Vision and Opponency

Reading the Dataset

```
warning('off','all');
color_checker_dataset = readtable("HW_Opponency_Data.xlsx",Sheet="ColorChecker");
lms_dataset = readtable("HW_Opponency_Data.xlsx",Sheet="LMS");
light_sources_dataset = readtable("HW_Opponency_Data.xlsx",Sheet="Sources");
```

Question 2

a. Calculate LMS values for all 24 ColorChecker samples, for both incandescent and daylight illumination.

```
%Change in Wavelength
wavelength = 380:10:730;
d_lambda = mean(diff(wavelength));

% Create LMS relative sensitivity matrix
lms_matrix = transpose(lms_dataset{:,2:4});

% calculate illuminant's spectral power distribution = S
% normalization is done to light sources to it's max value
s_incandescent = (normalize(light_sources_dataset{:,2},'range'));
s_daylight = (normalize(light_sources_dataset{:,3},'range'));


% Objectspectral reflectance factor = R
[color_checker_numRows,color_checker_numCols] = size(color_checker_dataset);
r_matrix = color_checker_dataset{:,2:color_checker_numCols};

% LMS values for all 24 ColorChecker samples
LMS_stimuli_incandescent = LMS_stimuli(lms_matrix, ...
    diag(s_incandescent), r_matrix,d_lambda)
```

```
LMS_stimuli_incandescent = 3×24
    6.1601   21.2330    8.7393    6.7169   12.0632   18.3657   20.3242    5.1711 ···
    3.4530   12.3532    6.5834    4.9096    8.0757   15.2685   10.0784    3.9806
    0.4673    1.9159    2.3974    0.5247    3.0230    3.3093    0.4609    2.5368
```

```
LMS_stimuli_daylight = LMS_stimuli(lms_matrix, ...
    diag(s_daylight), r_matrix, d_lambda)
```

```
LMS_stimuli_daylight = 3×24
    7.9622   28.3691   14.5461   10.4358   18.3820   32.2160   23.7162    9.0313 ···
    6.0577   22.5268   14.4775    9.5055   17.1260   32.8450   15.1642    9.5332
    3.1505   12.7636   16.7560    3.3499   21.3916   21.2245    2.8945   18.3435
```

2b. Compute the CATs (**M**vonKries), assuming complete chromatic adaptation to Illuminant E (also known as equal-energy illuminant, a spectral power distribution with a value of 1 at all wavelengths), for incandescent and daylight illumination.

```
% Equi-energy light source is Illuminant E (L2)
```

```
s_illuminantE = ones(36,1);


% LMS_lightsource
LMS_source_daylight = LMS_source(lms_matrix,s_daylight,d_lambda);
LMS_source_incandescent = LMS_source(lms_matrix,s_incandescent,d_lambda);
LMS_source_illuminantE = LMS_source(lms_matrix,s_illuminantE,d_lambda);

% LMS_source_daylight = LMS_source(lms_matrix,...
% diag(s_daylight),d_lambda);
% LMS_source_incandescent = LMS_source(lms_matrix,...
% diag(s_incandescent),d_lambda);
% LMS_source_illuminantE = LMS_source(lms_matrix,...
% diag(s_illuminantE),d_lambda);


% Computing the CAT (Von Kries M Matrix)
vonkries_incandescent = vonkries(LMS_source_incandescent,...
    LMS_source_illuminantE)
```

```
vonkries_incandescent = 3×3
    2.1469         0         0
         0    2.5771         0
         0         0    8.3953
```

```
vonkries_daylight = vonkries(LMS_source_daylight,...
    LMS_source_illuminantE)
```

```
vonkries_daylight = 3×3
    1.4316         0         0
         0    1.3059         0
         0         0    1.2243
```

```
% LMS values of stimuli after applying von kries matrix
LMS_stimuli_incandescent_vk = vonkries_incandescent * LMS_stimuli_incandescent;
LMS_stimuli_daylight_vk = vonkries_daylight * LMS_stimuli_daylight;
```

2c. Calculate opponent signals for all 24 ColorChecker samples, for both incandescent and daylight illumination using the CATs you computed. Set gamma to 2.4.

```
m_opponency = [0.64 0.39 -0.01; 1.12 -1.50 0.34; 0.35 0.15 -0.53];

gamma = 2.4;

opponency_incandescent = calcOpponency(m_opponency,...
    LMS_stimuli_incandescent_vk,gamma)
```

```
opponency_incandescent = 3×24
    2.8288    4.7605    3.4051    3.0500    3.8264    4.7202    4.5839    2.7379 ···
    0.1559    0.2391    0.1075   -0.2827    0.3405   -0.3923    0.1711    0.3125
    0.4626    0.6669   -0.1760    0.5128   -0.1498    0.1933    1.3392   -0.5457
```

```
opponency_daylight = calcOpponency(m_opponency,LMS_stimuli_daylight_vk,gamma)
```

```
opponency_daylight = 3×24
    2.6699    4.5598    3.5598    3.0706    3.8846    4.9869    4.1163    2.9379 ···
    0.1330    0.1730    0.0605   -0.2165    0.2251   -0.3337    0.2363    0.2074
    0.3897    0.5857   -0.1156    0.5541   -0.1514    0.3860    1.1429   -0.4922
```

Question 3

Look at the opponent signals for the ColorChecker neutral patches (19-24) as well as yellow (16). Do the values make sense, compared to what opponent signals would you expect for neutral samples? Does the CAT do what it's supposed to?

```
cmp_table_daylight = array2table([opponency_daylight(:,16), opponency_daylight(:,19:24
cmp_table_daylight.Properties.VariableNames = ["Pat16","Pat19","Pat20","Pat21","Pat22"
cmp_table_daylight
```

cmp_table_daylight = 3×7 table

	Pat16	Pat19	Pat20	Pat21	Pat22	Pat23	Pat24
1	5.6100	6.7731	5.6814	4.5980	3.5947	2.5885	1.6999
2	-0.3863	0.0561	0.0574	0.0528	0.0338	0.0243	0.0220
3	1.6810	0.5309	0.3951	0.3066	0.2425	0.1716	0.1061

```
cmp_table_incandescent = array2table([opponency_incandescent(:,16), opponency_incandes
cmp_table_incandescent.Properties.VariableNames = ["Pat16","Pat19","Pat20","Pat21","Pa
cmp_table_incandescent
```

cmp_table_incandescent = 3×7 table

	Pat16	Pat19	Pat20	Pat21	Pat22	Pat23	Pat24
1	5.8728	6.7924	5.6815	4.5962	3.5891	2.5832	1.6979
2	-0.4656	0.0658	0.0568	0.0491	0.0280	0.0204	0.0229
3	1.6794	0.5222	0.3892	0.3043	0.2381	0.1684	0.1059

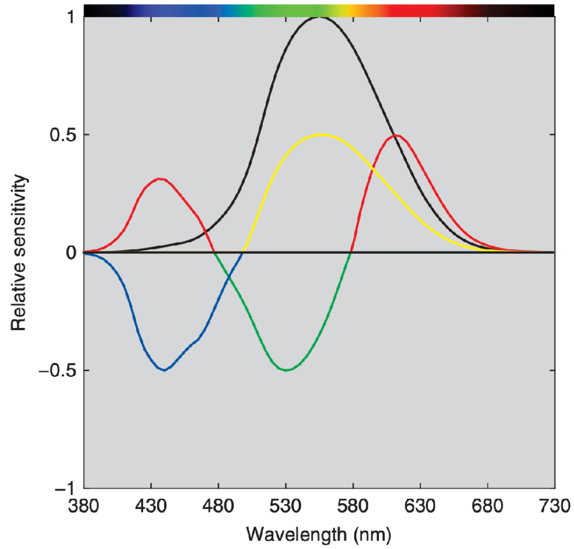

Figure 2.11 Relative spectral sensitivities of the human visual system's opponent channels.

The above graph represents "Relative spectral sensitives of the human visual system's opponent channels." The black curve, red-green curve, and yellow-blue curve represent the white opposed by black channel, red opposed by green channel, and yellow opposed by blue channel, respectively. L and M cones combine, forming the white opposed by black signal with nominal pulse rate between 0(complete black) to 1(pure weight) relative sensitivity.

The result of the opponency signals for the given neutral patches (19-24) as well as yellow (patch 16) from the ColorChecker values make sense. The neutral patches in the ColorChecker showed a decreasing order from left to right as well as opponency signal results for both incandescent and day light sources as the colors become darker.

To address whether the CAT did what it is supposed to do, I calculated the opponency with LMS before CATs.

```
opponency_incandescent_bef = calcOpponency(m_opponency,...
    LMS_stimuli_incandescent,gamma);
opponency_daylight_bef = calcOpponency(m_opponency,LMS_stimuli_daylight,gamma);

cmp_table_daylight_bef = array2table([opponency_daylight_bef(:,16), opponency_daylight_
cmp_table_daylight_bef.Properties.VariableNames = ["Pat16","Pat19","Pat20","Pat21","Pa
cmp_table_daylight_bef
```

cmp_table_daylight_bef = 3×7 table

	Pat16	Pat19	Pat20	Pat21	Pat22	Pat23	Pat24
1	4.8955	5.9118	4.9590	4.0134	3.1376	2.2594	1.4838
2	-0.5450	-0.1635	-0.1269	-0.0969	-0.0824	-0.0593	-0.0335
3	1.4102	0.3261	0.2274	0.1720	0.1370	0.0960	0.0569

```
cmp_table_incandescent_bef = array2table([opponency_incandescent_bef(:,16), opponency_
cmp_table_incandescent_bef.Properties.VariableNames = ["Pat16","Pat19","Pat20","Pat21"
cmp_table_incandescent_bef
```

cmp_table_incandescent_bef = 3×7 table

	Pat16	Pat19	Pat20	Pat21	Pat22	Pat23	Pat24
1	4.1687	4.8266	4.0371	3.2660	2.5503	1.8355	1.2065
2	-0.1599	-0.0204	-0.0243	-0.0193	-0.0221	-0.0162	-0.0050
3	1.5605	1.2197	1.0004	0.8049	0.6286	0.4512	0.2947

Comparing the opponent values from two tables cmp_table_incandescent_bef vs cmp_table_daylight_bef (before CAT) and cmp_table_daylight vs cmp_table_daylight_bef (after CAT), the opponency value are consistent in both the illuminant only after applying CAT (which is the expected outcome). And also, the absolute opponency value computed after CAT correlates with the relative sensitivity graph of the human vision opponency channel graph shown above. So the CAT did what it is supposed to do.

Question 1

Write Matlab code (with no loops!) to compute LMS values as in the sidebar on p20, and to compute opponent values by implementing Eqs. 2.27 and 2.28. The computation for matrix **M**vonKries is defined on p22. **M**opponency is defined on p25;

```
function T_stimuli = LMS_stimuli(t,s,r,d)
    T_stimuli = (t*s*r*d);
end
```

$$\mathbf{t} = \mathbf{TSR} \tag{2.7}$$

$$\mathbf{T} = \begin{pmatrix} l_\lambda & \dots & l_\lambda \\ m_\lambda & \dots & m_\lambda \\ s_\lambda & \dots & s_\lambda \end{pmatrix} \tag{2.8}$$

$$\mathbf{S} = \begin{pmatrix} S_\lambda & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & S_\lambda \end{pmatrix} \tag{2.9}$$

$$\mathbf{R} = \begin{pmatrix} R_{\lambda,1} & R_{\lambda,2} \\ \vdots & \vdots \\ R_{\lambda,1} & R_{\lambda,2} \end{pmatrix} \tag{2.10}$$

$$\mathbf{t} = \begin{pmatrix} L_1 & L_2 \\ M_1 & M_2 \\ S_1 & S_2 \end{pmatrix} \tag{2.11}$$

```
% Function to calculate Chromatic Adaptation Transformation - Von Kries
function M_factor = vonkries(LMS1,LMS2)
```

```
    M_factor = diag(LMS2./LMS1);
end
```

$$L_2 = L_1 \frac{L_{n,2}}{L_{n,1}} \tag{2.19}$$

$$M_2 = M_1 \frac{M_{n,2}}{M_{n,1}} \tag{2.20}$$

$$S_2 = S_1 \frac{S_{n,2}}{S_{n,1}} \tag{2.21}$$

$$\begin{pmatrix} L_2 \\ M_2 \\ S_2 \end{pmatrix} = \begin{pmatrix} \frac{L_{n,2}}{L_{n,1}} & 0 & 0 \\ 0 & \frac{M_{n,2}}{M_{n,1}} & 0 \\ 0 & 0 & \frac{S_{n,2}}{S_{n,1}} \end{pmatrix} \begin{pmatrix} L_1 \\ M_1 \\ S_1 \end{pmatrix} \tag{2.22}$$

```
% Function to calculate Opponency
function opp = calcOpponency(m_opponency, LMS_vk, gamma)
    opp = m_opponency * ((LMS_vk).^(1/gamma));
end
```

$$\begin{pmatrix} L \\ M \\ S \end{pmatrix}_{\text{reference illuminant}} = \mathbf{M}_{\text{von Kries } D \text{ factor}} \, \mathbf{TSR} \tag{2.27}$$

$$\begin{pmatrix} \text{white} \iff \text{black} \\ \text{red} \iff \text{green} \\ \text{yellow} \iff \text{blue} \end{pmatrix} = \mathbf{M}_{\text{opponency}} \begin{pmatrix} L^{1/\gamma} \\ M^{1/\gamma} \\ S^{1/\gamma} \end{pmatrix}_{\text{reference illuminant}} \tag{2.28}$$

```
% Function to calculate LMS of the light source
function T_source = LMS_source(t,s,d)
    T_source = (t*s*d);
end
```