Example:

Inputs for the code:

A. Image of the Color Checker Chart

B. Reference RGB values of all 24 color patches of Color Checker Chart

Reference R	Reference G	Reference B
115	82	68
194	150	130
98	122	157
87	108	67
133	128	177
103	189	170
214	126	44
80	91	166
193	90	99
94	60	108
157	188	64
224	163	46
56	61	150
70	148	73
175	54	60
231	199	31
187	86	149
8	133	161
243	243	242
200	200	200
160	160	160
122	122	121
85	85	85
52	52	52

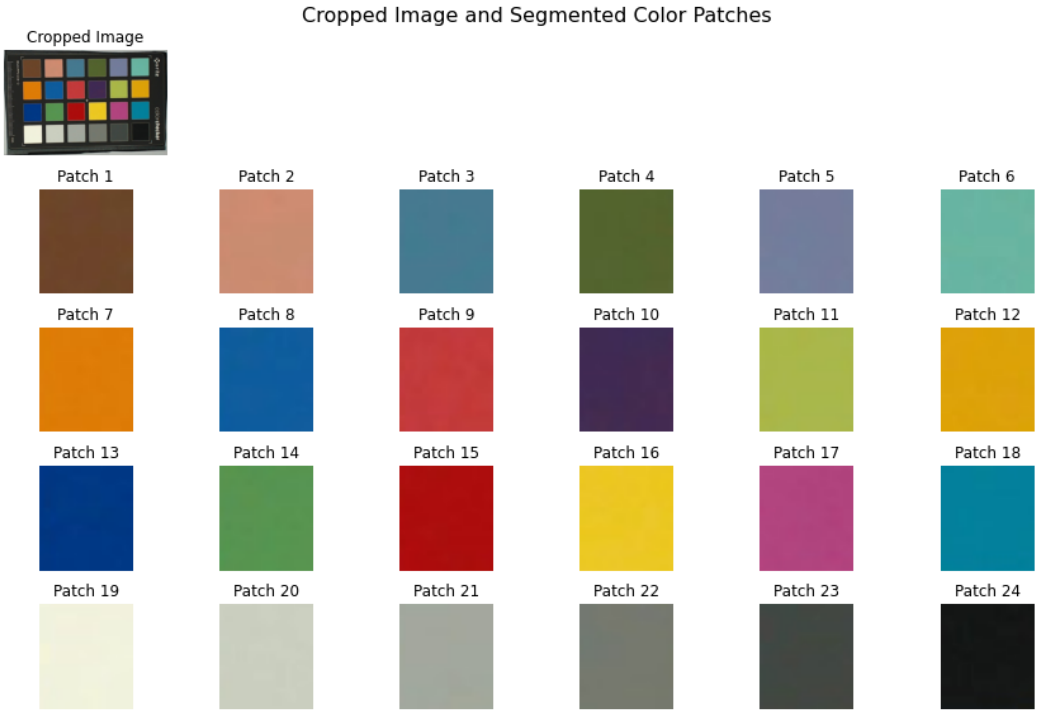
Output after running the code:

A. Save the corrected RGB values in excel file in the 'data/output/test_corrected_rgb_values.xlsx'

Corrected R	Corrected G	Corrected B
115	81	63
200	141	123
86	125	155
96	106	66
120	127	168
112	181	174
219	127	45
51	100	170
192	75	75
83	65	94
166	184	87
219	163	46
43	73	143
99	150	93
169	51	48
235	200	59
174	81	137
44	131	168
241	241	237
199	207	206
160	168	170
122	124	121
83	84	81
54	55	51

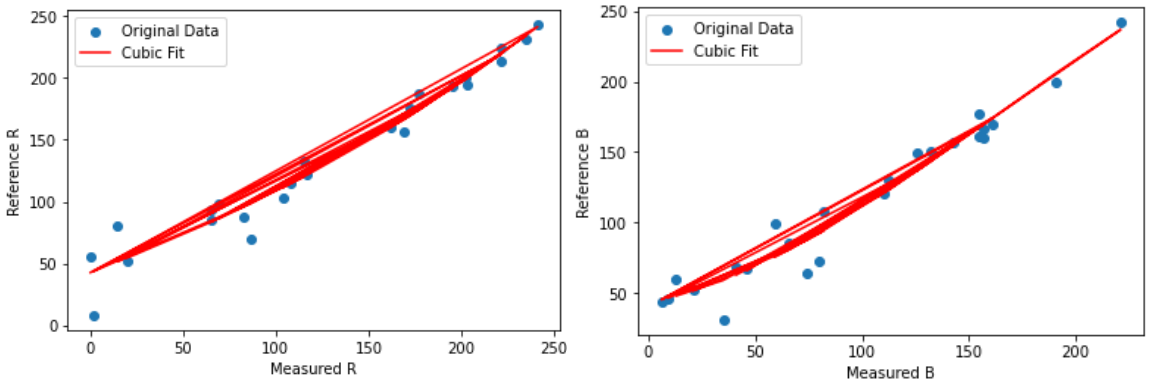
B. 1. Display of the cropped image and segmented patches

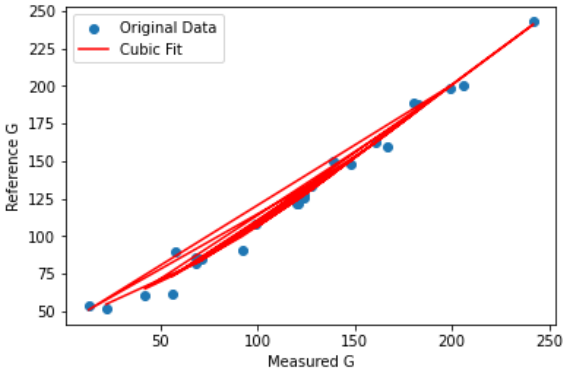
(1. While executing code in the Bash, the generated images need to be closed to maintain code flow. These images allow the user to verify whether the code is correctly extracting the colors from the patches. Once confirmed, the user should close the image. 2. In Python, these images are generated in single step and displayed in the console. Theres in no need to close the images manually.)



B. 2. Display of fit plot for the best model

(1. While executing the code in Bash, the generated graphs/ plots must be closed to maintain the code flow. These plots allow the user to verify whether the modeling is accurate. Once confirmed, the user should close the plots. 2. In Python, these plots are generated in single-step and displayed in the console. Theres in no need to close the plots manually.)





Table

Results of the code execution displayed in the bash interface.

(The measured RGB values, performance metrics of models, generated best-Fit equations, corrected RGB values shown in data frame)

```
santo@DESKTOP-B58AT7B MINGW64 ~/RGB_extraction_correction_code (main)
$ python
~/RGB_extraction_correction_code/src/single_step_chart_segmentation_rgb_extraction_and_
correction.py

RGB values extracted and saved to data/output/test_measured_rgb_values.xlsx

Debug: Measured RGB Values (from file):
Patch 1: R=108, G=68, B=41
Patch 2: R=203, G=139, B=112
Patch 3: R=69, G=121, B=143
Patch 4: R=83, G=99, B=46
Patch 5: R=115, G=124, B=155
Patch 6: R=104, G=180, B=161
Patch 7: R=221, G=124, B=6
Patch 8: R=15, G=92, B=157
Patch 9: R=195, G=58, B=59
Patch 10: R=65, G=42, B=82
Patch 11: R=169, G=183, B=74
Patch 12: R=221, G=161, B=9
Patch 13: R=0, G=56, B=132
Patch 14: R=87, G=148, B=80
Patch 15: R=172, G=13, B=13
Patch 16: R=235, G=199, B=35
Patch 17: R=177, G=68, B=126
Patch 18: R=2, G=128, B=155
Patch 19: R=241, G=242, B=221
```

```
Patch 20: R=202, G=206, B=191
Patch 21: R=162, G=167, B=157
Patch 22: R=117, G=120, B=110
Patch 23: R=65, G=71, B=66
Patch 24: R=20, G=22, B=21

R - Linear model: R²=0.9492, RMSE=14.3553
R - Quadratic model: R²=0.9580, RMSE=13.0571
R - Cubic model: R²=0.9581, RMSE=13.0474

Best model for R: R_corrected = 1.5979e-06 * R^3 + 6.3165e-04 * R^2 + 5.8058e-01 * R +
4.2519e+01
G - Linear model: R²=0.9774, RMSE=7.7110
G - Quadratic model: R²=0.9858, RMSE=6.1128
G - Cubic model: R²=0.9870, RMSE=5.8628

Best model for G: G_corrected = -6.7820e-06 * G^3 + 3.7672e-03 * G^2 + 2.9026e-01 * G +
4.6446e+01
B - Linear model: R²=0.9459, RMSE=13.0278
B - Quadratic model: R²=0.9555, RMSE=11.8195
B - Cubic model: R²=0.9571, RMSE=11.6042

Best model for B: B_corrected = -9.8484e-06 * B^3 + 4.8126e-03 * B^2 + 2.9351e-01 * B +
4.3284e+01

Generated Best-Fit Equations:
R: R_corrected = 1.5979e-06 * R^3 + 6.3165e-04 * R^2 + 5.8058e-01 * R + 4.2519e+01
G: G_corrected = -6.7820e-06 * G^3 + 3.7672e-03 * G^2 + 2.9026e-01 * G + 4.6446e+01
B: B_corrected = -9.8484e-06 * B^3 + 4.8126e-03 * B^2 + 2.9351e-01 * B + 4.3284e+01

Corrected Values DataFrame:
     Corrected R  Corrected G  Corrected B
0            115           81           63
1            200          141          123
2             86          125          155
3             96          106           66
4            120          127          168
5            112          181          174
6            219          127           45
7             51          100          170
8            192           75           75
9             83           65           94
10           166          184           87
11           219          163           46
12            43           73          143
13            99          150           93
14           169           51           48
15           235          200           59
16           174           81          137
17            44          131          168
```

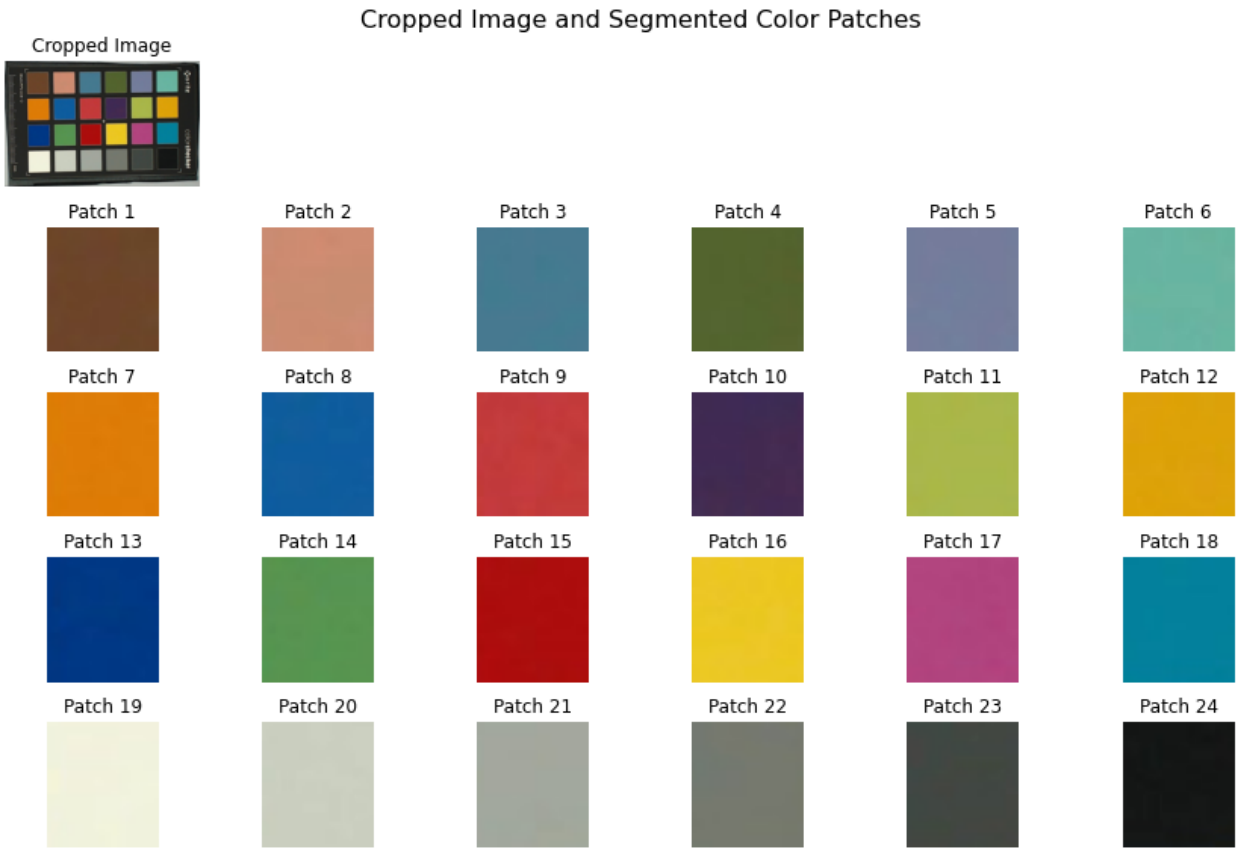
```
18          241          241          237
19          199          207          206
20          160          168          170
21          122          124          121
22           83           84           81
23           54           55           51

Corrected RGB values saved to data/output\test_corrected_rgb_values.xlsx
Total execution time: 8.21 seconds
```

Table

Results of the code execution displayed in the Python

(The measured RGB values, performance metrics of models, generated best-Fit equations, corrected RGB values shown in data frame)

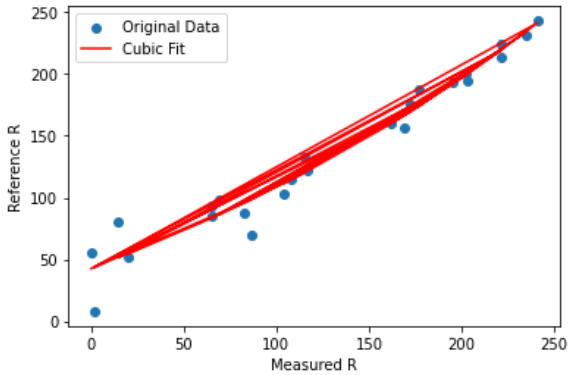
runfile('D:/PhD/Research topic/Thesis writing/Software impacts/extra/code for github repository.py', wdir='D:/PhD/Research topic/Thesis writing/Software impacts/extra')
RGB values extracted and saved to D:\PhD\Research topic\Thesis writing\Software impacts\RGB_extraction_correction_code\data\output\test_measured_rgb_values.xlsx



Debug: Measured RGB Values (from file):

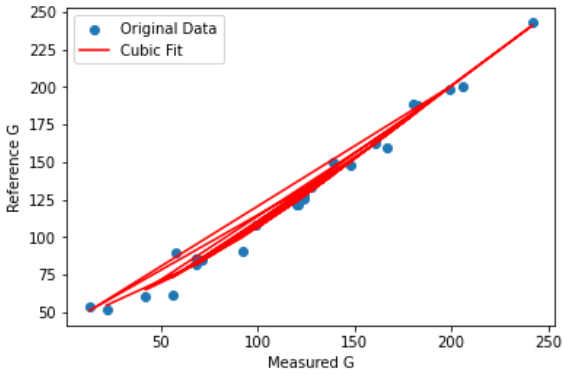
Patch 1: R=108, G=68, B=41
Patch 2: R=203, G=139, B=112
Patch 3: R=69, G=121, B=143
Patch 4: R=83, G=99, B=46
Patch 5: R=115, G=124, B=155
Patch 6: R=104, G=180, B=161
Patch 7: R=221, G=124, B=6
Patch 8: R=15, G=92, B=157
Patch 9: R=195, G=58, B=59
Patch 10: R=65, G=42, B=82
Patch 11: R=169, G=183, B=74
Patch 12: R=221, G=161, B=9
Patch 13: R=0, G=56, B=132
Patch 14: R=87, G=148, B=80
Patch 15: R=172, G=13, B=13
Patch 16: R=235, G=199, B=35
Patch 17: R=177, G=68, B=126
Patch 18: R=2, G=128, B=155
Patch 19: R=241, G=242, B=221
Patch 20: R=202, G=206, B=191
Patch 21: R=162, G=167, B=157
Patch 22: R=117, G=120, B=110
Patch 23: R=65, G=71, B=66
Patch 24: R=20, G=22, B=21
R - Linear model: R^2=0.9492, RMSE=14.3553
R - Quadratic model: R^2=0.9580, RMSE=13.0571
R - Cubic model: R^2=0.9581, RMSE=13.0474

Best model for R: R_corrected = 1.5979e-06 * R^3 + 6.3165e-04 * R^2 + 5.8058e-01 * R + 4.2519e+01



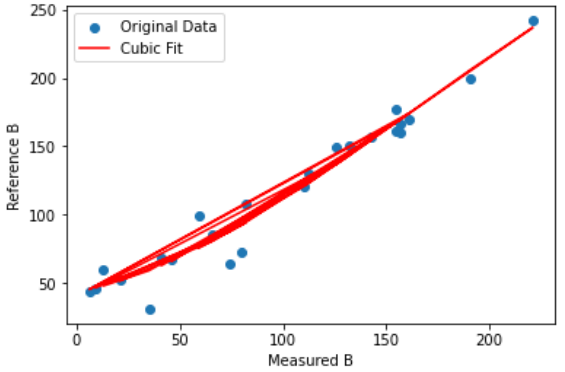
G - Linear model: R^2=0.9774, RMSE=7.7110
G - Quadratic model: R^2=0.9858, RMSE=6.1128
G - Cubic model: R^2=0.9870, RMSE=5.8628

Best model for G: G_corrected = -6.7820e-06 * G^3 + 3.7672e-03 * G^2 + 2.9026e-01 * G + 4.6446e+01



B - Linear model: R^2=0.9459, RMSE=13.0278
B - Quadratic model: R^2=0.9555, RMSE=11.8195
B - Cubic model: R^2=0.9571, RMSE=11.6042

Best model for B: B_corrected = -9.8484e-06 * B^3 + 4.8126e-03 * B^2 + 2.9351e-01 * B + 4.3284e+01



Generated Best-Fit Equations:
R: R_corrected = 1.5979e-06 * R^3 + 6.3165e-04 * R^2 + 5.8058e-01 * R + 4.2519e+01
G: G_corrected = -6.7820e-06 * G^3 + 3.7672e-03 * G^2 + 2.9026e-01 * G + 4.6446e+01
B: B_corrected = -9.8484e-06 * B^3 + 4.8126e-03 * B^2 + 2.9351e-01 * B + 4.3284e+01

Corrected Values DataFrame:

	Corrected R	Corrected G	Corrected B
0	115	81	63

1	200	141	123
2	86	125	155
3	96	106	66
4	120	127	168
5	112	181	174
6	219	127	45
7	51	100	170
8	192	75	75
9	83	65	94
10	166	184	87
11	219	163	46
12	43	73	143
13	99	150	93
14	169	51	48
15	235	200	59
16	174	81	137
17	44	131	168
18	241	241	237
19	199	207	206
20	160	168	170
21	122	124	121
22	83	84	81
23	54	55	51

Corrected RGB values saved to D:\PhD\Research topic\Thesis writing\Software impacts\RGB_extraction_correction_code\data\output\test_corrected_rgb_values.xlsx
Total execution time: 1.69 seconds