



**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

---

**Fakulta elektrotechnická  
Katedra radioelektroniky**

**Metody korekce barev a vyvážení bílé**

**Methods of Color Correction and White Ballance**

**Diplomová práce**

Studijní program: Elektronika a komunikace  
Studijní obor: Audiovizuální technika a zpracování signálů

Vedoucí práce: Ing. Stanislav Vítek, Ph.D.

**Bc. Jaroslav Fikr**

---

**Praha 2019**



# Čestné prohlášení

Prohlašuji, že jsem předloženou diplomovou prací vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Datum: 08. 1. 2019

.....

podpis



## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Fikr** Jméno: **Jaroslav** Osobní číslo: **425076**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávající katedra/ústav: **Katedra radioelektroniky**  
Studijní program: **Elektronika a komunikace**  
Studijní obor: **Audiovizuální technika a zpracování signálů**

## II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

**Metody korekce barev a vyvážení bílé**

Název diplomové práce anglicky:

**Methods of Color Correction and White Ballance**

Pokyny pro vypracování:

1. Provedte studii metod korekce barev a vyvážení bílé, včetně metod aplikace strojového učení.
2. Provedte porovnání účinnosti metod korekce barev a vyvážení bílé v laboratorních podmínkách.
3. Navrhněte metodu objektivního posouzení rozdílu kolorimetrických parametrů rozdílně korigovaného barevného obrazu. Metodu otestujte na reálných datech.

Seznam doporučené literatury:

- [1] GIJSENIJ, Arjan, et al. Computational color constancy: Survey and experiments. IEEE Transactions on Image Processing, 2011, 20.9: 2475-2489.  
[2] GIJSENIJ, Arjan; GEVERS, Theo. Color constancy using natural image statistics and scene semantics. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33.4: 687-698.  
[3] AGARWAL, Vivek; GRIBOK, Andrei V.; ABIDI, Mongi A. Machine learning approach to color constancy. Neural Networks, 2007, 20.5: 559-563.

Jméno a pracoviště vedoucí(ho) diplomové práce:

**Ing. Stanislav Vítek, Ph.D., 13137**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **26.09.2018**

Termín odevzdání diplomové práce: **08.01.2018**

Platnost zadání diplomové práce: **19.02.2020**

Ing. Stanislav Vítek, Ph.D.  
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta



# Abstrakt

Tato diplomová práce se zabývá metodami korekce barev a vyvážení bílé. V první části pojednává o základních pojmech obrazové techniky, seznamuje s různými přístupy k vyvážení barev v obraze a poskytuje teoretický základ k pochopení problematiky. Poté se zabývá aplikací některých algoritmů pro korekci barev a vyvážení bílé včetně aplikace algoritmu využívajícího strojového učení. Tyto výstupy vyhodnocuje objektivním testem zaměřeným na věrnost přenosu barev a věrohodné podání odstínů pleti. Následuje diskuse tohoto objektivního vyhodnocení a ukázka aplikace jednotlivých algoritmů na reálná data.

**Klíčová slova:** Vyvážení bílé, korekce barev, barevná stálost, barevný prostor, sémantická maska, CNN, strojové učení, šedý svět, retinex, CNNNSS

# Abstract

The thesis deals with color corection and white balance of image. First part contains some basic terms of image technology, introduces different approaches to color constancy and describes theory of this topic. Second part deals with application of color correcting algorithms and white balance algorithms, including machine learning approach too. Output of these algorithms are evaluated by objective testing which is based on true transfer of color and authentic skin tones evaluation. Discussion of this objective evaluation follows and the thesis ends with showing few samples of corrected imagery from practice.

**Index terms:** White balance, color correction, color constancy, color space, semantic mask, CNN, machine learning, gray world, white patch, retinex, CNNNSS





# Obsah

Úvod .....	11
1 Lidské oko .....	13
2 Kolorimetrie .....	15
2.1 Barevné prostory .....	15
2.1.1 RGB barevný prostor .....	15
2.1.2 XYZ barevný prostor .....	15
2.1.3 HSV barevný prostor .....	16
2.1.4 CIELAB barevný prostor .....	17
2.2 Vyvážení barev .....	18
2.2.1 Barevná stálost .....	18
2.2.2 Vyvážení bílé .....	18
2.3 Přístupy k barevné stálosti .....	20
2.3.1 Gray world .....	21
2.3.2 White patch Retinex .....	22
2.3.3 Multi Scale Retinex .....	23
2.3.3 Machine learning: Semantic White Balance .....	24
3 Hodnocení kvality vyvážení bílé .....	26
3.1 JND a $\Delta E$ .....	26
3.2 Objektivní hodnocení .....	27
3.2.1 Vyhodnocení barev color checkeru .....	27
3.2.2 Vyhodnocení pleťové barvy .....	29
4 Snímání vstupních dat .....	30
5 Korekce barev nafocených dat .....	33
5.1 Vyhodnocení barev color checkeru .....	33
5.1.1 Skript v programu MATLAB .....	33
5.1.2 Výsledky .....	34
5.1.3 Statistické vyhodnocení .....	35
5.1.4 Vyhodnocení v závislosti na vstupu .....	37
5.2 Vyhodnocení pleťové barvy .....	40
5.2.1 Skript v programu MATLAB .....	40
5.2.2 Výsledky .....	41
5.2.3 Vyhodnocení .....	42
5.2.4 Ukázka korekce vybraných fotografií .....	44
6 Aplikace algoritmů na reálná data .....	46
7 Sémantická maska u CNNNSS algoritmu .....	50
8 Závěr .....	53
9 Použitá literatura .....	55
10 Přílohy .....	57

# Seznam příloh

A: MATLAB skript metody Gray world

B: MATLAB skript metody White patch Retinex

C: MATLAB skript metody Multi Scale Retinex

D: MATLAB skript vyhodnocení barev color checkeru

E: MATLAB skript vyhodnocení barev pleťové barvy

F: Fotografie ze snímání testovacích snímků

G: Tabulkové hodnoty použitých grafů

H: CD s textem práce v pdf souboru, kompletními komentovanými MATLAB skripty, fotografiemi a daty z vyhodnocení

# Úvod

Barevné vlastnosti světla osvětlujícího určitou scénu se promítnou do záznamu této scény na jakékoli médium. Lidské oko je oproti záznamovým médiím této skutečnosti přizpůsobivé a tato jeho vlastnost je tedy zapotřebí vzít v úvahu při snímání obrazu. Výsledný obrazový vjem se pro pozorovatele musí jevit přirozený, ne naopak. Korekce barev se tak stává jedním z důležitých kroků při úpravě a snímání obrazu.

Dobře vyvážené barvy ve fotografii ovlivňují subjektivní vjem pozorovatele a vyvážení barev dokáže změnit přenesené emoce. V neprofesionální sféře je žádoucí, aby barvy odpovídaly co nejvíce konkrétní situaci tak, jak ji vnímá přítomný člověk. V umělecké oblasti fotografie a kinematografie je však vyvážení barev výrazovým prostředkem. Vzniká-li dílo nové, je čistě na autorovi, jaký bude mít výsledný vizuál. Problém však může vzniknout při přenosu již hotového díla z nějakého média na jiné. Například tedy při digitalizaci filmových kopií. Provedení správných a přesných barevných korekcí takového přepisu je mnohdy velice finančně a časově náročné. Nejsou-li však tyto korekce provedeny adekvátně, můžou vyvolat silné negativní ohlasy.

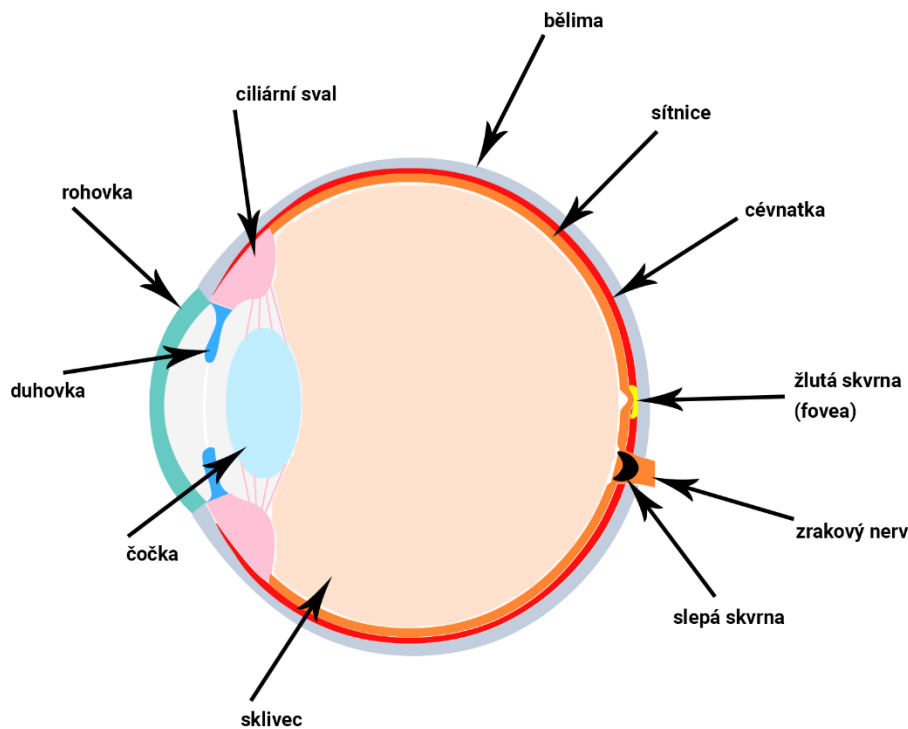
Ideální záznam obrazu v neprofesionální sféře je tedy takový, který přesně odpovídá vjemu oka. To znamená takový, který neutralizuje barevné vlastnosti světla osvětlujícího scénu. Toho je pak dobré dosáhnout automaticky, bez zásahu fotografa/kameramana, neboť jde povětšinou o osoby neorientující se v této problematice.

V profesionální umělecké sféře a sféře užitého umění je pak žádoucí, aby nad výsledným barevným podáním měl kontrolu autor a mohl tak dostat svému uměleckému záměru. Při digitalizaci filmových kopií je pak důležité držet se takového původního uměleckého záměru autora. Ideálním provedením je pak nastavení parametrů dle klíčových scén, které jsou dobře korigované v souladu s autorovým záměrem, a ty pak použít jako vstup pro úpravy celého záznamu.

Tato diplomová práce se zabývá různými druhy automatických barevných korekcí obrazu, diskutuje jejich jednotlivé výhody a nedostatky a objektivně porovnává jejich výstupy na sadě testovacích fotografiích pořízených v laboratorních podmínkách s různě barevnými světelnými zdroji. Zaměřuje se pak na výstupy nového algoritmu využívajícího strojového učení. Testovací algoritmy také aplikuje na reálné skeny filmové kopie a diskutuje výsledky v kontextu s uměleckým záměrem kameramana.

# 1 Lidské oko

Obrazový vjem vnímáme přes oko, a proto je důležité znát, jak funguje. Lidské oko samo o sobě je relativně nedokonalá optická soustava, ale celý zrakový vjem působí mnohem dokonaleji. To je dané zpracováním vjemů z oka v mozku, kde dochází k razantnímu „vylepšení“ výsledného obrazu.



Obr. 1: Schéma lidského oka

Celý proces zobrazení a snímání obrazu scény je velice podobný tomu ve fotoaparátu. Po průchodu světla rohovkou, čočkou a sklivcem, se obraz pozorované scény v lidském oku vytvoří na sítnici (ekvivalent čipu), kde je snímán tyčinkami (monochromatické fotocitlivé elementy) a čípkami (R, G a B fotocitlivé elementy). Množství pronikajícího světla do oka závisí na velikosti pupily, která je ovlivněna duhovkou (clona). Ostření provádí ciliární svaly napínáním a smršťováním samotné čočky. Tyčinky jsou citlivé pouze na jas a nerozeznají jednotlivé vlnové délky (barvu). O to se starají čípkové – konkrétně pak máme tři druhy čípků, a ty jsou citlivé na červené, zelené a modré barvy. [1]



Obr. 2: Ilustrační obrázek reálného obrazu na sítnici, převzato z [2]

Na rozdíl od fotoaparátu je ale reálný přenesený obraz na sítnici velice nekvalitní. To můžeme vidět na obrázku 2, který simuluje reálný obraz dopadající na sítnici a bere ohled na nerovnoměrné rozložení tyčinek/čípků, strukturu samotné sítnice, slepou skvrnu a pozici žluté skvrny. Oko samo o sobě má tedy daleko od optické dokonalosti. Velice zajímavé je pak uvědomit si, jak tento nedokonalý obraz dokáže mozek zpracovávat na základě kontinuálního vnímání scény okem a jejího skenování. Mozek pak dokáže vytvořit celistvý, téměř dokonalý obraz pozorovaného vjemu. [2]

Mimo jiné, během zpracování obrazu mozkiem, dochází i k vyvážení barev, a proto vnímáme barvu objektu stejně barevně pod jakýmkoli světlem, tedy nezávisle na barvě osvětlení [3]. Tomuto faktu se chci nadále věnovat v mé práci.

## 2 Kolorimetrie

Vnímání barvy je vlastně rozeznávání různých vlnových délek světla dopadajících do našeho oka. Respektive vlnových délek elektromagnetického vlnění ve viditelném spektru 380 nm – 750 nm. To však neznamena, že každá barva může být vnímána pouze na oné určité vlnové délce. Stejnou barvou můžeme vnímat světlo složené z různých vlnových délek a každou barvu tak můžeme reprezentovat minimálně třemi barevnými hodnotami. Tomu se říká barevný metamerismus. [4]

### 2.1 Barevné prostory

Existuje několik přístupů k tomu, jak matematicky popsat barevný svět. Každý tento přístup má své výhody a nevýhody a každý se ze své podstaty hodí pro jiné aplikace. Pro správnou volbu vhodného prostoru a pro efektivní práci s ním je proto dobré seznámit se s konceptem jednotlivých barevných prostorů. V této části práce se budu zabývat mnou použitými barevnými prostory.

#### 2.1.1 RGB barevný prostor

RGB barevný prostor vychází jak z funkce většiny senzorů záznamových zařízení, tak z funkce lidského oka. Obě zmíněné totiž obsahují červený (R), zelený (G) a modrý (B) kanál. Původní širokospektrální barevná informace je tedy komprimována do tří dimenzí (viz. obrázek 3) [5]. Výsledná barva je pak složením jednotlivých složek, a jedná se tedy tak o systém **adaptivní**, který je celý založen na barevném metamerismu. Bohužel tímto způsobem nemůžeme získat všechny reálné barvy. Aby tomu tak bylo, museli bychom některé barvy nejen přičítat, ale i odebírat; pro určité zelené a modré barvy tedy mohou kanály nabývat záporných hodnot. [1] Také nelze bez výpočtu určit celkový jas, neboť ten je rozložen do každého z kanálů.

#### 2.1.2 XYZ barevný prostor

Tento barevný prostor používá souřadnice  $X$ ,  $Y$  a  $Z$ , které na rozdíl od RGB barevného prostoru byly odvozeny čistě matematicky a mohou nabývat pouze **kladných hodnot**. Není tedy možné v této soustavě přímo zaznamenávat, ale

můžeme do ní barvy pouze přepočítat z RGB systému. Je ale výhodný v tom, že je schopný popsat veškeré reálné barvy a informace o jasů je uložena čistě v kanálu  $Y$ , přičemž kanály  $X$  a  $Z$  pak slouží pro definici barvy. [1]

Výpočet z RGB souřadnic do XYZ je dle [1] následovný:

$$X = 2,7690R + 1,7518G + 1,1300B \quad (1)$$

$$Y = 1,0000R + 4,5907G + 0,0601B \quad (2)$$

$$Z = 0,0000R + 0,0565G + 5,5943B \quad (3)$$

### 2.1.3 HSV barevný prostor

HSV (*hue, saturation, value*) je cylindrický systém s osami pro tón, sytost a jas. Na rozdíl od RGB nebo XYZ je velice vhodný pro výběr **konkrétní barvy**, případně pro barevné filtrace nebo posuny. Barva je totiž určena jednou osou (konkrétně osou pro tón ( $H$ )). Na další ose se nachází informace o sytosti ( $S$ ), případně o světlosti ( $V$ ) dané barvy. Matematicky jde však pouze o přepočet RGB souřadnic (dle vzorců 4-6), a tedy tento prostor neodpovídá nelineárnímu vnímání lidského oka. [6]

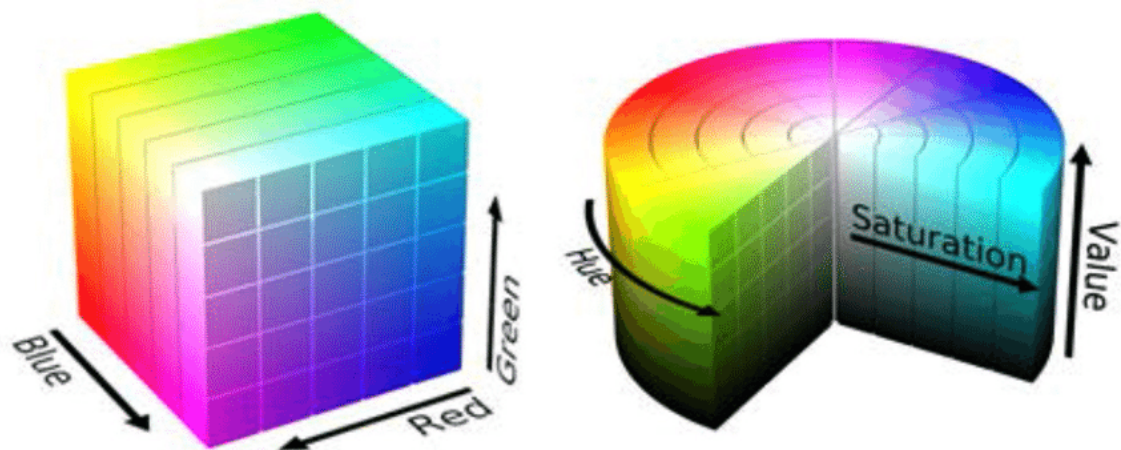
$$H = 60^\circ \times \begin{cases} 0 + \frac{G-B}{max-min}, & \text{pro max} = R \\ 2 + \frac{B-R}{max-min}, & \text{pro max} = G \\ 4 + \frac{R-G}{max-min}, & \text{pro max} = B \end{cases} \quad (4)$$

$$S = \frac{max-min}{max} \quad (5)$$

$$V = max \quad (6)$$

Kde  $max/min$  je maximální/minimální hodnota z  $R, G, B$  kanálů.





Obr. 3: Schéma RGB a HSV barevných prostorů, převzato z [7]

Na obr. 3 vidíme rozdíl definice barvy dle os tří zmíněných barevných prostorů.

#### 2.1.4 CIELAB barevný prostor

Pokud se chceme zabývat vnímáním lidského oka, nejvhodnější barevný prostor bude CIELAB. Ten totiž vychází z vlastností lidského oka a **velikost numerické změny** kdekoli v prostoru odpovídá **velikosti změny vnímání** při pozorování člověkem. Tento prostor byl založen na tisíci subjektivních testů a zpracován mezinárodní komisí pro osvětlování CIE. [8]

Jeho osy jsou  $L$  (Lightness/světlost), kde hodnota 0 znamená černou a hodnota 100 maximální viditelnou intenzitu bez poškození lidského oka;  $a$  (zeleno-červená) a  $b$  (modro-žlutá), které obě tyto osy nabývají hodnot -128 až +127. Ve středu (v případě  $a, b = 0$ ) pak tedy existují pouze barvy v šedo-tónové škále. [9]

Hodnoty  $L, a$  a  $b$  vektorů pak můžeme získat z prostoru XYZ těmito výpočty:

$$L = 116(Y/Y_0)^{1/3} - 16 \quad (7)$$

$$a = 500((X/X_0)^{1/3} - (Y/Y_0)^{1/3}) \quad (8)$$

$$b = 200((Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}) \quad (9)$$

## 2.2 Vyvážení barev

### 2.2.1 Barevná stálost

Barevná stálost je **kompenzace vlastností zdroje** (spektrální složení světla) osvětlujícího určitou scénu. Výsledný obraz je totiž kombinací odrazivosti povrchu ve scéně, charakteristiky vlastního senzoru a vlastností zdroje osvětlujícího scénu [6]. Při změně zdroje se tedy změní i barevné vlastnosti obrazu dopadajícího na senzor.

Lidský vizuální systém naopak ale vnímá barvu obrazu nezávisle na charakteristice osvětlení [10]. Je schopen charakteristiku zdroje kompenzovat.

Barevná stálost tedy znamená úprava obrazu pořízeného pod neznámým světlem tak, aby se jevil jako pořízený pod kanonickým (často bílým) světlem [10]. Toho lze dosáhnout tak, že nejdříve určíme barvu světelného zdroje a následně kompenzujeme původní obraz dle spektrální charakteristiky zdroje.

### 2.2.2 Vyvážení bílé

Při pozorování předmětů okem přizpůsobíme jejich vnímanou barvu na základě zkušenosti, a tedy vnímáme barevné předměty stejně barevně i v různých typech osvětlení. Člověku tedy nezáleží na barevném složení světla ozařujícího pozorovaný objekt, neboť se právě dokáže přizpůsobit barvě konkrétního osvětlení. Bílý papír je pro oko vždy bílý. Ať už za slunečného dne nebo v noci pod pouličním osvětlením. Tato vlastnost lidského oka je založena na minulé zkušenosti a děje se díky možnosti ovlivnit zisk jednotlivých RGB čípků. [4]

Senzor kamery a kamera samotná funguje podobně. Senzorem se snímá určitá intenzita osvětlení pro jednotlivé barevné kanály a tato hodnota pořízené fotografie je upravena tak, aby se člověku jevila fotografie přirozeně. Neděje se tak však přímo při snímání, ale až v **post-processingu** určitého záznamu. Na obrázku 4 můžeme vidět, jak vypadá špatně a dobře vyvážená bílá. Nevyvážené barvy působí na člověka velice nepřírozeně a rušivě.

Ve filmové fotografii se tento problém řešil barevnými filtry a volbou vhodné senzibilovaného materiálu.

V digitálním zpracování signálu využíváme různých algoritmů, které upraví raw data ze senzoru. K těmto úpravám jsou pak různé přístupy. V ideálním případě při snímání nastavíme barevnou teplotu zdroje světla manuálně. Jiný způsob umožňuje sejmutí bílého etalonu, který nastavíme jako referenční bílou. Na základě této definice bílého bodu pak algoritmus přizpůsobí výsledné zisky jednotlivých kanálů snímku tomuto charakteru osvětlení.



Obr. 4: Ukázka špatně vyvážené bílé na 3000 K při denním světle (nahore) a dobře vyvážené bílé na 5600 K (dole)

## 2.3 Přístupy k barevné stálosti

Správná korekce a úprava barev při snímání je problémem hned v několika odvětvích. Ať již při přesném **přenosu barevné informace** objektu při aplikaci strojového vidění v průběhu měnícího se denního světla [11], při **detekci tváří** nebo při **video trackingu** [12], při odstranění barevných artefaktů a zlepšení kontrastu v **lékařských aplikacích** [13], při zpracování **satelitních snímků** [14] nebo při **digitalizaci** kinematografických děl z filmových kopií [15].

Dle [16] lze přístupy ke korekci barev rozdělit na dva základní proudy. První z nich je korekce prováděná před samotným snímáním kalibrací senzoru. To vyžaduje znalost charakteristik senzoru pod různým typem osvětlení a poté mapování odraznosti povrchu ve známém kanonickém osvětlení na neznámé osvětlení snímané scény [16].

Druhý proud je založen na již nasnímaných datech. Ten pak můžeme dle [12] dále dělit na tři podkategorie. Konkrétně na přístupy založené na transformaci dat, na statistické přístupy a přístupy založené na strojovém učení.

Přístupy založené na transformaci dat vychází z vlastností scény nebo z vlastností oka a tyto poznatky se snaží vhodným algoritmem aplikovat přímo na již nasnímaná data. Mezi takové algoritmy patří například Gray world, White patch, Retinex přístupy, Von Kriesův algoritmu a další. [10]

Statistické, pravděpodobnostní nebo chcete-li Bayesianké přístupy mají společné to, že vycházejí ze statistických odezev RGB soustavy za různých světelných zdrojů při různých scénách. Dle těchto poznatků poté odhadují, jakým světelným zdrojem byla nasvícena určitá scéna a dle tohoto odhadu zdroje poté korelují výstupní barvy fotografie. K nejznámějším algoritmům této skupiny patří Color by corelation [17].

Barevné korekce pomocí strojového učení jsou založené na dvou fázích. Na fázi učení, kdy algoritmus hledá spojitosti mezi vstupními s výstupními daty a na základě těchto spojitostí poté dokáže v druhé fázi predikovat výstup předtím neviděných vstupů. To vše je samozřejmě velice závislé na trénovacích datech a výsledky korekcí jim budou přímo odpovídat. [18]

### 2.3.1 Gray world

Jedna z nejstarších, ale i nejjednodušších metod je metoda Gray world (šedý svět (dále GW)). Základním předpokladem této metody je, že průměrná odrazivost všech bodů ve scéně je za průměrného osvětlení ve všech kanálech stejná, a tedy že výsledkem je **šedá barva**. [19] Tento předpoklad je splněn ve většině reálných scén s rozmanitým rozložením barev, a proto i přes svoji jednoduchost dosahuje GW dobrých výsledků a je používán dodnes. Ze svého principu však ztroskotává v případech, kdy je ve scéně převaha jedné barvy nebo pouze vysoké/nízké jasy.

Realizuje se rozdělením fotografie na jednotlivé RGB kanály, v kterých se pak jednotlivě spočte jejich průměrná hodnota [4].

$$R_{avg} = \frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n R(x, y) \quad (10)$$

$$G_{avg} = \frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n G(x, y) \quad (11)$$

$$B_{avg} = \frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n B(x, y) \quad (12)$$

Kde  $m$  a  $n$  jsou rozměry obrázku v pixelech a  $R$ ,  $G$ ,  $B$  jednotlivé barevné kanály.

Z těchto průměrných hodnot se poté dopočítají koeficienty  $\alpha$  a  $\beta$ , které slouží k úpravě jednotlivých kanálů tak, aby jejich průměrná hodnota byla rovna. Jako referenční kanál, dle kterého se upraví dva zbývající, se bere kanál zelené barvy.

$$\alpha = \frac{G_{avg}}{R_{avg}} \quad (13)$$

$$\beta = \frac{G_{avg}}{R_{avg}} \quad (14)$$

Výsledné hodnoty jednotlivých kanálů budou tedy vypadat takto:

$$R_{out} = \alpha R \quad (15)$$

$$G_{out} = G \quad (16)$$

$$B_{out} = \beta B \quad (17)$$

### 2.3.2 White patch Retinex

Retinex teorie vycházejí z vnímání barev člověkem. Slovo „Retinex“ je složeninou ze slov „retina“ a „cortex“ a tedy se snaží zohlednit jak vjem přímo okem, tak zpracování zachyceného obrazu mozkiem. Teorie White patch patří k nejjednodušším a vychází z vlastnosti lidského vnímání určit jako bílý předmět ve scéně ten, který má nejvyšší jas. Na ten bod pak vyváží bílou stejným principem koeficientů jako metoda GW. Vstupní data však nejsou průměrný jas scény, ale maximální hodnota jednotlivých kanálů.

$$\alpha = \frac{G_{max}}{R_{max}} \quad (18)$$

$$\beta = \frac{G_{max}}{B_{max}} \quad (19)$$

Kde  $R_{max}$ ,  $G_{max}$  a  $B_{max}$  jsou maximální hodnoty v jednotlivých kanálech.

Následná úprava pro výstupní kanály pak probíhá dle rovnic 15-17 stejně jako u metody GW.

Tato teorie nefunguje pro fotografie, kde se vyskytují body za dynamickým rozsahem snímače. Tedy místa v saturaci – algoritmus je pak bere jako bílé i přes to, že o bílou vůbec nemusí jít; koeficienty  $\alpha$  a  $\beta$  budou mít v tomto případě hodnotu jedna – tedy fotografie zůstane beze změny. Tato korekce ze své podstaty také nebude mít dobré výsledky v případě, že nejsvětlejší bod ve scéně bude mít jinou než bílou barvu.

### 2.3.3 Multi Scale Retinex

Další teorie Retinex, tedy vycházející z fyziologického vnímání obrazové informace člověkem, je poněkud komplexnější než White patch. Multi scale Retinex je nelineární spektrální transformace obrazu převážně poskytující barevnou stálost, kompresi dynamického rozsahu a zlepšení ostrosti scény [20]. Tato transformace vychází z modelu lidského vidění a tím pádem bere v potaz to, že výsledný obraz je součinem zdroje světla a odrazivosti scény, přičemž zdroj světla se mění postupně – a tedy je lokálně téměř konstantní [21]. Její výstup je **rozdíl vstupu a jeho okolí**, dle [20] tedy:

$$R_i(x, y) = \log(I_i(x, y)) - \log(I_i(x, y) * F(x, y)) \quad (20)$$

Kde  $I_i$  je vstupní obrázek s jeho kanálem  $i$ ,  $R_i$  je výstup algoritmu Single scale Retinex na kanálu  $i$  a  $F$  je normalizovaná funkce okolí:

$$F(x, y) = C \exp[-(x^2 + y^2)/2\sigma^2] \quad (21)$$

Kde  $\sigma$  je standardní výchylka filtru a určuje kolik prostorových detailů je ponecháno a  $C$  je normalizační faktor, který udržuje plochu gaussovské křivky pod hodnotou 1 [22].

Tato původní transformace Single scale Retinex (předchůdce multi scale Retinex) má ovšem problémy s výběrem vhodného parametru filtru  $\sigma$ , neboť jde vždy o kompromis mezi dynamickým rozsahem a barevným podáním. Proto byl algoritmus vylepšen a výsledný obraz je u Multi scale Retinex kombinací více parametrů  $\sigma$ , tedy dle [21]:

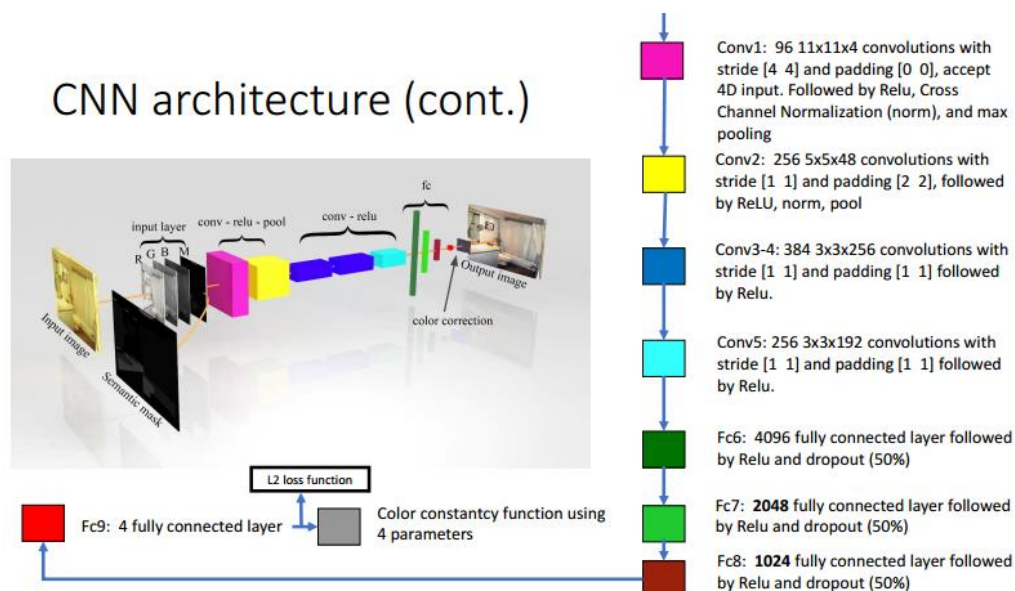
$$R_{MSRi} = \sum_{n=1}^N w_n R_{ni} = \sum_{n=1}^N w_n [\log(I_i(x, y)) - \log(I_i(x, y) * F_n(x, y))] \quad (22)$$

Kde  $N$  je počet parametrů filtru a  $w_n$  je váha každého jednotlivého parametru.

### 2.3.3 Machine learning: Semantic White Balance

Pro vyvážení bílé a barevnou stálost se v poslední době začalo využívat i neuronových sítí a strojového učení. To se ukázalo být efektivnější než tradiční algoritmy a již existuje několik přístupů k vyvažování barev pomocí strojového učení. [6]

Já jsem si pro porovnání ve svojí práci vybral nový článek od Mahmouda Afifi: *Semantic White Balance: Semantic Color Constancy Using Convolutional Neural Network* [23] (dále jen SCCCNN), kde se věnuje aplikaci neuronových sítí na vyvážení bílé ve fotografiích.



Obr. 5: Princip SCCCNN, převzato z [24]



Rychlý vývoj strojového učení měl velký dopad i na oddělení sémantické informace v obraze. Toho se snaží využít tento přístup k vyvážení barev a společně s barevnými kanály RGB využívá Afifi ve svém algoritmu i sémantické masky. Konvoluční neuronová síť CNN je trénovaná se sémantickou informací obrazu, a to vede k lepšímu barevnému vyvážení. Dle Afifiho se chyba výstupu algoritmu sníží až o 40 % [23].

Algoritmus využívá neuronové sítě AlexNet, kterou ale upraví pro čtyřdimenzionální vstup (čtvrtým kanálem je sémantická maska). Tyto úpravy a princip celé sítě je uveden na obrázku 5. Trénování CNN Afifi prováděl na souboru dat ADE20K, který obsahuje 20210 trénovacích obrazů s přesnou sémantickou maskou a 2 000 validačních obrazů. Tyto obrazy byly považovány za sadu s dobrým vyvážením bílé, proto pro každý obraz z této sady pak Afifi vytvořil i 769 syntetických obrazů se špatným barevným vyvážením, špatnou gamma korekcí a případně i posunuté o dva pixely v prostoru každou stranou. Tím obohatil učící proces. [23]

Celý algoritmus včetně natrénované neuronové sítě je dostupný online na [24] a lze spustit v programu MATLAB.

## 3 Hodnocení kvality vyvážení bílé

Kvalitu vyvážení bílé můžeme hodnotit **subjektivně** nebo **objektivně**. Subjektivní přístup vyžaduje provedení testů s dostatečně velkým vzorkem lidí a tyto testy jsou časově i finančně náročné. V této práci se ale chci věnovat objektivnímu vyhodnocení kvality vyvážení bílé. Objektivní hodnocení nevyžaduje provádění testů, je proveditelné pouze výpočtem a zaměřuje se pouze na hodnocení technických parametrů. Nevýhodou je, že výsledek nemusí vždy přesně odpovídat reálnému vjemu člověka.

### 3.1 JND a $\Delta E$

JND je jednotka, která udává hodnotu minimálního, právě postřehnutelného rozdílu (*Just noticeable difference*) standardním pozorovatelem. Standardní pozorovatel pak představuje vjem průměrného pozorovatele a byl určen mezinárodní komisí pro osvětlování v roce 1931. Toho můžeme využít právě v objektivním popisu subjektivních vjemů [9].

Metrika  $\Delta E$  je pak **euklidovská vzdálenost** dvou barev v určitém barevném prostoru. Pro barevný prostor RGB můžeme tedy vzdálenost dvou barev  $\Delta E$  vypočítat takto:

$$\Delta E_{RGB} = \sqrt{\Delta R^2 + \Delta G^2 + \Delta B^2} \quad (23)$$

Kde  $\Delta R$ ,  $\Delta G$  a  $\Delta B$  vyjadřují rozdíl v hodnotách jednotlivých kanálů pro jednotlivé barvy:

$$\Delta R = R_2 - R_1 \quad (24)$$

$$\Delta G = G_2 - G_1 \quad (25)$$

$$\Delta B = B_2 - B_1 \quad (26)$$

Vzhledem k tomu, že lidské oko je na některé barvy citlivější než na jiné, propojení JND a  $\Delta E$  je prakticky dobře proveditelné pouze v prostoru CIELAB, který vychází přímo z těchto vlastností oka.

Pro barevný prostor CIELAB bude mít pak euklidovská vzdálenost dvou barev následující tvar:

$$\Delta E^*_{ab} = \sqrt{\Delta L^2 + \Delta a^2 + \Delta b^2} \quad (27)$$

Tato vzdálenost pak dle D. Nickersona bude rovna přibližně 2,3 JND [9].

$$\Delta E^*_{ab} \approx 2,3JND \quad (28)$$

Standardní pozorovatel tedy dle statistických výsledků mnoha experimentů vnímá konkrétní rozdíly v euklidovských vzdálenostech takto [9]:

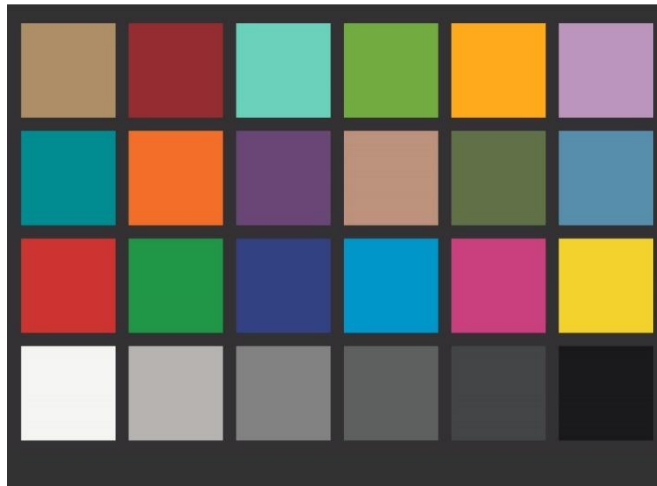
- $0 < \Delta E < 1$  – Pozorovatel nepozoruje rozdíl v barvách
- $1 < \Delta E < 2$  – Pouze zkušený/trénovaný pozorovatel si všimne rozdílu
- $2 < \Delta E < 3,5$  – I nezkušený/netrénovaný pozorovatel si všímá rozdílu
- $3,5 < \Delta E < 5$  – Čitelný rozdíl v barvě
- $5 < \Delta E$  – Pozorovatel si všímá dvou rozdílných barev

## 3.2 Objektivní hodnocení

Ve své práci se zabývám objektivním vyhodnocením korekcí fotografií pořízených v laboratorním prostředí. Zaměřuji se na dva aspekty, a to sice na vyhodnocení přesnosti reprodukce barev definovaného color checkeru a na správný odstín pleťové barvy obličeje. Lidské oko je na správný odstín pleti velice citlivé, neboť se s lidmi setkáváme denně. Nesprávně vyvážená barva u obličeje může tedy působit velice rušivě. Vyhodnocení color checkeru jsem vzal taktéž v potaz, neboť mohu porovnat výsledek s přesně definovanými barvami z tabulky.

### 3.2.1 Vyhodnocení barev color checkeru

Color checker je vlastně tabulka, na které jsou různé barvy a odstíny (viz. obr. 6). Tyto barvy mají přesně definované souřadnice, které získáme z jeho data sheetu (viz. tabulka 1). Tyto souřadnice jednotlivých barev je pak možné porovnat s reálnými barvami z korigovaných fotografií. Na to je ideální použití euklidovské vzdálenosti mezi barvami v prostoru CIELAB a následné převedení na JND. Tím získáme objektivní metriku rozdílu vnímání barev pozorovatelem.



Obr. 6: Color checker

		<b>X</b>	<b>x</b>	<b>y</b>	<b>L</b>	<b>a</b>	<b>b</b>
<b>A 1</b>	<b>Light Brown</b>	29,1	0,390	0,386	60,9	6,8	25,0
<b>A 2</b>	<b>Ruby</b>	8,4	0,530	0,322	34,8	44,1	21,9
<b>A 3</b>	<b>Mint Green</b>	52,3	0,026	0,362	77,5	-35,5	2,1
<b>A 4</b>	<b>Warm Yellow</b>	33,0	0,342	0,504	64,2	-36,5	47,8
<b>A 5</b>	<b>Lila</b>	50,7	0,493	0,439	76,5	22,9	76,4
<b>A 6</b>	<b>Turquoise Blue</b>	35,8	0,314	0,279	66,4	21,0	-14,7
<b>B 1</b>	<b>Orange</b>	19,4	0,193	0,317	51,2	-40,0	-14,4
<b>B 2</b>	<b>Purple</b>	30,3	0,544	0,385	61,9	47,5	59,8
<b>B 3</b>	<b>Light Skin</b>	8,7	0,295	0,227	35,4	24,3	-21,7
<b>B 4</b>	<b>Foliage</b>	32,7	0,381	0,357	64,0	13,5	17,0
<b>B 5</b>	<b>Sky</b>	14,7	0,341	0,421	45,3	-13,8	20,7
<b>B 6</b>	<b>Red</b>	24,2	0,239	0,277	56,3	-9,7	-21,4
<b>C 1</b>	<b>Green</b>	15,7	0,573	0,332	46,5	59,2	38,8
<b>C 2</b>	<b>Blue</b>	22,4	0,282	0,505	54,4	-49,3	33,0
<b>C 3</b>	<b>Cyan</b>	6,1	0,206	0,173	29,6	15,5	-38,6
<b>C 4</b>	<b>Magenta</b>	25,4	0,192	0,243	57,5	-18,8	-36,6
<b>C 5</b>	<b>Yellow</b>	17,7	0,432	0,256	49,1	59,2	-4,3
<b>C 6</b>	<b>White</b>	65,4	0,440	0,474	84,7	-3,2	78,0
<b>D 1</b>	<b>White</b>	91,0	0,313	0,331	96,4	-0,8	0,9
<b>D 2</b>	<b>Gray 1</b>	45,5	0,318	0,332	73,2	0,8	1,6
<b>D 3</b>	<b>Gray 2</b>	22,3	0,313	0,329	54,3	0,0	-0,1
<b>D 4</b>	<b>Gray 3</b>	11,6	0,310	0,327	40,5	-0,4	-0,6
<b>D 5</b>	<b>Gray 4</b>	5,9	0,307	0,327	29,1	-0,7	-0,8
<b>D 6</b>	<b>Black</b>	1,0	0,308	0,314	9,0	1,0	-1,4

Tabulka 1: Definice barev color checkeru

### 3.2.2 Vyhodnocení pleťové barvy

Pleťová barva by se měla dle [25] ve vektorových souřadnicích HSV prostoru pohybovat v rozmezí  $0^\circ$  až  $50^\circ$  pro kanál H a v rozmezí 0,23 až 0,68 pro kanál S. Tento rozsah je možné převést do barevného prostoru CIELAB a zjistit, jestli se opravdu pleťová barva nachází v tomto rozsahu. Pokud tomu tak není, je možné znovu vypočítat euklidovskou vzdálenost, tentokrát od hranice tohoto rozmezí, převést ji na JND a zjistit, jak moc se pleťová barva objektivně liší od standardních hodnot.

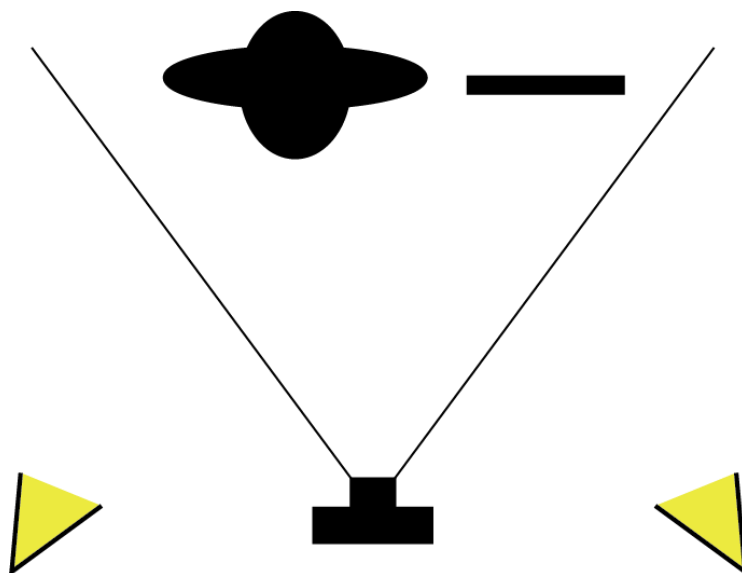


Obr. 7: Rozsah pleťové barvy v HSV prostoru dle [25]

## 4 Snímání vstupních dat

Pro objektivní porovnání jednotlivých algoritmů jsem se rozhodl využít mnou nafocené vstupní snímky. Můj test se zaměřuje na věrnou reprodukci barev a podání pleťové barvy, a proto bylo nutné mít v obraze jak obličej člověka, tak barevný etalon (color checker). Fotografie jsem pořizoval ve školním studiu se školními světly. Konkrétně pak s LED plošnými světly Fomei LED Light 1200-54, televizními zářivkami Fomei DESK 330M a fotografickými halogenovými světly. Focení probíhalo před zelenou klíčovací stěnou, pozice color checkeru byla vždy stejná (pro umožnění zpracování v dávkách) a pro každé světlo (resp. různá nastavení světel) jsem použil i tři barevné filtry posouvající barevnou teplotu zdrojů. Konkrétně pak modrý CTB filtr s hustotou  $\frac{1}{4}$ , oranžový CTO filtr s hustotou  $\frac{1}{4}$  a efektní žlutohnědý filtr.

Fotografie jsem snímал školním fotoaparátem Nikon s manuálním nastavením expozice. Tu jsem měřil za pomoci šedé tabulky digitálním jasoměrem Asahi Pentax.



Obr. 8: Schéma snímání testovacích snímků



Obr. 9: Náhledy všech pořízených testovacích snímků

<b>Fotografie</b>	<b>Osvětlení scény</b>	<b>Nastavení</b>	<b>Filtr</b>
<b>1</b>	Osvětlení místnosti	-	-
<b>2</b>	LED Light 1200-54	-	-
<b>3</b>	LED Light 1200-55	-	Oranžový
<b>4</b>	LED Light 1200-56	-	Žlutohnědý
<b>5</b>	LED Light 1200-57	-	Modrý
<b>6</b>	Zářivky DESK 330M	-	-
<b>7</b>	Zářivky DESK 330M	-	Oranžový
<b>8</b>	Zářivky DESK 330M	-	Žlutohnědý
<b>9</b>	Zářivky DESK 330M	-	Modrý
<b>10</b>	Halogenová světla	Model 6	-
<b>11</b>	Halogenová světla	Model 3	-
<b>12</b>	Halogenová světla	Model Full	-
<b>13</b>	Halogenová světla	Model Full	Modrý
<b>14</b>	Halogenová světla	Model 6	Modrý
<b>15</b>	Halogenová světla	Model 3	Modrý
<b>16</b>	Halogenová světla	Model Full	Oranžový
<b>17</b>	Halogenová světla	Model 6	Oranžový
<b>18</b>	Halogenová světla	Model 3	Oranžový
<b>19</b>	Halogenová světla	Model Full	Žlutohnědý
<b>20</b>	Halogenová světla	Model 6	Žlutohnědý
<b>21</b>	Halogenová světla	Model 3	Žlutohnědý

Tabulka 2: Tabulka způsobu osvětlení pořízených testovacích snímků



## 5 Korekce barev nafocených dat

V programu MATLAB jsem implementoval dle výše popsané teorie základní algoritmy pro úpravu vyvážení bílé. Konkrétně pak Gray world, White patch Retinex a Multi scale Retinex. Veškeré tyto komentované skripty uvádím v přílohách A, B, C.

Pro zástupce strojového učení, jsem zprovoznil za pomoci Deep learning toolboxu Afího algoritmus dostupný na [24].

Samotné algoritmy jsem pak doplnil o cyklus *for*, aby bylo možné hromadně zpracovávat a ukládat výstupní, korigovanou fotografii pro všechny testovací snímky. S takto připravenými daty jsem mohl přistoupit k vyhodnocení.

### 5.1 Vyhodnocení barev color checkeru

#### 5.1.1 Skript v programu MATLAB

Pro porovnání reálných barev color checkeru a korigovaných barev z nafocených fotografií bylo třeba napsat skript v programu MATLAB, jehož komentovaný kód uvádím v příloze D.

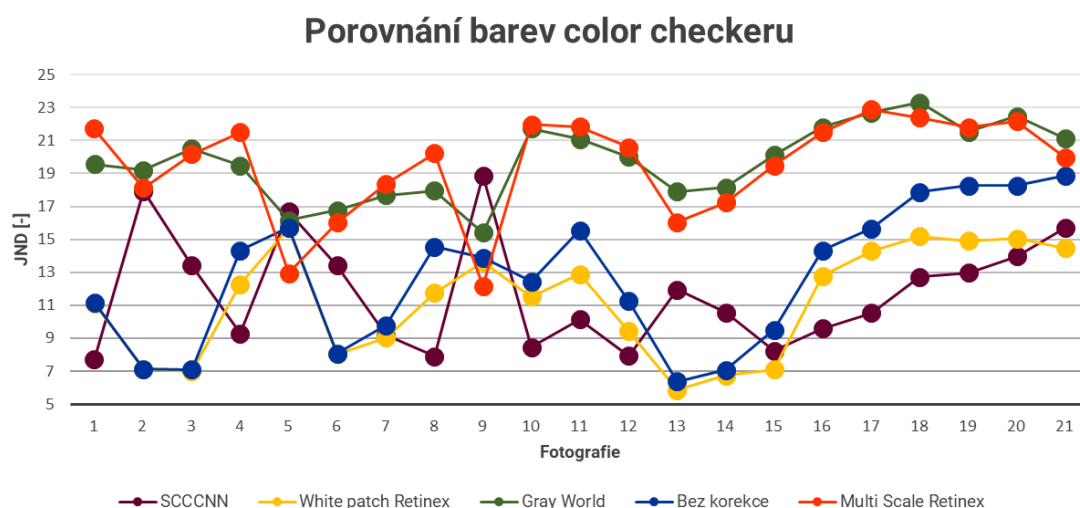
Pro účinné porovnání s referenční barvou dle data sheetu jsem vytvořil matici *barva\_ref* o velikosti 3x24. 3 řádky pro souřadnice CIELAB prostoru a 24 sloupců, pro každou barvu jeden.

Neboť se color checker nachází v každé testovací fotografii na stejném místě, pro výběr reálných barev color checkeru z vyfocených fotografií jsem definoval tuto pozici, která je použitelná pro všechny vyfocené fotografie. Vybral jsem čtverec o velikosti 250x250 pixelů a tento čtverec uložil do čtyřdimenzionální matice o velikosti 250x250x3x24 *barva\_foto*. Neboť pro samotné porovnání barev je třeba vybrat jednu reprezentující barvu, barevný čtverec jsem zprůměroval do jedné barvy do matice 3x24 *barva\_prumer*. Tyto hodnoty jsou však v barevném prostoru RGB a bylo tedy nutné je převést do CIELAB. To jsem realizoval vlastní funkcí *JardaRgb2lab* dle rovnic 1-3 a 7-9.

S takto připravenými maticemi jednotlivých barev jsem dle rovnice 27 spočetl Euklidovskou vzdálenost mezi jednotlivými barvami color checkeru a dle rovnice 28 potom hodnotu JND pro každou barvu. Tuto hodnotu JND jsem pak zprůměroval pro všechny barvy color checkeru a uložil do vektoru pro každou vyfocenou fotografii.

Skript tedy pro veškeré korigované fotografie vypočte hodnotu JND odchylky od definované barvy color checkeru. Tyto hodnoty jsem poté převedl do Excel tabulky pro jednodušší statistickou práci s daty a vyhodnocení.

### 5.1.2 Výsledky



Graf 1: JND barev color checkeru každé fotografie

V grafu 1 vidíme výsledné hodnoty JND pro každou korigovanou fotografii i pro originální nekorigovaný snímek (konkrétní tabulkové hodnoty pak uvádím v příloze G).

Metoda	Průměr	Medián	Směrodatná odchylka
<b>SCCCNN web</b>	11,8	10,6	3,3
<b>White patch Retinex</b>	11,2	11,7	3,2
<b>Gray World</b>	19,7	20,0	2,2
<b>Multi Scale Retinex</b>	19,5	20,2	3,0
<b>Bez korekce</b>	12,7	13,9	4,0

Tabulka 3: JND, medián a směrodatná odchylka jednotlivých korekčních metod

Tabulka 3 uvádí průměrné hodnoty JND, jejich medián a směrodatnou odchylku. Tabulka 4 pak stejné veličiny z rozdílů JND od originálního nekorigovaného snímku.

Metoda	Průměr	Medián	Směrodatná odchylka
<b>SCCCNN web</b>	4,5	5,0	2,1
<b>White patch Retinex</b>	1,5	1,4	1,3
<b>Gray World</b>	7,0	7,5	3,6
<b>Multi Scale Retinex</b>	7,2	7,2	3,2

Tabulka 4: průměr, medián a směrodatná odchylka absolutních hodnot rozdílů JND od originálního nekorigovaného snímku

### 5.1.3 Statistické vyhodnocení

Z grafu 1 je na první pohled patrné, že metody Multi scale Retinex a Gray world selhávají v porovnání s ostatními testovanými metodami korekce. V drtivé většině (až na dvě výjimky, kterým se budu věnovat později) dosahují nejhorších výsledků a obraz naopak oddalují od reality. Algoritmy White patch a SCCCNN pak přináší dobré výsledky, ale zajímavostí je, že tyto jejich výsledky spolu, zdá se, nekorelují.

Nedobry výsledek korekce pomocí metody Gray world je dán scénou testovaných fotografií. Tento algoritmus totiž vyhodnocuje součet všech hodnot v jednotlivých kanálech a koriguje fotografie tak, aby průměrná odrazivost objektů ve scéně byla jako od šedých objektů. Jelikož v naší scéně však převažuje jedna barva (konkrétně zelená na pozadí) a scéna se obsahově nemění, Gray world vyhodnotí korekci vždy špatně. Proto má také nejmenší směrodatnou odchylku ze všech testovaných algoritmů a jeho výsledky se konstantně drží okolo 20 JND. Naopak směrodatná odchylka absolutních hodnot rozdílů JND oproti nekorigované fotografii má Gray world nejvyšší. Původní fotografii tedy koriguje nezávisle na změně osvětlení zdroje. Každou fotografii tak korigoval stejným (a pro tento typ scény nevhodným) způsobem.

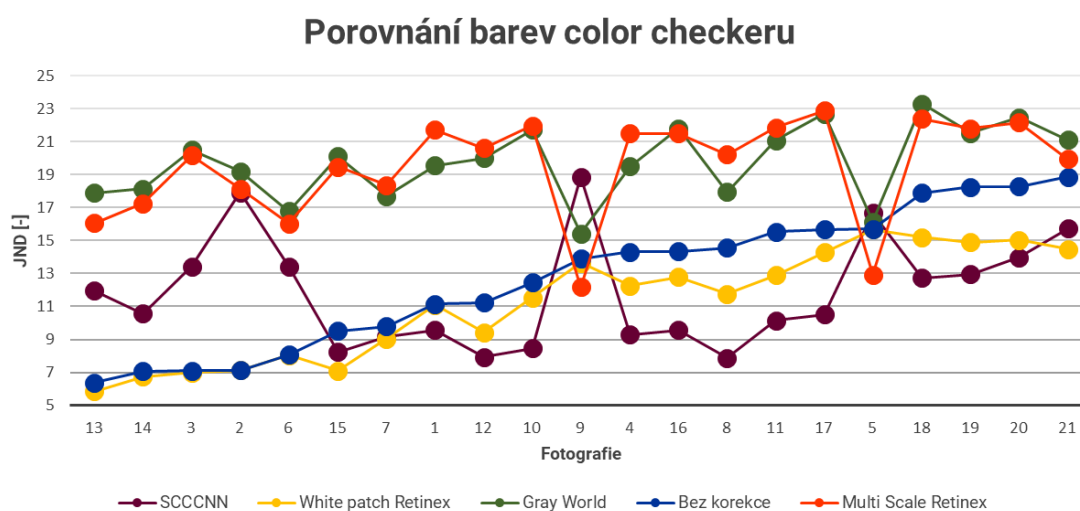
Výsledky korekce pomocí Multi scale Retinex jsou dle průměru druhé nejhorší. U tohoto algoritmu je to dané tím, že jeho použití není primárně určeno pro tento typ barevné korekce fotografie. Algoritmus se používá převážně pro zlepšení nějakým způsobem poškozených nebo nekvalitních záznamů (velké využití v letectví, NASA, zdravotnictví atd.) [13], [26]. Multi scale Retinex sice přináší více informace, či v některých případech lepší čitelnost obsahu fotografie, ale pro jemné barevné úpravy nenabývá nejlepších výsledků.

White patch Retinex dle hodnot průměru a směrodatné odchylky tabulky 3 naopak velice odpovídá svým výstupem posunu korigované fotografie. I z grafu 1 je patrné, že tato korekce silně koreluje se vstupem. Zároveň je ale korekce vždy „úspěšná“ a její výsledek se více přibližuje reálné barvě – nikdy naopak. To způsobuje vynikající výsledky této korekce a průměr dle tabulky 3 je nejlepší. To vše je způsobeno naší testovací scénou. Jelikož ve scéně se nachází jako nejsvětější bod vždy bílé políčko z color checkeru, White patch tím získává ideální předpoklady pro správné fungování. U některých fotografií je hodnota korekce pomocí této metody totožná s nekorigovanou fotografií. To je způsobeno tím, že bílé políčko color checkeru je již v oblasti saturace senzoru fotoaparátu.

Afifihho algoritmus SCCNN dosahuje taktéž vynikajících výsledků. Jeho průměr je po metodě White patch druhý nejlepší (odstup 0,6 JND však ani není okem pozorovatelný) a jeho medián je nejlepší ze všech. Dle tabulky 3 je ale patrné, že jeho korekce jsou mnohem razantnější než korekce metody White patch. Z grafu 1 je pak patrné, že ale ne vždy má jeho korekce pozitivní vliv na zlepšení věrnosti barev. V některých případech dojde naopak k ještě většímu barevnému zkreslení, než jaké obsahuje původní nekorigovaný snímek. Lepšímu popisu těchto výsledků a korelaci mezi kvalitou výstupu a zdrojem se věnuje další kapitola.

### 5.1.4 Vyhodnocení v závislosti na vstupu

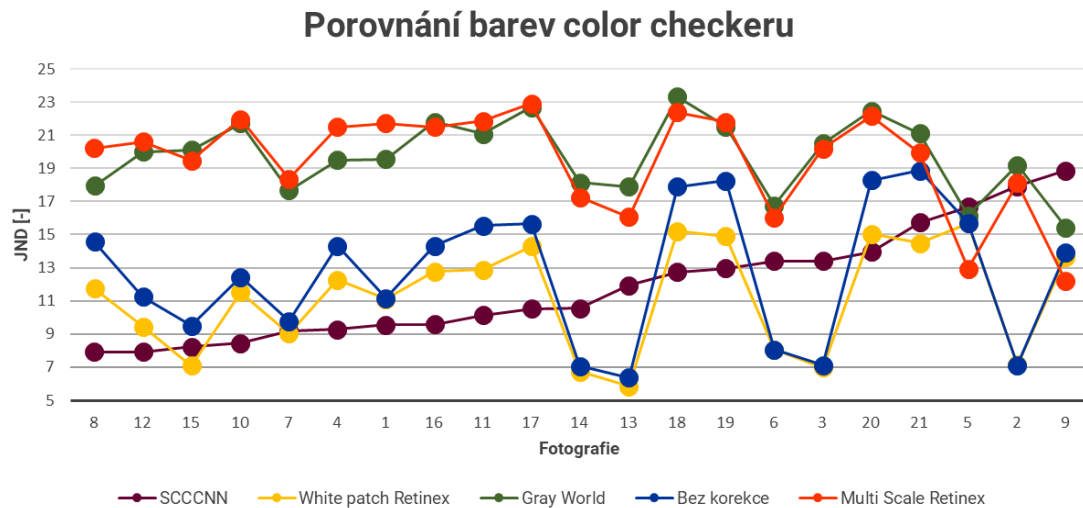
Pro lepší pozorování v závislosti na vstupu jsem graf 1 seřadil dle hodnot JND právě pro vstupní data:



Graf 2: JND barev color checkeru každé fotografie (řazeno dle vstupních dat)

Z tohoto grafu je patrné, že White patch silně koreluje se vstupem, neboť má stejnou stoupavou tendenci. Se vstupem (leč v jiné absolutní hodnotě) povětšinou korelují i metody MS Retinex a Gray world. Výjimka však nastane u jejich korekcí fotografií číslo 9 a 5. Zde dochází k nepředvídatelnému zlepšení výsledků těchto dvou metod, a naopak ke zhoršení výsledků SCCCNN algoritmu. Podíváme-li se na tyto fotografie, zjistíme, že obě tyto zmíněné fotografie mají velice silný modrý tón.

Výsledky korekce pomocí SCCCNN algoritmu jsou pak velice konstantní pro zkreslení vstupu od asi 10 JND do 15 JND. S výjimkou právě fotografie číslo 9, kdy dojde ke zhoršení výstupu téměř o 150 %. Seřadíme-li graf 1 dle hodnot výstupu algoritmu SCCCNN:



Graf 3: JND barev color checkeru každé fotografie (řazeno dle SCCCNN korekce)

Je lépe vidět, že SCCCNN korekce nabývá nejlepších výsledků pro zdroj odchýlený od reálných barev asi 10 až 15 JND. Pro vstup většího vstupního zkreslení, ale i pro hodnoty velice blízké reálným barvám, je výstup tohoto algoritmu ne tak dobrý. Nejhorší výsledek pak mají fotografie 2, 5, 9, které naopak překvapily dobrým výsledkem při aplikaci metod White patch a Gray world. Všechny tyto tři fotografie mají silný modrý tón, jak je možné vidět na obrázku Obr. 1010. To si můžeme ověřit v tabulce 1. Šlo vždy o fotografie, kde byl světelný zdroj studené barvy podpořený modrým filtrem.

Z těchto poznatků lze tedy usuzovat, že SCCCNN algoritmus (naopak od Gray world nebo MS Retinex) není účinný pro modře zbarvené fotografie na vstupu. I tak ale jeho výsledky pro tyto fotografie dosahují lepších nebo srovnatelných výsledků právě se zmíněnými algoritmy Gray world a White patch.



Obr. 10: Ukázka korekce barev problematických fotografií 2, 5, 9

## 5.2 Vyhodnocení pleťové barvy

### 5.2.1 Skript v programu MATLAB

Stejně jako pro vyhodnocení barev color checkeru, i pro analýzu pleťové barvy jsem napsal skript v programu MATLAB, jehož celý komentovaný kód uvádím v příloze E.

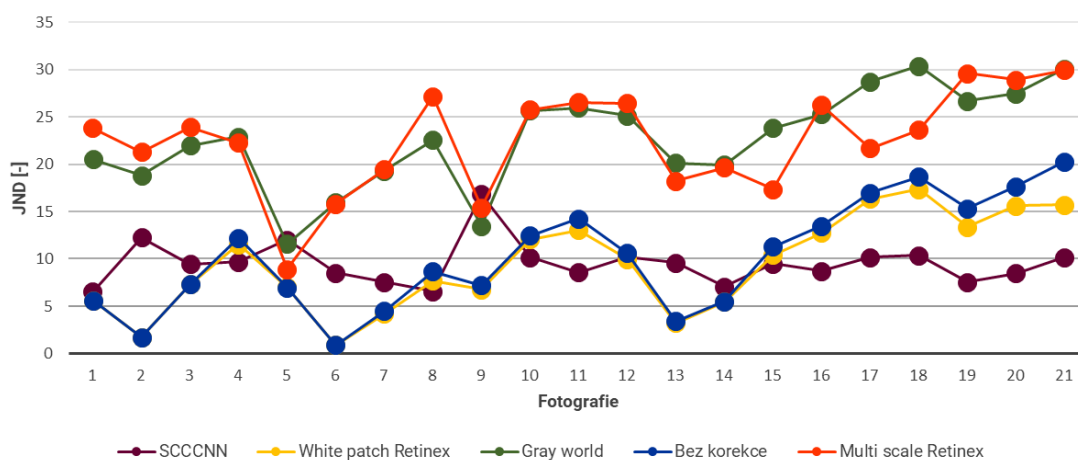
Pro vyhodnocení odchylky pleťové barvy od definovaného rozsahu bylo nejprve zapotřebí extrahovat zástupce pleťového odstínu z každé fotografie. To jsem pomocí funkce *ginput* udělal přímo v programu MATLAB, když jsem vytvořil matici *SouradnicePletovky.mat*, která pro každou fotografii obsahuje souřadnice s výskytem největší plochy pleti. Takto extrahované barvy jsem poté, stejně jako při vyhodnocení barev z color checkeru, zprůměroval a získal tak jednu referenční hodnotu.

Skript poté tuto referenční barvu převede do barevného prostoru HSV. Hranice rozsahu odstínů pleťové barvy jsou dle [25] definovány hodnotami kanálů *H* a *S*. Jako hodnotu kanálu *V* tedy vezme převedený kanál pro konkrétní barvu a celý tento rozsah je pak převeden do barevného prostoru CIELAB. Následně skript spočítá vzdálenost barvy na fotografii od definovaného rozsahu barev. Tato hodnota je poté též převedena na hodnotu JND a dána do porovnání se všemi korigovanými fotkami a všemi aplikovanými metodami, stejně jako u vyhodnocování barev z color checkeru.



## 5.2.2 Výsledky

### Vyhodnocení korekce pleťové barvy



Graf 4: JND pleťové barvy každé fotografie

Graf odchyly od definovaného prostoru pleťových barev v JND pro každou korigovanou fotografii i pro originální nekorigovaný snímek (tabulkové hodnoty uvádím v příloze G).

Metoda	Průměr	Medián	Směrodatná odchylnka
<b>SCCCNN web</b>	9,5	9,5	2,2
<b>White patch Retinex</b>	9,4	10,0	4,8
<b>gray World</b>	22,7	22,9	5,0
<b>Multi Scale Retinex</b>	22,5	23,6	5,2
<b>Bez korekce</b>	10,2	10,6	5,6

Tabulka 5: JND, medián a směrodatná odchylnka jednotlivých korekčních metod

Metoda	Průměr	Medián	Směrodatná odchylnka
<b>SCCCNN web</b>	5,2	5,1	3,2
<b>White patch Retinex</b>	0,8	0,6	1,0
<b>gray World</b>	12,4	12,6	3,0
<b>Multi Scale Retinex</b>	12,2	13,3	4,8

Tabulka 6: průměr, medián a směrodatná odchylnka absolutních hodnot rozdílů JND od definovaného prostoru pleťových barev

### 5.2.3 Vyhodnocení

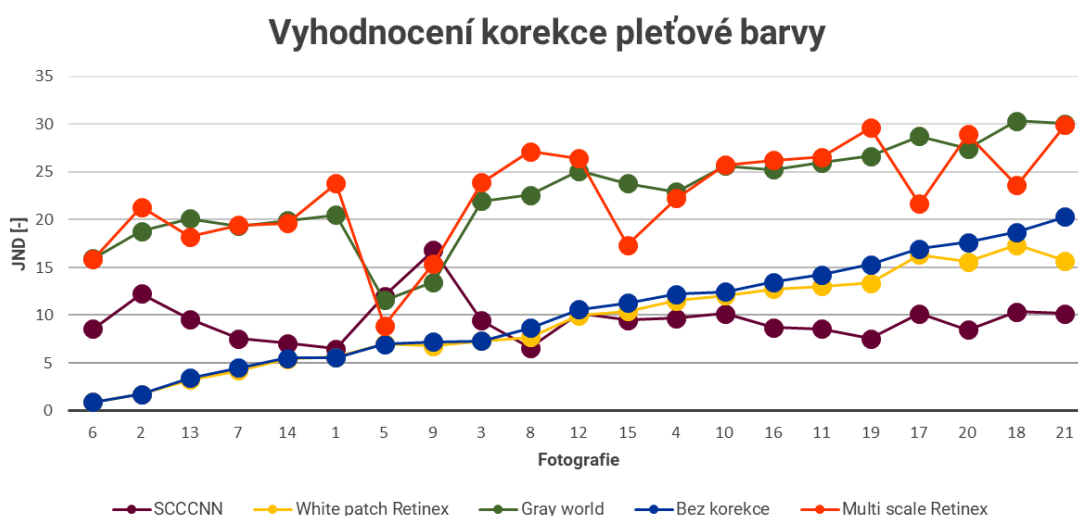
Průběhy v grafu 4 jsou velice podobné průběhům z vyhodnocení barev color checkeru (graf 1), avšak nejsou totožné. První a nejzajímavější odlišnost mezi těmito dvěma grafy je v průběhu hodnot algoritmu SCCNN. Oproti grafu 1 je zde průběh mnohem konstantnější a zdaleka ne tak neproměnlivý. To potvrzuje i menší směrodatná odchylka s hodnotou 2,2 oproti hodnotě 3,3 při vyhodnocení barev color checkeru.

Podíváme-li se na směrodatné odchylky ostatních algoritmů a porovnáme je s vyhodnocením barev color checkeru, s výjimkou SCCNN všechny vzrostly. To znamená, že tyto ostatní algoritmy mají při korekci pleťových barev více nestálé výsledky, kdežto SCCNN koriguje pleťové barvy s větší stálostí.

Tato korekce dosahuje lepších výsledků pro pleťovou barvu i v absolutních hodnotách JND. V tabulce 5 je vidět, že její průměr (o 2,3 JND nižší, než z tabulky porovnání barev 3) je rozdílný pouze o 0,1 JND s nejnižší hodnotou algoritmu White patch Retinex. Medián je pak o 1 JND nižší než v tabulce 4.

Mimo to se grafy 1 a 4 liší ještě v blízkosti průběhů originální, nekorigované fotografie a korekcí pomocí White patch Retinex. To potvrzuje i směrodatná odchylka rozdílů absolutních hodnot JND od našeho barevného etalonu, která u této korekce klesla z hodnoty 3,2 (tabulka 6) na hodnotu 1 (tabulka 4). To nasvědčuje tomu, že metoda White patch dobře koriguje barvy v celém spektru, ale přitom nemá tak velký vliv na odstíny pleti.

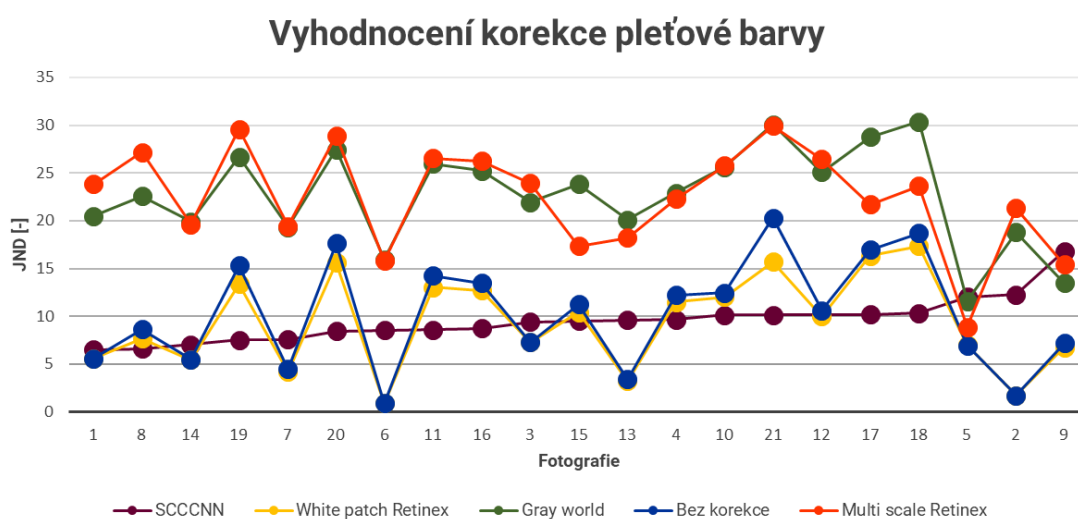
Tuto skutečnost potvrzuje i následující graf, kde jsou data seřazena dle hodnoty JND fotografií bez korekce:



Graf 5: JND pleťové barvy každé fotografie (řazeno dle vstupních dat)

Z grafu je zřetelný téměř totožný průběh nekorigované fotografie s výstupem algoritmu White patch a stejně tak i povětšinou stoupavý průběh metody Gray world a MS Retinex.

Konstantní výsledky korekce SCCCNN jsou zase patrné z grafu následujícího, kde jsou hodnoty seřazeny dle JND výstupu právě této metody a nejeví takové stoupavé tendence jako vidíme v grafu 5.



Graf 6: JND pleťové barvy každé fotografie (řazeno dle SCCCNN korekce)

Z tohoto grafu je i patrná korelace mezi všemi zbylými metodami, neboť (i přes jinou absolutní hodnotu) jsou vidět jejich podobné průběhy. Jak je z grafů patrné, rozmanitost vstupních dat neovlivňuje výstup SCCNN korekce tak výrazně jako u vyvážení barev color checkeru, ale problematické fotografie 2, 5, 9, které u minulého vyhodnocení měly nejhorší výsledky, zůstávají problematické i pro pleťové odstíny (chyba pro ně však není zdaleka tak výrazná).

#### **5.2.4 Ukázka korekce vybraných fotografií**

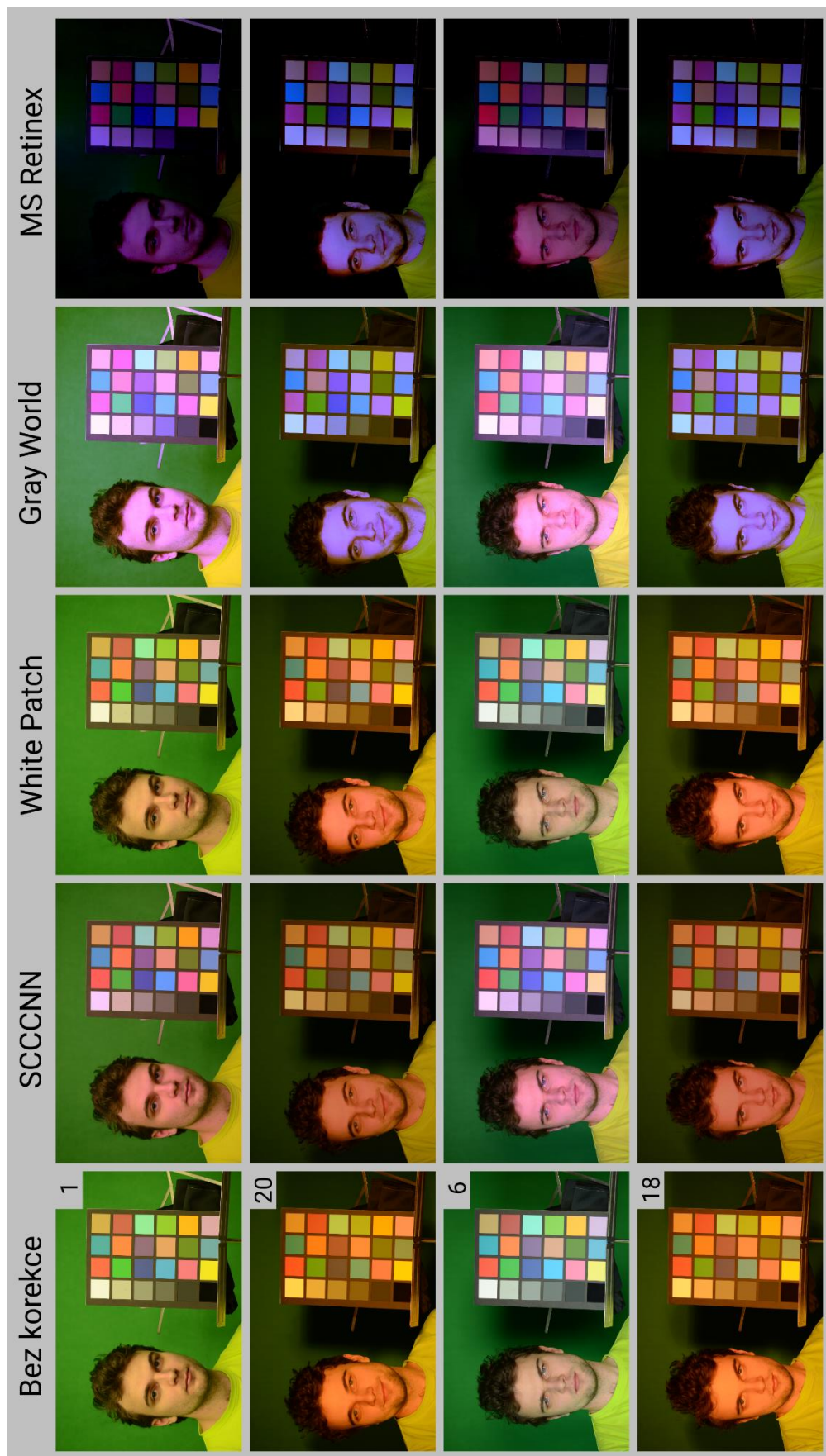
Pro vizuální ukázkou korekce pleťové barvy jsem vybral čtyři fotografie (ostatní veškeré korigované fotografie uvádím v digitální příloze).

Fotografie číslo 1 je nejlépe korigovanou fotografií SCCNN algoritmem. Zároveň dosahuje dobrých výsledků i pro algoritmy ostatních úprav. Scéna je nasvícena přirozeným světlem z místnosti a bez jakýchkoli filtrů, tedy se jedná o fotografii snad nejběžnější z celého souboru testovacích snímků. Můžeme si povšimnout selhání metody Gray world, neboť scéna neodpovídá průměrné odrazivosti.

Fotografie číslo 20 má taktéž dobré výsledky pro korekci algoritmem SCCNN, ale naopak nedosahuje tak dobrých výsledků u ostatních korekcí. Dle tab. 2 se jedná o nejoranžovější osvětlení scény, neboť je použit oranžový filtr s nejnižším výkonem halogenové lampy.

Fotografie 18 pak s velice podobným charakterem osvětlení dosahuje horších výsledků korekcí u všech algoritmů. Můžeme pozorovat, že obraz je i pro SCCNN začervenalý, což odpovídá hnědému filtru osvětlení. I přesto však algoritmus SCCNN dosahuje nejlepších výsledků ze všech (stejně jako v případě fotografie číslo 20) a velice dobře tyto teplé barvy koriguje.

Naopak u fotografie 6, u které dochází k razantnímu zlepšení korekce všech ostatních algoritmů, SCCNN dosahuje jeho průměrných výsledků. Tato scéna byla osvětlena modrým světlem zářivek bez filtrů, a tedy potvrzuje to, že SCCNN algoritmus není příliš vhodný na modře osvětlené scény (jak vycházelo i z vyhodnocení barev fotografií 2, 5, 9).

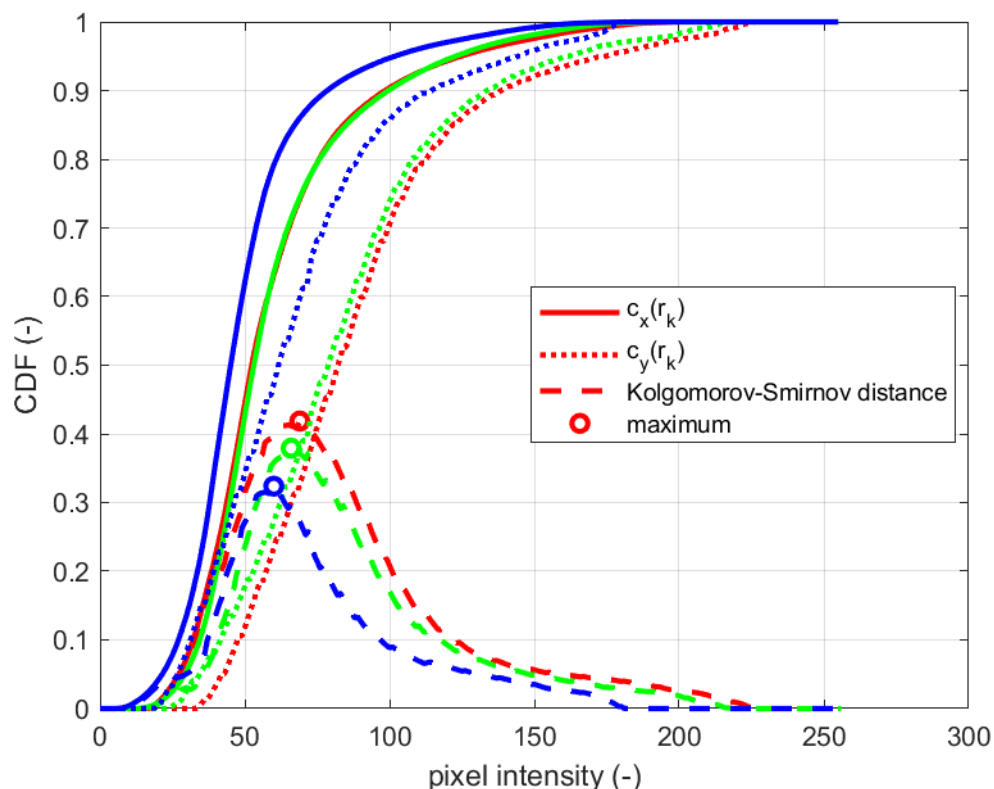


Obr. 11: Ukázka korekce vybraných fotografií

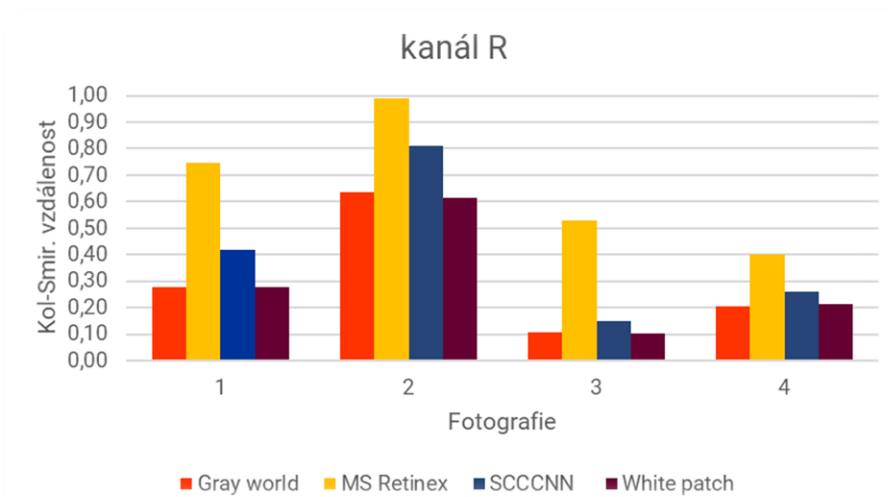
## 6 Aplikace algoritmů na reálná data

Mimo mé testovací fotografie jsem vybranými algoritmy korigoval i reálná data, konkrétně pak digitální faksimili referenční kopie díla Rozmarné léto. Digitální faksimilie je přesná kopie díla, která ale zachycuje i jeho vývoj v čase – tedy jeho poškození apod. K dispozici jsem měl i digitálně restaurovaný autorizát, tedy výsledek procesu digitálního restaurování a ten jsem mohl využít jako referenci pro úpravu barev digitální faksimilie.

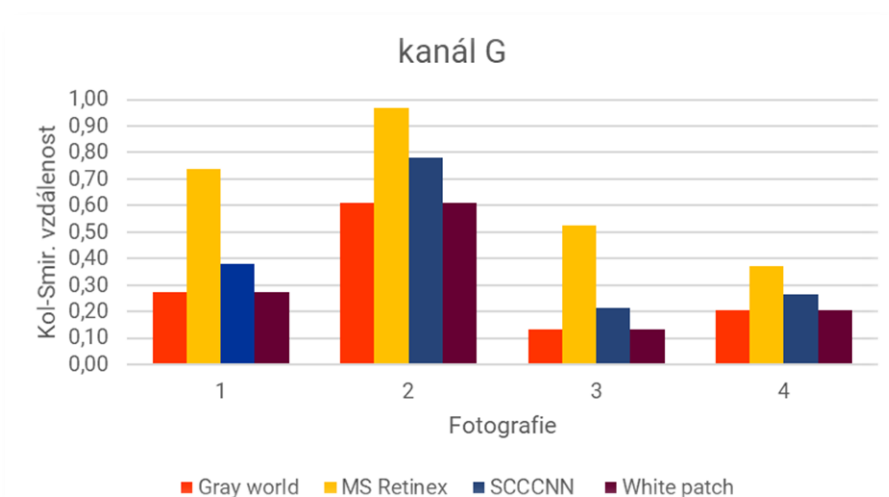
Pro vyhodnocení podobnosti mé reference a korigovaného algoritmu jsem pak dle [23] použil metody Kolmogorovy – Smirnovovy vzdálenosti pro jednotlivé RGB kanály. To je maximální rozdíl v průběhu kumulativní distribuční funkce histogramů jednotlivých snímků. Pro vyhodnocení jsem používal skript v programu MATLAB dle práce [23].



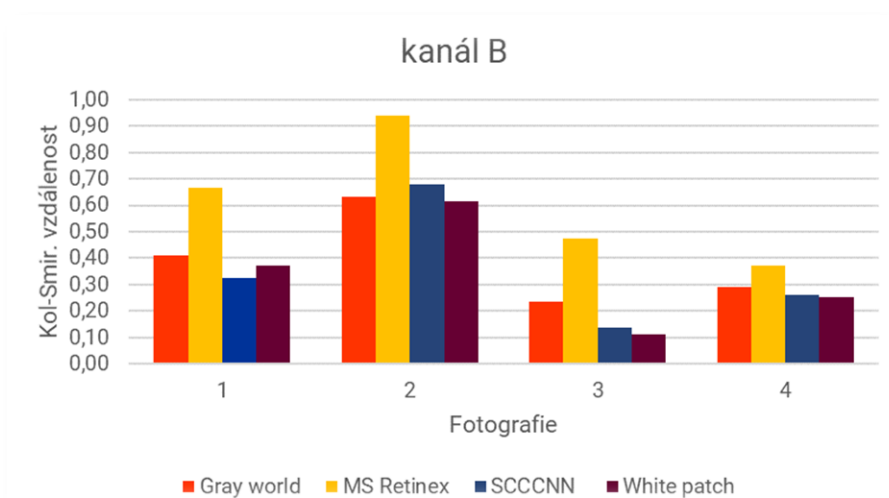
Graf 7: Kumulativní distribuční funkce jednotlivých barevných kanálů první korigované fotografie algoritmem SCCNN a znázorněný postup výpočtu Kolmogorovy-Smirnovovy vzdálenosti



Graf 8: Kolmogorovova-Smirnovova vzdálenost červeného kanálu pro jednotlivé fotografie



Graf 9: Kolmogorovova-Smirnovova vzdálenost zeleného kanálu pro jednotlivé fotografie



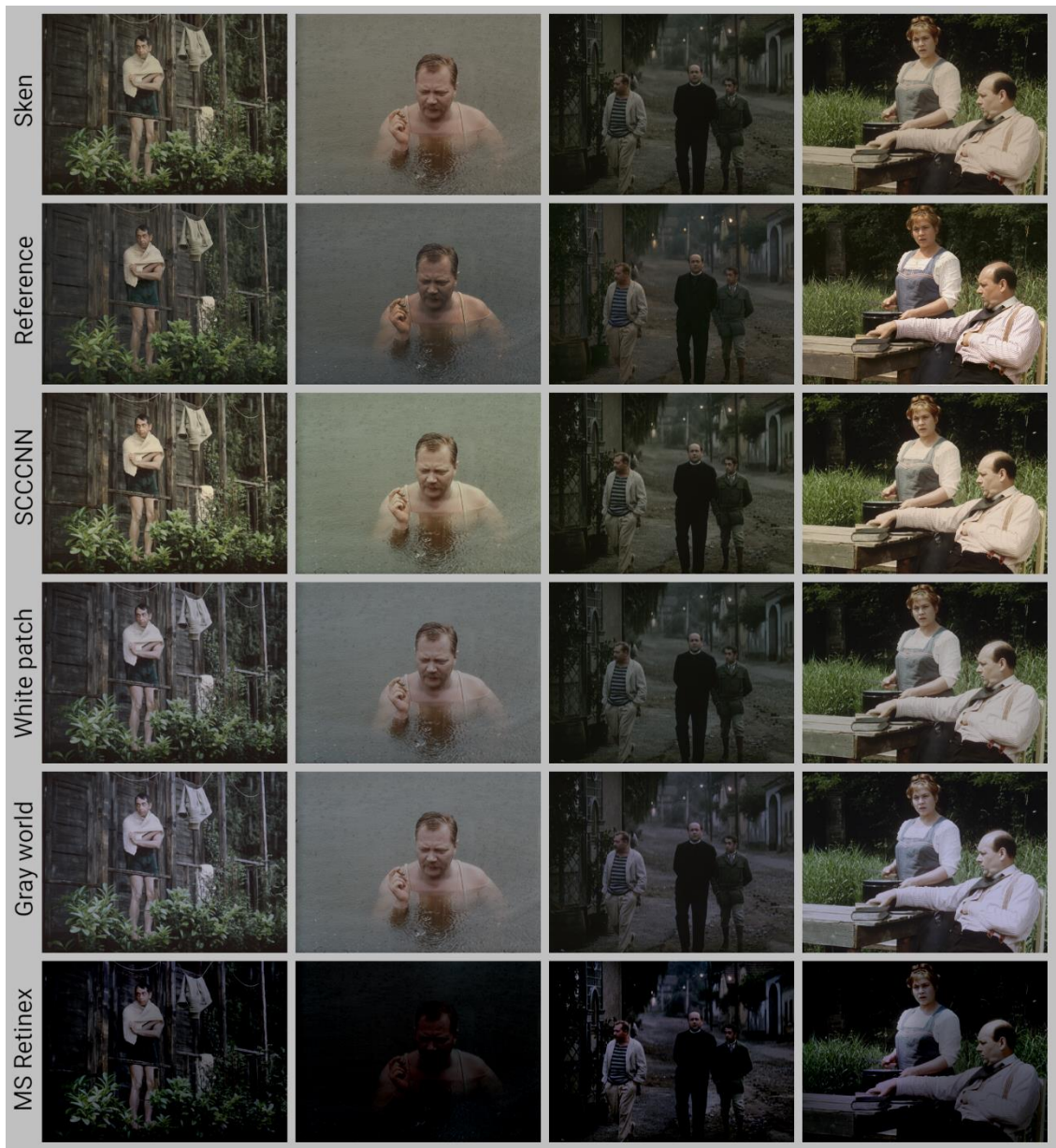
Graf 10: Kolmogorovova-Smirnovova vzdálenost modrého kanálu pro jednotlivé fotografie

Grafy 8 až 10 (jejich přesné tabulkové hodnoty uvádím v příloze G) zobrazují Kolmogorovovu-Smirnovovu vzdálenost pro jednotlivé kanály jednotlivých korigovaných fotografií. Z tabulek je patrné, že metoda Multi scale Retinex znovu selhává oproti ostatním korekcím. Tyto ostatní korekce pak dosahují různých výsledků pro různé kanály. SCCCNN si vede nejlépe v modrém kanálu, ale v červeném a zeleném dosahuje horších výsledků než Gray world nebo White patch. Tyto dvě metody také mají totožný zelený kanál, což je zřejmé z grafu 9. To je dané z jejich společného principu, který bere zelený kanál jako referenční.

Jelikož naším referenčním snímkem byl obraz upravený čistě subjektivně, autorem díla, neshodují se tyto výsledky s mým objektivním algoritmem. Technické a objektivní parametry totiž ne vždy odpovídají uměleckému záměru díla, proto by nebylo možné pro kvalitní barevnou korekci přepisu použít ani jeden ze zmíněných algoritmů. Výsledky SCCCNN korekce jsou také samozřejmě silně ovlivněny sadou fotografií, na kterých se neuronová síť trénuje. Pokud by tato sada odpovídala více konkrétnímu uměleckému záměru autora než klasickým technicky dobře vyváženým fotografiím, věřím, že by tento algoritmus dosahoval vždy lepších výsledků směrem k takovému uměleckému záměru.

Na obr. 12 pak je zřetelné, že SCCCNN algoritmus zabarvuje výsledek příliš do žluto-zelena oproti referenci. Metody White patch a Gray world pak naopak mají výsledek spíše fialovo-modrý, což více odpovídá kýženému uměleckému záměru autora.

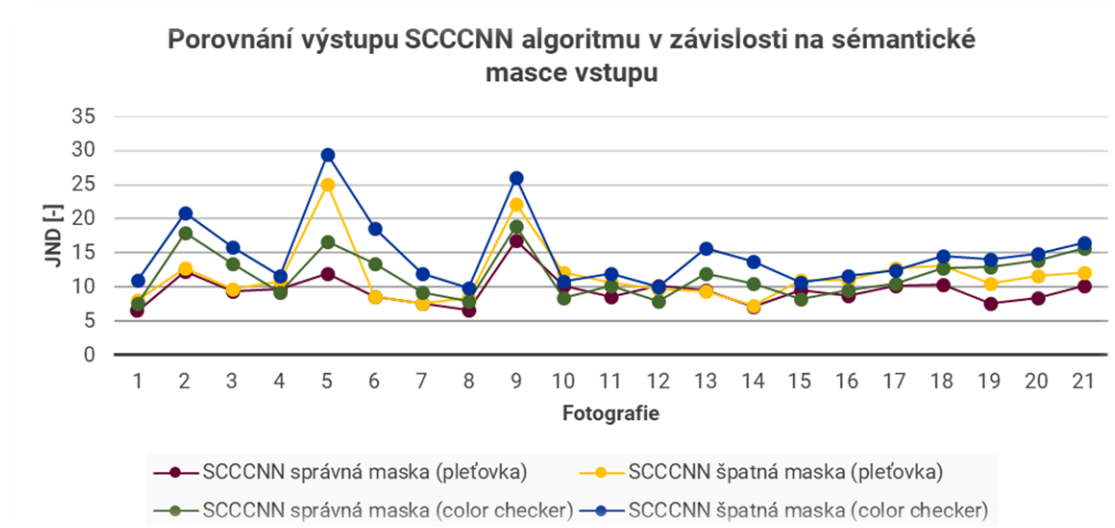




Obr 12: Aplikace algoritmů na reálná data

## 7 Sémantická maska u CNNSS algoritmu

Afifi ve svém článku [23] popisujícím algoritmus CNNSS říká, že volba nesprávné sémantické masky může až o 40 % nepříznivě ovlivnit výslednou barevnou korekci. V rámci méj práce jsem se tedy rozhodl zjistit, jaký dopad má maska při použití tohoto algoritmu na moje testovací data a jak moc se budou výsledky lišit oproti zpracování s maskou správnou.



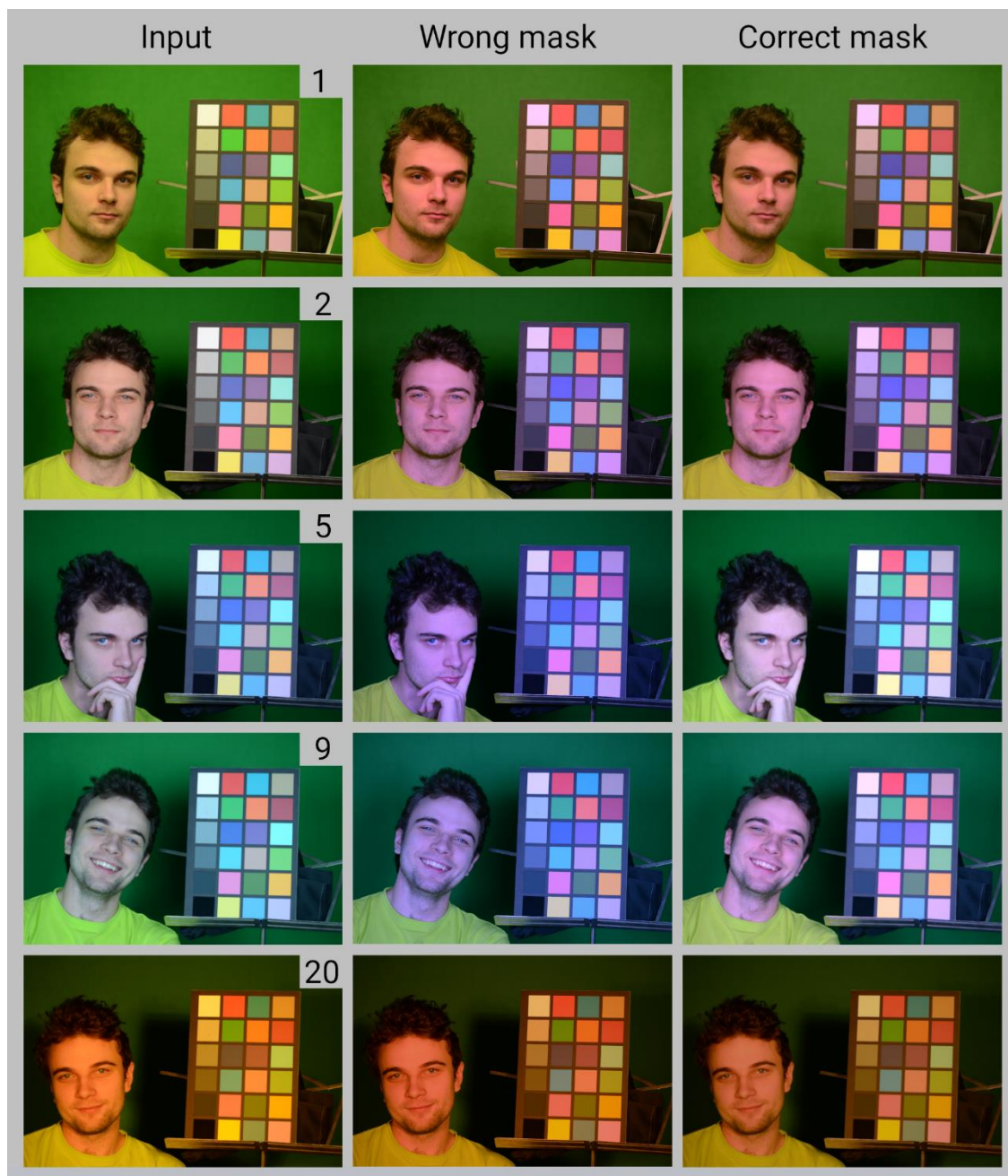
Graf 11: JND objektivního vyhodnocení SCCCN korekce se špatnou i správnou sémantickou maskou

	Průměr	Medián
<b>SCCCN správná maska (pleťovka)</b>	9,5	9,5
<b>SCCCN špatná maska (pleťovka)</b>	11,7	10,8
<b>SCCCN správná maska (color checker)</b>	11,8	10,6
<b>SCCCN špatná maska (color checker)</b>	14,9	13,7

Tabulka 7: Průměr a medián JND algoritmu SCCCN se správnou a nepsávnou sémantickou maskou

Z grafu 11 je zřejmé, že sémantická maska má opravdu značný pozitivní vliv na výstup SCCCN algoritmu. I přesto, že průměrná hodnota nedosahuje dle tab. 7 zlepšení o 40 %, které bylo zmiňováno ve článku [23], pro některé konkrétní, a právě problematické fotografie (číslo 5, 9) dosáhlo zlepšení až 100 % JND.

Sémantická maska je tedy určitě pro barevnou stálost přínosná, nejvíce pak právě při korekci problematických snímků.



Obr. 13: Ukázka korekce se špatně a dobře zvolenou sémantickou maskou



Obr. 14: Ukázka korekce reálných snímků se špatně a dobře zvolenou sémantickou maskou

Na obr. 14 vidíme metodu SCCNN aplikovanou na reálné scény v porovnání s referenčním snímkem a skenem filmové kopie. I zde je patrný velký rozdíl korekce barev se správnou maskou oproti masce nesprávné.

## 8 Závěr

V této práci jsem provedl studii metod korekce barev a vyvážení bílé a poté porovnal účinnost korekce barev čtyř popsaných algoritmů. Konkrétně pak algoritmy Gray world, White patch, Multi scale Retinex a Afifiho algoritmus SCCNN – zástupce strojového učení. Z toho první tři zmíněné algoritmy jsem realizoval vlastním kódem v programu MATLAB.

Navrhl jsem dva druhy objektivního vyhodnocení účinnosti algoritmů. Jedno zaměřené na přesnost přenesení konkrétních barev color checkeru. Druhé vyhodnocení se zaměřovalo na věrnost podání barevných odstínů pleti.

V obou vyhodnoceních jsem využil prostoru CIELAB k získání JND vzdálenosti korigovaných barev oproti reálným. Tyto metody pro objektivní vyhodnocení jsem realizoval v programu MATLAB. Jako vstupní data jsem sejmul snímky testovací scény za různého osvětlení v laboratorním prostředí. Tyto snímky jsem poté vyhodnocoval.

Ve vyhodnocení věrnosti přenesení barev color checkeru jsem pozoroval nejlepší výsledky u korekcí White patch a SCCNN. Korekce White patch měla silnou korelaci se vstupními daty, kdežto výstupy SCCNN algoritmu byly méně předvídatelné. Ukázalo se, že SCCNN algoritmus má nejmenší účinnost pro scénu nasvícenou výrazně modrým světlem.

Ve vyhodnocení věrnosti podání barevných odstínů pleti algoritmus SCCNN zaznamenal nejkonstantnější výsledků. Oproti ostatním metodám nebyl závislý na vstupních datech a dosahoval stejných JND výsledků pro většinu testovacích osvětlení. K nepatrnému zhoršení jeho účinnosti došlo znovu u modře nasvícených scén.

Testovací algoritmy jsem poté aplikoval i na reálná data, konkrétně pak na sken filmové kopie filmu Rozmarné léto. Zde nešlo výsledky objektivně vyhodnocovat, neboť jako reference mi sloužil obraz subjektivně korigovaný tvůrcem díla. Výsledky algoritmů jsem tak porovnal dle certifikované metodiky stanovení korelace mezi kolorimetrickými parametry u filmového obrazu a úrovněmi v R, G, B kanálech u digitálních obrazových souborů.

Nakonec jsem porovnal výsledky algoritmu SCCNN se správnou sémantickou maskou a bez ní. Ukázalo se, že sémantická maska má pozitivní vliv na účinnost korekce tímto algoritmem a největší vliv pak má právě u problematických scén nasvícených modrým světlem. Ukazuje se tak, že sémantická informace je jistě důležitá pro korekci barev obrazu.

## 9 Použitá literatura

- [1] V. Vít, *Televizní technika, přenosové barevné soustavy*, Praha: BEN - Technická literatura, 1997.
- [2] J. Clement, „Sciencenordic,“ 2018. [Online]. Available: <http://sciencenordic.com/eyes-look-brain-sees>.
- [3] D. H. Foster, „Color constancy,“ 2010.
- [4] J. Štěpánová, *Korekce vyvážení bílé v digitálním videu*, Praha: ČVUT, FEL, 2014.
- [5] J. Diviš, *Analýza metod vyvážení bílé*, Praha: ČVUT, FEL, 2014.
- [6] G. M. Qaryouti, Z. A. Alqadi a M. K. A. Zalata, „A Novel Method for Color Image Recognition,“ *International Journal of Computer Science and Mobile Computing*, 2016.
- [7] „Carnegie Mellon University,“ 2017. [Online]. Available: <https://www.cs.cmu.edu/afs/cs/academic/class/15494-s18/labs/lab7/lab7.html>.
- [8] S. Westland, C. Ripamonti a V. Cheung, *Computational Colour Science using MATLAB*, A John Wiley & Sons, Ltd., Publication, 2012.
- [9] W. Mokrzycki a M. Tatol, „Color difference Delta E - A survey,“ *Machine Graphics and Vision*, 2011.
- [10] A. Gijsenij, „Color Constancy Using Natural Image Statistics and Scene Semantics,“ 2011.
- [11] S. Ratnasingam, S. Collins a J. Hernández-Andrés, „Analysis of colour constancy algorithms using the knowledge of variation of correlated colour temperature of daylight with solar elevation,“ 2013.
- [12] V. Agarwal, „Ridge Regression Approach to Color Constancy,“ 2005.
- [13] S. I. Basanagoudar, C. M. V a M. Z. Kurian, „Medical Image Enhancement using Multi Scale Retinex Algorithm with Gaussian and Laplacian surround functions,“ 2015.
- [14] N. L. E. a. P. O. team, „How Landsat Images are Made,“ 2006.
- [15] M. Jícha a J. Šofr, „Živý film, Digitalizace Metodou DRA,“ Praha, 2016.

- [16] V. Agarwal, B. R. Abidi, A. Koschan a M. A. Abidi, „An Overview of Color Constancy Algorithms,” *Journal of Pattern Recognition Research*, 2006.
- [17] S. D. Hordley, „Scene Illuminant Estimation: Past, Present and Future,” University of East Anglia, Norwich.
- [18] V. Agarwala, A. V. Gribokb a M. A. Abidic, „Machine learning approach to color constancy,” 2007.
- [19] A. Gijsenij, „Computational Color Constancy: Survey and Experiments,” 2011.
- [20] G. Hines, Z.-u. Rahman, D. Jobson a G. Woodell, „Single-Scale Retinex Using Digital Signal Processors”.
- [21] A. B. Petro, C. Sbert a J.-M. Morel, „Multiscale Retinex,” *Image Processing On Line*, 16 04 2014.
- [22] M. Prabhu, V. Sarumathi a C. Vaishnavi, „Implementing Histogram Equalization and Retinex Algorithms for Image Contrast Enhancement,” 2014.
- [23] M. Afifi, „Semantic White Balance: Semantic Color Constancy Using Convolutional Neural Network,” 2018.
- [24] M. Afifi, „Semantic White Balance,” 2018. [Online]. Available: <http://130.63.97.233/semanticWB/>.
- [25] S. L. Phung a D. C. A. Bouzerdoum, „Skin Segmentation Using Color Pixel Classification: Analysis and Comparison,” *IEEE transactions on pattern analysis and machine intelligence*, 2005.
- [26] G. Woodella, Z.-u. Rahmanb, D. J. Jobsona a G. Hines, „Enhanced images for checked and carry-on baggage and cargo screening”.
- [27] K. Barnard a B. Funt, „Analysis and Improvement of Multi-Scale Retinex”.
- [28] „Color math and programming code examples,” IRO Group Limited, 2018. [Online]. Available: <http://www.easyrgb.com/en/math.php>.



# 10 Přílohy

## Příloha A: MATLAB kód metody Gray world

```
clc, clear all, close all
for i = 1:21      % hromadna uprava i fotografií
    nazev = sprintf('%d', i);          % číslo fotografie
    nazevFoto = ['NAZEV' nazev '.png']; % konkrétní název
    input=double(imread(nazevFoto));   % load input photo
    [m, n, ~]=size(input);

    R = input(:,:,1);                  % exktrakce kanálů
    G = input(:,:,2);
    B = input(:,:,3);

    R_prumer = sum(sum(R))/(m*n);      % prumerne hodnoty v kanalech
    G_prumer = sum(sum(G))/(m*n);
    B_prumer = sum(sum(B))/(m*n);

    alfa = G_prumer/R_prumer;          % Volba koeficientů
    beta = G_prumer/B_prumer;          % G kanál jako referenční

    R_new = alfa*R;                    % Korekce kanálů dle koeficientů
    B_new = beta*B;

    output = ones(m, n, 3);            % Složení kanálů do výstupu
    output(:,:,1) = R_new;
    output(:,:,2) = G;
    output(:,:,3) = B_new;
    output = uint8(output);

    nazevFoto = [nazev '.JPG'];        % Pojmenovani a uložení výstupu
    imwrite(output, ['Grayworld_' nazevFoto]);
end
```

## Příloha B: MATLAB kód metody White patch Retinex

```
for i = 1:21      % hromadna uprava i fotografií
    nazev = sprintf('%d', i);          % číslo fotografie
    nazevFoto = ['NAZEV' nazev '.png']; % konkrétní název
    I = double(imread(nazevFoto));     % load input photo
    [m, n, ~] = size(I);

    R = I(:,:,1);                      % exktrakce kanálů
    G = I(:,:,2);
    B = I(:,:,3);

    R_max = double(max(max(R)));        % Maximum každého kanálu
    G_max = double(max(max(G)));
    B_max = double(max(max(B)));

    alfa = G_max / R_max;               % Volba koeficientů
    beta = G_max / B_max;               % G kanál jako referenční

    R_new = alfa*R;                    % Korekce kanálů dle koeficientů
    B_new = beta*B;
```

```

output = ones(m, n, 3);
output(:, :, 1) = R_new;
output(:, :, 2) = G;
output(:, :, 3) = B_new;
output = uint8(output);

nazevFoto = [nazev '.JPG']; % Pojmenovani a uložení výstupu
imwrite(output, ['Whitepatch_' nazevFoto]);
end

```

## Příloha C: MATLAB kód metody Multi Scale Retinex

```

for i = 1:21 % hromadna uprava i fotografií
nazev = sprintf('%d', i); % číslo fotografie
nazevFoto = ['NAZEV' nazev '.png']; % konkrétní název
I = imread(nazevFoto); % load input photo
[m, n, ~] = size(I);

c1 = 200; % Volba koeficientů
c2 = 650;
c3 = 1100;

temp1 = JardaSingleScaleRetinex(I, c1); % Jednotlivé Retinexy
temp2 = JardaSingleScaleRetinex(I, c2);
temp3 = JardaSingleScaleRetinex(I, c3);
uprava = (1/3)*temp1 + (1/3)*temp2 + (1/3)*temp3; % Soucet

nazevFoto = [nazev '.JPG'];
imwrite(uprava, ['SkenMSRetinex_' nazevFoto]);
end

function output = JardaSingleScaleRetinex(I, c)
I = double(I);
R = I(:, :, 1); % Jednotlivé kanály
G = I(:, :, 2);
B = I(:, :, 3);

% Vytvoreni funkce F = C*exp[-(x^2+y^2)/2c^2]
[m, n] = size(R);
F = ones(m, n);
[Y, X] = meshgrid(1:n, 1:m);
F = exp(-(X.^2 + Y.^2) ./ (c.^2));
norm = 1/(sum(sum(F))); % Normalizace dle koeficientu C
F = norm.*F;

% Aplikace Retinexu, konvoluce
Funkce = fftshift(fft2(F));

konvoluceR = fftshift(fft2(R));
konvoluceR = Funkce.*konvoluceR;
konvoluceR = real(ifft2(ifftshift(konvoluceR)));

konvoluceG = fftshift(fft2(G));
konvoluceG = Funkce.*konvoluceG;
konvoluceG = real(ifft2(ifftshift(konvoluceG)));

konvoluceB = fftshift(fft2(B));
konvoluceB = Funkce.*konvoluceB;
konvoluceB = real(ifft2(ifftshift(konvoluceB)));

```

```

% Aplikace Retinexu, rozdíl logaritmu
RetR = log10(double(R)) - log10(konvoluceR);
RetG = log10(double(G)) - log10(konvoluceG);
RetB = log10(double(B)) - log10(konvoluceB);

% Převod na 0-255 a spojení kanálů
output(:, :, 1) = uint8(255*RetR);
output(:, :, 2) = uint8(255*RetG);
output(:, :, 3) = uint8(255*RetB);

```

## Příloha D: MATLAB kód vyhodnocení barev color checkeru

```

% Definice etalonu barev color checkeru v LAB prostoru
barva_ref = ones(3,24);
barva_ref(:, 1) = [61.4 9 25.6];
..... % Atd. pro všechny barvy color checkeru
barva_ref(:, 21) = [9 1.1 -1.4];

%Prealokace
JND_mean = ones(1, 21);
barva_prumer = ones(3,24);

% Vyber metody:
% 1 - SCCCNN with Gamma
% 2 - Retinex White patch
% 3 - Gray world
% 4 - Original photo (no adjustment)
% 5 - SCCCNN demo z webu /prejmenovat na .png!
% 6 - MultiScaleRetinex
vyberMetody = 1;
metoda = ["SCCCNN_wGamma", "Retinex_", "Grayworld_", ...
          '', 'SCCCNN_web_', 'MSRetinex'];
metoda_char = convertStringsToChars(metoda(vyberMetody));

for i = 1:21 % hromadna uprava i fotografií
    nizevFoto = [metoda_char 'DSC_000' cislo '.JPG'];
    input = imread(nizevFoto); % Stejna defaultni velikost
    input = imresize(input, [3264,4928], 'nearest');
    [m, n, ~] = size(input);

% Vyber barev z etalonu - automaticky - v sRGB prostoru
barva_foto = ones(250,250,3,24);
ctverec = 249; % velikost ctverecku s barvou
Y = 660; % y souradnice pro vyber barvy
barva_foto(:, :, :, 1) = input(Y:(Y+ctverec), 2728:(2728+ctverec), :);
..... % Atd. pro všechny barvy color checkeru
barva_foto(:, :, :, 24) = input(Y:(Y+ctverec), 3860:(3860+ctverec), :);

% Získání průměrné barvy
for ii = 1:1:24
    barva_prumer(1,ii) = mean(mean(barva_foto(:, :, 1, ii)));
    barva_prumer(2,ii) = mean(mean(barva_foto(:, :, 2, ii)));
    barva_prumer(3,ii) = mean(mean(barva_foto(:, :, 3, ii)));
end

%Prevod na prostor LAB
barva_prumer_lab = JardaRgb2lab(barva_prumer);

```

```

% Výpočet vzdalenosti barev
E = sqrt(...
    (barva_prumer_lab(2, :) - barva_ref(2, :)).^2 + ...
    (barva_prumer_lab(3, :) - barva_ref(3, :)).^2);

% Výpočet JND a jeho průměru
JND = E./2.3;
JND_mean(i) = mean(JND);
end

```

## Příloha E: Vyhodnocení pletových barev

```

load('SouradnicePletovky.mat')
JND_mean = ones(6,21);
for i = 1:21
    if i < 10
        cislo = sprintf('%d', i);
        nazevFoto = ['SCCCNN_' 'DSC_000' cislo '.png'];
    else
        cislo = sprintf('%d', i);
        nazevFoto = ['SCCCNN_' 'DSC_00' cislo '.png'];
    end
    input = imread(nazevFoto);
    input = imresize(input, [3264,4928], 'nearest');
    %% % Vyber pletovky - ručně - Pouze jednou
    %   imshow(input)
    pletovka = input;
    %   p = ginput(2);
    %   p1max = max(p(:, 1));
    %   p2max = max(p(:, 2));
    %   p1min = min(p(:, 1));
    %   p2min = min(p(:, 2));
    %   SouradnicePletovky(i, 1) = p1max;
    %   SouradnicePletovky(i, 2) = p2max;
    %   SouradnicePletovky(i, 3) = p1min;
    %   SouradnicePletovky(i, 4) = p2min;

    % Ulozeni souradnic pletovky pro opetovne nacteni
    p1max = SouradnicePletovky(i, 1);
    p2max = SouradnicePletovky(i, 2);
    p1min = SouradnicePletovky(i, 3);
    p2min = SouradnicePletovky(i, 4);
    pletovka(:, [1:p1min], :) = [];
    pletovka([1:p2min], :, :) = [];
    pletovka(:, [uint64(p1max-p1min+1):end], :) = [];
    pletovka([uint64(p2max-p2min+1):end], :, :) = [];

    %% % Test vyberu
    %   imshow(pletovka)
    %   w = waitforbuttonpress;

    % Urceni prumerne pletovky
    pletovka_barva(1) = mean(mean(pletovka(:, :, 1)));
    pletovka_barva(2) = mean(mean(pletovka(:, :, 2)));
    pletovka_barva(3) = mean(mean(pletovka(:, :, 3)));

    % Prevod mezi barevnymi prostory
    pletovka_barvaLAB = JardaRgb2lab(pletovka_barva');
    pletovka_barvaHSV = rgb2hsv(pletovka_barva./255);

```

```

% Definice pletove barvy dle definice, V kanal ze zdroje
pletovka_HSVmin = [0 0.23 pletovka_barvaHSV(3)];
pletovka_HSVmax = [0.1389 0.68 pletovka_barvaHSV(3)];
pletovka_RGBmin = 255.*hsv2rgb(pletovka_HSVmin);
pletovka_RGBmax = 255.*hsv2rgb(pletovka_HSVmax);
pletovka_LABmin = JardaRgb2lab(pletovka_RGBmin');
pletovka_LABmax = JardaRgb2lab(pletovka_RGBmax');
deltaE = VzdalenostOdCtverce(pletovka_barvaLAB, pletovka_LABmax,
pletovka_LABmin);

% Prepocet na JND
JND = deltaE./2.3;
JND_mean(1,i) = mean(JND);

% Vyhodnoceni pro ostatni metody - stejny postup v cyklu
for ii = 2:6
    vyberMetody = ii;
    metoda = [' "SCCCNN_wGamma", "Retinex_", "Grayworld_", ...
        ', "MSRetinex_small"];
    metoda_char = convertStringsToChars(metoda(vyberMetody));

    if i < 10
        cislo = sprintf('%d', i);
        nazevFoto = [metoda_char 'DSC_000' cislo '.JPG'];
    else
        cislo = sprintf('%d', i);
        nazevFoto = [metoda_char 'DSC_00' cislo '.JPG'];
    end
    input = imread(nazevFoto);
    input = imresize(input, [3264,4928], 'nearest');

    pletovka = input;
    pletovka(:, [1:p1min], :) = [];
    pletovka([1:p2min], :, :) = [];
    pletovka(:, [uint64(p1max-p1min+1):end], :) = [];
    pletovka([uint64(p2max-p2min+1):end], :, :) = [];

    ..... % Stejný postup, kompletní kod na CD

    JND = deltaE./2.3;
    JND_mean(ii,i) = mean(JND);
end
end

```

Příloha F: Fotografie z focení testovacích snímků



Příloha G: Tabulkové hodnoty použitých grafů

**Kolmogorovova-Smirnovova vzdálenost korigovaných snímků**

Metoda korekce	Barevný kanál	Kolmogorovova-Smirnovova vzdálenost pro fotografii číslo			
		1	2	3	4
Gray world	r	0,28	0,63	0,11	0,21
	g	0,27	0,61	0,13	0,21
	b	0,41	0,63	0,24	0,29
MSRetinex	r	0,74	0,99	0,53	0,40
	g	0,74	0,97	0,52	0,37
	b	0,67	0,94	0,48	0,37
Sken	r	0,11	0,64	0,10	0,15
	g	0,10	0,58	0,12	0,15
	b	0,14	0,41	0,12	0,12
SCCCNN	r	0,42	0,81	0,15	0,26
	g	0,38	0,78	0,21	0,26
	b	0,32	0,68	0,14	0,26
White patch	r	0,28	0,61	0,10	0,21
	g	0,27	0,61	0,13	0,21
	b	0,37	0,62	0,11	0,25

### JND vyhodnocení barev color checkeru

Fogografie	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	Průměr	Medián
SCCCNN	8	18	13	9	17	13	9	8	19	8	10	8	12	11	8	10	11	13	13	14	16	11,8	10,6
White patch Retinex	11	7	7	12	16	8	9	12	14	12	13	9	6	7	7	13	14	15	15	15	14	11,2	11,7
Gray World	20	19	20	19	16	17	18	18	15	22	21	20	18	18	20	22	23	23	22	22	21	19,7	20,0
Multi Scale Retinex	22	18	20	22	13	16	18	20	12	22	22	21	16	17	19	21	23	22	22	22	20	19,5	20,2
Bez korekce	11	7	7	14	16	8	10	15	14	12	16	11	6	7	10	14	16	18	18	18	19	12,7	13,9

### JND vyhodnocení pletové barvy

Fogografie	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	Průměr	Medián
SCCCNN	6	12	9	10	12	9	8	7	17	10	9	10	10	7	10	9	10	10	8	8	10	9,5	9,5
White patch Retinex	6	2	7	12	7	1	4	8	7	12	13	10	3	5	10	13	16	17	13	16	16	9,4	10,0
Gray world	21	19	22	23	12	16	19	23	13	26	26	25	20	20	24	25	29	30	27	27	30	22,7	22,9
Multi scale Retinex	24	21	24	22	9	16	19	27	15	26	27	26	18	20	17	26	22	24	30	29	30	22,5	23,6
Bez korekce	6	2	7	12	7	1	4	9	7	12	14	11	3	6	11	13	17	19	15	18	20	10,2	10,6