TAMPERE UNIVERSITY OF TECHNOLOGY

KASPERI SUMMANEN

APPLICATION FOR SUBJECTIVE AUTO WHITE BALANCE ACCURACY MEASUREMENT

Master's Thesis

Examiners:
University Lecturer Heikki Huttunen
M.Sc. (Tech.) Petri Nenonen
Examiner and topic approved by
the Faculty Council of Computing and
Electrical Engineering on 06.03.2013

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Master's Degree Programme in Signal Processing and Communications Engineering
SUMMANEN, KASPERI: Application for Subjective Auto White Balance Accuracy Measurement
Master of Science Thesis, 52 pages
May 2013
Major: Signal Processing
Examiners: University Lecturer Heikki Huttunen, M.Sc. (Tech.) Petri Nenonen
Keywords: Auto White Balance, Chromatic Adaption, Mobile application, Data analysis, Windows Phone 8, Mobile camera, Digital camera

Variation in illumination conditions will create situations where hue of the dominant illuminant is different. It can be said that the illumination sources have different color temperatures. The human visual system has developed to adapt to these changes of the illumination hue, and thus it is able to maintain the color constancy of the objects. The task of the Automatic White Balance (AWB) is to do the same in digital cameras. The purpose of the AWB is to get the output colors to look as much same as the human eye saw the scene when the image was taken, i.e., as natural.

The main objective of this thesis is to implement a mobile application which can collect data from the accuracy of the AWB algorithm and record the level of the color adaption of eye in different illumination conditions. With this application test users can capture images and adjust the colors immediately after the capturing. In this case, the adjustments are done in the same illumination condition as the capturing. This is important because the human visual system will adapt to see white as white in different illuminations. The final image from the camera is transformed manually to look correct in the sRGB color space, that corresponds to the color temperature of 6500K. When capturing images under other illumination sources with different color temperature we can measure the difference with the implemented mobile application even when the AWB algorithm has worked correctly.

The human visual system is not perfect and the color adaption will not be complete in low and high color temperatures. Therefore it is expected to get difference between the AWB result and the test user preferred output even though the algorithm has performed correctly. In addition, these color adaption results can be used to build behavior model of the color adaption when the adaption is incomplete. Further on, these color appearance models can be used to build better AWB algorithm that imitates the human eye better than before and is not always trying to make white look as white.

Based on the results of the study this concept is working as thought and designed. Anyhow, it contains few error sources that need to be taken into account when analyzing the collected data. The results from the data analysis can not be used to build an automatic classifier, and thus, human involved evaluation is often needed. This will be a major trend in future studies.

TIIVISTELMÄ

TAMPERE UNIVERSITY OF TECHNOLOGY
Signaalinkäsittelyn ja tietoliikennetekniikan koulutusohjelma
SUMMANEN, KASPERI: Sovellus subjektiiviseen automaattisen valkotasapainon tarkkuuden arviointiin
Diplomityö, 52 sivua
Toukokuu 2013
Pääaine: Signaalinkäsittely
Tarkastajat: Yliopistonlehtori Heikki Huttunen, DI Petri Nenonen
Avainsanat: Automaattinen valkotasapaino, väriadaptaatio, mobiilisovellus, datan analysointi, Windows Phone 8, mobiilikamera, digitaalikamera

Erilaiset valaistusolosuhteet muodostavat tilanteita, joissa vallitsevan valon värisävy vaihtelee. Voidaan sanoa, että valonlähteillä on eri värilämpötiloja. Ihmisen visuaalinen järjestelmä on kehittynyt mukautumaan näiden valonlähteiden sävymuutoksiin ja kykenee havaitsemaan esineiden värit samoina olosuhteista riippumatta. Automaattisen valkotasapainon tehtävä on tehdä tämä sama digitaalikameroissa. Sen tarkoitus on saada värit näyttämään lopullisessa kuvassa mahdollisimman samalta kuin ihmissilmä ne näki kuvanottohetkellä, toisin sanoen luonnollisilta.

Tässä diplomityössä toteutetaan mobiilisovellus, jolla voidaan kerätä dataa automaattisen valkotasapainoalgoritmin tarkkuudesta, sekä silmän adaptaation tasosta eri valaistusolosuhteissa. Mobiilisovelluksen avulla testihenkilöt voivat ottaa kuvia ja tehdä kuvalle värisäätöjä heti kuvan oton jälkeen. Tällöin säädöt tehdään samassa valaistusolosuhteessa, kun itse kuva otettiin. Tämä on tärkeää, koska ihmisen näköjärjestelmä adaptoituu näkemään valkoisen valkoisena eri värilämpötiloissa. Lopullinen kuva on säädetty näyttämään oikealta sRGB-väriavaruudessa, joka vastaa 6500K värilämpötilaa. Tällöin kuvattaessa muissa värilämpötiloissa voidaan toteutetulla mobiilisovelluksella mitata eroa lopputuloksessa algoritmin oikein toimiessakin.

Ihmisen näköjärjestelmä ei ole täydellinen vaan matalissa sekä korkeissa värilämpötiloissa väriadaptaatio ei ole enää täydellinen. On siis odotettavaa saada poikkeavia valkotasapainotuloksia näissä tapauksissa, vaikka algoritmi sinällään toimisikin oikein. Lisäksi ihmissilmän adaptaation mittaustuloksista voidaan rakentaa malli, kuinka adaptaatio käyttäytyy, kun se ei ole täydellisesti adaptoitunut, toisin sanoen matalissa ja korkeissa värilämpötiloissa. Nämä värivaikutelmamallit (engl. Color appearance models) mahdollistavat valkotasapainoalgoritmin, joka mukailee silmän toimintaa paremmin, eikä aina yritä tehdä valkoisesta valkoista.

Työhön sisältyi kirjallisuustutkimus, joka sisälsi perusteita näkemisestä ja erityisesti värinäöstä, sekä värin ymmärtämistä ja käsittelyä tieteen keinoin. Lisäksi luodaan katsaus digitaalisen kuvantamisen perusteisiin optiikasta lopulliseen kuvaan.

Työn tuloksien perusteella konsepti on toimiva, mutta sisältää joitakin virhetekijöitä, jotka pitää huomioida dataa analysoitaessa. Data-analyysista saatavat tulokset eivät itsessään sovellu automaattisen oikein-väärin luokittimen rakentamiseen. Tuloksia tarkasteltaessa usein tarvitaan ihmisarviointia vallitsevista olosuhteista, jotta voidaan tehdä luotettavia arvioita. Tämän ongelman ratkaiseminen on tärkeimpiä kehityskohtia jatkotutkimuksissa.

PREFACE

This thesis was carried out while working in Nokia Smart Devices, Imaging organization in Tampere, Finland.

I would like to thank my colleagues in the Imaging team and especially Algorithm and Middleware group for answering the questions and giving a helping hand when needed. Special thanks to M.Sc (Tech.) Petri Nenonen who was my supervisor on behalf of Nokia and University Lecturer Heikki Huttunen from Tampere University of Technology who was the examiner of this work.

And finally, I want to thank my girlfriend whose patience and support carried me through this work, my friends who gave me something else to think when it was needed, and last but not least my family who always remembered to ask "Is it ready yet?".

Tampere April 15th, 2013

Kasperi Summanen

CONTENTS

TERMS AND DEFINITIONS

f_{35mm} Focal length equivalent to 35mm full frame sensor or film.

A/D Analog to Digital Conversion.

AEC Automatic Exposure Control.

API Application Programming Interface.

AWB Automatic White Balance.

B/G The sensitivity or intensity of blue in relation to green color component.

BSI Backside illuminated.

CCD Charge-Coupled Device.

CCT Correlated Color Temperature.

CIE The Commission Internationale de l'Eclairage (International Commission on Illumination).

CIPA Camera & Imaging Products Association.

CMOS Complementary Metal-Oxide-Semiconductor.

DSLR Digital Single-Lens Reflex.

Exif Exchangeable image file format.

FSI Frontside illuminated.

HVS Human Visual System.

IPP Image processing pipeline.

IR Infrared.

ISP Image signal processor.

JND Just noticeably difference.

LAB CIELAB or L*a*b*, an advanced color space.

LGN Lateral Geniculate Nucleus.

LUV CIELUV or L*u*v*, an advanced color space.

NFC Near Field Communication.

OS Operating System.

R/G The sensitivity or intensity of red in relation to green color component.

RAW Unprocessed digital image format.

RGB Red, Green, and Blue color components.

SDK Software Development Kit.

sRGB Standard RGB, commonly used color space in digital imaging.

UI User Interface.

WP8 Windows Phone 8.

XAML Extensible Application Markup Language.

XYZ CIE XYZ or CIE 1931, a color space standard.

1. INTRODUCTION

Almost every smart phone and even quite many low-end mobile phones have an embedded digital camera nowadays. The quality of the captured image is something that every manufacturer tries to improve and be better than the rest. The overall image quality is a sum of multiple phases in the sequence that starts from light rays entering to a camera sensor through the optics of the camera and ends when an output image is shown on the screen of the device.

One of these phases is called white balance correction or color balancing. Automatic White Balance (AWB) correction in digital cameras attempts to adjust the captured color information to match the same as a human eye sees the captured scene. The result is always an estimation based on some presumed facts and measurements of the camera automatism. Today's AWB algorithms manage to achieve quite good results in most cases but still they fail to achieve color constancy in every situation. Human eyes have excellent ability for color constancy and, for example, they see red berries as red in direct sunlight, shadowy forest, or an office lighting. In certain situations like low and high color temperatures, for example, sunrise, and artificial illustrations, even human eyes can be misguided. However, human eyes are a lot reliable than the best algorithms and thus those are trying to achieve something more what humans already have.

In this thesis implementation of mobile application is targeted. The purpose of this application is to help expert users to provide data for analyzing differences between a human eye adaption and automatic white balance algorithms. This application allows the expert users to take images under different illumination conditions and adjust the color balance of the captured imaged to match what their eyes see in the same light condition. After adjusting the image the users will adjust the gray patch to match as much gray as they can. This adjustment is relying on that the users have a mental picture of the gray color. They gray patch adjustment allows us to estimate the eye adaption. This data is gathered along with the parameters of the automatic white balance algorithms run in the smart phone. By expert users we mean people inside the company who know our product and have experience related to imaging development.

One of the key aspect is to discuss and evaluate how well this kind of mobile application suits to collect data and is the data usable in statistical viewpoint. The study provides statistical data to improve our (Nokia) AWB algorithms and example situations where automatic approaches have difficulties to provide good results. This study does not fix any known problem but give us a tool and a way to improve our results.

This thesis is structured in two theory chapters, two implementing and analyzing chapters, and conclusion. The second chapter of this thesis provides introduction to the physics of light and the human visual system together with color science. In this case, the introduction means important and relevant theories behind the idea of the mobile application. The structure of digital camera and its processing pipeline is presented in the third chapter. It provides an overall picture of how light rays entering the camera lens are transformed in to a digital image. The actual work in this thesis is divided into two parts. In the first part, requirements and technologies needed for Windows Phone 8 mobile camera application creation are presented in Chapter 4. The purpose of this mobile application is to gather test data related to AWB behavior versus how human eye sees the situation and how the eyes were adapted. Fifth chapter contains the second work part, a solution for presenting and analyze the gathered data is implemented using MathWorks MATLAB[1]. The solution involves a script collection that visualize regions of interest of the data. The results of this study is also presented in this chapter. Chapter 5 concludes the thesis by evaluating implemented application and how the results corresponded in our expectations.

[1] http://www.mathworks.com/

2. INTRODUCTION TO COLOR SCIENCE AND HUMAN VISUAL SYSTEM

In this chapter the human visual system is explained from how human eye work to how our brains create the image "we see". In addition, color vision and science based on it is discussed in those parts that are related to this study. But first short background of the light is given. This chapter is meant to provide understanding behind the AWB algorithms and why we need those with digital cameras.

2.1 Basics of Light

Prerequisite for seeing things around us is light. The object seen must either emit light or reflect some existing source like the sun light outside. In physics light have dual nature and it can be described as a waveform or photon particles but neither theories overrun other.

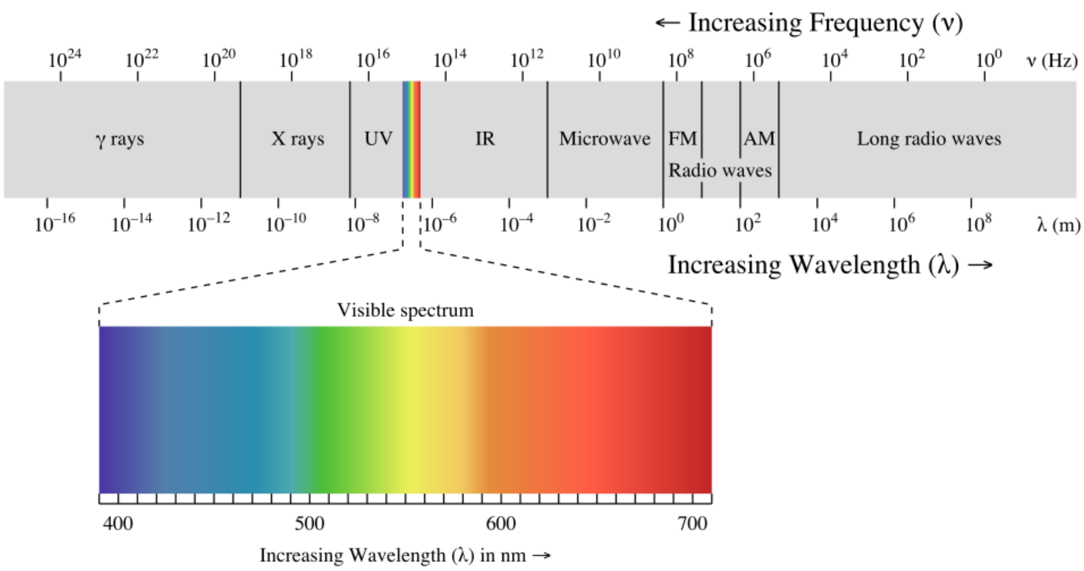

Figure 2.1: Electromagnetic spectrum and visible region.[1]

In the scope of this thesis the theory we need to know about light rely on waveform of the light. Also when studying optics light is often described as rays that travel in straight lines. Visible light is part of electromagnetic radiation and it differ from other electromagnetic radiation sources by wavelength and frequency which relate each other according to the relation:

$$\lambda \cdot f = c, \tag{2.1}$$

where λ is the wavelength of the radiation, f is the frequency, and c denotes the speed of light.

Then, the visible light can be defined by its intensity I and wavelength λ where intensity describes the power. The visible spectrum is a region between $\lambda_{min} = 360nm$ to $\lambda_{max} = 830nm$ in our normal environment, air, this same region is also valid for vacuum. Figure 2.1 shows how narrow region the visible part is from the whole electromagnetic spectrum. [1]

The light seen in our daily environments is composition of multiple wavelengths but single wavelength, monochromatic, light is used, for example, in laser. When light source radiates non monochromatic light, i.e., multiple wavelengths the radiated intensity differs on each wavelength. This radiation energy distribution can be measured at each wavelength and thus it will form the spectrum of the light source. For example, Figure 2.2 presents two spectra of incandescent and fluorescent light sources.

When light from illumination source reach the object, that does not emit light itself, it will absorb part of the radiation and reflect the other part. We can denote spectral reflectance of the object with $R(\lambda)$ and the illumination source with $I(\lambda)$. Now, the color stimulus of the object is [2]

$$S(\lambda) = I(\lambda)R(\lambda). \tag{2.2}$$

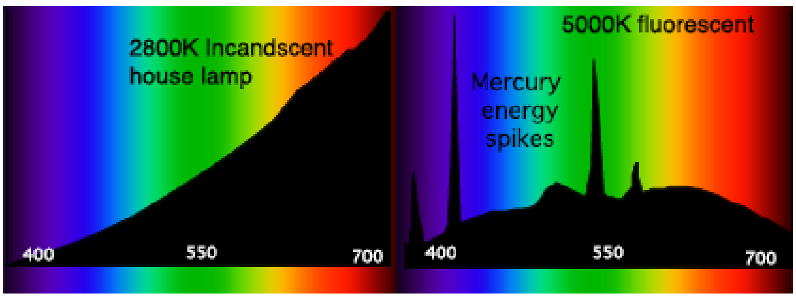

Figure 2.2: The radiation energy distribution of two artificial light sources.[2]

---

[1]Retrieved from: http://commons.wikimedia.org/wiki/File:EM_spectrum.svg
[2]Retrieved from:http://commons.wikimedia.org/wiki/File:Spectral_Power_Distributions.png

2.2 Eye and Image Formation Inside the Brain

This section describes the human vision in general level and provide the contrast to the camera structure presented in Chapter 3. It will help the reader to perceive the difficulties what camera implementations have when comparing the vision.

The actual vision and the seeing process is something that normal people rarely think and it is taken for granted. Most of the people use their eyes as a main sense. Vision is not irreplaceable sense but it is most likely the most advanced one. For example, the optic nerve which transmit the information from the retina to the visual cortex inside the brain contains around 1 million neural fibers [3, p.290]. In contrast, the auditory nerve contains 30,000 fibers and still the ability to hear is quite amazing. We often think consciously what we see but the part how we see is done subconsciously.

Why our eyes then have evolved to sense not just shades of gray but also different colors? Maybe for recognizing a healthy berry from a poisonous one. On the other hand why dogs have brilliant sense of smell but humans only a mediocre one. These are questions that biologists and scientists of the evolutionary theory are trying to figure out. Meanwhile other scientists learn more details how our eyes actually work.

Human visual system is extremely complex system which is still partly unclear for scientists. It is a combination of optics, detectors, neural processing, and cognition [4, p.13]. In this section we does not dive into details of the human visual system, but try to explain basic principles of it that are relevant for our study.

The Eye

Figure 2.3 presents a cross section of the human eye. Light entering the eye is projected to the retina which is located onto back of the eye. The retina contains our light receptors, rods and cones, which absorb the entered light from the scene and generate a neural signal for the brain. The quality of the image projected to the retina is a sum of the components in the eye. The cornea, lens and the fluids, aqueous and vitreous humor, effects how well image is optically focused. The combined quality of the work flow inside the eye are part of the spectral and spatial properties of the light receptors. [4, p.13]

As stated above, the retina contains two different kind of receptors, called rods and cones. The rods are sensitive for light intensity and, for example, provide us ability to see in quite dark places. The cones are sensitive for colors and thus in this thesis we are mainly interested on those. Cones are divided to be sensitive in different wavelengths of light, i.e., short (S), middle (M), and long (L). These

[3] Retrieved from:
http://commons.wikimedia.org/wiki/File:Schematic_diagram_of_the_human_eye_en.svg

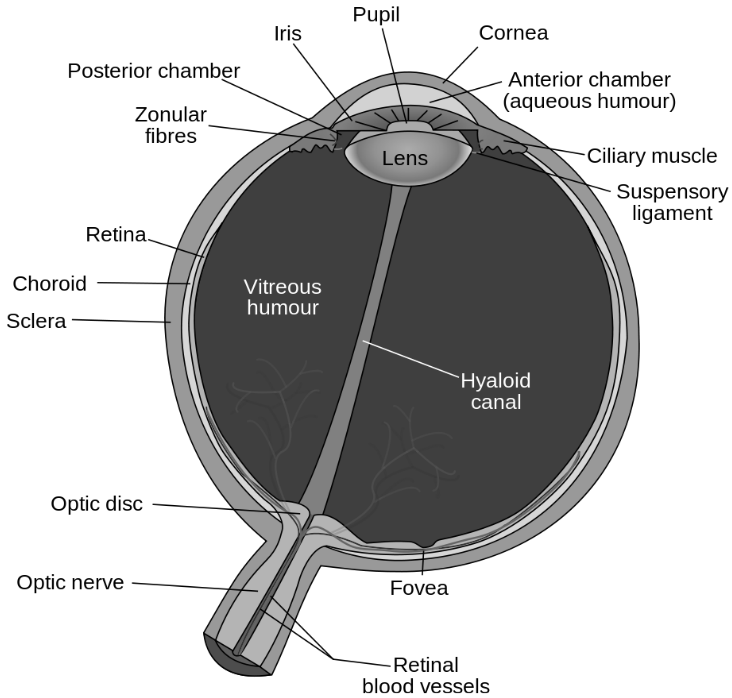

Figure 2.3: Schematic representation of the human eye.[3]

correspond to respectively blue, green, and red which lead us to scientific main colors, RGB color space. [5; 1]

Without multiple different type of sensors inside the eye we are not able to see colors. After all the color perception is combined work of cone cells and actually the information how their sensed signals differ from each other. By the present knowledge, the responses of the cone cells are compared and combined in early state before transmitting the information to the brain. This process is done in Ganglion cells [3, pp.307-308] which are located inside the retina. [3, pp.310-312]

Stockman et al. [6] have studied how different cone receptors, L, M, and S work and their spectral sensitivities. From their results, see Figure 2.4, that are scaled between 0.0 to 1.0 we see approximately the wavelength ranges which each cone type is sensitive and their peak sensitivities occur at $566nm$, $543nm$, and $440nm$ in order L, M, and S.

The Brain

While principles of the human eye is already quite well known the brain is still remaining as a bit of a mystery. As previously stated the processing starts already in the retina and is continued inside the neural pathways towards the visual cortex. The first stop is at the optic chiasm where right and left eye information meets. The

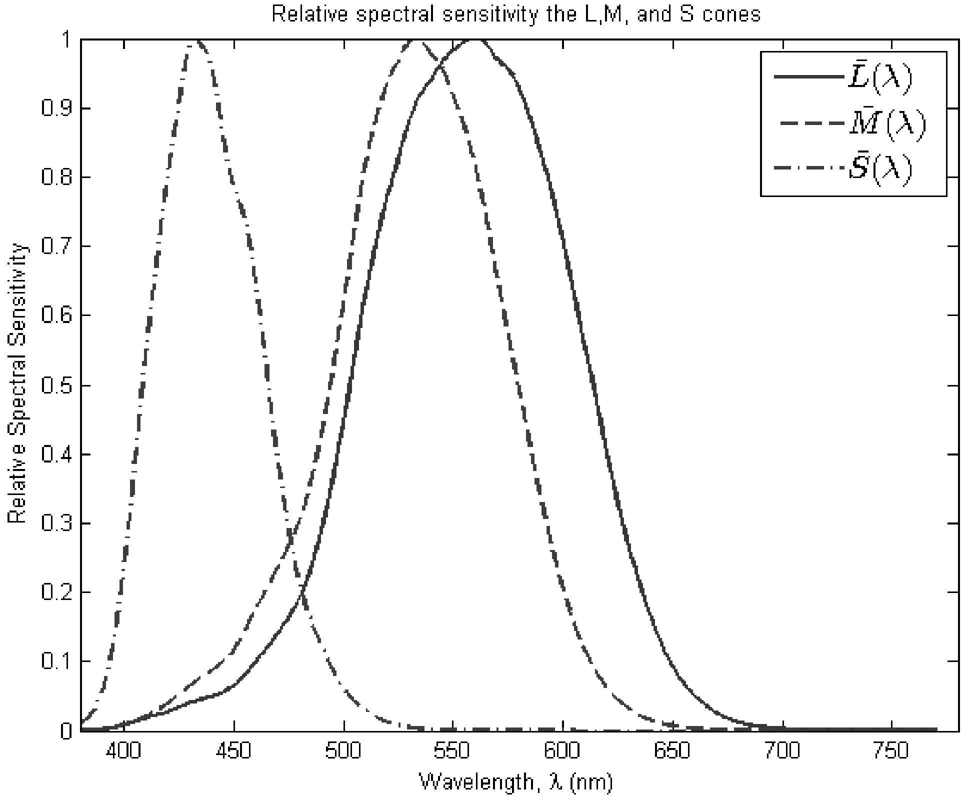

Figure 2.4: Relative trichromacy sensitivity functions of the cone receptors.

journey continues to Lateral Geniculate Nucleus (LGN) that is part of the central nervous system. LGN is responsible of performing anatomical calculations which are, for example, correlations between left and right eye information. After this the information is processed in the primary visual cortex which is most studied visual area inside the brain. It is understood that our pattern recognition unit is located here. Then there is multiple processing layers that are responsible rest of the magic inside the visual system. [7]

During the human evolution the Human Visual System (HVS) has become excellent to adjust on different light conditions. In most cases colors are same under different light conditions, for example, one can look an orange outside in direct sunlight, in shadowy forest, or indoor fluorescent illumination but the orange is still seen as same color. This is called color constancy and it is something that digital cameras are trying to achieve. Scientific meaning of the color constancy is that perceived color of an object remains relatively static under varying illumination sources. For example, if the spectral response of an orange would be measured under different illumination conditions the response would not be the same. But still humans see it as the same this property is a result of chromatic adaption of the human visual system.

2.3 Metamerism

Phenomenon that is closely related to the color constancy and caused by trichromatic stimulus is metamerism. It means that two objects appear to be same color but they actually have different spectral power distributions. As this is a lack in our visual system it is not pure negative feature. For example, it has made easier to simplify color reproduction methods in electronic devices, e.g., TV and printers. More precisely metamerism can be divided as illuminant and observer metamerism. [4, p.14]

Illuminant metamerism is phenomenon when two or more objects seem to be same color but they are not, i.e., they have different spectral reflectance curves. If by changing illumination the objects does not anymore match these objects are illuminant metamerism.

Observer metamerism is happening when two or more observers will see the object under study in different colors. For example, if the illumination stays constant and changing the observer of the objects that first observer saw as same color, is causing that the objects are not anymore interpreted as same color.

2.4 Chromatic Adaptation and Color Constancy

Chromatic adaption can be defined as a difference from adaption to white light. Furthermore white light is light that looks white or achromatic. However, these definitions made by common sense are not that straight forward, what looks white depends of the dominant illumination source, i.e., the viewing situation. For example, if we spend time doing a task in office which is illuminated by fluorescent lamp and go visit the office supply room that happens to be illuminated by tungsten lamp we get concrete example of chromatic adaption. We know that papers in the supply room are white but still they have yellow-orange color cast when we enter the room. After spending a moment inside the supply room we notice that white papers look more and more white. This phenomenon is a chromatic adaption of our vision. One of the finest features of the chromatic adaption is that it is not only adapting globally but also locally. [8]

Probably the most known model of chromatic adaption was proposed by von Kries in 1905 [9]. His idea was that each cone cell has an adaption gain on its own. The response model of the cones is

$$L_a = k_L L, \tag{2.3}$$

$$M_a = k_M M, \tag{2.4}$$

$$S_a = k_S S, \tag{2.5}$$

where L, M, and S are the initial cone responses, L_a, M_a, and S_a are the cone signals after the adaption, and k_L, k_M, and k_S are the gains to match this adaption. The gains are obtained as an inverse of the cone responses for the scene maximum stimulus and are presented in following equations,

$$k_L = 1/L_{max}, \tag{2.6}$$

$$k_M = 1/M_{max}, \tag{2.7}$$

$$k_S = 1/S_{max}. \tag{2.8}$$

Equations (2.3) to (2.8) can be extended to use for asymmetric matching that means matching of color stimuli under different adaption conditions. These matching colors can be calculated with following equations,

$$L_2 = (L_1/L_{max1})L_{max2}, \tag{2.9}$$

$$M_2 = (M_1/M_{max1})M_{max2}, \tag{2.10}$$

$$S_2 = (S_1/S_{max1})S_{max2}, \tag{2.11}$$

where subscript one describe the starting viewing condition cone responses and subscript two the cone responses under the new illuminant. This is known as a von Kries transform and it is widely used in rendering applications since it is quite simple. Anyhow, in later studies [1] it has been noticed that von Kries transformation can not explain completely all the phenomena got from psychophysical experiments. And it is been guessed that part of the chromatic adaption lies in the neural pathways towards the brain and is a nonlinear process.

2.5 Color Appearance

Color appearance is a wide area of study that is trying to explain how we see colors and how we can describe them mathematically. Color appearance can be described with six terms that are brightness, lightness, colorfulness, chroma, saturation, and hue. Colorfulness, saturation, and chroma are related to each other in way that knowing two of them one can describe the third one. One of the famous examples of color appearances is seen in Figure 2.5. In this Figure the inner gray patches are identical shades of gray but the outer gray box cause effect called simultaneous contrast [10, pp.113-114]. This cause that we interpret the gray patch on dark background to be brighter than the one on light background. This same behavior can be seen with colors also.

In the scope of this thesis we are especially interested color appearance and its models that are related to model the inconstancy of the human visual system. For example, we do not want a perfect white balance on the image captured during sunset. We want to capture the moment as average eye see it.

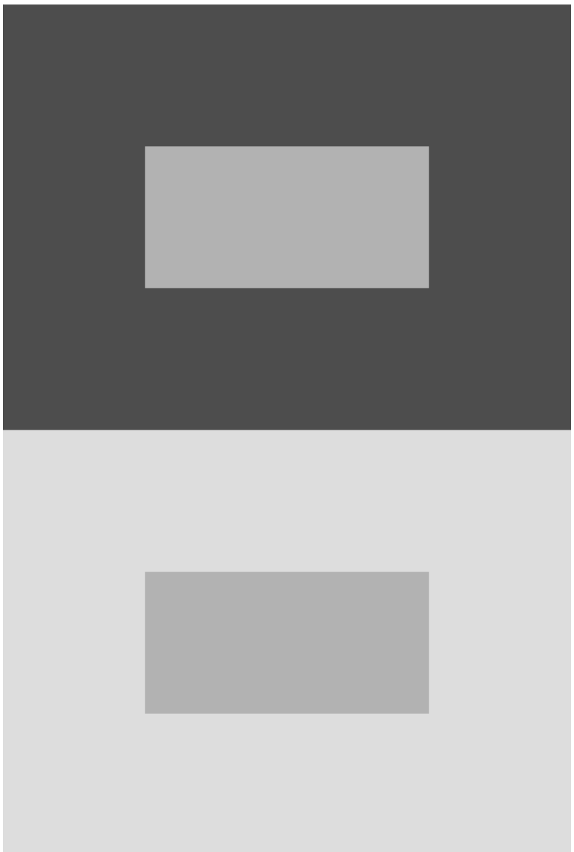

Figure 2.5: Famous color appearance phenomenon example of the simultaneous contrast. The inner gray patches are equal in the mean of tristimulus value. [5]

2.6 Color Spaces

In color science, different kind of color spaces are used to achieve optimal usage for needed tasks. The term color space is an area, a gamut, of the colors that can be reproduced. In addition and probably better known there are color models, ways to describe the color with mathematically. For example, well known color models are RGB and CMYK models. The Commission Internationale de l'Eclairage (International Commission on Illumination) (CIE) is the main and the first organization who started standardize color metrics and terminology. The first, well known, and base of modern colorimetry standard was defined by the CIE in 1931 known as CIE XYZ or CIE 1931 color space. This color space is used for base as newer spaces like CIE LUV and LAB. [11, pp.130-133]

[5]Retrieved from: http://commons.wikimedia.org/wiki/File:Simultaneous_Contrast.svg

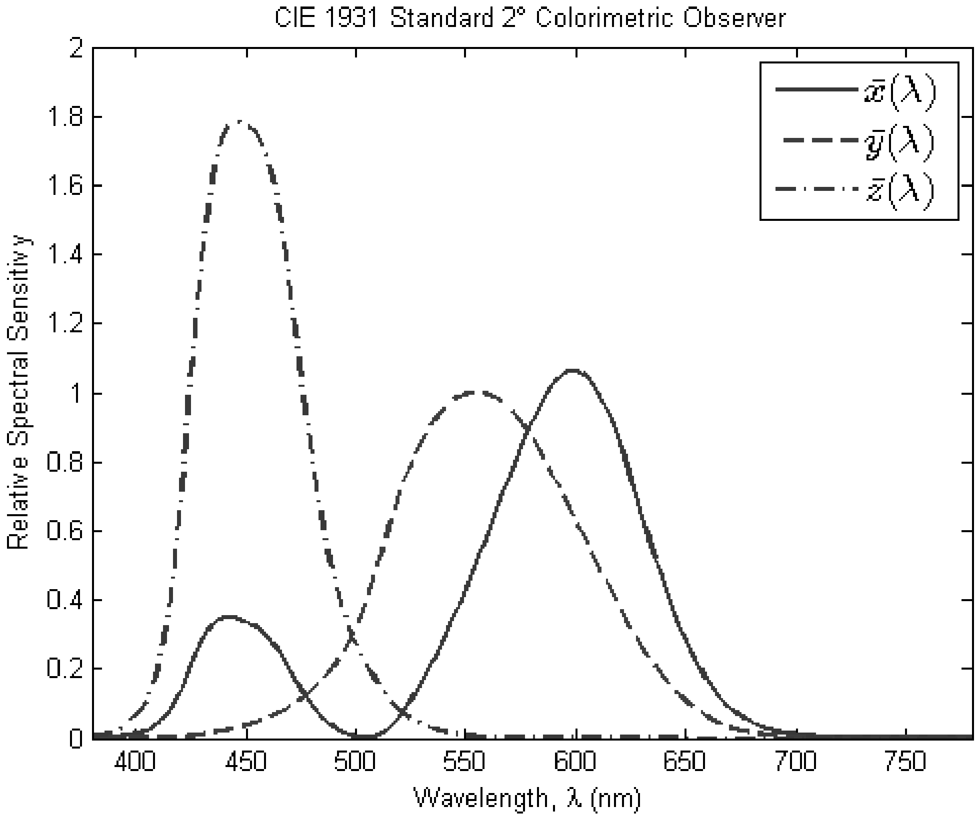

Figure 2.6: The CIE standard observer color matching functions in CIE XYZ color space.

Based on the study done by Wright [12] and Guild [13] in the late 1920s and start of 1930s the CIE created model known as the CIE 1931 Standard Observer. It was based on measurements where observers watched monochromatic stimuli in the wavelength range $\lambda = 400nm$ to $700nm$ using only 2^o of their fovea. Figure 2.6 describes the measured response CIE 1931 Standard Observer. These measurements are used to create the CIE XYZ color space. [11, pp.131-140]

Figure 2.7 describes the gamut that CIE XYZ color space can produce. In CIE XYZ color space each X, Y, and Z represents numerical value of the color. As one can see from Figure 2.7 that color gamut is in 2D-space. This means that original XYZ space is projected on 2D xy-plane. The transform to chromaticity values x and y is done following [11, p.156],

$$x = \frac{X}{X+Y+Z}, \tag{2.12}$$

$$y = \frac{Y}{X+Y+Z}. \tag{2.13}$$

[7] Retrieved from: http://commons.wikimedia.org/wiki/File:PlanckianLocus.png

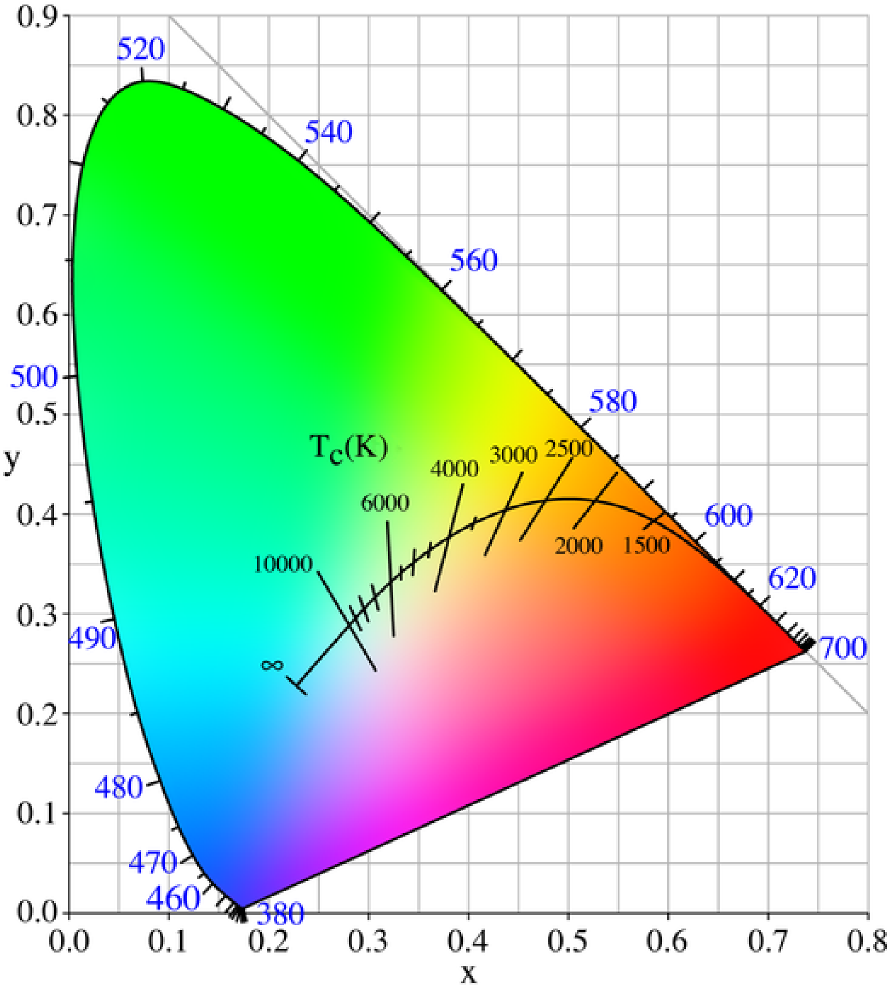

Figure 2.7: Planckian locus with isotemperature lines on top of CIE xy chromaticity diagram.[7]

The CIE xy chromaticity diagram represents all the colors that are visible for humans. The spectral locus, the curved edge of the gamut represents pure wavelength values. For example, if one draw line between 480nm blue and 620nm red all colors lying on that line can be reproduce mixing these two colors [4, pp.60-61]. Often color spaces used in technology do not offer color reproduction in full visible color gamut but much smaller. This is a limitation of technology used, for example, in computer screens.

The CIE XYZ color space is having problem related to a fact that it is not uniformly spaced. In certain situations it might cause that a small difference between two colors in the tristimulus values may not look at all the same tint. This was one of the reasons that drove CIE researchers to develop the CIE LUV (L*u*v*). Nevertheless, CIE LUV managed to solve the uniform spacing problem it does not follow any known models of chromatic adaption. Thus, it is only usable in certain comparison cases. CIE LAB (L*a*b*) color space is a bit more advanced and it is using von Kries type of model of chromatic adaption. It have been noted to work better than CIE LUV even it is more complicated and thus it is most widely used uniform color space. [3, pp.109-111]

The most known and used color space in electronic imaging is sRGB where s stand for standard and R, G, and B are the corresponding main colors. Basically every consumer electronic imaging or viewing device is built to use sRGB. Due to this, one can capture image with one manufacturer's device and view it on another one's without having any major problems on color reproduction. [3, pp.128-129]

2.7 Color Temperature

It is possible to define illumination sources by its hue. This is known as color temperature and it can be used to define some colors of the visible light that are comparable to a black body radiator. The Black body radiator is a theoretical term used to describe an object that absorbs all light coming from outside sources and will emit light when heated. During the black body is heated it will start emit light and the perceived color tint differs on different temperatures. In Figure 2.8 is given hue scale of color temperatures. For example, the sun light behaves very closely to this theoretical model. Often other, artificial, light sources are compared to the black body radiators at different temperatures. This comparison is used to specify Correlated Color Temperature (CCT) of the light source. [3, pp.43-44]

Widely used term Planckian locus refers the curve formed from the color points of changing color temperature in a color space. As one can see in Figure 2.7 the Planckian locus covers only very small part of the color space. This will lead to that quite often illuminant source does not fit into the Planckian locus curve and thus, is needed to get it CCT. It is defined to be that color temperature of the black body radiator that most closely resembles under specified viewing conditions. In Figure 2.7 we can see the isotemperature lines drawn through the Planckian locus curve. These lines are used as a help to get the correct CCT. [11, pp.224-225]

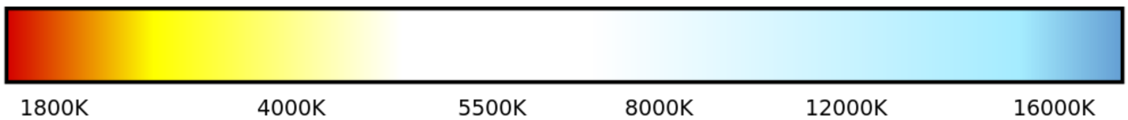

Figure 2.8: Hue scale of color temperatures.[8]

[8]Retrieved from: http://commons.wikimedia.org/wiki/File:Color_temperature.svg

3. DIGITAL IMAGING PIPELINE

In this chapter a brief introduction to optics, image sensors, and image processing is provided to familiarize the environment behind the topic of this thesis. In the following we assume that the reader understands the basics how digital cameras are combination of hardware and software that create and process the digital image from the light rays arriving to the sensor through the optics of the camera.

3.1 Overview

Digital camera modules are embedded in multiple products, e.g., toys, cars, tablets, and smart phones. More and more people use their phones than actual camera to capture moments of their lives. While people are using their mobile devices to replace a compact digital cameras in their daily life the amount of Digital Single-Lens Reflex (DSLR) cameras have also grown rapidly. According to Camera & Imaging Products Association (CIPA) statistics of yearly shipped camera products the DSLR camera shipments have grown steadily over 100% per year over past seven years [14]. On the other hand built-in lens digital cameras growth is started to slow down in the past years.

When more and more professional quality imaging devices became affordable for normal consumers their requirements of image quality have risen even on mobile devices. Mobile devices do not provide the best imaging quality when comparing to the DSLR cameras but they start to be in the same line with compact cameras. Even if they lose in properties they have a big advantage; they are almost always in our pockets.

3.2 Hardware

A typical digital camera is composed of the hardware components presented in Figure 3.1. When capturing the digital image the scene is illuminated by the scene lighting or by the camera flash. The light reflected by the scene is collected through the camera optics. The lens is responsible to focus the light on the image sensor. The sensor samples the light rays from analog to digital format that they are ready for the processing to produce the final output image. [15; 16]

The main differences between the compact digital cameras and the mobile digital cameras are mainly in the lens system and of course size. The mobile phone cameras do not typically contain optical zoom nor adjustable aperture due to their size requirements. Anyhow, these properties are found from most of the compact cameras.

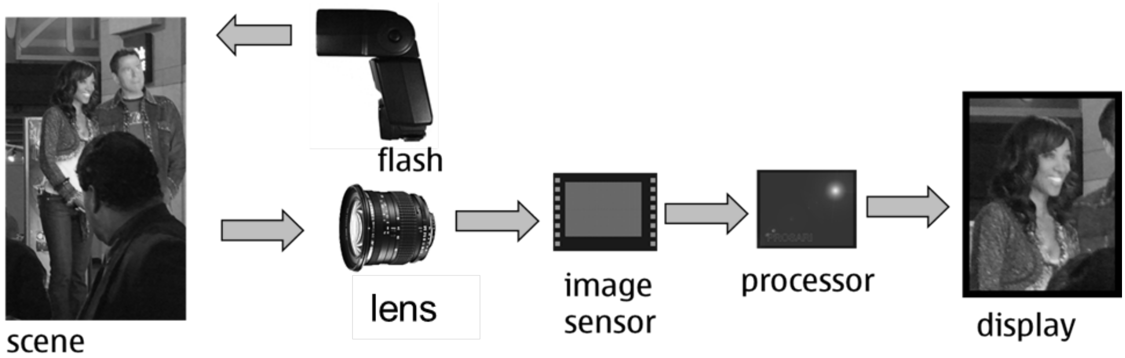

Figure 3.1: Components of a digital camera. [15]

3.2.1 Lenses

Lenses are one of the key components of the camera. They control the light entering the rest of the imaging system. The lens system is composed of multiple individual lenses, an adjustable or fixed aperture and often filters as seen in Figure 3.2 where is a fixed aperture, three lenses and an Infrared (IR) filter. The purpose of the IR filter is to cut out unwanted infrared light because image sensors are typically highly sensitive to those, see also Figure 3.3 [17, p.32]. Optical quality of the lenses is important and if one has low quality lenses it does not matter how many megapixels your camera can capture the result will be poor. Usually optical problems can be seen as soft images, i.e., part of the details in the image are lost. In addition to general optical quality, camera lenses have two key features *an aperture* and *a focal length* that contribute to the final result. [17, pp.21-51]

The former determines how wide is the opening where light rays go through inside the lens. Naturally this has effect to the amount of light going to the sensor when using same shutter speed but it will have also effect how image is focused or more precisely how wide is the depth of field in focused image [17, pp.24-25]. With wide aperture only the plane where lens is focused will be in focus and thus cause narrow depth of field. This effect is called *bokeh* and it is often something that photographers use on purpose to highlight the object from the background. The size of the aperture is described as f-number where smaller value means wider aperture. [19]

The latter will determine the angle of view and the magnification ratio [17, pp.22-23]. The focal length value, usually in mm, tells how far the lens is from the sensor.

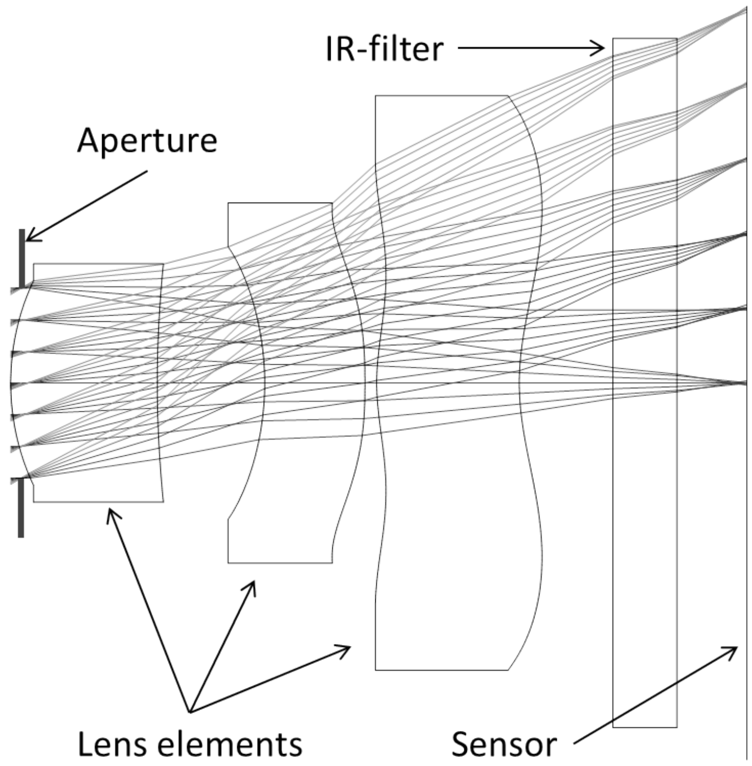

Figure 3.2: A typical mobile phone camera lens element. [18]

When using smaller sensors than so-called full frame, f_{35mm}, which means equivalent to 35-mm film format the actual lens distance from the sensor is smaller but focal length is given as f_{35mm} equivalent for the sake of simplicity [17, pp.13-14].

For a person who does not know anything about cameras the focal length could be explained with magnifying glass example. When playing with magnifying glass in sunlight one can see that moving the glass closer or further between the object and the sun the area illuminated by the sun beam will grow or become smaller. Most of us have played with magnifying glass in our youth and know that when the illuminated area is the smallest the beam is the hottest, i.e., the beam is on focus. When the beam is on focus one can measure the distance between the magnifying glass and the object and this distance is the focal length of the magnifying glass.

Typically there is no optical zoom in the mobile phone cameras due to limited space requirements, i.e., the focal length is fixed. The actual focal length on typical mobile cameras is varying from 3mm to 5mm providing slightly wide-angle f_{35mm} equivalent. [19]

3.2.2 Image Sensor

An image sensor is the film of the modern day and it have revolutionized and popularized photographing. Image sensors are semiconductors that have light-sensitive components. The first step to create a digital image is to convert light rays formed by the optics into an electrical charge using the photoelectric effect [20].

By default the image sensors are not only detecting visible light but any light which photons have enough energy to create a charge. Chosen materials and structure effect the initial sensitivity. However in this thesis we focus only for human visible light and thus the sensors discussed further on are limited by filters and design to detect photons between wavelength range from 380nm to 780nm. [17, p.54]

Color Filter Arrays

The image sensors themselves are 'color blind' and they only detect the amount of light falling into the detector. This will lead to usage of color filters front of image sensors to get individual color channel information together with intensity.

Digital cameras with multiple sensors usually use a trichroic prism as a beam splitter to break incoming light into color components and direct specific color channel to sensor reserved for it [21, p.12]. Digital cameras with three sensors are quite rare but often seen in high quality consumer video cameras. Another relatively new approach is technology called Foveon® from the company of the same name [22]. The Foveon X3 sensor has a single 2D-array of pixels, where every pixel contains three stacked photodiodes to sense different wavelengths of light [23]. This technology mimics the old color film which has three layers one for each color component.

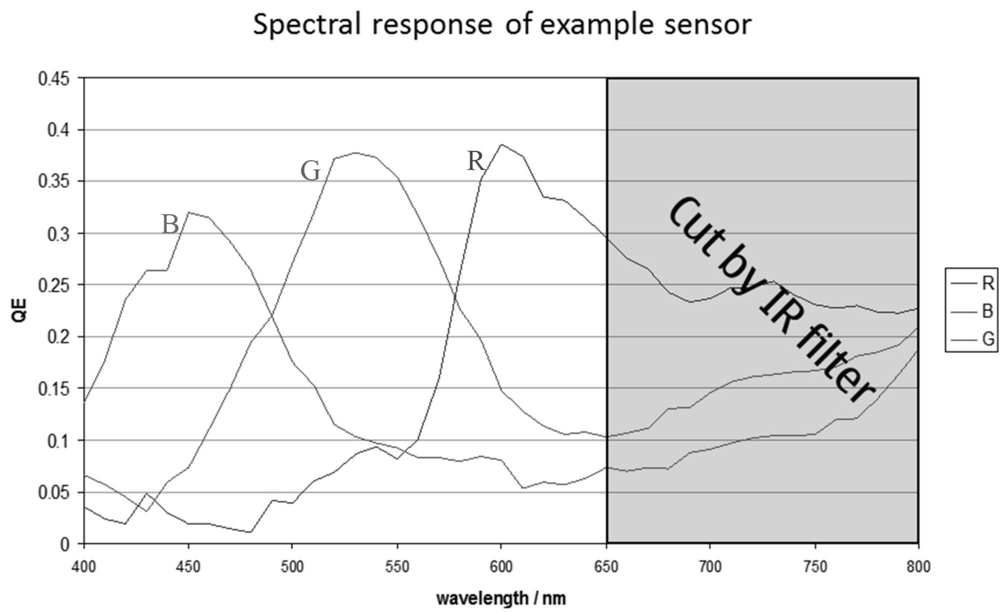

Figure 3.3: Measured spectral response of an example sensor with Bayer color filter array. [15]

Most of the digital cameras use one image sensor to capture the scene. This means that one pixel in the image sensor can not capture every color component but still output image has all three color components for every pixel. This is possible by using color filter array with specific pattern to allow only one color component to reach the specific pixel. The missing color components per pixel is then interpolated to match quite well to original scene.

Most used color filter array pattern is the Bayer pattern, see Figure 3.4. In this pattern there is 50% of G, and 25% of R and B pixels. This pattern is based on knowledge of the HVS which is most sensitive for wavelength corresponding to green color. However, a typical sensor using the Bayer pattern produces quite different spectral sensitivity curves than our cone receptors, compare Figures 2.4 and 3.3.

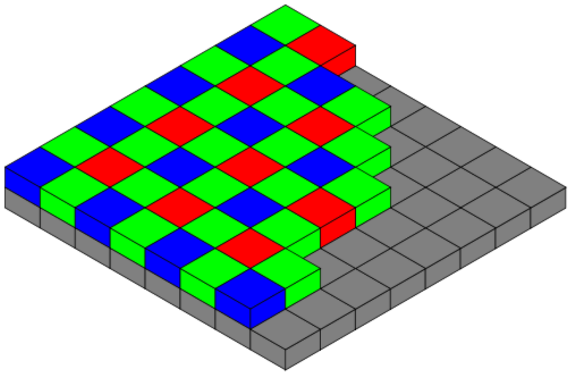

Figure 3.4: Bayer pattern color filter array.[1]

Differently ordered pattern approaches have been studied [24; 25] but Bayer pattern is still the most widely used and it qualifies well in color reproduction. In addition, the latest advances in array design, proposed and studied by Kodak [26], is to add colorless pixels to the array pattern. They have studied that it will improve especially low light performance of the sensor.

Sensor Types

Generally there are two different sensor type used, Charge-Coupled Device (CCD) and Complementary Metal-Oxide-Semiconductor (CMOS), in imaging devices. Their difference is related the technique to convert electrical charge into a voltage and how it is transferred from the sensor [27]. [21, pp.8-11]

Both technologies, CCD and CMOS, are widely used and have their advantages and drawbacks [21, pp.10-11]. CMOS sensors are more popular especially in mobile cameras due to much smaller power consumption and possibility to reduce imaging system size as a result of integration [28].

[1]Retrieved from: http://commons.wikimedia.org/wiki/File:Bayer_pattern_on_sensor.svg

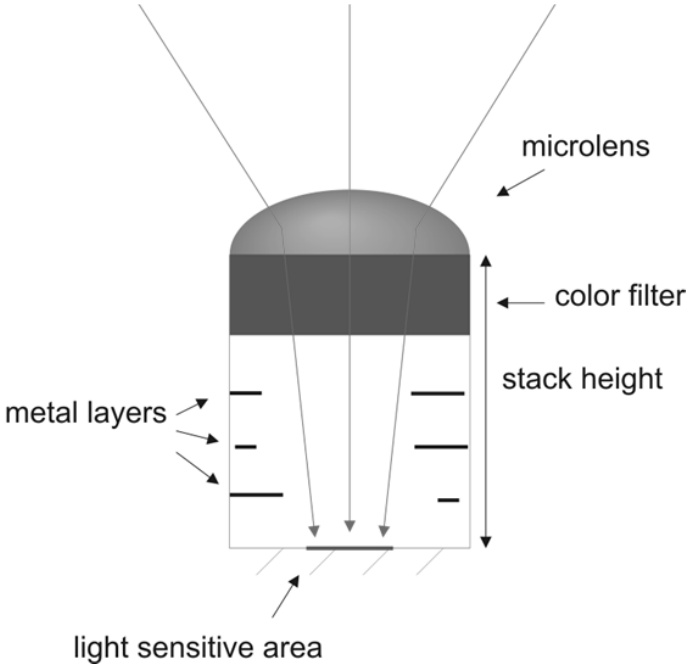

Figure 3.5: Simplified diagram of a pixel in CMOS FSI sensor [15]

Backside illuminated (BSI) sensors is the latest major achievement on the CMOS sensors. It has improved many of the problems that small size CMOS sensors had, e.g., low light performance. In the BSI sensors the pixel structure is turned upside down when comparing the old Frontside illuminated (FSI) sensors, see Figures 3.5 and 3.6. The advantages of this new structure is that the optical and electrical elements of the sensor structure can be separated. Thus these two elements can designed optimally and inpendently. BSI sensors are fairly new technique and more expensive than FSI sensors and it seems that this old and new technologies will continue going side-by-side for several years. [29]

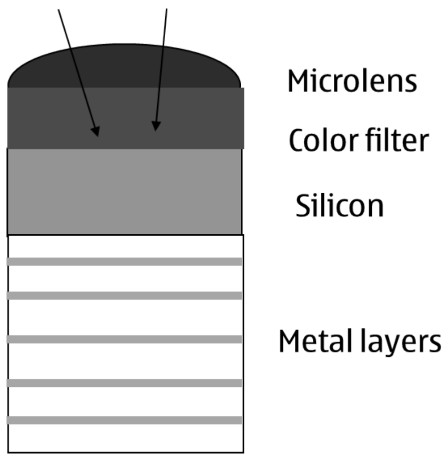

Figure 3.6: Simplified diadgram of a pixel in CMOS BSI sensor [15].

3.3 Image Formation

An unfamiliar camera user might have a feeling that the sensor inside the digital camera sees the scene and shows the output. This is partly correct but from that information what the image sensor sees is still many steps away from the actual output image. This extensive processing is often done inside a specific chip called Image signal processor (ISP). ISP implements process chain that is often referred as Image processing pipeline (IPP), see Figure 3.7. Processing blocks presented in Figure 3.7 are described briefly in following paragraphs.

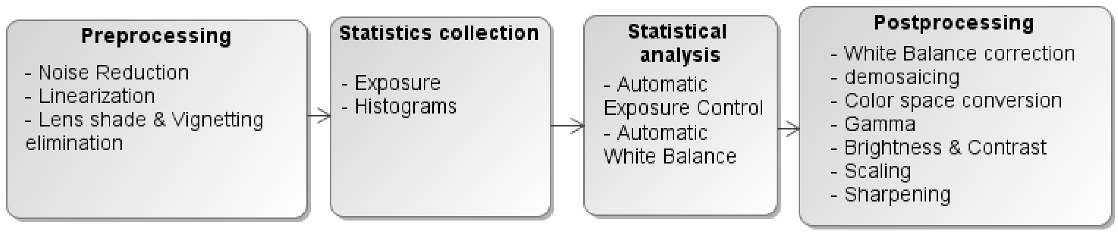

Figure 3.7: Example of an image processing pipeline of a digital camera.

3.3.1 Preprocessing

After the sensor has captured the scene by photoelectric effect and transformed electrons to a digital signals with Analog to Digital Conversion (A/D) is time to start the processing chain. The preprocessing block contains procedures that are performed to the raw data. As stated previous sections most of the times this means the Bayer patterned data.

Phases done during preprocessing block varies between manufacturers but often it contains

- defective pixel correction,
- noise reduction,
- data linearization,
- and lens shading and vignetting elimination. [30]

The image sensors will not be ever noise free thus it need to be compensated with removal methods. The largest source of noise is thermal noise. Together with actual image data the noise is also amplified and converted to result data get from the image sensor. Noise removal algorithms try to identify noise pattern and then remove it. Detailed noise removal algorithms are outside of scope of this thesis but an interested reader is referred to survey in [31].

Image data linearization is used to remove data pedestals and otherwise secure that data is linear for upcoming processing phases. Data pedestals mean an offset in color values after digitalization. These are result from effect known as dark current. It is a phenomenon in the image sensor that cause photodiodes to generate small charge even without actual exposure to light. The Figure 3.8 provide an example of usefulness of the linearization operation. [32]

Figure 3.8: Example images before(left) and after(right) data linearization. [32]

Vignetting is result of sensitivity variations in the image sensor. The greatest exposure is focused on the center of the sensor and decreased resembling a radial function towards the edges, see Figure 3.9. This phenomenon is sometimes specified as pixel vignetting because also lenses and their mechanical built can cause vignetting effect. In preprocessing phase the pixel vignetting is the one which is fixed.

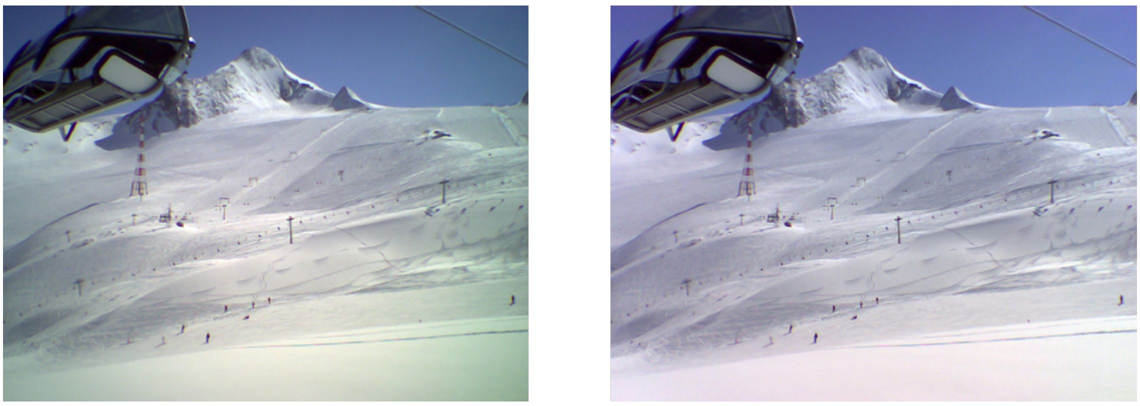

Figure 3.9: Example images before(left) and after(right) vignetting compensation. [32]

3.3.2 Automatic White Balance

White balance or color balance is very important processing phase to get color image look like it was seen by eyes, see Figure 3.10. The basic idea of typical advanced white balance algorithm used in these days cameras is to somehow determine assumed monochromatic area from the captured data and tune this to be correct. Then these same gain factors are used rest of the image. Originally in the raw data this area is not monochromatic thus it will need correcting coefficients and this is the task that white balance algorithms tries to achieve. In upcoming sections this topic is discussed with details and most known algorithms are presented.

Figure 3.10: Example to show meaning of having color correction. Left image does not have color correction and the image in the right has it. [32]

3.3.3 Automatic Exposure Control

Automatic Exposure Control (AEC) component is responsible to determine optimal shutter speed which determines the amount of light collected to the sensor. In Figure 3.11 we have an example of well and over exposed captures and corresponding histograms in Figure 3.12. From the histograms one can see that the over exposed image contains a lot of saturated pixels and less other information than the correctly exposed image. The purpose of the AEC is to trying preserve as good dynamic range in the image as possible. Where it can be boosted with sensor and digital gain values. Usually exposure is determined by analyzing histogram data calculated from the capture or actually from the viewfinder frames to get the actual capture to be correctly exposed. [32]

The human eye have better dynamic range than currently used image sensors and in addition the eye can adapt locally. This means that even we are faced towards the sun we can see details in the clouds and the ground front of us. On the other hand, if we take a picture with camera either the ground is very dark or the sky is over exposed. This dynamic range of the image can be improved, for example, capturing multiple images of the same scene with different exposure values. Then composing them to one image that have higher dynamic range and get the image look more like the human eye sees it.

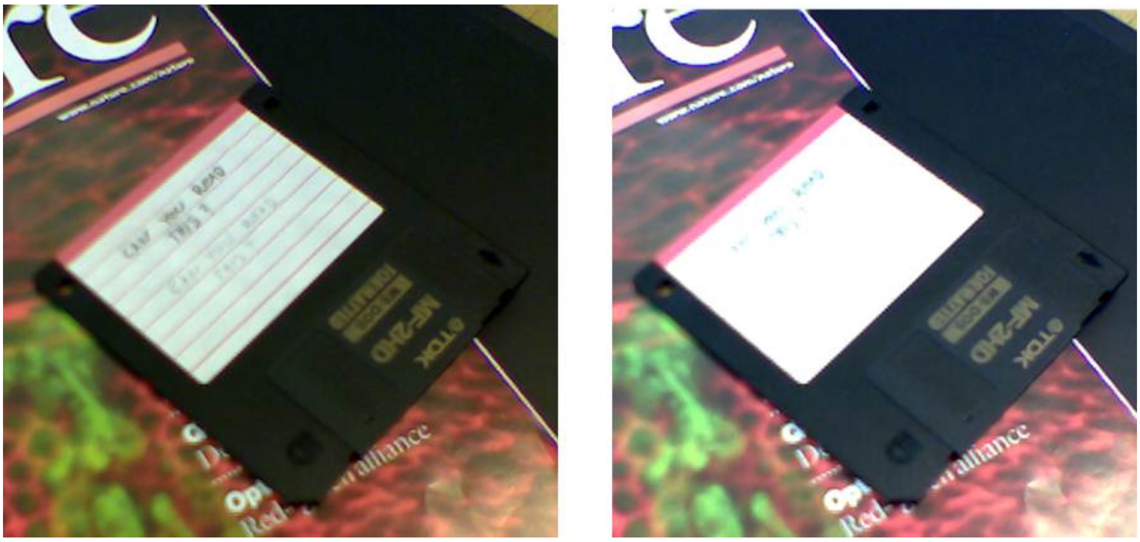

Figure 3.11: An example of well exposured (left) and over exposured (right) images. [32]

3.3.4 Post Processing

Post processing in the digital imaging could be compared to the part executed inside human brains after the eyes have "captured" the moment.

Demosaicing Color interpolation also known as demosaicing is a method to how Bayer raw data is converted to contain all color components for all pixels. This process is not just simple interpolation task but often things like edge detection and color channel correlation information is used to enhance the output. [21, pp.17-18]

Color Transforms After Demosaicing we do not have yet the final color data available. The data is in camera color space, that is, spectral characteristic of the camera components. This data is tuned based on collected and known statistics to get as good as possible color reproduction. Final step of color conversions is to convert the image data to some generally used color space. Most used ones are sRGB and AdobeRGB. [21, pp-18-19]

Gamma correction Purpose of the Gamma correction is to compensate image data for computer monitors which have non-linear light-intensity. Our vision also happens to be non-linear and it is more sensitive to small differences when intensity level of that area is low. As non linear curve gamma mimic the response of human eye and use more bits to represent shadow and mid light details than highlights.

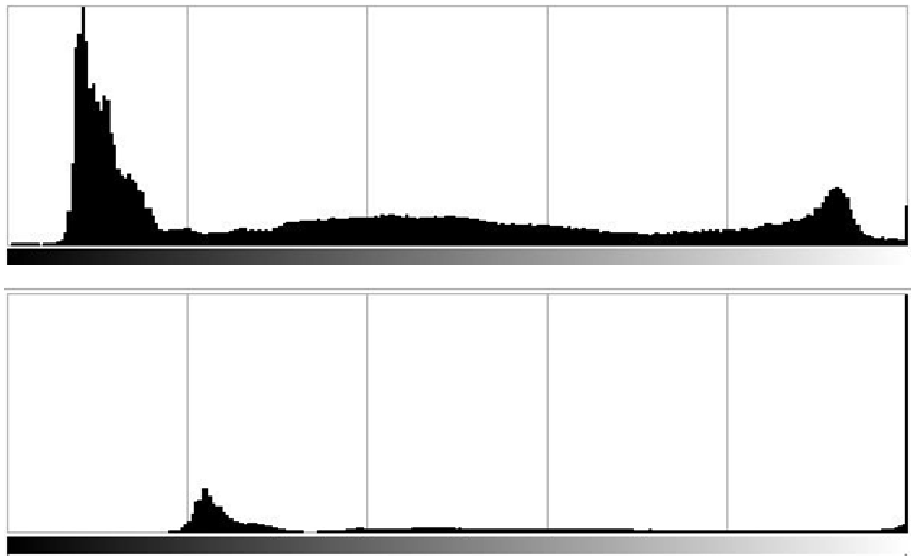

Figure 3.12: Histograms from the images seen in Figure 3.11. The histogram above correspond to the correctly exposed image and the one below to the over exposed.

Gamma correction in simple format can be presented [21; 33, pp.19-20]

$$y = x^{1/\gamma}. \tag{3.1}$$

Image enhancements Before the final image is ready, there is multiple image enhancements operations such as contrast and brightness tuning, sharpening, and noise reduction. It is not relevant to go in details in the scope of this study but interested readers can find more detailed information, for example, from the book Image Sensors and Signal Processing for Digital Still Cameras [17, pp.224-238]. Principle idea of the image enhancements is to make the final image more pleasant and vivid for the end users.

3.3.5 Automatic White Balance Methods

This section discusses the basic and most used AWB methods. Starting from the Gray World method that is often used as a foundation to create more sophisticated AWB algorithms and benchmarking them.

Actual solutions in the products often contain some combination of these algorithms with added manufacturer specific tweaks. Most of the electronics are designed for 6,500 K color temperature, thus it is used as reference where the final result of these algorithms used in device is projected.

In this thesis we used knowledge of these AWB algorithms to build and evaluate our test application.

Gray World & Scale by Max

The Gray World assumption [34] is probably the simplest algorithm trying to determine correct white balance. The approach is to calculate single statistic of the scene and assume that illumination in the scene is uniformly spread. Based on the assumption of uniformly spread data it leads to use a mean value of the scene as a statistics. In other words, the algorithm is based on an assumption that the average reflectance of the surfaces in a scene is achromatic, i.e., gray. The Gray World algorithm can be performed by finding the average of each color channel R, G, and B. Then using the average of averages as a gray value of the image. Each pixel in each color component is then scaled by the amount of how much the average of the color component deviated from the gray value. Weakness of the Gray World algorithm is easily concluded since it is based on idea of uniformly spread R, G, and B data. If the captured scene contains clearly more some of these channels than another this algorithm will fail, for example, taking image of grass field would cause failure.

Another very similar in algorithm point of view but with different base assumption is the Scale by Max algorithm [35]. The formula can be formed from the Gray World by replacing average values with the maximum values of each color channel. But once again, this algorithm is as well easily giving wrong results in real world situations. Quite often the best reflector in the scene will be clipped in the capture and thus the most correct information is lost.

Retinex Theory

Study done by E. H. Land during in the late 1970's [36] proposed a theory based on how human eye works, called the Retinex theory. The study investigates color constancy behavior based on a studies of lightness and color perception of the human visual system. The algorithm tries to classify changes in reflectance and changes in illumination based on assumption that illumination changes are gradual and reflectance changes are suddenly.

Gamut-Mapping

Gamut-mapping, originally presented by Forsyth [37], is one of the most successful AWB algorithms [38]. The goal of the algorithm is to transform the sensor response under unknown illuminant to the corresponding sensor response under canonical illumination. The algorithm is heavily depending the what area of the image is assumed to be white or monochromatic.

Other methods

Beside those presented in previously there are plenty variations of those algorithms and true implementations are often some combination of these. In addition, methods based on Bayesian models [39; 40], neural networks [41], and Color by correlation [42] are studied.

4. MOBILE APPLICATION FOR DATA GATHERING

Mobile or smart phones with digital camera have a solution to calculate the AWB regardless of the manufacturer. Professional photographers with DSLR cameras know or at least they should know the basics of the color theory and be able to tune their color balance from camera during the capture session or afterwards from the captured RAW images. The situation is different with average consumers and the smart phone cameras. Most of them do not know what AWB even means and they just might end up watching poor quality images and blame the quality of the smart phone. Thus, it is important to try achieve AWB algorithm that behaves well in all different illuminations.

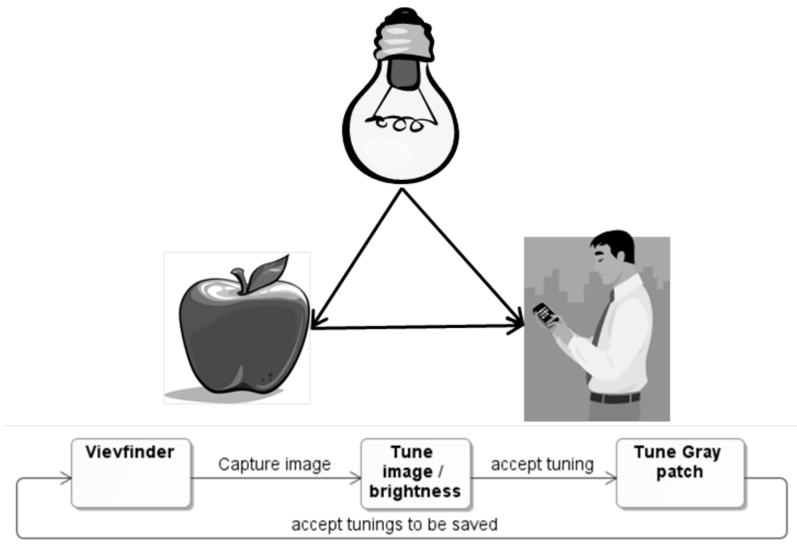

Figure 4.1: An overview of the application use cycle in a figurative environment.

With this mobile application statistical data is collected from selected test group to provide subjective reference to our AWB algorithms. The application allows us to tune the captured image in the same illumination condition immediately after the capturing. This is important since we want that the eyes of the user are still adapted to that illumination. This would not be possible later on the computer screen under office lighting. Also we assume that even though the screen of the device is illuminating different spectrum than the current dominant illumination source the screen is small enough to not interfere the adaption too much. Together with image tuning we provide a gray patch tuning part where the test user tunes the gray patch to look as gray as possible in that dominant illumination. With data collected from the gray tuning we can estimate the amount of color adaption of the eyes.

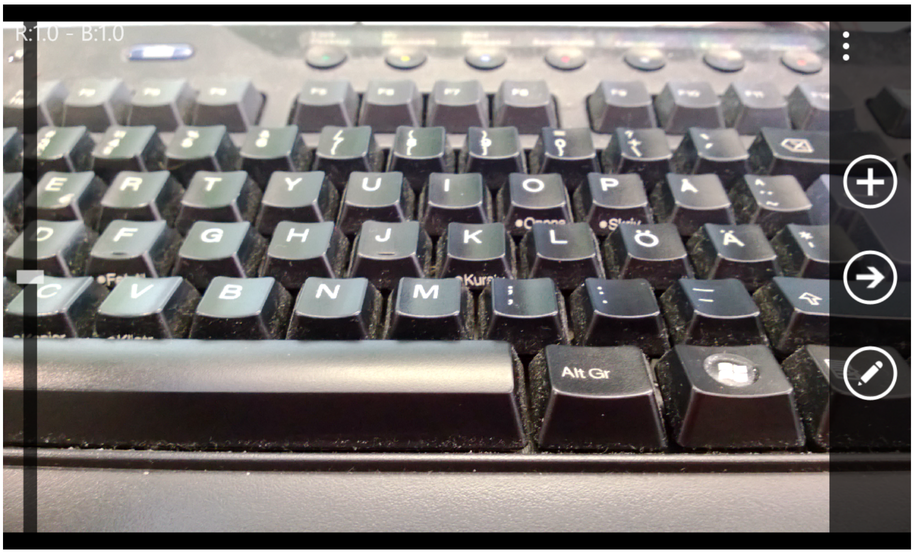

Figure 4.2: The image view mode UI after the capture.

Figure 4.1 describes the usage of the application in a figurative environment. The light bulb illuminates visible light that a person sees. A part of the light hits the apple and it will reflect some spectrum to the eyes of the person, and he sees the apple. The person sees the apple as same red as he would see it under daylight because his eyes have adapted to the dominant illumination source. Now, the person took an image with the application and after capturing the image he will tune the colors to match what his eyes see in that moment, see Figure 4.2. In addition, he can tune the brightness if the color tuning will saturate some pixels. After tuning the image the person will see a gray patch which he will tune to look as much gray as he can. This gray tuning is based on the memory of the gray color since what the person sees on the screen depends on his eye adaption level. When the gray patch is tuned to match his preference the person accepts the tuning and both image and gray tuning values are permanently recorded on the mass memory of the device. Now, the camera viewfinder is shown again and one can take an another image or quit the application.

4.1 Technologies

In this section the main technologies used in this specific application development is described. The new Windows Phone 8 Software Development Kit (SDK) is introduced in general level to provide information how it differs from the previous SDKs. Anyhow, even though we are using specific technologies in this implementation, the other mobile platforms have similar components and same kind of approaches can be used to data gathering and analysis.

4.1.1 Windows Phone 8 SDK

Windows Phone 8 (WP8) is the newest Windows Phone platform that was officially released 29th October 2012. For consumers, it provides a new start screen User Interface (UI) and few new technologies, for example, the Near Field Communication (NFC), mobile payment, and Kids Corner and a lot of small improvements to the whole Operating System (OS). One of the biggest improvements to previous SDK versions are given to application developers. They can now use native C++ or C code to create very effective libraries to their applications. In addition they can use a real multithreaded code to gain even better performance and maintain a good responsive UI layer. Since Windows Phone 8 is built on a shared core with Windows 8 it makes easier to build applications to share same code base. In practice, part of the Application Programming Interface (API)s are same for Windows Phone 8 and Windows 8. This includes also COM and Win32 APIs.

Usage of C++ or C components help developers to do also easier cross platform development. For example, some algorithm or logic component can be re-used on other platforms. Anyhow, this is not a silver bullet for cross platform development. The main application and components is still needed to do as a native for wanted platform. For example, Android, iOS, and Windows Phone 8 all have their own programming language in use but one can utilize c-library for all these.

C# is still the main development language together with Extensible Application Markup Language (XAML) which is meant for composing UIs. In addition to these Microsoft has created C++/CX(Component Extensions) language extension to work as a middle layer between managed C# and unmanaged C++ code [43] and enables C++ developers to write code for the new Windows Runtime platform.

4.1.2 Windows.Phone.Media.Capture Namespace

In this thesis we are relying the implementation on the top of the camera of the smart device. In Windows Phone 8 environment this lead us to use `Windows.Phone.Media.Capture` APIs. This API offers quite large control over the camera device. The physical device is abstracted under two different classes, `AudioVideoCaptureDevice` and `PhotoCaptureDevice`. As one can conclude from the names the former provide video functionality including audio and the latter, which we are also using, provide still imaging functionality

4.1.3 Direct3D & High Level Shading Language

When processing quite large sizes images graphical libraries offer very powerful way to achieve it with low latencies. For example, in the application implemented in this thesis we want to alter the colors and the brightness of the captured image. This could be done by directly altering the image data in the memory by going through every pixel inside a loop. With access to Direct3D [44] interfaces we can create pixel shader that applies provided coefficients to every pixel during the rendering process. This smooth the process and let the user to focus on the task and not the waiting. This will hopefully also help to get better results.

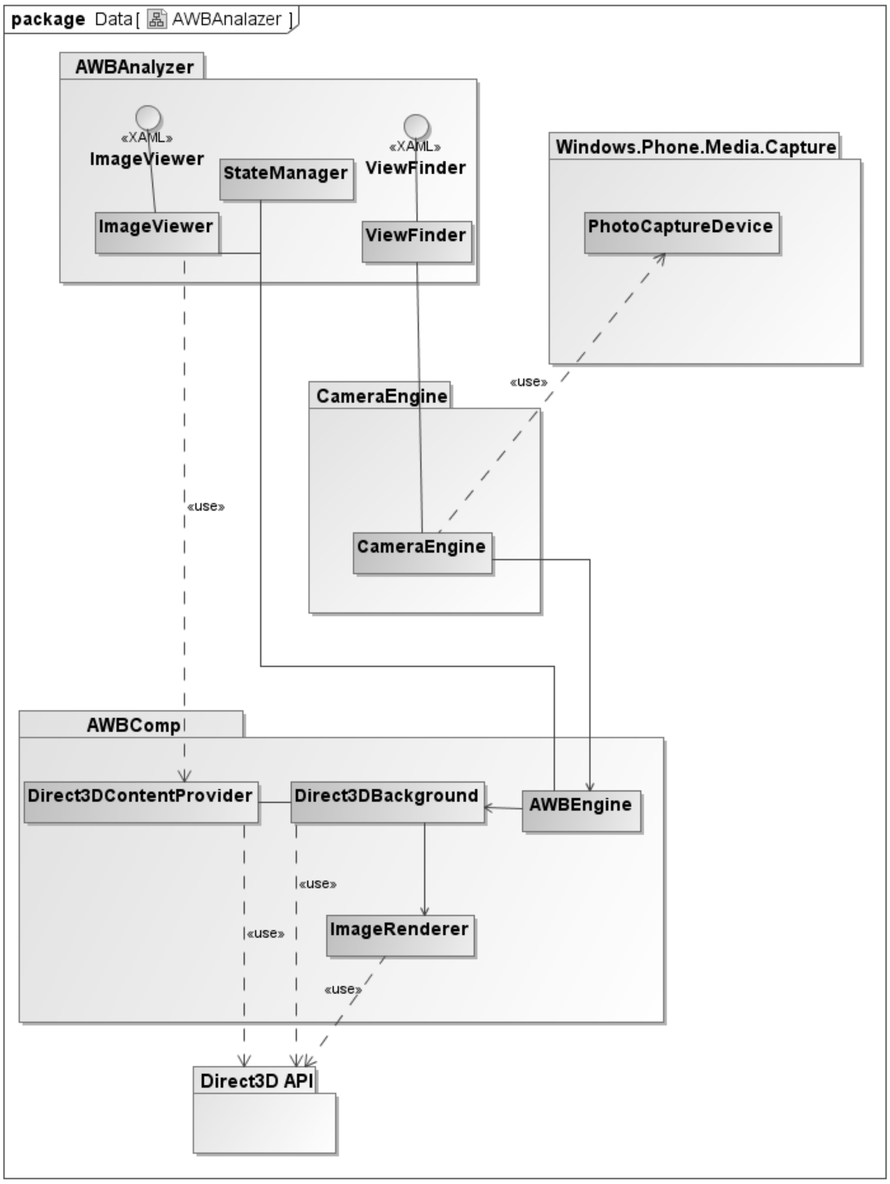

Figure 4.3: Simplified class diagram of the mobile application used for data gathering.

4.2 Architecture

Architecture of the data gathering application is designed to be simple as possible but offering good and smooth user experience. The architectural design is divided into three components, `AWBAnalyzer`, `CameraEngine`, and `AWBComp`, see Figure 4.3. This division is done to keep logical components in their own sub-projects.

AWBAnalyzer This component contains UI components that includes actual visual elements described by XAML and their control classes. `ViewFinder` component works together with `CameraEngine` class to offer camera functionality for a user. `ImageViewer` provide UI and functionality to see captured image and tune the colors, and gray adaption view. In addition, `StateManager` class tracks the different phases during the image view and tuning, and thus helps controlling.

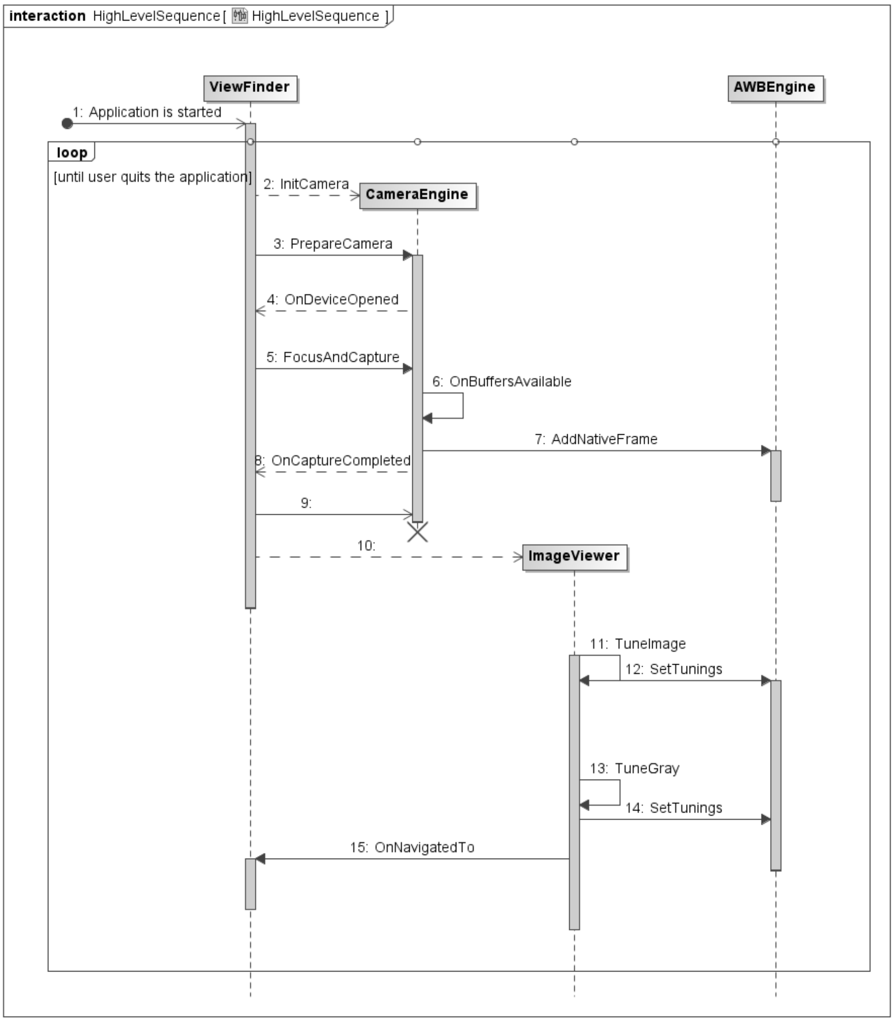

Figure 4.4: High level usage sequence diagram of the data gathering application.

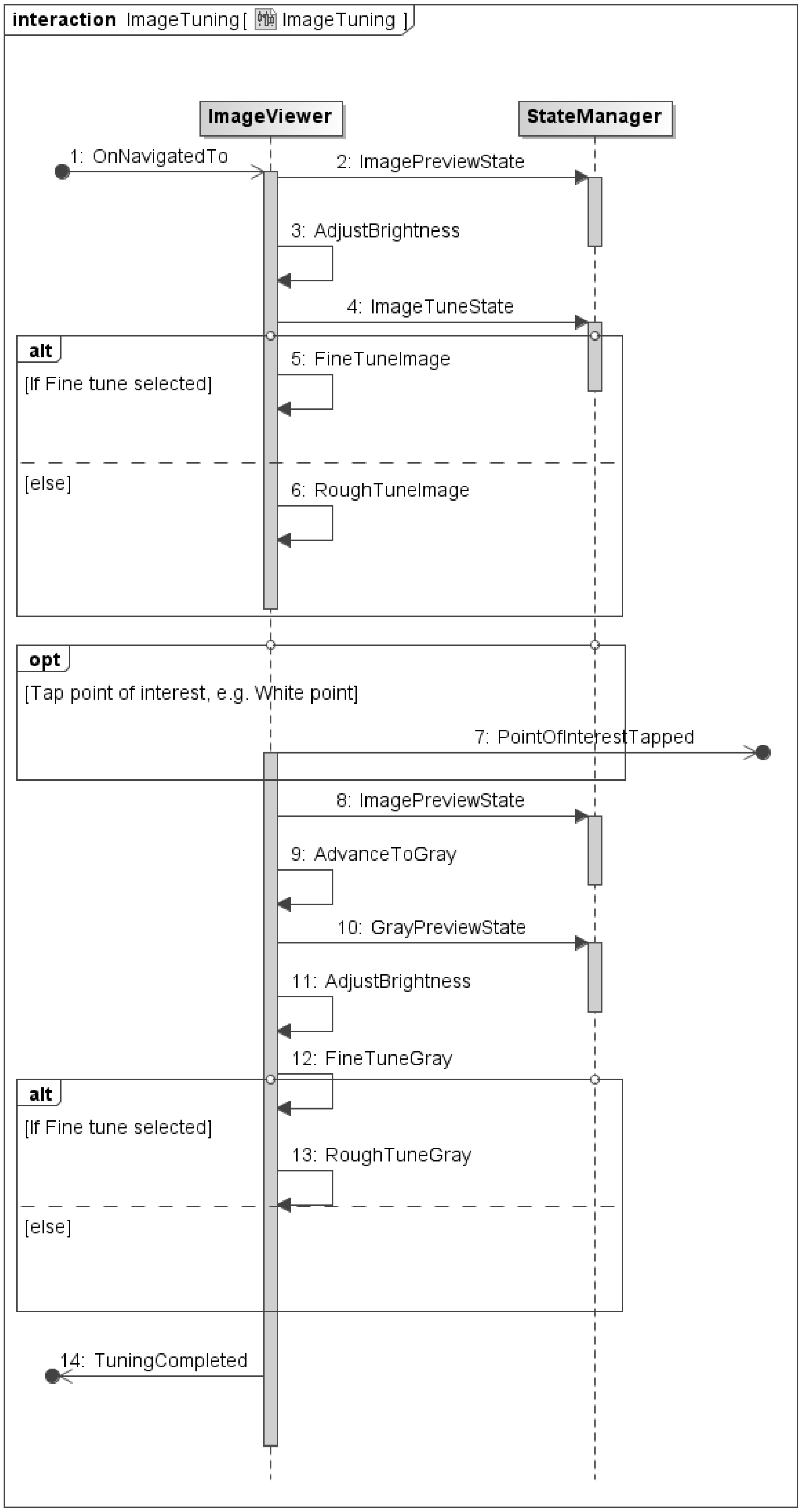

Figure 4.5: Detailed sequence diagram about how user advance during the tuning phase.

CameraEngine `CameraEngine` component is responsible to create and use connection to `Windows.Phone.Media.Capture` APIs. For example, functionality needed for this application are auto focus, flash control during the capture, specified path for output image, and YUV-frame saving for tunings. The `CameraEngine` also have connection point to the `AWBEngine`. It is needed to pass the YUV-frame for later processing.

AWBComp This component is providing tools to tune captured image together the visualization provided through the `ImageViewer`. Its responsibilities are to create Direct3D components and work as middle layer between user actions from the `ImageViewer` UI and Direct3D components.

Figures 4.4 and 4.5 describe the workflow of the application usage. Both sequence diagrams are simplified to contain only key parts of the processes. Figure 4.4 illustrates the complete process chain from starting the application to ending it. Key point from this sequence is to notice that this capture-tune entity is looped until user have took enough pictures. Where Figure 4.5 provide more detailed presentation of this image and gray tuning complex.

4.2.1 Data Format

When user uses the application to capture images and tunes them he or she most likely do not take one capture and upload the data to computer for archiving. This means that we need to be able to individualize the data related to the captured image. And also if user quits the application after capture but before completing the tuning or some unintentional shutdown of the application occur we do not want partial data to be saved and disturb us in later usage.

The uniqueness of the data is tried to achieve via creating unique image name and JSON [45] data format where that unique name work as a key value. The image name must stay unique during the application usage but also quitting and re-opening the application in later usage. This is achieved by persistent counter that saves the identifier in application specific permanent mass-memory. The rest properties saved from the process is thus uniquely bind together with that key value. In addition to the key value we save the tuning values R/G, B/G, and brightness for the image and the gray patch. Also if user have marked region of interest it is saved. Example of one data of the capture and tuning is presented in Listing 4.1.

The JSON format was chosen because it have wide support over common programming language either by native or by addons and generally it is very simple and easy to use. Windows Phone 8 development environment have it built in and for the MATLAB we used community built JSON parser.

Listing 4.1: Example of the JSON formatted data used in the application.

```
{
        "ID": 000001,
        "FullImageName": "AWBAnalyzer000001",
        "RperGForImage": 1.1156,
        "BperGForImage": 0.7925,
        "RperGForGray": 1.0146,
        "BperGForGray": 0.9456,
        "ROICenterX": 0.0,
        "ROICenterY": 0.0,
        "BrightnessForImage": 0.6734,
        "BrightnessForGray": 0.8761
}
```

4.3 Implementation

The implementation was done on Windows Phone 8 platform and is thus relies on the provided camera and graphic related APIs by Microsoft and its associates. Nevertheless, the concept can be implemented to any smart phone platform in the markets. Most likely detailed results of the manufacturers AWB algorithm results are not available to the 3rd party software developers, as also in our case, it will zone this solution to be used inside the manufacturers.

The implementation is divided to UI code and engine logic code. The UI code is done with C# and XAML and is thus managed code, but engine side is done using unmanaged C++/CX and pure C++ code.

4.3.1 Image Capture

The first part of the application is to capture the image in desired illumination conditions. This require us to create instance to the camera component, `PhotoCaptureDevice`. In this kind of special application we do not want ever use the flash of the camera because the light coming from the flash would end up giving us certain AWB results but our eyes would not be adapted to the color temperature of the flash. Through the API of the `PhotoCaptureDevice` we can alter the needed settings of the image to be captured, for example, to turn flash always off.

After the capture the output JPEG-image is saved but we are not interested much about that in this point. What we want, is to alter the color data and visualize it runtime. Thus, what we need is the actual image data not the compressed image file, like JPEG in this case. Luckily there is no need to do any gimmicks like loading the JPEG-file to RGB data or something similar. What we do is that we snap the YCbCr data and copy it to memory for further usage, in our case altering the colors and brightness. Next phase is to show the YCbCr data and give user the tools to alter color appearance, see Figure 4.2. As seen in Figure 4.2 the captured image is shown in the screen. The buttons provided in the right side from up to down provide functionality to add point of interest, the point or area that user have especially mind during the tuning, continue to gray patch tuning, and change to tune mode. In the left side is slider that alters the brightness of the image.

Figure 4.6: The image tuning mode UI of the mobile application.

4.3.2 Image View & Tune

For gaining the optimal performance and usability we are using graphic processor through Direct3D API when altering the pixel values in the image data. Actually the image data inside the memory does not get altered but we can create Direct3D pixel shader[1] that alters the pixel values with really low latency during the pixel is drawn on the screen. The user can adjust color and brightness level values. By altering the brightness level we assure that color adjustment will not saturate too much the actual information. This approach is basically just improvement for the usability but seeing the change instantly versus after latency we believe it improves the actual result as well. The tuning UI is seen in Figure 4.6. The color appearance is tuned by touch in free 2D space where center correspond to the original. The buttons seen in right provide reset and accept tuning, and the brightness slider is available in this mode as well.

[1]http://msdn.microsoft.com/en-us/library/windows/desktop/bb509561%28v=vs.85%29.aspx

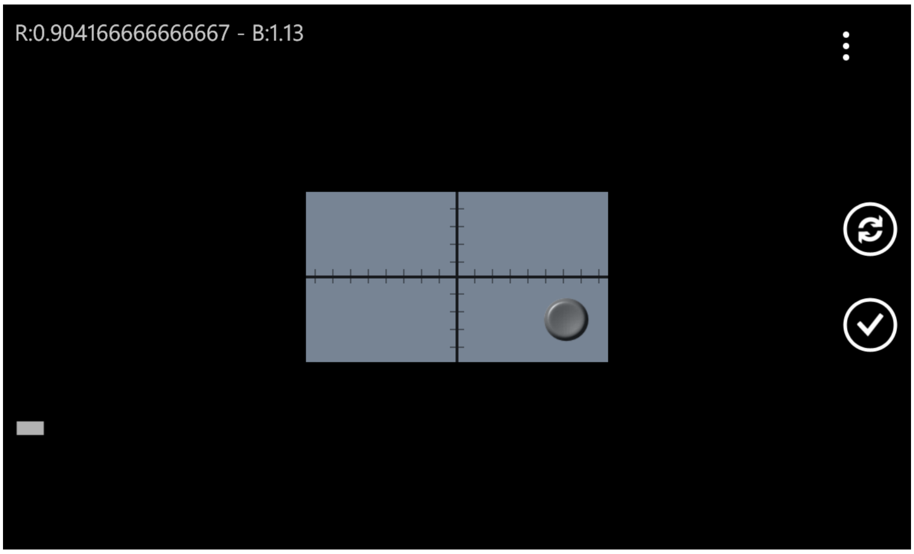

Figure 4.7: The gray tuning mode UI of the mobile application.

After the adjustments are done to the captured image the same kind of adjustments are done to gray box, see Figure 4.7. This is one of the most important phases of this application and also this study. Via tuning the gray box to look gray in dominant illumination we can get chromatic adaption of the user's eye.

5. DATA ANALYSIS METHODS AND RESULTS

In this chapter tools, methods, and results of the data analyzing are explained. As a result of the data analysis we try to determine were the results of our AWB algorithms successfully calculated and did they correspond to a predilection of the eyes of the user.

5.1 Tools for Analysis

MathWorks® MATLAB® [1] is a powerful computation and analyzing tool which provide high-level programming language to develop your own solutions to problems. One of the advantages of the MATLAB is that one can very quickly visualize the data and thus provide usually more better viewpoint for humans. Also, it has large amount of out-of-the-box libraries for many common numerical and statistical analysis. With MATLAB we created script package to first parse needed AWB data from the *MakerNote* that is included in the output JPEG-file which is saved during the data collection.

MakerNote is tag inside Exchangeable image file format (Exif) data that is standardized method to store interchange information in image files [46]. In the Exif 2.2 standard [46] MakerNote is stated as "A tag for manufacturers of Exif writers to record any desired information. The contents are up to the manufacturer, but this tag should not be used for any other than its intended purpose.". For purposes of this thesis study we can, for example, add detailed information of our AWB algorithm results. This means, for example, white balance gains, intermediate AWB results, e.g., from gray world algorithm, and CCT estimation of the scene.

Mobile Application, detailed in the previous chapter, is the key tool for this study because with that we can collect an user-specific data which is of course subjective. Assuming that most of the people have similar color vision this collected subjective data can be used quite reliable analysis of the correctness of the algorithms.

[1] http://www.mathworks.com/products/matlab/

5.2 Gathered Data

The data is gathered via mobile application used by few expert users inside the company. The data contains mainly images taken in randomly chosen locations but will contain also predefined setups, for example, in studio environment and known difficult situations, e.g., sunrise.

The actual data is numerical values that correspond to the adjustments done during the application usage. For example, in Table 5.1 is given adjustment values done during capturing process in certain illumination conditions. In this example setup the dominant light was coming from the horizon as morning sun. In Figure 5.1 left half of the image is the result of the AWB algorithm and right half is the result after user tuning. The AWB result does not look exactly wrong without knowing the capture situation. One could even say it looks correct. Nevertheless, the result is not the same as the average human eyes would have seen the situation. The right side of Figure 5.1 is the result when adjustments presented in Table 5.1 are applied to the image. The right side result correspond more closely to how human eyes saw the scene in that specific light condition.

The R/G and B/G values describes the ratio between red and green, and blue and green. In other words, when the R/G value is 1.0 the red-green ratio is unchanged. Logically when the ratio is larger than 1.0 the image get more red and vice versa. The same logic is valid for the B/G, values over 1.0 will increase the amount of blue and values under 1.0 will decrease it, i.e., the output will be more yellowish. The brightness is optional tuning parameter but often needed to keep part of the image unsaturated. For example, often some of the pixels start to saturate when more radical color tunings are applied.

Table 5.1: Adjustment values from example user in the example image capture.

	R/G	**B/G**	**Brightness**
Image Tunings	1.1156	0.7925	0.6734
Gray Tunings	1.0146	0.9456	0.8761

5.3 Source of Errors

The concept in this study was to figure out is the approach done in this thesis even possible. If yes, how to improve it actually be usable and accurate. Due to this proto concept it was clear that we will have some things that will cause inaccurate data. Partly this is related to the fact that this is subjective study and partly to that we do not know exactly what kind of result we should expect.

Since collected data are subjective opinions of the test users we can assume that

Figure 5.1: Example image before (left) and after (right) user adjustments.

there will be scattering in the results. This will definitely lead into false positive result of incorrect AWB algorithm detection in many cases. One most likely source of error to get inaccurate measurements is that test person is taking too quickly a new image after changing the scene illumination when taking images in studio environment with changeable illumination types. This is a result of our eyes adaption and from the fact that it does not happen instantly but takes few minutes. Subjectivity of this study and especially the gray box tuning part of the application might cause dispersion. The gray box is tuned to look gray that the current user perceives gray relying on his or her memory.

During these tests our AWB algorithms do not use any color appearance model that tries to mimic human eye color inconstancy. This will lead to that the test results taken under low or high color temperatures does have correct AWB, i.e., white is reasonable white. Anyhow, this does not correspond to that what average test user see.

In the application, the adjustments are done with finger gestures via touch screen. This is a compromise between usability and accuracy, and may cause enough inaccuracy that result will be false positive sometimes.

Mixed light situations are also very difficult to handle for AWB algorithms. Even if the algorithm has succeeded mixed light may cause problems. For example, that kind of situation would be scene with mixed illumination conditions where eyes of the user are adapted to other light source more and the camera algorithm to other one due to different field of view.

5.4 Analysis and Test Setup

First, the competence of the mobile application was evaluated inside an laboratory situations where light conditions are accurate and tunable. In controlled situation we can expect certain behavior and thus get confirmation of correct results of the mobile application. After few rounds of improvement iterations to the application the results were usable for gathering data for actual analyzes in daily situations and also known problem cases.

We arranged test scenery in studio environment to get specified data from multiple users. The setup contained studio scenery inside a lightbox, see Figure 5.2. The lightbox is a device that contains multiple light sources that can be controlled. This enables us to take images under certain color temperatures and repeat the process for multiple users to gain comparable data.

Figure 5.2: The scene used in the studio setup. The gray patch used for accuracy measurement is the second from the low left in the ColorChecker.

Three different light sources were used with three brightness level on each. The brightness was tuned to be roughly the same regardless the illumination source, see Table 5.2. Note that the CCT values shown in the table are reference values of the lamp type. In real life we get variation to the CCT values by dropping the intensity. In total we get 9 images per user per set. From the test images we can detect the ground truth color correction values since the studio setup contains MacBeth ColorChecker [47]. The reader should note that we are not expecting to get exact same values as the ColorChecker specification states to that used patch. We are interested to study are the R, G, and B values same or least nearby each other.

Table 5.2: Studio test setup details.

Illumination type	CCT	lux1	lux2	lux3
D65	6504K	500	250	75
F12	3000K	460	235	75
Halogen	2800K	480	260	70

Another test setup was planned to study the color appearance. This test did not involve any particular test group. The idea of this test was kinda similar than the AWB test and it is consisted from multiple images under different color temperatures from about 7500K to 2400K. With these color temperatures we can mainly study the color appearance behavior in low temperature end. To get results in the high temperature end of the color appearance we would need to be able to go higher temperatures but it is not possible in our current test environment. With this test we are trying to find some pattern how R/G and B/G values behave.

5.5 Results

The application can be used to visualize how well the user opinion and the actual AWB algorithm result match. The most interesting observation is that this approach can be used relatively well to get eye adaption. From the results we can easily visualize the output image with user tunings, see Figure 5.3.

During the study we noticed that this application is actually more usable for color appearance study than the actual AWB accuracy measurements. Even though, the application is still usable for pick clearly false AWB results. With these current implementations we can not create any automatic classifier to determine did the AWB went wrong or did we perform under color temperatures where color appearance should has been take into account. For automating we need to know more about the captured scene and also need more extensive data set.

5.5.1 Test Setup Results

The upcoming Tables 5.3 to 5.5 shows results of each illumination type used in the studio setup. The measured data is averaged to minimize the variation between test users. The last row of each table contains column wise average and the standard deviation, σ, is presented inside the parenthesis. Each user was supposed to take the setup three times but due to scheduling issues we managed to took only one set per user. In total we had 4 test users and one image of D65 lux3 category was failed resulting total of 35 images.

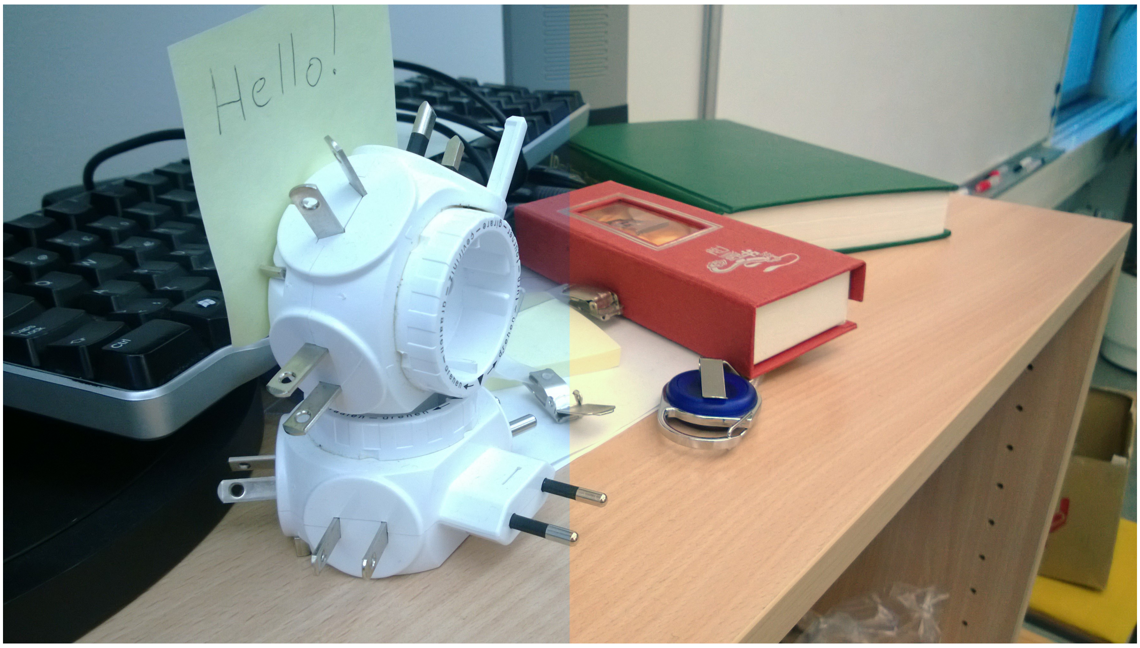

Figure 5.3: An example image of situation where AWB has not performed optimally. Right part of the image visualize the user tunings.

Test images taken under standard illumination D65 should not contain much of tunings if AWB algorithms have worked well. This is a result of sRGB color space where the output images are transformed during the imaging pipeline and sRGB whitepoint is defined to be the same as D65 whitepoint. As we can see from Table 5.3 the image tuning values are close to 1.0 what is expected. The gray tuning values are also the same and since we know that the display of the device works in sRGB color space this is the expected result once again. One can note that there is not really variation between the lux levels and this correspond to our AWB algorithm results of the estimated CCT.

Table 5.3: Studio test setup D65, average user results from mobile application.

	R/G Image	**B/G Image**	**R/G Gray**	**B/G Gray**
Avg. D65 lux1	0.9578	0.9772	0.9552	1.0081
Avg. D65 lux2	0.9802	1.0078	0.9818	1.0528
Avg. D65 lux3	1.0069	1.0375	0.9910	1.0204
Avg. and σ	0.982 (0.053)	1.008 (0.055)	0.976 (0.029)	1.027 (0.045)

The next illuminant was the F12. From the averaged results presented in Table 5.4 one can see that B/G Image values are a bit under 1.0. It means that there is slight reduce in blue values, i.e., the image is tuned toward yellow. This is expected result and corresponds the CCT value of the F12 illuminant.

The last illuminant in this test was halogen. The halogen was only lamp type

Table 5.4: Studio test setup F12, average user results from mobile application.

	R/G Image	**B/G Image**	**R/G Gray**	**B/G Gray**
Avg. F12 lux1	1.0636	0.9016	1.0230	1.0057
Avg. F12 lux2	1.0521	0.9056	1.0203	1.0281
Avg. F12 lux3	1.0755	0.9115	1.0104	1.0410
Avg. and σ	1.064 (0.057)	0.906 (0.038)	1.018 (0.020)	1.021 (0.074)

where we get visible difference when moving to lower luminance levels. One can see this from Table 5.5 where B/G Image ratios are clearly toward yellow and when looking B/G Image column wise the values are decreasing. Also with the lowest luminance, lux3, the light seen starts to look little reddish and R/G Image is clearly above 1.0. These halogen results are also behaving as expected when knowing its CCT value.

Table 5.5: Studio test setup Halogen, average user results from mobile application.

	R/G Image	**B/G Image**	**R/G Gray**	**B/G Gray**
Avg. Halo lux1	1.030	0.764	0.9813	1.007
Avg. Halo lux2	1.026	0.730	0.9898	0.988
Avg. Halo lux3	1.125	0.678	1.0120	1.000
Avg. and σ	1.060 (0.097)	0.723 (0.081)	0.994 (0.029)	0.998 (0.095)

In general, the results got from the gray patch tunings during the each illumination type did not appear as expected. It was assumed that with the illumination types F12 and specially Halogen there would have been clear values under 1.0 in the B/G Gray columns. From the standard deviation we can note that the test group have not been unanimous in B/G Gray cases with the illuminations F12 and Halogen. With our own tests we wanted to reduce the blue from the gray patch when we were under the Halogen lamp or any other illumination which CCT is below 6500K and our eyes were adapted to its yellowish light. And this is what logically should happen. Since when the eyes are adapted to the yellowish light from the Halogen they become less sensitive to yellow and more sensitive to blue. This should lead us to see the average gray patch little too bluish, and thus we would like to reduce the blue tint.

The reason for these errors might be that the test user was too deeply staring the device screen and started to adapt the sRGB comparable light from it. Partly this can result of unclear guidance during the test session.

Table 5.6: Studio test setup, ground truth white balance and AWB performance. Presented values are averages of the data set.

Illumination	**AWB R|G|B**	**Tuned R|G|B**	$\mathbf{\Delta E^*_{ab}}$
D65 lux1	168 | 175 | 184	169 | 175 | 178	3.494
D65 lux2	168 | 175 | 183	168 | 175 | 175	5.103
D65 lux3	167 | 172 | 180	170 | 172 | 183	4.812
F12 lux1	196 | 192 | 198	204 | 192 | 179	6.770
F12 lux2	193 | 190 | 196	199 | 190 | 174	7.949
F12 lux3	187 | 182 | 192	199 | 182 | 169	8.662
Halogen lux1	182 | 191 | 210	191 | 191 | 161	14.741
Halogen lux2	185 | 190 | 203	192 | 190 | 151	15.410
Halogen lux3	180 | 184 | 195	199 | 184 | 133	18.478

The AWB values from the camera automatic and the tuned values taken from the gray patch of the color checker and compares them to ground truth value are presented in Tables 5.6 and 5.7. As stated before, we are not interested to compare the exact RGB value to the ground truth value but compare is the result R, G, and B values equal when knowing that they are originally equal, i.e., is the patch seen as a monochromatic after the AWB. Table 5.6 lists the RGB values straight from the AWB and after tuning data is applied. We can see from the results that the AWB is behaving quite stably but time to time there is bit too much blue. And from the tuning results we can note that blue level is dramatically reduced when moving to the lower color temperatures.

The color difference between AWB result and the user preference can be described using the Euclidean distance, $\mathbf{\Delta E^*_{ab}}$. The distance is measured between the color values in the CIELAB color space. The literature [1] provide a value 2.3 to be a limit for Just noticeably difference (JND). JND values above 2.3 means that color difference is more and more visible. From these results we see the familiar trend that when moving to the lower color temperature the difference between AWB and the user preferences start to increase due to the color appearance.

In Table 5.7 the ratios R/G, and B/G are calculated. The G-channel is tied value as one can note from Table 5.6 that G values are not changing when comparing the AWB and the tuning results. The ratios should be 1.0 when the AWB algorithm has worked perfectly. As we can see from the table the results are scattered around 1.0 as they should be if ignoring the color appearance effect when moving to the lower temperatures.

Table 5.7: Studio test setup, error percentage when ground truth value compared AWB result before and after user tunings.

Illumination	**R/G**	**B/G**	**R/G Tuned**	**B/G Tuned**
D65 lux1	0.959	1.049	0.963	1.016
D65 lux2	0.960	1.049	0.961	1.003
D65 lux3	0.975	1.049	0.990	1.066
F12 lux1	1.023	1.035	1.063	0.935
F12 lux2	1.014	1.030	1.047	0.914
F12 lux3	1.027	1.053	1.091	0.926
Halogen lux1	0.953	1.101	1.000	0.844
Halogen lux2	0.971	1.070	1.008	0.792
Halogen lux3	0.977	1.063	1.084	0.721

5.5.2 Color Appearance Results

In addition to the AWB test we studied the color appearance. The main study was to test is the application usable for gathering color appearance data, i.e., is the data behaving according to what we expect or randomly.

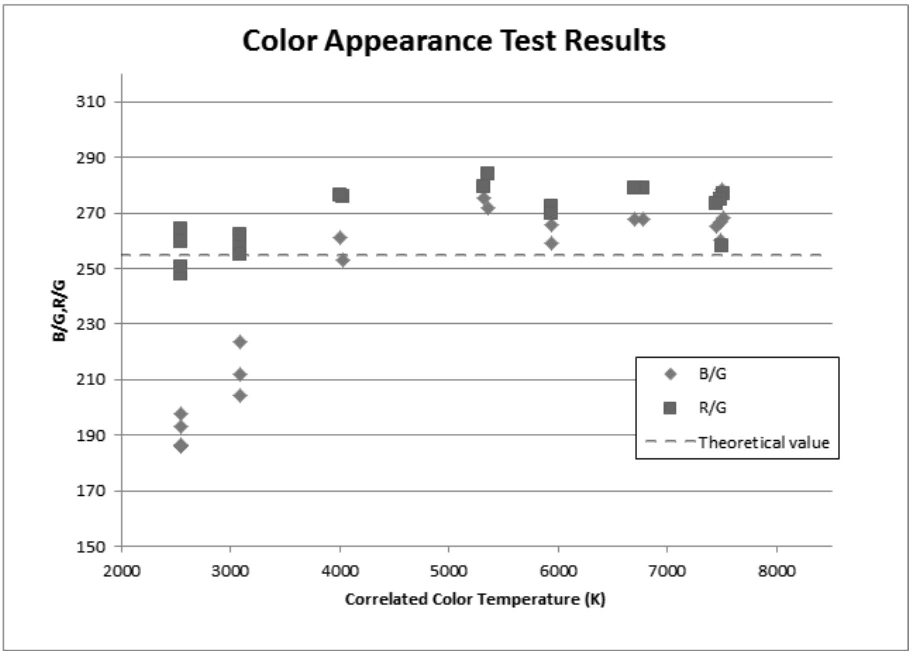

Figure 5.4: Color appearance test results.

In total this test consisted 21 test images taken from the light box containing a flat gray board. The results from the test is presented in Figure 5.4. We can see from the results that when moving to lower color temperatures we start to see the scene more and more yellowish. This means that the adaption of our eyes are not anymore perfect, i.e., we would see the gray board as gray, but it starts to lean to yellow. In the figure the dashed line shows the theoretical correct value when the adaptation is perfect.

This was a preliminary test but very promising. We need larger amount of test data from multiple users to be able to create model from it. And also the behavior in the high temperature end need to be studied.

6. CONCLUSIONS

In this thesis we created tools to determine the accuracy of the Automatic White Balance (AWB) algorithms used in our smart devices. The study was based on subjective analysis what the eyes of the test person see and how he or she would like to adjust the captured image to correspond to the seen. The idea behind the concept is based on the knowledge of the Human Visual System (HVS), especially chromatic adaption. This phenomenon together with the knowledge of that the captured image is transformed to the sRGB color space creates us a possibility to measure the difference between AWB result and the tuning result. This is valid even when the algorithms work perfectly as long as the dominant illumination differs from 6500K which is the white point temperature of the sRGB. In addition, these tools were used to measure color appearance behavior of the HVS.

The results of this study were quite positive even though we could not specify any automatic classifying for correct or incorrect AWB result. In that part more testing is needed with larger test material in the future. Anyhow, this proofed that very basic idea of tuning the colors of the image can be extended with another simple idea of tuning the gray patch. And, when combining the information gained from these tunings we can both examine the AWB algorithms performance and the color adaption of our eyes. Implementing this in a form of mobile application enables us to collect data from quite big test group inside the company.

The application was used to create a frame for the color appearance model to our current AWB algorithm. This convinced us even more that with this kind of simple concept we can get quite reliable result of the eyes adaption. This motivates to continue and improving the concept more on in the future.

Based on the AWB test setup, alternative approaches to the gray patch tuning is need to be studied that we are able to gain more accurate results. One idea could be following: we would not give full tuning option to the user but offer, for example, 3x3 grid of differently shaded gray patches to pick the most pleasant patch, i.e., the most gray. The study contained many possible error sources which are partly relating to the subjective nature of this work and partly can be improved with testing-feedback-fixing iteration. Thus, most important thing for further development would be to make the concept more error proof. In addition, we should study more the theory behind the concept, for example, do we need to pay attention to gamma and color correction matrices that are executed to the image after AWB gains.

For future this same concept could be used, for example, to evaluate the saturation in the images. Also for future usage with bigger test groups some database solution is needed together with more automatized analyzing solution.

BIBLIOGRAPHY

[1] Sharma G. In: Sharma G, editor. Color fundamentals for digital imaging. Digital Color Imaging Handbook. CRC Press; 2002. Available from: http://dx.doi.org/10.1201/9781420041484.ch1.

[2] Lam EY. Combining gray world and retinex theory for automatic white balance in digital photography. In: Consumer Electronics, 2005. (ISCE 2005). Proceedings of the Ninth International Symposium on; 2005. p. 134–139. ID: 1.

[3] Lee HC. Introduction to Color Imaging Science. Cambridge University Press; 2005. Available from: http://dx.doi.org/10.1017/CBO9780511614392.

[4] Berns RS, Billmeyer FW, Saltzman M. Billmeyer and Saltzman's principles of color technology. Wiley-Interscience publication. Wiley; 2000. Available from: http://books.google.fi/books?id=Ss_vAAAAMAAJ.

[5] Johnson G, Fairchild M. In: Sharma G, editor. Visual psychophysics and color appearance. Digital Color Imaging Handbook. CRC Press; 2002. Available from: http://dx.doi.org/10.1201/9781420041484.ch2.

[6] Stockman A, MacLeod DIA, Johnson NE. Spectral sensitivities of the human cones. J Opt Soc Am A. 1993 Dec;10(12):2491–2521. Available from: http://josaa.osa.org/abstract.cfm?URI=josaa-10-12-2491.

[7] Forsvik H. SGN-3637 Human Visual System; 2012. Lecture notes SGN-3637, Tampere University of Technology.

[8] Jameson D, Hurvich LM. Essay Concerning Color Constancy. Annual Review of Psychology. 1989;40(1):1–24. PMID: 2648972. Available from: http://www.annualreviews.org/doi/abs/10.1146/annurev.ps.40.020189.000245.

[9] J A von Kries. In: MacAdam DL, editor. Influence of adaptation on the effects produced by luminous stimuli. Sources of Color Science. MIT Press; 1970. p. 121–126.

[10] Fairchild MD. Color Appearance Models. The Wiley-IS&T Series in Imaging Science and Technology. John Wiley & Sons; 2005.

[11] Wyszecki G, Stiles WS. Color Science: Concepts and Methods, Quantitative Data and Formulae. Wiley classics library. John Wiley & Sons; 2000. Available from: http://books.google.fi/books?id=_51HDcjWZPwC.

[12] Wright WD. A re-determination of the trichromatic coefficients of the spectral colours. Transactions of the Optical Society. 1928;30.

[13] Guild J. The Colorimetric Properties of the Spectrum. Philosophical Transactions of the Royal Society of London Series A, Containing Papers of a Mathematical or Physical Character. 1932;230(681-693):149–187. Available from: http://rsta.royalsocietypublishing.org/content/230/681-693/149.short.

[14] Camera & Imaging Products Association. Statistical Data - Digital Cameras; 2012.

[15] Kalevo O. Advanced Camera&Optics; 2008. Nokia internal training material.

[16] Spaulding K, Parulski K. In: Sharma G, editor. Color image processing for digital cameras. Digital Color Imaging Handbook. CRC Press; 2002. Available from: http://dx.doi.org/10.1201/9781420041484.ch12.

[17] Nakamura J. Image Sensors and Signal Processing for Digital Still Cameras. 1st ed. Boca Raton, FL: CRC Press, Taylor & Francis Group; 2006.

[18] Nummela V. Camera Lenses; 2008. Nokia internal training material.

[19] Cambridge in Colour. UNDERSTANDING CAMERA LENSES;. [WWW] Cited: March 22, 2013. Available from: http://www.cambridgeincolour.com/tutorials/camera-lenses.htm.

[20] Truesense Imaging Inc. Converion of Light to Electronic Charge. Truesense Imaging Inc.; 2012. 1.0 PS-0061.

[21] Bazhyna A. Image Compression in Digital Cameras [Ph.D. dissertation]. Tampere University of Technology; 2009.

[22] Foveon Inc . Website;. [WWW] Cited: January 28, 2013. Available from: http://www.foveon.com/.

[23] Lyon RF, Hubel PM. Eyeing the Camera: into the Next Century. Foveon Inc. Santa Clara, California, USA; 2003.

[24] Lukac R, Plataniotis KN. Color filter arrays: design and performance analysis. Consumer Electronics, IEEE Transactions on. 2005 nov;51(4):1260 – 1267.

[25] Lukac R, Plataniotis KN. Color Filter Arrays for Single-Sensor Imaging. In: Communications, 2006 23rd Biennial Symposium on; 2006. p. 352 –355.

[26] Compton J, Hamilton J. Color Filter Array 2.0;. [WWW] Cited: January 26, 2013. Available from: http://pluggedin.kodak.com/pluggedin/post/?id=624876.

[27] Magnan P. Detection of visible photons in CCD and CMOS: A comparative view. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. 2003 5/21;504(1-3):199–212.

[28] Litwiller D. CMOS vs. CCD: Maturing Technologies, Maturing Markets. Laurin publishing: Photonics Spectra. 2005 August;.

[29] Aptina Imaging Corporation. An Objective Look at FSI and BSI. Aptina Imaging Corporation; 2010.

[30] Ramanath R, Snyder WE, Yoo Y, Drew MS. Color image processing pipeline. Signal Processing Magazine, IEEE. 2005 jan;22(1):34 – 43.

[31] Chatterjee P, Milanfar P. Is Denoising Dead? Image Processing, IEEE Transactions on. 2010;19(4):895–911.

[32] Kalevo O. Introduction to Nokia Image Perfection System.; 2005. Nokia internal training material.

[33] Poynton C. Frequently Asked Questions about Gamma; 1998. [WWW] Cited: April 4, 2013. Available from: http://www.poynton.com/GammaFAQ.html.

[34] Buchsbaum G. A spatial processor model for object colour perception. Journal of the Franklin Institute. 1980;310(1):1 – 26. Available from: http://www.sciencedirect.com/science/article/pii/0016003280900587.

[35] Barnard K, Cardei V, Funt B. A comparison of computational color constancy algorithms. I: Methodology and experiments with synthesized data. Image Processing, IEEE Transactions on. 2002;11(9):972–984. ID: 1.

[36] Land EH. The Retinex Theory of Color Vision. Scientific American. 1977;237(6):108 – 128.

[37] Forsyth DA. A novel algorithm for color constancy. International Journal of Computer Vision. 1990;5:5–35. Available from: http://dx.doi.org/10.1007/BF00056770.

[38] Gijsenij A, Gevers T, Weijer J. Generalized Gamut Mapping using Image Derivative Structures for Color Constancy. International Journal of Computer Vision. 2010;86:127–139. Available from: http://dx.doi.org/10.1007/s11263-008-0171-3.

[39] Brainard DH, Freeman WT. Bayesian color constancy. Journal of the Optical Society of America A. 1997;14:1393–1411.

[40] Rosenberg CR, Minka TP, Ladsariya A. Bayesian Color Constancy with Non-Gaussian Models. In: Thrun S, Saul LK, Schölkopf B, editors. Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, December 8-13, 2003, Vancouver and Whistler, British Columbia, Canada]. MIT Press; 2003. .

[41] Kuronen P. Estimation of Color Temperatures with Neural Networks [M.Sc. dissertation]. Tampere University of Technology; 2006.

[42] Finlayson GD, Hordley SD, HubeL PM. Color by correlation: a simple, unifying framework for color constancy. Pattern Analysis and Machine Intelligence, IEEE Transactions on. 2001;23(11):1209–1221. ID: 1.

[43] Kate Gregory. Managed, Unmanaged, Native: What Kind of Code Is This?;. [WWW] Cited: February 16, 2013. Available from: `http://www.developer.com/net/cplus/article.php/2197621/Managed-Unmanaged-Native-What-Kind-of-Code-Is-This.htm`.

[44] Microsoft. Direct3D for Windows Phone 8;. [WWW] Cited: March 24, 2013. Available from: `http://msdn.microsoft.com/en-us/library/windowsphone/develop/jj207062(v=vs.105).aspx`.

[45] Crockford D. The application/json Media Type for JavaScript Object Notation (JSON); 2006. RFC 4627. Available from: `http://tools.ietf.org/html/rfc4627`.

[46] Exchangeable image file format for digital still cameras: Exif Version 2.2 [JEITA Standard]. Japan Electronics and Information Technology Industries Association; 2002.

[47] X-Rite Inc . ColorChecker Classic;. [WWW] Cited: March 21, 2013. Available from: `http://xritephoto.com/ph_product_overview.aspx?ID=1192`.