



Segmentace I

Pokročilá analýza obrazu

Mgr. Markéta Trnečková, Ph.D.

Obraz

- Operace s obrazy:
 - Vstupem je obraz – výstupem je obraz
 - Vstupem je obraz – výstupem jsou nějaké atributy

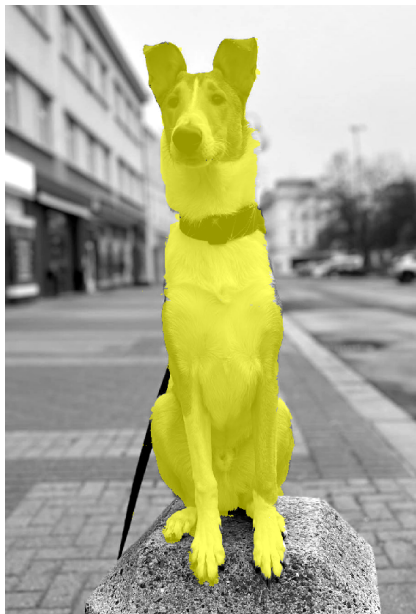
Příklad

Napadá vás nějaký konkrétní příklad operací, se kterými jsme se už setkali, které mají na výstupu obraz? A které mají na výstupu nějaké atributy?

- Segmentace (dělení obrazu na nějaké logické části)
 - dělení obrazu dle nějakých prudkých změn
 - dělení obrazu na regiony, které mají společné vlastnosti
 - jejich kombinace

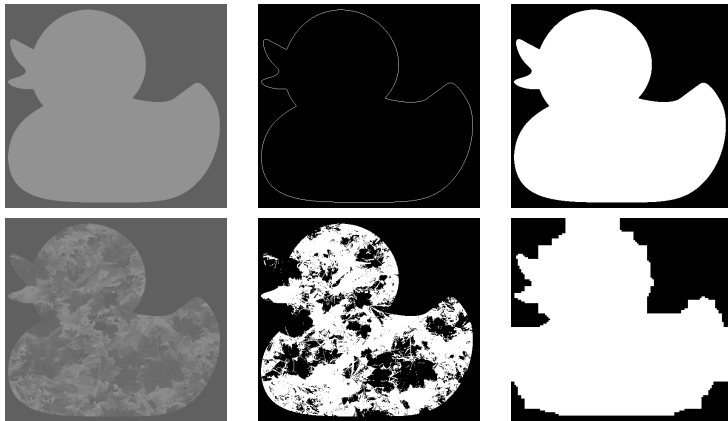
Segmentace

- R – celý region (obraz)
- **Segmentace** – rozdělení obrazu na jednotlivé regiony R_1, \dots, R_n :
 - $\bigcup_{i=1}^n R_i = R$
 - R_i jsou spojité
 - $R_i \cap R_j = \emptyset$, pro všechna $i \neq j$
 - $Q(R_i) = TRUE$ – všechny pixely v oblasti splňují nějakou podmínku
 - sousední regiony stejnou podmínku nesplňují
 $Q(R_i \cup R_j) = FALSE$
- dělení obrazu na části, které splňují nějakou podmínku:
 - pixely v oblasti jsou „shodné“
 - pixely v sousedních oblastech jsou „rozdílné“



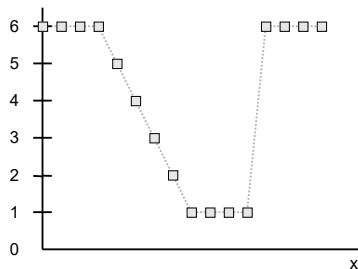
Segmentace

- 1. kategorie = snadno najdeme hranici regionu **Edge-based segmentace**
- 2. kategorie = dělíme oblasti dokud nejsou dostatečně shodné **Region-based segmentace**



Detekce bodů, linií a hran

- Hledáme ostré lokální změny
 - Body
 - Hrany – spojitá místa, kde dochází k velké změně
 - Linie – úzké hrany, které mají pozadí na obou stranách
- K hledání změn využíváme derivace
- **konstantní, přechod, skok**

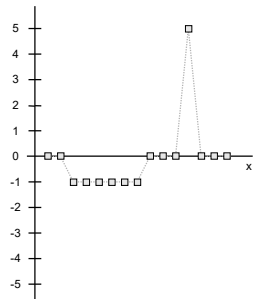
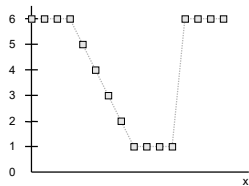


1. derivace

- je rovna 0 v oblastech s konstantní intenzitou
- nenulová na počátku na přechodech a skocích
- nenulová podél přechodu

- **Příklad:**

$$\frac{\partial f}{\partial x} = f(x + 1) - f(x)$$

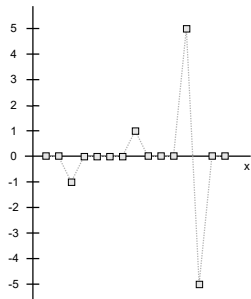
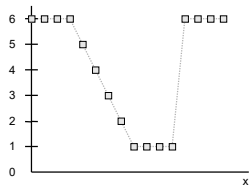


2. derivace

- je rovna 0 v oblastech s konstantní intenzitou
- nenulová na počátku a na konci přechodů a skoků
- nulová podél přechodu

■ **Příklad:**

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



Taylorova řada

- První derivaci v bodě můžeme aproximovat **Taylorovou řadou**

$$f(x + \delta x) = f(x) + \Delta x \frac{\partial f(x)}{\partial x} + \frac{(\Delta x)^2}{2!} \frac{\partial^2 f(x)}{\partial x^2} + \frac{(\Delta x)^3}{3!} \frac{\partial^3 f(x)}{\partial x^3} + \dots = \sum_{n=0}^{\infty} \frac{(\Delta x)^n}{n!} \frac{\partial^n f(x)}{\partial x^n}$$

Δx rozdíl mezi vzorky

- v případě obrazu je $\Delta x = 1$ (v případě že nás zajímá předchozí vzorek) nebo $\Delta x = -1$ (v případě že nás zajímá následující vzorek)
- pro $\Delta x = 1$

$$f(x + 1) = f(x) + \frac{\partial f(x)}{\partial x} + \frac{1}{2!} \frac{\partial^2 f(x)}{\partial x^2} + \frac{1}{3!} \frac{\partial^3 f(x)}{\partial x^3} + \dots = \sum_{n=0}^{\infty} \frac{1}{n!} \frac{\partial^n f(x)}{\partial x^n}$$

Příklad

Jak bude vypadat rozvoj pro $\Delta x = -1$?

Taylorova řada

- pro $\Delta x = -1$

$$f(x+1) = f(x) - \frac{\partial f(x)}{\partial x} + \frac{1}{2!} \frac{\partial^2 f(x)}{\partial x^2} - \frac{1}{3!} \frac{\partial^3 f(x)}{\partial x^3} + \dots = \sum_{n=0}^{\infty} \frac{(-1)^n}{n!} \frac{\partial^n f(x)}{\partial x^n}$$

- Pro rozdíl intenzit bereme jen pár prvních členů řady

- **1. derivace** = pouze lineární člen

- 3 směry rozdílu:

- **Dopředný rozdíl** = $\frac{\partial f(x)}{\partial x} = f'(x) = f(x+1) - f(x)$

- **Zpětný rozdíl** = $\frac{\partial f(x)}{\partial x} = f'(x) = f(x) - f(x-1)$

- **Centrální rozdíl** = $\frac{\partial f(x)}{\partial x} = f'(x) = \frac{f(x+1) - f(x-1)}{2}$

- Nepoužité členy řady představují chybu \rightarrow čím více členů, tím lepší aproximace
- Ukazuje se, že centrální rozdíl má lepší přesnost než dopředný nebo zpětný při stejném počtu členů

Taylorova řada

- **2. derivace** (centrální rozdíl) $\frac{\partial^2 f(x)}{\partial x^2} = f''(x) = f(x+1) - 2f(x) + f(x-1)$
- **3. derivace** (centrální rozdíl) $\frac{\partial^3 f(x)}{\partial x^3} = f'''(x) = \frac{f(x+2) - 2f(x+1) + 0f(x) + 2f(x-1) - f(x-2)}{2}$
- **4. derivace** (centrální rozdíl)
 $\frac{\partial^4 f(x)}{\partial x^4} = f^{(4)}(x) = f(x+2) - 4f(x+1) + 6f(x) - 4f(x-1) + f(x-2)$
- Pokud máme funkci dvou proměnných, každou proměnnou derivujeme zvlášť
- Například 2. derivace
 $\frac{\partial^2 f(x,y)}{\partial x^2} = f(x+1, y) - 2f(x, y) + f(x-1, y)$
 $\frac{\partial^2 f(x,y)}{\partial y^2} =$
- Z koeficientů u jednotlivých členů můžeme vytvořit filtr ($f(x, y)$ je ve středu)

Detekce izolovaných bodů

- Založeno na druhé derivaci

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

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

Příklad

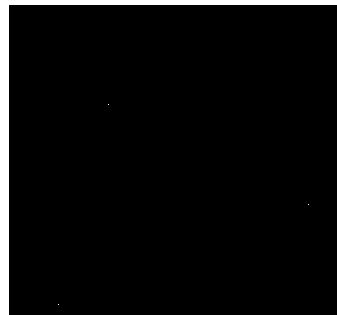
Odpovídá tento vztah nějakému filtru, který už známe?

- Dle síly odpovědi rozhodneme, zda se jedná o bod (použijeme nějakou prahovou hodnotu T)

$$g(x, y) = \begin{cases} 1 & \text{pokud } |Z(x, y)| > T \\ 0 & \text{jinak} \end{cases}$$

T prahová hodnota, Z je filtrovaný obraz a $g(x, y)$ je binární obrázek, kde 1 ukazují polohu izolovaných bodů

Detekce izolovaných bodů



Detekce linií

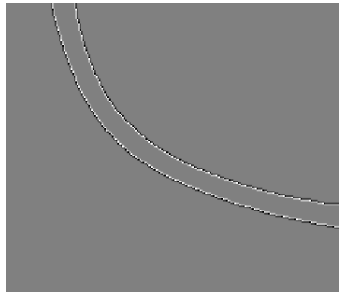
- Víme, že pomocí 2. derivace dostaneme tenčí linie, než první derivací

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

- Opět můžeme použít Laplaceův filtr

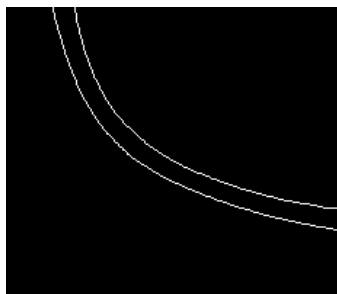
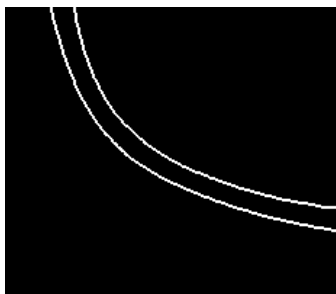
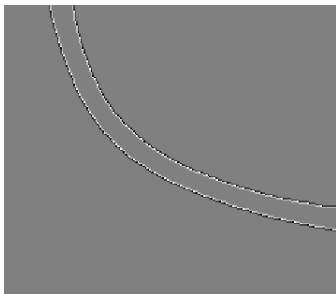
$$\nabla^2 f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y)$$

- Musíme se jen vypořádat s dvojitými liniemi



Detekce linií

- Když se negativních hodnot zbavíme absolutní hodnotou \rightarrow dvojité linie
- Většinou bereme v úvahu jen pozitivní hodnoty a ty prahujeme



Detekce linií

- Laplaceův filtr je nezávislý na směru linií
- V některých aplikacích chceme jen linie v určitém směru

Příklad

Linie v jakém směru budou detekovány pomocí následujících filtrů?

-1	-1	-1
2	2	2
-1	-1	-1

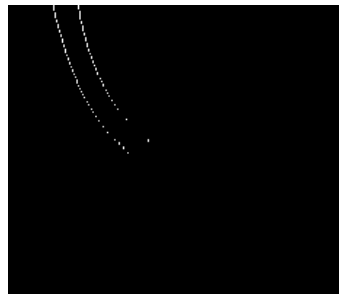
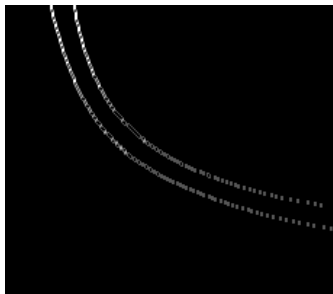
2	-1	-1
-1	2	-1
-1	-1	2

-1	2	-1
-1	2	-1
-1	2	-1

-1	-1	2
-1	2	-1
2	-1	-1

Detekce linií

- Zajímají nás svíslé hrany
- Ty mají větší odpověď na filtr než ostatní směry

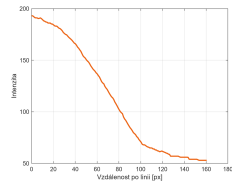
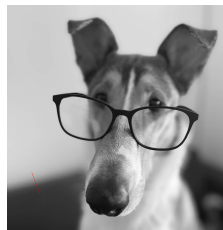
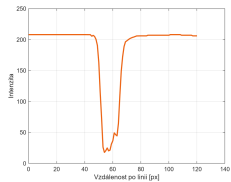
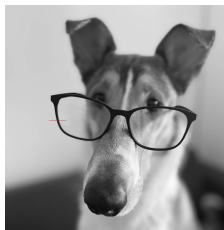


Detekce hran

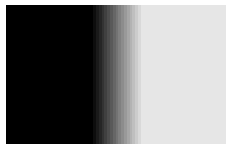
- Segmentace na základě změn v obraze
- Hranu můžeme modelovat různými způsoby dle jejich profilu intenzit:
 - **Step edges** (skokové hrany) = v rozmezí jednoho pixelu dojde k prudké změně intenzity
 - **Ramp edges** (hrany rampy) = chybí ostrý přechod
 - **Roof edges** (hrany ve tvaru střechy)
- V praxi se s ideálními hranami nesetkáme – rozmazání, šum
- Většinou se v obrázcích objevují hrany různých typů



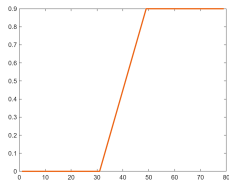
Detekce hran



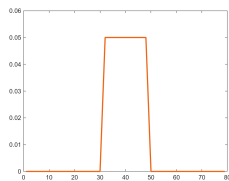
Detekce hran



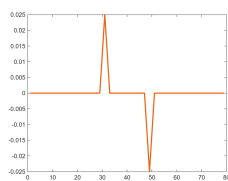
Přechod



Profil



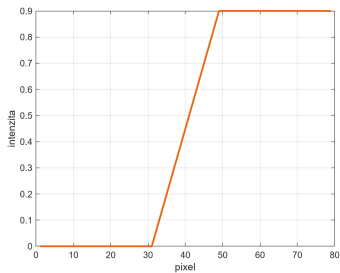
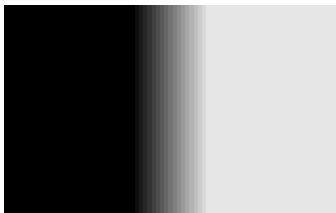
1. derivace



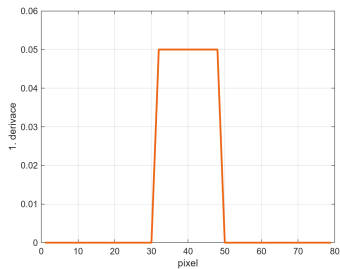
2. derivace

Detekce hran

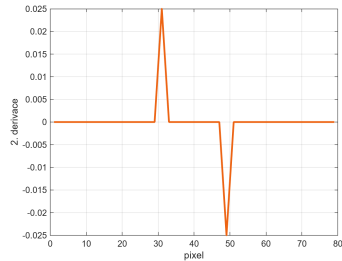
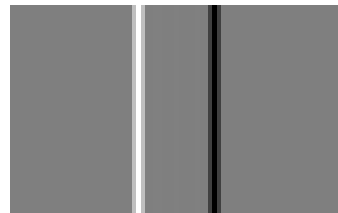
Bez šumu



Přechod



1. derivace



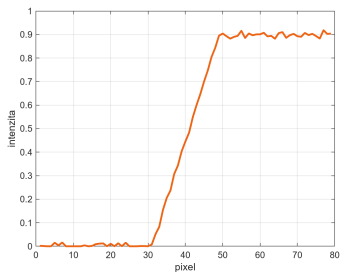
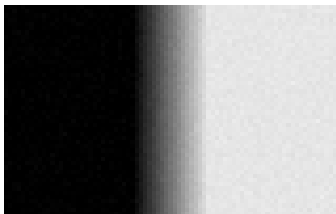
2. derivace

Detekce hran

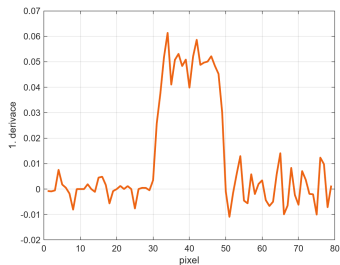
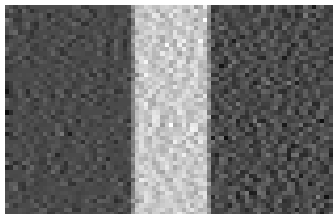
- Z magnitudy 1. derivace můžeme určit, zda se hrana v obrázku nachází
- Znaménko 2. derivace určuje, zda je pixel hrany na světlé nebo tmavé straně hrany
- 2. derivace pro každou hranu produkuje 2 peaky
- Když peaky spojíme, kde spojnice protíná osu x , je střed hrany
- Hranové modely jsou bez šumu, dále jak šum ovlivňuje derivace

Detekce hran

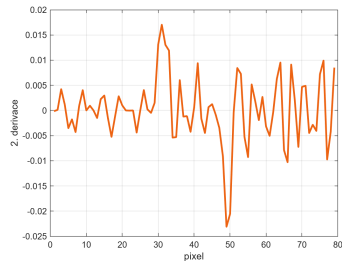
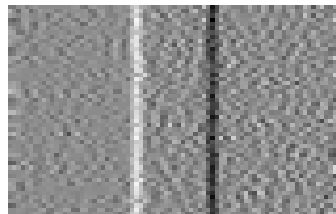
Gauss, $\sigma^2 = 0.0001$



Přechod



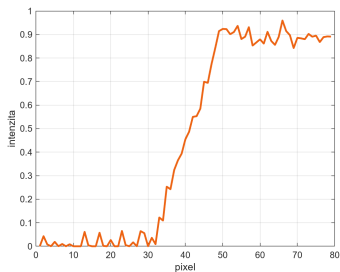
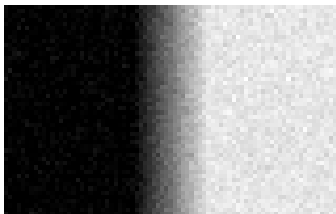
1. derivace



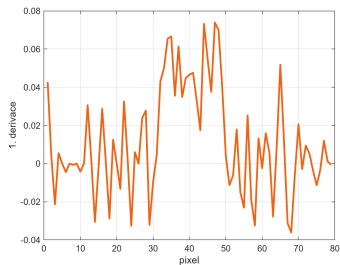
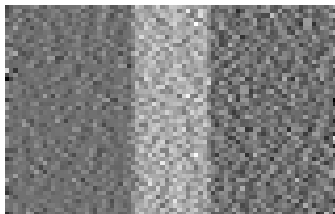
2. derivace

Detekce hran

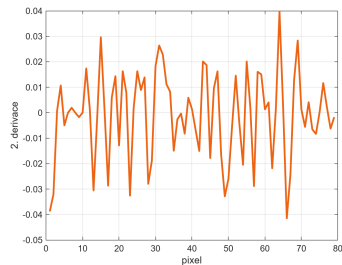
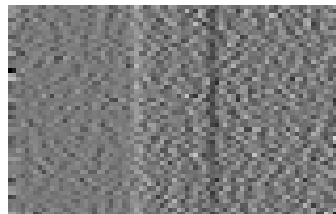
Gauss, $\sigma^2 = 0.001$



Přechod



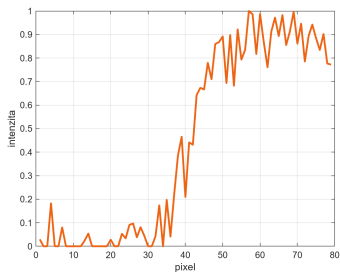
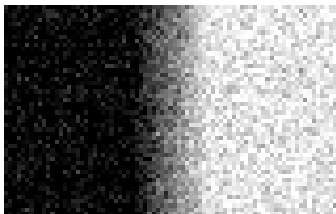
1. derivace



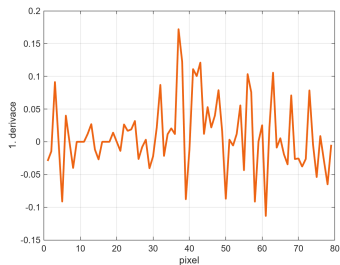
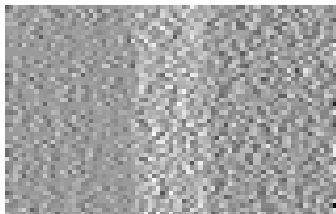
2. derivace

Detekce hran

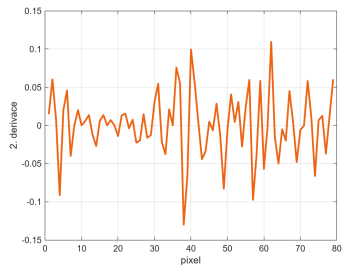
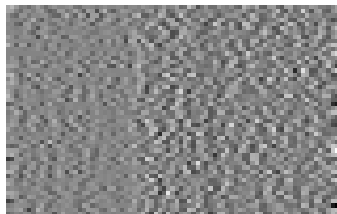
Gauss, $\sigma^2 = 0.01$



Přechod



1. derivace



2. derivace

Detekce hran

- Obecně se chceme nejprve zbavit šumu (vyhlazení)
- Detekce hranových bodů (kandidátů, kteří tvoří hranu):
 - Základní metody: gradient, gradient kombinovaný s prahováním
 - Pokročilejší metody: Marr-Hildreth detektor, Canny detektor
- Vybrání těch kandidátů, kteří tvoří hranu

Detekce hran

Gradient obrázku

- Gradient je vektor, který určuje nejen sílu hrany, ale i její směr

$$\nabla f(x, y) = \text{grad} [f(x, y)] = \begin{bmatrix} g_x(x, y) \\ g_y(x, y) \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

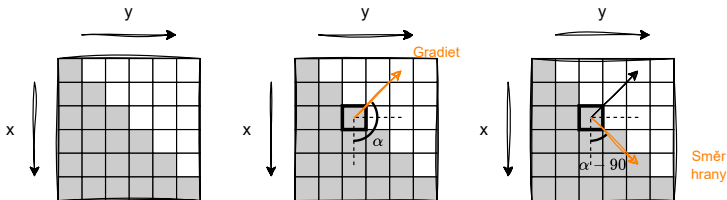
- $\nabla f(x, y)$ je obrázek vektorů (každému pixelu jsou přiřazeny 2 hodnoty)

- Magnituda vektoru (**gradient image**)

$$M(x, y) = \|\nabla f(x, y)\| = \sqrt{g_x^2(x, y) + g_y^2(x, y)}$$

- Úhel vektoru gradientu

$$\alpha(x, y) = \tan^{-1} \left[\frac{g_y(x, y)}{g_x(x, y)} \right]$$

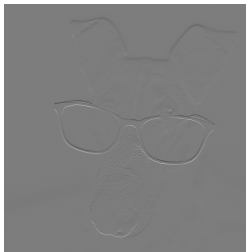


Detekce hran

Gradient obrázku



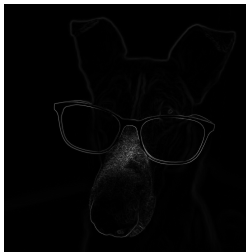
Originál



Gradient x



Gradient y



Magnituda



Úhly

Detekce hran

Gradient obrázku

- Výpočet parciálních derivací v obou směrech, nejčastěji se používá dopředný nebo centrální rozdíl

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

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

- Můžeme to aproximovat pomocí filtrování Sobelovým filtrem nebo filtrem Prewittové
- Magnitudu můžeme aproximovat součtem absolutních hodnot gradientů

$$M(x, y) \approx |g_x(x, y)| + |g_y(x, y)|$$

Příklad

Jak vypadá Sobelův filtr?

Jak vypadá filtr Prewittové?

Detekce hran

Gradient obrázku

Příklad

Jak vypadá Sobelův filtr?

Jak vypadá filtr Prewittové?

Prewittové filtr

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Sobelův filtr

-1	-2	-1
0	0	0
1	2	1

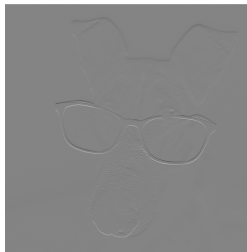
-1	0	1
-2	0	2
-1	0	1

Detekce hran

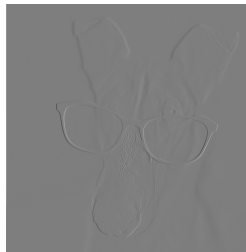
Gradient obrázku – aproximace Sobel



Originál



Gradient x



Gradient y



Magnituda

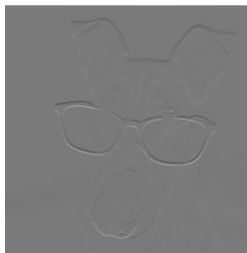
Detekce hran

Gradient obrázku

- Fotka z příkladu má velké rozlišení, objevují se i malé detaily (ty jsou většinou nežádoucí a chovají se jako šum)
- Před hledáním hran se obraz vyhlazuje



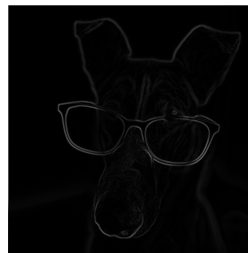
Originál



Gradient x



Gradient y

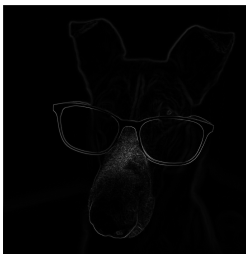


Magnituda

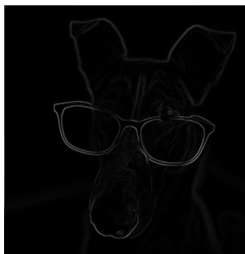
Detekce hran

Gradient obrázku + prahování

- Jiný přístup – naprahujeme gradient (magnitudu)
- Některé hrany jsou ale rozpojené
- Případně kombinujeme vyhlazení i prahování



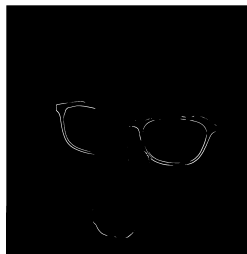
Originál



Vyhlazení



Prahování



Vyhlazení + prahování

Pokročilé metody detekce hran

- Metody zmíněné dříve jsou založené na filtrování
- Neberou v úvahu charakteristiky hran a šumu
- Pokročilé metody jsou založené na předchozích, ale charakteristiky hran berou v úvahu
- Příklad:
 - Mar-Hildreth hranový detektor
 - Canny hranový detektor

Pokročilé metody detekce hran

Mar-Hildreth hranový detektor

■ Hlavní myšlenky:

- změny v obraze nejsou nezávislé na jeho velikosti, je nutno vzít operátory různých velikostí
- prudké změny je možné detekovat pomocí první, nebo druhé derivace

■ Operátor, který se použije, by měl aproximovat první, nebo druhou derivaci

■ Měl by být škálovatelný (velký by měl detekovat rozmazané hrany, malý ostré)

■ Výsledný operátor: $\nabla^2 G$

∇^2 je aproximována Laplaceovým operátorem

G je Gaussovský filtr $e^{-\frac{x^2+y^2}{2\sigma^2}}$

σ je občas nazývá prostorová konstanta

$$\begin{aligned}\nabla^2 G(x, y) &= \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2} = \frac{\partial}{\partial x} \left(\frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right) + \frac{\partial}{\partial y} \left(\frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right) = \\ &= \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right) e^{-\frac{x^2+y^2}{2\sigma^2}} + \left(\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right) e^{-\frac{x^2+y^2}{2\sigma^2}}\end{aligned}$$

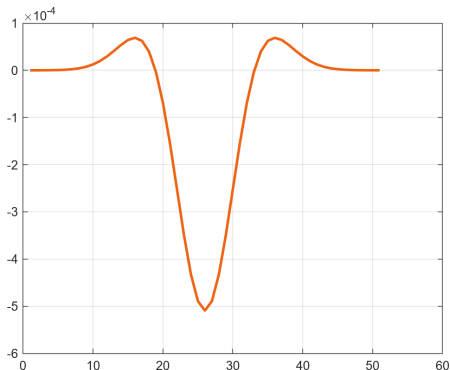
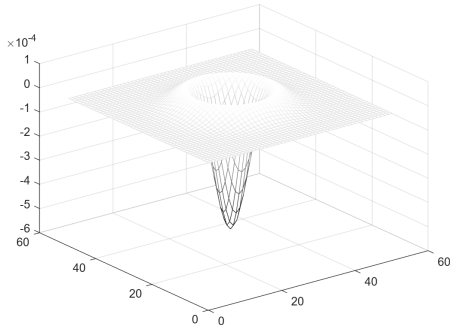
Pokročilé metody detekce hran

Mar-Hildreth hranový detektor

- Celkem:

$$\nabla^2 G(x, y) = \left(\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right) e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

- Tento operátor se nazývá **Laplacian of Gaussian (LoG)**
Díky svému tvaru se mu občas říká Mexican hat operator
- Hrany jsou detekovány jako **zero-crossings** druhé derivace



Pokročilé metody detekce hran

Mar-Hildreth hranový detektor

$\sigma = 5, n = 10$



LoG



LoG > 0



LoG > 5 % max

Pokročilé metody detekce hran

Mar-Hildreth hranový detektor

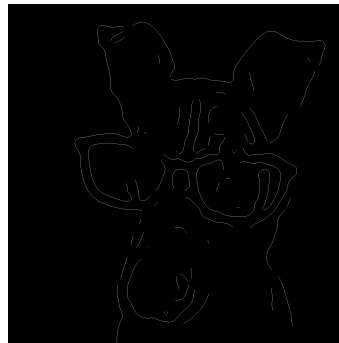
$\sigma = 20, n = 50$



LoG



LoG > 0



LoG > 5 % max

Pokročilé metody detekce hran

Canny hranový detektor

- Je sice složitější, ale zdaleka dává nejlepší výsledky
- Hlavní úkoly:
 - Malá chybovost: Všechny hrany by měly být nalezeny a neměly by se vyskytovat žádné falešné odezvy
 - Hrany by měly být dobře lokalizovány: Detekované hrany by měly být blízko skutečných hran
 - Jednobodové hrany: Hrany by měly být úzké (počet lokálních maxim by mělo být kolem hrany minimálně)
- Optimalizační přístup k detekci hran
- Kombinuje:
 - Gaussovské vyhlazení
 - gradient
 - potlačení nemaxim
 - hysterezní prahování

Detekce hran

Canny hranový detektor

- Magnituda směrový vektor gradientu Gaussovského vyhlazení

$$f_s(x, y) = G(x, y) \star f(x, y)$$

$$M_s(x, y) = \|\nabla f_s(x, y)\| = \sqrt{g_x^2(x, y) + g_y^2(x, y)}$$

$$\alpha(x, y) = \tan^{-1} \left[\frac{g_y(x, y)}{g_x(x, y)} \right]$$

$$g_x(x, y) = \frac{\partial f_s(x, y)}{\partial x}$$

$$g_y(x, y) = \frac{\partial f_s(x, y)}{\partial y}$$

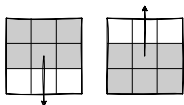
- Gradientní obraz typicky obsahuje široké pásy kolem lokálních maxim
- Chceme tyto pásy ztenčit – non-maxima suppression

Detekce hran

Canny hranový detektor

■ Ztenčení

- Specifikujeme počet diskrétních orientací normály hrany (pro 3×3 okolí = horizontální, vertikální a dva diagonální směry každý se dvěma orientacemi)



- Všechny možné orientace směrnic kvantifikujeme do 4 rozsahů

Příklad

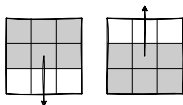
Jaké budou tyto rozsahy?

Detekce hran

Canny hranový detektor

■ Ztenčení

- Specifikujeme počet diskrétních orientací normály hrany (pro 3×3 okolí = horizontální, vertikální a dva diagonální směry každý se dvěma orientacemi)



- Všechny možné orientace směrnic kvantifikujeme do 4 rozsahů

Příklad

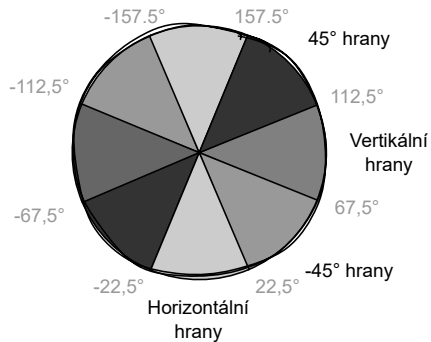
Jaké budou tyto rozsahy?

Detekce hran

Canny hranový detektor

Příklad

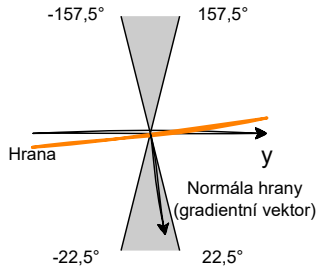
Jaké budou tyto rozsahy?



Detekce hran

Canny hranový detektor

- Ze směrnice normály odhadneme směr hrany



Detekce hran

Canny hranový detektor

- Pro každý pixel
 - Určíme jeden za 4 směrů d_k
 - K označuje hodnotu $\|\nabla f_s\|$ v bodě (x, y) . Pokud je K menší než $\|\nabla f_s\|$ nějakého sousedního pixelu ve směru d_k , nastavíme $g_N(x, y) = 0$ (potlačení), jinak necháme $g_N(x, y) = K$
- Nakonec g_N naprahujeme, tak, abychom potlačili falešné hrany (většinou používáme dva prahy T_H a T_L
$$g = T_L \leq g_N \leq T_H$$

Detekce hran

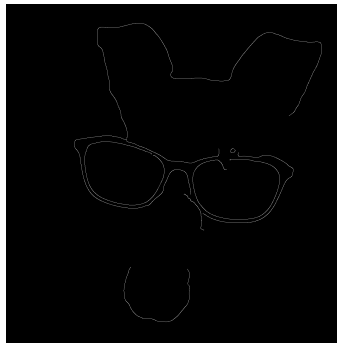
Canny hranový detektor



Originál



$T_L = 0, T_H = 0.01$



$T_L = 0, T_H = 0.5$

Spojování hranových bodů

- Detektory vrací body, které leží na nějaké hraně
- Kvůli například šumu nebo špatnému osvětlení se ale mohou objevit mezery mezi nimi
- Algoritmy pro jejich **spojování** v hrany, nebo okraje oblastí:
 - lokální přístup – potřebujeme znát informace o hranových bodech v okolí
 - globální přístup

Spojování hranových bodů

Lokální přístup

- Analyzujeme hodnoty pixelů v nějakém okolí okolo každého bodu (x, y) , který jsme označili za hranový
- Všechny body, které mají podobnou charakteristiku (dle zvoleného kritéria) jsou spojeny do hrany
- Nejčastěji se používá podobnost magnitudy gradientu a jeho směr
$$|M(s, t) - M(x, y)| \leq E$$
$$|\alpha(s, t) - \alpha(x, y)| \leq A$$

(s, t) jsou souřadnice bodu v okolí
 E a A jsou kladné prahové hodnoty
- Obecně je tento postup časově náročný

Spojování hranových bodů

Lokální přístup

1 Spočítáme magnitudu a úhly gradientu pro celý obrázek

2 Vytvoř obrázek $g(x, y)$ takto

$$g(x, y) = \begin{cases} 1 & \text{pokud } M(x, y) > T_M \text{ a } \alpha(x, y) = A \pm T_A \\ 0 & \text{jinak} \end{cases}$$

T_M je prahová hodnota, A definovaný směr, T_A prahová hodnota o kolik se může směr změnit

3 Postupujeme přes všechny řádky g a doplňujeme 1 všude, kde je menší mezera než nějaká vzdálenost L (každý řádek zpracováváme zvlášť)

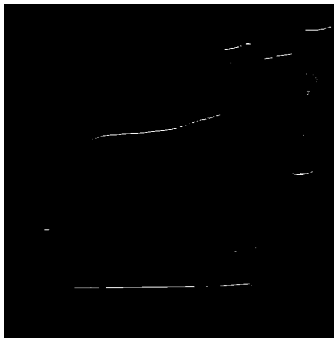
4 k detekci mezer mezi ostatními směry, otočíme g o tento úhel, opakujeme krok 3 a orotujeme výsledek zpět

Spojování hranových bodů

Lokální přístup



Originál



Vodorovné



Svislé

Spojování hranových bodů

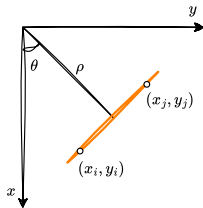
Globální přístup

- Nemáme další informace o struktuře obrazu
- Většinou vybíráme z množiny kandidátů body, které leží na nějaké specifikované křivce
- Máme n bodů, které považujeme za kandidáty, hledáme jejich podmnožinu tak, aby body ležely na jedné přímce
- **Naivní přístup**: Pro každou dvojici bodů spočítáme přímku a rozhodneme, které z ostatních bodů leží blízko této přímky
- Tento přístup je časově náročný – hledání $n(n-1)/2$ (přibližně n^2) přímek, a pro ně $n(n(n-1))/2$ (n^3) porovnání, zda je bod blízko.
- Lepší přístup = **Houghova transformace**

Spojování hranových bodů

Houghova transformace

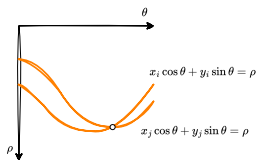
- Hledání přímek
- Přímka v kartézských souřadnicích: $y = ax + b$
- Problém:
 - vertikální přímky $\rightarrow a \rightarrow \infty$
 - nestabilní parametrizace
- Řešení: **parametrický prostor**
 - $\rho = x \cos \theta + y \sin \theta$
 - ρ = vzdálenost přímky od počátku
 - θ = úhel normály



Spojování hranových bodů

Houghova transformace

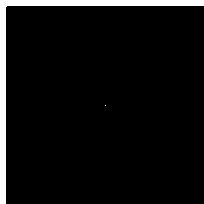
- Každá přímka odpovídá bodu (ρ, θ)
- Každý obrazový bod odpovídá sinusoidě v parametrovém prostoru



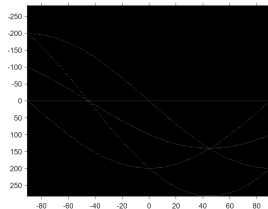
- Rozdělíme prostor na diskrétní množinu úhlů θ a vzdáleností ρ a použijeme pomocný akumulátor A
- **Algoritmus:**
 - Pro každý hranový bod (x_i, y_i) :
 - pro všechna θ
 - spočítáme $\rho = x_i \cos \theta + y_i \sin \theta$
 - inkrementujeme akumulátor $A(\rho, \theta)$
 - Přímky v obraze \rightarrow maxima v akumulátoru
 - Detekce přímek = hledání lokálních maxim

Spojování hranových bodů

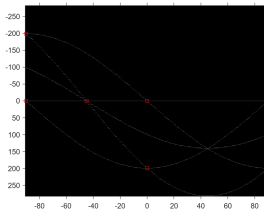
Houghova transformace



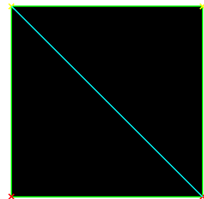
Obrázek



Sinusovky



Maxima



Vykreslené přímky

Spojování hranových bodů

Houghova transformace



Spojování hranových bodů

Houghova transformace

- Výhody:
 - robustní vůči šumu
 - detekuje i částečné přímky
 - globální metoda
- Nevýhody:
 - výpočetně náročná
 - citlivá na volbu diskretizace
 - detekuje nekonečné přímky
- Houghova transformace je obecně použitelná pro **parametrizovatelné křivky**
- Princip:
 - definujeme parametrický model křivky
 - každý bod hlasuje v prostoru parametrů
 - maxima akumulátoru odpovídají nalezeným objektům
- Dimenze akumulátoru odpovídá počtu parametrů modelu
- Čím více parametrů → tím vyšší výpočetní náročnost

Spojování hranových bodů

Houghova transformace – kružnice

- Rovnice kružnice:

$$(x - a)^2 + (y - b)^2 = r^2$$

(a, b) = střed

r = poloměr

- Parametrický prostor je 3D: (a, b, r)
- Každý hranový bod hlasuje pro všechny možné středy a poloměry
- Maxima akumulátoru odpovídají detekovaným kružnicím

Spojování hranových bodů

Houghova transformace

