



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

# LASEROVÝ SKENER PRO ROBOTICKÉ RAMENO UR5E

LASER SCANNER FOR UR5E ROBOTIC ARM

## BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

## AUTOR PRÁCE

AUTHOR

David Bystroň

## VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Najman, Ph.D.

BRNO 2023

## Zadání bakalářské práce

Ústav:	Ústav mechaniky těles, mechatroniky a biomechaniky
Student:	<b>David Bystroň</b>
Studijní program:	Mechatronika
Studijní obor:	bez specializace
Vedoucí práce:	<b>Ing. Jan Najman, Ph.D.</b>
Akademický rok:	2022/23

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijníma zkušebníma řádem VUT v Brně určuje následující téma bakalářské práce:

### Laserový skener pro robotické rameno UR5e

#### Stručná charakteristika problematiky úkolu:

Navzdory relativní bezpečnosti kolaborativních robotů je stále nutné i u těchto strojů dbát na možná rizika střetu s operátorem či náhodným kolemjdoucím. Jedním z nejrozšířenějších způsobů jak tento problém řešit, je vytvoření bezpečnostních zón v okolí robota.

Cílem této práce bude využití laserového skeneru (LIDAR) pro detekci pohybu v okolí robotického ramene UR5e včetně možnosti konfigurace bezpečnostních zón a propojení s robotem. Pro zpracování dat z laserového skeneru se předpokládá použití samostatné řídicí jednotky s displejem (např. Raspberry Pi).

#### Cíle bakalářské práce:

1. Proveďte rešerši v oblasti řešení bezpečnosti kolaborativních robotů se zaměřením na bezpečnostní zóny a využití laserových snímačů.
2. Na základě rešerše navrhnete vhodné řešení systému bezpečnostních zón s využitím laserového skeneru.
3. U vybraného laserového skeneru zprovozníte komunikaci s řídicí jednotkou, zmapujte možnosti jeho nastavení a vykreslování dat na displej jednotky.
4. Naprogramujte řešení pro robustní detekci osob v bezpečnostních zónách, propojení s robotem a možnost úpravy nastavení zón a parametrů systému (GUI).

#### Seznam doporučené literatury:

KOLÍBAL, Zdeněk. Roboty a robotizované výrobní technologie. Brno: Vysoké učení technické v Brně - nakladatelství VUTIUM, 2016. ISBN 978-80-214-4828-5.

BARATTINI, Paolo, Vicentini FEDERICO, Gurvinder Singh VIRK a Tamás HAIDEGGER. Human–Robot Interaction. 1. Milton: CRC Press, 2019. ISBN 9781138626751.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2022/23

V Brně, dne

L. S.

---

prof. Ing. Jindřich Petruška, CSc.  
ředitel ústavu

---

doc. Ing. Jiří Hlinka, Ph.D.  
děkan fakulty

## **ABSTRAKT**

Tato práce se zabývá bezpečností kolaborativních robotů se zaměřením na bezpečnostní zóny vytvořené pomocí laserového skeneru. Cílem je tedy vytvoření zón kolem robotického ramene UR5e pomocí RPLIDAR A3M1. Součástí práce je rešerše v oblasti bezpečnosti kobotů a možnosti zpracování dat z RPLIDARu. Následně je vyvinut algoritmus pro detekci osob v okolí robota a GUI pro možnost nastavování parametrů.

## **ABSTRACT**

This thesis deals with the safety of collaborative robots, focusing on safety zones created by a laser scanner. The goal is to create zones around the UR5e robotic arm using RPLIDAR A3M1. The work includes a research in the field of cobot safety and the possibility of processing RPLIDAR data. Subsequently, an algorithm for detecting people around the robot is developed and a GUI for the possibility of setting parameters.

## **KLÍČOVÁ SLOVA**

Laserové skenery, LIDAR, bezpečnostní zóny, kolaborativní roboti

## **KEYWORDS**

Laser scanners, LIDAR, safety zones, collaborative robots

## **BIBLIOGRAFICKÁ CITACE**

BYSTRONĚ, David. *Laserový skener pro robotické rameno UR5e*. Brno, 2023. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/149715>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav mechaniky těles, mechatroniky a biomechaniky. Vedoucí práce Jan Najman.

## **ČESTNÉ PROHLÁŠENÍ**

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. Jana Najmana, Ph.D. a s použitím literatury uvedené v seznamu literatury.

V Brně

.....

David Bystroň

## **PODĚKOVÁNÍ**

Tímto bych chtěl poděkovat panu Ing. Janu Najmanovi, Ph.D. za cenné rady a správné nasměrování při řešení práce.

# OBSAH

<b>1</b>	<b>ÚVOD</b> .....	<b>9</b>
<b>2</b>	<b>REŠŘŠE</b> .....	<b>10</b>
2.1	Bezpečnost kolaborativních robotů .....	10
2.1.1	Bezpečnostní normy a prvky kolaborativních robotů .....	10
2.1.2	Bezpečnostní zóny kolaborativních robotů .....	11
2.2	UR5e .....	13
2.2.1	Technické parametry.....	13
2.2.2	Pracovní prostor.....	13
2.2.3	Elektrické rozhraní.....	14
2.3	RPLIDAR A3M1 .....	15
2.3.1	Technická specifikace .....	16
2.3.2	Možnosti nastavení .....	16
2.3.3	Možnosti komunikace .....	17
2.3.4	Data z LIDARu.....	17
2.3.5	Knihovna pro zpracování dat .....	21
2.4	Detekce osob pomocí LIDARu .....	21
2.4.1	Clustering .....	21
2.4.2	DBSCAN.....	22
2.4.3	Knihovna pro clustering .....	23
<b>3</b>	<b>POSTUP A VÝSLEDKY ŘEŠENÍ</b> .....	<b>24</b>
3.1	Postup řešení na základě rešerše.....	24
3.2	Konstrukce.....	25
3.2.1	Kryt Raspberry Pi a displeje.....	25
3.2.2	Přípevnění setu .....	25
3.2.3	Přípevnění RPLIDARu A3M1 .....	26
3.3	Komunikační rozhraní.....	27
3.3.1	Komunikace RPi a RPLIDARu.....	28
3.3.2	Komunikace RPi a UR5e .....	28
3.4	Detekce pomocí porovnání dat vzdáleností .....	30
3.4.1	Získání a zpracování dat.....	30
3.4.2	Detekce.....	32
3.5	Detekce pomocí clusteringu .....	33
3.5.1	Porovnání cluster metod.....	33
3.5.2	Zpracování dat .....	34
3.5.3	Detekce.....	35
3.6	GUI .....	35
3.6.1	Stavy a jejich funkce .....	35
<b>4</b>	<b>ZÁVĚR</b> .....	<b>40</b>
<b>5</b>	<b>BIBLIOGRAFIE</b> .....	<b>41</b>
<b>6</b>	<b>SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK</b> .....	<b>44</b>
<b>7</b>	<b>SEZNAM PŘÍLOH</b> .....	<b>46</b>



# 1 ÚVOD

V dnešní době, kdy se stále více modernizuje průmyslové prostředí, můžeme častěji vidět využití kolaborativních robotů. Ačkoliv jsou kolaborativní roboti díky své konstrukci a pracovní rychlosti relativně bezpeční. Pokud se má zvýšit jejich efektivita, je třeba aby pracovali co největší dobu na vysoký výkon, to ovšem zvyšuje riziko srážky s obsluhou. Pokud tedy chceme, aby roboti pracovali efektivně a zároveň sdíleli stejné pracovní prostředí s člověkem, je třeba monitorovat okolí kolem robota.

Monitorování prostoru je v celku obvyklou záležitostí, ať už pomocí kamer nebo různých senzorů. V posledních letech můžeme v řadě aplikacích pozorovat využití LIDARu, to také vede k větší cenové dostupnosti LIDARů menších rozměrů. V této práci bude proto využit finančně dostupný 2D RPLIDAR A3M1, který díky vysoké rychlosti a přesnosti snímání dokáže generovat mračna dat a vytvořit tak mapu okolí.

Cílem této práce bude tedy vytvoření algoritmu, který s pomocí RPLIDARu a řídicí jednotky vytvoří kolem robotického ramene UR5e bezpečnostní zóny a bude schopen detekovat osoby v těchto zónách. Hlavní zaměření je tedy na samotný RPLIDAR a naprogramování algoritmu.

V rešeršní části jsou popsány principy bezpečnosti u kolaborativních robotů a popis UR5e pro které je algoritmus děláný. Velká část se pak věnuje samotnému RPLIDARu a možnosti detekce osob pomocí něj.

V praktické části jsou popsány jednotlivé kroky při vytváření algoritmu a GUI, díky kterému bude moct uživatel měnit parametry zón a systému. Součástí práce je také konstrukce pro uchycení RPLIDARu a ostatních komponent potřebné pro celkovou funkčnost.

## 2 REŠŘŠE

### 2.1 Bezpečnost kolaborativních robotů

ISO definuje kolaborativního robota jako „*robota navrženého pro přímou interakci s člověkem*“. Dle ISO/TS 1566: *Robots and robotic devices – collaborative robots*, jde o průmyslového kolaborativního robota, který pracuje ve vymezeném kolaborativním pracovním prostoru, kde kolaborativní pracovní prostor je chráněný prostor, kde robot a člověk mohou pracovat současně během výroby. [1]

Lze říci, že přímá spolupráce člověka a robota, věc kdysi těžko představitelná, je nyní v průmyslu obrovským trendem. I z tohoto důvodu existuje v této oblasti mnoho norem, které je třeba dodržovat.

Ačkoliv nelze s jistotou zabránit srážce robota s pracovníkem, jeho design je vytvořen tak, aby i v těchto výjimečných případech snižoval následky zranění, převážně díky pracovní rychlosti robota, brzdovému systému, detekci kontaktu nebo samotné konstrukci. [2]

#### 2.1.1 Bezpečnostní normy a prvky kolaborativních robotů

Bezpečnostní normy můžeme rozdělit na tři typy:

Normy typu A (základní bezpečnostní normy) stanovují základní pojmy a zásady pro projektování, konstrukci a obecná hlediska, která mohou být použita u všech strojů.

Normy typu B (skupina bezpečnostních norem) se zabývají jedním bezpečnostním aspektem nebo jedním typem bezpečnostního zařízení, které může být použito pro větší počet strojů. Sem patří například norma ČSN EN ISO 13857 (83 3212) *Bezpečnost strojních zařízení – Bezpečné vzdálenosti k zamezení dosahu do nebezpečných prostor horními a dolními končetinami*.

Normy typu C (bezpečnostní normy pro stroje) určují detailní bezpečnostní požadavky pro jednotlivý stroj nebo skupinu strojů. Zde jsou zařazeny nejdůležitější mezinárodní normy z hlediska bezpečnosti robotů a pracovišť s roboty. ČSN EN ISO 10218-1 a ČSN EN ISO 10218-2. [3; 4]

V tab. 1 je souhrn bezpečnostních norem, kterými se řídí výrobce kolaborativního ramene UR5e, Universal Robots. Nejedná se však o kompletní seznam používaných norem, společnost se řídí i dalšími normami, které nemusí být harmonizované.

Tab. 1: Přehled norem uplatňovaných při posuzování bezpečnosti pracovišť s roboty [3]

ČSN EN ISO 12100 (83 3001)	Bezpečnost strojních zařízení – Všeobecné zásady pro konstrukci – Posouzení rizika a snižování rizika
ČSN EN ISO 13849-1 (83 3205)	Bezpečnost strojních zařízení – Bezpečnostní části ovládacích systémů – Část 1: Obecné zásady pro konstrukci
ČSN EN ISO 13850 (83 3311)	Bezpečnost strojních zařízení – Nouzové zastavení – Zásady pro konstrukci
ISO 14118	Safety of machinery – Prevention of unexpected start-up (neharmonizovaná; jako ČSN nepřevzata)
ČSN EN ISO 13857 (83 3212)	Bezpečnost strojních zařízení – Bezpečné vzdálenosti k zamezení dosahu do nebezpečných prostor horními a dolními končetinami
ČSN EN ISO 13855 (83 3303)	Bezpečnost strojních zařízení – Umístění ochranných zařízení s ohledem na rychlosti přiblížení částí lidského těla
ČSN EN ISO 10218-1 (18 6502)	Roboty a robotická zařízení – Požadavky na bezpečnost průmyslových robotů – Část 1: Roboty
ČSN EN ISO 10218-2 (18 6502)	Roboty a robotická zařízení – Požadavky na bezpečnost průmyslových robotů – Část 2: Systémy robotů a integrace

Obecně můžeme bezpečnostní funkce kolaborativních robotů rozdělit na čtyři typy. „*Safety-rated monitored stop*“, „*Hand guiding*“, „*Speed and separation monitoring*“ a „*Power and force limiting*“. Podle Universal Robots [5] jsou pro užití v praxi nejlepší poslední dva typy.

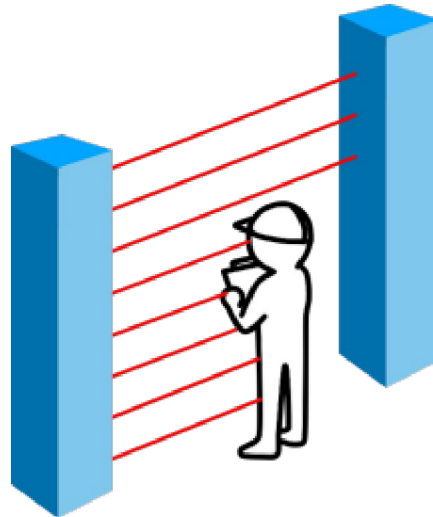
Power and force limiting je momentálně nejrozšířenější funkce a většina kobotů je tak navržena. Díky bezpečnostním funkcím pro monitorování pohybu, rychlosti, síly a výkonu mohou pracovník a robot pracovat současně na společném pracovišti. Pokud dojde ke kontaktu robota a obsluhy, systém se zastaví. Tato metoda předpokládá kontakt mezi člověkem a kolaborativním robotem, ovšem díky omezení rychlosti a výkonu nemůže dojít k vážným zraněním.

Pro použití metody Speed and separation monitoring jsou převážně používány externí zařízení, jako například laserové skenery, avšak tyto technologie nejsou zatím komerčně využívány. Metoda je výhodná, protože robot může pracovat při vyšších rychlostech a také s větším zatížením a zpomalit, případně zastavit, pouze pokud je v blízkosti člověk.

### 2.1.2 Bezpečnostní zóny kolaborativních robotů

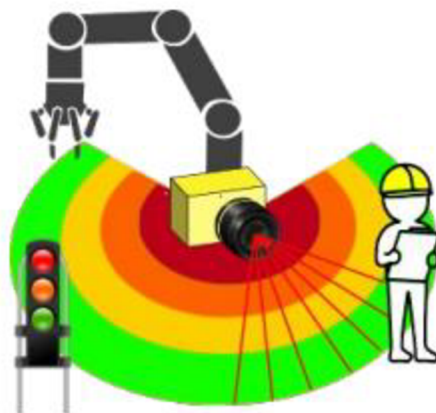
ISO/TS 15066 nám definuje kolem kolaborativního robota 5 bezpečnostních zón. Maximální zóna je prostor maximálního dosahu robota jakoukoliv jeho částí. Jen málo robotů je schopno se pohybovat v celé oblasti maximálního rozsahu, například kvůli bezpečnostním limitům. Tato zóna se pak nazývá omezená zóna. Část omezené zóny, ve které kobot reálně pracuje v rámci zadaného úkolu se pak nazývá pracovní zóna. Chráněná zóna je pak větší než maximální a jedná se o prostor mezi robotem a bezpečnostním oplocením, které nemusí být nutně fyzického charakteru, může se jednat například o obvodové zóny pomocí laserového skeneru. Poslední zóna je kolaborativní, tedy prostor, kde robot a člověk pracují současně. [6]

Pro detekci osob v chráněné zóně mohou být použity například světelné závěsy jako na obr. 1, bezpečnostní nášlapné rohože, kamery a laserové skenery, ilustrativní situace i s vyznačením zón na obr. 2. [7]



*Light curtains detect when a person or object has entered an area.*

Obr. 1: Využití světelného závěsu pro detekci osob [7]



Obr. 2: Využití laserového skeneru pro detekci osob [7]

Ve většině pracích zabývajících se detekcí osob pomocí laserových skenerů se používá technologie SLAM (simultaneous localisation and mapping) a 3D LIDAR. V [8] byl například použit Kinect. Častokrát se pak z mračen bodů filtrují osoby, měří se vzdálenosti bodů od robota, v některých případech se také predikuje pohyb člověka na základě strojového učení. Všechny tyto technologie jsou výpočetně poměrně náročné a doposud nejsou v bezpečnosti kolaborativních robotů komerčně používány. Dá se však očekávat, že po splnění certifikací se využití skenerů bude rozšiřovat.

## 2.2 UR5e

UR5e je kolaborativní robotické rameno od firmy Universal Robots, vydané v roce 2018. Ideální použití je při automatizaci procesů s nízkou hmotností.



Obr. 3: UR5e [9]

### 2.2.1 Technické parametry

Ideální použití robota je při automatizaci procesů s nízkou hmotností. Podrobné technické specifikace se nacházejí v tab. 2.

Tab. 2: Vybrané technické parametry UR5e [10]

Hmotnost včetně kabelů	20,6 kg
Dosah	850 mm
Počet stupňů volnosti	6
Maximální zatížení	5 kg
Rozsah senzoru síly	50 N
Rozsah senzoru momentu	10 Nm
Pracovní rozsah kloubů	$\pm 360^\circ$
Maximální rychlost kloubů	$\pm 180^\circ/\text{s}$

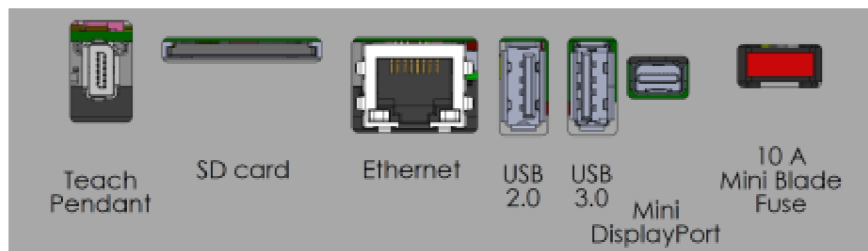
### 2.2.2 Pracovní prostor

Pracovní prostor robota dosahuje 850 mm od základny. Při výběru místa pro montáž robotického ramene je třeba věnovat pozornost válcovému prostoru bezprostředně nad a pod základnou robota. Používání nástrojů v blízkosti válcového prostoru je dobré se vyvarovat z důvodu velmi rychlého pohybu kloubů robota při pomalém pohybu nástroje. To způsobuje neefektivnost a složité posuzování rizik (viz obr. 4). [9]



Pomocí vstupních signálů pak lze použít různé funkce robota, z kterých by nás mohlo zajímat „Nouzové zastavení systému“ a především pak „Omezený režim“, ve kterém se rychlost ramene sníží dle nastavené hodnoty. Bezpečnostní systém zaručuje přechod do omezeného režimu během půl sekundy. [9]

Na spodní straně ovládací skříně se pak nachází porty pro připojení dalších zařízení. Je zde například SD karta s nahráním softwarem, USB porty, ke kterým můžeme připojit třeba klávesnici nebo ethernetový port pro připojení dalšího zařízení podporující MODBUS, Ethernet/IP a PROFINET. [11]



Obr. 6: Porty pro připojení externích zařízení k UR5e [11]

### 2.3 RPLIDAR A3M1

RPLIDAR A3M1, obr. 7, je 2D laserový skener představený v roce 2017 firmou SLAMTEC. Jedná se o nízkonákladový LIDAR, který nabízí stabilní výkon při detekci jak ve vnitřních prostorech, tak venku, s relativně dlouhým dosahem.



Obr. 7: RPLIDAR A3M1 [12]

### 2.3.1 Technická specifikace

Vybrané technické specifikace RPLIDARu A3M1 jsou shrnuty v tab. 4. Některé parametry se mohou lišit v závislosti na pracovním módu RPLIDARu, viz kapitola 2.3.2.

Tab. 4: Technické specifikace RPLIDAR A3M1 [12]

Měřicí rozsah	360 °
Vzdálenostní rozsah	25 m
Vzorkovací frekvence	16 kHz
Frekvence snímání	Typicky 10 Hz
Úhlové rozlišení	0,225 °
Baud rate	256000 bps

### 2.3.2 Možnosti nastavení

Od verze firmwaru 1.24. je zaveden nový koncept „*režim skenování*“. V jednotlivých režimech se může výkonnost RPLIDARu lišit. Různé modely RPLIDARu podporují jiné režimy skenování. Každý z režimů je optimalizovaný pro určité pracovní prostředí. Pomocí příkazu GET\_LIDAR\_CONF může uživatel zjistit jaké režimy zařízení podporuje. Režimy skenování podporované RPLIDAREm A3M1 jsou v tab. 5. [13]

Tab. 5: Režimy skenování RPLIDAR A3M1 [13]

Režim	Max. vzorkovací frekvence	Max. vzdálenost
Standard	4 kHz	16 m
Express	8 kHz	25 m
Boost	16 kHz	25 m
Sensitivity	16 kHz	25 m
Stability	16 kHz	25 m

Režim Boost je optimalizován pro vzorkovací frekvenci. Sensitivity je pak vhodný pro delší vzdálenosti s lepší citlivostí, ale slabou eliminaci okolního světla. Stability režim je pak optimální pro kratší vzdálenosti, dobře se vyrovnává se světlem v okolí.



### 2.3.3 Možnosti komunikace

Komunikace s RPLIDAREM A3M1 je možná prostřednictvím rozhraní UART, které umožňuje přenos dat mezi RPLIDAREM a hostitelským systémem. [12]

Druhou možností je pak využití USB adaptéru, který umožňuje přepínání mezi baudratem 115200 a 256000. Pro RPLIDAR A3M1 je třeba nastavit na 256000. K adaptéru se připojí napájecí a komunikační rozhraní, obr. 8, a následně se připojí pomocí Micro-USB kabelu k řídicí jednotce jako na obr. 9. Po připojení přes USB adapter je třeba naistalovat také volně dostupný ovladač. [14]



Obr. 8: Komunikační a napájecí rozhraní RPLIDAR [12]

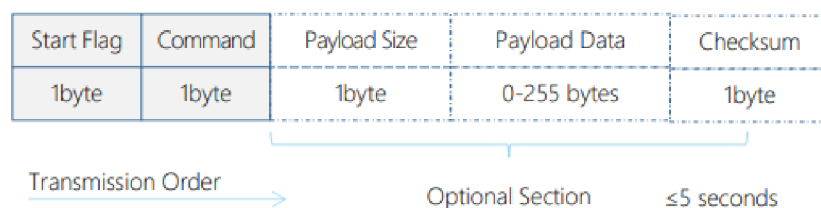


Obr. 9: Připojení RPLIDARu přes USB adaptér [14]

### 2.3.4 Data z LIDARu

RPLIDAR používá ke komunikaci protokol založený na netextových binárních data paketech. Komunikaci vždy zahajuje hostující systém, jako například mikrokontroler, nebo počítač.

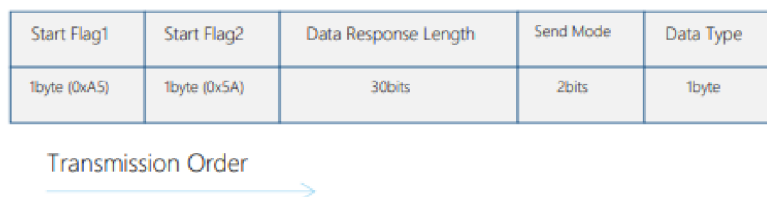
Data pakety odeslané z hostitelského systému se jsou označeny jako Request a všechny mají společný formát, obr. 10, s pořadím little – endian. [13] To znamená, že na paměťové místo s nejnižší adresou se uloží nejméně významný bajt.



Obr. 10: Request paket [13]

Bajt 0x5A je fixně daný pro Start Flag a používá se pro detekci nového Request paketu, po něm musí následovat Command. Všechny bajty v Request paketu musí být do RPLIDARU odeslány během 5 sekund, jinak bude paket vyřazen. [13]

Pakety, které pak odesílá RPLIDAR nazpět, jsou označovány jako Response a můžeme je rozdělit na 2 druhy: Response Descriptor a Data Response. RPLIDAR, pokud je potřeba, vždy odešle nejprve Data Descriptor paket a poté až jeden nebo více Data Response paketů. V deskriptoru jsou obsaženy informace o následujících datech a má daný formát, obr. 11. Pakety Data response budou popsány později, viz str. 20. [13]

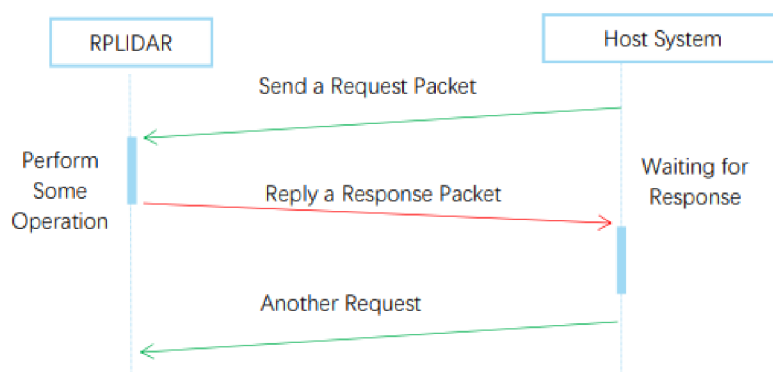


Obr. 11: Response Descriptor paket [13]

Dle typu Request paketu pak mohou nastat tři různé režimy komunikace.

### Režim Request/Response

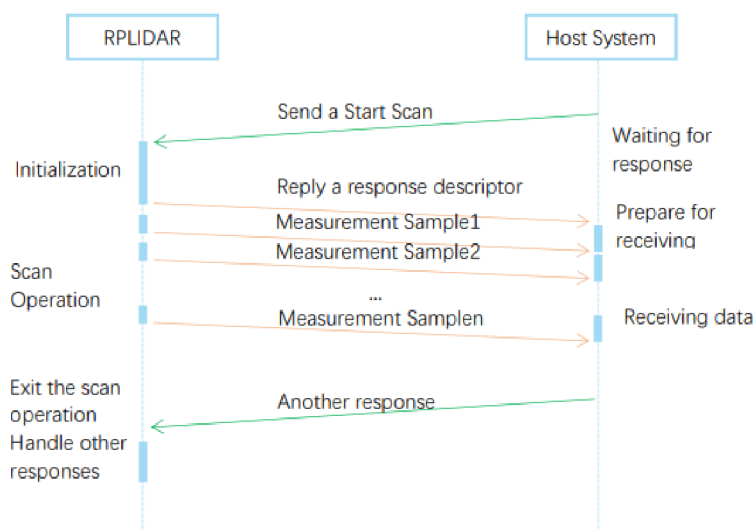
Pokud RPLIDAR zpracovává aktuální požadavek, mělo by se vyvarovat posílání dalšího. Extra paket by byl vyřazen a už by nebylo možné jej zpracovat. Schéma režimu ilustruje obr. 12. [13]



Obr. 12: Režim Request/Response [13]

### Režim Single Request - Multiple Response

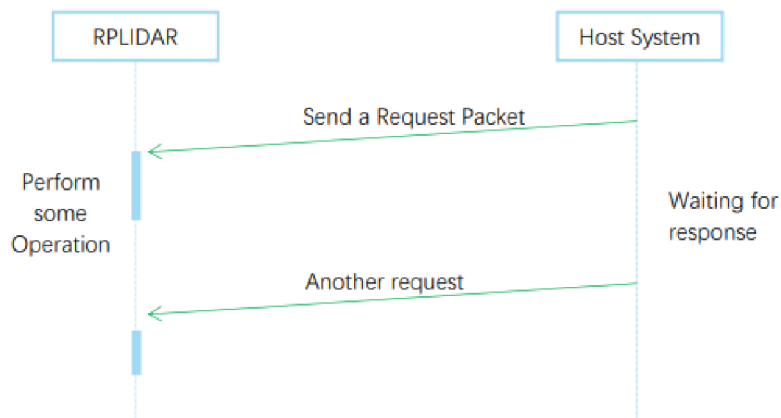
Tento režim je typický pro skenování. Poté co pošleme požadavek pro sken, RPLIDAR bude průběžně provádět měření a posílat jednotlivé pakety s daty. RPLIDAR lze vyrušit posláním STOP paketu nebo jakéhokoli jiného. Po opuštění režimu bude následně vyhodnocen Request paket, kterým jsme přerušili sken, viz obr. 13. [13]



Obr. 13: Režim Single Request – Multiple Response [13]

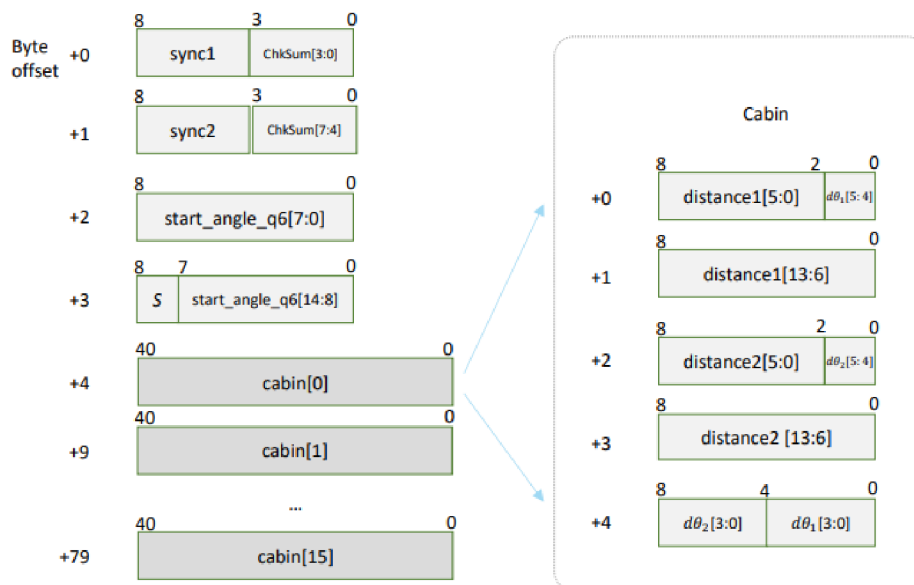
### Režim Single Request – No Response

Na obr. 14 je znázorněn režim Single Request – No Response, vhodný pro některé požadavky jako STOP, nebo RESET, kde není potřeba aby RPLIDAR odesílal Request paket. Je ovšem nutné po odeslání požadavku počkat, než RPLIDAR provede operaci, jinak by byl extra paket vyřazen. [13]



Obr. 14: Režim Single Request – No Response [13]

Vzhledem k využití RPLIDARu je pro nás nejdůležitější režim Single Request – Multiple Response. Nyní je třeba popsat Data Response pakety, které jsou průběžně odesílány RPLIDAREm. Ty se mohou lišit na základě pracovního režimu, tab. 5 na str. 16. V našem případě je použit režim Standard, který používá takzvanou Legacy verzi Data Response paketu., jehož formát se nachází na obr. 15.



Obr. 15: Legacy verze Data Response paketu [13]

Paket obsahuje 16 substruktur označovaných jako Cabin. Každá z nich obsahuje 5 bajtů a má specifickou strukturu odpovídající hodnotě úhlu a vzdálenosti ze dvou setů naměřených dat. [13]

### 2.3.5 Knihovna pro zpracování dat

Na zpracování protokolu k RPLIDAR A3M1 byla použita volně dostupná knihovna [15]. Pro celou knihovnu byl využit programovací jazyk Python a objektově orientované programování, díky kterému lze jednoduše přistupovat k jednotlivým funkcím RPLIDARu. Autor uvádí, že knihovna je kompatibilní s RPLIDAR A1, A2 a A3. Také je možné ji využít s operačním systémem Windows, Linux a MacOS. Při používání Windows je ovšem knihovna nefunkční a chyba není doposud opravena.

Nachází se zde také pár příkladů, kde je demonstrováno použití knihovny na jednoduchých úlohách.

V celém kódu, jak v příkladech, tak ve zpracování protokolu, je kompletní absence komentářů. Tento fakt zdatelně prodlužuje čas nutný k porozumění kódu a následnou práci s knihovnou.

## 2.4 Detekce osob pomocí LIDARu

Rozpoznávání lidí je pro roboty nebo inteligentní vozidla důležitou dovedností v nejrůznějších aplikacích. Roboti, kteří sdílejí prostor s lidmi potřebují detekovat, sledovat a analyzovat osoby kvůli bezpečnosti, interakce s uživatelem nebo navigaci. Zatímco je tento problém poměrně dobře prozkoumán v oblasti zpracování obrazu s využitím RGB – D kamer, v oblasti 2D vzdálenostních dat je stále velký prostor pro posun. [16]

Obecný postup u aplikací podobné této práci je takový, že se LIDAR upevní v určité výšce, a to buď ve výšce boků, nebo kolen, kde se následně vyhodnocuje detekce lidských nohou. Následně se z mračna dat získaných díky LIDARu pomocí clusteringu, viz kapitola 2.4.1, rozdělí data na shluky. Poté je třeba v datech správně definovat osobu, tento krok je náročný vzhledem k dynamice a rychlosti lidského pohybu.

Dalším krokem bývá vytvoření modelu pro strojové učení, který po trénování vede k poměrně vysoké efektivnosti. Následující výzvou je pak predikování pohybu uživatele.

Momentálně neexistuje mnoho volně dostupných modelů pro detekci osob pomocí 2D LIDARu, ale například projekt „Person Detection in 2D Range Data“ [17] obsahuje předtrénovaný model vytvořený pomocí Pytorch 1.6 a zároveň obsahuje i práci a video objasňující tuto problematiku.

### 2.4.1 Clustering

Clustering je proces při kterém rozdělujeme velké množství dat do shluků tak, aby datové body s podobnými vlastnostmi byly společně v jednom shluku. Tato definice je velmi obecná a metody clusteringu se výrazně liší na základě řešené problematiky. Mezi oblasti s využitím clusteringu patří například kolaborativní filtrování, segmentace zákazníků, sumarizace dat. [18]

V rámci rozdělení dat z LIDARu je vhodné vypíchnout algoritmy založené na vzdálenosti a algoritmy založené na hustotě.

### Algoritmy založené na vzdálenosti

Mnoho forem generativních algoritmů můžeme redukovat na algoritmy založené na vzdálenosti. Je to pro, že většina generativních algoritmů používá funkci vzdálenosti v rozdělení pravděpodobnosti. Například Gaussovo rozdělení představuje pravděpodobnosti generování dat v podobě euklidovské vzdálenosti od střední hodnoty. Výsledkem toho je, že generativní model s Gaussovým rozdělením má velmi blízko ke k-means algoritmu, jednomu z nejjednodušších algoritmů pro clustering. [18]

Modely založené na vzdálenosti jsou často používány pro svou jednoduchost a širokou škálu aplikací.

### Algoritmy založené na hustotě

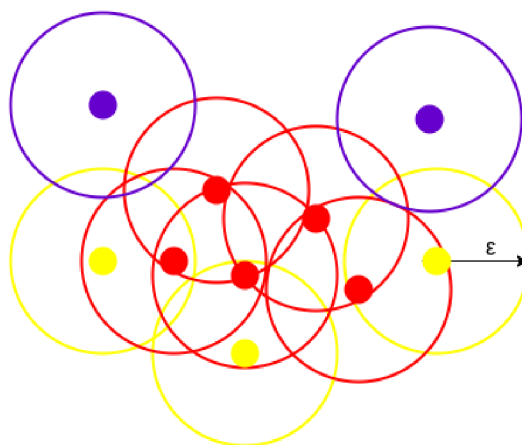
Algoritmy založené na hustotě hledají shluky pomocí množství datových bodů v určitém prostoru. [18] Mezi nejznámější metody se řadí DBSCAN (Density-Based Spatial Clustering of Applications with Noise), který je použit i v této práci a je popsán v následující kapitole 2.4.2.

#### 2.4.2 DBSCAN

Jak již bylo zmíněno DBSCAN je algoritmus založený na hustotě. Výhodou je, že při jeho použití nemusíme znát předem počet shluků.

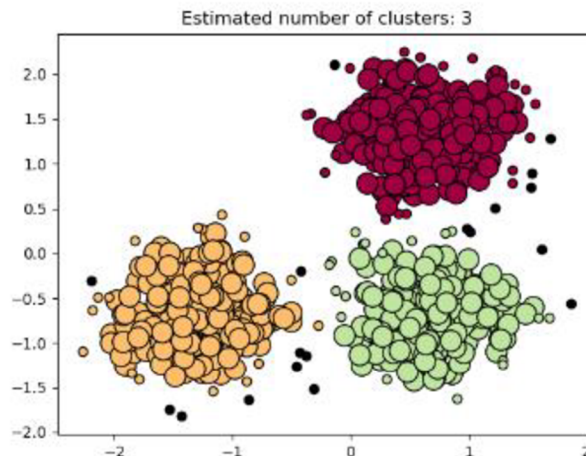
Při práci s 2D daty je třeba nastavit dva vstupní parametry,  $\epsilon$ , poloměr který určuje okolí bodu a minimální počet bodů, označovaný většinou jako minPoints.

Algoritmus funguje tak, že vybere libovolný bod a pokud se v jeho okolí daném radiusem  $\epsilon$  nachází alespoň minimální počet bodů, včetně vybraného, je bod označen jako jádro. V případě, že se v okolí nachází menší počet bodů, než je zadán, je bod označen jako hranice. Když se kolem bodu nenachází žádný další bod, je označen jako šum. Na obr. 16 jsou jádra vyznačena červeně, hranice žlutě a šum je označen modrou barvou, minPoints rovno 3. [19]



Obr. 16: Popis funkce DBSCAN [20]

Následně se vybere libovolné jádro a přiřadí se do prvního shluku, všechny ostatní jádra v jeho okolí se rovněž přiřadí do stejného shluku. Tak se rozdělí všechny body označené jako jádro do svých shluků. Následně se přidají ke shlukům body označené jako hranice. Pokud je tedy hraniční bod na pomezí dvou shluků, bude spadat do toho, který se vyhodnocoval jako první. Výsledek pak může vypadat jako na obr. 17. [19]



Obr. 17: Ukázka DBSCAN

### 2.4.3 Knihovna pro clustering

Pro clustering byla použita bezplatná softwarová knihovna scikit-learn [19], pro programovací jazyk Python. V knihovně je naprogramováno mnoho algoritmů pro clustering. U každé metody je navíc text popisující její funkce a přiložená ukázka použití.

Knihovna také obsahuje srovnání jednotlivých metod s doporučenými aplikacemi.

## 3 POSTUP A VÝSLEDKY ŘEŠENÍ

### 3.1 Postup řešení na základě rešerše

V této sekci bude nastíněn a odůvodněn zvolený postup praktické části na základě rešerše, konkrétní kroky pak budou popsány v jednotlivých kapitolách práce.

Pomocí RPLIDARu je pro robota vytvořena bezpečnostní zóna, ta je reprezentována kruhem kolem RPLIDARu o určitém velikosti, kterou si uživatel vybere. Aby bylo možné s RPLIDAREm komunikovat, byla zvolena řídicí jednotka Raspberry Pi 3B.

Jednotka obsahuje čtyři USB porty, s lidarem lze tedy komunikovat přes USB adaptér, kapitola 2.3.3, a stále je možnost připojení externích zařízení. Raspberry Pi 3B má zabudován také HDMI port, přes který byl připojen dotykový displej o velikosti uhlopříčky 5 palců, aby bylo možné získaná data zobrazovat a uživatel si mohl přehledně a jednoduše upravovat nastavení. Má také 17 GPIO pinů, pomocí kterých lze poslat signál do UR5e.

Ke psaní kódu byl použit programovací jazyk Python, který se těší stále větší popularitě a je pro něj vytvořeno mnoho knihoven. Pro detekci osob v zóně pak byly naprogramovány dva módy. První mód funguje na principu porovnání dat vzdáleností, druhý využívá DBSCAN.

K tvorbě uživatelského rozhraní byla použita multiplatformní knihovna Pygame, která je určena především pro tvorbu počítačových her v Pythonu. Zde je možnost vybrání velikosti bezpečnostní zóny, oblastí, ve kterých RPLIDAR nemá snímat nebo mód pro detekci osob. Získaná data z lidaru, ve smyslu vzdálenosti a úhlu, pak jsou zobrazovaná v kartézském souřadnicovém systému, pro snadnou představu o skenování.



## 3.2 Konstrukce

Aby nedošlo k poškození řídicí jednotky a displeje, byl vytvořen kryt, který umožňuje mít displej i jednotku jako jeden celek, anebo zvlášť. Toto pouzdro pak může být přichyceno k pracovnímu stolu robota pomocí vytvořeného držáku. Rovněž byl vytvořen držák pro RPLIDAR, aby mohl být upevněn zespod pracovního stolu a mohl tak skenovat okolí a nepřekážet v pracovním prostoru.

Všechny komponenty byly vytvořeny pomocí 3D tisku. Modely vytvořené v softwaru Autodesk Fusion 360 jsou obsaženy v příloze ve formátu STL.

### 3.2.1 Kryt Raspberry Pi a displeje

I přes popularitu RPi jsou pouzdra na jednotku a displej dostupné převážně pro displeje o velikosti 7 nebo 3,5 palců.

Kryt je navržen tak, že je možné Raspberry Pi používat samostatně v pouzdře a podle potřeby pak spojit s krytem na displej, aby vytvořili kompaktní set. K připevnění jednotky jsou třeba čtyři šrouby M3, stejný počet šroubů M3 je pak použit na připevnění displeje.



Obr. 18: Kryt Raspberry Pi a displeje

### 3.2.2 Připevnění setu

Pro upevnění setu k pracovnímu stolu byl zhotoven držák tak, aby se set nezasahoval do pracovního prostoru. K upevnění ke stolu jsou třeba dva šrouby M6, v držáku jsou také dvě díry, aby se set mohl připojit pomocí šroubů M3.



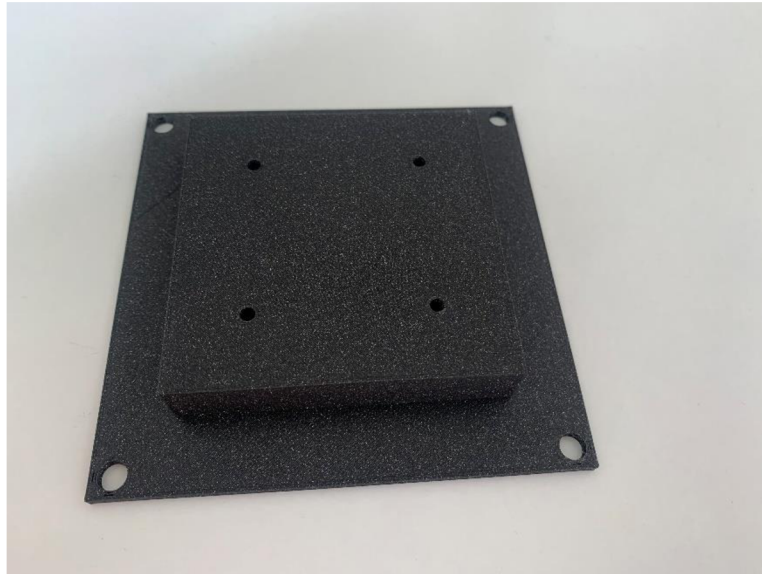
Obr. 19: Držák setu



Obr. 20: Upevnění setu k pracovnímu stolu

### 3.2.3 Připevnění RPLIDARu A3M1

V RPLIDARu A3M1 jsou čtyři závity M3, byla tak vytvořena podstava, ke které se dá RPLIDAR přidělat a celá podstava pak může být připevněna zespod pracovního stolu pomocí čtyř šroubů M6.



Obr. 21: Podstava pro RPLIDAR A3M1



Obr. 22: Upevnění RPLIDARu A3M1 k pracovnímu stolu

### 3.3 Komunikační rozhraní

V této kapitole bude popsáno komunikační rozhraní mezi robotem, RPLIDAREm a řídicí jednotkou.

### 3.3.1 Komunikace RPi a RPLIDARu

Komunikace je zprostředkována pře USB adaptér (kap. 2.3.2) s pomocí knihovny (kap. 2.3.5). V té je pak důležitý script „pyrplidar\_serial.py“ a funkce „connect“ ze scriptu „pyrplidar.py“. Připojení RPLIDARu v hlavních programu pak vypadá následovně.

---

```
#connect lidar
lidar = PyRPlidar()
lidar.connect(port="/dev/ttyUSB0", baudrate=256000 , timeout=3)
```

---

Je také nutné nastavit správnou hodnotu baudratu na samotném USB adaptéru.

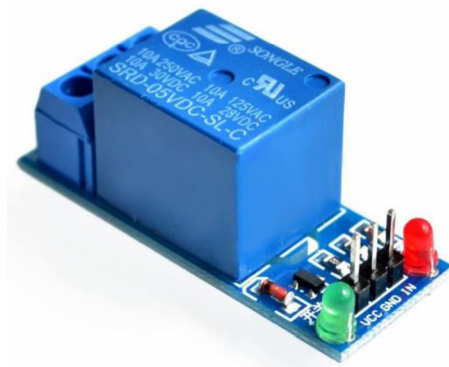
### 3.3.2 Komunikace RPi a UR5e

Pro komunikaci řídicí jednotky s robotem jsou využity GPIO piny, rozložení pinů RPi je vyobrazeno na obr. 21.

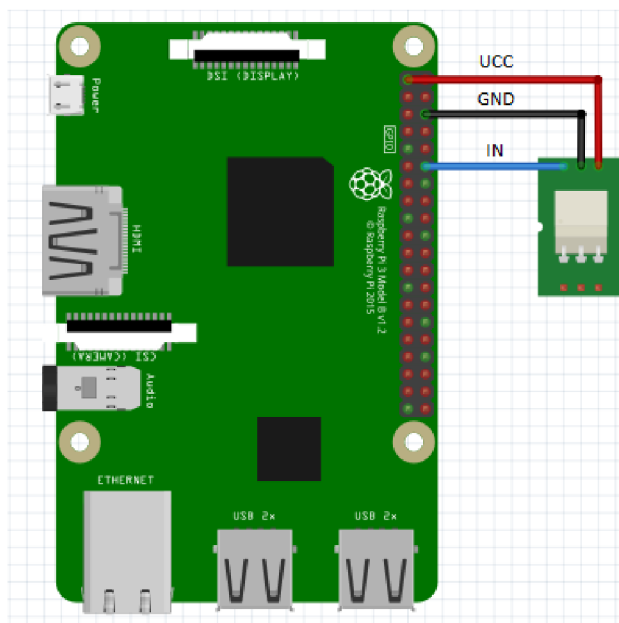
<b>3V3</b>	(1)	(2)	<b>5V</b>
<b>GPIO2</b>	(3)	(4)	<b>5V</b>
<b>GPIO3</b>	(5)	(6)	<b>GND</b>
<b>GPIO4</b>	(7)	(8)	<b>GPIO14</b>
<b>GND</b>	(9)	(10)	<b>GPIO15</b>
<b>GPIO17</b>	(11)	(12)	<b>GPIO18</b>
<b>GPIO27</b>	(13)	(14)	<b>GND</b>
<b>GPIO22</b>	(15)	(16)	<b>GPIO23</b>
<b>3V3</b>	(17)	(18)	<b>GPIO24</b>
<b>GPIO10</b>	(19)	(20)	<b>GND</b>
<b>GPIO9</b>	(21)	(22)	<b>GPIO25</b>
<b>GPIO11</b>	(23)	(24)	<b>GPIO8</b>
<b>GND</b>	(25)	(26)	<b>GPIO7</b>
<b>GPIO0</b>	(27)	(28)	<b>GPIO1</b>
<b>GPIO5</b>	(29)	(30)	<b>GND</b>
<b>GPIO6</b>	(31)	(32)	<b>GPIO12</b>
<b>GPIO13</b>	(33)	(34)	<b>GND</b>
<b>GPIO19</b>	(35)	(36)	<b>GPIO16</b>
<b>GPIO26</b>	(37)	(38)	<b>GPIO20</b>
<b>GND</b>	(39)	(40)	<b>GPIO21</b>

Obr. 23: Rozložení pinů Raspberry Pi 3B

Výstup GPIO u jednotky je 3,3 V, digitální piny kolaborativního robota UR5e ovšem pracují na 24 V logice, je proto potřeba připojit ještě jednobaný 5 V relé modul, který může řídit až 10 A 28 VDC. Ačkoliv je na 5 V, osvědčilo se, že funguje dobře i k RPi.



Obr. 24: Relé modul [21]



Obr. 25: Schéma zapojení RPi a relé modulu

Jak je znázorněno na obr. 23, k modulu je potřeba připojit napájecí napětí 3,3 V (červená), poté se propojí země (černá) a nakonec vstupní signál (modrá). V práci byl využit pin GPIO18. Práce s piny je v kódu následující.

---

```
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM) #refer to the GPIO by number
GPIO.setup(18, GPIO.OUT) #set GPIO18 as output
```

---

Při detekci osoby je pak poslán signál a zapnut relé modul.

---

```
GPIO.output(18, False)
```

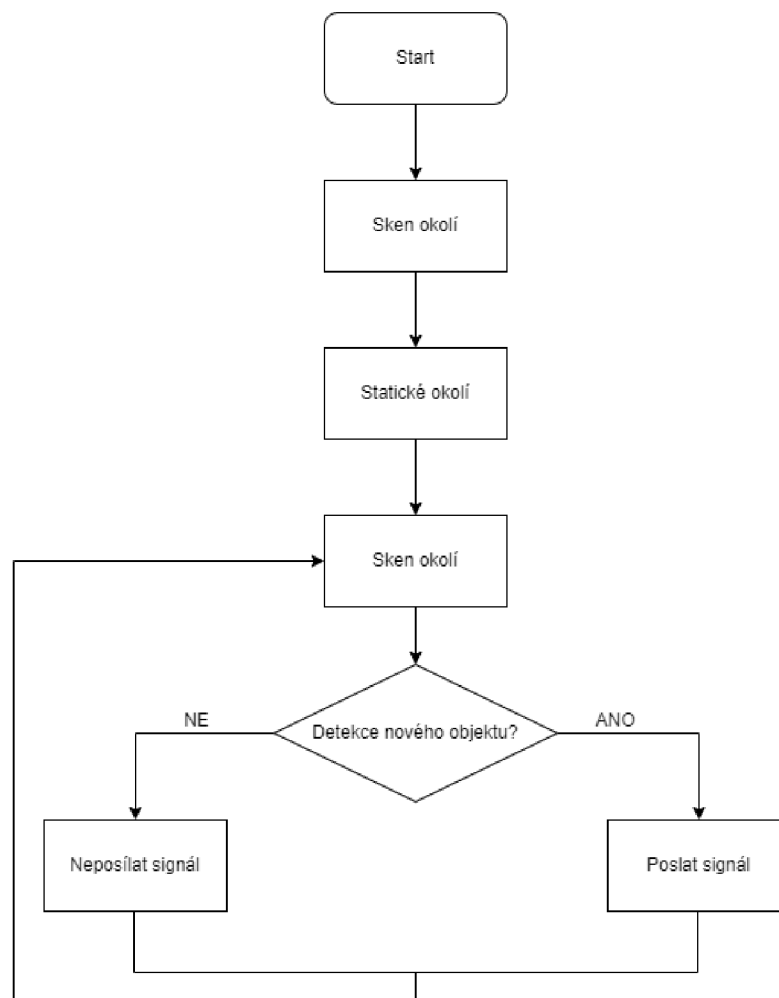
---

V opačném případě je druhý parametr v závorce nastaven jako „True“.

Připojení relé modulu k robotu již není součástí práce. Kontrola funkčnosti relé je prováděna LED diodou, která je součástí modulu.

### 3.4 Detekce pomocí porovnání dat vzdáleností

Tato kapitola se zabývá prvním ze dvou módů pro detekci osob v bezpečnostní zóně robota. Detekce zde probíhá na základě porovnání vzdáleností, kdy je nejprve naskenováno okolí, které považujeme za statické a slouží k porovnávání s následným skenováním.



Obr. 26: Vývojový diagram „Detekce pomocí porovnání dat vzdáleností“

#### 3.4.1 Získání a zpracování dat

Zapnutí lidarů a následné získání souřadnic probíhá vždy velmi podobně dle následujícího vzoru.

---

```

lidar = PyRPLidar()
lidar.connect(port="/dev/ttyUSB0", baudrate=256000 , timeout=3)#connect

coordinates = []

lidar.set_motor_pwm(660)#start motor
time.sleep(2)

scan_generator = lidar.start_scan_express(0)#start scan

for counter,scan in enumerate(scan_generator()):

    quality = scan.quality
    angle = scan.angle
    distance = scan.distance

    if quality != 0:#ignore low quality data
        coordinates.append((angle,distance))

    if counter == 720:#break after 720 iters
        break

#stop and disconnect lidar
lidar.stop()
lidar.set_motor_pwm(0)
lidar.disconnect()

```

---

Aby bylo možné srovnávat data, které bereme jako statická a definují nám neměnné původní okolí, s daty získané průběžným skenováním, je třeba je upravit.

Je tedy vytvořen list s dvojicemi dat ve tvaru (úhel, vzdálenost) seřazených vzestupně podle úhlu, kde každé polovině jednoho stupně úhlu je přiřazena daná vzdálenost získaná aritmetickým průměrem získaných vzdáleností kolem oné polohy. Pro úhly, které nebyly změřeny z důvodu špatné kvality dat je fixně určená vzdálenost 3 m, tj. největší možná bezpečnostní zóna.

Níže je demonstrováno zpracování dat pro získání statických vzdáleností, obdobné zpracování pak probíhá při každém detekčním skenování.

---

```

base_list = [(x/2,3000) for x in range(0,720)] #full angle list
#sorted coords low to high
base_coordinates = sorted(base_coordinates_fix, key = lambda tup: tup[0])
#round angle to the closest half
round_coords = [round_half(coord) for coord in base_coordinates]
#take list of tuples, do average of R for values with same Fi and create new
list looks like [(fi1,r1), (fi2,r2)...]
base_sort_coordinates = []
prev_fi = None
sum_r = 0
num_r = 0

```

---

---

```

for fi,r in round_coords:
    if fi != prev_fi:
        if prev_fi is not None:
            average_r = sum_r / num_r
            base_sort_coordinates.append((prev_fi,average_r))

            prev_fi = fi
            sum_r = r
            num_r = 1
        else:
            sum_r += r
            num_r += 1

#change dist in tuples with same angle
diction = dict(base_sort_coordinates)
for i, dist in enumerate(base_list):
    if dist[0] in diction:
        base_list[i] = (dist[0],diction[dist[0]])

#base distances
base_distance = [r for (fi,r) in base_list]

```

---

### 3.4.2 Detekce

Detekce osoby je založena na jednoduchém principu. Pokud u šesti bodů jdoucích po sobě bude vzdálenost menší, než jak je určena ve statických datech (base\_distance), program vyhodnotí výskyt nového objektu, pošle signál pro zapnutí relé modulu a změní barvu, kterou se zobrazují data na displeji.

---

```

dif_count = 0 #counter for 6 points in a row
for i in range(len(base_distance)):
    if dif_count == 6:
        break
    if base_distance[i] - new_dist[i] > 30:
        dif_count += 1
    else:
        dif_count = 0

if dif_count == 6: #new object detect
    GPIO.output(18,False)
    for x,y in scan_print:
        pygame.draw.circle(screen,red,(int((width/2)+x),int((height/2)+
y)),1)
else: #area is same
    GPIO.output(18,True)
    for x,y in scan_print:
        pygame.draw.circle(screen,white,(int((width/2)+x),int((height/2
)+y)),1)

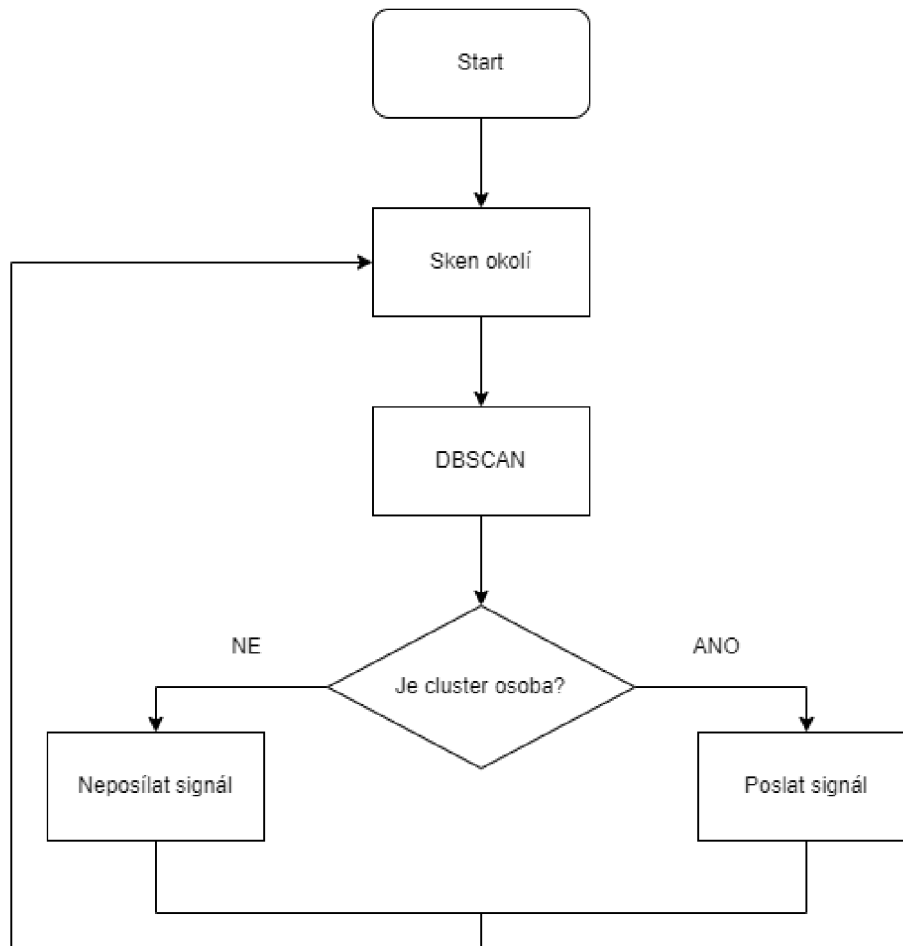
```

---



### 3.5 Detekce pomocí clusteringu

Druhý dostupný mód pro detekci osob v bezpečnostní zóně využívá clustering, konkrétně pak metodu DBSCAN (kap. 2.4.2).

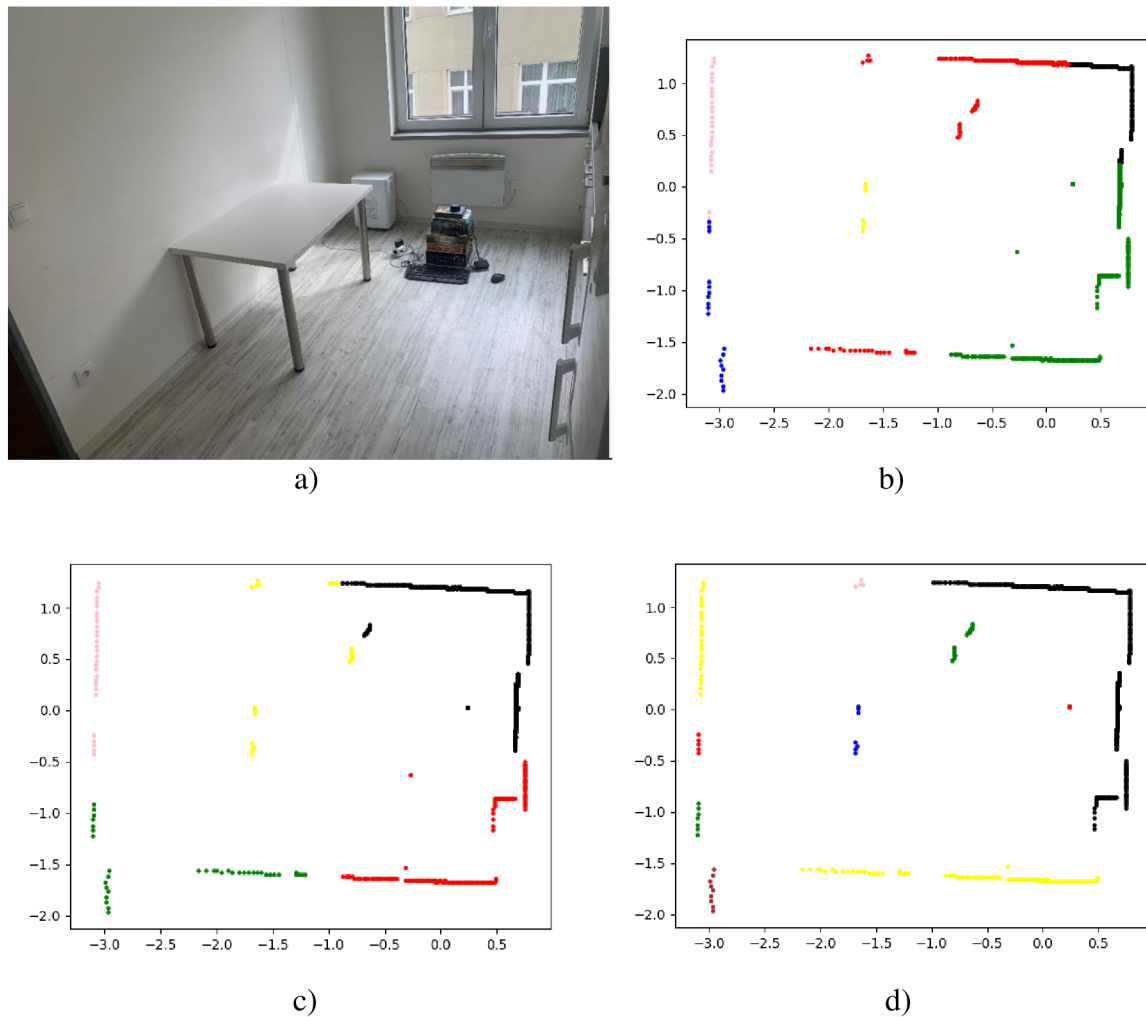


Obr. 27: Vývojový diagram „Detekce pomocí clusteringu“

#### 3.5.1 Porovnání cluster metod

K výběru metody DBSCAN vedlo porovnání tří rozdílných způsobů clusteringu: DBSCAN, BIRCH a Meanshift. Všechny metody byly odzkoušeny na stejném datasetu a vyhodnoceny na základě vykreslení prostoru, kde každý datový shluk je vyznačen jinou barvou.

Na obr. 26 lze vidět, že pomocí DBSCANu se podařilo označit každou ze dvou osob jako samostatný shluk dat.



Obr. 28: Porovnání cluster metod: a) měřený prostor b) BIRCH c) MeanShift d) DBSCAN

### 3.5.2 Zpracování dat

Úprava dat z RPLIDARu pro DBSCAN je taková, že nejprve se data převedou do kartézských souřadnic, tento krok není nutný, ale byl zvolen kvůli vykreslování v kapitole 3.5.1, poté se pomocí knihovny Numpy vytvoří z dat matice. Následně se pomocí funkcí knihovny scikit-learn (kap. 2.4.3) rozdělí data do shluků a jednotlivé datové shluky se označí.

---

```
#cartesian
cluster_scan_kart = [(int(r * cos(radians(fi))),int(r * sin(radians(fi))))
for (fi,r) in cluster_scan_sort]
#clustering
X = np.array(cluster_scan_kart)
X = StandardScaler().fit_transform(X)

dbscan = DBSCAN(eps = 0.35,min_samples = 6)
labels = dbscan.fit_predict(X)
```

---

### 3.5.3 Detekce

Vyhodnocení, zda je daný datový shluk osoba je zde provedeno pomocí kontroly velikosti shluku. Objekty podobné velikosti tak mohou být také označeny jako člověk.

Pokud je detekována osoba, opět se zapne relé modul a změní se barva zobrazující skenovaná data.

---

```
for i in range(max(labels)):
    cluster = X[labels == i]
    cluster_size = len(cluster)

    if 4 < cluster_size < 10:
        GPIO.output(18, False)
        for x, y in cluster_print:
            pygame.draw.circle(screen, red, (int((width/2)+x), int((height/2)+
y)), 1)
    else:
        GPIO.output(18, True)
        for x, y in cluster_print:
            pygame.draw.circle(screen, white, (int((width/2)+x), int((height/2)
)+y)), 1)
```

---

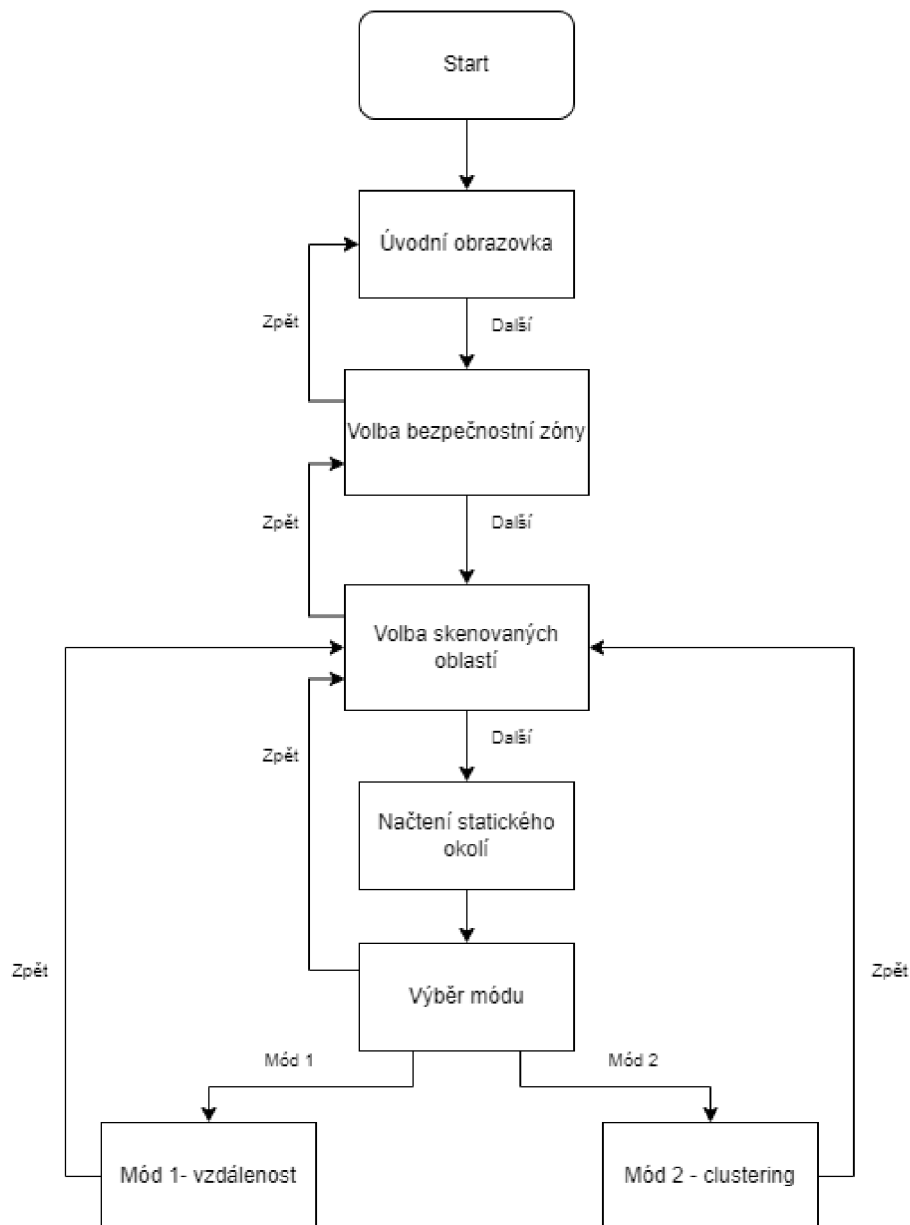
## 3.6 GUI

Aby mohl uživatel nastavovat parametry skenování, bylo vytvořeno GUI, které jde ovládat pouze pomocí dotyku na obrazovce displeje.

### 3.6.1 Stavy a jejich funkce

Rozhraní funguje jako stavový automat. Mezi jednotlivými stavy lze přepínat pomocí ikon [22]. Celé uživatelské prostředí bylo vytvořeno se snahou o intuitivní a jednoduché ovládání.

Jednotlivé stavy jsou navíc doplněné o pole s instrukcemi a informacemi o prováděných operacích.



Obr. 29: Diagram stavů GUI

Popis stavů:

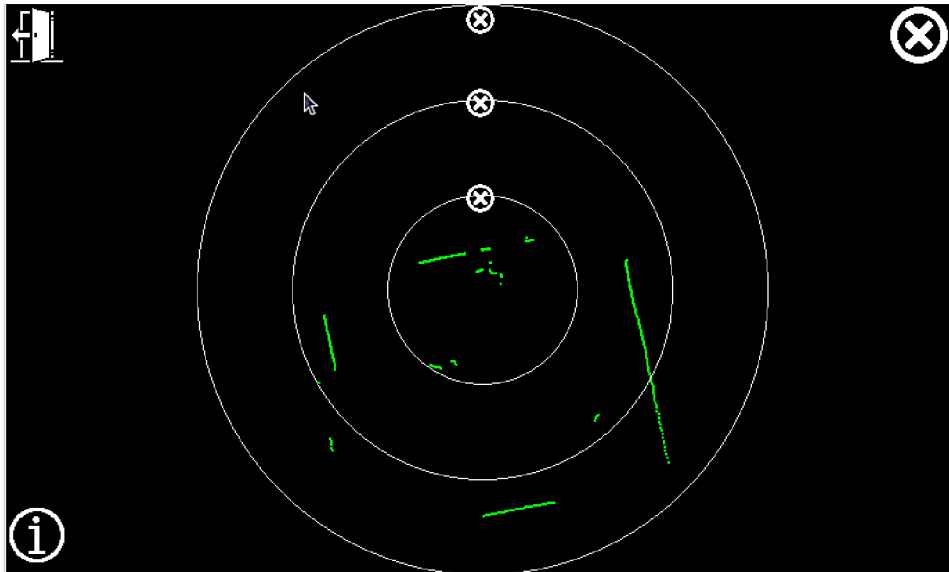
### Úvodní obrazovka

Po stisknutí ikony se zapne jednoduchý sken okolí, který slouží pouze pro vizualizaci dat v ostatních stavech, aby měl uživatel představu, co RPLIDAR skenuje.

### Volba bezpečnostní zóny

Zde si uživatel vybere jednu ze tří velikostí bezpečnostní zóny kolem robota, 3 m, 2 m a 1 m. Bezpečnostní zóny jsou demonstrovány bílými kružnicemi. Výběr probíhá pomocí stisknutí ikony křížku. Naměřená data jsou zobrazena zelenou barvou. Po vybrání zóny se přizpůsobí zobrazení dat pro lepší přehlednost a přejde se na další stav.

V levém dolním rohu se nachází ikona, díky které budou vypsány podstatné informace o daném stavu.



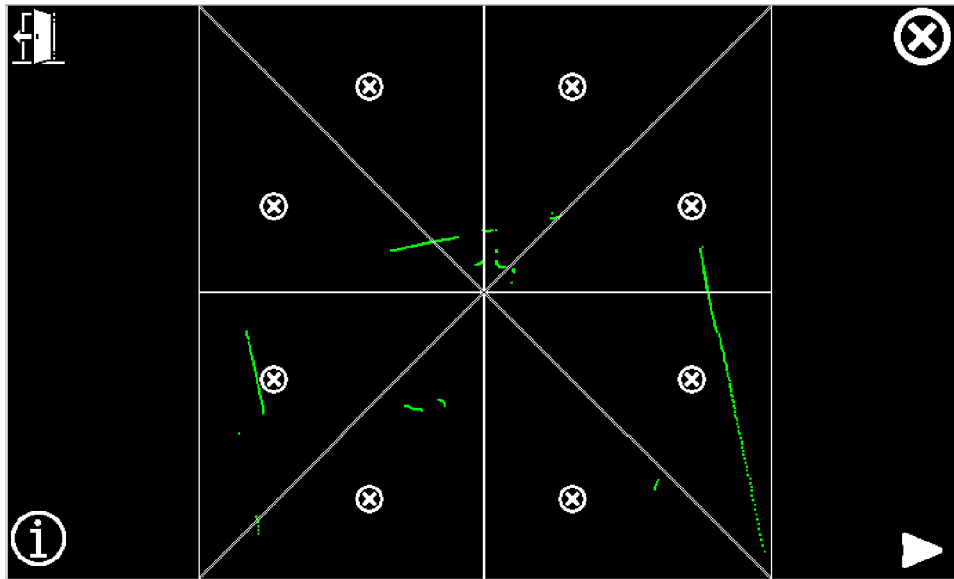
Obr. 30: Volba bezpečnostní zóny

### **Volba skenovaných oblastí**

V tomto stavu může uživatel vybrat, které části okolí nebude brát skenování v potaz. Může se tak jednat například o prostor kde je předpokládán bezpečný pohyb pracovníků. Pro volbu oblastí skenu jsou vytvořeny výseče ohraničené bílými hranicemi. Po stisknutí ikon křížku hranice změni barvu na červenou, výseče takto ohraničené pak nebudou skenovány.

Poté co se stiskne ikona pro další, zapne se sken, který načte a definuje statické okolí, které je následně využito pro detekci (kap. 3.4).

V levém dolním rohu je opět ikona s instrukcemi, v pravém pak ikonu pro zapnutí skenu.

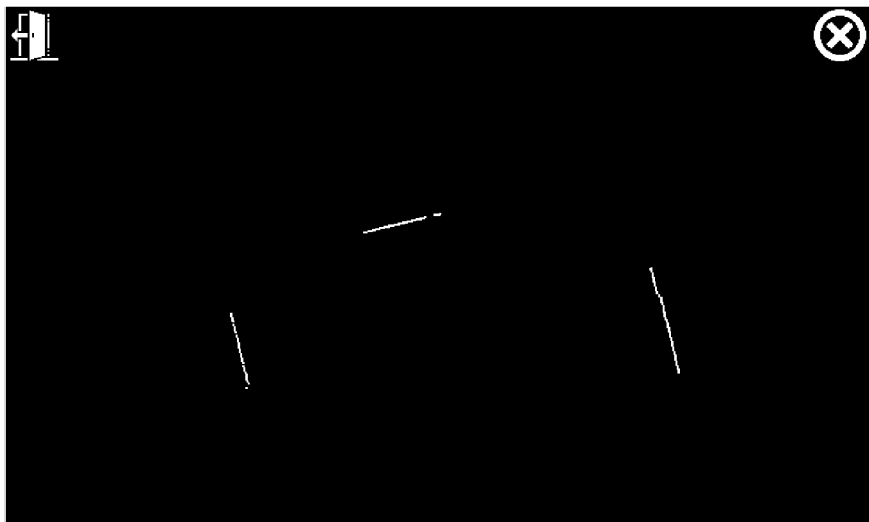


Obr. 31: Volba skenovaných oblastí

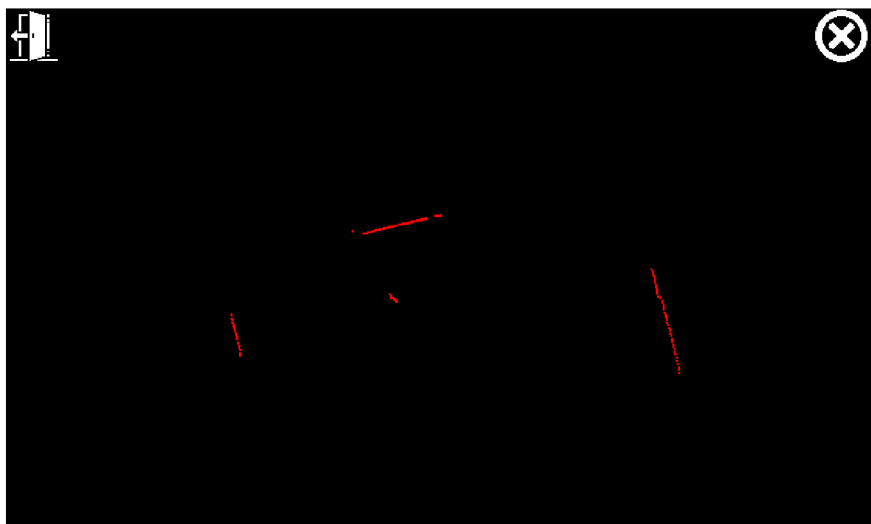
### Výběr módů

Na obrazovce se objeví dvě ikony, díky kterým si lze vybrat mód na základě kterého bude probíhat detekce. Jednotlivé módy byly popsány v kapitolách 3.4 a 3.5.

Po vybrání vypadá obrazovka pro obě varianty stejně, obr. 30.



Obr. 32: Zobrazení dat – osoba není detekována



Obr. 33: Zobrazení dat – osoba je detekována

## 4 ZÁVĚR

Cílem práce bylo naprogramování algoritmu, který vytvoří bezpečnostní zóny okolo kolaborativního robota UR5e s využitím laserového skeneru, konkrétně RPLIDARu A3M1.

Úvod práce se tak zabývá problematikou v oblasti bezpečnosti kolaborativních robotů se zaměřením na jejich pracovní zóny a kontakt s pracovníky.

Pro práci se samotným RPLIDAREm A3M1 bylo potřeba nastudovat jeho možnosti nastavení, strukturu posílaných dat a jak je možné s ním komunikovat. Pro komunikaci pak byla vybrána řídicí jednotka Raspberry Pi 3B.

Po prozkoumání možností RPLIDARu a jednotky, bylo přistoupeno k návrhu algoritmu, který bude umožňovat nastavení parametrů bezpečnostních zón uživatelem pomocí uživatelského rozhraní a dotykového displeje. Uživatel bude mít také možnost vybrat si jeden ze dvou módů pro detekci osob, které se liší principem fungování. Celý algoritmus je pak psán pomocí programovacího jazyka Python.

GUI je vytvořeno jako stavový automat. Umožňuje tak obsluze vybírat jednu ze tří bezpečnostních zón (3 m, 2 m, 1 m), dále je zde možnost nastavit konkrétní oblasti snímání okolí, a kromě zobrazování naměřených dat z laserového skeneru jsou zde také dostupné instrukce a informace pro snadnější práci s RPLIDAREm a volbu pracovního módu.

První mód k detekci osob využívá porovnání vzdáleností nově skenovaných dat, se vzdálenostmi, které jsou nastaveny jako výchozí a jsou tvořeny statickými překážkami v okolí robota. Algoritmus je funkční a dokáže detekovat nový objekt velmi rychle, avšak občas se stává, že po opuštění bezpečnostní zóny se program stále chová tak, jako by v zóně byla osoba. Tento problém je možné opravit pouhým znovu spuštěním daného módu. To ovšem vede ke snížení efektivity práce, bezpečnostní složka je ale zachována.

Druhý mód pak pro detekci osob využívá clustering, konkrétně metodu DBSCAN. Zde se získaná mračna dat rozdělí na samostatné shluky s podobnými vlastnostmi. Detekce osoby pak probíhá na základě velikosti daného shluku. Algoritmus je sice funkční, ale má nedostatky, jedním z nich je samotný clustering. Pokud se osoba nachází blízko překážky, nebo je více lidí vedle sebe, není vyhodnocena jako samostatný shluk a není pak detekována.

Aby bylo možné RPLIDAR připevnit ke kolaborativnímu robotovi a mít v blízkosti také dotykový displej pro obsluhu, byly pomocí 3D tisku vytvořeny komponenty sloužící k upevnění jednotlivých dílů k pracovnímu stolu ramene UR5e.

Možná budoucí vylepšení algoritmu jsou především v metodě detekce pomocí clusteringu. Zde by bylo možné nejprve pomocí RPLIDARu získat dataset, ten by se pak použil pro trénování modelu strojového učení. Následně by program mohl být schopen konkrétně rozlišit, zda se jedná o osobu a případně také predikovat její pohyb a tím zvýšit efektivitu celého algoritmu.



## 5 BIBLIOGRAFIE

- 1] MURASHOV, Vladimir, Frank HEARL a John HOWARD. Working safely with robot workers: Recommendations for the new workplace. *Journal of occupational and environmental hygiene* [online]. PHILADELPHIA: Taylor & Francis, 2016, **13**(3), 61-71 [cit. 2023-05-06]. ISSN 1545-9624. Dostupné z: doi:10.1080/15459624.2015.1116700
- 2] Collaborative "Cobot" robot safety. In: *Machine safety specialist* [online]. [cit. 2023-05-06]. Dostupné z: <https://www.machinesafetyspecialists.com/services/collaborative-robot-safety-cobots/>
- 3] BARTOŠÍK, Petr. Bezpečnost kolaborativních robotů. *Automa: časopis pro automatizační techniku* [online]. 2017, **2017**(8), 74-76 [cit. 2023-05-06]. Dostupné z: [https://automa.cz/cz/casopis-clanky/bezpecnost-kolaborativnich-robotu-2017\\_08\\_0\\_11040/](https://automa.cz/cz/casopis-clanky/bezpecnost-kolaborativnich-robotu-2017_08_0_11040/)
- 4] KOLÍBAL, Zdeněk. *Roboty a robotizované výrobní technologie*. Brno : Vysoké učení technické v Brně - nakladatelství VUTIUM, 2016, 787 stran : ilustrace (převážně barevné), portréty. ISBN 978-80-214-4828-5.
- 5] BAEK, Seungmin. *DEMYSTIFYING COBOT SAFETY: THE FOUR TYPES OF COLLABORATIVE OPERATION* [online]. **2016** [cit. 2023-05-06]. Dostupné z: <https://www.universal-robots.com/blog/demystifying-cobot-safety-the-four-types-of-collaborative-operation/>
- 6] OWEN-HILL, Alex. *Cobot Safety: Are You Too Close to Your Robot?* [online]. [cit. 2023-05-07]. Dostupné z: <https://blog.robotiq.com/cobot-safety-are-you-too-close-to-your-robot>
- 7] OWEN-HILL, Alex. *Adding extra sensors: How to do even more with collaborative robots* [online]. In: . 1. 2016, s. 16-21 [cit. 2023-05-07]. Dostupné z: [https://blog.robotiq.com/hubfs/eBooks/SensorsForCollaborativeRobots\\_Ebook.pdf?hslang=en-ca](https://blog.robotiq.com/hubfs/eBooks/SensorsForCollaborativeRobots_Ebook.pdf?hslang=en-ca)
- 8] CHEMWENO, Peter a Robbert-jan TORN. Innovative safety zoning for collaborative robots utilizing Kinect and LiDAR sensory approaches. *Procedia CIRP* [online]. Elsevier B.V, 2022, **106**, 209-214 [cit. 2023-05-07]. ISSN 2212-8271. Dostupné z: doi:10.1016/j.procir.2022.02.180
- 9] *Universal Robots e-Series User Manual: UR5e* [online]. Version 5.7. [cit. 2023-05-07]. Dostupné z: [https://s3-eu-west-1.amazonaws.com/ur-support-site/68265/99455\\_UR5e\\_User\\_Manual\\_en\\_US.pdf](https://s3-eu-west-1.amazonaws.com/ur-support-site/68265/99455_UR5e_User_Manual_en_US.pdf)

- UNIVERSAL ROBOTS. *UR5e technical details* [online]. [cit. 2023-05-07].  
10] Dostupné z: [https://www.universal-robots.com/media/1802778/ur5e-32528\\_ur\\_technical\\_details\\_.pdf](https://www.universal-robots.com/media/1802778/ur5e-32528_ur_technical_details_.pdf)
- UNIVERSAL ROBOTS. *OEM Control box installation guide* [online]. 2019  
11] [cit. 2023-05-08]. Dostupné z: <https://www.elmark.com.pl/amfile/file/download/file/134288/product/69832>
- SLAMTEC. *RPLIDAR A3 Introduction and Datasheet* [online]. Rev. 1.9. 2021  
12] [cit. 2023-05-08]. Dostupné z: [https://bucket-download.slamtec.com/bd39cf1da6f2c64fdb178440106561dc9fc0aa6d/LD310\\_SLAMTEC\\_rplidar\\_datasheet\\_A3M1\\_v1.9\\_en.pdf](https://bucket-download.slamtec.com/bd39cf1da6f2c64fdb178440106561dc9fc0aa6d/LD310_SLAMTEC_rplidar_datasheet_A3M1_v1.9_en.pdf)
- SLAMTEC. *RPLIDAR Interface Protocol and Application Notes* [online]. Rev.  
13] 2.2. 2021 [cit. 2023-05-08]. Dostupné z: [https://bucket-download.slamtec.com/6494fd238cf5e0d881f56d914c6d1f355c0f582a/LR001\\_SLAMTEC\\_rplidar\\_protocol\\_v2.4\\_en.pdf](https://bucket-download.slamtec.com/6494fd238cf5e0d881f56d914c6d1f355c0f582a/LR001_SLAMTEC_rplidar_protocol_v2.4_en.pdf)
- SLAMTEC. *Development Kit User Manual* [online]. Rev. 1.3. 2021 [cit. 2023-  
14] 05-10]. Dostupné z: [https://bucket-download.slamtec.com/03e2827d252c4429220f6540381d01a963aea171/LM310\\_SLAMTEC\\_rplidarkit\\_usermanual\\_A3M1\\_v1.3\\_en.pdf](https://bucket-download.slamtec.com/03e2827d252c4429220f6540381d01a963aea171/LM310_SLAMTEC_rplidarkit_usermanual_A3M1_v1.3_en.pdf)
- HYUN-JE. Pyrplidar. In: *GitHub* [online]. [cit. 2023-05-19]. Dostupné z:  
15] <https://github.com/Hyun-je/pyrplidar>
- HERMANS, Alexander, Alexander LINDER, Kai O. ARRAS, Bastian LEIBE  
16] a Lucas BEYER. *Deep Person Detection in 2D Range Data*.
- VISUALCOMPUTININSTITUT. *2D\_lidar\_person\_detection*. In: *GitHub*  
17] [online]. 2021 [cit. 2023-05-19]. Dostupné z: [https://github.com/VisualComputingInstitute/2D\\_lidar\\_person\\_detection](https://github.com/VisualComputingInstitute/2D_lidar_person_detection)
- AGGARWAL, Charu C a Chandan K REDDY. *Data Clustering*. 31. 1.  
18] Philadelphia, PA: CRC Press, 2014, . ISBN 9781466558212. Dostupné z: doi:10.1201/9781315373515
- Scikit-learn: Machine Learning in Python*. JMLR 12. 2011, pp.2825-2830.  
19] Dostupné také z: <https://scikit-learn.org/stable/modules/clustering.html#dbscan>
- SHARMA, Abhishek. How to Master the Popular DBSCAN Clustering  
20] Algorithm for Machine Learning. In: *Analytics Vidhya* [online]. 2022 [cit. 2023-05-19]. Dostupné z: <https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/>
- 1-kanál 5V relé modul Tongling, 250VAC 10A. In: *Laskakit* [online]. [cit.  
21] 2023-05-21]. Dostupné z: [https://www.laskakit.cz/1-kanal-5v-rele-modul-tongling--250vac-10a/?gclid=CjwKCAjwgqejBhBAEiwAuWHioPxxYciSxmy93dl0Bkx5nUGL1q2nDu2YAOoNLeHbE6xcBZcueFDaLLhoCuaIQAvD\\_BwE](https://www.laskakit.cz/1-kanal-5v-rele-modul-tongling--250vac-10a/?gclid=CjwKCAjwgqejBhBAEiwAuWHioPxxYciSxmy93dl0Bkx5nUGL1q2nDu2YAOoNLeHbE6xcBZcueFDaLLhoCuaIQAvD_BwE)

In: *Games-icons* [online]. [cit. 2023-05-22]. Dostupné z: <https://game-icons.net/tags/gui.html>

## 6 SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK

### Seznam obrázků

Obr. 1: Využití světelného závěsu pro detekci osob .....	12
Obr. 2: Využití laserového skeneru pro detekci osob .....	12
Obr. 3: UR5e .....	13
Obr. 4: Pracovní prostor UR5e.....	14
Obr. 5: Uspořádání I/O v ovládací skříni UR5e .....	14
Obr. 6: Porty pro připojení externích zařízení k UR5e .....	15
Obr. 7: RPLIDAR A3M1 .....	15
Obr. 8: Komunikační a napájecí rozhraní RPLIDAR .....	17
Obr. 9: Připojení RPLIDARu přes USB adaptér .....	17
Obr. 10: Request paket .....	18
Obr. 11: Response Descriptor paket .....	18
Obr. 12: Režim Request/Response .....	19
Obr. 13: Režim Single Request – Multiple Response .....	19
Obr. 14: Režim Single Request – No Response .....	20
Obr. 15: Legacy verze Data Response paketu .....	20
Obr. 16: Popis funkce DBSCAN .....	22
Obr. 17: Ukázka DBSCAN.....	23
Obr. 18: Kryt Raspberry Pi a displeje .....	25
Obr. 19: Držák setu.....	26
Obr. 20: Upevnění setu k pracovnímu stolu .....	26
Obr. 21: Podstava pro RPLIDAR A3M1 .....	27
Obr. 22: Upevnění RPLIDARu A3M1 k pracovnímu stolu .....	27
Obr. 23: Rozložení pinů Raspberry Pi 3B .....	28
Obr. 24: Relé modul .....	29
Obr. 25: Schéma zapojení RPi a relé modulu .....	29
Obr. 26: Vývojový diagram „Detekce pomocí porovnání dat vzdáleností“ .....	30
Obr. 27: Vývojový diagram „Detekce pomocí clusteringu“ .....	33
Obr. 28: Porovnání cluster metod: a) měřený prostor b) BIRCH c) Meanshift d) DBSCAN ..	34
Obr. 29: Diagram stavů GUI.....	36
Obr. 30: Volba bezpečnostní zóny .....	37
Obr. 31: Volba skenovaných oblastí .....	38
Obr. 32: Zobrazení dat – osoba není detekována .....	38
Obr. 33: Zobrazení dat – osoba je detekována.....	39

### Seznam tabulek

Tab. 1: Přehled norem uplatňovaných při posuzování bezpečnosti pracovišť s roboty .....	11
Tab. 2: Vybrané technické parametry UR5e .....	13
Tab. 3: Popis barevného schématu I/O UR5e .....	14
Tab. 4: Technické specifikace RPLIDAR A3M1 .....	16
Tab. 5: Režimy skenování RPLIDAR A3M1 .....	16

**Seznam zkratk**

I/O - Input/Output

PLC - Programmable logic controller

UART - Universal asynchronous receiver-transmitter

RGB – D - Red Green Blue – Depth

RPi - Raspberry Pi

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

GPIO - General-purpose input/output

GUI - Graphic user interface

## 7 SEZNAM PŘÍLOH

Složka „bp\_scripts“ obsahuje použité skripty v programovacím jazyce Python.

Složka „bp\_models“ obsahuje modely pro 3D tisk ve formátu stl.