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: **26. července 2024**
Termín odevzdání bakalářské práce: **23. srpna 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 29. července 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 zaoberá detekciou a segmentáciou 3D objektov na obrázkoch, čo je základný aspekt počítačového videnia. Teoretická časť predstavuje základné koncepty a metódy, zatiaľ čo praktická časť sa zameriava na testovanie predtrénovaných modelov, ako sú 3DSSD, PointPillars a Cylinder3D, pomocou rámca MMDetection3D. Hoci tieto metódy nie sú najmodernejšie a experimenty sú obmedzené systémovými zdrojmi a určitými obmedzeniami, cieľom práce je poskytnúť prístupný úvod do témy a ukázať praktické aplikácie s dostupnými nástrojmi. Napriek niektorým nedokonalostiam možno výsledky replikovať pomocou poskytnutého kódu a pokynov na nastavenie prostredia.

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

ABSTRACT

This bachelor's thesis investigates the detection and segmentation of 3D objects in images, a fundamental aspect of computer vision. The theoretical section introduces essential concepts and methods, while the practical part focuses on testing pre-trained models like 3DSSD, PointPillars, and Cylinder3D using the MMDetection3D framework. Although these methods are not state-of-the-art, and the experiments are limited by system resources and certain constraints, the thesis aims to provide an accessible introduction to the topic and demonstrate practical applications with available tools. Despite some imperfections, the results can be replicated using the provided code and environment setup instructions.

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

Detekcia a segmentácia trojrozmerných objektov v obrazoch predstavuje dôležitú súčasť výskumu v oblasti počítačového videnia. Hoci sa v súčasnosti vyvíjajú a používajú pokročilé a sofistikované metódy, táto práca sa nesnaží implementovať najmodernejšie techniky z dôvodu systémových obmedzení. Jej cieľom je skôr poskytnúť úvod do tejto problematiky a ukázať, ako je možné využívať dostupné nástroje a metódy na detekciu a segmentáciu 3D objektov.

V teoretickej časti práce je podrobne predstavený kontext 3D detekcie a segmentácie, vrátane základných pojmov, metodológií a výziev, s ktorými sa tento výskum stretáva. Praktická časť sa zameriava na aplikáciu týchto poznatkov pomocou frameworku MMDetection3D, ktorý umožňuje využívanie predtrénovaných modelov bez potreby náročného tréningu od základov.

V rámci práce boli testované vybrané metódy, ako napríklad 3DSSD, PointPillars a Cylinder3D, ktoré sú prístupné v rámci MMDetection3D. Výsledky týchto testov sú obmedzené dostupnými systémovými zdrojmi a výpočtovým výkonom, čo ovplyvnilo rozsah a hĺbku vykonaných experimentov. Cieľom práce je tak čitateľa uviesť do problematiky 3D detekcie a segmentácie a demonštrovať, aké možnosti ponúka tento výskum pri použití dostupných nástrojov.

Je potrebné zdôrazniť, že pri správnom nastavení systému a využití priložených kódov by malo byť možné replikovať výsledky tejto práce. Prílohy obsahujú konfiguráciu prostredia bez datasetov z dôvodu ich veľkosti, avšak všetky potrebné kroky na ich znovuvytvorenie sú podrobne opísané.

I. TEORETICKÁ ČASŤ

1 TEORETICKÉ VÝCHODISKÁ A PRÍSTUPY

V oblasti detekcie a segmentácie 3D objektov zohrávajú kľúčovú úlohu technologické nástroje a vývojové prostredia, ktoré umožňujú efektívne spracovanie a analýzu dát. Správny výber týchto nástrojov je nevyhnutný pre úspešnú implementáciu a testovanie modelov, pretože rôzne knižnice a frameworky ponúkajú rôzne úrovne flexibility, výkonu a podpory pre špecifické úlohy.

Táto kapitola sa zameriava na predstavenie základných technológií a vývojových prostredí používaných v oblasti 3D detekcie a segmentácie objektov. Budú predstavené nástroje, ktoré boli vybrané pre účely tejto práce, a diskutované ich výhody a nevýhody, predovšetkým v kontexte kompatibility s operačným systémom Windows [1] a požiadaviek na spracovanie 3D dát.

1.1 Vývojové prostredie a platformové požiadavky

Práca je sústredená na testovanie a porovnanie existujúcich metód detekcie a segmentácie 3D objektov, ktoré sú spustiteľné v prostredí operačného systému Windows [1]. Základom je knižnica strojového učenia PyTorch [2] používaná pre aplikácie v počítačovom videní.

Pre účely tejto práce sú vybrané technológie a nástroje, ktoré sú kompatibilné s Windows a zároveň podporujú pokročilé spracovanie 3D dát.

1.1.1 Prehľad dostupných technologií

MMDetection3D [3] je rozšírenie frameworku **MMDetection[4]**, primárne zameraného na 2D detekciu objektov, s podporou pre 3D objekty. Využíva PyTorch[2] a podporuje viaceré moderné architektúry ako napríklad PointPillars[5], SECOND[6]. Framework poskytuje rozsiahlu modularitu, ktorá umožňuje prispôsobenie modelov a efektívnu prácu s rôznymi druhmi senzorových dát, vrátane LiDAR a stereo kamier. Nevýhodou môže byť jeho vyššia náročnosť na konfiguráciu a potreba dodatočných závislostí pre inštaláciu v prostredí Windows.

Open3D je knižnica určená na manipuláciu a spracovanie 3D dát, so zameraním na bodové mraky a sieťové reprezentácie. Poskytuje intuitívne API, ktoré uľahčuje spracovanie 3D dát, a obsahuje funkcie na ich vizualizáciu. Aj keď Open3D obsahuje základné metódy pre spracovanie a manipuláciu s dátami, jeho využitie v pokročilých úlohách, ako je 3D detekcia

a segmentácia, je obmedzené. Pre pokročilejšie scenáre môže byť potrebné doplnenie o ďalšie knižnice. [7]

Point Cloud Library (PCL) je robustná open-source knižnica na spracovanie a analýzu 3D bodových mrakov. Obsahuje širokú paletu algoritmov pre segmentáciu, registráciu, filtráciu a rekonstrukciu povrchov. PCL je vhodná pre aplikácie v robotike a autonómnych systémoch, kde je potrebné efektívne spracovanie dát z 3D senzorov, ako sú LiDAR a RGB-D kamery. PCL je napísaná v C++ a integrácia v Python prostredí môže byť náročnejšia, čo zvyšuje komplexnosť práce s touto knižnicou. [8]

PyTorch3D je rozšírenie PyTorch frameworku, ktoré umožňuje efektívnu prácu s 3D dátami v rámci neurónových sietí. Je navrhnutý na výpočty v GPU, s podporou diferenciálnych operácií na 3D dátach, čo umožňuje ľahšiu integráciu 3D objektov do trénovacieho procesu hlbokých neurónových sietí. Knižnica poskytuje nástroje na manipuláciu s 3D bodovými mrakmi, povrchovými sieťami a inými formátmi 3D dát, pričom sa zameriava na rýchlosť a efektívnosť výpočtov. Je vhodná pre pokročilé scenáre, no vyžaduje znalosti v oblasti 3D výpočtovej geometrie. [9]

Detectron2 s rozšírením **Mesh R-CNN** je framework, ktorý umožňuje kombinovať 2D a 3D detekčné techniky. Detectron2 je navrhnutý pre rozpoznávanie a klasifikáciu objektov na 2D obrázkoch, zatiaľ čo Mesh R-CNN pridáva schopnosť rekonštruovať povrchy a generovať 3D mriežky z obrazových dát. Tento prístup je vhodný pre úlohy ako 3D rekonštrukcia objektov z viacerých pohľadov. Nevýhodou je zložité nastavenie a vyššia výpočtová náročnosť, čo môže vyžadovať výkonnejší hardvér. [10][11]

Kaolin je knižnica od NVIDIA, určená pre deep learning s 3D dátami, optimalizovaná pre výpočty na GPU. Podporuje rôzne formáty 3D dát a poskytuje efektívne nástroje na trénovanie neurónových sietí pre 3D úlohy, ako sú detekcia a segmentácia. Je navrhnutý na prácu s rozsiahlymi 3D datasetmi a je priamo prepojený s CUDA na zvyšovanie výpočtového výkonu. Kvôli svojej špecifickosti a náročnejšej štruktúre môže byť vhodný pre pokročilých používateľov. [12]

1.2 Výber technológií pre splnenie cieľa práce

Pri riešení problému 3D detekcie objektov v rámci tejto bakalárskej práce bol zvolený framework MMDetection3D (MMdet3D) [3]. MMdet3D je významným nástrojom vývojárov počítačového videnia, ktorý je Open-source. Pre túto prácu je vybraný

predovšetkým pre jeho širokú podporu rôznych algoritmov 3D detekcie a segmentácie. Kľúčovým faktorom voľby tohto nástroja pre túto prácu bola jeho experimentálna podpora pre operačný systém Windows, ktorá umožňuje využívať tento pokročilý nástroj na dostupnom hardvéri.

Prednosťami MMdet3D sú jeho robustné riešenia pre 3D detekciu a segmentáciu, vrátane podpory pre rôzne typy dát a modelov. Zatiaľ čo mnohé pokročilé nástroje pre počítačové videnie sú primárne vyvíjané pre systémy Unix/Linux, MMdet3D sa snaží o rozšírenie svojej dostupnosti aj pre Windows[1] užívateľov.

Tento výber bol realizovaný na základe niekoľkých kľúčových faktorov, ktoré z neho robia vhodnú voľbu pre účely tejto práce, aj pre niekoho s obmedzenými skúsenosťami v oblasti 3D počítačového videnia. MMDetection3D poskytuje predpripravené prostredie, ktoré obsahuje všetky potrebné komponenty na tréning a vyhodnocovanie modelov. Jeho integrácia je zložitejšia, kvôli experimentalnej podpore pre platformu Windows. Dôvodov prečo bol vybraný je viac:

- **Rozsiahla knižnica modelov:** MMDetection3D [3] ponúka širokú škálu predtrénovaných modelov na detekciu 3D objektov. Obsahuje osvedčené architektúry, ktoré sú často využívané vo vedeckom výskume a priemyselných aplikáciách. Obsahuje predpripravené modely, ktoré je možné testovať bez nutnosti začínať od nuly, čo je veľkou výhodou.
- **Predpripravené experimentálne prostredie:** Framework ponúka komplexné prostredie, ktoré obsahuje všetky potrebné nástroje na trénovanie, vyhodnocovanie a ladenie modelov. Táto integrácia umožňuje zamerať sa priamo na cieľ práce a nezaoberať sa všetkými implementačnými detailmi.
- **Aktívna komunita a dostupná dokumentácia:** MMDetection3D [3] je podporovaný aktívnou komunitou, ktorá neustále prispieva k jeho vývoju a zlepšovaniu. Dostupnosť dokumentácie, ktorá obsahuje detailné tutoriály a príklady, je kľúčová pre študentov, ktorí sa potrebujú rýchlo oboznámiť s novými technológiami a metodikami.

1.2.1 Voľba datasetu

V tejto práci bude tento nástroj využitý najmä na testovanie pred-trénovanych modelov pre 3D detekciu a segmentáciu a vyhodnocovanie ich presnosti na zvolenom datasete KITTI[13] a Semantic KITTI[14]. Datasety boli zvolené pre ich popularitu a dostupnosť, no aj relatívne nižšiu náročnosť na úložisko oproti iným datasetom. Detaily zvolenia datasetu sú popísane v kapitole 4. Výber datasetu.

1.3 Metodika práce

V tejto práci využijeme prostredie MMDetection3D, ktoré nám umožní nielen implementovať a testovať rôzne metódy detekcie a segmentácie 3D objektov, ale aj vizualizovať výsledky na základe toho, ako bol model trénovaný. MMDetection3D poskytuje integrované nástroje na vykonávanie experimentov s predpripravenými modelmi a váhami, čo nám umožní okamžite nasadiť trénované modely a analyzovať ich výkonnosť na vybraných datasetoch.

Pomocou tohto prostredia môžeme nielen spustiť testy na rôznych modeloch, ale aj vizualizovať výsledky, napríklad zobraziť detekované objekty v 3D priestore či posúdiť presnosť segmentácie objektov. Prostredie tiež umožňuje prispôsobovať a vizualizovať rôzne aspekty modelu, ako sú bounding boxy, 3D body a segmentačné masky, čo nám poskytne hlbší náhľad do fungovania modelu v závislosti od toho, ako bol trénovaný. Tento prístup nám umožní lepšie porozumieť, ako sa modely vyrovnávajú s reálnymi dátami, a zhodnotiť ich schopnosť riešiť kľúčové problémy pri detekcii a segmentácii 3D objektov.

2 DETEKCIA A SEGMENTÁCIA OBJEKTU V OBRAZE

Detekcia objektov v obraze je kľúčovým prvkom počítačového videnia, umožňujúcim rozpoznanie a lokalizáciu objektov na digitálnych obrázkoch alebo videách. Tento proces zahŕňa identifikáciu objektov rôznych kategórií, ako sú ľudia, budovy, vozidlá a iné, a ich presnú polohu v rámci obrazu. V posledných rokoch sa v tejto oblasti dosiahol významný pokrok vďaka technikám hlbokého učenia, najmä použitím konvolučných neurónových sietí (CNN), ktoré sa ukázali ako mimoriadne účinné pri detekcii objektov.

Pri detekcii 2D objektov sa využívajú pokročilé modely CNN, ktoré prechádzajú rôznymi vrstvami spracovania obrazu, aby extrahovali dôležité vlastnosti a následne identifikovali a lokalizovali objekty. Tieto metódy sú schopné efektívne rozpoznávať objekty v rôznych prostrediach a podmienkach, čo ich robí veľmi užitočnými pre široké spektrum aplikácií.

Zatiaľ čo detekcia 2D objektov poskytuje cenné informácie o obsahu obrazu, je obmedzená na rovinu obrazu a nezohľadňuje hĺbku a priestorové vlastnosti objektov. Pri mnohých aplikáciách, ako sú autonómne vozidlá, robotika a rozšírená realita, je však nevyhnutné vedieť nielen to, čo sa v obraze nachádza, ale aj kde presne sa to nachádza v trojrozmernom priestore.

Detekcia 3D objektov rozširuje možnosti 2D detekcie tým, že integruje hĺbkové informácie, čo poskytuje komplexnejší pohľad na scénu. Tento prístup zahŕňa niekoľko ďalších krokov a techník, ktoré umožňujú presné určenie polohy objektov v priestore.

Jedným z najvýraznejších rozdielov medzi 2D a 3D detekciou je množstvo dát, ktoré je potrebné spracovať. V 2D detekcii pracujeme s dvoma dimenziami (šírka a výška), pričom každý pixel predstavuje určitú hodnotu jasu alebo farby. Na druhej strane, 3D detekcia zahŕňa aj tretiu dimenziu (hĺbku), alebo aj ďašie vlastnosti, čo exponenciálne zvyšuje množstvo údajov, ktoré je potrebné analyzovať.

2.1 Vymedzenie pojmov Detekcie a Segmentácie 3D objektov v obraze

Detekcia 3D objektov je proces identifikácie a lokalizácie objektov v 3D priestore na základe obrazových údajov. Tým, že zohľadnňuje hĺbkové údaje umožňuje presnejšiu identifikáciu polohy a tvaru objektov. Detekcia 3D objektov zahŕňa procesy, ktoré umožňujú systému rozpoznať a určiť presnú polohu objektov v trojrozmernom priestore. To zahŕňa identifikáciu objektu, jeho lokalizáciu a často aj určenie jeho orientácie. Existuje mnoho metód na detekciu 3D objektov, moderné metódy využívajú hlboké učenie. Medzi

najpoužívanejšie techniky patria konvolučné neurónové siete (CNN) a ich varianty. Detekcia 3D objektov je kľúčová pre aplikácie, ako sú autonómne vozidlá, ktoré potrebujú presne rozpoznať a lokalizovať prekážky na ceste, robotika, kde je potrebné interagovať s okolitým prostredím, a rozšírená realita, ktorá pridáva virtuálne objekty do reálneho sveta. [15]

Segmentácia 3D objektov je proces rozdelenia obrazu na zmysluplné časti alebo objekty. Táto technika umožňuje presnú analýzu tvaru a štruktúry objektov v obraze a je často používaná na identifikáciu a analýzu jednotlivých objektov v komplexných scénach. Segmentácia objektov sa zameriava na rozdelenie obrazu do regiónov, ktoré zodpovedajú jednotlivým objektom alebo významným častiam objektov. Na rozdiel od detekcie, ktorá len identifikuje a lokalizuje objekty, segmentácia poskytuje podrobnejšiu analýzu tvaru a štruktúry. Segmentácia sa dosahuje pomocou rôznych techník najmodernejšie pokročilé metódy založené na hlbokom učení, ako sú plne konvolučné siete (FCN), U-Net a Lu-Net[16]. Segmentácia je využitelná v rôznych oblastiach, kde je potrebné presne identifikovať a analyzovať rôzne štruktúry. Týmto príkladom môže byť virtuálna realita, kde segmentácia umožňuje interakciu s virtuálnymi objektmi, a analýza scény, ktorá zlepšuje pochopenie a interpretáciu zložitých prostredí.[15]

2.2 Informácie na interpretáciu objektu

Pri detekcii 3D objektov vstupuje do hry niekoľko kritických faktorov a úvah, ktoré zásadne menia prístup k detekcii a interpretácii objektov v priestorovom prostredí. Aby sme mohli efektívne detekovať 3D objekty, musíme najprv správne interpretovať informácie o tom ako je objekt v priestore umiestnený.

2.2.1 Získavanie informácií o objektoch

K tomuto účelu sa používajú rôzne typy informácií o hĺbke. Existuje niekoľko spôsobov, ako tieto informácie získať. Medzi najčastejšie používané spôsoby patria:

- Stereovízia: Používa dve kamery umiestnené vedľa seba na zachytenie dvoch obrázkov z mierne odlišných uhlov pohľadu. Porovnaním týchto obrázkov a identifikáciou zodpovedajúcich bodov je možné vypočítať hĺbku každého bodu v obraze.
- Štruktúrované svetlo: Projekcia známeho vzoru svetla na scénu. Deformácie tohto vzoru spôsobené objektmi v scéne umožňujú výpočet hĺbky.

- LiDAR (Light Detection and Ranging): Vysiela laserové pulzy a meria čas, ktorý trvá, kým sa pulz odrazí od objektu a vráti späť. Táto technológia je známa svojou vysokou presnosťou pri meraní vzdialeností a často sa používa v autonómnych vozidlách.
- Monokulárne kamery: Používajú jednu kameru a algoritmy hlbokého učenia na odhad hĺbky z jedného obrazu. Tento prístup je náročnejší na presnosť, ale je lacnejší.

Získané hĺbkové informácie poskytujú základ pre ďalší krok, ktorým je reprezentácia týchto údajov v počítači.

2.2.2 Reprezentácia zozbieraných informácií

Reprezentácia 3D údajov je kľúčovým aspektom, ktorý ovplyvňuje celý proces detekcie. Výber vhodnej reprezentácie závisí od použitého senzorového systému a konkrétnych požiadaviek aplikácie. Spôsobov ako dáta reprezentovať môže byť viac ako je v tejto práci uvedené, práca sa ale zameriava na bežne používané reprezentácie. Tými sú:

- 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. Mračná bodov poskytujú vysokú úroveň detailov a sú často používané v aplikáciách, kde je potrebná presná lokalizácia objektov. Matematicky by sa mračno bodov P dalo reprezentovať reprezentovať ako:

$$P = \{p_1, p_2, p_3, \ldots, p_n\} \tag{2.1}$$

kde každý bod p_i je vektor v 3D priestore:

$$p_i = \{x_i, y_i, z_i\} \tag{2.2}$$

V niektorých prípadoch, tieto body môžu obsahovať aj atribúty ako napríklad farba alebo intenzita:

$$p_i = \{x_i, y_i\ z_i,\ r_i, g_i, b_i, I_i\} \tag{2.3}$$

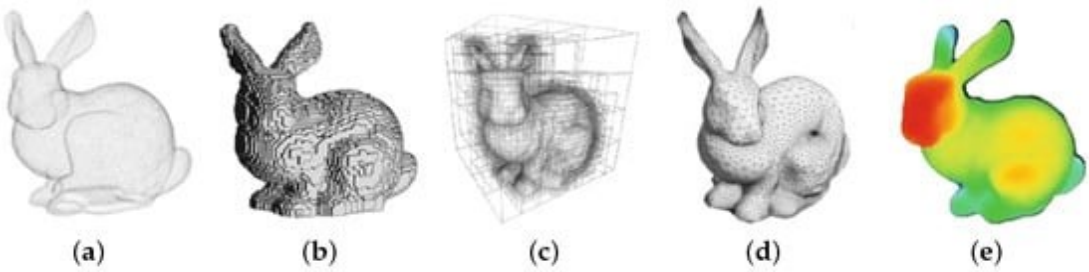
Kde r_i, g_i, b_i reprezentuje farby vo formate RGB a I_i reprezentuje intenzitu.
Mračná bodov bývajú riedke a nemajú pravidelnú štruktúru. Tento typ dát poskytuje vysokú úroveň detailu a presnosti, ale zťažuje priame spracovanie štandardnými algoritmami strojového učenia[17].

- Voxely: 3D údaje reprezentujú takým spôsobom, pri ktorom je 3D priestor rozdelený na pravidelnú mriežku malých „kociek“, známych ako voxely. Každý voxel obsahuje informacie o časti priestoru, ktorú reprezentuje, ako napríklad obsadenosť, farba a iné [20]. Reprezentácia voxelovej mriežky o veľosti 32 x 32 x 32, kde súradnice voxelu sú *i, j, k* s hodnotou $f_{i,j,k}$ reprezentujúcu vlastnosti uložené vo voxely a obsahujú hodnoty ako obsadenosť, farba alebo intenzita.[18]
- Reprezentované možu byť nasledovne:

$$V=\{v_{i,j,k} \mid i,j,k \in \{1,2,\ldots,32\}, f_{i,j,k}\} \tag{2.4}$$

- Octree - dátová štruktúra, ktorá efektívne reprezentuje trojrozmerné priestory delením na menšie segmenty. V kontexte 3D údajov je celý priestor rozdelený na osem častí, každá známa ako oktant. Každý oktant môže byť rekurzívne delený na ďalšie oktanty, až kým sa nedosiahne požadovaná úroveň detailu alebo presnosti [19].
- Siete sú súbory 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. Táto reprezentácia je často používaná v počítačovej grafike a pri modelovaní objektov[20].
- Obrázky z viacerých pohľadov: Tento prístup používa na odvodzovanie 3D informácií z viacero 2D snímkov nasnímaných z rôznych uhlov prostredníctvom triangulácie alebo modelov hlbokého učenia vycvičených na lepšie pochopenie priestorových vzťahov. Táto metóda je obzvlášť užitočná v situáciách, kde sú k dispozícii viaceré kamery alebo mobilné zariadenia [21].

- HÍbkové mapy: Poskytujú informácie o vzdialenosti každého bodu od kamery. Tieto mapy môžu byť generované napríklad stereo kamerami alebo inými senzormi. HÍbkové mapy sú často kombinované s RGB obrázkami na zlepšenie presnosti a poskytovanie farebných informácií. [21]



Obrázek 1 - Model reprezentovaný rôznym typom dát - (a) Mračno bodov, (b) Voxely, (c) Octree, (d) Siete, (e) Hlbkové mapy (prevzaté z [22])

3 NEURONOVÉ SIETE

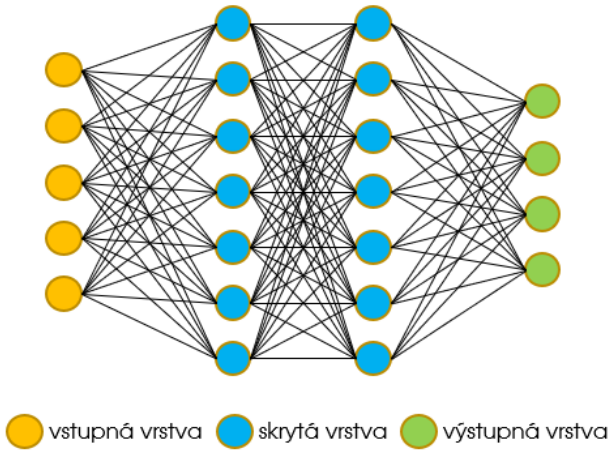
Keďže ako najpopulárnešími a najmodernejšími metodámi na detekciu 3D objektov sa používaju konvolučné neuronové siete je nutné vysveliť základne pojmy s nimi spojené. Základnym prvkom týchto sietí je umelý neurón.

Do umelého neurónu vstupujú váhy a prah, ktoré sa počas trénovania modelu prispôsobujú. Neurón obsahuje vnútorný potenciál aktivačnej funkcie a prenosovú funkciu. Váhy sú parametre v neurónovej sieti, ktoré určujú dôležitosť jednotlivých vstupov pri výpočte výstupu. Váhy transformujú vstupné dáta pomocou aktivačnej funkcie. Ich ulohou je prispôsobenie sa počas trénovania, umožňujúc modelu zachytiť vzory v dátach. Prah je adaptívny parameter, ktorý umožňuje modelu posunúť rozhodovaciu hranicu v priestore vstupov, čím sa pridáva flexibilita pri učení rôznych vzorov. Pomáha modelu zachytiť vzory, ktoré by inak neprešli. Matematická reprezentácia aktivačnej funkcie neurónu je popísaná rovnicou 2.5.

$$a = \sum_{i=1}^{n+1} x_i w_i \tag{2.5}$$

kde a predstavuje aktivačnú funkciu. w je vektor váh pre vstupy x je vektor vstupných hodnôt.

Pre detekciu 3D objektov v obraze sa používaju neuronové siete, ktoré sa skladajú z niekoľkých vrstiev, kde každá zohráva svoju úlohu a posúva výsledky spracovania ďalšej vrstve.



Obrázek 2 - Neurónová sieť [23]

3.1 Konvolučné neuronové siete

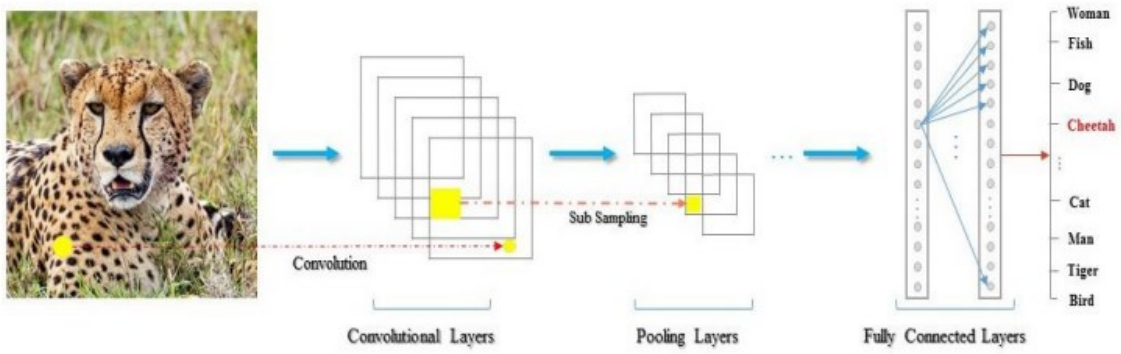
Konvolučná neurónová sieť (CNN) je typom umelej neurónovej siete a predstavuje dôležitý nástroj pre učenie pod dohľadom. CNN sú navrhnuté tak, aby automaticky extrahovali a identifikovali dôležité vlastnosti v dátach pomocou konvolučných operácií, ktoré simulujú fungovanie biologických neurónov v mozgu. Architektúra CNN sa skladá z niekoľkých kľúčových vrstiev. Konvolučné vrstvy slúžia na identifikáciu rôznych vzorov v obraze, ako sú hrany, textúry či geometrické tvary, a to pomocou filtrov aplikovaných na obrazové dáta. Pooling vrstvy (napr. Max-Pooling) následne znižujú dimenzionalitu mapy vlastností, čím minimalizujú výpočtovú náročnosť, pričom zachovávajú najdôležitejšie informácie. Nakoniec plne prepojené vrstvy sú zodpovedné za konečnú klasifikáciu alebo regresiu. Pre zvýšenie robustnosti siete sa často používajú techniky ako dropout, ktorý redukuje pretrénovanie, a dávková normalizácia, ktorá zrýchľuje tréning a stabilizuje učenie.[18]

3.1.1 Architektúra CNN

Typická architektúra CNN sa skladá z nasledujúcich vrstiev:

1. **Konvolučné vrstvy:** Tieto vrstvy sú zodpovedné za extrakciu rôznych vlastností z obrazových dát, ako sú hrany, textúry a základné geometrické tvary. Každá konvolučná vrstva používa filtre (alebo jadra), ktoré sa aplikujú na vstupný obraz. Výsledkom je mapa vlastností (feature map), ktorá zachytáva prítomnosť špecifických vzorov v obraze.[18]
2. **Pooling vrstvy:** Pooling (zväčša Max Pooling) znižuje rozlíšenie mapy vlastností tým, že vyberá najvýznamnejšie hodnoty v malých blokoch obrazu. Pooling pomáha zvyšovať robustnosť siete voči malým transformáciám, ako sú posuny alebo šumy v obraze.[18]
3. **Aktivačné funkcie:** Po každej konvolučnej vrstve sa často používa aktivačná funkcia, ako napríklad ReLU (Rectified Linear Unit), ktorá nelineárne transformuje vstupy tak, aby zachovala iba kladné hodnoty. ReLU zlepšuje rýchlosť učenia a zabraňuje saturácii gradientov.[18]
4. **Plne prepojené vrstvy:** Tieto vrstvy na konci CNN sa používajú na zhrnutie informácií z predchádzajúcich vrstiev a produkujú konečné predikcie, napríklad klasifikáciu objektov. V prípade 3D detekcie a segmentácie sa plne prepojené vrstvy

často používajú na výstupné hodnoty, ako sú súradnice objektov alebo masky segmentácie.[18]



Obrázek 3 - Architektúra CNN [24]

Pre spracovanie dát, ktoré obsahujú viac dimenzií sa využíva 3D CNN, alebo iné architektúry prispôsobene na účely spracovania priestorových dát. Rozdiel spočíva v jej schopnosti vykonávať konvolúcie v troch dimenziách (hĺbka, výška a šírka).

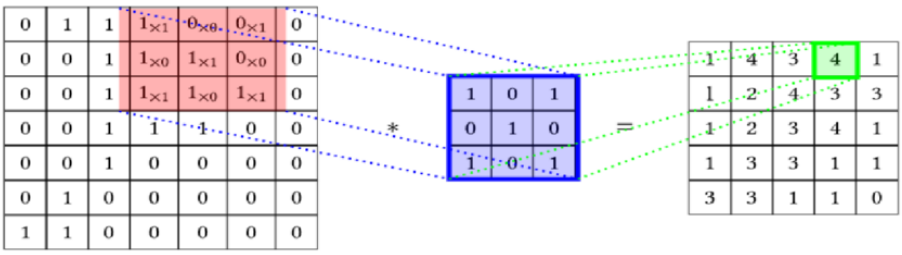
3.2 Popis operacií vykonávanych CNN

Ako je popísane vyššie, dáta ktoré vstupujú do CNN sa spracuvávajú vo rôznych vrstvách. Tieto operácie sa vykonávajú v nasledujúcom poradí:

1. Konvolúcia – Lineárna operácia na vstupný obraz alebo dáta.
2. Aktivačná Funkcia – Aplikácia nelinearity na výstup z konvolučnej vrstvy.
3. Pooling – Zníženie priestorovej veľkosti dát.
4. (Opakovanie týchto krokov pre ďalšie vrstvy).

Konvolúcie sú matematická operácia, ktorá sa vykonáva v konvolučnej vrstve. Na získanie základnych identifikačných prvkov v obrázku ako su hrany a základne tvary. Operacia je reprezentovaná rovnicou (3.1), kde I je vstupná matica (obraz), K je filter, (x, y) sú súradnice vstupného pixelu a (m, n) sú súradnice filtra. [26]

$$(I * K)(x, y) = \sum_{m} \sum_{n} I(x + m, y + n) \cdot K(m, n) \tag{3.1}$$



Obrázek 4 - Konvolúcia [25]

Pri spracovaní objemových dát sa využíva 3D konvolúcia, ktorá funguje na podobne len je rozšírena o jednu dimenziu. Túto operáciu je možne reprezentovať nasledujúcim predpisom (2.6), kde I je vstupná 3 rozmerná matica a K je 3 filter, (x, y, z) sú súradnice vstupných dát a (i, j, k) sú súradnice filtra. [27]

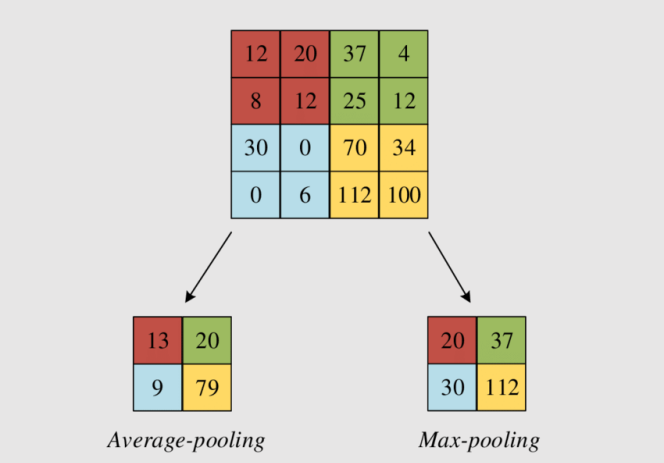
$$(I * K)(x, y, z) = \sum_{i} \sum_{j} \sum_{k} I(x + i, y + j, z + k) \cdot K(i, j, k) \tag{3.2}$$

Ďalšími parametrami pri konvolúcii sú parametre padding a stride:

- Padding pridáva extra pixely (často nuly) okolo okrajov obrazu, aby sa zachovala veľkosť výstupu rovnaká ako vstupu, čo je kľúčové pre zachovanie priestorových informácií pri hlbokých sieťach.
- Stride (krok) určuje, o koľko pixelov sa filter posúva po obraze; vyššie hodnoty stride vedú k menším výstupným mapám vlastností a znižujú výpočtovú náročnosť, avšak za cenu straty detailov.

Aktivačné funkcie majú zásadný vplyv na výstup neurónov v sieti, pretože ich úlohou je pridať nelinearitu do modelu, čo umožňuje neurónovej sieti zvládať komplexné problémy, ktoré nie sú lineárne oddeliteľné.

Pooling sa používa na zníženie rozmerov dát, čím sa zachovávajú dôležité informácie a znižuje sa výpočtová náročnosť modelu. Na základe preskúmanej literatúry sa najčastejšie využíva Max Pooling a Average Pooling. Ak definujeme veľkosť filtra a krok o koľko sa bude po obrazku posúvať. Obrázok 4. znázorňuje filter veľkosti 2x2 a krok 2. Max-pooling vyberie najväčšiu hodnotu z oblasti, kde sa filter nachádza. [28]



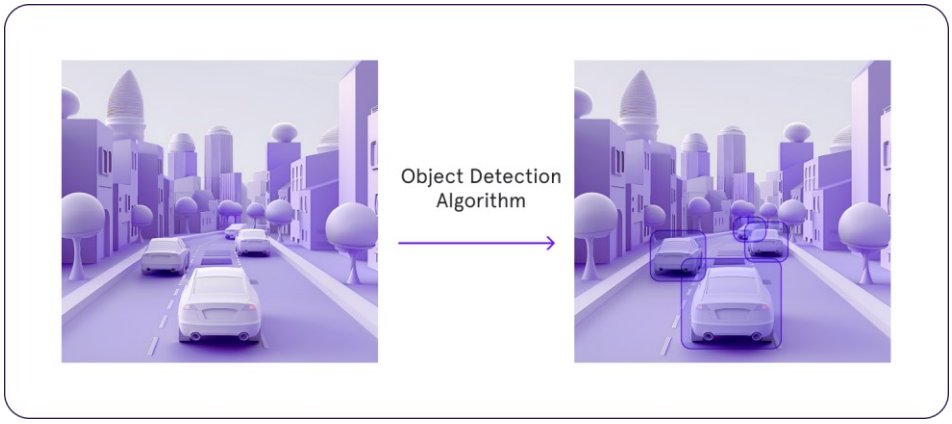
Obrázek 5 - Pooling s filtrom 2x2 a krokom 2 [28]

3.3 Proces detekcie 3D objektu

Na uvedenie do problematiky do detekcie by bolo vhodné použiť príklad. Povedzme, že chceme detegovať objekt. K jeho detekcií môžeme mať dostupné rôzne dáta napríklad mračna bodov alebo obrázky z viacerých pohľadov. Môžeme teda povedať, že máme nejaký typ záznamu a „odpoveď“, ktorú od konvolučnej neuronovej siete očakávame. Ak do tejto neuronovej siete dáme na vstup dáta, od ktorých očakávame odpoveď hrnček a vizuálnu reprezentáciu toho kde sa nachádza(3D box) na základe určenia jeho polohy, Túto detekciu popisuje na základe týchto krokov:

- Prvým krokom je zber dát a následuje vstup do prvej vrstvy. Z týchto dát sa odstraňuje šum. Zabezpečí sa konzistentnosť dát. Napríklad ak by sme mali skupinu bodov: P_1{0.1,1.2,3.4}, P_2{-0.5, 0.3, 2.1}, P_3{4.5, 3.2, 8.7}, P_4{0.2, 1.1, 3.5}. Bod P_3 je viditeľne príliš ďaleko od ostatných a je možné ho odstrániť ako šum.
- V druhom kroku sa extrahujú dôležité vlastnosti, ktoré daný objekt pomáhaju identifikovať. Týmito dátami pri mračnách bodov môžu byť súradnice bodu a vektory kolmé na povrch v každom bode. Vytvorí sa mapa vlastností.
- Tretím krokom je detekcia na základe priame spracovanie mračna bodov. Prípadne sa data voxelizujú alebo konvertujú na taký typ dát, s ktorým model dokáže pracovať. Napríklad metóda PointPillars [5], konvertuje mračna bodov na pravidelné mriežky alebo stĺpce, čím vykoná „pilarizáciu“.
- V štvrtom kroku prichádza na rad predikcia detekcií objektov. Tento krok zahŕňa identifikáciu a lokalizáciu objektov v 3D priestore.

Počet krokov sa pre každú metódu líši, preto je tento popis len orientačný, aby vystihol podstatu daného subjektu.

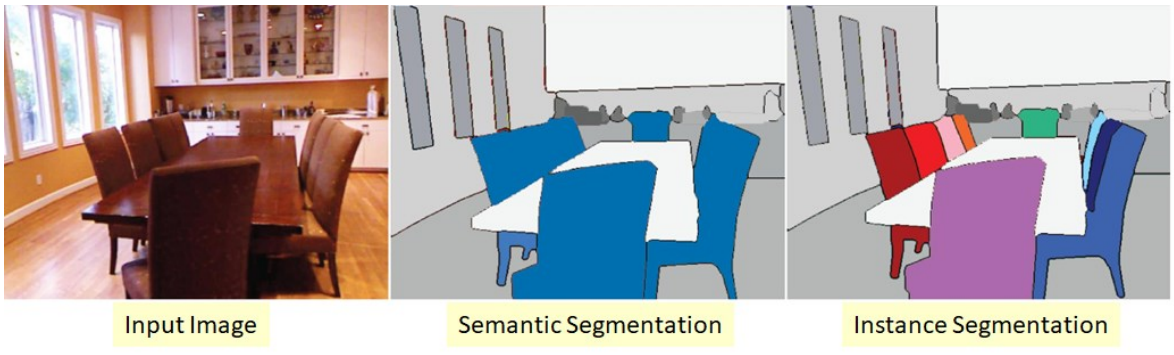


Obrázek 6 - Ilustračný obrázok detekcie objektu [21]

3.4 Segmentácia 3D objektov v obraze

Segmentácia v kontexte počítačového videnia a strojového učenia sa vzťahuje na proces rozdelenia obrazu alebo 3D mračna bodov na odlišné oblasti alebo segmenty, pričom každý segment predstavuje iný objekt alebo časť objektu. Proces segmentácie objektov zahŕňa niekoľko krokov a metód, ktoré sa môžu líšiť v závislosti od konkrétnej aplikácie a typu údajov. Vo všeobecnosti sa segmentácia delí na dve hlavné kategórie:

- Sémantická segmentácia: Cieľom tejto segmentácie je priradiť každému pixelu na obrázku triedu. Napríklad každý pixel, ktorý patrí autu, bude označený ako "auto". Nezohľadňuje jednotlivé inštancie objektov, ale len ich triedy.
- Segmentácia inštancií: Segmentácia inštancií rozlišuje medzi rôznymi inštanciami objektov. To znamená, že každý objekt (napr. každé auto) bude označený samostatne, aj keď patrí do rovnakej triedy.



Obrázek 7 - Ilustračný obrázok segmentácie [29]

3.4.1 Proces segmentácie 3D objektu

- Prvým krokom je zber dát a následuje vstup do prvej konvolučnej vrstvy. Rovnako ako pri detekcií.
- V druhom kroku sa extrahujú dôležité vlastnosti, ktoré daný objekt pomáhaju identifikovať. Týmito dátami pri mračnách bodov môžu byť súradnice bodu a vektory kolmé na povrch v každom bode. Pointnet nato využivá Multilayer Perceptron, čo je sada matematických operácií, ktoré transformujú vstupy na výstupy pomocou neurónov. Ako aktivačná funkcia sa využíva ReLU, a v jednoduchosti táto funkcia nastaví všetky záporne hodnoty na nulu a kladné ponechá. Je popísana rovnicou 2.6.

$$\mathrm{ReLU}(z) = \max(0, z) \tag{2.6}$$

- Tretím krokom je agregácia globálných vlastností. V tomto kroku sa všetky extrahované vlastnosti z jednotlivých bodov zlúčia do jedného globálneho vektora. Je možné si to predstaviť ako zozbieranie najdôležitejších informácií z každého bodu na vytvorenie celkovej reprezentácie objektu.
- V štvrtom kroku prichádza na rad detekcia a klasifikácia objektov. Do vrstiev na tento účel vstupuje globálny vektor, ktorý prechádza cez sériu plne prepojených vrstiev, ktoré transformujú tento vektor do predikcií tried. Každá vrstva aplikuje váhy a biasy na vektor a upravuje ho na základe naučených vzorov počas trénovania.

Po prechode týmito vrstvami je potrebné výsledné skóre pre jednotlivé triedy previesť na pravdepodobnosti, ktoré reprezentujú, do akej miery patrí objekt k danej triede. Tento krok je nevyhnutný, aby bolo možné interpretovať výstupy modelu ako pravdepodobnosti. Na normalizáciu výstupov do pravdepodobností pre rôzne triedy objektov sa využíva aktivačná funkcia softmax.

$$softmax(z_j) = \frac{e^{z_j}}{\Sigma_k e^{z_k}} \tag{2.7}$$

Kde z_j je skóre pre každú triedu a softmax zabezpečuje, že súčet pravdepodobností pre všetky triedy je 1.

Model potom lokalizuje objekty pomocou regresie ohraničujúcich boxov, ktoré sa používaju na ohraničenie detegovaného objektu. Regresia 3D ohraničujúcich boxov je proces, ktorý umožňuje modelu presne určiť rozmery, polohu a orientáciu boxu, ktorý obklopuje

detegovaný objekt. Je definovaný súradnicami x,y,z dvoch protiľahlých rohov. Okrem polohy a veľkosti môže 3D ohraničujúci box obsahovať aj informácie o orientácii, čo je často dôležité pre správne umiestnenie objektov v priestore. [22]

3.4.2 Voxelizácia mračna bodov

Ďalší prístup k segmentácii zahŕňa konverziu mračna bodov na voxelovú sieť, ktorá je základom metódy VoxNet. Voxelizácia transformuje 3D údaje na pravidelnú 3D mriežku, kde každý voxel (3D ekvivalent pixelu) obsahuje binárnu informáciu alebo informáciu o hustote prítomnosti bodov. Táto reprezentácia umožňuje použitie trojrozmerných konvolučných neurónových sietí (3D CNN), ktoré sú určené na extrakciu priestorových vlastností z údajov mriežky.

3D CNN používajú konvolučné vrstvy, ktoré na vstupné údaje aplikujú filtre na extrakciu lokálnych vzorov. Každý konvolučný filter sa pohybuje po vstupnej mriežke a vytvára mapu príznakov, ktorá odráža vzory v rôznych častiach objektu. Tento prístup umožňuje modelu identifikovať komplexné priestorové vzory potrebné na segmentáciu. Následné vrstvy združovania znižujú dimenzionalitu údajov a zachytávajú najdôležitejšie prvky, čo modelu umožňuje efektívne spracovať veľké objemy údajov.[18]

3.4.3 Grafové neurónové siete

Grafové neurónové siete (GNN) predstavujú inovatívny prístup k segmentácii, najmä v prípade neštruktúrovaných 3D údajov. V tomto modeli sú body reprezentované ako vrcholy grafu a hrany medzi nimi definujú ich vzájomné vzťahy, často založené na vzdialenosti, napríklad k-najbližších susedov (KNN). GNN iteratívne aktualizujú vlastnosti vrcholov na základe informácií zo susedných vrcholov, čo im umožňuje zachytiť komplexné priestorové vzťahy.

Tento prístup využíva rekurentné prechody grafov, kde každá iterácia aktualizuje vrcholové vlastnosti pomocou lineárnych transformácií a nelineárnych aktivačných funkcií, ako je napríklad ReLU. Aktualizované funkcie sa kombinujú s pôvodnými a novými informáciami od susedov, čo modelu poskytuje schopnosť naučiť sa podrobné interakcie medzi bodmi. Dynamické grafové konvolučné neurónové siete (DGCNN) sú príkladom takýchto sietí, ktoré adaptívne aktualizujú graf na základe vstupných údajov.[30]

3.4.4 Metódy zohľadňujúce prístupy z viacerých pohľadov

Viacpohľadové prístupy kombinujú informácie z viacerých 2D pohľadov na objekt s cieľom vytvoriť komplexnejšiu 3D reprezentáciu. Každý pohľad sa spracúva pomocou 2D CNN, ktorá extrahuje vizuálne vlastnosti z jednotlivých obrazov. Tieto vlastnosti sa potom kombinujú pomocou operácií združovania alebo iných metód fúzie, čo umožňuje vytvoriť globálnu reprezentáciu objektu.

CNN s viacerými pohľadmi integruje tieto techniky použitím konvolúcií na každý pohľad, extrahovaním lokálnych vzorov a následným zjednotením informácií z viacerých uhlov, čím sa zvyšuje presnosť segmentácie. Táto metóda je obzvlášť užitočná v aplikáciách, kde je k dispozícii viacero pohľadov na objekt, napríklad v autonómnych vozidlách alebo robotike. [31]

3.4.5 Octree a hierarchické štruktúry

Pri práci s veľkými mrakmi bodov sú veľmi užitočné hierarchické štruktúry, ako napríklad oktree. Octree rozdeľuje 3D priestor na oktanty, čo umožňuje efektívne prideľovanie pamäte a rýchle vyhľadávanie údajov na rôznych úrovniach podrobnosti. Táto štruktúra je výhodná pri spracovaní veľkých súborov údajov, kde je potrebné zachovať podrobné informácie v hustých oblastiach a efektívne spracovať riedke časti.

OctNet využíva octree na ukladanie a spracovanie údajov, pričom aplikuje CNN na hierarchicky usporiadané voxely. Tento prístup umožňuje dynamicky meniť úroveň detailu podľa potreby, čo je najmä užitočné v aplikáciách, kde sa vyžaduje presnosť v určitých oblastiach, ale efektívnosť v iných. [32]

3.4.6 Hybridné metódy

Hybridné metódy kombinujú viaceré prístupy a využívajú ich výhody na zvýšenie presnosti a účinnosti segmentácie. Napríklad kombinácia PointNet s CNN s viacerými pohľadmi môže využiť lokálne geometrické vzory a globálne informácie z viacerých pohľadov, čo vedie ku komplexnej reprezentácii objektu. Tieto metódy umožňujú flexibilitu pri návrhu modelu a poskytujú riešenia špecifických výziev spojených so segmentáciou 3D objektov, ako sú heterogénne údaje alebo potreba zachovať detailnú presnosť v špecifických oblastiach.

Každý z týchto prístupov ponúka rôzne výhody a výzvy, pričom výber metódy závisí od konkrétnej aplikácie a požiadaviek na presnosť, efektívnosť a výpočtových prostriedkov.

4 PROBLÉMY PRI DETEKCII A SEGMENTACII 3D OBJEKTOV

Pri detekcií a segmentácií objektov môže nastať veľa situácií, ktoré tento proces značne komplikujú. Táto časť sa zameriava na poukázanie detailov a rozdielov, možné riešenia týchto problémov na datasete KITTI, venuje sa primárne mračnu bodov a 2D obrázkom, ktoré daný dataset poskytuje. Problémy, ktoré sa vyskytujú sú popísane v nasledujúcich podkapitolách.

4.1 Oklúzie a neporiadok

V reálnych scenároch objekty zriedka existujú izolovane. Zvyčajne sú čiastočne alebo úplne zakryté inými objektmi. Prípadne je v scéne priveľa objektov, ktoré sú blízko seba. Takisto môže problém spôsobiť aj uhol pohľadu, pretože z iného uhla objekt vyzerá inak.

4.2 Variabilita vzhľadu a veľkostí objektov

Variabilita vzhľadu a veľkosti objektov predstavuje významný problém pri detekcií 3D objektov. Objekty môžu vyzerať inak v závislosti od uhla pohľadu, osvetlenia a inherentných rozdielov v rámci rovnakej kategórie. Tento problém komplikuje presnú detekciu a klasifikáciu, pretože detekčné systémy musia byť schopné spoľahlivo rozpoznať objekty, ktoré sa môžu výrazne líšiť vo vzhľade a veľkosti.

4.3 Definovanie polohy objektu v obraze

Na definovanie polohy objektu v priestore je možné využiť 6D pózu. 6D póza objektu je reprezentovaná kombináciou jeho 3D polohy (translácie) a 3D orientácie (rotácie). Tento koncept zahŕňa tri dimenzie pre polohu a tri dimenzie pre orientáciu. Popísať sa dá nasledovne:

- **3D poloha (translácia)**: Označuje sa ako vektor $T=[T1,T2,T3]T$, ktorý predstavuje polohu objektu vzhľadom na referenčný bod, zvyčajne kameru alebo globálny súradnicový systém. [33] MATURANA
- **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. [33] MATURANA

Pri práci s 3D objektmi je dôležité nielen ich detekovať a segmentovať, ale aj správne reprezentovať ich orientáciu v priestore. Pre túto úlohu sa často využívajú rôzne matematické nástroje a metódy, ktoré umožňujú presnú manipuláciu s rotáciami objektov.

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

- **Strata pri translácii:** Meria presnosť predpovedanej 3D polohy objektu. 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). [33]

$$L_T = \left|\left|T - \hat{T}\right|\right|^2 \tag{3.1}$$

- **Strata otáčaním:** Meria presnosť predpovedanej orientácie objektu. Zisťuje sa, či je objekt otočený v smere hodinových ručičiek: Reprezentácia a porovnávanie orientácií (rotácií) je zložitejšie ako polohy vzhľadom na vlastnosti priestorov rotácií. [33]

  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). [33]

$$L_R = ||log\left(R\hat{R}^T\right)||_F = \cos^{-1}\left(\text{tr}\left(R\hat{R}^T\right) - 1\right) / 2 \tag{3.2}$$

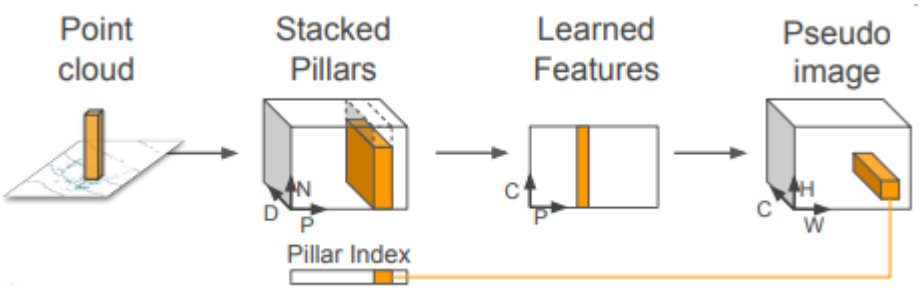
Celková strata polohy*(L_{pose})* kombinuje straty translácie a rotácie, prípadne s váhovým faktorom*(γ)* na vyváženie ich príspevkov, reprezentovaná vzťahom (1.3). [33] LEPETIT

$$L_{pose} = L_T . + \gamma L_T . \tag{3.3}$$

Tento štruktúrovaný prístup k výpočtu strát umožňuje modelom určiť detegovať 3D objekt na základe 2D obrázku. [33]

4.4 Problémy pri detekcii mračna bodov

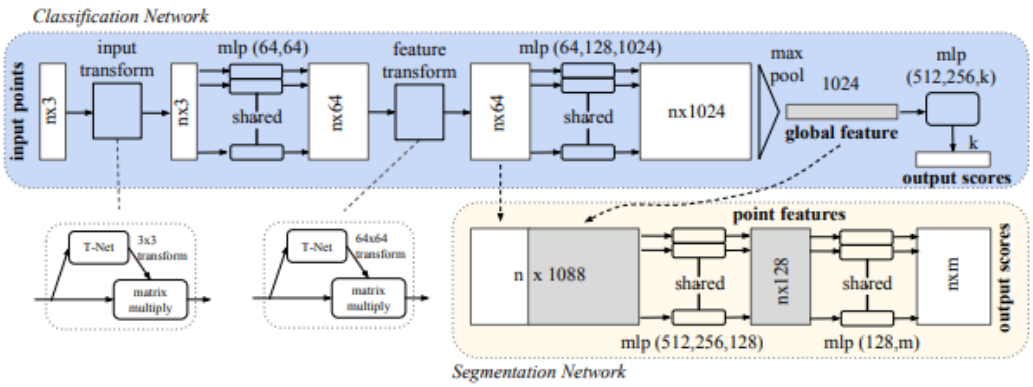
Mračná bodov nemajú pravidelnú štruktúru ako napr. 2D obrázky, čo štandardným konvolučným neurónovým sieťam (CNN) sťažuje ich spracovanie. Modely ako PointNet [17] a PointPillars spracovať neusporiadané mračná bodov a vytiahnuť vlastnosti priamo z mračien bodov. PointNet používa symetrické funkcie na získanie globálnych funkcií, ktoré nezávislé od poradia bodov vyhodnotia výsledok vždy rovnako, zatiaľ čo napríklad PointPillars[5] premieňa body na vertikálne „piliere“, aby mohli rýchlejšie extrahovať vlastnosti týchto pilierov.



Obrázek 9 - Konverzia mračna bodov na piliere[5]

Na to aby Pointnet[17] mohol spracovať mračná bodov a správne vyhodnotiť o aký objekt sa jedná, model musí byť invariantný voči N permutáciám, aby ich vstup nezávisel na poradí bodov, ktoré do siete vstupujú. Pretože body môžu byť pri vstupe rôzne usporiadané ale stále môžu znázorňovať rovnaký objekt, znázornené v rovnici (3.4). Model by mal byť taktiež invariantný aj voči geometrickým transformáciám, aby sme boli schopný rozpoznať objekt aj bez ohľadu na orientaciu, či polohu v scéne.

$$f(x_1, x_2, \dots x_n) = f\left(x_{\sigma(1)}, x_{\sigma(2)} \dots x_{\sigma(2)}\right) \tag{3.4}$$



Obrázek 10 – Architektúra siete PointNet, n označuje počet bodov, k označuje počet objektov, m označuje počet segmentov [17]

Na Obrázku 3 je vidieť architektúru siete Pointnet kde n je počet bodov, ktoré vstupujú do siete a číslo 3 predstavuje priestorové informácie x, y, z pre každý bod. T-Net znazorňuje transformačnú sieť predtrénovanú na predpovedanie transformačnej matice na mapovanie mračna bodov do kanonického tvaru. Tieto vlastnosti sa znovu vynasobia maticou 3x3, aby zvýšila robustnosť siete. Výsledkom celeho procesu na obrázku je skóre pravdepodobnosti klasifikácie objektu. Mnoho inovatívnych metód je inšpirovaných práve pointnet architektúrou. [17]

Ďalším veľkým problémom je šum a body, ktoré môžu byť príliš odľahlé. Pri svetelnom zázname pomocou LiDAR to môžu byť aj odrazy od okien a podobne. V architektúre pointnet tento problém rieši viacvrstvový perceptron (MLP), ktorý transformuje jeden bod a transformuje jeho vlastnosti do viac dimenzionálneho priestoru. Na druhej strane VoxelNet[34] to rieši efektívnym prevodom mračna bodov na voxely a použitím 3D konvolúcií na spracovanie priestorových informácií, čo umožňuje rýchlejšie spracovanie bez straty detailov.

PointPillars používa pohľad z vtáčej perspektívy na efektívne zachytenie obrysov objektov a tréning modelov s údajmi z viacerých uhlov môže túto flexibilitu zlepšiť. Byť úplne odolný voči všetkým zmenám uhla pohľadu je však stále ťažké.

Spracovanie mračien bodov znamená aj riešenie vysokých výpočtových nárokov, pretože môžu mať milióny bodov. Čo sťažuje aplikácie v reálnom čase, kde je kľúčová rýchlosť.

5 VÝBER DATASETU

V oblasti detekcie a segmentácie 3D objektov je dostupných mnoho datasetov, ktorými sa výskumnici zaoberajú:

- nuScenes – Multimodálny dataset pre autonómne riadenie, obsahuje dáta zo šiestich kamier, štyroch radárov a jedného LiDAR senzora.[35]
- Waymo Open Dataset – Dataset pre autonómne riadenie s vysoko kvalitnými dátami z piatich kamier a štyroch LiDAR senzorov.[36]
- S3DIS – Dataset zameraný na 3D segmentáciu vnútorných priestorov, vrátane kancelárií a chodieb.[37]
- SemanticKITTI – Rozšírenie KITTI s anotáciami pre semantickú segmentáciu bodových mračien.[38]
- ApolloScape – Dataset pre autonómne riadenie s 3D anotáciami, semantickou segmentáciou a sledovaním pohybu objektov.[39]
- Cityscapes – Dataset zameraný na semantickú segmentáciu mestských prostredí, obsahuje vysoko kvalitné 2D obrázky.[40]

Pre túto prácu bol vybraný dataset KITTI [13]na testovanie a porovnanie metód na detekciu a Semantic KITTI [38] na segmentáciu 3D objektov. Dataset je široko podporovaný v rámci MMDetection3D, ktorý poskytuje viacero predtrénovaných modelov optimalizovaných pre tento dataset. Predtrénované váhy umožňujú efektívnu implementáciu a testovanie rôznych prístupov bez potreby rozsiahleho trénovania modelov.

Datasety sú štruktúrované tak, aby ponúkal komplexnú škálu typov údajov relevantných pre úlohy autonómneho riadenia a 3D vnímania. Týmito údajmi sú:

- Kalibračné súbory – slúžia na zarovnanie dát medzi senzormi, konkrétne kamerou a LiDAR-om. Obsahujú projekčné a transformačné matice, ktoré zabezpečujú správnu projekciu 3D bodov na 2D obraz. Tento proces je nevyhnutný pre správnu synchronizáciu multimodálnych dát a presné umiestnenie objektov v priestore.
- Anotácie – poskytujú informácie o objektoch v scéne, ako sú vozidlá, chodci a cyklisti. Obsahujú 2D ohraničujúce boxy pre obrazy a 3D ohraničujúce boxy pre bodové mračná. Tieto anotácie sú využívané na trénovanie a testovanie modelov pre detekciu objektov v 2D a 3D priestore, umožňujú priestorové určenie objektov a presnosť ich rozpoznávania.

- Obrázky z kamery – sú záznamy z prednej ľavej kamery, ktoré sú použité na detekciu objektov v 2D priestore. Tieto obrázky poskytujú vizuálny kontext a v kombinácii s LiDAR dátami zlepšujú presnosť detekcie objektov a ich segmentáciu v scéne.
- LiDAR bodové mračná – slúžia na priestorovú detekciu objektov v 3D prostredí. Obsahujú súradnice (x, y, z) a intenzitu odrazu, čo umožňuje presné určenie tvaru, vzdialenosti a polohy objektov. Tieto dáta poskytujú kľúčovú hĺbkovú informáciu potrebnú pre efektívnu segmentáciu a analýzu v 3D prostredí.

Celkovo možno konštatovať, že vďaka štruktúrovanej podpore rôznych typov údajov, údajom z mračna bodov a integrácii so zvoleným prostredím sú zvolené datasety praktickou a efektívnou voľbou pre túto prácu.

5.1 Hodnotiace metriky vybraných datasetov

Hodnotenie výkonnosti 3D modelu detekcie objektov je kľúčovým krokom pri posudzovaní jeho presnosti a spoľahlivosti. Je vhodneé preto uviesť metriky, ktoré tento dataset využíva. Výkonnosť modelu detekcie 3D objektov sa zvyčajne zaznamenáva pomocou súboru metrík, ktoré poskytujú komplexné pochopenie jeho silných a slabých stránok.

Všeobecne využívanými metrikami v oblasti detekcie objektov je miera odozvy (recall rate), ktorá udáva schopnosť algoritmu správne identifikovať pozitívne prípady. Matematicky je definovaná ako:

$$Recall = \frac{TP}{\mathrm{TP} + \mathrm{FN}} \tag{5.1}$$

Kde TP (True Positive) predstavuje počet správne identifikovaných pozitívnych prípadov a

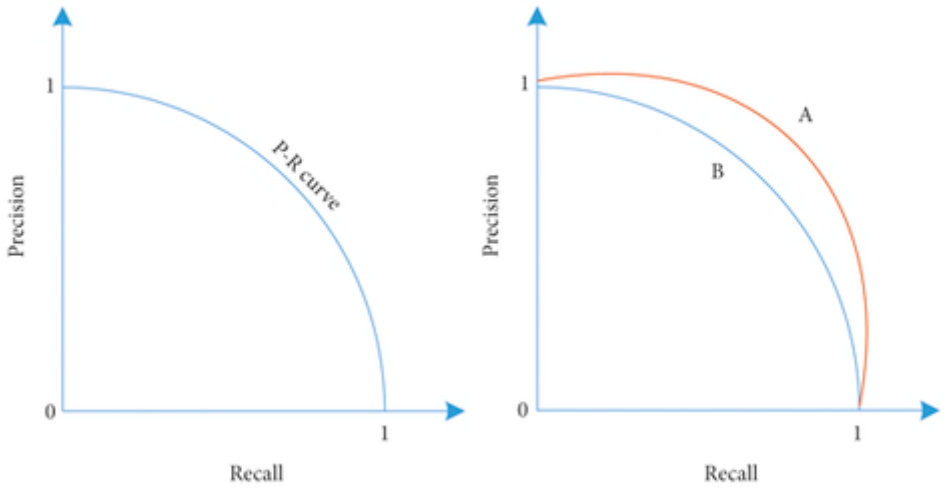
FN (False Negative) predstavuje počet pozitívnych prípadov nesprávne identifikovaných ako negatívne. [41]

Presnosť meria presnosť algoritmu pri predpovedaní pozitívnych detekciách a je daná vzťahom:

$$Presnosť = \frac{TP}{\mathrm{TP} + \mathrm{FN}} \tag{5.1}$$

kde

FP (False Positive) označuje počet negatívnych prípadov nesprávne identifikovaných ako pozitívne.Tieto dve metriky, odvolanie a presnosť, hodnotia výkonnosť algoritmu z rôznych hľadísk a často sú v nepriamom vzťahu. Na ich vyváženie sa používa krivka presnosti a odozvy (P-R). [41]



Obrázek 11 - Krivka P-R [41]

Z krivky P-R možno odvodiť priemernú presnosť (AP). Pre spojitú krivku P-R sa AP vypočíta takto:

$$AP = \int_0^1 p(r)\, d(r) \tag{5.2}$$

$$AP = \sum_{k=1}^{N} p(k)\, r(k) \tag{5.3}$$

Priemerná priemerná presnosť (mAP) je priemerom AP vo viacerých kategóriách na meranie celkovej presnosti algoritmu. Vypočíta sa ako:

$$mAP = \frac{\left(\sum_{i=1}^{C} Ap_i\right)}{C} \tag{5.3}$$

Pri detekcii objektov sa účinnosť algoritmu posudzuje predovšetkým prostredníctvom kvalitatívnych a kvantitatívnych hodnotení. Kvalitatívne hodnotenie sa opiera o pozorovanie a je vo svojej podstate subjektívne. Naproti tomu kvantitatívne hodnotenie využíva matematickú štatistiku na meranie výkonnosti algoritmu pomocou špecifických metrík. Na rozdiel od kvalitatívnych metód umožňuje kvantitatívne hodnotenie vedeckejšie,

spravodlivejšie a presnejšie porovnanie rôznych algoritmov. Presnosť sa meria pri nasledujúcich vlastnostiach modelu:

- Bbox (Bounding Box Average Precision): Meria presnosť predpovedí 2D ohraničujúceho boxu pre zistené objekty na obrázku.
- BEV (Bird's Eye View Average Precision - priemerná presnosť pohľadu z vtáčej perspektívy): Hodnotí výkonnosť detekcie objektov pri premietaní na scénu z vtáčej perspektívy zhora nadol.
- 3D AP (3D priemerná presnosť): Hodnotí presnosť predpovedí 3D ohraničenia, pričom hodnotí schopnosť modelu predpovedať presné umiestnenie, veľkosť a orientáciu objektov v trojrozmernom priestore.
- AOS (priemerná podobnosť orientácie): Hodnotí presnosť predpovedanej orientácie objektov, ktorá vyjadruje, ako dobre model odhaduje smer, ktorým sú objekty otočené.

Tieto metriky sú prevzate z oficiálnej stránky datasetu.[13], každá z nich poskytuje pohľad na iné schopnosti modelu. V praktickej časti tejto práce sa tieto metriky používajú na hodnotenie výkonnosti detekčných modelov testovaných na datasete KITTI.

II. PRAKTICKÁ ČÁST

6 ÚVOD DO PRAKTICKEJ ČASTI

Praktická časť tejto bakalárskej práce sa zameriava na testovanie modelov pre detekciu a segmentáciu 3D objektov. Hlavným cieľom je aplikovať teoretické znalosti získané v predchádzajúcich kapitolách na reálne dátové súbory a vyhodnotiť výkonnosť zvolených metód a spôsob akým k detekcii či segmentácií pristupujú. Prvým, krokom bolo nastavenie prostredia, MMDetection 3D.

6.1 Príprava prostredia

Na prostredie pre detekciu a segmentáciu bol vybraný Open-source systém správy prostredia Conda. Na prácu s Mmdetection 3D, bol potrebný Python verzie 3.8 a Pytorch s podporou pre CUDA toolkit verzie 11.8 vhodný pre testovací hardware. Ostatné knižnice boli inštalované podľa požiadaviek samotného MMDetection 3D zo súboru requirements.txt.

6.1.1 Nastavenie prostredia MMDetection 3D

Pre nastavenie prostredia MMDetection 3D bol použitý nástroj Anaconda a programovací jazyk Python verzie 3.8. Testovacie prostredie bolo stiahnuté z príslušného github repozitára /odkaz/. Ostatné podporné balíčky a drivery boli doinštalované na základe dokumentácie MMDetection 3D z oficiálnej stránky. Nezahrňuje však chyby, ktorým môže uživateľ čeliť. Repozitár bol stiahnutý z githubu:

```
Conda create -n mmdet3d python=3.8

Conda activate mmdet3d && cd mmdet3d

Git clone https://github.com/open-mmlab/mmdetection3d.git
```

Po úspešnom spustení prostredia sú prístupne nástroje pre testovanie, trénovanie a vizualizáciu dát, ktoré sú použité v tejto práci. Z dôvodu experimentálnej podpory, pri práci boli zistené nedostatky a dané prostredie nespĺňa všetky funkcionality. Pretože modely boli trénovane s rôznymi konfigurácimi a váhy teda nieje možné upraviť bez fine-tuningu alebo tréningu siete odznova.

6.1.2 Vizualizácia

Pomocou MMDet3D, v prostredí windows je možné vizualizovať dáta pomocou lokálneho vizualizéra „Det3DLocalVisualizer“ v prostredí windows je nutné vytvoriť premennu pre

zobrazovacie zariadenie, ktoré je v systéme linux automatický identifikované „Conda\envs\mmdet3d_env\Lib\site-packages\mmdet3d\visualization\local_visualizer.py“.

```
34      import open3d as o3d
35      from open3d import geometry
36      from open3d.visualization import Visualizer
37  except ImportError:
38      o3d = geometry = Visualizer = None
39
40  os.environ['DISPLAY'] = '1' # Display identificator
41
42  @VISUALIZERS.register_module()
43  class Det3DLocalVisualizer(DetLocalVisualizer):
44      """MMDetection3D Local Visualizer.
45
46      - 3D detection and segmentation drawing methods
47
```

Obrázek 12 - Nastavenie zobrazovacieho média v local_visualizer.py

Vizualizáciu detekčného modelu je potom možné spustiť príkazom:

python tools/test.py konfiguracny_subor.py váhy.pth --show --task lidar_det

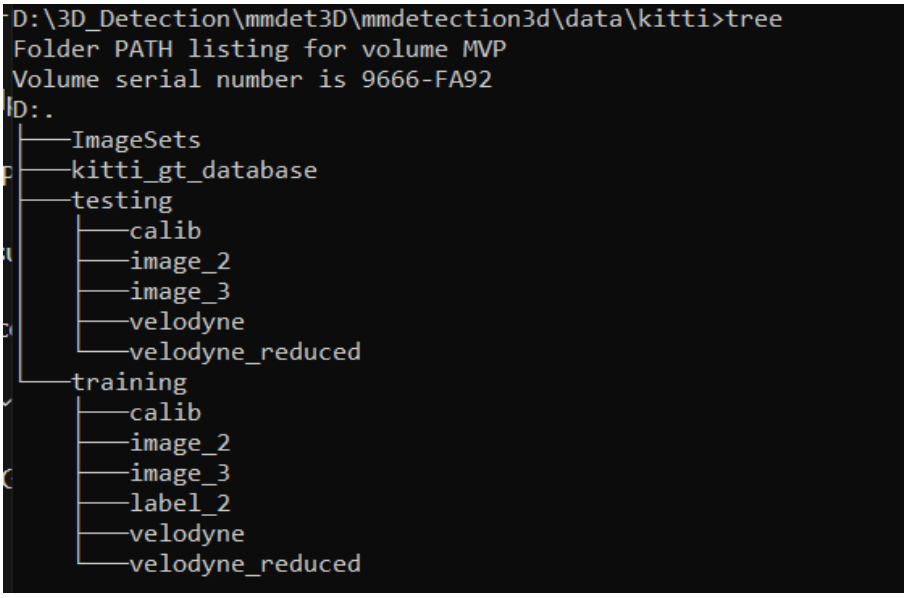
6.1.3 Priprava datasetu

Dataset bol stiahnutý z oficiálnej stránky. Na detekciu objektov bolo nutné stiahnuť nasledujúce súbory:

- Image_2: Ľavé farebné snímky z kamery
- Velodyne: 3D údaje LiDAR zachytené snímačom Velodyne.
- Labels: 3D ohraničujúce boxy a iné anotácie objektov na trénovanie.
- Calib: Kalibračné údaje snímača na transformáciu medzi rôznymi súradnicovými systémami.

Potom použitím nástroja od MMDetection si je možné tieto dáta a usporiadať ich do nasledujúcej štruktúry podľa Obrázku 13. Tieto dáta je nutné spracovať aby snimi prostredie vedelo pracovať pomocou príkazu:

```
python tools/create_data.py kitti --root-path ./data/kitti --out-dir ./data/kitti --extra-tag kitti
```



Obrázek 13 - Súborova štruktúra datasetu KITTI

6.2 Testovanie detekčných metód

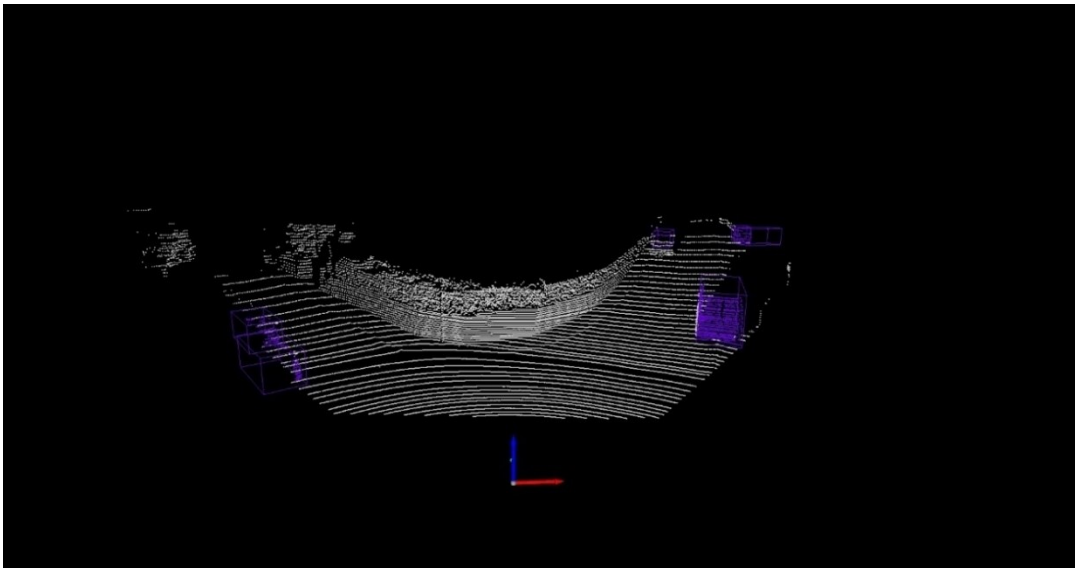
- **3DSSD (**3D Single Stage Object Detector) [42]

Základný princíp: Detektor je plne konvolučná sieť na detekciu objektov z bodových mračien. Metóda využíva inovatívny prístup k anchor-free detekcii, ktorý eliminuje potrebu preddefinovaných kotviacich boxov (anchors), čo je štandardná prax v mnohých detektoroch. Tento prístup znižuje počet hyperparametrov, ktoré je potrebné nastaviť, a zjednodušuje trénovací proces.

Mechanizmus fungovania: 3DSSD[42] implementuje novú metódu nazvanú Candidate Generation Module (CGM), ktorá efektívne vyberá kandidátne body z bodového mračna s vysokou presnosťou lokalizácie. Tento modul zvyšuje účinnosť detekcie tým, že priamo predikuje 3D bounding boxy bez potreby predbežného generovania veľkého počtu kandidátov, čo je často limitujúce a výpočtovo náročné.

Metóda 3DSSD[42] spracúva mračna bodov. Autori publikácie tejto metódy uvádzaju, že by mala byť rýchlejšia ako Pointpillars, no z testu vyplýva pravý opak. Model na viacerých snímkach při detekcií falošne detegoval objekty alebo naopak nededetegoval objekty, ktoré by mal. Pre kvantívne vyhodnotenie presnosti a chýb při detekcií bol vykonaný test, ktorého výsledky sú uvedené v Tabulke 1.

Použitým konfiguračným súborom je 3dssd_4xb4_kitti-3d-car.py a váhy budú dostupné v prílohách práce.Metóda bola predtrénovaná na detekciu aut jako vidieť na Obrázku 14.



Obrázek 14 - Detekcia objektov použitím 3DSSD

Obtiažnosť:	Ľahká	Stredná	Ťažka
Bbox	62.6442%	52.3111%	45.5579%
BEV	7.6546%	5.3482%	4.4792%
3D box	4.8523%	3.4972%	2.7970%
AOS	54.27%	45.45%	39.52%

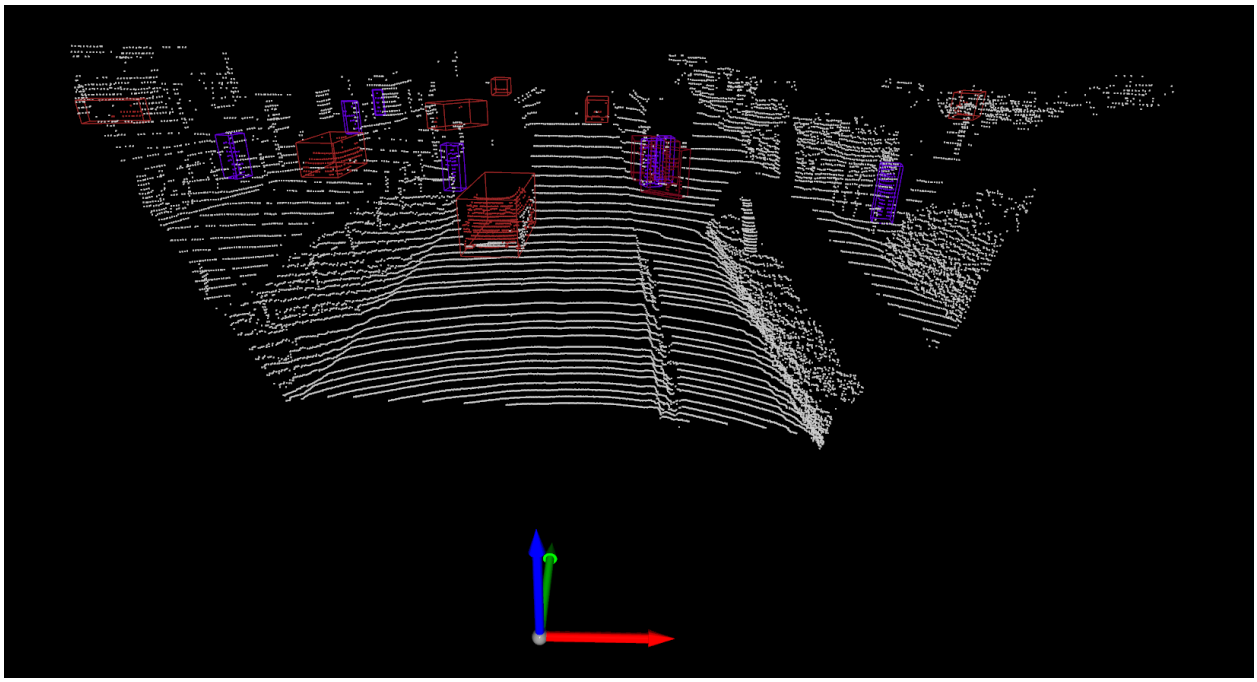
Tabulka 1 - Výsledky detekcie objektov podľa obtiažnosti - 3DSSD

- **Pointpillars**

Základný princíp: PointPillars [43] je metóda na rýchlu enkódovanie bodových mračien do pseudoobrazového formátu, ktorý môže byť spracovaný štandardnými 2D konvolučnými neurónovými sieťami. Tento prístup transformuje 3D data na 2D plochu, kde každý „stĺpec“ (pillar) predstavuje zoskupenie bodov v určitých oblastiach priestoru.

Mechanizmus fungovania: PointPillars [43] skladá bodové mračna do stĺpcov, kde sú jednotlivé body reprezentované ich atribútmi, ako sú poloha a odrazová intenzita. Každý stĺpec je potom enkódovaný do vlastností pomocou malých neurónových sietí, a tieto vlastnosti sú následne spracované pomocou 2D konvolučnej neurónovej siete na predikciu polohy a klasifikáciu objektov.

PointPillars [43], spracúva mračná bodov a preukázala významné výhody v presnosti a efektivite na tréningových datasetoch, ako je KITTI. Avšak, pri detailnejšom pohľade na jej aplikáciu sa ukazuje, že metóda má obmedzenia, ktoré môžu komplikovať jej širšie využitie. Najmä detekcia menších objektov, ako sú peší a cyklisti, je často nekonzistentná, čo môže byť dôsledkom nedostatočnej hustoty bodov, ktorá je kritická pre presné rozpoznávanie týchto objektov. Táto slabina závislá na spoľahlivej detekcii všetkých účastníkov premávky. Vizualizované na Obrázku 15.



Obrázek 15 - Vizualizácia detekcie použitím Pointpillars

Testovaním siete sme dostali nasledovné hodnoty:

Obtiažnosť:	Ľahká	Stredná	Ťažka
Bbox	84.0285	75.4338	72.4253
BEV	80.5906	70.2399	66.5576
3D box	76.5654	64.3254	60.7387
AOS	76.17	67.77	64.94

Tabulka 2 - Výsledky detekcie objektov podľa obtiažnosti – Pointpillars

- **ImVoxelNet** [44]

Základný princíp: ImVoxelNet [44] je metóda zameraná na prekonanie výziev spojených s použitím jednoduchých monokulárnych kamerových systémov pre 3D detekciu. Táto metóda kombinuje hĺbkové odhady získané z monokulárnych obrazov s technikami voxelizácie pre rekonštrukciu 3D scény.

Mechanizmus fungovania: ImVoxelNet [44] využíva neurónové siete na predikciu hĺbkových máp z jednotlivých 2D obrazov, ktoré sú následne transformované do 3D voxelovej mriežky. Každý voxel v tejto mriežke je klasifikovaný ako obsadený alebo voľný, čo umožňuje rekonštrukciu a detekciu objektov v 3D priestore.

ImVoxelNet [44] je monokulárny detektor 3D objektov. V dostupnej konfigurácií bol tento model predtrénovaný len na detekciu osobných automobilov. Na vykresľovanie boxov sa využíva vizualizér zakomponovaný v MMDet3D. Detekcie objektov sú na Obrázku 12.

Obrázek 16 - Vizualizácia detekcie ImVoxelNet

Obtiažnosť:	Ľahká	Stredná	Ťažka
Bbox	92.1176	80.9698	69.0902
BEV	62.2900	44.5477	37.7233
3D box	55.9647	39.2646	32.8319
AOS	89.77	77.45	65.59

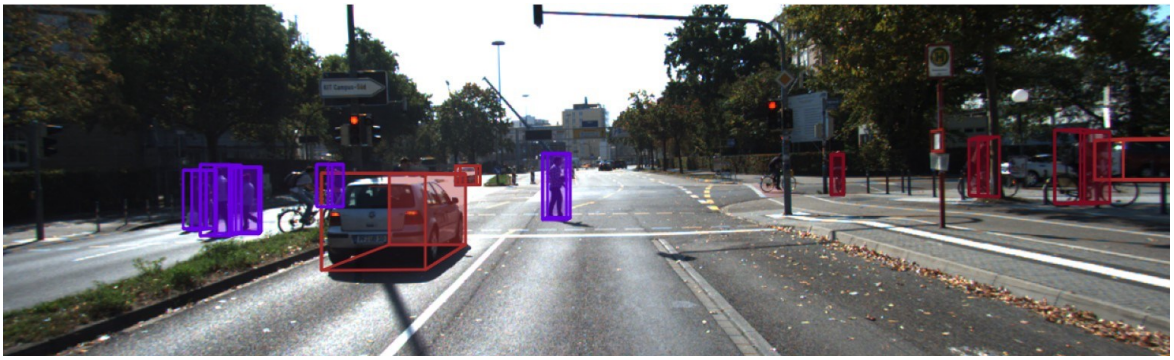
Tabulka 3 - Výsledky detekcie objektov podľa obtiažnosti - ImVoxelNet

- **Smoke** [45]

Základný princíp: Smoke [45] je single-stage detektor pre monokulárne 3D detekcie, ktorý sa sústreďuje na odhad kľúčových bodov (keypoints) z 2D obrazov na priamu predikciu 3D bounding boxov.

Mechanizmus fungovania: Smoke [45] predikuje centrálny kľúčový bod a rozmerové atribúty každého objektu na 2D obrázku. Na základe týchto informácií je možné rekonštruovať polohu a orientáciu 3D bounding boxu. Tento prístup minimalizuje potrebu zložitých a výpočtovo náročných postupov ako sú multi-stage detekčné pipeline.

Smoke [45] je taktiež monokulárny detektor 3D objektov, ktorý v kontraste s metodou ImVoxelNet metoda Smoke zefektívňuje proces detekcie objektov tým, že využíva metódu založenú na kľúčových bodoch na priamy odhad 3D ohraničujúcich boxov z jednotlivých obrázkov. Tento prístup eliminuje potrebu generovania 2D návrhov oblastí, pričom cieľom je znížiť výpočtovú zložitosť a zvýšiť rýchlosť spracovania.



Obrázek 17 - Vizualizacia detekcie SMOKE

Obtiažnosť:	Ľahká	Stredná	Ťažka
Bbox	62.6442	52.3111	45.5579
BEV	7.6546	5.3482	4.4792
3D box	4.8523	3.4972	2.7970
AOS	54.27	45.45	39.52

Tabulka 4 - Výsledky detekcie objektov podľa obtiažnosti - SMOKE

6.3 Testovanie prístupu k segmentácií

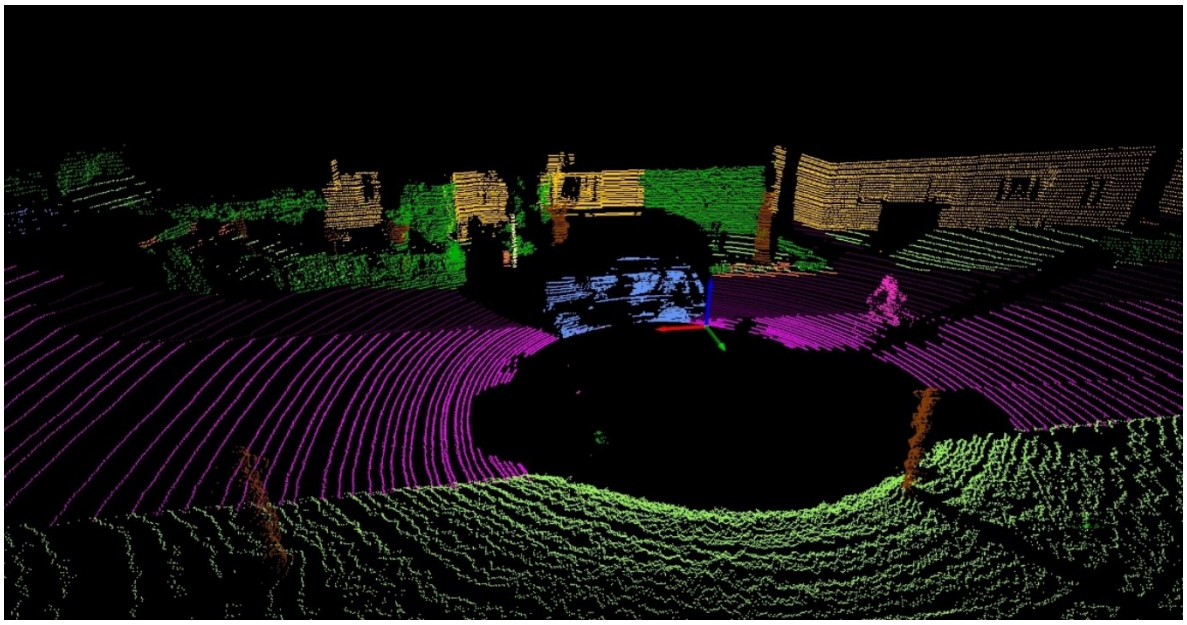
V tejto podkapitole sa budeme zaoberať praktickým skúškou semantickej segmentácie na datasete Semantic KITTI, ktorý je široko používaný v oblasti autonómneho riadenia a robotiky. Segmentácia je kľúčovým prvkom v pochopení vizuálneho vnímania a predstavuje základ pre rozhodovanie a navigáciu v dynamických prostrediach. Presná segmentácia scény umožňuje vozidlu rozlišovať medzi rôznymi typmi objektov ako sú vozidlá, chodci a iné prekážky. Cieľom praktických testov bude overiť schopnosti vybraných segmentačných modelov v reálnych podmienkach a posúdiť ich efektívnosť a presnosť.

- **Cylinder3D**

Testovanie metódy Cylinder3D [46] na datasete Semantic KITTI [38] je dôležité, aby sme overili, ako dobre dokáže táto metóda segmentovať objekty v reálnych mestských prostrediach. Cylinder3D rozdeľuje priestor do valcových segmentov, čo mu pomáha lepšie pracovať s mačnom bodov. Okrem toho používa špeciálne výpočtové metódy, aby sa zameral len na dôležité časti dát, čo je užitočné pre rýchle spracovanie v reálnom čase. Testovanie nám pomôžu lepšie pochopiť, ako môže Cylinder3D prispieť k zlepšeniu technológií na spracovanie 3D dát.

Výsledok sémantickej segmentácie na Obrázku 10 ukazuje, že dobre vycvičený model dokáže presne rozlíšiť veľké objekty, ako sú autá, budovy a cesty v 3D mračne bodov. Segmentácia je čistá s ostrými prechodmi, najmä v prípade veľkých objektov. Určité zlepšenie však môže byť potrebné v prípade menších objektov alebo menej zreteľných prvkov. Celkovo sa zdá, že je vhodný na aplikácie, ako je autonómne riadenie, ale môže byť potrebné ďalšie testovanie a ladenie pre okrajové prípady a detekciu menších objektov. výsledok sémantickej segmentácie ukazuje dobre natrénovaný model schopný presne rozlišovať veľké objekty, ako sú autá, budovy a cesty v 3D mračne bodov. Segmentácia je čistá s ostrými prechodmi, najmä v prípade veľkých objektov. Určité zlepšenie však môže byť potrebné v prípade menších objektov alebo menej zreteľných prvkov. Celkovo sa zdá

byť vhodný na aplikácie, ako je napríklad autonómne riadenie, ale môže byť potrebné ďalšie testovanie a ladenie pre okrajové prípady a detekciu menších objektov.



Obrázek 18 - Semanticka segmentacia – Cylinder3D

ZÁVER

Táto práca sa venovala problematike detekcie a segmentácie 3D objektov v obraze, pričom sa zamerala na využitie moderných prístupov a nástrojov dostupných v oblasti počítačového videnia a strojového učenia. Na základe literárnej rešerše boli predstavené viaceré metódy, ktoré sú v súčasnosti používané pre riešenie úloh spojených s 3D objektmi, pričom osobitná pozornosť bola venovaná frameworku MMDetection3D, ktorý poskytuje podporu pre prácu s predtrénovanými modelmi a rôznymi architektúrami pre detekciu a segmentáciu.

Výber MMDetection3D bol motivovaný jeho rozsiahlou funkcionalitou a dostupnosťou predtrénovaných modelov, ktoré umožnili testovanie bez nutnosti rozsiahleho trénovania nových modelov, čo by bolo vzhľadom na dostupné hardvérové zdroje značne obmedzujúce. Počas implementácie a testovania však vznikli problémy súvisiace s konfiguráciou prostredia, ktoré ukázali na limitácie práce v prostredí operačného systému Windows. MMDetection3D vyžaduje špecifické knižnice a nastavenia, ktoré sú pre Windows značne náročné na správne nasadenie, čo viedlo k obmedzeniam pri testovaní niektorých modelov.

V rámci praktickej časti práce boli otestované dostupné modely na vybranom datasete KITTI. Testovanie prebiehalo úspešne pri modeloch ako PointPillars, 3DSSD a ImVoxelNet, kde sa podarilo dosiahnuť konkrétne výsledky na zvolenom datasete. Výsledky testovania ukázali na silné aj slabé stránky jednotlivých modelov, pričom napríklad model PointPillars preukázal dobré výsledky pri detekcii väčších objektov, ale mal problémy s detekciou menších objektov. Naopak, 3DSSD sa stretol s nižšou presnosťou pri detekcii objektov, ako uvádzali teoretické predpoklady.

Pri semantickej segmentácii bol testovaný model Cylinder3D, ktorý bol vybraný z dôvodu svojej schopnosti pracovať s mračnami bodov pomocou valcovej reprezentácie dát. Segmentácia preukázala svoje schopnosti najmä pri väčších objektoch, ako sú vozidlá, budovy a cesty, no v prípade menších a zložitejších objektov sa vyskytli určité nedostatky. Vzhľadom na systémové obmedzenia nebolo možné testovať rôzne iné modely, pretože trénovanie a prispôsobenie nových modelov bolo nad rámec dostupných zdrojov.

Počas práce sa stretlo s viacerými výzvami, najmä s ohľadom na konfiguráciu prostredia a obmedzenia spojené s hardvérovými požiadavkami. MMDetection3D síce ponúka výkonné nástroje pre detekciu a segmentáciu, avšak jeho plné využitie si vyžaduje výkonnejšie hardvérové riešenia, prípadne presun do prostredí s lepšou podporou pre GPU a výkonnejšie výpočty, ako sú cloudové platformy.

V závere je potrebné zdôrazniť, že výsledky tejto práce ponúkajú pohľad na praktické využitie predtrénovaných modelov v kontexte detekcie a segmentácie 3D objektov, pričom ukazujú, že napriek dostupnosti moderných nástrojov a frameworkov, práca v tejto oblasti vyžaduje nielen teoretické znalosti, ale aj technickú pripravenosť na zvládnutie výpočtových nárokov týchto metód. V prílohách je priložená kópia prostredia bez datasetov, ktoré nie sú súčasťou pre ich veľkosť, čo reflektuje praktické obmedzenia pri práci s veľkými 3D dátami.

Do budúcna by mohlo byť prínosné uvažovať o širšom nasadení cloudových výpočtových prostriedkov alebo o použití alternatívnych systémov s lepšou podporou pre náročné výpočtové úlohy v oblasti 3D počítačového videnia. Získané skúsenosti ukazujú, že je potrebné optimalizovať nielen samotné modely, ale aj výpočtové prostredie, v ktorom sa tieto metódy realizujú.

ZOZNAM POUŽITEJ LITERATÚRY

[1] MICROSOFT CORPORATION, 2024. *Windows 10*. Online. Dostupné z: https://www.microsoft.com/cs-cz/download/windows. [cit. 2024-08-19].

[2] THE LINUX FOUNDATION, 2024. PyTorch. Online. Dostupné z: https://pytorch.org/get-started/previous-versions/. [cit. 2024-08-19].

[3] *MMDetection3D*, 2024. Online. MMDetection3D. Dostupné z: https://github.com/open-mmlab/mmdetection3d. [cit. 2024-08-15].

[4] OPEN-MMLAB, 2024. MMDetection. Online. Dostupné z: https://github.com/open-mmlab/mmdetection. [cit. 2024-08-19].

[5] PointPillars: Fast Encoders for Object Detection From Point Clouds, 2019. Online. Dostupné z: https://ieeexplore.ieee.org/document/8954311. [cit. 2024-08-15].

[6] SECOND: Sparsely Embedded Convolutional Detection, 2018. Online. Dostupné z: https://doi.org/10.3390/s18103337. [cit. 2024-08-15].

[7] Open3D, 2024. Online. Dostupné z: https://www.open3d.org/. [cit. 2024-08-15].

[8] Point Cloud Library, 2024. Online. Dostupné z: https://pointclouds.org/. [cit. 2024-08-15].

[9] META PLATFORMS, INC, 2024. PyTorch3D. Online. Dostupné z: https://pytorch3d.org/. [cit. 2024-08-15].

[10] Detectron2, 2019. Online. Dostupné z: https://github.com/facebookresearch/detectron2. [cit. 2024-08-14].

[11] Mesh-RCNN, 2019. Online. Dostupné z: https://github.com/facebookresearch/meshrcnn. [cit. 2024-08-14].

[12] Kaolin, 2024. Online. Dostupné z: https://developer.nvidia.com/kaolin. [cit. 2024-08-19].

[13] *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. Online. Dostupné z: https://www.cvlibs.net/datasets/kitti/. [cit. 2024-08-15].

[14] GEIGER, A.; LENZ, P. a URTASUN, R., 2012. *Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite*. Online. SemanticKITTI A Dataset for Semantic Scene Understanding using LiDAR Sequences. Dostupné z: Proc.~of the IEEE Conf.~on Computer Vision and Pattern Recognition (CVPR). [cit. 2024-08-15].

[15] KUMAR VINODKUMAR, Prasoon; KARABULUT, Dogus; AVOTS, Egils; OZCINAR, Cagri a ANBARJAFARI, Gholamreza, 2023. *A Survey on Deep Learning Based Segmentation, Detection and Classification for 3D Point Clouds*. Online. Dostupné z: https://doi.org/10.3390/e25040635. [cit. 2024-08-08].

[16] *LU-Net: An Efficient Network for 3D LiDAR Point Cloud Semantic Segmentation Based on End-to-End-Learned 3D Features and U-Net*, 2019. Online. Dostupné z: https://arxiv.org/pdf/1908.11656. [cit. 2024-07-05].

[17] R. QI, Charles; SU, Hao; MO, Kaichun; J. GUIBAS, Leonidas a , 2017. *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*. Online. Stanford University. Dostupné z: https://arxiv.org/pdf/1612.00593. [cit. 2024-08-08].

[18] MATURANA, Daniel a SCHERER, Sebastian, 2015. *VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition*. Online. Dostupné z: https://www.ri.cmu.edu/pub_files/2015/9/voxnet_maturana_scherer_iros15.pdf. [cit. 2024-08-08].

[19] CAMUFFO, Elena; MARI, Daniele a MILANI, Simone, 2022. *Recent Advancements in Learning Algorithms for Point Clouds: An Updated Overview*. Online. Dostupné z: https://www.mdpi.com/1424-8220/22/4/1357.[cit. 2024-08-08].

[20] Survey and Systematization of 3D Object Detection Models and Methods, 2023. Online. Dostupné z: https://arxiv.org/pdf/2201.09354. [cit. 2024-07-22].

[21] *Understanding Object Detection: A Comprehensive Guide*, 2024. Online. In: Pareto. Dostupné z: https://i.imgur.com/wnnGohh.png. [cit. 2024-07-28].

[22] *Semantic Segmentation vs. Instance Segmentation: Explained*, 2022. Online. In: Roboflow. Dostupné z:https://datascience.stackexchange.com/questions/52015/what-is-the-difference-between-semantic-segmentation-object-detection-and-insta?ref=blog.roboflow.com. [cit. 2024-07-28].

[23] *Umelainteligencia.sk*, 2019. Online. In: Umelainteligencia.sk. Dostupné z: https://umelainteligencia.sk/wp-content/uploads/2019/11/NN-uvod-siet.png. [cit. 2024-08-20].

[24] SARRAF, Arman; SARRAF, Saman a AZHDARI, Mohammad, 2021. *A Comprehensive Review of Deep Learning Architectures for Computer Vision*

Applications. Online. Dostupné z: https://www.researchgate.net/publication/349702934_A_Comprehensive_Review_of_Deep_Learning_Architectures_for_Computer_Vision_Applications. [cit. 2024-08-20].

[25] A comprehensive survey on deep-learning based gait recognition for humans in the COVID-19 pandemic - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/An-example-of-3X3-convolution_fig1_370430520 [cit. 2024-08-20]

[26] *Convolutional Networks*. Online. Dostupné z: https://www.deeplearningbook.org/contents/convnets.html. [cit. 2024-08-20].

[27] *Flattened Convolutional Neural Network*. Online. Dostupné z: https://iq.opengenus.org/flattened-cnn/. [cit. 2024-08-20].

[28] IHAB, S. Mohamed, 2017. *Detection and Tracking of Pallets using a Laser Rangefinder and Machine Learning Techniques*. Online. In: . Dostupné z: https://www.researchgate.net/profile/Ihab-S-Mohamed/publication/324165524/figure/fig4/AS:611103428063232@1522709819047/An-example-of-pooling-with-a-2-2-filter-and-a-stride-of-2.png. [cit. 2024-08-20].

[29] *Review: DeepMask (Instance Segmentation)*, 2018. Online. Dostupné z: https://towardsdatascience.com/review-deepmask-instance-segmentation-30327a072339. [cit. 2024-08-20].

[30] *Dynamic Graph CNN for Learning on Point Clouds*, 2019. Online. Dostupné z: https://arxiv.org/pdf/1801.07829. [cit. 2024-08-20].

[31] SU, Hang; MAJI, Subhransu; KALOGERAKIS, Evangelos a LEARNED-MILLER, Erik, 2019. *Multi-view Convolutional Neural Networks for 3D Shape Recognition*. Online. Dostupné z: https://arxiv.org/pdf/1801.07829. [cit. 2024-08-20].

[32] RIEGLER, Gernot; ULUSOY, Ali Osman a GEIGER, Andreas, 2017. *OctNet: Learning Deep 3D Representations at High Resolutions*. Online. Dostupné z: https://arxiv.org/pdf/1801.07829. [cit. 2024-08-20].

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

[34] ZHOU, Yin a TUZEL, Oncel, 2017. *VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection*. Online. Dostupné z: https://arxiv.org/abs/1711.06396. [cit. 2024-08-20].

[35] CAESAR, Holger; BANKITI, Varun; H. LANG, Alex; VORA, Sourabh; ERIN LIONG, Venice et al., 2019. *NuScenes: A multimodal dataset for autonomous driving*. Online. Dostupné z: https://www.nuscenes.org/. [cit. 2024-08-16].

[36] Waymo Open Dataset, 2019. Online. Dostupné z: https://waymo.com/open/. [cit. 2024-08-20].

[37] S3DIS (Stanford 3D Indoor Spaces Dataset), 2017. Online. Dostupné z: http://buildingparser.stanford.edu/dataset.html. [cit. 2024-08-16].

[38] SemanticKITTI, 2019. Online. Dostupné z: http://www.semantic-kitti.org/. [cit. 2024-08-16].

[39] ApolloScape, 2018. Online. Dostupné z: http://apolloscape.auto/. [cit. 2024-08-20].

[40] Cityscapes, 2016. Online. Dostupné z: https://www.cityscapes-dataset.com/. [cit. 2024-08-20].

[41] JIANG, Haiyang; LU, Yuanyao a CHEN, Shengnan, 2022. *Research on 3D Point Cloud Object Detection Algorithm for Autonomous Driving*. Online. Dostupné z: https://onlinelibrary.wiley.com/doi/10.1155/2022/8151805. [cit. 2024-08-20].

[42] 3DSSD: Point-based 3D Single Stage Object Detector, 2020. Online. Dostupné z: https://arxiv.org/pdf/2002.10187. [cit. 2024-07-18].

[43] LANG, Alex H.; VORA, Sourabh; CAESAR, Holger; ZHOU, Lubing; YANG, Jiong et al., 2019. *PointPillars: Fast Encoders for Object Detection from Point Clouds*. Online. Dostupné z: https://arxiv.org/pdf/1812.05784. [cit. 2024-07-18].

[44] RUKHOVICH, Danila; VORONTSOVA, Anna a KONUSHIN, Anton, 2021. *ImVoxelNet: Image to Voxels Projection for Monocular and Multi-View General-Purpose 3D Object Detection*. Online. Dostupné z: https://arxiv.org/pdf/2106.01178. [cit. 2024-07-19].

[45] LIU, Zechen; WU, Zizhang; TOTH, Roland a TECH, ZongMu, 2020. *SMOKE: Single-Stage Monocular 3D Object Detection via Keypoint Estimation*. Online. Dostupné z: https://arxiv.org/pdf/2002.10111. [cit. 2024-07-20].

[46] ZHU, Xinge; ZHOU, Hui; WANG, Tai; HONG, Fangzhou; MA, Yuexin et al., 2020. *Cylindrical and Asymmetrical 3D Convolution Networks for LiDAR*

Segmentation. Online. Dostupné z: https://arxiv.org/pdf/2011.10033. [cit. 2024-07-20].

Seznam použitých symbolů a zkratek

3D	Troj dimenzionálny
6D	Šesť dimenzionálny
AP	Average Precission
AOS	Average Orientation Similarity
AR	Average Recall
BEV	Bird's Eye View
CNN	Convolutional Neural Network
DGCNN	Dynamic Graph Convolutional Neural Network
CUDA	Compute Unified Device Architecture
FCN	Fully Convolutional Network
FN	False negative
FP	False positive
IoU	Intersection over Union
LiDAR	Light Detection and Ranging
MLP	Multilayer Perceptron
mAP	Mean Average Precision
NMS	Non-Maximum Suppression
P-R	Precision-Recall Curve
PCL	Point Cloud Library
R-CNN	Regions with Convolutional Neural Network
RGB-D	Red, Green, Blue - Depth
ROI	Region of Interest
RPN	Region Proposal Network
ZP	True Positive

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

SEZNAM PŘÍLOH

Príloha 1. CD so subormi a textom práce

PŘÍLOHA P I: CD SO SUBORMI A TEXTOM PRÁCE

Prílohy obsahujú screenshoty z priebehu práce, testovacie sekvencie a konfiguračný súbor pre virtuálne prostredie conda.