

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMEDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

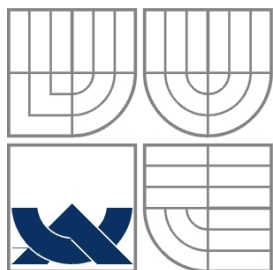
ZPRACOVÁNÍ DIGITÁLNÍ FOTOGRAFIE

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

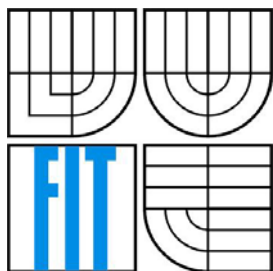
AUTOR PRÁCE
AUTHOR

Bc. VÍT ZDRAŽIL

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMEDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ZPRACOVÁNÍ DIGITÁLNÍ FOTOGRAFIE

PROCESSING OF DIGITAL PHOTOGRAPHY

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. VÍT ZDRAŽIL

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. IGOR POTÚČEK, Ph.D.

BRNO 2007

Zadání diplomové práce

Řešitel: **Zdražil Vít, Bc.**

Obor: Počítačová grafika a multimédia

Téma: **Zpracování digitální fotografie**

Kategorie: Počítačová grafika

Pokyny:

1. Prostudujte metody interpolace obrazu, vyvážení bílé barvy a další techniky určené pro zpracování digitální fotografie.
2. Navrhněte metodu pro převod RAW dat získaných z digitálních fotoaparátů do obrazových souborů ve vybraných formátech (JPG, TIFF, BMP....).
3. Při převodu by měly být využity i techniky pro předzpracování obrazu, jako je vyvážení bílé barvy, odstranění šumu, změna rozlišení a jiné.
4. Tyto techniky implementujte a vytvořte knihovnu funkcí s obecným rozhraním pro další použití.
5. Funkčnost a kvalitu implementovaných metod ověřte na reálných datech.
6. Zhodnoďte dosažené výsledky a navrhněte další možná vylepšení.

Literatura:

- Žára, J., Beneš, B., Felker, P.: Moderní počítačová grafika, Computer Press, 1998.
- Hlaváč, V.: Zpracování signálů a obrazů, Praha, ČVUT, 2005.
- Rafael C. Gonzalez, Richard E. Woods: Digital image processing, Prentice-Hall, 2002.
- Galbiati, L. J.: Machine vision and digital image processing fundamental, Prentice-Hall, 1990.
- ostatní na požádání u školitele

Při obhajobě semestrální části diplomového projektu je požadováno:

- Splnění prvních 3 bodů zadání.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% 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 paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Potůček Igor, Ing.**, UPGM FIT VUT

Datum zadání: 28. února 2006

Datum odevzdání: 22. května 2007

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



doc. Dr. Ing. Pavel Zemčík
vedoucí ústavu

Licenční smlouva

Čestné prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Igora Potůčka, Ph.D.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Jméno Příjmení

Datum

Poděkování

Na tomto místě bych rád poděkoval vedoucímu práce Ing. Igoru Potúčkovi, Ph.D. za odborné vedení a ochotu při řešení dílčích problémů. Dále bych rád poděkoval Ing. Tomáši Syrovému, Ph.D. za podnětné nápady a pomoc při testování.

Abstrakt

Tato práce se zabývá zpracováním tzv. RAW dat z digitálního fotoaparátu. V první části je popsán princip snímání obrazu digitálním fotoaparátem, obecný způsob zpracování obrazu v přístroji a co je to RAW formát, jeho výhody a nevýhody. Dále jsou popsány některé existující interpolační metody obrazových dat a další metody pro zpracování obrazu z RAW obrazových dat. Následně je analyzován RAW formát CR2 firmy Canon, včetně jeho struktury a popisu pro jeho zpracování. V další kapitole byla navržena vylepšená metoda pro interpolaci obrazu a také metoda pro účinné potlačení digitálního šumu. Na tomto základě byla implementována knihovna pro dekodování a základní zpracování souborů CR2, pomocí které byly porovnány výsledky zpracování obrazu vybranými metodami. V závěru je provedeno zhodnocení dosažených výsledků a také je nastíněn směr, pro další pokračování výzkumu dané problematiky.

Klíčová slova

RAW, CR2, digitální fotografie, zpracování, převod, konverze, Bayer, barevná interpolace, LS-JPEG, vyvážení bílé barvy, potlačení šumu

Abstract

This document focuses on processing of RAW image data from digital cameras. In first section is described principle of image capturing by digital camera, common way of image processing in the device and what the RAW format is, its pros and cons. In the next section is described existing demosaicing methods, methods for additional processing of RAW image data and there is analyzed specific RAW format, Canon's CR2, including structure and guide for its conversion. Next sections contains proposal of the new improved demosaicing method and method for suppressing digital noise. On this basis a library for basic CR2 files processing was implemented. There is comparison of methods for processing RAW image data with this library in the next section. In the conclusion there is summary of finished work and there is also mentioned outline of future work.

Key Words

RAW, CR2, digital photography, processing, conversion, Bayer, demosaicing, LS-JPEG, white balance, removing noise

Obsah

1	Úvod.....	11
2	Teorie	12
2.1.	Obrazové senzory v digitálních fotoaparátech	12
2.1.1.	Senzor CCD.....	12
2.1.2.	Senzor CMOS.....	12
2.1.3.	Barevné senzory s Bayerovou mřížkou.....	13
2.1.4.	Senzor FOVEON.....	14
2.2.	Zpracování obrazu v digitálním fotoaparátu	16
2.2.1.	RAW formát	16
2.3.	Interpolace barevného obrazu.....	17
2.3.1.	Bilineární metoda interpolace.....	19
2.3.2.	Pokročilá lineární metoda interpolace	20
2.3.3.	Metoda VNG (Threshold-based variable number of gradients).....	21
2.3.4.	Metoda Pixel Grouping	24
2.3.5.	Metoda AHD (Adaptive Homogeneity-Directed)	26
2.4.	Kalibrace barev.....	29
2.5.	Vyvážení bílé barvy.....	31
2.6.	Odstranění šumu.....	32
2.7.	Převody mezi barevnými modely.....	34
2.7.1.	Převod RGB na XYZ	34
2.7.2.	Převod XYZ na Lab	35
2.7.3.	Převod Lab na XYZ	36
2.7.4.	Převod XYZ na RGB	36
2.8.	Závěrečné úpravy obrazu	37
2.9.	Popis formátu CR2	37
2.9.1.	LS-JPEG.....	38
3	Řešení	39
3.1.	Navrhovaná adaptivní metoda interpolace	40
3.2.	Navrhovaná metoda potlačení šumu v digitální fotografii	41
3.3.	Implementace	42
3.3.1.	Knihovna cr2decode.....	42
3.3.2.	Práce se soubory CR2 pomocí knihovny cr2decode	43
3.3.3.	Struktura knihovny cr2decode.....	43
4	Vyhodnocení	44

4.1.	Linearizace jasových hodnot	44
4.2.	Interpolace	44
4.3.	Míra šumu.....	52
4.4.	Časová náročnost operací	53
5	Závěr.....	55
Literatura		56
Přílohy		57
Formát CR2		57
Hlavička souboru CR2		57
IFD		57
Záznam IFD.....		57
LS-JPEG		59
Knihovna cr2decode.....		60

1 Úvod

Digitální fotografie prodělala v posledních několika letech veliký rozmach. Jedním z lákadel pro uživatele, je možnost okamžitě vidět exponovaný snímek na displeji přístroje. Aby to bylo možné, musí přístroj provést několik kroků zpracování vstupního signálu do výsledného obrazu.

Pokročilejší přístroje nabízí i speciální režim ukládání pořízených snímků - tzv. surová (RAW) data, někdy též označovaná jako digitální negativ. Zde nedochází ke zpracování samotným digitálním fotoaparátem, ale uživatel má možnost snímky zpracovat dodatečně na PC a v lepší kvalitě, než by mohl dosáhnout jen pomocí digitálního fotoaparátu.

Hlavním cílem této práce je vylepšení existujících metod pro zpracování RAW obrazových dat a jejich porovnání s již existujícími metodami.

Teoretická část obsahuje popis procesu získání obrazových dat digitálním fotoaparátem a dále princip zpracování těchto obrazových dat. Dále jsou popsány některé současné interpolační metody, na základě jejichž praktických výsledků tato práce staví. Poté je analyzován formát RAW souborů CR2 firmy Canon, které budou částečně sloužit jako vstupní data pro zpracování obrazu v této práci.

V následující části je popsána navrhovaná vylepšená metoda interpolace obrazových dat a metoda pro potlačení typického šumu, který vzniká v digitální fotografii.

Předposlední část obsahuje zhodnocení dosažených výsledků. Výsledná data byla naměřena jak na reálných scénách, tak i na matematicky vytvořených testovacích datech. Protože výsledný obraz z digitálního fotoaparátu hodnotí především člověk, největší důraz je kladen na subjektivní hodnocení dosažených výsledků. Některé stěží vyčíslitelné chyby v obraze působí pro lidské oko velmi rušivě a naopak statisticky velkou chybu v obraze může člověk stěží postřehnout.

V závěrečné kapitole je zhodnocení této práce a nástin směru pro budoucí možné pokračování.

2 Teorie

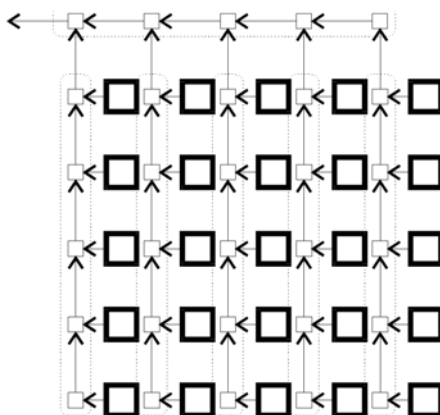
V této části budou rozebrány jednotlivé kroky od zachycení obrazu senzorem digitálního fotoaparátu až po vytvoření obrazového souboru v běžném formátu (BMP, TIFF, JPEG apod.). Hlavní důraz bude kladen na jednotlivé interpolační metody obrazových dat maskovaných tzv. Bayerovou mřížkou.

2.1. *Obrazové senzory v digitálních fotoaparátech*

Obrazový senzor je zařízení, které převádí optický obraz na elektrický signál. Je to obvykle pole CCD nebo CMOS senzorů.

2.1.1. Senzor CCD

Název CCD je zkratkou z anglického výrazu *Charge-Coupled Device* [16]. Jedná se o pole vzájemně propojených (coupled) fotocitlivých kondenzátorů, z kterých signál získává pomocí řízeného přesunu náboje. Přestože se dnes místo kondenzátorů používají fotocitlivé diody, kapacitní pohled se používá dál. Lépe to odpovídá historickému vývoji i funkčnímu principu.



Obrázek 1: Struktura CCD senzoru

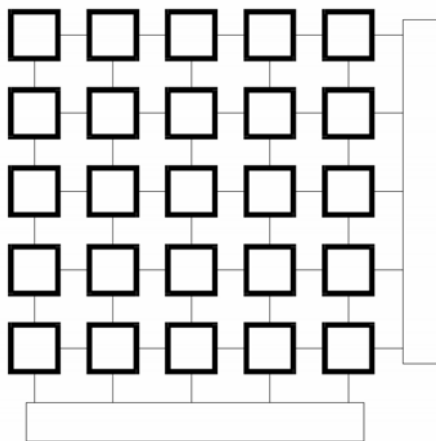
Po expozici CCD čipu způsobí řídicí obvod, že každý kondenzátor předá svůj náboj svému sousedovi. Poslední kondenzátor v poli předá náboj do analogového zesilovače, který převede náboj na napětí. Opakováním tohoto procesu řídicí obvod převede celý obsah CCD čipu na proměnné napětí (analogový signál), který je vzorkován, digitalizován a dále zpracováván.

2.1.2. Senzor CMOS

Zkratka CMOS [15] pochází z anglického výrazu *Complementary metal–oxide–semiconductor*. Jsou to polovodičové čipy vyráběné současnou standardní výrobní technologií. CMOS technologie umožňuje větší integraci logických funkcí na čipu.

Dvěma klíčovými vlastnostmi CMOS obrazových senzorů jsou vysoká odolnost proti šumu a nízký statický proudový odběr. Díky nižší spotřebě se CMOS čipy tolik nezahřívají (v porovnání s CCD).

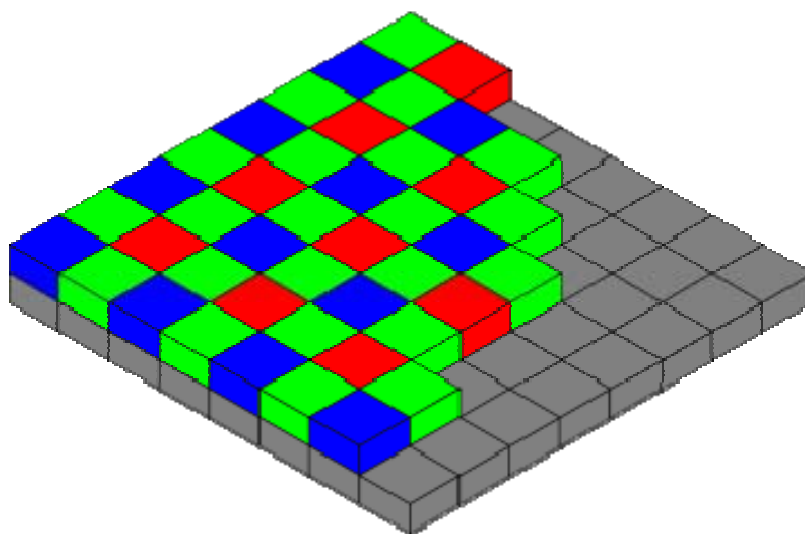
CMOS obrazové senzory umožňují adresovat každý pixel zvlášť.



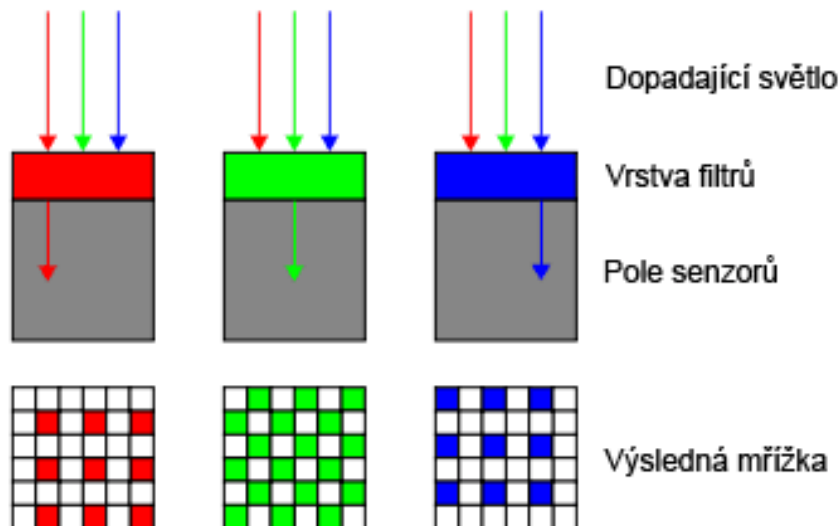
Obrázek 2: Struktura CMOS senzoru

2.1.3. Barevné senzory s Bayerovou mřížkou

Jednotlivé senzory reagují pouze na jasovou složku, k získání barevného obrazu si musíme dopomoci jiným způsobem. Nejčastěji se v digitálních fotoaparátech používá tzv. Bayerova mřížka [10] a [11]. Je to nejlevnější a nejrozšířenější řešení. Využívá se zde Bayerova filtru, který propouští červené, zelené a modré světlo k vybraným pixelům uspořádaným v prokládané mřížce. Výsledný plnobarevný obraz je pak z jednotlivých pixelů interpolován. Kvalita výsledného obrazu se liší podle použitých algoritmů.



Obrázek 3: Bayerovo uspořádání barevných filtrů



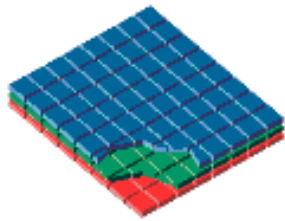
Obrázek 4: Mřížka filtrů jednotlivých základních barev

Barevný filtr se jmenuje podle svého vynálezce Dr. Bryce E. Bayera. Ten nazývá zelené pixely elementy citlivými na jas a červené a modré pixely pak elementy citlivými na barevný rozdíl (chrominanci).

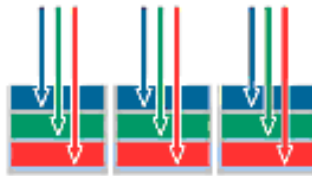
2.1.4. Senzor FOVEON

Začátkem roku 2002 vyvinula firma Foveon, Inc. čip, který umožňuje zachytit všechny tři základní barvy v jednom pixelu. Je toho dosaženo propustností jednotlivých vrstev na čipu pro různé barevné složky a jejich následné snímání v různých hloubkách. Je to poměrně nová technologie, její největší výhodou je plnobarevná informace v každém pixelu, odpadá tedy interpolace barev z Bayerovy mřížky. Dále tento čip poskytuje ostřejší obraz a měl by mít bohatší a sytější podání barev.

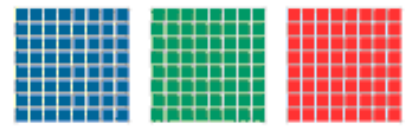
Foveon X3® Capture



Foveon X3 přímý obrazový senzor má 3 nezávislé vrstvy senzorů.

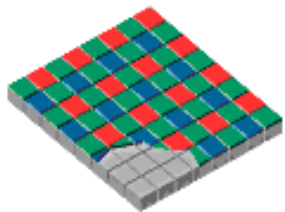


Protože křemík absorbuje různé vlnové délky světla v různých hloubkách, každá vrstva zachytává jinou barvu. Protože jsou vrstvy nad sebou, jeden pixel zachytí všechny barvy.

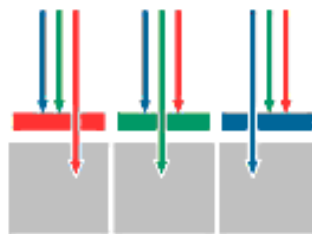


Pouze Foveon X3 zachytí červenou, zelenou a modrou barvu v každém pixelu snímače.

Mosaic Capture



V běžných systémech jsou barevné filtry použity v jedné vrstvě, v mřížkovém uspořádání.



Filtry propustí pouze jednu vlnovou délku světla. Každý pixel tedy zachytí jen jednu barvu.



Výsledkem je, že běžný senzor zachytí pouze 25% červeného a modrého světla a 50% zeleného.

Obrázek 5: Porovnání senzoru Foveon s běžným typem

Bohužel, tento čip má také své nevýhody. Přestože se nedá rozlišení čipu FOVEON s čipy CCD a CMOS přímo srovnávat, přece jenom FOVEON v rozlišení nestačí na dnešní senzory typu CCD či CMOS. Dále se tato technologie potýká s výrazným ořezáváním signálu vždy ve všech třech barevných složkách najednou a s nedostatečným odstupem signálu od šumu, opět v porovnání s dnešními moderními senzory konkurenčních technologií.

2.2. Zpracování obrazu v digitálním fotoaparátu

Pro vytvoření výsledného obrazu musí být výstupní signál z obrazového senzoru zpracován [10]. Jedná se o několik kroků, které vidíme na následujícím obrázku.



Obrázek 6: Řetězec zpracování obrazu digitálního fotoaparátu

Nejdříve je získán digitální signál ze senzoru (odpovídá to červené oblasti na obrázku). Ten mívá často větší než standardní 8 bitovou barevnou hloubku. Zpravidla je to 12 bitů na kanál, někdy 14 a u profesionálních (středofarmátových a větších) přístrojů i 16 bitů na kanál. Tento jen minimálně zpracovaný obraz přímo ze senzoru nazýváme surovými daty neboli RAW daty. Použití RAW formátu nám umožňuje všechny ostatní úpravy obrazu provést až následně na PC a to bez ztráty kvality.

Pokud chceme dostávat už přímo z digitálního fotoaparátu použitelné (hotové) obrázky, nejčastěji ve formátu JPEG, musí proběhnout další zpracování. Převod na 8 bitovou barevnou hloubku, Bayerova interpolace, vyvážení bílé barvy, odstranění šumu, úprava barev (kontrast, sytost), doostření a nakonec ještě samotná komprese do JPEG formátu. Těmto jednotlivým činnostem se budou věnovat další podkapitoly.

2.2.1. RAW formát

RAW formátem [14] označujeme digitalizovaná data přímo získaná ze snímacího čipu, společně s dalšími nutnými informacemi jako je hlavička souboru, malý náhledový obrázek a EXIF informace [8] (informace o expozici, ale i další informace jako např. jméno autora, název modelu přístroje atd.).

RAW formát není nijak standardizován. Každý výrobce používá vlastní způsob ukládání surových dat a tento není zpravidla nijak dokumentován.

Výhody:

- RAW má větší dynamický rozsah než JPEG, a tak je po expozici a až při převodu RAW v PC možné dělat "expoziční kompenzaci" ve velkém rozsahu. Z RAWu tak lze zachránit i poměrně dost podexponovaný nebo přeexponovaný snímek.
- Řadu nastavení důležitých parametrů (vyvážení bílé, kontrast snímku atd.) je možné rozhodnout a nastavit až v PC za pomoci myši, klávesnice a kvalitního monitoru.
- Z jednoho souboru RAW je možné udělat několik variant jednoho snímku a všechny budou maximálně kvalitní.
- RAW pracuje obvykle s 12 bitovou barevnou hloubkou, což je výrazně více než 8 bitový JPEG. Všechny úpravy kontrastu, jasu, barev atp. jsou tedy mnohem méně destruktivní pro kvalitu snímku.

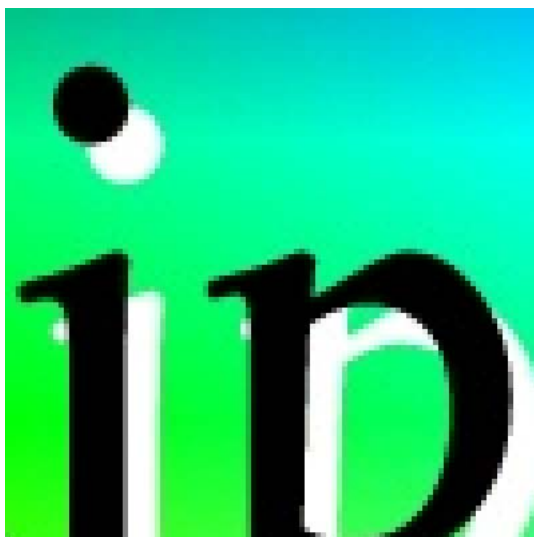
Nevýhody snímání do RAW jsou zhruba tyto:

- RAW produkuje větší soubory než JPEG, což snižuje počet fotografií, které se vejdou na paměťové médium.
- Větší soubory zpomalují zápis na kartu i všechny další manipulace, takže jak fotografování (zejména ukládání fotografií na kartu) tak i prohlížení a kopírování do PC bude pomalejší.
- Pro získání použitelných fotografií je třeba nějaký čas strávený u PC při převodu. Jeden RAW se převádí podle výkonu PC a rozlišení fotoaparátu kolem 10 až 30 vteřin.
- Nelze ihned předat nafocená data, RAW vyžaduje zpracování na PC. Fotografie v RAW formátu se i hůře prohlíží, protože k prohlédnutí je třeba fotografii alespoň pomocí rychlého a méně kvalitního algoritmu převést.

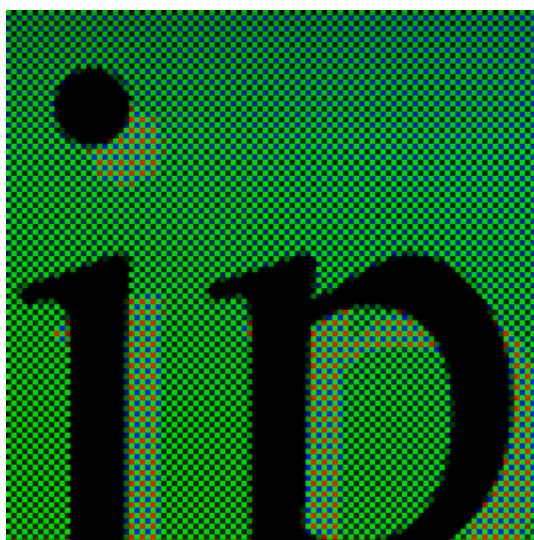
2.3. Interpolace barevného obrazu

Kvalitní interpolace má největší vliv na kvalitu výsledného obrazu. Nikdy nepůjde dosáhnout kvality původního obrazu, který dopadl na snímací čip digitálního fotoaparátu, protože část informace jednoduše chybí. Pokročilými algoritmy jsme schopni se k tomuto ideálnímu výsledku více či méně přiblížit.

Na následujících obrázcích bude demonstrováno, jak taková interpolace [11] v principu funguje. Na prvním obrázku vidíme originální motiv. Tento obraz dopadá na náš imaginární obrazový senzor. Ten díky Bayerově barevnému filtru [10] ovšem vidí obraz jako na následujícím obrázku. Jedná se o matici bodů v základních barvách.



Obrázek 7: Původní obraz



Obrázek 8: Obraz po aplikaci Bayerova barevného filtru

Pro větší názornost jsou na třech menších obrázcích znázorněny obrazy v jednotlivých základních barvách.



Obrázek 9: Pohled přes jednotlivé barevné filtry

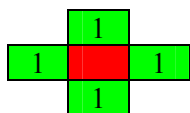
Na následujícím obrázku (obrázek 10) je výsledný obraz po jednoduché interpolaci. V souvislých plochách odpovídá obraz přesně (nebo téměř přesně) obrazu původnímu. Ale v místech s vyššími frekvencemi – typicky hrany a ostré přechody – jsou jasně vidět nevýhody interpolace. Dochází ke změkčení kresby (rozmazání obrazu) a k artefaktům v obraze. Nejvíce je to vidět na přechodech mezi černými a bílými plochami obrazu, kde dochází k posunu k jedné ze základních barev podle toho, do jakého místa barevné Bayerovy mřížky přechod na senzoru dopadl. Vedle je pro porovnání znovu umístěn originální obraz.



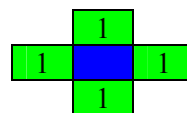
Obrázek 10: Obraz po barevné interpolaci a pro porovnání i původní obraz

2.3.1. Bilineární metoda interpolace

Bilineární interpolace Bayerovy mřížky je nejjednodušší metodou interpolace, která zachovává plné rozlišení výsledného obrazu. V implementaci pro interpolaci Bayerovy mřížky se v podstatě jedná o průměrování všech nejbližších sousedů požadované (počítané) barvy. Z podstaty průměrování je zřejmé, že výsledný obraz bude velmi měkký a bude postrádat drobné detaily. Díky pravidelné struktuře Bayerovy mřížky a neúplné barevné informaci jednotlivých pixelů vznikají v obraze výrazné barevné artefakty.



Výpočet zelené barvy červeného pixelu Bayerovy mřížky.



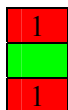
Výpočet zelené barvy modrého pixelu Bayerovy mřížky.



Výpočet červené barvy na zeleném pixelu v řadě s červenými pixely.



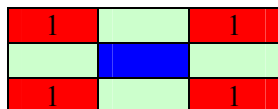
Výpočet modré barvy na zeleném pixelu v řadě s modrými pixely.



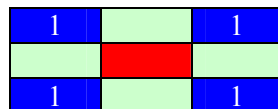
Výpočet červené barvy na zeleném pixelu v řadě s modrými.



Výpočet modré barvy na zeleném pixelu v řadě s červenými.



Výpočet červené barvy na modrém pixelu Bayerovy mřížky.

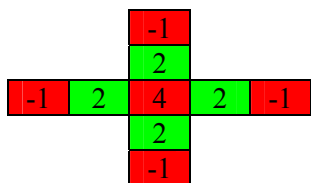


Výpočet modré barvy na červeném pixelu Bayerovy mřížky.

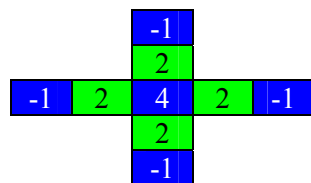
2.3.2. Pokročilá lineární metoda interpolace

Lineární interpolace obrazu vyniká především svou rychlostí. Tato pokročilá metoda vyniká nad bilineární interpolací podstatně vyšší ostroty obrazu i částečně potlačenými artefakty, které zaviňuje pravidelná struktura Bayerovy mřížky.

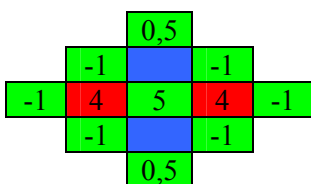
Následující metoda [1] v podstatě používá celkem 8 konvolučních filtrů o rozměru jádra 5×5.



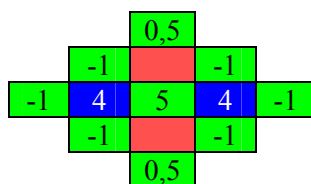
Výpočet zelené barvy, střed jádra se nachází na červeném pixelu.



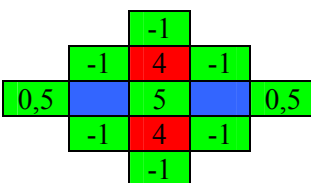
Výpočet zelené barvy, střed jádra se nachází na modrém pixelu.



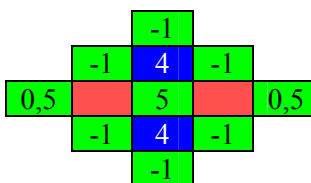
Výpočet červené barvy, střed jádra je v řadě s červenými pixely a ve sloupci s modrými.



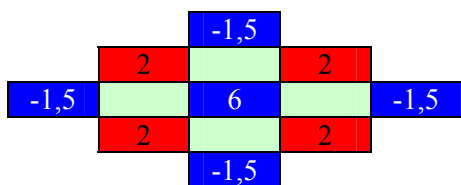
Výpočet modré barvy, střed jádra je v řadě s modrými pixely a ve sloupci s červenými.



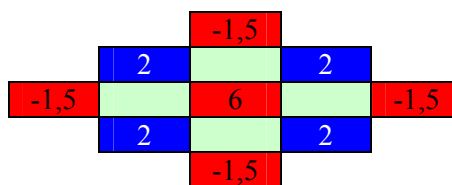
Výpočet červené barvy, střed jádra je v řadě s modrými pixely a ve sloupci s červenými.



Výpočet modré barvy, střed jádra je v řadě s červenými pixely a ve sloupci s modrými.



Výpočet červené barvy, střed jádra je v řadě s modrými pixely a ve sloupci též s modrými.



Výpočet modré barvy, střed jádra je v řadě s červenými pixely a ve sloupci též s červenými pixely.

Použití této metody při interpolaci obrazu dává velmi dobré obrazové výsledky, ostrý obraz bez nežádoucích rušivých barevných vad. Jako další lineární metody, ani tato nezabraňuje vzniku tzv. zipper efektu, tedy zubatých hran při nevhodné orientaci ostré hrany v obraze.

2.3.3. Metoda VNG (Threshold-based variable number of gradients)

Základní myšlenkou této metody je určení váženého gradientu ve všech osmi směrech v matici 5×5 bodů se středem v daném bodě obrazu, pro který se interpolují zbývající dvě barevné složky. Pro každou množinu vypočtených gradientů je určen práh, s jehož pomocí vybereme podmnožinu gradientů. Gradienty s nízkou hodnotou ukazují na pixely mající podobnou barevnou hodnotu, zatímco v oblastech s gradienty s vysokou hodnotou lze očekávat hodně jemných detailů a ostrých hran. Vybraná podmnožina gradientů nám určí množinu bodů, které váženě sečteme k získání výsledných hodnot.

Pro výpočet zelené a modré nebo červené hodnoty na červeném nebo modrém pixelu uvažujeme tuto Bayerovu mřížku:

R1	G2	R3	G4	R5
G6	B7	G8	B9	G10
R11	G12	R13	G14	R15
G16	B17	G18	B19	G20
R21	G22	R23	G24	R25

1) Vypočteme jednotlivé gradienty takto:

$$\Delta N = |G8 - G18| + |R3 - R13| + \frac{|B7 - B17|}{2} + \frac{|B9 - B19|}{2} + \frac{|G2 - G12|}{2} + \frac{|G4 - G14|}{2}$$

$$\Delta E = |G14 - G12| + |R15 - R13| + \frac{|B9 - B7|}{2} + \frac{|B19 - B17|}{2} + \frac{|G10 - G8|}{2} + \frac{|G20 - G18|}{2}$$

$$\Delta S = |G18 - G8| + |R23 - R13| + \frac{|B19 - B9|}{2} + \frac{|B17 - B7|}{2} + \frac{|G24 - G14|}{2} + \frac{|G22 - G12|}{2}$$

$$\Delta W = |G12 - G14| + |R11 - R13| + \frac{|B17 - B19|}{2} + \frac{|B7 - B9|}{2} + \frac{|G16 - G18|}{2} + \frac{|G6 - G8|}{2}$$

$$\Delta NE = |B9 - B17| + |R5 - R13| + \frac{|G8 - G12|}{2} + \frac{|G14 - G18|}{2} + \frac{|G4 - G8|}{2} + \frac{|G10 - G14|}{2}$$

$$\Delta SE = |B19 - B7| + |R25 - R13| + \frac{|G14 - G8|}{2} + \frac{|G18 - G12|}{2} + \frac{|G20 - G14|}{2} + \frac{|G24 - G18|}{2}$$

$$\Delta NW = |B7 - B19| + |R1 - R13| + \frac{|G12 - G18|}{2} + \frac{|G8 - G14|}{2} + \frac{|G6 - G12|}{2} + \frac{|G2 - G8|}{2}$$

$$\Delta SW = |B17 - B9| + |R21 - R13| + \frac{|G18 - G14|}{2} + \frac{|G12 - G8|}{2} + \frac{|G22 - G18|}{2} + \frac{|G16 - G12|}{2}$$

2) Vypočteme práh:

$T = 1,5 \cdot \min + 0,5 \cdot (MAX - \min)$, kde *min* je hodnota minimálního gradientu a *MAX* je hodnota maximálního gradientu

Hodnoty konstant 1,5 a 0,5 byly určeny empiricky. Výraz $1,5 \cdot \min$ je použit proto, pokud jsou všechny gradienty podobné a chceme je do dalších kroků výpočtu zahrnout všechny.

3) Pro všechny gradienty pod zvoleným prahem vypočteme sumu hodnot jednotlivých kanálů dle následující tabulky:

	R	G	B
N	$\frac{R3 + R13}{2}$	G8	$\frac{B7 + B9}{2}$
E	$\frac{R13 + R15}{2}$	G14	$\frac{B9 + B19}{2}$
S	$\frac{R13 + R23}{2}$	G18	$\frac{B17 + B19}{2}$
W	$\frac{R11 + R13}{2}$	G12	$\frac{B7 + B17}{2}$
NE	$\frac{R13 + R5}{2}$	$\frac{G4 + G8 + G10 + G14}{4}$	B9
SE	$\frac{R13 + R25}{2}$	$\frac{G14 + G18 + G20 + G24}{4}$	B19
NW	$\frac{R13 + R1}{2}$	$\frac{G2 + G8 + G6 + G12}{4}$	B7
SW	$\frac{R13 + R21}{2}$	$\frac{G12 + G16 + G18 + G22}{4}$	B17

Pro výpočet modré nebo červené hodnoty na zeleném pixelu uvažujeme následující Byerovu mřížku:

G1	R2	G3	R4	G5
B6	G7	B8	G9	B10
G11	R12	G13	R14	G15
B16	G17	B18	G19	B20
G21	R22	G23	R24	G25

1) Vypočteme jednotlivé gradienty takto:

$$\Delta N = |G3 - G13| + |B8 - B18| + \frac{|G7 - G17|}{2} + \frac{|G9 - G19|}{2} + \frac{|R2 - R12|}{2} + \frac{|R4 - R14|}{2}$$

$$\Delta E = |G15 - G13| + |R14 - R12| + \frac{|G9 - G7|}{2} + \frac{|G19 - G17|}{2} + \frac{|B10 - B8|}{2} + \frac{|B20 - B18|}{2}$$

$$\Delta S = |G23 - G13| + |B8 - B18| + \frac{|G19 - G9|}{2} + \frac{|G17 - G7|}{2} + \frac{|R24 - R14|}{2} + \frac{|R22 - R12|}{2}$$

$$\Delta W = |G11 - G13| + |R12 - R14| + \frac{|G17 - G19|}{2} + \frac{|G7 - G9|}{2} + \frac{|B16 - B18|}{2} + \frac{|B6 - B8|}{2}$$

$$\Delta NE = |G9 - G17| + |G5 - G13| + |R4 - R12| + |B10 - B18|$$

$$\Delta SE = |G19 - G7| + |G25 - G13| + |B20 - B18| + |R24 - R12|$$

$$\Delta NW = |G7 - G19| + |G1 - G13| + |B6 - B18| + |R2 - R14|$$

$$\Delta SW = |G17 - G9| + |G21 - G13| + |R22 - R14| + |B16 - B18|$$

2) Stejným způsobem vypočteme práh $T = 1,5 \cdot \min + 0,5 \cdot (\text{MAX} - \min)$

3) Pro všechny gradienty menší než vypočtený práh vypočteme sumu jednotlivých kanálů dle následující tabulky:

	R	G	B
N	$\frac{R2 + R4 + R12 + R14}{4}$	$\frac{G3 + G13}{2}$	B8
E	R14	$\frac{G13 + G15}{2}$	$\frac{B8 + B10 + B18 + B20}{4}$
S	$\frac{R12 + R14 + R22 + R24}{4}$	$\frac{G13 + G23}{2}$	B18
W	R12	$\frac{G11 + G13}{2}$	$\frac{B6 + B8 + B16 + B18}{4}$

NE	$\frac{R4 + R12}{2}$	G9	$\frac{B10 + B18}{2}$
SE	$\frac{R12 + R24}{2}$	G19	$\frac{B8 + B20}{2}$
NW	$\frac{R2 + R14}{2}$	G7	$\frac{B6 + B18}{2}$
SW	$\frac{R14 + R22}{2}$	G17	$\frac{B8 + B16}{2}$

Metoda VNG se dá rozšířit na výpočet pomocí 4 barev, kdy budeme považovat hodnoty zelených pixelů v lichých a sudých řádcích za různé barvy. Toto řešení částečně potlačuje barevné artefakty metody VNG.

2.3.4. Metoda Pixel Grouping

Tato metoda založená na vlastnostech přirozených scén má za úkol potlačovat některé artefakty interpolačních metod. I tato metoda patří mezi velice rychlé, podobně jako ostatní lineární metody.

Na začátku vývoje této metody stálo pozorování, že obrazy reálných scén se skládají ze skupin pixelů, které jsou rozumně propojeny a mají podobné vlastnosti, tedy přibližně stejný jas a barvu. Identifikací skupin těchto pixelů můžeme lépe interpolovat chybějící barevné informace a zlepšit vzhled výsledných obrázků.

Pouze zelený kanál je interpolován pomocí identifikace skupin pixelů. Červený a modrý kanál je generován pomocí hybridní barevné korekce a modifikovaným algoritmem, který zajišťuje hladké přechody barevného tónu a zlepšuje vzhled hran.

Zelený kanál je interpolován z minimálního gradientu 4-okolí červeného a modrého pixelu. Ve výsledném výpočtu je ve směru minimálního gradientu kladen trojnásobný důraz na tento pixel.

V okolí zelených pixelů jsou hodnoty červeného a modrého kanálu interpolovány pomocí funkce zajišťující hladké přechody odstínu pixelu.

Hodnoty modrého a červeného kanálu v okolí červených a modrých pixelů jsou vypočteny pomocí menšího ze severovýchodního či severozápadního gradientu všech odpovídajících pixelů a následně pomocí kombinace těchto pixelů zajišťující nejkonzistentnější odstín.

Uvažujme tuto Bayerovu mřížku:

R1	G2	R3	G4	R5
G6	B7	G8	B9	G10
R11	G12	R13	G14	R15
G16	B17	G18	B19	G20
R21	G22	R23	G24	R25

Výpočet zelené hodnoty na pozici modrého nebo červeného pixelu:

1. Vypočteme gradienty ve 4 směrech

$$\Delta N = 2 \cdot |R3 - R13| + |G8 - G18|$$

$$\Delta E = 2 \cdot |R13 - R15| + |G12 - G14|$$

$$\Delta W = 2 \cdot |R11 - R13| + |G12 - G14|$$

$$\Delta S = 2 \cdot |R13 - R23| + |G8 - G18|$$

2. Určíme minimální gradient a podle toho vypočteme výslednou hodnotu

$$\Delta N \text{ je minimální: } G13 = \frac{3 \cdot G8 + R13 + G18 - R3}{4}$$

$$\Delta E \text{ je minimální: } G13 = \frac{3 \cdot G14 + R13 + G12 - R15}{4}$$

$$\Delta W \text{ je minimální: } G13 = \frac{3 \cdot G12 + R13 + G14 - R11}{4}$$

$$\Delta S \text{ je minimální: } G13 = \frac{3 \cdot G18 + R13 + G8 - R23}{4}$$

Výpočet modré a červené hodnoty na pozici zeleného pixelu:

1. Nejdříve si definujeme funkci *hue_transit*

$$\text{hue_transit}(g1, g2, g3, v1, v3) =$$

pokud je $g1 < g2 < g3$ nebo $g1 > g2 > g3$ pak

$$\text{výsledek je roven } v1 + (v3 - v1) \cdot \frac{g2 - g1}{g3 - g1}$$

$$\text{jinak je výsledek roven } \frac{v1 + v3}{2} + \frac{2 \cdot g2 - g1 - g3}{4}$$

2. $B8 = \text{hue_transit}(G7, G8, G9, B7, B9)$

$$R8 = \text{hue_transit}(G3, G8, G13, R3, R13)$$

Výpočet modré nebo červené hodnoty na pozici červeného nebo modrého pixelu:

1. Vypočteme dva úhlopříčné gradienty

$$\Delta NE = |B9 - B17| + |R5 - R13| + |R13 - R21| + |G9 - G13| + |G13 - G17|$$

$$\Delta NW = |B7 - B19| + |R1 - R13| + |R13 - R25| + |G7 - G13| + |G13 - G19|$$

2. Podle menšího gradientu vypočteme výslednou hodnotu

$$B13 = \text{hue_transit}(G9, G13, G17, B9, B17) \text{ pro } \Delta NE \leq \Delta NW$$

$$B13 = \text{hue_transit}(G7, G13, G19, B7, B19) \text{ jinak}$$

2.3.5. Metoda AHD (Adaptive Homogeneity-Directed)

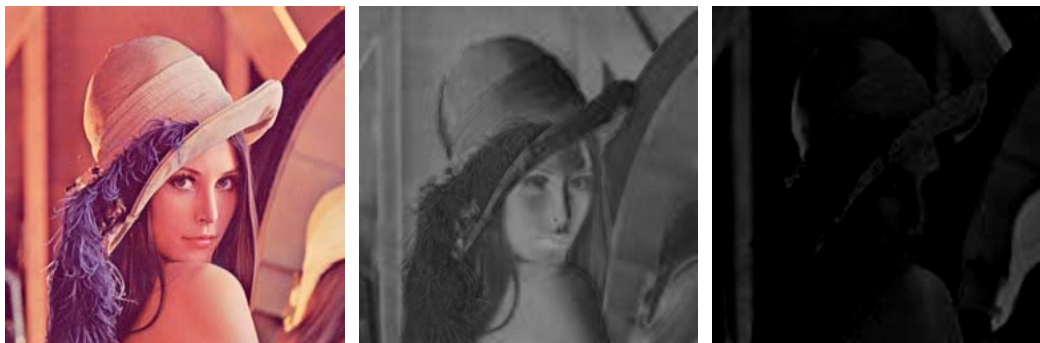
Metoda AHD [2] se snaží řešit tři nejčastější problémy interpolačních metod Bayerovy mřížky. Barevné artefakty vzniklé v důsledku interpolace ze špatně zvolených okolních pixelů, barevné artefakty vzniklé v důsledku vlastností interpolační metody a aliasing.

Barevným artefaktům vzniklým kvůli špatně zvolným okolním pixelům pro interpolaci se snaží AHD zamezit interpolací ve směru, kde je těchto barevných artefaktů méně. Aliasing je řešen pomocí filterbank techniky. Artefakty vzniklé v důsledku použité interpolační metody jsou potlačeny nelineárním iterativním filtrem.

V prvním kroku se předběžně interpoluje vstupní obraz s Bayerovou mřížkou pouze ve vertikálním a horizontálním směru. Vertikální a horizontální směr je díky pravoúhlé Bayerově mřížce podstatně jednodušší na interpolaci. Problém odhadu chybějících barevných komponent jednotlivých pixelů se může interpretovat jako potlačování aliasingu ze vzorkovaného signálu. V zeleném kanálu je toto vyřešeno pomocí lineární kombinace známých pixelů v červeném a modrém kanálu. Červené a modré kanály jsou rekonstruovány potlačením vysokých frekvencí vzorkovaného signálu použitím interpolovaného zeleného kanálu.

Nechť je X množina 2D pozic pixelů a Y je množina tří hodnot RGB: $y \in Y, y = [R, G, B]^T$, kde R, G a B reprezentují jednotlivé složky barevného modelu. Potom je barevný obraz $f: X \rightarrow Y$ je zobrazení mezi polohou pixelu a jeho RGB hodnotou. Samotné jednotlivé složky obrazu si označme jako $R(\cdot)$, $G(\cdot)$ a $B(\cdot)$.

Předpokládejme, že se rozdílový obraz $G(x) - R(x)$ mění jen velmi pomalu. To znamená, že vysoké frekvence v rozdílových obrazech zanikají mnohem rychleji než v samotném $G(x)$. Síla $G(x) - R(x)$ na frekvenci $\xi = \frac{\pi}{3}$ je přibližně pětina v porovnání se samotným kanálem $G(x)$ na stejné frekvenci.



Obrázek 11: Původní obraz a rozdílové barevné složky (R-G a G-B)

$G_0(x)$ a $G_1(x)$ jsou sudé a liché sloupce vzorkovaného signálu obrazu $G(x)$. Pro rozdílový obraz $G(x) - R(x)$ je na základě Bayerovy mřížky $G_0(x)$ přímo dostupný, zatímco $G_1(x)$ nikoliv. Abychom ze vztahu $G(x) = G_0(x) + G_1(x)$ získali $G_1(x)$, uvažujme lineární filtr h : $G(x) = h(x) * G(x)$. Pokud je $G(x)$ pásmově omezený, pak je $h(x)$ ideální filtr typu dolní propust. Po dosazení dostaneme:

$$G(x) = h_0(x) * G_0(x) + h_1(x) * G_0(x) + h_0(x) * G_1(x) + h_1(x) * G_1(x)$$

Díky vlastnostem, kdy je pro sudé sloupce $h_1(x) * G_0(x)$ a $h_0(x) * G_1(x)$ nulové a pro liché sloupce je $h_0(x) * G_0(x)$ a $h_1(x) * G_1(x)$ nulové je $G(x)$ rovno:

$$G(x) = \begin{cases} h_0(x) * G_0(x) + h_1(x) * G_1(x) & \text{pro } x \text{ sudé} \\ h_1(x) * G_0(x) + h_0(x) * G_1(x) & \text{pro } x \text{ liché} \end{cases}$$

Z Bayerovy mřížky vyplývá, že $h_0(x) * G_1(x)$ stále není dostupné. Proto je $h_0(x)$ určeno jako: $h_0(x) * G_1(x) \approx h_0(x) * R_1(x)$. Omezení tímto výrazem umožňuje substituci:

$$G(x) = \begin{cases} h_0(x) * G_0(x) + h_1(x) * G_1(x) & \text{pro } x \text{ sudé} \\ h_1(x) * G_0(x) + h_0(x) * G_1(x) & \text{pro } x \text{ liché} \end{cases}$$

$$G(x) = \begin{cases} G_0(x) & \text{pro } x \text{ sudé} \\ h_1(x) * G_0(x) + h_0(x) * G_1(x) & \text{pro } x \text{ liché} \end{cases}$$

$$G(x) = G_0(x) + h_1(x) * G_0(x) + h_0(x) * R_1(x)$$

K návrhu FIR filtru s nulovou fází $h(x)$, který splňuje naše omezení, je nutno vyřešit následující problém: $\hat{h}_{opt}(\xi) = \arg \min_h \left\| \hat{w}(\xi) (1 - \hat{h}(\xi)) \right\|^2$, kde $\hat{\cdot}$ značí Fourierovu transformaci, $w(\cdot)$ je váhová funkce. K zajištění omezujících podmínek předpokládejme, že $\hat{h}_0(\xi = 0) = \hat{h}_0(\xi = \pi) = 0$. Ve výsledku dostaneme výraz:

$$\hat{h}_0(\xi) \hat{R}_1(\xi) = \hat{h}_0(\xi) \hat{G}_1(\xi) = \left(\frac{1}{2\hat{h}(\xi)} + \frac{1}{2\hat{h}(\xi - \pi)} \right) \left(\frac{1}{2\hat{G}(\xi)} + \frac{1}{2\hat{G}(\xi - \pi)} \right)$$

V druhém kroku se metodou AHD obraz interpoluje do finální podoby. Aby se předešlo vzniku barevných artefaktů, jsou porovnávány vertikálně a horizontálně interpolované obrazy. Největší

chyby se jiné interpolační metody dopouštějí špatně zvoleným směrem interpolace. Takto vzniklé barevné artefakty jsou pro lidské oko velmi rušivé. AHD vytváří tzv. mapu homogenity k analýze a potlačení těchto barevných artefaktů.

Podobnost okolních pixelů použitých pro konečnou interpolaci metoda AHD určuje pomocí vzdáleností jednotlivých složek ve zvolené doméně. Kvůli zaměření této metody pro výstupy pozorované lidským okem, byl zvolen barevný model CIE Lab, který je normalizován na citlivost lidského oka. Podobnost pixelů v CIE Lab modelu je určena zvláště pro L kanál a ab kanály. Všechny pixely, které jsou v dané vzdálenosti v L kanálu a ab kanálech, jsou považovány za podobné. Protože jsou pixely porovnávány odděleně, v 3D modelu tvoří oblast zájmu válcový tvar, což je v kontrastu s kulovým tvarem, který by vzniknul porovnáváním Euklidovské vzdálenosti všech tří složek najednou. Válcový tvar lépe odpovídá modelu odrazivosti, který je produktem světelného spektra a odrazivosti objektu pro různé vlnové délky. Rozdíly v jasu objektu ovlivňuje mnoho věcí, zatímco rozdíly v chrominanci způsobuje zejména specifická odrazivost objektu. Proto je použito toto nezávislé porovnávání L a ab kanálu, které je pro AHD více efektivní.

Interpolační artefakty vznikají v důsledku špatné volby směru interpolace. Interpolace podél hranice objektu je preferovanější než interpolace v kolmém směru, tedy napříč hranicí, protože nespojitost signálu na hranici obsahuje velmi obtížně odhadovatelné vysokofrekvenční komponenty. Špatně zvolený směr interpolace, tedy napříč hranicí objektu, způsobuje vznik takového pixelu, jehož barva není v relaci s fyzickým objektem v obraze. Takto postižený pixel má ve svém okolí málo podobných pixelů a proto se dá považovat špatně zvolený směr interpolace za izolovanou událost. Pro porovnávání okolí pixelů se používá mapy homogenity. Ve výsledku jsou zkombinovány jen okolní pixely, jejichž hodnota homogenity je nejvyšší, tzn. jejichž vzdálenost v L a ab kanálu od daného pixelu leží pod zvoleným prahem.

V třetím volitelném kroku jsou potlačeny případné barevné artefakty iterativní metodou. I při perfektně interpolovaných pixelech mohou vzniknout ve výstupním obraze barevné artefakty. To je dáno omezením samotné interpolace. Normálně jsou interpolační artefakty podstatně méně viditelné než artefakty vzniklé volbou špatného směru interpolace, přesto mohou být stále rušivé. Iterativní metoda na jejich potlačení využívá předpokladu, že rozdílové obrazy R-G a B-G se mění velmi pomalu. Následující nelineární metoda potlačuje tyto malé variace v barvách a zároveň zachovává hrany:

$$1) R = \text{median}(R - G) + G$$

$$2) B = \text{median}(B - G) + G$$

$$3) G = \frac{\text{median}(G - R) + \text{median}(G - B) + R + B}{2}$$

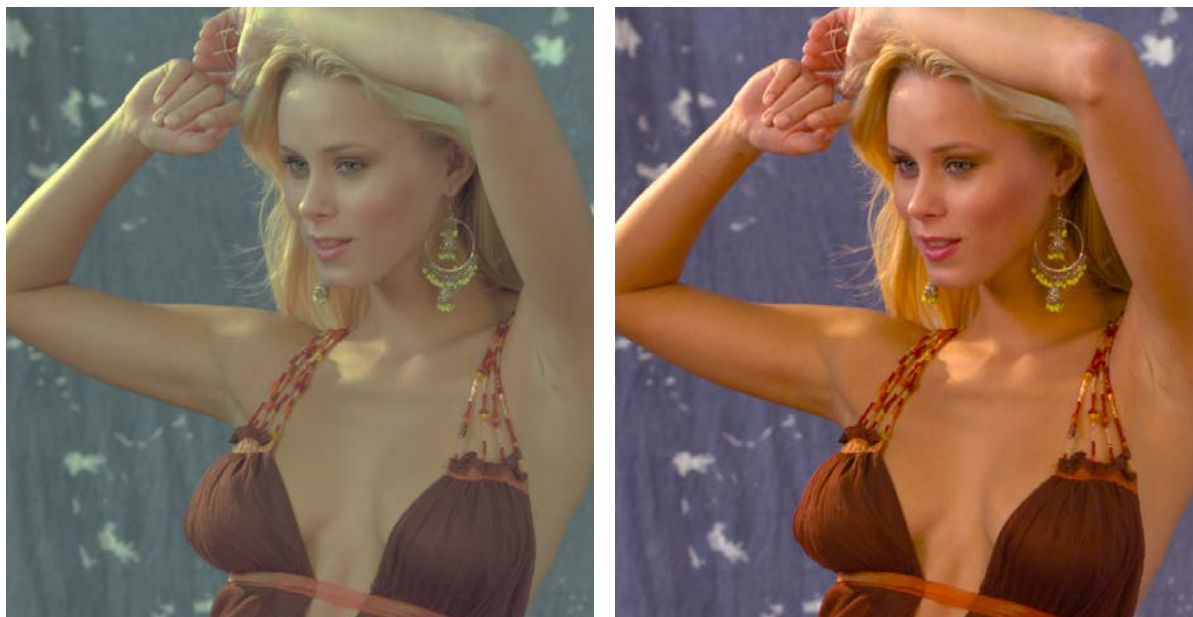
Výše uvedené kroky opakujeme n -krát.

Metoda AHD využívá jen interpolaci ve vertikálním a horizontálním směru. Obecně by bylo dobré interpolovat i v diagonálních směrech, ale díky omezením pravoúhlé Bayerovy mřížky je velmi

obtížné zahrnout interpolaci v diagonálních směrech do AHD. Tato nevýhoda je částečně potlačena výpočtem adaptivního prahu homogenity pro jednotlivé pixely obrazu.

2.4. Kalibrace barev

Obrazový senzor nemá přísně lineární citlivost na světelnou intenzitu. Stejně jako je různě citlivý na různé barvy. Toto je třeba eliminovat při zpracování obrazu. Bez úpravy barev by byl výsledný obraz mdlý (málo kontrastní) a nebude barevně věrný.



Obrázek 12: Původní obraz a obraz po aplikaci křivek

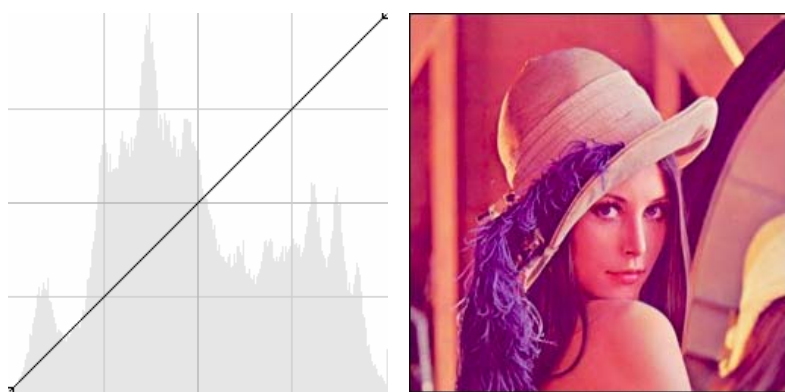
Úprava pomocí křivek zajistí použitelnost obrazu, ale barevné věrnosti se dá dosáhnout až pomocí barevných profilů pro jednotlivé typy digitálních fotoaparátů a zavedením správy barev v procesu konverze RAW souborů do běžných počítačových obrazových formátů.

Úprava obrazu pomocí tzv. tónových křivek.

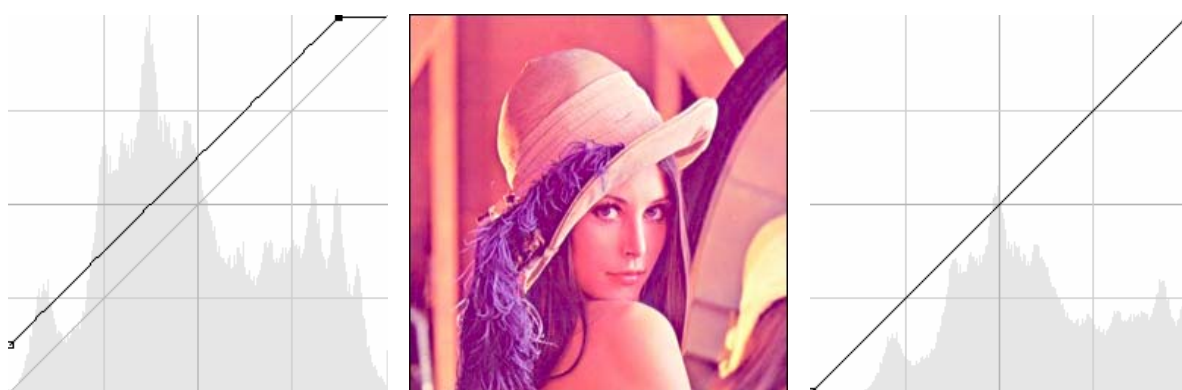
Křivky (z anglického *curves*) je jedna z celkových úprav obrazu. Pomocí křivek [18] se dá velmi dobře ovlivňovat histogram obrazu. Téměř všechny základní úpravy (úprava jasu, kontrastu, ale i inverze) se dají převést na vhodnou křivku.

Prakticky se jedná o křivku mapování vstupních hodnot obrazu na výstupní. Základem je tzv. základní (nebo lineární) křivka, kdy výstupní hodnoty funkce přesně odpovídají vstupním. Toto je vidět na obrázku č. 13.

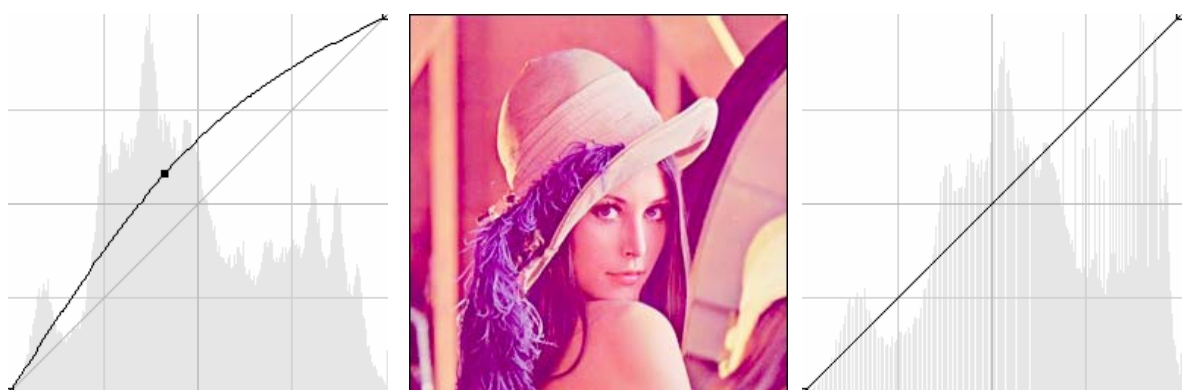
Úprava pomocí křivek je velmi rychlá a dá se naprogramovat pomocí vyhledávací tabulky (LUT). Barvy obrazu můžeme ovlivňovat nezávislou aplikací křivek na jednotlivé barevné kanály. Aby výsledný obraz vypadal přirozeně je důležité zachovat monotónnost křivky.



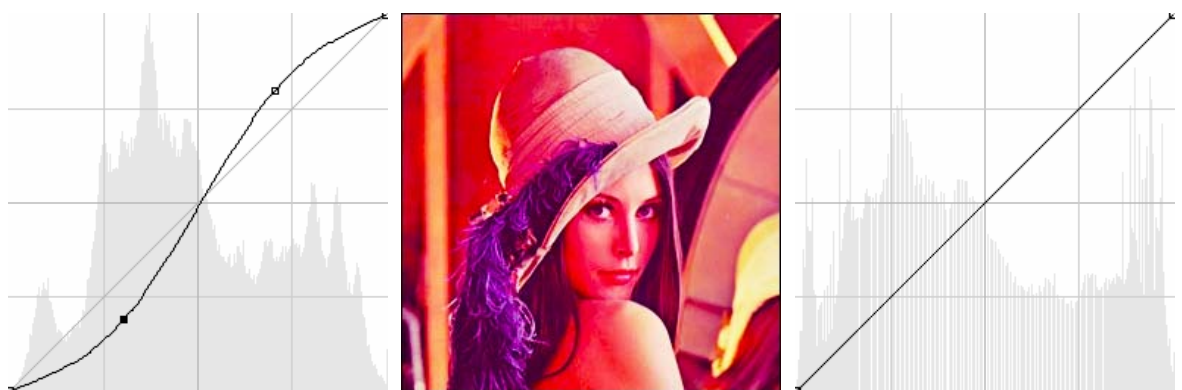
Obrázek 13: Obrázek bez úpravy (lineární křivka)



Obrázek 14: Zvýšení jasu celého obrázku (tvar křivky, obrázek po aplikaci křivky a výsledný histogram)



Obrázek 15: Zvýšení jasu středních tónů



Obrázek 16: Zvýšení kontrastu obrázu (S-křivka)



Obrázek 17: Inverze barev obrazu

2.5. *Vyvážení bílé barvy*

Snímky pořízené za různých světelných podmínek mají různé podání barev.

Lidské oko a mozek jsou velmi přizpůsobivé. Když víme, že nějaký předmět scény má být bílý, tak se nám jeví bílý, ať už je scéna osvětlena žlutým světlem žárovky, bílým světlem blesku či modrým až zeleným světlem různých výbojek. Digitální fotoaparát však vidí scénu absolutně, a tak bílý objekt osvětlený žlutým světlem bude na fotografii žlutý.

Digitální fotoaparáty se snaží toto řešit pomocí dodatečného automatického vyvažování bílé barvy. Vzájemným posouváním histogramu červeného a modrého kanálu se snaží vyvážit barvy na snímku tak, aby to odpovídalo vnímání lidským okem.

Existuje několik metod automatického vyvážení bílé barvy založených na předpokladech:

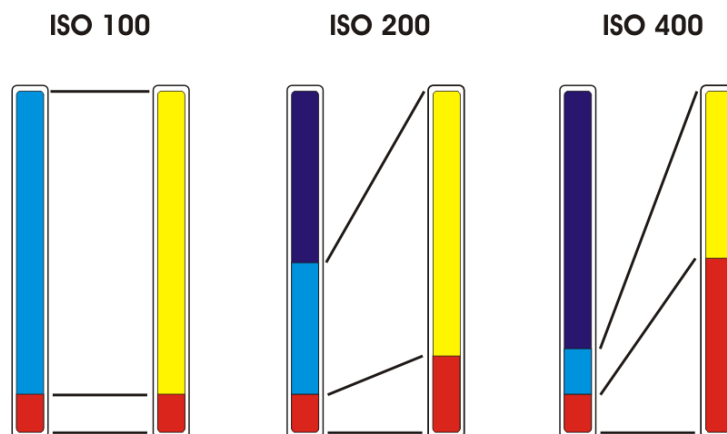
- součet hodnot všech bodů obrazu tvoří šedou barvu
- nejjasnější bod obrazu je bílé barvy

Automatické vyvážení bílé v praxi funguje dobře jen na dostatečně obsahově rozmanitých snímcích. Pokud je scéna složena jen z červené krychle na zeleném pozadí nebo pokud máme scénu osvětlenou více světelnými zdroji o různé barevné teplotě, všechny metody automatického vyvážení bílé budou podávat špatné výsledky.

Jako nejlepší řešení se jeví vyvažování bílé barvy za asistence uživatele při převodu RAW souborů na PC. Uživatel u kritických snímků zvolí bod nebo oblast, která by měla být bílá a aplikace už přesně vyváží bílou barvu sama.

2.6. Odstranění šumu

Obrazové senzory jsou elektronická analogová zařízení. Jako taková produkují kromě obrazové informace i jistou míru šumu. Ten se na snímcích z digitálních fotoaparátů projevuje tím více, čím více je potřeba signál zesilovat, tedy čím vyšší citlivost ISO je použita pro expozici snímku.



Obrázek 18: Růst vlivu šumu (červeně) při stoupající citlivosti

Šum se u snímků z digitálních fotoaparátů projevuje nevzhlednými a rušivými barevnými skvrnami (složenými z více pixelů). Toto se snažíme potlačit. Existuje několik metod, jak toho dosáhnout:

- rozmazání celého obrazu
- adaptivní rozmazání obrazu (zachovává ostré hrany)
- detekce vzoru šumu na jednolitě ploše a jeho odečtení od celého snímku
- adaptivní medián pouze barevné obrazové informace

První metoda značně degraduje obraz.

Druhá metoda už dává lepší výsledky a zdá se, že její obdobu používá hodně současných kompaktních fotoaparátů.

Pomocí detekce vzoru šumu (pro daný typ přístroje a konkrétní nastavení je vždy stejný a dají se tak použít i přednastavené profily) lze dosáhnout velmi pěkných výsledků. Takto lze téměř dokonale odstranit šum a zachovat maximum obrazové informace. Na tomto principu funguje software jako např. NeatImage či Noise Ninja.

Adaptivní medián v barevné oblasti snímku dává subjektivně velmi pěkné výsledky. Jsou vyhlazovány jen barevné skvrny v jednolitéch plochách, hrany jsou v maximální míře zachovávány. Další podstatnou věcí je, že v jasové oblasti je šum zachován. Zůstává tak jen drobný monochromatický šum, který subjektivně zvyšuje ostrost snímků. Tato metoda nejlépe funguje v barevných modelech typu Lab, proto je nutný převod do modelu Lab a zpátky do barevného modelu RGB.



Obrázek 19: Výřez z obrazu pořízeného při vysoké citlivosti



Obrázek 20: Vyhlazení obrazu, šum je potlačen, bohužel i včetně kresby



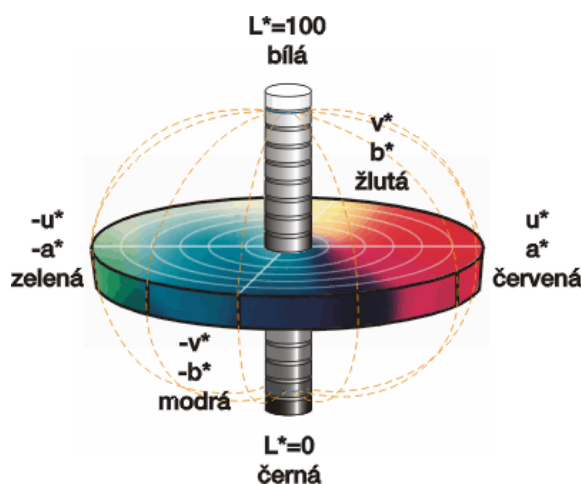
Obrázek 21: Obraz po aplikaci adaptivního vyhlazení v barově rozdílových kanálech

2.7. Převody mezi barevnými modely

Při úpravách obrazu je někdy výhodnější pracovat v jiném barevném modelu [9] než ve standardním RGB. Velmi výhodný je barevný model Lab.

Pro názorné a matematicky snadné určování barev byly navrženy barevné prostory CIE Lab a CIE Luv, které si získaly (zejména Lab) výsadní postavení při popisu barev nezávislých na zařízení. Barevné modely CIE Lab a CIE Luv byly definovány komisí CIE v roce 1976. Jejich konstrukce je založena na faktu, že barva nemůže být zároveň zelená a červená, stejně tak jako modrá a žlutá. Proto lze zavést hodnoty, popisující polohu odstínu mezi zelenou a červenou, respektive mezi modrou a žlutou. Máme-li barvu definovanou v CIELAB, pak L definuje jas, a udává polohu mezi primárními barvami R/G a b udává polohu mezi primárními barvami Y/B. Ve středu kruhového diagramu je neutrální oblast, v prostorovém tvaru pak středem tělesa popisujícího Lab prostor, procházejí neutrální barvy (černá, stupně šedé a bílá).

Polohu barvy v souřadnicích a , b vynášíme do kruhového diagramu, skutečnou polohu barvy obdržíme po zahrnutí hodnoty L a umístění barvy do prostoru.



Obrázek 22: Prostor barevného modelu Lab

Při převodu mezi barevnými modely RGB a Lab nelze postupovat přímo, ale musíme použít jako mezikrok barevný prostor XYZ. Ten byl sestaven roku 1931 jako model reflektující působení různých vlnových délek světla (na fyzikálním základě) na pozorovatele. Je to model, odpovídající vlivu světelného spektra na jednotlivá čidla oka (červená, zelená a modrá) - podobně pracují i některé měřicí přístroje.

2.7.1. Převod RGB na XYZ

Pro převod se předpokládá barevný prostor sRGB, viz. [19]. Vstupní hodnoty jsou R , G , B .

$$[XYZ] = [rgb][M]$$

$$r = \begin{cases} \frac{R}{12,92} & \text{pro } R \leq 0,04045 \\ \left(\frac{R + 0,055}{1,055} \right)^{2,4} & \text{pro } R > 0,0405 \end{cases}$$

$$g = \begin{cases} \frac{G}{12,92} & \text{pro } G \leq 0,04045 \\ \left(\frac{G + 0,055}{1,055} \right)^{2,4} & \text{pro } G > 0,0405 \end{cases}$$

$$b = \begin{cases} \frac{B}{12,92} & \text{pro } B \leq 0,04045 \\ \left(\frac{B + 0,055}{1,055} \right)^{2,4} & \text{pro } B > 0,0405 \end{cases}$$

Pokud máme chorminanční souřadnice RGB systému (x_r, y_r) , (x_g, y_g) , (x_b, y_b) a jejich referenční bílou (X_w, Y_w, Z_w) , pak se matice M vypočítá takto:

$$[M] = \begin{bmatrix} S_r X_r & S_r Y_r & S_r Z_r \\ S_g X_g & S_g Y_g & S_g Z_g \\ S_b X_b & S_b Y_b & S_b Z_b \end{bmatrix}$$

$$X_r = \frac{x_r}{y_r}, \quad Y_r = 1, \quad Z_r = \frac{1 - x_r - y_r}{y_r}$$

$$X_g = \frac{x_g}{y_g}, \quad Y_g = 1, \quad Z_g = \frac{1 - x_g - y_g}{y_g}$$

$$X_b = \frac{x_b}{y_b}, \quad Y_b = 1, \quad Z_b = \frac{1 - x_b - y_b}{y_b}$$

$$[S_r \quad S_g \quad S_b] = [X_w \quad Y_w \quad Z_w] \begin{bmatrix} X_r & Y_r & Z_r \\ X_g & Y_g & Z_g \\ X_b & Y_b & Z_b \end{bmatrix}^{-1}$$

Pro sRGB a osvětlení D65 [0.950456, 1, 1.088754] vychází hodnoty takto:

$$[M] = \begin{bmatrix} 0,412424 & 0,212656 & 0,0193324 \\ 0,357579 & 0,715158 & 0,119193 \\ 0,180464 & 0,0721856 & 0,950444 \end{bmatrix}$$

2.7.2. Převod XYZ na Lab

Převod z XYZ na Lab [20] opět potřebuje referenční bílou (X_w, Y_w, Z_w)

$$\varepsilon = \frac{216}{24389} \doteq 0,008856, \kappa = \frac{24389}{27} \doteq 903,3$$

$$x_r = \frac{X}{X_W}, y_r = \frac{Y}{Y_W}, z_r = \frac{Z}{Z_W}$$

$$f_x = \begin{cases} \sqrt[3]{x_r} & x_r > \varepsilon \\ \frac{\kappa x_r + 16}{116} & x_r \leq \varepsilon \end{cases}, f_y = \begin{cases} \sqrt[3]{y_r} & y_r > \varepsilon \\ \frac{\kappa y_r + 16}{116} & y_r \leq \varepsilon \end{cases}, f_z = \begin{cases} \sqrt[3]{z_r} & z_r > \varepsilon \\ \frac{\kappa z_r + 16}{116} & z_r \leq \varepsilon \end{cases}$$

$$L = 116f_y - 16$$

$$a = 500(f_x - f_y)$$

$$b = 200(f_y - f_z)$$

2.7.3. Převod Lab na XYZ

Převod Lab do XYZ [21] opět potřebuje referenční bílou (X_W, Y_W, Z_W)

$$\varepsilon = \frac{216}{24389} \doteq 0,008856, \kappa = \frac{24389}{27} \doteq 903,3$$

$$f_x = \frac{a}{500} + f_y, f_y = \begin{cases} \frac{L+16}{116} & y_r > \varepsilon \\ \frac{\kappa y_r + 16}{116} & y_r \leq \varepsilon \end{cases}, f_z = f_y - \frac{b}{200}$$

$$x_r = \begin{cases} f_x^3 & f_x^3 > \varepsilon \\ \frac{116f_x - 16}{\kappa} & f_x^3 \leq \varepsilon \end{cases}$$

$$y_r = \begin{cases} \left(\frac{L+16}{116}\right)^3 & L > \varepsilon \\ \frac{L}{\kappa} & L \leq \varepsilon \end{cases}$$

$$z_r = \begin{cases} f_z^3 & f_z^3 > \varepsilon \\ \frac{116f_z - 16}{\kappa} & f_z^3 \leq \varepsilon \end{cases}$$

$$X = x_r X_W, Y = y_r Y_W, Z = z_r Z_W$$

2.7.4. Převod XYZ na RGB

Výstupním barevným prostorem je sRGB, viz. [22].

$$[r \ g \ b] = [X \ Y \ Z][M]^{-1}$$

Pro osvětlení D65 platí:

$$[M]^{-1} = \begin{bmatrix} 3,24071 & -0,969258 & 0,0556352 \\ -1,53726 & 1,87599 & -0,203996 \\ -0,498571 & 0,0415557 & 1,05707 \end{bmatrix}$$

$$R = \begin{cases} 12,92r & r \leq 0,0031308 \\ 1,055r^{\frac{1}{2,4}} - 0,055 & r > 0,0031308 \end{cases}$$

$$G = \begin{cases} 12,92g & g \leq 0,0031308 \\ 1,055g^{\frac{1}{2,4}} - 0,055 & g > 0,0031308 \end{cases}$$

$$B = \begin{cases} 12,92b & b \leq 0,0031308 \\ 1,055b^{\frac{1}{2,4}} - 0,055 & b > 0,0031308 \end{cases}$$

2.8. Závěrečné úpravy obrazu

Nakonec se dá ještě obraz upravit několika způsoby. Optická soustava objektivu není nikdy dokonalá a některé její vady se dají dodatečně celkem úspěšně odstranit. Jde zejména o geometrické vady, barevné vady a úbytek světla v rozích. Tyto vady obrazu jsou pro danou kombinaci digitálního fotoaparátu, objektivu, použité ohniskové vzdálenosti a nastavené cloně vždy stejné. To by umožňovalo, po vybudování příslušné databáze, kde by se s těmito parametry svázaly příslušné korekční koeficienty příslušných funkcí, dělat tyto opravy zcela automaticky.

2.9. Popis formátu CR2

Formát Canon CR2 staví na základní struktuře formátu TIFF v6 [3].

Soubor obsahuje celkem 4 oddíly. Podstatné jsou ale pro nás jen první a poslední.

První oddíl obsahuje malý náhledový obrázek, typicky ve čtvrtinovém rozlišení v porovnání k maximálnímu.

Poslední oddíl obsahuje samotná obrazová RAW data. Ta jsou bezztrátově komprimována pomocí algoritmu LS-JPEG a mají 12 bitovou barevnou hloubku. Surová data obsahují informace ze všech buněk snímače, včetně oblastí, které se neúčastní tvorby výsledného obrazu. Tyto okrajové oblasti lze využít k detekci černé barvy apod.

Bajty jsou v tzv. little-endian pořadí, tedy jsou řazeny od nejméně významného k nejvíce významnému bajtu.

2.9.1. LS-JPEG

LS-JPEG je zkratkou z Lossless JPEG. Jedná se o dodatek z roku 1993 ke standardu JPEG [4]. Používá úplně odlišnou techniku než běžná ztrátová komprese JPEG. Jde o prediktivní schéma založené na třech nejbližších sousedech (horní, levý a horní-levý) a k distribuci chyby predikce je využito entropie kódování chyby [13], konkrétně jde o Huffmanův kód.

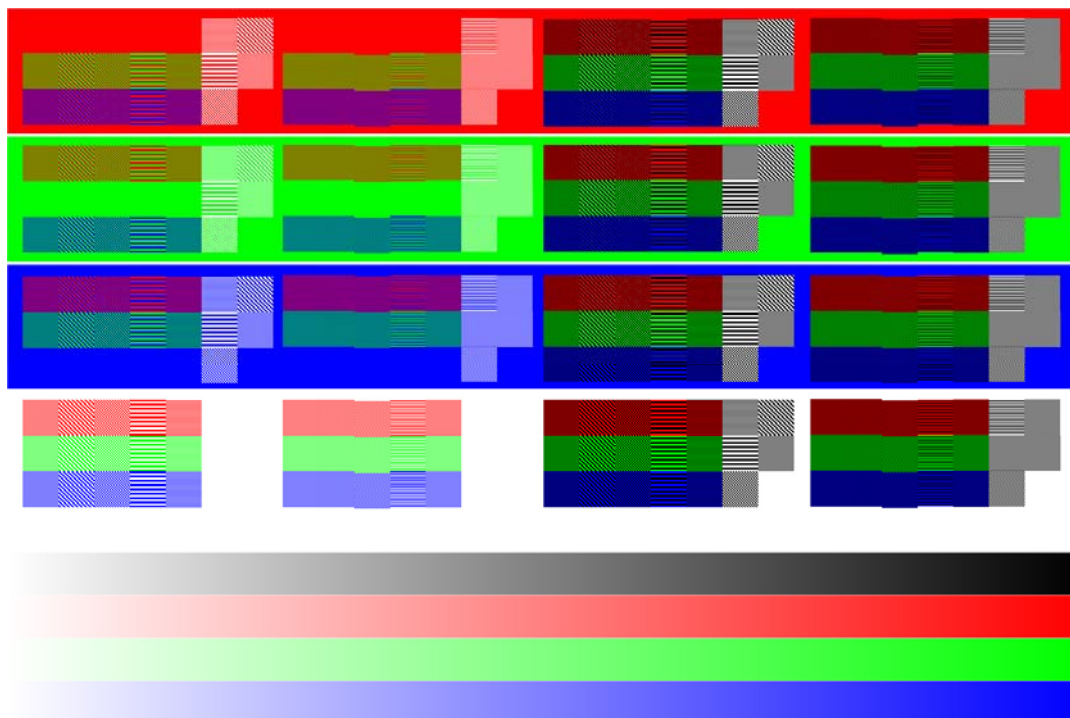
Tato komprese dokáže udržet velikost souboru CR2 v přijatelné míře. Typicky se jeho velikost drží zhruba na dvoj až dva-a-půl násobku velikosti plnobarevného obrazu v nejkvalitnější kompresi JPEG. Vezmeme-li v úvahu, že surová data mají barevnou hloubku 12 bitů na kanál vůči 8 bitům u standardního ztrátového JPEG, jedná se o velmi dobrý výsledek.

Pro dekódování surových dat v souboru CR2 stačí i omezená implementace dekodéru LS-JPEG.

3 Řešení

Tato práce se zaměřuje především na nalezení nejlepšího způsobu interpolace obrazu. Kvalita interpolace a metoda potlačení šumu mají při zpracování obrazu na výsledek největší vliv.

Jednotlivé metody interpolace popsané v teoretické části byly implementovány a testovány na různých vstupních datech. Jako vstupní data sloužily různé reálné RAW soubory, ale i matematicky vytvořené obrazce.



Obrázek 23: Uměle vytvořený testovací obraz

Dále byly testovány různé metody na potlačení šumu v obraze pořízeném digitálním fotoaparátem.

V následující části budou popsány jednotlivé (vylepšené) metody, které by měly poskytovat nejlepší kvalitu výsledného obrazu. Prvotním cílem bylo zajistit subjektivně nejlepší výstupní obraz, protože fotografie je pozorována a hodnocena člověkem. Na pozorovatele mohou velmi rušivě působit i matematicky nevýznamné vady obrazu.

Výsledný obraz ovlivňuje i pořadí operací při zpracování obrazu. Úpravy barev obrazu je dobré provádět ještě před samotnou interpolací, může pak vznikat obraz, který obsahuje více než jen 4096 (2^{12}) úrovní. To umožňuje dosahovat plynulejších barevných přechodů a potlačuje to vznik posterizace barev, například při korekcích expozice.

3.1. Navrhovaná adaptivní metoda interpolace

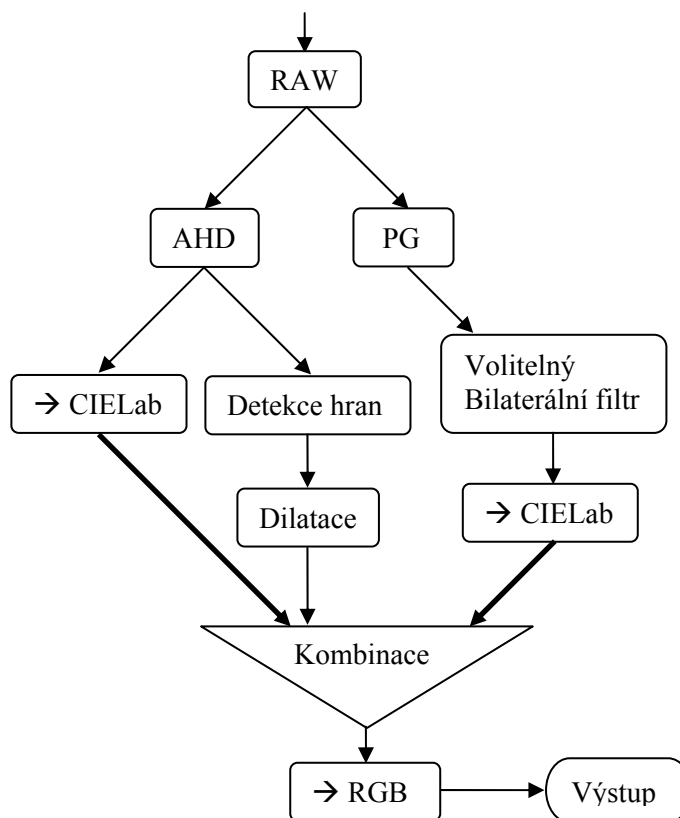
Navrhovaná metoda zavádí nový přístup k interpolaci Bayerovy mřížky. Využívá výhod jednotlivých výše popsaných metod a zároveň se snaží potlačovat jejich nevýhody a to pouze za cenu o něco vyšší výpočetní složitosti. Základní princip stojí na adaptivním přístupu k jednotlivým částem obrazu. Obecně by šlo o komplexní analýzu obsahu obrazu, ale prakticky se stačí zaměřit na specifické oblasti obrazu, které jsou pro interpolační algoritmy kritické. Nejsložitější a pro výsledný vzhled nejdůležitější je především okolí ostrých hran a jemných detailů, tedy vysoké frekvence v obrazu. U hran je kritická jejich ostrost a především výskyt interpolačních barevných artefaktů. Přestože se artefaktům nelze úplně vyhnout kvůli ztrátě informace po aplikaci Bayerovy mřížky, je snaha je potlačit i za cenu aplikace dalších filtrů na výstupní obraz. V hladkých plochách a tedy v oblastech s nízkými frekvencemi obrazového signálu je vhodné použít opačný přístup, tedy zajistit hladké barevné přechody s minimem šumu.

Navrhovaná metoda pracuje zároveň v barevných modelech RGB a Lab. Barevný model Lab má obrovskou výhodu v možnosti oddělení jasové a barevné složky, tedy lze opět jiným způsobem adaptivně přistupovat k obrazu.

Navrhovaná adaptivní interpolační metoda kombinuje upravené jednotlivé složky barevných modelů v závislosti na obsahu daného místa vstupního obrazu. Pro požadovanou analýzu obsahu obrazu se používá binární mapa. Ta je vytvořena pomocí detekce hran Sobelovým operátorem na interpolovaném obrazu převedeném do stupňů šedi. Oblasti detekovaných hran jsou následně rozšířeny pomocí matematické morfologie obrazu, aby bylo ve výsledku postiženo i okolí těchto hran a jemných detailů. Je použita dilatace obrazu s jednoduchým čtvercovým operátorem o velikosti 2×2 pixely. Obraz s rozšířenými hranami je prahován pomocí vhodně zvoleného prahu, závislém například na množství předpokládaného šumu v obraze, což závisí např. na citlivosti ISO, při které byl pořízen snímek.

Pomocí takto vytvořené binární mapy jsou zkombinovány dílčí výsledky různých interpolačních metod.

Na základě analýzy existujících interpolačních metod byly vybrány metody AHD a Pixel Grouping. V oblastech hran se používá metoda AHD, jako doposud celkově nejkvalitnější interpolační metoda. Na jednolitě hladké plochy je použita metoda Pixel Grouping, protože velmi dobře potlačuje šum obrazu a zajišťuje vysokou homogenitu pixelů hladkých ploch.



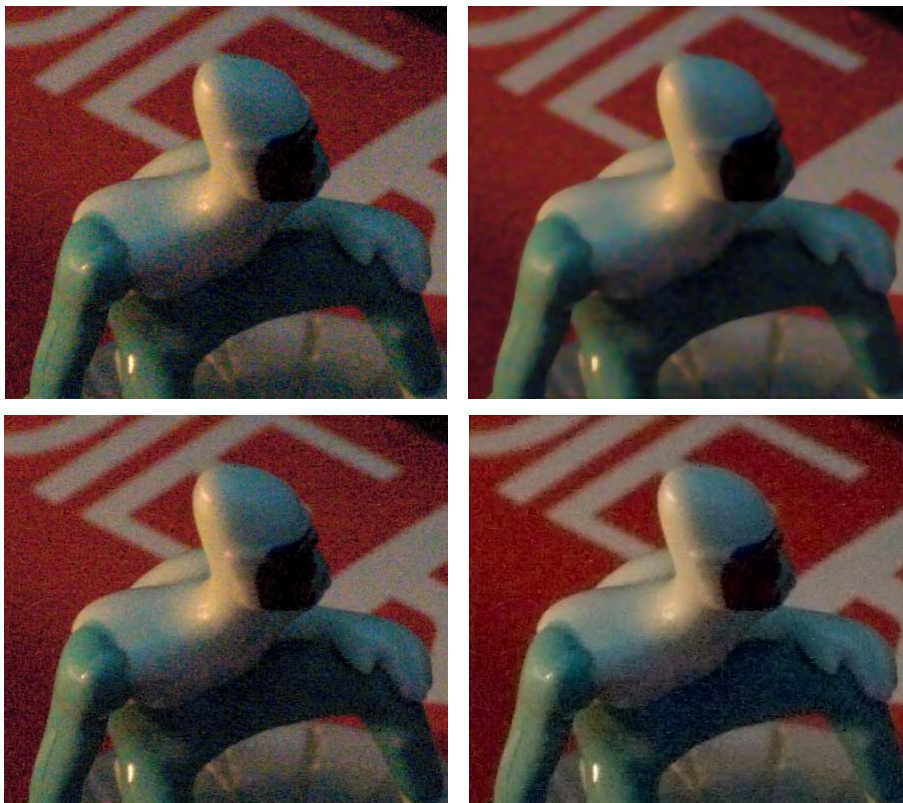
Tato metoda dává v praxi nejlepší výsledky. Čisté výstupní obrázky plné detailů a bez nežádoucích barevných artefaktů.

3.2. *Navrhovaná metoda potlačení šumu v digitální fotografii*

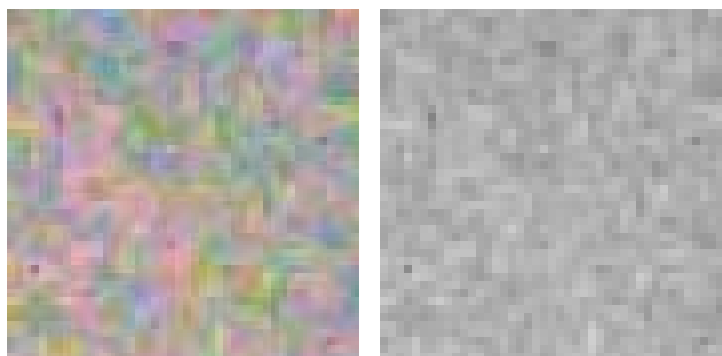
Šum v digitální fotografii má poměrně specifickou podobu. Nejde o jednotlivé pixely s falešnými (zašuměnými) hodnotami, ale spíše o shluky pixelů s falešnou barevnou informací. To se na fotografiích pořízených při vysokých citlivostech projevuje rušivými barevnými bloky. Celkové vyhlazení obrazu trochu tento jev potlačuje, ale za cenu výrazné ztráty ostrosti a detailů v obraze.

Navrhovaná metoda pracuje v barevném modelu Lab, kde pomocí mediánu vyhlazuje chrominanční kanály ab. To potlačuje výskyt shluků pixelů s falešnou barevnou informací a zároveň zachovává hrany v obraze. Jemný šum, který zůstává v jasovém L kanálu, naopak subjektivně zvyšuje ostrost výsledného obrazu a není tak na závadu.

Jedinou nevýhodou této metody je větší výpočetní náročnost, kdy je potřeba obraz převést mezi různými barevnými modely a počítat medián pro každý pixel obrazu. Po vizuální stránce se na výsledném obrazu projevují velké poloměry oblastí výpočtu mediánu, kdy dochází k mírnému tzv. vymývání barev v místech s jemnými detaily.



Obrázek 24: Porovnání šumu (původní neupravený obraz, medián všech 3 kanálů, navrhovaná metoda potlačení digitálního šumu a výsledek metody potlačení šumu v aplikaci Adobe Photoshop



Obrázek 25: Výřez z původního obrazu před a po aplikaci navrhované metody potlačení šumu

3.3. *Implementace*

Jako součást práce byly jednotlivé existující metody implementovány do jednoduché knihovny. To umožnilo porovnat jednotlivé metody nejen z hlediska kvality výstupu, ale i z hlediska rychlosti a paměťové náročnosti jednotlivých metod.

3.3.1. Knihovna cr2decode

Pro dekódování souborů CR2 a jejich převod do běžných grafických formátů je implementována knihovna v jazyce C.

3.3.2. Práce se soubory CR2 pomocí knihovny cr2decode.

Obsah souboru CR2 je čten přímo ze souboru. Nejdříve jsou načteny hlavičky a dekomprimována obrazová data. Na základě údajů z hlaviček (i z parametrů dekomprimovaných obrazových dat) jsou vypočítány parametry pro převod RAW dat na výsledný obraz, např. typ použitého barevného filtru, velikost aktivní oblasti snímacího obrazového senzoru atd.

3.3.3. Struktura knihovny cr2decode

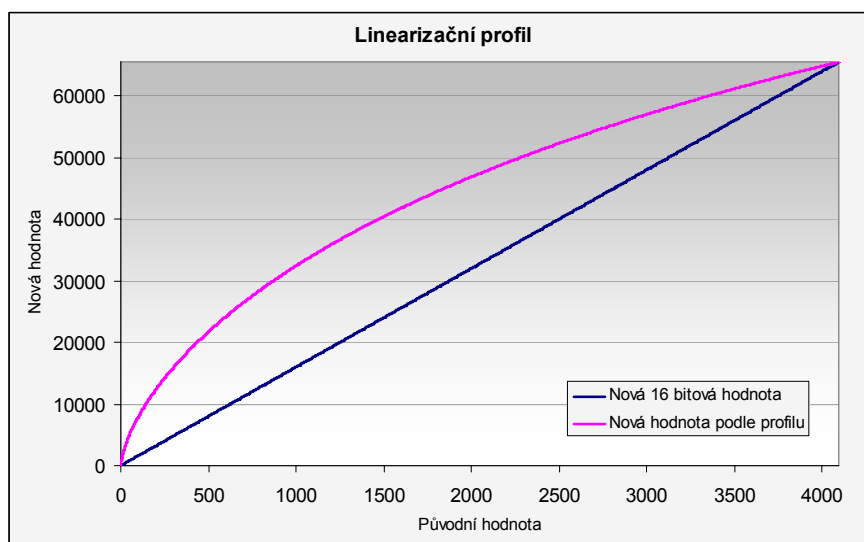
Podrobný popis knihovny se nachází v příloze této práce.

4 Vyhodnocení

V této kapitole budou prezentovány jednotlivé dosažené výsledky. Výsledky přizpůsobení citlivosti senzoru digitálního fotoaparátu citlivosti lidského oka, výsledky jednotlivých interpolačních metod obrazových dat maskovaných Bayerovou mřížkou a nakonec výsledky potlačení šumu fotografií pořízených digitálním fotoaparátem.

4.1. *Linearizace jasových hodnot*

Měření probíhala na kalibrační tabulce Gretag MacBeth Color Checker. Na základě porovnání výchozích hodnot z RAW obrazových dat a požadovaných hodnot jednotlivých polí tabulky naměřených spektrofotometrem byl sestaven linearizační profil. Pomocí tohoto profilu je při zpracovávání obrazu zajištěno věrné zobrazení obrazu, kdy velmi dobře odpovídá citlivosti lidského oka.



4.2. *Interpolace*

Hodnocení vlastností interpolačních algoritmů probíhalo na několika reálných fotografiích i na uměle vytvořených zdrojových datech, které měly prověřit jednotlivé algoritmy v kritických situacích.

Z výsledků je zřejmé, že interpolace má své omezení (nejprůkaznější to je na uměle vytvořených testovacích datech), protože maskování obrazu Bayerovou mřížkou způsobuje ztrátu informace. Jak je vidět, ta je nejmenší v zeleném kanálu, neboť buněk citlivých na zelenou barvu je na obrazovém senzoru dvakrát více než buněk citlivých na červenou nebo modrou barvu.

Původní obrázek



Červená (R)
0,00
Zelená (G)
0,00
Modrá (B)
0,00
Celkem
0,00

Vylepšená lineární metoda



Červená (R)
4,04
Zelená (G)
2,95
Modrá (B)
3,00
Celkem
3,81

Pixel Grouping



Červená (R)
6,90
Zelená (G)
2,03
Modrá (B)
6,31
Celkem
5,33

AHD



Červená (R)
2,25
Zelená (G)
1,80
Modrá (B)
2,42
Celkem
2,19

FLEDIC



Červená (R)
2,25
Zelená (G)
1,80
Modrá (B)
2,42
Celkem
2,19

V této reálné scéně poskytuje nejlepší výsledky metoda AHD a navrhovaná interpolační metoda, která v oblastech bohatých na detaily staví hlavně na metodě AHD. Nepochází u ní k tvorbě barevných interpolačních artefaktů. Naopak nejhůře zde dopadla metoda Pixel Grouping, která těchto rušivých barevných artefaktů vytvořila nejvíce.

Dále je vidět, že nejvíc informace nese zelený (jasový) kanál.

Původní obrázek



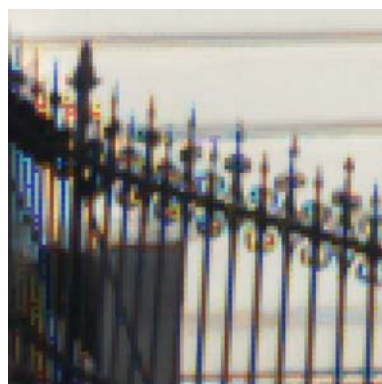
Červená (R)
0,00
Zelená (G)
0,00
Modrá (B)
0,00
Celkem
0,00

Vylepšená lineární metoda



Červená (R)
10,03
Zelená (G)
5,79
Modrá (B)
9,82
Celkem
8,92

Pixel Grouping



Červená (R)
12,45
Zelená (G)
6,71
Modrá (B)
15,54
Celkem
12,39

AHD



Červená (R)
4,61
Zelená (G)
3,10
Modrá (B)
4,21
Celkem
4,07

FLEDIC



Červená (R)
4,61
Zelená (G)
3,10
Modrá (B)
4,21
Celkem
4,07

Na tomto obrázku selhávají obě lineární metody. Výskyt rušivých barevných artefaktů je u těchto metod velmi výrazný. U vylepšené lineární metody je také velmi dobře vidět na ostrých hranách tzv. zipper efekt. Ten je způsoben pravidelností Bayerovy mřížky a díky tomu nedostatečnou informací z okolních buněk pro interpolaci (téměř) vodorovných a svislých hran.

Opět je zřejmé, že v reálné scéně nese nejvíce informace zelený kanál.

Původní obrázek



Červená (R)
0,00
Zelená (G)
0,00
Modrá (B)
0,00
Celkem
0,00

Vylepšená lineární metoda



Červená (R)
4,28
Zelená (G)
2,62
Modrá (B)
4,31
Celkem
3,85

Pixel Grouping



Červená (R)
6,31
Zelená (G)
2,58
Modrá (B)
5,83
Celkem
5,28

AHD



Červená (R)
3,57
Zelená (G)
1,93
Modrá (B)
2,94
Celkem
2,92

FLEDIC



Červená (R)
3,57
Zelená (G)
1,93
Modrá (B)
2,94
Celkem
2,92

Obě lineární metody vytvářejí tzv. zipper efekt, i když každá v jiné podobě. Obraz vytvořený vylepšenou lineární metodou ho obsahuje systematicky, ale ne tolik výrazně. Metoda Pixel Grouping naopak zipper efekt netvoří tak často, ale zato je tento nežádoucí efekt daleko výraznější.

Navrhovaná interpolační metoda a metoda AHD (každá trochu jinak) podávají výsledky velmi podobné originálnímu obrazu.

Původní obrázek



Červená (R)
0,00
Zelená (G)
0,00
Modrá (B)
0,00
Celkem
0,00

Vylepšená lineární metoda



Červená (R)
0,72
Zelená (G)
0,47
Modrá (B)
0,73
Celkem
0,67

Pixel Grouping



Červená (R)
0,76
Zelená (G)
0,56
Modrá (B)
0,77
Celkem
0,72

AHD



Červená (R)
0,82
Zelená (G)
0,63
Modrá (B)
0,75
Celkem
0,74

FLEDIC



Červená (R)
0,70
Zelená (G)
0,51
Modrá (B)
0,71
Celkem
0,64

Na hladkých plochách se plně projevují vlastnosti navrhované interpolační metody. Zde podává nejlepší výsledky, lepší nežli obě lineární metody. Metoda AHD má v tomto druhu scény nejhorší obraz ze všech testovaných metod.

Opět je zde vidět, že nejvíce informace nese zelený kanál obrazu.

Původní obrázek



Červená (R)
0,00
Zelená (G)
0,00
Modrá (B)
0,00
Celkem
0,00

Vylepšená lineární metoda



Červená (R)
4,02
Zelená (G)
2,45
Modrá (B)
3,67
Celkem
3,51

Pixel Grouping



Červená (R)
5,37
Zelená (G)
2,85
Modrá (B)
6,71
Celkem
5,34

AHD



Červená (R)
3,29
Zelená (G)
2,16
Modrá (B)
3,08
Celkem
2,92

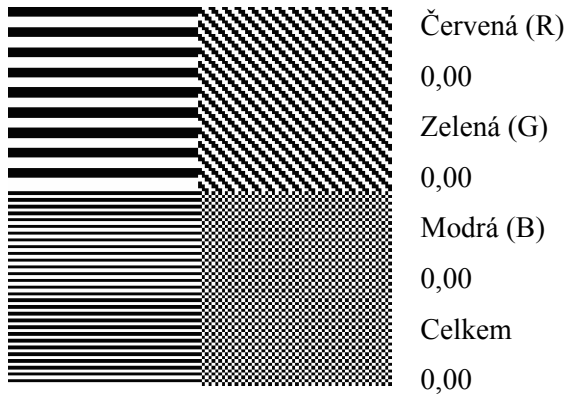
FLEDIC



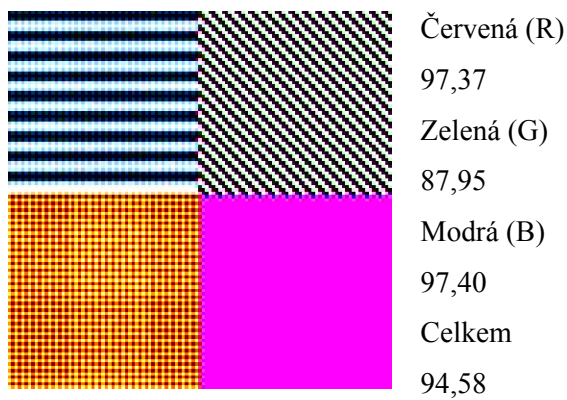
Červená (R)
3,57
Zelená (G)
2,51
Modrá (B)
3,36
Celkem
3,22

Zde podává nejlepší výsledky metoda AHD. Za ní následuje navrhovaná metoda a po ní vylepšená lineární metoda. S odstupem nejhorší je metoda Pixel Grouping.

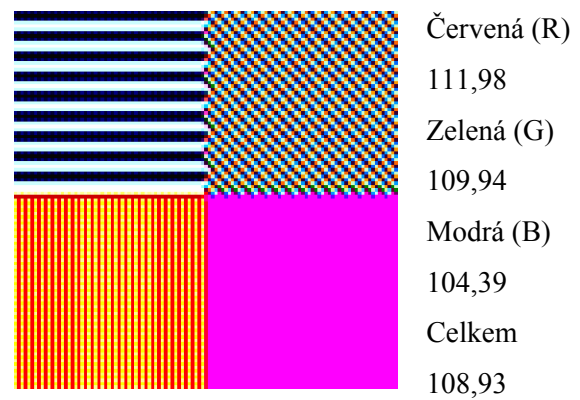
Původní obrázek



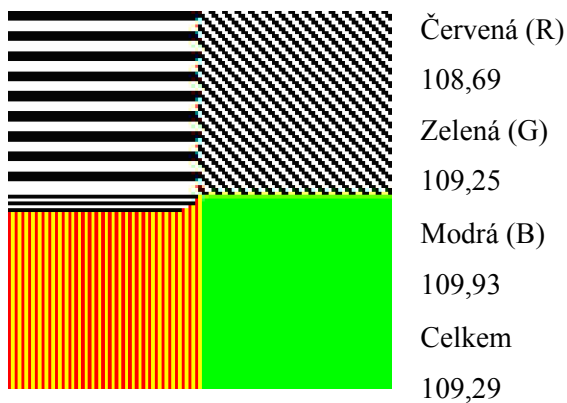
Vylepšená lineární metoda



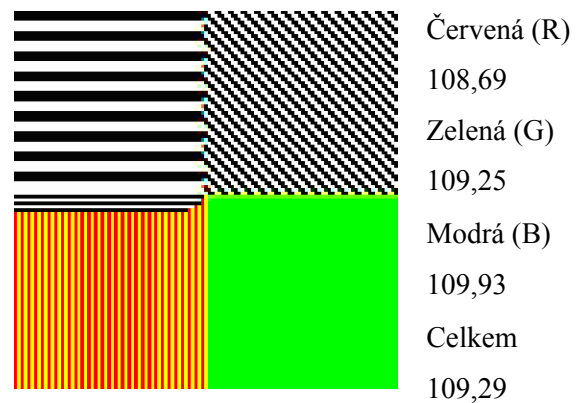
Pixel Grouping



AHD



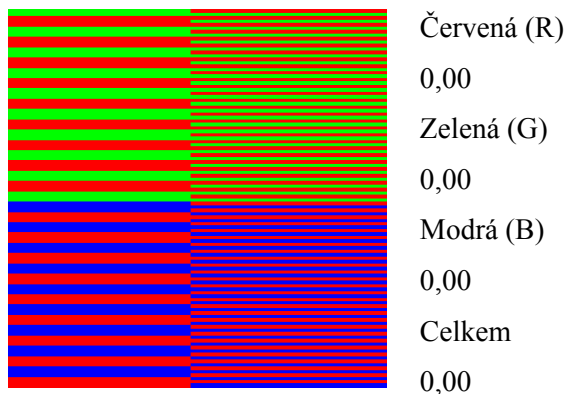
FLEDIC



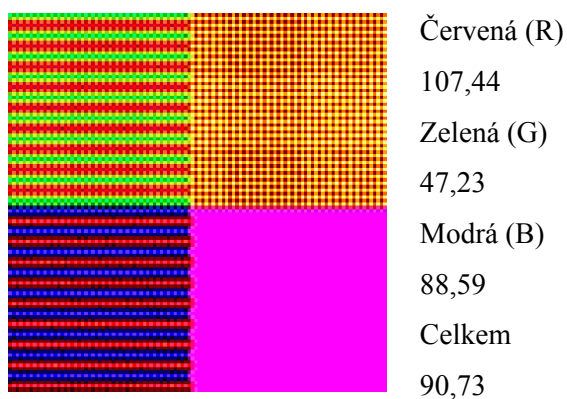
Uměle připravená testovací data plně ukazují na omezení interpolace v důsledku ztráty obrazové informace po aplikaci Bayerovy mřížky.

Přestože číselně nejhorší výsledek zde dává navrhovaná lineární metoda spolu s metodou AHD, tak po subjektivním hodnocení výsledného obrazu jsou tyto metody nejvěrnější originálu.

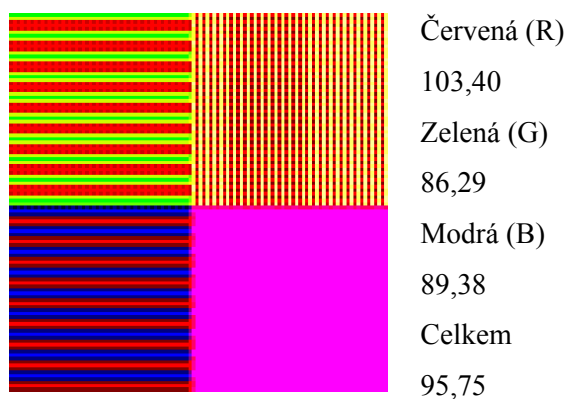
Původní obrázek



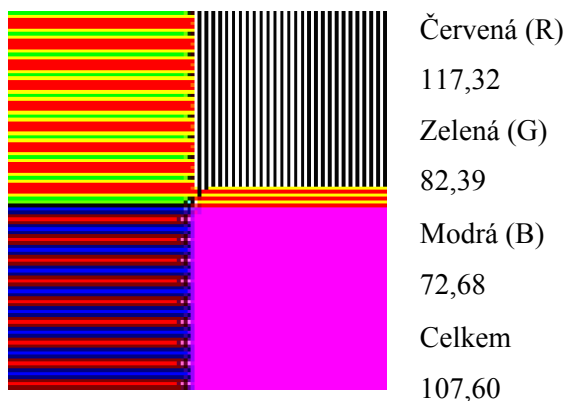
Vylepšená lineární metoda



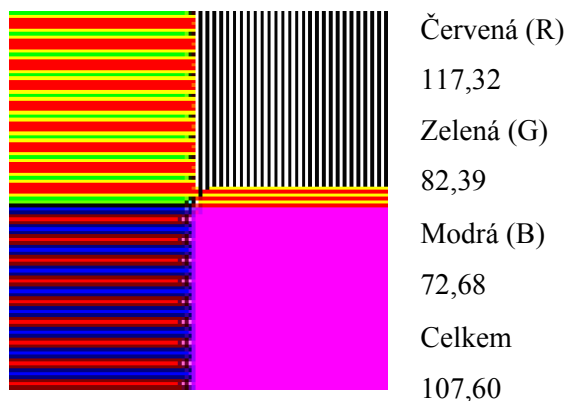
Pixel Grouping



AHD



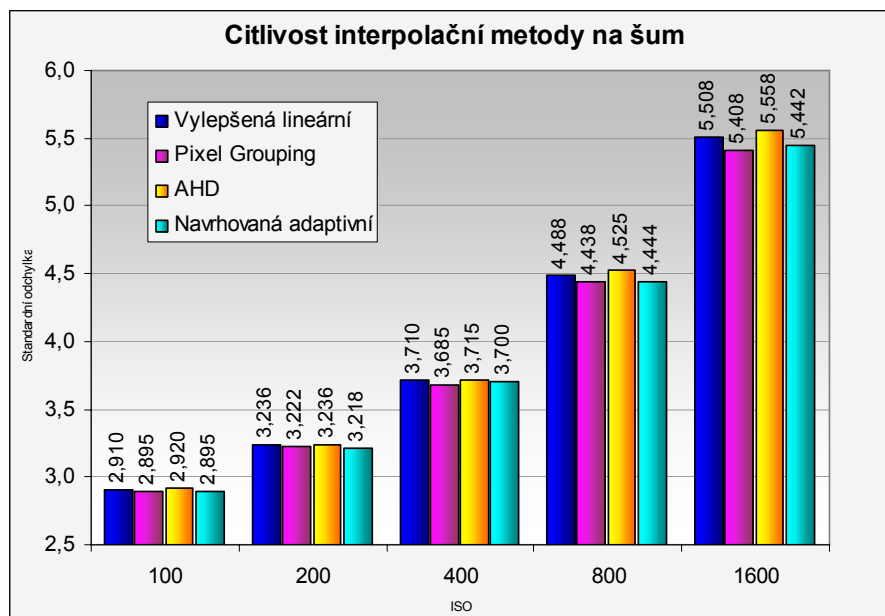
FLEDIC



Na tomto testovacím obrázci opět selhávají všechny metody. Zde není jasný vítěz ani poražený. Výsledné číselné hodnocení odchylky vůči originálnímu obrazu je zavadějící.

4.3. Míra šumu

Jednotlivé interpolační metody mírně ovlivňují výskyt šumu ve výsledném obraze. Nicméně rozdíly mezi jednotlivými metodami jsou v číselném vyjádření minimální.

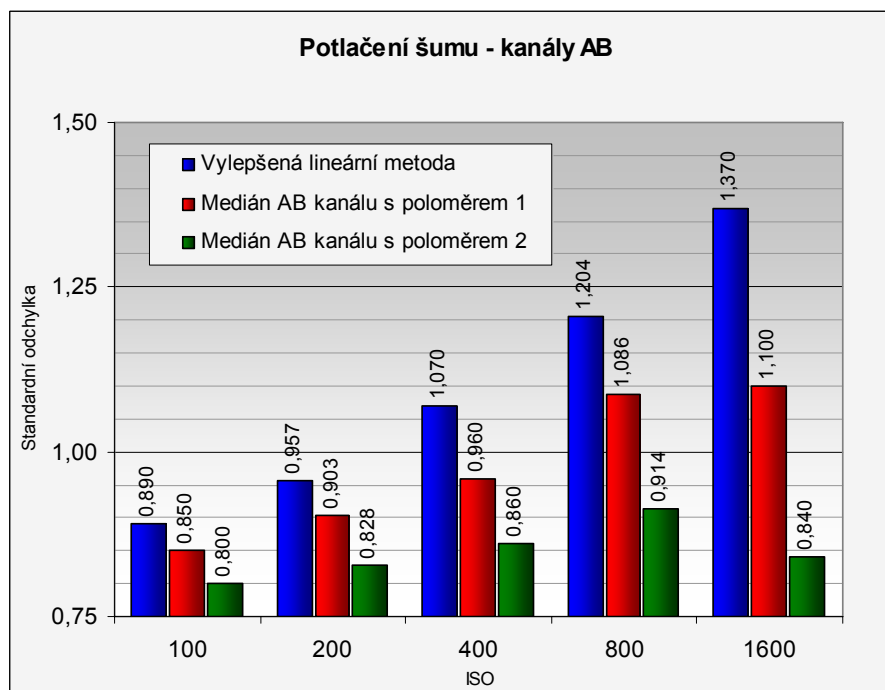
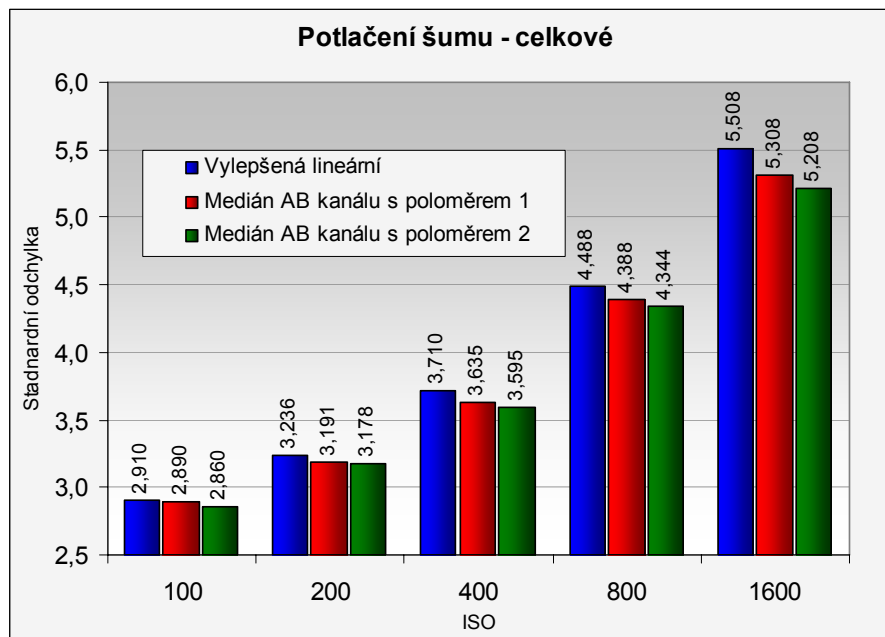


Pokud se podíváme na strukturu výsledného šumu, jsou mezi interpolačními metodami vidět větší rozdíly. Nejjemnějšího (a pro oko nejpříjemnějšího) šumu dosahuje vylepšená lineární metoda. Naopak nejvíce rušivě působí šum u metody AHD, kde jsou v obraze ostře ohraničené svislé a vodorovné shluky pixelů.



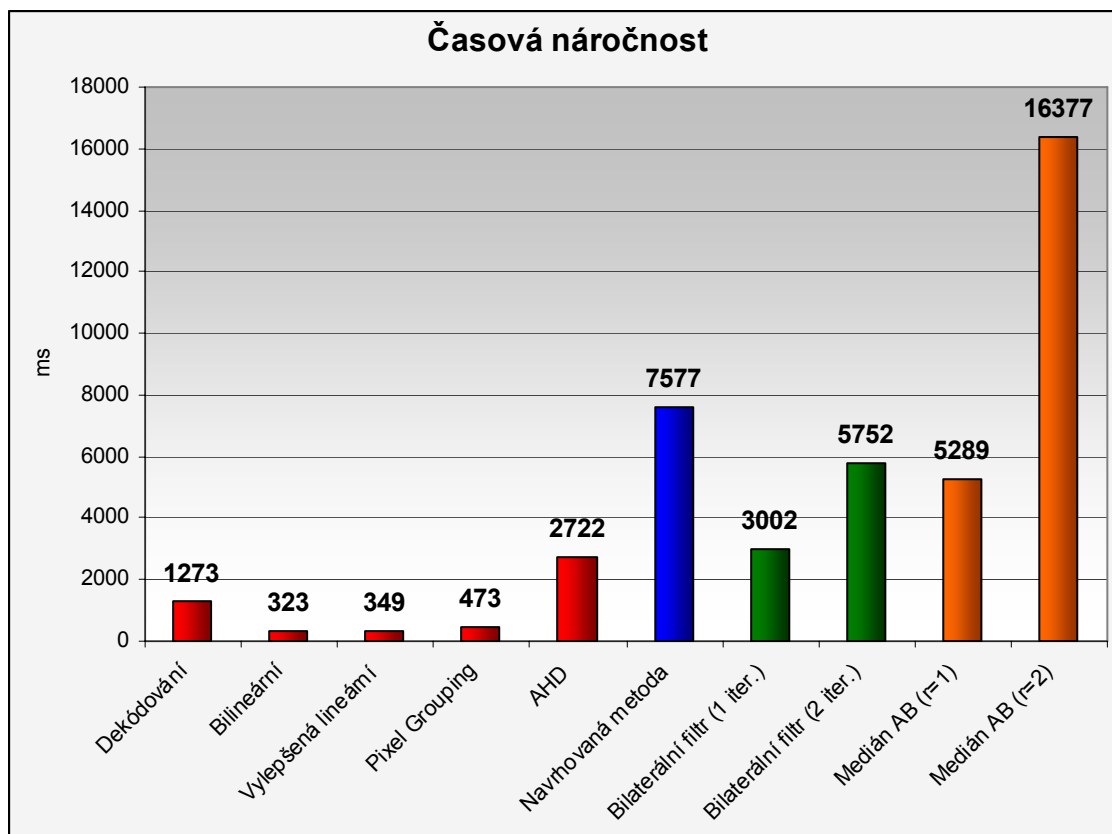
Obrázek 26: Šum ve výsledném obraze vytvořeném metodou Pixel Grouping, vylepšenou lineární metodou, metodou AHD a aplikace navrhované metody pro potlačení šumu

Pokud použijeme navrhovanou metodu potlačení barevného šumu, tak jsou už vidět výraznější rozdíly. Ovšem, pokud vezmeme v úvahu množství šumu v barevných kanálech ab barevného modelu Lab, tak je jasně vidět, že potlačení barevného šumu je velmi účinné.



4.4. Časová náročnost operací

V následujícím grafu je zachycena časová náročnost jednotlivých operací. Měření probíhalo na několika různých 4 Mpix RAW obrazových souborech.



Jak je vidět, interpolační metody Pixel Grouping a obě lineární metody jsou jasně nejrychlejší a hodí se tedy i pro okamžité náhledy RAW obrazových dat.

Časově nejnáročnější je navrhovaná interpolační metoda. Je to v důsledku několikanásobného převodu mezi barevnými modely. Zde je možnost urychlit výpočet při využití SIMD (Simple Instruction Multiple Data) instrukcí v plovoucí desetinné čárce, např. SSE.

5 Závěr

Tato práce se zabývá zpracováním surových obrazových dat (RAW) z digitálních fotoaparátů. Jedná se o zpracování digitalizovaného výstupu přímo ze snímacího čipu. To zajišťuje nejvyšší možnou kvalitu výstupního obrazu.

V dokumentu jsou popsány obecné postupy pro převod RAW obrazových dat. Různé interpolační metody, metody úpravy barev a další (doplňkové) metody pro úpravu obrazu.

Na základě analýzy současných používaných metod interpolace obrazu maskovaného Bayerovou mřížkou byl navržen nový algoritmus interpolace, který podává nejlepší výsledky. Hlavní důraz byl kladen na subjektivně nejvyšší kvalitu obrazu, protože výstupy jsou pozorované především člověkem.

Dále byla navržena metoda pro účinné potlačení typického šumu v obrazu, který vzniká při expozici digitálním fotoaparátem při vysoké citlivosti. Opět bylo jako hlavní měřítko kvality použito subjektivní hodnocení výsledného obrazu.

Navržené metody byly implementovány za účelem praktického ověření výsledků a možnosti provedení měření rychlosti jednotlivých navrhovaných metod v porovnání k již existujícím metodám.

Adaptivní přístup k obsahu obrazu lze využít různými způsoby a zde je prostor pro další vylepšování, zejména metody interpolace.

Literatura

- [1] Henrique S. Malvar, Li-wei He a Ross Cutler: High-Quality Linear Interpolation For Demosaicing of Bayer-Patterned Color Images. One Microsoft Way, Redmond, WA 98052, US, 2004
- [2] Keigo Hirakawa, Thomas W. Parks: Adaptive Homogeneity-Directed Demosaicing Algorithm. IEEE Transactions On Image Processing, Vol. 14, No. 3, březen 2005
- [3] Adobe Developers Association: TIFF Revision 6.0. 3. 6. 1992
- [4] JPEG Standard. JPEG ISO/IEC 10918-1 ITU-T Recommendation T.81, 1993
- [5] Žára, J., Beneš, B., Felker, P.: Moderní počítačová grafika. Computer Press, 1998.
- [6] Hlaváč, V.: Zpracování signálů a obrazů. Praha, ČVUT, 2005.
- [7] Rafael C. Gonzalez, Richard E. Woods: Digital image processing. Prentice-Hall, 2002.
- [8] Technical Standardization Committee on AV & IT Storage Systems and Equipment: Exchangeable image file format for digital still cameras: Exif Version 2.2. JEITA CP-3451, 2002
- [9] Bruce Fraser, Chris Murphy, Fred Bunting: Správa barev. Computer Press, ISBN: 80-722-6943-7, 2003
- [10] J. Adams, K. Parulski, and K. Spaulding: Color processing in digital cameras. IEEE Micro, vol. 18, no. 6, str. 20–31, 1998.
- [11] R. Ramanath, W. E. Snyder, G. L. Bilbro, and W. A. S. III: Demosaicing methods for bayer color arrays. Journal of Electronic Imaging, vol. 11, no. 3, str. 306–315, červenec 2002.
- [12] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau: Color Filter Array Interpolation in Single-Chip Digital Cameras. Draft for the IEEE SPM Special Issue on Color Image Processing
- [13] David A. Huffman: A Method for the Construction of Minimum-Redundancy Codes. PROCEEDINGS OF THE I.R.E., září 1952
- [14] http://en.wikipedia.org/wiki/Raw_image_format (prosinec 2006)
- [15] <http://en.wikipedia.org/wiki/CMOS> (prosinec 2006)
- [16] http://en.wikipedia.org/wiki/Charge-coupled_device (prosinec 2006)
- [17] <http://cybercom.net/~dcoffin/dcraw/> (prosinec 2006)
- [18] http://en.wikipedia.org/wiki/Curves_%28tonality%29 (prosinec 2006)
- [19] http://www.brucelindbloom.com/Equ_RGB_to_XYZ.html (květen 2007)
- [20] http://www.brucelindbloom.com/Equ_XYZ_to_Lab.html (květen 2007)
- [21] http://www.brucelindbloom.com/Equ_Lab_to_XYZ.html (květen 2007)
- [22] http://www.brucelindbloom.com/Equ_XYZ_to_RGB.html (květen 2007)

Přílohy

Formát CR2

Soubor je definován jako struktura 8 bitových bajtů číslovaných od 0 do N. Největší velikost je tak teoreticky 2^{32} bajtů. V současné době se největší soubory CR2 vyskytují ve velikostech do 25 MB.

Hlavička souboru CR2

Soubor začíná hlavičkou čítající 10 bajtů. Ta obsahuje identifikátor souboru a ukazatel na první tzv. *IFD (Image File Directory)*.

Struktura hlavičky

Bajt č.	0	1	2	3	4	5	6	7	8	9
Hex. Hodnota	49h	49h	2Ah	0h	10h	0h	0h	0h	43h	52h
Tisknutelný znak	I	I	*						C	R

První IFD začíná vždy na 16. bajtu ($n = 15$).

IFD

Image File Directory se skládá z 2 bajtové informace o počtu jednotlivých záznamů, 12 bajtových záznamů a 4 bajtového ukazatele na další IFD.

Struktura IFD:

Bajty č.	
0-1	Počet (x) záznamů IFD
2-13	Záznam IFD
...	Další záznamy
$2+(x \cdot 12)$	Ukazatel na další IFD

Záznam IFD

Záznam IFD (IFD Entry) má velikost 12 bajtů. První dva bajty tvoří značka identifikující pole. Další dva bajty udávají datový typ položky. Další 4 bajty udávají počet uložených hodnot daného typu a v posledních 4 bajtech je uložen ukazatel (pozice v souboru), kde je v souboru uložena daná hodnota. Předpokládá se zarovnání ofsetu na 16 bitů, tedy že to bude sudá hodnota. Ukazatel může ukazovat kamkoliv v souboru. Pokud by byla velikost uložené hodnoty do 4 bajtů, bude tato uložena přímo místo ukazatele (kvůli úspoře času a prostředků).

Struktura záznamu IFD:

Bajty č.	
0-1	Značka identifikující položku
2-3	Datový typ
4-7	Počet uložených hodnot
8-11	Ukazatel na uložená data nebo přímo hodnota

Datové typy:

Hodnota datového typu	Název datového typu	Datový typ
1	byte	8 bitové bezznaménkové celé číslo
2	ascii	8 bitový bajt, obsahující 7 bitový ASCII kód, poslední znak řetězce musí mít hodnotu 0
3	short	16 bitové bezznaménkové celé číslo
4	long	32 bitové bezznaménkové celé číslo
6	sbyte	8 bitové znaménkové celé číslo
8	sshort	16 bitové znaménkové celé číslo
9	slong	32 bitové znaménkové celé číslo

Identifikátory položek:

Hodnota identifikátoru	Název položky	Význam položky
256	ImageWidth	Šířka obrazu v pixelech
257	ImageHeight	Výška obrazu v pixelech
259	Compression	Použitá komprese obrazových dat
271	MakerNote	Výrobce přístroje
272	ModelNote	Model přístroje
273	StripOffsets	Začátek obrazových dat
274	Orientation	Orientace 1 = 0° 3 = 180° 6 = 90° po směru hodinových ručiček 8 = 90° proti směru hodinových ručiček
279	StripByteCounts	Velikost obrazových dat

LS-JPEG

Na začátku datového proudu bezztrátového JPEG, je uložena hlavička a dále Huffmanovy tabulky použité při kompresi. Z tabulek sestrojíme strom, který nám pomůže při dekodování. Následně čteme po jednotlivých bitech datový proud a dekodujeme data.

Hlavička

Bajty č.	
0-1	Identifikátor LS-JPEG. Dva bajty s hodnotou FFh a D8h.
2-3	Identifikátor pole
4-5	Délka pole v bajtech

Identifikátor pole

Hodnota identifikátoru	Význam
FFC3h	Hodnoty pole obsahuje informace o obrazu. Jeho šířku, výšku, barevnou hloubku, počet řádků komprimovaných najednou...
FFC4h	Pole obsahuje data potřebná k vytvoření Huffmanových tabulek.
FFDAh	Ukončující pole
FFDDh	Restart hodnota

LS-JPEG Data

Dále jsou v datovém proudu uložena komprimovaná obrazová data. Ta je nutno číst po jednotlivých bitech a postupně dekomprimovat.

Knihovna cr2decode

Dostupné funkce:

```
int get_cr2_info(*file_name, *make, *model, *preview_width,  
                *preview_height, *preview_offset, *preview_size, *width,  
                *height, *offset, *size);
```

Tato funkce ověří strukturu souboru se jménem *file_name* a načte základní informace z hlaviček:

- *make* – jméno výrobce přístroje
- *model* – jméno modelu přístroje
- *preview_width* – šířka náhledového obrázku
- *preview_height* – výška náhledového obrázku
- *preview_offset* – pozice obrazových dat vloženého náhledového obrázku v souboru cr2
- *preview_size* – velikost obrazových dat náhledového obrázku v bajtech
- *width* – šířka surových obrazových dat v pixelech
- *height* – výška surových obrazových dat v pixelech
- *offset* – pozice surových obrazových dat v souboru cr2
- *size* – velikost surových obrazových dat v bajtech

```
int get_cr2_data(*file_name, offset, size, *data);
```

Tato funkce načte jakákoliv data (začínající na pozici *offset* a mající velikost *size*) ze souboru se jménem *file_name* do proměnné *data*.

```
int get_cr2_raw_params(*file_name, offset, *model, *raw_width,  
                      *raw_height, *top_margin, *left_margin, *width,  
                      *height, *min_abs, *max_abs, *filters);
```

Tato funkce zjistí nebo vypočítá parametry potřebné pro převod surových obrazových dat:

- *file_name* – jméno souboru
- *offset* – pozice surových obrazových dat v souboru
- *model* – model digitálního přístroje
- *raw_width* – šířka surových obrazových dat v pixelech
- *raw_height* – výška surových obrazových dat v pixelech
- *top_margin* – levý okraj, pro ořezání nevyužitých surových obrazových dat
- *left_margin* – horní okraj, pro ořezání nevyužitých surových obrazových dat
- *width* – šířka výsledného obrazu v pixelech
- *height* – výška výsledného obrazu v pixelech
- *min_abs* – minimální hodnota vyskytující se v surových obrazových datech

- `max_abs` – maximální hodnota vyskytující se v surových obrazových datech
- `filters` – konfigurace barevných filtrů snímacího čipu

```
int decode_cr2_raw_data(*file_name, offset, raw_width, raw_height,  
                       top_margin, left_margin, width, height, filters,  
                       *min_abs, *min_absh, *max_abs, (*data)[4]);
```

Tato funkce dekóduje surová obrazová data (výsledný obraz, z jednotlivých barevných filtrů, v proměnné `data`). Vstupní parametry odpovídají výstupu funkce `get_cr2_raw_params`. Proměnná `min_absh` obsahuje minimální dekódovanou hodnotu z části senzoru, který se neúčastní tvorby obrazu (slouží pro detekci černé barvy). Proměnná `min_abs` obsahuje minimální dekódovanou hodnotu. Proměnná `max_asb` obsahuje maximální dekódovanou hodnotu.

```
int int scale_raw((*data)[4], width, height, gamma[4], *gamma_rgb, mul[4],  
                 min_abs, max_abs);
```

Tato funkce přepočítá barvy v obrazových datech RGB a aplikuje barevné křivky. Podle hodnoty `gamma` pro různé barevné kanály dochází k jedné z následujících operací:

`gammar = gammag = gammab = 1` dochází jen k vynásobení jednotlivých kanálů proměnnou `mul`

`gammar = gammag = gammab = 0` dochází k výpočtu korekční křivky v poměru jednotlivých kanálů dle proměnné `mul`

jinak dochází k výpočtu gamma křivky dle hodnoty `gamma`

```
int pre_scale_raw((*data)[4], width, height, gamma[4], *gamma_rgb, mul[4],  
                 min_abs, max_abs, filters);
```

Tato funkce pracuje stejně jako `scale_raw`, ale pracuje se vstupními daty maskovanými Bayerovou mřížkou.

```
void debayerize_linear_raw((*data) [4], width, height, min_absrh, max_abs,  
                           clipv, filters);
```

Tato funkce provede barevnou bilineární interpolaci obrazových dat podle příslušné konfigurace barevných filtrů.

```
void debayerize_elinear_raw((*data) [4], width, height, min_absrh,  
                            max_abs, clipv, filters);
```

Tato funkce provede barevnou vylepšenou lineární interpolaci obrazových dat podle příslušné konfigurace barevných filtrů.

```
void debayerize_ahd_raw((*data) [4], width, height, min_absrh, max_abs,
                        clipv, filters, *model);
```

Tato funkce provede barevnou interpolaci obrazových dat metodou AHD. Oproti lineárním metodám se navíc předává parametr s názvem modelu fotoaparátu, protože se pracuje s různými barevnými modely i prostory.

```
void debayerize_pgrouping_raw((*data) [4], width, height, min_absrh,
                              max_abs, clipv, filters);
```

Tato funkce provede barevnou interpolaci obrazových dat metodou Pixel Grouping.

```
void debayerize_fledic_raw((*data) [4], width, height, min_absrh, max_abs,
                           clipv, t, filters, *model);
```

Tato funkce provede barevnou interpolaci obrazových dat navrhovanou metodou. Oproti lineárním metodám se navíc předává parametr s názvem modelu fotoaparátu, protože se pracuje s různými barevnými modely i prostory, a parametr určující hodnotu prahu při tvorbě binární kombinační mapy.

```
void auto_wb((*data) [4], width, height, *r, *g, *b);
```

Tato funkce hledá poměr jednotlivých barevných kanálů, aby byla vyvážená bílá barva. Pracuje na principu průměrkování celého obrazu na šedou barvu. V proměnných r , g , b je poměr pro vyvážení jednotlivých kanálů.

```
void pre_auto_wb((*data)[4], width, height, *r, *g, *b, filters);
```

Tato funkce pracuje stejně jako `auto_wb`, ale pracuje se vstupními daty maskovanými Bayerovou mřížkou.

```
void auto_wb_h((*data)[4], width, height, *r, *g, *b);
```

Tato funkce vyvažuje bílou barvu na základě hledání nejsvětějšího bodu obrazu (neberou se v úvahu přepálená místa). V proměnných r , g , b je poměr pro vyvážení jednotlivých kanálů.

```
void pre_auto_wb_h((*data)[4], width, height, *r, *g, *b, filters);
```

Tato funkce pracuje stejně jako `auto_wb_h`, ale pracuje se vstupními daty maskovanými Bayerovou mřížkou.

```
void stretch_histogram((*data) [4], width, height, gamma[4], *gamma_rgb);
```

Tato funkce roztáhne histogram pro každý barevný kanál zvlášť. Je to další způsob vyvážení bílé barvy. Podle hodnoty $gamma$ pro různé barevné kanály dochází k jedné z následujících operací:

$gamma_r = gamma_g = gamma_b = 1$ dochází k lineárnímu roztažení histogramu dle detekovaného minima a maxima separátně pro každý barevný kanál

$gamma_r = gamma_g = gamma_b = -1$ dochází pouze k aplikaci LUT v proměnné *gamma_rgb*
jinak dochází k roztažení histogramu pomocí gamma křivky

```
void pre_stretch_histogram((*data) [4], width, height, gamma[4],
                           *gamma_rgb, filters);
```

Tato funkce roztáhne histogram pro každý barevný kanál zvlášť. Je to další způsob vyvážení bílé barvy. Tato funkce pracuje se vstupními daty maskovanými Bayerovou mřížkou.

```
void RGBImage2LabImage((*rgb)[4], width, height, (*lab)[3]);
```

Tato funkce převádí celý obraz z barevného modelu RGB na barevný model Lab. U vstupního barevného modelu RGB se předpokládá barevný prostor sRGB a u výstupního modelu Lab se předpokládá standardní osvětlení D65.

```
void LabImage2RGBImage((*lab)[3], width, height, (*rgb)[4]);
```

Tato funkce převádí celý obraz z barevného modelu Lab na barevný model RGB. U modelu Lab se předpokládá standardní osvětlení D65 a u barevného modelu RGB pak barevný prostor sRGB.

```
void remove_noise_med((*data)[4], width, height, pab);
```

Tato funkce potlačuje šum v obraze pořízeném digitálním fotoaparátem. Filtruje rozdílové barevné kanály ab v barevném modelu Lab.

```
void bilateral_filter((*data)[4], width, height, clipv, iter);
```

Tato funkce potlačuje barevné interpolační artefakty.

```
int read_lin_table(*filename, *gamma_rgb);
```

Tato funkce načte linearizační barevný profil ze souboru do LUT. Předpokládá se využití při nestandardní práci s linearizačním profilem.

```
void lin_image((*data)[4], width, height, *model, *path, *gamma_rgb,
              filters);
```

Tato funkce provede úpravu barev obrazu dle linearizačního profilu daného názvem modelu fotoaparátu.