

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

METODY PRO ODSTRANĚNÍ ŠUMU Z DIGITÁLNÍCH OBRAZŮ

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. ROMAN ČIŠECKÝ

BRNO 2012



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

METODY PRO ODSTRANĚNÍ ŠUMU Z DIGITÁLNÍCH OBRAZŮ

DIGITAL IMAGE NOISE REDUCTION METHODS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

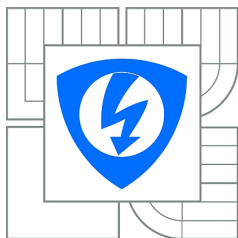
AUTOR PRÁCE
AUTHOR

Bc. ROMAN ČIŠECKÝ

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR ČÍKA, Ph.D.

BRNO 2012



**VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky
a komunikačních technologií**

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Roman Čišecký

ID: 106396

Ročník: 2

Akademický rok: 2011/2012

NÁZEV TÉMATU:

Metody pro odstranění šumu z digitálních obrazů

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte současné metody pro odstranění nežádoucího šumu z digitálních obrazů. Na základě získaných vědomostí navrhnete vlastní metodu (postup) pro odstranění šumu. Tuto metodu implementujete do aplikace RapidMiner a ověříte její funkčnost. Srovnáte ji s dosud známými technikami.

DOPORUČENÁ LITERATURA:

[1] BURGER, Wilhelm; BURGE, Mark J. Principles of Digital Image Processing: Fundamental Techniques. Londýn : Springer, 2009. 272 s. ISBN 978-1848001909.

[2] GONZALEZ, Rafael C.; WOODS, Richard E. Digital Image Processing. 3. Londýn : Pearson Pentice Hall, 2008. 954 s. ISBN 978-0-13-505267-9.

Termín zadání: 6.2.2012

Termín odevzdání: 24.5.2012

Vedoucí práce: Ing. Petr Číka, Ph.D.

Konzultanti diplomové práce:

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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.

ABSTRAKT

Diplomová práca sa zaoberá metódami na odstránenie šumu z digitálnych obrazov. V teoretickej časti sú vysvetlené niektoré základné pojmy súvisiace so spracovaním obrazu, obrazovým šumom, rozdelenie šumu a kritéria na určovanie kvality odšumovacieho procesu. Ďalej sú v práci popísané jednotlivé metódy na odstránenie šumu, spomenuté sú ich výhody a nevýhody. Praktická časť sa zaoberá vlastnou implementáciou vybraných metód v jazyku Java, v aplikácii RapidMiner. V závere sú porovnané výsledky dosiahnuté jednotlivými metódami.

KLÍČOVÁ SLOVA

šum, filtrácia obrazu, diskretná Fourierova transformácia, vlnková transformácia, diskretná kosínusová transformácia, Java, Wienerov empirický filter, NLM filter, waveletové prahovanie, RapidMiner

ABSTRACT

The master's thesis is concerned with digital image denoising methods. The theoretical part explains some elementary terms related to image processing, image noise, categorization of noise and quality determining criteria of denoising process. There are also particular denoising methods described, mentioning their advantages and disadvantages in this paper. The practical part deals with an implementation of the selected denoising methods in a Java, in the environment of application RapidMiner. In conclusion, the results obtained by different methods are compared.)

KEYWORDS

noise, image filtering, discrete Fourier transform, wavelet transform, discrete cosine transform, Java, Empirical Wiener filter, NLM filter, wavelet thresholding, RapidMiner

ČIŠECKÝ, Roman *Metody pro odstranění šumu z digitálních obrazu*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011. 64 s. Vedoucí práce byl Ing. Petr Číka, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Metody pro odstranění šumu z digitálních obrazů“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové 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.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Ďakujem vedúcemu diplomovej práce Ing. Petrovi Číkovi, Ph.D. za účinnú metodickú, pedagogickú a odbornú pomoc, za konzultácie a za ďalšie cenné rady pri vypracovávaní tejto diplomovej práce.

Brno

.....

(podpis autora)

Výzkum popsáný v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

OBSAH

1	Interpretácia obrazu	12
1.1	Základné vlastnosti obrazu a jeho spracovanie	12
1.2	Farebné modely	13
1.2.1	Farebné modely RGB a CMY(K)	13
1.2.2	Farebné modely YUV a YCbCr	13
1.3	Digitalizácia obrazu	15
1.4	Digitálny obraz a šum	16
1.5	Druhy šumu	17
1.6	Posudzovanie kvality	18
1.7	Šum metódy	19
2	Transformácie obrazu	21
2.1	Diskrétna Fourierova transformácia (DFT)	21
2.2	Diskrétna kosínusová transformácia (DCT)	21
2.3	Diskrétna vlnková transformácia (DWT)	22
3	Lokálne vyhladzovacie metódy	24
3.1	Gaussovo vyhladzovanie	24
3.2	Anizotropické filtre	25
3.3	Filtre okolia (Neighborhood filters)	26
4	Filtácia vo frekvenčnej oblasti	30
4.1	Fourier-Wienerov ideálny filter	31
4.2	Empirický Wienerov filter	31
4.3	Tvrde waveletové prahovanie	32
4.4	Mäkké waveletové prahovanie	33
4.5	Waveletové prahovanie invariantné voči posuvu	34
5	Nelokálne vyhladzovanie	36
5.1	Princíp Non Local-means (NLM) filtra	36
6	Implementácia vybraných metód	38
6.1	Implementácia empirického Wienerovho filtra	38
6.2	Implementácia waveletového prahovania	40
6.3	Implementácia NLM filtra	42
6.3.1	Paralelizácia kódu	44

7	Porovnanie dosiahnutých výsledkov	47
7.1	Operátor na porovnávanie kvality	47
7.2	Štatistické porovnanie kvality	47
7.3	Subjektívne porovnanie kvality	50
8	Záver	53
	Zoznam použitej literatúry	55
	Zoznam príloh	57
A	Prvá príloha	58
A.1	NLM filter – detailnejší popis blokov vývojového diagramu	58
A.2	Empirický wienerov filter – prahovanie	59
A.3	Empirický wienerov filter – výpočet matice DCT	60
A.4	Waveletové prahovanie – dekompozícia	61
A.5	Waveletové prahovanie – vývojový diagram prahovania	62
A.6	Použité obrázky	63
B	Druhá príloha	64
B.1	Obsah priloženého média	64

ZOZNAM OBRÁZKOV

1.1	Jednotková kocka modelu RGB a CMY.	14
1.2	Farebný model YCbCr.	15
2.1	Bloková schéma pyramidového algoritmu – 1. stupeň dekompozície. .	23
2.2	Ukážka dekompozície obrazu 1. stupňa pomocou 2D-DWT.	23
3.1	Vplyv lokálnych odšumovacích metód na prirodzený obraz.	29
4.1	Vplyv metód na odstránenie šumu vo frekvenčnej oblasti na obraz. .	35
5.1	Ukážka váhovania pixelov pri NLM algoritme.	37
6.1	Vývojový diagram empirického Wienerovho filtra.	39
6.2	Vývojový diagram waveletového prahovania.	41
6.3	Ukážka dekompozície 1. stupňa obrazu.	42
6.4	Vývojový diagram algoritmu NLM filtra.	46
7.1	Proces porovnávania metód v RapidMineri.	48
7.2	Hodnoty PSNR u rôznych obrázkov.	50
7.3	Filtrácia obrazu v RapidMineri	52

ÚVOD

Túžba zachytávať veci a dianie okolo seba bola človeku vlastná už od počiatkov jeho existencie. Využíval k tomu nástroje a techniky poplatné dobe. Počínajúc maľbami na stenách jaskýní, cez kresby a maľby na rôzne materiály, až po objavenie fotografie koncom 19. storočia. Rozvoj počítačov a digitálnej techniky priniesol vznik digitálnej formy obrazových dát. Takáto forma dát, hoci je pomerne stabilná, je často náchylná na rušenie. Toto rušenie, zvané šum, vzniká v obraze z rôznych príčin. Jednak môže šum vzniknúť v dôsledku fotenia za zhoršených svetelných podmienok, pri kompresii obrazu, prípadne môže vzniknúť už v elektronike zachytávacieho zariadenia.

Cieľ diplomovej práce je zameraný na oboznámenie sa problematikou odstraňovania šumu z digitálnych obrazov. V rámci tejto práce sú teoreticky rozobrané niektoré metódy používané na odstránenie šumu, pričom niektoré z popísaných metód budú aj prakticky implementované v aplikácii RapidMiner.

V prvej kapitole budú vysvetlené niektoré základné pojmy týkajúce sa spracovania obrazových dát. Nechýba popis najčastejšie používaných farebných modelov a prevodné vzťahy medzi nimi. Kapitola ďalej pojednáva o vzťahu šumu k vlastnému obrazu, budú popísané rôzne druhy šumu, ako aj kritéria na posudzovanie kvality filtrácie.

Druhá kapitola pojednáva o problematike obrazových transformácii, keďže pri spracovaní obrazových dát sa často využíva prevod týchto dát z priestorovej do transformačnej oblasti, predovšetkým do frekvenčnej oblasti. Popísané transformácie sa využívajú v praktickej časti práce pri implementácii konkrétnych metód.

Ďalšie tri kapitoly sa zaoberajú konkrétnymi metódami na odšumovanie obrazu. V tretej kapitole sú popísané lokálne vyhladzovacie filtre. Jedná sa o filtráciu v priestorovej oblasti. Štvrtá kapitola sa zaoberá filtráciou v oblasti frekvenčnej. Piata kapitola sa venuje nelokálnym metódam odšumovania obrazu. Každá metóda bude podrobne popísaná principiálne i matematicky, s dôrazom na odvodenie šumu danej metódy.

Šiesta kapitola je venovaná vlastnej implementácii vybraných metód. Prvou metódou je Wienerov empirický filter. Ide o filter vo frekvenčnej doméne využívajúci diskretnú kosínusovú transformáciu. Druhou implementovanou metódou je waveletové prahovanie, taktiež ide o frekvenčný filter, využívajúci tentokrát vlnkovú transformáciu. Poslednou implementovanou metódou je filtrácia tzv. NLM filtrom. Ide o filtráciu v priestorovej oblasti založená na váhovaní podobných bodov obrazu.

V poslednej kapitole sú jednotlivé metódy porovnané z hľadiska účinnosti odstraňovania šumu z obrazu. Toto porovnávanie je realizované jednak objektívne, na základe kvalitatívnych kritérií a jednak subjektívne, podľa vizuálneho vnemu. V závere nechýba vyhodnotenie dosiahnutých výsledkov.

1 INTERPRETÁCIA OBRAZU

1.1 Základné vlastnosti obrazu a jeho spracovanie

Definícia obrazu a jeho vznik: Obraz ako taký predstavuje dvojrozmernú projekciu trojrozmernej scény. Možno ho získať za pomoci vhodného prostriedku (napr. digitálny fotoaparát) v rôznych oblastiach viditeľného spektra (napr. denné svetlo), prípadne v iných oblastiach žiarenia, ktoré nie sme schopní okom zachytiť (napr. röntgenové alebo infračervené snímky).

Jas obrazu: Jas je veličina, ktorá určuje množstvo svetelnej energie prijatej zo zachytávanej scény. Veľkosť jasú závisí od viacerých faktorov, napr. od zdroja svetla, polohy zdroja voči zachytávanej scéne, od povrchu predmetov scény a ich reflektivity. Vo farebnom priestore sRGB sa pod jasom rozumie aritmetický priemer červenej, zelenej a modrej farebnej zložky.

Kontrast: Pod pojmom kontrast sa rozumie vlastnosť obrazu, ktorá určuje vzdialenosť medzi dvomi jasovými hladinami dvoch vedľajších oblastí obrazových bodov, ktoré majú rovnaký jas. Čím je táto vzdialenosť väčšia, tým je väčší aj kontrast.

Sýtosť farieb: Sýtosť je určená kombináciou intenzity svetla a jej rozložením do spektra rôznych vlnových dĺžok. Maximálnu sýtosť možno dosiahnuť pri farbe práve jednej vlnovej dĺžky s vysokou intenzitou svetla. Platí teda priama úmernosť medzi svetelnou intenzitou a sýtosťou.

Farebná hĺbka: Táto vlastnosť obrazu je určená počtom všetkých farieb, ktoré je možné v danom obraze uplatniť. Každý bod obrazu je daný hodnotou vyjadrenou v bitoch, v ktorej je zakódovaná farba. Napríklad farebná hĺbka s názvom *TrueColor* vyjadruje každý obrazový bod pomocou 24 bitov, na každú zložku RGB pripadá 8 bitov, teda 256 odtieňov. U monochromatického obrazu farebnú hĺbku nahrádzajú hladiny jasú, reprezentované úrovňami šedej v rozmedzí od čiernej po bielu.

Farebné vnímanie u človeka: Na sietnici ľudského oka sa nachádzajú 2 typy svetlocitlivých zrakových buniek: tyčinky (*angl. rods*) a čapíky (*angl. cones*).

- Tyčinky majú vysokú citlivosť, vytvárajú v mozgu šedotónový obraz (jas) a zaisťujú videnie pri nízkych hladinách osvetlenia.
- Čapíky sú receptory citlivé na určitú oblasť vlnových dĺžok a umožňujú tak farebný vnem. V oku sú tri druhy čapíkov citlivých zhruba na vlnové dĺžky zelenej (64 %), červenej (32 %) a modrej (2 %) farby. Keďže sú zastúpené nerovnomerne, citlivosť ľudského zraku na jednotlivé farby sa líši.

Tyčinky sú asi 10-krát citlivejšie ako čapíky a zároveň ich je zhruba 20-krát viac. Z tohto vyplýva, že ľudský zrak je výrazne citlivejší na jasové detaily ako na farebné detaily. Tento fakt sa využíva napr. pri kompresných technikách.

Informácie čerpané z [1, 2].

1.2 Farebné modely

Pod pojmom *farebný model* sa rozumie abstraktná matematická štruktúra, ktorá popisuje farby ako n -tice čísel, zvyčajne tri alebo štyri farebné zložky. Farebný model používa základné farby a miešaním týchto základných farieb možno dostať ostatné farby. Rozlišujeme dva typy miešania farieb:

- **aditívne miešanie:** jednotlivé farebné zložky sa sčítavajú a výsledkom je svetlo vyššej intenzity. Zmiešaním všetkých základných farieb vznikne biela.
- **subtraktívne miešanie:** pridaním každej ďalšej farby sa znižuje intenzita pôvodného svetla. Inak povedané, svetlo prechádzajúce jednotlivými farebnými vrstvami je čoraz viac pohlcované. Zmiešaním všetkých základných farieb vznikne čierna.

Informácie čerpané z [4]

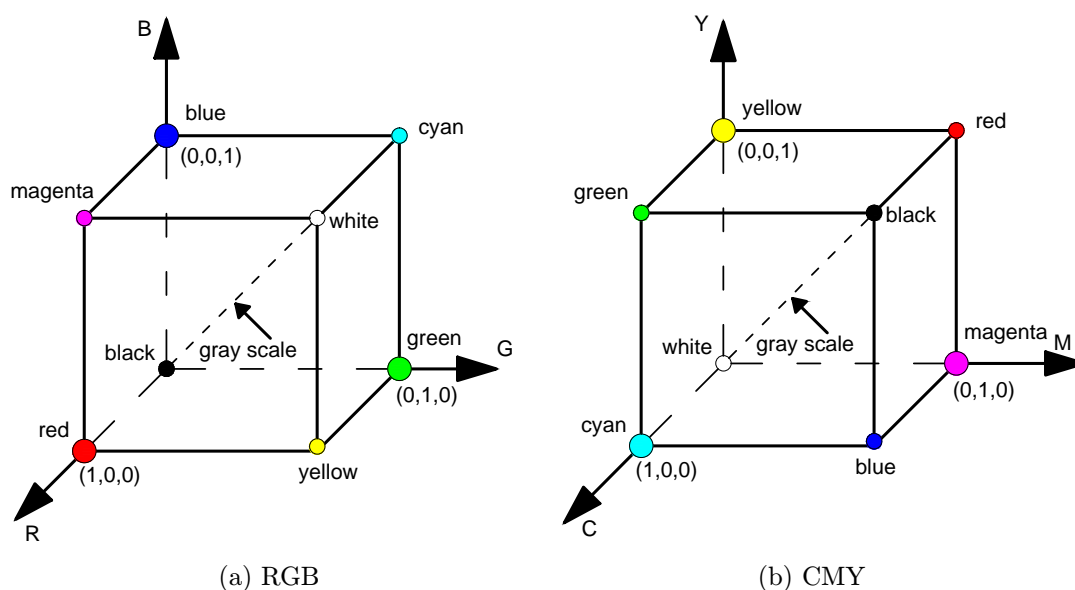
1.2.1 Farebné modely RGB a CMY(K)

Model RGB – základnými farbami sú červená (**red**), zelená (**green**) a modrá (**blue**). Patrí medzi aditívne modely a možno ho vyjadriť pomocou jednotkovej kocky, na ktorej sa v počiatku $(0, 0, 0)$ nachádza čierna farba, a v protiľahlom rohu $(1, 1, 1)$ biela, viď obrázok 1.1a. Vo všeobecnosti platí, že v protiľahlých vrchoch ležia vždy dve navzájom doplnkové farby, teda také, ktorých súčtom dostaneme bielu. Tento model sa najčastejšie využíva pre popis zobrazovacích jednotiek (monitory, skenery, a iné).

Model CMY – základnými farbami modelu sú azurová (**cyan**), purpurová (**magenta**) a žltá (**yellow**). Na rozdiel od RGB modelu ide o subtraktívny model. Jednotková kocka modelu CMY je znázornená na obrázku 1.1b. Je typický pre miešanie maliarskych alebo tlačiarenských farieb, preto sa hojne využíva hlavne v tlačiarenskej technike. Problémom modelu CMY je nedokonalé krytie jednotlivých farebných zložiek. Napríklad zmiešaním troch základných farieb nevznikne čistá čierna, ale len nejaká tmavohnedá farba. Z tohto dôvodu sa k týmto farbám pridáva ešte čierna farba (**black**) a vzniká tak model CMYK. Čerpané z [4]

1.2.2 Farebné modely YUV a YCbCr

Model YUV je základný farebný model používaný pri analógovom televíznom vysielaní (norma PAL). Hlavnou výhodou modelu bolo oddelenie luminančnej (jasovej) zložky (Y) od chrominančných (farbonosných) zložiek (U, V), čo pri nástupe farebnej televízie umožňovalo spätnú kompatibilitu s čierno-bielou televíziou. Pri prevode z modelu RGB je luminančná zložka vypočítaná váženým súčtom červenej, zelenej a modrej zložky, chrominančné zložky získame odčítaním luminančnej zložky



Obr. 1.1: Jednotková kocka modelu RGB a CMY.

od modrej, resp. červenej zložky. Prevodný vzorec má tvar:

$$\begin{aligned} Y &= 0,299R + 0,587G + 0,114B \\ U &= 0,492(B - Y) \\ V &= 0,877(R - Y) \end{aligned} .$$

Model YCbCr predstavuje digitálnu odnož modelu YUV. Bol vyvinutý ako súčasť doporučenia ITU-R BT.601. Oddelenie farebných zložiek sa využíva najmä pri kompresii digitálnych obrazov, napr. u formátu JPEG, pre ktorý je prevod z RGB definovaný takto:

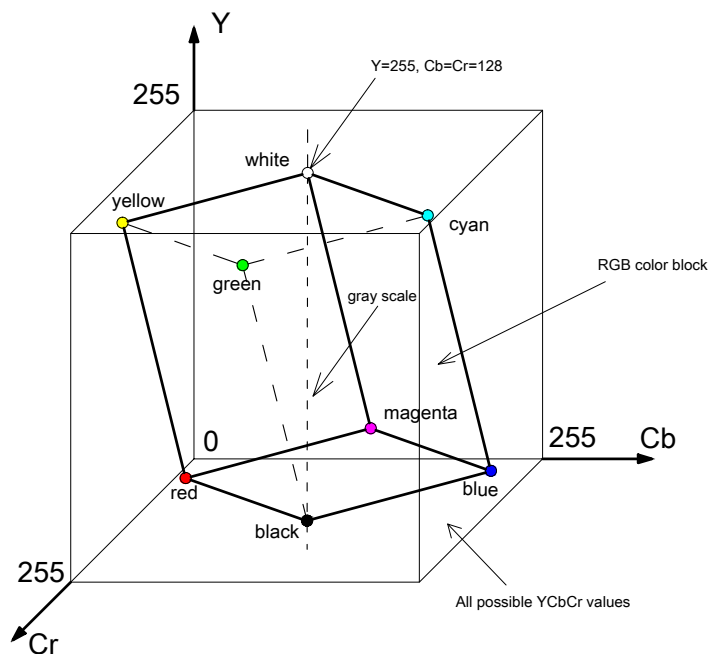
$$\begin{aligned} Y &= 0,299R + 0,587G + 0,114B \\ Cb &= -0,16874R - 0,33126G + 0,5B + 128 . \\ Cr &= 0,5R - 0,41869G - 0,08131B + 128 \end{aligned}$$

Spätný prevod z YCbCr je potom prepočítavaný podľa vzťahov:

$$\begin{aligned} R &= Y + 1,402Cr - 179,456 \\ G &= Y - 0,34414Cb - 0,71414Cr + 135,45984 . \\ B &= Y + 1,772Cb - 226,816 \end{aligned}$$

Na obrázku 1.2 je znázornený farebný priestor YCbCr, do ktorého je na základe vyššie uvedených prevodných vzťahov premietaný model RGB. Z obrázku je zrejmé, že farby RGB modelu zaberaajú len určitú časť farebného priestoru YCbCr, limitovanú nominálnym rozsahom. Z tohto dôvodu existuje mnoho YCbCr kombinácií,

ktoré majú v modeli RGB neplatné hodnoty. Pri prevode z modelu RGB do YCbCr vzniká aj určitá chyba spôsobená zaokrúhľovaním, keďže pri prepočte môžu výjsť niektoré hodnoty mimo nominálny rozsah $\langle 0; 255 \rangle$. Napríklad pri prevode červenej farby z RGB modelu $[255; 0; 0]$ do YCbCr získame súradnice $[76, 245; 84, 971; 255, 5]$. V tomto prípade je potrebné hodnotu Cr zložky orezať na 255. Čerpané z [4, 5].



Obr. 1.2: Farebný model YCbCr.

1.3 Digitalizácia obrazu

Statický obraz predstavuje v spracovaní signálov signál viacrozmerový. Obraz ako taký teda uvažujeme ako funkciu $f(x, y)$ dvoch priestorových rovinových súradníc x a y , ktorej funkčnou hodnotou je jas obrazu v danom mieste.

Pre digitálne spracovanie obrazu je potrebné realizovať operáciu na diskretizáciu súradníc a jas, konkrétne sa jedná o vzorkovanie a kvantovanie signálu.

Vzorkovanie signálu je realizované tak, že sa vodorovná os rozdelí na rovnomerné úseky a z každého úseku sa vyberie len jedna vzorka. Táto operácia je strátová, pretože zo spojitých čiar dostaneme len množinu diskretných bodov s intervalom použitej vzorkovacej frekvencie f_{vz} . Jej veľkosť je daná vzťahom

$$f_{vz} = \frac{1}{\Delta x}, \quad (1.1)$$

kde Δx je rozmedzie dvoch vzoriek. Platí tu Nyquistovo kritérium, ktoré hovorí, že signál možno obnoviť bez straty informácie, ak je vzorkovacia frekvencia f_{vz} aspoň

dvojnásobkom maximálnej frekvencie použitej v obraze, teda

$$f_{vz} > 2f_{\max}. \quad (1.2)$$

Pri nesplnení tejto podmienky dochádza k aliasingu. Pri aliasingu sa v obnovenom signáli mení frekvenčný obsah. Môžu nastať dva prípady:

- zmena amplitúd a fáz frekvencií, ktoré boli súčasťou pôvodného signálu,
- vznik zložiek na úplne nových frekvenciách.

Tieto zmeny sú spôsobené tým, že spektrálne zložky nad Nyquistovou frekvenciou sa pri nadvzorkovaní preklápajú a skresľujú tak zložky nižších frekvencií. Príkladom vplyvu aliasingu v obraze je napr. vznik *moiré* efektu, kedy pôvodné geometrické obrazce menia svoju štruktúru, tvar a podobne.

Kvantovanie je pojem označujúci prevod diskretného signálu na digitálny. Rozlišujeme dva typy kvantovania:

- uniformné – vzdialenosť medzi dvomi jasovými úrovňami je konštantná,
- neuniformné – interval medzi úrovňami jasu je rôzny.

Pri kvantovaní dochádza k strate informácie, vzniká tzv. *kvantizačný šum*, s typickým trojuholníkovým priebehom. Pre zobrazenie N jasových úrovní je potrebných k bitov, pričom platí: $N = 2^k$.

Pri vzorkovaní je teda obraz rozdelený na konečný počet obrazových bodov, ku ktorým je pri kvantovaní priradená konkrétna hodnota jasu z daného súboru diskretných hodnôt. Výsledkom je potom matica reálnych čísel s rozmermi $M \times N$ v tvare:

$$A = \begin{pmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{pmatrix}.$$

Hodnoty M a N môžu pri digitalizácii nadobúdať ľubovoľné prirodzené čísla, pričom každý jeden bod môže mať L diskretných úrovní šedej. Platí

$$L = 2^k, k = 0, 1, 2, \dots \quad (1.3)$$

Informácie boli čerpané z [1, 3, 4].

1.4 Digitálny obraz a šum

Potreba efektívnych metód na obnovu obrazu narastá s produkciou digitálnych obrazov a videí všetkých druhov, často získaných za zhoršených podmienok.

Digitálny obraz je vo všeobecnosti kódovaný ako matica šedotónových a farebných hodnôt. V prípade videa má táto matica 3 rozmery, kde tretím rozmerom je

čas. Každá dvojica $(i, u(i))$, kde $u(i)$ predstavuje hodnotu v bode i , sa nazýva obrazový element, skrátene pixel. V prípade šedotónového obrazu i predstavuje bod v dvojrozmernej mriežke a $u(i)$ je jeho reálna hodnota. Pri farebnom obraze $u(i)$ predstavuje trojicu hodnôt pre červenú, zelenú a modrú zložku.

Dvomi hlavnými obmedzeniami kvality obrazu sú rozmazanosť obrazu (angl. blur) a šum (angl. noise). Rozmazanosť obrazu je podstatná pre systémy snímania obrazu, keďže digitálny obraz má konečný počet vzoriek a musí spĺňať Shannon-Nyquistov vzorkovací teorém. Hodnoty pixelov $u(i)$ sú výsledkom merania svietivosti, zvyčajne realizovaného pomocou CCD (charge coupled device) snímačov. Každý CCD snímač predstavuje zhruba štvorec, v ktorom sa počíta počet prijatých fotónov za dobu určitú, ktorá zodpovedá času nasýtenia. Ak je zdroj svetla konštantný, počet fotónov prijatých každým pixelom sa pohybuje zhruba okolo ich priemeru, v súlade s centrálnou limitnou vetou. Inak povedané, pre n prijatých fotónov možno očakávať výkyvy v rádu \sqrt{n} . Každý snímač, ktorý nie je dostatočne chladný prijíma okrem toho parazitné tepelné fotóny. Výsledkom tohto rušenia je šum. Pri prvotnom hrubom odhade možno napísať:

$$v(i) = u(i) + n(i), \quad (1.4)$$

kde $i \in I$, $v(i)$ je získaná hodnota, $u(i)$ bude „skutočná“ hodnota pixelu i , a $n(i)$ je rušenie šumom. Veľkosť šumu závisí od signálu, teda $n(i)$ je väčšie vtedy, keď je väčšie $u(i)$. Šumový model, u ktorého normalizované hodnoty $n(i)$ a $n(j)$ dvoch rôznych pixelov sú považované za nezávislé náhodné veličiny, nazývame „biely šum“. Čerpané z [6].

1.5 Druhy šumu

V každom obraze je prítomný šum rôzneho pôvodu. Vyššie popísaný šum vzniká už v elektronike zachytávacieho zariadenia. Okrem toho môže šum vzniknúť v dôsledku fotenia za zhoršených svetelných podmienok, pri kompresii obrazu a podobne. Charakter šumu v obraze je často neznámy, no niektoré jeho vlastnosti sú zjavné a ľahko identifikovateľné, čo umožňuje zvoliť vhodný postup na jeho potlačenie. Šum je možné členiť na základe viacerých kritérií [7]:

1. Členenie podľa závislosti na obsahu obrazu:

- *Šum nezávislý na obraze* (napr. tepelný šum, šum komunikačného rušenia)
- *Šum závislý na obraze* (napr. filmové zrno) – v obraze sa vyskytuje s menšou pravdepodobnosťou, zvyčajne sa ťažko odstraňuje.

2. Členenie podľa veľkosti a rozloženia šumu:

- *Impulzný šum* (napr. salt & pepper) – má takmer binárne rozloženie extrémnych hodnôt, zvyčajne postihuje len izolované pixely alebo malé

skupiny pixelov, avšak hodnoty týchto pixelov musia byť považované za stratené.

- *Šedý šum* (napr. Gaussov šum) – zasahuje všetky pixely v obraze, ale hodnoty pixelov zväčša ostávajú zachované; hodnoty amplitúd šumu bývajú výrazne nižšie ako hodnoty pixelov.

3. Členenie podľa vzťahu obrazu k obsahu obrazu:

- *Aditívny šum* – najbežnejší prípad, vzniká pričítaním k pôvodným hodnotám pixelov a jeho spracovanie je najjednoduchšie.
- *Multiplikatívny šum* – vzniká vynásobením každej hodnoty pixelu amplitúdou šumu tohto pixelu.
- *Ostatné typy* (napr. konvolučný šum).

4. Členenie podľa charakteru šumu vo frekvenčnej oblasti:

- *Širokopásmový šum* – patrí sem väčšina vyššie uvedených druhov šumu
- *Úzkopásmový šum* – vzniká kombináciou viacerých úzkopásmových aditívnych alebo multiplikatívnych signálov s pásikovou štruktúrou alebo moiré vzormi.

1.6 Posudzovanie kvality

Kritérii na vyhodnotenie kvality filtrácie existuje mnoho, v tejto časti budú popísané najčastejšie používané.

- **Smerodajná odchýlka** σ – je vyjadrená ako odmocnina z rozptylu, teda ako odmocnina strednej hodnoty štvorca odchýlky veličiny X od strednej hodnoty tejto veličiny EX

$$\sigma = \sqrt{DX} = \sqrt{E(X - EX)^2} = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^N (x_i - \bar{x})^2}. \quad (1.5)$$

- **Súčet absolútnych chýb** (Sum of Absolute Errors) SAE – za predpokladu že $x(i, j)$ predstavuje originálny obraz a $\tilde{x}(i, j)$ je obraz po filtrácii, platí

$$SAE = \sum_{i=1}^M \sum_{j=1}^N |x(i, j) - \tilde{x}(i, j)|, \quad M, N, i, j \in \mathbb{Z}. \quad (1.6)$$

- **Stredná absolútna chyba** (Mean Absolute Error) MAE

$$MAE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |x(i, j) - \tilde{x}(i, j)|, \quad M, N, i, j \in \mathbb{Z}. \quad (1.7)$$

- **Stredná kvadratická chyba** (Mean Square Error) MSE

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |x(i, j) - \tilde{x}(i, j)|^2, \quad M, N, i, j \in \mathbb{Z}. \quad (1.8)$$

- **Pomer signálu k šumu** (Signal to Noise Ratio) SNR - možno ho definovať pri meraní úrovne šumu pomocou jeho smerodajnej odchýlky $\sigma(n)$ ako

$$SNR = \frac{\sigma(u)}{\sigma(n)}, \quad (1.9)$$

kde $\sigma(u)$ zodpovedá empirickej smerodajnej odchýlke obrazu u ,

$$\sigma(u) = \left(\frac{1}{|I|} \sum_{i \in I} (u(i) - \bar{u})^2 \right)^{\frac{1}{2}}, \quad (1.10)$$

a $\bar{u} = \frac{1}{|I|} \sum_{i \in I} u(i)$ je priemerná hodnota úrovne šedej. Smerodajnú odchýlku šumu možno získať empirickým meraním, prípadne výpočtom, ak je známy model šumu a jeho parametre.

- **Špičkový pomer signálu k šumu** (Peak Signal to Noise Ratio) $PSNR$

$$PSNR = 10 \log \frac{(2^n - 1)^2}{MSE} [dB], \quad (1.11)$$

kde n vyjadruje počet bitov.

Najlepším spôsobom testovania vplyvu šumu na digitálny obraz je pridať Gaussov biely šum do obrazu. Tento šum má rovnomerne a nezávisle rozložené Gaussovske reálne premenné. Pri šume so smerodajnou odchýlkou $\sigma(n) = 3$, teda pri $SNR \simeq \frac{60}{3} \simeq 20$ zvyčajne v obraze nie je viditeľný vplyv šumu. Dokonca aj pri použití bieleho šumu so $SNR = \frac{2}{1}$ ľudské oko v obraze rozozná všetky pôvodné detaily. Algoritmy na odstránenie šumu však nepoznajú rozdiel medzi drobnými detailami obrazu a šumom, a preto ich odstraňujú. V mnohých prípadoch potom vznikajú nové deformácie obrazu, napr. rozostrenie, šachovnicový efekt, schodiskový efekt, atď. V experimentálnych podmienkach je model šumu dokonale presný. Slabinou odšumovacích algoritmov je teda nedostatočný obrazový model. Všetky metódy predpokladajú, že šum je kmitavý, a obraz je hladký, prípadne po častiach hladký. Preto sa snažia oddeliť hladké časti (obraz) od tých oscilujúcich (šum). V skutočnosti však veľa jemných štruktúr v obraze je oscilačných, rovnako ako šum, a naopak biely šum má nízkofrekvenčné zložky, teda hladké oblasti. Z tohto dôvodu sú separačné metódy založené len na argumente hladkosti na odšumovanie obrazu nie sú veľmi vhodné. Čerpané z [6], [8].

1.7 Šum metódy

Všetky odšumovacie algoritmy sú závislé od parametra filtrácie h , ktorý určuje mieru filtrácie aplikovanej na zdrojový obraz. Pre väčšinu metód parameter h závisí na

odhade rozptylu šumu σ^2 . Možno definovať výsledok odšumovacej metódy D_h ako dekompozíciu (rozklad) ľubovoľného obrazu v ako

$$v = D_h v + n(D_h, v), \quad (1.12)$$

kde $D_h v$ je hladšie ako v a $n(D_h, v)$ je zložka obrazu považovaná metódou za šum. Avšak nestačí len vyhladiť obraz v s určitosťou, že $n(D_h, v)$ vyzerá ako šum. Novšie metódy si nevystačia len s vyhladením, ale snažia sa obnoviť stratené informácie z D_h, v . Treba sa teda ďalej zamerať na $n(D_h, v)$.

Nech u je (nie nutne zašumený) obraz a D_h odšumovací operátor, závislý na parametre metódy h . Potom možno definovať šum metódy ako rozdielnosť obrazu

$$n(D_h, u) = u - D_h(u). \quad (1.13)$$

Šum metódy by teda mal mal byť čo možno najviac podobný bielemu šumu a okrem toho, keďže je nežiadúce, aby pôvodný obraz bol pozmenený odšumovacou metódou, mal by byť tento šum čo najmenší. Informácie čerpané z [6].

2 TRANSFORMÁCIE OBRAZU

Pri mnohých technikách spracovania digitálnych obrazových dát sa využíva prevod týchto dát z časovej oblasti (v prípade obrazových dát hovoríme o priestorovej oblasti) do oblasti transformačnej. Pri odstraňovaní šumu z obrazu sa vo veľkej miere využívajú predovšetkým transformácie z priestorovej do frekvenčnej oblasti. V tejto kapitole budú stručne popísané niektoré z nich, konkrétne diskretná Fourierova transformácia, diskretná kosínusová transformácia a diskretná vlnková transformácia.

2.1 Diskretná Fourierova transformácia (DFT)

Podstata Fourierovej transformácie vychádza z myšlienky francúzskeho matematika Jean Baptiste Joseph Fouriera, podľa ktorej možno každú funkciu $f(x)$ rozložiť na súčet sínusových a kosínusových harmonických funkcií rôznych frekvencií, vynásobených váhovým koeficientom (v prípade periodickej funkcie, kedy hovoríme o *Fourierovej rade*), resp. váhovaciou funkciou (v prípade neperiodickej funkcie, jedná sa o *Fourierovu transformáciu*). Pri spracovaní obrazových dát možno využiť 2D-DFT, ktorá je definovaná vzťahom

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})}, \quad (2.1)$$

kde $f(x, y)$ predstavuje obrazovú funkciu v priestorovej oblasti s rozmermi $M \times N$, $F(u, v)$ je obraz danej funkcie vo frekvenčnej oblasti. Často sa používa skrátený zápis Fourierovej transformácie:

$$F(u, v) = \mathcal{F}(f(x, y)). \quad (2.2)$$

Inverzná 2D-DFT je definovaná vzťahom

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(\frac{ux}{M} + \frac{vy}{N})}, \quad (2.3)$$

skrátene

$$f(x, y) = \mathcal{F}^{-1}(F(u, v)). \quad (2.4)$$

Informácie čerpané z [3], [9].

2.2 Diskretná kosínusová transformácia (DCT)

Diskretná kosínusová transformácia vychádza z DFT a vyjadruje vzťah medzi vzorkami signálu v priestorovej a frekvenčnej oblasti. Existuje celkovo 8 variácií DCT,

ktoré sa líšia spôsobom vytvorenia periodickej postupnosti. V praxi najčastejšie používanou variantou je DCT typu II. U obrazových dát sa využíva dvojrozmerná DCT 2. typu (skrátene 2D DCT-II), ktorá je definovaná vzťahom

$$F(u, v) = C_u C_v \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cos \frac{(2x+1)u\pi}{2M} \cos \frac{(2y+1)v\pi}{2N}, \quad (2.5)$$

kde $f(x, y)$ predstavuje obrazovú funkciu v priestorovej oblasti s rozmermi $M \times N$, $F(u, v)$ je obraz danej funkcie vo frekvenčnej oblasti. Konštanty C_u a C_v sú definované ako

$$C_f = \begin{cases} \frac{1}{\sqrt{2}} & \text{pre } f = 0, \\ 1 & \text{inak.} \end{cases} \quad (2.6)$$

Pre inverznú 2D-DCT platí vzťah

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C_u C_v F(u, v) \cos \frac{(2x+1)u\pi}{2M} \cos \frac{(2y+1)v\pi}{2N}. \quad (2.7)$$

Informácie čerpané z [9], [10].

2.3 Diskrétna vlnková transformácia (DWT)

Vlnková transformácia umožňuje analýzu nestacionárnych signálov. Rozkladom signálu pomocou vlnkovej transformácie získame dvojrozmernú funkciu času a mierky (tzv. scalegram). Základnými bazovými funkciami tejto transformácie sú krátke vlnky, u ktorých dochádza k zmene časovej mierky a k posuvu v čase. Základná vlnka sa označuje ako materská vlnka (angl. mother wavelet) $\psi(t)$. Od tejto vlnky sú zmenou mierky a posunom po časovej osi odvodené ďalšie vlnky rovnakého tvaru, ktoré sú však roztiahnuté (expanzia) a posunuté. V rôznych oblastiach spracovania signálov sa používajú rôzne typy vlniek. Medzi najčastejšie používané patrí Haarova vlnka, vlnka mexický klobúk, Daubechies vlnka, atď.

Diskrétna vlnková transformácia je definovaná vzťahom

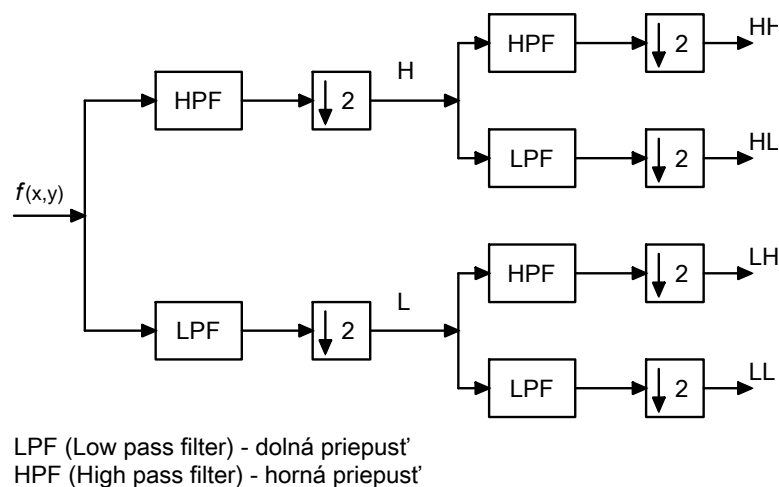
$$F(j, k) = 2^{-\frac{j}{2}} \sum_i f \psi(2^{-j}i - k), \quad (2.8)$$

kde ψ je materská vlnka, $i, j, k \in \mathbb{N}$, j je použitá miera, k je posun materskej vlnky. Spätná diskretná transformácia je definovaná vzťahom

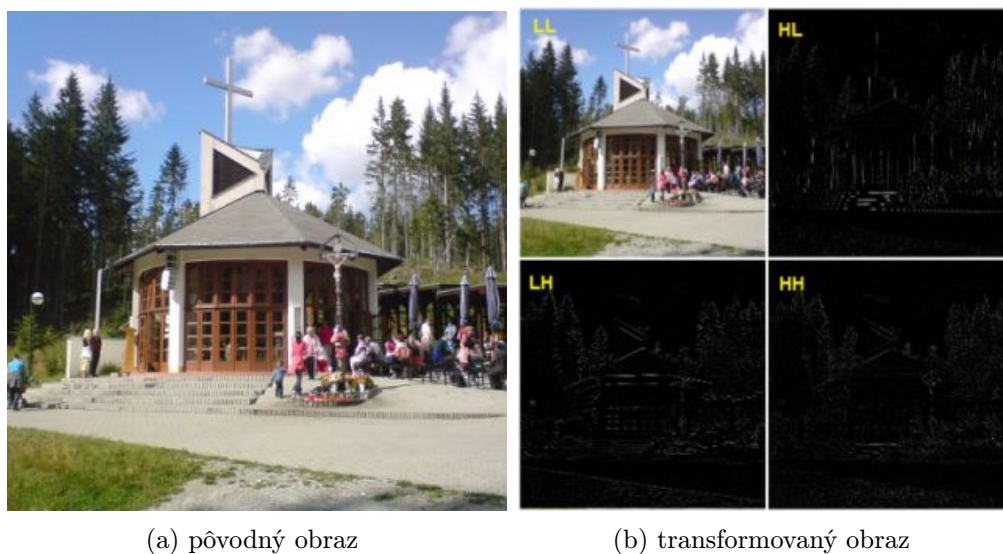
$$f(i) = \sum_j \sum_k F(j, k) 2^{-\frac{j}{2}} \psi(2^{-j}i - k). \quad (2.9)$$

Pri práci s obrazom je potrebné využiť 2D-DWT, ktorá vychádza z vyššie uvedených vzťahov. Táto transformácia je založená na myšlienke rozloženia na aproximčné koeficienty (nízko-frekvenčné zložky) a detailné koeficienty (vysoko-frekvenčné

zložky). Aproximačné zložky možno v ďalšom kroku opäť rozložiť. Tomuto postupu sa hovorí dekompozícia a môže byť aplikovaná aj viacnásobne. Na obrázku 2.1 vidieť blokovú schému tzv. pyramidového algoritmu, ktorý realizuje dekompozíciu obrazu prvého stupňa. Najskôr je realizovaná filtrácia po riadkoch, následne po stĺpcoch obrazovej matice. Vzniknú 4 typy koeficientov: aproximačné koeficienty LL, horizontálne koeficienty LH, vertikálne koeficienty HL a diagonálne koeficienty HH (viď obrázok 2.2). Po následnom spracovaní koeficientov (napr. prahovaním) je realizovaná rekonštrukcia obrazu, analogicky opačným postupom. Informácie čerpané z [9], [10], [11].



Obr. 2.1: Bloková schéma pyramidového algoritmu – 1. stupeň dekompozície.



Obr. 2.2: Ukážka dekompozície obrazu 1. stupňa pomocou 2D-DWT.

3 LOKÁLNE VYHLADZOVACIE METÓDY

Originálny obraz u je definovaný v uzavretej oblasti $\Omega \subset \mathbb{R}^2$ a označuje sa ako $u(\mathbf{x})$ pre $\mathbf{x} = (x, y) \in \mathbb{R}^2$. Tento spojitý obraz je väčšinou interpretovaný ako Shannonova interpolácia diskkrétnej mriežky vzoriek a je preto analytický. Vzdialenosť medzi dvomi po sebe idúcimi vzorkami bude označená ako ε .

Samotný šum predstavuje diskrétny jav na vzorkovacej mriežke. Vzorky šumu n_i neinterpolujeme ako pásmovo obmedzenú funkciu, ale radšej ako po častiach konštantnú funkciu, ktorá je konštantná vo všetkých pixeloch i a rovná hodnote n_i .

Platí, že $|\mathbf{x}| = (x^2 + y^2)^{\frac{1}{2}}$ a $\mathbf{x}_1 \cdot \mathbf{x}_2 = x_1x_2 + y_1y_2$ je ich skalárny súčin a parciálne derivácie obrazu u sú $u_x = \frac{\partial u}{\partial x}$, $u_y = \frac{\partial u}{\partial y}$, $u_{xy} = \frac{\partial^2 u}{\partial x \partial y}$. Gradient obrazu u možno definovať ako $D_u = (u_x, u_y)$ a jeho Laplacián ako $\Delta u = u_{xx} + u_{yy}$. Čerpané z [6].

3.1 Gaussovo vyhladzovanie

Podľa Rieszovho teorému spočíva izotropická lineárna filtrácia obrazu v konvolúcii obrazu s lineárnou radiálnou maskou. Požiadavky na vyhladenie obrazu sú teda vyjadrené v konvolučnej maske. Gaussova maska má tvar: $\mathbf{x} \rightarrow G_h(\mathbf{x}) = \frac{1}{4\pi h^2} e^{-\frac{|\mathbf{x}|^2}{4h^2}}$, kde G_h má smerodajnú odchýlku h . Šum metódy konvolúcie s Gaussovým jadrom G_h je

$$u - G_h * u = -h^2 \Delta u + o(h^2). \quad (3.1)$$

Podobný výsledok platí pre všetky kladné radiálne masky s obmedzeným rozptýlením, teda ak smerodajná odchýlka h je dostatočne malá. Na druhej strane vlastnosti redukcie šumu sú závislé na skutočnosti, že okolie zahrnuté do vyhladzovania musí byť dostatočne veľké na to, aby bolo možné šum redukovať priemerovaním. Za predpokladu, že $h = k\varepsilon$, kde k je počet vzoriek funkcie u a šumu n na intervale dĺžky h , musí byť priestorový pomer k omnoho väčší ako 1 na zaistenie redukcie šumu.

Vplyv Gaussovho vyhladzovania na šum možno ukázať na pixelu $i = 0$. Na tomto pixeli,

$$G_h * n(0) = \sum_{i \in I} \int_{P_i} G_h(\mathbf{x}) n(\mathbf{x}) d\mathbf{x} = \sum_{i \in I} \varepsilon^2 G_h(i) n_i, \quad (3.2)$$

kde $n(\mathbf{x})$ je interpolovaná ako po častiach konštantná funkcia, štvorcové pixely P_i so stredom v bode i majú veľkosť ε^2 , a $G_h(i)$ označuje strednú hodnotu funkcie G_h na pixelu i .

Označením $Var(\mathbf{X})$ ako rozptyl náhodnej veličiny \mathbf{X} a dosadením za rozptyly nezávislých náhodných premenných dostaneme

$$Var(G_h * n(0)) = \sum_i \varepsilon^4 G_h(i)^2 \sigma^2 \simeq \sigma^2 \varepsilon^2 \int G_h(\mathbf{x})^2 d\mathbf{x} = \frac{\varepsilon^2 \sigma^2}{8\pi h^2}. \quad (3.3)$$

Teda platí nasledujúci teorém: nech $n(x)$ je po častiach konštantný biely šum, kde pre každý štvorcový pixel i platí $n(\mathbf{x}) = n_i$. Za predpokladu, že hodnoty n_i sú nezávislé a rovnomerne rozložené s nulovou strednou hodnotou a rozptylom σ^2 , pozostatok šumu po konvolúcii bude

$$Var(G_h * n(0)) \simeq \frac{\varepsilon^2 \sigma^2}{8\pi h^2}. \quad (3.4)$$

Inými slovami, smerodajná odchýlka šumu, ktorá predstavuje amplitúdu šumu, je násobená výrazom $\frac{\varepsilon}{h\sqrt{8\pi}}$.

Vyššie uvedený teorém popisuje krehkú rovnováhu medzi redukciou šumu a poškodením obrazu pri lineárnom vyhladzovaní. Odšumovanie nepozmení pôvodný obraz v prípade, ak hladkosť v rozsahu h je omnoho väčšia ako vzorkovací rozsah ε . Gaussovská konvolúcia dosahuje optimálne výsledky pri harmonických funkciách, a naopak, horšie výsledky pri jednotvárných častiach obrazu u , napr. hrany či textúry (viď obrázky 3.1), kde má Laplacián vysoké hodnoty. Informácie boli čerpané z [6].

3.2 Anizotropické filtre

Anizotropický filter (AF) sa snaží predísť efektu rozostrenia vznikajúceho u Gaussovho filtru konvolúciou obrazu u iba v smere kolmom na $Du(\mathbf{x})$. Pre \mathbf{x} teda možno napísať

$$AF_h u(\mathbf{x}) = \int G_h(t) u\left(x + t \frac{Du(\mathbf{x})^\perp}{|Du(\mathbf{x})|}\right) dt, \quad (3.5)$$

kde $Du(\mathbf{x}) \neq 0$ a kde $(x, y)^\perp = (-y, x)$ a $G_h(t) = \frac{1}{\sqrt{2\pi}h} e^{-\frac{t^2}{2h^2}}$ je jednorozmerná Gaussova funkcia s rozptylom h^2 . Šum metódy anizotropickej filtrácie AF_h je

$$u(\mathbf{x}) - AF_h u(\mathbf{x}) \simeq -\frac{1}{2} h^2 D^2 u \left(\frac{Du^\perp}{|Du|}, \frac{Du^\perp}{|Du|} \right) = -\frac{1}{2} h^2 |Du| \text{curv}(u)(\mathbf{x}), \quad (3.6)$$

ak $Du(\mathbf{x}) \neq 0$.

Výraz $\text{curv}(u)(\mathbf{x})$ označuje zakrivenie, t. j. prevrátená hodnota polomeru krivosti vrstevnice prechádzajúcej bodom \mathbf{x} . Ak $Du(\mathbf{x}) \neq 0$, platí

$$\text{curv}(u) = \frac{u_{xx}u_y^2 - 2u_{xy}u_xu_y + u_{yy}u_x^2}{(u_x^2 + u_y^2)^{\frac{3}{2}}}. \quad (3.7)$$

Šum tejto metódy je nulový všade tam, kde sa obraz u správa lokálne ako funkcia jednej premennej, $u(x, y) = f(ax + by + c)$. V tomto prípade bude vrstevnica obrazu u lokálne priamkou popísanou rovnicou $ax + by + c = 0$ a gradient funkcie f bude veľmi veľký. Z toho vyplýva, že anizotropická filtrácia zachováva hrany v obraze. Čo týka redukcie Gaussovho šumu, šum na súradnicovej sieti nie je izotropický, preto Gaussov priemer v prípade, ak Du je paralelné s jednou súradnicovou osou, je zostavený z takmer $\sqrt{2}$ -krát viac vzoriek ako Gaussov priemer v uhlopriečnom smere.

Pri anizotropickom vyhladzovaní, kedy ε je dostatočne malý v porovnaní s h , zostatku šumu v obraze zodpovedá

$$\text{Var}(AF_h(n)) \leq \frac{\varepsilon}{\sqrt{2\pi}h} \sigma^2. \quad (3.8)$$

Inými slovami, smerodajná odchýlka šumu n je násobená činiteľom rovným maximálne $(\frac{\varepsilon}{\sqrt{2\pi}h})^{\frac{1}{2}}$, pričom táto maximálna hodnota býva dosiahnutá na uhlopriečkach.

Existuje veľa verzií anizotropických filtrov, ako napr. známy *mediánový filter*. Všetky tieto filtre majú spoločné to, že dokážu dobre uchovávať hrany v obraze, avšak slabšie výsledky dosahujú pri rovných oblastiach, kde sú horšie ako Gaussovo rozostrenie, viď obrázok 3.1. Čerpané z [6].

3.3 Filtre okolia (Neighborhood filters)

Predchádzajúce filtre sú založené na predstave priestorového okolia alebo príbuzenstva. Oproti tomu filtre okolia kladú dôraz na hodnotu úrovne šedej pri definícii susedných pixelov. V najjednoduchšom prípade bude odšumená hodnota pixelu i predstavovať priemer hodnôt všetkých pixelov, ktoré majú úroveň šedej blízku $u(i)$. Okolie je teda definované ako

$$B(i, h) = \{j \in I \mid u(i) - h < u(j) < u(i) + h\}. \quad (3.9)$$

Keďže na odhad hodnoty pixelu i sa používajú pixely z celého obrazu, jedná sa o nelokálny algoritmus. Algoritmus možno popísať vzťahom:

$$NF_h u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{\Omega} u(\mathbf{y}) e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} d\mathbf{y}, \quad (3.10)$$

kde $\Omega \subset \mathbb{R}^2$ je otvorená a ohraničená množina a $C(\mathbf{x}) = \int_{\Omega} e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} d\mathbf{y}$ je normalizačný faktor. Prvou otázkou, ktorá sa pri tomto type filtra ponúka, je zhodnosť, teda do akej miery je obraz po odšumení podobný pôvodnému obrazu.

Predpokladajme, že u je lipschitzovsky spojitá funkcia na oblasti Ω , kde Ω je otvorená a ohraničená podmnožina množiny \mathbb{R}^2 . Potom $|u(\mathbf{x}) - NF_h u(\mathbf{x})| = O(h\sqrt{-\log h})$ pre všetky malé h , $0 < h < 1$, $\mathbf{x} \in \Omega$.

Nech teda \mathbf{x} je bod v množine Ω a pre zvolené B a h , kde $B, h \in \mathbb{R}$, uvažujme množinu $D_h = \{\mathbf{y} \in \Omega \mid |u(\mathbf{y}) - u(\mathbf{x})| \leq Bh\}$. Potom platí:

$$|u(\mathbf{x}) - NF_h u(\mathbf{x})| \leq \frac{1}{C} \int_{D_h} e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} |u(\mathbf{y}) - u(\mathbf{x})| d\mathbf{y} + \frac{1}{C} \int_{D_h^c} e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} |u(\mathbf{y}) - u(\mathbf{x})| d\mathbf{y}. \quad (3.11)$$

Na jednej strane, ak uvažujeme $\int_{D_h} e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} d\mathbf{y} \leq C(\mathbf{x})$ a $|u(\mathbf{y}) - u(\mathbf{x})| \leq Bh$ pre $\mathbf{y} \in D_h$ možno vidieť, že prvý výraz je ohraničený Bh . Na druhej strane, ak uvažujeme $e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} \leq e^{-B^2}$ pre $\mathbf{x} \notin D_h$, $\int_{D_h^c} |u(\mathbf{y}) - u(\mathbf{x})| d\mathbf{y}$ je ohraničený $C \geq O(h^2)$, z čoho možno vyvodiť, že druhý výraz bude mať rád $O(h^{-2}e^{-B^2})$. Napokon, ak za B zvolíme $B^2 = -3 \log h$, dostaneme

$$|u(\mathbf{x}) - NF_h u(\mathbf{x})| \leq Bh + O(h^{-2}e^{-B^2}) = O(h\sqrt{-\log h}) + O(h), \quad (3.12)$$

z čoho vyplýva, že šum metódy dosahuje rádu $O(h\sqrt{-\log h})$.

Yaroslavského filter okolia uvažuje tzv. zmiešané okolie, $B(i, h) \cap B_\rho(i)$, kde $B_\rho(i)$ je kruh so stredom i a polomerom ρ . Metóda teda počíta priemer hodnôt pixelov, ktoré sú si blízke jednak priestorovo, a jednak úrovňou šedej. Tento filter možno popísať v spojitom tvare ako:

$$YNF_{h,\rho}(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{B_\rho(\mathbf{x})} u(\mathbf{y}) e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} d\mathbf{y}, \quad (3.13)$$

kde $C(\mathbf{x}) = \int_{B_\rho(\mathbf{x})} e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}} d\mathbf{y}$ je normalizačný faktor.

Pre jednoduchosť bude najkôr odvodenie šumu metódy $YNF_{h,\rho}$ vysvetlené na príklade 1D signálu u . Predpokladajme, že $u \in \mathcal{C}^2((a, b))$, $a, b \in \mathbb{R}$. Potom pre $0 < \rho \ll h$ a $h \rightarrow 0$ platí

$$u(s) - YNF_{h,\rho} u(s) \simeq -\frac{\rho^2}{2} f\left(\frac{\rho}{h} |u'(s)|\right) u''(s), \quad (3.14)$$

kde

$$f(t) = \frac{\frac{1}{3} - \frac{3}{5}t^2}{1 - \frac{1}{3}t^2}.$$

Z vyššie uvedených závislostí vyplýva, že šum tejto metódy je úmerný druhej derivácii signálu. To znamená, že sa správa ako vyvážená tepelná rovnica. Funkcia f dáva znamienko a amplitúdu tejto tepelnej rovnice. Ak funkcia f nadobúda kladné hodnoty, šum metódy sa správa ako pravá tepelná rovnica, ak nadobúda záporné hodnoty, správa sa ako prevrátená tepelná rovnica. Nulové body a body nespojitosti funkcie f predstavujú singulárne body, v ktorých sa mení správanie metódy. Veľkosť tejto zmeny je omnoho väčšia v blízkosti bodov nespojitosti funkcie f , kde vzniká šok efekt.

Podobné výsledky možno dosiahnuť aj v prípade dvojrozmerného obrazu. Predpokladajme, že $u \in C^2(\Omega)$, $\Omega \subset \mathbb{R}^2$. Potom pre $0 < \rho \ll h$ a $h \rightarrow 0$ platí

$$u(\mathbf{x}) - YNF_{h,\rho}u(\mathbf{x}) \simeq -\frac{\rho^2}{8} \left(g\left(\frac{\rho}{h}|Du|\right) u_{\eta\eta} + h\left(\frac{\rho}{h}|Du|\right) u_{\xi\xi} \right), \quad (3.15)$$

kde

$$u_{\eta\eta} = D^2u \left(\frac{Du}{|Du|}, \frac{Du}{|Du|} \right), \quad u_{\xi\xi} = D^2u \left(\frac{Du^\perp}{|Du|}, \frac{Du^\perp}{|Du|} \right) \quad (3.16)$$

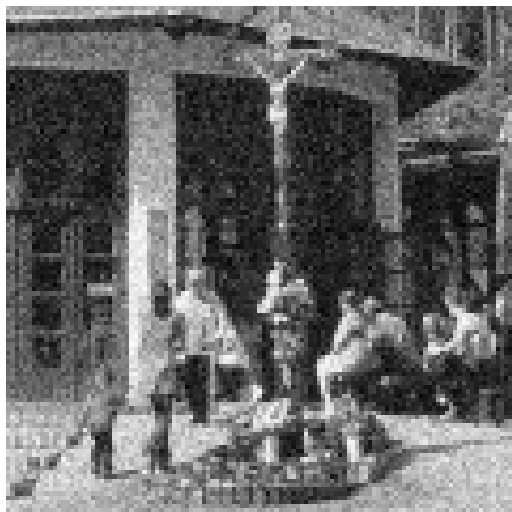
a

$$g(t) = \frac{1 - \frac{3}{2}t^2}{1 - \frac{1}{4}t^2}, \quad h(t) = \frac{1 - \frac{1}{2}t^2}{1 - \frac{1}{4}t^2}. \quad (3.17)$$

Šum metódy filtru okolia možno napísať ako súčet rozptylového člena v tangentovom smere $u_{\xi\xi}$ a rozptylového člena v normálovom smere $u_{\eta\eta}$. Znamienko a veľkosť oboch členov závisí na znamienku a veľkosti funkcií g a h . Obe funkcie môžu nadobúdať kladné i záporné hodnoty. Preto sa oba rozptylové členy môžu javiť ako smerová priama tepelná rovnica, resp. smerová prevrátená tepelná rovnica, v závislosti na hodnote gradientu. Obdobne ako v 1D prípade, algoritmus sa chová ako filtračný, resp. vylepšovací algoritmus, v závislosti na veľkosti gradientu. Ak $B_1 = \frac{\sqrt{2}}{\sqrt{3}}$ a $B_2 = \sqrt{2}$, resp. vyznačením nulových bodov funkcií g a h , možno rozlíšiť nasledovné prípady:

- Ak $0 < |Du| < B_2 \frac{h}{\rho}$ algoritmus sa správa ako Perona-Malik filter [12]. V prvom kroku je aplikovaná tepelná rovnica, ale akonáhle $|Du| > B_1 \frac{h}{\rho}$ normálový rozptyl sa zmení na prevrátený, ktorý vylepšuje hrany, zatiaľ čo tangentový rozptyl ostáva kladný.
- Ak $|Du| > B_2 \frac{h}{\rho}$, bude aplikovaná na základe hodnoty gradientu buď tepelná rovnica alebo prevrátená tepelná rovnica. Zmena správania medzi týmito dvomi dynamikami je poznamenaná asymptotickými nespojitostami, ktoré vedú k zvýšenému šok efektu.

Informácie čerpané z [6].



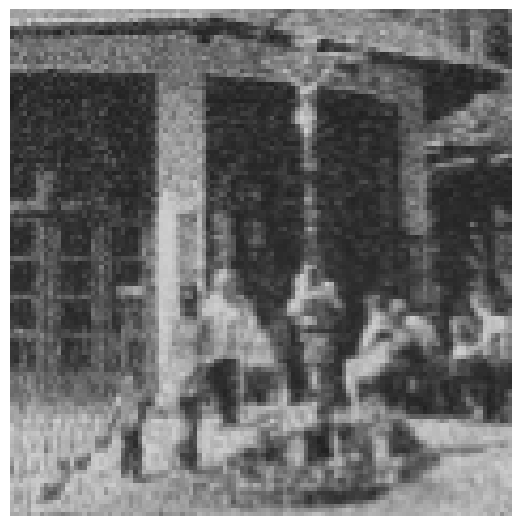
(a) Zašumený obraz



(b) Gaussov filter



(c) Anizotropický filter



(d) Yaroslavského filter okolia

Obr. 3.1: Vplyv lokálnych odšumovacích metód na prirodzený obraz.

4 FILTÁCIA VO FREKVENČNEJ OBLASTI

Nech u je pôvodný obraz definovaný na súradnicovej sieti I a do tohoto obrazu je pridaný biely šum N . N je náhodný proces, ktorého hodnoty $N(i)$ sú nezávislé a rovnomerne rozložené s nulovou strednou hodnotou a konštantným rozptylom σ^2 . Výsledný proces je teda závislý na náhodnej šumovej zložke a je prezentovaný náhodným poľom V ,

$$V(i) = u(i) + N(i). \quad (4.1)$$

Ak teda $n(i)$ predstavuje pozorovaný šum, $v(i)$ označuje pozorovaný zašumený obraz,

$$v(i) = u(i) + n(i). \quad (4.2)$$

Nech $\mathcal{B} = \{g_\alpha\}_{\alpha \in A}$ je pravouhlá oblasť množiny $\mathbb{R}^{|I|}$. Výsledný zašumený proces možno transformovať na tvar:

$$V_{\mathcal{B}}(\alpha) = u_{\mathcal{B}}(\alpha) + N_{\mathcal{B}}(\alpha), \quad (4.3)$$

kde

$$V_{\mathcal{B}}(\alpha) = \langle V, g_\alpha \rangle, \quad u_{\mathcal{B}}(\alpha) = \langle u, g_\alpha \rangle, \quad N_{\mathcal{B}}(\alpha) = \langle N, g_\alpha \rangle, \quad (4.4)$$

sú skalárne súčiny V , u , N a g_α , pričom $g_\alpha \in \mathcal{B}$. Šumové koeficienty $N_{\mathcal{B}}(\alpha)$ zostávajú nekorelované, s nulovou strednou hodnotou, ale rozptyly sú násobené hodnotou $\|g_\alpha\|^2$:

$$\begin{aligned} E[N_{\mathcal{B}}(\alpha)N_{\mathcal{B}}(\beta)] &= \sum_{m,n \in I} g_\alpha(m)g_\beta(n)E[N(m)N(n)] = \\ &= \langle g_\alpha, g_\beta \rangle \sigma^2 = \sigma^2 \|g_\alpha\|^2 \delta[\alpha - \beta]. \end{aligned} \quad (4.5)$$

Pri filtrácii vo frekvenčnej oblasti býva najskôr filter aplikovaný nezávisle na všetky transformované koeficienty $V_{\mathcal{B}}(\alpha)$ a následnou spätnou transformáciou nových koeficientov vznikne výsledný obraz. Pôvodné koeficienty $V_{\mathcal{B}}(\alpha)$ sú modifikované na $a(\alpha)V_{\mathcal{B}}(\alpha)$. Keďže $a(\alpha)$ je závislé na hodnote $V_{\mathcal{B}}(\alpha)$, ide o nelineárny algoritmus. Spätnou transformáciou možno dostať odhad

$$\hat{U} = DV = \sum_{\alpha \in A} a(\alpha) V_{\mathcal{B}}(\alpha) g_\alpha. \quad (4.6)$$

D sa taktiež nazýva *diagonálny operátor*. Tento odhad je ovplyvnený určitou chybou, ktorá je založená na kvadratickej euklidovskej vzdialenosti a je priemerovaná cez rozloženie šumu. Príkladom frekvenčného filtra ktorý sa používa na minimalizáciu tejto chyby je *Fourier-Wienerov filter*. Čerpané z [6].

4.1 Fourier-Wienerov ideálny filter

Nech u je pôvodný obraz, N je biely šum, $V = u + N$ je obraz so šumom a D je frekvenčný filter. Potom možno chybu filtra D definovať ako

$$r(D, u) = E\{\|u - DV\|^2\}, \quad (4.7)$$

kde pravdepodobnosť je získaná z rozloženia šumu. Diagonálny operátor D_{inf} , ktorý minimalizuje túto chybu možno zapísať ako:

$$D_{inf} = \arg \min_D r(D, u). \quad (4.8)$$

Operátor D_{inf} je daný množinou $\{a(\alpha)\}_\alpha$, kde

$$a(\alpha) = \frac{|u_{\mathcal{B}}(\alpha)|^2}{|u_{\mathcal{B}}(\alpha)|^2 + \|g_\alpha\|^2 \sigma^2}, \quad (4.9)$$

a tomu odpovedajúca chyba je

$$r_{inf}(u) = \sum_{s \in S} \|g_\alpha\|^4 \frac{|u_{\mathcal{B}}(\alpha)|^2 \sigma^2}{|u_{\mathcal{B}}(\alpha)|^2 + \|g_\alpha\|^2 \sigma^2}. \quad (4.10)$$

Predchádzajúci operátor potlačuje všetky šumové koeficienty za účelom minimalizácie chyby. Obmedzením hodnôt $a(\alpha)$ na hodnoty 0 alebo 1 možno získať operátor, ktorý časť koeficientov ponechá bez zmeny a zvyšnú časť odstráni. Tento operátor je definovaný množinou $\{a(\alpha)\}_\alpha$, kde

$$a(\alpha) = \begin{cases} 1 & \text{pre } |u_{\mathcal{B}}(\alpha)|^2 \geq \|g_\alpha\|^2 \sigma^2, \\ 0 & \text{inak} \end{cases} \quad (4.11)$$

a odpovedajúca chyba je

$$r_p(u) = \sum \|g_\alpha\|^2 \min(|u_{\mathcal{B}}(\alpha)|^2, \|g_\alpha\|^2 \sigma^2). \quad (4.12)$$

Fourier-Wienerov filter sa nazýva taktiež ideálny, pretože je závislý na koeficientoch $u_{\mathcal{B}}(\alpha)$ pôvodného obrazu, ktoré však väčšinou nie sú známe. Použitím Fourierovej transformácie môžu globálne charakteristiky obrazu prevažovať nad lokálnymi, čo sa v obraze potom prejaví vytváraním parazitných periodických vzorov. Aby sa tomu zabránilo, je treba brať do úvahy viac lokálnych rysov, ako tomu je napr. pri waveletovej alebo lokálnej DCT transformácii. Informácie čerpané z [6], [3].

4.2 Empirický Wienerov filter

Tento filter patrí do skupiny lokálne adaptívnych filtrov v transformačnej oblasti. U týchto filtrov ja zašumený obraz analyzovaný pomocou pohybujúceho sa okna.

V každej pozícii tohto okna sa vypočíta jeho spektrum a následne sa toto spektrum upraví. Napokon je aplikovaná spätná transformácia, pomocou ktorej sa odhaduje iba hodnota centrálného pixelu v danom okne.

Nech $i \in I$ predstavuje pixel a $W = W(i)$ je okno so stredom v pixelu i . Následne je pomocou DCT transformácie vypočítané a modifikované spektrum okna W . V ďalšom kroku sa odhadnú koeficienty pôvodného obrazu v okne W , $u_{\mathcal{B},W}(\alpha)$ a je na ne aplikovaný vhodne zvolený útlm. Nakoniec sa použije práve pixel zo stredu takto upraveného okna. Za účelom aproximácie koeficientov $u_{\mathcal{B},W}(\alpha)$ pôvodného obrazu možno použiť priemerné hodnoty aditívneho šumového modelu, teda

$$E|V_{\mathcal{B},W}(\alpha)|^2 = |u_{\mathcal{B},W}(\alpha)|^2 + \sigma^2 \|g_\alpha\|^2. \quad (4.13)$$

Označením $\mu = \sigma \|g_\alpha\|$ možno neznáme pôvodné koeficienty zapísať ako

$$|u_{\mathcal{B},W}(\alpha)|^2 = E|V_{\mathcal{B},W}(\alpha)|^2 - \mu^2. \quad (4.14)$$

Pozorované koeficienty $|v_{\mathcal{B},W}(\alpha)|^2$ sú použité na aproximáciu $E|V_{\mathcal{B},W}(\alpha)|^2$ a odhadované pôvodné koeficienty sú nahradené optimálnym útlmom, čo vedie k množine $\{a(\alpha)\}_\alpha$, kde

$$a(\alpha) = \max \left\{ 0, \frac{|v_{\mathcal{B},W}(\alpha)|^2 - \mu^2}{|v_{\mathcal{B},W}(\alpha)|^2} \right\}. \quad (4.15)$$

Nech $EW F_\mu(i)$ značí filter získaný z množiny vyššie popísaných koeficientov, u je obraz definovaný na súradnicovej sieti I , $i \in I$ je pixel, a $W = W(i)$ je okno so stredom v pixelu i . Potom šum metódy možno definovať ako

$$u(i) - EW F_\mu(i) = \sum_{\alpha \in \Lambda} v_{\mathcal{B},W}(\alpha) g_\alpha(i) + \sum_{\alpha \notin \Lambda} \frac{\mu^2}{|v_{\mathcal{B},W}(\alpha)|^2} v_{\mathcal{B},W}(\alpha) g_\alpha(i), \quad (4.16)$$

kde $\Lambda = \{\alpha \mid |v_{\mathcal{B},W}(\alpha)| < \mu\}$.

Prítomnosť hrán v okne W spôsobuje výskyt veľkého počtu vysokých koeficientov. V dôsledku rušenia týchto koeficientov pri filtrácii vznikajú vo výslednom obraze oscilácie, ktoré sú bádateľné vo forme šachovnicových vzorov. Informácie čerpané z [6].

4.3 Tvrdé waveletové prahovanie

Nech $\mathcal{B} = \{g_\alpha\}_{\alpha \in A}$ je ortonormálna báza vlniek (waveletov). Procedúra nazvaná *tvrdé vlnkové prahovanie* predstavuje operátor, ktorý aproximuje ideálnu projekciu (viď rovnica 4.9) a koeficienty nižšie ako zvolený prah μ odstráni,

$$a(\alpha) = \begin{cases} 1 & \text{pre } |v_{\mathcal{B}}(\alpha)| > \mu, \\ 0 & \text{inak.} \end{cases} \quad (4.17)$$

Postup tvrdého prahovania $HWT_\mu(v)$ je založený na myšlienke, že samotný obraz je reprezentovaný veľkými vlnkovými koeficientmi, ktoré sú zachované, zatiaľ čo šum je rozložený v malých koeficientoch, ktoré sa odstraňujú. Účinnosť tejto metódy je značne závislá na schopnosti aproximovať pôvodný obraz u malým množstvom veľkých koeficientov.

Šum metódy tvrdého prahovania $HWT_\mu(u)$ možno za predpokladu, že u je obraz definovaný na súradnicovej sieti I , zapísať ako

$$u - HWT_\mu(u) = \sum_{\{\alpha \mid |u_{\mathcal{B}}(\alpha)| < \mu\}} u_{\mathcal{B}}(\alpha) g_\alpha. \quad (4.18)$$

Bohužiaľ, prítomnosť hrán v obraze znamená výskyt značného množstva vlnkových koeficientov nižších ako nastavený prah, avšak nie nulových. Odstránením týchto koeficientov z obrazu vznikajú malé oscilácie v blízkosti hrán, známe ako Gibbsov jav. Tieto rušivé vlnky sa v dôsledku odstránenia nízkych koeficientov objavujú aj v odfiltrovanom obraze. Informácie čerpané z [6].

4.4 Mäkké waveletové prahovanie

Metóda mäkkého prahovania umožňuje čiastočne obmedziť nežiaduce efekty, vznikajúce pri filtrácii tvrdým prahovaním. Pre operátor tejto metódy označený ako $SWT_\mu(v)$ platí

$$a(\alpha) = \begin{cases} \frac{v_{\mathcal{B}}(\alpha) - \text{sgn}(v_{\mathcal{B}}(\alpha))\mu}{v_{\mathcal{B}}(\alpha)} & \text{pre } |v_{\mathcal{B}}(\alpha)| \geq \mu, \\ 0 & \text{inak.} \end{cases} \quad (4.19)$$

Redukciou oscilácií v blízkosti nespojitostí mäkké prahovanie lepšie zachováva štruktúru vlnkových koeficientov. Metóda mäkkého prahovania sa vyznačuje tým, že tlmí všetky koeficienty s cieľom znížiť úroveň šumu.

Za predpokladu, že u je obraz definovaný na súradnicovej sieti I , možno šum metódy mäkkého prahovania $HWT_\mu(u)$ zapísať ako

$$u - SWT_\mu(u) = \sum_{\{\alpha \mid |u_{\mathcal{B}}(\alpha)| < \mu\}} u_{\mathcal{B}}(\alpha) g_\alpha + \mu \sum_{\{\alpha \mid |u_{\mathcal{B}}(\alpha)| > \mu\}} \text{sgn}(u_{\mathcal{B}}(\alpha)) g_\alpha. \quad (4.20)$$

Stanovenie hodnoty prahu možno ukázať na jednoduchom príklade. Za predpokladu, že pôvodný obraz u je nulový, platí $v_{\mathcal{B}}(\alpha) = n_{\mathcal{B}}(\alpha)$, a teda prah μ musí byť vyšší ako maximálny koeficient šumu pre zaistenie dostatočného potlačenia šumu a obnovenia pôvodného obrazu. V praxi je optimálna hodnota prahu μ veľmi vysoká a býva tak odstránených veľa koeficientov, ktoré nie sú tvorené šumom. Preto býva často prah zvolený nižší ako optimálny, aby sa zaručili lepšie výsledky. Pre tvrdé prahovanie sa používa hodnota $\mu = 3\sigma$. Pre mäkké prahovanie je táto hodnota príliš vysoká, preto sa používa hodnota $\mu = \frac{3}{2}\sigma$. Informácie čerpané z [6].

4.5 Waveletové prahovanie invariantné voči posuvu

Metóda vznikla zdokonalením vlnkových prahovacích metód priemerovaním odhadu všetkých posuvov degradovaného signálu. Ak $c^p(i)$ značí posunutý signál $v(i - p)$, vlnkové koeficienty pôvodného a posunutého signálu sa môžu podstatne líšiť, pričom navzájom nesúvisia jednoduchou výmenou poradia či posuvom,

$$v_{\mathcal{B}}^p(\alpha) = \langle v(n - p), g_{\alpha}(n) \rangle = \langle v(n), g_{\alpha}(n + p) \rangle. \quad (4.21)$$

Vektory $g_{\alpha}(n + p)$ vo všeobecnosti nepatria do báze $\mathcal{B} = \{g_{\alpha}\}_{\alpha \in A}$, a preto odhad posunutého signálu nesúvisí s odhadom signálu v . Tento nový algoritmus dáva odhad \hat{u}^p pre každý posunutý signál v^p pôvodného obrazu,

$$\hat{u}^p = Dv^p = \sum_{\alpha \in A} a(\alpha) v_{\mathcal{B}}^p(\alpha) g_{\alpha}. \quad (4.22)$$

Prahovanie invariantné voči posuvu, označované ako $TIWT(v)$, sa dosiahne spriemerovaním všetkých týchto čiastočných odhadov po posuve v opačnom zmysle,

$$\frac{1}{|I|} \sum_{p \in I} \hat{u}^p(i + p). \quad (4.23)$$

Účinok Gibbsovho javu sa pri prahovaní invariantnému voči posuvu značne zníži, pretože priemerovanie rôznych odhadov obrazu obmedzuje vznik oscilácii v blízkosti nespojitostí obrazu. Informácie boli čerpané z [6].



(a) Pôvodný obraz



(b) Zášumený obraz



(c) Wienerov filter



(d) Empirický Wienerov filter



(e) Tvrdé prahovanie



(f) Mäkké prahovanie

Obr. 4.1: Vplyv metód na odstránenie šumu vo frekvenčnej oblasti na obraz.

5 NELOKÁLNE VYHLADZOVANIE

Lokálne vyhladzovacie metódy a frekvenčné filtre sa zameriavajú na redukciu šumu a na rekonštrukciu hlavných geometrických charakteristík obrazu, ale väčšinou nedokážu uchovať jemné štruktúry, detaily a textúry pôvodného obrazu. Na základe pravidelnostných predpokladov predošlé metódy z pôvodného obrazu odfiltrujú detaily a jemné štruktúry obrazu, nakoľko sa vo všetkých funkčných aspektoch chovajú ako šum. Non Local-means algoritmus, popísaný v tejto kapitole, využíva vysoký stupeň nadbytočnosti (redundancie) každého prirodzeného obrazu. Pod týmto sa jednoducho myslí to, že ku každému malému oknu v prirodzenom obraze možno v tom istom obraze nájsť veľa podobných okien. Táto metóda je inšpirovaná myšlienkou filtru okolia, popísaného v kapitole 3.3. Vo všeobecnosti možno za „okolie pixelu i “ považovať akúkoľvek množinu pixelov j v tomto obraze, ak platí, že okno okolo pixelu j vyzerá podobne ako okno okolo pixelu i . Všetky pixely tohto okolia možno zahrnúť do odhadu novej hodnoty pixelu i .

5.1 Princíp Non Local-means (NLM) filtra

Keďže uvažovaný obraz je definovaný na diskretnej súradnicovej sieti, je potrebný diskretný popis algoritmu. Ak $v = \{v(p) | p \in I\}$ označuje diskretný obraz so šumom, odhadovaná hodnota odšumeného obrazu $NL(v(p))$ sa vypočíta ako vážený priemer všetkých pixelov v obraze,

$$NL(v(p)) = \sum_{q \in I} w(p, q) v(q), \quad (5.1)$$

kde váhy $\{w(p, q)\}_q$ závisia na podobnosti medzi pixelami p a q a splňajú zvyčajné podmienky $0 \leq w(p, q) \leq 1$ a $\sum_j w(p, q) = 1$.

Aby bolo možné vypočítať podobnosť medzi pixelmi obrazu, je potrebné na súradnicovej sieti I definovať tzv. *systém okolia* (z angl. *neighborhood system*). Systém okolia na súradnicovej sieti I je skupina takých $\mathcal{N} = \{\mathcal{N}_p\}_{p \in I}$ podmnožín množiny I , kde pre všetky $p \in I$ platí $p \in \mathcal{N}_p$ a súčasne $q \in \mathcal{N}_q \Rightarrow p \in \mathcal{N}_q$. Podmnožina \mathcal{N}_p sa nazýva okolie alebo okno podobnosti pixelu p .

Okná podobnosti môžu nadobúdať rôzne veľkosti a tvary za účelom lepšej adaptácie na konkrétny obraz. Pre jednoduchosť a lepšiu názornosť sú v tejto práci uvažované štvorcové okná s pevne definovanou veľkosťou $2f + 1 \times 2f + 1$ pixelov, kde f je polomer tohto okna. Obmedzenie obrazu v na okolie \mathcal{N}_p bude označované ako $v(\mathcal{N}_p)$:

$$v(\mathcal{N}_p) = (v(q), q \in \mathcal{N}_p). \quad (5.2)$$

Podobnosť medzi dvomi pixelmi p a q bude závisieť na podobnosti medzi intenzitami úrovne šedej vektorov $v(\mathcal{N}_p)$ a $v(\mathcal{N}_q)$. Pixely s podobným šedotónovým okolím, ako je $v(\mathcal{N}_p)$, budú mať pri priemerovaní vyššiu váhu, viď obrázok 5.1. Pixely $q1$ a $q2$ na obrázku majú veľké váhy, pretože ich okná podobnosti sú podobné oknu pixelu p . Naopak, váha pixelu $q3$ bude malá.

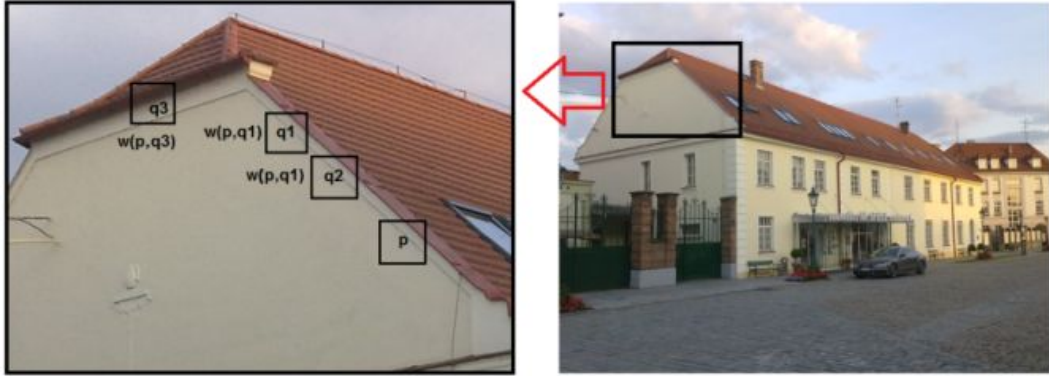
Aby bolo možné vypočítať podobnosť intenzít šedotónových úrovní vektorov $v(\mathcal{N}_p)$ a $v(\mathcal{N}_q)$, je potrebné najskôr vypočítať Gaussovy váženú Euklidovskú vzdialenosť, označenú ako $d^2(v(\mathcal{N}_p) - v(\mathcal{N}_q))$. Táto miera sa veľmi dobre dokáže prispôbiť akémukoľvek prídavnému bielemu šumu v obraze. Platí:

$$d^2(v(\mathcal{N}_p) - v(\mathcal{N}_q)) = \frac{1}{3(2f+1)^2} \sum_{j \in (0,f)} (v(p+j) - v(q+j))^2. \quad (5.3)$$

Váhy priradené ku štvorcovým vzdialenostiam sú definované ako

$$w(p, q) = \frac{1}{C(p)} e^{-\frac{\max(d^2 - 2\sigma^2, 0)}{h^2}}, \quad C(p) = \sum_q e^{-\frac{\max(d^2 - 2\sigma^2, 0)}{h^2}}, \quad (5.4)$$

kde σ je smerodajná odchýlka šumu, $C(p)$ je normalizačný faktor, a parameter h je filtračný parameter, ktorý riadi pokles exponenciálnej funkcie a tým pádom aj pokles hodnôt váh ako funkcie Euklidovských vzdialeností. Čerpané z [6].



Obr. 5.1: Ukážka váhovania pixelov pri NLM algoritme.

6 IMPLEMENTÁCIA VYBRANÝCH METÓD

Teoretické princípy jednotlivých metód na odšumovanie digitálnych obrazov boli popísané v predchádzajúcich kapitolách. V tejto kapitole bude rozobraná praktická realizácia týchto metód v programovacom jazyku Java, pre platformu RapidMiner.

V rámci tejto diplomovej práce boli implementované metódy filtrácie vo frekvenčnej oblasti pomocou empirického Wienerovho filtru, waveletového prahovania a NLM filtru v podobe operátorov v prostredí RapidMiner. Okrem toho bol vytvorený operátor na porovnávanie kvality obrazu, pomocou ktorého boli vytvorené algoritmy porovnávané s už existujúcimi operátormi z hľadiska účinnosti redukcie šumu.

6.1 Implementácia empirického Wienerovho filtra

Algoritmus bol pôvodne implementovaný ako konzolová aplikácia a následne bol pretvorený do podoby operátora v prostredí RapidMiner. Pozostáva z nasledujúcich častí:

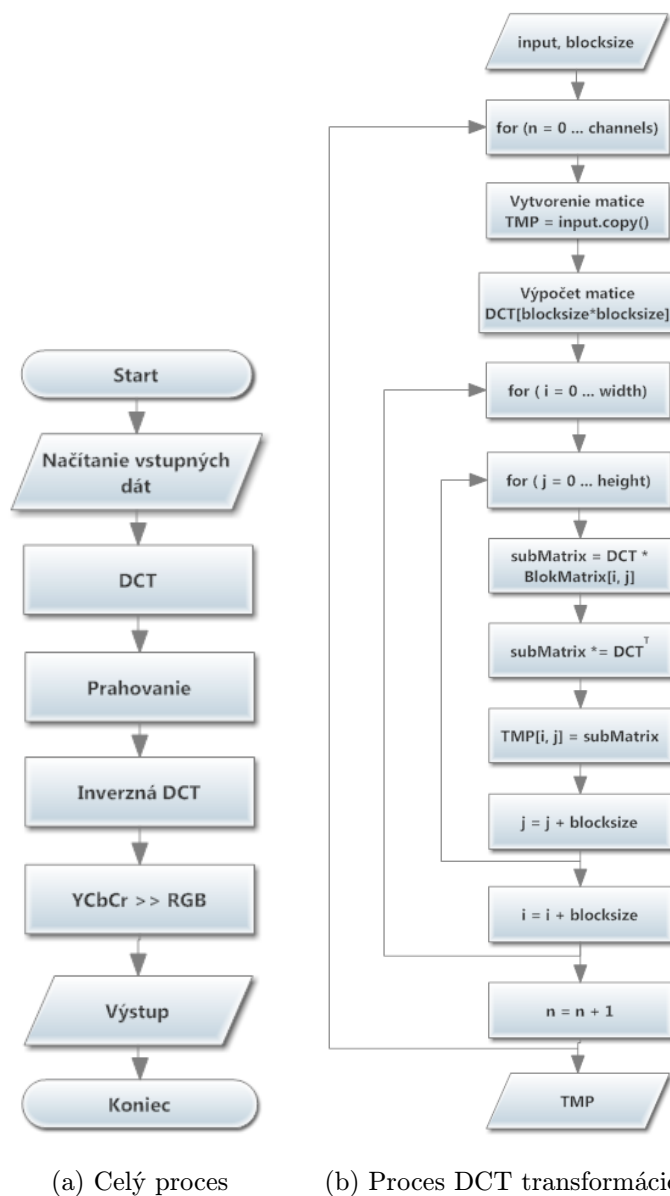
- predspracovanie vstupného obrazu
- transformácia DCT
- prahovanie
- spätná DCT transformácia

Operátor má tri parametre:

- *Block size* – udáva veľkosť bloku pri DCT transformácii; na výber je viacero možností.
- *Threshold* – slúži na stanovenie referenčnej hodnoty prahu pri prahovaní. Možno zadávať hodnoty v rozsahu $\langle 0; 255 \rangle$.
- *Attenuation* – určuje hodnotu útlmu, ktorou budú násobené koeficienty nižšie ako je zvolený prah. Nadobúda hodnoty v intervale $\langle 0; 1 \rangle$. Ak je nastavený na hodnotu 0, všetky koeficienty nižšie ako prah budú zrušené.

V rámci predspracovania obrazu je obraz najskôr prevedený do farebného priestoru *YCbCr*. Jednotlivé zložky sú uložené do samostatných matíc, ktoré sa využívajú pri ďalšom spracovaní. Algoritmus bol navrhovaný tak, aby dokázal spracovať tak šedotónové, ako aj farebné obrazy.

Pri pretváraní aplikácie do podoby operátora v RapidMineri bola časť predspracovania obrazu realizovaná mimo vlastný operátor, ktorý tak realizuje len samotnú filtráciu. Na prevod do farebného priestoru *YCbCr* je použitý operátor **Color Component Extractor**, ktorý zo vstupného obrazu extrahuje jednotlivé zložky. Na vstupy operátora **Empirical Wiener filter** sú potom privedené tieto zložky. Celkový vývojový diagram procesu je znázornený na obrázku 6.1a.



Obr. 6.1: Vývojový diagram empirického Wienerovho filtra.

Pri realizácii doprednej a spätnej DCT transformácii boli využité vzťahy (2.5) a (2.7). Pri výbere veľkosti blokov transformácie treba brať ohľad na rozmery vstupného obrazu. Oba rozmery musia byť deliteľné veľkosťou bloku. Najčastejšie používaná veľkosť bloku 8×8 túto podmienku u štandardných rozmerov väčšinou spĺňa. Problém môže nastať pri väčších blokoch. Vývojový diagram znázorňujúci proces doprednej DCT transformácie je na obrázku 6.1b. Na začiatku je potrebné vytvoriť kópiu obrazových dát vstupného obrazu, je teda vytvorená matica TMP . Následne sa vypočíta transformačná matica DCT na základe definovanej veľkosti bloku. Samotný proces tohto výpočtu je znázornený vývojovým diagramom v prílohe A.3. Po získaní

tejto matice sa prechádza maticou **TMP** po jednotlivých blokoch, pričom sa vždy násobí transformačná matica s aktuálnym blokom ($DCT \times \text{BlockMatrix}$). Výsledok tohto maticového súčinu sa ešte vynásobí transponovanou transformačnou maticou DCT^T . Tento výsledok sa uloží na príslušnú pozíciu výslednej matice **TMP** a pokračuje sa ďalším blokom. Proces inverznej transformácie prebieha podobne, s tým rozdielom, že spomínané operácie maticového násobenia prebiehajú v opačnom poradí.

Pri prahovaní sa jednotlivé DCT koeficienty v obraze porovnávajú s hodnotou prahu (definovanou parametrom *threshold*). Ak je hodnota konkrétneho koeficientu vyššia ako prahová hodnota, je koeficient ponechaný, inak je daný koeficient potlačený (jeho hodnota je násobená hodnotou útlmu definovanou parametrom *attenuation*). Tento proces je znázornený vývojovým diagramom v prílohe A.2.

6.2 Implementácia waveletového prahovania

Podobne ako u predchádzajúcej metódy, aj algoritmus waveletového prahovania bol najskôr implementovaný a testovaný ako konzolová aplikácia a následne bol pretvorený do podoby operátora v prostredí RapidMiner. V tomto operátore sú implementované hneď dve metódy, konkrétne tvrdé waveletové prahovanie (viď kap. 4.3) a mäkké waveletové prahovanie (viď kap. 4.4). Operátor má tieto parametre:

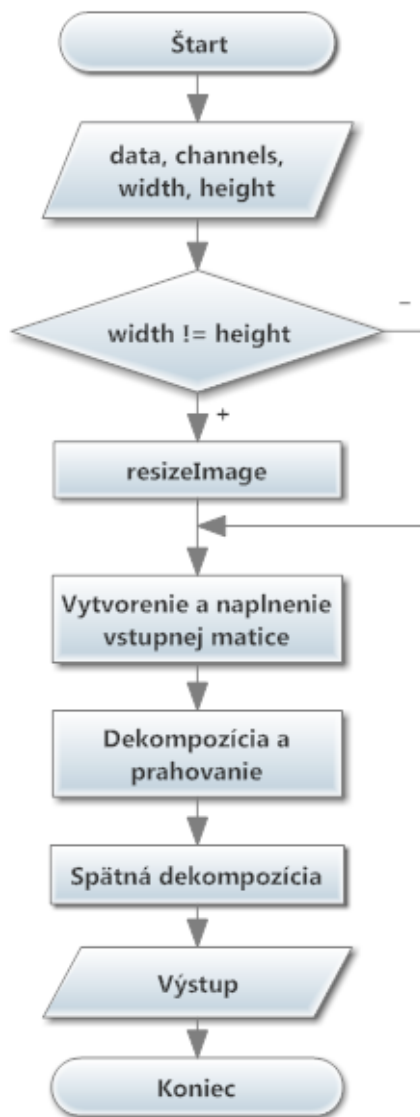
- *Method* – slúži na výber metódy prahovania.
- *Threshold* – slúži na stanovenie referenčnej hodnoty prahu. Možno zadávať hodnoty v rozsahu $\langle 0; 255 \rangle$.
- *Level of decomposition* – udáva stupeň dekompozície obrazu, možno zadávať celočíselné hodnoty v rozsahu $\langle 0; 6 \rangle$.

Samotný algoritmus pozostáva z nasledujúcich častí:

- predspracovanie vstupného obrazu
- DWT obrazu pomocou Haarovej vlnky
- prahovanie detailných koeficientov
- spätná DWT transformácia

Algoritmus je zjednodušene znázornený pomocou vývojového diagramu na obrázku 6.2.

V predspracovaní obrazu sa jednotlivé zložky obrazu (R, G, B) uložia do samostatných matíc. Keďže pri dekompozícii obrazu sa využívajú maticové operácie (napr. transpozícia, maticový súčin), je potrebné, aby vstupný obraz mal štvorcové rozmery, aby nedochádzalo ku kolízii indexovania pixelov. Z tohto dôvodu sú v rámci predspracovania overované rozmery vstupného obrazu. Ak obraz nevyhovuje podmienke, je pomocou metódy `resizeImage()` doplnený o pixely s hodnotou 0.



Obr. 6.2: Vývojový diagram waveletového prahovania.

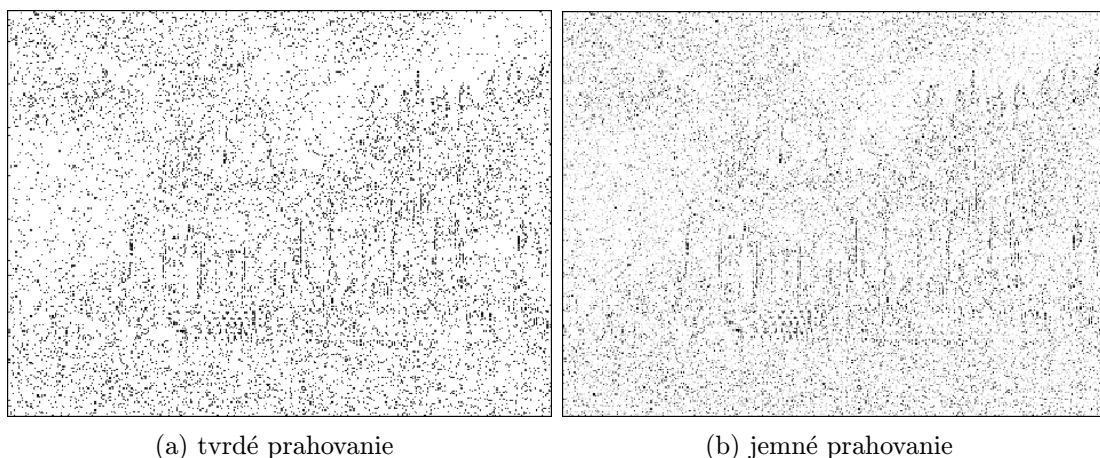
Dopredná a spätná DWT transformácia je realizovaná pomocou Haarovej vlnky. Jedná sa o najjednoduchšiu ortogonálnu waveletovú transformáciu. Jej výhodou je rýchly výpočet, nevýhodou väčšia nespojitosť ako napríklad u vlniek Daubechies.

Algoritmus realizujúci dekompozíciu je znázornený vývojovým diagramom v prílohe A.4. Na začiatku sa vytvorí pomocné pole matíc $F_q[n]$ pre n kanálov vstupného obrazu. Samotné obrazové dáta sa uložia do matice $DWT[n]$. Pri prechádzaní obrazu sa zoberú vždy hodnoty dvoch susedných pixelov matice DWT (v rámci riadku) a vypočíta sa ich priemer a rozdiel. Výsledky sa uložia do matice F_q v tvare:

$$\begin{bmatrix} a_{12} & a_{34} & b_{12} & b_{34} \\ a_{56} & a_{78} & b_{56} & b_{78} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix},$$

kde prvky a predstavujú priemery a prvky b rozdiely hodnôt dvoch susedných pixelov. Po naplnení celej matice sa táto matica transponuje a celý cyklus sa opakuje. Týmto spôsobom získame prvý stupeň dekompozície. Ďalší stupeň získame opakovaním vyššie uvedeného procesu aplikovaného na „prvý kvadrant“ obrazu, ktorý predstavuje aproximačné koeficienty. Proces možno opakovať teoreticky až kým nedostaneme submatice o veľkosti 1×1 , ale vytvorená aplikácia je obmedzená do šiesteho stupňa dekompozície.

Pri prahovaní sa opäť prechádza obrazom pixel po pixelu, cez všetky kanály obrazu. Na základe vybranej metódy (tvrdé alebo mäkké prahovanie) je potom realizované samotné prahovanie podľa vzťahov 4.17, resp. 4.19. Prahujú sa len detailné koeficienty, aproximačné koeficienty ostávajú nepozmenené. Rozdiel medzi výstupmi pri metóde tvrdého a mäkkého prahovania najlepšie vidieť z detailných koeficientov, ktoré vzniknú po dekompozícii obrazu. Pri mäkkom prahovaní sú koeficienty vyššie ako prahová hodnota potlačené a maximálne hodnoty koeficientov nižších ako prahová hodnota sú obmedzené. Tým pádom dochádza k lepšiemu potlačeniu šumu, ako v prípade tvrdého prahovania, viď obrázok 6.3, kde sú zobrazené HL koeficienty dekomponovaného obrazu. Vývojový diagram popisujúci prahovanie je zobrazený v prílohe A.5.



Obr. 6.3: Ukážka dekompozície 1. stupňa obrazu.

6.3 Implementácia NLM filtra

Implementácia tejto metódy vychádza z teoretického popisu a definičných vzťahov popísaných v kapitole 5.1. Kvôli výpočtovej náročnosti však nie je vhodné pre odhad hodnoty každého pixelu počítať vážený priemer všetkých pixelov obrazu. Z tohto dôvodu sa vymedzuje tzv. vyhľadávacia oblasť, o rozmeroch $2r + 1 \times 2r + 1$ pixelov.

Potom pre farebný obraz u ($u = (u_1, u_2, u_3)$, teda R,G,B zložky obrazu) a konkrétny pixel p možno NLM filter popísať vzťahom:

$$\hat{u}_i(p) = \sum_{q \in B(p,r)} u_i(q)w(p,q), \quad (6.1)$$

kde $B(p,r)$ predstavuje okolie so stredom v pixelu p a veľkosťou $2r + 1 \times 2r + 1$ pixelov. Veľkosť tejto vyhľadávacej oblasti, ako aj veľkosť porovnávacieho okna sa mení podľa smerodajnej odchýlky šumu. Pri vyššej hodnote smerodajnej odchýlky je potrebná väčšia veľkosť okna, aby bolo okno robustnejšie. Takisto treba zväčšiť aj vyhľadávaciu oblasť, aby sa zvýšila schopnosť algoritmu nájsť viacero podobných pixelov. Veľkosť filtračného parametra h je $h = k \cdot \sigma$. Hodnota k sa pri zväčšovaní okna znižuje. Pri väčších rozmeroch sa vzdialenosť dvoch blokov sústredí bližšie k $2\sigma^2$ a preto môže byť k filtrácii použitá nižšia hodnota k . Prehľad týchto parametrov je v tab. 6.1 pre šedotónové obrazy a v tab. 6.2 pre farebné obrazy.

Tab. 6.1: Parametre NLM filtra pre šedotónový obraz.

σ	Okno podobnosti	Vyhľadávacia oblasť	h
$0 < \sigma \leq 15$	3×3	21×21	0.40σ
$15 < \sigma \leq 30$	5×5	21×21	0.40σ
$30 < \sigma \leq 45$	7×7	35×35	0.35σ
$45 < \sigma \leq 75$	9×9	35×35	0.35σ
$75 < \sigma \leq 100$	11×11	35×35	0.30σ

Tab. 6.2: Parametre NLM filtra pre farebný obraz.

σ	Okno podobnosti	Vyhľadávacia oblasť	h
$0 < \sigma \leq 25$	3×3	21×21	0.55σ
$25 < \sigma \leq 55$	5×5	35×35	0.40σ
$55 < \sigma \leq 100$	7×7	35×35	0.35σ

V porovnaní s ostatnými metódami implementovanými v tejto práci, je metóda NLM filtra implementačne a zároveň aj výpočetne najnáročnejšia. Vývojový diagram popisujúci činnosť algoritmu je na obrázku 6.4a. Jediným vstupným parametrom vytvoreného operátora je smerodajná odchýlka σ , od ktorej sú odvodené podľa vyššie uvedených tabuliek ostatné potrebné parametre, ako napríklad veľkosť porovnávacieho okna, veľkosť vyhľadávacej oblasti apod. Okrem inicializácie parametrov sú v predspracovaní obrazu vstupné obrazové dáta uložené do príslušných

polí, s ktorými sa ďalej pracuje. Ďalej bola vytvorená LUT tabuľka s hodnotami funkcie e^{-x} , ktorá bude potrebná pre výpočet váh pixelov.

Pri prechádzaní po jednotlivých pixeloch p obrazu pri každom kroku treba kontrolovať aktuálnu pozíciu pixela vzhľadom na okraje obrazu a v prípade potreby obmedziť veľkosť vyhľadávacej oblasti, resp. okna podobnosti. Taktiež je potrebné vždy na začiatku cyklu inicializovať (vynulovať) váhy a pomocné pole `fDenoised`. Toto pole predstavuje štvorec s veľkosťou okna podobnosti.

Pri prechádzaní vyhľadávaciou oblasťou sa najskôr vypočíta Euklidovská vzdialenosť pixelov p a q , z ktorej sa následne pomocou vyššie spomínanej LUT tabuľky určí váha pixela q . LUT tabuľka bola vytvorená z dôvodu urýchlenia výpočtu hodnoty exponenciálnej funkcie. Zistená váha sa pripočíta k celkovej váhe a porovná sa s maximálnou váhou. V prípade, že je aktuálna váha väčšia, prepíše hodnotu maximálnej váhy. Na všetky prvky v okolí pixelu q je aplikovaná táto váha, a hodnoty pixelov sa pripočítajú k hodnotám v pomocnom poli `fDenoised`. Tento krok sa opakuje až do prejdania celej vyhľadávacej oblasti, viď obrázok 6.4b.

Na pixely v okolí bodu p je aplikovaná maximálna váha, a tieto pixely sú taktiež pripočítané do poľa `fDenoised`. Hodnota celkovej váhy sa navýši o hodnotu maximálnej váhy (viď príloha A.1a).

Ak súčet váh nie je veľmi blízky nule, treba hodnoty pixelov okolia bodu p normalovať. Aktuálny obsah poľa `fDenoised` sa podelí hodnotou celkovej váhy a výsledok sa pripočíta k výstupnému obrazu do okna so stredom v bode p . Zároveň sa inkrementuje počet podobných pixelov (viď príloha A.1b).

Týmto bol popísaný cyklus pre stanovenie hodnoty jediného pixela. Po skončení celého cyklu je potrebné ešte raz prejsť všetky body, skontrolovať či neboli nájdené nejaké podobné pixely. Ak áno, výstup sa vydolí počtom týchto podobných pixelov. V opačnom prípade sa na výstup zapíše hodnota rovnakého (pôvodného) pixelu (viď príloha A.1c).

6.3.1 Paralelizácia kódu

Nakoľko vyššie popísaný algoritmus je výpočtovo veľmi náročný, bolo v tomto prípade potrebné zaviesť paralelizáciu behu programu. K tomuto účelu bola využitá štruktúra `Fork/Join` [16], implementácia rozhorenia `ExecutorService`, ktorá umožňuje využiť výhody viacerých procesorov. Je určená na prácu, ktorú možno rekurzívne rozdeliť na menšie časti. Cieľom je využiť celý dostupný výpočtový výkon a zvýšiť tak výkonnosť aplikácie. Podobne ako u ostatných inštancií triedy `ExecutorService`, `Fork/Join` rozdeľuje úlohy do pracovných vlákien v súbore vlákien (*thread pool*). `Fork/Join` sa líši tým, že využíva tzv. *work-stealing* algoritmus. Pracovné vlákna, ktoré sú voľné, kradnú prácu od iných vlákien, ktoré sú zaneprázdnené. Základom

štruktúry Fork/Join je trieda `ForkJoinPool`, ktorá je rozšírením `AbstractExecutorService`. `ForkJoinPool` implementuje jadro algoritmu work-stealing a umožňuje spúšťať `ForkJoinTasks`.

Použitie štruktúry Fork/Join je jednoduché. Prvým krokom je napísať kód ktorý vykonáva segment práce. Tento kód môže vyzeráť napríklad takto:

ak (práca je dostatočne malá)

vykonaj prácu sekvenčne

inak

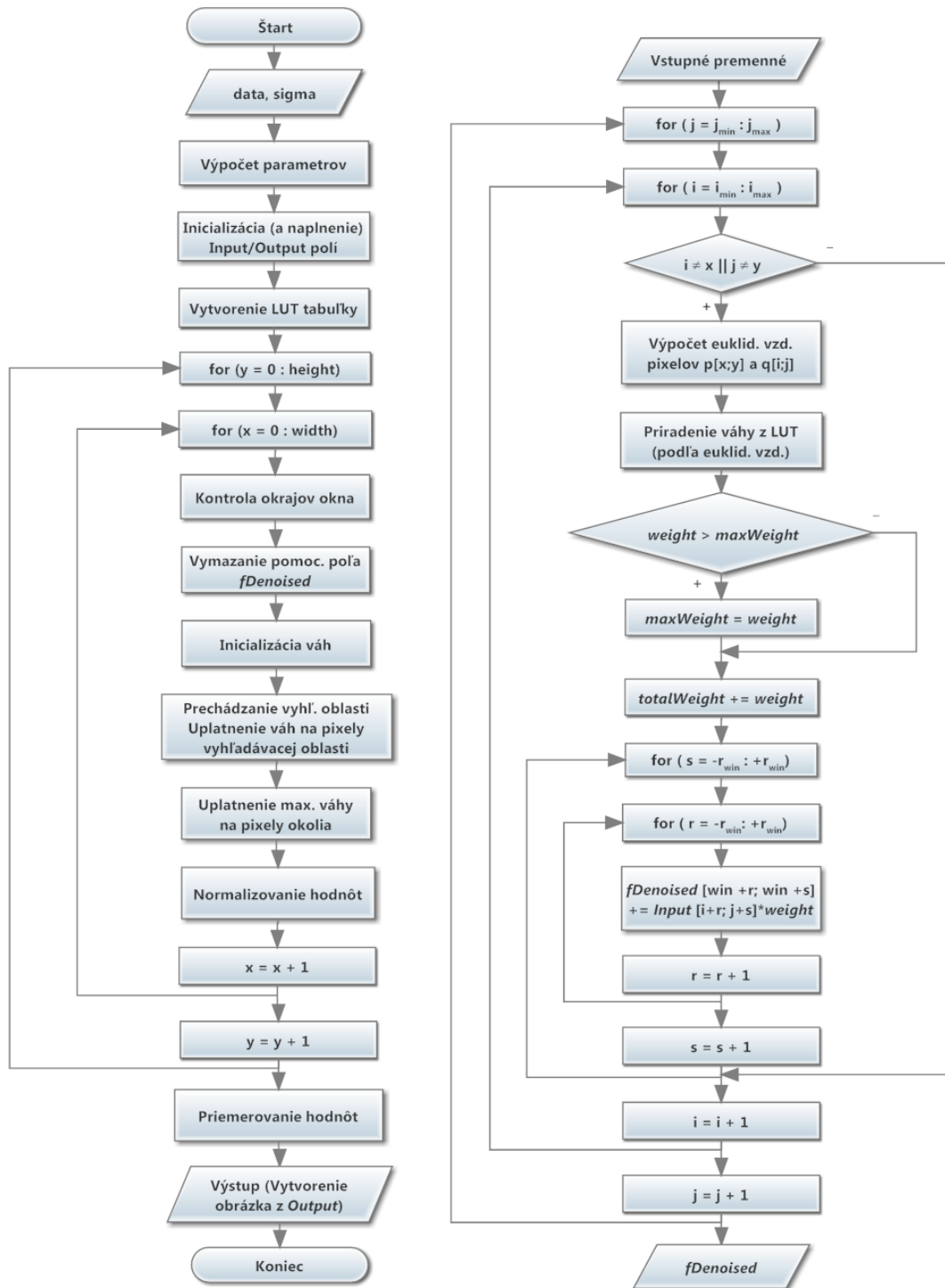
rozdeľ prácu na dve časti

zavolaj obe časti a počkaj na výsledky

Kód musí byť podtriedou triedy `ForkJoinTask`, zvyčajne jedným zo špecializovaných typov `RecursiveTask` alebo `RecursiveAction`. Ak je `ForkJoinTask` pripravený, treba vytvoriť reprezentanta celej vykonávanej práce a predať ho metóde `invoke()`, ktorá je inštanciou `ForkJoinPool`.

Pri implementácii NLM filtra bola vytvorená trieda `NLMeansFilter` ako podtrieda `RecursiveAction`. Podľa veľkosti rozmerov vstupného obrazu sa rozhoduje, či bude úloha spracovaná sekvenčne, alebo naopak rozdelená na viacero častí. Toto rozdelenie sa týka hlavného cyklu pri prechádzaní obrazu po pixeloch, teda obraz je horizontálne rozdelený na n (n = počet dostupných procesorov) častí, ktoré sú spracovávané zvlášť. Ako reprezentant celej úlohy bol vytvorený objekt `nlm` triedy `NLMeansFilter`, ktorý sa predáva ako vstupný argument metóde `invoke()`.

Pri spracovaní však vznikol problém pri prechodoch medzi jednotlivými úsekmi obrazu, ktoré boli spracovávané zvlášť. Tieto prechody neboli úplne hladké, čo bolo spôsobené tým, že pri hľadaní podobných pixelov v blízkosti spodného okraja jedného segmentu obrazu neboli k dispozícii pixely nasledujúceho segmentu. Toto sa podarilo vyriešiť čiastočným prekryvom segmentov.



(a) Zjednodušený diagram algoritmu

(b) Prechádzanie vyhľadávacej oblasti

Obr. 6.4: Vývojový diagram algoritmu NLM filtra.

7 POROVNANIE DOSIAHNUTÝCH VÝSLEDKOV

V rámci tejto diplomovej práce boli vytvorené tri nové operátory na odstraňovanie šumu z digitálnych obrazov. Tieto operátory boli porovnané s niektorými už existujúcimi filtermi v rozšírení ImageMiner. Konkrétne sa jednalo o mediánový filter a Gaussov filter. V tejto kapitole budú teda vyhodnotené výsledky filtrácie a bude porovnaná účinnosť jednotlivých filtrov.

7.1 Operátor na porovnávanie kvality

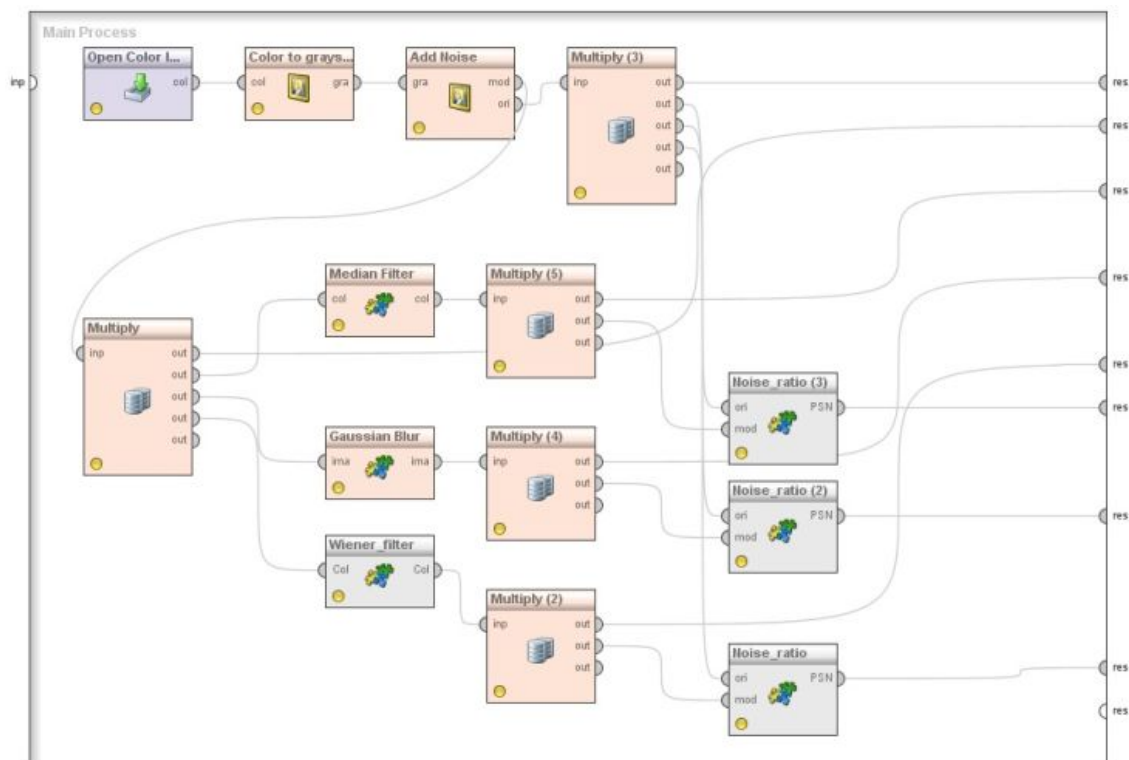
Na porovnávanie kvality obrazu bol implementovaný operátor `NoiseRatio`, porovnávajúci obrazy na základe vybraných kritérií popísaných v kapitole 1.6. Na vstupy operátora sú privedené jasové zložky pôvodného obrazu a obrazu po filtrácii. Jasová zložka bola zvolená preto, lebo je v nej sústredená väčšina energie digitálneho obrazu. Výstupom operátora je tabuľka typu `ExampleSet` s vypočítanými hodnotami jednotlivých kritérií pomeru šumu v obraze. Na obrázku 7.1 je ukážka procesu na porovnanie filtrov v prostredí RapidMiner. Na tomto obrázku operátor `Add Noise` pridáva do vstupného šedotónového obrazu Gaussov šum s rozptylom 0,1 (10%). Tento zašumený obraz je potom privedený na vstup jednotlivých filtrov. Po filtrácii je výsledný obraz privedený na výstup a zobrazený. Výsledok filtrácie je vidieť na obrázku 7.3.

7.2 Štatistické porovnanie kvality

Pre porovnanie boli zvolené kritéria *SAE*, *MAE*, *MSE* a *PSNR* (viď kap. 1.6). Ako referenčný obrázok bol použitý obrázok `lena.jpg`, ktorý sa bežne používa v oblasti spracovania obrazu. Do tohto obrazu bol vnesený Gaussov šum so smerodajnou odchýlkou $\sigma = 25$. U jednotlivých operátorov boli jednotlivé parametre nastavené nasledovne:

- NLM filter: Sigma = 25,
- Tvrdé waveletové prahovanie: 2. stupeň dekompozície; threshold = 48,
- Mäkké waveletové prahovanie: 2. stupeň dekompozície; threshold = 20,
- Empirický wienerov filter: block size 8×8 ; threshold = 45; atten. = 0,2245,
- Mediánový filter: počet iterácií = 1,
- Gaussov filter: počet iterácií = 1; strength = 2.

Dosiahnuté výsledky sú zobrazené v tabuľke 7.1. Pre kritéria *SAE*, *MAE* a *MSE* obecné platí, že čím nižšiu majú hodnotu, tým je obrázok kvalitnejší. U totožných



Obr. 7.1: Proces porovnávania metód v RapidMineri.

obrázkov sú teda nulové. U *PSNR* naopak u kvalitných obrázkov očakávame hodnotu čo najvyššiu. Pri obrazových a video kompresiách sa táto hodnota typicky pohybuje v rozmedzí 30 – 50 dB.

Tab. 7.1: Zistené hodnoty kvalitatívnych kritérií.

Operátor	SAE ($\times 10^6$)	MAE (–)	MSE (–)	PSNR (dB)
Obráz so šumom	3,347	12,768	258,331	24,009
Tvrde prahovanie	2,517	9,602	158,654	26,126
Mediánový filter	1,645	6,275	110,943	27,680
Mäkké prahovanie	1,841	7,023	84,645	28,855
Gaussov filter	1,565	5,970	77,840	29,219
DCT Wienerov filter	1,827	6,971	77,286	29,250
NLM filter	1,502	5,728	61,639	30,232

V tabuľke sú jednotlivé filtre vzostupne zoradené podľa účinnosti filtrácie na základe *PSNR*. Najlepšie výsledky boli v tomto prípade dosiahnuté pri aplikovaní NLM filtra. Výsledky sa môžu jemne líšiť pri rôznych nastaveniach parametrov jednotlivých filtrov. Treba však podotknúť, že pri zvýšení počtu iterácií u Gaussovho

alebo mediánového filtru dôjde k väčšiemu rozmazaniu hrán, čím sa kvalitatívne kritéria zhoršia. Podobne pri nastavení vyššieho prahu pri frekvenčných filtroch dochádza k vzniku výraznejších blokových artefaktov. Dané výsledky boli získané pri optimálnom nastavení parametrov jednotlivých operátorov.

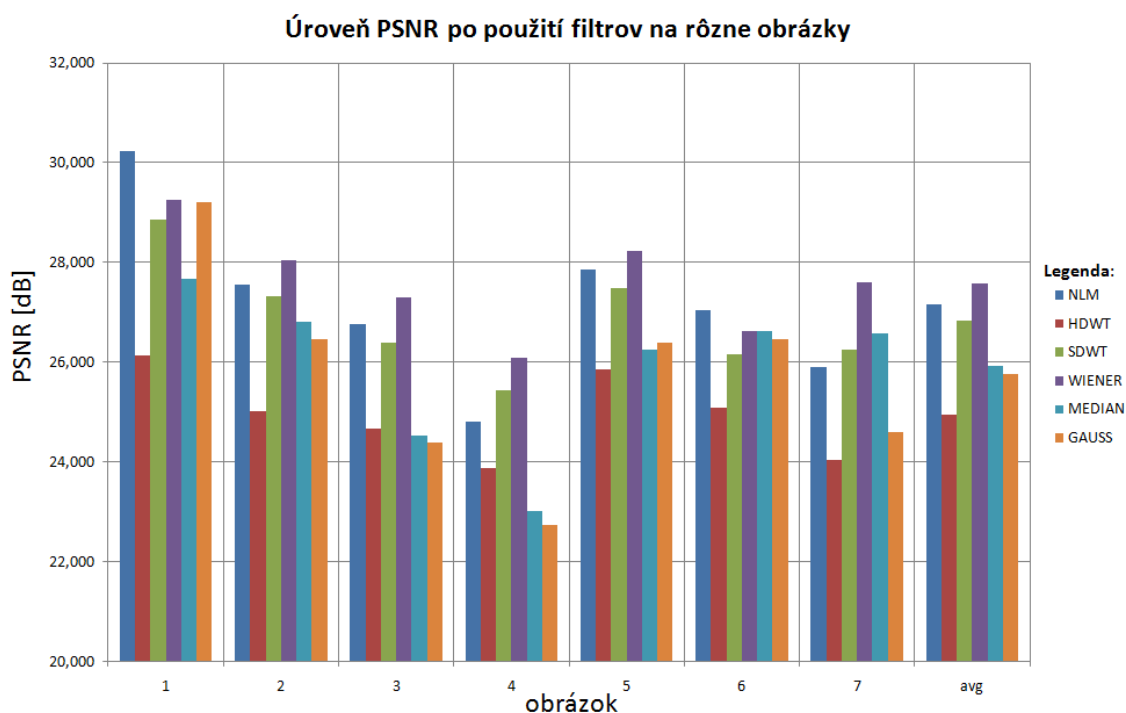
Objektívne porovnanie účinnosti použitých filtrov však nie je možné realizovať len na základe ich aplikovania na jeden konkrétny obrázok. Účinnosť odstránenia šumu do značnej miery závisí aj od štruktúry vstupných dát, teda od charakteru obrazu. Z tohto dôvodu boli jednotlivé filtry testované na viacerých zdrojových obrázkoch s rôznym obsahom. V tabuľke 7.2 sú uvedené hodnoty *PSNR* zistené pri tejto analýze. Zdrojové obrázky, ktoré boli použité, sú zobrazené v prílohe A.6. Pre lepšiu názornosť boli tieto hodnoty vynesené do stĺpcového grafu, viď obrázok 7.2. Zo zistených dát možno vyvodiť niekoľko záverov:

- NLM filter si dokáže najlepšie poradiť s obrazom obsahujúcim veľké plochy s pomerne malými zmenami jasu.
- Najväčšiu stálosť výsledkov má empirický Wienerov filter, ktorý spomedzi všetkých dosahuje vo väčšine prípadov štatisticky najlepšie výsledky.
- U členitých obrazov, alebo obrazov s častou zmenou úrovne jasu sa znižuje schopnosť filtrov potlačiť šum (viď výsledky pre obrázky č. 4).

Nakoniec ešte stojí za zmienku fakt, že účinnosť odstraňovania šumu podstatne závisí aj od vlastností samotného šumu. V rámci tejto práce boli navrhované filtre testované prakticky len na jeden druh, a to síce Gaussov aditívny šum, ktorého presné parametre boli známe. V praxi sa však väčšinou stretneme so šumom s neznámymi parametrami, a teda účinnosť odstránenia šumu do značnej miery závisí od schopnosti odhadnúť charakter šumu prítomného v obraze.

Tab. 7.2: Hodnoty *PSNR* (v dB) rôznych obrázkov po filtrácii.

Obrázok	NLM	HDWT	SDWT	Wiener	Median	Gaussian
1	30,232	26,126	28,855	29,250	27,680	29,219
2	27,545	25,023	27,318	28,043	26,816	26,473
3	26,764	24,676	26,387	27,296	24,526	24,398
4	24,799	23,888	25,428	26,091	23,015	22,727
5	27,856	25,846	27,490	28,234	26,247	26,395
6	27,051	25,078	26,165	26,619	26,622	26,457
7	25,898	24,043	26,248	27,592	26,587	24,607
Priemer	27,164	24,954	26,841	27,589	25,928	25,754



Obr. 7.2: Hodnoty PSNR u rôznych obrázkov.

7.3 Subjektívne porovnanie kvality

V praxi sa často stáva, že obraz s lepšími kvalitatívnymi kritériami (napr. vyššou hodnotou PSNR) sa nám nie vždy javí ako kvalitnejší. Subjektívne hodnotenie kvality obrazu sa zakladá na vizuálnom posudku hodnotiacej osoby.

Na obrázku 7.3 možno vidieť výsledky filtrácie pomocou popisovaných metód. Pre porovnanie, hodnoty *PSNR* pre tento konkrétny obrázok sú uvedené v tabuľke 7.2 v 2. riadku (obrázok č. 2). Pri subjektívnom hodnotení kvality odstránenia šumu možno konštatovať nasledovné fakty:

- Z obrázku 7.3 (c) je zrejmé, že Gaussov filter šum z obrazu neodstráni celkom, iba ho rozmáže a s ním aj hrany a detaily obrazu.
- Mediánový filter možno s výhodou použiť na odstránenie impulzného šumu (ako je napr. salt & pepper), kedy sa v obraze nachádzajú izolované pixely, no v prípade Gaussovho šumu (viď obrázok 7.3 (b)) dosahuje nie veľmi presvedčivé výsledky (viď obrázok 7.3 (d)). Podobne ako u Gaussovho filtru, aj tu dochádza k rozmazaniu detailov a textúr, avšak v menšej miere.
- Wienerov filter naopak dobre uchováva hrany a textúry, avšak do obrazu sú vplyvom DCT vnášané šachovnicovité artefakty, ktoré zhoršujú výslednú kvalitu obrazu (viď obrázok 7.3e).
- Pri tvrdom prahovaní (viď obr. 7.3f) sú bádateľné výrazné nespojitosti spô-

sobené Gibsovým javom. Tieto sú čiastočne potlačené metódou mäkkého prahovania (viď obr. 7.3g), ale v obraze aj tak ostáva pomerne značný šum, zvlášť viditeľný na rovných plochách.

- Pri NLM filtrácii (obrázok 7.3h) oproti tomu v obraze na rovných plochách neostávajú takmer žiadne stopy šumu, hrany sú v obraze pomerne dobre zachované, avšak metóda potláča jemnejšie detaily obrazu.



(a) Pôvodný obraz



(b) Obraz so šumom



(c) Gaussov filter



(d) Mediánový filter



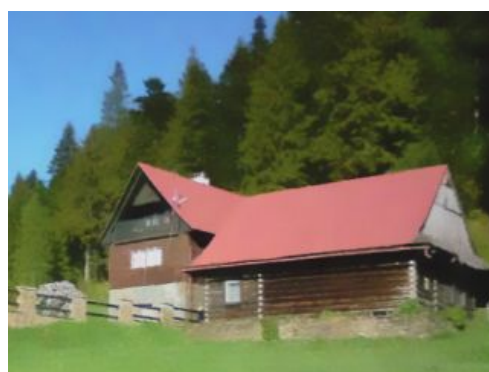
(e) Empirický Wienerov filter



(f) Tvrdé prahovanie



(g) Mäkké prahovanie



(h) NLM filter

Obr. 7.3: Filtrácia obrazu v RapidMineri

8 ZÁVER

V rámci diplomovej práce bola spracovaná problematika odstraňovania šumu z digitálnych obrazov. Práca je rozdelená na teoretickú a praktickú časť.

V teoretickej časti boli najskôr vysvetlené niektoré základné náležitosti spracovania obrazu, šumu v obraze a obrazových transformácii. Následne boli popísané jednotlivé metódy odšumovania obrazu, ktoré boli logicky členené podľa princípu ich činnosti do troch kategórií: lokálne metódy, nelokálne metódy a metódy vo frekvenčnej oblasti. Ku každej kategórii boli uvedené príklady konkrétnych filtrov, ktoré boli podrobne popísané matematicky i principiálne. Každá z uvedených metód má svoje výhody aj nevýhody. Gaussov filter dosahuje dobré výsledky pri spojitých, hladkých častiach obrazu, naopak pri jednotvárných častiach, ako sú hrany či textúry dochádza k ich nežiadúcemu rozmazaniu. Anizotropické filtre dokážu v porovnaní s Gaussovým filtrom lepšie uchovávať hrany, no pri rovných oblastiach majú horšie výsledky. Yaroslavského filter okolia, ktorý predstavuje akýsi medzičlánok medzi lokálnymi a nelokálnymi metódami, lepšie uchováva hrany, avšak v obraze vznikajú nespojitosti ktoré spôsobujú vznik šok efektu. Pri Wienerovom ideálnom filtri vznikajú v obraze vplyvom fourierovej transformácie parazitné periodické vzory. Podobne aj u empirického wienerovho filtru, s typickým šachovnicovým efektom vznikajúcim vplyvom diskkrétnej kosínusovej transformácie (DCT). Pri waveletovom prahovaní vznikajú z dôvodu odstránenia nižších koeficientov drobné oscilácie, ktoré sú potom bádateľné vo výslednom obraze. Nelokálny algoritmus (Non local-means) dosahuje v mnohých ohľadoch lepšie výsledky, no zároveň sa vyznačuje omnoho väčšou výpočtovou náročnosťou, keďže základnou myšlienkou algoritmu je, že vyhľadáva okolie v celom obraze.

V praktickej časti práce boli na základe poznatkov z teoretickej časti implementované niektoré z popisovaných metód. Konkrétne šlo o metódy filtrácie obrazu vo frekvenčnej oblasti pomocou empirického Wienerovho filtru a waveletového prahovania a filtráciu NLM filtrom. Všetky spomínané metódy boli najskôr odladené ako konzolové aplikácie v programovacom jazyku Java, vo vývojovom prostredí Eclipse. Po overení správnej funkčnosti implementovaných filtrov boli tieto aplikácie transformované do podoby operátorov používaných platformou RapidMiner. V tomto prostredí bola taktiež otestovaná ich funkčnosť a boli porovnané výstupy jednotlivých metód.

Výpočtovo najnáročnejšou metódou, ako som už vyššie spomínal, bola metóda NLM filtrácie. Ako vyplýva z popisu algoritmu uvedeného v kapitole 6.3 venovanej implementácii tejto metódy, najväčší problém spôsobuje vyhľadávanie a váhovanie podobných pixelov (resp. okien). Vytvorené aplikácie boli testované na počítači s inštalovaným procesorom Intel[®] Core[™] 2 Duo 2,00 GHz. Paralelizáciou

výpočtu, kedy bolo spracovanie realizované na oboch jadrách súčasne sa podarilo dobu výpočtu skrátiť na polovicu, avšak stále bola veľmi dlhá (pri obrázku s rozlíšením 512×512 približne 32 sekúnd). Jedným možným riešením, ako skrátiť dobu výpočtu, by bolo použiť počítač s výkonnejším multiprocesorom, alebo clusterový počítač. Druhou možnosťou by bolo prenechať samotný výpočet grafickej karte optimalizovanej na spracovanie obrazu.

Keďže cieľom práce bolo porovnať výsledky odstránenia šumu implementovaných metód so známymi technikami, bol navyše vytvorený operátor pre aplikáciu RapidMiner realizujúci výpočet kvalitatívnych kritérií obrazu. Tento operátor na základe vstupných obrazových dát vypočíta podobnosť daných obrazov na základe rôznych kritérií, napr. MSE, PSNR, a podobne. Do porovnávania boli zahrnuté výstupy operátorov implementovaných v rámci tejto práce a výstupy už existujúcich operátorov (Gaussov filter a mediánový filter). Štatisticky najlepšie výsledky dosahoval vytvorený empirický Wienerov filter. Po vizuálnom (subjektívnom) zhodnotení výsledkov filtrácie je zrejmé, že tento filter v obraze zanecháva viditeľné šachovnicové artefakty, ktoré potom degradujú vizuálny vnem. Z tohto hľadiska sa ako najlepšia javí metóda NLM filtrácie, pri ktorej na rovných plochách nie je bádateľný takmer žiaden šum. Nižšia hodnota PSNR je spôsobená tým, že filter rozmazáva jemné detaily obrazu.

ZOZNAM POUŽITEJ LITERATURY

- [1] DOBEŠ, M., *Zpracování obrazu a algoritmy v C#*. BEN – technická literatura, Praha 2008. ISBN 978-80-7300-233-6.
- [2] ŠMIRG, O., *Detekce lidské postavy v obrazové scéně*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 57 s. Vedoucí diplomové práce Ing. Michal Kohoutek.
- [3] ŘÍHA, K., *Pokročilé techniky zpracování obrazu*. Elektronické texty VUT. Ústav telekomunikací FEKT VUT v Brně, 2007.
- [4] RAJMIC, P., *Moderní počítačová grafika*. Elektronické texty VUT. Ústav telekomunikací FEKT VUT v Brně, 2011.
- [5] Image Color Conversion. [online]. Intel, 2010 [cit. 2012-05-15]. Dostupné z WWW: <http://software.intel.com/sites/products/documentation/hpc/ipp/ippi/ippi_ch6/ch6_Intro.html>
- [6] BUADES, A.; COLL, C.; MOREL, J.-M., *A review of image denoising algorithms, with a new one*, SIAM J. Multiscale Model. Simul., 2005, vol. 4, s. 490. Dostupný z WWW: http://www.math.upenn.edu/trietle/teaching/fall11/reports/NLM_morel.pdf
- [7] JAN, J., *Medical image processing, reconstruction and restoration: concepts and methods*. New York (USA): CRC Press - Taylor & Francis Group, 2006. 730 s. ISBN 0-8247-5849-8.
- [8] VALOUCH, L., *Implementace vlnkové transformace v jazyku C++*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 58 s. Vedoucí diplomové práce Ing. Radek Beneš.
- [9] SMÉKAL, Z., *Číslíkové zpracování signálu*. Elektronické texty VUT. Brno: Ústav telekomunikací FEKT VUT v Brně, 2009. 122 s.
- [10] ČÍKA, P., *Digitální vodoznačení obrazu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 102 s. Vedoucí disertační práce doc. Ing. Karel Němec, CSc.
- [11] ŠMÍD, R., *Úvod do vlnkové transformace* [online]. CTU in Prague, Faculty of Electrical Engineering, Department of Measurements, [cit. 2011-11-23]. Dostupný z WWW: <<http://measure.feld.cvut.cz/usr/staff/smid/wavelet-intro-html.html>>

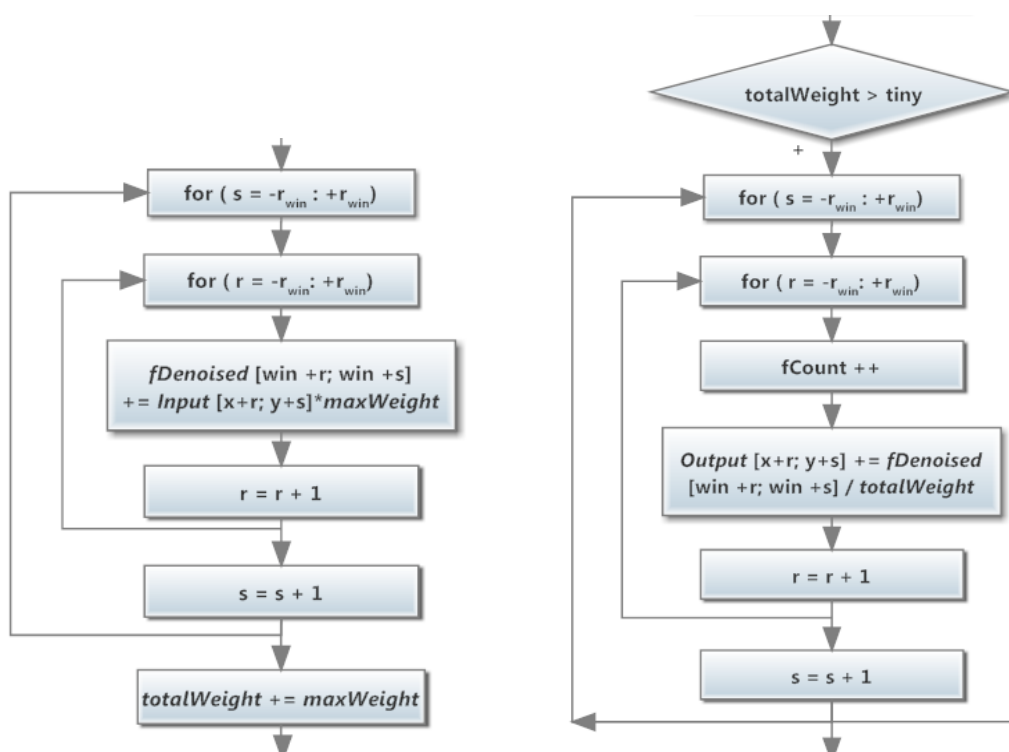
- [12] PERONA, P.; MALIK, J., *Scale space and edge detection using anisotropic diffusion*, IEEE Trans. Patt. Anal. Mach. Intell., 12 (1990), pp. 629–639.
- [13] BURGER, W.; BURGE, M. J., *Principles of Digital Image Processing: Fundamental Techniques*. Londýn: Springer, 2009. 272 s. ISBN 978-1848001909.
- [14] GONZALEZ, R. C.; WOODS, R. E., *Digital Image Processing. 3.* Londýn: Pearson Pentice Hall, 2008. 954 s. ISBN 978-0-13-505267-9.
- [15] ZAKHOUR, S., *Java 6: výukový kurz. Vyd. 1.* Brno: Computer Press, 2007, 534 s. ISBN 978-80-251-1575-6.
- [16] Fork/Join (The Java™ Tutorials>Essential Classes>Concurrency). [online]. [cit. 2012-05-21]. Dostupné z: <http://docs.oracle.com/javase/tutorial/essential/concurrency/forkjoin.html>

ZOZNAM PRÍLOH

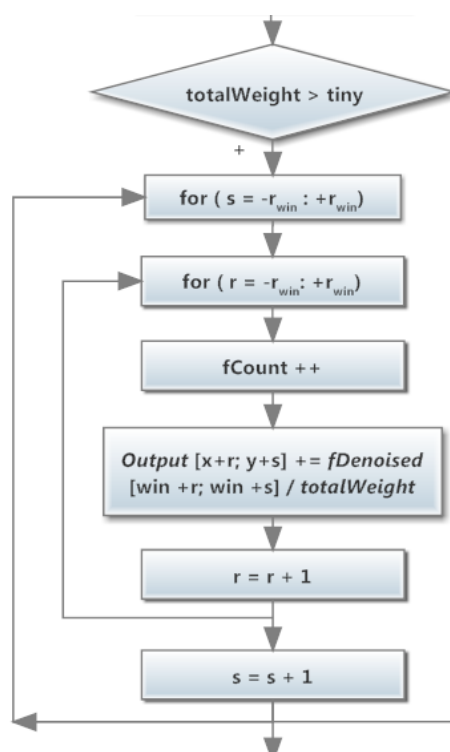
A Prvá príloha	58
A.1 NLM filter – detailnejší popis blokov vývojového diagramu	58
A.2 Empirický wienerov filter – prahovanie	59
A.3 Empirický wienerov filter – výpočet matice DCT	60
A.4 Waveletové prahovanie – dekompozícia	61
A.5 Waveletové prahovanie – vývojový diagram prahovania	62
A.6 Použité obrázky	63
B Druhá príloha	64
B.1 Obsah priloženého média	64

A PRVÁ PRÍLOHA

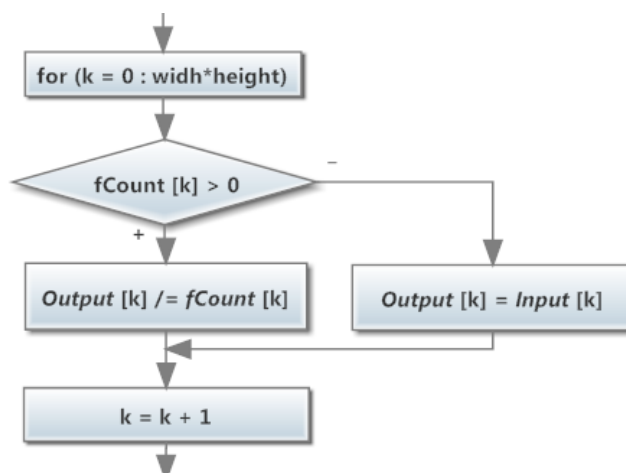
A.1 NLM filter – detailnejší popis blokov vývojového diagramu



(a) Pridelenie maximálnej váhy

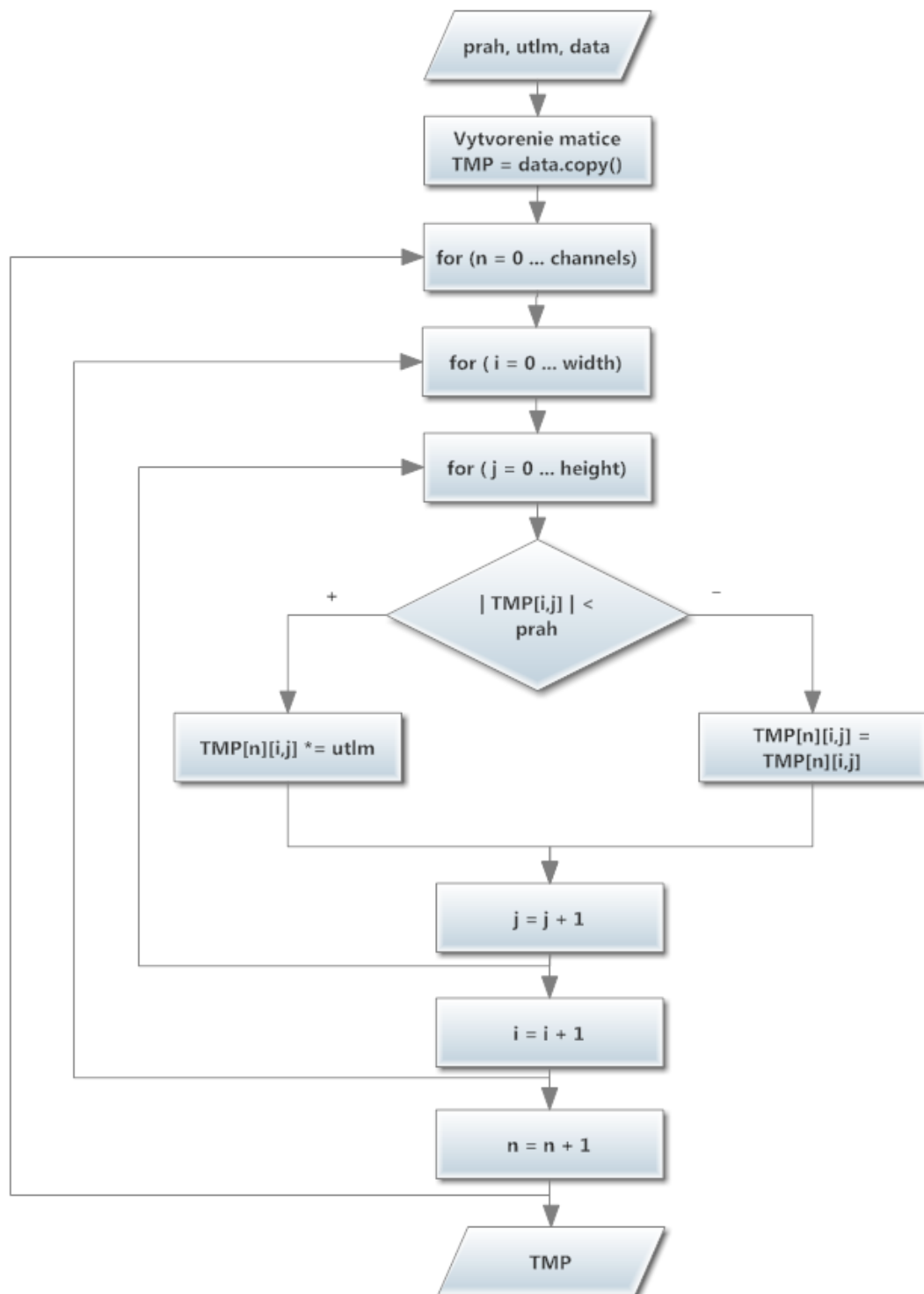


(b) Normalizácia hodnôt

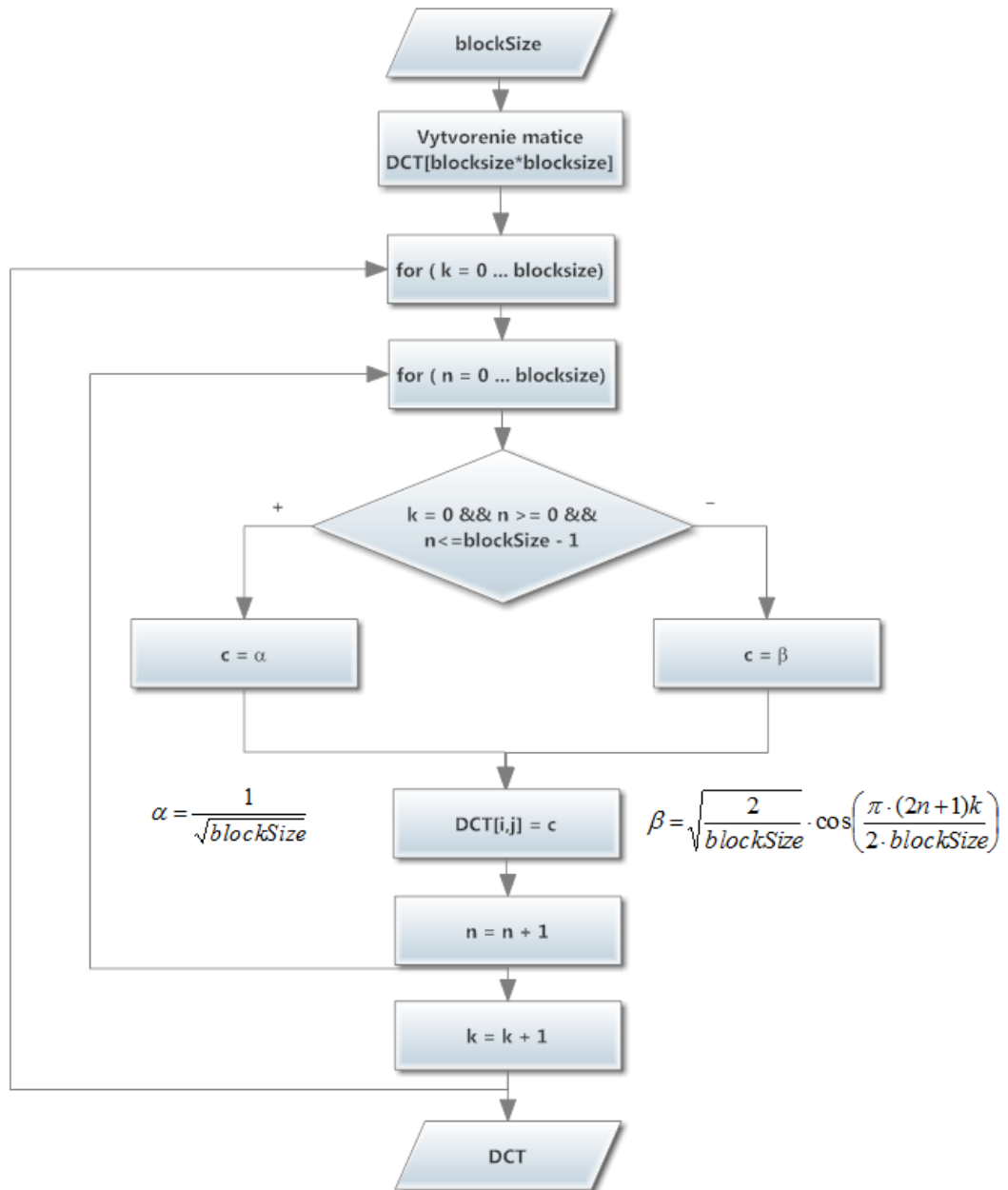


(c) Priemerovanie hodnôt

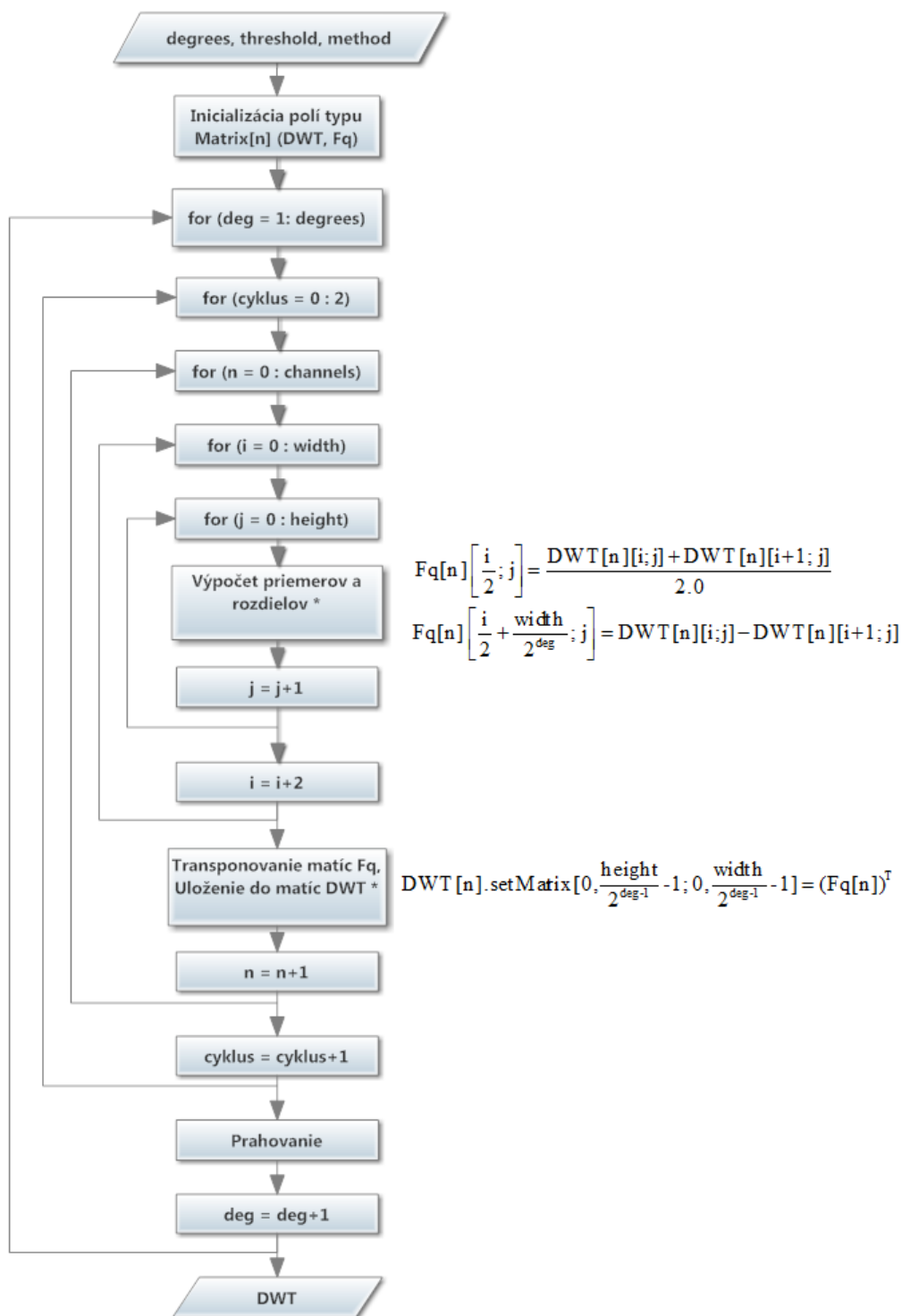
A.2 Empirický wienerov filter – prahovanie



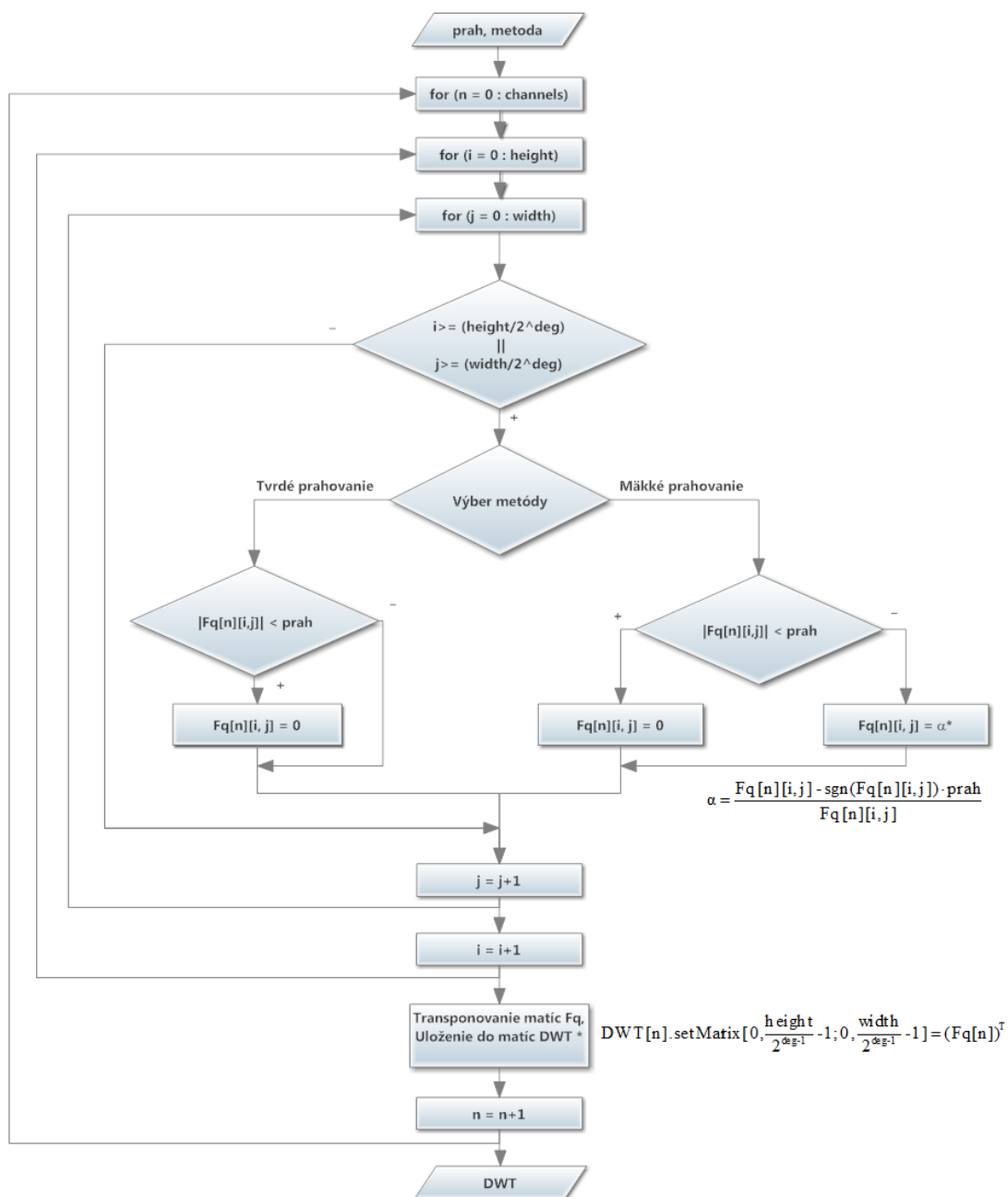
A.3 Empirický wienerov filter – výpočet matice DCT



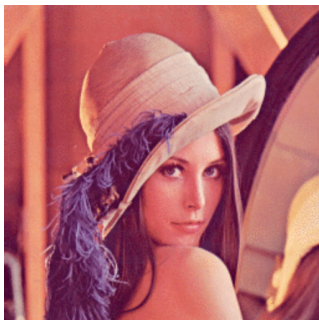
A.4 Waveletové prahovanie – dekompozícia



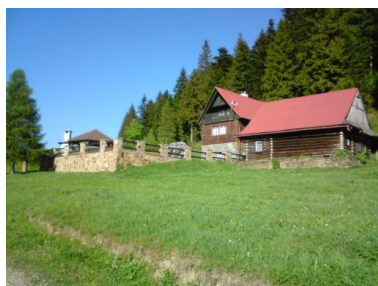
A.5 Waveletové prahovanie – vývojový diagram prahovania



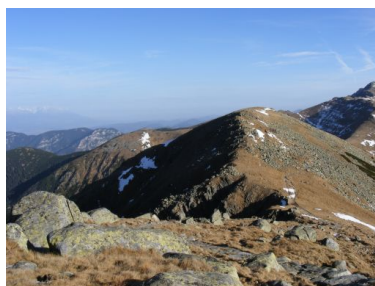
A.6 Použité obrázky



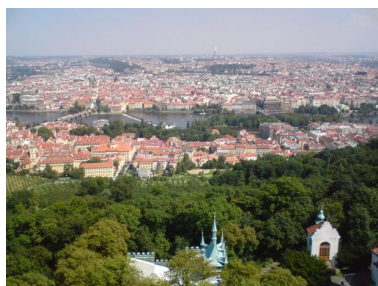
(a) obrázok č. 1



(b) obrázok č. 2



(c) obrázok č. 3



(d) obrázok č. 4



(e) obrázok č. 5



(f) obrázok č. 6



(g) obrázok č. 7

B DRUHÁ PRÍLOHA

B.1 Obsah priloženého média

- Elektronická verzia diplomovej práce
- Zdrojové súbory pre Latex
- Obrázky s ukážkami výsledkov
- Zdrojové kódy k projektom vytvorených v Eclipse