

Uniwersytet Kardynała Stefana Wyszyńskiego w  
Warszawie  
Wydział Matematyczno-Przyrodniczy  
Szkoła Nauk Ścisłych

Mariusz Niedziółka

Nr albumu: 99287

Kierunek: informatyka

# Przestrzenie barw konstruowane na podstawie serii fotografii o różnym naświetleniu

Praca magisterska  
Promotor:  
dr Paweł Łubniewski

Warszawa 2020

.....  
Imię i nazwisko studenta/studentki

.....  
Nr albumu

.....  
Wydział

.....  
Kierunek

Dziekan Wydziału .....

### OŚWIADCZENIE STUDENTA

Świadomy/a odpowiedzialności, która wynika z przepisów prawa powszechnie obowiązującego, oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami prawa.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w innej uczelni.

Oświadczam, że zostałem poinformowany przez UKSW o zasadach dotyczących kontroli samodzielności prac dyplomowych i zaliczeniowych.

Przyjmuję do wiadomości, że moja praca dyplomowa powstała w UKSW oraz że będzie przechowywana i przetwarzana<sup>1</sup> w celach realizowanej procedury antyplagiatowej w ogólnopolskim repozytorium pisemnych prac dyplomowych.

.....  
podpis studenta

### OŚWIADCZENIE PROMOTORA

Oświadczam, że niniejsza praca dyplomowa napisana przez Pana/Panią .....  
..... nr albumu ..... została przygotowana pod moim kierunkiem bez  
mojego współautorstwa w rozumieniu prawa własności intelektualnej.

Stwierdzam, że praca dyplomowa Pana/Pani ....., nr albumu .....  
spełnia warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Na recenzenta pracy proponuję osobę spośród następujących:

1. ....
2. ....

.....  
podpis promotora

---

<sup>1</sup>Przez przetwarzanie pracy rozumie się porównywanie przez Jednolity System Antyplagiatowy jej treści z innymi dokumentami (w celu ustalenia istnienia nieuprawnionych zapożyczeń) oraz generowanie raportu podobieństwa.

## **Słowa kluczowe**

Przestrzenie barw, przestrzeń Ohty, RGB, XYZ, sRGB, chrominancja, luminancja, cyfrowe wywoływanie zdjęć, ekspozycja, czas naświetlania, fotografia

## **Dziedzina Socrates-Erasmus**

11.3 Informatyka

## **English title**

Color spaces constructed on the basis of series of photographs with different exposures

# Spis treści

<b>Streszczenie</b> . . . . .	5
<b>1. Wprowadzenie</b> . . . . .	7
1.1. Cel pracy i charakterystyka problemu . . . . .	7
1.2. Definicje wstępne . . . . .	8
1.3. Zewnętrzne oprogramowanie i biblioteki . . . . .	10
1.4. Oprogramowanie własne . . . . .	11
1.5. Plan rozdziałów . . . . .	11
<b>2. Cyfrowe wywoływanie zdjęć</b> . . . . .	13
2.1. Parametry rejestracji światła . . . . .	13
2.2. Dekodowanie i przetwarzanie plików RAW . . . . .	14
2.3. Balans bieli i normalizacja . . . . .	15
2.4. Demozaikowanie . . . . .	15
2.4.1. Filtry Bayera . . . . .	15
2.4.2. Interpolacja brakujących składowych – demozaikowanie . . . . .	16
2.5. Konwersja obrazu do liniowej przestrzeni RGB . . . . .	17
2.5.1. Macierz przekształceń z metadanych pliku . . . . .	18
2.5.2. Obliczenia ręczne . . . . .	18
2.5.3. Przejście do liniowej przestrzeni RGB . . . . .	19
2.6. Korekcja gamma i konwersja do przestrzeni sRGB . . . . .	19
<b>3. Rejestracja serii zdjęć</b> . . . . .	21
3.1. Parametry rejestracji zdjęć . . . . .	21
3.1.1. Parametry oświetlenia . . . . .	21
3.1.2. Parametry aparatu . . . . .	22
3.2. Serie zdjęć . . . . .	22
3.2.1. Seria 1 . . . . .	23
3.2.2. Seria 2 . . . . .	24
3.2.3. Seria 3 . . . . .	24
3.2.4. Seria 4 . . . . .	25
3.2.5. Seria 5 . . . . .	25
3.2.6. Seria 6 . . . . .	26
3.2.7. Seria 7 . . . . .	26
3.2.8. Seria 8 . . . . .	27

<b>4. Tworzenie przestrzeni barw</b> . . . . .	29
4.1. Przejście z i do przestrzeni Ohty . . . . .	29
4.2. Schemat badania zmienności chrominancji . . . . .	31
4.3. Wyniki badań zmienności chrominancji . . . . .	34
4.4. Ujednolicenie i uproszczenie wyników . . . . .	38
4.5. Przestrzeń barw $R_T G_T B_T$ jako transformacja z przestrzeni RGB . . . . .	42
<b>5. Dyskusja</b> . . . . .	45
5.1. Problem braku normalizacji . . . . .	45
5.2. Normalizacja do przedziału $[0; 1]^3$ . . . . .	46
5.3. Zmiana ekspozycji zdjęć w znormalizowanej przestrzeni $R_T G_T B_T$ . . . . .	49
5.4. Obrazy HDR z adaptacją chrominancji . . . . .	52
5.5. Projekcja na konwencjonalne przestrzenie barw . . . . .	53
<b>6. Podsumowanie</b> . . . . .	55

## Streszczenie

Praca porusza temat generowania przestrzeni barwowych na podstawie serii fotografii o różnym naświetleniu, uwzględniających zmiany chrominancji, wraz ze zmniejszaniem bądź zwiększaniem poziomu ekspozycji wykonywanych fotografii. Na początku zostaje omówiony proces cyfrowego wywoływania zdjęć, ze szczególnym uwzględnieniem procesu demozaikowania, a następnie prezentowanych jest osiem serii fotografii, wykonanych różnym scenom, zawierającym obiekty o szerokim spektrum odcienia, jasności i saturacji. W procesie wykonywania zdjęć steruje się jedynie jednym parametrem – czasem naświetlania matrycy, a wszelkie inne pozostają niezmiennie, w szczególności inne ustawienia aparatu, kadr oraz światło oświetlające rejestrowaną scenę. Na podstawie zarejestrowanych serii zdjęć przeprowadzono badania zmienności chrominancji, w odniesieniu do zmiany luminancji – serie zdjęć zostały uporządkowane od najmniej doświetlonego do najbardziej prześwietlonego, po czym porównywane były kolejne pary zdjęć z serii, sprawdzając jakim zmianom ulegają parametry saturacji i odcienia koloru, w zależności od zmiany jasności odpowiadających sobie na obu obrazach pikseli. W dalszej części pracy wygenerowana przestrzeń została poddana aproksymacji szeregami Fouriera, w celu jej prostszego przedstawienia oraz procesom normalizacji, w celu poprawy zdolności arytmetycznych tej przestrzeni, a także rozpięcia jej do przedziałów standardowej kostki RGB, z zachowaniem zmian chrominancji i krzywej szarości. W ostatniej części pracy przestrzeń poddana jest praktycznemu testowi, w postaci procesu zmiany ekspozycji zdjęć oraz generowania zdjęć HDR z jednego, neutralnie doświetlonego zdjęcia, a także porównana do powszechniej stosowanych przestrzeni barw HSL, HSV oraz Ohty.

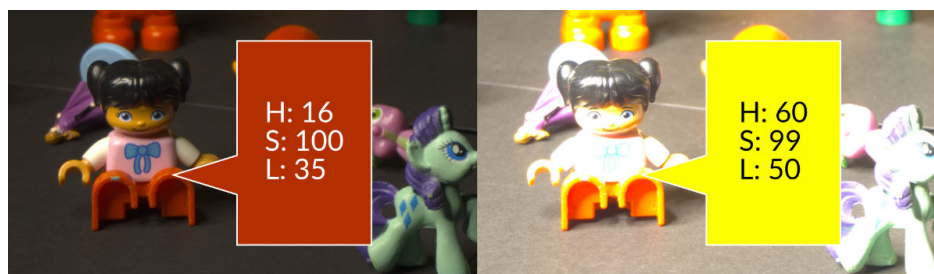


# Rozdział 1

## Wprowadzenie

### 1.1. Cel pracy i charakterystyka problemu

Większość używanych przestrzeni barw (np. sRGB, HSL, CIELab) nie bierze pod uwagę zmiany odcienia oraz saturacji barwy wraz ze zmianą poziomu naświetlenia sceny. Już wstępnie wykonane przeze mnie badania wykazały, że wraz ze zwiększaniem (bądź zmniejszaniem) poziomu naświetlania sceny saturacja, a w szczególności odcień barwy zmienia się zauważalnie. Na rysunku 1.1 zaprezentowana została zmiana chrominancji barwy czerwonej, w zależności od czasu naświetlania matrycy światłoczułej aparatu. Widać wyraźnie, że wraz ze zmianą jasności, znacząco zmieniają się inne składowe barwy. Jasnością sceny można sterować w dwojaki sposób – zmieniając natężenie sztucznie generowanego światła lub zmieniając ekspozycję poprzez zmianę czasu otwarcia migawki, średnicy otwarcia przysłony, czy poprzez korekcję ekspozycji w ustawieniach aparatu.



Rysunek 1.1: Porównanie parametrów HSL tej samej barwy, zarejestrowanej przy neutralnej (po lewej) i znacznie zwiększonej (po prawej) ekspozycji. Scena naturalna została zarejestrowana z ekspozycją 1/10s, zaś prześwietlona z ekspozycją 2s.

O ile naturalnym zjawiskiem jest, że wzrosła jasność barwy (parametr L), o tyle znacząco zmienił się też parametr odcienia barwy (H). Jeśli scenę po lewej chcielibyśmy rozjaśnić komputerowo, stosując konwencjonalne przestrzenie barw, zmieniłby się jedynie parametr jasności, parametry chrominancji zaś pozostałyby bez zmian, bądź w przypadku osiągnięcia barwy białej lub czarnej, przyjęłyby skrajne wartości.

Celem pracy jest stworzenie takiej przestrzeni barwowej, która bierze pod uwagę te zmiany, tj. wraz ze zmianą jasności barwy odpowiednio steruje zmianą saturacji oraz odcienia. W celu wygenerowania takiej przestrzeni wykonałem osiem serii zdjęć o różnym naświetleniu.



Sceny były oświetlone sztucznym światłem, pochodzącym z lampy LED, a sam aparat został osadzony na stabilnym statywie. Chodziło o to, aby wraz z wykonywaniem kolejnych zdjęć z poszczególnych serii, zmianie ulegał wyłącznie jeden parametr wpływający na poziom naświetlenia materiału światłoczułego – czas otwarcia migawki bądź średnica przysłony.

Otrzymane serie zdjęć były następnie poddane obróbce, według schematu opisanego w rozdziale drugim. Umożliwiło to wstępne przetworzenie danych i wyeksportowanie obrazów w formie umożliwiającej łatwą analizę statystyczną i generowanie przestrzeni barw. Każde zdjęcie po wywołaniu zostało wyeksportowane do formatu zawierającego wyłącznie poziomy poszczególnych pikseli. Ten proces z kolei pozwolił na umieszczenie tych wartości w przestrzeni 3D (np. RGB lub  $I_1I_2I_3$ ) oraz umożliwił obserwację zmian parametrów barw, w zależności od poziomu jasności poszczególnych kolorów.

## 1.2. Definicje wstępne

Mając ogólny pogląd na cel pracy, przejdźmy teraz do niezbędnych dla zrozumienia późniejszych opisów i schematów definicji:

- **Przestrzeń barw** – jest to matematyczna (liczbowa) reprezentacja kolorów w przestrzeni trójwymiarowej. Istnieje wiele modeli przestrzeni barwowych (np. RGB, sRGB, CMYK, LMS, CIE XYZ 1931, HSL) i mają one różnorakie zastosowania, począwszy od odzwierciedlania ludzkiego systemu widzenia, wyświetlania obrazu, druku, czy w końcu różnego rodzaju obliczeń na obrazach, np. segmentacji czy składania obrazów.
- **Przysłona** – to niewielkie urządzenie, znajdujące się w obiektywie, zbudowane z niewielkich blaszek, które zachodząc na siebie zmniejszają lub zwiększają otwór, przez który wpada światło do obiektywu [12]. Im mniejsza wartość przysłony, tym obiektyw jest jaśniejszy, przepuszczając więcej światła.
- **Migawka** – jest to urządzenie służące do odsłaniania na pewien czas materiału światłoczułego w celu jego ekspozycji na światło wpadające przez obiektyw [12].
- **EV** (z ang. **Exposure Value**) – wartość ekspozycji, czyli jednostka używana w fotografii określająca poziom naświetlenia fotografowanej sceny. Wartość 0 odpowiada takiemu natężeniu światła, które powoduje prawidłowe naświetlenie matrycy światłoczułej przy ISO równym 100, podczas ekspozycji trwającej jedną sekundę i przysłonie równej  $f/1$  [9]. Skala ta jest logarytmiczna i może być ujemna, jeśli scena jest niedoświetlona. Najczęściej w dobrej klasie aparatach możemy sterować zmianą ekspozycji o  $\pm\frac{1}{3}EV$ . Zmianę wartości ekspozycji wykonuje się zazwyczaj poprzez modyfikację średnicy przysłony bądź czasu otwarcia migawki. Przeskok do sąsiedniej jednostki otwarcia migawki bądź średnicy przysłony odpowiada właśnie zmianie ekspozycji o  $\pm\frac{1}{3}EV$ .
- **RAW** (z ang. raw – surowy) – to nieustandaryzowany format danych, który może mieć dowolną budowę oraz przechowywać (skompresowane lub nie) dane w dowolnej postaci. Ogólnie mówiąc RAW nie jest formatem samym w sobie, jest to jedynie nazwa opisująca typy plików, w których zapisane są nieprzetworzone dane, pozyskane np. przez naświetlanie materiału światłoczułego (matrycy) aparatu.
- **EXIF (Exchangeable Image File Format)** – to format plików stworzony przez **Japan Electronic Industries Development Association**, za pomocą którego zapisuje się metadane w plikach audio, wideo i zdjęciowych. Format ten używany jest

powszechnie w kamerach cyfrowych, smartfonach oraz skanerach. Zawiera podstawowe informacje o pliku, takie jak data i godzina wykonania zdjęcia, rodzaj zastosowanej kompresji, czy rozdzielczość obrazu, jak również metadane specyficzne dla wykonanego zdjęcia czy urządzenia, takie jak np. wartość przysłony, czas otwarcia migawki, przestrzeń barw oraz wiele informacji niezbędnych przy odczytywaniu i obróbce formatu RAW. Format ten czerpie z systemu tagów definiowanych w specyfikacji TIFF [25], jednak metadane mogą być również umieszczane w innych formatach plików, np. RIFF, JPEG oraz WAV.

Aparat użyty w celu wykonania serii zdjęć to **Olympus OM-D E-M10 Mark II**. Jest to średniej klasy bezlusterkowiec, umożliwiający pełną kontrolę nad wszelkimi parametrami wykonywanego zdjęcia, co jest niezbędne do zarejestrowania poprawnych serii zdjęć. W tabeli 1.1 została przedstawiona część specyfikacji tego aparatu, w celu nakreślenia z jakim sprzętem mamy do czynienia. Widoczna specyfikacja została ograniczona do niezbędnego minimum, pełną można znaleźć na stronie producenta [24].



Rysunek 1.2: Aparat użyty w pracy – Olympus OM-D E-M10 Mark II wraz z obiektywem M.Zuiko Digital ED 14-42mm F3.5-5.6 EZ

Tabela 1.1: Fragment specyfikacji aparatu Olympus OM-D E-M10 Mark II oraz obiektywu M.Zuiko Digital ED 14-42mm F3.5-5.6 EZ

Wymiary sensora	17.3 mm (szer.) x 13.0 mm (wys.)
Typ sensora	4/3 Live MOS Sensor
Głębina bitowa formatu RAW	12 bitów
Rozdzielczość sensora	4640 x 3472 pikseli
Schemat filtra Bayera sensora	$\begin{bmatrix} R & G \\ G & B \end{bmatrix}$
Ustawienia ekspozycji	$\pm 5EV$ ze skokiem $\pm \frac{1}{3}EV$
Czas otwarcia migawki	Od 60 s do 1/4000 s
Zastosowany obiektyw	M.Zuiko Digital ED 14-42mm F3.5-5.6 EZ
Przedział ogniskowej	14 – 42mm
Przedział najjaśniejszej przysłony	f/3.5 - f/5.6

Wraz z aparatem został użyty dobrej klasy statyw, umożliwiający stabilne umiejscowienie aparatu oraz lampa oświetleniowa wykonana w technologii LED, która oświetla scenę

światłem ciągłym, nie powodując migotania podczas wykonywania zdjęć, niezależnie od zastosowanej wartości przysłony oraz czasu otwarcia migawki.

### 1.3. Zewnętrzne oprogramowanie i biblioteki

W swojej pracy korzystam z kilku otwartoźródłowych programów i bibliotek zewnętrznych.

- **dcraw** – to program oraz biblioteka w języku C, stworzona przez Dave’a Coffina pozwalająca na konwersję surowych danych z aparatów na formaty PCM oraz TIFF, które z racji ich popularności są łatwiejsze do odczytu, niż surowe dane [18]. Biblioteka definiuje algorytmy dekodowania oraz dekompresji dla większości współcześnie używanych aparatów cyfrowych, pozwalając nie tylko wywoływać zdjęcia, ale również dekodować skompresowane surowe dane do postaci prawdziwie surowej, oraz sterować wieloma parametrami przy wywoływaniu zdjęć, takimi jak współczynnik krzywej gamma, wyjściowa przestrzeń barw, algorytm demozaikowania i wiele innych.
- **VTK (Visualization Toolkit)** – jest wieloplatformową biblioteką opensource służącą do generowania grafiki 3D, przetwarzania obrazów oraz wizualizacji danych przeznaczoną dla programów pisanych w języku C++, lecz posiadającą także swoją uboższą w możliwości implementację dla języków Tcl/Tk, Java oraz Python [16]. Jednym z jej najczęstszych zastosowań jest wszelkiego rodzaju wizualizacja danych, zarówno tych dwu-, jak i trójwymiarowych.
- **Qt** – jest to biblioteka służąca do tworzenia szeroko rozumianych Graficznych Interfejsów Użytkownika [15]. W połączeniu z biblioteką VTK, umożliwiła mi wizualizację stworzonych przestrzeni barwowych, a także wygodne porównywanie ich parametrów.
- **OpenCV** – to otwartoźródłowa biblioteka napisana dla języka C++, Python oraz JavaScript, umożliwiająca obróbkę obrazów w czasie rzeczywistym [14]. Pozwala ona w łatwy sposób reprezentować dowolne pliki obrazowe, konwertując je do prostej macierzy, na której wykonuje się następnie operacje obrazowe. Macierz taka ma liczbę wierszy i kolumn zależną od rozdzielczości wczytywanego obrazu, a elementami macierzy są liczby całkowite bądź zmiennopozycyjne (w przypadku obrazów monochromatycznych) lub trzejelementowe wektory tych liczb (w przypadku obrazów RGB). Warto zwrócić uwagę, że OpenCV zapisuje obrazy barwowe w postaci BGR. Należy o tym pamiętać w szczególności podczas mnożenia wartości pikseli przez różnorakie macierze.
- **Adobe DNG Converter** – to program autorstwa firmy Adobe, konwertujący dowolny format RAW na format **DNG** (również autorstwa Adobe) [17]. Format ten przetłumaczyć można jako „**cyfrowy negatyw**”. Powstał on w celu standaryzacji formatów RAW, aby zminimalizować konieczność przetwarzania niezliczonej liczby różnych surowych formatów danych obrazowych oraz uprościć wymianę plików RAW, niezależnie od producenta aparatu.
- **ExifTool** – to program autorstwa Phila Harvey’a umożliwiający wyświetlanie wartości zapisanych w pliku tagów EXIF w konsoli [19]. Przykładowe działanie programu zaprezentowano na rysunku 1.3.
- **gnuplot** – konsolowy, otwartoźródłowy program służący do generowania dwu- i trójwymiarowych wykresów danych [20].

```
mario -- zsh -- 80x24
File Size           : 13 MB
File Modification Date/Time : 2020:05:06 01:47:06+02:00
File Access Date/Time   : 2020:05:12 20:03:50+02:00
File Inode Change Date/Time : 2020:05:06 01:56:06+02:00
File Permissions       : rwxrwxrwx
File Type             : ORF
File Type Extension    : orf
MIME Type             : image/x-olympus-orf
Exif Byte Order        : Little-endian (Intel, II)
Image Width           : 4640
Image Height          : 3472
Bits Per Sample        : 16
Compression           : Uncompressed
Photometric Interpretation : BlackIsZero
Image Description       : OLYMPUS DIGITAL CAMERA
Make                  : OLYMPUS CORPORATION
Camera Model Name      : E-M10MarkII
Strip Offsets         : 1485312
Orientation           : Horizontal (normal)
Samples Per Pixel      : 1
Rows Per Strip         : 3472
Strip Byte Counts      : 12131169
X Resolution           : 350
Y Resolution           : 350
```

Rysunek 1.3: Przykładowe wyjście programu ExifTool.

## 1.4. Oprogramowanie własne

Do obróbki i generowania przestrzeni barwowej napisane zostały dwa programy w języku C++14 i jeden w języku Python:

- Pierwszy z nich został napisany przy użyciu biblioteki OpenCV. Jego głównym zadaniem było **cyfrowe wywoływanie zdjęć** i wyeksportowanie ich do różnych formatów, gdzie jedne z nich służą do obliczeń w liniowej przestrzeni RGB (16-bitowy TIFF), inne do prezentowania wywołanych zdjęć w przestrzeni sRGB. Dokładny opis działania programu jak i procesu wywoływania zdjęć znajduje się w rozdziale drugim.
- Drugi zaś oparty jest o bibliotekę VTK i Qt. Jego zadaniem była wizualizacja danych, generowanie przestrzeni barw oraz reprezentacja wyników w przestrzeni 3D, a także eksport danych 2D do gnuplota. Dokładniejszy schemat jego działania, jak i sam proces generowania przestrzeni barw w oparciu o wykonane serie zdjęć został opisany w rozdziale czwartym.
- W Pythonie został napisany prosty program konsolowy służący do manipulacji, mapowania, filtrowania, redukcji i przetwarzania danych, które były następnie wykorzystywane w programie gnuplot do wizualizacji wykresów.

## 1.5. Plan rozdziałów

Po wprowadzeniu niezbędnych definicji oraz charakterystyki używanego sprzętu i oprogramowania, możemy przejść do dalszych rozdziałów pracy. W rozdziale drugim został szczegółowo omówiony proces cyfrowego wywoływania zdjęć. Jak zostało pokazane, od momentu naciśnięcia spustu migawki i zapisania danych w pamięci aparatu do momentu wyświetlenia gotowego zdjęcia na ekranie komputera czy aparatu trzeba przejść przez dosyć skomplikowany i kosztowny obliczeniowo proces wywoływania zdjęć. Proces ten rozpoczyna się od pozyskania surowych danych nieskompresowanych, następnie omówione są filtry Bayera, proces demosaikowania, balansu bieli, aż w końcu odpowiednie przekształcenia na przestrzeniach barw

w celu uzyskania barwowego zdjęcia w liniowej przestrzeni RGB oraz przestrzeni sRGB po korekcji gamma. Wraz z końcem rozdziału drugiego mamy gotowy algorytm wywoływania zdjęć, który był zastosowany w celu wywołania wszystkich serii zdjęć, zaprezentowanych w rozdziale trzecim.

Kolejny rozdział traktuje o rejestrowaniu serii fotografii o różnym naświetleniu. Opisuje on zarówno parametry aparatu, jak i rejestrowanych zdjęć, otoczenia i oświetlenia, niezbędne do uzyskania prawidłowych serii fotografii. Zostały tam także zaprezentowane uzyskane serie zdjęć, wraz z opisem zastosowanych ustawień aparatu.

Wraz z końcem rozdziału trzeciego byłem już wyposażony w materiał badawczy, przy pomocy którego w kolejnym – czwartym – rozdziale wygenerowane zostały przestrzenie barwowe. W rozdziale tym opisana jest nieskorelowana przestrzeń Ohty, na podstawie której była generowana przestrzeń adaptująca zmiany chrominancji do zmian luminancji. Pierwszym krokiem generowania przestrzeni było porównywanie par zdjęć dla każdej serii, poczynając od zdjęć najmniej naświetlonych, poprzez zdjęcia neutralne, aż do całkowicie prześwietlonych. Porównywane były sąsiednie zdjęcia z serii, np. najmniej naświetlone z jego nieco mniej niedoświetlonym sąsiadem itd. Na podstawie zmian chrominancji w zależności od zmian luminancji w badanych parach zdjęć były tworzone listy zmian chrominancji dla każdego poziomu jasności. Na koniec badania każdej z serii listy te zostały zsumowane, a ich wartość uśredniona, aby uzyskać średnie zmiany chrominancji dla danego poziomu jasności. Wyniki uzyskane w ośmiu seriach zostały w następnym kroku uśrednione, a otrzymane funkcje, reprezentujące zmiany chrominancji na osiach  $I_2$  i  $I_3$  przestrzeni Ohty scałkowane w celu uzyskania funkcji przekształcających przestrzeń Ohty w przestrzeń adaptującą obliczone zmiany chrominancji. Następnym krokiem była definicja analogicznego przekształcenia dla standardowej przestrzeni RGB, w celu uniknięcia potrzeby korzystania z przestrzeni Ohty. Uzyskane funkcje adaptacji chrominancji były ostatecznie aproksymowane szeregami Fouriera, w celu zmniejszenia szumów na wykresach funkcji, a także łatwiejszego przenoszenia wygenerowanej przestrzeni barw.

Przedostatni rozdział został poświęcony dyskusji, jednak jego początkowa część jest niejako kontynuacją rozdziału czwartego. Poruszony został problem braku normalizacji przestrzeni, przez który wykonywanie operacji arytmetycznych dawało czasem błędne wyniki, w szczególności w dziedzinie liczb ujemnych. Problem ten został rozwiązany poprzez odpowiednie skalowanie, translację oraz zdefiniowanie funkcji dystrybucji kolorów wewnątrz standardowej kostki RGB, przy zachowaniu krzywej szarości oraz zmian chrominancji. Wygenerowana przestrzeń barw została poddana praktycznej analizie na podstawie zmiany ekspozycji zdjęć, poruszony został także temat obrazów HDR, a w szczególności uzyskiwania obrazów z jednego, neutralnie oświetlonego zdjęcia, z zachowaniem zmian chrominancji wynikającej z otrzymanej przestrzeni barw. Pod koniec rozdziału piątego otrzymana przestrzeń barw była poddana projekcji na inne przestrzenie barw, w celu porównania.

Ostatni rozdział pracy został poświęcony krótkiemu podsumowaniu. Prześladowany jest jeszcze raz cały proces badawczy, począwszy od rejestracji zdjęć, poprzez ich wstępną, cyfrową obróbkę, dochodząc w końcu do punktu kluczowego badań – generowania przestrzeni barwowych, a kończąc na zestawieniu cech, wad i zalet wygenerowanej przestrzeni barw oraz usuwaniu niektórych z jej wad poprzez procesy normalizacji i aproksymacji. Podsumowano także użyte i zaprojektowane oprogramowanie na potrzeby pracy.

## Rozdział 2

# Cyfrowe wywoływanie zdjęć

Pierwszym procesem jaki trzeba wykonać przed rejestracją i analizą serii zdjęć jest omówienie procesu ich wywoływania. Cyfrowe wywoływanie zdjęć to szereg procesów korzystających z wielu algorytmów i technik, mających na celu przetworzenie surowych danych, pozyskanych przez naświetlanie matrycy światłoczułej aparatu i zapisanych następnie w postaci pliku RAW w pamięci aparatu, do postaci umożliwiającej wyświetlenie i obróbkę fotografii. O ile termin „wywoływanie zdjęć” kojarzy nam się z fotografią analogową, ma on również swoje zastosowanie w fotografii cyfrowej. Proces prowadzący od rejestracji zdjęcia, do finalnej, wywołanej (przetworzonej) postaci jest dosyć skomplikowany i podzielony na wiele etapów. W tym rozdziale omówione zostały etapy niezbędne do prawidłowego wywoływania fotografii zapisanej w formacie RAW, zaczynając od przetwarzania plików RAW, a kończąc na konwersji uzyskanego obrazu do standardowej, wspieranej przez większość współczesnych urządzeń przestrzeni sRGB. Pominiętych zostało kilka końcowych kroków używanych przez algorytmy wywołujące zdjęcia aparatów, czy popularne programy graficzne, takie jak. np. odszumianie, korekcja krzywej bazowej czy usuwanie zniekształceń związanych z krzywizną obiektywu. Interesują nas bowiem jedynie te procesy, które z surowych danych w postaci siatki Bayera uzyskują dane w postaci RGB (zarówno tej liniowej, jak i z korekcją gamma). Proces wywoływania, niektóre parametry, a także ich wartości zostały zainspirowane artykułami [6, 7, 26].

### 2.1. Parametry rejestracji światła

Każda kamera cyfrowa ma swoją specyfikę pracy, budowę oraz stałe parametry rejestracji natężenia światła. Najważniejszym z nich jest dokładność rejestracji sensora i dokładność zapisu danych. W przypadku omawianego aparatu firmy Olympus mamy do czynienia z 12-bitową głębią zapisu danych, zatem teoretycznie możemy zapisać  $2^{12} = 4096$  odcieni jednej ze składowych RGB.

Jednak poza tą wartością istnieją jeszcze dwa znaczące parametry – poziom czerni i bieli. Poziom czerni to minimalna wartość odczytu sensora światłoczułego, poniżej której dane uznawane są za niepoprawne. Analogicznie jest w wypadku poziomu bieli, jest to maksymalny poziom natężenia światła, jaki może zarejestrować sensor. Zazwyczaj wartości te są znacznie oddalone od wartości maksymalnych i minimalnych. Tak w przypadku mojego aparatu poziom czerni wynosi 254, zaś poziom bieli 4000. Sensor nie rejestruje poniżej i powyżej tych wartości. Im większa głębia bitowa rejestrowanego obrazu, tym bardziej te wartości oddalają się od wartości 0 i  $2^n - 1$ , gdzie  $n$  to liczba bitów, na których zapisuje się natężenie światła w danym

pikselu surowego obrazu.

## 2.2. Dekodowanie i przetwarzanie plików RAW

Pliki RAW (ang. raw – surowe), jak sama nazwa wskazuje zawierają surowe dane, zarejestrowane poprzez odczyt natężenia światła przez komórki matrycy światłoczułej. W najprostszej formie plik RAW można traktować jako cyfrowy negatyw, który zawiera w sobie dwuwymiarową tablicę dyskretnych wartości pochodzących z rejestracji światła przez matrycę światłoczułą, przetworzonych przez elektronikę aparatu (w szczególności konwerter analogowo-cyfrowy) do postaci dyskretnej i zdigitalizowanej, którą można zapisać w postaci pliku cyfrowego. Można by zatem pomyśleć, że wystarczy znać budowę pliku RAW używanego przez danego producenta, a będziemy w stanie odczytać składowe RGB dla każdego piksela obrazu. Niestety nie jest to takie trywialne.

Przede wszystkim RAW nie jest żadnym ustandaryzowanym formatem plików. Pod tą nazwą może się kryć dowolny format pliku, opisany, bądź nie, przez producenta, przechowujący dane w dowolny sposób, a nawet używający bezstratnej kompresji. W moim przypadku mam do czynienia z plikiem RAW firmy Olympus, który nosi rozszerzenie ORF. Po głębszej analizie jego budowy w edytorze bajtowym i przejrzeniu anglojęzycznych for internetowych fascynatów fotografii musiałem niestety poszukać innej drogi, niż próba rozszyfrowania tego pliku samodzielnie. Niestety firma Olympus nie stworzyła żadnej dokumentacji dla swojego formatu, więc musiałem skorzystać z pomocy zewnętrznych open source'owych narzędzi. Przy pomocy programu dcraw pozbywam się kompresji danych, eksportując surowe dane do postaci prostego, monochromatycznego, 16-bitowego pliku TIFF.

Dla ścisłości warto podać dokładną komendę programu dcraw, którą można przekonwertować plik RAW na 16-bitowy plik TIFF bądź PPM, bez jego przetwarzania poza dekompresją danych, zachowując w pełni surową strukturę obrazu [1]:

```
dcraw -4 -T nazwa_pliku.orf
```

gdzie T – eksportuj do TIFF, 4 – eksportuj w głębi 16-bitowej.

Na tym etapie mamy już wyeksportowany plik do postaci 16-bitowego formatu TIFF (bądź PPM), dcraw nie zmienił rzecz jasna wartości pikseli zapisanych w pliku. Te pozostały dokładnie takie, jakie były zapisane w momencie konwersji analogowo-cyfrowej przez elektronikę aparatu.

Mając już plik TIFF w czytelnej postaci, mogłem przejść do dalszych procesów wywoływania zdjęć. Jednak nawet mając w pełni surowe dane, nie byłem jeszcze w stanie odczytać z nich wartości RGB dla każdego piksela obrazu. Problem polega na tym, że większość popularnych matryc nie rejestruje w jednym punkcie obrazu 3 odcieni RGB, a jedynie jeden z nich – wyjątkiem są np. sensory X3 firmy Foveon opisane w [23]. Aby dla każdego piksela obrazu uzyskać wszystkie składowe RGB, należało surowe dane poddać procesowi zwanemu demozajkowaniem, opisanemu w podrozdziale 2.4.

## 2.3. Balans bieli i normalizacja

Przed przystąpieniem do demozaikowania i dalszych etapów wywoływania, uzyskany wcześniej monochromatyczny obraz trzeba poddać balansowi bieli i normalizacji do przedziału  $[0; 1]$ . W pierwszej kolejności wykonuje się proces normalizacji, który zależy od opisanych wcześniej poziomów czerni  $l_{black}$  i bieli  $l_{white}$ . Mając te parametry, każdy piksel surowego obrazu przetwarzamy następująco. Aktualnie pracujemy na obrazie w poziomach szarości, więc jasność danego piksela oznaczmy jako  $Y$ :

$$Y_{norm} = (Y - l_{black}) / (l_{white} - l_{black}) \quad (2.1)$$

Konieczność zbalansowania bieli wynika z dwóch czynników. Po pierwsze zostało zarejestrowanych dwukrotnie więcej punktów o odcieniu zielonym, aniżeli punktów o innych odcieniach. Należy zniwelować tę różnicę, tak, aby zrównoważyć wszystkie składowe RGB. Po drugie, trzeba wziąć pod uwagę ustawienia balansu bieli stosowane podczas wykonywania zdjęć. W zależności od zastosowanego balansu bieli, należy zastosować inne współczynniki  $W_r$ ,  $W_g$  i  $W_b$ , takie, aby obraz po zbalansowaniu bieli był jak najbliższy rzeczywistej scenie, w zależności od stosowanej temperatury barwowej światła. Balans bieli możemy wykonać albo przed demozaikowaniem, mnożąc wartość piksela przez odpowiednią wagę, albo tuż po demozaikowaniu, mnożąc piksele barwowego już wtedy obrazu przez macierz.

Wykonując balans bieli przed demozaikowaniem, mnożymy piksel przez jedną z wag –  $W_r$ ,  $W_g$  lub  $W_b$ . To, którą z wag zastosować wiadomo po indeksach  $(i, j)$  danego piksela oraz stosowanego przez aparat schematu siatki Bayera. Współrzędne (dla spójności z kodem programu) liczymy od 0. W przypadku – jak również w moim aparacie – schematu RGGB, składową czerwoną mamy gdy  $i$  oraz  $j$  są parzyste, niebieską gdy  $i$  oraz  $j$  są nieparzyste, zieloną w pozostałych przypadkach:

$$\begin{cases} Y_{bal} = Y_{norm} \cdot W_r, & \text{gdy } i \text{ oraz } j \text{ są parzyste} \\ Y_{bal} = Y_{norm} \cdot W_b, & \text{gdy } i \text{ oraz } j \text{ są nieparzyste} \\ Y_{bal} = Y_{norm} \cdot W_g, & \text{w p. p.} \end{cases} \quad (2.2)$$

Jeśli zaś wykonujemy balans bieli po demozaikowaniu, mamy do czynienia z obrazem barwowym, wtedy każdy piksel obrazu wystarczy pomnożyć następująco:

$$\begin{bmatrix} R_{bal} \\ G_{bal} \\ B_{bal} \end{bmatrix} = \begin{bmatrix} W_r & 0 & 0 \\ 0 & W_g & 0 \\ 0 & 0 & W_b \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.3)$$

Niezależnie czy wykonujemy balans bieli przed, czy po demozaikowaniu, należy sprawdzić czy żaden z poziomów jasności ( $Y$ ,  $R$ ,  $G$  lub  $B$ ) nie jest większy niż 1. Jeśli jest, należy po prostu obciąć tę wartość do 1.

W moim wypadku balans bieli został wykonany przed demozaikowaniem, zatem jeszcze w tym momencie mamy do czynienia obrazem monochromatycznym.

## 2.4. Demozaikowanie

### 2.4.1. Filtry Bayera

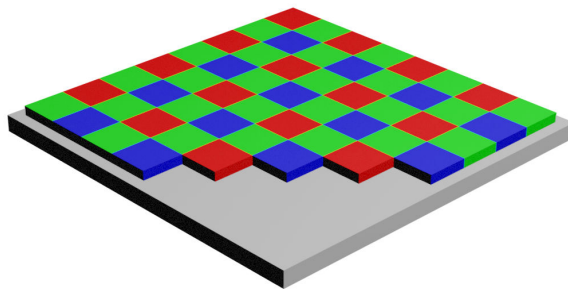
Przed zdefiniowaniem czym jest demozaikowanie, przyjrzyjmy się jak w rzeczywistości „widzi” kamera. W opisie tym pominąłem wpływ obiektywu na proces rejestracji zdjęć, jako że



interesuje nas wyłącznie sposób w jaki matryca światłoczuła odbiera i rejestruje natężenie światła.

Matryce najczęściej zbudowane są z tzw. Color Filter Array (CFA) [2], a najpopularniejszą z nich jest Filtr Bayera. Matryca oparta o filtr Bayera składa się najczęściej z sensorów odbierających światło czerwone, niebieskie i zielone, jednak istnieje też wiele odmian, które odbierają dodatkowo barwę cyjanową, bądź całkowicie opierają się na przestrzeni CMY. Zarówno w przypadku większości aparatów, jak i opisywanego tu urządzenia firmy Olympus, mamy do czynienia z najbardziej rozpowszechnioną siatką Bayera w postaci RGGB, która rejestruje światło czerwone, niebieskie i dwukrotnie światło zielone. Podwójna rejestracja światła zielonego wynika z faktu, że ludzki system widzenia odbiera takiego światła najwięcej. Autorzy w [13] opisują kilka powszechnie znanych wzorów na proporcję odbieranych przez ludzki wzrok barw, jednak najczęściej spotykanym z nich jest:

$$Y = 0.299R + 0.587G + 0.114B, \text{ gdzie } Y \text{ to luminancja.}$$



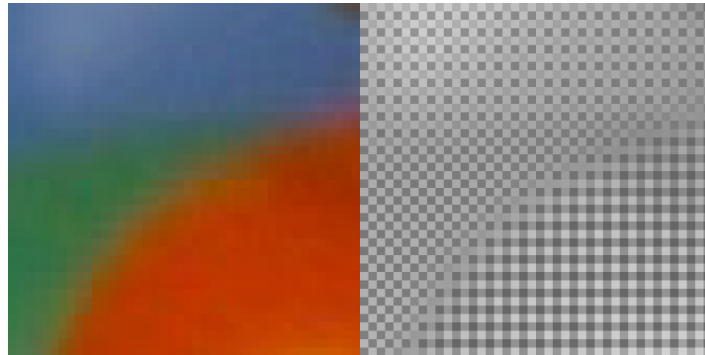
Rysunek 2.1: Przykładowa siatka filtru Bayera o schemacie RGGB.

Jedna komórka w siatce Bayera odpowiada jednemu pikselowi na surowym obrazie **monochromatycznym**. Jak zatem widać, w każdym pikselu obrazu brakuje dwóch składowych. O tym w jaki sposób interpolować brakujące składowe traktuje kolejny podrozdział.

#### 2.4.2. Interpolacja brakujących składowych – demozaikowanie

Zatem skoro filtr Bayera w jednej, najmniejszej komórce rejestruje jedynie jedną składową koloru, należy odzyskać utracone informacje – w komórkach G musimy odzyskać składowe R i B, w komórkach R składowe G i B, a w komórkach B składowe R i G. Proces takiego odzyskiwania brakujących składowych dla każdego piksela obrazu nazywa się demozaikowaniem. Demozaikowanie polega na stosowaniu mniej lub bardziej skomplikowanych algorytmów interpolowania brakujących składowych. Istnieje wiele algorytmów demozaikowania [28], począwszy od najprostszych, które interpolują wyłącznie na podstawie wartości najbliższych sąsiadów, aż po bardziej skomplikowane, jednak kosztowne obliczeniowo algorytmy Pixel Grouping (PPG), Aliasing Minimization and Zipper Elimination (AMaZE) czy zaimplementowany i zastosowany przez mnie algorytm autorstwa King-Hong Chung’a i Yuk-Hee Chan’a opisany w [3], korzystający z wariacji różnicy kolorów. Algorytm ten interpolując brakujące składowe danego piksela korzysta z okna 9x9, obliczając wstępnie wariację kolorów w pionie, poziomie i na przekątnej, interpoluje składowe w zależności w jakim kierunku nastąpiła najmniejsza zmiana w kolorze. Co prawda algorytm ten nie należy do najszybszych obliczeniowo,

jednak zdecydowałem się na niego, ponieważ daje świetne rezultaty, jednocześnie minimalizując artefakty powstające w procesie demozaikowania.



Rysunek 2.2: Fragment obrazu po demozaikowaniu, w porównaniu do obrazu surowego.

Warto zaznaczyć, że proces demozaikowania zmniejsza nieznacznie rozdzielczość obrazu. Fakt ten wynika z tego, że demozaikowanie przebiega w pewnej odległości od krańców obrazu, ponieważ piksele bliskie krańcom nie posiadają odpowiedniej liczby sąsiadów w celu interpolacji wartości. W przypadku algorytmu użytego w pracy rozdzielczość zmniejsza się o 16 pikseli w pionie i poziomie – ucinanych jest po 8 pikseli z każdego krańca obrazu.

## 2.5. Konwersja obrazu do liniowej przestrzeni RGB

Na tym etapie mamy już do czynienia z barwowym obrazem po procesie demozaikowania. Niestety kolory, a w szczególności poziomy jasności wyświetlanych zdjęć, odbiegają dalece od naszych oczekiwań. Wynika to z faktu, że obraz zapisany jest w przestrzeni barw aparatu, a nie konkretnej przestrzeni, np. sRGB. Następnym krokiem było takie przekształcenie wartości pikseli, aby uzyskać liniową przestrzeń RGB (w celach obliczeniowych), a następnie skorygować tę przestrzeń korekcją gamma, w celach prezentacji. Jeśli chodzi o konwersję do przestrzeni RGB istnieją dwie drogi, omówione w poniższych podrozdziałach.



Rysunek 2.3: Porównanie obrazu zaraz po procesie demozaikowania (strona lewa) do końcowego obrazu w przestrzeni sRGB (strona prawa).

### 2.5.1. Macierz przekształceń z metadanych pliku

Omówię najpierw najprostszą z dróg. Jest ona najwygodniejsza, jednak wymaga tego, aby w metadanych surowego pliku z obrazem znajdowała się specjalna macierz przekształceń, która przekonwertuje wartości pikseli z przestrzeni barw urządzenia, bezpośrednio do liniowej przestrzeni RGB. To, czy plik posiada taką macierz, możemy sprawdzić przy pomocy programu ExifTool. W moim przypadku macierz ta prezentuje się następująco:

$$\begin{bmatrix} 1.671875 & -0.5625 & -0.109375 \\ -0.265625 & 1.46875 & -0.203125 \\ 0.0625 & -0.5625 & 1.5 \end{bmatrix}$$

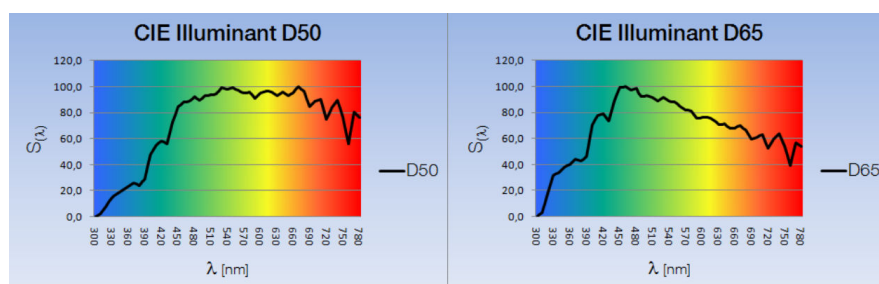
Jeśli pomnożymy każdy piksel obrazu przez powyższą macierz uzyskamy obraz w liniowej przestrzeni RGB:

$$\begin{bmatrix} R_{lin} \\ G_{lin} \\ B_{lin} \end{bmatrix} = \begin{bmatrix} 1.671875 & -0.5625 & -0.109375 \\ -0.265625 & 1.46875 & -0.203125 \\ 0.0625 & -0.5625 & 1.5 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.4)$$

W tym etapie otrzymaliśmy obraz gotowy do obliczeń, który możemy również przekonwertować do przestrzeni sRGB, stosując korekcję gamma.

### 2.5.2. Obliczenia ręczne

Jeśli jednak nie posiadamy opisanej powyżej macierzy, musimy obliczyć ją sami. Pierwszym krokiem jest uzyskanie macierzy adaptacji chromatycznej<sup>1</sup>, która przekonwertuje obraz z przestrzeni barw zależnej od urządzenia, do przestrzeni CIE XYZ D50. Poprzez D50 rozumie się tutaj standardowy iluminant o temperaturze barwowej 5003 K. Standardowy iluminant CIE [10] to teoretyczny model źródła światła, stworzony przez CIE (Commission Internationale de l'Eclairage), w celu standaryzacji zasad kolorimetrii i porównywania kolorów zarejestrowanych w oświetleniu o różnych temperaturach barwowych. Zarówno modele D50, jak i D65 reprezentują światło dzienne, jednak model D65 jest modelem nowszym i powszechniej stosowanym, a jego temperatura barwowa jest chłodniejsza (6504 K) i reprezentuje światło dzienne w południe, na obszarach północnych. Standardowy iluminant można też rozumieć jako rozkład poszczególnych długości fal, składających się na światło białe.



Rysunek 2.4: Porównanie iluminantów D50 i D65.<sup>2</sup>

<sup>1</sup>Angielska nazwa tej macierzy to **forward matrix**.

<sup>2</sup>[https://www5.konicaminolta.eu/fileadmin/content/eu/Measuring\\_Instruments/4\\_Learning\\_Centre/L\\_D/Light\\_sources\\_and\\_illuminants/Apps\\_Note\\_1\\_-\\_Light\\_sources\\_and\\_illuminants.pdf](https://www5.konicaminolta.eu/fileadmin/content/eu/Measuring_Instruments/4_Learning_Centre/L_D/Light_sources_and_illuminants/Apps_Note_1_-_Light_sources_and_illuminants.pdf)

Skąd zatem pozyskać tę macierz? Możemy albo obliczyć ją ręcznie, korzystając np. z instrukcji opisanych w [4] lub skorzystać z programu Adobe DNG Converter, który podczas konwersji z pliku ORF (lub innego) zapisze nam taką macierz w metadanych pliku. Jeśli pomnożymy każdy piksel (po procesie demozajkowania i balansu bieli) przez obliczoną macierz adaptacji chromatycznej, otrzymamy obraz w przestrzeni barw CIE XYZ D50.

W oparciu o listę macierzy przejść z [22] można wybrać najbardziej odpowiednią dla swoich zdjęć oraz wykonać dodatkowe konwersje, np. przejść z przestrzeni D50 do D65, uzyskując chłodniejszą temperaturę barwową wynikowego zdjęcia. W moim przypadku testowałem oba rozwiązania (z i bez przejścia do D65), jednak po przejściu na iluminant D65, uzyskiwałem zbyt chłodne barwy.

W ten sposób uzyskałem obraz w przestrzeni XYZ, którą w następnym kroku zamieniłem na powszechniej stosowaną w wyświetlaczach przestrzeń RGB.

### 2.5.3. Przejście do liniowej przestrzeni RGB

W zależności od wyjściowej przestrzeni XYZ, a dokładniej mówiąc standardowego iluminantu, musimy zastosować odpowiednią macierz przejścia. W przypadku moich zdjęć (przestrzeń barw XYZ D50) idealnie spisywała się następująca macierz:

$$\begin{bmatrix} R_{lin} \\ G_{lin} \\ B_{lin} \end{bmatrix} = \begin{bmatrix} 2.0413690 & -0.5649464 & -0.3446944 \\ -0.9692660 & 1.8760108 & 0.0415560 \\ 0.0134474 & -0.1183897 & 1.0154096 \end{bmatrix} \begin{bmatrix} X_{D50} \\ Y_{D50} \\ Z_{D50} \end{bmatrix} \quad (2.5)$$

Na tym etapie kończy się proces wywoływania zdjęć, w celach tworzenia przestrzeni barw. Obliczenia opisane w następnych rozdziałach były wykonywane na obrazach w liniowej przestrzeni RGB, aby uzyskać jak najprostszą, najszerszą i najmniej przekształconą przestrzeń początkową. Dodatkowym procesem, jaki kończy cały proces cyfrowego wywoływania zdjęć, opisany w niniejszej pracy, jest przejście z liniowej przestrzeni RGB do przestrzeni sRGB.

## 2.6. Korekcja gamma i konwersja do przestrzeni sRGB

Ten krok służy jedynie w celu uzyskania obrazów, które będzie można poprawnie wyświetlać na monitorach i w niniejszym dokumencie. Jak wiadomo ekrany korzystają nie z liniowej przestrzeni RGB, lecz z przestrzeni sRGB z korekcją gamma. Fakt ten wynika ze sposobu, w jaki ludzie postrzegają zmiany w luminancji. Autorzy w [21] szeroko opisują problem korekcji gamma w różnych dziedzinach elektroniki, wynikający z faktu, że ludzie chętniej postrzegają zmiany jasności ciemnych barw, aniżeli tych jaśniejszych. Biologicznie ma to oczywiście swoje uzasadnienie – dzięki nieliniowości naszego wzroku lepiej rozróżniamy kształty nocą, kiedy światła jest mało. Jednak od strony technicznej niesie to za sobą konieczność konwersji przestrzeni liniowej na nieliniową. W tym celu na przestrzeni RGB wykonuje się korekcję gamma, uzyskując przestrzeń sRGB, stosowaną powszechnie przez wyświetlacze. Dzięki poniższej konwersji możemy poprawnie wyświetlić wywołane zdjęcia (pamiętajmy, że pracujemy na przedziale  $[0; 1]$ ):

$$\gamma(u) = \begin{cases} 12.92u, & \text{jeśli } u \leq 0.0031308 \\ 1.055u^{\frac{1}{2.4}} - 0.55, & \text{w p. p.} \end{cases} \quad (2.6)$$

Poprzez u rozumie się tutaj jasność piksela jednego z kanałów RGB, bądź po prostu jasność danego piksela, w przypadku obrazów monochromatycznych. Jeżeli wartość  $\gamma(u)$  wyjdzie poza zakres  $[0; 1]$ , należy przyciąć uzyskany wynik do 0 lub 1.

Do tego momentu pracowaliśmy na funkcjach obrazowych, w których wartości pikseli były zapisane na przedziale  $[0; 1]$ . W tym momencie jednak ze względu na zapis wywołanych zdjęć do 16-bitowego pliku TIFF, przechodzimy na dyskretny przedział od 0 do  $2^{16}$ . Przedział ten jest kilkakrotnie modyfikowany w rozdziale czwartym, szczególnie przy konwersji z przestrzeni RGB do przestrzeni barw Ohty oraz przy obliczeniach związanych z generowaniem przestrzeni barw na podstawie serii fotografii o różnym naświetleniu. Wszelkie zmiany niniejszego przedziału zostały wyraźnie zaznaczone w tekście.

Powyższy etap algorytmu wieńczy niezbędne elementy cyfrowego wywoływania zdjęć. W tym momencie mamy przepis na uzyskanie ze zdjęcia surowego, zdjęcia przetworzonego, gotowego zarówno do generowania przestrzeni barw, jak i do wyświetlenia go na monitorze komputera. Mając zatem niezbędne narzędzia do przetwarzania wykonanych zdjęć, w następnym rozdziale został opisany proces rejestracji serii zdjęć, które zostały użyte do generowania przestrzeni barwowej.

## Rozdział 3

# Rejestracja serii zdjęć

Niniejszy rozdział opisuje proces rejestracji ośmiu serii zdjęć o różnym naświetleniu oraz definiuje zarówno parametry rejestrowanej sceny, jak i parametry aparatu oraz oświetlenia. Aby zarejestrowane serie zdjęć były przydatne badawczo, zdjęcia należało wykonać w sterylnych warunkach, przy zachowaniu pełnej stabilności oświetlenia i parametrów aparatu, a także kadru. Aby uzyskać kontrolowane naświetlenie sceny, jedynie jeden parametr mógł podlegać zmianie, reszta zaś pozostawała stała. Jako parametr podlegający zmianie wybrałem czas naświetlania matrycy światłoczułej aparatu. Zmiana tego parametru gwarantuje dokładną kontrolę nad poziomem naświetlania sceny. Rejestracja każdej serii rozpoczynała się od ustalenia takiego naświetlenia sceny, aby jej ekspozycja była jak najbliższa zeru, tj. aby zdjęcie reprezentowało scenę tak, jak faktycznie jest ona oświetlona. Po wykonaniu zdjęcia neutralnego zmniejszany był czas naświetlania tak, aby ekspozycja zmniejszyła się o  $\frac{1}{3}$ EV. Te kroki powtarzane były do momentu uzyskania najciemniejszego ze zdjęć. Ten sam algorytm został zastosowany do rejestracji zdjęć prześwietlonych, z tym że, poczynając od zdjęcia neutralnego, czas naświetlania był zwiększany o krok  $\frac{1}{3}$ EV. Poniższe podrozdziały dokładniej opisują parametry, jakie muszą zostać zachowane, aby wykonać właściwe serie fotografii.

### 3.1. Parametry rejestracji zdjęć

#### 3.1.1. Parametry oświetlenia

Pierwszym czynnikiem, jaki ma ogromny wpływ na wykonywanie fotografii, jest sposób oświetlenia sceny. Aby wykonać właściwą serię zdjęć, potrzeba było stabilnego i niezmiennego w czasie oświetlenia. W takim wypadku światło dzienne nie wchodziło w grę – nie ma absolutnie żadnej gwarancji, że w czasie wykonywania serii zdjęć barwa czy natężenie tego światła nie ulegnie zmianie. Z racji tego, że kolory rejestrowanej sceny nie mogą ulec zmianie, należało zastosować światło sztuczne o stałej temperaturze barwowej i natężeniu oraz takie, które nie powodowałoby migotania podczas rejestrowania fotografii.

Do tego celu idealnie nadają się studyjne lampy LED-owe, stosowane w studiach fotograficznych. Zapewniają one stabilne, niezmiennie w czasie źródło światła, którym dodatkowo można sterować. Użyta przeze mnie lampa posiada zarówno sterowanie natężeniem światła, jak i jego temperaturę. Temperatura barwowa światła została ustawiona tak, aby subiektywnie odzwierciedlać neutralne światło białe. Jednak aby ustawienie to było bardziej adekwatne, zostało dobrane tak, aby nie tylko subiektywnie osiągnąć barwę najbliższą bieli, ale także aby ta wartość była bliska standardowemu iluminantowi CIE D50, która dla przypomnienia wynosi 5003 K . Natężenie światła zaś zostało ustawione na wartość maksymalną, jaką oferuje

opisywana lampa LED. Temperatura barwowa została ostatecznie ustawiona na 4800 K i dla takiej temperatury barwowej został wybrany balans bieli w ustawieniach aparatu.

### 3.1.2. Parametry aparatu

Jak zostało wspomniane wcześniej, prawie wszystkie parametry aparatu pozostawały niezmiennie. Jedyne czas naświetlania matrycy światłoczułej ulegał zmianie o  $\pm\frac{1}{3}$  EV. Poniższa lista prezentuje parametry aparatu niezmiennie podczas rejestracji wszystkich zdjęć. Wartości zmienne, takie jak czasy naświetlania i ogniskowa są zaprezentowane w kolejnym podrozdziale, wraz z prezentacją każdej z serii fotografii. Sterowanie aparatem odbywało się poprzez aplikację mobilną i moduł WiFi, dzięki czemu uniknąłem drgań spowodowanych naciskaniem spustu migawki.

Ważnym parametrem w procesie rejestracji jest poziom ISO. Jego wielkość bezpośrednio wpływa na szumy na zdjęciach, których – z uwagi na utratę danych obrazowych – należy za wszelką cenę unikać. W związku z tym parametr ten został ustawiony na najniższą wartość, natywnie wspieraną przez użyty aparat firmy Olympus. Warto zwrócić uwagę, że podczas procesu demozaikowania rozdzielczość obrazu ulega nieznacznemu pomniejszeniu. Powód tego zabiegu został omówiony w rozdziale 2.4.2. Wszelka automatyka aparatu, taka jak np. autofokus czy lampa błyskowa, a także wszelkie algorytmy poprawiające jakość obrazu (np. odsumiające) zostały wyłączone, ponieważ wpłynęłyby negatywnie na stabilność i niezmienniczość danych badawczych:

- Rozdzielczość – 4640 x 3472 pikseli;
- Rozdzielczość po demozaikowaniu – 4624 x 3456 pikseli;
- ISO – 200 (możliwie najmniejsze);
- Autofokus – wyłączony;
- Lampa błyskowa – wyłączona;
- Przestrzeń barw – sRGB (parametr nie wpływający na surowe dane);
- Balans bieli – 4800 K ( $W_r = 1.859375$ ,  $W_g = 1$ ,  $W_b = 2.039063$ ).

## 3.2. Serie zdjęć

W tym podrozdziale zaprezentowane zostały wykonane na potrzeby pracy serie zdjęć. Sceny zostały zaprojektowane w taki sposób, aby zdjęcia zawierały szerokie spektrum barw – nie tylko w dziedzinie odcienia, ale także jasności i saturacji. Każdą serię rozpoczynałem od wykonania zdjęcia z czasem naświetlania, który subiektywnie gwarantował neutralną ekspozycję. W tym celu wspomagałem się podpowiedziami z oprogramowania aparatu, które informują fotografa o tym, czy matryca jest prześwietlona, niedoświetlona czy też oświetlona idealnie. Na rysunkach zdjęcia neutralne mogą wydawać się nieco niedoświetlone, jednak fakt ten wynika z tego, że domyślnie aparat stosuje dodatkowe algorytmy mające na celu „upiększenie” wywoływanego zdjęcia, takie jak korekcja krzywej bazowej, która znacząco zwiększa jasność i kontrast zdjęcia. Zdjęcia prezentowane w zestawieniach zostały zapisane w przestrzeni sRGB.

Wykonawszy neutralnie naświetlone zdjęcie, zmniejszałem czas naświetlania sceny o jedną jednostkę, uzyskując ekspozycję mniejszą o  $\frac{1}{3}EV$ , po czym wykonywałem kolejne zdjęcie. Procedurę tę powtarzałem dla kilkunastu zdjęć, aż do momentu, kiedy zdjęcie robiło się zbyt ciemne. Podobnie postępowałem następnie w stronę zdjęć prześwietlonych – cofałem się do ustawień neutralnych i zwiększałem czas naświetlania o jedną jednostkę, uzyskując kolejno ekspozycję większą o  $\frac{1}{3}EV$ ,  $\frac{2}{3}EV$ , itd. aż do momentu, kiedy zdjęcia stawały się zbyt prześwietlone. Nie stosowałem tutaj żadnej formuły na „zbyt” prześwietlone lub niedoświetlone zdjęcie, chodziło po prostu o subiektywne odczucie, kiedy kolejne zdjęcie nie będzie już dawało zauważalnej różnicy. Dzieje się tak, kiedy zdjęcie jest bardzo prześwietlone – większość pikseli jest już biała, więc jaśniejsze być nie mogą, analogicznie jest, kiedy zdjęcie jest całkowicie niedoświetlone – kiedy piksele są czarne i bardziej ciemne już być nie mogą.

W ten sposób podzieliłem każdą serię na trzy części – część A oznaczała zdjęcia prześwietlone, część C zdjęcia niedoświetlone, a zdjęcie neutralne oznaczone zostało jako B. W ten sposób uzyskiwałem serię zdjęć  $\{A1, \dots, An, B, C1, \dots, Cm\}$ , gdzie n, m to odpowiednio liczba prześwietlonych i niedoświetlonych zdjęć, ułożoną od zdjęcia najbardziej naświetlonego do najmniej naświetlonego. Tę procedurę powtórzyłem dla kolejnych siedmiu serii fotografii.

Oto zestawienie wszystkich ośmiu serii. Listy prezentują parametry aparatu specyficzne dla poszczególnych serii, zaś zestawienia zdjęć prezentują wybrane zdjęcia z serii. Zdjęcie neutralne znajduje się po prawej stronie każdego zestawienia. W nawiasach zaznaczono czas naświetlania zdjęcia naświetlonego neutralnie.

### 3.2.1. Seria 1

- Liczba zdjęć – 37 (18 prześwietlonych, 18 niedoświetlonych);
- Czasy naświetlania – 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, 1/10s, 1/13s, 1/15s, 1/20s, 1/25s, 1/30s, 1/40s, 1/50s, (1/60s), 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s, 1/500s, 1/640s, 1/800s, 1/1000s, 1/1250s, 1/1600s, 1/2000s, 1/2500s, 1/3200s, 1/4000s;
- Ogniskowa – 21 mm;



Rysunek 3.1: Wybrane zdjęcia z serii numer 1. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A4, A7, niedoświetlone – C5, C10, C15.



### 3.2.2. Seria 2

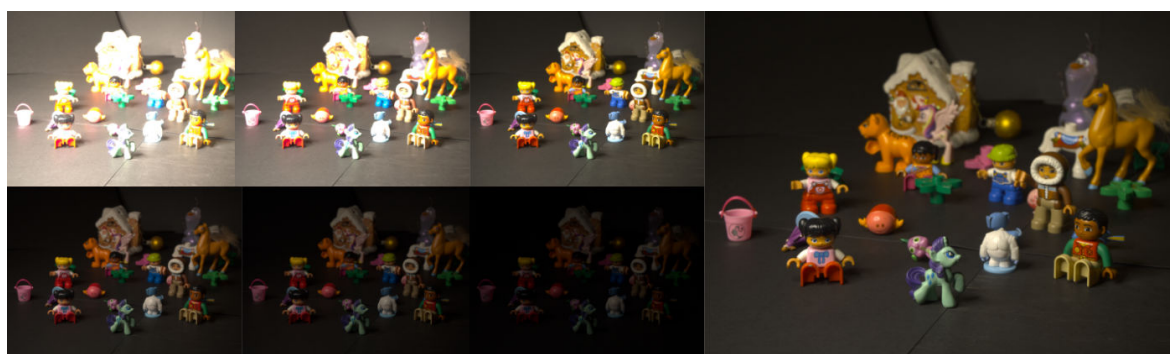
- Liczba zdjęć – 32 (16 prześwietlonych, 15 niedoświetlonych);
- Czasy naświetlania – 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, 1/10s, 1/13s, 1/15s, 1/20s, 1/25s, (1/30s), 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s, 1/500s, 1/640s, 1/800s, 1/1000s;
- Ogniskowa – 21 mm;



Rysunek 3.2: Wybrane zdjęcia z serii numer 2. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A6, A12, niedoświetlone – C3, C8, C12.

### 3.2.3. Seria 3

- Liczba zdjęć – 31 (13 prześwietlonych, 17 niedoświetlonych);
- Czasy naświetlania – 2s, 1.6s, 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, (1/10s), 1/13s, 1/15s, 1/20s, 1/25s, 1/30s, 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s, 1/500s;
- Ogniskowa – 42 mm;



Rysunek 3.3: Wybrane zdjęcia z serii numer 3. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C7, C13.

### 3.2.4. Seria 4

- Liczba zdjęć – 33 (14 prześwietlonych, 18 niedoświetlonych);
- Czasy naświetlania – 2s, 1.6s, 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, 1/10s, (1/13s), 1/15s, 1/20s, 1/25s, 1/30s, 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s, 1/500s, 1/640s, 1/800s;
- Ogniskowa – 30 mm;



Rysunek 3.4: Wybrane zdjęcia z serii numer 4. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C14.

### 3.2.5. Seria 5

- Liczba zdjęć – 35 (18 prześwietlonych, 16 niedoświetlonych);
- Czasy naświetlania – 2.5s, 2s, 1.6s, 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, 1/10s, 1/13s, 1/15s, 1/20s, (1/25s), 1/30s, 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s, 1/500s, 1/640s, 1/800s 1/1000s;
- Ogniskowa – 22 mm;



Rysunek 3.5: Wybrane zdjęcia z serii numer 5. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A6, A12, niedoświetlone – C3, C8, C13.

### 3.2.6. Seria 6

- Liczba zdjęć – 30 (13 prześwietlonych, 16 niedoświetlonych);
- Czasy naświetlania – 2s, 1.6s, 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, (1/10s), 1/13s, 1/15s, 1/20s, 1/25s, 1/30s, 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s;
- Ogniskowa – 15 mm;



Rysunek 3.6: Wybrane zdjęcia z serii numer 6. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C13.

### 3.2.7. Seria 7

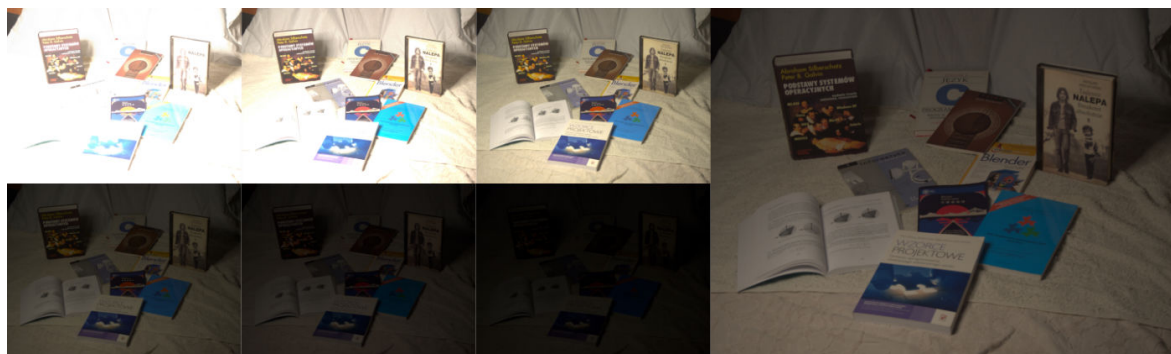
- Liczba zdjęć – 31 (13 prześwietlonych, 17 niedoświetlonych);
- Czasy naświetlania – 4s, 3.2s, 2.5s, 2s, 1.6s, 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, (1/5s), 1/6s, 1/8s, 1/10s, 1/13s, 1/15s, 1/20s, 1/25s, 1/30s, 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s;
- Ogniskowa – 21 mm;



Rysunek 3.7: Wybrane zdjęcia z serii numer 7. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C13.

### 3.2.8. Seria 8

- Liczba zdjęć – 32 (16 prześwietlonych, 15 niedoświetlonych);
- Czasy naświetlania – 2s, 1.6s, 1.3s, 1s, 0.8s, 0.6s, 0.5s, 0.4s, 0.3s, 1/4s, 1/5s, 1/6s, 1/8s, 1/10s, 1/13s, 1/15s, (1/20s), 1/25s, 1/30s, 1/40s, 1/50s, 1/60s, 1/80s, 1/100s, 1/125s, 1/160s, 1/200s, 1/250s, 1/320s, 1/400s, 1/500s, 1/640s;
- Ogniskowa – 31 mm;



Rysunek 3.8: Wybrane zdjęcia z serii numer 8. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C13.

Powyższe serie zawierają łącznie 261 zdjęć, zatem pozyskany został dosyć obfity materiał badawczy. Kolejnym krokiem, opisanym w rozdziale drugim, było wywołanie każdego zdjęcia, czyli przetworzenie go z postaci surowej do zdjęcia barwowego w liniowej przestrzeni RGB. Zdjęcia takie po wywołaniu zostały zapisane w formie 16-bitowych plików TIFF i właśnie od obrazów w takiej formie zaczyna się kolejny rozdział, który opisuje problem badawczy zawarty w tytule niniejszej pracy, mianowicie generowanie przestrzeni barw na podstawie ośmiu serii fotografii o różnym naświetleniu.



## Rozdział 4

# Tworzenie przestrzeni barw

Posiadając dane obrazowe w postaci wywołanych zdjęć w liniowej przestrzeni RGB, możemy przejść do przetwarzania obrazów i badania zmienności parametrów barw. Zaczynamy od wczytania zapisanych wcześniej 16-bitowych, barwowych plików TIFF, zatem mówiąc obraz, będę miał na myśli macierze, których każda komórka będzie trójelementową tablicą liczb (w tym momencie całkowitych, jednak podczas obliczeń zakładam, że liczby te mogą być zmiennopozycyjne), reprezentujących piksel w modelu RGB, a początkowe wartości poziomów RGB będą wynosić od 0 do  $2^{16} - 1$ , co bezpośrednio wynika z głębi bitowej, w jakiej zostały zapisane wywołane obrazy.

Zdefiniuję teraz jakie parametry będą podlegały obserwacji. Dowolna barwa składa się z dwóch podstawowych składowych – chrominancji i luminancji. Luminancją sterowałem ręcznie, poprzez zmianę wartości czasu naświetlania matrycy światłoczułej aparatu. Pozostawał zatem do zbadania drugi parametr, mianowicie jak zmiana jasności wpływa na chrominancję, a dokładniej mówiąc jak zmienia się odcień i nasycenie koloru wraz ze zmianą jasności. Do tego celu potrzebowałem przestrzeni barwowej, której składowe chrominancji i luminancji byłyby nieskorelowane, zatem korzystanie z przestrzeni RGB, z racji na silną korelację jej składowych, byłoby bezsensowne. Należało zatem skorzystać z nieskorelowanej przestrzeni barw, którą można łatwo uzyskać, startując z przestrzeni RGB i od której można łatwo przejść z powrotem do przestrzeni RGB. Ścisłej mówiąc, chodzi o przestrzeń, do której prowadzi proste przekształcenie liniowe i równie proste przekształcenie odwrotne. Idealną przestrzenią do tego celu okazała się być przestrzeń Ohty.

### 4.1. Przejście z i do przestrzeni Ohty

Jak wspomniałem, przestrzeń RGB jest przestrzenią, której składowe są silnie skorelowane. Oznacza to, że zmiana barwy w tej przestrzeni prawie zawsze oznacza zmianę wartości wszystkich trzech składowych. O ile taka przestrzeń idealnie nadaje się do wyświetlania obrazów, nie nadaje się ona do bardziej wyrafinowanych zadań, takich jak np. kompresja czy segmentacja.

Ohta zdefiniował swoją przestrzeń jako proste, liniowe przekształcenie z przestrzeni RGB. Przestrzeń Ohty [27] zbudowana jest z trzech nieskorelowanych składowych  $I_1$ ,  $I_2$  i  $I_3$ , gdzie oś  $I_1$  niesie informację intensywnościową, zaś  $I_2$  i  $I_3$  informację barwową, z czego te dwie ostatnie można porównać do mechanizmu barw przeciwstawnych. O ile pierwotnie ta przestrzeń była tworzona w celu lepszej segmentacji obrazów, nadawała się idealnie do moich badań.

Spójrzmy teraz jak wygląda wzór na przekształcenie z przestrzeni RGB do przestrzeni Ohty<sup>1</sup>:

$$\begin{cases} I_1 = (R + G + B)/3 \\ I_2 = (R - B)/2 \\ I_3 = (2G - R - B)/4 \end{cases} \quad (4.1)$$

Problem z powyższym przekształceniem jest taki, że stosując liczby całkowite, obliczone wyniki ulegają zaokrągleniu, przez co są niedokładne. Co prawda operujemy na liczbach zmienno-  
popycyjnych, jednak zależy mi, aby przy przejściu z jednej przestrzeni do drugiej nie musieć dzielić liczb, pomoże mi to uzyskać prostszy wzór funkcji obliczającej zmiany chrominancji w zależności od zmian luminancji.

Z powyższego równania wystarczy usunąć dzielenie, aby uzyskać prostszy wzór, który zachowa wspomniane cechy przy konwersji do oraz z przestrzeni Ohty, likwidując konieczność niepotrzebnego zaokrąglania wyników. Zakładając, że piksel zapisujemy w postaci trójelementowego wektora liczb, otrzymane równanie możemy zapisać następująco:

$$\begin{bmatrix} I'_1 \\ I'_2 \\ I'_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & -1 \\ -1 & 2 & -1 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4.2)$$

Zdefiniuję teraz przekształcenie odwrotne. Poniższy wzór prezentuje przejście z otrzymanej ze wzoru (4.2) przestrzeni Ohty, z powrotem do przestrzeni RGB:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & -\frac{1}{6} \\ \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{2} & -\frac{1}{6} \end{bmatrix} \begin{bmatrix} I'_1 \\ I'_2 \\ I'_3 \end{bmatrix} \quad (4.3)$$

Zakładając, że na wejściu mamy do czynienia z 16-bitowym plikiem TIFF, przedziały dla poszczególnych składowych tak otrzymanej przestrzeni Ohty, prezentują się następująco:

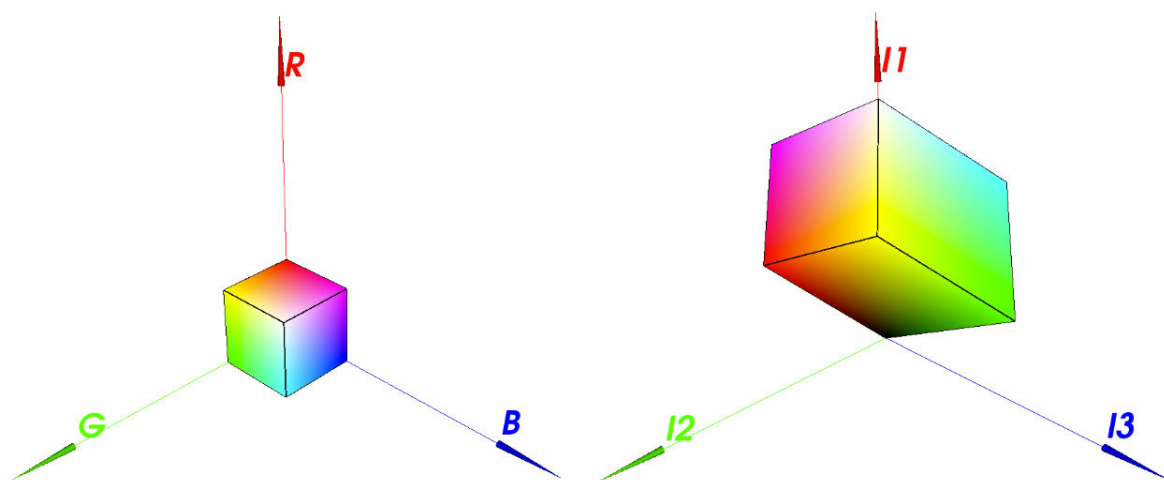
- $[0; 3 \cdot (2^{16} - 1)]$  – dla osi  $I_1$ ;
- $[-(2^{16} - 1); (2^{16} - 1)]$  – dla osi  $I_2$ ;
- $[-2 \cdot (2^{16} - 1); 2 \cdot (2^{16} - 1)]$  – dla osi  $I_3$ .

Co prawda wychodzimy tutaj poza początkowy zakres  $[0; (2^{16} - 1)]$ , jednak w tym momencie nie stanowi to żadnej przeszkody, bowiem operujemy na 64-bitowych liczbach zmienno-  
popycyjnych o podwójnej precyzji, zapisywanych w pamięci, a nie w pliku TIFF.

Zastosujmy zatem powyższe przekształcenie dla każdego obrazu z serii. Uzyskane zostały dane obrazowe zapisane w przestrzeni Ohty, zatem mogę już przejść do badania zmienności chrominancji w zależności od zmiany luminancji. Dokładniej rzecz biorąc, w ramach następnego podrozdziału zostały obliczone parametry  $\Delta I_2$  i  $\Delta I_3$ , czyli odchylenia saturacji i odcienia względem luminancji (osi  $I_1$ ).

---

<sup>1</sup>A dokładnie rzecz biorąc jeden z wielu, jakie można znaleźć w literaturze.



Rysunek 4.1: Porównanie klasycznej kostki RGB (po lewej) do przestrzeni Ohya (po prawej).

## 4.2. Schemat badania zmienności chrominancji

Zgodnie z rozdziałem trzecim, każda seria składa się ze zdjęcia „neutralnego”, tj. takiego, który zawiera zapis sceny oświetlonej w sposób neutralny (scena nie jest ani prześwietlona, ani niedoświetlona), zdjęć prześwietlonych oraz zdjęć niedoświetlonych. Zdjęcia podczas rejestracji zostały odpowiednio ponumerowane – od najjaśniejszego do najciemniejszego. W tym momencie odwróćmy tę kolejność, definiując każdą serię zdjęć jako składającą się ze zdjęć od 1 do  $n$ , gdzie  $n$  to liczba zdjęć. Wówczas uzyskałem szereg, według którego porównywałem zdjęcia parami – zdjęcie 1 z 2, 2 z 3, ..., zdjęcie  $n-1$  ze zdjęciem  $n$  (najbardziej prześwietlonym). Badanie zmienności parametrów  $\Delta I_2$  i  $\Delta I_3$  należało zacząć od porównania zdjęcia najbardziej niedoświetlonego z kolejnym (nieco mniej niedoświetlonym) zdjęciem z serii. Tę procedurę powtórzyłem dla kolejnych par zdjęć w szeregu, aż do zdjęć o numerach  $n-1$  i  $n$ .

Zdefiniuję proces badania parametrów  $\Delta I_2$  i  $\Delta I_3$ , na przykładzie. Niech piksel ma współrzędne  $(i, j)$ , identyczne na obu obrazach z serii o numerach  $m-1$  i  $m$ , gdzie  $m \in \{1, \dots, n\}$ , a jego wartości niech będą następujące:

$$P_1(i, j)^{m-1} = \begin{cases} I_1^{m-1} = 500 \\ I_2^{m-1} = 20 \\ I_3^{m-1} = -10 \end{cases} \quad P_2(i, j)^m = \begin{cases} I_1^m = 1000 \\ I_2^m = -30 \\ I_3^m = 20 \end{cases}$$

W obu przypadkach mamy do czynienia z ciemnymi odcieniami szarości, jednak w pierwszym wypadku szarość ta zawiera w sobie domieszkę koloru fioletowego, zaś w drugim domieszkę koloru cyjanowego (i jest to kolor odrobinę jaśniejszy od pierwszego). To, jakie są tu dokładnie zapisane kolory jest sprawą drugorzędną, najważniejsze jest zrozumienie, w jaki sposób zostaną obliczone parametry zmienności chrominancji  $\Delta I_2$  i  $\Delta I_3$ .

Zacznijmy od tego, że chcemy obliczyć zmienność chrominancji (osie  $I_2$  i  $I_3$ ), w zależności od zmienności luminancji (oś  $I_1$ ). Musimy zatem w pierwszej kolejności sprawdzić czy i jaka nastąpiła zmiana w luminancji między badanymi pikselami na tych obrazach. Jeśli zmiana



luminancji nie nastąpiła (jest równa 0), pomijamy wszelkie obliczenia. Niech parametr  $\Delta I_1$ , oznacza różnicę w poziomie luminancji między badanymi pikselami:

$$\Delta I_1^m = |I_1^m - I_1^{m-1}| \quad (4.4)$$

Jeśli ten parametr jest większy od zera (w ramach podanego przykładu jest i wynosi 500), kontynuujemy obliczenia.

W następnym kroku zdefiniujemy dwie funkcje, które służyły do obliczania przyrostów chrominancji na osiach  $I_2$  i  $I_3$ , w odniesieniu do poziomu luminancji na osi  $I_1$ . Co prawda do tego momentu oś luminancji zawierała wyłącznie liczby całkowite, jednak zdefiniuję te funkcje bardziej ogólnie – na zbiorze liczb rzeczywistych – aby można było je swobodnie przekształcać na późniejszych etapach generowania przestrzeni:

$$\begin{aligned} f_{\Delta I_2} : R - \{0\} \times \mathbb{R}^2 &\rightarrow \mathbb{R} \\ f_{\Delta I_2}(\Delta I_1^m, I_2^{m-1}, I_2^m) &= (I_2^m - I_2^{m-1}) / \Delta I_1^m \end{aligned} \quad (4.5)$$

$$\begin{aligned} f_{\Delta I_3} : R - \{0\} \times \mathbb{R}^2 &\rightarrow \mathbb{R} \\ f_{\Delta I_3}(\Delta I_1^m, I_3^{m-1}, I_3^m) &= (I_3^m - I_3^{m-1}) / \Delta I_1^m \end{aligned} \quad (4.6)$$

Tak więc w podanym przykładzie  $\Delta I_2$  wyniesie  $-\frac{1}{10}$ , a parametr  $\Delta I_3$  wyniesie  $\frac{3}{50}$ .

Powyższy algorytm definiuje przepis na obliczanie przyrostów chrominancji. Podczas obliczeń możemy wielokrotnie napotkać się na taką samą wartość  $I_1$ . Należy więc stworzyć strukturę danych, która zapamięta wszystkie obliczone delty dla danego poziomu jasności. Zdefiniujemy dwie krotki o identycznej budowie, które będą przypisane dla każdego poziomu jasności w badanych obrazach o indeksie  $m$  (tj. tych jaśniejszych z badanej pary). Krotki te składają się z tego poziomu jasności i listy obliczonych parametrów  $\Delta I_2$  i  $\Delta I_3$ . Chodziło mianowicie o to, aby dla każdego poziomu jasności zgromadzić wszelkie zmiany chrominancji w postaci listy, którą na koniec zsumowałem i uzyskałem ostateczny parametr  $\Delta I_2$  lub  $\Delta I_3$ , dla każdego poziomu jasności w odniesieniu do całej tworzonej przestrzeni barw. Zamiast tego można by również sumować i uśredniać od razu, jednak stworzenie takich list jest szybsze, a w szczególności bardziej dokładne obliczeniowo i przejrzyste pod kątem programistycznym. Zmienna  $j$  jest indeksem poziomu jasności, zaś zbiór  $J$  zawiera wszystkie poziomy jasności, dla których przeprowadzono obliczenia. Omawiane krotki prezentują się następująco:

1.  $(I_1^j, \{f_{\Delta I_2}(\Delta I_1^j, I_2^{j-1}, I_2^j) : j \in J\})$ ;
2.  $(I_1^j, \{f_{\Delta I_3}(\Delta I_1^j, I_3^{j-1}, I_3^j) : j \in J\})$ .

W powyższych krotkach zostały zapisane wszystkie delty dla wszystkich napotkanych poziomów jasności w jaśniejszych obrazach wszystkich badanych par. W ostatnim kroku należało zsumować zapisane w krotkach listy i podzielić wynik przez długość danej listy, uzyskując średnią arytmetyczną parametrów  $\Delta I_2$  i  $\Delta I_3$ , przypisanych do danego poziomu jasności. Otrzymaną strukturę można było wówczas traktować jako mapę, której klucz to poziom jasności (oś  $I_1$ ), zaś wartość to średnia arytmetyczna ze zmian chrominancji  $\Delta I_2$  lub  $\Delta I_3$ , przyporządkowana do danego poziomu jasności. Otrzymaną mapę posortowano następnie rosnąco względem klucza.

W ten sposób uzyskałem algorytm, który bada, jak zmienia się chrominancja między dwoma sąsiednimi obrazami z serii, w zależności od zmiany jasności. Dla lepszego zrozumienia zagadnienia, warto przytoczyć pseudokod algorytmu. Poprzez  $A(i, j)$  i  $B(i, j)$  oznaczmy piksel o współrzędnych  $(i, j)$  odpowiednio dla obrazu o numerze  $m-1$  i  $m$ . Parametry  $\Delta I_1$ ,  $\Delta I_2$ ,  $\Delta I_3$  to odpowiednio  $\text{delta1}$ ,  $\text{delta2}$  i  $\text{delta3}$ . Dodatkowo poprzez  $A(i, j).I1$  rozumiem tutaj pierwszą składową piksela obrazu  $A$ , reprezentowanego w przestrzeni Ohty itd.:

```

mapDelta2, mapDelta3 <- zainicjuj jako puste mapy

Dla każdego piksela (i, j) z obrazów A i B:
  delta1 <- abs(B(i, j).I1 - A(i, j).I1)
  Jeśli delta1 != 0:
    delta2 <- (B(i, j).I2 - A(i, j).I2) / delta1
    delta3 <- (B(i, j).I3 - A(i, j).I3) / delta1

    Jeśli mapDelta2 zawiera klucz B(i, j).I1:
      Dodaj do listy pod kluczem B(i, j).I1 wartość delta2
    W przeciwnym wypadku
      Utwórz jednoelementową listę z wartością delta2
      Dodaj listę do mapDelta2 pod kluczem B(i, j).I1

    Jeśli mapDelta3 zawiera klucz B(i, j).I1:
      Dodaj do listy pod kluczem B(i, j).I1 wartość delta3
    W przeciwnym wypadku
      Utwórz jednoelementową listę z wartością delta3
      Dodaj listę do mapDelta3 pod kluczem B(i, j).I1

Dla każdej pary klucz, lista z mapDelta2:
  średnia <- średnia arytmetyczna z elementów listy
  zamień listę w mapie na obliczoną średnią

Dla każdej pary klucz, lista z mapDelta3:
  średnia <- średnia arytmetyczna z elementów listy
  zamień listę w mapie na obliczoną średnią

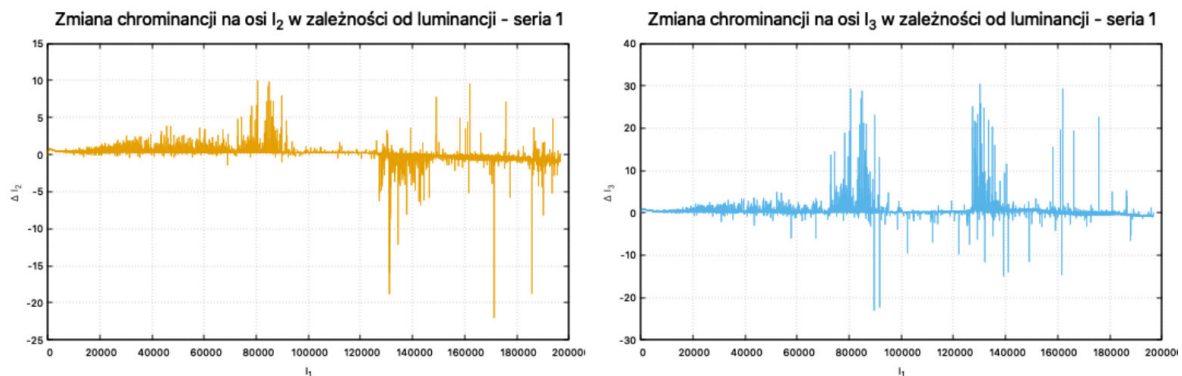
```

Korzystając z powyższego algorytmu wyeksportowałem pary liczb jasność –  $\overline{\Delta I_2}$  i jasność –  $\overline{\Delta I_3}$  do dwóch plików tekstowych, po jednej parze na każdą linię<sup>2</sup>. Za pomocą tych plików stworzyłem wykresy, a także prowadziłem opisane w dalszej części pracy obliczenia. Na tym etapie obliczone były już zmiany chrominancji dla każdego (lub prawie każdego, jeśli ten nie wystąpił) poziomu jasności, w odniesieniu do omawianej przestrzeni Ohty – dla przypomnienia poziomy jasności wynoszą od 0 do  $3 \cdot (2^{16} - 1)$ . Mogło się zdarzyć, że jakiś poziom jasności nie wystąpił na żadnym z obrazów, jednak tym nie należy się przejmować. Problemem byłby jedynie brak poziomów 0 i  $3 \cdot (2^{16} - 1)$ , jeśli tych brakowało, zostały dodane jako klucze do obliczonych map, jako wartości przypisując im 0.

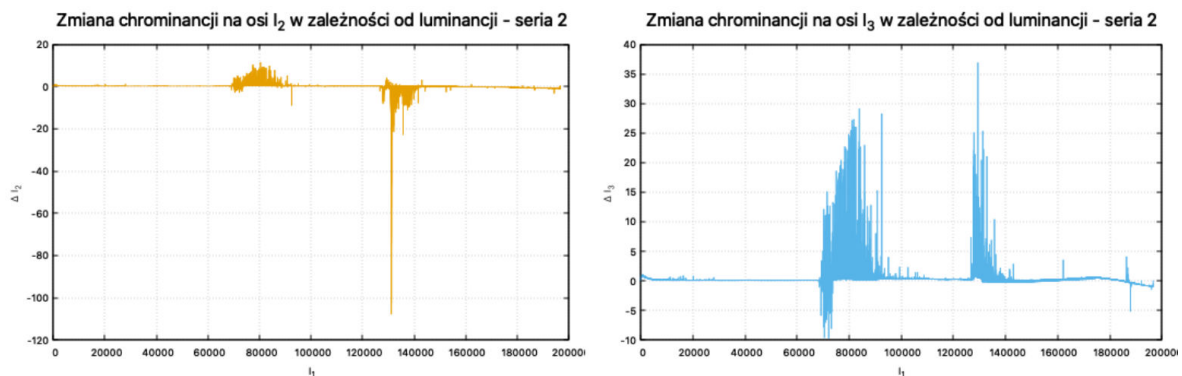
<sup>2</sup>Pozioma kreska oznacza tu średnią arytmetyczną list parametrów  $\Delta I_2$  oraz  $\Delta I_3$ .

### 4.3. Wyniki badań zmienności chrominancji

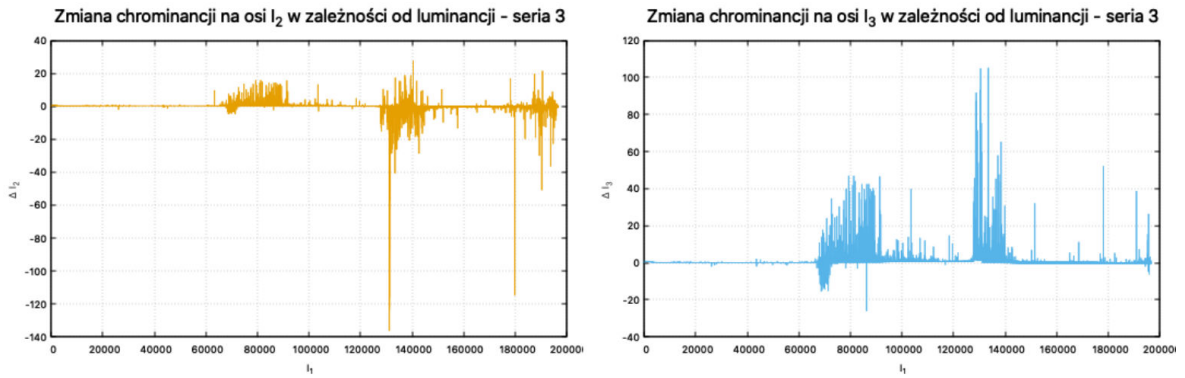
Za pomocą powyższego schematu została przetworzona każda seria zdjęć. Niniejszy podrozdział prezentuje wartości parametrów  $\Delta I_2$  i  $\Delta I_3$ , w zależności od poziomów jasności. Poniżej zaprezentowane zostały stosowne wykresy dla każdej serii zdjęć. Oś pozioma to poziom jasności (oś  $I_1$ ), zaś osie pionowe to odpowiednie zmiany chrominancji  $\Delta I_2$  i  $\Delta I_3$ , w zależności od poziomu jasności.



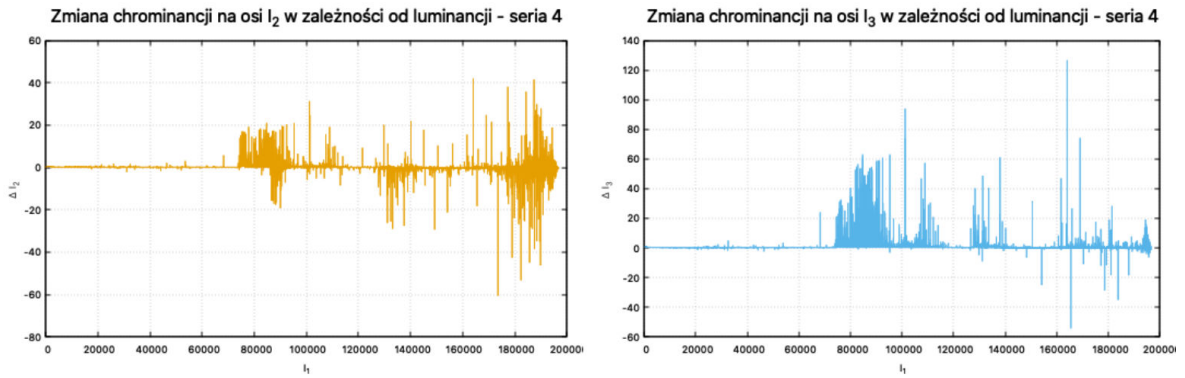
Rysunek 4.2: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 1.



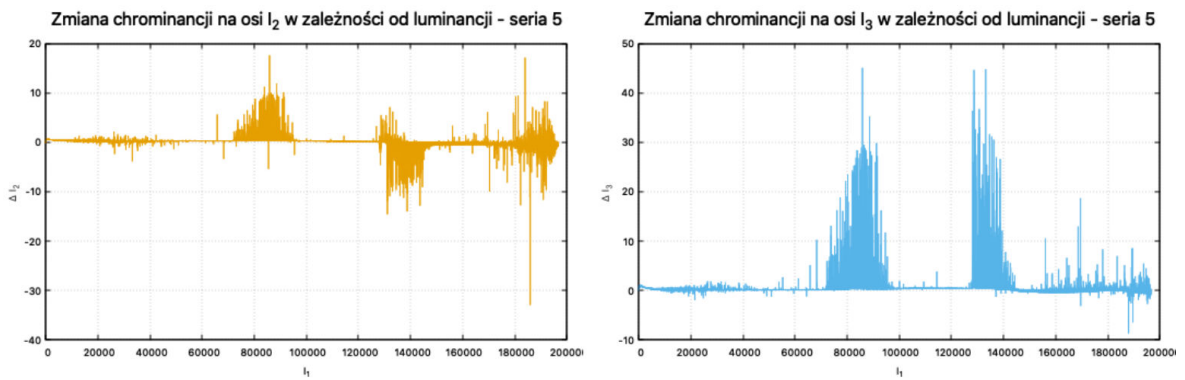
Rysunek 4.3: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 2.



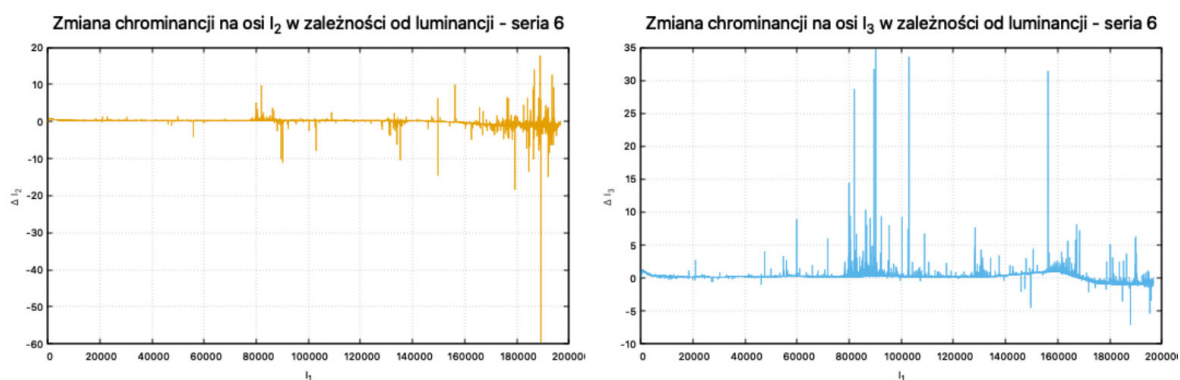
Rysunek 4.4: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 3.



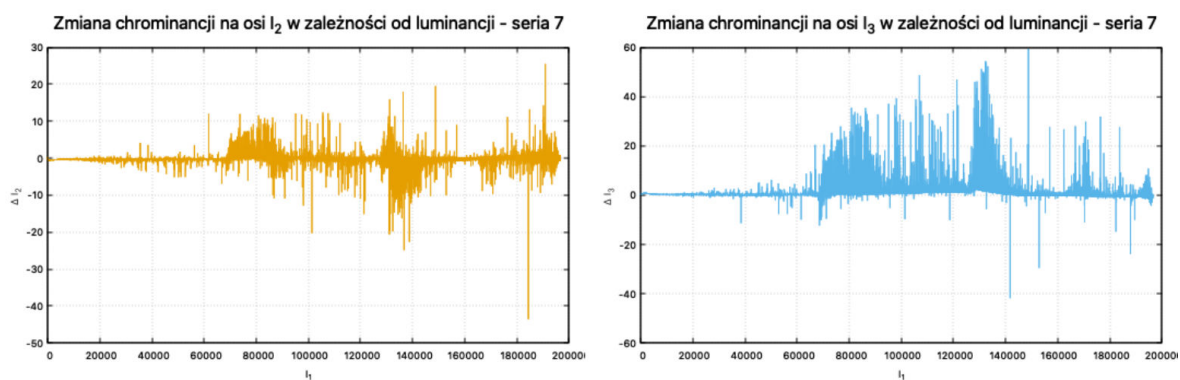
Rysunek 4.5: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 4.



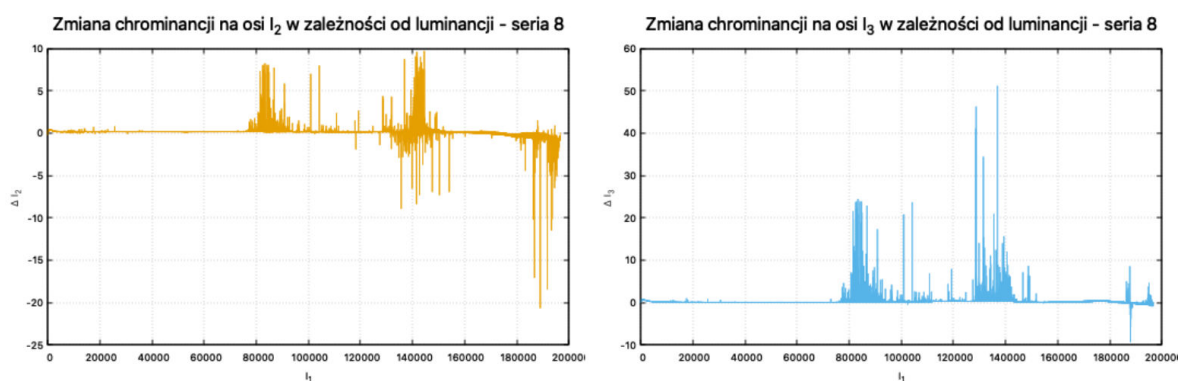
Rysunek 4.6: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 5.



Rysunek 4.7: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 6.



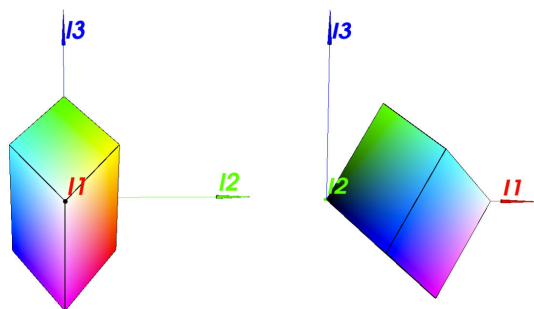
Rysunek 4.8: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 7.



Rysunek 4.9: Zmiany chrominancji na osiach  $I_2$  i  $I_3$ , w zależności od wartości na osi  $I_1$  – seria nr 8.

Zastanówmy się teraz jaką informację niosą te wykresy. To, że zmiana chrominancji wraz ze zmianą luminancji jest istotna, widać na pierwszy rzut oka. Co więcej zmiany te są dosyć spore, co dokładniej widać w kolejnym podrozdziale, kiedy obliczane są kolejne sumy częściowe z uśrednionych wyników. Mimo że przyrosty  $\Delta I_2$  i  $\Delta I_3$  rzędu  $\pm 10$  jednostek na osi pionowej wydają się małe, w porównaniu do rozległości przedziału osi poziomej, przyrosty te są dosyć stałe wzdłuż większej długości osi  $I_1$ , co w konsekwencji dało znaczącą deformację przestrzeni Ohty.

Warto dodatkowo spojrzeć co poszczególne delty oznaczają. Na rysunku 4.10 przedstawiono rzuty prostopadłe do osi  $I_1$  i  $I_2$ , które pomogą uświadomić nam co oznaczają poszczególne wartości delt. W przypadku gdy  $\Delta I_2$  się zmienia, w zależności od tego czy rośnie, czy maleje, poruszamy się od odcieni niebieskich lub błękitnych do odcieni żółtych lub czerwonych. Natomiast w przypadku parametru  $\Delta I_3$ , poruszamy się od odcieni niebieskich lub czerwonych w stronę odcieni błękitnych, zielonych lub żółtych. Spostrzeżenia te są dosyć ogólne, ponieważ to w jakim kierunku następuje zmiana chrominancji, zależy nie od jednego, lecz od obydwóch współczynników  $\Delta$ .



Rysunek 4.10: Rzut na przestrzeń Ohty prostopadłe do osi  $I_1$  i  $I_2$ .

Prawie we wszystkich wykresach możemy zauważyć dwa skupiska największych zmian, co więcej – skupiska te występują zarówno w przypadku parametru  $\Delta I_2$ , jak i  $\Delta I_3$ , w okolicy liczb 80000 i 140000 na osi poziomej. W przypadku pierwszej z delt, pierwsze lokalne maksimum zmian jest „górką”, drugie zaś na większości wykresów tworzy wgłębienie. Górka oznacza, że parametr  $\Delta I_2$  rośnie, a co za tym idzie odcień koloru staje się bardziej ciepły. Jeśli chodzi o wgłębienie, to sytuacja jest odwrotna – kolory stają się chłodniejsze. Widać zatem, że wraz ze wzrostem poziomu jasności, kolory stają się coraz cieplejsze, zaś po przekroczeniu połowy poziomu luminancji, stają się nieco chłodniejsze. Na uwagę zasługują również spore fluktuacje  $\Delta I_2$  w obszarze największych poziomów jasności. Zmiany te są jednak bardzo różnorodne i żeby powiedzieć o nich coś więcej, należy złączyć osiem otrzymanych przestrzeni w jedną, uśredniając wyniki.

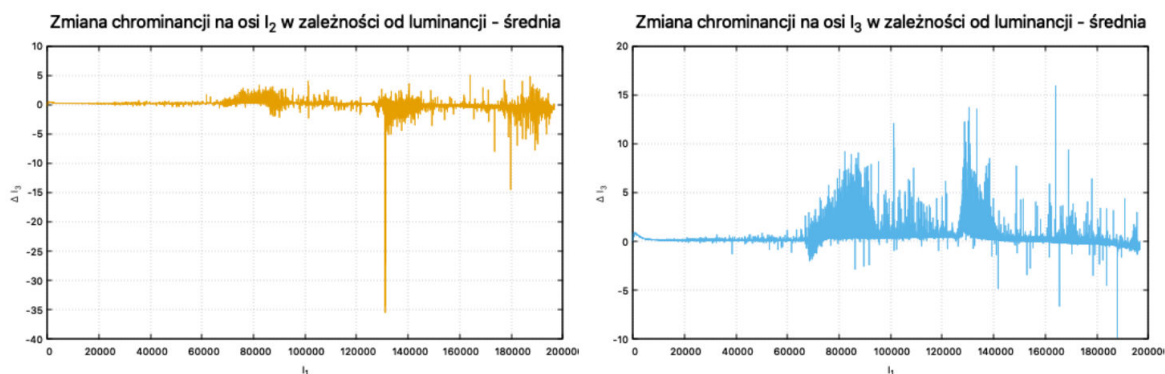
W podobnych przedziałach luminancji występują również gwałtowne przyrosty w parametrze  $\Delta I_3$ , jednak w tym wypadku obie zmiany tworzą górkę, oznaczając, że wraz ze wzrostem jasności barwy przechodzą stopniowo od odcieni głęboko niebieskich, czerwonych i fioletowych w stronę zieleni, żółci i turkusów.

Osiem zbadanych serii zdjęć dało bogaty zestaw wyników, jednak aby móc stworzyć jednolitą przestrzeń barw, musiałem zebrać otrzymane wyniki w całość. W następnej kolejności opisany został sposób, w jaki można skonsolidować otrzymane dane, a także uprościć otrzy-

many wykres funkcji poprzez aproksymację, po to, aby w jeszcze prostszy sposób móc reprezentować generowaną przestrzeń, bez konieczności podawania dziesiątek tysięcy wartości obliczonych parametrów.

#### 4.4. Ujednolicenie i uproszczenie wyników

Obliczonych zostało osiem par funkcji  $f_{\Delta I_2}$  i  $f_{\Delta I_3}$  – po jedną parę na każdą serię fotografii. Znając wartości tych funkcji, mogłem następnie obliczyć średnie arytmetyczne ich wartości, w celu ujednolicenia wyników. Poniższe wykresy prezentują uśrednione wartości parametrów  $\Delta I_2$  i  $\Delta I_3$ :



Rysunek 4.11: Średnie arytmetyczne parametrów  $\Delta I_2$  i  $\Delta I_3$ .

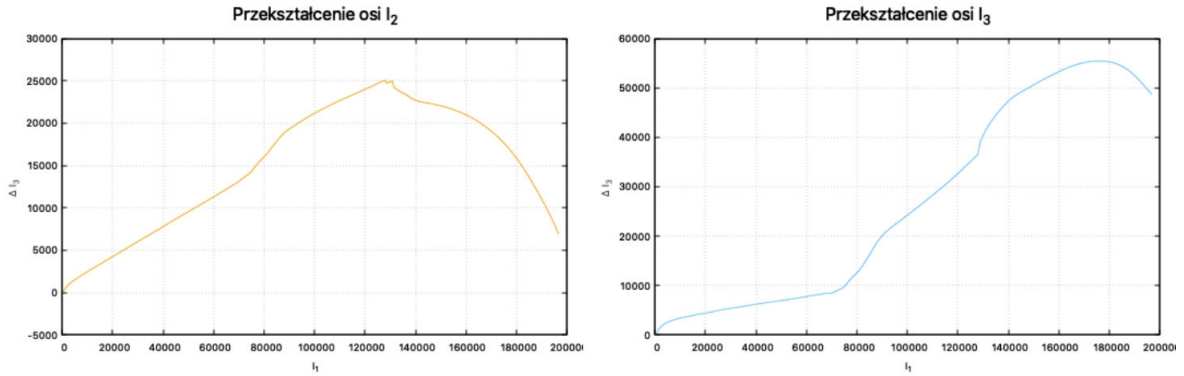
Powyższe funkcje możemy traktować jako różniczki funkcji, którą mamy ostatecznie obliczyć. Mamy zdefiniowane przyrosty funkcji, zatem scałkujemy nasze funkcje  $f_{\Delta I_2}$  i  $f_{\Delta I_3}$ , uzyskując właściwe przekształcenia przestrzeni Ohty:

$$F_{\Delta I_2}(I_1^i) = \sum_{j=0}^i f_{\Delta I_2}(\Delta I_1^j, I_2^{j-1}, I_2^j) \quad (4.7)$$

Analogicznie należy scałkować funkcję  $f_{\Delta I_3}$ :

$$F_{\Delta I_3}(I_1^i) = \sum_{j=0}^i f_{\Delta I_3}(\Delta I_1^j, I_3^{j-1}, I_3^j) \quad (4.8)$$

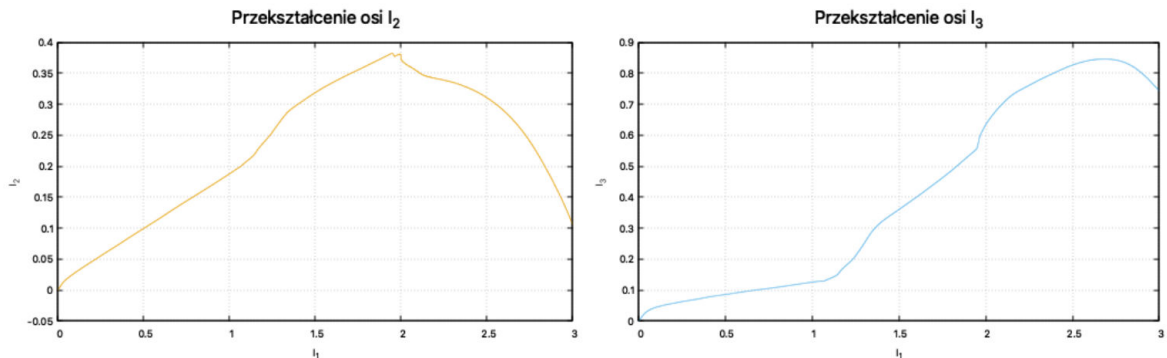
Na rysunku 4.12 zaprezentowane zostały funkcje scałkowane według powyższego schematu.



Rysunek 4.12: Funkcje  $F_{\Delta I_2}$  i  $F_{\Delta I_3}$  – właściwe przekształcenia osi  $I_2$  i  $I_3$ .

Funkcje te są jednak zdefiniowane na przedziale uzależnionym od 16-bitowego formatu TIFF. Zeskalujmy teraz jednorodnie funkcje  $F$ , aby znormalizować przedział na osi poziomej do  $[0; 3]$ . Dzięki temu otrzymujemy bardziej ogólne przekształcenia osi chrominancji, które będzie można stosować niezależnie od głębi bitowej obrazu. Wówczas mogliśmyby przekształcić kostkę RGB rozpiętą na przedziale  $[0; 1]^3$  do przestrzeni Ohty według wzoru 4.2, zastosować opisane przekształcenie osi przy pomocy funkcji  $F$ , a następnie powrócić do przestrzeni RGB, zachowując poprzednią skalę. Do tej pory oś pozioma określona była na przedziale  $[0; 3 \cdot (2^{16} - 1)]$ , zatem zarówno elementy dziedziny, jak i przeciwdziedziny trzeba podzielić przez  $2^{16} - 1$ :

$$\begin{aligned} N_{\Delta I_2}(I_1/(2^{16} - 1)) &= F_{\Delta I_2}(I_1)/(2^{16} - 1) \\ N_{\Delta I_3}(I_1/(2^{16} - 1)) &= F_{\Delta I_3}(I_1)/(2^{16} - 1) \end{aligned} \quad (4.9)$$



Rysunek 4.13: Funkcje  $F_{\Delta I_2}$  i  $F_{\Delta I_3}$  znormalizowane do przedziału  $[0; 3]$  na osi  $I_1$ .

Te wykresy są już stanowczo bardziej czytelne i wygładzone, jednak jeśli chcielibyśmy w tym momencie zdefiniować taką przestrzeń w celu jej dalszego wykorzystania, musielibyśmy podać aż  $2 \cdot 2^{16}$  parametrów, gdzie każdy z nich byłby 64-bitowym typem zmiennopozycyjnym o podwójnej precyzji. Nie jest to ani wygodne, ani tym bardziej oszczędne pamięciowo. Należy zatem zaproponować prostą, ale dokładną aproksymację obliczonych funkcji  $N$ . Wydajnym



pamięciowo sposobem aproksymacji funkcji jest zastosowanie szeregów Fouriera [11]:

$$p(t) = a_0 + \sum_{k=1}^n (a_k \cos kt + b_k \sin kt)$$

W oparciu o [8] stworzony został program w Pythonie wykonujący aproksymację powyższych funkcji przy pomocy szeregów Fouriera. Dało mi to możliwość zdefiniowania funkcji N jako sumy funkcji sin i cos o stosunkowo małej liczbie współczynników. W przypadku funkcji  $N_{\Delta I_2}$  bardzo dobre przybliżenie szeregiem Fouriera uzyskałem w liczbie kroków  $n = 5$ , dla funkcji  $N_{\Delta I_3}$  lepiej było jednak zastosować o jeden krok więcej. Szeregi Fouriera aproksymują na podstawie funkcji sin i cos, więc przed wykonaniem obliczeń należało funkcje N przeskalować i przesunąć do przedziałów naturalnych dla funkcji trygonometrycznych, czyli  $[-\pi; \pi]$  w przypadku osi x oraz  $[-1; 1]$  w przypadku osi y. Niech  $max_{\Delta I_2}$  i  $max_{\Delta I_3}$  będą maksymalnymi, co do modułu wartościami funkcji  $N_{\Delta I_2}$  i  $N_{\Delta I_3}$ . Wówczas przekształcenie z funkcji N takie, aby ich dziedzina mieściła się w przedziale  $[-\pi; \pi]$ , a przeciwdziedzina w przedziale  $[-1; 1]$  wygląda następująco (analogiczny wzór stosujemy dla  $N_{\Delta I_3}$ ):

$$N'_{\Delta I_2}(I_1) = \frac{N_{\Delta I_2}(\frac{2\pi \cdot I_1}{3} - \pi)}{0.5 \cdot max_{\Delta I_2}} - 1 \quad (4.10)$$

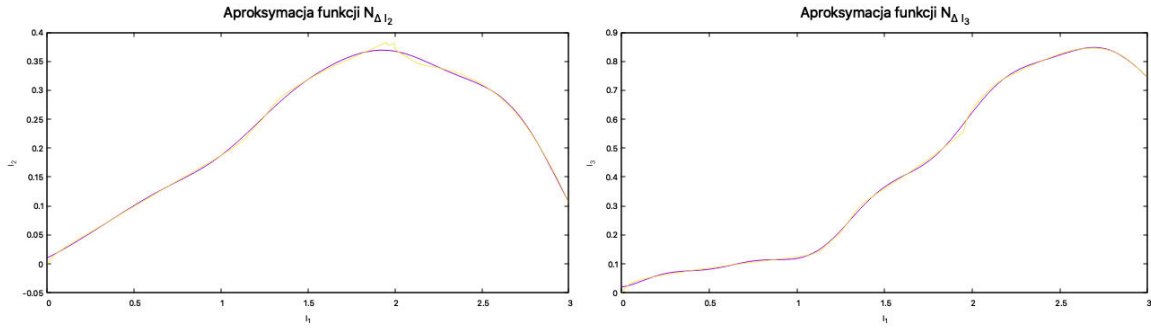
Poprzez  $\tilde{N}_{\Delta I_2}$  i  $\tilde{N}_{\Delta I_3}$  oznaczmy aproksymację odpowiednio funkcji  $N'_{\Delta I_2}$  i  $N'_{\Delta I_3}$ . Po aproksymacji należy wykonać przekształcenie odwrotne, powracając do pierwotnego przedziału funkcji N (gdzie dziedziną funkcji był przedział  $[0; 3]$ ). Poniższy wzór zawiera już odpowiednie przekształcenie powrotne, więc mogłem go bezpośrednio zastosować, aby przeprowadzić transformację przestrzeni Ohty, w której przedział luminancji wynosi  $[0; 3]$  (wzór dla  $\tilde{N}_{\Delta I_3}$  jest analogiczny):

$$\tilde{N}_{\Delta I_2}(I_I) = a_0 + \sum_{k=1}^n \left( a_k \cos(k \cdot w \cdot (I_I \cdot (\frac{\pi}{1.5}) - \pi)) + b_k \sin(k \cdot w \cdot (I_I \cdot (\frac{\pi}{1.5}) - \pi)) + 1 \right) \cdot \frac{max_{\Delta I_2}}{2} \quad (4.11)$$

Wystarczyło potem podstawić odpowiednie współczynniki z tabeli 4.1, aby uzyskać pełne wzory funkcji  $\tilde{N}_{\Delta I_2}$ , która przekształca przestrzeń Ohty względem osi  $I_2$  oraz  $\tilde{N}_{\Delta I_3}$ , która przekształca niniejszą przestrzeń względem osi  $I_3$ .

Tabela 4.1: Współczynniki  $a_k$ ,  $b_k$  oraz  $w$  dla szeregu Fouriera aproksymującego przekształcenia osi  $I_2$  i  $I_3$ .

+	Współczynniki $a_k$ i $b_k$ dla $\tilde{N}_{\Delta I_2}$		Współczynniki $a_k$ i $b_k$ dla $\tilde{N}_{\Delta I_3}$	
k	$a_k$	$b_k$	$a_k$	$b_k$
0	0.04666385	–	-0.07227473	–
1	0.6986570	0.5764447	-0.01532187	0.9494362
2	-0.1564030	-0.005462326	-0.1475944	-0.1642738
3	0.1058694	0.02893575	0.03216962	0.08744193
4	-0.03452314	0.01390640	0.02435149	-0.04875140
5	0.006722671	-0.02007896	0.007665409	0.003521962
6	–	–	0.03575303	-0.01355053
w:	0.8576045		0.7743006	

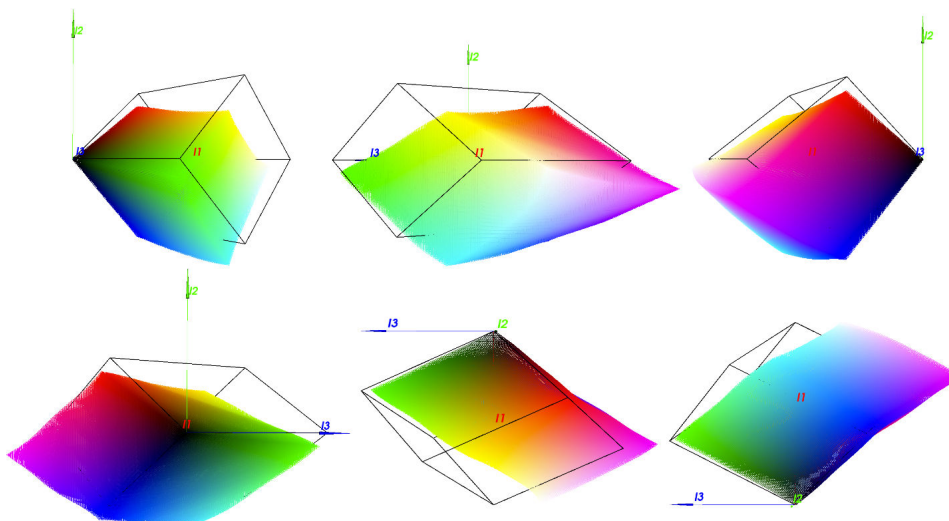


Rysunek 4.14: Aproksymacje funkcji  $N_{\Delta I_2}$  oraz  $N_{\Delta I_3}$  za pomocą szeregów Fouriera. Funkcje  $N$  zostały oznaczone kolorem żółtym, zaś funkcje aproksymujące kolorem fioletowym.

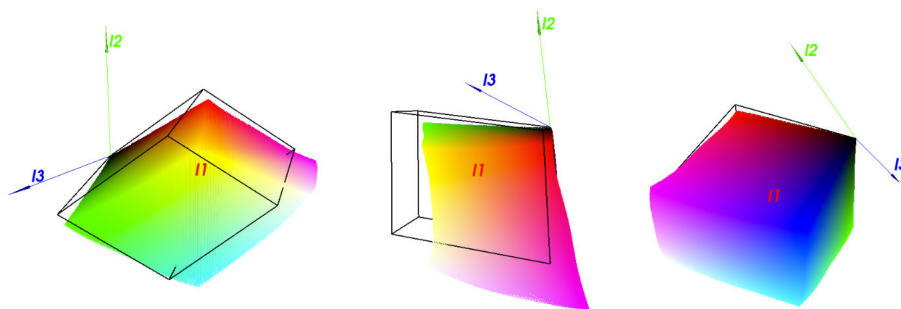
Nadeszła w końcu pora, aby zastosować omawiane przekształcenia  $N_{\Delta I_2}$  i  $N_{\Delta I_3}$  w praktyce. Zaczynając od standardowej przestrzeni Ohty, której przedział osi  $I_1$  to  $[0; 3]$ , dla każdego piksela  $P(i, j)$  obrazu zastosujemy następujące przekształcenia:

$$P(i, j) \begin{cases} I_1 = I_1 \\ I_2 = I_2 - N_{\Delta I_2}(I_1) \\ I_3 = I_3 - N_{\Delta I_3}(I_1) \end{cases} \quad (4.12)$$

W ten sposób uzyskaliśmy przekształcenie przestrzeni Ohty na podstawie danych pozyskanych z ośmiu serii fotografii o różnym naświetleniu. Warto tutaj zwrócić uwagę dlaczego stosujemy odejmowanie, a nie dodawanie. Obserwacje wynikające z serii fotografii mówią, w jakim kierunku następują zmiany chrominancji, czyli jak zmienia się odcień i nasycenie koloru wraz ze zmianą jasności. Aby, podążając wzdłuż osi  $I_1$  (na dowolnej prostej równoległej do niej, przechodzącej przez punkty przestrzeni barw), naśladować te zmiany w sposób właściwy, należy wykonać transformację przestrzeni z przeciwnym znakiem, w przeciwnym wypadku uzyskalibyśmy odwrotną transformację – np. zamiast barw cieplejszych, otrzymalibyśmy barwy chłodniejsze. Pod koniec niniejszego rozdziału następuje powrót do przestrzeni RGB, aby przyjrzeć się bliżej uzyskanej przestrzeni barw w porównaniu do standardowej kostki RGB, a w rozdziale poświęconym dyskusji, porównać uzyskaną przestrzeń barw do innych, powszechnie stosowanych przestrzeni, stosując odpowiednie transformacje z przestrzeni RGB.



Rysunek 4.15: Przedstawienie wygenerowanej przestrzeni barw poprzez rzuty prostopadłe do osi  $I_1$ ,  $I_2$  i  $I_3$ . Czarne kontury reprezentują standardową przestrzeń  $I_1I_2I_3$ .



Rysunek 4.16: Wygenerowana przestrzeń barw widziana z dodatkowych perspektyw.

#### 4.5. Przestrzeń barw $R_TG_TB_T$ jako transformacja z przestrzeni RGB

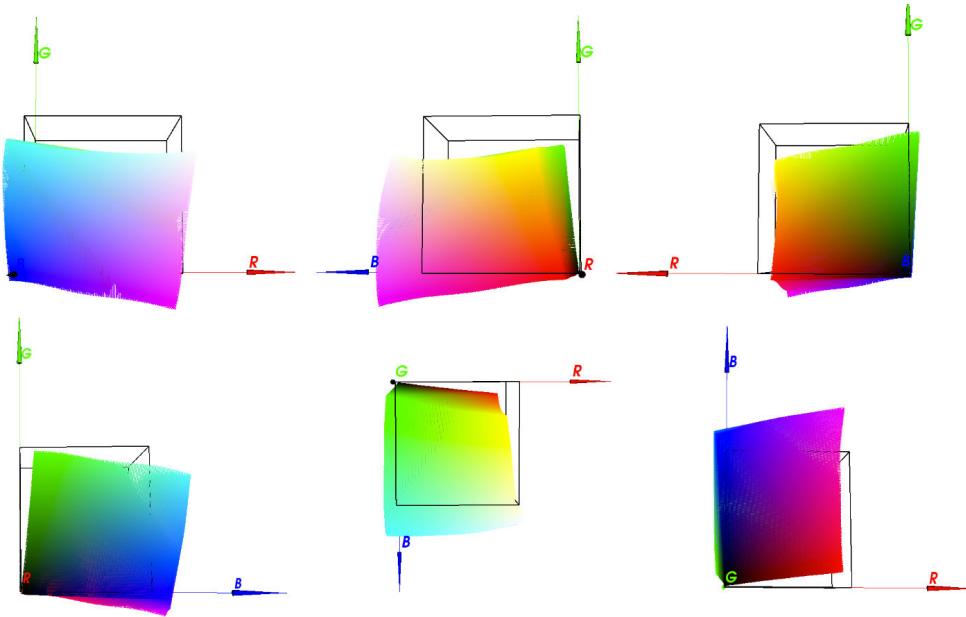
Mając zdefiniowane funkcje  $N_{\Delta I_2}(I_1)$  i  $N_{\Delta I_3}(I_1)$ , możemy łatwo przekształcać przestrzeń Ohty za pomocą obliczonych szeregów Fouriera, jednak aby takiej transformacji dokonać, należy uprzednio przejść z przestrzeni RGB do przestrzeni  $I_1I_2I_3$ . Zdecydowanie lepszym rozwiązaniem byłoby, abyśmy analogiczne funkcje  $\tilde{N}$  uzyskali dla przestrzeni RGB. Wówczas moglibyśmy ją bezpośrednio przekształcać, stosując odpowiednie przesunięcia punktów (kolorów) w przestrzeni, przy pomocy obliczonych wcześniej parametrów  $\Delta I_2$  i  $\Delta I_3$ . Zdefiniuję zatem funkcję  $N_{RGB} : \mathbb{R} \rightarrow \mathbb{R}^3$ , która będzie przekształcała standardową kostkę RGB:

$$N_{RGB}(R + G + B) = \begin{bmatrix} 0 \\ N_{\Delta I_2}(R + G + B) \\ N_{\Delta I_3}(R + G + B) \end{bmatrix} \quad (4.13)$$

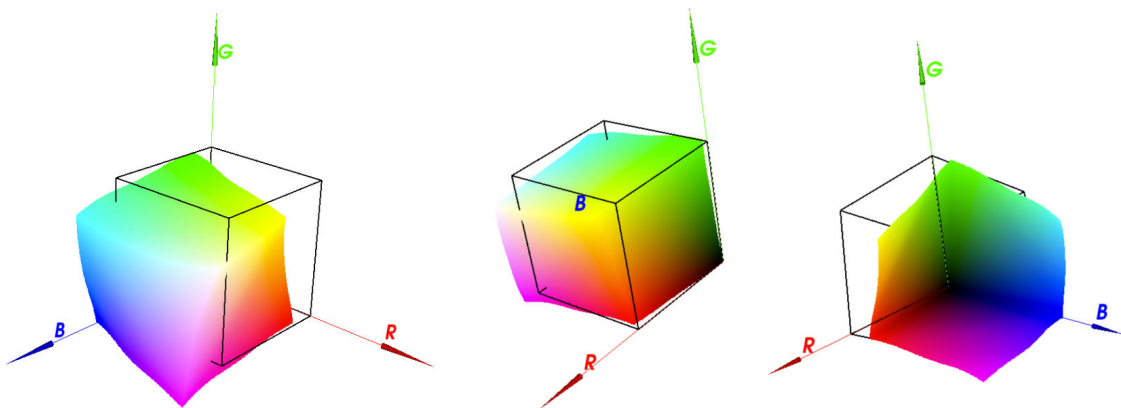
Jeżeli teraz pomnożymy wektor uzyskany w 4.13 przez przekształcenie odwrotne z przestrzeni Ohty (wzór 4.3), otrzymamy wektor przekształcenia dla punktów w przestrzeni RGB:

$$\vec{v}_{RGB} = \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & -\frac{1}{6} \\ \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{2} & -\frac{1}{6} \end{bmatrix} \begin{bmatrix} 0 \\ N_{\Delta I_2}(R+G+B) \\ N_{\Delta I_3}(R+G+B) \end{bmatrix} \quad (4.14)$$

Weźmy teraz standardową kostkę RGB, rozpiętą na przedziałach  $[0; 1]$  na wszystkich osiach, zakładając że każdy punkt w przestrzeni (kolor) traktujemy jako wektor. Wystarczy wówczas od każdego punktu przestrzeni odjąć wektor  $v_{RGB}$ , aby uzyskać przestrzeń  $R_T G_T B_T$ , uwzględniającą obliczone przesunięcia chrominancji.

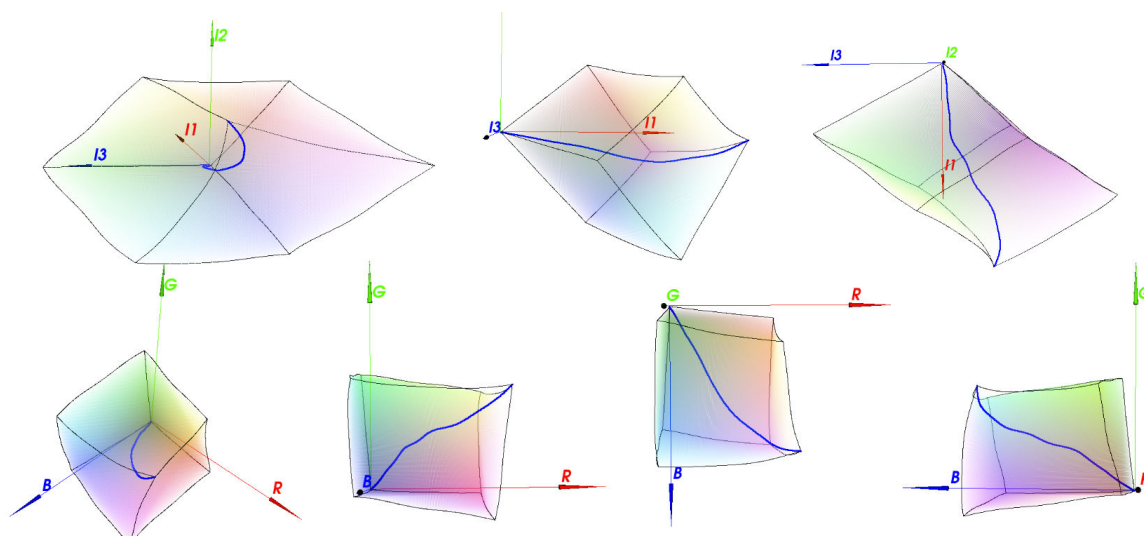


Rysunek 4.17: Przedstawienie wygenerowanej przestrzeni barw  $R_T G_T B_T$  poprzez rzuty prostopadłe do osi R, G i B. Czarne kontury reprezentują standardową kostkę RGB.



Rysunek 4.18: Przestrzeń  $R_T G_T B_T$  widziana z dodatkowych perspektyw.

Na koniec zobaczymy jeszcze jak przebiega krzywa odcieni szarych. Górny rząd odnosi się do przekształconej przestrzeni Ohty, zaś dolny reprezentuje przestrzeń  $R_TG_TB_T$ .



Rysunek 4.19: Krzywa szarości w wygenerowanych przestrzeniach barw (kolor niebieski). Kontury przestrzeni oznaczono kolorem czarnym.

Powyższy krok kończy proces generowania przestrzeni barw na podstawie serii fotografii o różnym naświetleniu. Jak widać przesunięcia w przestrzeni są znaczne, największym jednak problemem jest fakt, że po pierwsze uzyskana przestrzeń posiada punkty o współrzędnych ujemnych, a po drugie część punktów, również w dodatniej dziedzinie, wychodzi poza standardowe przedziały kostki RGB –  $[0; 1]$ . Problem ten jest mało istotny jeśli pracujemy jedynie na przekształconej przestrzeni, bądź wykonujemy jedynie dodawanie lub odejmowanie obrazów, jednak pogłębia się on znacząco, w momencie mnożenia obrazów lub próby zmiany ekspozycji zdjęcia zapisanego w wygenerowanej przestrzeni  $R_TG_TB_T$ .

W następnym rozdziale poświęconym dyskusji problem ten został zaadresowany i zaproponowała została normalizacja przestrzeni, w celu uzyskania pełnej zdolności arytmetycznej – takiej samej jaką ma standardowa kostka RGB. Dyskusja porusza także inne wady oraz zalety przestrzeni  $R_TG_TB_T$ , a sama przestrzeń jest w niej poddawana analizie w praktyce, podczas operacji zmiany ekspozycji zdjęć oraz uzyskiwania zdjęć HDR, z uwzględnieniem zmian chrominancji wraz ze zmianą jasności zdjęcia. Na koniec, przestrzeń barw  $R_TG_TB_T$  została porównana do innych, powszechnie używanych przestrzeni barw, w celu przeniesienia obliczonych zmian chrominancji na inne przestrzenie barw.

# Rozdział 5

## Dyskusja

W przedostatnim rozdziale otrzymana przestrzeń barw jest poddana analizie, omówione zostają jej wady i zalety oraz jest ona normalizowana w celu umożliwienia przeprowadzania poprawnych operacji arytmetycznych na obrazach, w szczególności zmianę ekspozycji i generowanie obrazów HDR. Po procesie normalizacji następuje praktyczna próba wykorzystania przestrzeni  $R_T G_T B_T$ , a następnie zostaje ona porównana z innymi, konwencjonalnymi przestrzeniami barw. W ramach wstępu do niniejszego rozdziału, poruszę temat normalizacji przestrzeni, aby móc użyć ją w kolejnych podrozdziałach, do wykonania różnych operacji arytmetycznych na obrazach.

### 5.1. Problem braku normalizacji

Przyszła pora, aby pochylić się nad otrzymaną przestrzenią barw, poddać ją analizie, sprawdzić jak zachowuje się w praktyce i omówić jej wady i zalety, a także porównać ją z innymi, powszechnie stosowanymi przestrzeniami barw. Na początek rozważmy problem zmiany ekspozycji, kiedy przestrzeń jest nieznormalizowana. Główną cechą stworzonej przestrzeni ma być odpowiednia zmiana chrominancji wraz ze zmianą jasności zdjęć, sprawdźmy zatem jak adaptuje się chrominancja wraz ze zmianą ekspozycji.

Założmy, że wczytujemy zdjęcia w zwykłej przestrzeni RGB. Pierwszym krokiem jaki należy wykonać przed przystąpieniem do obróbki zdjęć, jest transformacja ze zwykłej przestrzeni RGB do przestrzeni  $R_T G_T B_T$ . W tym celu przesuwamy wszystkie punkty przestrzeni RGB, na podstawie wzoru 4.14, odejmując wektor  $\vec{v}_{RGB}$  od każdego punktu w przestrzeni RGB.

Mimo tego przekształcenia, dalej pracujemy układzie RGB, zatem zmianę ekspozycji możemy spróbować obliczyć standardowo:

$$\begin{cases} R'_T = R_T \cdot 2^{\Delta_{EV}} \\ G'_T = G_T \cdot 2^{\Delta_{EV}} \\ B'_T = B_T \cdot 2^{\Delta_{EV}} \end{cases} \quad (5.1)$$

Jednak przestrzeń  $R_T G_T B_T$  zawiera także punkty o współrzędnych ujemnych, a powyższy wzór jest prawdziwy jedynie dla kolorów o współrzędnych dodatnich. Dodatkowo, jeśli zmniejszamy ekspozycję, startując z koloru o współrzędnych dodatnich, nie osiągniemy nigdy koloru, który ma choć jedną współrzędną ujemną, mimo że kolor ten byłby właściwym

odpowiednikiem dla koloru, którego ekspozycję zmieniamy. Na potrzeby tego podrozdziału pozostaniemy jednak przy przestrzeni nieznormalizowanej, aby zobaczyć w praktyce, jak wyglądają zdjęcia – szczególnie te ze zmniejszoną ekspozycją – które z racji opisywanego problemu ze zmianą znaku, przyjmują nieodpowiednie kolory. Nie jest to problem znaczący dla jaśniejszych zdjęć, jednak jeśli chcielibyśmy zmniejszyć ekspozycję o stosunkowo mały współczynnik (np.  $-4$  i mniejszy), zamiast zdjęcia dążącego do czerni, uzyskamy ciemne zdjęcie z granatowymi odcieniami.



Rysunek 5.1: Zmiana odcienia barwy szarej (50% jasności), wraz ze zmianą ekspozycji o  $\frac{1}{3}EV$ . Zdjęcie neutralne oznaczono literą „s”.



Rysunek 5.2: Zmiana ekspozycji zdjęcia neutralnego z serii nr 5 w przestrzeni  $R_T G_T B_T$  bez normalizacji.

O ile spore zmiany balansu bieli są naturalne dla omawianej przestrzeni, o tyle brak możliwości wykonywania poprawnych obliczeń arytmetycznych jest nie do zaakceptowania. Aby wyeliminować ten problem, w następnym kroku przestrzeń ta jest poddana procesowi normalizacji do przedziałów  $[0; 1]$  na wszystkich osiach.

## 5.2. Normalizacja do przedziału $[0; 1]^3$

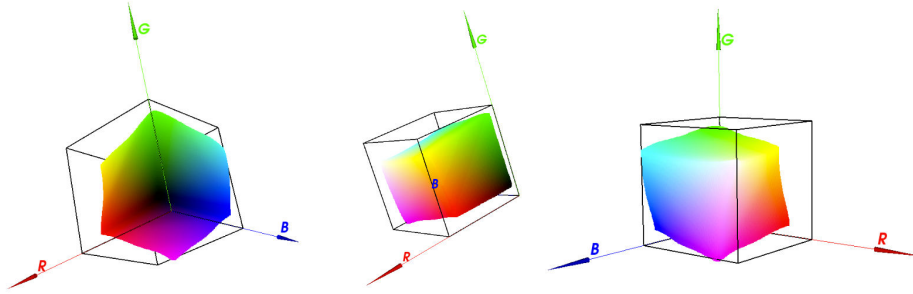
Aby znormalizować przestrzeń  $R_T G_T B_T$  w taki sposób, aby wypełniła całą kostkę RGB, jednocześnie zachowując zmiany chrominancji i proporcje krzywej szarości, normalizację należało wykonać w kilku krokach. Przede wszystkim trzeba było przeskalować i przesunąć punkty przestrzeni tak, aby te mieściły się w granicach kostki RGB. Przestrzeń musiała zostać przeskalowana jednorodnie tak, aby w pełni zmieściła się w kostce rozpiętej na przedziałach  $[0; 1]$ , jednocześnie co najmniej jedna oś musiała w pełni wypełniać oś R, G lub B. Należało zatem obliczyć wartość minimalną ( $r_{min}$ ,  $g_{min}$  i  $b_{min}$ ) oraz maksymalną ( $r_{max}$ ,  $g_{max}$  i  $b_{max}$ ) dla każdej z osi RGB, a następnie objąć wartości minimalne od maksymalnych, uzyskując szerokość przedziału dla każdej z osi. Wówczas największą rozpiętość uzyskujemy dla osi B, oznaczmy ten współczynnik jako  $s$ :

$$s = b_{max} - b_{min} = 1.29806 - 0.00840835 = 1.28965165$$

W następnym kroku należy przeskalować każdy punkt przestrzeni jednorodnie przez ten współczynnik, po czym od każdego punktu odjąć wektor przesunięcia, obliczony jako iloraz uprzednio obliczonych wartości minimalnych przez współczynnik  $s$ . Poniższy wzór podsumowuje dotychczasowe przekształcenia przestrzeni  $R_T G_T B_T$ :

$$\begin{cases} R'_T = \frac{R_T}{s} - \frac{r_{min}}{s} \\ G'_T = \frac{G_T}{s} - \frac{g_{min}}{s} \\ B'_T = \frac{B_T}{s} - \frac{b_{min}}{s} \end{cases} \quad (5.2)$$

Rysunek 5.3 prezentuje częściowo znormalizowaną przestrzeń, której wszystkie punkty mieszczą się w przedziale  $[0; 1]^3$ .



Rysunek 5.3: Częściowa normalizacja przestrzeni  $R_T G_T B_T$ .

Ostatnim krokiem normalizacji jest dystrybucja punktów wewnątrz całej kostki RGB, z zachowaniem krzywej szarości i zmian chrominancji. Weźmy dla przykładu kolor  $A = (1.0, 0.5, 0.0)_{RGB}$ . Wówczas, stosując przekształcenia do i z przestrzeni Ohty obliczymy, że kolor ma ten sam poziom jasności, co odcień szarości  $B = (0.5, 0.5, 0.5)_{RGB}$ . Na koniec przekształćmy te kolory najpierw wg wzoru 4.14, a następnie wzoru 5.2, aby uzyskać kolory  $A'$  i  $B'$  w częściowo znormalizowanej przestrzeni  $R'_T G'_T B'_T$ . Załóżmy, dla prostoty obliczeń, że wówczas kolor  $A'$  będzie miał swój szary odpowiednik  $B'$  (o takiej samej luminancji), taki że kolor  $B'$  przyjmie na osi R wartość 0.5 dla  $R = 0.7$ .

Można podejść do tego zagadnienia liniowo – punkty poniżej 0.7 dystrybuować rzadziej, a powyżej 0.7 gęściej, ale tak, żeby rozdystrybuować wszystkie poziomy na danej osi, zachowując punkt szarości w  $R = 0.7$ . Jednak chcę zaproponować nieliniowy sposób na dystrybucję punktów, w postaci funkcji  $y = x^n$ . Nawiązując do przykładu, obliczmy potęgę  $n$ , która zdefiniuje funkcję  $y$ , podstawiając za  $y$  i  $x$  odpowiednio 0.5 (poziom szarości) i 0.7 (tam, gdzie ten poziom występuje na osi R):

$$\begin{aligned} y &= x^n \\ 0.5 &= 0.7^n \\ \log 0.5 &= n \cdot \log 0.7 \\ n &= \frac{\log 0.5}{\log 0.7} \end{aligned} \quad (5.3)$$

Takie funkcje należy zdefiniować dla każdego punktu krzywej szarości, otrzymując po trzy funkcje dystrybucji kolorów (w dziedzinie osi R, G oraz B) dla każdego poziomu szarości (w zależności od stosowanej głębi kolorów). Należy zauważyć, że punkty krzywej szarości bliskie wartościom 0 lub 1 na dowolnej z osi RGB, będą dawać potęgę  $n$  odpowiednio dążącą do nieskończoności albo do 0. Należy zatem ograniczyć dziedzinę zmiennej  $n$  do rozsądnego

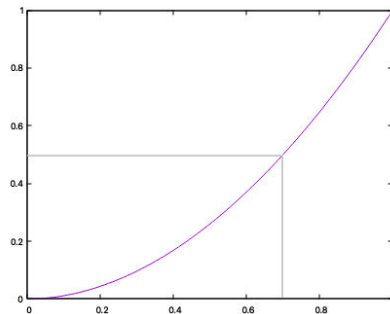


przedziału, zachowującego nieliniowość dystrybucji punktów wraz z rozsądną równomiernością tej dystrybucji. Proponuję, aby zmienną  $n$  ograniczyć do przedziału  $[0.5; 5]$ .

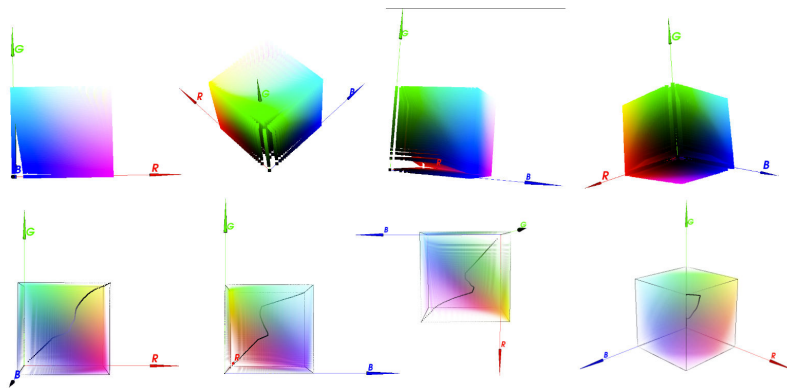
Zdefiniujmy zatem funkcje  $y_r$ ,  $y_g$  i  $y_b$ , które według poniższego schematu będą dystrybuować kolory na poszczególnych osiach, w zależności od poziomów jasności tych kolorów:

1. Weź dowolny kolor  $(r, g, b)$  w przestrzeni RGB;
2. Przekształć przy pomocy wzoru 4.2 kolor na przestrzeń Ohty, wyzeruj wartości  $I_2$  i  $I_3$ , a następnie powróć do przestrzeni RGB (wzór 4.3);
3. Przekształć otrzymany w kroku drugim kolor do znormalizowanej przestrzeni  $R_T G_T B_T$ ;
4. Oblicz potęgi  $n_r$ ,  $n_g$  oraz  $n_b$  funkcji  $y_r$ ,  $y_g$  i  $y_b$  na podstawie wzoru 5.3;
5. Jeśli potęga  $n$  jest mniejsza od 0.5, przyjmij wartość 0.5, jeśli zaś większa od 5, przyjmij wartość 5;
6. Na koniec podnieś składowe koloru  $(r, g, b)$  do potęg  $n$ , poszczególnych funkcji  $y$ :

$$\begin{cases} r' = r^{n_r} \\ g' = g^{n_g} \\ b' = b^{n_b} \end{cases} \quad (5.4)$$



Rysunek 5.4: Funkcja dystrybucji punktów na osi R dla powyższego przykładu.



Rysunek 5.5: Widoki znormalizowanej przestrzeni  $R_T G_T B_T$  oraz przebieg krzywej odcieni szarości.

W ten sposób uzyskaliśmy w pełni znormalizowaną przestrzeń, która rozpina się na przedziale  $[0; 1]^3$ , wypełniając całą kostkę RGB, przy zachowaniu obliczonych wcześniej zmian chrominancji. Rysunek 5.5 ilustruje znormalizowaną przestrzeń  $R_T G_T B_T$  oraz przedstawia końcową krzywą szarości.

### 5.3. Zmiana ekspozycji zdjęć w znormalizowanej przestrzeni $R_T G_T B_T$

Pora na zastosowanie omawianej przestrzeni barw w praktyce. W tym podrozdziale opisany został proces zmiany ekspozycji zdjęć reprezentowanych w omawianej przestrzeni barw. Wzór 5.1 prezentuje sposób na zmianę ekspozycji zdjęć zapisanych w standardowej przestrzeni RGB. Wymaga on, aby kolory były rozprowadzone liniowo<sup>1</sup>, natomiast przestrzeń  $R_T B_T G_T$  została przekształcona przy pomocy serii funkcji jednomianowych, zatem liniową z pewnością nie jest. Należy zatem wprowadzić korekcję wzoru na obliczanie zmiany ekspozycji zdjęć, uwzględniając nieliniowość przestrzeni, wynikającą ze zmian chrominancji wzdłuż krzywej szarości.

Obliczmy dla każdej z osi RGB średnią z potęg  $n$  jednomianów  $y_r, y_g$  i  $y_b$ . Aby obliczenia dawały lepsze wyniki, sumując podniosmy potęgi do kwadratu, a następnie spierwiastkujemy sumę. Dla badanej przestrzeni barw współczynniki te przyjmują wartości:  $n_r \approx 0.953515$ ,  $n_g \approx 0.916607$  oraz  $n_b \approx 1.77676$ .

Możemy teraz zmodyfikować wzór 5.1, aby uzyskać lepszy przepis na zmianę ekspozycji zdjęć w przestrzeni  $R_T G_T B_T$ :

$$\begin{cases} R'_T = R_T \cdot 2^{\frac{\Delta EV}{n_r}} \\ G'_T = G_T \cdot 2^{\frac{\Delta EV}{n_g}} \\ B'_T = B_T \cdot 2^{\frac{\Delta EV}{n_b}} \end{cases} \quad (5.5)$$

Sprawdźmy zatem jak sprawdzają się omawiane przekształcenia w praktyce. Poniższe rysunki prezentują zestawienia zdjęć poddanych zmianie ekspozycji wraz z uwzględnieniem

<sup>1</sup>Oczywiście przestrzeń sRGB nie jest liniowa, jednak nasz system wzrokowy postrzega ją jako taką.

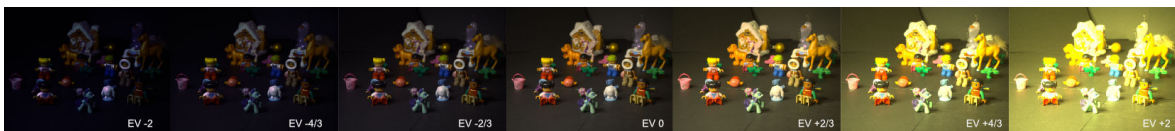
zmian chrominancji.



Rysunek 5.6: Zmiana ekspozycji zdjęcia z serii nr 1 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.7: Zmiana ekspozycji zdjęcia z serii nr 2 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.8: Zmiana ekspozycji zdjęcia z serii nr 3 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.9: Zmiana ekspozycji zdjęcia z serii nr 4 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.10: Zmiana ekspozycji zdjęcia z serii nr 5 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.11: Zmiana ekspozycji zdjęcia z serii nr 6 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.12: Zmiana ekspozycji zdjęcia z serii nr 7 w znormalizowanej przestrzeni  $R_T G_T B_T$ .



Rysunek 5.13: Zmiana ekspozycji zdjęcia z serii nr 8 w znormalizowanej przestrzeni  $R_T G_T B_T$ .

Prześledźmy teraz cechy uzyskanej przestrzeni, wyliczając widoczne wady i zalety. Pierwszą rzeczą, jaka od razu rzuca się w oczy, jest znacząca zmiana balansu bieli na zdjęciach o zmienionej ekspozycji. Jeśli zwiększamy ekspozycję, temperatura barwowa maleje, uzyskujemy wówczas cieplejsze zdjęcia. Odwrotna sytuacja występuje w przypadku zmniejszania ekspozycji. Tutaj zdjęcia ulegają ochłodzeniu, jednak w balans bieli wkrada się delikatnie kolor fioletowy – zdjęcia nie uzyskują wyłącznie niebieskiego odcienia. Zmiany te są jak najbardziej zgodne z oczekiwaniami – jeśli spojrzymy na to jak zmienia się wygenerowana przestrzeń wzdłuż osi układu (rysunki 4.15 i 4.17), zauważymy że takie właśnie zmiany temperatury barwowej zdjęć i balansu bieli powinny mieć miejsce.

O ile zmiany chrominancji zależą wyłącznie od poziomu jasności, największe zmiany (wizualnie) widać na odcieniach szarości. Problem ten wynika z faktu, że podczas obliczeń zmian chrominancji w rozdziale 4, obliczana była jedynie średnia parametrów  $\Delta I_2$  i  $\Delta I_3$  dla każdego poziomu jasności  $I_1$ . Można by badaną przestrzeń zmodyfikować w taki sposób, że parametr zmiany chrominancji zależałby nie tylko od aktualnego poziomu jasności, ale także od saturacji barwy. Wówczas można spróbować uzyskać przestrzeń, w której barwy szare są bardziej odporne na zmiany chrominancji, przez co przestrzeń taka dawałaby rezultaty bliższe rzeczywistości rejestrowanym obrazom.

Kolejną ważną cechą omawianej przestrzeni jest jej nieliniowość, a także sposób dystrybucji kolorów wzdłuż osi RGB przy pomocy funkcji jednomianowych, zależnych od krzywej szarości. Początkowo uzyskana przestrzeń była nieznormalizowana, przez co obliczenia arytmetyczne na obrazach dawały błędne wyniki. Po serii przekształceń udało się jednak uzyskać znormalizowaną przestrzeń, reprezentowaną w przedziałach standardowej kostki RGB, ale też taką, która zachowuje przebieg krzywej szarości, a co za tym idzie uwzględnia obliczone wcześniej zmiany chrominancji.

W końcu warto pomyśleć o artystycznym aspekcie takiej przestrzeni. Z pewnością daje ona spore możliwości uzyskiwania bardziej realistycznych zdjęć, symulujących różne efekty oświetleniowe. Zdjęcia w których zmniejszamy ekspozycję, poprzez zyskiwanie odcieni fioletowych i granatowych, dobrze imitują zdjęcia robione nocą, zaś zdjęcia w których ekspozycję zwiększamy, mogą dla przykładu symulować fotografie wykonane podczas zachodu słońca.

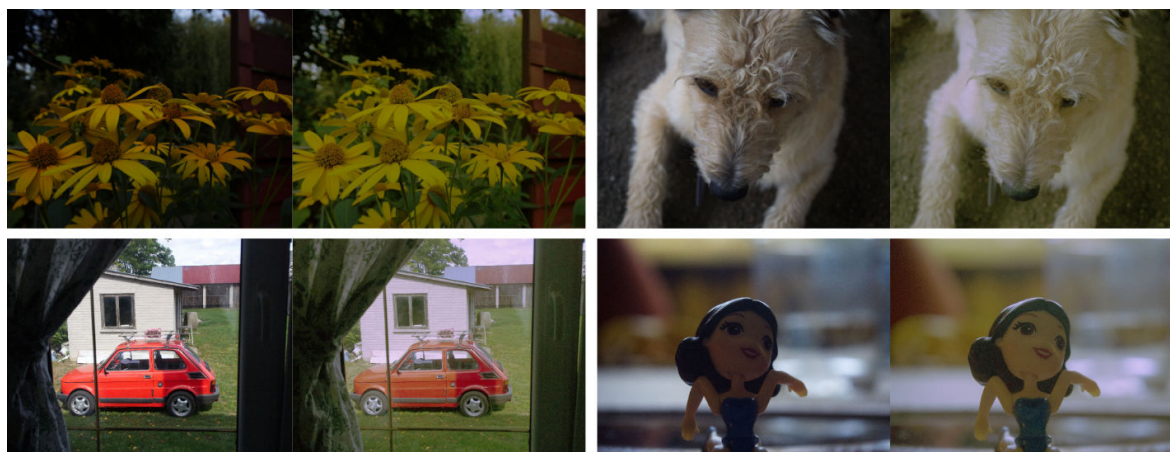
Należy także wziąć po uwagę jakość użytego sprzętu. O ile sam aparat, jak i lampa wykorzystana w procesie badawczym, są urządzeniami dobrej jakości i stworzyły solidne fundamenty do przeprowadzenia badań, z pewnością nie był to sprzęt istic laboratoryjny. Zdecydowanie taki sam eksperyment należałoby przeprowadzić z profesjonalnym sprzętem w idealnej ciemni.

W następnym podrozdziale poruszony został temat fotografii HDR, gdzie otrzymana przestrzeń barw  $R_T G_T B_T$  została użyta w celu wygenerowania zdjęć HDR, z zachowaniem zmian chrominancji przy zmianie ekspozycji.

## 5.4. Obrazy HDR z adaptacją chrominancji

**HDR (High Dynamic Range)** to technika pozyskiwania obrazów w szerokim zakresie barw, szczególnie w dziedzinie jasności. Polega ona na wykonywaniu serii zdjęć, z których część jest prześwietlona, część niedoświetlona, a następnie taka seria jest składana w całość, dzięki czemu uzyskuje się zdjęcia równomiernie oświetlone, bez prześwietleń i niedoświetleń. O ile HDR polega na wykonywaniu serii zdjęć, niniejszy podrozdział traktuje o generowaniu zdjęcia HDR wyłącznie z jednej fotografii. Chodzi mianowicie o to, aby mając zdjęcie neutralne, poddać je zmianom ekspozycji (w stronę prześwietleń i niedoświetleń), stosując wygenerowaną przestrzeń, a co za tym idzie adaptując odpowiednio chrominancję do zmian luminancji.

Weźmy zatem dowolne zdjęcie o neutralnym naświetleniu i poddamy je serii zmian ekspozycji (podobnie jak na rysunku 5.3), po czym zsumujemy obrazy i obliczymy średnią arytmetyczną. Zaczynamy od obliczania zdjęć niedoświetlonych, przez neutralne, aż do zdjęć prześwietlonych, w zakresie  $\pm 2EV$  z krokiem  $\frac{1}{3}EV$ .



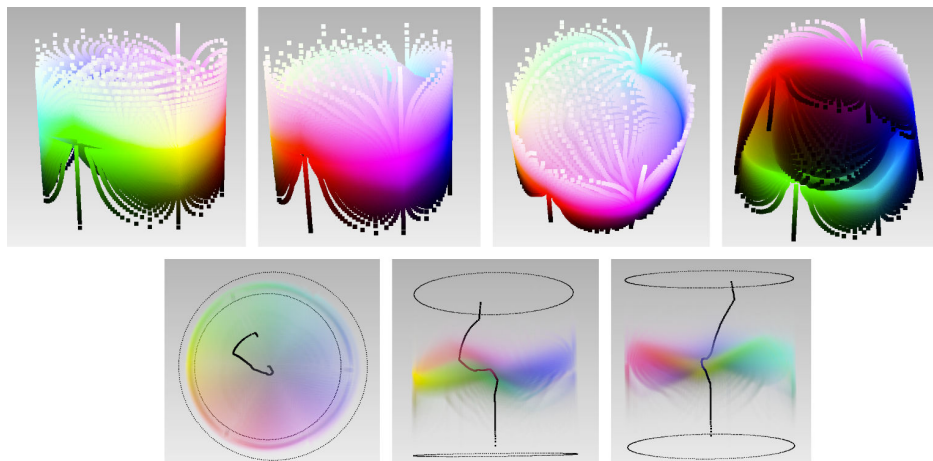
Rysunek 5.14: Przykłady uzyskanych zdjęć HDR (strona prawa) na podstawie zdjęć oryginalnych (strona lewa).

Widać wyraźnie, że prześwietlenia i niedoświetlenia zostały zmniejszone, jednak można zauważyć, że odcienie szarości (w szczególności biel) uzyskały fioletowo-granatowy odcień. Problem wynika z tego, że podczas zmniejszania ekspozycji waga kolorów chłodnych jest zbyt duża w stosunku do wagi kolorów ciepłych, w przypadku zwiększania ekspozycji. W ostatniej sekcji dyskusji, znormalizowana przestrzeń  $R_T G_T B_T$  została zaprojektowana na kilka po-

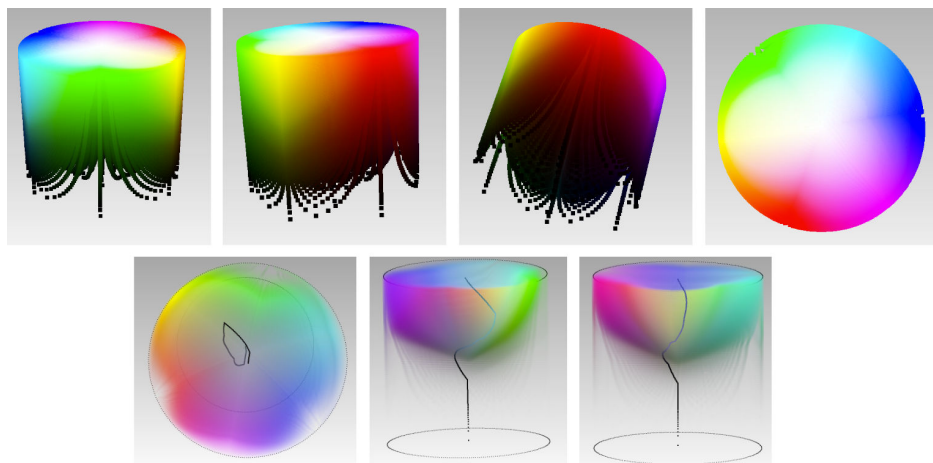
wszecznie stosowanych przestrzeni barw, w celu przyjrzenia się dystrybucjom kolorów obraz krzywym szarości.

## 5.5. Projektacja na konwencjonalne przestrzenie barw

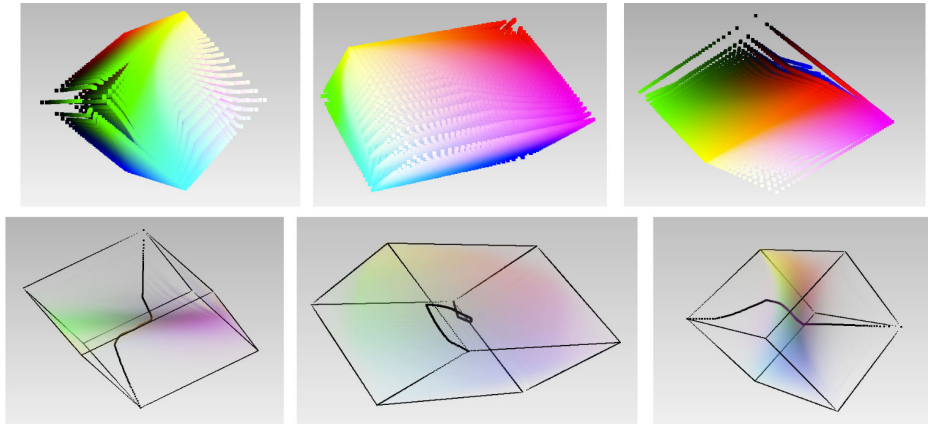
W ramach zakończenia dyskusji przeprowadźmy projekcję uzyskanej przestrzeni barw na standardowe, powszechnie stosowane przestrzenie barw, stosując wzory na przekształcenia przestrzeni RGB na inne przestrzenie z [5].



Rysunek 5.15: Projektacja przestrzeni  $R_TG_TB_T$  na przestrzeń HSL. Dolne rysunki przedstawiają przebieg krzywej szarości.



Rysunek 5.16: Projektacja przestrzeni  $R_TG_TB_T$  na przestrzeń HSV. Dolne rysunki przedstawiają przebieg krzywej szarości.



Rysunek 5.17: Projekcja przestrzeni  $R_T G_T B_T$  na przestrzeń Ohty. Dolne rysunki przedstawiają przebieg krzywej szarości.

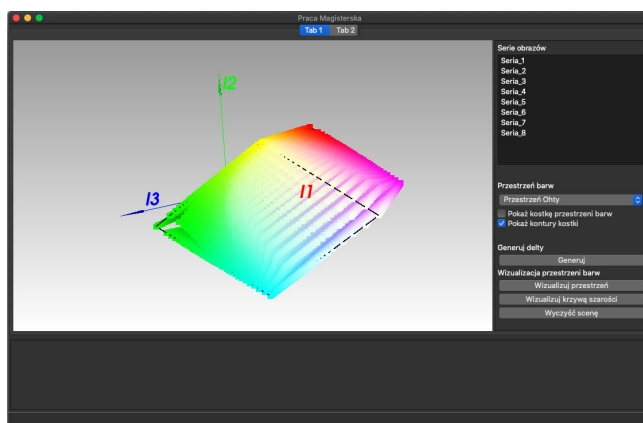
Powyzszym zestawieniem konczę treść niniejszej pracy. Ostatni rozdział podsumuje przebieg pracy, poczawszy od procesu rejestracji zdjec, poprzez ich wywoływanie, generowanie przestrzeni barw, a na podsumowaniu cech wygenerowanej przestrzeni koncząc.

## Rozdział 6

# Podsumowanie

Przebieg mojej pracy rozpoczął się od głębszego poznania budowy i działania aparatów cyfrowych, sposobu rejestracji i przechowywania zdjęć, a także samych przestrzeni barwowych. Celem pracy było utworzenie przestrzeni barw na podstawie serii fotografii o różnym naświetleniu, poprzez zbadanie wszystkich serii zdjęć pod kątem zmian chrominancji wraz ze zmianą czasu naświetlania matrycy aparatu. Przystępując do rejestracji zdjęć, ciężko było na początku przewidzieć jak wielkie te zmiany będą i w jakim kierunku będą przebiegać, nie stawiałem więc żadnej konkretnej tezy, poza tą ogólną, mówiącą, że takie zmiany będą.

Pierwszy rozdział miał na celu wprowadzenie zarówno mnie, jako autora pracy, jak i czytelnika w tematykę pracy. Tematyka ta okazała się być stosunkowo obszerna – aby w pełni ją zrozumieć należy posiąść wiedzę z dziedziny fotografii, przestrzeni barwowych, przetwarzania obrazów oraz należy zrozumieć w jaki sposób „widzi” kamera i w jaki sposób zapisuje surowe dane. W rozdziale tym zostały wprowadzone podstawowe pojęcia, a także scharakteryzowane zostało krótko użyte i napisane przeze mnie oprogramowanie. Największym z programów był program służący do wizualizacji i generowania przestrzeni barw, napisany w oparciu o biblioteki Qt i VTK w języku C++. Jego zadaniem było także obliczanie zmian chrominancji między parami sąsiednich zdjęć z serii, a także sterowanie przebiegiem tych operacji poprzez wygodny interfejs użytkownika.



Rysunek 6.1: Główne oprogramowanie napisane na potrzeby niniejszej pracy.

Kolejnym programem o kluczowej funkcjonalności był program napisany w języku C++, w oparciu o bibliotekę OpenCV. Za jego pomocą przeprowadzony był proces wywoływania



zdjęć, a także zmiany ekspozycji oraz generowania obrazów HDR. Dodatkowo zostało napisanych kilka krótkich programów w języku Python, służących do przetwarzania otrzymanych danych, eksportowania wyników do plików tekstowych, a także generowania szeregów Fouriera na podstawie obliczonych funkcji  $N$ , w rozdziale czwartym. Dodatkowo zostało użytych kilka bibliotek oraz programów, w celu generowania wykresów oraz eksportowania plików ORF do surowych, monochromatycznych, 16-bitowych plików TIFF.

Pliki te stanowiły punkt wejściowy do rozdziału drugiego, gdzie został dokładnie opisany proces cyfrowego wywoływania zdjęć. Rozdział drugi został celowo umieszczony przed rozdziałem traktującym o wykonywaniu serii zdjęć, żeby na początku skupić się na bardziej teoretycznych zagadnieniach, a w dalszej kolejności zająć się jedynie rejestracją zdjęć, bez obawy o przebieg ich wywoływania. Proces cyfrowej, wstępnej obróbki surowych zdjęć okazał się dosyć skomplikowany i początkowo przysporzył mi wiele problemów, gdzie największym z nich był fakt, że surowe pliki ORF firmy Olympus, kompresują bezstratnie dane obrazowe. W wyniku braku możliwości znalezienia algorytmu, jakim te dane są kompresowane, postanowiłem wspomóc swoje prace programem dcrw. Dzięki niemu udało mi się zdekompresować dane obrazowe do postaci monochromatycznych plików TIFF. Dalsze procesy wywoływania obejmowały demozaikowanie, czyli interpolowanie brakujących składowych RGB na monochromatycznym zapisie matrycy światłoczułej, konwersję z przestrzeni barw aparatu na liniową przestrzeń RGB, aż w końcu uzyskanie przestrzeni sRGB, służącej do prezentacji zarejestrowanych serii zdjęć i wyników.

Mając teoretyczny i praktyczny wstęp w proces wstępnej obróbki surowych zdjęć, mogłem przystąpić do rejestracji serii zdjęć. Zważywszy na liczbę wielobarwnych eksponatów, jakimi służyłem podczas ich rejestracji, postanowiłem wykonać 8 serii zdjęć, prezentujących obiekty, które dawały szeroki zakres chrominancji i jasności rejestrowanych barw. Proces ten był potraktowany przeze mnie bardzo precyzyjnie, bowiem stanowił on bardzo ważne podłoże do badań. Obrazy musiały być zarejestrowane starannie, parametry aparatu i oświetlenia musiały być pod kontrolą, a o przesunięciach obrazu między rejestrowanymi scenami nie mogło być mowy. Dzięki zastosowaniu statywu oraz sterowaniu aparatem poprzez moduł WiFi, udało mi się uniknąć jakichkolwiek drgań aparatu podczas wykonywania serii zdjęć, a jedynym zmiennym, kontrolowanym przeze mnie parametrem był czas naświetlania matrycy. Zarejestrowawszy 8 serii zdjęć, byłem gotów do ich wywołania na podstawie procesu opisanego w rozdziale drugim i generowania na ich podstawie przestrzeni barw.

Rozdział czwarty rozpocząłem od poruszenia problemu przestrzeni skorelowanych i zaproponowałem, aby generowanie przestrzeni odbywało się na podstawie przestrzeni Ohty, jako że przestrzeń ta jest nieskorelowana (oddziela składowe chrominancji od luminancji) oraz jej przekształcenie z oraz do przestrzeni RGB jest łatwe i ma charakter liniowy. W dalszej kolejności został zdefiniowany schemat badania zmienności chrominancji. Zdefiniowane zostały funkcje, które reprezentują te zmiany, następnie podany został algorytm, za pomocą którego zostały obliczone zmiany chrominancji w seriach fotografii. Obliczenia te rozpoczynałem zawsze od zdjęcia najciemniejszego, porównując je z kolejnym w serii, kierując się w stronę zdjęć najbardziej prześwieconych. Niosło to za sobą pewne konsekwencje, mianowicie dla zdjęcia najciemniejszego nie były obliczane zmiany chrominancji, co skutkowało tym, że punkty w generowanej przestrzeni w okolicy początku układu współrzędnych były najmniej przekształcone. Zastosowałem taką metodę, ponieważ na parach zdjęć najmniej doświetlonych zmiany chrominancji były najmniejsze, przez to rozpoczęcie badania od zdjęcia najciemniejszego powodowało najmniejsze błędy obliczeniowe, szczególnie w porównaniu do

rozpoczęcia obliczeń w serii od zdjęcia neutralnego w dwóch kierunkach – w stronę zdjęć prześwietlonych i niedoświetlonych. W następnym kroku obliczone zmiany zostały uśrednione, w celu wygenerowania jednej, spójnej przestrzeni barw. Obliczone funkcje zmian chrominancji były różniczkami – opisywały jak zmienia się funkcja przekształcająca osie  $I_2$  i  $I_3$ , zatem w następnym kroku funkcje te zostały scałkowane, dając funkcje  $F_{\Delta I_2}(I_1)$  oraz  $F_{\Delta I_3}(I_1)$ , które opisywały jak należy przekształcać osie  $I_2$  i  $I_3$ , w zależności od parametru luminancji  $I_1$ , aby uzyskać przestrzeń barw uwzględniającą obliczone zmiany chrominancji. Funkcje te były obliczone dla poziomów  $I_1$  na przedziale  $[0; 3 \cdot (2^{16} - 1)]$ , zatem w następnym kroku zostały znormalizowane do przedziału  $[0; 3]$ , aby móc je wykorzystać do przekształcenia obliczonej przestrzeni na kostkę RGB rozpiętą na przedziale  $[0; 1]^3$ . W kolejnym kroku znormalizowane funkcje zostały zaproksymowane szeregami Fouriera, w celu usunięcia szumów, a także prostszej dystrybucji tych funkcji – bez tego przekształcenia, aby zdefiniować każdą z funkcji  $F$ , należałoby podać  $2 \cdot 2^{16}$  parametrów typu zmiennopozycyjnego o podwójnej precyzji. Po aproksymacji liczba tych parametrów zmalała do 26, a sama aproksymacja dała dobre rezultaty. Na koniec rozdziału czwartego zostało zdefiniowane przekształcenie bezpośrednio z przestrzeni RGB do wygenerowanej przestrzeni  $R_T G_T B_T$ , które niwelowało konieczność uprzedniej konwersji do przestrzeni Ohty. Dzięki temu można wczytywać fotografie zapisane w przestrzeni RGB i od razu przenosić je do przestrzeni  $R_T G_T B_T$ , przy pomocy jednej funkcji.

Rozdział piąty poświęcony został dyskusji, ale był także niejako kontynuacją rozdziału czwartego. Jako pierwszy został zaadresowany problem braku normalizacji uzyskanej przestrzeni. Problem polegał na tym, że punkty przestrzeni wychodziły poza krańce kostki RGB rozpiętej na przedziale  $[0; 1]^3$ , a w szczególności na tym, że przestrzeń zawierała punkty o współrzędnych ujemnych. Powodowało to niemożność przeprowadzania prawidłowych operacji na obrazach zapisanych w tej przestrzeni, np. mnożenia obrazów oraz zmiany ekspozycji zdjęć. Problem ten został rozwiązany poprzez odpowiednie skalowanie, przesunięcie, a następnie zaproponowanie dystrybucji kolorów poprzez funkcje jednomianowe o zmiennych potęgach, zależnych od położenia punktów szarości w przestrzeni. Uzyskana wówczas przestrzeń była znormalizowana do przedziałów  $[0; 1]$  oraz zachowywała przebieg krzywej szarości i zmiany chrominancji. Dzięki temu można było przejść do wykorzystania przestrzeni  $R_T G_T B_T$  w praktyce, wykonując operacje zmiany ekspozycji zdjęć oraz generując zdjęcia HDR. Na koniec uzyskana przestrzeń została poddana projekcji w celu porównania z przestrzeniami barw HSL, HSV oraz Ohty, gdzie pokazano jak w tych przestrzeniach dystrybuują się barwy, na podstawie wcześniej zdefiniowanych przekształceń oraz jak przebiegają w nich krzywe szarości.

Przeźnięć barw  $R_T G_T B_T$  okazuje się w znaczący sposób zmieniać odcienie kolorów, wraz ze zmianą jasności. Badania dowodzą, że wraz ze zwiększeniem ekspozycji, barwy stają się coraz cieplejsze, kierując się w stronę żółceni. Idąc w drugą stronę – przy scenach niedoświetlonych, barwy idą w kierunku granatu oraz fioletu. Przynosi mi to na myśl skojarzenie z cyklem dnia i nocy, nocą światło jest chłodne, czasem wręcz granatowe, a dniem, szczególnie przy ostrym słońcu, ciepłe i przyjmuje odcienie żółci i czerwieni. Powstaje jednak pytanie na ile zarejestrowane zmiany wynikają z rzeczywistych zmian dystrybucji światła na rejestrowanych scenach, a na ile uzależnione są one od budowy matrycy aparatu. Niemniej jednak otrzymana przestrzeń stanowi ciekawą bazę, stawiając fundamenty pod budowę przestrzeni barw opartych na rzeczywistych zmianach chrominancji wraz ze zmianą jasności fotografowanej sceny.



# Bibliografia

- [1] Bednarski G., Kosmala J.: *DCRAW – konwersja plików RAW*  
[http://home.agh.edu.pl/~sniecho/prezentacje\\_pdf/Kosmala,Bednarski.pdf](http://home.agh.edu.pl/~sniecho/prezentacje_pdf/Kosmala,Bednarski.pdf),  
dostęp 27.06.2020
- [2] Chung K-H. and Chan Y-H.: *Low-complexity color demosaicing algorithm based on integrated gradients*, J. Electronic Imaging nr 19, 2010
- [3] Chung K-H. and Chan Y-H.: *Color Demosaicing Using Variance of Color Differences*, Image Processing, IEEE Transactions on nr 15, 2006, s. 2944-2955
- [4] Color: Determining a Forward Matrix for Your Camera  
<http://www.strollswithmydog.com/determining-forward-color-matrix/>,  
dostęp 29.06.2020
- [5] Color space transformations,  
<https://pdfs.semanticscholar.org/739d/20ce3cc41ee74f3db7748949abd326899537.pdf>  
dostęp 30.06.2020
- [6] Developing a RAW photo file 'by hand' - Part 1  
<https://www.odelama.com/photo/Developing-a-RAW-Photo-by-hand/>,  
dostęp 29.06.2020
- [7] Developing a RAW photo file 'by hand' - Part 2  
[https://www.odelama.com/photo/Developing-a-RAW-Photo-by-hand/Developing-a-RAW-Photo-by-hand\\_Part-2/](https://www.odelama.com/photo/Developing-a-RAW-Photo-by-hand/Developing-a-RAW-Photo-by-hand_Part-2/),  
dostęp 29.06.2020
- [8] Fourier Series,  
[https://symfit.readthedocs.io/en/stable/examples/ex\\_fourier\\_series.html](https://symfit.readthedocs.io/en/stable/examples/ex_fourier_series.html),  
dostęp 30.06.2020
- [9] Hoddinott R.: *Poradnik fotograficzny*, Galaktyka sp. z o.o., 2013, ISBN 978-83-7579-285-0
- [10] Light sources and illuminants  
[https://www5.konicaminolta.eu/fileadmin/content/eu/Measuring\\_Instruments/4\\_Learning\\_Centre/L\\_D/Light\\_sources\\_and\\_illuminants/Apps\\_Note\\_1\\_-\\_Light\\_sources\\_and\\_illuminants.pdf](https://www5.konicaminolta.eu/fileadmin/content/eu/Measuring_Instruments/4_Learning_Centre/L_D/Light_sources_and_illuminants/Apps_Note_1_-_Light_sources_and_illuminants.pdf),  
dostęp 29.06.2020
- [11] Løv E., Winther R., *Fourier analysis*, Uniwersytet w Oslo, Instytut Matematyki, 2011
- [12] Mokrzycki W. S.: *Wprowadzenie do przetwarzania informacji wizualnej, t. 1: Percepcja, akwizycja, wizualizacja*, Akademicka Oficyna Wydawnicza EXIT, 2010, ISBN 978-83-7837-505-0

- [13] Nguyen R., Brown M. S.: *Why You Should Forget Luminance Conversion and Do Something Better*, National University of Singapore, York University, 2017
- [14] Oficjalna strona biblioteki OpenCV  
<https://opencv.org/>, dostęp 25.06.2020
- [15] Oficjalna strona biblioteki Qt  
<https://www.qt.io/>, dostęp 25.06.2020
- [16] Oficjalna strona biblioteki VTK  
<https://vtk.org/>, dostęp 25.06.2020
- [17] Oficjalna strona programu Adobe DNG Converter  
<https://helpx.adobe.com/pl/photoshop/using/adobe-dng-converter.html>,  
dostęp 25.06.2020
- [18] Oficjalna strona programu dcraw  
<https://www.dechifro.org/dcraw/>, dostęp 25.06.2020
- [19] Oficjalna strona programu ExifTool  
<https://exiftool.org/>, dostęp 25.06.2020
- [20] Oficjalna strona programu gnuplot  
<http://www.gnuplot.info/>, dostęp 25.06.2020
- [21] Poynton C. "*Gamma*" and Its Disguises: *The Nonlinear Mappings of Intensity in Perception, CRTs, Film, and Video*, SMPTE Motion Imaging Journal nr 15, 1993, s. 1099-1008
- [22] RGB/XYZ Matrices  
[http://www.brucelindbloom.com/index.html?Eqn\\_RGB\\_XYZ\\_Matrix.html](http://www.brucelindbloom.com/index.html?Eqn_RGB_XYZ_Matrix.html),  
dostęp 29.06.2020
- [23] Rush A., Hubel P.: *X3 sensor characteristics*, J. Soc. Photogr. Sci. Technol. Japan nr 66, 2003, s. 57–60
- [24] Specyfikacja aparatu Olympus OM-D E-M10 Mark II  
<https://asia.olympus-imaging.com/product/dslr/em10mk2/spec.html>, dostęp 13.05.2020
- [25] Specyfikacja pliku TIFF 6.0  
<https://www.itu.int/itudoc/itu-t/com16/tiff-fx/docs/tiff6.pdf>  
dostęp 30.06.2020
- [26] Visualizing the raw (sensor) highlight clipping  
<https://www.darktable.org/2016/10/raw-overexposed/>, dostęp 29.06.2020
- [27] Yu-Ichi Ohta, *A Region-Oriented Image-Analysis System by Computer*, Kyoto University, 1980
- [28] Zapryanov G., Nikolova I.: *Comparative Study of Demosaicing Algorithms for Bayer and Pseudo-Random Bayer Color Filter Array*, 2008

# Spis tabel

1.1. Fragment specyfikacji aparatu Olympus OM-D E-M10 Mark II oraz obiektywu M.Zuiko Digital ED 14-42mm F3.5-5.6 EZ . . . . .	9
4.1. Współczynniki $a_k$ , $b_k$ oraz $w$ dla szeregu Fouriera aproksymującego przekształcenia osi $I_2$ i $I_3$ . . . . .	40



# Spis rysunków

1.1.	Porównanie parametrów HSL tej samej barwy, zarejestrowanej przy neutralnej (po lewej) i znacznie zwiększonej (po prawej) ekspozycji. Scena naturalna została zarejestrowana z ekspozycją 1/10s, zaś prześwietlona z ekspozycją 2s.	7
1.2.	Aparat użyty w pracy – Olympus OM-D E-M10 Mark II wraz z obiektywem M.Zuiko Digital ED 14-42mm F3.5-5.6 EZ . . . . .	9
1.3.	Przykładowe wyjście programu ExifTool. . . . .	11
2.1.	Przykładowa siatka filtra Bayera o schemacie RGGB. . . . .	16
2.2.	Fragment obrazu po demoaikowaniu, w porównaniu do obrazu surowego. . .	17
2.3.	Porównanie obrazu zaraz po procesie demoaikowania (strona lewa) do końcowego obrazu w przestrzeni sRGB (strona prawa). . . . .	17
2.4.	Porównanie iluminantów D50 i D65. . . . .	18
3.1.	Wybrane zdjęcia z serii numer 1. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A4, A7, niedoświetlone – C5, C10, C15. . . . .	23
3.2.	Wybrane zdjęcia z serii numer 2. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A6, A12, niedoświetlone – C3, C8, C12. . . . .	24
3.3.	Wybrane zdjęcia z serii numer 3. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C7, C13. . . . .	24
3.4.	Wybrane zdjęcia z serii numer 4. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C14. . . . .	25
3.5.	Wybrane zdjęcia z serii numer 5. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A6, A12, niedoświetlone – C3, C8, C13. . . . .	25
3.6.	Wybrane zdjęcia z serii numer 6. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C13. . . . .	26
3.7.	Wybrane zdjęcia z serii numer 7. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C13. . . . .	26
3.8.	Wybrane zdjęcia z serii numer 8. Zdjęcie neutralne widoczne jest po prawej stronie. Widoczne zdjęcia prześwietlone – A1, A5, A10, niedoświetlone – C3, C8, C13. . . . .	27
4.1.	Porównanie klasycznej kostki RGB (po lewej) do przestrzeni Ohty (po prawej).	31



4.2. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 1. . . . .	34
4.3. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 2. . . . .	34
4.4. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 3. . . . .	35
4.5. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 4. . . . .	35
4.6. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 5. . . . .	35
4.7. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 6. . . . .	36
4.8. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 7. . . . .	36
4.9. Zmiany chrominancji na osiach $I_2$ i $I_3$ , w zależności od wartości na osi $I_1$ – seria nr 8. . . . .	36
4.10. Rzut na przestrzeń Ohty prostopadłe do osi $I_1$ i $I_2$ . . . . .	37
4.11. Średnie arytmetyczne parametrów $\Delta I_2$ i $\Delta I_3$ . . . . .	38
4.12. Funkcje $F_{\Delta I_2}$ i $F_{\Delta I_3}$ – właściwe przekształcenia osi $I_2$ i $I_3$ . . . . .	39
4.13. Funkcje $F_{\Delta I_2}$ i $F_{\Delta I_3}$ znormalizowane do przedziału $[0; 3]$ na osi $I_1$ . . . . .	39
4.14. Aproksymacje funkcji $N_{\Delta I_2}$ oraz $N_{\Delta I_3}$ za pomocą szeregów Fouriera. Funkcje $N$ zostały oznaczone kolorem żółtym, zaś funkcje aproksymujące kolorem fioletowym. . . . .	41
4.15. Przedstawienie wygenerowanej przestrzeni barw poprzez rzuty prostopadłe do osi $I_1$ , $I_2$ i $I_3$ . Czarne kontury reprezentują standardową przestrzeń $I_1I_2I_3$ . . . . .	42
4.16. Wygenerowana przestrzeń barw widziana z dodatkowych perspektyw. . . . .	42
4.17. Przedstawienie wygenerowanej przestrzeni barw $R_TG_TB_T$ poprzez rzuty prostopadłe do osi R, G i B. Czarne kontury reprezentują standardową kostkę RGB. . . . .	43
4.18. Przestrzeń $R_TG_TB_T$ widziana z dodatkowych perspektyw. . . . .	43
4.19. Krzywa szarości w wygenerowanych przestrzeniach barw (kolor niebieski). Kontury przestrzeni oznaczono kolorem czarnym. . . . .	44
5.1. Zmiana odcienia barwy szarej (50% jasności), wraz ze zmianą ekspozycji o $\frac{1}{3}EV$ . Zdjęcie neutralne oznaczono literą „s”. . . . .	46
5.2. Zmiana ekspozycji zdjęcia neutralnego z serii nr 5 w przestrzeni $R_TG_TB_T$ bez normalizacji. . . . .	46
5.3. Częściowa normalizacja przestrzeni $R_TG_TB_T$ . . . . .	47
5.4. Funkcja dystrybucji punktów na osi R dla powyższego przykładu. . . . .	48
5.5. Widoki znormalizowanej przestrzeni $R_TG_TB_T$ oraz przebieg krzywej odcieni szarości. . . . .	49
5.6. Zmiana ekspozycji zdjęcia z serii nr 1 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	50
5.7. Zmiana ekspozycji zdjęcia z serii nr 2 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	50
5.8. Zmiana ekspozycji zdjęcia z serii nr 3 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	50
5.9. Zmiana ekspozycji zdjęcia z serii nr 4 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	50
5.10. Zmiana ekspozycji zdjęcia z serii nr 5 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	50
5.11. Zmiana ekspozycji zdjęcia z serii nr 6 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	50
5.12. Zmiana ekspozycji zdjęcia z serii nr 7 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	51
5.13. Zmiana ekspozycji zdjęcia z serii nr 8 w znormalizowanej przestrzeni $R_TG_TB_T$ . . . . .	51

5.14. Przykłady uzyskanych zdjęć HDR (strona prawa) na podstawie zdjęć oryginalnych (strona lewa). . . . .	52
5.15. Projekcja przestrzeni $R_T G_T B_T$ na przestrzeń HSL. Dolne rysunki przedstawiają przebieg krzywej szarości. . . . .	53
5.16. Projekcja przestrzeni $R_T G_T B_T$ na przestrzeń HSV. Dolne rysunki przedstawiają przebieg krzywej szarości. . . . .	53
5.17. Projekcja przestrzeni $R_T G_T B_T$ na przestrzeń Ohty. Dolne rysunki przedstawiają przebieg krzywej szarości. . . . .	54
6.1. Główne oprogramowanie napisane na potrzeby niniejszej pracy. . . . .	55