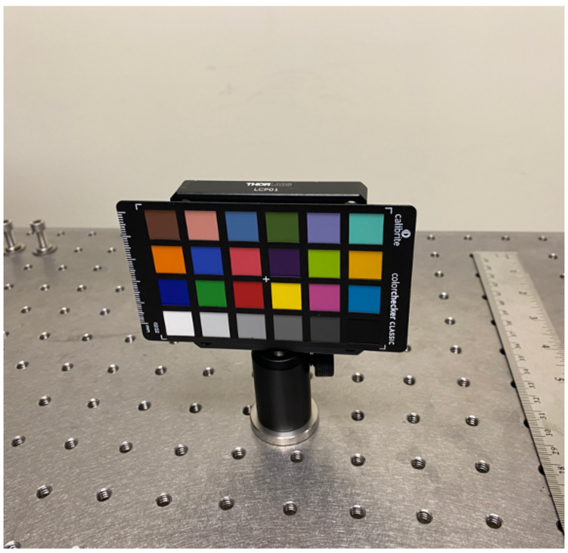
Camera Color Calibration and Radiant Colorimeter Exploration - Michelle Meng Shi

Team Members: Michelle Meng Shi

Introduction:

Camera sensor color calibration is a very general question every engineer working in the field would face. To make the camera 'see' color which human eyes would see requires certain calibration efforts. Camera sensor is usually manufactured on silicon wafers, so the spectral response of silicon material for converting photons to electrons at different wavelength ranges are different to how human eyes would respond. For example, silicon material would respond to NIR wavelength range and create signals which human eyes would not see, so filtering out signals out of visible wavelength range is necessary for higher accuracy data. In industry, the most general and direct way to characterize the spectral response of camera sensor is to use monochromatic light source at discrete wavelength for mapping, which requires a lot of time and experiments to be done repeatedly for the final results. In the first part of this project, Google Pixel 4 camera sensor spectral response curve will be estimated by the data which includes Macbeth color checker pictures under different illumination conditions. It is an interesting and quick and easy way to estimate the camera sensor spectral response without going to more lengthy experiments. In the second part of the project, the experiment is setup with Radiant Vision System colorimeter which simulates human color response curve, and the light source spectra were being estimated. [1]



Pic1. Followed Joyce's advice and setup a mini color checker in the lab.

Part 1: Estimate Pixel 4a Camera Sensor Spectral Responses

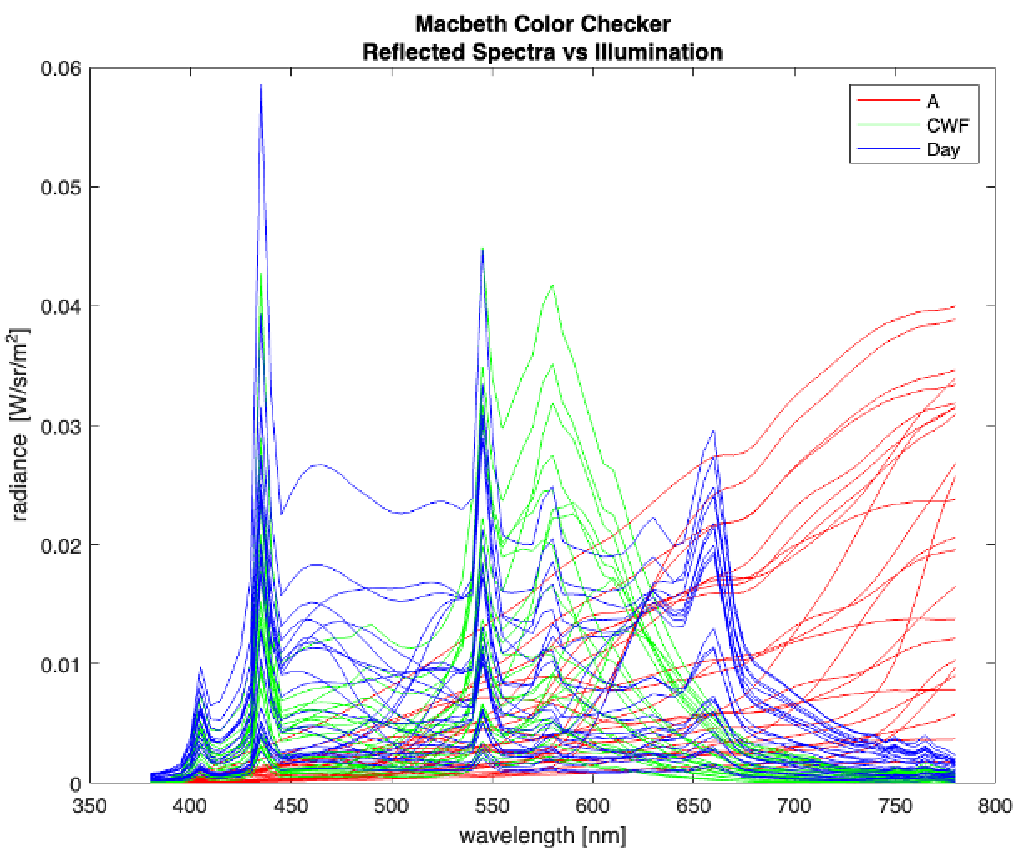
Data:

The experiment were setup with three different illuminations, illuminant A, cold white florescent and daylight lamp, with 24 color patch Macbeth color checker placed in the middle of the controlled box, see picture 2. The spectroradiometer measurements of 24 color patches under 3 illuminations were given, together with the three pictures taken by Pixel 4a camera. The Macbeth color checker reflected spectra vs. illumination under three different illumination conditions is plotted in picture 3. Three illuminations have very different radiance curve, illuminant A has increasing response towards higher wavelength range, illuminant CWF has bigger peaks in red and green wavelength range, while illuminant daylight lamp has multiple peaks throughout the visible wavelength range. The difference in illumination conditions would be helpful in creating more discrete data sets for estimating the solutions.

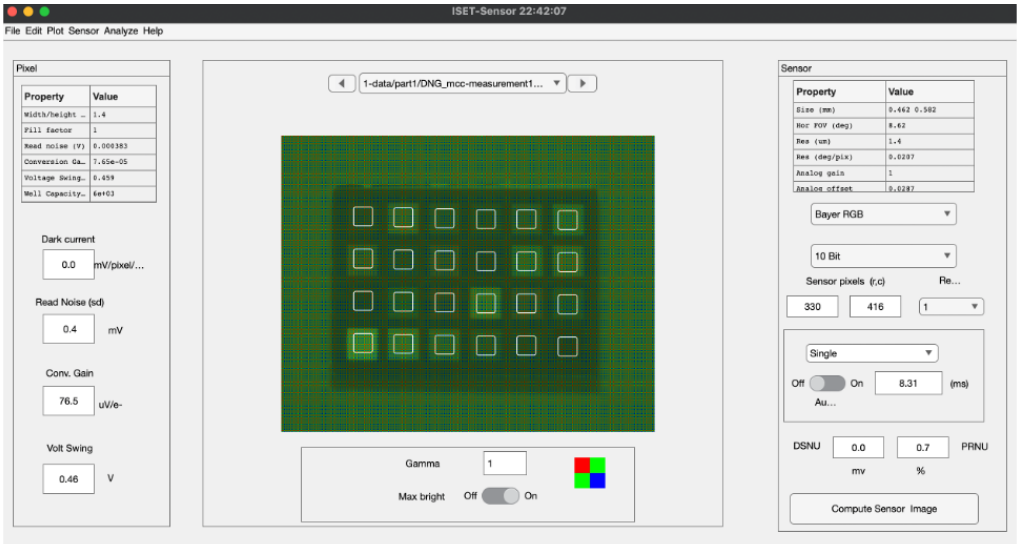
Using ISETCam functions, the Macbeth color checker patch RGB info can be extracted from the DNG files provided, see picture 4. If we look closely, ISETCam is showing the Bayer pattern of the image sensor with RGB filters. The RGB data in each color patch is then spatially averaged and extracted. The signal to noise ratio can be a good estimate in the data quality for the analysis. Looking at SNRs from three illuminants conditions, all of them showing more then 20dB values, which suggests the data has good quality for the following analysis.

Formula : SNR = 20 $\log_{10}$ (Mean / Variance)

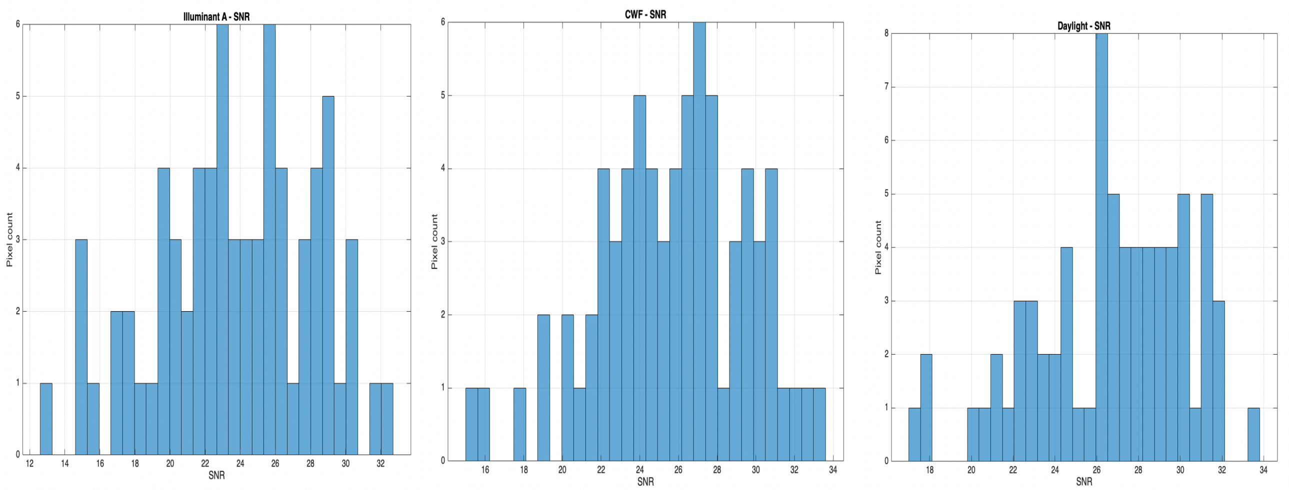
Pic2. Macbeth color checker original pictures under three illumination conditions.



Pic3. Macbeth color checker reflected spectra vs. illumination conditions.



Pic4. ISETCam camera sensor reading to extract color patch info.



Pic5. Color Patch signal to noise ratios under three different illumination conditions.

Model and Optimization Methods:

Model:

The model can be understand as following:

linearRGB = Camera Spectral Response * Scene Spectra

We are looking to estimate camera spectral response by given linear RGB values and the spectroradiometer data of 24 color patches under 3 illuminations which is the scene spectra. Three optimization methods will be used in the following session.

I. Direct Method - Pseudo-inverse:

Pseudo-inverse is a direct and simple method to estimate a matrix inverse problem. We are looking to solve camera spectral response by given linear RGB values and Scene Spectra, by using pseudo-inverse, which is a function provided by Matlab, the equation can be written as below:

Using simple linear algebra:

Here A = linearRGB, B = Camera Spectral Response, C = Scene Spectra

$A = B * C$

$A * C^{-1} = B * C * C^{-1}$

$A * C^{-1} = B * I$

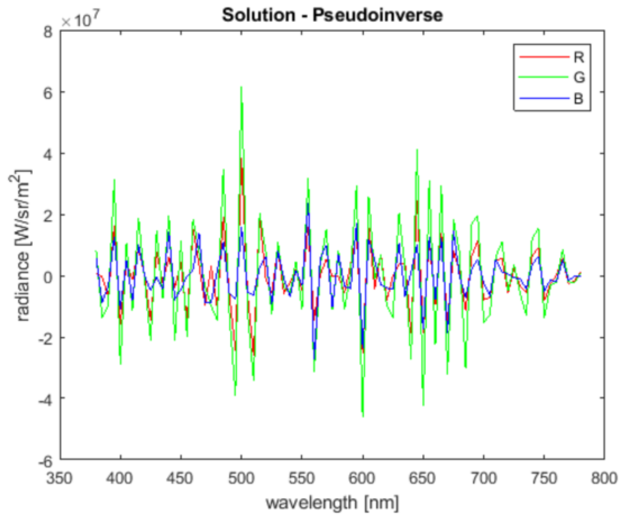
$B = A * C^{-1}$

So we get:

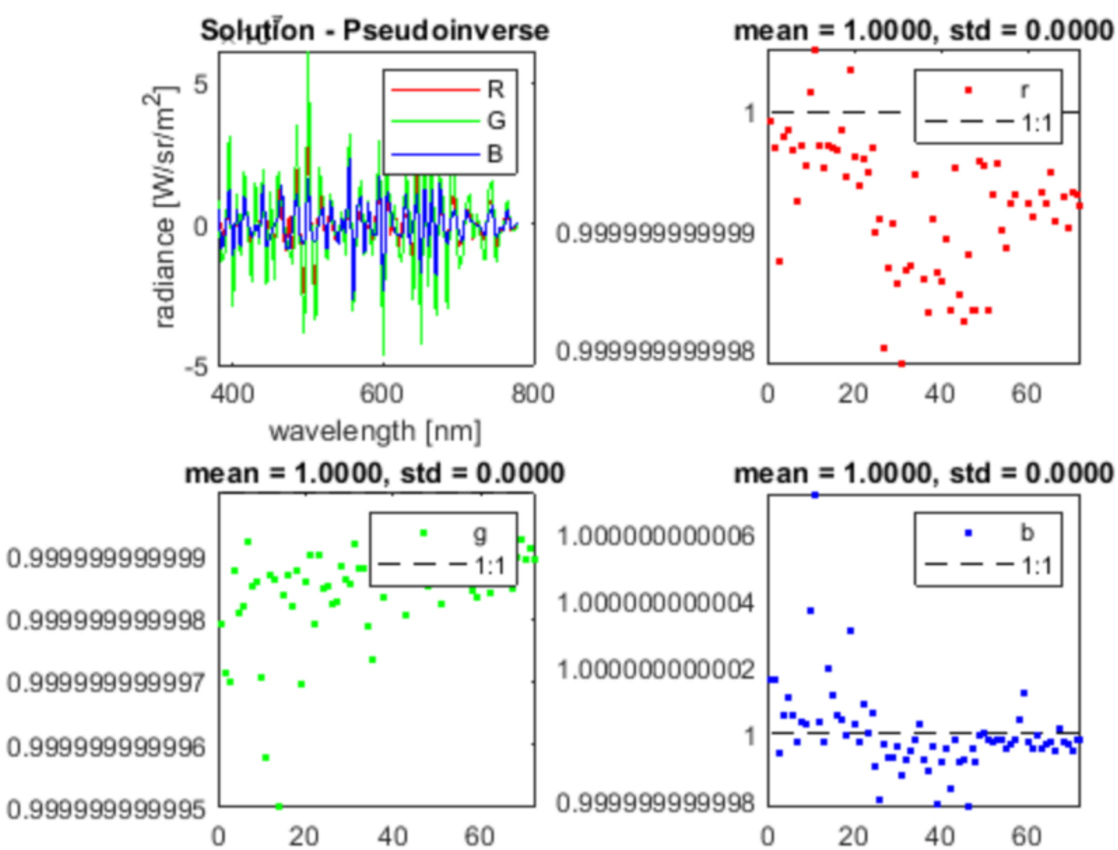
Camera Spectral Response = linearRGB * pinv (Scene Spectra)

Results:

The results with pseudo - inverse show lots of peaks across all visible wavelength range, which is not realistic for a camera sensor to have such response, see picture 6. Even though the comparison between measured and estimated data are very close to each other, see picture 7. The results is not acceptable as a solution to the real world problem. The reason can be that pseudo - inverse tend to over fitting the data which may result in unwanted results.



Pic6. Pseudo-inverse solution with RGB channels.



Pic7. Pseudo-inverse solution measured vs. model predicted values.

II. Direct Method - Truncated SVD:

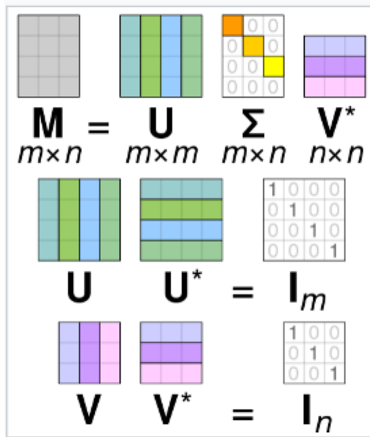
The second method used here is truncated SVD (singular value decomposition), which can be explained in the picture 8 below. A matrix M and be decomposed as three matrix multiply together, the center matrix sigma is representing the singular matrix, with terms being adding more values along the diagonal line of the matrix. Based on the model above:

Camera Spectral Response = linearRGB * inverse (Scene Spectra)

We can decompose Scene Spectra matrix to three terms:

Scene Spectra = U * Sigma * V'

Here we would start from 1 diagonal term in the Sigma matrix to higher terms, compare the measured and estimated solution, at the same time, using human judgement on curve shapes which may represent the real sensor spectral response curves.



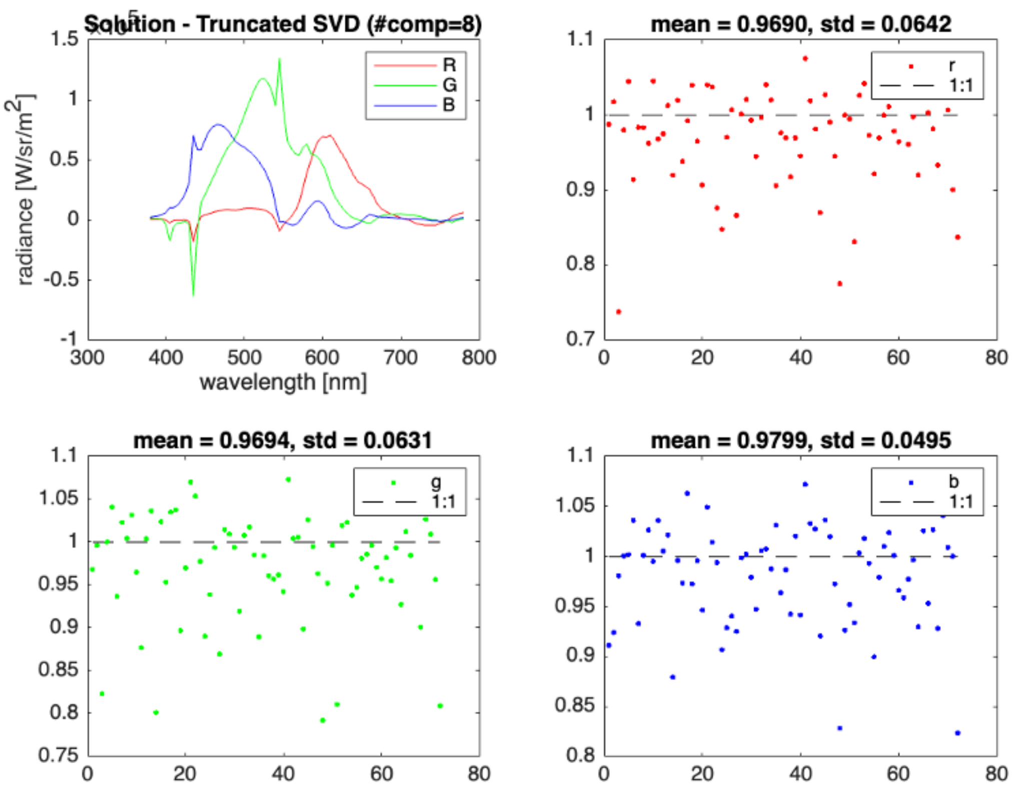
Visualization of the matrix multiplications in singular value decomposition

Pic8. Wikipedia SVD method explanation.

Results:

We found out by using 8 terms in SVD method, we are getting reasonably looking camera spectral response curve, with acceptable differences between measured and estimated solutions, see picture 9.

Looking at the spectral response curves, we are still seeing some unrealistic negative values in RGB curves, with some peak responses in random area. The results are better than the pseudo-inverse results, but still we are looking for space of improvement to optimize the solution.



Pic9. Truncated SVD solution measured vs. model predicted values.

III Direct Method - ADMM:

ADMM which stands for alternating direction method of multiplier is a very powerful tool in solving optimization problems. In our case, since camera spectral response should not be negative values. It translates to a non-negative least square case in ADMM, which can be illustrated in picture 10.

We are looking to optimize x (Camera Spectral Response), while we are adding constraints to y following the updating sequence. [2]

There are three constraints we are putting in y to optimize the solution:

(1) non-negative least squares (NNLS)

We will force y to drop negative values after x update. So the camera spectral response x will optimize towards positive values.

(2) spectral response smoothness

Since the camera spectral response curves usually do not have random spikes, the curves should look much smoother.

We would impose a moving average constraint, so within 15nm of wavelength window, if x is showing the trend of forming spikes, it will be optimized to take smoother value.

(3) minimize reconstruction error

The optimizing problem includes a term to compare the measured value and the estimated value, and to minimize the difference between both values in the update rule in the updates.

By constructing the ADMM optimizer with the constraints, 1500 iterations were computed and the spectral response curves are being optimized in the updates.

Solving NNLS by ADMM algorithm

$$(\mathcal{P}) \;:\; \underset{\mathbf{x}\geq 0}{\operatorname{argmin}}\; f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2.$$

Algorithm 1: ADMM for NNLS

Result: A solution $\mathbf{x}$ that approximately solves $(\mathcal{P})$

Initialization Set $\mathbf{x}_0, \mathbf{y}_0 \in \mathbb{R}^n_+$, $\lambda_0 \in \mathbb{R}^n$, $\tau > 0$, $\mu_0 = \tau^{-1}\lambda_0$, $k = 1$

while *stopping condition is not met* **do**

$$\mathbf{x}_k = (\mathbf{Q}^\top\mathbf{Q} + \tau\mathbf{I})^{-1}\Big(\mathbf{Q}^\top\mathbf{p} + \tau(\mathbf{y}_{k-1} - \mu_{k-1})\Big)$$
$$\mathbf{y}_k = [\mathbf{x}_k + \mu_{k-1}]_+$$
$$\mu_k = \mu_k + \mathbf{x}_k - \mathbf{y}_k$$
$$k = k + 1$$

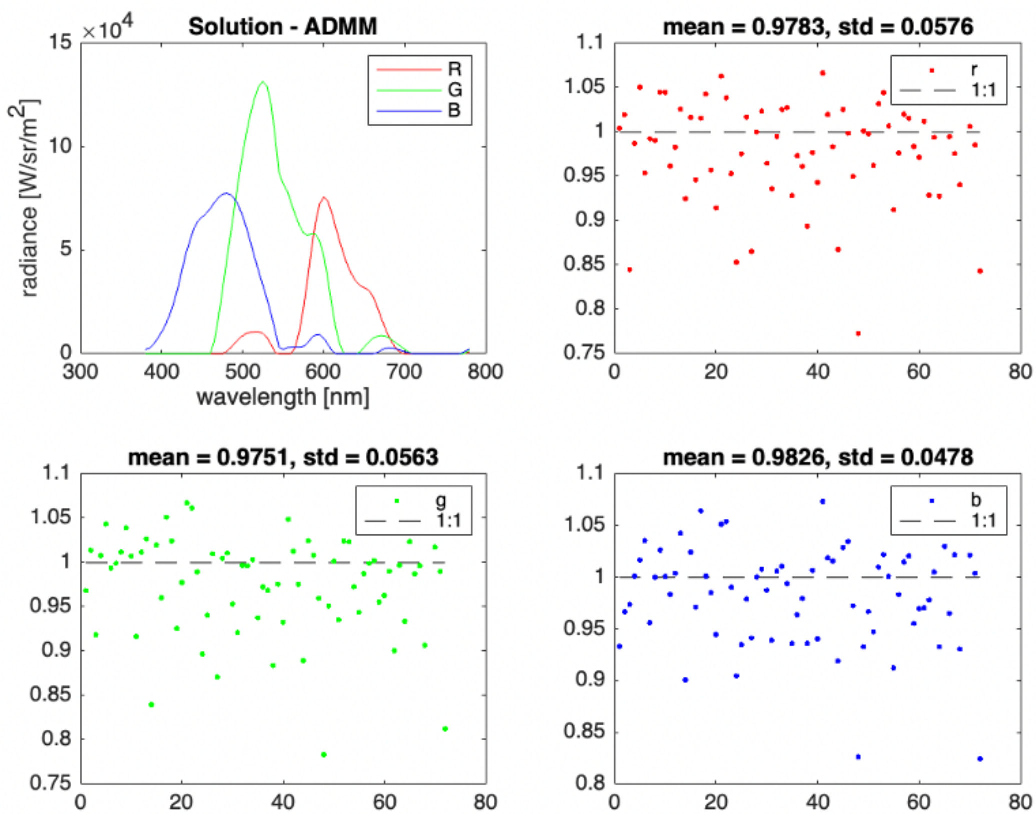
end

Implementation issues : constant terms should be pre-computed outside the loop.

Pic10. Solving NNLS by ADMM algorithm. Reference: https://angms.science/doc/NMF/nnls_admm.pdf

Results:

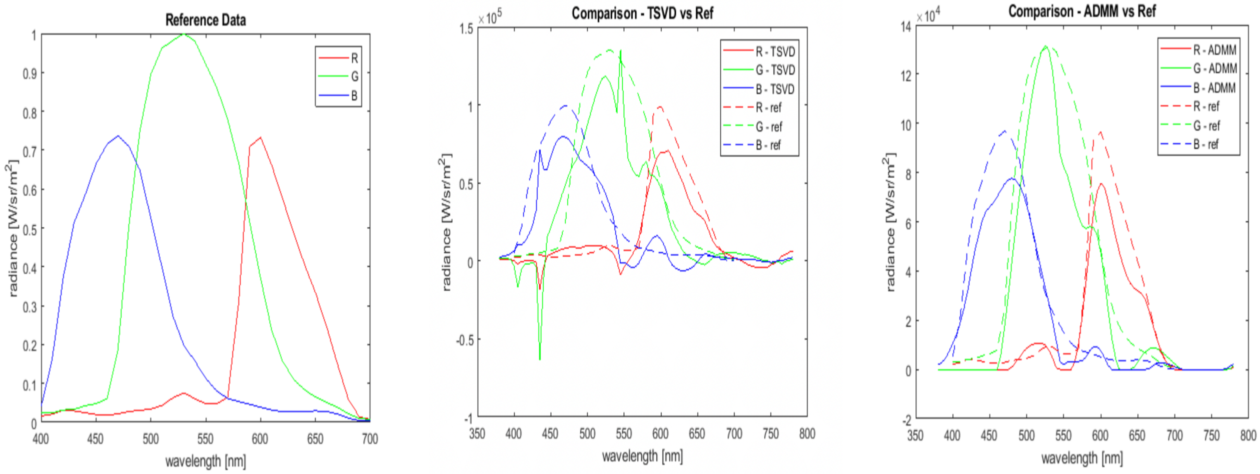
The results from 1500 iterations of ADMM optimization method is shown in picture 11. We can see that the R G B spectral response curves are having positive values with smoother looking curves, which is more realistic looking compare to the previous two methods. The measured and estimate value comparisons are showing reasonable mean and variance values.



Pic11. ADMM solution measured vs. model predicted values.

Validation:

Based on Joyce's reference suggestion, Pixel 4 camera response curve was downloaded and plotted, see picture 12. It is very interesting to see that the SVD method shows general trend with some negative and peak values. The ADMM performs the best with non-negative value and smooth curve, the wavelength response curves are almost matching the reference data. So we can see here ADMM optimizer is giving us reasonable estimates and being the best method among the 3 solutions we worked on. 3



Pic12. Reference sensor spectrum compare to SVD and ADMM predicted spectra.

Part 2: Estimate Light Source Spectrum Using Calibrated XYZ Camera

Experiment Design and Setup:

The second part of the project is designed to estimate the light source spectra from the Radiant Camera XYZ output and spectral response.

The experimental setup is shown in picture 13. Two spectrum unknown light sources were used in the experiment, illumination 1 is from the ceiling LED light in the lab, illumination 2 is from a high brightness LED flashlight. The picture is showing the Radiant camera setup with the mini Macbeth color checker. The focal distance is around 0.3m, with 43MP sensor. The large field of 120deg x 80deg view image is the PNG picture output from the Radiant camera.

Radiant Vision System Colorimeter is designed to simulate human eye response to color, by using four color filter wheels Red X, Red Xb, Green Y and Blue Z and taking four images with the monochromatic sensor and combine the output in true color format. It can also output X, Y, Z values separately for color analysis, see picture 14. The lens used in the experiment is a wide field of view AR / VR lens, which simulate human eyes with small entrance aperture and long lens structure, see picture 15. 4

The model is shown below:

XYZ from Radiant Colorimeter Output = Radiant Camera Spectral Response * diag(Light Source Spectrum)* MCC Reflectance Spectrum

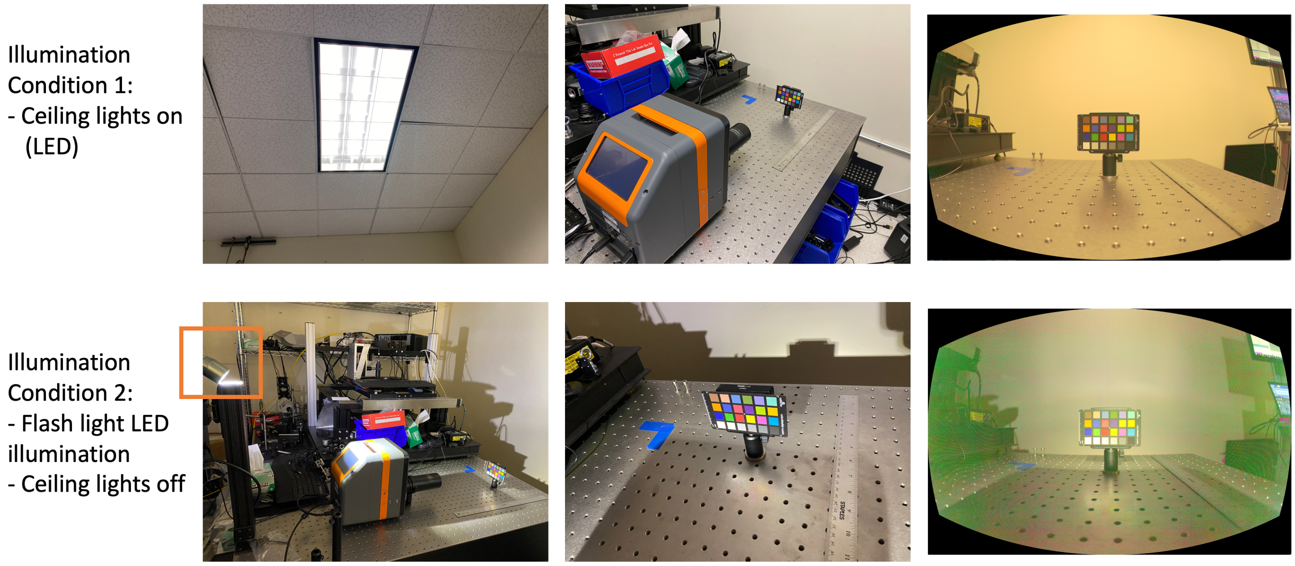
Here the Radiant Camera Spectral Response = CIE color matching functions, Light Source Spectrum is a diagonal matrix

We are looking to solve Light Source Spectrum by knowing the XYZ output, CIE color matching function and Macbeth reflectance spectrum.

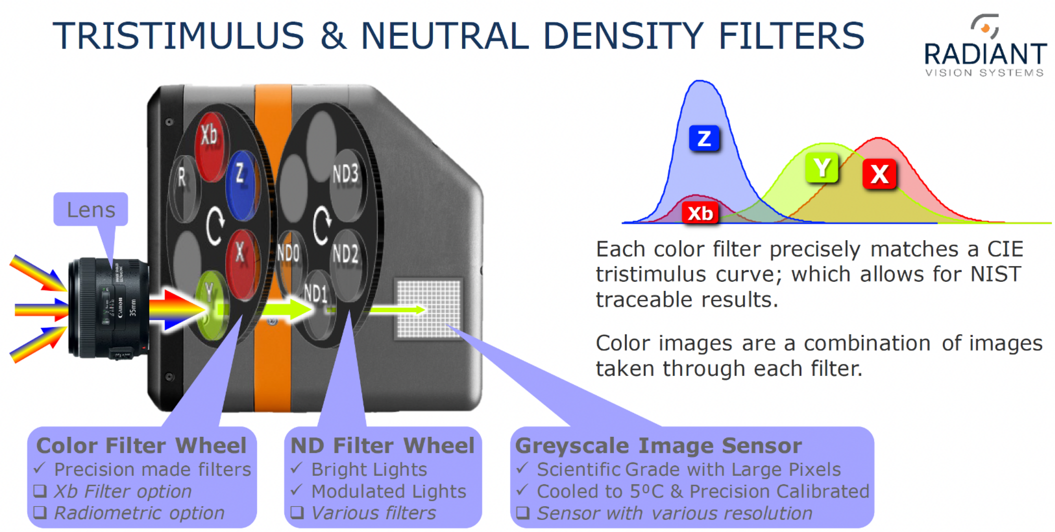
To simplify mathematically, the model can be re-written using vectorizing matrix as below by combining Radiant Camera Spectral Response and MCC Reflectance Spectrum part to solve for Light Source Spectrum.

XYZ from Radiant Colorimeter Output = (Radiant Camera Spectral Response _ MCC Reflectance Spectrum Matrix)* Light Source Spectrum

Here the Radiant Camera Spectral Response = CIE color matching functions



Pic13. Optics lab experimental setup under two illumination conditions.

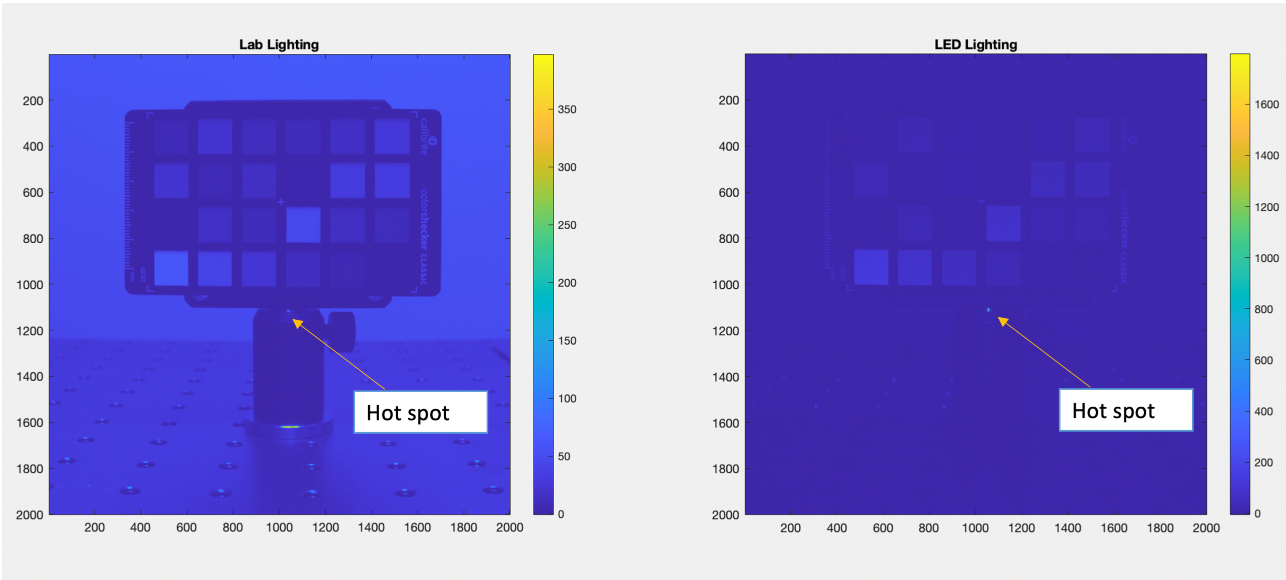


Pic14. Radiant vision system colorimeter filter wheel and working principal explanation.

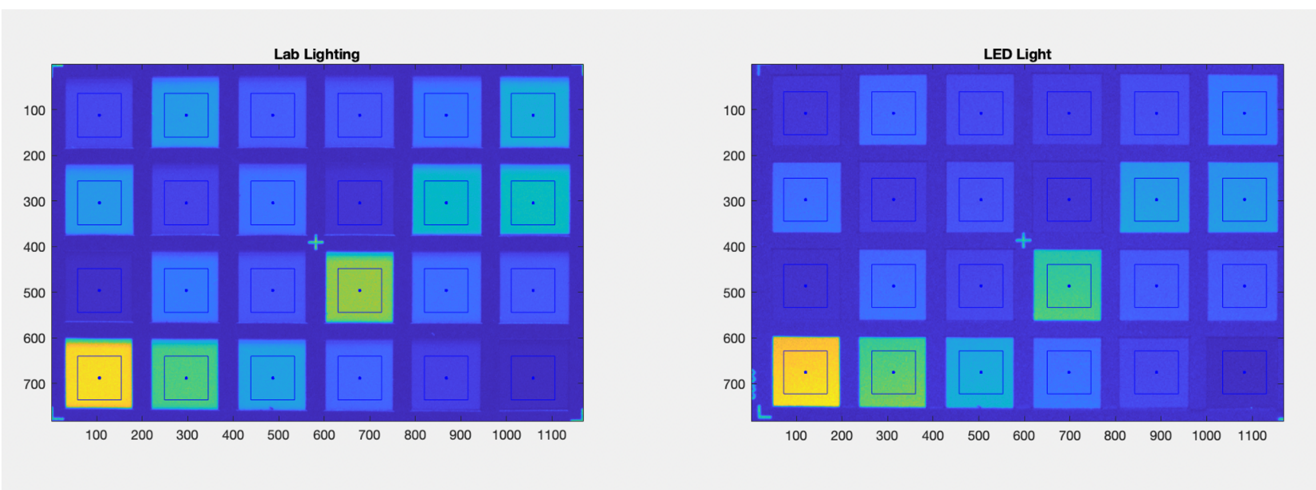
Pic15. Radiant vision system AR/VR lens explanation.

Data:

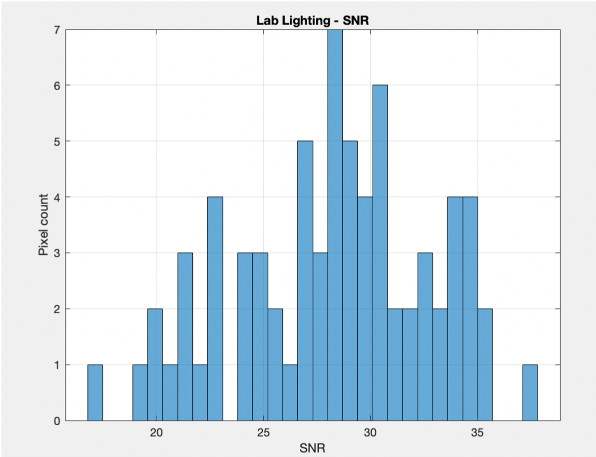
The images of Macbeth Colorchecker illuminated under 2 unknown light conditions in the lab were taken. The camera output XYZ values directly for the image. By cropping the Luminance channel Y image into a smaller area, see picture 16. We found there are some hot spot which shows very high luminance value due to the metal post reflection from the light source. The images are further cropped to including only the color patch, see picture 17. XYZ data from each color patches with 50% cropping ratio is collected for two lighting conditions. After examining the SNRs of the two lighting conditions, we can see that the SNRs are acceptable for both illuminations with lab lights showing higher SNR values compare to LED flashlight, see picture 18.

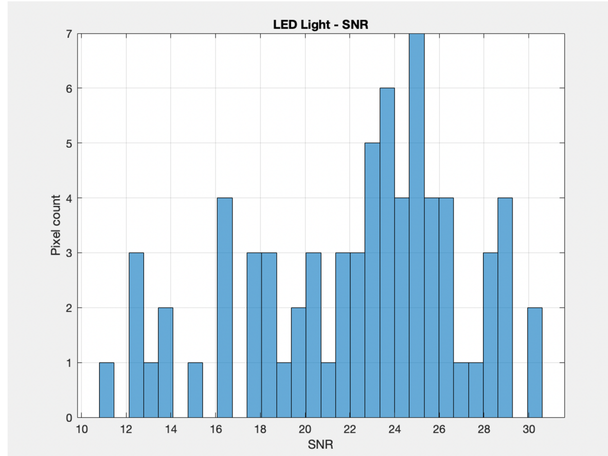


Pic16. Luminance channel two lighting conditions, hot spot seen in picture.



Pic17. Color patches extracted with two lighting conditions.





Pic18. Color Patch signal to noise ratios under two different illumination conditions.

Methods: ADMM Optimizer

Here, again, we are using ADMM optimizer to finding a better solution for the light source spectra.

XYZ from Radiant Colorimeter Output = (Radiant Camera Spectral Response _ MCC Reflectance Spectrum Matrix)* Light Source Spectrum

Here the Radiant Camera Spectral Response = CIE color matching functions

Similarly to the first part, we construct ADMM optimizer with the target and constraints below.

Our target x here is light source spectrum, constraint y is added for more realistic data simulation.

There are three constraints we are putting in y to optimize the solution:

(1) non-negative least squares (NNLS)

We will force y to drop negative values after x update. So the light source spectrum x will optimize towards positive values.

(2) spectral response smoothness

Since the light source spectrum curves usually do not have random spikes, the curves should look much smoother.

We would impose a moving average constraint, so within 15nm of wavelength window, if x is showing the trend of forming spikes, it will be optimized to take smoother value.

(3) minimize reconstruction error

The optimizing problem includes a term to compare the measured value and the estimated value, and to minimize the difference between both values in the update rule in the updates.

By constructing the ADMM optimizer with the constraints, 1500 iterations were computed and the light source spectra curves are being optimized in the updates.

Results:

ADMM optimizer results are shown in picture 19. We can see two curves, which represents two different LED lightings in the lab are having similar shape with blue LED peak showing realistic expectations. While there are two peaks in the green and red wavelength range, which may not represent the real LED light source spectrum. The results are reasonable for one to estimate light source spectrum, but will need some improvement and further investigation on the source of estimation error.

To look at how accurate the estimation compare to the measured data with different lights. We are using CIE deltaE formula to estimate the color differences between measured and the estimated data with the formula below:

$$\Delta E^*_{ab} = \sqrt{(L^*_2 - L^*_1)^2 + (a^*_2 - a^*_1)^2 + (b^*_2 - b^*_1)^2}.$$

The reference white point is selected as the white patch in the Macbeth checker board.

The calculation shows results below with Lab light:

deltaE of patches: Lab Lighting

4.0561	4.0944	3.4441	3.6884	2.1658	2.1772
1.9729	1.4473	1.5447	2.2468	0.8186	4.3955
2.6380	1.8856	4.3347	3.2313	1.6437	4.5077
3.1980	1.4375	1.2347	1.1634	1.7587	3.2371

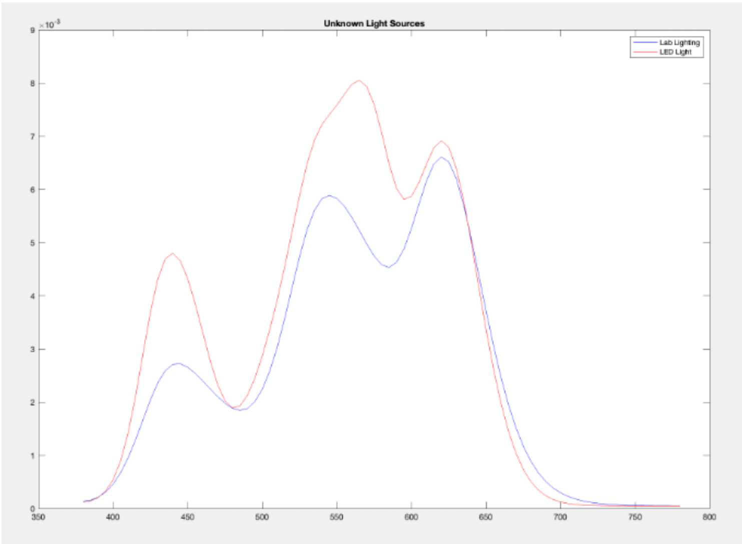
LED flashlight:

deltaE of patches: LED Lighting

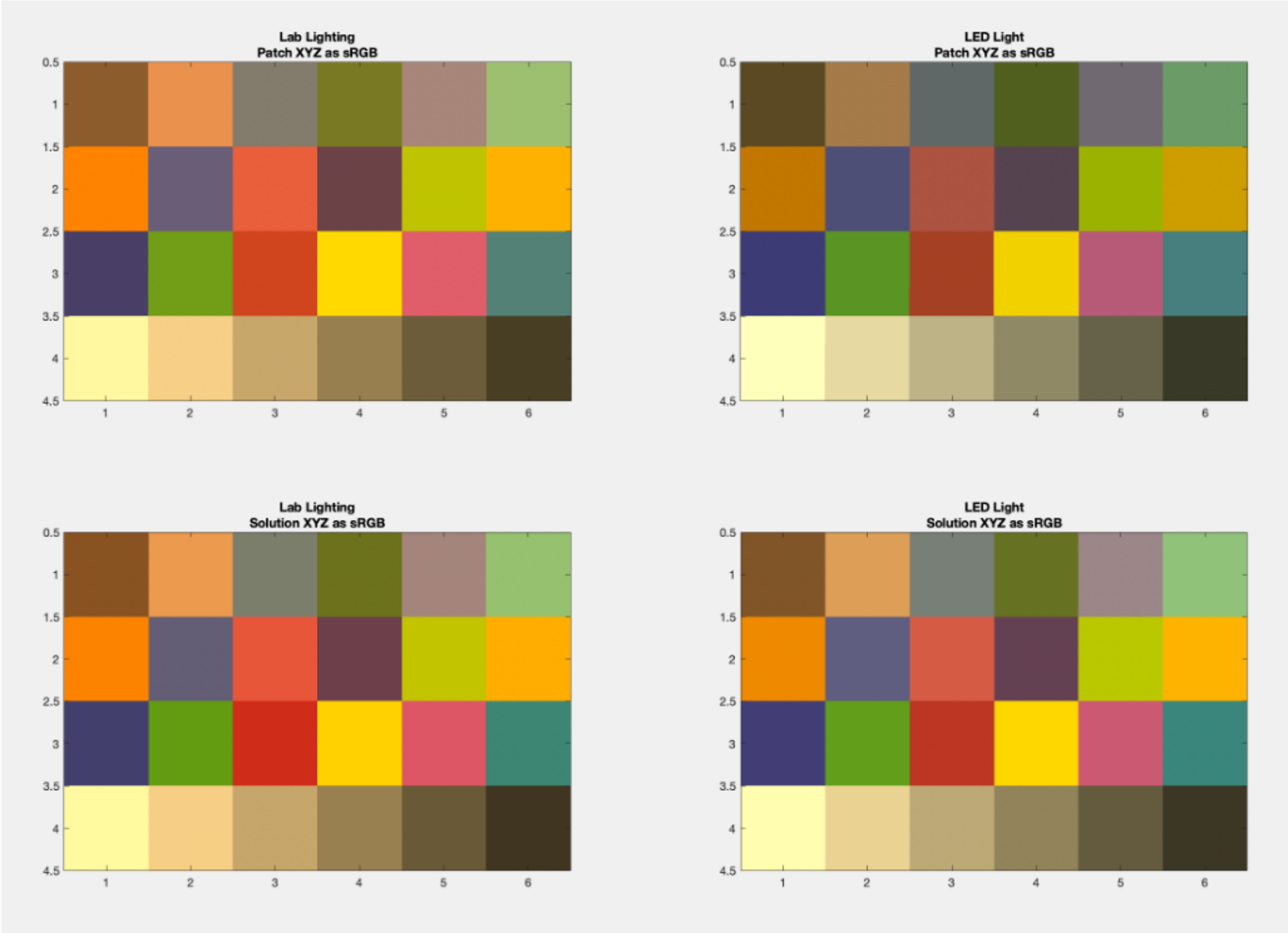
8.0693	3.8217	2.5805	2.4901	5.4132	3.7645
2.8710	4.2051	4.0896	6.9700	5.8756	4.5988
5.4590	7.7829	5.6502	9.4743	8.9730	9.4010
12.1954	13.6959	14.7583	13.9366	10.3524	6.5782

We can see that lab lighting shows smaller deltaE values at some of the rows, such as less than 2, which suggests a smaller color differences noticeable by most general population. While the LED flashlight case, there are some higher numbers suggests the color matching is not good enough. The reason can be the higher SNR in lab lighting compare to LED lighting. Also lab lighting from the ceiling is more uniform compare to a smaller LED flashlight illumination, which may cause some illumination non-uniformity.

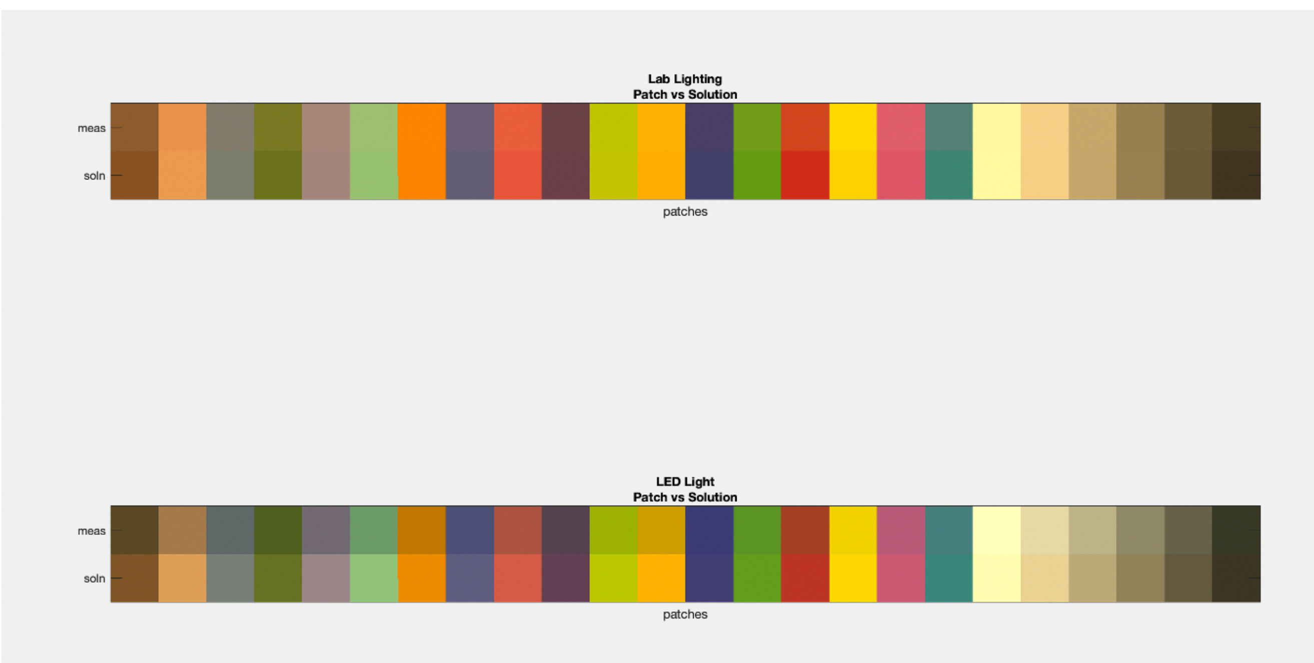
For visual comparison, the results are shown in picture 20 and 21. It looks like that lab lighting has better match between measured and simulated compare to LED flashlight case, which is consistent with the deltaE analysis.



Pic19. Predicted two light sources spectra.



Pic20. Color checker measurement vs. estimated solution.



Pic21. Color checker measurement vs. estimated solution side by side comparison.

Future Development Plan:

I would like to develop a better controlled lighting conditions and background in the lab for better color calibration and analysis. To confirm the part 2 light spectrum, a spectral-radiometer will be very helpful in getting the reference lighting data and compare with model to see where is the gap.

As an engineer working in AR / VR field, I would like to explore more on the color spec and calibration of AR / VR glasses for future product development.

Reference:

1 Tominaga, S.; Nishi, S.; Ohtera, R. Measurement and Estimation of Spectral Sensitivity Functions for Mobile Phone Cameras. *Sensors* **2021**, *21*, 4985. https://doi.org/10.3390/s21154985

2 https://angms.science/doc/NMF/nnls_admm.pdf

3 http://ohlab.kic.ac.jp/mobilespec_database.html

4 https://www.radiantvisionsystems.com/

Matlab Code:

Part I : 

PSYCH221 - Camera Color Calibration Project

PDF

Part II : 

PDF