



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**POŘIZOVÁNÍ HDR SNÍMKŮ NA MOBILNÍCH
ZAŘÍZENÍCH**

HDR IMAGE ACQUISITION ON MOBILE DEVICES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MICHAL PADYŠÁK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MUSIL MARTIN

BRNO 2017

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2016/2017

Zadání bakalářské práce

Řešitel: **Padyšák Michal**

Obor: Informační technologie

Téma: **Pořizování HDR snímků na mobilních zařízeních
HDR Image Acquisition on Mobile Devices**

Kategorie: Zpracování obrazu

Pokyny:

1. Prostudujte dostupnou literaturu týkající se pořizování HDR obrazu ze sekvence snímků s různou expozicí. Uvažujte statickou scénu.
2. Seznamte se s vývojem aplikací pro platformu Android a prostudujte možnosti pořizení sekvence snímků s proměnnou expozicí pomocí kamery zvoleného zařízení.
3. Implementujte aplikaci, která pořídí sekvenci snímků potřebnou pro složení HDR obrazu a aplikuje vybranou metodu tone-mappingu pro zobrazení na displeji zařízení.
4. Diskutujte dosažené výsledky a navrhněte možné pokračování práce.

Literatura:

- Hoefflinger, B. "High-Dynamic-Range (HDR) Vision: Microelectronics, Image Processing." *Computer Graphics (Springer Series in Advanced Microelectronics)*, Springer-Verlag New York, Inc., Secaucus, NJ (2007).
- Banterle, Francesco, et al. *Advanced high dynamic range imaging: theory and practice*. CRC Press, 2011.
- Reinhard, Erik, et al. *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Musil Martin, Ing.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2016

Datum odevzdání: 17. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Tato bakalářská práce se věnuje problematice digitální fotografie s vysokým dynamickým rozsahem. Řešení této problematiky je provedeno složením několika snímků stejné scény s různou expozicí do jednoho snímku s vysokým dynamickým rozsahem. Metoda je implementována jako aplikace pro operační systém Android a pořízený snímek je schopna zobrazit na svém displeji.

Abstract

This bachelor thesis deals with high dynamic range in digital photography. This problem is solved by composing multiple images with different time exposure of the same scene into single one including high dynamic range. Method is implemented as an application for Android operating system and the result can be shown on display of the device.

Klíčová slova

Vysoký dynamický rozsah, tónové mapování, digitální fotografie, mobilní aplikace

Keywords

High dynamic range, tone mapping, digital photography, mobile application

Citace

PADYŠÁK, Michal. *Pořizování HDR snímků na mobilních zařízeních*. Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Martin Musil

Pořizování HDR snímků na mobilních zařízeních

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Martina Musila. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Michal Padyšák

17. května 2017

Poděkování

Chci poděkovat svému vedoucímu Ing. Martinu Musilovi za jeho ochotu, věcné připomínky a cenné rady. Dále také za zapůjčení pomůcek, které mi přispěli k vypracování této bakalářské práce.

Obsah

1	Úvod	2
2	High Dynamic Range a další pojmy	3
2.1	Digitální fotografie	3
2.2	Vysoký dynamický rozsah	7
2.3	Tone-mapping	12
3	Specifikace zadání	16
3.1	Postup řešení	16
3.2	Implementace	17
4	Realizace	19
4.1	Grafické rozhraní	19
4.2	Implementace	20
5	Měření a výsledky	24
5.1	Rychlost zpracování	24
5.2	Tónové mapování	25
6	Závěr	30
	Literatura	31
A	Obsah DVD	33

Kapitola 1

Úvod

Běžně se setkáváme s problémem, kdy je snímek digitální fotografie pořízen jako příliš tmavý nebo světlý. Může za to technologie snímače fotoaparátu, protože dokáže zachytit pouze malý podíl jasu nejsvětějšího a nejtmavšího bodu – dynamického rozsahu. Snímek je pak zobrazován na displejích, které už sice dokáží širší dynamický rozsah z části zobrazit, stále ale zůstane zakryta podstatná část scény. Jelikož je lidské oko opravdu hodně citlivý snímač, dokáže vnímat scénu ve vysokém dynamickém rozsahu – tento rozsah je označován zkratkou HDR¹.

Problém snímání i zobrazení lze z části obejít, i když se pak nedosahuje plné věrohodnosti a je na první pohled jasné, že je něco v nepořádku. Stále ale můžeme na snímku zobrazit poměrně velkou část informací, které by jinak byly zahozeny.

Pro vytvoření dokonalé fotografie se stále nevyvinul dostatečně dokonalý snímač, který by zachytil scénu s tak vysokým dynamickým rozsahem, a proto se využívá buď několika snímačů zároveň, nebo se scéna zachytí několikrát s různým nastavením expozice. Každý takto pořízený snímek má nízký dynamický rozsah – ten je označován jako LDR².

Obraz, který byl vytvořen z několika LDR snímků s různou expozicí je možné zobrazit věrohodněji původní scéně. K tomu je však ještě zapotřebí maska, která upravuje jasy jednotlivých bodů. Existují ale i poměrně nákladné displeje, které dokáží tento rozsah téměř celý zobrazit. Je-li k dispozici displej, který zobrazuje pouze omezený dynamický rozsah, pak je možné obsah s HDR i tak zobrazit. K tomu nám pomáhá tónové mapování a lze s ním důležité informace ze snímku zobrazit.

V kapitole 2 je rozebrána problematika HDR, vytváření snímku, jeho ukládání a zobrazení. Jako podkapitola je zde rozebráno i tónové mapování. Kapitola 3 specifikuje dopodrobna, co je cílem této práce a jak bude provedena. V další kapitole 4 je obsažen návrh grafického rozhraní aplikace a detaily implementování. Kapitola 5 obsahuje výsledky měření, které ukazují rozdíly použitých metod při implementaci a také porovnání různých výkonů zařízení, na kterých byla aplikace spuštěna. Poslední kapitola 6 pak hodnotí výsledky této práce a uvádí další možné pokračování práce.

¹High Dynamic Range, více v kapitole 2

²Low Dynamic Range

Kapitola 2

High Dynamic Range a další pojmy

High Dynamic Range (HDR), nebo také High Dynamic Range Imaging (HDRI). Používá se hlavně v oblasti fotografií pro zachování věrohodnosti a v herním průmyslu pro zpříjemnění herního prožitku. Ve spojení s patřičnou zobrazovací technikou je HDRI blízko realitě. Nejdříve si ale ukážeme základní pojmy potřebné pochopení této problematiky.

2.1 Digitální fotografie

Fotoaparáty používají různé snímače, techniky a práci s barvami. V této podkapitole je popsán způsob, jakým je možné získat barvu z reálného světa na snímači. Dále druhy barevných prostorů, ve kterých je možné barvu reprezentovat. Nakonec parametry, které jsou uplatněny při pořizování snímku.

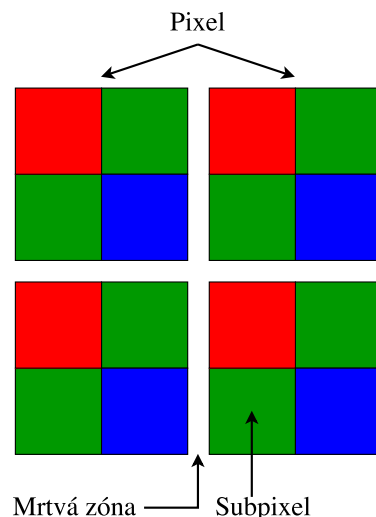
2.1.1 Snímač

Budeme-li vycházet z [10], tak snímání digitálního obrazu probíhá ve specializovaném integrovaném obvodu s imlementovanými senzory, ty převádí příchozí fotony na úměrné elektrické napětí. Jako na kinofilmu, tato událost probíhá na celém snímači v jeden moment. Každá hodnota pixelu pak je převedena do sekvence bitů a tu je možné uložit – vše proběhne v řádu stovek milisekund.

Pro zachycení digitálního obrazu scény bylo vyvinuto mnoho technologií, mezi ně se mimo jiné řadí CCD¹ a CMOS². Obě obsahují matici senzorů (pixelů) a vodiče pro ovládání, měření a přenos dat. Optika před tímto snímačem zajišťuje, že bude možné scénu změřit právě přes integrované senzory. Pokud je snímek barevný, musíme hodnoty červené (R), zelené (G) a modré (B) změřit zvlášť – většina snímačů CCD a CMOS používá tzv. *Bayerovu masku* (zobrazena na obrázku 2.1). Každý subpixel má svůj barevný filtr, aby mohl měřit jen jednu barevnou složku. Červený pro červenou, zelený pro zelenou a modrý pro modrou. Nevýhodou této masky je ztráta barevné informace. Pokud například dopadne foton, jehož vlnová délka reprezentuje červenou barvu, na zelený subpixel – není mu umožněn průchod a nedopadne. Stejný případ platí i pro situaci, kdy dopadne do mrtvé zóny mezi pixely. Mrtvou zónou je myšlen prostor bez senzoru.

¹Charge Coupled Devices

²Complementary Metal Oxide Semiconductor



Obrázek 2.1: Čtyři pixely Bayerovy masky, které obsahují 4 subpixely. Každý subpixel má barevný svůj filtr.

2.1.2 Barevný prostor

Dle knihy [8] rozumíme barevným prostorem rozsah barev v rámci viditelného spektra. Dále je zmíněno, že barevný prostor je většinou variantou nějakého barevného modelu. Podkapitola zahrnuje představení původního modelu CIE a modelu RGB.

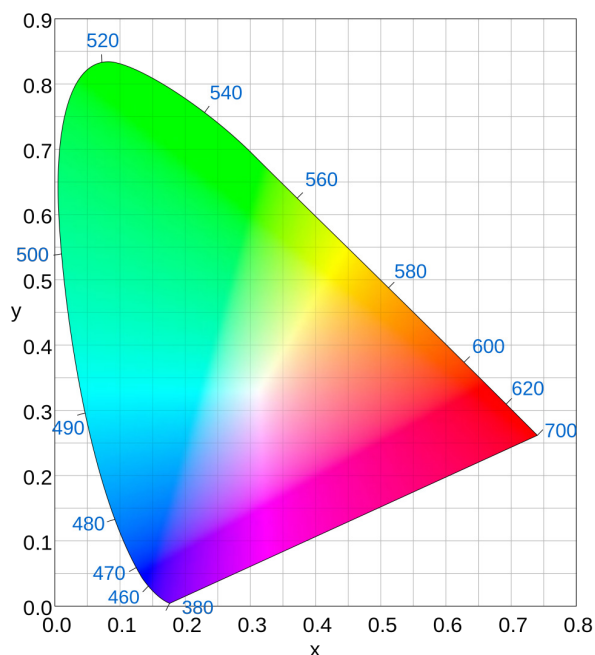
CIE 1931

Ve stejné knize je uvedeno, že matematicky definovaný barevný prostor je odvozený podle vlastností lidského oka. Lidské oko totiž vnímá barvy třemi druhy fotoreceptorů na sítnici oka (tzv. trichromatickými členiteli), které byly stanoveny na základě empirického měření velkého počtu pozorovatelů. Je používán v situacích, kdy záleží na přesném matematickém vyjádření barvy.

Každou barvu je možné v trichromatické měrné soustavě popsat. Jednak množstvím tří barevných podnětů (tzv. trichromatickými složkami) – ty jsou označovány velkými písmeny X, Y a Z. Jejich výpočet je proveden integrací spektrálního barevného podnětu a trichromatických členitelů v celém rozsahu viditelného spektra. Dále je barvy možné popsat poměrem tří barevných podnětů (tzv. trichromatickými souřadnicemi) vyjádřených pomocí normových podílů x, y a z , kde

- $x = X/(X + Y + Z)$ je normový podíl červené barvy,
- $y = Y/(X + Y + Z)$ je normový podíl zelené barvy,
- $z = Z/(X + Y + Z)$ je normový podíl modré barvy.

Trichromatické složky tvoří kolorimetrický prostor CIE XYZ. V praxi se však prostorové zobrazení v souřadnicích X, Y a Z jeví jako nepraktické, protože v řadě případů bylo potřeba uvažovat pouze chromatičnost barvy – k tomu stačí pouze hodnoty X a Y, ignorován je tedy jas. Redukce na dvě hodnoty umožnila používat graficky jednodušší rovinný řez tímto prostorem (tzv. chromatický diagram, obrázek 2.2), do něj je možné zakreslovat různé jevy (tím může být například gamut barevného prostoru).



Obrázek 2.2: Chromatický diagram CIE 1931 [1].

Z tohoto prostoru byly vytvořeny dva další a to CIELAB a CIELUV. Převody mezi prostory jsou možné jednoduchým přepočtem. Informace o těchto úpravách a důvodech vzniku jsou v [2].

RGB

Model RGB je založen na aditivním míchání barev a je reprezentován třemi barevnými složkami – červenou, zelenou a modrou. Dle [6] je ho možné reprezentovat graficky v RGB kostce (na obrázku 2.3). Jakýkoliv bod uvnitř kostky reprezentuje barvu. Pokud bychom vytvořili diagonálu ze souřadnic $(0, 0, 0)$ až do $(1, 1, 1)$, získáme tak všechny hodnoty šedé, začínající z absolutní černé do absolutní bílé.

V praxi je model používán na různorodých zobrazovacích zařízeních, které používají různé rozsahy pro barevný kanál. Běžně může být jeden kanál reprezentován 8-bitovou hodnotou 0 – 255 nebo 16-bitovou 0 – 65535, více není prakticky potřeba.

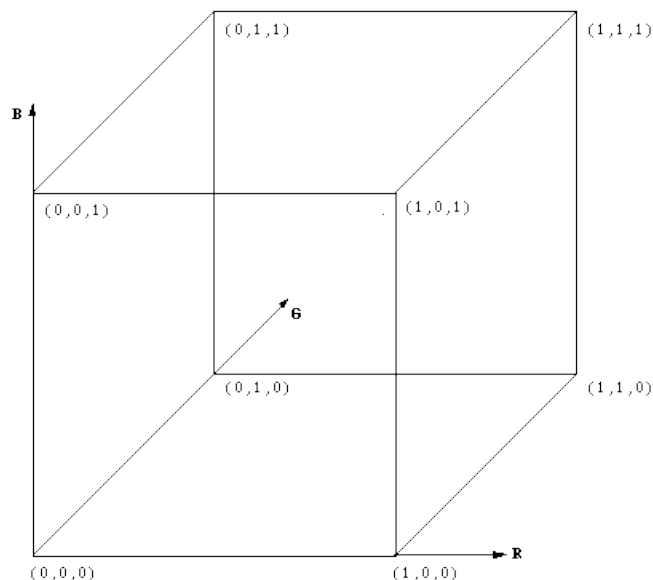
Na tomto modelu jsou založeny i další, například sRGB nebo Adobe RGB [18].

2.1.3 Parametry vzniku fotografie

Výsledná fotografie je závislá na nastavení několika parametrů. Patří mezi ně volba ISO, clony a délka expozice. Jejich nastavením je možné měnit část světelného rozsahu, který je zachycen. Změna ISO přidává šum a změna clony hloubku ostření – proto je nejčastěji používána změna délky expozice.

ISO

Jak je zmíněno v knize *Understanding Digital Photography* [15], pro klasický film je ISO míra, jakou reaguje na světlo. Podobný vliv funguje i u digitální fotografie, tady je však



Obrázek 2.3: RGB kostka s koordináty a klíčovými barvami. V místě $(0,0,0)$ je černá, $(1,1,1)$ je bílá, $(0,1,1)$ je azurová, $(1,1,0)$ je žlutá a $(1,0,1)$ je purpurová.

citlivost stále stejná – musí se tedy použít jiná metoda. Každý bod ze snímáče je před předáním výsledného snímku, třeba na paměťovou kartu nebo displej, vynásoben právě mírou nastavené hodnoty ISO. Touto metodou, kdy vlastně nepřímo upravujeme citlivost každého bodu, lze dosáhnout uspokojivých výsledků i při vytváření snímku v šeru nebo tmě. Má však jednu velkou nevýhodu, a tou je stále se zvětšující míra šumu spolu s vyšší hodnotou ISO.

Clona

Tento výraz je nejčastěji používán u zrcadlovek. Jedná se o mechanické stínítko, které dokáže regulovat průchod světla k senzoru. Jelikož jsou ale mobilní zařízení omezená svým prostorem, není časté, aby byla možnost regulace clony dostupná. Tohle stínítko je tedy mnohdy předem nastaveno výrobcem. Jednotka clony je označena písmenem F . V uvedeném vzorci 2.1 je hodnota clony určena poměrem ohniskové vzdálenosti (f) a průměrem otvoru clony (d).

$$F = \frac{f}{d} \quad (2.1)$$

Expozice

Dalším pojmem je expozice [13] [5]. Tvoří ji celkem tři parametry, jedním je již prezentovaná clona, druhým také prezentované ISO a třetím je čas. U běžné fotografie je nežádoucí, aby byl snímek podexponovaný nebo přexponovaný, protože jsou skryté světlé nebo naopak tmavé detaily. Pokud je délka expozice kratší, tak je snímek tmavý a vyniknou detaily hodně světlých objektů – například vlákno svítící žárovky. Pokud je délka expozice dlouhá, tak může být snímek velice jasný ale vyniknou detaily tmavých objektů – například objekt ve stínu. Z těchto důvodů je nastavení expozice využíváno v HDR fotografii.

Kombinací předchozích parametrů je upravováno množství světla dopadajícího na snímač – to lze vyjádřit absolutně tzv. *expoziční hodnotou*, zkráceně EV³. Udává poměr mezi nejtmaším a nejsvětlejším bodem a stupnice EV je logaritmická. Hodnotou 0EV je rozuměno nastavení ISO na 100, clona $F = 1$ a délka expozice je 1 vteřina. Pokud chceme upravit expoziční hodnotu, lze toho dosáhnout následujícími vzorci

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

$$ISO_{EV_{n+1}} = 2 * ISO_{EV_n}, \quad (2.3)$$

$$F_{EV_{n+1}} = \frac{F_{EV_n}}{\sqrt{2}}. \quad (2.4)$$

V uvedených vzorcích 2.2, 2.3 a 2.4 vyjadřuje T čas, ostatní významy označení již byly uvedeny v předchozích podkapitolách.

2.2 Vysoký dynamický rozsah

V této podkapitole je obsaženo vysvětlení jak je HDR snímek vytvořen z více LDR snímků, formáty uložení a způsoby zobrazení na LDR displejích. Je čerpáno z knihy [17] a [4].

Lidské oko vnímá okolní svět ve vysokém dynamickém rozsahu, to však neplatí pro snímače zařízení, které dennodenně používáme. Navíc většina mobilních zařízení má pro zobrazování obrazu vestavěný LDR displej, a proto je nutné obsah HDR upravovat. Není to však jen zobrazování, které nás omezuje pro práci s HDR, je to i kamera.

2.2.1 Vytvoření HDR snímku

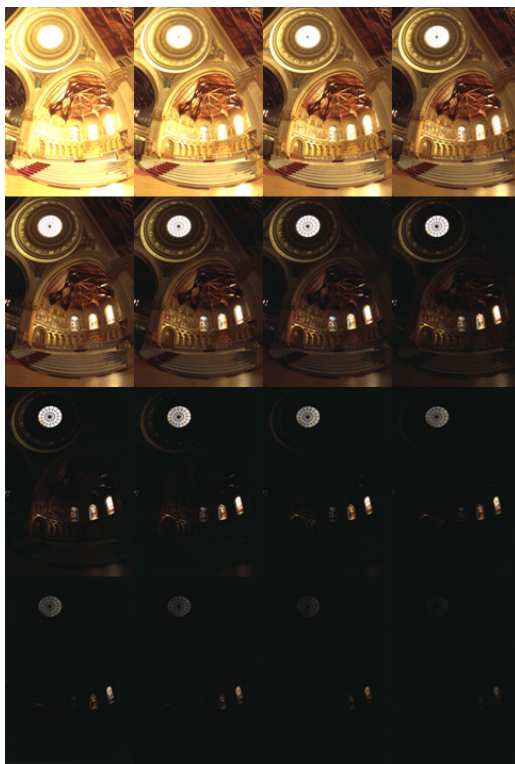
Pro sestavení HDR snímku z několika LDR snímků, jsou zapotřebí snímky s různou expozicí. Právě tak můžeme získat potřebné informace z širšího dynamického rozsahu, jelikož každý takový snímek obsahuje informace z jiných regionů snímku.

Jestliže že má snímač zařízení perfektní odezvu, může být každý snímek vydělen svým časem expozice do stejné domény pro další práci. Pokud jsou všechny snímky ve stejných jednotkách, mohou být zprůměrovány – zároveň jsou odstraněny nad a pod exponované pixely. Výsledkem tohoto procesu je HDR snímek. Za předpokladu, že byla použita váhová funkce – jejíž popis je pak uveden v 2.2.1, je snímek vypočítán následujícím vzorcem

$$L_{ij} = \sum_{k=1}^N \frac{Z_{ij} w(Z_{ij})}{\Delta t_k} / \sum_{k=1}^N w(Z_{ij}). \quad (2.5)$$

Zde je délka expozice dána Δt_k pro expozici k . Bohužel nejsou ve skutečnosti snímače tak perfektní, aby měly lineární odezvu. Pokud nám záleží na profesionální věrohodnosti, je třeba tuto záležitost zohlednit. Pro běžné použití však stačí právě odezva lineární. Použitím inverzní funkce odezvy snímače, kterou aplikujeme na hodnoty pixelu, dokážeme získat korektní expozice. Funkci je možné dopočítat ze série seřazených expozic nebo ji získat od výrobce snímače. Existují například techniky *Mann and Picard*, *Debevec and Malik* nebo

³Z angl. exposure value



Obrázek 2.4: 16 snímků jedné scény zachycené s různými expozičními, jejich kombinací lze sestavit HDR snímek a mapu jasů. Obrázek je složen z expozičních dostupných v [9].

Mitsunagara and Nayar [17]. Předchozí vzorec je pak upraven právě o inverzní funkci

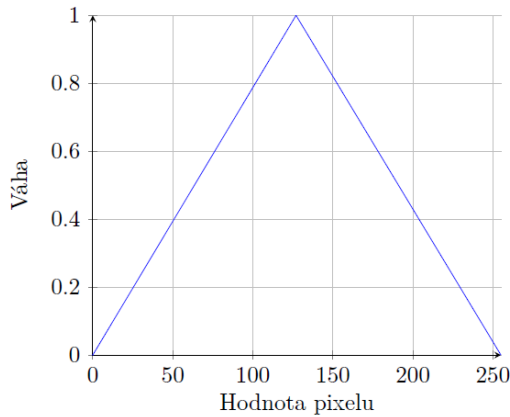
$$L_{ij} = \sum_{k=1}^N \frac{f^{-1}(Z_{ij})w(Z_{ij})}{\Delta t_k} / \sum_{k=1}^N w(Z_{ij}). \quad (2.6)$$

Jedna z nevýhod této metody, získání HDRI z několika LDR snímků, je právě v nutnosti použít více snímků z různých časů pořízení. Snímky například nemusí být mezi sebou vyrovnány a je třeba toto vyrovnání provést. V průběhu pořizování snímků se také může nějaký objekt pohybovat, takže je pak ve výsledku nepříjemný efekt ducha. [17]

Kombinací LDR snímků, dokážeme také reprodukovat mapu jasů (E) jednotlivých pixelů pro výsledný HDR snímek, to ovšem za stejného předpokladu, jestliže má kamera stejnou odezvu snímače jako při vytváření HDR snímku. Mapa nám je pak užitečná při zpětné reprodukci snímku pro zobrazení v celém svém dynamickém rozsahu, za předpokladu, je-li k dispozici již tónově namapovaný snímek. Více v podkapitolách 2.2.2 a 2.3.

$$E(x) = \frac{\sum_{i=1}^{N_e} \frac{1}{\Delta t_i} w(I_i(X)) I_i(X)}{\sum_{i=1}^N w(I_i(X))}, \quad (2.7)$$

kde I_i značí snímek i té expoziční, Δt_i je délka expoziční snímku I_i , N_e je počet snímků s různou expoziční a $w(I_i(X))$ je váhová funkce, která odstraňuje nežádoucí části. Výsledek použití tohoto vzorce je vyobrazen v pravém snímku obrázku 2.8. [4]



Obrázek 2.5: Grafická reprezentace průběhu váhové funkce Debevec a Malik 2.8

Váhová funkce

Ve vzorcích 2.5, 2.6 a 2.7 je jejich součástí váhová funkce, která přidává různou váhu různým bodům ve snímku. Váhou je rozuměna míra, jakou bude výsledný snímek bodem ovlivněn. Cílem je dát nižší váhu přeexponovaným nebo naopak podexponovaným bodům, které by mohly ve výsledku tvořit šum nebo artefakty, což je nežádoucí. Tvar průběhu váhových funkcí je několik, některé se soustředí jen na extrémní expozice, jiné zase jen na určitou část. Současně záleží na snímcích, na které bude funkce aplikována, nic ale nebrání vytvořit si svůj vlastní průběh funkce, protože nakonec budeme my, kdo rozhodne, jestli je výsledná výstupní kvalita dostatečná. Pro ukázkou je zde uvedeno několik těchto funkcí. Uvedený zápis je pro váhovou funkci od pánů Debevec a Malik

$$w(x) \begin{cases} x - Z_{min} & \text{pokud } x \leq \frac{1}{2}(Z_{max} + Z_{min}), \\ Z_{max} - x & \text{pokud } x > \frac{1}{2}(Z_{max} + Z_{min}). \end{cases} \quad (2.8)$$

2.2.2 Formáty pro uložení HDR obsahu

Když je snímek s HDR vygenerovaný, je vhodné ho také uložit pro pozdější prohlížení. Snímek samotný je složen z pixelů, které v nekomprimované podobě používají pro model RGB 3 samostatné hodnoty v plovoucí desetinné čárce. To při použití rozměrové velikosti 1920×1080 a velikosti pixelu v paměti 12 bajtů znamená celkem 24 MB. Je však zbytečné uchovávat snímek takhle velký, pokud s ním ihned nezamýšlíme nějakou další práci. Součástí tohoto kroku je tedy i komprimace, aby snímek v zařízení nezabíral příliš mnoho místa. Komprimace samotná je ale časově náročnější, proto je vhodné ji provádět až jako poslední. Uložení v komprimované podobě může být například provedeno až po zobrazení na displeji zařízení.

Pro pořizování sekvence několika rychlých snímků za sebou, nebo rovnou videa, ještě nejsou mobilní zařízení připravena, avšak vhodným kompromisem lze tuto záležitost provést. Například snížením rozlišení, počtem vstupních snímků LDR s různou expozicí pro výpočet HDRi nebo frekvencí jednotlivých snímků.

0	8	16	24	31
8-bit červená	8-bit zelená	8-bit modrá	8-bit exponent	

Obrázek 2.6: 32-bitů pixelu zakódovaného ve formátu RGBE

0	1	16	24	31
±	15-bit logL	8-bit u	8-bit v	

Obrázek 2.7: 32-bitů pixelu zakódovaného ve formátu LogLuv. V tomto kódování je používán jeden bit pro znaménko, 15 bitů pro zakódování jasu, další dva bajty pak pro vyjádření barvy v barevném prostoru CIELUV (více v 2.1.2), kvůli dosažení perceptuální uniformity.

Každý formát má své výhody a nevýhody. Pro použití na mobilních zařízeních nám jde o rychlost zpracování a o výslednou velikost snímku, proto je třeba zvolit vhodnou metodu podle našich požadavků. Specifikace formátů jsou čerpány z publikací [4] a [12].

RGBE

Formát někdy nazývaný také jako Radiance HDR (.hdr, .pic), je to jeden z prvních formátů. Datový soubor je tvořen textovou hlavičkou za kterou následuje zakódovaný snímek. V hlavičce jsou základní informace o snímku. Pixely jsou zakódovány v RGBE nebo také XYZE reprezentaci, rozdíl je pouze v použitém barevném prostoru.

RGBE pixel je tvořen 4 bajty. První tři zastupují červený, modrý a zelený kanál. Poslední pak vyjadřuje společný exponent pro všechny kanály. Ten je získán z nejjasnějšího kanálu. Každá hodnota je potom zkrácena na 8-bitovou mantisu (frakční část). Výsledkem je komprese 6:1, a to na úkor snížené věrnosti barev. Na obrázku 2.6 je pak uvedeno rozdělení graficky.

LogLuv TIFF

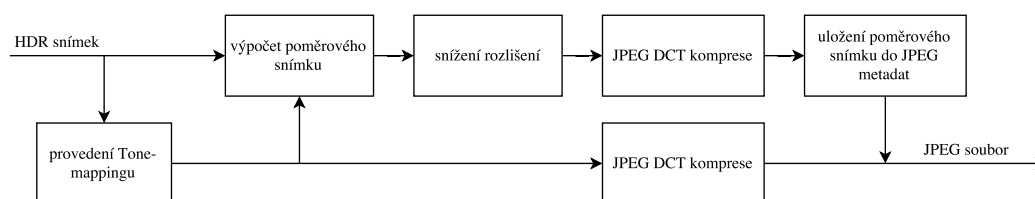
Hlavní nevýhodou reprezentace pixelů v plovoucí desetinné čárce je rychlost, jakou s nimi můžeme pracovat a pak obtížnost jejich komprimace. LogLuv (.tiff, .tif) pro zakódování vyžaduje pouze celá čísla. Tento způsob kódování využívá vlastností lidského oka, jelikož není stejně citlivé pro různé světelné podmínky. V temném prostředí dokážeme rozeznat jasové rozdíly v setinách cdm^{-2} , ve slunečném pak v desetinných cdm^{-2} . Jestliže je místo jasu samotného uvažován jeho logaritmus, nezáleží pak tak moc na detekovatelných prahových hodnotách, tudíž konstantní hodnoty mohou být konzervativní aproximací viditelného prahu. Pokud je tedy použit logaritmus jasu pro zakódování v celých číslech, vzniklé chyby jsou pak přípustné pro lidské oko. Pokud by se na zařízení pracovalo s tímto formátem, je třeba nejdříve převést snímek do LogLuv formátu.

JPEG-HDR

Tento ztrátový formát rozšiřuje známý JPEG (.jpg, .jpeg) [14] právě o možnost uchovat informaci pro zobrazení na HDR displeji. Jeho hlavní výhoda je ve zpětné kompatibilitě, se snímkem lze totiž pracovat s jako běžným JPEG souborem. Součástí souboru JPEG-HDR [3] jsou dva snímky. Jeden, na kterém bylo provedeno tónové mapování a druhý, na kterém je druhý snímek (mapa) s poměry jasu jednotlivých pixelů, které obsahují informace pro obnovení snímku do jeho HDR stavu. Druhý snímek je uchován v metadatech, proto jej



Obrázek 2.8: Vlevo snímek s provedeným tónovým mapováním (o něm více v podkapitole 2.3), vpravo mapa poměrů jasu, se kterou lze při kombinaci s prvním snímkem obnovit HDR vlastnosti.



Obrázek 2.9: Vývojový diagram procesu vytvoření snímku v JPEG-HDR formátu

běžný zobrazovací software nebere v úvahu a zobrazí se rovnou první snímek. Pro představu jsou snímky uvedeny na obrázku 2.8.

Vytvoření tohoto souboru probíhá nejprve provedením tónového mapování na HDR snímek, ten je pak zkomprimován do standardního 8-bit JPEG formátu. Tento nový snímek je posléze použit k vypočítání druhého snímku, který obsahuje poměry jasu pro obnovení do HDR stavu – na něj se pak opět aplikuje komprimace. Tento proces je nastíněn na vývojovém diagramu v obrázku 2.9.

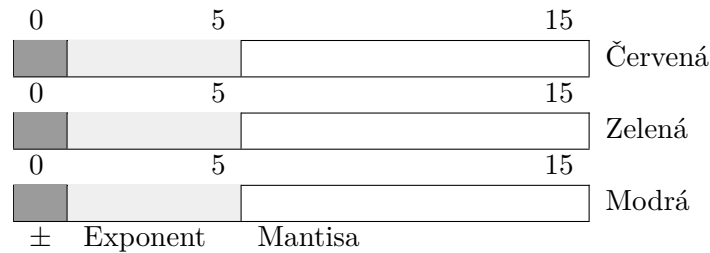
OpenEXR

Formát jenž má příponu .exr, je implementován v jazyce C++. Uvedla jej společnost *Industrial Light and Magic*⁴ a nyní je proklamován jako formátový standard pro průmysl se speciálními efekty – mnoho profesionálních programů tento formát už navíc podporuje.

Pixely mohou být uloženy v 16 a 32-bitové plovoucí desetinné čárce a nebo ve 4 bajtech pro celé číslo. Podporuje také bezztrátovou kompresi, přidávání atributů⁵ (bez ovlivnění zpětné kompatibility s existujícími aplikacemi) nebo možnost vložit další funkce pro rozší-

⁴<http://www.openexr.org/>

⁵Například řetězce, čísla, vektory a jiné.



Obrázek 2.10: 48-bitů použitých na jeden pixel formátu OpenEXR. Rozděleno do trojice dle barevných kanálů – první bit je znaménkový, dalších 5 je exponent a zbývajících 10 mantisa.

ření knihovny. Jak vypadá složení jednoho pixelu při použití 16 bitů v plovoucí desetinné čárce je uvedeno v obrázku 2.10.

2.3 Tone-mapping

Obecně se můžeme setkat se dvěma způsoby zobrazování, a to nativním vizualizérem nebo tónovým mapováním pro LDR displej. Nativní vizualizér je jedna z nejstarších metod, která byla pro práci s HDR využita, navíc u ní není třeba provádět úpravu HDR snímku. Potřebné je jen stínítko, které s různou intenzitou blokuje průnik světla skrz určité body. Výhodou této metody je kvalitní reprodukce původní scény.

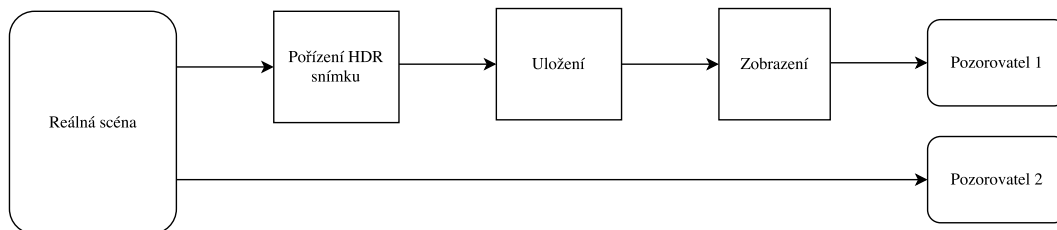
Aby se mohl HDR snímek zobrazit i na displeji s LDR, je zapotřebí ho tak i upravit. Tato podkapitola vychází z publikací [4] a [17]. Princip spočívá v přepočítání z širšího dynamického rozsahu do užšího, v tomto procesu je však ztracena část informace při snaze zachovat původní vzhled. Základní displeje LCD mají kontrastní poměr začínající na 200 : 1, podíváme-li se však na novější LCD, setkáváme se s poměrem začínajícím na 10,000 : 1. Běžně se diskretizuje na 8 nebo 10 bitů pro barevný kanál, a to znamená minimálně 255 odstínů – takto zobrazený snímek ale nedisponuje HDR, nýbrž LDR. Cílem tónového mapování je tedy zobrazit HDR snímek na LDR displeji s co nejlepším pokrytím původních detailů důležitých pro lidský zrak. Tento cíl navozuje obrázek 2.11. Pro bližší nastínění této záležitosti je uvedena funkce

$$f(I) : \mathbb{R}_i^{w \times h \times c} \rightarrow \mathbb{D}_0^{w \times h \times c}, \quad (2.9)$$

kde je I fotografie s HDR, w a h je šířka a výška fotografie I , c je počet barevných kanálů ve fotografii (běžně je $c = 3$, protože ve většině případů používáme barevný prostor RGB). Dále platí $\mathbb{R}_i \subseteq \mathbb{R}$, $\mathbb{D}_0 \subset \mathbb{R}_i$, $\mathbb{D}_0 = [0, 255]$ pro běžné LDR displeje.

Způsoby, jakými je možné Tone-mapping provádět, můžeme rozdělit do několika skupin. Hlavní rozdělení je na globální a lokální, ty jsou detailněji popsány v dalších bodech. Dalšími jsou frekvenční a segmentační, těmi se ale v této práci nebudeme zabývat. Pro úplnost však zmíním jejich vlastnosti a rozdíly. Segmentační provádí rozdělení obrazu do několika částí (segmentů), na které jsou posléze aplikovány různé metody tónového mapování. Frekvenční, nebo také gradientní, pak rozděluje oblastí s nízkou a vysokou frekvencí. Většinou se zde aplikuje mapování na nižší frekvence, vyšší pak zůstávají nezměněny.

Váhové funkce v tomto dělení ještě dělíme na empirické a perceptuální.



Obrázek 2.11: Srovnání procesu vytváření HDR snímku a jeho zobrazení Pozorovateli 1, a pozorováním reálné scény Pozorovatelem 2. Cílem této činnosti je zajistit, aby se oba pozorovatelé mohli mezi sebou střídát a nepoznali rozdíl.

2.3.1 Globální operátory

Mapping je aplikován na všechny pixely stejnou funkcí, přičemž je pracováno vždy s jedním bodem, a to bez ohledu na stavy ostatních. Tento přístup zpracování je nejrychlejší, a proto je také vhodný pro mobilní zařízení.

Operátor může z prvního průchodu snímku zvážit, jestli je možné redukcí dynamického rozsahu optimalizovat. Je tak vyvozeno ze statistik, které zahrnují maximální a minimální jas, logaritmické nebo aritmetické střední hodnoty.

Lineární operátor

Tato prostá metoda, která je jednoduchá na implementaci, je nejrychlejší způsob jak provést převod z HDRI do LDR snímku. Počáteční HDR snímek je vynásoben koeficientem e – tento krok je také nazýván normalizací, vzorec je

$$L_d(x, y) = eL_w(x, y). \quad (2.10)$$

Koeficient je získán z nejvyšší hodnoty jasu ve snímku rovnicí

$$e = \frac{1}{L_{w,max}}. \quad (2.11)$$

Protože se tato metoda řídí pouze jedním kritériem a navíc má lineární průběh, není se pak čemu divit, že důsledek normalizace může způsobit celkově tmavý výsledek LDR snímku. Koeficient je možné přizpůsobit naší scéně i jiným přístupem, avšak ani tak nedokáže tato metoda pokrýt širší dynamický rozsah – tedy je tato pro nás nepoužitelná.

Logaritmický operátor

V tomto případně je na snímek s HDR obsahem aplikována logaritmická funkce. Jejím základem je opět nejvyšší hodnota jasu ve snímku pro namapování nelineárního rozsahu $[0, 1]$. Operátor je dán

$$L_d(x, y) = \frac{\log_{10}(1 + qL_w(x, y))}{\log_{10}(1 + kL_{w,max})}, \quad (2.12)$$

kde $q \in [1, \text{inf})$ a $k \in [1, \text{inf})$ jsou konstanty, které když změníme, můžeme tak upravit vzhled výsledného snímku.

Exponenciální operátor

Na snímek je aplikována exponenciální funkce. Mapuje hodnoty z intervalu $[0, 1]$, kde je každá hodnota rozdělena aritmetickým průměrem. Operátor je dán

$$L_d(x, y) = 1 - \exp\left(-\frac{qL_w(x, y)}{kL_{w,H}}\right), \quad (2.13)$$

kde $q \in [1, \text{inf})$ a $k \in [1, \text{inf})$ jsou konstanty, které jsou stejně jak v logaritmickém mappingu voleny námi.

Reinhardův operátor

Předchozí uvedené operátory se řadí k těm jednodušším. Reinhardův operátor se řadí k těm složitějším, má ale lepší výsledky a je také proto více používán. Je uveden v knize *Photographic Tone Reproduction for Digital Images* [16].

Nejdříve je nutné vypočítat klíč scény, který vyjadřuje míru, jak moc je scéna světlá. Například pokud budeme mít snímek focený v noci jen s mírným osvětlením, bude tato hodnota nízká a naopak. Získání klíče scény je dáno vzorcem

$$\bar{L}_w = \frac{1}{N} \exp\left(\sum_{x,y} \log(\delta + L_w(x, y))\right). \quad (2.14)$$

N určuje celkový počet bodů ve snímku, $L_w(x, y)$ je světelnost bodu, který zohledňuje globální scénu a δ je malá hodnota zabraňující vzniku singularity – ta může vzniknout pokud se ve snímku vyskytnou absolutně černé body. Pokud je klíč scény nízký nebo vysoký, je možné scénu upravit dle našeho uvážení. Ve vzorci

$$L(x, y) = \frac{a}{\bar{L}_w} L_w(x, y) \quad (2.15)$$

je ukázáno, jak tuto hodnotu změnit. Hodnota a je volena z intervalu $[0, 1]$, typicky ale volíme z rozsahu hodnot 0.09–0.72. Problémem ve vzorci 2.15 je, že mnoho scén má normální dynamický rozsah, zároveň ale obsahuje i pár regionů s vysokým (nízkým) jasem. Tento problém byl u dřívějších fotoaparátů řešen kompresí, v moderních zařízeních se ale tato metoda nepoužívá. Reinhardův operátor proto používá vzorec

$$L_d(x, y) = \frac{L_w(x, y)}{1 + L_w(x, y)}. \quad (2.16)$$

Body s vysokým jasem se přibližují své inverzní hodnotě, zatímco body s nízkým jasem zůstávají téměř stejné. Jmenovatel ve vzorci zajišťuje hladký přechod v celém dynamickém rozsahu. Vzorec lze však dále rozšířit a jeho úpravou lze docílit zachování detailů ve světlých regionech.

$$L_d(x, y) = \frac{L_w(x, y) \left(1 + \frac{L_w(x, y)}{L_{\text{bílá}}^2}\right)}{1 + L_w(x, y)} \quad (2.17)$$

Ve vzorci 2.17 je použito spojení lineárního mapování a vzorce 2.16. Hodnotou $L_{\text{bílá}}$ je rozuměna nejvyšší hodnota jasu ve snímku – všechny vyšší hodnoty jsou pak bílé. Tímto jsou zachovány detaily ve tmavých regionech. Stále však není zaručeno, že se úplně všechny důležité detaily dostanou do výsledného snímku, a proto je pro tyto případy vhodné zvolit nějakou z metod lokálních operátorů.

2.3.2 Lokální operátory

U tohoto přístupu probíhá tónové mapování na každý pixel zvlášť, jakou hodnotu bude mít závisí na jeho okolí. Výhodou oproti globálním operátorům je jejich snaha reprodukovat lokální i globální kontrast. Okolní body však musí být vybrány opatrně, protože jinak hrozí halo efekt okolo hran. Někdy je ale přímo žádoucí tento halo efekt ponechat, zvláště tehdy, pokud chceme zdůraznit určitou oblast. Pro ukázkou, jak tyto operátory mohou fungovat, je dále uveden jeden ze zástupců.

Spatially Nonuniform Scalling

Oficiální název z publikace [7] lze volně přeložit jako *Nerovnoměrná úprava v prostoru*. Operátor je jeden z prvních, který se snažil o zachování lokálního kontrastu. Pro výpočet jasů bodu používá totiž jeho okolní body jako vstup. Lze uvést vzorec

$$L_d(x) = L_w(x)s(x), \quad (2.18)$$

kde $s(x)$ je funkce míry, ve které je použit lokální průměr z okolních bodů, definována je vzorcem

$$s(x) = (k(L_w \otimes G_\sigma)(x))^{-1}, \quad (2.19)$$

kde G_σ je Gaussovský filtr a k je konstanta, která upravuje finální výstup. Jeden z problémů tohoto operátoru je hodnota σ – pokud je příliš malá, výsledek je tmavý, pokud je vysoká tak se ve snímku objevuje halo efekt.

Kapitola 3

Specifikace zadání

Předcházející kapitola rozebírá problematiku dynamického rozsahu ve fotografii. Cíl práce a použité metody tedy budou specifikovány na základě poznatků a informací zde rozebranych. Cílem práce je navrhnout a vytvořit aplikaci pro mobilní zařízení, která bude moci pořizovat snímky s různou expozicí, následně pak z těchto snímků vytvořit fotografii s vysokým dynamickým rozsahem. Fotografie je pak zobrazena pomocí tónového mapování na LDR displeji. Pro vytvoření sekvence snímků je uvažována statická scéna.

3.1 Postup řešení

Aby skládání mohlo produkovat HDRI, je třeba pořídit minimálně 2 snímky s rozdílnou expozicí. Tato aplikace však bude fotografii sestavovat z minimálně 3 snímků, aby se mohlo pracovat s širším dynamickým rozsahem. První snímek bude podexponovaný, aby byla vidět všechna nejsvětlejší místa s co nejvyššími detaily, poslední snímek pak bude nadexponovaný, aby byla detailně vidět zase místa tmavá – snímky mezi nimi pak seřazené dle stoupající délky expozice. Po pořízení následuje skládání a vytvoření nového snímku. Aby se zabránilo vstoupení šumu do výsledku, bude na výběr z několika váhových funkcí, jako přednastavenou základní zvolím funkci *Debevec and Malik* (uvedena v 2.2.1). Aplikace si bude brát na důraz rychlost zpracování při práci s více snímky s nízkým dynamickým rozsahem najednou. Je si třeba ale uvědomit, že čím více snímků bude potřeba zpracovat, tím déle bude celý proces trvat. Bude-li to možné, využije se souběžně vícejádrový procesor zařízení pro maximální efektivitu.

Je-li snímek vytvořen, je na čase ho uložit do nějakého formátu. Původně jsem si vybral formát JPEG-HDR, protože poskytuje zpětnou kompatibilitu a výsledný snímek má malou velikost. Nicméně jsem narazil na problém s vlastnictvím tohoto formátu americkou společností *Dolby Laboratories*¹, která ač sice dovoluje vytvořit prohlížeč, nedovoluje bez předchozího licencování v tomto formátu ukládat. Poohlédnul jsem se po jiném řešení a zvolil formát OpenEXR (uveden v podkapitole 2.2.2).

Na zařízení však bude výsledný snímek i zobrazován, a proto bude implementováno i tónové mapování pro LDR displej. Jelikož je na výběr ze dvou hlavních skupin operátorů mapování a jde mi hlavně o rychlost, zvolím globální operátory. V aplikaci bude možné toto mapování vybrat, jako základní uvedu logaritmické.

¹<https://www.dolby.com/>

3.2 Implementace

Celé řešení bude implementováno pro operační systém Android² v jazycích Java³, XML⁴ a C++⁵. Jazyk Java je použit pro chod celé aplikace – je v ní řízen celý životní cyklus. Bude se zde řešit obsluha interakce ze strany uživatele přes grafické rozhraní a volání kódu v nativní formě. Jazyk C++ dovoluje v operačním systému Android vykonávat kód nativně, a proto je zvolen k vykonávání složitějších výpočtů. Jako poslední je jazyk XML, kterým je definován styl všech prvků, které uživatel vidí.

Použití knihoven

Aplikace bude používat knihovny, které zjednoduší práci při implementaci. Pro práci fotoaparátem zařízení bude použit balíček s API Camera2 [11], která umožňuje daleko pokročilejší možnosti pro práci se snímačem – je možné například nastavit délku expozice, volbu ISO, vyvážení bílé a jiné. Jako další knihovna je OpenCV⁶, ta umožňuje pokročilejší práci se snímkem fotografie, je to ale poměrně rozsáhlá knihovna s velkým množstvím funkcí a metod – pro moji aplikaci využiji jen zlomek této nabídky, mimo jiné ji použiji pro práci se souborovými formáty OpenEXR a JPEG.

Grafické rozhraní

Na obrázku 3.1 je uveden návrh hlavní obrazovky, ta bude zobrazena po spuštění a dovoluje uživateli obsluhu aplikace. Kliknutím na spoušť je pořízen snímek dle nastavených parametrů. Tlačítko nastavení zobrazuje menu s nastavením aplikace. Stisknutím tlačítka galerie asi tušíme, co bude následovat. Zobrazí se seznam pořízených snímků s možností jejich náhledu. Obrázek 3.2 uvádí návrh prvků obsažených v menu nastavení a jejich rozmístění.

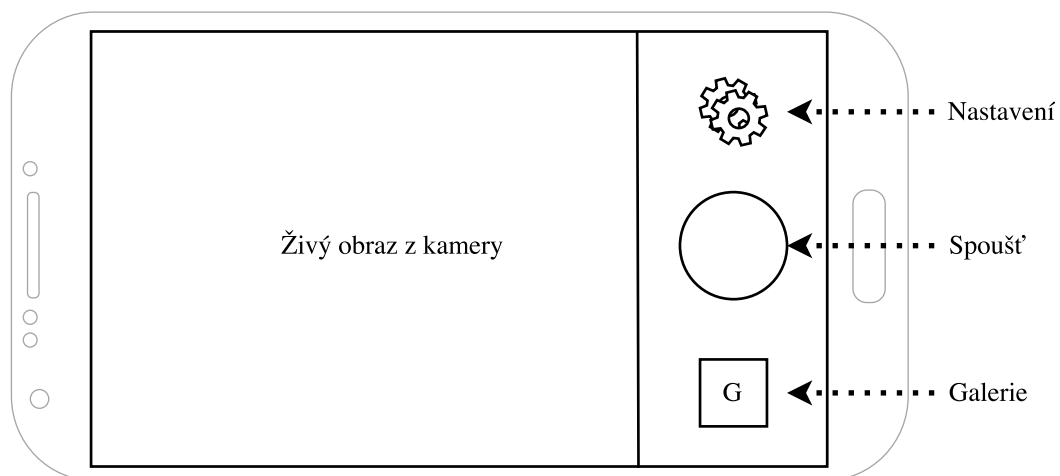
²<https://www.android.com/>

³<https://www.java.com>

⁴<https://www.w3schools.com/xml/>

⁵<http://www.cplusplus.com/>

⁶<http://opencv.org/>



Obrázek 3.1: Návrh uspořádání prvků na hlavní obrazovce. Na pozadí je neustále živě zobrazován obraz z kamery. Pravá část průhledně překrývá pozadí a obsahuje tlačítka, které tvoří základní prvky na ovládání.

Vytvoření HDR	
Počet snímků	3
Váhová funkce	Debevec&Malik
Tónové mapování	
Globální operátor	Logaritmický
Jiné	
Rozměrová velikost	1920x1080
Zobrazit po pořízení	<input checked="" type="checkbox"/> ON
Uložit jako	JPEG

Formát JPEG ukládá snímek s provedeným tónovým mapováním

Obrázek 3.2: Menu nastavení

Galerie	
	Název: Snímek_HDR_01032017_142925 Velikost: 13MB Formát: EXR
	Název: Snímek_HDR_01032017_142855 Velikost: 13MB Formát: EXR
	Název: Snímek_HDR_01032017_142115 Velikost: 2MB Formát: JPEG

Obrázek 3.3: Galerie

Kapitola 4

Realizace

Kapitola si klade za cíl seznámit čtenáře s praktickou částí práce. Na začátku jsou probrány jednotlivé vlastnosti zvolených nástrojů, jazyků a knihoven. Dále je popsána implementace grafického rozhraní a celé funkcionality aplikace.

Zvolený nástroj pro implementaci aplikace do operačního systému Android je *Android Studio*¹. Nabízí oficiální podporu pro tento systém a mimo pokročilého editoru kódu i veškeré nástroje k vývoji grafického rozhraní. Kód aplikace je napsán v jazyce Java – ten umožňuje objektově orientovaný přístup a usnadňuje tak orientaci v kódu. Pro definici stylu grafického rozhraní je použito XML – umožňuje snadný popis grafických prvků zobrazovaných v aplikaci.

Do řešení byla použita knihovna OpenCV, která je napsána v jazyce C++. Použití metod a funkcí v této knihovně je tedy vykonáváno nativně, což se dá označit jako další výhoda, protože přidává rychlosti zpracování.

Implementace aplikace byla provedena na operačním systému *Windows 10*² s využitím zařízení *NVIDIA SHIELD*³ se systémem Android verze 7.0⁴ pro testování dílčích verzí.

4.1 Grafické rozhraní

Je třeba si předem uvědomit, kdo bude aplikaci používat. Cíleno je jak na běžné uživatele, kteří toho o HDR fotografii moc neví, tak i pro uživatele, kteří o této metodě pořizování snímků mají přehled. Proto je v základu aplikace uloženo základní nastavení a zároveň použito jednoduché ovládání v hlavní aktivitě aplikace. Pokročilé úpravy snímání a zobrazování je skryto v menu *Nastavení*.

Hlavní menu

Po spuštění aplikace je zobrazeno hlavní okno, které nabízí základní ovládací prvky. Styl rozhraní je v souboru `activity_main`. Výsledek po aplikování tohoto stylu je vidět na obrázku 4.1. Vrchní část obsahuje tlačítko pro spuštění nové aktivity a zobrazení ovládacích prvků nastavení. Prostřední tlačítko započne sekvenci pro vytvoření HDR snímku. Během tohoto vytváření je uživateli zobrazeno okno o probíhajícím procesu (obrázek 4.2), které nelze zrušit a uživatel je donucen počkat na výsledek, tím je zabráněno nepovolenému

¹<https://developer.android.com/studio/index.html>

²<https://www.microsoft.com/>

³<https://www.nvidia.com/en-us/shield/>

⁴<https://www.android.com/versions/nougat-7-0/>



Obrázek 4.1: Rozmístění ovládacích prvků na displeji dominuje tlačítko spouště (uprostřed), záměrně nevýrazné je tlačítko pro zobrazení menu (nahore) a nakonec tlačítko pro otevření galerie (dole). Snímek byl pořízen na tabletu s velkou úhlopříčkou displeje – na jiném zařízení může grafické rozhraní vypadat jinak, například tlačítka jsou závislá na rozlišení displeje.

zacházení s aplikací. Hlavní menu je propojeno s aktivitou `MainMenu`. Veškerá interakce ze strany uživatele je tedy zpracována právě zde.

Nastavení

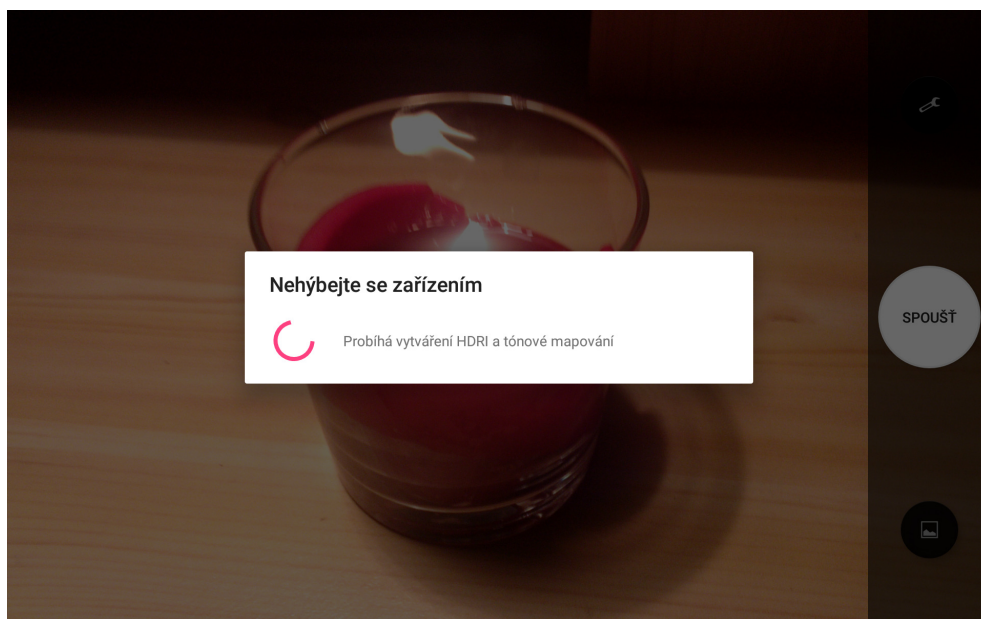
Rozšiřuje hlavní menu o dodatečné volby pro nastavení snímání a zpracování. Styl rozhraní je definován v souboru `activity_settings` a `content_settings`, výsledek kombinace těchto dvou stylů je vidět z části na obrázku 4.3. Uživatel si může vybrat jakou váhovou funkci použít, počet snímků pro vytvoření HDRI a další. Veškeré změny jsou ihned ukládány do statických proměnných třídy `CameraSettings`. Tytéž proměnné slouží k opětovnému nastavení stavů a voleb v grafickém rozhraní, a to pokud je spuštěna aktivita pro dodatečnou úpravu. Hodnoty jsou ale drženy jen po dobu běhu aplikace, po opětovném spuštění jsou hodnoty opět ve svém původním stavu.

4.2 Implementace

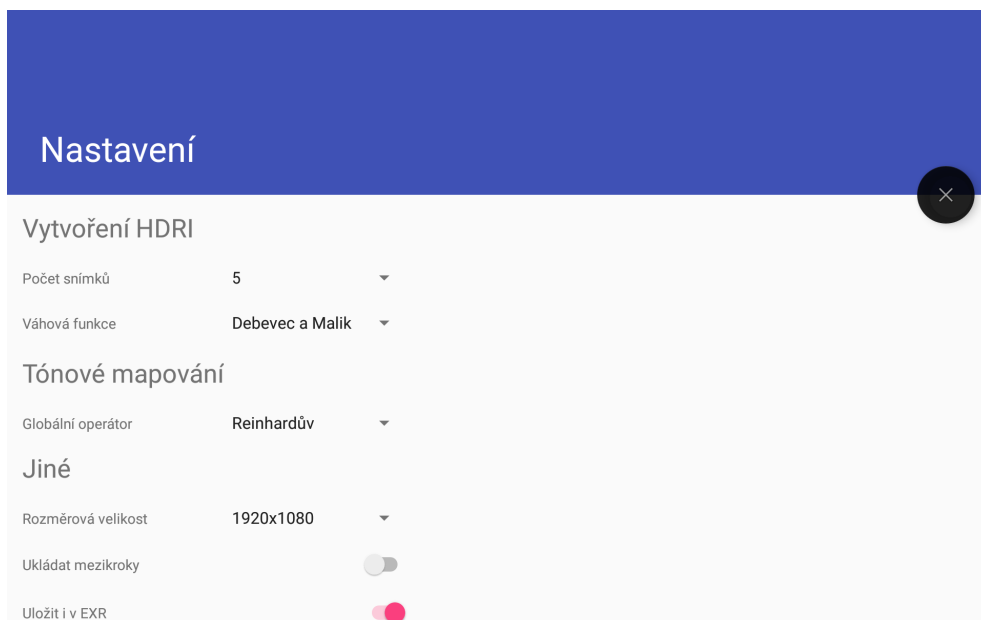
V této podkapitole je popsán životní cyklus aplikace. Nejprve je uvedena hlavní aktivita 4.2, kterou je řízen zbytek aplikace. Dále jsou uvedeny klíčové třídy, jež zahrnují metody a funkce pro další zpracování snímků.

Hlavní aktivita

Jedná se o vstup do celého programu a je spuštěna jako první. Je zastoupena třídou `MainActivity`, která provádí počáteční inicializaci. Při tomto procesu je zajištěno načtení



Obrázek 4.2: Okno zobrazené jako nejvyšší vrstva v aplikaci. Zabraňuje jakémukoliv ovládní před dokončením práce v pozadí. Jakmile je aplikace připravena znovu přijímat příkazy, okno zmizí.



Obrázek 4.3: Náhled do části menu v nastavení aplikace. Sekce jsou rozděleny do skupin a umožňují úpravu parametrů pro pořizování snímků, ukládání nebo zobrazení. V pravé části obrázku je umístěn černý křížek, který umožňuje uzavření tohoto menu.

knihovny OpenCV, dále povolení přístupových práv od uživatele do zařízení, inicializace kamery a plátna `TextureView` pro zobrazení živého obrazu.

Po dokončení inicializace je aktivita schopna reagovat na podněty z grafického rozhraní. Obsluha kliknutí na tlačítko *SPOUŠŤ*⁵ zavolá metodu `takePicture(View v)` a započne fáze pro pořizování sekvence snímků. Jednotlivé snímky mají před pořízením upravenou délku expozice a další parametry dle nastavení statických proměnných třídy `CameraSettings`. Je-li nastavena hodnota pro ukládání mezikroků, každý snímek je ihned uložen na úložiště zařízení. Snímky jsou akumulovány ve statické proměnné třídě `ImageBuffer` a jsou zde vkládány postupně. Jsou zde uloženy všechny snímky najednou – výhoda tohoto řešení je pak rychlost zpracování, protože se vše vyřizuje najednou.

Aktivita dále obstarává spuštění práce na pozadí při výpočtu HDRI, tónovém mapování, zobrazení okna průběhu (obrázek 4.2) nebo zajištěním spuštění aktivity náhledu `PreviewActivity` po vytvoření tónově namapovaného snímku.

Třída HDRI

Po pořízení sekvence snímků je volán objekt této třídy pro vygenerování snímku s vysokým dynamickým rozsahem. Konstruktore je předán počet snímků, pole s daty, délky expozic a rozměry. Na každý pixel snímků je aplikována váhová funkce, jejíž hodnoty jsou obsaženy ve statických proměnných třídě `WeightFunction`. Každá hodnota expozice je vyjádřena v nanosekundách, pro použití ve výpočtu (vzorec 2.5) je ale potřeba tuto hodnotu přepočítat na sekundy, proto jsou poděleny hodnotou 10^9 . Pro přístup k hodnotám v poli snímků je použito posouvání bitů a konjunktory pro získání bajtové hodnoty. Pixel je v barevném prostoru RGB (viz 2.1.2) s alfa kanálem tvořen celkem 32 bity. Prvních 8 bitů je vynecháno, protože právě zde je nepotřebný alfa kanál. Dalšími posuny po 8 bitech jsou postupně získány ostatní kanály. Dle již zmíněného vzorce je vypočítán a do paměti uložen HDR snímek.

Pokud uživatel přes grafické rozhraní nastavil ukládání snímku i v OpenEXR formátu (viz 2.2.2), je tak provedeno za použití knihovny OpenCV. Nejprve je vytvořen objekt ze třídy `org.opencv.core.Mat`, který obsahuje 3 kanály typu `CvType.CV_32FC1`. Jelikož OpenCV pracuje s obráceným uspořádáním kanálů BGR, je tato skutečnost zohledněna při ukládání. Následně jsou kanály spojeny dohromady a uloženy na zařízení s příponou `.exr`.

Třída ToneMapping

Instance třídy zpracovává snímek pro uložení do užšího dynamického rozsahu. V konstruktore třídy je spuštěno tónové mapování na statických proměnných uložených v třídě `HDRI`. Přístup je řešen přes statickou proměnnou, protože v jazyce Java nelze vkládat parametr do funkce nebo metody jako reference, vždy je hodnota zkopírována – to prodlužuje celý proces.

Metoda pro provedení tónového mapování je vybrána dle zvoleného nastavení uživatelem, které bylo uloženo ve statické proměnné třídě `CameraSettings`. Pro výpočet maximální hodnoty jasu je vytvořena metoda `getMax()`, která provede průchod HDR snímkem a tuto hodnotu vyhledá a uloží. Další pomocná funkce je pro vážený průměr `averageLw()` – pomocí pomocných proměnných se tento průměr vypočítá a následně uloží.

Každá hodnota je před uložením přepočítána pro hodnoty kanálu 0 – 255. To je provedeno vynásobením výsledné hodnoty pro kanál pixelu hodnotou 255. Po dokončení tónového

⁵viz obrázek 4.1

mapování je výsledný snímek připraven pro uložení v bitové mapě, je však provedena konverze do formátu JPEG.

Ovládání fotoaparátu

Jelikož se celý HDR snímek skládá z několika různě exponovaných fotografií, je nutné nějakým způsobem nastavit snímání. Toho bylo docíleno za pomoci API⁶ Camera2, které nabízí operační systém Android – není totiž možné mít přímý přístup ke snímači.

Nejdříve je vytvořen objekt `CameraService`, přes který je možné zjistit dostupné snímače. Je-li vše v pořádku, tak následuje ověření práv k přístupu a vytvoření žádosti k připojení. Pokud bylo vyhověno tomuto požadavku, je možné se ke kameře připojit a instance připojení je uložena v objektu `CameraDevice`.

Vytvoření jednoho nebo sekvence snímku začíná u nastavení parametrů snímání. Ty jsou aplikovány na objekt `CameraDevice` do funkce `createCaptureRequest(...)`. Aby byl snímek vytvořen, je ze stejného objektu využita funkce `capture(...)`, která zadá požadavek fotoaparátu. Až je snímek hotový, je zavolána naslouchací funkce `onImageAvailable(...)`, která snímek buď uloží do objektu instance třídy `ImageBuffer` nebo ho uloží na paměť zařízení.

⁶Application Programmable Interface

Kapitola 5

Měření a výsledky

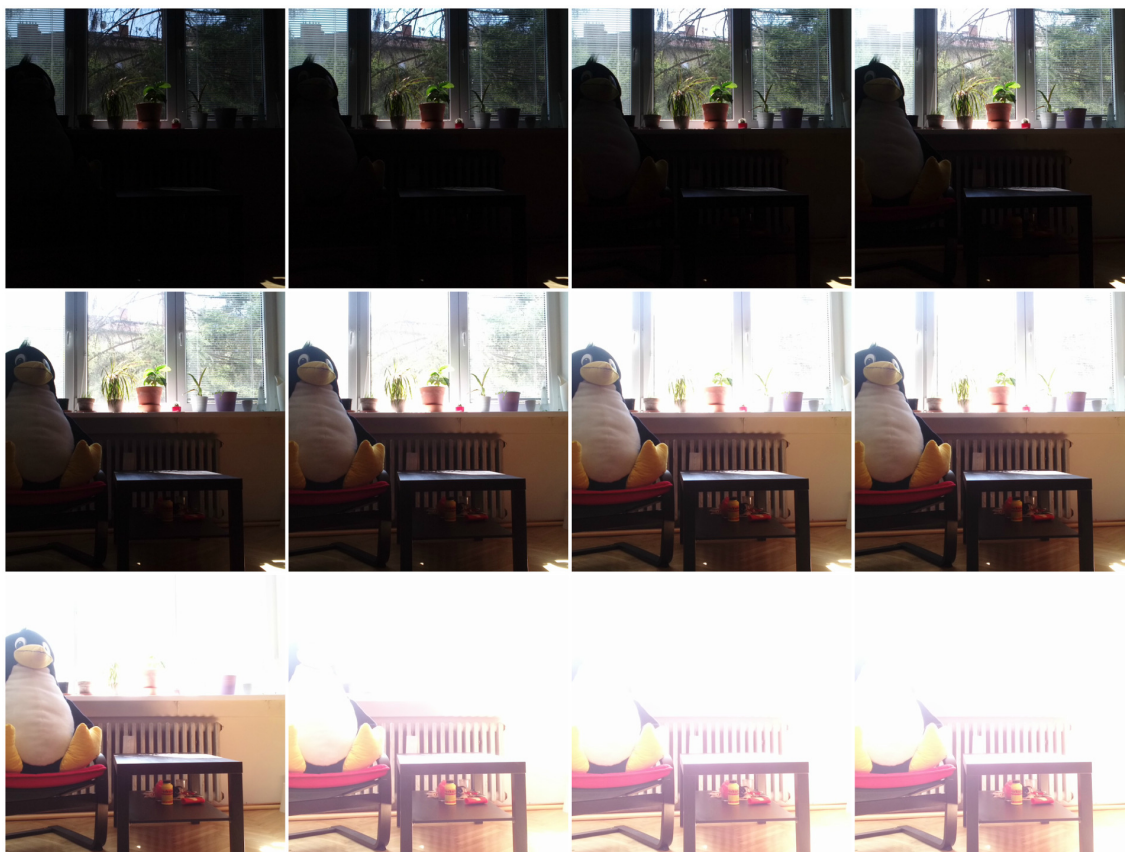
Tato kapitola prezentuje výsledky práce a rychlost zpracování jednotlivých částí implementovaných dle 3. Pro realizaci měření byl použit stativ kvůli zachování statičnosti scény, protože by jinak pořízené snímky nebyly vyrovnané a objevoval by se efekt ducha. Na stativ bylo připevněno zařízení pro snímání a následně vytvořeny jednotlivé snímky pro vytvoření HDRI, pak následovalo tónové mapování a uložení výsledků. Scéna byla zvolena tak, aby byl ve výsledku tónového mapování vidět detail tmavé a zároveň i světlé scény.

5.1 Rychlost zpracování

Aplikací bylo pořízeno celkem 131 HDR snímků pro vytvoření statistik. Po pořízení fotografií bylo použito zařízení uvedené v kapitole 4 – NVIDIA SHIELD. Pokud by byla aplikace spuštěna na jiném zařízení, je pravděpodobné, že se časy zpracování můžou mírně či více lišit. Jako parametry pro měření byla použita různá rozměrová velikost snímků a prezentované metody globálního tónového mapování v části práce 2.3.1. Čas začátku a konce měření byl získán pomocí systémové funkce `System.currentTimeMillis()`, která vrací aktuální čas v milisekundách. Rozdílem těchto dvou hodnot byla zaznamenána hodnota, která vyjadřuje délku trvání. Zprůměrované hodnoty z více měření jsou pak umístěny do tabulky 5.1. Zde je uvedena průměrná rychlost složení 13 LDR snímků do výsledného HDRI a dále průměrné rychlosti tónových mapování. Nejrychlejší, ale s očekávanou horší kvalitou výsledku, je lineární operátor naopak nejpomalejší je logaritmický operátor. Za povšimnutí stojí zmínit lineární vztah, který je tvořen mezi počtem pixelů ke zpracování a časem potřebným pro výpočet. Toto platí u všech operátorů, protože je zpracováván jen samostatný bod a není třeba provádět kontrolu okolních, jako je tomu u lokálních operátorů (více v 2.3.2).

	Test 1	Test 2	Test 3
Rozměrová velikost	1920x1080	1280x720	640x480
Složení LDR do HDRI	2168 ms	824 ms	332 ms
Lineární operátor	128 ms	49 ms	29 ms
Exponenciální operátor	975 ms	358 ms	166 ms
Logaritmický operátor	1022 ms	376 ms	140 ms
Reinhardův operátor	810 ms	299 ms	132 ms

Tabulka 5.1: Výsledky měření jednotlivých výpočetních částí aplikace. Pro skládání bylo použito 13 vstupních LDR snímků.



Obrázek 5.1: Vybraných 12 zdrojových snímků před složením do HDRI. Výsledný snímek byl použit pro práci s tónovým mapováním. Délka expozice se pohybovala od $\frac{1}{2000}$ s až do 2s.

5.2 Tónové mapování

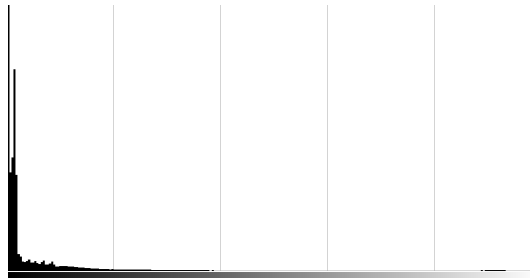
Každý z operátorů má svoje výhody a nevýhody – zde je uveden souhrn těchto vlastností a vyhodnocení výsledků snímání. Nejdříve byly vytvořeny jednotlivé snímky s různými expozicemi a následovalo vytvoření HDRI. Na tento snímek byla aplikována vždy jedna z metod tónového mapování. Jsou zde výsledky implementovaných globálních operátorů popsaných v podkapitole [2.3.1](#).

5.2.1 Lineární operátor

Tento operátor zde je uveden jen pro porovnání. Jelikož se jedná o nejjednodušší operátor s prostým výpočtem, rychlost zpracování byla rychlá, kvalita výsledku je však neuspokojivá, scéna obsahovala region s nadexponovanou částí a celý snímek se tak ponořil do tmavého nádechu. Výsledek procesu je vidět na obrázku [5.2](#).



Obrázek 5.2: Výsledek lineárního operátoru tónového mapování aplikovaného na HDR snímek.



Obrázek 5.3: Histogram jasu snímku 5.2. Levá strana reprezentuje četnost tmavých hodnot a pravá zase světlých.

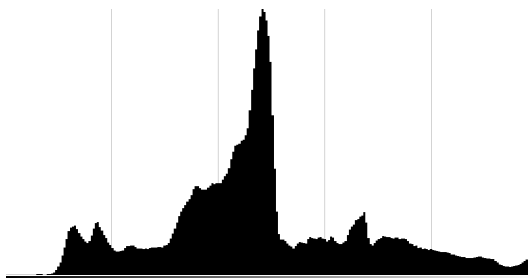
Aby byl výsledný snímek pěkný, je vhodné fotografovat scénu s nižším dynamickým rozsahem, a pak se zvláště vyvarovat vysokému jasu scény, protože právě ta určuje nejsvětlejší bod potřebný k výpočtu. Postup použitý pro tento výpočet je v podkapitole 2.3.1. Na obrázku histogramu jasu 5.3 je jasně vidět dominance tmavých hodnot.

5.2.2 Logaritmický operátor

Je to nejpomalejší operátor, má však daleko uspokojivější výsledky. Výhodou tohoto operátoru je možnost úpravy parametrů pro zpracování, nicméně je výsledek často takový nevýrazný a objevuje se dost šedi. Úpravou těchto parametrů je spíše měněn celkový jas, šed zůstává. Na obrázku 5.4 je vidět detail interiéru i exteriéru zároveň. Například je vidět střecha za oknem a hračka tučňáka v místnosti. Postup pro výpočet je uveden v podkapitole 2.3.1. Parametry byly nastaveny takto: $q = 10.0$ a $k = 0.005$. V histogramu jasu 5.5 je vidět podstatněji více hodnot než v předchozím lineárním.



Obrázek 5.4: Výsledek logaritmického operátoru tónového mapování aplikovaného na HDR snímek.



Obrázek 5.5: Histogram jasu snímku 5.4

5.2.3 Exponenciální operátor

O něco málo rychlejší jak logaritmický operátor. Při správném nastavení parametrů má uspokojivé výsledky. Na obrázku 5.6 je vidět interiér místnosti s vyšší světelností. Postup pro výpočet je uveden v podkapitole 2.3.1. Parametry byly nastaveny takto: $q = 0.5$ a $k = 1.0$. Jejich úpravou je nastavena světelnost scény, kontrast je spíše zachován. Historogram 5.7 ukazuje výsledek mého nastavení. Tmavé části jsou poměrně detailní, světlé mají efekt vysoké expozice.



Obrázek 5.6: Výsledek exponenciálního operátoru tónového mapování aplikovaného na HDR snímek.



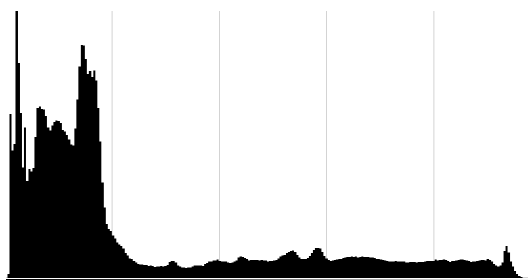
Obrázek 5.7: Historogram jasu snímku 5.6

5.2.4 Reinhardův operátor

Druhý nejrychlejší operátor z testovaných. Má nejlepší výsledek, co se týče barevné věrnosti a obsáhlosti detailů z různě jasných částí. Vytvoření tohoto snímku je popsáno v podkapitole 2.3.1. Klíč scény byl nastaven na hodnotu $a = 0.18$. Jeho úpravou lze nastavit jas scény podobně jako v exponenciálním operátoru. Pro výpočet je z části použit vzorec pro exponenciální operátor, je ale upravena výsledek je daleko lepší.



Obrázek 5.8: Výsledek Reinhardova operátoru tónového mapování aplikovaného na HDR snímek.



Obrázek 5.9: Histogram jasu snímku 5.8

Kapitola 6

Závěr

Tato práce si kladla za cíl popsat problematiku fotografie s vysokým dynamickým rozsahem. Na základně popsané problematiky byla vytvořena aplikace v jazyce Java, která dokáže vytvořit sekvenci snímků fotoaparátem zařízení s různou délkou expozice. Ze sekvence statických snímků je složena výsledná fotografie s vysokým dynamickým rozsahem, který je následně zobrazen na displeji zařízení za pomoci předem zvoleného operátoru tónového mapování. Aplikace používá knihovnu OpenCV napsanou v jazyce C++, k ukládání snímků ve formátu OpenEXR. Dále je použito API Camera2 dostupné v operačním systému Android pro pokročilou práci s fotoaparátem zařízení. Vzhled celé aplikace je popsán jazykem XML a byl vytvořen za pomoci designéru v programu Android Studio.

V aplikaci jsou navíc implementovány různé druhy tónového mapování, u kterých je možné upravovat jejich parametry v přehledném grafickém rozhraní. Stejně tak i parametry výsledné fotografie lze měnit. Výsledek zpracování snímků lze navíc uložit ve formátu JPEG a OpenEXR pro pozdější práci s libovolným nástrojem, který tento formát podporuje.

Vytvořená aplikace je poměrně rychlá a časová odezva pro běžného uživatele je dostatečná. Pouze v případě ukládání ve formátu OpenEXR je proces delší.

Další pokračování práce je možné zaměřit například na rozšíření základny dostupných operátorů tónového mapování, jak pro globální tak i pro lokální, aby se z aplikace mohla stát profesionální pomůcka pro fotografy. Zařízení, které tuto aplikaci mohou využívat, totiž mnohdy nedisponují profesionálním snímačem nebo optikou, takže je zatím vyhodnocena jako pomůcka. Dále by se při náhledu snímku po provedeném tónovém mapování mohla přidat možnost úpravy parametrů, dle kterých bylo tónové mapování provedeno, například formou jezdců nebo tlačítek přímo na displeji bez nutnosti znovu snímek pořídit. Také je možné vytvořit pro výpočetně náročnější části implementaci v nativním kódu využívající více jádrové procesory, která by ve výsledku mohla proces vytváření snížit o desítky milisekund, avšak by stále neovlivnila nutnost pořizovat snímky s delší expozicí, takže by urychlení aplikace nebylo tak dramatické. Zajímavé by také byla podpora záznamu videa, i když je tento proces časově náročný. Snížením frekvence snímků a upravením rozměrové velikosti jednotlivých pro výsledný záznam by se tohoto cíle mohlo dosáhnout. Se stále se zvyšující rychlostí bezdrátových sítí by také bylo možné aplikaci upravit pro posílání nezpracovaných snímků na vzdálený server, kde by se již představa videa mohla naplnit, a tak proměnit poměrně levné zařízení v pokročilý hardware.

Literatura

- [1] CIE 1931 Color Space | Colblindor. Colblindor | All about Color Blindness [online]. [cit. 06.05.2017].
URL <http://www.color-blindness.com/2007/01/18/cie-1931-color-space/>
- [2] Colorimetry - Systems CIELAB and CIELUV. Optique pour l'ingénieur [online]. [cit. 06.05.2017].
URL http://www.optique-ingenieur.org/en/courses/OPI_ang_M07_C02/co/Contenu_08.html
- [3] JPEG-HDR: A Backwards-Compatible, High Dynamic Range Extension to JPEG [online]. [cit. 28.04.2017].
URL <http://www.anyhere.com/gward/papers/cic05.pdf>
- [4] Banterle, F.; aj.: *Advanced high dynamic range imaging: Theory and Practice*. A K Peters, Ltd., 2011, iISBN: 978-1-56881-719-4.
- [5] Bočík, A.: *Velká kniha HDR fotografie: kouzlo fotografií s vysokým dynamickým rozsahem. 2., aktualiz. vyd.* Brno: Computer Press, 2011, přeložila Hana MAJEROVÁ, přeložil Michal BRŮHA. ISBN 978-80-251-2210-5.
- [6] Bourke, P.: Colour spaces. [cit. 01.05.2017].
URL http://paulbourke.net/texture_colour/colourspace/
- [7] Chiu, K.; aj.: *Spatially Nonuniform Scaling Functions for High Contrast Images*. [online]. [cit. 25.04.2017] Morgan Kaufmann Publishers Inc., Květen 1993, 245–244 s.
URL <http://sol.cs.trinity.edu/~jhowland/class.files.cs357.html/gi93.pdf>
- [8] Dannhoferová, J.: *Velká kniha barev: kompletní průvodce pro grafiky, fotografie a designéry*. Brno: Computer Press, 2012, iISBN 978-80-251-3785-7.
- [9] Debevec, P.: Recovering High Dynamic Range Radiance Maps from Photographs [online]. [cit. 22.03.2017].
URL <http://www.pauldebevec.com/Research/HDR/>
- [10] Donnelly, J.; Massa, N.: *Light: introduction to optics and photonics*. MA: New England Board of Higher Education, 2010, iISBN 978-0-981-53180-9.
- [11] van Drongelen, M.: *Android Studio Cookbook*. Packt Publishing Ltd., 2015, iISBN 978-1-78528-618-6.
- [12] Hoefflinger, B.: *High-Dynamic-Range (HDR) Vision: Microelectronics, Image Processing*. Springer Berlin Heidelberg, 2007, iISBN 978-3-540-44432-9.

- [13] Myška, M.: *333 tipů a triků pro digitální fotografie*. Brno: Computer Press, 2009, iISBN 978-80-251-2210-5.
- [14] Pennebaker, W. B.; Mitchell, J. L.: *JPEG: Still Image Data Compression Standard*. Springer Science and Business Media, 1992, iISBN 0442012721, 9780442012724.
- [15] Peterson, B.: *Understanding digital photography : Techniques for Getting Great Pictures*. Watson-Guption Publications, 2005, iISBN 0-8174-3796-7.
- [16] Reinhard, E.; Stark, M.; Shirley, P.; aj.: *Photographic Tone Reproduction for Digital Images*. SIGGRAPH, 2002, iISBN 1-58113-521-1.
- [17] Reinhard, E.; aj.: *High dynamic range imaging: acquisition, display, and image-based lighting*. Elsevier Inc., 2010, iISBN: 978-0-12-374914-7.
- [18] Rodney, A.: *Color management for photographers: hands on techniques for Photoshop users*. MA: Focal Press, 2015, iISBN 978-0-240-80649-5.

Příloha A

Obsah DVD

Součástí této práce je i přiložené DVD, které obsahuje následující složky a soubory:

- Android/ - zdrojové kódy a knihovny implementované aplikace
- Build/ - přeložená aplikace pro instalaci
- HDR/ - vytvořené HDR snímky v OpenEXR formátu
- LDR source/ - ukázka zdrojových snímků, ze kterých aplikace počítá výsledný HDRI
- Text/ - zdrojové soubory textu této práce
- ToneMapping/ - snímky vytvořené tónovým mapováním
- README.txt - krátký návod pro instalaci aplikace a spuštění v programu Android Studio