

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

VYHLEDÁNÍ VÝZNAČNÝCH BODŮ V RASTROVÉM OBRAZE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

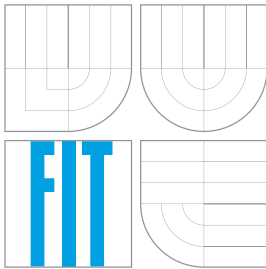
AUTHOR

Bc. PETR KANĚČKA

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

VYHLEDÁNÍ VÝZNAČNÝCH BODŮ V RASTROVÉM OBRAZE

INTEREST POINTS DETECTION IN RASTER IMAGE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. PETR KANĚČKA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. ADAM HEROUT, Ph.D.

BRNO 2007

Zadání diplomové práce

Řešitel: **Kaněčka Petr, Bc.**

Obor: Počítačová grafika a multimédia

Téma: **Vyhledání význačných bodů v rastrovém obraze**

Kategorie: Počítačová grafika

Pokyny:

1. Seznamte se s problematikou zpracování obrazu a počítačového vidění.
2. Popište použití význačných bodů ("rohů") ve zpracování obrazu.
3. Prostudujte a popište existující algoritmy vyhledání význačných bodů.
4. Vyberte vhodné metody a implementujte je.
5. Vyhodnoťte a srovnajte vlastnosti implementovaných metod vyhledání význačných bodů.
6. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování projektu; prezentujte projekt plakátkem.

Literatura:

- dle pokynů vedoucího

Při obhajobě semestrální části diplomového projektu je požadováno:

- body 1.-3.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Herout Adam, Ing., Ph.D.,** UPGM FIT VUT

Datum zadání: 28. února 2006

Datum odevzdání: 22. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 66 Brno, Božetěchova 2

L.S.



doc. Dr. Ing. Pavel Zemčík
vedoucí ústavu

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Bc. Petr Kaněčka**
Id studenta: 49164
Bytem: Přísecká 115, 257 68 Dolní Kralovice
Narozen: 07. 06. 1981, Vlašim
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
diplomová práce

Název VŠKP: Vyhledání význačných bodů v rastrovém obraze
Vedoucí/školitel VŠKP: Herout Adam, Ing., Ph.D.
Ústav: Ústav počítačové grafiky a multimédií
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě	počet exemplářů: 1
elektronické formě	počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3


Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....

Nabyvatel


.....

Autor

Abstrakt

Tento dokument se zabývá možnostmi hledání význačných bodů v obraze, především rohovými detektory. Spousta aplikací z oblasti počítačového vidění potřebuje takové body pro svůj výpočet jako nezbytný krok při zpracování obrazu. Text popisuje, proč je pro tyto aplikace výhodné takové body najít a také základní metody, jakým způsobem je lze detekovat. Na závěr jsou porovnány vlastnosti jednotlivých metod.

Klíčová slova

význačné body, rohový detektor, rozeznávání objektů, počítačové vidění, předzpracování obrazu, navigace robota, prostředí robota, panorama, Moravcův operátor, Harrisův operátor, Plesseyův operátor, Trajkovićův operátor, SUSAN, USAN, CSS, prostor měřítek

Abstract

This document deals with an image points of interest detection possibilities, especially corner detectors. Many applications which are interested in computer vision needs these points as their necessary step in the image processing. It describes the reasons why it is so useful to find these points and shows some basic methods to find them. There are compared features of these methods at the end.

Keywords

interest points, keypoint, dominant point, corner detector, object recognition, computer vision, image preprocessing, stereo matching, motion tracking, video tracking, match moving, robot navigation, robot's environment, panorama stitching, Moravec operator, Harris operator, Plessey operator, Trajkovic operator, SUSAN, USAN, CSS, scale space

Citace

Petr Kaněčka: Vyhledání význačných bodů v rastrovém obraze, diplomová práce, Brno, FIT VUT v Brně, 2007

Vyhledání význačných bodů v rastrovém obraze

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Adama Herouta, Ph.D.

.....
Petr Kaněčka

22. 5. 2007

© Petr Kaněčka, 2007.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Problematika detekce význačných bodů	5
2.1	Formální význam význačných bodů	5
2.2	Použití význačných bodů	5
2.3	Požadavky na rohový detektor	6
2.4	Hodnocení a porovnávání rohových detektorů	8
2.5	Historie vývoje detektorů	10
2.6	Metody detekce význačných bodů	11
2.6.1	Metody založené na hranách	11
2.6.2	Topologické metody	11
2.6.3	Autokorelační metody	12
2.6.4	Alternativní metody	12
2.7	Algoritmus detekce rohů	13
2.7.1	Aplikace rohového operátoru	14
2.7.2	Prahování mapy rohovitosti	14
2.7.3	Potlačení nemaximálních hodnot	14
3	Rohové detektory	15
3.1	Moravcův detektor	15
3.2	Harris a Stephens / Plesseyův detektor	18
3.3	Trajkovićův detektor (4-okolí)	23
3.3.1	Způsob ohodnocení rohovitosti	23
3.3.2	Víceraštrový přístup	27
3.4	Smallest Univalve Segment Assimilating Nucleus	28
3.5	Curvature Scale Space	30
3.5.1	Reprezentace obrazových dat ve více měřítkách	30
3.5.2	Nalezení hran	32
3.5.3	Odhad celkového zakřivení	36
3.5.4	Způsob detekce rohů	41
4	Praktické zhodnocení detektorů	44
4.1	Moravcův detektor	45
4.1.1	Nestabilita	46
4.1.2	Citlivost na šum	46
4.1.3	Hodnocení	46

4.2	Harris a Stephens / Plesseyův detektor	47
4.2.1	Stabilita	47
4.2.2	Citlivost na šum	48
4.2.3	Hodnocení	48
4.3	Smallest Univalve Segment Assimilating Nucleus	49
4.3.1	Nestabilita	49
4.3.2	Odolnost vůči šumu	49
4.3.3	Hodnocení	51
4.4	Curvature Scale Space	51
4.4.1	Stabilita	51
4.4.2	Odolnost vůči šumu	52
4.4.3	Hodnocení	52
5	Závěr	53
5.1	Zhodnocení výsledků	53
5.2	Možnosti dalšího vývoje	53
	Literatura	55

Kapitola 1

Úvod

Mnoho dnešních aplikací potřebuje znát souvislosti mezi dvěma nebo více obrazy, aby z nich mohly extrahovat informace. Často jde o po sobě jdoucí snímky v rámci jedné video sekvence, ze kterých je možné extrahovat informace týkající se hloubky objektů v prostředí či rychlosti kamery. Hrubá síla v podobě metody porovnávání všech pixelů ve dvou obrazech je výpočetně nedostupná pro většinu aplikací. Je tedy vhodné určit několik bodů v obraze, které tomuto účelu vyhoví a porovnávat pouze lokality, které jsou v něčem zajímavé. Taková místa nazýváme *význačnými body* a lze je lokalizovat s použitím detektoru význačných bodů. Nalezení vztahu mezi obrazy se potom provádí s využitím pouze těchto bodů. Tento přístup významně redukuje požadovaný výpočetní čas. V této práci budeme nejčastěji za význačné body považovat body rohové.

Význačné body bývají velmi stabilní místa v obraze. Této stability lze využít v aplikacích různého druhu – při spojování panoramatických fotografií, při rozeznávání objektů, při detekci pohybu atd. Význačné body musí co nejlépe odolávat různým změnám v obraze – změně v osvětlení, změně pohledu do scény, či jiným rušivým prvkům, jako je např. zkraslení obrazového signálu šumem.

Kapitola 2 nás nejprve seznámí s některými důležitými pojmy, popíše smysl detekce význačných bodů v několika aplikacích a definuje požadavky na ideální detektor takových bodů. Dále nahlédneme do historie vývoje a řešení této problematiky, které vyústilo v několik od sebe rozlišitelných přístupů k detekci význačných bodů. Popíšeme si základní kroky algoritmu detekce, které jsou vlastní mnohým detektorům.

Hlavní částí této práce je kapitola 3 s podrobným popisem principu několika významných detektorů, kterými jsou Moravcův operátor, Harris/Plesseyův operátor, SUSAN detektor, CSS detektor a Trajkovićův operátor. Za touto kapitolou následuje několik praktických ukávek aplikace prvních čtyř jmenovaných detektorů. Detektory jsou zde podrobeny několika základním testům, na základě kterých jsou popsány jejich praktické vlastnosti. Podle těchto vlastností se můžeme později rozhodnout, který detektor zvolíme pro naši případnou aplikaci, jež bude detekci význačných bodů vyžadovat.

Tato práce navazuje na stejnojmenný semestrální projekt. Z větší části byla přejata kapitola 2, která byla rozšířena o sekce 2.4 a 2.7. Kapitola 3 byla také přejata a rozšířena o Trajkovićův, SUSAN a CSS rohový detektor.

Kapitola 2

Problematika detekce význačných bodů

2.1 Formální význam význačných bodů

Roh můžeme definovat jako průnik dvou hran. Roh můžeme také definovat jako bod, v jehož blízkém okolí se objevují dva dominantní směry hran. Význačný bod je bod v obraze, který má pevně danou pozici a lze ho vždy dobře detekovat. To ale znamená, že význačným bodem může být kromě rohu také například izolovaný bod, zakončení čáry nebo bod, který je součástí nějaké křivky, ale v daném bodě dosahuje její zakřivení lokálního maxima. V praxi většina tzv. rohových detektorů detekuje spíše obecnější význačné body než skutečné rohové body. Pokud chceme získat skutečné rohové body, je nutné udělat lokální analýzu detekovaných význačných bodů, abychom určili, které z nich jsou opravdovými rohovými body a které ne.

2.2 Použití význačných bodů

Použití význačných bodů (a tedy i rohových detektorů) je důležité v hledání korespondujících bodů napříč množinou obrazů. Jde o klíčový krok ve zpracování obrazu a v aplikacích počítačového vidění. Mezi nejdůležitější příklady použití patří:

- *stereo matching*,
- registrace obrazů,
- spojování panoramatických fotografií,
- detekce a rozeznávání objektů,
- *motion tracking*,
 - *video tracking*,

– *match moving*,

- detekce pohybu,
- navigace robota.

Stereo matching je metoda identifikace stejných bodů v různých snímcích stejné scény.

Registrace obrazů [9] nebo též lícování obrazů představuje nalezení vhodné geometrické transformace mezi dvěma obrazy, která ztotožní pozici a prostorovou orientaci odpovídajících si objektů obou obrazů. Nalezená geometrická transformace převádí oba registrované obrazy do společného souřadnicového prostoru.

Registrace se používá pro zvětšování zorného pole senzorů, zlepšování kvality obrazových dat znehodnocených zkreslením, sledování změn ve snímaném objektu.

Video tracking je proces lokalizace pohybujícího se objektu (nebo více objektů) v čase při použití kamery.

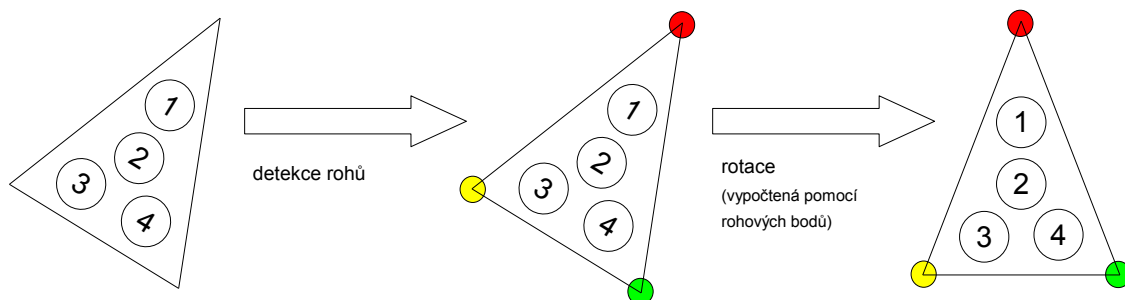
Match moving je speciální efektní technologie, která umožňuje vložení virtuálního objektu do reálné stopáže na správnou pozici, ve správném měřítku a se správným pohybem v relaci s ostatními objekty scény.

Obrázek 2.1 představuje část hypotetického systému, který ilustruje, jakým způsobem lze rohový detektor použít na automatické lince. Na této lince se plní dárková krabička ve tvaru trojúhelníku čtyřmi druhy čokolády. Krabičky musí být správně umístěné na dopravním pásu, aby se zaručilo správné vložení čokolád dovnitř krabiček. Horní kamera snímá obraz každé krabičky na pásu tak, jak pod ní po pásu přijede a počítač snímek porovná s uloženým vzorem správně zarovnané krabičky. Díky nalezení rohů každé krabičky lze snadno vypočítat, jak je třeba nesprávně zarovnanou krabičku otočit.

2.3 Požadavky na rohový detektor

Existuje několik důležitých vlastností, které jsou po rohovém detektoru zpravidla požadovány:

1. detektor by měl najít všechny správné rohy,
2. neměl by detekovat žádné nesprávné rohy,
3. místa rohů by měla být přesně lokalizována,



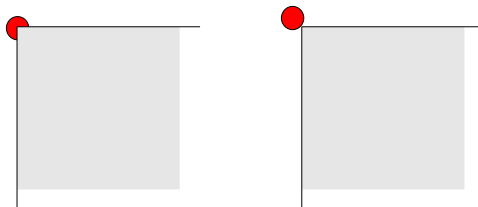
Obrázek 2.1: Část systému se zarovnáváním krabiček na dopravním pásu.

4. detektor by měl být stabilní,
5. detektor by měl být robustní s ohledem na šum,
6. detektor by měl být výpočetně efektivní.

Stabilita znamená, že by detektor měl na více podobných snímcích téže scény vyhledat množiny bodů s co největší společnou podmnožinou a s co nejmenšími rozdíly. Někdy se této vlastnosti říká *míra opakovatelnosti*.

Pojem rohu v šedotónovém obrázku není přesně definován, přesto v mnoha obrazech intuitivně cítíme místa, kde by se rohy měly vyskytovat. Na takových obrazech pak můžeme zkusit aplikovat různé rohové detektory a hodnotit je.

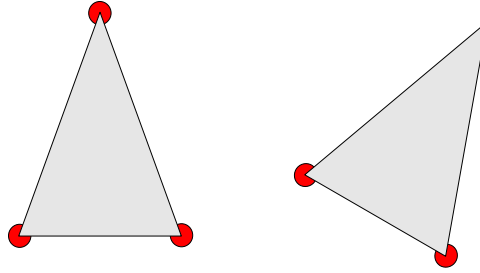
Lokalizace rohu má vztah k tomu, s jakou přesností je nalezena pozice rohu. To může být kritické pro některé aplikace, které vyžadují precizní zarovnání více obrazů, například při registraci lékařských snímků. Na obrázku 2.2 je ukázána správně a nesprávně detekovaná pozice rohu, která je označena červeným kolečkem. Přesná lokalizace je vhodnou vlastností pro všechny aplikace. V některých typech aplikací však není nutnou podmínkou.



Obrázek 2.2: Správná a nesprávná lokalizace rohového bodu

Představme si robota s jednou kamerou, který bloudí v chodbách. Mnoho přístupů v mobilní robotice potřebuje analyzovat snímky zachycené kamerou, aby se dalo interpretovat

robotovo prostředí. Prvním krokem ve spoustě takových aplikací bývá potřeba nalezení korespondujících bodů mezi snímky. Dva po sobě jdoucí snímky budou velmi podobné, ale budou přeci jen mírně rozdílné kvůli pohybu robota, změně osvětlení a dalším vlivům. Rohový detektor, který je robustní vůči takovým změnám, dokáže najít stejná místa a říkáme, že je stabilní. Obrázek 2.3 demonstruje příklad, kdy se nepodaří nalézt všechny rohy na pootočeném objektu. Podíl počtu nalezených rohů ku celkovému počtu rohů mezi snímky se anglicky nazývá *repeatability rate*. Obrázek 2.3 představuje tedy $2/3$ *repeatability rate*.



Obrázek 2.3: Chybující rohový detektor, který po rotaci objektu nedetekoval jeden z rohů.

V nasnímaných obrázcích se často nevyhnutelně objevuje šum, proto musí být rohový detektor robustní i vůči šumu. Detektor by neměl falešně označit šum za rohové body a šum by měl mít minimální vliv na určení pozice rohu.

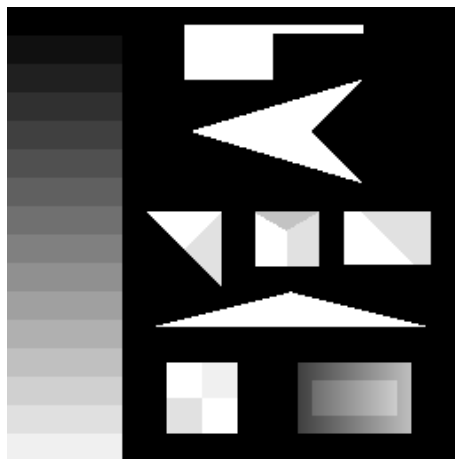
V mnoha real-time aplikacích je kritická výpočetní náročnost rohového detektoru, takové aplikace jsou často nuceny volit jednodušší, avšak méně robustní, rohový detektor. Některé rohové detektory byly vyvinuty speciálně pro časově kritické aplikace.

2.4 Hodnocení a porovnávání rohových detektorů

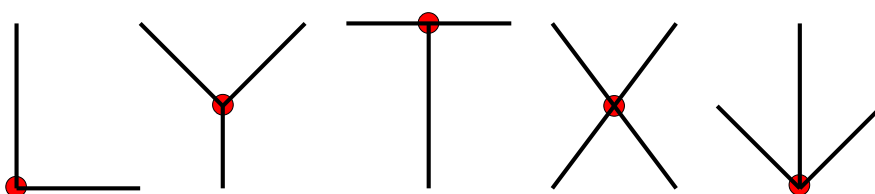
Při výběru detektoru je nejdůležitější znát náš cíl jeho aplikace. Použití rohového detektoru přímo předurčuje, který detektor si vybereme, protože každý má určité klady a zápory z hlediska požadavků, které byly popsány výše. Například při zarovnávání obrazu¹ je kritická míra opakovatelnosti, ale na detekci všech rohů naopak nezáleží, protože pokud je v obraze nalezen dostatečný počet význačných bodů, stačí to k jeho přesnému zarovnání. Naproti tomu detekce falešných rohů v aplikaci detekující objekty v obraze může vést ke zkreslenému popisu takovýchto objektů a k jejich nesprávné klasifikaci.

Obrázek 2.4 obsahuje několik typů rohů (uzly ve tvaru písmene L, Y, T, X a šipky, které jsou vyobrazeny na obr. 2.5) a je velmi často používán k hodnocení schopnosti rohového

¹Zarovnání obrazu se vztahuje k nalezení nejlepší transformace z daného referenčního obrazu do nového obrazu scény. Tento proces je často provázený podobnostními měřeními mezi obrazy, výpočetně založený na obrazových datech. Tyto metody pro výpočet podobnosti jsou obvykle příliš pomalé. Při výpočtu podobnosti se namísto použití dat z celého obrazu používá pouze podmnožina pixelů, aby se zajistila dostatečná rychlost výpočtu za cenu ztráty přesnosti.

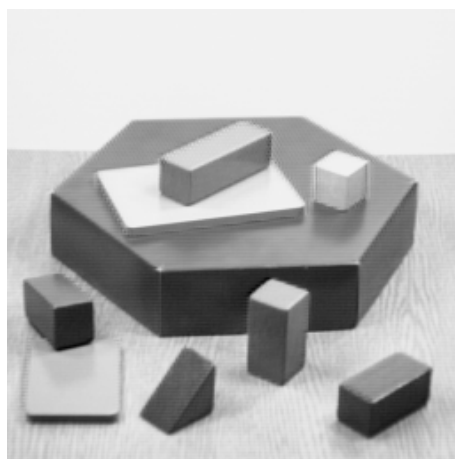


Obrázek 2.4: Uměle vytvořený obrázek, který obsahuje různé typy rohů.



Obrázek 2.5: Příklad rohů ve tvaru písmene L, Y, T, X a šipky.

detektoru reagovat na každý z těchto typů rohů. Tento testovací obrázek také obsahuje rohy tvořené sadou šedotónů různých intenzit.



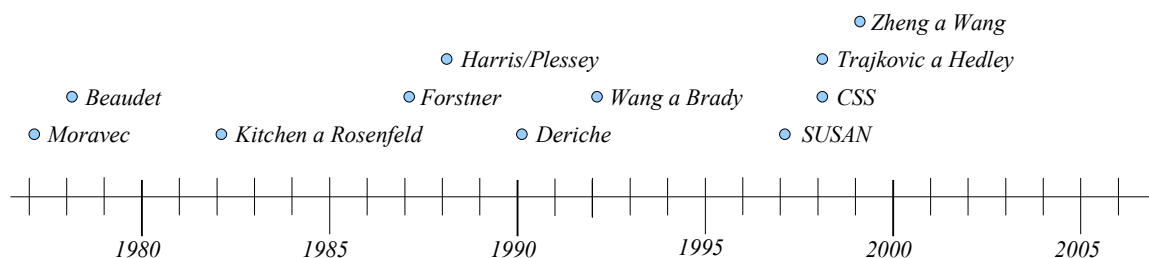
Obrázek 2.6: Reálná scéna.

Reálné nasazení detektorů rohů bude pracovat nad reálnými obrazy, proto je velmi důležité sledovat, jak si s nimi detektory poradí. Obrázek 2.6 představuje reálnou scénu, ve

kteře jsou intuitivně zřejmá místa rohů, protože je zde homogenní pozadí a jsou zde téměř homogenní odstíny a textury jednotlivých objektů. Obrazy toho typu, kdy si sami můžeme scénu alespoň vhodně nasvítit, či vhodně rozmístit objekty, se používají při snímání ve výrobním procesu a jsou proto také reprezentativním druhem. Dalším typem obrazů jsou pak snímky reálných prostředí bez jakékoliv možnosti zásahu do scény či změny osvětlení, které bývají náročnější na zpracování a najdou uplatnění např. v kamerových systémech či v budoucích robotických aplikacích. Z těchto reálných snímků je patrné, proč dosud nebyl formalizován pojem rohu. Je totiž velmi málo pravděpodobné, že by se i dva lidé naprosto shodli v počtu a umístění rohů na snímku složitější scény.

2.5 Historie vývoje detektorů

V posledních dvaceti letech bylo detekci rohů v obraze věnováno mnoho pozornosti jako klíčovému problému při zpracování obrazu v počítačovém vidění. Bylo publikováno velké množství článků, které se zabývají rohovými detektory, první z nich se objevily na konci sedmdesátých let minulého století. Zuniga a Haralick přizpůsobili souvislý povrch malému sousedství každého bodu a brali v úvahu poměr změny gradientu v jednotlivých směrech. Wang a Brady vyzorovali, že celkové zakřivení šedotónového obrázku je úměrné druhé derivaci ve směru hranové normály a nepřímo úměrné k intenzitě hrany. Moravec definoval sféru zájmu bodů jako body, ve kterých je možné sledovat velké změny intenzity (jasu) v každém směru. Harris a Stephens použili derivaci obrazu k odhadu autokorelace obrazu. Rangarajan, Shah a Brackle našli optimální funkci reprezentující rohový detektor, která když se konvoluje s šedotónovou funkcí, vynáší maximum v rohových bodech. Trajkovic a Hedley implementovali přímočarou vlastnost rohů, která říká, že změna jasu by měla být vysoká ve všech směrech. Davies detekoval rohy způsobem, který je založen na zobecněné Houghově transformaci. Xie vymyslel cenovou funkci, aby zachytil různé žádoucí charakteristiky rohů a zlepšoval detekci rohů jako problém cenové optimalizace. Smith a Brady navrhli neobvyklý algoritmus pro detekci rohů známý jako SUSAN corner detector založený na jasovém porovnávání.



Obrázek 2.7: Historie výrazných rohových detektorů

I přesto, že se detekce rohů jeví na první pohled jako jednoduchá úloha, je tento problém stále otevřen. Neexistuje žádný univerzální „dobrý“ rohový detektor.

2.6 Metody detekce význačných bodů

Spousty různých detektorů význačných bodů bylo navrženo se značným vymezením toho, čím jsou ony význačné body v obraze zajímavé. Některé detektory hledají oblasti s velkou změnou struktury (textury), jiné lokalizují rohové body. Rohové body jsou zajímavé tím, že jsou tvořeny ze dvou nebo více hran a hrany obvykle definují hranici mezi dvěma různými objekty nebo částmi stejného objektu.

Většina populárních rohových detektorů pracuje přímo na šedotónových snímcích. Nepotřebují nejprve extrahovat hrany před tím, než detekují rohy, což by mohlo vést k nejednoznačné struktuře rohových bodů. Rohové detektory, které závisí pouze na hraniční analýze, trpí chybami vzniklými při segmentaci hran. Obecný hranový detektor často nemůže správně lokalizovat hranové body kolem rohů kvůli efektu zakulacení rohů, který vede k chybám při hledání rohů.

Existují odlišné přístupy k detekci rohů a dají se částečně kategorizovat do tří základních proudů: metody založené na hranách, topologické metody a autokorelační metody. Některé detektory se kvůli svým vlastnostem dají zařadit do více kategorií, některé nespadají do žádné kategorie, rozdělení je proto poněkud svévolné, avšak patří mezi nejčastější.

2.6.1 Metody založené na hranách

Kitchen a Rosenfeld byli v roce 1982 první, kdo aplikovali diferenciální geometrii pro hledání rohů. Navrhli ohodnocení rohovitosti každého bodu tak, že se vypočítá změna směru gradientu (druhá derivace) podél hrany, která je vážena (násobena) velikostí lokálního gradientu. Rohy jsou pak identifikovány pomocí lokálních maxim. Tento výpočet lze efektivně provést potlačením odezev mimo maxima na velikosti gradientu před násobením s druhou derivací. Takový detektor pak trpí velkou citlivostí, protože se spoléhá na druhou derivaci obrazu a ukázalo se, že má i poměrně špatnou stabilitu a lokalizaci.

Populární rohový detektor, který byl vyvinut pro odhad pohybu, navrhli Wang a Brady. Aplikovali diferenciální geometrické operátory při detekci rohů, která je založena na měření křivosti povrchu. Detekce je však odvozena ve zjednodušené formě vhodné pro časově kritické aplikace.

2.6.2 Topologické metody

Beaudet vyvinul v roce 1978 jeden z prvních rohových detektorů a svým přístupem ovlivnil mnoho dalších rohových detektorů. Beaudetův operátor je rotačně invariantní výpočet rohovitosti pomocí determinantu Hessianovy matice. Protože Hessiana matice zahrnuje výpočet druhých derivací, je tento operátor citlivý na šum. A navíc, ukázalo se, že není

stabilní při změnách velikosti. Na tento přístup lze nahlížet jako na hledání hran s vysokým zakřivením (například sedlové body v obrazové funkci) pomocí výpočtu míry Gaussovy křivosti obrazu (jako výsledek dvou základních zakřivení).

Deriche rozšířil Beaudetův operátor jeho použitím ve více měřítkách. Linie jsou taženy mezi korespondujícími rohy v každém měřítku a průnik těchto linií ve spojení s průchodem nulou Laplaciánu je označen za pozici rohu. Tímto způsobem se sice vylepší lokalizace rohového bodu, ale operátor dále zůstává citlivý na šum.

2.6.3 Autokorelační metody

Do této skupiny patří jedna z úplně prvních metod z roku 1977 pana Moravce, která inspirovala mnoho dalších dodnes používaných rohových detektorů. Metoda je založena na malém lokálním okně, které se pohybuje po obraze a vrací hodnoty rohovitosti z několika málo směrů posunutí. Tato operace je aplikována na každý pixel obrazu a každému je přidělena minimální hodnota ze všech hodnot posunutí v různých směrech. Body s relativně velkým ohodnocením jsou pak prohlášeny za rohy. Podrobně je tato metoda popsána v 3.1. Moravec implementoval tento přístup výpočtem nenormalizované lokální autokorelace ve čtyřech základních směrech. Výsledkem je však anizotropní odezva. Tento operátor se ukázal být citlivý na šum podél hran, protože i minimální změna intenzity podél hrany stačí k tomu, aby byl bod na hraně mylně označen za roh.

Špatné vlastnosti Moravcova operátoru se později mnozí snažili odstranit. O trochu lépe na tom je Harris/Plesseyův operátor, který odhaduje lokální míru autokorelace pomocí derivací prvního řádu, která je analytickým rozšířením Moravcova operátoru. Přínosem tohoto operátoru je jeho izotropní odezva (nezávislá na směru), více naleznete v 3.2. Tento detektor je považován za velmi dobrý při detekci pravých rohů. Děje se tak na úkor výpočetní náročnosti a ani jeho lokalizační přesnost není dobrá.

Zheng a Wang reagovali na výpočetní náročnost Harris/Plesseyova operátoru a navrhli výpočetně zjednodušené hodnocení rohovitosti tak, že analyzovali Harris/Plesseyův operátor a využili z něho klíčové aspekty zodpovědné za detekci rohů. Jak předpokládali, jejich rohový detektor mírně ztratil na kvalitě detekce, zato redukoval složitost výpočtu a přidal některá vylepšení při lokalizaci rohových bodů.

Förstnerův operátor používá podobné měření rohovitosti jako Harris/Plesseyův operátor (ačkoliv se jeho výpočet podstatně liší) a využívá lokálních statistik k výpočtu vybraného prahu. Výsledkem je operátor s lepší lokalizací za cenu vyšších výpočetních nároků. V praxi se Förstnerův operátor často používá a je snadno rozšiřitelný pro detekci středu u oblých rohů.

2.6.4 Alternativní metody

Existují další alternativní metody, které jsou často založeny na heuristickém přístupu k problému. Důležité jsou především proto, protože jsou schopny za určitých podmínek

dávat velmi dobré výsledky. Z tohoto důvodu bude podrobně popsána metoda CSS (*Curvature Scale Space*), zmíníme se o metodě SUSAN (*Smallest Univalued Segment Assimilating Nucleus*), kterou vyvinuli Smith a Brady a také o Trajkovic a Hedleyově způsobu detekce.

2.7 Algoritmus detekce rohů

Několik algoritmů detekce rohových bodů bude popsáno v další kapitole. Byly vybrány právě tyto, protože mají buď historický význam, jsou široce používané, jsou vhodné pro real-time aplikace a nebo jsou robustní. Některé tyto detektory mohou být použity i jako detektory význačných bodů, protože přiřazují každému pixelu v obraze nějakou hodnotu rohovitosti, ačkoliv počítají tuto rohovitost velmi odlišným způsobem. Naopak jeden z detektorů hledá rohy zcela odlišným způsobem, protože sleduje kontury objektů a hledá lokální maxima celkového zakřivení.

Následující detekční algoritmy budou rozebrány v další kapitole:

- Moravcův detektor (1977)

Jedná se o první způsob detekce význačných bodů, proto má velkou historickou hodnotu.

- Harris/Plessey (1988)

Zobecňuje Moravcův detektor matematickými postupy, široce používaný.

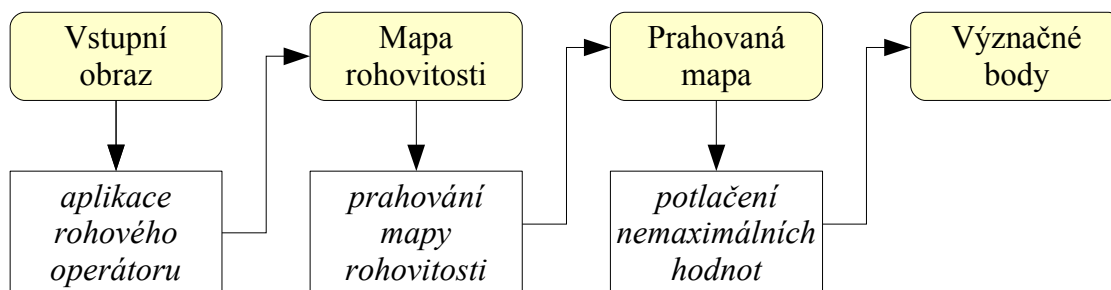
- Smallest Univalued Segment Assimilating Nucleus (1997)

Má zcela odlišný přístup k detekci, velmi rychlý algoritmus.

- Curvature Scale Space (1998)

Používá opět odlišný způsob detekce, je však pomalejší, zato mnohem robustnější a dosahuje nejlepších výsledků.

První dva zmiňované používají ke hledání rohů základní kroky, které si nyní popíšeme obecným způsobem. Obrázek 2.8 zobrazuje tyto základní kroky diagramem.



Obrázek 2.8: Diagram základních kroků algoritmu pro detekci význačných bodů.

2.7.1 Aplikace rohového operátoru

V tomto prvním kroku máme na vstupu původní obrazová data. Musíme určit několik parametrů, které rohový operátor požaduje. Na každý pixel vstupních obrazových dat se aplikuje rohový operátor, jehož výsledkem je hodnota míry rohovitosti (rohovitost). Tato hodnota představuje jakousi míru pravděpodobnosti, že daný pixel je rohovým bodem. Jednotlivé algoritmy se od sebe liší právě způsobem výpočtu této hodnoty, všechny ale počítají tuto hodnotu z hodnot pixelů v okolí zkoumaného pixelu. Výsledkem je ohodnocení každého pixelu obrazu tímto operátorem, tzv. *mapa rohovitosti*. Protože se operátor aplikuje na každý pixel obrazu, má výsledná mapa rohovitosti stejný rozměr, jako vstupní obraz.

2.7.2 Prahování mapy rohovitosti

Detektory rohových bodů definují rohy jako lokální maxima mapy rohovitosti. Bohužel ale mapa rohovitosti typicky obsahuje mnoho takových lokálních maxim, které mají relativně malou hodnotu a nejsou skutečnými rohy. Abychom potlačili takové nepravé rohy, jsme nuceni mapu rohovitosti prahovat. Všechny hodnoty v mapě rohovitosti pod hodnotou prahu jsou nastaveny na nulu. Problémem je, ostatně jako vždy, výběr prahové hodnoty, která bývá určena aplikací a děje se tak často empirickou metodou. Hodnota prahu musí být nastavena dostatečně vysoko, aby odstranila lokální maxima nepravých rohů, na druhou stranu musí být dostatečně nízká, aby ponechala lokální maxima rohů skutečných. V praxi takováto ideální hodnota prahu typicky neexistuje a je třeba se rozhodnout podle povahy aplikace, zda je výhodnější detekovat více rohů, i když nepravých, nebo naopak posunout hodnotu prahu výše, abychom měli větší pravděpodobnost detekce opravdových rohů za tu cenu, že jich spoustu ani nenalezneme.

2.7.3 Potlačení nemaximálních hodnot

Prahovaná mapa rohovitosti obsahuje lokální maxima, které potřebujeme nalézt, abychom je mohli označit za význačné body. Toho docílíme potlačením nemaximálních hodnot (angl. *non-maximum suppression*). Potlačení nemaximálních hodnot se používá jako mezikrok v mnoha algoritmech počítačového vidění. Pro každý bod v mapě rohovitosti zkoumáme jeho jisté okolí a pokud nalezneme v tomto okolí hodnotu větší, než je hodnota středového bodu, nastavíme hodnotu středového bodu na nulu. Po aplikaci potlačení nemaximálních hodnot jsou význačnými body jednoduše nenulové hodnoty, které v mapě rohovitosti zůstaly.

Kapitola 3

Rohové detektory

V této kapitole detailně popíšeme princip dvou detektorů. Začneme detektorem, který patří mezi historicky první a na kterém jsou vidět základní principy při hledání rohů. Tento detektor je poměrně jednoduchý, a proto se na něm ukáží i jistá úskalí a potíže. Ty se pokouší řešit další detektor, který na něj bezprostředně navazuje, ač je o jedenáct let mladší. Detektory v této kapitole předpokládají jako svůj vstup diskrétní 2D obraz.

3.1 Moravcův detektor

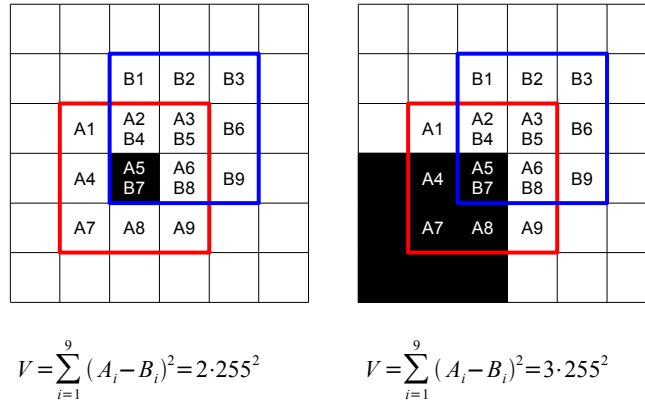
Tento operátor vyvinul Hans Peter Moravec v roce 1977 pro svůj výzkum zahrnující navigaci Stanford Cart ¹. Moravec definoval pojetí „bodů zájmu“ jako body, které uzavírají zřetelně odlišné regiony v obrazech a které by mohly být použity k nalezení sobě si rovných oblastí v po sobě jdoucích snímcích. Toto byl zásadní nízkoúrovňový krok ve zpracování obrazu, který mu dovolil určit existenci a umístění objektů v okolí (prostředí) vozíku.

Moravcův operátor je považovaný za rohový detektor od té doby, kdy definoval význačné body jako body, kde jsou velké změny jasu v každém směru a přesně to platí v případě rohů. Nicméně Moravec se přímo nezabýval hledáním rohů, ale odlišnými regiony v obraze, které se dají použít k hledání stejných oblastí v po sobě jdoucích snímcích. Mnozí doporučili tuto „definici“ toho, co vlastně roh je, protože do té doby nebyl pojem rohu pro šedotónové obrazy přesně stanovený.

Jaká je intenzita změny V naměřená v jednotlivých pozicích P v obraze? Moravec navrhol měření změny intenzity umístěním malého čtvercového okna (typicky, 3×3 , 5×5 , nebo 7×7 bodů) centrovaného v P a poté posouvání tohoto okna o jeden pixel v každém z osmi základních směrů (vodorovně, svisle a úhlopříčně). Intenzita změny pro daný posun je vypočtena jako součet čtverců z intenzit rozdílů odpovídajících si pixelů v těchto dvou oknech. Obrázek 3.1 ukazuje tento výpočet pro úhlopříčný posun na izolovaném černém pixelu (intenzita rovna 0) na bílém pozadí (intenzita rovna 255) a na ideálním rohu. Červený

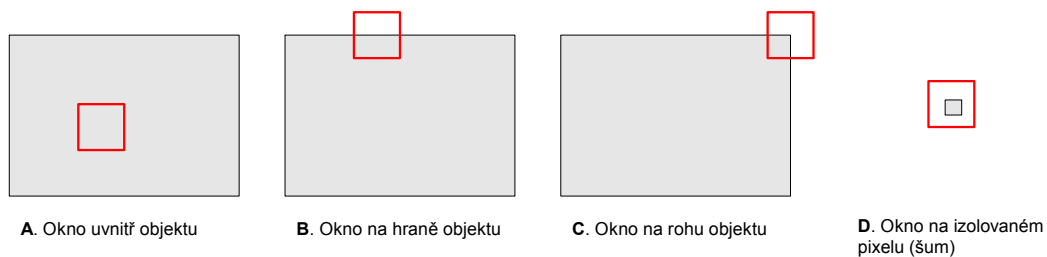
¹pojízdný „robotický“ vozík vyvíjený 45 let [3]

čtverec představuje původní okno a modrá barva ukazuje posun tohoto okna. Změna intenzity v P je pak *minimální* změnou intenzity, která byla vypočtena v osmi hlavních směrech.



Obrázek 3.1: Výpočet změny intenzity pro okno 3×3 v pravém horním diagonálním směru.

Abychom lépe pochopili, proč je Moravcův operátor rohovým detektorem, poslouží obrázek 3.2. Ten ukazuje čtvercová okna ve čtyřech různých typech pozic. Pozice A je uvnitř objektu (nebo na pozadí), kde se předpokládá, že intenzita obrazových bodů je v rámci okna relativně konstantní. Posun okna v kterémkoli směru vrací pouze nepatrnou změnu intenzity. Pro okno umístěné na hraně, jako je tomu v pozici B, platí, že jeho posouváním kolmo k hraně získáme vysoké změny jasu, ale jeho posouváním po hraně získáme naopak malé změny jasu. Pozice C a D, které odpovídají rohu a izolovanému pixelu, nám dávají velké změny jasu pro všechny směry posunu. Tímto jsme ukázali, že Moravcův operátor je skutečně rohovým detektorem, avšak s tím, že bude detekovat také izolované body jako rohy. Poznamenejme, že pokud máme velké změny jasu v každém směru, je to totéž, jako když řekneme, že minimální změna jasu ve všech směrech je velká hodnota.



Obrázek 3.2: Různé typy pozic okna.

Moravcův operátor může být použit pro změření „rohovitosti“ každého pixelu obrazu. Hodnotou je minimální změna jasu ve všech osmi směrech. Aplikací Moravcova detektoru na

každý pixel obrazu dostaneme *mapu rohovitosti*. Obrázek 3.3 ukazuje mapu rohovitosti pro jednoduchý obrázek při použití okna velikosti 3×3 . Tato mapa ilustruje několik důležitých bodů Moravcova operátoru²:

1. roh je lokálním maximem,
2. hodnota rohovitosti izolovaného pixelu je stejná jako hodnota rohovitosti rohu,
3. existuje oblast v blízkosti hranic obrazu, kde nemůže být Moravcův operátor přímo aplikován.

x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
x	x	0	0	0	0	0	0	0	0	0	1	1	1	x	x
x	x	0	0	0	0	0	1	1	0	0	1	2	1	x	x
x	x	0	0	0	0	0	2	1	0	0	1	1	1	x	x
x	x	0	0	0	0	0	0	0	0	0	0	0	0	x	x
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Obrázek 3.3: Mapa rohovitosti na jednoduchém malém obrazu.

První z bodů říká, že rohy jsou lokálními maximy mapy rohovitosti. Lokální maxima lze nalézt použitím potlačení odezev mimo maxima (angl. *non-maximal suppression*). Avšak tento postup označí izolované body jako rohy. Z tohoto důvodu je Moravcův rohový detektor citlivý na šum. Použití většího okna u tohoto algoritmu způsobí větší odolnost vůči šumu tak, že opravdový roh bude mít větší odezvu než izolovaný bod. Nicméně izolovaný bod zůstane stále lokálním maximem. Protože je každému pixelu obrazu přidělena hodnota rohovitosti, je snadné si představit, že v běžném šedotónovém snímku se objeví spousta lokálních maxim, které nekorespondují se skutečnými rohy. Tento problém lze částečně překonat nastavením všech hodnot rohovitosti pod určitým prahem na nulovou hodnotu. Určení takového prahu je obtížné, musí být nastaven dostatečně vysoko, aby se vyhnul falešným rohům a dostatečně nízko, aby udržel co nejvíce opravdových rohů. Body blízké okraji snímku se typicky ignorují, nebo se nastaví hodnota jejich rohovitosti na hodnotu 0. V lepším případě se upraví metoda, která zjišťuje hodnotu pixelu tak, aby hodnoty bodů mimo obraz byly stejné jako nejbližší hodnoty na okraji obrazu.

²Všechny hodnoty v mapě rohovitosti v obrázku 3.3 byly poděleny 255^2 , hodnota 2 pak ve skutečnosti reprezentuje hodnotu $2 \cdot 255^2$

Moravcův hranový detektor je poměrně jednoduchý algoritmus, který je v dnešní době většinou nevyhovující a je považovaný za zastaralý. Není rotačně invariantní, což je častý požadavek na moderní rohové detektory. Jeho velkou nevýhodou je vysoká citlivost na šum. Patří však mezi detektory, které jsou výpočetně efektivní.

Vzhledem k růstu výkonnosti počítačů za posledních několik desítek let používá dnes většina aplikací, i přes vyšší výpočetní nároky, metody s lepšími vlastnostmi. Jeden z nejrozšířenějších hranových detektorů, který se dnes používá, je Harris/Plesseyův detektor.

3.2 Harris a Stephens / Plesseyův detektor

Tento operátor vyvinuli Chris Harris a Mike Stephens v roce 1988 jako krok ve zpracování obrazu, aby napomohli výzkumům, které se pokoušely interpretovat robotovo prostředí. Harris a Stephens se zajímali o použití technik analýzy pohybu k interpretaci prostředí, které byly založené na snímání okolí z jedné mobilní kamery. Stejně jako Moravec potřebovali metodu, která by našla korespondující body v po sobě jdoucích snímcích. Na rozdíl od něj se však kromě sledování rohů mezi snímky zajímali i o hrany.

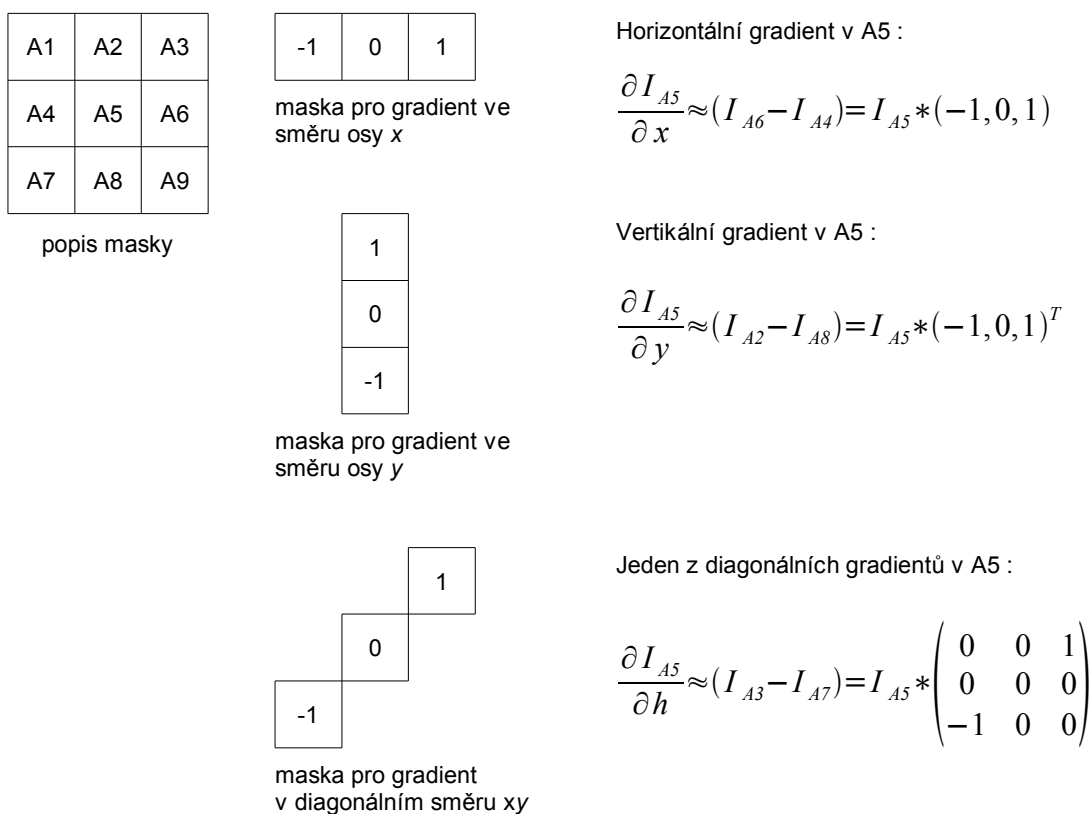
Harris a Stephens vyvinuli tento kombinovaný rohový a hranový detektor kvůli nedostatkům Moravcova operátoru. Výsledkem je mnohem vhodnější detektor ve smyslu lepší detekce a opakovatelnosti (stability) za cenu výrazně vyšších výpočetních nároků. I přes tyto vyšší požadavky je tento algoritmus v praxi široce používán.

Literatura se odkazuje na tento detektor jako na Harrisův rohový detektor i Plesseyův rohový detektor.

Plesseyův operátor se liší od Moravcova operátoru v tom, jakým způsobem je odhadována míra lokální autokorelace. Tato míra umožňuje získat odchylku autokorelace nad všemi různými orientacemi.

Moravcův operátor má anizotropní odezvu, která je způsobena výpočtem změn intenzit v okolí bodu pouze danou množinou směrů. Abychom odstranili toto omezení, potřebujeme funkci, která umožní vypočítat změnu intenzity v kterémkoli směru. Harris a Stephen vymysleli analytickou podobu rozložení Moravcova operátoru. Jedná se o derivaci, která není složitá, ale vyžaduje proniknutí do podstaty derivace Plesseyova operátoru. Zde si ukážeme mnohem intuitivnější přístup, který není tak matematicky rigorózní, ale dojdeme s ním k požadované funkci.

Prewittové operátor se obvykle používá k aproximaci gradientu obrazu. V mnoha aplikacích je gradient prvního řádu aproximován jednoduchými formulacemi, viz obrázek 3.4. S ohledem na tyto vzorce je vhodné navrhnout výpočet součtu rozdílů mezi korespondujícími pixely ve dvou oknech dostupnou aproximací gradientu. Jak je ukázáno na obrázku 3.5, změny jasu pro Moravcův operátor lze nyní aproximovat použitím gradientu obrazu.



Obrázek 3.4: Jednoduché diskrétní aproximace gradientu

Výše uvedená analýza je vyjádřením, že změna jasu může být zapsána jako funkce gradientu obrazu. Pro libovolný posun (u, v) můžeme vyjádřit změnu intenzity jako:

$$V_{u,v}(x, y) = \sum_{\forall i \text{ v okně centrovaném na } (x,y)} \left(u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2, \quad (3.1)$$

kde hodnoty $\frac{\partial I_i}{\partial x}$ a $\frac{\partial I_i}{\partial y}$ jsou vypočteny tak, jak je ukázáno na obrázku 3.5.

Pro posun ve vodorovném směru nebo ve svislém směru dostáváme přesnou aproximaci změny intenzity odvozenou v obrázku 3.5. Výsledky získané aplikací analytického odvození Moravcova operátoru jsou ukázány v rovnici 3.1.

Rovnice 3.1 není zcela přesně rovna změně intenzity, kterou bychom získali výpočtem pomocí Moravcova operátoru. Avšak díky tomu, že byla navrhnutá podle Moravcova operátoru, dává podobné výsledky změny intenzity. Výhodou tohoto nového přístupu je, že lze spočítat změnu intenzity v jakémkoli směru příslušným výběrem hodnot u a v . Kritici Plesseyova operátoru poukazují na to, že ačkoli umožňuje odhadovat změnu intenzity v každém směru, je stále anizotropní kvůli odhadům založeným na horizontálním a vertikálním gradientu. Navzdory této nežádoucí vlastnosti se Plesseyův rohový detektor široce používá.

A1	A2 B1	A3 B2	B3
A4	A5 B4	A6 B5	B6
A7	A8 B7	A9 B8	B9

Horizontální změna intenzity:

$$V_x = \sum_{i=1}^9 (A_i - B_i)^2 = \sum_{i=1}^9 (B_i - A_i)^2 \approx \sum_{i=1}^9 \left(\frac{\partial I_i}{\partial x} \right)^2,$$

$$\text{kde } \frac{\partial I_i}{\partial x} \equiv I_i * (-1, 0, 1) \approx B_i - A_i$$

B1	B2	B3
A1 B4	A2 B5	A3 B6
A4 B7	A5 B8	A6 B9
A7	A8	A9

Vertikální změna intenzity:

$$V_y = \sum_{i=1}^9 (A_i - B_i)^2 = \sum_{i=1}^9 (B_i - A_i)^2 \approx \sum_{i=1}^9 \left(\frac{\partial I_i}{\partial y} \right)^2,$$

$$\text{kde } \frac{\partial I_i}{\partial y} \equiv I_i * (-1, 0, 1)^T \approx B_i - A_i$$

	B1	B2	B3
A1	A2 B4	A3 B5	B6
A4	A5 B7	A6 B8	B9
A7	A8	A9	

Úhlopříčná změna intenzity:

$$V_h = \sum_{i=1}^9 (A_i - B_i)^2 = \sum_{i=1}^9 (B_i - A_i)^2 \approx \sum_{i=1}^9 \left(\frac{\partial I_i}{\partial h} \right)^2,$$

$$\text{kde } \frac{\partial I_i}{\partial h} \equiv I_i * \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} \approx B_i - A_i$$

Obrázek 3.5: Moravcův operátor aproximovaný použitím gradientu obrazu

Jak zvolit hodnoty u a v pro daný směr? Obrázek 3.6 ukazuje, že u a v jsou jednoduše x a y vzdálenosti potřebné k sestrojení čáry uvnitř trojúhelníku, která je v požadovaném směru. Není možné použít výpočet změn jasů ve všech směrech k tomu, abychom odhadli velikost lokální autokorelace (existuje nekonečně mnoho směrů). Tato zjevná potíž se projeví velkou odezvou na hrany.

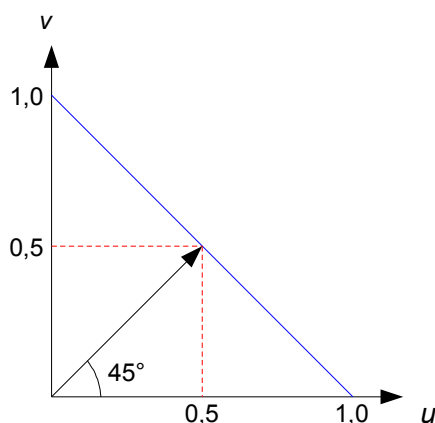
Použití čtvercového okna představuje v Euklidovském prostoru různé vzdálenosti od středu okna k jeho okraji různými směry. Toto lze jednoduše řešit použitím kruhového okna. Také není vhodné klást stejnou váhu pixelům bez ohledu na jejich vzdálenost od středu okna. Větší váhu by měly mít pixely ležící blíže středu. Oba tyto nedostatky Moravcova operátoru lze vyřešit použitím Gaussova okna. Velikost změny intenzity je nyní ukázána na obrázku 3.7 při horizontálním posunu a velikosti Gaussova okna 3×3 .

Změna intenzity může být vyjádřena jako:

$$V_{u,v}(x, y) = \sum_{\forall i \text{ v okně centrovaném na } (x,y)} w_i \left(u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2, \quad (3.2)$$

kde w_i je váha Gaussova okna na pozici i .

Je známo, jak snadno Moravcův operátor reaguje na hrany kvůli mnoha nedokonalostem na hranách způsobených šumem, pixelizací nebo kvantizací, díky kterým se objeví poměrně


 Obrázek 3.6: Volba hodnoty u a v pro libovolný směr

w1 0.04	w2 0.12	w3 0.04
w4 0.12	w5 0.36	w6 0.12
w7 0.04	w8 0.12	w9 0.04

Vážená horizontální změna intenzity:

$$V_x = \sum_{i=1}^9 w_i (A_i - B_i)^2 = \sum_{i=1}^9 w_i (B_i - A_i)^2 \approx \sum_{i=1}^9 w_i \left(\frac{\partial I_i}{\partial x} \right)^2,$$

$$\text{kde } \frac{\partial I_i}{\partial x} \equiv I_i * (-1, 0, 1) \approx B_i - A_i$$

A1	A2 B1	A3 B2	B3
A4	A5 B4	A6 B5	B6
A7	A8 B7	A9 B8	B9

Obrázek 3.7: Výpočet horizontálního posunu při použití Gaussova okna

veliké lokální minimum změn intenzit ve všech směrech. Plesseyův operátor předchází těmto nedostatkům přeformulováním míry rohovitosti tak, že je brán zřetel na změnu intenzity mezi hodnotami v různých směrech.

Rovnici 3.2 můžeme upravit následovně:

$$\begin{aligned} V_{u,v}(x,y) &= \sum_{\forall i \text{ v okně centrovaném na } (x,y)} w_i \left(u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2 \\ &= \sum_{\forall i \text{ v okně centrovaném na } (x,y)} w_i \left(u^2 \frac{\partial I_i^2}{\partial x} + 2uv \frac{\partial I_i}{\partial x} \frac{\partial I_i}{\partial y} + v^2 \frac{\partial I_i^2}{\partial y} \right) \\ &= Au^2 + 2Cuv + Bv^2, \end{aligned} \tag{3.3}$$

kde

$$A = \left(\frac{\partial I}{\partial x} \right)^2 \star w, \quad B = \left(\frac{\partial I}{\partial y} \right)^2 \star w, \quad C = \left(\frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \right) \star w.$$

Harris a Stephen si všimli, že je předchozí rovnici možné zapsat maticovým způsobem:

$$\begin{aligned} V_{u,v}(x, y) &= Au^2 + 2Cuv + Bv^2 \\ &= \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}, \end{aligned}$$

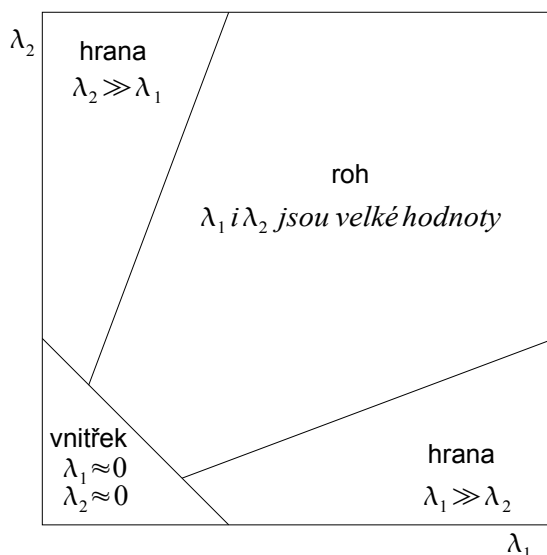
kde

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}.$$

Matice M obsahuje pouze diferenciální složky a popisuje geometrii povrchu obrazu v bodě (x, y) . Vlastní hodnoty matice M budou úměrné zakřivení povrchu obrazu.

Uvažujme opět čtyři různé typy pozic okna v obraze, jako tomu bylo na obrázku 3.2. Pozice A je uvnitř objektu (nebo na pozadí) a předpokládá se relativně konstantní jas uvnitř okna. Vlastní hodnoty matice M budou relativně malé. U oken umístěných na hraně (pozice B) je výrazné zakřivení kolmo k hraně a velmi malé podél hrany, proto jedna z vlastních hodnot matice bude velká a druhá malá. Pozice C a D, které korespondují s rohem a izolovaným pixelem, budou mít výrazné zakřivení v obou směrech a obě vlastní hodnoty budou velké.

Označme vlastní hodnoty matice M jako λ_1 a λ_2 . Rovinu popsanou hodnotami λ_1 a λ_2 můžeme rozdělit do tří odlišných regionů tak, jak je ukázáno na obrázku 3.8.



Obrázek 3.8: Rozdělení roviny vlastních čísel matice M do odlišných regionů.

Pro úplnost dodejme, jak vypočítáme vlastní hodnoty matice M :

$$\lambda_{1,2} = \frac{(A + B) \pm \sqrt{(A - B)^2 + 4C^2}}{2}.$$

Harris a Stephens navrhli následující výpočet rohovitosti:

$$\begin{aligned} C(x, y) &= \det(M) - k(\text{trace}(M))^2, \\ \det(M) &= \lambda_1 \lambda_2 = AB - C^2, \\ \text{trace}(M) &= \lambda_1 + \lambda_2 = A + B, \\ k &= \text{konstanta.} \end{aligned}$$

Empiricky bylo zjištěno, že nejlepších výsledků se dosahuje s $k \in \langle 0.04, 0.06 \rangle$.

Existují i jiné způsoby ohodnocení rohovitosti. Příkladem je Förstnerův operátor, který má mírně lepší lokalizaci za cenu větší výpočetní náročnosti. Výpočet hodnoty rohovitosti je následující:

$$C(x, y) = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} = \frac{\det(M)}{\text{trace}(M)}.$$

K odlišení rohů od bodů uvnitř objektů je zapotřebí stanovit práh. Body uvnitř objektů mají velmi malou hodnotu rohovitosti.

3.3 Trajkovicův detektor (4-okolí)

Tento operátor vyvinuli Miroslav Trajkovic a Mark Hedley v roce 1998 s úmyslem získat porovnatelný rohový detektor s dosavadními, co se stability týká, avšak s mnohem menší výpočetní náročností. Porovnali svůj operátor s jinými a zjistili, že má sice mírně nižší stabilitu, ale lokalizace je srovnatelná na L-rozích a na ostatních typech rohů je dokonce lepší. Empiricky ukázali, že je jejich detektor až pětkrát rychlejší než Plesseyův a nejméně třikrát rychlejší než ostatní typy detektorů.

Trajkovic a Hedley převzali stejnou definici rohu jakou použil Moravec: rohy jsou místa, kde dochází k prudké změně intenzity obrazu ve všech směrech. A stejně jako Moravec definovali rohovitost jako minimální změnu intenzity ve všech směrech. Výkonnost byla zvýšena použitím interpixelové aproximace k tomu, aby se odhadla změna intenzity ve všech směrech (jinak než u Moravce, který uvažuje pouze konečný počet směrů). Výkonnost je zvýšena použitím multigradního (vícerastrového) přístupu, což znamená, že se rohy hledají nejdříve v podvzorkovaném obraze.

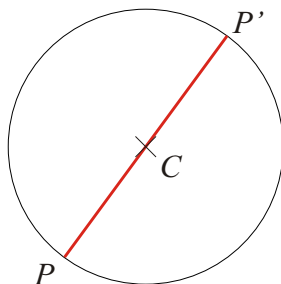
Minimální výpočetní nároky tohoto detektoru ho činí vhodným pro *real-time* aplikace. Analýzy ale ukázaly, že detektor není rotačně invariantní, je citlivý na šum a příliš lehce reaguje na diagonální hrany.

3.3.1 Způsob ohodnocení rohovitosti

Trajkovicův detektor uvažuje malé okno kruhovitěho tvaru a všechny úsečky, které prochází středem okna a jsou zakončeny na jeho hranici. Označme střed kružnice písmenem S . Nyní

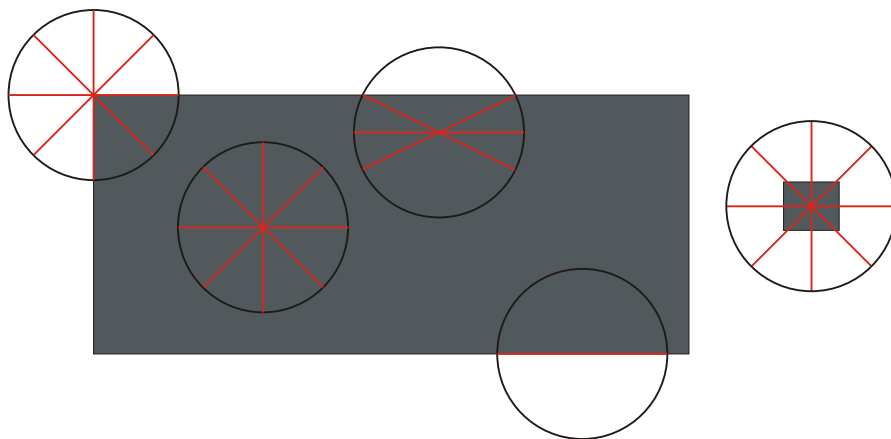
proložíme libovolnou přímkou středem kružnice S , která ji protne v bodě P a P' . Intenzitu v libovolném bodě označíme písmenem I . Potom je hodnota rohovitosti Trajkovicova detektoru dána vztahem

$$C(x, y) = \min ((I_P - I_S)^2 + (I_{P'} - I_S)^2), \forall P, P'.$$



Obrázek 3.9: Značení pro Trajkovicův detektor.

Hodnocení funkce rohovitosti bude dobře vidět na odlišných oblastech, které nyní popíšeme a které naznačuje obr. 3.10:



Obrázek 3.10: Princip Trajkovicova detektoru na rohu, na hraně, uvnitř objektu a na izolovaném pixelu.

Vnitřní oblast objektu - v případě, že větší část kruhového okna leží uvnitř téměř homogenního objektu, existuje nejméně jedna úsečka, na které je intenzita středu I_S přibližně rovna intenzitám I_P a $I_{P'}$. Proto je hodnota rohovitosti nízká a výpočet odolný vůči šumu.

Hrana - v případě, že se střed okna nachází na hraně objektu, existuje pouze jedna úsečka, na které je intenzita středu I_S přibližně rovna intenzitám I_P a $I_{P'}$. Výpočet rohovitosti je v tomto případě velmi citlivý na šum.

Roh - v případě, že je střed okna nad rohem objektu, potom na každé úsečce je alespoň jeden bod I_P nebo $I_{P'}$ mimo oblast vnitřku objektu, má tedy jinou hodnotu než I_S . Z toho důvodu je hodnota rohovitosti vysoká. Odolnost vůči šumu není v tomto případě příliš velká.

Izolovaný pixel - v případě, že je střed okna nad pixelem s výrazně odlišnou hodnotou od svého okolí, je na všech úsečkách hodnota v bodě I_P i $I_{P'}$ odlišná od hodnoty I_S , tudíž je hodnota rohovitosti vysoká. Izolovaný pixel představuje šum v obraze.

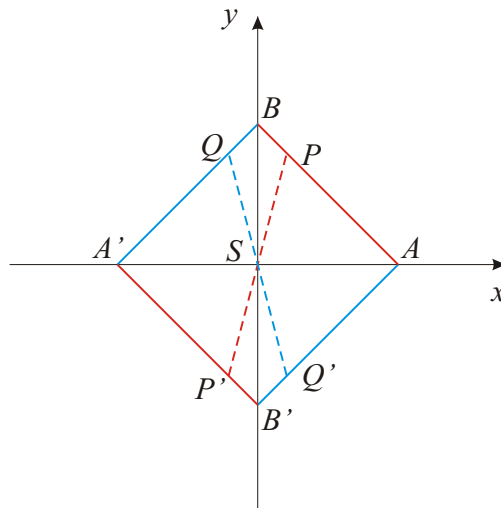
Z výše uvedené analýzy vyplývá, že použití Trajkovicova detektoru bude vhodné na relativně čisté, nezašumělé obrazy. Tomuto problému můžeme částečně předejít předzpracováním vstupního obrazu pomocí Gaussova rozmazání.

Abychom mohli provést výpočet ohodnocení rohovitosti, musíme nejdříve předchozí postup aproximovat. Pokud bychom brali v potaz pouze horizontální r_A a vertikální r_B změnu intenzity (jak je značeno na obr. 3.11), pak by byl výpočet dán jednoduchým vztahem

$$C_{SIMPLE}(x, y) = \min(r_A, r_B), \text{ kde}$$

$$r_A = (I_A - I_S)^2 + (I_{A'} - I_S)^2,$$

$$r_B = (I_B - I_S)^2 + (I_{B'} - I_S)^2.$$



Obrázek 3.11: Interpixelové pozice.

Abychom byli schopni zjistit změnu intenzity i na ostatních úsečkách vedených středem S , např. na PP' a QQ' (viz obr. 3.11) je zapotřebí ji vypočítat pomocí lineární interpixelové aproximace. Všechny úsečky vedené středem S můžeme určit zavedením parametru x , který dává pozici bodu P a Q (resp. P' a Q') na úsečkách AB a BA' (resp. $A'B'$ a $B'A$). Například

- když $x = 0$, pak $P = Q' = A$ a $P' = Q = A'$,
- když $x = 1$, pak $P = Q = B$ a $P' = Q' = B'$,
- když $x = 0,5$, pak P je v půli AB , P' je v půli $A'B'$, Q je v půli BA' a Q' je v půli $B'A$.

Ohodnocení rohovitosti můžeme nyní vyjádřit jako

$$C_{INTERPIXEL}(x, y) = \min_{x \in (0,1)} (r_1(x), r_2(x)), \text{ kde} \quad (3.4)$$

$$r_1(x) = (I_P - I_S)^2 + (I_{P'} - I_S)^2,$$

$$r_2(x) = (I_Q - I_S)^2 + (I_{Q'} - I_S)^2.$$

Stále nám ale chybí hodnota intenzit v bodech P, P', Q a Q' . Protože máme na vstupu diskrétní obraz, musíme hodnoty v těchto bodech aproximovat pomocí hodnot ostatních pixelů. Obrázek 3.12 ukazuje stejný případ jako obrázek 3.11 v diskrétní mřížce 3×3 pixelů. Je zřejmé, že hodnoty intenzit bodů A, B, A' a B' jsou známé. Intenzita libovolného bodu na úsečkách není známá, proto musí být aproximována použitím známých hodnot. V případě lineární aproximace jsou hodnoty intenzit v bodech P, P', Q a Q' rovny:

$$I_P = (1-x)I_A + xI_B, \quad (3.5)$$

$$I_{P'} = (1-x)I_{A'} + xI_{B'},$$

$$I_Q = (1-x)I_{A'} + xI_B,$$

$$I_{Q'} = (1-x)I_A + xI_{B'}.$$

Substituujeme-li hodnoty z rovnic 3.5 do vyjádření 3.4, můžeme vyjádřit hodnotu rohovitosti následovně:

$$C_{INTERPIXEL}(x, y) = \min_{x \in (0,1)} (r_1(x), r_2(x)), \text{ kde} \quad (3.6)$$

$$r_1(x) = A_1x^2 + 2B_1x + C,$$

$$r_2(x) = A_2x^2 + 2B_2x + C,$$

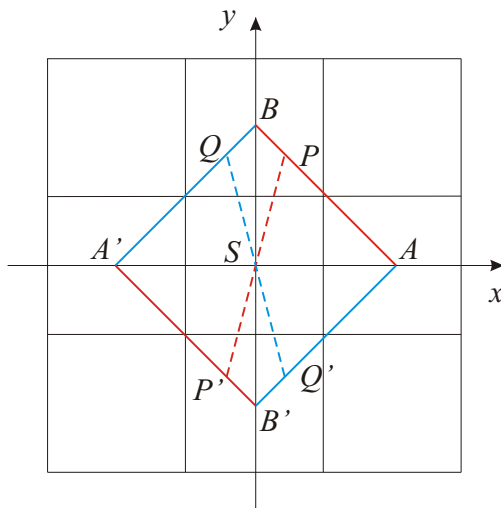
$$B_1 = (I_B - I_A)(I_A - I_S) + (I_{B'} - I_{A'})(I_{A'} - I_S),$$

$$B_2 = (I_B - I_{A'})(I_{A'} - I_S) + (I_{B'} - I_A)(I_A - I_S),$$

$$C = r_A,$$

$$A_1 = r_B - r_A - 2B_1,$$

$$A_2 = r_B - r_A - 2B_2.$$

Obrázek 3.12: Interpixelová aproximace pro okno 3×3 s použitím 4-okolí.

Hodnota rohovitosti je minimální změnou intenzity ve všech směrech, tedy

$$C_{INTERPIXEL}(x, y) = \begin{cases} C - \frac{B^2}{A} & \text{pokud } B < 0 \text{ a } (A + B) > 0 \\ C_{SIMPLE}(x, y) & \text{jinak,} \end{cases}$$

kde

$$\begin{aligned} B_1 &= (I_B - I_A)(I_A - I_S) + (I_{B'} - I_{A'})(I_{A'} - I_S), \\ B_2 &= (I_B - I_{A'})(I_{A'} - I_S) + (I_{B'} - I_A)(I_A - I_S), \\ C &= r_A, \\ B &= \min(B_1, B_2), \\ A &= r_B - r_A - 2B. \end{aligned}$$

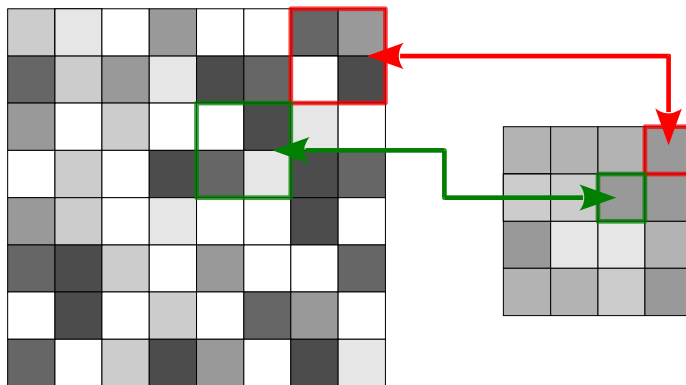
Demonstrovali jsme způsob výpočtu na okně 3×3 pixely, který byl založen na hodnotách 4-okolí středového pixelu. Použitím většího poloměru okna můžeme potlačit citlivost na šum, ale bude nás to stát horší lokalizací a hlavně vyšší výpočetní výkon.

3.3.2 Vícerastrový přístup

Původ rohových bodů můžeme rozdělit na dva zdroje – geometrické a texturové. Geometrické rohy náležejí hranicím objektů v obraze, kdežto texturové vznikají z textur objektů. Typicky nalezneme více rohů texturových než geometrických, protože objekty mají jen několik málo geometrických rohů a čím větší objekt, tím více texturových rohů na něm nalezneme.

Vícerastrový (angl. *multigrid*) algoritmus se pokouší redukovat počet texturových rohů, zatímco základní geometrické v obraze zachovává. Geometrické rohy jsou stabilnější než texturové, proto je vhodné ty texturové eliminovat.

Texturové rohy potlačíme tak, že zmenšíme rozlíšení pôvodného obrazu. Je vhodné použiť průměrování, např. obraz o velikosti 8×8 zmenšíme na čtvrtinu tak, že každou takovou čtveřici pixelů nahradíme jedním zástupným bodem, jehož hodnota odpovídá průměru původní čtveřice tak, jak ukazuje obrázek 3.13. Protože texturové rohy odpovídají většinou malým oblastem, budou mít tendenci díky takovému průměrování mizet.



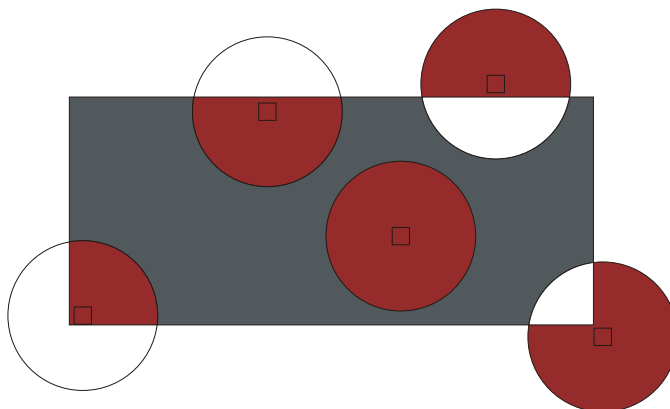
Obrázek 3.13: Zmenšení vstupního obrazu průměrováním hodnot intenzit pixelů.

Trajkovicův detektor rohů používá nižší rozlíšení vstupního obrazu, aby se tím redukoval počet rohů v texturách a aby se snížila výpočetní náročnosť. Rohy detekujeme pomocí Trajkovicova detektoru v obraze se zmenšeným rozlíšením. Potenciální rohové body se potom přenesou do pôvodného rozlíšení a podrobí se ďalšiemu zkoumání. Optimální měřítko zmenšeného obrazu závisí na konkrétnej aplikaci.

3.4 Smallest Univalue Segment Assimilating Nucleus

S novým přístupem k detekci rohů přišli v roce 1997 Smith a Brady. *Smallest Univalue Segment Assimilating Nucleus* (SUSAN) představoval zcela odlišný přístup k této problematice v porovnání s dosavadními postupy. Tento detektor nepoužívá žádné derivace (nepočítá se ani zakřivení ani se nehledají hrany). Způsob této detekce řadíme do skupiny alternativních detektorů rohů.

Tato metoda předpokládá, že uvnitř malé kruhové oblasti budou mít pixely daného objektu téměř stejný jas. Algoritmus zjistí počet pixelů s podobnou hodnotou jasu vzhledem k hodnotě středového pixelu masky (jádro masky). Takové pixely se pak nazývají *Univalue Segment Assimilating Nucleus* (USAN). Rohové body se detekují tak, že se tato maska aplikuje na každý bod v obraze a poté se hledají lokální minima. Obrázek 3.14 ilustruje nejčastější případy aplikace kruhové masky na jednoduchý objekt. Červená místa představují pixely s podobnou hodnotou jasu jako má střed masky.



Obrázek 3.14: Kruhové masky na různých částech obdélníka, červená barva představuje USAN oblasti.

Označme oblast pod kruhovou maskou jako M , potom tuto oblast reprezentují pixely $\vec{m} \in M$. Jádru masky je v \vec{m}_0 . Každý pixel porovnáme s jádrem použitím srovnávací funkce

$$c(\vec{m}) = e^{-\frac{(I(\vec{m}) - I(\vec{m}_0))^6}{t}},$$

kde t vymezuje šířku a mocnost exponentu byla určena empiricky. Oblast USAN je dána vztahem

$$n(M) = \sum_{\vec{m} \in M} c(\vec{m}).$$

Odezva USAN operátoru je dána vztahem

$$R(M) = \begin{cases} g - n(M) & \text{pokud } n(M) < g \\ 0 & \text{jinak,} \end{cases}$$

kde g je „geometrický práh“. Jinými slovy, USAN operátor má kladné hodnocení pouze, pokud je oblast dostatečně malá. Nejmenší USAN lze nalézt potlačením nemaximálních hodnot a tímto dostáváme celý SUSAN operátor.

Hodnota t říká, jak podobné musí být hodnoty bodů vzhledem ke středu masky, aby mohly patřit do segmentu bodů s podobnou intenzitou. Hodnota g určuje minimální velikost segmentu bodů s podobnou intenzitou. Pokud je g veliké, pak se operátor stává detektorem hran.

Při detekci rohů se používají další dva kroky. Nejdříve se nalezne těžiště oblasti USAN. U skutečného rohu nalezneme těžiště daleko od středu masky. Další krok spočívá v tom, že se všechny body na spojnici středu masky s těžištěm USAN oblasti musí nalézat uvnitř této USAN oblasti.

Detektor umožňuje hledat stejně dobře hrany jako rohy, je poměrně odolný vůči šumu a díky této vlastnosti není potřeba obraz nejprve vyhladit filtrem. Odolnost vůči šumu zvýšíme nastavením větší tolerance hodnoty t . Detektor je rychlý na výpočet, ale jeho hlavní nevýhodou je poměrně malá stabilita.

3.5 Curvature Scale Space

Tento detektor patří do skupiny alternativních metod detekce rohů. Jeho název je odvozen od metody, kterou při detekci rohů používá a která je známá pod zkratkou CSS. Technika CSS známá především z oblasti obnovy invariantních geometrických rysů rovinných křivek v různých měřítkách. Jednoduše řečeno jde o hledání objektů ve 2D prostoru, které jsou zabírány z různých pohledů a mohou být i částečně zakryté jinými objekty, jak je vidět na obrázku 3.15.



Obrázek 3.15: Trénovací obrázky pro rozpoznání dvou objektů jsou umístěny vlevo. Ty mohou být rozpoznány, i když je částečně překrývají jiné předměty. Výsledek rozpoznání je vpravo. Rovnoběžník obkresluje rozpoznané objekty po afinní transformaci, která byla vypočtena během rozpoznávání. Malé čtverečky indikují význačné body, které byly při rozpoznání použity.

Abychom mohli hledat význačné body s využitím této techniky, je nutné se seznámit s problematikou prostoru měřítek, se způsobem detekce hran a také s odhadem velikosti zakřivení křivek.

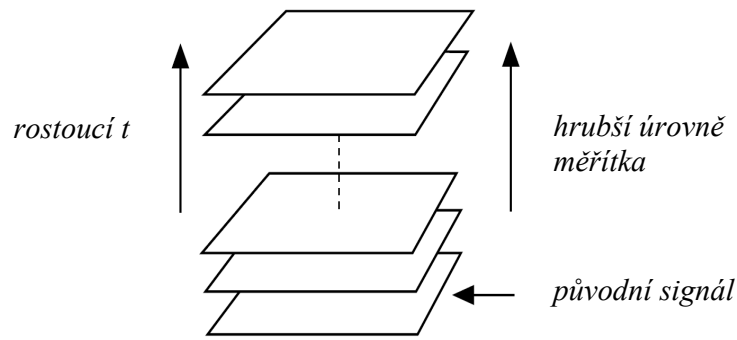
3.5.1 Reprezentace obrazových dat ve více měřítkách

Neodmyslitelná vlastnost skutečného světa kolem nás je ta, že věci dávají smysl v určitém rozsahu měřítek. Jednoduchým příkladem je představa větve stromu, která je srozumitelná pouze ze vzdálenosti, řekněme, několika centimetrů až po několik metrů. Nedává význam diskutovat koncept stromu ze vzdálenosti úrovně nanometrů či kilometrů. V takových měřítkách je mnohem relevantnější mluvit o molekulách listů stromu, resp. o lese, ve kterém strom roste. Tento fakt, že se stejné objekty v reálném světě jeví různými způsoby v závislosti na měřítku pozorování, je významným hlediskem, pokud je zamýšlíme popisovat. Tato základní potřeba byla od počátku respektována např. v kartografii, kde jsou mapy tvořeny v různých stupních abstrakce. Poznamenejme, že podoba popisu může být silně závislá na měřítkách, ve kterých je svět modelován a to je ve zřetelném kontrastu

k určitým idealizovaným matematickým konceptům, kterým je „bod“ nebo „čára“, které jsou nezávislé na měřítku pozorování.

Potřeba *multi-scale* reprezentace obrazových dat je vyloženě nutná, pokud navrhujeme metody pro automatickou analýzu a odvození informací ze skutečných měření reálného světa. Abychom byli schopni extrahovat nějakou informaci z obrazových dat, budeme na ně zřejmě muset aplikovat určité operátory. Typ informace, kterou můžeme získat, je silně omezena vztahem mezi velikostí aktuálních struktur v datech a velikostí (rozlišením) operátorů. Jeden ze základních problémů ve zpracování obrazu se týká toho, *jaké* operátory použít, *kde* je aplikovat a *jak velké* by měly být. Pokud nejsou tyto problémy vhodně řešeny, pak úloha interpretace odezvy operátoru může být velmi obtížná.

3.5.1.1 Základní myšlenky reprezentace dat v prostoru měřítek



Obrázek 3.16: Reprezentace signálu v několika měřítkách je seřazenou množinou odvozených signálů, které reprezentují původní signál v různých úrovních měřítka.

Scale-space teorie je základ pro prvotní obrazové operace, které byly vyvinuty komunitou pro počítačové vidění (jmenovitě Witkin [14], Koenderink [8], Yuille a Poggio [15], Lindeberg [10] a Florack [4]), aby se mohlo pracovat s výše zmíněnými vlastnostmi obrazových dat v prostoru měřítek. Hlavním argumentem tohoto přístupu je to, že pokud neexistuje žádná apriorní informace o tom, jaká jsou vhodná měřítka pro daná vstupní data, pak jedinou racionální cestou pro počítačové vidění je reprezentovat vstupní data v několika měřítkách. To znamená, že původní signál by měl být začleněn do jednoparametrových odvozených signálů, ve kterých jsou struktury v jemném měřítku postupně potlačovány, viz obr. 3.16. Jak by měla být tato myšlenka realizována v praxi? Rozhodujícím požadavkem je, že struktury v hrubých měřítkách by měly představovat zjednodušení korespondujících struktur v jemných měřítkách. Neměly by být náhodnými fenomény, které by byly vytvořeny metodou pro potlačení struktur v jemném měřítku. Tato myšlenka byla formalizována různými autory různými způsoby. Stojí však za povšimnutí, že tito autoři došli k podobným závěrům, i když každý začínal jinak. Koenderink a Lindeberg ukázali, že různé důvodné předpoklady pro to stát se jádrem konvoluce v prostoru měřítek, má pouze Gaussova funkce. Existuje

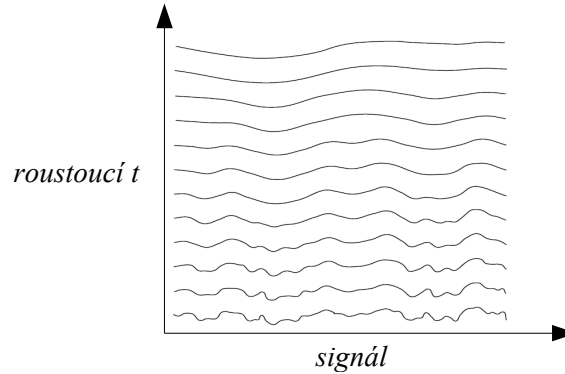
totiž několik požadavků (axiomů prostoru měřítek) [10], které musí taková funkce splňovat. V našem případě 2D prostoru je pro danou obrazovou funkci $f(x, y)$ odvozena sada signálů $L(x, y, t)$ definovaných jako konvoluce $f(x, y)$ s jádrem Gaussovy funkce:

$$L(x, y, t) = f(x, y) \star g(x, y, t),$$

kde

$$g(x, y, t) = \frac{1}{2\pi t} e^{-(x^2+y^2)/2t},$$

kde $t = \sigma^2$ je rozptyl Gaussovy funkce.



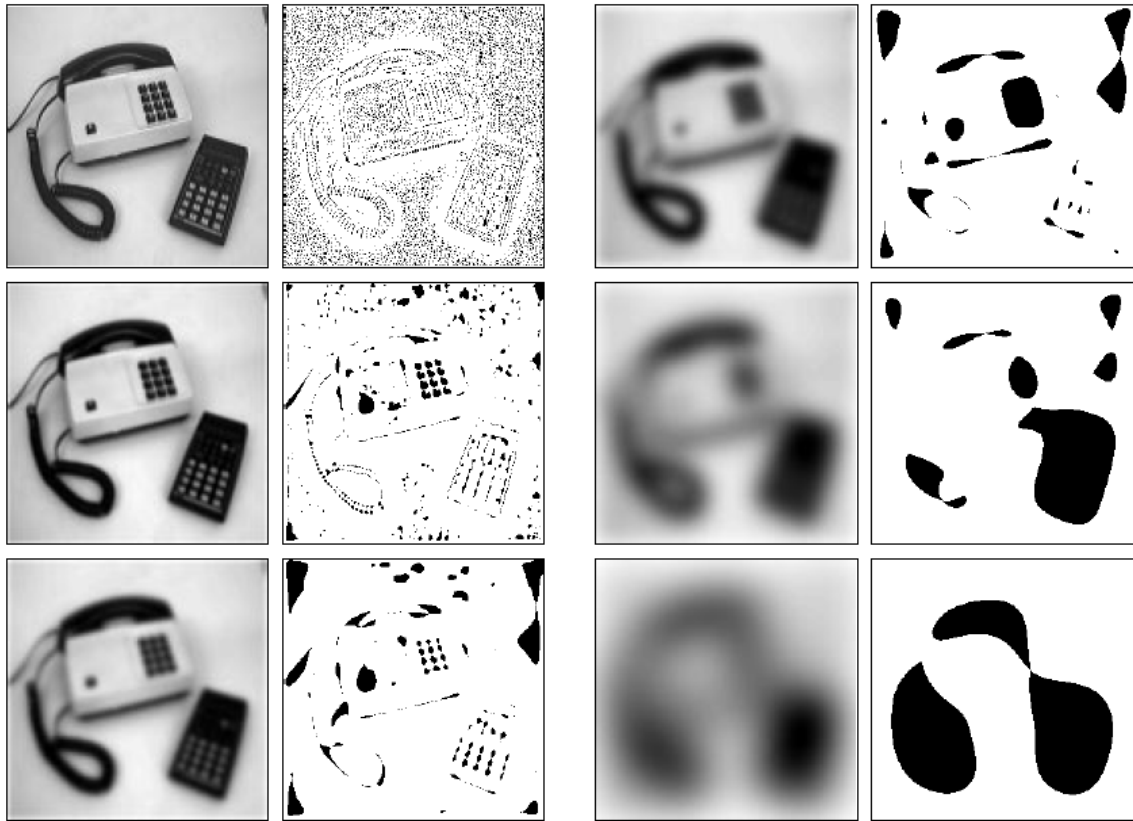
Obrázek 3.17: Hlavní myšlenkou reprezentace signálu v prostoru měřítek je vytvoření jedno-parametrových odvozených signálů, ve kterých je detailní informace postupně potlačována. Tento obrázek ukazuje jednorozměrný signál, který byl postupně vyhlazen konvolucí s jádrem Gaussovy funkce s rostoucím rozptylem.

3.5.2 Nalezení hran

Při hledání význačných bodů budeme potřebovat najít hrany v obraze, abychom mohli sledovat změny v jejich směru. Naším cílem bude hrany z obrazu extrahovat, proto musíme získat bitovou mapu tenkých hran. Pro tento účel můžeme použít *Cannyho hranový detektor*, jehož výstupem jsou tenké hrany, tedy hrany o tloušťce jednoho pixelu. Obecně však můžeme použít i jiný hranový detektor [7] a hrany následně ztenčit. Tato zaměnitelnost může být důležitou vlastností, neboť většinu času detekce rohových bodů metodou CSS stráví detektor právě detekováním hran.

Cannyho cílem bylo sestavit optimální algoritmus pro detekci hran. Pojmem „optimální“ se myslí:

- dobrá detekce,
detektor by měl označit co nejvíce skutečných hran v obraze,
- dobrá lokalizace,
nalezené hrany by měly být co nejbližší skutečným hranám v obraze,



Obrázek 3.18: Repräsentace dvojrozměrného obrazu v různých úrovních prostoru měřítek ($t = 0, 2, 8, 32, 128$ a 512) společně s vyznačenými lokálními minimy (černou barvou).

- minimální odezva,

hrana v obraze by měla být označena pouze jednou a kde je to možné, šum v obraze by neměl vytvářet falešné hrany.

Aby byly tyto požadavky uspokojeny, používá Cannyho hranový detektor několik kroků, které dávají přijatelný výsledek.

3.5.2.1 Redukce šumu

Protože Cannyho hranový detektor využívá při detekci hran filtr, který je založen na první derivaci Gausiánu[1], je citlivý na šum přítomný v obraze. V prvním kroku se nejdříve redukuje šum v obraze tím, že se obraz konvoluje s Gaussovou funkcí. Výsledkem je mírně rozostřená verze originálu. V diskrétní podobě se používá Gaussova maska, která je mnohem menší, než původní obrázek, např. 5×5 pixelů. Čím větší velikost Gaussovy masky (rozptyl Gaussovy funkce), tím menší citlivost na šum detektoru. Při příliš velkém rozptylu se ale

hrany začínají rozpadat. Diskrétní podoba Gaussovy funkce s rozptylem $\sigma = 1,4$ má tvar:

$$g_{5 \times 5} = \frac{1}{115} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}.$$

3.5.2.2 Gradient hran

Dalším krokem je získání gradientu hran. Gradient hrany v daném bodě představuje její velikost i směr. K aproximaci první derivace můžeme s výhodou použít Sobelův operátor. Tento operátor je založen na konvoluci a skládá se vždy z dvojice komplementárních konvolučních masek:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}.$$

Absolutní velikost gradientu $|G|$ je potom získána aplikací obou komplementárních konvolučních jader následujícím vztahem:

$$|G| = \sqrt{G_x^2 + G_y^2} \quad \text{nebo} \quad |G| = |G_x| + |G_y|.$$

Důležité je i získání směru gradientu. Protože se Sobelův operátor skládá ze dvou masek, z nichž každá je citlivá na jiný směr hran, je získání směru gradientu triviální:

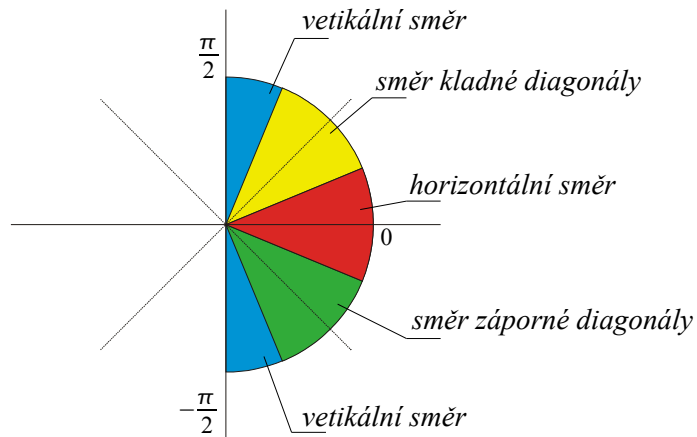
$$\theta = \arctan \frac{|G_y|}{|G_x|}.$$

Z vlastností funkce \arctan je zřejmé, že musíme ošetřit stav, kdy $|G_x| = 0$, pak $\theta = \pi/2$. Pokud je ale současně $|G_x| = 0$ a $|G_y| = 0$, potom úhel $\theta = 0$.

Směr gradientu použijeme v dalším kroku pro ztenčení hrany. Proto nám bude stačit pouze přibližná orientace hrany a rozdělíme si směry hran na pouhé čtyři směry: horizontální směr, vertikální směr, pozitivní a negativní diagonální směr. Získané úhly θ zařadíme do jedné z těchto oblastí tak, jak charakterizuje obrázek 3.19.

3.5.2.3 Ztenčení

Ztenčením hran máme na mysli potlačení nemaximálních hodnot. Hrany ztenčíme s využitím znalosti směru jejich gradientu. Sledujeme pixely podél hrany a potlačíme hodnoty takových



Obrázek 3.19: Čtyři možné směry hran.

pixelů, které nejsou maximy ve směru gradientu hrany. Výsledkem jsou hrany o tloušťce jednoho pixelu.

Algoritmus:

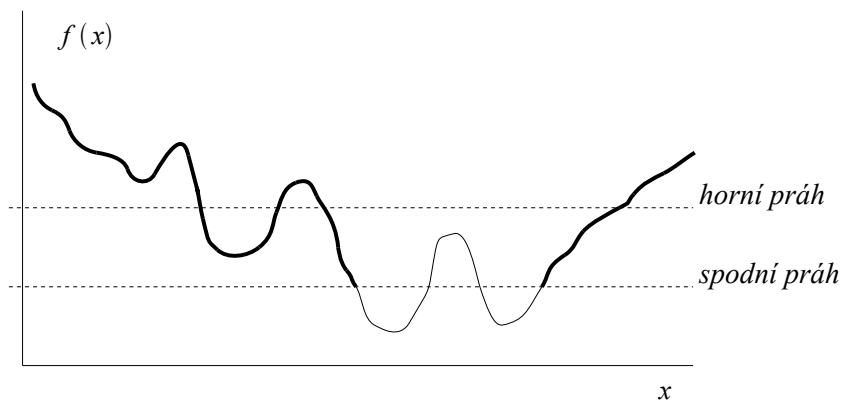
```

foreach pixel  $P$  v obraze do
  if pixel  $P$  má velikost  $|G| > 0$  then
    else if směr gradientu  $G \approx$  horizontální směr then
      if  $\max(|G|_{x-1}, |G|, |G|_{x+1}) \neq |G|$  then
        gradient v bodě  $P$ :  $G = \vec{0}$ 
      end
    end
    else if směr gradientu  $G \approx$  vertikální směr then
      if  $\max(|G|_{y-1}, |G|, |G|_{y+1}) \neq |G|$  then
        gradient v bodě  $P$ :  $G = \vec{0}$ 
      end
    end
    else if směr gradientu  $G \approx$  směr kladné diagonály then
      if  $\max(|G|_{x-1,y+1}, |G|, |G|_{x+1,y-1}) \neq |G|$  then
        gradient v bodě  $P$ :  $G = \vec{0}$ 
      end
    end
    else if směr gradientu  $G \approx$  směr záporné diagonály then
      if  $\max(|G|_{x+1,y+1}, |G|, |G|_{x-1,y-1}) \neq |G|$  then
        gradient v bodě  $P$ :  $G = \vec{0}$ 
      end
    end
  end
end

```

3.5.2.4 Prahování s hysterezí

Výsledek z předchozích kroků můžeme prahovat. Pokud bychom jej prahovali obyčejným prahem, dostali bychom hrany, které by pravděpodobně byly velmi přerušované, díky přítomnosti šumu v obraze. Abychom tento jev potlačili, použijeme prahování s hysterezí, které využívá prahy dva, spodní a horní. Každý pixel s hodnotou, která je větší než horní práh, zůstane zachován. Každý pixel s hodnotou, která leží mezi oběma prahy a současně tento pixel sousedí s pixelem, jehož hodnota je větší než horní práh nebo s pixelem, který již byl označen za hranu, zůstane zachován. A konečně, všechny pixely s hodnotou nižší, než je hodnota spodního prahu, budou potlačeny. Nejlépe tuto situaci demonstruje obrázek 3.20.



Obrázek 3.20: Prahování s hysterezí, příklad na jednorozměrné funkci.

3.5.3 Odhad celkového zakřivení

Existuje několik přístupů k tomu, jak odhadnout velikost zakřivení křivky. Často jsou tyto přístupy založené na heuristických metodách, ale existují i metody založené na matematických konceptech ve spojitém prostoru. Roh lze (neformálně) definovat jako místo s vysokým zakřivením oblouku nebo křivky.

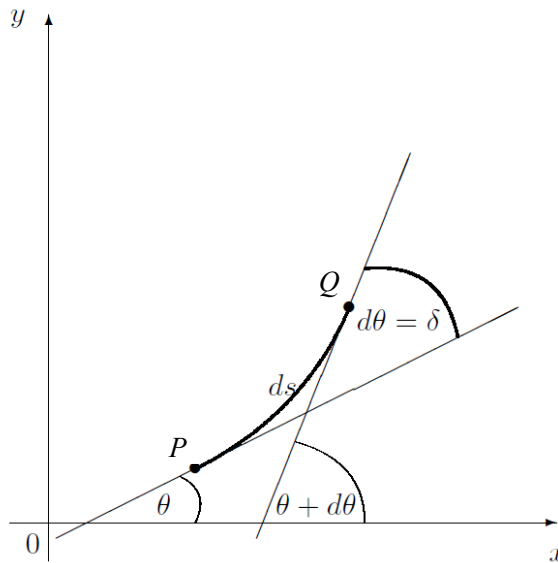
Uvažujme osm základních směrů jednoduché křivky ρ v diskrétní rovině \mathbb{Z}^2 . Pixely P_i tvořící takovou diskrétní křivku $\rho = P_0, P_1, \dots, P_{n-1}$ mají souřadnice (x_i, y_i) . Dolní indexy jsou násobkem čísla n , např. pixel P_{i-k} s $i - k < 0$ se shoduje s pixelem $P_{n-1+i-k}$.

Abychom detekovali roh v pixelu P_i na křivce ρ , je běžnou praxí, že rohový detektor provede měření, které je založeno na předchozím pixelu P_{i-k_b} , samotném P_i a na následném pixelu P_{i+k_f} , kde $k_b, k_f > 0$ jsou konstanty (např. obě rovny $k = 0,02 \cdot n$) nebo proměnné s definovaným intervalem hodnot. V bodě P_i se pak rozhoduje o tom, zda je rohovým bodem či nikoliv na základě úhlových měření s možnými hodnotami k_b a k_f .

Neadaptivní určení možných hodnot k_b a k_f neodráží tvar daný diskretní křivkou. Například pevně daná hodnota k ($k = 6$) vymezuje lokální přístup pro detekci rohu. Adaptivní určení k_b nebo k_f může být založeno například na algoritmu segmentace diskretních přímých úseků nebo na jeho aproximaci³, viz literatura [2][1][6]. Výhodou tohoto přístupu jsou unikátně definované k_b a k_f pro každý bod P_i a takové hodnoty odrážejí tvar křivky.

3.5.3.1 Derivace tangenty úhlu

Odhad hodnoty křivosti lze provést na základě změn v orientaci tangenty [6].



Obrázek 3.21: Odhad zakřivení založený na tangenti.

Nechť P a Q jsou body na rovinné křivce a δ je úhel mezi kladnými směry obou tangent v těchto bodech (viz obrázek 3.21). Zakřivení κ v bodě P je definováno jako

$$\kappa(P) = \lim_{|PQ| \rightarrow 0} \frac{\delta}{|PQ|}.$$

³*Digital straight segment* (DSS) je algoritmus, který dokáže rozdělit diskretní podobu křivky na sekvenci menších úsečků maximální délky. Příkladem takového algoritmu je algoritmus [2] s lineární časovou složitostí, který je založen na aritmetické geometrii.

Algoritmus:

```

foreach bod  $P_i$  na křivce  $\rho$  do
  vypočti  $k_b$  a  $k_f$  (pomocí algoritmu [13]);
   $l_b = |P_{i-k_b}P_i|$ ;
   $\theta_b = \tan^{-1} \left( \frac{|x_{i-k_b}-x_i|}{|y_{i-k_b}-y_i|} \right)$ ;
   $l_f = |P_{i+k_f}P_i|$ ;
   $\theta_f = \tan^{-1} \left( \frac{|x_{i+k_f}-x_i|}{|y_{i+k_f}-y_i|} \right)$ ;
   $\theta = \frac{1}{2}\theta_b + \frac{1}{2}\theta_f$ ;
   $\delta_b = |\theta_b - \theta|$ ;
   $\delta_f = |\theta_f - \theta|$ ;
  zakřivení  $\kappa(P_i) = \frac{\delta_b}{2l_b} + \frac{\delta_f}{2l_f}$ ;
end

```

Algoritmus vrací pouze kladné hodnoty, tudíž nelze získat informaci o konvexnosti či konkávnosti.

3.5.3.2 Poloměr tečné kružnice

Tečnou kružnici v bodě P na hladké křivce ρ můžeme definovat v diferenciální geometrii jako kružnici, která protíná křivku ρ v bodě P a další dva body P_b a P_f nalevo a napravo od P . Pohybem obou bodů směrem k P dostaneme tečnou kružnici v P se středem C (viz obrázek 3.22). Absolutní hodnota zakřivení v bodě P je potom definována jako převrácená hodnota poloměru $r = |CP|$.

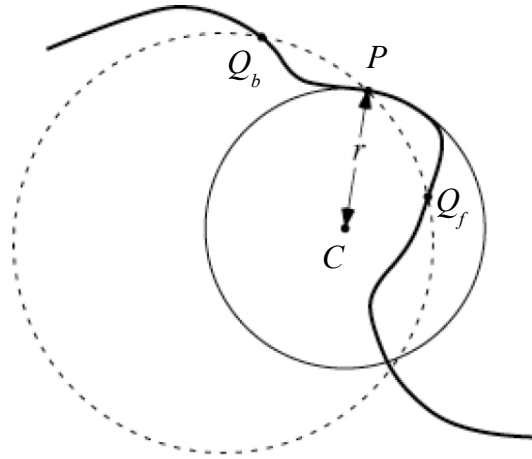
Následující výpočet tečné kružnice využívá geometrické vlastnosti, že tři různé body neležící v jedné přímce definují kružnici. V bodě P_i vypočteme dvě hodnoty DSS[2] stejným způsobem jako v případě předchozího algoritmu.

Algoritmus:

```

foreach bod  $P_i$  na křivce  $\rho$  do
  vypočti  $k_b$  a  $k_f$  (pomocí algoritmu [13]);
  vypočti půlící osy  $g_b$  a  $g_f$  segmentů  $P_{i-k_b}P_i$  a  $P_{i+k_f}P_i$ ;
  vypočti  $C$  jako průnik  $g_b$  a  $g_f$ ;
  vypočti poloměr  $r = |CP_i|$ ;
  zakřivení  $\kappa(P_i) = \frac{1}{r}$ ;
end

```



Obrázek 3.22: Čárkovaná kružnice náhodně umístěna tak, že prochází bodem P a protne křivku v dalších dvou bodech Q . Ty se pak posouvají do bodu P , až vytvoří tečnou kružnici se středem C .

3.5.3.3 Derivace křivky

Parametrizovaná křivka $\rho(t) = (x(t), y(t))$ umožňuje vypočítat velikost zakřivení na základě derivací. Zakřivení κ se vypočítá podle vzorce

$$\kappa = \frac{\begin{vmatrix} x' & y' \\ x'' & y'' \end{vmatrix}}{(x'^2 + y'^2)^{\frac{3}{2}}}. \quad (3.7)$$

Využívá se přitom polynom druhého stupně, který diskrétní křivku ρ lokálně aproximuje v bodě P_i s využitím pixelů P_{i-k_b} a P_{i+k_f} . Aproximující polynom je definován jako

$$\begin{aligned} x(t) &= a_2 t^2 + a_1 t + a_0, \\ y(t) &= b_2 t^2 + b_1 t + b_0, \end{aligned}$$

kde $t \in \langle -1; 1 \rangle$. Nechť $t = -1$ definuje P_{i-k_b} , $t = 0$ dává pixel P_i a konečně $t = 1$ definuje P_{i+k_f} . V tomto konkrétním případě dostaneme z rovnice (3.7) následující tvar

$$\kappa = \frac{2(a_1 b_2 - a_2 b_1)}{(a_1^2 + b_1^2)^{\frac{3}{2}}} \quad (3.8)$$

v bodě P_i .

Hodnoty a_1, a_2, b_1 a b_2 získáme z rovnic

$$\begin{aligned} a_2 - a_1 + a_0 &= x_{i-k_b}, \\ a_0 &= x_i, \\ a_2 + a_1 + a_0 &= x_{i+k_f} \end{aligned}$$

a analogickým způsobem dosadíme pro y . Nakonec si vyjádříme všechny proměnné pro náš případný algoritmus, které dosadíme do výpočtu křivosti podle rovnice (3.8):

$$\begin{aligned} a_1 &= \frac{1}{2}(x_{i+k_f} - x_{i-k_b}), \\ a_2 &= \frac{1}{2}(x_{i+k_f} + x_{i-k_b}) - x_i, \\ b_1 &= \frac{1}{2}(y_{i+k_f} - y_{i-k_b}), \\ b_2 &= \frac{1}{2}(y_{i+k_f} + y_{i-k_b}) - y_i. \end{aligned}$$

3.5.3.4 Heuristický přístup

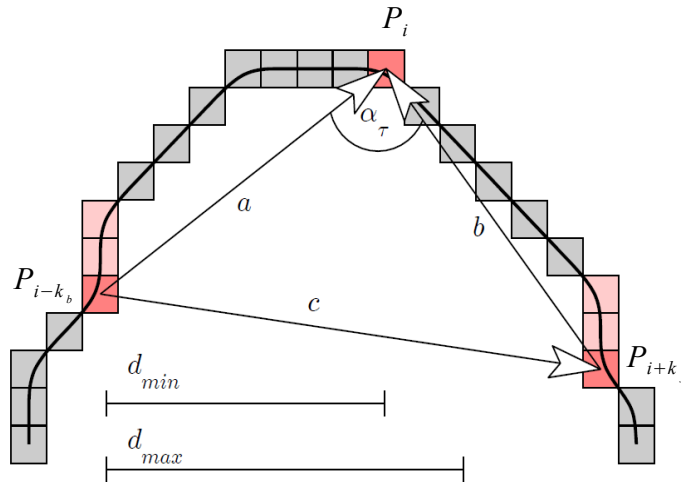
Uvažujme množinu T_i všech trojúhelníků $\triangle P_{i-k_b} P_i P_{i+k_f}$ s $k_b, k_f > 0$ takových, že

$$\begin{aligned} d_{min} &\leq |P_i P_{i+k_f}| \leq d_{max}, \\ d_{min} &\leq |P_i P_{i-k_b}| \leq d_{max}, \end{aligned}$$

kde d_{min} a d_{max} jsou pevně dané vzdálenosti, např. $d_{min} = 7$ a $d_{max} = 9$. Nechť má takový trojúhelník $\tau \in T_i$ velikosti stran rovny $a = |P_i P_{i-k_b}|$, $b = |P_i P_{i+k_f}|$, $c = |P_{i-k_b} P_{i+k_f}|$. Potom pomocí kosinové věty dostaneme

$$\alpha_\tau = \arccos\left(\frac{a^2 + b^2 - c^2}{2ab}\right),$$

což je úhel, pomocí kterého můžeme intuitivně odhadnout zakřivení v bodě P_i , jak je vidět na obrázku 3.23. Pixel P_i je potenciálním rohem, pokud $\min\{\alpha_\tau : \tau \in T_i\} < \alpha_{max}$, kde α_{max} je třetí pevně danou hodnotou, např. 150° .



Obrázek 3.23: Definování potenciálních rohů.

3.5.4 Způsob detekce rohů

Označme křivku symbolem ρ . Křivku nejdříve parametrizujme obloukovým parametrem t :

$$\rho(t) = (x(t), y(t)).$$

V prostoru měřítek má pak taková křivka tvar:

$$\rho_\sigma = (\mathcal{X}(t, \sigma), \mathcal{Y}(t, \sigma)),$$

kde

$$\mathcal{X}(t, \sigma) = x(t) \star g(t, \sigma), \quad \mathcal{Y}(t, \sigma) = y(t) \star g(t, \sigma),$$

kde symbol \star značí operátor konvoluce a $g(t, \sigma)$ značí Gaussovu funkci s rozptylem σ . Nyní je potřeba vypočítat zakřivení, např. pomocí derivace křivky (sekce 3.5.3.3):

$$\kappa(t, \sigma) = \frac{\left| \begin{array}{cc} \frac{\partial \mathcal{X}(t, \sigma)}{\partial t} & \frac{\partial \mathcal{Y}(t, \sigma)}{\partial t} \\ \frac{\partial^2 \mathcal{X}(t, \sigma)}{\partial t^2} & \frac{\partial^2 \mathcal{Y}(t, \sigma)}{\partial t^2} \end{array} \right|}{\left[\left(\frac{\partial \mathcal{X}(t, \sigma)}{\partial t} \right)^2 + \left(\frac{\partial \mathcal{Y}(t, \sigma)}{\partial t} \right)^2 \right]^{\frac{3}{2}}},$$

kde

$$\begin{aligned} \frac{\partial \mathcal{X}(t, \sigma)}{\partial t} &= x(t) \star \frac{\partial g(t, \sigma)}{\partial t}, & \frac{\partial^2 \mathcal{X}(t, \sigma)}{\partial t^2} &= x(t) \star \frac{\partial^2 g(t, \sigma)}{\partial t^2}, \\ \frac{\partial \mathcal{Y}(t, \sigma)}{\partial t} &= y(t) \star \frac{\partial g(t, \sigma)}{\partial t}, & \frac{\partial^2 \mathcal{Y}(t, \sigma)}{\partial t^2} &= y(t) \star \frac{\partial^2 g(t, \sigma)}{\partial t^2}. \end{aligned}$$

Rohy jsou pak (neformálně) definovány jako lokální maxima absolutní hodnoty křivosti. Ve velmi jemném měřítku existuje mnoho takových lokálních maxim díky šumu a digitální kontuře. Pokud se začne měřítko zvyšovat, začne se šum rozmazávat a zůstanou pouze maxima, které korespondují ke skutečným rohovým bodům. CSS detekce rohových bodů hledá rohy právě takovýmto způsobem.

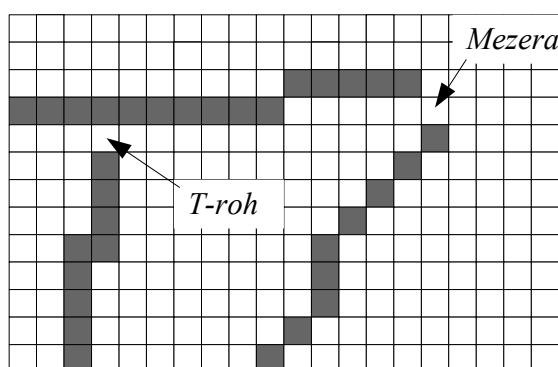
Proces detekce rohových bodů metodou CSS probíhá v následujících krocích:

1. užití Cannyho hranového detektoru na originální obrázek,
2. extrahování hranových kontur z obrazu hran,
 - zaplnění děr v konturách hran,
 - zaplnění T-uzlů a jejich označení za T-rohy,
3. výpočet křivosti v největším měřítku σ_{high} a určení kandidátních rohových bodů porovnáním maxim zakřivení vzhledem k prahu t a k sousedním lokálním minimům,
4. sledování nalezených rohů přes menší měřítka za účelem zpřesnění pozice těchto bodů,

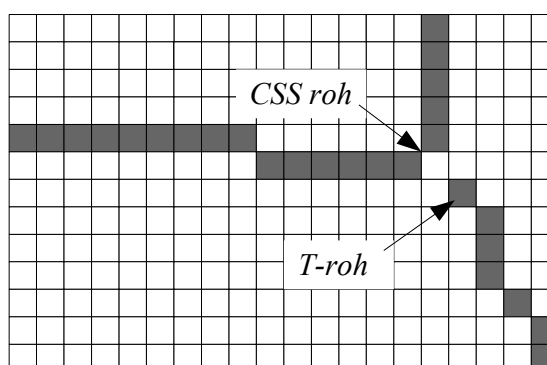
5. porovnání T-rohů s rohy, které byly nalezeny touto procedurou a odstranění těch, které jsou příliš blízko sebe.

Následuje podrobnější vysvětlení každé z těchto fází CSS rohového detektoru.

1. Zde byl použit Cannyho hranový detektor, může být ale nahrazen i jiným.
2. Cannyho detektor může způsobit mezery a T-spojky, potom by se nemuselo podařit takové rohy správně nalézt CSS metodou.
 - Pokud je koncový bod téměř spojen s jiným koncovým bodem, mezeru zaplníme a pokračujeme v extrakci.
 - Pokud je koncový bod téměř spojen s konturou hrany, ale ne s dalším koncovým bodem, označíme tento bod jako T-roh.



Obrázek 3.24: Dva případy mezery v kontuře hrany.



Obrázek 3.25: Příklad, kdy je označen jeden roh dvakrát.

3. Kontury hran jsou extrahovány z obrazu a v počátečním měřítku σ_{high} se počítá absolutní hodnota zakřivení. Lokální maxima velikostí těchto zakřivení jsou možnými kandidáty na rohové body. Lokální maximum je buď roh, nejvyšší hodnota zakulaceného rohu, nebo vrcholek způsobený šumem. Poslední dvě jmenované by neměly být označeny za rohy. Velikost zakřivení skutečného rohu má větší hodnotu než zaoblený roh nebo šum. Rohové body se porovnají se dvěma sousedními lokálními minimy. Velikost zakřivení by měla být alespoň dvojnásobná oproti jednomu ze sousedních lokálních minim. To je splněno tehdy, pokud je tvar obrysu velmi zakulacený a hodnota zakřivení přesáhne práh t . Práh t závisí na použitém měřítku σ_{high} a nastavuje se podle něho.
4. Poté co jsou nalezeny první rohové body, vstupuje do hry jejich sledování (angl. *tracking*). Lokalizace rohových bodů nebude nejpřesnější, protože tyto body jsme detekovali v měřítku σ_{high} . Velikost zakřivení se vypočte v menším měřítku a kolem kandidátních míst na rohy se prohledává malé okolí výsledků z předchozího měřítka. Pokud je to nutné, lokalizace rohů se aktualizuje novou pozicí. Sledování rohů pokračuje, dokud není měřítko velmi jemné. Tímto procesem dosáhneme velmi přesné lokalizace. Během sledování není třeba žádného prahování. Počet rohových bodů je určen v počátečním měřítku σ_{high} a během sledování dochází pouze ke změnám v umístění rohů, nikoli ke změně jejich počtu. Rohové body během sledování nemění svou pozici dramaticky. Výhodné je počítat velikosti zakřivení v jemnějších měřítkách pouze tam, kde je to třeba, tj. v malém okolí rohových bodů úvodního měřítka.
5. Jak již bylo popsáno výše, rohové body se určují dvěma metodami a v některých případech se může stát, že tentýž roh označí obě metody, jak je vidět na obrázku 3.25. Algoritmus extrakce hran zkoumá její malé okolí, když dorazí na její konec. CSS metoda nalezne roh v místě vysokého zakřivení souvislé hrany, zatímco algoritmus extrakce hran označí za T-roh konec hrany, která je téměř napojena na další souvislou hranu. Závěrečná část algoritmu spočívá ve zkoumání bodů, které byly označeny algoritmem pro extrakci hran. Označené T-rohy se porovnají s rohy nalezenými pomocí CSS metody a pokud se nachází příliš blízko sebe, jsou T-rohy odebrány.

Kapitola 4

Praktické zhodnocení detektorů

V této kapitole se budeme snažit zhodnotit výsledky čtyř implementovaných rohových detektorů na třech různých typech obrázků – k uměle vytvořenému obrázku 2.4 a „průmyslovému“ obrázku 2.6 přibude neupravená reálná scéna 4.1.



Obrázek 4.1: Ruční papírna ve Velkých Losinách.

Zpravidla není třeba detekovat význačné body ve fotografii s velkým rozlišením a získávat takové body ve velmi jemných strukturách. Detaily ve vysokém rozlišení bývají značně zatížené šumem či ztrátovou kompresí, a proto jsou z hlediska detektoru význačných bodů velmi nestabilní. Největší výhodou při detekci je tedy ta vlastnost, že můžeme obrazy vhodně podvzorkovat tak, aby zanikly nestabilní detaily textur, ale zároveň tak, aby zůstaly hlavní rysy objektů zachovány. Všechny obrázky podrobíme testům v rozlišení 256×256 bodů s 256 odstíny šedi. Z takto podstatné redukce obrazových dat vyplývá i výhoda podstatné redukce časové náročnosti detekce.

Nyní přistoupíme k testům jednotlivých detektorů. Detektory podrobíme několika typům testů z hlediska požadavků na detektory. Zaměříme se na

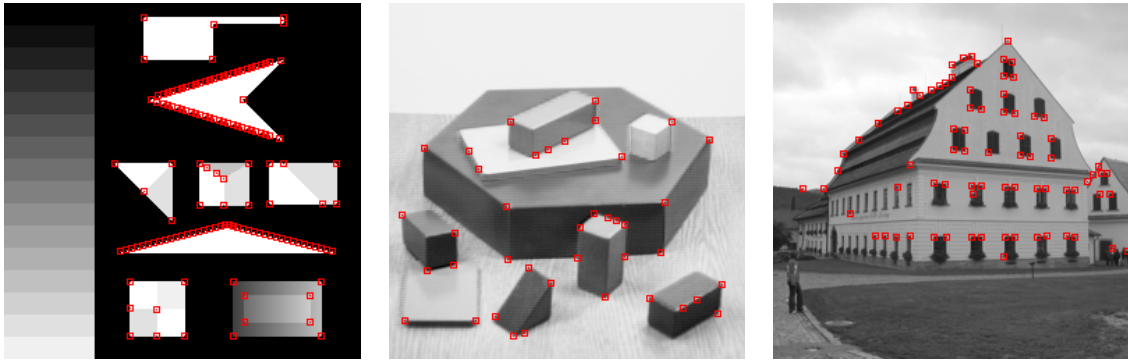
- stabilitu rohů (transformace rotace a zvětšení),

- odolnost vůči šumu,
- lokalizaci rohů a
- výpočetní efektivitu detektoru.

Otázkou detekce skutečného rohu se můžeme zabývat pouze v případě uměle vytvořené nebo velmi jednoduché scény, kde většina z nás intuitivně najde rohové body na správných místech.

Výsledek detekce závisí nejen na použitém detektoru, ale i na použitých parametrech. Detektory mají většinu parametrů individuálních a nelze je proto nastavit stejně. Společným parametrem je pouze nastavení úrovně výstupního prahu. V konečném důsledku však pomocí všech parametrů ovlivníme jedině – množství bodů, které označíme za význačné.

4.1 Moravcův detektor



Obrázek 4.2: Moravcův detektor – detekce na třech typech obrázků.

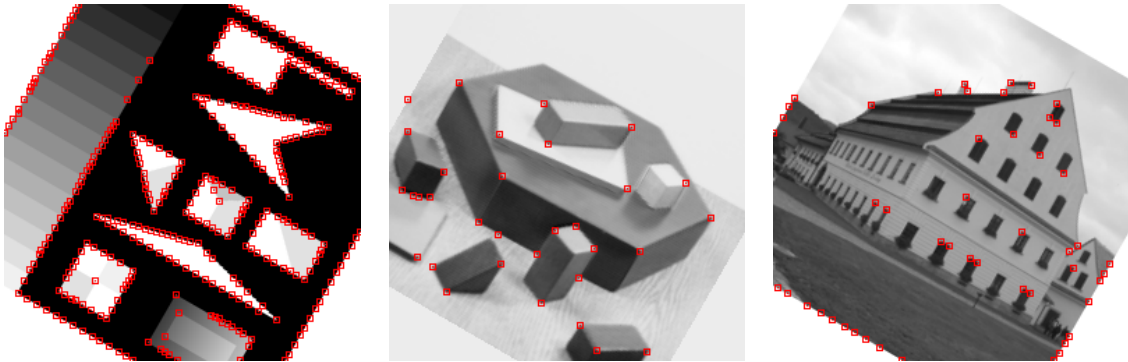
Obrázky 4.2 ukazují aplikaci Moravcova detektoru na třech typech obrázků. První používá okno o velikosti 5×5 bodů s prahem blízkým nule, ostatní mají práh vyšší a používají okno 3×3 .

Na prvním obrázku je Moravcův detektor schopen detekovat většinu rohů, je však zároveň citlivý na diagonální hrany. Tato citlivost je logickým důsledkem principu operátoru, který měří změny pouze v několika základních směrech, a proto na diagonální hraně změřív v těchto směrech poměrně velkou hodnotu.

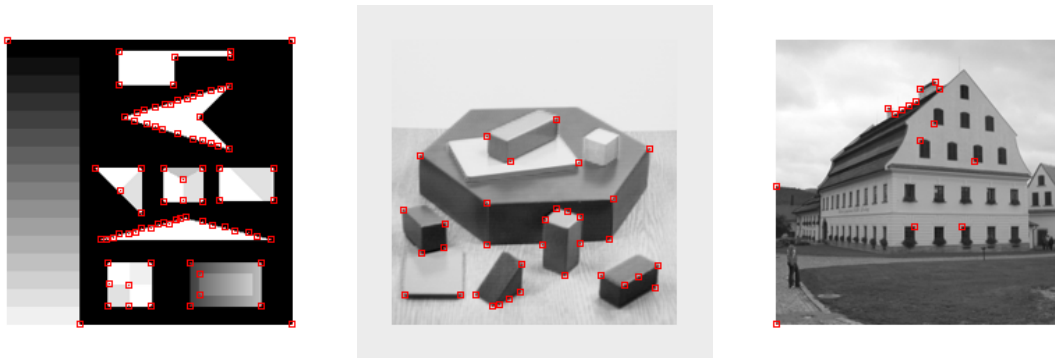
Na dalších dvou obrázcích je opět zřejmá detekce falešných rohů na diagonálách. Některé skutečné rohy mají hodnotu rohovitosti výrazně nižší, než na jakou byl nastaven práh. Většinou najdeme jen několik málo pravých rohů (jednotky), které mají hodnotu rohovitosti znatelně vyšší než rohy falešné.

V následujících testech byly všechny parametry detektoru u jednotlivých obrázků ponechány na stejných hodnotách, abychom mohli úspěšnost detektoru lépe pozorovat.

4.1.1 Nestabilita



Obrázek 4.3: Moravcův detektor – detekce na pootočených obrazech o 30° v záporném směru.



Obrázek 4.4: Moravcův detektor – detekce na zmenšených obrazech na 80% velikost.

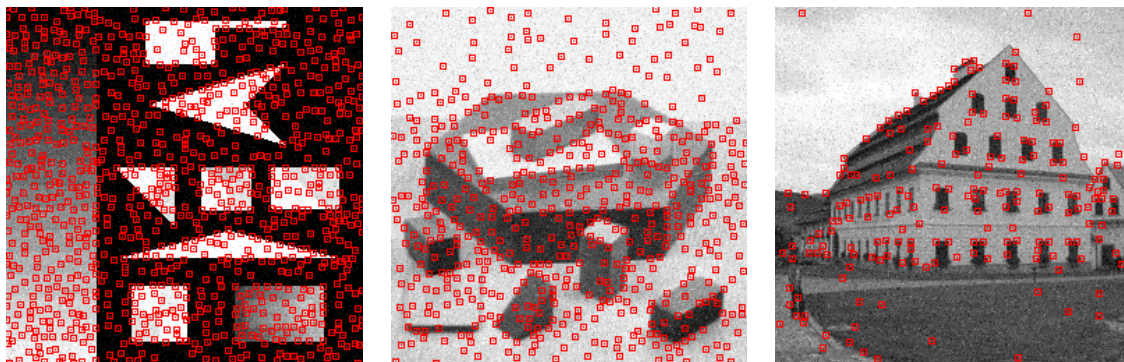
Velkým problémem Moravcova detektoru je jeho anizotropní odezva. Tato vlastnost souvisí s již avizovanou citlivostí detekce na diagonální hrany. Detektor tedy není rotačně invariantní, především proto má malou stabilitu.

4.1.2 Citlivost na šum

Moravcův detektor detekuje izolované pixely jako význačné body, což je vidět na obrázcích 4.5, které jsou zatíženy šumem s normálním rozložením.

4.1.3 Hodnocení

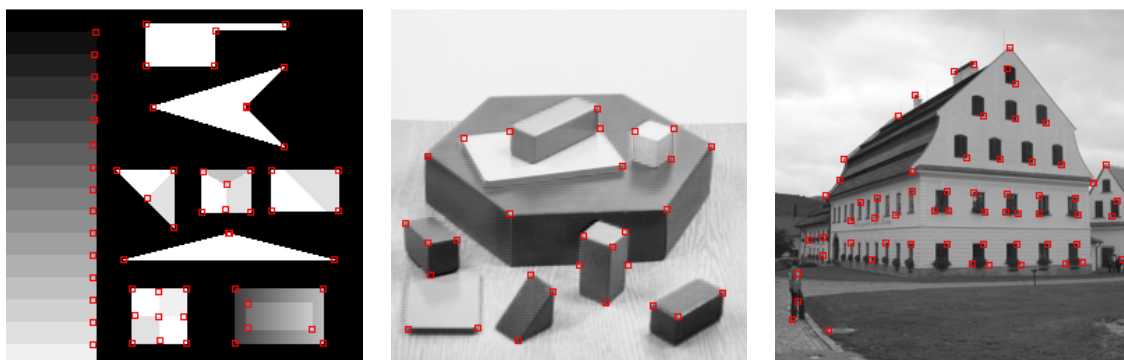
Moravcův detektor je relativně jednoduchý algoritmus, který má dva zásadní problémy – není rotačně invariantní (což se nedaří i jiným novějším detektorům) a je velmi citlivý na šum. Tyto vlastnosti brání jeho nasazení v aplikacích, kde je důležitá stabilita detekovaných bodů. Výpočetní náročnost je naopak velmi malá – na dnešních osobních počítačích trvá



Obrázek 4.5: Moravcův detektor – detekce na obrazech, které byly zatíženy Gaussovým šumem.

zpracování každého z předchozích obrázků do jedné sekundy. Díky rostoucím výpočetním výkonům je lepší tento detektor nahradit kvalitnějším algoritmem.

4.2 Harris a Stephens / Plesseyův detektor

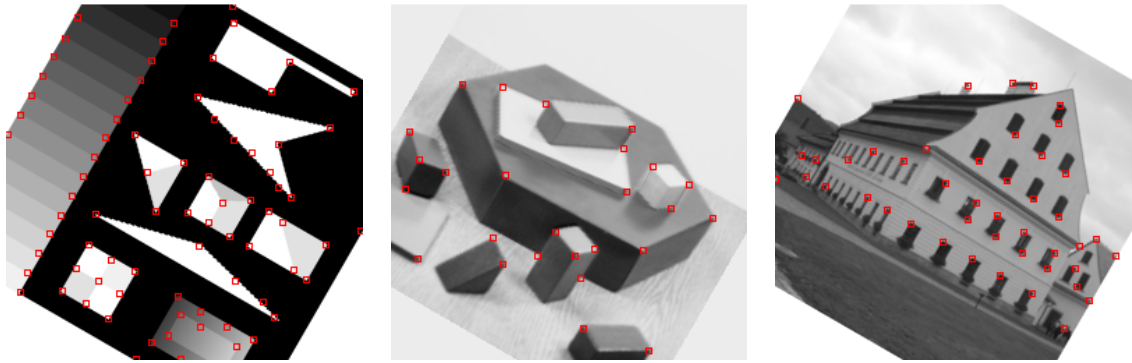


Obrázek 4.6: Plesseyův detektor – detekce na třech typech obrázků.

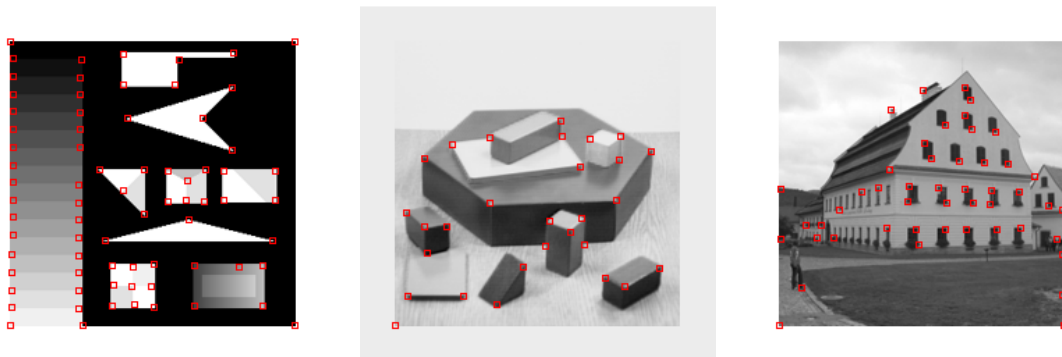
Obrázky 4.6 ukazují aplikaci Plesseyova detektoru opět na stejných třech typech obrázků. Na rozdíl od Moravcova operátoru již příliš netrpí špatnou detekcí na diagonálních hranách, nalezne mnohem méně falešných rohů, ale občas pravé rohy nepřilíží přesně lokalizuje. Na obrázku s bloky nalezne většinu pravých rohů, ostatní pravé rohy by se začaly objevovat společně s nepravými, pokud bychom práh snížili.

4.2.1 Stabilita

Stejně jako Moravcův detektor má i tento detektor anizotropní odezvu. Tento operátor má však rotačně invariantní autokorelační matici, a proto detektor bývá někdy nesprávně označován za rotačně invariantní. Ona autokorelační matice se však počítá pouze s využitím



Obrázek 4.7: Plesseyův detektor – detekce na potočených obrazech o 30° v záporném směru.



Obrázek 4.8: Plesseyův detektor – detekce na zmenšených obrazech na 80% velikost.

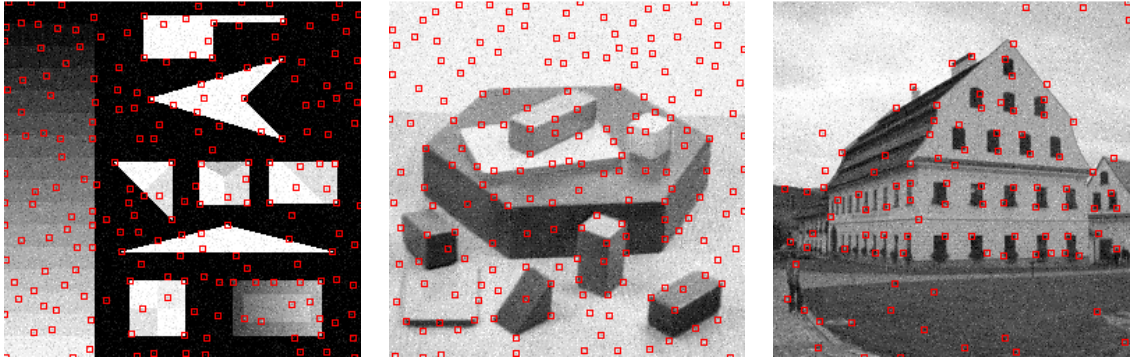
horizontálních a vertikálních gradientů. Anizotropní odezva je pozorovatelná na obrázcích 4.7.

4.2.2 Citlivost na šum

Každý algoritmus, který používá gradient nebo derivace vyššího řádu, je citlivý na šum. Izolované pixely mají vysoký gradient v každém směru. Jedinou možností, jak tento gradient částečně potlačit, je zvětšit Gaussovo okno a tím zahrnout do výpočtu více okolních pixelů. Tímto způsobem ale výrazně zvýšíme výpočetní nároky detektoru.

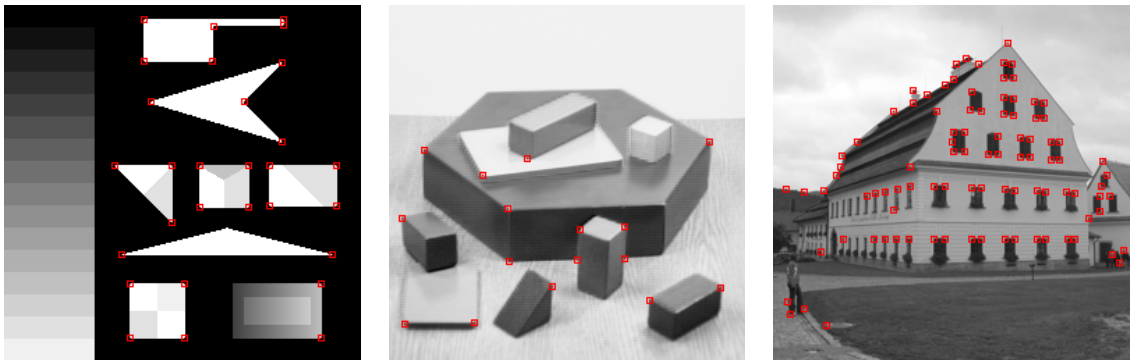
4.2.3 Hodnocení

Plesseyův detektor je poměrně jednoduchý detektor, který vychází z Moravcova detektoru a přináší mnohem větší stabilitu. Cenou za toto vylepšení je mnohem větší výpočetní náročnost. Další podstatnou nevýhodou je přetrvávající citlivost na šum a má i špatné lokalizace některých typů rohů. Existuje několik modifikací tohoto detektoru, které potlačují anizotropní odezvu a činí jej stále populárnější.



Obrázek 4.9: Plesseyův detektor – detekce na obrazech, které byly zatíženy Gaussovým šumem.

4.3 Smallest Univalue Segment Assimilating Nucleus



Obrázek 4.10: SUSAN – detekce na třech typech obrázků.

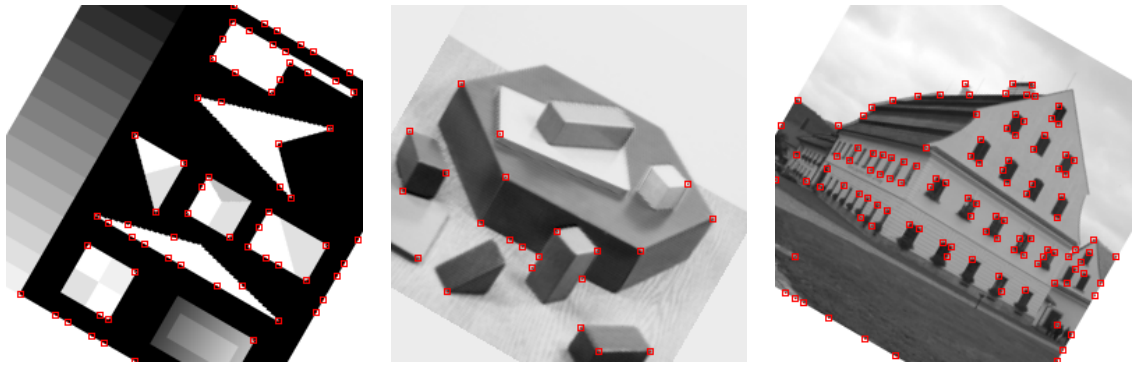
Obrázky 4.10 ukazují aplikaci SUSAN detektoru na známých obrázcích. Tento operátor, na rozdíl od předchozích, pracuje na zcela jiném principu, a tak je možné očekávat i jiné výsledky.

4.3.1 Nestabilita

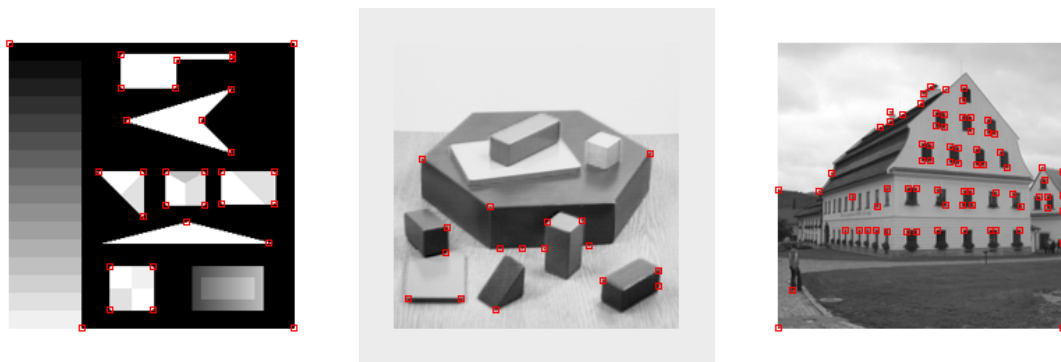
SUSAN detektor vykazuje nestabilitu především při otáčení obrazu. Při změně měřítka se chová lépe, především pak, pokud zvětšíme velikost kruhové masky. Tím se ovšem dostáváme k vyšším výpočetním nárokům.

4.3.2 Odolnost vůči šumu

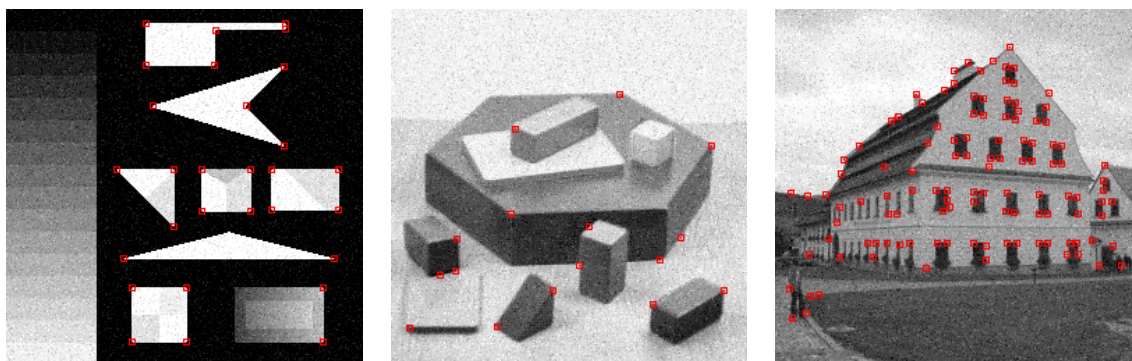
Tento detektor vykazuje velkou odolnost vůči šumu, což je vidět na obrázcích 4.13, které jsou zatíženy velkým šumem s normálním rozložením. To je dáno výpočtem USAN oblasti,



Obrázek 4.11: SUSAN – detekce na potočených obrazech o 30° v záporném směru.



Obrázek 4.12: SUSAN – detekce na zmenšených obrazech na 80% velikost.



Obrázek 4.13: SUSAN – detekce na obrazech, které byly zatíženy Gaussovým šumem.

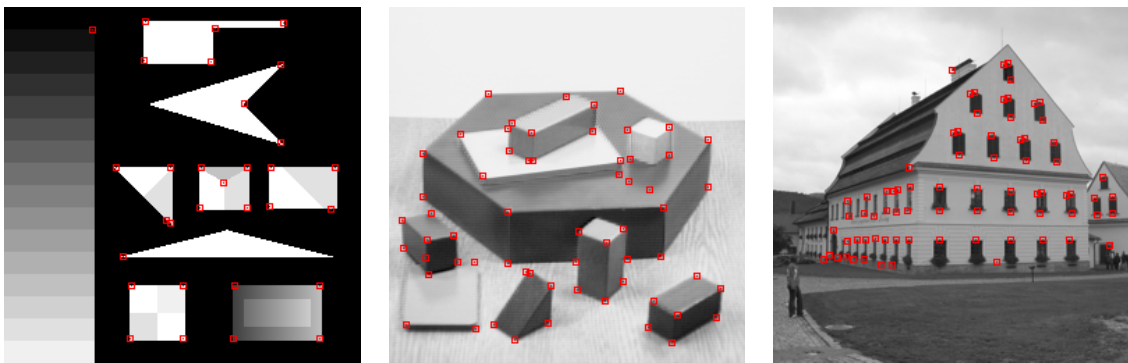
která zahrnuje pixely s podobnou intenzitou středu masky. Výpočet podobnosti obsahuje vážení hodnot pixelů funkcí, která je podobná Gaussově rozložení.

Nastavením parametrů algoritmu tak, aby byl odolný vůči šumu, přicházíme o možnost detekovat se stejnými parametry méně výrazné rohy v nezašuměném obraze.

4.3.3 Hodnocení

SUSAN detektor je poměrně rychlý algoritmus. Jeho nevýhodou je velmi malá stabilita, naopak největší výhodou je jeho odolnost vůči šumu. Lze říci, že se hodí pro jiný druh aplikací, než detektory testované výše, protože má opačné vlastnosti. Pokud bychom si mohli dovolit použít tento detektor společně s Plesseyovým detektorem v aplikaci, ve které by tolik nezáleželo na čase, získali bychom velmi robustní detekci význačných bodů.

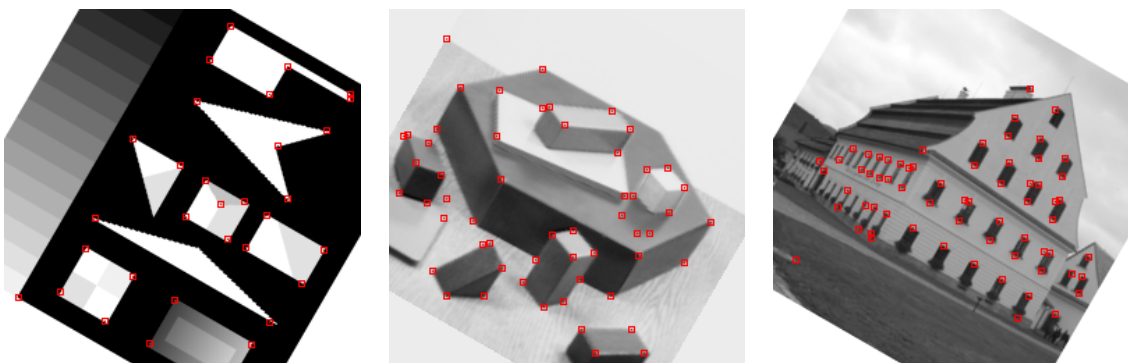
4.4 Curvature Scale Space



Obrázek 4.14: CSS – detekce na třech typech obrázků.

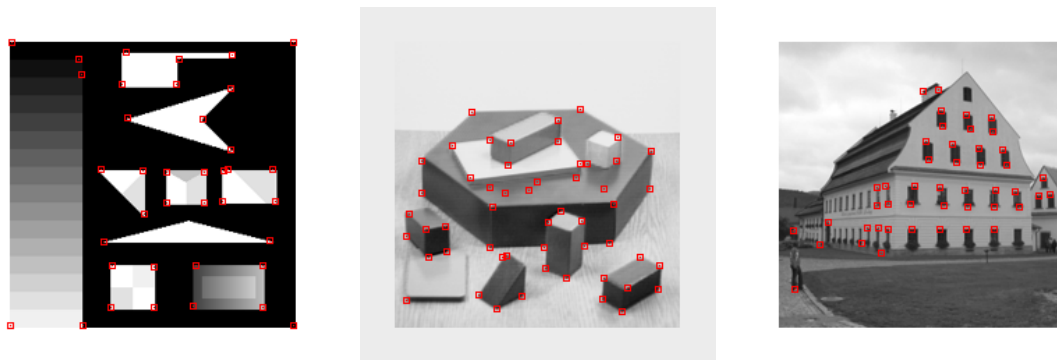
Posledním testovaným detektorem je CSS detektor. Jde opět o naprosto odlišný způsob detekce se zatím nesložitějším algoritmem.

4.4.1 Stabilita



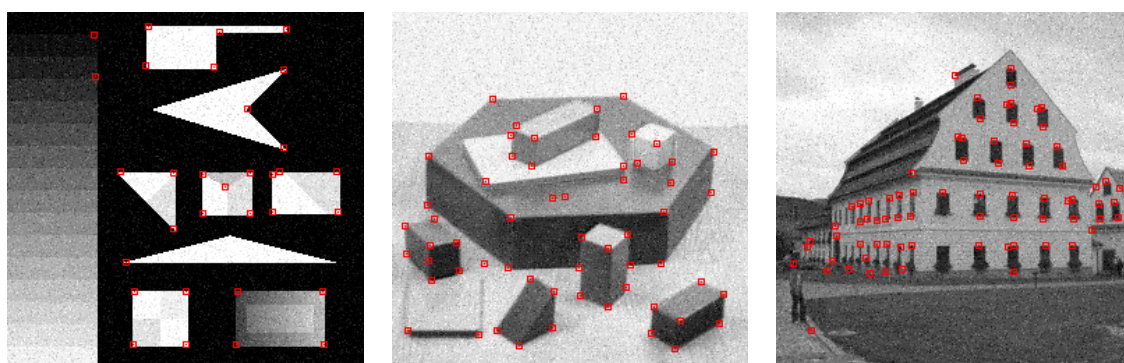
Obrázek 4.15: CSS – detekce na potočených obrazech o 30° v záporném směru.

Jak je patrné z obrázků 4.15 a 4.16, vykazuje tento detektor dobrou stabilitu v obou případech transformace rotace i zvětšení.



Obrázek 4.16: CSS – detekce na zmenšených obrazech na 80% velikost.

4.4.2 Odolnost vůči šumu



Obrázek 4.17: CSS – detekce na obrazech, které byly zatíženy Gaussovým šumem.

Na obrázcích 4.17 zatížených Gaussovým šumem zůstává většina rohů na stejných místech. Tato vlastnost vyplývá především z toho, že se používá Gaussův filtr před detekcí hran. Navíc se používá i hrubší měřítko při výpočtu zakřivení hran, čímž se citlivost na šum opět potlačí.

4.4.3 Hodnocení

Tento detektor v sobě skrývá robustní způsob detekce význačných bodů, protože nejvíce vyhovuje požadavkům na rohový detektor. Jeho jedinou nevýhodou je jeho výpočetní náročnost, která je dána paradoxně touto snadnější částí implementace – detektorem hran. Po extrakci hran je míra redukce pixelů, které se dále zpracovávají, významná a další kroky algoritmu nejsou tak časově náročné. Algoritmus se skládá z několika kroků, které lze implementovat různými způsoby a dokonce některé části můžeme i vynechat, pokud je nepožadujeme. Poznamenejme, že na experimentální obrázky nebyly aplikovány části, které se týkají hledání T-rohů a vylepšování lokalizace.

Kapitola 5

Závěr

5.1 Zhodnocení výsledků

Práce si kladla za cíl seznámit čtenáře s problematikou hledání význačných bodů a odvození vlastností jednotlivých způsobů detekce. Potřeba detekovat tyto body se objevila už před třiceti lety. Za tak dlouhé období prošla tato problematika dlouhým vývojem, který byl ale zpočátku silně brzděn tehdejším dostupným výpočetním výkonem. Teprve v poslední době zažívá tato oblast velký rozvoj. Práce přináší popis čtyř od sebe naprosto odlišných způsobů detekce význačných bodů.

Z popisu principu jednotlivých metod vždy vyšly najevo některé jejich nedostatky a úskalí. Ne vždy nám ale tyto nedostatky působí nepřekonatelnou překážku v praktickém použití. Ukazuje se, že použití metod pro vyhledání význačných bodů v obraze je často založeno na kompromisu mezi kvalitou a rychlostí. To potvrzují i novější a sofistikovanější způsoby detekce, které jsou více výpočetně náročné a přináší s sebou kvalitnější výsledky.

Na základě implementace vybraných algoritmů se podařilo ověřit a upřesnit vlastnosti jednotlivých metod, které velmi stručně shrnuje tabulka 5.1.

5.2 Možnosti dalšího vývoje

Problematika hledání význačných bodů, ač se zdá být na první pohled jednoduchou záležitostí, není vůbec jednoduchá. Je však velmi důležitou součástí v oblasti zpracování obrazu, zejména v aplikacích počítačového vidění. Této problematice se nelze vyhnout a je třeba ji řešit, protože se stále jedná o otevřený problém. Ukazuje se, že lze jistými postupy dosáhnout relativně uspokojivých výsledků. Aby byly výsledky relativně uspokojivé, bývají postupy vedoucí k jejich získání značně specifické a náročné – specifické z hlediska řešeného problému a náročné z hlediska implementace a výpočetního výkonu. Neexistuje žádná univerzální metoda, která by dávala obecně nejlepší výsledky.

V této práci jsou neřešenou otázkou možnosti automatických nastavení hodnot mnoha dílčích metod a jejich optimální výběr podle zvolených kritérií.

Tabulka 5.1: Porovnání vybraných detektorů, které byly podrobeny předchozím testům.

Operátor	Stabilita	Lokalizace	Odolnost vůči šumu	Výpočetní náročnost
Moravec	špatná	dobrá	velmi špatná	malá
Plessey	dobrá při vylepšeném izotropním výpočtu gradientů, jinak špatná	na L-rozích dobrá, v ostatních případech špatná	špatná	při vylepšeném výpočtu gradientů velká, jinak střední
SUSAN	špatná	dobrá (špatná v rozmazaných obrazech)	výborná	malá
CSS	výborná	dobrá	velmi dobrá	velká (závislá na použitém detektoru hran)

Studium co největšího publikovaného množství metod detekce význačných bodů nám umožní lépe poznat možnosti a omezení těchto detektorů. Pomocí získaných znalostí je možné lépe řešit případné specifické problémy z oblasti počítačového vidění. Vhodným pokračováním může být studium nových metod, které mívají často pracovní názvy typu „nový rohový detektor“ a které nejsou dosud dostatečně známé, i když mohou být kvalitní a mohly by přinést nový pohled na detekci význačných bodů v obraze.

Literatura

- [1] Coeurjolly, D. – Miguet, S. – Tougne, L.: *Discrete curvature based on osculation circle estimation*. In Proceedings of int. workshop visual form, LNCS 2059, s. 300–312, Springer, Berlin, 2001.
- [2] Debled-Rennesson, I. – Reveilles, J. P.: *A Linear Algorithm for Segmentation of Digital Curves*. In Proceedings of Pattern Recognition and Artificial Intelligence (PRAI), 9:635–662, 1995.
- [3] Earnest, L.: *Stanford Cart [online]*. 2005-08 [cit. 2007-05-21]. Dostupný z WWW: <http://www.stanford.edu/~learnest/cart.htm>.
- [4] Florack, L. M. J.: *The Syntactical Structure of Scalar Images*. PhD thesis, Dept. Med. Phys. Physics, Univ. Utrecht, NL-3508 Utrecht, Netherlands, 1993.
- [5] Hermann, S. – Klette, R.: *Global Curvature Estimation for Corner Detection*. In Proceedings of international conf. Image and vision computing (IVCNZ), Dunedin, New Zealand, s 272-277, 2005.
- [6] Hermann, S. – Klette, R.: *Multigrid analysis of curvature estimators*. In Proceedings of Image and vision computing, New Zealand, s. 108–112, 2003.
- [7] Kaněčka, P.: *Detekce hran v obraze*. 2005. 29 s. Fakulta informačních technologií Vysokého učení technického v Brně. Vedoucí bakalářské práce Ing. Petr Blatný.
- [8] Koenderink, J. J.: *The structure of images*. Biological Cybernetics, 50:363–370, 1984.
- [9] Kubečka, L.: *Optimization methods for image registration*. In Proceedings of 9th conference and competition STUDENT EEICT 2003, Volume 1. Student EEICT 2003. Brno: FEKT VUT Brno, 2003, s. 240–242. ISBN 80-214-2377-3.
- [10] Lindeberg, T.: *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, Netherlands, 1994, ISBN 0-7923-9418-6.
- [11] Parks, D. – Gravel, J.: *Corner detectors [online]*. Faculty of Engineering at McGill University, Canada, [cit. 2007-05-21]. Dostupný z WWW: <http://www.cim.mcgill.ca/~dparks/CornerDetector>
- [12] Shen, F. – Wang, H.: *A local edge detector used for finding corners [online]*. School of Electrical and Electronic Engineering Nanyang Technological University, Singapore, 2001-12 [cit. 2007-05-21]. Dostupný z WWW: <http://www.ntu.edu.sg/home/hw/ICICS2001.pdf>.

- [13] Wikipedia: *Corner detection [online]*. 2006 [cit. 2007-05-21]. Dostupný z WWW:
http://en.wikipedia.org/wiki/Corner_detection.
- [14] Witkin, A. P.: *Scale-space filtering*. In Proceedings of 8th international joint conf. art. intell., s. 1019-1022, Karlsruhe, West Germany, 1983.
- [15] Yuille, A. L. – Poggio, T. A.: *Scaling theorems for zero-crossings*. IEEE Trans. Pattern Analysis and Machine Intell., 8:15–25, 1986.
- [16] Žára, J. – Beneš, B. – Sochor, J. – Felkel, P.: *Moderní počítačová grafika*. Druhé, přepracované a rozšířené vydání. Brno: Computer Press, 2004, s. 169–171. ISBN 80-251-0454-0