Imagerie Couleur

Grading Method and

Final Report Instructions

Grading Method

- 3-Minute Presentation 5 points
- Final Report
  - Task 1: Camera Color Correction 10 points
  - Task 2: Short-term Repeatability 3 points
  - Task 3: Inter-instrument Agreement 2 points

Report Instructions

- You need to write a report containing the results of the Tasks that will be explained in the next slides. Task 1 is <u>mandatory</u>, Tasks 2 and 3 are recommended if you want to get a better grade.

- The report should be given as a PDF file, and you are free to choose your own layout for the document (you can do it in Word or Powerpoint if you want). There is no minimum nor maximum number of pages or slides.

- Please send your reports to:

  ricardo.sapaico@gmail.com

- The deadline is Friday, July 12th

Task 1: Camera Color Correction

Task 1: Camera Color Correction

- We want to calibrate the color of two cameras, so that the colors can be computationally corrected to match the "real" color[1].

- We have two different cameras:

1. A Canon 5D Mark iii — 22.3 MPix, Sensor size: 24 x 36 mm
2. An IDS UI1490LE — 10.5 MPix, Sensor size: 6.4 x 4.6 mm

- We use the ColorChecker chart to calibrate each camera.

To check how robust the camera is with different illuminations, we calibrate each camera under two illuminant conditions: D50 and A

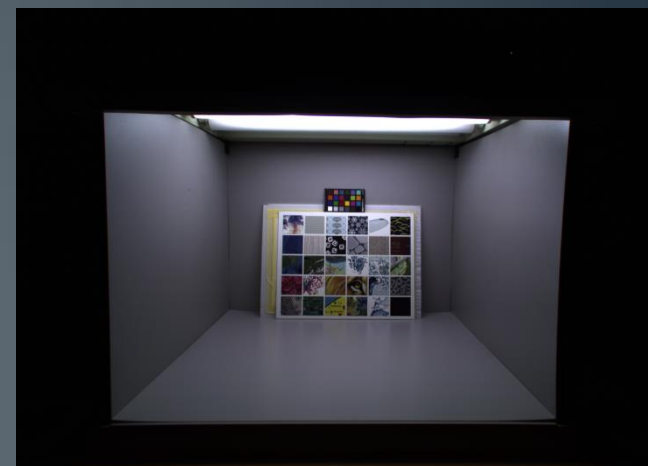
[1] In this case, "real" color = color measured with a spectrophotometer

Task 1: Camera Color Correction

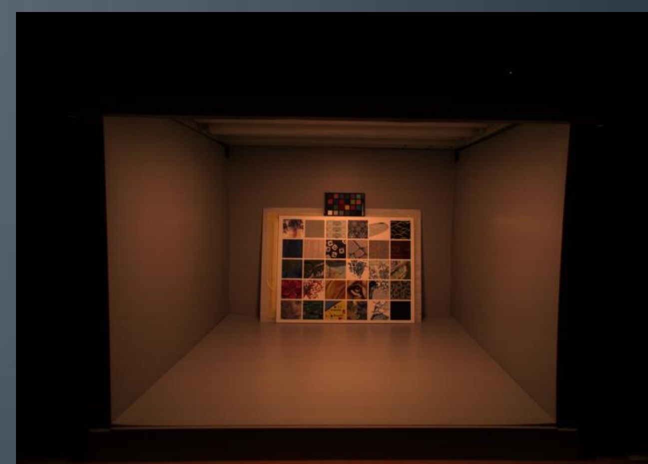
These are the images you will work with (in the '*Task_1*' folder):

Illuminant

D50



A

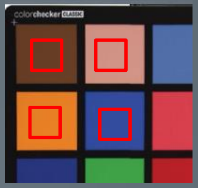


5D Mark iii

UI1490LE

Task 1: Camera Color Correction

What you need to do is, for each image:

1. Compute the average RGB value of each color of the ColorChecker chart (i.e., you will obtain 24 RGB values):
    - You can compute the average of a 16 x 16 square located in the center of each color. For instance: 
    - You can set the positions of the rectangles manually (<u>recommended</u>), or you can try to detect them automatically using the "colour – checker detection" package in Python: https://www.colour-science.org/colour-checker-detection/ However, note that this package is not robust, and it will not detect the chart, so you will need to improve its image processing algorithms.
    - Be sure that your RGB values are stored from left to right, and from top to bottom. In other words, follow the same order as the one given in the slides of the course (dark skin, light skin, blue sky,…, neutral 5, neutral 3.5, black)

Task 1: Camera Color Correction

2. Normalize your RGB values between 0 and 1, considering that the maximum value is 255.

3. Get the reference XYZ value of the 24 colors in the ColorChecker chart (FYI, the reference illuminant for the chart is D50):
   - You can import it directly using the Color Science library (recommended):

   CC = colour.COLOURCHECKERS[**'ColorChecker24 - After November 2014'**]

   XYZ_ref = colour.xyY_to_XYZ(list(CC.data.values()))

   - Or you can download the Lab data from X-Rite's page:

   https://xritephoto.com/documents/literature/EN/ColorChecker24_After_Nov2014.zip

   In this case, you will need to re-arrange the reference data, as it's ordered from top to bottom, and from left to right. Then, you need to convert from Lab to XYZ.

Now that you have the reference XYZ values and the camera RGB values, you can learn the correction matrix M

Task 1: Camera Color Correction

4. Compute the correction matrix M for the following methods:

- Linear Method
- Polynomial Method (2nd Degree)
- Root Polynomial Method (2nd Degree)

Pay attention you create the necessary data in matrix P, as explained during the course

As a reminder, for all cases, you will need to compute M by doing:

$$M = XP^T(PP^T)^{-1}$$

Pay attention that the dimensions of your matrices are good, according to the 'm' and 'N' parameters we saw in the second lecture

Now that you have computed the correction matrices M, you can correct the colors of the entire image

Task 1: Camera Color Correction

5. Correct the colors in the image, by applying the following transformation to all pixels in the image:

$$X = MP$$

In this case, the dimension of P will be: N x T, where T = # of pixels in the image

6. The above transformation gives you the corrected XYZ values in matrix X, so you need to convert all XYZ values back to RGB. You can use the following function:

```
rgb = colour.XYZ_to_RGB( XYZ_values, illuminant_reference, illuminant_target,
          colour.RGB_COLOURSPACES['sRGB'].XYZ_to_RGB_matrix)
```

- Pay attention that the XYZ matrix used in this conversion should have one XYZ value on each row.
- Your reference illuminant is D50 and your target is D65 (white point of sRGB space), and you should use the CIE 1931 Standard observer (check: colour.ILLUMINANTS)

Task 1: Camera Color Correction

7. RGB values you obtain will be in the [0,1] range, so you need to convert them to the [0,255] range. You will also need to clip negative values, and clip those values which are above 255.

Now you have obtained three corrected RGB images (one for each method), which you can save to disk. To check them I'd suggest using Photoshop or GIMP. If you use the built-in image viewer, the image colors might be totally wrong (in Windows at least).

8. The next –and final– step is to evaluate the accuracy of each method.

Task 1: Camera Color Correction

9. We repeat step 5. However, this time, instead of using all the RGB values from the image, we use only the 24 RGB values of the ColorChecker chart that you have found in Step 2.

Up to this point, you should have obtained the 24 corrected XYZ values of the ColorChecker chart.

10. To evaluate the accuracy of the color correction, we need to compute the delta-E 2000 between the reference Lab color, and the Lab color obtained after the correction. Therefore:
    - You need to convert the reference XYZ values from Step 3 to Lab values.
    - You also need to convert the corrected XYZ values from Step 9 to Lab values. You can use: `colour.XYZ_to_Lab(XYZ_values, illuminant)`
    - For the conversion, use the illuminant D65 as reference.

Task 1: Camera Color Correction

As a result, you should have obtained the 24 delta-E 2000 values of the ColorChecker chart, for each method.

11. Finally, for each method, compute the average delta-E, and the delta-E standard deviation.

12. Repeat Steps 1-11 for each of the 4 images.

Task 1: Camera Color Correction

In the report you must include the following information:

- All the corrected images (in total, 12 images). Feel free to resize them to avoid having a very heavy report.
- All sub-images of the corrected color checker (in total, 12 ColorChecker images)
- The 24 reference Lab values of the ColorChecker chart, and the 12 sets of 24 corrected Lab values (one per method/per image).
- The accuracy result (average delta-E and standard deviation) of each method, **<u>and your analysis and justifications</u>** regarding: (i) which method you think is more accurate; (ii) which method gives more pleasant results; and (iii) which camera is more accurate.

Task 2: Short-term Repeatability

Task 2: Short-term Repeatability

- We want to measure the short-term repeatability of a spectrophotometer (let's call it "Spectrophotometer A").

- To do it, we have measured 10 times the following 26 colors:

- The data of the measurements is given in the '*Task_2*' folder, where "t1" means 1^{st} measurement, t2 means 2^{nd} measurement, and so on.

- In each csv file, you will find:
  - In the first row the spectral range of the spectrophotometer in nanometers.
  - From the second row to the last, each row contains the spectral response (i.e., the Reflectance) of each color patch. P1 means patch 1, P2 means patch 2, and so on.

Task 2: Short-term Repeatability

What you need to do is:

1. Parse the data and for each color patch create a dictionary of wavelengths versus spectral value.
2. Convert the Reflectance to XYZ tristimulus value using the CIE 1931 2 Degree Observer and the D65 Illuminant as references.
3. Convert the XYZ value to Lab value.
4. Repeat for each of the 10 measurements (at the end you should have 260 Lab values: 10 Lab values for each color patch).
5. For each of the 26 color patches:
   - Compute the delta-E 2000 color difference between the first measurement and the rest (i.e., you should obtain 9 delta-Es).
   - Compute the average delta-E 2000 for the patch.
6. Finally, compute the average delta-E 2000 for all 26 patches.
7. Considering that the short-term repeatability tolerance for this instrument is $0.1\Delta E_{2000}$, is Spectrophotometer A in or out of specification?

Task 2: Short-term Repeatability

In the report you must include the following information :

- All Lab values of Patches P1, P2 and P3 (30 Lab values in total)
- The average delta-E between the first and the other measurements, for all 26 patches (26 delta-E values in total)
- Your response and explanation of whether Spectrophotometer A is in or out of specification regarding its short-term repeatability.

Task 2: Short-term Repeatability

In Python, you can use the Color Science Toolbox:

https://www.colour-science.org/

Some commands that might be useful:

```
import colour
colour.SpectralDistribution(spectral_data_dictionary, name='Sample')
colour.STANDARD_OBSERVERS_CMFS['CIE 1931 2 Degree Standard Observer']
colour.ILLUMINANTS_SDS['D65']
colour.sd_to_XYZ(spectral_distribution, observer, illuminant)
        [be sure you divide the output by 100 to normalize the XYZ value]
colour.XYZ_to_Lab(XYZ_value_normalized)
colour.difference.delta_E_CIE2000(Lab_1, Lab_2)
```

Task 3: Inter-instrument agreement

Task 3: Inter-instrument agreement

- Now, we bought a second spectrophotometer (let's call it "Spectrophotometer B"), and we want to know if this new instrument gives the same results as the previous one (Spectrophotometer A).

- But first, we need to measure the short-term repeatability of the new spectrophotometer.

- Therefore, you need to first repeat the same experiment you have done for Spectrophotometer A. The data of the measurements is given in the '*Task_3' folder*, where "t1" means 1st measurement, t2 means 2nd measurement, and so on.

- Then you can compare the measurements of both instruments.

Task 3: Inter-instrument agreement

What you need to do is:

1. Repeat Task 1 for Spectrophotometer B, and say whether the instrument is in or out of specification.
2. For each instrument, compute the average Lab value for each color patch (i.e., you will obtain 26 Lab values for each spectrophotometer)
3. For each color patch, compute the delta-E 2000 difference between Spectrophotometers A and B (i.e., you will obtain 26 delta-E 2000 values).
4. Finally, compute the average delta-E 2000 for all 26 patches, and the maximum delta-E 2000 among all colors.
5. Considering that the inter-instrument agreement tolerance between these instruments is: average = $0.5\Delta E_{2000}$ and maximum = $0.5\Delta E_{2000}$

   Are Spectrophotometers A and B in or out of specification?

Task 3: Inter-instrument agreement

In the report you must include the following information :

- The average delta-E between the first and the other measurements, for all 26 patches (26 delta-E values in total), for Spectrophotometer B.
- Your response and explanation of whether Spectrophotometer B is in or out of specification regarding its short-term repeatability.
- The 26 delta-E 2000 values between the measurements of Spectrophotometer A and Spectrophotometer B, corresponding to each color patch.
- Your response and explanation of whether the two Spectrophotometers are in or out of specification regarding their inter-instrument agreement, and whether we can use them both, or if we should only use one of them.