

TECHNICKÁ UNIVERZITA V LIBERCI



**FAKULTA MECHATRONIKY A MEZIOBOROVÝCH
INŽENÝRSKÝCH STUDÍÍ**

DIPLOMOVÁ PRÁCE

**System pro automatizované zpracování a
rozpoznávání obrazu**

2005

Aleš Jirásko

TECHNICKÁ UNIVERZITA V LIBERCI



**FAKULTA MECHATRONIKY
A MEZIOBOROVÝCH INŽENÝRSKÝCH STUDIÍ**

Studijní program: M2612 - Elektrotechnika a informatika
Studijní obor: 3902T005 - Automatické řízení a inženýrská informatika

**Systém pro automatizované
zpracování a rozpoznávání obrazu**

(System for automatic image processing and recognition)

Aleš Jirásko

Vedoucí diplomové práce: Ing. Josef Chaloupka
Konzultant: Prof. Doc. Ing. Jan Nouza, CSc.

Rozsah práce a příloh:

Počet stran: 73

Počet obrázků: 68

Počet vzorců: 56

Počet tabulek: 1

Datum: 17.05.2005

Anotace:

Hlavním cílem této diplomové práce bylo navržení a realizace systému pro automatizované zpracování a rozpoznávání obrazu.

V první části práce je uvedena teorie metod pro předzpracování obrazu, geometrické transformace obrazu, filtrace šumu, nalezení hran v obraze, segmentaci, matematickou morfologii a rozpoznávání objektů.

V druhé části je popsán vytvořený systém, který obsahuje graficky přívětivé a jednoduše ovladatelné prostředí. V něm jsou naprogramovány jednotlivé algoritmy pro zpracování a rozpoznávání obrazu ve formě samostatných a nezávislých modulů. Jednotlivé moduly lze řadit za sebe a tím vytvářet rozpoznávací řetězec, který je možné uložit a aplikovat na jakýkoliv jiný obraz. Celý systém je navržen a vytvořen tak, aby byl otevřený, tj. aby se do něho v budoucnu daly přidávat nové moduly.

Annotation:

The main aim of this work is to design and realize a system for automatic image processing and recognition.

In the first part of this work the theory and methods of image pre-processing, image geometric transformation, noise filtration, edge detection in image, segmentation, mathematical morphology, and object recognition are dealt with.

The second part of this work describes the designed system that contains graphically pleasant and simply mastered background. In the system there are particular algorithms for image processing and recognition, programmed as independent modules. It is possible to line particular modules in a row and, that way, create a chain of recognition that can be saved and later applied to any image. The system is designed and created to be open. This means that other modules can be added if needed.

Abstract:

The work aims to design and realize a system of automatic image processing and recognition. In the system particular methods are implemented, being in the form of independent modules that are possible to be lined in a recognition chain.

Poděkování:

Na tomto místě bych chtěl poděkovat Ing. Josefu Chaloupkovi za odborné vedení, pomoc při zpracování diplomové práce, za cenné rady a poskytnuté informace.

Prohlášení:

Byl jsem seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 o autorském právu, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení, apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Prohlašuji, že jsem celou diplomovou práci vypracoval samostatně pod vedením vedoucího diplomové práce a uvedl jsem všechny použité podklady a literaturu.

V Liberci dne 17.05.2005

.....

Obsah

| | |
|--|----|
| 1. Úvod..... | 9 |
| 2. Obraz a metody jeho zpracování | 10 |
| 2.1. Obraz, vzorkování a kvantování obrazu..... | 10 |
| 2.2. Barevné modely | 11 |
| 2.2.1. RGB model | 11 |
| 2.2.2. Y model | 12 |
| 2.2.3. CMY(K) model..... | 12 |
| 2.2.4. YCbCr model..... | 13 |
| 2.2.5. YIQ model | 14 |
| 2.2.6. HSV model..... | 14 |
| 2.2.7. HSI model..... | 15 |
| 2.2.8. HSL model..... | 15 |
| 2.3. Jasové transformace | 16 |
| 2.3.1. Histogram..... | 16 |
| 2.3.2. Jasové korekce | 17 |
| 2.3.3. Transformace jasové stupnice..... | 17 |
| 2.3.3.1. Negativ..... | 18 |
| 2.3.3.2. Změna kontrastu | 18 |
| 2.3.3.3. Změna jasu | 19 |
| 2.3.3.4. Ekvalizace histogramu | 19 |
| 2.4. Geometrické transformace..... | 20 |
| 2.4.1. Transformace souřadnic bodů..... | 20 |
| 2.4.1.1. Změna měřítka..... | 21 |
| 2.4.1.2. Zkosení..... | 21 |
| 2.4.1.3. Rotace | 21 |
| 2.4.2. Aproximace jasové hodnoty | 21 |
| 2.4.2.1. Interpolace metodou nejbližšího souseda | 22 |
| 2.4.2.2. Lineární interpolace | 22 |
| 2.4.2.3. Bikubická interpolace | 23 |
| 2.5. Filtrace šumu a poruch | 24 |
| 2.5.1. Lineární metody vyhlazování | 24 |
| 2.5.1.1. Prostý průměr | 24 |
| 2.5.2. Nelineární metody vyhlazování | 25 |
| 2.5.2.1. Metoda rotující masky | 25 |
| 2.5.2.2. Metoda mediánu | 26 |
| 2.6. Detekce hran v obraze | 26 |
| 2.6.1. Operátory aproximující derivaci obrazové funkce..... | 27 |
| 2.6.1.1. Robertsův operátor | 27 |
| 2.6.1.2. Laplaceův operátor | 28 |
| 2.6.1.3. Operátor Prewittové..... | 28 |
| 2.6.1.4. Sobelův operátor..... | 29 |
| 2.6.1.5. Robinsonův operátor..... | 29 |
| 2.6.1.6. Kirschův operátor | 29 |
| 2.6.2. Detekce hran pomocí druhé derivace obrazové funkce..... | 29 |
| 2.6.2.1. LoG operátor | 30 |
| 2.7. Segmentace | 31 |
| 2.7.1. Prahování | 31 |
| 2.8. Matematická morfologie | 31 |

| | |
|--|----|
| 2.8.1. Binární matematická morfologie..... | 32 |
| 2.8.1.1. Dilatace | 32 |
| 2.8.1.2. Eroze..... | 33 |
| 2.8.1.3. Otevření..... | 33 |
| 2.8.1.4. Uzavření..... | 33 |
| 2.8.1.5. Tref či miň..... | 34 |
| 2.8.2. Šedotonová matematická morfologie | 34 |
| 2.8.2.1. Šedotónová dilatace | 35 |
| 2.8.2.2. Šedotónová eroze..... | 35 |
| 2.8.2.3. Šedotónové otevření a uzavření | 35 |
| 2.9. Rozpoznávání objektů..... | 36 |
| 2.9.1. Barvení..... | 36 |
| 2.9.2. Korelace | 37 |
| 3. Aplikace IPRTool | 39 |
| 3.1. Popis programu | 39 |
| 3.1.1. Hlavní okno aplikace | 39 |
| 3.1.1.1. Nový projekt (New project) | 41 |
| 3.1.1.2. Otevřít projekt ... (Open project ...) | 41 |
| 3.1.1.3. Uložit projekt jako ... (Save project as ...) | 42 |
| 3.1.1.4. Otevřít obrázek ... (Open picture ...)..... | 42 |
| 3.1.1.5. Uložit obrázek ... (Save picture ...)..... | 44 |
| 3.1.1.6. Konec (Exit)..... | 44 |
| 3.1.1.7. Zpět (Undo)..... | 44 |
| 3.1.1.8. Znovu (Redo) | 44 |
| 3.2. Struktura programu | 45 |
| 3.2.1. Tvorba vlastního rozpoznávacího řetězce | 45 |
| 3.2.2. Otevření vytvořeného projektu | 46 |
| 3.3. Vizualizace výsledků..... | 47 |
| 3.4. Implementované metody | 48 |
| 3.4.1. Záložka Transformace barev (Color transformations) | 50 |
| 3.4.2. Záložka Transformace hodnot jasu (Pixel brightness transformations) | 53 |
| 3.4.3. Záložka Geometrické transformace (Geometric transformations)..... | 56 |
| 3.4.4. Záložka Filtrace šumu (Image smoothing) | 58 |
| 3.4.5. Záložka Hledání hran (Edge detectors) | 59 |
| 3.4.6. Záložka Segmentace (Segmentation) | 61 |
| 3.4.7. Záložka Binární matematická morfologie (Binary mathematic morphology) | 62 |
| 3.4.8. Záložka Šedotónová matematická morfologie (Grayscale mathematic morphology) | 64 |
| 3.4.9. Záložka Rozpoznávání objektu (Pattern recognition) | 65 |
| 4. Závěr | 67 |
| 5. Praktické ukázky | 68 |
| 5.1. Detekce obličejů..... | 68 |
| 5.2. Velikost mincí..... | 69 |
| 5.3. Hledání mincí..... | 70 |

1. Úvod

Jednou z intenzivně se rozvíjejících vědních oblastí je v současné době zpracování a rozpoznávání obrazu. Jedná se o úlohu, pomocí níž jsou v obraze hledány a popisovány objekty, které jsou předmětem rozpoznávání. K dosažení požadovaného cíle musí být provedeno několik kroků. Za prvé je to pořízení obrazu, například pomocí kamery nebo fotoaparátu, a jeho následné uložení v digitální podobě v počítači. Pořízení obrazu hraje jednu z důležitých rolí ve zpracování a rozpoznávání obrazu, protože ze špatně či nekvalitně pořízeného obrazu je úloha rozpoznávání velmi obtížná, je-li nemožná. Ve druhé fázi je provedeno předzpracování obrazu, při kterém jsou odstraněny menší chyby vzniklé při snímání a je upraven jas. Takto upravený obraz vstupuje do třetí části, ve které je obvykle provedena jeho segmentace. Segmentací je dosaženo rozdělení obrazu na jednotlivé objekty. Dalším krokem je popis jednotlivých objektů. Poslední částí je porozumění obsahu obrazu, což je například rozdělení objektů do předem známých skupin.

Cílem diplomové práce je navrhnout a realizovat systém pro automatické zpracování a rozpoznávání obrazu. Takto navržený systém, jehož rozhraní by mělo být graficky přívětivé a jednoduše ovladatelné s vizualizací výsledků z průběhu zpracování a rozpoznávání, bude obsahovat naprogramované algoritmy pro zpracování a rozpoznávání obrazu, zejména metody pro předzpracování obrazu, geometrické transformace obrazu, filtrace šumu, nalezení hran v obraze, segmentaci a matematickou morfologii. Tyto metody zde budou umístěny ve formě samostatných a nezávislých modulů, které bude možné řadit za sebe a tím tak vytvářet rozpoznávací řetězec. Jestliže jednotlivé moduly budou vyžadovat nastavení parametrů pro ně typických, bude toto umožněno například pomocí dialogových oken. Takto vytvořený řetězec bude možné uložit a aplikovat později na jakýkoliv jiný obraz, přičemž jeho vykonání bude automatické. Navržený systém by měl umožňovat ukládání obrazu v průběhu jeho zpracování a rozpoznávání. Celý systém by měl být navržen a vytvořen tak, aby byl otevřený, tj. aby se do něho v budoucnu daly přidávat nové moduly.

Tento systém bude sloužit k rychlé analýze obrazu, podle které bude možné určit, pomocí kterých metod a algoritmů pro zpracování a rozpoznávání obrazu by bylo možné daný obraz vyhodnocovat. Tento systém bude možné dále využít při výuce, kde bude umožněno s jeho pomocí demonstrovat jednotlivé úlohy zpracování a rozpoznávání obrazu.

Výhodou tohoto systému by měla být jeho snadná přenositelnost a malá kapacitní a výpočetní náročnost.

2. Obraz a metody jeho zpracování

2.1. Obraz, vzorkování a kvantování obrazu

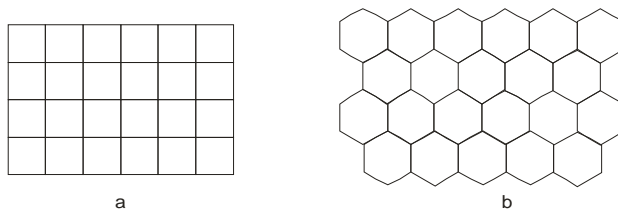
Obraz lze popsat jako spojitou funkci $f(x,y)$, která se nazývá obrazová funkce [1]. Tato funkce má dva argumenty x a y , jenž představují souřadnice v ploše. Častou funkční hodnotou této funkce je jas. V případě, že je funkční hodnota pouze jedna, je obraz monochromatický. Pokud je funkční hodnota obrazové funkce tvořena vektorem, pak je obraz multispektrální (barevný). Barva objektů je vlastnost odrážet elektromagnetické vlnění o různých vlnových délkách, přičemž pro lidské oko jsou důležité vlnové délky spektra v rozmezí 400nm (fialová barva) - 700 nm (červená barva), což je interval představující viditelnou část (část, kterou vnímá lidské oko) spektra (obr. 2.2).

V reálném světě je obraz reprezentovaný spojitým signálem. Pro reprezentaci obrazu v počítači je nutné signál převést na digitální. Tento převod má dvě části. Za prvé je to *vzorkování* signálu v matici $M \times N$ a za druhé *kvantování*.

V případě vzorkování [4] dvourozměrného signálu je intervalem vzorkování vzdálenost mezi dvěma nejbližšími body. O tom jaká by měla být vzdálenost mezi jednotlivými vzorkovanými body pojednává Shannonova věta o vzorkování, podle které musí být vzorkovací frekvence alespoň dvakrát větší než je nejvyšší frekvence ve vzorkovaném signálu, tj. vzorkovací interval musí být volen tak, aby byl roven nejméně polovině velikosti nejmenších objektů, které mají být v obraze zaznamenány. Snahou je zmenšit interval tak, aby byl nejmenší objekt reprezentován alespoň 5-ti a více body. Čím je menší vzorkovací interval, tím lépe jsou interpretovány detaily v digitálním obraze.

Kvantováním [3] je provedeno přiřazení určité diskrétní jasové hodnoty spojitě jasové funkci ve vzorkovaném bodě. Zde platí pravidlo, že čím je větší počet kvantovacích intervalů, tím přesněji jsou vyjádřeny detaily obrazu. Obvykle je použito 256 kvantovacích úrovní (~ 256 jasových úrovní), což lze vyjádřit 8-bity. Lze se setkat i se 4-bitovým (~16 jasových úrovní) či 1-bitovým (~ 2 jasové úrovně - bílá a černá barva) vyjádřením. Většinou se volí lineární kvantování, což znamená, že všechny úrovně jsou stejně velké.

Jednotlivé vzorkované body se nazývají pixely a jejich jasová hodnota je dána kvantovací úrovní. Pixely jsou zpravidla uspořádány do čtvercové mřížky (obr. 2.1a), lze se, mimo jiné, setkat i s mřížkou hexagonální [3] (obr. 2.1b).



Obr. 2.1 (a) čtvercová mřížka, (b) hexagonální mřížka

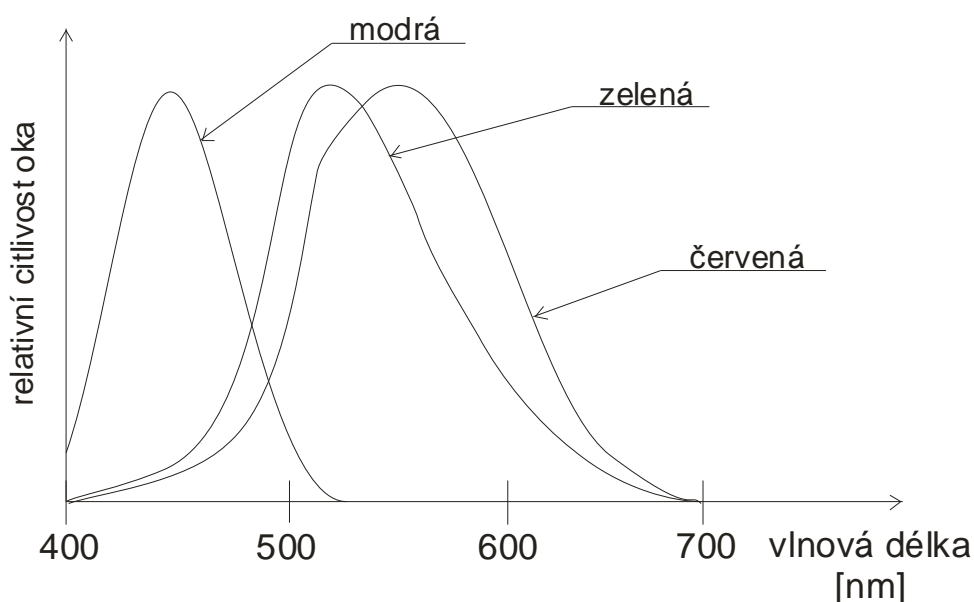
2.2. Barevné modely

Sítnice lidského oka je tvořena světlocitlivými buňkami, tyčinkami a čípkami [5]. Tyčinky umožňují černobílé vidění a čípky barevné. V lidském oku jsou 3 druhy čípků:

- červené - reagují zejména na červeno-žlutou barvu
- zelené - reagují zejména na zeleno-žlutou barvu
- modré - reagují zejména na modro-fialovou barvu.

Jejich maximální relativní citlivost přibližně odpovídá vlnovým délkám:

- červené (red, R) - $\lambda_R = 564 \text{ nm}$
- zelené (green, G) - $\lambda_G = 533 \text{ nm}$
- modré (blue, B) - $\lambda_B = 437 \text{ nm}$.

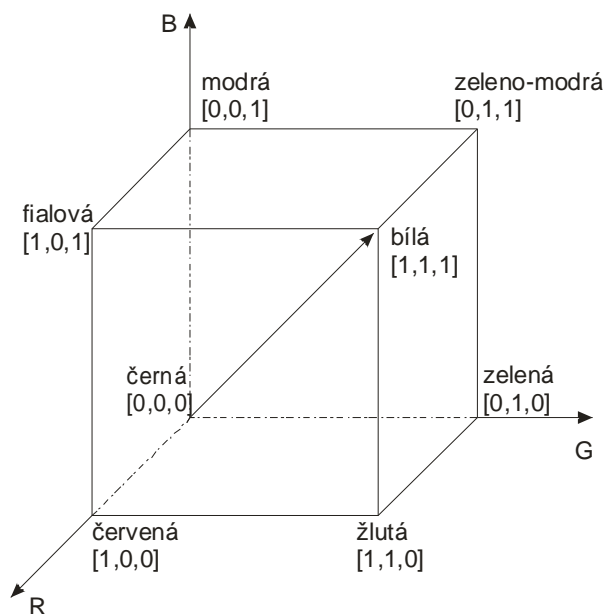


Obr. 2.2 Citlivost lidského oka na vnímané barvy

Pro tyto barvy R , G , B jsou definovány spektrální křivky, jejichž kombinací lze obsáhnout téměř celý rozsah lidského barevného vidění.

2.2.1. RGB model

Základní barevný model [3], z něhož jsou odvozeny další barevné modely, je RGB model. Výsledná barva je dána složením tří základních složek: červená (R), zelená (G) a modrá (B) s danou intenzitou. Z toho vyplývá, že každý pixel obrazu je reprezentován vektorem $[R \ G \ B]$ s danými intenzitami jednotlivých složek. Barevný prostor RGB je geometricky reprezentován jednotkovou krychlí, jejíž vrcholy jsou tvořeny základními barvami prostoru (obr. 2.3).



Obr. 2.3 Prostor *RGB*

Někdy je tento barevný model rozšířen o složku *A* (Alfa-kanál), kterou je stanovena průhlednost. Průhlednost může nabývat hodnot v intervalu $(0, 1)$. Hodnotou 0 je obraz označen jako neprůhledný vzhledem k pozadí, 1 jako zcela průhledný. Tento model je pak označován jako *RGBA*.

2.2.2. *Y* model

Y představuje jasovou hodnotu (šedotónový prostor), jde o jednorozměrný prostor, který se objevuje jako jedna ze složek vícerozměrných prostorů.

Převodní vztah z *RGB* na *Y* je:

$$Y = 0,299 \cdot R + 0,587 \cdot G + 0,114 \cdot B. \quad (2.1)$$

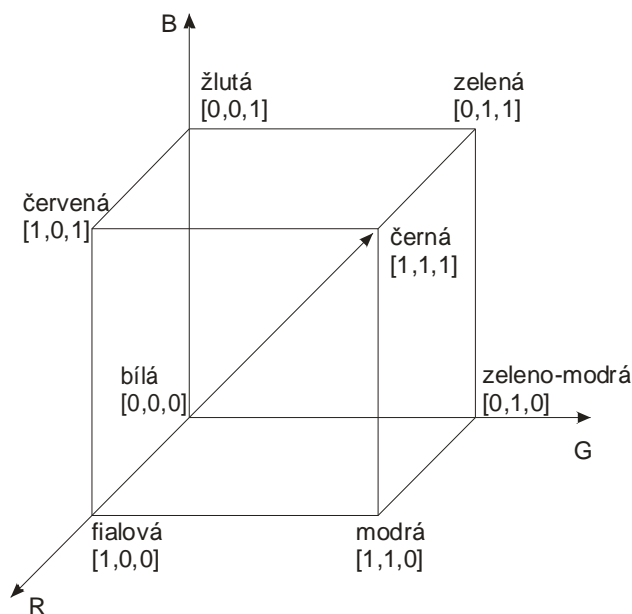
Pro tento prostor neexistuje inverzní transformace, z toho vyplývá, že po převodu *RGB* na *Y* ho nelze triviálně zpět rekonstruovat na *RGB*.

2.2.3. *CMY(K)* model

Model *CMY(K)* [7, 8] je doplňkovým modelem k modelu *RGB* a je složen ze tří základních barev:

- *C* - cyan, azurová
- *M* - magenta, fialová
- *Y* - yellow, žlutá,

jejichž kombinací je určena výsledná barva. Tento prostor je geometricky reprezentován jednotkovou krychlí (obr. 2.4).



Obr. 2.4 Prostor *CMY*

Převodní vztah mezi modely *RGB* a *CMY* je následující:

$$\begin{pmatrix} C \\ M \\ Y \end{pmatrix} = \begin{pmatrix} 1-R \\ 1-G \\ 1-B \end{pmatrix} \quad \begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1-C \\ 1-M \\ 1-Y \end{pmatrix}. \quad (2.2)$$

CMY barevný model je používán například u barevných inkoustových tiskáren. V praxi je obtížné namíchat z *C*, *M*, *Y* složek černou barvu a často vzniká barva tmavě hnědá. Tento model je proto ještě doplněn o černou barvu *K* (black), čímž jsou šetřeny při tisku jednotlivé *C*, *M*, *Y* barvy.

2.2.4. YCbCr model

YCbCr [7, 8] je barevný prostor, kde jednotlivé složky označují:

- *Y* - svítelnost, představuje jas
- *Cb* a *Cr* - chromizační signály, představují barvu.

Tento model je používán například pro barevné digitální televizní vysílání a vychází z prostoru *YUV*. Ve složce *Y* uchována informace o jasu (pro černobílé televize) a barevné informace jsou uloženy v *Cb* (dominantní informace o modré barvě) a *Cr* (dominantní informace o červené barvě) složkách.

Dále je tento model používán při ztrátové komprimaci obrázků (např. formát *JPEG*). Protože je lidské oko více citlivé na změnu jasových hodnot (složka *Y*) než na změny v barvách (chromizační signály), jsou ztráty (cca 50%) v barvách nepostřehnutelné, čímž dochází ke kompresi obrazu.

Vzájemná transformace mezi RGB a $YCbCr$ modelem je:

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} 0,257 & 0,504 & 0,098 \\ -0,148 & -0,291 & 0,439 \\ 0,439 & -0,368 & -0,071 \end{pmatrix} \cdot \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} 16 \\ 128 \\ 128 \end{pmatrix} \quad (2.3)$$

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1,164 & 0 & 1,596 \\ 1,164 & -0,391 & -0,813 \\ 1,164 & 2,018 & 0 \end{pmatrix} \cdot \begin{pmatrix} Y-16 \\ Cb-128 \\ Cr-128 \end{pmatrix}.$$

2.2.5. YIQ model

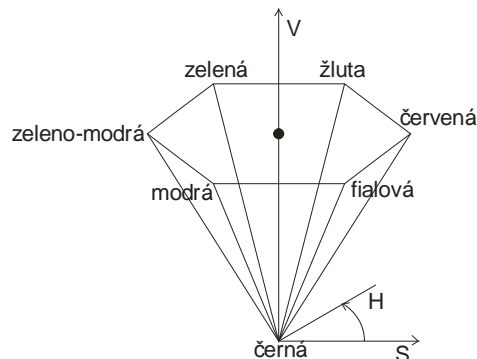
I tento model [6] vychází z modelu YUV a je opět určen pro barevné televizní vysílání (norma $NTSC$). Tento model je třírozměrný a opět je zde obsažena složka Y (nositelka jasové informace), kterou je zajištěna kompatibilita s černobílými televizemi. Ve složkách I (InPhase, ve fázi) a Q (Quadrature, způsob modulace) jsou uloženy informace o barvě.

Převodní vztahy mezi modely RGB a YIQ jsou:

$$\begin{pmatrix} Y \\ I \\ Q \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ 0,596 & -0,275 & -0,321 \\ 0,211 & -0,523 & 0,312 \end{pmatrix} \cdot \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad \begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1 & 0,956 & 0,621 \\ 1 & -0,272 & 0,647 \\ 1 & -1,107 & 1,704 \end{pmatrix} \cdot \begin{pmatrix} Y \\ I \\ Q \end{pmatrix}. \quad (2.4)$$

2.2.6. HSV model

Dalším barevným modelem je model HSV [7, 8]. Zde je opět barva reprezentována třemi složkami. Složkou H (Hue, barevnost) je dán barevný tón (převládající barva) v rozsahu $(0,360)^\circ$, S (Saturation, saturace) je označení sytosti (příměs jiných barev) a jeho rozsah je 0 (bílá) až 1 (spektrální barva). Třetí složkou je V (Value, hodnota), kterou je vyjádřeno množství bílého světla. Zde hodnoty mohou nabývat 0 (černá) až 1 (bílá). Tento prostor je geometricky zobrazen jako jehlan (obr. 2.5).



Obr. 2.5 Prostor HSV

Převodní vztah z *RGB* na *HSV* model je oproti výše popsaným modelům složitější:

$$\begin{aligned}
 V &= \max(R, G, B) \\
 \text{pro monochromatický obraz platí :} \\
 S &= 0 \\
 H &= 0 \\
 \text{pro barevný obraz platí :} \\
 S &= \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \\
 H &= \frac{G - B}{\max(R, G, B) - \min(R, G, B)} \quad \text{pro max} = R \\
 &= 2 + \frac{B - R}{\max(R, G, B) - \min(R, G, B)} \quad \text{pro max} = G \\
 &= 4 + \frac{R - G}{\max(R, G, B) - \min(R, G, B)} \quad \text{pro max} = B
 \end{aligned} \tag{2.5}$$

2.2.7. HSI model

Tento model [3] je velmi podobný modelu *HSV*. Jsou zde zachovány první dvě složky *H* a *S*, složka *V* je nahrazena složkou *I*, kterou je označena intenzita (Intenstity). Tento model je v prostotu zobrazen jako jehlan.

Převodní vztah z modelu *RGB* na *HSI* model je:

$$\begin{aligned}
 I &= \frac{R + G + B}{3} \\
 \text{pro monochromatický obraz platí :} \\
 S &= 0 \\
 H &= 0 \\
 \text{pro barevný obraz platí :} \\
 S &= \frac{3}{R + G + B} \cdot \min(R, G, B) \\
 H &= \text{stejně jako u modelu HSV}
 \end{aligned} \tag{2.6}$$

2.2.8. HSL model

Model *HSL* [7, 8] je třírozměrný, kde složkami *H* a *S* je dán barevný tón a sytost (jako u *HSV*). Složkou *L* je označena světlost (Lightness, světelnost). Geometrický lze tento prostor zobrazit jako dvojici rotačních kuželů.

Převod z modelu *RGB* na *HSL* je:

$$L = \frac{\max(R, G, B) + \min(R, G, B)}{2}$$

pro monochromatický obraz platí :

$$S = 0$$

$$H = 0$$

pro barevný obraz platí :

$$S(L < 0,5) = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B) + \min(R, G, B)}$$

$$S(L > 0,5) = \frac{\max(R, G, B) - \min(R, G, B)}{2 - \max(R, G, B) - \min(R, G, B)}$$

H = stejné jako u modelu HSV

(2.7)

2.3. Jasové transformace

Jasové transformace lze rozdělit do dvou skupin, jasové korekce a transformace jasové stupnice, v závislosti na tom, zda hodnota jasu při transformaci je závislá na poloze bodu v obraze.

2.3.1. Histogram

Histogramem [2] je vyjádřeno, kolik pixelů daného jasu je přítomno v obraze. Ve většině případů je zobrazen jako sloupcový graf. Na ose x jsou zobrazeny jednotlivé jasové úrovně a na ose y jsou zobrazeny počty bodů jednotlivých jasových úrovní. Histogram lze použít jako pomůcku, například při segmentaci obrazu nebo při změně jasové stupnice. Jelikož je v něm uložena informace o počtech bodů, nikoliv jejich poloha, nelze z histogramu zpětně rekonstruovat obraz, je tedy čistě informačního charakteru. Z toho také plyne, že jeden histogram lze přiřadit více obrazům. Jedinou podmínkou je zachování stejného počtu bodů obrazu. Histogram lze vytvořit pro šedotónové, i pro barevné obrazy. V případě barevného obrazu, je histogram zobrazen buď pro každou barevnou složku zvlášť (počet barevných složek ~ počet histogramů), nebo jsou jednotlivé složky složeny do jednoho histogramu, například průměrem jednotlivých složek nebo obraz je převeden na šedotónový, z kterého je pak vytvořen histogram.

2.3.2. Jasové korekce

U této transformace [3] je poloha bodu výstupního obrazu závislá na poloze bodu ve vstupním obraze. Jasové korekce jsou používány v případech, kdy snímací zařízení není stejně citlivé ve všech bodech obrazu. Ve snímacím zařízení je procházející světlo zeslabováno se vzrůstající vzdáleností od optické osy (tato vada se označuje jako vinětace). Další poruchy jsou nesejná citlivost CCD snímače, prach na optické soustavě a nerovnoměrné osvětlení.

Jestliže jsou tyto chyby systematické a lze je identifikovat, pak je možné pomocí jasových korekcí je eliminovat. Pokud $e(x, y)$ je multiplikativní koeficient (\sim chyba) a $g(x, y)$ snímaný obraz, pak platí:

$$f(x, y) = e(x, y) \cdot g(x, y), \quad (2.8)$$

kde $f(x, y)$ označuje zkreslený výstupní obraz. Výstupní nezkreslený obraz je dán vztahem:

$$g(x, y) = \frac{f(x, y)}{e(x, y)} = \frac{c \cdot f(x, y)}{f_c(x, y)}. \quad (2.9)$$

Proměnná f_c označuje etalonový obraz, který byl sejmutý při konstantním jasu c (etalonová šedá plocha).

Takto lze korigovat chyby obrazů, u kterých platí neměnnost snímacích podmínek.

2.3.3. Transformace jasové stupnice

Transformace s obrazem, které jsou zahrnuty do této skupiny, jsou nezávislé na poloze pixelu ve vstupním obraze. Těmito operacemi je modifikována jasová stupnice (barvy) tak, že jednotlivé hodnoty jasu jsou od sebe více vzdáleny a tím je dosaženo většího kontrastu obrazu. Tyto transformace jsou používány například při vizualizaci nekонтрастního obrazu, protože lidské oko není schopno rozeznat jasy, které jsou si blízké svojí hodnotou. U některých transformací jasové stupnice není možné provést zpětný převod, protože při nich dochází ke ztrátě informace (blízké hodnoty jasu mohou splynout v jednu hodnotu jasu).

Obecný zápis transformace jasové stupnice je:

$$q\langle q_0, q_k \rangle = t(p\langle p_0, p_k \rangle), \quad (2.10)$$

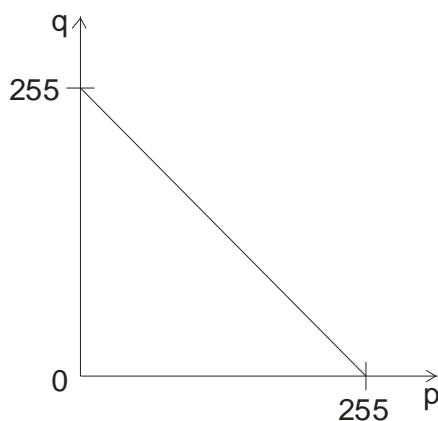
kde $p\langle p_0, p_k \rangle$ představuje interval vstupní jasové stupnice, $q\langle q_0, q_k \rangle$ je nová výstupní stupnice a τ označuje transformační funkci.

2.3.3.1. Negativ

Touto transformací jasové stupnice [3] je provedena inverze jednotlivých jasových hodnot (obr. 2.6) následujícím způsobem:

$$q = 255 - p \quad (2.11)$$

za předpokladu 256-ti úrovní jasu. Operace negativ je využívána například při skenování negativních předloh nebo při matematické morfologii (eroze, dilatace). Použitím funkce negativ nedochází ke splývání hodnot jasu .



Obr. 2.6 Převodní funkce pro negativ

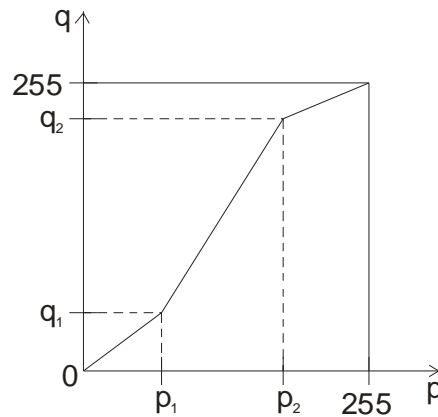
2.3.3.2. Změna kontrastu

Změnou kontrastu [3] (obr. 2.7) lze zvětšit kontrast výstupního obrazu pro původně nekонтastní obraz.

Převod je dán rovnicemi:

$$\begin{aligned} q(0, q_1) &= \frac{q_1}{p_1} \cdot p(0, p_1) \\ q(q_1, q_2) &= \frac{q_2 - q_1}{p_2 - p_1} \cdot p(p_1, p_2) \\ q(q_2, 255) &= \frac{255 - q_2}{255 - p_2} \cdot p(p_2, 255) \end{aligned} \quad (2.12)$$

za předpokladu jasového rozsahu (0, 255). q je výstupní stupnice jasu, p vstupní stupnice jasu a q_1 , q_2 , p_1 a p_2 jsou převodní koeficienty. Při této operaci může docházet ke splýnutí blízkých hodnot jasu v závislosti na převodních koeficientech.



Obr. 2.7 Převodní funkce pro změnu kontrastu

2.3.3.3. Změna jasu

Při této transformaci je přičten (odečten) všem bodům obrazu konstantní jas. Při tomto převodu může dojít ke ztrátě jasových úrovní z důvodu saturace, přetečení (podtečení) intervalu, v závislosti na velikosti jasu daného bodu a přičítaného (odčítaného) koeficientu. Pokud je uvažován rozsah jasové stupnice (0, 255), pak je nutno tento problém ošéřit tak, že výstupní hodnoty větší než 255 jsou touto hodnotou omezeny. Obdobně to platí pro hodnoty jasů menší než 0, které jsou pak touto hodnotou též omezeny.

Převod lze obecně zapsat:

$$\begin{aligned}
 q(q_0, q_k) &= c + p(p_0, p_k) \\
 \text{pro } q > q_k &\Rightarrow q = q_k \\
 \text{pro } q < q_0 &\Rightarrow q = q_0
 \end{aligned}
 \tag{2.13}$$

kde q je výstupní stupnice jasu, p vstupní stupnice jasu, c označuje konstantní jasovou hodnotu. Funkcí změny jasu lze upravit příliš tmavé nebo světlé obrazy na obrazy s vhodnou jasovou úrovní.

2.3.3.4. Ekvalizace histogramu

Někdy je tato metoda nazývána vyrovnání histogramu [3]. Tímto algoritmem je změněno rozložení jasů v obraze tak, aby v něm byly jednotlivé jasové úrovně zastoupeny přibližně stejně četně. Touto metodou je zvýšen kontrast (vzdálenost mezi jasy) blízko maximálních hodnot histogramu a naopak, u jasů, které nejsou příliš zastoupeny, je kontrast snižen.

Matematický popis ekvalizace histogramu je:

$$\sum_{i=0}^k G(q_i) = \sum_{i=0}^k H(p_i). \quad (2.14)$$

$H(p)$ označuje výchozí histogram s jasovou stupnicí $p \langle p_0, p_k \rangle$ a $G(q)$ vyvážený histogram se stupnicí $q \langle q_0, q_k \rangle$. Výsledná transformace jasové stupnice je::

$$q = \frac{q_k - q_0}{M \cdot N} \cdot \sum_{i=p_0}^p H(i) + q_0, \quad (2.15)$$

kde M, N jsou rozměry obrázku.

2.4. Geometrické transformace

Geometrické transformace jsou transformace, kterými je možné podle daného transformačního vztahu vypočítat polohu bodu ve výstupním obraze na základě znalosti jeho polohy ve vstupním obraze. Jednotlivé souřadnice bodů obrazové mřížky jsou tvořeny celými čísly. Při výpočtu nové polohy bodu nejsou jeho souřadnice celá čísla, ale reálná čísla. Z toho vyplývá, že výstupnímu bodu neodpovídá přesně bod ve vstupním obrazu a tím nelze jednoduše určit jas tohoto bodu. Proto je nutné jasovou hodnotu aproximovat. Úloha geometrických transformací je rozdělena na dvě části, transformace souřadnic bodů a aproximace jasové funkce.

2.4.1. Transformace souřadnic bodů

Obecný zápis transformace souřadnic [3] je:

$$x' = \sum_{r=0}^m \sum_{k=0}^{m-r} a_{rk} x^r y^k \quad y' = \sum_{r=0}^m \sum_{k=0}^{m-r} b_{rk} x^r y^k, \quad (2.16)$$

kde koeficienty a_{rk} , b_{rk} jsou parametry transformace. Tyto koeficienty lze určit pomocí metody nejmenších čtverců při známé poloze odpovídajících dvojic bodů ve vstupním a výstupním obraze x , y a x' , y' . Výše uvedená rovnice je často nahrazena bilineární transformací. Pro operace změna měřítka, zkosení a rotace se v praxi používá zvláštní případ bilineární transformace, tzv. afinní transformace, kterou lze v případě homogenních souřadnic vyjádřit maticově:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_0 \\ b_1 & b_2 & b_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \quad (2.17)$$

2.4.1.1. Změna měřítka

Touto transformací [3] je možno změnit velikost obrazu, šířku a výšku. Transformační matice je zapsána:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}, \quad \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{a} & 0 \\ 0 & \frac{1}{b} \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix}, \quad (2.18)$$

kde x', y' jsou souřadnice ve výstupním obraze, x, y jsou souřadnice ve vstupním obraze, a, b jsou koeficienty změny měřítka x -ové a y -ové souřadnice.

2.4.1.2. Zkosení

Operací zkosení je [3] upraveno naklopení obrazu ve směru osy x . Osa y je nezměněná.

Převodní matice jsou:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & \operatorname{tg}(\varphi) \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}, \quad \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\operatorname{tg}(\varphi) & 1 \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix}, \quad (2.19)$$

kde x', y' jsou souřadnice ve výstupním obraze, x, y jsou souřadnice ve vstupním obraze, φ označuje úhel zkosení.

2.4.1.3. Rotace

Touto transformací [3] je obraz otočen o požadovaný úhel. Transformace je provedena dle vztahů:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}, \quad \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix}, \quad (2.20)$$

kde x', y' jsou souřadnice ve výstupním obraze, x, y jsou souřadnice ve vstupním obraze, φ označuje úhel rotace (otočení).

2.4.2. Aproximace jasové hodnoty

Vstupní i výstupní obraz je tvořen celočíselnou mřížkou. V případě, kdy je určována hodnota jasu bodu (x', y') výstupního obrazu z polohy tohoto bodu (x, y)

ve vstupním obraze, je jeho poloha (x, y) ve vstupním obraze dána použitou geometrickou transformační funkcí. Při převodu polohy bodu (x', y') z výstupního na vstupní obraz, hodnoty souřadnic bodu (x, y) nenabývají celočíselných hodnot. Z toho vyplývá, že není jednoznačně určena hodnota jasu pro bod (x', y') výstupního obrazu z polohy tohoto bodu (x, y) ve vstupním obraze. Hodnota jasu pro výstupní bod (x', y') obrazu je určena aproximací jasu v okolí bodu (x, y) ve vstupním obraze. V závislosti na tom, jak velké okolí je použito pro určení jasu, jsou aproximace rozděleny do několika skupin.

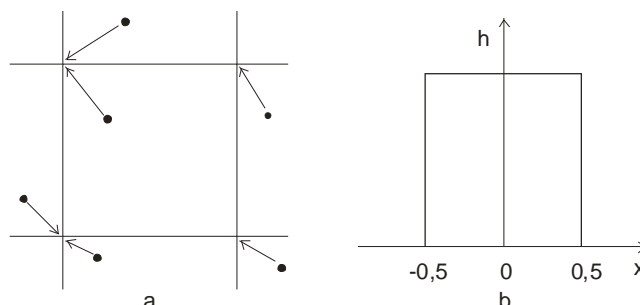
2.4.2.1. Interpolace metodou nejbližšího souseda

Interpolace metodou nejbližšího souseda je základní a výpočetně nejrychlejší aproximační metoda [9], kterou je přiřazena bodu (x, y) ve vstupním obraze hodnota jasu nejbližšího bodu v celočíselné mřížce (obr. 2.8a).

Interpolaci lze zapsat:

$$f(x, y) = g(\text{round}(x), \text{round}(y)), \quad (2.21)$$

kde f je jas, g označuje celočíselnou mřížku, x, y jsou reálné souřadnice. Chyba, která může při tomto způsobu interpolace vzniknout, je maximálně půl pixelu (obr. 2.8b).



Obr. 2.8 (a) interpolace metodou nejmenšího souseda, (b) chyba

Interpolací výše popsanou metodou nejsou rozmazány hrany, ale transformovaný obraz je schodovitý.

2.4.2.2. Lineární interpolace

U lineární interpolace [3] je použito k výpočtu hodnoty jasu v daném bodě (x, y) vstupního obrazu okolí čtyř bodů (obr. 2.9a). Vliv jednotlivých okolních bodů na bod (x, y) je úměrný jejich vzdálenosti.

Výpočet lineární interpolace je dán vztahem:

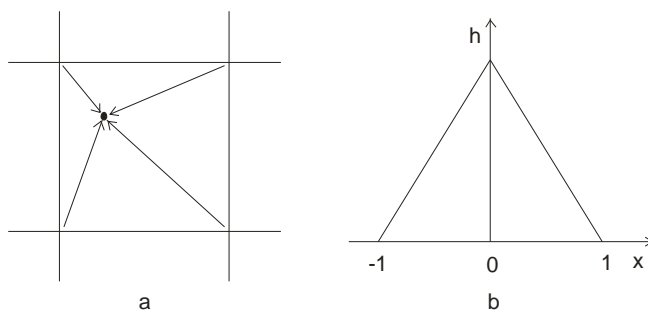
$$f(x, y) = (1-a) \cdot (1-b) \cdot g(l, k) + a \cdot (1-b) \cdot g(l+1, k) + (1-a) \cdot b \cdot g(l, k+1) + a \cdot b \cdot g(l+1, k+1), \quad (2.22)$$

$$l = \text{round}(x), \quad a = x - l$$

$$k = \text{round}(y), \quad b = y - l$$

kde f je jas, g označuje celočíselnou mřížku, x, y jsou reálné souřadnice, l, k jsou souřadnice interpolačního jádra, a, b představují koeficienty, vyjadřující vzdálenost k bodům interpolačního jádra.

Touto metodou interpolace jsou hrany obrazu mírně rozmazány, ale nedochází zde ke schodovitosti tak jako u interpolace metodou nejbližšího souseda. Zlepšení této metody oproti metodě nejbližšího souseda jsou viditelné především na šikmých čarách (zvláště u zkosení a rotace, které jsou více vyhlazené).



Obr. 2.9 (a) lineární interpolace, (b) interpolační jádro v 1D případě

2.4.2.3. Bikubická interpolace

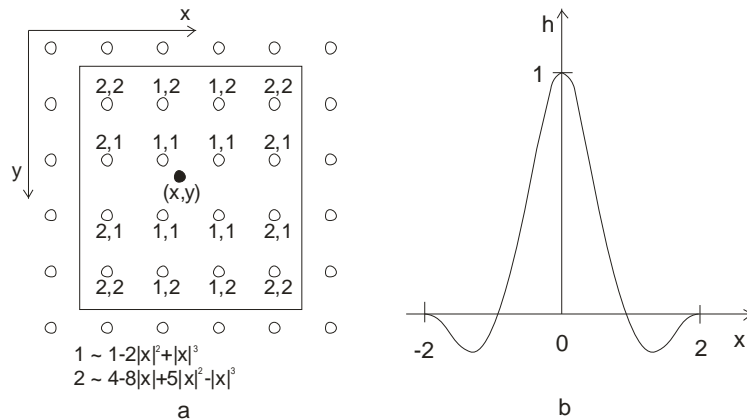
Při bikubické interpolaci je použito k určení hodnoty jasu okolí 16-ti bodů (obr. 2.10a). Vliv jednotlivých bodů okolí na jas vyšetřovaného bodu je dán bikubickým polynomem [3].

Pro jednoduchost je zde uveden výpočet jednotlivých vah interpolačního jádra (obr. 2.10b) pouze pro jeden rozměr, výpočet pro druhý rozměr je stejný. Bikubická interpolace je popsána:

$$h(x) = \begin{cases} 1 - 2 \cdot |x|^2 + |x|^3 & \text{pro } 0 \leq |x| < 1 \\ 4 - 8 \cdot |x| + 5 \cdot |x|^2 - |x|^3 & \text{pro } 1 \leq |x| < 2 \\ 0 & \text{jinde} \end{cases} \quad (2.23)$$

Výsledný jas je dán součtem součinů jednotlivých vah $h(x), h(y)$ s hodnotou jasu v příslušných bodech bikubického jádra. Bikubická interpolace je časově nejnáročnější na výpočet, na druhou stranu však dává nejlepší výsledky z uvedených interpolací,

protože nejlépe aproximuje jasy detailů (pokud jsou detaily nebo hrany tvořeny více jak jedním pixelem).



Obr. 2.10 (a) bikubická interpolace, (b) interpolační jádro 1D případě

2.5. Filtrace šumu a poruch

Metodami filtrace šumu je z obrazu eliminován šum, tzn. že jsou z něho odstraněny nežádoucí informace, a následně je obraz vyhlazen. Obecně lze rozdělit metody filtrace šumu do dvou základních skupin. Jsou to lineární metody vyhlazování a nelineární metody vyhlazování. V obou případech se jedná o metody, které jsou zařazeny do skupiny lokálních metod vyhlazování. Těmito metodami je vypočtena nová hodnota jasu pixelu na základě jeho malého okolí.

2.5.1. Lineární metody vyhlazování

Nová hodnota pixelu je určena lineární kombinací jasů, které tvoří zkoumané okolí.

2.5.1.1. Prostý průměr

Při použití této metody jsou průměrovány jasy bodů, které jsou zahrnuty ve zvoleném okolí. Při filtrování prostým průměrem je volena čtvercová maska, přičemž filtrovaný bod leží uprostřed této masky. Z toho vyplývá, že maska je tvořena lichým počtem řádků a sloupců.

Výchozí maskou je maska o rozměrech 3x3 (viz 2.24a), další rozměry masek jsou 5x5, 7x7 atd. V některých případech se zvětšuje váha středového bodu masky (viz 2.24b) nebo váha jeho 4-sousedů (viz 2.24c).

$$h = \frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad h = \frac{1}{10} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad h = \frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \quad (2.24)$$

a b c

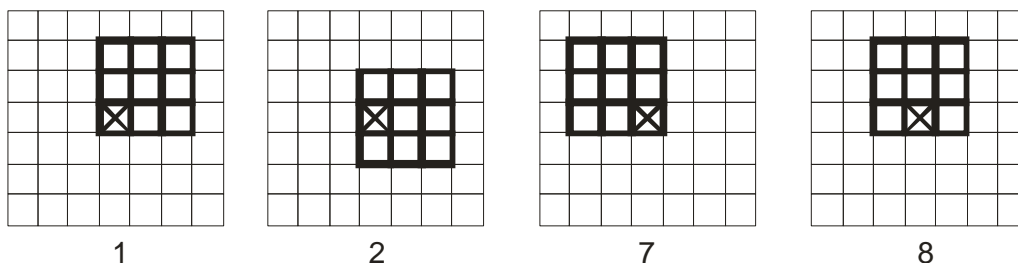
Velikost masky je volena v závislosti na velikostech detailů, které se mají být v obraze zachovány. Nevýhodou této metody je rozmazávání detailů obrazu.

2.5.2. Nelineární metody vyhlazování

Těmito metodami je vypočítána nová hodnota jasu daného pixelu jenom z takové části okolí, do kterého opravdu patří. Tím lze částečně eliminovat rozmazání obrazu.

2.5.2.1. Metoda rotující masky

Filtrace touto metodou [10] je z uváděných metod časově nejnáročnější. Pro tento způsob filtrace je převážně volena čtvercová maska, která rotuje kolem reprezentativního (filtrovaného) bodu. Pro masku 3x3 je počet možných poloh osm (obr. 2.11).



Obr. 2.11 Metoda rotující masky

V každé masce je spočítán rozptyl jasů. Dále je vybrána maska s nejmenším rozptylem, ve které je spočítán aritmetický průměr jasů, který je pak dosazen na místo filtrovaného bodu.

Výpočet rozptylu je:

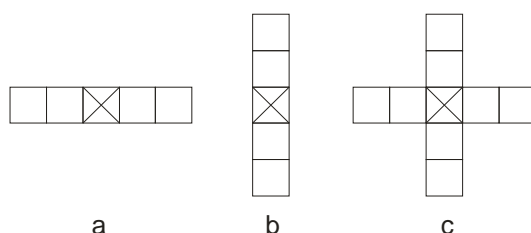
$$r = \frac{\sum_{n=0}^{N-1} \sqrt{\left(\bar{p} - p_n \right)^2}}{N-1}, \quad (2.25)$$

kde t je rozptyl, \bar{p} aritmetický průměr z hodnot p_n .

Metodou rotující masky není obraz nerozmazáván a je mírně ostřen.

2.5.2.2. Metoda mediánu

Metodou mediánu [11] je stanovena nová hodnota jasu filtrovaného bodu na základě nalezeného mediánu (prvek, který je po seřazení uprostřed) z hodnot jasů v použité masce. Obvykle jsou používány masky se čtvercovým okolím. U obrazů, ve kterých jsou obsaženy tenké čáry nebo ostré hrany, mohou být touto metodou porušeny. Aby byly eliminovány tyto nepříjemnosti, jsou používány masky, které umožňují tyto čáry zachovat neporušené (obr. 2.12).



Obr. 2.12 Medián - masky zachovávající hrany

Filtrací mediánem je částečně sníženo rozmazání hran a je velmi dobře redukován impulsní šum v obraze.

2.6. Detekce hran v obraze

Detektory hran [2] jsou v obraze hledány hranice, které jsou tvořeny rozdílnými hodnotami jasů sousedních bodů. Hrana je určena svojí velikostí a směrem. Tyto veličiny lze určit z gradientu obrazové funkce $f(x, y)$ a indikují rychlost změny obrazové funkce v ploše. Gradient je dán parciálními derivacemi dle proměnných x a y v daném bodě.

Velikost gradientu je poté:

$$|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}, \quad (2.26)$$

a směr gradientu je:

$$\Psi = \arctg \left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}} \right). \quad (2.27)$$

Protože tyto vztahy platí pro spojitou funkci a s obrazem je pracováno jako s diskrétní funkcí, je parciální derivace aproximována diferencemi:

$$\begin{aligned}\Delta_x g(x, y) &= g(x, y) - g(x - n, y) \\ \Delta_y g(x, y) &= g(x, y) - g(x, y - n)\end{aligned}\quad (2.28)$$

Gradientní operátory lze rozdělit do skupin:

1) Operátory, jež aproximují derivaci obrazové funkce pomocí diference. První podskupinou jsou takové operátory, jež jsou tvořeny jedinou maskou, která je invariantní vůči otočení. Druhá podskupina je reprezentována operátory, jež jsou tvořeny více maskami, které jsou směrově závislé. Z nich je vybrána maska, která nejlépe aproximuje obrazovou funkci. Tím je určen směr gradientu.

2) Do této skupiny jsou zařazeny operátory, jejichž princip detekce hran je založen na hledání míst, kde druhá derivace obrazové funkce prochází nulou. Příkladem je Cannyho hranový detektor.

3) V této skupině jsou zahrnuty operátory, jež aproximují obrazovou funkci jednoduchým parametrickým modelem (polynom dvou proměnných).

Někdy jsou operátory detekce hran využívány k ostření obrazů [4] s cílem upravit obraz tak, aby hrany v něm měly větší strmost:

$$f(x, y) = g(x, y) - C \cdot S(x, y), \quad (2.29)$$

kde $f(x, y)$ je výsledek ostření, $g(x, y)$ je ostřený obraz, C označuje koeficient ostření udávající sílu ostření a $S(x, y)$ je obraz $g(x, y)$, na který byl aplikován gradientní operátor.

2.6.1. Operátory aproximující derivaci obrazové funkce

V této části jsou zmíněny operátory, které k detekci hran využívají konvoluci operátoru a okolí reprezentativního bodu.

2.6.1.1. Robertsův operátor

U tohoto velmi jednoduchého operátoru [12] je použito jen malého okolí vyšetřovaného bodu. Masky jsou rozměru 2×2 :

$$h_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad h_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad (2.30)$$

a velikost gradientu se počítá podle:

$$|\nabla g(x, y)| = |g(x, y) - g(x+1, y+1)| + |g(x, y+1) - g(x+1, y)|. \quad (2.31)$$

Nevýhodou Robertsova operátoru je jeho velká citlivost na šum.

2.6.1.2. Laplaceův operátor

Operátor [2] využívá dvě základní masky, které jsou definovány buď pro 4-sousedství nebo 8-sousedství (viz 2.32). Obě masky jsou nezávislé na otočení a jsou definovány pro okolí 3x3:

$$h_4 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad h_8 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (2.32)$$

Někdy se používají masky, které berou v potaz vzdálenost k reprezentativnímu bodu masky (viz 2.33), v tomto případě se ztrácí invariantnost vůči otočení:

$$h = \begin{bmatrix} 2 & -1 & 2 \\ -1 & -4 & -1 \\ 2 & -1 & 2 \end{bmatrix}, \quad h = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}. \quad (2.33)$$

Nevýhodu těchto operátorů je jejich citlivost na šum v okolí.

2.6.1.3. Operátor Prewittové

Tento operátor [2] je tvořen osmi maskami a každá z nich je specifická pro jeden směr. Po aplikaci masek je vybrána ta, které odpovídá největší modul gradientu (součet jasu pod maskou). Základní maska je označena h_1 , ostatní masky jsou vytvořeny otáčením okolo středového bodu masky:

$$h_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, \quad h_2 = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}, \quad h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad h_4 = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix},$$

$$h_5 = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \quad h_6 = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}, \quad h_7 = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}, \quad h_8 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & - \end{bmatrix}. \quad (2.34)$$

2.6.1.4. Sobelův operátor

Pro tento operátor [2] je také definováno osm směrových masek, které jsou vytvořeny stejným způsobem jako u operátoru Prewittové, proto jsou zde uvedeny pouze první tři:

$$h_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, \quad h_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}, \quad h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}. \quad (2.35)$$

Pro Sobelův operátor jsou zajímavé masky h_1 a h_3 , které lze použít samostatně jako detektory vodorovných a svislých hran.

2.6.1.5. Robinsonův operátor

Pro Robinsonův operátor [2] platí analogie s operátorem Prewittové:

$$h_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}, \quad h_2 = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix}, \quad h_3 = \begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix}. \quad (2.36)$$

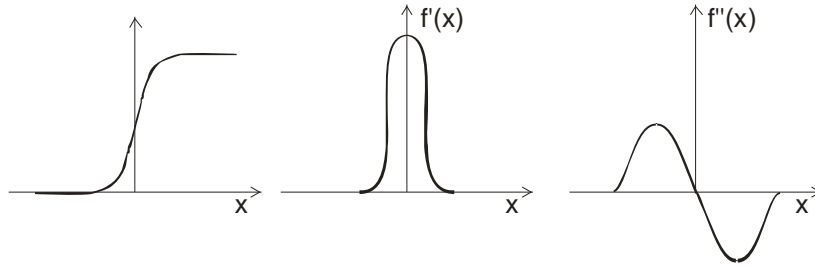
2.6.1.6. Kirschův operátor

Opět analogie s operátorem Prewittové [2]:

$$h_1 = \begin{bmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{bmatrix}, \quad h_2 = \begin{bmatrix} 3 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & -5 & 3 \end{bmatrix}, \quad h_3 = \begin{bmatrix} -5 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & 3 & 3 \end{bmatrix}. \quad (2.37)$$

2.6.2. Detekce hran pomocí druhé derivace obrazové funkce

Operátory uvedené v kapitole 2.6.1. jsou citlivé na šum, na velikost detailů a na měřítko obrazu, protože aproximují derivaci diferencemi ve velmi v blízkém okolí bodu (zvláště Robertsův operátor). Detekce hran pomocí druhé derivace obrazové funkce [13] je založena na průchodu druhé derivace nulou (obr. 2.13).



Obr. 2.13 Derivace obrazové funkce

Aby operátor detekce hran nebyl citlivý na šum, musí být navržen tak, aby byl hladký, ve frekvenční oblasti se choval přibližně jako pásmová propust, ale zároveň aby pracoval pouze s body v blízkém okolí hledané hrany.

2.6.2.1. LoG operátor

Jedním z operátorů, který patří do této skupiny, je LoG operátor (Laplacian of Gaussian). Vztah pro výpočet jednotlivých koeficientů v konvoluční masce vychází z 2D gaussovského rozložení:

$$G(x, y) = e^{-\frac{x^2+y^2}{2s^2}}, \quad (2.38)$$

po provedení první a druhé derivace $G(x,y)$ je:

$$h(x, y) = c \cdot \left(\frac{x^2 + y^2 - s^2}{s^4} \right) \cdot e^{-\frac{x^2+y^2}{2s^2}}, \quad (2.39)$$

kde h jsou hodnoty v konvoluční masce, c je koeficient, který zajišťuje, že součet v masce je nula, $x^2 + y^2$ je euklidovská vzdálenost od středu Gaussiánu, σ označuje střední kvadratickou odchylku, která udává, jak je operátor velký.

Aproximace LoG operátoru v masce 5x5 [14] je:

$$h = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}. \quad (2.40)$$

Z masky je zřejmé, že body vzdálené více od středu jsou brány s menší vahou.

2.7. Segmentace

Segmentace [1] jsou metody předzpracování a rozpoznávání obrazu, které vedou k hledání objektů v obraze. Jako objekty v obraze jsou označeny části, které jsou dále předmětem zpracování a rozpoznávání. Segmentací je obraz rozdělen na objekty a na pozadí (část obrazu, která nebudou dále předmětem zpracování a rozpoznávání).

Segmentaci obrazu lze rozdělit podle očekávaných výsledků do dvou skupin - kompletní segmentace a částečná segmentace.

Při kompletní segmentaci odpovídají oblasti v segmentovaném obrazu objektům ve vstupním obraze. Tento typ segmentace je používán v úlohách, kde je předem známa řešená problematika.

Opakem je částečná segmentace, kde segmentované oblasti přímo nesouhlasí objektem ve vstupním obraze. V této metodě je využíváno k určování objektů obrazových vlastností, například stejná barva (nebo velmi blízká) nebo jas uvnitř oblasti.

2.7.1. Prahování

Prahování [1] je jednoduchá a zároveň velmi rychlá metoda segmentace na základě určení jasové konstanty označované jako práh. Hodnoty jasu v jednotlivých bodech obrazu jsou pak následně převedeny (uvažován jasový rozsah (0, 255)) podle:

$$\begin{aligned} g(x, y) &= 255 \quad \text{pro } f(x, y) \geq P \\ g(x, y) &= 0 \quad \text{pro } f(x, y) < P \end{aligned} \quad (2.41)$$

P označuje práh, $g(x, y)$ je výstupní obraz a $f(x, y)$ vstupní obraz.

Někdy je použito prahování pomocí intervalu, převodní funkce je:

$$\begin{aligned} g(x, y) &= 0 \quad \text{pro } f(x, y) \in D \\ g(x, y) &= 255 \quad \text{mimo interval} \end{aligned} \quad (2.42)$$

kde D označuje interval, ve kterém jsou hodnoty jasu nahrazeny nulovým jasem (černá barva). Touto modifikací prahování je umožněno z obrazu odstranit světlé a tmavé oblasti.

Prahováním je původní obraz převeden na binární, obsahuje pouze 2 jasové úrovně - 0 a 255 (~ 1).

2.8. Matematická morfologie

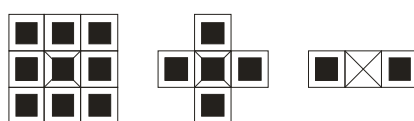
Zpracování obrazu s využitím matematické morfologie [15, 16] je založeno na předpokladu, že obraz lze chápat jako bodové množiny a tím využívat jejich

vlastností. Při těchto operacích jsou využity množinové pojmy: podmnožina, průnik, sjednocení, prázdná množina, množinový doplněk a množinový rozdíl.

2.8.1. Binární matematická morfologie

Binární matematické transformace [16, 17] jsou určeny pro práci s binárním obrazem. To znamená, že před použitím těchto operací musí být obraz převeden na binární (pomocí segmentace – obsahuje dvě barvy - černou (0) a bílou (1)). Binární obraz je pak vyjádřen jako 2D množina. Množina X obsahuje body obrazu s hodnotou 1 (objekty), a X^c je množina bodů obrazu s hodnotou 0 (pozadí).

Morfologická transformace je relace mezi množinami, v tomto případě mezi množinou X a B (tzv. strukturní element). Typické strukturní elementy jsou na obr. 2.14.



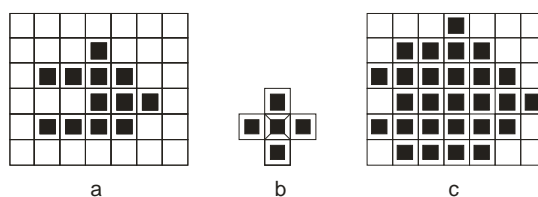
Obr. 2.14 Typické elementy pro binární matematickou morfologii

Morfologickou transformaci si lze představit tak, že je strukturní element B posouván po obraze X a výsledek relace (mezi množinou X a B) je zapsán do výstupního souboru.

2.8.1.1. Dilatace

Dilatace [16, 17] (obr. 2.15) je zapsána:

$$X \oplus B = \{p \in e^2 : p = x + b, x \in X, b \in B\} \quad (2.43)$$



Obr. 2.15 Dilatace - (a) vstupní obraz, (b) strukturní element, (c) dilatace

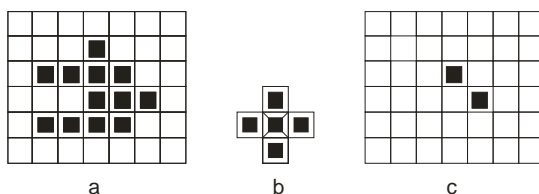
Strukturní element je aplikován v bodech, kde je hodnota vstupního obrazu 1 . V těchto bodech je element přiložen a zapsán do výstupního obrazu. Operace dilatace rozšiřuje objekty a zaplňuje malé díry v závislosti na velikosti strukturního elementu.

2.8.1.2. Eroze

Operace eroze [16, 17] (obr. 2.16) je duální operaci k dilataci, tj. objekty jsou zmenšovány. Tyto operace nejsou vzájemně inverzní, z toho vyplývá, že po provedení eroze a následně dilatace nebo dilatace a poté eroze nemusí odpovídat výsledný obraz původnímu.

Eroze je dána předpisem:

$$X \ominus B = \{p \in e^2 : p + b \in X \text{ pro každé } b \in B\}. \quad (2.44)$$



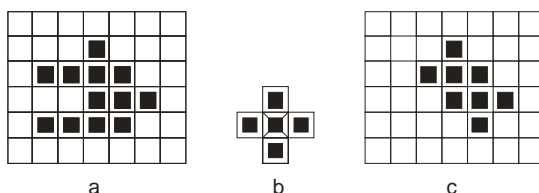
Obr. 2.16 Eroze - (a) vstupní obraz, (b) strukturální element, (c) eroze

Vstupní obraz je postupně procházen a v bodech, kde je jeho hodnota rovna 1, je strukturální element přiložen reprezentativním bodem. V případě, že je element shodný s okolím vyšetřovaného bodu, je do výstupního obrazu zapsána 1, jinak 0.

2.8.1.3. Otevření

Otevření [16, 17] (obr. 2.17) je transformace, kde je eroze následována dilatací:

$$X \circ B = (X \ominus B) \oplus B \quad (2.45)$$



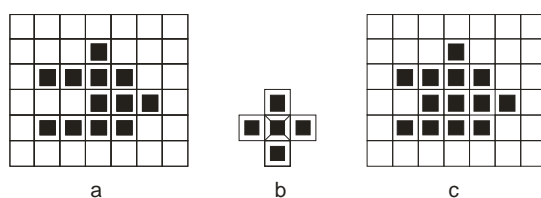
Obr. 2.17 Otevření - (a) vstupní obraz, (b) strukturální element, (c) otevření

Při těchto operacích je používán stejný strukturální element. Operací otevření jsou z obrazu odstraněny detaily, které jsou menší než strukturální element, a odděleny objekty spojené úzkou šňůrkou.

2.8.1.4. Uzavření

Operace uzavření [16, 17] (obr. 2.18) označuje transformaci, kde je dilatace následována erozí se stejným strukturálním elementem:

$$X \bullet B = (X \oplus B) \ominus B. \quad (2.46)$$



Obr. 2.18 Uzavření - (a) vstupní obraz, (b) strukturní element, (c) uzavření

Operací otevření jsou z obrazu odstraněny detaily, které jsou menší než strukturní element, spojeny objekty, které jsou blízko sebe a zaplněny malé díry.

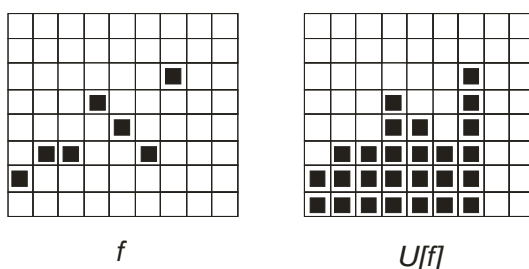
2.8.1.5. Tref či miň

Tímto operátorem [16, 17] je hledána úplná shoda části obrazu se strukturním elementem. V případě shody je do výstupního zapsána 1, jinak 0. Zápis této operace je:

$$X \otimes B = (x : B_1 \subset X \text{ a } B_2 \subset X^c). \quad (2.47)$$

2.8.2. Šedotónová matematická morfologie

Operace šedotónové matematické morfologie jsou používány na šedotónové obrazy. U binárních matematických morfologií je uvažován 2D prostor, v tomto případě je uvažován 3D euklidovský prostor, kde jsou jednotlivé body dány třemi souřadnicemi. První dvě jsou souřadnice v rovině a třetí je výška, v tomto případě je to hodnota jasů. Důležitými pojmy jsou vršek a stín [17]. Vršek je nejvyšší hodnota poslední souřadnice prostoru. Stín je množina složená z vršku a prostoru pod ním.



Obr. 2.19 Obrazová funkce f a její stín $U[f]$

Pro tyto morfologické operace jsou opět definovány operátory dilatace, eroze, otevření a uzavření.

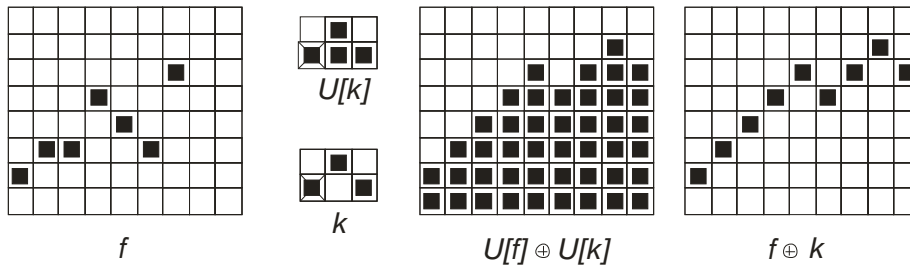
2.8.2.1. Šedotónová dilatace

Šedotónová dilatace je dána vztahem:

$$f \oplus k = T\{U[f] \oplus U[k]\}$$

$$(f \oplus k)(x) = \max\{f(x-z) + k(z), z \in K, x-z \in F\}$$
(2.48)

kde $U[f]$ představuje stín dilatovaného obrazu a $U[k]$ stín strukturního elementu.



Obr. 2.20 Šedotónová dilatace

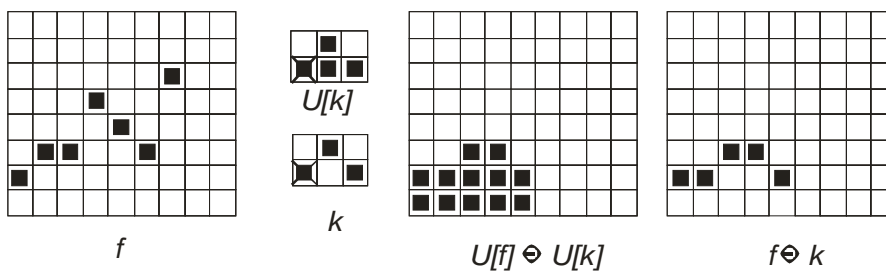
Na obr. 2.20 je zobrazen příklad šedotónové dilatace obrazu f se strukturním elementem k .

2.8.2.2. Šedotónová eroze

Šedotónová eroze je definována:

$$f \ominus k = T\{U[f] \ominus U[k]\}$$

$$(f \ominus k)(x) = \min\{f(x+z) - k(z), z \in K\}$$
(2.49)



Obr. 2.21 Šedotónová eroze

Na obr. 2.21 je zobrazen příklad šedotónové eroze obrazu f se strukturním elementem k .

2.8.2.3. Šedotónové otevření a uzavření

Pro tyto operace platí analogický vztah jako u binárního otevření a uzavření, postup při vykonávání dílčích operací je stejný. Pro otevření, respektive uzavření jsou

dány vztahy:

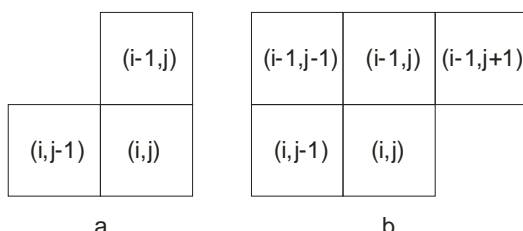
$$\begin{aligned} f \circ k &= (f \ominus k) \oplus k \\ f \bullet k &= (f \oplus k) \ominus k. \end{aligned} \tag{2.50}$$

2.9. Rozpoznávání objektů

Metod, které provádějí rozpoznávání objektů je mnoho. Proto jsou dále zmíněné metody, které jsou implementovány v programu.

2.9.1. Barvení

Barvení [18] je základní úlohou v rozpoznávání, která poskytuje popis jednotlivých oblastí. Barvením jsou rozděleny a rozlišeny souvislé plochy (tvary) v obraze od pozadí a výsledkem je obraz, ve kterém jsou souvislé plochy od sebe barevně odděleny. K určování, zda jednotlivé body tvoří spojitou plochu, jsou používány masky pro barvení (obr. 2.22).



Obr. 2.22 Masky pro barvení, (a) 4-soussednost, (b) 8-soussednost

Průběh operace barvení (obr. 2.23) je rozdělen do dvou fází.

V první fázi je binární obraz procházen po řádcích od shora dolů po jednotlivých bodech. Každému nenulovému bodu je přiřazena hodnota v závislosti na hodnotách sousedních bodů, které jsou dány maskou pro barvení (obr. 2.22 a, b). Všechny body pod maskou byly obarveny v předchozích krocích. Danému bodu je přiřazena hodnota podle vzniklé situace:

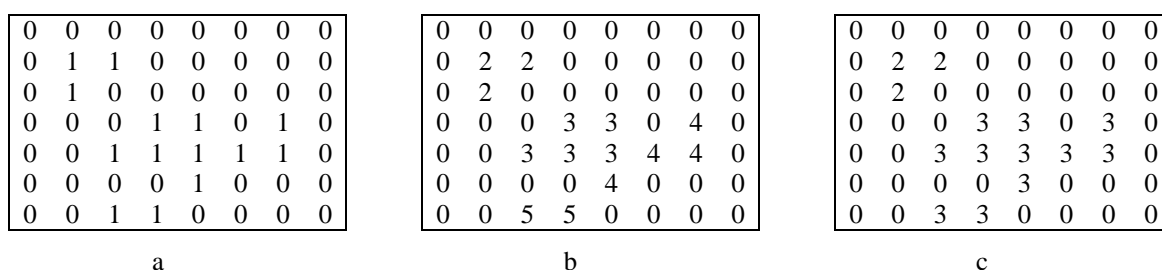
- 1) všechny body v okolí (pod maskou) jsou nulové, pak je tento bod označen novou hodnotou.
- 2) v jeho okolí nachází body, které jsou stejně označeny. Tento bod je pak označen stejně.
- 3) v okolí se nachází body, které jsou označeny různými hodnotami. Bod je označen kteroukoliv hodnotou z nich a do tabulky je zapsána ekvivalence barev (~sousedství dvou hodnot, sousední oblasti).

Při druhém průchodu (druhá fáze) jsou označeny stejnou hodnotou oblasti, které jsou uvedeny v tabulce ekvivalence barev. Oblasti se stejnou hodnotou jsou obarveny stejnou barvou. Při tomto průchodu je zároveň spočten počet bodů a součet souřadnic os x a y jednotlivých oblastí. Z těchto hodnot lze vypočítat těžiště jednotlivých oblastí (objektů) dle vztahu:

$$x_i = \frac{m_{10}}{m_{00}}, \quad y_i = \frac{m_{01}}{m_{00}}, \quad (2.51)$$

kde x_i, y_i jsou souřadnice ve směru osy x a y , dále m_{10}, m_{01} je součet jednotlivých souřadnic stejných objektů pro osy x a y , m_{00} udává velikost objektu v bodech (počtu pixelů).

Tyto informace lze použít k dalšímu zpracování a rozpoznávání obrazu.



| |
|---|
| tabulka ekvivalence barev: 3 - 4; 4 - 5 |
|---|

d

Obr. 2.23 Barvení - (a) binární obraz, (b) hodnoty jednotlivých oblastí, (c) obarvený obraz, (d) tabulka ekvivalence barev

2.9.2. Korelace

Touto metodou [19] je hledána shoda mezi strukturním elementem a částí obrazu. Strukturní element je posouván po obraze a 2D korelací je zjišťována shoda. Z důvodu náročnosti a rychlosti výpočtu je obraz i strukturní element transformován do frekvenční oblasti pomocí dvojrozměrné diskretní Fourierovy transformace (2D FFT):

$$F(u, v) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \cdot \exp \left[-2\pi j \left(\frac{mu}{M} + \frac{nv}{N} \right) \right]. \quad (2.52)$$

Ve frekvenční oblasti je proveden výpočet korelace (vynásobení odpovídajících prvků dvou matic, výsledkem je matice o stejných rozměrech):

$$O = F \cdot G^*, \quad (2.53)$$

kde O je výsledek korelace, F je vstupní obraz a G konjugovaný strukturní element.

Poté je provedena zpětná 2D FFT:

$$f(m, n) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} F(u, v) \cdot \exp \left[2\pi j \left(\frac{mu}{M} + \frac{nv}{N} \right) \right]. \quad (2.54)$$

a ve výsledném obraze se označí oblast(i), kde je část obrazu nejvíce shodná s původním vzorem.

3. Aplikace IPRTool

Úkolem této diplomové práce bylo vytvořit program, který pomocí implementovaných metod a algoritmů umožní automatické zpracování a rozpoznávání obrazu.

Program byl vytvořen v programovacím jazyce C [20] v prostředí Microsoft Visual Studio .Net [21] s využitím programovatelného rozhraní Windows, označované jako API (Application Programmable Interface) [22, 23, 24].

Tento program umožňuje:

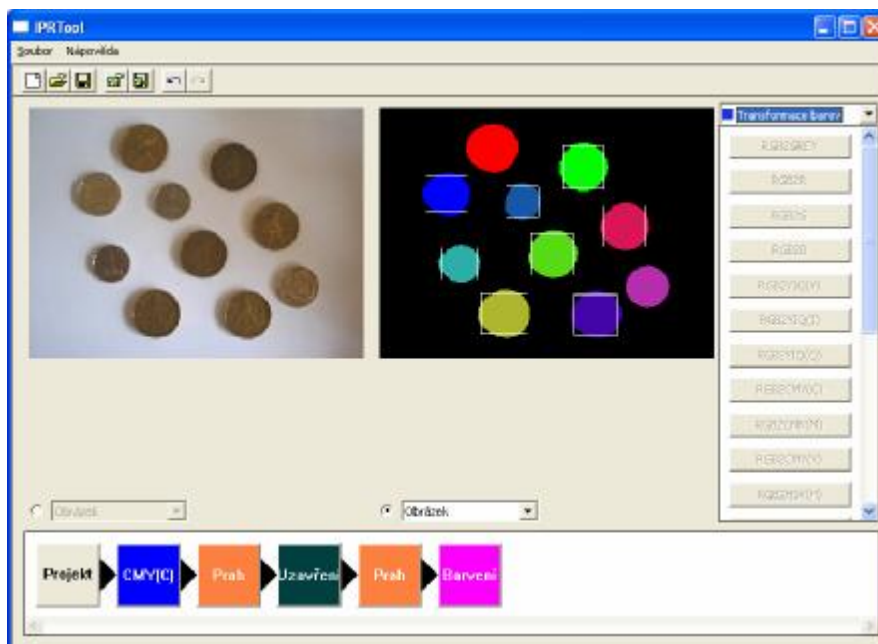
- načíst a zobrazit obrazy uložené v běžných grafických formátech.
- pomocí implementovaných metod a algoritmů zpracovávat a rozpoznávat obraz s okamžitou vizualizací s výběrem typu zobrazení.
- uložení obrazu v jakémkoliv kroku zpracování a rozpoznávání.
- tvorbu a ukládání řetězců pro automatické zpracování a rozpoznávání obrazu.
- přidávání nových metod.

3.1. Popis programu

V následující části budou popsány jednotlivé části programu.

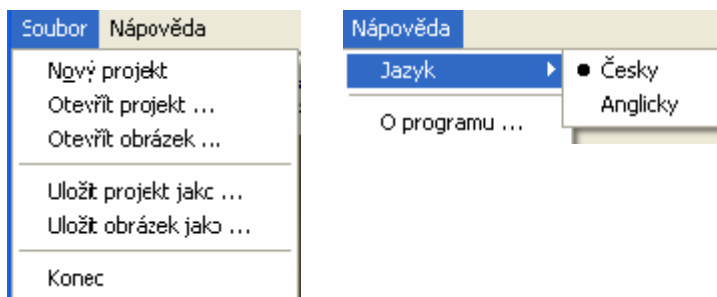
3.1.1. Hlavní okno aplikace

Po spuštění programu je otevřeno hlavní okno aplikace.



Obr. 3.1 Hlavní okno aplikace

Pod titulkem okna programu je umístěno menu, kde jsou přítomny 2 základní nabídky - Soubor a Nápověda, které dále obsahují položky (obr. 3.2)



Obr. 3.2 Menu aplikace

Stručný popis významu jednotlivých položek menu je následující (v závorce je uveden anglický ekvivalent):

- nabídka **Soubor (File)**:

- *Nový projekt (New project)* - jsou vymazány všechny přidané operace v řetězci a v obou oknech je zobrazen výchozí obraz

- *Otevřít projekt... (Open project...)* - je otevřeno dialogové okno s možností výběru uloženého řetězce pro zpracování a rozpoznávání

- *Otevřít obrázek... (Open picture...)* - je zobrazeno dialogové okno pro výběr obrazu, který bude zpracováván

- *Uložit projekt jako ... (Save project as...)* - je zobrazeno dialogové okno pro uložení vytvořeného řetězce s dotazem

- *Uložit obrázek jako...(Save picture as...)* - je vyvoláno dialogového pro uložení zpracovaného obrazu

- *Konec (Exit)* - ukončení aplikace s dotázáním

- nabídka **Nápověda (Help)**:

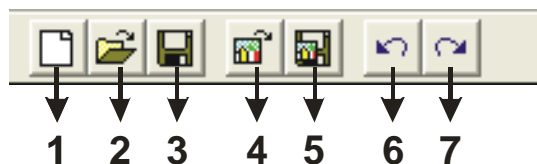
- *Jazyk (Language)* - obsahuje položky pro výběr jazyka aplikace

- *Česky (Czech)* - v aplikaci je nastaveno české prostředí

- *Anglicky (English)* - v aplikaci je nastaveno anglické prostředí

- *O programu... (About...)* - je otevřeno dialogové okno s informacemi o programu.

Z výše uvedených funkcí programu lze některé obsluhovat tlačítky umístěnými pod menu, toolbarem.



Obr. 3.3 Toolbar

Vysvětlení významu a provázanosti jednotlivých tlačítek toolbaru (obr. 3.3) s menu je:

- | | |
|----------------------------|---------------------------------|
| 1 - Nový projekt | 5 - Uložit obrázek jako... |
| 2 - Otevřít projekt ... | 6 - Zpět - vysvětleno později |
| 3 - Uložit projekt jako... | 7 - Znovu - vysvětleno později. |
| 4 - Otevřít obrázek... | |

Z obr. 3.1 je zřejmé, že plochu aplikace se lze rozdělit na tři důležité oblasti.

První z nich je oblast, kde jsou umístěna dvě okna, ve kterých je zobrazen zpracováváný obraz. V této aplikaci nelze zpracovávat více obrazů najednou, ale dvě okna jsou zde přítomna z důvodu vizualizace, aby bylo možné porovnávat obraz v různých fázích zpracování a rozpoznávání, například v jednom okně je zobrazen obraz před transformací (operací) a v druhém po transformaci. V případě, že je provedena transformace s obrazem, je její výsledek zobrazen do okna, které je označeno jako aktivní. Výběr aktivního okna je proveden pomocí přepínačů, které jsou umístěny pod každým oknem. Vždy může být aktivní pouze jedno okno, tzn. obsah druhého okna je v případě transformace nezměněn.

Druhá oblast je v pravé části aplikace. Zde jsou umístěny dialogová okna s jednotlivými implementovanými metodami a algoritmy, které jsou rozděleny do skupin dle charakteru operace. Výběr skupiny operací je prováděn pomocí comboboxu, který je umístěn nad oknem s operacemi. Vždy je zobrazena pouze jedna skupina operací.

V dolní části okna aplikace je třetí důležitá oblast, ve které jsou zobrazeny jednotlivé přidané (provedené) operace s obrazem v pořadí v jakém byly vybírány. Přidané operace jsou ve formě tlačítek s popiskem a barevným rozlišením. Při kliknutí na jakékoliv přidané tlačítko je zobrazen do příslušného okna výsledek této operace.

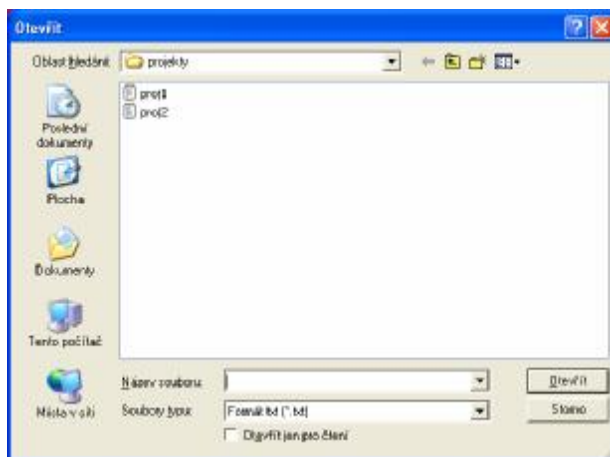
3.1.1.1. Nový projekt (New project)

Po výběru této položky v menu či toolbaru (obr. 3.3 - 1) jsou všechny operace, které tvořily řetězec odebrány. Veškeré dílčí výsledky jsou z paměti smazány a v obou oknech je zobrazen výchozí obraz. Tato operace je nevratná.

3.1.1.2. Otevřít projekt ... (Open project ...)

Projekt označuje řetězec pro zpracování a rozpoznávání obrazu, je uložen v textovém souboru *txt* a lze jej otevřít pomocí dialogového okna. V případě, že je projekt vybrán, stiskem *OK* je automaticky aplikován na otevřený obraz. Výsledek je

zobrazen do aktivního okna, dílčí výsledky zpracování a rozpoznávání jsou uloženy v paměti a je možné je zobrazit. Pokud je stisknuto *Storno*, dialog je uzavřen a projekt není otevřen. Aby bylo možné otevřít projekt, musí být otevřen obraz. Jestliže tato podmínka není splněna, je položka pro otevření projektu nedostupná.



Obr. 3.4 Ukázka dialogu pro otevření (uložení)

3.1.1.3. Uložit projekt jako ... (Save project as ...)

Prostředkem pro uložení projektu je dialogové okno. Jméno projektu je dáno vyplněním kolonky *Název souboru*. V případě, že již byl vytvořen projekt se stejným jménem, je zobrazeno okno a dotazem, zda se má být stávající projekt přepsán novým. Pokud *Ano*, je původní projekt přepsán novým, v opačném případě je umožněno zadat jiné jméno. Jestliže je stisknuto tlačítko *Storno* v dialogovém okně uložení, je okno zavřeno bez uložení projektu.

Ukázka projektu uloženého v souboru:

```

3 //počet operací v projektu
4 //jedinečné identifikační číslo operace
0 0 0 0 0 0 0 0 0 //parametry operace
no text //textový parametr operace
70
0 127 0 0 0 0 0 0 0
no text
101
1 0 0 0 0 0 0 0 0
C:\Diplomka\Dipl\se1.bmp

```

3.1.1.4. Otevřít obrázek ... (Open picture ...)

Otevřít obraz lze pomocí dialogového okna, které je vyvoláno pomocí menu nebo toolbaru (obr. 3.3 - 5). Podporované formáty obrazů jsou *bmp*, *jpg* a *gif*. Formát *bmp* je bezztrátový a nekomprimovaný, *jpg* ztrátový a komprimovaný a *gif*

komprimovaný a bezztrátový. Pro tyto formáty jsou v dialogovém okně vytvořeny filtry (položka *Soubor typu*), tak aby bylo možné vybírat z požadovaného formátu. Po výběru obrazu ve zvoleném umístění (adresáři) a stisku tlačítka *OK*, je obraz nejdříve převeden na soubor typu *bmp* se 24-bitovou barevnou hloubkou (8 bitů pro každou pro barvu *R,G,B*) a zapsán do pomocného souboru na disk. Následně je tento pomocný soubor otevřen do obou oken bez ohledu na to, které je označeno jako aktivní. Dále je provedeno načtení obrazu do paměti, tak aby mohl být dále zpracováván a rozpoznáván. Pro tento účel byla vytvořena datová struktura *im24*, ve které jsou uloženy informace o obraze a jednotlivá obrazová data:

```
im24:  int s[4]      - souřadnice levého dol. a pravého hor. rohu obrazu
      int r[2]      - velikost obrazu
      sim24 **d     - pole dat, jednotlivé pixely s hodnotami barev
sim24: unsigned char c[3] - vektor, jednotlivé barvy (R,G,B) pro daný pixel.
```

Načtení obrazu lze rozdělit do dvou fází. Nejprve je přečtena hlavička (standardně 54 bytů) pomocného souboru *bmp*, z které jsou zjištěny rozměry obrazu. Podle těchto informací je naalokována potřebná paměť pro data (barvy jednotlivých pixelů).

Hlavička formátu *bmp* je:

| počet B | význam |
|---------|--|
| 2 | identifikátor BMP souboru, vždy "BM" |
| 4 | celková délka souboru v bytech s hlavičkou |
| 4 | vždy 0 |
| 4 | začátek dat, posun od začátku |
| | druhá část hlavičky |
| 4 | velikost hlavičky (druhé části), vždy 40 |
| 4 | šířka obrazu v bodech |
| 4 | výška obrazu v bodech |
| 2 | počet bitových rovin |
| 2 | počet bodů na pixel - 1, 4, 8, 24 |
| 4 | typ komprese, 0 - žádná, 1 - 8bit. RLE, 2 - 4bit. RLE |
| 4 | velikost obrazu (bez hlavičky) |
| 8 | horizontální a vertikální rozlišení - počet pixelu/metr |
| 4 | počet aktuálních barev v paletě, 0 - všechny barvy z tabulky |
| 4 | počet důležitých barev pro zobrazení, 0 - všechny barvy důležité |

Tab 3.1 Hlavička souboru typu *bmp*

Ve druhé fázi jsou načteny obrazová data ze souboru. Data jsou v souboru uložena od levého dolního rohu obrazu po řádcích směrem nahoru, tzn. že na konci souboru je uložen bod, který je v obraze vpravo nahoře. Řádky (šířka) obrazu jsou v souboru uloženy tak, že jejich délka je dělitelná čtyřmi beze zbytku, tzn. že znalosti šířky obrazu je určeno kolik bytů je v každém řádku je navíc. Tento fakt je potřeba při vyčítání dat ze souboru ošetřit. Příklad viz kapitola 3.1.1.5.

V případě stisku tlačítka *Storno* je dialog uzavřen bez otevření obrazu.

3.1.1.5. Uložit obrázek jako ... (Save picture as ...)

Výběrem příslušné položky v menu nebo toolbaru (obr. 3.3. - 5) je zobrazeno dialogové okno, pomocí kterého je umožněno uložení obrazu zobrazeného v okně, jenž je označeno jako aktivní. V dialogovém okně je zvolen adresář, do kterého bude soubor uložen. Do kolonky *Název souboru* se zadává požadované jméno souboru (možno s příponou i bez). Po stisku tlačítka *OK* je obraz uložen. V případě, že ve zvoleném umístění je soubor se stejným jménem, pak je zobrazeno okno s dotazem, zda má být tento soubor přepsán novým. Pokud *Ano*, je tento soubor přepsán novým, v opačném případě lze zadat nové jméno ukládaného souboru. Při zápisu do souboru je nejdříve zapsána hlavička s potřebnými údaji a pak obrazová data od levého dolního rohu po řádcích směrem k pravému hornímu rohu obrazu s ošetřením šířky řádku na dělitelnost čtyřmi. Obraz je uložen ve formátu 24-bit *bmp*.

Př.: šířka obrazu je 105 bodů, každý bod je reprezentován 3 barvami

--> $3 * 105 = 315$ --> nejbližší vyšší dělitelné 4 je 316 --> řádek se doplní na délku 316 nulami

Při stisku tlačítka *Storno* je dialog opuštěn bez uložení příslušného obrazu.

3.1.1.6. Konec (Exit)

Ukončení aplikace s dotázáním. Stiskem tlačítka *Ano*, je aplikace ukončena, v opačném případě lze pokračovat v práci.

3.1.1.7. Zpět (Undo)

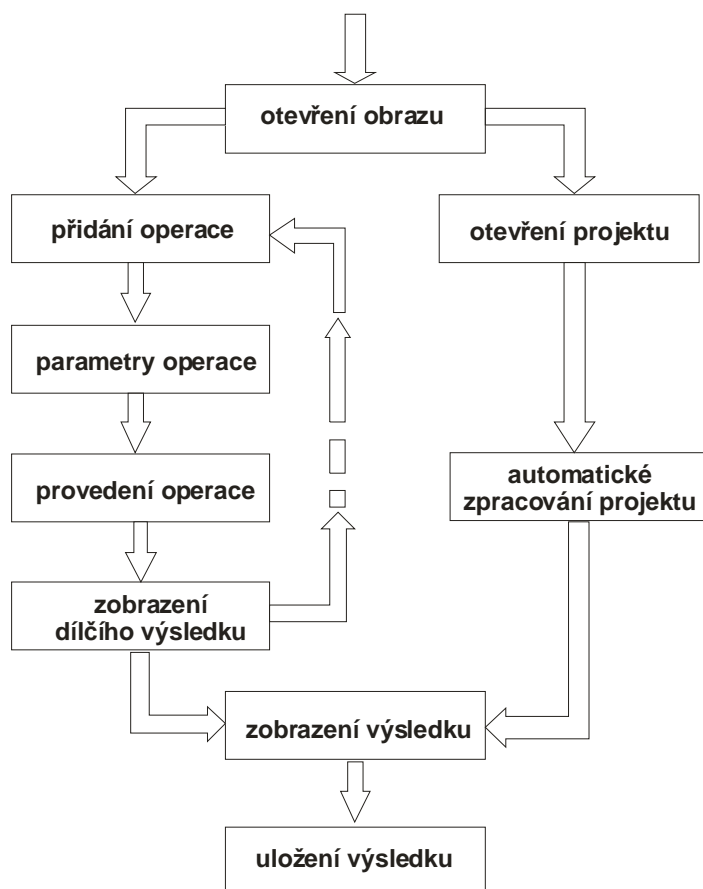
Funkcí *Zpět*, kterou lze vybrat pouze pomocí toolbaru (obr. 3.3 - 6), kde je symbolizovaná modrou zpětnou šipkou, je umožněno odebírat z vytvářeného nebo otevřeného rozpoznávacího řetězce jednotlivé operace (moduly). Operace nelze mazat libovolně, ale pouze odzadu, tzn. od poslední přidané. Počet kroků zpět je dán počtem operací v řetězci. V případě, že jsou smazány všechny, je zobrazeno okno s hlášením, že všechny operace byly odebrány. V aktivním okně je zobrazen výsledek poslední operace v řetězci.

3.1.1.8. Znovu (Redo)

Touto funkcí, která je umístěna pouze toolbaru (obr. 3.3 - 7), kde je symbolizována modrou šipkou se směrem dopředu, lze vrátit zpět poslední krok. Posledním krokem je rozuměna poslední smazaná operace z řetězce (pomocí *Zpět*), nikoliv otevření a smazání celého projektu nebo otevření nového obrazu. Tato funkce je jednokroková, tzn. že lze zpětně vrátit pouze poslední odebranou funkci z řetězce.

3.2. Struktura programu

Strukturu systému nejlépe popisuje následující obrázek.



Obr. 3.5 Struktura programu

Po otevření obrazu jsou dvě možnosti, jak lze s programem dále pracovat:

- 1) tvorba vlastního rozpoznávacího řetězce (obr. 3.5 - levá část)
- 2) otevření uloženého projektu a jeho automatické vykonání na obraze (obr. 3.5 - pravá část).

3.2.1. Tvorba vlastního rozpoznávacího řetězce

Postupným vybíráním operací z dialogových oken umístěných v pravé části aplikace lze tvořit nový řetězec nebo pokračovat v otevřeném projektu. Přidání operace do projektu lze shrnout do tří kroků:

- 1) Výběr záložky s požadovaným typem operací pomocí comboboxu umístěným nad záložkami a dále výběr požadované transformace z této záložky, který je dán kliknutím myši na příslušné tlačítko s jejím názvem.

2) V případě, že je u transformace vyžadováno nastavení pro ni typických parametrů, je zobrazeno příslušné dialogové okno, pomocí kterého je lze nastavit. Po stisku tlačítka *Proved'* je provedena požadovaná operace a zobrazena do aktivního zobrazovacího okna. Nastavovací dialog zůstane otevřený. Jestliže je dosaženo požadovaného výsledku transformace, je kliknutím na tlačítko *OK* dialog zavřen a operace je přidána na konec řetězce (viz bod 3). Pokud není dosaženo předpokládaného výsledku, lze parametry nastavit znova a opětovným stiskem *Proved'* zobrazit výsledek. Tuto fázi lze opakovat do té doby, než je dosaženo předpokládaného výsledku. Jestliže ani po provedení několika pokusů vybrané transformace s různými parametry není dosaženo požadovaného cíle, lze opustit dialog stisknutím tlačítka *Storno*. Tím je výběr operace zrušen, v aktivním okně je obnoveno předchozí zobrazení a operace není přidána do řetězce.

Pokud transformace nevyžaduje nastavení parametrů, je automaticky vykonána a její výsledek je zobrazen v příslušném aktivním okně pro vizualizaci.

3) Jestliže je transformace úspěšně vykonána, je její přidání do řetězce symbolizováno v dolní části aplikace, kde je okno s již provedenými operacemi, tlačítkem, které je automaticky vygenerováno, popsáno a barevně rozlišeno podle toho, do jaké skupiny implementovaných metod patří. Jednotlivě přidávané operace jsou chronologicky řazeny za sebou a mezi nimi je zobrazena šipka symbolizující sled operací.

Bod 1) a část bodu 2) (nastavení parametru operace) je závislý jen na uživateli, zbývající body jsou vykonány systémem automaticky bez možnosti zásahu uživatele.

Tento uvedený postup lze provádět tak dlouho, dokud není dosaženo žádaného výsledku zpracování a rozpoznávání obrazu. V řetězci není dovoleno jednotlivé operace zaměňovat a měnit jejich pořadí. Informace o přidávaných operacích jsou uchovány ve vytvořené struktuře:

```
PARAMETERS: int operacel      - index operace, každá operace má svoje unikátní číslo,
                               viz ukázka souboru
               int param[10]   - číselné parametry operací
               char text[_MAX_PATH] - textový parametr operace.
```

3.2.2. Otevření vytvořeného projektu

Projekt, který lze otevřít pomocí dialogového okna (kapitola 3.1.1.2.), je automaticky aplikován na otevřený obraz a v aktivním vizualizačním okně je zobrazen výsledek, dílčí výsledky zpracování a rozpoznávání jsou uloženy v paměti. V případě, že je v průběhu vytváření nového řetězce otevřen uložený projekt, je vytvářený řetězec smazán a nahrazen otevřeným projektem. Tato operace je nevratná. Otevřený projekt je možné modifikovat, tzn. odebírat postupně jednotlivé operace či přidávat nové.

V průběhu vytváření nového řetězce nebo jeho modifikace lze jej uložit jako projekt (kapitola 3.1.1.3.). Řetězec je uložen vždy celý, tzn. že nelze z něho uložit jenom jednotlivé operace nebo sekvence operací.

3.3. Vizualizace výsledků

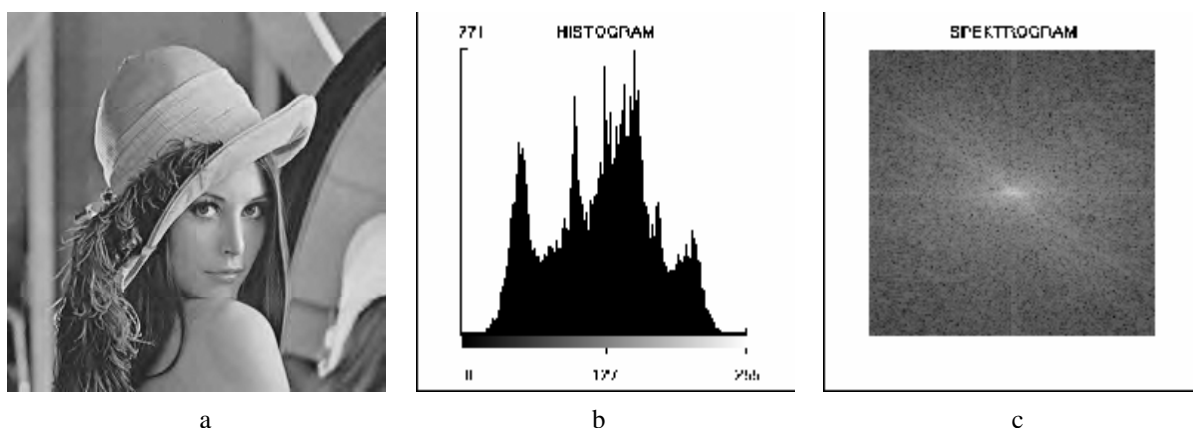
V průběhu zpracování a rozpoznávání obrazu jsou zobrazovány průběžné výsledky do okna, které je nastaveno jako aktivní, což je dáno přepínači pod vizualizačními okny. Pokud je označeno pravé okno za aktivní, levé se automaticky stává neaktivním a naopak. Obě zobrazovací okna jsou stejně velká a jejich velikost je dána velikostí okna aplikace (proměnná) a velikostmi oken s implementovanými metodami a přidávanými operacemi do projektu (pevné velikosti).

V aplikaci jsou dvě okna pro zobrazování především z důvodu možnosti porovnání obrazu v různých fázích zpracování. Protože jsou dílčí výsledky zpracování uloženy v paměti, dokud nedojde k odebrání dané operace z řetězce nebo smazání řetězce, otevření nového obrazu nebo ukončení aplikace, lze je kdykoliv vyvolat a zobrazit do požadovaného okna.

Zobrazení požadované operace je dáno kliknutím myši na příslušné tlačítko v dialogu přidávaných operací (dole v okně aplikace). Program zjistí, které tlačítko bylo stisknuto a do zvoleného okna zobrazí odpovídající výsledek z dané operace. Toto zobrazení je velmi rychlé, protože jednotlivé dílčí výsledky jsou uloženy v paměti, takže je výsledek pouze zkopírován do zvoleného okna.

Výstupem z každé transformace a operace s obrazem jsou obrazová data, která lze interpretovat a zobrazovat třemi způsoby:

- 1) Obraz (Picture) - standardní zobrazení (obr. 3.6a)
- 2) Histogram (Histogram) - z obrazu je vytvořen histogram (~ sloupcový graf, kapitola 2.3.1., obr. 3.6b)
- 3) Spektrogram (Spectrogram) - obraz je převeden pomocí dvojrozměrné Fourierovy transformace dle vztahu (2.52) do frekvenční oblasti a zobrazen jako frekvenční spektrum (obr. 3.6c). Obraz je chápán jako jedna 2D perioda periodické funkce.



Obr. 3.6 (a) vstupní obraz, (b) histogram, (c) spektrogram

Výběr způsobu zobrazení obrazu se provádí pomocí comboboxu, který je umístěn pod každým zobrazovacím oknem zvlášť. Pokud je vybrán histogram nebo spektrogram, je toto zobrazení vytvořeno z obrazu v daném okně. Histogram ani spektrogram nemají vliv na jednotlivé transformace a nijak nemění obrazová data (obraz je nadále uložen v paměti nezměněn).

3.4. Implementované metody

V systému jsou implementovány jednotlivé metody a algoritmy pro zpracování a rozpoznávání obrazu, které jsou dále rozděleny do devíti základních skupin podle charakteru operace. Jednotlivé skupiny (~záložky) jsou pro větší přehlednost barevně odlišeny. Pro jednotlivé metody v řetězci jsou vygenerována tlačítka s barvou skupiny, ze které bylo tlačítko vybráno. Jednotlivé skupiny mají tyto barvy:

- | | |
|--------------------------------------|---------------|
| 1) Transformace barev | - modrá |
| 2) Transformace hodnot jasu | - červená |
| 3) Geometrické transformace | - zelená |
| 4) Filtrace šumu | - hnědá |
| 5) Detekce hran | - zelenohnědá |
| 6) Segmentace | - oranžová |
| 7) Binární matematická morfologie | - tmavě hnědá |
| 8) Šedotónová matematická morfologie | - šedá |
| 9) Rozpoznávání objektů | - fialová. |

IPRTool

Transformace barev

| | | | |
|-------|------------|------------|---------------|
| RGB2Y | RGB2YIQ(Y) | RGB2HSV(H) | RGB2HSI(S) |
| RGB2R | RGB2YIQ(I) | RGB2HSV(S) | RGB2HSI(I) |
| RGB2G | RGB2YIQ(Q) | RGB2HSV(V) | RGB2YCbCr(Cb) |
| RGB2B | RGB2CMY(C) | RGB2HSL(S) | RGB2YCbCr(Cr) |
| | RGB2CMY(M) | RGB2HSL(L) | RGB2LT |
| | RGB2CMY(Y) | | |

Transformace hodnot jasu

| | | | |
|-----------------------|--------------|-----|----------|
| Negativ | Korekce jasu | Jas | Kontrast |
| Ekvalizace histogramu | | | |

Geometrické transformace

| | | | |
|---------------|---------|--------|--------------|
| Změna měřítka | Zkosení | Rotace | H-překlopení |
| V-překlopení | | | |

Filtrace šumu

| | | |
|--------------------|----------------|--------|
| Prosté průměrování | Rotující maska | Medián |
|--------------------|----------------|--------|

Hledání hran

| | | | |
|---------------------|--------------------|---------------------|------------------|
| Robertsův operátor | Laplaceův operátor | Operátor Prewittové | Sobelův operátor |
| Robinsonův operátor | Kirschův operátor | LoG operátor | |

Segmentace

| |
|-----------|
| Prahování |
|-----------|

Binární matematická morfologie

| | | | |
|-------------|-------|----------|----------|
| Dilatace | Eroze | Otevření | Uzavření |
| Tref či miň | | | |

Šedotónová matematická morfologie

| | | | |
|----------|-------|----------|----------|
| Dilatace | Eroze | Otevření | Uzavření |
|----------|-------|----------|----------|

Rozpoznávání objektů

| | |
|---------|----------|
| Barvení | Korelace |
|---------|----------|

Obr. 3.7 Implementované metody a jejich rozdělení do skupin

3.4.1. Záložka Transformace barev (Color transformations)

Protože tento systém umožňuje zpracování a rozpoznávání obrazu v šedotónové stupnici, jsou na této záložce jsou umístěny transformace, pomocí kterých je obraz převeden z barevného prostoru na šedotónový. Transformace barev je první operace s obrazem, kterou je nutno vždy provést, jinak jsou všechny další operace zablokovány. Pro převod do šedotónové stupnice je zde implementováno celkem 20 metod. Ve většině případů se jedná o transformace na barevné modely, ze kterých je pak vybrána pouze jedna složka.

Veškeré transformace barev uvedené v této kapitole jsou provedeny z barevného obrázku *lena.jpg* (www.gridflow.ca/latest/images/).

- **RGB2GREY** - převod RGB na šedotónový model Y dle vztahu (2.1)



Obr. 3.8 Převod RGB na Y

- **RGB2R** - z RGB modelu je použita pouze složka R (červená)
- **RGB2G** - z RGB modelu je použita pouze složka G (zelená)
- **RGB2B** - z RGB modelu je použita pouze složka B (modrá)



Obr. 3.9 (a) převod RGB na R , (b) RGB na G , (c) RGB na B

- **RGB2YIQ(Y)** - převod RGB na Y modelu YIQ dle vztahu (2.4)
- **RGB2YIQ(I)** - převod RGB na I modelu YIQ dle vztahu (2.4)
- **RGB2YIQ(Q)** - převod RGB na Q modelu YIQ dle vztahu (2.4)



Obr. 3.10 Převod *RGB* na jednotlivé složky (a) *Y*, (b) *I*, (c) *Q* z modelu *YIQ*

- **RGB2CMY(C)** - převod z *RGB* na *C* modelu *CMY* dle vztahu (2.2)
- **RGB2CMY(M)** - převod z *RGB* na *M* modelu *CMY* dle vztahu (2.2)
- **RGB2CMY(Y)** - převod z *RGB* na *Y* modelu *CMY* dle vztahu (2.2)



Obr. 3.11 Převod *RGB* na jednotlivé složky (a) *C*, (b) *M*, (c) *Y* z modelu *CMY*

- **RGB2HSV(H)** - převod z *RGB* na *H* modelu *HSV* dle vztahu (2.5)
- **RGB2HSV(S)** - převod z *RGB* na *S* modelu *HSV* dle vztahu (2.5)
- **RGB2HSV(V)** - převod z *RGB* na *V* modelu *HSV* dle vztahu (2.5)



Obr. 3.12 Převod *RGB* na jednotlivé složky (a) *H*, (b) *S*, (c) *V* z modelu *HSV*

- **RGB2HSL(S)** - převod z *RGB* na *S* modelu *HSL* dle vztahu (2.7)
- **RGB2HSL(L)** - převod z *RGB* na *L* modelu *HSL* dle vztahu (2.7)



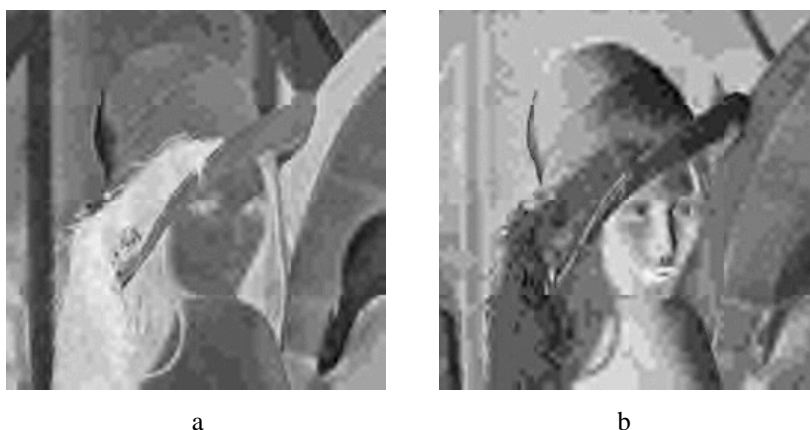
Obr. 3.13 Převod *RGB* na jednotlivé složky (a) *S*, (b) *L* z modelu *HSL*

- **RGB2HSI(S)** - převod z *RGB* na *S* modelu *HSI* dle vztahu (2.6)
- **RGB2HSI(I)** - převod z *RGB* na *I* modelu *HSI* dle vztahu (2.6)



Obr. 3.14 Převod *RGB* na jednotlivé složky (a) *S*, (b) *I* z modelu *HSI*

- **RGB2YCbCr(Cb)** - převod *RGB* na *Cb* modelu *YCbCr* dle vztahu (2.3)
- **RGB2YCbCr(Cr)** - převod *RGB* na *Cr* modelu *YCbCr* dle vztahu (2.3)

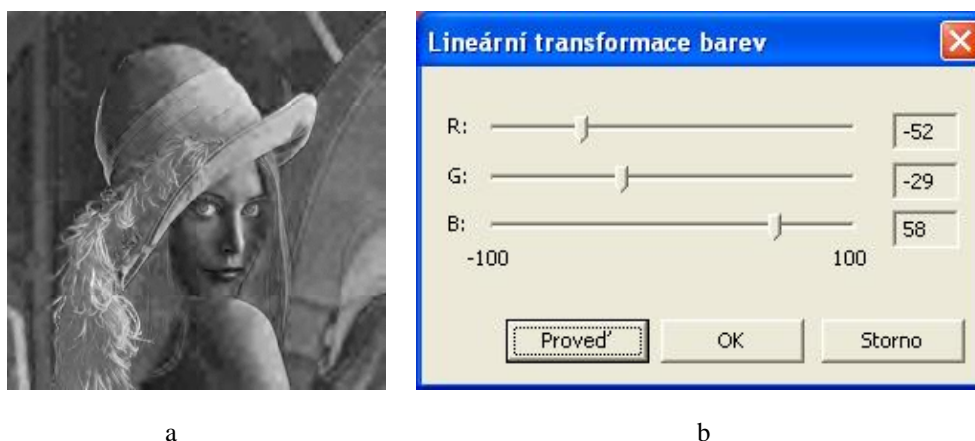


Obr. 3.15 Převod *RGB* na jednotlivé složky (a) *Cb*, (b) *Cr* z modelu *YCbCr*

- **RGB2LT** - převod z *RGB* na šedotónový obraz pomocí lineární transformace. Pro každou z barev je nastaven váhový koeficient a výsledná barva je dána vztahem:

$$O = k_R \cdot R + k_G \cdot G + k_B \cdot B, \quad (3.1)$$

kde O je výstupní jas, R, G, B jsou hodnoty jednotlivých barev a k_R, k_G, k_B jsou koeficienty převodu, které mohou nabývat hodnot $(-100, 100)$. Parametry transformace barev jsou jednotlivé koeficienty převodu (3 parametry).



Obr. 3.16 (a) převod *RGB* pomocí lineární transformace barev, (b) dialogové okno LT

Po provedené transformaci barev nemusí být výsledná hodnota jasu v intervalu $(0, 255)$ (viz modely *HSV, HSL, LT*), proto je po transformaci ještě proveden převod na interval $(0, 255)$, tak aby bylo možné obraz zobrazit.

Převodní funkce je:

$$g = \frac{f - f(\min)}{f(\max) - f(\min)} \cdot 255, \quad (3.2)$$

kde g je výstupní obraz převedený na interval $(0, 255)$, f je vstupní obraz.

3.4.2. Záložka Transformace hodnot jasu (Pixel brightness transformations)

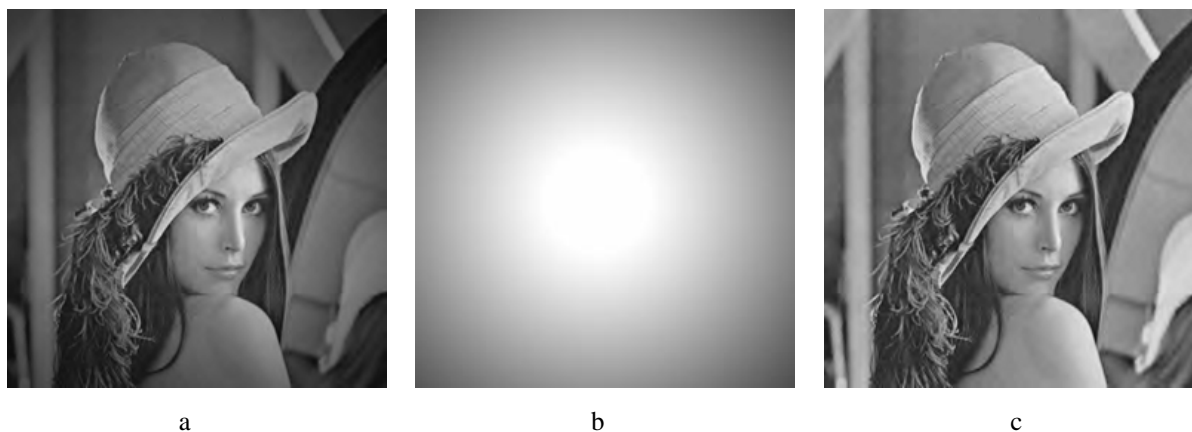
Na této záložce jsou umístěny operace, pomocí kterých je transformována šedotónová stupnice obrazu. Velmi často se u těchto transformací používají tzv. převodní tabulky (look-up table). Podle zvolené transformace je vypočtena převodní tabulka pro jednotlivé jasy, tj. ze vstupní jasové stupnice je určena výstupní jasová stupnice dle transformačního vztahu. Poté je v každém bodě obrazu nahrazen jas odpovídajícím jaselem z převodní tabulky.

- **Negativ (Negativ)** - transformace je popsána v kapitole 2.3.3.1., výpočet je dán vztahem (2.11).



Obr. 3.17 Negativ, (a) vstupní obraz, (b) negativ obrazu

- **Korekce jasu (Brightness correction)** - popis je uveden v kapitole 2.3.2. včetně výpočetního vztahu (2.9). Při použití této operace je otevřeno dialogové okno (obr. 3.18d), ve kterém se nastavuje korekční koeficient v intervalu (0, 255) a etalonový obraz. Ten je načten pomocí dialogového okna, které je zobrazeno po kliknutí na tlačítko s popiskem '...'. Etalonový obraz musí mít stejné rozměry jako zpracováváný obraz. V případě, že tomu tak není, operace korekce jasu není provedena. Parametry této operace jsou korekční koeficient a jméno souboru s etalonovým obrazem.



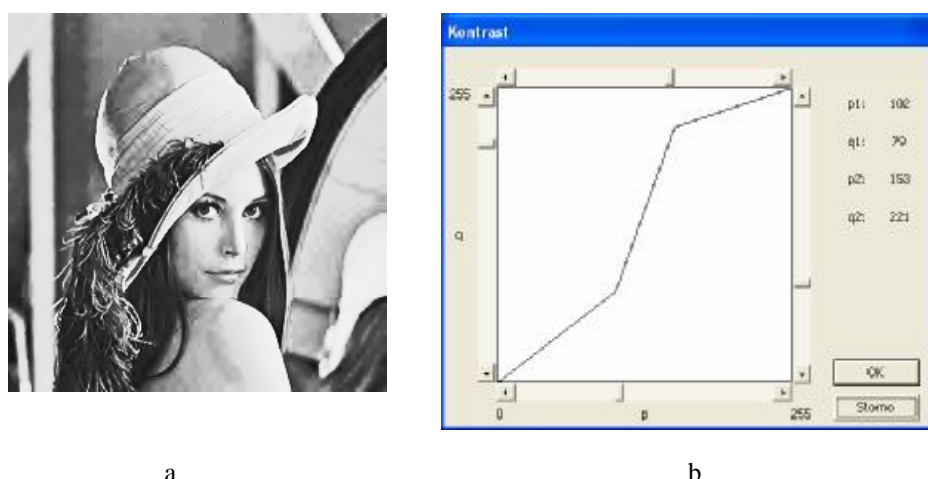
Obr. 3.18 Korekce jasu - (a) vstupní obraz, (b) etalonový obraz, (c) korekce jasu, (d) dialogové okno Korekce jasu

- **Jas (Brightness)** - operace je uvedena v kapitole 2.3.3.3. s převodním vztahem (2.13). Parametr této operace lze nastavit pomocí posuvníku v příslušném dialogovém okně (obr. 3.19c) v intervalu (-255, 255). Po stisku tlačítka *Proved'* je provedena operace s nastaveným parametrem a výsledek je zobrazen do příslušného aktivního okna.



Obr. 3.19 Změna jasu obrazu, vstupní obraz obr. 3.8 - (a) jas = -74, (b) jas = 108, (c) dialogové okno Jas

- **Kontrast (Contrast)** - operace je popsána v kapitole 2.3.3.2. Při výběru funkce kontrast je otevřeno dialogové okno (obr. 3.20b), ve kterém lze nastavit požadovanou převodní lomenou funkci tak, aby byl mezi dvěma jasy zvětšen nebo zmenšen kontrast (rozestup mezi jednotlivými hodnotami jasu na daném intervalu). Pokud je zvětšen, jsou krajní intervaly zmenšeny z čehož vyplývá, že světlé a tmavé části obrazu více splývají a jasy mezi body p_1 a p_2 jsou roztaženy na větší interval. Při zmenšení intervalu je tomu naopak. Převodní funkce je nastavována pomocí posuvníků umístěných okolo okna, ve kterém je zobrazena. Horní a dolní posuvníky rozdělují vstupní jasovou stupnici na tři intervaly. Levý a pravý posuvník určuje jak velký interval jim bude vymezen ve výstupní (převodní) jasové stupnici. Při změně polohy kteréhokoliv z posuvníků je automaticky překreslena převodní funkce a v okně, které je aktivní, je transformován obraz dle této funkce. V případě stisku *OK* je přidána operace do řetězce a jsou uloženy její čtyři parametry p_1 , p_2 , q_1 a q_2 .



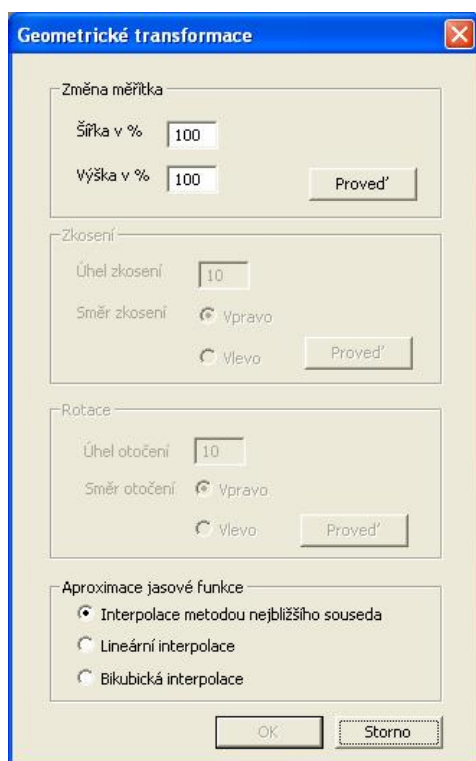
Obr. 3.20 Kontrast, vstupní obraz obr. 3.8 - (a) změna kontrastu, (b) dialogové okno Kontrast

- **Ekvalizace histogramu (Histogram equalization)** - tato transformace je popsána v kapitole 2.3.3.4. Ekvalizace je provedena dle vztahu (2.15). Tato operace je bez parametrů.



Obr. 3.21 Ekvalizace histogramu, (a) vstupní obraz, (b) výstupní vpravo

3.4.3. Záložka Geometrické transformace (Geometric transformations)



Obr. 3.22 Společné dialogové okno transformace Změna měřítka, Zkosení a Rotace

Jak je uvedeno v kapitole 2.4., jsou geometrické transformace rozděleny do dvou kroků. Za prvé je to transformace souřadnic bodů (kapitola 2.4.1.) a za druhé aproximace jasové hodnoty (kapitola 2.4.2.). Implementované metody používají společné dialogové okno, ve kterém jsou dostupné jen ty položky, které odpovídají zvolené geometrické transformaci, ostatní jsou zakázány. Pro aproximaci jasové funkce jsou přítomny tři typy interpolací:

- 1) Interpolace metodou nejbližšího souseda
- 2) Lineární interpolace
- 3) Bikubická interpolace.

- **Změna měřítka (Change of scale)** - popis této transformace je proveden v kapitole 2.4.1.1., transformační rovnice (2.18). U této transformace se nastavují tři parametry v příslušných boxech. Lze měnit šířku (%) a výšku (%) nového obrazu. Pro oba parametry lze zadávat hodnoty v intervalu (50, 500)% (50%~zmenšení 2x, 500%~zvětšení 5x). V případě, že jsou nastaveny hodnoty mimo interval, program na toto upozorní a operaci není provedena. Třetím parametrem je výběr interpolační metody pro aproximaci jasové hodnoty.
- **Zkosení (Skewing)** - transformační vztah (2.19) je uveden v kapitole 2.4.1.2. Zkosení obrazu lze ovlivnit třemi parametry. Prvním je úhel zkosení, který lze nastavit v rozmezí (1, 45)°, dále směr zkosení (vlevo, vpravo) a nakonec interpolační metodu pro aproximaci jasové hodnoty.



Obr. 3.23 Zkosení, vstupní obraz obr. 3.8 - (a) interpolace metodou nejmenšího souseda, (b) bikubická interpolace

- **Rotace (Rotation)** - popis je uveden v kapitole 2.4.1.3. včetně transformačního vztahu (2.20). Také i zde se nastavují tři parametry. Prvním je úhel rotace v rozmezí (1, 360)°, druhým je smysl rotace (vlevo, vpravo) a třetím je metoda interpolace jasové hodnoty.



Obr. 3.24 Rotace, vstupní obraz obr. 3.8 - (a) interpolace metodou nejmenšího souseda, (b) lineární interpolace

- **H-překlopení (H-flip)** - obr. 3.25b
- **V-překlopení (V-flip)** - obr. 3.25c

Tyto dvě transformace nemají parametry. Dochází k přesunu jednotlivých pixelů v obraze v závislosti na tom, jestli bylo vybráno horizontální nebo vertikální překlopení (zrcadlení).



Obr. 3.25 (a) vstupní obraz, (b) H-překlopení, (c) V-překlopení

3.4.4. Záložka Filtrace šumu (Image smoothing)

Těmito operacemi lze z obrazu odstranit šum, kterým mohou být jednotlivé pixely obrazu zatíženy, a následně obraz vyhladit. K určení nové hodnoty pixelu je použito jeho blízké okolí.

- **Prostý průměr (Simple averaging)** - popis této metody je v kapitole 2.5.1.1. Při použití této metody je zobrazeno dialogové okno (obr. 3.26c), ve kterém lze zvolit velikost okolí pro filtraci - 3x3, 5x5, 7x7. Při výběru okolí 3x3 lze ještě nastavit váhy jednotlivých bodů masky. U masek 5x5 a 7x7 má každý bod masky váhu 1. Počet parametrů je závislý na zvoleném typu masky. Jeden parametr určuje typ masky, a v případě masky 3x3 jsou ještě uloženy jednotlivé váhy.



Obr. 3.26 Prostý průměr, vstupní obraz obr. 3.8 - (a) filtrace s maskou 3x3 s největší vahou uprostřed, (b) maska 7x7, (c) dialogové okno Prostý průměr

- **Rotující maska (Rotating mask)** - tato metoda filtrace šumu je popsána v kapitole 2.5.2.1. V příslušném dialogovém okně (obr. 3.27c) lze vybrat jednu ze tří masek - 3x3, 5x5, 7x7. Parametrem této funkce je vybraný typ (velikost) masky.



Obr. 3.27 Rotující maska - (a) vstupní obraz, (b) filtrace maskou 3x3, (c) dialogové okno Rotující maska

- **Medián (Median)** - filtrace mediánem je uvedena v kapitole 2.5.2.2. U této operace lze vybrat v příslušném dialogovém okně (obr 3.28c) jednu ze šesti masek. Použití lze masky 3x3, 5x5, 7x7 nebo 1 ze 3 masek zachovávající hrany. Ukládaným parametrem je typ masky.



Obr. 3.28 Medián, vstupní obraz obr. 3.8 - (a) filtrace maskou 7x7, (b) filtrace pomocí vertikální masky, (c) dialogové okno Medián

3.4.5. Záložka Hledání hran (Edge detectors)

Problematice hledání (detekce) hran je věnována kapitola 2.6.

- **Robertsův operátor (Roberts operator)** - tento operátor je popsán v kapitole 2.6.1.1. a výpočet je dle vztahu (2.31) s maskou (2.30). Při výběru tohoto detektoru je zobrazeno dialogové okno (obr. 3.29c), ve kterém lze nastavit, zda operátor je

použit k ostření obrazu. Pokud ano, je zatržena položka Ostření a následně je nastaven koeficient ostření. Parametrem této funkce je pak tento koeficient.



Obr. 3.29 Robertsův operátor, vstupní obraz obr. 3.8 - (a) detekce hran, (b) ostření s koeficientem 0.8, (c) dialogové okno Robertsův operátor

- **Laplaceův operátor (Laplace operator)** - popis tohoto operátoru je uveden v kapitole 2.6.1.2. včetně jeho masek (2.30), (2.31). Při použití tohoto operátoru lze v dialogovém okně (obr. 3.30c) vybrat jednu čtyř masek a dále nastavit, zda se operátor využije k ostření obrazu. Pokud ano, pak je postup analogický jako u nastavení Robertsova operátoru. Parametrem operace je typ masky a v případě ostření ještě také koeficient ostření.



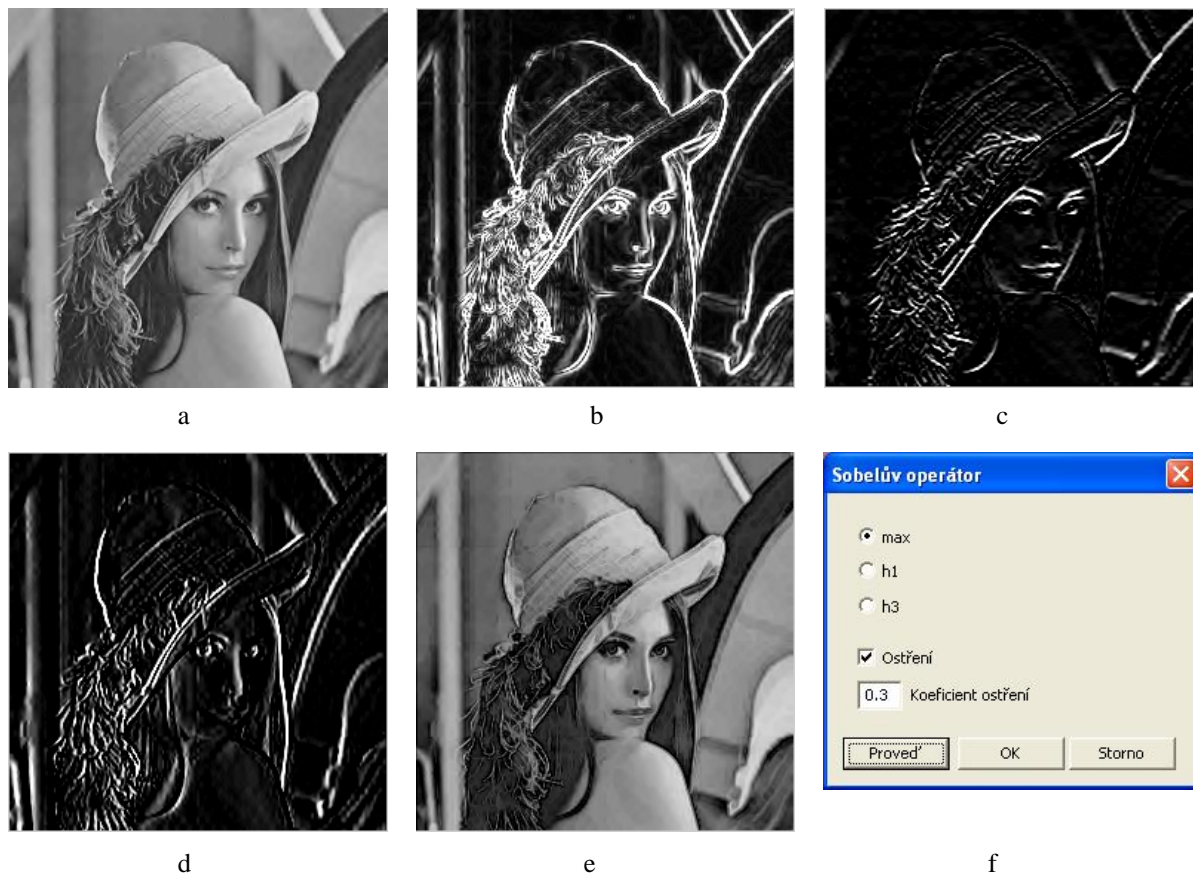
Obr. 3.30 Laplaceův operátor, vstupní obraz obr. 3.8 - (a) detekce hran maskou h_3 , (b) ostření s koeficientem 0.8, (c) dialogové okno Laplaceův operátor

- **Operátor Prewittové (Prewitt operator)** - kapitola 2.6.1.3.
- **Sobelův operátor (Sobel operator)** - kapitola 2.6.1.4.
- **Robinsonův operátor (Robinson operator)** - kapitola 2.6.1.5.
- **Kirschův operátor (Kirsch operator)** - kapitola 2.6.1.6.

Pro tyto čtyři operátory je společné dialogové okno (obr. 3.31f), nastavení i výpočetní algoritmus, rozdíl je v koeficientech masky jednotlivých operátorů ((2.34), (2.35), (2.36), (2.37)). Při výběru jednoho z těchto operátorů je otevřeno dialogové okno. Pokud je vybráno h_{max} , jsou hrany hledány výběrem masky s největším modulem

gradientu. Při vybrání h_1 jsou detekovány vodorovné hrany a při výběru položky h_3 jsou detekovány svislé hrany. Tyto operátory lze použít i k ostření hran zaškrtnutím položky Ostření a nastavením koeficientu ostření. Parametrem těchto operací je způsob detekce hran a v případě ostření ještě ostřicí koeficient.

Uvedené příklady jsou vytvořeny pomocí Sobelova operátoru.



Obr. 3.31 Sobelův operátor - (a) vstupní obraz, (b) h_{max} , (c) h_1 , (d) h_3 , (e) ostření s koeficientem 0.3, (f) dialogové okno Sobelův operátor

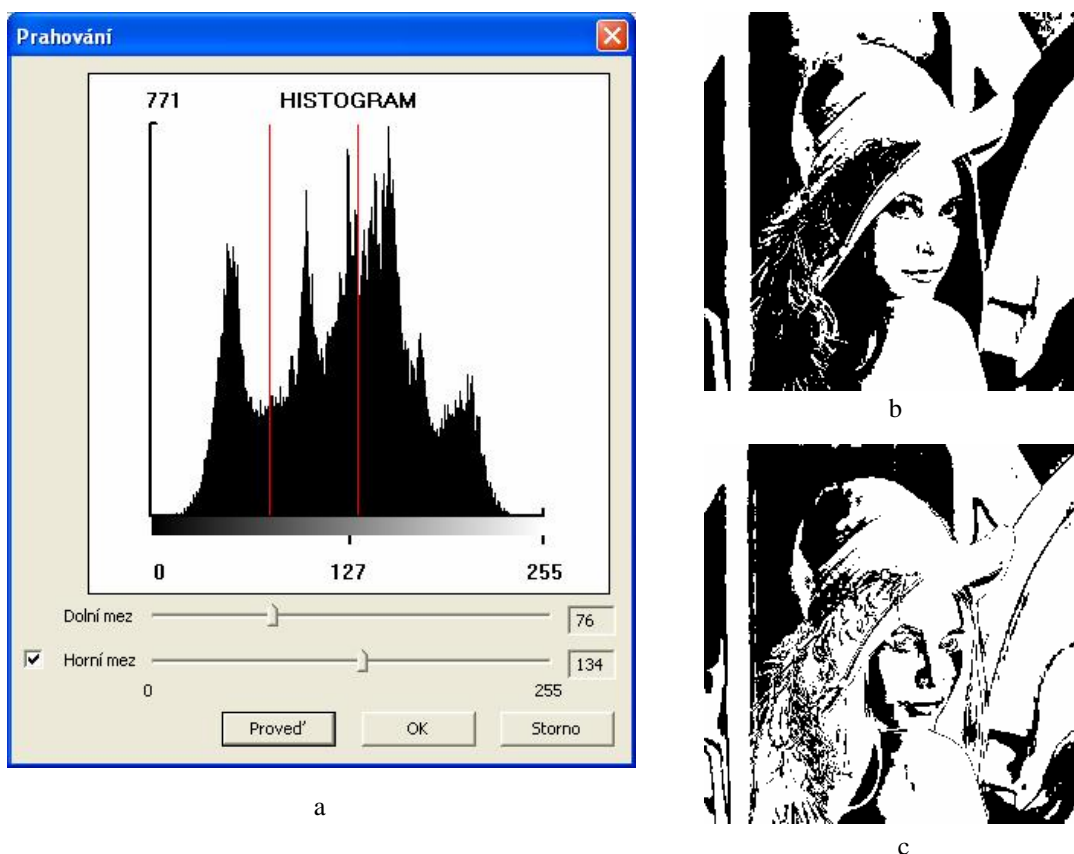
- **LoG operátor (LoG operator)** - popis tohoto operátoru je proveden v kapitole 2.6.2.1. s maskou (2.40). K nastavení parametru je použito stejné dialogové okno jako u Robertsova operátoru. Způsob nastavení parametru je s ním analogický.

3.4.6. Záložka Segmentace (Segmentation)

Tato záložka obsahuje pouze jednu operaci, kterou je obraz převeden ze šedotónové stupnice na binární obraz.

- **Prahování (Thresholding)** - tato metoda segmentace je popsána v kapitole 2.7.1. K nastavení parametrů prahování je použito dialogové okno (obr. 3.32a), ve kterém je zobrazen histogram segmentovaného obrazu. Pod ním jsou umístěny dva posuvníky, pomocí jejichž polohy jsou nastaveny prahy pro segmentaci. Aktivní

může být buď jeden, pak je prováděno prahování s jedním prahem dle vztahu (2.41) a parametr operace je jeden (poloha posuvníku) nebo jsou aktivní oba (zaškrtnutím boxu před spodním posuvníkem), pak poloha posuvníků určuje interval a převodní funkce je dána vztahem (2.42). V tomto případě jsou parametry operace dva (polohy posuvníků). Horní posuvník udává dolní hranici intervalu a spodní horní hranici intervalu. Poloha posuvníků je v histogramu symbolizována červenými čarami. Vpravo vedle posuvníků je zobrazena jejich aktuální poloha.



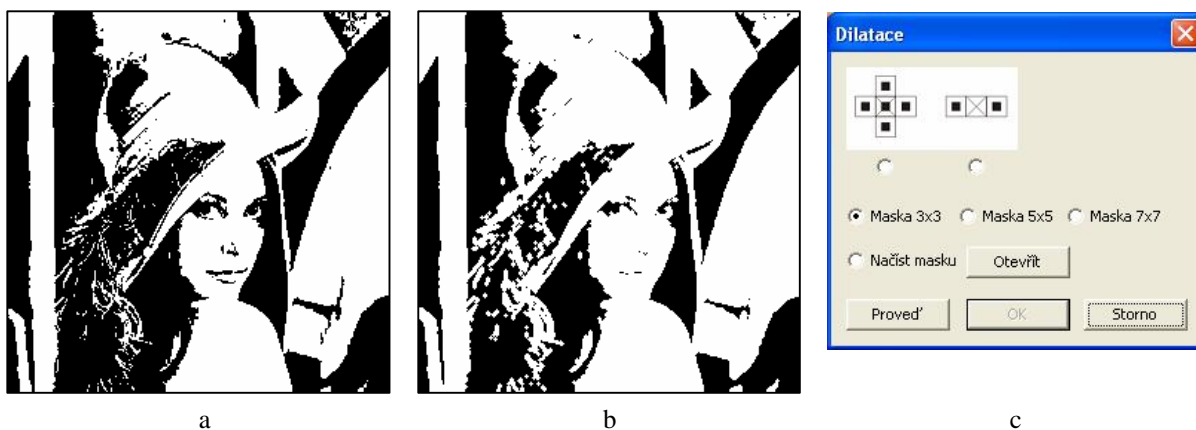
Obr. 3.32 Prahování, vstupní obraz obr. 3.8 - (a) dialogové okno Prahování, (b) prahování jedním prahem (123), (c) prahování s dvěma prahy (76, 134)

3.4.7. Záložka Binární matematická morfologie (Binary mathematics morphology)

Binární morfologické operace lze provádět pouze na binárních obrazech, které obsahují pouze 2 barvy, černou a bílou. Proto obraz musí být před používáním těchto operací, převeden pomocí Segmentace (Prahování) z šedotónového na binární, tzn. dokud není provedena segmentace, jsou binární morfologické operace nedostupné. Tyto operace jsou naprogramovány binárně (algoritmy byly převzaty, částečně upraveny a uzpůsobeny pro tuto diplomovou práci).

Binární matematické morfologické operace umístěné na této záložce používají společné dialogové okno, proto je popis uveden pouze u první z nich, dilatace, u ostatních je popis analogický.

- **Dilatace (Dilation)** - popis této morfologické operace je v kapitole 2.8.1.1. Při dilataci je otevřeno dialogové okno (obr. 3.33c), ve kterém je umožněno vybrat strukturní element, se kterým bude tato operace prováděna. Lze zvolit předdefinované elementy nebo načíst vlastní ze souboru. Výběr elementu je dán přepínačem. Při načítání ze souboru je element vždycky převeden na šedotónový a pak dále s prahem 128 na binární. Parametrem této operace je buď vybraný předdefinovaný typ elementu nebo jméno souboru s elementem.



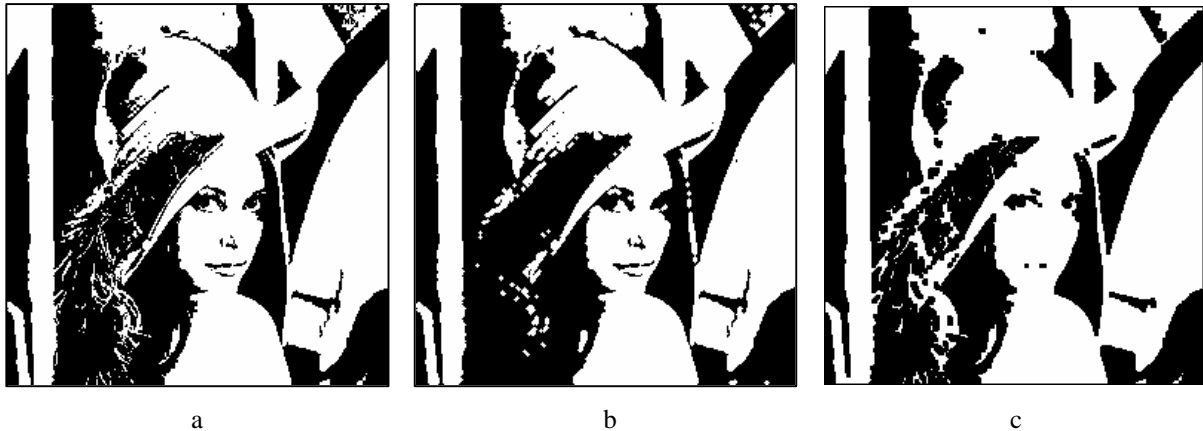
Obr. 3.33 Dilatace - (a) vstupní obraz, (b) dilatace, (c) dialogové okno Dilatace

- **Eroze (Erosion)** - popis je uveden v kapitole 2.8.1.2.



Obr. 3.34 Eroze - (a) vstupní obraz, (b) eroze

- **Otevření (Opening)** - popis je uveden v kapitole 2.8.1.3. (obr. 3.35b)
- **Uzavření (Closing)** - popis je uveden v kapitole 2.8.1.4. (obr. 3.35c)



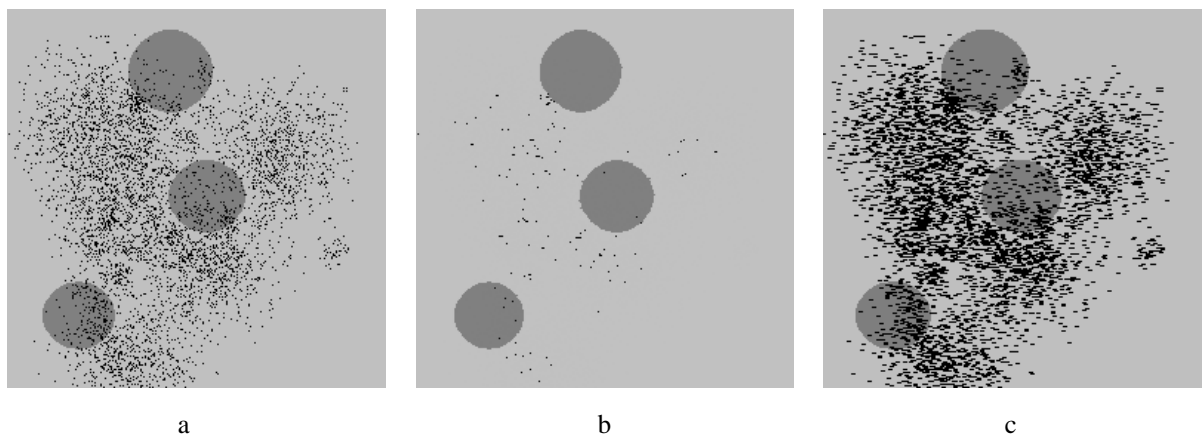
Obr. 3.35 (a) vstupní obraz, (b) otevření, (c) uzavření

- **Tref či miň (Hit or miss)** - pro tuto operaci je použito společné dialogové okno s výše uvedenými operacemi s tím rozdílem, že předdefinované strukturní elementy jsou nedostupné a strukturní element musí být načten ze souboru. Parametrem operace je jméno soubor s elementem.

3.4.8. Záložka Šedotónová matematická morfologie (Grayscale mathematic morphology)

Šedotónové morfologické operace jsou zobecněny pro šedotónové obrazy a tím odpadá nutnost je převádět na binární. Tyto operace jsou bez parametrů.

- **Dilatace (Dilation)** - popis je uveden v kapitole 2.8.2.1. (obr. 3.36b)
- **Eroze (Erosion)** - popis je uveden v kapitole 2.8.2.2. (obr. 3.36c)



Obr. 3.36 Šedotónová matematická morfologie - (a) vstupní obraz, (b) dilatace, (c) eroze

- **Otevření (Opening)** - popis je uveden v kapitole 2.8.2.3. (obr. 3.37b)
- **Uzavření (Closing)** - popis je uveden v kapitole 2.8.2.4. (obr. 3.37c)

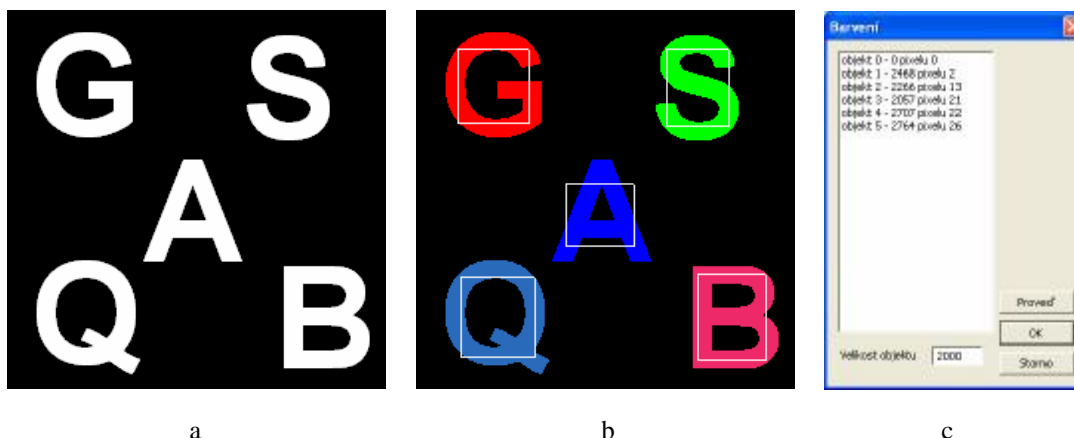


Obr. 3.37 Šedotónová matematická morfologie - (a) vstupní obraz, (b) otevření, (c) uzavření

3.4.9. Záložka Rozpoznávání objektu (Pattern recognition)

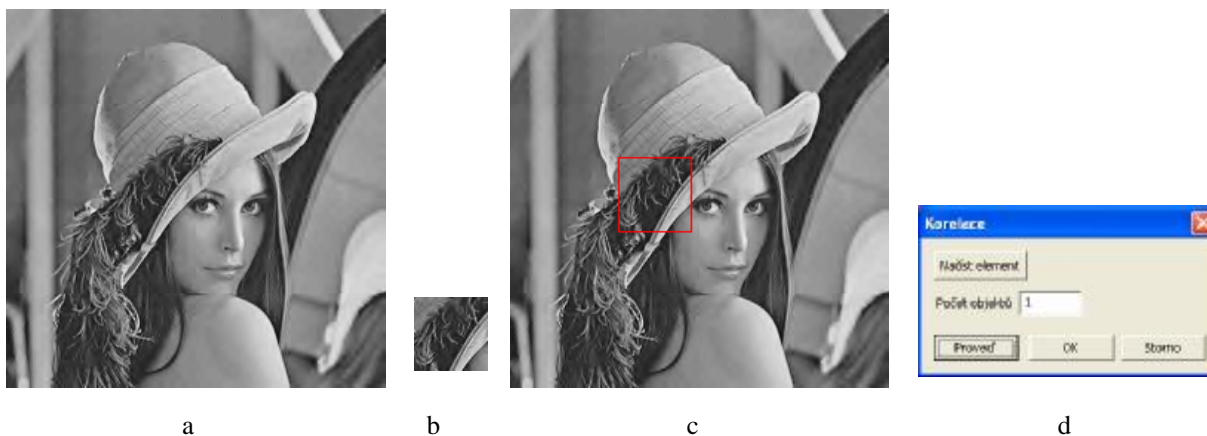
Na této záložce jsou umístěny dvě základní metody zpracování a rozpoznávání obrazu.

- **Barvení (Colouring)** - tato metoda rozpoznávání je popsána v kapitole 2.9.1. Barvení lze použít na binární obraz, tzn. musí být obraz před touto operací segmentován, jinak je operace barvení nedostupná. Její průběh je rozdělen ho několika kroků. Nejprve je provedeno barvení a výpočet těžišť jednotlivých oblastí. Po této části je zobrazen výsledek barvení v aktivním vizualizačním okně, kde jsou jednotlivé oblasti barevně odlišeny a ohraničeny těžištěm. Dále je vyvoláno dialogové okno (3.38c), ve kterém jsou vypsány jednotlivé oblasti společně s jejich velikostí. Zadáním čísla do kolonky *Velikost objektu* je určen práh pro eliminaci malých oblastí. Po stisku *Proved'* je provedena eliminace oblastí menších než je tento práh. Při kliknutí na některou obarvenou oblasti se tato oblast označí modře ve výpise v dialogovém okně. Parametrem této operace je eliminační práh.



Obr. 3.38 Barvení - (a) vstupní obraz, (b) barvení obrazu, (c) dialogové okno Barvení

- **Korelace (Correlation)** - tato operace je popsána v kapitole 2.9.2. Pomocí dialogového okna (obr. 3.39d) je načten (tlačítko *Načíst element*) vzor (strukturní element) ze souboru a nastaven počet hledaných objektů (kolonka *Počet objektů*). Nalezené objekty jsou pak ohraničeny červenou čarou. Parametry této metody rozpoznávání jsou jméno souboru s hledaným vzorem a počet hledaných objektů.



Obr. 3.39 Korelace - (a) vstupní obraz, (b) element, (c) korelace, (d) dialogové okno Korelace

4. Závěr

Cílem mé diplomové práce bylo navrhnout a realizovat systém pro automatizované zpracování rozpoznávání obrazu s graficky přívětivým a jednoduše ovladatelným rozhraním, který umožní rychlou prvotní analýzu obrazu. Celý systém byl naprogramován v jazyce C v programovacím prostředí Microsoft Visual Studio .NET s využitím programovatelného rozhraní Windows (API). Do tohoto systému jsou implementovány metody pro zpracování a rozpoznávání obrazu, jejichž teoretický popis byl proveden v kapitole 2..Celkem je těchto metod 52. Jednotlivé metody jsou naprogramovány ve formě samostatných na sobě nezávislých modulů a jsou rozděleny do skupin dle charakteru operace viz obr. 3.7. Jednotlivé skupiny obrazových operací jsou v programu pro větší přehlednost barevně odlišeny. U metod, jenž vyžadují nastavení pro ně typických parametrů, je zobrazeno dialogové okno, které toto nastavení umožňuje. Dále tento systém dovoluje řadit tyto moduly za sebe a tím tak vytvářet rozpoznávací řetězec. Tento řetězec lze uložit do souboru a použít ho později na jakýkoliv obraz, jeho aplikace na zvolený obraz probíhá automaticky. Jednotlivé mezivýsledky zpracování a rozpoznávání obrazu jsou v systému uloženy v paměti a zobrazovány s možností výběru způsobu jejich interpretace (obraz, histogram, spektrogram). Systém umožňuje návrat a zobrazování předešlých dílčích výsledků kliknutím počítačové myši na danou operaci v řetězci. Z tohoto důvodu jsou v systému umístěny dvě vizualizační okna, čímž je umožněno porovnávat jednotlivé dílčí výsledky z různých fází zpracování a rozpoznávání obrazu. Dále lze tyto "mezivýsledky" ukládat v jakékoliv fázi do grafického nekomprimovaného formátu bmp. Podporované formáty otevíraných souborů jsou bmp, gif a jpg. Celý systém je navržen a vytvořen jako otevřený, tj. je možné do něho přidávat dle potřeby nové moduly.

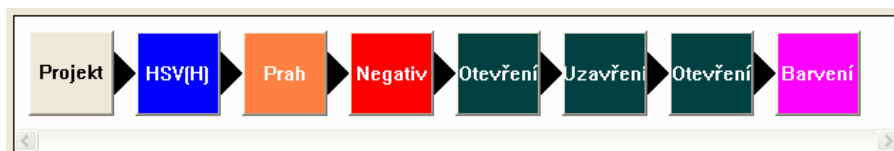
V systému jsou implementovány jednotlivé metody a algoritmy dle zadání. V závěru této diplomové práce se podařilo systém rozšířit oproti zadání o dvě základní metody pro rozpoznávání obrazu. Jsou to metody barvení a korelace.

V budoucnu by se mohlo pokračovat v implementaci dalších metod do tohoto systému, především půjde o algoritmy a metody zabývající se rozpoznáváním obrazu.

5. Praktické ukázky

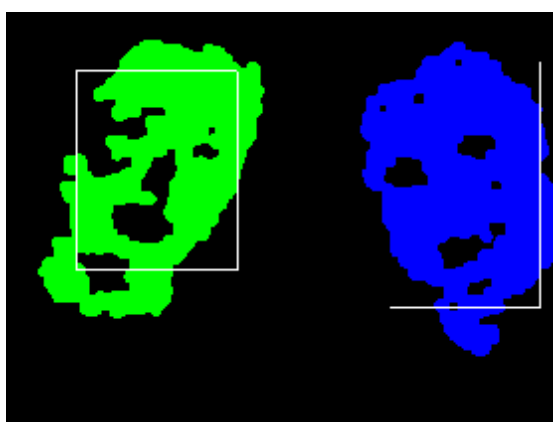
5.1. Detekce obličejů

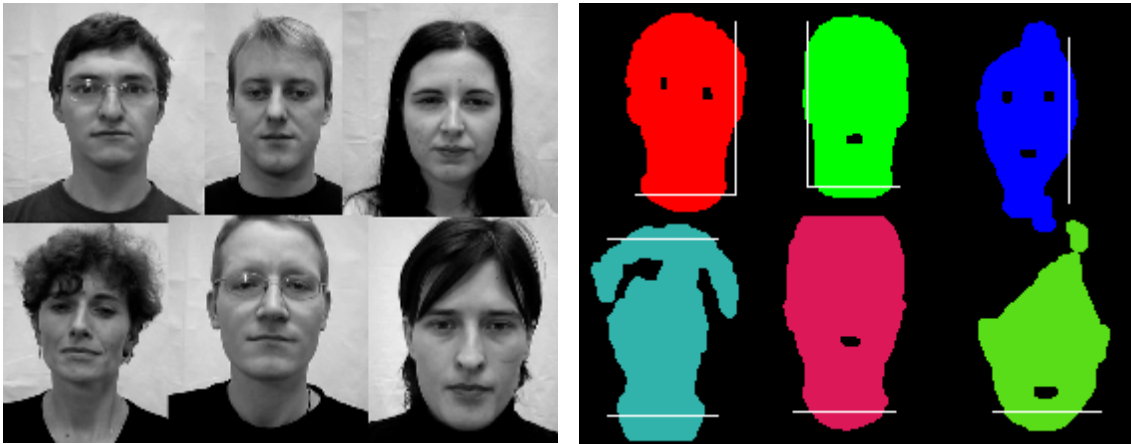
Na tomto příkladě je ukázáno použití stejného rozpoznávacího řetězce na obrazy, na kterých jsou zobrazeny lidské obličeje. Výsledkem jsou jednotlivé obarvené oblasti, kde se vyskytují obličeje. Rozpoznávací řetězec je složen z následujících operací:



Obr. 5.1 Rozpoznávací řetězec pro detekci obličejů

První operací je operace ze záložky Transformace barev, kterou je převeden barevný obraz na šedotónový. V tomto případě bylo použito transformace na složku H z barevného modelu HSV . Dále je obraz převeden z šedotónové stupnice pomocí prahování na binární obraz, s kterým je následně provedena operace Negativ. Poté jsou aplikovány operace binární matematické morfologie s cílem z obrazu odstranit objekty menší než jsou použité strukturální elementy, spojit oblasti, které jsou blízko u sebe a zaplnit malé díry. Poslední operací je barvení, kterým jsou barevně odlišeny jednotlivé oblasti (obličeje). Prahem pro eliminaci malých objektů je velikost objektů menší než 3000 bodů.

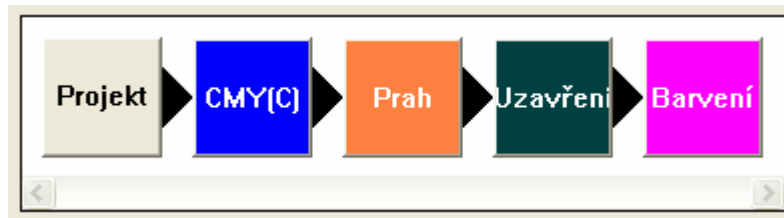




Obr. 5.2 Barvení oblastí jednotlivých obrazů s různými počty obličejů

5.2. Velikost mincí

Na tomto příkladě je ukázána detekce mincí různých hodnot, které jsou různě velké. Kritériem je tedy jejich velikost.

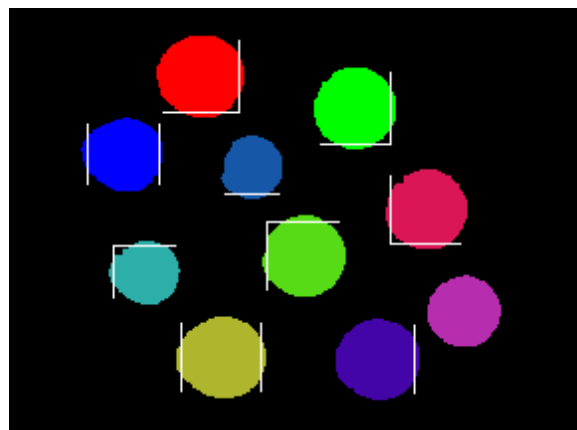


Obr. 5.3 Rozpoznávací řetězec pro detekci mincí

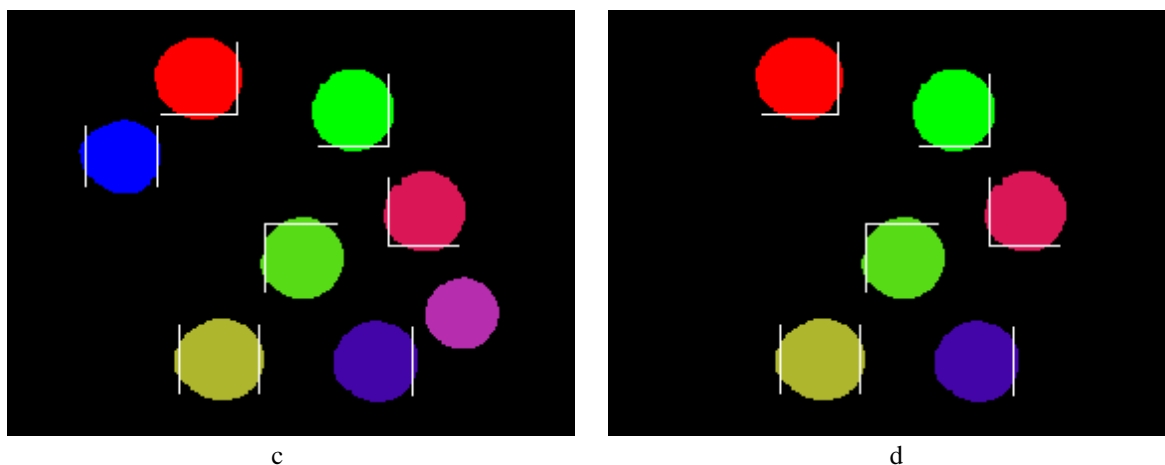
První operací je převod z barevného obrazu na šedotónový. Pro převod je použita složka C z barevného modelu CMY . Dále je provedeno prahování, kterým je obraz převeden na binární. Předposlední operací je operace uzavření, kterou jsou zaplněny malé díry a vyhlazen obrys. Nakonec je provedeno barvení, kde jsou jednotlivé mince obarveny různými barvami. V závislosti na velikosti eliminačního prahu jsou z obrazu odstraněny mince, které jsou menší než je tento práh.



a



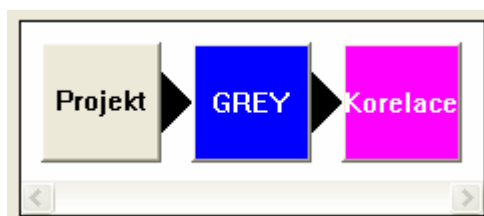
b



Obr. 5.4 (a) vstupní obraz s mincemi, (b) barvení bez prahu, (c) barvení s eliminací nejmenších mincí (koruny), (d) barvení se zachováním největších mincí (20-ti koruny, 5-ti koruny a koruny jsou eliminovány)

5.3. Hledání mincí

Na tomto příkladě je ukázáno hledání mincí na základě vzoru. Vzor je načten ze souboru a je hledána jeho shoda s částí obrazu. Výsledkem jsou pak ohraničené oblasti, kde byla nalezena nejvíce shodná část obrazu a vzoru. Počet hledaných oblastí je parametrem korelace.



Obr. 5.5 Řetězec pro nalezení shody části obrazu se vzorem

Barevný obraz je nejprve převeden na šedotónovou stupnici barevnou transformací RGB2GREY a dále je operací korelace hledána shoda etalonu s částí obrazu, shodné části jsou ohraničeny červeně a jejich počet je dán parametrem korelace.



a



b



c



d



e

Obr. 5.6 (a) vstupní obraz s mincemi, (b) vzor - rub mince, (c) vzor - líc mince, (d) korelace - rub mince, (e) korelace - líc mince

Použitá literatura:

- [1] HLAVÁČ, V., ŠONKA, M.: Počítačové vidění, Grada, Praha 1992, ISBN 80-85424-67-3
- [2] HLAVÁČ V., SEDLÁČEK, M.: Zpracování signálu a obrazu, Skripta FEL ČVUT, Praha 2000, ISBN 80-01-02114-9
- [3] ŠONKA, M., HLAVÁČ, V., BOYLE, R.: Image Processing, Analysis, and Machine Vision, PWS Publishing, 1998, ISBN 0-534-953-93
- [4] ROSENFELD, A., KAK, A., C.: Digital Picture Processing. Academic Press, New York, 2nd edition, 1982
- [5] PIHAN, R.: Techniky fotografování, [online], staženo 20.11.2004, dostupné z: <http://www.fotoroman.cz/techniques/eye_camera.htm>
- [6] SMITH, A., R.: Color gamut transform pairs, Computer Graphics, 12(3):12-19, 1978.
- [7] Barevné prostory, [online], staženo 11.11.2004, dostupné z: <<http://goro.pescz.cz/colorsaces.php>>
- [8] EasyRGB: Color Conversion Formulas, [online], staženo 15.11.2004, dostupné z: <<http://www.easyrgb.com/math.html>>
- [9] MOIK, J. G.: Digital Processing of Remotely Sensed Images, NASA SP- 431, Washington, DC, 1980
- [10] NAGAO, M., MATSUYAMA, T.: A Structural Analysis pf Complex Aerial Photographs, Plenum Press, New York, 1980
- [11] TYAN, S. G.: Median filtering, deterministic properties, In T S Huang editor, Two Dimensional Digital Signal Processing, volume II. Springer Verlag, Berlin, 1981
- [12] ROBERTS, L. G.: Machine perception of three-dimensional solids, In J T Tippett, editor, Optical and Electro-Optical Information Processing, pages 159-197, MIT Press, Cambridge, MA, 1965
- [13] MARR, D., HILDRETH, E.: Theory of edge detection, Proceedings of the Royal Society, B 207:187-217, 1980
- [14] JAIN, R., KASTURI, R., SCHUNCK, B., G.: Machine Vision, McGraw-Hill, New York, 1995
- [15] MATHERON, G.: Eléments pour une theorie des milieux poreux (in French), Masson, Paris, 1967

- [16] SERRA, J.: Image Analysis and Mathematical Morphology, Academic Press, London, 1982
- [17] HARALICK, R., M., SHAPIRO, L.G.: Computer And Robot Vision, Volume I., Addison-Wesley, Reading, MA, 1992
- [18] APPEL, K., HAKEN, W.: Every planar map is four colourable: Part I: discharging, Illinois Journal of Mathematics, 21:429-490, 1970
- [19] BALLARD, D., H., BROWN, C., M.: Computer Vision, Prentice-Hall, Englewood Cliffs, NJ, 1982
- [20] HEROUT, P.: Učebnice jazyka C, III. upravené vydání, KOPP, České Budějovice 2001, ISBN 80-85828-21-9
- [21] MICROSOFT: Msdn library, [online], staženo 6.11.2004, dostupné z: <<http://msdn.microsoft.com/>>
- [22] PETZOLD, CH.: Programování ve Windows, Computer Press, Praha 1999, ISBN 80-7226-206-8
- [23] SIMON, R., J., GOUKER, M., BARNES, B., C.: Win32 API, svazek 1 a 2, UNIS Publishing s.r.o., Brno 1997, ISBN 80-86097-06-4
- [24] CHALUPA, R.: Učíme se WinAPI, [online], staženo 21.10.2004, dostupné z: <<http://www.builder.cz/serial91.html>>