

# Krásy počítačové grafiky: Úpravy rastrového obrazu

Tomáš Maršálek

9. března 2012

## 1 Zadání

Vyzkoušejte si naprogramovat metody úpravy digitalizovaného obrazu z přednášky, jako je ostření, reliéf, warping, morphing.... Za program umějící aspoň jednu techniku získáte 5 bodů, za každou další naprogramovanou techniku získáte max. 3 body podle obtížnosti, dohromady nejvýše 17 bodů. Program musí být schopen načíst ze souboru obraz v rastrovém formátu a zase jej uložit, zobrazit původní a změněný obraz s možností návratu o 1 akci. Odevzdáváte jako obvykle zdrojový text, EXE a dokumentaci.

## 2 Implementované filtry

### 2.1 Detekce hran

Detekce hran je horní propust pro obrazový signál. Vysokou frekvencí v rastrovém obrazu se rozumí velký rozdíl v intenzitě barvy v jednotlivých kanálech.

#### 2.1.1 Sobel, Prewitt a Roberts Cross

Jedná se o metody, které pro každý obrazový bod aplikují numerickou aproximaci první derivace v tomto bodě. Pro dva rozměry se vypočte velikost první derivace jako velikost gradientu, kde jednotlivé parciální derivace mají své příslušné konvoluční matice. Vyšší změna barvy se projeví vyšší hodnotou derivace, což vidíme jako detekovanou hranu.

Například Sobelův operátor používá konvoluční matice:

$$D_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad D_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$



Obrázek 1: Originální obrázek

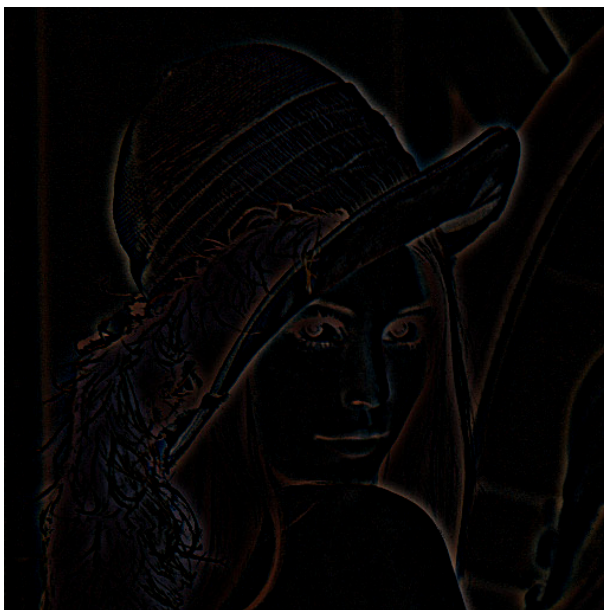


Obrázek 2: Detekce hran, Sobel

### 2.1.2 Rozdíl Gaussovských rozostření

Rozostření funguje jako dolní propust pro obrazový signál. Největší změny oproti původnímu obrázku nastanou právě v místech, kde se nachází hrany. Ty se relativně rozostří nejvíce. Proto když od původního obrázku odečteme jeho rozostřenou verzi, největší změnu uvidíme právě v místech hran.

Rozdíl dvou různě silných rozostření je pouze zobecnění rozdílu rozostření s původním obrázkem.

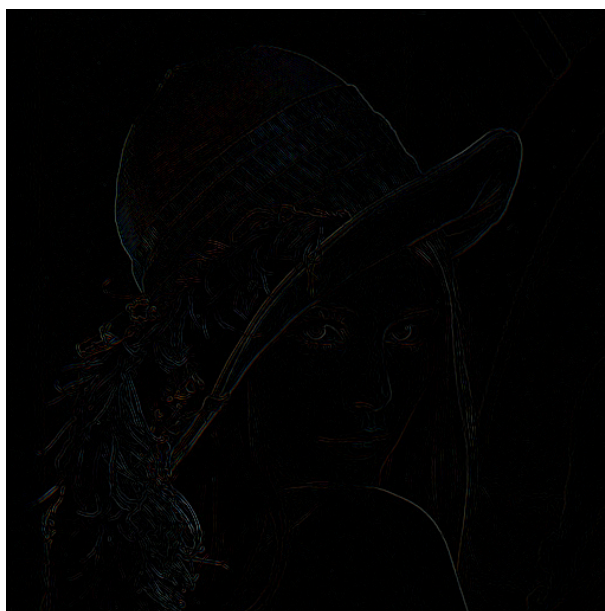


Obrázek 3: Rozdíl Gaussovských rozostření s variancemi 5 a 0.1

### 2.1.3 Laplacian of Sobel

Laplaceův operátor je numerickou aproximací druhé derivace. Aplikací na rastr získáme velmi tenké hrany, mnohem tenčí, než získané z výše uvedených metod. Nevýhodou je, že je velmi citlivý na jakýkoliv šum. Proto některé filtry před použitím Laplaceova operátoru odstraní šum, například Gaussovským rozostřením (Laplacian of Gaussian) nebo Mediánovým filtrem.

Zde je Laplaceův operátor aplikován po Sobelově operátoru, výsledkem jsou velmi tenké hrany s minimálním okolním šumem.



Obrázek 4: Laplacian of Sobel

## 2.2 Doostření

Doostření obrázku je výsledkem součtu původního obrázku s jeho Laplaciánem. Standardně je implementována možnost měnit intenzitu doostření pomocí koeficientu ostření.

$$B = A + c \cdot \Delta A$$

$A$  je původní obrázek,  $\Delta$  je Laplaceův operátor,  $B$  je doostřený obrázek a  $c$  je koeficient ostření.



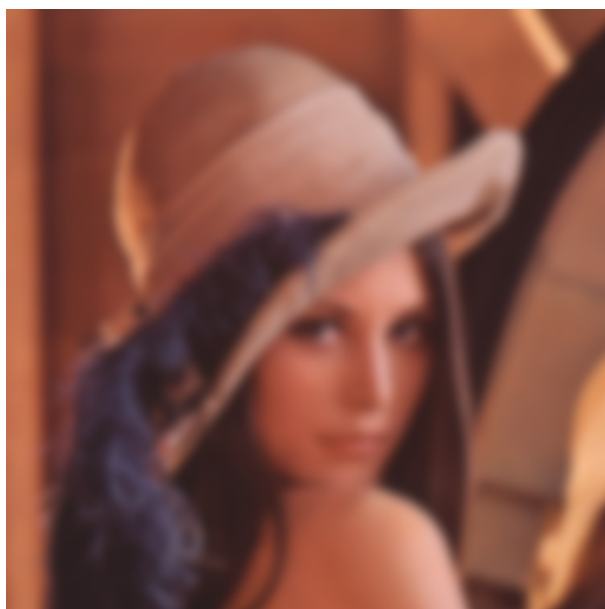
Obrázek 5: Doostření s koeficientem 5

## 2.3 Gaussové rozostření

Obecně filtr rozostření můžeme chápat jako konvoluci s nějakou průměrující funkcí. Například standardní rozostření, tzn. průměr okolních bodů je konvoluce obrázku s dvourozměrnou jednotkovou funkcí. Gaussové rozostření pouze místo jednotkové funkce používá Gaussovu křivku jako funkci s koeficienty průměrování.

Konvoluce s větší konvoluční maticí je výpočetně náročná operace, pro každý bod musí provést  $k^2$  operací, kde  $k$  je velikost strany konvoluční ma-

tice. Při detekci hran tento problém není třeba řešit, protože pracujeme pouze s maticemi velikosti  $3 \times 3$ . Při standardním nastavení Gaussovského rozostření však už pracujeme s maticí  $16 \times 16$ , při větších rozptylech pak ještě mnohem většími, proto je použita známá optimalizace rozdělení konvoluce do jednotlivých komponent. Dvourozměrná Gaussova funkce je totiž součin dvou jednorozměrných, využití spočívá v provedení dvou jednorozměrných konvolucí po sobě, namísto jedné dvourozměrné.



Obrázek 6: Gaussovské rozostření s variancí 3



## 2.4 Inverze barev

Je velmi jednoduchý filtr, ale je zahrnut do programu, aby bylo možné pozorovat oba způsoby detekce hran: světlé na tmavém pozadí a tmavé na světlém pozadí. Filtr pouze odečte barvu od maximální hodnoty pro každý pixel.



Obrázek 7: Inverze barev

## 2.5 Odbarvení

Tento filtr je také jednoduchý, ale je zajímavý, protože existuje více způsobů, jak odbarvení dosáhnout. První je jednoduchý průměr všech tří kanálů. Pro lidské oko je však kvalitnější volba luminance, protože průměruje kanály podle vnímání lidského oka ( $.2125 \cdot R + .7152 \cdot G + .0722 \cdot B$ ).



Obrázek 8: Odbarvení pomocí luminance

## 3 Uživatelské rozhraní

Grafické rozhraní je vytvořeno pomocí knihovny GTK+ 3.0, proto by měl vzhled vypadat nativně na jednotlivých operačních systémech.

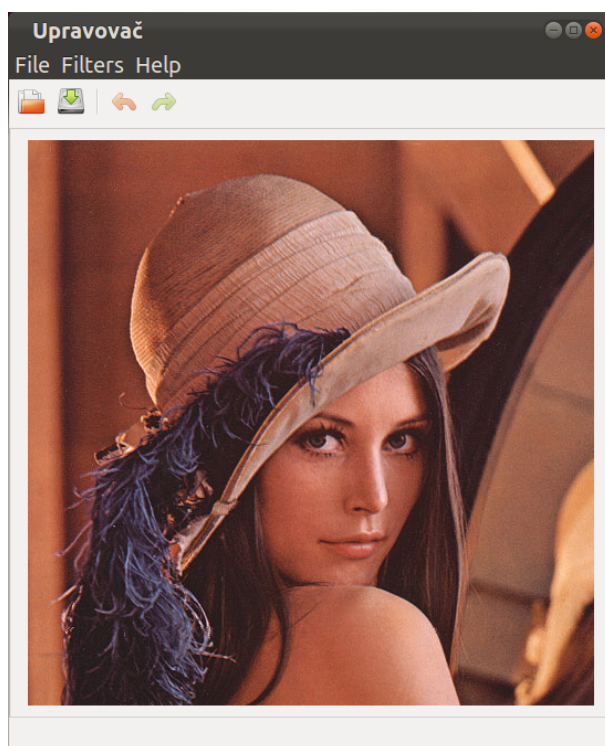
Uživatel může otevřít obrázek ve formátu .bmp, .jpg nebo .png a uložit ho pod stejným nebo změněným názvem. Pro možnost porovnání provedených změn je implementován systém historie zpět a vpřed. Kromě těchto základních funkcí je uživateli umožněno používat výše zmíněné filtry, další funkcionality není implementována.



## 4 Implementace

Jazyk programu je ANSI C, použitá knihovna grafického prostředí je GTK+ 3.0 a Glade 3.

Pro spuštění programu musí být ve stejném adresáři přítomen soubor s uživatelským rozhraním upravovac.glade.



Obrázek 9: Ukázka uživatelského rozhraní

## 5 Závěr

Tento program má velký potenciál na další úpravy, především v uživatelském rozhraní a dalších filtrech. Nicméně v takovéto podobě by měl splňovat uvedené zadání.