Detekce a segmentace 3D objektu v obraze

Daniel Vaško

Bakalářská práce
2024

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2023/2024

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	**Daniel Vaško**
Osobní číslo:	**A21213**
Studijní program:	**B0613A140020 Softwarové inženýrství**
Forma studia:	**Prezenční**
Téma práce:	**Detekce a segmentace 3D objektů v obraze**
Téma práce anglicky:	**Detection and Segmentation of 3D Objects in Images**

Zásady pro vypracování

1. Proveďte literární rešerši metod v oblasti detekce 3D objektů v obraze.
2. Proveďte literární rešerši metod v oblasti segmentace 3D objektů v obraze.
3. Otestujte dostupné kódy metod z provedené rešerše na zvoleném testovacím datasetu.
4. Srovnejte dosažené výsledky.
5. Proveďte doporučení a závěr.

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. BIASUTTI, Pierre, et al. Lu-net: An efficient network for 3d lidar point cloud semantic segmentation based on end--to-end-learned 3d features and u-net. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 2019. p. 0-0.
2. WANG, Yue, et al. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In: *Conference on Robot Learning*. PMLR, 2022. p. 180-191.
3. DENG, Zhuo; JAN LATECKI, Longin. Amodal detection of 3d objects: Inferring 3d bounding boxes from 2d ones in rgb--depth images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017. p. 5762-5770.
4. SHI, Shaoshuai; WANG, Xiaogang; LI, Hongsheng. Pointrcnn: 3d object proposal generation and detection from point cloud. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019. p. 770-779.
5. SAMET, Nermin, et al. You Never Get a Second Chance To Make a Good First Impression: Seeding Active Learning for 3D Semantic Segmentation. *arXiv preprint arXiv:2304.11762*, 2023.

Vedoucí bakalářské práce: **prof. Ing. Zuzana Komínková Oplatková, Ph.D.**
Ústav informatiky a umělé inteligence

Datum zadání bakalářské práce: **5. listopadu 2023**
Termín odevzdání bakalářské práce: **13. května 2024**

UNIVERZITA TOMÁŠE BATI VE ZLÍNĚ
Fakulta aplikované informatiky
DĚKAN
L.S.

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 5. ledna 2024

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

Daniel Vaško v.r.
podpis studenta

ABSTRAKT

Táto bakalárska práca sa zameriava na detekciu a segmentáciu trojrozmerných objektov v obrázkoch, čo je kľúčová oblasť výskumu v dynamicky sa rozvíjajúcom poli počítačového videnia. Práca preskúmava a analyzuje existujúce metodiky pre identifikáciu a vymedzenie objektov v trojrozmernom priestore z dvojrozmerných obrazov. Tieto techniky sú nevyhnutné pre široké spektrum aplikácií, vrátane autonómnych vozidiel a robotických systémov. Cieľom práce je poskytnúť pohľad na aktuálne používané prístupy v detekcii a segmentácii, ako aj diskutovať o ich význame v kontexte počítačového videnia a ich aplikáciách.

Klíčová slova: počítačové videnie, 3D objekty, detekcia objektov, segmentácia, hlboké učenie, konvolučné neuronové siete

ABSTRACT

This bachelor thesis focuses on the detection and segmentation of three-dimensional objects in images, a key research area in the dynamically developing field of computer vision. The thesis reviews and analyzes existing methodologies for identifying and de-mystifying objects in three-dimensional space from two-dimensional images. These techniques are essential for a wide range of applications, including autonomous vehicles and robotic systems. The aim of this thesis is to provide insight into currently used approaches in detection and segmentation, as well as to discuss their importance in the context of computer vision and their applications.

Keywords: computer vision, 3D objects, object detection, segmentation, deep learning, convolutional neural networks

Týmto chcem poďakovať svojej školiteľke prof. Ing. Zuzane Komínkovej Oplatkovej, Ph.D. za vedenie pri mojej práci a užitočné rady pri písaní bakalárskej práce a flexibilitu pri zodpovedaní dotazov s prácou spojených. Taktiež chcem poďakovať rodine a kamarátom za podporu popri štúdiu.

Prohlašuji že při tvorbě této práce jsem použil nástroj generativního modelu AI [https://chatgpt.com/] za účelem generování nápadů, zlepšení struktury textu, rešerše akademických materiálů a sumarizace obsahu. Po použití tohoto nástroje jsem provedl/a kontrolu obsahu a přebírám za něj plnou zodpovědnost.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD

V rýchlo sa rozvíjajúcej oblasti počítačového videnia sa detekcia a segmentácia trojrozmerných objektov v obrazoch ukazuje ako kritická oblasť výskumu, ktorá odráža širší trend smerujúci k pochopeniu trojrozmerného sveta a interakcii s ním prostredníctvom digitálnych obrazov. Táto práca sa zaoberá skúmaním existujúcich metodík na detekciu a segmentáciu 3D objektov, čo je snaha, ktorá je nielen akademicky obohacujúca, ale aj nesmierne dôležitá v dnešnej technologickej dobe. Schopnosť presne identifikovať a vymedziť objekty v trojrozmernom priestore z dvojrozmerných obrazov je základom mnohých aplikácií, počnúc autonómnymi vozidlami navigujúcimi v zložitých prostrediach, robotickými systémami vykonávajúcimi presné manipulácie, až po lekárske zobrazovanie ponúkajúce život zachraňujúcu diagnostiku a pohlcujúce zážitky vytvorené rozšírenou a virtuálnou realitou. Svojím spôsobom by sa dala táto doba charakterizovať aj ako éra 3D videnia. Nástup hlbokého učenia a výrazný pokrok vo výpočtových možnostiach posunul oblasť detekcie a segmentácie 3D objektov do popredia výskumu počítačového videnia. Tento vývoj otvoril nové možnosti riešenia vnútorných výziev spojených s interpretáciou 3D štruktúr z 2D obrazov, ako je napríklad riešenie kolízií, rôznorodého vzhľadu objektov a obrovskej rozmanitosti kontextov, v ktorých sa objekty nachádzajú. Keďže stojíme na pokraji technologickej revolúcie, v ktorej sa od strojov čoraz viac očakáva, že budú rozumieť fyzickému svetu a komunikovať s ním podobne ako ľudia, nemožno preceňovať význam pokroku v technológiách detekcie a segmentácie 3D objektov. Je to trend, ktorý vystihuje snahu umelej inteligencie vnímať hĺbku a dimenzionalitu, čo strojom umožňuje chápať svet komplexnejším a diferencovanejším spôsobom.

I. TEORETICKÁ ČASŤ

1 DETEKCIA OBJEKTU V OBRAZE

Detekcia objektov v obrázkoch je technika počítačového videnia, ktorá sa používa na identifikáciu a lokalizáciu objektov na digitálnych obrázkoch alebo snímkach videa. Tento proces zahŕňa rozpoznávanie prípadov špecifických kategórií (ako sú ľudia, budovy, autá alebo nábytok) vo vizuálnom obsahu.

Proces detekcie objektov na obrázkoch sa výrazne vyvinul s príchodom techník hlbokého učenia. Preto je v tejto práci fungovanie týchto techník zjednodušene opísane a priblížené. Detekcia objektu v obraze sa vykonáva, najmä pomocou konvolučných neurónových sietí (CNN), ktoré v súčasnosti patria medzi najefektívnejšie metódy na túto úlohu. Priebeh spracovania obrázku je vysvetlený v nasledujúcich krokoch.

1.1 Kroky pri spracovaní 2D obrázku

Prvý krok zahŕňa prípravu obrazu na spracovanie. To môže zahŕňať zmenu veľkosti obrázku na pevnú veľkosť, normalizáciu hodnôt pixelov a prípadne rozšírenie údajov (napr. prevrátením alebo otočením obrazu) s cieľom zlepšiť robustnosť detekčného modelu.[1]

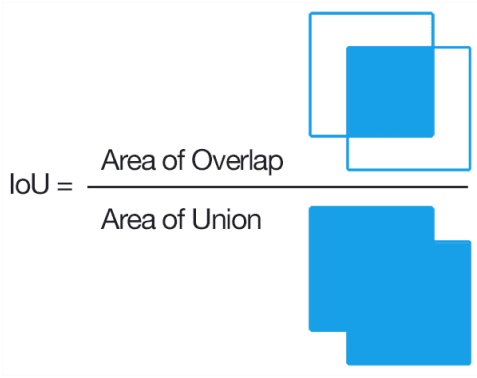
V druhom kroku prichádza na rad extrakcia príznakov. Pomocou CNN sa model automaticky naučí identifikovať relevantné vlastnosti z trénovaných údajov. CNN pozostáva z viacerých vrstiev konvolúcií (z angl. convolution layers) a vrstiev združovania (z angl. pooling layers), ktoré pomáhajú sieti identifikovať rôzne vlastnosti v obrazoch, od jednoduchých hrán a textúr v prvých vrstvách až po komplexné objekty v hlbších vrstvách. Vrstvy konvolucií zjednodušene vytvoria malé regióny posunom po obrázku generujúce K počet výstupov (kanálov). Vytvoria tak ďalší obrázok s rôznymi šírkami, výškami aj hĺbkami. Tieto vrstvy transformujú regióny so spoločnými príznakmi na jednotlivé výstupné hodnoty, pričom zachovávajú významné vlastnosti. Môže sa jednať o maximálne združovanie (Max Pooling) alebo priemerné združovanie(Average pooling), pričom maximálne združovanie vyberá najvyššiu hodnotu daného regiónu. To zviditeľní významne vlastnosti tohto regiónu, naopak priemerné združovanie na priemernú hodnotu z tohto regiónu [1], [2], [3].

Tretí krok zahŕňa klasifikáciu a lokalizáciu. Modely na detekciu objektov zvyčajne produkujú štítky tried (čo je objekt) a ohraničujúce polia (kde sa objekt v obraze nachádza). Existujú dva hlavné typy detekčných modelov:

- Dvojstupňové detektory, ako napríklad R-CNN a jej rýchlejšie varianty (Fast R-CNN, Faster R-CNN), na začiatku vytvárajú návrhy oblastí, ktoré by potenciálne mohli obsahovať objekty, a potom tieto oblasti klasifikujú a spresňujú ich ohraničujúce súradnice. [1]
- Jednostupňové detektory, ako napríklad YOLO (You Only Look Once) a SSD (Single Shot MultiBox Detector), vynechajú krok generovania návrhov a priamo predikujú triedy a umiestnenie objektov v jednom kroku, vďaka čomu sú rýchlejším riešením a sú vhodné pre aplikácie v reálnom čase. [1]

Štvrtým krokom je nemaximálna supresia (NMS - non-maximum Suppresion). Vzhľadom na to, že okolo toho istého objektu sa môže predpovedať viac ohraničujúcich boxov, na vyriešenie týchto prekrývaní sa použije nemaximálna supresia. Nemaximálna supresia zachová pre každý objekt len ohraničujúci box s najvyšším bodovým hodnotením dôveryhodnosti. (Obrázok 1.)Dôveryhodnosť a kritéria výberu si môžeme vybrať sami. Zvyčajne je týmto kritériom miera prekrytia (IoU - Intersection over union).Táto hodnota vznikne ako podiel dvoch hodnôt. Prvou je prercentuálna hodnota prieniku dvoch boxov a druhou je percetualna hodnota zjednotenia týchto boxov (Obrázok 2.).[1] [4]

Obrázok 1. – Aplikovanie NMS), prevzaté z [4]



Obrázok 2 – Ukážka IoU), prevzaté z [5]

Záverečný piaty krok zahŕňa post-processing, interpretáciu výstupu modelu, pričom zahŕňa filtrovanie detekcií na základe hodnôt dôveryhodnosti, prípadné uplatnenie ďalších obmedzení a prípravu výsledkov na ďalšie spracovanie alebo zobrazenie. [1]

1.2 Poloha 3D objektu

Pri spracovávaní a detekcií 3D objektu v obraze musíme brať do úvahy už vyššie spomínané vlastnosti ako veľkosť, poloha či orientácia objektu.

V prezentácii Vincenta Lepetita "3D porozumenie scény z obrázkov" [6] je 3D poloha objektu reprezentovaná kombináciou jeho 3D polohy a orientácie v priestore. Konkrétne reprezentácia zahŕňa:

- **3D polohu (transláciu**): Označuje sa ako vektor *T=[T1,T2,T3]T*, ktorý predstavuje polohu objektu vzhľadom na referenčný bod, zvyčajne kameru alebo globálny súradnicový systém. [6]
- **3D orientácia (rotácia**): Je reprezentovaná maticou rotácie *R*, čo je matica 3x3, ktorá opisuje orientáciu objektu jeho otočením z referenčnej orientácie do jeho aktuálnej orientácie. Matica rotácie je pravouhlá, s determinantom +1, čo zabezpečuje, že predstavuje čistú rotáciu bez škálovania alebo zrkadlenia. [6]

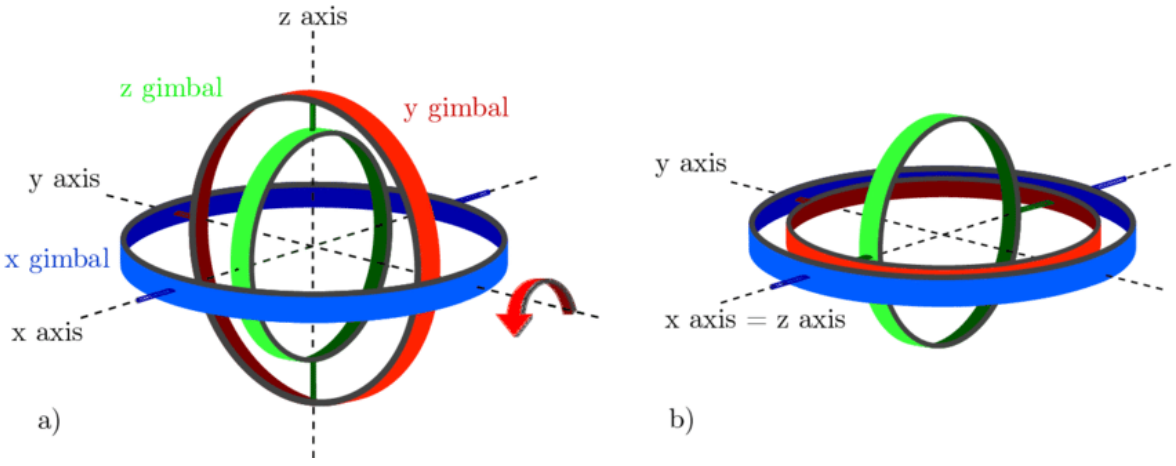
Kombinácia týchto prvkov poskytuje 6D pózu (3 polohy + 3 orientácie), ktorá zahŕňa kompletný priestorový stav objektu v 3D priestore:

- **Translačný vektor *T*** umiestňuje objekt do scény.
- **Matica rotácie *R*** zabezpečuje, aby objekt smeroval správnym smerom.

Okrem toho sa v prezentácii rozoberajú rôzne metódy parametrizácie matice rotácie *R*, pričom sa uznávajú problémy, ktorým každá metóda čelí (napr. gimbal lock pomocou Eulerových uhlov, normalizácia pomocou kvaterniónov). Cieľom týchto metód je ponúknuť spojitú, jednoznačnú reprezentáciu 3D rotácií vhodnú na optimalizačné a učebné úlohy, ktoré sú kľúčové pre presný odhad polohy v aplikáciách počítačového videnia. [6]

1.2.1 Gimbal lock s Eulerovými uhlami

Na opis tejto problematiky použijeme objekt ako napr. vozidlo. Na preklopenie a otočenie vozidla (jeho orientácie) sa používa jedna metóda, ktorá sa nazýva **Eulerove uhly**. Existuje však zložitý problém nazývaný **gimbal lock** (Obrázok 3.). Predstavte si, že máte tri rotujúce osi spojené do kopy, pričom každá sa môže otáčať jedným smerom. Ak dve z osi nastavíte tak, aby sa točili rovnakým smerom, zrazu stratíte schopnosť pohybovať sa nezávisle v jednom smere - to je gimbal lock. Pri Eulerových uhloch, ak sa orientácia auta nastaví určitým spôsobom, už nedokážeme rozlíšiť, či sa auto točí alebo nie. Stratíme jeden smer pohybu. [7]



Obrázok 3- Gimbal lock s Eulerovými uhlami), prevzaté z [7]

1.2.2 Normalizácia pomocou kvaterniónov

Iný spôsob, ako opísať, že sa auto prevracia a točí, je pomocou niečoho, čo sa nazýva **kvaternióny**. Predstavte si kvaternióny ako zložitejší, ale inteligentnejší spôsob, ktorý spočíval v pridaní ďalších dvoch imaginárnych dimenzií ku komplexným číslam a tým sa vyhnúť gimbal locku. Kvaternióny však musia byť vždy zachované v určitej dĺžke (normalizované), aby dávali zmysel, podobne ako by roztiahnutý alebo pokrčený kompas neukazoval presne sever. To znamená ďalšie matematické výpočty, aby bolo všetko v poriadku. [8]

1.3 Strata pri predpovedaní polohy

Aby sme zabezpečili, že predikcia polohy objektu modelom je čo najpresnejšia, porovnávame predpovedanú polohu so základnou pravdou (skutočnou polohou) pomocou stratovej funkcie. Stratová funkcia vypočítava, ako veľmi sa predikcia líši od pravdy, a usmerňuje model, aby počas trénovania upravoval a zlepšoval svoje predpovede. Strata pre predikciu pózy zvyčajne zahŕňa dve hlavné zložky[6]:

1. Strata pri translácii

Účel: Meria presnosť predpovedanej 3D polohy objektu.

Výpočet: Často sa počíta ako euklidovská vzdialenosť (norma L2) medzi predpovedaným vektorom translácie *($\hat{T}$)* a skutočným vektorom translácie *(T)*. Matematicky sa reprezentuje pomocou nasledujúcej rovnice (1.1). [6]

$$L_T = \left|\left|T - \hat{T}\right|\right|^2 \tag{1.1}$$

2. Strata otáčaním

Účel: Meria presnosť predpovedanej orientácie objektu.

Výzvy: Zisťuje sa, či je objekt otočený v smere hodinových ručičiek: Reprezentácia a porovnávanie orientácií (rotácií) je zložitejšie ako polohy vzhľadom na vlastnosti priestorov rotácií.

Výpočet: Jednou z bežných metód je použitie geodetickej vzdialenosti v priestore rotácií, ktorá meria najkratšiu cestu medzi dvoma orientáciami na jednotkovej guli. Možno ju vypočítať na základe stopy súčinu skutočnej matice rotácie *(R)* a transpozície predpovedanej matice rotácie *($\hat{R}$)*, formulovanej vzťahom (1.2). [6]

$$L_R = ||log\left(R\hat{R}^T\right)||_F = \cos^{-1}\left(\mathrm{tr}\left(R\hat{R}^T\right) - 1\right) / 2 \tag{1.2}$$

3. Kombinácia strát

Celková strata polohy*(*L_{pose}) kombinuje straty translácie a rotácie, prípadne s váhovým faktorom*(γ)* na vyváženie ich príspevkov, reprezentovaná vzťahom (1.3).

$$L_{pose}=L_{T}.+\gamma L_{T}. \tag{1.3}$$

Tento štruktúrovaný prístup k výpočtu strát umožňuje modelom učiť sa zo svojich chýb v predpovediach polohy aj orientácie a vykonávať potrebné úpravy na zlepšenie presnosti v priebehu času. Prezentácia Vincenta Lepetita poskytuje pohľad na zložitosti 3D odhadovania polohy a dôležitosť starostlivo navrhnutých stratových funkcií pri tréningu efektívnych modelov. [6]

1.4 Detekcia 3D objektu v obraze

Pri prechádzaní z 2D na 3D detekciu objektov do hry vstupuje niekoľko kritických faktorov a úvah, ktoré zásadne menia prístup k detekcii a interpretácii objektov v priestorovom prostredí. Na rozdiel od 2D detekcie, ktorá sa primárne zameriava na identifikáciu objektov v rámci roviny obrazu, 3D detekcia sa zameriava na pochopenie objektov v scéne na hlbšej, objemovej úrovni, pričom sa zohľadňuje ich veľkosť, tvar, orientácia a presná poloha v priestore. Nižšie uvádzame niektoré z hlavných aspektov 3D detekcie objektov.

1.4.1 Informácie o hĺbke

Pri detekcii 3D objektov je jedným zo základných rozdielov nutnosť presnej interpretácie informácií o hĺbke. Toto je možné dosiahnuť rôznymi prostriedkami, napríklad stereovidením, štruktúrovaným svetlom, kamerami s časom letu alebo snímačmi LiDAR, pričom každý z nich poskytuje údaje o hĺbke v rôznych formách a rozlíšeniach.

1.4.2 Reprezentácia údajov

Reprezentácia 3D údajov je kľúčovým aspektom, ktorý ovplyvňuje proces detekcie. Medzi bežné reprezentácie patria:

- **Mračná bodov:** kolekcie bodov v 3D priestore, zvyčajne získané zo senzorov LiDAR alebo štruktúrovaného svetla. Každý bod predstavuje časť povrchu objektu a obsahuje informácie o jeho polohe v priestore. [9]
- **Voxelové mriežky:** Voxelizácia rozdeľuje 3D priestor na mriežku objemových prvkov (voxelov), podobne ako pixely v obraze, ale s hĺbkou. Toto vyjadrenie mô-

že zjednodušiť spracovanie mračien bodov tým, že poskytuje jednotnú štruktúru, ale môže zaviesť chyby kvantizácie a zvýšiť výpočtové zaťaženie.[9]

- **Siete:** Súbor vrcholov, hrán a plôch, ktoré definujú tvar 3D objektu. Siete sa menej často používajú na detekciu, ale môžu byť užitočné na rekonštruovanie povrchu detegovaných objektov. [9]
- **Obrázky z viacerých pohľadov:** Niektoré prístupy používajú na odvodzovanie 3D informácií viacero 2D snímok nasnímaných z rôznych uhlov prostredníctvom triangulácie alebo modelov hlbokého učenia vycvičených na lepšie pochopenie priestorových vzťahov. [9]

1.4.3 Extrakcia vlastností a učenie

Získavanie zmysluplných vlastností z 3D údajov si so sebou prináša jedinečné výzvy vzhľadom na zložitosť a variabilitu 3D tvarov a riedkosť mračien bodov. Metódy ako PointNet [6] a jeho nasledovníci prispôsobujú modely hlbokého učenia na priame spracovanie mračien bodov, pričom sa učia vlastnosti, ktoré sú nemenné voči rôznym zmenám a odolné voči šumu.

1.4.4 Lokalizácia objektov a odhad polohy

Detegovanie objektu v 3D si vyžaduje okrem iného aj určenie jeho orientácie a polohy vzhľadom na daný referenčný rámec. Preto sú potrebné modely, ktoré predpovedajú nielen ohraničujúci rámček, ale aj rotáciu a prípadne celú 6D polohu (3D pozíciu a 3D orientáciu) objektu. [6]

1.4.5 Škálovateľnosť a výpočtová efektívnosť

Trojrozmerné dáta, obzvlášť husté mračná bodov alebo voxelové mriežky s vysokým rozlíšením, môžu byť podstatne väčšie a zložitejšie ako dvojrozmerné obrázky, čo predstavuje výzvu pre spracovanie a analýzu v reálnom čase. Techniky, ako sú konvolúcie s malým množstvom údajov, hierarchické spracovanie a efektívne dátové štruktúry, sú kľúčové pre škálovateľnosť a praktickosť detekcie 3D objektov pre aplikácie, ako je autonómne riadenie a robotika. [6]

1.4.6 Metódy detekcie objektov v obraze

Prieskum pri dnešnom objeme a rôznorodosti v danej oblasti nebolo možné spraviť komplexne pre všetky existujúce metódy. V nasledujúcej kapitole sú podrobnejšie popísane nasledujúce metódy, ktoré boli pri tomto prieskume nájdene. Zohľadnená bola aj popularita či konkurencie schopnosť jednotlivých modelov ako aj subjektívna stránka v rámci záujmu o danú tému.

V ďalšej kapitole sú predstavené a podrobnejšie popísane modely pre detekciu objektov v obraze.

2 MODELY PRE DETEKCIU OBJEKTOV V OBRAZE

Táto kapitola sa zaoberá podrobnejším popisom jednotlivých metód pre detekciu objektov v obraze. Výkonnosť modelov v tejto kapitole je testovaná v praktickej časti.

2.1 Faster R-CNN

Štúdia uvádza Faster R-CNN ako vylepšenie predošlého modelu Fast R-CNN. Integruje sieť návrh regiónov (RPN – Region proposal Network) s modelom na detekciu objektov Fast R-CNN. Tým rieši v štúdií uvedenú výpočtovú neefektívnosť samostatných RPN čím bola vytvorená jednotná sieť pre detekciu objektov. Zjednodušené povedané RPN hovorí kde by mal Fast R-CNN objekty hľadať a detegovať ich. [10]

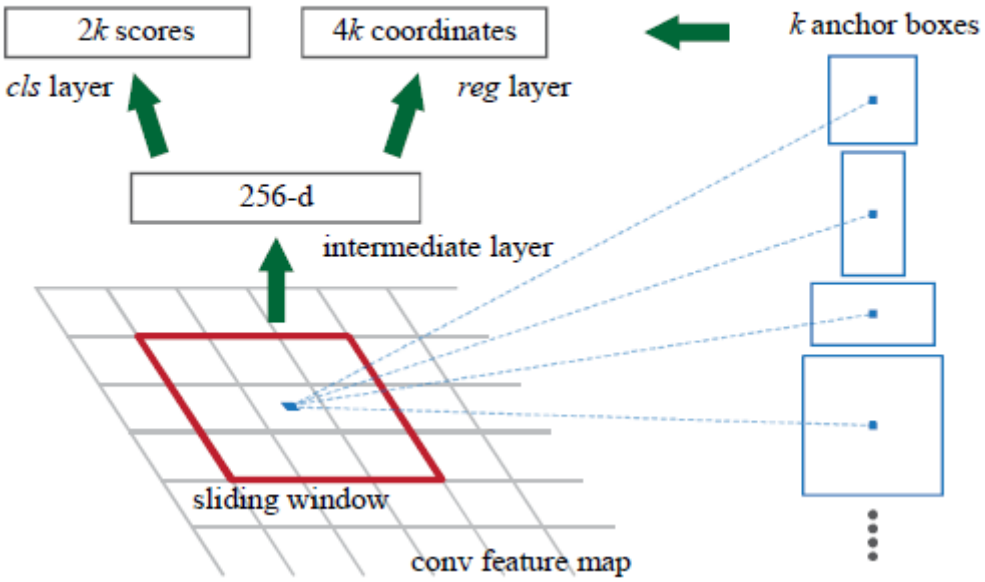
2.1.1 Sieť návrhu regiónov (RPN)

RPN je plne konvolučná sieť čo znamená, že sa dokáže sa prispôsobiť ľubovoľnej veľkosti vstupného obrázku je navrhnutá na predpovedanie hraníc objektov a skóre „objektovosti“ (z angl. objectness scores) na každej pozícii na mape prvkov extrahovaných z obrazu. Táto sieť v podstate poskytuje návrhy regiónov takmer bez dodatočných výpočtových nákladov. RPN pridáva niekoľko konvolučných vrstiev na zdieľané konvolučné funkcie, ktoré vytvárajú súbor obdĺžnikových návrhov objektov a skóre označujúce prítomnosť objektu v týchto návrhoch. Sieť používa kotviace rámčeky (z angl. Anchor boxes). [10]

Na Obrázku 4. môžeme vidieť schému fungovania RPN. Fungovanie je popísané v nasledujúcich bodoch:

- **Mapa prvkov (conv feature map):** RPN pracuje s mapou prvkov získanou z poslednej spoločnej konvolučnej vrstvy. Táto mapa obsahuje bohaté vizuálne vlastnosti vysokej úrovne získané zo vstupného obrazu.
- **Posuvné okno (sliding window):** Malá sieť (označovaná ako minisieť) sa posúva po mape prvkov. Táto sieť sa aplikuje na každom mieste (t. j. na každej pozícii) na mape prvkov a spracúva malé okno mapy prvkov.
- **Medzivrstva:** „256-d“ znamená, že výstupom medzivrstvy je vektor príznakov s 256 dimenziami.
- **Vrstva klasifikácie polí (cls layer):** Táto vrstva produkuje skóre „objektovosti“, ktoré hodnotí, či okno obsahuje objekt bez ohľadu na triedu objektu.

- **Regresná vrstva (reg layer):** Táto vrstva vypisuje súradnice ohraničujúcich boxov, ktoré predpovedajú umiestnenie objektu vzhľadom na kotviace boxy.



Obrázok 4. – Sieť návrhu regiónov (RPN), prevzaté z [10]

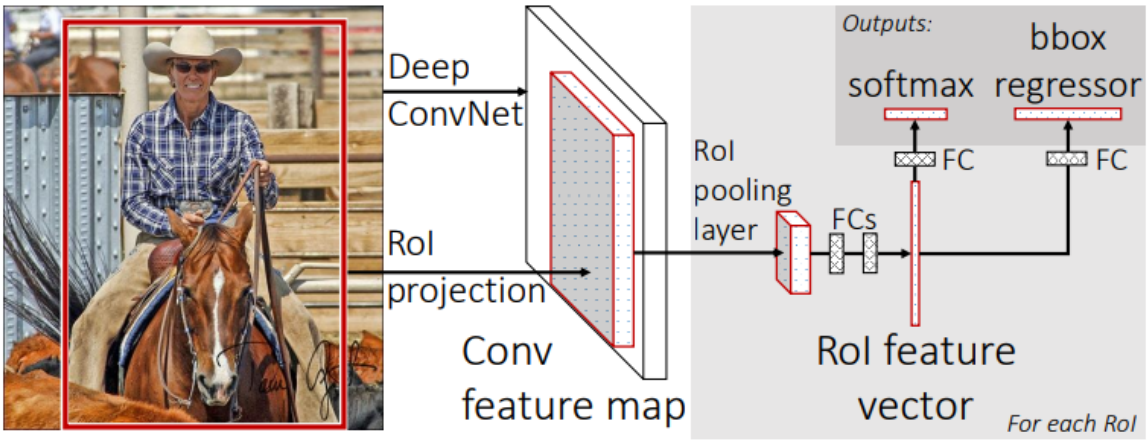
RPN kombinuje dve stratové funkcie. Funkcia pre klasifikačnú stratu hodnotí, ako presne dokáže sieť určiť, ktoré kotviace rámčeky pravdepodobne obsahujú objekt. Rozlišuje oblasti s objektmi (popredie) a oblastí bez objektov (pozadie). Druhou funkciou je regresná strata posudzujúca ako presne sa predpovedané rámčeky zhodujú so skutočnými hranicami objektov. Kombináciou týchto dvoch funkcií dokáže sieť presne predpovedať prítomnosť objektov, ale aj presne lokalizovať a určiť veľkosť týchto objektov v rámci obrazu.[10]

2.1.2 Fast R-CNN na detekciu objektov

Fast R-CNN deteguje objekty spracovaním celého obrazu cez niekoľko konvolučných vrstiev (deep conv net) a vrstiev maximálneho združovania (max pooling layers), aby sa vytvorila mapa príznakov. Táto mapa zachytáva komplexné vizuálne vlastnosti obrazu, ktoré sa využívajú na detekciu objektov. Pre každý návrh objektu vrstva RoI (Region of Interest) pooling extrahuje z mapy príznakov vektor príznakov pevnej veľkosti. To sa dosiahne rozdelením RoI na pevnú mriežku a použitím maximálneho združovania na každú bunku mriežky, čím sa vytvorí jednotný výstup, ktorý je vhodný na klasifikáciu bez ohľadu na pôvodnú veľkosť RoI. [11]

Výstupom sú spracované funkcie pre každý RoI. Tie potom prechádzajú cez plne prepojené vrstvy, čo vedie k dvom kľúčovým výstupom: jedna vrstva predpovedá pravdepodobnosti softmax pre klasifikáciu objektov (vrátane kategórie pozadia) a ďalšia vrstva vystupuje ako regresory ohraničenia, ktoré spresňujú súradnice pre každú zistenú triedu objektov.

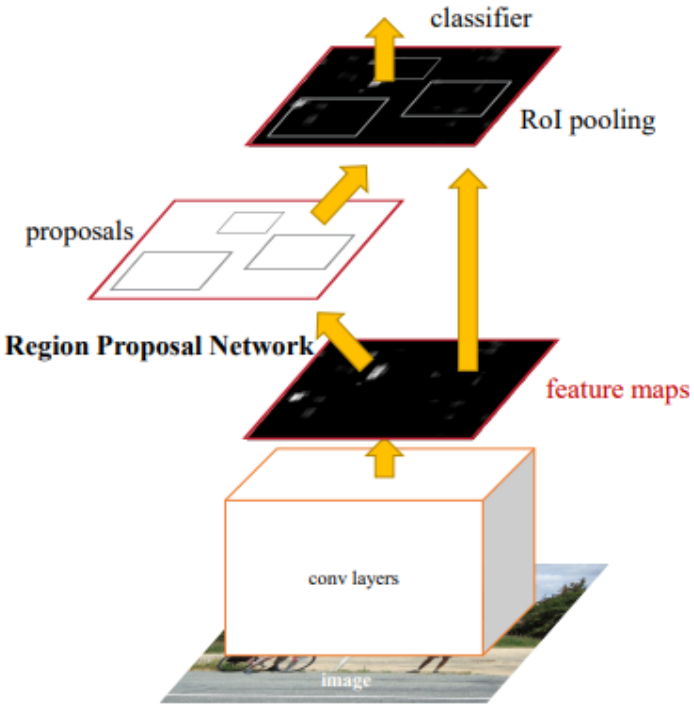
Fast R-CNN zvýšil účinnosť detekcie objektov začlenením jednotného modelu, ktorý zvláda klasifikáciu objektov aj regresiu ohraničujúcich polí. Tento model používa spoločné konvolučné funkcie v rôznych návrhoch regiónov prostredníctvom metódy združovania regiónov záujmu (RoI). Tento prístup využíva detekčný kanál, čím urýchľuje proces detekcie a zároveň ignoruje čas strávený generovaním návrhov regiónov. Na obrázku 5. je znázornená architektúra tohto systému.[11]



Obrázok 5. Architektúra Fast R-CNN, prevzaté z [11]

2.1.3 Prínos Faster R-CNN

Štúdia o Faster R-CNN rieši problematické miesto Fast R-CNN súvisiace s pomalým a výpočtovo nákladným procesom generovania návrhov regiónov pomocou externých metód. Faster R-CNN to rieši zavedením siete návrhov regiónov (RPN), ktorá je integrovaná do CNN na predpovedanie hraníc objektov priamo z konvolučných funkcií, čo umožňuje systému generovať návrhy vyššou rýchlosťou a s vyššou efektívnosťou. Táto integrácia umožňuje Faster R-CNN využívať spoločné konvolučné funkcie na generovanie návrhov aj detekciu objektov, čím sa celý proces výrazne urýchľuje. Trénovaním RPN spoločne s detekčnou sieťou Faster R-CNN optimalizuje aj celkový výkon, čo vedie k zlepšeniu presnosti detekcie objektov. Na Obrázku 6. je schematicky naznačené ako tento model funguje.[10]



Obrázok 6. Architektúra Faster R-CNN, prevzaté z [10]

2.2 RetinaNet

Retina net je oproti Fast R-CNN jednostupňovým detektorom objektov. Štúdia uviedla fokálnu stratovú (Focal Loss) matematickú funkciu, čím sa snažila vyriešiť problém medzi jednostupňovými detektormi. RetinaNet bola založená na architerktúre ResNet popisanej v nasledujúcej podkapitole. [12]

2.2.1 ResNet

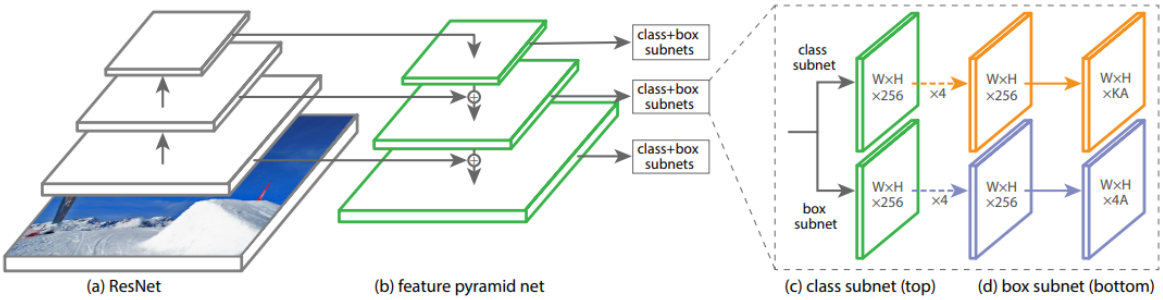
Výskumníci pridávali do neurónových sietí ďalšie vrstvy, aby sa zlepšili v rozpoznávaní obrázkov. Pridávanie príliš veľkého počtu vrstiev však spôsobilo problém nazývaný mizнúci alebo explodujúci gradient. Siete sa tak ťažko trénovali a ich presnosť klesala so zvyšujúcim sa počtom vrstiev. Napríklad pri porovnávaní 20-vrstvovej siete s 56-vrstvovou sieťou dosahovala 56-vrstvová sieť kvôli tomuto problému horšie výsledky na tréningových aj testovacích údajoch. [13]

Potom v roku 2015 prišli výskumníci zo spoločnosti Microsoft Research s riešením nazvaným ResNet. Namiesto jednoduchého pridávania ďalších vrstiev zaviedol ResNet niečo, čo sa nazýva "zvyškové bloky" (residual blocks). Tieto bloky umožňujú, aby informácie preskočili cez určité vrstvy, čím sa sieť ľahšie učí. [13]

V sieti ResNet sa každý blok učí upravovať rozdiel medzi tým, čo sieť predpovedá, a tým, čo by mala predpovedať. Takto sa sieť sústredí na učenie častí, na ktoré ešte neprišla, namiesto toho, aby všetko prerábala od začiatku. Tento prístup pomáha sieti ResNet trénovať rýchlejšie a dosahovať lepšie výsledky aj pri veľkom počte vrstiev. [13]

2.2.2 Architektúra RetinaNet

Predošlé jednostupňové detektory museli pre každý pixel vygenerovať nejaké kotviace rámčeky a pomery strán a tým vytvárali nepresné informácie. Fokálna stratová funkcia venuje menej pozornosti ľahkým príkladom a viac sa zameriava na tie ťažšie. Robí to pomocou špeciálneho vzorca, ktorý počas trénovania znižuje dôležitosť ľahkých príkladov a zvyšuje dôležitosť ťažkých príkladov. Táto sieť je zložená zo základnej siete (backbone network) a dvoch podsietí so svojou špecifickou úlohou. Úlohou základnej siete je výpočet konvolučnej mapy príznakov nad celým vstupným obrázkom pomocou štandardnej konvolučnej siete. Prvá podsieť na výstupe základnej siete vykoná konvolučnu klasifikáciu objektov, zatiaľ čo druhá podsieť vykonáva konvolučnú regresiu ohraničujucích polí. Konvolučná klasifikácia objektov zahŕňa identifikáciu a kategorizáciu objektov v rámci obrazu pričom, každému ukotvenému rámčeku priradí klasifikačné ciele na základe prítomnosti objektov a ich zodpovedajúcich tried. Regresia ohraničujúcich boxov sa vzťahuje na proces spresňovania umiestnenia a veľkosti ohraničujúcich boxov okolo objektov zistených v obraze. Táto sieť využíva pyramídovú sieť FPN (Feature Pyramid Network), ktorá založená na architektúre ResNet. Základna siete sa tým rozširuje o cestu z hora na dol a bočne spojenia, čím konštruuje pyramídu prvkov vo viacerých mierkach zo vstupného obrazu s jedným rozlíšením. Na Obrázku 7. a) Je vidieť schému ResNet architektúry spojenej s FPN architektúrou Obrázok 7. b). Výstup potom smeruje do vyššie spomínaných podsietí znázornených na Obrázku 7. c) a d). [13]



Obrázok 7. Architektúra RetinaNet, prevzaté z [13]

2.3 RTMDet

RTMDet je architektúra využívaná na detekciu objektov v obraze v reálnom čase. Zaraďuje sa k jednostupňovým detektorom podobne ako RetinaNet. Výskumníci sa zamerali na viac vlastností, ktoré sa snažili vylepšiť oproti iným doteraz používaným metódam, ktoré sa zameriavali na rozpoznanie objektov na obrázku. Našli spôsob, ako toto porozumenie zlepšiť bez toho, aby sa proces spomalil alebo skomplikoval. Namiesto použitia predchádzajúcich metód, ktoré boli príliš pomalé, vyskúšali niečo nové, čo sa nazýva konvolúcia veľkých jadier. Tie pomáhajú počítačom pochopiť viac detailov z obrázkov bez toho, aby pridávali príliš veľa práce navyše. Túto myšlienku otestovali a zistili, že vďaka nej je ich systém presnejší. V porovnaní s inými technikami bola táto nová metóda jednoduchšia a stále účinná, čo z nej robí dobrú voľbu na zlepšenie rozpoznávania obrázkov. [14]

2.3.1 Popis architektúry RTMDet

Architektúra systému je rozdelená na chrbticu, krk a hlavu. Zameriava sa na vylepšenie základných stavebných blokov s cieľom zlepšiť presnosť detekcie a segmentácie objektov pomocou hĺbkových konvolúcií. Systém využíva stratégie na vyváženie šírky a hĺbky modelu s cieľom optimalizovať výpočet bez straty presnosti. Okrem toho sa zlepšuje detekcia viacrozmerných prvkov bez výrazného zvýšenia výpočtového zaťaženia. Nakoniec sa na detekciu objektov v reálnom čase používa prístup zdieľanej detekčnej hlavy, pričom sa kladie dôraz na zdieľanie parametrov v rôznych mierkach pri zachovaní presnosti modelu. [14]

3 METÓDY VYUŽÍVANÉ NA DETEKCIU 3D OBJEKTOV V OBRAZE

Pri dnešnom rozmachu umelej inteligencie sa vo svete nachádza mnoho prístupov a spôsobov, ktoré sa testujú a skúšajú. Zložitosť a príliš veľký objem dát, ktorý sa s obrazom ako takým spája, priamo navádza na to aby sa tieto metódy prispôsobovali resp. vytvárali tak aby spĺňali určitý účel. V nasledujúcich kapitolách sú uvedené inovatívne prístupy využitia detekcie a segmentácie 3D objektov prispôsobené na rôzne účely. Nasledujúce prístupy boli zvolené na základe ich rôznorodosti v aplikáciách a odlišnosti k prístupu riešenia problematík v danom odvetví.

3.1 Semi-Supervised Learning (SSL) na detekciu mikroskopických defektov

Výskum predstavuje nový rámec využívajúci hlboké učenie na automatizovanú detekciu a segmentáciu vysokopásmových pamätí (HBM) v 3D röntgenových snímkach. Použitá metóda je charakterizovaná ako "Semi-Supervised Deep Learning", ktorá výrazne zvyšuje presnosť lokalizácie mikroskopických defektov, ako sú napríklad hrbole a dutiny, s minimálnym úsilím pri manuálnom označovaní.

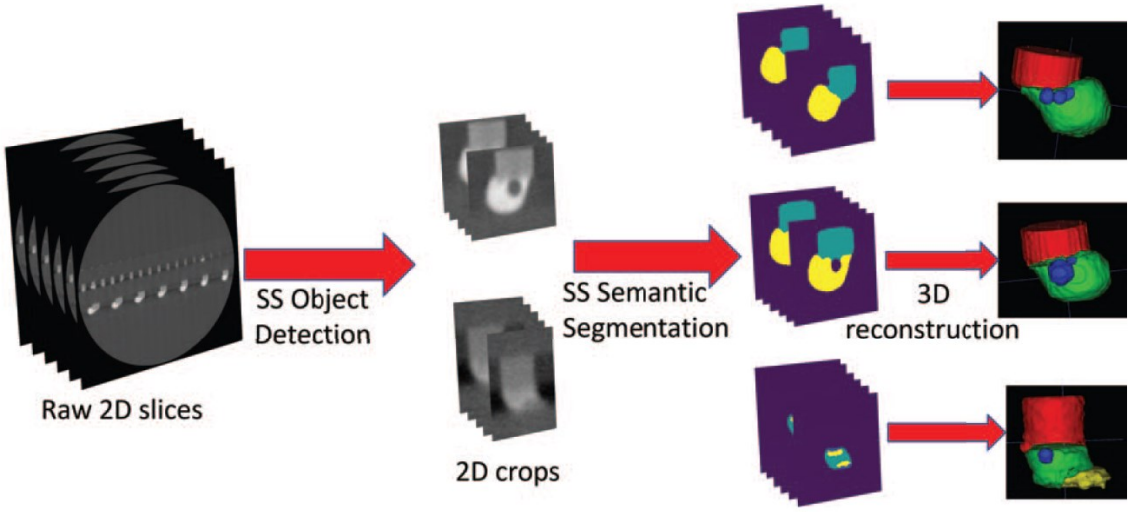
- **Detekcia objektov s čiastočnou supervíziou**: Identifikuje a lokalizuje objekty záujmu (napr. pamäťové a logické výpadky) v 2D rezoch extrahovaných z 3D skenov. [15]
- **Sémantická segmentácia pod čiastočnou supervíziou**: Klasifikuje každý pixel v detekovaných oblastiach do špecifických kategórií (napr. spájky, dutiny, medené stĺpiky, medené podložky), čím uľahčuje podrobnú analýzu komponentov. [15]

3.1.1 Metodika a testovanie

Na začiatku bolo nutné vytvoriť dáta na ktorých bude testovanie prebiehať. Testovacie vozidlá simulujúce polovodičové obaly v tejto štúdií boli vyrobené so zámerne vyvolanými defektmi. Tieto boli skenované pomocou 3D röntgenovej mikroskopie (XRM) s cieľom vytvoriť voxelové údaje s vysokým rozlíšením na analýzu. [15]

Po vytvorení požadovaných anomálií sa prešlo na extrakciu 2D rezov. Z 3D XRM skenov sa získali virtuálne 2D rezy na analýzu. Extrahujeme surové 2D rezy pre každý 3D XRM

sken, vykonáme detekciu objektov s čiastočnou supervíziou pre každý 2D rez, po ktorom nasleduje sémantická segmentácia s čiastočnou supervíziou pre každú jednotlivú zložku. Následne zrekonštrujeme segmentácie v 3D na účely kvality a vizuálnej analýzy. [15]

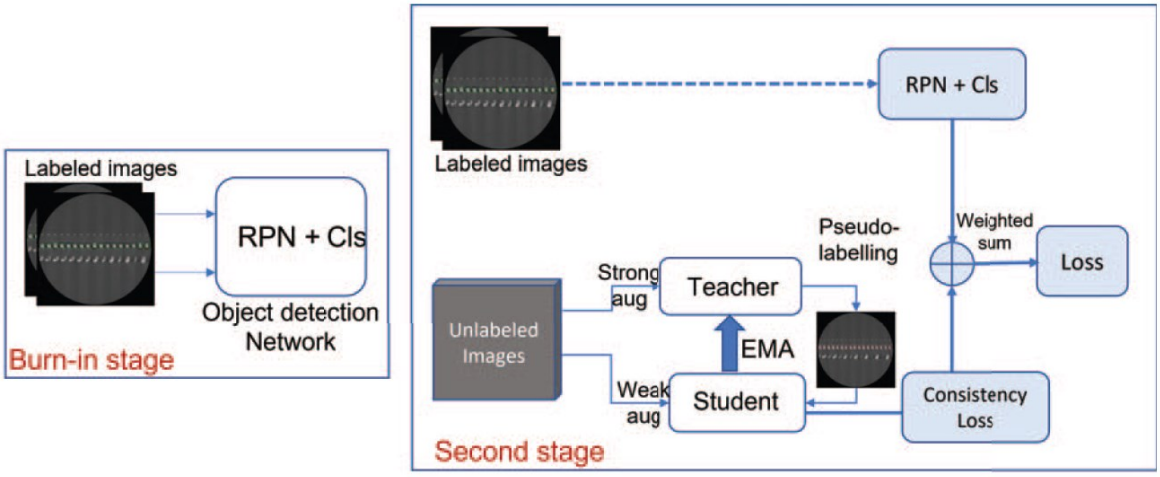


Obrázok 8. Extrakcia rezov, prevzaté z [15]

1. **Detekcia objektov**: Cieľom frameworku detekcie objektov je identifikovať a lokalizovať objekty v 2D rezoch extrahovaných z 3D röntgenových snímok. Tento proces zahŕňa rozpoznávanie špecifických prvkov alebo štruktúr (napr. pamäťových a logických hrbolčekov) na rezoch. [15]
    1. **Prístup učenia s čiastočnou supervíziou**: Rámec využíva model čiastočne supervizovaného učenia, ktorý využíva označené aj neoznačené údaje. Označené údaje prejdú fázou vypaľovania, aby sa vycvičil počiatočný model, ktorý sa potom zdokonalí pomocou väčšieho objemu neoznačených údajov. [15]
    2. **Model učiteľa a študenta**: Kľúčovou zložkou je architektúra učiteľ - študent. Model učiteľa, vycvičený na označených údajoch, vytvára pseudoznačenia pre neoznačené údaje. Študentský model sa učí na základe skutočných označení aj týchto pseudoznačení pod vedením učiteľského modelu. [15]

3. **Rozšírenie údajov**: Na vstupné údaje pre modely učiteľa a študenta sa používajú rôzne techniky rozšírenia údajov. Učiteľský model dostáva slabo rozšírené údaje, zatiaľ čo študentský model je trénovaný na silne rozšírených údajoch, aby sa zabezpečilo robustné učenie. [15]

4. **Funkčnosť**: Model detekcie objektov vypisuje ohraničujúce polia pre každý detekovaný objekt v 2D rezoch, pričom každé ohraničujúce pole určuje polohu a veľkosť objektu. Tieto identifikované oblasti sa potom orežú na ďalšie spracovanie, napríklad na segmentáciu. [15]



Obrázok 9. Fungovanie detekčného frameworku, prevzaté z [15]

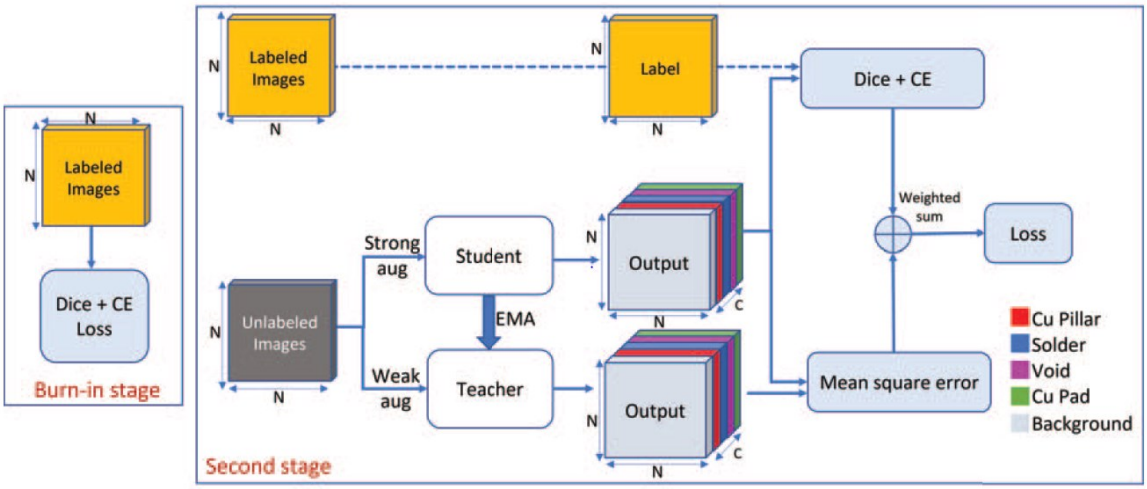
2. **Sémantická segmentácia**: Sémantický segmentačný framework klasifikuje každý pixel v rámci oblastí identifikovaných modelom detekcie objektov. Tento krok je kľúčový pre pochopenie zloženia detekovaných objektov, rozlišovanie medzi rôznymi materiálmi alebo komponentmi (napr. spájky, dutiny, medené stĺpiky, medené podložky). [15]

    1. **Prístup učenia s čiastočným dohľadom**: Podobne ako v prípade detekcie objektov, aj model segmentácie využíva učenie s polovičnou supervíziou, pričom využíva obmedzený súbor označených údajov aj rozsiahlejší súbor neoznačených údajov. [15]

    2. **Učenie založené na konzistentnosti**: Model kladie dôraz na konzistentnosť predpovedí pri rôznych rozšíreniach tých istých vstupných údajov, vďaka čomu je model schopný spoľahlivo predpovedať aj napriek prítomnosti odchýliek. [15]

3. **Architektúra a stratové funkcie**: Segmentačný model zvyčajne využíva architektúru kodéra a dekodéra, ako napríklad U-Net[20], optimalizovanú pomocou kombinácie stratových funkcií, ktoré môžu zahŕňať vzájomnú entropiu a stratu Dice. Tieto funkcie merajú presnosť klasifikácie podľa pixelov v porovnaní so základnou pravdou. Stratova funkcia vzájomnej entropie meria rozdiel medzi predikovanými pravdepodobnosťami a skutočnými štítkami. Je to miera neistoty modelu pri predikcii správnej triedy pixelu. Diceova strata je založená na Diceovom koeficiente, ktorý je štatistickou mierou podobnosti medzi predpokladanou segmentáciou a skutočnou segmentáciou. Je obzvlášť užitočná, keď sú triedy nevyvážené, teda niektoré triedy sú zastúpené oveľa menej ako iné. [15]

Výstupom je podrobná segmentačná mapa pre každú orezanú oblasť, pričom každý pixel je priradený k určitej triede na základe materiálu alebo komponentu, ktorý predstavuje. Táto podrobná klasifikácia umožňuje komplexnú analýzu štruktúry objektu a všetkých prítomných chýb. [15]



Obrázok 10. Segmentačný framework, prevzaté z [15]

Integrácia pre 3D rekonštrukciu

Frameworky pre detekciu objektov aj sémantickú segmentáciu sa podieľajú na procese, ktorý nielen identifikuje a klasifikuje prvky v 2D rezoch, ale aj uľahčuje rekonštrukciu týchto prvkov v 3D. Tento jednotný prístup poskytuje detailné pochopenie 3D štruktúry objektov a materiálového zloženia, čo je nevyhnutné pre aplikácie, ako je kontrola kvality pri výrobe polovodičov. [15]

3.1.2 Zhrnutie

Metodika využíva učenie s čiastočnou supervíziou a efektívne využíva neoznačené údaje, čím rieši problém obmedzených súborov označených údajov v komplexných zobrazovacích aplikáciách. Tento prístup ukazuje potenciál pokročilých techník strojového učenia na výrazné zlepšenie automatizovanej analýzy v kontexte priemyselného zobrazovania.

3.2 Unsupervised object discovery method (metóda objavovania objektov bez učiteľa)

Štúdia predstavuje metódu objavovania objektov bez učiteľa na zisťovanie zmien v 3D skenoch v priebehu času bez predchádzajúcej znalosti objektov prítomných v scéne. Táto metóda jedinečným spôsobom kombinuje detekciu 3D zmien s 2D segmentáciou, pričom využíva 2D segmentačné masky na spresnenie počiatočného súboru detekcií 3D zmien. Počiatočné detekcie získané prostredníctvom prístupu vykresľovania a porovnávania sú zvyčajne neúplné a pravdepodobne zodpovedajú pohyblivým objektom. Tie sa potom spresňujú pomocou optimalizácie grafov, pričom sa informácie z 2D segmentačných masiek prenašajú do 3D priestoru, aby sa zachytil celý rozsah zmien objektu. [16]

3.2.1 Metodika a testovanie

Metóda opísaná v štúdii zahŕňa diferencovaný prístup k detekcii zmien v 3D prostredí s využitím 2D segmentačných masiek generovaných z hĺbkových aj farebných obrazov pomocou modelu Segment Anything Model (SAM). [16]

Aplikácia na farebné a hĺbkové obrázky: SAM sa aplikuje bez akéhokoľvek dolaďovania na farebné obrazy zachytené v sekvencii RGB-D aj na hĺbkové obrazy. Táto aplikácia je pozoruhodná, pretože SAM nebol špeciálne trénovaný na hĺbkové mapy. Schopnosť aplikovať SAM na hĺbkové obrazy vzhľadom na konzistentnú hĺbku objektov vzhľadom na ich pozadie ukazuje jeho účinnosť pri identifikácii konzistentných 2D objektov, ako to opisujú odvodené segmentačné masky. [16]

3.2.2 Počiatočná detekcia zmien a tvorba supervoxelov

Počiatočná detekcia: Počiatočná sada detekcií, označovaná ako "seeds", sa identifikuje prostredníctvom prístupu vykresľovania a porovnávania, pričom sa odhaľujú oblasti s výraznými hĺbkovými rozdielmi. Táto počiatočná detekcia je však neúplná, zachytáva najmä rozsiahle zmeny a môže prehliadnuť menšie alebo jemnejšie zmeny. [16]

Supervoxelová reprezentácia: Trojrozmerné scény sú reprezentované pomocou supervoxelov, ktoré ponúkajú hrubú segmentáciu, ktorá zachováva hranice objektov zohľadnením priestorových, farebných a geometrických charakteristík. Táto reprezentácia je kľúčová pre segmentáciu scény a objavovanie objektov, pričom slúži ako zmysluplná reprezentácia nad bežným priestorovým informačným obmedzením voxelovej reprezentácie. [16]

Obrázok 11. Aplikácia segmentačných masiek na hlbkové a farebne obrazy, prevzatý z [16]

3.2.3 Optimalizácia grafov a šírenie zmien

Kľúčovou zložkou metodiky je **optimalizácia grafu**, ktorá spresňuje počiatočné detekcie presadzovaním obmedzenia segmentačnej masky. Tento optimalizačný proces destiluje informácie z 2D segmentačných masiek do 3D priestoru, čím zabezpečuje komplexnú detekciu zmien v celej scéne. Dosahuje to šírením zmien zo „seedov“ do všetkých častí scény, ktoré majú rovnakú masku SAM, čím sa identifikuje kompletný objekt. [16]

Na riešenie problému nadmernej segmentácie objektov, keď môžu byť objekty v odvodených maskách rozdelené do viacerých segmentov, sa nasadzuje **analýza prepojených**

komponentov. Táto analýza spája priestorovo prepojené segmenty, čím zabezpečuje, že viacero 2D segmentov zodpovedajúcich tomu istému objektu vedie k jednotnej reprezentácii v 3D priestore. [16]

3.2.4 Integrácia a nasadenie

Automatický generátor masiek SAM sa nasadzuje na farebné obrazy aj hĺbkové obrazy sekvencie RGB-D. Podnetom každého obrazu s mriežkou bodov a spracovaním 20 zväčšených výrezov obrazu generátor masiek vytvára segmentačné masky, ktoré sa potom spätnou projekciou priradia k príslušným 3D bodom. Toto priradenie umožňuje priradiť masku SAM každému supervoxelu, čo uľahčuje identifikáciu segmentov objektov, ktoré sa pohybujú spoločne. [16]

3.2.5 Zhrnutie

Táto metodika vyniká inovatívnym využitím 2D segmentačných masiek na spresnenie počiatočných detekcií 3D zmien, čím demonštruje robustný prístup k identifikácii a segmentácii zmenených objektov v 3D scénach bez toho, aby sa spoliehala na vopred definovaný súbor tried objektov. Aplikácia SAM na farebné aj hĺbkové obrazy spolu s optimalizáciou grafov a analýzou prepojených komponentov demonštruje významný pokrok v oblasti detekcie 3D zmien.

3.3 Dvojstupňový algoritmus na segmentáciu satelitných snímok

Štúdia predstavuje dvojstupňový algoritmus na segmentáciu satelitných snímok so zameraním na detekciu rozsiahlych oblastí s textúrou a menších objektov s vysokou presnosťou. Metóda využíva reprezentáciu viacmiestnych digitálnych snímok ako sady bitových obrazov a využíva matematický framework dvojrozmerných Markovových reťazcov. [19]

Markovov reťazec je matematický systém, ktorý prechádza z jedného stavu do druhého. Je to spôsob modelovania rôznych scenárov, v ktorých to, čo sa stane ďalej, závisí len od aktuálnej situácie, a nie od toho, ako sa do nej veci dostali. Táto vlastnosť je známa ako bezpamäťovosť. Zjednodušene povedané, Markovov reťazec nám hovorí, že budúcnosť je určená len prítomnosťou, nie minulosťou. [19]

V kontexte spracovania satelitných snímok sú Markovove reťazce neuveriteľne užitočné. Povedzme, že každý pixel alebo skupina pixelov na obrázku predstavuje stav, ktorý je definovaný vlastnosťami, ako je ich farba alebo jas. Ide o to, pozrieť sa na tieto pixely a predpovedať, ako by mohli vyzerať okolité pixely. Táto predpoveď je založená na pravdepodobnostiach, ktoré sú usporiadané v štruktúre známej ako prechodová matica - módny výraz pre mriežku, ktorá nám ukazuje pravdepodobnosť prechodu z jedného stavu pixela do druhého.[19]

V prípade satelitných snímok, ktoré môžu byť rozsiahle a zložité, použitím dvojrozmerného Markovovho reťazca umožňuje zohľadniť priestorové vzťahy medzi pixelmi. To znamená, že sa nepozeráme len doľava a doprava (horizontálne), ale aj hore a dole (vertikálne). Pochopením týchto vzťahov Markovov reťazec pomáha zmapovať, s akou pravdepodobnosťou sa určité textúry alebo farby nachádzajú vedľa seba. Napríklad zhluk stavov zelených pixelov môže naznačovať les a znalosť toho, ako často sú zelené pixely vedľa seba, to pomáha potvrdiť. [19]

Hlavným cieľom tohto algoritmu je uľahčenie rôznych úloh súvisiacich s hospodárením so zdrojmi Zeme a monitorovaním zmien životného prostredia. Medzi tieto úlohy patrí identifikácia rôznych typov terénu, odhad využitia pôdy a detekcia prírodných javov, ako sú záplavy, požiare a úniky ropy. Metóda sa snaží zlepšiť presnosť segmentácie a zároveň minimalizovať potrebné výpočtové zdroje, vďaka čomu je vhodná na použitie v palubných počítačových systémoch s obmedzenými možnosťami. [19]

3.3.1 Prístup a použité metódy

Algoritmus je rozdelený do dvoch hlavných etáp, z ktorých každá je prispôsobená na segmentáciu rôznych prvkov v rámci satelitných snímok:

1. **Prvá etapa - detekcia textúry oblasti:**

   Cieľ: Identifikovať rozšírené oblasti s homogénnymi štatistickými charakteristikami, ako sú lesy, polia a mestské štruktúry.

   Metódy: Využíva techniky texturálnej segmentácie, ktoré hodnotia štatistické charakteristiky a spektrálne vlastnosti záujmových oblastí. Algoritmus vypočíta pravdepodobnosti prechodu pre dvojrozmerný Markovov reťazec, pričom tieto pravdepodobnosti spolu s jasovými charakteristikami používa ako hlavné texturálne vlastnosti na detekciu rozšírených objektov. [19]

2. **Druhá etapa - detekcia objektov malých rozmerov:**

   Cieľ: Detekcia a zvýraznenie menších objektov, ako sú budovy a cesty, v rámci segmentovaných texturálnych oblastí.

   Metódy: Využíva techniky, ako sú konvolučné neurónové siete (CNN), detektory obrysov a operácie na odstránenie šumu a zjednodušenie tvaru. Táto fáza sa zameriava na analýzu kontúr a farebných prvkov s cieľom presne určiť menšie objekty na širšom textúrnom pozadí. [19]

Jadrom tohto algoritmu je použitie dvojrozmerných Markovových reťazcov na vyhodnotenie texturálnych vlastností obrazov. Každá farebná zložka RGB obrazu sa považuje za g-bitový digitálny poltónový obraz (DHI), ktorý možno rozložiť na súbor g-bitových bitových obrazov (BBI). [19]

Tento rozklad umožňuje:

Vyhodnotenie vlastností: Vlastnosti sa vyhodnocujú pomocou binárnych obrazov, ktoré predstavujú najvyššie a najinformatívnejšie číslice obrazových údajov. Táto selektívna analýza pomáha znižovať výpočtové nároky. [19]

Výpočty pravdepodobnosti: Vypočítavajú sa pravdepodobnosti prechodov medzi rôznymi stavmi obrazových prvkov s cieľom vytvoriť textúrnu mapu oblasti, čo pomáha v procese segmentácie. Pravdepodobnosť prechodu každého stavu v Markovovom reťazci pred-

stavuje pravdepodobnosť prechodu z jedného stavu pixelu do druhého v rámci daného okolia, čím sa poskytuje štatistický model textúrnej variability v celom obraze. [19]

Jas a lokálne vlastnosti: Algoritmus zohľadňuje lokálne zmeny pravdepodobnostných a jasových charakteristík v celom obraze, pričom na efektívny výpočet týchto metrík používa dvojrozmerné posuvné okno. G-bitová reprezentácia je rozhodujúca, pretože zachytáva úrovne stupňov šedej v obraze, čím zvyšuje schopnosť algoritmu odhaliť jemné rozdiely v textúre a farbe. [19]

3.3.2 Zhrnutie

Dvojstupňový algoritmus, o ktorom sa hovorí v tejto kapitole, ukazuje spôsob segmentácie satelitných snímok, vďaka ktorému je možné s vysokou presnosťou identifikovať veľké oblasti aj malé detaily. Pomocou systému založeného na Markovových reťazcoch algoritmus účinne spracováva snímky tak, že skúma len aktuálne informácie v pixeloch, nie ich históriu. Táto metóda nielen zvyšuje presnosť segmentácie, ale aj znižuje potrebný výpočtový výkon, čo je ideálne pre systémy s obmedzenými zdrojmi. Študia môže inšpirovať svojím prístupom k danej problematike iných výskumníkov alebo aj laikov zaujímajúcich o túto oblasť. Či už z hľadiska spracovania satelitných snímkov alebo aj záujemcov o vyžutie podobnej metódy na v palubných počítačoch a podobne.

3.4 LU-Net: Pokročilá sémantická segmentácia 3D mračien bodov LiDAR

Sémantická segmentácia 3D mračien bodov je dôležitým prvkom v rôznych aplikáciách, ako je autonómne riadenie a mobilná robotika. Tradičné metódy sa často spoliehajú na ručne vytvorené funkcie, ktoré vyžadujú rozsiahle ladenie parametrov a sú výpočtovo náročné. Táto štúdia predstavuje „LU-Net“, inovatívny prístup k sémantickej segmentácii 3D mračien bodov generovaných pomocou LiDAR. LU-Net využíva novú end-to-end architektúru, ktorá využíva silné stránky U-Net [20], osvedčeného sieťového modelu v segmentácii obrazu, prispôsobeného pre 3D dáta mračna bodov.

Hlavnou inováciou siete LU-Net je jej schopnosť efektívne spracovať 3D údaje LiDAR tak, že najprv extrahuje vysokoúrovňové 3D prvky zo surových mračien bodov a potom tieto prvky projektuje do 2D viackanálového diaľkového obrazu. Táto projekcia zohľadňuje špecifickú topológiu snímača LiDAR, vďaka čomu je tento prístup vysoko efektívny a účinný v aplikáciách v reálnom čase. Metóda preukázala výrazné zlepšenie oproti existujú-

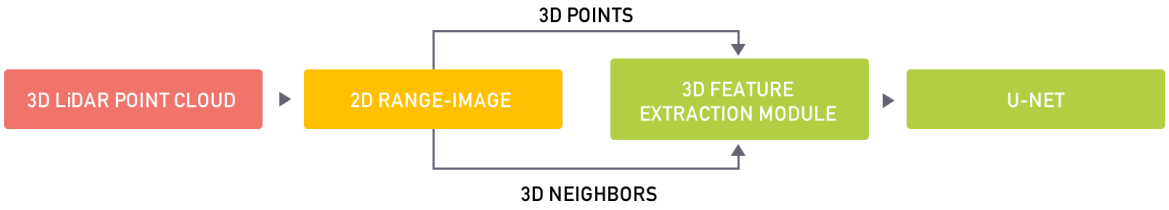
cim najmodernejším technikám na datasete KITTI, najmä pri spracovaní komplexného mestského prostredia, kde je presná a rýchla segmentácia prvoradá. [19]

Technológia LU-Net poskytuje robustné riešenie nielen na sémantickú segmentáciu v reálnom čase, ale otvára aj možnosti ďalšieho výskumu v oblasti integrácie LiDAR a techník spracovania obrazových údajov na zlepšenie vnímania autonómnych systémov. [19]

3.4.1 Popis architektúry LU-Net

V sieti LU-Net sa detekcia a identifikácia objektov v 3D mračnách bodov LiDAR dosahuje prostredníctvom podrobného procesu, ktorý zahŕňa transformáciu 3D údajov do štruktúrovaného 2D formátu a následnú efektívnu segmentáciu pomocou neurónových sietí.

LU-Net začína extrakciou vysokoúrovňových 3D prvkov z každého bodu v mračnách bodov LiDAR. To sa vykonáva pomocou modulu extrakcie 3D prvkov, ktorý zohľadňuje priestorové vzťahy medzi bodmi na základe ich fyzickej blízkosti. Na extrakciu vlastností, ktoré zachytávajú lokálnu 3D štruktúru údajov, sa spracúva každý bod a jeho susedné body.[19]



Obrázok 12. pracovný postup sémantickej segmentácie 3D mračien bodov, prevzaté z [19]

Obrázkok 12. znázorňuje navrhovaný pracovný postup sémantickej segmentácie 3D mračien bodov LiDAR. Na začiatku sa využíva topológia snímača na určenie 8 prepojených susedstiev v okolí každého bodu. Následne sa tieto body spolu s ich susednými bodmi spracujú prostredníctvom modulu na extrakciu 3D prvkov na vysokej úrovni. Tento modul konvertuje zozbierané údaje na viackanálový 2D diaľkový obraz. Tento obraz rozsahu sa potom použije ako vstup pre segmentačnú sieť založenú na U-Net. [19]

Tento modul využíva topológiu snímača LiDAR pri transformácii 3D mraku bodov na 2D viackanálový diaľkový obraz. Pri tejto transformácii sa body premietajú na základe ich uhlových a hĺbkových meraní voči senzoru, pričom sa zachováva priestorová organizácia vlastná snímaciemu vzoru LiDAR. Tento diaľkový obraz slúži ako štruktúrovaná reprezentácia, kde každý kanál zodpovedá rôznym modalitám údajov z mračna bodov, napríklad hĺbke alebo odrážavosti.

3.4.2 Architektúra U-Net na segmentáciu

Po vytvorení diaľkového obrazu LU-Net použije na úlohu sémantickej segmentácie modifikovanú architektúru U-Net. Architektúra U-Net použitá v LU-Net pozostáva zo štruktúry kodér-dekodér:

- **Kodér:** Kódovacia časť siete komprimuje diaľkový obraz, čím znižuje jeho dimenzionalitu za súčasného zvýšenia hĺbky reprezentácie príznakov. Táto časť siete je rozhodujúca pre extrakciu a konsolidáciu kontextových informácií z obrazu. [19]
- **Dekodér:** Dekódovacia časť siete potom pracuje na rozšírení reprezentácie príznakov na pôvodnú dimenzionalitu vstupného obrazu. Vynechanie spojení medzi vrstvami kodéra a zodpovedajúcimi vrstvami dekodéra pomáha zachovať jemné detaily opätovným začlenením prvkov stratených počas redukovania vzoriek. [19]

3.4.3 Zhrnutie

Dôležitým aspektom siete LU-Net je jej schopnosť spracovania v reálnom čase, čo je nevyhnutné pre aplikácie, ako je autonómne riadenie. LU-Net dosahuje tento cieľ spracovaním údajov rýchlosťou 24 snímok za sekundu na jednom GPU[19], čo zodpovedá rýchlosti snímania typických senzorov LiDAR používaných v automobilovom priemysle alebo ju prevyšuje.

Účinnosť LU-Net sa preukázala prostredníctvom rozsiahleho testovania na súbore údajov KITTI, ktorý je štandardným referenčným kritériom pre technológie autonómnych vozidiel. Výsledky prezentované v štúdii naznačujú, že LU-Net nielenže vykonáva sémantickú segmentáciu presnejšie ako predchádzajúce metódy, ale robí to s dostatočnou rýchlosťou pre aplikácie v reálnom čase. Metóda vyniká najmä v mestskom prostredí, kde je rozhodujúce presné a rýchle rozpoznávanie rôznych objektov - napríklad vozidiel, chodcov a cyklistov.

II. PRAKTICKÁ ČÁST

4 TESTOVANIE VÝKONNOSTI A DETEKCIE OBJEKTOV

Praktická časť baklárskej prace je zameraná na testovanie modelov spomínanych v teoretickej časti na zvolenom datasete. Z dôvodu veľkej popularity vo vedeckých publikáciách bol zvolený dataset COCO (Common Objects in Context) [21]. Mnoho dnes využívaných konvolučných neurónových sietí je trénovaných alebo minimálne testovaných na tomto datasete. Na základe toho sme sa rozhodli využiť Open-MMLab [22][24], ktorý je Open-source systém algoritmov počítačového videnia. Jeho dostupnosť a podrobná dokumentácia nám umožnila testovať už pred-trénované modely, bez potreby ich trénovať.

4.1 Príprava prostredia

Prvým krokom bola príprava prostredia pri ktorej sme postupovali podľa direktív z dokumentácie. Pre jednoduchšie nastavenie sme využili Anacondu [23], systém správy balíkov bežiaci aj v systéme Windows [24]. Nutné bolo dohľadať si balíčky kompatibilné s hardvérom. Testovanie prebiehalo na operačnom systéme Windows 10. Dotupné bolo 16GB operačnej pamäte a 8GB grafickej pamäte s použitím grafickej karty Nvidia GeForce RTX3070(laptop). Odporúčane bolo využiť aj virtuálne prostredie, aby nainštalované balíčky nespôsobili konflikt pri iných projektoch. MMDetection využíva PyTorch a na základe toho bolo nutné nainštalovať správnu verziu grafických ovládačov s týmto frameworkom, zároveň bolo potrebné zistiť akú verziu je potrebné využívať z balíkov od MMDetection [22]. Vhodnú verziu PyTorch [26] bola nainštalovaná pomocou nasledujúceho príkazu vo vytvorenom virtualnom prostredí:

```
conda install pytorch==2.1.2 torchvision==0.16.2 torchaudio==2.1.2 pytorch-
cuda=12.1 -c pytorch -c nvidia
```

Po tomto kroku bolo možné stiahnuť git repozitár príkazom:

```
pip install -r requirements/build.txt
pip install "git+https://github.com/open-mmlab/cocoapi.git#subdirectory=pycocotools"
pip install -v -e .
```

Po nainštalovaní všetkých potrebných balíčkov a prípadnom dolaďovaní podľa požiadaviek systému je nutne prejsť do zložky „mmdetection\mmdet“ v zložké config sú dostupné konfiguračné súbory pre modely na trénovanie alebo testovanie. Z Githubu je možné k niektorým týmto súborom stiahnuť predtrénovaný model napr. „nazov-modelu.pth“ ktorý

má všetky vlastnosti po ukončení tréningu. Tieto modely potom musia byť stiahnuté do zložky \mmdetection\mmdet\checkpoints

V koreňovej zložke „mmdetection“ sme museli vytvoriť zložku dáta\coco\annotations do ktorého sme umiestnili anotácie datasetu coco2017. V zložke coco je už potom nutné rozbaliť len validačný dataset val2017 v ktorom sa nachádza presne 5000 obrázkov formátu .jpg.

Niektoré zdroje odporúčali mierne odlišnú súborovú štruktúru, ale je potrebné skontrolovať súbor v zložke config/tools/test.py. V tomto súbore sa nachádza to ako pri testovaní toto prostredie začína testovaciu epochu.

4.2 Testovanie jednotlivých prístupov na datasete COCO

Výstup pri testovani nástrojmi od MMDtetection je formulovaný pomocou dvoch dôležitých metrík používaých na hodnotenie výkonnosti modelov a detekcie objektov. Priemerná presnosť (AP) a priemerná spätná väzba (AR). Tieto metriky sa počítajú pri rôznych konfiguráciách a prahových hodnotách Intersection over Union (IoU), veľkosti objektov a počtu detekcií (maxDets). Popis jednotlivých metrík je nasledovný:

1. **Priemerná presnosť (AP)**
    - AP [IoU=X | area=Y | maxDets=Z]: Ide o priemernú presnosť detekcií pri zadanej prahovej hodnote IoU alebo rozsahu, veľkosti oblasti objektu a maximálnom počte uvažovaných detekcií.
    - IoU (Intersection over Union): Meria sa tu prekrývanie predpovedaných ohraničujúcich boxov a ohraničujúcich boxov základnej pravdy. Hodnota IoU 0,50 znamená 50 % prekrytie. Vyššia hodnota IoU vyžaduje väčšiu presnosť predpovede ohraničujúcich boxov.
    - **Area:**
        - **All:** Všetky veľkosti.
        - **small, medium, large:** Špecifické kategórie veľkosti na základe plochy objektu v pixeloch. Prahové hodnoty pre tieto kategórie sa môžu líšiť v závislosti od súboru údajov.
    - **maxDets:** Maximálny počet detekcií objektov, ktoré sa berú do úvahy pri výpočte metriky. Napríklad **maxDets**=100 znamená, že metrika sa vypočíta s použitím 100 najlepších detekcií na obrázok.

2. **Priemerná spätná väzba (AR)**
    - **AR [IoU=X | area=Y | maxDets=Z]:** Táto hodnota meria schopnosť modelu odhaliť všetky relevantné objekty až do maximálneho počtu zadaných detekcií. Je menej citlivá na presnosť ohraničujúcich polí ako AP.
    - **IoU:** To isté ako pri AP, je to rozsah alebo jedna prahová hodnota.
    - **Plocha**: Definuje sa rovnako ako pri AP, pričom sa určuje veľkosť objektov.
    - **maxDets:** Udáva, koľko detekcií sa na jednom obrázku berie do úvahy. Vyššie hodnoty **maxDets** môžu poukazovať na to, ako dobre model deteku-je viacero objektov na obrázkoch.

Samotné testovanie sa spúšťa jednoduchým príkazom:

```
python  tools/test.py ${CONFIG_FILE} ${CHECKPOINT_FILE}
```

4.2.1 Faster-RCNN

Faster R-CNN výrazne zlepšuje detekciu objektov zavedením siete návrhu regiónu (RPN), ktorá zdieľa konvolučné funkcie v celom obraze so sieťou detekcie, čo umožňuje takmer úplne beznákladové návrhy regiónov. Takáto plne konvolučná sieť predpovedá hranice objektov aj skóre objektov na každom mieste.

Tabuľka 1 Výsledok testu - (AP) Faster-RCNN

Average Precision	IoU	Area	MaxDets
0.374	0.50:0.95	all	100
0.581	0.50	all	1000
0.404	0.75	all	1000
0.212	0.50:0.95	small	1000
0.409	0.50:0.95	medium	1000
0.481	0.50:0.95	large	1000

Tabuľka 2 Výsledok testu - (AR) Faster-RCNN

Average Recall	IoU	Area	MaxDets
0.517	0.50:0.95	all	100
0.517	0.50:0.95	all	1000
0.517	0.50:0.95	all	1000
0.326	0.50:0.95	small	1000
0.557	0.50:0.95	medium	1000
0.648	0.50:0.95	large	1000

4.2.2 RetinaNet

RetinaNet je jednostupňový detektor objektov navrhnutý tak, aby bol rýchly a presný a riešil problémy spojené s nerovnováhou tried počas trénovania. Využíva novú metódu Focal Loss, ktorá sa zameriava viac na objekty, ktoré sú čiastočne zakryté, sú v neprehľadných scénach alebo majú nezvyčajné tvary. Naopak sa nezameriava na pozadia a iné scény, kde sa dá ľahko klasifikovať, že neobsahujú žiadne objekty zájmu, čím výrazne zlepšuje výkonnosť detekcie. To umožňuje sieti RetinaNet vyrovnať sa rýchlosti predchádzajúcich jednostupňových modelov a zároveň dosiahnuť presnosť, ktorá prekonáva tradičné dvojstupňové detektory.

Tabuľka 3 Výsledok testu - (AP) RetinaNet

Average Precision	IoU	Area	MaxDets
0.365	0.50:0.95	all	100
0.554	0.50	all	1000
0.391	0.75	all	1000
0.204	0.50:0.95	small	1000
0.403	0.50:0.95	medium	1000
0.481	0.50:0.95	large	1000

Tabuľka 4 Výsledok testu - (AR) RerinaNet

Average Recall	IoU	Area	MaxDets
0.540	0.50:0.95	all	100
0.540	0.50:0.95	all	300
0.540	0.50:0.95	all	1000
0.336	0.50:0.95	small	1000
0.584	0.50:0.95	medium	1000
0.691	0.50:0.95	large	1000

4.2.3 RTMDet

RTMDet je pokročilý detektor objektov v reálnom čase, ktorý sa sústredí na dosiahnutie vysokého výkonu v úlohách rozpoznávania objektov, ako je segmentácia inštancií a detekcia otočených objektov. Využíva vyváženú architektúru s hĺbkovými konvoluciami s veľkým jadrom a integruje mäkké značky na dynamické prideľovanie značiek s cieľom zvýšiť presnosť. RTMDet je navrhnutý tak, aby bol vysoko efektívny a škálovateľný a ponúkal optimálny kompromis medzi parametrami a presnosťou pre rôzne aplikačné scenáre.

Tabuľka 5 Výsledok testu - (AP) RTMDet

Average Precision	IoU	Area	MaxDets
0.528	0.50:0.95	all	100
0.704	0.50	all	100
0.577	0.75	all	100
0.361	0.50:0.95	small	100
0.574	0.50:0.95	medium	100
0.692	0.50:0.95	large	100

Tabuľka 6 Výsledok testu - (AR) RTMDet

Average Recall	IoU	Area	MaxDets
0.394	0.50:0.95	all	1
0.652	0.50:0.95	all	10
0.703	0.50:0.95	all	100
0.540	0.50:0.95	small	100
0.756	0.50:0.95	medium	100
0.856	0.50:0.95	large	100

4.2.4 Analýza údajov o výkonnosti z testov

Analýza údajov o výkonnosti z testov troch modelov - RTMDet, RetinaNet a Faster-RCNN - odhaľuje významné rozdiely v ich účinnosti pri rôznych meraniach detekcie objektov. Hodnoty, na ktoré je poukázané sú v tabuľkách vyznačené tučným písmom.

Celková výkonnosť RTMDet vykazuje najvyšší výkon spomedzi troch modelov s AP 0,528 pri prahových hodnotách IoU od 0,50 do 0,95 vo všetkých oblastiach. Tento model je obzvlášť silný pri detekcii veľkých objektov s AP 0,692. Hodnoty spätnej väzby AR 0,703 pri prahových hodnotách IoU od 0,50 do 0,95 pre všetky oblasti s maximálnym počtom 100 detekcií, čo je najvyššia hodnota spomedzi všetkých modelov.

Celkový výkon RetinaNet s hodnotou AP 0,365 pri prahových hodnotách IoU od 0,50 do 0,95 vo všetkých oblastiach, čo je výrazne menej ako v prípade RTMDet. Má lepší výkon pri veľkých objektoch než pri malých, ale stále zaostáva za RTMDet s AP 0,481 pre veľké objekty. Hodnoty o spätnej väzbe zodpovedá jeho celkovému strednému výkonu s AR 0,540 pri prahových hodnotách IoU od 0,50 do 0,95 vo všetkých oblastiach.

Celkový výkon Faster-RCNN mierne vyšší ako RetinaNet s AP 0,374 pri prahových hodnotách IoU od 0,50 do 0,95 vo všetkých oblastiach. V prípade stredných a veľkých objektov dosahuje o niečo lepší výkon ako sieť RetinaNet s AP 0,409 pre stredne veľké objekty a 0,481 pre veľké objekty. Hodnoty spätnej väzby sú vyššie ako RetinaNet, dosahuje AR 0,517 pri rôznych prahových hodnotách IoU a horných hraniciach detekcie, čo naznačuje dobré celkové schopnosti detekcie objektov, ale nevyniká v presnosti.

4.2.5 Zhrnutie

RTMDet vyniká ako najschopnejší model z hľadiska presnosti aj spätnej väzby, čo je pozoruhodné najmä pri efektívnom spracovaní detekcie väčších objektov. RetinaNet aj Faster-RCNN predstavujú životaschopné možnosti so slušnými výkonmi, ale nedosahujú vysokú účinnosť, ktorú preukázal RTMDet, najmä pokiaľ ide o presnosť a spracovanie väčších objektov. Zdá sa, že sieť RetinaNet má väčšie problémy s menšími objektmi v porovnaní so sieťou Faster-RCNN, ktorá vykazuje vyrovnanejší výkon pri rôznych veľkostiach objektov, ale stále nedosahuje schopnosti siete RTMDet. Tieto údaje naznačujú, že pre aplikácie vyžadujúce vysokú presnosť a rýchlosť, najmä v scenároch s väčšími objektmi, by bola vhodnejšou voľbou sieť RTMDet.

ZÁVER

V tejto bakalárskej práci som sa zameral na prieskum súčasných metód detekcie a segmentácie 3D objektov v digitálnych obrazoch. Literárna rešerš bola vykonaná so zameraním na stručný popis existujúcich prístupov, pričom som sa snažil objasniť ich aplikácie a obmedzenia bez zbytočného prenikania do hĺbky technických detailov. Zámerom bolo poskytnúť pohľad do daných oblastí a poukázať aj na využívanie týchto techník v rôznych oblastiach. Tieto preskúmane metódy môžu ďalej slúžiť aj na inšpiráciu možnosti aplikácie v rôznych oblastiach každodenného života.

Praktická časť práce zahŕňala testovanie vybraných metód na datasete COCO, pričom som využil nástroje pre počítačové videnie voľne dostupné z internetu. Tie mi umožnili testovanie bez potreby rozsiahleho trénovania modelov. Výsledky týchto testov boli analyzované a porovnané, čo prispelo k objektívnemu hodnoteniu efektivity jednotlivých metód.

Záverečné odporúčania a zistenia z tejto práce poukazujú na potrebu ďalšieho výskumu v oblasti efektívneho spracovania 3D objektov z dvojrozmerných obrazov, s dôrazom na zlepšenie presnosti a zníženie výpočtovej náročnosti. Odporúča sa tiež skúmanie nových pokročilých techník hlbokého učenia, ktoré by mohli priniesť zlepšenie v detekcii a segmentácii objektov v reálnom čase.

Celkovo táto práca potvrdila základné princípy a preukázala praktickú aplikovateľnosť rôznych metód v počítačovom videní.

ZOZNAM POUŽITEJ LITERATÚRY

[1] Deep Learning for Generic Object Detection: A Survey, 2019. Online. Dostupné z: https://link.springer.com/article/10.1007/s11263-019-01247-4. [cit. 2024-02-28].

[2] SAVYAKHOSLA, 2023. CNN | Introduction to Pooling Layer. Online. Dostupné z: https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/. [cit. 2024-05-09].

[3] GEEKSFORGEEKS, 2024. Introduction to Convolution Neural Network. Online. Dostupné z: https://www.geeksforgeeks.org/introduction-convolution-neural-network/. [cit. 2024-05-09].

[4] PRAKASH, Jatin, 2021. Non Maximum Suppression: Theory and Implementation in PyTorch. Online. Dostupné z: https://learnopencv.com/non-maximum-suppression-theory-and-implementation-in-pytorch/. [cit. 2024-05-09].

[5] ROSEBROCK, Adrian, 2016. Intersection over Union (IoU) for object detection. Online. In: Pyimagesearch. Dostupné z: https://b2633864.smushcdn.com/2633864/wp-content/uploads/2016/09/iou_equation.png?lossy=2&strip=1&webp=1. [cit. 2024-05-09].

[6] LEPETIT, Vincent, 3D Scene Understanding from Images. In: DeepLearn 2022 Summer 6th International Gran Canaria School on Deep Learning.

[7] Nominal and observation-based attitude realization for precise orbit determination of the Jason satellites, 2019. Online. Dostupné z: https://www.researchgate.net/figure/llustrates-the-principle-of-gimbal-lock-The-outer-blue-frame-represents-the-x-axis-the_fig4_338835648. [cit. 2024-05-04].

[8] HUGHES, Mark, 2017. Don't Get Lost in Deep Space: Understanding Quaternions. Online. Dostupné z: https://www.allaboutcircuits.com/technical-articles/dont-get-lost-in-deep-space-understanding-quaternions/. [cit. 2024-05-09].

[9] LEKKALA, Dedeepya, 2023. From Pixels to 3D Shapes: An Overview of 3D Data Representations. Online. Dostupné z: https://medium.com/@deepyachowdary/from-pixels-to-3d-shapes-an-overview-of-3d-data-representations-90c57f85a900. [cit. 2024-05-09].

[10] REN, Shaoqing; HE, Kaiming; GIRSHICK, Ross a SUN, Jian, 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Online. 2015-06-04. Dostupné z: https://arxiv.org/pdf/1506.01497. [cit. 2024-05-10].

[11] GIRSHICK, Ross, 2015. Fast R-CNN. Online. Dostupné z: https://arxiv.org/pdf/1504.08083. [cit. 2024-05-10].

[12] LIN, Tsung-Yi; GOYAL, Priya; GIRSHICK, Ross; HE, Kaiming a DOLLAR, Piotr, 2018. Focal Loss for Dense Object Detection. Online. Dostupné z: https://arxiv.org/pdf/1708.02002. [cit. 2024-05-10].

[13] PAWANGFG, 2023. Residual Networks (ResNet) – Deep Learning. Online. 2023-01-10. Dostupné z: https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/. [cit. 2024-05-10].

[14] LYU, Chengqi; ZHANG, Wenwei; HUANG, Haian; ZHOU, Yue; WANG, Yudong et al., 2022. Https://arxiv.org/pdf/2212.07784#page=14&zoom=100,412,674. Online. Dostupné z: https://arxiv.org/pdf/2212.07784#page=14&zoom=100,412,674. [cit. 2024-05-10].

[15] R. S. Pahwa *et al*., "Automated Detection and Segmentation of HBMs in 3D X-ray Images using Semi-Supervised Deep Learning," *2022 IEEE 72nd Electronic Components and Technology Conference (ECTC)*, San Diego, CA, USA, 2022, pp. 1890-1897, doi: 10.1109/ECTC51906.2022.00297. [cit. 2024-04-09]

[16] AIKATERINI, Adam; KONSTANTINOS, Karantzalos; TORSTEN, Karantzalos a SATTLER, Torsten, 2023. Has Anything Changed? 3D Change Detection by 2D Segmentation Masks. Online. Dostupné z: https://arxiv.org/pdf/2312.01148. [cit. 2024-04-11].

[17] POGUDIN, Mikhail a Elena MEDVEDEVA. Two-stage algorithm for segmentation of satellite images. Kirov: Vyatka State University, VySU, 2022. Dostupné z: https://ieeexplore.ieee.org/document/9790776 [cit. 2024-04-09]

[18] 2023. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation. Dostupné z: https://cs.wikipedia.org/wiki/Markov%C5%AFv_%C5%99et%C4%9Bzec. [cit. 2024-04-09].

[19] BIASUTTI, Pierre, et al. Lu-net: An efficient network for 3d lidar point cloud semantic segmentation based on end-to-end-learned 3d features and u-net. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. 2019. p. 0-0.

[20] ADITYA_TAPARIA, 2023. U-Net Architecture Explained. Online. 2023-07-08. Dostupné z: https://www.geeksforgeeks.org/u-net-architecture-explained/. [cit. 2024-05-10].

[21] LIN, Tsung-Yi a kol. 2017. COCO: Common Objects in Context, včetně Testovacího datasetu [test2017.zip] a Informací o testovacích obrázcích [image_info_test2017.zip] [dataset]. Dostupné z: https://cocodataset.org/#download [cit. 2024-04-30].

[22] OpenMMLab. GitHub: Open MMLab [online]. GitHub. Dostupné z: https://github.com/open-mmlab [cit. 2024-05-10].

[23] Anaconda. Anaconda | The Operating System for AI [online]. Dostupné z: https://www.anaconda.com/ [cit. 2024-04-25].

[24] Microsoft. Windows 11 [online]. Dostupné z: https://www.microsoft.com/software-download/windows11 [cit. 2024-05-10].

[25] MMDetection. Instalace MMDetection [online]. Verze 2.1.0. Dostupné z: https://mmdetection.readthedocs.io/en/v2.1.0/install.html [cit. 2024-05-10].

[26] PyTorch. Začínáme s PyTorch: Předchozí verze [online]. Kompatibilní s MMdetection v2.1 a CUDA 12.1. Dostupné z: https://pytorch.org/get-started/previous-versions/ [cit. 2024-05-10].

[27] 2023. faster_rcnn_r50_fpnx_coco.pth: Predtrénovaný model na test [Fast R-CNN]. In: GitHub repository: MMDetection – faster_rcnn. GitHub. Dostupné z: https://download.openmmlab.com/mmdetection/v2.0/faster_rcnn/faster_rcnn_r50_fpn_mstrain_3x_coco/faster_rcnn_r50_fpn_mstrain_3x_coco_20210524_110822-e10bd31c.pth [cit. 2024-05-03].

[28] 2023. retinanet_r50_fpn_1x_coco_20200130-c2398f9e.pth: Pretrénovaný model na test[retinanet_r50_fpn_1x]. In: GitHub repository: MMDetection - retinanet. GitHub. Dostupné z: https://download.openmmlab.com/mmdetection/v2.0/retinanet/retinanet_r50_fpn_

mstrain_3x_coco/retinanet_r50_fpn_mstrain_3x_coco_20210718_220633-88476508.pth [cit. 2024-05-03].

[29] 2023. rtmdet_x_8xb32_300e_coco.pth: Predtrénovaný model na test [RTMDet-x]. In: GitHub repository: MMDetection - rtmdet. GitHub. Dostupné z: https://download.openmmlab.com/mmdetection/v3.0/rtmdet/rtmdet_x_8xb32-300e_coco/rtmdet_x_8xb32-300e_coco_20220715_230555-cc79b9ae.pth [cit. 2024-05-03].

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

CNN	Convolutional neural network
NMS	non-maximum Suppresion
YOLO	You Only Look Once
SSD	Single Shot MultiBox Detector
R-CNN	Regions with Convolutional Neural Network
LIDAR	Light Detection and Ranging
RPN	Region Proposal Network
ROI	Region of Interest
HBM	High Bandwidth Memory
XRM	X-ray Microscopy
SAM	Segment Anything Model
RGB-D	Red Green Blue - Depth
DHI	Digital Holographic Imaging
BBI	Bit Based Image
COCO	Common Objects in Context
GB	Gigabyte
AR	Average Recall
AP	Average Precision

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

SEZNAM PŘÍLOH