

## TN0021 - GENERATION OF A COLOR CORRECTION MATRIX

---

'Camelot.DLL' uses its internal default Color Correction Matrix, CCM, settings – unless otherwise specified in a 'CC.ini' file located in the same folder as the Camelot application.

A CCM can be generated using a simple least squares algorithm integrated into 'Camelot.DLL' with GUI provided by 'CamelotView'. Otherwise, some 3<sup>rd</sup> party SW such as **IMATEST**© must be used. See the description of 'CC.ini' below, which explains how the 12 CCM elements output by such a 3<sup>rd</sup> part SW are used in the framework of 'Camelot' applications.

### 1. Generation of CCM Using CamelotView (v2.1.1615 and up)

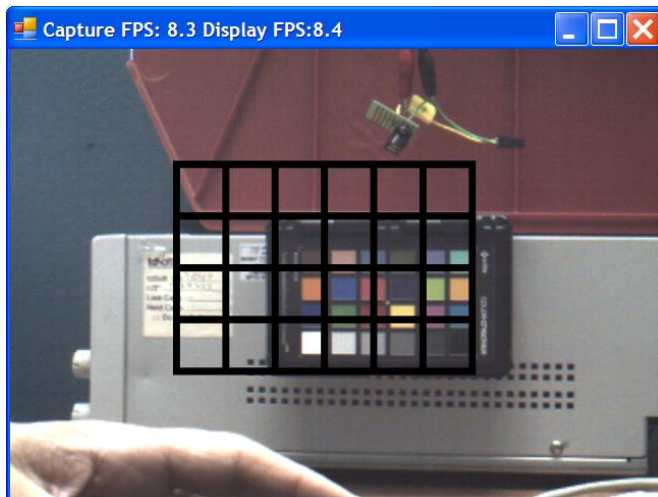
The X-Rite 24 x 6 ColorChecker should be placed in parallel to the camera lens and appear in the image aligned vertically and horizontally. Please note that the orientation of the ColorChecker should be such that the brown cell should be at the top left and the black one at the bottom right.

1. Add a 'recalc=1' entry to the [CCParam] section of 'CC.ini' located in the same folder as 'CamelotView.exe'.
2. Run video in as high a resolution that gives a reasonable frame rate.
3. Click SHIFT+"Processing/Color Matrix"; this starts the embedding of a 24 x 6 grid in the image. Since the grid is embedded in the image, the window may be stretched as desired.
4. Set focus (keyboard) to any window other than that of 'CamelotView' – as the keyboard arrows will be used to manipulate the grid – and not the UI.
5. Using the Up/Down/Left/Right arrows in conjunction with SHIFT, CTRL & ALT keys to align grid with image of 24 x 6 ColorChecker in frame.
  - ALT+Arrows expands or contracts the grid while preserving aspect ratio.
  - Arrows move the grid appropriately in a coarse manner.
  - Shift+Arrows move the grid in a fine manner.
  - CTRL+Left/Right decrease/increase the width of the grid in coarse steps.
  - CTRL+Up/Down increase/decrease the height of the grid in coarse steps.
  - SHIFT+CTRL+Left/Right decrease/increase the width of the grid in fine steps.
  - SHIFT+CTRL+Up/Down increase/decrease the height of the grid in fine steps.

Take care not to allow the grid to overshoot the frame's boundaries...

6. The grid is sufficiently well aligned when the interior of each of the 24 cells – minus a margin the double-thickness of the grid lines – contains pixels the cell's color only - and none of the grid's black.
7. Press both of the SHIFT keys down together to proceed with the calculation of the CCM. The 12 elements of the CCM will be calculated and output to the 'CCM' entry of the [CCParam] section of 'CC.ini'.
8. Note: if 'CC.ini' is not present, default values will be used.
9. Remove the 'recalc' entry of 'CC.ini' to prevent accidental entry into the CCM generation mode.

Please note: CCM's generated are only optimal for the lighting conditions similar to those in which the image of the ColorChecker was captured (image brightness due to CamelotView settings and ambient lighting conditions). A CCM can be adjusted to a certain extent to other lighting conditions by changing the 'ccm\_bri' entry of 'CC.ini' appropriately.



## 2. Color Correction Matrix (CCM)

The CCM is a 4 x 3 matrix which is applied to each pixel RGB triplet to provide a corrected pixel. Since the CCM is the result of least squares optimization it cannot provide an exact correction to the image's color distortion. The farther the distorted colors are from the nominal values (of the 6 x 4 ColorChecker standard) and the closer the distorted colors are to each other – the less accurate the results provided by the CCM...

On conclusion of generation of the CCM, it is written to the CCM entry of 'cc.ini' and a file is generated that provides details regarding the generation of the CCM - 'cch.txt'.

Here is a description of the data in 'cch.txt' (view without "Word Wrap"):

- **O**(bserved): 24 RGB triplets of the ColorCheckers 24 colors as averaged from the pixels of the 24 fields in the image captured by the camera.
- **T**(heoretical): 24 RGB triplets of the ColorCheckers 24 nominal colors (top to bottom, left to right) – scaled according to relative brightness of observed / theoretical.
- **CC**: 4 x 4 Matrix with which to solve least squares optimization problem.
- **A**: the calculated CCM
- **Goal**: same as **T**.
- **Achieved**: application of CCM to observed RGB triplets.
- **RMS** and maximal error are calculated from the difference between the goal and the achieved.

Experience shows that the best results are obtained by averaging the corrected pixel with the original uncorrected one according to suitable weights. In addition, it has been observed that best results were obtained when the DC correction was ignored - **dc\_numer=0** below.

A 'cc.ini' file specifies parameters used in the application of the CCM. A typical 'cc.ini' appears below. It describes in full the mathematical operation applied to each pixel with the CCM generated by CamelotView – or with 3<sup>rd</sup> party SW.

### 3. Screenshot of a typical CC.ini file

```
[CCParam]          ;Post-Bayer Color Correction Matrix (CCM)
worig=1            ;weight for original pixel RGB
wcorr=1           ;weight for pixel after application of CCM
                  ;final_pixel = (original_pixel * worig + corrected_pixel * wcorr) / (worig + wcorr)

dc_numer=0        ;weighting of the DC components in the CCM - which we usually ignore...
dc_denom=1

;CCM              correction              R G B DC original
;
;
; corr_pixel =   / \      / \      / \      / \      / \
;               | R |      bri    | c0 c1 c2 d0 | | r |      |d0|   DC_NUMER   |c3|
;               | G | = ----- * | c4 c5 c6 d1 | | g |      |d1| = ----- * |c7|
;               | B |      ccm_bri | c8 c9 cA d2 | | b |      |d2|   DC_DENOM   |cB|
;               \ /      \ /      \ /      \ /      \ /
;
; bri           - is the average brightness of the image to which the CCM is applied
; ccm_bri       - is the average brightness of the image from which the CCM was applied
;
; c0,          c1,          c2,          c3,          c4,          c5,          c6,          c7,          c8,          c9,          cA,          cB
CCM=1.806057, -0.786011, -5.182156, 11.972606, -0.267757, 1.224061, -0.244633, 17.292366, 0.096119, -1.282767, 1.850983, 18.116403
ccm_bri=100
; if the CCM entry is missing or empty - no color correction matrix is applied
```