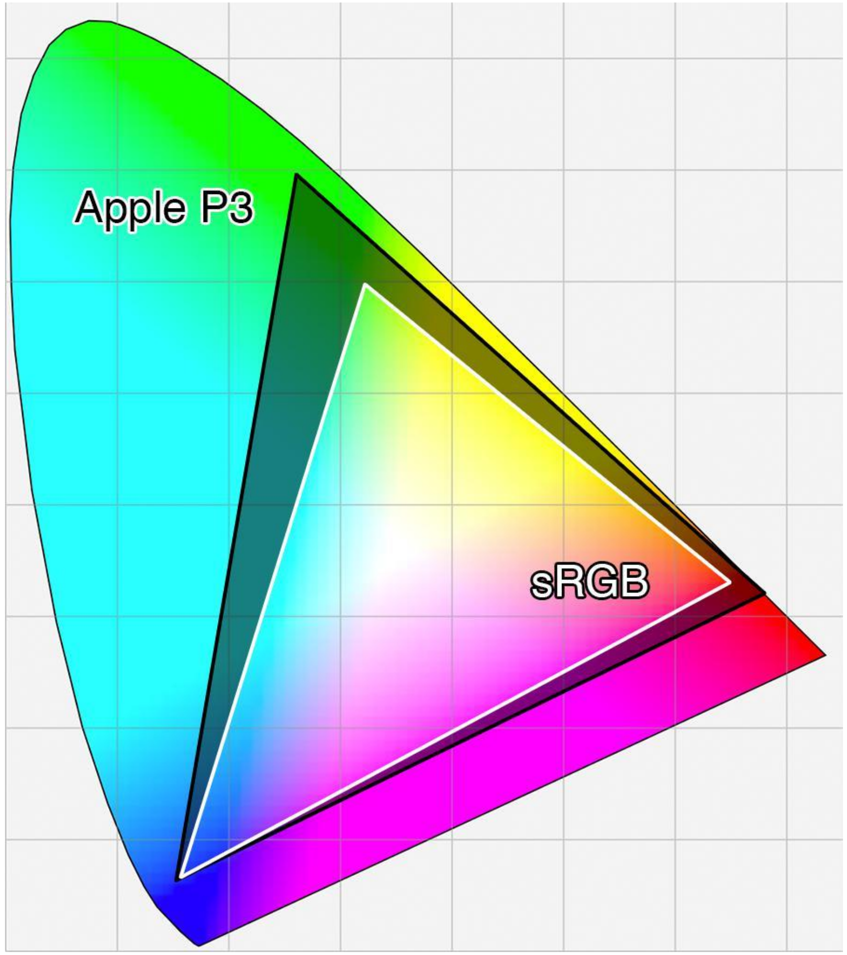
This describes a basic digital camera pipeline in more detail.

RGB values linked to the raw-RGB are considered "scene referred".

After the color transform to sRGB they are denoted as "output referred" color encodings.

Note: sRGB/JPEG is slowly being replaced

- sRGB was developed for monitors in the 1990s – it is an old standard.
- High Efficient Image Encoding (HEIC)
  - Better compression than JPEG
- Apple iPhone has started to use HEIC to *replace JPEG*
- HEIC supports multiple color spaces. Apple uses Display P3 – a variation on a Digitial Cinema Initiative P3 space.
- The P3 gamut is 25% wider than sRGB
- There is also a gamma encoding similar to sRGB.
- Pixel 4 and other Android devices are will support this color space soon.

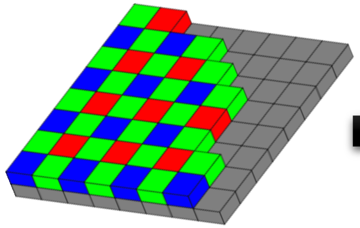


Pipeline comments

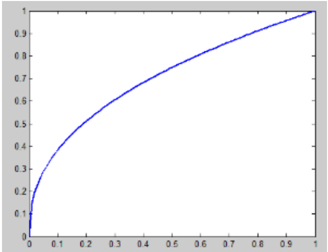
- Again, important to stress that the exact steps mentioned in these notes only serve as a guide of what takes place in a camera
- Modern pipelines are more complex, however, you will find steps similar to what was described
- Note: for different camera makes/models, the operations could be performed in different order (e.g. white-balance after demosaicing) and in different ways (e.g. combining sharpening with demosaicing)

What about machine vision cameras?

- Some industrial/machine vision cameras provide minimal ISP processing
- For example, some will only perform white-balance and apply a gamma to the raw-RGB values.
- This means the output is in a camera-specific color space



→ ISO gain and raw-image processing → RGB Demoasicing → White-Balance (per-channel gain) →



per-channel gamma

Typical machine vision pipeline

Point grey grasshopper camera.

Congratulations!

Tutorial schedule

- Part 1 (General)
  - ~~Motivation~~
  - ~~Review of color & color spaces~~
  - ~~Overview of in-camera imaging pipeline~~

1.30pm – 3.30pm

Break
3.30pm – 4.30pm

- Part 2 (Imaging and Computer Vision)
  - Misconceptions in the computer vision community regarding color
  - Recent work on color and cameras
  - Concluding remarks

4.30pm – 6.00pm

Part 2: Mistakes and misconceptions in the computer vision community regarding color

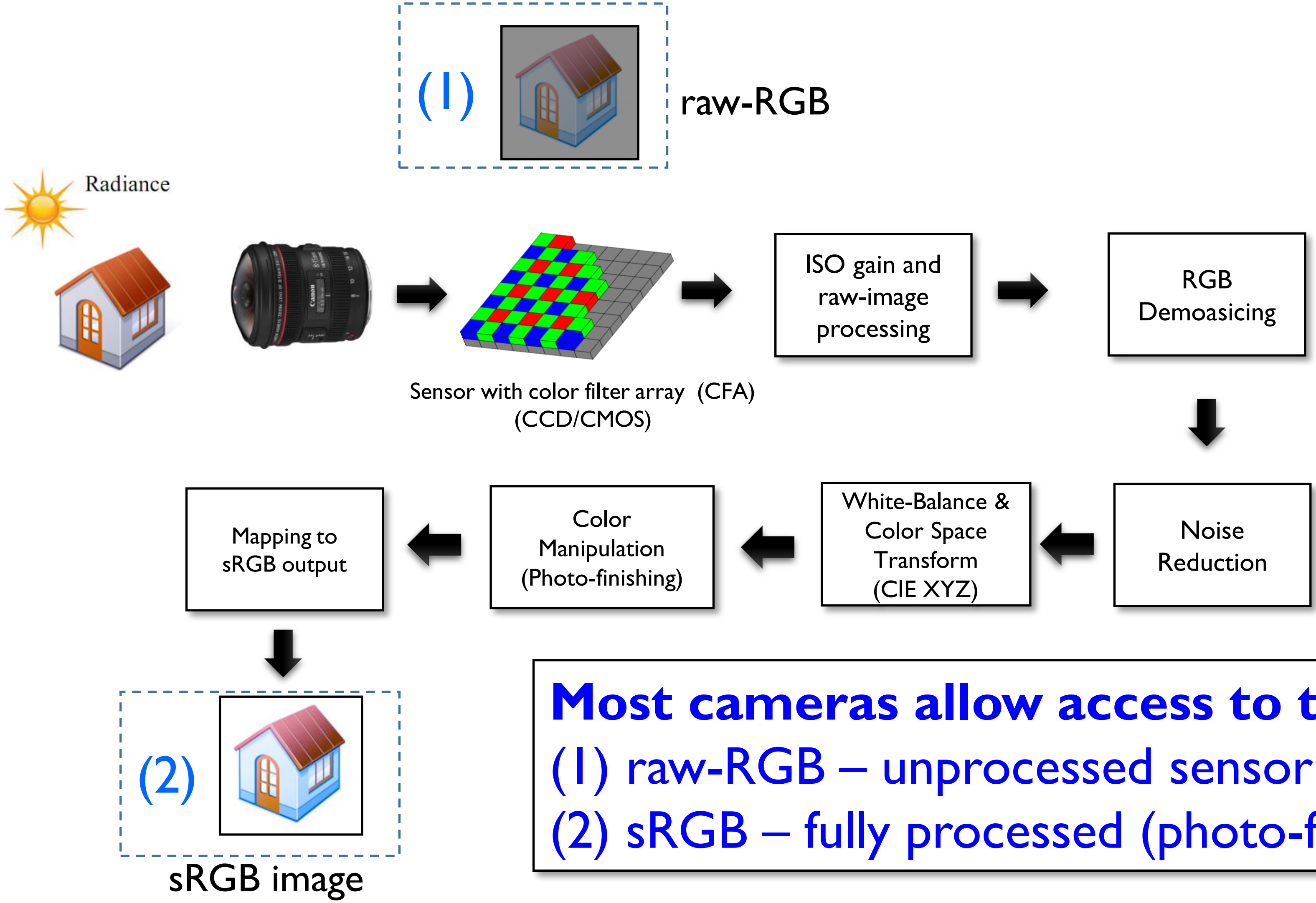
Mistake: Working in the wrong color space

- This problem stems from a lack of understanding of color spaces

- You can't just say "RGB", you need to specify **which** RGB (e.g. , raw-RGB, sRGB, NTSC, P3, etc). If you got the image off the web, it is highly likely to be sRGB.

- Certain color spaces are more suitable than others depending on the application.

Current access to different image states



Most cameras allow access to two image states.

(1) raw-RGB – unprocessed sensor image

(2) sRGB – fully processed (photo-finished) image

Advantage of raw-RGB

- It is linear with respect to the scene's physical light. This means the pixel values are related to the scene.

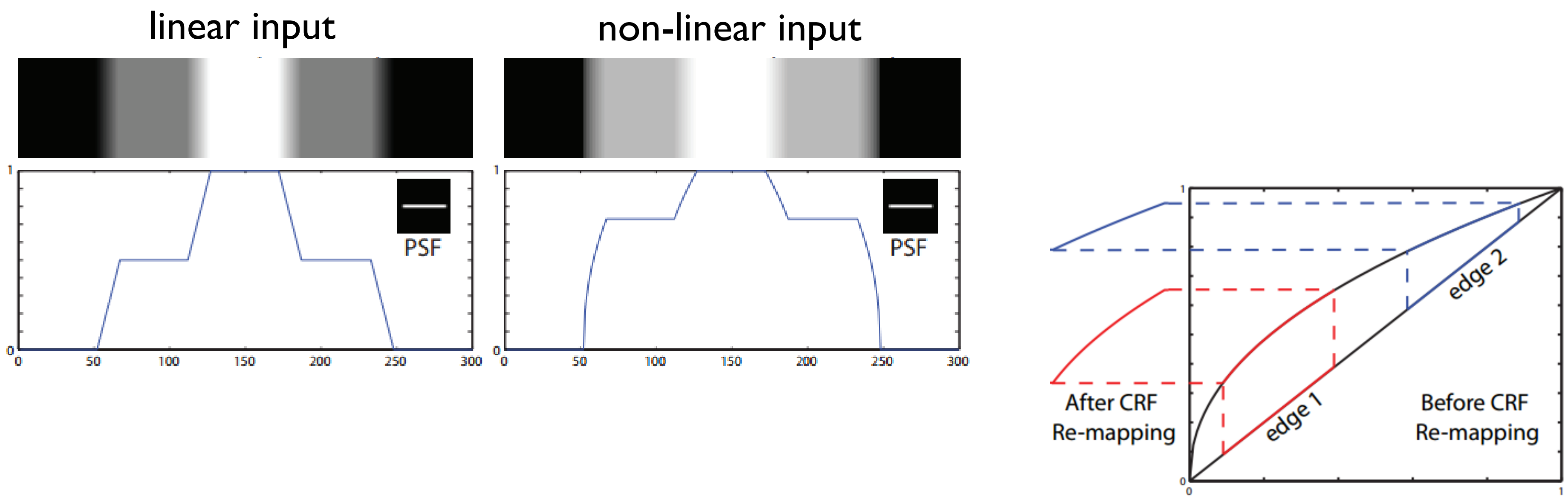
- Certain CV problems make this assumption
  - Shape from shading (photometric stereo)
  - Intrinsic image decomposition
  - Image deblurring
  - However, many times these problems are attempted in the sRGB space!

Consider image deblurring

Assumption is: linear color space (raw-RGB or linear-sRGB). Just look at the equation, convolution with I is a linear process.

Reality: sRGB image has been run through the pipeline and some non-linear modification ***f***

sRGB's nonlinear manipulation effect on blur



Tai et al. TPAMI 2012.

Deblurring in raw-RGB vs sRGB

Input with blur | Ground Truth | Deblur on raw-RGB | Deblur on sRGB

(Note ringing in the sRGB image)

Common misconception regarding sRGB

- **Assuming sRGB images are directly related to the physical scene.**

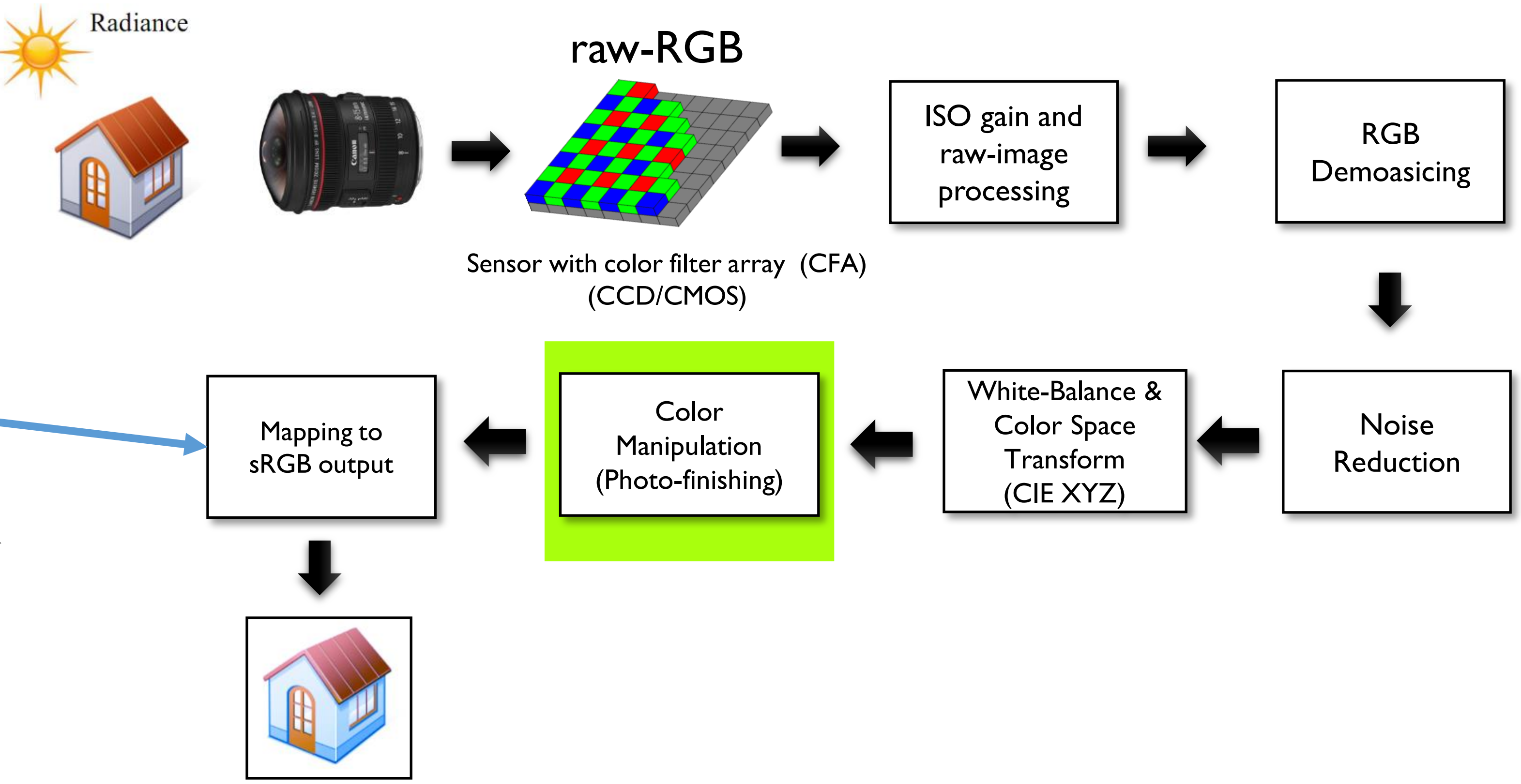
- The mistake occurs because sRGB is considered a "standard".

- sRGB is an ISO standard to describe how a photo-finished image should be encoded for display. sRGB does not state how the image would be processed before it is encoded to sRGB.

- In color science, sRGB is referred to as a "display referred" color space.

Common misconception regarding sRGB

Mapping to sRGB happens after the image has been manipulated to look good.



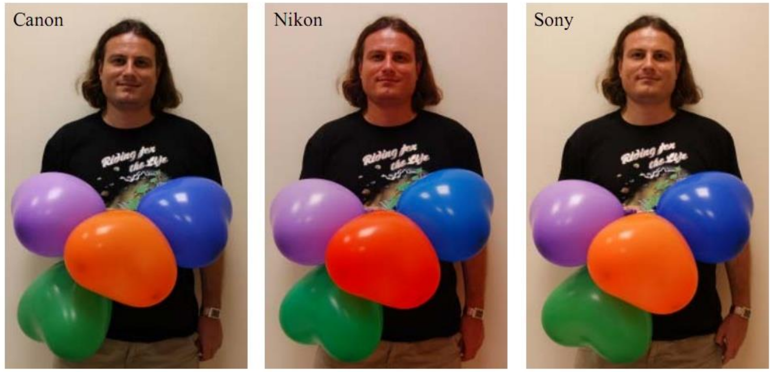
Because of the nonlinear color manipulation (i.e., photo-finishing), the sRGB images relationship to the scene radiance is **broken**.

This is a very common misconception!

sRGB images

- When you are working with an sRGB image, it represents the photo-finished image
- The colors represent the colors of the "photograph", not the colors of the "scene"
- These colors will be different for different cameras or even same camera with different picture styles!

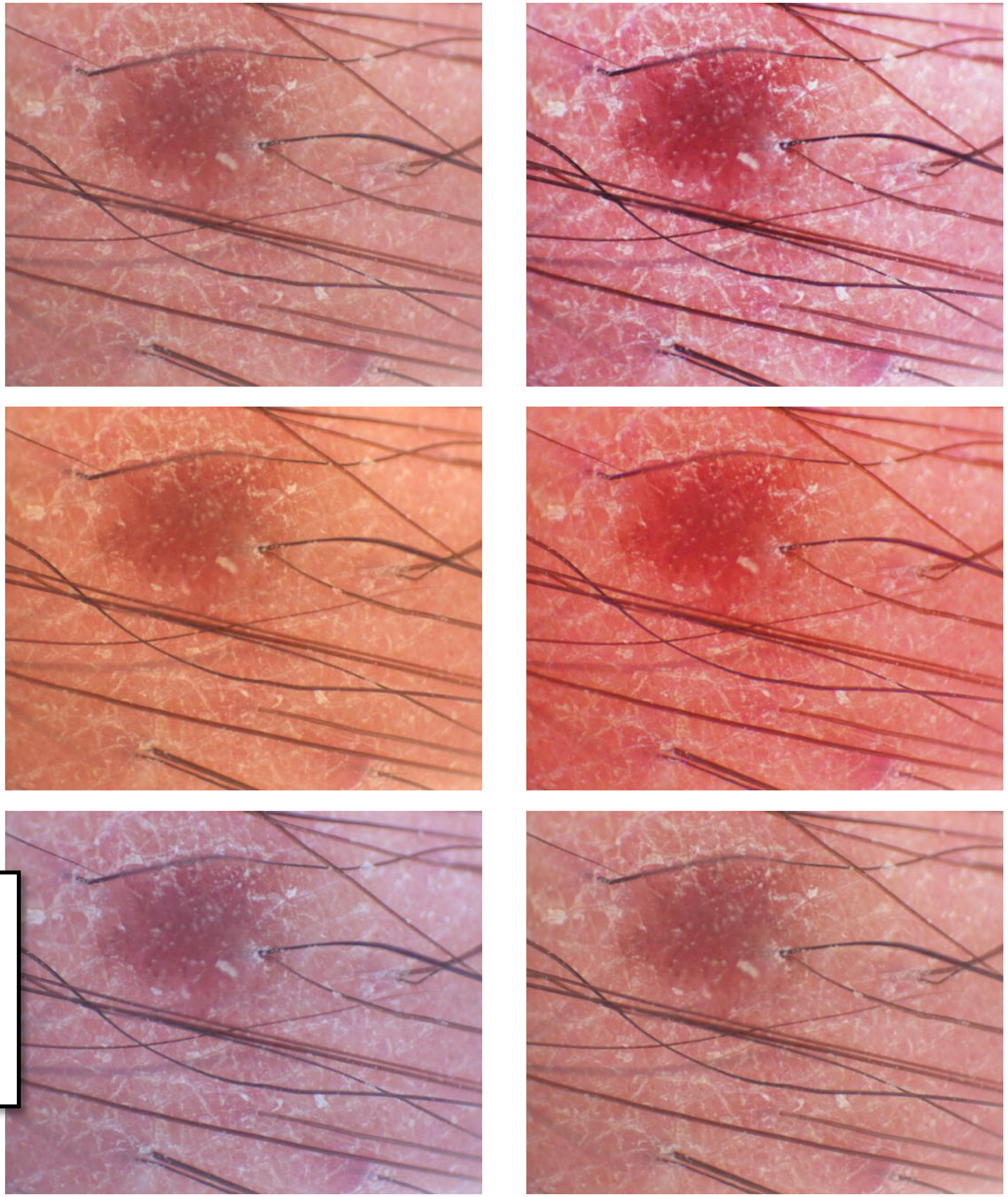
sRGB output on different cameras.



sRGB output on the same camera, w/ different color manipulation.

This assumption can led to serious mistakes in a computer vision application.

This type of processing is not suitable for scientific applications!



Which one is correct?

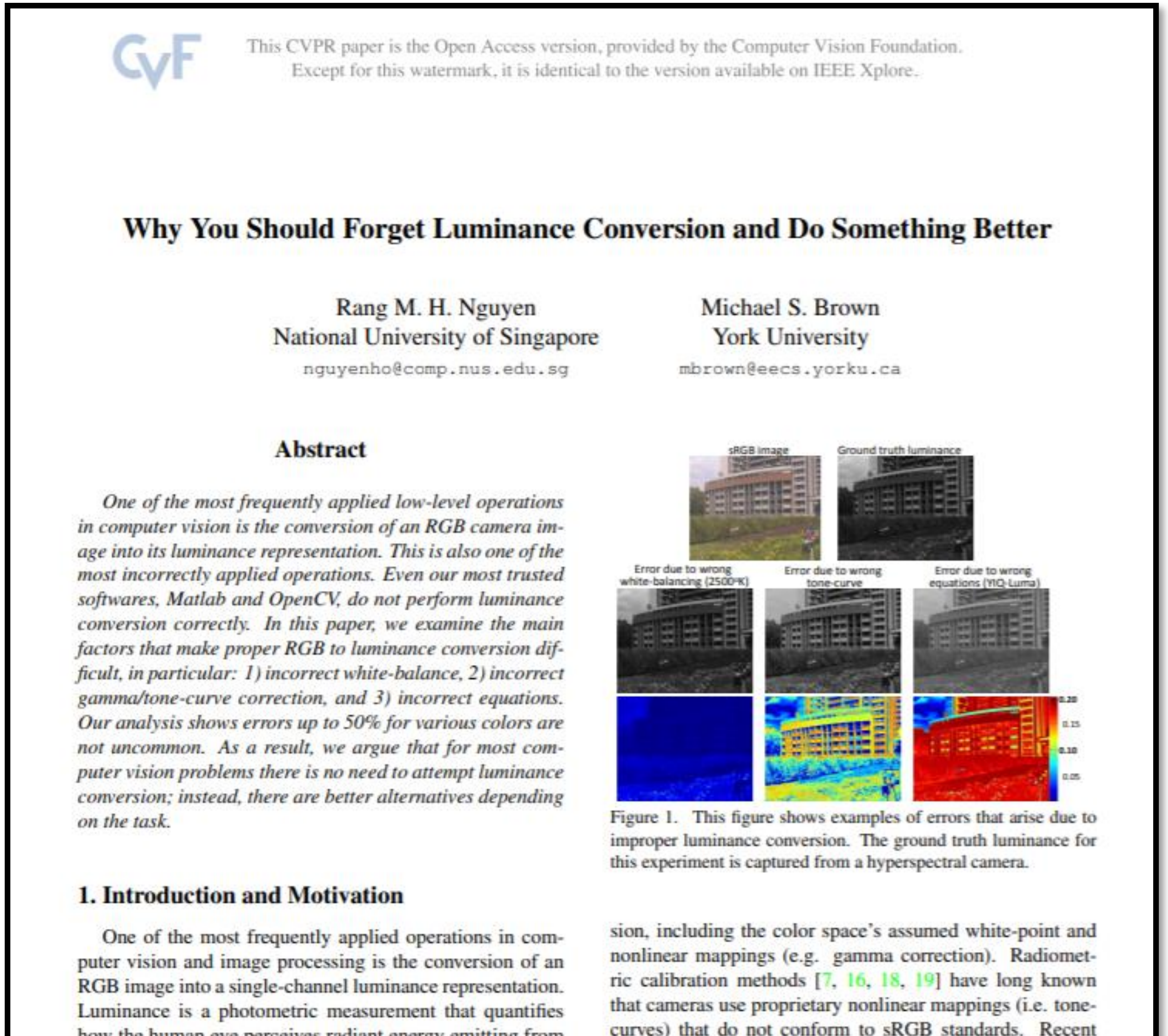
Misconception: sRGB to scene luminance

- Have you seen this before?

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

- Papers claim that by applying this equation, they are recovering the luminance of the physical scene!

Forget luminance conversion



Why You Should Forget Luminance Conversion and Do Something Better

Rang M. H. Nguyen
National University of Singapore
nguyenho@comp.nus.edu.sg

Michael S. Brown
York University
mbrown@eecs.yorku.ca

Abstract

One of the most frequently applied low-level operations in computer vision is the conversion of an RGB camera image into its luminance representation. This is also one of the most incorrectly applied operations. Even our most trusted softwares, Matlab and OpenCV, do not perform luminance conversion correctly. In this paper, we examine the main factors that make proper RGB to luminance conversion difficult, in particular: 1) incorrect white-balance, 2) incorrect gamma/tone-curve correction, and 3) incorrect equations. Our analysis shows errors up to 50% for various colors are not uncommon. As a result, we argue that for most computer vision problems there is no need to attempt luminance conversion; instead, there are better alternatives depending on the task.

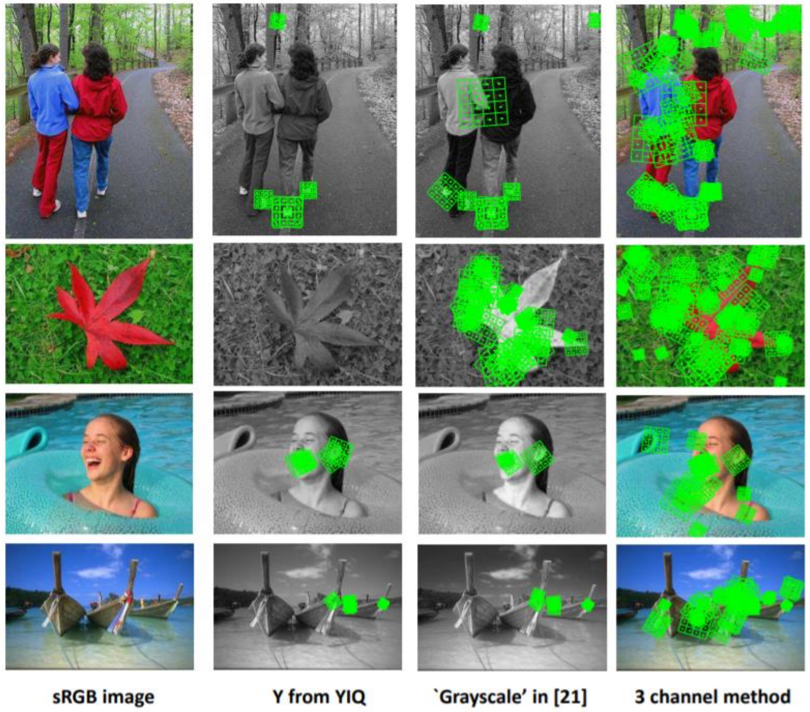
sRGB image
Ground truth luminance
Error due to wrong white-balancing (2500°K)
Error due to wrong tone-curve
Error due to wrong equations (YIQ-Luma)

Figure 1. This figure shows examples of errors that arise due to improper luminance conversion. The ground truth luminance for this experiment is captured from a hyperspectral camera.

1. Introduction and Motivation

One of the most frequently applied operations in computer vision and image processing is the conversion of an RGB image into a single-channel luminance representation. Luminance is a photometric measurement that quantifies

sion, including the color space's assumed white-point and nonlinear mappings (e.g. gamma correction). Radiometric calibration methods [7, 16, 18, 19] have long known that cameras use proprietary nonlinear mappings (i.e. tone-curves) that do not conform to sRGB standards. Recent



Usually we apply luminance conversion because we want to process only a single image.

It is OK to just use the *green* channel, or a contrast preserving decolorization method (Lu et al IJCV'14)

Misconception: Post-capture correction of WB

- Another misconception is that it is easy to correct color mistakes made by the camera post-capture (i.e., by modifying the sRGB image)

- For example, if your white balance was incorrect at capture time, you can just apply a diagonal white-balance matrix to correct it.

sRGB image with the wrong white-balance

Matlab's result by applying an inverse sRGB gamma, the correct WB, then re-applying the sRGB gamma. This doesn't undo the photo-finishing!

Properly white-balanced result when white-balance is applied before photo-finishing

Correcting an incorrectly WB image

input: sRGB with wrong WB

Post-capture correction using a WB matrix

$$\begin{bmatrix} r_{wb} \\ g_{wb} \\ b_{wb} \end{bmatrix} = \begin{bmatrix} 1/\ell_r & 0 & 0 \\ 0 & 1/\ell_g & 0 \\ 0 & 0 & 1/\ell_b \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix}$$

Captured with the right WB

Applying the 3x3 diagonal white-balance post-capture doesn't correct the problem

Correcting an incorrectly WB image

input: sRGB with wrong WB

Post-capture correction using a WB matrix

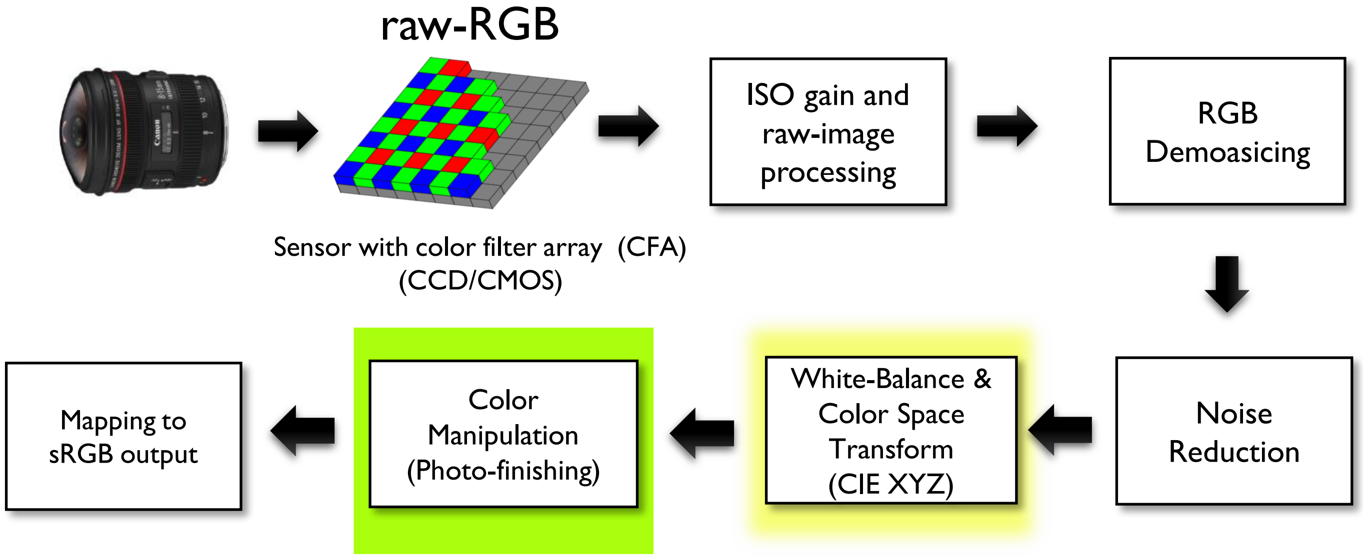
$$\begin{bmatrix} r_{wb} \\ g_{wb} \\ b_{wb} \end{bmatrix} = \begin{bmatrix} 1/\ell_r & 0 & 0 \\ 0 & 1/\ell_g & 0 \\ 0 & 0 & 1/\ell_b \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix}$$

Captured with the right WB

Applying the 3x3 diagonal white-balance post-capture doesn't correct the problem.
In some case, it can make it even worse!

Why is this hard to correct?

Mistakes made when capturing photos



Mistakes made **early** in the camera pipeline **are propagated**.

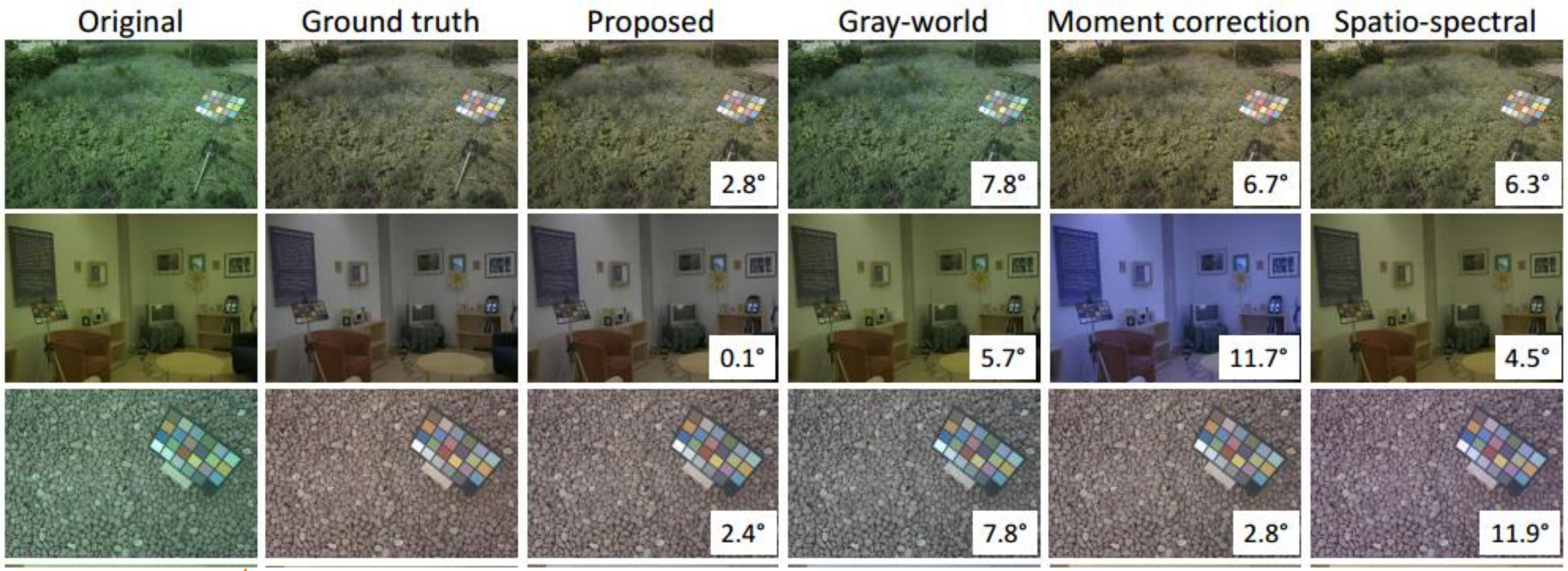
Surprisingly, many resources erroneously suggest this will work!

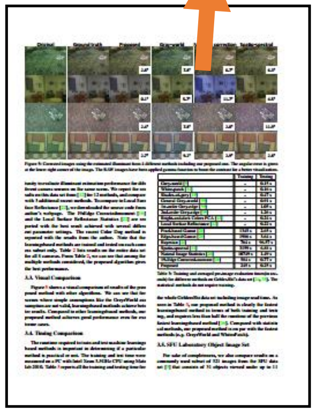
$$\begin{bmatrix} r_{wb} \\ g_{wb} \\ b_{wb} \end{bmatrix} = \begin{bmatrix} 1/ & & \\ 0 & & \\ 0 & 0 & 1/\ell_b \end{bmatrix} \begin{bmatrix} \\ \\ b \end{bmatrix}$$

Post-capture white-balance correction

- Because white-balance happens before the non-linear photofinishing, applying a diagonal white-balance correction will not work.
- You would need to reverse all the non-linear processing applied on the camera and then re-apply the camera pipeline.
- Note, the non-linear processing is not jus the sRGB gamma encoding, but all the camera-specific (and picture style) specific color manipulation

Another mistake: interpreting images "out-of-context"





From Cheng et al CVPR 2015

"Subjective" white balance results shown
(images are in the camera's raw-RGB color space)

These subjective results have **absolutely *no*** visual meaning.
The camera-raw is not a standard color space!

Tutorial schedule

- Part 1 (General)
  - ~~Motivation~~
  - ~~Review of color & color spaces~~
  - ~~Overview of in-camera imaging pipeline~~

1.30pm – 3.30pm

Break
3.30pm – 4.30pm

- Part 2 (Imaging and Computer Vision)
  - ~~Misconceptions in the computer vision community regarding color~~
  - Recent work on color and cameras
  - Concluding remarks

4.30pm – 6.00pm

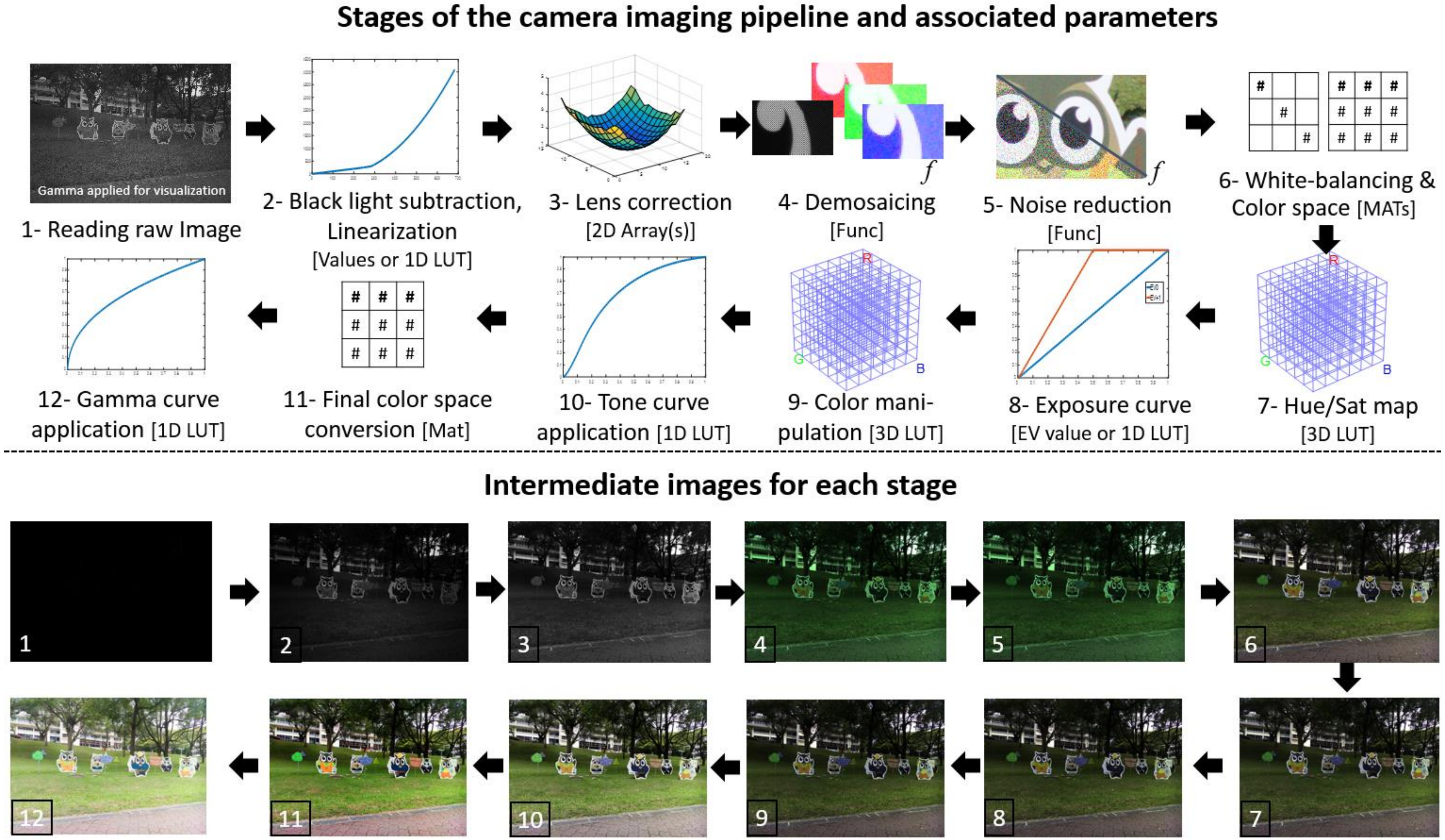
Part 2: Recent research in camera pipelines

Opening the camera pipeleine

- Camera pipelines and ISPs are "closed hardware"
  - This makes it difficult to do research on individual components and see the final output. Also, it is hard to get access to intermediate image states (e.g., CIE XYZ values)
- There are some nice libraries for raw-RGB processing
  - **libRaw**
  - **Adobe DNG SDK**
- Google's camera API
  - Opening up low-level access to make camera parameters

Software camera pipeline

[ECCV'16]
Karaimer and Brown



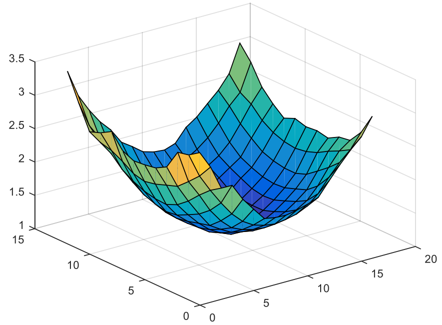
Walking through the pipeline (1)

Camera raw-RGB

Walking through the pipeline (2)

Black level
subtraction and
linearization +
defective pixel mask

Walking through the pipeline (3)



Lens correction
(non-uniform gain)

Walking through the pipeline (4)

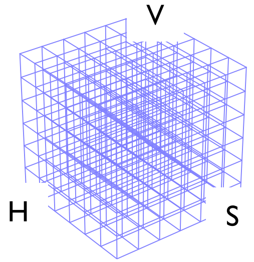
Demosaicing
+ Noise Reduction

Walking through the pipeline (5)

$$\begin{bmatrix} \# & & \\ & \# & \\ & & \# \end{bmatrix} \quad \begin{bmatrix} \# & \# & \# \\ \# & \# & \# \\ \# & \# & \# \end{bmatrix}$$

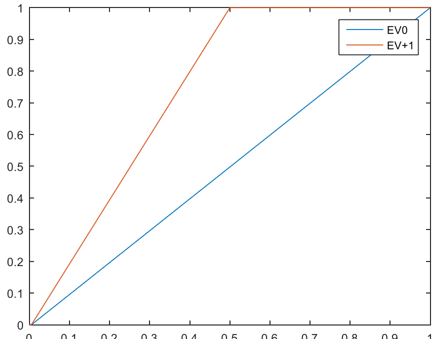
White balance
+ color space transform
(CIE XYZ/Pro-photo)

Walking through the pipeline (6)



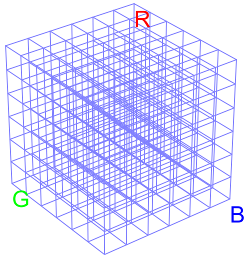
Hue-saturation adjustment

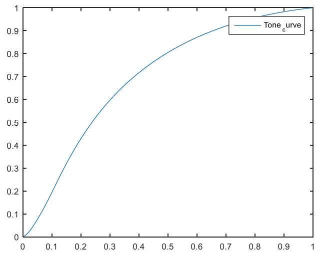
Walking through the pipeline (7)



Exposure
Compensation

Walking through the pipeline (8)





Color rendering

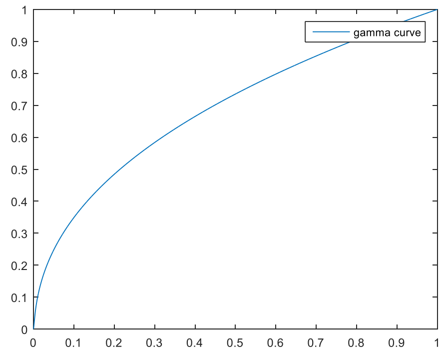
Walking through the pipeline (9)

#	**#**	**#**
#	#	#
#	#	#

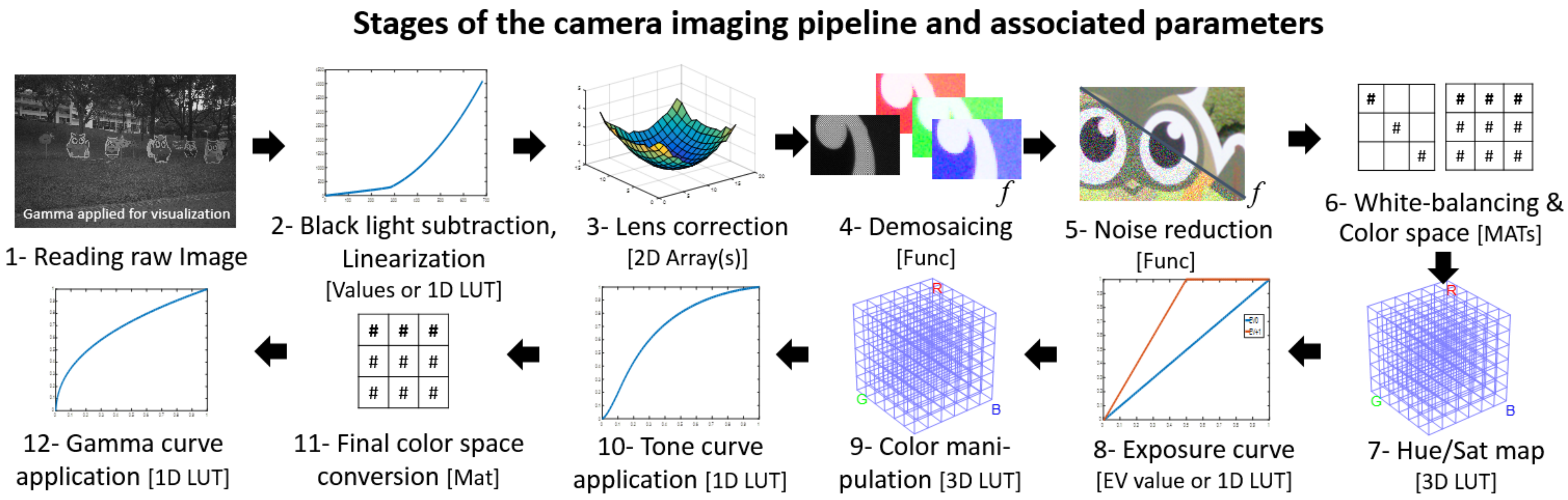
sRGB conversion

Walking through the pipeline (10)

sRGB gamma

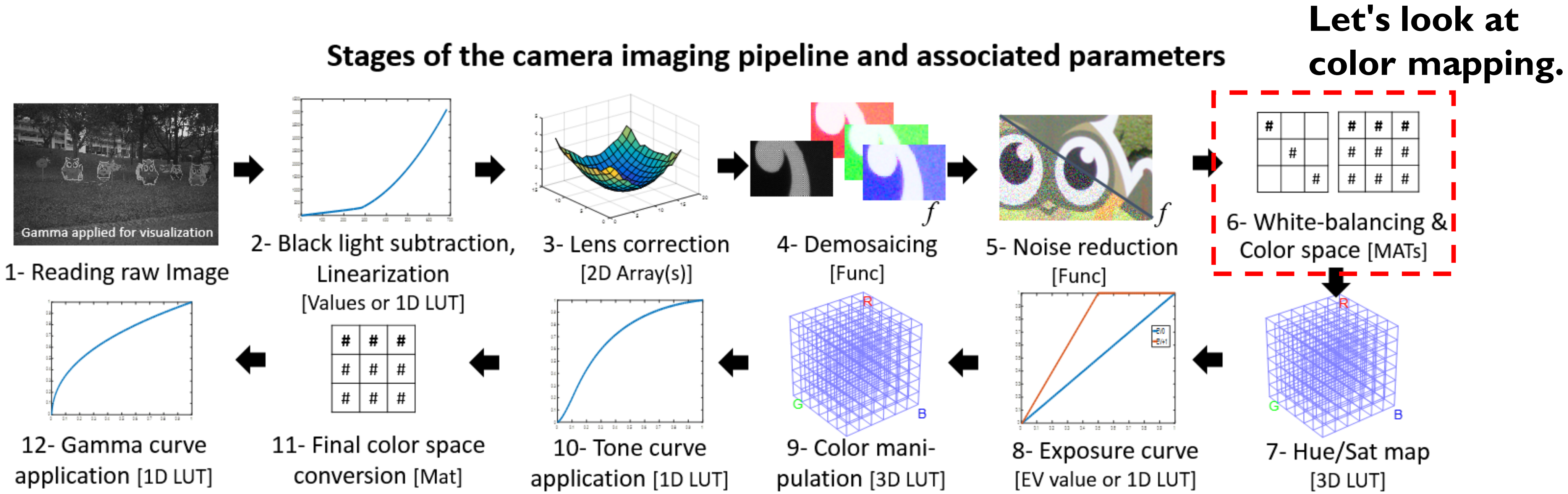


Lots of opportunities



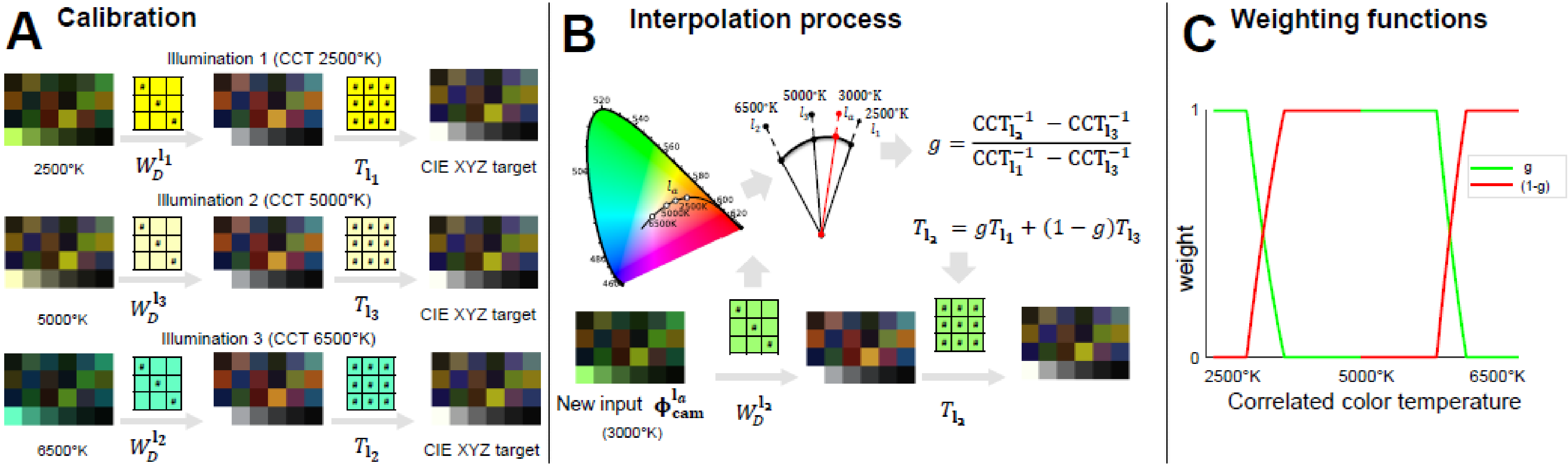
Now, we can analyze each step of the pipeline.

Lots of opportunities

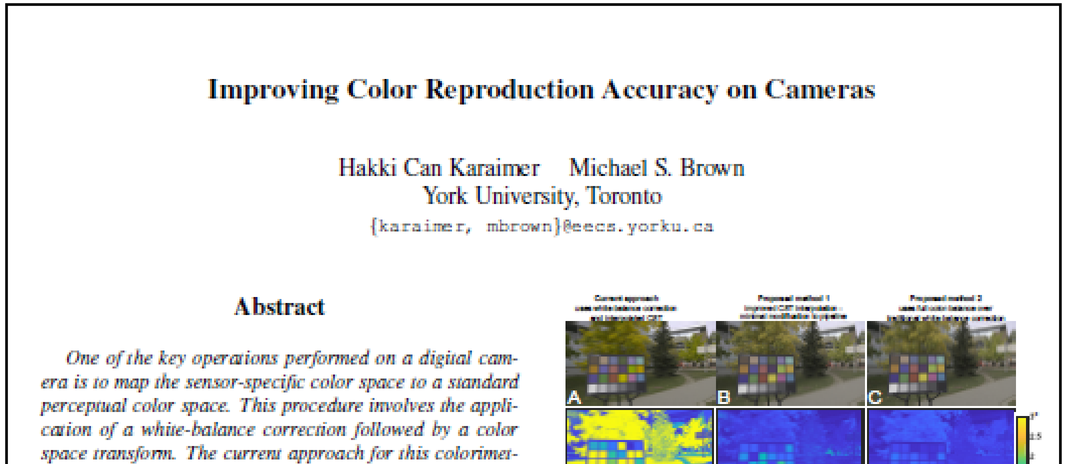


Improving color for cameras

Simply add another calibrated illumination.



[CVPR'18]



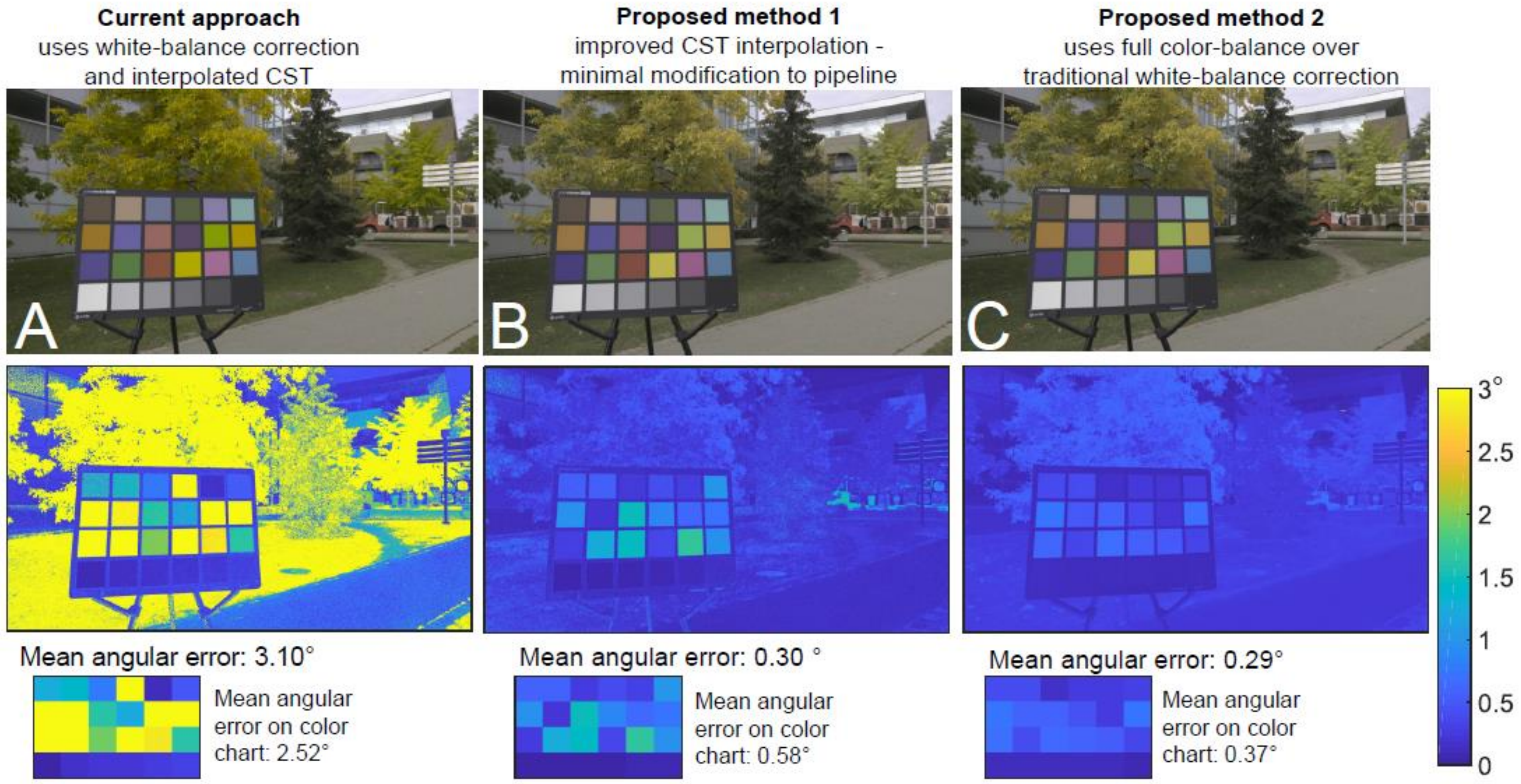
Improving Color Reproduction Accuracy on Cameras

Hakki Can Karaimer Michael S. Brown
York University, Toronto
{karaimer, mbrown}@eecs.yorku.ca

Abstract

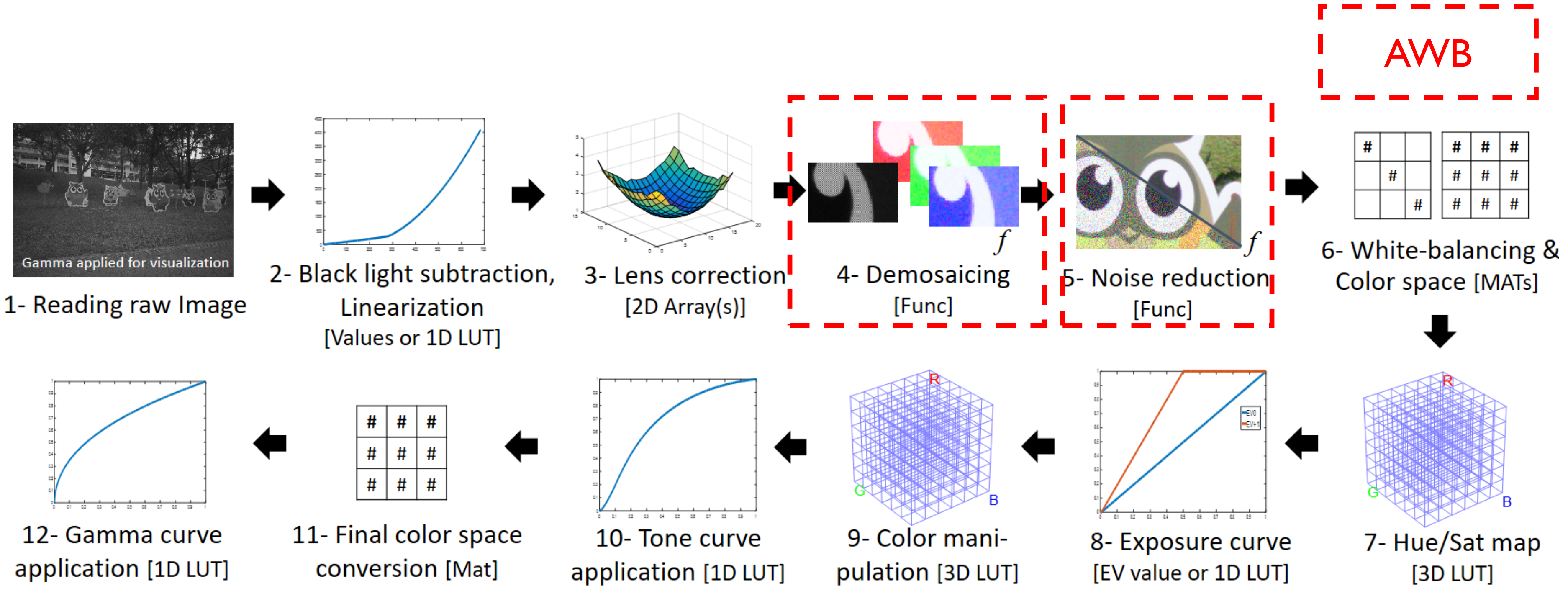
One of the key operations performed on a digital camera is to map the sensor-specific color space to a standard perceptual color space. This procedure involves the application of a white-balance correction followed by a color space transform. The current approach for this colorimet-

Improving color for cameras



Deep learning and the camera pipeline?

- Deep-learning outperforms many traditional methods
- Where is it appropriate for camera pipelines?
  - Devote to hard tasks (demoasicing, NR, AWB)



Improving AWB

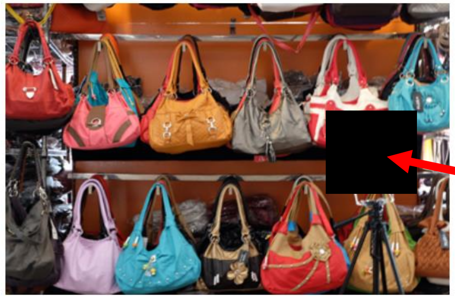
- Lou et al "Color Constancy by Deep Learning", BMVC 2015
- Hu et al "FC4: Fully Convolutional Color Constancy with Confidence-weighted Pooling", CVPR 2017
- Barron "Convolutional Color Constancy", ICCV 2015
- Oh et al "Approaching the computational color constancy as a classification problem through deep learning", Pattern Recognition, 2017
- **Many many more . . .**

Training data for AWB

Training images
raw-RGB with a neutral or achromatic object inserted.
The label/ground-truth for the training data is the raw-RGB value in the region of the neutral object.

Neutral object in the scene. The raw-RGB values at this pixel location is considered to be the entire scenes illumination.
In the case of the color chart, only the white-patches are used.

Testing/validation
The neutral object is masked out.



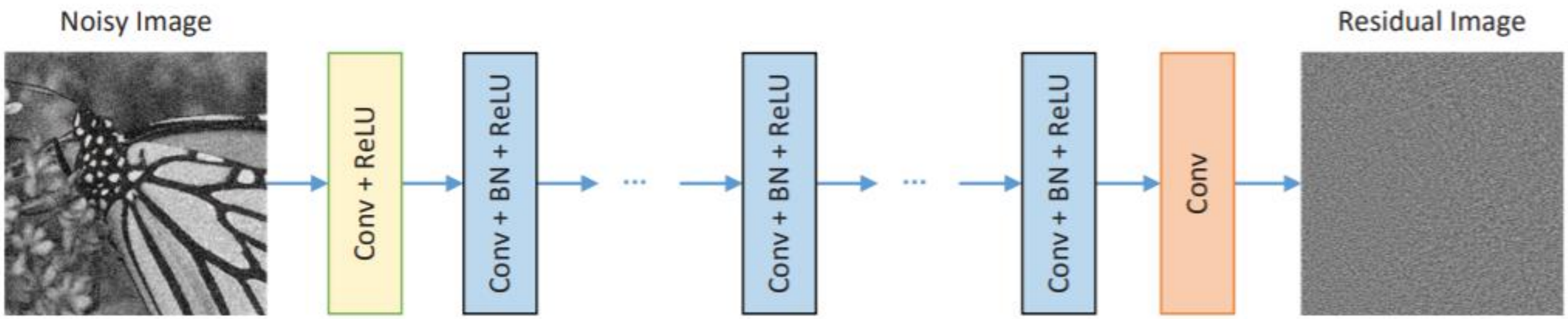
Testing image

Neutral object masked out.

Original image

DNNs for denoising

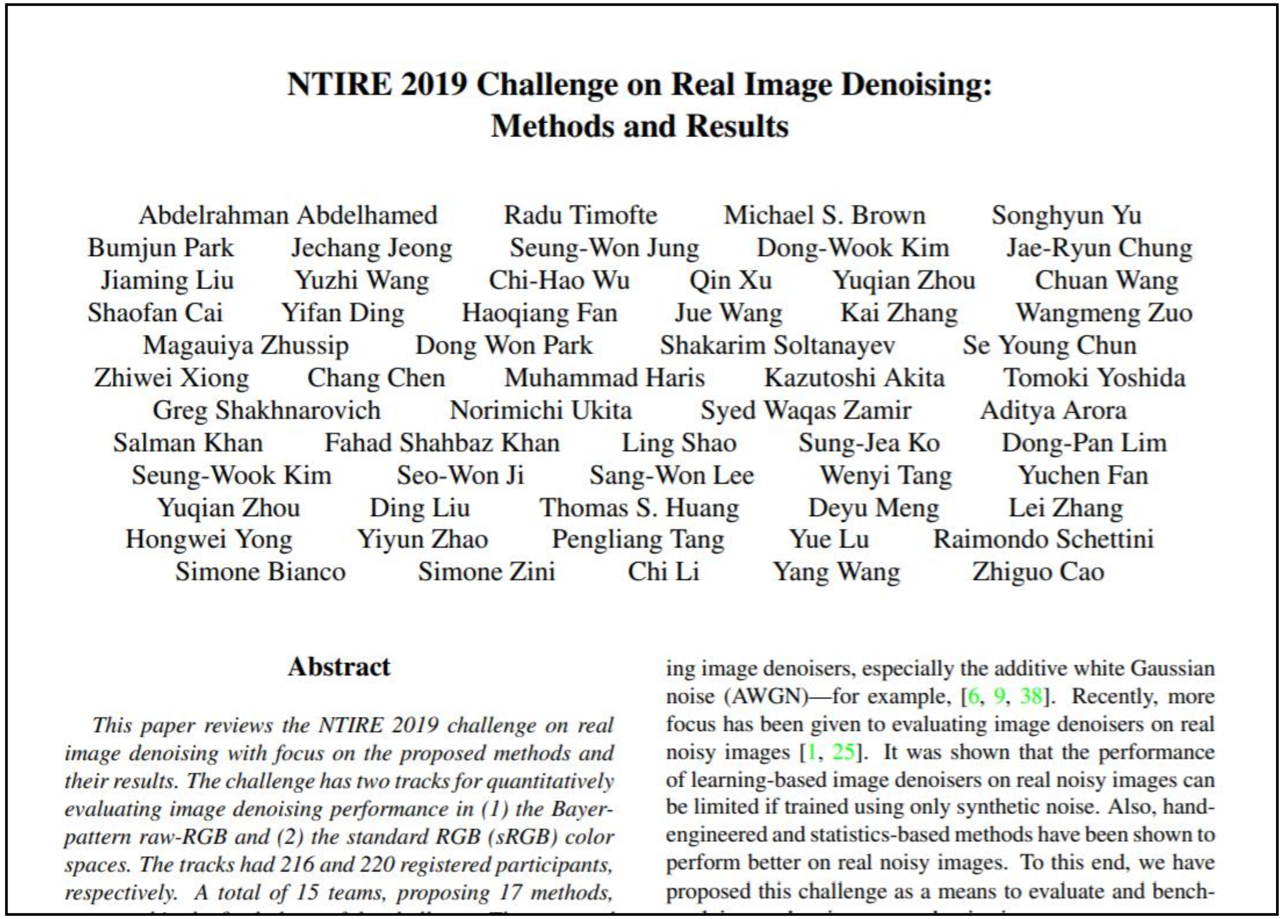
- Zhang et al "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising", TIP 2017
- The following is a website with many denoising papers with code available:
  https://paperswithcode.com/task/image-denoising?page=2



Many methods predict the noise image instead of the denoised image (DnNN from TIP 2017)

Denoising contest at CVPR'19

NTIRE denoising contest at CVPR'19



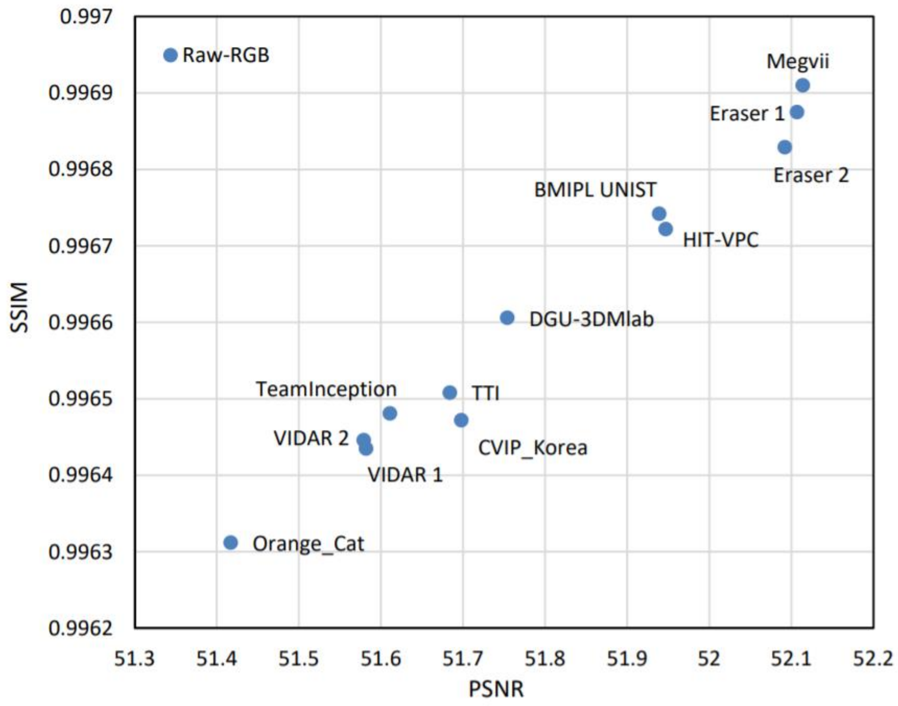
NTIRE 2019 Challenge on Real Image Denoising: Methods and Results

Abdelrahman Abdelhamed, Radu Timofte, Michael S. Brown, Songhyun Yu, Bumjun Park, Jechang Jeong, Seung-Won Jung, Dong-Wook Kim, Jae-Ryun Chung, Jiaming Liu, Yuzhi Wang, Chi-Hao Wu, Qin Xu, Yuqian Zhou, Chuan Wang, Shaofan Cai, Yifan Ding, Haoqiang Fan, Jue Wang, Kai Zhang, Wangmeng Zuo, Magauiya Zhussip, Dong Won Park, Shakarim Soltanayev, Se Young Chun, Zhiwei Xiong, Chang Chen, Muhammad Haris, Kazutoshi Akita, Tomoki Yoshida, Greg Shakhnarovich, Norimichi Ukita, Syed Waqas Zamir, Aditya Arora, Salman Khan, Fahad Shahbaz Khan, Ling Shao, Sung-Jea Ko, Dong-Pan Lim, Seung-Wook Kim, Seo-Won Ji, Sang-Won Lee, Wenyi Tang, Yuchen Fan, Yuqian Zhou, Ding Liu, Thomas S. Huang, Deyu Meng, Lei Zhang, Hongwei Yong, Yiyun Zhao, Pengliang Tang, Yue Lu, Raimondo Schettini, Simone Bianco, Simone Zini, Chi Li, Yang Wang, Zhiguo Cao

Abstract

This paper reviews the NTIRE 2019 challenge on real image denoising with focus on the proposed methods and their results. The challenge has two tracks for quantitatively evaluating image denoising performance in (1) the Bayer-pattern raw-RGB and (2) the standard RGB (sRGB) color spaces. The tracks had 216 and 220 registered participants, respectively. A total of 15 teams, proposing 17 methods,

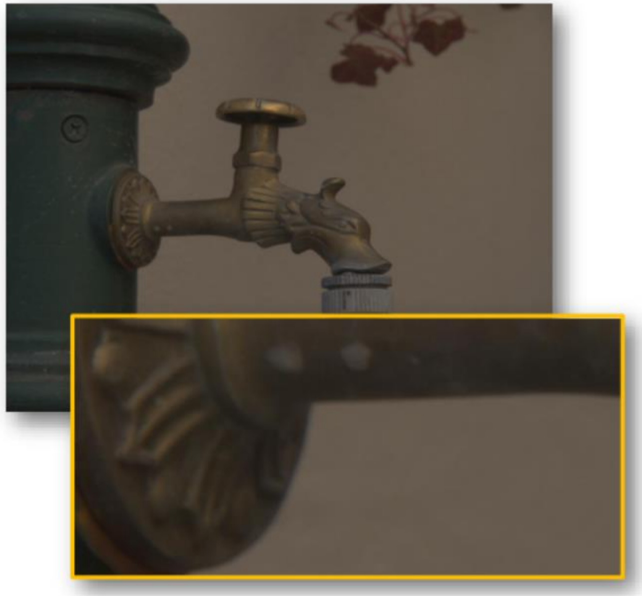
ing image denoisers, especially the additive white Gaussian noise (AWGN)—for example, [6, 9, 38]. Recently, more focus has been given to evaluating image denoisers on real noisy images [1, 25]. It was shown that the performance of learning-based image denoisers on real noisy images can be limited if trained using only synthetic noise. Also, hand-engineered and statistics-based methods have been shown to perform better on real noisy images. To this end, we have proposed this challenge as a means to evaluate and bench-



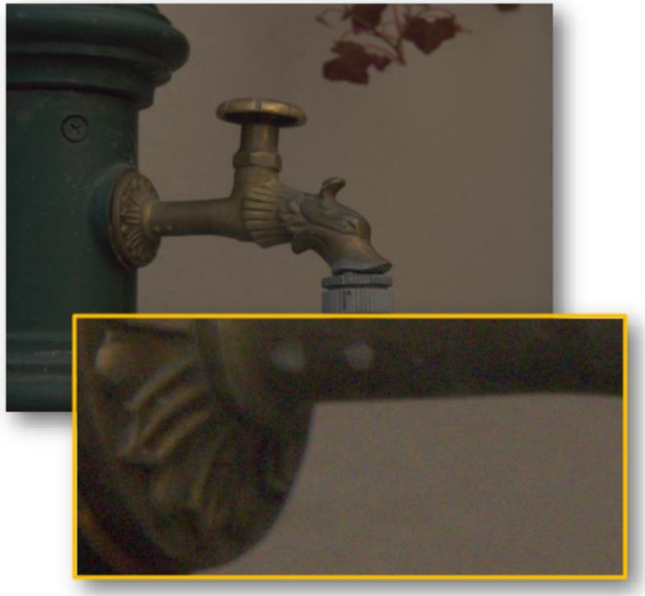
More than 10 DNN-based submissions

Training data for noise reduction (denoising)

Training/validation/testing images
Real images captured under low-ISO and high-ISO.
Or, synthetic images with noise added.



Label/ground truth image
Low ISO setting



Noisy image
High ISO setting

Example from Darmstadt Noise Dataset

Most DNNs operate on small patches (32x32). So, a single image provides many training samples.

Denoising dataset

SIDD: Smartphone Image Denoising Dataset

Abdelhamed et al CVPR 2018

A High-Quality Denoising Dataset for Smartphone Cameras

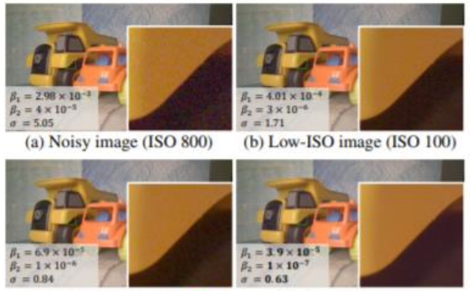
Abdelrahman Abdelhamed
York University
kamel@eecs.yorku.ca

Stephen Lin
Microsoft Research
stevelin@microsoft.com

Michael S. Brown
York University
mbrown@eecs.yorku.ca

Abstract

The last decade has seen an astronomical shift from imaging with DSLR and point-and-shoot cameras to imaging with smartphone cameras. Due to the small aperture and sensor size, smartphone images have notably more noise than their DSLR counterparts. While denoising for smartphone images is an active research area, the research community currently lacks a denoising image dataset representative of real noisy images from smartphone cameras with high-quality ground truth. We address this issue in this paper with the following contributions. We propose a systematic procedure for estimating ground truth for noisy images that can be used to benchmark denoising performance for smartphone cameras. Using this procedure, we have captured a dataset – the Smartphone Image Denoising Dataset (SIDD) *– of ~30,000 noisy images from 10 scenes under different lighting conditions using five representative smartphone cameras and generated their ground truth images. We used this dataset to benchmark a number of denoising algorithms. We show that CNN-based methods perform better when trained on our high-quality dataset than when trained using alternative strategies, such as low-ISO images used as a proxy for ground truth data.*



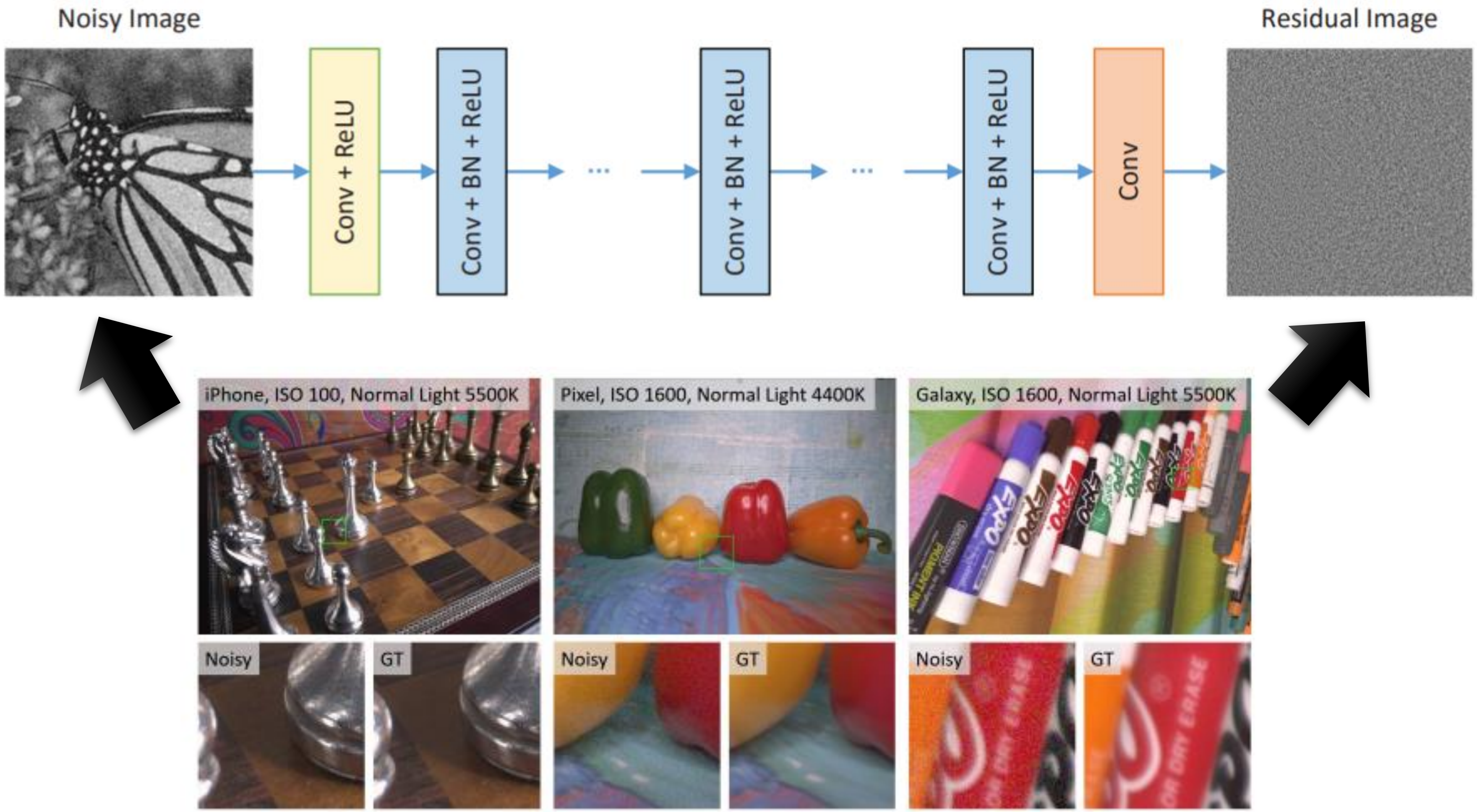
(a) Noisy image (ISO 800) (b) Low-ISO image (ISO 100)
(c) Ground truth using [25] (d) Our ground truth

Figure 1: An example scene imaged with an LG *G4* smartphone camera: (a) a high-ISO noisy image; (b) same scene captured with low ISO – this type of image is often used as ground truth for (a); (c) ground truth estimated by [25]; (d) our ground truth. Noise estimates (β_1 and β_2 for noise level function and σ for Gaussian noise – see Section 3.2) indicate that our ground truth has significantly less noise than both (b) and (c). Images shown are processed in raw-RGB, while sRGB images are shown here to aid visualization.

dataset is essential both to focus attention on denoising of

- -30,000 images
- -5 cameras
- -160 scene instances
- -15 ISO settings
- -Direct current lighting
- -Three illuminations

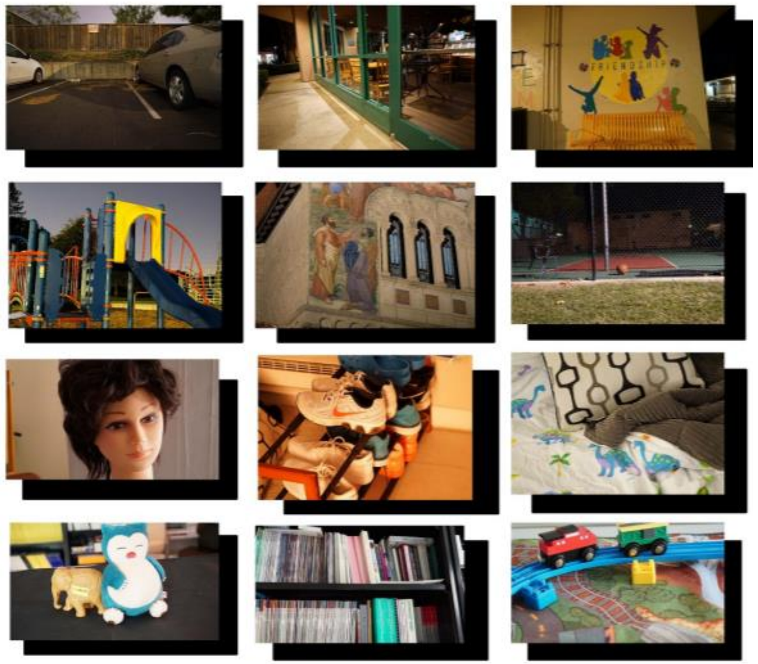
Instant denoising improvements with better training data



Learning-based DNNs get better performance when trained on real noisy images.

"Learning to see in the dark"

Sony α7S II	Filter array	Exposure time (s)	# images
x300	Bayer	1/10, 1/30	1190
x250	Bayer	1/25	699
x100	Bayer	1/10	808
Fujifilm X-T2	**Filter array**	**Exposure time (s)**	**# images**
x300	X-Trans	1/30	630
x250	X-Trans	1/25	650
x100	X-Trans	1/10	1117



Chen et al CVPR 2018

Learning to See in the Dark

Chen Chen, UIUC
Qifeng Chen, Intel Labs
Jia Xu, Intel Labs
Vladlen Koltun, Intel Labs

(a) Camera output with ISO 8,000 (b) Camera output with ISO 409,600 (c) Our result from the raw data of (a)

Figure 1. Extreme low-light imaging with a convolutional network. Dark indoor environment. The illuminance at the camera is < 0.1 lux. The Sony α7S II sensor is exposed for 1/30 second. (a) Image produced by the camera with ISO 8,000. (b) Image produced by the camera with ISO 409,600. The image suffers from noise and color bias. (c) Image produced by our convolutional network applied to the raw sensor data from (a).

Abstract

Imaging in low light is challenging due to low photon count and low SNR. Short-exposure images suffer from noise, while long exposure can induce blur and is often impractical. A variety of denoising, deblurring, and enhancement techniques have been proposed, but their effectiveness is limited in extreme conditions, such as video-rate imaging at night. To support the development of learning-

cal means to increase SNR in low light, including opening the aperture, extending exposure time, and using flash. But each of these has its own characteristic drawbacks. For example, increasing exposure time can introduce blur due to camera shake or object motion.

The challenge of fast imaging in low light is well-known in the computational photography community, but remains open. Researchers have proposed techniques for denoising, deblurring, and enhancement of low-light im-

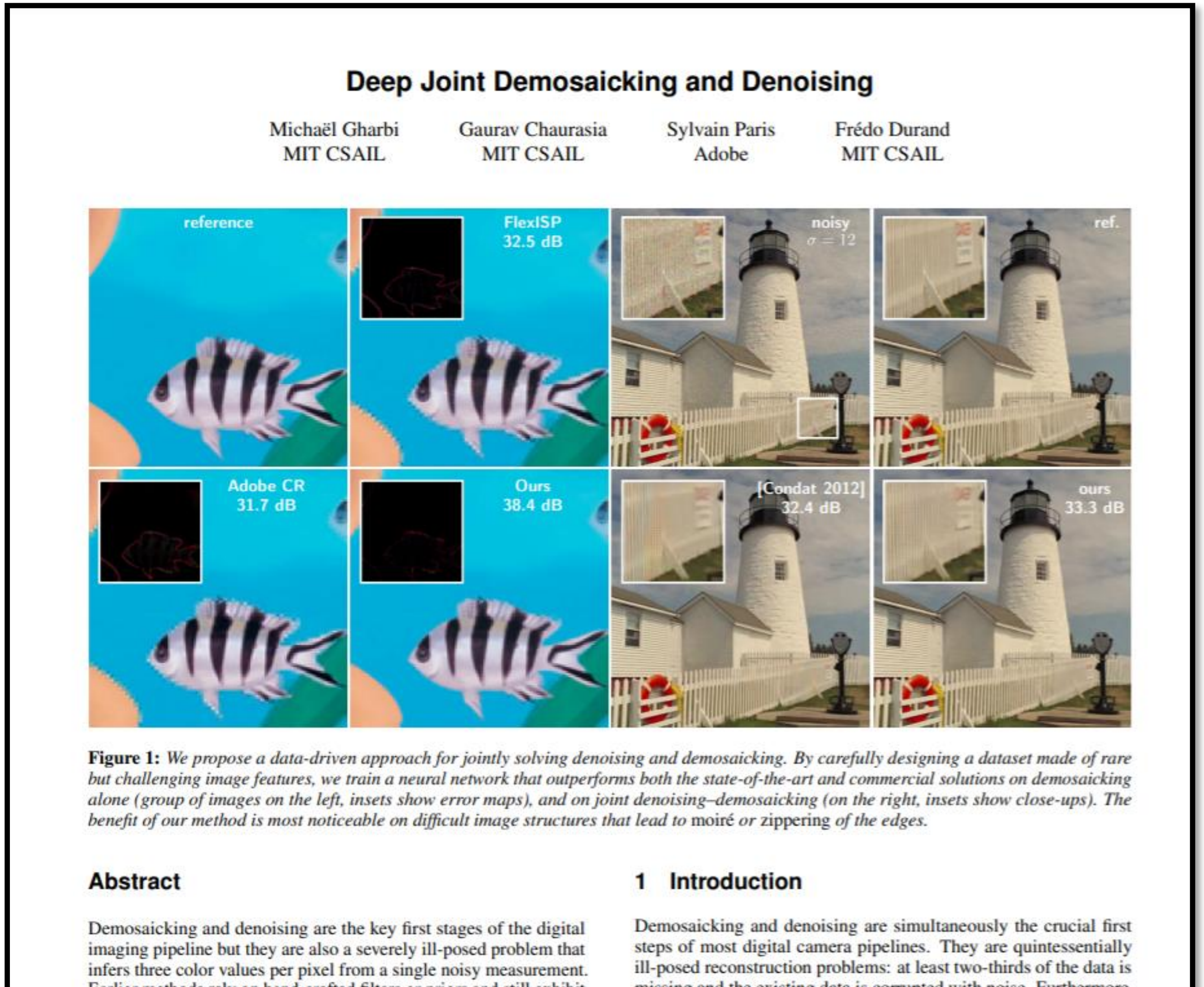
SID dark data generated 5000+ pairs of raw-RGB images captured with very short and very long exposures.

Given a short exposure image, the DNN predicts the long exposure image.

While note a denoising paper, this extended ISO feature is quite similar.

DNN for demosaicing (and denoising)

Gharbi et al SIGGRAPH Asia, 2016



Deep Joint Demosaicking and Denoising

Michaël Gharbi, MIT CSAIL — Gaurav Chaurasia, MIT CSAIL — Sylvain Paris, Adobe — Frédo Durand, MIT CSAIL

Figure 1: *We propose a data-driven approach for jointly solving denoising and demosaicking. By carefully designing a dataset made of rare but challenging image features, we train a neural network that outperforms both the state-of-the-art and commercial solutions on demosaicking alone (group of images on the left, insets show error maps), and on joint denoising–demosaicking (on the right, insets show close-ups). The benefit of our method is most noticeable on difficult image structures that lead to moiré or zippering of the edges.*

Abstract

Demosaicking and denoising are the key first stages of the digital imaging pipeline but they are also a severely ill-posed problem that infers three color values per pixel from a single noisy measurement.

1 Introduction

Demosaicking and denoising are simultaneously the crucial first steps of most digital camera pipelines. They are quintessentially ill-posed reconstruction problems: at least two-thirds of the data is

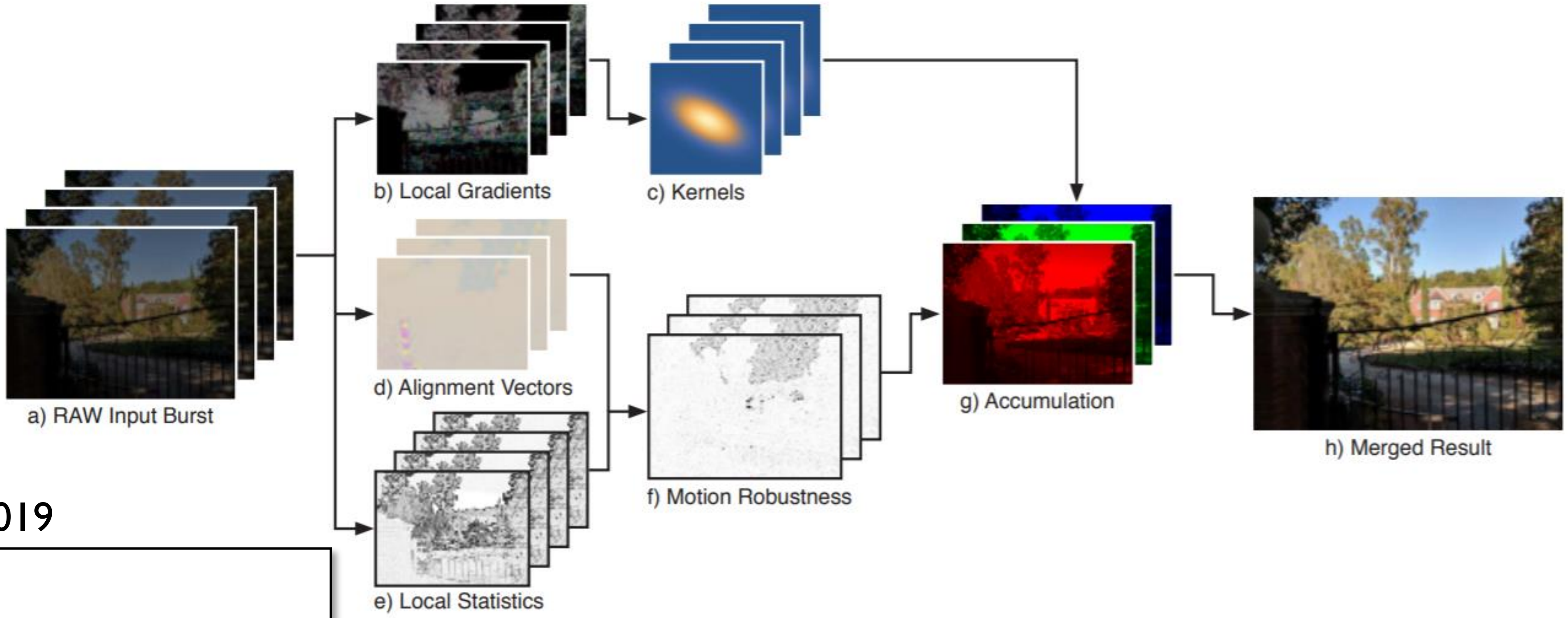
This is a very interesting paper that examines using a DNN to perform demoasicing.

Interestingly, the act of demosaicing to a clean ground truth image implicitly performs denoising.

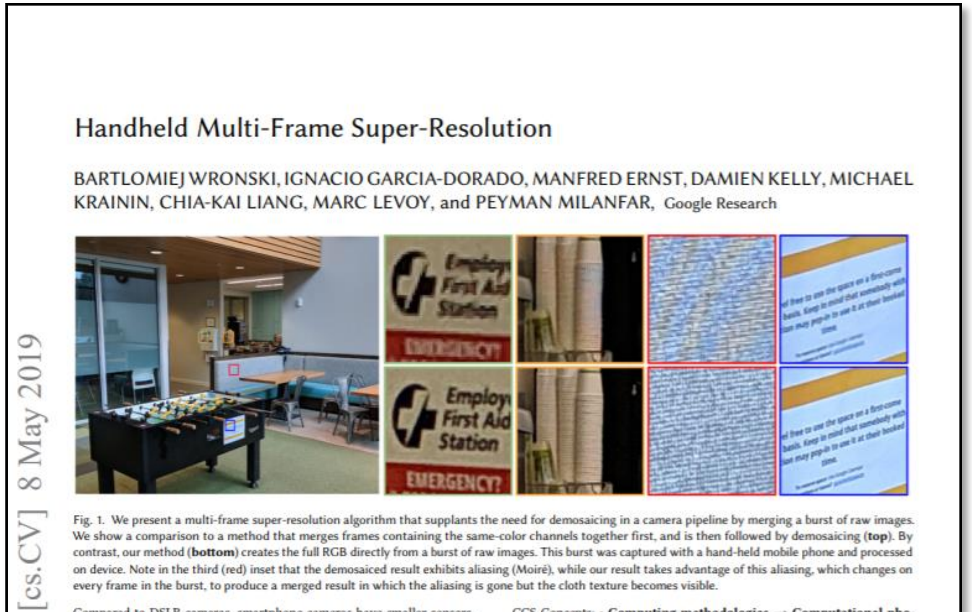
The only drawback of this paper is that training/testing demosiaced images were synthetically generated by sampling sRGB images and adding noise.

A lack of a raw-RGB dataset is a problem and would be a nice addition in this area.

Non-learning improvement for demosaicing



Wronski et al SIGGRAPH 2019

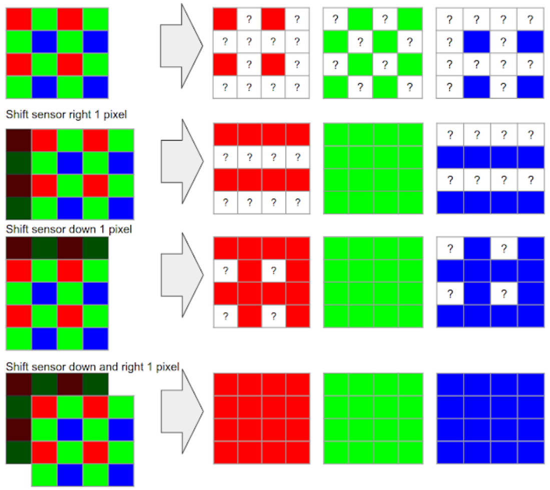


Handheld Multi-Frame Super-Resolution

BARTLOMIEJ WRONSKI, IGNACIO GARCIA-DORADO, MANFRED ERNST, DAMIEN KELLY, MICHAEL KRAININ, CHIA-KAI LIANG, MARC LEVOY, and PEYMAN MILANFAR, Google Research

[cs.CV] 8 May 2019

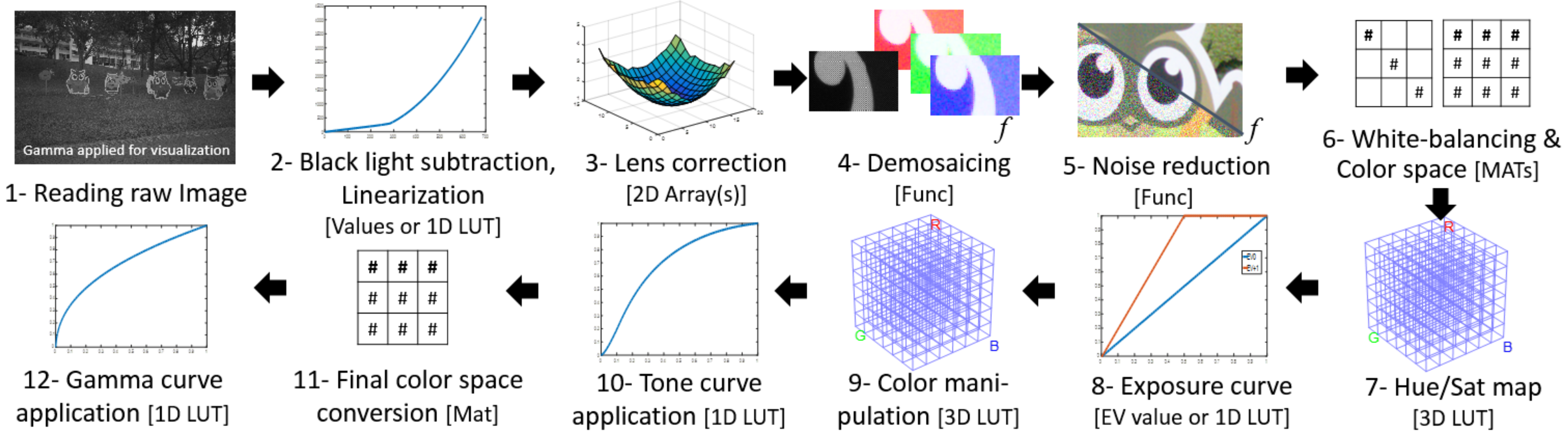
Fig. 1. We present a multi-frame super-resolution algorithm that supplants the need for demosaicing in a camera pipeline by merging a burst of raw images. We show a comparison to a method that merges frames containing the same-color channels together first, and is then followed by demosaicing (**top**). By contrast, our method (**bottom**) creates the full RGB directly from a burst of raw images. This burst was captured with a hand-held mobile phone and processed on device. Note in the third (red) inset that the demosaiced result exhibits aliasing (Moiré), while our result takes advantage of this aliasing, which changes on every frame in the burst, to produce a merged result in which the aliasing is gone but the cloth texture becomes visible.



This paper uses multiple frames and very small camera motion (from hand tremors) to perform demosaicing and super-resolution. By exploiting motion, they can fill in missing Bayer data.

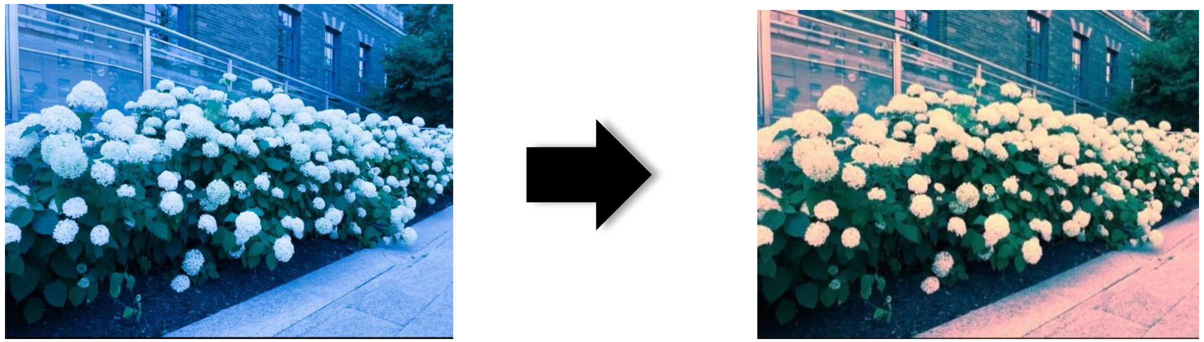
Other opportunities

Stages of the camera imaging pipeline and associated parameters



Better image encoding for post-capture correction.

Recall:
Problem of
incorrect WB.



raw-RGB disadvantages

✗ **File size is too large**

raw-RGB files are significantly larger than JPEG

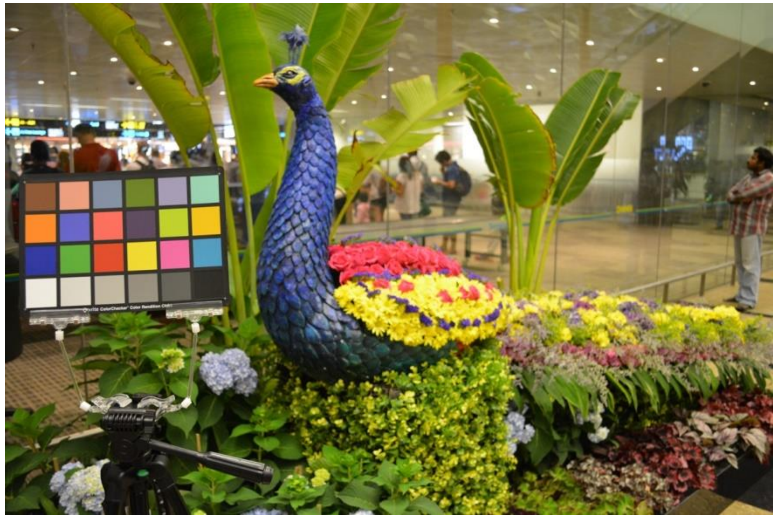
(e.g. 25-80 MB *vs* 5-10 MB per image)

✗ **Limited support**

Some (computer vision) image workflows do not support raw-RGB

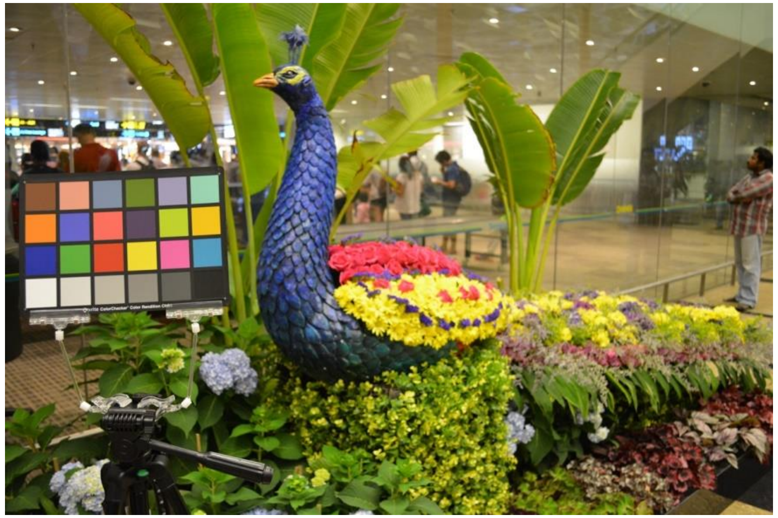
But what if . . .

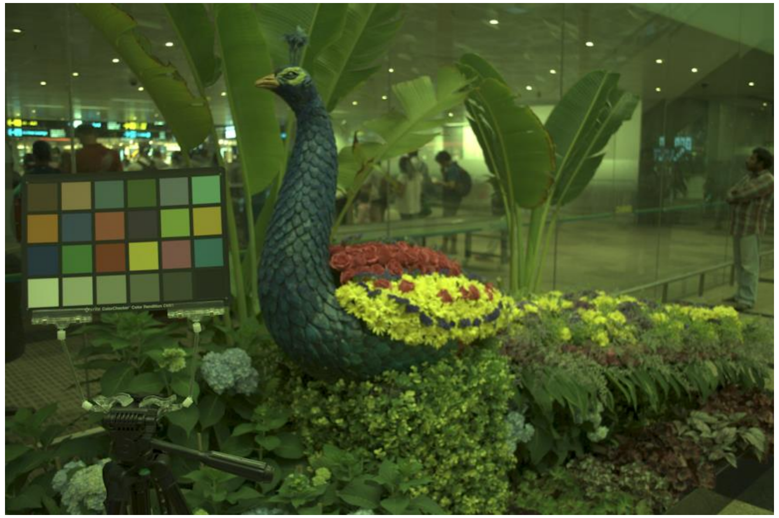
. . we could encoded the raw-RGB image *inside* the JPEG image for (almost) free?



But what if . . .

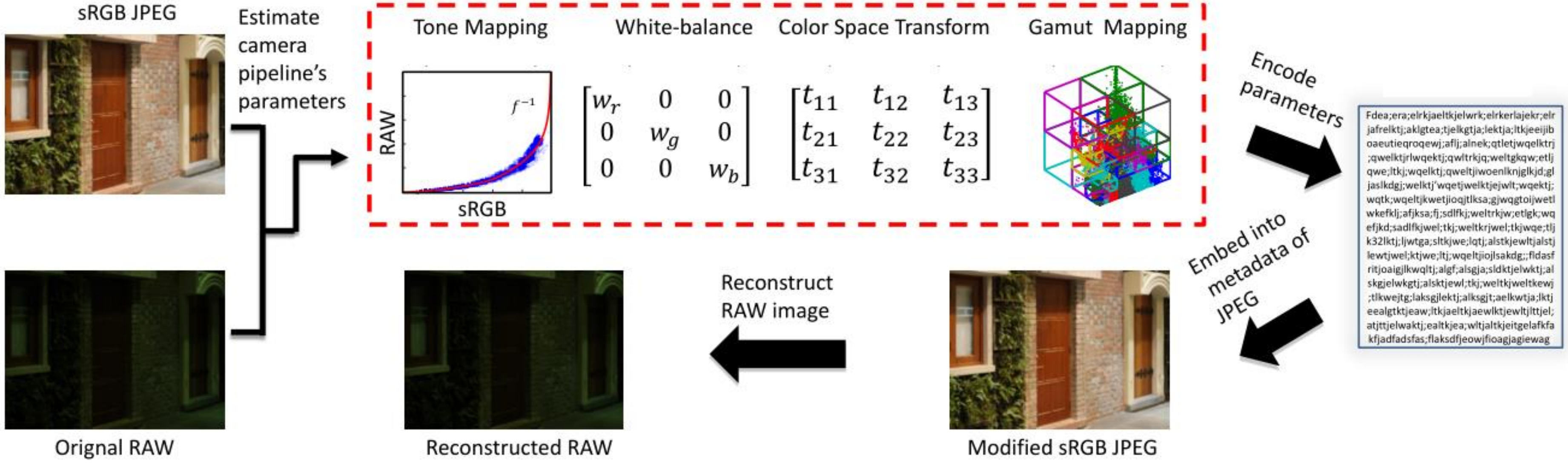
. . we could encoded the raw-RGB image *inside* the JPEG image for (almost) free?



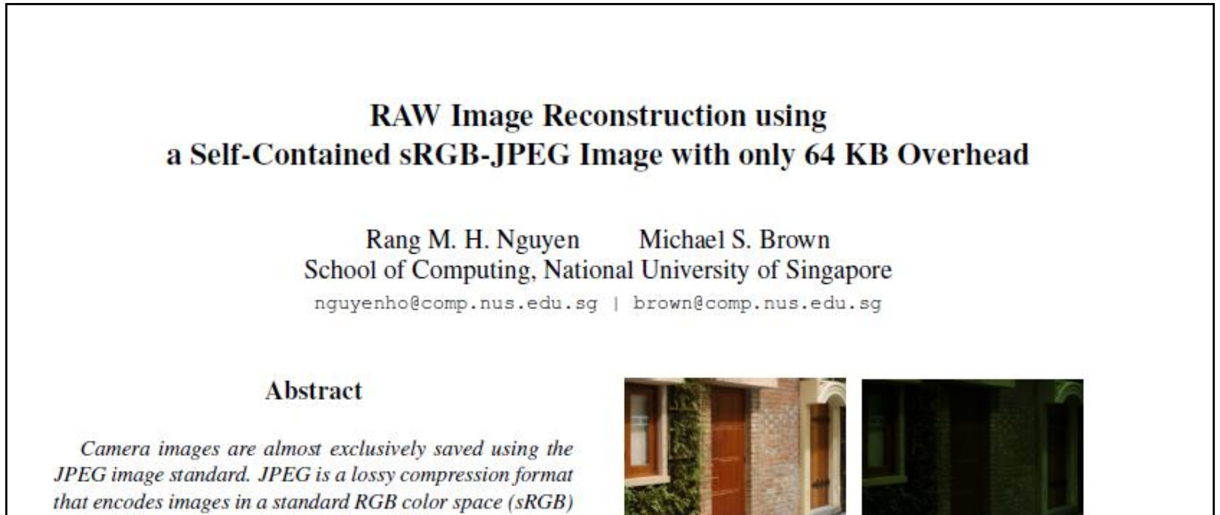


extract raw-RGB image when needed.

Hybrid image encoding



[CVPR'16]



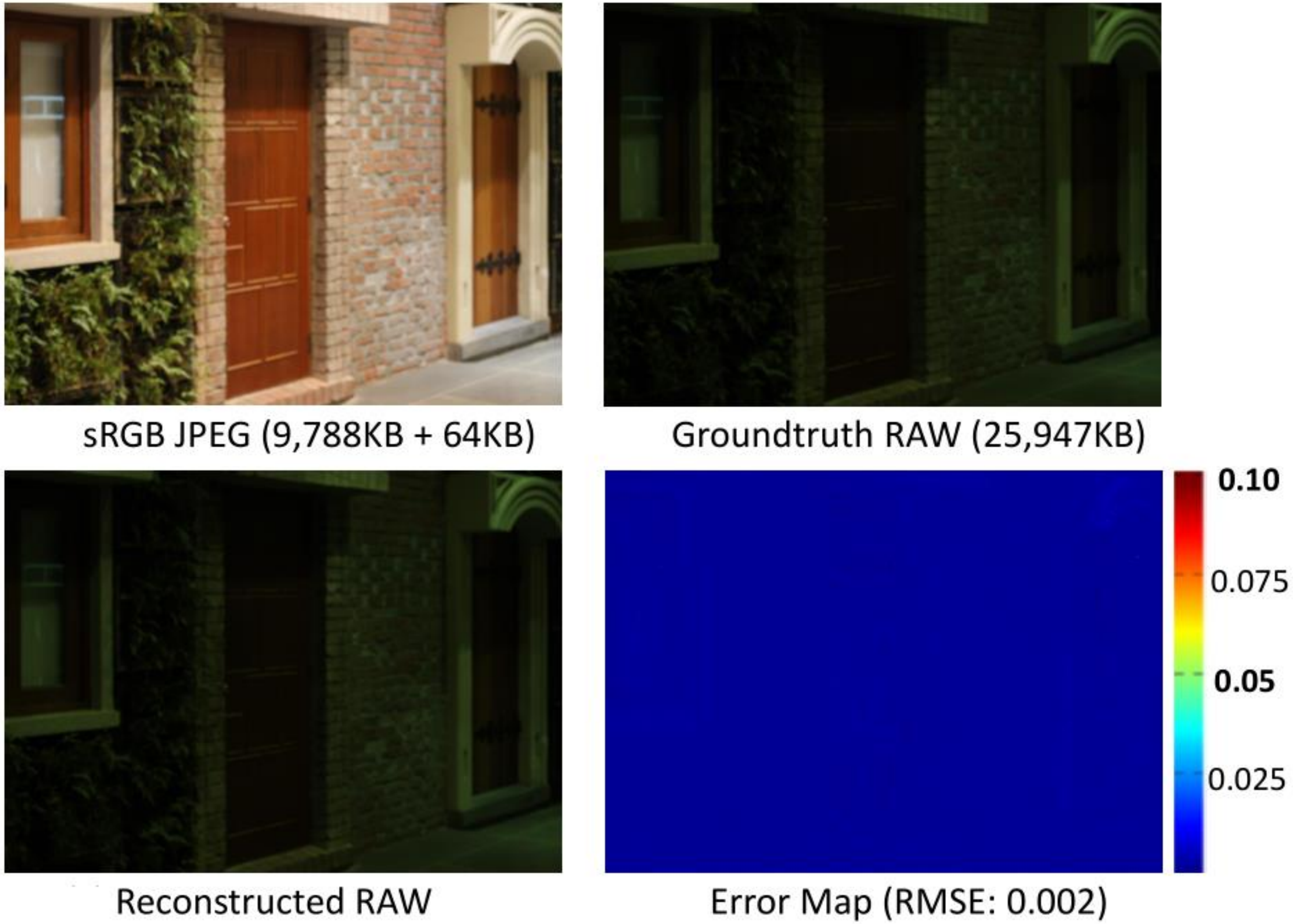
RAW Image Reconstruction using a Self-Contained sRGB-JPEG Image with only 64 KB Overhead

Rang M. H. Nguyen   Michael S. Brown
School of Computing, National University of Singapore
nguyenho@comp.nus.edu.sg | brown@comp.nus.edu.sg

Abstract

Camera images are almost exclusively saved using the JPEG image standard. JPEG is a lossy compression format that encodes images in a standard RGB color space (sRGB)

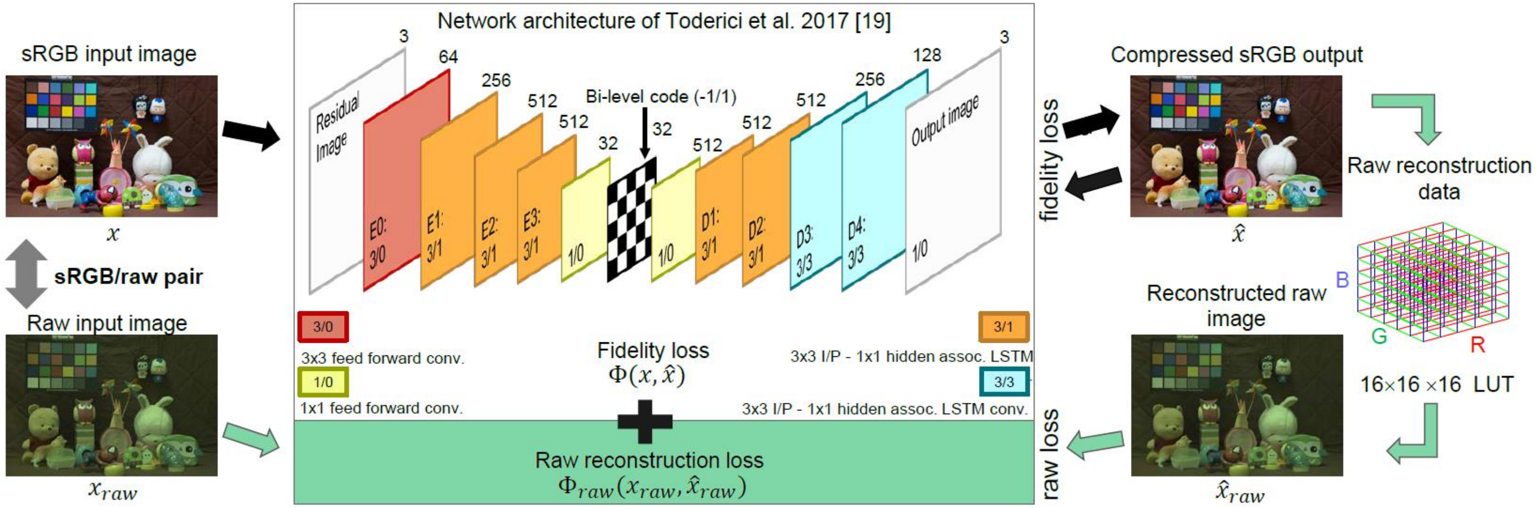
Hybrid image encoding



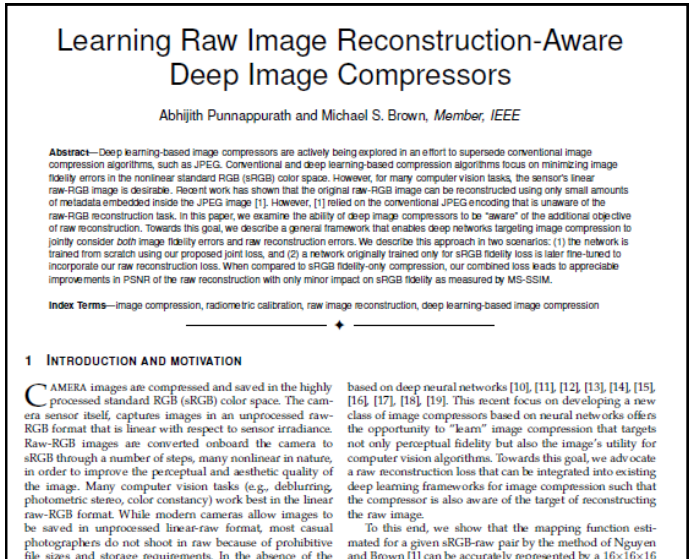
Hybrid image encoding application

Input | Ground truth | White-balancing on our reconstructed RAW | White-balancing on sRGB

Extended to deep-learning compression



TPAMI 2019



Learning Raw Image Reconstruction-Aware Deep Image Compressors

Abhijith Punnappurath and Michael S. Brown, *Member, IEEE*

Abstract—Deep learning-based image compressors are actively being explored in an effort to supersede conventional image compression algorithms, such as JPEG. Conventional and deep learning-based compression algorithms focus on minimizing image fidelity errors in the nonlinear standard RGB (sRGB) color space. However, for many computer vision tasks, the sensor's linear raw-RGB image is desirable. Recent work has shown that the original raw-RGB image can be reconstructed using only small amounts of metadata embedded inside the JPEG image [1]. However, [1] relied on the conventional JPEG encoding that is unaware of the raw-RGB reconstruction task. In this paper, we examine the ability of deep image compressors to be "aware" of the additional objective of raw reconstruction. Towards this goal, we describe a general framework that enables deep networks targeting image compression to jointly consider *both* image fidelity errors and raw reconstruction errors. We describe this approach in two scenarios: (1) the network is trained from scratch using our proposed joint loss, and (2) a network originally trained only for sRGB fidelity loss is later fine-tuned to incorporate our raw reconstruction loss. When compared to sRGB fidelity-only compression, our combined loss leads to appreciable improvements in PSNR of the raw reconstruction with only minor impact on sRGB fidelity as measured by MS-SSIM.

Index Terms—image compression, radiometric calibration, raw image reconstruction, deep learning-based image compression

1 INTRODUCTION AND MOTIVATION

CAMERA images are compressed and saved in the highly processed standard RGB (sRGB) color space. The camera sensor itself, captures images in an unprocessed raw-RGB format that is linear with respect to sensor irradiance. Raw-RGB images are converted onboard the camera to sRGB through a number of steps, many nonlinear in nature, in order to improve the perceptual and aesthetic quality of the image. Many computer vision tasks (e.g., deblurring, photometric stereo, color constancy) work best in the linear raw-RGB format. While modern cameras allow images to be saved in unprocessed linear-raw format, most casual photographers do not shoot in raw because of prohibitive

based on deep neural networks [10], [11], [12], [13], [14], [15], [16], [17], [18], [19]. This recent focus on developing a new class of image compressors based on neural networks offers the opportunity to "learn" image compression that targets not only perceptual fidelity but also the image's utility for computer vision algorithms. Towards this goal, we advocate a raw reconstruction loss that can be integrated into existing deep learning frameworks for image compression such that the compressor is also aware of the target of reconstructing the raw image.

To this end, we show that the mapping function estimated for a given sRGB-raw pair by the method of Nguyen

AI-methods will replace JPEG. This gives us an opportunity to develop new deep-learning compression schemes that incorporate in the need to RAW or linear data.

Tutorial schedule

- Part 1 (General)
  - ~~Motivation~~
  - ~~Review of color & color spaces~~
  - ~~Overview of in-camera imaging pipeline~~

1.30pm – 3.30pm

Break
3.30pm – 4.30pm

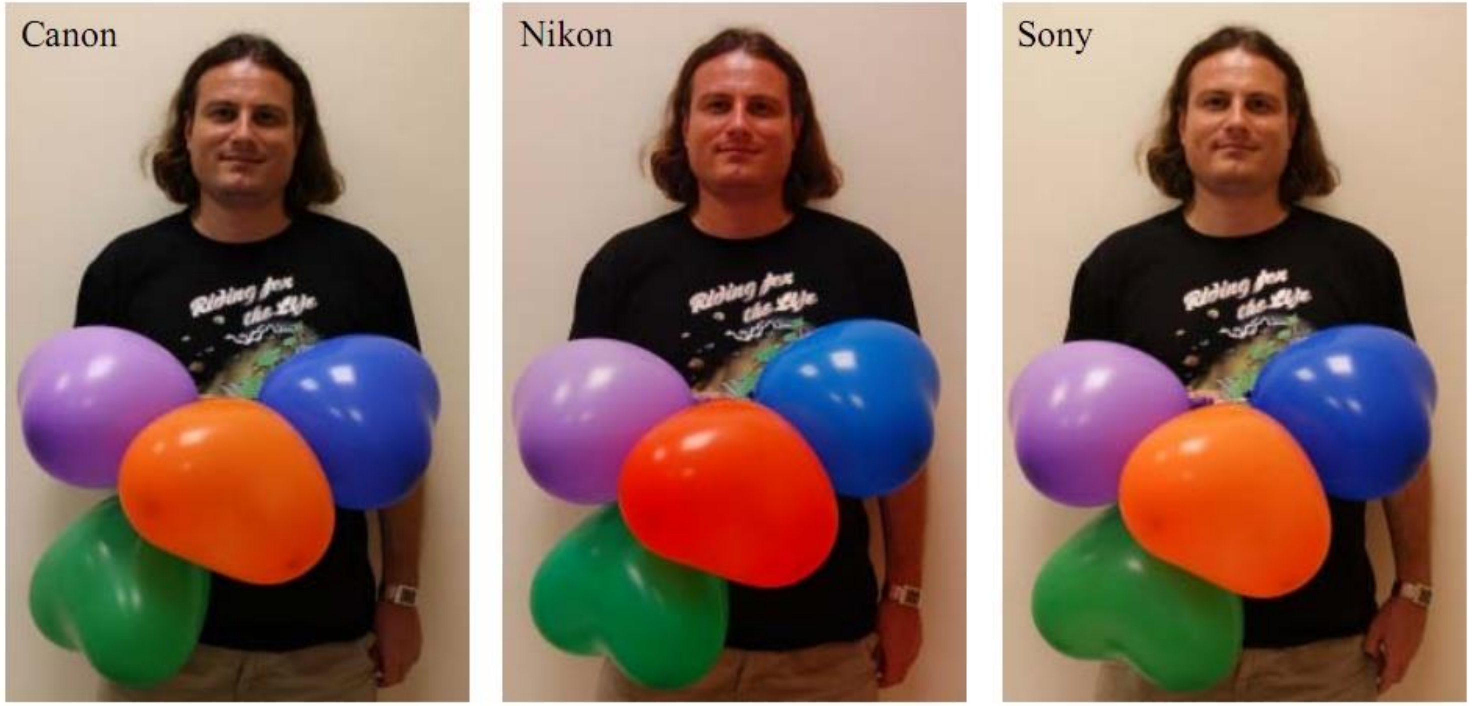
- Part 2 (Imaging and Computer Vision)
  - ~~Misconceptions in the computer vision community regarding color~~
  - ~~Recent work on color and cameras~~
  - Concluding remarks

4.30pm – 6.00pm

Part 2: Concluding remarks

"Take away" messages

1) Consumer cameras do not attempt to reproduce accurate color!



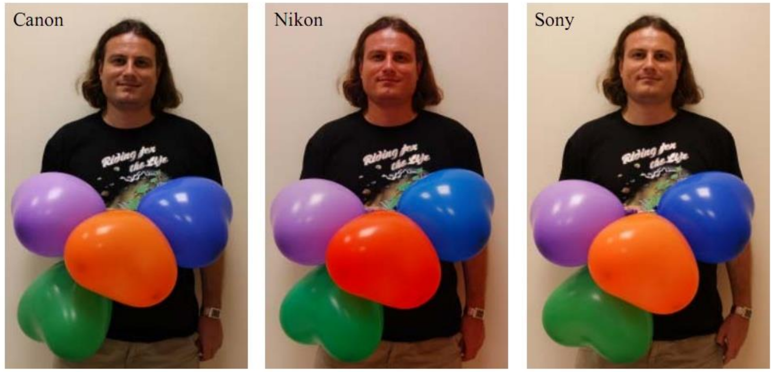
"Take away" messages

2) sRGB is a standard color pace, but colors are not accurate with respect to the physical world.

Remember, sRGB values have been manipulated by your camera hardware and do not represent the physical world, they represent the "photo"!

sRGB output on the same camera, different photo options!

sRGB output on different cameras.

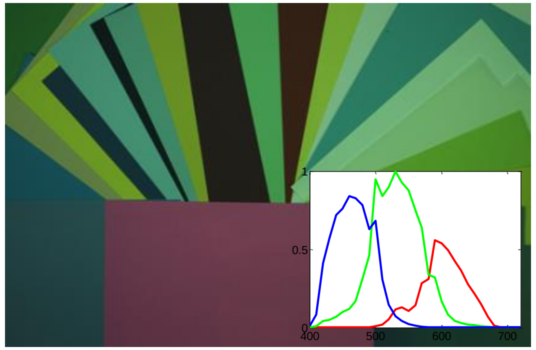


"Take away" messages

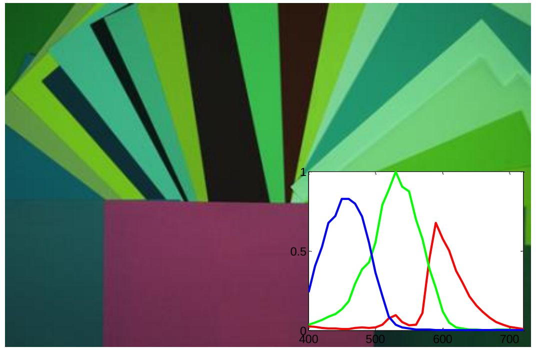
3) raw-RGB images are linear with respect to irradiance

If you need to measure the physical world, capture raw-RGB images.

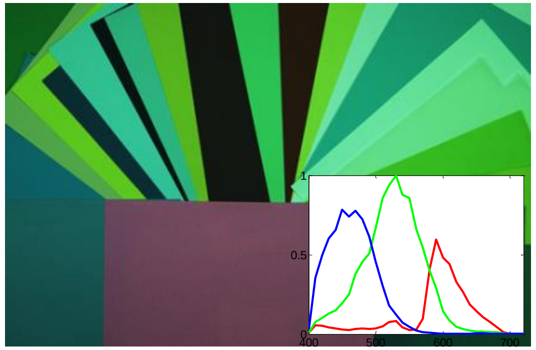
However, keep in mind that raw-RGB images are not standard between devices! The raw-RGB is device specific.



Canon 1D

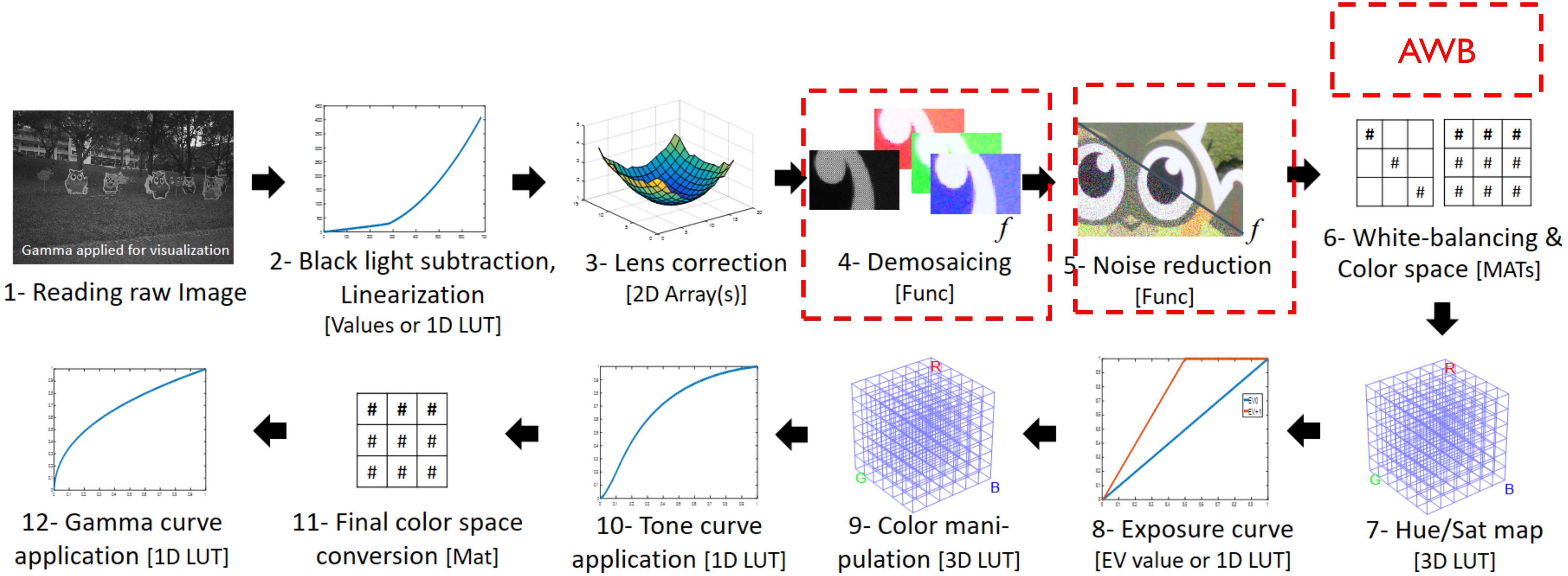


Nikon D40



Sony $\alpha 57$

Many opportunities to improve the camera pipeline



Last slide (almost)

- I hope you have learned more about color and the in-camera processing pipeline.
- I hope you have a better idea of which color spaces are more suitable for different computer vision tasks.
- I encourage you in your papers to denote your assumptions about your image's color space in your research papers:

  For example, replace this: "Our input is an RGB image …"
  to: "Our input is an RGB image encoded in standard RGB…"

  - Such a small clarification in your paper will greatly help other researchers.

Thank you for attending

Special thanks to my students and colleagues for contributing images, code, and materials to this tutorial.

Mahmoud
Afifi

Abhijith
Punnappurath

Abdelrahman
Abdelhamed

Hakki C.
Karaimer

Abdullah
Abuolaim

Li Yu

Rang Nguyen

Cheng Dongliang

Understanding color perception

Hermann Grassmann

Johannes von Kries

W. David Wright

James Clerk Maxwell

Johann von Goethe

Thomas Young

Digital cameras

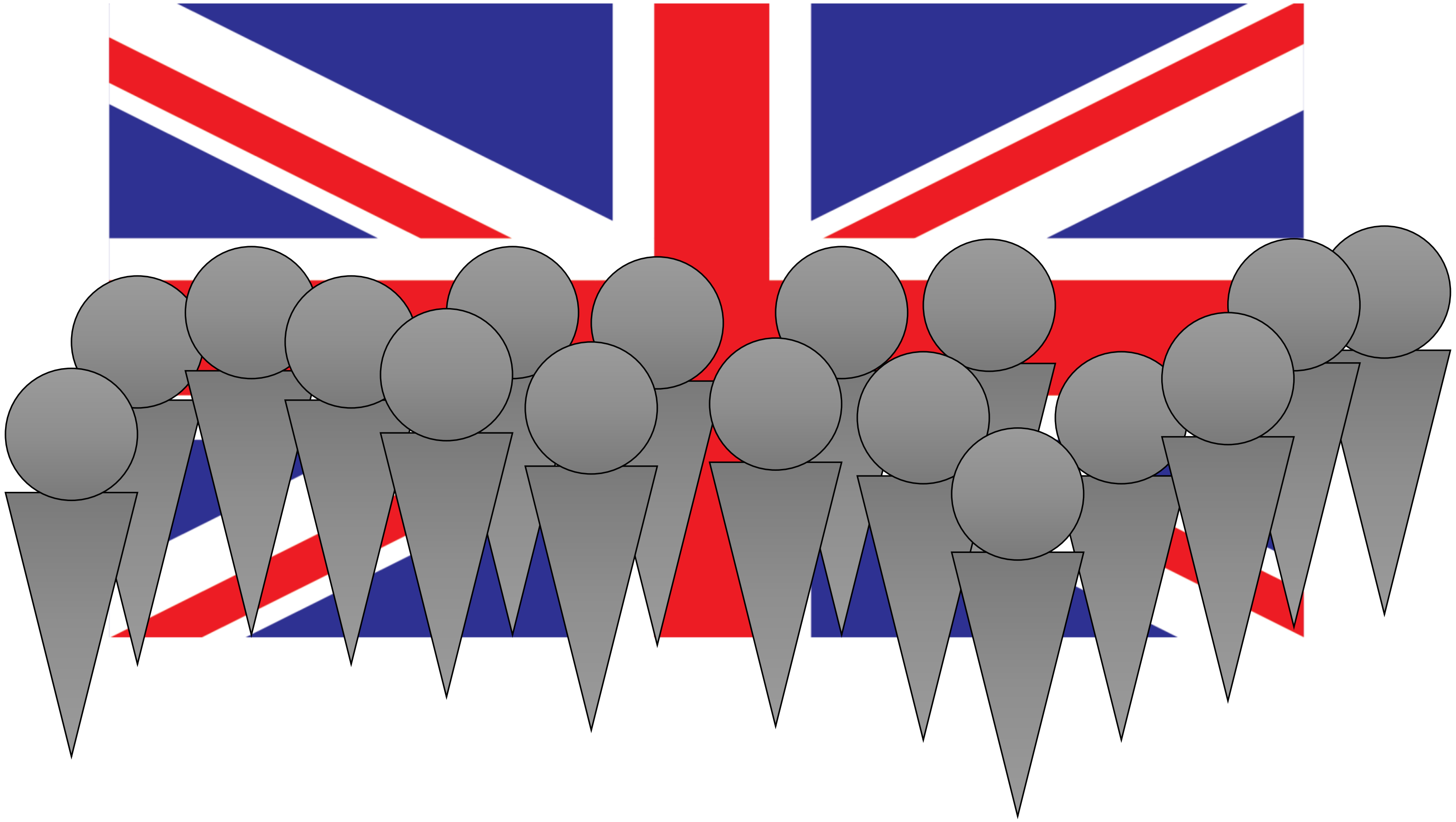
Bryce E. Bayer
(Color Filter Arrays)

Willard Boyle and George Smith
(Nobel Prize for inventing the CCD)
Photo: Reuters

Eric R. Fossum
(Invented CMOS)

Steven Sasson
(Attributed with building the first digital camera)

And of course…



"The Standard Observers"