

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## SKLÁDÁNÍ HDR OBRAZU Z VÍCE SNÍMKŮ

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ALEŠ RASZKA

BRNO 2015



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## **SKLÁDÁNÍ HDR OBRAZU Z VÍCE SNÍMKŮ**

HDR IMAGE COMPOSITION FROM MULTIPLE IMAGES

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**ALEŠ RASZKA**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. MARTIN MUSIL**

BRNO 2015

## **Abstrakt**

Bakalářská práce se zabývá problematikou nízkého dynamického rozsahu fotoaparátů. Tento problém je řešen metodou skládání více snímků s různou expozicí do výsledného HDR snímku. Součástí práce je také představení další části procesu, kterou je tónové mapování. Problematika je zabalena do výsledné knihovny v jazyce C++, která dokáže vytvořit HDR snímek.

## **Abstract**

Bachelor thesis deals with the problems of the low dynamic range of today's cameras. This problem is solved by composing multiple images with different exposures into the final HDR image. The thesis also includes performances next part of the process, which is tone mapping. The issue is packed into the resulting library in C++ that can create HDR image.

## **Klíčová slova**

Vysoký dynamický rozsah, fotografie, knihovna, tonové mapování, síťová kamera, vícenásobná expozice.

## **Keywords**

High dynamic range, photo, library, tone mapping, network camera, multiple exposure.

## **Citace**

Aleš Raszka: Skládání HDR obrazu z více snímků, bakalářská práce, Brno, FIT VUT v Brně, 2015

# Skládání HDR obrazu z více snímků

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Martina Musila a uvedl jsem všechny zdroje, ze kterých jsem čerpal.

.....  
Aleš Raszka  
16. května 2015

## Poděkování

Tímto bych chtěl poděkovat svému vedoucímu Ing. Martinu Musilovi za cenné rady a ochotu, kterou projevoval po celou dobu tvorby této práce.

© Aleš Raszka, 2015.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

<b>1 Úvod</b>	<b>2</b>
<b>2 Problematika HDR</b>	<b>3</b>
2.1 Co je to HDR . . . . .	3
2.2 Pořízení HDR snímku . . . . .	8
2.3 Zobrazení HDR snímku . . . . .	12
2.4 Formáty ukládání . . . . .	13
2.5 Tonové mapování . . . . .	14
<b>3 Specifikace zadání</b>	<b>17</b>
<b>4 Návrh a implementace</b>	<b>19</b>
4.1 Návrh a implementace knihovny . . . . .	19
4.2 Komunikace s kamerou . . . . .	21
4.3 Grafické uživatelské rozhraní . . . . .	22
4.4 Informace z exifu . . . . .	24
4.5 Formáty ukládání . . . . .	25
<b>5 Vyhodnocení výsledků</b>	<b>27</b>
<b>6 Závěr</b>	<b>30</b>
<b>A Obsah DVD</b>	<b>33</b>

# Kapitola 1

## Úvod

Fotoaparát v dnešní době disponuje skoro každý, ať už jde o běžný fotoaparát nebo snímací čip v mobilních zařízeních, a tak se s fotografiemi setkáváme dennodenně. Mnohdy však s výslednou fotografií nejsme spokojeni, za což mohou dva faktory. Jedním z nich je umění zachytit ten správný moment, což je věc každého člověka, kterou musí mít v sobě nebo se ji naučit. Druhou věcí, kterou bohužel moc ovlivnit nemůžeme je nedokonalost techniky s kterou fotíme.

Snažit se získat fotografii stejnou jako v realitě vidí naše oči příliš nejde. Jde o to, že lidské oko je mnohem dokonalejší než technologie snímacích čipů. Nedokonalosti fotoaparátu se snažíme řešit následnou úpravou fotografií, tak abychom z fotografie získali co nejlepší výsledek. Jedním z nedostatků dnešních snímacích čipů je nedostatečný dynamický rozsah. Fotoaparáty dokáží vyprodukovat fotografii s nízkým dynamickým rozsahem, takzvaně LDR snímkem (Low dynamic range), zatím co naše oko vidí scénu s vysokým dynamickým rozsahem HDR (High dynamic range). Každý už určitě zažil situaci, kdy viděl krásnou scénu, kterou se pokoušel vyfotografovat, bohužel však výsledná fotografie vůbec neodpovídala realitě. Část fotografie byla buďto přexponovaná nebo naopak podexponovaná a fotoaparát se nedařilo nastavit tak, aby dokázal pořídit fotografii takovou jaká je reálná scéna. Právě tato situace je z velké části ovlivněna nízkým dynamickým rozsahem fotoaparátu.

Jako další problém související s touto tematikou je i zobrazování fotografií s vysokým dynamickým rozsahem. Dnešní displeje nejsou schopny takového snímky zobrazit, a proto je nutné použít techniku tónového mapování. Tónové mapování se postará o to, že vyšší dynamický rozsah je namapován do takového rozsahu, který je možné na displejích zobrazit.

Tato práce si klade za cíl seznámit čtenáře s problematikou vysokého dynamického rozsahu a metodami tónového mapování. Na základě získaných znalostí bude vytvořena knihovna, která bude schopna složit HDR fotografii ze sekvence snímků s různou expozicí. Na výsledný obraz pak bude možné aplikovat operátory tónového mapování a zobrazit výsledek. Pro demonstraci funkčnosti knihovny bude vytvořeno grafické uživatelské rozhraní pro zjednodušení obsluhy programu. Součástí programu bude také komunikace s kamerou přes síťové rozhraní, která se bude starat o získání vstupních LDR fotografií.

V následující kapitole **2** je teoretický úvod k fotografii a jaké jsou základní veličiny pro její vznik. Následně bude představena problematika HDR, různé techniky jejího vytváření a možnosti dalšího zpracování. V kapitole **3** jsou stanoveny přesné cíle této práce. Kapitola **4** se bude věnovat návrhu knihovny a implementačním detailům. Kapitola **5** se zabývá porovnáním jednotlivých tónovacích operátorů a poukazuje na výhody a nevýhody každého z nich. V závěru budou zhodnoceny dosažené výsledky a další možné pokračování této práce.

## Kapitola 2

# Problematika HDR

### 2.1 Co je to HDR

Abychom si mohli vysvětlit co je to HDR neboli vysoký dynamický rozsah, je nutné si nejdříve objasnit pár základních pojmů, které s tímto tématem souvisí. Tyto pojmy jsou základními prvky k tomu, aby bylo vůbec možné vytvořit fotografii jako takovou.

#### 2.1.1 Obecné vlastnosti fotografie

Digitální fotografie vzniká tak, že okolní světlo fotografované scény je usměrněno soustavou čoček v objektivu fotoaparátu na snímač. Ten je schopen převést dopadající světlo na elektrickou informaci nul a jedniček, které je možno uložit v určitém formátu na paměťové médium. Fotoaparáty k redukci dopadajícího světla, a tím řízením expozice, využívají následující veličiny:

##### Clona

Pojmem clona se ve fotografování označuje otvor v objektivu fotoaparátu, jehož průměr se dá měnit. Tím že měníme velikost clony, můžeme regulovat množství dopadajícího světla na snímač fotoaparátu.

Velikost clony se udává pomocí clonového čísla a je označována písmenem  $F$ . Je dána vztahem

$$F = f/d$$

, kde  $f$  je ohnisková vzdálenost a  $d$  je průměr otvoru clony. Platí tedy, že při zvětšení otvoru clony se clonové číslo zmenší a naopak. Aby byl zavedený standard clonových čísel, používá se mezinárodní řada clon. Tato řada začíná číslem 1 a každá další hodnota je násobena předchozím číslem v řadě a  $\sqrt{2}$ . Řada clonových čísel potom vypadá následovně: 1 - 1,4 - 2 - 2,8 - 4 - 5,6 - 8 - 11 - 16 - 22 -, ...

S pojmem clona souvisí i hloubka ostrosti. Hloubku ostrosti si můžeme představit jako pomyslný rozdíl vzdáleností, v níž jsou objekty na fotografii ostré. Při velkém clonovém čísle je hloubka ostrosti velká a se zmenšováním clonového čísla se zmenšuje i hloubka ostrosti.

##### Expoziční čas

Spolu s clonou je expoziční čas veličina, kterou můžeme regulovat množství dopadajícího světla na snímač fotoaparátu. Určuje dobu, po kterou může světlo dopadat na snímač.

Tuto dobu určuje zařízení zvané uzávěrka. Ta může být buďto mechanická nebo elektronická. Mechanickou uzávěrku si můžeme představit jako stínítko, které se při zmáčknutí spouště odklopí, tak aby mohlo světlo pronikat a po určité době se opět sklopí a tím zakryje senzor. Mechanická uzávěrka se v dnešních fotoaparátech kombinuje s elektronickou pro získání lepších výsledků. Zvláště v případech, kdy je zvolen velmi krátký expoziční čas a mechanická uzávěrka by se nestihla tak rychle uzavřít. Další uplatnění elektronické uzávěrky je v situacích, kdy je fotoaparát využíván na natáčení videa.

Expoziční čas je udáván v jednotkách sekund. Stejně jako v případě clony byly zavedeny standarty i zde. Expoziční čas se pohybuje v rozmezí několika desítek sekund až po tisíce sekund. Touto veličinou lze určovat nejen množství světla dopadající na snímač, ale i jak bude vyfocená scéna vypadat. Pokud zvolíme krátký čas, tak scénu budeme vnímat jako statickou. Pokud ale čas zvětšíme můžeme docílit toho efektu, že scéna bude vypadat dynamicky.

## Citlivost ISO

Tato hodnota vyjadřuje jak moc bude optický snímač citlivý na dopadající světlo. Ve skutečnosti je snímač citlivý na světlo pořád stejně. Mění se pouze způsob jakým se zpracují data o hodnotě světla. Při vyšší citlivosti ISO se množství světla získané na snímači vynásobí a tím se ve výsledku zvýší citlivost. Z toho plyne i jeden zásadní problém. Při zesílení obrazových dat se zesiluje i šum, který při snímání vzniká. A jelikož je odstup signálu a šumu stále stejný, začne se šum při vyšších hodnotách ISO viditelně projevovat.

Hodnoty ISO se pohybují v rozmezí 12 - 3200. Obvyklé hodnoty ISO jsou 100, 200, 400, 800 ... Při zdvojnásobení hodnoty ISO říkáme, že stejné světlo dopadající na snímač bude mít dva krát větší hodnotu.

## Expoziční hodnota

Množství světla dopadající na snímač se měří v tzv. absolutních jednotkách EV, neboli exposure value. Expoziční hodnota také určuje poměr mezi nejsvětlejším a nejtmaším místem na snímku. Stupnice EV je logaritmická. Udává se, že hodnota 0EV znamená 18% středně šedé tabulky při citlivosti ISO 100, cloně f 1 a expozičním času 1 vteřina [2].

Pokud je hodnota jednoho bodu o 1EV větší než druhého, znamená to tedy, že jas prvního bodu je dvojnásobně větší než jas druhého bodu. Posunutí o +1EV se dá několika způsoby. Buďto zvýšíme dvojnásobně citlivost ISO nebo zdvojnásobíme expoziční čas a nebo posuneme clonové číslo o jeden stupeň níže. Tyto vztahy ukazují následující rovnice.

$$T_{EV_{n+1}} = 2 * T_{EV_n} \quad (2.1)$$

$$ISO_{EV_{n+1}} = 2 * ISO_{EV_n} \quad (2.2)$$

$$F_{EV_{n+1}} = \frac{F_{EV_n}}{\sqrt{2}} \quad (2.3)$$

### 2.1.2 Vysoký dynamický rozsah

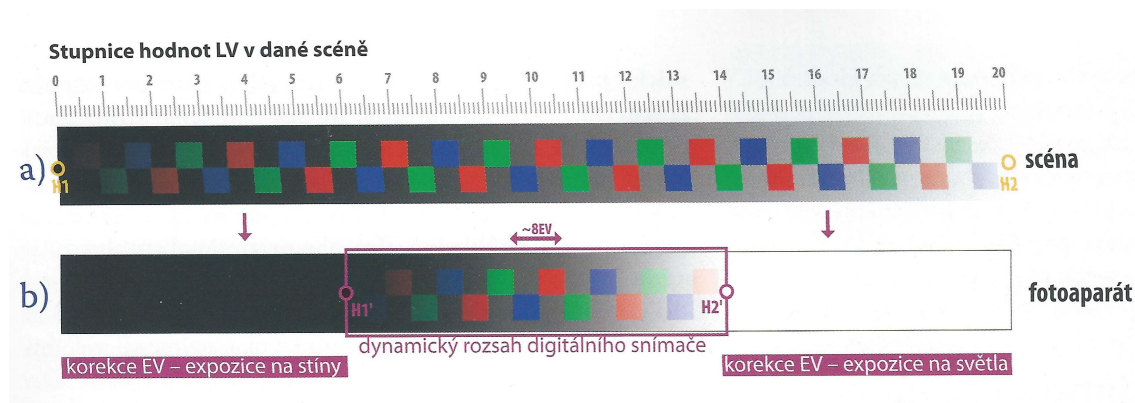
Pojmem dynamický rozsah rozumíme poměr mezi nejtmaším a nejsvětlejším místem na snímku. Potom můžeme rozlišovat mezi nízkým dynamickým rozsahem a vysokým dynamickým rozsahem. Zatímco naše oko vidí obraz s vysokým dynamickým rozsahem, tak většina dnešních fotoaparátů toto zachytit nedokáže. Je to dáno omezenou rozlišovací schopností



pro jeden bod obrazu. Pokud vezmeme v úvahu klasické fotoaparáty, tak většina z nich dokáže zaznamenat obraz na osmi bitech na kanál pro jeden pixel. Z toho plyne, že poměr nejtmašího a nejsvětějšího místa na snímku nemůže být větší než 1:256.

Tento problém si můžeme demonstrovat na situaci, kdy potřebujeme vytvořit fotografii, na které se nachází hodně světlá oblast (například slunce) a hodně tmavá oblast (například oblast ve stínu). Automatika fotoaparátu podle nastavení většinou exponuje na místo kde ostříme. Podle toho je pak buď dobře exponovaná světlá oblast a tmavá je podexponovaná natolik, že oblast splývá s černou barvou a nebo je dobře exponovaná tmavá oblast a světlá je přexponovaná a tudíž vidíme jen bílou barvu.

Fotoaparát si podle svého rozsahu zvolí nejtmaší a nejsvětější místo ve scéně a všechny body, které jsou nad nebo pod těmito hodnotami bude brát jako černé nebo naopak bílé.



Obrázek 2.1: Ukázka toho, jak snímač fotoaparátu vnímá okolní světlo podle svého rozsahu. [2]

### 2.1.3 Struktura fotografie

S fotografií kromě výše uvedených veličin souvisí i to, jak fotoaparát danou fotografii zpracovává. Světlo dopadající na snímač je nutné zpracovat, tak abychom z něj následně mohli vytvořit soubor, který si můžeme prohlížet a ukládat na paměťová média.

#### Formát RAW

Dnes již téměř každý digitální fotoaparát dokáže nějakým způsobem nabídnout ukládání snímku do formátu RAW [11]. Název pochází z anglického slova raw, což znamená nijak nezpracovaný snímek. Obecně tedy platí, že data v tomto formátu jsou minimálně zpracovaná data zachycená snímačem fotoaparátu. Občas je tento formát nazýván digitálním negativem, avšak s klasickým negativem jak ho známe ze starších fotoaparátů nemá mnoho společného. Snad jen to, že se také musí „vyvolat“. Slovem vyvolat se rozumí převedení dat z formátu raw do běžného formátu pro fotografie za pomoci specializovaného softwaru.

Formát RAW není nijak standardizován, a tak dochází k tomu, že výrobci fotoaparátů si vytváří své formáty k uložení dat do souboru. Rozdíly mezi formáty výrobců jsou značné a dokumentace k takovýmto formátům jsou mnohdy nedostupné. A tak dochází k tomu, že jednotlivé formáty dokáže zpracovat pouze program výrobce. O standardizaci se již několikrát pokoušely firmy, jako je například firma Adobe, která definovala svůj otevřený

standardní formát DNG. Tento formát byl následně nabídnut firmám. Bohužel výrobci, kteří ho přijali je jen několik.

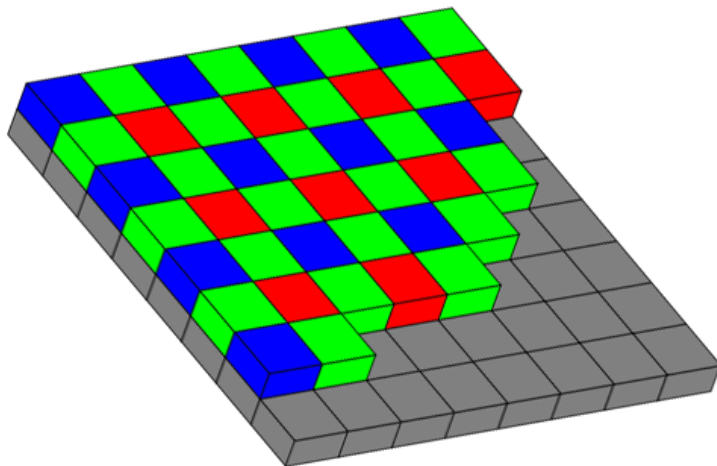
Obsahem RAW souboru jsou pouze digitální data toho co viděl senzor a informace o snímku v podobě metadat. RAW tedy není vyznačen v žádném standardním barevném prostoru, není zde ani vyvážení bílé či použita gama křivka atd.

Světlo dopadající na čip fotoaparátu je převedeno na elektrickou informaci. Z tohoto plyne, že čip je ve své podstatě analogově digitální převodník. Každý takový analogově digitální převodník má svou rozlišovací schopnost. V závislosti na počtu bitů převodníku, je schopen převodník poskytnout různý rozsah výstupních hodnot. V dnešních fotoaparátech jsou běžné možnosti, že dokáží ukládat do RAW formátu snímky s 8, 12 a dokonce i 14 bitovou hloubkou na barevný kanál.

### Bayerova maska

Rozměr snímku je dán počtem pixelů. Z technologického hlediska je obtížné vytvořit senzor tak, aby každý záznamový bod zachycoval barvu. Každá buňka barevného snímače nedokáže zachytit celé barevné spektrum. Buňky jsou tedy barvoslepe a zachycují pouze jas dopadajícího světla. Samotná barva se tomu dodává tím způsobem, že nad každým bodem je barevný filtr určující na jaké vlnové délky bude bod citlivý. V převážné většině se jedná o filtry propouštějící červenou (R), zelenou (G) a modrou (B) barvu.

Tyto barevné filtry jsou uspořádány do pravidelné struktury, kde jsou dvě zelené, jedna červená a jedna modrá barva. Tato struktura je označovaná jako Bayerova maska. Konečnou barvu je potřeba určit za pomoci sousedních pixelů Bayerovou interpolací.



Obrázek 2.2: Bayerova maska nad senzorem s pravidelnou strukturou barevných filtrů. [11]

### Barevné prostory

Světlo jak ho známe je elektromagnetické záření o určité vlnové délce. Podle vlnové délky se pak rozlišuje barva. Lidské oko dokáže zachytit světlo o vlnové délce 390-790nm. Hodnoty nad i pod tímto spektrem nevidíme, protože se jedná o ultrafialové, respektive infračervené světlo. Barevný prostor [4] se potom dá chápat jako množina barev, ve které existuje systém

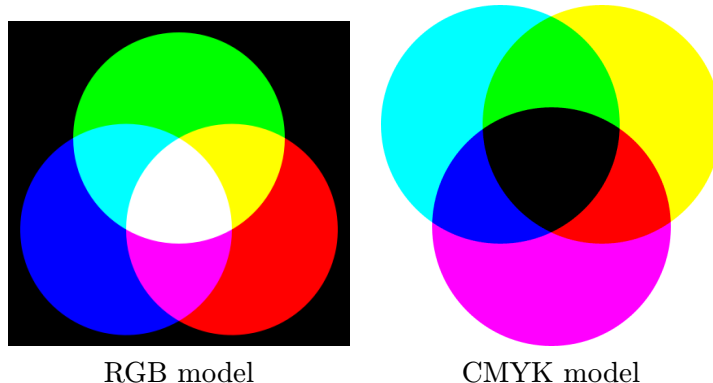
souřadnic, na které je možno se odkazovat pomocí čísel. Je to tedy matematický model vyjadřující všechny barvy, které je možno v daném systému uchovat a odkazovat se na ně.

Jeden z nejnámějších a ve fotoaparátech používaným barevným prostorem je RGB. Jedná se o prostor, který se dá znázornit jako krychle, kde každá osa reprezentuje jednu ze základních barev a to červenou (R), zelenou (G) a modrou (B). Tento prostor je založen na aditivním míchání barev<sup>1</sup>. Znamená to tedy, že při nulové intenzitě barevných složek je výsledná barva černá. Naopak při nejvyšší intenzitě je to barva bílá. S tímto barevným prostorem se setkáváme nejčastěji, a to převážně v displejích, kde každý pixel je reprezentován právě trojicí subpixelů. RGB prostor lze převést na obraz ve stupních šedi. Tento převod je dán následující rovnicí:

$$L = 0.2126R + 0.7152G + 0.0722B \quad (2.4)$$

Tato rovnice je dána vnímáním lidského oka. Oko je nejvíce citlivé na zelenou barvu, a proto je ve výsledné hodnotě nejvíce zastoupená a naopak nejméně citlivé na modrou barvu.

Opakem RGB prostoru je model CMYK. Ten nalezneme v barevných tiskárnách, kde přidávání intenzity barev znamená přibližování se k černé. Je tvořen čtveřicí barev a to azurovou (C), purpurovou (M), žlutou (Y) a černou (K).



Obrázek 2.3: RGB a CMYK model reprezentován třemi kružnicemi základních barev. [4]

Dalším používaným barevným prostorem v informatice je prostor CIE XYZ. Tento prostor byl vytvořen v roce 1931, a proto je někdy označován jako CIE 1931 XYZ a je základem pro všechny ostatní barevné prostory. Je založen na odděleném vnímání jasu a barvy. Hodnota Y vyjadřuje jas a zbylé dvě X a Z vyjadřují barvu.

Prostory RGB a CIE XYZ jsou navzájem převoditelné. Převod z RGB na CIE XYZ je dán následující rovnicí:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.5)$$

a převod z CIE XYZ na RGB je následovný:

<sup>1</sup>Míchání barev založené na intenzitě světla dané barevné složky.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.2405 & -1.5371 & -0.4985 \\ -0.9693 & 1.8760 & 0.0416 \\ 0.0556 & -0.2040 & 1.0572 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (2.6)$$

## 2.2 Pořízení HDR snímku

Vytvořit fotografii s vysokým dynamickým rozsahem přímo vyfotografováním scény není v dnešní době obvyklé. Je zapotřebí speciálního hardwaru, který je finančně náročný. Proto se pokoušíme nalézt způsoby jak HDR fotografii vytvořit i jiným způsobem. V následujícím textu je popsáno několik metod jak takovou fotografii vytvořit alternativním způsobem.

### 2.2.1 Získání HDR obrazu z více snímků

Jelikož dnešní běžný hardware neumožňuje expozičně pokrytí celé scény, tak je možnost poskládat obraz s vysokým dynamickým rozsahem z více snímků. Tyto snímky mají různou expozici odstupňovanou po několika EV. Obvykle je nutno vytvořit 3 a více snímků. Na každém takovém snímku vynikne určitá oblast. Na snímcích s nižším expozičním časem vyniknou oblasti, které mají vyšší intenzitu světla, jako je například slunce nebo pouliční lampa a naopak na snímku s vysokým expozičním časem vyniknou oblasti tmavé nebo třeba oblasti ve stínu.



Obrázek 2.4: Snímky s různou expozicí a výsledný obrázek, jenž je namapovaný lokálním tónovým mapováním. [6]

Pro získání více snímků stejné scény je potřeba zajistit, aby daná scéna byla statická. Jedním z faktorů možného pohybu je pohyb samotného fotoaparátu. Proto je nezbytné použití kvalitního stativu. Lepších výsledků lze docílit zatížením stativu závažím. Jelikož je nutné vytvořit více snímků a měnit expozici, nastává i problém kdy se fotoaparátu dotýkáme a tím s ním můžeme nechtěně pohnout. Pro tento případ lze využít režim fotoaparátu zvaný automatický expoziční bracketing (AEB) [2]. V tomto režimu můžeme nastavit kolik snímků chceme vytvořit a jaký mají mít mezi sebou expoziční rozestup. Automatika fotoaparátu již vše vytvoří za nás. Navíc časový rozestup mezi snímky není tak velký jako kdybychom to dělali ručně. Pokud fotoaparát tento režim nemá, lze použít dálkovou spoušť.

Dalším negativním faktorem může být pohyb objektů ve scéně. Může se jednat o pohyb objektu zapříčiněný větrem a nebo pohyb lidí, či jiným pohybujícím se objektem. Proto je dobré vyhnout se takovýmto scénám a fotit statické děje neboť při pohybu objektu mezi snímky se mohou ve výsledném HDR snímku objevovat takzvaní „duchové“.

Expozici můžeme měnit pomocí již zmiňovaných tří veličin. Clony, expozičního času a citlivosti ISO. Různou kombinací těchto hodnot můžeme dosáhnout stejného výsledku. Není však žádoucí, aby se scéna mezi snímky měnila. Proto právě není dobré měnit clonu a citlivost ISO. Změnou clony totiž měníme i hloubku ostroty, a tak by se mohlo stát, že každý snímek bude mít zaostřenou jinou oblast. Zvyšování citlivosti ISO se zase projevuje viditelnějším šumem. Proto je jediným řešením tyto dvě hodnoty nechat pevně nastavené a měnit pouze expoziční čas. Doporučeno je také nechat fotoaparát nastaven na manuální režim ostření.

Většina dnešních moderních zrcadlovek má možnost výstupu ve formě 12 bitového RAW. Hodnoty ze snímače tedy můžeme brát jako lineární. To znamená, že zdvojnásobení množství světla dopadajícího na snímač má za následek zdvojnásobení hodnoty výstupu ze snímače. Abychom mohli složit HDR obraz musíme znát expoziční čas, hodnotu pixelu a váhovou funkci, která určuje váhu pixelu z jednotlivých snímků. Výsledný pixel je dán následující rovnicí [12]

$$L_{ij} = \sum_{k=1}^N \frac{Z_{ij} * w(Z_{ij})}{\Delta t_k} \Bigg/ \sum_{k=1}^N w(Z_{ij}), \quad (2.7)$$

kde  $k$  je počet snímků,  $Z_{ij}$  je hodnota LDR snímku,  $\Delta t$  je expoziční čas a  $w(Z_{ij})$  je váhová funkce 2.2.1. Pro váhovou funkci můžeme použít tento vzorec

$$w(z) = \begin{cases} z - Z_{min} & \text{pro } z \leq \frac{1}{2}(Z_{min} + Z_{max}) \\ Z_{max} - z & \text{pro } z \geq \frac{1}{2}(Z_{min} + Z_{max}) \end{cases}, \quad (2.8)$$

který dává větší váhu pixelům blízkým se k průměru než pixelům podexponovaným nebo přexponovaným.

Pokud bychom však chtěli pracovat se snímky klasického formátu, musíme do výpočtu zahrnout funkci odezvy snímače fotoaparátu. Vzorec by potom vypadal následovně.

$$L_{ij} = \sum_{k=1}^N \frac{f^{-1}(Z_{ij}) * w(Z_{ij})}{\Delta t_k} \Bigg/ \sum_{k=1}^N w(Z_{ij}) \quad (2.9)$$

Jedinou změnou v rovnici je přepočtení hodnoty pixelu inverzní funkcí k funkci odezvy snímače. Funkci odezvy snímače můžeme zjistit buďto přímo od výrobce nebo pomocí metody, kterou představili pánové Debevec a Malik [5]. Tato metoda dokáže zjistit tuto funkci ze sekvence snímků s různou expozicí. V této práci budeme však používat snímky s lineární

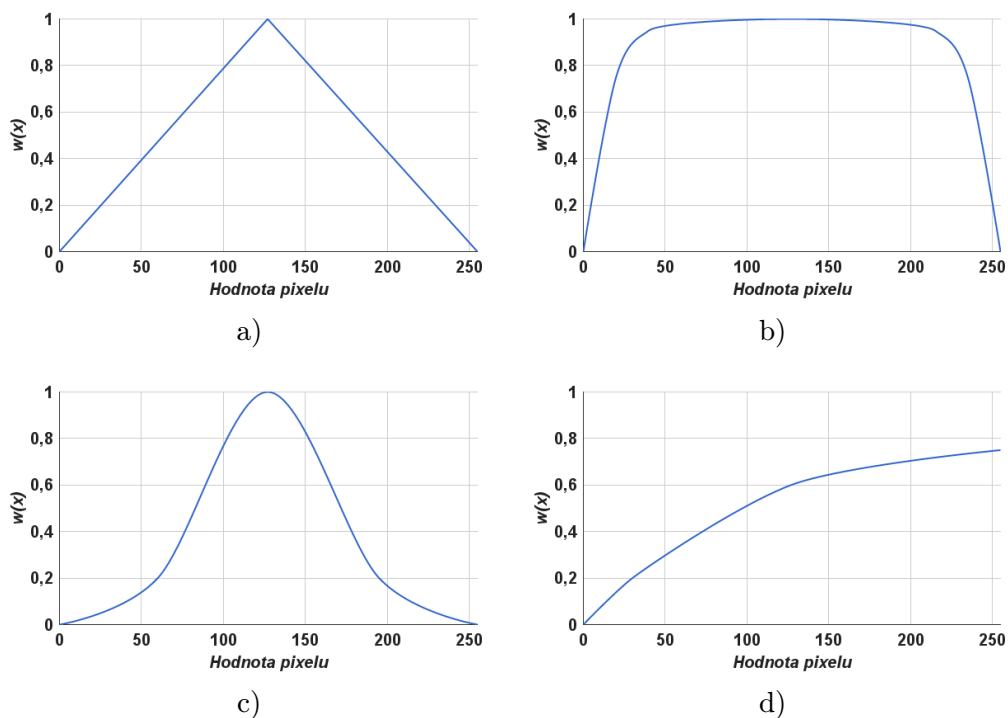


odezvou a tudíž tuto funkci v této práci zjišťovat nebudeme. Knihovna, kterou budu vytvářet bude ale mít předpřipravenou část pro případné rozšíření této práce.

## Váhová funkce

Z předchozí rovnice 2.7 je patrné, že výsledný snímek ovlivňují dvě veličiny. Jednou z nich jsou samotné fotografie, ze kterých se skládá HDR obraz a druhou je váhová funkce. Vstupní fotografie jsou do vzorce pevně dány a při skládání je jen na nás zda poskytneme správné sekvence fotografií s různou expozicí. Co však můžeme ovlivnit při vytváření algoritmu, který bude fotografie skládat je právě váhová funkce. Váhová funkce nám určuje jak moc uplatní hodnotu pixelu ze vstupní fotografie ve výsledném obraze. Účelem váhové funkce je dát přednost správně exponovanému pixelu před pixelem, který je podexponovaný nebo přexponovaný. Tímto můžeme například eliminovat šum fotografie. Jako další věc, kterou nám váhová funkce dovoluje, je také kompletně ignorovat některé části histogramu<sup>2</sup> fotografie tím, že v těchto oblastech dáme váhové funkci nízké hodnoty.

Doporučených váhových funkcí pro skládání HDR obrazu je hned několik. Můžeme si vytvořit vlastní funkce a podle výsledku se rozhodovat, která poskytuje kvalitní výsledky pro určité sekvence. Ve své podstatě je však jen na nás jakou použijeme, a která nám přijde vhodná.



Obrázek 2.5: Ukázka jednotlivých váhových funkcí. a) Funkce Debevec-Malik[1] b) Hat funkce ořezávající pouze extrém v expozici[10] c) Gaussova funkce[10] d) Funkce Mitsunaga & Nayar[7]

<sup>2</sup>Histogram nám graficky znázorňuje počet pixelů, které náleží konkrétnímu jasu v rozsahu hodnot fotografie.

Grafy 5.1 znázorňují jak taková váhová funkce vůbec vypadá. Každá z těchto funkcí má svá specifika ve výsledném HDR. Jedna z neznámějších funkcí je funkce Debevec Malik. Tato funkce je nastavena tak, že dává největší váhu pixelům ve středu expozice a lineárně tato váha klesá s pixely podexponovanými nebo přeexponovanými. Podobná k této funkci je klasická Gaussova křivka. Ta však klade ještě větší váhu na pixely ve středu expozice. Další váhovou funkci představili pánové Mitsunaga a Nayar. Jejich funkce dává malou váhu podexponovaným pixelům a s rostoucí hodnotou pixelu stoupá i váha. Je to opodstatněno tím, že hodnoty jsou sice přeexponované, ale zato se u nich nevyskytuje takový šum jako u hodnot podexponovaných.

### 2.2.2 Skenování negativu

Tato technika je v dnešní době poněkud neaktuální, jelikož se jedná o skenování negativu ze starších fotoaparátů. Ideální případ je takový, že focená scéna je statická. V tomto případě předpokládáme, že funkce odezvy fotoaparátu je nám známa. Navíc musíme taky předpokládat, že zařízení na pořízení LDR snímku (například 8-bitový skener negativu se známou funkcí odezvy snímače) umožňuje ovládat přesně expozici snímku.

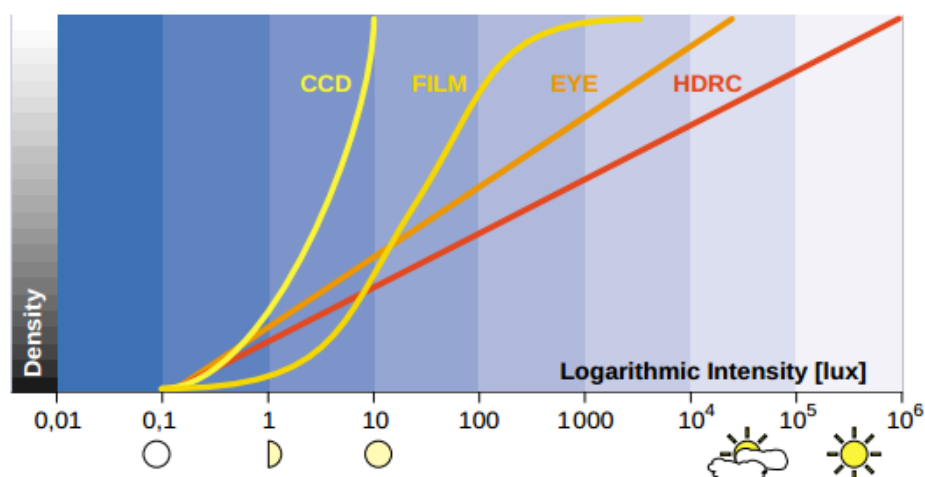
Celý proces začíná oskenováním negativu skenerem. Následně je potřeba získané hodnoty linearizovat. To se provede za pomoci inverzní funkce k funkci odezvy snímače. Každý oskenovaný snímek je vynásoben kalibračním faktorem závislým na jeho expozici a zkombinován do výsledného HDR. Jedinou otázkou zůstává jakou váhovou funkci pro skládání zvolit. Samozřejmě nechceme, aby se velkou váhou na výsledném HDR podílely nízké nebo naopak vysoké hodnoty pixelu. Nabízí se tedy využít váhovou funkci Debevec Malik, která těmto hodnotám nedává příliš velkou váhu. Další variantou je použít váhovou funkci Mitsunaga a Nayar. Ta sice dává velkou váhu i pixelům s vyšší hodnotou, avšak tyto pixely jsou méně ovlivněné šumem.

### 2.2.3 Specializovaný hardware

Jak již bylo řečeno v kapitole 2.2.1, tak složení HDR obrazu z více snímků je vhodné pro statické scény. V ostatních případech je obtížné napojení více snímků na sebe, tak aby se ve výsledném snímku neobjevovaly mlhavé čmouhy. Proto je vhodné použít specializovaný hardware, kterým dokážeme scénu zachytit na jeden snímek.

Obecně můžeme říct, že u fotoaparátu, který by měl snímat HDR obraz, potřebujeme zvýšit bitovou hloubku, což má za následek rozlišit více hodnot daného pixelu. Pouze toto ale nestačí. Snímače fotoaparátů jsou technologicky omezené a citlivost na intenzitu osvětlení bývá nevhodná.

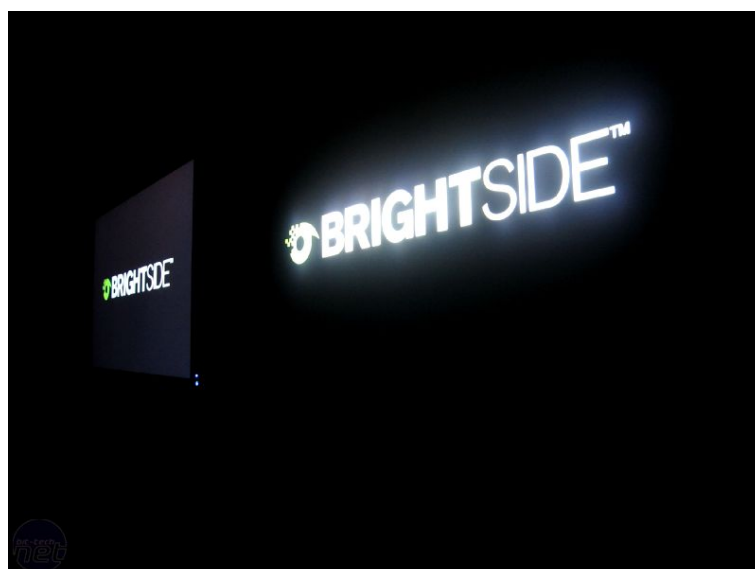
Lidské oko vidí scénu s vyšším dynamickým rozsahem než dnešní fotoaparáty. Je to dáno také tím, že lidské oko má logaritmickou citlivost na okolní světlo, zatímco běžné snímače jako například CCD mají lineární citlivost. Tuto problematiku se snaží řešit snímače zvané HDRC[9]. Ty by se svou charakteristikou měly blížit k logaritmické citlivosti lidského oka a poskytnout tak podobné výsledky jako naše oko.



Obrázek 2.6: Graf znázorňující citlivost snímačů na dopadající světlo. [9]

### 2.3 Zobrazení HDR snímku

Nynější pokrok techniky bohužel trochu opomíjí displeje. Trendem je spíše zvyšování rozlišení, ale co se týče vyššího dynamického rozsahu, tak to je zanedbáváno. Dnešním nejrozšířenějším typem displeje je LCD. Průchod světla u tohoto typu displeje je řízen tekutými krystaly, které se v závislosti na napětí otáčejí a tím propouštějí světlo. Problémem není nedostatečně vysoký svit pixelu. Problém je právě opačný. Úplné zhasnutí displeje skoro není možné. Je to zapříčiněno globálním podsvícením a nedokonalostí tekutých krystalů zcela zabránit pronikajícímu světlu projít skrz.



Obrázek 2.7: Porovnání klasického a HDR displeje. [14]



Možným řešením je ovládní podsvícení každého pixelu zvlášť diodami. Tak by šlo diodu úplně vypnout a tím získat zcela černou barvu. Na takovém principu fungují OLED a AMOLED displeje. Tato technologie se zatím objevuje pouze na mobilních zařízeních nikoli na velkých displejích nikoli. Výroba takovýchto velkých panelů s technologií podsvícení každého obrazového bodu zvlášť je totiž velice nákladná a zatím se výrobcům nevyplatí.

Známým výrobcem HDR displejů je například společnost Sim2. Na trh kupříkladu uvedla displej HDR47E [15], který uvádí kontrast větší jak 1 000 000 : 1. Toto odpovídá rozsahu 19.9EV.

## 2.4 Formáty ukládání

Jelikož výstup ze skládání HDR obrazu z více snímků převyšuje původních většinou osm bitů, je potřeba použít speciální formáty [8], které si s takovou bitovou hloubkou poradí. Klasické formáty jako JPGE nebo PNG nemůžeme použít, protože jsou navrženy na právě zmiňovaných 8 bitů na kanál.

Nekomprimovaný HDR pixel je reprezentován třemi čísly s plovoucí řádovou čárkou. To znamená, že je pro jeden pixel potřeba 12 bajtů v paměti. Při snímku s Full HD rozlišením, tedy 1920 x 1080, by snímek zabíral přibližně 24MB. Toto je mnohem více než 6MB, což je potřeba k uložení klasického LDR snímku bez komprese. Proto se někteří výzkumníci snažili přijít na způsob, jak efektivně uložit HDR obraz tak, aby neměl takové paměťové nároky. Některé formáty pro uložení HDR snímku si zde představíme.

### LogLuv

Jedna z možných variant je LogLuv formát. Tento formát přiřazuje více bitů jasu v logaritmické oblasti, než barvám v lineární oblasti. Na začátku je obraz konvertován do Luv barevného prostoru. Formát obsahující 32 bitů na pixel přiděluje 15 bitů jasu a 16 bitů barevnosti a je definován jako

$$L_e = [256(\log_2 Y_W + 64)], \quad u_e = [410u'], \quad v_e = [410v'] \quad (2.10)$$

a dekódován jako

$$Y = 2^{(L_e+5.5)/256-64}, \quad u' = \left[ \frac{1}{410}(u_e) \right], \quad v' = \left[ \frac{1}{410}(v_e) \right]. \quad (2.11)$$

Pro formáty pracující s jiným počtem bitů na pixel jsou konstanty v rovnici 2.10 a 2.11 trochu odlišné. Hlavní výhodou LogLuv formátu je, že formát ukládá jas a informace o barvě odděleně. Tím, že jsou tyto složky odděleny, ulehčuje práci následnému zpracování jako například tónovému mapování nebo manipulaci s barvami.

### RGBE

Tento formát byl původně představen v roce 1989 jako součást renderovacího systému. Někdy je také nazýván jako Radiance HDR (.hdr, .pic). Od té doby našel široké uplatnění v grafické komunitě. Soubor obsahuje krátkou ASCII hlavičku následovanou řetězcem označující velikost obrazu a orientaci. A nakonec samotné hodnoty pixelů. Jeden pixel je zakódován jako čtveřice bajtů. První tři nesou hodnoty barevných složek (RGB) a čtvrtý bajt označuje společný exponent všech tři barevných složek. Dohromady je tedy pro jeden pixel určeno 32 bitů, což se od klasického RGB liší i pouhý jeden bajt.

Hodnoty barevných složek a exponentu se počítají následovně

$$E = \lceil \log(\max(R_W, G_W, B_W)) + 128 \rceil \quad (2.12)$$

$$R_M = \left\lceil \frac{256 * R_W}{2^{E-128}} \right\rceil \quad (2.13)$$

$$G_M = \left\lceil \frac{256 * G_W}{2^{E-128}} \right\rceil \quad (2.14)$$

$$B_M = \left\lceil \frac{256 * B_W}{2^{E-128}} \right\rceil \quad (2.15)$$

Pro zpětný převod z formátu RGBE na tři barevné složky se používají následující rovnice

$$R_W = \frac{R_M + 0.5}{256} 2^{E-128} \quad (2.16)$$

$$G_W = \frac{G_M + 0.5}{256} 2^{E-128} \quad (2.17)$$

$$B_W = \frac{B_M + 0.5}{256} 2^{E-128} \quad (2.18)$$

## 2.5 Tonové mapování

Naprostá většina dnešních zobrazovacích zařízení není schopná zobrazit obraz s vysokým dynamickým rozsahem. Kvůli této překážce musíme HDR obraz upravit tak, abychom jej mohli zobrazit na klasických monitorech. Tato technika se nazývá tónové mapování. V principu jde o to, že se rozsah HDR obrazu musí přemapovat do rozsahu 8 bitů, jenž je dnešní technika schopná zobrazit. Na první pohled by se mohlo zdát jako nesmyslné nejdříve přepočítávat obraz na obraz s vysokým dynamickým rozsahem a poté jej zpátky transformovat na LDR obraz. Musíme tedy vybírat takový algoritmus, který respektuje detaily získané při vytváření HDR obrazu.

Tonové mapování můžeme rozdělit do dvou základních kategorií.

### 2.5.1 Globální operátory

Jedná se o techniku při níž je brán zřetel pouze na pixel, se kterým se zrovna pracuje. Jedná se o jednodušší techniku, a proto není tak časově náročná. Tento algoritmus nebere zřetel na okolí pixelu a tudíž se může stát, že obraz přijde o část detailu. V následující části si představíme jak základní [1], tak i složitější [13] globální operátory.

- **Lineární operátor** - jedná se o nejjednodušší techniku jak přemapovat vysoký rozsah do nízkého. Podle maximální hodnoty obrazu se určí koeficient  $e$ .

$$e = \frac{1}{L_{w,MAX}} \quad (2.19)$$

Tímto koeficientem poté vynásobíme všechny pixely v obraze.

$$L_d(x) = e * L_w(x) \quad (2.20)$$

Tímto sice získáme zpět LDR obraz, ale výsledek nebude nijak kvalitní. Proto je tento operátor spíše ukázkový.

- **Logaritmické mapování** - na logaritmické mapování se používá logaritmická funkce HDR hodnot.

Pro výpočet se používá následující vzorec:

$$L_d(x) = \frac{\log_{10}(1 + qL_w(x))}{\log_{10}(1 + kL_{w,MAX})} \quad (2.21)$$

kde  $q \in [1, \infty)$  a  $k \in [1, \infty)$  jsou konstanty voleny uživatelem pro zvolení požadovaného vzhledu obrazu.

- **Exponenciální mapování** - pro toto mapování se používá exponenciální funkce HDR hodnot. Přemapovává HDR hodnoty do intervalu  $[0, 1]$ , ze kterého je dokážeme převést do požadovaného intervalu.

Operátor je definován jako

$$L_d(x) = 1 - \exp\left(-\frac{qL_w(x)}{kL_{w,H}}\right) \quad (2.22)$$

kde  $q \in [1, \infty)$  a  $k \in [1, \infty)$  jsou konstanty voleny uživatelem.

- **Reinhardův operátor** - algoritmus tohoto operátoru je založen na hodnotě jasu celé fotografie. Vyjadřuje ji hodnota nazývaná klíč scény, jejíž hodnota udává o jak jasnou scénu se jedná. Ve většině případů je tato hodnota spojována s logaritmickým průměrem všech obrazových bodů. Tento průměr a zároveň tedy klíč scény je možno vypočítat jako:

$$\bar{L}_w = \exp\left(\frac{1}{N} \sum_{x,y} \log(\delta + L_w(x, y))\right) \quad (2.23)$$

kde  $L_w(x, y)$  určuje jas konkrétního obrazového bodu  $(x, y)$  reálné scény,  $N$  je celkový počet obrazových bodů a  $\delta$  je určená malá hodnota, proto aby nedocházelo k nedefinovaným výpočtům logaritmu v případě, že by se jednalo o úplně černý pixel s hodnotou nula. Pokud by scéna měla normální hodnotu klíče chtěli bychom obraz namapovat do středně šedé hodnoty zobrazené fotografie a nebo na hodnotu 0.18 pokud bychom chtěli mapovat do intervalu nula až jedna. K tomuto je potřeba využít následující rovnice

$$L(x, y) = \frac{a}{\bar{L}_w} L_w(x, y) \quad (2.24)$$

kde  $L(x, y)$  představuje zmenšený rozsah jasu a hodnota  $a$  určuje na jaký rozsah bude jas namapován. Jako doporučená hodnota se uvádí  $a = 0.18$ . Typicky se ale může lišit podle jasu dané scény. Uvádějí se hodnoty od zmiňovaných 0.18 přes 0.36 až po 0.72.

Hlavní problém rovnice 2.24 je v tom, že většina focených scén má v převážné většině normální dynamický rozsah, ale objevují se taky části snímků s vysokým jasem, které z fotografie vyčnívají. Tento problém řeší následující rovnice:

$$L_d(x, y) = \frac{L(x, y)}{1 + L(x, y)} \quad (2.25)$$

kterou lze dále upravit na rovnici následující

$$L_d(x, y) = \frac{L(x, y) \left( 1 + \frac{L(x, y)}{L_{white}^2} \right)}{1 + L(x, y)} \quad (2.26)$$

kde  $L_{white}$  představuje hodnotu jasu, která bude namapována do čistě bílé barvy. Pokud tedy za tuto hodnotu dosadíme maximální jas celé fotografie, tak nebude docházet k žádným přexponovaným místům na výsledném snímku. Pokud by však tato hodnota byla nastavena na hodnotu nekonečna, tak by se výsledek rovnice shodoval s výsledkem rovnice 2.25.

### 2.5.2 Lokální operátory

Lokální operátory vylepšují kvalitu tónového mapování obrazu skrz globální operátory pokusem o reprodukci jak lokálního, tak globálního kontrastu. V praxi to znamená, že na pixel ve světlé oblasti je nahlíženo jinak než na pixel v tmavé oblasti. Toho je dosaženo mapovacím operátorem, který bere v potaz intenzitu okolních pixelů právě upravovaného bodu. Nicméně sousední pixely je nutno vybírat opatrně. Při špatném výběru se může projevit halo efekt na hranách. Halo efekt je ale někdy žádoucí, pokud chceme zdůraznit nějakou část obrazu.

## Kapitola 3

# Specifikace zadání

V předchozí kapitole byl představen jeden z problémů týkající se digitální fotografie. Na základě informací o tomto problému bude specifikován cíl práce a metody použité k odstranění problému nízkého dynamického rozsahu.

Cílem práce je tedy navrhnout a následně vytvořit knihovnu, která bude schopna složit snímek s vysokým dynamickým rozsahem a v druhém kroku na tento snímek použít operátory tónového mapování, které umožní zobrazení snímku na běžném displeji. Knihovna tedy bude řešit problematiku nízkého dynamického rozsahu a to tak, že bude využívat metodu skládání sekvence snímků s různou expozicí. Pro tuto metodu je potřeba tři a více snímků s různou expozicí. V této práci je stanoveno minimum tří snímků z důvodu získání konkrétních detailů z každého z nich. Prvním z nich je snímek podexponovaný, kde vyniknou velmi světlé oblasti, dále snímek s expozicí zaměřenou do středu světlosti scény, který obsáhne většinu detailu s výjimkou extrémů. Poslední je snímek přexponovaný, kde vyniknou oblasti s nízkou hodnotou jasu. Touto trojicí snímků jsme schopni pokrýt většinu světelných podmínek scény. Horní hranice počtu snímků však omezena není a bude možno nahrát sérii snímků s drobnějšími rozdíly mezi expozicemi. Je nutné si ale uvědomit, že se zvyšováním počtu snímků roste i čas zpracování a výsledky nejsou o tolik lepší, než při použití tří snímků.

Důležitou součástí algoritmu skládání je také zvolení správné váhové funkce. V této knihovně bude použita váhová funkce Debevec & Malik a to z toho důvodu, že je to jedna z neznámějších váhových funkcí v problematice HDR fotografií a je popsána ve většině publikací zabývajících se tímto tématem. Dalším důvodem pro zvolení této funkce je i to, že mi připadá velice rozumné použít právě tuto funkci, protože se zaměřuje hlavně na hodnoty snímku ve středu expozice, které nesou nejvíce obrazových informací a naopak nepřikládá velkou váhu hodnotám přexponovaným či podexponovaným.

Po provedení algoritmu skládání jsou dvě možnosti co se se snímkem dá dále provádět. První z nich bude možnost uložení takto vytvořeného snímku do souboru s patřičným formátem. Pro tyto účely byl zvolen formát HDRE z důvodu jeho popularity a zároveň také proto, že je k jeho uložení potřeba pouze čtyř bajtů pro jeden obrazový bod. Pro tento formát byla použita knihovna, která je detailněji popsána v kapitole 4.5. Druhou možností zpracování HDR snímku bude použití metod tónovacího mapování. Při použití této metody bude mít uživatel možnost nechat si výsledný snímek zobrazit na běžném displeji. V této práci jsem se rozhodl použít globální tónovací operátory, jelikož dokáží poskytnout dobré výsledky v relativně krátkém čase. Konkrétně se jedná o operátor lineární, logaritmický, exponenciální a Reinhardův. První z nich je operátor spíše ukázkový, od kterého se neočekávají příliš kvalitní výsledky. Důvodem jeho zařazení do knihovny je snaha poukázat na

rozdíl ve výsledcích jednoho z nejjednodušších operátorů a těch ostatních. Dalšími dvěma operátory jsou logaritmický a exponenciální. Oba jsou založeny na elementárních matematických funkcích a jejich kvalita by měla být o úroveň vyšší než u předchozího. Rozhodl jsem se pro ukázkou zobrazení lepších výsledků zařadit mezi nabízené operátory i Reinhardův globální operátor. Tento operátor by měl dosahovat velice kvalitních výsledků. Operátor byl vybrán na základě webové stránky [16], kde jsou srovnány výsledky několika operátorů, přičemž právě Reinhardův operátor se řadil mezi ty nejlepší v kategorii globálních. Výsledný snímek po aplikaci tónových operátorů bude možno vyexportovat do známých a často používaných formátů.

Další součástí výsledné aplikace bude propojení kamery s ukázkovou aplikací skrze síťové rozhraní. Kamera bude komunikovat s aplikací a bude poskytovat sekvence snímků s určitou expozicí, které jí jsou zasílány v požadavcích z aplikace. Aplikace bude obsahovat i grafické uživatelské rozhraní, které bude demonstrovat vybrané funkce z knihovny a tím zjednodušovat práci s knihovnou.

## Implementace

Výsledkem této práce bude tedy knihovna, kterou mohou ostatní programátoři přidat do svého projektu a použít ji k vytváření HDR fotografií. Jako implementační jazyk byl vybrán jazyk C++ kvůli tomu jak je rozšířený a existuje pro něj spousta knihoven, které dokáží usnadnit práci. Aplikace bude postavena na frameworku QT verze 5.4 a bude vyvíjena ve vývojovém prostředí QT Creator 3.3.2 v operačním systému Ubuntu linux ve verzi 14.04 Trusty Tahr. Tento framework a s ním spojené vývojové prostředí bylo vybráno z důvodu lepší práce s projektem a možností vytváření grafického uživatelského rozhraní.

V tomto programu bude využívána jedna z neznámějších knihoven pro práci s fotografií a videem OpenCV 2.4.1. Tato knihovna nabízí velké množství funkcí pro práci s grafikou a soustředí se na obor počítačového vidění. Aplikace však bude využívat jen zlomek její funkcionality. Jedná se především o možnosti ukládání fotografií do datové struktury matic a přístupu k jednotlivým bodům obrazu. S tímto jsou spojeny také funkce pro načítání již hotových fotografií do těchto struktur a následné uložení do souboru ve známých formátech.

V aplikaci budou využity i další knihovny, které řeší části, jež nesouvisí úplně s tematikou dynamického rozsahu. Jedná se o knihovnu Exiv 2.2 pracující s daty, které nesou informace o tom s jakými parametry fotografie vznikla. Další využitou knihovnou bude knihovna pro komunikaci se síťovou kamerou, jež mi byla poskytnuta spolu s kamerou.

## Kapitola 4

# Návrh a implementace

### 4.1 Návrh a implementace knihovny

Cílem práce je vytvoření knihovny a tudíž je důležité udělat správný návrh této knihovny, tak aby byla co nejjednodušší na použití v jiných projektech a zároveň, aby měla všechny požadované vlastnosti spojené s problematikou skládání obrazu s vysokým dynamickým rozsahem. Dalším důležitým kritériem je navrhnout knihovnu tak, aby se dala jednoduše rozšiřovat pro další operátory tónového mapování nebo třeba použití jiných váhových funkcí, či použití inverzní funkce odezvy snímače pro fotografie v jiném formátu než RAW, které nejsou lineární.

Knihovna je tedy rozčleněna do logických celků, které buďto uchovávají informace o snímku a nebo provádějí činnosti spojené s procesem tvorby HDR. Jelikož je pro tvorbu knihovny použit jazyk C++, který je objektově orientovaný, tak je ve většině případů využita možnost dědičnosti tříd, kdy specifické třídy dědí základní vlastnosti od svých předků.

#### Třída `LDRImage`

Tato třída tvoří základní strukturu pro uchování klasického snímku, tedy snímku s nízkým dynamickým rozsahem. Kromě samotných dat fotografie uchovává informace o expozici snímku ve formě expozičního času. Základem této třídy je objekt `cv::Mat`, ve kterém jsou uchovány informace o samotných obrazových datech. Jelikož se jedná o snímek s nízkým dynamickým rozsahem, tak struktura této matice je tvořena obrazovými body, které mohou nabývat hodnot 0 až 255 a tudíž k uchování těchto hodnot postačuje datový typ `unsigned char`. Každý z obrazových bodů je tvořen trojicí barevných kanálů RGB. Pro přístup k jednotlivým bodům je připravena funkce `LDRImage::getPixel(int x, int y)`, která jako svůj výsledek vrátí strukturu `Vec3b`, jenž tvoří pole s hodnotami pro barevné kanály.

Aby bylo možné fotografii zobrazit za pomoci QT grafického uživatelského rozhraní, obsahuje tato třída i datovou konverzi mezi OpenCV strukturou `cv::Mat` a strukturou `QImage`. Ta je součástí QT knihovny a uchovává obrazové informace o fotografii a lze snadno zobrazit v QT grafickém elementu `QLabel`. Pro převod mezi těmito datovými strukturami, je nejprve nutné převést barevný model OpenCV matice. OpenCV si totiž v základním nastavení ukládá matice ve formátu BRG, což znamená, že je nejdříve uložena modrá, poté červená a nakonec zelená barva, zatímco `QImage` má tuto strukturu klasickou RGB. Pro tento převod je tedy použita funkce `cv::cvtColor()` s parametrem `cv::CV_BRG2RGB` a z takto získaných dat je vytvořen objekt `QImage`.

## Třída HDRImage

Podobně jako předchozí třída i tato slouží k uchování obrazových dat. Rozdílem však je, že dokáže uchovat data s vysokým dynamickým rozsahem. Základem je opět matice `cv::Mat`, ale tentokrát s datovým typem `double`, který je schopen uchovat hodnoty s vyšším rozsahem. Stejně tak jako v předchozím případě je pro přístup k obrazovým bodům připravena funkce `HDRImage::getPixel(int x, int y)`, která ale vrací strukturu `Vec3f`, jenž je tvořena polem barevných kanálů s datovým typem `double`.

Tato třída také umožňuje uložit nebo načíst svůj obsah ze souboru ve formátu HDRE. K tomuto je určena funkce `HDRImage::saveHDR()`.

## Třída WeightFunciton

Tato třída reprezentuje vlastnosti váhové funkce. Jelikož typů váhových funkcí může být více, tak se jedná pouze o třídu, která obsahuje všechny společné vlastnosti váhových funkcí a konkrétní váhové funkce od ní dědí tyto vlastnosti. V knihovně je takto řešena konkrétní váhová funkce Debevec Malik pod názvem `DebevecMalikWF`. Všechny hodnoty váhové funkce jsou vypočítány předem a uloženy do statického 256 položkového pole. Pokud pak přijde požadavek na získání váhy pro určitý pixel, je tato hodnota vrácena z předem vypočítaného pole. Důvodem k takovému typu řešení je hlavně rychlost, za kterou je hodnota váhové funkce získána. Nemusí se tedy hodnota váhové funkce počítat pokaždé zvlášť, a tudíž i odpadá problém počítání stejných hodnot opakovaně. Požadavky na hodnoty váhové funkce jsou totiž velice časté a to konkrétně pro každý barevný kanál všech obrazových bodů fotografie.

## Třída HDRCreator

Třída `HDRCreator` zaštiťuje proces vytváření snímků s vysokým dynamickým rozsahem. Hlavním vstupem pro tento proces je seznam minimálně tří fotografií s různou expozicí. Každá z těchto fotografií je uložena již ve zmiňovaném objektu `LDRImage`. Dalším důležitým vstupním objektem pro tuto třídu je i zvolena váhová funkce. Rozhraní této třídy umožňuje i použití inverzní funkce odezvy snímače, ale jelikož s ní v této práci nepočítáme, tak je jako výchozí převod použit poměr jedna k jedné, což znamená že hodnoty vstupních snímků jsou brány přímo bez jakékoli úpravy.

Nejdůležitější metodou této třídy je metoda `HDRCreator::buildHDR()`, která jako první zkontroluje zda vstupní fotografie jsou rozměrově stejně velké a mají stejný počet barevných kanálů. Poté již započne samotný proces skládání HDR obrazu. Jelikož proces skládání HDR obrazu nijak nezkoumá okolní body, tak se v jednom momentě zpracovává pouze jeden pixel. Tento proces je také mnohdy nazýván „per pixel“. Výsledkem tohoto procesu je obraz s vysokým dynamickým rozsahem reprezentován maticí typu `cv::Mat`, kde každý obrazový bod je tvořen strukturou typu `cv::CV_32FC3`. Tato matice je poté zaobalena do specifitějšího objektu `HDRImage`, který je výsledkem této funkce.

## Třída ToneMapping

Touto třídou je reprezentován závěrečný krok celého procesu vytváření HDR obrazu a to konkrétně proces tónového mapování. Opět je i v tomto případě možnost volby více specifických operátorů a tudíž třída `ToneMapping` zaobaluje všechny vlastnosti, které jsou společné pro specifické operátory. Vstupem pro proces tónového mapování je objekt typu `HDRImage`,



který je nutné namapovat do zobrazitelného rozsahu. Hlavní funkcí této třídy je abstraktní funkce `ToneMapping::process()`. Tuto funkci je nutné pro každý tónový operátor, který dědí své vlastnosti od základní třídy, naimplementovat podle rovnic uvedených v sekci 2.5.1. Každý tónový operátor vyžaduje specifické číselné parametry, které jsou zadány v konstruktoru při vytváření objektu.

V této práci jsou využity celkem čtyři globální operátory tónového mapování a to konkrétně operátory uvedené následující tabulce.

Název operátoru	Třída v knihovně	Vstupní parametry
Lineární operátor	LinearOperator	–
Logaritmický operátor	LogOperator	$q \in [0, 1]$ , $k \in [0, 1]$
Exponenciální operátor	ExpOperator	$q \in [0, 1]$ , $k \in [0, 1]$
Reinhardův operátor	ReinhardGlobalOperator	–

Tabulka 4.1: Tabulka použitých tónovacích operátorů a k nim uvedeny odpovídající třídy.

Stejně tak jako u skládání HDR obrazu se i zde pracuje pouze s jedním obrazovým bodem v jeden moment. Výsledkem tohoto procesu je opět matice `cv::Mat`, ale tentokrát s typem `cv::CV_8UC3`, který reprezentuje `unsigned char`, jenž už jsme schopni zobrazit na dnešních běžných monitorech.

## 4.2 Komunikace s kamerou

Jednou z možností jak získat vstupní snímky pro algoritmus skládání fotografií je za použití připojených kamer skrze síťové rozhraní. Tyto kamery mi byly zapůjčeny pro testovací účely v rámci této práce skrze Ústav počítačové grafiky a multimédií. Jednalo se o dva typy kamer. Jedna z nich byla černobílá a druhá barevná. Obě však zasílaly informace jen o jasů jednotlivých obrazových bodů. Barva v barevné kameře se musela dopočítat za pomoci Bayerovy interpolace. Pro tuto interpolaci byla použita funkce z OpenCV knihovny `cv::cvtColor()`, která dokáže převést obraz z jakéhokoli barevného modelu do jiného. Konkrétně byla použita možnost `cv::CV_BayerGR2BRG`, jenž interpoluje každou čtveřici sousedních pixelů, kterou tvoří dva zelené body, jeden červený a jeden modrý bod, na hodnoty jasu pro RGB pixel.

### Využitá knihovna

Jak již bylo zmíněno, tak obě kamery komunikují skrze síťové rozhraní za pomoci protokolu definovaného v manuálu kamer. Pro komunikaci jsem tedy použil již předpřipravenou knihovnu. Tato knihovna však neobsahovala vše potřebné pro účely této práce, a tak musela být upravena. Kamera využívá pro komunikaci UPD pakety. Po počátečním navázání spojení je potřeba kameře zaslat informace o sekvenci, kterou má posílat. Tento proces je založen na přečtení konfiguračního souboru a poslání strukturovaných dat kameře. V konfiguračním souboru jsou kromě síťových adres kamery a lokálního počítače a taky portu, na kterém kamera naslouchá, uvedeny také informace o počtu snímků v sekvenci, clone pro všechny snímky, hodnotě ISO a hodnotách expozičního času pro jednotlivé snímky v sekvenci. Možnost editace tohoto konfiguračního souboru je v GUI aplikaci řešena pomocí formuláře s uvedenými hodnotami. Kamera má v sobě zabudovanou paměť, kde si ukládá informace o jednotlivých snímcích sekvence. Je tedy možno dopředu nahrát až osmnáct

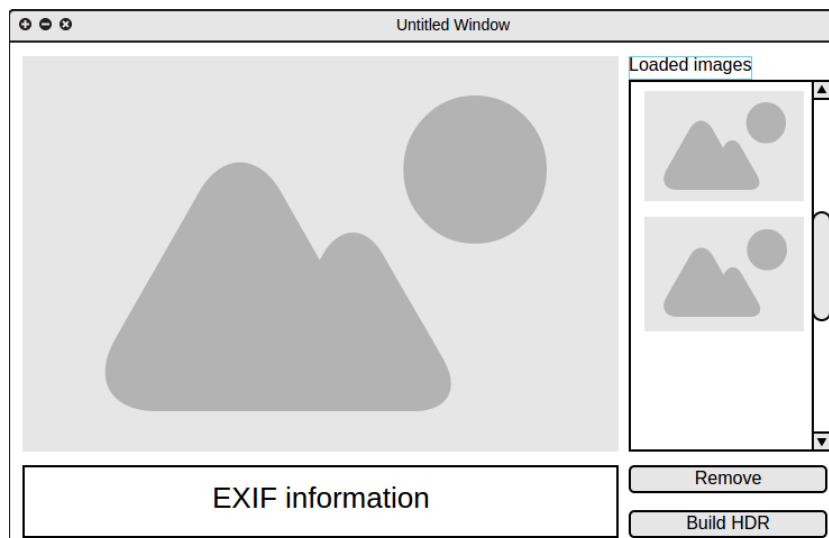
konfigurací pro požadované snímky. Poté už kamera posílá snímky po jednotlivých bajtech výsledného obrazu. Po poslání všech snímků je opět nutné zaslat informace o sekvenci, aby kamera věděla jaké snímky má pořizovat. Jednou za sto poslaných snímků je taky potřeba zaslat takzvaný „HeyYou“ paket, který stále udržuje komunikaci aktivní.

Knihovna disponuje třemi režimy poskytování snímků. Prvním z nich je pouhé zobrazení sekvence snímku na obrazovku. Jelikož kamera dokáže vyprodukovat okolo 12 snímků za sekundu, je ve výsledku z těchto snímků složeno video. O tento režim se stará funkce `CameraController::show()`. Druhým režimem je režim ukládání sekvence do souboru. Vznikne tak série fotografií uložená na disku. Tento režim je implementován ve funkci `CameraController::capture()`. Třetí režim je tvořen kombinací předchozích dvou a je k němu vytvořena odpovídající funkce `CameraController::capture()` s příznakem pro zobrazení výsledku na obrazovku.

Bohužel žádný z těchto režimů nebyl vhodný pro tuto aplikaci, a tak bylo potřeba přidat ke knihovně funkci, která zašle kameře informace o sekvenci snímků a vrátí seznam tří snímků v předpřipravené datové struktuře. Konkrétně se jedná o novou funkci `CameraController::getImages()`. V této metodě byly funkce knihovny použity pouze pro samotnou komunikaci s kamerou a vše ostatní bylo potřeba implementovat. Jelikož byl zvolen počet tří snímků v sekvenci, bylo nutné poslat kameře informace o této sekvenci spolu s patřičnými identifikátory. Tyto identifikátory je nutné si uložit, jelikož kamera spolu s hotovým snímkem pošle i příslušný identifikátor, pomocí kterého je možno spojit snímek s požadovaným expozičním časem. Expoziční čas je důležité si uložit spolu se snímkem, protože je potřebný v pozdějších fázích při skládání HDR obrazu.

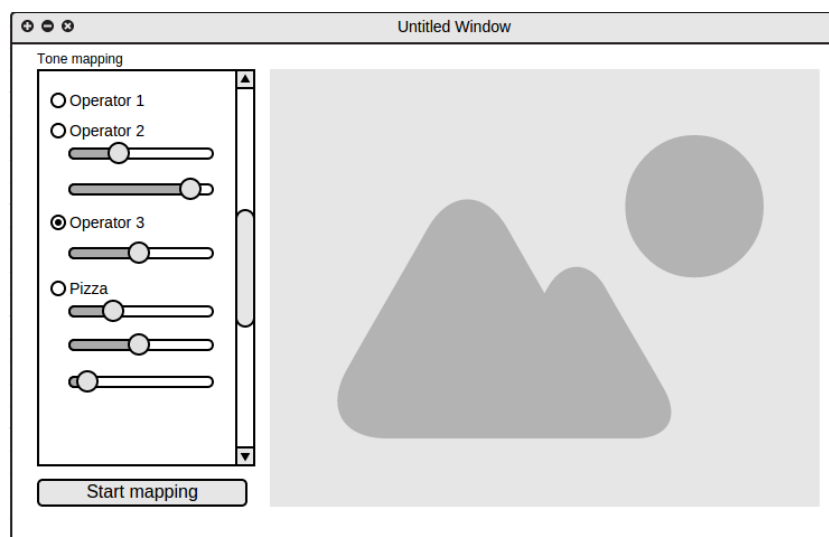
### 4.3 Grafické uživatelské rozhraní

Pro usnadnění používání knihovny a pro demonstraci funkčnosti jsem se rozhodl naimplementovat grafické uživatelské rozhraní. Celé rozhraní jsem se snažil navrhnout tak, aby jeho použití bylo přehledné a intuitivní. Pro implementaci rozhraní jsem použil jeden z nejpopulárnějších multiplatformních frameworků pro grafickou nadstavbu QT. Tento Framework zpřístupňuje základní nástroje pro tvorbu uživatelských rozhraní. Dalším z důvodů proč jsem zvolil tento framework, je ten, že obsahuje spoustu funkcí, které rozšiřují základní knihovnu pro C++ a značně zjednoduší funkčnost celého programu. QT nabízí dvě možnosti implementace rozhraní. Buď pomocí QT Widgetu čistě v jazyce C++ nebo pomocí Qt Quick, kde je možno použít kombinace C++ s jazykem QML, který vychází z jazyka JavaScript. Já jsem se rozhodl pro první variantu z důvodu předchozích zkušeností.



Obrázek 4.1: První okno pro načtení fotografií.

Aplikace je rozdělena do dvou oken. V prvním okně aplikace nabízí načtení a prohlížení vstupních snímků. V pravé části se nachází posuvná část, kde se zobrazují miniatury načtených snímků s možností kliknutí a zobrazení snímku v levé části ve větší velikosti. Informace o aktuálním snímku se nachází vespod, kde se zobrazují data získaná z exifu fotografie. Toto okno obsahuje i dvě tlačítka, z nichž jedno maže načtené fotografie a druhé zajistí přepnutí do dalšího okna a zároveň začne proces sestavení HDR snímku. Druhé okno je rozděleno na dvě části. V levé části se nachází nabídka tónovacích operátorů ve formě přepínačů. Uživatel si tedy vybere jeden z operátorů a popřípadě nastaví hodnoty vstupních parametrů, které si daný operátor žádá. Pro zadání korektních hodnot je použit posuvník. Jakmile je operátor navolen stačí kliknout na tlačítko, které se postará o tónové mapování a zobrazení výsledné fotografie v pravé části.



Obrázek 4.2: Druhé okno pro tónové mapování a zobrazení výsledné fotografie.

Společný pro obě okna je stavový řádek, který informuje uživatele o právě probíhajících činnostech. Dalším společným prvkem je i menu v horní části okna. Toto menu má dvě části. První z nich nabízí práci se soubory. Jedná se o otevírání snímků, export výsledných fotografií, uložení HDR snímku nebo načtení snímku z připojené kamery. Druhá část nabízí možnosti spojené s kamerou. Jelikož kamera ke svému chodu potřebuje vstupní informace o pořízení sekvence snímku, implementoval jsem i dialogové okno které umožní uživateli zadat potřebné parametry o expozici snímku.

Celá aplikace je velikostně škálovatelná, to znamená že obsah vnitřních prvků přizpůsobí svou velikost na základě velikosti celého okna a nedochází tedy k překrývání jednotlivých widgetů. Uživatelské rozhraní jsem se snažil ladit do tmavších odstínů, tak jak to známe z dnešních moderních programů, kde se také čím dál častěji prosazují tmavá témata.

## 4.4 Informace z exifu

Pro skládací algoritmus pro HDR je nutné znát informace o fotografii a to především o její expozici. Jedná se hlavně o rychlost uzávěrky, která je přímo ve vzorci skládání. Takovéto informace o fotografii je možné buďto manuálně zapisovat nebo je najít v metadatech o fotografii v takzvané části *exif*. V této práci jsem řešil dva způsoby získávání těchto informací. V prvním případě se jednalo o fotografie z připojené kamery přes síťové rozhraní. Tam bylo potřeba si pamatovat zasílané informace fotoaparátu o jednotlivých expozicích a následně je správně přiřazovat k zasílaným fotografiím z fotoaparátu. Tato problematika týkající se kamery připojené přes síťové rozhraní je detailněji popsána v sekci 4.2. V druhém případě, kdy uživatel může nahrát libovolnou fotografii ze svého fotoaparátu, je využita právě možnost zmiňovaných dat z exifu. Většina dnešních fotoaparátů umožňuje zápis těchto informací do souboru fotografie společně s obrazovými daty.

Pro získání dat z exifu jsem se rozhodl použít knihovnu Exiv2. Jedná se o C++ knihovnu společně s nástrojem pro příkazovou řádku, která dokáže snadno a rychle zpracovat data z metadat fotografie. Tato knihovna nabízí možnost čtení a zápisu nejen do exifu, ale i do IPTC a XMP metadat fotografie v různých formátech. Při módu čtení nabídne spoustu informací. Jedná se jak o standardní informace pro každou fotografii, tak i informace, které si volí každý výrobce fotoaparátu sám. V tomto programu byla využita jen mizivá část, z toho co tato knihovna nabízí. Jedná se hlavně o expoziční hodnoty spojené s clonou, rychlostí uzávěrky a hodnotou ISO. Kromě těchto hodnot jsou uživateli mimo jiné nabízeny informace o datu vytvoření a ohniskové vzdálenosti.

Název prvku	Klíč pro získání	Formát hodnoty
Šířka [px]	Exif.Image.XResolution	Rational
Výška [px]	Exif.Image.YResolution	Rational
Expoziční čas	Exif.Photo.ExposureTime	Rational
Clona	Exif.Image.ApertureValue	Rational
ISO	Exif.Image.ISOSpeedRatings	Short
Datum vytvoření	Exif.Image.DateTime	ASCII
Ohnisková vzdálenost	Exif.Image.FocalLength	Rational

Tabulka 4.2: Tabulka hodnot a klíčů pro získání informací o fotografii.

V programu byla tato knihovna použita při načítání fotografií mimo připojenou kameru. Při vytváření objektu `LDRImage` je zavolaná funkce `LDRImage::getExifInfo()`, která se postará o vytvoření hlavního objektu knihovny `Exiv2` `Exiv2::Image::AutoPtr` funkcí `Exiv2::ImageFactory::open(name)`. Aby bylo možné se dostat k jednotlivým položkám je nutné načít data funkcí `readMetadata()` a převést je do asociativního pole. Toto pole je možno indexovat řetězcem. Použité klíče k tomuto poli jsou v tabulce 4.2. Pro lepší pozdější zpracování bylo rozumné převést data z formátu `rational` do klasického datového typu `double`. Takto zpracovaná data jsou uložena v patričním objektu společně s fotografií.

## 4.5 Formáty ukládání

Fotografie jsou většinou spojené s formátem v jakém jsou uloženy v souboru. Tato knihovna umožní uživateli nahrát si i sérii fotografií ze svého fotoaparátu. Bylo tedy nutné naimplementovat podporu vstupních formátů pro načítání a uložení do matice, se kterou se dá dále pracovat. K tomuto problému jsem využil funkce z knihovny `OpenCV`. `OpenCV` totiž dokáže přečíst a poradit si se spoustou známých formátů. Konkrétně se jedná o funkci `cv::imread()`, která potřebuje ke svému provedení cestu k souboru a informaci o tom, v jaké bitové hloubce má předloženou fotografii číst a zda se jedná o černobílou či barevnou fotografii. Výsledek této funkce vrátí matici `cv::MAT`, ve které je uložena struktura fotografie spolu s dalšími informacemi. K podporovaným vstupním formátům tedy patří<sup>1</sup>:

- Windows bitmap (\*.bmp, \*.dib)
- JPEG soubory (\*.jpeg, \*.jpg, \*.jpe)
- Portable network graphics (\*.png)
- Portable image format (\*.pbm, \*.pgm, \*.ppm)
- Sun rasters (\*.sr, \*.ras)
- TIFF soubory (\*.tiff, \*.tif)

Výsledkem této aplikace je zobrazení HDR obrazu po aplikování tónovacího operátoru. Aby však byla výsledná fotografie přenositelná, tak byla ke knihovně přidána možnost ukládání konečného snímku. Stejně tak jako u načítání byla použita funkce `OpenCV` knihovny. Konkrétně se jedná o funkci `cv::imwrite()`, které je nutno předat informace o cestě, kde se má fotografie uložit a samotnou matici, která vznikla po procesu tónového mapování.

Jakýmsi mezistupněm mezi vstupními fotografiemi a snímkem po zpracování tónovým mapováním je obraz s vysokým dynamickým rozsahem. Jednotlivé hodnoty takového snímku převyšují rozsah běžných formátů pro ukládání fotografií. Z tohoto důvodu nelze použít pro uložení tohoto mezistupně výše zmiňované funkce. Rozhodl jsem se tedy pro použití formátu `RGBE`. Tento formát převede každou hodnotu obrazového bodu do klasického rozsahu 8 bitů. Aby byla zachována informace o původní hodnotě je přidán k trojici barevných kanálů `RGB` ještě společný exponent. Prvním krokem převodu na tento formát je spočítání společného exponentu. To je prováděno tím způsobem, že je vybrán barevný kanál s nejvyšší hodnotou a na něj je uplatněna funkce  $\log_2()$  a výsledná hodnota je posunuta offsetem 128. Pro tento výpočet je použita rovnice 2.12. Hodnoty pro každý barevný kanál jsou vypočítány za pomoci exponentu prostřednictvím rovnic 2.13 až 2.15.

<sup>1</sup>Jelikož názvy převážně nemají svůj český překlad, tak jsou ponechány v anglickém jazyku.

Zpětný převod je v této aplikaci taky použit, a to v případě, že chce uživatel nahrát již existující obraz s vysokým dynamickým rozsahem v tomto formátu a následně na něj použít tónovací operátory. U tohoto převodu se z exponentu a tří barevných složek opět zrekonstruují hodnoty vysokého dynamického rozsahu. Převod se provádí za pomoci rovnic 2.16 až 2.18.

Pro práci s tímto formátem byla použita volně dostupná knihovna ze stránek Bruce Waltera [3], který vytvořil soubory `rgbe.c` a `rgbe.h`, v nichž jsou jednoduché funkce pro zápis a čtení RGBE formátu. Soubor s takovýmto formátem obsahuje textovou hlavičku, kde jsou uvedeny základní informace o snímku a po ní následují samotná data. K uložení jsou pak použity funkce `RGBE_WriteHeader()` a `RGBE_WritePixels()` a to samé platí i u načítání, kde jsou použity funkce `RGBE_ReadHeader()` a `RGBE_ReadPixels_RLE()`.

## Kapitola 5

# Vyhodnocení výsledků

V této kapitole si představíme získané výsledky a popíšeme kvalitu jednotlivých tónovacích operátorů. Zapůjčenými kamerami bylo pořízeno několik ukázkových scén, kde každá série fotografií demonstrovala nedostatky techniky týkající se nízkého dynamického rozsahu. Fotografie byly tedy pořizované v interiéru, kde není tolik světla, ale zároveň byla v záběru i část venkovní prosluněné scény, kde bylo světla daleko více. Ve většině případů se jednalo o statické scény s tím, že byla kamera upevněna na stativu a expozice byla ovládána přes počítač, tak aby nedocházelo k posunům mezi fotografiemi.

Na všechny sekvence fotografií byl aplikován algoritmus pro skládání obrazu s vysokým dynamickým rozsahem a následně použit každý nabízený tónovací operátor. Na kvalitu těchto operátorů měla zásadní vliv i focená scéna, ale při větším vzorku dat se dají vypořádat určitá specifika každého z nich. V následujícím výčtu vlastnosti jsou tato specifika uvedena.

### Vlastnosti tónovacích operátorů

- **Lineární operátor** - jak je uvedeno v popisu tohoto operátoru v druhé kapitole, tak se jedná o pouze ukázkový operátor toho, jak vlastně tónové mapování funguje. Jeho výsledky nejsou nijak kvalitní, což je však logické, protože pokud se v obraze vyskytují velice jasné špičky, tak tím tento operátor zdegraduje všechny ostatní hodnoty v obraze a výsledný snímek je velice tmavý, v některých případech skoro úplně černý.
- **Logaritmický operátor** - výsledky tohoto operátoru jsou převážně lepší než u lineárního operátoru, ale bohužel obraz není příliš realistický. Většina snímků po použití logaritmického operátoru je zabarvena do šedých až bílých barev. Je to dáno vlastnostmi samotného logaritmu, z jehož grafu se dá vyčíst, že i pro velké vstupní hodnoty vykazuje malé výsledky a v kombinaci s rovnicí 2.21 dosahuje takovýchto výsledků. Výsledky tohoto operátoru můžeme řídit i vstupními konstantami. Testy na vzorku sekvencí však ukázaly, že tyto konstanty na výslednou kvalitu nemají takový vliv a spíše jen zesvětlují nebo ztmavují scénu.
- **Exponenciální operátor** - i když je tento operátor zařazen spíše do kategorie základních, tak překvapivě poskytuje docela dobré výsledky. Podle své charakteristiky je přesným opakem operátoru logaritmického a tudíž upřednostňuje tmavé barvy před světlými. Výsledné fotografie jsou tedy kvalitní a jejich barvy jsou výrazné. Stejně tak jako u předchozího operátoru jsou i zde konstanty, kterými lze výsledný obraz modifikovat. V tomto případě ale mají na výsledek velký vliv. Podle původního jasu snímku



lze doladit výsledný snímek tak, aby nebyl nikde přexponován nebo podexponován.

- **Reinhardův operátor** - tento operátor je z kategorie pokročilejších a na výsledcích je to znát. Na testovací sadě lze jasně prokázat, že se jedná o nejlepší snímky. Na snímcích s opravdu velkými jasově rozdílnými místy dokázal celou scénu namapovat tak, aby se přiblížil k tomu co můžeme vidět svými očima. V porovnání s exponenciálním operátorem si tento operátor poradí i s velice tmavými místy na obraze a navíc ve výsledku podá daleko reálnější barvy.



a)

b)

c)



d)

Obrázek 5.1: a), b), c) Sekvence snímků s různou expozicí pořizena síťovou kamerou odstupňovány čtyřnásobnou expoziční dobou. d) Výsledný snímek složen HDR algoritmem a tónově namapován Reinhardovým operátorem.



## Výkonnostní testy

Aplikace byla podrobena souborem výkonnostních testů, které měly za úkol zjistit jak se knihovna vypořádá s různě velkými snímky. Byla tedy vytvořena série tří snímků, která byla dále upravena na čtyři různá rozlišení. Tato testovací sada snímků byla převedena na snímky s vysokým dynamickým rozsahem a následně namapována zpět do zobrazitelného rozsahu pomocí tónových operátorů. Čas každého procesu byl získán objektem `QTime`. Každý proces byl opakován pět krát a výsledná hodnota udává aritmetický průměr. V tabulce uvedené níže je uveden přehled získaných časů pro jednotlivé procesy.

	Sekvence 1	Sekvence 2	Sekvence 3	Sekvence 4
Rozlišení [px]	250x187	500x375	1000x750	2000x1500
Složení HDR	49 ms	185 ms	754 ms	2,9 s
Lineární mapování	17 ms	54 ms	210 ms	835 ms
Logaritmické mapování	30 ms	110 ms	420 ms	1,68 s
Exponenciální mapování	21 ms	72 ms	275 ms	1,1 s
Reinhardovo mapování	47 ms	175 ms	700 ms	2,82 s

Tabulka 5.1: Tabulka zobrazuje časy jednotlivých procesů při různých rozlišeních snímků.

Z výsledků je patrné, že se zvyšováním rozlišení snímků lineárně stoupá i čas, jenž je potřeba pro vykonání algoritmu. Je to dáno tím, že se pracuje s každým obrazovým bodem, a tudíž zvýšení rozlišení čtyři krát znamená prodloužení doby na čtyřnásobek. Nejnáročnějším procesem je určitě proces samotného skládání vstupních snímků. Toto je taky opodstatněné, jelikož proces pracuje s třemi snímky a propočítává každý jejich bod. Co se týče poměru kvality a strávené doby u tónového mapování, tak jednoznačným vítězem je exponenciální operátor, který trvá nejkratší dobu, pokud pomíneme ukázkový lineární operátor, a má velice kvalitní výsledky.

## Kapitola 6

# Závěr

Tato práce si kladla za cíl vyřešit nedostatky nízkého dynamického rozsahu dnešních fotoaparátů. Na základě tohoto problému byla navržena a následně vytvořena knihovna v jazyce C++, jež dokáže vytvořit snímek s vysokým dynamickým rozsahem za pomoci metody skládání snímků s různou expozicí. Knihovna navíc umožní uživateli zvolit si mezi více možnostmi tónového mapování, které je součástí procesu tvorby HDR fotografie. Algoritmus skládání fotografií však zatím podporuje pouze statickou scénu, kterou není potřeba zarovnávat.

Oproti původnímu zadání se navíc podařilo propojit knihovnu s kamerou, jež poskytuje série snímků s nízkým dynamickým rozsahem s různou expozicí a celá aplikace je zabalená do jednoduchého, avšak dobře fungujícího a přehledného uživatelského rozhraní. Dalším prvkem, který se podařilo vytvořit nad rámec původního zadání práce, jsou funkce pro ukládání a manipulaci se snímkem s vysokým dynamickým rozsahem ve formě HDRE formátu. Součástí této práce bylo i rozšíření knihovny, která se stará o komunikaci s kamerami o další funkcionalitu demonstrovanou ve výsledné aplikaci.

Vytvořená aplikace byla otestována na sérii snímků a ukázalo se, že i exponenciální operátor, jež patří spíše do kategorie základních může poskytovat velice dobré výsledky. Z testů také vyplynulo, že doba po kterou se výpočty realizují je dostatečně krátká na to, aby uživatele moc nezdržovala při tvorbě HDR fotografie.

Možností jak dále pokračovat v této práci je hned několik. Jednou z nich je přidání dalších možností tónového mapování, a to konkrétně z kategorie lokálních operátorů, které mohou vytvářet kvalitnější HDR fotografie. V tom kontextu je ale nutné si uvědomit, že kvalita fotografie je každým jedincem vnímána individuálně a neexistuje jednoznačný způsob, který by určil, která fotografie je lepší. Dalším možným pokračováním by mohlo být aplikování knihovny na video. V tomto případě by se ale musely optimalizovat algoritmy tak, aby jejich provedení trvalo podstatně kratší dobu a tím pádem by bylo možné vytvářet HDR video v reálném čase.

V poslední době je HDR fotografie velice rozšířena mezi uživateli, kteří v ní vidí pouze krásnou scénu, jež je mnohdy až za hranicí reálného vidění. Avšak největší přínos fotografie s vysokým dynamickým rozsahem vidím spíše v odvětvích průmyslu, kde může mít daleko širší a užitečnější využití. Právě na tyto případy je vytvořená knihovna ideální. V nově vznikajících projektech zaměřujících se na práci s obrazem (například rozpoznávání SPZ, kontrola výrobních vad, a mnoho dalších) mohou vývojáři tuto knihovnu použít a soustředit více na řešený problém a nikoli na získání lepšího obrazu formou HDR.

# Literatura

- [1] Banterle, F.: *Advanced high dynamic range imaging : theory and practice*. Natick: PETERS A. K., první vydání, 2011, 260 s., iISBN 978-1-56881-719-4.
- [2] Bočík Andrej, H. M. a. M. B.: *Velká kniha HDR fotografie: kouzlo fotografií s vysokým dynamickým rozsahem*. Brno: Computer Press, druhé vydání, 2011, 224 s., iISBN 978-80-251-3367-5.
- [3] Bruce, W.: RGBE Image Format Specifications. online, [cit. 11. 5. 2015].  
URL <http://www.graphics.cornell.edu/online/formats/rgbe/>
- [4] Březina, P.: Barevné prostory a fotografování. online, [cit. 5. 5. 2015].  
URL <http://digiarena.e15.cz/barevne-prostory-a-fotografovani-pro-zacatecniky>
- [5] Debevec Paul, M. J.: Recovering High Dynamic Range Radiance Maps from Photographs. online, [cit. 11. 5. 2015].  
URL <http://www.pauldebevec.com/Research/HDR/debevec-siggraph97.pdf>
- [6] EasyHDR: HDR Photography - Multiple Exposures. online, [cit. 5. 5. 2015].  
URL <http://www.easyhdr.com/examples/>
- [7] Granados Miguel, B. A. a. M. W.: Optimal HDR reconstruction with linear digital cameras. In *CVPR*, 2010, s. 215–222.
- [8] Höddlinger, B.: *High-dynamic-range (HDR) vision : microelectronics, image processing, computer graphics*. Natick: PETERS A. K., první vydání, 2011, 260 s., iISBN 978-1-56881-719-4.
- [9] IMS Chips: HDRC - More than You Can See. online, [cit. 2. 5. 2015].  
URL [http://www.ims-chips.de/content/pdf/text/HDRC\\_Imager\\_Camera\\_Feature3.pdf](http://www.ims-chips.de/content/pdf/text/HDRC_Imager_Camera_Feature3.pdf)
- [10] Mediachance: Dynamic photo HDR. online, [cit. 11. 5. 2015].  
URL <http://mediachance.com/hdri/help/index.html?options.htm>
- [11] Pihan, R.: Vše o formátu RAW. online, [cit. 5. 5. 2015].  
URL <http://www.digimanie.cz/vse-o-formatu-raw-1dil/2182>
- [12] Reinhard Erik, P. D.: *High dynamic range imaging : Acquisition, Display, and Image-Based Lighting*. Burlington: Morgan Kaufmann, druhé vydání, 2010, 650 s., iISBN 978-0-12-374914-7.

- [13] Reinhard Erik, S. M.: Photographic Tone Reproduction for Digital Images. online, [cit. 5. 5. 2015].  
URL <http://www.cs.utah.edu/~reinhard/cdrom/tonemap.pdf>
- [14] Richards, G.: BrightSide DR37-P HDR display. online, [cit. 5. 5. 2015].  
URL [http://www.bit-tech.net/hardware/2005/10/04/brightside\\_hdr\\_edr/8](http://www.bit-tech.net/hardware/2005/10/04/brightside_hdr_edr/8)
- [15] SIM2 Multimedia: HDR47E S 4K. online, [cit. 5. 5. 2015].  
URL [http://www.sim2hdr.com/hdrdisplay/hdr47e\\_s\\_4k](http://www.sim2hdr.com/hdrdisplay/hdr47e_s_4k)
- [16] Čadík, M.: Evaluation of tone mapping operators. online, [cit. 11. 5. 2015].  
URL <http://cadik.posvete.cz/tmo/>

# Příloha A

## Obsah DVD

Součástí této práce je i přiložené DVD, které obsahuje následující části:

- **HDRLib** - zdrojové kódy knihovny a grafické aplikace
- **Images** - složka obsahující vstupní i výstupní fotografie
  - **Sequence** - vstupní sekvence snímků s různou expozicí
  - **RGBE output** - snímky s vysokým dynamickým rozsahem ve formátu HDRE
  - **Tonemapping output** - výsledné snímky po použití metod tónového mapování
  - **Test** - snímky použité při testování výkonnosti knihovny
- **Latex src** - zdrojové soubory tohoto textu
- **Text** - výsledné pdf soubory tohoto textu ve dvou verzích (pro tisk a s hypertextovými odkazy)
- **Libs** - knihovny potřebné k spuštění aplikace