

SLOVENSKÁ TECHNICKÁ UNIVERZITA, FAKULTA ELEKTROTECHNIKY A
INFORMATIKY

VIZS – úloha 2

Detekcia hrán pomocou laplaceovho operátora

Adam Sojka 72515, Filip Štec 72520
9.4.2017

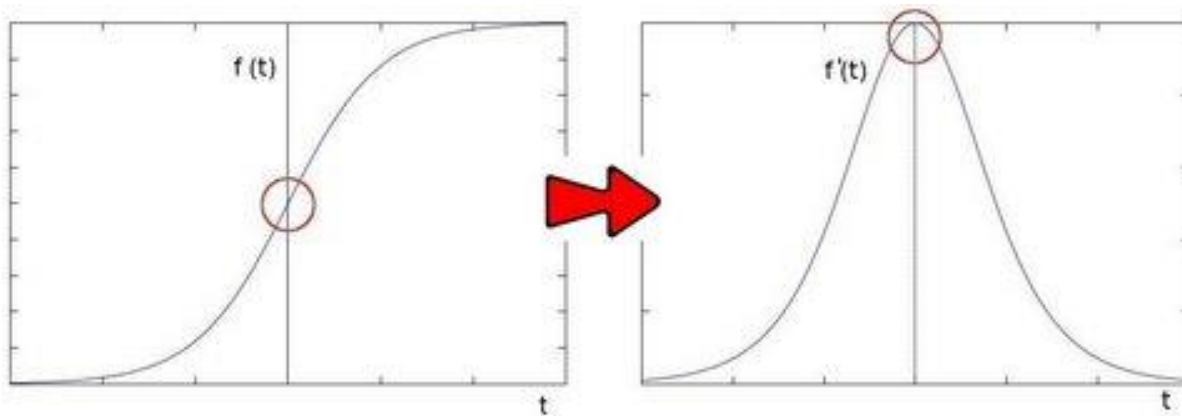
Úloha

Naprogramujte algoritmus detekcie hrán laplaceovým operátorom, pričom ako vstup použijete obrázok vo formáte .bmp alebo .jpg. Nemôžete použiť žiadnu knižnicu pre spracovanie obrazu!

Teoretický postup

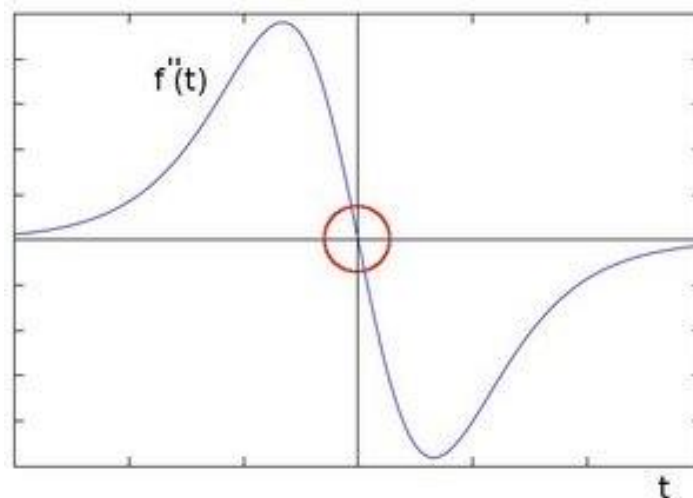
Najprv je potrebné konvertovať RGB model obrazu z kamery na monochromatický model, teda čiernobiely aby bolo možné na obraz aplikovať laplaceov operátor.

Filtrovanie laplaceovým operátorom je divergencia gradientu funkcie jasů pixelov obrazu, ktorý závisí od x a y súradníc pixelu. Prvou deriváciou obrazu získame obraz, kde je pri skokovej zmene hodnota nového obrazu najvyššia:



Obrázok 1 Prvá derivácia funkcie

Pri druhej derivácii obrazu bude pri najväčšej zmene hodnota nového obrazu nulová:



Obrázok 2 Druhá derivácia funkcie

Odvozenie vzťahu pre funkciu vyfiltrovaného obrazu, kde f je funkcia jasú pixelu a x, y sú súradnice pixelu:

$$\nabla^2 f = \frac{\partial^2 f}{\partial^2 x} + \frac{\partial^2 f}{\partial^2 y}$$

Obraz má konečný počet pixelov, takže diferencia bude rozdielom medzi susednými pixelmi:

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

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

Vznikne maska – štvorcová matica 3x3:

0	1	0
1	-4	1
0	1	0

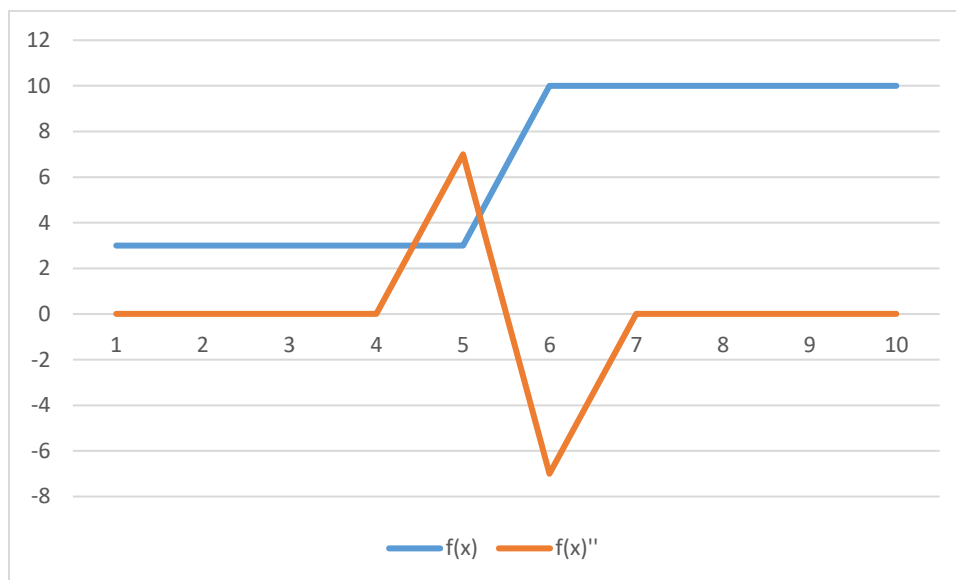
Potom spravíme dvojrozmernú konvolúciu masky a pôvodného obrazu.

Spravíme absolútnu hodnotu novo vypočítaných hodnôt pixelov čísla väčšie ako 255 orežeme na číslo 255, za predpokladu, že používame model s 256 úrovňami jasú.

Príklad aplikácie laplaceovho operátora na jeden riadok obrazu:

$$\frac{d^2 f}{d^2 x} = f(x-1) - 2f(x) + f(x+1)$$

f(x)	3	3	3	3	3	3	10	10	10	10	10	10
F(x)''	-3	0	0	0	0	7	-7	0	0	0	0	-10



Obrázok 3 Aplikácia laplaceovho operátora na funkciu

Programové vybavenie

Ku zdrojovým kódom ako aj tejto dokumentácii je možné sa dostať na umiestnení:

<https://github.com/Smadas/VIZSul2>

Všetky zdrojové kódy sú písané v C++ s použitím openCv 3.1, aj staršie verzie by mali byť kompatibilné.

Funkcia main

1. Deklarácia obrazových matíc
2. Načítanie obrázku
3. Overenie, či bol obrázok načítaný, ak nie skončenie programu
4. Vytvorenie okien pre originálny obrázok, openCv laplace obrázok a naprogramovaný laplace obrázok
5. Aplikácia openCv laplace - `applyLaplace(imageOrig)`
6. Aplikácia naprogramovaného laplace - `applyLaplaceProg(imageOrig)`
7. Vykreslenie obrázkov do okien
8. Zapísanie spracovaného obrázka do súboru `vystup.png`
9. Čakanie, kým užívateľ stlačí klávesu

Funkcia `applyLaplace`

1. Deklarácia obrazových matíc

2. Deklarácia parametrov laplaceového operátora
3. Transformácia obrazu na monochromatický model
4. Aplikácia laplaceovho operátora na obraz openCv funkciou
5. Zmena mierky hodnôt jasu výsledného obrazu
6. Vrátenie upraveného obrazu

Funkcia `applyLaplaceProg`

1. Deklarácia obrazových matíc
2. Deklarácia parametrov laplaceového operátora
3. Transformácia obrazu na monochromatický model
4. Naklonovanie vstupného obrazu do novej matice
5. Prejdenie každého pixelu obrazu
 - a. Výpočet novej hodnoty pixelu - `computeOnePixel(src, changed, i, j)`
6. Vrátenie upraveného obrazu

Funkcia `computeOnePixel`

1. Deklarácia premennej pre pixel
2. Pripočítanie váhovaného pôvodného pixelu váhou -4
3. Pripočítaj horný pixel ak nie je mimo obrazu
4. Pripočítaj ľavý pixel ak nie je mimo obrazu
5. Pripočítaj dolný pixel ak nie je mimo obrazu
6. Pripočítaj pravý pixel ak nie je mimo obrazu
7. Vráť hodnotu nového pixelu v novej mierke - `scalePixelValSymmetric(pixelLaplaceVal)`

Funkcia `scalePixelValSymmetric`

1. Vynásob hodnotu pixelu konštantou 3
2. Absolútna hodnota pixelu
3. Orezanie hodnoty pixelu na hodnotu 255 ak je jeho hodnota väčšia ako 255
4. Vrátenie novej hodnoty pixelu

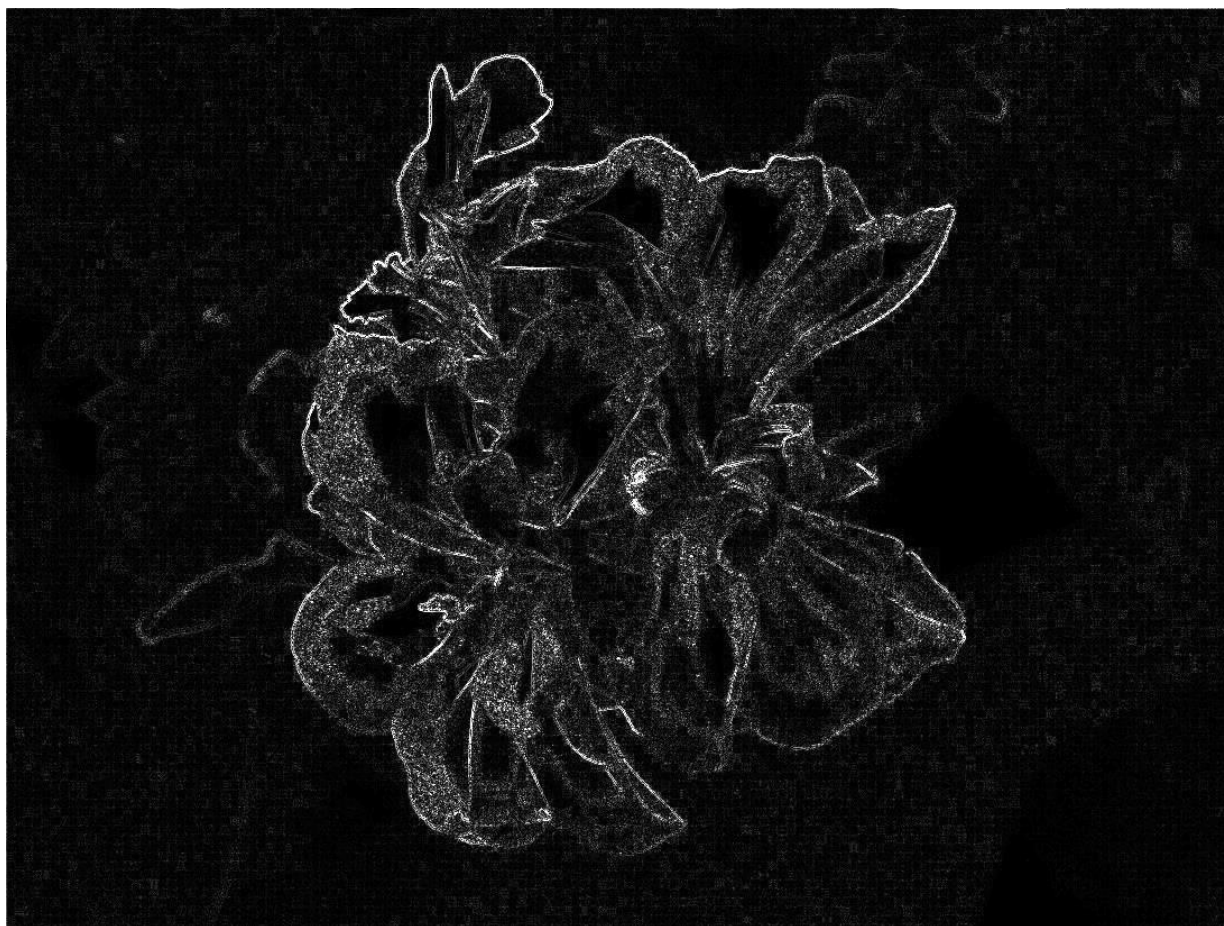
Funkcia `writelImageToFile`

1. Deklarácia parametrov pre zapísanie obrázka do súboru
2. Pokus o zapísanie obrázka do súboru `vystup.png`
3. Ak sa nepodarilo zapísať do súboru, vypíše chybovú hlášku a vráti nepravdu
4. Inak vráti pravdu

Výsledok aplikácie laplaceovho operátora naším algoritmom



Obrázok 4 Pôvodný obrázok kvetu



Obrázok 5 Obrázok kvetu, na ktorý bol aplikovaný laplaceov operátor

Zdroje

http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/laplace_operator/laplace_operator.html

<https://www.youtube.com/watch?v=kJKgCwUzkmc&t=200s>