

Česká zemědělská univerzita v Praze

Technická fakulta

Katedra technologických zařízení staveb



Diplomová práce

Identifikace objektů pomocí lidarového systému

Bc. Tomáš Krlička

© 2023 ČZU v Praze

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Tomáš Krlička

Informační a řídicí technika v agropotravinářském komplexu

Název práce

Identifikace objektů pomocí lidarového systému

Název anglicky

Objects Identification using the lidar system

Cíle práce

Cílem práce je na základě dat z lidarů identifikovat typické objekty, optimálně použitím neuronové sítě zvoleného typu.

Metodika

1. Úvod
2. Cíl práce
3. Metodika
4. Perimetrická ochrana
5. Prostorová ochrana
6. Analýza technických prvků a jejich možností, datové výstupy
7. Neuronové sítě, jejich klasifikace a použití
8. Návrh modelu neuronové sítě pro konkrétní nasazení
9. Realizace modelu, učení, testování, vyhodnocení
10. Zpracování výsledků, analýza úspěšnosti identifikace
11. Možnost použití a nasazení, cenová kalkulace
12. Závěr a vyhodnocení

Doporučený rozsah práce

50 – 60 stránek včetně obrázků a grafů

Klíčová slova

lidar, neuronová síť, identifikace objektů

Doporučené zdroje informací

BISHOP C. M.: *Neural Networks for Pattern Recognition*. Oxford University Press, New York, 1995, 498 s., ISBN 0-38-731073-8

CANTY, M J. *Image analysis, classification and change detection in remote sensing : with algorithms for Python*. Boca Raton ; London ; New York: CRC Press, Taylor & Francis Group, 2019. ISBN 978-1-138-61322-5.

CONLAN, C. *Algorithmic trading with Python : quantitative methods and strategy development*. Bethesda: Chris Conlan, 2020. ISBN 979-8632784986.

DONG, J.: *LiDAR Remote Sensing and Applications*, Taylor & Francis Ltd, 2017, ISBN: 9781138747241

KERANEN, K. – KOLVOORD, R. *Making spatial decisions using GIS and lidar : a workbook*. Redlands: Esri Press, 2016. ISBN 978-1589484290.

MAŘÍK V., ŠTĚPÁNKOVÁ O., LAŽANSKÝ J.: *Umělá inteligence 4*, Academia, Praha, 2003, 476 s., ISBN 80-200-1044-0

NOVÁK, M., et al.: *Umělé neuronové sítě teorie a aplikace*, C.H.Beck, 1998, Praha, 382 s., ISBN 80-7179-132-6

PILGRIM, M. *Ponořme se do Python(u) 3 = Dive into Python 3*. Praha: CZ.NIC, 2010. ISBN 978-80-904248-2-1.

1906

Předběžný termín obhajoby

2022/2023 LS – TF

Vedoucí práce

Ing. Zdeněk Votruba, Ph.D.

Garantující pracoviště

Katedra technologických zařízení staveb

Elektronicky schváleno dne 1. 2. 2022

doc. Ing. Jan Malaták, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 23. 2. 2022

doc. Ing. Jiří Mašek, Ph.D.

Děkan

V Praze dne 26. 03. 2023

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Identifikace objektů pomocí lidarového systému" jsem vypracoval(a) samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor(ka) uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 31. března 2023

Poděkování

Rád bych touto cestou poděkoval mému vedoucímu práce, Ing. Zdeňkovi Votrubovi, PhD. za opětovné, a především skvělé vedení mé závěrečné práce. Dále bych rád poděkoval kolegům z práce za příležitost, kterou jsem dostal a bez kterých bych ani toto téma nemohl vypracovat, především bych chtěl poděkovat Honzovi Kadeřábkovi, bez kterého bych tak obsáhlé téma nezvládl. V neposlední řadě, bych chtěl poděkovat rodině za podporu při celém studiu a mým přátelům z IŘT, bez kterých bych se tak daleko nikdy nedostal.

Identifikace objektů pomocí lidarového systému

Abstrakt

Cílem práce je ověřit možnost využití neuronové sítě při vyhodnocování poplachů u bezpečnostního systému založeném na lidarové technologii. Na začátku práce je probrána technologie zabezpečovacích technologií a podrobně popsána lidarová technologie a její využití. Následně jsou popsány lidarové zabezpečovací systémy a neuronové sítě, ze kterých vychází praktická část. V praktické části, je popsán zabezpečovací systém, který poskytuje data, za pomoci vlastní anotační platformy pro neuronovou síť a následně je popsáno schéma neuronové sítě společně s trénováním na datech a dosažené výsledky.

Klíčová slova: lidar, neuronová síť, identifikace objektů

Object identification using lidar system

Abstract

The aim of the work is to verify the possibility of using neural network in the evaluation of alarms in a security system based on lidar technology. At the beginning of the thesis, the security technology is discussed and lidar technology and its applications are described in detail. Subsequently, lidar-based alarm systems and neural networks are described, from which the practical part is based. In the practical part, the security system that provides the data is described, using a custom annotation platform for the neural network and then the neural network scheme is described along with the training on the data and the results obtained.

Keywords: lidar, neural network, object identification

Obsah

1	Úvod	1
2	Cíl práce a metodika	2
2.1	Cíl práce	2
2.2	Metodika	2
3	Perimetrická ochrana	3
3.1	Pasivní	4
3.2	Aktivní.....	6
4	Prostorová ochrana	7
4.1	Další typy zařízení.....	7
4.2	LiDAR.....	7
4.2.1	Typy a využití lidarů	13
4.2.2	Výrobci.....	16
5	Hexagon 3D Surveillance	22
5.1	Leica BLK247	22
5.2	Accur8vision	23
5.2.1	MIND	25
5.3	Ostatní lidarové bezpečnostní systémy	26
6	Umělá inteligence	27
6.1	Neuronová síť	28
7	Praktická část	32
7.1	Příprava serveru	32
7.2	Příprava dat	35
7.3	Model Neuronové sítě.....	44
8	Závěr	54
9	Zdroje	55
	Příloha A: Schéma neuronové sítě	59

Seznam obrázků

Obrázek 1 PIR detektor (JABLOSHOP.cz, 2022) a jeho pokrytí (Detektory Pyronix, 2015).....	5
Obrázek 2 Schéma lidarů (Optical encoders and LiDAR scanning, 2022)	8
Obrázek 3 Zobrazení point cloudu s informací o reflexivitě [3].....	9
Obrázek 4 Multiple return lidar princip a výsledný point cloud (A. Wasser, 2022)	14
Obrázek 5 Distribuce paprsků (Accur8vision - Handbook, 2022).....	16
Obrázek 6 Lidary VLP-16 Puck a Ultra Puck (Velodyne lidar, 2022).....	17
Obrázek 7 Ouster lidary OS1 a OS2 (About Ouster, 2022).....	18
Obrázek 8 Leica BLK247 x5 a i5 (BLK247, 2022).....	19
Obrázek 9 Lidary Blickfeld Cube 1 a Cube Range 1 (Blickfeld, 2022)	20
Obrázek 10 Lidar Falcon (Innovusion, 2022).....	21
Obrázek 11 BLK247 systém (vlastní).....	23
Obrázek 12 Ukázka klientské části A8V při poplachu (Accur8vision, 2022)	24
Obrázek 13 MIND (vlastní)	25
Obrázek 14 Schéma neuronu (Durčák, 2017).....	29
Obrázek 15 Vizualizace vrstev neuronové sítě (Durčák, 2017).....	31
Obrázek 16 Schéma tabulky point (vlastní).....	34
Obrázek 17 Vizualizace nastavení systému (vlastní).....	35
Obrázek 18 Záznam člověka v A8V (vlastní).....	37
Obrázek 19 Class diagram (vlastní)	38
Obrázek 20 Vývojový diagram (vlastní).....	39

Obrázek 21 GUI anotační platformy (vlastní)	42
Obrázek 22 Převod model souboru do point cloudu (vlastní)	43
Obrázek 23 Struktura neuronové sítě PointNet (Qi et al., 2017)	44
Obrázek 24 Predikce neuronové sítě pro 150 bodů (vlastní)	50
Obrázek 25 Predikce neuronové sítě pro 100 bodů (vlastní)	51
Obrázek 26 Predikce neuronové sítě pro 50 bodů (vlastní)	52
Obrázek 27 Predikce neuronové sítě pro 25 bodů (vlastní)	52
Obrázek 28 Predikce neuronové sítě pro 10 bodů (vlastní)	53

Seznam tabulek

Tabulka 1 Porovnání dvou vlnových délek u lidarů (vlastní).....	12
Tabulka 2 Základní parametry lidarů VLP-16 Puck a Ultra Puck (Velodyne lidar, 2022)	17
Tabulka 3 Základní parametry lidarů OS1 a OS2 (About Ouster, 2022)	18
Tabulka 4 Základní parametry lidarů BLK247 x5 a i5 (BLK247, 2022)	19
Tabulka 5 Základní parametry lidarů Cube 1 a Cube Range 1 (Blickfeld, 2022) ...	20
Tabulka 6 Základní parametry lidarů Falcon (Innovusion, 2022)	21
Tabulka 7 Výsledky měření clusteringu (vlastní).....	33
Tabulka 8 Výsledky neuronové sítě při 150 bodech (vlastní)	49
Tabulka 9 Výsledky neuronové sítě při 100 bodech (vlastní)	50
Tabulka 10 Výsledky učení neuronové sítě pro 50 a 25 bodů (vlastní).....	51
Tabulka 11 Výsledky neuronové sítě při 10 bodech (vlastní)	53

1 Úvod

V současné době, se stále častěji využívá elektronických bezpečnostních systémů, jakou jsou kamery, různé typy senzorů a alarmy. Tyto systémy, poskytují ochranu před neoprávněným vstupem, a tím zajišťují bezpečí osob a majetku. Systémy rovněž mohou být propojeny s bezpečnostním centrem, díky kterému je možné reagovat okamžitě, na neoprávněné vniknutí a v případě vniknutí mohou incident zaznamenat, popřípadě identifikovat pachatele, což může být užitečné při vyšetřování zločinu.

Jeden z modernějších senzorů, které se začaly integrovat do bezpečnostních systémů, jsou lidary, které jsou schopny monitorovat velký objem prostoru s velkou přesností, rychlostí a současně s malou chybovostí. I díky těmto vlastnostem se lidary využívají v různých aplikacích u automobilů, robotů, k digitálnímu mapování a dalších.

V bezpečnostních systémech, které využívají právě lidary, se vyhodnocují point cloudy, jestli se nejedná o narušitele. Pro vyhodnocení narušitele systémy často využívají nějakou formu umělé inteligence, která je naučená na identifikaci objektů z point cloudu.

2 Cíl práce a metodika

2.1 Cíl práce

Hlavním cílem práce, je ověřit a analyzovat možnost nasazení neuronové sítě, v lidarovém zabezpečovacím systému, pro vyhodnocování poplachů a rovněž analyzovat, s jakou přesností, je neuronová síť schopna rozeznat narušitele od falešných poplachů.

2.2 Metodika

Teoretická část se zaměří na popis komerčních bezpečnostních technologií, které se využívají v bezpečnostních systémech. Dále bude detailně popsána technologie lidarů a jeho využití napříč obory, stejně tak i v zabezpečovacích systémech. Následně se popíší stávající bezpečnostní systémy, které využívají lidary, jako primární snímač.

V praktické části, se připraví a nastaví všechny klíčové prvky pro detekci v systému Accur8vision, který bude sloužit jako hlavní zdroj dat pro neuronovou síť. Následně se vytvoří anotační platforma, která bude data operovat ve dvou módech. V selekčním módu, bude platforma ze serveru shromažďovat data a v anotačním módu, bude data připravovat pro učení neuronové sítě. Rovněž se využijí i data z jiných zdrojů, pro získání co možná největšího počtu trénovacích dat. V poslední části se otestuje schopnost a přesnost vyhodnocování objektů za pomoci neuronové sítě, vycházející ze schématu PointNet, při učení na různém počtu vstupních bodů v point cloudu.

3 Perimetrická ochrana

Jedná se o druh bezpečnostního systému, který monitoruje a chrání perimetr objektu, jako jsou například okraje budov, přilehlé pozemky, nebo jeho hranice. Cílem je zajistit ochranu majetku nebo osob, před neoprávněným vstupem, krádeží nebo útokem. Současně s perimetrickou ochranou je nutná i nějaká forma mechanické zábrany, jako je plot, pro definování narušení. Perimetrická ochrana se skládá z více prvků, které jsou navzájem propojené a fungují jako celek pro zajištění bezpečnosti objektu (Křeček, 2006). Mezi tyto prvky patří:

Senzory – Slouží pro detekci narušitele a mohou být různého typu. Instalují se na místa, kde může dojít k neoprávněnému vniknutí, tak aby měli co možná největší pokrytí. Senzory budu detailněji popisovat v následujících podkapitolách.

Řídící jednotka – Jedná se o centrální prvek, který spojuje všechny zařízení a je na ní provázena všechna logika vyhodnocování. Slouží k monitorování a řízení střeženého objektu a může být umístěna v místech střeženého objektu, nebo v bezpečnostním centru.

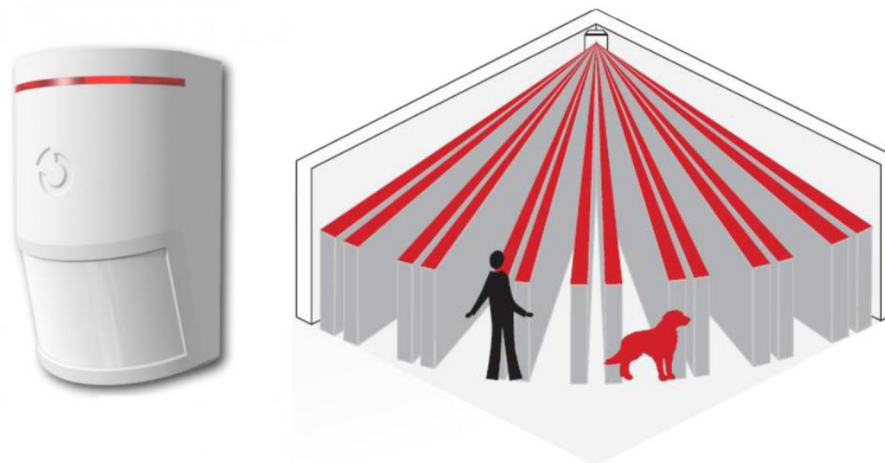
Signalizační zařízení – Jedná se o zařízení, které má v případě podezřelé aktivity spustit vizuální nebo akustický signál, který má za úkol upozornit na podezřelou aktivitu v okolí objektu nebo vystrašit případného narušitele, aby zanechal svého protiprávního jednání. Buzený může být v případě, že senzor detekuje podezřelou činnost, nebo pokud se senzor dostane do podezřelého stavu, jako je jeho sabotáž. Alarm může být vyslán na centrální řídicí jednotku, nebo přímo na složky bezpečnosti.

Bezpečnostní personál – Jedná se o vyškolené zaměstnance objektu, nebo externí bezpečnostní firmy, kteří reagují v případě alarmu a prověří danou skutečnost. (Perimetrická ochrana, 2020).

3.1 Pasivní

Senzory pasivní nevysílají do svého okolí žádný signál, pouze reagují na změny prostředí, který střeží. Jsou určeny pro detekci v menších prostorech, nebo detekovat změny přímo před ním. Jsou cenově a energeticky méně náročné než senzory aktivní.

Pasivní infračervené čidlo (PIR) je detektor pohybu, který snímá infračervené záření okolí, které emituje každé těleso s teplotou nad absolutní nulou a s teplotou pod 560 °C, kdy dochází k emitování na vlnové délce viditelného spektra, ve svém zorném poli. Pro vyhodnocování, využívá teplotní čidla, která jsou většinou zapojena do dvojic, pro zamezení vzniku falešných poplachů. Jako teplotní čidlo se typicky využívá pyroelektrický člen, který je schopný generovat elektrický náboj při změně teploty. Na sensor je promítnut teplotní obraz prostředí a je tedy schopen detekovat pouze změny v okolí. Zorné pole senzoru je rozdělené optikou na viditelné a zakryté části. Pokud narušitel projde mezi těmito částmi, na sensor dopadne jiné množství infračerveného záření, což má za následek změnu napětí na senzoru a vyhlásí se poplach. PIR detektory, se vyrábí i v kombinovaných formách, s ultrazvukovým nebo mikrovlnným detektorem, které popíšu v dalších kapitolách. Podle použité čočky, se určuje tvar velikost zorného pole, při kruhovitém uspořádání optik, může sensor detekovat i v zorném poli 360°. Podle citlivosti senzoru se odvíjí, na jakou vzdálenost je sensor schopen detekovat. (Křeček, 2006) (Intrepid, 2012)



Obrázek 1 PIR detektor (JABLOSHOP.cz, 2022) a jeho pokrytí (Detektory Pyronix, 2015)

Mikrofonické kabely slouží především pro zabezpečení obvodových bariér. Pokud se narušitel snaží jakýmkoliv způsobem dostat přes bariéru, vytváří při tom na bariéře mechanické vlnění, které se zachytává na těchto kabelech. Na kabelu dochází ke změně tohoto vlnění na elektrický signál, který putuje kabelem. Na jeho konci se nachází vyhodnocovací jednotka, která vyhodnotí, zda se nejedná o falešný poplach, popřípadě se vyvolá poplach. Výhodou tohoto kabelu je, že je možný akustický odposlech a je možné určit, podle zvuku, co se děje. Délka jednoho úseku, může činit až 300 m. (Křeček, 2006)

Magnetické spínače se skládají z jazýčkového kontaktu a permanentního magnetu. Jazýčkový kontakt je tvořen skleněnou trubičkou, která je naplněná ochrannou atmosférou, ve které jsou umístěny dva feromagnetické kontakty. Pokud je magnet v blízkosti jazýčkového kontaktu, jeho magnetické pole působí na feromagnetické kontakty a ty se navzájem přitahují. Pokud se magnet vzdálí a feromagnetické kontakty nebudou v působení jeho magnetického pole, kontakt se rozezne, a přeruší se tak obvod, který může ústředna zaznamenat a vyvolat poplach. (Intrepid, 2012)

Mechanické spínače jsou malé spínače, konstrukčně přizpůsobené tak, aby se daly vložit do malých prostor, jako je západka v rámu. Slouží k detekci pokusu o odstranění nebo poškození předmětu. Při takovém incidentu se podobně jako u magnetického spínače rozezne kontakt a přeruší se tak obvod, který následně v ústředně vyvolá poplach.

Plotová tenzometrická čidla se skládají z kombinace mechanické a elektronické ochrany. Mechanická ochrana tvoří napnuté žiletkové, ostnaté nebo hladké dráty, s rozestupy menšími, než 10 cm. Elektronická vyhodnocovací část, zase snímá změnu jejich tahové difference, která je způsobena vnějším působením. Čidla je možné nastavit, aby spouštěla poplach při překonání určitého tahu, nebo naopak jeho poklesu. (Křeček, 2006)

3.2 Aktivní

Senzory aktivní vysílají aktivně signál do prostoru, který se odráží zpět, pokud narazí na překážku. Senzor následně signál detekuje a určí, zda se nejedná o narušitele. Tyto senzory jsou určené na střežení větších prostor ale jsou energeticky a cenově náročnější.

Infračervené závory a bariéry jsou jedny z nejrozšířenějších obvodových ochran. Skládá se vždy z emitoru a detektoru, emitor vysílá jeden nebo více infračervených paprsků, které dopadají na detektor. Pokud dojde k přerušení paprsků, detektor tuto skutečnost vyhodnotí a vyhlásí poplach. Dosah těchto senzorů je mezi 50–150 m, technicky je možné střežit i na větší vzdálenost, ale tím přicházíme o přibližnou informaci, v jakém místě byl vstup na střežený pozemek. Závory jsou vybaveny interním topením, aby nedocházelo k zarosení optiky nebo námraze na senzoru. Někteří výrobci využívají i optiku pro kontrolu propustnosti, pokud tedy dojde ke zhoršení počasí, závory se vyřadí z provozu a nebude vyvolávat falešný poplach. Při instalaci, je vždy důležité, aby se jednotlivé závory překrývaly, jinak by došlo k vytvoření slepého místa, díky kterému by se narušitel dostal na pozemek nepozorovaně. (Křeček, 2006)

Štěrbinové kabely jsou typicky dvojice koaxiálních kabelů, které jsou od sebe v odstupech uloženy v zemi. Kabely mají snížené stínění pomocí definovaných štěrbin. Jeden z páru vytváří elektromagnetické pole, které druhý kabel zachytává a vyhodnocuje jeho změny. Při průchodu narušitele je toto elektromagnetické pole narušeno a dojde ke změně výstupního signálu z druhého kabelu. (Intrepid, 2012)

4 Prostorová ochrana

Jedná se v podstatě o stejný bezpečnostní systém jako perimetrický, nicméně je rozšířený o senzory, které dokáží střežit větší plochu najednou.

4.1 Další typy zařízení

Ultrazvuková čidla využívají mechanické vlnění vysoké frekvence nad pásmem kmitočtu slyšitelného lidským uchem. Při použití je však nutné si uvědomit, že tyto frekvence jsou slyšitelné pro některá zvířata. Vysílač neustále vysílá do svého okolí mechanické vlnění, které se odráží v uzavřeném prostředí. Přijímač následně vyhodnotí odražený signál, pokud je ve stejném vztahu, jako vyslaný signál, zůstává v klidu, pokud je signál fázově změněný, vyhlásí poplach. Senzor využívá tzv. Dopplerova jevu, kdy dochází k fázovému posunu, pokud se signál odrazí od objektu pohybujícího se radiálně k senzoru. Dosah těchto senzorů je 10 metrů. Citlivost senzoru se odvíjí také od předmětů, které pohlcují ultrazvuk, jako jsou koberce nebo pěnové materiály.

Mikrovlnná čidla fungují na stejném principu jako ultrazvuková čidla. Využívají však namísto mechanického vlnění mikrovlny, většinou se jedná o pásma 2,4 GHz, 10 GHz nebo 24 GHz. (Křeček, 2006)

4.2 LiDAR

Lidar (Light Detection and Ranging), označovaný taktéž jako Ladar, je aktivní systém dálkového měření vzdálenosti, kterou měří pomocí laserového pulzu. Systém vzdálenost počítá následujícím vzorcem 1.

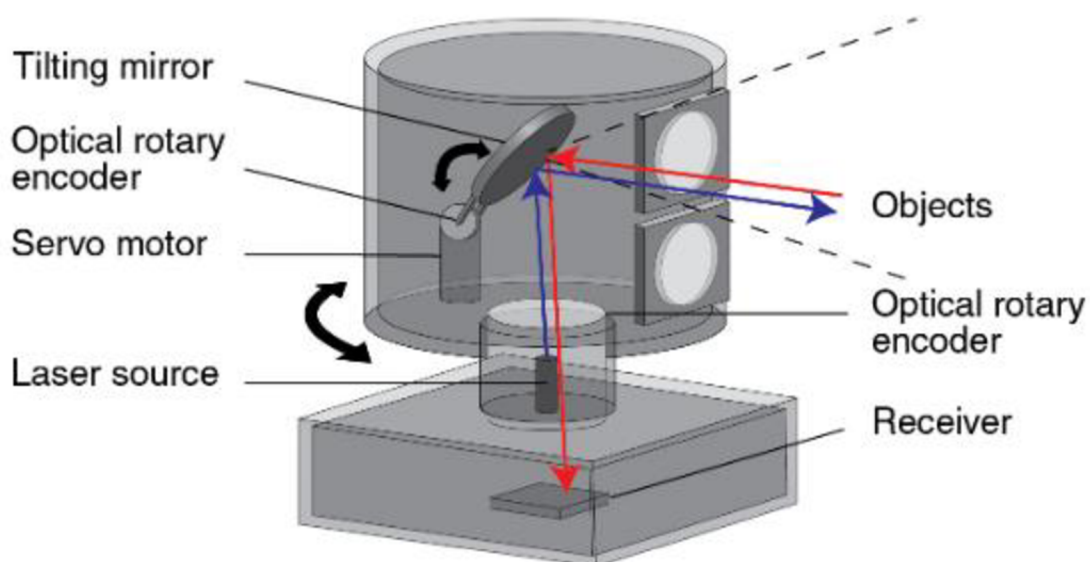
$$d = \frac{c \cdot t}{2}$$

(1)

Kde c je rychlost světla a t je časový úsek mezi vysláním laserového impulsu a detekcí odraženého laserového impulsu od detekovaného předmětu. Lidar se obvykle

skládá z párů diod, které emitují laserový pulz a fotodiod, které zachycují odražený laserový pulz. Na rozdíl od klasických laserových dálkoměrů, které měří vždy jen jeden bod, lidary používají několik zdrojů pulzů najednou. Výrobci neměnný počet paprsků označují jako kanály (channels) a často je vkládají do názvu daného lidaru (VLP-16, OS1-32). Lidary jsou tak schopné naskenovat během okamžiku i celé místnosti, a to díky jejich většímu skenovacímu zornému poli, u lidaru označovaného jako FOV (field of view).

Lidary využívají polohovací a orientační systém, známý jako POS (positioning and orientation systems), který se používá ke zvětšení horizontálního nebo vertikálního skenovacího zorného pole. Pro zvětšení zorného pole se využívá především polygonální zrcadlové systémy, čočky nebo kyvadlové mechanismy, které distribuují paprsek postupně pod jiným úhlem a zvyšují tak zorné pole. Pro panoramatické skenování, vertikální zorné pole 360°, se využívají elektromotory, které otáčejí pouzdem lidaru kolem základny rotační rychlostí až 20 Hz.

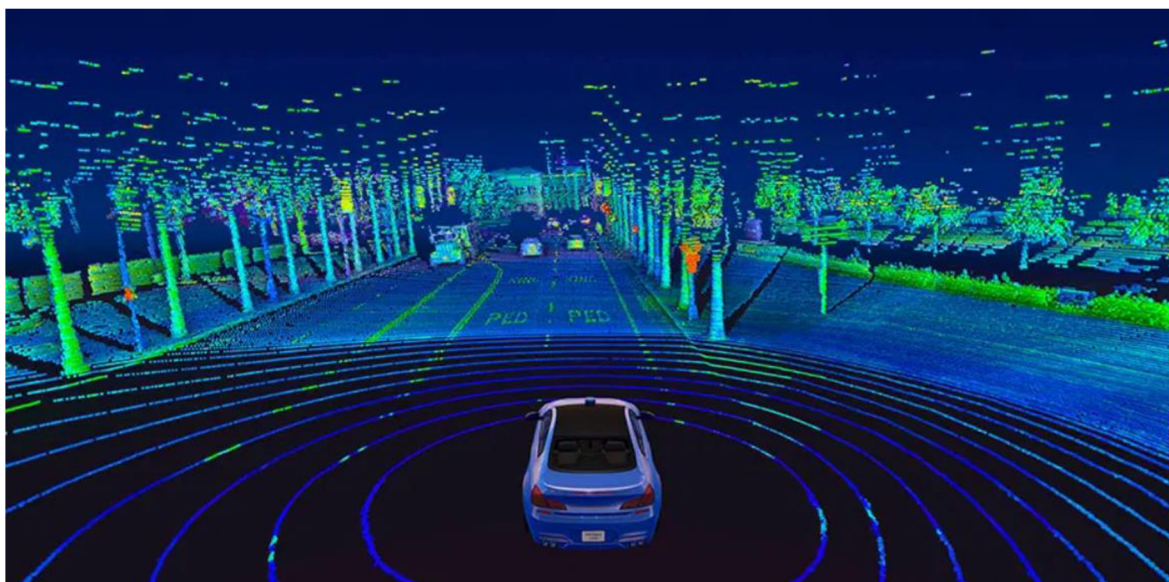


Obrázek 2 Schéma lidaru (Optical encoders and LiDAR scanning, 2022)

Výstupem z lidarů je mračno bodů, známé jako points cloud. Jedná se o matici dat, která označuje vzdálenost detekované překážky, od které se daný pulz odrazil. Lidary s ohledem na jejich rozlišení dokážou vyprodukovat obrovské množství bodů za sekundu, s rozlišením 512, které označují počet bodů ve vertikální ose a při otáčkách 10 Hz, je lidar schopen vyprodukovat přes 600 000 bodů za sekundu. Takové množství dat se rovná datovému toku 66 Mbps. Při rozlišení 2048 s rychlostí otáčení 10 Hz, lidar vyprodukuje přes 2,5 milionů bodů za sekundu, to se rovná 255 Mbps.

Přesnost měření lidarů, je u této technologie, velmi vysoká, s nepřesností menší než 2 cm při detekci na 100 m a úhlovým rozlišením $\pm 0,1$ stupně. Jednotlivé body jsou pak posílány typicky ve dvou formátech, jako X Y Z souřadnicích, nebo jako vzdálenost detekované překážky a úhel vyslaného pulzu. Výsledné body z lidarů, je pak možné zobrazovat v programech jako 3D objekty.

Některé lidary však nezaznamenávají pouze vzdálenost od detekovaného objektu, ale také jejich odraz, nebo i barvu povrchu, od kterého se pulz odrazil. S těmito někdy nadbytečnými daty, samozřejmě roste i datový tok, který lidar vyprodukuje.



Obrázek 3 Zobrazení point cloudu s informací o reflexivitě [3]

Pro lepší synchronizaci dat, a využití lidarů pro skenování velkých oblastí, například továren, celých měst nebo dálkového průzkumu země, lidar používá INS (inertial navigation system), která kombinuje GPS (global positioning system) a IMU (inertial measurement unit) neboli inerciální měřicí jednotku, díky kterým je možné zavést data do společného referenčního systému. (What is LiDAR and How Does it Work?, 2022)

Lidary používají především dvě rozhraní vlnových délek laserových impulzů, 905 nm a 1550 nm. Každá z těchto vlnových délek, představuje určitý kompromis pro skenování, které budou detailněji probrány níže. Rovněž každý komerčně prodávaný lidar, musí získat certifikaci bezpečnosti pro oči. Konkrétně se jedná o normu 60825-1 třídy 1, Mezinárodní elektrotechnické komise pro bezpečnost laserových zařízení, která vyznačuje záření v rozsahu vlnových délek od 180 nm do 1 mm a jsou zcela bezpečná pro člověka. (Optical encoders and LiDAR scanning, 2022)

Absorpce voda

Voda může silně ovlivnit integritu laserového pulzu vyslaným z lidarů. U pulzů dochází k velkému útlumu signálu při průchodem vodou, protože voda má silnou absorpci těchto pulzů. Při zhoršeném počasí, se síla pulzů rychle snižuje, což má za následek snížení síly pulzu a tím pádem i dosah lidarů. U pulzů s vlnovou délkou 1550 nm je absorpce signálu přibližně 145krát větší, než u pulzů s vlnovou délkou 905 nm.

Při zhoršeném počasí, dešti nebo mlze, dochází ke zhoršení návratnosti dat, používaných k vytváření mračen bodů, a tedy k celkovému zhoršení detekce. Při nepříznivém počasí, se detekce u pulzů s vlnovou délkou 1550 nm zhorší, přibližně 4 až 5násobně oproti pulzům s vlnovou délkou 905 nm.

U přenosu sněhem nebo ledem rovněž dochází k degradaci laserového pulzu. U signálu s vlnovou délkou 1550 nm je tato degradace až o 90 % horší než u pulzů s vlnovou délkou 905 nm. Toto zhoršení se však netýká pouze zhoršeného počasí, vztahuje se na jakýkoli povrch, který je mokrá nebo pokrytý ledem či sněhem. Laserové pulzy o vlnové délce 1550 nm může odrazivost na mokřím povrchu klesnout až o 60 % ve

srovnání s normálním stavem, u pulzů s 905 nm vlnové délky je pokles mnohem menší a to 15 %.

Z těchto informací je zřejmé, že využití lidarů, který využívá laserové pulzy o vlnové délce 905 nm, jsou více odolné vnějšímu vlivu počasí, zatímco laserové pulzy o vlnové délce 1550 nm, mohou být použity dle přesně stanovených podmínek, aby byl jejich potenciál plně využit. Z výše uvedených důvodů, zpráva Národního úřadu pro oceán a atmosféru (NOAA) o lidarech pro dálkový průzkum a mapování pobřeží doporučila nepoužívat lasery s vlnovou délkou 1550 nm s odkazem na jejich špatný výkon ve vlhkých podmínkách. (LiDAR Sensors, Simplified: Part 1, 2022)

Spotřeba energie

Aby se lidary, které využívají laserové impulzy o vlnové délce 1550 nm vyrovnaly v detekci laserovým impulzům o vlnové délce 905 nm za deště, mlhy nebo sněhu, musí zvýšit jejich výkon více než 10krát. To má za následek vznik většího tepla a nutnost chladících komponent, která opět zvyšují spotřebu elektrické energie, náklady na výrobu a složitost celého systému.

Dosah

Lidary jsou schopné při využití obou vlnových délek detekce na mnoha stovek metrů. Při zvýšení výkonu, jsou lidary využívající vlnovou délku 1550 nm, schopné detekce až 500 m, pulzy s vlnovou délkou 905 nm jsou schopné detekce do 200 m.

Detekce na větší vzdálenosti, bez ohledu na použitou vlnovou délku, je limitovaná, protože každý laserový puls je emitován ve tvaru kužele. Při divergenci laserového pulzu, dochází k postupnému rozprostření energie na větší plochu, což vede k nižšímu poměru signálu vůči šumu. Toto je jedna z oblastí, kde rastrové skenovací systémy mají výraznou výhodu. (Hadji, 2021)

Dostupnost

Lidary využívají pro generování laserových pulzů vlnové délky 905 nm, diody na bázi GaAs, které jsou v dnešní době velmi levné a výrobci jsou schopni tyto diody vyrábět na velkých wafrech. Detekce pulzu je u tohoto typu řešena přes klasické CMOS členy a běžných fotodiod. Pro vytvoření laserového pulzu o vlnové délky 1550 nm, se využívá dioda na bázi InP, jejich wafery nejsou tak velké s porovnáním waferů u GaAs diod. Pro detekci takového záření, je nutné použít fotodiodu na bázi InGaAs, které jsou dražší a jejich výroba není tak rozšířená.

Podle průzkumu provedený firmou IHS Markit, se cena za kanál, u lidarů využívající 905 nm vlnovou délku, pohybuje mezi 4-20 \$, zatímco za kanál s 1550 nm vlnovou délku se cena pohybuje okolo 275 \$. Průzkum dále odhaduje výrazný pokles cen do roku 2025, které se pohybují okolo 2-10 \$ za kanál o 905 nm a přibližně 155 \$ za kanál o 1550 nm. (Velodyne's Guide to Lidar Wavelengths, 2021)

	905 nm	1550 nm	Shrnutí
Voda	+	-	Voda absorbuje záření 1550 nm přibližně 145krát více než záření 905 nm.
Vliv počasí	+	-	Detekce při zhoršeném počasí je u záření 1550 nm až 5krát horší než u záření 905 nm.
Sníh	+	-	Reflexe záření 1550 nm od sněhu je o 97% horší než u záření 905 nm.
Spotřeba energie	+	-	Energetická náročnost u lidarů, která využívají záření 1550 nm, je až 10krát větší.
Dosah	-	+	Při optimálních podmínkách, mají lidary využívající záření 1550 nm, dosah více než dvojnásobný.
Cena/Dostupnost	+	-	Lidary využívající záření 1550 nm, jsou až 10krát dražší, v porovnání s lidary operující se zářením 905 nm.

Tabulka 1 Porovnání dvou vlnových délek u lidarů (vlastní)

4.2.1 Typy a využití lidarů

Lidary našly uplatnění v mnoha oborech, od dálkového průzkumu země, autonomních aut nebo stavebnictví. Pro každý obor se využívá jiný druh lidarů, které níže popíšu.

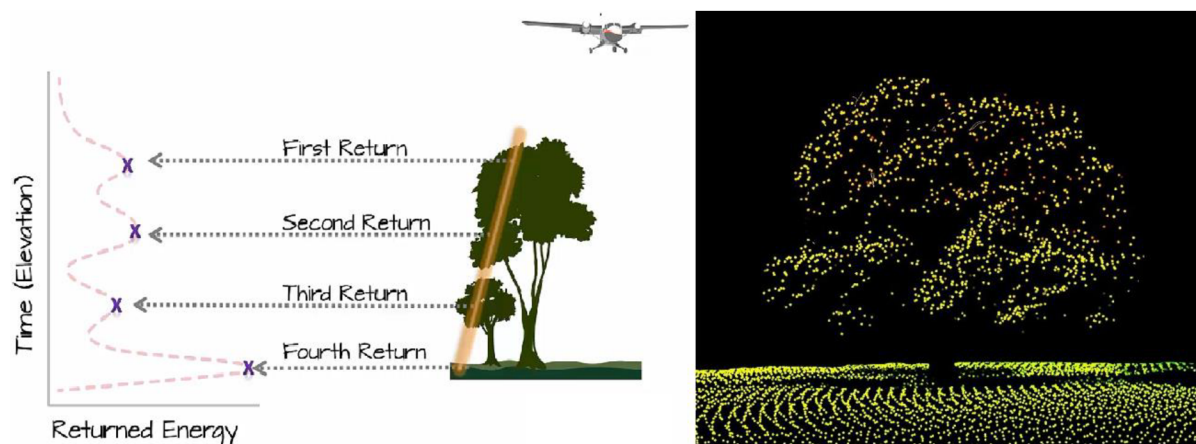
Pozemní lidary, jak název napovídá, jsou určeny pro aplikaci, kde je vyžadován podrobný průzkum okolí nebo terénu. Operují na zemi a mohou být upevněny na statickém místě, nebo autě. Tento typ lidarů se používá ve stavebnictví, průzkumu země a vozovek, autonomních aut nebo v zabezpečovacích systémech. Pozemní lidary se dají dále rozdělit do dvou skupin.

Pozemní mobilní lidar – Využívá se při skenování za pohybu, pro synchronizaci dat využívá INS, GPS a IMU, některé modely využívají i kamery. Využívají se pro skenování velkých objektů pro analýzu nebo tvorbu 3D modelů. Některé systémy využívají lidary čistě pro zlepšení fotogrammetrie.

Pozemní statické lidary – Tento druh lidarů se pevně umístí v jednom bodě, jako je například stativ. Tyto lidary nejsou mobilní, ale jsou zcela přenosné. Využívají se jako senzory pro detekování narušitelů v bezpečnostních systémech, strojírenství nebo stavebnictví.

Letecké lidary jsou určeny pro dálkový průzkum země z ptáčích perspektiv. Bývají upevněny na letadlech nebo helikoptérách, které se vznášejí nad zemí a jsou určeny pro skenování rozsáhlých oblastí o rozloze i několika hektarech. Typicky využívají kyvadlový mechanický POS v jedné ose, protože pohyb v druhé ose, zprostředkovává pohybem stroje, na kterém je upevněn. Na rozdíl od pozemních lidarů, nedetekují pouze první odražený pulz, ale je schopen zpracovat vícenásobný odraz daného pulzu. Vícenásobný návrat je způsoben divergencí pulzu, kterou jsem více popisoval. Při skenování z výšky u leteckých lidarů má laserový pulz konečný průměr i 10 cm. Kvůli této skutečnosti je možné, že na objekt dopadne pouze část průměru. Tato část pulzu se od něj odrazí, zatímco zbytek pulzu pokračuje v cestě, dokud nenarazí na další objekty, které způsobí odraz dalších částí

pulzu. To se využívá například při skenování zalesněných oblastí, kde nám výsledné mračno bodů může poskytnout obraz o jeho struktuře.



Obrázek 4 Multiple return lidar princip a výsledný point cloud (A. Wasser, 2022)

Letecké topografické lidary – Využívají se ke snímání jakéhokoliv typu pozemku a měření poklesu nebo vzestup povrchu. Jsou především určeny k vytváření topografických map. (Multiple return LiDAR, 2022)

Letecké batymetrické lidary – Na rozdíl od topografických lidarů, které nejsou velmi vhodné pro skenování vodní hladiny, jelikož operují především na vlnových délkách NIR a SWIR, batymetrické lidary využívají pulzy s vlnovou délkou okolo zelené barvy, konkrétně 532 nm. Pulzy o této vlnové délce, jsou schopny penetrovat vodu a poskytnout údaje o hloubce vodních ploch.

Satelitní lidary, se umísťují na vesmírné satelity, které obíhají kolem Země. Využívají se pro skenování nejen velkých částí Země, ale i atmosféry nad Zemí. NASA tento typ lidarů využívá pro skenování vegetace na Zemi, tání ledovců na obou pólech planety a rozmístění mraků v atmosféře. (Wang et al., 2014)

Diferenciální absorpční lidary využívají laditelný laser, který produkuje laserové pulzy o dvou vlnových délkách najednou. Lidar pak zaznamenává úroveň jejich absorpce a díky tomu je schopen detekovat složení plynu, kterým se pulz šířil. Tento typ se využívá

pro sledování úrovně ozónu, koncentrace plynů v atmosféře nebo měření znečištění ovzduší. (A. Wasser, 2022)

Větrné lidary, jsou pozemní měřicí zařízení, určené pro měření rychlosti a směru větru. Výška větru se vyhodnocuje na základu odraženého pulzu od částic a aerosolů ve vzduchu, směr se vyhodnocuje na principu Dopplerova posunu. Tyto lidary jsou určeny pro projektanty větrných elektráren. Na rozdíl od klasických měřících stanic, jsou lidarové systémy schopné určit rychlost a směr větrů do větší výšky, a na rozdíl od klasických anemometrů i s větší přesností. (The Different Types Of LiDAR Systems, 2022)

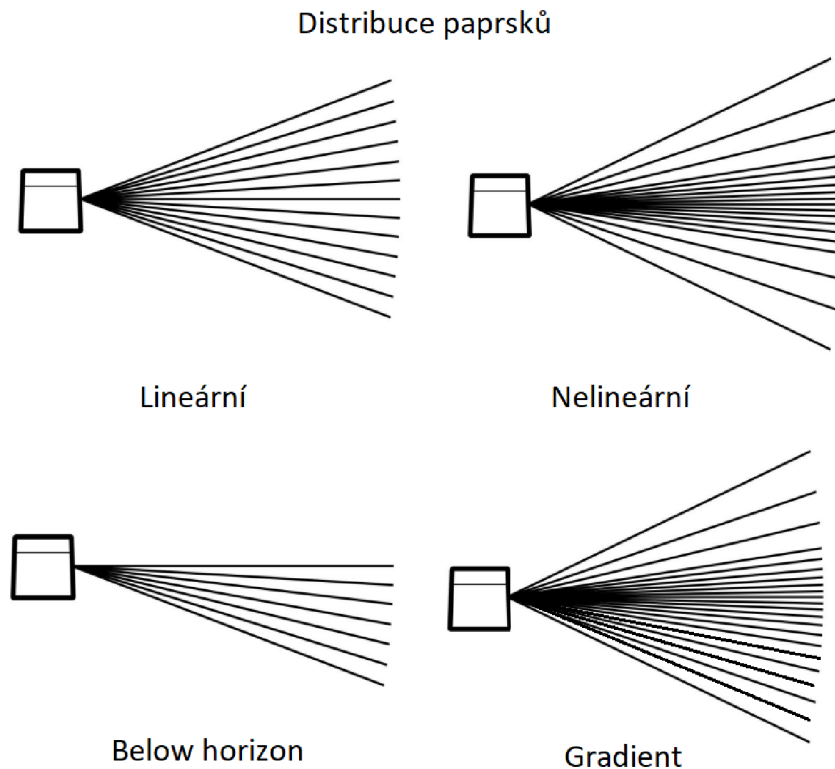
Dále můžeme lidary dělit podle jejich distribuce laserových pulzů. Jednotlivé skupiny popíšu níže.

Lineární lidary mají stejné rozestupy mezi jednotlivými paprsky. Díky tomu mají ideální poměr prostorové a dálkové detekce.

Nelineární lidary mají více nahuštěné paprsky ve středu. Ideální pro detekci na dlouhé vzdálenosti v rovině.

Gradientní lidary mají rovněž nahuštěné paprsky ve středu, jako nelineární lidary, současně však mají spodní část paprsků více zhuštěnou než tu horní. Tento druh lidaru je určen pro detekci, především pokud je upevněn ve vyšších výškách.

Below horizon lidary mají distribuci paprsků od středu dolů pod sebe, se stejnými rozestupy mezi nimi. Využívají se především pro vnitřní detekci, protože jejich paprsky dokáží perfektně pokrýt celé místnosti. (Accur8vision - Handbook, 2022)



Obrázek 5 Distribuce paprsků (Accur8vision - Handbook, 2022)

4.2.2 Výrobci

S rozmachem lidarové technologie, se na trhu objevilo velké množství výrobců. Níže popíšu některé z nich, především se budu věnovat lidarům, se kterými mám vlastní zkušenost.

Velodyne je americký výrobce lidarů. Především je tento výrobce znám svými prostorovými lidary VLP-16 Puck a Ultra Puck. Tyto modely lidarů jsou jedny z nejlevnějších na trhu, ale poslední řady trpěly konstrukčními chybami a obecně tyto lidary trpí velkou poruchovostí. Především trpí na poruchy ložisek, které otáčí celou základnou lidarů. Současně i fakt, že jejich poslední firmware, byl vydán v polovině roku 2019 naznačuje, aktuální nezájem ze strany výrobce. (Velodyne lidar, 2022)



Puck



Ultra Puck

Obrázek 6 Lidary VLP-16 Puck a Ultra Puck (Velodyne lidar, 2022)

	VLP-16 Puck	Ultra Puck
Počet kanálů	16	32
Zorné pole (Vertikální)	30° (-15.0° do +15.0°)	40° (-25° do +15°)
Zorné pole (Horizontální)	360°	360°
Úhlové rozlišení	2°	Minimální 0,33° (nelineární distribuce)
Dosah	100 m	200 m
Vlnová délka	903 nm	903 nm
Rychlost otáčení	5 Hz - 20 Hz	6 Hz - 20 Hz

Tabulka 2 Základní parametry lidarů VLP-16 Puck a Ultra Puck (Velodyne lidar, 2022)

Ouster je rovněž americká společnost, která vyrábí lidary řady OS. Jedná se o velice kvalitního výrobce prostorových lidarů. Na rozdíl od konkurence, Ouster neustále vydává aktualizace firmwaru pro své lidary, rovněž tak přidává i nové funkce, jako je změna struktury paketů, která má za následek zmenšení datového toku, což byl dlouhodobě jeden z jejich problémů. Při aplikaci lidarů od tohoto výrobce, se rovněž ukázal problém s konektorem. Při instalaci, kdy docházelo k ohybu kabelu u lidarů, nedoléhala izolace konektoru správně, což mělo za následek při venkovních instalacích jeho zoxidování. Výrobce tedy začal dodávat lidary nově s nevodivou pastou a bužírkou, která se navleče na konektor a celý ho zaizoluje. (About Ouster, 2022)



Obrázek 7 Ouster lidary OS1 a OS2 (About Ouster, 2022)

	OS1	OS2
Počet kanálů	32, 64 nebo 128	32, 64 nebo 128
Zorné pole (Vertikální)	45° (-22,5° do +22,5°)	22,5° (-11,25° do +11,25°)
Zorné pole (Horizontální)	360°	360°
Úhlové rozlišení	0.18°	0,09°
Dosah	120 m	240 m
Vlnová délka	865 nm	865 nm
Rychlost otáčení	10 Hz nebo 20 Hz	10 Hz nebo 20 Hz

Tabulka 3 Základní parametry lidarů OS1 a OS2 (About Ouster, 2022)

Leica je švýcarský výrobce lidarů, který exceluje na trhu s technikou určenou na detailní skenování budov, vytváření point cloudových map a tvorbou 3D modelů. Aktuálně disponují i lidarem, který je kompatibilní s roboty od Boston Dynamics pro automatizovaný průzkum a skenování oblastí. Jejich nejznámější řada lidarů je BLK247 i5, která je určena na vnitřní instalaci a BLK247 x5, která je určena i pro venkovní instalaci. Současně tyto lidary disponují i bezpečnostním systémem a umělou inteligencí na

vyhodnocování narušitelů, kamerami a v případě řady x5 i termokamerami. (BLK247, 2022)



Obrázek 8 Leica BLK247 x5 a i5 (BLK247, 2022)

	x5	i5
Počet kanálů	1	1
Zorné pole (Vertikální)	270°	270°
Zorné pole (Horizontální)	360°	360°
Úhlové rozlišení	nelineární distribuce	nelineární distribuce
Dosah	30 m	30 m
Vlnová délka	830 nm	830 nm
Rychlost otáčení	2 Hz	2 Hz

Tabulka 4 Základní parametry lidarů BLK247 x5 a i5 (BLK247, 2022)

Blickfeld je německý výrobce statických lidarů, který se zaměřuje na využití lidarů v IoT aplikaci a autonomní řízení. Lidary mají velmi uživatelsky přívětivé rozhraní, přes které je možné sledovat živě přenos dat. (Blickfeld, 2022)



Cube 1 Outdoor



Cube Range 1

Obrázek 9 Lidary Blickfeld Cube 1 a Cube Range 1 (Blickfeld, 2022)

	Cube 1	Cube Range 1
Počet kanálů	50	50
Zorné pole (Vertikální)	30°	12°
Zorné pole (Horizontální)	70°	18°
Úhlové rozlišení	0,4° - 1°	0.24° - 0.4°
Dosah	75 m	150 m
Vlnová délka	905 nm	905 nm
Obnovovací rychlost	1,5 Hz - 50 Hz	2,5 Hz - 50 Hz

Tabulka 5 Základní parametry lidarů Cube 1 a Cube Range 1 (Blickfeld, 2022)

Innovusion je americký výrobce statických lidarů, který vyrábí lidar s nejdelším dosahem a největším rozlišením na trhu v sériové výrobě. Jejich lidary se používají v automobilovém průmyslu. (Innovusion, 2022)

Falcon



Obrázek 10 Lidar Falcon (Innovusion, 2022)

	Falcon
Počet kanálů	neuveveno výrobcem
Zorné pole (Vertikální)	25°
Zorné pole (Horizontální)	120°
Úhlové rozlišení	0,05°
Dosah	250 m
Vlnová délka	1550 nm
Obnovovací rychlost	10 Hz

Tabulka 6 Základní parametry lidarů Falcon (Innovusion, 2022)

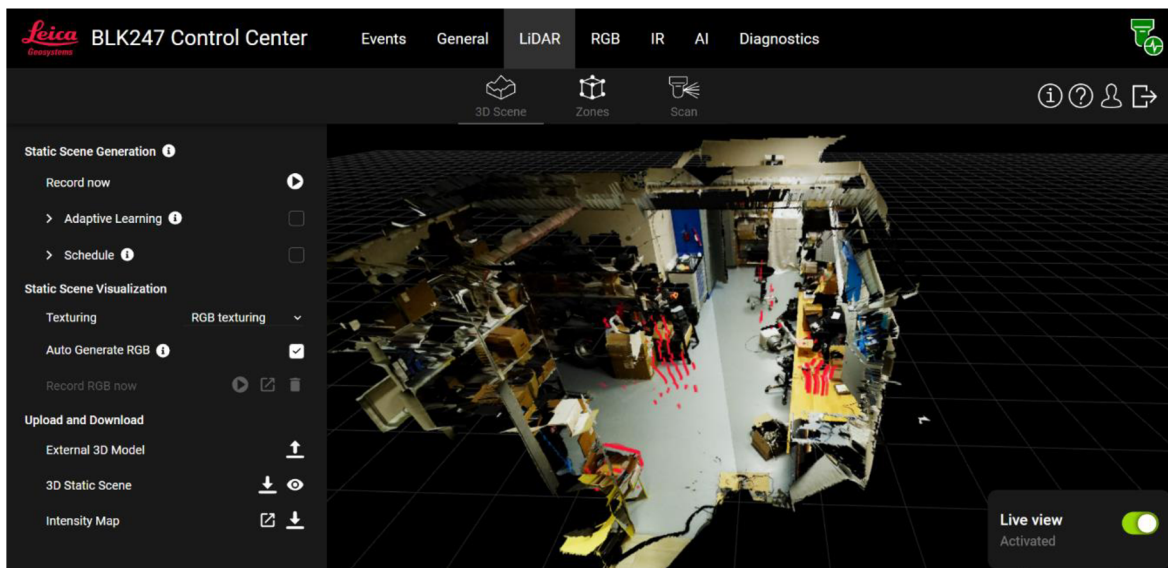
5 Hexagon 3D Surveillance

Hexagon je švédská firma, která se specializuje na automatizaci, softwarový vývoj a konstrukci senzorů. Angažuje se v celé řadě odvětví, od zemědělství, těžebního průmyslu, až po automatizaci ve strojnictví. Především však vyniká v geosystémech, disponuje jedním z nejdetailnějších průzkumů země s takřka plně naskenovaným územím USA a půlky Evropy. Jejich služby využívají především armády.

Aktuálně je také světovým lídrem ve volumetrických bezpečnostních systémech, které využívají pro detekci narušitele jako primární senzor právě lidar. Pod Hexagon aktuálně spadají dvě firmy, které tyto systémy vyvíjí. Konkrétně se jedná o Leicu, která je známá především výrobou optických zařízení a také vyrábí již zmíněné lidary řady BLK. Tou druhou je česká firma Tacticaware, která vyvíjí zabezpečovací systém Accur8vision. (Hexagon, 2023)

5.1 Leica BLK247

Jedná se o volumetrický bezpečnostní systém vyvinutý firmou Leica. Tímto bezpečnostním systémem disponují lidarové systémy řady BLKx5 a BLKi5. Systém spojuje lidarovou detekci, kamerový systém, kterým BLK247 disponuje a jejich vyhodnocování společně s umělou inteligencí, která pomáhá v klasifikaci point cloudových objektů. Bezpečnostní systém je přístupný přes webové rozhraní a jedná se o ideální *one device solution*, jelikož spojuje CCTV a EZS v jednom zařízení. Současně není nutná příprava speciální kabeláže, jelikož komunikace probíhá v rámci Ethernetové sítě a BLK disponují PoE ++ pro napájení i komunikaci. (BLK247, 2022)



Obrázek 11 BLK247 systém (vlastní)

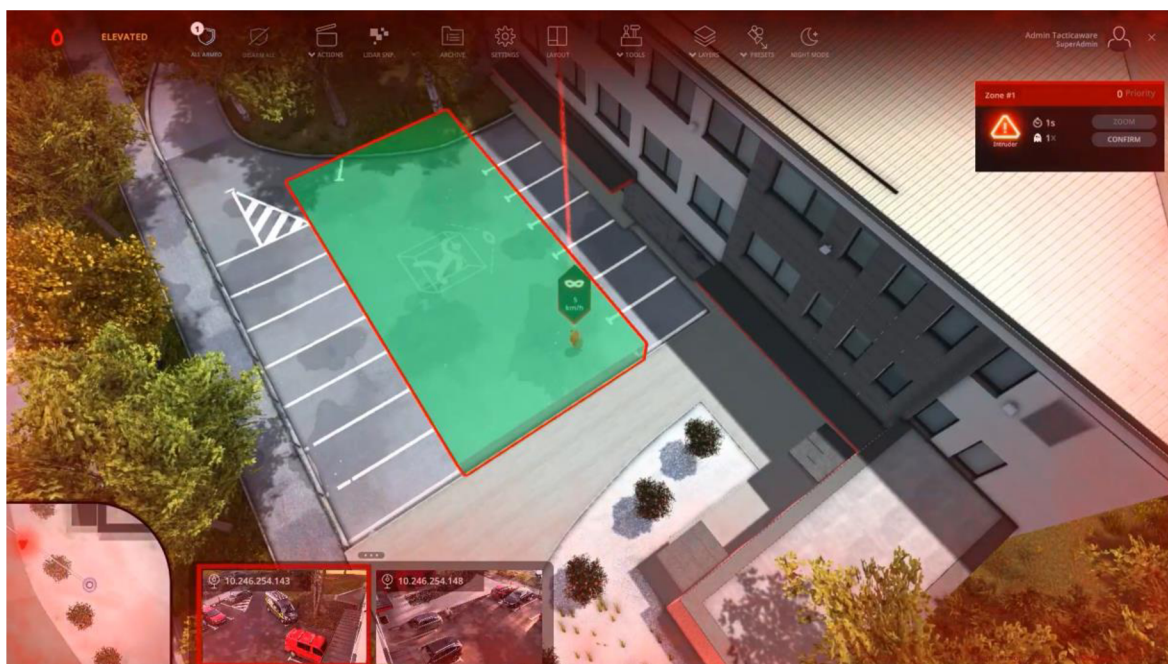
5.2 Accur8vision

Accur8vision je pokročilý volumetrický bezpečnostní systém, vyvíjený českou firmou Tacticaware, který se skládá ze serveru, na kterém se provádí klasifikace, jiné detekční procesy a klient pro operátory, který je určen na monitoring oblasti. Na rozdíl od konkurence sama lidary nevyrobí, ale podporuje všechny lidary od všech větších výrobců. Současně firma podporuje i integraci jiných systémů, jako jsou CCTV softwary nebo PZS. Accur8vision současně od lidarů, podporuje i integraci PTZ nebo non PTZ kamer, které jsou pilotovány automaticky, vždy na narušitele. Jako vůbec první systém pilotuje kamery, na základě polohy narušitele v prostoru a nikoli podle klasifikace obrazu z kamer.

Systém, díky využívání jako primárního senzoru lidary, dokáže nejen současně fungovat jako perimetrický a prostorový detekční systém, ale rovněž dokáže střežit, jako jeden z prvních komerčních bezpečnostních systémů i vzdušný prostor před špionážními nebo sebevražednými drony. Jediné nároky na systém, je dostatečně robustní datová síť, a výkonný server. Pokud by na síti bylo velké množství lidarů a switche by nedokázaly distribuovat dostatečně rychle tak velké množství dat, je možné nasadit detekční nody. Jedná se o zařízení navržené firmou, který redukuje data lidarů a na server se ke

zpracování pošle pouze rozdíl bodů od klidového stavu. Díky tomu se jedná o velmi snadno rozšiřitelný systém.

Na rozdíl od klasických zabezpečovacích systémů se nastavení a kalibrace celého systému Accur8vision provádí pomocí klientské části. V klientu se rovněž vytváří zóny, které označují střeženou oblast rovněž i jejich typ, detekci a další parametry. To umožňuje velmi snadné a rychle zprovoznění. Lze zde nastavit, pokud systém disponuje požadovaným hardwarem, manuální akce, které může snadno a rychle spustit operátor, nebo automatické akce, které systém vykoná sám, při obdržení signálu. Systém je aktuálně nasazen například na jaderných elektrárnách Dukovany a Temelín, kde střeží vzdušný prostor nad klíčovými budovami elektráren, rovněž tak na dalších desítkách instalací po celé Evropě. (Accur8vision for intelligent security systems, 2022)



Obrázek 12 Ukázka klientské části A8V při poplachu (Accur8vision, 2022)

5.2.1 MIND

Jedná se o mobilní verzi zabezpečovacího systému Accur8vision, který je určen pro rychlé nasazení a zajištění oblastí v terénu. Konstrukce je designovaná do bojových podmínek a její základ je tvořen z odolného PELO kufru oplátovaným ocelovými pláty, ve které se nachází veškeré nutné komponenty. Systém disponuje interními bateriemi o výdrži 16 hodin a pomocí konektoru je možné systém napájet i z externí baterie o stejné výdrži. Dále systém obsahuje 4 kamery s pokrytím 360° a lidarem OS1-32, OS1-64 nebo OS1-128, podle potřeby zákazníka. Pro vzdálené připojení do systému zde slouží Wi-Fi síť s externí anténou pro zvýšení dosahu.

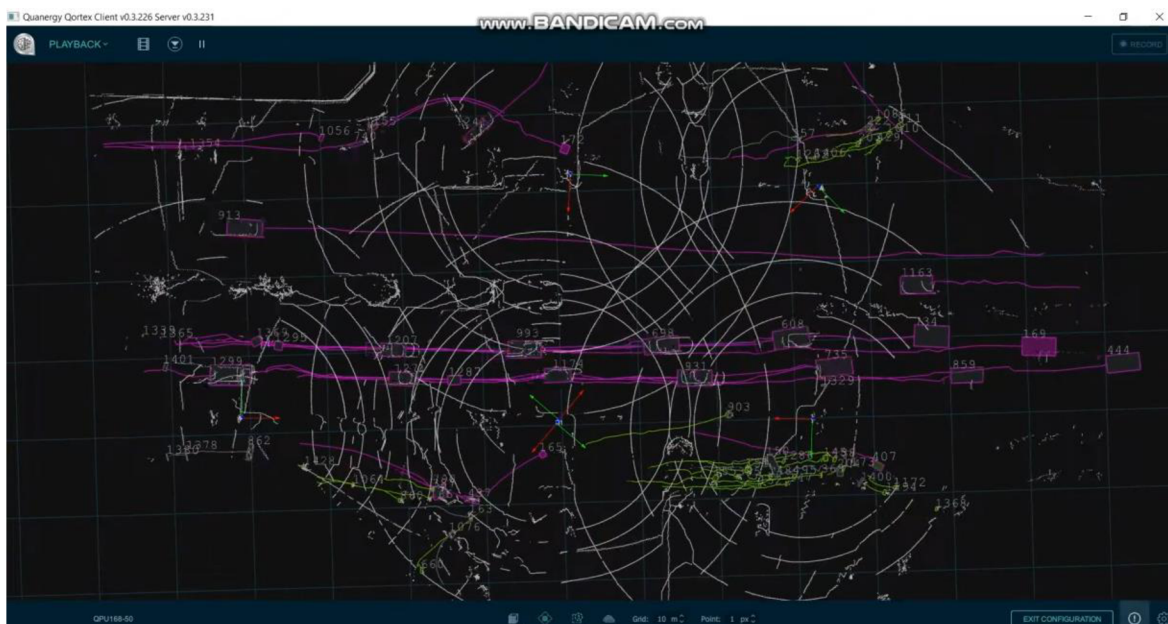


Obrázek 13 MIND (vlastní)

5.3 Ostatní lidarové bezpečnostní systémy

Po úspěchu skupiny Hexagon 3D Surveillance, se na trhu objevily další konkurenční firmy, které také začaly vytvářet vlastní bezpečnostní systémy.

Quanergy, je americký výrobce lidarů, který se zaměřuje na vývoj stejnojmenného softwarů, využívající právě jejich lidary. Softwary jsou určeny pro automatizaci výrobních procesů v průmyslu, monitorování infrastruktury a dopravy ve městech nebo nově i zabezpečovací systém. Na rozdíl od konkurence, jejich systém není tak propracovaný a nedisponuje tak pokročilou detekcí, klientem a nepodporuje tak široké spektrum zařízení jako konkurence. (Quanergy, 2023)



Obrázek Ukázka Quanergy bezpečnostního systému QORTEX (Quanergy Systems, 2020)

Gemini je nový volumetrický systém, který vyvíjí společnost Ouster. Oznámený byl v době psaní mé diplomové práce a bližší informace, ještě nejsou odhaleny veřejnosti. Jediné ukázky, které přidali na jejich web, pouze naznačují silnou inspiraci od konkurenčního systému Accur8vision. (Ouster, 2023)

6 Umělá inteligence

Jeden z nejperspektivnějších oborů v informatice je umělá inteligence (Artificial Intelligence, AI). Tento obor se zabývá tvorbou počítačových programů a systémů, které jsou schopny simulovat lidskou inteligenci a jsou schopny se učit z vlastních chyb. Protože pro samotnou inteligenci neexistuje jedna správná definice, existuje i pro umělou inteligenci celá řada definic. Podle definice evropského parlamentu (2021) je umělá inteligence: „*schopnost strojů napodobovat lidské schopnosti, jako je uvažování, učení se, plánování nebo kreativita.*“ V minulosti se systémy hodnotily jako inteligentní, na základě Turingova testu. Test však nedokáže pokrýt všechny aspekty umělé inteligence a byl zatížen velmi subjektivním hodnocením. (Umělá inteligence: definice a využití, 2020)

Tyto systémy a programy v poslední době nabraly na velké popularitě a integrují se takřka do všech možných systémů, jako jsou například autonomní auta, chatboty v zákaznických linkách, rozpoznávání obrazů nebo i jejich samotnou tvorbu, což se ještě do nedávna zdálo, jako nejsilnější lidská stránka. S konkrétní příklady aplikované umělé inteligence, se můžeme setkat například v počítačových hrách, kde se vývojáři snaží vytvořit postavy, které neřídí samotný hráč, ale počítač a mají za úkol navodit hráči pocit, že se pohybuje v opravdovém světě, ve kterém žijí skuteční lidé. Nebo u robotických vysavačů, které musí naplánovat cestu tak, aby pokryli celou plochu a současně se přizpůsobit svému okolí. (Louthánová, 2021)

I přes fakt, že neexistuje konkrétní jedna přesná definice umělé inteligence, je obecná shoda v dělení umělé inteligence, podle složitosti a pokrytí problémů, které je schopna řešit. (Umělá inteligence pro každého, 2023)

Slabá AI dokáže řešit konkrétní úlohy v úzce specifikované oblasti lépe než člověk. Všechny současně vytvořené umělé inteligence lidstvem, spadají do této kategorie. I když se zdá, že systém se dokáže rozhodovat v reálném čase sám a „mysl“, ve skutečnosti pouze koordinuje pár omezených procesů a rozhoduje v mezích předem určeného rámce.

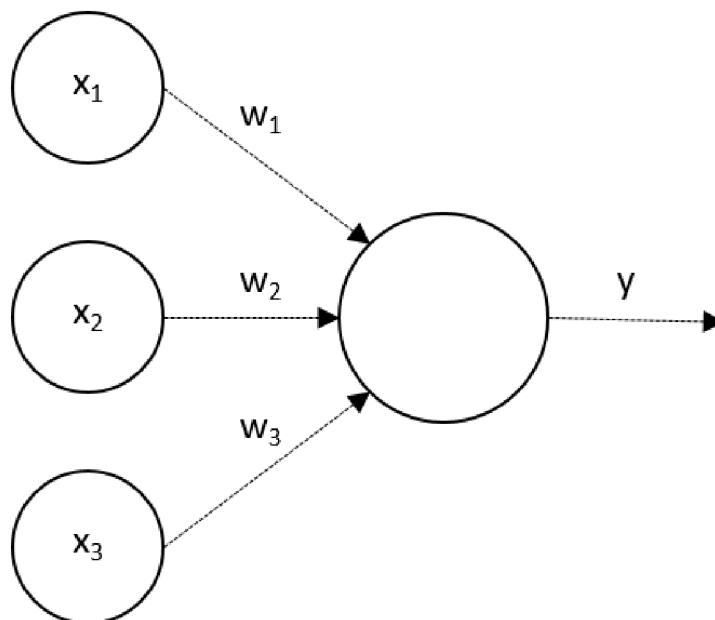
Obecná AI představuje úroveň inteligence minimálně srovnatelné s člověkem. Systém by měl být schopen učit se z jakýchkoliv typů úkolů a rozhodovat se podle vlastního uvažování s ohledem na vlastní předešlé zkušenosti a se schopností adaptovat se na řešení zcela nových problémů. Tento typ umělé inteligence, bývá zobrazován ve filmech a doposud se nepodařilo zcela vytvořit tento typ umělé inteligence.

Umělá super inteligence je nejvyšší úroveň umělé inteligence. Tento stupeň inteligence, by překonal všechny největší lidské kapacity ve všech ohledech. Tato úroveň inteligence zatím neexistuje a je spíše předmětem filozofických debat. (Umělá inteligence, 2023)

6.1 Neuronová síť

Problém umělé inteligence je, že nedokáže řešit některé zadané úlohy tak snadno, jako živé organismy. Proto výzkumníci začali tyto organismy napodobovat a vznikla tak nová kategorie umělé inteligence, takzvaná soft computing, kam spadají neuronové sítě, fuzzy logika nebo evoluční algoritmy. Výhodou soft computingu je, že výpočty aproximuje. Díky tomu získáváme výsledky mírně zkreslené, ale použitelné, které se uplatňují u řešení neřešitelných, nebo časově velmi náročných problémů pro současný hardware. (Contributor, 2023)

Neuronová síť je inspirována strukturou lidské neuronové soustavy. Hlavní prvkem je neuron, označovaný také jako perceptron. Jednotlivé neurony, jsou navzájem propojeny s ostatními a předávají signál dalším neuronům. Každý neuron může mít více vstupů, ale jen jeden výstup, který však může posílat současně více neuronům najednou. Vstupy mohou být vstupní data z venku, nebo výstupy z jiných neuronů. Každý vstup má určitou váhu, se kterou vstupuje do neuronu, kde se každý vstup vynásobí s hodnotou váhy.



Obrázek 14 Schéma neuronu (Durčák, 2017)

Aktivace neuronu, se následně provádí na základě vzorce 2. Nejprve všechny vstupy x vynásobí s odpovídající vahou w . Jedná se o přenosovou funkci a od součtu vážených vstupů se následně odečte práh neuronu θ . Výsledná hodnota se pak předá ke zpracování předem definované aktivační funkci f neuronu a pokud je signál dostatečně silný pošle jej na výstup neuronu jako y . (Durčák, 2017)

$$y = f\left(\sum_{i=1}^n w_i \cdot x_i - \theta\right)$$

(2)

Existuje mnoho typů aktivačních funkcí, zde jsou některé z nejčastěji používaných.

Sigmoidní funkce převádí vstupní signál na hodnotu mezi 0 a 1, což znamená, že aktivuje neuron pouze tehdy, když je vstupní signál dostatečně silný.

Tanh funkce převádí vstupní signál na hodnotu mezi -1 a 1. Podobně jako sigmoidní funkce aktivuje neuron pouze tehdy, když je vstupní signál dostatečně silný.

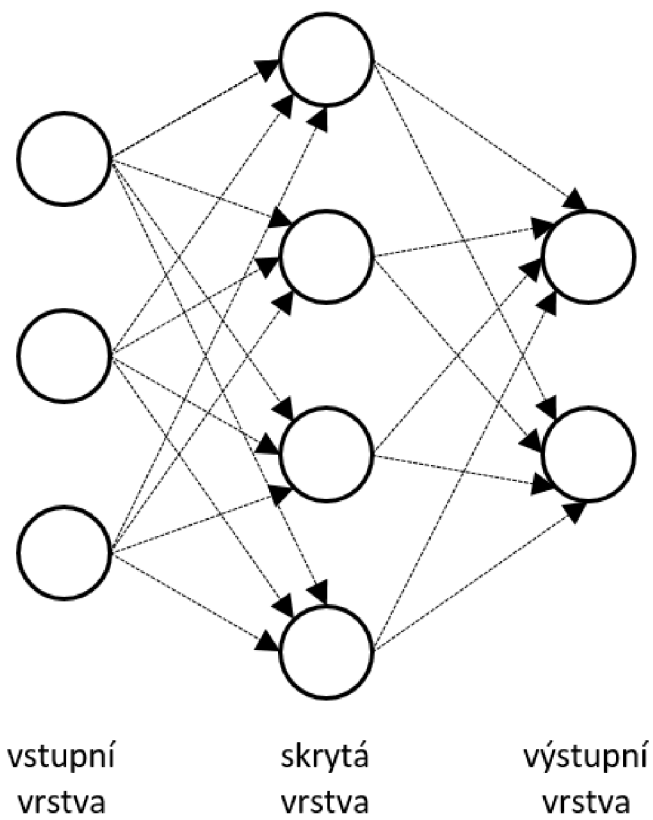
ReLU (Rectified Linear Unit) vrací vstupní signál, pokud je kladný, a 0, pokud je záporný, lze ji zapsat také jako $f(x) = \max(0, x)$. Tato funkce je velmi populární pro svou jednoduchost a vysokou účinnost.

Leaky ReLU funkce je podobný jako ReLU, ale místo toho, aby vracel 0 pro záporné vstupní signály, vrací malou zápornou hodnotu. To pomáhá předejít "mrtvým" neuronům, které by jinak zůstaly neaktivní.

Softmax funkce se používá pro klasifikační úlohy. Převádí vstupní signály na pravděpodobnosti, že vstupní data patří do různých tříd. Součet pravděpodobností všech tříd je vždy roven 1.

Tyto funkce se používají v závislosti na konkrétním problému a architektuře neuronové sítě. Každá z nich má své výhody a nevýhody a volba správné funkce může mít významný vliv na výkon sítě.

Při řešení komplexnějších úloh, je za potřebí využít více než jeden neuron. Tyto neurony, jak jsem již zmínil, jsou navzájem propojeny. Neurony se pak propojují vzájemně v podobě vrstev, kde vstupní vrstva slouží k příjmu vstupních dat, výstupní vrstva k předání výsledku celé neuronové sítě a skryté vrstvy, které zprostředkovávají vyhodnocení. (Valkovič, 2023)



Obrázek 15 Vizualizace vrstev neuronové sítě (Durčák, 2017)

7 Praktická část

Praktická část bude rozdělena do tří hlavních částí, kde bude postupně popsán proces přípravy Accur8vision systému a jeho nastavení, který bude sloužit jako hlavní zdroj dat, pro neuronovou síť. V následující části bude popsán, jakým způsobem, byla trénovací data získána, z jakých zdrojů a rovněž popis vlastní anotační platformy, která byla vytvořena pro tyto účely. Poslední část, se bude zabírat návrhem neuronové sítě a jejich výsledky.

7.1 Příprava serveru

Na serveru s operačním systémem Ubuntu 22 bylo nainstalována a nakonfigurována služba Accur8vision s Postgresql databází, která je využívána pro tuto práci. Server je pro práci klíčový, protože bez něj by nebylo možné získávat point cloudy, které by odpovídaly skutečným podmínkám a současně slouží také pro vizualizaci.

Na Accur8vision systém byly nakonfigurovány základní parametry v podobě přidání a umístění lidarů do scény, který systému poskytuje údaje o okolí v podobě point cloudu. Společně s lidarem bylo nutné vygenerovat snapshot, který se využívá jako maska, podle které se porovnávají data z lidarů v reálném čase. Rozdíl se následně zobrazí ve scéně rozlišnou barvou a je nad ním prováděn clustering. Dále byla přidána do scény detekční zóna, ve které se provádí již zmíněný clustering. Tento clustering funguje na jednoduchém principu. Body jsou sdružovány do celku, pokud jsou od sebe vzdáleny horizontálně nebo vertikálně v předem definované vzdálenosti a celek samotný disponuje větším nebo stejným počtem bodů, který je určen v zóně. Pro systém byly nastaveny následující hodnoty clusteringu:

- Horizontální vzdálenost – 30 cm
- Vertikální vzdálenost – 180 cm
- Minimální počet bodů – 6 bodů

Hodnoty byly zvoleny s ohledem na skutečnost, že v zóně bude systém detekovat výskyt člověka a při měření se prokázaly jako nejvhodnější. Měření bylo prováděno na detekci člověka při vzdálenosti 20 metrů od lidarů. Při využití nižších hodnot horizontální vzdálenosti, se člověk začal rozpadat na více clusterů, naopak při vyšších hodnotách, byl člověk častokrát spojován s okolními clustery do jednoho. U vertikální vzdálenosti, se tento problém neprokával v takové míře, ale pro tvorbu jednoduchého clusteru se tato hodnota prokázala jako nejvhodnější, jelikož je nejvýhodnější použít největší možné hodnoty, které nezpůsobují falešné clusterování s okolím.

Horizontální vzdálenost [cm]	Počet clusterů na člověka	Clustery s okolím	Vertikální vzdálenost [cm]	Počet clusterů na člověka	Clustery s okolím
10	3	0	100	2	0
20	2	0	120	2	0
30	1	0	140	1	0
40	1	1	160	1	0
50	1	1	180	1	0
60	1	2	200	1	1

Tabulka 7 Výsledky měření clusteringu (vlastní)

Maximální horizontální vzdálenost mezi body, které systém bude slučovat do clusteru, odpovídá přibližně polovině délky šířky mezi rameny člověka a maximální vertikální vzdálenost zase odpovídá přibližné výšce člověka. Rovněž systém spojuje i body do clusteru, pokud jsou body od sebe vzdáleny v menší vzdálenosti, než bylo definováno v zóně systému.

Pokud by se tedy v zóně objevily body, které by splnily podmínky clusteringu, v zastřežené zóně se vyvolá poplach a příslušné body, se zapíše automaticky do databáze do tabulky *point*. V tabulce se ukládá vždy jeden záznam pro jeden bod ve formátu, který je zobrazen níže.

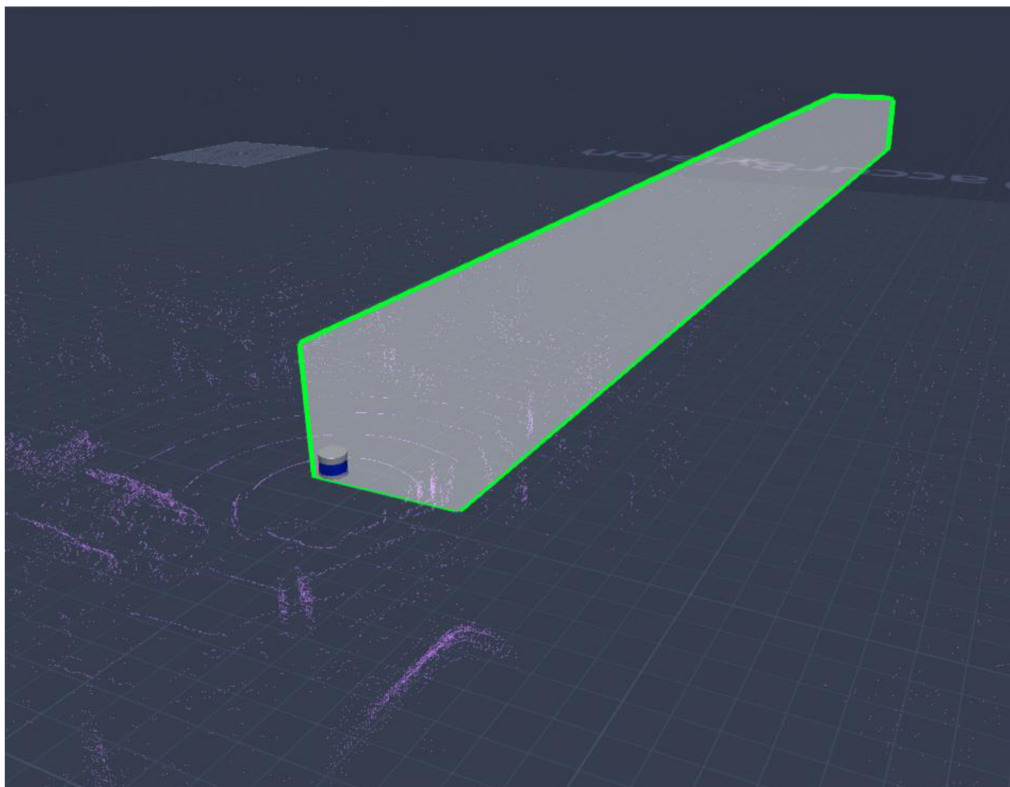
Column	Type	Collation	Nullable	Default
bbox_id	integer			
detection_time	timestamp without time zone		not null	
inside_zone	boolean		not null	
position	vector3		not null	

Obrázek 16 Schéma tabulky point (vlastní)

- bbox_id – Odkazuje na přesný cluster, ke kterému bod patří.
- detection_time – Reprezentuje přesný čas, kdy byl bod zaznamenán.
- inside_zone – Reprezentuje stav, zda se bod nacházel v době zaznamenání v zóně.
- position – Reprezentuje trojrozměrný vektor v trojrozměrném prostoru (x, y, z) a odkazuje na přesnou polohu bodu v době zaznamenání. Jedná se v podstatě o jednorozměrné pole, kde se ukládá pozice bodu v následujícím formátu
 - X – souřadnice vodorovné osy
 - Y – souřadnice svislé osy
 - Z – souřadnice hloubkové osy

Pozice bodů z lidarů, se vždy odvíjí od polohy samotného lidarů ve scéně. Podle pozice lidarů, se tedy odvíjí i souřadnice bodů, které produkuje.

Na klientské části systému, je následně možné vidět vizualizaci zóny, lidarů a snapshotu ve scéně. Snapshot, je zde zobrazen jako point cloud fialových bodů. Rovněž je vidět model lidarů, konkrétně se jedná o lidar Ultra Puck, jehož volba bude objasněna v nadcházející kapitole a detekční zóna, která podle barvy okrajů indikuje, že střeží prostor, který pokrývá.



Obrázek 17 Vizualizace nastavení systému (vlastní)

7.2 Příprava dat

Při přípravě dat, byl brán ohled na hlavní příčiny vyvolávání poplachu u systému Accur8vision. Třemi hlavními příčinami vyvolání poplachu je narušitel, u kterého se poplach spouští úmyslně, nepříznivé počasí a projíždějící auta, nebo velké předměty ve střežených zónách, způsobené především nesprávným snapshotem systému, které vyvolávají falešné poplchy. Dále budou popsány jednotlivé postupy, jak byla data pro jednotlivé kategorie získána.

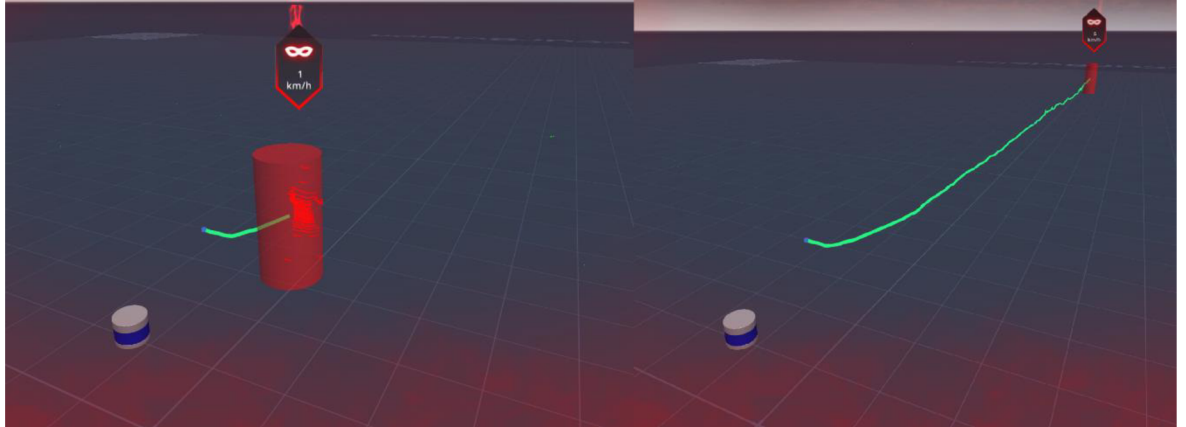
Jak již bylo zmíněno v předešlé kapitole, systém využívá lidar type Ultra Puck od firmy Velodyne, jehož parametry jsou vypsány v tabulce č. 2. Jeho hlavní přednost je jeho velký dosah, který je dosažen nelineární distribucí paprsků, které jsou nahuštěny na středu,

díky kterým má vyšší dosah a rovněž počtem kanálů, díky kterém generuje dostatečně velké mračno bodů pro přesnou detekci.

Pro tvorbu učících a validačních dat pro neuronovou síť, je klíčové disponovat velkým množstvím dat, které rovněž odpovídají skutečnosti, které chceme neuronovou síť naučit. Kvůli tomu, byla zaznamenána celková komunikace lidarů se serverem. Díky tomu je možné pustit data z lidarů kdykoliv v podmínkách, jaké vyžadujeme, s možností filtrování nežádoucího hluku v podobě projíždějících aut, nebo pohyb větví u stromu způsobené větrem. Naopak, pokud budeme potřebovat data, která by naopak nerepresentovala člověka, jelikož neuronovou síť nemůžeme pouze učit na jednom typu dat, můžeme snadno a rychle systém pozměnit a začít nahrávat data jiného typu.

Jelikož tento typ lidarů komunikuje a posílá point cloud po síťovém rozhraní, byl použit nástroj *tcpdump*, který zaznamenává packety na vybraném síťovém rozhraní. Po zaznamenání reálných dat z lidarů a ukončení nahrávání, se automaticky vytvoří file ve formátu *.pcap*, který obsahuje veškerou komunikaci s časovým údajem zachycení jednotlivých packetů. Při nahrávání komunikace, se však nahrává všechna komunikace, která v ten čas probíhá na daném síťovém rozhraní, proto je nutné provést dodatečnou filtraci packetů. Jelikož lidar posílá point cloud ve formě UDP packetů a současně je posílá na známý port, je možné tyto packety odfiltrovat v programu Wireshark, který vybrané packety uloží do nového *.pcap* filu. Tato filtrace dat, byla nutná, jelikož se velikost filu zmenšila přibližně o třetinu a současně, pro opětovné přehrání nahrané komunikace bude využit nástroj *tcpreplay*, který má problém s rozdílnými velikostmi a typy packetů při přehrávání. Za pomoci *tcpreplay*, je tedy možné, přehrát již odfiltrovaná data a simulovat tak reálnou situaci v plně kontrolovaném prostředí, což je pro práci klíčové.

Zaznamenaná situace simuluje pochod člověka od lidarů stále rovně vpřed, pokračoval po přímce a neodbočoval. V důsledku toho se postupně zmenšoval počet bodů, avšak stále byla zachována silueta člověka.

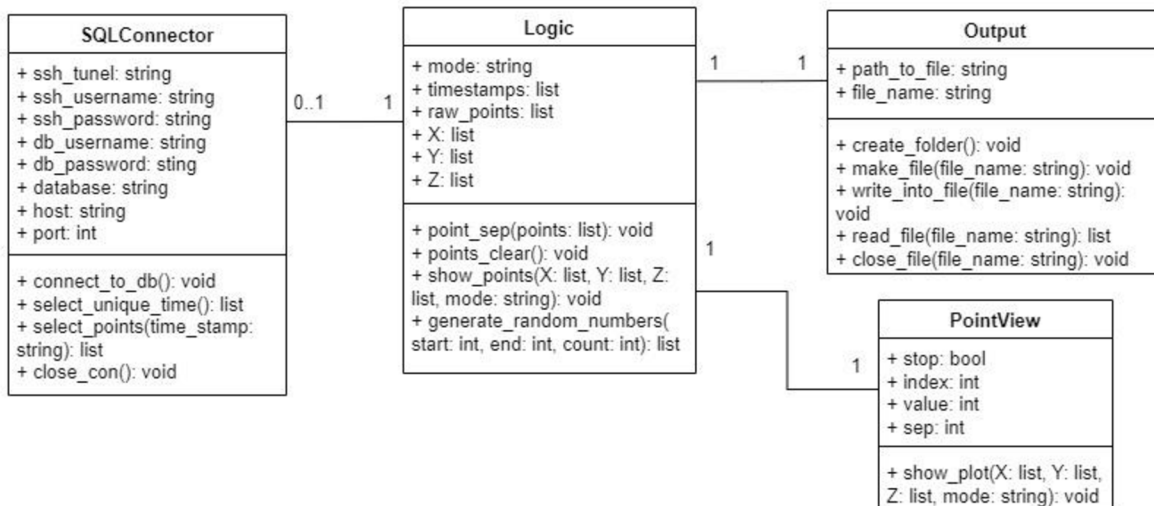


Obrázek 18 Záznam člověka v A8V (vlastní)

Pro účely vytváření trénovacích a testovacích dat člověka, byla vytvořena vlastní anotační platforma. Program anotační platformy byl napsán v jazyce Python, verze 3.10 a byly využity následující knihovny.

- Matplotlib – Slouží pro zobrazování bodů ve 3D a poskytuje interaktivní uživatelské rozhraní.
- SSHTunel – Umožňuje navázání zabezpečeného kanálu pro komunikaci.
- psycopg2 – Umožňuje práci s relačními databázemi PostgreSQL

Program se skládá z následujících tříd, které budou následně podrobně popsány.



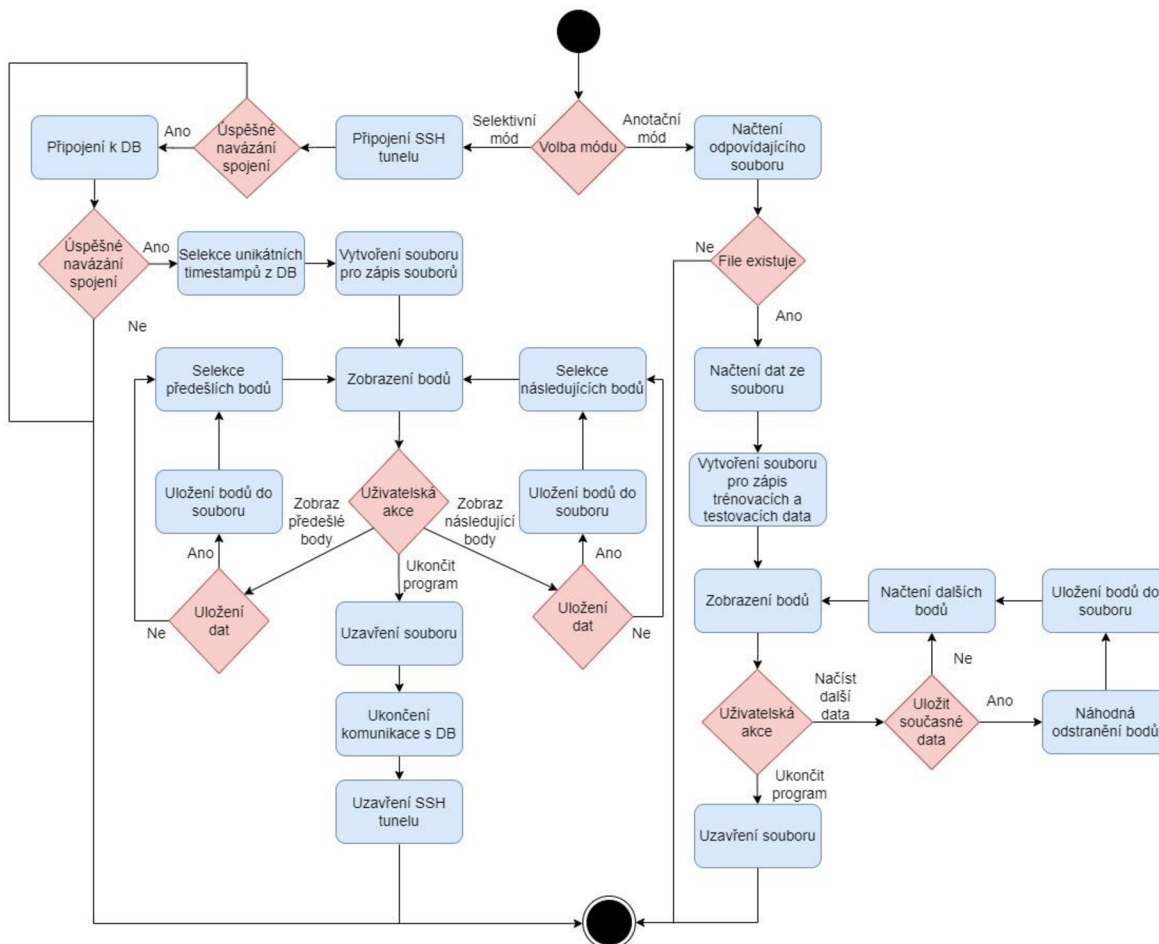
Obrázek 19 Class diagram (vlastní)

SQLConnector, slouží k navázání spojení s databází systému Accur8vision. Třída současně slouží pro selekci vybraných dat z databáze.

Třída **PointView** se využívá jako grafické uživatelské rozhraní anotační platformy. Poskytuje grafické zobrazení bodů ve 3D a umožňuje uživateli interagovat s anotační platformou.

Output třída se stará o tvorbu odpovídajících filů a současně i čtení dat a zapisování bodů do filů.

Třída **Logic** následně slouží jako celkový ovladač anotační platformy, který vznáší požadavky na ostatní třídy a zpracovává pro ně data.



Obrázek 20 Vývojový diagram (vlastní)

Anotační platforma, před spuštěním, vyžaduje výběr módu, ve kterém bude operovat. Při spuštění selektivního módu, program naváže SSH spojení se serverem, a při úspěšném připojení se následně provede připojení k databázi systému Accur8vision. Program, po úspěšném připojení k databázi, provede *DISTINCT* selekci všech časů z tabulky *point* a ve formátu timestamp je následně uloží do stejnojmenného pole, poté vytvoří složku a soubor, kam se budou ukládat uživatelem vybraná data. Údaje z timestampů se následně používají v cyklu pro zobrazování jednotlivých point cloudů ze systému Accur8vision. Při selektování jednotlivých point cloudů, je v podmínce brán ohled na *bbox_id*, díky kterému program dostane pouze body, které korespondují ke clusteru, který splňuje podmínky, které byly nastaveny v předešlé kapitole.

```
SELECT position
FROM   a8v.PUBLIC.point
WHERE  bbox_id IS NOT NULL
       AND detection_time = Make_timestamp({str(time).translate(timestamp_mask)});
```

Následně dochází k zobrazení bodů v grafickém rozhraní, kde si s nimi může uživatel rotovat a přibližovat, a rozhodnout, zda tyto body uloží do filu a nechá si zobrazit další body, popřípadě předchozí body. Pokud uživatel zvolí, že chce program opustit, k názvu, do kterého se zapisovala data, se přidá aktuální čas a uzavře se. Následně se ukončí připojení k databázi a SSH tunel a program skončí.

Pokud si uživatel zvolí anotační mód, načtou se data ze zvoleného filu. Následně se vytvoří nový file, který bude sloužit pro zápis již trénovacích nebo testovacích dat pro neuronovou síť. Z načteného filu se následně zobrazí body v uživatelském rozhraní a uživatel může rozhodnout, zda daný point cloud bude chtít uložit, pod jakou kategorií budou spadat a popřípadě může i zvolit, kolik procent z daného point cloudu je možné odebrat, pro stálé zachování dané struktury.

Motivace k tomuto chování byla zvolena s ohledem na velké množství dat, které neuronová síť potřebuje na učení. Na rozdíl od klasifikace na základě obrázků, je možné u point cloudů velmi snadno manipulovat s body. Pokud zachováme přibližně stejnou strukturu bodů, je možné z původního point cloudu o velkém množství bodů, vytvořit nespočet dalších point cloudů s menším počtem bodů, které však budou ve velmi podobné struktuře.

Výše popsanou augmentaci dat, program provádí vymazáním náhodných bodů z listu bodů. Na generování jedinečných náhodných čísel, využívá program metodu *generate_random_numbers*, která generuje do setu specifikovaný počet náhodných čísel, který se odvíjí od uživatelem zvoleným procentem a počtem bodů, v rovněž uživatelem definovaném rozsahu.

```

def generate_random_numbers(self, start, end, count):
    numbers = set()
    while len(numbers) < count:
        numbers.add(random.randint(start, end))
    return list(numbers)

```

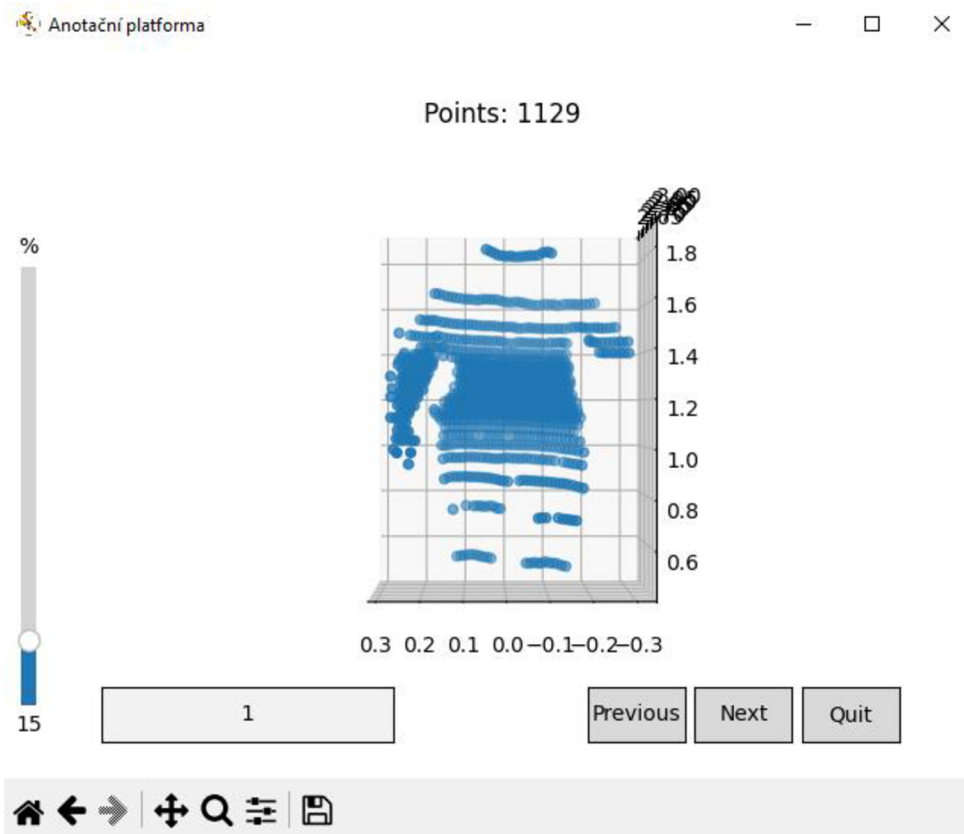
Tento set se následně použije pro vymazání odpovídajících prvků z listu bodů a proces se provede, v závislosti od počtu celkových bodů několikrát. Tím systém dokáže z jednoho point cloudu vygenerovat desítky dalších point cloudů, které se zapisují do filu. Tento proces se provádí pro každý point cloud ze zvoleného filu, podle uživatelských požadavků.

```

for number in range(len(self.X)//10):
    points_clone = points.copy()
    sep = self.generate_random_numbers(
        0,
        len(points_clone)-1,
        int(len(points_clone)*self.view.sep))
    sorted_sep = sorted(sep, reverse=True)
    for s in sorted_sep:
        del points_clone[s]
    self.write_points(
        points_clone,
        self.view.value,
        "Human_TD")

```

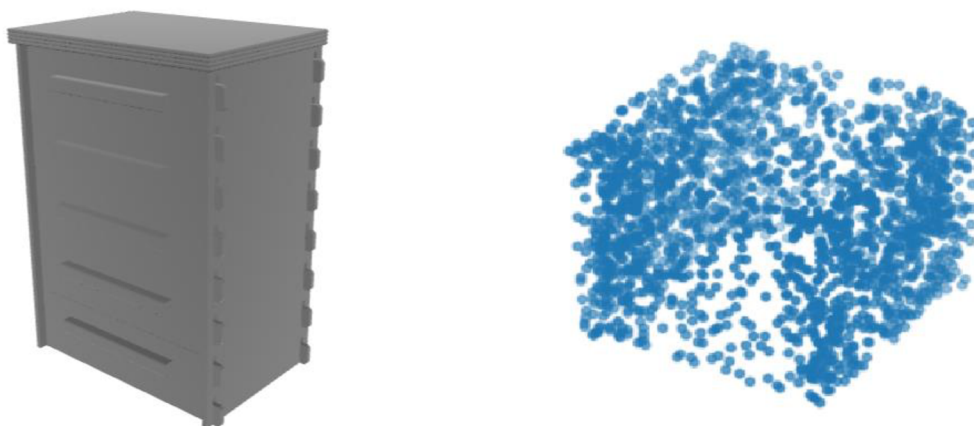
Pokud uživatel zvolí, že chce program opustit, file, do kterého se zapisovaly point cloudy, se uzavře a k názvu se rovněž přidá aktuální čas. Následně, má tedy uživatel k možnosti sadu trénovacích dat, které si zvolil. Data jsou do souboru ukládaný ve formě listu, kde první prvek reprezentuje list bodů z point cloudu a prvek druhý zase kategorii, která byla zvolena uživatelem přes grafické rozhraní v textovém poli.



Obrázek 21 GUI anotační platformy (vlastní)

Pro rozšíření datové sady reprezentující statické objekty a auta, byla použita veřejně dostupná knihovna ModelNet, kterou spravuje univerzita Princeton. Konkrétně byla použita verze ModelNet10, která disponuje 10 typy objektů, ve více než 150 provedení, ve formátu shapefile. Pro tvorbu dat byly použity modely předmětů, které reprezentují statické předměty a auta, které jak již bylo zmíněno dříve, jsou jedním z hlavních důvodů vzniku falešných poplachů.

Z jednotlivých provedení předmětů, bylo tedy nutné, převést stávající shapefile na point cloudy, reprezentující daný předmět. To bylo provedené za pomoci knihovny *Trimesh*, kdy se nejdříve jednotlivá provedení načetla a následně se z nich vytvořil point cloud funkcí *load()* o daném počtu bodů. Jednotlivé objekty, byly však vytvořeny v jiném měřítku než point cloudy, které jsem zaznamenal přes anotační platformu z Accur8vision systému, proto bylo provedeno dodatečné zmenšení point cloudů na desetinu.



Obrázek 22 Převod model souboru do point cloudu (vlastní)

Pro rozšíření datové sady point cloudu, který by reprezentoval nepříznivé počasí, až s podivem stačilo vygenerovat náhodné mračno bodů, které velmi přesně reprezentují situace za nepříznivého počasí, jako jsou páry nebo mlhy, v systému Accur8vision. Pro tvorbu těchto point cloudů, stačila jednoduchá funkce *random.uniform()*, pro souřadnice X, Y a Z.

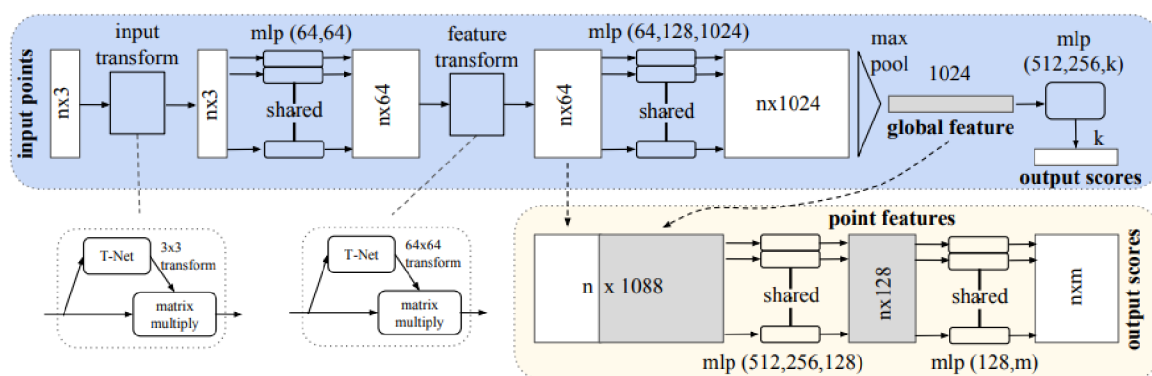
Po sdružení všech typů point cloudů, za využití výše popsaných metod, bylo dosaženo přibližně 14000 vzorků, které byly následně, společně s označením jednotlivých point cloudů, do dvou polí, *train_dataset* a *test_dataset*, v poměru 4:1 pro trénování a validaci výsledků neuronové sítě. Rovněž každý point cloud disponuje odpovídající kategorií ve sdružených polích *train_categories* a *test_categories*.

Ještě před vstup point cloudů do neuronové sítě, bylo provedeno náhodné promíchání položek těchto listů, společně s tzv. jitterem. Jitter je technika, která se používá při práci s point cloudy, kdy dochází k mírnému posuvu bodů, tak aby došlo k co možná nejmenšímu překrytu bodů.

7.3 Model Neuronové sítě

Při testovacím spuštění neuronové sítě, bylo zaznamenáno velmi pomalé učení neuronové sítě. Délka první epochy se pohybovala okolo 5 minut, což bylo způsobené, že neuronová síť využívala pouze výpočetní kapacity procesoru. Z tohoto důvodu, bylo pro neuronovou síť vytvořeno a nastaveno v softwaru Anaconda speciální prostředí, které usnadňuje výběr interpretu jazyka Python a správu balíčků. Pro projekt byla využita knihovna TensorFlow verze 2.10, která umožňuje, jako poslední verze využívat grafické procesory pro rychlejší trénování neuronových sítí pro operační systém Windows 10. Pro využití grafického procesoru bylo nutné nainstalovat příslušné ovladače. Při využívání tohoto prostředí a použití stejné neuronové sítě, se délka první epochy snížila na pouhých 19 sekund.

Neuronová síť byla navržena podle modelu neuronové sítě PointNet, která je navržena pro klasifikaci objektů, segmentaci dílů nebo sémantickou segmentaci na základě point cloudů. V této práci bude využita pro vyhodnocení point cloudy, které byly vytvořené v předchozí kapitole. Celé schéma neuronové sítě PointNet, je zobrazené na obrázku 23 uvedený níže, v práci však byla použita pouze poloviční hustota vrstev, s ohledem na nízký počet klasifikovaných tříd.



Obrázek 23 Struktura neuronové sítě PointNet (Qi et al., 2017)

Velká výhoda PointNet neuronové sítě, jak je již zřejmé na vstupní matici je, že na rozdíl od jiných neuronových sítí, PointNet neuronová síť zpracovává přímo point cloudy

a data, tak nemusí procházet voxelizací. Voxelizace je proces, při kterém dochází k převedení bodů z point cloudu na voxely ve voxelové mřížce, která na rozdíl od point cloudu, uchovává informace o tvaru objektu diskreditovaně a neumožňuje tak zachytit detailní geometrii a rovněž jsou náchylné na prostorové skreslení.

Prvním krokem v neuronové síti PointNet, je transformace vstupů, která slouží k předzpracování vstupního point cloudu, před jeho zpracováním v hlavní části síti. Transformace vstupu, se skládá ze dvou hlavních částí. První část je T-Net (Transformation Network), která se skládá z několika plně propojených vrstev a slouží k naučení transformační matice pro rotaci, škálování a posun celého vstupního point cloudu. Druhá část je samotný transformační blok, který používá naučenou transformační matici k transformaci každého bodu ze vstupního point cloudu. Výstupem, je pak matice 3×3 , která se aplikuje na vstupní matici $n \times 3$ a transformuje tak vstupní point cloud na kánonický tvar. Cílem transformace vstupu, je zajistit, že vstupní point cloud je invariantní vůči rotaci, škálování a posunu, což zlepšuje schopnost PointNet sítě rozpoznávat a klasifikovat objekty a scény bez ohledu na jejich polohu a orientaci v prostoru.

Samotná transformační síť, se skládá z několika propojených konvolučních a dense vrstev. V programu jsou konvoluční a dense vrstvy definovány následovně.

Definovaná vrstva `conv_bn()`, provádí konvoluci pomocí jednorozměrné konvoluční vrstvy s daným počtem filtrů a jádrem o velikosti 1. Poté se aplikuje normalizace dávkou `BatchNormalization` s nastaveným koeficientem trénování `momentum` na hodnotu 0, což znamená, že se použije standardní trénování dávkou. Nakonec se na výstup aplikuje `ReLU` aktivace.

```
def conv_bn(x, filters):
    x = layers.Conv1D(filters, kernel_size=1, padding="valid")(x)
    x = layers.BatchNormalization(momentum=0.0)(x)
    return layers.Activation("relu")(x)
```

Definovaná vrstva `dense_bn()` používá jako vstup x , který představuje výstup z předchozí vrstvy neuronové sítě. Následně použije metodu `Dense()`, aby vytvořila plně

propojenou vrstvu s počtem neuronů uvedený v proměnné *filters*. To znamená, že každý neuron v této vrstvě je propojen s každým neuronem v předchozí vrstvě. Výstup z této vrstvy je následně normalizován pomocí *BatchNormalization()*, což zlepšuje stabilitu a rychlost učení sítě tím, že normalizuje výstup každé jednotlivé vrstvy neuronové sítě před použitím aktivační funkce. Nakonec je použita aktivační funkce ReLU.

```
def dense_bn(x, filters):  
    x = layers.Dense(filters)(x)  
    x = layers.BatchNormalization(momentum=0.0)(x)  
    return layers.Activation("relu")(x)
```

Samotná T-Net, se skládá z výše definovaných vrstev. Nejprve se vstupní point cloud zpracuje třemi konvolučními vrstvami o 32, 64 a 512 neuronech. Následně se nad výstupem z těchto vrstev provede metoda *GlobalMaxPooling*, která pro každý bod z point cloudu vybere nejvyšší hodnotu z vektorů v každé dimenzi. Na výstupy se pak aplikují dvě *dense_bn* vrstvy o 256 a následně 128 neuronech. Nejdůležitější částí T-Net je poslední vrstva, která spočítá transformační matici, kterou aplikuje na vstupní data. Tato vrstva se skládá z vrstvy *dense*, která inicializuje váhy na nuly a bias na jednotkovou matici, čímž se zajistí, že původní vstupní data zůstanou nezměněna, a vrstvy *Reshape*, která změní tvar výstupu z vrstvy *dense* na matici. Nakonec se použije funkce *Dot()*, která provede maticové násobení mezi vstupními daty a transformační maticí. Tento výstup pak reprezentuje transformovaná vstupní data, na kterých se mohou provést další výpočty.

Po transformaci dat, se na data aplikuje vícevrstvý perceptron o dvou konvoluční vrstvě, každá disponující 32 neurony, kde je každý neuron v jedné vrstvě, spojen s každým neuronem ve vrstvě následující. Tyto vrstvy transformují původní normalizovanou matici $n \times 3$ na matici $n \times 32$. Tato matice pak reprezentuje mapu vlastností, ve formě příznakových vektorů pro jednotlivé body nezávisle na ostatních bodech normalizovaného point cloudu.

Na příznakovou matici $n \times 32$ s příznakovými vektory, se za pomoci T-Netu opět provede normalizace matice, tentokrát však pro samotné příznakové vektory a již na normalizovanou příznakovou matici, se opět aplikují vícevrstvé perceptron. Tentokrát však

o 3 vrstvách o 32, 64 a na konec 512 neuronech. Výsledkem je tedy opět příznaková matice o rozměru $n \times 512$.

Následně se nad příznakovou maticí $n \times 512$, provede metoda *GlobalMaxPooling1D()*, která ze vstupní příznakové matice vždy vybere nejvyšší hodnotu pro daný řád a výstupem této vrstvy, je jednorozměrná matice o 512 příznacích, která reprezentuje globální vlastnosti matice bodů. Tímto způsobem, se vyhodnocování point cloudu v neuronové síti stává permutačně invariantní vůči pořadí, v jakém jsou body z point cloudu vkládány na vstup.

Posledním krokem této neuronové sítě, je vyhodnocení globálních příznaků, což se provádí opět za pomoci vícevrstvých perceptronů, které jsou ve třech vrstvách o 256, 128 a následně k neuronech, kde k reprezentuje počet tříd, které se budou v neuronové síti vyhodnocovat. Současně, je za každou vrstvou perceptronů volaná funkce *Dropout()*, která náhodně nastavuje na nulu některé prvky vstupního tensoru s pravděpodobností zadanou jako argument funkce, v tomto případě 0.3, což odpovídá 30% pravděpodobnosti. Tím dochází k náhodnému vypuštění určitých prvků vstupu v každé iteraci trénování modelu. Náhodné vypuštění prvků pomáhá zabránit přeučení a učení na vlastnostech, které by mohli být specifické pouze pro trénovací sadu. Pro vyhodnocení vstupních point cloudů, se používá pro aktivační funkce výstupní vrstvy o k neuronech, funkce *softmax*.

Celá, výše popsaná neuronová síť, se vytváří následujícím kódem a schéma bude uvedené v příloze.

```

inputs = keras.Input(shape=(NUM_POINTS, 3))
x = tnet(inputs, 3)
x = conv_bn(x, 32)
x = conv_bn(x, 32)
x = tnet(x, 32)
x = conv_bn(x, 32)
x = conv_bn(x, 64)
x = conv_bn(x, 512)
x = layers.GlobalMaxPooling1D()(x)
x = dense_bn(x, 256)
x = layers.Dropout(0.3)(x)
x = dense_bn(x, 128)
x = layers.Dropout(0.3)(x)
outputs = layers.Dense(NUM_CLASSES, activation="softmax")(x)

```

Další část neuronové sítě, která je označena v schématu neuronové sítě žlutým rámečkem na obrázku č. 23, se využívá k sémantické a dílové segmentaci, kterou však v této práci nevyužívám, proto nebyla tato část vytvořena.

Neuronová síť je následně připravena ke spuštění na předpřipravených datech a jsou definovány parametry pro zahájení učení a vyhodnocení. Pro neuronovou síť byla nastavena funkce loss na *sparse_categorical_crossentropy*, která bude hodnotit rozdíly mezi vyhodnocenými kategoriemi. Tato funkce se převážně používá, právě u případů, kdy je jednotlivým vstupům, přiřazena výstupní hodnota ve formě celého kladného čísla. Dále se volí pro neuronovou síť učící algoritmus, který byl zvolen algoritmus Adam. Tento učící algoritmus, vypočítává individuální adaptivní rychlost učení pro různé parametry z odhadů prvního a druhého momentu gradientů a vychází z algoritmů AdaGrad (Adaptivní gradientový algoritmus) a RMSProp (Root Mean Square Propagation). Jako finální parametr se volila metrika na přesnost kategorie, která se zaměřuje na to, jak často se předpovědi rovnají kategoriím. Po nastavení těchto parametrů, je neuronová síť připravena na učení, které se spouští metodou *fit()* ve které byli specifikovány trénovací data, validační data pro hodnocení modelu a počet epoch učení.

```

model.compile(
    loss="sparse_categorical_crossentropy",
    optimizer=keras.optimizers.Adam(learning_rate=0.0001),
    metrics=["sparse_categorical_accuracy"],
)

model.fit(train_dataset, epochs=10, validation_data=test_dataset)

```

Při učení neuronové sítě, bylo vždy použito 10 epoch se zásobníkem pro jednotlivé cykly v epoše na 32 vzorků a vyhodnocovali se vždy 2 třídy. Nicméně vstupní počet bodů ve vstupním point cloudu, se pro každé měření měnilo, za účelem, do jaké míry je tento model neuronové sítě schopen rozpoznat člověka od jiných předmětů. Učení bylo prováděné s učitelem, vždy reprezentující kategorii, do které point cloud spadal.

Při učení neuronové sítě pro vstupní point cloud o 150 bodech, byli zaznamenány následující hodnoty.

Epoch	Loss	Accuracy	Validation Accuracy
1	1.4606	0.9693	0.9987
2	1.1423	0.9887	0.9997
3	1.1495	0.9836	1
4	1.1016	0.993	0.993
5	1.0963	0.9947	0.9993
6	1.1018	0.9899	1
7	1.0975	0.9919	0.995
8	1.0845	0.9961	0.9678
9	1.0748	0.9982	1
10	1.0915	0.9932	0.998

Tabulka 8 Výsledky neuronové sítě při 150 bodech (vlastní)

Podle konstantního nárůstu přesnosti a snižování ztráty, lze usoudit, že při učení neuronové sítě nedošlo k přeučení. Rovněž výsledky přesnosti a finální predikce, která je zaznamenána na obrázku níže, naznačují velmi dobré schopnosti neuronové sítě rozeznat jednotlivé kategorie.

pred: Human_TD, label: Human_TD pred: Human_TD, label: Human_TD



Obrázek 24 Predikce neuronové sítě pro 150 bodů (vlastní)

Při učení neuronové sítě pro vstupní point cloud o 100 bodech, byli zaznamenány následující hodnoty.

Epoch	Loss	Accuracy	Validation Accuracy
1	1.5109	0.9671	0.9904
2	1.1628	0.9923	1
3	1.1192	0.9954	0.9997
4	1.1152	0.9941	0.9917
5	1.1098	0.9936	0.9997
6	1.1006	0.9941	1
7	1.0908	0.9965	0.9977
8	1.0828	0.997	1
9	1.0758	0.9989	1
10	1.0767	0.997	1

Tabulka 9 Výsledky neuronové sítě při 100 bodech (vlastní)

Podobně jako u učení neuronové sítě na 150 bodech, naměřená data nenapovídají přeučení a rovněž výsledky přesnosti a predikce naznačují velmi dobré výsledky při učení na 100 bodech.

pred: Human_TD, label: Human_TD pred: Human_TD, label: Human_TD



Obrázek 25 Predikce neuronové sítě pro 100 bodů (vlastní)

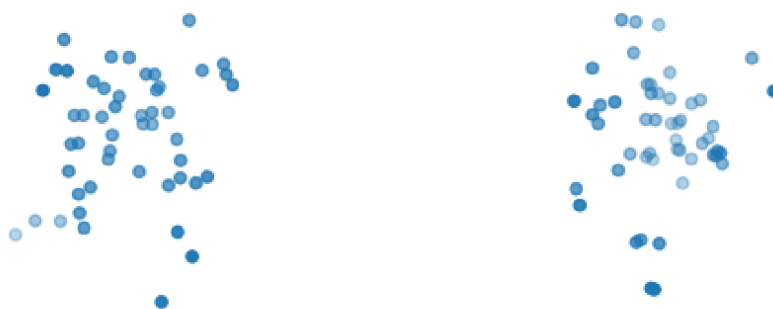
Rovněž bylo provedené učení na 50 a 25 bodech a výsledky, které byli zaznamenány jsou uvedeny níže.

Epoch	Výsledky pro 50 bodů			Výsledky pro 25 bodů		
	Loss	Accuracy	Validation Accuracy	Loss	Accuracy	Validation Accuracy
1	1.4753	0.9733	0.996	1.5202	0.9715	0.998
2	1.1525	0.9902	0.9997	1.1616	0.9895	1
3	1.1165	0.9949	1	1.1214	0.9906	1
4	1.0968	0.9947	1	1.0995	0.9938	0.9987
5	1.0917	0.9949	0.9997	1.1059	0.9908	0.9967
6	1.0932	0.994	1	1.1006	0.9925	1
7	1.077	0.998	1	1.0938	0.9935	0.9963
8	1.0695	0.9993	0.9787	1.092	0.9944	0.994
9	1.0657	0.9993	1	1.0885	0.9956	0.9997
10	1.0705	0.9978	1	1.0897	0.993	0.989

Tabulka 10 Výsledky učení neuronové sítě pro 50 a 25 bodů (vlastní)

Opět, jako u předchozích učení, i zde nedošlo ke zřejmému přeučení, ale při porovnání s předešlými výsledky, je zde zřejmé zhoršení výsledků u učení s 25 body. I přes tento fakt, však tato neuronová síť vykazuje skvělé výsledky, jak je vidět na predikcích obou sítí.

pred: Human_TD, label: Human_TD pred: Human_TD, label: Human_TD



Obrázek 26 Predikce neuronové sítě pro 50 bodů (vlastní)

pred: Human_TD, label: Human_TD pred: Human_TD, label: Human_TD



Obrázek 27 Predikce neuronové sítě pro 25 bodů (vlastní)

Jako poslední měření, bylo provedeno učení neuronové sítě na 10 bodech. Výsledky jsou zaznamenaný níže.

Epoch	Loss	Accuracy	Validation Accuracy
1	1.4323	0.9746	1
2	1.1348	0.9911	1
3	1.1031	0.9956	1
4	1.0883	0.9967	1
5	1.1033	0.9932	0.9977
6	1.0952	0.9933	0.9857
7	1.0886	0.9948	0.998
8	1.0802	0.9969	0.9944
9	1.0747	0.9982	1
10	1.0811	0.9963	1

Tabulka 11 Výsledky neuronové sítě při 10 bodech (vlastní)

Výsledky mohou naznačovat, že neuronová síť si dokáže poradit s identifikací při velmi nízkém počtu bodů, avšak po zobrazení predikcí, je zřejmé, že se neuronová síť naučila pouze na dílčí části člověka, jak je zřejmé z obrázku níže.

pred: Human_TD, label: Human_TD pred: Human_TD, label: Human_TD



Obrázek 28 Predikce neuronové sítě pro 10 bodů (vlastní)

8 Závěr

Cílem práce bylo ověřit schopnost neuronové sítě vyhodnocovat vstupní data ze zabezpečovacího systému využívající lidary.

Po přípravě dat a samotném učení neuronové sítě, se prokázalo, že síť navržená na základě modelu PointNet, je schopná identifikovat člověka od ostatních objektů s přesností přes 99 %. Učení na rozdílném množství vstupních bodů, rovněž prokázalo, že neuronová síť, je schopná správné identifikace i při poskytnutí nízkého počtu bodů a to, až do 25 bodového point cloudu. Podle výsledků, lze rovněž usoudit, že trénování na tomto počtu trénovacích dat, je dostačující na 9 epoch, jelikož při všech zaznamenaných učení neuronové sítě, se výsledky přesnosti a loss funkce znatelně zhoršily v 10 epoše. Což bylo nejspíše způsobené začínajícím přeučováním neuronové sítě. Proto by se při aplikaci konkrétně těchto modelů vždy využilo 9. epochy, která disponovala vždy nejlepšími výsledky.

Tyto velmi kladné výsledky rovněž prokázaly, že toto schéma neuronové sítě, je možné uplatnit v zabezpečovacích systémech, které využívají lidary. Konkrétně by se jednal o bezpečnostní systém Accur8vision, který byl v průběhu práce zmíněn a současně sloužil jako hlavní zdroj pro trénovací data. V budoucnu tedy bude snaha, implementovat tuto strukturu neuronové sítě do zabezpečovacího systému Accur8vision, kde by na základě výsledků zaznamenaných v této práci, mohla odfiltrovat velké množství falešných poplachů, které by byly způsobené nevhodným počasím, nebo vlivem špatného snapshotu, a tím pádem zlepšit kvalitu samotného bezpečnostního systému.

V rámci navázání na tuto práci se může rovněž stejným způsobem vyhodnotit přesnost jiných schémat neuronových sítí, jako je například Point Cloud Transformer a porovnat jejich výsledky s touto prací.

9 Zdroje

- A. WASSER, Leah, 2022. The Basics of LiDAR: Light Detection and Ranging - Remote Sensing. In: *Neonscience* [online]. Neonscience [cit. 2023-03-18]. Dostupné z: <https://www.neonscience.org/resources/learning-hub/tutorials/lidar-basics>
- About Ouster: Ouster. High-performance digital lidar* [online], 2022. <https://ouster.com/company/>: Ouster, Inc. [cit. 2023-03-18]. Dostupné z: <https://ouster.com/company/>
- Accur8vision - Handbook*, 2022. 1. Praha.
- Accur8vision for intelligent security systems, 2022. In: *Hexagon: 3D Surveillance* [online]. Hexagon [cit. 2023-03-18]. Dostupné z: <https://3dsurveillance.hexagon.com/products/accur8vision>
- Accur8vision: Zone types in Accur8vision, 2022. In: *Youtube* [online]. Youtube Video: Youtube [cit. 2023-03-18]. Dostupné z: https://www.youtube.com/watch?v=GttXupSfRKQ&ab_channel=accur8vision
- Blickfeld: Lidar/scan your world* [online], 2022. [blickfeld.com](https://www.blickfeld.com/): Blickfeld [cit. 2023-03-18]. Dostupné z: <https://www.blickfeld.com/>
- BLK247: Smart 3D Surveillance System, 2022. In: *Shop.leica-geosystems.com/* [online]. Leica Geosystems AG • Part of Hexagon [cit. 2023-03-18]. Dostupné z: <https://shop.leica-geosystems.com/leica-blk/blk247>
- CONTRIBUTOR, TechTarget, 2023. Soft Computing. In: *Techtarget* [online]. Techtarget [cit. 2023-03-18]. Dostupné z: <https://www.techtarget.com/whatis/definition/soft-computing>
- Detektory Pyronix: ABBAS - Elektronické zabezpečení majetku, kamerové systémy, požární signalizace* [online], 2015. [abbas.cz](https://www.abbas.cz) [cit. 2023-03-18]. Dostupné z: <https://www.abbas.cz/clanky/recenze-technika/detektory-pyronix-osvedcena-kvalita-detekce/>
- DURČÁK, Pavel, 2017. Neuronové sítě a princip jejich fungování. In: *NaPočítači* [online]. NaPočítači [cit. 2023-03-18]. Dostupné z: <https://www.napocitaci.cz/33/neuronove->

site-a-princip-jejich-fungovani-
uniqueidgOkE4NvrWuNY54vrLeM670eFNQh552VdDDulZX7UDBY/

HADJI, Bahman, 2021. Understanding wavelength choice in LiDAR systems. In: *Embedded* [online]. Embedded [cit. 2023-03-18]. Dostupné z: <https://www.embedded.com/understanding-wavelength-choice-in-lidar-systems/>

Hexagon: 3D Surveillance, 2023. In: *Hexagon: 3D Surveillance* [online]. Hexagon [cit. 2023-03-18]. Dostupné z: <https://3dsurveillance.hexagon.com/>

Innovusion: Falcon [online], 2022. Innovusion [cit. 2023-03-18]. Dostupné z: <https://innovusion.com/products/falcon/>

Intrepid: DETEKČNÍ SYSTÉM SE ZEMNÍMI ŠTĚRBINOVÝMI KABELY, 2012. In: *Adiglobal.cz* [online]. adiglobal.cz [cit. 2023-03-18]. Dostupné z: [https://adiglobal.cz/iiWWW/docs.nsf/all/2B38E66070A3F10CC1257A78002F3F7D/\\$FILE/KL_INTREPID%20MicroTrack%20II.pdf](https://adiglobal.cz/iiWWW/docs.nsf/all/2B38E66070A3F10CC1257A78002F3F7D/$FILE/KL_INTREPID%20MicroTrack%20II.pdf)

JABLOSHOP.cz: JA-150P Bezdrátový PIR detektor pohybu k CA2103 - Jablotron [online], 2022. JABLOSHOP.cz [cit. 2023-03-18]. Dostupné z: https://www.jabloshop.cz/ja-150p-bezdratovy-pir-detektor-pohybu-k-ca-2103?gclid=Cj0KCQiA8t2eBhDeARIsAAVEga3xhXTL0MkcMJs_JGLJ68gicyZ0XH Hhgs4AMbswMh0pJlbRguDWffYaAl84EALw_wcB#3937

KŘEČEK, Stanislav, 2006. *Příručka zabezpečovací techniky*. 3. Praha: Cricetus. ISBN 978-80-87603-13-0.

LiDAR Sensors, Simplified: Part 1: LiDAR System Point Density, Returns and Beam Divergence, 2022. In: *Geodetics* [online]. Geodetics [cit. 2023-03-18]. Dostupné z: <https://geodetics.com/lidar-sensors-simplified/>

LOUTHÁNOVÁ, Pavla, 2021. Umělá inteligence. In: *Bud' FIT: Časopis FIT ČVUT* [online]. FIT ČVUT [cit. 2023-03-18]. Dostupné z: <https://casopis.fit.cvut.cz/tema/umela-inteligence/umela-inteligence/>

Multiple return LiDAR, 2022. In: *IIT Kanpur* [online]. IIT Kanpur [cit. 2023-03-18]. Dostupné z: http://home.iitk.ac.in/~blohani/LiDAR_Tutorial/Multiple%20return%20LiDAR.htm

- Optical encoders and LiDAR scanning, 2022. In: *Renishaw* [online]. Renishaw [cit. 2023-03-18]. Dostupné z: <https://www.renishaw.com/en/optical-encoders-and-lidar-scanning--39244>
- Ouster: Gemini, 2023. In: *Ouster* [online]. www.ouster.com: Ouster [cit. 2023-03-18]. Dostupné z: <https://ouster.com/software/gemini/>
- Perimetrická ochrana: Z čeho se skládá a kdy je vhodná?, 2020. In: *Securitas* [online]. Securitas [cit. 2023-03-18]. Dostupné z: <https://www.securitas.cz/novinky--blog/blog/perimetricka-ochrana-z-ceho-se-sklada-a-kdy-je-vhodna/>
- Qi, Charles R. et al., b.r. *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*. 3. Dostupné z: doi:<https://doi.org/10.48550/arXiv.1612.00593>
- Quanergy Systems: Quanergy QORTEX Intro, 2020. In: *Youtube* [online]. Youtube video: Youtube [cit. 2023-03-18]. Dostupné z: https://www.youtube.com/watch?v=nJMbxCcMLpI&ab_channel=QuanergySystems
- Quanergy: Security, 2023. In: *Quanergy: Security* [online]. Quanergy: Quanergy [cit. 2023-03-18]. Dostupné z: <https://quanergy.com/applications/security/>
- The Different Types Of LiDAR Systems, 2022. In: *Lidar and Radar Information* [online]. Lidar and Radar Information [cit. 2023-03-18]. Dostupné z: <https://lidarradar.com/info/the-different-types-of-lidar-systems>
- Umělá inteligence pro každého: Co je umělá inteligence, 2023. In: *Machine Learning College* [online]. Machine Learning College: Machine Learning College [cit. 2023-03-18]. Dostupné z: <https://www.mlcollege.com/co-je-umela-inteligence/>
- Umělá inteligence: definice a využití: Co je umělá inteligence a jak ji využíváme?, 2020. In: *Evropský parlament: Zpravodajství* [online]. europarl.europa.eu: Evropský parlament [cit. 2023-03-18]. Dostupné z: <https://www.europarl.europa.eu/news/cs/headlines/society/20200827STO85804/umela-inteligence-definice-a-vyuziti>
- Umělá inteligence, 2023. In: *Wikisofia* [online]. Wikisofia: Wikisofia [cit. 2023-03-18]. Dostupné z: https://wikisofia.cz/wiki/Um%C4%9B1%C3%A1_inteligence

- VALKOVIČ, Patrik, 2023. Neuronové síť: Perceptron. In: *IT Network* [online]. IT Network [cit. 2023-03-18]. Dostupné z: <https://www.itnetwork.cz/python/neuronove-site/pokrocile/neuronove-site-perceptron/>
- Velodyne lidar: Envision the Future* [online], 2022. Velodyne Lidar, Inc. 2023 [cit. 2023-03-18]. Dostupné z: <https://velodynelidar.com/>
- Velodyne's Guide to Lidar Wavelengths: A Guide to Lidar Wavelengths for Autonomous Vehicles and Driver Assistance, 2021. In: *Velodyne Lidar* [online]. Velodyne [cit. 2023-03-18]. Dostupné z: <https://velodynelidar.com/blog/guide-to-lidar-wavelengths/>
- WANG, Chisheng et al., 2014. A comparison of waveform processing algorithms for single-wavelength LiDAR bathymetry. *ScienceDirect* [online]. **2014**, 26-28 [cit. 2023-03-18]. Dostupné z: [doi:https://doi.org/10.1016/j.isprsjprs.2014.11.005](https://doi.org/10.1016/j.isprsjprs.2014.11.005)
- What is LiDAR and How Does it Work?: What is LiDAR, 2022. In: *Synopsys* [online]. Synopsys [cit. 2023-03-18]. Dostupné z: <https://www.synopsys.com/glossary/what-is-lidar.html#A>

Příloha A: Schéma neuronové sítě

Příloha Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, n, 3)]	0	[]
conv1d (Conv1D)	(None, n, 32)	128	['input_1[0][0]']
batch_normalization (BatchNormalization)	(None, n, 32)	128	['conv1d[0][0]']
activation (Activation)	(None, n, 32)	0	['batch_normalization[0][0]']
conv1d_1 (Conv1D)	(None, n, 64)	2112	['activation[0][0]']
batch_normalization_1 (BatchNormalization)	(None, n, 64)	256	['conv1d_1[0][0]']
activation_1 (Activation)	(None, n, 64)	0	['batch_normalization_1[0][0]']
conv1d_2 (Conv1D)	(None, n, 512)	33280	['activation_1[0][0]']
batch_normalization_2 (BatchNormalization)	(None, n, 512)	2048	['conv1d_2[0][0]']
activation_2 (Activation)	(None, n, 512)	0	['batch_normalization_2[0][0]']
global_max_pooling1d (GlobalMaxPooling1D)	(None, 512)	0	['activation_2[0][0]']
dense (Dense)	(None, 256)	131328	['global_max_pooling1d[0][0]']
batch_normalization_3 (BatchNormalization)	(None, 256)	1024	['dense[0][0]']
activation_3 (Activation)	(None, 256)	0	['batch_normalization_3[0][0]']
dense_1 (Dense)	(None, 128)	32896	['activation_3[0][0]']
batch_normalization_4 (BatchNormalization)	(None, 128)	512	['dense_1[0][0]']
activation_4 (Activation)	(None, 128)	0	['batch_normalization_4[0][0]']
dense_2 (Dense)	(None, 9)	1161	['activation_4[0][0]']
reshape (Reshape)	(None, 3, 3)	0	['dense_2[0][0]']
dot (Dot)	(None, n, 3)	0	['input_1[0][0]', 'reshape[0][0]']
conv1d_3 (Conv1D)	(None, n, 32)	128	['dot[0][0]']
batch_normalization_5 (BatchNormalization)	(None, n, 32)	128	['conv1d_3[0][0]']
activation_5 (Activation)	(None, n, 32)	0	['batch_normalization_5[0][0]']
conv1d_4 (Conv1D)	(None, n, 32)	1056	['activation_5[0][0]']
batch_normalization_6 (BatchNormalization)	(None, n, 32)	128	['conv1d_4[0][0]']
activation_6 (Activation)	(None, n, 32)	0	['batch_normalization_6[0][0]']
conv1d_5 (Conv1D)	(None, n, 32)	1056	['activation_6[0][0]']
batch_normalization_7 (BatchNormalization)	(None, n, 32)	128	['conv1d_5[0][0]']
activation_7 (Activation)	(None, n, 32)	0	['batch_normalization_7[0][0]']

conv1d_6 (Conv1D)	(None, n, 64)	2112	['activation_7[0][0]']
batch_normalization_8 (BatchNormalization)	(None, n, 64)	256	['conv1d_6[0][0]']
activation_8 (Activation)	(None, n, 64)	0	['batch_normalization_8[0][0]']
conv1d_7 (Conv1D)	(None, n, 512)	33280	['activation_8[0][0]']
batch_normalization_9 (BatchNormalization)	(None, n, 512)	2048	['conv1d_7[0][0]']
activation_9 (Activation)	(None, n, 512)	0	['batch_normalization_9[0][0]']
global_max_pooling1d_1 (GlobalMaxPooling1D)	(None, 512)	0	['activation_9[0][0]']
dense_3 (Dense)	(None, 256)	131328	['global_max_pooling1d_1[0][0]']
batch_normalization_10 (BatchNormalization)	(None, 256)	1024	['dense_3[0][0]']
activation_10 (Activation)	(None, 256)	0	['batch_normalization_10[0][0]']
dense_4 (Dense)	(None, 128)	32896	['activation_10[0][0]']
batch_normalization_11 (BatchNormalization)	(None, 128)	512	['dense_4[0][0]']
activation_11 (Activation)	(None, 128)	0	['batch_normalization_11[0][0]']
dense_5 (Dense)	(None, 1024)	132096	['activation_11[0][0]']
reshape_1 (Reshape)	(None, 32, 32)	0	['dense_5[0][0]']
dot_1 (Dot)	(None, n, 32)	0	['activation_6[0][0]','reshape_1[0][0]']
conv1d_8 (Conv1D)	(None, n, 32)	1056	['dot_1[0][0]']
batch_normalization_12 (BatchNormalization)	(None, n, 32)	128	['conv1d_8[0][0]']
activation_12 (Activation)	(None, n, 32)	0	['batch_normalization_12[0][0]']
conv1d_9 (Conv1D)	(None, n, 64)	2112	['activation_12[0][0]']
batch_normalization_13 (BatchNormalization)	(None, n, 64)	256	['conv1d_9[0][0]']
activation_13 (Activation)	(None, n, 64)	0	['batch_normalization_13[0][0]']
conv1d_10 (Conv1D)	(None, n, 512)	33280	['activation_13[0][0]']
batch_normalization_14 (BatchNormalization)	(None, n, 512)	2048	['conv1d_10[0][0]']
activation_14 (Activation)	(None, n, 512)	0	['batch_normalization_14[0][0]']
global_max_pooling1d_2 (GlobalMaxPooling1D)	(None, 512)	0	['activation_14[0][0]']
dense_6 (Dense)	(None, 256)	131328	['global_max_pooling1d_2[0][0]']
batch_normalization_15 (BatchNormalization)	(None, 256)	1024	['dense_6[0][0]']
activation_15 (Activation)	(None, 256)	0	['batch_normalization_15[0][0]']

dropout (Dropout)	(None, 256)	0	['activation_15[0][0]']
dense_7 (Dense)	(None, 128)	32896	['dropout[0][0]']
batch_normalization_16 (BatchNormalization)	(None, 128)	512	['dense_7[0][0]']
activation_16 (Activation)	(None, 128)	0	['batch_normalization_16[0][0]']
dropout_1 (Dropout)	(None, 128)	0	['activation_16[0][0]']
dense_8 (Dense)	(None, 2)	258	['dropout_1[0][0]']