

**UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY**

**POROVNANIE 2D A 3D LOKÁLNYCH
PRÍZNAKOV**

Bakalárska práca

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

**POROVNANIE 2D A 3D LOKÁLNYCH
PRÍZNAKOV**

Bakalárska práca

Študijný program: Aplikovaná informatika
Študijný odbor: 2511 Aplikovaná informatika
Školiace pracovisko: Katedra Aplikovanej Informatiky
Školiteľ: RNDr. Paula Budzáková



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Sabína Fačkovcová
Študijný program: aplikovaná informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: aplikovaná informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Porovnanie 2D a 3D lokálnych príznakov
Comparison of 2D and 3D local features

Cieľ:

1. Prehľad existujúcich metód na detekciu lokálnych príznakov na 2D a 3D dátach.
2. Naštudovanie jednotlivých metód.
3. Porovnanie 2D a 3D metód.

Vedúci: Mgr. Paula Budzáková
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: prof. Ing. Igor Farkaš, Dr.

Spôsob sprístupnenia elektronickej verzie práce:
bez obmedzenia

Dátum zadania: 04.10.2016

Dátum schválenia: 09.10.2016

doc. RNDr. Damas Gruska, PhD.
garant študijného programu

študent

vedúci práce

Pod'akovanie

Ďakujem svojej vedúcej práce za profesionálne vedenie, cenné rady a za ľudský prístup pri našom spoločne strávenom čase pri konzultáciách a vytváraní práce.

Abstrakt

V počítačovom videní sú lokálne príznaky dôležitou súčasťou pri rozpoznávaní objektov. V našej práci sme spravili prehľad piatich 2D a piatich 3D metód na vyhľadávanie a deskripciu lokálnych príznakov. Vybrali sme také metódy, ktoré sú z nášho pohľadu najpoužívanejšie a najzaujímavejšie. 2D metódy, aj s novými kombináciami detektora a deskriptora, sme prakticky otestovali na základe metodológie testovania na dátach a testoch, ktoré sme si pripravili. Vyhodnotili sme ich výsledky, urobili analýzu a porovnali sme ich. V prípade 3D metód sme analýzu a vyhodnotenie vypracovali na základe teoretických poznatkov.

Kľúčové slová: *lokálne príznaky, detektor, deskriptor, 2D, 3D*

Abstract

In computer vision, local features are an important part of object recognition. In our work, we reviewed five 2D and five 3D methods for searching and description local features. We chose methods that are, in our view, the most used and interesting. We tested 2D methods practically, including their new detector and descriptor combinations, on the basis of testing methodology on data and tests which we prepared. We evaluated the results, made analysis and compared them. For 3D methods, analysis and evaluation was made by the basis of our theoretical knowledge.

Keywords: *local features, detector, descriptor, 2D, 3D*

Obsah

1	Problematika	12
1.1	Príznamy	12
1.2	Lokálne a globálne príznaky	13
1.2.1	Globálne príznaky	13
1.2.2	Lokálne príznaky	13
1.3	Detektory a deskriptory	14
1.3.1	Detektory	15
1.3.2	Deskriptory	16
2	Metódy	19
2.1	2D metódy	19
2.2	3D metódy	30
3	Testovanie	38
3.1	Metodológia	38
3.2	Databáza	39
3.3	Používané technológie	41
3.4	Testovanie	42
3.4.1	Testovanie časovej náročnosti	42
3.4.2	Testovanie deskriptorov	43
3.4.3	Testovanie deskriptorov - koláže	45
3.5	Analýza a porovnanie	47
3.5.1	2D metódy	47
3.5.2	3D metódy	51
4	Záver	55
	Zoznam použitej literatúry	58

Zoznam obrázkov

1.1	Typy kľúčových bodov. (Zľava) Schod, mriežka, čiara alebo hrana [10] . . .	14
1.2	Typy kľúčových bodov. (Zľava) Dvojitá čiara, oblúk alebo kontúra, oblasť maxima [10]	14
2.1	Pre každú oktávu škálového priestoru je začiatočný obraz opakovane konvolovaný Gaussiánmi pre vytvorenie množiny škálovo-priestorových oprazov naľavo. Prilahlé Gaussiánové obrazy sú navzájom odpočítavané aby vznikli Gaussiánové rozdiely obrazov, ktoré sú napravo. Po každej oktáve je Gaussiánový obraz prevzorkovaný faktorom 2 a proces sa znova opakuje. [13]	20
2.2	Detekcia maxima a minima pre bod X [13]	20
2.3	Obrázok zobrazuje pole deskriptora 2x2 vypočítané z množiny vzoriek 8x8 a ilustruje jeho výpočet. [13]	21
2.4	Zľava doprava: Gaussián druhej parciálnej derivácie v smere y a xy . Šedé oblasti sa rovnajú nule. [1]	22
2.5	Vstupy deskriptora predstavujú povahu základnej intenzity štruktúry. Vľavo: V prípade homogénnej oblasti sú všetky hodnoty relatívne nízke. Stred: V prítomnosti frekvencií v smere x hodnota $\sum d_x $ je vysoká ale všetky ostatné zastávajú nízke. Vpravo: Ak sa intenzita postupne zvyšuje v smere x , obe hodnoty $\sum d_x$ a $\sum d_x $ sú vysoké. [1]	24
2.6	Autokorelačné zakrivenie - hrubé čiary zaznamenávajú klasifikáciu rohov/okrajov/plôch a jemné čiary ekvivalentné kontúry. α a β sú úmerné hlavným zakriveniam lokálnej autokorelačnej funkcie. [6]	25
2.7	Znázornenie škálovo-priestorovej detekcie zaujímavých bodov [11]	26
2.8	Vzor pre deskriptor BRISK [11]	27
2.9	Hustotou invariantné vyhladzovanie. Vn_i^s množina susedov v_i^s prvého rádu. [4]	30
2.10	Lokálny súradnicový systém vypočítaný vo vrchole V_i [4]	31

2.11	Ukážka detekcie 3D zaujímavých bodov. a) voxelizácia tvaru do mriežky, b) detegované body a naspäť zložený do tvaru, c) detegované body reprezentované sférami [9]	32
2.12	Bod v a jeho okruhy susedov. v je označený ako čierny bod, okruh $V_1(v)$ je tvorený zelenými bodmi, okruh $V_2(v)$ je tvorený zelenými a modrými bodmi a okruh $V_k(v)$ je tvorený všetkými bodmi až po žlté. [16]	33
2.13	3D izotropická sférická mriežka - štruktúra signatúry pre deskriptor SHOT [19]	35
2.14	Znázornenie detekcie príznakov pomocou detektora MeshDOG - a) počítateľná sieť s bodmi, b) body po hľadani škálovo-priestorového extrému, c) body po určení prahu určitého povoleného percenta, d) body po detekcii rohov [22]	36
2.15	a) 3D histogram - polárne mapovanie používané na vytváranie histogramov pomocou binovania 3D vektorov; b) výber troch ortogonálnych rovín, na ktoré sa premieta 3D histogram; c) polárny koordinačný systém, ktorý sa používa na vytváranie histogramov (príklad s 8 rezmi); d) príklad typických priestorových a orientačných histogramov s použitím 4 priestorových polárnych rezov a 8 orientácií. [22]	37
3.1	Ukážka databázy [8]: a) okruh hodiny, b) okruh kvety, c) okruh lampy, d) okruh stoličky, e) okruh stoly, f) okruh vázy.	40
3.2	5 hlavných objektov, ktoré sa nachádzajú na kolážiach a z ktorými prebehlo párovanie.	40
3.3	15 vytvorených koláží, ktoré obsahujú najmenej 2 objekty z množiny 5 hlavných objektov.	41
3.4	10 hlavných objektov	43
3.5	Ukážka obrázkov objektu z rôznych uhlov, ktoré sa nachádzajú v databáze na porovnanie	43
3.6	Metóda BRISK - 159 zhodných deskriptorov (TP pre prah 5)	44
3.7	Metóda BRISK - 0 zhodných deskriptorov (FN pre prah 5)	44
3.8	Znázornenie výsledkov pre 2D metódy v grafe v pomere presnosť/úplnosť	45
3.9	Metóda Harris SURF - 39 zhodných deskriptorov (TP pre prah 5)	46
3.10	Metóda Harris SURF - 14 zhodných deskriptorov (FP pre prah 5)	46
3.11	Znázornenie výsledkov pre 2D metódy v grafe v pomere presnosť/úplnosť	47

Zoznam tabuliek

3.1	Výsledky pre vybraný obrázok kvetu	42
3.2	Výsledky pre porovnávanie hľadaných objektov s databázou	45
3.3	Výsledky pre porovnávanie hľadaných objektov s kolážami	47
3.4	Analýza 2D metód	48
3.5	Analýza 3D metód	51

Úvod

V dobe vývoja autonómnych vozidiel, elektronických zdravotných pomôcok a mnohých iných nápadov na zlepšenie kvality života obyčajného človeka je priam nutnosťou vedieť správne rozpoznávať objekty alebo živé bytosti v reálnom čase alebo na statickom obraze. V oblasti počítačového videnia sa časť zaoberá práve korektným rozpoznávaním objektov a to pomocou lokálnych príznakov, ktoré definujú objekt podľa niekoľkých charakteristických bodov a zaručia tak možnosť rozoznať daný objekt.

Cieľom našej práce je naštudovanie metód, ktoré slúžia na detekciu a deskripciu lokálnych príznakov a vytvoriť ich prehľad. Vyberú sa metódy, ktoré pracujú na 2D aj 3D dátach a nájde sa databáza, pre ktorú sa vytvoria korektné testy na zistenie kvality metód na základe metodológie testovania. Výsledky z testovania nám poslúžia ako okruhy porovnávania metód.

V prvej kapitole sme sa rozhodli predstaviť pojem príznak a následne vysvetliť prečo sú práve lokálne príznaky vhodné pre rozpoznávanie objektov. Vysvetlíjeme postup ich hľadania pomocou detektorov a opisu pomocou deskriptorov.

V druhej kapitole sa pozrieme na konkrétne metódy, ktoré sme si vybrali a približujeme postup výpočtu lokálnych príznakov u týchto metód podľa oficiálnych dokumentácií od ich tvorcov. Kapitola je rozdelená na dve časti. Jedna časť pre 2D metódy a druhá pre 3D metódy.

Tretia kapitola je venovaná testovaniu a všetkému s ním spojeným. Predstavujeme metodológiu, podľa ktorej sme testovali funkčnosť vybraných metód a databázu, na ktorej testy prebiehali. Následne spomíname technológie, ktoré sme pri testovaní potrebovali a používali. Nakoniec príde samotné testovanie, jeho zhrnutie a výsledky, ku ktorým sme sa dopracovali. V poslednej časti tejto kapitoly metódy analyzujeme a porovnávame.

V závere celkovo zhrnieme našu prácu a spomenieme možnosti budúceho zamerania.

1. Problematika

V tejto kapitole sme sa zamerali na vysvetlenie pojmu príznakov a aké základné triedy príznakov v počítačovom videní poznáme. V podkapitole 1.1 sme sa venovali príznakom, pokúsili sa tento pojem vysvetliť a spomenuli sme aké rôzne typy príznakov poznáme. Podkapitola 1.2 bola zameraná na určité druhy príznakov a to konkrétne na globálne a lokálne príznaky, objasnili sme si tieto pojmy a spomenuli sme pozitíva a negatíva pri práci s nimi. Následne podkapitola 1.3 sme sa venovali problematike detektorov a deskriptorov. Spomenuli sme ako sa používajú a aký je rozdiel medzi detektorom a deskriptorom.

1.1 Príznaky

V oblasti počítačového videnia je podstatné vedieť ako je pozorovaný objekt reprezentovaný v obraze. Preto sa stal pojem príznak v oblasti počítačového videnia ustáleným pojmom. Pre porozumenie pojem príznak je výsledok merania, ktorý znázorňuje nejakú vlastnosť objektu. Vďaka používaniu príznakov pri spracovaní obrazu je objem dát, s ktorými pracujeme podstatne menší ako keby sa pracovalo bez nich a to so všetkými údajmi z obrazu.

Príznaky sa delia na rôzne typy na základe rôznych atribútov. Na základe úrovne abstrakcie príznaky rozdelíme na: [20]

- **Nízkoúrovňové príznaky** – sú to také príznaky, ktoré opisujú najzákladnejšie vlastnosti pozorovaného objektu ako sú príznaky farby, textúry a pod.
- **Stredoúrovňové príznaky** – tieto príznaky sú už spojením viacerých menších nízkoúrovňových príznakov. Je to napríklad kontrast, paleta farieb a pod.
- **Vysokoúrovňové príznaky** - tieto príznaky sa nachádzajú na najvyššej úrovni a opisujú celkové zobrazenie scény.

Nízkoúrovňové príznaky sa ďalej delia na bodové, globálne a lokálne. Bodové príznaky opisujú každý bod pozíciou, intenzitou alebo farbou. Globálnym a lokálnym príznakom sa budeme venovať v ďalšej podkapitole.

1.2 Lokálne a globálne príznaky

Pri rozpoznávaní objektov na obraz sa často používajú globálne príznaky, ktoré popisujú obraz ako celok, alebo lokálne príznaky, ktoré reprezentujú obraz pomocou zaujímavých častí v obraze.

1.2.1 Globálne príznaky

Globálne príznaky opisujú **celý obraz do jedného vektora** (vektora príznakov) a práve preto je používanie týchto príznakov pri štandardných technikách na rozpoznávanie objektov pomerne jednoduché. Väčšina deskriptorov tvaru a textúry patrí práve preto do tejto kategórie príznakov. Sú atraktívne na používanie, pretože globálny príznak obsahuje kompaktnú reprezentáciu obrazu.

Na druhej strane pri práci s globálnymi príznakmi je problém, ak sa na obraze nachádzajú rušivé elementy, pretože potom sa tieto príznaky získavajú ťažšie. V dôsledku toho sa predpokladá, že spracovávaný obraz obsahuje len jeden objekt alebo že je daný objekt dobre segmentovaný od pozadia. Avšak vo väčšine prípadov obraz neobsahuje len jeden objekt ale skladá sa z viacerých častí, preto je v takýchto prípadoch lepšie pracuje s lokálnymi príznakmi.

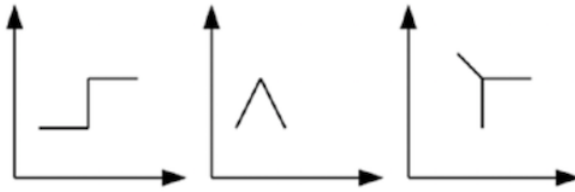
1.2.2 Lokálne príznaky

Lokálne príznaky sú príznaky, ktoré reprezentujú obraz pomocou zaujímavých bodov, ktoré sa v danom obraze nachádzajú. Zaujímavé body sa detegujú na obraze pomocou detektorov. Z týchto zaujímavých bodov vieme následne extrahovať lokálne príznaky použitím deskriptorov. Viac o detektoroch v podkapitole 1.3.

Lokálne príznaky sú využívané pri rozpoznávaní konkrétnych objektov na rôznych zmenených scénach. Dokážu definovať samotný objekt, ktorý sa nachádza na obrázku, a spárovať ho s rovnakým objektom na úplne odlišnom obrázku.

Lokálne príznaky vieme z obrázka extrahovať pomocou nájdenia zaujímavých bodov. Pod pojmom zaujímavé body si môžeme predstaviť také časti obrazu, ktoré sú výrazne odlišné od ostatných častí obrazu a vieme ho podľa nich identifikovať. Na jednom obraze sa nachádza hneď niekoľko zaujímavých bodov a preto lokálne príznaky dokopy dávajú o danom obraze viac informácií ako globálne príznaky.

V rôznych literatúrach sa stretávame s rôznymi termínmi pre zaujímavé body. Často sa používa aj termín kľúčové body (keypoints). [10]



Obr. 1.1: Typy kľúčových bodov. (Zl'ava) Schod, mriežka, čiara alebo hrana [10]



Obr. 1.2: Typy kľúčových bodov. (Zl'ava) Dvojitá čiara, oblúk alebo kontúra, oblasť maxima [10]

Na základe toho, koľko zaujímavých bodov sa na obraze nachádza, tak taký maximálny počet lokálnych príznakov vieme z obrazu extrahovať. Ak na obraze nedetegujeme žiadne zaujímavé body, tak z neho nevieme získať žiadne lokálne príznaky. Príkladom takéhoto obrazu je biele plátno alebo obraz, na ktorom sa nenachádzajú žiadne objekty. Na takýchto obrazoch by sme nevedeli detegovať žiadne zaujímavé body, z ktorých by sme následne vedeli extrahovať lokálne príznaky.

Metódy, ktoré sa používajú pri hľadaní zaujímavých bodov sa snažia detegovať také body, ktoré budú rovnaké pre ten istý objekt aj na rôznych obrazoch. Lokálne príznaky, ktoré sú detegované z takýchto zaujímavých bodov, sú označované ako **invariantné** vzhľadom transformáciu, škálovanie a orientáciu. [12]

1.3 Detektory a deskriptory

Algoritmy, ktoré sú používané na **vyhľadávanie zaujímavých bodov** sa nazývajú detektory a algoritmy, ktoré sú používané na **opis ich príznakov** sa nazývajú deskriptory.

V počítačom videní sa používa mnoho algoritmov na detekciu zaujímavých bodov v obrazoch. Najprv pomocou detektora nájdeme zaujímavé body a následne sa z nich pomocou deskriptorov získajú príznaky a to z okolitých pixelov, ktoré obkolesujú zaujímavý bod. Pre deskriptor zaujímavý bod poskytuje informácie o polohe, orientácii a osvetlení. Po spracovaní deskriptorom nám ešte k týmto informáciám pribudne ešte viac detailov a

atribútov. Skupiny viacerých zaujímavých bodov a príznakových deskriptorov nám potom dokopy podávajú dostatočnú informáciu o danom objekte.

Na popis príznakov používa veľa metód a ich variácií. Niektoré metódy na extrakciu príznakov nepoužívajú len detegované zaujímavé body. Napríklad deskripty mnohouholníkového tvaru vypočítavajú príznaky cez väčšie segmentované štruktúry alebo regióny mnohouholníkového tvaru v obraze.

1.3.1 Detektory

Každý detektor pre najlepšie výsledky pre daný obraz pracuje s inými parametrami na doladenie. Pri každom obraze môžu vzniknúť problémy spôsobené osvetlením, kontrastom a predspracovaním obrazu. Každý detektor je navrhnutý tak, aby bol užitočný pri rôznych triedach zaujímavých bodov a musí byť odladený pre určitý filter výsledkov, ktoré môžu byť následne vhodne použité špecifickým príznakovým deskriptorom. Pre každý detektor existujú deskripty, s ktorými sa mu pracuje najlepšie.

Kľúčové body sú preto filtrované podľa toho, aby boli použiteľné pre vybraný deskriptor. V niektorých prípadoch ale kľúčový bod nie je vhodný pre tvorbu vybraného príznakového deskriptora ani ak má vysoké skóre a odozvu. Ak príznakový deskriptor vypočíta pre daný kľúčový bod nízke skóre, tak by mali byť oba, aj kľúčový bod aj zodpovedajúci deskriptor, zamietnuté. Pre tieto prípady boli navrhnuté rôzne metódy, ktoré umožňujú používateľovi vyskúšať rôzne detektory a deskripty v spoločnom rámci a automaticky sa nastaví parametre pre ladenie.

Ako sme spomínali v predchádzajúcej podkapitole, tak poznáme niekoľko typov zaujímavých bodov, ktoré sú zobrazené na obrázku 1.1 a obrázku 1.2. Zaujímavé body by mali byť ľahké pre nájdenie a rýchlo vypočítateľné. **Zaujímavý bod** je teda kvalifikátor, okolo ktorého môže byť príznak vypočítaný.

Pri výbere zaujímavých bodov sú preferované skôr rohy ako hrany alebo izolované maximálne vrcholy, pretože roh je štruktúra a môže byť použitá na výpočet uhla, akým je objekt orientovaný, pre príznak. Zaujímavé body sa dobre počítajú pri farbených obrazoch ako aj pri šedotónových. Mnohé z metód ale najprv prejdú obraz Gaussovým filtrom pre odstránenie zbytočného šumu a až potom vykonajú gradientový operátor, pretože inak by mohol byť šum vplyvom gradientu ešte zosilnený.

Každý detektor lokalizuje prvky s rôznym stupňom nemennosti atribútov ako je

rotácia, perspektíva, oklúzia a osvetlenie. Na zhodnotenie kvality a výkonnosti hľadania zaujímavých bodov merajú metódy rôzne kritériá robustnosti a invariance štandardných dátových setov. Pre detekciu zaujímavých bodov je dôležitá miera invariance, pretože zaujímavé body sa môžu dramaticky zmeniť pri rôznych mierach invariance.

V mnohých metódach na detekciu zaujímavých bodov sa používajú výpočty z oblasti lineárnej algebry. Pixely sú uložené do matíc a na nich sa aplikujú výpočty ako je rozsah a smer gradientu, Laplacián, determinanty a pod. [10]

1.3.2 Deskriptory

Zaujímavé body, ktoré sú nájdené pomocou detektorov je potrebné spracovať pomocou deskriptorov. Tie opíšu zaujímavý bod a jeho lokálne okolie vo forme lokálnych príznakov, ktoré nám poskytnú obširnejšie informácie o objekte.

S efektívnymi algoritmami sa dokáže extrahovať veľké množstvo príznakov typických pre obrázky. Tieto príznaky sú dostatočne charakteristické a dokážu byť správne spárované s veľkou pravdepodobnosťou.

Príznakové deskriptory môžeme rozdeliť do viacerých samostatných kategórií. Rôzne kategórie sa často zaoberajú odlišnými problémami a je len málo záujmov, ktoré majú viaceré kategórie spoločné.

Príznakové deskriptory sa rozdeľujú na [10]:

- **Lokálne binárne deskriptory**
- **Spektrálne deskriptory**
- **Bázové priestorové deskriptory**
- **Mnohouholníkové deskriptory**
- **3D a 4D objemové deskriptory**

Lokálne binárne deskriptory

Táto kategória deskriptorov predstavuje príznaky ako binárno bitové vektory. Na výpočet príznaku sa porovnáva **páry bodov** a výsledok sa ukladá ako binárna hodnota vektora. Lokálne binárne deskriptory sú efektívne pre výpočet a efektívne na ukladanie. Oproti iným

metódam sú lokálne binárne vzory odolné a presné.

Príkladom pre lokálne binárne deskriptory sú metódy ORB, BRISK, LBP, FREAK a iné.

Spektrálne deskriptory

Skupina spektrálnych deskriptorov oproti lokálnym binárnym deskriptorom zahŕňa intenzívnejšie výpočty a algoritmy, ktoré často majú výpočty s pohyblivou desatinnou čiarkou a spotrebujú tak veľkú časť pamäte. **Spektrum** je množstvo, ktoré môžeme merať alebo počítať (ako napríklad intenzita svetla, farba, lokálny gradient, lokálny štatistický príznak, povrchové normály) a ukladať dáta (ako 2D a 3D histogramy akéhokoľvek spektrálneho typu ako napríklad histogram lokálneho smeru gradientu).

Najznámejšie spektrálne deskriptory sú SIFT a SURF, ktorým sme sa venovali v nasledujúcej kapitole.

Bázové priestorové deskriptory

Bázové priestorové deskriptory sa skladajú zo skupiny funkcií, bázových funkcií, ktoré sú spojené dokopy ako skupina.

Bázové funkcie môžu byť navrhnuté a použité na opis, rekonštrukciu alebo na tvorbu signálu. Vyžadujú transformáciu hodnôt do bázovej skupiny a inverznú transformáciu na prenos dát na originálne hodnoty.

Niekedy je užitočné transformovať sadu dát z jedného bázového priestoru do druhého na získanie prehľadu dát alebo na ich spracovanie alebo filtrovanie.

Bázové priestory sú podobné súradnicovým systémom, pretože oba majú nezvratné transformácie na príbuzných priestoroch. V niektorých prípadoch jednoduchá transformácia príznakového spektra do iného súradnicového systému urobí analýzu a výpočet jednoduchší a efektívnejší.

Mnohouholníkové deskriptory

Mnohouholníkové deskriptory vypočítavajú skupinu príznakov tvaru pre ľubovoľný mnohouholník. Tieto príznaky tvaru sú založené **na obvode tvaru mnohouholníka**. Metódy používané na určenie obvodu objektu, na zvýraznenie tvarov pred meraním a opisom sú často zložené ako napríklad prahovanie, segmentácia a morfológia. Keď už je tvar mnohouholníka zvýraznený, tak deskriptory tvaru sú vypočítané. Mnohouholníkové sú použiteľné pre väčšie oblasti príznakov.

3D a 4D objemové deskriptory

S príchodom množstva 3D senzorov (stereo kamery, akcelometre) a iných podobných senzorov zabudovaných v mobilných zariadeniach sa začali viac vyvíjať 3D a multimodálne príznakové deskriptory. Mnoho 3D deskriptorov sa spája práve s robotickým výskumom a 3D lokalizáciou.

Kľúčovou aplikáciou na výskume 3D deskriptorov zahŕňa robotiku a zaznamenávanie aktivity, kde sú príznaky sledované od jedného rámca k druhému ako sa formujú a deformujú. Cieľom je lokalizovať pozície a rozpoznávať ľudskú aktivitu ako je chodenie, kývanie rukou, skákanie a podobne. Vzhľadom na to, že 2D príznaky sa pohybujú počas sledovania aktivity, tak čas je tretia dimenzia zahrnutá v deskriptoroch.

Jednou z kľúčových konceptov pri rozpoznávaní aktivity je **rozšírenie už známych 2D príznakov do 3D priestoru**, ktorý je časopriestorový.

Informácia o hĺbke je v 3D metódach uchovávaná v rôznych formátoch v závislosti od použitia. Je potrebné aby sa tieto formáty dali medzi sebou konvertovať aby boli vhodné pre rôzne algoritmy. Medzi bežné formáty na reprezentáciu 3D dát patria [10]:

- **3D hĺbková mapa** (3D Depth Map) - pre 2D je v pamäti predvolený formát 2D pole pixelov. Pre obrázky hĺbkovej mapy je Z hodnota pixelu alebo hodnota hĺbky. Každý bod v poli môže obsahovať hodnotu pre farbu alebo hĺbku.
- **3D voxelový objem** (3D Voxel Volume) - 3D objemová dátová štruktúra je zložená z 3D poľa voxelov. Voxel definuje bod v trojrozmernom priestore (v 2D pixel). Každý voxel môže obsahovať hodnotu pre farbu alebo normálu. Štruktúra objemu je implicitná pre hĺbkové súradnice.
- **3D polygónová sieť** (3D Polygon Mesh) - 3D body sú uložené do štandardnej 3D polygónovej siete. Tá obsahuje množinu spojených bodov alebo vrcholov. Každý bod v sieti môže obsahovať súradnice, farbu, textúru alebo normálu a je spojený so susednými bodmi v štandardnom vzore.
- **3D mračno bodov** (3D Point Cloud) - Táto štruktúra sa dá ľahko skonvertovať na 3D polygónovú sieť. Každý bod v mračne bodov môže obsahovať informáciu o súradniciach, farbe alebo normále.

2. Metódy

V kapitole dva sme sa zamerali na konkrétne metódy a ich postup pri hľadaní a opise lokálnych príznakov. V podkapitole 2.1 sme sa venovali 2D metódam a priblížili si tie najznámejšie a najpoužívanejšie. Následne sme v podkapitole 2.2 prešli na 3D metódy.

2.1 2D metódy

SIFT

SIFT alebo **Scale Invariant Feature Transform** je metóda na vyhľadávanie zaujímavých bodov a príznakových deskriptorov vytvorená Lowem [13]. Táto metóda patrí do rodiny spektrálnych deskriptorov, ktoré sme si spomenuli v predchádzajúcej kapitole. Prevádza obrazové dáta do škálovo nemenných súradníc vzhľadom na lokálne príznaky.

Detektor SIFT v prvej fáze výpočtu deguje extrém v škálovom priestore. **Škálový priestor** je definovaný ako funkcia, $L(x, y, \sigma)$, ktorá je produktom konvolúcie škálovo-variabilného Gaussiána, $G(x, y, \sigma)$, a vstupného obrazu, $I(x, y)$:

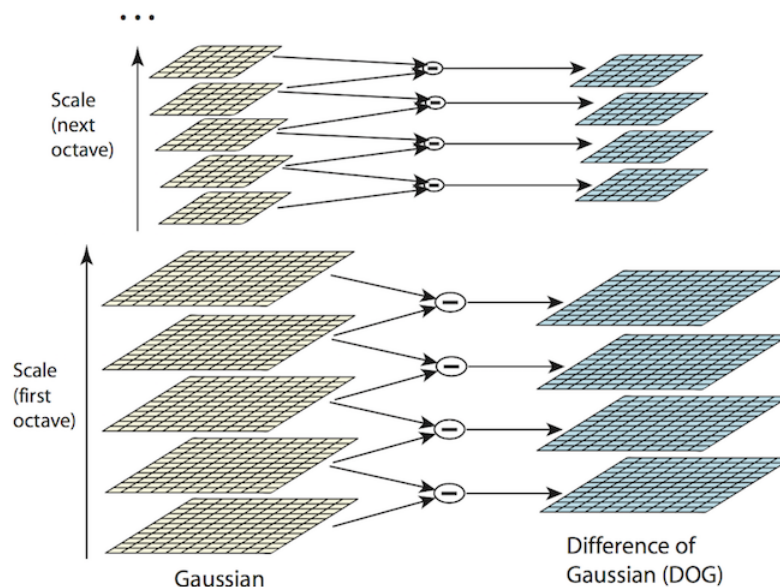
$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

kde $*$ je konvolúcia x a y , a:

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

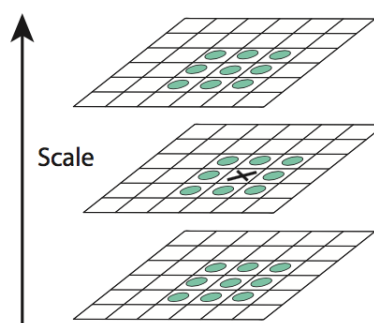
Pre efektívnu detekciu stabilnej pozície zaujímavého bodu sa použije priestorovo-škálový extrém pomocou **rozdielu Gaussiánov** (difference-of-Gaussian function) konvolovanej s obrazom, $D(x, y, \sigma)$, ktorý sa vypočítava z rozdielu dvoch blízkych mierok oddelených konštantným násobným faktorom k :

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned}$$



Obr. 2.1: Pre každú oktávu škálového priestoru je začiatkový obraz opakovane konvolovaný Gaussiánmi pre vytvorenie množiny škálovo-priestorových oprazov naľavo. Príslahlé Gaussiánové obrazy sú navzájom odpočítavané aby vznikli Gaussiánové rozdiely obrazov, ktoré sú napravo. Po každej oktáve je Gaussiánový obraz prevzorkovaný faktorom 2 a proces sa znova opakuje. [13]

Na detekciu lokálneho maxima a minima $D(x, y, \sigma)$ musí byť každý vzorkový bod (sample point) porovnaný s jeho ôsmymi susednými bodmi v aktuálnom obraze a s deviatimi susedmi v škále hore aj dole (Obr. 2.1). Náklady na túto kontrolu sú veľmi malé vzhľadom na fakt, že po niekoľkých kontrolách je eliminovaná väčšina vzorkových bodov.



Obr. 2.2: Detekcia maxima a minima pre bod X [13]

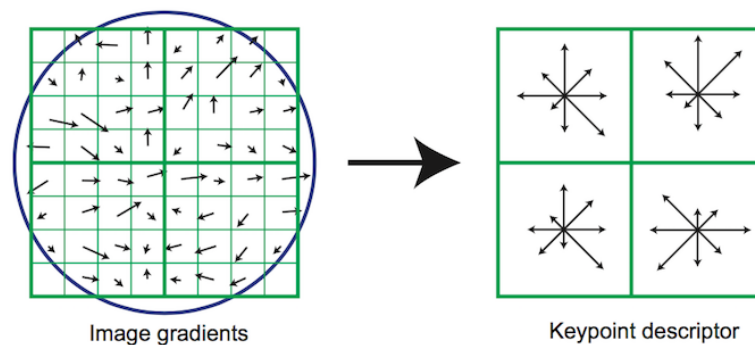
Akonáhle je nájdený kandidát na kľúčový bod porovnávaním pixelu s jeho susedmi, tak sa vykoná podrobné porovnanie s blízkymi dátami pre polohu, škálu a pomer hlavného zakrivenia. Táto informácia umožňuje odmietnuť body ak majú nízky kontrast (čiže sú citlivé na šum) alebo sú chudobnejšie rozmiestnené okolo hrany.

Chudobne definované vrcholy majú veľké hlavné zakrivenie medzi hranami ale malé v kolmo smere. Hlavné zakrivenie sa vypočíta z **Hesseho matice** 2x2 na polohe a škále

klúčového bodu. To je následne porovnané či náhodou príliš veľké.

Následne sa priradí konzistentná orientácia každému klúčovému bodu, vďaka čomu sa dosiahne invariancia k rotácii obrazu. **Histogram orientácie** je tvorený z gradientových operácií vzorkových bodov v oblasti okolo klúčových bodov. Vrcholy v histograme orientácie zodpovedajú dominantným smerom lokálnych gradientov. Najvyšší vrchol v histograme je detegovaný a potom každý iný lokálny vrchol, ktorý je 80 percent z najvyššieho vrcholu, sa taktiež používa na vytvorenie klúčového bodu s rovnakou orientáciou. Preto sa pre miesta s viacerými vrcholmi, ktoré sú rovnakého rozsahu, vytvorí viac klúčových bodov na rovnakom mieste ale s rôznymi smermi orientácie.

Predchádzajúce operácie priradili polohu, škálovanie a orientáciu každému klúčovému bodu. Tieto parametre uložili opakovateľný lokálny 2D súradnicový systém, ktorý opisuje lokálnu obrazovú oblasť a preto poskytuje invarianciu týchto parametrov. V nasledujúcom kroku sa vypočíta **deskriptor** pre lokálnu obrazovú oblasť, ktorá je už sama o sebe vysoko charakteristická.



Obr. 2.3: Obrázok zobrazuje pole deskriptora 2x2 vypočítané z množiny vzoriek 8x8 a ilustruje jeho výpočet. [13]

Gaussová váhová funkcia s σ , ktorá sa rovná polovici šírky okna deskriptora, sa používa na priradenie váhy veľkosti každému vzorkovému bodu. To je znázornené na ľavej strane obrázku 2.3. Účelom tohto Gaussovho okna je vyhnúť sa náhlym zmenám v deskriptore s malými zmenami v polohe okna a menším dôrazom na gradienty, ktoré sú ďaleko od stredu deskriptora, pretože práve oni sú najviac ovplyvnené chybou nesprávneho zaradovania.

Deskriptor klúčových bodov je zobrazený na pravej strane obrázku 2.3. Umožňuje posun pozícií gradientov vytvorením orientačných histogramov v oblasti vzoriek 4x4. Obrázok znázorňuje osem smerov pre každý orientačný histogram, pričom dĺžka každej šípky zodpovedá veľkosti záznamu histogramu. Zostava gradientu vľavo môže posunúť až 4 pozície vzorky, pričom stále prispieje k rovnakému histogramu vpravo, čím sú umožnené väčšie lokálne posuny v pozícií.

Deskriptor je vytvorený z vektora obsahujúceho hodnoty všetkých záznamov orientácie

histogramu zodpovedajúcich dĺžkam šípok na pravej strane obrázku 2.3. Nakoniec je príznakový vektor upravený tak, aby znižoval účinky zmeny osvetlenia. Preto je deskriptor invariantný k afinným zmenám v osvetlení.

Metóda SIFT takto dokáže vygenerovať veľké množstvo lokálnych príznakov, ktoré husto pokrývajú obraz v celom rozsahu. Ak sa pracuje s bežným obrázkom rozmerov 500x500 pixelov dostane sa z nej približne 2000 stabilných príznakov (záleží to od obsahu obrazu). Väčšie množstvo príznakov je dôležité pre následné rozpoznávanie objektov na obrazoch, kde schopnosť rozpoznávať malé objekty v preplnených obrazoch najmenej 3 správne spárované príznaky z každého objektu pre spoľahlivú identifikáciu.

SURF

SURF alebo **Speeded Up Robust Features** je deskriptor, ktorý vznikol ako vylepšenie SIFT-u a iných podobných metód na získavanie príznakov. Autormi tejto metódy sú Bay a kolektív [1]. Na detekciu zaujímavých bodov sa používa škálovo a rotačne nemenný detektor a na deskripciu lokálnych príznakov metóda SURF.

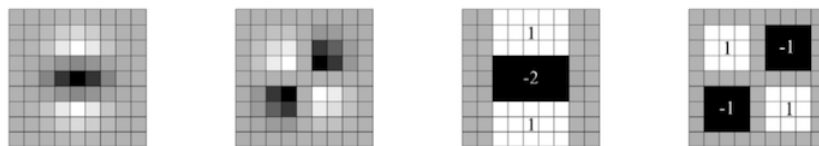
SURF oproti starším metódam pracuje lepšie s ohľadom na opakovanie, rozlišovanie, mohutnosť a ešte k tomu vypočítava príznaky omnoho rýchlejšie.

Na detekciu zaujímavých bodov sa používa **Fast-Hessian Detector**. Tento detektor je založený na **Hesseho maticiach**, kvôli ich dobrému výkonu vo vypočítavaní s ohľadom na presnosť a čas. Taktiež ako pri SIFT-e sa pracuje v škálovom priestore. Na jeho vytvorenie sa používa konvolúcia pôvodného obrazu pomocou aplikovaním postupne sa zväčšujúcich filtrov. Následne na nájdenie zaujímavých bodov sa v obraze potlačia maximálne hodnoty v okolí 3x3x3.

Pre bod $\mathbf{x} = (x, y)$ v obraze I je Hesseho matica $\mathcal{H}(\mathbf{x}, \sigma)$ v \mathbf{x} a škále σ je definovaná ako:

$$\mathcal{H}(\mathbf{x}, \sigma) = \begin{bmatrix} L_{xx}(\mathbf{x}, \sigma), L_{xy}(\mathbf{x}, \sigma) \\ L_{xy}(\mathbf{x}, \sigma), L_{yy}(\mathbf{x}, \sigma) \end{bmatrix}$$

kde $L_{xx}(\mathbf{x}, \sigma)$ je konvolúcia Gaussiána druhej parciálnej derivácie $\frac{\partial^2}{\partial x^2}g(\sigma)$.



Obr. 2.4: Zľava doprava: Gaussián druhej parciálnej derivácie v smere y a xy . Šedé oblasti sa rovnajú nule. [1]

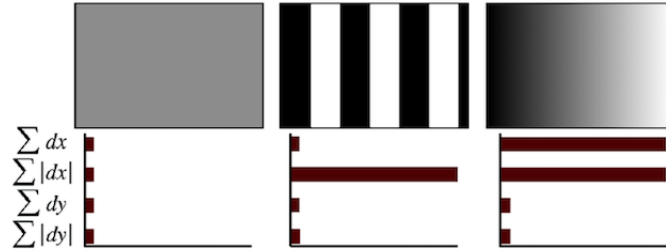
Deskriptor SURF je založený na podobných vlastnostiach ako SIFT ale oproti nemu je menej zložitý. Prvým krokom je zafixovať reprodukovateľnú orientáciu založenú na informáciách z kruhovej oblasti okolo zaujímavého bodu. Následne sa vytvorí štvorcová oblasť zarovnaná do vybraného smeru a extrahuje sa z neho deskriptor SURF-u.

Na identifikovanie orientácie pre zaujímavé body sa vypočíta **odozva Haarovho waveletu** v kruhovom okolí v smere x a y . Odozvy sú reprezentované vektormi v priestore s horizontálnou silou odozvy pozdĺž osi x a vertikálnou silou odozvy pozdĺž osi y . Dominantná orientácia je odhadnutá výpočtom **súčtu všetkých odoziev v posuvnej orientácii** so sprievodným uhlom $\frac{\pi}{3}$. Vodorovné a zvislé odozvy sú spočítané a z nich vznikne nový vektor. Ten najdlhší vektor prepožičia orientáciu zaujímavému bodu.

Na extrakciu deskriptora sa najskôr skonštruuje **štvorcová oblasť** sústredená okolo zaujímavého bodu a orientovaná pozdĺž orientácie. Oblasť sa rozdelí na pravidelných 4×4 štvorcových podoblastí. To udrží podstatnú informáciu vnútri. Pre každú podoblasť sa vypočíta pár jednoduchých príznakov na pravidelne rozmiestnených vzorkových bodoch 5×5 . Kvôli jednoduchosti sa zavolá d_x odozva Haarovho waveletu v horizontálnom smere a d_y odozva Haarovho waveletu vo vertikálnom smere. Horizontálny a vertikálny smer je tu definovaný v závislosti od zvolenej orientácie zaujímavého bodu. Pre zvýšenie odolnosti voči geometrickým deformáciám a lokalizácií chýb sú najprv odozvy d_x a d_y vyvážené s **Gaussiánom** centrovaným na mieste zaujímavého bodu.

Následne sa odozva Haarovho waveletu d_x a d_y z každej oblasti zhrnie a vytvorí sa prvá množina príznakových vektorov. Aby sa získali informácie polarizácie zmien intenzity, tak sa extrahuje súčet absolútnych hodnôt odoziev $|d_x|$ a $|d_y|$. Preto má každá podoblasť štvorrozmerný deskriptorový vektor \mathbf{v} pre svoju základnú štruktúru intenzity $\mathbf{v} = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$. Výsledkom je deskriptorový vektor pre všetky 4×4 podoblasti s dĺžkou 64. Odozvy Haarovho waveletu sú invariantné voči ovplyvneniu osvetlením. Invariantnosť kontrastu (škálový faktor) sa dosiahne otočením deskriptora na jednotkový vektor. Obrázok 2.5 znázorňuje vlastnosti deskriptora pre tri rozdielne odlišné vzory intenzity obrazu v rámci podoblastí.

Na rýchle indexovanie počas fázy porovnávania sa používa **Laplacián** (t.j. stopa Hesseho matice) pre príslušný zaujímavý bod. Zvyčajne sú zaujímavé body nájdené v štruktúrach typu blob (škvrna). Laplacián rozlišuje jasné škvrny na tmavom pozadí od opačnej situácie. Táto funkcia je k dispozícii bez ďalších výpočtových nákladov, pretože bola vypočítaná už počas detekčnej fázy. V porovnávacom štádiu sa porovnávajú iba príznaky, ktoré majú rovnaký kontrast. Tieto minimálne informácie umožňujú rýchlejšie párovanie a mierny nárast výkonu.



Obr. 2.5: Vstupy deskriptora predstavujú povahu základnej intenzity štruktúry. Vľavo: V prípade homogénnej oblasti sú všetky hodnoty relatívne nízke. Stred: V prítomnosti frekvencií v smere x hodnota $\sum |d_x|$ je vysoká ale všetky ostatné zastávajú nízke. Vpravo: Ak sa intenzita postupne zvyšuje v smere x , obe hodnoty $\sum d_x$ a $\sum |d_x|$ sú vysoké. [1]

Harrisov detektor rohov

Harrisov detektor rohov (Harris corner detector) je detektor vytvorený Chrisom Harrisom a Mikeom Stephensonom [6]. Vytvorili kombinovaný rohový a okrajový detektor založený na lokálnej autokorelačnej funkcii.

Na filtrovanie okrajov používa **Moravecov detektor rohov** (Moravec's corner detector) [14]. Ten funguje tak, že najskôr vyberie lokálne rámeček v obraze a určujú sa priemerné zmeny intenzity obrazu, ktoré sú výsledkom malých posunov okna v rôznych smeroch. Avšak Moravecov detektor rohov mal pár problémov, ktoré bolo treba upraviť pre potreby nového detektora.

Autokorelačná matica, ktorá sa používa na detekciu príznakov a opis lokálnej štruktúry je definovaná ako:

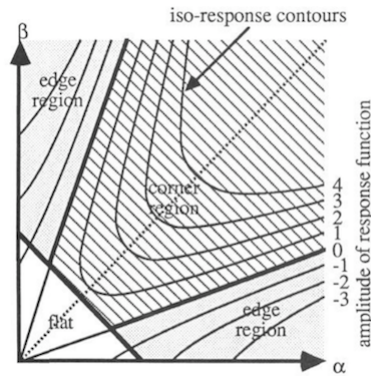
$$M(x, y) = \sum_{u, v} w(u, v) * \begin{bmatrix} I_x^2(x, y) & I_x I_y(x, y) \\ I_x I_y(x, y) & I_y^2(x, y) \end{bmatrix}$$

kde I_x a I_y sú lokálne obrazové derivácie v smere x a y , a $w(u, v)$ je váhové okno v oblasti (u, v) .

Po detekcii rohov treba následne zistiť rozsah kvality alebo odozvy rohov a okrajov. To je potrebné na výber izolovaných rohových pixelov a zúženie okrajových pixelov. Preto sa pre každý pixel vypočítava **Harrisova miera**:

$$h(x, y) = \det(M) - k \cdot (\text{trace}(M))^2$$

kde k je konštanta.



Obr. 2.6: Autokorelačné zakrivenie - hrubé čiary zaznamenávajú klasifikáciu rohov/okrajov/plôch a jemné čiary ekvivalentné kontúry. α a β sú úmerné hlavným zakriveniam lokálnej autokorelačnej funkcie. [6]

Pixel rohovej oblasti (má pozitívnu mieru) sa vyberie za nominovaný ak jeho miera je 8 miestnym lokálnym maximom. Podobne sú pixely okrajovej oblasti považované za okraje ak ich miery sú negatívne a taktiež aj lokálne minimá v x-vom alebo y-vom smere (podľa toho či je veľkosť prvého gradientu väčšia v x-vom alebo y-vom smere). Vďaka tomu vzniknú tenké hrany.

Použitím nízkych a vysokých prahových hodnôt je možné vykonať zmenu okrajov, čo môže zvýšiť kontinuitu okrajov. To má za následok 5-úrovňový obraz, ktorý pozostáva z pozadia, dvoch rohových tried a dvoch okrajových tried. Ďalšie spracovanie odstráni okrajové hrany a krátke izolované okraje a spojí krátke prerušenia okrajov. Vďaka tomu má obraz tenké okraje, ktoré končia v rohových oblastiach. Rohové zakončenia sú potom spojené s rohovými pixelmi, ktoré sa nachádzajú v rohových oblastiach, aby vytvorili prepojený graf okrajových vrcholov.

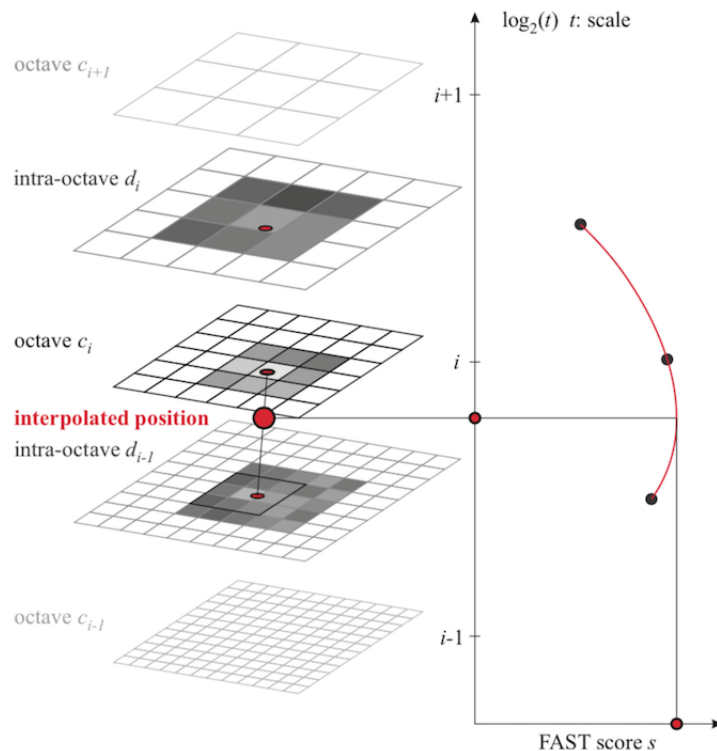
Keďže Harrisov detektor rohov je len detektor a neobsahuje deskriptor, tak ho treba pri rozpoznávaní objektov skombinovať s inými deskriptormi. Najčastejšie sa používa v kombinácii s deskriptorom SIFT-u. Táto kombinácia zabezpečí rýchle škálovo-invariantné rozpoznávanie objektov.

BRISK

BRISK alebo celým názvom **Binary Robust Invariant Scalable Keypoints** je metóda, ktorú vytvorili Stefan Leutenegger a kolektív [11]. Táto nová metóda vznikla ako variantIFT-u a SURF-u, ktoré sú lídrami medzi metódami v spracovaní obrazu. BRISK obsahuje detektor, ktorý je rotačne a škálovo invariantný a deskriptor bitových reťazcov.

Detektor BRISK v pracuje kontinuálnom škálovom priestore, kde odhaduje skutočnú škálu

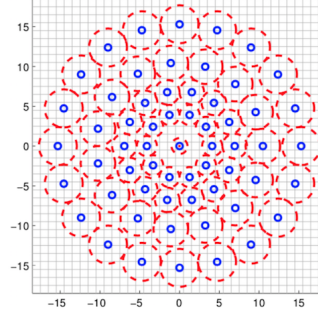
každého kľúčového bodu. Vrstvy v škálovo priestorovej pyramíde sú zložené z n oktáv c_i a n intra-oktáv d_i , kde $i = \{0, 1, \dots, n - 1\}$ a väčšinou $n = 4$. Tieto oktávy sú tvorené postupne polovičnými vzorkami z originálneho obrázku (zodpovedajúcemu c_0). Následne každá intra-oktáva je umiestnená medzi vrstvami c_i a c_{i+1} . Prvá intra-oktáva d_0 sa získa prevzorkovaním originálneho obrazu c_0 faktorom 1.5, kým zvyšné intra-oktávové vrstvy odvodené z postupných polovičných vzoriek. Preto ak t je škála, tak $t(c_i) = 2^i$ a $t(d_i) = 2^i * 1.5$.



Obr. 2.7: Znáročenie škálovo-priestorovej detekcie zaujímavých bodov [11]

V BRISK-u sa na detekciu kľúčových bodov používa 9-16 **FAST** [17], ktorý v podstate vyžaduje minimálne 9 po sebe idúcich pixelov v 16-pixelovej kružnici tak, aby boli buď dostatočne jasnejšie alebo tmavšie ako stredný pixel. Preto sa na začiatku aplikuje na každú oktávu a intra-oktávu tento detektor FAST 9-16 osobitne a vypočíta sa skóre s pre určenie oblastí záujmu. Skóre s je definované ako maximálna prahová hodnota, ktorá ešte stále uvažuje o bode rohu. Nemaximálne body sa potláčajú a to na každej oktáve a intra-oktáve. Následne sa vykonáva sub-pixelové a kontinuálne vylepšovanie rozsahu pre každé zistené maximum. Použije sa 2D kvadratická funkcia na vzorke 3x3 (zistené maximum, susediace pixely a sub-pixel). To sa vykoná na každej vrstve nad a pod. Následne sa tieto zdokonalené maximá použijú na prispôbenie 1D paraboly pozdĺž osi stupnice, ktorá prináša odhad konečného skóre a škály v maxime. Ako posledný krok sú opätovne interpolované súradnice obrazu medzi oblastami vo vrstvách vedľa určenej škály.

Deskriptor BRISK sa skladá ako binárny reťazec zlučovaním výsledkov jednoduchých testov porovnávaním jasou. Identifikuje sa charakteristický smer každého kľúčového bodu aby sa dosiahla rotačná invariancia.



Obr. 2.8: Vzor pre deskriptor BRISK [11]

Kľúčová koncepcia deskriptora BRISK používa vzor na odber vzoriek z okolia kľúčového bodu. Aby sa predišlo efektu aliasu pri odoberaní obrazovej intenzity bodu p_i , tak sa používa **Gaussovú vyhladzovanie** so smerodajnou odchýlkou σ_i úmerne k vzdialenosti medzi bodmi v príslušnom kruhu. Umiestnenie a zmenšovanie vzoru zodpovedá určitému kľúčovému bodu k na obrázku pre jeden z $N * (N - 1) / 2$ párov vzorkovacích bodov (p_i, p_j) . Vyhladené hodnoty intenzity v týchto bodoch sú $I(p_i, \sigma_i)$ a $I(p_j, \sigma_j)$ a tie sa používajú na určenie lokálneho gradientu $g(p_i, p_j)$ a to:

$$g(p_i, p_j) = (p_j - p_i) * \frac{I(p_j, \sigma_j) - I(p_i, \sigma_i)}{\|p_j - p_i\|^2}$$

\mathcal{A} je množina všetkých páry vzorkovacích bodov:

$$\mathcal{A} = \{(p_i, p_j) \in \mathbb{R}^2 \times \mathbb{R}^2 | i < N \wedge j < i \wedge i, j \in \mathbb{N}\}$$

pričom sa uvažuje podmnožina krátko-vzdialených (short-distance) párov \mathcal{S} a ďaleko-vzdialených (long-distance) párov \mathcal{L} :

$$\mathcal{S} = \{(p_i, p_j) \in \mathcal{A} | \|p_j - p_i\| < \delta_{max}\} \subseteq \mathcal{A}$$

$$\mathcal{L} = \{(p_i, p_j) \in \mathcal{A} | \|p_j - p_i\| > \delta_{min}\} \subseteq \mathcal{A}$$

Celkový charakteristický smer vzoru kľúčového bodu k sa hľadá:

$$g = \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{L} * \sum_{(p_i, p_j) \in \mathcal{L}} g(p_i, p_j)$$

Na vytvorenie rotačného a škálovo normalizovaného deskriptora BRISK sa aplikuje vzorkovací vzor otočený o $\alpha = \arctan 2(g_y, g_x)$ okolo kľúčového bodu k . Bit-vektorový deskriptor d_k sa zostavuje vykonaním všetkých porovnaní intenzity krátko-vzdialených párov bodov $(p_i^\alpha, p_j^\alpha) \in \mathcal{S}$, tak že každý bit b zodpovedá:

$$b \begin{cases} 1, & I(p_j^\alpha, \sigma_j) > I(p_i^\alpha, \sigma_i) \\ 0, & \text{inak} \end{cases} \quad \forall (p_i^\alpha, p_j^\alpha) \in \mathcal{S}$$

Následné hľadanie zhôd dvoch deskriptorov metódy BRISK je jednoduché vďaka tomu, že je to bitový deskriptor. Počet bitov odlišných medzi dvoma deskriptormi je mierou ich rozdielnosti.

ORB

ORB je ďalšia metóda, ktorá bola vytvorená ako efektívna alternatíva pre SIFT a SURF. Na jej tvorbe sa podieľali Ethan Rublee a kolektív [18]. Ako detektor používa modifikovaný detektor **oFAST** (Oriented FAST) a ako deskriptor sa používa modifikovaný deskriptor **rBRIEF** (Rotated BRIEF). To vysvetľuje, prečo sa metóda volá ORB - **O**riented **F**AST and **R**otated **B**RIEF.

Detektor ORB-u je založený na detektore **FAST** [17]. Avšak ten nedrží informáciu o orientácii kľúčového bodu a tak sa k nemu pridal dodatočný výpočet orientácie.

Začína sa detekciou kľúčových bodov pomocou FAST-9. FAST potrebuje jeden parameter a to prah intenzity medzi stredným pixelom a prah v kruhovom okolí stredu (keďže FAST-9, tak polomer kruhu je 9). Avšak FAST neprodukuje informáciu o miere rohov (measure of cornerness) a tak vznikajú príliš veľké odozvy pozdĺž okrajov. Preto sa použije **Harrisov detektor rohov** [6] na usporiadanie kľúčových bodov. Následne sa vytvorí škálová pyramída obrazu a produkujú sa kľúčové body, ktoré sú odfiltrované Harrisom, pre každú úroveň.

Na zistenie orientácie sa používa **intenzita centroidu**. Intenzita centroidu predpokladá, že intenzita rohu je odsunutá od jej stredu a tento vektor môže byť použitý na zistenie orientácie. Ak vzorka vyzerá ako:

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y)$$

tak sa nájde centroid:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

Z neho sa vytvorí vektor zo stredu rohu O do centroidu, \vec{OC} . Orientácia z neho sa vypočíta ako:

$$\theta = \text{atan2}(m_{01}, m_{10})$$

Deskriptor ORB-u je založený na binárnom deskriptore **BRIEF** [3], ktorý opisuje bitový

reťazec vzorky obrazu vytvorenej zo súboru testov binárnej intenzity.

Vzorka \mathbf{p} sa vyberie z vyhladeného obrazu a binárny test τ je na nej definovaný ako:

$$\tau(\mathbf{p}; x, y) := \begin{cases} 1 & : \mathbf{p}(x) < \mathbf{p}(y) \\ 0 & : \mathbf{p}(x) \geq \mathbf{p}(y) \end{cases}$$

kde $\mathbf{p}(x)$ je intenzita \mathbf{p} v bode x . Následne sa pre množinu príznakov n binárnych testov na mieste (x_i, y_i) definuje matica $2 \times n$:

$$\mathbf{S} = \begin{pmatrix} x_1, \dots, x_n \\ y_1, \dots, y_n \end{pmatrix}$$

Použije sa orientácia vzorky θ zodpovedajúca rotačnej matici \mathbf{R}_θ a následne sa vytvorí nasmerovaná verzia \mathbf{S} a to $\mathbf{S}_\theta = \mathbf{R}_\theta \mathbf{S}$. A následne sa vektor príznakov vypočíta ako:

$$f_n(\mathbf{p}, \theta) := \sum_{(x_i, y_i) \in \mathbf{S}_\theta} 2^{i-1} \tau(\mathbf{p}; x_i, y_i)$$

Po pridaní orientácie sa stráca invariancia a preto ešte treba redukovať koreláciu medzi binárnymi testami a to pridaním prehľadávacieho algoritmu greedy (pažravý), ktorý metódou učenia získa dobrú podmnožinu binárnych testov.

V konečnom párovaní ORB používa hashovanie citlivé na lokalitu, kde sú príznaky ukladané do hashovacích tabuliek a hashované do rôznych sektorov. ORB pri párovaní najbližšieho suseda nad veľkými databázami obrázkov prekonáva vo svojom výkone SIFT aj SURF.

2.2 3D metódy

LD SIFT

3D metód, ktoré sú založené na 2D metóde SIFT, vzniklo viacero. Tal Darom a Yosi Keller vytvorili **Local Depth SIFT** [4], ktorý deteguje zaujímavé body v trojrozmerných sieťach a vypočítava ich korešpondujúce deskriptory. Detektor LD SIFT na výpočet zaujímavých vrcholov a odhad lokálnej škály využíva rozdiel Gausiánov. Deskriptor sa vytvára reprezentovaním okolia zaujímavého vrcholu ako mapa hĺbky.

Detekcia začína aplikovaním **hustotou nemenného** (density-invariant) **Gaussového filtra** na sieťach. Odfiltrované siete sú označené ako sieťové oktávy $M^s = V^s, F$, kde $M^0 = M$. Tieto oktávy sú použité vo funkcii **rozdielu Gausiánov** a definujú sa lokálne maximá. Sieťové oktávy sa vypočítavajú pomocou lokálneho filtra s rovnomernou váhou. Pre každý vrchol v_i^s v oktáve s sa vypočíta vrchol v_i^{s+1} v ďalšej oktáve ako:

$$v_i^{s+1} = \frac{1}{|Vn_i^s|} \sum_{v_j^s \in Vn_i^s} v_j^s$$

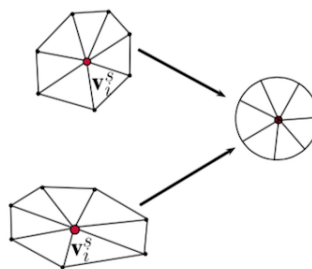
kde Vn_i^s je množina susedov v_i^s prvého rádu. Preto je filter invariantný k vzdialenosti medzi vrcholmi ale nie k ich umiestneniu. Nech D_i je lokálna hustota v v_i^s , tak:

$$D_i = \frac{1}{|Vn_i^s|} \sum_{v_j^s \in Vn_i^s} |v_i^s - v_j^s|$$

vd'aka ktorej sa nastaví odchýlka $\sigma_0 = D_i$. Funkcia rozdielu Gausiánov d_i^s v škále s , ktorá je invariantná k hustote, vyzerá takto:

$$d_i^s = \frac{1}{\sigma_s^2} |v_i^s - v_i^{s+1}|$$

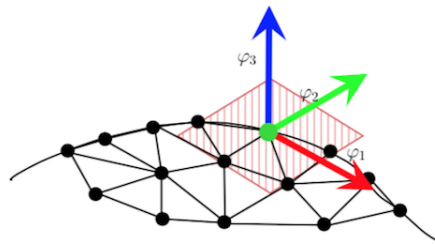
Následne body d_i^s v škále s , ktoré sú lokálne maximá v škále aj mieste sa vyberú za príznakové body.



Obr. 2.9: Hustotou invariantné vyhladzovanie. Vn_i^s množina susedov v_i^s prvého rádu. [4]

Deskriptor LD SIFT na všetkých detegovaných zaujímavých bodoch vypočíta rovinu. Nech $V = \{v_i^{S_i}\}$ je množina vrcholov v okruhu S_i od zaujímavého vrcholu V_i , kde S_i je lokálna škála V_i . Mapa hĺbky je vypočítaná premietaním vrcholov V na dominantnú rovinu P , ktorá je definovaná normálou \mathbf{n} a vrcholom V_i na nej. Normála sa vypočíta pomocou dvoch vedúcich vlastných vektorov $\{\varphi_1, \varphi_2\}$ v okolí V_i a lokálnej škále S_i na základe **princípu analýzy komponentov** (ako príznakové body sa vyberú vrcholy, ktorých pomer medzi dvoma hlavnými vlastnými hodnotami je maximálny). A teda $\mathbf{n} = \varphi_1 \times \varphi_2$.

Vypočíta sa vzdialenosť každého bodu na povrchu od P . Výsledkom je dvojrozmerné pole, pre ktoré je vypočítaný deskriptor SIFT. Pre získanie rotačnej invariencie sa definuje dominantný uhol, ktorého výpočet je robustný a opakovateľný. Využije sa smer vedúceho vlastného vektora φ_1 , ktorý je určite leží v rovine P . Takže dominantný uhol je nulový uhol a lokálny súradnicový systém, v ktorom je deskriptor vypočítaný je zarovnaný s týmto uhlom.



Obr. 2.10: Lokálny súradnicový systém vypočítaný vo vrchole V_i [4]

LD SIFT používa hĺbkový obraz ako zdroj svojho deskriptora a vypočítaný príznakový deskriptor je podobný tomu u 2D SIFT-u, ktorý je založený na deriváciách.

3D SURF

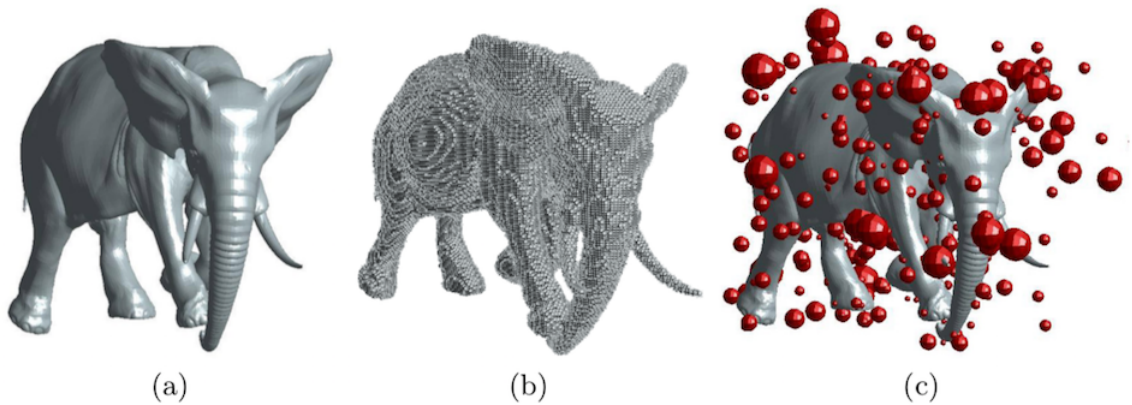
Taktiež podľa úspešného 2D predchodcu vytvorili Jan Knopp a kolektív [9] metódu na klasifikáciu 3D tvaru. Rozšírili 2D príznakový deskriptor SURF aby sa dal použiť na 3D tvary.

Každý **3D tvar** reprezentujú ako $m \in \mathcal{M}$, kde \mathcal{M} je množina tvarov. Na vytvorenie lokálne rotačných a škálovo invariantných zaujímavých bodov pre každý tvar z množiny \mathcal{M} trebalo SURF rozšíriť na 3 dimenzie.

Detekcia 3D zaujímavých bodov začína **voxelizáciou tvaru**. Objekt sa voxelizuje v 3D kocke veľkosti 256^3 použitím priesečníkov plôch stien s mriežkami. Každý tvar je rovnomerne zmenšený aby zapadol do kocky veľkosti 40 na každej strane. Následne sa vypočíta miera význačných rysov \mathcal{S} pre každú mriežku \mathbf{x} a niekoľko mierok σ (cez tri oktávy). \mathcal{S} je definovaná ako absolútna hodnota determinantu **Hesseho matice** $H(\mathbf{x}, \sigma)$ **Gaussiána druhej parciálnej derivácie** $L(\mathbf{x}, \sigma)$.

$$S(\mathbf{x}, \sigma) = |H(\mathbf{x}, \sigma)| = \begin{vmatrix} L_{xx}(\mathbf{x}, \sigma) & L_{xy}(\mathbf{x}, \sigma) & L_{xz}(\mathbf{x}, \sigma) \\ L_{yx}(\mathbf{x}, \sigma) & L_{yy}(\mathbf{x}, \sigma) & L_{yz}(\mathbf{x}, \sigma) \\ L_{zx}(\mathbf{x}, \sigma) & L_{zy}(\mathbf{x}, \sigma) & L_{zz}(\mathbf{x}, \sigma) \end{vmatrix}$$

Na rozdiel od 2D SURF-u pozitívna hodnota S nezaručuje, že všetky vlastné hodnoty H majú identické znaky. V dôsledku toho sa hľadajú nielen body podobajúce sa bloku, ale aj sedlovým (saddle) typom bodov. Finálna množina K_m zaujímavých bodov \mathbf{d}_{mk} , kde $k \in \{1 \dots K_m\}$, sa extrahuje z objemu použitím maximálnej supresie.



Obf. 2.11: Ukážka detekcie 3D zaujímavých bodov. a) voxelizácia tvaru do mriežky, b) detegované body a naspäť zložený do tvaru, c) detegované body reprezentované sférami [9]

V ďalšej fáze sa okolo každého detegovaného zaujímavého bodu vypočíta rotačne a škálovo invariantný **deskriptor 3D SURF**. Vo vzdialenosti $3 \times \sigma$ pozdĺž všetkých troch osí od každého bodu sa odoberie **odozva Haarovho waveletu**. Každá odozva je vážená **Gaussiánovým centrom**. Následne je prenesená do grafu v trojrozmernom priestore. Vektory odozvy sa spočítajú vo všetkých možných kužeľoch s uhlom otvárania $\frac{\pi}{3}$. Tak sa definuje smer najdlhšieho výsledného vektora ako dominantnú orientáciu. Po získaní dominantného smeru sa všetky odozvy premietajú pozdĺž tohto smeru, po ktorom sa druhá orientácia nájde pomocou posuvného okna. Dva získané smery úplne definujú lokálny rámec. Definovaná je mriežka $N \times N \times N$ (predpokladá sa $N = 3$) okolo príznaku. V každej bunke sa uloží **6-rozmerný opisný vektor** odoziev Haarovho waveletu.

Príznak k tvaru m obsahuje množinu informácií a to:

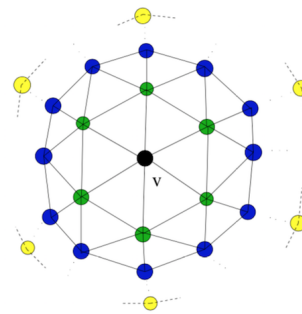
$$\mathbf{d}_{mk} = \{\mathbf{p}_{mk}, \sigma_{mk}, \mathbf{s}_{mk}\}$$

kde \mathbf{p}_{mk} reprezentuje relatívnu 3D pozíciu príznaku k stredu tvaru, σ_{mk} je škála príznakového bodu a \mathbf{s}_{mk} je 162-rozmerný vektor ($3 \times 3 \times 3 \times 6 = 162$) deskriptora 3D SURF-u príznakového vektora \mathbf{d}_{mk} .

3D Harris

Harrisov detektor rohov sa pre svoju jednoduchosť a efektívnosť používa v mnohých aplikáciách na spracovanie. Pre spracovanie 3D objektov však bolo treba základnú metódu upraviť. Pri 3D dátach sa oproti 2D bolo treba vziať do úvahy ľubovoľnosť topológie a akým spôsobom treba vypočítať derivácie potrebné k výpočtu Harrisovej miery. Na vytvorenie 3D verzie sa podujali Ivan Sipiran a Benjamin Bustos [16].

3D objekt je reprezentovaný ako množina vrcholov V a množina strán F s informáciou o susedoch medzi týmito entitami. Nech v je vrchol a $V_k(v)$ je susedná oblasť okolo neho a zvažované kruhy k v okolí vrcholu.



Obr. 2.12: Bod v a jeho okruhy susedov. v je označený ako čierny bod, okruh $V_1(v)$ je tvorený zelenými bodmi, okruh $V_2(v)$ je tvorený zelenými a modrými bodmi a okruh $V_k(v)$ je tvorený všetkými bodmi až po žlté. [16]

Z daných údajov sa vypočíta **centroid** z $V_k(v)$ a množina bodov sa transformuje tak, aby centroid bol v pôvodnom 3D súradnicovom systéme. Následne sa vypočíta najlepšia rovina na zmenené body. Pre výpočet derivácií sú zmenené body prispôbené na kvadratickú rovinu a je vypočítaný paraboloid v tvare:

$$z = f(x, y) = \frac{p_1}{2}x^2 + p_2xy + \frac{p_3}{2}y^2 + p_4x + p_5y + p_6$$

Následne sa použije integrácia derivácií s kontinuálnou Gaussiánovou funkciou, ktorá odstráni aj prípadný šum, a to nasledovne:

$$A = \frac{1}{\sqrt{2\pi}\sigma} \int_{R^2} e^{-\frac{(x_2+y_2)}{2\sigma^2}} \cdot f_x(x, y)^2 dx dy$$

$$B = \frac{1}{\sqrt{2\pi}\sigma} \int_{R^2} e^{-\frac{(x_2+y_2)}{2\sigma^2}} \cdot f_y(x, y)^2 dx dy$$

$$C = \frac{1}{\sqrt{2\pi}\sigma} \int_{R^2} e^{-\frac{(x_2+y_2)}{2\sigma^2}} \cdot f_x(x, y) \cdot f_y(x, y) dx dy$$

kde σ je parameter, ktorý definuje podporu Gaussiánovej funkcie.

Nakoniec je sformulovaná matica M priradenú bodu v z hore vypočítaných hodnôt a to:

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

a z nej vypočítať **Harrisova miera**.

Na základe hodnoty Harrisovej miery sa vyberú body, ktoré sú lokálne maximá. Z týchto bodov sa výsledná množina **zaujímavých bodov** môže vybrať dvoma spôsobmi. Prvá možnosť je vybrať tie body, ktoré majú najväčšiu odozvu a tak získať zaujímavé body s vyššou dôležitosťou a niektoré časti zostanú bez zaujímavých bodov. Alebo druhá možnosť je nájsť zástupcov skupín zaujímavých bodov a to tak, že najprv sa zoradia predtým vyčlenené zaujímavé body zaujímavé body na základe Harrisovej miery v zostupnom poradí a následne sa použije algoritmus na zhromaždenie utriedených bodov a výber konečného súboru zaujímavých bodov. Druhým spôsobom sa získajú body, ktoré sú rovnomerne rozložené na povrchu objektu

SHOT

SHOT je lokálny 3D deskriptor, ktorý bol vytvorený Samuelom Santim a kolektívom [19]. Táto metóda je inšpirovaná metódami založenými na prístupoch ako sú Signatúry (podpisy) a Histogramy, ktoré zdôrazňujú popisnosť a robustnosť.

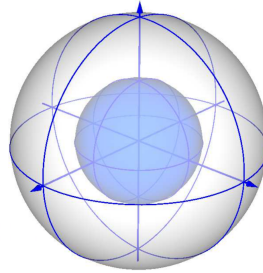
Pre zabezpečenie invariantnosti deskriptora k rotácii, transformácií a škálovaniu autori vytvorili **lokálny referenčný rámec** z vlastného vektora modifikovanej matice M . Zvažujú vzorové body p_i , ktoré ležia v podpornej oblasti polomeru r na základe ich vzdialenosti od zvažovaného bodu p .

$$M = \frac{1}{\sum_{i:d_i \leq r} (r - d_i)} \sum_{i:d_i \leq r} (r - d_i)(p_i - p)(p_i - p)^T$$

kde $d_i = \|p_i - p\|_2$. Na vytvorenie jedinečného lokálneho referenčného rámca a odstránenie nejednoznačnosti znaku je orientácia lokálnej osi x a z orientovaná smerom k hlavnému smeru vektorov, ktoré reprezentujú. Os y je získaná ako $y = x \times z$.

Na vytvorenie signatúry ako štruktúry sa použije **3D izotropická sférická mriežka**, ktorá je v súlade s odhadovaným lokálnym referenčným rámcom. Táto 3D sférická mriežka má 32 častí vzniknutých z 8 azimutov, 2 výškových a 2 radiálnych deliacich plôch. 3D rozdelenie bodov v každom z 32 oblastí je reprezentované lokálnym histogramom vytvoreným zhromaždením bodov do zásobníkov podľa kosínusu uhla medzi osou z v príznakovom bode p a susednými bodmi p_i , ktoré ležia v podpornej oblasti. Jednotné

ukladanie bodov do zásobníkov zabezpečuje robustnosť v malých rozdieloch povrchových normál.



Obr. 2.13: 3D izotropická sférická mriežka - štruktúra signatúry pre deskriptor SHOT [19]

Pri hraničných efektoch vyplývajúcich z histogramu založenom na **binovaní** (ukladaní do zásobníkov) a malom rušení v lokálnom referenčnom rámci sa použije technika kvadrilineárnej interpolácie. Následne je deskriptor normalizovaný tak, aby bol robustný proti zmenám hustoty.

Príznakový deskriptor SHOT je **352 dimenzionálny vektor** pozostávajúci z 11 binových histogramov, ktoré vznikli z 32 priestorových mriežok v 3D priestore. Každý histogram reprezentuje uhly, ktoré sú vytvorené povrchovými normálami v určitej priestorovej mriežke s lokálnym referenčným rámcom pre daný kľúčový bod. [15]

MeshDOG a MeshHOG

Po vzoroch detektorov a deskriptorov lokálnych príznakov pre 2D obrázky vznikla nová dvojica detektoru a deskriptoru pre 3D dáta vytvorená Andreiom Zaharescuom a kolektívom [22]. **3D detektor MeshDOG** a **3D deskriptor MeshHOG** boli navrhnuté pre 3D siete invariantné k zmenám rotácie, transformácie a škálovania. Deskriptor je schopný stručne zachytiť lokálne geometrické a fotometrické vlastnosti.

Detekcia príznakov **detektorom MeshDOG** sa začína hľadaním extrémov funkcie Laplaciána naprieč celou škálou susedstva prvého rádu. Na sieti je definovaná **skalárna funkcia** f . Použije sa konvolúcia na sieťach so základom k , ktorá je definovaná ako:

$$(f * k)(x) = \frac{\sum_{y \in \mathbf{N}_r(x)} k(\|x - y\|) f(y)}{\sum_{y \in \mathbf{N}_r(x)} k(\|x - y\|)}$$

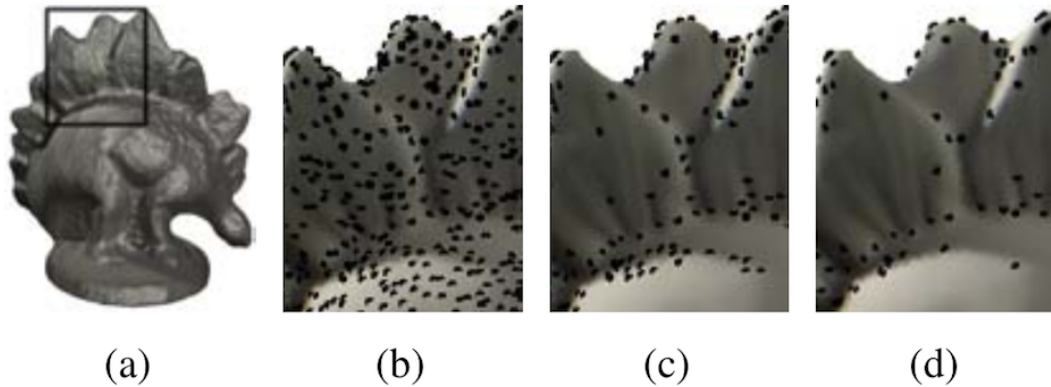
kde $\|x - y\|$ je vzdialenosť medzi bodmi x a y a \mathbf{N}_r je množina susedov bodu x prvého rádu. Následne sa konvolúciou funkcie f s Gausiánovým jadrom g_σ šírky σ zostrojí škálový priestor ($f_0 = f$, $f_k = f_{k-1} * g_\sigma$). Operátor **rozdielu Gausiánov** (DoG) v škále je definovaný ako $DoG_k = f_k - f_{k-1}$.

Maximá DoG škálového priestoru naprieč škálou sú vybrané ako zaujímavé body, pričom

sa zvolí pevné percento bodov s najvyššími hodnotami DoG. Pre elimináciu nestabilných odoziev sa zachovávajú vlastnosti, ktoré predstavujú rohy. To sa zabezpečí vypočítaním **Hesseho operátora** pre každý bod x :

$$\mathbf{H} = \begin{pmatrix} f_{uu}(x) & f_{uv}(x) \\ f_{uv}(x) & f_{vv}(x) \end{pmatrix}$$

kde f_{uu} , f_{uv} a f_{vv} sú parciálne derivácie druhého rádu f na x .



Obr. 2.14: Znáozornenie detekcie príznakov pomocou detektora MeshDOG - a) počiatočná sieť s bodmi, b) body po hľadaní škálovo-priestorového extrémumu, c) body po určení prahu určitého povoleného percenta, d) body po detekcii rohov [22]

Príznakový deskriptor sa nazýva **MeshHOG**. Prvým krokom k jeho nájdeniu je zvolenie lokálneho súradnicového systému aby sa stal invariantný k rotáciám. Následne sa vypočíta histogram gradientu. Keďže gradientové vektory sú trojrozmerné, histogramy sú vypočítané v 3D.

Deskriptor t_v pre vrchol v je vypočítaný pomocou podpornej oblasti, ktorá je definovaná podľa veľkosti okruhu susedstva r . Pre každý vrchol zo susedstva $v_i \in \mathbf{N}_r(v)$ sa vypočíta **gradientový operátor** $\nabla_S f(v_i)$ z f vo vrchole $v_i \in S$ a to takto:

$$\nabla_S f(v_i) = \sum_{v_j \in \text{rg}(v_i, 1)} (w_{ij} D_{e_{ij}} \rightarrow f(v_i)) \vec{u}_{ij}$$

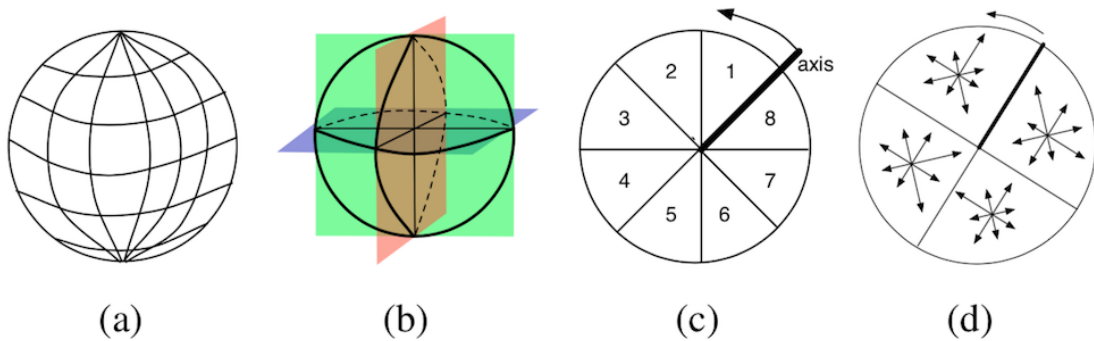
kde w_{ij} je váha diskkrétnej smerovej derivácie $D_{e_{ij}} \rightarrow$ a \vec{u}_{ij} je normalizovaný smer $v_i \vec{v}_j$ v rovine dotýčnice vo vrchole v_i .

Lokálny súradnicový systém je navrhnutý použitím normálového vektora \vec{n}_v a dvoch jednotkových vektorov, ktoré sa nachádzajú v rovine \mathcal{P}_v dotýčnice k vrcholu v . Ak je daný jednotkový vektor $\vec{a}_v \in \mathcal{P}_v$, tak lokálny súradnicový systém je určený pomocou $\{\vec{a}_v, \vec{n}_v, \vec{a}_v \times \vec{n}_v\}$. Vektor \vec{a}_v je vypočítaný ako smer priradený k dominantnému zásobníku v polárnom histograme, pričom $b_a = 36$ zásobníkov (binov). Následne je vypočítaný histogram a to tak, že sa posúdia vrcholy v_i premietané na \mathcal{P}_v , kde sa berie do úvahy ich

gradientová odozva.

Gradientové vektory sú premietnuté na 3 ortonormálne roviny opisujúce lokálny súradnicový systém. Pre každú z 3 rovín sa vypočíta histogram s dvoma úrovňami. Najskôr sa rovina rozdelí na $b_s = 4$ rezy začínajúc počiatkom a pokračujú v smere pravidla pravej ruky vzhľadom na druhý vektor ortonormálnej osi. Každý vrchol v_i spadá do jedného z priestorových rezov. Následne sa pre priestorový každý rez sa vypočíta orientačný histogram s $b_o = 8$ zásobníkov pre každý z premietnutých gradientových vektorov $\nabla_S f(v_i)$ vrcholov v_i .

Konečný deskriptor sa získa spojením $b_s \times b_o$ hodnôt histogramov pre každú z troch rovín.



Obr. 2.15: a) 3D histogram - polárne mapovanie používané na vytváranie histogramov pomocou binovania 3D vektorov; b) výber troch ortogonálnych rovín, na ktoré sa premieta 3D histogram; c) polárny koordinačný systém, ktorý sa používa na vytváranie histogramov (príklad s 8 rezmi); d) príklad typických priestorových a orientačných histogramov s použitím 4 priestorových polárnych rezov a 8 orientácií. [22]

3. Testovanie

Tretia kapitola je zameraná na testovanie metód. Nachádza sa tu metodológia, podľa ktorej sme vyhodnocovali naše dosiahnuté výsledky, použitá databáza, z ktorej sme čerpali, a technológie, s ktorými sme pri testovaní pracovali. Dôležitou súčasťou tejto kapitoly je postup samotného testovania a dosiahnuté výsledky. V neposlednom rade sa v tejto kapitole nachádza analýza a porovnanie metód.

3.1 Metodológia

Pri testovaní metód z kapitoly 2 sme sa rozhodli pre sledovanie určitých vlastností, podľa ktorých sme ich následne porovnávali.

Testovanie časovej náročnosti

Časová náročnosť je jednou z najčastejších skúmaných vlastností na metódach nielen v počítačovom videní ale celkovo v oblasti informačných technológií. My sme sa zamerali na zistenie času, za koľko sekúnd sa jednotlivým metódam podarí detegovať zaujímavé body a za aký čas z nich deskriptor vypočíta lokálne príznaky, na rovnakých objektoch.

Testovanie deskriptorov

Heinly a kolektív vo svojej práci [7] porovnávali binárne deskriptory a podľa ich príkladu a príkladu hodnotenia kvality klasifikácie v knihe p.Šikudovej a kolektívu [20] sme testovali vlastnosti ako je **presnosť, úplnosť a efektívnosť**.

Pre lepšiu zrozumiteľnosť testovania sú definované 4 možnosti aké môžu pri klasifikácií nastať. Ak máme objekt, ktorý chceme nájsť na druhom obraze a vieme, že sa na druhom obraze aj reálne nachádza, tak môžu nastať dve možnosti. Ak klasifikátor správne deteguje, že hľadaný objekt sa nachádza na druhom obraze, tak je to **správne pozitívny výsledok** (true positive - TP) ale ak ho na tomto obraze nenájde, tak ide o **nesprávne negatívny výsledok** (false negative - FN). Podobne to funguje v opačnom prípade, keď sa hľadaný objekt na danom obraze určite nenachádza. Ak objekt na takomto obraze deteguje, tak ide o **nesprávne pozitívny výsledok** (false positive - FP) ale ak určí, že sa na obraze nenachádza, tak je to **správne negatívny výsledok** (true negative - TN).

Presnosť je definovaná ako počet správnych určených zhôd z množiny všetkých zhôd. Nazýva sa aj ako pozitívna prediktívna hodnota. Čím menej je nesprávne pozitívnych výsledkov, tým viac sa hodnota blíži k jednotke. Výpočet vyzerá takto:

$$\text{Presnosť} = \frac{TP}{TP + FP}$$

Úplnosť (recall) alebo aj citlivosť klasifikátora vyčíslí uje koľko bolo nájdených správnych pozitívnych zhôd z možných správnych zhôd. Táto hodnota závisí od schopnosti detektora vytvárať zhodu. Čím je viac nesprávne negatívnych výsledkov, tým je citlivosť klasifikátora nižšia. To znamená, že klasifikátor nevie odhaliť všetky výskyty objektu. Vzorec na výpočet je:

$$\text{Úplnosť} = \frac{TP}{TP + FN}$$

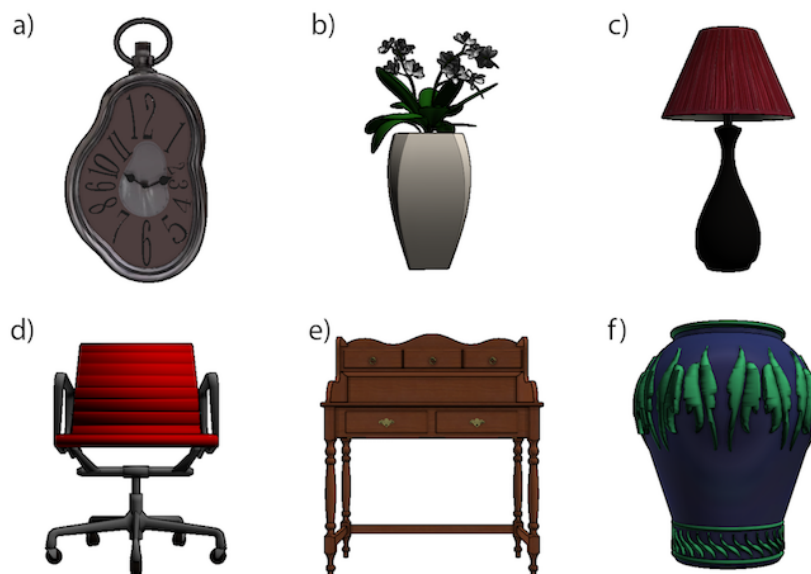
Efektívnosť alebo účinnosť je pomer všetkých správnych zhôd (správne pozitívnych aj správne negatívnych) a všetkých skúmaných príznakov. Čím je menej nesprávnych výsledkov, tým je efektívnosť vyššia. Výpočet prebieha ako:

$$\text{Efektívnosť} = \frac{TP + TN}{TP + FP + TN + FN}$$

3.2 Databáza

Pre účely testovania na metódach spomenutých v kapitole 2 sme sa rozhodli použiť databázu **ShapeNet** [8], ktorá obsahuje rozsiahle súbory 3D tvarov radené podľa okruhov. Táto databáza vznikla spoločným úsilím výskumníkov z Princeton-u, Stanford-u a TTIC a poskytuje údaje pre výskum v oblasti počítačovej grafiky, počítačového videnia, robotiky a ďalších súvisiacich disciplín. ShapeNet je voľne dostupná pre nekomerčné výskumy a vzdelávacie účely.

Pre naše testovanie sme si vytvorili databázu 6 okruhov a to: hodiny, kvety, lampy, stoličky, stoly a vázy. Každý okruh obsahuje 10 objektov, čiže dokopy je naša databáza tvorená 60 3D objektmi. Tieto 3D objekty máme uložené aj vo forme 2D ako PNG súbory. Takýto jeden 3D objekt je zvyčajne uložený vo forme 14 PNG obrázkov. Pod pojmom obrázkov v našej databáze sa myslí daný objekt na bielom pozadí z rôznych uhlov pohľadu.



Obr. 3.1: Ukážka databázy [8]: a) okruh hodiny, b) okruh kvety, c) okruh lampy, d) okruh stoličky, e) okruh stoly, f) okruh vázy.

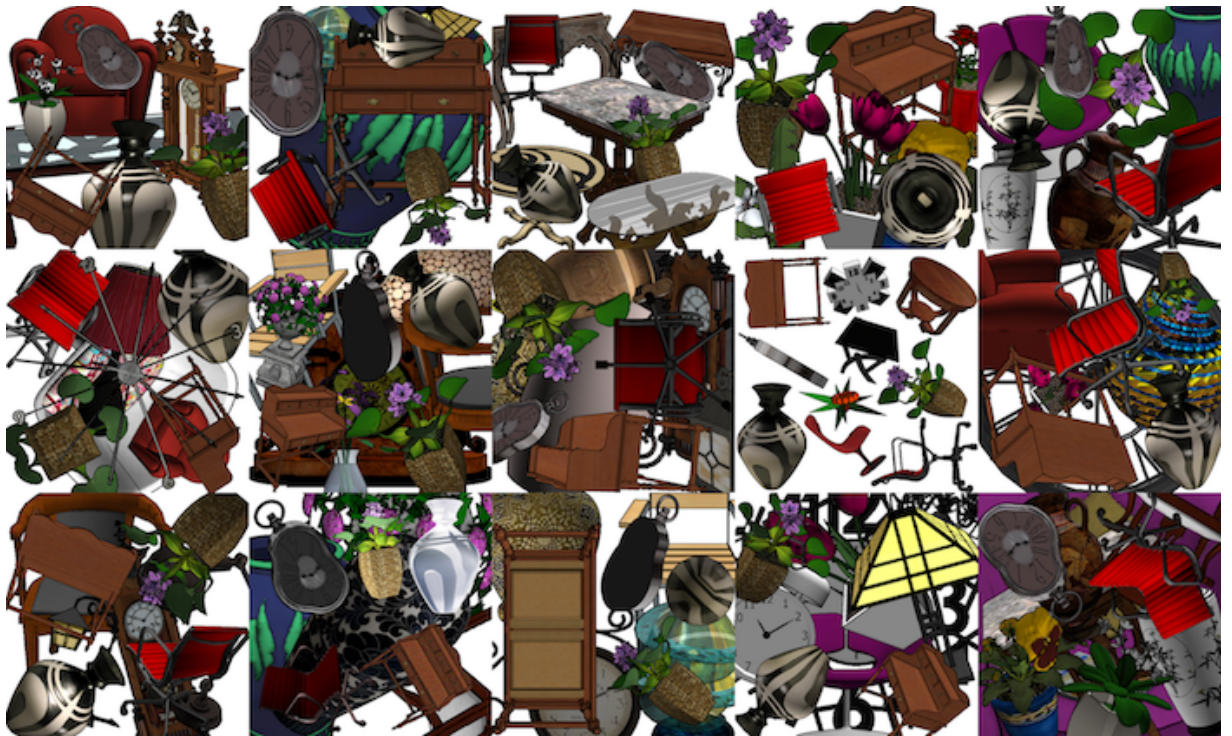
Koláže

Pre otestovanie schopnosti metód, ktoré boli spomenuté v predchádzajúcej kapitole, správne spárovať lokálne príznaky sme si z objektov z databázy vybrali 5 hlavných, ktoré sme zakomponovali do 15 koláží.

Tieto objekty sú na vytvorených kolážach vždy pozmenené oproti hlavnému obrázku. Buď sú pootočené, zmenšené, zväčšené, zaznamenané z iného uhlu pohľadu alebo inak transformované, aby sa mohlo korektne testovať ako veľmi sú metódy pri vytváraní detektorov a deskriptor invariantné voči zmenám a či sa im správne darí spárovať rovnaký objekt na obrázkoch. Koláže obsahujú najmenej 2 objekty z množiny 5 hlavných objektov.



Obr. 3.2: 5 hlavných objektov, ktoré sa nachádzajú na kolážach a z ktorými prebehlo párovanie.



Obr. 3.3: 15 vytvorených koláží, ktoré obsahujú najmenej 2 objekty z množiny 5 hlavných objektov.

3.3 Použité technológie

Testovanie metód prebiehalo v programovacom jazyku MATLAB a s podpornou knižnicou OpenCV.

MATLAB

MATLAB je maticovo založený programovací jazyk, ktorý má vlastné programovacie prostredie s rovnakým názvom. Vyvíjaný je spoločnosťou Mathworks a momentálne je dostupný pre operačné systémy Windows, Linux a MacOS. Zameraný je na numerické výpočty, modelovanie, vizualizáciu, návrhy algoritmov a pod. Názov vznikol podľa hlavného prvku, ktorý sa používa na výpočty, matice a preto MATLAB - MATrix LABoratory (maticové laboratórium).

MATLAB sme si vybrali z dôvodu, že medzi jeho vstavanými funkciami sa nachádzajú 3 metódy, ktoré sme testovali a to Harris, SURF a BRISK. Oficiálna implementácia SIFT je taktiež naprogramovaná v jazyku MATLAB. Konkrétne pri našom testovaní sme používali verziu r2017b a knižnicu OpenCV 3.4.1.

OpenCV (Open Source Computer Vision Library) [2] je voľne dostupná knižnica pre počítačové videnie. Bola navrhnutá pre výpočtovú efektívnosť so zameraním na aplikácie v reálnom čase. Táto knižnica je pôvodne navrhnutá pre jazyk C++ ale pomocou balíčka

mexopencv, ktorý vyvinul Kota Yamaguchi [21], je možné využívať túto knižnicu aj v prostredí MATLAB. Spojením MATLAB-u a OpenCV tak vznikne nástroj pre vývoj algoritmov, analýzu obrazu, videa a dizajn systémového videnia.

V tejto knižnici je naprogramovaná metóda ORB, ktorú sme testovali.

3.4 Testovanie

Všetky testy prebehli na operačnom systéme Windows 10 procesore 2,7 GHz Intel Core i5 s pamäťou 8 GB. Pri testovaní sme používali metódy Harris, SURF a BRISK, ktoré sú štandardne predprogramované v MATLAB-e, metódu ORB, ktorá je naprogramovaná v knižnici OpenCV [2] a demo verziu SIFT od Lowe-ho [13].

3.4.1 Testovanie časovej náročnosti

Prvé testovanie bolo zamerané na zistenie časovej náročnosti jednotlivých pôvodných metód a ich kombinácií. Pod kombináciami sa myslí skombinovanie detektora jednej metódy s deskriptorom inej. V našom prípade sme kombinovali detektor Harris, BRISK a SURF s deskriptormi SURF-u a BRISK-u. Zaujímali nás informácie o tom koľko jednotlivé metódy (pôvodné aj skombinované) dokážu vyprodukovať deskriptorov a za aký čas. Taktiež bolo naším cieľom zistiť či niektoré naše kombinácie, ktoré sme skúšali, nemajú lepšie výsledky ako pôvodné metódy.

Na testovanie sme si pripravili súbor 200 obrázkov z našej databázy 60 objektov spomenutej skôr 3.1. Každý objekt z 60 sa nachádzal najmenej na troch obrázkoch a to zachytený z rôznych uhlov. Obrázky obsahovali objekt na bielom pozadí. Pre ukážku sme sa rozhodli zobrazit' v tabuľke 3.1 výsledky pre jeden vybraný obraz a to konkrétne pre prvý kvet zobrazený na obrázku 3.5.

Tabuľka 3.1: Výsledky pre vybraný obrázok kvetu

detektor	deskriptor	počet detegovaných bodov	z toho deskriptorov	čas (ms)
SIFT	SIFT	778	778	911,440
SURF	SURF	519	519	54,710
	BRISK		490	224,126
Harris	/	366	/	/
	SURF		366	58,376
	BRISK		366	361,547
BRISK	BRISK	659	619	390,521
	SURF		659	245,762
ORB	ORB	3 515	3 515	33,051

3.4.2 Testovanie deskriptorov

Druhé testovanie prebiehalo na báze hľadania zhôd medzi dvoma obrázkami, kde sme skúmali schopnosti deskriptorov. Konkrétne sme zisťovali hodnoty pre presnosť, úplnosť a efektívnosť, ktoré sme spomenuli v metodológii testovania.

Vybrali sme si 10 hlavných obrázkov (obrázok 3.4), ktoré sme vyhľadávali v upravenej databáze z prvého testovania. V upravenej databáze sa hľadaný objekt nachádzal 10 krát a nie 3 krát ako v pôvodnej pre lepšie porovnanie. Na týchto 10 obrázkoch bol objekt zachytený z rôznych uhlov ako na príklade pre hľadaný kvet na obrázku 3.5.



Obr. 3.4: 10 hlavných objektov

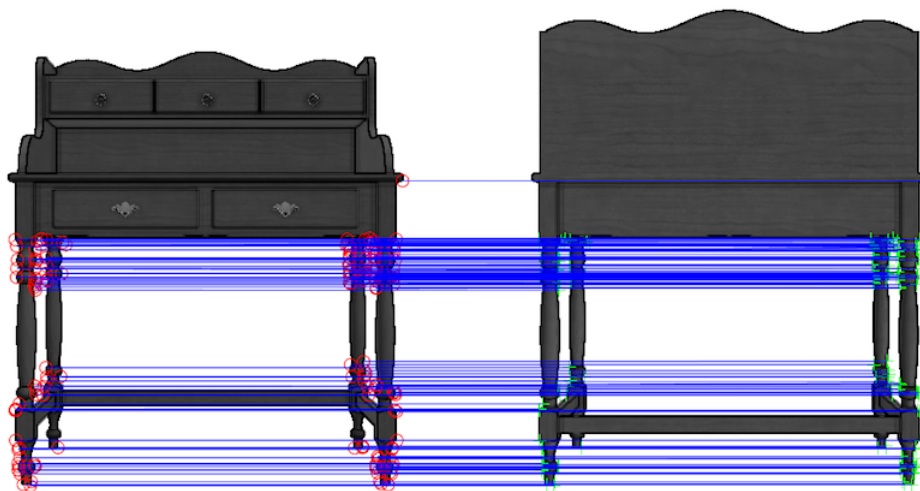


Obr. 3.5: Ukážka obrázkov objektu z rôznych uhlov, ktoré sa nachádzajú v databáze na porovnanie

Na každom obrázku s hľadaným objektom z hlavných objektov a obrázkoch z upravenej databázy prebehla detekcia a deskripcia metódami ako v prvom testovaní (aj pôvodné aj skombinované) a následne sa medzi nimi hľadala zhoda párovaním ich deskriptorov. Na párovanie sme si vybrali metódu hľadania **druhého najbližšieho suseda** (second nearest neighbour). Táto metóda najskôr podľa vzdialenosti utriedi deskriptory, ktoré chceme porovnávať, a následne ich porovná. Za zhodu sa považuje taký deskriptor,

ktorého vzdialenosť je 0,6 krát menšia ako vzdialenosť druhého najbližšieho deskriptora od porovnávaného deskriptora.

Na obrázkoch 3.6 a 3.7 je znázornené párovanie metódy BRISK. Hľadaný stôl je porovnaný s tým istým stolom ale z iného pohľadu. Pre obrázok 3.6 sa metóde podarilo úspešne spárovať deskriptory a pre prah 5 vyhodnotiť toto porovnávanie za správnu pozitívnu zhodu. Avšak na obrázku 3.7 sa nepodarilo spárovať ani jeden deskriptor a preto je daná zhoda nesprávne negatívna. Na obrázkoch sú v červenom kruhu znázornené zaujímavé body v hľadanom obrázku a modrou čiarou je znázornená zhoda so spárovanými bodmi v druhom obrázku, ktoré sú znázornené zeleným krížom.



Obr. 3.6: Metóda BRISK - 159 zhodných deskriptorov (TP pre prah 5)



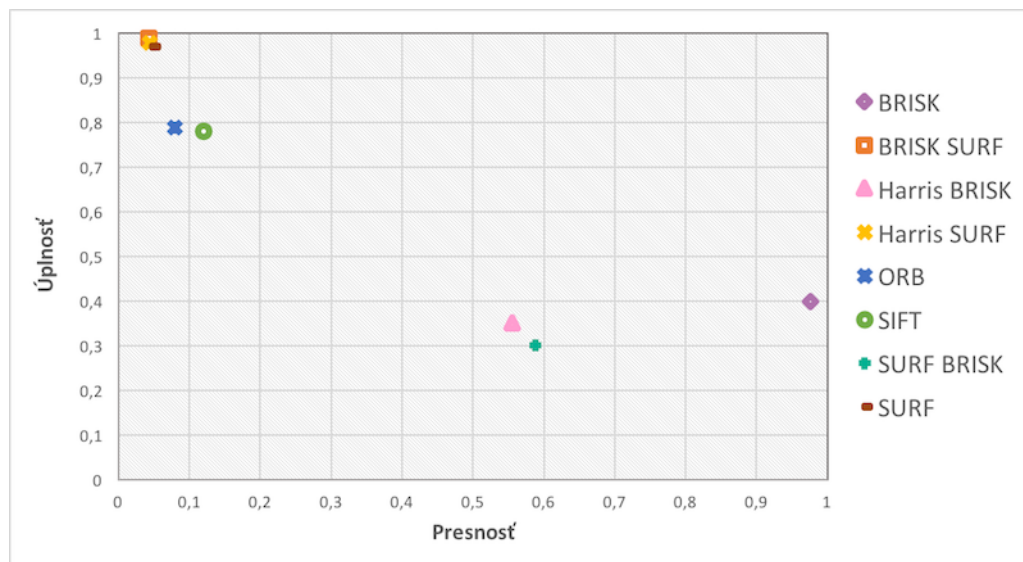
Obr. 3.7: Metóda BRISK - 0 zhodných deskriptorov (FN pre prah 5)

Pre všetky hľadané obrázky sme spracovali jednotlivé počty zhôd do hodnôt TP, TN, FP, FN (metodológia testovania) pre prah 5. To znamená, že za správnu zhodu je považované, ak

medzi dvoma obrázkami je najmenej 5 spárovaných deskriptorov. Keďže niektoré metódy dokážu vyprodukovať znateľne viac deskriptorov ako druhé, čo si môžeme všimnúť na výsledkoch v tabuľke 3.1, tak sme tomu museli prispôbiť prah pre správnu zhodu. Je väčšia šanca, že sa spárujú falošne správne deskriptory ak je na danom obrázku nájdených viac deskriptorov. Skúsili sme hodnoty pre prah 3, 5 aj 10 zhôd ale pre najrelevantnejšie hodnoty vzhľadom na všetky metódy sme sa rozhodli pre prah 5. Výsledky v percentách pre tento prah sú v tabuľke 3.2 a na obrázku 3.8 je znázornený graf pre pomer medzi hodnotami presnosti a úplnosti.

Tabuľka 3.2: Výsledky pre porovnávanie hľadaných objektov s databázou

	Presnosť	Úplnosť	Efektívnosť
BRISK	97,561%	40%	97,654%
BRISK SURF	4,304%	99%	15,308%
Harris BRISK	55,556%	35%	96,423%
Harris SURF	4,432%	98%	18,654%
ORB	7,948%	79%	64%
SIFT	11,963%	78%	77,077%
SURF BRISK	58,824%	30%	96,5%
SURF	4,792%	97%	25,769%



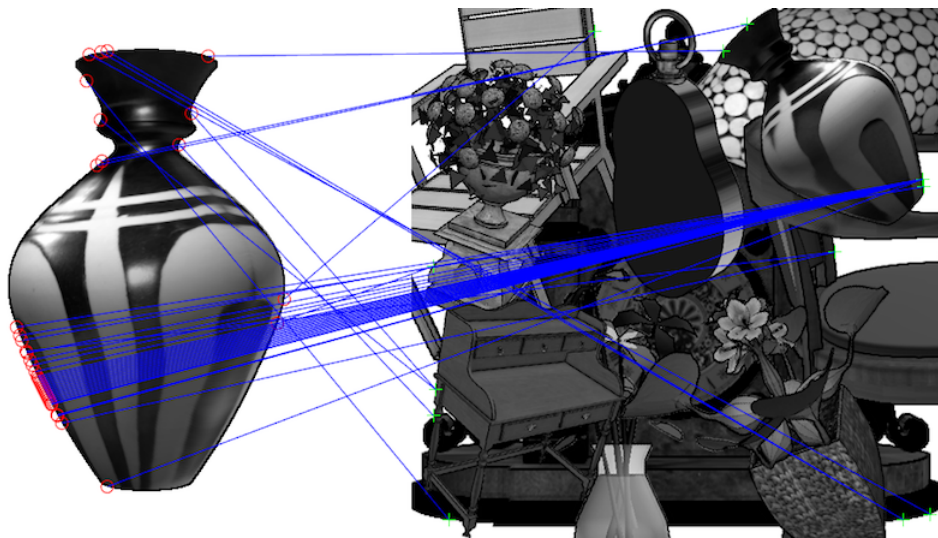
Obr. 3.8: Znázornenie výsledkov pre 2D metódy v grafe v pomere presnosť/úplnosť

3.4.3 Testovanie deskriptorov - koláže

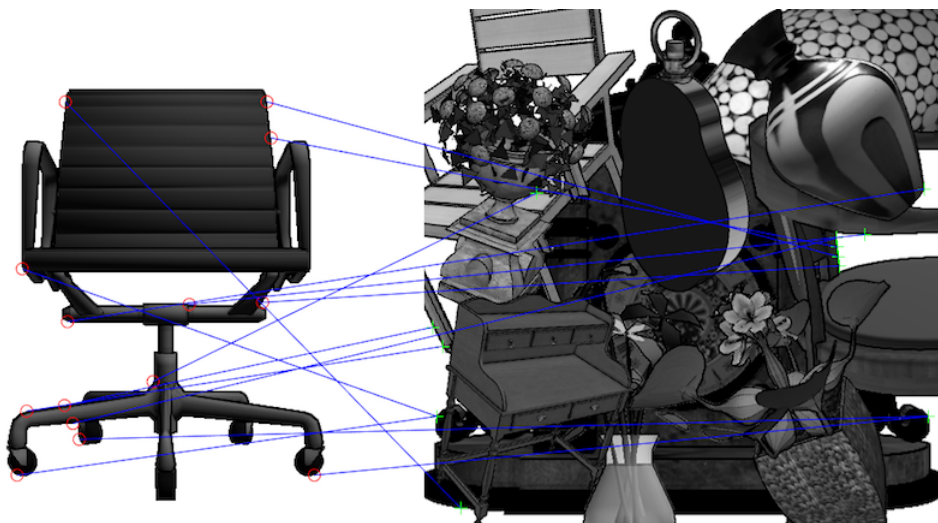
Na podobnom princípe prebiehalo aj tretie testovanie. Avšak tu sme sa zamerali na to, či sú metódy schopné nájsť hľadané objekty aj v takých obrázkoch, ktoré sú zložené z viacerých obrázkov objektov (naše koláže 3.3). Hľadané objekty môžu byť v týchto kolážach otočené,

zmenšené, zväčšené alebo inak transformované ako sme si už spomenuli v 3.2.1. Dokopy bolo 5 hlavných objektov (3.2), ktoré sme vyhľadávali v 15 kolážiach (3.3). Na kolážach sa nachádzali najmenej 2 objekty z hľadaných a najviac 5.

Na obrázkoch 3.9 a 3.10 je znázornené párovanie metódy Harris SURF medzi hľadanými objektami a kolážami. Na obrázku 3.9 sú zobrazené zhody medzi objektom a kolážou, na ktorej sa daný objekt nachádza. Niektoré body táto metóda spárovala nesprávne avšak väčšina bola správna a pre prah 5 možno toto párovanie správne uznať za zhodu. Na druhej strane v obrázku 3.10, kde sa porovnával objekt, ktorý sa v koláži nenachádza, sa našlo až 14 zlých zhôd a pre prah 5 je toto porovnávanie klasifikované ako nesprávna zhoda. Červeným kruhom sú označené zaujímavé body hľadaného objektu a zelený kríž je ich korešpondujúci spárovaný bod v koláži. Korešpondencia je medzi bodmi označená modrou čiarou.



Obr. 3.9: Metóda Harris SURF - 39 zhodných deskriptorov (TP pre prah 5)



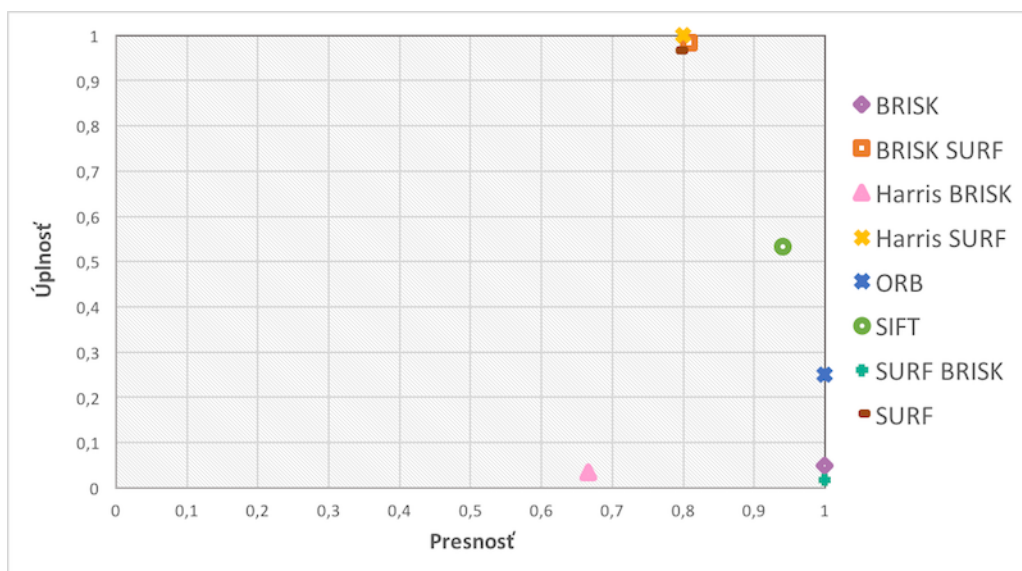
Obr. 3.10: Metóda Harris SURF - 14 zhodných deskriptorov (FP pre prah 5)

Na základe metodológie testovania sme pri tomto testovaní dospeli k výsledkom, ktoré sú

uvedené v tabuľke 3.3 a znázornili ich v grafe 3.11.

Tabuľka 3.3: Výsledky pre porovnanie hľadaných objektov s kolážami

	Presnosť	Úplnosť	Efektívnosť
BRISK	100%	5%	24%
BRISK SURF	80,822%	98,333%	80%
Harris BRISK	66,667%	3,333%	20,270%
Harris SURF	80%	100%	80%
ORB	100%	25%	38,356%
SIFT	94,118%	53,333%	58,333%
SURF BRISK	100%	1,667%	21,333%
SURF	79,452%	96,667	77,333%



Obr. 3.11: Znázornenie výsledkov pre 2D metódy v grafe v pomere presnosť/úplnosť

3.5 Analýza a porovnanie

3.5.1 2D metódy

Na základe nášho štúdia a prehľadu 2D metód v kapitole 2.1 sme vytvorili tabuľku 3.4 s hlavnými bodmi, ktoré sú charakteristické pre jednotlivé metódy a odvíja sa od nich výpočet.

Metódy SIFT a SURF sú predstavitelia spektrálnych deskriptorov, ktoré sme si spomenuli v prvej kapitole. Ich základným prvkom pri hľadaní zaujímavých bodov je vytvorenie škálovej pyramídy, ktorá sa vytvára s ohľadom na vstupný obraz. SIFT sa pri deskripcii zameriava na gradientové operácie, z ktorých vytvára histogram orientácie. Pomocou Gaussovej váhovej

funkcie sa z neho vypočíta deskriptor. Na druhej strane SURF je menej zložitý a na zistenie orientácie používa odozvu Haarovho waveletu.

Tabuľka 3.4: Analýza 2D metód

metóda	detekcia	deskriptor	deskripcia
SIFT	rozdiel Gaussiánov adaptívna škála škálová pyramída detekcia maxima a minima	spektrálny	gradientové operácie histogram orientácie Gaussová váhová funkcia
SURF	Fast-Hessian Detector Hesseho matice adaptívna škála škálová pyramída Gaussián druhej parciálnej derivácie	spektrálny	odozva Haarovho waveletu suma vertikálnych a horizontálnych odoziev suma absolútnych hodnôt odoziev
Harris	Moravecov detektor rohov Harrisova miera	/	/
BRISK	FAST 9-16 adaptívna škála škálovo priestorová pyramída	binárny	Gaussové vyhladzovanie lokálny gradient
ORB	oFAST Harrisov detektor rohov intenzita centroidu	binárny	rBRIEF bitový reťazec vzorky orientácia z ťažiska intenzity okolia hashovanie

V našom testovaní sme zistili, že na konkrétnom obrázku je metóda SURF (54ms) výrazne rýchlejšia ako metóda SIFT (911ms), čo môže byť následkom jednoduchšieho hľadania orientácie zaujímavých bodov a následnej tvorby deskriptorov. V počte detegovaných bodov a vytvorených deskriptorov sa líšia o vyše 250. Schopnosť SIFT-u nájsť a opísať príznaky je o cca polovicu lepšia ako u SURF-u ale na úkor rýchlosti. Preto je pri väčšej vzorke dát dobré použiť radšej SURF a SIFT použiť pri hľadaní príznakov v menšom súbore.

Ak sa zameriame na výsledky pri párovaní v databáze, tak hodnota pre presnosť vzhľadom na zvolený prah bola pre obe metódy veľmi malá (pod 15%). To môže byť následkom toho, že tieto metódy dokážu vytvoriť okolo 100000 deskriptorov pre 200 obrázkov. Pri takom množstve sa pri párovaní aj na obrázkoch, kde sa hľadaný objekt nenachádza, podarí spárovať viac deskriptorov ako náš prah a teda je daná dvojica obrázkov považovaná za falošnú zhodu. Vzhľadom na úplnosť ale dopadli lepšie a pri SURF-e bola citlivosť klasifikátora správnych zhôd takmer 100%. Efektívnosť oboch dosiahla nad 50%, čo je uspokojivý výsledok, keďže tieto dve metódy sú najpoužívanejšie.

Pri treťom testovaní sa takmer všetky hodnoty zmenili k lepšiemu a vďaka tomu môžeme

povedať, že pri párovaní objektov v scénach patria medzi najlepšie vzhľadom na všetky tri testované atribúty.

Na druhej strane máme metódy BRISK a ORB, ktoré patria medzi binárne deskriptory. Ich princíp je založený na porovnávaní párov pixelov. Výsledky sa následne ukladajú ako binárna hodnota vektora a sú efektívnejšie pre výpočet a ukladanie ako spektrálne deskriptory. BRISK vznikol ako variant pre SIFT a SURF a taktiež pracuje v škálovom priestore, kde na detekciu kľúčových bodov používa metódu FAST. ORB taktiež používa metódu FAST avšak na usporiadanie kľúčových bodov používa Harrisov detektor rohov. Deskriptory oboch sú na binárnom princípe. BRISK skladá binárny reťazec na základe porovnávania jasů. ORB ako deskriptor používa rBRIEF, ktorý vytvára binárny reťazec testovaním binárnej intenzity.

Vzhľadom na výsledky časovej náročnosti BRISK dokázal vytvoriť 619 deskriptorov čím sa zaradil medzi SIFT a SURF a z časového hľadiska sa tiež zaradil skôr medzi priemer. Zato ORB dosiahol vynikajúce výsledky a to s cca 3515 deskriptormi za 33ms. Medzi našimi testovanými metódami je metóda ORB najrýchlejšia.

V testoch pri párovaní v databáze dosiahol BRISK najlepšie výsledky pre presnosť a efektívnosť zo všetkých testovaných metód s takmer 100%. Podarilo sa mu najlepšie určiť správne zhody a taktiež určiť správne negatívne zhody. Avšak schopnosť vytvoriť zhodu bola len 40%. ORB na tomto type testovania pochodil výrazne horšie ako BRISK a to s presnosťou 8%. Efektívnosť tiež nebola vysoká vzhľadom na ostatné a citlivosť dosiahla 79%.

Hľadanie zhôd medzi objektami a kolážami výrazne ovplyvnilo výsledky pre úplnosť. BRISK bol schopný vytvárať zhody už len na 5% a ORB na 25% síce ich presnosť dosiahla 100%. Efektívnosť sa pohybovala pod 40%, čo im tiež výrazne nepomohlo. Z toho vyplýva, že síce ORB je výrazne rýchlejší ako BRISK ale pri párovaní dosahuje lepšie výsledky BRISK.

Metóda Harrisov detektor rohov obsahuje iba detektor, ktorý treba skombinovať s deskriptorom inej metódy. Harris sa inšpiroval Moravecovým detektorom rohov a doplnil spracovanie o výpočet pomocou Harrisovej miery.

Harrisov detektor sme pri našom testovaní skombinovali s deskriptormi metód SURF a BRISK. Z pohľadu časovej náročnosti kombinácia Harris a SURF bola trošku pomalšia ako oficiálna metóda SURF a to iba o 5ms a stále ju môžeme zaradiť medzi rýchlejšie metódy.

Kombinácia Harris BRISK bola časovo náročná priemerne.

Kombinácia Harris BRISK pre zhody v databáze neprekonali základnú metódu BRISK, dokonca sa hodnoty pre presnosť a úplnosť zhoršili. Pri kombinácii Harris SURF zostali hodnoty takmer rovnaké ako u samotného SURF-u.

V testovaní zhôd medzi kolážami Harris BRISK znova zhoršil schopnosti BRISK-u avšak Harris SURF dosiahol ešte lepšie výsledky ako samotný SURF. Síce zmena v hodnotách nie je výrazná ale s dosiahnutými výsledkami je táto metóda najlepšia z testovaných metód s citlivosťou 100% a presnosťou a efektívnosťou 80%.

Poslednými dvoma testovanými metódami bol experiment s kombináciou detektora SURF a deskriptora BRISK a naopak. Deskriptor BRISK spomalil výpočet a kombinácia SURF BRISK bola pomalšia ako samotný SURF a naopak deskriptor SURF zrýchlil kombinovanú metódu oproti základnej. Počty deskriptorov sa oproti hlavným metódam takmer nezmenili.

Testovanie v databáze na kombinovaných metódach ukázalo vplyv zmeny. BRISK SURF oproti BRISK-u samotnému zvýšilo svoju úplnosť na 99% ale výrazne sa znížila presnosť a efektívnosť. Pre SURF BRISK to dopadlo práve naopak oproti SURF-u. Presnosť a efektívnosť sa zvýšila a úplnosť padla na 30%.

Na párovaní koláží sa však hodnoty zmenili inak ako sa predpokladalo po testovaní na databáze. Kombinácia BRISK SURF dosiahla takmer také výsledky ako Harris SURF a teda je druhou najlepšou kombináciou metód na párovanie v scénach. SURF BRISK sa spresnil na 100% ale úplnosť a efektívnosť bola nedostatočná.

Na základe poznatkov a výsledkov z nášho testovania sme dospeli k záveru, že každá z nami testovaných metód je najlepšia pre určitú oblasť rozpoznávania a teda je dobré vybrať si takú metódu, ktorá bude zodpovedať budúcemu použitiu. Ak treba použiť metódu, ktorá je najmenej náročná čas a po upravení prahu správnej zhody dosahuje relevantné výsledky, tak je najlepšie použiť metódu ORB. Ak čas nie je podstatný a porovnávajú sa len čisto objekty bez scény, tak je podľa našich zistení najlepším kompromisom metóda BRISK. Ak treba porovnávať v scénach plných rôznych objektov, tak najlepšie obstála kombinovaná metóda detektor Harris s deskriptorom SURF.

3.5.2 3D metódy

Pre 3D metódy sme rovnako ako pre 2D vytvorili tabuľku 3.5, v ktorej sme zhrnuli hlavné kroky pri detekcii a deskripcii jednotlivých metód a ich reprezentáciu 3D dát, s ktorými pracovali.

Tabuľka 3.5: Analýza 3D metód

metóda	3D dáta	detekcia	deskripcia
LD SIFT	sieť	hustotou nemenný Gaussový filter rozdiel Gaussiánov odhad lokálnej škály	mapa hĺbky gradientový histogram orientácie deskriptor podobný ako SIFT
3D SURF	sieť	voxelizácia siete miera význačných rysov adaptívna škála	odozva Haarovho waveletu lokálny rámec 162-rozmerný deskriptorový vektor
3D Harris	sieť	body transformované na centroid kontinuálna Gaussiánová funkcia Harrisova miera	/
SHOT	sieť	/	lokálny referenčný rámec 3D izotropická sférická mriežka lokálne histogramy 352-rozmerný deskriptorový vektor
MeshDOG MeshHOG	sieť	rozdiel Gaussiánov Hesseho operátor adaptívna škála	gradientový operátor gradientový histogram gradientové vektory

3D metódy sme mali v pláne prakticky otestovať podobne ako 2D avšak z technických príčin sa nám to nepodarilo a spravili sme porovnanie z teoretického hľadiska s pomocou článku vytvoreného Yulanom Guom a kolektívom [5].

Nami vybrané metódy, ktoré sme si už v predchádzajúcich kapitolách predstavili, pracujú zo vstupnými dátami vo forme 3D sietí. Tie sú často využívané v grafických aplikáciách a rôznych simuláciách v 3D priestore.

Detekcia na 3D sieťach prebieha na princípe tvorby škálového priestoru podobne ako aj pri 2D metódach. 3D Harris si taktiež ako jeho 2D verzia vytvára fixnú škálu, v ktorej si definuje fixný bod, ktorý je charakteristický pre vopred určené susedstvo, a ten sa vyberie ako kľúčový. Veľkosť susedstva je definovaná škálou. Aplikuje sa kontinuálna Gaussiánová funkcia a výpočtom Harrisovej miery sa zistia lokálne maximá. Vďaka tomu, že je škála fixná, tak je implementácia detekcie jednoduchá. Na druhej strane sa kvôli tomu môže stať, že metóda kvôli fixnej škále nájde veľmi málo kľúčových bodov a nedostatočne opíše daný

objekt, čo by mohlo mať za následok problém pri jeho rozpoznávaní. Preto je pri presnejšom hľadaní zaujímavých bodov a ich následnom párovaní lepšie použiť metódy, ktoré fungujú na adaptívnej škále.

Detektory LD SIFT, 3D SURF a MeshDOG vytvárajú škálu prispôsobenú k vstupnému objektu. Následne sa vyberú také body, ktoré sú najcharakteristickejšie v priestore aj škále susedstva a určia sa za zaujímavé body. Metódy sa však líšia pri postupe vytvárania škálového priestoru.

LD SIFT a 3D SURF sú obe založené na ich 2D predchodcoch, ktorými sa inšpirovali. LD SIFT taktiež ako 2D SIFT pracuje na oktávach. Avšak oproti 2D sa aplikuje hustotou nemenný Gaussový filter. Siete sú vďaka tomu sú odolnejšie voči rôznym mierkam a nerovnomernému vzorkovaniu. Následne sa ako v 2D použije rozdiel Gaussiánov na nájdenie lokálnych extrémov. V 3D SURF-e sa rozhodli pre iný postup a to prevedenie vstupného objektu na voxely, kde sa pre každý voxel vypočítava Gaussián parciálnej derivácie. Škála sa rovnako ako u 2D SURF-u vytvorí na základe Hesseho matice.

MeshDOG vytvára škálu iným spôsobom a to postupným vyhladzovaním geometrických atribútov rozsahu obrazu. Filtrovanie sa aplikuje najskôr na geometrické atribúty až potom na samotný rozsah, čiže sa nevykoná žiadna zmena vonkajšej geometrie 3D tvaru a zachová sa súvislosť vlastností daného priestoru. V MeshDOG je najskôr definovaná skalárna oblasť pre každý bod a skalárna oblasť s množinou Gaussiánových jadri. Následnými výpočtami rozdielu Gaussiánov sa vytvoril škálový priestor a určia sa lokálne maximá. Vypočítaním Hesseho operátora sa odstránia nežiadúce rohy a zostane už len dostatočný počet opakovateľných kľúčových bodov. Avšak tento výpočet pri hľadaní kľúčových bodov je citlivý na zmeny v rozlíšení siete.

Metódy, ktoré pracujú s adaptívnou škálou sú schopné rozpoznať prirodzenú škálu priradenú ku kľúčovému bodu. Táto schopnosť zlepšuje výkon pri deskripcii príznakov a následnom rozpoznávaní objektov.

Akonáhle sú nájdené kľúčové body prichádza na rad extrakcia geometrických vlastností lokálneho povrchu okolo bodu a ich zakódovanie do príznakového deskriptora. Metódy LD SIFT, SHOT a MeshHOG, pri zostavovaní príznakového deskriptora používajú histogramy.

Metóda 3D SURF je ako jediná založená na transformácií, čo znamená, že sa najprv transformuje obraz rozsahu z priestorovej oblasti na inú. Následne sa opisuje 3D oblasť

susedstva daného bodu kódovaním informácií už v transformovanej oblasti. Ako sme si spomenuli už skôr, tak pri 3D SURF-e sa obraz transformuje voxelizáciou. Na už detegovaných kľúčových bodoch sa vypočíta odozva Haarovho waveletu a aplikuje sa Gausiánové centrovanie. Vytvorí sa lokálny referenčný rámec okolo každého bodu a na základe odoziev sa vytvorí deskriptorový vektor.

Metódy, ktoré sú založené na histogramoch, do nich opisujú lokálne susedstvo okolo kľúčových bodov zhromaždením geometrických a typologických rozmerov podľa konkrétnej oblasti. LD SIFT a MeshHOG pracujú na báze histogramov gradientu. Vytvárajú sa opisom lokálneho susedstva okolo kľúčového bodu generovaním histogramov podľa orientovaných gradientov lokálneho povrchu. Deskriptor LD SIFT je vypočítaný reprezentovaním lokálneho susedstva v podpornej oblasti definovanej škálou ako hĺbkovú mapu. Vypočítaný deskriptor je podobný deskriptoru u 2D SIFT-u, ktorý je jeho 2D predchodcom. MeshHOG vytvoril gradientový histogram na základe gradientov zo skalárnej funkcie, ktorá bola vypočítaná pri detekcii. Vytvoril sa lokálny referenčný rámec okolo každého kľúčového bodu a na jeho tri roviny sa premietli gradientové vektory. Výsledný deskriptor sa získal spojením histogramov všetkých podoblastí. Takýto deskriptor je efektívny na stabilných aj nestabilných objektoch, ktoré však nepodliehajú veľkým deformáciám.

Posledná z našich skúmaných metód je SHOT, ktorá obsahuje len deskriptor. Ten je taktiež založený na báze histogramov. Avšak narozdiel od predchádzajúcich metód, SHOT do histogramu opisuje lokálne okolie kľúčového bodu podľa geometrických atribútov lokálneho povrchu. Pri deskripcii si najprv vytvorí lokálny referenčný rámec pre každý kľúčový bod a následne sa skonštruuje aj 3D sférická mriežka. Pre každú dimenziu v mriežke sa vytvorí lokálny histogram tým, že sa zhromaždí počet bodov na základe uhlov medzi normálou v kľúčovom bode a susednými bodmi. Následne sa spoja všetky lokálne histogramy a z nich sa vytvorí príznakový deskriptor. Takýto deskriptor má veľmi dobrú popisnú schopnosť, je efektívny pohľadu výpočtu a odolný voči šumu. Jeho nedostatkom však je citlivosť na rôznu hustotu bodov.

Metódy založené na histogramoch sú používané najčastejšie a aj nami vybrané metódy sú väčšinou tohto typu. Ich výpočty závisia od výpočtu derivácií povrchu prvého alebo druhého rádu a to má za následok, že tieto metódy sú citlivejšie na šum. Avšak pri SHOT-e sa tento nedostatok pokúsili napraviť a ich deskriptor by mal byť voči šumu odolný. Tiež ako sme si mohli všimnúť, tak vo väčšine metód sa vytváral lokálny referenčný rámec, vďaka čomu sú príznaky invariantné voči transformáciám.

Výber najlepšej zo spomenutých metód závisí na type vstupných dát. 3D Harris je vďaka fixnej škále rýchlejší ako ostatné spomínané metódy ale oproti ostatným nie je taký presný. LD SIFT a 3D SURF sú zase viac odolné na nerovnomerných dátach. MeshDOG a MeshHOG je výkonnejší avšak je citlivý na zmeny v sieti. SHOT je dostatočne efektívny a odolný voči šumu avšak problémom pre neho znamenajú nerovnomerné objekty. Preto pri práci s lokálnymi príznakmi treba brať ohľad na predpokladaný vzhl'ad 3D dát a podľa toho si vybrať najlepšiu vhodnú metódu.

4. Záver

Cieľom bakalárskej práce bolo vytvoriť prehľad 2D a 3D metód na detekciu a deskripciu lokálnych príznakov, naštudovať si ich a navzájom porovnať.

V práci sme sa rozhodli vybrať si 5 2D a 5 3D metód, ktoré sú v počítačovom videní najznámejšie a najpoužívanejšie. Pri hľadaní vhodných metód sme sa taktiež snažili nájsť také, ku ktorým je prístupná dokumentácia a zdrojový kód. Pre to sme si ako 2D metódy vybrali SIFT, SURF, Harrisov detektor rohov, BRISK a ORB. Tieto metódy sme si naštudovali z oficiálnych článkov od ich tvorcov a popísali ich postup pri hľadaní a opise lokálnych príznakov. Následne sme si ako 3D metódy, chceli vybrať také, ktoré boli založené na 2D predchodcoch už nami vybraných metód v predchádzajúcom štúdiu. To sa nám aj podarilo v prípade 3D Harrisa, 3D SURF-u a LD SIFT-u. Avšak v prípade SIFT-u sme pôvodne plánovali použiť metódu 3D SIFT, lenže sme sa nedostali k jej dokumentácií, pretože bola platená, a preto sme si radšej zvolili metódu LD SIFT. Tieto tri metódy sme ešte doplnili dvoma inými metódami a to deskriptorom SHOT a metódou MeshDOG a MeshHOG. Podobne ako u 2D sme si ich naštudovali a spísali ich postupy.

Pre účely porovnania sme sa rozhodli otestovať vybrané metódy na našich dátach. Vybrali sme si preto databázu 3D objektov, z ktorej sme si vytvorili vlastnú testovaciu databázu pre 2D metódy. Vybrali sme si metodológiu testovania testovali sme v troch fázach. Všetky testy sme si sami naprogramovali a aplikovali v programovacom jazyku MATLAB. Prvá fáza testovania pozostávala z testu časovej zložitosti a schopnosti vytvárať deskripty na súbore 200 obrázkov z našej vybranej databázy. Druhá fáza vychádzala z metodológie testovania deskriptorov, kde sme zistovali hodnoty pre presnosť, úplnosť a efektívnosť metód pri párovaní obrázkov. Pre toto testovanie sme si vybrali 10 objektov, ktoré sme následne porovnávali s databázou 260 obrázkov, ktorá obsahovala aj dané objekty a to konkrétne v počte na 10 obrázkoch. Tieto obrázky obsahovali daný objekt na bielom pozadí a boli zaznamenané z rôznych uhlov pohľadu. Následne prebiehala tretia fáza, ktorá bola podobne založená na párovaní, avšak v tomto prípade sme párovali objekty s kolážami. Vybrali sme si konkrétne 5 objektov, ktoré sme zakomponovali do 15 rôznych koláží. V kolážach sme tieto objekty vždy nejak pozmenili a to napríklad zmenšením, pootočením alebo pohľadom z iného uhlu. Koláže vždy obsahovali minimálne 2 z 5 vybraných objektov, ktoré sme potom

v nich vyhl'adávali. V týchto testoch sme používali oficiálne metódy ale taktiež sme skúsili skombinovať detektor z jednej metódy s deskriptorom druhej. Výsledky týchto testov sme vyhodnotili a na ich základe sme metódy porovnali.

Pri testovaní bolo najt'ažšie sfunkčniť vybrané metódy, pretože napríklad pri metóde ORB bolo treba doinštalovať knižnicu OpenCV, ktorá však je pre C++ a teda bolo treba nájsť vhodný a funkčný spôsob ako túto knižnicu do MATLAB-u dostať. Druhou komplikáciou bolo pochopenie rôzneho výpočtu a ukladania rovnakých dát odlišnými spôsobmi, čo bol problém najmä pri SIFT-e, keďže sme mali prístupnú len demo verziu. Výpočet v nej prebiehal v samospustiteľnom súbore a preto sme sa nemohli dostať k niektorým dátam, ktoré sme testovali.

Podobné testovanie ako u 2D metód sme plánovali uskutočniť aj na 3D metódach, lenže tie sa nám nepodarilo sfunkčniť vôbec. Preto sme sa rozhodli spraviť pre 3D metódy len teoretickú analýzu a porovnanie bez uskutočnených testov.

Preto by bolo vhodné pri budúcej práci zamerať sa len na 3D metódy, ktorým by bol venovaný celý čas a primerane by sa otestovali a porovnali podobným postupom ako v tejto práci.

Zoznam použitej literatúry

- [1] BAY, Herbert - TUYTELAARS, Tinne - VAN GOOL, Luc, „Surf: speeded up robust features“, *Computer vision–ECCV 2006*, s. 404–417, 2006.
- [2] BRADSKI, Gary - KAEHLER, Adrian, „Opencv“, *Dr. Dobb's journal of software tools*, vol. 3, 2000.
- [3] CALONDER, Michael - LEPETIT, Vincent - STRECHA, Christoph - FUA, Pascal, „Brief: binary robust independent elementary features“, *Computer Vision–ECCV 2010*, s. 778–792, 2010.
- [4] DAROM, Tal - KELLER, Yosi, „Scale-invariant features for 3-d mesh models“, *IEEE Transactions on Image Processing*, vol. 21, no. 5, s. 2758–2769, 2012.
- [5] GUO, Yulan - BENNAMOUN, Mohammed - SOHEL, Ferdous - LU, Min - WAN, Jianwei, „3d object recognition in cluttered scenes with local surface features: a survey“, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, s. 2270–2287, 2014.
- [6] HARRIS, Chris - STEPHENS, Mike, „A combined corner and edge detector.“, in *Alvey vision conference*, Manchester, UK, vol. 15 : 1988. S. 10–5244.
- [7] HEINLY, Jared - DUNN, Enrique - FRAHM, Jan-Michael, „Comparative evaluation of binary features“, in *Computer Vision–ECCV 2012* : Springer, 2012. S. 759–773.
- [8] CHANG, Angel X - FUNKHOUSER, Thomas - GUIBAS, Leonidas - HANRAHAN, Pat - HUANG, Qixing - LI, Zimo - SAVARESE, Silvio - SAVVA, Manolis - SONG, Shuran - SU, Hao, „Shapenet: an information-rich 3d model repository“, *arXiv preprint arXiv:1512.03012*, 2015.
- [9] KNOPP, Jan - PRASAD, Mukta - WILLEMS, Geert - TIMOFTE, Radu - VAN GOOL, Luc, „Hough transform and 3d surf for robust three dimensional classification“, *Computer vision–ECCV 2010*, s. 589–602, 2010.
- [10] KRIG, Scott 2016. *Computer Vision Metrics: Textbook Edition* : Springer, 2016. .

- [11] LEUTENEGGER, Stefan - CHLI, Margarita - SIEGWART, Roland Y, „Brisk: binary robust invariant scalable keypoints“, in *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE : 2011. S. 2548–2555.
- [12] LISIN, Dimitri A - MATTAR, Marwan A - BLASCHKO, Matthew B - LEARNED-MILLER, Erik G - BENFIELD, Mark C, „Combining local and global image features for object class recognition“, s. 47–47, 2005.
- [13] LOWE, David G, „Distinctive image features from scale-invariant keypoints“, *International journal of computer vision*, vol. 60, no. 2, s. 91–110, 2004.
- [14] MORAVEC, Hans P, „Obstacle avoidance and navigation in the real world by a seeing robot rover.“, STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, Tech. Rep., 1980.
- [15] PRAKHYA, Sai Manoj - LIU, Bingbing - LIN, Weisi, „B-shot: a binary feature descriptor for fast and efficient keypoint matching on 3d point clouds“, in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, IEEE : 2015. S. 1929–1934.
- [16] PRATIKAKIS, I - SPAGNUOLO, M - THEOHARIS, T - VELTKAMP, R, „A robust 3d interest points detector based on harris operator“, in *Eurographics workshop on 3D object retrieval*, vol. 5 : 2010.
- [17] ROSTEN, Edward - DRUMMOND, Tom, „Machine learning for high-speed corner detection“, *Computer Vision–ECCV 2006*, s. 430–443, 2006.
- [18] RUBLEE, Ethan - RABAUD, Vincent - KONOLIGE, Kurt - BRADSKI, Gary, „Orb: an efficient alternative to sift or surf“, in *Computer Vision (ICCV), 2011 IEEE international conference on*, IEEE : 2011. S. 2564–2571.
- [19] SALTI, Samuele - TOMBARI, Federico - DI STEFANO, Luigi, „Shot: unique signatures of histograms for surface and texture description“, *Computer Vision and Image Understanding*, vol. 125, s. 251–264, 2014.
- [20] ŠIKUDOVÁ, Elena - ČERNEKOVÁ, Zuzana - BENEŠOVÁ, Wanda - HALADOVÁ, Zuzana - KUČEROVÁ, Júlia 2013. *Počítačové videnie Detekcia a rozpoznávanie objektov*. Praha : Wikina, Praha, 2013. P. 397 . ISBN: 978-80-87925-06-5.
- [21] YAMAGUCHI, K, *Mexopencv-collection of mex functions for opencv library*, 2012.
- [22] ZAHARESCU, Andrei - BOYER, Edmond - VARANASI, Kiran - HORAUD, Radu, „Surface feature detection and description with applications to mesh matching“, in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE : 2009. S. 373–380.

Prílohy

CD obsahujúce:

- Elektronickú verziu bakalárskej práce
- Zdrojové súbory použitých metód
- Databázu použitú na testovanie
- Výsledky testovania