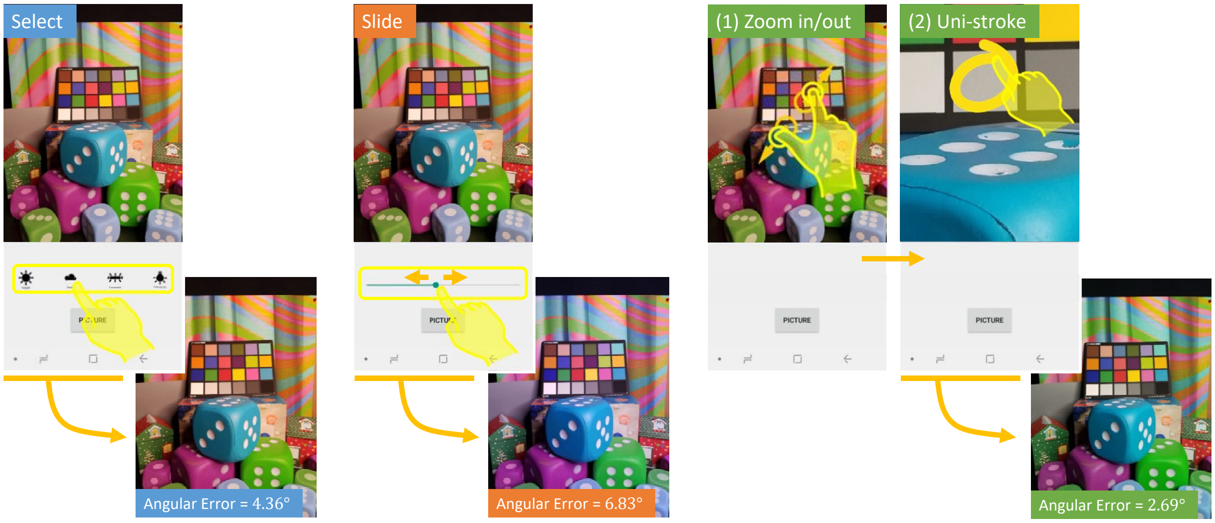
MarkWhite: An Improved Interactive White-Balance Method for Smartphone Cameras

Abdelrahman Abdelhamed* I. Scott MacKenzie† Michael S. Brown‡

Department of Electrical Engineering and Computer Science, York University



(a) Predefined Illuminants **(b) Temperature Slider** **(c) MarkWhite**

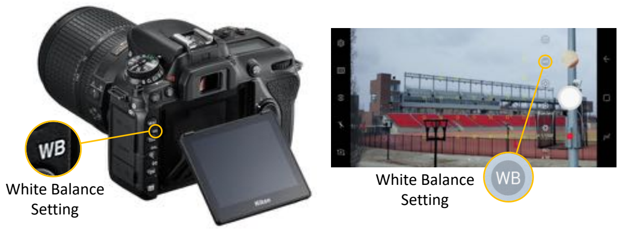
Figure 1: Comparison of three interactive white balance methods. The proposed method, MarkWhite (c), results in more accurate image colours and lower angular errors than the existing methods (a) and (b).

Abstract

White balance is an essential step for camera colour processing. The goal is to correct the colour cast caused by scene illumination in captured images. In this paper, three user-interactive white balance methods for smartphone cameras are implemented and evaluated. Two methods are commonly used in smartphone cameras: predefined illuminants and temperature slider. The third method, called *MarkWhite*, is newly introduced into smartphone camera apps. Two user studies evaluated the accuracy and task completion time of MarkWhite and compared it to the existing methods. The first user study revealed that a basic version of MarkWhite is more accurate, slightly faster, and slightly more preferred over the two existing methods. The main user study focused on the full version of MarkWhite, revealing that it is even more accurate than the basic version and better than state-of-the-art industrial white balance methods on the latest smartphone cameras. The collective findings show that MarkWhite is a more accurate and efficient user-interactive white balance method for smartphone cameras, and more preferred by users as well.

Keywords: Mobile user interfaces, user-interactive white balance, smartphone camera, MarkWhite

Index Terms: Human-centered computing: Gestural input; Smartphones; Empirical studies in ubiquitous and mobile computing



(a) DSLR (b) Smartphone

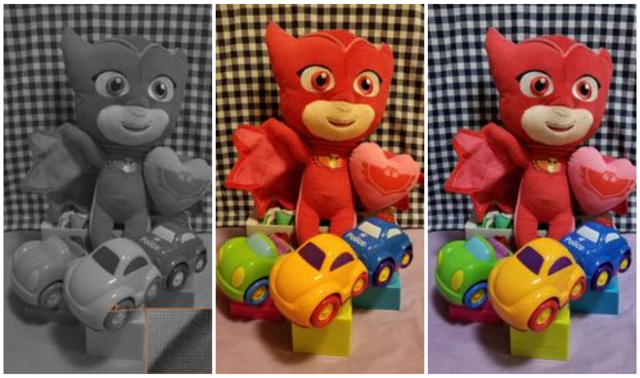
Figure 2: Example user interface design of (a) Nikon D7500 DSLR camera [20] vs. (b) Samsung Galaxy S8 smartphone camera app [22]. White balance features are indicated in the zoomed-in regions.

1 Introduction

Smartphones are now one of the most prevalent human-interactive devices, with 1.5 billion units sold in 2017 [11]. Smartphones include touch-screen user interfaces, built-in cameras, and pre-installed camera apps. The user interface for a smartphone camera app is significantly different from a conventional camera, such as a digital single-lens reflex (DSLR) camera. In a typical DSLR camera, shown in Figure 2a, there are physical buttons and switches to control the camera settings. For smartphone camera apps, however, buttons and switches are implemented on the small touch-sensitive display. See Figure 2b.

One of the main camera settings is the *white balance (WB)*, featured in Figure 2. White balance is an essential step in a camera's colour processing pipeline. It aims to minimize the colour cast caused by the scene light source (also referred to as *illuminant*) while capturing images or videos. Figure 3a shows an example of a scene imaged under a fluorescent illuminant. Shown is the raw-

*e-mail: kamel@eecs.yorku.ca
†e-mail: mack@eecs.yorku.ca
‡e-mail: mbrown@eecs.yorku.ca



(a) Raw-RGB (b) Incorrect WB (c) Correct WB

Figure 3: An example of white balancing under fluorescent illuminant. The raw-RGB in (a) is white balanced with (b) incorrect illuminant (daylight) and (c) correct illuminant (fluorescent).

RGB image which is the immediate output of camera sensor without colour processing. The raw-RGB image undergoes many stages in the camera's onboard imaging pipeline to be converted into a final standard sRGB colour space that is tailored for human perception on device displays (full details are beyond the scope of this paper; interested readers are referred to [17]). White balancing is an essential stage. The image in Figure 3a is white balanced based on the assumption of daylight and fluorescent illuminants in Figures 3b and 3c, respectively. Indeed, the correct assumption of fluorescent illuminant yields perceptually more accurate colours. This example emphasizes the importance of white balance in processing images through camera pipelines.

Due to the importance of the WB process, many algorithms have been proposed for automatic white balance (AWB) of images (e.g., [3, 4, 9, 10, 15]); however, cameras and apps also provide *user-interactive* ways for WB. For example, some camera apps provide a set of predefined illuminants for the user to select as the scene illuminant (Figure 1a). In addition, some apps provide a fine-scale slider to select the temperature (in Kelvin) of the scene illuminant (Figure 1b). This is more common for professional photographers or users who want more control of the image capturing process.

The goal of this paper is not to solve AWB. Rather, we approach WB in an interactive, efficient way, we call it *MarkWhite*. The method, shown in Figure 1c, is inspired by a common practice followed by photographers to white balance images. This involves measuring the reflectance of scene illuminant over a gray surface, then correcting the camera's colours based on the measured illuminant. This is found only in DSLR cameras and involves the tedious steps of (1) capturing a gray card, (2) adjusting the camera settings to use the captured image for white balance calibration, (3) then imaging the scene. In MarkWhite, with the aid of Camera2 API [13], this process is bundled into one step: When the user marks a gray region in the camera preview, the scene illuminant is measured and instantaneously fed back to the camera for white balance calibration and capturing the image. This described interactive technique itself is not novel, however, the contribution is in the integration of this technique in the image capturing process on smartphone camera apps. Such integration does not require any post-capture photo editing, where existing methods use the same technique in a *post-capture* or a post-processing manner and require saving the raw-RGB image and then reloading it in a photo editing software (e.g., Boyadzhiev et al. [4], Lightroom, and PhotoShop).

Despite the existence of user-interactive white balance methods, there is no prior work studying such interfaces to assess their efficiency and ease of use. To this end, we focus on comparing and evaluating such *interactive* methods of white balance. All methods are implemented as smartphone camera apps. The evaluation is through user studies that compare the interfaces in terms of quantitative and qualitative measures.

2 Related Work

Despite the existance of many automatic white balance algorithms (e.g., [3, 4, 9, 10, 15]), there is little work on the user-interactive alternatives. In this section, some interaction-aided white balance methods are discussed. Then, an evaluation scheme for assessing the accuracy of white balance methods is reviewed. Also, a simple discussion of the white balance interaction styles is presented.

2.1 User Interaction for White Balance

User interaction has been used to help with white balancing images, for example, to distinguish between multiple illuminants in the same scene [4]. The user is asked to mark objects of neutral colour (i.e., white or gray) in an image, and regions that look fine after the standard white balance. Then, to overcome any colour variations, the user marks which regions should have a constant colour. With the help of user intervention, this produces images free of colour cast. In another line of work, user preference has guided the white balance correction of two-illuminant (e.g., indoor and outdoor) images [8] based on the finding that users prefer images that are white balanced using a mixture of the two illuminants, with a higher weight for the outdoor (e.g., daylight) illuminant. A typical weighting of 75% − 25% of the outdoor and indoor illuminants, respectively, was shown to work best in most two-illuminant cases.

The aforementioned works mainly focus on having more than one illuminant in the scene, which usually requires time-consuming post-processing of images. On the other hand, user-interactive white balance methods typically deal with single-illuminant cases in real-time and try to correct image colours during the image capturing process. All interactive WB methods evaluated in this paper are on-camera, real-time, and deal with the single-illuminant case.

2.2 White Balance Evaluation and Angular Error

To measure the accuracy of white balance, ground truth illuminant colours are needed. A perfectly white balanced image ("ground truth") is obtained by placing a colour rendition chart (e.g., an X-Rite ColorChecker® [25]) in the scene; capturing a raw-RGB image; and manually measuring the scene illuminant reflected by the achromatic (gray/white) patches of the chart. The scene illuminant (referred to as "ground truth" illuminant) is denoted as $I_{\mathrm{gt}} = [r_{\mathrm{gt}}, g_{\mathrm{gt}}, b_{\mathrm{gt}}]$. This illuminant colour is used to perfectly white balance the raw-RGB image. One simple method for applying white balance on an image is the von-Kries diagonal model [23], or the diagonal model for short. This model maps an image $\mathbf{X}$ taken under one illuminant (e.g., I_{gt}) to another (e.g., a canonical illuminant $I_{\mathrm{c}} = [1, 1, 1]$) by simply scaling each channel independently:

$$I_{\mathrm{c}}(\mathbf{X}) = \mathrm{diag}\left(\frac{I_{\mathrm{c}}}{I_{\mathrm{gt}}}\right) I_{\mathrm{gt}}(\mathbf{X}), \quad (1)$$

where diag$(\cdot)$ indicates the operator of creating a diagonal matrix from a vector and the image $\mathbf{X}$ would be reshaped as a 3-by-N matrix, where N is the number of pixels and each column represents the RGB values of one pixel.

Afterwards, the accuracy of white balance is calculated by measuring the *angular error* [9] between the ground truth illuminant and the illuminant adjusted by the user. Assuming the ground truth illuminant is $I_{\mathrm{gt}} = [r_{\mathrm{gt}}, g_{\mathrm{gt}}, b_{\mathrm{gt}}]^T$ and the user-selected illuminant is $I_{\mathrm{u}} = [r_{\mathrm{u}}, g_{\mathrm{u}}, b_{\mathrm{u}}]^T$, the angular error E_{ang} (in degrees) between the two illuminats would be

$$E_{\mathrm{ang}} = \frac{180^\circ}{\pi} \cos^{-1}\left(\frac{I_{\mathrm{gt}} \cdot I_{\mathrm{u}}}{\|I_{\mathrm{gt}}\| \|I_{\mathrm{u}}\|}\right), \quad (2)$$

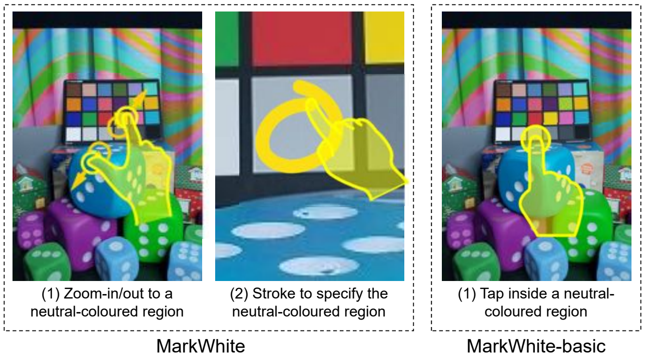

(1) Zoom-in/out to a neutral-coloured region
(2) Stroke to specify the neutral-coloured region
(1) Tap inside a neutral-coloured region
MarkWhite
MarkWhite-basic

Figure 4: The MarkWhite vs the MarkWhite-basic method.

where T is a transpose operator, $(\cdot)$ is dot product, and $\|\cdot\|$ is vector norm (for more details on white balance correction; interested readers are referred to [1, 7]). The angular error is used as the main dependent variable in the user studies (more details in Section 3).

2.3 Interaction Styles for White Balance

The three white balance methods evaluated in this paper are categorized in terms of interaction style. Method 1, the predefined illuminants, is a form of categorical scale. Method 2, the temperature slider, is a form of fine-scale selection (or simply, a slider). Method 3, MarkWhite, involves target selection, where the user selects or marks a white/gray object on the camera preview using pinch-to-zoom [2, 18] and uni-stroke drawing gestures [6, 12, 16]. One study [21] found no statistically significant differences between categorical responses and sliders. However, another study [5] suggested more weaknesses than strengths for using slider scales on smartphones. This study also suggested that preference for such touch-centric interfaces varies across devices and may not be as highly preferred as traditional categorical response interfaces.

Of course, a slider scale is attractive and user-friendly, compared to a categorical scale. However, a categorical scale is more functional and efficient. While a slider gives the user more control, it may take more to decide on the response. On the other hand, choosing a common categorical response may save time. In the next section, another approach is proposed for comparing such interfaces for the white balance problem.

3 Pilot Study: MarkWhite-basic

To compare the three user-interactive white balance methods, a pilot user study was conducted. The goal was to compare the interfaces in terms of quantitative measures, user preference, and ease of use. To ensure a fair comparison, we included a simple version of MarkWhite, namely, MarkWhite-basic. MarkWhite-basic only includes the essential feature of tapping on a reference object to adjust white balance. It does not include zooming or drawing. The difference between MarkWhite and MarkWhite-basic is illustrated in Figure 4.

3.1 Participants

Twelve participants were selected using convenience sampling [19, p. 172] from a pool of staff and students at the local university campus (9 males, 3 females). The average age was 33.3 years (min 23, max 45). All participants were comfortable using smartphone cameras and had no issues with colour vision.

3.2 Apparatus

3.2.1 Smartphone and Camera

The user interfaces were run on a Samsung Galaxy S8 smartphone [22] (Figure 2b), with Google's Android Nougat (7.0). The device has a 5.8" display with resolution of 2960 × 1440 pixels and density of 570 pixels/inch (ppi). Only the rear camera was used for the user study, which has 12 mega pixels, sensor size of 1/2.55", and F1.7 aperture. It also has the capability of saving raw-RGB images, which is necessary for the camera app.

3.2.2 Camera App

A smartphone camera app was developed with the following features: (1) adjusting white balance by setting the scene illuminant using any of the three interactive methods shown in Figures 1a - 1c; (2) capturing images in both sRGB and raw-RGB formats; (3) providing a viewfinder that shows the user live updates on the scene after any adjustment to help them decide when to capture an image. Additionally, the same app collected and logged user data, performance measurements, and questionnaire answers (more details in the Design subsection). The implementation of the app was mainly based on Google Camera2 API [13] and the Camera2Raw [14] code sample from Google Samples.

The predefined illuminants and temperature slider methods were implemented using direct calls to Camera2 APIs and an algorithm to convert colour temperatures to colour primaries (red, green, and blue) [24]. For MarkWhite, a new technique was followed that involves capturing two raw-RGB images. First, the auto white balance algorithm on the camera is turned off. When the user taps on a gray/white region in the camera viewfinder, the first raw-RGB image is captured and the region selected by the user is extracted from the image. The extracted region is used to calculate the illuminant colour which is fed back to the camera, using the Camera2 APIs, to set the adjusted white balance parameters. Finally, the second image, that is now white-balanced, is captured. This procedure is illustrated in Figure 5. This process runs in real time and does not introduces any time delay in the image capturing process.

3.3 Procedure

Before starting the user study, the problem of white balance was briefly introduced to participants. Then, they were informed how to use the camera app. The three interfaces were shown in operation once by the experimenter. The participants were shown how to use each interface to adjust the WB of the scene and capture the image. Before starting, participants were given the chance to practice once using all three WB interfaces under one test condition.

Once started the experiment, participants were asked to adjust the white balance of an image using all three interfaces under six test conditions (discussed in the Design subsection). At the end, they completed a questionnaire asking them to rank the three interfaces based no their preference and the ease of use. Figure 6 shows two participants performing the experiment tasks. The scene content and the adjustable light source can be seen. All scenes included a colour chart and a gray card to help with selecting a neutral material. The experiment was carried out in a dark room with only one light source to control lighting conditions. The light source was DC-powered to minimize light flicker effects. It also has features to simulate a wide range of illuminant temperatures. Participants' data and performance measures were logged and later analysed on a personal computer. To counterbalance the order effect of the three WB control methods, the participants were divided into six groups with each group using the methods in a different order. This procedure covers all possible orders of the three methods $(3! = 6)$.

3.4 Design

3.4.1 Independent Variables

The pilot study was a $3 \times 3 \times 2$ within-subjects design with the following independent variables:

Control Method. This independent variable indicates the type of user interaction method used to adjust the white balance of the image before capturing it. See Figures 1a - 1c. This variable has three levels:

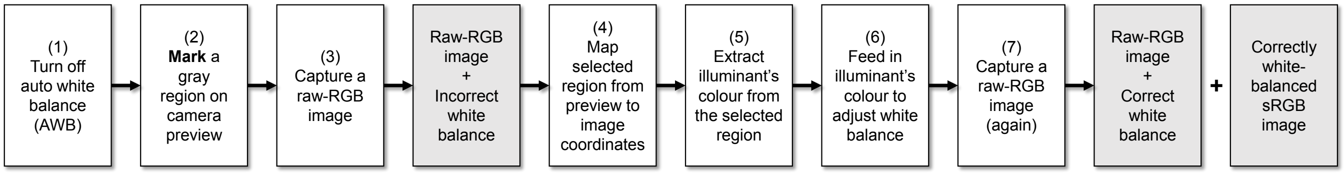
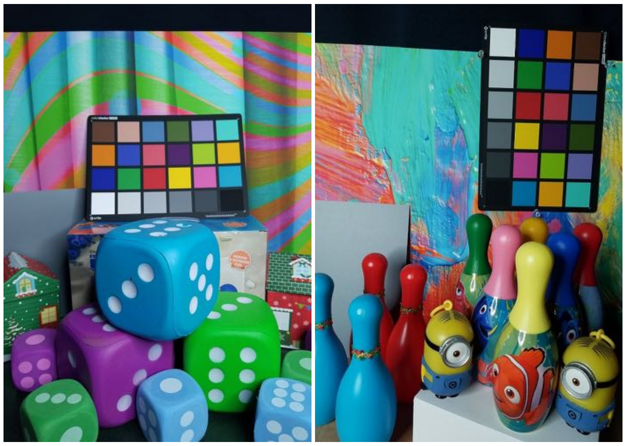

(1) Turn off auto white balance (AWB)
(2) **Mark** a gray region on camera preview
(3) Capture a raw-RGB image
Raw-RGB image + Incorrect white balance
(4) Map selected region from preview to image coordinates
(5) Extract illuminant's colour from the selected region
(6) Feed in illuminant's colour to adjust white balance
(7) Capture a raw-RGB image (again)
Raw-RGB image + Correct white balance
+
Correctly white-balanced sRGB image

Figure 5: A step-by-step procedure of the MarkWhite-basic method. These steps mainly depend on the Camera2 APIs [13].

Figure 6: Two participants performing the experiment tasks. The adjustable DC-powered light source can be seen.



(a) Scene 1

(b) Scene 2

Figure 7: The two scenes used in the user study as test conditions for the third independent variable (scene content).

- *Predefined illuminants* – the user selects the scene illuminant from a predefined set (e.g., daylight, cloudy, fluorescent, and incandescent).
- *Temperature slider* – the user adjusts the scene illuminant by sliding a bar representing the illuminant's temperature.
- *MarkWhite-basic* – the user taps on the display to select a region of neutral colour (i.e., gray) to set the scene illuminant. In MarkWhite-basic, zooming and drawing features are not allowed. See Figure 4.

Scene Illuminant. This independent variable represents the type of light source under which the user is imaging a scene. The levels are daylight, fluorescent, and incandescent.

Scene Content. This independent variable corresponds to the objects being imaged (e.g., natural scene, building, person, etc.). For easier reproduction of the scenes, only two indoor scenes were considered (shown in Figure 7); due to outdoor scenes and illuminants being harder to reproduce exactly.

With the above design, the user study consisted of 18 within-subjects assignments for each participant: 3 white balance control methods × 3 scene illuminants × 2 scene contents. With 12 participants, the total number of assignments was $12 \times 18 = 216$.

3.4.2 Dependent Variables

Three dependent variables were examined: angular error (degrees), task completion time (seconds), and user experience (i.e., user preference and ease of use).

Angular Error – the angle (in degrees) between the illuminant selected by the user to adjust white balance and a ground truth illuminant extracted from the raw-RGB image, as discussed earlier in section 2 [9]. The lower the angular error, the higher the accuracy of image colours. Therefore, angular error indicates how accurately the user adjusted the white balance of the image before capturing it.

Task Completion Time – the time (in seconds) the user takes to adjust the white balance and capture an image.

User Experience – the user's experience with the three interfaces. This was captured using a questionnaire. Participants ranked the interfaces based on preference and ease of use.

4 Results and Discussion (Pilot Study)

Post-study analyses resulted in interesting findings across the three white balance methods regarding quantitative and qualitative measures. Both are discussed next.

4.1 Angular Error

The angular error (in degrees) is the main dependent variable and represents the accuracy of white balance in images; the lower the angular error, the higher the accuracy. Over all 216 assignments, the grand mean for angular error was 8.23°.

4.1.1 Control Method

By control method, angular errors were 10.9° for predefined illuminants, 8.07° for temperature slider, and 5.68° for MarkWhite-basic. See Figure 8a. MarkWhite-basic had 48.1% and 29.7% lower angular errors than predefined illuminants and temperature slider, respectively. The differences were statistically significant ($F_{2,11} = 34.44,\ p < .0001$). A post hoc Scheffé test revealed that all three pairwise differences were significant. This finding indicates that MarkWhite-basic is a more accurate white balance control method than the two existing methods. It also indicates that temperature slider is more accurate than predefined illuminants.

It was no surprise that MarkWhite-basic was the most accurate among the three methods, as the user directly taps on a white/gray region in the image that reflects the illuminant. On the other hand, for predefined illuminants and temperature slider, the user iteratively adjusts the illuminant and observes the image more than once before capturing the image. However, the success of MarkWhite-basic depends on having a gray/white object in the scene. Despite the accurate results, there is no mechanism to make selection easier if the target is small or far from the camera. Also, there is no mechanism to select an arbitrary shape as a target area, instead of tapping on a single point. Such limitations are addressed in the full version of the method (MarkWhite).

4.1.2 Scene Illuminant

The effect of the second independent variable, scene illuminant, on angular error, was also investigated. The angular errors by scene illuminant were 12.3° for incandescent, 4.53° for fluorescent, and 7.89° for daylight. See Figure 8b. The differences were statistically

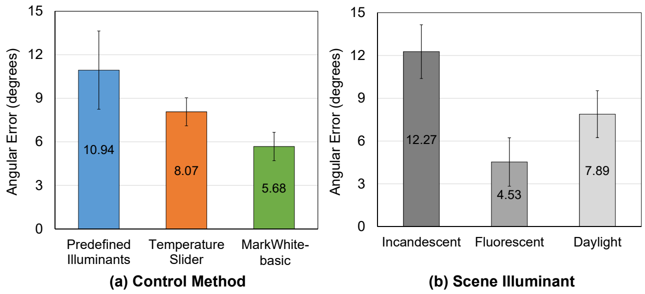


Figure 8: Angular error (degrees) by (a) white balance method and (b) scene illuminant. Error bars show ±1 *SD*.

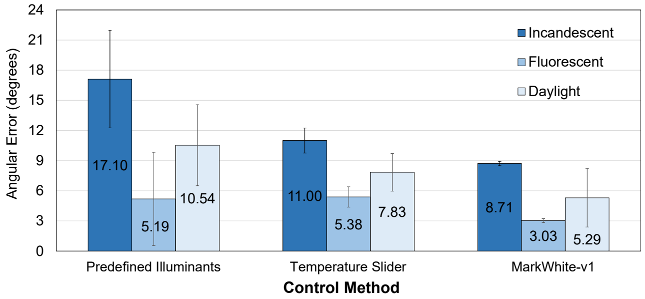


Figure 9: Angular error (degrees) by control method and scene illuminant. Error bars show ±1 *SD*.

significant ($F_{2,11} = 69.13,\ p < .0001$). A post hoc Scheffé test revealed that all three pair-wise differences were significant. This finding may indicate that users' ability to adjust the white balance of images using any of the three methods is more accurate under fluorescent light sources, followed by daylight then incandescent illuminants.

4.1.3 Interaction Effects

For further analysis, the interaction effects between the two independent variables, control method and scene illuminant, are reported. The results for angular error by control method and scene illuminant are shown in Figure 9. The differences were statistically significant ($F_{2,2} = 3.89,\ p < .05$). Additionally, a post hoc Scheffé test between all pairwise conditions of the two independent variables – total of $\binom{9}{2} = 36$ comparisons – was analysed. Out of the 36 pairwise comparisons, 18 were statistically significant. These results confirm the findings regarding the main effects of control method and scene illuminant on angular error.

4.2 Task Completion Time

Task completion time, the second dependent variable, represents the time (in seconds) a user takes to adjust the white balance before capturing an image. Over all 216 assignments, the grand mean for task completion time was 18.3 seconds.

4.2.1 Control Method

By control method, task completion times were 19.1 s for predefined illuminants, 21.8 s for temperature slider, and 14.1 s for MarkWhite-basic. See Figure 10a. MarkWhite-basic was 26% and 35% faster than predefined illuminants and temperature slider, respectively. However, the differences were statistically not significant ($F_{2,11} = 3.24, p > .05$). This is likely due to variability in the time users spent deciding on the correct illuminant; it was noticed that some users take noticeably longer before they decide to capture an image, while others decide faster.

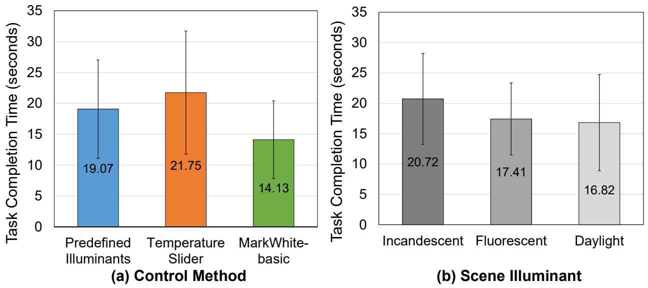


Figure 10: Task completion time (seconds) by (a) white balance control method and (b) scene illuminant. Error bars represent ±1 *SD*.

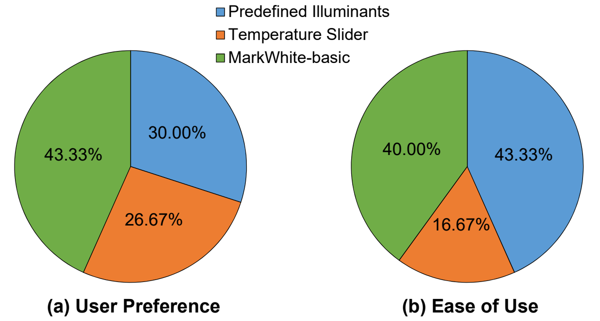


Figure 11: Questionnaire results: (a) user preference and (b) ease of use, as percentages per control method. The percentages are of the overall ratings on a 3-point scale.

4.2.2 Scene Illuminant

The task completion times by scene illuminant were 20.7 s for incandescent, 17.4 s for fluorescent, and 16.8 s for daylight. See Figure 10b. These differences were again not statistically significant ($F_{2,11} = 2.71, p > .05$). Since the main effects for control method and scene illuminant were not significant, no further analysis was needed to examine the interaction effects.

4.3 User Experience

After the experiment, participants answered a questionnaire asking them to rate the three WB methods on preference and ease of use. The responses were extracted as a 3-point rating scale. For user preference, ratings 1 and 3 indicated the least and the most preferred method, respectively. For ease of use, ratings 1 and 3 indicated the hardest and the easiest method to use, respectively. Results for both variables are discussed next.

4.3.1 User Preference

On user preference, the average ratings out of 3 were 1.8 (30% of overall rating points) for predefined illuminants, 1.6 (27%) for temperature slider, and 2.6 (43%) for MarkWhite-basic. See Figure 11a. By applying a Friedman test, the differences were not statistically significant ($\chi^2 = 5.60, p > .05, df = 2$). However, the p-value was low ($p = 0.061$), suggesting that the differences are not a result of randomness. Post hoc pairwise comparisons revealed a significant difference between the temperature slider and MarkWhite-basic. Thus, MarkWhite-basic is preferred over temperature slider. On the other hand, there was no clear preference between predefined illuminants and both temperature slider and MarkWhite-basic.

4.3.2 Ease of Use

On ease of use, the average ratings out of 3 were 2.6 (43% of overall rating points) for predefined illuminants, 1.0 (17% of overall rating points) for temperature slider, and 2.4 (40% of overall rating points) for MarkWhite-basic. See Figure 11b. By applying a Friedman test, the differences in ease of use were statistically significant (χ^2 = 15.20, $p < .001$, df = 2). Post hoc pairwise comparisons indicated significant differences between (1) predefined illuminants and temperature slider and (2) temperature slider and MarkWhite-basic. Thus, predefined illuminants and MarkWhite-basic are both easier to use than the temperature slider, even though temperature slider provides more fine-scale control. This is likely because the control given by the temperature slider makes it harder to decide on the correct illuminant. This was noted by some users during the experiment. On the other hand, between the predefined illuminants and MarkWhite-basic, it is inconclusive which is easier.

4.4 Group Effect

Concerning possible group effects, an ANOVA indicated no statistical significant for angular error ($F_{5,6} = 0.83$, ns) or task completion time ($F_{5,6} = 1.73$, ns). Thus, counterbalancing successfully minimized order effects.

5 Conclusion of Pilot Study

The results of the pilot study revealed that MarkWhite-basic produces the most accurate white-balanced images. MarkWhite-basic was also preferred over two existing methods. Although MarkWhite-basic required less task completion time, such finding was not statistically significant. Also, MarkWhite-basic came second in terms of ease of use, following the predefined illuminants.

Despite MarkWhite-basic being a more accurate white balance method, it has limitations, for example, when the target is small or far from the camera. In such cases, perhaps zooming in to the target would help. Another possibility is marking a region of the reference target, instead of tapping a small point. This might make the white balance more accurate, since more pixels are used to measure the illuminant colour. These features are implemented in the full version of MarkWhite and evaluated in the second user study.

6 Second User Study: MarkWhite

This second user study focused on MarkWhite which extends MarkWhite-basic with the following features:

- Pinch to zoom in or out of the camera preview. This facilitates selecting neutrally-coloured targets that are small or far away.
- Single stroke on the camera preview to select a region, such as a circle, for white balance calibration. It is expected that the larger the region, the more accurate the calibration, since more pixels are used to estimate the scene illuminant. The difference between MarkWhite-basic and MarkWhite was illustrated in Figure 4.

The second user study was similar to the pilot study. The main comparison was between MarkWite-basic and MarkWhite, since the pilot study showed that MarkWhite-basic is already more accurate than predefined illuminants and temperature slider. However, a baseline condition was also included, described next.

6.1 Evaluation Against Automatic White Balance

AWB was included as a baseline condition as it represents typical usage on smartphones. Of course, AWB is not interactive; it runs an efficient algorithm to *automatically* estimate the illuminant colour and white balance the images. Such on-board algorithms are proprietary; however, since the camera used in this study is recent (Galaxy S8, released in 2017), it likely uses a state-of-the-art AWB algorithm.

The task completion time for AWB is zero, since there isn't any user interaction. It is worth noting that the goal including AWB as a baseline is to see if MarkWhite can improve white balance accuracy beyond the state-of-the-art industrial AWB methods. Investigating the whole literature of AWB is beyond the scope of this paper.

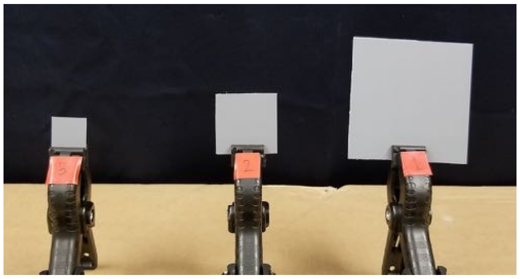


Figure 12: Gray targets used in the second user study. Sizes are 2 × 2, 4 × 4, and 8 × 8 cm.

6.2 Participants

Sixteen participants were selected using the same procedure as in the pilot study. There were 12 males and 4 females. The average age was 26.8 years (min 16, max 47). To counterbalance learning effects, participants were divided into two equal groups. Group 1 started with MarkWhite-basic, group 2 started with MarkWhite. Participants signed a consent form and were compensated $10 for their participation.

6.3 Apparatus

The same smartphone camera and other equipment used in the pilot study were also used here. The implementation of the MarkWhite included the two new features mentioned earlier (i.e., zooming in/out and marking a closed-shape region on the camera preview).

6.4 Procedure

Similar procedure to the pilot study was followed here, however, slight modifications to the procedure were needed. Since MarkWhite requires more control of the camera pose (i.e., rotation around x, y, and z axes), the participants were asked to hand-hold the camera instead of fixing it on a tripod. Also, users sat in the same position and held the camera in the same pose by resting their arms on the edge of a table. This ensures that all participants had the camera approximately in the same position and pose. Each scene included a single gray card to help with selecting neutral material.

6.5 Design

6.5.1 Independent Variables

The second user study was a 3 × 9 within-subjects design with the following independent variables and levels.

Control Method with three levels: *MarkWhite-basic*, *MarkWhite*, and *auto white balance (AWB).*

Target size. This is the size of the neutrally-coloured target that the user taps/marks for white balance calibration. The size is the number of pixels occupied by the target in the captured image. We used nine sizes by combining three targets placed at three distances from the camera (i.e., 60, 120, and 180 cm). See Figure 12.

The total number of trials was 432 (= 16 participants × 3 control methods × 9 target sizes).

6.5.2 Dependent Variables

The same dependent variables from the pilot study were used: angular error and task completion time. Additionally, user experience was investigated through a questionnaire including the questions shown in Figure 17.

7 Results and Discussion (Second Study)

Post-study analyses resulted in interesting findings across the two versions of MarkWhite and AWB regarding quantitative and qualitative measures as well.

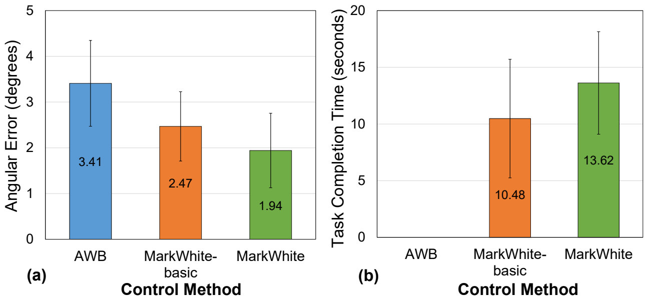


Figure 13: (a) Angular error (degrees) and (b) task completion time (seconds) by control method. Error bars show ±1 *SD*. Note: task completion time for AWB is 0.

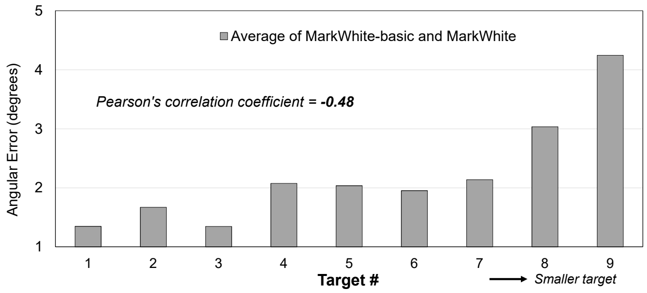


Figure 14: Angular error (degrees) by target size.

7.1 Angular Error

Over all 432 assignments, the grand mean for angular error was 2.61°. The main and interaction effects on angular error are now examined.

7.1.1 Control Method

By control method, angular error was 3.41° for AWB, 2.47° for MarkWhite-basic, and 1.94° for MarkWhite. See Figure 13a. MarkWhite had 43% and 21% lower angular errors than AWB and MarkWhite-basic, respectively. The differences were statistically significant ($F_{2,15} = 21.96,\ p < .0001$). A post hoc Scheffé test revealed that two pairwise differences (i.e., AWB vs. MarkWhite-basic and AWB vs. MarkWhite) were statistically significant. This finding shows that both versions of MarkWhite are more accurate interactive WB methods than the AWB method on-board the camera.

7.1.2 Target Size

Angular errors by target size and control method is shown in Figure 14b. AWB was not included as it does not depend on target size. The differences in angular errors due to target size were statistically significant ($F_{8,15} = 22.61,\ p < .0001$). This indicates a negative effect of smaller targets on angular error: The smaller the target, the harder for the user to mark it for adjusting white balance. This is also suggested by the Pearson correlation coefficient between target sizes and angular errors with a strong value of -0.48.

7.1.3 Interaction Effects

Angular error by control method and target size is shown in Figure 15. The differences were statistically significant ($F_{2,8} = 8.43,\ p < .0001$). These results align with the previous findings on the main effects of control method and target size on angular error. Furthermore, the Pearson correlation coefficient between target size and angular error was -0.52 for MarkWhite-basic and -0.36 for MarkWhite. This indicates is a noticeable negative effect of the target size on MarkWhite-basic (i.e., the smaller the target the higher the angular error). However, this negative effect was reduced by about 32% when using MarkWhite. This is clearly due to the zooming feature that helps the user zoom-in to smaller targets and easily mark them.

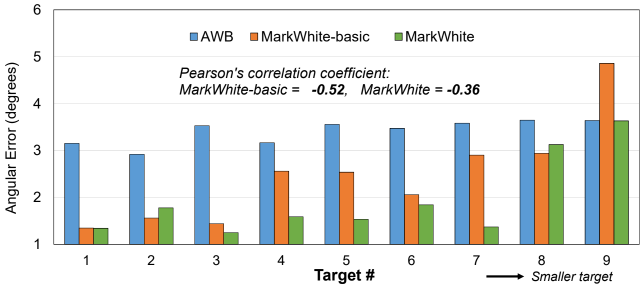


Figure 15: Angular error (degrees) by control method and target size.

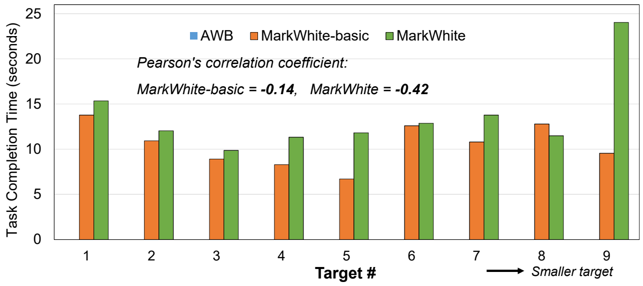


Figure 16: Task completion time (seconds) by control method and target size. Note: task completion time for AWB is 0.

7.2 Task Completion Time

Over all 432 assignments, the grand mean for task completion time was 12.1 s. More analysis and discussion follow.

7.2.1 Control Method

By control method, the task completion times were 10.5 s for MarkWhite-basic and 13.6 seconds for MarkWhite. See Figure 13b. The task completion time for AWB was 0 s because it is automatic and does not require user interaction. AWB was included as a baseline representing state-of-the-art techniques used on the latest smartphone cameras. Despite the previous finding indicating that MarkWhite was more accurate, it was about 30% slower than MarkWhite-basic. The differences were statistically significant ($F_{1,15} = 16.89,\ p < .05$).

7.2.2 Target Size

The differences in task completion times due to target size were statistically significant ($F_{8,15} = 3.75,\ p < .001$). The AWB is not included because it does not depend on the target size.

7.2.3 Interaction Effects

Task completion time by control method and target size are shown in Figure 16. The differences were statistically significant ($F_{1,8} = 2.43,\ p < .05$). These results align with the previous findings on the main effects of control method and target size on task completion time. Furthermore, the Pearson correlation coefficient between target size and task completion time was -0.14 for MarkWhite-basic and -0.42 for MarkWhite. For MarkWhite-basic, the correlation was weak between target size and task completion time due to the simplicity of the task (only tapping on a specific region on the camera preview) which mostly requires the same time for any target size. On the other hand, for MarkWhite, the correlation between target size and task completion time was stronger: the smaller the target, the longer the task completion time. In other words, users spend more time to accurately zoom in to smaller targets.

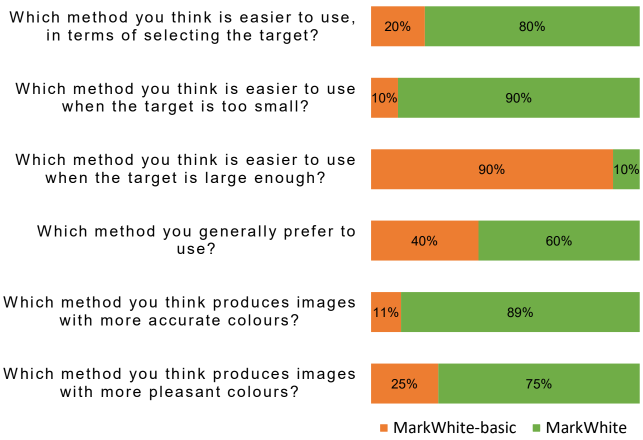


Figure 17: Results of the questionnaire from the second user study comparing the two versions of MarkWhite.

7.3 Group Effect

The differences in angular errors among the two participant groups were not statistically significant ($F_{1,14} = 0.15$, ns). The differences in task completion times among groups were also statistically not significant ($F_{1,14} = 0.36$, ns). Thus, counterbalancing was effective in offsetting learning effects.

7.4 User Experience

Figure 17 shows the results of the questionnaire for the second user study. Most participants agreed that MarkWhite was easier in selecting the neutrally-coloured target (80%), especially when the target was too small or too far. In this case, 90% agree it was easier to use. However, when the target was large, 90% of participants felt MarkWhite-basic was easier to use. In terms of general preference, participants were divided between MarkWhite-basic (40%) and MarkWhite (60%). That is probably due to Mark-White-basic being easier in selecting large targets with a single tap. On the other hand, MarkWhite is easier in selecting small targets with a bit more interaction using the zooming feature. In terms of produced images, most participants felt that MarkWhite produced more accurate image colours (89%) than MarkWhite-basic, while only 75% felt these colours were more pleasant. However, these two percentages do not conflict, since accurate colours are not always subjectively pleasant, as some people prefer warmer-coloured images even if such colours are not the most accurate [8].

8 Conclusion

In this paper, a novel interactive white balancing method for smartphone cameras, called MarkWhite, has been designed, implemented, and evaluated against existing traditional interactive white balance methods. Two user studies revealed that MarkWhite is more accurate than the existing interactive methods as well as start-of-the-art automatic methods found on the latest commercial smartphones.

MarkWhite has a limitation in the absence of neutrally-coloured material. However, this limitation is the same for all existing method that use the same interaction technique. When neutral material is absent, the next interactive options are the presets or the slider. Otherwise, automatic white balance algorithms are to be used. In realistic scenarios, there would be trade-offs between the three methods. If accuracy is the higher concern, then MarkWhite is the top choice. If efficient and fast capture is desired, then the basic version of MarkWhite would be preferred. On the other hand, if the target is capturing aesthetic images, then manipulating the temperature slider or choosing from the presets could be the better option.

References

[1] A. Abdelhamed. Two-illuminant estimation and user-preferred correction for image color constancy. Master's thesis, National University of Singapore (NUS), Singapore, 2016.

[2] J. Avery, M. Choi, D. Vogel, and E. Lank. Pinch-to-zoom-plus: an enhanced pinch-to-zoom that reduces clutching and panning. In *UIST'14*, pp. 595–604. ACM, 2014.

[3] J. T. Barron and Y.-T. Tsai. Fast fourier color constancy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR'17, pp. 886–894. IEEE, July 2017.

[4] I. Boyadzhiev, K. Bala, S. Paris, and F. Durand. User-guided white balance for mixed lighting conditions. *ACM Trans. Graph.*, 31:200:1–200:10, 2012.

[5] T. D. Buskirk. Are sliders too slick for surveys? an experiment comparing slider and radio button scales for smartphone, tablet and computer based surveys. *Methods, Data, Analyses*, 9(2):32, 2015.

[6] S. J. Castellucci and I. S. MacKenzie. Graffiti vs. unistrokes: an empirical comparison. In *CHI'08*, pp. 305–308. ACM, 2008.

[7] D. Cheng. *A Study Of Illuminant Estimation And Ground Truth Colors For Color Constancy*. PhD thesis, National University of Singapore (NUS), Singapore, 2015.

[8] D. Cheng, A. Kamel, B. Price, S. Cohen, and M. S. Brown. Two illuminant estimation and user correction preference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR'17, pp. 469–477. IEEE, 2016.

[9] D. Cheng, D. K. Prasad, and M. S. Brown. Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution. *JOSA A*, 31(5):1049–1058, 2014.

[10] D. Cheng, B. Price, S. Cohen, and M. S. Brown. Effective learning-based illuminant estimation using simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR'15, pp. 1000–1008. IEEE, June 2015.

[11] Gartner, Inc. Worldwide sales of smartphones. Online, February 2017. Retrieved March 9, 2018 from `https://www.gartner.com/newsroom/id/3609817`.

[12] D. Goldberg and C. Richardson. Touch-typing with a stylus. In *INTERACT'93 and CHI'93*, pp. 80–87. ACM, 1993.

[13] Google. Camera2 api. Online, 2018. Retrieved April 14, 2018 from `https://developer.android.com/reference/android/hardware/camera2/package-summary.html`.

[14] Google. Camera2raw code sample. Online, 2018. Retrieved April 14, 2018 from `https://github.com/googlesamples/android-Camera2Raw`.

[15] E. Hsu, T. Mertens, S. Paris, S. Avidan, and F. Durand. Light mixture estimation for spatially varying white balance. *ACM Trans. Graph.*, 27(3):70:1–70:7, Aug. 2008. doi: 10.1145/1360612.1360669

[16] P. Isokoski. Model for unistroke writing time. In *CHI'01*, pp. 357–364. ACM, 2001.

[17] H. C. Karaimer and M. S. Brown. A software platform for manipulating the camera imaging pipeline. In *European Conference on Computer Vision*, pp. 429–444. Springer, 2016.

[18] J. Kim, D. Cho, K. J. Lee, and B. Lee. A real-time pinch-to-zoom motion detection by means of a surface emg-based human-computer interface. *Sensors*, 15(1):394–407, 2014.

[19] I. S. MacKenzie. *Human-computer interaction: An empirical research perspective*. Morgan Kaufmann, Waltham, MA, 2013.

[20] Nikon. Dslr d7500. Online, 2018. Retrieved March 9, 2018 from `http://imaging.nikon.com/lineup/dslr/d7500/`.

[21] C. A. Roster, L. Lucianetti, and G. Albaum. Exploring slider vs. categorical response formats in web-based surveys. *Journal of Research Practice*, 11(1):1, 2015.

[22] Samsung. Samsung galaxy. Online. Retrieved March 9, 2018 from `http://www.samsung.com/global/galaxy/`.

[23] G. Sharma. Color fundamentals for digital imaging. *Digital Color Imaging Handbook*, 20, 2003.

[24] G. Wyszecki and W. S. Stiles. *Color science*. Wiley New York, 1982.

[25] X-Rite. Colorchecker® classic. Online, 2018. Retrieved March 12, 2018 from `https://www.xrite.com/categories/calibration-profiling/colorchecker-classic`.