Color Calibration on Human Skin Images

Mahdi Amani[1], Håvard Falk[1,2], Oliver Damsgaard Jensen[1,3], Gunnar Vartdal[1,4], Anders Aune[1], and Frank Lindseth[1]

[1] Norwegian University of Science and Technology (NTNU), Norway
mahdi.amani,a.aune,frankl@ntnu.no
[2] Bekk Consulting AS, Akershusstranda 21, 0150 Oslo, Norway
havard.falk@bekk.no
[3] Itera ASA, Nydalsveien 28, 0484 Oslo, Norway
oliver.jensen@itera.no
[4] Picterus AS, Professor Brochs gate 8A, 7030 Trondheim, Norway
gunnar@picterus.com

Abstract. Many recent medical developments rely on image analysis, however, it is not convenient nor cost-efficient to use professional image acquisition tools in every clinic or laboratory. Hence, a reliable color calibration is necessary; *color calibration* refers to adjusting the pixel colors to a standard color space.
During a real-life project on neonatal jaundice disease detection, we faced a problem to perform skin color calibration on already taken images of neonatal babies. These images were captured with a smartphone (Samsung Galaxy S7, equipped with a 12 Mega Pixel camera to capture 4032x3024 resolution images) in the presence of a specific calibration pattern. This post-processing image analysis deprived us from calibrating the camera itself. There is currently no comprehensive study on color calibration methods applied to human skin images, particularly when using amateur cameras (e.g. smartphones). We made a comprehensive study and we proposed a novel approach for color calibration, Gaussian process regression (GPR), a machine learning model that adapts to environmental variables. The results show that the GPR achieves equal results to state-of-the-art color calibration techniques, while also creating more general models.

Keywords: Color Calibration · Color Correction · Skin Imaging

1 Introduction

Medical imaging is the core of recent medical science and the accuracy of the information highly relies on the environmental parameters such as light sources and camera sensors [1] which cause image inconsistencies. To overcome these issues (i.e., to make images illumination, camera sensors, and actual pixel values independent), we study, implement, and improve the state-of-the-art of color calibration techniques and we propose a novel approach for color calibration; the Gaussian process regression (GPR), a machine learning model that adapts

to environmental variables. The results show that the GPR achieves equal results to state-of-the-art color calibration techniques, while also creating more general models. We had faced this problem while working on a dataset of already taken images of neonates for jaundice disease detection.

1.1 Paper Structure

In the rest of this section, an introduction on human skin's light absorption, related color theory, colorcheckers, and color calibration preliminaries is presented. Then, the proposed solutions on color calibration and related machine learning regression techniques are presented in Section 2. In Section 3, the performance of color calibration techniques are presented and evaluated in terms of $\Delta E*_{a,b}$ using colorcheckers as ground truth and the results are compared. Finally, based on the given results, a conclusion is given in Section4.

1.2 Light Absorption on Human Skin

To understand optical skin diagnosis, we must first understand human skin and how each component's concentration alters the perceived color of skin. The perceived color of an object is due to the light that is reflected off the object. Molecules only reflect a subset of in-coming frequencies [2]. Skin, a layered organ protecting the human organism against environmental stress (e.g. heat, radiation, and infections) contains light-absorbing substances.

Skin consists of three main layers; epidermis, dermis, and the hypodermis. The visual features of skin captured by a camera recording the RGB values and their pattern can reveal a lot of information about the underlying substances, by different lightening and camera sensors. In practice, these RGB values are always dependent to each other (which they should not), hence, having a robust color calibration technique can solve this issue.

1.3 Delta E (ΔE)

While comparing results, a unified metric for color difference is required to express the difference between two colors correctly. The idea is that a color difference of $1\Delta E$ is the smallest color difference a human eye can detect. ΔE was first presented alongside the CIE L*a*b* color space in [3]. Given two colors in the CIE L*a*b* color space, (L_1^*, a_1^*, b_1^*) and (L_2^*, a_2^*, b_2^*), the simplest formula of ΔE is the Euclidean distance between these colors, as follows.

$$\Delta E_{ab}^* = \sqrt{(L_2^* - L_1^*)^2 + (a_2^* - a_1^*)^2 + (b_2^* - b_1^*)^2} \tag{1}$$

It is estimated that a human regards a $\Delta E_{ab}^* \approx 2.3$ difference as 'just noticeable' [3].

1.4 CIE Illuminant

The quality and energy of a *light source* is not always consistent and are often seen as unreliable and cannot technically be *reproduced*. To create light suited for colorimetric calculations, the CIE introduced the concept of standard illuminants. A standard illuminant is a theoretical source of visible light where its spectral power distribution is explicitly defined. Illuminants are divided into series describing source characteristics such as: Incandescent/Tungsten, Direct sunlight at noon, Average/North sky Daylight, Horizon Light, Noon Daylight, etc. Full list can be found on [4] and [5].

1.5 ColorChecker

A *colorchecker*, also known as a *calibration card*, is a physical set of colors defined in the CIE L*a*b* color space. The colors are, thus, defined regardless of light source or image capturing device[5]. If a colorchecker is included in an image, the RGB errors at each color patch can be calculated and provides information about the RGB variations in the image, which again can be used to correct the color errors. Here, we introduce two colorcheckers; the *Macbeth generic colorcheckers* and the *Picterus skin colorchecker*.

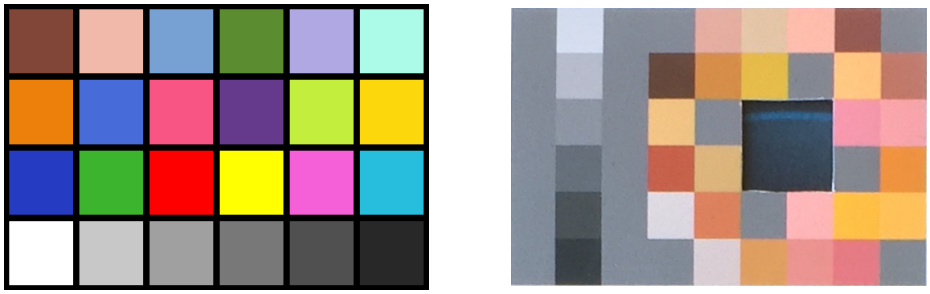
Macbeth Generic Colorcheckers The classic Macbeth colorchecker by [6] (Fig.1a) is one of the most commonly used reference targets in photographic work. The checker is designed to approximate colors found in nature. Six patches are different neutral gray, from black to white, where the spectral response of each patch is constant at all wavelengths and differ only in intensity. We referred to these six patches as the *grayscale* of the colorchecker. Other adaptations of McCamy's colorchecker have since been produced, here we present Datacolor's **SpyderCHECKR 24**. SpyderCHECKR 24 and its all target values are given in Fig. 2a and 2c.

Picterus Skin ColorChecker For jaundice detection on neonatal babies, we designed a custom colorchecker (Fig. 1b) targeted for human skin[6]. We will refer to this colorchecker as the **SkinChecker**, and all target values are given in Fig. 2b and 2d. The colors on the SkinChecker are based on simulated reflection spectra of neonate's skin with varying skin parameters. These reflection spectra have been printed using spectral printing, a technique which attempts to recreate the whole reflection spectrum instead of just the RGB color values [7].

Fig. 3 shows the color diversity of a SpyderCHECKR 24 and a SkinChecker. The figure highlights the difference between general and specialized color correction, drawing triangles to visualize the RGB color sub-spaces defined by the two color checkers. For general color correction, a model must be able to reproduce a

[5] However, by specifying a light source from CIE illuminant list, one can approximately define a colorchecker in RGB, but such generic RGB values are not to be fully trusted.

[6] For further information please contact Picterus AS at www.picterus.com.



(a) The classic Macbeth ColorChecker (b) The SkinChecker by Picterus AS

Fig. 1: The Macbeth ColorChecker Classic and the SkinChecker.

wide range of colors, as reflected by the SpyderCHECKR 24. The SkinChecker, on the other hand, contains substantially more data points in its focus area, allowing a model to optimize for more accurate color correction of human skin.

1.6 Color Calibration; Preliminaries and Related Works

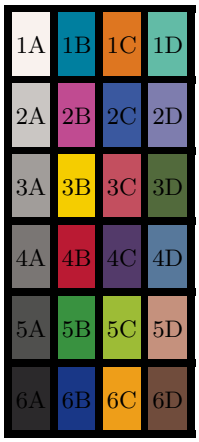
Variations in digital cameras' color responses (RGB values), caused by inaccurate sensors, result in device **dependent** values of R, G, and B. The task of correcting errors, in the captured RGB values, is referred to as *color calibration* (or *color correction*[7]). More precisely, *color calibration* refers to adjusting the pixel colors to a default/known/standard color space [9]. Color calibration involves mapping device dependent RGBs to corresponding **independent** color values (e.g. RGB or CIE XYZ) usually by using a colorchecker as reference. The device independent color values are often referred to as tristimulus values, and represent the same color regardless of visual system.

Look-up tables, least-squares linear and polynomial regression and neural networks are some of the methods described in literature regarding the mapping between RGB and tristimulus values.

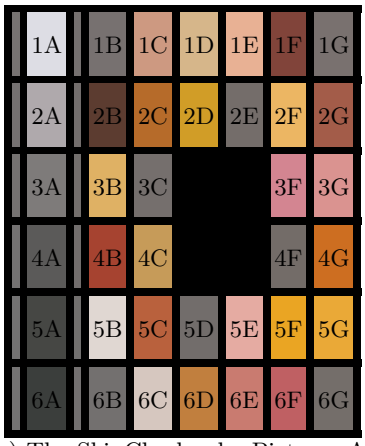
Look-Up Tables The trivial *look-up table* is a large collection of camera RGB examples and the corresponding target values, manually created to define a mapping between the two color spaces.

Least-Squares Linear Regression A linear mapping from camera RGBs to CIE L*a*b* triplets can be achieved through a 3×3 linear transform. If we let ρ define a three element vector representing the three camera responses (R, G, B) and $\mathbf{q}$ define the three corresponding L*, a*, b* values, a simple linear transform can be written as: $\mathbf{q} = \mathbf{M}\rho$,

[7] It's worth mentioning that a device (e.g. camera) is to be *calibrated* while images are to be *corrected*, hence, color calibration and color correction are slightly different. However, the concept is the same and here we consider them equivalent.



(a) The SpyderCHECKR 24



(b) The SkinChecker by Picterus AS

Patch	Name	L*a*b* L*	a*	b*	sRGB R	G	B
1A	Card White	96.04	2.16	2.60	249	242	238
2A	20% Gray	80.44	1.17	2.05	202	198	195
3A	40% Gray	65.52	0.69	1.86	161	157	154
4A	60% Gray	49.62	0.58	1.56	122	118	116
5A	80% Gray	33.55	0.35	1.40	80	80	78
6A	Card Black	16.91	1.43	-0.81	43	41	43
1B	Primary Cyan	47.12	-32.52	-28.75	0	127	159
2B	Primary Magenta	50.49	53.45	-13.55	192	75	145
3B	Primary Yellow	83.61	3.36	87.02	245	205	0
4B	Primary Red	41.05	60.75	31.17	186	26	51
5B	Primary Green	54.14	-40.76	34.75	57	146	64
6B	Primary Blue	24.75	13.78	-49.48	25	55	135
1C	Primary Orange	60.94	38.21	61.31	222	118	32
2C	Blueprint	37.80	7.30	-43.04	58	88	159
3C	Pink	49.81	48.50	15.76	195	79	95
4C	Violet	28.88	19.36	-24.48	83	58	106
5C	Apple Green	72.45	-23.57	60.47	157	188	54
6C	Sunflower	71.65	23.74	72.28	238	158	25
1C	Aqua	70.19	-31.85	1.98	98	187	166
2D	Lavender	54.38	8.84	-25.71	126	125	174
3D	Evergreen	42.03	-15.78	22.93	82	106	60
4D	Steel Blue	48.82	-5.11	-23.08	87	120	155
5D	Classic Light Skin	65.10	18.14	18.68	197	145	125
6D	Classic Dark Skin	36.13	14.15	15.78	112	76	60

(c) Values from the SpyderCHECKR 24 by [8]

Patch	L*a*b* L*	a*	b*	sRGB R	G	B
1A	88.38	1.51	-3.32	221	221	228
2A	70.14	2.72	-0.74	175	169	172
3A	51.97	1.43	0.66	126	123	122
4A	38.53	-0.15	0.93	91	90	89
5A	30.25	-1.12	2.06	70	71	68
6A	26.02	-1.50	0.71	59	62	60
1B	48.32	1.82	1.49	119	113	112
2B	28.82	12.48	13.19	92	60	48
3B	74.93	8.10	45.14	223	177	100
4B	42.03	39.70	31.89	166	67	48
5B	86.81	2.21	3.52	224	215	210
6B	47.62	0.74	1.01	115	112	111
1C	67.77	15.92	20.13	205	153	129
2C	52.60	24.83	47.00	182	107	42
3C	47.52	2.17	1.81	117	111	109
4C	66.73	8.33	40.28	198	155	89
5C	51.02	32.66	35.69	185	97	61
6C	81.35	4.01	5.79	214	199	191
1D	75.98	5.20	26.58	214	182	138
2D	68.04	8.81	63.77	209	157	39
5D	46.63	1.55	1.91	114	109	107
6D	58.96	19.64	44.55	193	127	62
1E	76.70	16.15	22.70	232	177	148
2E	46.84	2.26	2.76	116	109	106
5E	75.25	20.00	13.09	229	171	162
6E	59.69	29.13	18.90	201	123	112
1F	36.42	24.92	17.63	129	68	58
2F	77.67	10.08	49.47	236	182	99
3F	64.09	31.40	6.45	211	133	145
4F	46.33	2.08	3.07	115	108	104
5F	72.85	15.48	70.20	234	165	35
6F	52.44	38.26	16.14	191	96	99
1G	48.53	2.43	2.41	121	113	111
2G	46.97	26.89	23.61	163	92	73
3G	68.04	26.16	12.62	218	147	144
4G	56.56	32.21	55.49	205	110	33
5G	73.81	14.00	64.45	234	169	55
6G	47.21	1.44	3.02	116	110	106

(d) Values from SkinChecker

Fig. 2: SpyderCHECKR 24 and Picterus' SkinChecker.

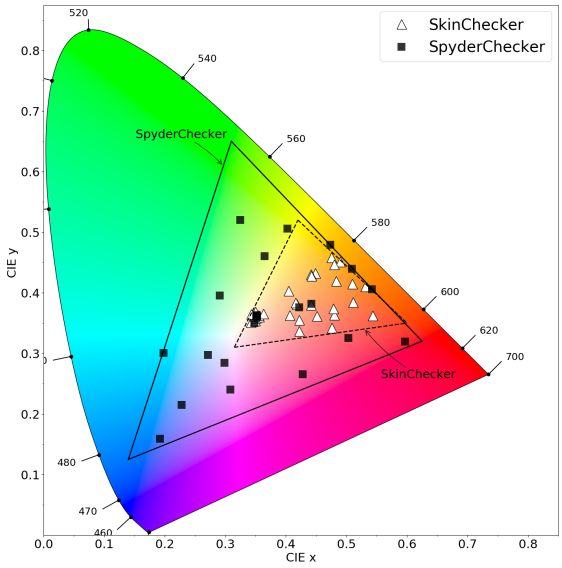


Fig. 3: The CIE xy Chromaticity diagram with the SpyderChecker 24 and SkinChecker's colors individually positioned. The SkinChecker contains skin related colors and resides only in the red-yellow part of the color space while the SpyderChecker 24 is a diverse colorchecker and is spread in a wider range.

M holds the coefficients (d_{ij}) of the transform that performs the actual color correction. The **linear color correction (LCC)** has proved to perform well in numerous studies, with the advantage of being independent of camera exposure ([10]), hence, we skip the details. It has also been known to produce significant errors when mapping RGBs to CIE L*a*b* values. Some studies show that LCC can, for some surfaces, generate errors up to $14\Delta E^*_{ab}$ [11]. However, the same study shows that on average the 3×3 LCC transform yields an error of $2.47\Delta E^*_{ab}$ on the 8-bit Professional Color Communicator ([12]).

Least-Squares Polynomial Regression A more modern approach for color correction is to assume that the relationship between RGB and target values is polynomial, not linear. This leads to a more complex method for color correction, **polynomial color correction (PCC)**, where the R, G, and B values at a pixel are extended by adding additional polynomial terms of increasing degree.

$$\rho = (R, G, B)^T \xrightarrow{\theta_k} \hat{\rho}_k = (R, G, B, ..., m)^T \tag{2}$$

We denote the k^{th} polynomial extension of an RGB-vector (ρ) by $\hat{\rho}_k$. The extension operator θ_k transforms a three-element column vector to an m-element column vector with a set of added polynomial terms, further derails in [11]. For a simple RGB case i.e. $\rho = (R, G, B)^T$ the polynomial expansions of 2^{nd}, 3^{rd}

and 4^{th} degrees are given below:

$$\hat{\rho_2} = [r, g, b, rg, rb, gb, r^2, g^2, b^2, 1] \tag{3}$$

$$\begin{aligned}\hat{\rho_3} = [&r, g, b, rg, rb, gb, r^2, g^2, b^2,\\ &rg^2, rb^2, gr^2, gb^2, br^2, bg^2, r^3, g^3, b^3, rgb, 1]\end{aligned} \tag{4}$$

$$\begin{aligned}\hat{\rho_4} = [&r, g, b, rg, rb, gb, r^2, g^2, b^2,\\ &rg^2, rb^2, gr^2, gb^2, br^2, bg^2, rgb, r^3, g^3, b^3,\\ &r^3g, r^3b, g^3r, g^3b, b^3r, b^3g, r^2g^2, r^2b^2, g^2b^2,\\ &r^2gb, g^2rb, b^2rg, r^4, g^4, b^4, 1]\end{aligned} \tag{5}$$

Using the extension operator, the three RGB values recorded in a pixel are extended, represented by 9, 19, and 34 numbers respectively. As apposed to LCC's 3×3 matrix, PCC is carried out by 9×3, 19×3, and 34×3 matrices. Similar to LCC, we find the coefficients that minimize $\mathbf{M}$, the $m \times 3$ PCC matrix.

Root-Polynomial Color Correction If the correct polynomial fit is chosen, PCC can significantly reduce the colorometric error from LCC. However, the PCC fit depends on sensors and illumination, where exposure alters the vector of polynomial components in a non-linear way. Hence, choosing the right polynomial fit is very important in PCC. To solve the exposure sensitivity of PCC, ([10]) present a polynomial-type regression related to the idea of fractional polynomials. Their method, named **root-polynomial color correction (RPCC)**, takes each term in a polynomial expansion to its kth root of each k-degree term, and is designed to scale with exposure. The root-polynomial extensions for $k = 2$, $k = 3$ and $k = 4$ are defined as:

$$\bar{\rho_2} = [r, g, b, \sqrt{rg}, \sqrt{rb}, \sqrt{gb}, 1] \tag{6}$$

$$\begin{aligned}\bar{\rho_3} = [&r, g, b, \sqrt{rg}, \sqrt{rb}, \sqrt{gb}, \sqrt[3]{rg^2}, \sqrt[3]{rb^2}, \sqrt[3]{gr^2}, \sqrt[3]{gb^2},\\ &\sqrt[3]{br^2}, \sqrt[3]{bg^2}, \sqrt[3]{rgb}, 1]\end{aligned} \tag{7}$$

$$\begin{aligned}\bar{\rho_4} = [&r, g, b, \sqrt{rg}, \sqrt{rb}, \sqrt{gb},\\ &\sqrt[3]{rg^2}, \sqrt[3]{rb^2}, \sqrt[3]{gr^2}, \sqrt[3]{gb^2}, \sqrt[3]{br^2}, \sqrt[3]{bg^2}, \sqrt[3]{rgb},\\ &\sqrt[4]{r^3g}, \sqrt[4]{r^3b}, \sqrt[4]{g^3r}, \sqrt[4]{g^3b}, \sqrt[4]{b^3r}, \sqrt[4]{b^3g},\\ &\sqrt[4]{r^2gb}, \sqrt[4]{g^2rb}, \sqrt[4]{b^{2rg}}, 1]\end{aligned} \tag{8}$$

We denote a root-polynomial extension of an RGB column vector ρ as $\bar{\rho}$. As with PCC, the RGB-extension naturally increases the size of the transform matrix $\mathbf{M}$. Thus, RPCC is performed by a 7×3, 13×3, and 22×3 matrices.

Machine Learning Approaches An alternative approach for color correction is the use of machine learning approaches. The most used techniques are

Support Vector Machines (SVMs) and fully-connected feed-forward neural networks. Artificial neural network (ANN) have shown to be robust when optimized correctly, and achieve equally good results as a well fitted polynomial approach ([13]). However, both SVMs and fully-connected neural networks require extensive hyperparameter-tuning, a tedious process performed through trial and error, or using the computational expensive grid-search with cross-validation.

On the other hand, **Gaussian processes (GPs)** are widely recognized as a powerful, yet practical tool for solving both classification and non-linear regression ([14]). A GP has not been applied to the field of color calibration. A GP can be thought of as a generalization of the Gaussian probability distribution over a finite vector space to a function space of infinite dimensions ([15]). The processes are probabilistic models of functions and are used for solving both classification and non-linear regression problems. To be more precise, a GP is used to describe a distribution over functions $f(x)$ such that any finite set of function values $f(x_1), f(x_2), ..., f(x_n)$ have a joint Gaussian distribution [16, Chapter 2].

2 Color Calibration Techniques and Results

This section presents the color calibration techniques which are performed on our dataset. The dataset is a collection of 564 images acquired from St. Olav's hospital[8] depicting neonates. In each image, a SkinChecker is placed on the chest, exposing the skin color within the SkinChecker. This dataset is collected from 141 unique neonates by smartphones, where four images were taken with different ranges and lightening (e.g. with or without flash) from each neonate. For each image (Fig. 4a), illumination varies, and inconsistently shifts RGB values. Then the SkinChecker segment is extracted (Fig. 4b). With the SkinChecker as the reference point for the color correction solutions, our goal is to apply pixelwise color correction (Fig. 4c), trying to restore the original colors.

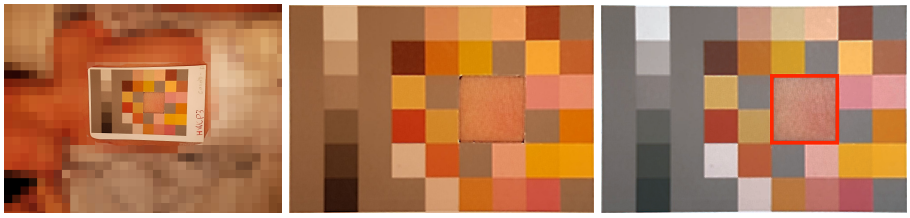
For fieldwork applications, we require the color correction algorithm to be device independent and show high robustness towards illumination. We used SkinChecker[9] to compare these methods using an evaluation method as follows.

2.1 Leave-one-out cross-validation evaluation method (EM)

For each individual color (on the in-scene colorchecker), we exclude 10% of the sample area on all sides, to eliminate potentially faulty segmentation, and sample the average RGB value. The resulting average RGBs and corresponding CIE L*a*b* values (from the colorchecker) are used to create color correction models.

To evaluate the performance of a model, we perform leave-one-out cross-validation which is an iterative approach where one color patch is withheld every run. For each iteration, the remaining colors are used to build a color correction model, predicting the CIE L*a*b* value of the withheld color patch. By the end of the iteration, all color patches have been withheld and the ΔE^*_{ab} is calculated.

[8] NTNU's hospital located in Trondheim, Norway.
[9] We also performed the same evaluation on SpyderCHECKR achieving similar results.



(a) Raw image (censored) (b) Raw segmented SkinChecker (c) Color corrected SkinChecker

Fig. 4: Color calibration pipeline. The visible skin patch within the SkinChecker (highlighted by the red square in Fig. 4c), represents the color calibrated image.

2.2 Color Calibration Frameworks

Literature describes a wide range of polynomial and root-polynomial fits to perform color correction. We are, however, unaware of previous work that proposes a solution as to which extension is to be applied in a given scenario. To overcome the practical issues related to color correction, first, we create two color correction frameworks, polynomial color correction framework (PCCF) and root-polynomial color correction framework (RPCCF), that are implementations of widely used polynomial and root-polynomial extensions. Finally, we test our framework, the novel GPR for color correction. Due to its extreme modeling flexibility, and added noise kernels, in the section 3, we show that the GPR well compete and even outperforms state-of-the-art approaches, creating highly complex models without overfitting. These frameworks are defined by their set of extension operators that transform sensor RGB values to m-element column vectors of an arbitrary combination or power. We will refer to these extension operators as internal methods θ_k (e.g. $(r,g,b)^T \xrightarrow{\theta_k} \hat{\rho}_2(r,g,b,r^2,g^2,b^2)^T$). Each framework builds a color correction model for each of its internal methods (i.e. polynomial expansion) using least-squares regression to solve. Doing so, results in an $m \times 3$ matrix $\mathbf{M}$, mapping

$$\hat{\rho} \xrightarrow{\mathbf{M}} (L^*, a^*, b^*). \tag{9}$$

Polynomial Color Correction Framework Solution (PCCF) To create the set of polynomial extension operators (i.e., the set of internal methods), we apply an iterative scaling approach, increasing the complexity of internal methods for each iteration. We start off by adding a constant term to the LCC:

$$\rho = (r,g,b)^T \xrightarrow{\theta_1} \hat{\rho}_1 = (r,g,b,1)^T \tag{10}$$

While not a significant alteration, the added term gives the least-squares regression some leeway. Continuing, we follow the polynomial orders, and combinations of R, G, and B, and create the following collection of RGB-extension

operators. The first subscript denotes the polynomial order, and the second subscript denotes the method id within the polynomial order.

$$\hat{\rho}_{1,1} = [r, g, b, 1] \tag{11}$$

$$\hat{\rho}_{2,1} = [r, g, b, rg, rb, gb, 1] \tag{12}$$

$$\hat{\rho}_{2,2} = [r, g, b, r^2, g^2, b^2, 1] \tag{13}$$

$$\hat{\rho}_{3,1} = [r, g, b, rg, rb, gb, r^2, g^2, b^2, rgb, 1] \tag{14}$$

$$\hat{\rho}_{3,2} = [r, g, b, r^2g, r^2b, g^2r, g^2b, b^2r, b^2g, rgb, 1] \tag{15}$$

$$\hat{\rho}_{3,3} = [r, g, b, rg, rb, gb, r^2, g^2, b^2, r^3, g^3, b^3, rgb, 1] \tag{16}$$

$$\hat{\rho}_{3,4} = [r, g, b, rg, rb, gb, r^2, g^2, b^2, r^2g, r^2b, g^2r, g^2b, b^2r, b^2g, rgb, 1] \tag{17}$$

$$\hat{\rho}_{3,5} = [r, g, b, rg, rb, gb, r^2, g^2, b^2, r^2g, r^2b, g^2r, g^2b, b^2r, b^2g, r^3, g^3, b^3, rgb, 1] \tag{18}$$

Root-Polynomial Color Correction Framework Solution (RPCCF) For a fixed exposure, PCC has shown significantly better results than a 3×3 LCC [11]. However, as pointed out by [10], exposure changes the vector of polynomial components in a nonlinear way, resulting in hue and saturation shifts. Their solution, RPCC, claims to fix these issues by expanding the RGB terms with root-polynomial extensions instead. We adopt their idea and create the RPCCF. The framework includes the root-polynomial extensions suggested in their paper, where the subscript of $\bar{\rho}$ denotes the root-polynomial order.

$$\bar{\rho}_2 = [r, g, b, \sqrt{rg}, \sqrt{rb}, \sqrt{gb}, 1] \tag{19}$$

$$\bar{\rho}_3 = [r, g, b, \sqrt{rg}, \sqrt{rb}, \sqrt{gb}, \sqrt[3]{rg^2}, \sqrt[3]{rb^2}, \sqrt[3]{gr^2}, \sqrt[3]{gb^2}, \sqrt[3]{br^2}, \sqrt[3]{bg^2}, \sqrt[3]{rgb}, 1] \tag{20}$$

$$\bar{\rho}_4 = [r, g, b, \sqrt{rg}, \sqrt{rb}, \sqrt{gb}, \sqrt[3]{rg^2}, \sqrt[3]{rb^2}, \sqrt[3]{gr^2}, \sqrt[3]{gb^2}, \sqrt[3]{br^2}, \sqrt[3]{bg^2}, \sqrt[3]{rgb}, \sqrt[4]{r^3g}, \sqrt[4]{r^3b}, \sqrt[4]{g^3r}, \sqrt[4]{g^3b}, \sqrt[4]{b^3r}, \sqrt[4]{b^3g}, \sqrt[4]{r^2gb}, \sqrt[4]{g^2rb}, \sqrt[4]{b^2rg}, 1] \tag{21}$$

Gaussian Process Regression Solution GPR is a very flexible machine learning technique that requires no explicit tuning of hyperparameters (other than the choice of kernel). For color correction, we train three separate GPs, one for each color channel (R, G, B). To predict a new color, the three GPR models take the RGB input color and predicts their respective CIE L*a*b* intensity

value (e.g., L* or a* or b*). The three individual results are combined to create the new CIE L*a*b* color coordinate.

The GPR implementation is initiated with a prior's covariance, specified by a kernel object. The hyperparameters of the kernel are optimized using gradient-descent on the marginal likelihood function during fitting, equivalent to maximizing the log of the marginal likelihood (LML). This property makes GPR superior to other supervised learning techniques, as it avoids heavy computational validation approaches, like cross-validation, to tune its hyperparameters. The LML may have multiple local optima, and trail and error testing is employed to verify that the optimal or close to the optimal solution is found by starting the optimizer repeatedly.

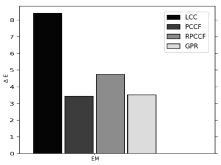
Choosing Kernel: GPR is a general method that can be extended to a wide range of problems. To implement GPs for regression purposes, the prior of the GP needs to be specified by passing a kernel. Widely used kernels are empirically tested; *RBF*, *Matérn*, and *rational quadratic (RQ)*. Additionally, combinations of *Constant-* and *White kernel* are added to the kernel function, to find the best performing kernel composition.

3 Experimental Results

In this section, using skinChecker, the proposed color calibration methods are evaluated. The models are evaluated on all images in the jaundice dataset, thus testing each solution in terms of skin color correction. Based on the sampled data and the corresponding CIE L*a*b* triplets, each color correction solution builds a model, evaluated according EM. Figures 5a and 5b show the results of the experiment and their bar plot visualization, evaluating the LCC, PCCF, RPCCF, and GPR. The dataset is collected through fieldwork, and the light sources are not reproducible in terms of a single CIE illuminant.

Solutions	LCC	PCCF	RPCCF	GPR
Results	8.42	**3.45**	4.74	**3.53**

(a) EM results.



(b) Bar plot visualization.

Fig. 5: Experimental results of LCC, PCCF, RPCCF, and GPR.

To give insight in the proposed color correction frameworks, we present the results for each internal method. Table 1a shows the internal results for the PCCF, where $\hat{\rho}_{3,1}$ is the best performing internal methods in terms of EM. The RPCCF results are more polarized, where $\bar{\rho}_2$ outperforms the rivaling internal methods by a large margin.

method	$\hat{\rho}_{11}$	$\hat{\rho}_{21}$	$\hat{\rho}_{22}$	$\hat{\rho}_{31}$	$\hat{\rho}_{32}$	$\hat{\rho}_{33}$	$\hat{\rho}_{34}$	$\hat{\rho}_{35}$
size	4x3	7x3	7x3	11x3	11x3	14x3	17x3	20x3
EM	4.84	**3.63**	**3.69**	**3.53**	**3.77**	4.05	5.04	6.64

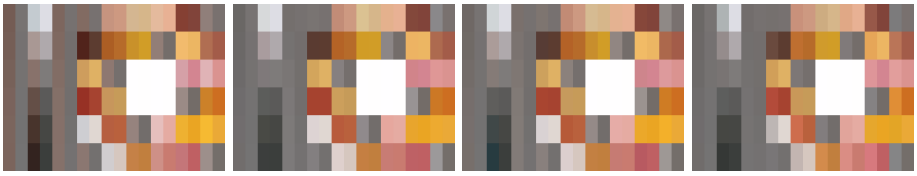
(a) EM results for all internal PCCF methods.

method	$\bar{\rho}_2$	$\bar{\rho}_3$	$\bar{\rho}_4$
size	4x3	7x3	7x3
EM	**4.74**	6.04	13.73

(b) Internal RPCCF methods.

Table 1: All color correction evaluation method results for all internal PCCF methods (a) and all internal RPCCF methods (b). The best performing (i.e. lowest ΔE^*_{ab}) internal methods are highlighted in bold.

Four visualizations of color corrected SkinCheckers using LCC, PCCF, RPCCF, and GPR respectively, are illustrated in Fig. 6. The color corrected image is randomly selected from the dataset.



(a) LCC (b) PCCF (c) RPCCF (d) GPR

Fig. 6: Color correction models are trained on an imaged SkinChecker and evaluated on the same image. Each color patch, in the reconstructed image, is divided into *two parts*: **left** shows the color corrected color, **right** shows the ground truth target color. All images are recreated from CIE L*a*b* coordinates to RGB with CIE D65 illuminant.

4 Conclusion

A reliable color calibration is necessary to avoid using professional image acquisition tools in every clinic or laboratory. To our best knowledge, there is currently no comprehensive study on color calibration methods applied to human skin images, particularly when using amateur cameras. In this work, we made a comprehensive study and we also proposed a novel approach for color calibration, Gaussian process regression (GPR), a machine learning model that adapts to environmental variables. The results indicate that our extended version of Polynomial Color Correction (PCCF) and GPR are viable solutions when color correcting skin, while GPR also creates more general models. We conclude that our solution can be used in a variety of human skin analyses and is an affordable screening alternative to expensive professional image acquisition tools.

References

[1] G. Dougherty, *Digital image processing for medical applications*. Cambridge University Press, 2009.

[2] M. Kerker, *The scattering of light and other electromagnetic radiation: physical chemistry: a series of monographs*. Academic press, 2013, vol. 16.

[3] G. Sharma, *Digital Color Imaging Handbook*. Boca Raton, FL, USA: CRC Press, Inc., 2002, ISBN: 084930900X.

[4] D. Pascale, "A review of rgb color spaces... from xyy to r'g'b'", *Babel Color*, vol. 18, pp. 136–152, 2003.

[5] HunterLab, "Equivalent white light sources and cie illuminants", Applications note, 2005, [Online]. Available: https://web.archive.org/web/20050523033826/http://www.hunterlab.com:80/appnotes/an05_05.pdf.

[6] C. S. McCamy, H. Marcus, J. Davidson, *et al.*, "A color-rendition chart", *J. App. Photog. Eng*, vol. 2, no. 3, pp. 95–99, 1976.

[7] C. C. Brito, "Spectral printing for monitoring jaundice in newborns", Internal article, 2016.

[8] D. company website, *Spydercheckr 24*, https://www.datacolor.com/wp-content/uploads/2018/01/SpyderCheckr_Color_Data_V2.pdf, [Online; Last accessed 14 May 2019], 2008.

[9] H.-C. Lee, *Introduction to Color Imaging Science*. Cambridge University Press, 2005. DOI: 10.1017/CBO9780511614392.

[10] G. D. Finlayson, M. Mackiewicz, and A. Hurlbert, "Color correction using root-polynomial regression", *IEEE Transactions on Image Processing*, vol. 24, no. 5, pp. 1460–1470, 2015, ISSN: 1057-7149. DOI: 10.1109/TIP.2015.2405336.

[11] H. Guowei, L. M. Ronnier, and R. P. A., "A study of digital camera colorimetric characterization based on polynomial modeling", *Color Research & Application*, vol. 26, no. 1, pp. 76–84, 2001. DOI: 10.1002/1520-6378(200102)26:1⟨76::AID-COL8⟩3.0.CO;2-3.

[12] J. Park and K. Park, "Professional colour communicator-the definitive colour selector", *Coloration Technology*, vol. 111, no. 3, pp. 56–57, 1995.

[13] V. Cheung, S. Westland, D. Connah, and C. Ripamonti, "A comparative study of the characterisation of colour cameras by means of neural networks and polynomial transforms", *Coloration Technology*, vol. 120, no. 1, pp. 19–25, Jan. 2004, ISSN: 1478-4408. DOI: 10.1111/j.1478-4408.2004.tb00201.x.

[14] C. K. I. Williams, "Prediction with gaussian processes: From linear regression to linear prediction and beyond", in *Learning and Inference in Graphical Models*, Kluwer, 1997, pp. 599–621.

[15] D. J. MacKay, "Introduction to gaussian processes", *NATO ASI Series F Computer and Systems Sciences*, vol. 168, pp. 133–166, 1998.

[16] C. E. Rasmussen and C. K. Williams, *Gaussian process for machine learning*. MIT press, 2006, ISBN: 978-0-262-18253-9.