A STUDY OF COLOUR RENDERING IN THE IN-CAMERA IMAGING PIPELINE

HAKKI CAN KARAIMER

A THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

AUGUST 2019

ABSTRACT

Consumer cameras such as digital single-lens reflex camera (DSLR) and smartphone cameras have onboard hardware that applies a series of processing steps to transform the initial captured raw sensor image to the final output image that is provided to the user. These processing steps collectively make up the in-camera image processing pipeline. This dissertation aims to study the processing steps related to colour rendering which can be categorized into two stages. The first stage is to convert an image's sensor-specific raw colour space to a device-independent perceptual colour space. The second stage is to further process the image into a display-referred colour space and includes photo-finishing routines to make the image appear visually pleasing to a human.

This dissertation makes four contributions towards the study of camera colour rendering. The first contribution is the development of a software-based research platform that closely emulates the in-camera image processing pipeline hardware. This platform allows the examination of the various image states of the captured image as it is processed from the sensor response to the final display output. Our second contribution is to demonstrate the advantage of having access to intermediate image states within the in-camera pipeline that provide more accurate colourimetric consistency among multiple cameras. Our third contribution is to analyze the current colourimetric method used by consumer cameras and to propose a modification that is able to improve its colour accuracy. Our fourth contribution is to describe how to customize a camera imaging pipeline using machine vision cameras to produce high-quality perceptual images for dermatological applications. The dissertation concludes with a summary and future directions.

Acknowledgments

I would like to thank everyone who supported the contributions presented in this dissertation. First of all, my adviser Prof. Michael S. Brown for his continuous feedback at every moment of this journey. Without his guidance, I would not be able to accomplish researching and writing this dissertation.

Also, I would like to thank my supervisory committee members—namely, Prof. Richard P. Wildes and Prof. Marcus A. Brubaker, together with the dissertation examiners—namely, Prof. Gene Cheung, Prof. Sai-Kit Yeung, and Prof. Roger Kempers for evaluating this dissertation.

I would like to thank our graduate program assistants from York University and NUS—namely, Ouma Jaipaul Gill and Loo Line Fong. Their guidance helped me so much in puzzling circumstances.

Additionally, I would like to thank my co-authors Dr. Iman Khodadad and Dr. Farnoud Kazemzadeh, for their contribution to my research. Furthermore, I would like to thank my labmates Dr. Li Yu, Dr. Dongliang Cheng, Dr. Rang Nguyen, Dr. Mahsa Paknezhad, Dr. Abhijith Punnappurath, Hu Sixing, Abdelrahman Kamel, Mahmoud Afifi, Abdallah Abuolaim, and Shane Segal as well as my friends and labmates from NUS and York University.

I would like to thank my family for always being there for me. Finally, I would like to thank my beloved girlfriend, Arleen. I managed to complete this dissertation with her love and support.

Contents

List of Tables

List of Figures

xiii

xiv

xv

List of Abbreviations

3D Three Dimensional.

AM Additional Materials.

API Application Programming Interface.

BM3D Block-Matching and 3D Filtering.

CB Colour Balance.

CC Colour Chart.

CCD Charge-Coupled Device.

CCT Correlated Colour Temperature.

CIE Commission Internationale de l'Eclairage.

CIE XYZ Device independent colour space approved in 1931 by the CIE to model human perception.

CMOS Complementary Metal-Oxide-Semiconductor.

CMU Carnegie Mellon University.

CST Colour Space Transform.

DNG Adobe Digital Negative.

DSLR Digital Single-Lens Reflex Camera.

EV Exposure Value.

HSV Hue, Saturation, Value.

HVS Human Visual System.

I Images.

IRLS Iteratively Re-Weighted Least Squares.

JPEG Joint Photographic Experts Group.

LED Light-Emitting Diode.

LUT Look-Up-Table.

NIR Near-Infrared.

NR Noise Reduction.

NUS National University of Singapore.

OS Operating System.

PC Personal Computer.

PSF Point-Spread-Function.

PSNR Peak Signal to Noise Ratio.

raw-RGB Unprocessed Raw Sensor Image.

RGB Red, Green, Blue.

ROMM Reference Output Medium Metric.

SC Skin Colours.

SDK Software Development Kit.

SPD Spectral Power Distribution.

sRGB Standard RGB colour space defined in 1996 for display devices such as monitors.

UV Ultraviolet.

WB White Balance.

YCbCr Luminance; Chroma: Blue; Chroma: Red.

Chapter 1

Introduction

When we take a photograph using a digital camera, we often assume the output image is a direct output of the camera's sensor. However, there are a number of processing steps applied on cameras to convert the digital response of the light falling on the camera's sensor to a final red, green, blue (RGB) colour image intended for human perception and use on digital displays. These processing steps are applied in sequence and make up what is known as the "camera imaging pipeline". Figure 1.1 shows a standard diagram of the in-camera processing pipeline (Ramanath, Snyder, Yoo, & Drew, 2005). At a high level, the overall pipeline can be categorized into two stages: (1) a colourimetric conversion stage and (2) photo-finishing stage. The first stage converts sensor values from their sensor-specific colour space to a perceptual colour space. We often refer to this as the colour space transform (CST). This first stage preserves a linear relationship between the scene radiance and the image irradiance. In colour science such image states are called "scene-referred" and imply the image has a direct relationship to the physical scene that was imaged (Sharma, 2003).
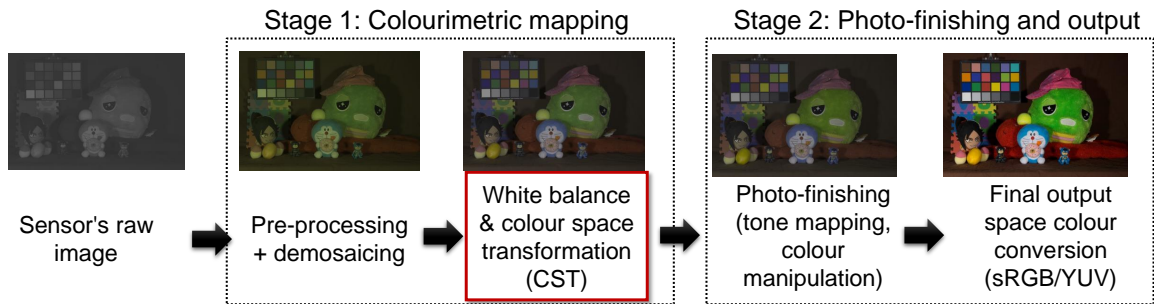
Figure 1.1: A diagram of a typical in-camera imaging pipeline. Work in this dissertation targets the colourimetric mapping in the first stage in the pipeline that converts the camera-specific colour space to a perceptual colour space. The procedure targeted by our work is highlighted in red and represents an intermediate step in the overall pipeline.

After the colourimetric conversion, the second stage further processes the colourmetric image such that it is suitable for consumer displays. This requires the image values to be mapped to a standard RGB format (sRGB). This second stage includes camera-specific proprietary colour manipulation to improve the perceptual qualities of the image—that is, to make the image more visually appealing. As a result, we can consider this stage to also include photo-finishing manipulation. In colour science, such image states are called "display-referred" or "output-referred" as the image data is not longer directly linked to the physical scene, but instead processed for display or printing (Sharma, 2003).

1.1 Motivation

This thesis studies the colour rendering applied onboard cameras. Surprisingly, a study of this nature is lacking in the computer vision literature. This stems from the difficulty in accessing camera hardware. For many years it was possible only to obtain the display-referred photo-finished sRGB images intended for display. Only

recently did camera manufacturers allow consumers and developers access to the initial unprocessed raw sensor image (often called a raw-RGB image). In the context of computer vision, the drawback of sRGB (output) images is that they have been manipulated by camera-specific photo-finishing routines that alter the colours. From a computer vision application standpoint, such color manipulation makes it difficult to develop a high-level application that relies on colour, because colours can appear different depending on the camera. In a similar manner, the drawback of raw-RGB images is that the RGB values are in the sensor-specific colour space. This means the raw-RGB value can be vastly different when cameras using different sensors are observing the exact same scene under the same illumination. As with sRGB, it is challenging to design robust algorithms that work across multiple cameras using unprocessed raw-RGB images.

In principle, the colourimetric conversion stage on the camera used to map the raw-RGB values to the same perceptual colour space should make colour values similar across multiple cameras and sensors. The problem, however, is that access to the individual processing steps or intermediate image stages within the camera hardware remains closed as camera manufacturers do not allow access to their underlying in-camera pipelines. This makes it difficult to assess the quality of the individual components, including those that make up colourimetric conversion, due to the lack of access to camera hardware. To address this issue, open hardware implementations have been proposed (e.g., Adams et al. (2010);Rowe, Goode, Goel, and Nourbakhsh (2007)). Similar open design and functionality can be provided via a software-based platform. The benefits of a software framework over a hardware solution are that it

can work on images saved from a variety of different cameras and in an off-line manner. Moreover, a software platform allows greater flexibility in processing the image at intermediate stages than possible on fixed hardware implementations. Therefore, providing a software-based research platform is needed to facilitate experimentation and analysis. The need for a software-based camera emulator platform is the impetus for our first contribution in this dissertation.

With a software camera pipeline platform established, we can analyze the intermediate stages in the hardware pipeline to see if an intermediate stage is indeed better than raw-RGB or sRGB in terms of colourimetric consistency. This also gives us an opportunity to examine the quality of the existing implementation for colourimetric conversion. Finally, based on our knowledge of how cameras work, we can advise researchers developing their own software platforms for machine vision cameras on best practices for camera calibration and consumer camera emulation.

In the following, we provide a discussion on the scope of the work. Afterwards, the chapter concludes with a road map for the rest of the dissertation.

1.2   Objective

Our first objective is to implement a new software platform that allows easy access to each stage of the camera imaging pipeline. The platform allows modification of the parameters for individual components as well as the ability to access and manipulate the intermediate images as they pass through different stages. This first objective forms a basis for the rest of the dissertation and makes it possible to complete the following objectives.

Our second objective is to investigate a consistent image state in the camera

imaging pipeline. The goal is to make the colours consistent among multiple cameras imaging the same scene. Our approach leverages our ability to access the camera imaging pipeline stages, and proposes that cameras should be allowed to output a new image format related directly to the perceptual color space.

Directly related to our second objective, our third objective is to improve colour reproduction accuracy within the camera imaging pipeline. To this end, we analyze the colorimetric procedure that is currently applied, and state its limitations. We then propose two methods to improve the colour reproduction accuracy.

Our final objective is to use our knowledge of camera pipelines to outline a procedure for developing a custom imaging pipeline for use with machine vision cameras. Most machine vision cameras used in computer vision apply minimal processing and provide little detail to engineers and researchers on how to interpret their image values. We detail a straightforward procedure to calibrate such devices. We do this for a specific application scenario of dermatological imaging for skin analysis. This work also describes methods to integrate non-visible spectral information into the visible sRGB image. Quantitative and subjective results, including feedback from clinicians, suggest the effectiveness of our customization procedure.

1.3 Contributions

This dissertation makes four contributions on the colour-rendering process in the camera imaging pipeline. These contributions are as follows:

- First, we present a software platform to allow easy access to each stage of the imaging pipeline. Our approach operates on images saved in Adobe Digital

Negative (DNG) raw-RGB image format, which represents the minimally processed sensor response from the camera and the starting point for the camera processing pipeline. Our platform allows images to be opened and run through a software rendering application programming interface (API) that parallels the onboard processing steps, including the individual processing components and their associated parameters. More specifically, our platform provides API calls that allow the modification of processing components' parameters and full access to the intermediate images at each processing stage. Such intermediate images can be modified and inserted back into the pipeline to see the effect on the final output. The proposed software platform is easily integrable with other softwares, such as Matlab, and provides a much needed environment for improving camera imaging, or performing experiments within the proper context of the full camera imaging pipeline. This work has been published in Karaimer and Brown (2016).

- Second, we advocate for a new image format to be made available to consumers and application developers. Specifically, we argue that the internal image state after it has been processed by the colourimetric steps in the camera pipeline is more suitable for applications that assume imaged scene values are consistent among multiple cameras. To demonstrate this, we perform experiments that analyze the consistency of colour values at different image states for scene points imaged by multiple cameras. We show that the colourimetric image state is by far the most consistent across multiple cameras. These experiments reinforce our advocacy for camera manufacturers to provide access to a colourimetric image format. This work has been published in Karaimer and Brown (2019).

- Third, we describe two methods to improve colour reproduction accuracy on digital cameras. Toward this goal, we first overview the current colourimetric mapping process and discuss why limitations in white balance create the need for per-illumination CSTs. Next, two methods for improving the colourimetric mapping are described. The first is to extend the current interpolation method to include an additional pre-calibrated illumination. This simple modification provides improved results and can be easily incorporated into the existing in-camera pipeline. Our second strategy requires no interpolation and uses a single fixed CST. This second method relies on true colour constancy (i.e., full colour balance) instead of traditional white balance and is currently suitable for use in an off-camera platform. Our experiments show that both proposed strategies offer notable improvements in the colour reproduction accuracy for both DSLR and mobile phone cameras. As part of this work, we have also created a dataset of 700 carefully calibrated colourimetric images for research in this area. This work has been published in Karaimer and Brown (2018).

- Lastly, we describe how to customize a machine vision camera image pipeline to produce high-quality perceptual images for dermatological applications. As part of this work, we describe the features provided by a typical machine vision camera and discuss why they are not suitable to produce high-quality perceptual images. Our primary contribution is to overview a calibration procedure that not only colourimetrically calibrates the machine vision camera but also allows it to mimic the photo-finishing applied on consumer cameras. Our dermatological imaging system also has the ability to capture images from select narrow spectral bands in both the visible and near-infrared (NIR) range. As part of this case

study, our secondary contribution is to describe how select spectral images can be incorporated into the visible RGB image to provide enhanced imagery. To this end, we gathered feedback from clinicians to understand their preference regarding which bands and integration methods are preferred. This work has been published in Karaimer, Khodadad, Kazemzadeh, and Brown (2019).

1.4   Road Map

The rest of this dissertation is organized as follows. Chapter 2 provides preliminaries to understand colour and work related to the topics in this dissertation. Our software platform for manipulating the camera imaging pipeline is discussed in Chapter 3. Chapter 4 presents a third image format that makes imaged scene values that are consistent among multiple cameras. Chapter 5 describes methods for improving colour reproduction accuracy on cameras. Our customization approach for dermatological imaging is described in Chapter 6. Finally, Chapter 7 concludes the dissertation with a summary and a discussion on possible future work.

Chapter 2

Preliminaries and Related Work

This chapter presents background for colour and imaging, as well as related work for white balance, colour constancy, and colour calibration.

2.1 Preliminaries

From the biological standpoint, colour is defined based on how we perceive electromagnetic radiation emitting or reflecting from physical objects. These definitions limit themselves explaining how the electromagnetic radiations are perceived and the change of perception under different viewing conditions. In the following, a short summary of the human visual system is provided. Then the physical aspects of colour are presented. Afterwards, *colourimetry*, which is the low-level (i.e., sensor-level) response of the human visual system to the physical phenomenon, is introduced. Finally, the section concludes with a summary of image formation in cameras and the camera imaging pipeline.

2.1.1 The Human Visual System

The human visual system (HVS) consists of the eyes, the optic nerves, and the components in the brain. Optical elements in the eye are responsible for collecting light reflecting from the surfaces, and focus the light onto the *retina*. The *retina* is the curved surface at the back of the eye. The retina consists of photoreceptors, and neural processes start in the retina. The photoreceptors contain *rods* and three types of *cones* that convert the light into a neural signal. Rod photoreceptors are highly sensitive to light and used for monochromatic vision under very low light levels. At higher levels of light the rods become saturated. Under normal lighting conditions cone photoreceptors are responsible for colour vision and come in three types with different spectral sensitivities. The three types of cones are long, medium, and short wavelength sensitive. Figure 2.1 shows the normalized spectral sensitivities of long, medium, and short wavelength sensitive cones. Note that in addition to the photoreceptor, there are additional processing units in the retina, e.g., bipolar, horizontal, ganglion and amacrine cells,which combineto provide non-trivial processing (Fairchild, 2013).

Models of Colour Vision Young (1802) and von Helmholtz (1925) introduced the *trichromatic theory*, which is established on the existence of three types of cone photoreceptors in the HVS. Based on the theory, weighted combination of the three sets of receptors is used to sort out the colour appearance in the brain. The phenomenon of colour perception is better described in the *opponent colour theory* (Hering, 1964). In this theory, three types of cone photoreceptors are accepted, but they function based on opponent mechanisms (i.e., red - green, blue - yellow, dark - light). The *dual process theory* accepts that both trichromatic theory and the opponent process theory
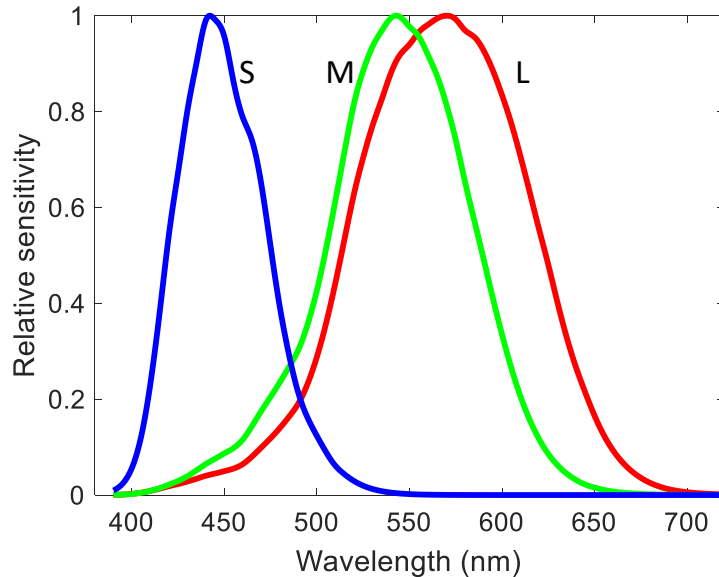
Figure 2.1: Spectral sensitivities of the long-, medium-, and short-wavelength-sensitive cones.

are correct, and hypothesises multi-stage processing in the visual system. According to the theory, the first stage of processing photoreceptors is trichromatic, but these signals are encoded into opponent signals in the following stages of the processing (e.g., Haupt (2000), MacAdam (1970), Hurvich and Jameson (1957)).

Human vision is an highly complex combination of optics, biochemistry, and neurology. A good grasp of the human visual system may lead to improvements on the design of imaging systems. For a more thorough overview of the human visual system see, e.g., (Fairchild, 2013). The next section provides the remaining background on colour in light and objects.

2.1.2 Colour in Light and Objects

Colour of light Isaac Newton hypothesized that colour was a property of light through his experimentation with the diffraction of light in a prism (Newton, 1672). *Light sources* are actual physical emitters of radiant energy. We can specify the colour properties of light sources by measuring their spectral power distribution (SPD). Another way to specify colour properties of light sources is standardization. Hence, a *standard light source* is specified using a group of physical and chemical specifications. An *illuminant* is a specific SPD. Therefore, an illuminant SPD does not have to correspond to a real source. Some of the sources can be black-body radiators, the sun, sky, photographic flash, incandescent, fluorescent, or light-emitting diode (LED). Among them a *black-body radiator*, or *Planckian radiator*, is an idealized source which absorbs incoming radiation in all wavelengths and emits radiation based on its temperature. When more energy is emitted by the black body, the temperature of the black body increases as well as the energy shifts towards shorter wavelengths. Given the absolute temperature T, the SPD of the black-body radiator can be calculated using Planck's equation:

$$B_\lambda(\lambda, T) = \frac{2hc^2}{\lambda^5} \frac{1}{e^{\frac{hc}{\lambda k_B T}} - 1},\tag{2.1}$$

where λ is wavelength, k_B is the Boltzmann constant, h is the Planck constant, and c is the speed of light. Black-body's temperature T is called *colour temperature* since colour (i.e., SPD) changes based on T. Since black-body radiators cannot be found commonly, another quantity called *correlated colour temperature* (CCT) is often used. Therefore, for the other sources CCT can be used. The CCT of a source is the colour temperature of a Planckian radiator that has almost the same chromaticity as the

source.

Colour of objects Objects become visible only when illuminated by light (unless the object emits light). Since energy can be neither created or destroyed, the radiant energy incident on an object is reflected, absorbed, or transmitted. In other words, the sum of the reflected, absorbed, and transmitted power must sum to the radiant energy incident on an object at each wavelength. In this context, the law of conservation of energy can be expressed as:

$$\Phi(\lambda) = R(\lambda) + T(\lambda) + A(\lambda), \tag{2.2}$$

where Φ is the radiant energy incident on an object, $R(\lambda)$ is the reflected component, $T(\lambda)$ is the transmitted component, and $A(\lambda)$ is the absorbed component. Since the quantities always sum to the incident light, the quantities of reflectance, transmittance, and absorptance are measured as percentages of the radiant energy incident on an object. Therefore, reflectance is the fraction of reflected light divided by the incident light. Transmittance is the fraction of light that goes through divided by the incident light. Absorbance or absorptance is the fraction of absorbed light divided by the incident light.

Wavelength dependence of the absorption, transmission, and reflection determines what colour the HVS sees. For example, a green leaf under white light (which has energy in all regions of the spectrum) reflects back mostly in the region that corresponds to the middle wavelength of the spectrum. In this way, the perception of green is produced. Therefore, the objects that we see as red reflect more light at longer wavelengths ($\sim$ 600–700 nm), objects that we see as green reflect more light at middle wavelengths ($\sim$ 500–600 nm), and objects that we see as blue reflect more

light at shorter wavelengths ($\sim$ 400–500 nm). Absorptance, transmittance, and reflectance are linear spectral metrics since they change linearly with respect to the incident light. In the following, we will present a summary of colourimetry.

2.1.3   Colourimetry

Colourimetry is the measurement of perceived colour. The work in Wyszecki (1973) makes a clear distinction on what is *basic colourimetry* and *advanced colourimetry*. *Basic colourimetry* is a tool to check if two SPDs match for a standard observer. However, *advanced colourimetry* incorporates techniques to assess colour appearance methods for observer adaptation, including colour difference quantification and chromatic adaptation modeling.

In order to move deeper into colourimetry, we need to define the following terms that will be used in colourimetry. *Radiometry* is the field of research that studies measurement of electromagnetic radiation. *Photometry* studies measurement of the HVS response to electromagnetic radiation. Transition from radiometric to photometric quantities can be done using the spectral response of the HVS. For example, Figure 2.3 shows the achromatic (photopic) sensitivity of the HVS over the wavelengths of visible light. Radiation sensed in the achromatic channel of the HVS is measured as *luminance*. Hence, luminance can be computed as follows:

$$L = k \int_{\omega} S(\lambda)R(\lambda)V(\lambda)d\lambda, \qquad (2.3)$$

where λ is the wavelength, ω is the visible spectrum 380–720 nm, $S(\lambda)$ is the radiant flux coming from the illumination source, $R(\lambda)$ is the reflectance/transmittance of the object, $V(\lambda)$ is achromatic sensitivity of the HVS, and k is the constant that changes

the unit of luminance. When $S(\lambda)$ is in the unit of watt per steradian per square meter, then k should be set to 683.002 to make the unit of luminance candelas per square meter.

Weighting functions for colour vision The $V(\lambda)$ function describes the achromatic response of the HVS, but as we mentioned in Section 2.1.1 three types of cones are responsible for colour vision. Hence the low-level colour vision is a linear combination of the three cone functions. The first attempt to estimate the sensitivity functions came in 1931 while the cone spectral sensitivities were not known. Therefore, linear transformations of the cone spectral sensitivities were determined using a colour matching experiment by Guild (1931) and by Wright (1928–1929). Users of the colour matching experiments tried to match a test colour with three visually independent primary colours by leveraging Grassmann's Laws (Grassmann, 1853), which apply an additive mixture of colours. When taking additive combination of the primary colours almost all test colours could be produced. In the cases that a colour could not be produced, the users were allowed to add a primary to the test colour. Using primaries at 436, 546, and 700 nm, users were showed a number of test colours and this way the CIE RGB colour matching functions are produced (see Figure 2.2). The implication of this result is that we can specify matching in terms of the response of three channels. Given a SPD of a stimulus, CIE RGB makes it possible to specify the stimulus in terms of red, green, and blue that are required to match it. Threfore, the CIE RGB system for colour rendering forms a system to specify a colour.

Some of the properties of the colour matching functions can be summarized as

follows: Colour matching functions are invertible linear transformations of cone spectral sensitivities. Each set of primaries makes up a set of colour matching functions. Transformation between different sets of colour matching functions can be done using a 3×3 matrix. Tristimulus values for a test colour can be computed using simple integration:

$$R = k \int_\omega \bar{r}(\lambda)S(\lambda)R(\lambda)d\lambda,$$
$$G = k \int_\omega \bar{g}(\lambda)S(\lambda)R(\lambda)d\lambda, \quad (2.4)$$
$$B = k \int_\omega \bar{b}(\lambda)S(\lambda)R(\lambda)d\lambda,$$

where $\bar{r}, \bar{g}, \bar{b}$ are CIE RGB colour matching functions, λ is the wavelength, ω is the visible spectrum 380–720nm, $S(\lambda)$ is the radiant flux coming from the illumination source, $R(\lambda)$ is the reflectance/transmittance of the object, and k is the constant that changes the unit, as discussed before. For the relative colour and gamut measurements the constant k does not factor in. Stimuli that produce the same R G B values will have a visual match under the same adaptation conditions.

In CIE RGB, the selected primaries were a real set of physical primaries and this caused negative lobes in CIE RGB colour matching functions. Negative values in CIE RGB were found to be prone to errors. Also, researchers wanted to make one of the trisitimulus values directly proportional to the luminance. For all these reasons, CIE XYZ is introduced as a linear transform to CIE RGB:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (2.5)$$
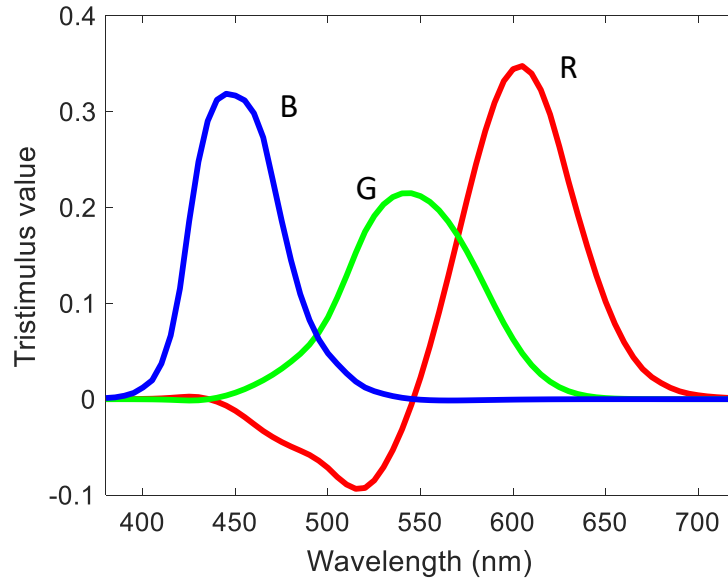
Figure 2.2: CIE RGB Colour matching functions using primaries at 436, 546, and 700 nm. The disadvantage of using real physical primaries is that colour matching functions might have negative lobes.

where $[R\ G\ B]^T$ is a CIE RGB trisitimulus value and $[X\ Y\ Z]^T$ is a CIE XYZ trisitimulus value. The same 3×3 matrix transformation can also be applied to CIE RGB colour matching functions and CIE XYZ colour matching functions can be obtained. This helped to make CIE RGB colour matching functions positive, but in this case the corresponding primaries are not physical. Figure 2.4 shows the CIE XYZ system for colour rendering.

When researchers began the systematic study of colour, 3D visualization was difficult; so, they often worked with 2D representations. The 2D representations are called *chromaticity* diagrams. One of the examples is the CIE xy chromaticity diagram. It is obtained by normalizing the XYZ tristimulus values by the sum of all

17
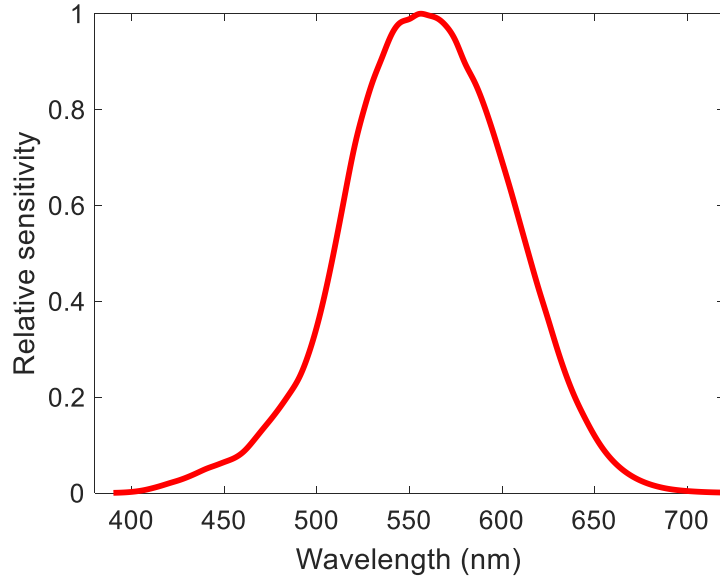
Figure 2.3: Achromatic (i.e., photopic) sensitivity function of the human visual system.

three. Formally:

$$
\begin{aligned}
x &= \frac{X}{X+Y+Z}, \\
y &= \frac{Y}{X+Y+Z}, \\
z &= \frac{Z}{X+Y+Z}.
\end{aligned}
\tag{2.6}
$$

A clear implication of this is that the sum of x, y, and z is 1; in fact only two of them are independent. Invariably z is dropped. Also, given the x, y, and Y value, X, Y, and Z can be reconstructed. In this way the CIE xyY space is obtained. Figure 2.5 shows the CIE xy chromaticity diagram for monochromatic stimuli for wavelengths ranging from 380nm to 720 nm in the xy space. The curved boundary is called the *spectral locus*. The line connecting the bottom of the spectral locus is called the *purple line*. Based on the sensitivities of HVS, all trisitimulus colours lie inside the
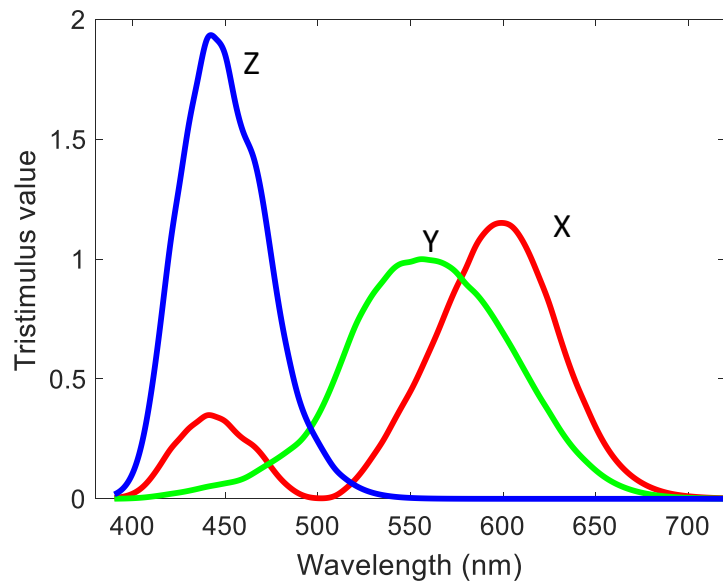
Figure 2.4: CIE XYZ Colour matching functions with respect to imaginary primaries. The advantage of using imaginary primaries is that all tristimulus values are in the positive octant. However, the primaries are not physical.

spectral locus and purple line. The chromaticity transformation can be thought of as removing the luminance information. As we discussed before, since the CIE XYZ primaries are not physical they are not part of the locus. The CIE XYZ primaries are located at (1,0), (0,1), and (0,0).

When chromaticity transform is applied to black-body (Planckian) radiators, they all lie on a curve called the *Planckian locus.* In order to visualize the Planckian locus, Planck's law needs to be applied to obtain the SPD of black-body radiators first. Then the SPDs can be projected to chromaticity coordinates. Figure 2.5 shows the Planckian locus on CIE xy chromaticity and the crossing lines on the curve indicate constant correlated colour temperature.

Although the CIE XYZ and CIE xy are based on human perception, both were not perceptually uniform (i.e., when the Euclidian distances are computed the distances do not correspond to perceived equal differences). Also, in a uniform chromaticity space the crossing lines on the Planckian curve are perpendicular lines to the curve. Therefore, attempts to obtain uniform colour spaces have been made. The first uniform chromaticity colour space that is built targeting perpendicular CCT lines was CIE u-v. X, Y, Z and u, v are related as follows:

$$
\begin{aligned}
u &= \frac{4X}{X + 15Y + 3Z}, \\
v &= \frac{6Y}{X + 15Y + 3Z}.
\end{aligned}
\tag{2.7}
$$

Considering the advantages, CIE u-v is used to calculate the CCT of an illumination. In the next section we will summarize the image formation in cameras and the camera imaging pipeline.
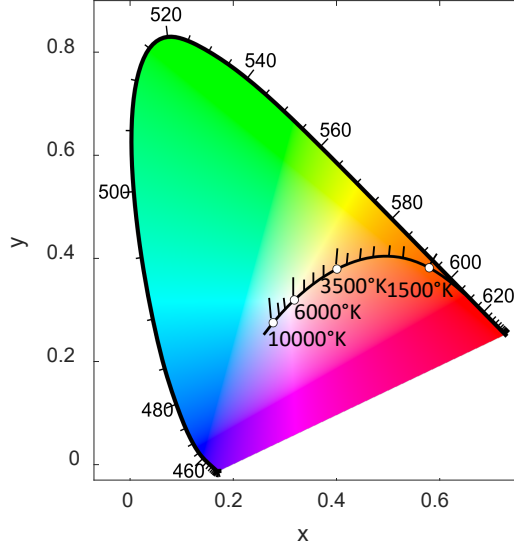
Figure 2.5: CIE xy chromaticity diagram and the Planckian locus.

2.1.4   Image Formation and the Camera Pipeline

The early stages of the camera pipeline mimic the initial stages of the HVS. In particular, camera sensors have three spectral filters placed over the light-sensitive photodiodes that respond to different parts of the visible spectrum similar to how the human retina has three cone cells sensitive to the long, medium, and short visible wavelengths. The raw-RGB image $I(\mathbf{x})$ is formed at a pixel $\mathbf{x}$, which can be written as the following:

$$I(\mathbf{x}) = \int_{\omega} Q_c(\lambda)S(\lambda)R(\mathbf{x},\lambda)d\lambda, \tag{2.8}$$

where Q_c is the spectral sensitivity of the sensor's colour filter array, λ is the wavelength, ω is the visible spectrum 380–720 nm, $S(\lambda)$ is the radiant flux coming from the illumination source, $R(\mathbf{x}, \lambda)$ is the reflectance/transmittance of the object at a pixel $\mathbf{x}$, and c is the colour channel $c = r, g, b$.

After the raw-RGB image is formed, it is processed by camera pipeline steps until a visually-pleasing image is obtained. The raw-RGB image first undergoes a series of pre-processing operations, such as linearizing the raw-RGB values and correcting for issues related to the lens's vignetting and chromatic aberrations. Then it is demosaiced to provide three full colour channels. Afterwards, the image undergoes colourimetric conversation routines to convert its values to device-independent values based on the CIE XYZ colour space. Note that this includes a white-balance step realized as 3×3 diagonal matrix operation followed by a linear colourimetric mapping step as a 3×3 matrix operation. Most cameras include an additional nonlinear colourimetric mapping step that is performed as a 3D look-up-table (LUT). After this stage, the image values are in a colourimetric colour space that is directly related to the scene (i.e., a scene-referred colour space). After the colourimetric conversion, the final stage is to apply photo-finishing operations to modify the image further to be visually appealing and to be suitable for use on display devices. The final image is encoded in a display-referred sRGB colour space. The following section summarizes the literature that is related to the topics of this dissertation.

2.2 Related Work

In this section, we describe work related to the objectives in this thesis.

2.2.1 Computational Cameras

The basic steps composing the camera processing pipeline are illustrated in Figure 1.1 and may vary among different camera makes and models. Full details of each component are outside the scope of this dissertation and readers are referred to Ramanath et al. (2005) for an excellent overview. As mentioned in Section 1.1, many of the components in the pipeline (e.g., white balance, noise reduction) are stand-alone research topics in the computer vision and image processing community. Unfortunately, the hardware implementation of the camera pipeline and closed nature of proprietary cameras make it difficult for most researchers to directly access the individual components.

To address this issue, open hardware platforms have been proposed. Early work included the CMU-camera (Rowe et al., 2007) targeting robotic vision. While individual components (e.g., white balance) were not accessible, the camera provided low-level image access and subsequent hardware releases allowed capture of the raw-RGB sensor response. A more recent and comprehensive hardware platform was the FrankenCamera introduced by Adams et al. (2010). The FrankenCamera was designed as a fully operational camera that provided full access to the underlying hardware components, including control of the flash and shutter, as well as the underlying imaging pipeline. The FrankenCamera platform targeted computational photography applications; however, the platform was suitable for modifying individual components in the imaging pipeline. While the FrankenCamera project has officially stopped, much of the platform's functionality has been incorporated into the recent Android Camera2 API (Google, 2015) that is available on devices running Android OS. The benefits of a software framework over a hardware solution are that it can work on

images saved from a variety of different cameras and in an off-line manner. Moreover, a software platform allows greater flexibility in processing the image at intermediate stages than possible on fixed hardware implementations. One of the earliest known works in this area was proposed by Farrell, Xiao, Catrysse, and Wandell (2004). The proposed software platform required radiometric image data as input and allowed modifications on optics, scene, sensor, and processing pipeline. However, the tool was designed for synthetic imagery, and the pipeline that has been used was suitable for the first generation of digital cameras using a simple camera processing pipeline.

2.2.2   Colour Constancy and White Balance

HVS has been shown to be highly adaptive to changing conditions. One adaptation mechanism is colour constancy. Hence, this section presents related works in colour constancy and white-balance research.

Colour constancy is the ability of the human visual system to see objects as being of the same colour under different light sources. According to the von Kries hypothesis (von Kries, 1878), independent gains are applied to the human retinal cones' response. An enhanced version of the von Kries hypothesis has been studied as the *retinex theory* (Land & McCann, 1971; Land, 1977, 1986). In the most recent form (Land, 1986), three colour mechanisms with the spectral responsivities of the cone photoreceptors have been proposed. These mechanisms are called *retinexes*. Retinexes are a combination of retinal and cortical mechanisms. Within this context a three-dimensional colour space has also been proposed. The dimensions of the colour space are the long-, middle-, and short-wavelength retinexes. To determine the output of a retinex, the signal at any point in the scene is normalized with an

average of the signals in that retinex throughout the scene. The influence of the spatial distribution of colours in a scene can be modelled by changing the spatial distribution of the retinex signals that are used to normalize a given point in the scene.

Cameras mimic this ability by applying computational colour constancy in the camera imaging pipeline. The process starts with estimating the response of the illumination in the sensor-specific colour space. After the illumination is estimated, a transformation is applied to remove the effect of the illumination. A large body of research has been proposed to estimate the illumination. These include statistical methods (e.g., van de Weijer, Gevers, and Gijsenij, 2007; Finlayson and Trezzi, 2004; Barnard, Cardei, and Funt, 2002), gamut-based methods (e.g., Forsyth, 1990; Finlayson, 1996; Gijsenij, Gevers, and van de Weijer, 2010), and machine-learning methods (e.g., Barron, 2015; Barron and Tsai, 2017; Cheng, Price, Cohen, and Brown, 2015b; Chakrabarti, 2015; Hu, Wang, and Lin, 2017; Bianco, Cusano, and Schettini, 2017; Shi, Loy, and Tang, 2016; Oh and Kim, 2017). Once the illumination is estimated, a simple 3×3 diagonal matrix can be used to normalize the illumination response to appear white.

While a diagonal matrix is sufficient to make sure that the achromatic colours are correct, attempts to correct all the colours (i.e., true colour constancy) have also been proposed (e.g., Finlayson, Drew, and Funt, 1993a, 1993b; Chong, Gortler, and Zickler, 2007; Cheng, Price, Cohen, and Brown, 2015a). Work by Finlayson et al. (1993a, 1993b) proposed the use of a spectral sharpening transform in the form of a 3×3 full matrix that was applied to the camera-specific colour space. The transformed colour space allowed the subsequent 3×3 diagonal white balance correction to perform

better. To establish the sharpening matrix, it was necessary to image known spectral materials under different illuminations. Chong et al. (2007) later extended this idea to directly solve for the sharpening matrix by using the spectral sensitivities of the sensor. When the estimated sharpening matrix is combined with the diagonal matrix, these methods are effectively performing a full colour balance.

The drawback of these methods, however, is the need to have knowledge of the spectral sensitivities of the underlying sensor or imaged materials. Recent work by Cheng et al. (2015a) proposed a method to compute a full colour-balance correction matrix without the need of any spectral information of the camera or scene. Their work found that when scenes were imaged under specific types of broadband illumination (e.g., sunlight), the diagonal white balance correction was sufficient to achieve full colour balance. They proposed a method that allowed images under other illuminations to derive their full colour correction using colours observed under the broadband spectrum. A full review of colour constancy methods falls outside of the scope of this dissertation. We refer the interested readers to excellent surveys: Gijsenij, Gevers, and van de Weijer (2011) and Barnard, Martin, Coath, and Funt (2002).

2.2.3 Colour Calibration

After white balancing, the colours are still in the sensor-specific colour space. A transformation that maps the sensor-specific colour space to a device-independent perceptual colour space is needed. Most of the methods calculate a direct transformation to measurements obtained from a colour rendition chart or other calibration device that has known values in a standard perceptual colour space (e.g., CIE

XYZ) (e.g., Finlayson, Darrodi, and Mackiewicz, 2015; Finlayson, Gong, and Fisher, 2016; Finlayson, Mackiewicz, and Hurlbert, 2015; Bastani and Funt, 2014; Hong, Luo, and Rhodes, 2001; Bianco, Bruna, Naccari, and Schettini, 2012). Most work focuses on the types of mathematical functions used to perform the colour mappings, such as high-order polynomial terms (e.g., Hong et al., 2001; Finlayson, Mackiewicz, and Hurlbert, 2015), LUTs (e.g., Hung, 1993), or neural networks (e.g., Cheung and Westland, 2002). Issues such as non-uniform illumination are also addressed (e.g., Bastani and Funt, 2014; Finlayson, Darrodi, and Mackiewicz, 2015; Finlayson et al., 2016) as it can be difficult to ensure uniform illumination on the physical calibration pattern. There has even been work that demonstrates that additional physical filters can be used to achieve better colourimetric properties (e.g., Farrell and Wandell, 1995; Finlayson, Zhu, and Gong, 2018). Camera manufacturers have undoubtedly incorporated these research findings into their camera pipelines; however, users do not have access to such results unless they manually perform and apply the calibration methods discussed in these works.

2.2.4 Spectral Image Fusion

Another area that is related to the work presented in this dissertation is spectral image fusion. There are a number of methods suitable to perform image fusion. Early work typically relied on copying data from frequency decompositions between images and then reconstructing a new image (e.g., de Zeeuw, Piella, and Heijmans, 2004; Misiti, Misiti, Oppenheim, and Poggi, 2007). More modern methods involve joint-image filtering (e.g., Paris and Durand, 2006; Paris, Hasinoff, and Kautz, 2011). While these methods are generic in nature, they are suitable for transferring narrow-band

spectral data to RGB images.

There are several works focused directly on processing visible images with the help of non-visible data. For example, Krishnan and Fergus (2009) showed how noisy low-light images could be denoised using a corresponding high-quality NIR image. Similar work by Wu, Samadani, and Gunawardane (2011) removed light glows for low dynamic range web cameras using NIR imagery. Fredembach, Barbuscia, and Süsstrunk (2009) and Süsstrunk, Fredembach, and Tamburrino (2010) proposed methods that used NIR images to remove skin imperfections, like freckles, pores, warts, and wrinkles from RGB images. Zhang, Sim, and Miao (2008) showed how to combine NIR images to create high-dynamic-range RGB images. These methods employed variations on joint-image filtering as discussed above. Recent work by Connah, Drew, and Finlayson (2014) and Finlayson and Hayes (2015) proposed gradient domain approaches that avoid both filtering and frequency decomposition.

In Chapter 6, we evaluate several of the methods above to test their suitability for our application for dermatological evaluation.

2.3   Summary

Our research differs from previously published methods for the following reasons. First, prior work on camera simulation platforms often have to motivate their arguments via synthetic imagery generated using relatively simple camera processing models. The proposed research platform in this dissertation aims to provide the FrankenCamera's functionality via a software-based platform. Second, our research focuses on how the colour mapping is performed onboard the camera without the assistance of a calibration pattern but instead relying on methods related to factory

calibration. Finally, the lack of a concrete example that details how to customize a machine vision camera beyond simple colour mapping is one impetus for this dissertation.

Chapter 3

A Software Platform for Manipulating the Camera Imaging Pipeline

In this chapter, we will investigate the drawbacks of the hardware embedding of the camera imaging pipeline. The hardware embedding not only hinders research but also makes evaluating the effects of modifying an individual pipeline component on the final camera output challenging, if not impossible. As a result, many researchers work on topics outside the proper context of the full imaging pipeline. Therefore, this chapter presents a new software-based camera simulator that allows access to each stage of the pipeline. The simulator allows modification of the parameters for individual components as well as the ability to access and manipulate the intermediate images as they pass through different stages. In the following, we detail our platform design and demonstrate its usefulness on a number of examples.

3.1 Introduction

Digital cameras are the cornerstone for virtually all computer vision applications as they provide the image input to our algorithms. While camera images are often modeled as simple light-measuring devices that directly convert incoming radiance to numerical values, the reality is that there are a number of processing routines onboard digital cameras that are applied to obtain the final RGB output. These processing steps are generally performed in sequence and collectively make up the camera imaging pipeline. Examples of these processing steps include Bayer pattern demosiacing, white balance, colour space mapping, noise reduction, tone-mapping, and colour manipulation. Many of these processing steps are well-known research topics in their own right—for example, white balance, colour space mapping (colourimetry), and noise reduction.

Although cameras are the most prominent hardware tools in computer vision, it is surprisingly difficult to get access to the underlying imaging pipeline. This is because these routines are embedded in the camera's hardware and may involve proprietary image manipulation that is unique to individual camera manufacturers. This is a significant drawback to the research community. In particular, it forces many researchers to work on topics outside the proper context of the full imaging pipeline. For example, much of the work targeting white balance and colour constancy is performed directly on the camera-specific raw-RGB images without the ability to demonstrate how it would affect the final output on the camera. Another example includes noise reduction (NR) targeting sensor noise. On a camera, NR is applied before many of the non-linear photo-finishing routines (e.g., tone-curve manipulation);

31

(A) Stages of the camera imaging pipeline and associated parameters

1- Reading raw Image

2- Black light subtraction, linearization [Values or 1D LUT]

3- Lens correction [2D Array(s)]

4- Demosaicing [Func]

5- Noise reduction [Func]

6- White-balancing & colour space [MATs]

12- Gamma curve application [1D LUT]

11- Final colour-space conversion [Mat]

10- Tone curve application [1D LUT]

9- Colour manipulation [3D LUT]

8- Exposure curve [EV value or 1D LUT]

7- Hue/Sat map [3D LUT]

(B) Intermediate images for each stage

Figure 3.1: The camera processing pipeline routines accessible by our software platform are shown in (A). Each component is denoted by the type of parameters it takes—for example, scalar values, 2-D arrays, 3×3 matrices [MAT], function calls [func], or 1-D or 3-D LUTs [LUT]. In addition, the software platform API supports direct access and manipulation to the intermediate images at each stage as shown in (B).

however, researchers are generally forced to apply NR on the final non-linear sRGB image due to a lack of access to the camera pipeline. This presents a significant mismatch between assumptions made in the academic literature and real industry practice.

3.2 Platform Overview

Our platform uses images that are saved in the DNG format. While this format is not yet supported by many of the DSLR cameras, it is currently being supported by the

newer Android phones that implement the Camera 2 API. With Android's adoption of DNG, the number of raw-RGB images captured by mobile devices is expected to increase significantly. However, in the event that images are not captured in DNG, camera-specific raw-RGB formats can be converted to DNG using the Adobe DNG conversion software tool (Adobe, 2018). The DNG image format not only contains the raw-RGB image data but also contains metadata that specifies parameters (e.g., scalar values or a 1D or 3D LUT) intended to be used by different stages in the processing pipeline.

Our platform is made possible by rewriting the interface of the open source Adobe DNG SDK software (Adobe, 2012) that provides a full software implementation of a camera pipeline to convert the DNG raw-RGB image to its final sRGB output. While this is an engineering feat, the implementation is non-trivial. The stand-alone Adobe DNG SDK is not designed to allow changes to the parameters of the individual stages; instead the SDK uses the values in the DNG files, metadata directly. Thus the processing pipeline had to be decomposed into its individual stages and API calls designed to access and modify the underlying parameters. In addition, the unmodified SDK uses a multi-threaded design that breaks the image into a number of small tiles and processes them separately. This makes it difficult to access coherent intermediate images in the pipeline using the native SDK. Our modification changes this tiling structure to allow access to the intermediate image at each stage. We have also added API calls to allow customized demosaicing and noise reduction, which are not supported in the native SDK.

Figure 3.1-(A) overviews the processing steps that are available in the proposed camera imaging platform. The top shows the steps with the associated parameters

used by each of the components, while Figure 3.1-(B) shows the intermediate images at each stage in the pipeline. In the following, we detail each stage and its associated parameters that can be modified. The types of parameters used by the individual stages are also discussed. In the case of a 1D LUT, the same LUT is applied to each colour channel individually.

Stage 1: Reading the raw-RGB image (`Params:` *None*). The unmodified raw-RGB image is read from the DNG image file. This is the unprocessed image produced by the sensor that is still in its mosaiced Bayer pattern format.

Stage 2: Black light subtraction and linearization (`Params:` *Level values* or *1D LUT*). The unmodified raw-RGB image is linearized such that its values range from [0-1] in the processing pipeline. Many cameras provide a *BlackLevel* parameter that represents the black level of the sensor that deviates from 0 due to sensor noise. This is often image specific and related to other camera settings, including ISO and gain. An additional *WhiteLevel* (maximum value) can also be specified. If nothing is provided, the `min` and `max` values of all intensities in the image are used to normalize the image. Another alternative is to provide a 1D LUT to perform the linearization. The 1D LUT shown in the Figure 3.1-(A) is from an Nikon D40.

Stage 3: Lens/Flat Field correction (`Params:` $4\times\mathrm{Array}_{N\times M}$). Many cameras provide a spatially varying correction that compensates for lens distortion and uneven light fall. For example, the Motorola Nexus 6 provides four (one for each Bayer pattern pixel where G is repeated twice) scene-dependent 13×17 2D arrays that are used to provide this flat field correction. These arrays are scaled and bilinearly interpolated to the image size, and then multiplied to the mosaiced image.

Stage 4: Demosaicing (`Params: func`). The demosiacing step converts the single-channel raw-RGB image to three full-size R/G/B colour channels by interpolating the missing values in the Bayer pattern. We denote this operation as an arbitrary function, func. The default interpolation is a standard bilinear interpolation based on the Bayer pattern layout.

Stage 5: Noise reduction (`Params: func`). Similar to the demosaicing stage, noise reduction is denoted as an arbitrary function, func. This function (not provided in the Adobe SDK) has access to the intermediate image and returns a filtered image to the pipeline.

Stage 6: White-balancing and colour space conversion (`Params: Two 3×3 matrices`). This stage performs the necessary colour space conversion between the camera-specific RGB colour space and a standard colour space (e.g., CIE XYZ or ProPhoto RGB). This colourimetric procedure involves a 3×3 white-balance matrix (generally a diagonal matrix) and a 3×3 colour space transformation matrix. The default colour space used by the Adobe SDK is the ProPhoto RGB, which is a wide gamut colour space commonly used for photographic colour manipulation.

Stage 7: Hue/Sat map application (`Params: 3D LUT`). This optional procedure is intended to be part of the colour space conversion to allow a non-linear transformation to be incorporated to improve the colour rendition. While this is referred to as a 'hue' and 'saturation' modification, it is implemented as a 3D LUT applied directly to the RGB values obtained in Stage 6. For example, when saving a DNG file using the X-Rite camera calibration software (X-Rite, 2010), X-Rite adds a $6 \times 6 \times 3$ LUT to the DNG metadata. From our experience, most cameras DNG files do not include this step.

Stage 8: Exposure compensation (`Params:` *EV value*, 1D LUT). The exposure compensation is a digital exposure adjustment. While the input is given as an exposure value (EV) that is used to control shutter and aperture settings on a camera, in the digital case, this simply applies a linear gain (either up or down) to the intensities values. The EV value passed as a parameter will generate a 1D LUT with 4096 values. Alternatively, a 1D LUT can be provided directly.

Stage 9: Colour manipulation (`Params:` 3D LUT). Cameras often apply their own proprietary colour manipulation that is linked to different picture styles on the camera (Kim et al., 2012). Like the Hue/Sat map, this is applied as a 3D LUT where RGB values are interpolated based on the table's entries. The size of this table can be arbitrary—for example, images saved using Nikon D40's Camera Vivid setting have a $36 \times 16 \times 16$ LUT added in the DNG metadata.

Stage 10: Tone-curve application (`Params:` 1D LUT). A camera-specific tone-map can be specified. This is part of the photo-finishing process on board the camera. For example, the Nikon D40's Camera Vivid profile includes a LUT with 248 entries. If no tone-curve is specified, the Adobe DNG has a default tone-curve that is shown in Figure 3.1.

Stage 11: Final colour space conversion (`Params:` 3×3 Matrix). This colour space conversion converts the internal camera working colour space into the final output-referred colour space. This is done using a 3×3 matrix and is assumed to be related to colour space used at stage 6. The most common colour space for cameras is the sRGB and Adobe RGB. In this chapter, the sRGB colour space is used for all examples.

Stage 12: Gamma curve application (`Params:` 1D LUT). The final stage is a gamma curve that is applied as a 1D LUT with 4096 entries. This is intended to

represent the sRGB gamma correction that is part of the sRGB specification; however, it can also be used for additional colour modification and photo-finishing.

These twelve steps make up the collective stages that can be controlled via API calls or direct image modification to intermediate images. Access to this suite of components provides a wide means for manipulating the image from the input raw-RGB to its final sRGB output. Note that it is not necessary that all steps be applied. For example, exposure compensation, noise reduction, hue/sat map modification, and so forth can be skipped as desired.

3.3   Results

We have developed a fully functioning software platform for use on a Windows-based PC. The software framework is developed in C++; however, it has also been modified such that API calls can be made directly from Matlab. In this section, we demonstrate several examples that serve to illustrate the various benefits of our platform. The examples are divided loosely into three categories: (1) basic functionality; (2) evaluating stages at certain points in the pipeline; and (3) evaluating stages within the proper context of the full pipeline. Specifically, Section 3.3.1 demonstrates several examples that show the basic ability to manipulate the pipeline components (e.g., EV levels, tone-curve modification, and demosiacing). Section 3.3.2 provides an example to evaluate the colour conversion stage, a task that is currently difficult to do with existing tools. Section 3.3.3 provides examples targeting white balance, image de-noising, and image blurring that show the benefits of considering these tasks within the full pipeline.

(A) Different EVs applied to stage 8 (exposure compensation)

(B) Different 1D LUT applied to stage 10 (tone-curve application)
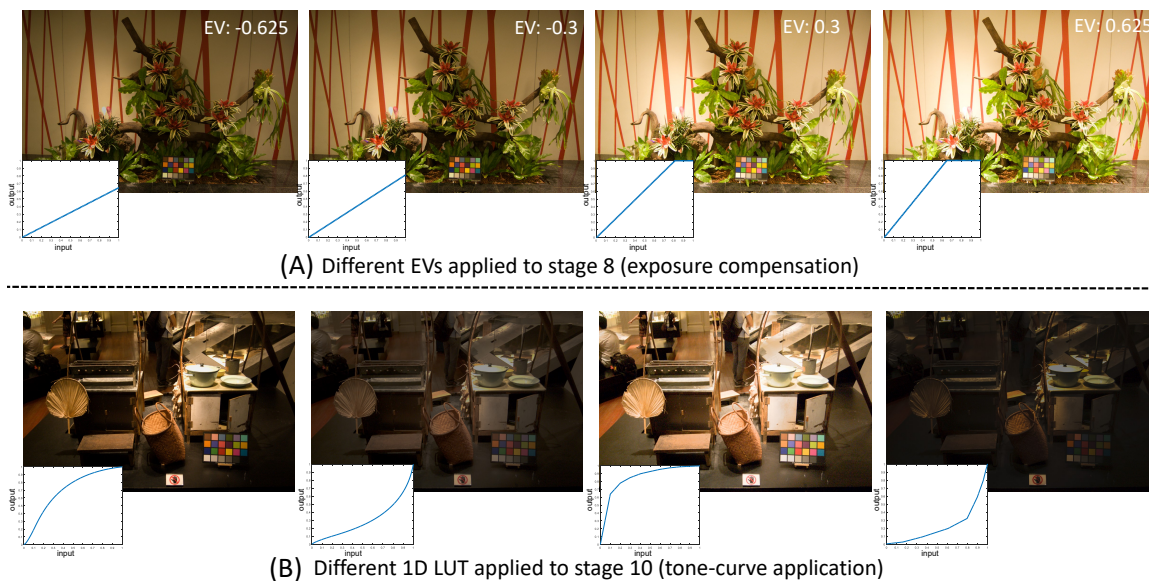
Figure 3.2: (A) Examples of applying different EV and 1D LUTs for exposure compensation (stage 8) and (B) tone-mapping (stage 10).

3.3.1 Basic Processing

Exposure compensation and tone-mapping Figure 3.2 starts with a simple example showing the effects of manipulating parameters for the exposure compensation and tone-mapping stages. Figure 3.2-(A) shows a number of EV values that are passed directly to our platform's API which generates the 1D LUT shown. In the case of the tone-mapping, the 1D LUTs are directly passed to the API as shown in Figure 3.2-(B). The images shown represent the final sRGB output obtained using these parameters in the full camera pipeline.

Demosaicing Figure 3.3-(A,B) demonstrates examples of two different demosaicing procedures applied to an image. In particular, we use the default bilinear interpolation and the work by Gunturk, Altunbasak, and Mersereau (2002) that uses alternating projections. Interestingly, in the work by Gunturk et al. (2002), the results were

(A)   Demosaicing results   (B)

Bilinear interpolation   Alternating projections (Gunturk et al., 2002)
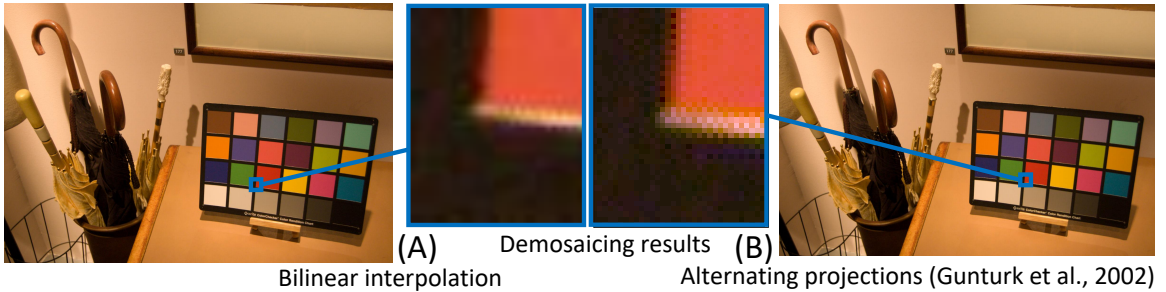
Figure 3.3: Demonstrating the results of two different demosiacing algorithms, in particular (A) bilinear interpolation and (B) Gunturk, Altunbasak, and Mersereau (2002). In this example, the intermediate image passed to stage 4 (demosiacing) is modified using Gunturk, Altunbasak, and Mersereau (2002) and inserted back into the imaging pipeline to obtain the final sRGB output.

demonstrated by simulating a mosaiced image by using an sRGB image and arranging its colours into a Bayer pattern structure. In our example, their approach is applied directly to a real mosaiced raw-RGB image and then returned back to the pipeline to produce a realistic result.

3.3.2   Evaluation of Components

Colourimetry example   One challenge for existing computer vision and image processing research is the ability to obtain intermediate images in the camera pipeline to evaluate the effectiveness at individual stages. An excellent example of this is the colour conversion component (stage 6). This stage is crucial in making sure that different camera-specific colour spaces align to the same canonical colour space after colour conversion. Examining this stage in the camera pipeline is essentially evaluating the quality of the colourimetric calibration of the camera.

To demonstrate our platform's ability to assist with this task, we captured standard colour rendition charts with four different mobile cameras (LG-G4, Motorola Nexus 6, Samsung S6-Edge, and an HTC One M9) under different illuminations.
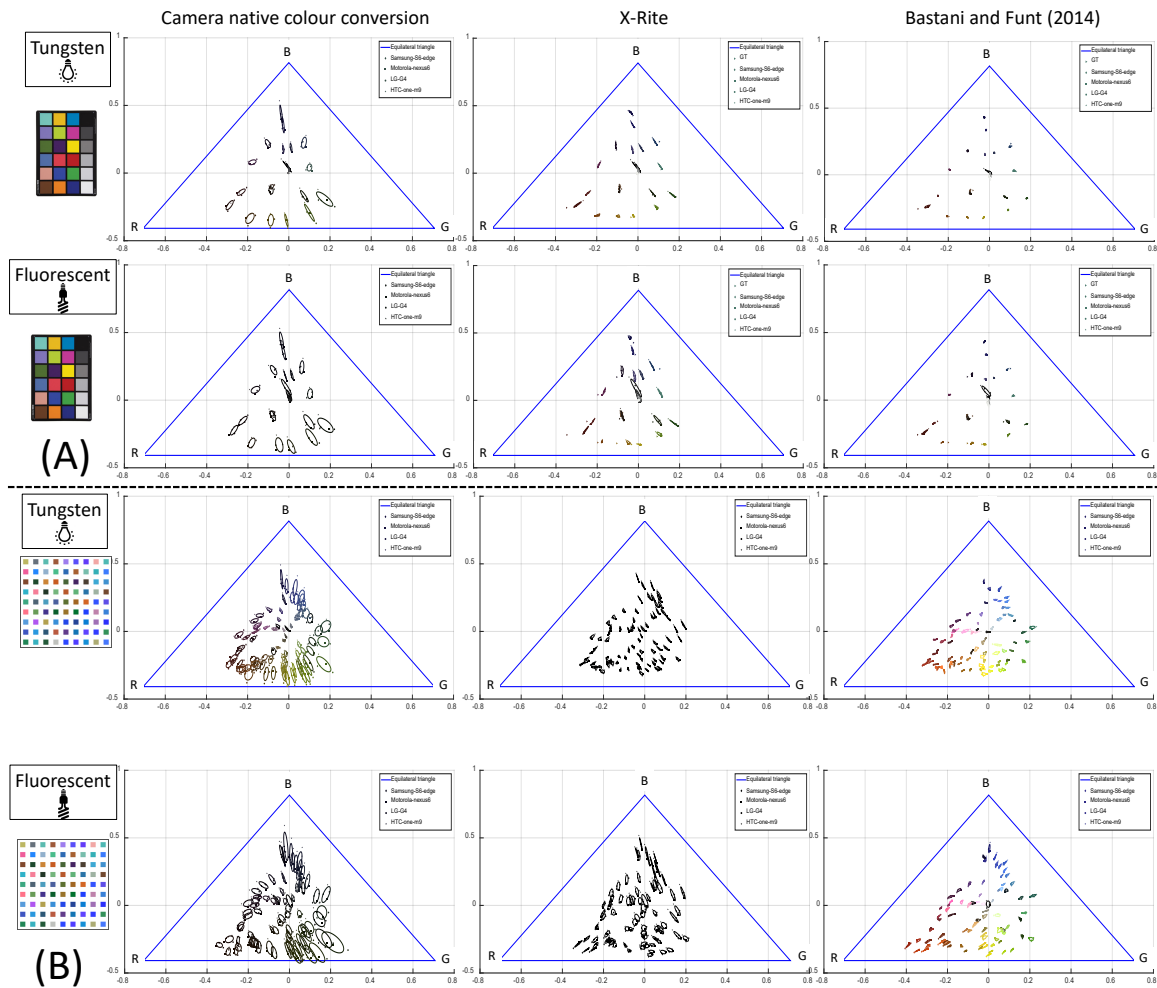
Figure 3.4: This figure demonstrates the ability to evaluate different colour space conversion (stage 6) methods applied to four cameras. See Section 3.3.2 for details.

These mobile device cameras all support the DNG file format and have embedded in their DNG metadata the camera's onboard parameters for this colour conversion. This allows us to compare the results of the native camera's colourimetric ability with two other approaches: (1) the widely used X-Rite calibration software (X-Rite, 2010), and (2) a recent method by Bastani and Funt (2014).

We use the 24-patch colour rendition chart to calibrate the colour space conversion parameters using the X-Rite software and the method by Bastani and Funt (2014). In this case, the colour space conversion is computed from the camera raw-RGB colour space to the ProPhoto colour space. For all methods, including the camera native, the white-balance matrix is estimated using the neutral colours on the colour chart under a specific illumination. In order to compare these three methods, we need to apply the 3×3 colour conversion matrices to stage 6. In the case of X-Rite, we also apply the additional hue/sat map (stage 7) that is used by X-Rite to provide a further non-linear correction for the colour space mapping. The colour pipeline is stopped at the appropriate location for each method and the intermediate image is obtained and the colour patches' average chromaticity values are compared.

The results are shown in Figure 3.4-(A) for colour rendition charts captured under two different illuminations. The plots show the average chromaticity of the 24 colour patches from the four different cameras. Under ideal colourimetric mapping, the chromaticity values should all lie at the exact same location in the chromaticity plot; however, due to errors in the colour conversion matrices, they are not the same. To help with the visualization, we fit a Gaussian ellipsoid to show the spread for each colour patch among the four cameras. Our experiment shows that the method by Bastani and Funt (2014) provides the most consistent colour space mapping.

Error	Incandescent			Fluorescent			Outdoor		
	CC	AM#1	AM#2	CC	AM#1	AM#2	CC	AM#1	AM#2
Camera	0.84	1.67	3.43	0.97	2.37	4.49	0.68	1.80	3.01
X-Rite	0.33	1.06	1.57	0.48	1.68	2.72	0.15	0.74	0.95
Bastani and Funt (2014)	**0.14**	**0.75**	**1.07**	**0.25**	**1.16**	**1.38**	**0.12**	**0.70**	**0.70**

Table 3.1: The table shows the comparisons of error between native cameras, X-Rite's, and Bastani and Funt (2014) colour calibration (CC: colour chart, AM: additional materials).

For the example in Figure 3.4-(A), X-Rite and Bastani and Funt (2014) have an unfair advantage as they were calibrated using the same 24-colour rendition chart that is used to show the results. To test these methods' ability on additional materials, we use their estimated colour space conversion parameters on a new set of colour patches consisting of 81 different types of materials. This is shown in Figure 3.4-(B), where again we see the work by Bastani and Funt (2014) obtain the best results.

We further evaluated these three methods by computing quantitative errors with respect to the ground truth colour values of the colour rendition chart (X-Rite, 1976) in the ProPhoto RGB colour space. In this case, we follow the procedure common in colour research and consider the angular error $\epsilon_{angle}(e_{colour})$ of a colour e_{colour} from the ground truth colour e_{gt} is computed as follows:

$$\epsilon_{angle}(e_{colour}) = cos^{-1}\left(\frac{e_{colour} \cdot e_{gt}}{||e_{colour}|| \; ||e_{gt}||}\right). \tag{3.1}$$

Table 3.1 shows the angular errors for the colour chart (CC) and the additional 81 materials (AM). As demonstrated in the plots in Figure 3.4-(A,B), the method by Bastani and Funt (2014) provides the best results. This type of analysis is challenging without the support of our platform.

3.3.3 Evaluating Tasks Within the Full Pipeline

In the following, we show several tasks that benefit from having access to the full processing pipeline.

White-balancing/colour constancy One of the key processing steps applied to virtually all images is white-balancing. This procedure falls into the larger research of colour constancy that mimics our human perceptual ability to perceive materials under different illuminations as the same colour. White balance approximates this ability by attempting to ensure that at least the neutral scene materials appear achromatic in the camera colour space. White balance is applied directly to the camera raw-RGB image while the image values are still in the camera's RGB colour space.

A number of research papers on this topic (e.g., Chakrabarti, Hirakawa, and Zickler, 2012, Cheng et al., 2015b) provide subjective results of their white-balance result directly on the camera raw-RGB images. Such results, however, have little visual meaning as they provide a visual comparison in an non-standard camera-specific colour space. A more appropriate way to evaluate the results subjectively would be to run the white-balanced result through the full camera pipeline to produce a realistic output that would be produced by the camera. Figure 3.5 shows the difference in these two approaches. In particular, two well-known white-balance methods, grey world (Buchsbaum, 1980) and grey edge (van de Weijer et al., 2007), are applied to an input image. Figure 3.5-(A) shows the results on the raw-RGB image and Figure 3.5-(B) shows the sRGB outputs. The sRGB output provides a much more realistic comparison of the two algorithms.

Noise reduction Similar to white-balancing, another research area that is at a disadvantage by not having access to the full camera pipeline is image denoising. Noise

<table>
<tr><td>(A)</td><td>Grey world
(Buchsbaum, 1980)</td><td>Grey edge
(van de Weijer et al., 2007)</td><td>(B)</td><td>Grey world
(Buchsbaum, 1980)</td><td>Grey edge
(van de Weijer et al., 2007)</td></tr>
<tr><td></td><td colspan="2">White balance applied to raw image input</td><td colspan="3">Final sRGB output after processing through the full pipeline</td></tr>
</table>
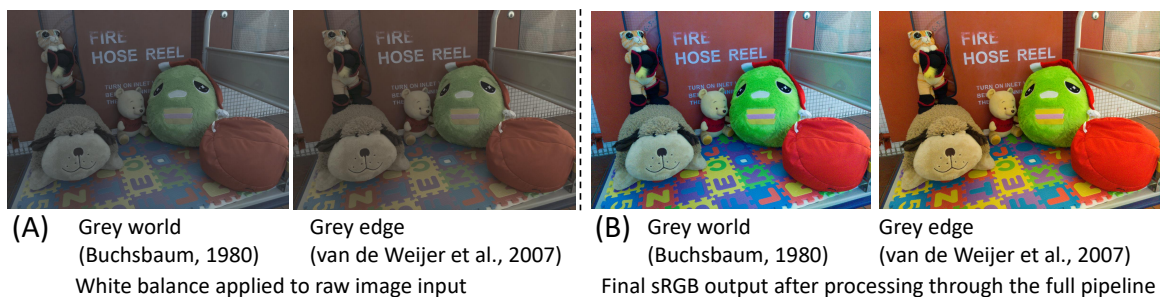
Figure 3.5: (A) White-balancing algorithms applied on raw-RGB images. (B) sRGB output of white-balanced images.

reduction is a well-researched topic and interested readers are referred to Chatterjee and Milanfar (2010) for an excellent overview. One of the major sources of image noise is what is collectively referred to as sensor noise and is attributed to an underlying imaging sensor (CMOS or CCD). Because this noise is present on the sensor, it is present in the raw-RGB image at the start of the pipeline. As such, noise reduction is often applied before the non-linear stages in the camera pipeline. However, since few researchers have access to the camera pipeline, noise reduction methods—for example, the popular BM3D method by Dabov, Foi, Katkovnik, and Egiazarian (2007)—are typically applied and evaluated on the sRGB output. Figure 3.6 demonstrates the disadvantages of applying image denoising outside the proper context of the full imaging pipeline. Our example works from a synthetic image to provide a ground truth input to compute the peak signal to noise ratio (PSNR). Figure 3.6-(A) shows a raw-RGB image that has been corrupted with zero-mean Gaussian noise. The noise profile for three different homogenous patches with increasing intensity values is shown. We can see that this Gaussian noise profile appears uniform over the different patches in the raw-RGB image, with only a shift by the mean intensity. Below this, we show the corresponding noisy raw-RGB image that has been processed through the whole

imaging pipeline, including the non-linear stages (e.g., stage 9, 10, 12). We can see that the noise distribution for different patches is significantly affected by the non-linear processing stages. In fact, it is not possible to make a uniform noise assumption for the sRGB image as the non-linear process has changed this property of the noise.

Figure 3.6-(B) demonstrates BM3D applied at two different places in the pipeline. The top shows BM3D applied to the raw-RGB image. The filtered result is then processed through the remaining pipeline. This represents the proper application of NR. The bottom shows BM3D applied directly to the noisy sRGB image as done in most academic literature. The PSNR for both results in the sRGB final output is computed against the ground truth (noise-free) sRGB image. The PSNRs are drastically different, where the denoising applied at the right place in the pipeline is 58.27, while the application on the final non-linear sRGB stage is only 30.49.

This example serves to motivate the need for having access to the full camera pipeline when examining image denoising. Even in the inevitable case that noise reduction must be applied to the non-linear sRGB image, our software platform provides an excellent means to study sensor noise and how it is affected by the camera processing steps as done in Figure 3.6-(A).

Deblurring Work by Tai et al. (2013) demonstrated that the non-linear processes on the camera have a negative impact on image deblurring. Similar to the denoising example, Tai et al. (2013) showed that the non-linear tone-mapping function (stage 10 and/or 12) changes the distortion profile, in this case the blur's point-spread-function (PSF), such that the PSF was no longer spatially uniform over the image. Tai et al. used this to argue either that deblurring should be applied directly to the raw-RGB image, or that care must be taken to undo any non-linear processing applied to the
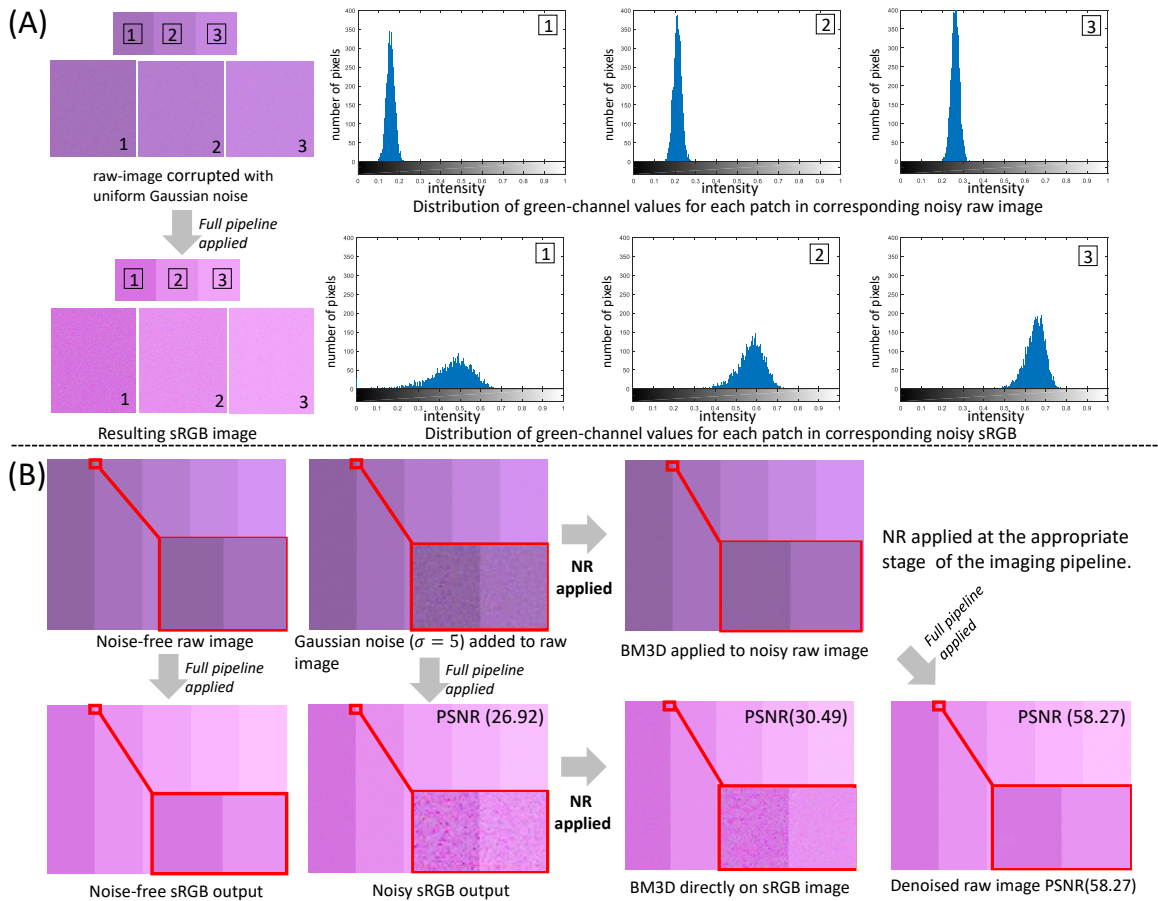
Figure 3.6: (A) Noise profiles for three colour patches for a raw-RGB image and corresponding sRGB image. (B) PSNR comparison of application of BM3D (Dabov, Foi, Katkovnik, & Egiazarian, 2007) on the raw-RGB image at stage 5 and on the final sRGB output.
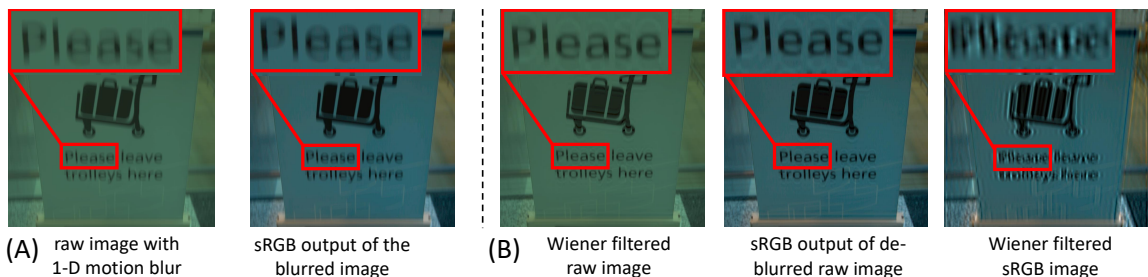
Figure 3.7: (A) Motion blur is applied on the raw-RGB image and then run through the full pipeline to obtain the sRGB output. (B) Deblurring applied to the raw-RGB image and then its output using the camera pipeline, and the results obtained by directly deblurring the blurred sRGB image.

image before deblurring is applied. In that work, Tai et al. used a very simple camera model that applied a single 1D LUT to the raw-RGB images. This is equivalent to applying only stage 1 and 10 in our software platform.

With our platform, we are able to provide a much more realistic demonstration of Tai et al.'s argument. Figure 3.7-(A) shows a raw-RGB image that has been blurred with a PSF modeling linear motion across 50 pixels and its corresponding sRGB output. This is applied directly to a demosaiced raw-RGB image (available at stage 4 in our pipeline). We then apply deconvolution via Wiener filtering (Gonzalez & Woods, 2008) in two manners as shown in Figure 3.7-(B). The first, as advocated by Tai et al. (2013), is on the raw-RGB image that is then processed through the full pipeline. In the second example, we apply the Wiener filter directly to the sRGB image. As expected, the results applied on the raw-RGB image are significantly better than those applied on the sRGB image. As with the prior noise reduction experiment, this example demonstrates the benefit of being able to perform various computer vision and image processing tasks within the appropriate context of the camera pipeline.

3.4 Software Pipeline to Support Imaging Research

Our approach can also be used to support imaging research. For example, Akkaynak and Treibitz (2019) showed how sea water could be removed from underwater images. The method estimates backscatter using dark pixels and their range information. Then, it uses spatially varying illuminant to estimate the range-dependent

47

attenuation. The authors apply their water removal method after stage stage 4 and insert the resulting image back to the pipeline before stage 6. This work serves to motivate the need for having access to the full camera pipeline to support imaging research.

3.5 Discussion

One limitation of our approach is that it can operate only from the captured DNG image saved by the camera. This means camera parameters such as ISO settings that directly affect analog amplification on the sensor hardware or the image's exposure at capture time cannot be modified with our platform. This can impact work targeting tasks, such as high dynamic imaging (e.g., Serrano, Heide, Gutierrez, Wetzstein, and Masia, 2016). For such cases, it will be necessary to capture a number of DNG images with varying ISO or exposure settings to simulate the manipulation on the camera.

We also note that this work represents the current camera architecture where each stage in the pipeline is self-contained. Recent work by Heide et al. (2014) demonstrated the benefits of considering a re-engineered onboard camera processing system that provides a holistic consideration to image formation that is more readily able to incorporate known priors about nature images. We also envision that in the coming years the traditional pipeline described in this chapter will likely see significant changes. Once again, it will be important for researchers to have access to a software platform that allows research to be performed in the proper context of the onboard imaging system.

3.6   Summary

This chapter has presented a new software platform that allows low-level access to the individual components in the camera imaging pipeline. Specifically, our platform leverages the DNG image file format and makes the necessary modifications to the available DNG SDK to provide an extensive API for modifying the parameters of the pipeline, as well as allowing access and modification to intermediate images that can then be inserted back into the pipeline to compute the final output that would be obtained on a camera.

The usefulness of this platform has been demonstrated on a number of examples, including white balance, noise reduction, and colourimetry. While this work is engineering in nature, we believe this platform provides a much needed mechanism for researchers to modify individual components in the pipeline and demonstrate their results within the appropriate context of the full camera imaging pipeline. Furthermore, with the adoption of the DNG raw-RGB image format by the Android OS via the Camera 2 API, the availability of DNG files is going to significantly increase, further adding to the timeliness of this platform for the computer vision community.

Chapter 4

Beyond raw-RGB and sRGB: Advocating Access to a Colourimetric Image State

Using the software camera pipeline described in the previous chapter, we can analyze the intermediate stages to see if an intermediate stage is better than raw-RGB or sRGB in terms of colourimetric consistency. In this chapter we investigate an image stage, consistent across cameras, in the camera imaging pipeline. In particular, most modern cameras allow captured images to be saved in two colour spaces: (1) raw-RGB and (2) sRGB. The raw-RGB image is specific to the colour sensitivities of the sensor's colour filter array. The sRGB image represents a display-referred image that has been rendered through the pipeline. For applications that desire to use a camera for purposes beyond photography, both of the images' colour spaces are undesirable. As we discussed earlier, the colourimetric mapping stage converts the sensor-specific

colours to a device-independent colour space. However, this image state is not accessible. Therefore, we advocate for the ability to access the colourimetric image state and recommend that cameras include the ability to output a third image format that is based on this device-independent colourimetric space. To this end, we perform experiments that show that image values in a colourimetric space are more similar across different makes and models than sRGB and raw-RGB.

4.1   Introduction

As described in Chapters 1 and 2, camera producers apply proprietary photo-finishing operations, such as tone and colour manipulation, to make images look visually pleasing. After the photo-finishing steps the image is finally encoded in an sRGB colour space. Recently, DSLR and mobile phone cameras are capable of saving a raw-RGB format which corresponds to the beginning of the pipeline. The raw-RGB format is useful for many computer vision tasks because of its linear relationship to the scene radiance. However, as we discussed in Section 2.1.4 the raw-RGB values are specific to the sensor being used in the camera. Considering each of the pipeline's steps as changing the state of the image, currently the two output states of the pipeline are inconsistent across cameras.

In this chapter, we advocate for an additional (third) image format to be made available to consumers and application developers. We propose that the internal image state after it has been processed by the colourimetric mapping steps in the camera pipeline is more suitable for applications that assume imaged scene values are consistent among multiple cameras. In the following, we present an analysis of the consistency of colour values at different image states for scene points imaged by
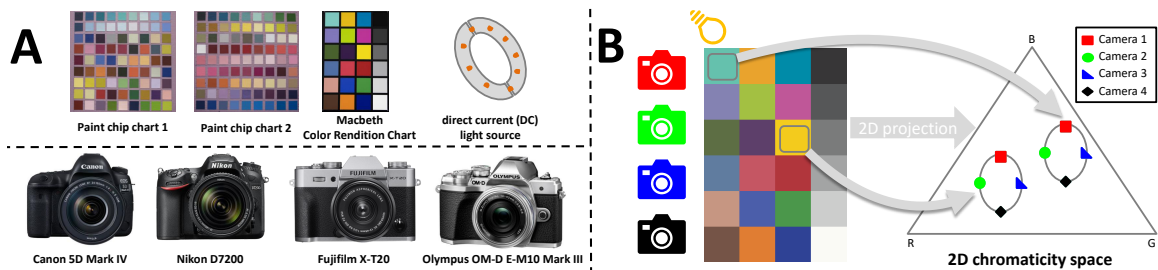
Figure 4.1: (A) Our experiment setup. Images from patches from three different physical patterns (Macbeth Color Rendition Chart, and two custom charts of 81 flat paint samples each) are imaged using four different cameras. A direct current (DC) light source is used to illuminate the scene under different correlated colour temperatures. (B) For each colour patch observed by the four cameras, we compare their values in a 2D chromaticity space. In particular, we are interested in the variance of these measures in different internal camera pipeline image states.

multiple DSLR cameras.

4.2 Experimental Setup and Analysis

As described in the previous section, the goal of this chapter is to demonstrate that providing access to image values after the colourimetric image conversion is the most suitable way to provide consistent image values among different cameras.

Towards this goal, we perform the following experiments illustrated in Figure 4.1. The following four DSLR cameras are used to capture scenes under various lighting conditions: (1) a Canon 5D Mark IV, (2) a Nikon D7200, (3) a Fujifilm X-T20, and (4) an Olympus OM-D E-M10. These cameras are placed in a controlled environment where each camera captures three types of planar scenes with different physical materials. Specifically, we image a Macbeth Color Rendition Chart (with 24 colour patches), and two charts of 81 paint samples that we have created, denoted as Paint

Samples #1 and Paint Samples #2. The scenes are illuminated by a tunable direct current (DC) lighting rig. We use illuminations that correspond to correlated colour temperatures (CCTs) of 3200°K, 4340°K, and 5500°K.

Each scene and lighting condition are captured with each camera and saved in a raw-RGB image format. For these particular DSLR cameras, the raw-RGB image format embeds colourimetric conversion metadata that provides the parameters for the linear transformation and nonlinear 3D LUTs. Using the software camera pipeline from Karaimer and Brown (2016) we are able to render each state of the imaging pipeline. To avoid errors due to each camera's proprietary auto-white-balance functions, we perform the white-balance operation based on the neutral colour patches in the Macbeth Color Rendition Chart. This imaging setup is shown in Figure 4.1-(A).

In order to assess the consistency among the cameras, we map each observed patch's RGB values to 2D chromaticity space as shown in Figure 4.1-(B). We use the 2D chromaticity space to factor out differences related to overall image brightness among the cameras due to issues such as image gain and differences in exposure. We compute the mean variance of each patch in the 2D chromaticity space for the following image states: (1) raw-RGB, (2) raw-RGB with white balance applied, (3) raw-RGB with the linear colourimetric operation applied, and (4) raw-RGB with the additional non-linear (3D LUT) colourimetric operation applied.

The results for the Macbeth Color Rendition Chart, Paint Samples #1, and Paint Samples # 2, are shown in Table 4.1. The table summarizes the results for each CCT as well as the overall result where all illumination conditions are combined. It is clear that the full colourimetric conversion image state (raw-RGB with linear and non-linear colourimetric conversion applied) is by far the most consistent state.

(1) Colour Rendition Chart				
Stage	3200°K	4340°K	5500°K	Combined
raw-RGB	0.046	0.049	0.051	0.130
white-balanced raw-RGB	0.018	0.017	0.018	0.016
colourimetric image state (linear)	0.125	0.117	0.103	0.095
colourimetric image state (non-linear)	**0.004**	**0.004**	**0.003**	**0.004**
sRGB	0.041	0.041	0.042	0.037

(2) Paint Samples #1				
Stage	3200°K	4340°K	5500°K	Combined
raw-RGB	0.121	0.131	0.139	0.411
white-balanced raw-RGB	0.031	0.028	0.031	0.029
colourimetric image state (linear)	0.213	0.174	0.147	0.148
colourimetric image state (non-linear)	**0.006**	**0.004**	**0.005**	**0.009**
sRGB	0.042	0.029	0.038	0.041

(3) Paint Samples #2				
Stage	3200°K	4340°K	5500°K	Combined
raw-RGB	0.127	0.135	0.143	0.424
white-balanced raw-RGB	0.034	0.033	0.035	0.031
colourimetric image state (linear)	0.218	0.177	0.155	0.152
colourimetric image state (non-linear)	**0.004**	**0.004**	**0.005**	**0.007**
sRGB	0.045	0.049	0.056	0.051

Table 4.1: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches from (1) the Macbeth Color Rendition Chart, (2) our first custom chart with 81 paint samples, (3) our second custom chart with 81 different paint samples. The table reports the average variance of each patch's 2D chromaticity values among the four cameras in the respective image states. The most consistent method is highlighted in bold. Results are shown for each illumination CCT and all illuminations combined. To avoid errors due to each camera's proprietary auto-white-balance functions, we perform the white-balance operation based on the neutral colour patches in the Macbeth Color Rendition Chart.

Figures 4.2–4.4 show the corresponding plots in chromaticity space for the three imaged charts. Here, each cluster of points represents the same patch imaged by the four cameras. An ellipse is shown to show the spread of the measurements. From these plots, we can also see that the full colourimetric conversion image state has the minimum spread (i.e., lowest variance among the cameras).

Failure cases due to proprietary auto-white-balance functions Table 4.1 shows the results using the white-balance operation based on the neutral colour
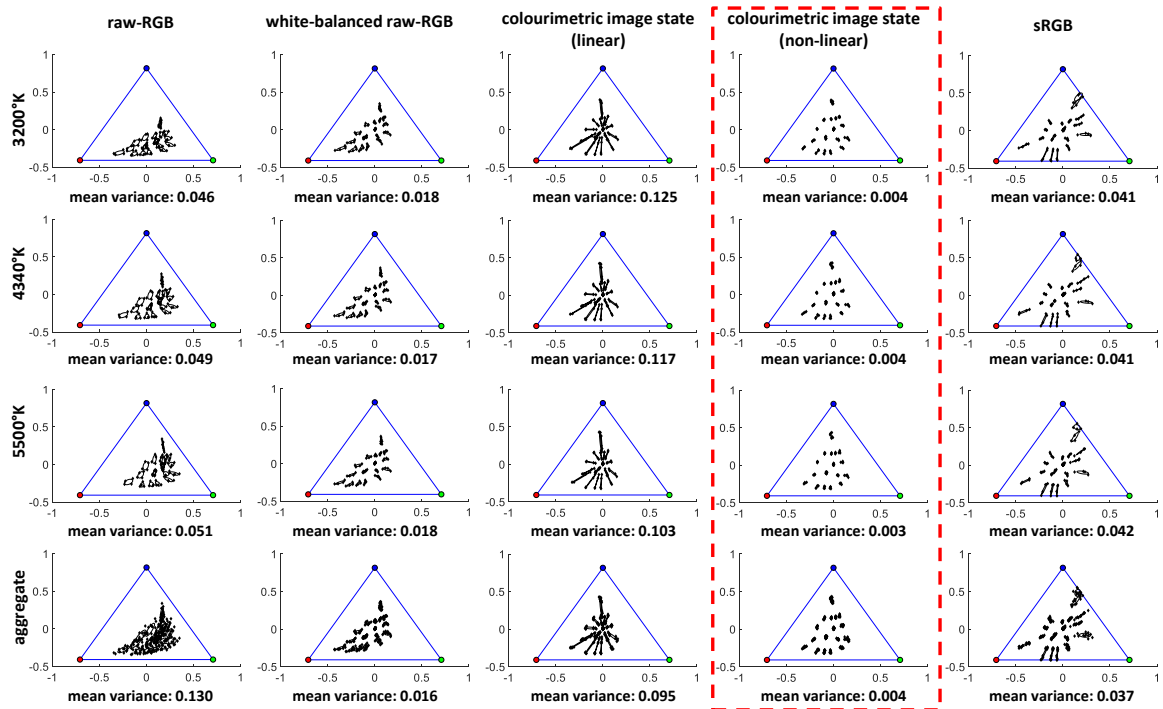
Figure 4.2: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches of the Macbeth Color Rendition Chart. It is clear that the full colourimetric conversion image state (raw-RGB with linear and non-linear colourimetric conversion applied) is by far the most consistent state. Results are shown for each illumination CCT and all illuminations combined. To avoid errors due to each camera's proprietary auto-white-balance functions, we perform the white-balance operation based on the neutral colour patches in the Macbeth Color Rendition Chart.
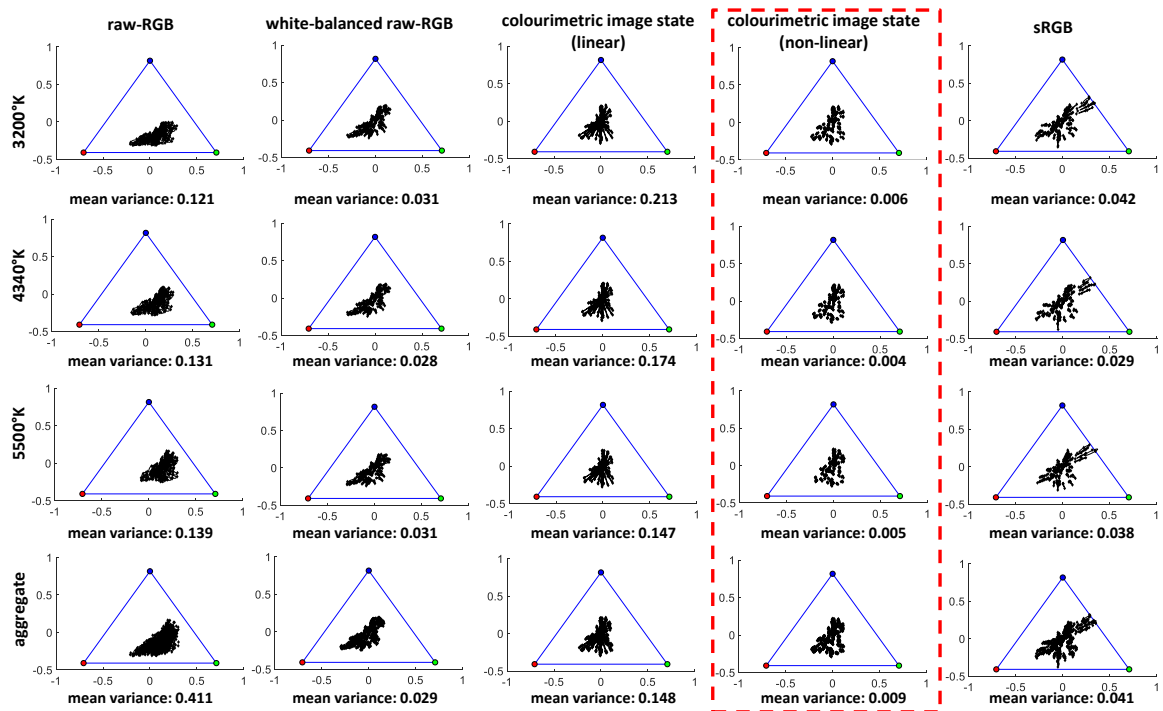
Figure 4.3: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches of the 81 Paint Samples #1. It is clear that the full colourimetric conversion image state (raw-RGB with linear and non-linear colourimetric conversion applied) is by far the most consistent state. Results are shown for each illumination CCT and all illuminations combined. To avoid errors due to each camera's proprietary auto-white-balance functions, we perform the white-balance operation based on the neutral colour patches in the Macbeth Color Rendition Chart.
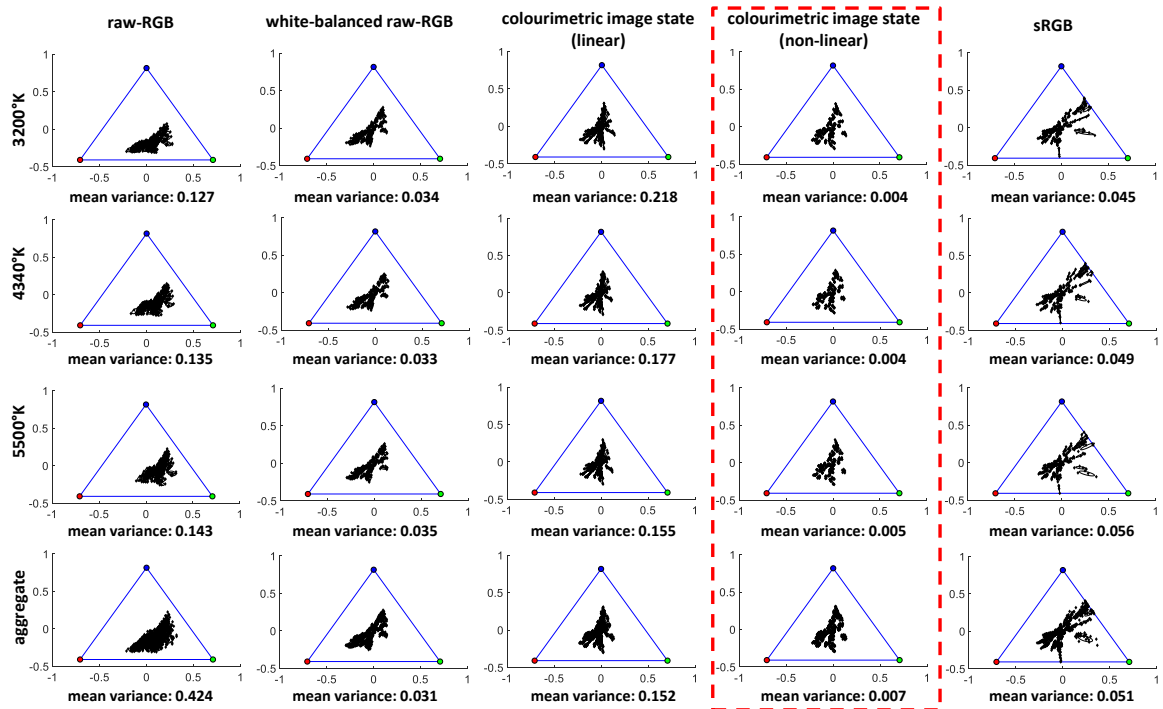
Figure 4.4: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches of the 81 Paint Samples #2. It is clear that the full colourimetric conversion image state (raw-RGB with linear and non-linear colourimetric conversion applied) is by far the most consistent state. Results are shown for each illumination CCT and all illuminations combined. To avoid errors due to each camera's proprietary auto-white-balance functions, we perform the white-balance operation based on the neutral colour patches in the Macbeth Color Rendition Chart.

(1) Colour Rendition Chart				
Stage	3200°K	4340°K	5500°K	Combined
raw-RGB	0.046	**0.049**	0.051	0.130
white-balanced raw-RGB	0.043	0.094	0.050	0.058
colourimetric image state (linear)	0.136	0.160	0.116	0.116
colourimetric image state (non-linear)	**0.026**	0.082	**0.031**	**0.047**
sRGB	0.066	0.097	0.053	0.068

(2) Paint Samples #1				
Stage	3200°K	4340°K	5500°K	Combined
raw-RGB	0.121	**0.131**	0.139	0.411
white-balanced raw-RGB	0.059	0.402	0.039	**0.220**
colourimetric image state (linear)	0.210	0.544	0.161	0.305
colourimetric image state (non-linear)	**0.034**	0.483	**0.011**	0.267
sRGB	0.058	0.480	0.053	0.253

(3) Paint Samples #2				
Stage	3200°K	4340°K	5500°K	Combined
raw-RGB	0.127	**0.135**	0.143	0.424
white-balanced raw-RGB	0.092	0.356	0.103	**0.238**
colourimetric image state (linear)	0.212	0.468	0.238	0.306
colourimetric image state (non-linear)	**0.078**	0.407	**0.079**	0.276
sRGB	0.091	0.383	0.123	0.261

Table 4.2: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches from (1) the Macbeth Color Rendition Chart, (2) our first custom chart with 81 paint samples, (3) our second custom chart with 81 different paint samples. The table reports the average variance of each patch's 2D chromaticity values among the four cameras in the respective image states. The most consistent method is highlighted in bold. Results are shown for each illumination CCT and all illuminations combined using the proprietary auto-white-balance functions of the cameras.

patches in the Macbeth Color Rendition Chart. Here we present how the consistency fails due to the proprietary auto-white-balance functions. Table 4.2 shows the results using the proprietary auto-white-balance functions and Figures 4.5–4.7 show the corresponding plots in chromaticity space for the three imaged charts. In particular, notice how using each camera's propriety white balance function no longer allows for best cross camera colourimetric consistency following non-liinear colourimetric processing.
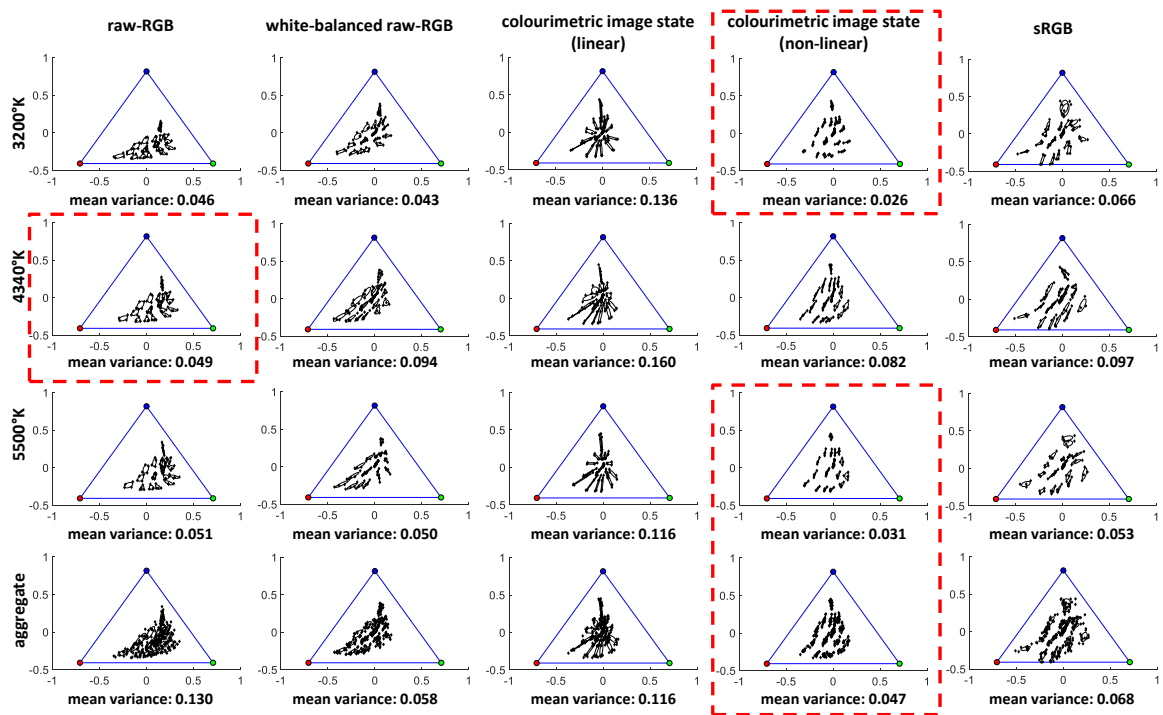
Figure 4.5: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches of the Macbeth Color Rendition Chart. Results are shown for each illumination CCT and all illuminations combined using the proprietary auto-white-balance functions of the cameras.
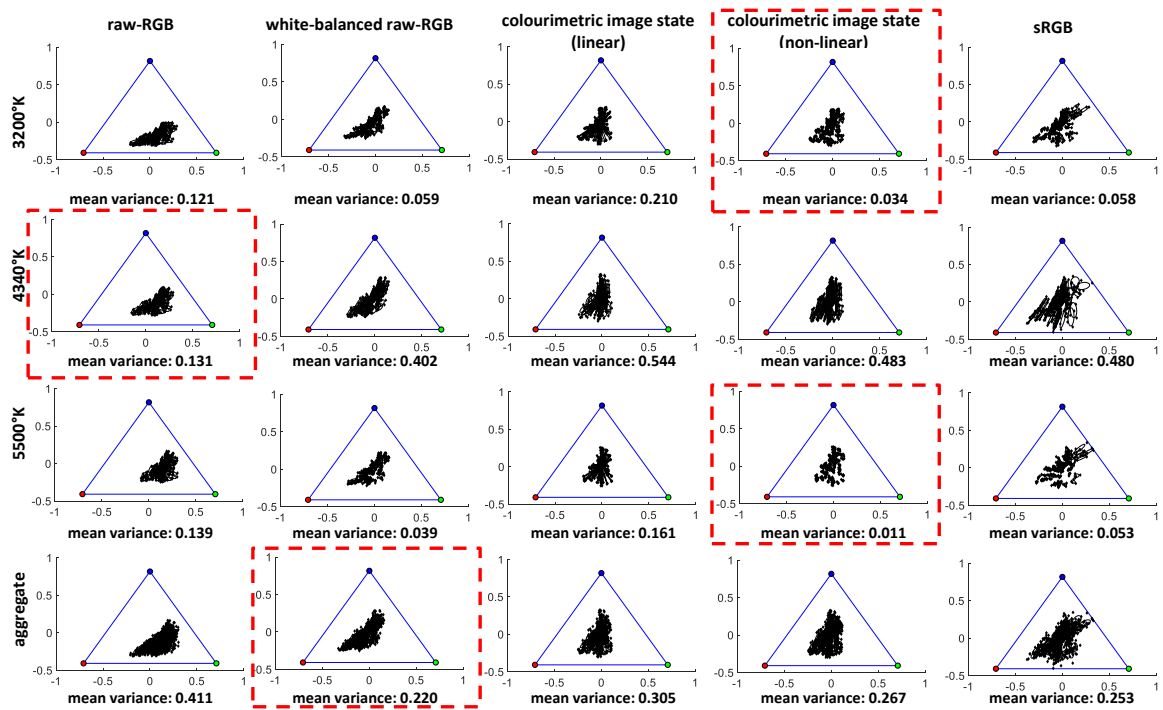
Figure 4.6: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches of the 81 Paint Samples #1. Results are shown for each illumination CCT and all illuminations combined using the proprietary auto-white-balance functions of the cameras.
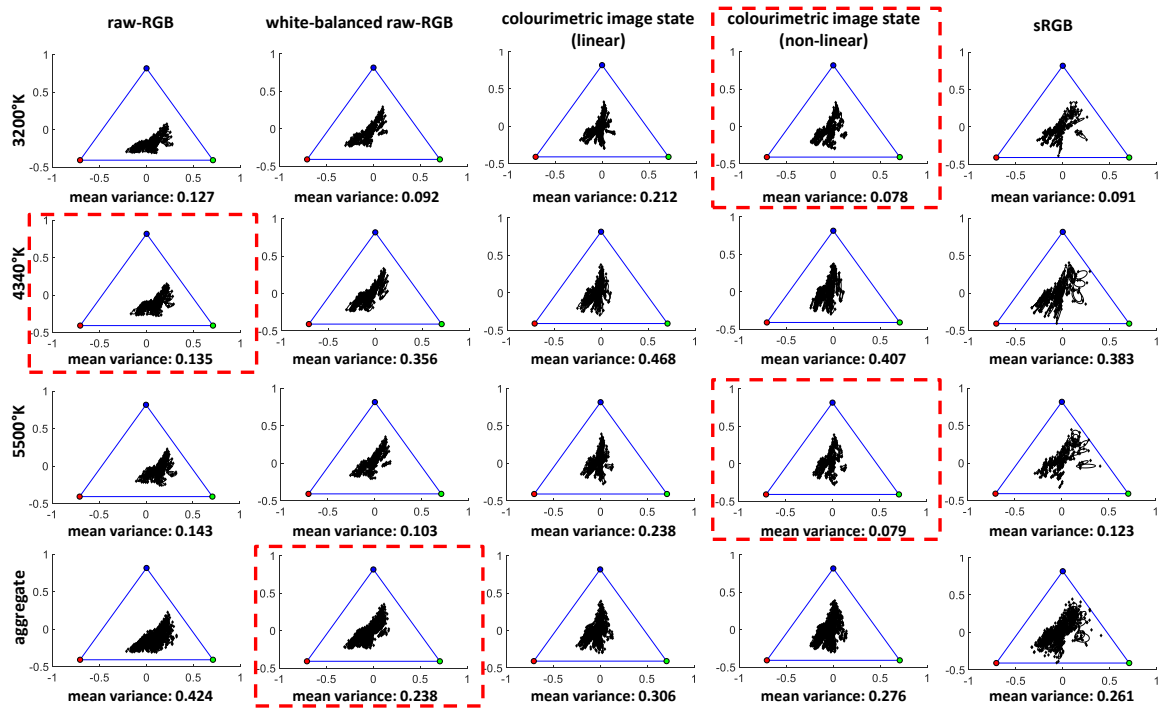
Figure 4.7: Colourimetric consistency between four cameras at five different processing stages of the camera imaging pipeline using the patches of the 81 Paint Samples #2. Results are shown for each illumination CCT and all illuminations combined using the proprietary auto-white-balance functions of the cameras.

4.3 Discussion

We have performed a simple set of experiments to demonstrate the advantages to allowing access to the colourimetric image state in the camera imaging pipeline. Our experiments showed that image values in the colourimetric image state have the least amount of variance among different cameras when compared to other image states. Currently only the raw-RGB and sRGB image states are available to users, which represent the arguably two worst image states for multi-camera consistency. We point out that the white-balanced raw-RGB image state included in our experiments was provided only for comparison; like the colourimetric image state, the white-balance image state is not easily accessible.

It is worth noting that our experiments compared the variance of values in different images states that represent different colour spaces. This is important to consider as variance is not invariant to scaling which undoubtedly happens in the conversion between colour spaces. A more proper evaluation would involve attempting to normalize each colour space into a canonical 3D volume and then estimate the variance. However, such normalization does not represent how image processing and computer vision application developers use such values, nor how the camera pipeline manipulates the image's RGB values. Instead, applications developers would tune their algorithms based on the colour space used, and in such cases, having lower variance among different devices is always preferred.

Finally, our experiments were limited to DSLR cameras, as smartphone cameras currently do not include the nonlinear colourimetric metadata in their raw-RGB files. This lack of metadata made it impossible for us to evaluate the true colourimetric

conversion performed onboard smartphone cameras. This issue with smartphone cameras further bolsters our advocacy for more open access to various image states within the camera image pipeline.

4.4 Summary

Our experiments show that the colourimetric mapping stage achieves the most consistent results. More importantly, we obtain these results without the need for any colourimetric calibration; this is simply a matter of accessing the appropriate image state within the camera imaging pipeline. As a result, we advocate the need for cameras to allow a third image format beyond raw-RGB and sRGB, named a colourimetric image state.

Chapter 5

Improving Colour Reproduction Accuracy on Cameras

In this chapter, we examine the colourimetric mapping procedure currently used on cameras. In particular, we discuss the limitations of the current colourimetric mapping approach and propose two methods that are able to improve colour accuracy. We evaluate our approach on seven different cameras and show improvements of up to 30% (DSLR cameras) and 59% (mobile phone cameras) in terms of colour reproduction error. This work can be categorized as part of the colour calibration method. However, our research focuses on how the colour mapping is performed onboard the camera without the assistance of the standard calibration pattern that needs to be placed in the scene.

5.1 Introduction

Leveraging the experimental platform that we presented in Chapter 3, we focus our attention on the colourimetric mapping procedure. The colourimetric mapping procedure currently used on cameras involves pre-computing two CSTs that correspond to two fixed illuminations. The calibration needed to compute these CSTs is performed in the factory and the transform parameters are part of the camera's firmware. The illuminations that correspond to these calibrated CSTs are selected to be "far apart" in terms of CCT so that they represent sufficiently different illuminations. When an image is captured that is not one of the two illuminations, an image-specific CST is interpolated by linearly blending the two pre-calibrated CSTs. This reliance on an interpolated CST can result in lower overall perceptual colour reproduction accuracy as shown in Figure 5.1. Therefore, providing an improvement for this process allows the camera to act as a colour reproduction, or a colourimetric, device.

Although the colour calibration methods are related to our overall goal of colour reproduction, these methods rely on a colour rendition chart imaged under the scene's illumination to perform the colourimetric mapping. As a result, they compute a direct mapping between the sensor's RGB values and a target perceptual colour space without the need for illumination correction. However, the proposed work in this chapter restricts itself to the current pipeline's two-step procedure involving a camera colour space illumination correction followed by a colour space transform.
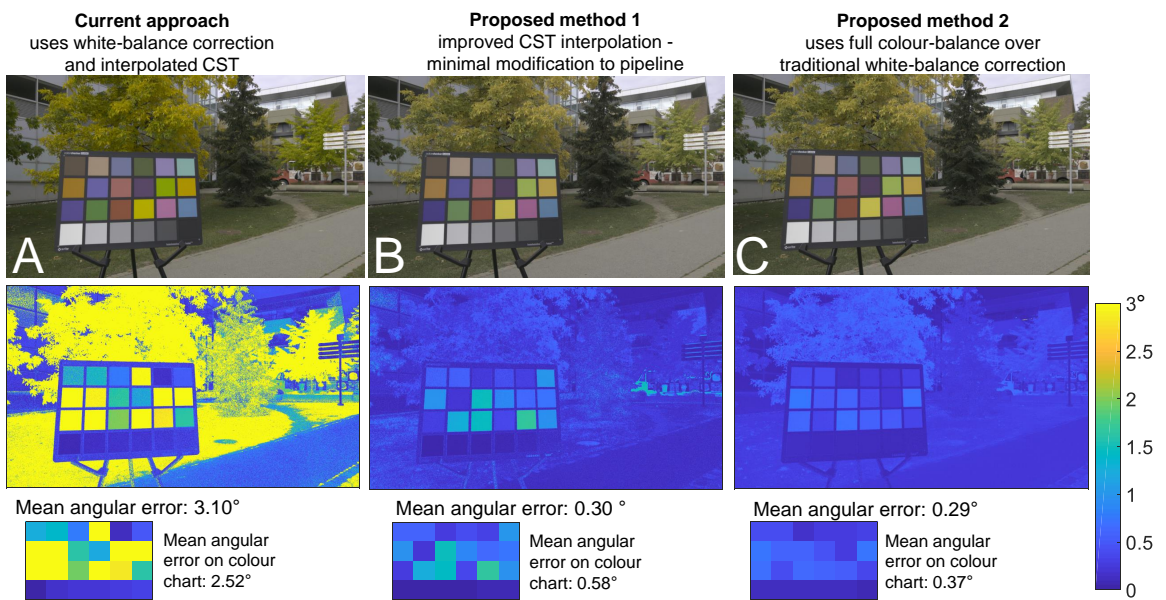
Figure 5.1: (A) Angular colour reproduction error using the current method based on white balance and CST interpolation using two pre-calibrated illuminations. (B) Improvements by our first method using white balance and CST interpolation with three pre-calibrated illuminations. (C) Improvements by our second method using a full colour balance and a fixed CST.
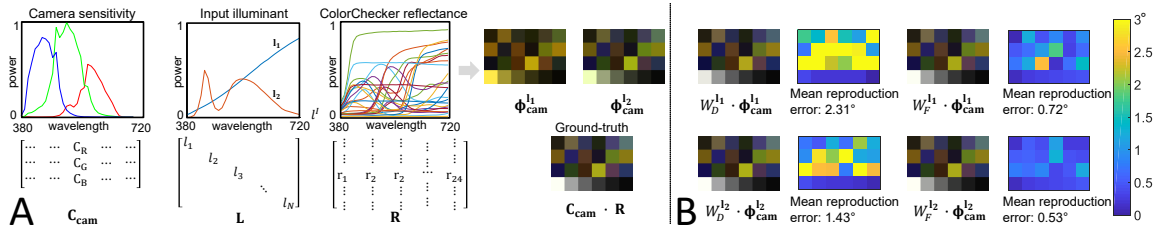
Figure 5.2: This figure is modeled after Cheng, Price, Cohen, and Brown (2015b). (A) Image formation in the camera colour space for two different illumination sources. (B) Result of a conventional diagonal white balance (W_D) and full colour correction (W_F) applied to the input images. Errors are computed as angular reproduction error (see Section 5.4.2). White balance introduces notable errors for non-neutral colours. In addition, errors affect different scene materials depending on the illumination. Full colour balance introduces fewer reproduction errors for all materials.

5.2 Existing Camera Colourimetric Mapping

In this section, we describe the current colourimetric mapping process applied on cameras. We begin by first providing preliminaries of the goal of the colourimetric mapping procedure and the limitations arising from white balance. This is followed by a description of the current interpolation-based method used on cameras.

5.2.1 Preliminaries

Goal of the colourimetric mapping Following the notation of Cheng et al. (2015a), we model image formation using matrix notation. Specifically, let $\mathbf{C_{cam}}$ represent a camera's spectral sensitivity as a $3 \times N$ matrix, where N is the number of spectral samples in the visible range (380 nm to 720 nm). The rows of the matrix $\mathbf{C_{cam}} = [\mathbf{c_R}; \mathbf{c_G}; \mathbf{c_B}]^T$ correspond to the spectral sensitivities of the camera's R, G, and B channels.

Since our work is focused on colour manipulation, we can ignore the spatial location of the materials in the scene. As a result, we represent the scene's materials as a matrix $\mathbf{R}$, where each matrix column, $\mathbf{r}_i$, is a $1 \times N$ vector representing the spectral reflectance of a material. When using this notation, the camera's sensor responses to the scene materials, $\mathbf{\Phi}^{\mathbf{l}}_{cam}$, for a specific illumination, $\mathbf{l}$, can be modeled as:

$$\mathbf{\Phi}^{\mathbf{l}}_{cam} = \mathbf{C_{cam}} \operatorname{diag}(\mathbf{l}) \mathbf{R} = \mathbf{C_{cam}} \mathbf{L} \mathbf{R}, \tag{5.1}$$

where $\mathbf{l}$ is a $1 \times N$ vector representing the spectral illumination and the $\operatorname{diag}(\cdot)$ operator creates a $N \times N$ diagonal matrix from a vector (see Figure 5.2).

The goal of the colourimetric mapping of a camera is to have all the colours in the scene transformed to a perceptual colour space. This target perceptual colour space can be expressed as:

$$\mathbf{\Phi}_{xyz} = \mathbf{C_{xyz}} \mathbf{R}, \tag{5.2}$$

where $\mathbf{C_{xyz}}$ is defined similar to $\mathbf{C_{cam}}$ but uses the perceptual CIE XYZ matching functions (Guild, 1931; Wright, 1928–1929). In addition, the effects of the illumination matrix $\mathbf{L}$ are ignored by assuming the scene is captured under ideal white light—that is, all entries of $\mathbf{l}$ are equal to 1.

The colourimetric mapping problem therefore is to map the camera's $\mathbf{\Phi}^{\mathbf{l}}_{cam}$ under a particular illumination $\mathbf{l}$ to the target perceptual colour space with the illumination corrected—that is, $\mathbf{\Phi}_{xyz}$.

Deficiencies in white balance As previously discussed, the colourimetric mapping on cameras is performed in two steps. The first is to remove the effects of the illumination (i.e., computational colour constancy) in the camera sensor's colour space.
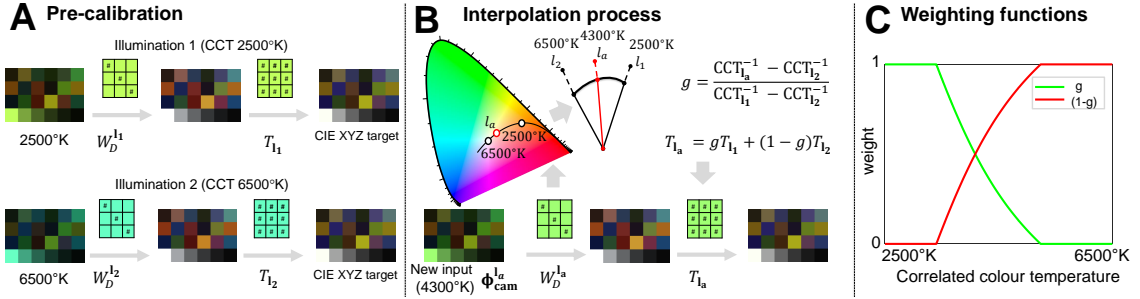
Figure 5.3: (A) Pre-calibrated white balance and CSTs for two illuminations with sufficiently different CCT. (B) Interpolation procedure for an image captured under an arbitrary illumination. (C) A plot of the weights used to interpolate the new CST.

Ideally, computational colour constancy estimates a 3×3 linear transform, W_F, to remove the illumination as follows:

$$W_F = \arg\min_{W_F} ||\mathbf{C_{cam}}\,\mathbf{R} - W_F\,\mathbf{\Phi^l_{cam}}||^2, \tag{5.3}$$

where W_F minimizes the error between *all* the observed scene materials. Here the subscript F is used to denote that this matrix is a full 3×3 matrix.

However, most colour constancy methods target only the estimation of the illumination in the camera space. This is equivalent to observing a neutral object in the scene (i.e., an achromatic material $\mathbf{r}$ that reflects spectral energy at every wavelength equally) and thus $\mathbf{r} = \mathbf{l}$. The matrix to correct neutral patches can be derived directly from the observed illumination $\mathbf{C_{cam}}\,\mathbf{l}$ as:

$$W_D = \mathrm{diag}(\mathbf{C_{cam}}\,\mathbf{l})^{-1}, \tag{5.4}$$

where $\mathbf{l}$ is the observed scene illumination. The subscript D denotes that W_D is

69

restricted to a diagonal 3×3 matrix. It should be clear that this diagonal white-balance correction is not the same as the full colour correction formulation in Eq. 5.3.

The obvious drawback to white balance is that it cannot guarantee that non-neutral scene materials are properly corrected. Moreover, errors in the non-neutral scene materials are dependant on the scene illumination. Figure 5.2 shows an example where the scene materials $\mathbf{R}$ are the spectral properties of colour patches from a standard colour rendition chart. In this figure, the results of using the full colour balance matrix W_F and the white-balance matrix, W_D, are applied to two different illuminations, $\mathbf{l_1}$ and $\mathbf{l_2}$. The errors are plotted as:

$$
\begin{aligned}
\mathrm{Err}_{W_F^{\mathbf{l_i}}} &= ||\mathbf{C_{cam}}\,\mathbf{R} - W_F^{\mathbf{l_i}}\,\mathbf{\Phi}_{cam}^{\mathbf{l_i}}||^2 \\
\mathrm{Err}_{W_D^{\mathbf{l_i}}} &= ||\mathbf{C_{cam}}\,\mathbf{R} - W_D^{\mathbf{l_i}}\,\mathbf{\Phi}_{cam}^{\mathbf{l_i}}||^2,
\end{aligned}
\tag{5.5}
$$

where the index $\mathbf{i}$ is used to denote the different illuminations, $\mathbf{l_1}$ or $\mathbf{l_2}$. In this figure, the $W_F^{\mathbf{l_i}}$ and $W_D^{\mathbf{l_i}}$ are computed for the respective illuminations $\mathbf{l_i}$. We can see that the diagonal matrix incurs notable errors for non-neutral materials. These errors must be considered when computing the subsequent colour space transform.

Colour space transform (CST) Once the white-balance step has been applied, we now need to perform a mapping that considers the differences in the camera sensitivities $\mathbf{C_{cam}}$ and the desired perceptual colour space $\mathbf{C_{xyz}}$. Because the camera sensitivities of the sensor are generally unknown, it is common to derive this transform by imaging an object with known CIE XYZ values, most commonly a colour rendition chart.

Working from Eq. 5.1, we assume the matrix $\mathbf{R}$ captures the spectral properties of the colour rendition chart's patches. Our goal is to compute a 3×3 CST matrix,
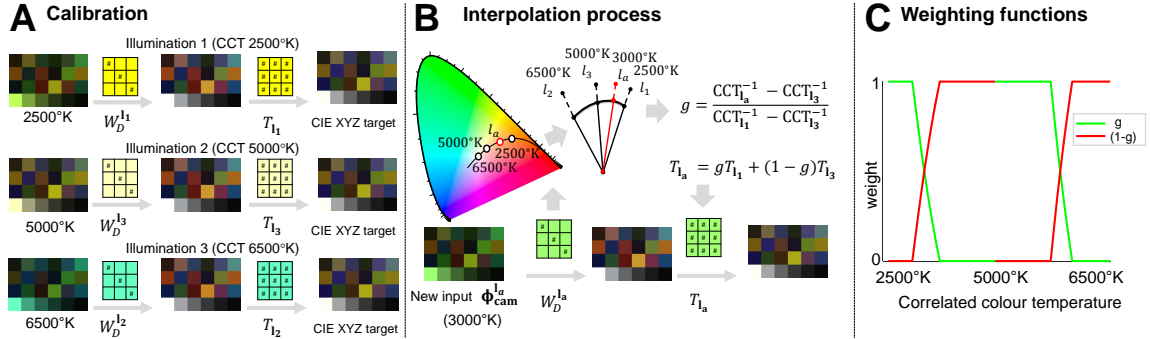
Figure 5.4: Our first method to improve the colourimetric mapping in the in-camera processing pipeline. (A) An additional illumination is calibrated and added to the interpolation process. (B) Shows the overall procedure to compute the weights to interpolate the CST. (C) Shows the weighting function for different estimated CCTs.

$T_{\mathbf{l}}$, that minimizes the following:

$$T_{\mathbf{l}} = \arg\min_{T_{\mathbf{l}}} ||\mathbf{C_{xyz}}\,\mathbf{R} - T_{\mathbf{l}} W_D^{\mathbf{l}}\,\mathbf{\Phi_{cam}^{l}}||^2, \tag{5.6}$$

where $W_D^{\mathbf{l}}$ is the estimated white-balance correction for the given illumination $\mathbf{L}$. Since the results of $W_D^{\mathbf{l}}\,\mathbf{\Phi_{cam}^{l}}$ differ depending on illumination, the matrix $T_{\mathbf{l}}$ needs to be estimated *per illumination* to compensate for the differences.

5.2.2 Current Interpolation Approach

Instead of computing a $T_{\mathbf{l}}$ per illumination, the current procedure used on cameras is to interpolate a CST based on two pre-calibrated illuminations. Figure 5.3 provides a diagram of the overall procedure. The two illuminations are selected such that their CCTs are sufficiently far apart. For each illumination, the illumination-specific CST is estimated as shown in Figure 5.3-(A).

When an image is captured, its estimated illumination value is used to compute the

correlated colour temperature of the illumination (Wyszecki & Stiles, 1982; Robertson, 1968). Based on the correlated colour temperature, the two pre-computed CSTs are interpolated (see Figure 5.3-(B)) to obtain the final CST to be applied as follows:

$$T_{\mathbf{l}} = g \ T_{\mathbf{l_1}} + (1 - g) \ T_{\mathbf{l_2}}, \tag{5.7}$$

where

$$g = \frac{\mathrm{CCT}_{\mathbf{l}}^{-1} - \mathrm{CCT}_{\mathbf{l_2}}^{-1}}{\mathrm{CCT}_{\mathbf{l_1}}^{-1} - \mathrm{CCT}_{\mathbf{l_2}}^{-1}}. \tag{5.8}$$

The factory pre-calibrated CCTs for $\mathbf{l_1}$ and $\mathbf{l_2}$ for most cameras are selected to be 2500°K and 6500°K. The interpolation weights of g and $1 - g$ are shown in Figure 5.3-(C), where the horizontal axis is the CCT of the image's estimated illumination.

As shown in Figure 5.1, this interpolation procedure based on two fixed illuminations does not always provide good results. In the following sections, we describe two methods to improve the colourimetric mapping process.

5.3   Proposed Improvements

We introduce two methods to improve the colourimetric mapping procedure. The first approach is a simple extension of the interpolation method to include an additional calibrated illumination in the interpolation process. The second method relies on a full colour correction matrix discussed in Section 5.2.1 and uses a fixed CST matrix for all input images.

Method 1: Extending interpolation. The most obvious way to improve the current colourimetric mapping procedure is to incorporate additional calibrated illuminations into the interpolation process. Ideally we would like many illuminations to

be represented as control points in this process; however, this is likely unrealistic in a factory pre-calibration setting. As a result, we consider the case of adding only a single additional interpolation control point with a CCT at approximately 5000°K, which is the median CCT of the dataset. Figure 5.4-(A) shows a diagram of our approach.

When a new image is obtained, we estimate the scene illumination and then select a pair of pre-calibrated T_1 based on the estimated illumination's CCT. The blending weights g and $1 - g$ are adjusted accordingly between the pair of T selected using Eq. 5.8. The final CST, T_1, is computed using Eq. 5.7. An example of this approach is shown in Figure 5.4-(B). The weighting factors for g are shown in Figure 5.4-(C). This simple inclusion of a single additional pre-calibrated illumination in the interpolation process gives surprisingly good results and can be readily incorporated into an existing in-camera pipeline.

Method 2: Using full colour balance. As discussed in Section 2.2.2, our second method leverages the full colour-balance approach proposed by Cheng et al. (2015a). A brief overview of the approach is provided here.

Cheng et al. (2015a) found that under certain types of broadband illumination, a diagonal correction matrix was sufficient to correct all colours in the camera's colour space. Based on this finding, they proposed a method that first estimated a set of ground truth colour values in the camera's sensor space by imaging a colourful object under broadband illumination (e.g., sunlight). This image is then corrected using a diagonal white balance. Images of the same object under different illuminations could then map their observed colours to these ground truth camera-specific colours. The benefit of this approach is that unlike traditional colour calibration, this approach
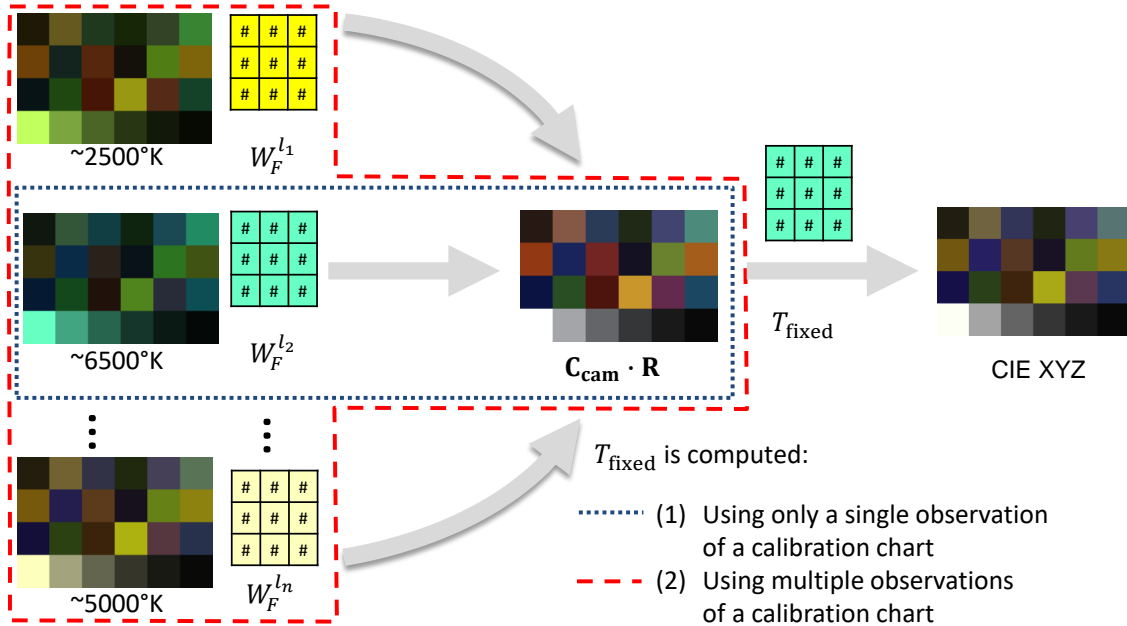
Figure 5.5: Our second method relies on full colour balance matrices estimated using Cheng, Price, Cohen, and Brown, 2015a's approach. A fixed CST is estimated using either a single observation of a calibration pattern or multiple observations.

does not require a colour rendition chart with known CIE XYZ values; instead, any colour pattern with colour variation will work. Consequently, this approach falls short of colourimetric calibration, but does allow full colour balance to be achieved.

Cheng et al. (2015a) also proposed a machine-learning approach that trained a Bayesian classifier to estimate the full colour balance matrix, $W_F^{\mathbf{l}}$, for a given camera image $\mathbf{\Phi_{cam}^l}$ under an arbitrary illumination $\mathbf{l}$. Since the full colour balance attempts to correct all colours, it is not necessary to compute an illumination-specific CST. Instead, we can estimate a single *fixed* CST, T_{fixed} as follows:

$$T_{\text{fixed}} = \arg\min_{T} || \sum_{i} (\mathbf{C_{xyz}}\,\mathbf{R} - T\,W_F^{\mathbf{l_i}}\,\mathbf{\Phi_{cam}^{l_i}})||^2, \tag{5.9}$$

74

where the index **i** selects an image in the dataset, $\mathbf{l_i}$ represents the illumination for that image's scene, and $\mathbf{R}$ is again assumed to be calibration chart patches' spectral responses.

In our second approach, we assume that the full colour balance matrix can be obtained. We estimate the fixed CST, T_{fixed}, in two ways. The first is to use only a single observation of the colour chart. Therefore Eq. 5.9 can be simplified such that i indexes to only a *single* observation of the colour chart with a single illumination (we use an image with CCT of 6500°K). The second approach is to consider all observations of the colour chart for each different illumination. Figure 5.5 provides an illustration of these two methods to estimate T_{fixed}. In our experiments, we distinguish the results obtained with these two different approaches.

Method 2 (extension): Full colour balance with interpolation. While the full colour balance allows the computation of a fixed CST that should be applicable to all illuminations, from Eq. 5.9, it is clear the errors for the estimated CST mapping will be minimized for a particular illumination when only a single i is used as described above. As a result, we can use the same interpolation strategy as described in Sec. 5.2.2 that used white balance corrected images, but instead use full colour balance and CST estimated using Eq. 5.9. Results for this extension approach are not included in the dissertation, but can be found in Appendix A.

5.4   Experimental Results

In this section, our experimental setup used to test the proposed colourimetric mapping methods is discussed along with an evaluation of the proposed methods.

| | Apple iPhone 7 | | Google Pixel | | LG-G4 | | Canon1D | | NikonD40 | | Sony α57 | | Olympus E-PL6 | |
Method	CC	I	CC	I	CC	I	CC	I	CC	I	CC	I	CC	I
CB + Fixed CST (all)	**0.80**	**0.91**	**0.88**	1.05	**0.81**	**0.86**	**0.84**	**0.63**	**0.93**	0.77	1.00	0.77	0.85	0.69
CB + Fixed CST (single)	0.95	1.04	1.41	1.45	1.17	1.17	0.95	0.76	0.98	0.80	**0.99**	0.76	**0.82**	0.67
WB + 3 CSTs	1.36	1.08	1.27	**1.04**	1.55	1.24	1.04	0.70	1.07	**0.65**	1.04	**0.74**	0.86	**0.61**
WB + 2 CSTs (re-calibrated)	1.76	1.42	1.92	1.46	1.98	1.54	1.16	0.82	1.33	0.82	1.13	0.81	1.05	0.71
WB + 2 CSTs (factory)	3.07	2.46	2.28	1.75	2.74	2.04	1.58	1.18	2.02	1.26	1.14	0.79	1.12	0.75

Table 5.1: The table shows the comparisons of error between full colour balance with fixed CST, diagonal matrix correction with three CST, native cameras (re-calibrated for the datasets we use), and native cameras (factory calibration). Errors are computed on colour chart colours only (denoted as CC), and on full images (denoted as I). The top performance is indicated in bold.
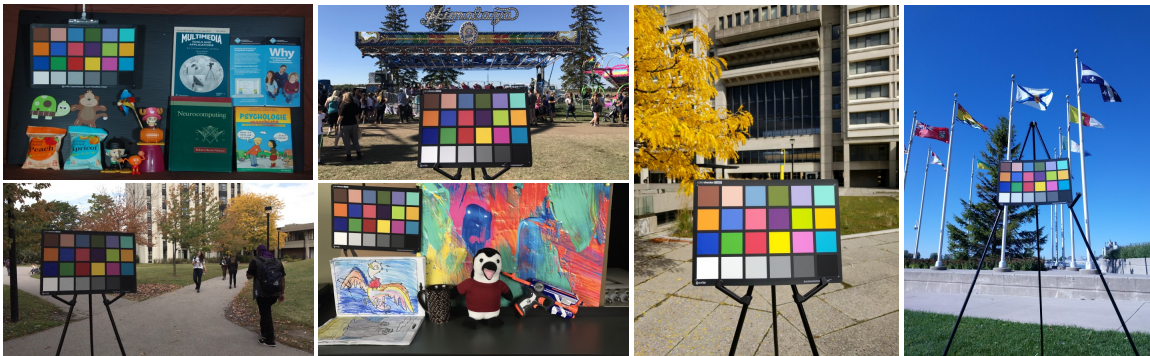
Figure 5.6: Sample images from our dataset, including selected images from the publicly available NUS dataset (Cheng, Prasad, & Brown, 2014) together with our mobile phone camera images.

5.4.1 Setup and Ground Truth Dataset

Our dataset consists of four DSLR cameras (Canon 1D, Nikon D40, Sony α57, and Olympus E-PL6) and three mobile phone cameras (Apple iPhone 7, Google Pixel, and LG-G4). For each camera we generated 100 colourimetrically calibrated images. For the DSLR cameras, we selected images from the NUS dataset (Cheng, Prasad, & Brown, 2014) for calibration. The NUS dataset was created for research targeting colour constancy and provides only ground truth for the illumination. This dataset has over 200 images per camera, where each camera is imaging the same scene. We

selected a subset of this dataset, considering images in which the colour chart is sufficiently close to the camera and fronto-parallel with the respect to the image plane. For the mobile phone cameras, we captured our own images, taking scenes outdoors and in an indoor laboratory setting with multiple illumination sources. All scenes contained a colour rendition chart. Like the NUS dataset, our dataset also carefully positioned the cameras such that they were imaging the same scene. Figure 5.6 shows an example of some of the images from our dataset.

Colourimetric calibration for each image in the dataset is performed using the X-Rite's camera calibration software (X-Rite, 2010) that produces an image-specific colour profile for each image. The X-Rite software computes a scene-specific white-balance correction and CST for the input scene. This is equivalent to estimating Eq. 5.4 and Eq. 5.6 based on CIE XYZ values of the Macbeth Color Rendition Chart.

To obtain the colourimetrically calibrated image values, we used the software platform by Karaimer and Brown (2016) with the X-Rite calibrated colour profiles to process the images. The camera pipeline is stopped after the colourimetric calibration stage, as discussed in Section 2.2. This allows us to obtain the image at the colourimetric conversion stage *without* photo-finishing applied as shown in Figure 1.1.

Note that while the discussion in Section 5.2.1 used CIE XYZ as the target perceptual colour space, cameras instead use the Reference Output Medium Metric (ROMM) (Spaulding, Giorgianni, & Woolfe, 2000) colour space, also known as ProPhoto RGB. ProPhoto RGB is a wide-gamut colour space that is related to CIE XYZ by a linear transform. For our ground truth images, we stopped the camera pipeline after the values were transformed to linear-ProPhoto RGB colour space. Thus, our 700-image dataset provides images in their unprocessed raw-RGB colour

space and their corresponding colourimetric calibrated colour space in ProPhoto RGB.

5.4.2   Evaluation

We compare three approaches: (1) the existing interpolation method based on two calibrated illuminations currently used on cameras; (2) our proposed method 1 using interpolation based on three calibrated illuminations; (3) our proposed method 2 using the full colour balance and a fixed CST.

The approach currently used on cameras is performed in two manners. First, we directly use the camera's factory calibration of the two fixed CSTs. This can be obtained directly from the metadata in the raw-RGB files saved by the camera. To provide a fairer comparison, we use the X-Rite software to build a custom camera colour profile using images taken from the dataset. Specifically, the X-Rite software provides support to generate a camera profile that replaces the factory default CST matrices. This colour profile behaves exactly like the method described in Section 5.2. To calibrate the CSTs, we select two images under different illuminations (approximately 2500°K and 6500°K) from the dataset and use them to build the colour space profile.

We then use the same 2500°K and 6500°K images to calibrate the CSTs used by our proposed method 1. We add an additional image with a CCT of approximately 5000°K as the third illumination. For the interpolation-based methods, we estimate the illumination in the scene using the colour rendition chart's white patches. This avoids any errors that may occur due to improper illuminant estimation. Similarly, for our method 2 that relies on full colour balance, we compute the direct full colour-balance matrix for a given input method based on the approach proposed by Cheng

et al. (2015a). This can be considered Cheng et al.'s method providing optimal performance.

Since each image in our dataset has a colour rendition chart, errors are reported on the entire image as well as the colour patches in the rendition chart. Moreover, since the different approaches (i.e., ground truth calibration and our evaluated methods) may introduce scale modifications in their respective mappings that affect the overall magnitude of RGB values, we do not report absolute RGB pixel errors, but instead report angular reproduction error (Finlayson, Zakizadeh, & Gijsenij, 2017). This error can be expressed as:

$$\epsilon_r = \cos^{-1}\left(\frac{w^r \cdot w^{gt}}{||w^r|| \; ||w^{gt}||}\right), \tag{5.10}$$

where w^r is a linear-ProPhoto RGB value expressed as a vector produced by one of the methods and w^{gt} is its corresponding ground truth linear-ProPhoto RGB value also expressed as a vector.

Individual camera accuracy Table 5.1 shows the overall results for each camera and approaches used. The approaches are labeled as follows. CB+fixed CST refers to our method 2, using full colour balance followed by a fixed CST. The (all) and (single) refer to the estimation of whether the CST uses either all images or a single image as described in Section 5.3. Using white balance (WB), WB+3 CSTs refers to our method 1 using an additional calibrated illumination. WB+2 CSTs refers to the current camera approach, where (recalibrated) indicates this approach is using the X-Rite colour profile described above and (factory) indicates the camera's native CST.

The columns show the different errors computed: (CC) is colour chart patches

Method	Mobile Phones 2900°K	Mobile Phones 4500°K	Mobile Phones 5500°K	Mobile Phones 6000°K	DSLRs 3000°K	DSLRs 3500°K	DSLRs 4300°K	DSLRs 5200°K
CB + Fixed CST (all)	**4.6**	**2.1**	**1.1**	**1.0**	**1.6**	**1.8**	**1.2**	**1.1**
CB + Fixed CST (single)	10.0	7.1	4.9	6.2	2.7	2.9	2.1	1.7
WB + 3 CSTs	37.9	6.9	6.6	8.1	4.4	2.8	2.3	2.1
WB + 2 CSTs (re-calibrated)	44.9	11.3	9.5	16.7	6.7	3.2	3.4	5.9
WB + 2 CSTs (factory)	34.2	38.2	32.6	36.2	13.3	7.1	7.1	7.7

Table 5.2: This table reproduces the mean variance for colour reproduction (in ProPhoto RGB chromaticity space) for mobile phone cameras and DSLR cameras of the 24 colour patches on a colour rendition chart. Results are shown for different scenes captured under different colour temperatures. A lower variance means the colour reproduction is more consistent among the cameras. (Variances values are ×1.0E-3.) The top performance is indicated in bold.
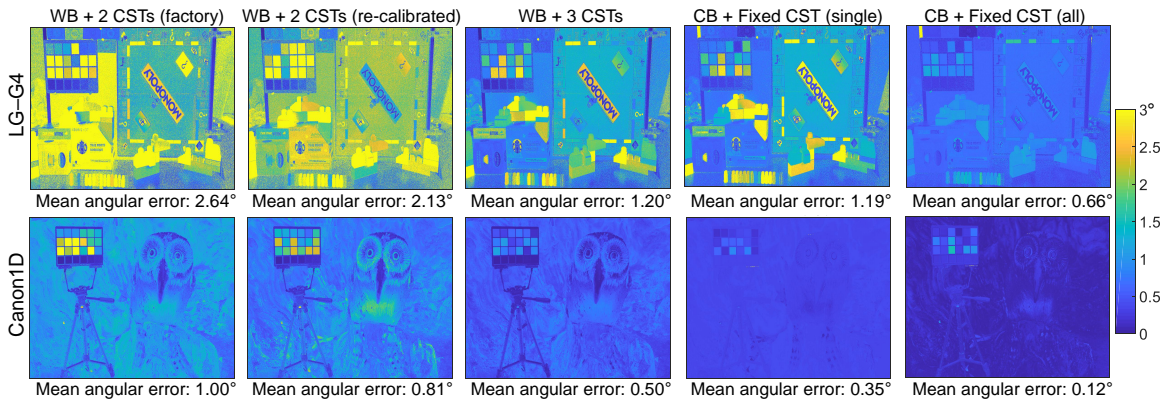
Figure 5.7: Visual comparison for the LG-G4 and Canon-1D. Methods used are the same as used for Table 5.2.

errors only, and (I) for full images. The overall mean angular colour reproduction error is reported. We can see that for all approaches our method 2 based on full colour balance generally performs the best. Our method 1 performs slightly better in a few cases. The best improvements are gained from the mobile phone cameras; however, DSLRs do show a notable improvement as well.

Figure 5.7 shows visual results on whole images for two representative cameras, the LG-G4 and the Canon 1D. These results are accompanied by heat maps to reveal which parts of the images are being most affected by errors.

Multi-camera consistency One of the key benefits to improved colourimetric conversion is that cameras of different makes and models will capture scene colours in a more consistent manner. We demonstrate the gains made by our proposed approach by examining the reproduction of the colour rendition chart patches among multiple cameras. From the dataset images, we select images under four different illumination temperatures. We then map each image using the respective methods and compute the mean variance of the colour chart patches between the various approaches. The variance is computed in chromaticity space to factor out changes in brightness among the different cameras. Table 5.2 shows the result of this experiment. Our two proposed methods offer the most consistent mapping of perceptual colours.

5.5   Discussion

The work by Cheng et al. (2015a) requires a machine-learning method to predict the colour balance matrix. Because most cameras still rely on fast statistical-based white-balance methods, our second proposed method is not yet suitable for use onboard a camera, but can be beneficial to images processed off-line. These results, however, suggest that future generations of camera designs would benefit from exploiting advances offered by machine-learning methods as part of the colour reproduction pipeline.

5.6   Summary

We have presented two methods that offer improvements for colour reproduction accuracy for cameras. Our first method expands the current interpolation procedure to use three fixed illumination settings. This simple modification offers improvements

up to 19% (DSLR cameras) and 33% (mobile phone cameras); moreover it can be easily incorporated into existing pipelines with minimal modifications and overhead. Our second method leverages recent work (Cheng et al., 2015a) that allows full colour balance to be performed in lieu of white balance. This second method provides the best overall colour reproduction, with improvements up to 30% (DSLR cameras) and 59% (mobile phone cameras).

Chapter 6

A Customized Camera Imaging Pipeline for Dermatological Imaging

The work in Chapter 6 aims to leverage our knowledge of camera pipelines to develop a procedure to assist computer vision researchers working with machine vision cameras. This procedure is described for a custom-built hand-held dermatological imaging device. The device uses a combination of visible and non-visible spectral LEDs to allow capture of visible RGB imagery as well as selected non-visible wavelengths. Our customization involves two components. The first component is a colour calibration procedure that ensures the captured images are colourimetrically more accurate than those obtained through the machine vision camera's native API. The need for colour calibration is a critical component that is often overlooked or poorly understood by computer vision engineers. Our second component is a fast method to integrate the

83

Output directly from the machine vision camera's API

Our customized colour processing

Enhanced RGB image (B) using a narrow band image

A B C
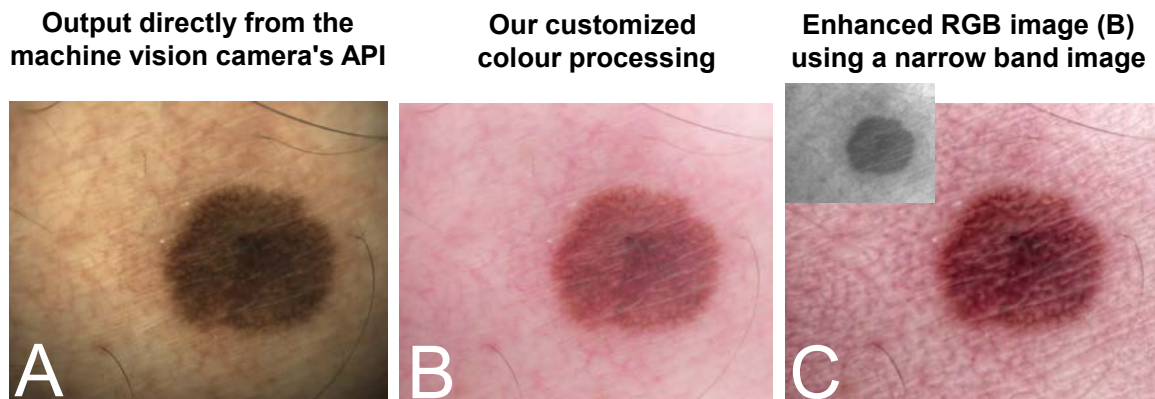
Figure 6.1: (A) Image directly obtained using the machine vision camera's API. (B) Result from our customized camera processing pipeline after colour calibration and photo-finishing. (C) Enhancement of the visible imaging using a selected spectral band to highlight melanin pigmentation. The narrow band image is shown as an inset.

narrow band spectral images (some of which are outside the visible range) into the visible RGB image for enhanced visualization. This component of our pipeline involves evaluating several algorithms capable of multiple image fusion to determine the most suitable one for our application. Quantitative and subject results, including feedback from clinicians, suggest the effectiveness of our customization procedure. While this chapter targets a specific application, the description for the machine vision camera calibration and perceptual mapping is generic and can be applied to other application domains.

6.1 Introduction

This chapter describes a custom imaging device for dermatological inspection constructed with an integrated machine vision camera. Machine vision cameras offer several advantages over consumer-oriented cameras, such as DSLR and mobile phone

cameras. These advantages include: (1) a larger sensor offering improved signal-to-noise performance; (2) the lack of an NIR filter that allows non-visible spectral data to be captured; and (3) supporting software and APIs that allow low-level control over camera settings and the ability to perform external event triggering for image capture. One crucial disadvantage of machine vision cameras, often overlooked by computer vision researchers, is that these cameras are typically not colourimetrically calibrated. Moreover, their supporting APIs often do not provide any mechanism for colour correction beyond simple per-channel gain manipulation. In addition, machine vision cameras lack onboard photo-finishing that allows consumer-oriented cameras to produce perceptually pleasing images. When machine vision camera images are used in applications where the image content needs to be displayed to users (especially non-vision experts), the perceived quality of the machine vision imagery often appears lower than consumer-camera imagery—even though the overall sensor performance of the machine vision camera is better (see Figure 6.1).

In addition, for applications that benefit from images within specific spectral bands—for example NIR spectral bands—there are a range of different options regarding how to use this information to enhance a visible three-channel RGB image. In this chapter, we explore these options within the context of dermatological imaging.

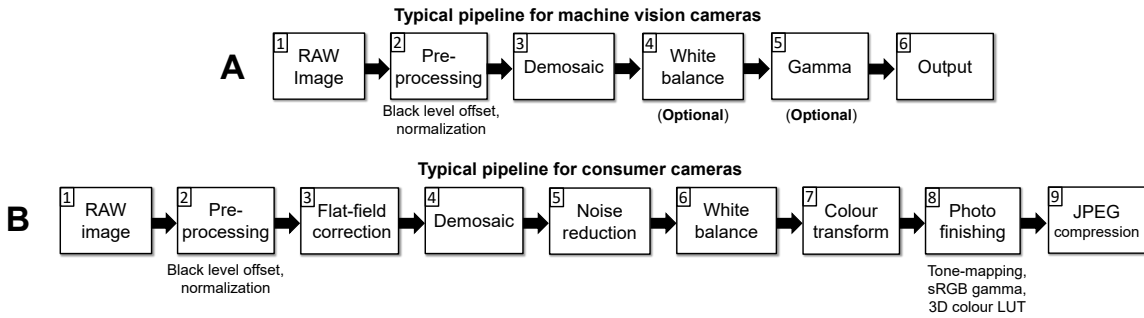## 6.2   Imaging Device and Application

The customization described in this chapter is for a dermatological imaging device that is undergoing clinical evaluation. Figure 6.2-(A) shows a diagram of the device's construction. A machine vision camera using a Sony IMX172 12MP RGB sensor is integrated into a closed housing with an LED light ring. The light ring is composed
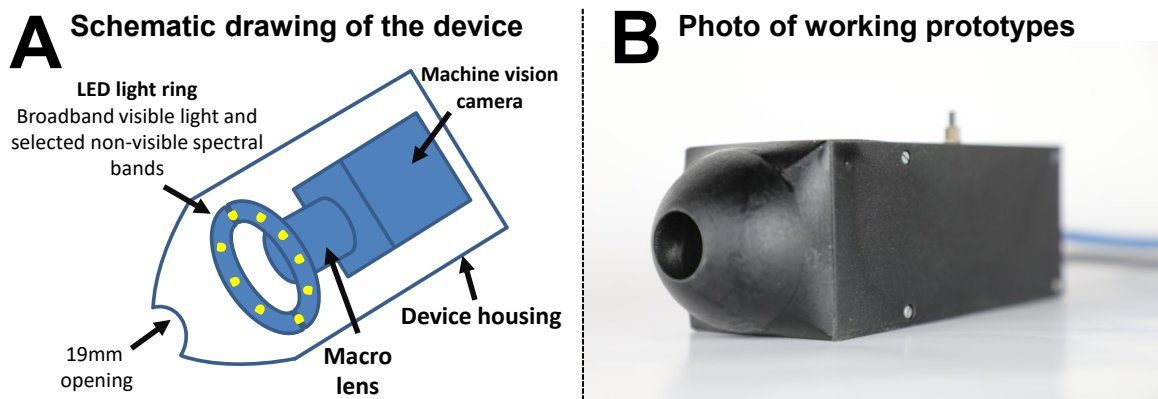
Figure 6.2: Our device: (A) Schematic diagram of our imaging device. A machine vision camera is integrated inside a closed housing with a ring of LEDs. The LEDs emit a range of visible and non-visible spectra targeting different biomarkers for dermatological disorders. (B) Image of the current prototype device.

for eight (8) LEDs that emitted selected visible and NIR wavelengths, each targeting different biomarkers (e.g., eumelanin and hemoglobin). The corresponding LED wavelengths range between 400 nm and 1100 nm. An additional LED that emits broadband visible light is included. The sensor lacks a hot mirror that is typically found on consumer cameras to block NIR light. By triggering image capture with the LED emission, a visible RGB image in addition to the multiple spectral bands can be captured in a single image capture session. Figure 6.2-(B) shows an image of the actual device.

The visible and non-visible images obtained from this device are used with a proprietary deep-learning module that provides dermatologists with objective data to decide if a biopsy is needed for further investigation of a skin lesion. In addition, the doctors can use the device for magnification of the target and their own visual inspection. When used manually, our device can serve as a potential replacement, or auxiliary device, for existing hand-held dermatoscopes that use a magnifying lens

Typical pipeline for machine vision cameras

| 1 RAW Image | → | 2 Pre-processing | → | 3 Demosaic | → | 4 White balance | → | 5 Gamma | → | 6 Output |

Black level offset, normalization (under step 2)
(Optional) (under step 4)
(Optional) (under step 5)

Typical pipeline for consumer cameras

| 1 RAW image | → | 2 Pre-processing | → | 3 Flat-field correction | → | 4 Demosaic | → | 5 Noise reduction | → | 6 White balance | → | 7 Colour transform | → | 8 Photo finishing | → | 9 JPEG compression |

Black level offset, normalization (under step 2)
Tone-mapping, sRGB gamma, 3D colour LUT (under step 8)

Figure 6.3: (A) A typical camera imaging pipeline overviews the common steps applied onboard a machine vision camera. (B) A typical camera imaging pipeline overviews the common steps applied onboard a consumer camera.

with an illumination source. We note that this chapter does not discuss the associated deep-learning system. Instead our focus is on the calibration of the imaging rig, with emphasis on processing the machine vision data for perceptual output.

6.3 Machine Vision Versus Consumer Camera Pipelines

Here we describe the processing steps found on machine vision cameras and consumer cameras. This section helps to reveal why machine vision cameras' images appear visually different from consumer cameras. The section also concludes with a short discussion on common issues that arise when working with machine vision cameras.

Machine vision camera pipeline Figure 6.3-(A) shows a diagram of a machine vision camera pipeline. Although machine vision cameras are generally more expensive than consumer cameras, the onboard processing is typically much simpler. The following steps are typical of most machine vision cameras.

Step 1: Raw-RGB capture. The raw-RGB image is obtained from the camera

sensor in a mosaiced Bayer pattern format. The data is typically 10–16 bits per channel.

Step 2: Pre-processing. The raw-RGB image is linearized to transform its values to range between 0 and 1. This includes a black-level offset correction based on the camera's current sensor readings.

Step 3: Demosaicing. The image is demosaiced into three full-sized channels.

Step 4: White balance. An optional white balance can be applied. This is typically applied as an independent gain on the red and blue channels only. Most machine vision cameras will provide an auto white balance function or allow this to be manually set via software API calls. Note that most APIs do not call this white balance.

Step 5: Gamma. Most machine vision cameras will allow an optional gamma function to be applied to the RGB image.

Step 6: Output. The final output image can be optionally quantized to 8-bit, but is typically not compressed.

It is important to note that the machine vision camera image is still in the sensor-specific RGB space. As a result, the image is not suitable for display on monitors which assume a sRGB display-referred colour space. Figure 6.1-(A) shows an example of an output image. Although the image may appear correct, the colours are being incorrectly interpreted as sRGB.

Consumer camera pipeline As we have already described the details of the consumer camera pipeline in Chapter 3, we skip it in this chapter. However, we will provide a summary in order to provide a comparison. Figure 6.3-(B) shows a diagram of a consumer camera pipeline. Consumer cameras aim for perceptually pleasing

images. Steps 1 and 2 are the same as a machine vision camera and are not discussed here. Note that some of the steps may be applied in different orders.

Step 3: A flat-field correction. A spatially varying gain is applied to correct for uneven light fall on the sensor due to the camera's form factor and lens characteristics.

Step 4: Demosaicing. This is applied similarly to the machine vision camera. Often edge sharpen may be included to enhance the appearance of the image.

Step 5: Denoising. Many consumer cameras, especially smartphone cameras, incorporate some type of image denoising.

Step 6: White balance. Unlike machine vision cameras, this is not optional. All consumer cameras apply a white-balance step. White balance compensates for the scene illumination and prepares the image for a subsequent colour space transform to map the sensor-specific raw-RGB to a device-independent perceptual colour space.

Step 7: Colour space transform. Consumer cameras convert the raw-RGB colour space to a perceptual colour space based on CIE XYZ, such as a linear sRGB. This transform is reliant on the quality of the previous white-balance step and does have limitations (for more details see Chapter 5); however, in general, the colour transform step serves as a reasonable colourimetric conversion of the raw-RGB image.

Step 8: Photo-finishing. Consumer cameras apply proprietary colour manipulation to improve the appearance of the image. While this is shown as a single step in Figure 6.3-(B), photo-finishing often involves multiple substeps depending on the camera. These steps include per channel tone-curves, application of 3D LUTs, and selective colour manipulation. Spatially varying photo-finishing can also be applied. The photo-finishing step generally imparts a unique look and feel to the final output image that is associated with a particular make and model of a camera (e.g., Nikon,

Canon, iPhone, Samsung).

Step 9: Compression and output. Consumer cameras apply JPEG compression to the output image. The image is encoded in a display-referred colour space, typically sRGB. These images are suitable for display.

Remarks Many computer vision researchers are not aware of the differences between a machine vision and consumer camera pipeline. Often the API documentation provided with machine vision cameras does not help to clarify these issues. The machine vision camera based on the Sony IMX172 sensor provided additional API calls that allowed manipulation of the captured image in the Hue, Saturation, Value (HSV) space. While HSV can be used to re-parameterize an arbitrary RGB colour space, it is most often associated with a gamma-encoded sRGB colour space. The Sony IMX172 documentation failed to mention that the HSV is in the camera's raw-RGB colour space. Only by calling the manufacturer were we able to assess this. In fact, the camera's documentation never specified if a colourimetric conversion was performed onboard the camera.

Not understanding the colour space images are saved in can have serious consequences. For example, when displaying sensor-specific RGB images, the observer is not seeing an image that is correctly calibrated for a display. Another serious consequence is when the imagery is used to train machine learning modules. If a deep-learning module is trained using a sensor-specific RGB input, it may not generalize well to other input colour spaces—as a result, an application can be restricted to work with only a particular make and model of a camera. The following section discusses how this can be avoided by customizing the camera's pipeline of a device.
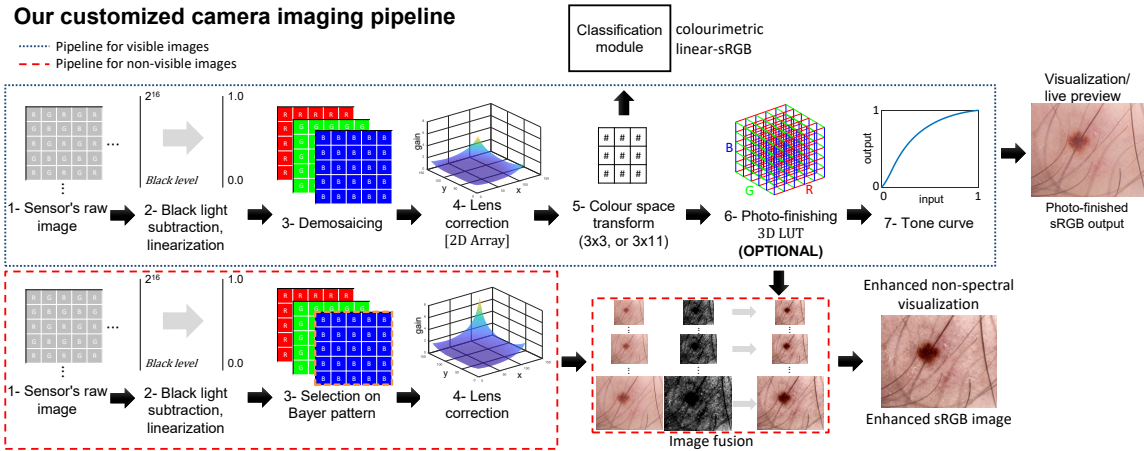
Figure 6.4: This figure shows the customized pipeline design which allows the machine vision camera to behave more similarly to a consumer camera pipeline. A noticeable difference is that we do not need to perform a white-balance step since we will always be imaging under the same lighting conditions. This means the white balance can be absorbed into the colour space transform. We include additional tone curve processing and optional colour manipulation (3D LUT) to mimic the appearance of consumer cameras. Our pipeline also includes a select spectral band image pipeline that processes the narrow band spectral image and then integrates it with the RGB image.

6.4 Custom Camera Pipeline and Results

This section describes our customized camera pipeline for our dermatological imaging device. Further, given the availability of the narrow band spectral data (including non-visible bands), we provide a mechanism to use this information to enhance our RGB image. Figure 6.4 shows an overview of our customized pipeline.

Flat-field correction Because our LED lights are at different positions, we need to compute a flat-field correction for each LED. We can do this by imaging one of the achromatic patches on a colour rendition chart with each LED turned on. This patch has a uniform reflection and therefore any deviation in the captured image is

assumed to be due to a combination of light position and lens. To correct the non-uniform response, we construct a map that inversely scales each pixel's intensity to the intensity level in the middle of the image that the lens focuses on best.

Figure 6.5 shows the vignetting maps computed for each nine LEDs. Due to slight misalignment of the camera's optical axis with the 3D printed housing of our prototype device, the flat-field distortion maps are all dominated by strong fall near the lower region of the imaging field. Such issues should be resolved with proper industrial design of the production-level device.

Colour calibration Our sensor needs to be colourimetrically calibrated as discussed in the previous section. We tested both a 3×3 and a 3×11 matrix CST to map our raw-RGB values to the CIE XYZ colour space. The 3×11 matrix is based on the polynomial equations proposed by Hong et al. (2001). Since we have a fixed light source in our imaging device, we do not need to compute a white-balance matrix. Instead we can directly compute a transform from the raw-RGB to our target colour space. In our case, we map to linear-sRGB. To compute our CST, we place our device over each of the colour patches in a Macbeth chart and take a reading with the visible band LED light turned on. Flat-field correction is applied to the captured images before extracting the raw-RGB colour values of each patch.

We use iteratively re-weighted least squares (Huber & Ronchetti, 2009) to compute the mapping functions based on the colour correspondence. To examine the quality of our mapping, we can visualize the angular error between the target patch colours and our mapped colours.

We also examine our estimated CSTs on a different set of colour patches consisting of 128 skin colours obtained from the *Munsell Book of Color* reference. The *Munsell*
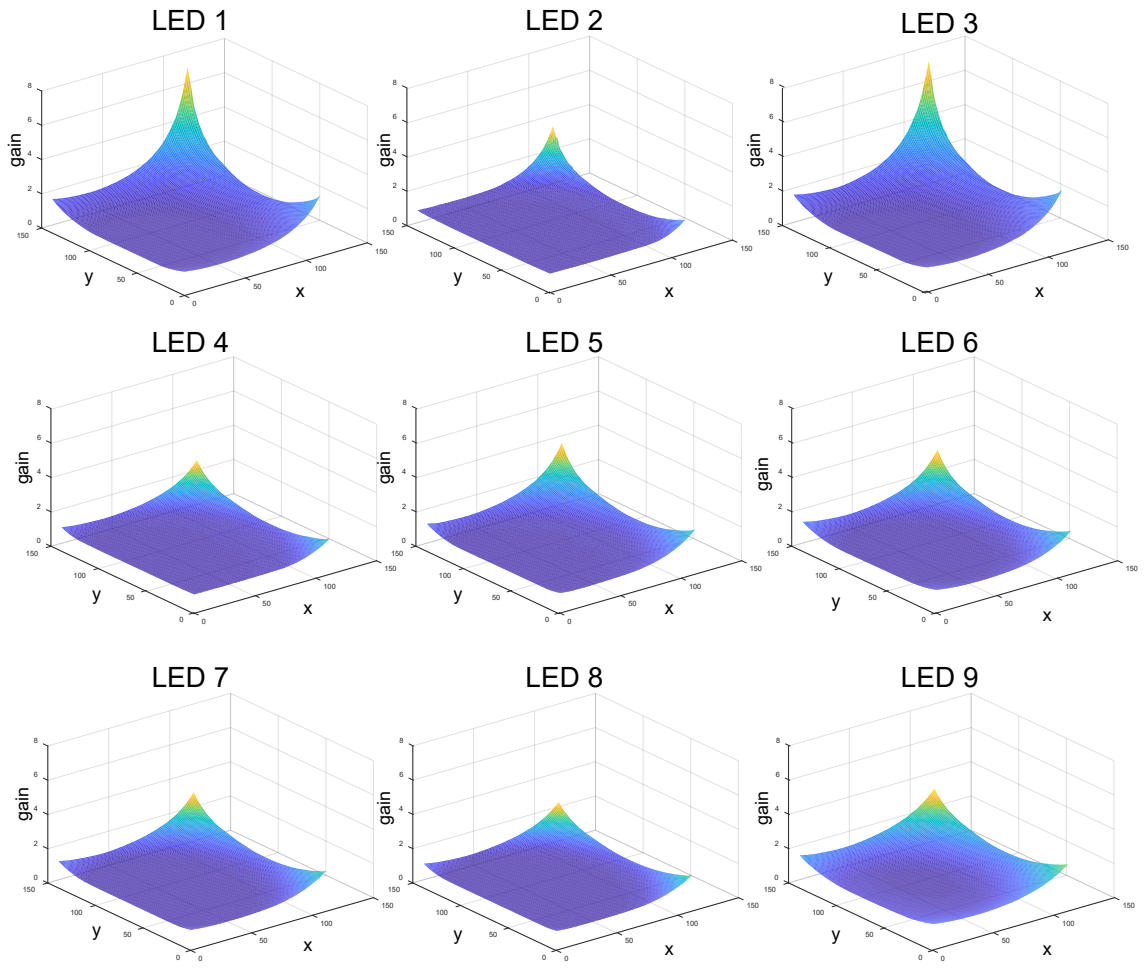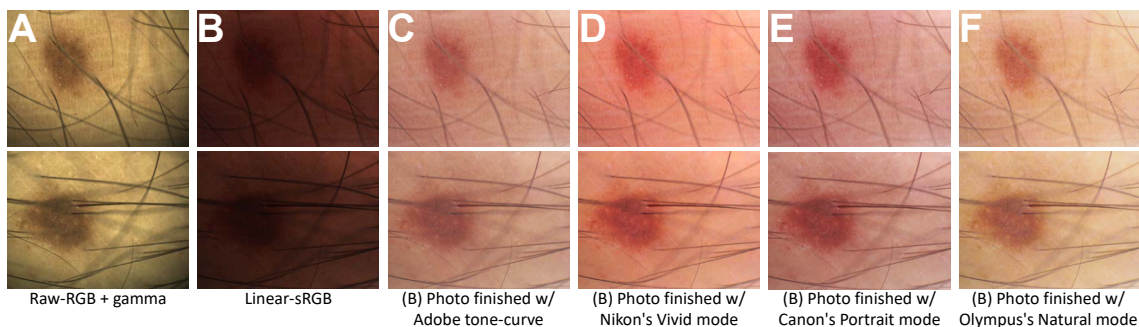
Figure 6.5: Flat-field correction maps corresponding to the nine different LED lights.
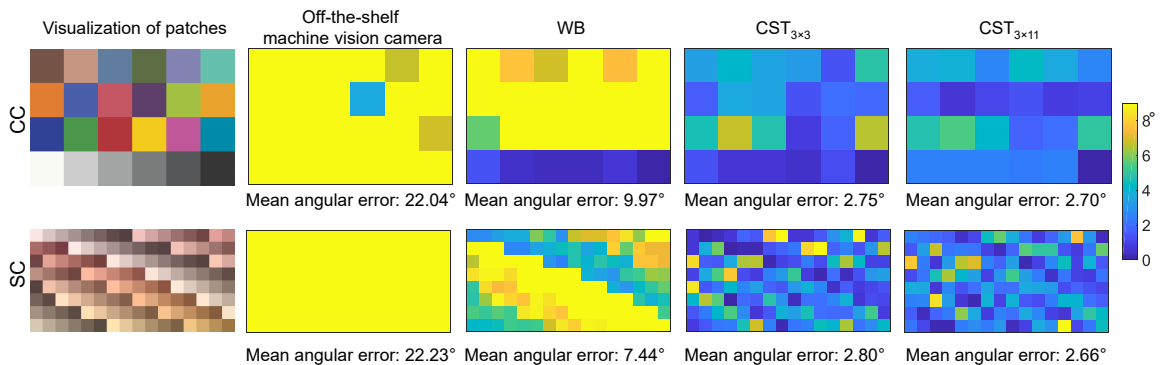
Figure 6.6: Angular errors for colour chart colours (denoted as CC), and on skin colours (denoted as SC) from the *Munsell Book of Color* reference.

Book of Color provides patches that are indicative of skin tones. We also compare the results of applying white balance to the raw-RGB image. The white-balance gains are computed by measuring an achromatic patch with the device and then adjusting the camera's red and blue gain such that $R = G = B$ for all pixel intensities in that patch. The angular RGB errors for each of these methods are shown in Figure 6.6. We can see that colourimetric calibration drastically improves the accuracy of mapping to a perceptual space.

Photo-finishing After colourimetric conversion, we can send the linear-sRGB images off to our AI module for image analysis. However, for visualization, we still need to apply an additional non-linear mapping. The sRGB standard specifies a 2.2^{-1} gamma encoding to brighten the image. However, all cameras apply a proprietary tone-curve to process the image instead of the 2.2^{-1} gamma encoding. These tone curves can often be extracted from metadata saved in the Adobe DNG file format (Karaimer & Brown, 2016). Additionally, many cameras, especially DSLRs, include a more complicated colour manipulation based on a 3D LUT. The LUTs are associated with different picture styles settings on the camera (see Kim et al., 2012).

A	B	C	D	E	F
Raw-RGB + gamma	Linear-sRGB	(B) Photo finished w/ Adobe tone-curve	(B) Photo finished w/ Nikon's Vivid mode	(B) Photo finished w/ Canon's Portrait mode	(B) Photo finished w/ Olympus's Natural mode

Figure 6.7: This figure shows the output of the machine vision camera with a gamma mapping applied on two different examples (A). (B) shows the linear-sRGB mapping computed from the colourimetric calibration. (C, D, E, F) show various versions of photo-finishing applied to image (B) to produce different appearances. These photo-finishing versions can be emulated in our custom pipeline.

For our customization pipeline we are able to do both. Figure 6.7 shows an example. Figure 6.7-(A) shows the standard output of the machine vision camera. Figure 6.7-(B) shows the linear-sRGB result. Figure 6.7-(C) shows a tone-curve applied based on Adobe's standard photo-finishing tone-curve. This tone-curve is not associated with any particular camera, but is used by Adobe's Lightroom software when no tone-curve is specified. This produces a perceptual image that looks much more similar to a consumer camera.

Figure 6.7-(D,E,F) uses metadata extracted from DNG files to emulate the following camera modes: D-Nikon Vivid mode, E-Canon Portrait mode, and F-Olympus Natural mode. For our working system, we use the Adobe tone-curve (or a custom curve), however; when desired, we can produce outputs that mimic existing camera models. Our colourimetric conversion and photo-finishing can be performed in real time and are part of the live-view display feed from the camera based on the visible RGB image.
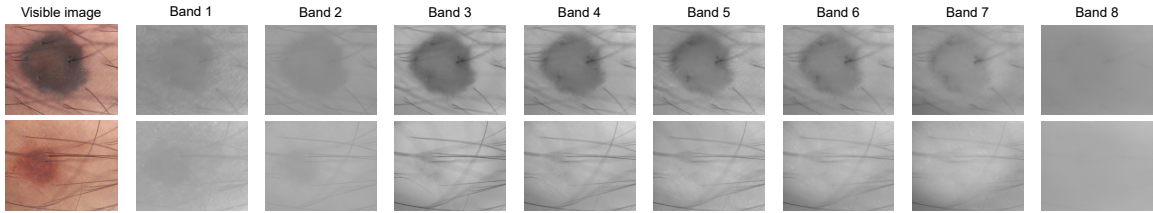
Figure 6.8: The visible and corresponding narrow band images for two different melanocytic nevus lesions. The first lesion (top row) has a structure that extends deeper into the skin; this is apparent in the individual spectral bands. The second lesion (bottom row) is primarily on the outer layer of the skin and contributes very little to the narrow spectral bands.

6.5 Spectral Data Fusion

As part of our customized pipeline, we allow the RGB image to be enhanced using selected spectral bands. Which band to use for enhancement is currently manually specified. Note that this process cannot yet be performed in real time, but instead takes around 1.5–3 seconds per image. This part of our setup is currently not integrated onboard the camera processor, but can be incorporated in future designs.

As shown in Figure 6.4, our device must first capture and process the single-channel spectral bands using a simplified pipeline. Since these images are not obtained from the visible spectrum, there is no need to perform colourimetric calibration. Moreover, these bands are not demosaiced, but instead the channel most sensitive to the imaged spectra is used (e.g., the raw-B channel for the light's corresponding shorter wavelengths, the raw-R channel for longer wavelengths, such as NIR). Once the image is captured, flat-field correction is applied and the images are upsampled to the RGB image's size. Using this image we can perform enhancement. Figure 6.8 shows the visible and the corresponding spectral images for two sample skin lesions.

96

We have examined the following three (3) spectral fusion methods. Each is based on a different strategy, including bilateral filtering, Laplacian pyramid filtering, and wavelet-based image fusion.

(1) Bilateral filter (Fredembach et al., 2009): The first method that we use is based on work by Fredembach et al. (2009). While the authors proposed a method that fused NIR to a visible image to remove skin imperfections, we used the spectral data instead to boost the nonvisible details. This method is based on the fast bilateral filter method introduced by Paris and Durand (2006), which is applied on both the visible luminance image and single-channel spectral image. The method works by decomposing the photo-finished sRGB image into the YCrCb colour space. A detail layer is obtained from the spectral image by subtracting the input image from its filtered version. This layer is added to the filtered Y channel of the sRGB filtered image. The modified Y channel is recombined with the CrCb channels to produce the final output with enhanced details from the spectral image.

(2) Fast local Laplacian filters (Aubry, Paris, Hasinoff, Kautz, & Durand, 2014): Our second method is based on the fast local Laplacian filters. This method was proposed by Aubry et al. (2014), who provide a performance modification over local Laplacian filters first introduced by Paris et al. (2011). While local Laplacian filters were not used for image fusion, the method is fairly straightforward to allow sharing of information between two images. Specifically, we use the algorithm outlined in Aubry et al. (2014) to compute nine (9) levels of the Gaussian pyramid for the spectral image, as well as the corresponding Gaussian pyramid and the Laplacian pyramid of the visible image's Y channel. Afterwards, we remapped the narrow band image using the S-shaped curve as done in Aubry et al. (2014) to boost the local
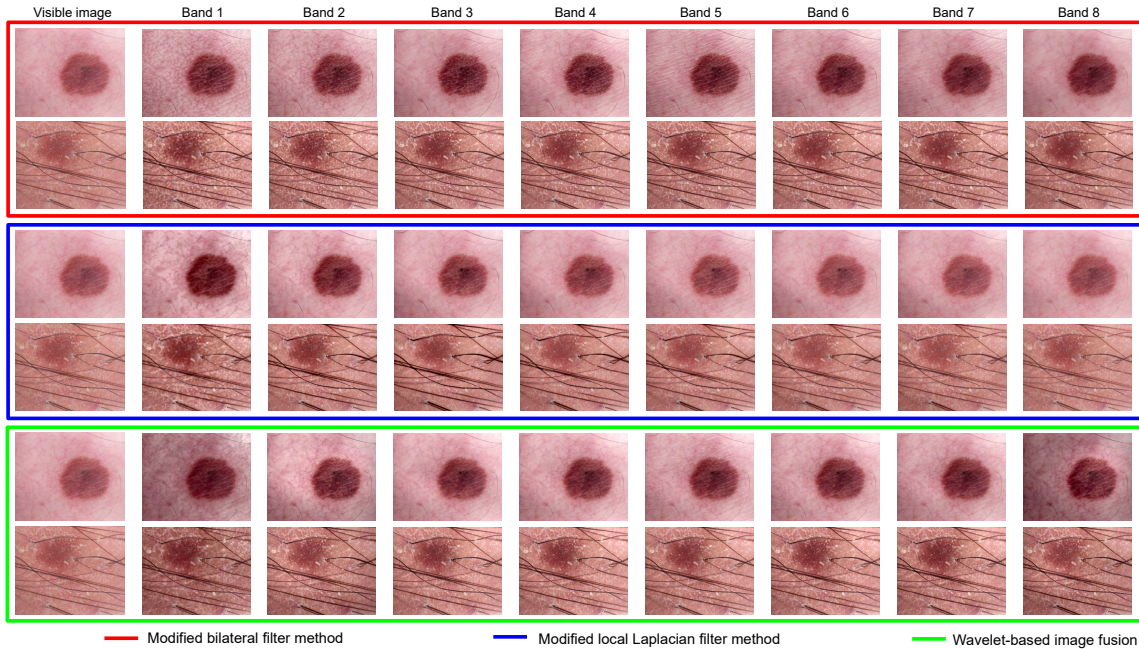
Figure 6.9: This figure shows the results of several different narrow band spectral images being used to modify the photo-finished sRGB image. The methods applied are the bilateral filter based on a method by Fredembach, Barbuscia, and Süsstrunk (2009), a local Laplacian filtering approach based on the method proposed by Aubry, Paris, Hasinoff, Kautz, and Durand (2014), and a wavelet-based method. While there are subtle differences, the bilateral filter and local Laplacian give similar performance. The wavelet-based method tends to introduce low-frequency intensity shifts in the overall image. Note that each image to the right of the visible is the result of fusing the image from the narrow band with the visible.

gradients of the spectral image. These boosted gradients are blended with the local gradient information from the Y channel of the RGB image. Before we collapse the combined Laplacian pyramid, the last level of the Gaussian pyramid is copied from the visible Y channel. This effectively transfers the spectral image's details to the visible image Y channel. The modified Y channel is recombined with the CrCb channels to produce the final output.

(3) Wavelet-based image fusion: Our last method is a wavelet-based image fusion approach that was also discussed in Fredembach et al. (2009). This method also starts by converting the photo-finished sRGB image to the YCbCr colour space. A wavelet decomposition is applied using a Symlet wavelet generator on the Y channel of the visible image and the selected spectral image. After the coefficients are calculated, the wavelet coefficients in the visible image's Y channel are replaced by the corresponding spectral image's wavelet coefficient if its coefficient's magnitude is larger. Reconstructing the image with the fused wavelet coefficients performs the image fusion on the Y channel. The new Y image is recombined with the CrCb channels of the visible image to produce the final output.

Figure 6.9 shows the results of the three approaches. While these results are subjective in nature, our observations are that the local Laplacian filter and bilateral filter methods give similar performance. The wavelet-based methods tend to introduce low-frequency intensity shifts in the overall image. The bilateral and local Laplacian filter methods both run in about 2 seconds on a standard workstation PC. The narrow band spectral fusion is still an area of ongoing investigation and will require several rounds of user study to determine the most preferred methods by practitioners. However, regardless of which method will be eventually be adopted,

the overall processing pipeline for this image fusion will be based on our current customized pipeline design.

Clinical feedback In order to find out if there was a preference between the methods and the different spectral bands, we conducted a simple study with practicing dermatologists who have been evaluating the device.

The clinicians examined the visible image and the corresponding spectral images as well as the three fusion methods. We showed ten cases and for each case we asked three questions:

(1) Among the three methods used to perform spectral image fusion, do you have a preferred method?

(2) Is there a particular spectral image that you feel provides the most information?

(3) Do you feel this type of fusion is useful for you in a clinical setting (i.e., would it help you make a more informed decision)?

The feedback was as follows. For question 1, dermatologist 1 chose the wavelet-based method eight times out of ten samples. Similarly, dermatologist 2's preference was the wavelet-based method, which was chosen seven times out of ten examples. Dermatologist 3 chose the method based on bilateral filtering of all their choices.

Regarding which wavelengths to fuse, band 8 (the highest NIR band) was the most preferred. Dermatologist 1 preferred band 1 (the lowest UV band) seven times out of ten cases. Dermatologist 2 equally chose band 2 and band 8 three times out of ten samples, and dermatologist 3's preference was band 8 nine times out of ten examples. This suggests that the NIR data, especially for longer wavelengths, provided useful information to the clinicians.

For the last question, comments from our participants include "Yes, I think that the fusion is very helpful" and "I could see the pattern of each lesion much better (reticular, dots, borders)." One dermatologist who was neutral on the idea of fusion commented, "Only in certain cases." While the preferences of the dermatologists were not always consistent with each other, their individual preferences were consistent.

6.6 Discussion

Specific to our dermatological application, we are still exploring ways to integrate the narrow spectral bands for image enhancement. Currently we use only a single spectral band at a time. Combining multiple bands may allow doctors to assess skin lesions faster. Addressing these issues is part of ongoing work. Based on our pilot user study, however, given the diversity in opinions regarding which bands are most useful, we advocate allowing manual selection over an automated strategy.

6.7 Summary

This section has detailed a custom in-camera processing pipeline for a dermatological imaging device that captures both visible and selected narrow band spectral information. As part of this work, we have described the basic camera processing pipelines found on machine vision cameras and discussed limitations that are often overlooked in applications of this nature—namely, the lack of colourimetric calibration and photo-finishing on machine vision cameras. We have provided a detailed description of how to calibrate our camera to improve the colourimetric accuracy as well as mimic the photo-finishing routines found on consumer cameras. While our

work targets a dermatological device, we believe information described in this chapter will be useful for the design of similar devices.

Chapter 7

Conclusion and Future Directions

This chapter concludes the dissertation with a collective summary of the contributions made and potential future directions for follow-on research.

7.1   Summary

The goal of this dissertation has been to examine the colour rendering procedure in the context of the in-camera imaging pipeline. In Chapter 2 we presented the required background on colour and related work in the literature.

Chapter 3 presented our camera simulator for researchers to have access to the onboard imaging system. The software platform allows modification of the parameters for individual components as well as the ability to access and manipulate the intermediate images as they pass through different stages. The simulator made it possible to complete the other chapters of the dissertation.

In order to make the colours consistent across multiple cameras, Chapter 4 investigated a consistent image state in the camera imaging pipeline. Our approach leverages the camera imaging pipeline stages, and offers cameras the chance to produce an extra output at the colourimetric conversion stage.

Chapter 5 presented two methods to improve colour reproduction accuracy in the camera imaging pipeline. Our first method uses white balance and CST interpolation with three pre-calibrated illuminations. The second method uses a full colour balance and a fixed CST. Although our second method requires a machine-learning method to predict the colour balance matrix, it achieves the best results. On the other hand, our first method can be easily incorporated into existing pipelines with minimal modifications and overhead.

Finally, Chapter 6 presented how to customize a camera imaging pipeline for a constrained use case scenario. To avoid errors due to proprietary functionality of the cameras, the goal was to customize the camera imaging pipeline for a custom-built device for a dermatological imaging application. Quantitative and subject results, including feedback from clinicians, suggest the effectiveness of our customization procedure.

In summary, this dissertation examined the colour rendering procedure in the camera imaging pipeline. In order to perform experiments within the proper context of the pipeline, a camera simulation tool is developed. A new image output format that makes imaged objects consistent across cameras is proposed. Improvements on the individual camera colourimetric accuracy and consistency between multiple cameras are presented. Finally, a customization procedure for an imaging pipeline in the context of dermatological imaging is described.

7.2 Future Directions

7.2.1 Cameras for Purposes Beyond Photography

In Chapter 4, we show compelling evidence that accessing the colourimetric image is useful; however, access to all image states would be welcomed. As we discussed before, smartphone cameras currently do not include the nonlinear colourimetric metadata in their raw-RGB files. This lack of metadata made it impossible for us to evaluate the true colourimetric conversion performed onboard smartphone camera. In order to use the camera for purposes beyond photography, analysis of other image states for both DSLR and smartphone cameras is part of our ongoing and future efforts.

7.2.2 A Full Learning-Based Camera Simulator

Although attempts at learning-based camera simulation software are proposed (Jiang, Tian, Farrell, & Wandell, 2017; Nam & Kim, 2017; Jaroensri, Biscarrat, Aittala, & Durand, 2019), currently none of them are able to simulate the effect of ISO change and optics. Recent mobile phones are capable of changing aperture size and optical zoom. Therefore, considering incorporating these features may help to serve the goal of open research platforms. Moreover, scene-dependent photo-finishing routines could also be included since this kind of processing is one of the core functionalities for mainstream photography applications.

Appendix A

Improving Colour Reproduction Accuracy on Cameras (Supplemental Material)

Overview

Here we provide supplemental details and results to accompany Section 5.4. The supplemental material is organized as follows. Section A.1 presents how we use the same interpolation strategy as described in Sec. 5.2.2 that used white balance corrected images, but instead use full colour balance and CST estimated using Eq. 5.9. Section A.2 provides details for the algorithm used to estimate the correlated colour temperature (CCT) of an observed neutral object in the camera's RGB colour space.

This corresponds to Section 5.2.2 of the dissertation. Section A.3 shows the corresponding 2D plots of the chromaticity values of the colour rendition colour chart patches from multiple cameras. This corresponds to the experiments in Section 5.4.2 (multi-camera consistency) of the dissertation. Section A.4 provides additional results in terms of colour reproduction angular error. This corresponds to the experiments in Section 5.4.2 (individual camera accuracy).

A.1 Method 2 (Extension): Full Colour Balance with Interpolation

While the full colour balance allows the computation of a fixed CST that should be applicable to all illuminations, from Eq. 5.9, it is clear the errors for the estimated CST mapping will be minimized for a particular illumination when only a single i is used as described in Section 5.3. As a result, we can use the same interpolation strategy as described in Section 5.2.2 that used white-balance corrected images, but instead use full colour balance and CST estimated using Eq. 5.9. Results for this extension approach are not included in the dissertation, but can be found in this appendix. We present results of this extension (CB + 2 CSTs) incorporated into the tables in the dissertation.

Method	Apple iPhone 7		Google Pixel		LG-G4		Canon1D		NikonD40		Sony α57		Olympus E-PL6	
	CC	I	CC	I	CC	I	CC	I	CC	I	CC	I	CC	I
CB + Fixed CST (all)	**0.80**	**0.91**	**0.88**	1.05	**0.81**	**0.86**	**0.84**	**0.63**	**0.93**	0.77	1.00	0.77	0.85	0.69
CB + Fixed CST (single)	0.95	1.04	1.41	1.45	1.17	1.17	0.95	0.76	0.98	0.80	0.99	0.76	0.82	0.67
CB + 2 CSTs	0.83	0.95	1.16	1.21	0.88	0.91	0.91	0.66	0.93	0.72	**0.93**	**0.73**	**0.82**	0.65
WB + 3 CSTs	1.36	1.08	1.27	**1.04**	1.55	1.24	1.04	0.70	1.07	**0.65**	1.04	0.74	0.86	**0.61**
WB + 2 CSTs (re-calibrated)	1.76	1.42	1.92	1.46	1.98	1.54	1.16	0.82	1.33	0.82	1.13	0.81	1.05	0.71
WB + 2 CSTs (factory)	3.07	2.46	2.28	1.75	2.74	2.04	1.58	1.18	2.02	1.26	1.14	0.79	1.12	0.75

Table A.1: The table shows the comparisons of error between full colour balance with fixed CST, diagonal matrix correction with 3 CST, native cameras (re-calibrated for the datasets we use), and native cameras (factory calibration). Errors are computed on colour chart colours only (denoted as CC), and on full images (denoted as I). The top performance is indicated in bold.

Method	Mobile Phones 2900°K	Mobile Phones 4500°K	Mobile Phones 5500°K	Mobile Phones 6000°K	DSLRs 3000°K	DSLRs 3500°K	DSLRs 4300°K	DSLRs 5200°K
CB + Fixed CST (all)	**4.6**	**2.1**	**1.1**	**1.0**	**1.6**	**1.8**	**1.2**	**1.1**
CB + Fixed CST (single)	10.0	7.1	4.9	6.2	2.7	2.9	2.1	1.7
CB + 2 CSTs	7.3	4.7	3.5	4.8	2.2	3.4	1.9	1.7
WB + 3 CSTs	37.9	6.9	6.6	8.1	4.4	2.8	2.3	2.1
WB + 2 CSTs (re-calibrated)	44.9	11.3	9.5	16.7	6.7	3.2	3.4	5.9
WB + 2 CSTs (factory)	34.2	38.2	32.6	36.2	13.3	7.1	7.1	7.7

Table A.2: This table reproduces the mean variance for colour reproduction (in ProPhoto RGB chromaticity space) for mobile phone cameras and DSLR cameras of the 24 colour patches on a colour rendition chart. Results are shown for different scenes captured under different colour temperatures. A lower variance means the colour reproduction is more consistent among the cameras. (Variances values are ×1.0E-3). The top performance is indicated in bold.

A.2 Estimating the Illumination's CCT

The methods used to compute the colour space transform (CST) based on interpolation in the dissertation rely on the ability to estimate the correlated colour temperature (CCT) of the scene's illumination. The algorithm for determining the CCT is presented here. More specifically, we describe the algorithm to map the estimated illumination of the scene (i.e., the camera's observation of an achromatic or neutral object) to its CIE xy chromaticity coordinates. From the CIE xy chromaticity coordinates, the correlated colour temperature can be determined by projecting the CIE xy coordinates to its location on the Planckian locus in the CIE u-v chromaticity space that is the basis for the correlated colour temperature. The details of this CCT projection are outside the scope of this work and readers are referred to Wyszecki and Stiles (1982) for more details.

Instead, we focus here on estimating the CIE xy coordinates. We feel this algorithm is worth noting, because as stated in the dissertation, the goal of the combined white-balance correction and CST is to establish a mapping from the camera's colour space to the CIE XYZ perceptual colour space. However, the interpolation of the CST itself relies on mapping the neutral colour to CIE XYZ, before the CST is established. The algorithm described here provides this mapping of the scene's illumination in the sensor's colour space to CIE XYZ.

The algorithm presented is based on the algorithm used in DNG SDK (Adobe, 2012). The algorithm uses the two pre-calibrated CSTs and their corresponding CCTs. For the camera native method, as well as our proposed interpolation-based method described in Section 5.2.2 of the dissertation, we use the pre-calibrated CSTs,

T_{l_1} and T_{l_2} for CCTs values: 2500°K and 6500°K respectively.

The algorithm works by initially assuming that the CCT of the input is 5000K. It then estimates a new CST, T_l based on this CCT using Equations 5.7 and 5.8 in the dissertation. The new CST is then used to map the camera's neutral point to a new CIE xy value (and a new CCT). This process is repeated until the neutral colour's mapping to CIE xy converges to a specific CIE xy value. The final CCT of the scene's illumination is computed using this converged CIE xy value. The algorithm is presented in Algorithm 1.

A.3   Plots of Multi-Camera Variance

In this section, we present the chromaticity plots from the multi-camera consistency experiments described in Section 5.4.2 of the dissertation. Recall that our multi-camera consistency experiments are performed per camera type (DSLR or mobile phone camera) and for four different illuminations.

The following plots correspond to each result reported in Table 5.2 of the dissertation. The plots show the mapping of a colour rendition chart colour patches projected into ProPhoto RGB chromaticity space. This experiment is performed to reveal which colour rendition method results in the least amount of variance in the overall colour values among multiple cameras.

Figures A.1, A.2, A.3, A.4, A.5, and A.6 show the variance for the mobile phone cameras under illumination 2900°K.

Figures A.7, A.8, A.9, A.10, A.11, and A.12 show the variance for the mobile phone cameras under illumination 4500°K.

Figures A.13, A.14, A.15, A.16, A.17, and A.18 show the variance for the mobile

Algorithm 1 Mapping the estimated illumination to its CIE xy value and CCT

Input: 1. Observed neutral object in the camera's RGB: *neutral*

              2. Pre-calibrated CSTs: T_{l_1} and T_{l_2} and their corresponding CCT values 2500K and 6500K

              3. Lookup function $\mathrm{CCT}(x, y)$ that returns the CCT of an CIE x, y chromaticity.

1: $last.x \leftarrow 0.34$ // Starting values correspond to CCT 5000K
2: $last.y \leftarrow 0.35$ // in CIE xy
3: $maxNumberOfPasses \leftarrow 30$
4: **for** $pass \in \{0, \ldots, maxNumberOfPasses-1\}$ **do**
5:      $\mathrm{CCT}_{\mathrm{last}} \leftarrow \mathrm{CCT}(last.x,\ last.y)$
6:      $g = \mathrm{CCT}_{\mathrm{last}}^{-1} - \mathrm{CCT}_{\mathbf{6500}}^{-1} / \mathrm{CCT}_{\mathbf{2500}}^{-1} - \mathrm{CCT}_{\mathbf{6500}}^{-1}$
7:      $T_l = g\ T_{l_1} + (1 - g)\ T_{l_2}$
8:      Apply T_l to *neutral* to map it to the updated CIE XYZ
9:      $next \leftarrow$ the updated CIE xy
10:     **if** $|next.x - last.x| + |next.y - last.y| < 0.0000001$ **then**
11:         $last \leftarrow next$
12:         BREAK;
13:     **end if**
14:      $last \leftarrow next$
15: **end for**
16: **if** $pass = maxNumberOfPasses - 1$ **then**
17:       $last.x \leftarrow (last.x + next.x) * 0.5$
18:       $last.y \leftarrow (last.y + next.y) * 0.5$
19: **end if**
20: $last \leftarrow next$
Output: $\mathrm{CCT}(last.x,\ last.y)$.
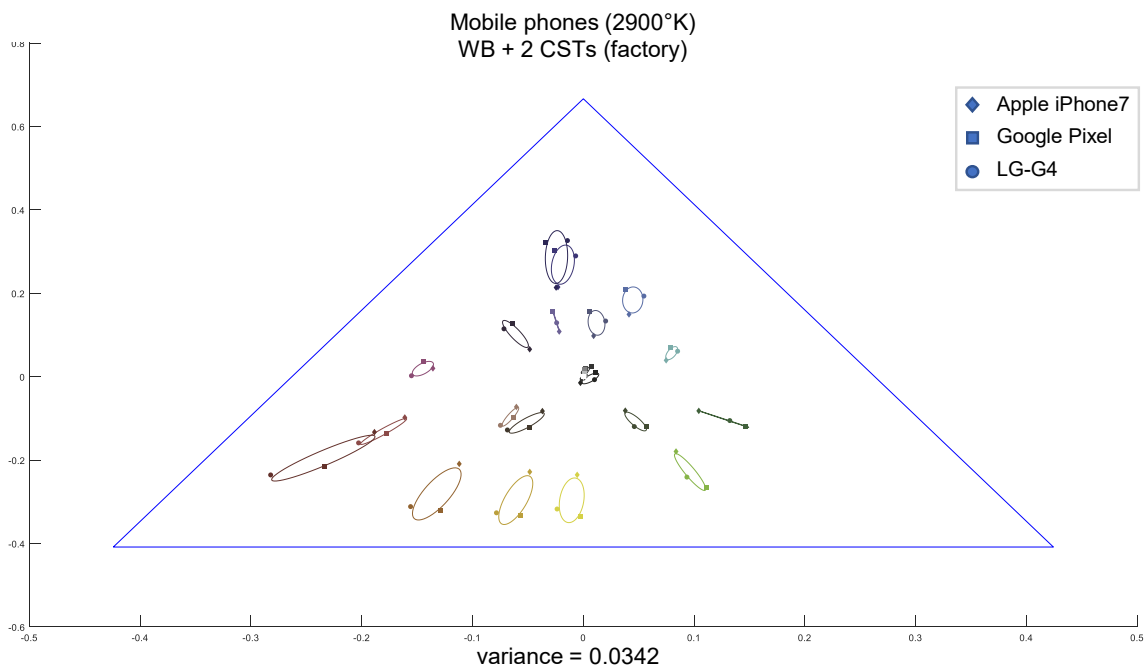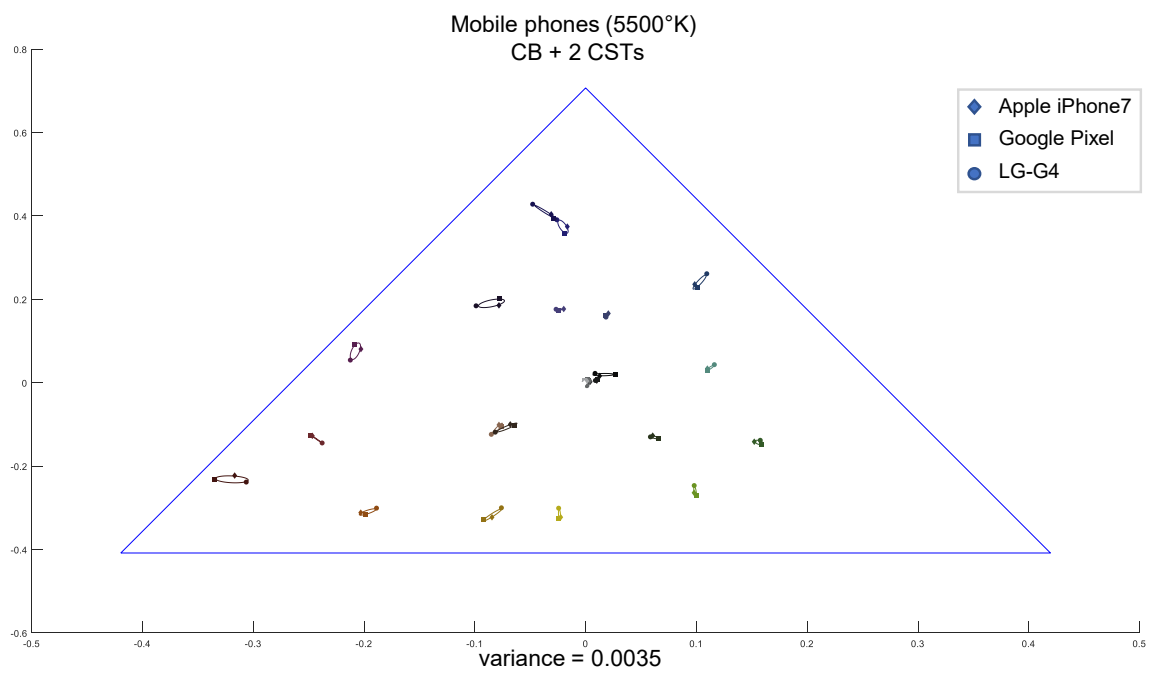
Figure A.1: WB + CSTs (factory) consistency among mobile phone cameras capturing the same scene under illumination 2900°K.

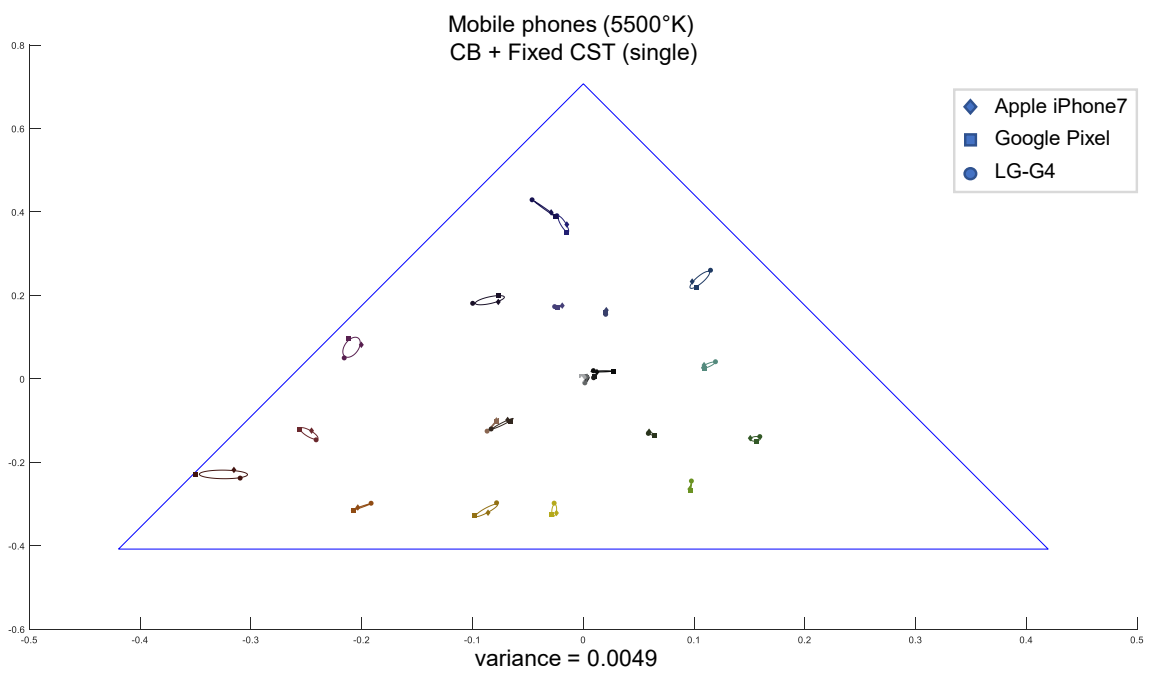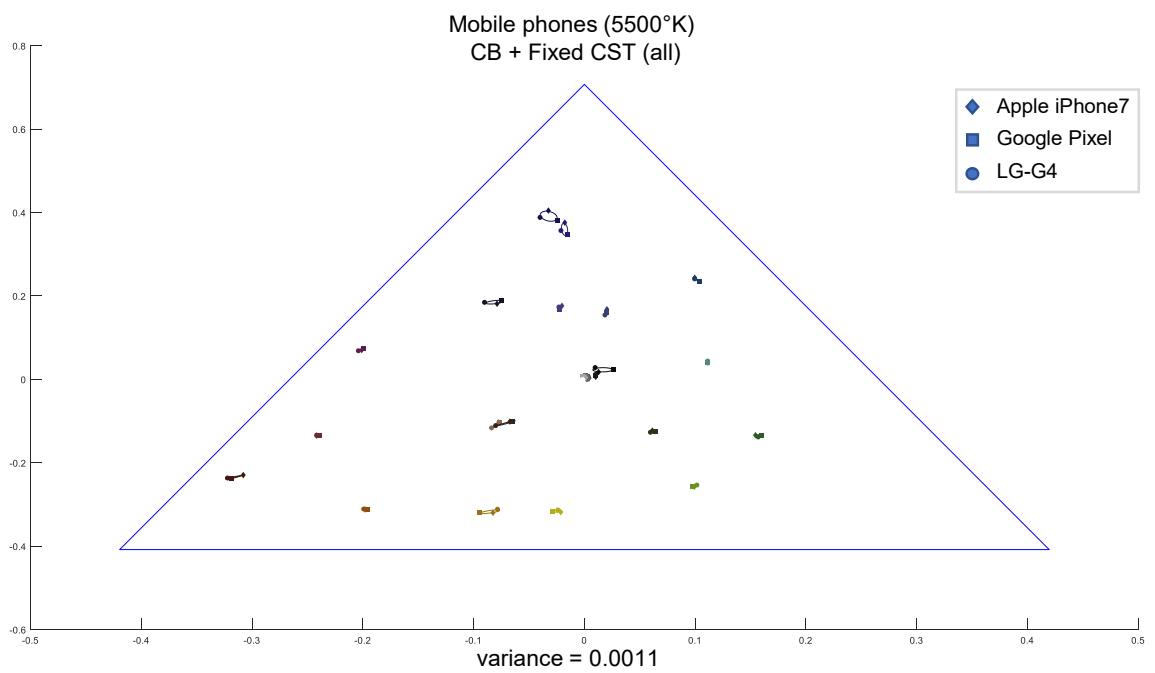phone cameras under illumination 5500°K.

Figures A.19, A.20, A.21, A.22, A.23, and A.24 show the variance for the mobile phone cameras under illumination 6000°K.

Figures A.25, A.26, A.27, A.28, A.29, and A.30 show the variance for the DSLR cameras under illumination 3000°K.

Figures A.31, A.32, A.33, A.34, A.35, and A.36 show the variance for the DSLR cameras under illumination 3500°K.

Figures A.37, A.38, A.39, A.40, A.41, and A.42 show the variance for the DSLR cameras under illumination 4300°K.

Figures A.43, A.44, A.45, A.46, A.47, and A.48 show the variance for the DSLR cameras under illumination 5200°K.
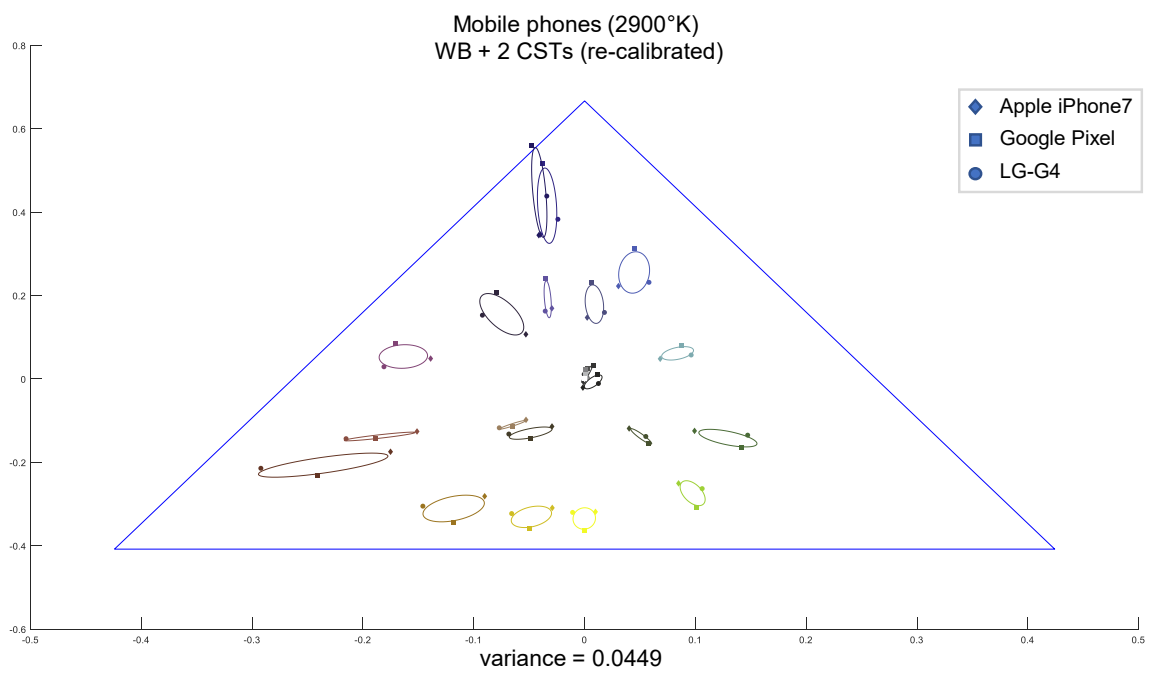
Figure A.2: WB + CSTs (re-calibrated) consistency among mobile phone cameras capturing the same scene under illumination 2900°K.
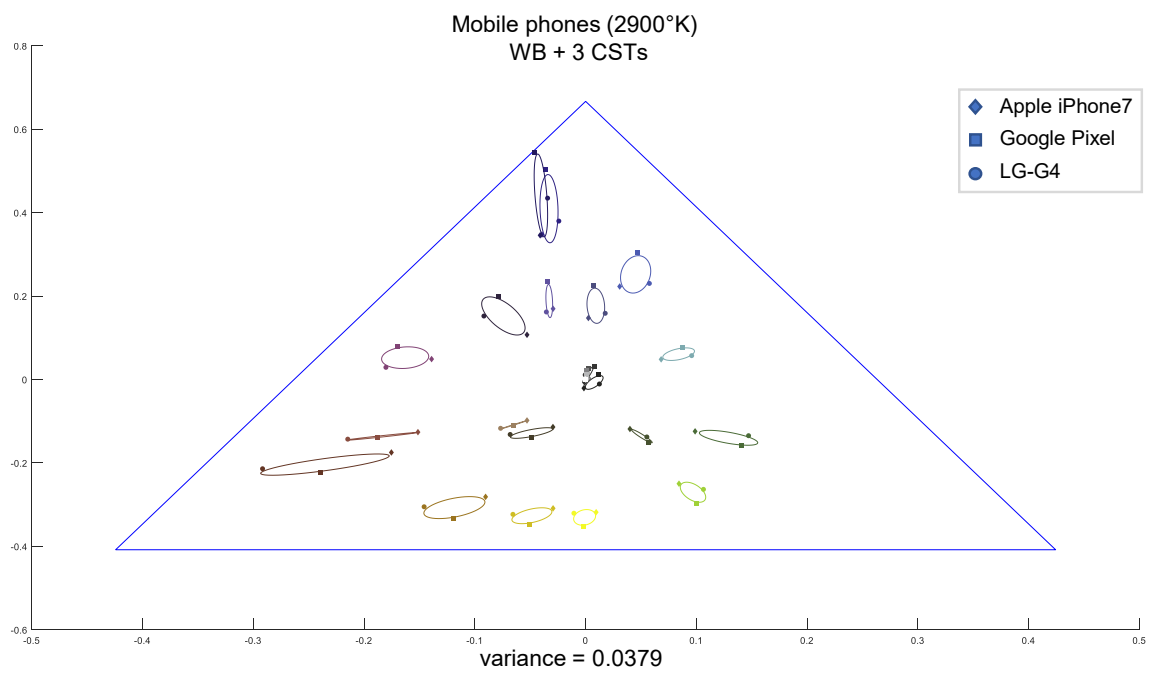
Figure A.3: WB + 3 CSTs consistency among mobile phone cameras capturing the same scene under illumination 2900°K.
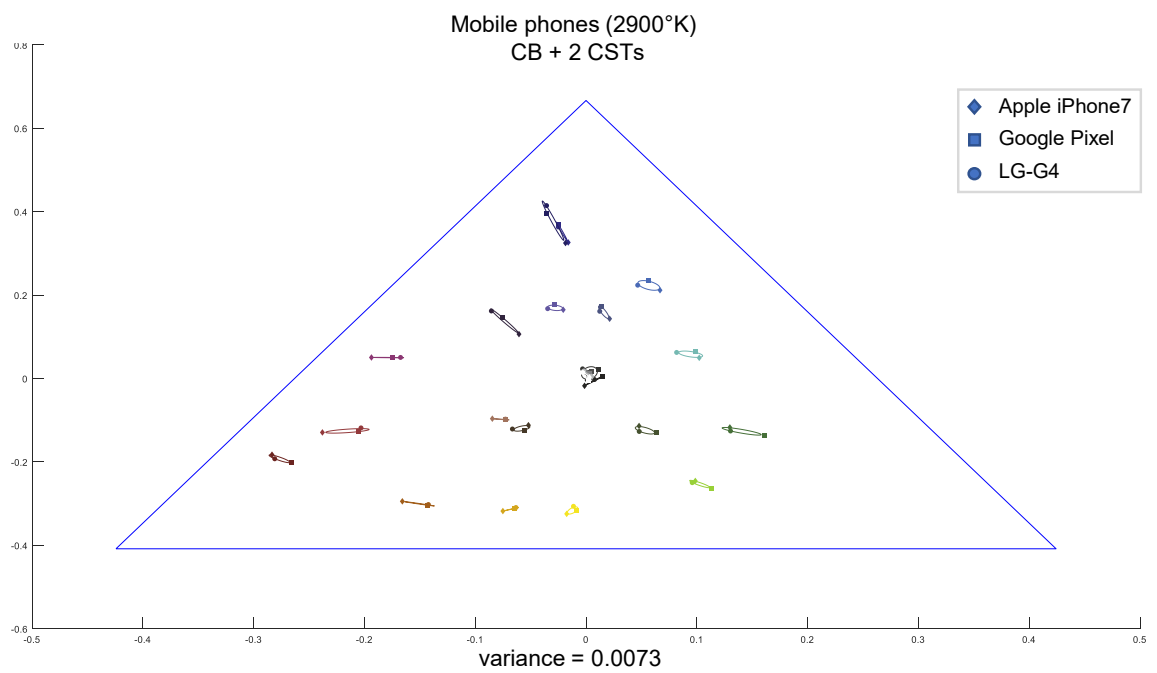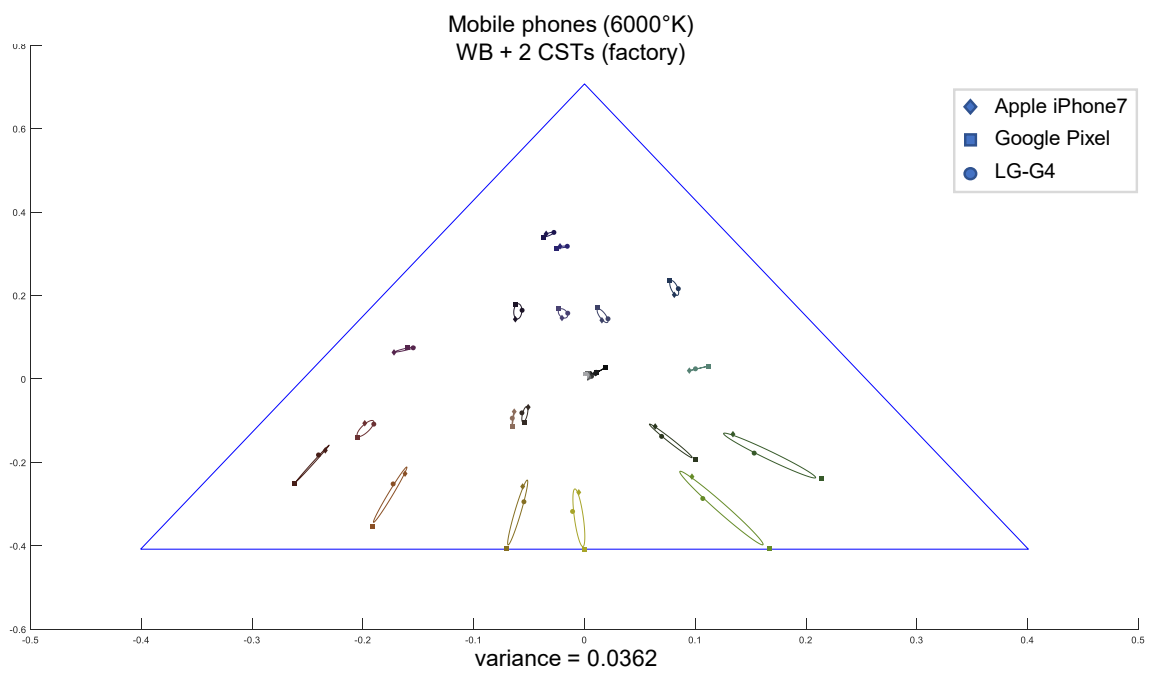
Figure A.4: CB + 2 CSTs consistency among mobile phone cameras capturing the same scene under illumination 2900°K.
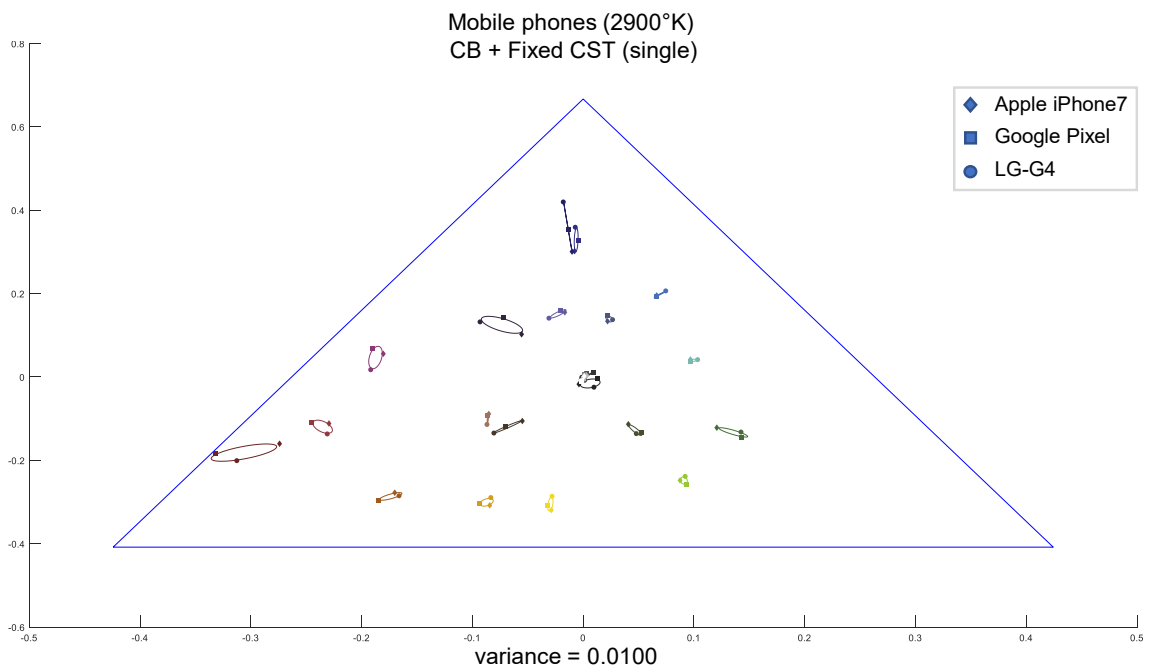
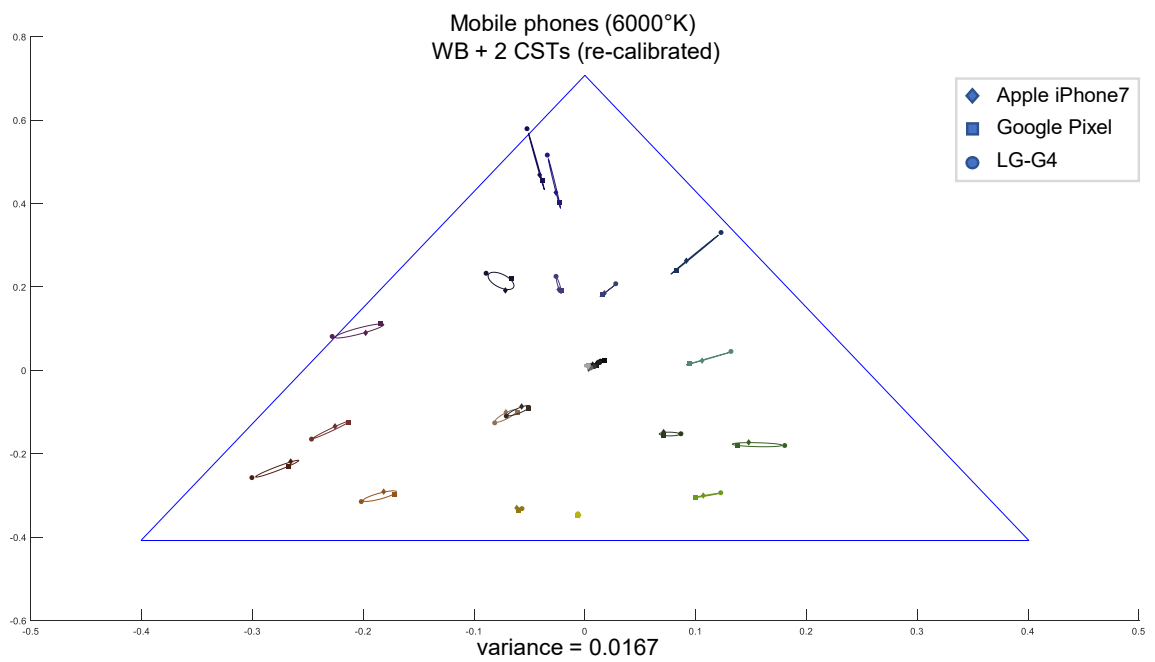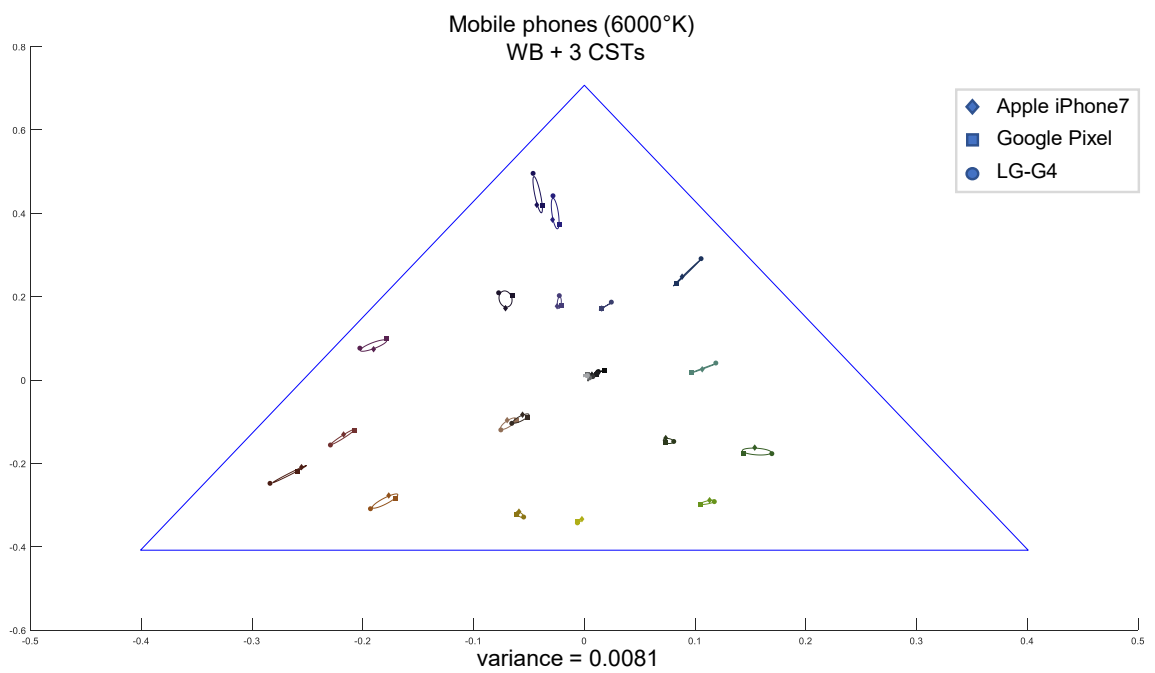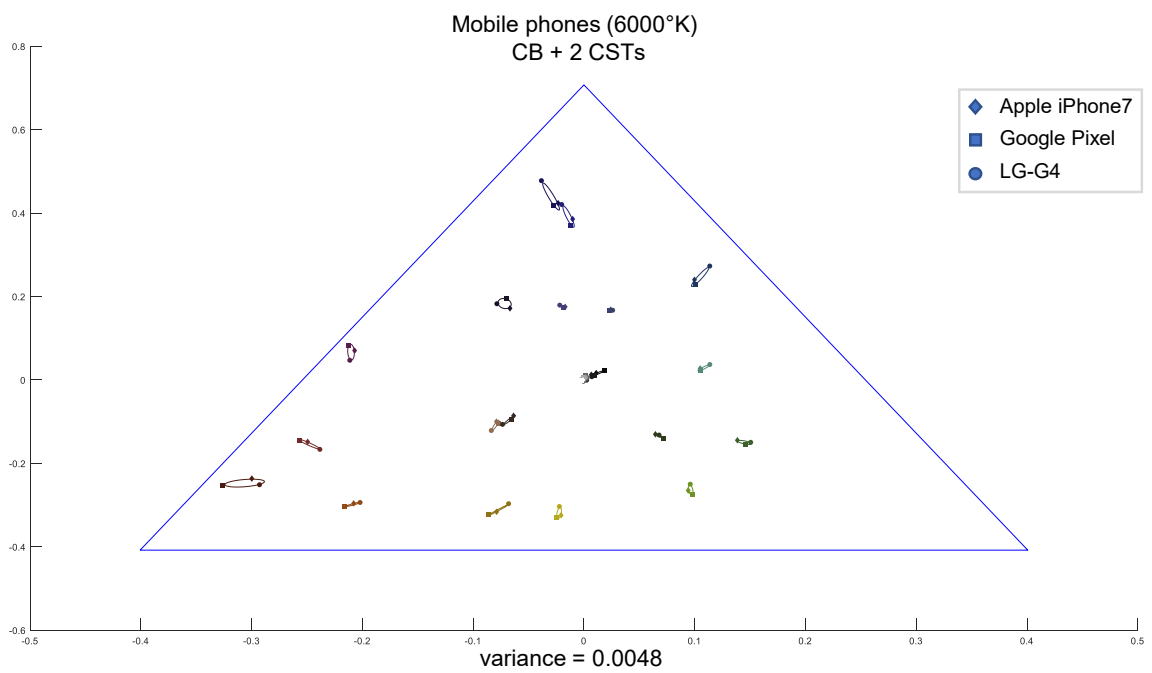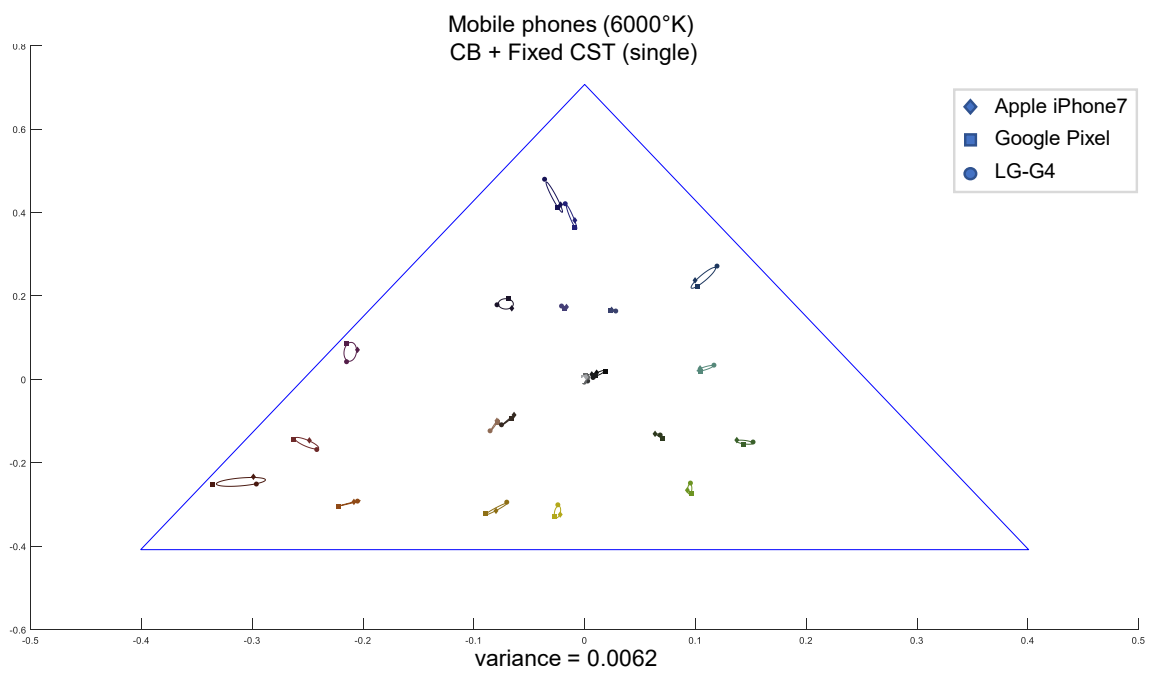**Figure A.5:** CB + Fixed CST (single) consistency among mobile phone cameras capturing the same scene under illumination 2900°K.
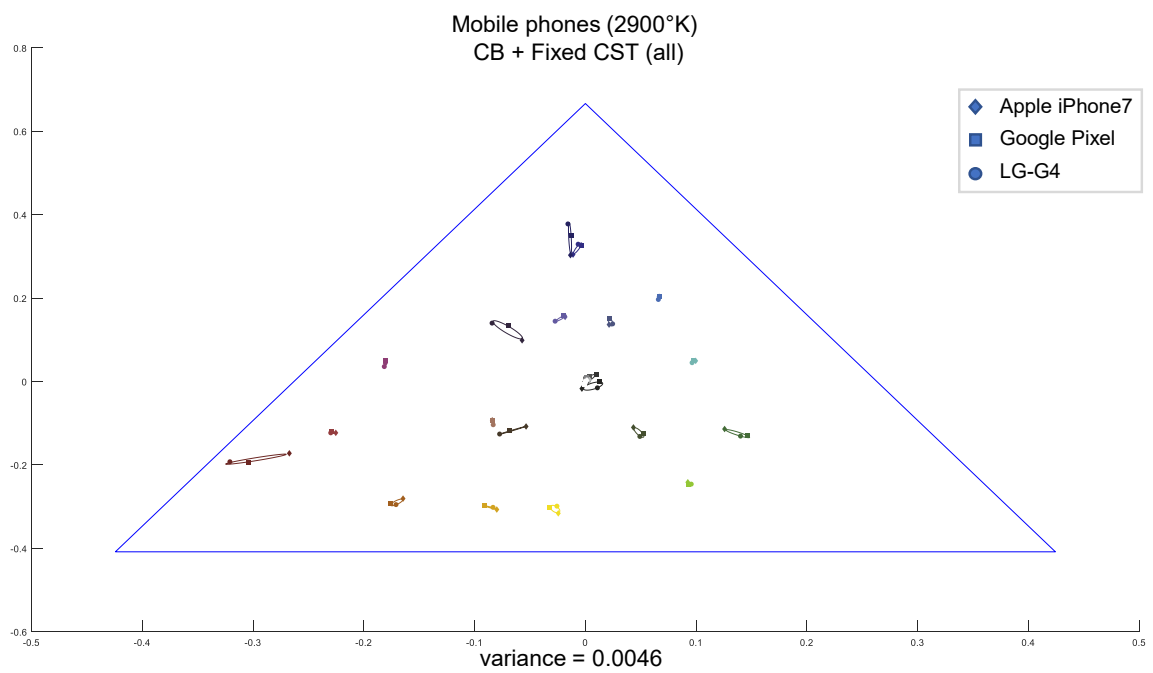
Figure A.6: CB + Fixed CST (all) consistency among mobile phone cameras capturing the same scene under illumination 2900°K.
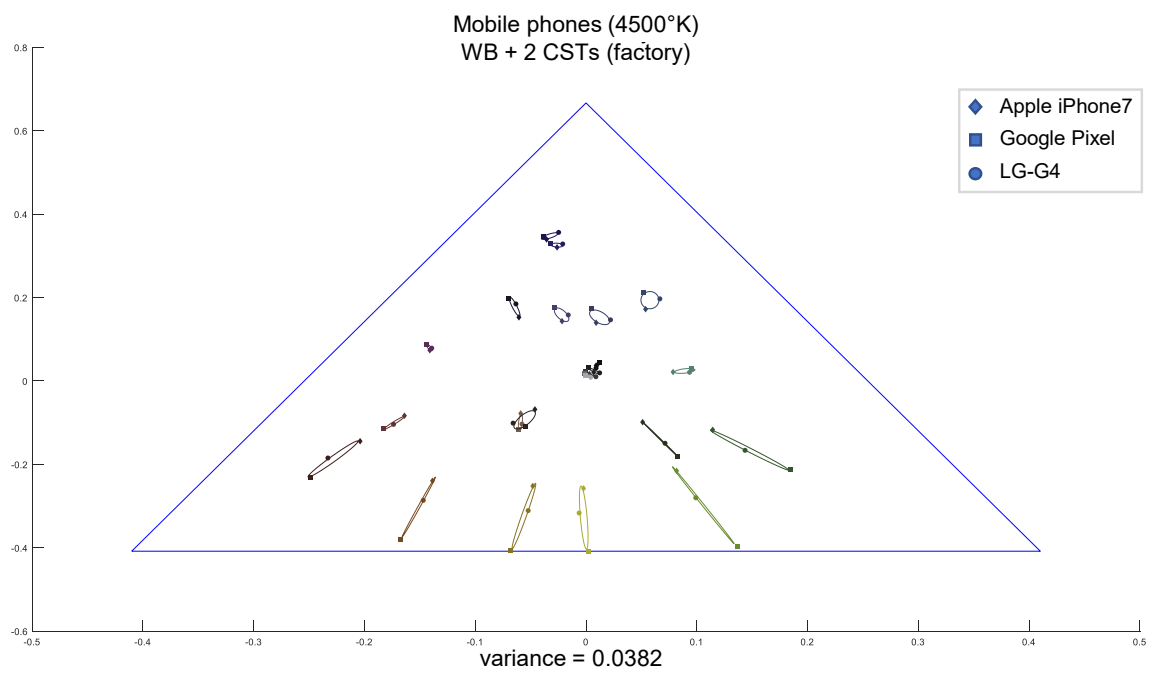
Figure A.7: WB + CSTs (factory) consistency among mobile phone cameras capturing the same scene under illumination 4500°K.
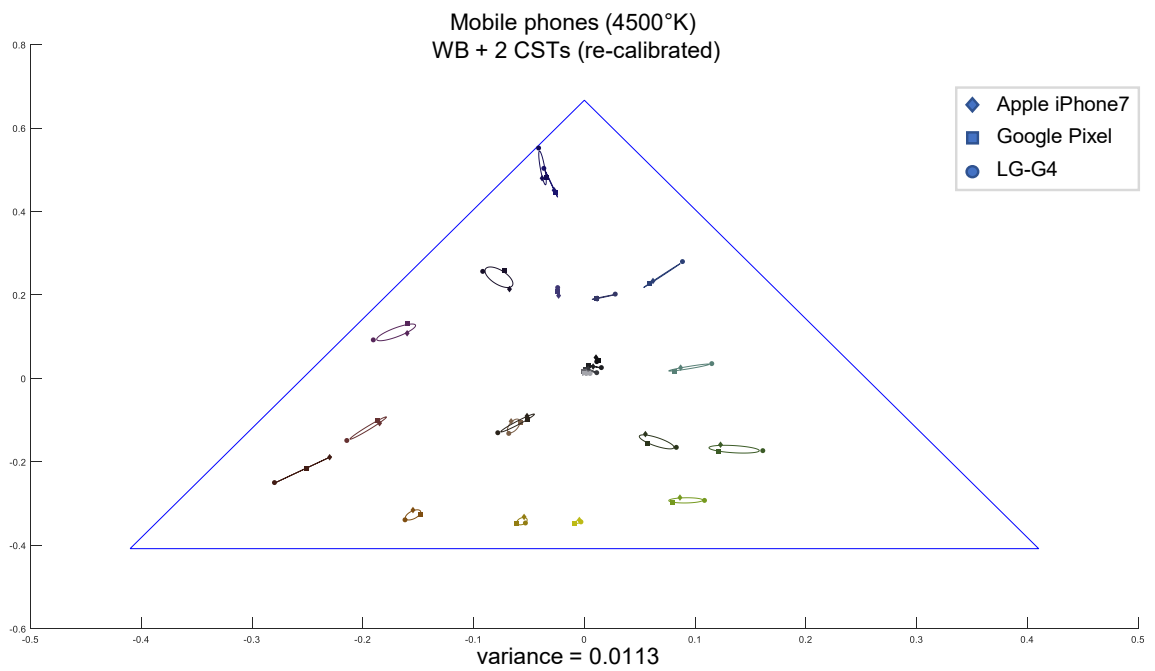
Figure A.8: WB + CSTs (re-calibrated) consistency among mobile phone cameras capturing the same scene under illumination 4500°K.
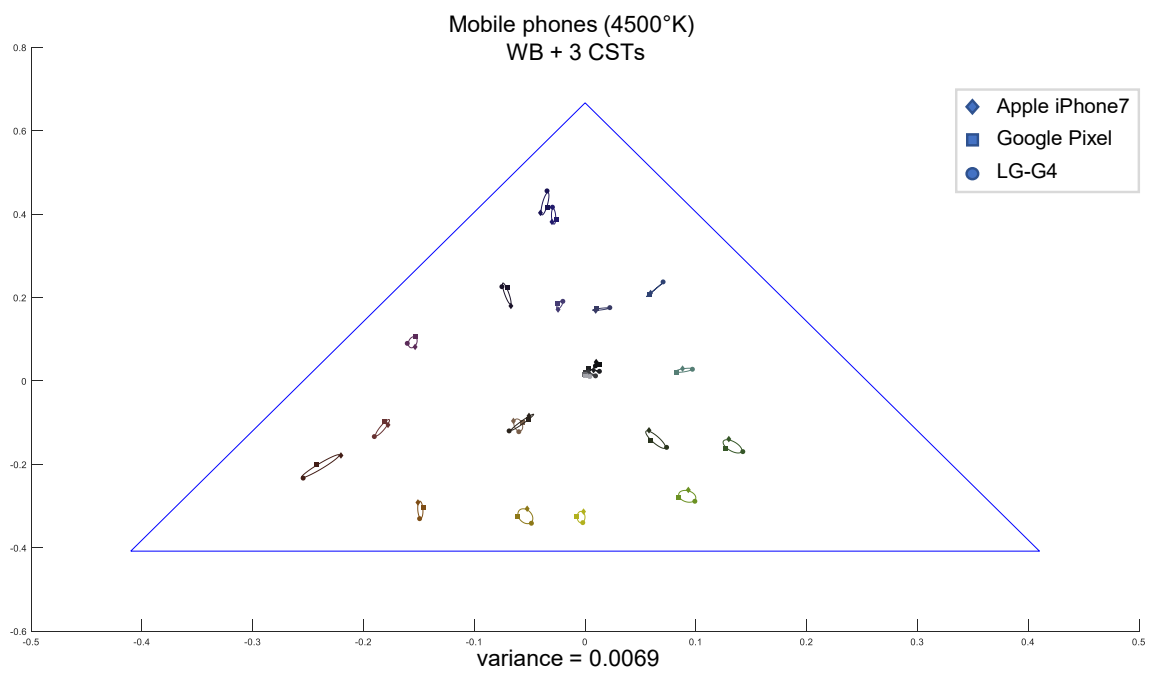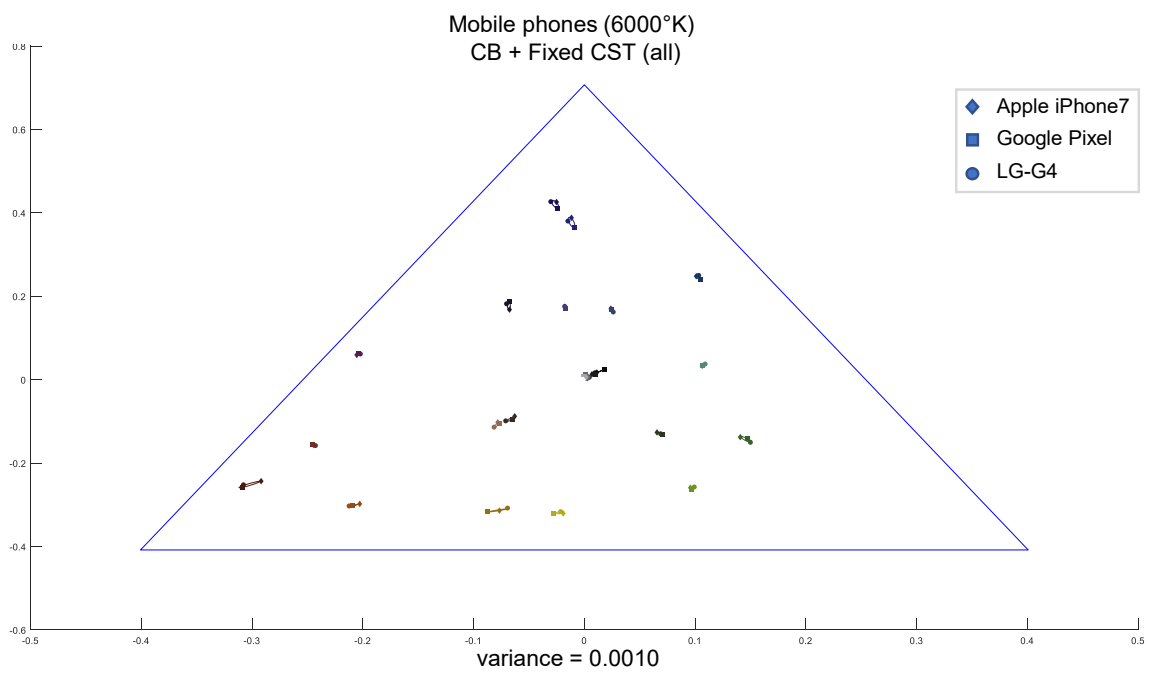
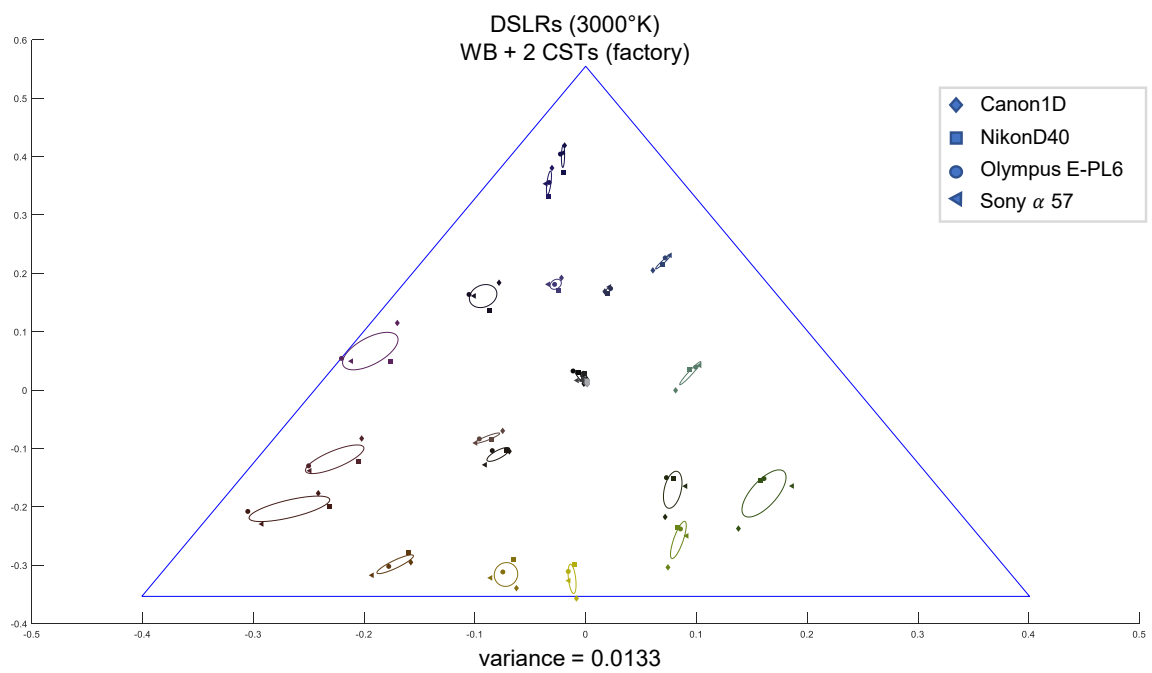**Figure A.9:** WB + 3 CSTs consistency among mobile phone cameras capturing the same scene under illumination 4500°K.
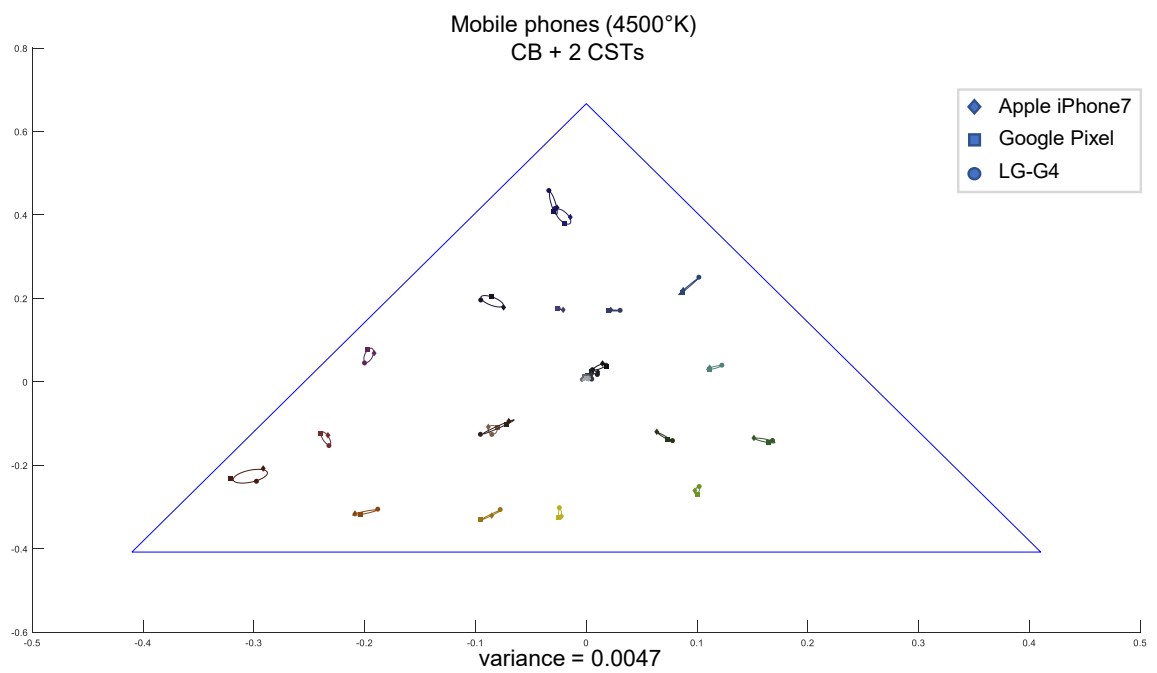
Figure A.10: CB + 2 CSTs consistency among mobile phone cameras capturing the same scene under illumination 4500°K.
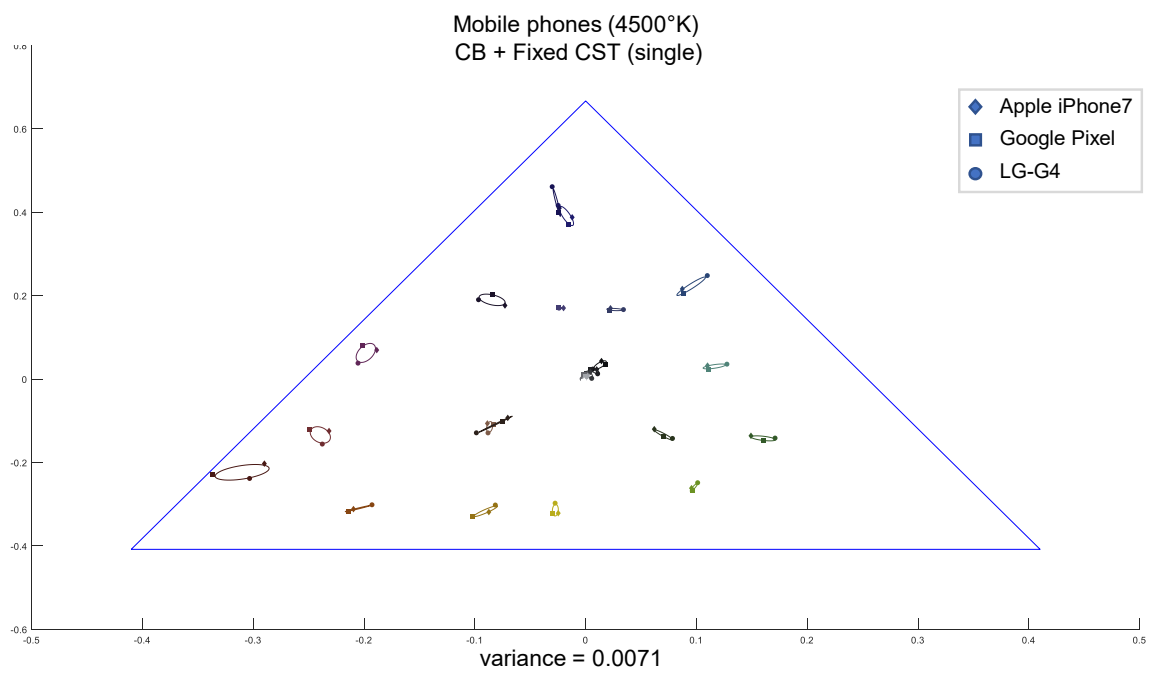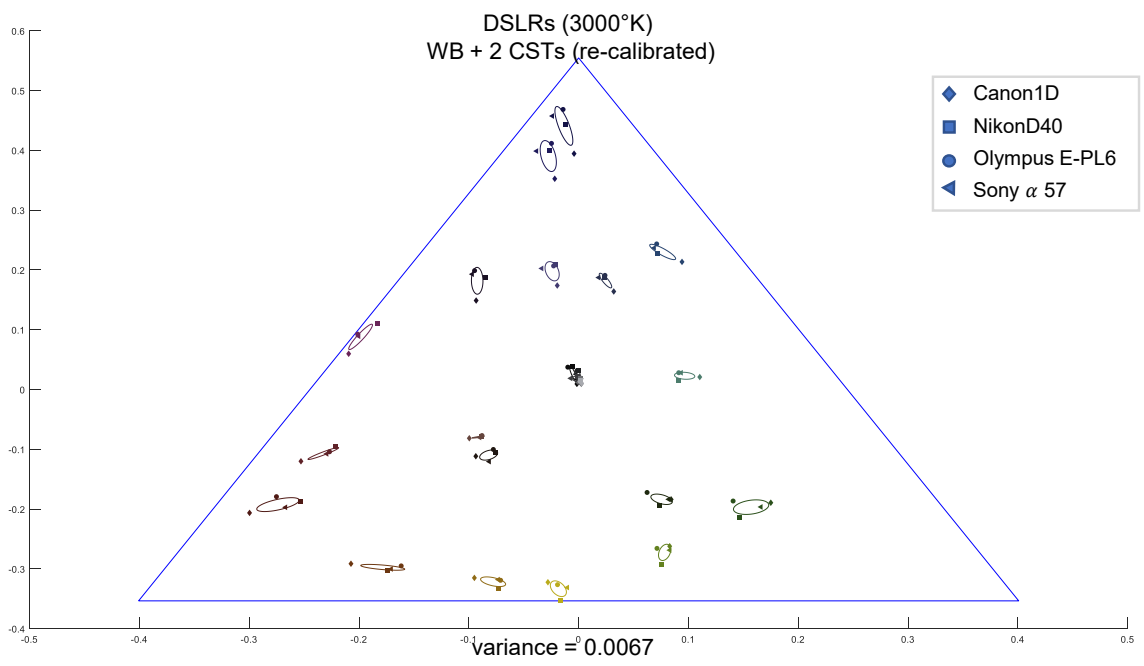
Figure A.11: CB + Fixed CST (single) consistency among mobile phone cameras capturing the same scene under illumination 4500°K.
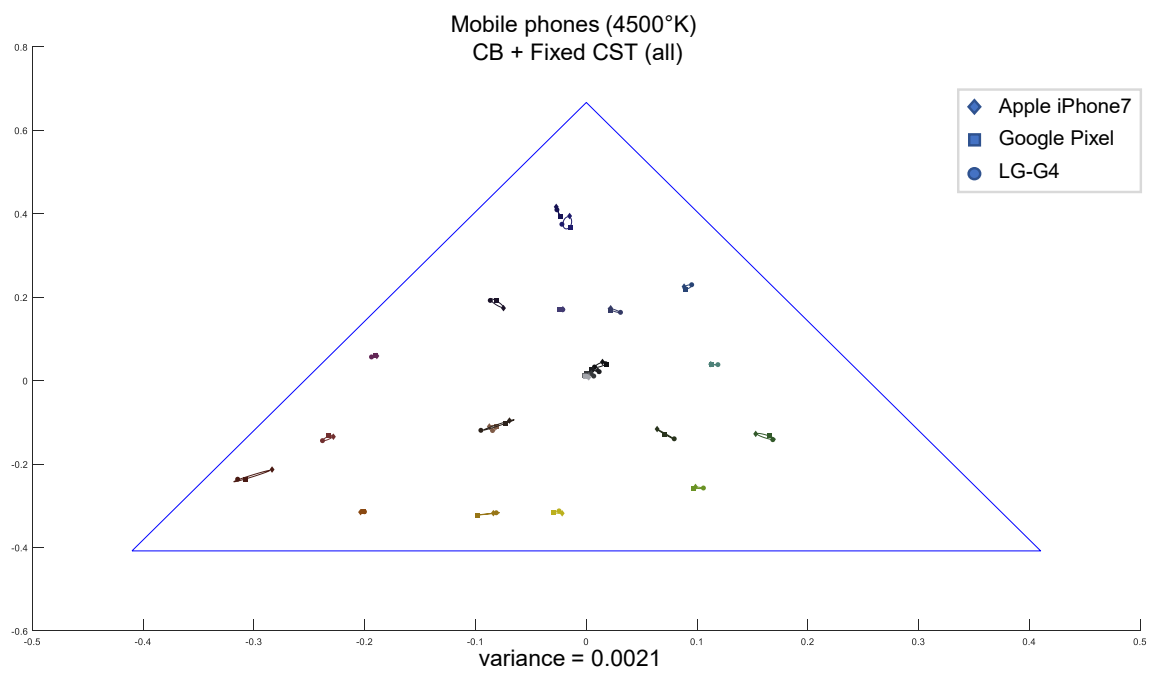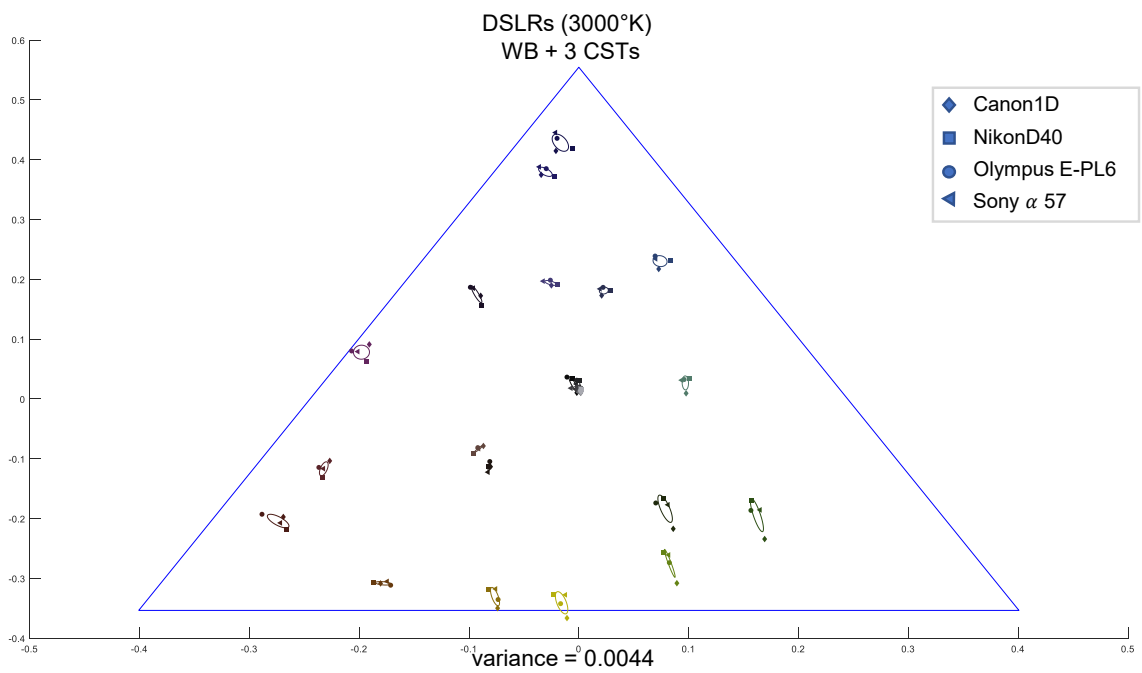
Figure A.12: CB + Fixed CST (all) consistency among mobile phone cameras capturing the same scene under illumination 4500°K.
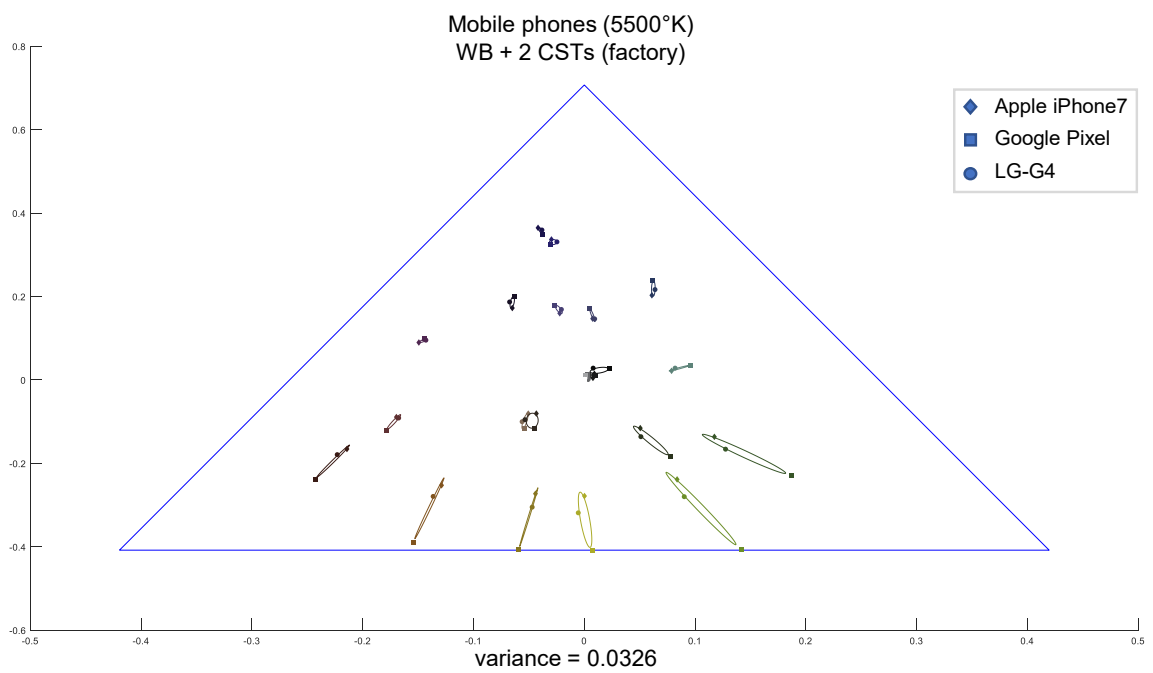
Figure A.13: WB + CSTs (factory) consistency among mobile phone cameras capturing the same scene under illumination 5500°K.
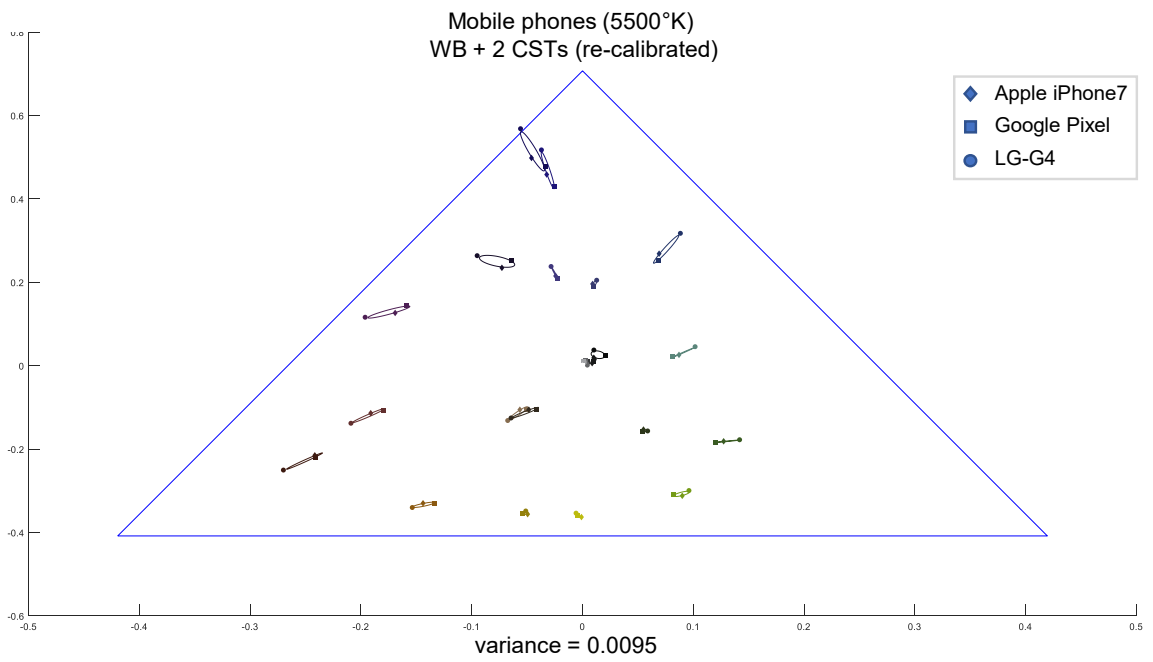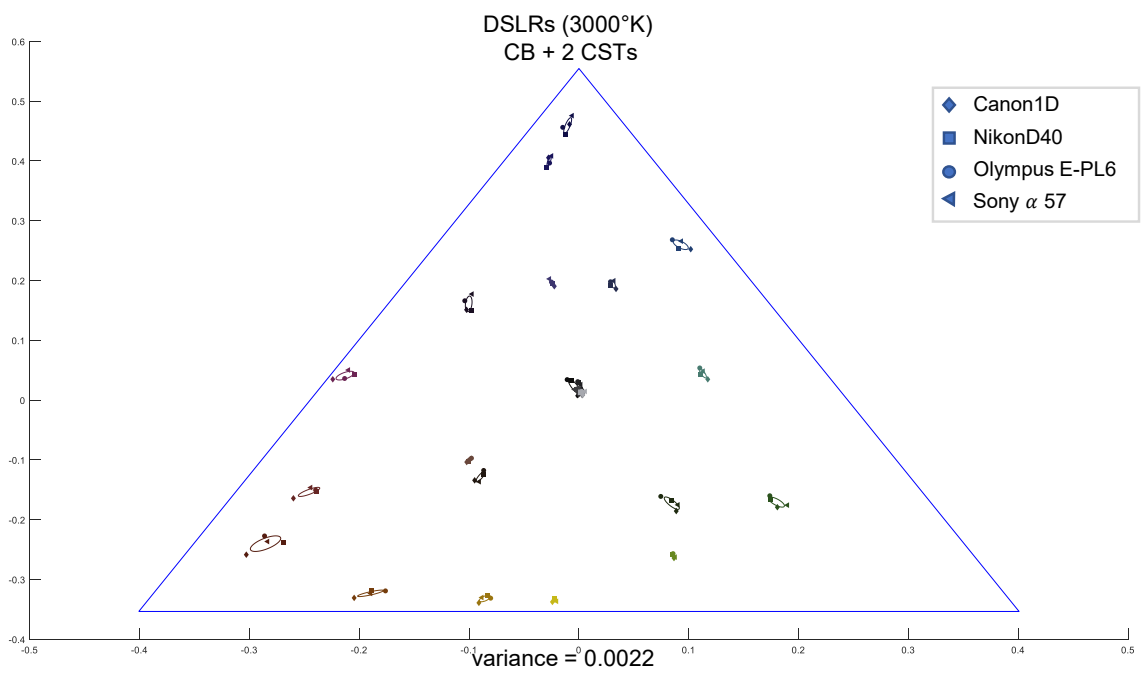
Figure A.14: WB + CSTs (re-calibrated) consistency among mobile phone cameras capturing the same scene under illumination 5500°K.
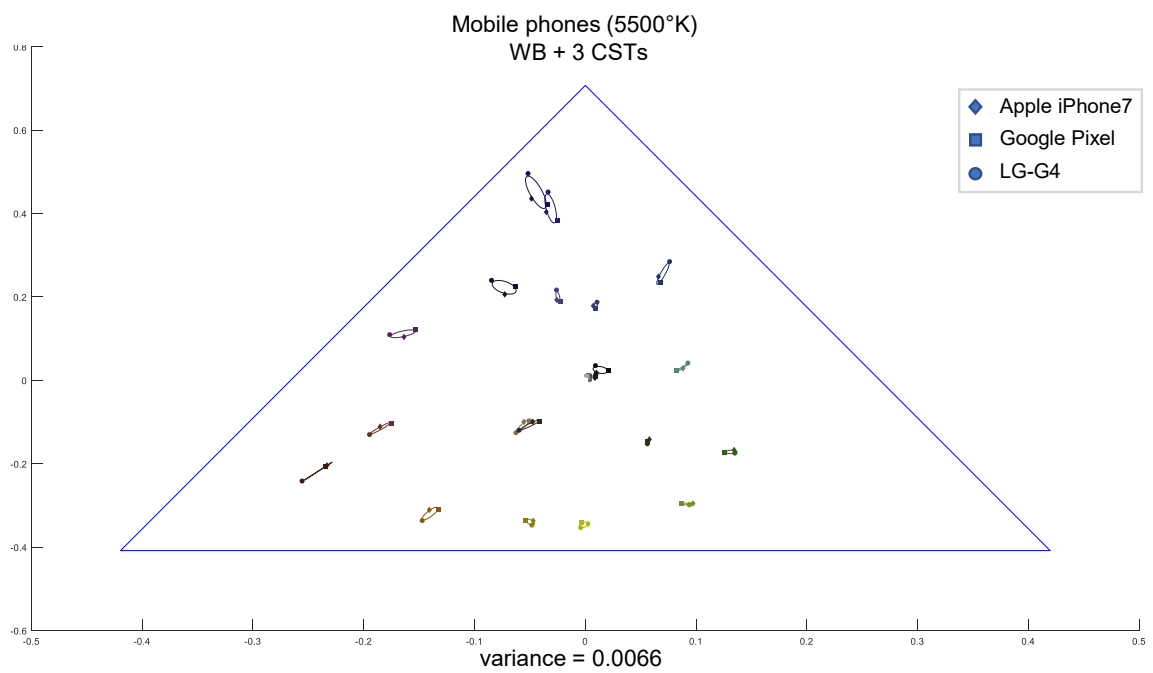
Figure A.15: WB + 3 CSTs consistency among mobile phone cameras capturing the same scene under illumination 5500°K.

Figure A.16: CB + 2 CSTs consistency among mobile phone cameras capturing the same scene under illumination 5500°K.

Figure A.17: CB + Fixed CST (single) consistency among mobile phone cameras capturing the same scene under illumination 5500°K.

Figure A.18: CB + Fixed CST (all) consistency among mobile phone cameras capturing the same scene under illumination 5500°K.

Figure A.19: WB + CSTs (factory) consistency among mobile phone cameras capturing the same scene under illumination 6000°K.

Figure A.20: WB + CSTs (re-calibrated) consistency among mobile phone cameras capturing the same scene under illumination 6000°K.
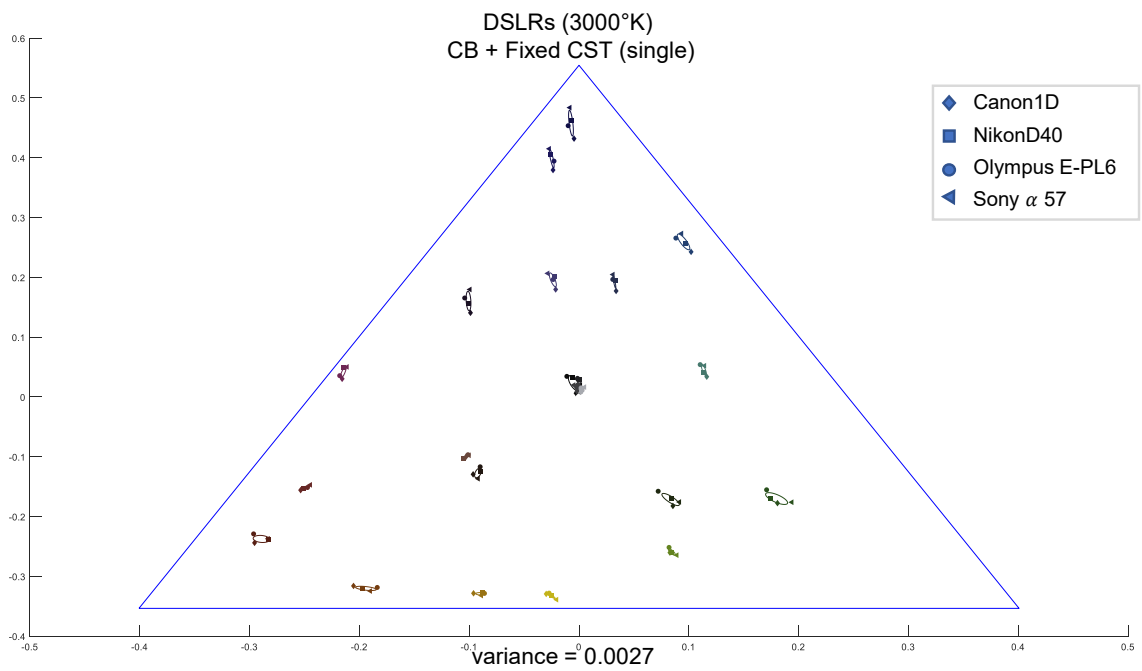
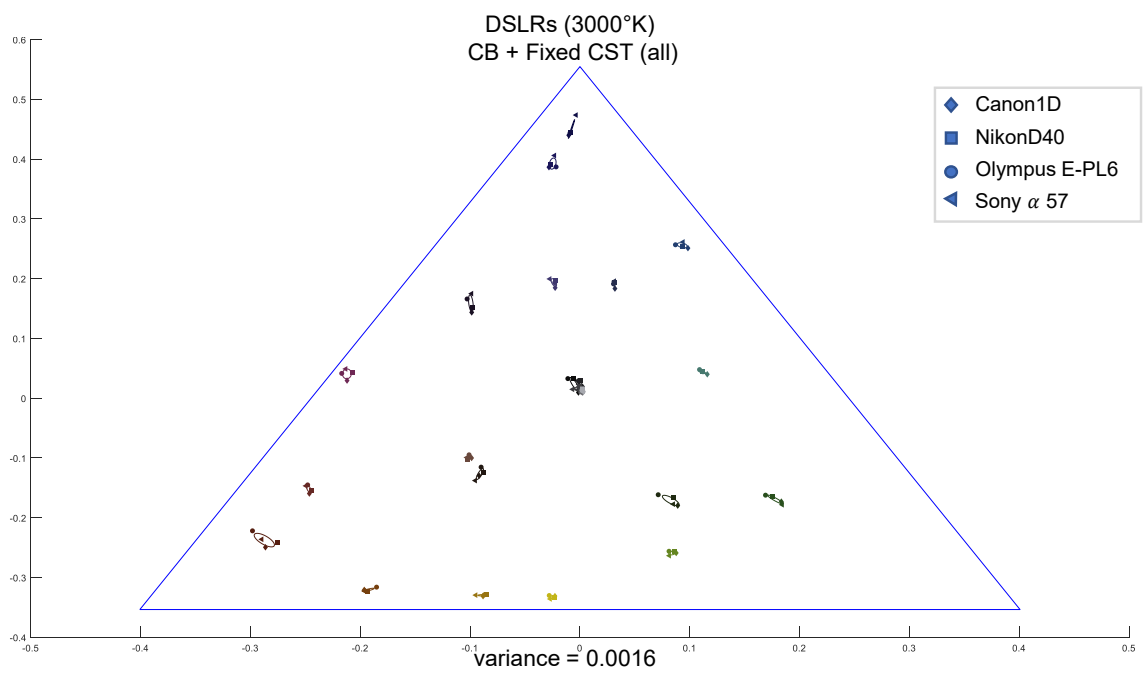**Figure A.21:** WB + 3 CSTs consistency among mobile phone cameras capturing the same scene under illumination 6000°K.
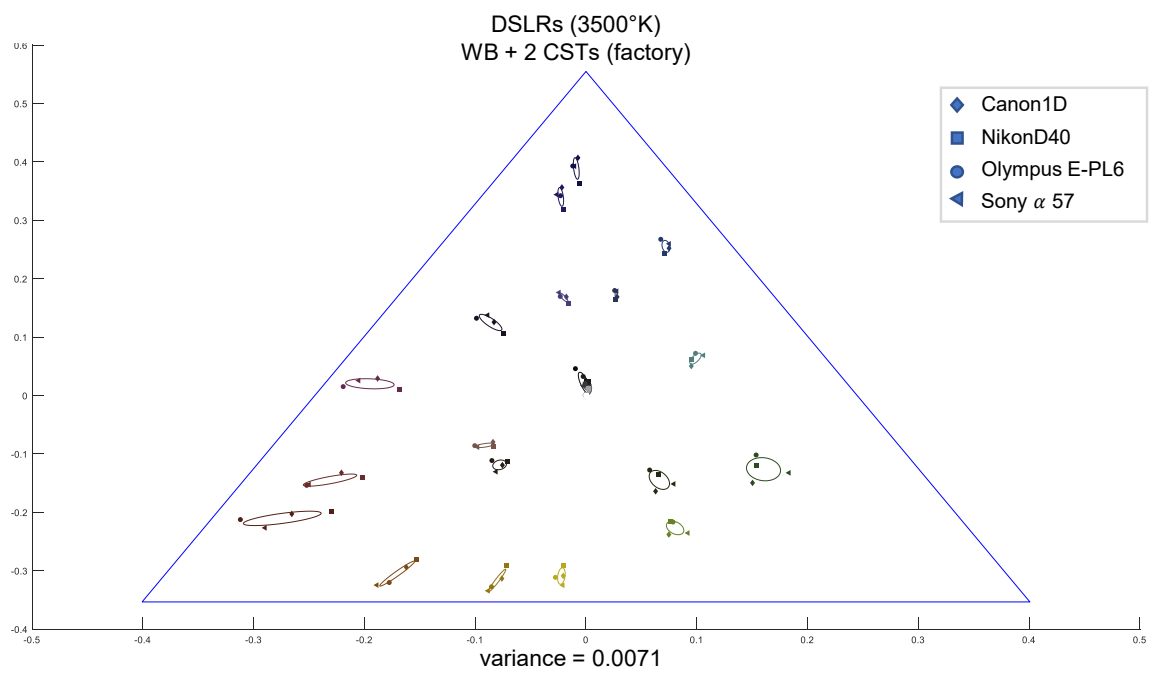
Figure A.22: CB + 2 CSTs consistency among mobile phone cameras capturing the same scene under illumination 6000°K.
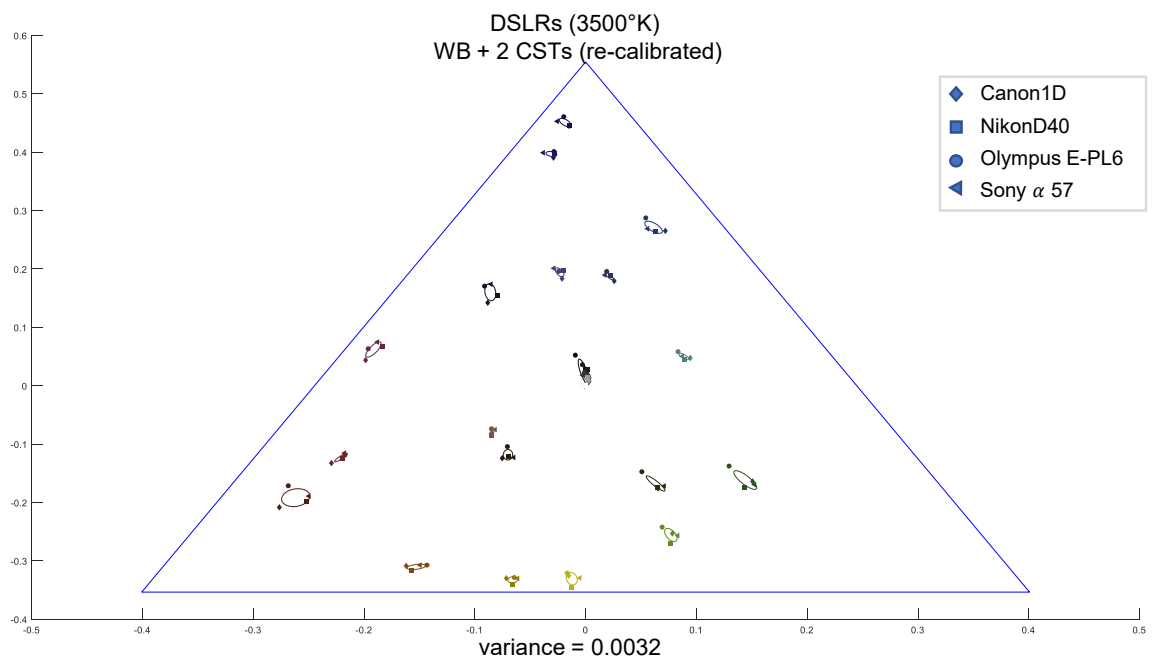
Figure A.23: CB + Fixed CST (single)consistency among mobile phone cameras capturing the same scene under illumination 6000°K.
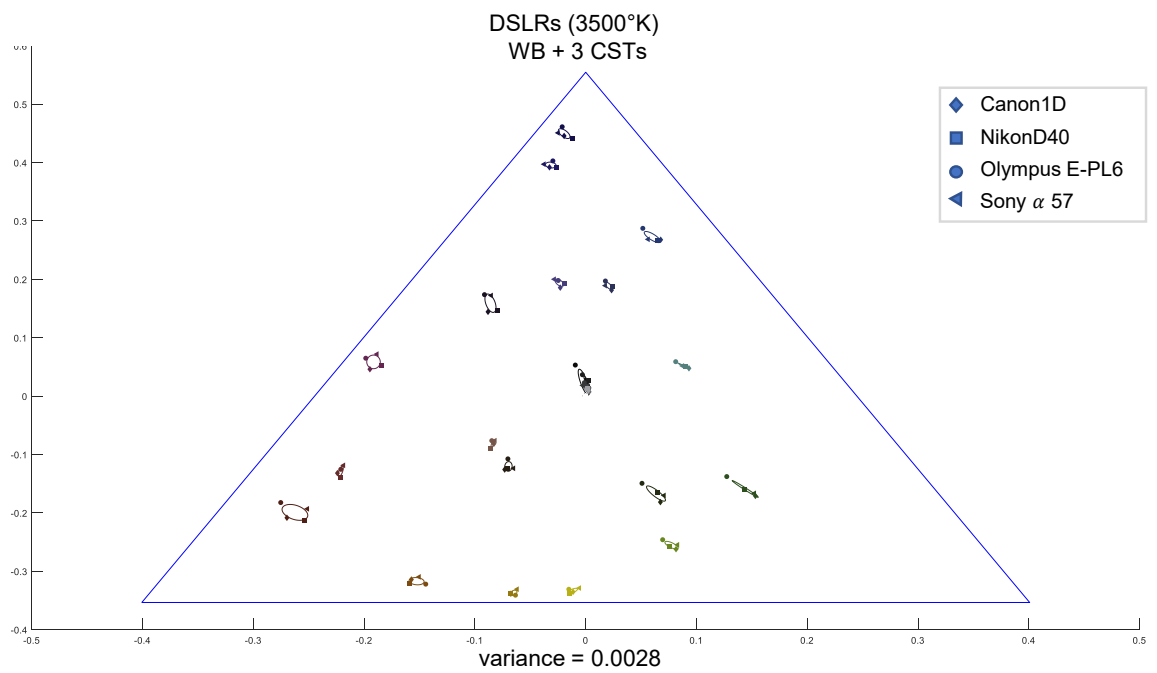
Figure A.24: CB + Fixed CST (all) consistency among mobile phone cameras capturing the same scene under illumination 6000°K.
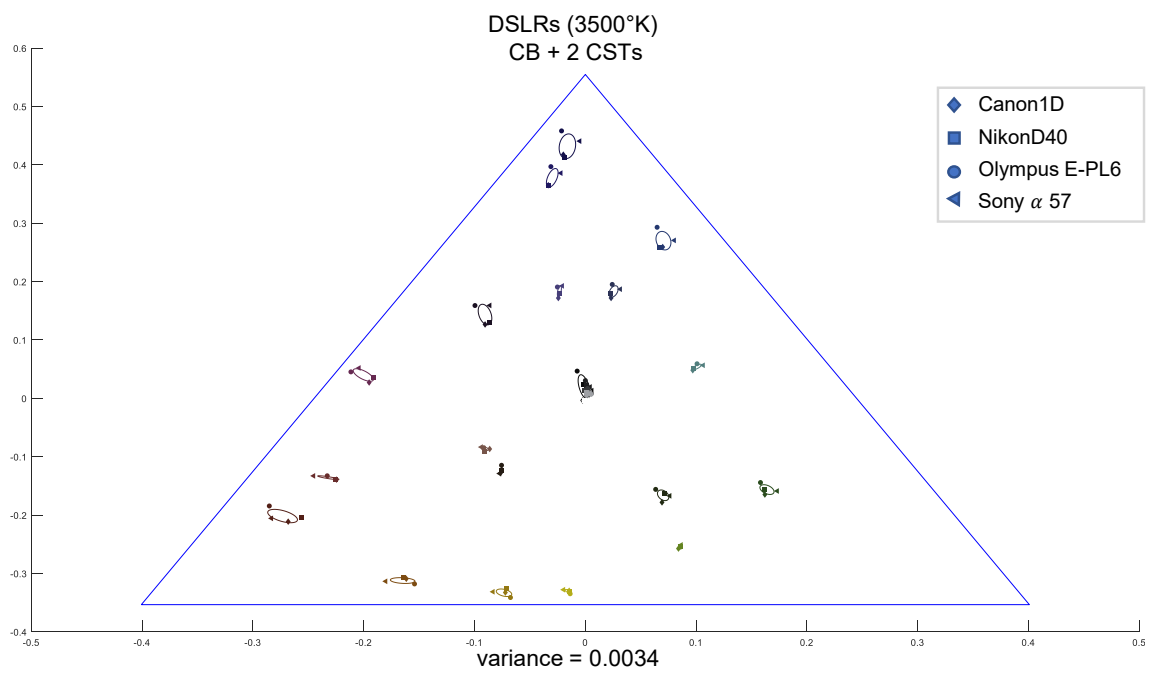
Figure A.25: WB + CSTs (factory) consistency among DSLR cameras capturing the same scene under illumination 3000°K.

Figure A.26: WB + CSTs (re-calibrated) consistency among DSLR cameras capturing the same scene under illumination 3000°K.

Figure A.27: WB + 3 CSTs consistency among DSLR cameras capturing the same scene under illumination 3000°K.

Figure A.28: CB + 2 CSTs consistency among DSLR cameras capturing the same scene under illumination 3000°K.

Figure A.29: CB + Fixed CST (single) consistency among DSLR cameras capturing the same scene under illumination 3000°K.

Figure A.30: CB + Fixed CST (all) consistency among DSLR cameras capturing the same scene under illumination 3000°K.

Figure A.31: WB + CSTs (factory) consistency among DSLR cameras capturing the same scene under illumination 3500°K.
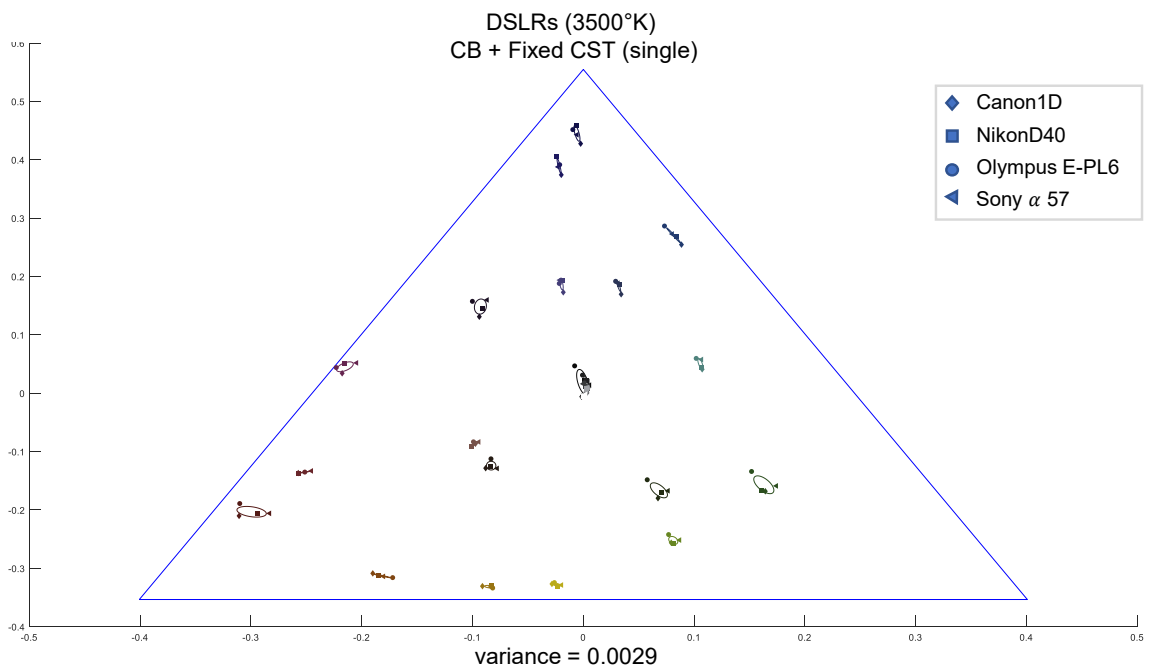
Figure A.32: WB + CSTs (re-calibrated) consistency among DSLR cameras capturing the same scene under illumination 3500°K.

Figure A.33: WB + 3 CSTs consistency among DSLR cameras capturing the same scene under illumination 3500°K.
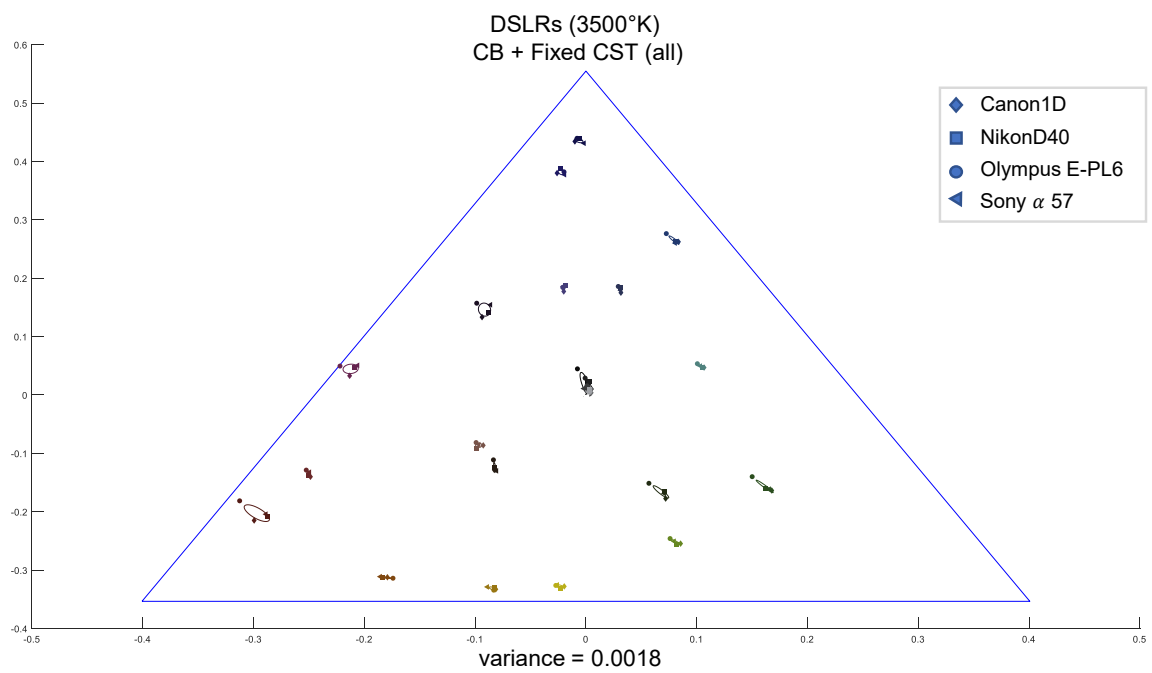
Figure A.34: CB + 2 CSTs consistency among DSLR cameras capturing the same scene under illumination 3500°K.
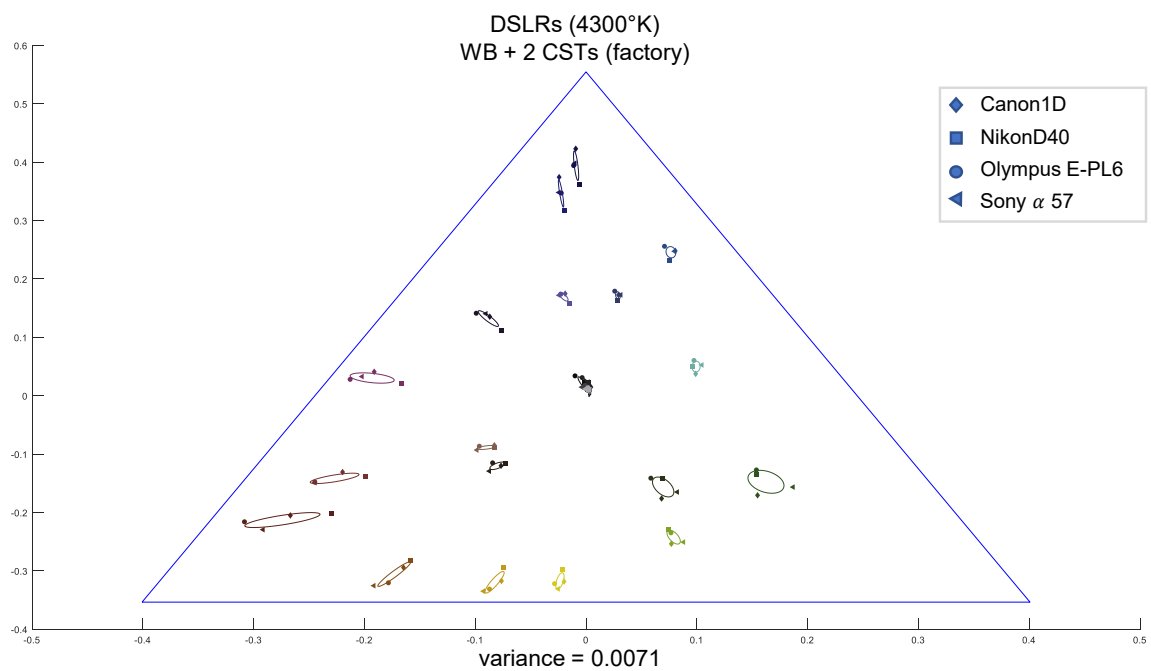
Figure A.35: CB + Fixed CST (single) consistency among DSLR cameras capturing the same scene under illumination 3500°K.
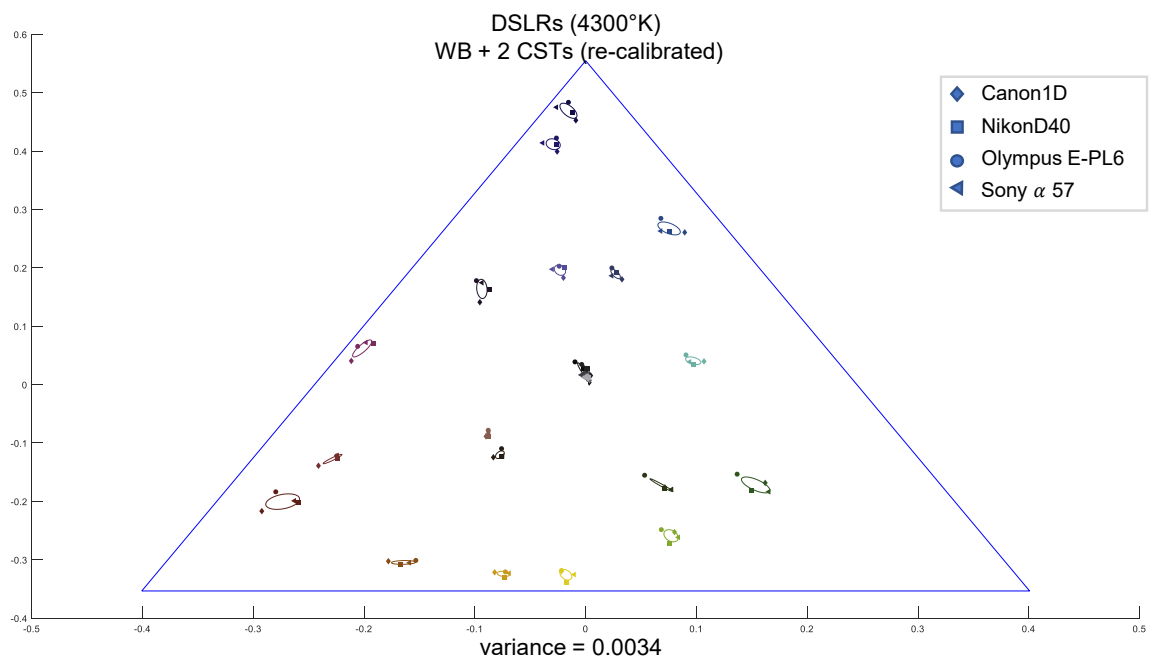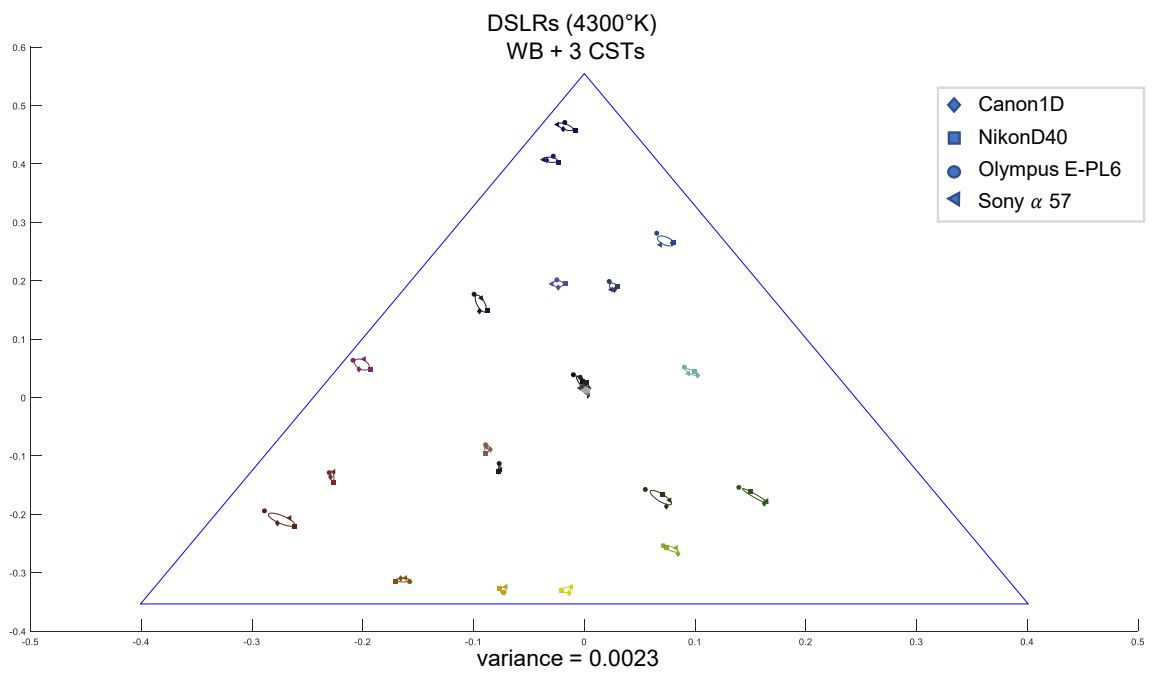
Figure A.36: CB + Fixed CST (all) consistency among DSLR cameras capturing the same scene under illumination 3500°K.
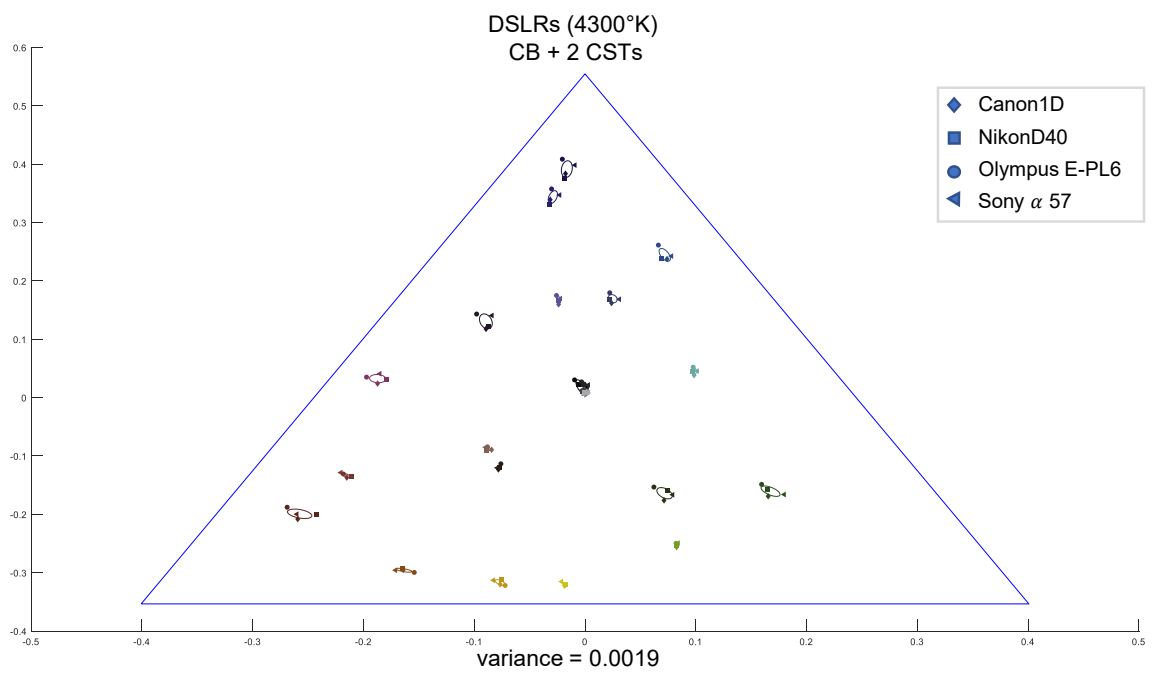
Figure A.37: WB + CSTs (factory) consistency among DSLR cameras capturing the same scene under illumination 4300°K.
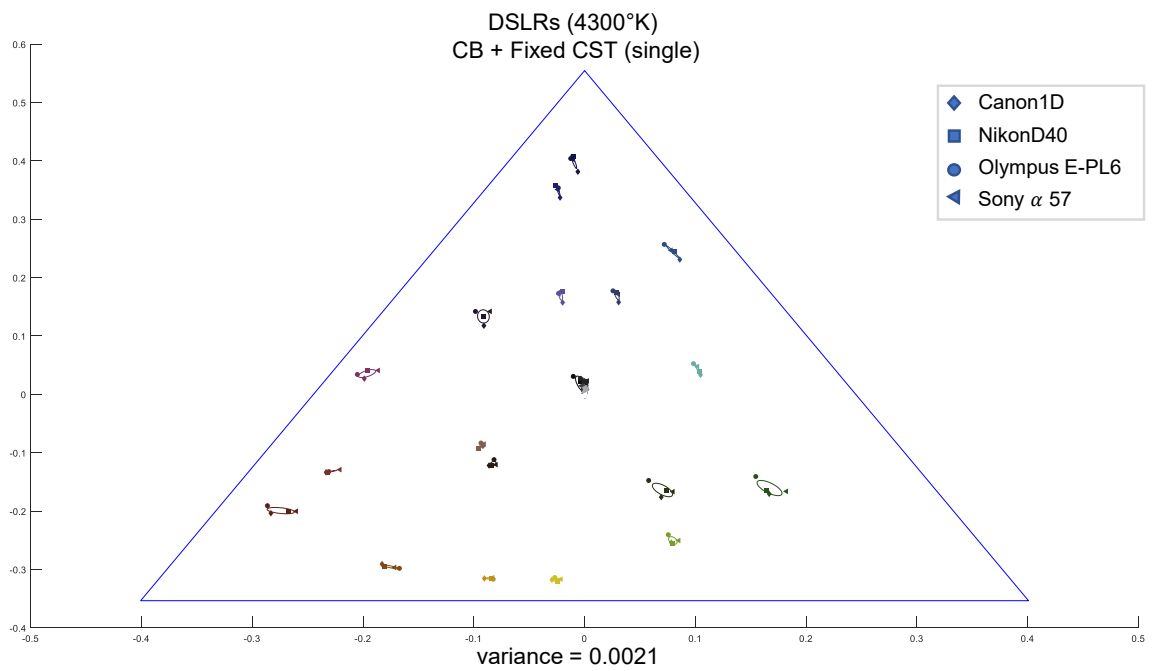
Figure A.38: WB + CSTs (re-calibrated) consistency among DSLR cameras capturing the same scene under illumination 4300°K.

Figure A.39: WB + 3 CSTs consistency among DSLR cameras capturing the same scene under illumination 4300°K.

Figure A.40: CB + 2 CSTs consistency among DSLR cameras capturing the same scene under illumination 4300°K.

Figure A.41: CB + Fixed CST (single) consistency among DSLR cameras capturing the same scene under illumination 4300°K.

Figure A.42: CB + Fixed CST (all) consistency among DSLR cameras capturing the same scene under illumination 4300°K.

Figure A.43: WB + CSTs (factory) consistency among DSLR cameras capturing the same scene under illumination 5200°K.

Figure A.44: WB + CSTs (re-calibrated) consistency among DSLR cameras capturing the same scene under illumination 5200°K.
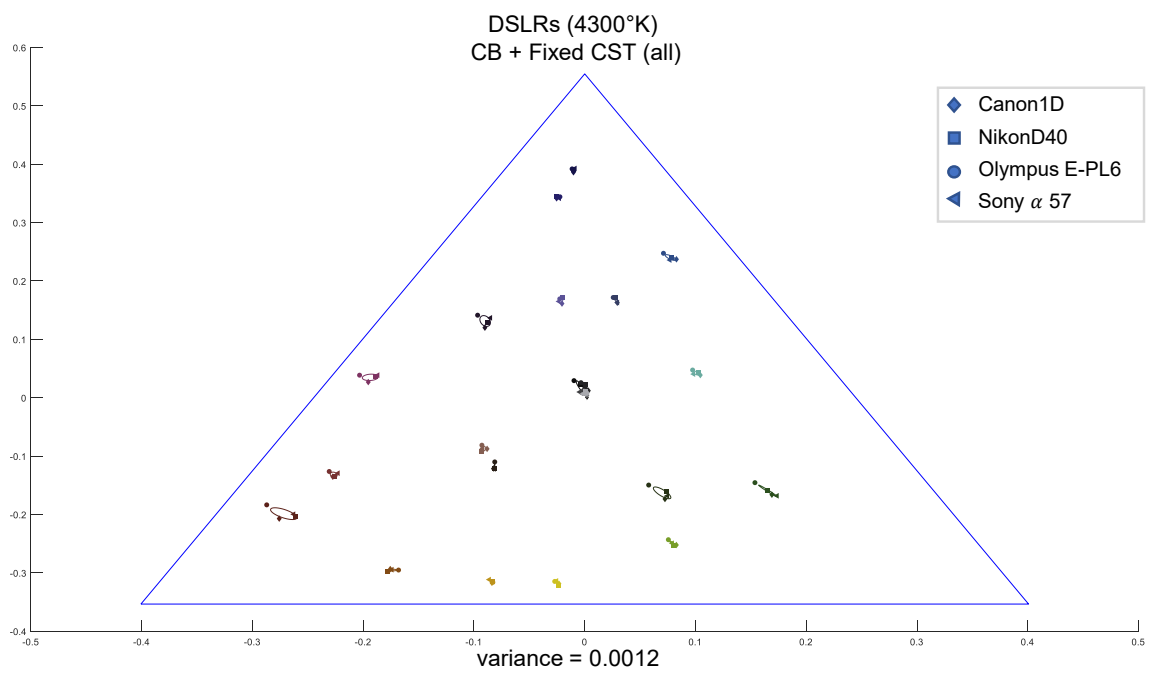
Figure A.45: WB + 3 CSTs consistency among DSLR cameras capturing the same scene under illumination 5200°K.
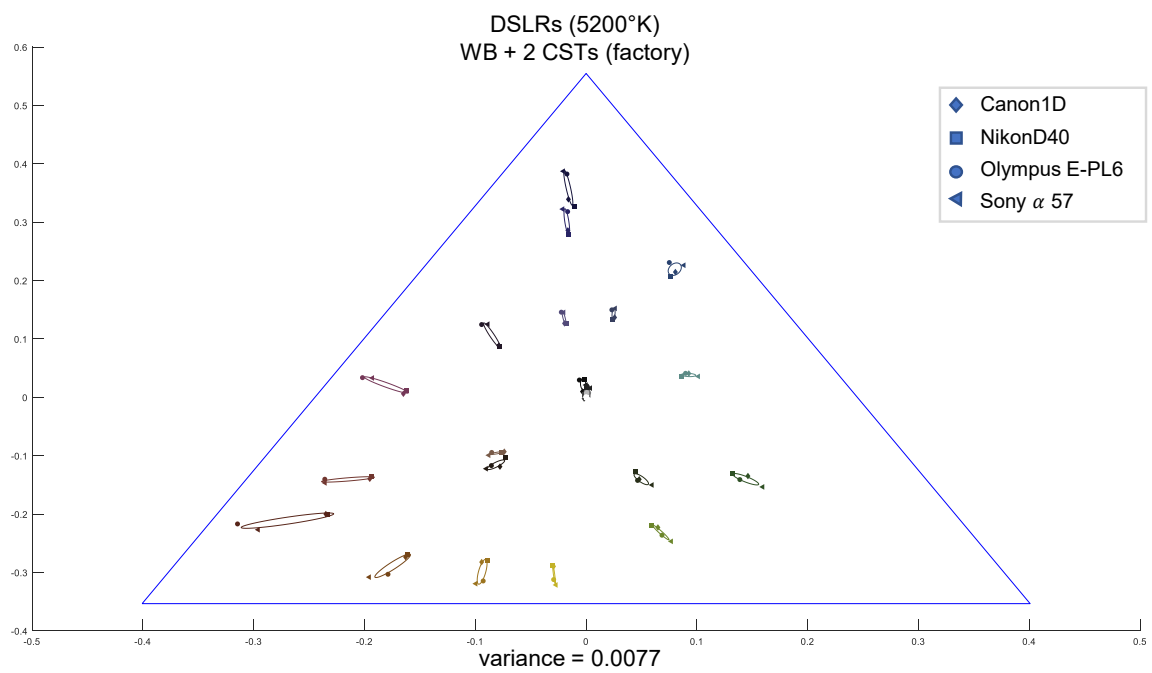
Figure A.46: CB + 2 CSTs consistency among DSLR cameras capturing the same scene under illumination 5200°K.

Figure A.47: CB + Fixed CST (single) consistency among DSLR cameras capturing the same scene under illumination 5200°K.
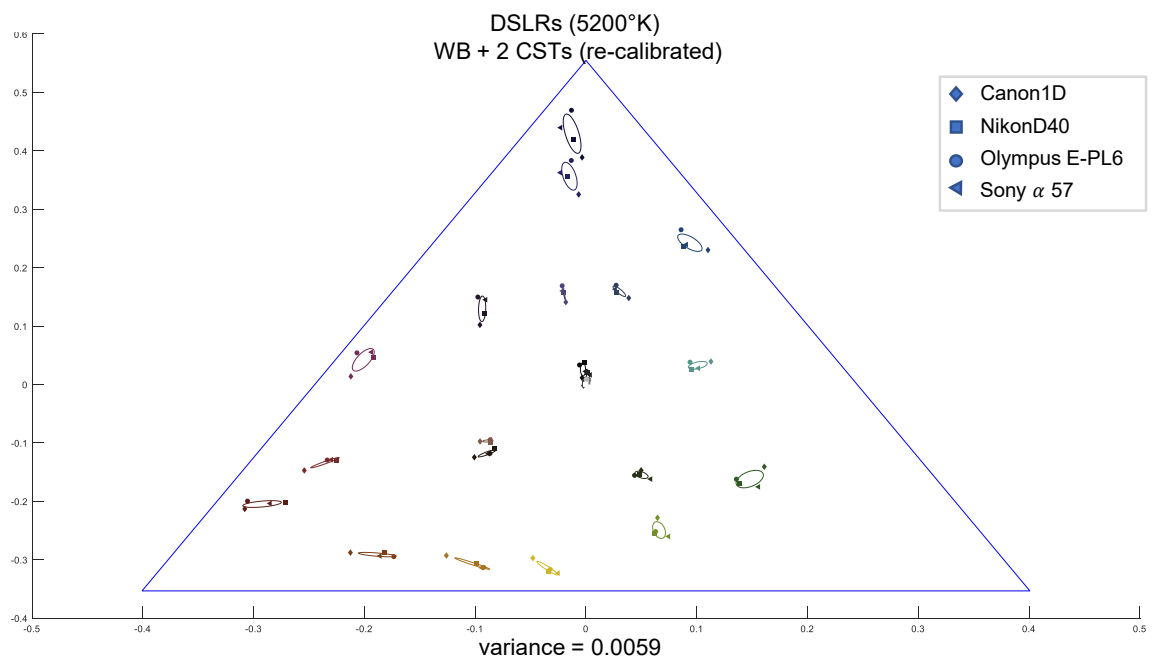
Figure A.48: CB + Fixed CST (all) consistency among DSLR cameras capturing the same scene under illumination 5200°K.
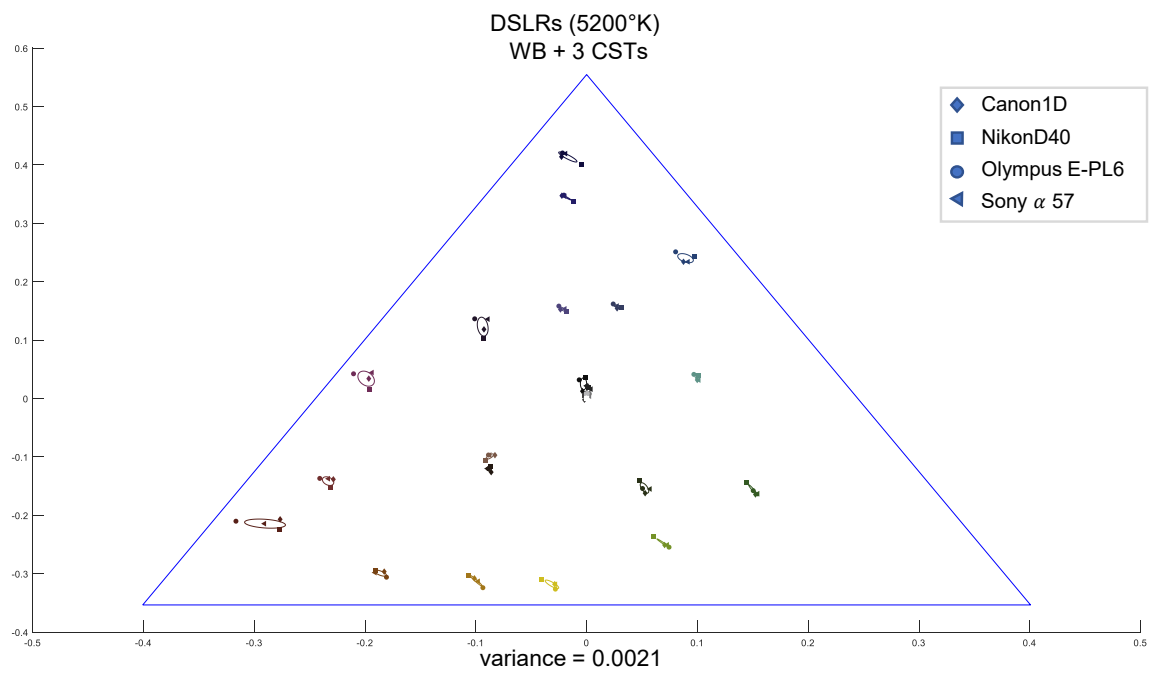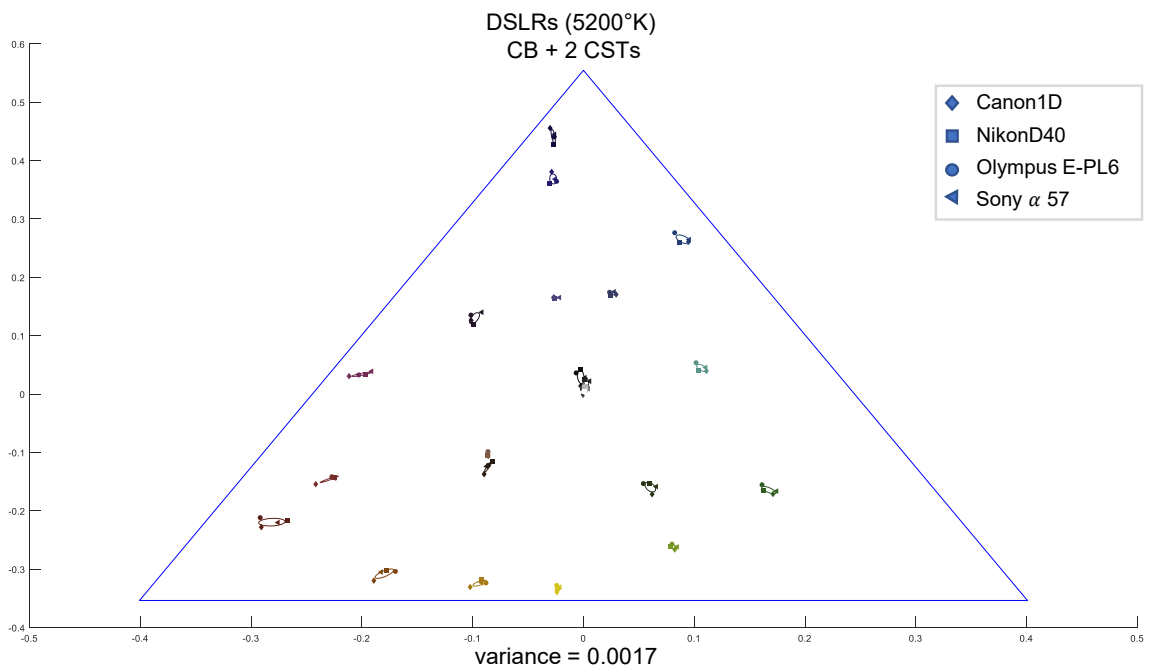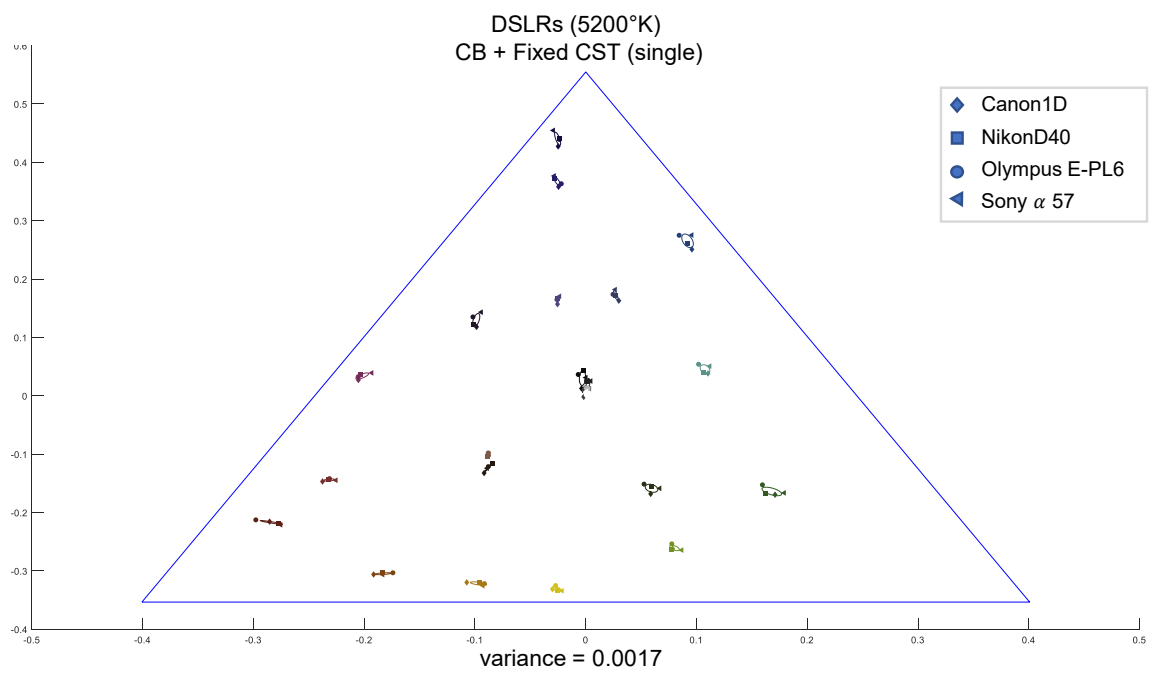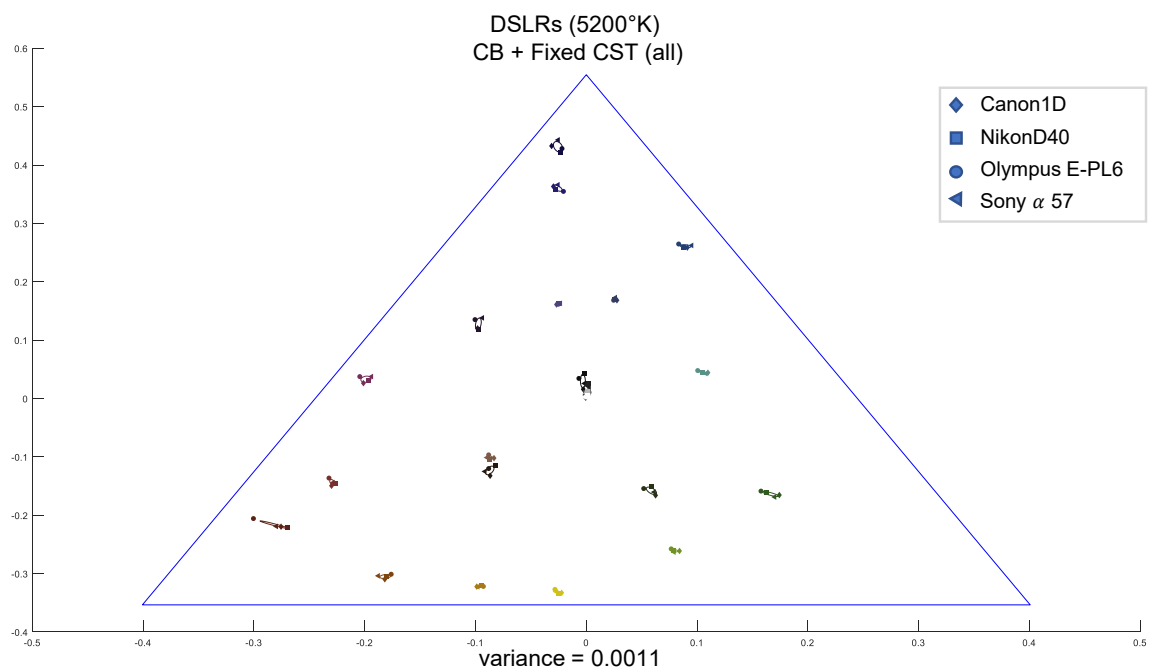
A.4 Additional Visual Results

As we stated in the dissertation, we compute the T_{fixed} for method #2 in two ways. The first is to use only a single observation of the colour chart image with CCT of 6500°K. The second approach is to consider all observations of the colour rendition chart for each different illumination. In our experiments, we distinguish the results obtained with these two different approaches to estimate T_{fixed}. In this appendix section we also present visual results for "Method 2 (extension)" shown as CB + 2 CSTs.

When using all observations, we use a standard ten-fold cross validation procedure that uses 90 images for estimating the T_{fixed} with the iteratively re-weighted least squares (IRLS) (Huber & Ronchetti, 2009) algorithm and the remaining 10 images for computing the error. The final error is the average error across the ten-fold cross validation.

Figures A.49, A.50, A.51, A.52, A.53, A.54, and A.55 show visual results on whole images for the Apple iPhone7, the Google Pixel, the LG-G4, the Canon1D, the NikonD40, the Sony α57, and the Olympus E-PL6 respectively. These results are accompanied by heat maps to reveal which parts of the images are being most affected by errors. For all figures, the corresponding sRGB (as rendered by the camera) is also provided.

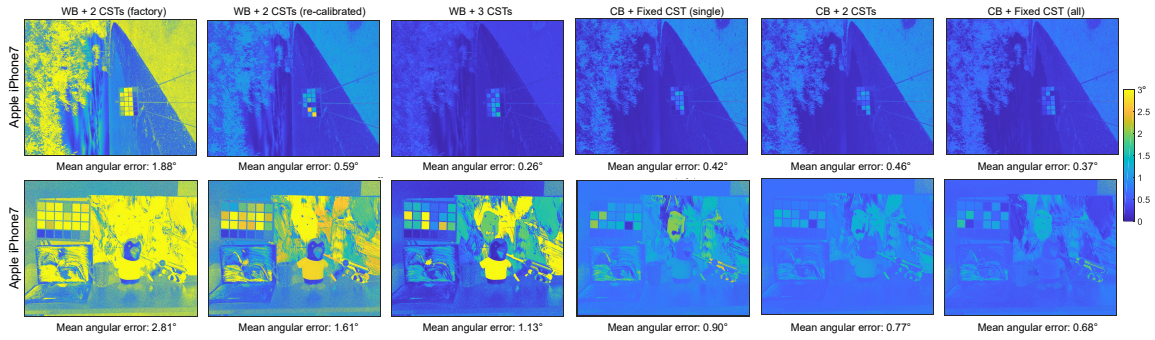**Figure A.49:** Visual comparison for the Apple iPhone7.
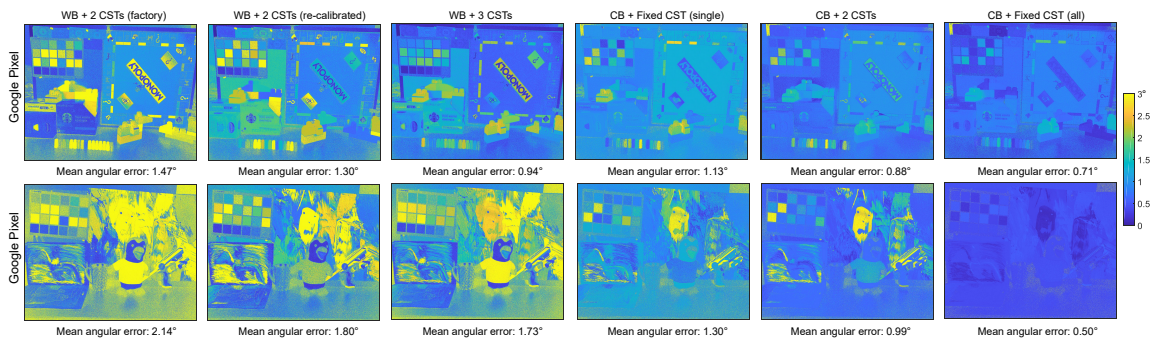

Figure A.50: Visual comparison for the Google Pixel.
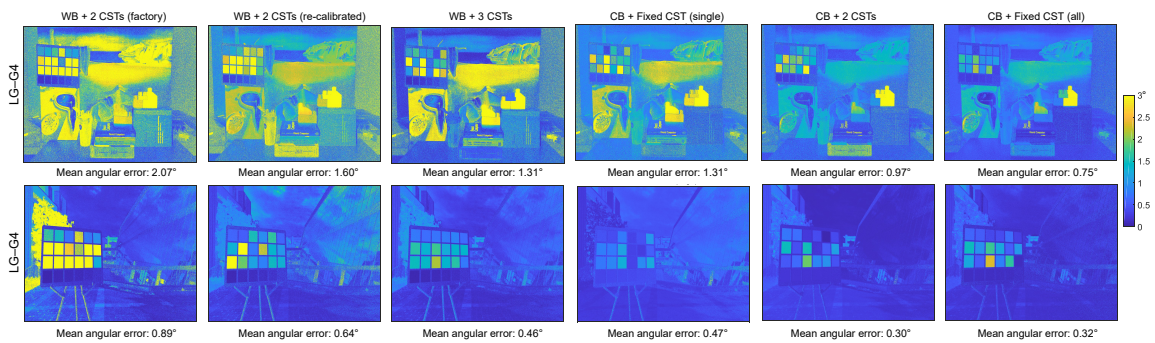

Figure A.51: Visual comparison for the LG-G4.
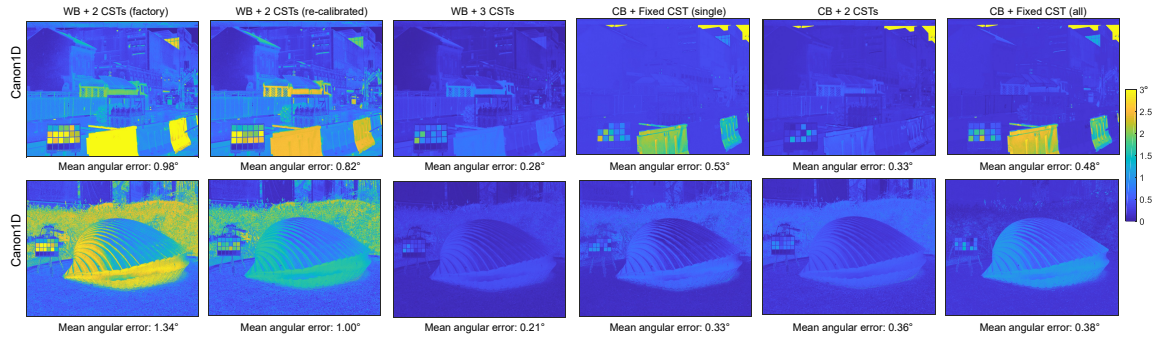
Figure A.52: Visual comparison for the Canon1D.
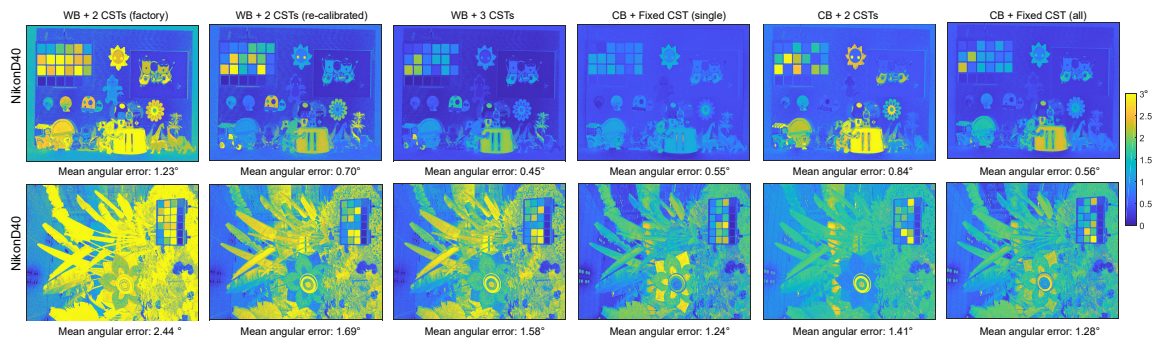
Figure A.53: Visual comparison for the NikonD40.
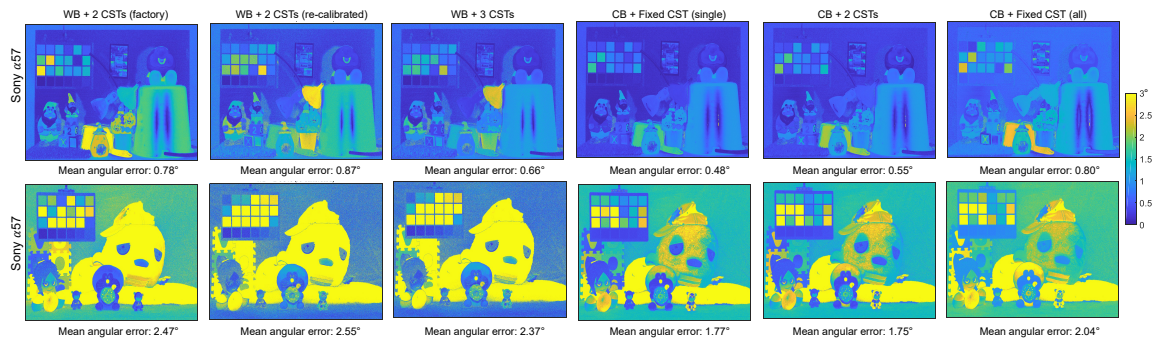
Figure A.54: Visual comparison for the Sony α57.
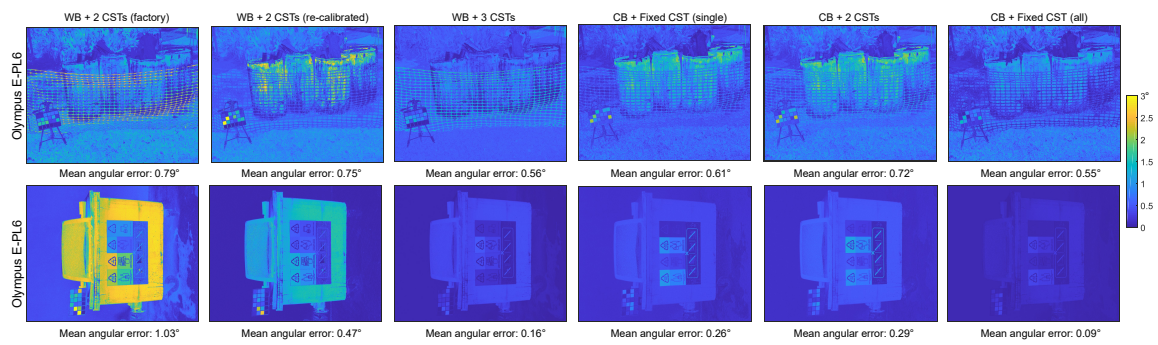
Figure A.55: Visual comparison for the Olympus E-PL6.

References

Adams, A., Talvala, E.-V., Park, S. H., Jacobs, D. E., Ajdin, B., Gelfand, N., . . . Levoy, M. (2010). The Frankencamera: An Experimental Platform for Computational Photography. In *SIGGRAPH*.

Adobe. (2012). DNG Software Development Kit (SDK). https://www.adobe.com/support/downloads/detail.jsp?ftpID=5475. Accessed: 2019-05-05.

Adobe. (2018). *Adobe Camera Raw and DNG Converter for Windows.* Accessed: 2019-05-05.

Akkaynak, D. & Treibitz, T. (2019). Sea-Thru: A Method for Removing Water From Underwater Images. In *CVPR*.

Aubry, M., Paris, S., Hasinoff, S. W., Kautz, J., & Durand, F. (2014). Fast Local Laplacian Filters: Theory and Applications. In *SIGGRAPH*.

Barnard, K., Cardei, V., & Funt, B. (2002). A Comparison of Computational Color Constancy Algorithms. I: Methodology and Experiments with Synthesized Data. *IEEE Transactions on Image Processing, 11*(9), 972–984.

Barnard, K., Martin, L., Coath, A., & Funt, B. (2002). A Comparison of Computational Color Constancy Algorithms. II. Experiments with Image Data. *IEEE Transactions on Image Processing, 11*(9), 985–996.

Barron, J. T. (2015). Convolutional Color Constancy. In *ICCV*.

Barron, J. T. & Tsai, Y.-T. (2017). Fast Fourier Color Constancy. In *CVPR*.

Bastani, P. & Funt, B. (2014). Simplifying Irradiance Independent Color Calibration. In *Color and Imaging Conference*.

Bianco, S., Bruna, A., Naccari, F., & Schettini, R. (2012). Color-Space Transformations for Digital Photography Exploiting Information About the Illuminant Estimation Process. *Journal of Optical Society America A*, *29*(3), 374–384.

Bianco, S., Cusano, C., & Schettini, R. (2017). Single and Multiple Illuminant Estimation Using Convolutional Neural Networks. *IEEE Transactions on Image Processing*, *26*(9), 4347–4362.

Buchsbaum, G. (1980). A Spatial Processor Model for Object Color Perception. *Journal of the Franklin Institute*, *310*(1), 1–26.

Chakrabarti, A. (2015). Color Constancy by Learning to Predict Chromaticity from Luminance. In *NIPS*.

Chakrabarti, A., Hirakawa, K., & Zickler, T. (2012). Color Constancy with Spatio-spectral Statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(8), 1509–1519.

Chatterjee, P. & Milanfar, P. (2010). Is Denoising Dead? *IEEE Transactions on Image Processing*, *19*(4), 895–911.

Cheng, D., Prasad, D. K., & Brown, M. S. (2014). Illuminant Estimation for Color Constancy: Why Spatial-Domain Methods Work and the Role of the Color Distribution. *Journal of Optical Society America A*, *31*(5), 1049–1058.

Cheng, D., Price, B., Cohen, S., & Brown, M. S. (2015a). Beyond White: Ground Truth Colors for Color Constancy Correction. In *ICCV*.

Cheng, D., Price, B., Cohen, S., & Brown, M. S. (2015b). Effective Learning-Based Illuminant Estimation Using Simple Features. In *CVPR*.

Cheung, T. L. V. & Westland, S. (2002). Color Camera Characterization Using Artificial Neural Networks. In *Color and Imaging Conference*.

Chong, H. Y., Gortler, S. J., & Zickler, T. (2007). The von Kries Hypothesis and a Basis for Color Constancy. In *ICCV*.

Connah, D., Drew, M. S., & Finlayson, G. D. (2014). Spectral Edge Image Fusion: Theory and Applications. In *ECCV*.

Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2007). Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering. *IEEE Transactions on Image Processing, 16*(8), 2080–2095.

de Zeeuw, P., Piella, G., & Heijmans, H. (2004). *A Matlab Toolbox for Image Fusion (MATIFUS)* (tech. rep. No. CWI-PNA-E0424). Probability, Networks and Algorithms [PNA], CWI.

Fairchild, M. D. (2013). *Color Appearance Models (3rd edition)*. Chichester, UK: Wiley-IS&T.

Farrell, J. E. & Wandell, B. A. (1995). Method and Apparatus for Identifying the Color of an Image. U.S. Patent No. 5,479,524, Washington, DC: U.S. Patent and Trademark Office, 1995.

Farrell, J. E., Xiao, F., Catrysse, P. B., & Wandell, B. A. (2004). A Simulation Tool for Evaluating Digital Camera Image Quality. In *Electronic Imaging*.

Finlayson, G. D. (1996). Color in Perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 18*(10), 1034–1038.

Finlayson, G. D., Darrodi, M. M., & Mackiewicz, M. (2015). The Alternating Least Squares Technique for Nonuniform Intensity Color Correction. *Color Research & Application, 40*(3), 232–242.

Finlayson, G. D., Drew, M. S., & Funt, B. V. (1993a). Color Constancy: Enhancing von Kries Adaption via Sensor Transformations. In *Human Vision, Visual Processing and Digital Display IV*.

Finlayson, G. D., Drew, M. S., & Funt, B. V. (1993b). Diagonal Transforms Suffice for Color Constancy. In *ICCV*.

Finlayson, G. D., Gong, H., & Fisher, R. B. (2016). Color Homography Color Correction. In *Color and Imaging Conference*.

Finlayson, G. D. & Hayes, A. E. (2015). POP Image Fusion - Derivative Domain Image Fusion Without Reintegration. In *ICCV*.

Finlayson, G. D., Mackiewicz, M., & Hurlbert, A. (2015). Color Correction Using Root-Polynomial Regression. *IEEE Transactions on Image Processing, 24*(5), 1460–1470.

Finlayson, G. D. & Trezzi, E. (2004). Shades of Gray and Colour Constancy. In *Color and Imaging Conference*.

Finlayson, G. D., Zakizadeh, R., & Gijsenij, A. (2017). The Reproduction Angular Error for Evaluating the Performance of Illuminant Estimation Algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(7), 1482–1488.

Finlayson, G. D., Zhu, Y., & Gong, H. (2018). Using a Simple Colour Pre-Filter to Make Cameras More Colorimetric. In *Color and Imaging Conference*.

Forsyth, D. A. (1990). A Novel Algorithm for Color Constancy. *International Journal of Computer Vision, 5*(1), 5–35.

Fredembach, C., Barbuscia, N., & Süsstrunk, S. (2009). Combining Visible and Near-Infrared Images for Realistic Skin Smoothing. In *Color and Imaging Conference.*

Gijsenij, A., Gevers, T., & van de Weijer, J. (2010). Generalized Gamut Mapping Using Image Derivative Structures for Color Constancy. *International Journal of Computer Vision, 86*(2), 127–139.

Gijsenij, A., Gevers, T., & van de Weijer, J. (2011). Computational Color Constancy: Survey and Experiments. *IEEE Transactions on Image Processing, 20*(9), 2475–2489.

Gonzalez, R. C. & Woods, R. E. (2008). *Digital Image Processing (3rd edition).* Upper Saddle River, NJ: Pearson Prentice Hall.

Google. (2015). Camera2 API Package Summary. http://developer.android.com/reference/android/hardware/camera2/package-summary.html. Accessed: 2019-05-05.

Grassmann, H. G. (1853). Zur Theorie der Farbenmischung. *Annalen der Physik und Chemie, 165*(5), 69–84.

Guild, J. (1931). The Colorimetric Properties of the Spectrum. *Philosophical Transactions of the Royal Society of London. A, 230*, 149–187.

Gunturk, B. K., Altunbasak, Y., & Mersereau, R. M. (2002). Colorplane Interpolation Using Alternating Projections. *IEEE Transactions on Image Processing, 11*(9), 997–1013.

Haupt, E. J. (2000). *Encyclopedia of Psychology.* New York, NY: Oxford University Press.

Heide, F., Steinberger, M., Tsai, Y.-T., Rouf, M., Pajak, D., Reddy, D., . . . Pulli, K. (2014). FlexISP: A Flexible Camera Image Processing Framework. In *SIGGRAPH Asia*.

Hering, E. (1964). *Outlines of a Theory of the Light Sense*. Cambridge, MA: Harvard University Press.

Hong, G., Luo, M. R., & Rhodes, P. A. (2001). A Study of Digital Camera Colorimetric Characterisation Based on Polynomial Modelling. *Color Research & Application, 26*(1), 76–84.

Hu, Y., Wang, B., & Lin, S. (2017). FC[4]: Fully Convolutional Color Constancy with Confidence-Weighted Pooling. In *CVPR*.

Huber, P. J. & Ronchetti, E. M. (2009). *Robust Statistics (2nd edition)*. Hoboken, N.J.: Wiley.

Hung, P.-C. (1993). Colorimetric Calibration in Electronic Imaging Devices Using a Look-Up-Table Model and Interpolations. *Journal of Electronic Imaging, 2*(1), 53–61.

Hurvich, L. M. & Jameson, D. (1957). An Opponent-Process Theory of Color Vision. *Psychological Review, 64*(6), 384–404.

Jaroensri, R., Biscarrat, C., Aittala, M., & Durand, F. (2019). Generating Training Data for Denoising Real RGB Images via Camera Pipeline Simulation. arXiv: 1904.08825.

Jiang, H., Tian, Q., Farrell, J., & Wandell, B. A. (2017). Learning the Image Processing Pipeline. *IEEE Transactions on Image Processing, 26*(10), 5032–5042.

Karaimer, H. C. & Brown, M. S. (2016). A Software Platform for Manipulating the Camera Imaging Pipeline. In *ECCV*.

Karaimer, H. C. & Brown, M. S. (2018). Improving Color Reproduction Accuracy on Cameras. In *CVPR*.

Karaimer, H. C. & Brown, M. S. (2019). Beyond raw-RGB and sRGB: Advocating Access to a Colorimetric Image State. In *Color and Imaging Conference*.

Karaimer, H. C., Khodadad, I., Kazemzadeh, F., & Brown, M. S. (2019). A Customized Camera Imaging Pipeline for Dermatological Imaging. In *CVPRW*.

Kim, S. J., Lin, H. T., Lu, Z., Süsstrunk, S., Lin, S., & Brown, M. S. (2012). A New In-Camera Imaging Model for Color Computer Vision and Its Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(12), 2289–2302.

Krishnan, D. & Fergus, R. (2009). Dark Flash Photography. In *SIGGRAPH*.

Land, E. H. (1977). The Retinex Theory of Color Vision. *Scientific American*, *237*(6), 108–128.

Land, E. H. (1986). Recent Advances in Retinex Theory. *Vision Research*, *26*(1), 7–21.

Land, E. H. & McCann, J. J. (1971). Lightness and Retinex Theory. *Journal of Optical Society America*, *61*(1), 1–11.

MacAdam, D. L. (1970). *Sources of Colour Science*. Cambridge, MA: MIT Press.

Misiti, M., Misiti, Y., Oppenheim, G., & Poggi, J.-M. (2007). *Wavelets and Their Applications*. Newport Beach, CA: Wiley-ISTE.

Nam, S. & Kim, S. J. (2017). Modelling the Scene Dependent Imaging in Cameras with a Deep Neural Network. In *ICCV*.

Newton, I. (1672). New Theory of Light and Colours. *Philosophical Transactions*, *6*(80), 3075–3087.

Oh, S. W. & Kim, S. J. (2017). Approaching the Computational Color Constancy as a Classification Problem Through Deep Learning. *Pattern Recognition, 61,* 405–416.

Paris, S. & Durand, F. (2006). A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach. In *ECCV.*

Paris, S., Hasinoff, S. W., & Kautz, J. (2011). Local Laplacian Filters: Edge-Aware Image Processing with a Laplacian Pyramid. In *SIGGRAPH.*

Ramanath, R., Snyder, W. E., Yoo, Y., & Drew, M. S. (2005). Color Image Processing Pipeline. *IEEE Signal Processing Magazine, 22*(1), 34–43.

Robertson, A. R. (1968). Computation of Correlated Color Temperature and Distribution Temperature. *Journal of Optical Society America, 58*(11), 1528–1535.

Rowe, A., Goode, A., Goel, D., & Nourbakhsh, I. (2007). *CMUcam3: An Open Programmable Embedded Vision Sensor* (tech. rep. No. CMU-RI-TR-07-13). Robotics Institute, Carnegie Mellon University.

Serrano, A., Heide, F., Gutierrez, D., Wetzstein, G., & Masia, B. (2016). Convolutional Sparse Coding for High Dynamic Range Imaging. *Computer Graphics Forum, 35*(2), 153–163.

Sharma, G. (2003). *Digital Color Imaging Handbook.* Boca Raton, FL: CRC Press.

Shi, W., Loy, C. C., & Tang, X. (2016). Deep Specialized Network for Illuminant Estimation. In *ECCV.*

Spaulding, K. E., Giorgianni, E., & Woolfe, G. (2000). Reference Input/Output Medium Metric RGB Color Encodings (RIMM/ROMM RGB). In *Image Processing, Image Quality, Image Capture, Systems Conference.*

Süsstrunk, S., Fredembach, C., & Tamburrino, D. (2010). Automatic Skin Enhancement with Visible and Near-Infrared Image Fusion. In *ACM MM*.

Tai, Y.-W., Chen, X., Kim, S., Kim, S. J., Li, F., Yang, J., . . . Brown, M. S. (2013). Nonlinear Camera Response Functions and Image Deblurring: Theoretical Analysis and Practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(10), 2498–2512.

van de Weijer, J., Gevers, T., & Gijsenij, A. (2007). Edge-Based Color Constancy. *IEEE Transactions on Image Processing, 16*(9), 2207–2214.

von Helmholtz, H. (1925). *Treatise on Physiological Optics (3rd edition)*. New York, NY: Dover Publications.

von Kries, J. (1878). Beitrag zur Physiologie der Gesichtsempfindung. *Archiv für Physiologie, 2*, 503–524.

Wright, W. D. (1928–1929). A Re-Determination of the Trichromatic Coefficients of the Spectral Colours. *Transactions of the Optical Society, 30*(4), 141–161.

Wu, C., Samadani, R., & Gunawardane, P. (2011). Same Frame Rate IR to Enhance Visible Video Conference Lighting. In *ICIP*.

Wyszecki, G. (1973). Current Developments in Colorimetry. *AIC Color, 73*, 21–51.

Wyszecki, G. & Stiles, W. S. (1982). *Color Science: Concepts and Methods, Quantitative Data and Formulae*. New York, NY: Wiley.

X-Rite. (1976). X-Rite ColorChecker Chart. http://xritephoto.com/colorchecker-classic. Accessed: 2019-05-05.

X-Rite. (2010). X-Rite ColorChecker Camera Calibration Software. http://xritephoto.com/ph_product_overview.aspx?ID=1257&Action=Support&SoftwareID=986&catid=28. Accessed: 2019-05-05.

Young, T. (1802). On the Theory of Light and Colours. *Philosophical Transactions of the Royal Society of London*, *92*, 12–48.

Zhang, X., Sim, T., & Miao, X. (2008). Enhancing Photographs with Near-Infrared Images. In *CVPR*.