Table of Contents


```
%Max Shooster and Malcolm Zale
%Team 4
```

```
close all;
```

Step 1: Returns a structure of CIE observer and illuminant data

```
cie = loadCIEdata;


function [cie] = loadCIEdata()
%LOADCIEDATA Loads the CIE data.
%Detailed explanation goes here
    file = fopen('CIE_2Deg_380-780-5nm.txt');
    Vals = textscan(file, '%d %f %f %f');
    cie = {};
    cie.lambda = double(Vals{1});
    cie.cmf2deg = [Vals{2},Vals{3},Vals{4}];
    fclose(file);

    file = fopen('CIE_10Deg_380-780-5nm.txt');
    Vals = textscan(file, '%d %f %f %f');
    cie.cmf10deg = [Vals{2},Vals{3},Vals{4}];
    fclose(file);

    file = fopen('CIE_IllA_380-780-5nm.txt');
    Vals = textscan(file, '%d %f');
    cie.illA = Vals{2};
    fclose(file);

    file = fopen('CIE_IllC_380-780-5nm.txt');
    Vals = textscan(file, '%d %f');
    cie.illC = Vals{2};
    fclose(file);
```

```
    file = fopen('CIE_IllD50_380-780-5nm.txt');
    Vals = textscan(file, '%d %f');
    cie.illD50 = Vals{2};
    fclose(file);

    file = fopen('CIE_IllD65_380-780-5nm.txt');
    Vals = textscan(file, '%d %f');
    cie.illD65 = Vals{2};
    fclose(file);

    cie.illE = zeros(81,1) + 100;

    file = fopen('CIE_IllF_1-12_380-780-5nm.txt');
    Vals = textscan(file, '%d %f %f %f %f %f %f %f %f %f %f %f %f');
    Data = [];
    for x = 2:13
        Data = [Data, Vals{x}];
    end
    cie.illF = Data;

    cie.PRD = zeros(81,1) + 1.0;
end
%%
%include a listing of the indicated function in a published report
%<include>loadCIEdataMkII.m</include
```

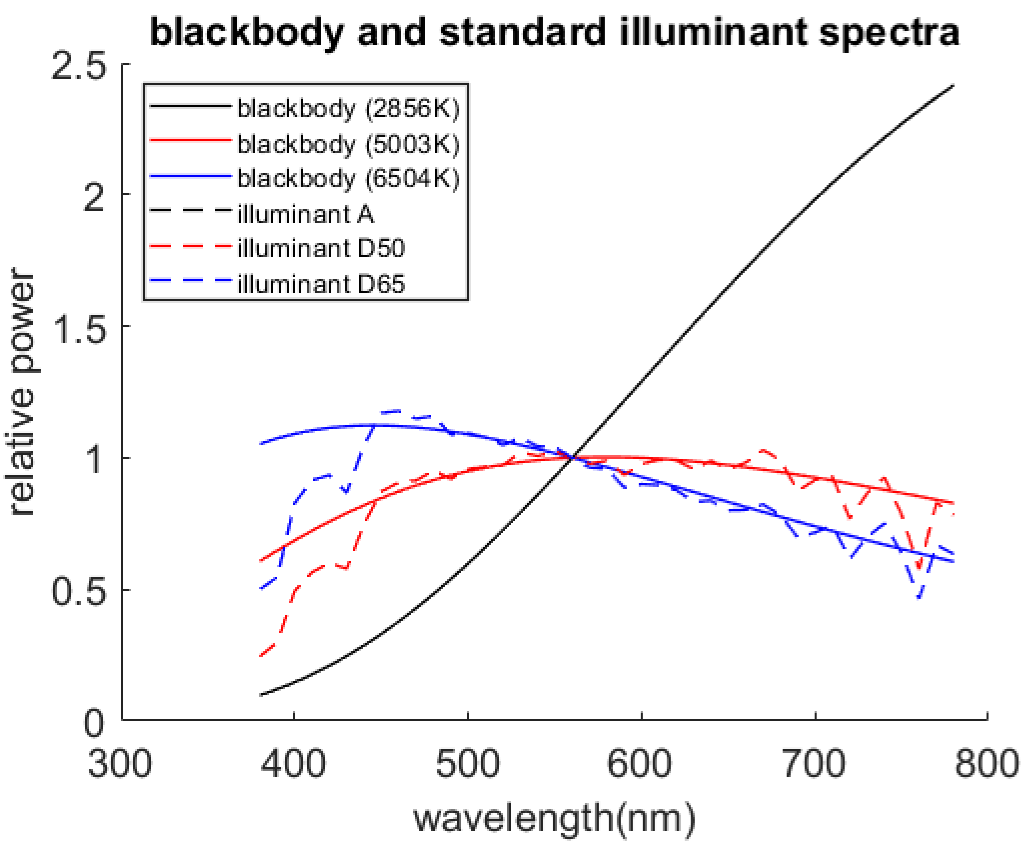
```
%Normalize the data to 1.0
cie.illA = cie.illA/100;
cie.illD50 = cie.illD50/100;
cie.illD65 = cie.illD65/100;
```

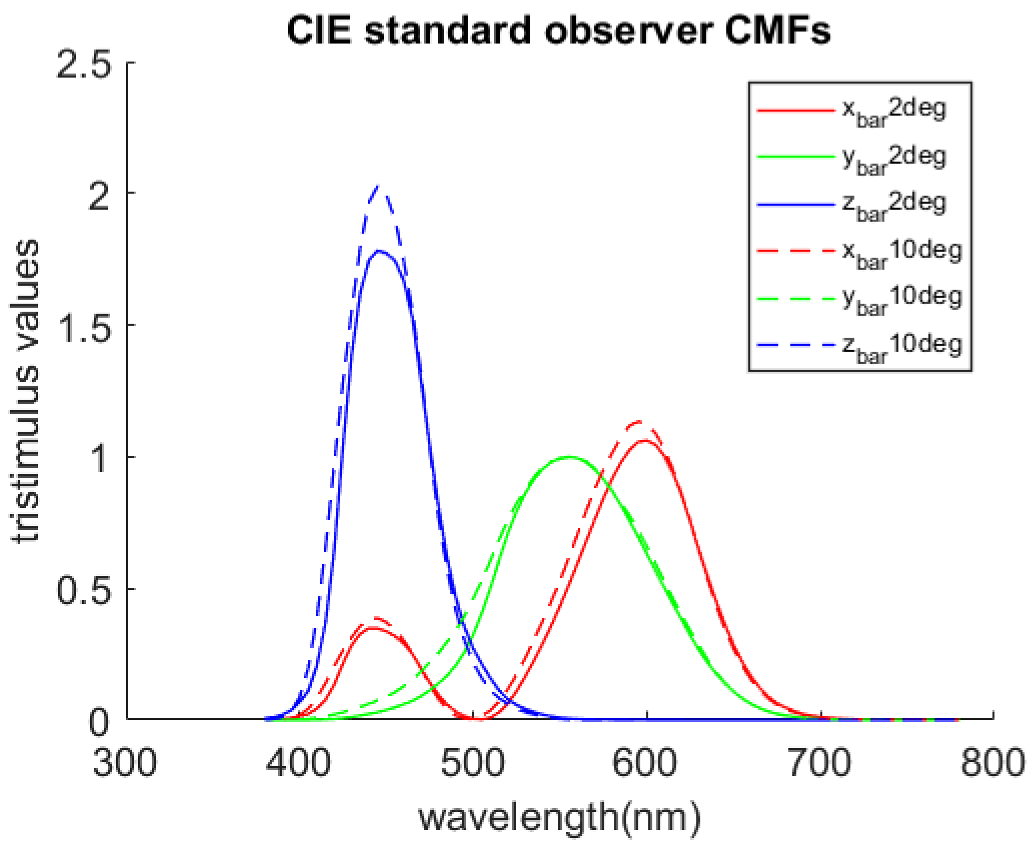
Step 3

Plot the blackbody spectra agains the illuminants

```
figure;
hold on;
% Adjust the font size of the graph
set(gca,'FontSize',15)
plot(cie.lambda, blackbody(2856, cie.lambda), '-k', 'LineWidth', 1)
plot(cie.lambda, blackbody(5003, cie.lambda), '-r', 'LineWidth', 1)
plot(cie.lambda, blackbody(6504, cie.lambda), '-b', 'LineWidth', 1)

plot(cie.lambda, cie.illA, '--k', 'LineWidth', 1)
plot(cie.lambda, cie.illD50, '--r', 'LineWidth', 1)
plot(cie.lambda, cie.illD65, '--b', 'LineWidth', 1)

xlabel('wavelength(nm)', 'FontSize', 15)
ylabel('relative power', 'FontSize', 15)
title('blackbody and standard illuminant spectra', 'FontSize', 15)
```

```
lgd = legend('blackbody (2856K)', 'blackbody (5003K)', 'blackbody
 (6504K)', ...
    'illuminant A', 'illuminant D50', 'illuminant D65', 'FontSize',
 10);
lgd.Location = 'northwest';
lgd.LineWidth = 1;

%Plot the CMF graphs
figure;
hold on;
% Adjust the font size of the graph
set(gca,'FontSize',15)
ylim([0, 2.5]);
plot(cie.lambda, cie.cmf2deg(:,1), '-r', 'LineWidth', 1);
plot(cie.lambda, cie.cmf2deg(:,2), '-g', 'LineWidth', 1');
plot(cie.lambda, cie.cmf2deg(:,3), '-b', 'LineWidth', 1);
plot(cie.lambda, cie.cmf10deg(:,1), '--r', 'LineWidth', 1);
plot(cie.lambda, cie.cmf10deg(:,2), '--g', 'LineWidth', 1);
plot(cie.lambda, cie.cmf10deg(:,3), '--b', 'LineWidth', 1);
lgd =
 legend('x_{bar}2deg','y_{bar}2deg','z_{bar}2deg','x_{bar}10deg',...
    'y_{bar}10deg','z_{bar}10deg', 'FontSize', 10);
lgd.LineWidth = 1;
xlabel('wavelength(nm)', 'FontSize', 15)
ylabel('tristimulus values', 'FontSize', 15)
title('CIE standard observer CMFs', 'FontSize', 15)
```





Steps 4-5

```
function XYZ = ref2XYZ(refs, cmfs, illum)
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
    k = 100/(sum(cmfs(:,2).*illum));
    X = k*sum(cmfs(:,1).*illum.*refs);
    Y = k*sum(cmfs(:,2).*illum.*refs);
    Z = k*sum(cmfs(:,3).*illum.*refs);
    XYZ = [X,Y,Z];
end
```

```
%Equations for calculating XYZ tristimulus Values
CC_spectra = importdata('ColorChecker_380_780_5nm.txt');
for patch_num = 2:25
    CC_XYZs(:,patch_num-1) =
 ref2XYZ(CC_spectra(:,patch_num),cie.cmf2deg,cie.illD65);
end
CC_XYZs
```

CC_XYZs =

```
  Columns 1 through 7

   11.5145   39.1346   18.3488   11.1492   25.8437   31.7110   37.1457
   10.3819   36.5981   19.6332   13.8551   24.3868   43.8600   29.5592
    7.1502   27.0564   35.6470    7.4267   45.6142   44.8778    6.5006

  Columns 8 through 14

   13.8627   29.1328    8.5889   33.9174   46.1864    8.9183   15.0353
   12.3179   19.8475    6.4569   44.1533   42.4957    6.4177   24.1079
   39.3093   14.9941   15.4745   11.4297    8.6771   32.2736    9.6379

  Columns 15 through 21

   19.3447   55.8457   29.6768   14.4138   87.8402   57.9621   35.2286
   11.3576   58.9726   19.3515   19.9750   92.3781   61.0426   37.0414
    5.5526    9.6411   32.2626   39.0008   95.6125   65.4909   40.2256

  Columns 22 through 24

   19.3492    8.7646    3.2111
   20.4708    9.2915    3.3763
   22.1545   10.3188    3.9312
```

Step 6-7

```
function xyY = XYZ2xyY(XYZ)
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
x = XYZ(1,:) ./ (XYZ(1,:)+XYZ(2,:)+XYZ(3,:));
y = XYZ(2,:) ./ (XYZ(1,:)+XYZ(2,:)+XYZ(3,:));
xyY = [x;y;XYZ(2,:)];
end
```

```
%Calculates x, y chromaticity coordinates
CC_xyYs = XYZ2xyY(CC_XYZs);
CC_xyYs
```

```
CC_xyYs =

  Columns 1 through 7

    0.3964    0.3807    0.2492    0.3438    0.2696    0.2633    0.5074
    0.3574    0.3561    0.2667    0.4272    0.2544    0.3641    0.4038
   10.3819   36.5981   19.6332   13.8551   24.3868   43.8600   29.5592

  Columns 8 through 14

    0.2117    0.4554    0.2814    0.3790    0.4744    0.1873    0.3082
```

```
   0.1881    0.3102    0.2116    0.4933    0.4365    0.1348    0.4942
  12.3179   19.8475    6.4569   44.1533   42.4957    6.4177   24.1079

 Columns 15 through 21

   0.5336    0.4487    0.3651    0.1964    0.3185    0.3142    0.3132
   0.3133    0.4738    0.2381    0.2722    0.3349    0.3309    0.3293
  11.3576   58.9726   19.3515   19.9750   92.3781   61.0426   37.0414

 Columns 22 through 24

   0.3122    0.3089    0.3053
   0.3303    0.3275    0.3210
  20.4708    9.2915    3.3763
```

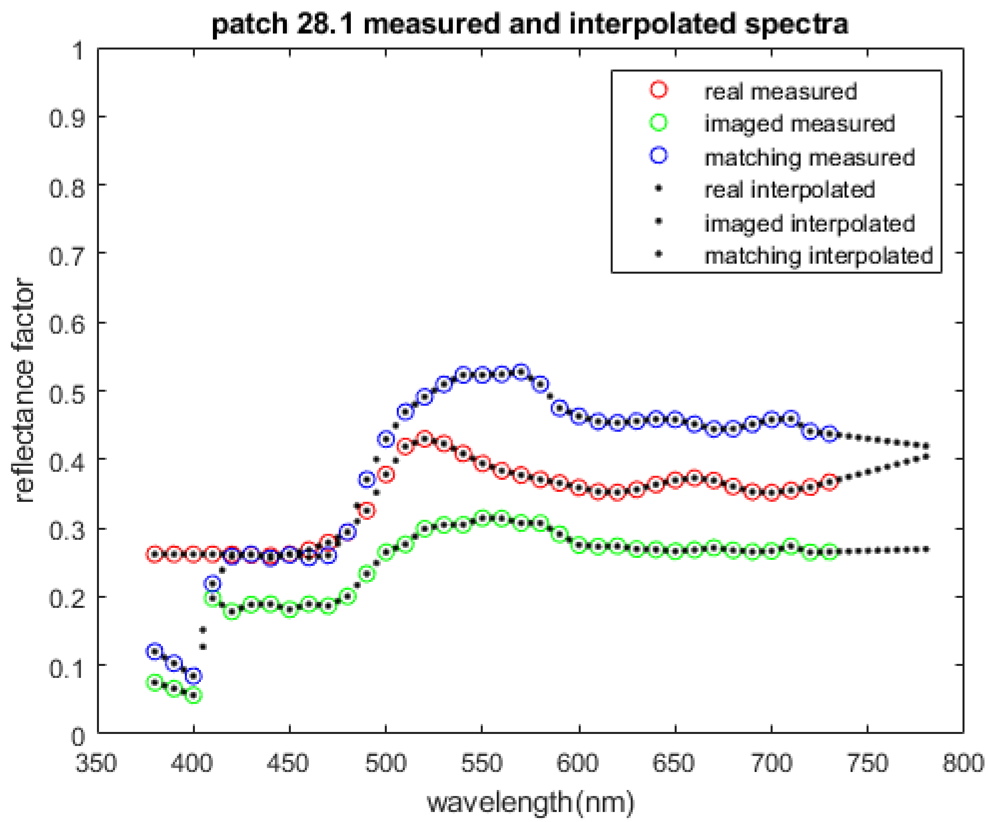
step 8

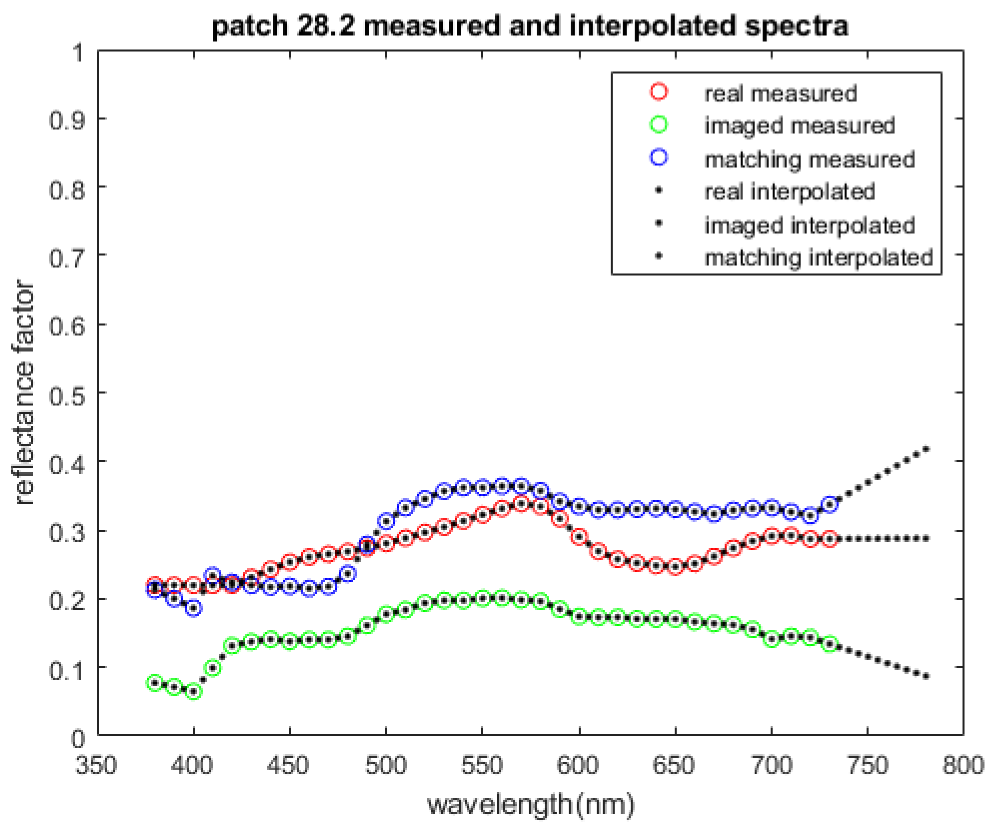
```
%now we are going to import the color munki data
%defines color munki spotread wavelength incriments
cm_lams = (380:10:730);

%defines header offset for reading the .sp files
cm_h_offset = 19;

%load and normalize the measured spectral data from patch 1
data= importdata('28.1_real.sp', ' ', cm_h_offset);
real_281 = data.data/100;

data = importdata('28.1_imaged.sp', ' ', cm_h_offset);
imaged_281 = data.data/100;

data = importdata('28.1_matching.sp', ' ', cm_h_offset);
matching_281 = data.data/100;

%load and normalize the measured spectral data from patch 2
data= importdata('28.2_real.sp', ' ', cm_h_offset);
real_282 = data.data/100;

data = importdata('28.2_imaged.sp', ' ', cm_h_offset);
imaged_282 = data.data/100;

data = importdata('28.2_matching.sp', ' ', cm_h_offset);
matching_282 = data.data/100;

%interpolates data
real_281i = interp1(cm_lams, real_281,
 cie.lambda(:), 'linear', 'extrap');
imaged_281i = interp1(cm_lams, imaged_281,
 cie.lambda(:), 'linear', 'extrap');
matching_281i = interp1(cm_lams, matching_281,
 cie.lambda(:), 'linear', 'extrap');
```

```
real_282i = interp1(cm_lams, real_282,
 cie.lambda(:), 'linear', 'extrap');
imaged_282i = interp1(cm_lams, imaged_282,
 cie.lambda(:), 'linear', 'extrap');
matching_282i = interp1(cm_lams, matching_282,
 cie.lambda(:), 'linear', 'extrap');
```

step 9

```
%now we're going to plot the interpolated and measured spectral data

figure
plot(cm_lams, real_281, 'or', cm_lams, imaged_281, 'og', cm_lams,
 matching_281, 'ob')
hold on
plot(cie.lambda, real_281i, '.k', cie.lambda, imaged_281i, '.k',
 cie.lambda, matching_281i, '.k')
ylim ([0 1])
legend('real measured', 'imaged measured', 'matching measured', 'real
 interpolated', 'imaged interpolated', 'matching interpolated')
xlabel('wavelength(nm)')
ylabel('reflectance factor')
title('patch 28.1 measured and interpolated spectra')

figure
plot(cm_lams, real_282, 'or', cm_lams, imaged_282, 'og', cm_lams,
 matching_282, 'ob')
hold on
plot(cie.lambda, real_282i, '.k', cie.lambda, imaged_282i, '.k',
 cie.lambda, matching_282i, '.k')
ylim ([0 1])
legend('real measured', 'imaged measured', 'matching measured', 'real
 interpolated', 'imaged interpolated', 'matching interpolated')
xlabel('wavelength(nm)')
ylabel('reflectance factor')
title('patch 28.2 measured and interpolated spectra')
```


patch 28.1 measured and interpolated spectra
real measured
imaged measured
matching measured
real interpolated
imaged interpolated
matching interpolated
reflectance factor
wavelength(nm)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
350
400
450
500
550
600
650
700
750
800


patch 28.2 measured and interpolated spectra
real measured
imaged measured
matching measured
real interpolated
imaged interpolated
matching interpolated
reflectance factor
wavelength(nm)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
350
400
450
500
550
600
650
700
750
800

step 10

```
%import measured XYZ values
real_281m = importdata('28_XYZ_Labs_Real.txt');
measured.real_281m = real_281m.data(1, 2:4);

real_282m = importdata('28_XYZ_Labs_Real.txt');
measured.real_282m = real_282m.data(2, 2:4);

imaged_281m = importdata('28_XYZ_Labs_imaged.txt');
measured.imaged_281m = imaged_281m.data(1, 2:4);

imaged_282m= importdata('28_XYZ_Labs_imaged.txt');
measured.imaged_282m = imaged_282m.data(2, 2:4);

matching_281m= importdata('28_CM_XYZ_Labs_matching.txt');
measured.matching_281m= matching_281m.data(1, 2:4);

matching_282m= importdata('28_CM_XYZ_Labs_matching.txt');
measured.matching_282m= matching_282m.data(2, 2:4);


%convert to XYZ for patch 28.1
calc.real_281iXYZ = ref2XYZ(real_281i, cie.cmf2deg, cie.illD50);
calc.imaged_281iXYZ= ref2XYZ(imaged_281i, cie.cmf2deg, cie.illD50);
calc.matching_281iXYZ = ref2XYZ(matching_281i, cie.cmf2deg,
 cie.illD50);

%convert to XYZ for patch 28.2
calc.real_282iXYZ = ref2XYZ(real_282i, cie.cmf2deg, cie.illD50);
calc.imaged_282iXYZ = ref2XYZ(imaged_282i, cie.cmf2deg, cie.illD50);
calc.matching_282iXYZ = ref2XYZ(matching_282i, cie.cmf2deg,
 cie.illD50);

%prints the tables
fprintf('\t\t\t                     patch 28.1\n');
fprintf('\t\t           measured\t\t                 calculated\n');
fprintf('\t        X         Y         Z         X         Y
 Z\n');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'real', measured.real_281m', calc.real_281iXYZ');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'imaged', measured.imaged_281m', calc.imaged_281iXYZ');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'matching', measured.matching_281m', calc.matching_281iXYZ');

fprintf('\n\n\t\t\t                     patch 28.2\n');
fprintf('\t\t           measured\t\t                 calculated\n');
fprintf('\t        X         Y         Z         X         Y
 Z\n');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'real', measured.real_282m', calc.real_282iXYZ');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
```

```
    'imaged', measured.imaged_282m', calc.imaged_282iXYZ');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'matching', measured.matching_282m', calc.matching_282iXYZ');
```

```
                        patch 28.1
           measured                        calculated
        X         Y         Z         X         Y         Z
    real 34.213207 38.210821 23.096516 34.220571 38.200633 23.121053
  imaged 26.402367 29.166966 16.124004 26.400970 29.155587 16.134726
matching 43.517236 48.654233 23.067332 43.516594 48.631933 23.090876


                        patch 28.2
           measured                        calculated
        X         Y         Z         X         Y         Z
    real 27.822804 30.467375 21.064402 27.817479 30.457476 21.066850
  imaged 17.065465 18.826895 11.789639 17.062978 18.820423 11.789882
matching 31.461865 34.425328 18.971788 31.462863 34.413566 18.994412
```

step 11

```
%converts measured XYZ to xyY
bin.real_281m = XYZ2xyY(transpose(measured.real_281m));
bin.real_282m = XYZ2xyY(transpose(measured.real_282m));
bin.imaged_281m = XYZ2xyY(transpose(measured.imaged_281m));
bin.imaged_282m = XYZ2xyY(transpose(measured.imaged_282m));
bin.matching_281m = XYZ2xyY(transpose(measured.matching_281m));
bin.matching_282m = XYZ2xyY(transpose(measured.matching_282m));

%converts interpolated XYZ to xyY
bin.real_281i = XYZ2xyY(transpose(calc.real_281iXYZ));
bin.real_282i = XYZ2xyY(transpose(calc.real_282iXYZ));
bin.imaged_281i = XYZ2xyY(transpose(calc.imaged_281iXYZ));
bin.imaged_282i = XYZ2xyY(transpose(calc.imaged_282iXYZ));
bin.matching_281i = XYZ2xyY(transpose(calc.matching_281iXYZ));
bin.matching_282i = XYZ2xyY(transpose(calc.matching_282iXYZ));


%printing the tables
fprintf('\t\t\t                    patch 28.1\n');
fprintf('\t\t           measured\t\t                    calculated\n');
fprintf('\t         X         Y         Z         X         Y
 Z\n');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'real', bin.real_281m', bin.real_281i');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'imaged', bin.imaged_281m', bin.imaged_281i');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'matching', bin.matching_281m', bin.matching_281i');

fprintf('\n\n\t\t\t                    patch 28.2\n');
fprintf('\t\t           measured\t\t                    calculated\n');
```

```
fprintf('\t          X          Y          Z          X          Y
 Z\n');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'real', bin.real_282m', bin.real_282i');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'imaged', bin.imaged_282m', bin.imaged_282i');
fprintf('%8s% 10f% 10f% 10f% 10f% 10f% 10f\n', ...
    'matching', bin.matching_282m', bin.matching_282i');
```

patch 28.1

	measured			calculated		
	X	Y	Z	X	Y	Z
real	0.358176	0.400027	38.210821	0.358172	0.399830	38.200633
imaged	0.368268	0.406830	29.166966	0.368259	0.406682	29.155587
matching	0.377627	0.422204	48.654233	0.377619	0.422008	48.631933

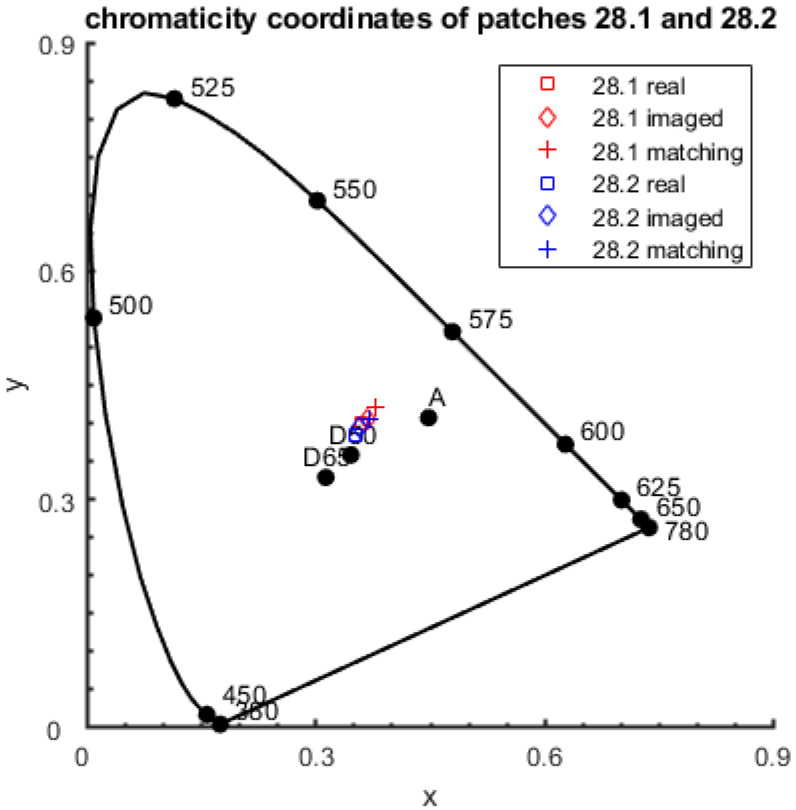
patch 28.2

	measured			calculated		
	X	Y	Z	X	Y	Z
real	0.350614	0.383940	30.467375	0.350603	0.383877	30.457476
imaged	0.357902	0.394843	18.826895	0.357915	0.394779	18.820423
matching	0.370755	0.405677	34.425328	0.370715	0.405482	34.413566

step 12

this is the given code to plot the shell of the chromaticity diagram

```
plot_chrom_diag_skel;
hold on

%graphs the chromaticity coordinates of the patch data
set(gca,'FontSize',10)
t1= plot(bin.real_281i(1), bin.real_281i(2), 'rs');
t2= plot(bin.imaged_281i(1), bin.imaged_281i(2), 'rd');
t3= plot(bin.matching_281i(1), bin.matching_281i(2), 'r+');
t4= plot(bin.real_282i(1), bin.real_282i(2), 'bs');
t5= plot(bin.imaged_282i(1), bin.imaged_282i(2), 'bd');
t6= plot(bin.matching_282i(1), bin.matching_282i(2), 'b+');

lgd2 = legend(([t1, t2, t3, t4, t5, t6]), '28.1 real', '28.1
 imaged', '28.1 matching', '28.2 real', '28.2 imaged', '28.2
 matching');
lgd2.Location = 'northeast';
lgd2.LineWidth = (0.5);
title('chromaticity coordinates of patches 28.1 and 28.2');
```



Feedback

```
%Max did steps 1 through 7, Malcolm did steps 8 through 13.
% I (Max) struggled with the initial set-up, particularly regarding
 ensureing
% that all of the graphs matched the expected output. Both in terms of
 content
% as well as style. I (Max) felt that this was an invaluable
 experience that
% will be very useful for both future lab work and general future work
 in
% Imaging Science.

%I (malcolm) struggled a lot with the print statements but was able to
%figure it out thanks to help from friends and Max's code from the
 previos
%lab where print statements werre needed.
% I thought it was valuable to learn how to build and use functions in
% MATLAB
%I have no notes on how to improve this lab.
```

Published with MATLAB® R2019b