COLOR

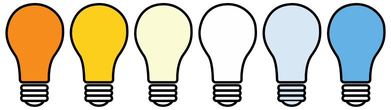
Color – Basics

COLOR PERCEPTION OF THE HUMAN VISUAL SYSTEM

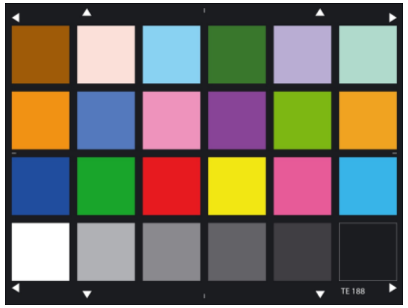
Color – Human perception

What is color?

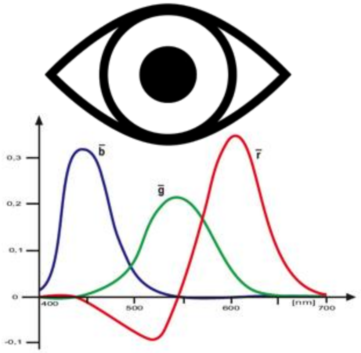
- A subjective perception
- This perception is formed by three components:



Light
spectral distribution

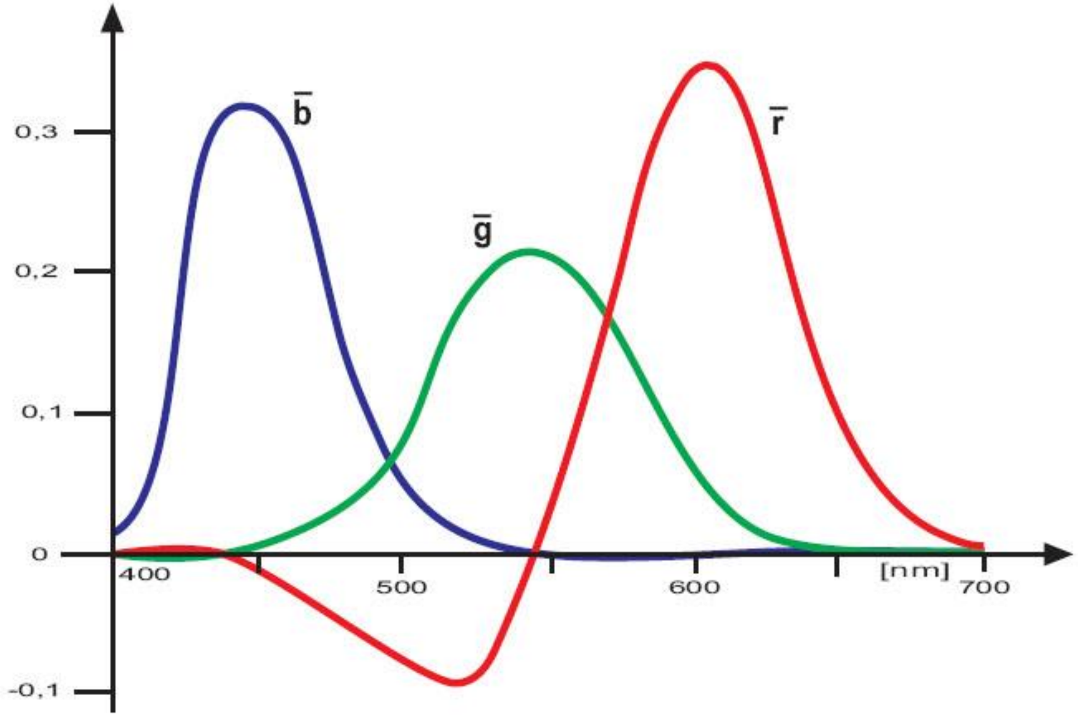


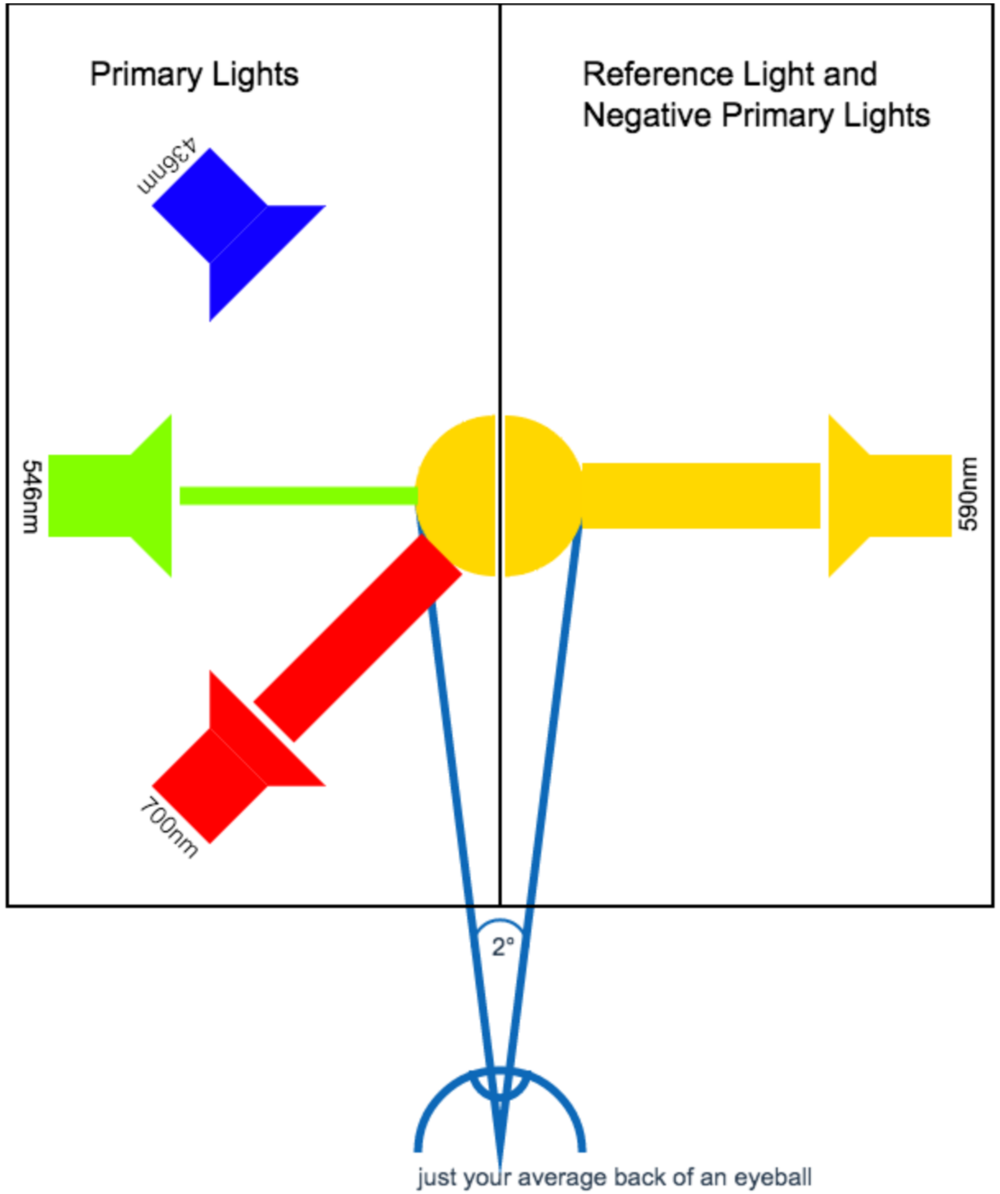
Object
spectral reflection

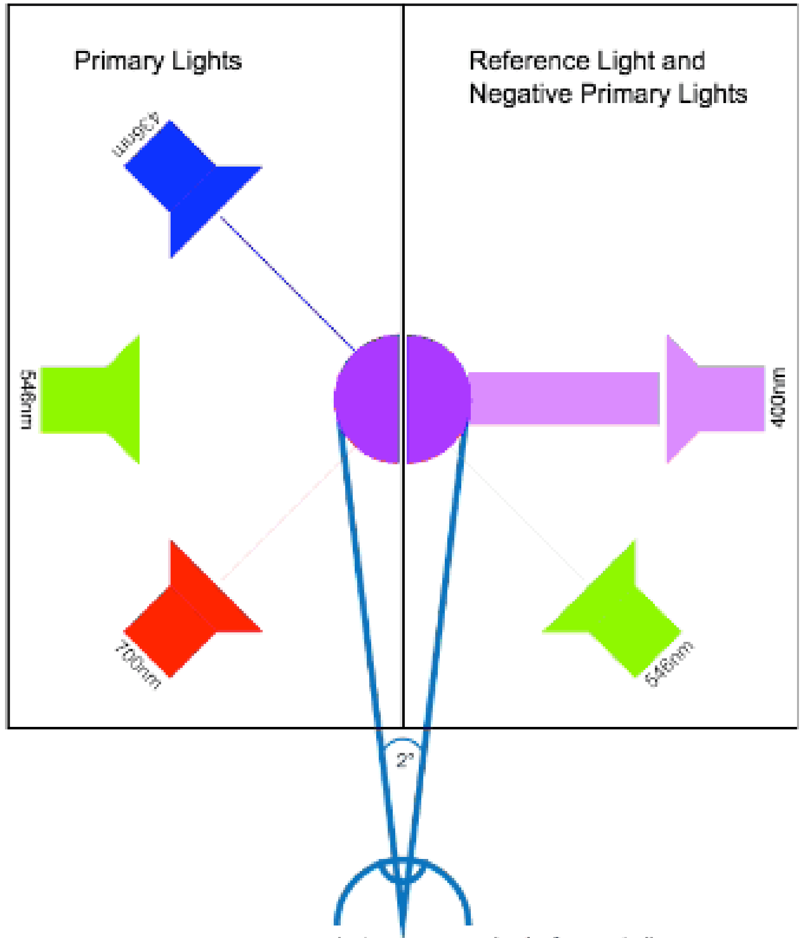


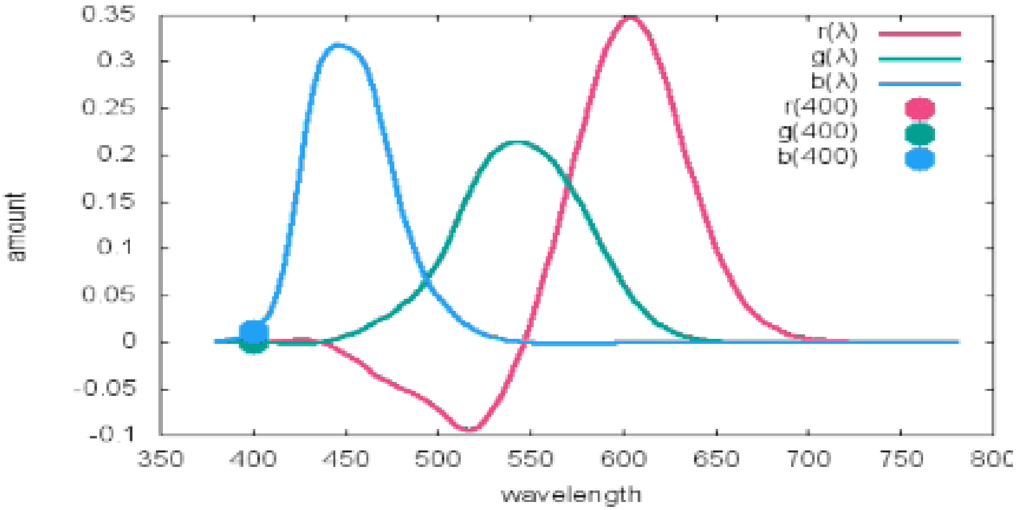
Eye
spectral sensitivity

The three primary valences $\bar{r}, \bar{g}, \bar{b}$ are used to simulate the spectral sensitivity of the human eye.




Primary Lights
Reference Light and
Negative Primary Lights
436nm
546nm
700nm
590nm
2°
just your average back of an eyeball



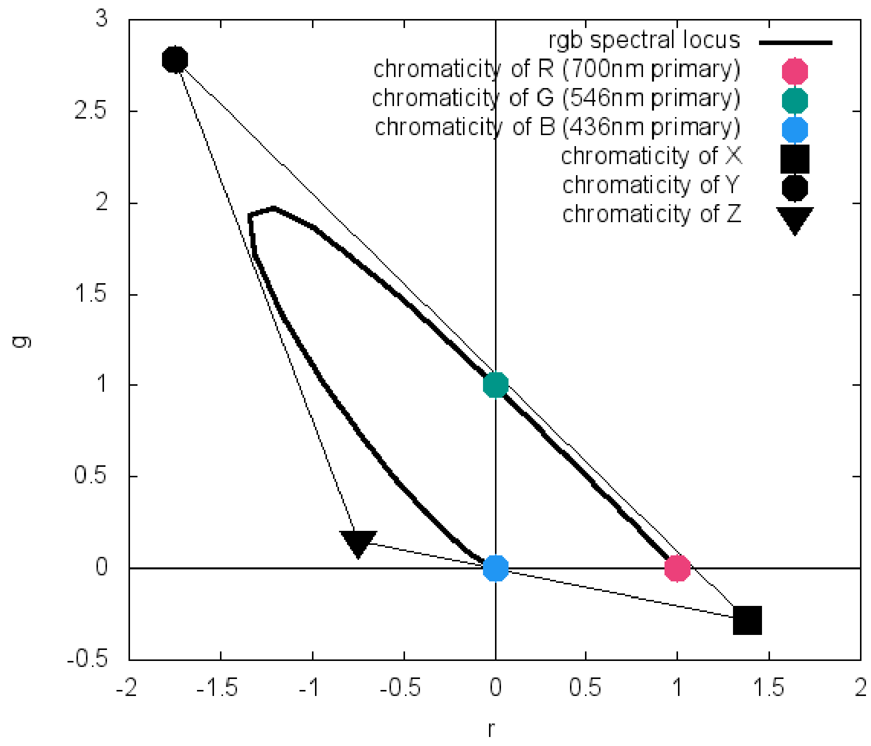


https://medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a

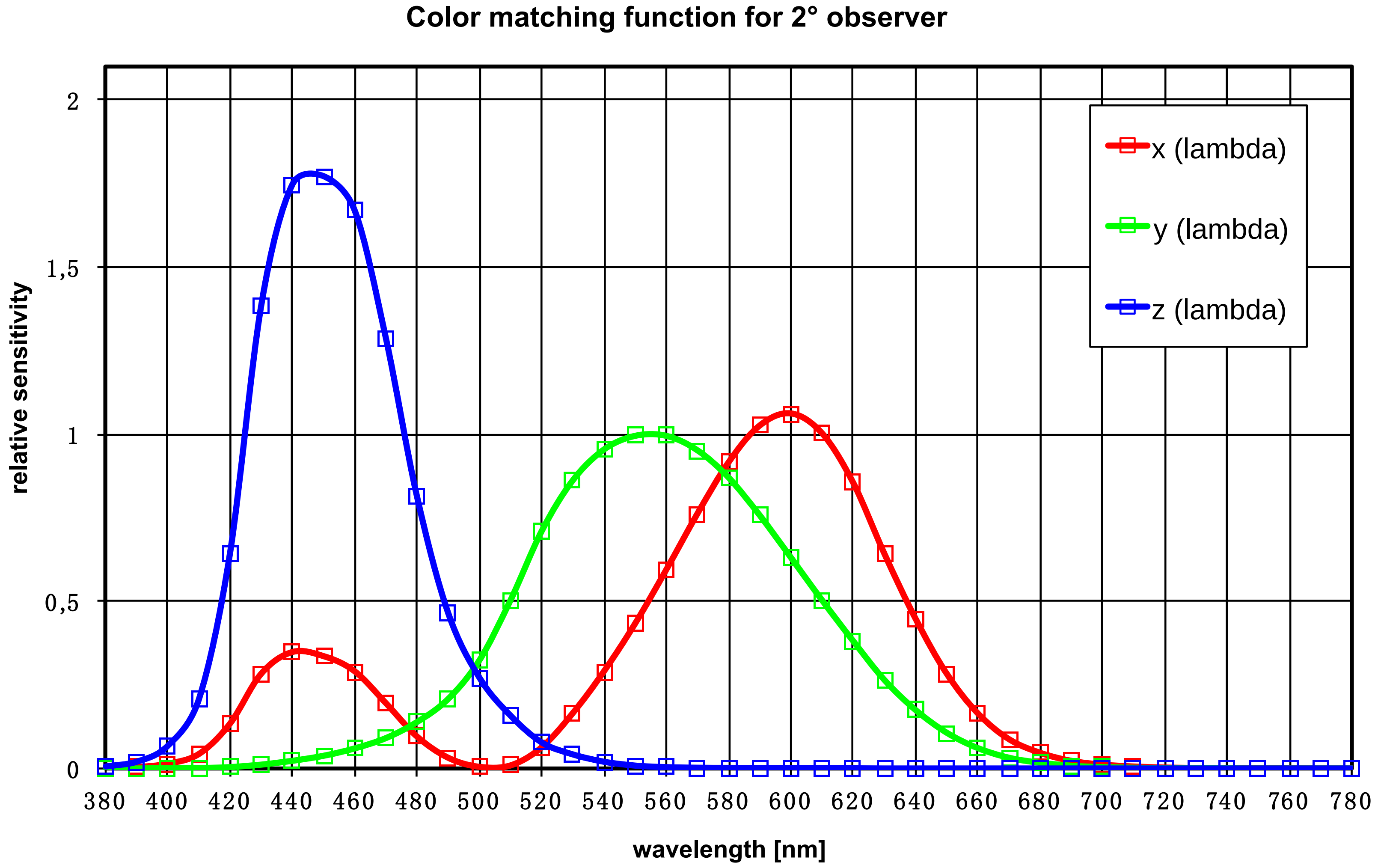
```
Matrix coefficents that convert the RGB space into one where the
carefully chosen XYZ primaries exist at coordinates (1,0) (0,1) and
(0,0) in the new chromaticity space.

[X]   | 2.768 1.751 1.130|   [R]
[Y] = | 1.000 4.590 0.060| * [G]
[Z]   | 0     0.056 5.594|   [B]


source: Colorimetry by Janos Schanda, page 30
```



➔ Transform to XYZ valences

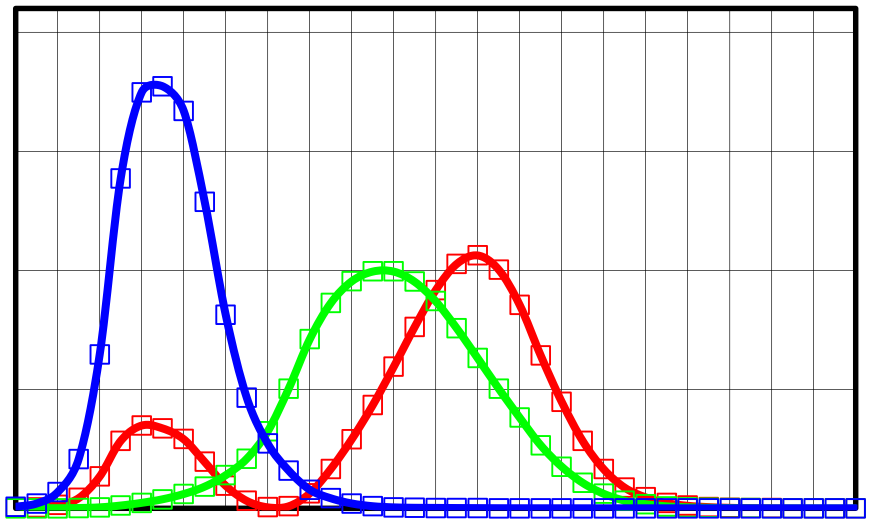


Color – Human perception

$$X = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot \bar{x}(\lambda) \cdot d\lambda$$

$$Y = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot \bar{y}(\lambda) \cdot d\lambda$$

$$Z = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot \bar{z}(\lambda) \cdot d\lambda$$



with:

s(λ) being the spectral distribution of the light source

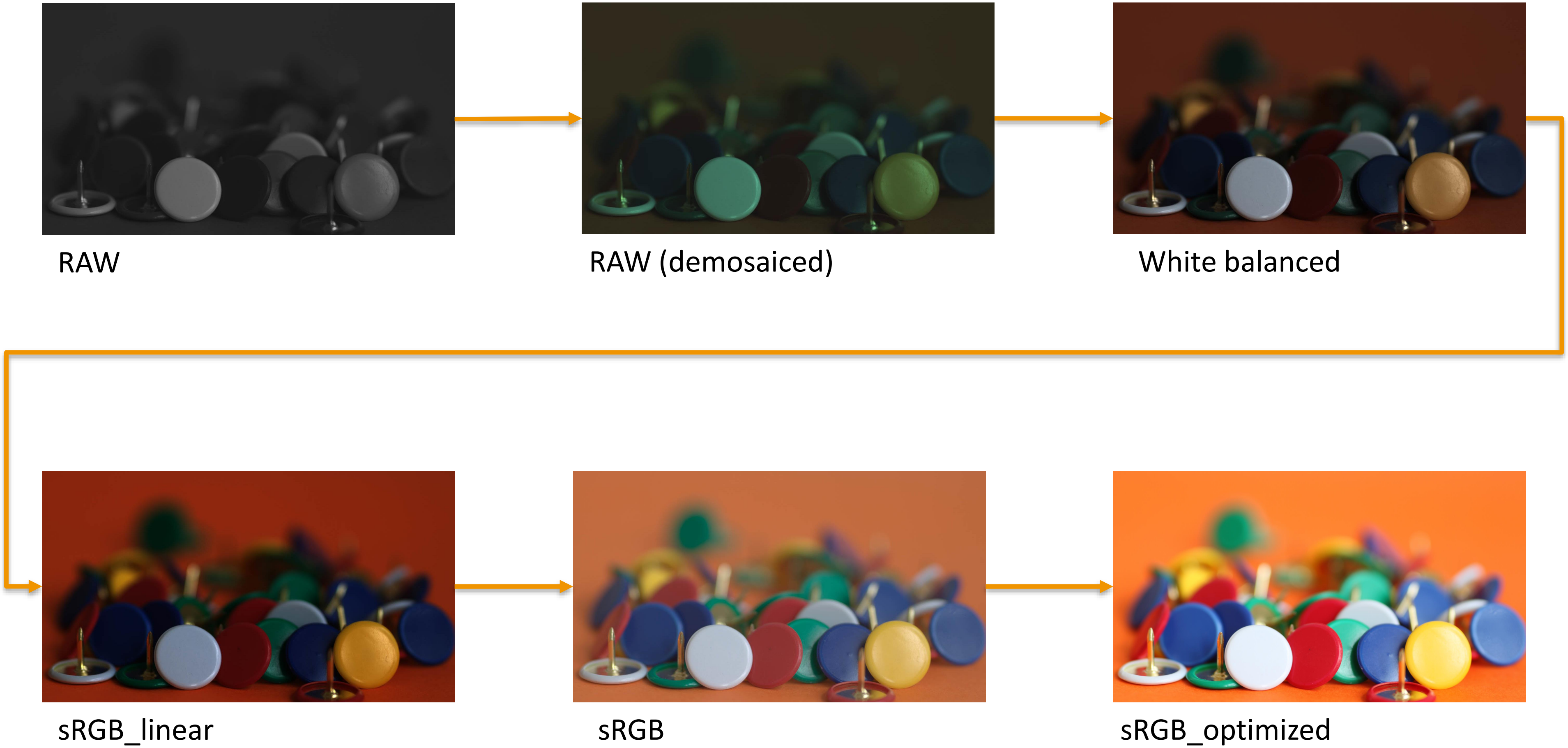
r(λ) the spectral reflectance of the object

x, y, z(λ) the color matching functions of the human eye

Color – Basics

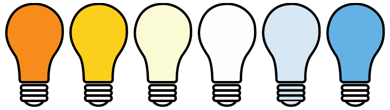
COLOR RENDERING IN A CAMERA

Color

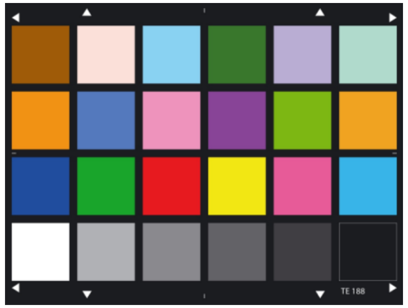

RAW
RAW (demosaiced)
White balanced
sRGB_linear
sRGB
sRGB_optimized

Color rendering of a camera

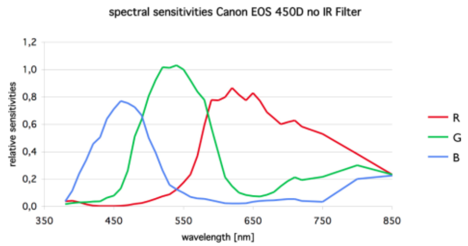
RGB values created by the camera



X



X



= RGB

Light source
spectral distribution

Chart / Object
spectral reflection

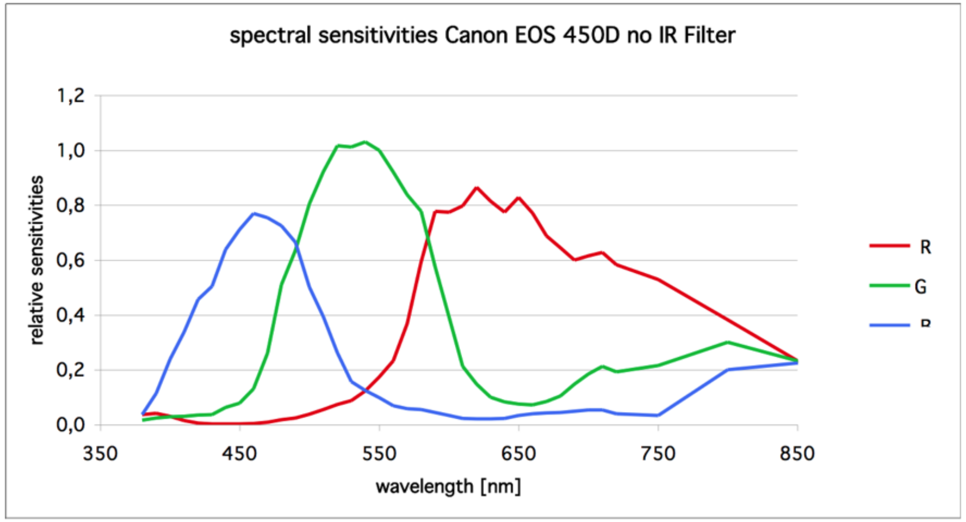
Camera
spectral sensitivity

Color correction matrix

$$R_{camera} = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot c_R(\lambda) \cdot d\lambda$$

$$G_{camera} = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot cG(\lambda) \cdot d\lambda$$

$$B_{camera} = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot cB(\lambda) \cdot d\lambda$$



with:

$s(\lambda)$ spectral distribution of the light source

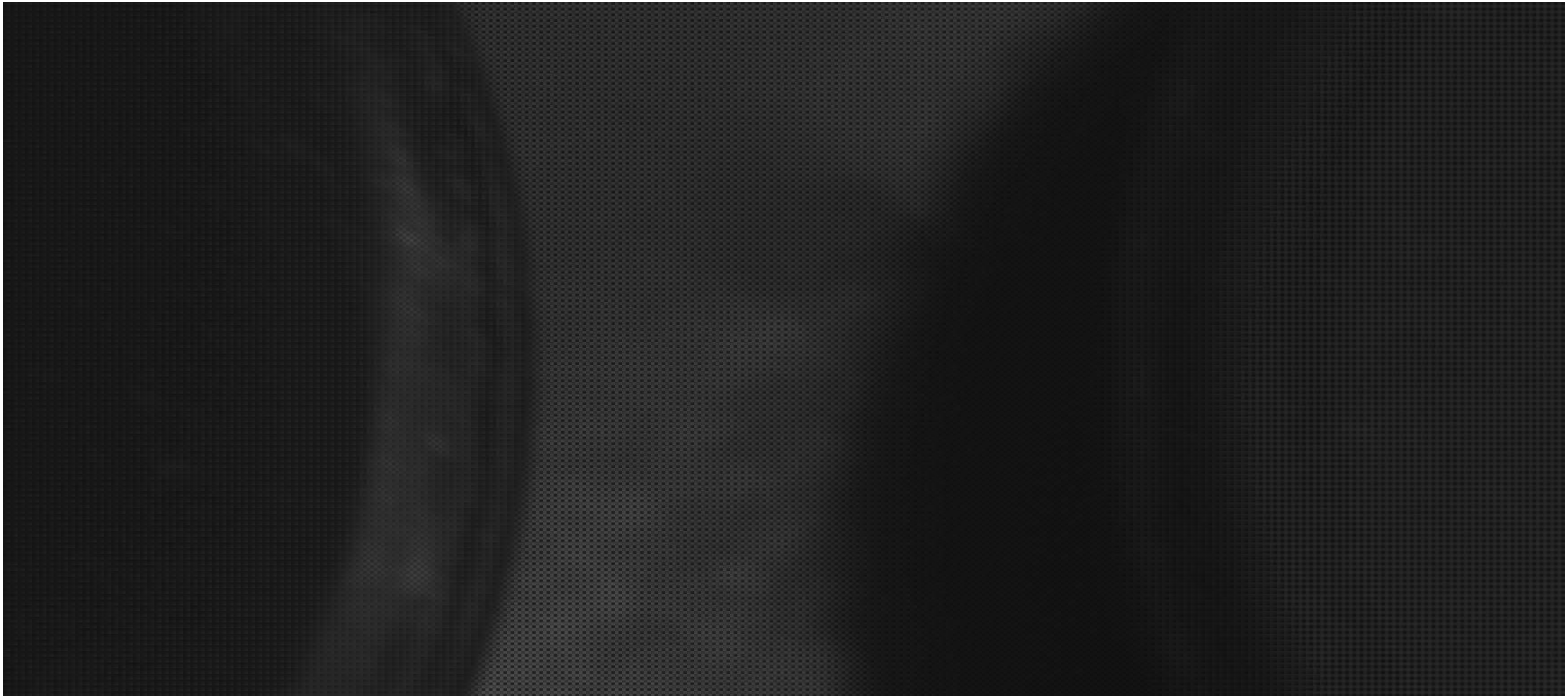
$r(\lambda)$ spectral reflectance of the object

$c_X(\lambda)$ spectral sensitivity of the camera

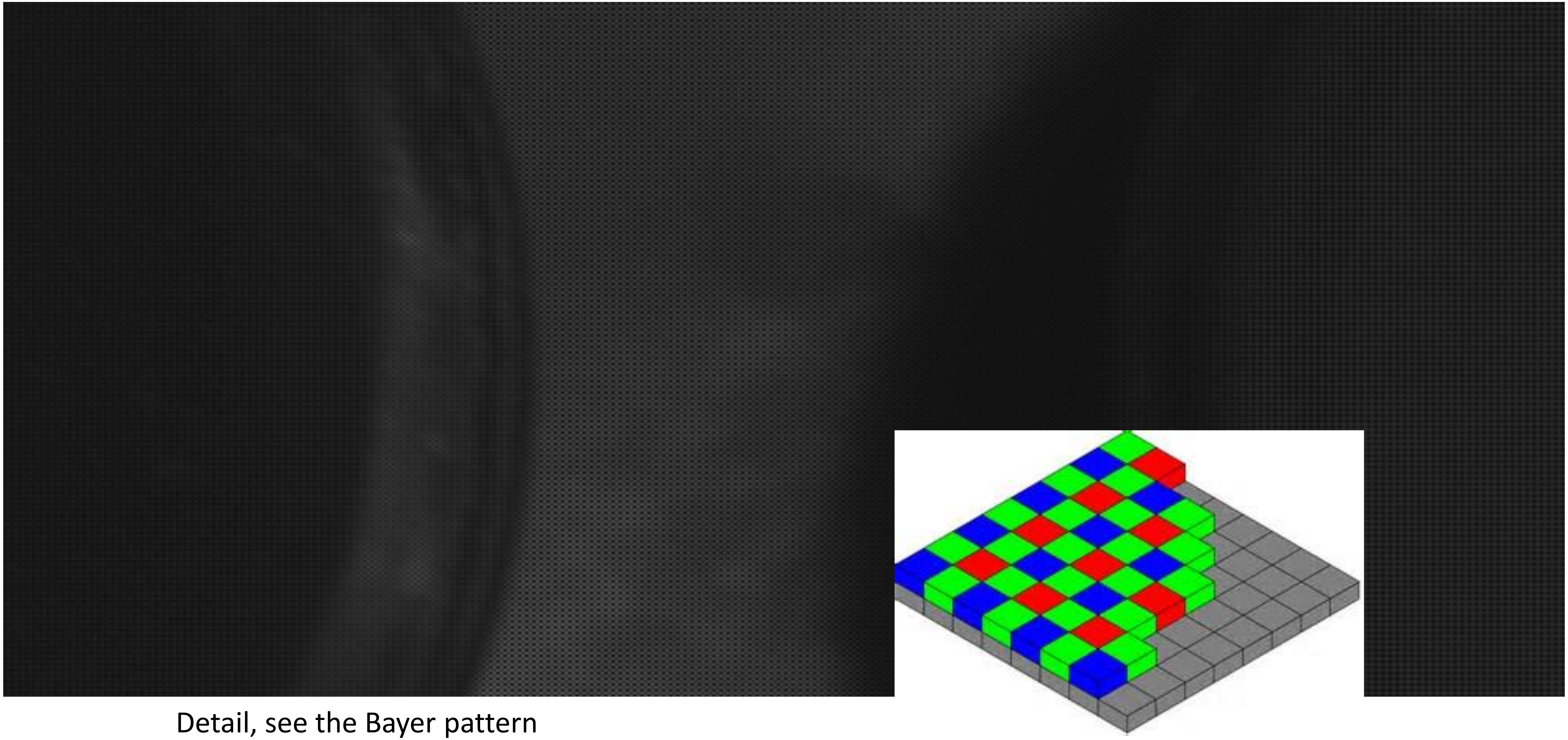
RAW

RAW image (visualized to 8bit)

RAW - Detail

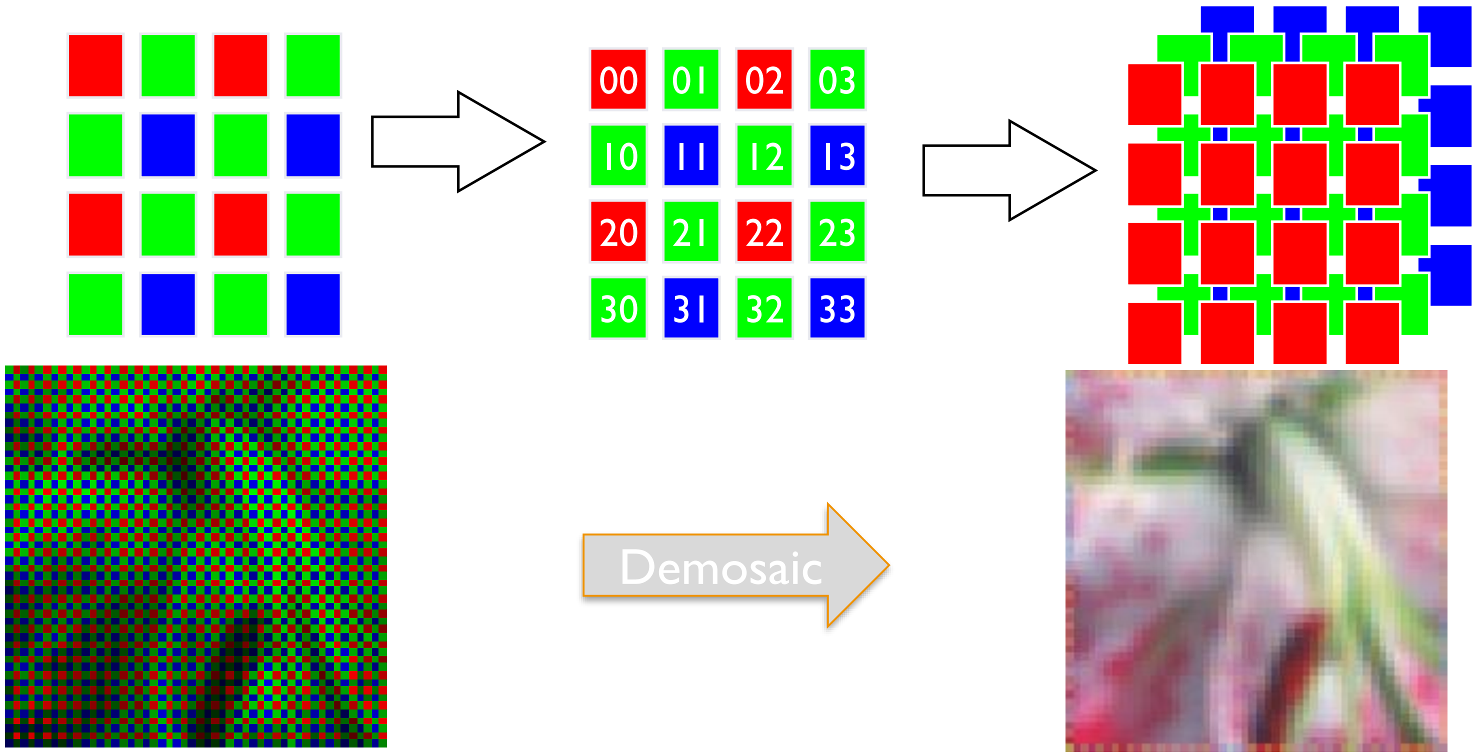


Detail



Detail, see the Bayer pattern

Demosaic



- The demosaic algorithm takes the raw R,G,B pixel data and interpolates the missing colors for each of the pixels.
- Computationally intensive and important to overall image quality
- Algorithms are closely guarded secrets
- Good algorithms
  - Sharp
  - Free from Artifacts
  - Visually plausible fakes for pixel colors not sampled
  - Doesn’t amplify noise

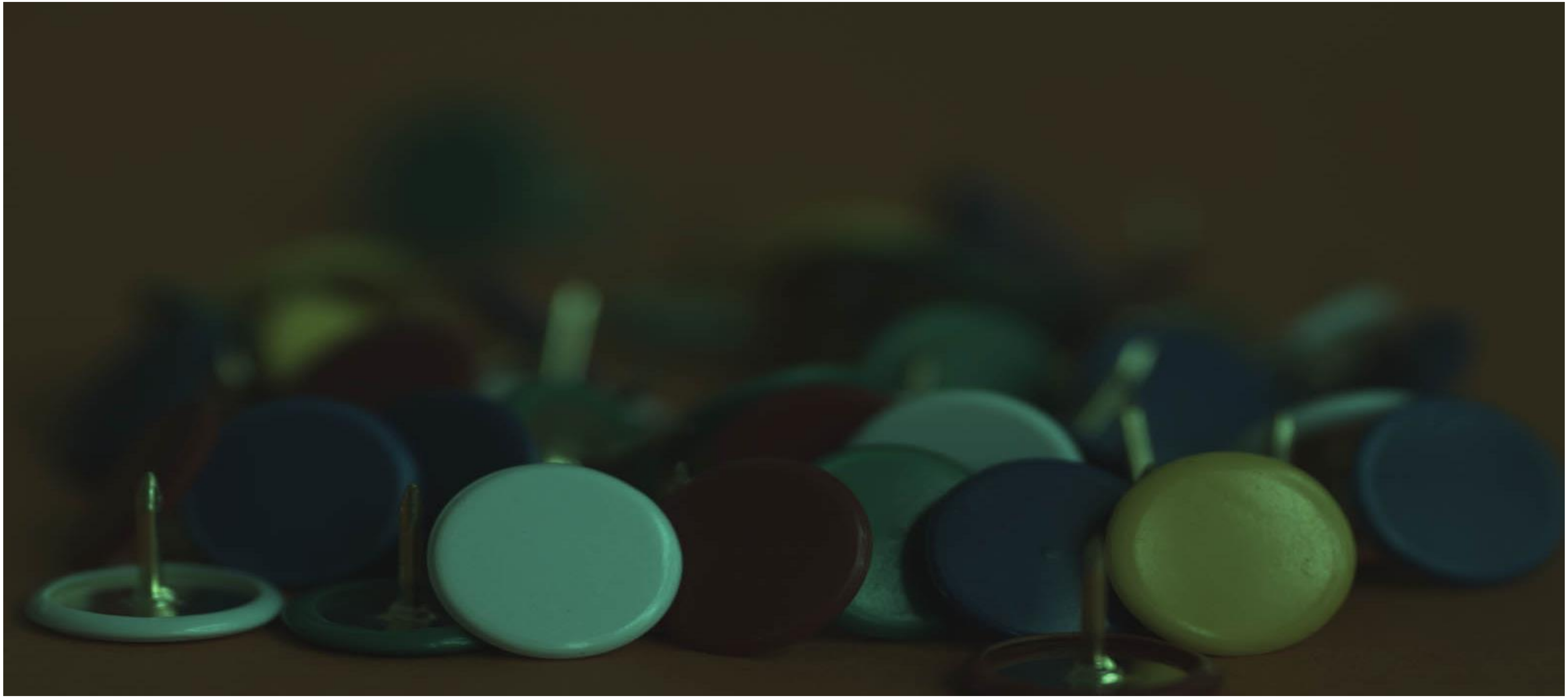
Comparison of Bilinear and Edge-Directed

Bilinear

Edge Directed

Images courtesy R. Kimmel

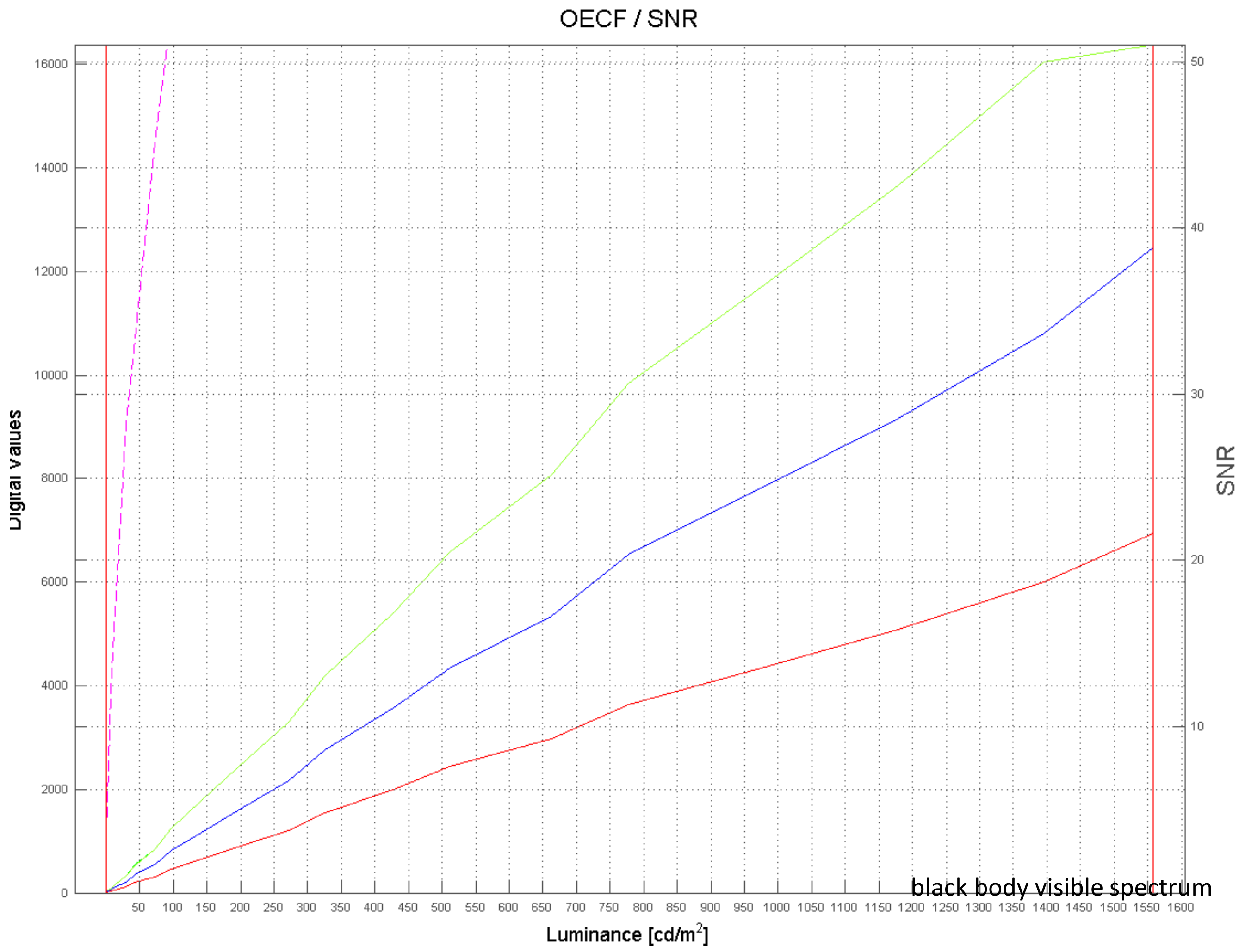
Demosaiced



RGB image, created from RAW

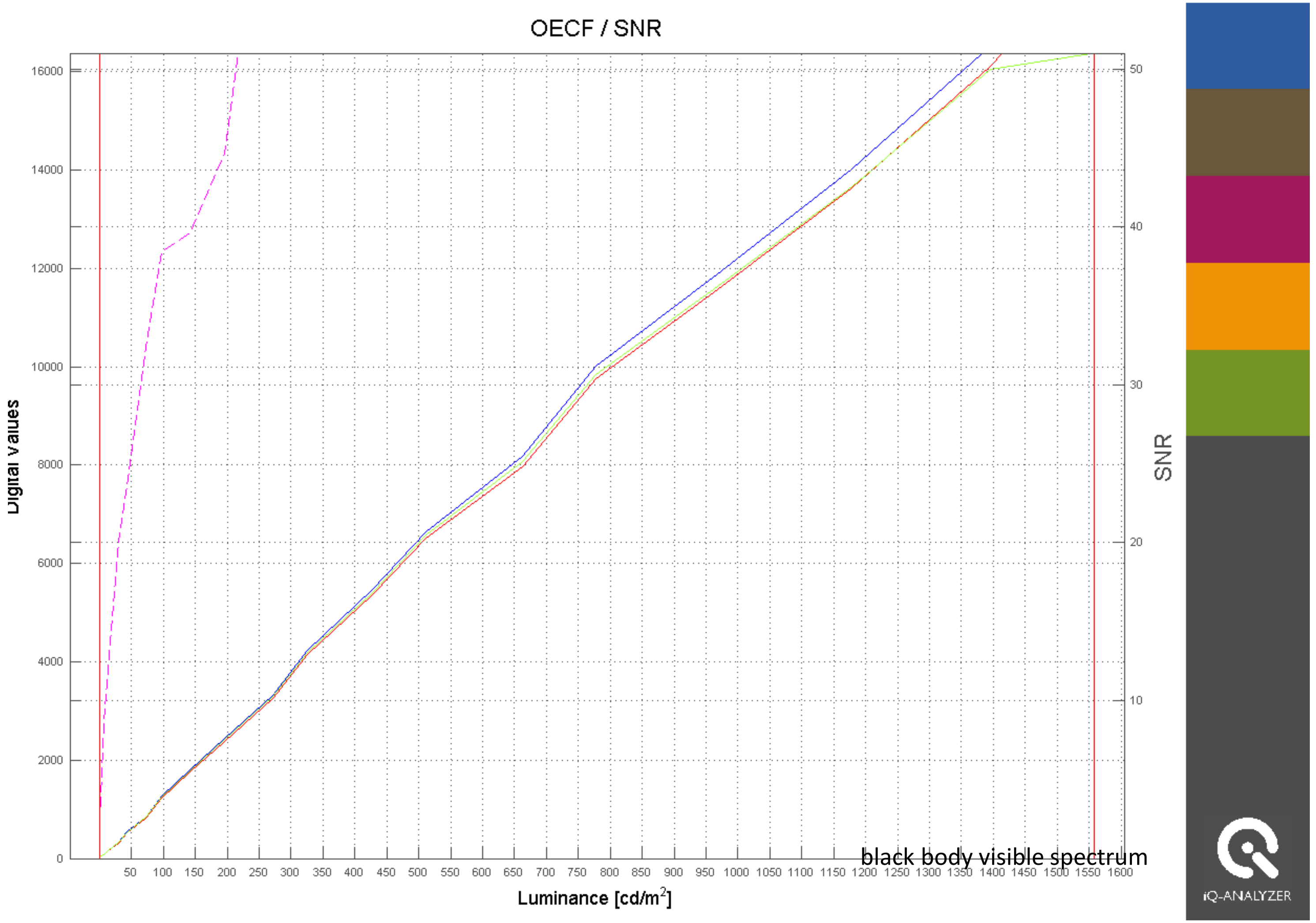
Displaying the OECF (ISO 14524)

linear



Linear – no whitebalance

Displaying the OECF (ISO 14524)



black body visible spectrum

linear

Linear – whitebalance

AWB algorithms

- Brightness-value qualification
  - Certain illuminants unlikely to occur at certain scene brightness values
- Gray-world color constancy
  - Correct data to an average gray in the image
- Maximum RGB
  - assumes the scene contains a white lambertian surface
- Retinex
  - Estimates illuminant by maximum response of each channel
- Color by correlation
  - Establish a correlation statistic under which illuminants given image colors are possible.

- Others based on scene and camera information…

Assumption:

Certain illuminants can only occur at certain brightness values

Pro: Algorithm generally works well

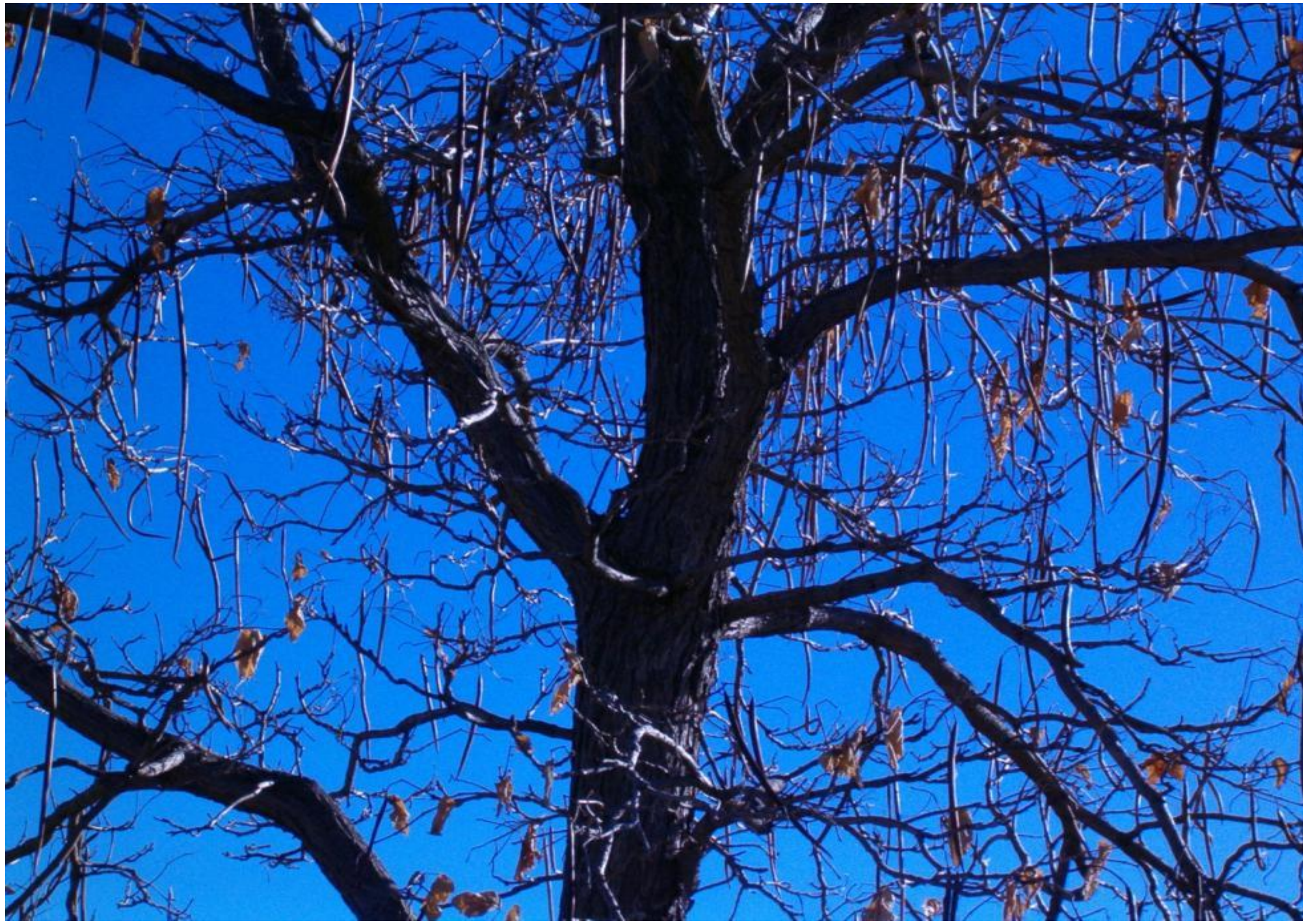
Con: May fail in certain daylight conditions

Typical BV Qualification Table

Illuminant 0 Name = D75
Illuminant 1 Name = D65
Illuminant 2 Name = D55
Illuminant 3 Name = D50
Illuminant 4 Name = D45
Illuminant 5 Name = D40 and so on…

Illuminant 0 BV Range = { 2, 6 } --> this means allow 2 <= BV <= 6
Illuminant 1 BV Range = { -128, 8 }
Illuminant 2 BV Range = { -128, 127.9 }
Illuminant 3 BV Range = { -128, 127.9 }
Illuminant 4 BV Range = { -128, 5.01 }
Illuminant 5 BV Range = { -128, 5.01 }
Illuminant 6 BV Range = { -128, 5.01 }
Illuminant 7 BV Range = { -128, 5.01 }
Illuminant 8 BV Range = { -128, 5.01 }
Illuminant 9 BV Range = { -128, 5.01 }
Illuminant 10 BV Range = { -128, 5.01 }
Illuminant 11 BV Range = { -128, 5.01 }
Illuminant 12 BV Range = { -128, 5.01 }
Illuminant 13 BV Range = { -128, 5.01 }

BV Qualification failure



It is daylight, but the object will require long exposure and is therefore not qualified as daylight.

Gray World

Assumption:

Scenes contain a variety of colors. An average of all the colors is gray.

- One of the oldest white balance algorithms

Pro: Very simple to implement. In simplest form, doesn't require camera calibration

Con: Vivid colors can skew the average resulting in an estimate that is the complement of the color.

What does gray mean?

"Gray-World failure"

The purple background shifts the average.
If neutral is set to the average, then flesh goes green.

Photo courtesy J. Holm

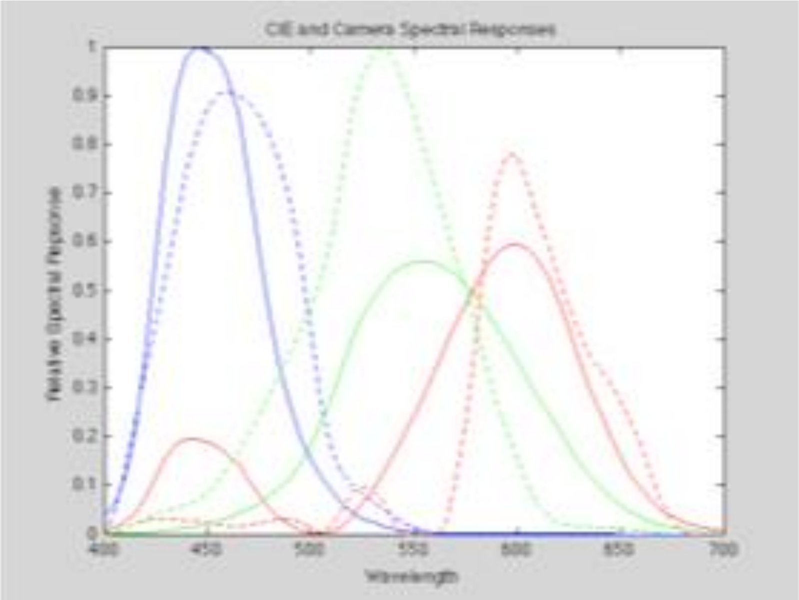
- Another venerable white-balance algorithm
- Based on theory that brightest pixel represents a white object reflecting the illuminant source.
- Pro: Can help find a white object in an otherwise pastel scene.
- Con: Fails when brightest object is not white or when there are clipped pixels.

Max RGB Failure

White Balanced

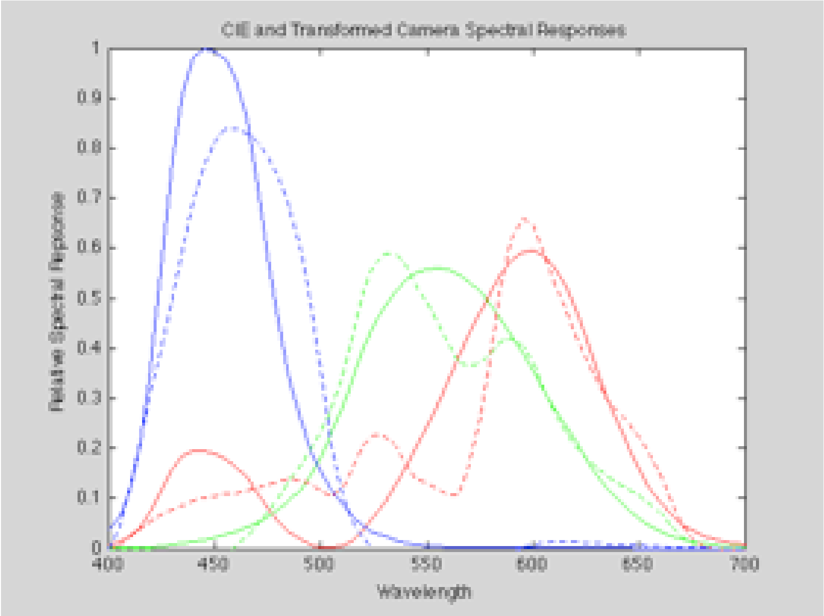
White balanced image

The Luther Condition



Transform

M



- Camera spectral responses are not generally a linear combination of CIE color matching functions and therefore fail the Luther-Maxwell-Ives condition
- Failing the Luther condition will necessarily tradeoff the reproduction of some colors in favor of others using any linear camera colorimetric transform
- Highly dimensional color transforms can improve color matching performance somewhat

Color correction matrix

The most simple form is a 3x3 matrix – the color correction matrix

$$\underbrace{\underbrace{\begin{bmatrix} R_{camera} \\ G_{camera} \\ B_{camera} \end{bmatrix}}_{\text{Sensor-output}} \underbrace{\begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}}_{\text{Color correction matrix}}}_{\text{device dependent}} = \underbrace{\underbrace{\begin{bmatrix} R_{sRGB} \\ G_{sRGB} \\ B_{sRGB} \end{bmatrix}}_{\text{sRGB}}}_{\text{device independent}}$$

Color correction matrix

More advanced systems use Multidimensional Look-up-Tables (MLUT)

$$\underbrace{\begin{bmatrix} R_{camera} \\ G_{camera} \\ B_{camera} \end{bmatrix} MLUT}_{\text{device dependent}} = \underbrace{\begin{bmatrix} R_{sRGB} \\ G_{sRGB} \\ B_{sRGB} \end{bmatrix}}_{\text{device independent}}$$

Sensor-output

Multidimensional Look-up-table

sRGB

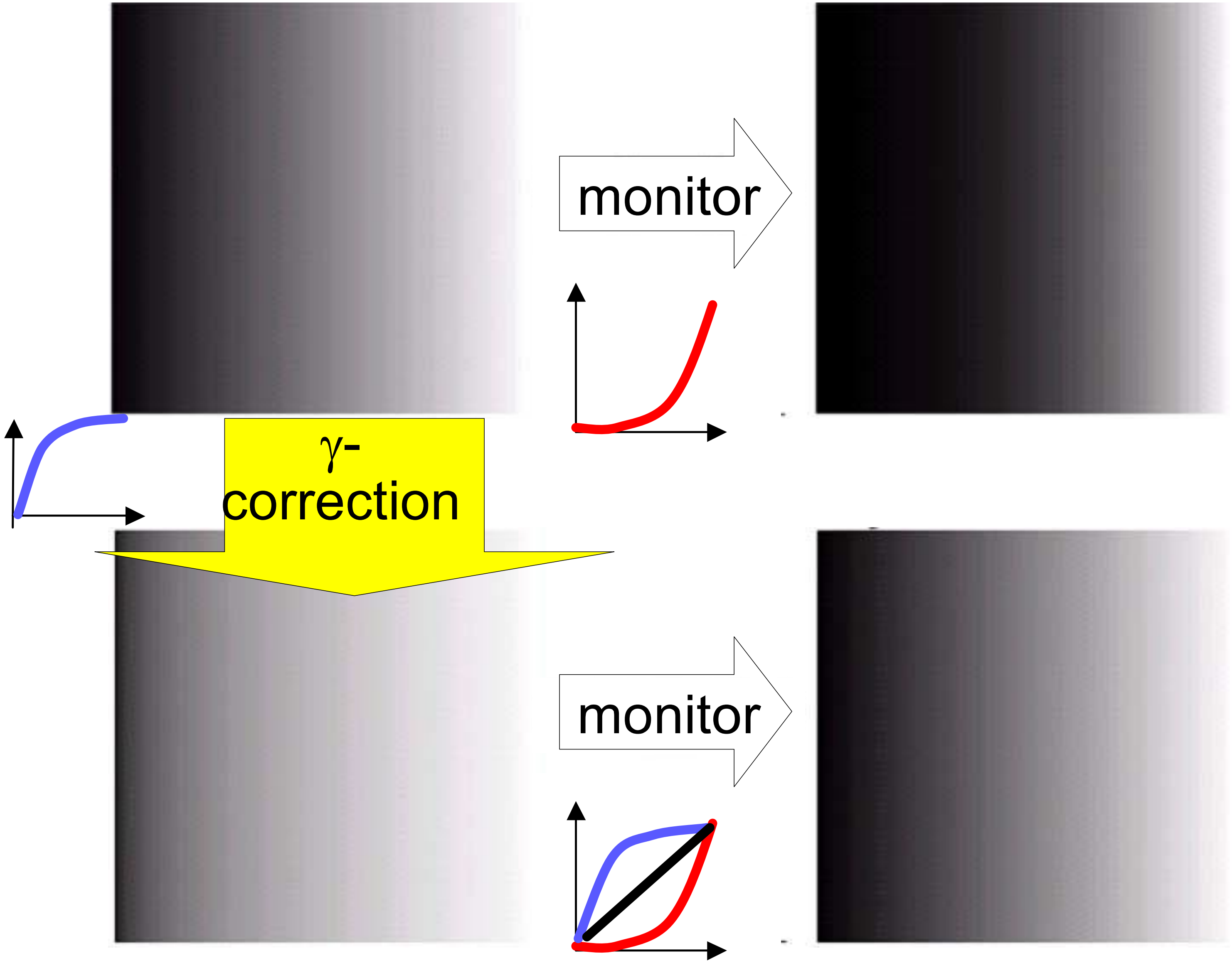
device dependent

device independent

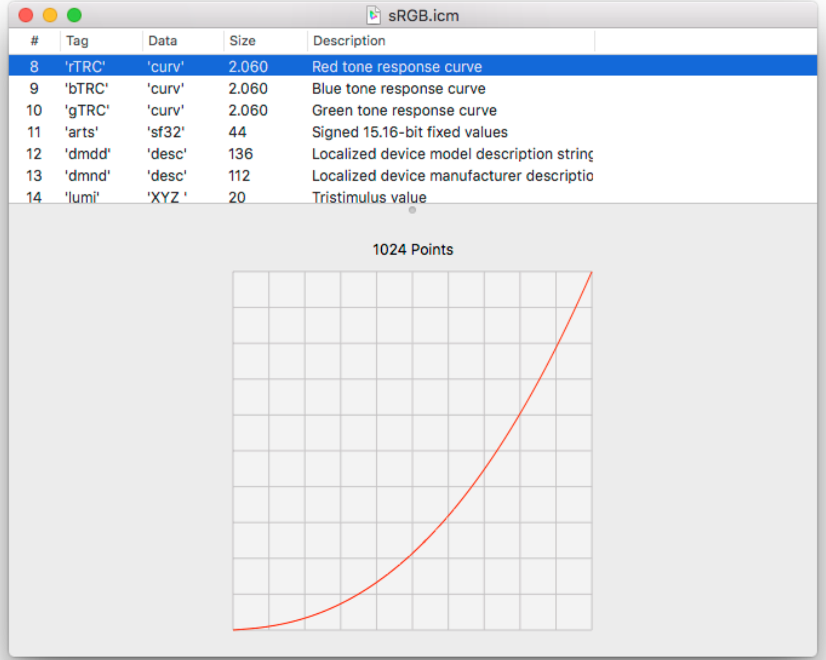
sRGB_linear

Image with a linear tone curve

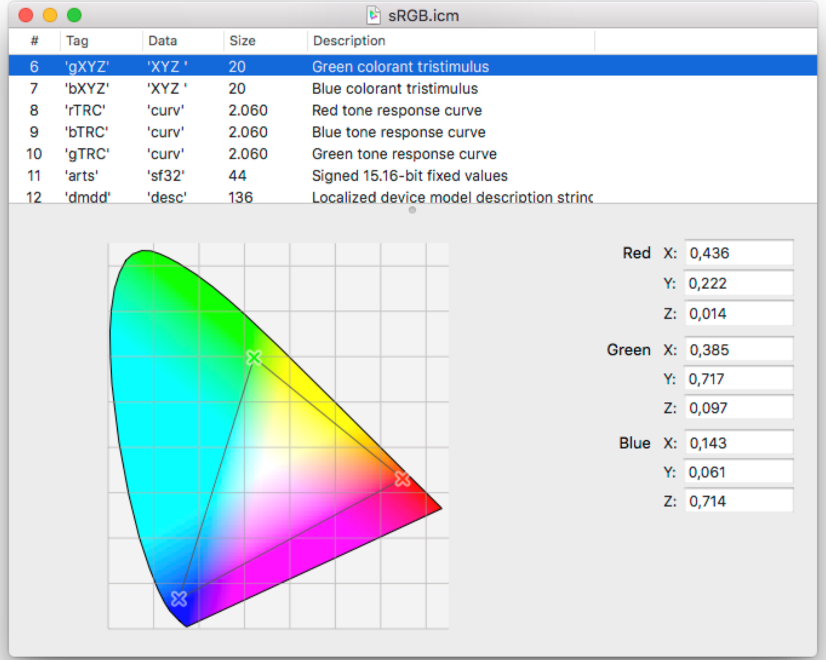
Tone Curve / Gamma



Tone curve



sRGB.icm
#	Tag	Data	Size	Description
8	'rTRC'	'curv'	2.060	Red tone response curve
9	'bTRC'	'curv'	2.060	Blue tone response curve
10	'gTRC'	'curv'	2.060	Green tone response curve
11	'arts'	'sf32'	44	Signed 15.16-bit fixed values
12	'dmdd'	'desc'	136	Localized device model description strinc
13	'dmnd'	'desc'	112	Localized device manufacturer descriptio
14	'lumi'	'XYZ '	20	Tristimulus value
1024 Points



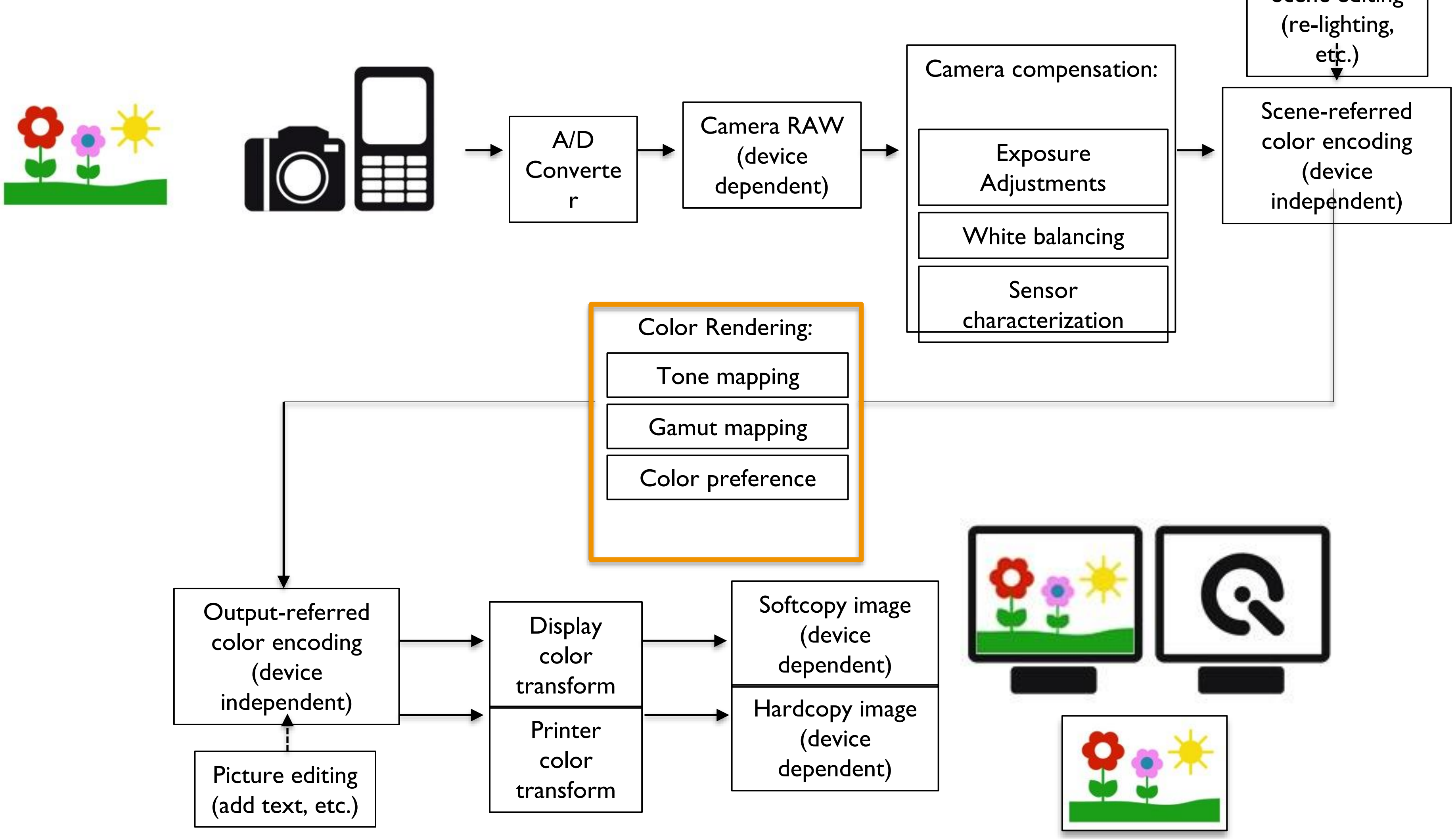
sRGB.icm
#	Tag	Data	Size	Description
6	'gXYZ'	'XYZ '	20	Green colorant tristimulus
7	'bXYZ'	'XYZ '	20	Blue colorant tristimulus
8	'rTRC'	'curv'	2.060	Red tone response curve
9	'bTRC'	'curv'	2.060	Blue tone response curve
10	'gTRC'	'curv'	2.060	Green tone response curve
11	'arts'	'sf32'	44	Signed 15.16-bit fixed values
12	'dmdd'	'desc'	136	Localized device model description strinc
Red X: 0,436
Y: 0,222
Z: 0,014
Green X: 0,385
Y: 0,717
Z: 0,097
Blue X: 0,143
Y: 0,061
Z: 0,714

sRGB_gamma

Final image – best for reproduction

Color rendering of a camera

Color Image Workflow

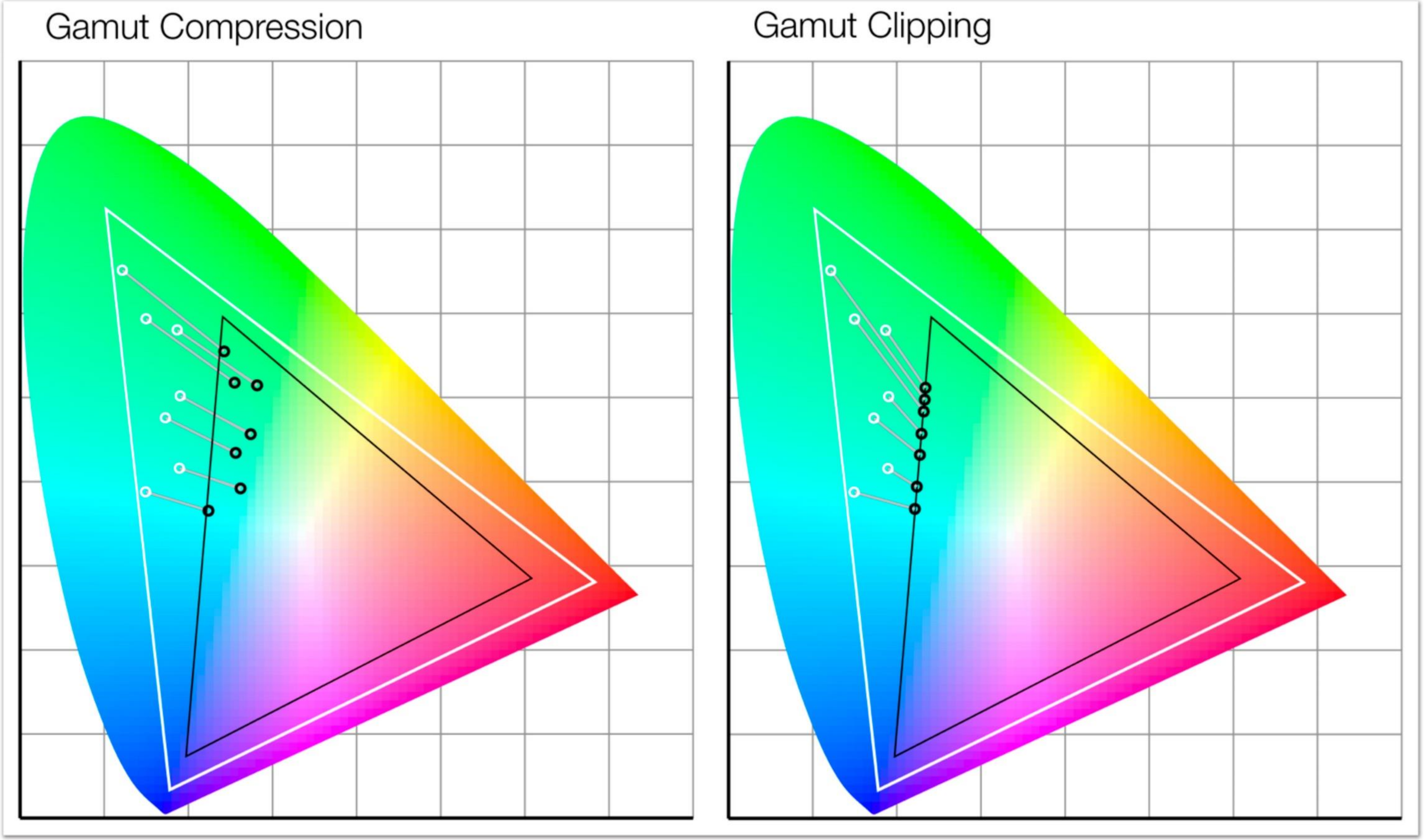


- Color rendering is the process where the analyzed scene colors are altered to produce pleasing reproductions
- The optimal color rendering will depend on:
  - Scene characteristics
  - Output medium characteristics
  - Customer preferences


Edited Raw-ACR 4.6


Unedited Jpeg

Gamut Mapping



sRGB_final

Further optimization in the ISP – mainly for human observer

Color Space

- RGB_camera is a device depended measurement of the incoming light, it is not a description of color

- To give RGB values a meaning, they need to be provided in a defined color space, e.g. sRGB

100

<

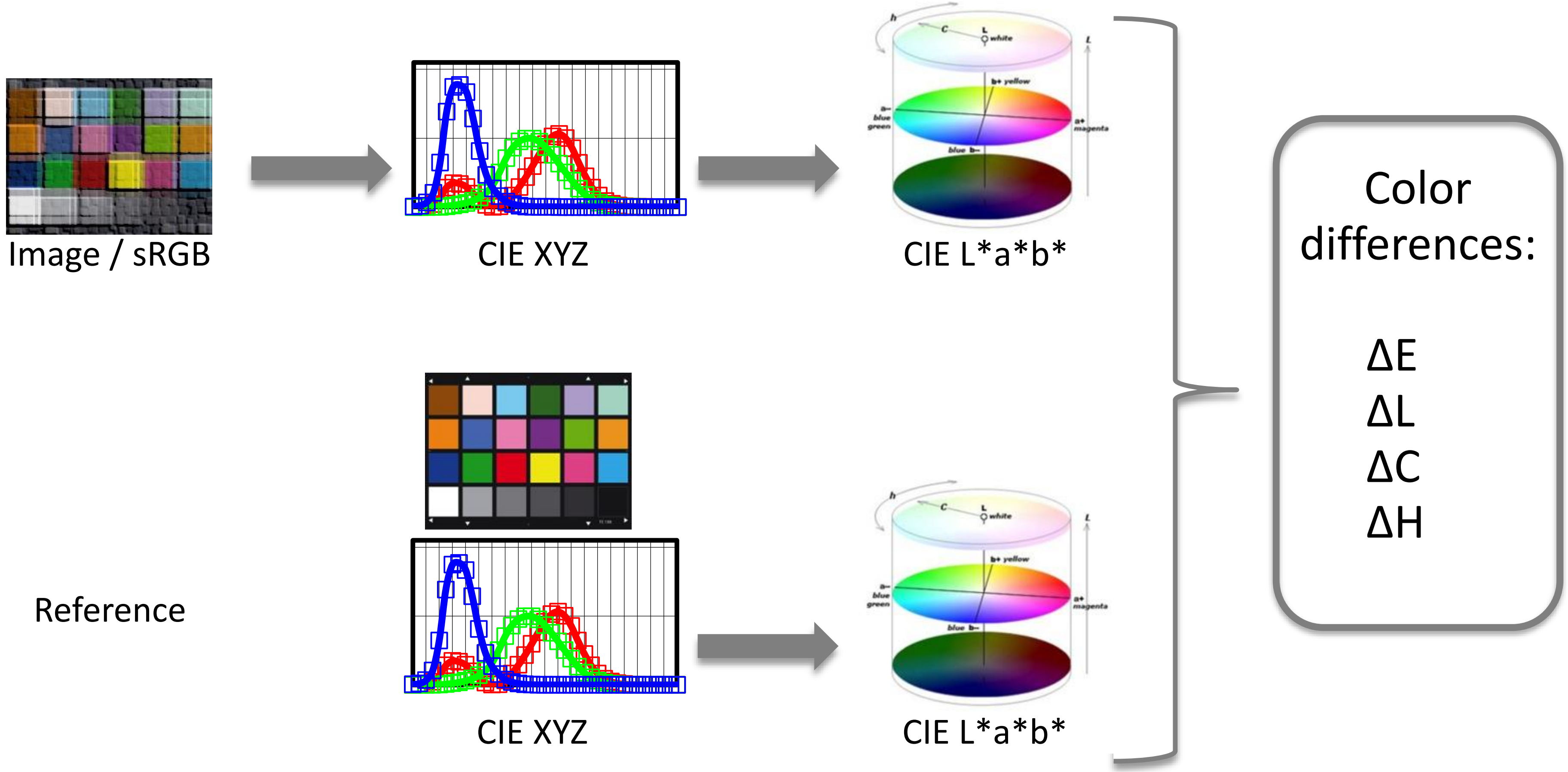
€, $, £

500

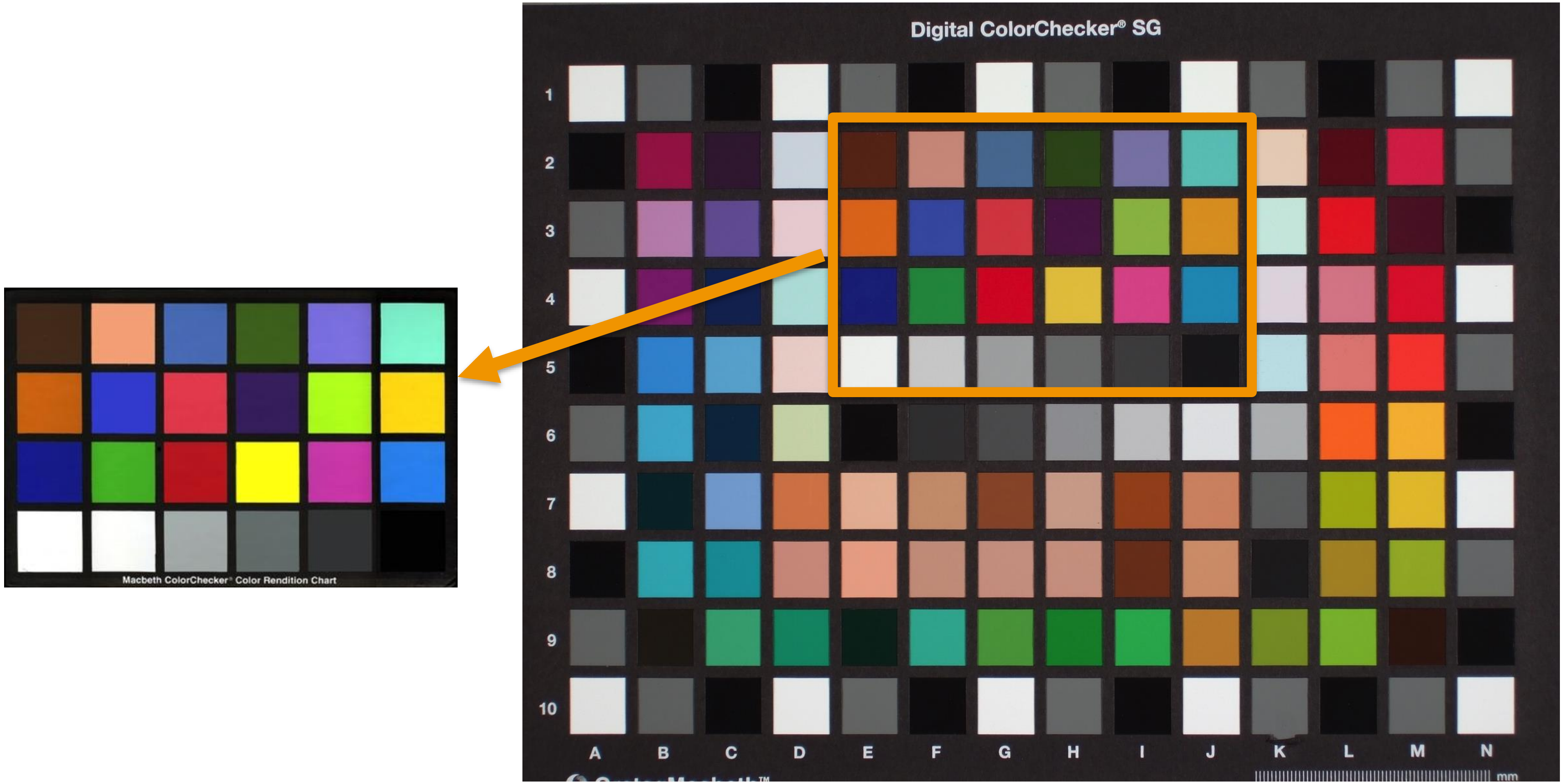
QUANTIFY COLOR REPRODUCTION

Quantify color reproduction

Concept of color reproduction quality evaluation



Colorchecker

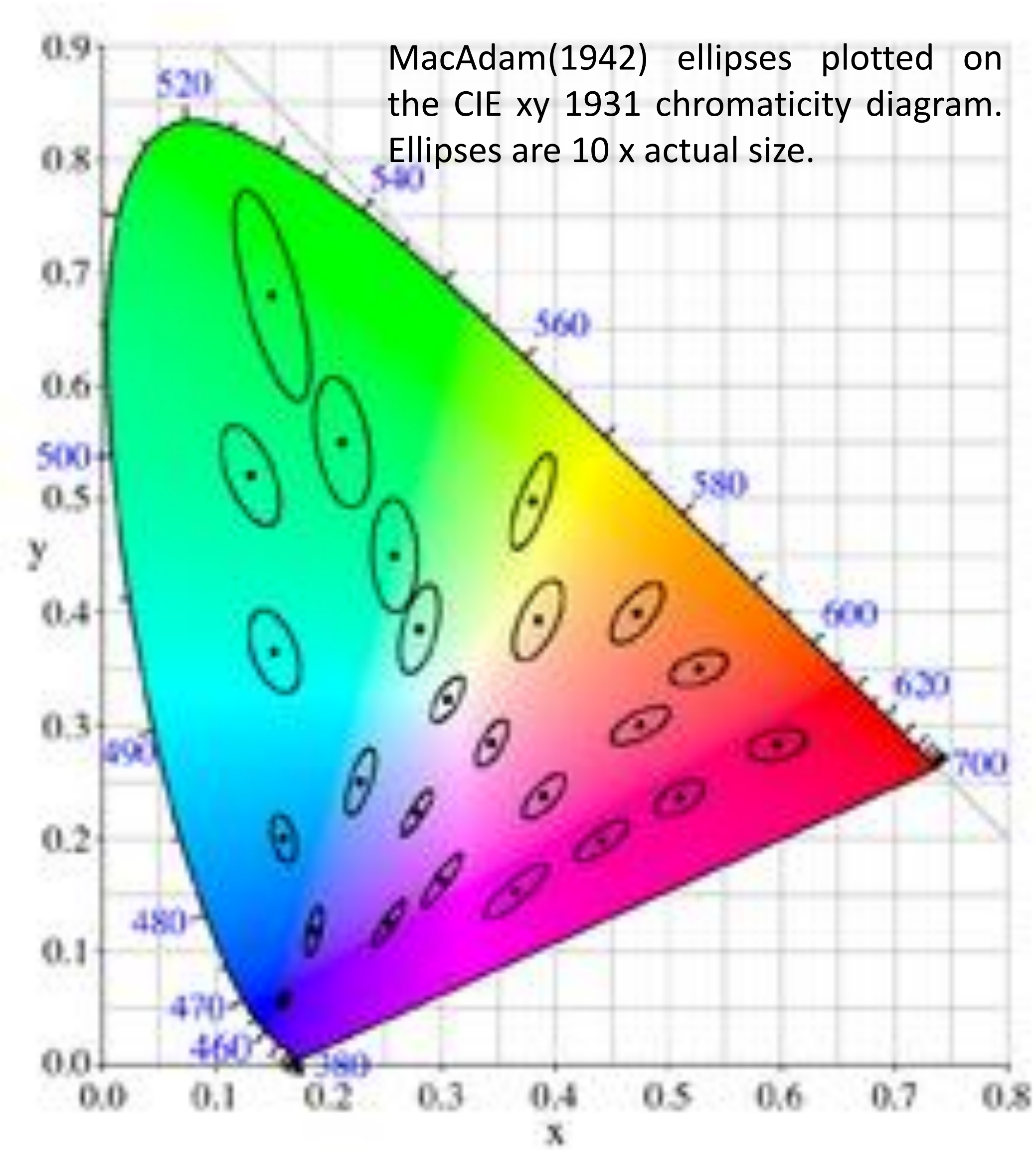


Quantify color reproduction

XYZ are linear but human vision is not

So called „MacAdamEllipse“ show that the perseption of color difference is not equal over the different colors.

So CIE-XYZ can not be used, the CIE-LAB is used for the description of color difference.



MacAdam(1942) ellipses plotted on the CIE xy 1931 chromaticity diagram. Ellipses are 10 x actual size.

https://commons.wikimedia.org/wiki/File:CIExy1931_MacAdam.png

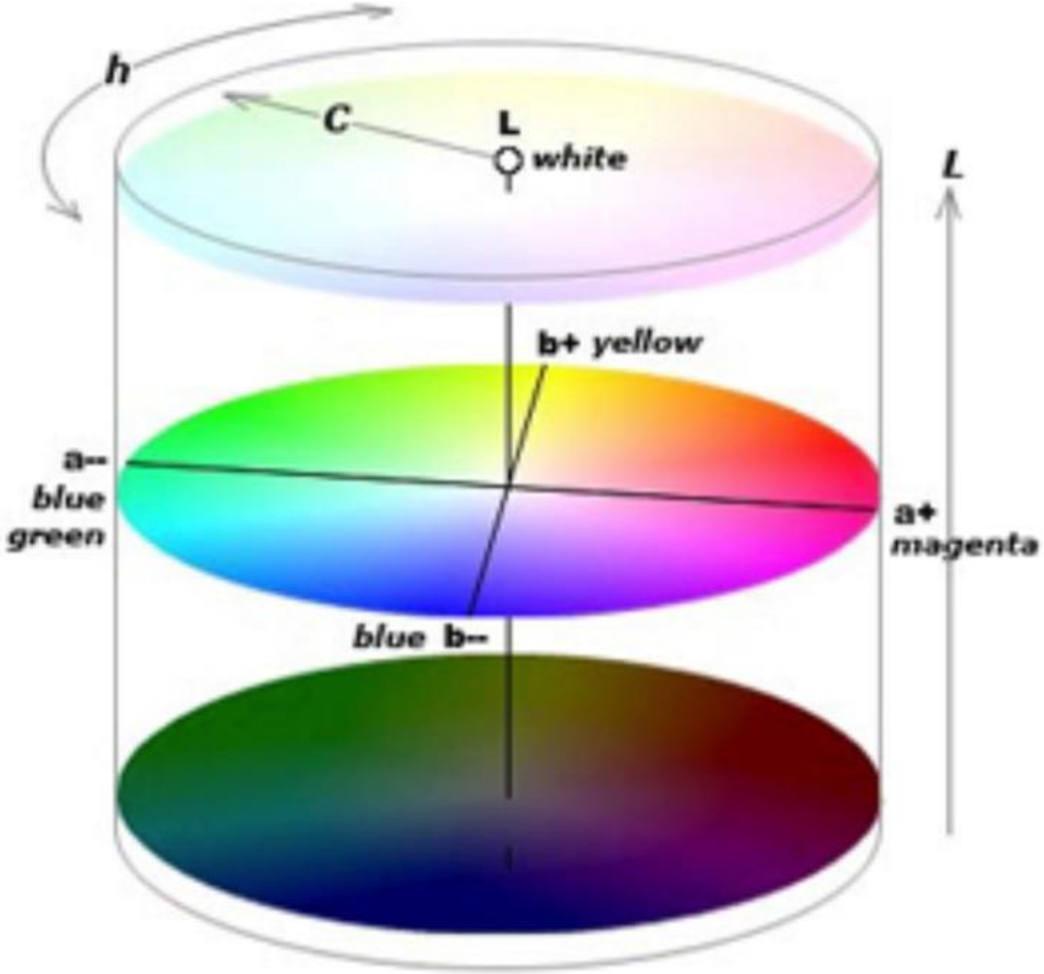
L*a*b* colorspace

➔ Transformation to L*a*b*

$$L^* = 116 \times \sqrt[3]{\frac{Y}{Y_n}} - 16$$

$$a^* = 500 \times \left[\sqrt[3]{\frac{X}{X_n}} - \sqrt[3]{\frac{Y}{Y_n}} \right]$$

$$b^* = 200 \times \left[\sqrt[3]{\frac{Y}{Y_n}} - \sqrt[3]{\frac{Z}{Z_n}} \right]$$



3D color space

Cartesian:

L* → Lightness

a* →cyan to magenta

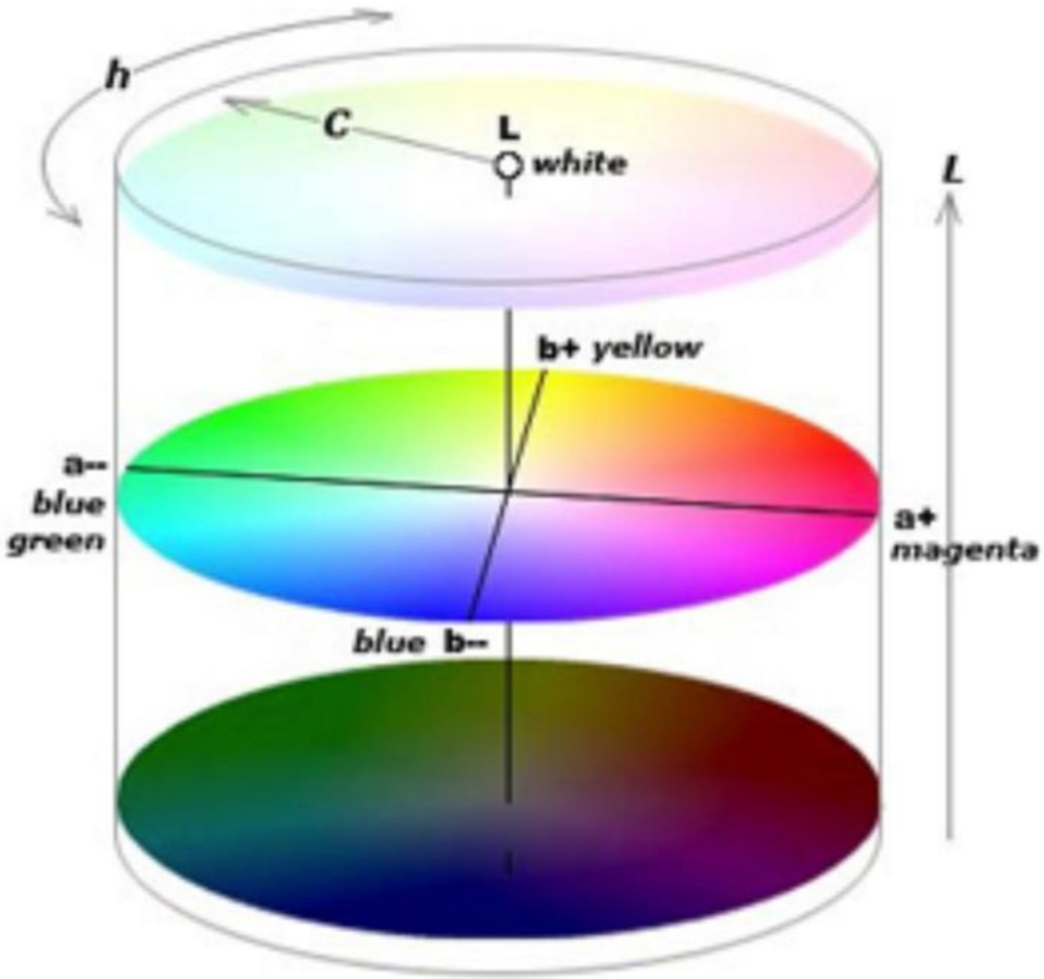
b* → blue to yellow

Polar:

L* → Lightness

C → Chroma / Saturation

H → Hue / Color Tone



Concept of color reproduction quality evaluation

$$\Delta E^*_i = \sqrt{(L^*_i - L^*_{ref})^2 + (a^*_i - a^*_{ref})^2 + (b^*_i - b^*_{ref})^2}$$

$$\Delta L^*_i = \sqrt{(L^*_i - L^*_{ref})^2}$$

$$DC_{abi} = \sqrt{a_i^{*2} + b_i^{*2}} - \sqrt{a_{ref}^{*2} + b_{ref}^{*2}}$$

$$\Delta H^*_{ab\,i} = \sqrt{(a^*_i - a^*_{ref})^2 + (b^*_i - b^*_{ref})^2 - \Delta C^{*\,2}_{ab_i}}$$

Color differences:

ΔE
ΔL
ΔC
ΔH

Color Difference CIE 1976

When the CIE-Lab ColorSpace was developed, the computer power was not high enough to get it perfect. Therefore the Color Difference Formula have been updated after that:

- CIE1976 – the original formula
- CIE1994 – updated version, lower error on higher saturated colors
- CIE2000 – further update

When you read or write specifications, you need to mention which formula to be used !

Quantify color reproduction

Workflow with iQ-Analyzer software

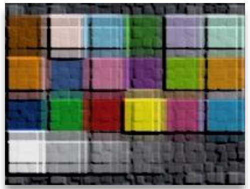


Image file

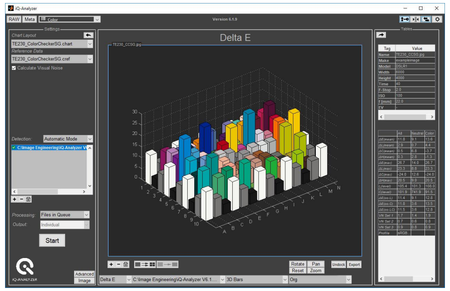
iQ-Analyzer

Reference

Define:

- White point
- Color difference formula
- Color space
- Reference values

Graphical results

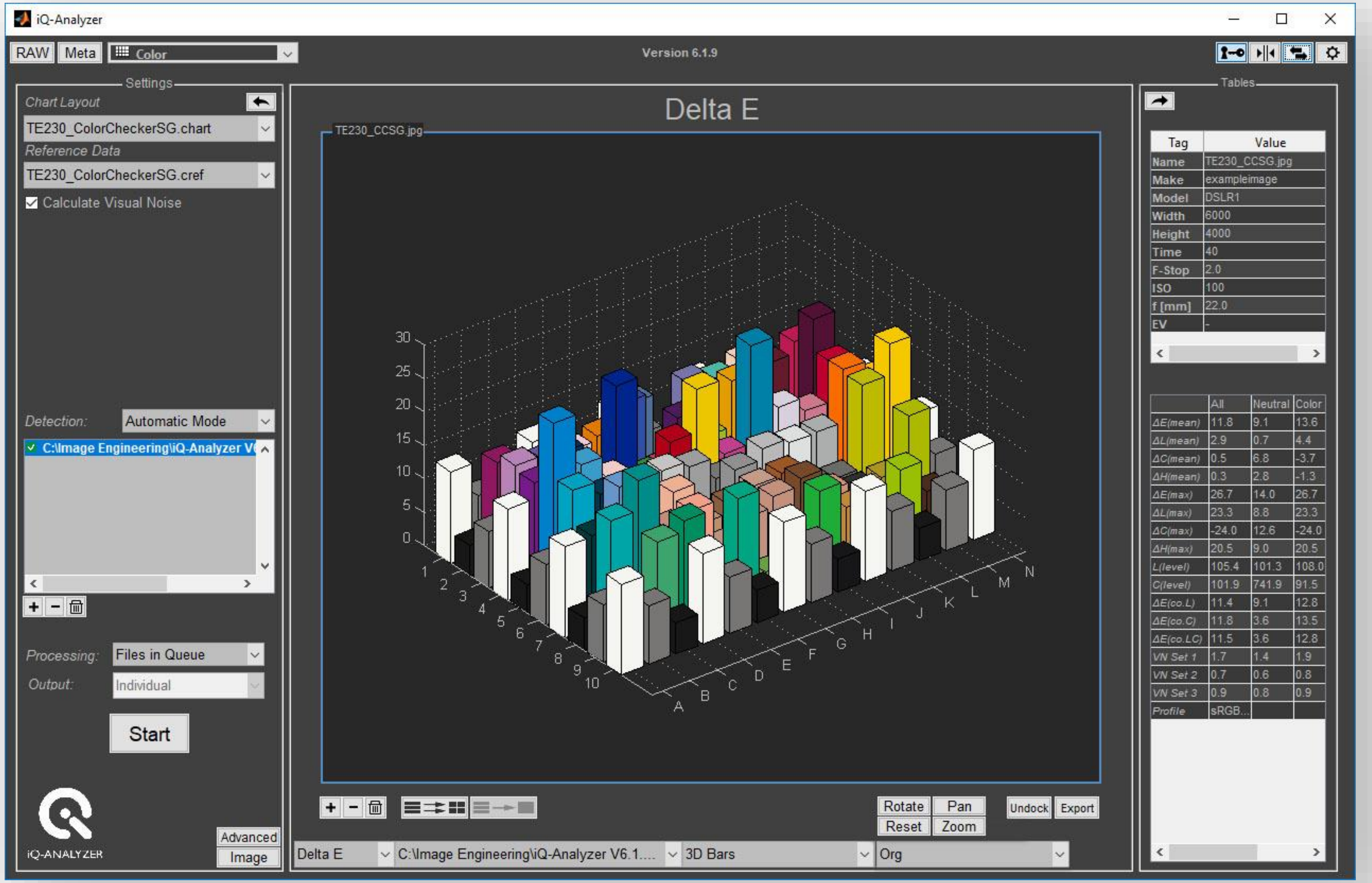


Numerical results

	All	Neutral	Color
ΔE(mean)	11.8	9.1	13.6
ΔL(mean)	2.9	0.7	4.4
ΔC(mean)	0.5	6.8	-3.7
ΔH(mean)	0.3	2.8	-1.3
ΔE(max)	26.7	14.0	26.7

Quantify color reproduction

Result for ColorChecker SG / 3D-bar projection



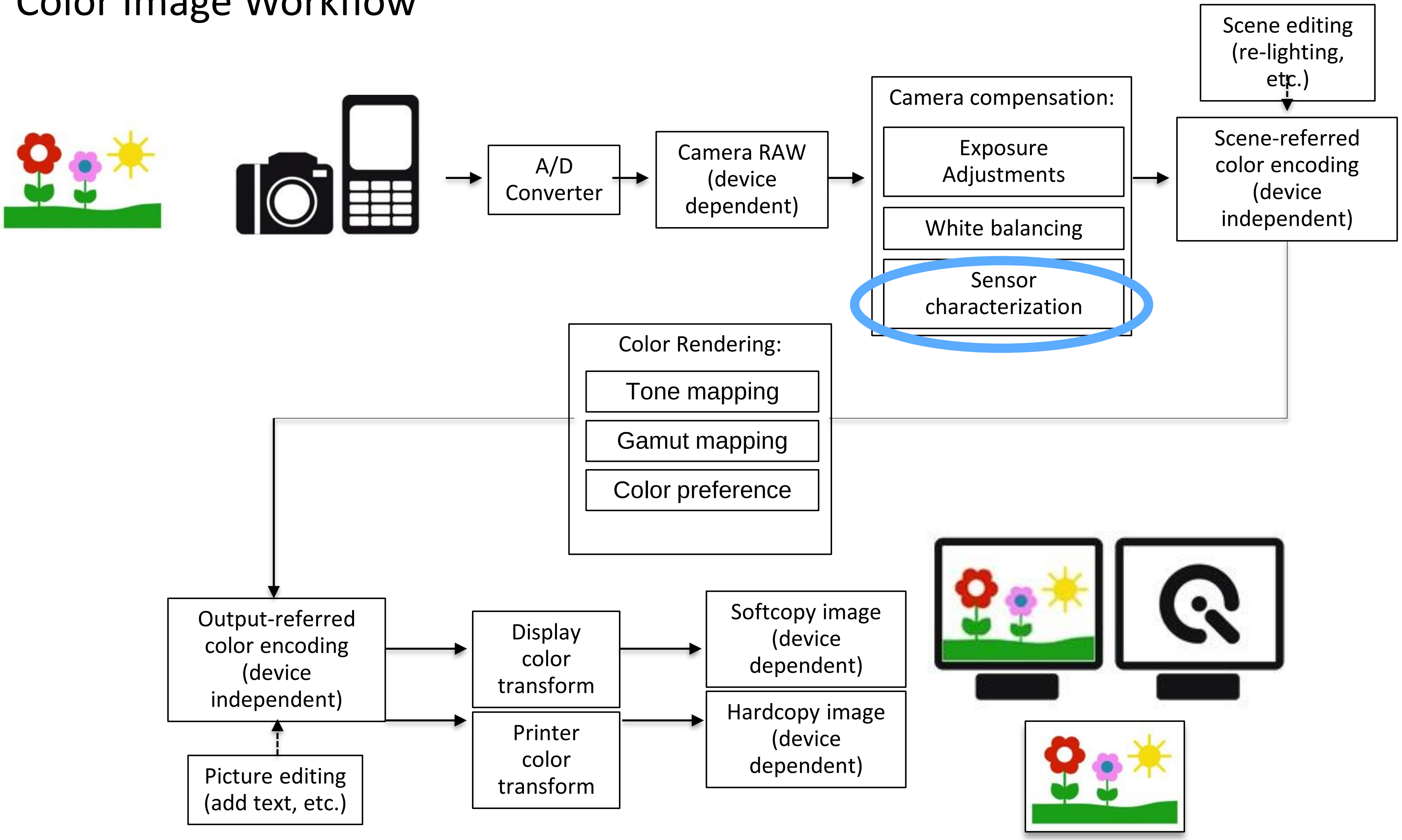
Color reproduction

- Most cameras are optimized for „nice colors“, not for perfect color reproduction.

- Color reproduction measurement indicates issues and should be part of all measurements

COLOR PROCESSING

Color Image Workflow

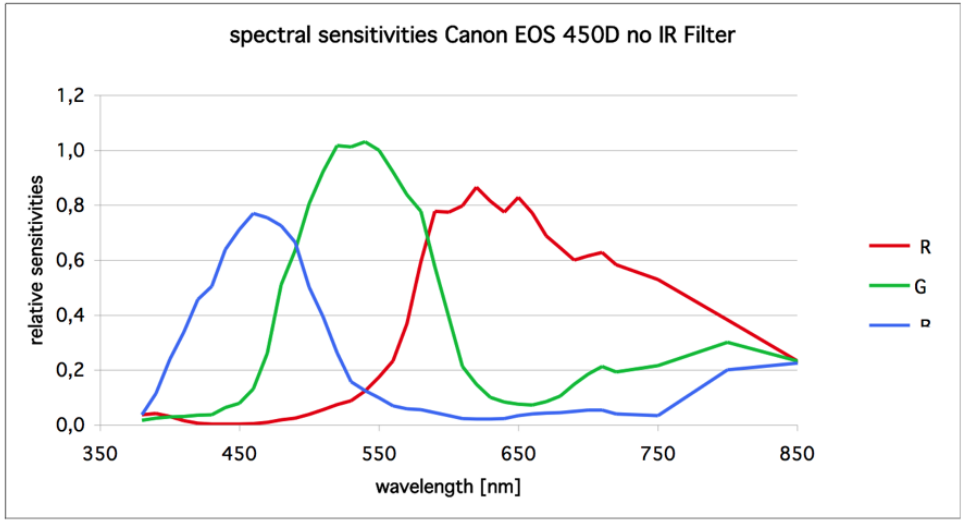


Color correction matrix

$$R_{Camera} = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot c_R(\lambda) \cdot d\lambda$$

$$G_{Camera} = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot c_G(\lambda) \cdot d\lambda$$

$$B_{Camera} = k \cdot \int_{380nm}^{780nm} s(\lambda) \cdot r(\lambda) \cdot c_B(\lambda) \cdot d\lambda$$



with:

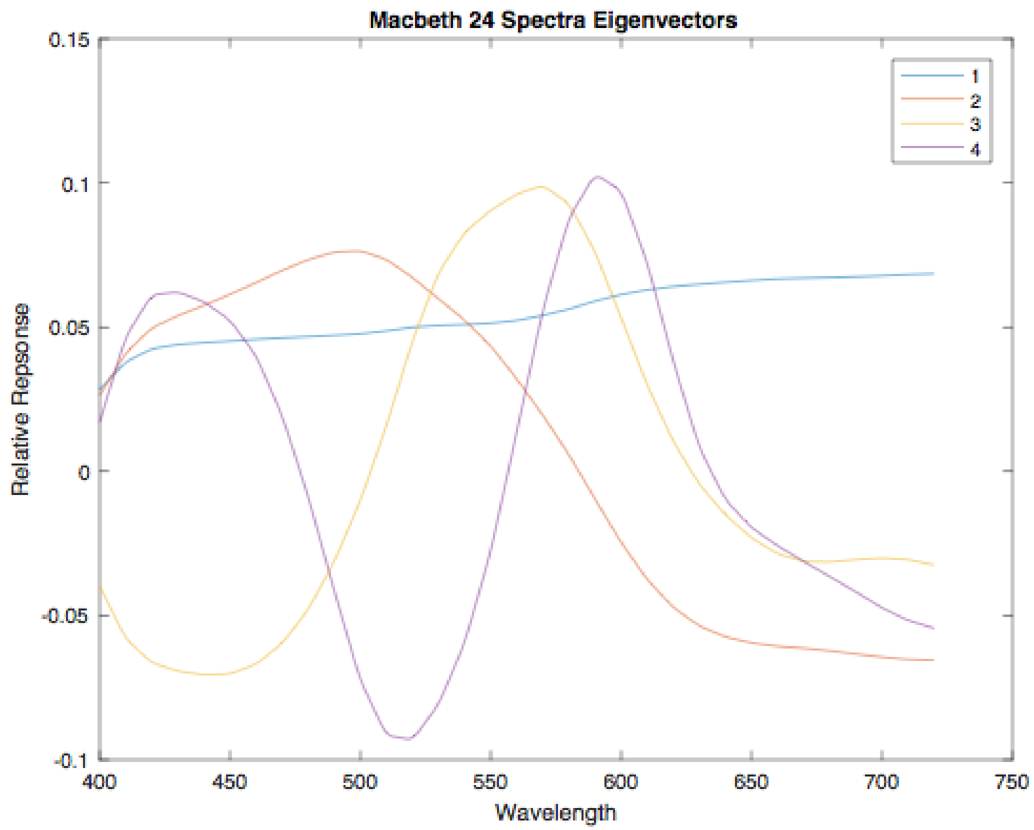
$s(\lambda)$ spectral distribution of the light source

$r(\lambda)$ spectral reflectance of the object

$c_X(\lambda)$ spectral sensitivity of the camera

Average and Maximum CIECAM16 ΔE errors for 28 cameras

Transform	Ave	Max
3X3	1,92	6,29
2D MLUT (D=28)	1,36	5,07
3D MLUT (D=9)	1,18	4,41

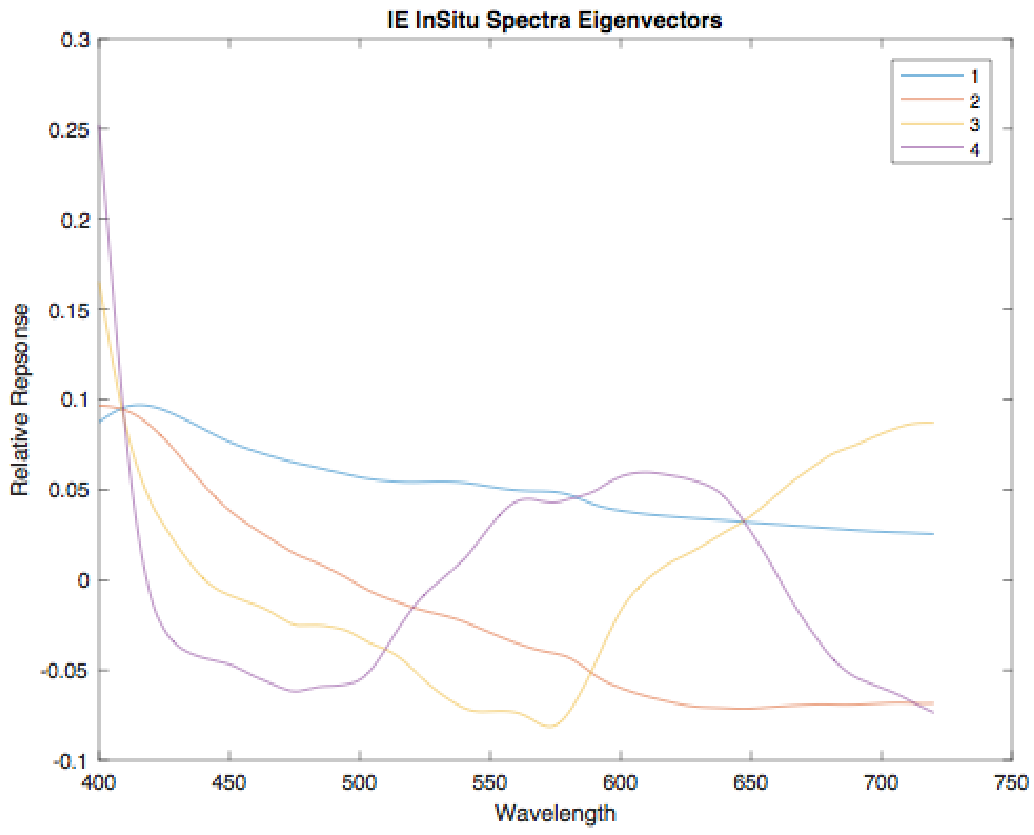


Test Data: X-Rite ColorChecker

Colormetric Performance

Average and Maximum CIECAM16 ΔE errors for 28 cameras

Transform	Ave	Max
3X3	0,84	11.0
2D MLUT (D=28)	0,83	10,3
3D MLUT (D=9)	0,81	8,07



Test Data: IE InSitu spectra

in-situ measured spectral radiances of natural objects.

- More than 2000 objects measured
- Many skintones off all different kinds
- Spectral range 380 to 780 nm
- In 5 nm steps
- Radiances with and without tile correction
- Image provided with each measurement

ı example for a typical measurement setup to measure skin tones.

The image shows a typical measurement setup to measure skin tones.

in-situ database

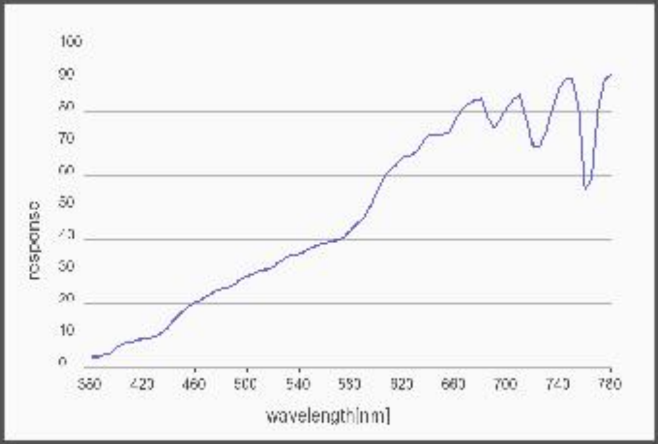
element details

add

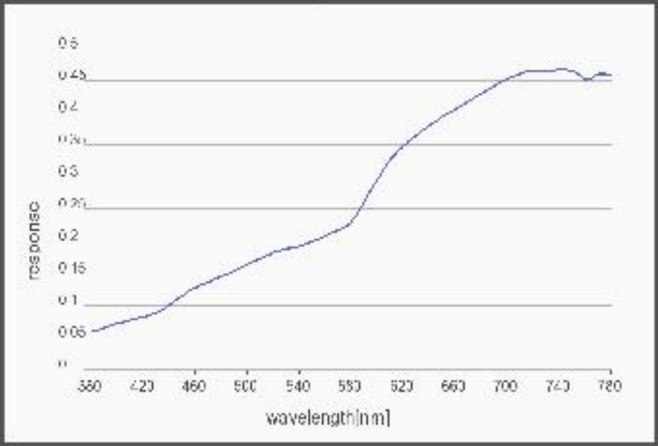
back

ID 2440	storage ID ea146
description skin | arm	category portrait
illumination daylight sunny	color skin asian
brightness level highlight	

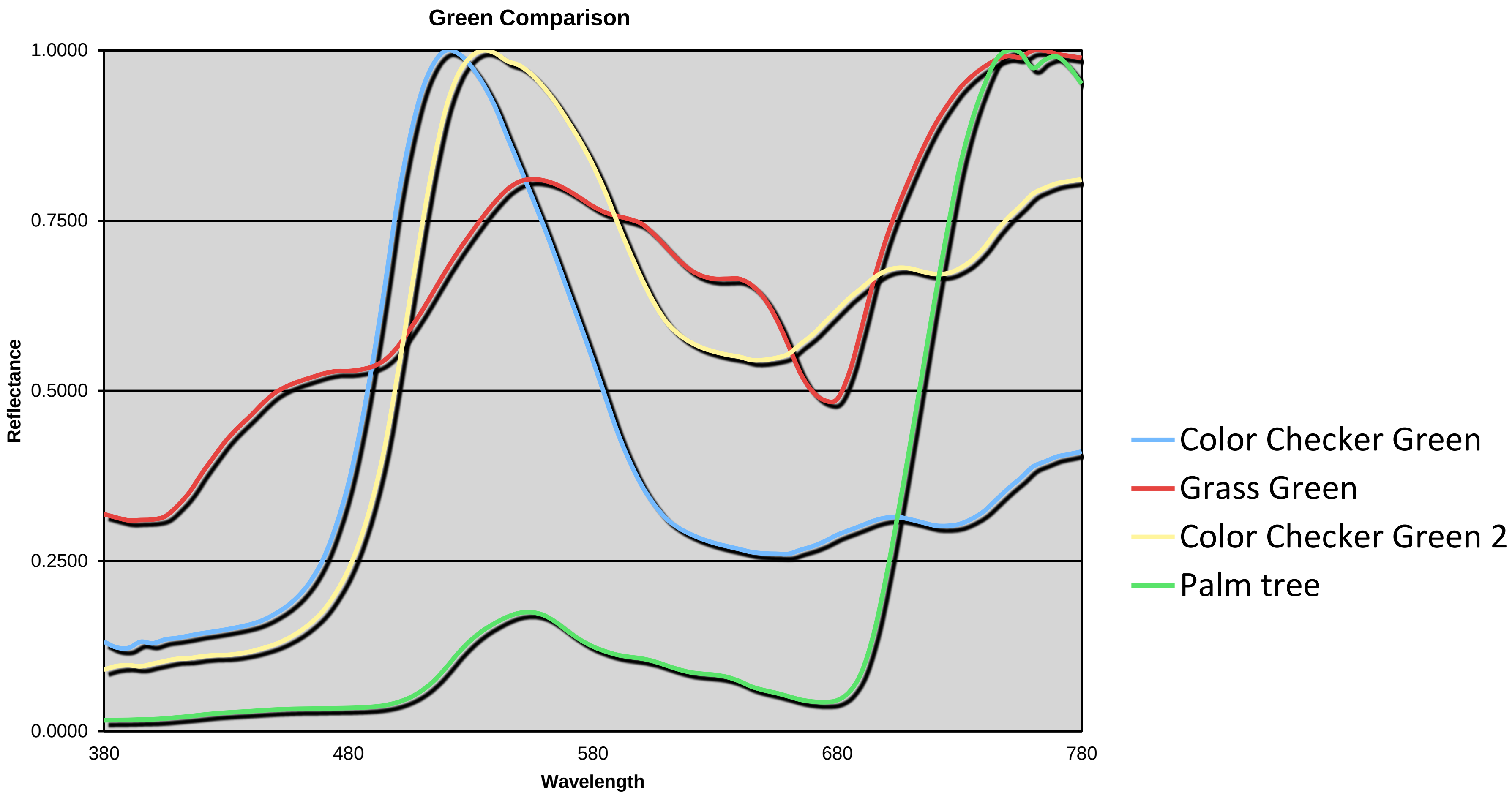
original data:



white tile corrected:



Real world colors vs. ColorChecker.



Tradeoffs

Linear (3X3):

✓ Simple, fast, and easy

✓ Exposure invariant

✓ Smooth

✓ Low memory and computational overhead

- Limited accuracy
- Stationary throughout color space
- Limited flexibility and customizability

MLUT:

✓ High accuracy

✓ Exposure invariant (2D and 2.5D MLUTs)

✓ Can optimize noise performance variably throughout color space

✓ Gamut-mappable

✓ Can encode nonlinear rendering transforms

✓ Highly flexibile and customizable

- Higher memory and computational overhead

Transform type determines accuracy, noise performance, and flexibility

- Linear transform (3x3) is simple, fast, easy, smooth, and exposure invariant but compromises accuracy, noise performance, and customizability

- MLUTs improve accuracy, noise performance, and customizability but have a greater footprint

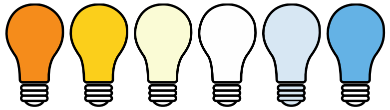
CHARACTERIZATION METHODS

In general, there are two different methods to characterize a camera/sensor:

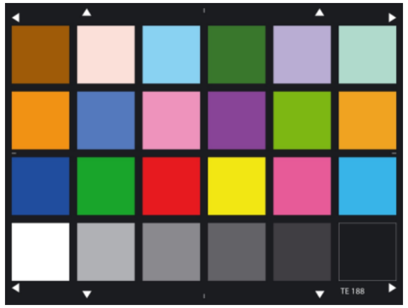
- Chart-based method
- Spectra-based method

In all cases, the workflow requires to obtain pairs of camera RGB data and device independent color information CIE-XYZ

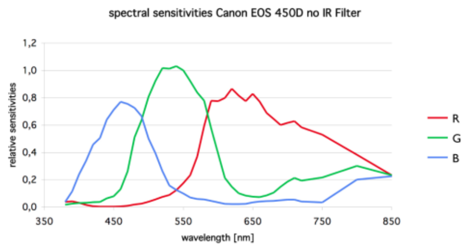
Chart-based method



X



X

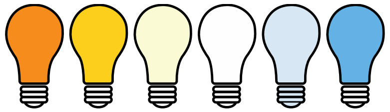


= RGB

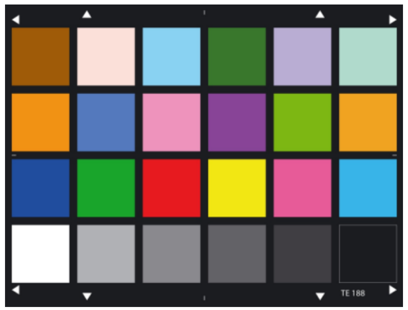
Lightsource
spectral distribution

Chart / Object
spectral reflection

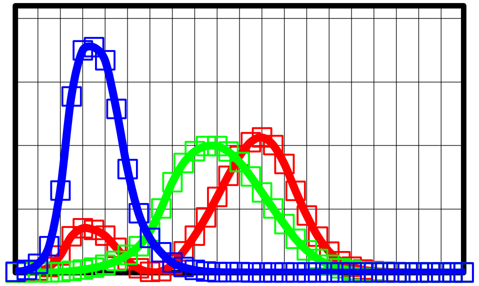
Camera / human
spectral sensitivity



X



X

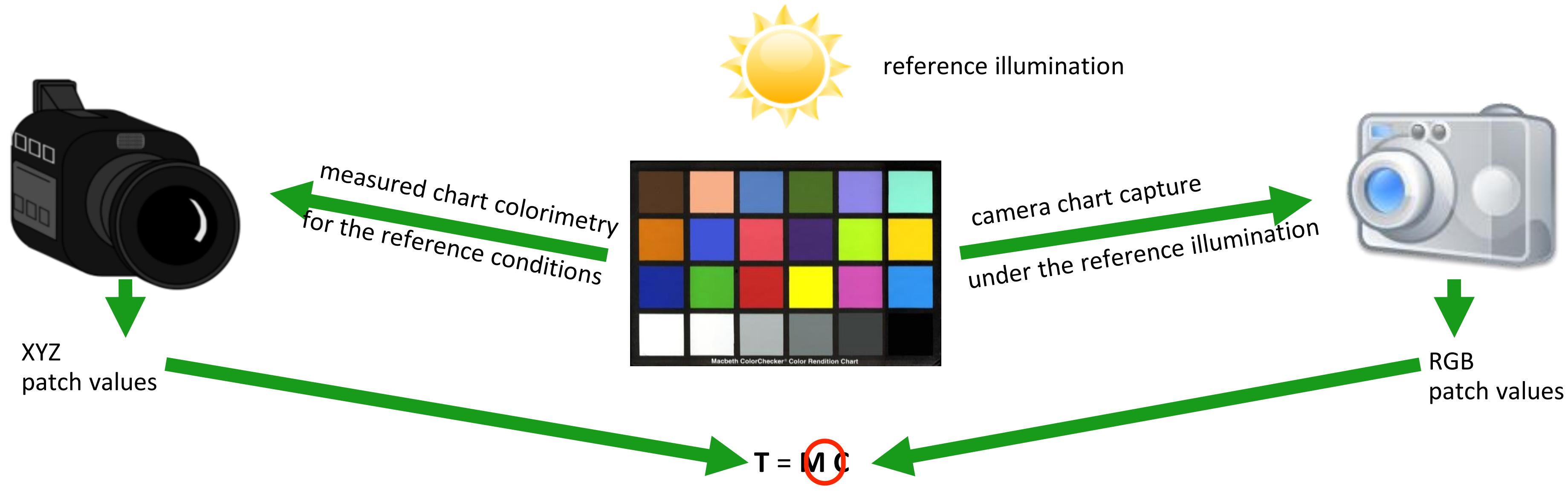


= XYZ

Chart-based method

$$\underbrace{\underset{C}{\begin{bmatrix} R_{camera} \\ G_{camera} \\ B_{camera} \end{bmatrix}} \underset{M}{\begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}}}_{\text{device dependent}} = \underbrace{\underset{T}{\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}}}_{\text{device independent}}$$

Chart-based method



Idealized, assuming raw, linear, and noiseless where **M** is the estimator to be determined that transforms between camera responses **C** and tristimulus values **T** that minimizes colorimetric residual errors:

$$\| \mathbf{MC} - \mathbf{T} \|^2 \Rightarrow 0$$

First by linear estimation:

$$\mathbf{M} = \mathbf{TC}^t(\mathbf{CC}^t)^{-1}$$

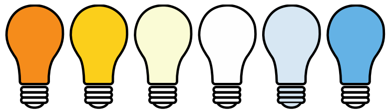
Generally followed by nonlinear optimization in perceptual coordinates (e.g. L*a*b*, CIECAM02):

$$\text{Iterate } \mathbf{M}\text{: } \| \Delta E_{f(\mathbf{T})}^{f(\mathbf{MC})} \|^2 \Rightarrow 0$$

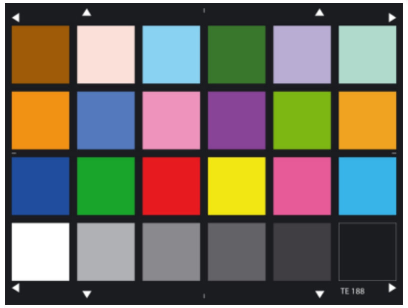
Degrees of freedom can be reduced by constraining ***M*** *to the scene adopted white point*

Chart-based method

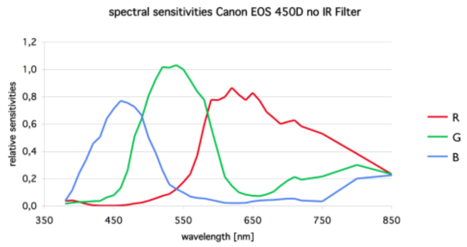
RGB values created by the camera.



X



X



= RGB

Lightsource
spectral distribution

Chart / Object
spectral reflection

Camera
spectral sensitivity

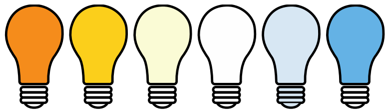
Taken as given

unknown

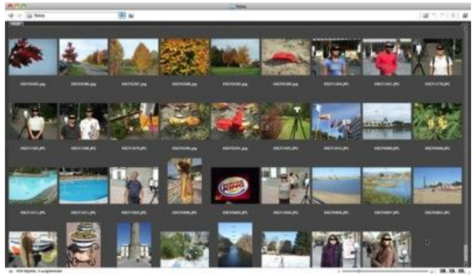
measured

Spectra-based method

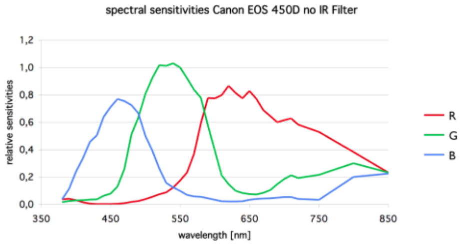
RGB values created by the camera.



X



X



= RGB

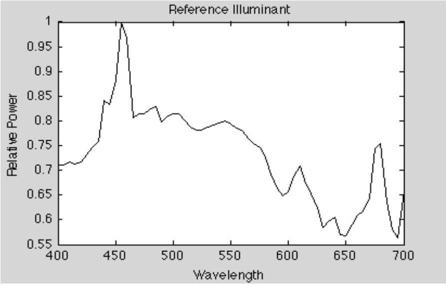
Lightsource
spectral distribution

Chart / Object
spectral reflection

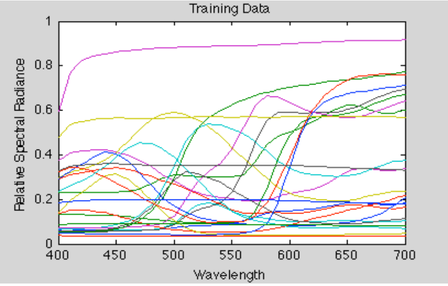
Camera
spectral sensitivity

$\underbrace{\hspace{6em}}_{\text{Intended simulation data}}$ $\underbrace{\hspace{3em}}_{\text{measured}}$ $\underbrace{\hspace{3em}}_{\text{calculated}}$

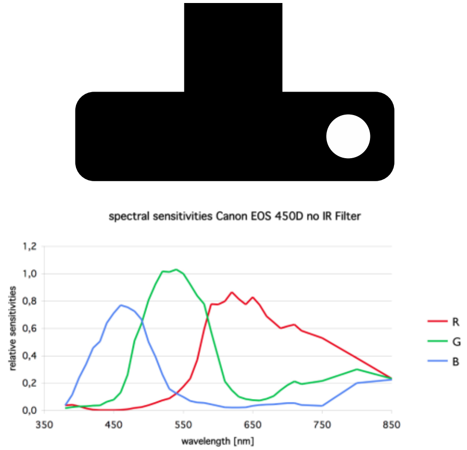
Spectra-based method



X



X

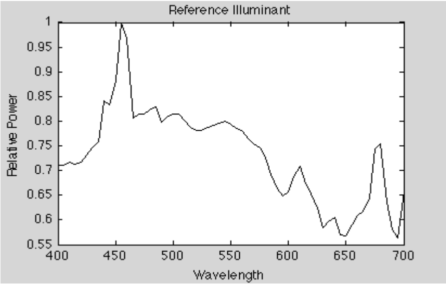


= RGB

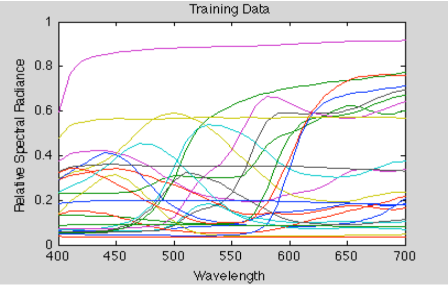
Lightsource
spectral distribution

Chart / Object
spectral reflection

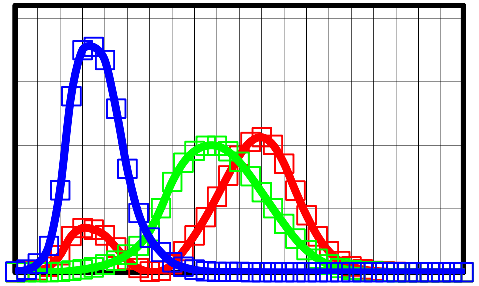
Camera / human
spectral sensitivity



X



X

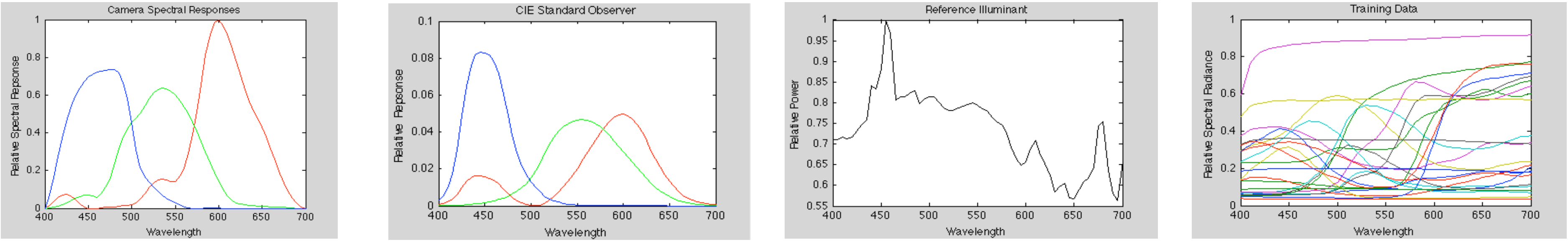


= XYZ

Spectra-based method

T = **O I S**

Training spectra colorimetric values



C = **R I S**

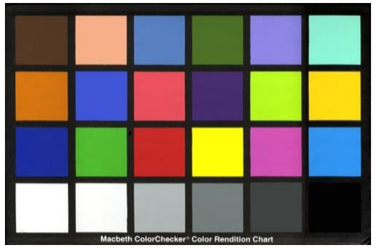
Training spectra camera values

then: **T** = **M C** as before

solve for **M** as illustrated previously

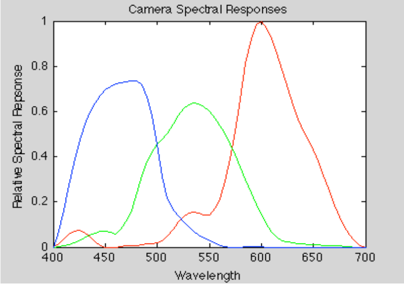
Tradeoffs

Chart:



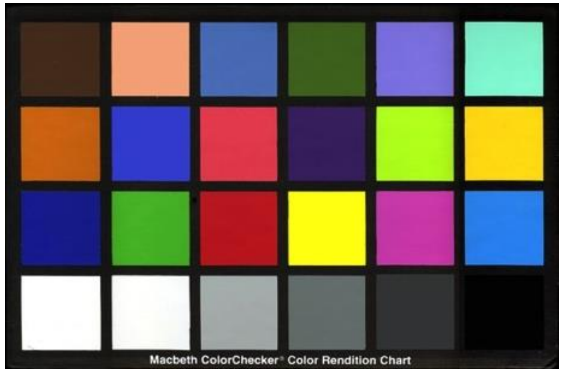
- ✓ Simple, fast, and easy
- ✓ Minimal equipment required (target and colorimetric data)
- \- Valid transform domain is limited to the target gamut and capture conditions
- \- Multiple sets of target captures and pre-computed transforms required for each Original Scene adopted white point
- \- Chart colorants are not generally representative of likely Original Scene objects resulting in metameric errors
- \- Limited to low-dimensionality transforms (matrix)

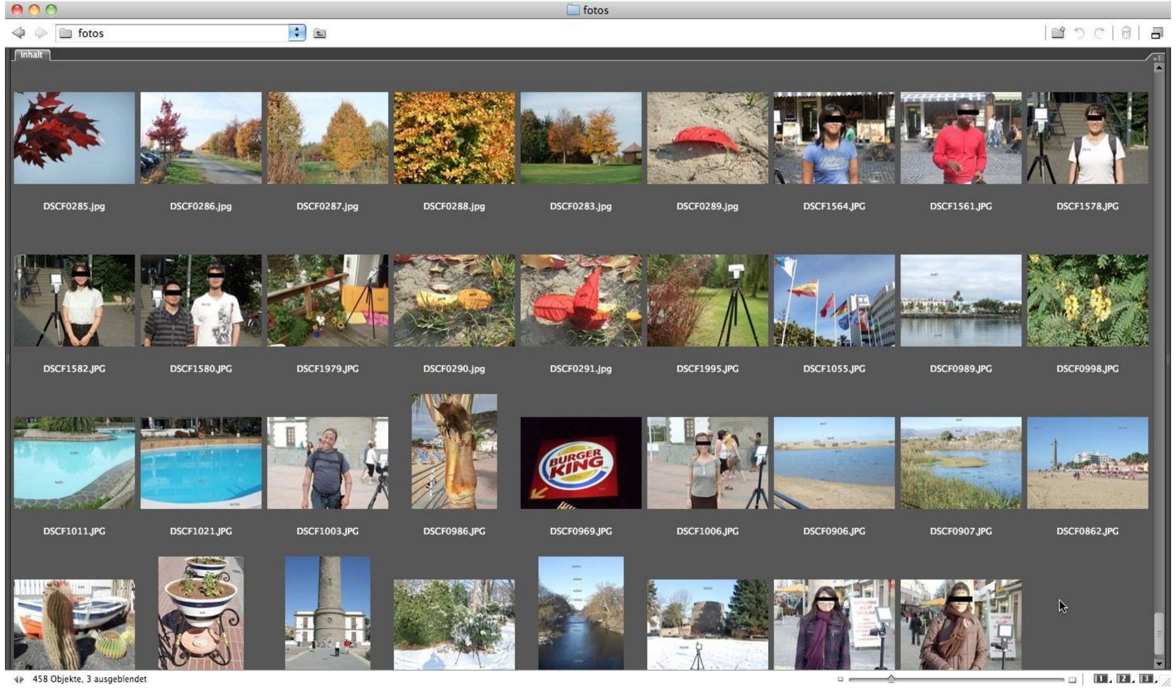
Spectra:



- ✓ Transform can be computed for any Original Scene adopted white point
- ✓ Training data is selected to be representative of Original Scene objects and radiation modes thereby minimizing metameric errors and optimizing for the likely use-cases
- ✓ Transforms are robust over a wide range of capture conditions and radiation modes
- ✓ Transforms are easily updated
- ✓ Suited to higher-dimensionality transforms (MLUT)
- \- Requires to measure camera spectral sensitivities

Chart vs. „real live“™



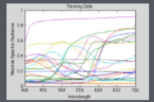
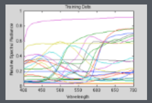


Training

For the same camera, two different color transforms CCMs were created: $CCM_{spectra}$ and CCM_{chart}

- $CCM_{spectra}$ was created using the spectra approach, the used training data was a large set of colors from the InSitu database
- CCM_{chart} was created using a color checker test target, so the training data is only the patches from that chart

So “Spectra” uses the spectral data for training.
“Chart” uses the Chart for Training.

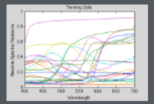
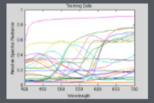
	Training	Test
Spectra	.7 / 5.7 	.8 / 2.2 
Chart	.6 / 1.9 	1.1 / 13.4 

Typical Average / Maximum ΔE colorimetric errors

Training

With each CCM, the color error has been tested.

So “Spectra” was tested using the ColorChecker
“Chart” was tested using the InSitu Data.

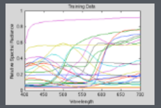
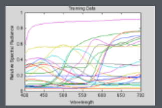
	Training	Test
Spectra	.7 / 5.7 	.8 / 2.2 
Chart	.6 / 1.9 	1.1 / 13.4 

Typical Average / Maximum ΔE colorimetric errors

Example: .7 / 5.7

Average color error for all tested colors in ΔE = 0.7

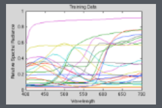
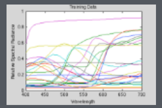
Maximum color error for all tested colors in ΔE = 5.7

	Training	Test
Spectra	.7 / 5.7 	.8 / 2.2 
Chart	.6 / 1.9 	1.1 / 13.4 

Typical Average / Maximum ΔE colorimetric errors

CCM_spectra

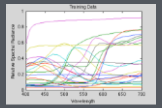
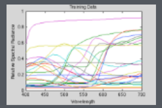
- The $CCM_{spectra}$ was created using the InSitu data.
- When calculating the error for this data, the average ΔE is 0.7, max, 5.7
- When calculating the error for ColorChecker data, the average ΔE is 0.8 and max ΔE is 2.2

	Training	Test
Spectra	.7 / 5.7 	.8 / 2.2 
Chart	.6 / 1.9 	1.1 / 13.4 

Typical Average / Maximum ΔE colorimetric errors

CCM_spectra

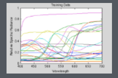
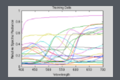
- The CCM_{chart} was created using the Colorchecker.
- When calculating the error for the color checker, the average ΔE is 0.6, max, 1.9
- When calculating the error for InSitu data, the average ΔE is 1.1 and max ΔE is 13.4

	Training	Test
Spectra	.7 / 5.7 	.8 / 2.2 
Chart	.6 / 1.9 	1.1 / 13.4 

Typical Average / Maximum ΔE colorimetric errors

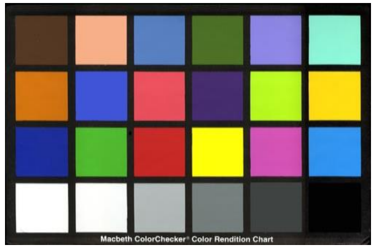
- The $CCM_{spectra}$ performes only slightly worse on the ColorChecker than the dedicated CCM_{chart} but very good on the real (in situ) colors.
- The CCM_{chart} performs well for a ColorChecker, but significantly worse on the real (in situ) colors.

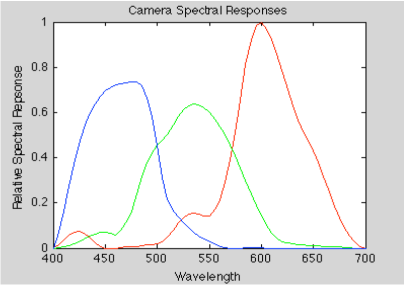
→ For real world colors the $CCM_{spectra}$ is the much better choice!

	Training	Test
Spectra	.7 / 5.7 	.8 / 2.2 
Chart	.6 / 1.9 	1.1 / 13.4 

Typical Average / Maximum ΔE colorimetric errors

Tradeoffs

Chart: 

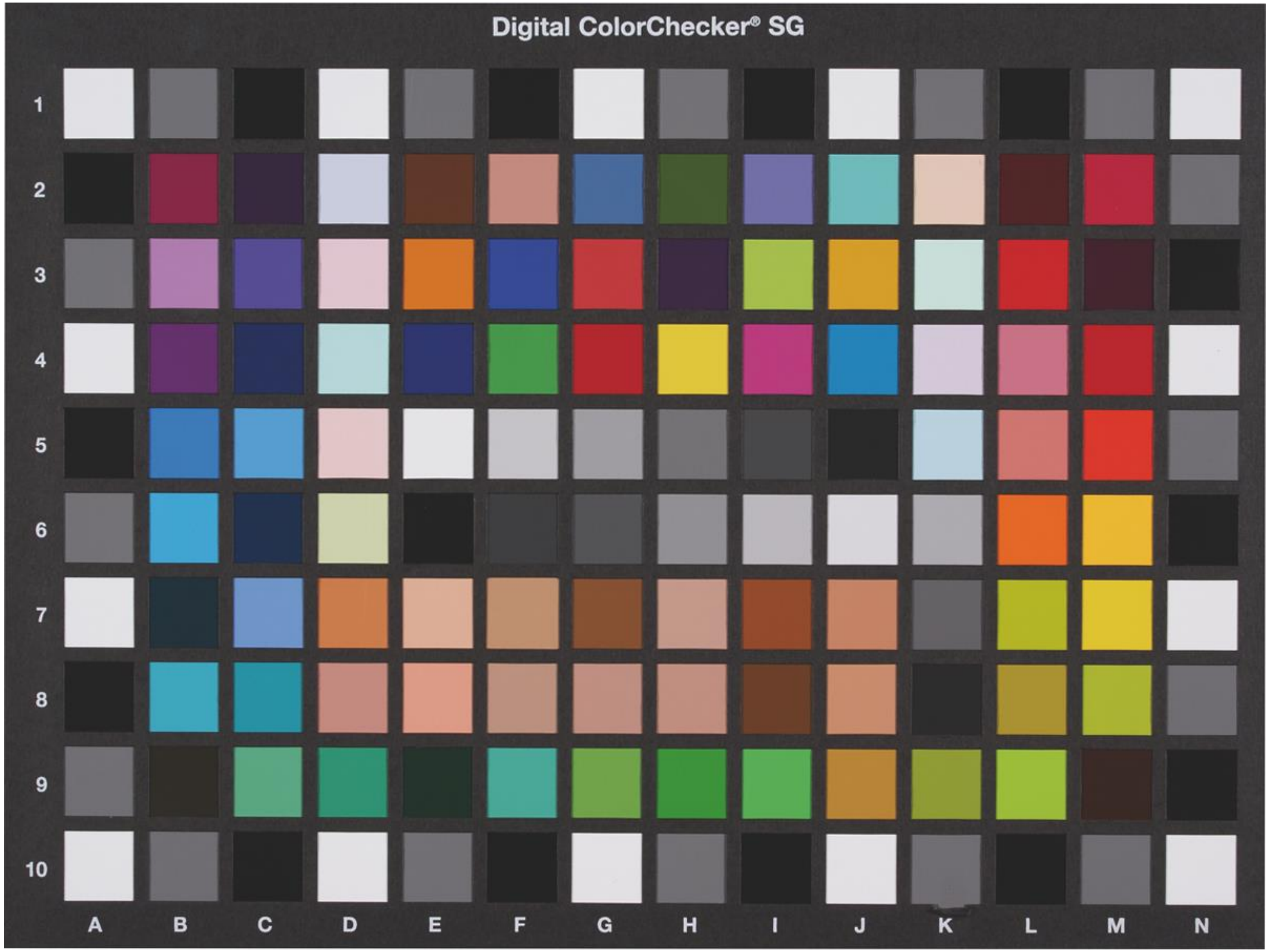
Spectra: 

- ✓ Simple, fast, and easy
- ✓ Minimal equipment required (target and colorimetric data)
- \- Valid transform domain is limited to the target gamut and capture conditions
- \- Multiple sets of target captures and pre-computed transforms required for each Original Scene adopted white point
- \- Chart colorants are not generally representative of likely Original Scene objects resulting in metameric errors
- \- Limited to low-dimensionality transforms (matrix)

- ✓ Transform can be computed for any Original Scene adopted white point
- ✓ Training data is selected to be representative of Original Scene objects and radiation modes thereby minimizing metameric errors and optimizing for the likely use-cases
- ✓ Transforms are robust over a wide range of capture conditions and radiation modes
- ✓ Transforms are easily updated
- ✓ Suited to higher-dimensionality transforms (MLUT)
- \- Requires to measure camera spectral sensitivities

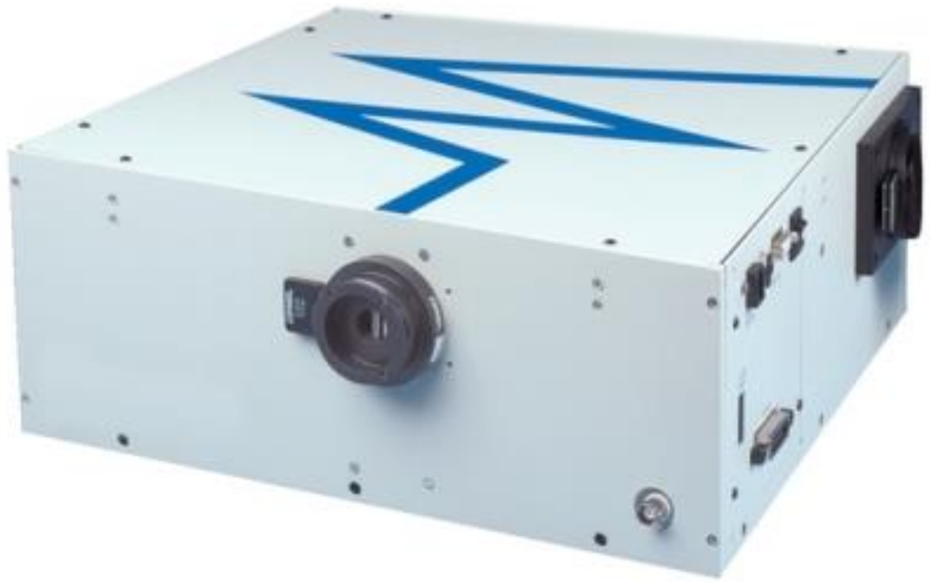
CALIBRATION TOOLS

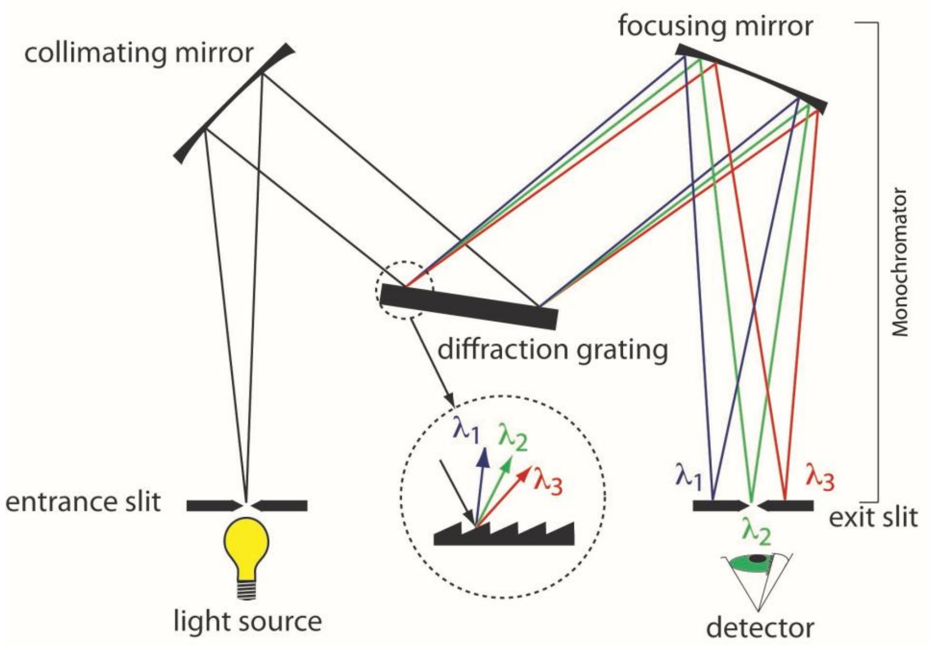
X-Rite ColorChecker SG



Color Chart used in Photography

Monochromator

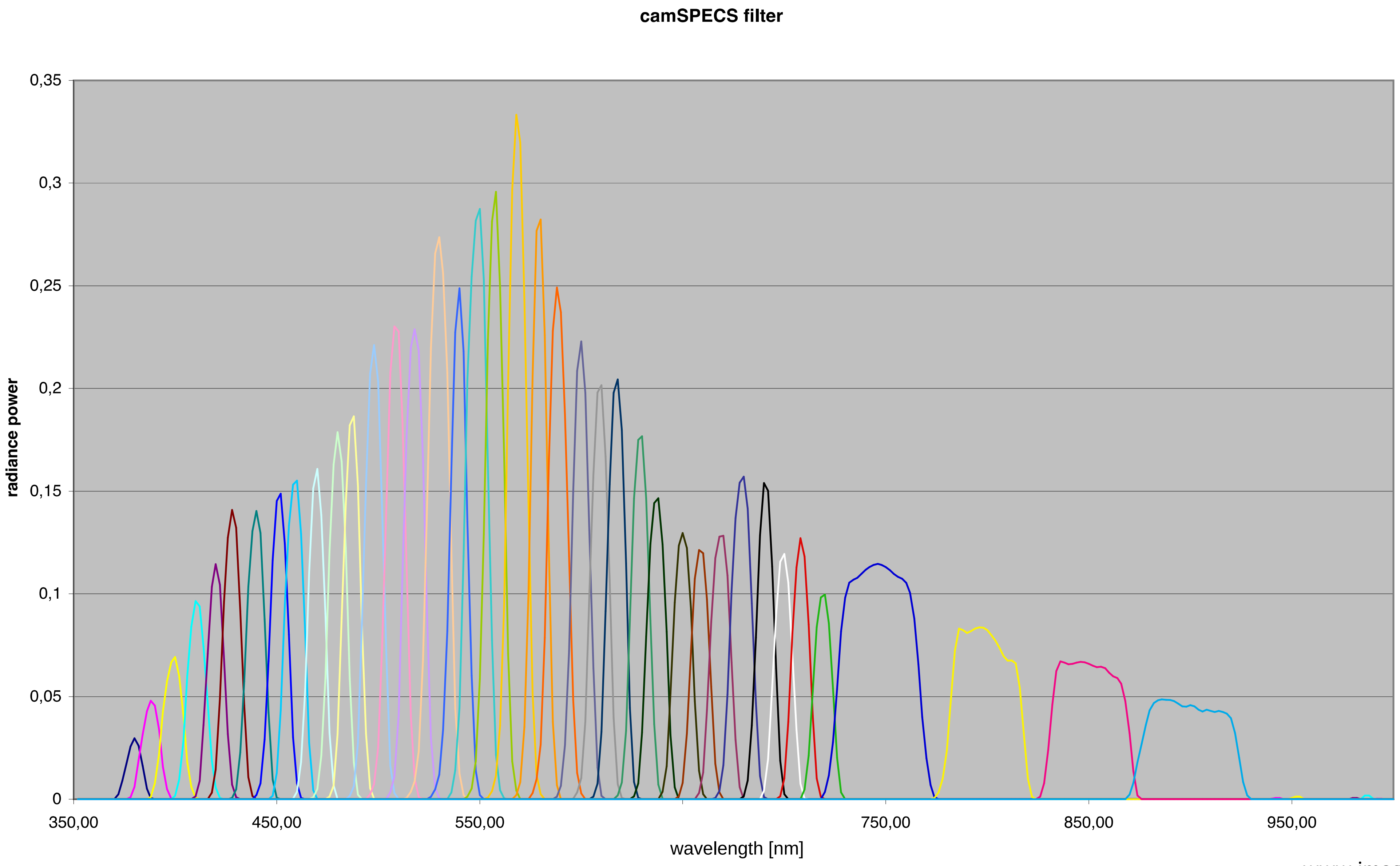




*http://chemwiki.ucdavis.edu/Analytical_Chemistry/*Analytical_Chemistry_2.0/10_Spectros

Filter based system

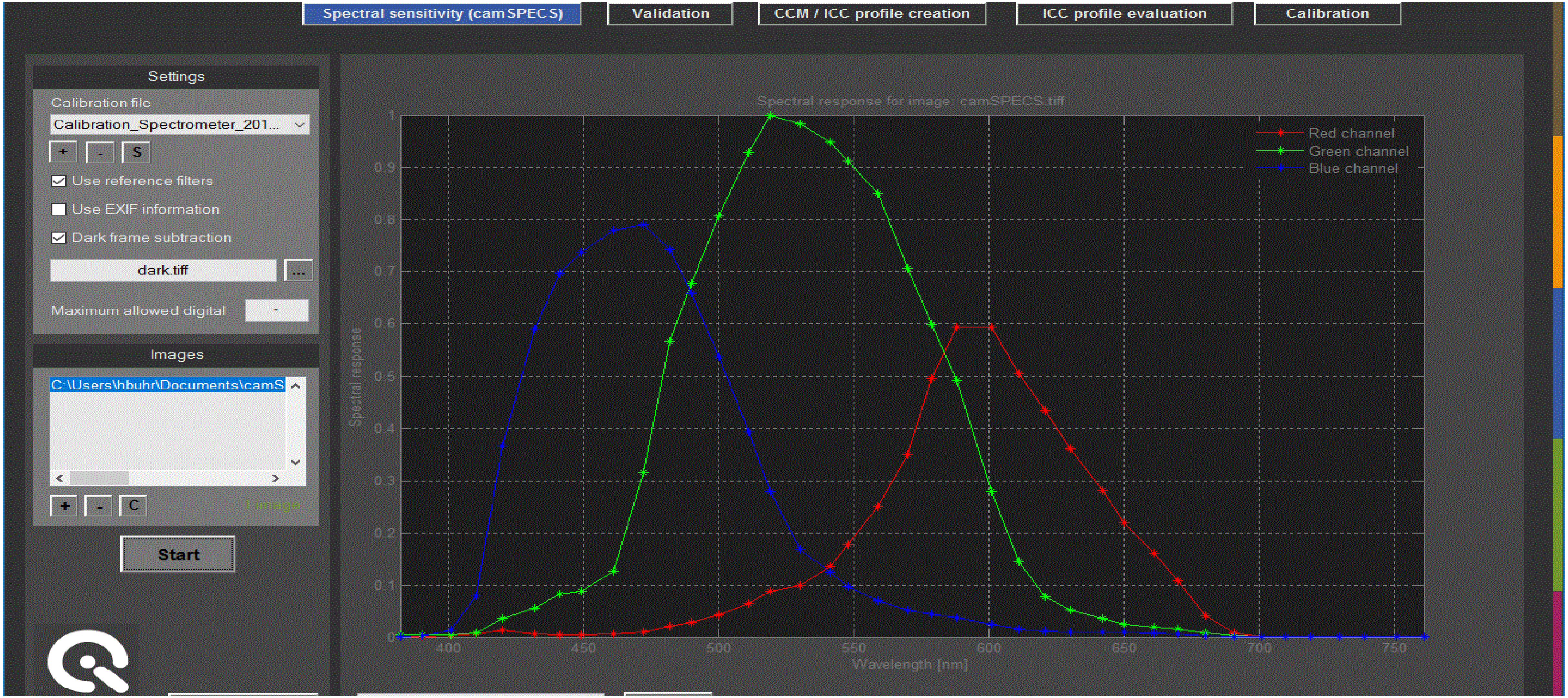
Interference filter



Filter based system

RAW image capture of 39 filter

Filter based system



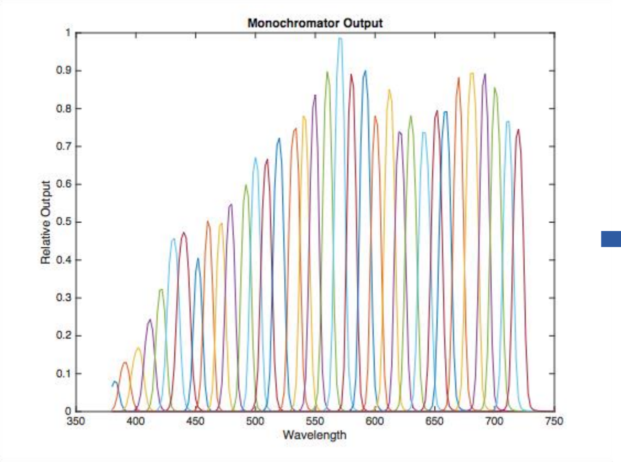
Direct measurement

Camera Capture

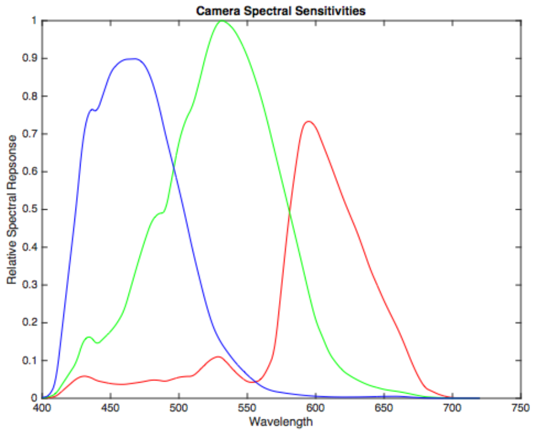
Camera Output

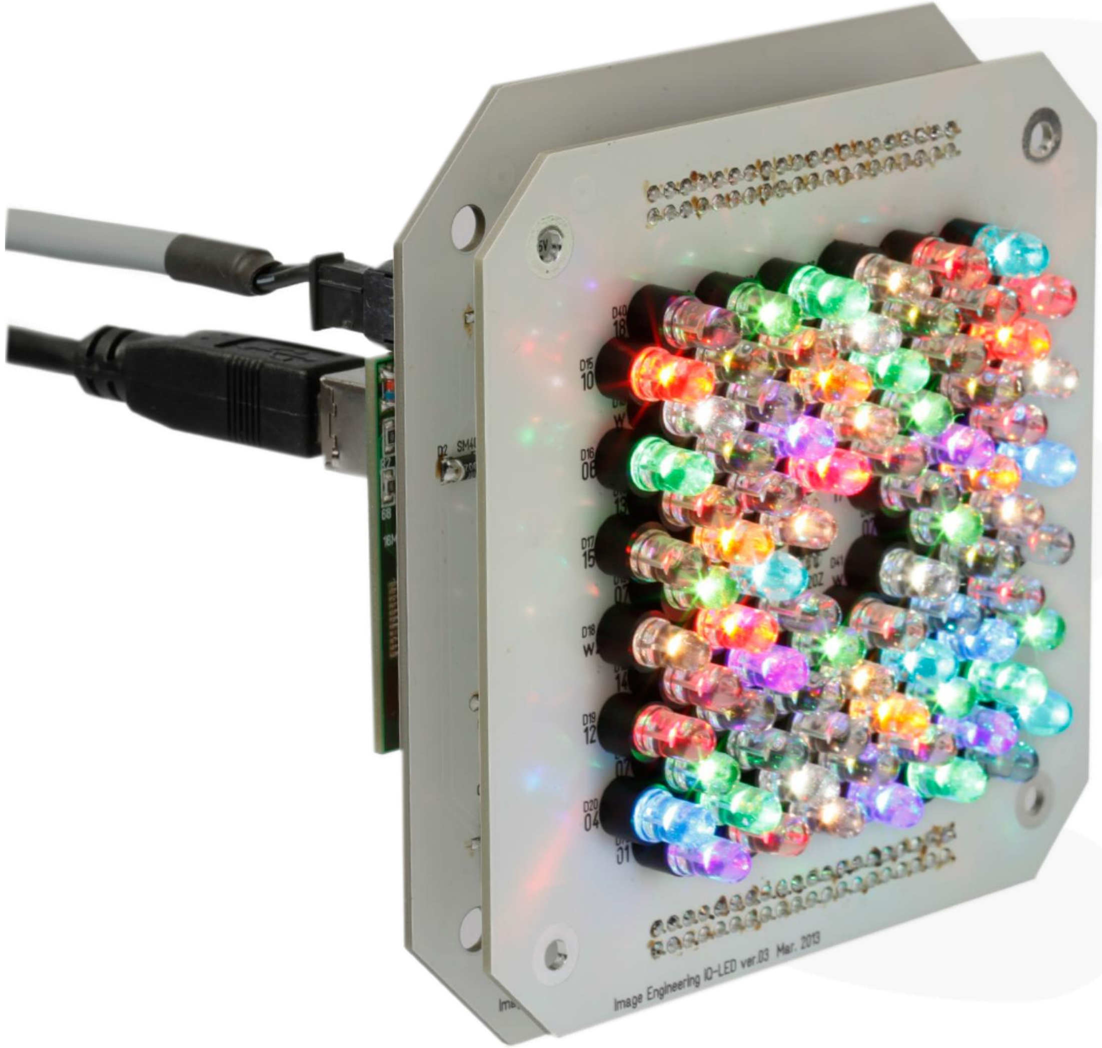
need current clean spectra examples

Monochromator Output



MonochromatorSpectra

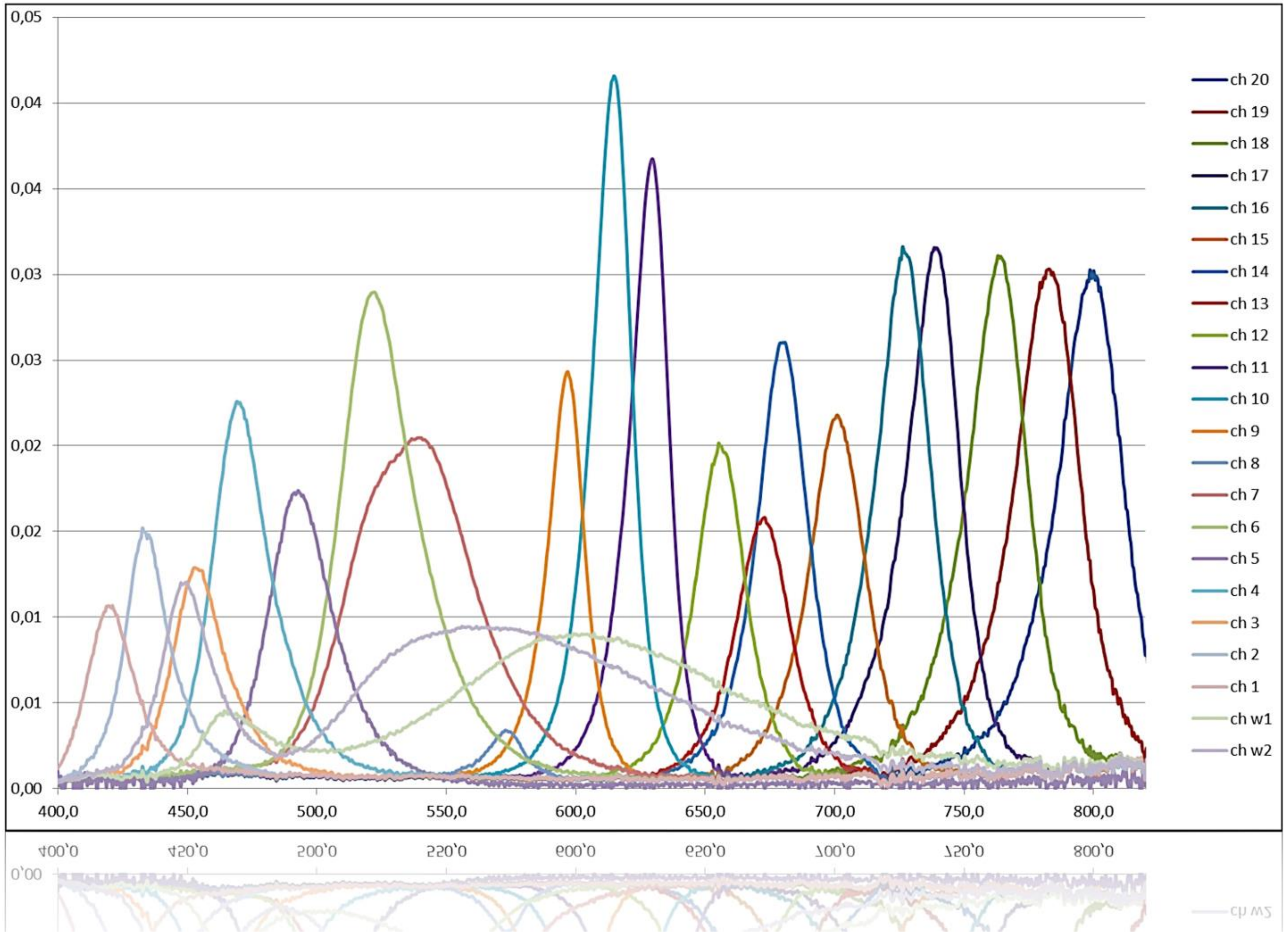




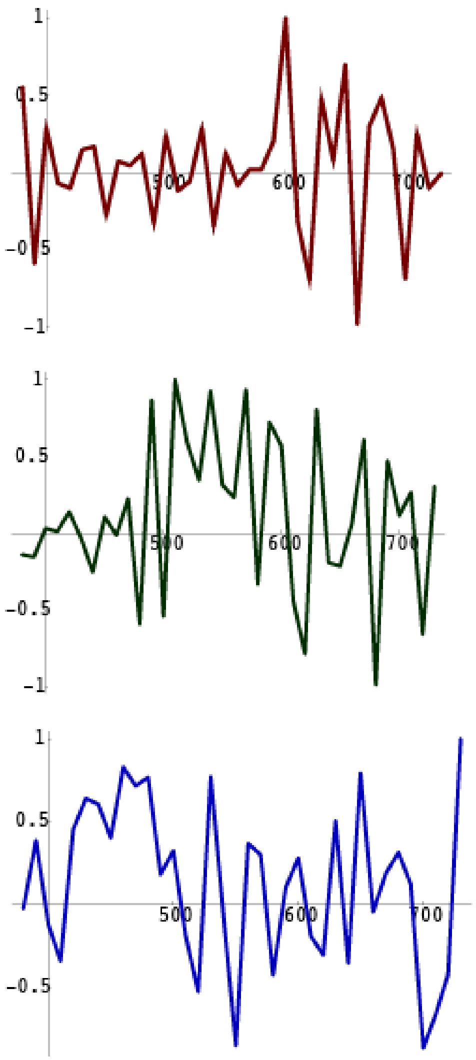
IQ-LED

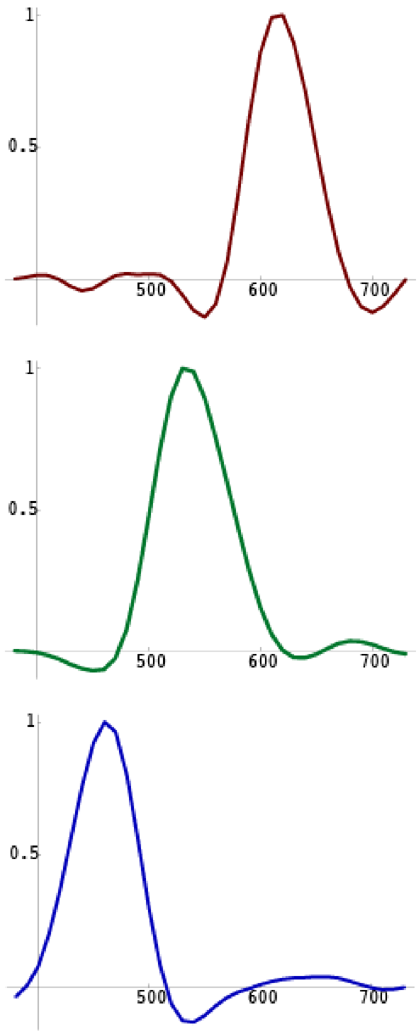
iQ-LED for calibration

Spectra of the 22 different channels



Estimation

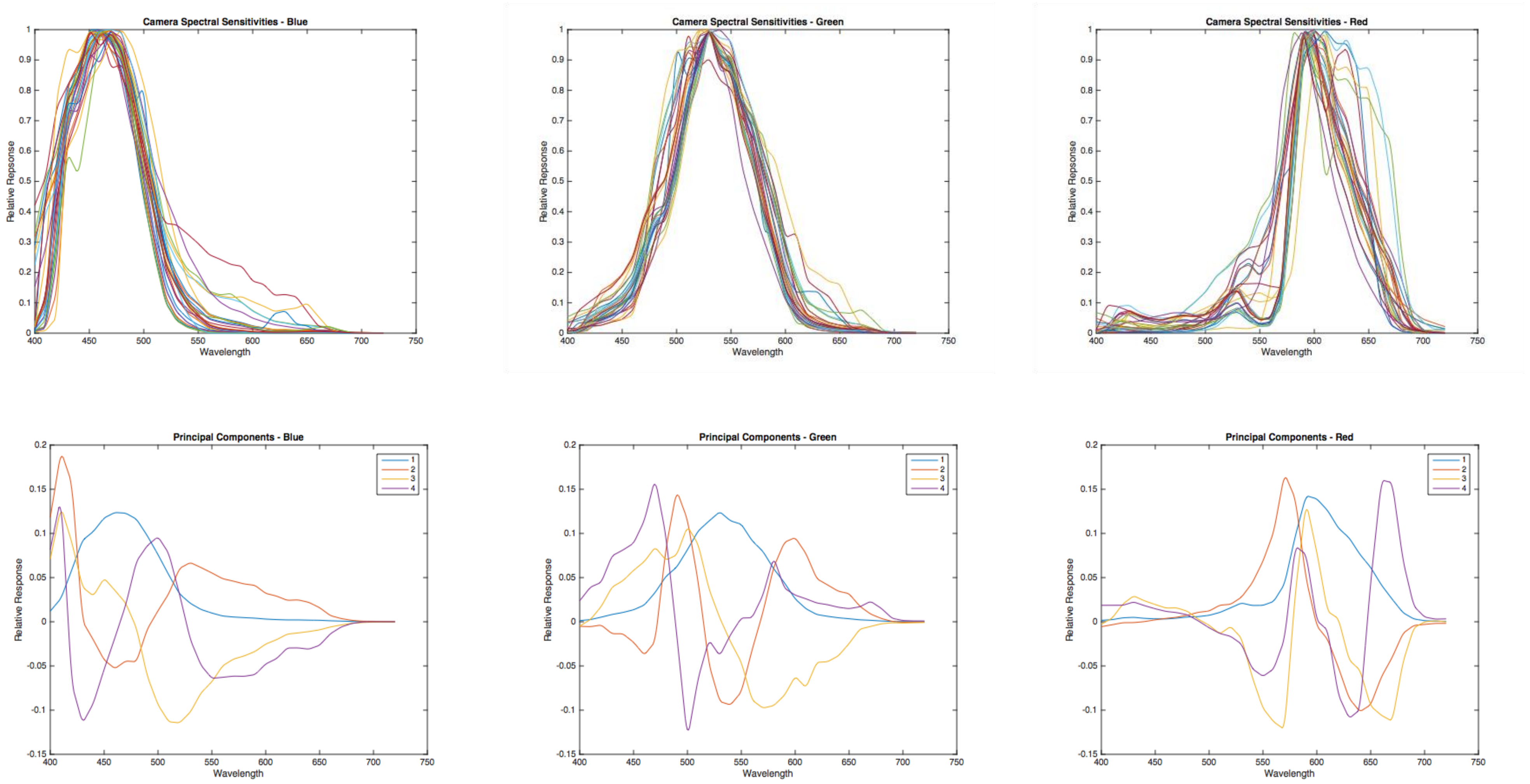




Without any knowledge about the expected sensitivity, many solutions are possible

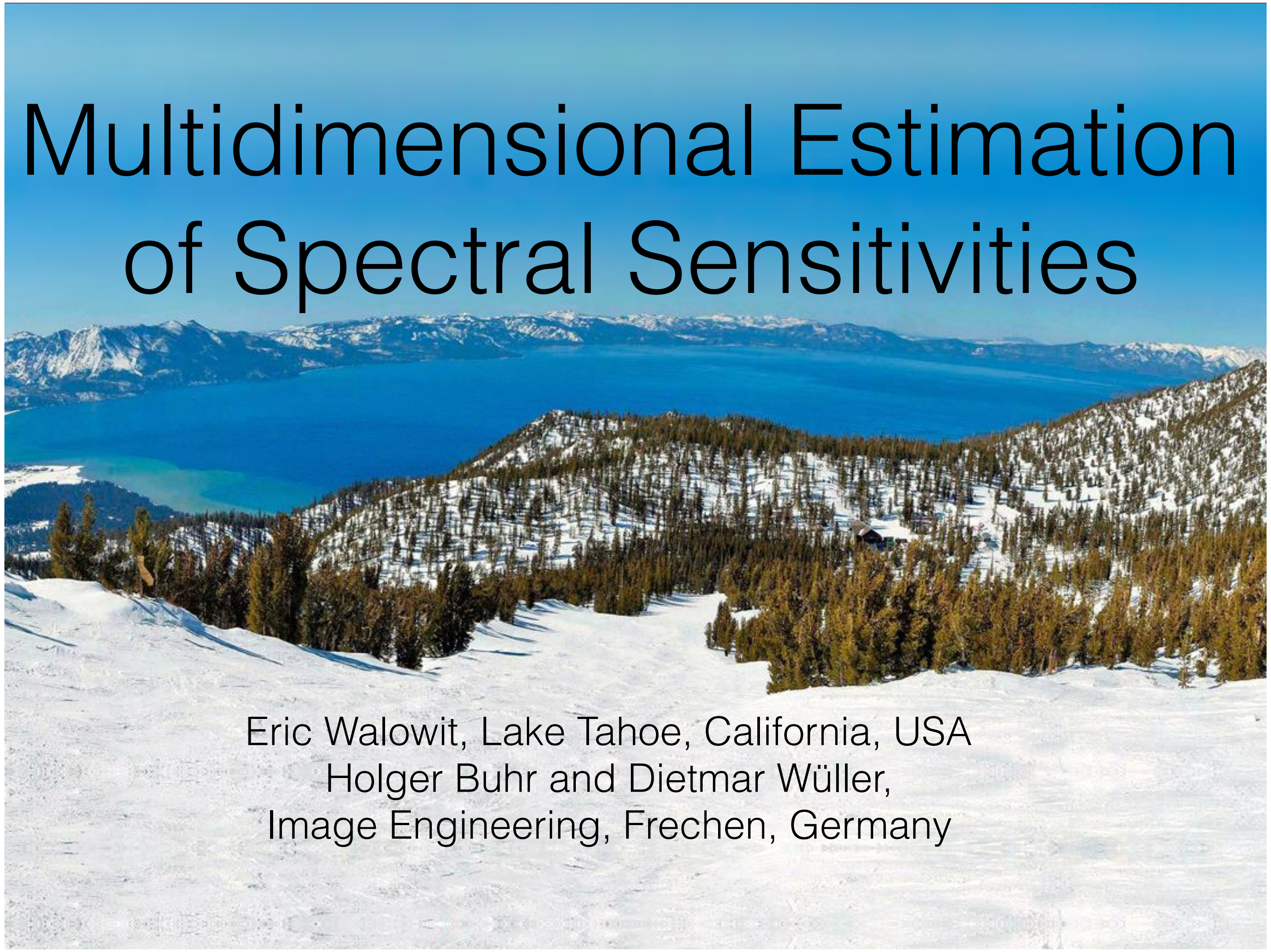
Camera dataset

Camera Dataset from Jiang et al. (RIT)



Dataset used for a principal component analysis (PCA)

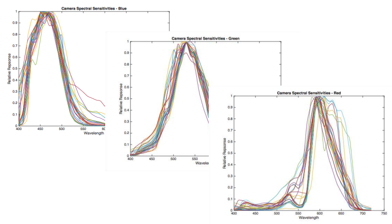
Paper at CIC25



Core concept

R&D

direct measurements (>15 devices)

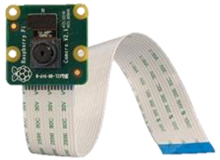


alternative: *spec. sens. from literature*

camera database

production line

iQ-LED base device



sample device (multiple, one by one)

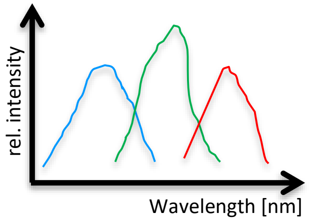
~ weekly recalibration

iQ-LED calibration data

captured images

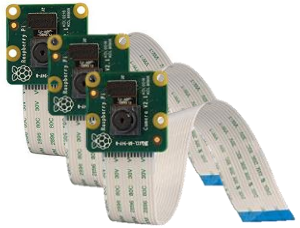
22 images per sample

spectral sensitvity estimation

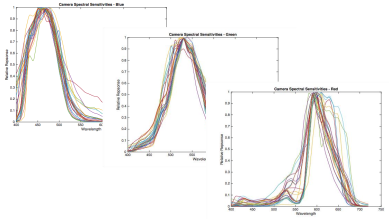


Core concept

R&D

direct measurements (>15 devices)



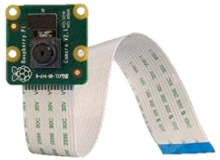
alternative: *spec. sens. from literature*

camera database

Done by Image Engineering with devices (>15) or data from customer!

production line

iQ-LED base device



sample device (multiple, one by one)

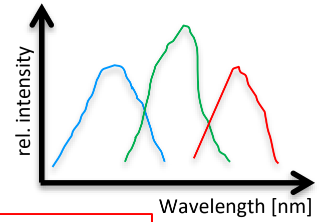
~ weekly recalibration

iQ-LED calibration data

captured images

22 images per sample

spectral sensitvity estimation

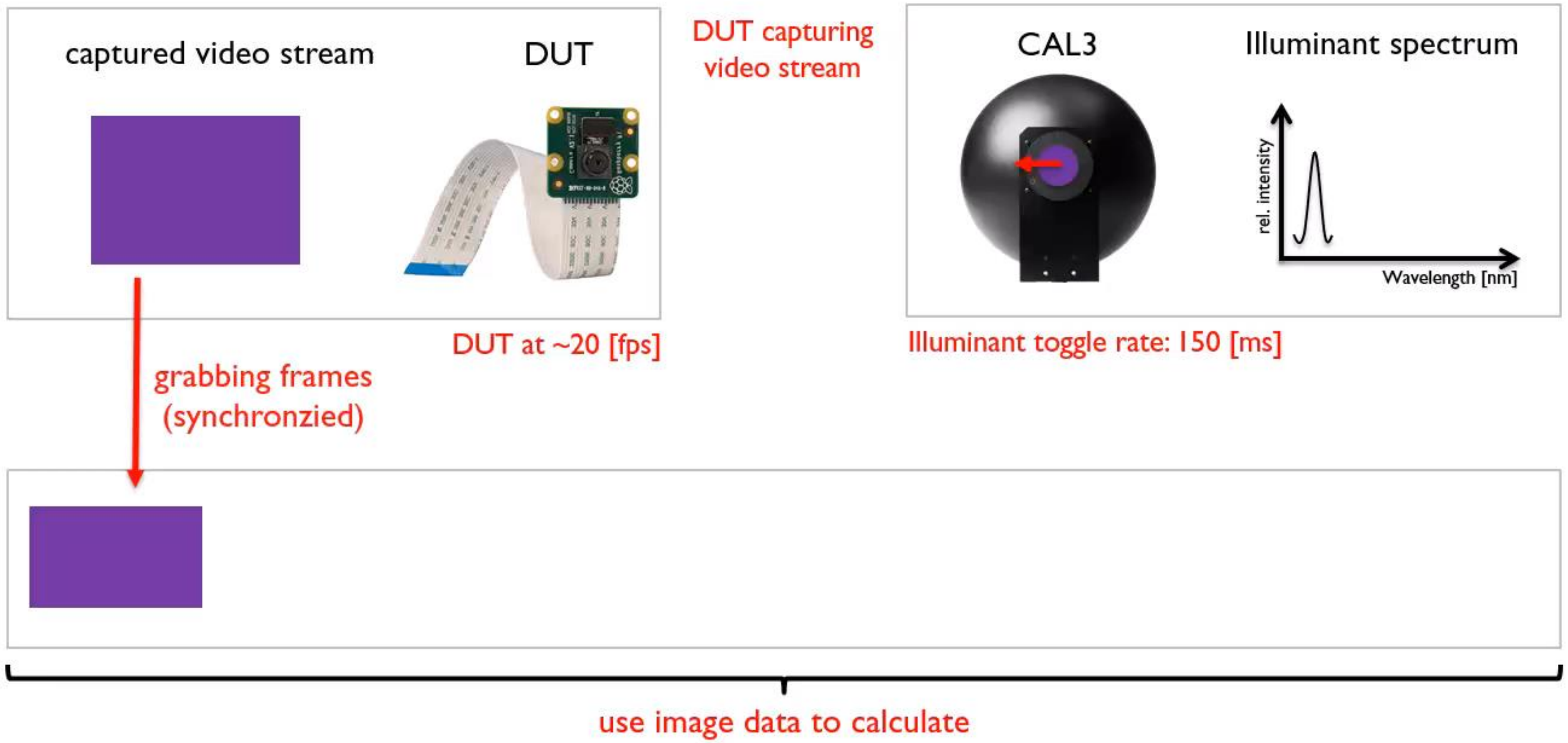


End-of-line implementation with Image Engineering and Trioptics products!

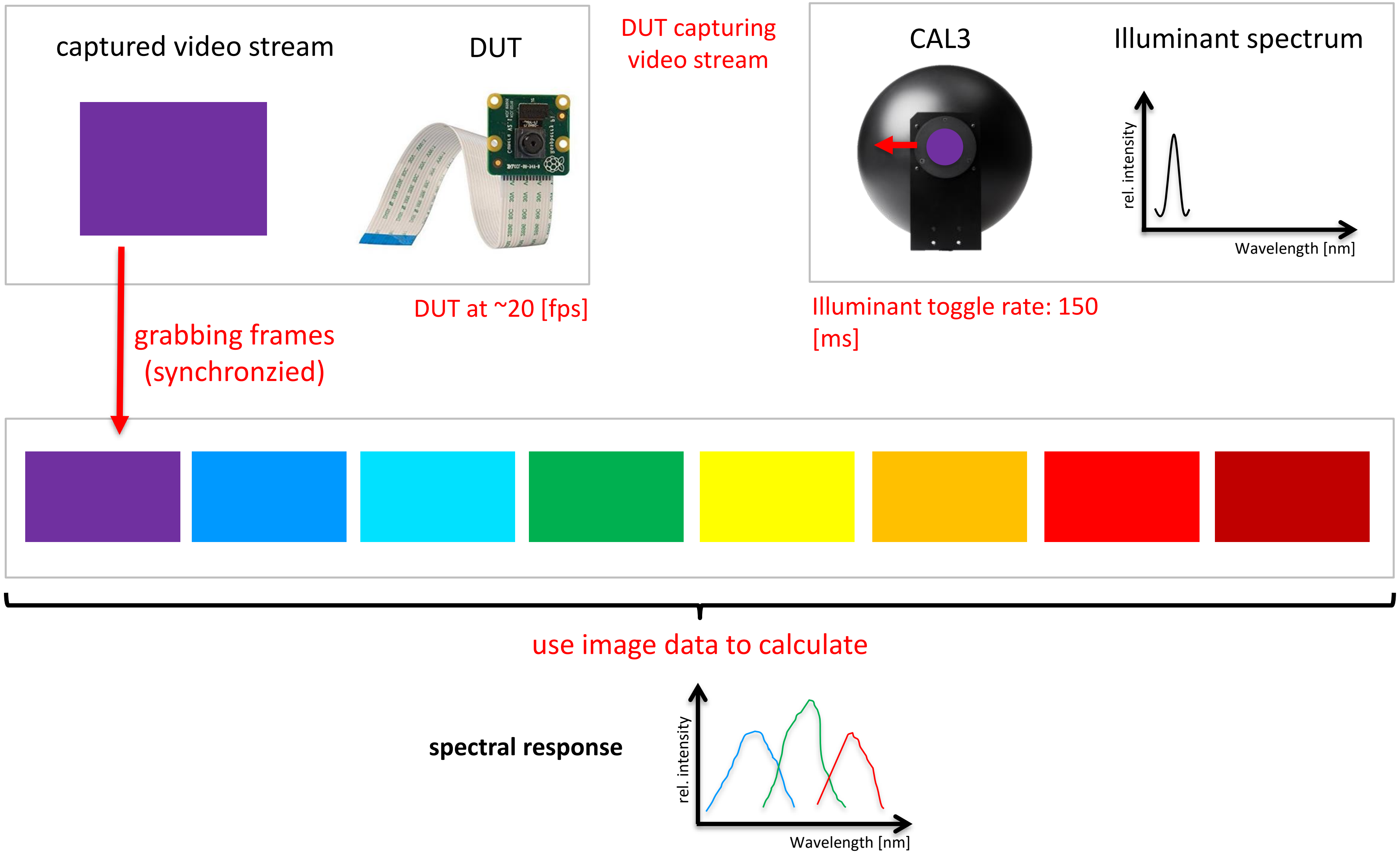
Captured data

each device captures all channels + black + D65 = 22 images (“flat field”)

Spectral sensitivity measurement

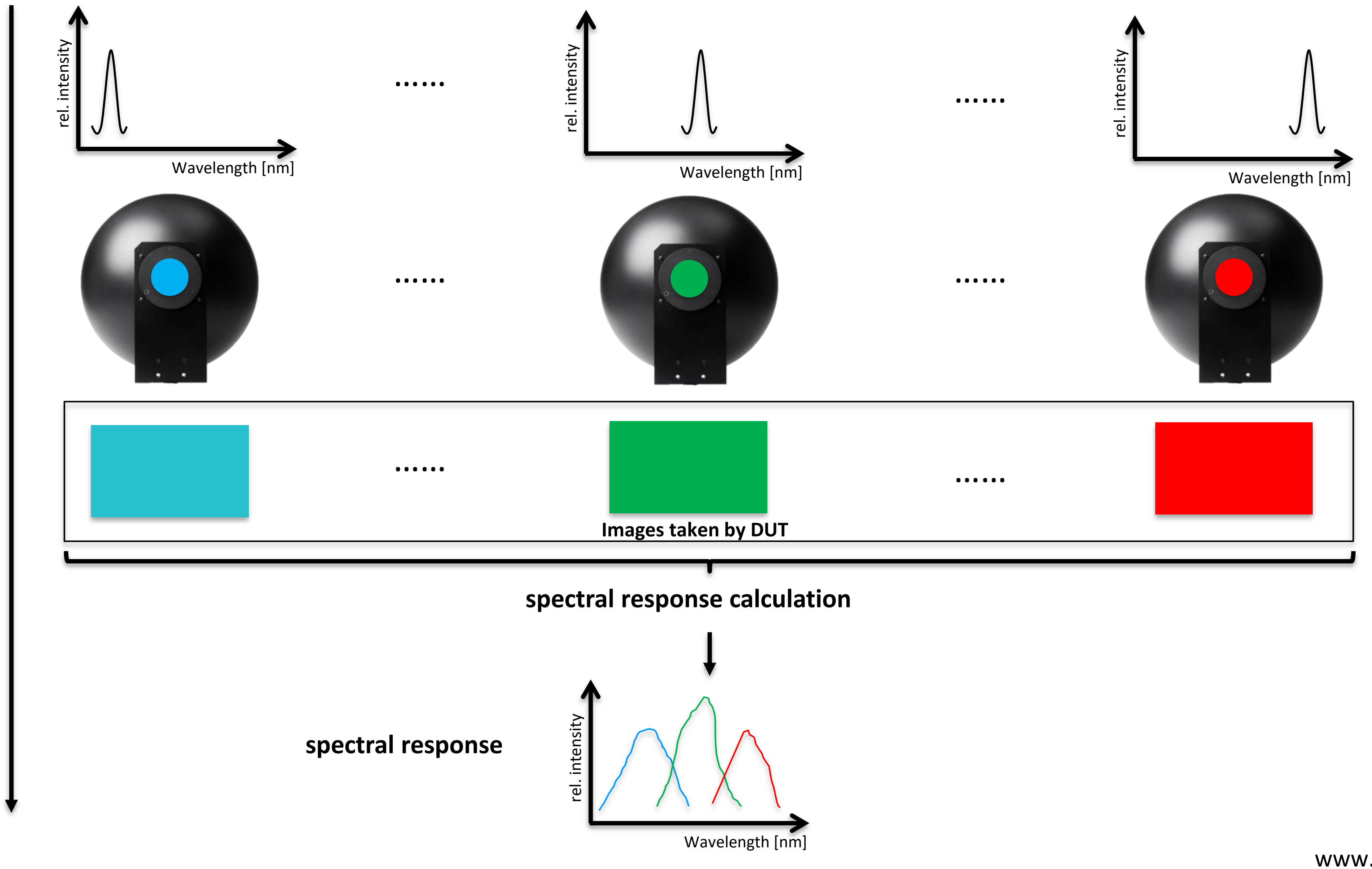


Details on workflow



Details on workflow

Capture all 20 single LED spectral distributions (+black +std. illu.)



Process time estimation

- spectral response toggles 22 illuminants each at 150 [ms]

  22 * 150 [ms] = **3300[ms]**
- camera should capture a video stream >20 [fps]
- frame grabbing from camera stream has to be synchronized for more accurate results
- depending on camera‘s transfer rate, start up time etc. additional

  ~ 2500 [ms]
- handling and report time

  ~2500 [ms]

total time: ~ 8.3 [sec]

Analysis software „camSPECS“