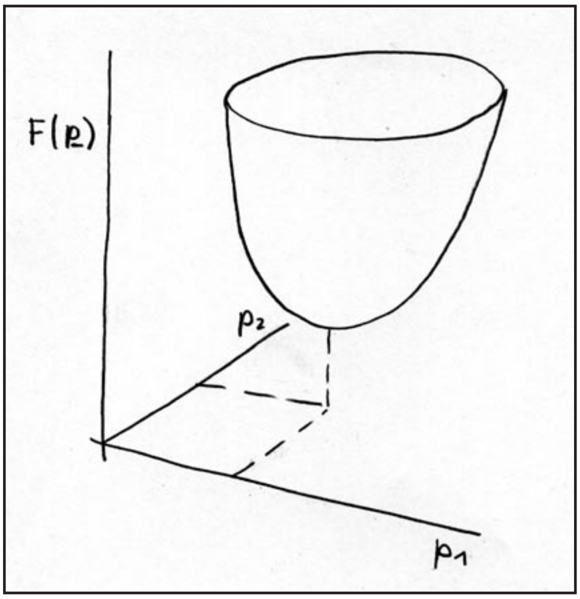
Gernot Hoffmann

Least Squares Algorithm
for Overdetermined Linear Equations



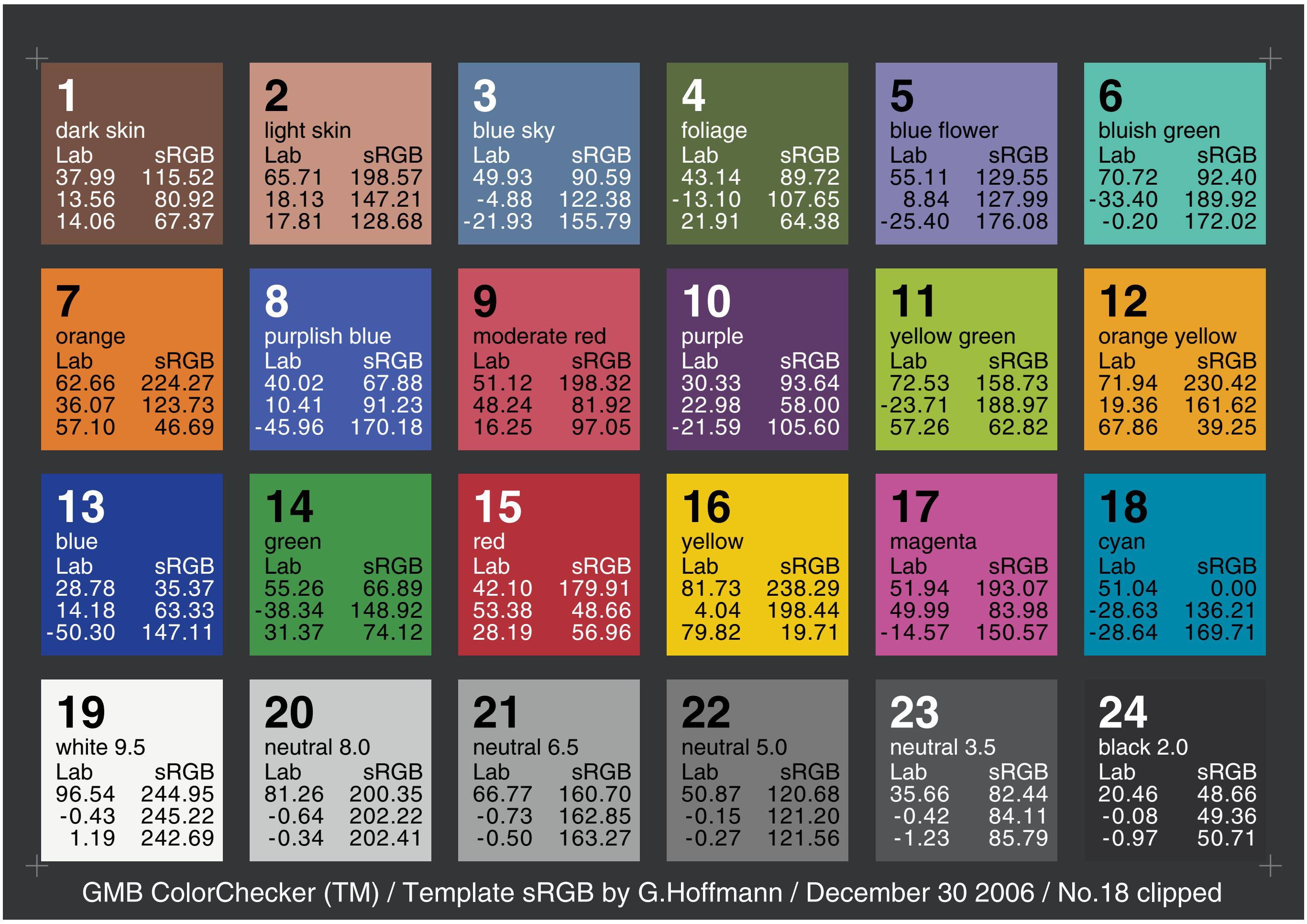
Contents

1. Introduction

Berns [3] describes in appendix G how to find a linear relation between CIE XYZ values and RGB values as measured by an RGB device (scanner or camera): $\mathbf{X}_i = \mathbf{C}\mathbf{R}_i$.
The target GretagMacbeth ColorChecker contains m=24 patches with known CIELab values, from which the XYZ values can be derived. The 3x3 matrix **C** contains n=9 unknown parameters.
The matrices **X** and **R** for target values and measured values have 3 rows but 24 columns. The problem is linear but overdetermined.
The Gaussian least squares method is used to find the best estimation by averaging. *Berns* introduces for this purpose the so-called pseudoinverse in appendix A.

This document explains the basic method for a linear vector equation and for a linear matrix equation as necessary for the color space transform. Furtheron it is shown that one does not need any inverse matrix - the task results in solving three vector equations with symmetric system matrices.

According to [1] and [5], a rectangular matrix can have several pseudoinverses. The pseudoinverse as defined by the least squares method is a special case, often called Moore-Penrose inverse. One does not need more than the old Gaussian normal equation for deriving this inverse.



GMB ColorChecker (registered trademark)
PostScript version with LAB and sRGB numbers

2.1 Matrix fundamentals

The nomenclature and some non-trivial results are essentially based on *Zurmühl*, *Falk* [1]. In fact this book was written by *Falk* more than twenty years ago as an entirely new undertaking.
A set of n real variables is written as a column matrix. A vector is a geometrical or mechanical object which can be assigned to a column matrix, and vice versa. In this doc one does not need vectors, but occasionally the word *vector* is used as an abbreviation. A column matrix can be written transposed as a row matrix.

Column matrix:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}$$

Row matrix:

$$\mathbf{x}^T = [x_1, x_2, \dots, x_n]$$

Inverse product for square matrices with existing inverse matrices:

$$(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$$

Transposed product for compatible rectangular matrices:

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T$$

Quadratic forms:

$$Q = \mathbf{x}^T\mathbf{A}\mathbf{x}$$

$$P = \mathbf{y}^T\mathbf{A}\mathbf{x} = \mathbf{x}^T\mathbf{A}^T\mathbf{y}$$

Derivatives:

$$\frac{\partial Q}{\partial \mathbf{x}} = 2\mathbf{A}\mathbf{x} = \begin{bmatrix} \partial Q/\partial x_1 \\ \partial Q/\partial x_2 \\ \dots \\ \partial Q/\partial x_n \end{bmatrix}$$

$$\frac{\partial P}{\partial \mathbf{x}} = \mathbf{A}^T\mathbf{y}$$

$$\frac{\partial P}{\partial \mathbf{y}} = \mathbf{A}\,\mathbf{x}$$

2.2 Overdetermined vector equation

A set of m linear equations for n unknowns with m=n can be solved directly. For m>n the Gaussian least squares algorithm is quite common. In fact the solutions for somewhat contradictional equations are averaged by minimizing the sum of the squared errors (residuals) of each equation. Algorithms for *Gaussian* and *Cholesky* equation solvers are available in [1],[2] and in many other text books about numerical mathematics.

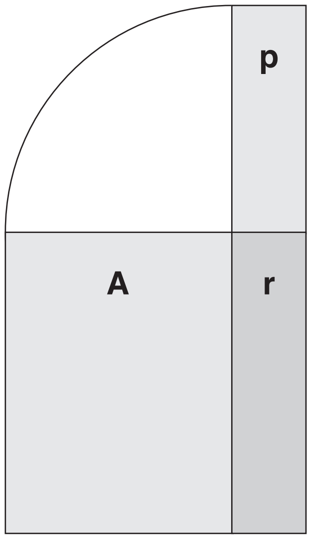
Use m measured values **r** for n unknown parameters in **p**:

$$\mathbf{p} = [p_1, p_2, \dots, p_n]^T$$

$$\mathbf{r} = [r_1, r_2, \dots, r_m]^T$$

$$\mathbf{A} = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \dots & \dots & \dots \\ a_{m1} & \dots & a_{mn} \end{bmatrix}$$

$$\mathbf{Ap} = \mathbf{r}$$



The structural visualization for matrix multiplications was invented about 1950 by *Sigurd Falk*.

For $m = n$ apply a Gaussian equation solver.

For $m > n$ use the Gaussian least squares algorithm.

Error vector:

$$\mathbf{e} = \mathbf{Ap} - \mathbf{r}$$

Scalar error function, the Euclidian norm:

$$\begin{aligned} F(\mathbf{p}) = \mathbf{e}^T\mathbf{e} &= (\mathbf{Ap}-\mathbf{r})^T(\mathbf{Ap}-\mathbf{r}) \\ &= \mathbf{p}^T\mathbf{A}^T\mathbf{A}\,\mathbf{p} - \mathbf{r}^T\mathbf{A}\,\mathbf{p} - \mathbf{p}^T\mathbf{A}^T\mathbf{r} + \mathbf{r}^T\mathbf{r} \\ &= \mathbf{p}^T\mathbf{A}^T\mathbf{A}\,\mathbf{p} - 2\mathbf{p}^T\mathbf{A}^T\mathbf{r} + \mathbf{r}^T\mathbf{r} \end{aligned}$$

Find local minimum:

$$\frac{\partial F}{\partial \mathbf{p}} = 2\mathbf{A}^T\mathbf{A}\,\mathbf{p} - 2\mathbf{A}^T\mathbf{r} = \mathbf{0}$$

Gaussian normal equation:

$$\mathbf{A}^T\mathbf{A}\,\mathbf{p} = \mathbf{A}^T\mathbf{r}$$

$\mathbf{A}^T\mathbf{A}$ is symmetric and (normally) positive definite.
Solve n equations for n unknowns by Cholesky's method or by a Gaussian equation solver if n is small.

It can be shown that the solution **p** is orthogonal to the error vector **e**:

$$\mathbf{e}^T\mathbf{p} = 0$$

A differential variation $\delta\mathbf{p}$ does not change the error.

Formally we can find the solution by the so-called pseudoinverse.
This is never recommended for numerical calculations.

$$\mathbf{p} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{r}$$

2.3 Overdetermined matrix equation / Concept

This chapter is closely related to *Berns* [3], appendix A and G. The GretagMacbeth ColorChecker consists of m=24 color patches which are described with high accuracy by CIELab coordinates. These values L*,a*,b* can be converted into CIE XYZ values [4].
An RGB device (camera or scanner) measures for each patch the values RGB in a linear RGB space. The most simple characterization of the device is a plain matrix **C**. This matrix with n=9 parameters converts RGB into XYZ.
The unknown matrix **C** is found by solving three overdetermined vector equations in the gray box for each column - no pseudo-inverse is required.

A patch color is predefined by $\mathbf{X}_i = \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix}$.

A measured value set given by $\mathbf{R}_i = \begin{bmatrix} R_i \\ G_i \\ B_i \end{bmatrix}$.

The task: find the best fitting matrix **C** for

$\mathbf{X}_i \;=\; \mathbf{C}\mathbf{R}_i$ with row vectors $\mathbf{c}^{iT}$

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{c}^{1T} \\ \mathbf{c}^{2T} \\ \mathbf{c}^{3T} \end{bmatrix}$$

The predefined colors are combined in the matrix **X**:

$$\mathbf{X} = \begin{bmatrix} X_1 & X_2 & \dots & X_m \\ Y_1 & Y_2 & \dots & Y_m \\ Z_1 & Z_2 & \dots & Z_m \end{bmatrix} = \left[\mathbf{X}_1, \dots, \mathbf{X}_m\right] = \begin{bmatrix} \mathbf{x}^T \\ \mathbf{y}^T \\ \mathbf{z}^T \end{bmatrix}$$

The measured values are combined in the matrix **R**:

$$\mathbf{R} = \begin{bmatrix} R_1 & R_2 & \dots & R_m \\ G_1 & G_2 & \dots & G_m \\ B_1 & B_2 & \dots & B_m \end{bmatrix}$$

The first equation can be tra**n**sposed:

$$\mathbf{X}_i^T = \mathbf{R}_i^T \mathbf{C}^T$$

For all sets together we get:

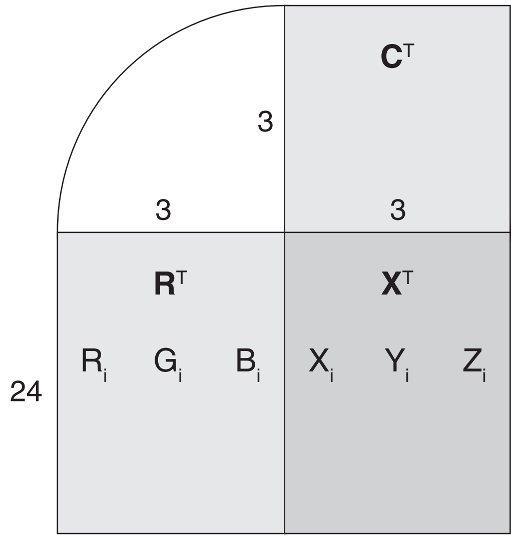
$$\mathbf{X}^T = \begin{bmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ \dots & \dots & \dots \\ X_m & Y_m & Z_m \end{bmatrix} = \mathbf{R}^T\mathbf{C}^T = \begin{bmatrix} R_1 & G_1 & B_1 \\ R_2 & G_2 & B_2 \\ \dots & \dots & \dots \\ R_m & G_m & B_m \end{bmatrix} \begin{bmatrix} c_{11} & c_{21} & c_{31} \\ c_{12} & c_{22} & c_{32} \\ c_{13} & c_{23} & c_{33} \end{bmatrix} = \mathbf{R}^T\left[\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3\right]$$

Now we extract the first column on both sides: =

$$\mathbf{x} = \begin{bmatrix} X_1 \\ X_2 \\ \dots \\ X_m \end{bmatrix} = \begin{bmatrix} R_1 & G_1 & B_1 \\ R_2 & G_2 & B_2 \\ \dots & \dots & \dots \\ R_m & G_m & B_m \end{bmatrix} \begin{bmatrix} c_{11} \\ c_{12} \\ c_{13} \end{bmatrix} = \mathbf{R}^T\mathbf{c}_1$$

Similarly for **y** and **z**, one can find three sets of linear equations:

$$\mathbf{R}^T\mathbf{c}_1 = \mathbf{x}$$
$$\mathbf{R}^T\mathbf{c}_2 = \mathbf{y}$$
$$\mathbf{R}^T\mathbf{c}_3 = \mathbf{z}$$



This arrangement of transposed matrices shows:
24 equations for 3x3 unknowns.

2.4 Overdetermined matrix equation / Solution and Pseudoinverse

So far we have three overdetermined vector equations.

Each one is handled by the Gaussian least squares algorithm
(multiply by the transposed system matrix):

$$\mathbf{R}\mathbf{R}^{\mathrm{T}}\mathbf{c}_1 = \mathbf{R}\mathbf{x}$$
$$\mathbf{R}\mathbf{R}^{\mathrm{T}}\mathbf{c}_2 = \mathbf{R}\mathbf{y}$$
$$\mathbf{R}\mathbf{R}^{\mathrm{T}}\mathbf{c}_3 = \mathbf{R}\mathbf{z}$$

For the numerical solution it is recommended to honour the symmetry of $\mathbf{R}\mathbf{R}^{\mathrm{T}}$ by Cholesky's method.
The three sets of linear equations can be combined in one:

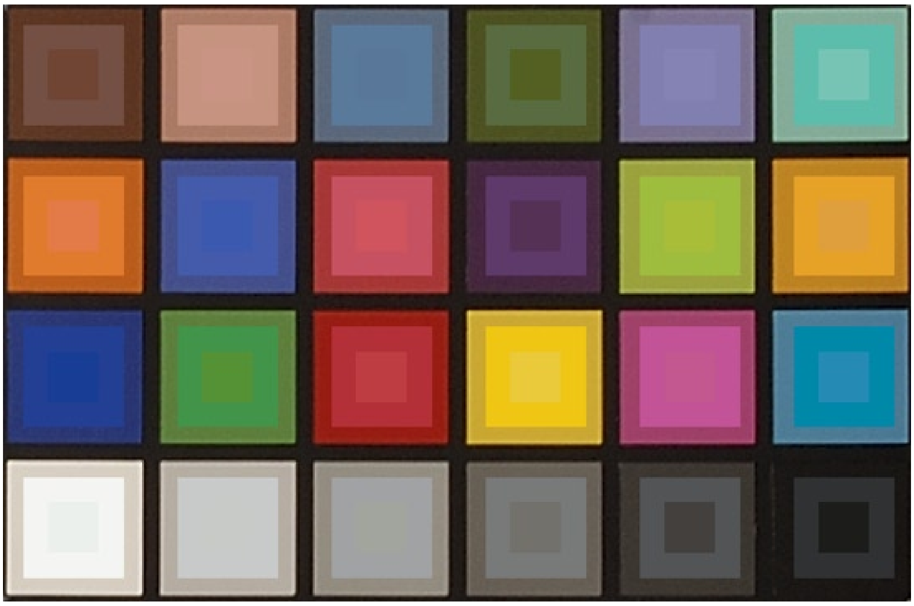
$$\mathbf{C}^{\mathrm{T}} = [\,\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3\,] = (\mathbf{R}\mathbf{R}^{\mathrm{T}})^{-1}\mathbf{R}[\mathbf{x}, \mathbf{y}, \mathbf{z}] = (\mathbf{R}\mathbf{R}^{\mathrm{T}})^{-1}\mathbf{R}\mathbf{X}^{\mathrm{T}}$$
$$\mathbf{C} = \mathbf{X}\mathbf{R}^{\mathrm{T}}(\mathbf{R}\mathbf{R}^{\mathrm{T}})^{-1}$$

This is the so-called pseudoinverse. For practical applications the pseudoinverse is not required.
Thanks to Szymon Bęczkowski for correcting an error in the formula for **C**.

3. Example

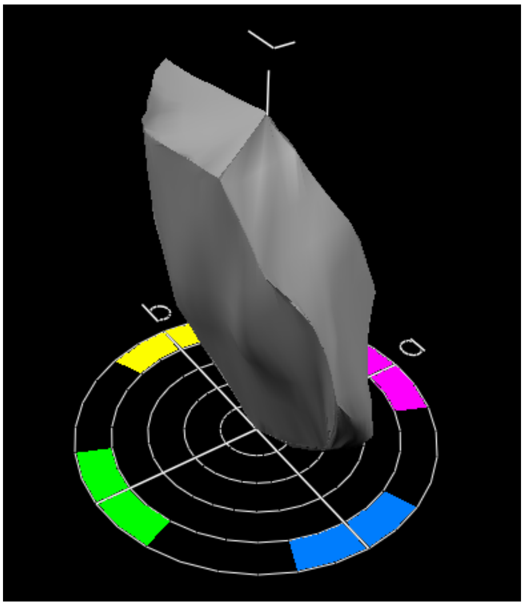
The camera calibration by a 3x3 matrix **C**, using the target GretagMacbeth ColorChecker, was tested for the digital camera Nikon D100 by a student [7].

The image shows the ColorChecker (on top of a painting). The squares in the color patches were drawn by MatLab .
Middle Square: true sRGB colors. Small square: photo colors after correction by matrix **C** .



Ideally, the middle square and the small square would have the same color. The correction is obviously not perfect. Probably the sensitivities of the camera sensor are not linear combinations of the so-called color matching functions [6].

Better results were achieved by nonlinear corrections, based on an ICC Profile, using commercial software [8]. The shape of the CIELab gamut volume indicates, that the ICC profile is not simply a matrix profile.



4. References

[1] R.Zurmühl + S.Falk
Matrizen und ihre Anwendungen, Teil 1
Springer-Verlag Berlin, Heidelberg ...
1984

[2] H.R.Schwarz
Numerische Mathematik
B.G.Teubner Stuttgart
1993

[3] R.S.Berns
Billmeyer and Saltzman‘s Principles of Color Technology
Third edition
John Wiley & Sons, New York, Chichester ...
2000

[4] G.Hoffmann
CIELab Color Space
http://docs-hoffmann.de/cielab03022003.pdf
2005

[5] http://mathworld.wolfram.com/Pseudoinverse.html

[6] G.Hoffmann
CIE Color Space
http://docs-hoffmann.de/ciexyz29082000.pdf

[7] I.B.Agutagama
Camera Color Calibration
Technical project 2007
Fachhochschule Oldenburg Ostfriesland Wilhelmshaven

[8] G.Hoffmann
Camera Calibration for Reproduction
http://docs-hoffmann.de/camcal17122006.pdf

This document
http://docs-hoffmann.de/leastsqu16112006.pdf

Gernot Hoffmann
November 16 / 2006 – March 04 / 2014
Website
Load browser / Click here