

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Korekce složek barevných prostorů s využitím
trojrozměrné interpolace



2020

Vedoucí práce: RNDr. Eduard
Bartl, Ph.D.

Jiří Palacký

Studijní obor: Aplikovaná informatika,
kombinovaná forma

Bibliografické údaje

Autor: Jiří Palacký
Název práce: Korekce složek barevných prostorů s využitím trojrozměrné interpolace
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2020
Studijní obor: Aplikovaná informatika, kombinovaná forma
Vedoucí práce: RNDr. Eduard Bartl, Ph.D.
Počet stran: 42
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Jiří Palacký
Title: Correction of color space elements by using three-dimensional interpolation
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2020
Study field: Applied Computer Science, combined form
Supervisor: RNDr. Eduard Bartl, Ph.D.
Page count: 42
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Cílem bakalářské práce bylo seznámit se s jednotlivými barevnými prostory, jejich využitím a dopadem na reálná schémata. Dále vysvětlit pojem barevné korekce a její využití, využití trojrozměrné interpolace v barevné korekci a vytvoření programu provádějící barevnou korekci za pomoci trojrozměrné interpolace.

Synopsis

Objective of this bachelor's thesis was to introduce color spaces and their usage in real applications. Further explain the concept of color correction, the use of three-dimensional interpolation in color correction and the creation of a program performing color correction using three-dimensional interpolation.

Klíčová slova: Korekce barev; barevný prostor; interpolace, LUT, RGB, HSL

Keywords: Color correction; color space; interpolation, LUT, RGB, HSL

Děkuji panu doktoru Bartlovi za vedení bakalářské práce a za volnost při práci. Dále pak odbornému konzultantu panu Hejzlarovi za pomoc v krušných začátcích tvorby programu.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

| | | |
|----------|--|-----------|
| 1 | Úvod | 8 |
| 2 | Vnímání barev | 9 |
| 2.1 | Aditivní míchání barev | 9 |
| 2.2 | Subtraktivní míchání barev | 10 |
| 3 | Barevné prostory | 10 |
| 3.1 | Gamut | 10 |
| 3.2 | RGB | 11 |
| 3.3 | RGBA | 11 |
| 3.4 | CMY, CMYK | 11 |
| 3.5 | HSV, HSL | 11 |
| 3.6 | CIE XYZ | 13 |
| 4 | Barevná korekce | 14 |
| 4.1 | Úprava intenzit barevných kanálů | 14 |
| 4.2 | Úprava pomocí HSL | 14 |
| 4.3 | Úprava jasu a kontrastu | 14 |
| 4.4 | Gamma korekce | 15 |
| 4.5 | Vyvážení bílé | 16 |
| 4.6 | Barevná teplota | 16 |
| 5 | Lookup table | 17 |
| 5.1 | Limitace 1D-LUT | 17 |
| 5.2 | Struktura 3D-LUT | 17 |
| 5.3 | Limitace 3D-LUT | 17 |
| 5.4 | Interpolace | 18 |
| 5.4.1 | Lineární interpolace | 19 |
| 5.4.2 | Bilineární interpolace | 19 |
| 5.4.3 | Trilineární interpolace | 20 |
| 5.4.4 | Neuniformní LUT | 23 |
| 5.5 | Aplikace 3D-LUT | 23 |
| 6 | Program ColorCorrection | 24 |
| 6.1 | Použité technologie | 24 |
| 6.1.1 | Jazyk C# | 24 |
| 6.1.2 | NCalc | 24 |
| 6.1.3 | OxyPlot | 24 |
| 6.2 | Návrh programu | 24 |
| 6.2.1 | Logická část | 24 |
| 6.2.2 | Obsluha oken | 25 |
| 6.3 | Manuál | 26 |
| 6.3.1 | Hlavní okno - Color Correction | 26 |

| | | |
|-------|--|-----------|
| 6.3.2 | Okno - Color modification | 29 |
| 6.3.3 | Okno - Contrast modification | 30 |
| 6.3.4 | Okno - Gamma modification | 30 |
| 6.3.5 | Okno - Hue range conversion | 31 |
| 6.3.6 | Okno - Color temperature | 31 |
| 6.3.7 | Okno - Tone curve modification | 31 |
| 6.4 | Testování | 33 |
| 6.4.1 | Test - Lineární úpravy | 33 |
| 6.4.2 | Test - Úprava kontrastu | 34 |
| 6.4.3 | Test - Gamma korekce | 36 |
| 6.4.4 | Zhodnocení | 37 |
| | Závěr | 38 |
| | Conclusions | 39 |
| | A Obsah přiloženého CD/DVD | 40 |
| | Reference | 41 |

Seznam obrázků

| | | |
|----|--|----|
| 1 | Viditelné spektrum | 9 |
| 2 | Aditivní míchání barev | 10 |
| 3 | Subtraktivní míchání barev | 10 |
| 4 | Vnímaná světelnost okem a kamerou | 15 |
| 5 | Barevné teploty | 16 |
| 6 | Uniformě navzorkovaná 3D-LUT | 18 |
| 7 | Lineární interpolace | 19 |
| 8 | Bilineární interpolace | 20 |
| 9 | Trilineární interpolace | 21 |
| 10 | Hlavní okno programu | 26 |
| 11 | Ukázka .cube souboru | 27 |
| 12 | Okno - Color modification | 29 |
| 13 | Okno - Contrast modification | 30 |
| 14 | Okno - ModifyGamma | 30 |
| 15 | Okno - ColorRangeChange | 31 |
| 16 | Okno - ColorTemperature | 31 |
| 17 | Okno - Tone curve modification | 32 |
| 18 | Aplikace první operace | 33 |
| 19 | Aplikace druhé operace | 33 |
| 20 | Aplikace obou operací na obrázek | 34 |
| 21 | Srovnání výsledných obrázků | 34 |
| 22 | Contrast test koeficient | 35 |
| 23 | Srovnání výsledných obrázků contrast testu | 35 |
| 24 | Aplikace inverzní gamma korekce na obrázek | 36 |
| 25 | Aplikace gamma korekce na obrázek | 36 |
| 26 | Aplikace funkcí na obrázek | 37 |

Seznam tabulek

| | | |
|---|----------------------------------|----|
| 1 | Výsledky prvního testu | 34 |
| 2 | Výsledky druhého testu | 35 |
| 3 | Výsledky třetího testu | 37 |

Seznam vět

Seznam zdrojových kódů

1 Úvod

S příchodem moderních informačních technologií, zapříčiňující digitalizaci grafiky a neustálého zrychlování výkonu moderních počítačů, se neustále posouvají možnosti grafika využívat výpočetně a paměťově náročné operace. Tato bakalářská práce slouží jako úvod do problematiky barevné korekce za využití trojrozměrné interpolace a její aplikace. Tuto bakalářskou práci jsem si vybral z osobního zájmu k úpravám fotografií, kterému jsem se věnoval jen povrchně, jako uživatele a viděl jsem výběr, jako vhodnou volbu k rozšíření obzorů.

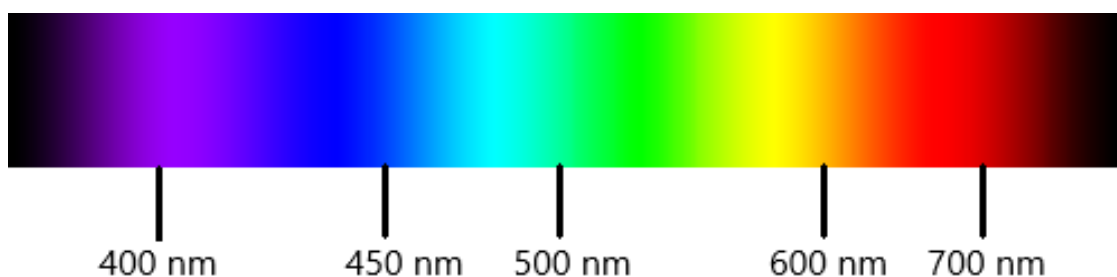
První část bakalářské práce je zaměřená na seznámení se s teorií. Nejprve se zaměříme na barvy a barevné prostory, jejich využití a případné převody. V další řadě si objasníme co všechno zahrnuje pojem **barevná korekce**. V poslední teoretické kapitole si konečně vysvětlíme co je to trojrozměrná interpolace, co k ní bude potřeba vytvořit a jak a kde se používá.

Poslední kapitola je věnována praktické části, kde si představíme program, na kterém je možné vyzkoušet poznatky nabyté v teoretické části. Program otestujeme a z výsledku testů vyvodíme závěr o použití trojrozměrné interpolace v moderních technologiích.

2 Vnímání barev

Abychom se mohli bavit o barvách a jejich korekcích v počítačové grafice je vhodné si nejdříve objasnit, co je to barva a jak jí vnímáme. Základem barevné informace je elektromagnetické záření (světlo) ze světelného zdroje, obsahující záření o vlnových délkách v tzv. viditelném spektru. Působením takového světla na povrch materiálu se poté odráží podmnožina elektromagnetického záření podle povrchové charakteristiky materiálu, která je pak zachycena lidským okem [1].

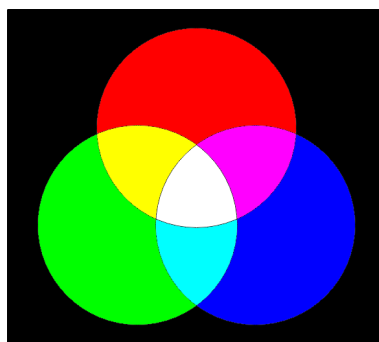
Barva by bylo možné charakterizovat jako vjem vytvářený na sítnici oka. Tato schopnost vnímat a rozeznávat barvy vzniká pomocí jednotlivých fotoreceptorů, tyčinek a čípků[1]. Tyčinky reagují především na intenzitu světla a to i na velmi malé změny intenzity za nízkého osvětlení a tedy slouží primárně jako noční vidění. Oproti tyčinkám nám zprostředkovávají barevné vidění tři druhy čípku. Každý druh čípku je citlivý na rozdílný rozsah vlnových délek elektromagnetického záření ve viditelném spektru viz. obr. č. 1. Jeden druh čípků, označován jako modrý, je citlivý na kratší vlnové délky viditelného spektra okolo 445 nm až 520 nm, druhé čípky označovány jako zelené jsou citlivé na vlnové délky okolo 535 nm až 680 nm a třetí čípky označovány jako červené začínají také okolo 535 nm a končí až 750 nm (intervaly vlnových délek se z velké části pro zelené i červené čípky překrývají, ale s tím, že zelené čípky jsou citlivější na kratší vlnové délky a červené na delší) [1].



Obrázek 1: Viditelné spektrum

2.1 Aditivní míchání barev

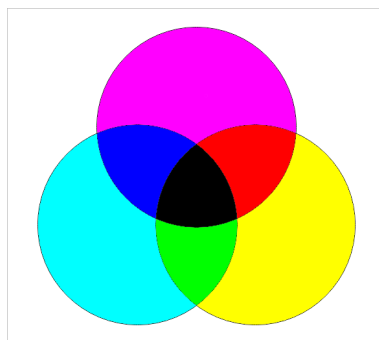
Aditivní míchání barev je založeno na sčítání jednotlivých vlnových délek světla. Nejjednodušeji se tato metoda dá popsat např. skládáním paprsků ze tří světél a to červeným, zeleným a modrým. Kombinací různých intenzit světél jsme pak schopni vytvořit velikou škálu barev. Hlavní využití aditivního míchání je v počítačových monitorech, televizních obrazovkách a v počítačové grafice.



Obrázek 2: Aditivní míchání barev

2.2 Subtraktivní míchání barev

Oproti aditivnímu míchání je subtraktivní míchání založeno na odčítávání neboli absorpci daných vlnových délek světla. Docílit absorpce se dá např. pomocí filtrů. Zeleným filtrem odstraníme červenou z barevného spektra, fialovým filtrem odstraníme zelenou a třeba žlutým filtrem odstraníme modrou část spektra. Subtraktivní míchání využívají malíři k namíchání chtěné barvy a barevný model CMY(K), který je používán k tisku.



Obrázek 3: Subtraktivní míchání barev

3 Barevné prostory

3.1 Gamut

Gamut, nebo také barevný gamut je často používaný pojem v počítačové grafice, který se váže na vše co dokáže barvy reprodukovat nebo reprezentovat. Jedná se o množinu všech barev, kterou dokáže např. některý barevný prostor reprezentovat nebo které může nějaké zařízení zobrazit či zachytit.

3.2 RGB

Nejznámější a nejpoužívanější barevný prostor, ze kterého vychází další podobné barevné prostory je založen na aditivním míchání tří barev - červené (R , *red*), zelené (G , *green*) a modré (B , *blue*). Každá složka je popsána svojí intenzitou často uváděnou v rozsahu $0 - 1$. V počítačové grafice se intenzita barevné složky nejčastěji kóduje do jednoho bytu. Kódováním intenzity barevné složky do jednoho bytu můžeme zobrazit $256^3 = 16\,777\,216$ barev. Používají se však i jiná kódování, např. pomocí 12 nebo 16 bitů [1].

3.3 RGBA

Dalším často používaným barevným prostorem je RGBA (red, green, blue a alpha), který je podobný RGB akorát navíc obohacený o další kanál *alpha*, který uchovává hodnotu (např. v rozmezí 0 až 1) o průhlednosti zobrazovaného bodu [1].

3.4 CMY, CMYK

Je barevný prostor založen oproti RGB na subtraktivním míchání barev, skládající se z barev - tyrkysová (C , *cyan*), purpurová (M , *magenta*), žlutá (Y , *yellow*) a často přidávaná černá (K , *key* nebo *black*). Hlavním důvodem vzniku CMY prostoru byla realita, že tiskárny tisknou především na papír o světlé barvy, nejčastěji bílé. Složením všech tří složek vznikne hodně velmi hnědá barva ale ne černá. Bylo tedy zapotřebí přidat do modelu ještě černou složku [10].

3.5 HSV, HSL

Oba barevné prostory reprezentují alternativní zobrazení barevného modelu RGB. Hlavní myšlenkou pro vznik obou modelů je snaha přizpůsobit model lidskému vnímání barev. Oba prostory HSL i HSV[3] definují barvu pomocí tří složek.

Pro HSV jsou to složky barevný tón (H , *hue*), sytost (S , *saturation*) a jasová hodnota (V , *value*) a pro HSL místo jasové hodnoty je složka světlosti (L , *lightness*). Prostor HSV je geometricky reprezentován jako šestiboký jehlan s vrcholem ležícím v počátku soustavy souřadnic. Jehlan je odvozen z RGB kostky (prostor RGB) pro hodnoty R , G , B z intervalu $[0, 1]$ tak, že pootočíme RGB kostku, aby bod reprezentující černou ležel v počátku soustavy souřadnic. Poté jsou ostatní barvy přemapovány do podstavy jehlanu. Bod reprezentující bílou je pak hlavní vrchol jehlanu. Složka V je výška jehlanu nabývají hodnot od 0 do 1 a složka reprezentující H je úhel měřený ze středu podstavy ke štěně jehlanu popisován v $(0^\circ, 360^\circ)$ [3].

Naproti tomu je HSL reprezentován místo jednoho jehlanu dvěma jehlany. Odvození je velice podobné jako u HSV akorát s tím rozdílem, že přemapování barev do

podstavy obou jehlanů provedem tentokrát, i bez bodu bílé barvy. Podstava obou jehlanů bude ležet ve výšce dané souřadnicí $l = 0, 5$. Vrcholem jednoho jehlanu bude bod bílé barvy ležící na souřadnici $l = 1$ a vrcholem druhého jehlanu bude bod černé barvy ležící na souřadnici $l = 0$ [3].

Pro převod z RGB do HSL/HSV je důležité si objasnit nový pojem *Chroma*. Projekcí pootočené RGB kostky, aby bod reprezentující černou ležel na počátku soustavy souřadnic, do roviny kolmé k ose L nebo V nám vznikne v počátku soustavy souřadnic pravidelný šestiúhelník s vrcholy reprezentujícími barvy známý také jako „chromatická rovina“. Chromou, budeme rozumět vzdálenost od středu vzniklého šestiúhelníku k promítnutému bodu [3].

Pro kterýkoliv bod, kde platí $R = G = B$, si můžeme všimnout, že výsledkem promítnutí do chromatické roviny, bude střed a tedy *chroma* rovna 0. Tedy pokud odebereme nebo přičteme stejnou hodnotu ke všem složkám R , G a B , tak se pohybujeme vertikálně po naší pootočené RGB kostce, aniž bychom jakkoliv ovlivnili pozici promítnutého bodu. *Chroma* bodu, který má některou složku rovnu 0, je tedy rovna větší ze dvou zbývajících složek. Pro situaci, kde není nejmenší složka nulová, vypočítáme *chromu*, jako rozdíl největší a nejmenší složky [3].

Můžeme tedy definovat:

$$\mathbf{Max} = \max(R, G, B), \quad (1)$$

$$\mathbf{Min} = \min(R, G, B), \quad (2)$$

$$\mathbf{Chroma} = \mathbf{Max} - \mathbf{Min}. \quad (3)$$

Barevný tón H se pak počítá jako poměr vzdálenosti hrany šestiúhelníku, procházejícím promítnutým bodem, a vzdáleností šestiúhelníku. V minulosti popisováno na intervalu $[0, 1)$ ale v dnešní době spíše převáděno na stupně.

$$H' = \begin{cases} \text{nedefinováno,} & \text{pro } \mathbf{Chroma} = 0, \\ \frac{G-B}{\mathbf{Chroma}} \bmod 6, & \text{pro } \mathbf{Max} = R, \\ \frac{B-R}{\mathbf{Chroma}} + 2, & \text{pro } \mathbf{Max} = G, \\ \frac{R-G}{\mathbf{Chroma}} + 4, & \text{pro } \mathbf{Max} = B, \end{cases} \quad (4)$$

$$H = 60^\circ \cdot H'. \quad (5)$$

V populární četbě se HSL i HSV graficky zobrazuje jako válec a v tomto kroku se přemapují souřadnice ze šestiúhelníku do kruhu [3].

Dalším krokem je vypočítání světelnosti.

- V HSV prostoru, se k vypočítání jasové složky V , využívá největší složka **Max**.

$$V = \mathbf{Max} \quad (6)$$

- V HSL prostoru, je světelnost L definována jako průměr největší složky **Max** a nejnižší složky **Min**.

$$L = 1/2(\mathbf{Max} + \mathbf{Min}) \quad (7)$$

Vyobrazování barev pomocí chromy by pro hodnoty $H \in [0^\circ, 360^\circ)$ $C \in [0, 1]$ a $V \in [0, 1]$ a polovina výsledných hodnot by spadla mimo zobrazitelné barvy. Výpočet saturace se tohoto problému snaží zbavit vydělením chromy maximální hodnotou chromy, dané světelnosti [3].

- Pro HSV:

$$S = \begin{cases} 0, & \text{pokud } V = 0 \\ \frac{\mathbf{Chroma}}{V}, & \text{jinak} \end{cases} \quad (8)$$

- Pro HSL:

$$S = \begin{cases} 0, & \text{pokud } L = 1 \vee L = 0 \\ \frac{\mathbf{Chroma}}{1 - |2L - 1|}, & \text{jinak} \end{cases} \quad (9)$$

3.6 CIE XYZ

V roce 1931 byl vytvořen mezinárodní standart tak, aby bylo možné barvy měřit a vyjadřovat bez subjektivního lidského vnímání. Základem jsou tři primární barvy (Červená, zelená a modrá). Pomocí kolometrických experimentů zaměřených na srovnávání vytvořených barev (skládání barev) byly stanoveny srovnávací funkce $(r(A), g(A), b(A))$ [1].

Problémem takového vyjádření barvy je, že průběh funkce červené barvy nabývá záporných hodnot, protože reálně není možné skládáním vytvořit všechny barvy jenom třemi primárními. Z toho důvodu byl vytvořený nový model CIE XYZ, ve kterém je barva určena srovnávacími funkcemi, tvořené třemi imaginárními barvami, které nenabývají nulových hodnot [1].

Následně bylo celé spektrum přetransformováno do jednotkové roviny XYZ vytvářející tzv. „spektrální podkovu“ ve které je barva určena souřadnicemi x a y , tvořící chromatický diagram. Z souřadnice bývá často zanedbávána a nazývá se osvětlení [1].

4 Barevná korekce

Abychom se mohli zabývat barevnou korekcí, je důležité si nejdříve objasnit co si pod barevnou korekcí představít. Intuitivně by se dalo říci, že se jedná o dosažení co nejvíce přirozené barevnosti avšak, ne vždy je snahou přirozená barevnost, ale barevnost požadovaná autorem [12].

Změna barev obrazu je často prováděná akce, pokud si přejeme např. navýšit jas obrazu, nebo splývajícímu obrazu upravit kontrast a docílit tak požadovaného výsledku. Nicméně požadavek na barevnou korekci nevyplývá jen z potřeb uzpůsobit si obraz tak, aby vypadal podle představ autora, ale aby byl obraz přizpůsoben pro akci některého ze zařízení (např. tiskáren), která jsou schopna vytisknout nebo zobrazit jen určitý počet barev, který se neshoduje s počtem barev obrazu. Můžeme také chtít docílit opravy barevných odchylek mezi obrazem zachyceným kamerou a reálně vnímaném obraze lidským okem.

4.1 Úprava intenzit barevných kanálů

Je asi nezákladnější a nejlehčí barevná korekce. Hodí se nám k tomu, abychom navýšili např. intezitu modré kanálu, čímž docílíme aby obraz dostal modrý nádech. Takovéto úpravy jsou však málo používané kvůli často nechtěnému zabarvení bílých bodů.

4.2 Úprava pomocí HSL

Digitální obrazy jsou nejčastěji zakódovány pomocí některé variace RGB, avšak jak už bylo v kapitole výše popsáno, lidskému vnímání je barva popsána pomocí HSL bližší. Přidávání saturace a světlosti barevným odstínům nebo zaměnění barevných odstínů za jiné je jedna z nejčastějších funkcí moderních grafických editorů. Je však důležité dát si pozor na ukládání si obrazu, protože po převodu např. modré barvy zpět do RGB, které nastavíme maximální světelnost, ztratíme informaci o odstínu a saturaci barvy.

4.3 Úprava jasu a kontrastu

Jsou základní běžně používané barevné korekce. Přidáním jasu budeme rozumět zesvětlení cílových barev a odebráním jasu naproti tomu ztmavení cílových barev. Výpočet jasu by bylo možné naivně popsat jako vážený průměr RGB složek, avšak lidské oko vnímá různým způsobem intenzitu jednotlivých barevných složek (nejcitlivější je na zelenožlutou). Pro výpočet jasu I se tedy nejčastěji

používá empirický vztah [1]:

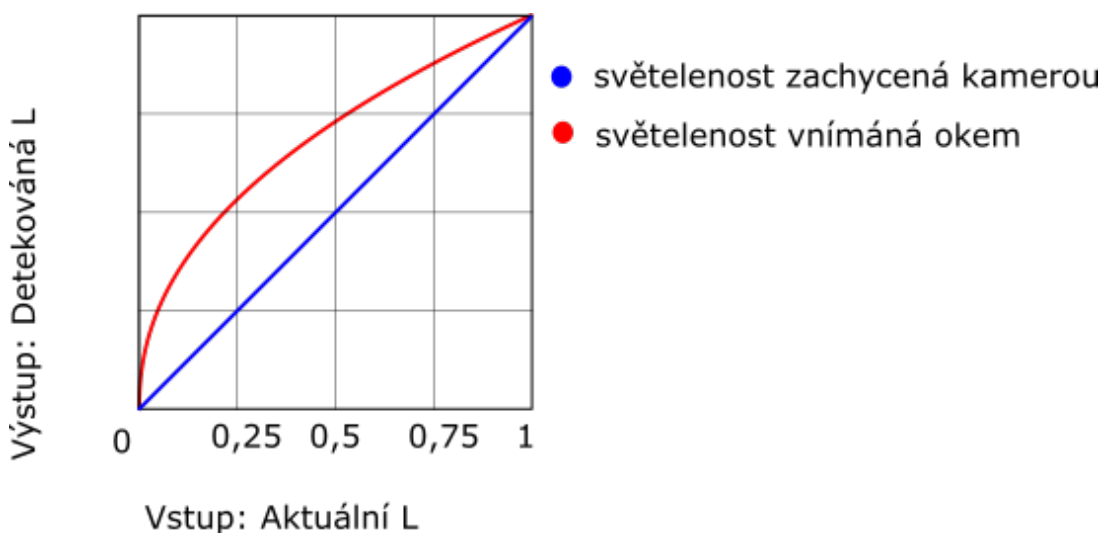
$$I = 0,229R + 0,587G + 0,114B. \quad (10)$$

Kontrastem Budeme rozumět rozlišitelnost mezi světlejšími a tmavšími barvami. Zvyšováním kontrastu docílíme zesvětlení světlých barev a zároveň ztmavení tmavých barev. Snížením kontrastu se světlé a tmavé barvy příliš nezmění, ale obraz bude působit jako by barvy začaly do sebe splývat [13].

4.4 Gamma korekce

Gamma je velice důležitá charakteristika prakticky téměř všech digitálních zobrazovacích systémů. Gamma korekce se definuje jako vztah mezi numerickou hodnotou pixelu a reálnou světlostí. Bez gammy, by odstíny zachycené kamerou, nepůsobili stejně jako by v reálném světě působili na oči.

Hlavním důvodem použití je, že lidské oči nevnímají světlo stejně jako digitální kamery, např. při dopadu dvojnásobného počtu fotonů na senzor kamery, kamera zašle dvakrát signál a lineárně se dopravuje k výsledné hodnotě pixelu. Lidské oči ovšem takto lineární navýšení jasu nevnímají a místo toho, se nám obraz jeví, jen o zlomek jasnější a s přibývajícím světlem se nelineárně navyšuje [7].



Obrázek 4: Vnímaná světelnost okem a kamerou

Funkce gamma korekce je potom definovaná takto:

$$L_{out} = L_{in}^{gamma}. \quad (11)$$

4.5 Vyvážení bílé

Je další barevná korekce, pomocí které, se snažíme docílit obrazu zachyceného např. fotoaparátem, aby vypal, tak jak by ho vnímal člověk okem. Každý světelný zdroj v závislosti na barevné teplotě, zanechává na zachycovacím zařízení daleko větší rozdíl barevnosti nežli zachytí lidské oko. Způsobující, že světlem o malé hřejivosti vznikne na zachycovacím zařízení namodralý obraz a naopak hodně hřejivým světlem vznikne mírně oranžovo-načervenalý obraz [5][16].

4.6 Barevná teplota

Barevná teplota světelného zdroje je teplota tzv. absolutně černého tělesa[15], které vyzařuje barvy podobné světelnému zdroji. Teplota barvy se stala běžnou charakteristikou viditelného světla používaná v osvětlovací technice, fotografování, filmování, počítačové grafice a ve spoustě dalších oborů. Teplota barvy má smysl pouze jen pro ty barvy, které dokáže vyzářit absolutně černé těleso za různých teplot, tudíž se jedná o světla barev jdoucích od červené do oranžové, z oranžové do žluté, z žluté do bílé a z bílé do modro-bíle viz. obr. č.16. Teplota barvy se udává v kelvinech. Světlu o barevných teplotách od 1800 K do 4000 K se říká teplé světlo, od 4000 K do 6500 K neutrální (běžné denní světlo) a od 6500 K výše studené světlo.



Obrázek 5: Barevné teploty

5 Lookup table

Lookup table (vyhledávací tabulka, zkráceně LUT)[4][2] je vyhledávací datová struktura, která pro daný vstup vrátí náležitý výstup. Vyhledávací tabulky jsou v grafice vhodné optimalizační struktury, které se využívají v momentě, kdy výpočet funkce je náročný nebo transformace prováděné nad daty jsou obtížně uložitelné (např. spousta ručních úprav barev obrazu). Nejjednodušší variantou jsou tzv. 1D-LUT (jednoprostorová LUT), které jsou charakterizovány právě jedním vstupním parametrem. Pro barevnou korekci *vstupem/výstupem* rozumíme např. barevný kanál. Muže se také stát, že vstupní hodnota padne mezi námi navzorkované hodnoty a je zapotřebí pomoci aproximační metody, v našem případě interpolace, dopočítat přibližné hodnoty.

5.1 Limitace 1D-LUT

Hlavním problémem této variace LUT je, že ne vždy je možné na-mapovat výstup pro všechny barevné operace právě pro jednotlivý vstupní parametr. Příkladem by mohla být operace, která nám převede barevný obraz do korespondujícího „*greyscale*“ obrazu. K takové operaci se používá hodnota světelnosti (luminance), která je složena z váženého průměru všech tří barevných kanálů. Jako řešení takové limitace, může sloužit 3D-LUT [4].

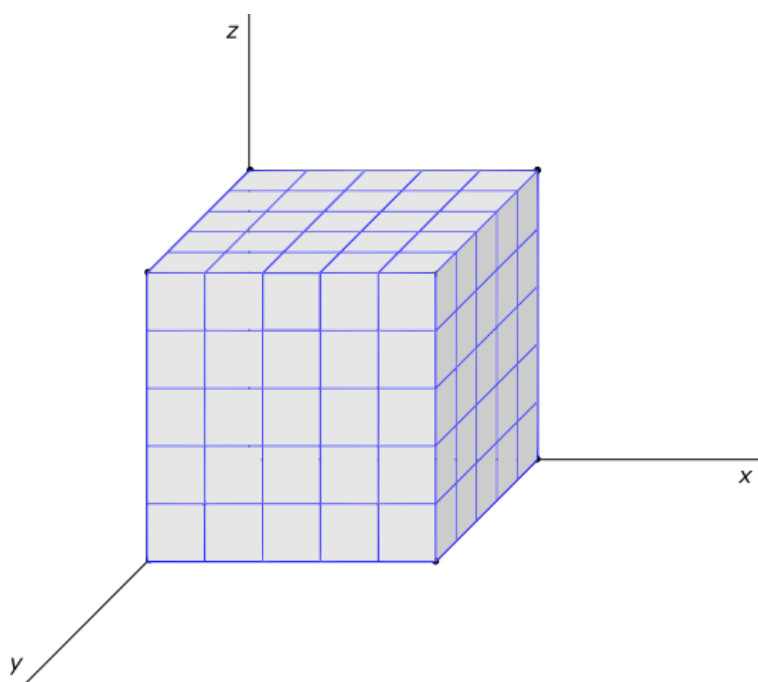
5.2 Struktura 3D-LUT

Jak už název napovídá 3D-LUT je trojrozměrná LUT, která jako vstup očekává právě tři vstupní parametry. Největším problémem 3D-LUT oproti 1D-LUT je mnohonásobně větší náročnost na paměť. K vyhledání např. požadované RGB (v této bakalářské práci se budeme zabývat barevným prostorem HSL) hodnoty za pomoci 1D-LUT je zapotřebí vytvořit 1D-LUT pro každý barevný kanál, o počtu položek B^2 (B - počet bitů na barevný kanál). Výsledný počet položek pak bude pro 24-bitové kódování do RGB obsahovat $255 \cdot 3$ a naproti tomu 3D-LUT pro stejné kódování bude obsahovat 256^3 položek [2].

Tvorba a užití 3D-LUT se nejčastěji dělí na tři části - zabalení, extrakci a výpočítání hledané hodnoty (v našem případě se bude jednat o interpolaci). Zabalení je proces, který zabalí náš výchozí barevný prostor a vhodně ho na-vzorkuje. Extrakce je další krok, ve kterém se nalezne pozice vstupního bodu a k němu nejbližší body. Body pak použijeme k výpočtu interpolace vstupního bodu.

5.3 Limitace 3D-LUT

Nejčastější používané barevné operace jako například úprava světelnosti, kontrastu atd. se dají na-mapovat, stejně tak i složitější transformace jako změna *saturation*, *hue*, nebo ještě složitější na-mapování na uživatelem nebo systémem vytvořený (pokřivený) barevný prostor.



Obrázek 6: Uniformě navzorkovaná 3D-LUT

Bohužel, ani 3D-LUT není schopna na-mapovat všechny reálně možné transformace, protože ne všechny transformace se dají jednoduše popsat přímo jako vstup/výstup funkce. Je tedy vhodné si nadefinovat podmínky transformace, které musí být dodrženy, aby bylo možné 3D-LUT použít [4].

- Počítání pixelů musí být prostorově nezávislé na pozici obrázku. Barevné operace, které jsou ovlivněné sousedními hodnotami, nejsou vyjádřitelné vyhledávací tabulkou.
- Navzorkované hodnoty barevnou transformací musí být ideálně co nejvíce spojité. Pokud navzorkované hodnoty jsou málo spojité, tak interpolace může vracet nepřipustné hodnoty.

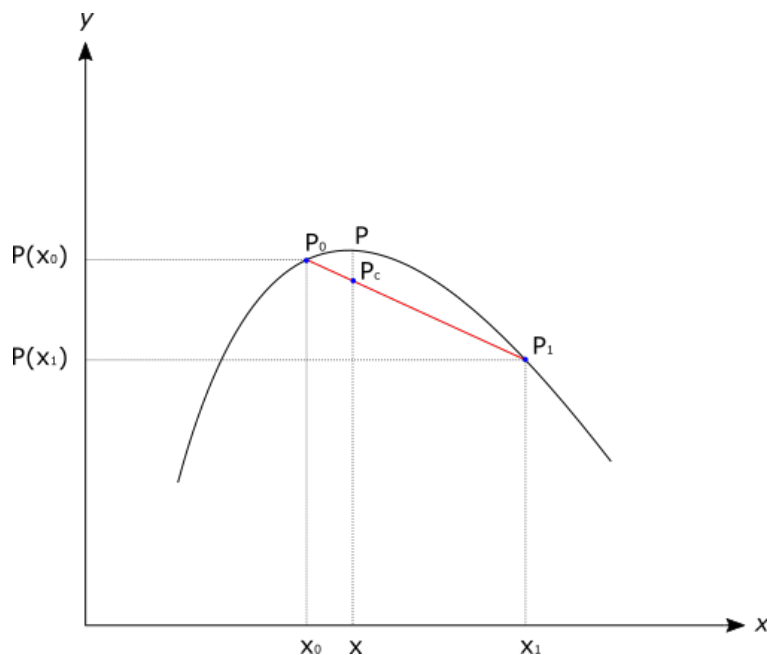
5.4 Interpolace

Obečně si pod pojmem interpolace[2][8] můžeme představit nalezení přibližné hodnoty dané funkce v kterémkoliv intervalu, kde známe hodnoty jen v některých bodech.

Jak už název práce napovídá, budeme se hlavně zabývat geometrickou trojrozměrnou interpolací. K tomu, abychom byli schopni pochopit trojrozměrnou interpolaci, si nejdříve objasníme lineární a poté bilineární interpolace, ze kterých se trojrozměrná interpolace skládá.

5.4.1 Lineární interpolace

Na názorném grafu č. 7 máme zobrazený bod P , ležící na křivce mezi body (prakticky v našem případě námi vhodně vybrané body mřížky vyhledávací tabulky) P_1 a P_2 . Interpolovaná hodnota $P_c(x)$ je lineárně úměrná poměru $(x - x_0)/(x_1 - x_0)$ [2][8].



Obrázek 7: Lineární interpolace

Z grafu si odvodíme rovnici:

$$\frac{P_c(x) - P_0}{P_1 - P_0} = \frac{x - x_0}{x_1 - x_0}. \quad (12)$$

Vyřešením rovnice pro $P_c(x)$ dostaneme:

$$P_c(x) = P_0 + (x - x_0) \frac{P_1 - P_0}{x_1 - x_0}. \quad (13)$$

Odchylka je definována jako:

$$\delta = P(x) - P_c(x). \quad (14)$$

5.4.2 Bilineární interpolace

Bilineární interpolace je funkce $P_c(x, y)$ ve dvou rozměrech a čtyřmi body P_{00} , P_{01} , P_{10} , P_{11} , aneb jak je znázorněno na grafu č. 8. Abychom mohli dostat hodnotu

P , tak nejdříve musíme získat využitím lineární interpolace, nad body P_{00} a P_{10} , bod P_0 pro konstantní y_0 [2][8].

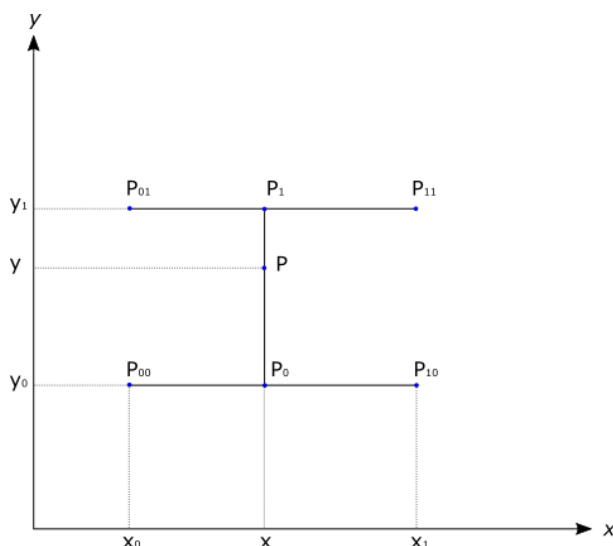
$$P_0 = P_{00} + (P_{10} - P_{00}) \frac{x - x_0}{x_1 - x_0} \quad (15)$$

Stejně tak vypočítáme P_1 pro konstantní y_1 .

$$P_1 = P_{01} + (P_{11} - P_{01}) \frac{x - x_0}{x_1 - x_0} \quad (16)$$

Poté dopočítané body využijeme k aplikování třetí lineární interpolace, tentokrát pro konstantní x :

$$P(x, y) = P_0 + (P_1 - P_0) \frac{y - y_0}{y_1 - y_0}. \quad (17)$$



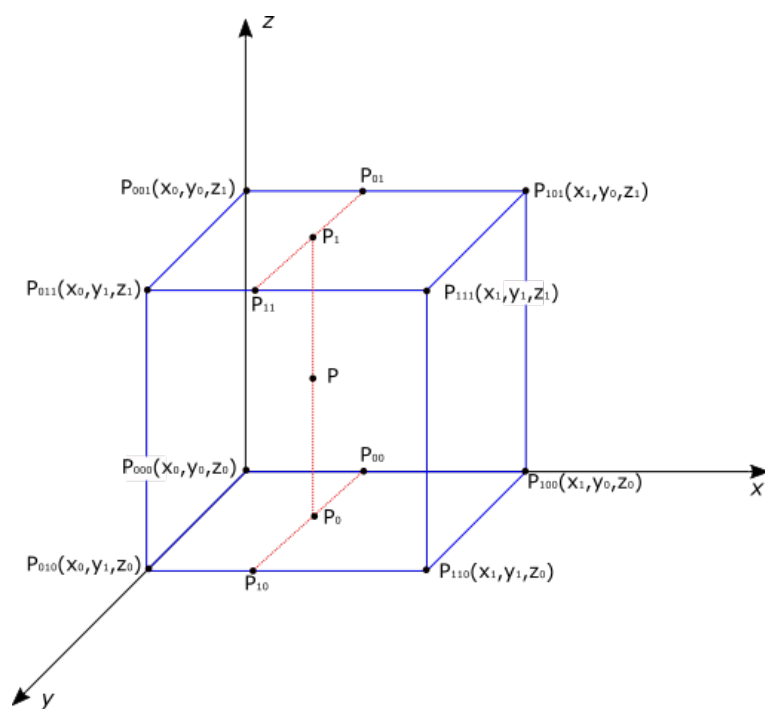
Obrázek 8: Bilineární interpolace

5.4.3 Trilineární interpolace

Trilineární interpolace je odvozena podobně jako bilineární interpolace, akorát nám přibývá další rozměr, a pro vypočítání $P(x, y, z)$ je potřeba aplikovat lineární interpolaci sedm-krát.

Rovnici trilineární interpolace si můžeme obecně zapsat takto [2]:

$$P(x, y, z) = c_0 + c_1 \Delta x + c_2 \Delta y + c_3 \Delta z + c_4 \Delta x \Delta y + c_5 \Delta y \Delta z + c_6 \Delta z \Delta x + c_7 \Delta x \Delta y \Delta z, \quad (18)$$



Obrázek 9: Trilineární interpolace

kde Δx , Δy a Δz jsou vzdálenosti bodů vzhledem k nulovému bodu P_{000} na x , y a z osách [2].

$$\Delta x = \frac{x - x_0}{x_1 - x_0} \quad (19)$$

$$\Delta y = \frac{y - y_0}{y_1 - y_0} \quad (20)$$

$$\Delta z = \frac{z - z_0}{z_1 - z_0} \quad (21)$$

Koeficienty c_j jsou dopočítány z hodnot vrcholů [2].

$$\begin{aligned}
c_0 &= P_{000}; \\
c_1 &= (P_{100} - P_{000}); \\
c_2 &= (P_{010} - P_{000}); \\
c_3 &= (P_{001} - P_{000}); \\
c_4 &= (P_{110} - P_{010} - P_{100} + P_{000}); \\
c_5 &= (P_{011} - P_{001} - P_{010} + P_{000}); \\
c_6 &= (P_{101} - P_{001} - P_{010} + P_{000}); \\
c_7 &= (P_{111} - P_{011} - P_{101} - P_{110} + P_{100} + P_{001} + P_{010} - P_{000}).
\end{aligned} \tag{22}$$

Rovnice by se dala zapsat zkráceně v lineárním tvaru [2] jako:

$$p = C^T Q_1, \tag{23}$$

Kde C je vektor koeficientů:

$$C = [c_0 \ c_1 \ c_2 \ c_3 \ c_4 \ c_5 \ c_6 \ c_7]^T, \tag{24}$$

Q_1 je vektor všech vzdáleností vzhledem k Δx , Δy a Δz , takže C můžeme napsat jako:

$$C = [1 \ \Delta x \ \Delta y \ \Delta z \ \Delta x \Delta y \ \Delta y \Delta z \ \Delta z \Delta x \ \Delta x \Delta y \Delta z]^T. \tag{25}$$

Je důležité si uvědomit, že vektory Q_1 a C musí mít stejnou velikost. Dále jsme schopni vložit koeficienty C do maticového tvaru [2] jak je ukázáno na (26).

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} P_{000} \\ P_{001} \\ P_{010} \\ P_{011} \\ P_{100} \\ P_{101} \\ P_{110} \\ P_{111} \end{bmatrix} \tag{26}$$

nebo zkráceně:

$$C = B_1 P. \tag{27}$$

Kde vektor P obsahuje všech osm vrcholů,

$$P = [P_{000} \ P_{001} \ P_{010} \ P_{011} \ P_{100} \ P_{101} \ P_{110} \ P_{111}]^T. \quad (28)$$

A matice B_1 o velikosti 8×8 , reprezentuje binární konstanty. Substitucí rovnice (27) do (23), dostaneme rovnici na výpočet cílové hodnoty p [2].

$$p = C^T Q_1 = P^T B_1^T Q_1. \quad (29)$$

5.4.4 Neuniformní LUT

Pokud se nachází převážné množství hledaných bodů jen v určitých mezích, např. průměrná fotografie má nejvíce barev v odstínech zelené, tak interpolace zapříčiňuje větší průměrnou odchylku výpočtu výsledného bodu. Pro rovnoměrně navzorkovanou LUT se odchylka může snížit jen zvětšením velikosti LUT, které není vždy žádoucí. Řešením takového problému je nerovnoměrné navzorkování LUT za pomoci vhodného mapujícího algoritmu [4].

5.5 Aplikace 3D-LUT

V dnešní době se v digitální grafice použití 3D-LUT stalo běžnou záležitostí, kterou nabízí všechny známější programy na úpravu obrazu jako je Adobe photoshop, DaVinci Resolve atd. Další využití má v grafice pomocí např. grafické karty úpravu obrazu videa v reálném čase ať už ve snaze docílit reálně vypadajícího obrazu nebo požadovaného efektu [4].

6 Program ColorCorrection

K tomu abychom mohli otestovat poznatky o barevné korekci za pomoci trojrozměrné interpolace je v rámci bakalářské práce vyvíjen program *ColorCorrection*, který slouží jako podpora pro dosažené poznatky a jejich možné reálné aplikace. V první části této kapitoly se budeme věnovat použitým technologiím a návrhu, další část bude sloužit jako manuál k užití programu a závěr bude zaměřen na testování, tedy srovnání dosažených výsledků s očekávanými výsledky a možnými optimalizacemi.

6.1 Použité technologie

6.1.1 Jazyk C#

Program je psaný na platformě .NET v jazyce C#, první myšlenka vývoje však byla pro jazyk C++, který by byl z výkonových důvodů vhodnější, avšak na základě preference, možnosti pracovat s pointerovou aritmetikou i v jazyce C# a výkonu moderních počítačových zařízení, nebylo potřeba jej zvolit.

6.1.2 NCalc

Je volně dostupná knihovna (PCL - *portable class library*), která parsuje a zpracovává textové výrazy, které si uloží do třídy *NCalc.expression*, nad kterou se dají volat metody, vypočítající výraz dle nastaveného parametru [6].

6.1.3 OxyPlot

OxyPlot je cross-platform knihovna (PLC) pro .NET vykreslující grafy. Libovolně upravitelný ovládací prvek je implementován pro WPF, Windows 8, Windows Phone, Windows Phone Silverlight, Windows Forms, Xamarin.iOS, Xamarin.Forms, Xamarin.Android a GTK# [9].

6.2 Návrh programu

Z důvodu jednoduchosti práce s obrazy spustitelný program nevyužívá RGB, ale HSL LUT, jinak by vznikalo zbytečně veliké množství zaokrouhlovacích chyb, které by nebyly vhodné pro testování správnosti vypočítaných hodnot pomocí trojrozměrné interpolace.

6.2.1 Logická část

- Třída *Algorithm*
 - Je třída obsahující pouze statické metody, které jsou převážně určeny pro matematické operace.

- Třída ***__3DLut***
 - obsahuje struktury pro uchování HSL bodu, RGB bodu a veškeré převody a práce s nimi.
 - obsahuje převážnou logickou část práce. Instance třídy si uchovává adresu LUT, a obsahuje veškerou logiku pracující s ní.
 - Stěžejní metodou celého programu je metoda provádějící trilineární interpolaci ***triInterpolateHslColor()***.
- Třída ***ImgProcessor***
 - obsahuje veškerou práci s obrazy i aplikace LUT je-li instance třídy ***__3DLut*** předaná

6.2.2 Obsluha oken

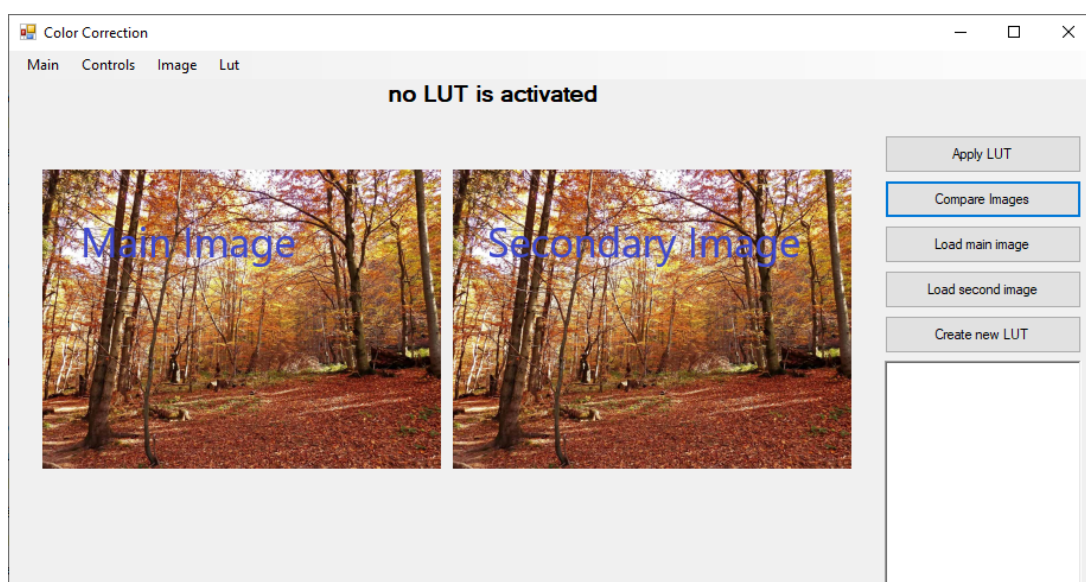
- Třída ***CCForm***
 - obstarává a vytváří hlavní okno uživatelského prostředí programu.
- Třída ***ModifyForm***
 - Obsluhuje okno podle stavu měnicí buď nahranou LUT nebo nahraný obrázek.
- Třída ***ContrastForm***
 - Obsluhuje okno podle stavu měnicí kontrast buď nahranou LUT nebo nahraný obrázek.
- Třída ***ToneFunctionForm***
 - Obsluhuje okno podle stavu měnicí buď nahranou LUT nebo nahraný obrázek za pomoci funkce.
- Třída ***ColorRangeChange***
 - Obsluhuje okno, které konvertuje barevný rozsah odstínů na jiný.
- Třída ***ModifyGamma***
 - Obsluhuje okno, které mění jas pomocí gamma funkce.
- Třída ***ColorTemperature***
 - Obsluhuje okno, které mění barevnou teplotu.

6.3 Manuál

Program je psaný na platformě .NET s využitím *Window Forms*, tedy vzhled programu připomíná běžné *Windows* aplikace. Ke spuštění programu není potřeba žádná instalace, stačí pouze spustit buildnutý *exe* soubor.

6.3.1 Hlavní okno - Color Correction

Na obrázku č. 10 je ukázáno úvodní a taktéž hlavní okno programu. Hlavní obrázek slouží pro účely aplikace načtené, nebo vytvořené LUT a druhý obrázek slouží buď k úpravě nebo jako reference k hlavnímu obrázku.



Obrázek 10: Hlavní okno programu

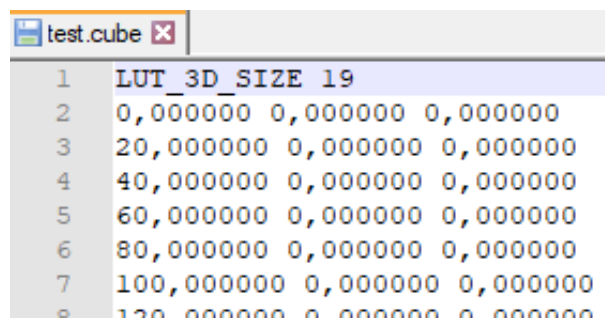
Tlačítka a textové pole na pravé straně:

- Tlačítko *Create new LUT* - Slouží k vytvoření nové LUT, kde si můžeme zvolit velikost z intervalu $\langle 4, 64 \rangle$, pro jiné hodnoty by LUT neměla smysl.
- Tlačítko *Load main image* - Slouží k nahrání hlavního obrázku z libovolné složky.
- Tlačítko *Load secondary image* - Slouží k nahrání druhého obrázku z libovolné složky.
- Tlačítko *Apply LUT* - Tlačítko aplikuje LUT na hlavní obrázek.

- **Tlačítko *Compare*** - Vypočítá rozdíl mezi hlavním a druhým obrázkem. Přidáno z analytických důvodů.
- **Textové pole** - Slouží k logování analytických informací, jako je čas provedených akcí, nebo rozdíl dvou obrázků v procentech.

Hlavní menu:

- **Main** -
 - *New LUT* - Stejně jako tlačítko, slouží k vytvoření nové LUT.
 - *Save LUT* - Slouží k uložení vytvořené nebo upravené LUT do souboru ve formátu *.cube*. Formát *.cube* je charakterizován tím, že obsahuje na prvním řádku direktivu co za trojrozměrné data, alias kostku, bude soubor obsahovat a celé číslo značící *velikost* strany kostky, tak jak je vidět na obrázku č. 11.



```

1 LUT_3D_SIZE 19
2 0,000000 0,000000 0,000000
3 20,000000 0,000000 0,000000
4 40,000000 0,000000 0,000000
5 60,000000 0,000000 0,000000
6 80,000000 0,000000 0,000000
7 100,000000 0,000000 0,000000
8 120,000000 0,000000 0,000000

```

Obrázek 11: Ukázka *.cube* souboru

- *Load LUT* - Slouží k načtení LUT ve formátu *.cube*.
- **Controls** -
 - *Apply LUT to image* - Stejně jako tlačítko slouží aplikování LUT na hlavní obrázek.
 - *Compare Images* - Stejně jako tlačítko slouží k porovnání obrázků.
 - *Load main image* - Stejná funkce jako tlačítko.
 - *Load secondary image* - Stejná funkce jako tlačítko.
 - *Save main image* - Uloží hlavní obrázek.
 - *Save secondary image* - Uloží druhý obrázek.
- **Image** -

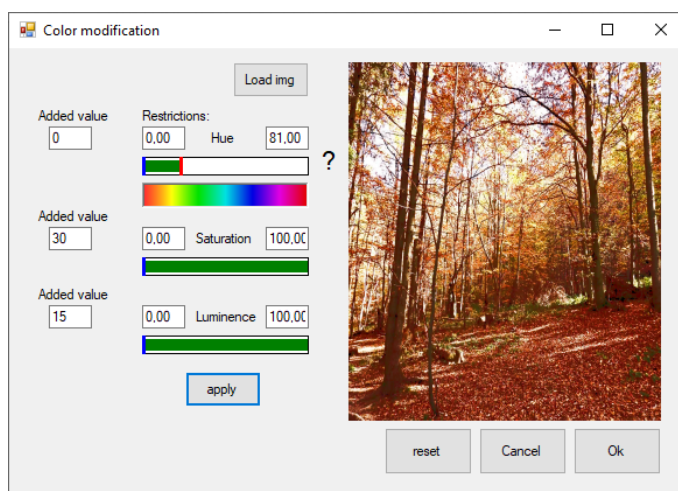
- *Modify images* - Přepne uživatele do nového okna *Color modification* v módu úpravy obrázku.
- *Modify Image contrast* - Přepne uživatele do nového okna *Contrast modification* v módu úpravy obrázku.
- *Modify gamma* - Přepne uživatele do nového okna *ModifyGamma* v módu úpravy obrázku.
- *Color temperature* - Přepne uživatele do nového okna *ColorTemperature* v módu úpravy obrázku.
- *Color range swap* - Přepne uživatele do nového okna *ColorRangeSwap* v módu úpravy obrázku.
- *Color tone curves* - Po výběru atributu (hue, saturation, luminence), který chceme modifikovat přepne uživatele do nového okna *Tone curve modification* v módu úpravy obrázku.

- **LUT** -

- *Modify LUT* - stejné jako *modify images* akorát pro LUT.
- *Modify contrast* - stejné jako *Modify Image contrast* akorát pro LUT.
- *Modify gamma* - stejné jako předchozí *Modify gamma* akorát pro LUT.
- *Color range swap* - stejné jako předchozí *Color range swap* akorát pro LUT.
- *Color tone curves* - stejné jako předchozí *Color tone curves* akorát pro LUT.

6.3.2 Okno - Color modification

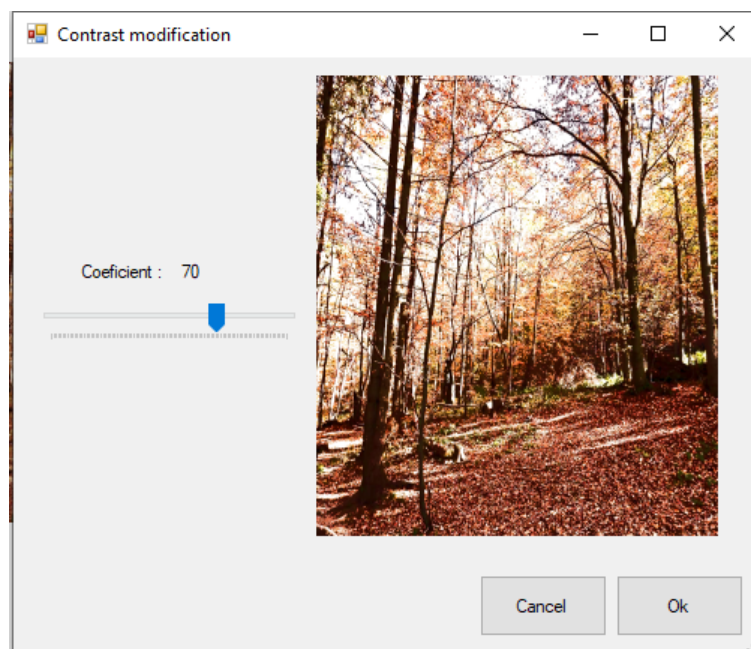
Jak je vidět na obrázku č. 12, okno Color modification, je jednoduché okno nabízející buďto úpravu obrázku nebo LUT podle nakliknutého tlačítka. Pokud už byl na hlavním okně načtený obrázek (pro LUT hlavní a pro modifikaci obrázků druhý), tak se nám zobrazí v okně. Okno nám nabízí úpravu HSL složek pomocí jednoduché matematiky a to tak, že přidá vyplněnou hodnotu dané složce, pokud bod daného obrázku nebo LUT splní všechny podmínky určené pomocí *slidebaru*. Tlačítkem *reset* zresetujeme vše do původního stavu a tlačítkem *apply* aplikujeme změny na obrázek nebo LUT. Pokud je vše jak si přejeme, potvrdíme změny tlačítkem *Ok*.



Obrázek 12: Okno - Color modification

6.3.3 Okno - Contrast modification

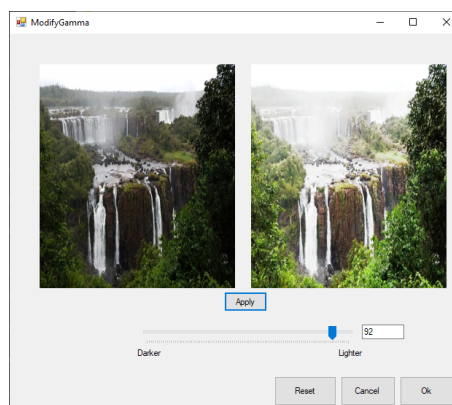
Další okno, nabízející běžně používanou úpravu kontrastu pomocí koeficientu zadaného pomocí *trackbaru*.



Obrázek 13: Okno - Contrast modification

6.3.4 Okno - Gamma modification

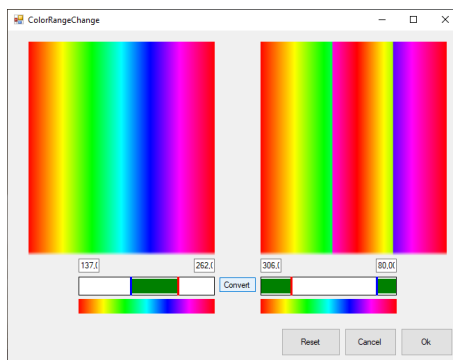
Okno nelineárně upravující jas pomocí gamma funkce zadáním koeficientu pomocí *trackbaru* nebo vepsáním do *Textboxu*. Koeficient má rozsah 0 až 100, pro nižší hodnoty než 50 ztmavuje a pro vyšší zesvětluje.



Obrázek 14: Okno - ModifyGamma

6.3.5 Okno - Hue range conversion

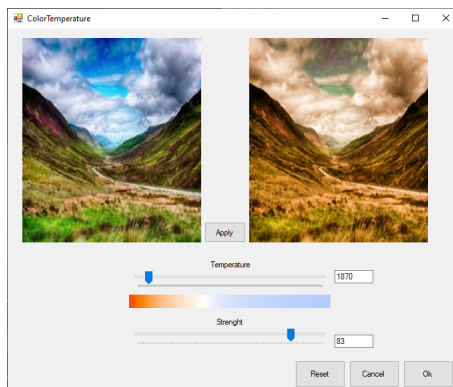
Okno, které konvertuje rozsah odstínů na jiný rozsah odstínů, jak je ukázáno na obrázku č. 15. Je taktéž možné nastavit rozsah jako doplněk, pokud nastavíme druhou hodnotu rozsahu menší než tu první. Tím docílíme, že jsme např. zkonvertovali všechny červené odstíny.



Obrázek 15: Okno - ColorRangeChange

6.3.6 Okno - Color temperature

Okno upravující barevnou teplotu pomocí dvou *trackbarů*. Jeden pro nastavení teploty a druhý pro nastavení síly s jakou bude teplota barvy aplikovaná. Vzhledem k používání HSL LUT je úprava možná jen pro obrázek.

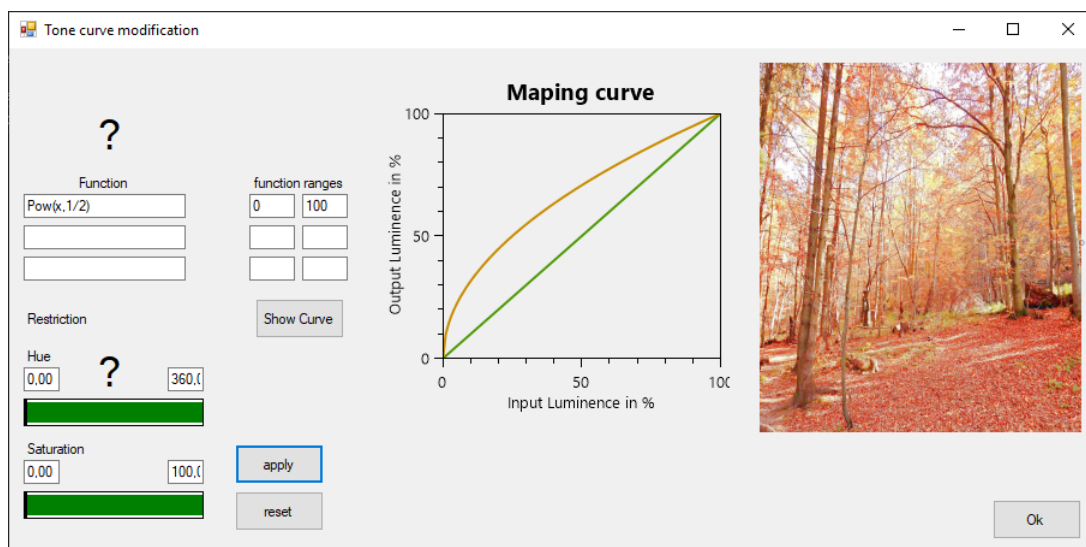


Obrázek 16: Okno - ColorTemperature

6.3.7 Okno - Tone curve modification

Je poslední a nejsložitější okno, ukázáno na obrázku č. 17. Okno je spíše určené pro experimentální účely, tedy není příliš vhodné na reálnou barevnou korekci. Účelem okna je aplikovat na vybraný atribut (hue, saturation, luminence) funkci,

kteřá bude aplikovaná na všechny hodnoty daného atributu ve vybraných mezích. Funkce se aplikuje buďto na LUT nebo na druhý obrázek podle zmáčnutého tlačítka v hlavním okně.



Obrázek 17: Okno - Tone curve modification

Funkce může obsahovat operátory a funkce vypsány níže:

- +
- -
- *
- \
- Abs(-1)
- Ceiling(1.5)
- Cos(0)
- Exp(0)
- Log(1, 10)
- Min(1, 2)
- Pow(3, 2)
- Sin(0)
- Sqrt(4)

Další použitelné funkce jsou popsány v programu. Každá funkce má jen jeden parametr x a je možné ji použít jak je uvedeno na obrázku č. 17. Je vhodné si uvědomit, že graf znázorňující funkci a omezení jsou pro saturaci a jas v procentech, zatímco hodnota parametru x bude desetinné číslo. Program také povoluje napsat až tři funkce, které jsou aplikované postupně od zhora.

6.4 Testování

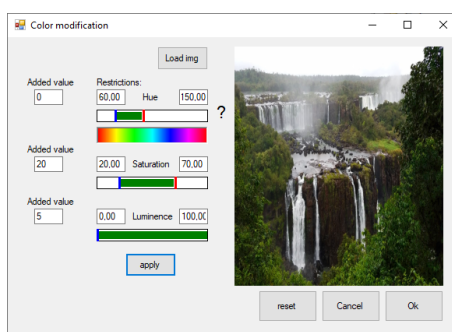
Vzhledem k tomu, že LUT používá interpolaci k výpočtu, tak jak bylo popsáno výše v kapitole o interpolaci, bude vznikat odchylka. Tuto odchylku a čas potřebný k provedení akcí si otestujeme v této kapitole. Testy budou prováděny pro LUT o velikostech 17, 33 a 64, jejichž výsledky porovnáme s referenčním obrázkem.

Odchytkou δ_{rgb} , budeme rozumět průměrný rozdíl pixelů, dvou obrázků, na stejné pozici. Další měřený údaj, uveden spíše pro zajímavost, bude čas potřebný jak pro aplikaci úpravy LUT tak pro aplikaci LUT na obrázek.

6.4.1 Test - Lineární úpravy

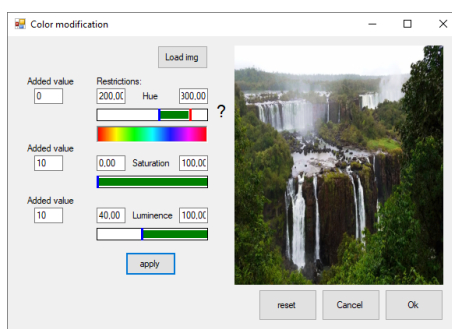
Jako první a nejjednodušší test provedeme dvě lineární úpravy LUT:

- **Operace 1** - Zkusíme navýšit saturaci a jas odstínům zelené, aby obrázek lehce ožil. obr. č. 18.



Obrázek 18: Aplikace první operace

- **Operace 2** - Pokusíme se ještě navýšit kontrast zeleně a vodopádu. obr. č. 19.



Obrázek 19: Aplikace druhé operace



Původní obrázek

obrázek po první operaci

obrázek po druhé operaci

Obrázek 20: Aplikace obou operací na obrázek

Tabulka 1: Výsledky prvního testu

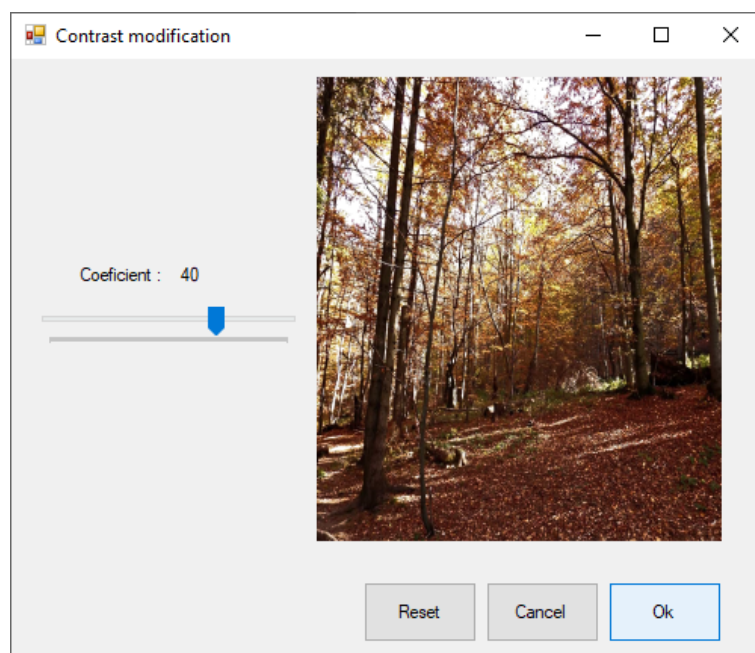
| | LUT-17 | LUT-33 | LUT-64 | bez LUT |
|-------------------------|---------|---------|---------|---------|
| δ_{rgb} | 0,5544% | 0,4961% | 0,4311% | - |
| Doba modifikace LUT | 1 ms | 1 ms | 8 ms | - |
| Doba modifikace obrázku | 656 ms | 658 ms | 672 ms | 437 ms |



Obrázek 21: Srovnání výsledných obrázků

6.4.2 Test - Úprava kontrastu

V tomto testu se zaměříme na změnu kontrastu. Funkce ne-přemapovává jas lineárně a pro námi používanou uniformní LUT je očekávaná daleko větší odchylka pro menší LUT. Test budeme provádět pro koeficient rovnající se 40ti, tak jak je ukázáno na obrázku č. 22.



Obrázek 22: Contrast test koeficient



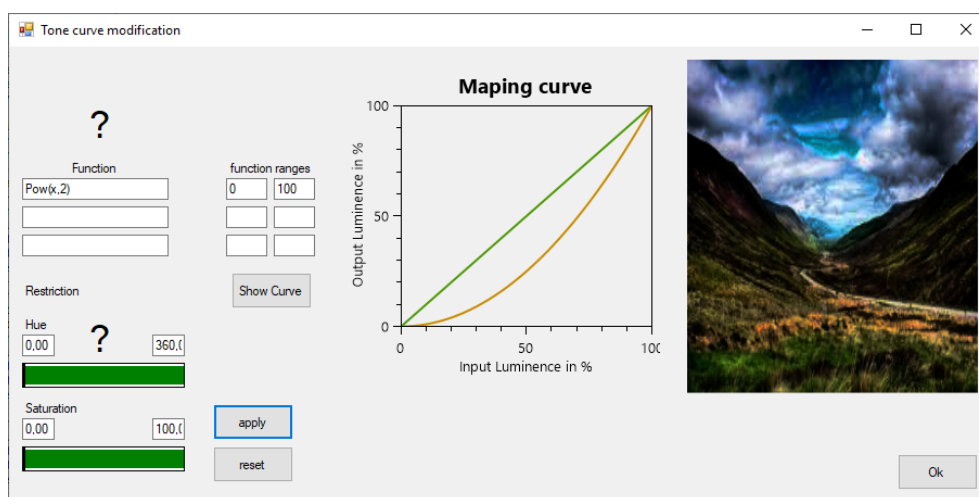
Obrázek 23: Srovnání výsledných obrázků contrast testu

Tabulka 2: Výsledky druhého testu

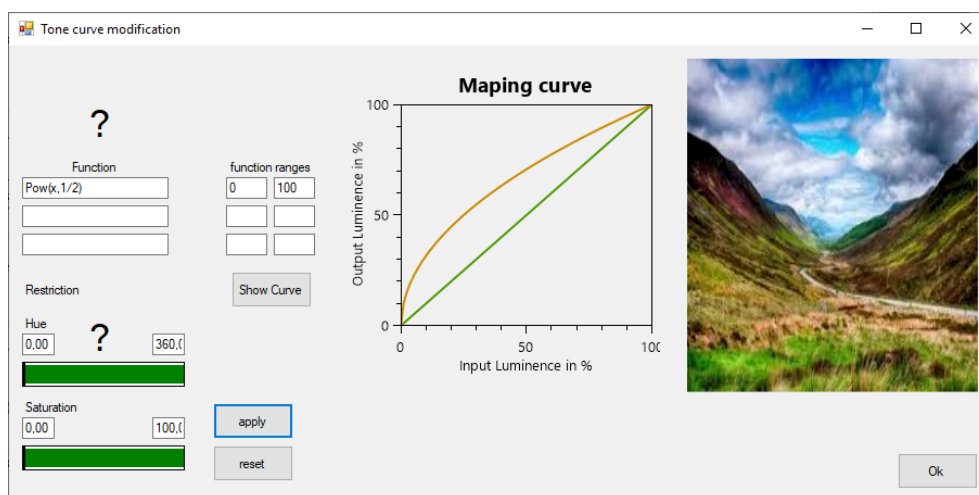
| | LUT-17 | LUT-33 | LUT-64 | bez LUT |
|-------------------------|---------|---------|---------|---------|
| δ_{rgb} | 1,0233% | 0,5674% | 0,2731% | - |
| Doba modifikace LUT | 1 ms | 1 ms | 4 ms | - |
| Doba modifikace obrázku | 861 ms | 891 ms | 897ms | 275 ms |

6.4.3 Test - Gamma korekce

Jako poslední test, provedeme gamma korekci a to tak, že použijeme funkce $f(x) = x^{1/2}$ pro x v intervalu $\langle 0, 1 \rangle$ obr.č. 25. Nejdříve si, ale obrázek upravíme inverzní funkcí $f'(x) = x^2$, taktéž pro x v intervalu $\langle 0, 1 \rangle$ obr.č. 24. Vzhledem k tomu, že gamma korekce by byla znovu pouze korekce jasu, zkusíme si ještě k tomu aplikovat úplně stejné funkce na saturaci, aby byl test komplexnější.

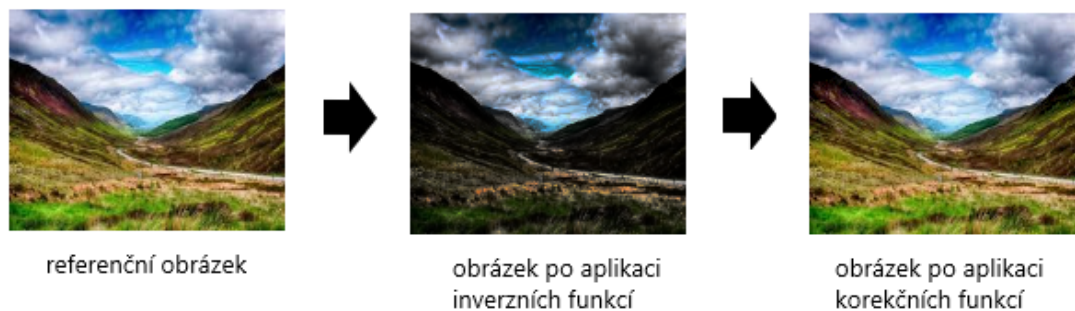


Obrázek 24: Aplikace inverzní gamma korekce na obrázek



Obrázek 25: Aplikace gamma korekce na obrázek

Abychom výsledná data podrobili větší zkoušce, použijeme na každý test dvě LUT, jednu pro inverzní funkce a další pro korekční funkce.



Obrázek 26: Aplikace funkcí na obrázek

Tabulka 3: Výsledky třetího testu

| | LUT-17 | LUT-33 | LUT-64 | bez LUT |
|-------------------------|---------|--------|---------|---------|
| δ_{rgb} | 2,0767% | 1,441% | 0,9381% | - |
| Doba modifikace LUT | 24 ms | 156 ms | 1104 ms | - |
| Doba modifikace obrázku | 88 ms | 86 ms | 86 ms | 256 ms |

6.4.4 Zhodnocení

Ze všech provedených testů, nám vyšla největší odchylka pro LUT o velikosti 17, pro třetí a nejsložitější test. Průměrná odchylka δ_{rgb} nepřekračuje 1%. Výsledky můžeme považovat jako velice uspokojivé, protože s výslednými odchylkami budou i pro nejmenší LUT, okem těžce rozpoznatelné. Důvodem takových výsledků může být fakt, že ani jedna aplikovaná funkce, nebyla rychle rostoucí, nebo klesající pro vstupní hodnoty. Dalším postřehem může být rychlost modifikace obrázku, kde až na třetí test je běžná aplikace rychlejší. To může být způsobeno aplikací parseru NCalc, který není napsán pro výpočetně náročné aplikace. Avšak i přes to nám tato informace dává dobrou představu o tom, že výhoda použití LUT, bude záviset na počtu operací použitých k barevné korekci.

Závěr

Cílem této práce bylo popsat barevnou korekci pomocí trojrozměrné interpolace. K tomu abych mohl něco takového popsat, musel jsem si zadefinovat, co to jsou barevné prostory a jaké mají využití. Bylo potřeba si taktéž objasnit nad čím, a jak bude aplikovaná trojrozměrná interpolace. Výsledkem praktické části této práce je program, na kterém si může kdokoliv, kdo má zájem o tematiku barevné korekce za pomoci LUT ozkoušet teorii nabytou čtením této práce.

Jakožto člověka, který se zabýval v minulosti úpravami fotografií a LUT vnímal jen jako předpřipravené filtry a nic více, mě téma i přes krušné začátky při vypracovávání velice nadchlo. Hlavním přínosem bylo vypracování interpolace a převodů, které jsem musel z neustálých zaokrouhlovacích chyb přepisovat, kde i malá chyba měla ohromně veliký dopad na výsledek. Velice mě překvapila, výsledná přesnost, ať už malých, či velkých LUT. Důvodem, tak velké přesnosti je nejspíš použití LUT tvořené nad HSL prostorem, na radu odborného konzultanta pana Hejzlara, který mě tím připravil o spousty odchylek.

Na bakalářskou práci, by se dalo navázat diplomovou prací, ve které bych se mohl pokusit o vytvoření bohatší GUI, pro práci a zobrazení LUT, nebo vytvořit plugin, pracující s více druhy LUT, do Adobe Photoshopu.

Conclusions

The aim of this work was to describe color correction using three-dimensional interpolation. To describe this, I had to define what color spaces are and what they are used for. It was also necessary to clarify what three-dimensional interpolation is and how to apply it. The result of the practical part of this work is a program where anyone who is interested in the topic of color correction with the help of LUT can test the theory acquired by reading this work.

Being a person who has worked with photo editing in the past and perceived LUT as just pre-made filters and nothing more, I was very excited about the topic, despite the difficult beginnings. The main benefit was the development of interpolation and transfers, which I had to rewrite from constant rounding errors, where even a small error had a tremendous impact on the result. I was very surprised by the resulting accuracy, whether small or large LUT. The reason for such a great accuracy is probably the use of LUT formed over HSL space, on the advice of expert consultant Mr. Hejzlar, who thus deprived me of many rounding errors.

The bachelor's thesis could be followed by a master's thesis in which I could try to create a richer GUI to work and display LUT, or create a plugin working with multiple types of LUT, to Adobe Photoshop.

A Obsah přiloženého CD/DVD

Na samotném konci textu práce je uveden stručný popis obsahu přiloženého CD/DVD, tj. jeho závazné adresářové struktury, důležitých souborů apod.

bin/

Adresář obsahuje spustitelný soubor programu.

doc/

Adresář obsahuje dokumentaci jak v souboru pdf, tak její zdrojové soubory.

src/

Adresář obsahuje zdrojový kód programu.

readme.txt

Soubor readme.txt obsahuje postup spuštění programu. Dále obsahuje odkazy na veškeré cizí, převzaté materiály.

Reference

- [1] Jiří Žára, Bedřich Beneš, Jiří Sochor, Petr Felkel - Moderní počítačová grafika, Computer Press Brno 2004. ISBN 80-251-0454-0.
- [2] Henry R. Kang - Computational Color Technology, SPIE Publications, 2006. ISBN-13 978-0819461193
- [3] Wikipedia. HSL and HSV. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: https://en.wikipedia.org/wiki/HSL_and_HSV
- [4] Nvidia. GPU Gems - chapter 24. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: https://developer.nvidia.com/gpugems/GPUGems2/gpugems2_chapter24.html
- [5] Wikipedia. Color balance. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: https://en.wikipedia.org/wiki/Color_balance
- [6] Sebastien Ros - NCalc dokumentace. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <https://archive.codeplex.com/?p=ncalc>
- [7] Cambridge in colour - understanding gamma correction. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <https://www.cambridgeincolour.com/tutorials/gamma-correction.htm>
- [8] Paul Borke - Paulbourke.net interpolation methods, written in 1999. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <http://paulbourke.net/miscellaneous/interpolation/>
- [9] Oystein Bjorke - oxyPlot dokumentace. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <https://oxyplot.github.io/>
- [10] Wikipedia. CMYK. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <https://cs.wikipedia.org/wiki/CMYK>
- [11] Wikipedia. Color Space. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: https://en.wikipedia.org/wiki/Color_space
- [12] Wikipedia. Color grading. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: https://en.wikipedia.org/wiki/Color_grading
- [13] Lighting Design and Simulation Knowledgebase. Lighting Design Glossary. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <https://www.schorsch.com/en/kbase/glossary/contrast.html>
- [14] Matt Pharr, Randima Fernando - GPU Gems 2, Addison-Wesley Professional 2005. ISBN-13 978-0321335593
- [15] Jaroslav Reichl and Martin Všeticka - Záření absolutně černého tělesa [online] 2019-10-08 [cit. 2019-10-08]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/538-zareni-absolutne-cerneho-telesa>

- [16] Cambridge in colour - White balance. [online] 2019-06-08 [cit. 2019-06-08]. Dostupné z: <https://www.cambridgeincolour.com/tutorials/white-balance.htm>