

# **Metody přesné lokalizace hran**



## Obsah

<b>1</b>	<b>Úvod.....</b>	<b>4</b>
<b>2</b>	<b>Klasické hranové detektory .....</b>	<b>5</b>
2.1	Metody založené na první derivaci .....	5
2.1.1	Robertsův operátor .....	5
2.1.2	Prewittův operátor .....	5
2.1.3	Sobelův operátor .....	6
2.2	Metody založené na druhé derivaci.....	6
2.2.1	Algoritmus Marr-Hildreth .....	6
2.2.2	Haralickův algoritmus .....	8
2.3	Cannyho detektor.....	8
<b>3</b>	<b>Detektory se subpixelovou přesností.....</b>	<b>12</b>
3.1	Momentové metody.....	12
3.1.1	Použití gray-level momentů pro lokalizaci se subpixelovou přesností.....	12
3.1.2	Použití prostorových momentů pro lokalizaci se subpixelovou přesností .....	16
3.1.3	Zernikovy momenty.....	18
3.2	Aproximační metody .....	21
3.2.1	Aproximace hrany ERF funkcí .....	21
3.2.2	Aproximace hrany funkcí arcus tangens.....	24
3.2.3	Aproximace gradientního obrazu Gaussovou funkcí.....	27
3.3	Interpolační metody .....	29
3.3.1	Bilineární, bikvadratická a bikubická interpolace .....	29
3.3.2	Kubická konvoluce .....	30
3.3.3	ENO interpolace.....	31
3.3.4	Hranou řízená interpolace .....	33
3.4	Ostatní metody .....	36
3.4.1	Lokalizace hran za použití efektu částečné plochy pixelu (partial area effect) .....	36
<b>4</b>	<b>Závěr .....</b>	<b>39</b>
<b>5</b>	<b>Reference .....</b>	<b>41</b>

## 1 Úvod

V této práci bych rád shrnul současný stav v oblasti detekce a lokalizace hran v obraze a to především metody přesné lokalizace se subpixelovou přesností. Detekce hran je jednou ze základních a současně nejstarších úloh zpracování obrazu a počítačového vidění. Úspěšné zvládnutí tohoto problému je nutné pro řešení řady dalších úloh, jako jsou např. segmentace či extrakce kontur.

Nároky na kvalitu a přesnost dat získaných z obrazu jsou v řadě aplikací velmi vysoké a v současnosti je spíše výjimkou, aby detekce hrany s přesností jednoho celého pixelu dostačovala. Původní hranové detektory byly proto postupně nahrazovány sofistikovanějšími metodami, které byly schopny kromě lepší (rozumějte důvěryhodnější) detekce hran i výrazně lepší lokalizace s přesností desetin až setin pixelu.

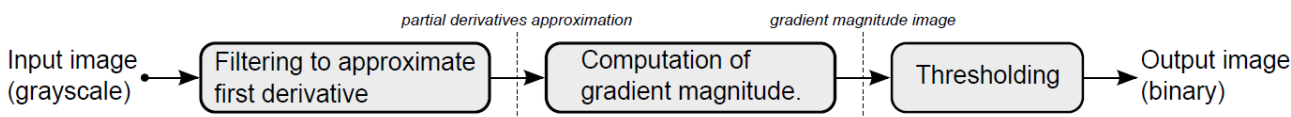
Text tohoto článku je členěn do kapitol. V kapitole 2 je uveden stručný přehled klasických diferenciálních hranových detektorů založených na hledání lokálních maxim první derivace obrazové funkce, případně detekce průchodů nulou druhé derivace obrazové funkce. Popsány jsou rovněž sofistikovanější algoritmy Marr-Hildrethová a Maralickův a nejčastěji používaný Cannyho detektor. Kapitola 3 se věnuje metodám lokalizace hran se subpixelovou přesností. Tyto metody lze rozdělit podle principu detekce zhruba do tří skupin – metody aproximační, momentové a interpolační. V některých případech mohou algoritmy využívat principy více metod současně. Při zpracování textu vycházel autor z mnoha zdrojů, na něž je v odpovídajících místech odkazováno. V úvodu stojí za zmínku práce [1], která přehledně shrnuje state-of-the-art metody detekce kontur. Obsahuje téměř 300 odkazů na jednotlivá témata a na rozdíl od jednoduché detekce hran uvažuje i rozhraní mezi oblastmi s různou texturou. V kapitole 4 (Závěr) jsou popsány metody shrnuty a uvedeny jejich výhody a nevýhody ve srovnání s ostatními.

## 2 Klasické hranové detektory

Klasické hranové detektory vznikly v sedmdesátých letech 20. století jako první odezva na potřebu detekce hran v obraze. Přehledně jsou popsány v mnoha učebnicích počítačového vidění, nebo např. v [2]. Hrana, jakožto oblast s náhlou změnou intenzity obrazové funkce se vyznačuje vysokou hodnotou gradientu. Právě na detekci těchto vysokých hodnot cílí první diferenciální operátory – Robertsův, Prewittův a Sobelův. Další generací jsou pak detektory založené na detekci průchodu nulou druhé derivace obrazové funkce (tj. lokální maxima první derivace). Mezi ně patří popisované Marr-Hildrethovy a Maralickovy algoritmy.

### 2.1 Metody založené na první derivaci

Všechny metody založené na první derivaci obrazové funkce pracují s operátory aproximujícími hodnotu derivace spojité obrazové funkce v okolí každého bodu obrazu a to za pomoci diskretizovaných a kvantizovaných hodnot této funkce. Liší se právě způsobem aproximace parciálních derivací. Následuje výpočet gradientu z parciálních derivací a prahování (viz Obr. 1). V některých případech je na závěr zařazena skeletizace, aby se vyloučila vícenásobná odezva na hranu a našla se nejpravděpodobnější poloha hrany.



Obr. 1 Princip práce detekčních algoritmů založených na první derivaci

#### 2.1.1 Robertsův operátor

Robertsův operátor(-y) aproximuje první parciální derivaci diferencí, resp. prvním členem Taylorova rozvoje obrazové funkce. Nepočítá však parciální derivace v základních směrech os  $x$  a  $y$ , ale ve směrech diagonálních.

-1	0
0	1

0	-1
1	0

Obr. 2 Robertsovy masky pro 2 ortogonální směry

#### 2.1.2 Operátory podle Prewittové

Masky velikosti  $2 \times 2$  jsou sice konceptuálně i výpočetně jednoduché, ale mají jednu velkou nevýhodu – nejsou symetrické vzhledem ke zkoumanému centrálnímu bodu a tudíž zanáší do polohy detekované hrany offset. Požadavek na symetričnost detekovaných hran je velmi důležitý, z toho důvodu jsou téměř vždy používány masky s lichým počtem řádků a sloupců. Tyto masky využívají větší okolí bodu k odhadu velikosti gradientu a jsou tudíž méně náchylné na šum, avšak za cenu lokalizace s nižší přesností. Prewittová použila nejmenší možnou velikost masky s lichým počtem řádků/sloupců o velikosti  $3 \times 3$ . Normalizované verze masek jsou následující:

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

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

Obr. 3 Masky podle Prewittové pro detekci hrany ve vodorovném a svislém směru

Na první masku lze nahlédnout jako na konvoluci dvou filtrů – difference ve svislém směru s maskou  $[-1 \ 0 \ 1]^T$  a plovoucí průměr ve vodorovném směru s maskou  $[1 \ 1 \ 1]$ . Tyto operátory lze tudíž interpretovat jako postupnou aplikaci difference ve svislém směru a nízkofrekvenční propusti ve vodorovném směru. Obdobnou úvahu lze provést pro druhou masku.

### 2.1.3 Sobelův operátor

Sobelovy operátory jsou mírnou modifikací předchozích operátorů navržených Prewittovou, kladou pouze větší váhu na centrální koeficienty masky:

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

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

Obr. 4 Masky Sobelových operátorů pro detekci hrany ve vodorovném a svislém směru

Lze na ně nahlížet jako na konvoluci difference s maskou  $[-1 \ 0 \ 1]^T$  v jednom směru a vyhlazovacího filtru s koeficienty  $[1 \ 2 \ 1]$  ve druhém ortogonálním směru. Tento filtr má blíže Gaussově okénku než předchozí prostý aritmetický průměr s konstantními váhami a tudíž nabízí lepší filtrační a lokalizační vlastnosti, než operátory navržené Prewittovou. Za použití obecného algoritmu pro konvoluci obrazu s maskou je výpočetní náročnost obou operátorů identická. Navíc je ale možné pro tento konkrétní operátor realizovat optimalizované verze algoritmů – při konvoluci podle Sobelových masek se používá pouze násobení a dělení mocninami čísla 2 (konkrétně 2 a 8), což lze nahradit rychlejšími operacemi bitového posunu vlevo či vpravo a výpočet tak dále zrychlit. Použití Sobelových operátorů se tudíž preferuje vždy.

## 2.2 Metody založené na druhé derivaci

Algoritmy detekce hran založené na druhé derivaci hledají průchody nulou druhé derivace obrazové funkce – tj. inflexní body derivace první. V případě, že se nejedná o stacionární bod (obrazová funkce je v okolí zkoumaného bodu konstantní), se pak jedná o hranu. Výhodou těchto algoritmů je jednoznačná odezva na hranu – v případě „rozmazaných“ hran zabírajících více pixelů je hrana indikována v místě největší změny.

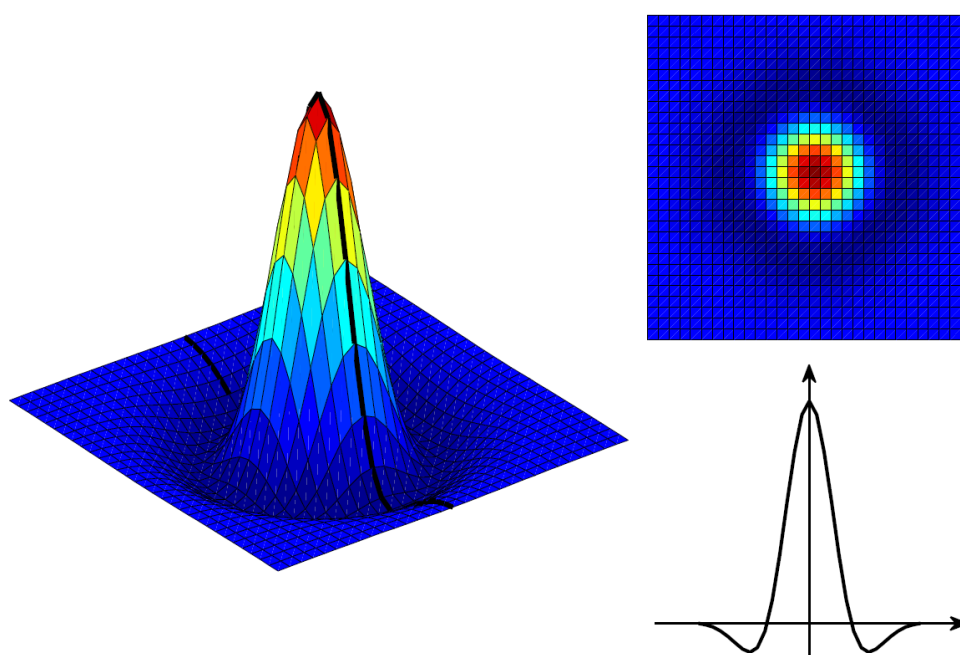
### 2.2.1 Algoritmus Marr-Hildreth

Algoritmus dvojice Marr-Hildrethová byl publikován v [3]. Podle něj se nejprve vyhledá obraz konvolucí s Gaussovou 2D maskou – jedná se o nízkofrekvenční propust, filtr s koeficienty Gaussova 2D okénka. Tento krok je nutný, neboť diferenciální operátory jsou velmi citlivé na šum. Následuje konvoluce s maskou reprezentující aproximaci Laplaceova operátoru:

1	1	1
1	-8	1
1	1	1

Obr. 5 Diskretizovaná aproximace Laplaceova operátoru

Jelikož se jedná o lineární operace, je možné a vhodné sloučit je do jedné – aplikací Laplaceova operátoru na Gaussovu funkci získáme tzv. LoG funkci (Laplacian of Gaussian) a jejím vzorkováním v příslušných bodech pak LoG masku. S pomocí této masky lze pak obraz zpracovat v jednom průchodu. Při návrhu Gaussovy či LoG masky je třeba správně zvolit jejich velikost tak, aby obsahovala všechny významné koeficienty. Podle [2] se volí pro velikost Gaussovy masky nejbližší liché číslo větší než  $6\sigma$ . Pro LoG je nutná maska cca o 18% větší.



Obr. 6 LoG maska pro  $\sigma = 4$  a  $n = 31$

V následujícím kroku je nutné detekovat body, v nichž dochází k průchodu nulou u konvolvaného obrazu  $L(i,j)$ . Znamená to, že body na opačných stranách od zkoumaného bodu se liší znaménkem. Testovány jsou jak horizontálně a vertikálně protilehlé body, tak i diagonální směry. Aby se vyloučila detekce nevýrazných hran, posiluje se tato podmínka dalším testem – vzdálenost jasu protilehlých bodů musí být větší, než předem daný práh  $t_h$ . Práh  $t_h$  lze stanovit jako pevnou konstantu, nebo ho lze určit jako poměrnou část z maximální hodnoty, kterou nabývá funkce  $L(i,j)$  nad obrazem. Test pro jeden ze 4 směrů (horizontální) je následující, ostatní podmínky lze sestavit analogicky:

$$\text{sign}(L(i-1,j)) \neq \text{sign}(L(i+1,j)) \ \& \ |L(i-1,j) - L(i+1,j)| > t_h \quad (1)$$

Body vyhovující těmto podmínkám jsou označeny za hrany.

### 2.2.2 Haralickův algoritmus

R. M. Haralick publikoval svůj algoritmus v [4]. Hlavní myšlenka jeho algoritmu je shodná s předchozím – detekovat průchody nulou druhé derivace obrazové funkce. Nicméně Haralick lokálně aproximuje obrazovou funkci bikubickou polynomiální plochou:

$$f(x, y) = k_1 + k_2(x-x_0) + k_3(y-y_0) + k_4(x-x_0)^2 + k_5(x-x_0)(y-y_0) + k_6(y-y_0)^2 + k_7(x-x_0)^3 + k_8(x-x_0)^2(y-y_0) + k_9(x-x_0)(y-y_0)^2 + k_{10}(y-y_0)^3 \quad (2)$$

u které následně určuje druhou derivaci (a její klesající průchod nulou) analyticky. Aby bylo možné jednoznačně určit koeficienty  $k_1 - k_{10}$ , je třeba vzít v potaz 10 a více bodů obrazové funkce. Haralick sestavuje 25 rovnic pro okolí  $5 \times 5$  a koeficienty pak určuje metodou nejmenších čtverců. Soustavu algebraických rovnic lze po úpravách vyjádřit jako maticové násobení předpočítané matice s hodnotami obrazové funkce v okolí zkoumaného bodu. (Podrobnosti viz [4]). Zajímavé je, že tato předpočítaná matice je konstantní pro všechny body obrazu a koeficienty  $k_1 - k_{10}$  tak lze vyjádřit konvolucí předpočítaných masek velikosti  $5 \times 5$  s okolím zkoumaného bodu.

Haralick následně převádí polynom substitucí do polárních souřadnic a zkoumá, zda je jeho druhá derivace pro nějaký bod okolí nulová a třetí derivace záporná (klesající průchod nulou druhé derivace). Splnění těchto podmínek indikuje bod hrany. V původní práci je detektor použit s pixelovou přesností, nicméně principiálně jej lze zařadit k aproximativním metodám subpixelové detekce hran popsanych níže a po mírné modifikaci jej lze použít i pro subpixelovou lokalizaci.

### 2.3 Cannyho detektor

Cannyho detektor je sofistikovaným detektorem, který je tak, jak byl prvně publikován v [5] řazen mezi klasické hranové detektory, nicméně s mírnými modifikacemi jej lze použít i pro subpixelovou lokalizaci [7]. K problému detekce hrany je přistupováno jako k optimalizační úloze. Canny stanovuje 3 kritéria optimálního hranového detektoru:

1. **spolehlivá detekce** – detektor by měl s nízkou chybou detekovat hranu v místech, kde se opravdu nachází a naopak pravděpodobnost false-positives, tj. detekce hrany v místě, kde se nenachází, by měla být co nejnižší
2. **dobrá lokalizace** – body označené jako hrana by měly být co nejbližší ke středu skutečné hrany
3. **jednoznačná odezva** – na jednu hranu by měl detektor reagovat pouze v jednom místě. Tato podmínka je implicitně obsažena již v 2. kritériu, neboť pokud detektor reaguje na dvou místech, je jedna z odezev chybná – je od středu hrany dále. Nicméně požadavek je explicitně přidán z důvodů snazší matematické formulace.

V odvození detektoru začíná s 1D případem a předpokládá skokovou hranu v počátku s aditivním bílým šumem. Zásadním úkolem je formulovat předchozí intuitivní kritéria matematicky a následně získané funkcionály optimalizovat. První kritérium v podstatě koresponduje poměru signál-šum. Označme  $f(x)$  impulsní odezvu filtru,  $G(x)$  skokovou funkci hrany lokalizovanou



v počátku, interval  $\langle -W; W \rangle$  oblast, kde je odezva filtru nenulová a  $n_0$  střední hodnota čtverce amplitudy šumu. Potom odezva filtru na tento skok je:

$$H_G = \int_{-W}^{+W} G(-x)f(x)dx \quad (3)$$

odezva na šum samotný:

$$H_n = n_0 \sqrt{\int_{-W}^{+W} f^2(x)dx} \quad (4)$$

a poměr signál-šum:

$$SNR = \frac{\left| \int_{-W}^{+W} G(-x)f(x)dx \right|}{n_0 \sqrt{\int_{-W}^{+W} f^2(x)dx}} \quad (5)$$

Ze Schwarzovy nerovnosti lze ukázat, že tento poměr je shora omezen členem:

$$n_0^{-1} \sqrt{\int_{-W}^{+W} G^2(x)dx} \quad (6)$$

který bude nadále předmětem optimalizace.

Obdobně je tomu u druhého kritéria (kvalita lokalizace) – zde Canny zvolil jako měřítko převrácenou střední hodnotu vzdálenosti detekované hrany od hrany skutečné. Odvození vychází z faktu, že hranu detekujeme v místě lokálního maxima odezvy filtru – první derivace odezvy je zde tudíž nulová:

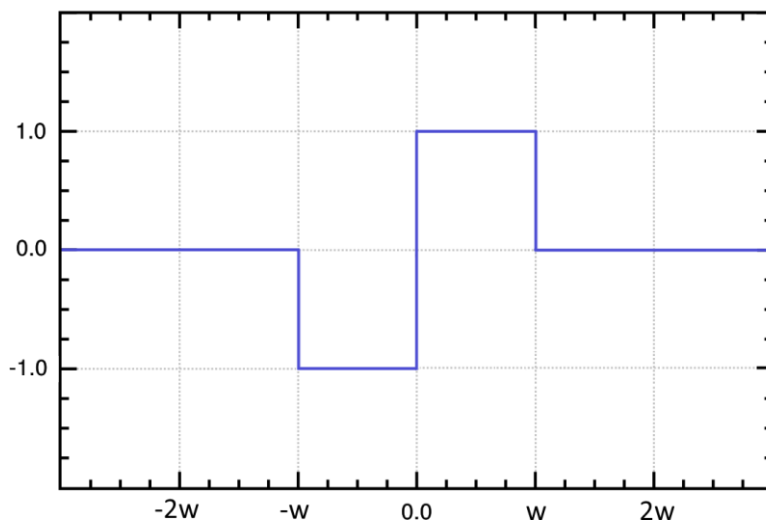
$$L = \frac{\left| \int_{-W}^{+W} G'(-x)f(x)dx \right|}{n_0 \sqrt{\int_{-W}^{+W} f'^2(x)dx}} \quad (7)$$

Pomocí Schwarzovy nerovnosti odvozená horní mez tohoto výrazu je pak předmětem optimalizace lokalizačního kritéria:

$$n_0^{-1} \sqrt{\int_{-W}^{+W} G'^2(x)dx} \quad (8)$$

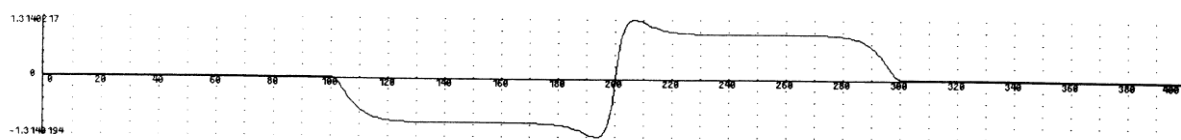
Výsledkem optimalizace těchto dvou funkcí je funkce záporného a následně kladného obdélníkového pulsu (Obr. 7). Nicméně tento filtr má velmi široké pásmo a proto u zašuměných hran generuje dodatečná maxima (falešné hrany). Z toho důvodu se přidává třetí podmínka,

vyžadující jednoznačnou odezvu na hranu – omezení na počet odezev v okolí jedné hrany, tj. na počet lokálních maxim filtrovaného signálu v této oblasti.

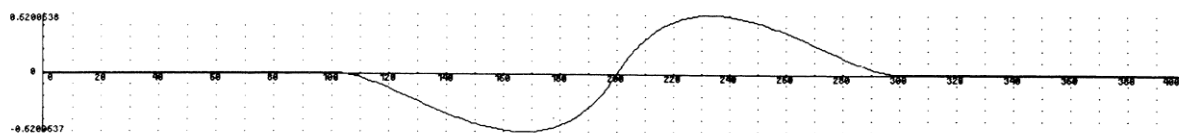


Obr. 7 Optimální filtr pro detekci a lokalizaci nezašuměných hran

Odvození funkcionálu k následné optimalizaci se odvíjí od střední hodnoty vzdálenosti maxim v bílém šumu, podrobnosti viz [5]. Analytické řešení optimalizační úlohy je velmi obtížné až nemožné (a to i pro nejjednodušší modely hran). Canny provádí analytický výpočet pouze pro skokovou hranu v 1D, pro ostatní tvary hran navrhuje numerické řešení. Ideální filtr má tvar superpozice několika komplexních sinusoid, hodnota parametrů této funkce závisí na zvoleném prostorovém měřítku (míře potlačení blízkých hran) a pro vzdálenost maxim 1 a více pixelů se zhruba podobá derivaci Gaussovy funkce.



Obr. 8 Optimální filtr pro detekci skokové hrany,  $x_{max} = 0.15$  ( $x_{max}$  značí průměrnou vzdálenost maxim v odezvě filtru na bílý šum)



Obr. 9 Optimální filtr pro detekci skokové hrany,  $x_{max} = 1.00$

Odvozením optimálního filtru pro detekci skokové hrany v 1D však Canny nekončí. Postup zobecňuje pro 2D obrazy a získaný gradientní obraz podrobuje prahování s hysterezí a dvěma prahy  $t_{high}$  a  $t_{low}$ . Pokud je bod indikován jako hranový s pravděpodobností převyšující  $t_{high}$ , je tento bod automaticky označen za hranový. Pokud má bod váhu vyšší než  $t_{low}$ , avšak nižší než

$t_{high}$ , je považován za hranu pouze v případě, že sousedí s jiným bodem hrany. V případě, že detektor indikuje pravděpodobnost nižší než  $t_{low}$ , nejedná se o bod hrany.

Canny dále rozebírá vliv škálování na detekci hran a poukazuje na důležitost použití různě „širokých“ (rozuměj prostorově rozměrných) filtrů. Filtry s úzkou odezvou obecně lépe lokalizují hranu, avšak zanáší falešné hrany na místa zatížená šumem. Prostorově více rozložené filtry lépe odstraňují šum, avšak za cenu horší lokalizace. Tyto dva požadavky jdou proti sobě a je nutné nalézt vhodný kompromis. Canny navrhuje heuristický postup, při kterém je postupně aplikována sada filtrů se zvětšující se šířkou a klesajícím prahem. Nejprve jsou tak detekovány a přesně lokalizovány výrazné hrany a postupně jsou přidávány méně výrazné a zašuměné hrany, avšak již s menší přesností.

Cannyho detektor je velmi populární. Vzhledem k výpočetním nárokům srovnatelným s dříve uvedenými metodami a lepšími vlastnostmi (mezi něž mimo jiné patří i jednoznačná odezva) je v současnosti preferovanou metodou pro detekci hran s přesností na celé pixely. Je implementován i ve známé knihovně OpenCV [6]. Zde je však oproti původnímu článku použita lehce modifikovaná verze. Obraz je nejprve vyhlazen Gaussovým okénkem a následně jsou Sobelovými operátory detekovány potenciální hrany. Z matematického hlediska optimální filtry odvozené Cannyem zde tudíž nejsou použity. Následuje non-maximum suppression algoritmus pro zachování lokálních maxim a potlačení ostatních neextremálních bodů (čímž se zajistí jednoznačná odezva na hranu). Na závěr je provedeno opět Cannyem navržené prahování s hysterezí.

### 3 Detektory se subpixelovou přesností

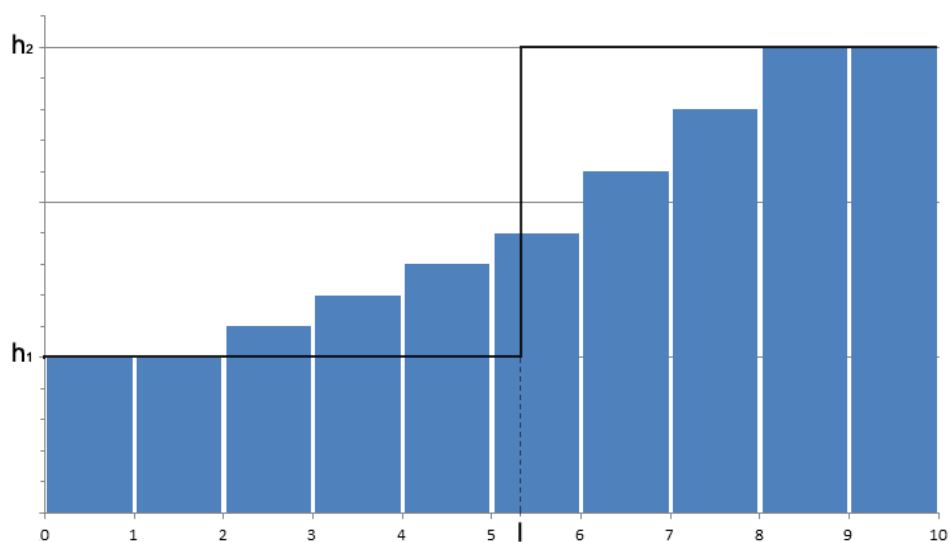
#### 3.1 Momentové metody

Momentové metody využívají k určení vlastností hrany momenty okolí bodu. Jejich základem je analogie obrazové funkce s deskou materiálu s proměnlivou hustotou. Hustota materiálu v daném bodě je dána hodnotou obrazové funkce. Nejprve jsou numericky vypočteny hodnoty zvolených momentů v okolí bodu hrany. Hrana je modelována nějakou vhodnou funkcí s parametry. Následně se hledají takové parametry této funkce, aby se jejich analyticky vypočtené momenty shodovaly s numerickými výsledky z prvního kroku. Z několika jasových či prostorových momentů jsou tak odvozeny přímé vztahy pro výpočet parametrů hrany – vzdálenost hrany od středu pixelu, úhel natočení hrany, jasové úrovně na obou stranách hrany a případně další parametry (závisí na typu modelu). Momentové metody pracují na principu integrálních operátorů a tudíž se výborně vyrovnávají se šumem. Většina pramenů uvádí jako první práci v této oblasti článek autorů Tabatabaie a Mitchella [10] z roku 1984, nicméně již o 3 roky dříve publikovali prakticky použitelný algoritmus používající momenty Machuca a Gilbert [9].

##### 3.1.1 Použití gray-level momentů pro lokalizaci se subpixelovou přesností

###### 3.1.1.1 1-dimenzionální analýza

Princip metody vysvětlíme nejprve na 1D případě. Metoda předpokládá, že reálná hrana je nespojité rozhraní mezi dvěma oblastmi - skoková změna intenzity z hodnoty  $h_1$  na hodnotu  $h_2$  v místě  $l$ , která se v obraze projeví jako neklesající (nebo nerostoucí) sekvence obrazové funkce:



Obr. 10 Reálná skoková hrana zobrazená po průchodu optickým systémem jako sekvence pixelů s různými úrovněmi šedi

Tabatabaie a Mitchell [10] používají pro vyjádření parametrů hrany  $h_1$ ,  $h_2$  a  $l$  první až třetí gray-level moment, přičemž  $i$ -tý moment funkce  $f(x)$  na intervalu  $\langle 0, n \rangle$  je definován takto:

$$m_i = \int_0^n f^i(x) dx, \quad i = 0, 1, 2, \dots \quad (9)$$

i-tý moment diskrétní obrazové funkce lze pak vyjádřit takto:

$$\overline{m_i} = \frac{1}{n} \sum_{j=1}^n x_j^i, \quad i = 0, 1, 2, \dots \quad (10)$$

kde  $n$  je počet pixelů zkoumaného okolí. Zavedeme-li následující substituci:

$$\begin{aligned} p_1 &= \frac{l}{n} \\ p_2 &= \frac{n-l}{l} = 1 - p_1 \\ h &= h_1 \\ h_2 &= h_1 + k \end{aligned} \quad (11)$$

lze pak z (9) analyticky odvodit vztahy pro momenty  $m_1 - m_3$ :

$$\begin{aligned} m_1 &= (1 - p_2)h + p_2(h + k) = p_1 h_1 + p_2 h_2 \\ m_2 &= (1 - p_2)h^2 + p_2(h + k)^2 = p_1 h_1^2 + p_2 h_2^2 \\ m_3 &= (1 - p_2)h^3 + p_2(h + k)^3 = p_1 h_1^3 + p_2 h_2^3 \end{aligned} \quad (12)$$

Dosazením vyčíslených hodnot  $\overline{m_1} - \overline{m_3}$  z (10) tak získáváme soustavu 3 rovnic o třech neznámých  $p_2, h_1, k$ , pro jejíž řešení lze odvodit následující vztahy:

$$\begin{aligned} p_2 &= \frac{1}{2} \left( 1 - s \sqrt{\frac{1}{4 + s^2}} \right) \\ h_1 &= \overline{m_1} - \beta \sqrt{\frac{p_2}{p_1}} \\ k &= 2\beta \sqrt{\frac{p_2}{p_1}} \end{aligned} \quad (13)$$

kde:

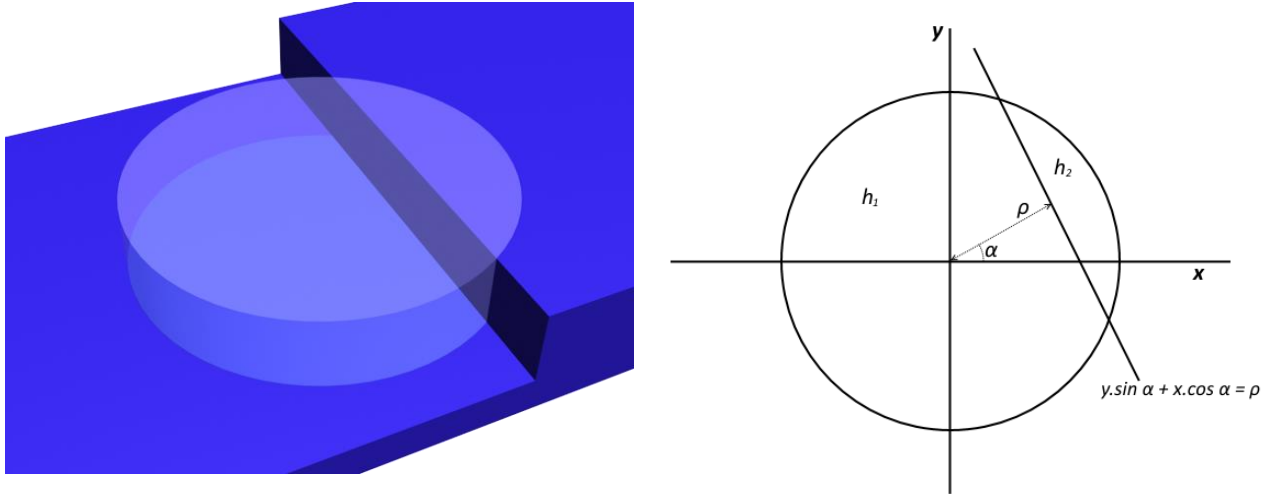
$$s = \frac{\overline{m_3} + 2\overline{m_1}^3 - 3\overline{m_1}\overline{m_2}}{\beta^3} \quad (14)$$

$$\beta^2 = \overline{m_2} - \overline{m_1}^2$$

Zpětným dosazením do substituce (11) pak získáme přesné umístění hrany  $l$  a jasové úrovně na obou stranách hrany  $h_1, h_2$ .

### 3.1.1.2 2-dimenzionální analýza

Ve 2D případě se uvažuje ideální skokové rozhraní o dvou intenzitách  $h_1$  a  $h_2$ , hranice mezi oblastmi je přímka.



Obr. 11 Vizualizace skokové hrany a její matematický model na kruhovém okolí  $\Omega$ .  
Dělicí přímka s rovnicí  $y.\sin \alpha + x.\cos \alpha = \rho$

Analýza obrazové funkce  $I(x,y)$  se provádí na kruhovém okolí zkoumaného bodu  $\Omega$ , přičemž se uvažují plošné integrály reprezentující gray-level momenty:

$$m_i = \frac{1}{\pi} \iint_{\Omega} I^k(x,y) dx dy = h_1^k p_1 + h_2^k p_2 \quad (15)$$

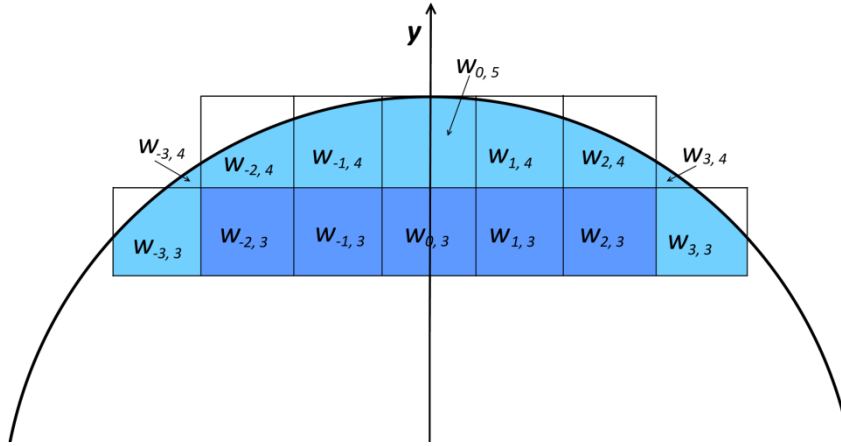
kde podobně jako v (11) je  $p_1$  podíl plochy zastoupený body s intenzitou  $h_1$  a  $p_2$  podíl plochy zastoupený body s intenzitou  $h_2$ .

V případě diskretizované obrazové funkce lze vztah přepsat:

$$\overline{m}_i = \sum_{\Omega} w_{xy} I^i(x,y) \quad (16)$$

Kde  $w_{xy}$  jsou váhy sloužící pro srovnání měřítek mezi analyticky počítaným integrálem a integrálem diskretizované obrazové funkce počítaném přes pixely. Váhy dále zohledňují započítávanou plochu okrajových pixelů (viz Obr. 12).

Pro stanovení okolí lze váhy  $w_{xy}$  předpočítat pro celý obraz jednou a následně numericky vyhodnocovat diskretizované momenty  $\overline{m}_i$ . Jakmile jsou k dispozici hodnoty momentů  $\overline{m}_1 - \overline{m}_3$ , lze nahlédnout, že (15) odpovídá (10) a podobně jako v 1D případě použít vztahy (13) pro výpočet neznámých parametrů  $h_1, h_2, p_1$  a  $p_2$ .



Obr. 12 Váhy pixelů obrazové funkce pro výpočet momentů

Z poměrného zastoupení jasů  $p_1$  a  $p_2$  úrovní  $h_1$  a  $h_2$  lze ze vzorců pro výpočet plochy kruhové úseče sestavit vztah mezi vzdáleností  $\rho$  hraniční přímky od středu zkoumané oblasti a plochy, kterou zaujímá. Získanou transcendentální rovnici, jejíž odvození je uvedeno v [10] je nutné řešit numericky. Výsledkem je vzdálenost  $\rho$  středu oblasti od hledané hrany.

Pro výpočet úhlu  $\alpha$ , jenž určuje směrnici normály hrany, jsou nutné další doplňující informace, samotné statistické momenty k tomu nestačí. Vhodné jsou 1. prostorové momenty, jak je navrženo v [9].  $i$ -tý prostorový moment v 1D je definován vztahem:

$$M_i = \int_I x^i f(x) dx, \quad i = 0, 1, 2, \dots \quad (17)$$

ve 2D pak:

$$M_{ij} = \iint_{\Omega} x^i y^j f(x, y) dx dy, \quad i, j = 0, 1, 2, \dots \quad (18)$$

Chápeme-li funkci  $f(x, y)$  jako hustotu, udává bod

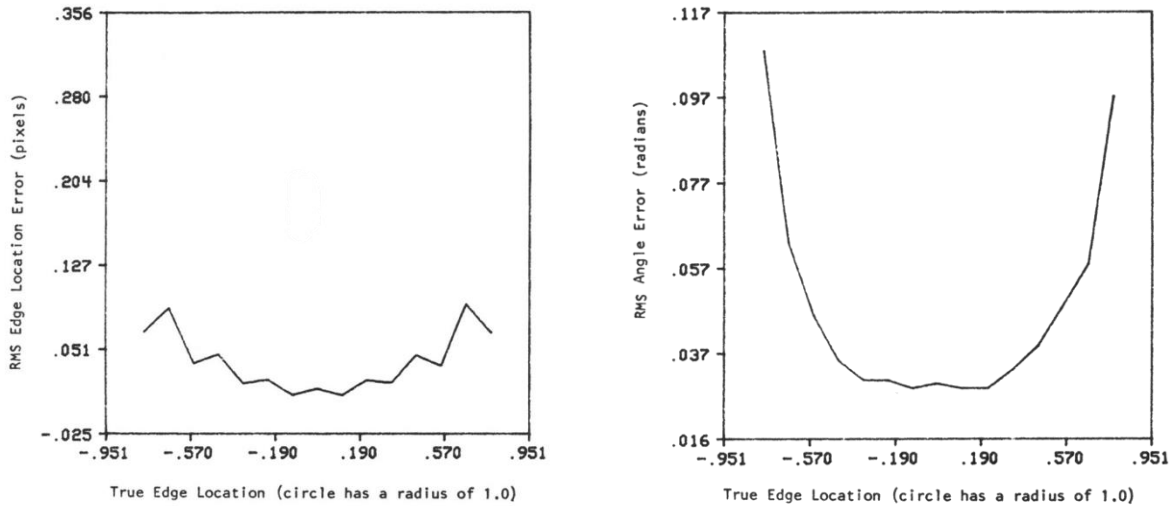
$$C = \left[ \frac{M_{10}}{M_{00}}; \frac{M_{01}}{M_{00}} \right] \quad (19)$$

souřadnice těžiště oblasti  $\Omega$  (centroid). V případě, že uvažujeme ideální model hrany podle Obr. 11, je poloha centroidu vůči středu kruhové oblasti  $\Omega$  vektorem normály hrany (plyne z osově souměrnosti vzhledem k ose procházející středem oblasti  $\Omega$  a kolmé k hraně). Nahradíme-li ideální modelové momenty momenty vypočtenými z diskretizované obrazové funkce (podle (20)), získáme odhad polohy centroidu a následně směrnici normály hrany (tj. hledaný úhel  $\alpha$ ).

$$\overline{M}_{ij} = \sum_{\Omega} w_{xy} x^i y^j I(x, y) \quad (20)$$

### 3.1.1.3 Vliv šumu

V [10] a [11] je detailně odvozen a rozebrán vliv aditivního bílého šumu na přesnost lokalizace hrany a to jak pro 1D, tak 2D případ. Ukazuje se, že s rostoucím podílem šumu v obraze se detekovaná poloha hrany posouvá blíže ke středu zkoumané oblasti  $\Omega$ . Toto zjištění je pochopitelné – samotný šum má nulovou střední hodnotu a tudíž centroid ve středu zkoumané oblasti. Centroid původně vychýlený užitečným signálem je aditivním šumem posouván blíže středu a to tím více, čím víc je šum zastoupen. Metoda jako taková je díky použitým integrálním operátorům působícím jako nízkofrekvenční propust vůči šumu značně robustní.



Obr. 13 Vliv aditivního bílého šumu na chybu určení parametrů hrany  $\rho$  a  $\alpha$ . SNR (signal-to-noise ratio) pro zobrazený případ je 20 dB.

## 3.1.2 Použití prostorových momentů pro lokalizaci se subpixelovou přesností

### 3.1.2.1 1-dimenzionální analýza

Lokalizace hran za pomoci prostorových momentů funguje podobně jako u statistických („gray-level“) momentů – z vhodné sady prostorových momentů se vyjádří explicitní vztahy pro výpočet parametrů hrany. Následně se hodnoty potřebných momentů vypočtou numericky z obrazových dat a ty se použijí pro vyčíslení parametrů hrany. Detailní odvození pro 1D i 2D případy je uvedeno v [12].

Definice prostorového momentu v 1D již byla v textu uvedena, nicméně tematicky spadá do této kapitoly, proto ji zopakujeme:

$$M_i = \int_I x^i f(x) dx, \quad i = 0, 1, 2, \dots \quad (21)$$

Pro model skokové hrany (viz Obr. 10), kde  $h_1 = h$ ,  $h_2 = h+k$  lze pro první tři momenty analyticky odvodit:



$$M_0 = 2h + k(1 - l)$$

$$M_1 = \frac{1}{2}k(1 - l^2) \quad (22)$$

$$M_2 = \frac{2}{3}h + \frac{1}{3}k(1 - l^3)$$

Řešením soustavy pak získáme explicitní vztahy pro parametry hrany  $h$ ,  $k$ ,  $l$ :

$$\begin{aligned} l &= \frac{3M_2 - M_0}{2M_1} \\ k &= \frac{2M_1}{1 - l^2} \\ h &= \frac{1}{2}[M_0 - k(1 - l)] \end{aligned} \quad (23)$$

### 3.1.2.2 2-dimenzionální analýza

Uvažujme ideální model skokové hrany ve 2D podle Obr. 11. Pro výpočty založené na prostorových 2D momentech se ukazuje vhodné transformovat problém do základní polohy a otočit zkoumané kruhové okénko o úhel  $-\alpha$  tak, aby byla hrana rovnoběžná s osou  $y$ . Není však nutné transformovat obrazová data, stačí provést rotaci momentů.

Nejprve definujme prostorový 2D moment:

$$M_{ij} = \iint_{\Omega} x^i y^j f(x, y) dx dy, \quad i = 0, 1, 2, \dots \quad (24)$$

Rotací zkoumaného okolí o libovolný úhel  $\varphi$  se transformují prostorové momenty podle následujících vztahů:

$$M'_{pq} = \sum_{r=0}^p \sum_{s=0}^q \binom{p}{r} \binom{q}{s} (-1)^{q-s} (\cos \varphi)^{p-r+s} (\sin \varphi)^{q+r-s} M_{p+q-r-s, r+s} \quad (25)$$

Pro momenty do řádu 2 tak lze rozepsat:

$$M'_{00} = M_{00} \quad (26)$$

$$M'_{10} = \cos \varphi M_{10} + \sin \varphi M_{01} \quad (27)$$

$$M'_{01} = -\sin \varphi M_{10} + \cos \varphi M_{01} \quad (28)$$

$$M'_{11} = \sin \varphi \cos \varphi (M_{02} - M_{20}) + M_{11}(\cos^2 \varphi - \sin^2 \varphi) \quad (29)$$

$$M'_{20} = \cos^2 \varphi M_{20} + 2\cos \varphi \sin \varphi M_{11} + \sin^2 \varphi M_{02} \quad (30)$$

$$M'_{02} = \sin^2 \varphi M_{20} - 2\cos \varphi \sin \varphi M_{11} + \cos^2 \varphi M_{02} \quad (31)$$

Dále požadujeme otočení momentů o takový úhel  $\varphi$ , aby byla zkoumaná oblast osově souměrná podle osy  $x$  a tudíž hledaná hrana rovnoběžná s osou  $y$ . To nastává v případě, že je  $M'_{01} = 0$ . Zkombinováním tohoto požadavku s (28) dostáváme vztah

$$M'_{01} = M_{01}\cos \varphi - M_{10}\sin \varphi = 0 \quad (32)$$

Jeho řešením je výraz

$$\varphi = \arctan \frac{M_{01}}{M_{10}}, \quad (33)$$

který je současně hledaným úhlem natočení hrany  $\alpha$  ( $\alpha = -\varphi$ ).

Je-li po provedené rotaci hrana rovnoběžná s osou  $y$ , pracujeme již se zjednodušeným modelem hrany, pro jehož momenty je možné z (26), (27), (30) a (24) analyticky odvodit:

$$\begin{aligned} M'_{00} &= h\pi + \frac{k}{2}\pi - k\sin^{-1}l - kl\sqrt{1-l^2} \\ M'_{10} &= \frac{2}{3}k\sqrt{(1-l^2)^3} \\ M'_{20} &= \frac{h}{4}\pi + \frac{k}{8}\pi + \frac{k}{2}l\sqrt{(1-l^2)^3} - \frac{k}{4}l\sqrt{1-l^2} - \frac{k}{4}\sin^{-1}l \end{aligned} \quad (34)$$

Řešením této soustavy (34) pak získáváme vztahy pro parametry hrany:

$$\begin{aligned} l &= \frac{4M'_{20} - M'_{00}}{3M'_{10}} \\ k &= \frac{3M'_{10}}{2\sqrt{(1-l^2)^3}} \\ h &= \frac{1}{2\pi} \left[ 2M'_{00} - k \left( \pi - 2\sin^{-1}l - 2l\sqrt{1-l^2} \right) \right] \end{aligned} \quad (35)$$

Na závěr zbývá z obrazových dat numericky dopočítat odhady původních (nerotovaných) momentů, z nichž po dosazení do (26)-(31) získáme momenty rotované a z nich pak podle (35) již parametry hledané hrany. Výpočet diskretizovaných momentů se provádí numerickou integrací přes zkoumané kruhové okolí za předpokladu, že hodnota obrazové funkce je v rámci jednoho pixelu konstantní. Na okrajích kruhového okolí se uvažuje pouze část plochy pixelu obsažená ve zkoumané oblasti (podobně jako u metody gray-level momentů, Obr. 12). Praktická realizace je taková, že pro jednotlivé momenty se předpočítají masky, jejichž korelací (či konvolucí) s obrazem se získají požadované momenty.

### 3.1.3 Zernikovy momenty

#### 3.1.3.1 Definice a vlastnosti Zernikových momentů

Zernikovy momenty byly poprvé použity pro subpixelovou lokalizaci hran v [13] a od té doby se objevily v mnoha dalších publikacích. Princip lokalizace je podobný jako u předchozích dvou

metod. Na rozdíl od prostorových momentů, které jsou projekcí obrazové funkce na polynomy  $x^p y^q$ , jež tvoří ortogonální bázi, jsou Zernikovy momenty projekcí obrazové funkce na tzv. Zernikovy polynomy, které ortogonální jsou (na jednotkovém kruhu). To přináší několik výhod, nejdůležitější z nich je redukce informační redundance, takže k určení parametrů hrany je potřeba menší počet ortogonálních momentů než v případě momentů prostorových. Lze ukázat, že existuje nekonečný počet ortogonálních polynomiálních bází ve 2 proměnných (a tudíž i nekonečný počet typů „momentů“). Zernikovy polynomy byly vybrány jako vhodná báze z důvodů invariance některých svých vlastností (zdůvodnitelné z teorie grup) a jednoduchých transformačních vztahů při rotaci zdrojových dat. Zernikův moment řádu  $p$  a s opakováním  $q$  je definován následujícím vztahem:

$$Z_{pq} = \frac{n+1}{\pi} \iint_{\Omega} f(x,y) V_{pq}^*(\rho, \theta) dx dy, \quad p, q = 0, 1, 2, \dots \quad (36)$$

kde  $\Omega$  je oblast jednotkového kruhu daná podmínkou  $x^2 + y^2 \leq 1$ . Zlomek před integrálem je normalizační faktor a lze jej pro účely lokalizace vynechat. V diskrétní podobě lze Zernikovy momenty vyjádřit následovně:

$$Z_{pq} = \sum_x \sum_y w_{xy} f(x,y) V_{pq}^*(\rho, \theta), \quad \begin{matrix} p, q = 0, 1, 2, \dots \\ x^2 + y^2 \leq 1 \end{matrix} \quad (37)$$

Komplexní Zernikovy polynomy  $V_{pq}^*(\rho, \theta)$  jsou komplexně sdružené polynomy k  $V_{pq}(\rho, \theta)$ , jež lze zapsat s pomocí radiálního polynomu  $R_{pq}(\rho)$  v polárních souřadnicích následovně:

$$V_{pq}(\rho, \theta) = R_{pq}(\rho) e^{-jq\theta} \quad (38)$$

$$R_{pq}(\rho) = \sum_{k=0}^{\frac{p-|q|}{2}} (-1)^k \frac{(p-k)!}{k! \left(\frac{p+|q|}{2} - k\right)! \left(\frac{p-|q|}{2} - k\right)!} \rho^{p-2k} \quad (39)$$

kde  $0 \leq q \leq p$  a  $(p - |q|)$  je sudé

Jak již bylo řečeno, Zernikovy polynomy jsou na jednotkovém kruhu ortogonální, tj. platí pro ně:

$$\iint_{\Omega} V_{nm}^*(\rho, \theta) V_{pq}(\rho, \theta) dx dy = \begin{cases} \pi/(n+1) & \text{pro } n = p \text{ a } m = q \\ 0 & \text{jinak} \end{cases} \quad (40)$$

Dále pro funkci  $f(x,y)$  otočenou kolem středu zkoumaného jednotkového kruhu o úhel  $\varphi$  platí jednoduchý vztah:

$$Z'_{pq} = Z_{pq} e^{-jq\varphi} \quad (41)$$

odtud je zřejmé, že komplexní Zernikovy momenty pro otočený obraz mají konstantní velikost, mění pouze svoji fázi. Tato vlastnost je důležitá pro rotačně invariantní rozpoznávání vzorů.

### 3.1.3.2 Použití Zernikových momentů pro lokalizaci se subpixelovou přesností

Uvažujme opět jednoduchý skokový model hrany ve 2D podle Obr. 11. Podobně jako v případě prostorových momentů jej otočme o úhel  $\varphi = -\alpha$  do základní polohy, kdy je hledaná hrana rovnoběžná s osou  $y$ . Pro odvození parametrů hrany lze použít Zernikovy momenty  $Z_{00}$ ,  $Z_{11}$  a  $Z_{20}$  s následujícími Zernikovými polynomy (2 z nich mají reálné koeficienty, jeden je komplexní):

$$\begin{aligned} V_{00}^* &= 1 \\ V_{11}^* &= x - jy \\ V_{20}^* &= 2x^2 + 2y^2 - 1 \end{aligned} \quad (42)$$

Pro momenty obrazové funkce otočené do základní polohy dosazením do (41) platí:

$$\begin{aligned} Z'_{00} &= Z_{00} \\ Z'_{11} &= Z_{11}e^{j\varphi} \\ Z'_{20} &= Z_{20} \end{aligned} \quad (43)$$

Uvažujme model hrany v základní poloze s obrazovou funkcí  $f'(x,y)$ . Dosazením  $V_{11}^*$  z (42) do (36) (při vynechání normalizačního členu) dostáváme:

$$Z'_{11} = \iint_{\Omega} x f'(x,y) dx dy - j \iint_{\Omega} y f'(x,y) dx dy \quad (44)$$

Model hrany v základní poloze je souměrný podle osy  $x$ , tudíž imaginární složka reprezentovaná druhým integrálem je nulová. Rotovaný moment  $Z'_{11}$  je tedy čistě reálný. Z (41) a (44) pro imaginární složku  $Z'_{11}$  platí:

$$\text{Im}[Z'_{11}] = 0 = \sin \varphi \text{Re}[Z_{11}] - \cos \varphi \text{Im}[Z_{11}] \quad (45)$$

Řešením této rovnice (45) získáme orientaci hrany  $\varphi$ :

$$\varphi = \arctan\left(\frac{\text{Im}[Z_{11}]}{\text{Re}[Z_{11}]}\right) \quad (46)$$

Dále z analyticky vyjádřených integrálů (36) momentů  $Z_{00}$ ,  $Z_{11}$  a  $Z_{20}$  pro model hrany v základní poloze získáváme:

$$\begin{aligned} Z'_{00} &= h\pi + \frac{k\pi}{2} - k \arcsin l - kl\sqrt{1-l^2} \\ Z'_{11} &= \frac{2k(1-l^2)^{3/2}}{3} \\ Z'_{20} &= \frac{2kl(1-l^2)^{3/2}}{3} \end{aligned} \quad (47)$$

Řešením této soustavy (47) a nahrazením otočených Zernikových momentů původními z (43) získáme zbývající parametry hrany:

$$\begin{aligned}
l &= \frac{Z_{20}}{Z_{11}e^{j\varphi}} \\
k &= \frac{3Z_{11}e^{j\varphi}}{2(1-l^2)^{3/2}} \\
h &= \frac{Z_{00} - \frac{k\pi}{2} + k \arcsin l + kl\sqrt{1-l^2}}{\pi}
\end{aligned} \tag{48}$$

K výpočtu čtyř parametrů ideální skokové hrany jsou třeba 3 Zernikovy momenty  $Z_{00}$ ,  $Z_{11}$  a  $Z_{20}$ . K odhadu těchto momentů se používají předpočítané masky, které lze získat integrací za předpokladu, že diskretizovaná obrazová funkce je v rámci jednoho pixelu konstantní. Volba velikosti masky závisí na velikosti okolí zahrnutého do odhadu momentů. V původní práci [13] je zvoleno okolí 5x5 pixelů, v jiných pracích lze nalézt okolí velikosti 5x5 – 9x9 pixelů. Dvě z korelačních masek obsahují čistě reálné koeficienty, jedna je komplexní. Nicméně výpočet momentu lze separovat na výpočet reálné a imaginární složky a pro vyhodnocení všech parametrů hrany jsou tedy zapotřebí 4 konvoluce s reálnými maskami. To dává oproti 6 konvolucím v případě prostorových momentů 33% úsporu výpočetního času. Navíc i následná analýza vlivu šumu na lokalizaci je díky jednoduchým vztahům mezi rotovanými a původními momenty značně jednodušší než v případě prostorových momentů.

### 3.2 Aproximační metody

Aproximační metody prokládají obrazovou funkci v okolí hrany křivkou (1D) nebo plochou (2D). U křivky či plochy je dána struktura (parametrický model) a hledají se optimální hodnoty parametrů tak, aby analyticky modelovaná hrana co nejlépe odpovídala skutečné hraně v obraze. Optimální hodnoty parametrů se hledají metodou nejmenších čtverců z přeurčené soustavy rovnic sestavené z okolních bodů zkoumaného pixelu. Díky tomu je metoda robustní vůči šumu, podobně jako momentové metody. Výpočetní náročnost oproti momentovým metodám je obecně vyšší, totéž ale platí o přesnosti.

#### 3.2.1 Aproximace hrany ERF funkcí

##### 3.2.1.1 Model ERF hrany v 1D a její lokalizace

Způsob lokalizace popsáný v této kapitole je state-of-the-art metodou subpixelové lokalizace hran co do přesnosti. V rychlosti ho předčí momentové či interpolační metody, ale pokud je prioritou přesnost lokalizace, je volbou číslo jedna. Hrana v obrazové funkci je modelována tzv. Gaussovou chybovou funkcí (ERF) a zohledňuje způsob, jakým obraz vzniká. Tj. způsob, jakým je skutečná skoková hrana zobrazena optickým systémem – konvolucí skokové funkce s Gaussovým okénkem. Je třeba zmínit, že uvedený model stoprocentně neodpovídá rozmazání způsobenému optikou a rovněž oblé hrany (jako jsou např. „okraje“ válců či koulí) jím nejsou přesně modelovány. Nicméně je matematicky zvládnutelný a fyzikálně akceptovatelný. Detailní analýza této problematiky včetně srovnání přesnosti a robustnosti s momentovými a interpolačními metodami je provedena v [16].

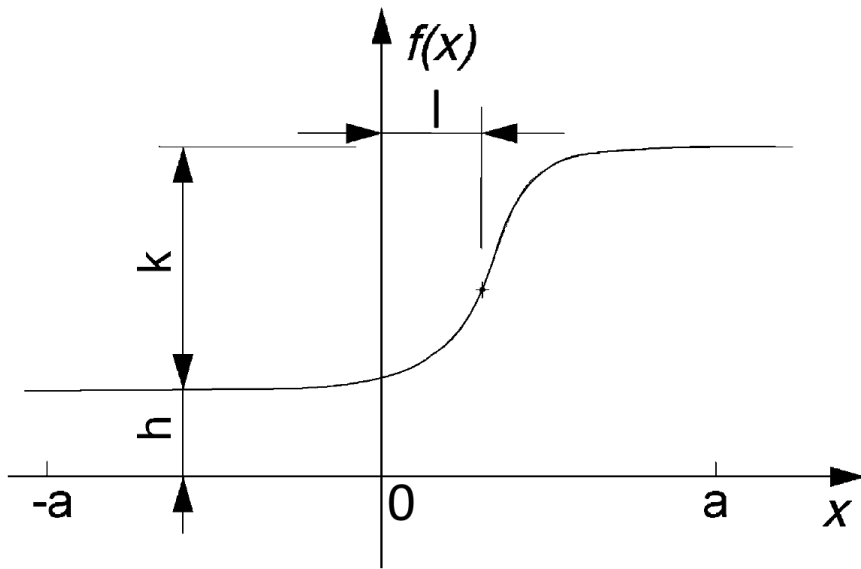
Definujeme-li  $ERF$  funkci vztahem:

$$ERF(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (49)$$

pak lze model rozostřené hrany v 1D případě z Obr. 14 vzniklý konvolucí skokové funkce a Gaussova okénka zapsat ve tvaru:

$$f(x) = \frac{k}{2} \left( ERF\left(\frac{x-l}{\sqrt{2}\sigma}\right) + 1 \right) + h \quad (50)$$

kde  $h$  značí intenzitu pozadí,  $k$  kontrast hrany,  $l$  pozici hrany a  $\sigma$  rozmazání hrany.



Obr. 14 Model rozostřené hrany s parametry  $h$ ,  $k$ ,  $l$  a  $\sigma$ .  $a$  je polovina velikosti okénka, na němž je prováděna identifikace.

Na senzoru vzniká obraz akumulací fotonů dopadajících na jednotlivé pixely senzoru. Celkový jas registrovaný pro daný pixel  $i$  v 1D je tedy dán integrací jasové funkce (50) v oblasti pixelu:

$$G(i) = \int_{i-0.5}^{i+0.5} f(x) dx \quad (51)$$

Máme-li vyjádřeny předpokládané hodnoty jasu jednotlivých pixelů podle modelu, lze je srovnat se skutečně nasnímanými hodnotami - stanovit chybovou funkci kvantifikující odchylku modelu od skutečných hodnot (nejmenší čtverce):

$$\Delta(l, h, k, \sigma) = \sum_{i=-a}^a [\tilde{G}(i) - G(i)]^2 \quad (52)$$

kde  $\tilde{G}(i)$  jsou nasnímané intenzity jednotlivých pixelů ve zkoumaném okénku a  $G(i)$  intenzity pixelů vypočtené na základě modelu (51).

Následná vícedimenzionální optimalizační úloha hledá hodnoty parametrů  $l, h, k, \sigma$  tak, aby se minimalizovala hodnota chybové funkce  $\Delta$ . Pro její numerické řešení lze použít řadu metod. Díky tomu, že je chybová funkce  $\Delta$  diferencovatelná, lze stanovit její gradient a pro řešení použít např. Newtonovu metodu.

Za zmínku stojí volba počátečních podmínek pro optimalizaci. Pro první bod hrany je možné použít odhad parametrů hrany s pomocí nějakého jednoduchého hranového detektoru pracujícího s pixelovou přesností (např. Sobelův operátor). Pro další body hrany je kromě uvedeného k dispozici ještě další možnost – jako počáteční hodnoty použít nalezené optimální parametry předchozího pixelu.

### 3.2.1.2 Model ERF hrany ve 2D a její lokalizace

Popsaný jednodimenzionální případ lze rozšířit do 2D. Autor tohoto článku se inspiroval prací [16], nicméně v ní uvedená 2D analýza podchycuje pouze hrany, jejichž projekce do obrazové roviny je vyjádřitelná explicitně ve tvaru  $y=f(x)$ . Tj. nelze podchytit např. hrany rovnoběžné s osou  $y$ . Navíc pro mapování vzdálenosti bodu od křivky ve 2D je použita vzdálenost bodu ke křivce ve směru osy  $y$ . Pro lineární hrany to nepředstavuje problém, změna metriky se projeví změnou parametru  $\sigma$ . Pro zakřivené hrany to však znamená, že rozostření (parametr  $\sigma$ ) se v rámci okénka mění, což neodpovídá představenému modelu. Zde provedená analýza je zobecněním uvedeného a tato omezení odstraňuje.

Uvažujme, že projekce hrany do obrazové roviny má tvar parametrické křivky  $P[\tilde{x}(t), \tilde{y}(t)]$ . Pro model jasové funkce ve 2D pak lze zapsat:

$$f(x, y) = \frac{k}{2} \left( \text{ERF} \left( \frac{\|[x, y], P\|}{\sqrt{2}\sigma} \right) + 1 \right) + h \quad (53)$$

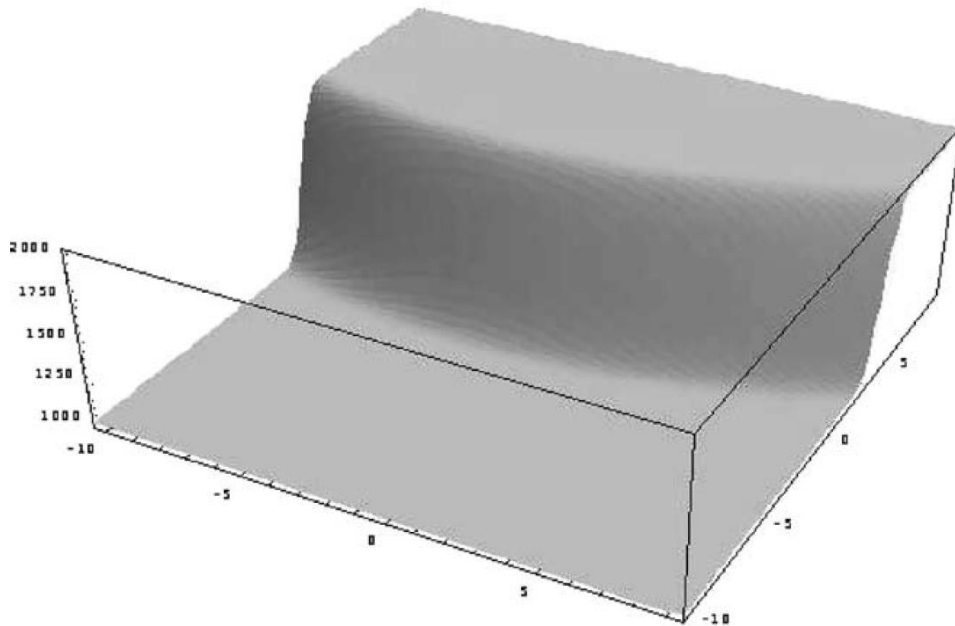
Kde  $\|[x, y], P\|$  je vzdálenost bodu  $[x, y]$  od hrany (tj. od kolmého průmětu bodu na tuto hranu). Např. pro přímkou  $P$  danou parametricky

$$P_l[At + B, Ct + D] \quad (54)$$

lze normu a následovně model hrany vyjádřit takto:

$$\|[x, y], P_l\| = \sqrt{\left[ x - B - A \frac{Ax + Cy - AB - CD}{A^2 + C^2} \right]^2 + \left[ y - D - C \frac{Ax + Cy - AB - CD}{A^2 + C^2} \right]^2} \quad (55)$$

$$f_l(x, y) = \frac{k}{2} \left( \text{ERF} \left( \frac{\sqrt{\left[ x - B - A \frac{Ax + Cy - AB - CD}{A^2 + C^2} \right]^2 + \left[ y - D - C \frac{Ax + Cy - AB - CD}{A^2 + C^2} \right]^2}}{\sqrt{2}\sigma} \right) + 1 \right) + h \quad (56)$$



Obr. 15 Ukázka 2D hrany modelované ERF funkcí

Lokalizace hrany se provádí na okolí zkoumaného bodu o velikosti  $(2a+1) \times (2b+1)$ . Rozměry okénka jsou lichá čísla ze zřejmého důvodu – aby byl zkoumaný pixel středem okénka. Pro jas registrovaný senzorem na pixelu  $[i, j]$  pak platí:

$$G(i, j) = \int_{i-0.5}^{i+0.5} \int_{j-0.5}^{j+0.5} f(x, y) dx dy \quad (57)$$

a chybová funkce (analogie výrazu (52) v 1D) určená k optimalizaci:

$$\Delta(A, B, C, D, h, k, \sigma) = \sum_{i=-a}^a \sum_{j=-b}^b [\tilde{G}(i, j) - G(i, j)]^2 \quad (58)$$

$\tilde{G}(i)$  jsou opět skutečné intenzity pixelů v obraze a  $G(i)$  intenzity pixelů vypočtené na základě modelu (57).

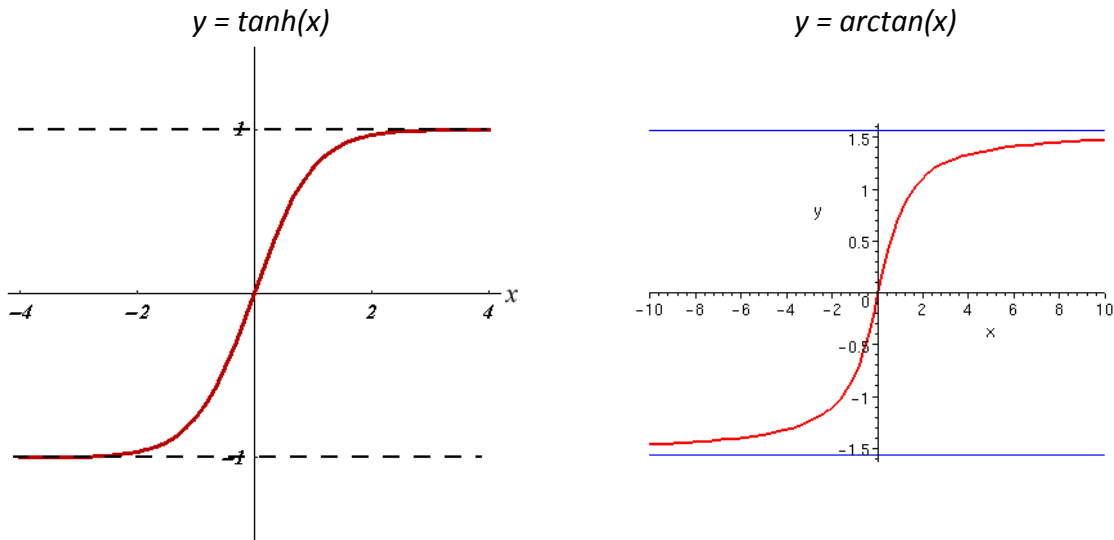
Na závěr se opět jako v 1D případě řeší vícedimenzionální optimalizační úloha s cílem nalézt takové hodnoty parametrů  $A, B, C, D$  (parametry průmětu lineární hrany (54) do obrazové roviny) a  $h, k, \sigma$  (parametry hrany známé z 1D případu), jež minimalizují hodnotu chybové funkce  $\Delta$  (58).

### 3.2.2 Aproximace hrany funkcí arcus tangens

Metoda uvedená v předchozí kapitole 3.2.1 modeluje hranu pomocí ERF funkce, tj. způsobem odpovídajícím fyzikálním principům formování obrazu. Nicméně vyjádření hrany je natolik složité, že významně prodlužuje dobu výpočtu. V některých případech je nutné získat výsledky



rychleji a na úkor drobných odchylek použít jednodušší aproximační funkci podobného tvaru - např.  $\tanh$  [19] nebo  $\arctan$  [18].



Obr. 16 Grafy funkcí hyperbolický tangens ( $\tanh$ ) a arcus tangens ( $\arctan$ )

Podle [18] je uvažován model 1D hrany ve tvaru

$$f(x) = a_1 \arctan(a_2 x + a_3) + a_4 \quad (59)$$

Oproti předchozí kapitole 3.2.1, kde je intenzita pixelu modelována integrálem spojitě obrazové funkce přes plochu pixelu, je zde pro odvození modelového jasu pixelu použito jednoduché navzorkování hodnoty spojitě obrazové funkce ve středu pixelu:

$$G(i) = f(i) = a_1 \arctan(a_2 i + a_3) + a_4 \quad (60)$$

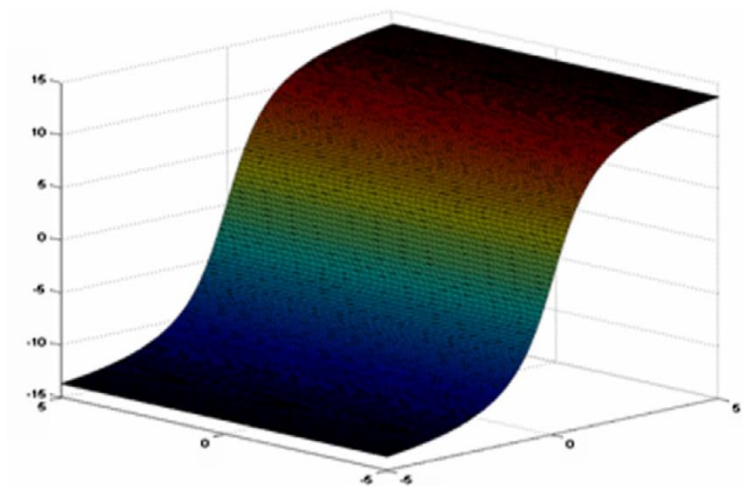
Chybová funkce již odpovídá standardní metodě nejmenších čtverců, pro zpracování dat na intervalu  $\langle -a, a \rangle$  platí:

$$\Delta(a_1, a_2, a_3, a_4) = \sum_{i=-a}^a [\tilde{G}(i) - G(i)]^2 \quad (61)$$

kde  $\tilde{G}(i)$  je skutečná intenzita  $i$ -tého pixelu v obraze. Vícedimenzionální minimalizací chybové funkce  $\Delta$  pro sadu parametrů  $a_1, a_2, a_3$  a  $a_4$  získáme parametry hrany (59). Pro polohu hrany  $l$  podle Obr. 14 platí:

$$l = -\frac{a_3}{a_2} \quad (62)$$

Rozšíření metody na 2D je provedeno opět zjednodušujícím způsobem, aby byla zachována rychlost. V [18] je předpokládána hrana s lineární projekcí do obrazové roviny („rovná hrana“) a s identickým náběhem v celém zkoumaném okénku. Předpokládá se tedy tvar hrany odpovídající ploše vzniklé pohybem křivky  $\arctan(x)$  podél vlastní binormály (tj. podél přímky kolmé na rovinu původní 1D křivky) – viz Obr. 17.



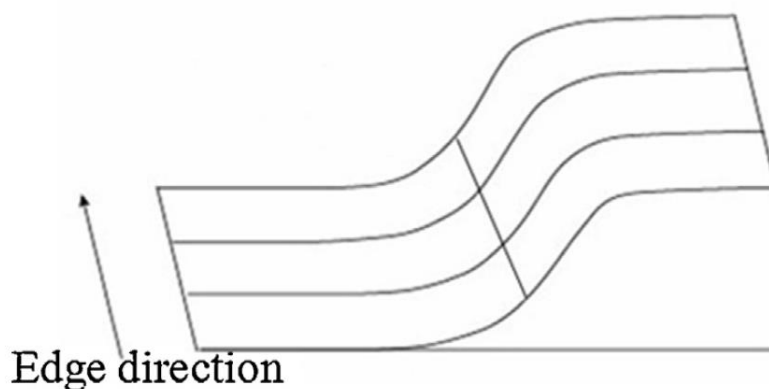
Obr. 17 Model hrany ve 2D

Nejprve se nalezne přibližný směr  $\theta_0$  normály hrany v obraze s pomocí Sobelových operátorů:

$$\begin{aligned}\vec{N}_0 &= [S_{x0}, S_{y0}] \\ \theta_0 &= \arctan \frac{S_{y0}}{S_{x0}}\end{aligned}\tag{63}$$

Tento odhad se následně zpřesňuje hledáním nejlepší shody obrazu s plochou obecného válce  $V(x,y)$  tvořeného kubickou křivkou (Obr. 18):

$$\begin{aligned}V(x, y) &= v_0 + v_1 X + v_2 X^2 + v_3 X^3 \\ X &= x \cos \theta + y \sin \theta\end{aligned}\tag{64}$$



Obr. 18 Obecná válcová plocha tvořená kubickou křivkou použitá pro určení směru hrany

Curve-fitting je prováděno opět metodou nejmenších čtverců a díky relativně přesnému počátečnímu odhadu vyžaduje většinou jen několik málo iterací.

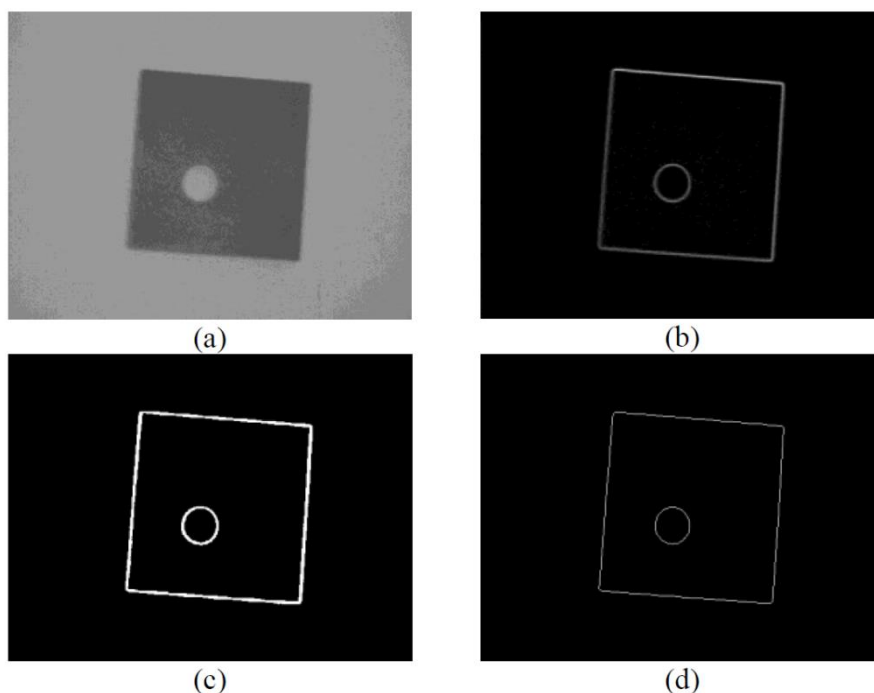
V posledním kroku algoritmu se ortogonálně promítnou všechny pixely zkoumaného okolí na normálu hrany (danou zpřesněným odhadem úhlu  $\theta$ ), čímž se 2D problém redukuje na dříve popsáný 1D problém. Na závěr lze nalezené parametry hrany  $\theta$  a  $l$  převést z polárních souřadnic do kartézských.

### 3.2.3 Aproximace gradientního obrazu Gaussovou funkcí

Další zajímavá aproximační metoda je uvedena v [20] a [21]. Práce se zaměřuje na lokalizaci výrazně rozostřených hran (motivací byla analýza snímků vzorků horkých kovů). Neprovádí se zde aproximace obrazové funkce, nýbrž jejího gradientu. Nejprve je obraz zpracován Sobelovými operátory  $S_x$  a  $S_y$  a určena velikost gradientu

$$\|\nabla I(i,j)\| = \sqrt{[S_x(I(i,j))]^2 + [S_y(I(i,j))]^2} \quad (65)$$

Následně je gradientní obraz prahován metodou ISODATA [23], což je iterativní metoda pro automatické určení vhodné hodnoty prahu. Pracuje tak, že je spočítána průměrná intenzita pixelů nad a pod počátečním odhadem prahu. Nově odhadnutá hodnota prahu je vyčíslena jako aritmetický průměr těchto dvou hodnot a celý proces se opakuje, dokud se hodnota prahu neustálí. Tímto procesem jsou nalezeni kandidáti hranových bodů. Následuje skeletizace, aby měly získané hrany tloušťku 1 pixelu. Proces předzpracování obrazu je ilustrován na Obr. 19.



Obr. 19 Postup předzpracování obrazu: a) původní obrázek, b) gradientní obraz získaný Sobelovými operátory, c) prahovaný gradientní obraz, d) prahovaný obraz po skeletizaci

Následně se provádí v okolí hranových bodů prokládání (curve fitting) původní gradientní funkce Gaussovou křivkou:

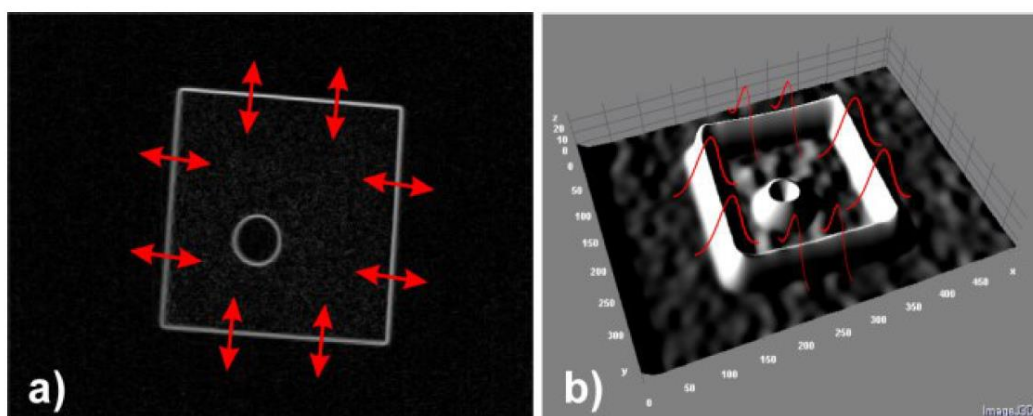
$$f(x) = Ae^{\frac{-(x-l)^2}{2\sigma^2}} \quad (66)$$

$A$  značí maximální hodnotu gradientu na hraně,  $l$  přesnou pozici hrany (se subpixelovou přesností) a  $\sigma$  míru rozostření.

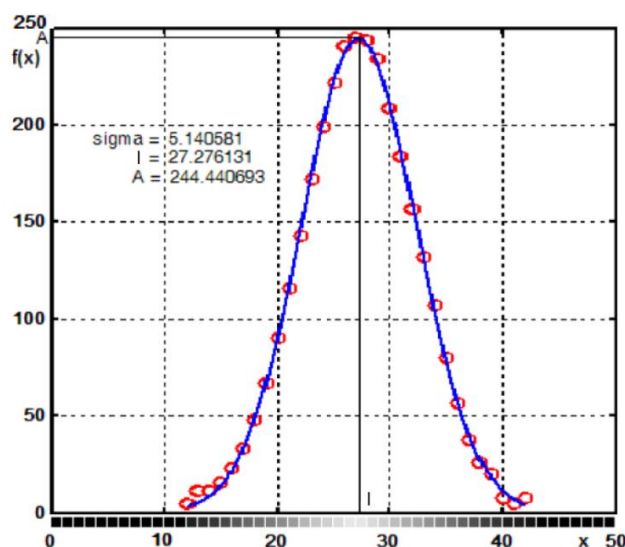
Uvažuje se prokládání pouze v 1D – ve směru normály hrany. Odhad směru normály hrany lze provést z poměru hodnot získaných Sobelovými operátory:

$$\varphi = \arctan \frac{S_y}{S_x} \quad (67)$$

Parametry Gaussovy funkce (66) se získají vícedimenzionální minimalizací chybové funkce (nejmenší čtverce) analogické s (52). V rozebíraném článku [20] je proces subpixelové lokalizace završen proložením kubické spline křivky celou hranou (myšleno projekcí hrany do obrazové roviny), čímž se dále vyhladí drobné odchylky jednotlivých bodů hrany.



Obr. 20 Princip prokládání gradientního obrazu Gaussovou funkcí. a) směr lineárního okolí uvažovaný pro rekonstrukci 1D gradientu naznačený v několika bodech, b) proložení Gaussovy funkce v těchto bodech



Obr. 21 Gaussova křivka proložená vzorky gradientního obrazu v okolí bodu hrany

Za největší nevýhodu uvedené metody lze považovat nutnost diferenciacie původního snímku. Tento proces je značně náchylný na šum a dokáže do výsledků zanést výrazné nepřesnosti. Nicméně v kombinaci s dolní propustí (konvoluce s Gaussovým okénkem) lze tento nedostatek potlačit. Vzhledem k typu zpracovávaných obrázků (obrázky s nejasnými hranami) a robustnosti metody vůči dodatečnému rozostření (dáno principem metody) však není tento fakt zásadní překážkou.

### 3.3 Interpolační metody

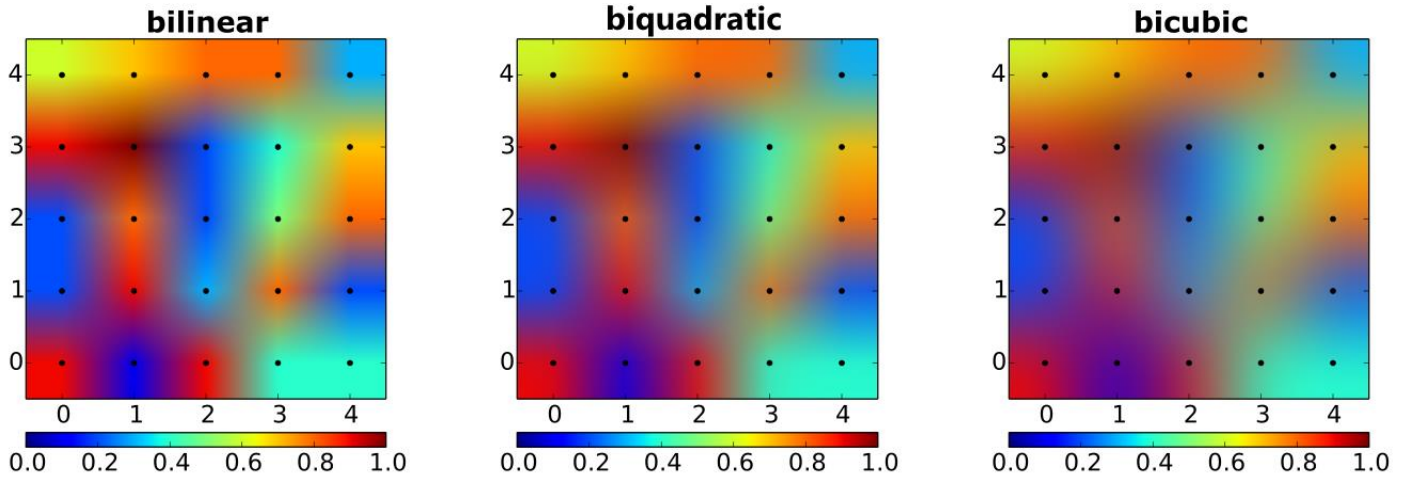
Interpolační metody provádí převzorkování původního obrazu v okolí předpokládané hrany ve vyšším rozlišení a následně hledají hranu v převzorkovaném výřezu. Používá se interpolace bilineární, bikvadratická či bikubická polynomem v původním či gradientním obraze [24], Lanczos algoritmus nebo různé varianty pokročilých nelineárních interpolací [25]-[30]. Přesnost metody i výpočetní nároky závisí na způsobu interpolace (za použití běžných interpolačních technik patří mezi rychlejší). V případě, že máme dodatečné informace o scéně, je vhodné je využít. Tím je možné částečně obejít velkou náchylnost těchto metod na šum.

#### 3.3.1 Bilineární, bikvadratická a bikubická interpolace

Metoda využívá proložení obrazové funkce polynomiální plochou tvaru:

$$f(x, y) = \sum_{i=0}^P \sum_{j=0}^P a_{ij} x^i y^j, \quad P = 1, 2, 3 \quad (68)$$

kde  $P$  je stupeň polynomu.  $P=1$  pro bilineární interpolaci,  $P=2$  pro kvadratickou interpolaci a  $P=3$  pro kubickou interpolaci. Koeficienty  $a_{ij}$  se určují ze soustavy algebraických rovnic sestavené pro zkoumané okénko velikosti  $2 \times 2$  pixelů u bilineární interpolace ( $3 \times 3$  u bikvadratické a  $4 \times 4$  u bikubické). Počet rovnic je identický s počtem hledaných parametrů. Soustava má tedy jediné řešení, při kterém hledaná plocha prochází přesně všemi navzorkovanými body obrazové funkce. Zkoumané okénko se posouvá po obraze (podobně jako u konvoluce) a nalezená aproximující plocha je použita k odhadu intenzity subpixelů ve středové oblasti okénka o velikosti  $1 \times 1$  pixel. Tato oblast je pro lichá  $P$  mezi středovými pixely  $\langle \frac{P-1}{2}, \frac{P-1}{2} + 1 \rangle \times \langle \frac{P-1}{2}, \frac{P-1}{2} + 1 \rangle$ , pro sudá  $P$  v okolí středového pixelu rozprostírající se do vzdálenosti  $\pm \frac{1}{2}$  pixelu na každou stranu, tj.  $\langle \frac{P}{2} - \frac{1}{2}, \frac{P}{2} + \frac{1}{2} \rangle \times \langle \frac{P}{2} - \frac{1}{2}, \frac{P}{2} + \frac{1}{2} \rangle$ .



Obr. 22 Ukázky bilineární, bikvadratické a bikubické interpolace.

Převzato z [https://en.wikipedia.org/wiki/Bicubic\\_interpolation](https://en.wikipedia.org/wiki/Bicubic_interpolation)

Bilineární interpolace zachovává ostřejší hrany, hůře však interpoluje plynulé přechody. Interpolace polynomy vyšších řádů vycházejí z většího okolí zkoumaného bodu a jsou tak robustnější vůči šumu a lépe vyhlazují pozvolné přechody. Nicméně u bikubické interpolace mohou vznikat v blízkosti ostrých hran aliasingové artefakty. Rovněž dochází k posunu hrany a změně rozměru oblastí, což není pro subpixelovou lokalizaci, která se používá právě z důvodů přesného určení pozice nejlepší řešení.

### 3.3.2 Kubická konvoluce

Kubická konvoluce je speciálním případem kubické aproximace, kdy se provádí aproximace obrazové funkce 1D kubickým polynomem postupně ve směru osy  $x$  a následně  $y$ . Uvažujme okénko velikosti  $4 \times 4$  pixely  $\langle x_{i-1}, x_{i+2} \rangle \times \langle y_{j-1}, y_{j+2} \rangle$ . Libovolný bod  $P[x_*, y_*]$  uvnitř středové oblasti  $\langle x_i, x_{i+1} \rangle \times \langle y_j, y_{j+1} \rangle$  o rozměru  $1 \times 1$  pixel lze pak interpolovat následujícím způsobem:

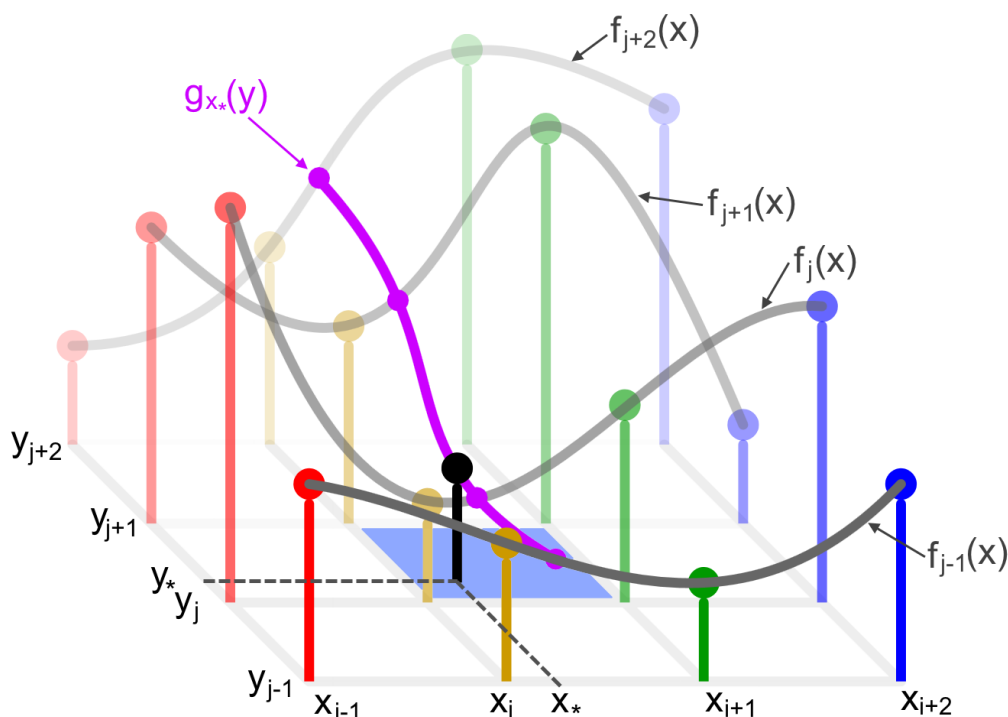
Pro každé  $k = j-1, j, j+1, j+2$  se čtyřmi body obrazové funkce  $x_{i-1,k}$ ,  $x_{i,k}$ ,  $x_{i+1,k}$  a  $x_{i+2,k}$  proloží kubický polynom tvaru

$$f_k(x) = a_{k,3}x^3 + a_{k,2}x^2 + a_{k,1}x + a_{k,0} \quad (69)$$

Jedná se o prokládání 1D obrazových dat ve směru osy  $x$ . Čtveřice funkcí  $f_k(x)$  se vyčíslí v bodě  $x_*$  a 4-mi nově získanými body se proloží kubický polynom, tentokrát ve směru osy  $y$

$$g_{x^*}(y) = b_{x^*,3}y^3 + b_{x^*,2}y^2 + b_{x^*,1}y + b_{x^*,0} \quad (70)$$

Následným vyčíslením polynomu  $g_{x^*}(y)$  v bodě  $y_*$  získáme interpolovanou hodnotu v požadovaném bodě  $P[x_*, y_*]$ .



Obr. 23 Princip kubické konvoluce

Na rozdíl od plné bikubické aproximace, kde se řeší soustava 16 rovnic o 16 neznámých, u kubické konvoluce se řeší výpočetně méně náročná sestava 5-ti soustav o 4 neznámých. S rostoucím počtem subpixelů (při jemnějším převzorkování) se však tato výhoda postupně stírá, neboť u bikubické interpolace stačí po nalezení koeficientů bikubického polynomu pouhé vyčíslování tohoto polynomu pro každé  $[x_*, y_*]$ , kdežto u kubické konvoluce je nutné pro každé jednotlivé  $y_*$  řešit dodatečnou soustavu.

### 3.3.3 ENO interpolace

Essentially Non-Oscillatory (ENO) interpolace byla poprvé zveřejněna v [25], aby se autoři vyhnuli numerickým oscilacím při simulacích proudění tekutin. Její aplikace pro účely lokalizace hran byla popsána např. v [26]. Metoda funguje podobně, jako výše popsaná kubická interpolace. I zde se nejprve určí aproximační 1D polynomy ve směru osy  $x$  a posléze v požadovaném místě 1D aproximace ve směru osy  $y$ . Rozdíl je v tom, že oblast interpolace není ve středu zpracovávaného okénka. Hlavní myšlenka metody je použít pro interpolaci oblast s co největší kontinuitou a s co nejmenším rozptylem. Tj. pokud možno zahrnout do aproximace pouze oblast z jedné strany hrany a úsek na druhé straně hrany z aproximace vyloučit. Toho lze docílit volbou takové sady bodů, pro které má nejvyšší derivace aproximačního polynomu nejnižší hodnotu. Toto kritérium je ekvivalentní požadavku na nejnižší hodnotu dělené difference (divided difference, [https://en.wikipedia.org/wiki/Divided\\_differences](https://en.wikipedia.org/wiki/Divided_differences)). Interpolace jednou sadou parametrů se provádí v okolí zkoumaného bodu, do vzdálenosti  $\pm \frac{1}{2}$  pixelu na každou stranu, tj. v oblasti  $\langle x_i - \frac{1}{2}, x_i + \frac{1}{2} \rangle \times \langle y_j - \frac{1}{2}, y_j + \frac{1}{2} \rangle$ . Pro tuto interpolovanou oblast se pak vyhledává sekvence sousedních bodů tak, aby jejich aproximací

vznikl polynom výše uvedených vlastností. Jedná se o nelineární metodu, neboť pixely určující aproximující polynomy jsou vybírány selektivně a pro každý interpolovaný bod může být použit jiný tvar okolí.

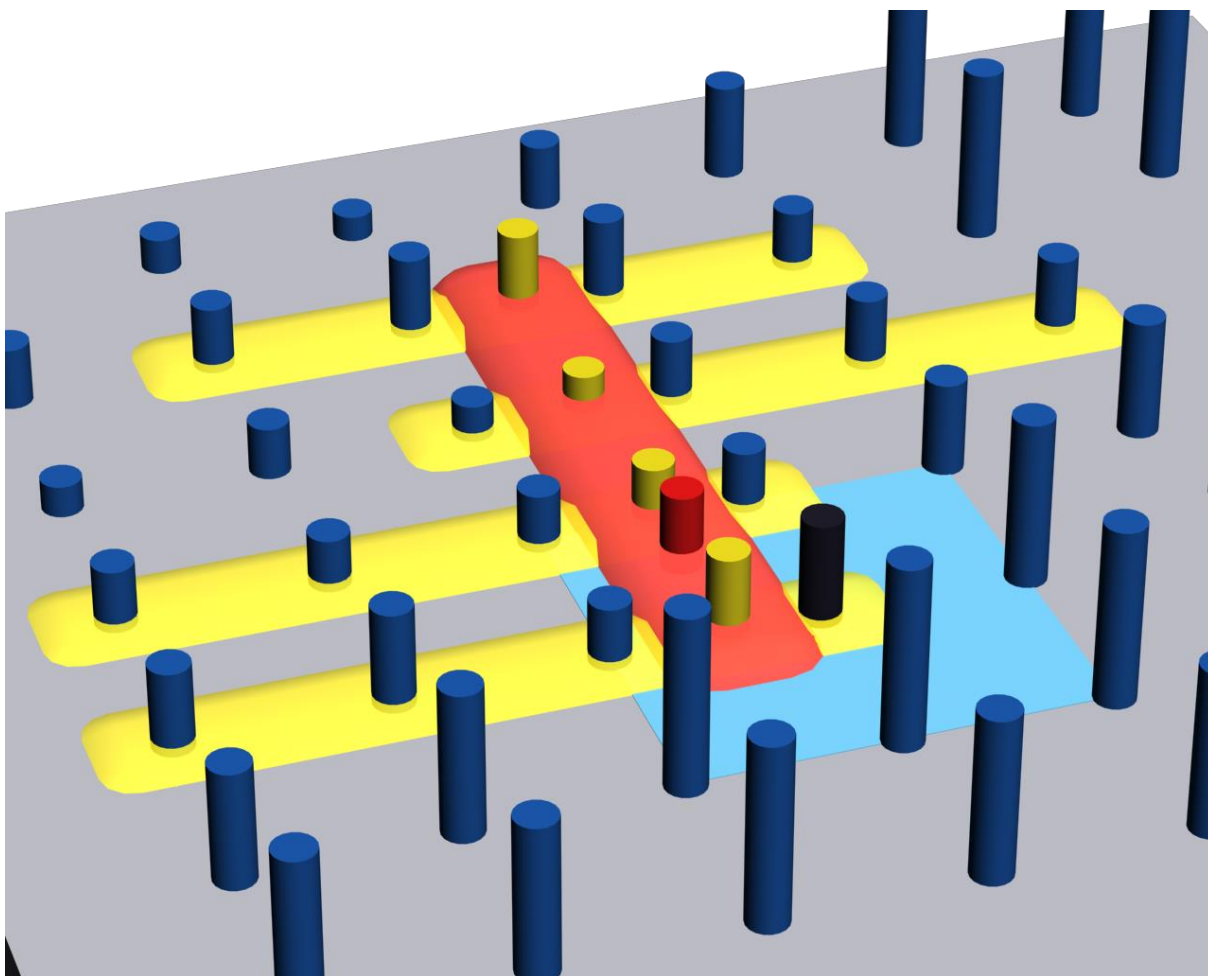
Definujme nejprve dělenou diferenci. Pro  $k+1$  datových vzorků  $[x_0, f(x_0)], \dots [x_k, f(x_k)]$  je dopředná dělená diference definována následujícím rekursivním vztahem:

$$\begin{aligned} f[x_i] &= f(x_i) \\ f[x_i, \dots, x_{i+j}] &= \frac{f[x_{i+1}, \dots, x_{i+j}] - f[x_i, \dots, x_{i+j-1}]}{x_{i+j} - x_i} \end{aligned} \quad (71)$$

Algoritmus ENO interpolace 4. řádu (kubickým polynomem) je následující:

1. Uvažujme interpolovaný bod  $P[x_*, y_*]$  nalézající se v okolí  $\langle x_i - \frac{1}{2}, x_i + \frac{1}{2} \rangle \times \langle y_j - \frac{1}{2}, y_j + \frac{1}{2} \rangle$  bodu  $[x_i, y_j]$ . Pro zpracování aktuálního řádku  $j$  položme  $k=j$ .
2. Pro  $k$ -tý řádek hledáme takové okolí o velikosti 4 pixelů, jež obsahuje  $x_i$  a má nejnižší dělenou diferenci. Proces je iterativní, hledané okolí se postupně zvětšuje. Nejprve do něj zařadíme bod  $x_i$  samotný. Následně vypočteme dělené difference zleva  $f[x_{i-1}, x_i]$  i zprava  $f[x_i, x_{i+1}]$ . Nově do okolí zařadíme toho ze sousedů, pro který má dělená diference nižší hodnotu. Proces opakujeme – pro nově získané okolí opět vypočteme dělené difference (o 1 vyššího řádu) zleva i zprava a přidáme sousední bod s nižší diferencí. Třetím opakováním tohoto postupu získáme 4-bodové okolí  $\{x_{k,i_{min}}, x_{k,i_{min}+1}, x_{k,i_{min}+2}, x_{k,i_{min}+3}\}$  aktuálního bodu ve směru osy  $x$ .
3. Získaným okolím  $\{x_{k,i_{min}}, x_{k,i_{min}+1}, x_{k,i_{min}+2}, x_{k,i_{min}+3}\}$  proložíme aproximační polynom 3. stupně stejně jako v (69) v případě kubické konvoluce. Vyčíslíme hodnotu v bodě  $f(x_*, y_k)$ .
4. Nyní hledáme 4-okolí  $\{y_{*,j_{min}}, y_{*,j_{min}+1}, y_{*,j_{min}+2}, y_{*,j_{min}+3}\}$  s nejmenší variancí ve směru osy  $y$  pro  $x = x_*$ . Postup je opět iterativní, stejný jako v případě hledání okolí ve směru osy  $x$ . Požadované hodnoty  $f(x_*, y_k)$  získáme dosazením do aproximačních polynomů  $k$ -tého řádku. Polynomy všech řádků  $y_k$  potřebných pro výpočet (rozsah indexu  $k$  závisí na výsledcích porovnání dělených diferencí a bodech přidávaných k aproximovanému okolí) získáme dle postupu v bodech 2 a 3 tohoto algoritmu.
5. Získaným okolím  $\{y_{*,j_{min}}, y_{*,j_{min}+1}, y_{*,j_{min}+2}, y_{*,j_{min}+3}\}$  proložíme aproximační polynom  $g_{x_*}(y)$  podle (70) a vyčíslíme jeho hodnotu v  $g_{x_*}(y_*)$ , čímž získáme požadovanou interpolovanou hodnotu v bodě  $P[x_*, y_*]$ .





Obr. 24 *Princip ENO interpolace. Hledaná interpolovaná hodnota v místě  $P[x_*, y_*]$  (červený bod) se nachází v okolí bodu rastru  $[x_i, y_j]$  (černý bod, světle modrá oblast). Řádky jsou interpolovány kubickými polynomy, jejichž koeficienty jsou určeny na základě hodnot uvnitř žlutých oblastí. Sloupec  $x_*$  (červená oblast) je interpolován na základě hodnot žlutě vyznačených bodů  $[x_*, y_k]$ . Oblasti jsou vybírány s ohledem na princip nejmenší variance.*

### 3.3.4 Hranou řízená interpolace

Hranou řízené interpolační techniky („edge-guided“ či „edge-directed interpolation“) [27]–[30] tvoří zajímavou skupinu metod, které se snaží zohlednit fakt, že obrazová funkce se ve směru hrany mění pomaleji, než ve směru k ní kolmém. První metody ([27], [28]) z této skupiny se snažily nejprve explicitně odhadnout směr hrany ve zkoumaném okénku a na jeho základě upravit koeficienty aproximačního filtru. Nicméně tyto explicitní techniky kvantizují směr hrany do konečné množiny směrů (horizontální, vertikální a případně diagonální směry), čímž limitují shodu zvoleného modelu hrany se skutečností. Zde popisovaná metoda uvedená v [29] používá k adaptaci interpolačních koeficientů kovarianci mezi pixely zpracovávaného okénku. Klíčovou myšlenkou je fakt, že vztahy mezi pixely pro dané směry by měly zůstat stejné v původním i převzorkovaném obraze. To odpovídá tomu, že kovarianční matice původního obrazu odpovídá kovarianční matici obrazu převzorkovaného.

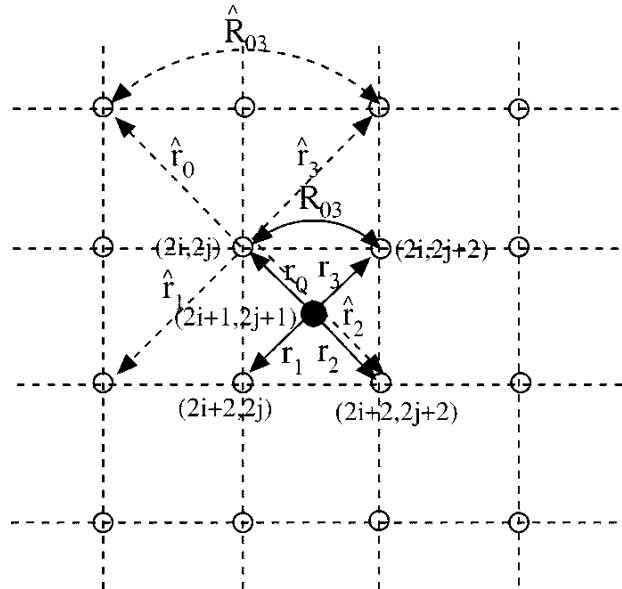
Algoritmus hranou řízené interpolace s adaptací na základě kovariance provádí převzorkování původního obrazu obrazem s dvojnásobným rozlišením. Beze ztráty obecnosti lze pohlédnout na původní obraz  $X_{ij}$  velikosti  $W \times H$  s nízkým rozlišením jako na převzorkovanou variantu cílového obrazu  $Y_{2i,2j} = X_{ij}$  velikosti  $2W \times 2H$ . Nejprve si uvědomme, jak interpolujeme bod  $Y_{2i+1,2j+1}$  s pomocí okolních bodů. Omezme se na lineární interpolaci v 4-okolí:

$$Y_{2i+1,2j+1} = \sum_{k=0}^1 \sum_{l=0}^1 \alpha_{2k+l} Y_{2(i+k),2(j+l)} \quad (72)$$

U klasické bilineární interpolace jsou váhy všech 4 okolních bodů identické a koeficienty  $\alpha_1$  až  $\alpha_4$  jsou rovny  $\frac{1}{4}$ . Jak však bylo uvedeno výše, obrazová funkce se mění pomaleji ve směru hrany a rychleji ve směru na ní kolmém. Rovnoměrné nastavení vah tudíž není optimální. Pro většinu obrazů lze předpokládat, že lze modelovat lokálně stacionárními Gaussovskými procesy. Podle klasické Wienerovy filtrační teorie [31] jsou optimální interpolační koeficienty (optimalizované podle MMSE kritéria (Minimum Mean Squared Error)) následující:

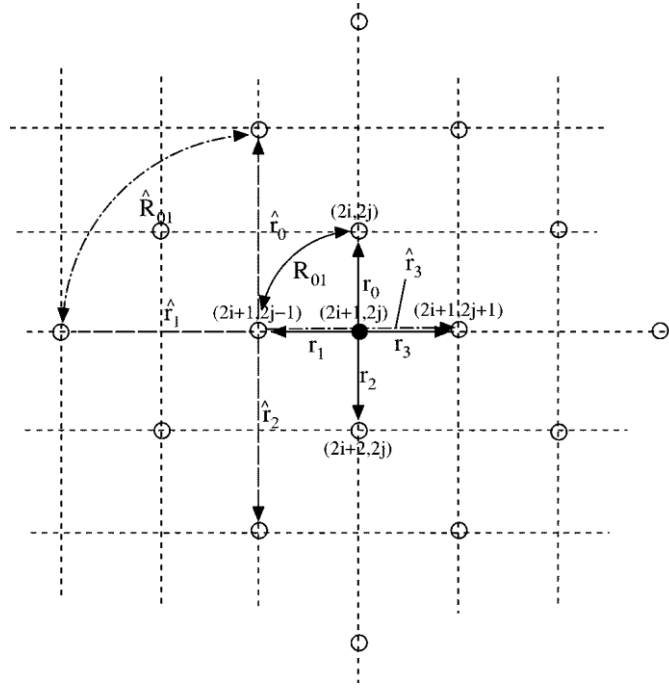
$$\vec{\alpha} = R^{-1} \vec{r} \quad (73)$$

kde  $R = [R_{kl}]$ ,  $0 \leq k, l \leq 3$  a  $\vec{r} = [r_k]$ ,  $0 \leq k \leq 3$  jsou lokální kovariance obrazu s velkým rozlišením -viz Obr. 25. Tentýž obrázek ilustruje vztah mezi kovariancemi  $\hat{R}_{kl}$ ,  $\hat{r}_k$  v obraze s nízkým rozlišením a kovariancemi  $R_{kl}$ ,  $r_k$  v obraze s vysokým rozlišením, jenž váže dvojice kovariancí v různém rozlišení, ale pro stejný směr.



Obr. 25 Vzájemné kovariance použité pro interpolaci bodu  $Y_{2i+1,2j+1}$  a geometrická dualita s kovariancemi obrazu s nižším rozlišením

Analogicky lze odvodit vztahy pro interpolaci chybějících bodů  $Y_{ij}$ , kde  $i+j$  je liché, z mřížky již známých bodů  $Y_{ij}$ , kde  $i+j$  je sudé – viz Obr. 26. Ve skutečnosti jsou obrázky Obr. 25 a Obr. 26 izomorfní až na škálovací faktor  $\sqrt{2}$  a rotaci  $\frac{\pi}{4}$ .



Obr. 26 Kovariance použité pro interpolaci bodů  $Y_{ij}$ , kde  $i+j$  je liché, z již známých bodů  $Y_{ij}$ , kde  $i+j$  je sudé. Dále je naznačena dualita s kovariancemi obrazu s nižším rozlišením

Kovariance obrazu s nízkým rozlišením lze odhadnout z obrazových dat:

$$\begin{aligned}\hat{R} &= \frac{1}{M^2} C^T C \\ \hat{\vec{r}} &= \frac{1}{M^2} C^T \vec{y}\end{aligned}\tag{74}$$

kde  $\vec{y} = [y_1, \dots, y_k, \dots, y_{M^2}]$  je vektor obsahující  $M \times M$  pixelů uvnitř lokálního okénka a  $C$  je matice  $4 \times M^2$  jejíž  $k$ -tý sloupec obsahuje 4 nejbližší sousedy bodu  $y_k$ .

Asymptoticky, pro vzorkovací interval  $h$  jdoucí k nule platí, že  $\lim_{h \rightarrow 0} R = \hat{R}$  a  $\lim_{h \rightarrow 0} \vec{r} = \hat{\vec{r}}$ . Proto předpokládáme  $R \cong \hat{R}$  a  $\vec{r} \cong \hat{\vec{r}}$  (tj. že kovariance obrazu s vysokým rozlišením odpovídá kovarianci obrazu s nízkým rozlišením). Z (73) a (74) pak:

$$\vec{a} = (C^T C)^{-1} (C^T \vec{y})\tag{75}$$

Dosazením koeficientů optimálního filtru  $\alpha_i$  do (72) pak získáme interpolovanou hodnotu v bodě  $Y_{2i+1, 2j+1}$ . Analogicky se postupuje s výpočtem chybějících vzorků  $Y_{ij}$ , kde  $i+j$  je liché.

Výhodou uvedené metody je zachování ostrých hran v převzorkovaném obraze (díky tomu, že filtrace je prováděna na daném okénku víceméně ve směru hrany). Problémy mohou vznikat u blízkých hran (např. texturované plochy), kdy je v jednom okénku obsaženo 2 a více hran. V tomto případě dochází ke vzniku falešných aliasingových artefaktů. Vizualní kvalitu převzorkovaného obrazu to neovlivní, ale pro strojové zpracování to znamená vnesení nových frekvenčních složek, které v původním obraze nebyly přítomny.

### 3.4 Ostatní metody

Kromě uvedených momentových, aproximačních a interpolačních metod subpixelové lokalizace hran existuje celá řada technik slučujících principy více uvedených metod, případně aplikujících jejich myšlenky poněkud odlišným způsobem. Bývají optimalizovány pro použití za určitých podmínek nebo pro určité typy obrazů či úloh. Příkladem může být např. metoda lokalizace za použití efektu částečné plochy pixelu.

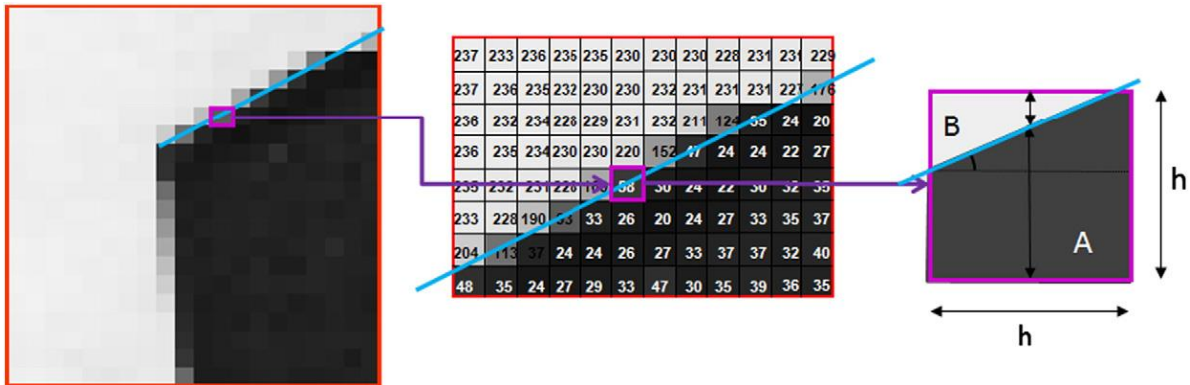
#### 3.4.1 Lokalizace hran za použití efektu částečné plochy pixelu (partial area effect)

Princip metody publikované v [32] je podobný metodě gray-level momentů, nicméně díky rozdělení zkoumané oblasti do 3 částí a použití velkého množství heuristik jsou získané výsledky výrazně lepší, než u uvedené momentové metody.

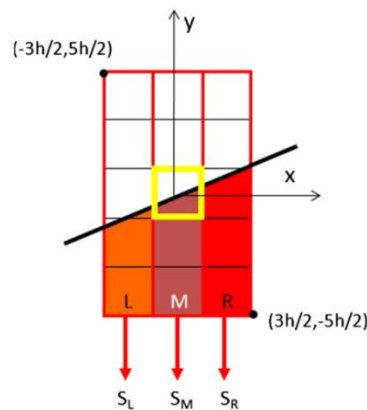
Předpokládejme ideální skokovou hranu o intenzitách  $A$ , resp.  $B$  na opačných stranách hrany a procházející pixelu  $(i,j)$  viz Obr. 27. Pak lze pro intenzitu pixelu psát:

$$F_{i,j} = \frac{AS_A + BS_B}{h^2} = B + \frac{A - B}{h^2} S_A \quad (76)$$

Kde  $S_A$  a  $S_B$  jsou plochy pixelu pokryté intenzitou  $A$ , respektive  $B$  a  $h$  je šířka pixelu. Pro celkovou plochu pixelu samozřejmě platí  $h^2 = S_A + S_B$ .



Obr. 27 Partial area effect – určení intenzity pixelu na základě modelu skokové hrany dělící pixel na dvě oblasti o intenzitách  $A$  a  $B$



Obr. 28 Okénko pro určení parametrů hrany procházející pixelu  $(i,j)$

Dále se omezme na skokové hrany se směrnicí v prvním oktantu ( $0^\circ - 45^\circ$ ) tvaru

$$y = a + bx \quad (77)$$

Uvažujme okénko o rozměrech  $3 \times 5$  pixelů se středem v  $(i, j)$  (). 5 řádků je třeba proto, aby všechny uvažované hrany protínaly okénko na levém i pravém okraji. Označme  $S_L$ ,  $S_M$  a  $S_R$  součty intenzit pixelů v jednotlivých sloupcích:

$$S_L = \sum_{n=j-2}^{j+2} F_{i-1,n} = 5B + \frac{A-B}{h^2} L \quad (78)$$

$$S_M = \sum_{n=j-2}^{j+2} F_{i,n} = 5B + \frac{A-B}{h^2} M \quad (79)$$

$$S_R = \sum_{n=j-2}^{j+2} F_{i+1,n} = 5B + \frac{A-B}{h^2} R \quad (80)$$

Kde  $L$ ,  $M$  a  $R$  reprezentují plochy oblastí pod hranou v daném sloupci. Tyto plochy lze analyticky vyjádřit:

$$L = \int_{-3\frac{h}{2}}^{-\frac{h}{2}} \left( a + bx + 5\frac{h}{2} \right) dx = ah - bh^2 + \frac{5}{2}h^2 \quad (81)$$

$$M = \int_{-\frac{h}{2}}^{\frac{h}{2}} \left( a + bx + 5\frac{h}{2} \right) dx = ah + \frac{5}{2}h^2 \quad (82)$$

$$R = \int_{\frac{h}{2}}^{3\frac{h}{2}} \left( a + bx + 5\frac{h}{2} \right) dx = ah + bh^2 + \frac{5}{2}h^2 \quad (83)$$

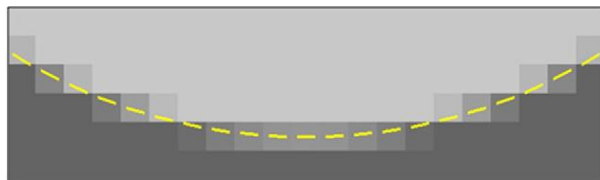
Odhadneme-li intenzity  $A$  a  $B$  jako průměr jasu horního, resp. spodního řádku okénka získáváme:

$$\begin{aligned} A &= \frac{1}{3} (F_{i-1,j+2} + F_{i,j+2} + F_{i+1,j+2}) \\ B &= \frac{1}{3} (F_{i-1,j-2} + F_{i,j-2} + F_{i+1,j-2}) \end{aligned} \quad (84)$$

Dále pak z (79), (82) a (84), resp. z (78), (80), (81), (83) a (84) získáme vztahy pro parametry hrany  $a$  a  $b$ :

$$\begin{aligned}
 a &= \frac{2S_M - 5(A + B)}{2(A - B)} h \\
 b &= \frac{S_R - S_L}{2(A - B)}
 \end{aligned}
 \tag{85}$$

Určené parametry  $a$  a  $b$  určují jak přesnou polohu hrany, tak její normálu.



*Obr. 29 Ukázka určení polohy a orientace hrany popsanou metodou*

Uvedená analýza platila pro hrany se směrnicí  $0^\circ - 45^\circ$ . Analogicky lze odvodit vztahy pro hrany se směrnici v ostatních oktantech. U hran se strmým sklonem (blíže k ose  $y$ ) se zamění význam os  $x$  a  $y$  a uvažuje se okénko široké 5 pixelů a vysoké 3. V článku je dále odvozeno zobecnění pro zakřivené hrany aproximované polynomem 2. stupně a řešení pro blízké hrany (2 a více hran uvnitř okénka). Dle publikovaných dat dosahuje metoda s relativně nízkými výpočetními nároky pozoruhodných výsledků. Cenou za to je nutnost použití několika heuristik, jež na základě orientace hrany a jejího okolí větvi algoritmus do velkého množství variant.

## 4 Závěr

V práci byly představeny různé způsoby subpixelové detekce a lokalizace hran. Většinu algoritmů lze zařadit do jedné ze 3 skupin: momentové, aproximační a interpolační.

Momentové metody mají ve srovnání s aproximačními technikami obecně nižší výpočetní náročnost a díky charakteru použitých integrálních operátorů jsou odolné vůči šumu. Jsou dobrým kompromisem, vyžadujeme-li vyšší rychlost a odhad polohy s rozumnou přesností. Srovnáme-li jednotlivé metody z této kategorie, pak co se týče chyb způsobených diskretizací i odolnosti vůči šumu, vychází metoda prostorových momentů oproti gray-level momentům o něco lépe. Zernikovy momenty obsahují podobně jako prostorové momenty informaci o rozložení jasu uvnitř zkoumané oblasti, tudíž jejich použití sdílí výhody prostorových momentů. Navíc jsou však v této oblasti ortogonální a tudíž odstraňují redundanci v extrahovaných datech a lokalizaci hrany lze provést s cca  $\frac{2}{3}$  výpočetními náklady oproti klasickým prostorovým momentům. Jsou proto v rámci momentových metod upřednostňovány. Existují i varianty momentových metod, kdy jsou momenty použité pro vyčíslení parametrů hrany odhadovány analyticky po proložení oblasti parametrickou plochou (např. kubickou spline plochou v [15]). Výsledky těchto metod jsou však na základě článků dle mého názoru neprůkazné a proto nejsou v pojednání zařazeny.

Skupina aproximačních metod pracujících na principu curve-fitting (případně surface-fitting) hledá takové parametry modelu hrany, které co nejlépe odpovídají obrazovým datům. Prokládání hrany ERF funkcí ve 2D je modelem, který nejlépe odpovídá skutečnosti a procesu formování obrazu na senzoru. Díky tomu dává metoda špičkové výsledky a to i za přítomnosti šumu. Cenou za to jsou velmi vysoké výpočetní nároky. Je-li však přesnost lokalizace hlavním měřítkem, je tato metoda nejlepší volbou.

Interpolační metody jsou za použití klasických interpolačních schémat (bilineární, bikvadratická či bikubická interpolace) rychlé. Nicméně jsou velmi náchylné na šum a přesnost lokalizace není ve srovnání s podobně výpočetně náročnými momentovými metodami nejlepší. Rovněž rozostřování hran je zde problém. Smysl dává použití různých adaptivních a nelineárních technik. Doba zpracování, náchylnost na šum a kvalita výsledků závisí na použitém interpolačním algoritmu. Je-li možné implementovat interpolační algoritmus tak, aby reflektoval očekávaný tvar výstupu a používal apriorní znalosti o scéně, jedná se o akceptovatelné řešení. Uvedená metoda hranou řízené interpolace je zajímavou alternativou momentových metod.

V případě, že je prioritou rychlost, stojí za úvahu naposledy uvedená a nezařazená metoda efektu částečné plochy pixelu. Uvedené výsledky vypadají velmi slibně, bohužel chybí srovnání s ostatními rozebíranými metodami.

Všechny uvedené metody kromě hranou řízené interpolace předpokládají symetrický model hrany, kdy má nástup jasu od tmavé úrovně k inflexnímu bodu hrany stejný průběh, jako druhá část křivky od inflexního bodu k maximu. Tento předpoklad zavádí u nesymetrických hran, které vzniknou např. na rozhraní válcové plochy a pozadí ofset. U měření vyžadujících extrémně

přesnou lokalizaci je třeba vzít v potaz i tento efekt. Tento problém však nebyl v současné literatuře téměř vůbec adresován.



## 5 Reference

- [1] Giuseppe Papari, Nicolai Petkov: „Edge and line oriented contour detection: State of the art“, Image and Vision Computing 29 (2011), p. 79–103, Elsevier
- [2] Haldo Spontón, Juan Cardelino: „A Review of Classic Edge Detectors“, Image Processing Online, 2015-06-07
- [3] D. Marr, E. C. Hildreth: „Theory of edge detection“, Proceedings of the Royal Society, London B, 207 (1980), p. 187-217
- [4] R. M. Haralick: „Digital step edges from zero crossing of the second directional derivatives“, IEEE Transactions on Pattern Analysis and Machine Intelligence, 6 (1984), p. 58-68, <http://dx.doi.org/10.1109/TPAMI.1984.4767475>.
- [5] John Canny: „A Computational Approach to Edge Detection“, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, November 1986, IEEE
- [6] OpenCV: „Canny Edge Detector“, [https://docs.opencv.org/3.1.0/da/d5c/tutorial\\_canny\\_detector.html](https://docs.opencv.org/3.1.0/da/d5c/tutorial_canny_detector.html)
- [7] Rafael Grompone von Gioi, Gregory Randall: „A Sub-Pixel Edge Detector: an Implementation of the Canny/Devernay Algorithm“, Image Processing On Line, 2017-11–28, <https://doi.org/10.5201/ipol.2017.209>
- [8] Peter Rockett: „The Accuracy of Sub-Pixel Localisation in the Canny Edge Detector“, Proceedings of the British Machine Vision Conference 1999, p. 392-401
- [9] R. Machuca, A. L. Gilbert: „Finding edges in noisy scenes“, IEEE Transactions on Pattern Analysis and Machine Intelligence 3(1) (1981), p. 103–111
- [10] A. J. Tabatabai, O. R. Mitchell: „Edge location to subpixel values in digital imagery“, IEEE Transactions on Pattern Analysis and Machine Intelligence 6(2) (1984), p. 188–201
- [11] A. Tabatabai: "Edge location and data compression for digital imagery," Ph.D. dissertation, School of Electrical Engineering, Purdue University, 1981
- [12] E. P. Lyvers, O. R. Mitchell, M. L. Akey: „Subpixel measurements using a moment-based edge operator“, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No. 12, Dec 1989
- [13] Sugata Ghosal, Rajiv Mehrotra: „Orthogonal Moment Operators for Subpixel Edge Detection“, Pattern Recognition, Vol. 26, No. 2, p. 295-306, 1993
- [14] Fengtao Xiang, Zhengzhi Wang, Xingsheng Yuan: „Subpixel Edge Detection: An Improved Zernike Orthogonal Moments Method“, 5th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2013, IEEE
- [15] Qiucheng Suna, Yueqian Houb, Qingchang Tanc, Chunjing Lia, Ming Liu: „A robust edge detection method with sub-pixel accuracy“, Optik 125 (2014), p. 3449–3453, Elsevier
- [16] J. Ye, G. Fu, U.P. Poudel: „High-accuracy edge detection with blurred edge model“, Image and Vision Computing 23 (5) (2005), p. 453–467, Elsevier

- [17] M. Hagara, P. Kulla: „Edge Detection with Sub-pixel Accuracy Based on Approximation of Edge with Erf Function“, Radioengineering, Vol. 20, No. 2, June 2011, p. 516-524
- [18] Qiucheng Suna, Yueqian Houb, Qingchang Tan: „A subpixel edge detection method based on an arctangent edge model“, Optik 127 (2016), p. 5702–5710, Elsevier
- [19] V.S. Nalwa, T.O. Binford: "On detecting edges", IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (6) (1986), p. 699–714
- [20] Anna Fabijańska: „Gaussian-Based Approach to Subpixel Detection of Blurred and Unsharp Edges“, Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, p. 641–650
- [21] Anna Fabijańska: „Subpixel Edge Detection in Blurry and Noisy Images“, International Journal of Computer Science and Applications, Vol. 12, No. 2, p. 1–19, 2015
- [22] Anna Fabijańska: „A Survey of Subpixel Edge Detection Methods for Images of Heat-Emmiting Metal Specimens“, International Journal of Applied Mathematics and Computer Science 22 (2012), No. 3, p. 695–710
- [23] T. Ridler, S. Calvard: „Picture thresholding using an iterative selection method", IEEE Transactions on Systems, Man, and Cybernetics 8 (1978), p. 630-632  
<http://dx.doi.org/10.1109/TSMC.1978.4310039>
- [24] C. Steger: „Subpixel-precise extraction of lines and edges“, International Archives of Photogrammetry and Remote Sensing. Vol. XXXIII, Part B3, p. 141-156, Amsterdam 2000
- [25] A. Harten, B. Engquist, S. Osher, S.R. Chakravarthy: „Uniformly high order accurate essentially non-oscillatory schemes“, Journal of Computational Physics 71 (1987), p. 231–303
- [26] T. Hermosilla, E. Bermejo, A. Balaguer, L.A. Ruiz: „Non-linear fourth-order image interpolation for subpixel edge detection and localization“, Image and Vision Computing 26 (2008), Elsevier, p. 1240–1248
- [27] V. R. Algazi, G. E. Ford, R. Potharlanka: „Directional interpolation of images based on visual properties and rank order filtering“, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing 1991, Vol. 4, p. 3005–3008
- [28] J. Allebach, P. W. Wong: „Edge-directed interpolation“, Proceedings of the IEEE International Conference on Image Processing 1996, Vol. 3, p. 707–710
- [29] X. Li, M. T. Orchard: „New edge-directed interpolation“, IEEE Transactions on Image Processing, Vol. 10, No. 10, p. 1521–1527, October 2001
- [30] Lei Zhang, Xiaolin Wu: „An edge-guided image interpolation algorithm via directional filtering and data fusion“, IEEE Transactions on Image Processing, Vol. 15, No. 8, p. 2226-2238, August 2006
- [31] N. Jayant, P. Noll: „Digital Coding of Waveforms: Principles and Applications to Speech and Video“, Englewood Cliffs, Prentice-Hall, 1984

- [32] Agustín Trujillo-Pino, Karl Krissian, Miguel Alemán-Flores, Daniel Santana-Cedr s:  
„Accurate subpixel edge location based on partial area effect“, Image and Vision Computing  
2013, Vol. 31, Issue 1, p. 72-90