



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

DIGITÁLNÍ FILTRY PRO OBRAZOVÁ DATA

DIGITAL FILTERS FOR IMAGE PROCESSING

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

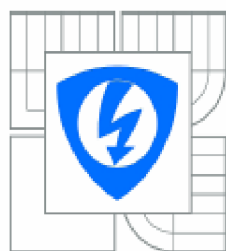
AUTOR PRÁCE
AUTHOR

VÁCLAV VŠETIČKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR ČÍKA, Ph.D.

BRNO 2010



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Václav Všetička

ID: 74610

Ročník: 3

Akademický rok: 2009/2010

NÁZEV TÉMATU:

Digitální filtry pro obrazová data

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte běžně používané digitální filtry pro úpravu fotografií. Vyberte alespoň 8 z nich, detailně je popište a navrhnete aplikaci, ve které budou tyto filtry aplikovány. Navrženou aplikaci realizujte v programovacím jazyku JAVA.

DOPORUČENÁ LITERATURA:

[1] DOBEŠ, Michal. Zpracování obrazu a algoritmy v C#. Praha : BEN, 2008., s. 144. ISBN 978-80-7300-233-6.

[2] ŘÍHA, Kamil. Pokročilé techniky zpracování obrazu. Brno : VUT v Brně, 2007. 126 s.

Termín zadání: 29.1.2010

Termín odevzdání: 2.6.2010

Vedoucí práce: Ing. Petr Číka, Ph.D.

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ANOTACE

Práce se zabývá filtračními metodami pro úpravu digitálního obrazu. Kdy je potřeba vhodným algoritmem zvýraznit požadovanou informaci z obrazu, nebo naopak potlačit rušivé elementy v obraze. Práce popisuje několik nejzásadnějších filtračních úprav obrazu, které mají nesporné využití v technické i lékařské praxi. Výstupem práce je implementace některých filtrů do prostředí JAVA.

Klíčová slova:

Barevný prostor, RGB, HSV, CCD, CMOS, filtrace obrazu, negativ, gama korekce, histogram, vyhlazování obrazu, medián, Sobelův operátor, šum

ABSTRACT

This work is dedicated to methods for editing the digital image. When you need an suitable algorithm to highlight the desired information from the image, or conversely, suppress disturbing elements in the picture. This work describes a few most basic image editing filters to use common in technical and medical practice. The result of this work is implementation filters into Java.

Keywords:.

color space, RGB, HSV, CCD, CMOS, image filter, negative picture, gamma correction, histogram, image smoothing, median filter, Sobel operator, noise

Citace:

Václav Všetická: Digitální filtry pro obrazová data, bakalářská práce, Brno:FEKT VUT

Digitální filtry pro obrazová data

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma „Digitální filtry pro obrazová data“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....
podpis autora

Poděkování

Děkuji vedoucímu práce Ing. Petru Číkovi, Ph.D. za odborné vedení mé práce a pomoc.

V Brně dne

.....
podpis autora

Obsah

OBSAH.....	1
ÚVOD.....	2
1 SVĚTLO A BARVA.....	2
1.1 VNÍMÁNÍ BAREV.....	2
1.2 VNÍMÁNÍ SVĚTLA LIDSKÝM.....	3
2 BAREVNÉ PROSTORY.....	4
2.1 BAREVNÉ PROSTORY RGB.....	4
2.2 PŘEVOD BAR OBRAZU.....	4
2.3 MODEL Y CMY A CMYK.....	5
2.4 MODEL HSV.....	5
3 SNÍMÁNÍ A ZPRACOVÁNÍ OBRAZU.....	7
3.1 CMOS SENZORY.....	8
3.2 CCD SENZORY.....	8
3.3 ŠUM V OBRAZE.....	9
3.4 ZPRACOVÁNÍ OBRAZU.....	10
3.5 REPREZENTACE MONOCHROMATICKÉHO OBRAZU.....	11
4 FILTRACE OBRAZU V PROSTOROVÉ DOMÉNĚ.....	11
4.1 JASOVÉ TRANSFORMACE.....	12
4.1.1 NEGATIV.....	13
4.1.2 GAMA KOREKCE.....	15
4.1.3 HISTOGRAM.....	15
5 FILTRACE ZAHRNUJÍCÍ OKOLÍ.....	18
5.1 VYHLAZOVÁNÍ OBRAZU.....	19
5.1.1 LINEÁRNÍ VYHLAZENÍ.....	19
5.1.2 MEDIÁNOVÁ FILTRACE.....	20
5.2 DETEKCE HRAN A IZOLOVANÝCH BODŮ A NESPOJITOSTÍ.....	21
5.2.1 PRVNÍ DERIVACE.....	22
5.2.2 SOBELŮV OPERÁTOR.....	23
5.2.3 DRUHÁ DERIVACE, LAPLACEŮV OPERÁTOR.....	24
6 KOMENTÁŘ K PRAKTICKÉ ČÁSTI.....	26
7 ZÁVĚR.....	29
8 LITEATURA.....	30

Úvod

Úprava digitalizovaného obrazu se v dnešní době často využívá například k odstranění nežádoucích artefaktů z obrazu, způsobených buď nesprávnou expozicí obrazu, rozmazáním pohybem, příliš dlouhou expozicí a podobně. Některé digitální filtry dokonce přejímají znalosti z počátků fotografie, tedy z dob, kdy výpočetní technika neexistovala.

Zpravidla pro správnou volbu filtrační metody je nutné specifikovat funkční parametr rušivého elementu (šumu) v obraze. Pak lze na základě filtračních metod rušivý element z obrazu odstranit.

Ne vždy se musí jednat o rušivý element v podobě šumu, filtrační metody naleznou uplatnění i širokém spektru průmyslového odvětví, kdy lze pomocí obrazové analýzy kontrolovat a vlastně i zautomatizovat kontrolu jakosti výrobků, zda odpovídají nastavenému standardu. Jmenovitě se může jednat o kontrolu v potravinářství, kde se kontroluje správná velikost zrn píce, rozlišení poškozených plodů, kontrola vajec a podobně. Rozpoznávací metody jsou hojně využívány i v mikrobiologii pro zautomatizované počítání kolonií bakterií a buněk. Dále se využívají k analyzování nalezených objektů co do tvaru, velikosti, barvy a podobně.

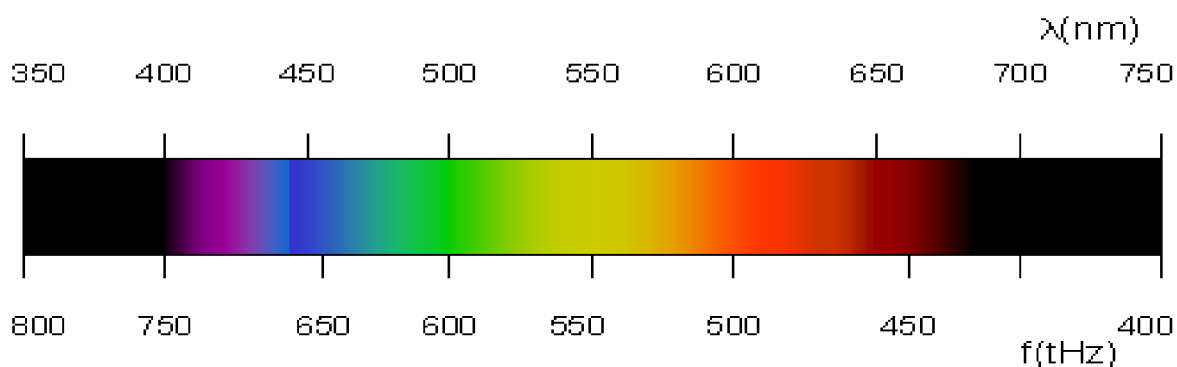
Filtrační metody jsou nedílnou součástí moderního lékařského vybavení, kdy pomohou správně diagnostikovat lokální místo v obraze, které je středem zájmu. Protože ne vždy je technicky možné dané místo zachytit jako obraz v dokonale kontrastním snímku.

V současné době existuje mnoho software pracujících s obrazovými daty. Ne vždy je však nabízený filtr vhodný a nezbude nám tedy nic jiného, než si danou filtrační metodu na základě znalostí algoritmů a samozřejmě programovacího jazyka napsat.

1. Světlo a barva

1.1 Vnímání barev v různých vlnových délkách

Viditelné světlo je elektromagnetické vlnění o vlnových délkách zhruba od 380nm do 700nm. Rozdílná frekvence světla uděluje vnímanému obrazu rozdílnou barevnost. Od nejnižší frekvence pro červenou složku po nejvyšší fialovou složku.



Obrázek 1.1 Barevné spektrum

1.2 Vnímání světla lidským okem

Na sítnici oka se nachází dva typy receptorů citlivých na světlo – tyčinky a čípky.

Tyčinky jsou receptory oka, jejichž citlivost je nastavena přibližně na oblast 505 nm vlnové délky světla. Lze to přisoudit evolučnímu přizpůsobení slunečnímu svitu.

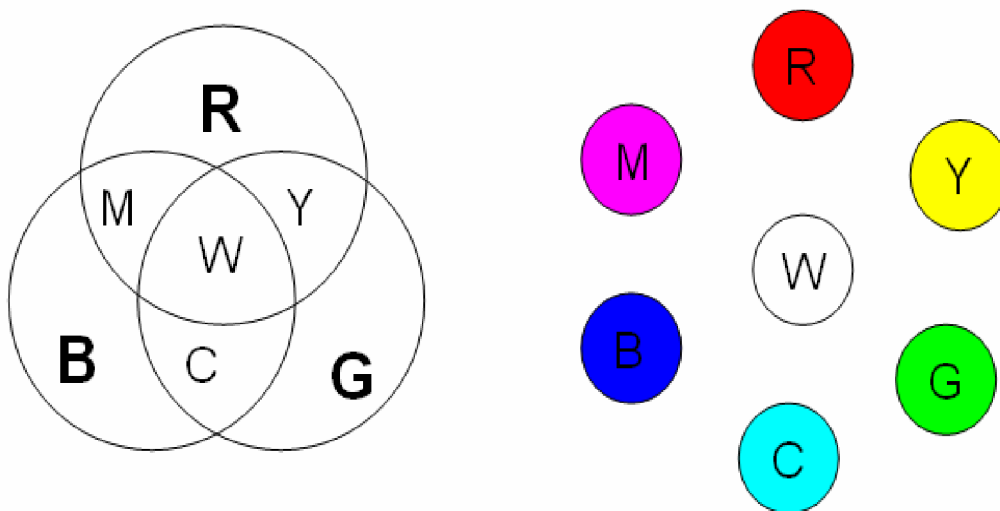
Tyčinky jsou propojeny do skupin a zvyšují tak citlivost na velmi malou dávku světla. Zajišťují tedy vnímání obrazu za šera - tzv. skotopické vidění, nebo také černobílé.

Na druhou stranu tyčinky slouží jako detekce pohybu při periferním vidění, kdy právě největší množství tyčinek je obsaženo na periférii sítnice a vdaný okamžik barva není to podstatné. Tyčinky nemají příliš velkou rozlišovací schopnost, takže například čtení novin za špatných světelných podmínek je nemožné, kdy sice rozeznáme tisk – tedy písmo na papíře, ale nejsme schopni jej přečíst právě z důvodu nízké rozlišovací schopnosti.

Youngova a Helmholtzova teorie popisuje vnímání barvy pomocí **čípků**, které jsou uzpůsobeny k vnímání specifické barvy.

- **Krátkovlnné** - citlivé na modrou barvu (400 – 510 nm)
- **Střední** - citlivé na zelenou barvu (450 – 610 nm)
- **Dlouhovlnné** - citlivost na červenou barvu (465 690 nm)

Barevný vjem je tedy tvořen kombinací jednotlivých signálů tří čípků, kdy oko zaznamenává jednotlivé barevné kombinace kdy jde o podíl červené, zelené a modré. Jednotlivé signály jsou v mozku zkombinovány a vzniká výsledná barva. Jde tedy o aditivní skládání. Viz obrázek 1.2.

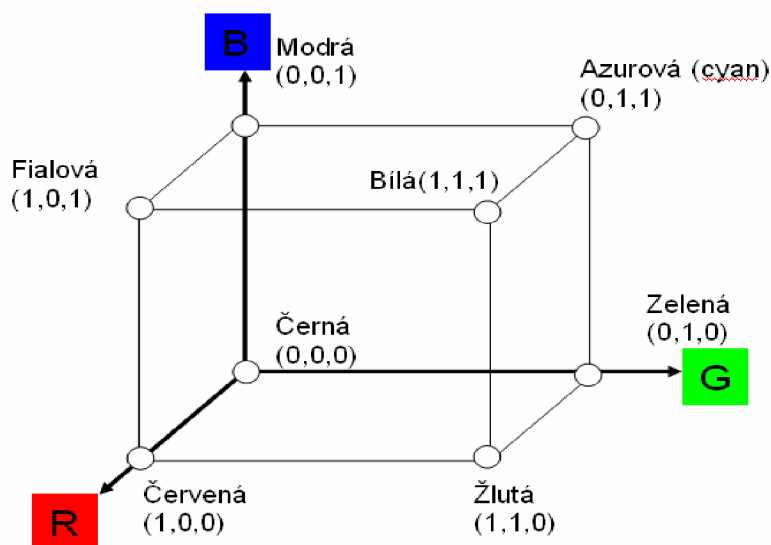


Obrázek 1.2 RGB barevný model aditivního skládání barev

2. Barevné prostory

2.1 Barevné prostory RGB

Barevný prostor je geometrické vyjádření daného barevného profilu a slouží k vizualizaci a představě o upořádání barev. V počátku souřadnic x, y, z leží černá barva. Naopak v rozích krychle (stále na ose x, y, z) leží základní barvy bez příměsí. V ostatních rozích se nachází doplňkové barvy CMY (Cyan, Magenta, Yellow). Na tělesové úhlopříčce se nachází neutrální barvy.



Obrázek 2.1 Barevný prostor RGB

2.2 Převod barevného obrazu RGB na odstíny šedé (Grayscale)

Říká se, že barevná fotografie je o barvách a ta černobílá je o tom ostatním, neboť černobílá fotografie je stále velmi oblíbená mezi fotografy. Kdy zvolit převod barevné fotografie na černobílou záleží na povaze fotografie. Pro převod fotografie je vhodnější objekt s jednoduchým motivem a dobrým kontrastem, neboť u černobílé fotografie může docházet k splývání detailů v pozadí (např. krajiny). Tedy je černobílá fotografie ideální prostředek pro pořízení rozličných aktů, kdy dává prostor představivosti.

Při zpracování barevného obrazu v technické praxi, někdy není potřeba zkoumat jednotlivé barevné složky separátně. Například v černobílém (lépe šedo-tónovém) obrazu se hledají hrany předmětů.

Jednotlivé barevné složky (R-Red, G-Green, B-Blue) mají v součtu rozdílné zastoupení z důvodu různé citlivosti lidského oka. Kdy neváhovým sečtením barevných složek RGB dojde k vytvoření výsledného jasu f .

$$f = 0,299R + 0,587G + 0,114B$$

2.3 Modely CMY a CMYK

Jedná se o barevný model primárně určený pro tisk a označuje se jako subtraktivní. Barvy CMY a CMYK se vyjadřují též čísly, jako je tomu u RGB, ale narozdíl od aditivního režimu přidáním barvy ubíráme světlo odražené od papíru. Takže vyšší hodnota čísla reprezentuje tmavší barvu. Zpravidla postačují pouze barvy CMY k vytvoření všech odstínů barvy. Spojením všech tří barev CMY by se mělo dosáhnout černé barvy. Ve skutečnosti je černá barva spíše tmavě šedivá. Z tohoto důvodu a ještě z důvodu úspory barviva se do tisku přidává ještě čtvrtá barva – černá. Stejně jako RGB model i model CMYK má svá omezení v rozsahu barev – tzv. gamut. Barevná omezení subtraktivního režimu jsou obvykle menší než u aditivního režimu.

Přepočítání mezi RGB a CMY

Nejjednodušší varianta přepočtu vychází z toho, že barvy CMY jsou protikladem světla RGB.

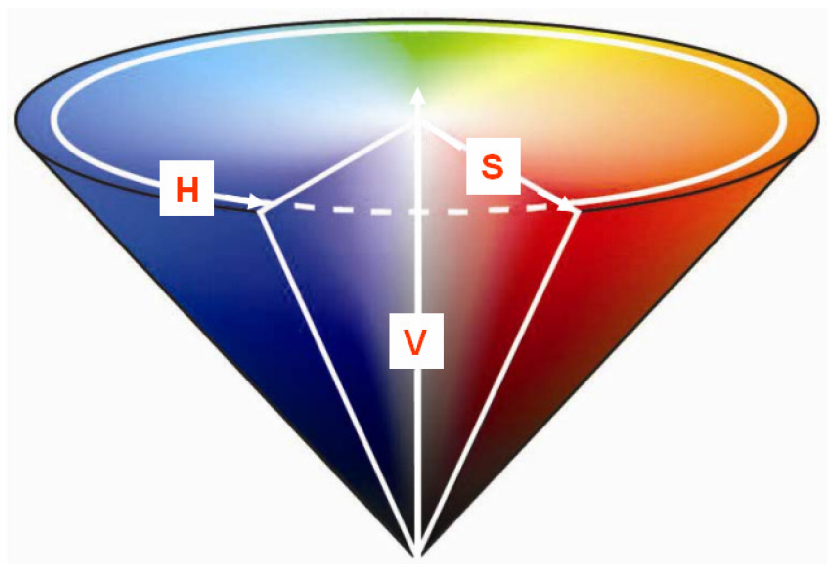
Budeme-li předpokládat, že jsou úrovně normovány, lze uvést vztah:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.1)$$

Intenzitu světla a barev lze uvést i v procentuálním vztahu a tedy například pro Cyan: $C = 100\% - R$.

2.4 Model HSV

HSV (Hue, Saturation, Value), též známý pod názvem HSB (Hue, Saturation, Brightnes) je barevný model, který se nejvíce přibližuje lidskému vnímání barvy. Zatímco modely RGB a CMYK jsou modely založené na míchání světla respektive barev, tak tento model definuje barvy přirozeným způsobem. Takže řeší problém, jak změnit odstín vybrané barvy při zachování sytosti.



Obrázek 2.4 Barevný model HSV

Hue (H) - definuje barvu v tónu v rozsahu 0° až 360° na barevném kotouči. Dohodou se červená barva skrývá pod úhlem 0° Přehled tří základních barev je uveden v tabulce 1

Barva	Úhel natočení
Červená	0° a 360°
Zelená	120°
Modrá	240°

Tabulka 1 Úhel natočení H

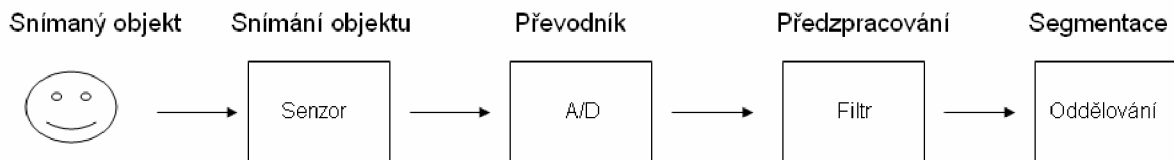
Saturation (S) - sytost barvy, příměs jiné barvy. Udává čistotu barvy, kdy představuje množství šedi v poměru k odstínu, udává se v procentech od 0% (šedá) do 100% tedy plně sytá barva.

Value (V) – udává hodnotu jasu, množství bílého světla. Relativní světlost nebo tmavost barvy..

Kombinací těchto tří informací vzniká reálná představa o barvě.

3 Snímání a zpracování obrazu

Pokud budeme zpracovávat nějakou spojitou veličinu a v našem případě obraz pomocí výpočetní techniky je nutno převést signál na diskrétní veličiny. Proces převodu spojitě veličiny na diskrétní se nazývá digitalizace signálu. Vlastní průběh se dá rozdělit do následujících kroků.



Obrázek 3 Zpracování obrazu

Senzor (snímače) a jeho základní parametry

- použitá technologie CCD, CMOS
- typ řádkový, plošný
- rozměru senzoru
- spektrální citlivost senzoru

Převodník A/D

- provádí vzorkování a kvantování
- může řešit i jednoduché předzpracování obrazu

Předzpracování

- obsahuje funkce na potlačení šumu
- potlačení, nebo zvýraznění hran apod.

Segmentace

- analýza obsahu obrazových dat

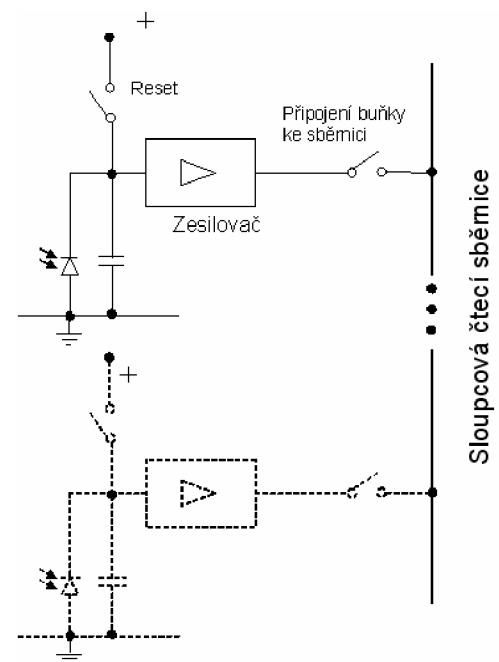
3.1 Senzory (snímače)

Jedná se o elektronické součástky, používané pro snímání obrazové informace.

3.1.1 CMOS senzory (Complimentary Metal-Oxide Semiconductor)

Nejjednodušší CMOS senzor je **pasivního** charakteru jak je zobrazeno na obrázku vpravo. Ve zjednodušeném popisu nalezneme resetovací spínač, který nabije kapacitu připojenou paralelně k fotodiodě. Následně danou kapacitu fotodiody vybíjí. Výsledný náboj kapacity je úměrný expozici buňky, kdy po skončení expozice je náboj zesílen a předám ve vhodný čas na čtecí sběrnici. V reálném zapojení se využívá kapacitních vlastností fotodiody.

Z praktického hlediska nedosahuje pasivní snímač vysoké kvality, proto je každá buňka CMOS senzoru doplněna o analytický obvod (APS) eliminující šum. CMOS senzor je v principu vyroben stejnou technologií jako jsou vyrobeny ostatní integrované obvody a právě z toho důvodu je i nižší výrobní cena záznamového senzoru ve srovnání CCD. Další výhodou CMOS může být téměř okamžitá připravenost snímače pro další snímek.



3.1.2 CCD senzory (Charge Coupled Device)

Základem činnosti CCD senzoru je tzv. fotoefekt, kdy foton při nárazu do atomu dokáže převést některý z jeho elektronů ze základního do excitovaného stavu. Podle způsobu využití lze konstrukce CCD snímače rozdělit na dvě základní skupiny:

- **Lineární CCD**

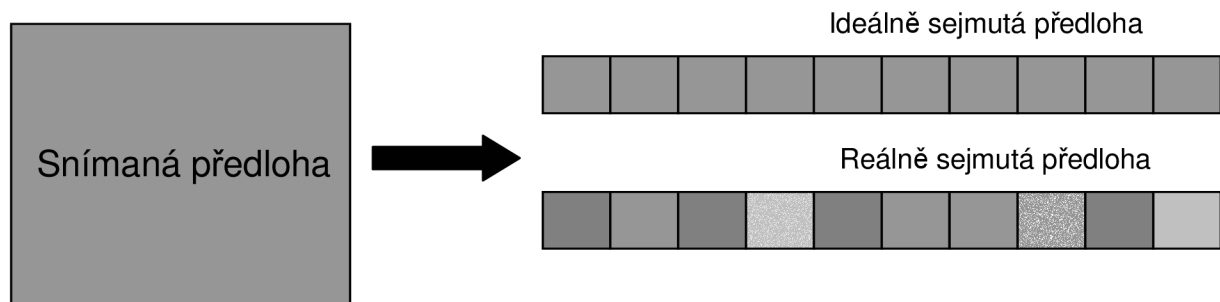
Jedná se o snímání pouze jednoho směru předlohy. Způsob vyžití najde u jednoduchých snímačů, kde se snímá pouze jeden směr předlohy. Jako typický příklad lze uvést čtečku čárového kódu, kdy na výstupu snímače je skupina pulsů, odpovídajících číslicím (tedy kódu). Své využití může nalézt například i ve scanneru, kdy se sejme jeden řádek předlohy a na další řádek se posouvá čidlo (nebo předloha), pomocí servomotoru.

- **Plošné CCD**

Základem plošného CCD je spojení mnoha lineárních snímačů do jednoho čipu. Své využití nalezne v digitálních kamerách apod.

3.2 Šum v obraze

Pokud bychom fotografovali rovnoměrně nasvícenou šedou tabulku ve velmi dobře rozmístěném osvětlení, lze v teoreticky předpokládat, že každý sensor snímače sejme stejnou hodnotu. Stejnou hodnotu bychom měli získat i při opakování pokusu se snímáním předlohy. Avšak v reálném případě všechny buňky snímače nemají přesně identické parametry. Navíc výsledné číslo z buňky je ovlivněno i dalšími parametry jako jsou například různé fluktuace, nepřesnosti apodobně.



Obrázek 3.3 Šum v obraze a jeho snímání

Druhy šumu a jeho základní rozdělení

A. Podle vzniku

- **Šum při nulování**
Před započítáním expozice jsou jednotlivé buňky CMOS senzoru nastaveny na svou výchozí hodnotu připojením, nebo lépe vedením tranzistoru, který nabíjí kapacitu fotodiody. Nulovací šum je způsoben ne naprosto stejným nabitím kondenzátoru na výchozí hodnotu.
- **Šum samotné buňky**
Část šumu je způsobena použitou technologií, např. náhodné poruchy v systému. Druhá podstatnější část šumu je způsobena tepelným šumem a kvantovými jevy.
- **Šum příliš dlouhé expozice (Hot Pixel)**
Při příliš dlouhé expozici (řádově minuty, hodiny) se na sejmutém obraze mohou objevit poměrně kontrastní lokální body. Šum je způsoben teplotně generovanými elektrony.
- **Kvantovací šum**
Vzniká jako odchylka mezi přesnou hodnotou vzorku a kvantovací hladinou.

- **Šum příliš dlouhé expozice**
Je způsoben teplotně generovanými elektrony, které v buňce produkují falešný signál (častý jev dlouhé expozice při focení noční oblohy se stativem)

B. Podle rozložení pravděpodobnosti

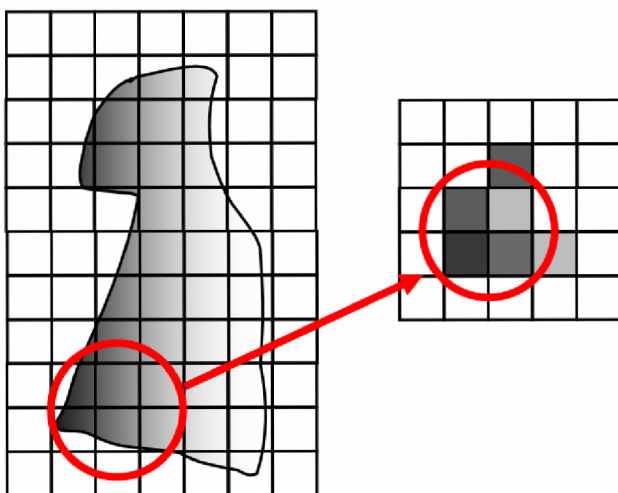
- **Gaussův šum**
Je popsán hustotou pravděpodobností výskytu $f(x)$; odstranění – např. Gaussův filtr
- **Bílý šum**
Pravděpodobnost výskytu šumu je stejná pro celé spektrum frekvencí

3.3 Zpracování obrazu

Vzorkování znamená periodické čtení hodnot (vzorků) z předlohy. Tento postup lze popsat jako jednorozměrnou funkci $f(x)$, kdy budeme hodnoty zaznamenávat diskrétních intervalech o vzdálenosti Δx . Tedy budeme mít sumu vzorků počínaje $x_0, x_1 \dots$ až x_n . Převrátíme-li vzorkovací hodnotu vzdálenosti, je výsledkem vzorkovací frekvence $f_{\text{sampilg}} = 1/\Delta x$. Vzorkovací frekvence úzce souvisí s možností zachytit detail ve sledovaném spojitým analogovém signálu. Pro správnou rekonstrukci obrazu je nutno mít alespoň dvojnásobnou vzorkovací frekvenci, než je frekvence vzorkovaného signálu.

Kvantování je proces, při kterém při kterém je navzorkované hodnotě přiřazena diskrétní hodnota přibližně odpovídající hodnotě jasu. Nejběžnější kvantování je uniformní a hovoříme o něm tehdy, pokud je rozdělení mezi jednotlivými jasovými úrovněmi stejná. S kvantováním je spojena i kvantizační chyba, kdy dochází ke ztrátě informace. Pro eliminaci chyby je možno využít kvantování, kdy bude interval mezi jednotlivými úrovněmi rozdílný a hovoříme tak o neuniformním kvantování. Pro reprezentaci N jasových úrovní je potřeba K bitů. Kdy jedním bitem lze reprezentovat pouze dva stupně jasu a to například bílou a černou. V digitální fotografii se setkáváme běžně s osmi bity na barvu, což se značí jako barevná hloubka. Dále se můžeme setkat i s vysokou barevnou hloubkou např. $2^{48} = 280$ triliónů barev.

Astronomický počet se může zdát zbytečný, neboť lidské oko není schopno tolik barev rozlišit, ale použití tolika barev umožňuje bezproblémové úpravy obrazu, kdy vysoké hodnoty eliminují význam chyb.



Obrázek 3.4.1 Digitalizace snímané předlohy

Obrázek 3.4.1 znázorňuje kontinuální obraz promítaný na snímač obrazu. Je patrné, že kvalita sejmutého obrazu je ovlivněna počtem vzorků a úrovní šedi.

3.4 Re prezentace monochromatického obrazu

Monochromatický obraz je reprezentován dvourozměrnou maticí $f(x, y)$ [1]

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, N-1) \\ f(1,0) & f(1,1) & \dots & f(1, N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1, N-1) \end{bmatrix} \quad (3.1)$$

kde $x \in \langle 0, \dots, M-1 \rangle, y \in \langle 0, \dots, N-1 \rangle$

Matice má M řádků a N sloupců, kde každý prvek této matice se nazývá obrazový element, obrazový prvek, pixel případně pel.

Z programátorského pohledu se jedná o dvourozměrné pole hodnot. Pro reprezentaci barevného obrazu se používá trojrozměrné pole $f(x, y, barva)$

4 Filtrace obrazu a úprava obrazu v prostorové doméně

Pro svou práci jsem zvolil zpracování v prostorové doméně, která dosahuje kvalitativně rovnocenných výsledků, jako zpracování obrazu ve frekvenční doméně.

Mějme původní obraz s hodnotami jasu $f(x, y)$ s definičním obrem hodnot $\langle x_{\min}, x_{\max} \rangle \times \langle y_{\min}, y_{\max} \rangle$, který je transponován do nových jasových úrovní $g(x, y)$ podle vztahu $g(x, y) = T[f(x, y)]$. Jas je určen v reálném oboru hodnot v rozsahu $\langle 0; 1 \rangle$, kdy nulovou hodnotu zpravidla zastupuje bílá složka a černá složka je udána hodnotou 1.

Pro uložení a zpracovávání jasové informace se využívá vztahu $\langle 0; L-1 \rangle$, kde L udává počet jasových úrovní.

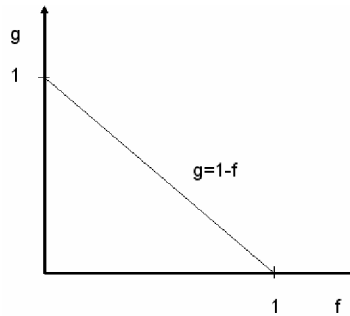
4.1 Jasové transformace (Image Enhancement)

Jedná se o poměrně jednoduchou úpravu vzhledu, kdy změna vzhledu původního obrazu se transponuje do nových jasových úrovní. Důvodem použití jasové transformace může být například nesprávná expozice snímku, kdy v daném snímku převládá tmavá, nebo světlá jasová složka. Extrémním osvětlením fotografovaného objektu se ze světlých míst ztrácí rozlišitelné detaily (tedy ztrácí se kresba) a vzniká tzv. „přepal“. V tomto případě se už jedná o ztrátu informace v obraze, kdy řešením může být použit informaci z jiného snímku, případně zkusit poškozenou oblast zrekonstruovat.

4.1.1 Negativ (Image Negatives)

Slovo **negativ** pochází z latinského základu *negativus* neboli převrácený. Toto slovo se začalo objevovat v době vynálezu fotografie. Tehda se jednalo ještě o negativ, pořízený na skleněnou desku. Klasický černobílý negativ zná snad asi každý, kdy už po vyvolávací proceduře foto-filmu bylo možno určit nakolik kvalitně byl daný snímek filmu exponován a zda má smysl daný obraz osvětlit na foto-papír. U barevného filmu zůstává princip shodný, jen jednotlivé negativní barvy jsou vzájemnými doplňkovými barvami ostatním.

Negativ patří mezi nejjednodušší jasové transformace, kdy transformace nevyužívá hodnot z předchozí analýzy obrazu. Intenzita každého bodu v obraze je nahrazena jinou jasovou úrovní. Tento typ zpracování je zejména vhodný pro snazší čitelnost obrazu, pokud jsou v obraze dominantní tmavé složky barev (resp. bílé), kdy nositelem informace je minoritní složka.



Obrázek 4.1 Křivka hodnot jasu transformovaných na negativ

Negativ monochromatického obrazu lze vyjádřit [1] podle vztahu:

$$g(x, y) = 1 - f(x, y) \quad (4.1)$$

v případě diskrétních hodnot:

$$g(x, y) = L - 1 - f(x, y) = 255 - f(x, y) \quad (4.2)$$

Negativ barevného obrazu:

$$g(x, y, z) = 255 - f(x, y, z) \quad (4.3)$$

4.1.2 Gama korekce (Power-Law Transformations)

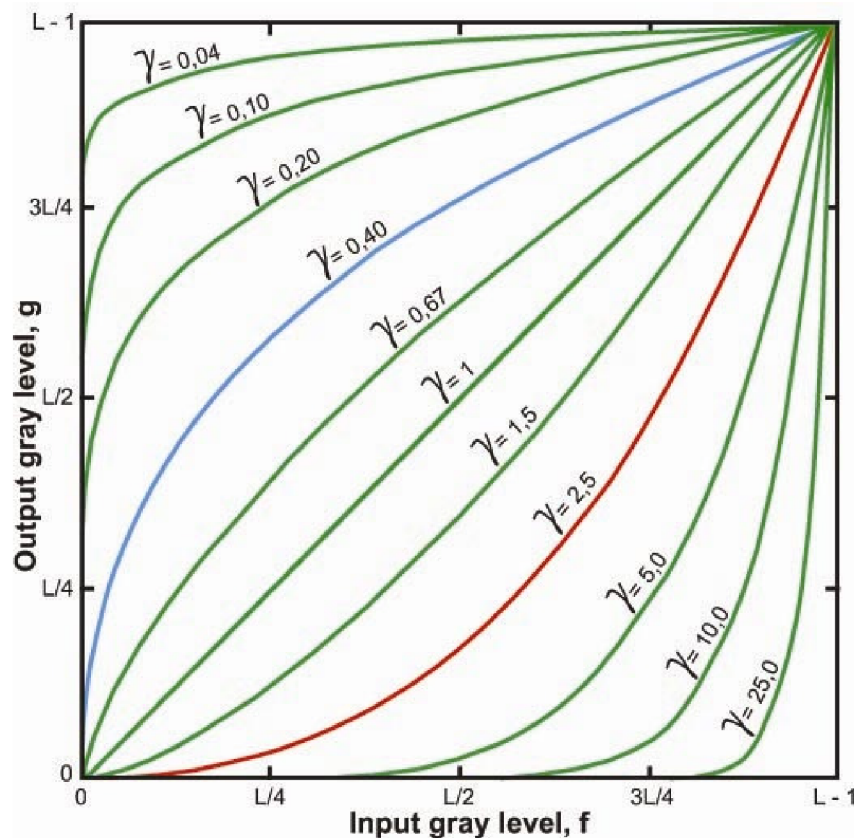
V anglické literatuře též známa jako Power-Law Transformation. Jedná se o nelineární kompenzaci jasových složek, případně barevných odstínů. Původním důvodem vzniku gama korekce, byla nelineární závislost CRT obrazovek. Gama korekce se dále využívá k opravě jasu při špatné expozici, kdy požadovaný objekt v obraze je ukryt v příliš světlé, nebo naopak tmavé části obrazu.

Základem gama korekce je výpočtový vztah, který mění původní jasový signál f na průběh jasu intenzity g .

$$g = c(f + \varepsilon)^\lambda \quad (4.4)$$

Kde λ a c jsou kladné konstanty a ε značí vliv offsetu (tedy úroveň výstupu, kdy vstup je nastaven na nulovou hodnotu). Nicméně offset je zpravidla zahrnut do kalibrace „obrazového zařízení“ a v tom případě je často zanedbáván. A tedy:

$$g = cf^\lambda \quad (4.5)$$



Obrázek 4.3 Křivky Gama korekce

Z obrázku 4.3 je zřejmé, že křivky generované s hodnotami $\gamma \leq 1$ mají přesně opačný efekt, než křivky nabývající hodnot $\gamma \geq 1$. Červeně je vyznačena křivka CRT obrazovky se žhavenou katodou, kdy obrazovka zpravidla produkuje snímek tmavší

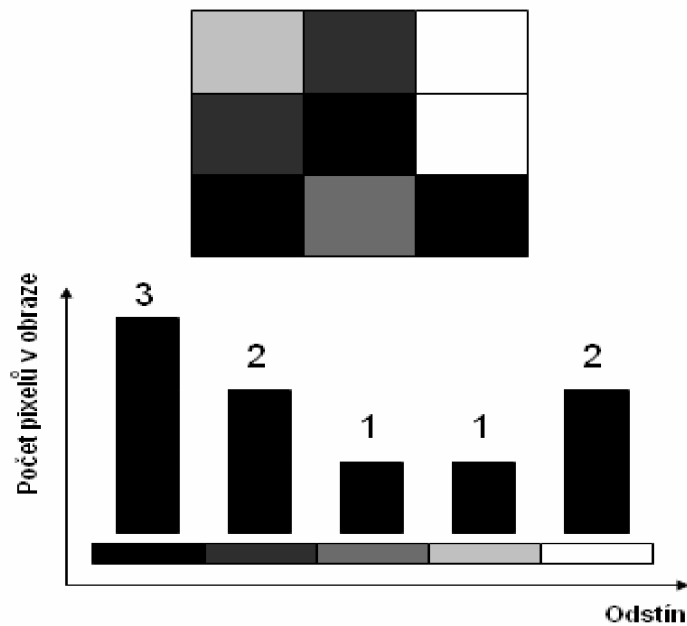
než je tomu na vstupu monitoru. Je to dáno tím, že obrazovky nezobrazují přiváděné vstupní napětí lineárně, kdy křivka $g = c \cdot f^\gamma$ přibližně odpovídá exponentu $\gamma \approx 1,8$ až 2,5. V tomto případě se předzpracuje vstupní obraz monitoru provedením transformace $g = f^{1/2,5}$, která je vyznačena modrou křivkou. Výsledkem je tedy vyzářený snímek na monitor, který má přibližně stejnou hodnotu, jako by byl zobrazován na jakémkoli jiném zobrazovacím zařízení. Podobná operace se provádí i na dalších zařízeních, jako jsou tiskárny a scannery i snímací zařízení. Pro reprodukci barev je potřeba též přesně specifikovat gama korekci, protože různé změny gama korekce způsobují i změny barevných složek v obraze. Gama korekce je též užitečná pro základní manipulaci s kontrastem. Kdy je požadovaná část snímku ukryta v ztmavené části obrazu. Typickým příkladem může být snímek magnetické rezonance (MR). Pořízený snímek obsahuje převážně tmavou složku a je zapotřebí snímek rozlišit na větší škálu šedi. Například snížením gama korekce na hodnotu $\gamma = 0,4$ [2] lze dosáhnout optimálních výsledků, kdy je pozorovatelné více detailů v obraze.

4.1.3 Histogram

Histogram je jednou z charakteristik konkrétního obrazu. Z histogramu lze vyčíst základní expoziční vlastnosti obrazu. Dále lze získat i informace o barevném vyvážení obrazu. Jednoduchý histogram obsahují i moderní digitální fotografické přístroje, které nám mohou pomoci k nalezení ideální expozice obrazu fotografie. Dále histogram slouží pro rychlou informaci o pořízené fotografii, kdy fotografie sama o sobě je dobrá, ale stále jí něco chybí. Samozřejmě histogram nalezne využití i jiných oborech než pouze zkoumání a editace fotografie.

Princip histogramu

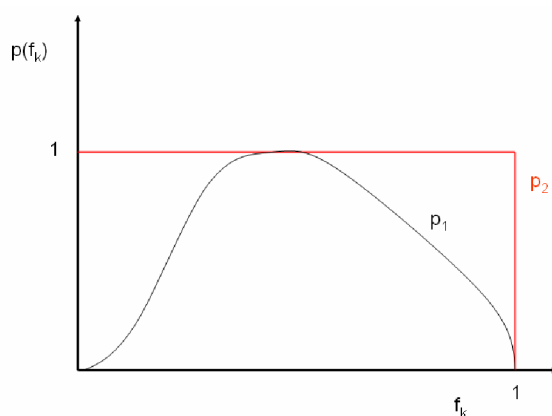
Na obrázku 4.4 je zobrazen princip histogramu, kdy na horizontální ose zleva doprava je naznačen jas od černé po bílou. Vertikální osa nám udává kolik buněk obrazu bylo exponováno příslušným jasnem.



Obrázek 4.4 Princip histogramu

Vyrovnnání histogramu

Označme horizontální osu f_k , kde „k“ určuje počet (jasových) úrovní v obraze a vertikální osa $p(f_k)$ udává pravděpodobnost rozložení hustoty výskytu jasu.



Obr 4.5. Vyrovnnání histogramu.

Chceme-li v diskrétním případě vyrovnat histogram a mít tedy rovnoměrnou hodnotu jasů v obraze, tak jak je naznačeno v obrázku 4.5 použijeme sumační vztah [1]

$$g_k = \sum_{j=0}^k p(f_j) \quad (4.6)$$

kde $p(f_j)$ udává normalizovaný vztah histogramu a suma všech složek je rovna jedné.

$$p(f_j) = \frac{n_k}{N} \quad (4.7)$$

a po dosazení

$$g_k = \frac{1}{N} \sum_{j=0}^k n_j \quad (4.8)$$

V případě, že nestačí pouhé vyrovnání histogramu, kdy nás zajímá jen část jasových složek, a je potřeba zbytek jasových složek potlačit použijeme metodu přímé specifikace histogramu.

Kritéria pro posouzení histogramu

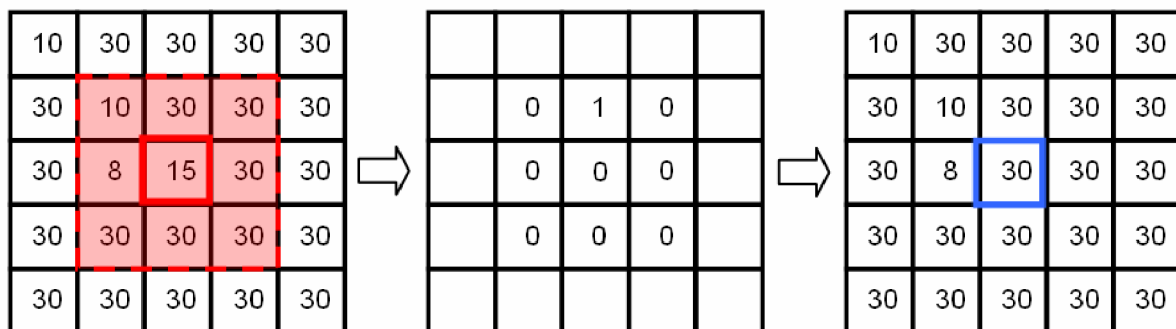
- Histogram který je posunutý na jednu, či druhou stranu - svědčí o špatné expozici obrazu..
- V histogramu jsou zastoupeny převážně střední hodnoty – tedy dojem zamlženého obrazu. V tomto případě chybí krajní polohy grafu histogramu, tedy je zúžený tonální rozsah.
- V histogramu se vyskytují převážně krajní hodnoty – obraz je zpravidla velmi kontrastní, ale v obraze chybí přechod.
- Posterizace – úbytek odstínů v celé ploše histogramu. Obvykle se jedná o vadu, která je důsledkem úprav obrazu.

5 Filtrace zahrnující okolí

Při filtraci obrazu velice často narazíme na **diskrétní dvourozměrnou konvoluci**, někdy též nazývanou diskrétní 2D konvolucí. Jedná se o operace s maticí pomocí jiné matice, nazývané jako konvoluční jádro nebo maska. Obvyklý rozměr matice je 3 x 3, případně 5 x 5. Maska se postupně pohybuje po zpracovávaném obraze a postupně se provádí součin každého koeficientu masky s hodnotou v obraze a dále se provede součet všech těchto součinů. Výsledná hodnota se uloží jako nový obrazový bod. Jinými slovy konvoluční matice definuje to, nakolik je dotýčný segment dat ovlivněn sousedními segmenty pomocí konvoluce. Tento postup je vyjádřen následujícím vztahem:

$$g(x, y) = \sum_{s=-1}^1 \sum_{t=-1}^1 m(s, t) f(x-s, y-t) \quad (5.1)$$

$g(x, y)$ g – výsledný segment obrazu
 m - konvoluční matice
 x, y - pozice v prostoru



Obrázek 4.6 Princip konvoluce

Vlevo je matice, která obsahuje hodnoty jednotlivých pixelů. Červeně označená oblast matice zobrazuje střed matice a okolní pixely, které jsou využity pro konvoluci. Prostřední matice je maska a vpravo je výsledná matice konvoluce. Pro řešení krajních hodnot na obraze se zpravidla používají tyto metody:

- Obraz je zvětšený o poloviny velikosti masky a okraje se doplní nulami.
- Maska se pohybuje v rozsahu obrazu, kdy výsledkem je zmenšený obraz
- Zřídka se používá zmenšená maska

Lokální filtrace obrazu

Jedná se o metody používaných k filtracím (předzpracováním) obrazu, kdy se k výpočtu nové hodnoty pixelu uvažuje i bezprostřední okolí daného bodu. Systém je založen na představě, že se celý obraz po jednotlivých bodech obrazu prochází a zkoumá se i okolí jednotlivých bodů. Ke zkoumání okolí se využívá konvoluce, tedy lineární funkce, kterou lze s úspěch využít podle účelu na:

- **Vyhlazení obrazu** operace při níž se potlačují velké změny v obraze tak, že výslední jasový bod se průměruje pomocí okolních bodů bezprostředně obklopující uvažovaný bod. Metoda najde uplatnění všude tam, kde potřebujeme potlačit šum a odstranit různě osamocené fluktuace v obraze.
- **Detekce hran** - funkce se snaží z okolí reprezentativního bodu odhadnout derivaci obrazové funkce.

Výše jmenované funkce jsou svými vlastnostmi vlastně v protikladu, kdy například potřebujeme vyhladit obraz a zachovat ostrost hran. Z tohoto důvodu se využívají i nelineární metody.

5.1 Vyhlažování obrazu (Image Smoothing)

Vyhlazení obrazu se zpravidla používá k odstranění náhodných fluktuací z obrazu a šumu. Obraz může být vystaven šumu z několika zdrojů, může to být dáno snímacím čidlem, které je nastaveno na velmi vysokou citlivost, dále to může být zrno použitého filmu, nebo vliv rušení v přenosové cestě. Poškozené body obrazu se často výrazně liší od svých sousedních tím, že obsahují obecně vyšší složky kmitočtového spektra charakteristiky obrazu a tomto poznatku je postavena metoda vyhlazování obrazu.

5.1.1 Lineární vyhlazení

Vyhlazení v prostorové doméně pomocí konvoluce

Základem metody vyhlazování je **průměrování**, kde každému bodu přiřadíme novou zprůměřovanou jasovou hodnotu pomocí konvoluce. [1]

$$g(x, y) = \frac{1}{M} \sum_{s=-1}^1 \sum_{t=-1}^1 f(x-s, y-t) \quad (5.2)$$

Kde pro M - počet bodů v masce:

$$M = (2s_{MAX} + 1) \cdot (2t_{MAX} + 1) \quad (5.3)$$

Běžné masky 3 x 3 low-pass:

Maska 1:

$$h = \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad (5.4)$$

Následující maska (4.13) je více zajímavější, kdy v masce jsou rozdílné hodnoty. Lze si povšimnout, že středová hodnota matice obsahuje vyšší hodnotu, než ostatní. Takže v konečném důsledku je na středový pixel kladen větší důraz, než na ostatní pixely v uvažovaném průměru.

Maska 2:

$$h = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \quad (5.5)$$

Hodnoty (body) na diagonále matice jsou dále od středu, než na ortogonále a váží méně (menší důležitost), než v okolí středu matice. Tato matice se značí někdy i jako váhová. Tato myšlenka napomáhá redukovat rozmazání objektů (tedy hlavně hran), při vyhlazování. Je samozřejmě možné zvolit i jiné váhové koeficienty, avšak podle vzoru, kdy impulsní odezva je definována:

$$h = \frac{1}{b+2} \begin{pmatrix} 1 & b & 1 \\ b & b^2 & b \\ 1 & b & 1 \end{pmatrix} \quad (5.6)$$

Samozřejmě lze použít matici s větší velikosti pole např. 5 x 5, 7 x 7, 15 x 15. Kdy se sice dosahuje lepšího vyhlazení obrazu, ze cenu větší ztráty detailů v obraze. I tuto metodu lze s úspěchem použít pro eliminaci nepodstatných a vlastně matoucích objektů v obraze. Typickým příkladem může být Hubblův teleskop, nebo

mikroskopické sledování objektů v laboratoři a vyhodnocování pomocí podobného algoritmu apod.. Proto volba závisí na povaze sledovaného objektu.

5.1.2 Mediánová filtrace

Nelineární metody dokáží poskytnout větší kompromis mezi vyhlazením a ztrátou jemných detailů v obraze.

Mediánová filtrace vykazuje velmi dobré výsledky například při potlačení specifického šumu černých a bílých bodů v obraze, tedy vzhledu „pepře a soli“. V jednodimenzionální podobě se mediánový filtr skládá z posuvného okna, které tvoří lichý počet pixelů. Medián ξ je vybraný prvek ze souboru uspořádaných hodnot, kdy při lichém počtu prvků je to prvek $\xi = p_{n+1}/2$ - tedy prostřední hodnota uspořádaných prvků od minima po maximum. Pokud se vyskytne sudý počet prvků, vyberou se prostřední dva prvky s tím, že se dále utvoří jejich aritmetický průměr. Například v prostoru 3 x 3 je medián pátá největší hodnota prvků a v prostoru 5 x 5 třináctá nejvyšší hodnota. Princip filtrace spočívá v posuvu okna masky po obraze a výběru mediánu z hodnot ležících pod touto maskou.

5.2 Detekce hran, izolovaných bodů a nespojitostí

Hrany odpovídají prudkým změnám přechodu jasu. Takže cílem detekce hran je nalézt lokalitu, kde dochází k prudké změně jasu.

5.2.1 První derivace

Metody využívající první derivaci jsou založeny na poznacích o této funkci. Kdy derivace musí být nulová v oblastech s konstantní úrovní jasu a nabývá nenulových hodnot v oblastech přechodu jasu. Výsledkům první derivace obrazu říkáme gradient. Provedením parciální derivace ve směrech x a y získáme vektor, udávající velikost Δg a směr gradientu φ . [2]

$$\Delta g = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (5.7)$$

$$\varphi = \arctan \frac{\left(\frac{\partial f}{\partial y}\right)^2}{\left(\frac{\partial f}{\partial x}\right)^2} \quad (5.8)$$

Dále je potřeba určit podle osy x a y parciální derivace. Zpravidla se pro diskrétní obrazovou funkci používá vztahu aproximace [4]:

$$\Delta f(x) = \frac{(f(x+1) - f(x-1))}{2h} + O(h^2) \quad (5.9)$$

Kde funkce O udává řád chyby.

Běžnějším a používanějším způsobem pro výpočet jednotlivých složek gradientu je využití hranového operátoru, nebo též nazývaného konvoluční maska či jádro.

5.2.2 Sobelův operátor, použití první derivace

Jedná se o lokální filtr, který pro svou činnost využívá své okolí. Sobelův operátor se často využívá k detekci vodorovných a svislých hran v obrazu. Je reprezentován pomocí dvou základních matic (jader).

$$h_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & 1 \end{bmatrix} \quad h_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (5.10)$$

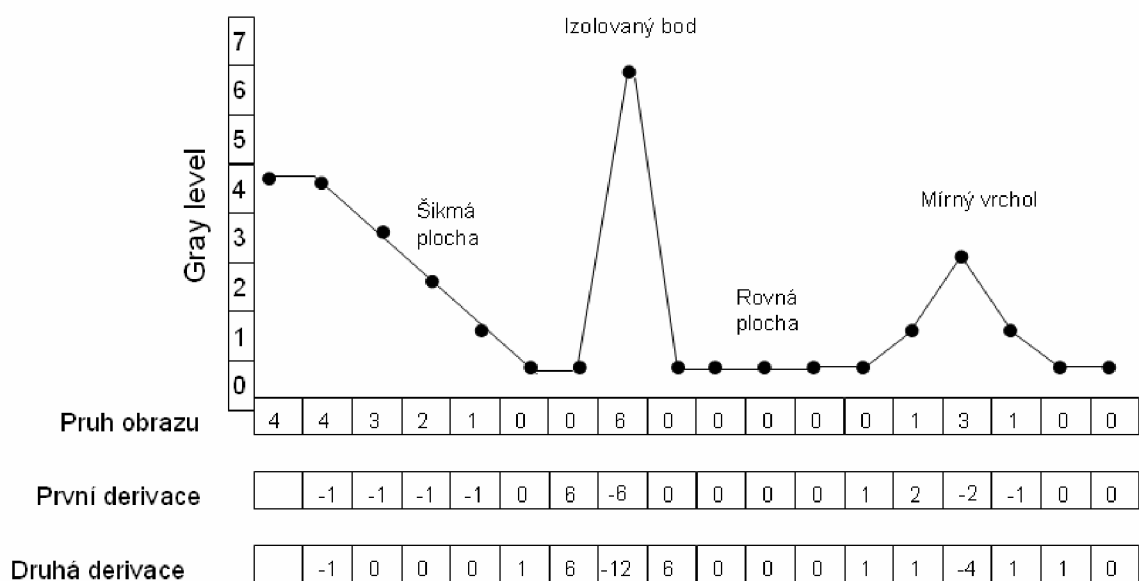
Pokud chceme detekovat hrany pouze v horizontální rovině použijeme matice

$$h_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & 1 \end{bmatrix} \quad h_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (5.11)$$

Dále tento filtr umožňuje detekovat i šikmé hrany obrazu, natočením masky o 45° .

5.2.3 Druhá derivace

Pokud nepotřebujeme zjišťovat, velikost a směr hran vystačíme si s druhou derivací.



Obrázek 4.7 Model transformace pomocí první a druhé derivace

Na obrázku 4.7 je uveden zjednodušený model transformace obrazu pomocí první a druhé derivace, kde počet úrovní šedi je zjednodušen na osm úrovní. Z modelu transformace, lze vyčíst, že první derivace na „šikmé ploše“ obsahuje hodnoty různé od nuly, zatímco druhá derivace má nenulové hodnoty pouze na začátku a na konci „šikmé plochy“. Nyní se zaměříme na „izolovaný bod“, který může představovat šum v obraze. Reakce na izolovaný bod je mnohem silnější u druhé derivace. Ano druhá derivace nechá vyniknout jemným detailům v obraze (jako jsou tenké čáry) mnohem více, než je tomu u první derivace. Ale na druhou stranu nechá vyniknout i šumu, než je tomu u první derivace. V mnoha aplikacích je výhodnější použít druhé derivace pro zlepšení obrazu, pro schopnost nechat vyniknout jemným detailům.

5.2.4 Použití druhé derivace, Laplaceův operátor

Druhá derivace představuje rychlost změny hodnot jasu, kdy se projevuje na strmých, nebo izolovaných hranách. Druhou derivaci lze pro směr x v diskrétním obraze počítat jako rozdíl rozdílů jasových hodnot nacházejících se vedle sebe: [2]

$$\frac{\partial^2 f}{\partial x^2} = [f(x+1, y) - f(x, y)] - [f(x, y) - f(x-1, y)] \quad (5.13)$$

a po úpravě:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y) \quad (5.14)$$

obdobně pro směr y jako:

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y) \quad (5.15)$$

Pro detekci izolovaných bodů lze využít Laplaceův operátor, který je definován pro funkci $f(x, y)$ dvou proměnných vztahem: [2]

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (5.16)$$

Laplaceův operátor $\nabla^2 f$ s implementací vztahů 4.18 a 4.19

$$\nabla^2 f(x, y) = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y) \quad (5.17)$$

V praxi je tato rovnice implementována použitím masky pro detekci izolovaných bodů.

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

Obrázek 4.8 Maska pro Laplaceův operátor v pozitivní variantě

I diagonální směr masky může být začleněn do filtrační masky, pomocí obdobného vztahu 4.18, nebo 4.19, kdy se jedná o ten samý princip. Vzhledem k tomu, že každý diagonální směr také obsahuje hodnotu $-2f(x, y)$, tak celkový odečet hodnot bude $-8f(x, y)$.

Ke zvýraznění hran a tedy silnějšího zostření obrazu, lze hodnoty derivací přímo přičíst k originálnímu obrazu pomocí superpozice.

6 Komentář k praktické části návrhu filtrů - v programovacím jazyku JAVA (NetBeans IDE 6.8)

Program ImageFilters implementuje návrhu třídy filtrů, schopné pracovat na vícejádrových procesorech. Zejména lokální filtry jsou obtížné na realizaci, neboť každý procesor potřebuje nejen svou část obrázku, ale i část jejího okolí. Program je koncipován tak, že každý procesor dostane jeden výřez obrazu typu Raster a produkuje výstup do jiného výřezu - WritableRaster, který už má velikost výstupu - tedy bez lokálního okolí. Koncepte je vhodná hned ze dvou důvodů, jednak se na architektuře PC NUMA uplatní zrychlení díky faktu, že jednotlivé procesory mají svá data každý ve své vlastní paměti a zadruhé tato modulární architektura může být dobrý výchozí bod pro realizaci obrazových filtrů v FPGA.

Inverze

Jednoduchý filtr, který vezme každý bod v obraze a udělá jedničkový doplněk.

Převod barevného obrazu RGB na odstíny šedé (Grayscale)

Filtr převodu do stupňů šedi je jednoduchý bodový operátor, který si bere obrazová data ve formátu RGB a 8-bit, přičemž provádí součin vektoru barev s následujícím vektorem: (0.299, 0.587, 0.114). Výsledkem je potom hodnota šedi, v rozsahu [0, 255).

Sépie

Filtr sépiového tónování funguje podobně, ale součiny se provádí tři, pro každou výslednou barvu jeden:

$$\begin{aligned} r_{\text{Out}} &= \min(255, r * .393 + g * .769 + b * .189) \\ g_{\text{Out}} &= \min(255, r * .349 + g * .686 + b * .168) \\ b_{\text{Out}} &= \min(255, r * .272 + g * .534 + b * .131) \end{aligned} \quad (6.1)$$

Je nutno ošetřit přetečení nad hodnotu 255, jinak se projevují nepříjemné vizuální artefakty.

Sobelův operátor

Filtr sobelova operátoru je jednoduchý lokální filtr, který potřebuje k výpočtu bodu jeho bezprostřední okolí. Výsledek získáme kombinací dvou konvolucí s následujícími jádry:

$$\text{MaticeA} = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \qquad \text{MaticeB} = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

Jádra aplikujeme na vstupní obraz následujícím způsobem: $\text{out} = \min(255, |\text{in} * \text{A}| + |\text{in} * \text{B}|)$. Původní sobelův vzorec totiž obsahuje odmocninu, která je pomalá na výpočet a navíc na některých zařízeních ani nemusí být možné ji spočítat. Pokud pracujeme s barevnými obrazy, spočteme sobelův filtr pro každou složku zvlášť a výsledkem jsou potom pestře barevné hrany.

Mediánová filtrace

Mediánový filtr je, velmi podobný sobelovu filtru, jedná se lokální obrazový filtr. Oproti sobelově filtru ale má proměnnou velikost okolí, což do důsledku otestuje adresovací logiku programu. Velikost okolí se nastavuje pomocí ovládacího políčka v panelu vlastností filtru. Při výpočtu mediánového filtru nejprve přečteme všechny pixely v okolí právě vyhodnocovaného pixelu, poté je seřadíme funkcí `java.Arrays.sort()` a vybereme si prvek, který leží uprostřed pole. Aby bylo možné vybrat prvek uprostřed, musí mít pole lichou délku. Z toho důvodu je uživatelské rozhraní nastavené tak, aby dovovalo zvyšovat velikost jádra jen s krokem dva.

Histogram

Poměrně složitá filtrační metoda na realizaci. Tato filtrační metoda je globální, protože každý výstupní pixel závisí na hodnotách celého obrazu. Filtr postupuje ve dvou krocích. Prvním je analýza vstupního obrazu tak, že se počítá kolik je v obrazu pixelů s danou stejnou hodnotou. I tato část je paralelizována, každý procesor tvoří histogram své části obrazu. Na konci výpočtu se všechny histogramy sečtou. Histogram je poté použit k odhadu parametrů ekvalizace, jimiž je minimální vyskytující se hodnota a maximální vyskytující se hodnota. Histogram je poté graficky zobrazen uživateli, který může odhadnuté parametry podle své volby ještě upravovat. Při dalším výpočtu se už histogram chová jako bodový filtr, kdy každý bod závisí pouze na korespondujícím bodu ve vstupním obraze, a na nastavení koeficientů. Funkci filtru můžeme popsat následující rovnicí:

$$f(x, low, high): y = \max(0, \min(x - low) / (high - low))$$

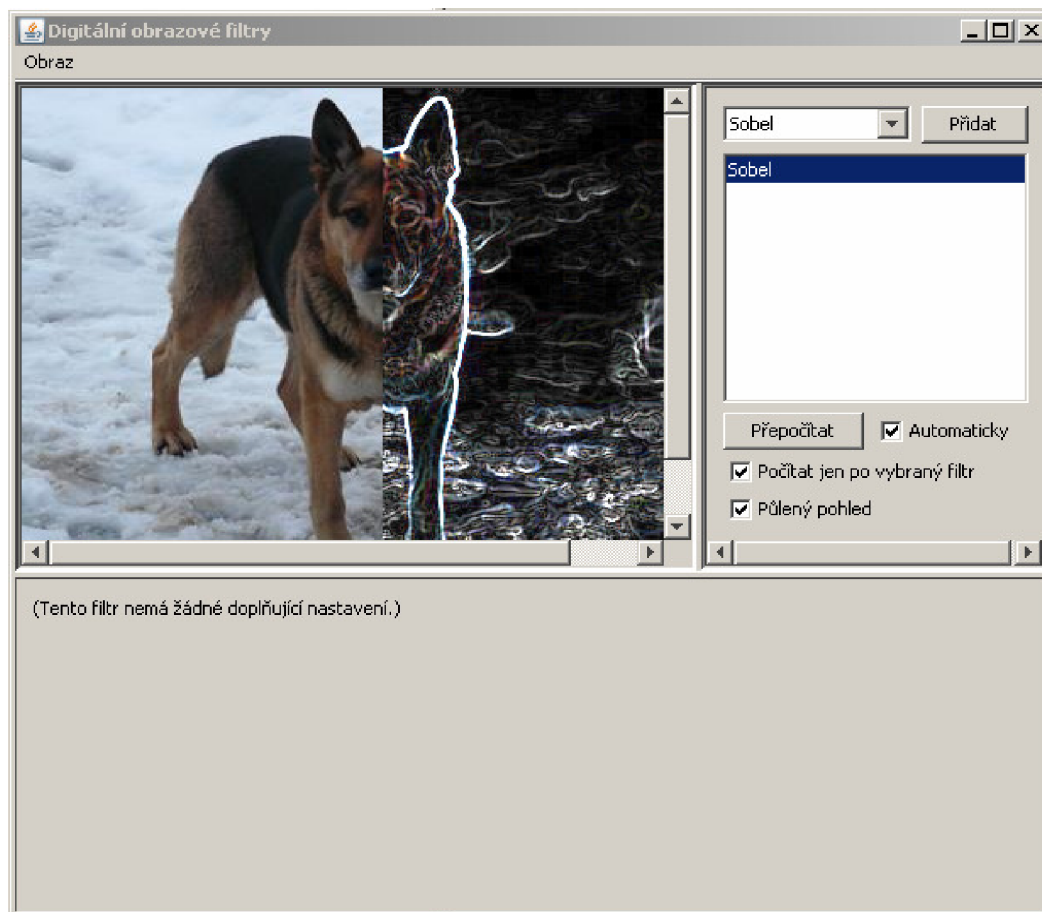
Za povšimnutí stojí, že výsledek se musí upravovat tak, aby nedošlo k přetečení, či dokonce k podtečení - obojí je zde možné, v závislosti na pozici posuvníků.

Posledním úskalím je, že posuvníky nesmí mít stejnou hodnotu, protože potom by docházelo k dělení nulou. Pokud posuvníky mají stejnou hodnotu, dělíme v implementaci nejbližším vyšším číslem - jedničkou.

Pokud ekvalizujeme histogram vícesložkového obrazu, koeficienty odhadujeme společné pro všechny složky (tedy při tvorbě histogramu se sčítají jak červené, tak zelené i modré hodnoty). Pokud by tomu tak nebylo, bylo by velmi lehké ekvalizací histogramu změnit barevnou kompozici, což u zpracování fotografií může být nepříjemné.

Grafické prostředí ImageFilters

- obsahuje výše jmenované filtry z praktické části



Obrázek 6.1 ImageFilters

7 Závěr

Tuto práci jsem věnoval filtračním metodám obrazu, kdy po zachycení snímku je často potřeba vyzískat maximum z požadovaného snímku. Během práce jsem se ujisti, že problematika filtračních metod je natolik hluboká, že není snadné, se ve všech filtračních metodách orientovat.

Tato práce mi přinesla nejedno úskalí v podobě programovacího jazyku JAVA, nad kterým jsem strávil převážnou část volného času, vlastně i na úkor širšího rozboru teoretické části. Bylo nelehké navrhnout rozhraní filtru takové, aby pod ním mohly spolehlivě pracovat všechny filtry. Nakonec se řešení ustálilo na třech bodech, a to typu filtru - bodový, lokální, globální, dále funkce pro analýzu obrazu před zobrazením ovládacího panelu, a funkce pro vlastní filtrování obrázku.

Liteatura

- [1] Dobeš, Michal. Zpracování a algoritmy v C#.Praha: Ben,2008.,s144.ISBN
- [2] Gonzales, R. C., Woods,R. E.: Digital Image Processing, Prentice Hall, 2002
- [3] GONZALEZ, WOODS, EDDINS. *Digital Image Processing Using MATLAB*. Upper Saddle River, N. J.: Prentice Hall, 2004. 609 s
- [4] Jahne and B. Jahne, *Digital Image Processing: Concepts, Algorithms and Scientific Applications*, 4th ed., Springer-Verlag, Berlin, 1997.
- [5] RÍHA, Kamil. Pokročilé techniky zpracování obrazu. Brno: VUT v Brně, 2007.126s.
- [6] Rudolf, Pecinovský. Myslíme objektivě v jazyku JAVA 5.0 Praha.,2004
- [7] Pavel, Herout. Java, Grafické uživatelské prostředí a čeština,2004 Koop
- [8] N. Efford, *Digital Image Processing: A Practical Introduction Using Java*, Addison Wesley,Reading MA, 2000.

Seznam příloh:

CD obsahuje ImageFilters, který implementuje návrh třídy filtrů v programovém vybavení JAVA.