

A Color Image Analysis Tool to Help Users Choose a Makeup Foundation Color*

Yafei Mao^a, Christopher Merkle^b, Jan P. Allebach^a

^a School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, U.S.A.

^b MIME Inc. Vancouver, WA 98660, U.S.A.

Abstract

This paper presents an approach to predict the color of skin-with-foundation based on a no makeup selfie image and a foundation shade image. Our approach first calibrates the image with the help of the color checker target, and then trains a supervised-learning model to predict the skin color. In the calibration stage, We propose to use three different transformation matrices to map the device dependent RGB response to the reference CIE XYZ space. In so doing, color correction error can be minimized. We then compute the average value of the region of interest in the calibrated images, and feed them to the prediction model. We explored both the linear regression and support vector regression models. Cross-validation results show that both models can accurately make the prediction.

Introduction

With the rapid evolution of mobile phone technologies, people can do more and more things on their phones. Virtual makeup try-on is one of the popular mobile applications nowadays. It allows users to test out makeup shades via images or live camera, which makes cosmetics shopping a lot more convenient and fun. Many researches have focused on this field. Bhatti et al [1] developed a mobile image-based application to give women personalized cosmetics recommendations. Tong et al [2] extracted makeup information from before-and-after image pairs and transferred the makeup to a new face. Work in [3] is based on a similar idea. There are also some papers using deep neural networks [4, 5] to analyze makeup styles. Most of these papers are targeting facial attribute analysis and style transfer. In this paper, we will study the color change of the skin before-and-after the makeup is applied. We will focus on a specific line of foundation products.

Taking images with a mobile phone can be easy, but extracting reliable colors from the images remains a problem. Because of imperfect lighting conditions, different camera sensors' sensitivities, and various post-processing in the image processing pipeline of the camera, the same product can look different in different images.

To address this color disparity issue, researchers have developed various color correction algorithms. These algorithms can be generally classified into two groups: color constancy and image calibration. Color constancy is done by estimating and removing the influence of illuminations. Some popular color constancy algorithms include Gray World and Max RGB. A major limitation of these two algorithms is that they strongly rely on particular assumptions. If some of the assumptions do not hold, then the

estimation can be inaccurate. Some more recent neural network algorithms [6–8] can achieve high accuracy but also require lots of computer resources, compared to some traditional methods. On the other hand, image calibration approaches are simple and effective. They directly map the device-dependent RGB color values to some standard color values with the help of a calibration target. This mapping can be applied to any camera and makes very few assumptions about the characteristics of the images taken by the camera. The mapping includes three-dimensional look-up tables combined with interpolation and extrapolation [9], machine learning models [10], and neural networks [11, 12]. Despite the fact that a wide variety of calibration methods are available, researchers have favored regression-based approaches [13–15] due to their simplicity and feasibility.

In this paper, we propose an image analysis tool that can predict the skin-with-foundation color based on the color information retrieved from calibrated selfie and foundation swatch images. To avoid illuminance inconsistency across multiple images of the same subject, we use a protocol to collect image data under controlled lighting conditions. To minimize color correction errors, we group the color patches on the color checker into three sets and compute the mapping from the camera-dependent RGB space to the standard CIE XYZ for these three sets separately. Next, the CIE $L^*a^*b^*$ color coordinates of the skin pixels, foundation pixels, and the skin with foundation pixels are extracted. Finally, a linear regression model and a support vector regression (SVR) [16] model are trained using these color coordinates.

Data Collection

The image calibration method proposed in this paper relies on a standard calibration target. We used the X-rite ColorChecker digital SG 140, which is shown in Fig. 1. The target foundation products are shown in Fig. 2 (b). The detailed procedure is as the following:

- Take a selfie photo with no foundation applied.
- Choose 3 or 4 foundation shades that are close to the actual skin tone by testing each possible match on the skin.
- Evenly apply the foundation on the skin using a new and clean sponge.
- Wait about three minutes for the foundation to dry, and then take a selfie photo.
- Remove the foundation completely using makeup removal wipes.
- Wait about three minutes for the skin to calm down, and apply the next shade.
- Repeat the process until all chosen shades are applied and

*Research supported by MIME Inc. Vancouver, WA.

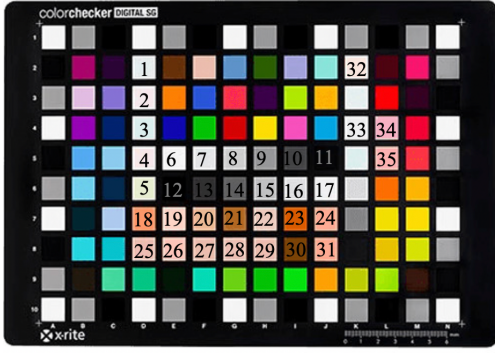


Figure 1: X-rite digital SG 140 color checker. Patches that are used in the color correction stage are numbered from 1 to 35.

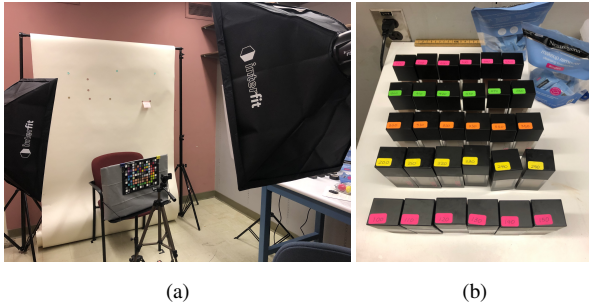


Figure 2: Data collection experiment (a) lab setting and (b) the makeup foundation bottles.

photographed.

The experiment was conducted under controlled lighting in a lab. The lab setting is shown in Fig. 2 (a). The light sources were three 4700K LED light bulbs. We installed diffuser panels to make the light as well-diffused as possible. The color checker and the mobile phone are mounted on tripods. The subject used a Bluetooth remote button to take selfie photos. Each subject was instructed to sit at a fixed distance from the camera, so that the amount of the light reflected from his or her face is about the same for all images. The images in Fig. 3 are sample original photos that we collected during this experiment.

Image Calibration

In this section, we will present the details of the image calibration procedure, which will lay the groundwork for an accurate prediction model. We will start by introducing the skin pixel detection algorithm. And then we describe the two major steps in the proposed calibration framework. Finally, the calibration performance is evaluated by means of the color differences in CIE $L^*a^*b^*$ space.

Skin Detection

Since we are mainly interested in analyzing and predicting skin colors, the facial skin is first segmented. A fast and efficient RGB-H-CbCr model [17] for this purpose is adopted. Under a uniform illumination condition, a skin pixel, defined by [17], should satisfy all of the following three criteria.

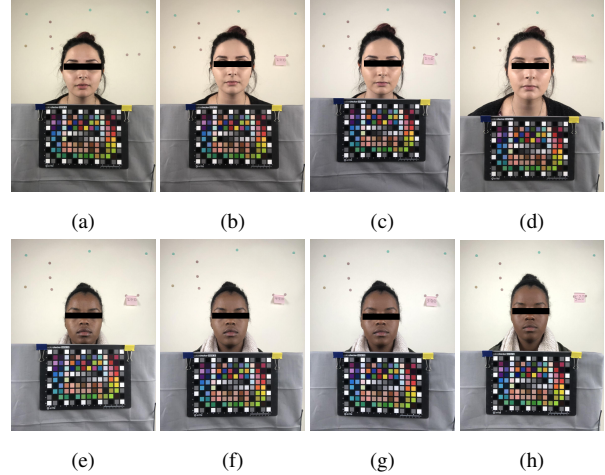


Figure 3: Sample original images. Subject No.1 (a light-skinned subject): (a) skin with no foundation, (b) skin with foundation shade No. 130, (c) skin with foundation shade No. 150, and (d) skin with foundation shade No. 200. Subject No. 2 (a dark-skinned subject): (a) skin with no foundation, (b) skin with foundation shade No. 450, (c) skin with foundation shade No. 500, and (d) skin with foundation shade No. 520.

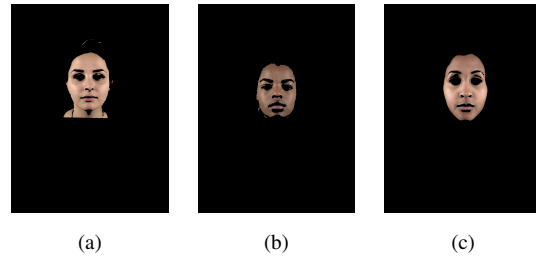


Figure 4: Skin detection output of the skin with no foundation images of (a) subject No.1 (light-skinned), (b) subject No. 2 (dark-skinned), and (c) subject No. 3 (tan-skinned).

Criterion 1:

$$\begin{aligned} R > 95 \quad \wedge \quad G > 40 \quad \wedge \quad B > 20 \\ \max(R, G, B) - \min(R, G, B) > 15 \\ |R - G| > 15 \quad \wedge \quad R > G \quad \wedge \quad R > B \end{aligned} \quad (1)$$

Criterion 2:

$$\begin{aligned} C_r &\leq 1.5862 \cdot C_b + 20 \quad \wedge \\ C_r &\geq 0.3448 \cdot C_b + 76.2069 \quad \wedge \\ C_r &\geq -4.5652 \cdot C_b + 234.5652 \quad \wedge \\ C_r &\leq -1.15 \cdot C_b + 301.75 \quad \wedge \\ C_r &\leq -2.2857 \cdot C_b + 432.85 \quad \wedge \end{aligned} \quad (2)$$

Criterion 3:

$$H < 25 \quad \vee \quad H > 230 \quad (3)$$

Here, \wedge denotes the logical AND operation, and \vee denotes the logical OR operation. An example output of this skin detection algorithm is shown in Fig. 4. It can be seen from Fig. 4 that although some non-skin pixels are picked up by the segmentation mask, most of the skin pixels are correctly identified. Note that

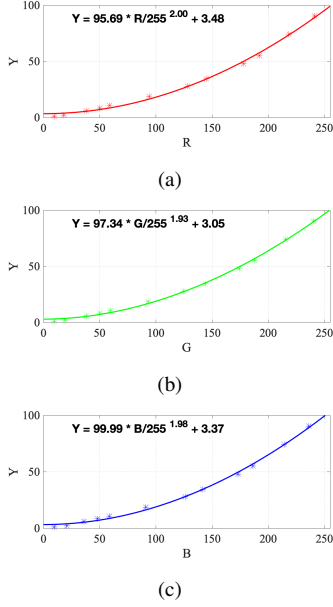


Figure 5: The gray balancing curves of the skin with no foundation image of subject No. 1. (a) The R channel. (b) The G channel. (c) The B channel.

we will find the average value within the skin region, as discussed later, so a few non-skin pixels will not significantly degrade the estimation. It can also be seen from Fig. 4 that the algorithm can effectively handle various skin complexions across different ethnicities.

Gray Balancing and Polynomial Transformation

Image calibration is a process that converts device-dependent RGB values into CIE XYZ values. So we need to obtain the RGB values of the target color patches in the original image as well as their corresponding CIE XYZ values. The patches of interest are labelled 1 through 35 in Fig. 1. The CIE XYZ values are measured with an X-Rite spectrophotometer under D50 illuminant, and the RGB values are extracted by averaging over the center region of each patch for each color channel.

Now that we have the $(RGB, CIE XYZ)$ pairs, we can estimate the color mapping between them. We will utilize a two-step process [15], namely gray balancing and polynomial regression. Gray balancing aims to remove the color cast and avoid having one particular dominant hue in the image. The methodology is based on [15]. We assume that the linear RGB values of each patch and the CIE Y (Luminance) value of the neutral gray patches are related by

$$R_l = G_l = B_l = Y. \quad (4)$$

Twelve neutral gray patches (patches No. 6 to No. 17 in Fig. 1) on the color checker are used in this step. This gives us twelve pairs of (R_γ, R_l) , (G_γ, G_l) , and (B_γ, B_l) for each image. We then

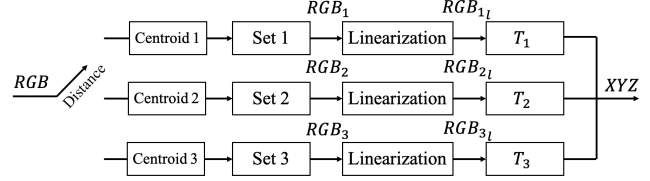


Figure 6: Block diagram of the calibration procedure.

fit a Gain-Gamma-Offset model [15] to them, such that

$$\begin{aligned} R_l &= \alpha_R \left(\frac{R_\gamma}{255} \right)^{\gamma_R} + O_R, \\ G_l &= \alpha_G \left(\frac{G_\gamma}{255} \right)^{\gamma_G} + O_G, \\ B_l &= \alpha_B \left(\frac{B_\gamma}{255} \right)^{\gamma_B} + O_B, \end{aligned} \quad (5)$$

where α_i , γ_i , and O_i ($i = R, G, B$) are the gain, gamma, and offset. The gray balancing curves of a sample image are shown in Fig. 5. For this particular image, $\alpha_R = 95.69$, $\gamma_R = 2.00$, $O_R = 3.48$, $\alpha_G = 97.34$, $\gamma_G = 1.93$, $O_G = 3.05$, $\alpha_B = 99.99$, $\gamma_B = 1.98$, and $O_B = 3.37$.

After obtaining the linearized RGB values, we can continue to the regression step. To maximize calibration accuracy, the target patches are classified into three sets, and the calibration mapping is trained separately on each set. We refer to these three training sets as Set 1, Set 2, and Set 3, respectively. We use the following procedure for patch grouping. To form Set 1, we first compute the mean RGB values of the pixels within the skin segmentation mask described in the previous section. We refer to this mean as the centroid of Set 1. Then, we find the patches for which the Euclidean distance is less than 80 to the centroid of Set 1. Similarly, the patches in Sets 2 and 3 are grouped based on the centroids determined by the K-means algorithm. The centroids of these three training sets will be used again when classifying image pixels for the entire image, as discussed later. Finally, we compute three different transformation matrices T_i using polynomial regression. Let Q denote the $N_i \times 11$ matrix consisting of the polynomial terms of the linearized RGB values of the target patches.

$$Q_i = [R_{i,l} \quad G_{i,l} \quad B_{i,l} \quad R_{i,l}^2 \quad G_{i,l}^2 \quad B_{i,l}^2 \quad R_{i,l}G_{i,l} \quad G_{i,l}B_{i,l} \quad R_{i,l}B_{i,l} \quad R_{i,l}G_{i,l}B_{i,l} \quad 1], \quad (6)$$

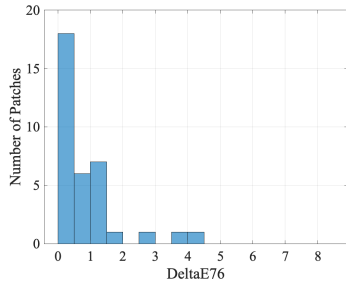
where N_i indicates the number of patches in the i -th set. Let P denote the $N_i \times 3$ matrix consisting of the measured CIE XYZ values

$$P_i = [X_i \quad Y_i \quad Z_i]. \quad (7)$$

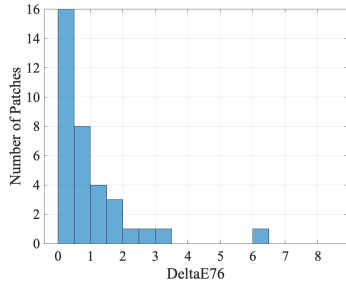
Then, the optimal 3×11 transformation matrix can be computed as

$$T_i = (Q_i^T Q_i)^{-1} Q_i^T P_i. \quad (8)$$

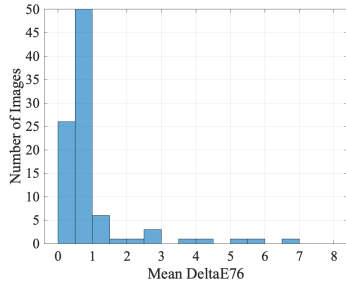
Now, we are ready to apply the transformation matrices to the entire image. As illustrated by Fig. 6, the image pixels will be classified into three sets based on their distance to the three pre-computed centroids, and then the linearized RGB values will be converted to CIE XYZ using the corresponding matrices.



(a)



(b)



(c)

Figure 7: (a) The histogram of the calibration $\Delta E76$ across all 35 patches for the skin with no foundation image of subject No. 1 and (b) subject No. 2. The mean calibration $\Delta E76$ values over all 35 patches for subject No. 1 and subject No. 2 are 0.75 and 0.91, respectively. (c) The histogram of the mean calibration $\Delta E76$ values across all images of all subjects.

Calibration Performance

The color correction stage is now completed, and we can evaluate the accuracy of it. To better perceive the color coordinates, they are converted from CIE XYZ to the 1976 CIE $L^*a^*b^*$ space. The conversion is defined as

$$\begin{aligned} L^* &= 116 \cdot f\left(\frac{Y}{Y_n}\right) - 16 \\ a^* &= 500 \cdot \left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right) \\ b^* &= 200 \cdot \left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right), \end{aligned} \quad (9)$$

where

$$f(t) = \begin{cases} \frac{1}{3} \left(\frac{116}{24} \right)^2 + \frac{16}{116}, & \text{if } x \leq \frac{24}{116}, \\ x^{\frac{1}{3}}, & \text{if } x > \frac{24}{116}. \end{cases} \quad (10)$$

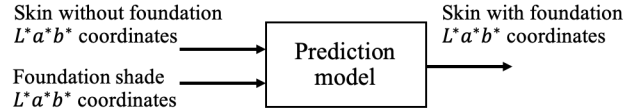


Figure 8: Illustration of the input and output of the prediction model.

Here, (X_n, Y_n, Z_n) are the CIE XYZ tristimulus values of the white point. With CIE standard D50 illumination, $X_n = 96.42, Y_n = 100, \text{ and } Z_n = 82.52$.

Then, the color difference between the ground truth and the calibrated CIE $L^*a^*b^*$ coordinates of each of the patches is computed. The color difference is defined as the Euclidean distance between the color coordinates (L_1^*, a_1^*, b_1^*) and (L_2^*, a_2^*, b_2^*) , such that

$$\Delta E76 = \sqrt{(L_1^* - L_2^*)^2 + (a_1^* - a_2^*)^2 + (b_1^* - b_2^*)^2}. \quad (11)$$

The resulting histograms for two sample images are shown in Figs. 7 (a) and (b). As can be seen from Figs. 7 (a) and (b), out of 35 patches, 33 patches have an $\Delta E76$ value less than 3 for both cases. So for these two images, color correction is quite effective. Figure 7 (c) shows the histogram of the mean $\Delta E76$ value over 35 patches for all images. It can be seen that the mean calibration error for the majority of the images is too small to be noticed, i.e. $\Delta E76 \leq 1$. This suggests that the proposed calibration algorithm works well on the dataset. The five images with a mean $\Delta E76$ value greater than 3 are considered to be outliers, and are therefore excluded from the subsequent analysis.

Experimental Results

With the outliers taken out, the remaining images consist of 63 skin with and without foundation pairs. These images are taken by 19 subjects. The distribution of the skin tone types is summarized in Table 1. After color correcting all the images using our

Skin Tone Type	Fair	Light	Medium	Tan	Dark
Number of Subjects	4	4	4	3	3
Number of Image Pairs	14	13	14	12	10

Table 1: Number of subjects and number of images in each skin tone category.

proposed method, we obtain 63 pairs of CIE $L^*a^*b^*$ coordinates. Apart from the selfie photos of the subjects, we also collect an image that contains the swatches of all the foundation shades. The foundation shades are applied on a white cardboard using makeup sponges. The original foundation swatches image is shown in Fig. 9. The same color correction procedure is performed on the image, except that the three sets are now all determined by K-means. The mean calibration $\Delta E76$ of this foundation swatches image is 0.34.

Now, we can develop a model to predict the CIE $L^*a^*b^*$ coordinates of the skin-with-foundation color, given the CIE $L^*a^*b^*$ coordinates of a skin-with-no-foundation color and the CIE $L^*a^*b^*$ coordinates of a foundation color, as illustrated in Fig. 8. We explored two common machine learning models. They are linear regression and SVR [16] with a linear kernel.

We denote the 6-dimensional input vector (i.e., the CIE $L^*a^*b^*$ coordinates of a skin-with-no-foundation color and the



Figure 9: The original foundation swatches image. From top to bottom and from left to right, the shades are No. 100, 110, 120, 130, 140, 150, No. 200, 210, 220, 230, 240, 250, 300, 310,, 540, 550.

CIE $L^*a^*b^*$ coordinates of a foundation color) of the i -th sample as x_i and the corresponding 3-dimensional ground truth label (i.e., the CIE $L^*a^*b^*$ coordinates of the skin-with-foundation color) as y_i . Then, the linear regression problem can be formulated as

$$J(\theta) = \sum_{i=1}^N \mathcal{L}(\theta^T x_i, y_i) = \|\theta^T x_i - y_i\|^2, \quad (12)$$

where $\theta = [w, b]^T$, $\theta^T x_i$ is the prediction made by the model, and N is the number of samples in the dataset. We further define a $N \times 7$ matrix

$$A = \begin{pmatrix} x_1^T & 1 \\ x_2^T & 1 \\ \dots & \dots \\ x_N^T & 1 \end{pmatrix}, \quad (13)$$

and a $N \times 3$ matrix

$$Y = \begin{pmatrix} y_1^T \\ y_2^T \\ \dots \\ y_N^T \end{pmatrix}. \quad (14)$$

Then, (12) can be converted to the vector form such that $J(\theta) = \|A\theta - Y\|^2$. Thus, the optimized solution can be expressed as

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} J(\theta) = (A^T A)^{-1} A^T Y. \quad (15)$$

The SVR algorithm is based on the same concept as the support vector machine, except that it uses the support vectors for soft margins in the regression process rather than classification. The regression in SVR can be formulated as:

$$\underset{w, b}{\operatorname{argmin}} \frac{1}{2} \|w\|^2 + C \sum_i^N (\xi_i + \xi_i^*), \quad (16)$$

subject to

$$\begin{aligned} y_i - wx_i - b &\leq \varepsilon + \xi_i \\ wx_i + b - y_i &\leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* &> 0, \end{aligned} \quad (17)$$

where C is the box constraint, ε is the margin around the decision boundary, and ξ_i, ξ_i^* are the slack variables. With a linear kernel, the prediction function can be written as:

$$f(x) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) \langle x_i, x \rangle + b, \quad (18)$$

where α_i and α_i^* are the Lagrange multipliers. The Python package *Scikit-learn* is used to implement linear regression and SVR.

A common way to evaluate the performance of a regression model is to use K -fold cross-validation. In K -fold cross-validation, the dataset is randomly split into K equal-sized folds. We train the model on the data in $K - 1$ of the folds and evaluate the model on the remaining one fold, namely, the validation fold. We then repeat this process K times. Given the fact that we have limited data in the dataset, we choose $K = N$, where N is the number of all data points. This method is referred to as the leave-one-out cross-validation (LOOCV) method. That is to say, in each trial, the predictor is trained on all but one data point, and the prediction is made for that excluded point. Then the performance can be computed as the average over the N trials. The advantage of using LOOCV is that each data point gets the chance to be allocated into both the testing set and $N - 1$ of the training sets. The selection bias is therefore decreased.

To determine the goodness of model fit, the coefficient of determination, denoted by R^2 , is computed. In the context of regression, it is a measure of the proportion of the prediction error that can be attributed to the variance in the independent input variables. It is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2}, \quad (19)$$

where $y_i - \hat{y}_i$ is the error from the prediction \hat{y}_i to the ground truth y_i of the i -th data point, and \bar{y} is the mean of the ground truth y values. The R^2 score is in the range of $[0, 1]$. A score of 0 means that the dependent variable cannot be predicted from the independent variable, and a score of 1 means the dependent variable can be predicted with no error from the independent variable.

Besides, the mean squared error (MSE) and mean absolute error (MAE) are also used to evaluate the accuracy of prediction results. The MSE can be expressed as

$$MSE = \frac{\sum_{i=1}^n (p_i - g_i)^2}{n}, \quad (20)$$

where p_i and g_i are the predicted value and the ground truth value of the i -th sample point, respectively. Similarly,

$$MAE = \frac{\sum_{i=1}^n |p_i - g_i|}{n}. \quad (21)$$

Table 2 summarizes the LOOCV results of the linear regression model and the SVR model in terms of R^2 , average MSE, and average MAE. It can be seen that the average MSE and MAE values are less than $1.5 \Delta E$ and the R^2 value is high for both of the models. This implies that the prediction models can accurately predict the skin with foundation color on the dataset.

Model	R^2	Average MSE	Average MAE
Linear Regression	0.83	1.50	0.91
SVR with a Linear Kernel	0.82	1.49	0.87

Table 2: LOOCV R^2 , MSE, and MAE results.

Conclusion

The selfie images are calibrated using a subset of color checker patches. The pixels are classified into three sets according to the Euclidean distance from the RGB values of the pixel to the three designated centroids. Three different transformation matrices are computed separately and then applied to the corresponding pixels in the image. The calibration accuracy is measured by the color difference ΔE between the reference value and the calibrated value in CIE $L^*a^*b^*$ space. The ΔE results indicate that the error produced by the proposed method is almost not distinguishable for most of the images. A prediction model is then built upon the calibrated selfie images. The prediction performance is measured by R^2 , MSE , and MAE . LOOCV results show that the prediction made by both linear regression and SVR with a linear kernel is reliable.

References

- [1] N. Bhatti, H. Baker, H. Chao, S. Clearwater, M. Harville, J. Jain, N. Lyons, J. Marguier, J. Schettino, and S. Süssstrunk, "Mobile cosmetics advisor: an imaging based mobile service," in *Proceedings of SPIE*. International Society for Optics and Photonics, 2010, vol. 7542.
- [2] Wai-Shun Tong, Chi-Keung Tang, Michael S Brown, and Ying-Qing Xu, "Example-based cosmetic transfer," in *15th Pacific Conference on Computer Graphics and Applications (PG'07)*. IEEE, 2007, pp. 211–218.
- [3] Tam V Nguyen and Luoqi Liu, "Smart mirror: Intelligent makeup recommendation and synthesis," in *Proceedings of the 25th ACM International Conference on Multimedia*, 2017, pp. 1253–1254.
- [4] Taleb Alashkar, Songyao Jiang, Shuyang Wang, and Yun Fu, "Examples-rules guided deep neural network for makeup recommendation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017, vol. 31.
- [5] Jianshu Li, Chao Xiong, Luoqi Liu, Xiangbo Shu, and Shuicheng Yan, "Deep face beautification," in *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015, pp. 793–794.
- [6] Dongliang Cheng, Dilip K Prasad, and Michael S Brown, "Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution," *JOSA A*, vol. 31, no. 5, pp. 1049–1058, 2014.
- [7] Jonathan T Barron and Yun-Ta Tsai, "Fast Fourier color constancy," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 886–894.
- [8] Yuanming Hu, Baoyuan Wang, and Stephen Lin, "FC4: Fully convolutional color constancy with confidence-weighted pooling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4085–4094.
- [9] P.-C. Hung, "Colorimetric calibration in electronic imaging devices using a look-up-table model and interpolations," *Journal of Electronic Imaging*, vol. 2, no. 1, pp. 53–62, 1993.
- [10] C. Zhao, J. Niu, G. Li, H. Wang, and C. He, "Facial color management for mobile health in the wild," *IEEE Transactions on NanoBioscience*, vol. 15, no. 4, pp. 316–327, 2016.
- [11] V. Cheung and S. Westland, "Color camera characterisation using artificial neural networks," in *10th Color Imaging Conference: Color Science and Engineering Systems, Technologies, Applications*. Society for Imaging Science and Technology, 2002, vol. 2002, pp. 117–120.
- [12] H. R. Kang and P. G. Anderson, "Neural network applications to the color scanner and printer calibrations," *Journal of Electronic Imaging*, vol. 1, no. 2, pp. 125–136, 1992.
- [13] R. S. Berns and M. J. Shyu, "Colorimetric characterization of a desktop drum scanner using a spectral model," *Journal of Electronic Imaging*, vol. 4, no. 4, pp. 360–373, 1995.
- [14] G. D. Finlayson and M. S. Drew, "Constrained least-squares regression in color spaces," *Journal of Electronic Imaging*, vol. 6, no. 4, pp. 484–494, 1997.
- [15] S. Gindi, "Color characterization and modeling of a scanner," M.S. thesis, Dept. Elect. Comput. Eng., Purdue Univ., West Lafayette, IN, USA, Aug. 2008.
- [16] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in Neural Information Processing Systems*, vol. 9, pp. 155–161, 1996.
- [17] N. A. A. Rahman, K. C. Wei, and J. See, "RGB-H-CbCr skin colour model for human face detection," *Proceedings of MMU International Symposium on Information & Communications Technologies*, 2006.

Author Biography

Yafei Mao received her BS in Electrical and Computer Engineering from Purdue University (2016). Since then, she has been working on her PhD in Electrical and Computer Engineering at Purdue University. Her research interests are halftoning, image processing, and computer vision.