

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра комп'ютерної інженерії

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри,
д-р техн. наук, проф.

_____ І. М. Журавська

«__» _____ 202__ р.

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

**Апаратно-програмний комплекс для навігації
роботів в приміщеннях на основі LiDAR та камер**

Спеціальність «Комп'ютерна інженерія»
123 – КМР.1 – 605.21810517

Студент

_____ О.С.Плохой
підпис

«__» _____ 202__ р.

Керівник канд. техн. наук, доцент

_____ В.Ю.Савінов
підпис

«__» _____ 202__ р.

Миколаїв – 2024

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	3
ВСТУП	4
1. ПРИЗНАЧЕННЯ ТА МЕЖІ ПРОЄКТУ.....	7
1.1 Призначення системи	7
1.2 Погодження, що ухвалені в документації	10
1.3 Межі використання проєкту	13
Висновки до розділу 1	18
2. ЗАГАЛЬНИЙ ОПИС.....	20
2.1 Сфера застосування.....	21
2.2 Характеристики користувачів.....	23
2.3 Загальна структура і склад.....	27
2.4 Загальні обмеження.....	31
Висновки до розділу 2.....	37
3. АПАРАТНО-ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ.....	38
3.1 Вибір апаратної платформи.....	38
3.2 Опис апаратної платформи та компонентів.....	45
3.3 Розробка програмної частини.....	53
3.4 Бібліотека.....	56
Висновки до розділу 3.....	60
4. ЕКСПЕРЕМЕНТАЛЬНІ ДОСЛІДЖЕННЯ. АНАЛІЗ РЕЗУЛЬТАТІВ ОБЧИСЛЕНЬ.....	61
4.1 Виконання налаштувань.....	61
4.2 Побудова графіків навігації.....	63
4.3 Розробка робота.....	67
Висновки до розділу 4.....	69
ВИСНОВКИ.....	70
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	72

Додаток А.....	76
Додаток Б.....	79

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ROS	–	Robot Operating System
ICP	–	Iterative Closest Point
GPS	–	Global Positioning System
TOF	–	Time of light
NDT	–	Normal Distribution Transform
LOAM	–	LIDAR Odometry and Mapping
LLAM	–	LIDAR Localization and Mapping
GNSS	–	GNSS-Inertial Navigation System
NQR	–	Nuclear Quadrupole Resonance
GPR	–	Ground Penetrating Radar
DSI	–	Display Interface

ВСТУП

Сучасний розвиток робототехніки та автономних систем привертає все більшу увагу наукової та індустріальної спільнот. Завдяки поєднанню передових технологій в області датчиків та обробки даних, відкриваються нові перспективи для створення ефективних та надійних систем навігації для роботів у внутрішніх приміщеннях. В даній роботі досліджується проблематика створення програмно-апаратного комплексу, що базується на використанні технологій LiDAR та камер для навігації роботів у внутрішніх приміщеннях.

Передові методи обробки даних, в особливості алгоритми візуального та глибинного аналізу зображень, спільно з точними вимірюваннями, отриманими за допомогою LiDAR, створюють унікальну можливість для побудови точних та ефективних систем навігації. Використання цих технологій може значно підвищити рівень автономності роботів у складних умовах внутрішнього середовища, таких як офіси, магазини або промислові підприємства.

Метою цієї роботи є дослідження та розробка програмно-апаратного комплексу, який забезпечить роботам здатність навігації у внутрішніх приміщеннях з високою точністю та надійністю. Акцент буде зроблено на вивченні та оптимізації алгоритмів обробки даних LiDAR та камер, а також на розробці високоефективних стратегій управління рухом робота на основі отриманих даних.

Ця робота спрямована на вирішення актуальних завдань в галузі робототехніки та має потенціал стати важливим внеском у розвиток автономних систем у внутрішніх умовах.

Актуальність магістерської роботи полягає в постійному розвитку робототехніки та необхідності вдосконалення систем навігації для роботів у внутрішніх приміщеннях

Мета магістерської роботи: Розробка та оптимізація апаратно-програмного комплексу для навігації роботів у внутрішніх приміщеннях на базі LiDAR та камер

Об'єкт дослідження магістерської роботи: Оптимізація процесів обробки даних, а також практичні аспекти використання даного комплексу в різноманітних сферах

Предмет дослідження магістерської роботи: Алгоритми та методи, що становлять основу апаратно-програмного комплексу для навігації роботів у внутрішніх приміщеннях на основі LiDAR та камер.

Для досягнення поставленої мети необхідно вирішити такі **завдання**:

- розробити та реалізувати програмне забезпечення для збору, обробки та аналізу даних, отриманих від LiDAR та камер;
- інтеграція сенсорів LiDAR та камер з апаратною платформою, зокрема Raspberry Pi;
- створення апаратного комплексу, який забезпечить точне виявлення роботами оточуючих об'єктів та перешкод у внутрішньому просторі;
- розробка алгоритмів сканування поверхонь для навігації роботів;
- розробка питань з охорони праці та безпеки життєдіяльності для забезпечення безпечної експлуатації.

Наукова новизна полягає у створенні апаратно-програмного комплексу для навігації роботів у внутрішніх приміщеннях на основі LiDAR та камер, і поки немає патентів на цю технологію в Україні.

Практичне значення отриманих результатів можливе використання розробленого апаратно-програмного комплексу для підвищення ефективності та автономності роботів у внутрішніх приміщеннях, що може бути важливим у різних галузях, включаючи виробництво, логістику, медицину та інші сфери.

Робота пройшла **апробацію** під час XVI міжнародної науково-практичної конференції «Інформаційні технології і автоматизація - 2023» (Одеса, 19–20 жовтня 2023 р.).

Публікації. Основні положення магістерської роботи опубліковані у збірнику матеріалів XVI міжнародної науково-практичної конференції «Інформаційні технології і автоматизація - 2023» [1].

1. ПРИЗНАЧЕННЯ ТА МЕЖІ ПРОЄКТУ

Призначенням проєкту є розробка та впровадження програмно-апаратного комплексу для навігації роботів у внутрішніх приміщеннях на базі технологій LiDAR та камер. Межі проєкту включають в себе розробку високоефективних алгоритмів обробки даних, інтеграцію сенсорів з апаратурою, створення апаратного комплексу для точного виявлення просторового розташування об'єктів та розробку програмного забезпечення для забезпечення безперешкодного руху роботів у внутрішньому середовищі.

Додатково, проєкт передбачає тестування та валідацію розробленого комплексу в реальних умовах внутрішніх приміщень, а також оптимізацію системи з метою досягнення максимальної надійності та продуктивності. Крім того, межі проєкту включають аналіз практичної застосовності отриманих результатів у різних галузях та розробку рекомендацій щодо подальшого вдосконалення та використання програмно-апаратного комплексу.

1.1 Призначення системи

Робота спрямована на розробку високорівневої робототехнічної системи, здатної автономно навігувати та картографувати своє оточення з використанням технології LiDAR. LiDAR (виявлення і вимірювання світла) виступає як потужна технологія сприйняття, яка дозволяє здійснювати точне та детальне двовимірне картографування оточуючого середовища за допомогою випромінювання лазерних променів та вимірювання їх відбиття. Використовуючи LiDAR, ця робота має на меті покращити можливості автономних роботів та розкрити їхній потенціал у різноманітних застосуваннях. Основними цілями цієї роботи є кілька ключових аспектів. По-перше, це включає інтеграцію датчика LiDAR з існуючими апаратними та програмними системами робота, створення безперервної комунікації та обміну даними. Фокусується на розробці алгоритмів та методів для картографування середовища за допомогою даних LiDAR. Це передбачає

використання вимірів на основі лазера для побудови точних і детальних двовимірних карт оточення робота. Реальний час роботи та стійкість є важливими у цій роботі. Система буде оптимізована з метою забезпечення швидкого та реагуючого прийняття рішень на основі даних LiDAR, що дозволить роботу адаптуватися до динамічних середовищ та змінних умов. Вона буде розроблена для роботи з різними сценаріями, такими як різні умови освітлення, різноманітний рельєф та наявність рухомих об'єктів. У кінцевому підсумку успішне завершення цього завдання призведе до розробки автономного робота, який може самостійно досліджувати невідомі середовища, докладно їх картографувати та безпечно та ефективно навігувати в них з використанням технології LiDAR.

Інтеграція LiDAR з автономними роботами пропонує ряд переваг. Вона дозволяє роботам сприймати та розуміти своє оточення з високою точністю та деталізацією. Це, у свою чергу, полегшує безпечну та ефективну навігацію, надаючи реальний час інформацію про перешкоди, відстані та геометричні особливості навколишнього середовища, яка може використовуватися для подальшого аналізу, планування та прийняття рішень.

Основні цілі цієї роботи охоплюють кілька ключових аспектів. По-перше, вона передбачає інтеграцію датчика LiDAR з існуючими апаратними та програмними системами робота, забезпечення безперервної комунікації та обміну даними. По-друге, робота спрямована на розробку алгоритмів та технік для картографування оточення з використанням даних LiDAR. Це передбачає використання вимірів на основі лазера для побудови точних та детальних двовимірних карт навколишнього середовища робота.

Робота надійності та роботи в реальному часі є важливими аспектами цієї роботи. Система буде оптимізована з метою забезпечення швидких та відповідальних прийняття рішень на основі даних LiDAR, що дозволить роботу адаптуватися до динамічних середовищ та змінних умов. Вона буде

розроблена для роботи з різними сценаріями, такими як різні умови освітлення, різноманітний рельєф та наявність рухомих об'єктів.

В остаточному підсумку успішне завершення цієї роботи призведе до розробки автономного робота, який може самостійно досліджувати невідомі середовища, докладно їх картографувати та безпечно та ефективно навігувати в них з використанням технології LiDAR. Застосування такої системи широкі, вони включають автономне дослідження та нагляд, а також виконання завдань у галузях, таких як сільське господарство, будівництво та пошук та рятування. Використовуючи потужність LiDAR, ця робота має на меті розширити можливості автономної робототехніки та відкрити шлях для інноваційних рішень у різних галузях.

Обмежені можливості сприйняття: Традиційні автономні роботи часто покладаються на обмежені датчики, такі як камери або датчики близькості, які мають обмеження у точності сприйняття середовища. Це може призвести до неповної або ненадійної інформації про оточення, що ускладнює навігацію робота та прийняття обґрунтованих рішень. Робота має на меті подолати це обмеження за допомогою інтеграції технології LiDAR, яка надає точне 2D картографування та детальне сприйняття середовища.

Картографування невідомих середовищ: Навігація та картографування невідомих середовищ становлять значні виклики для автономних роботів. Без попереднього знання або наявності попередніх карт, роботи можуть зазнати труднощів у створенні точних репрезентацій свого оточення. Робота має на меті розробити алгоритми та техніки, що використовують дані LiDAR для створення детальних та надійних карт в реальному часі, що дозволяє роботам більш ефективно досліджувати та розуміти невідомі середовища.

Робота в реальному часі та надійність: Автономні роботи, які діють у сценаріях реального світу, потребують прийняття рішень в реальному часі та надійності. Затримки у обробці даних або неточності у сприйнятті можуть призвести до проблем з безпекою або неефективною роботою. Робота має на

меті оптимізувати систему для роботи з великими обсягами даних LiDAR у реальному часі, забезпечуючи швидке та відповідальне прийняття рішень, при цьому забезпечуючи надійність в різних умовах оточення.

Система складається з мобільного робота, який працює на платформі Raspberry PI та обладнаний датчиком LIDAR для сканування приміщень. Датчик LIDAR випромінює лазерні промені та вимірює час польоту відбитих променів, надаючи точні виміри відстані. Raspberry PI відповідає за керування рухами робота та отримання даних від датчика LIDAR. Для бездротового зв'язку між роботом та віддаленим сервером використовується Bluetooth. Робот сканує приміщення, систематично рухаючись та обертаючись, збираючи дані LIDAR під різними кутами. Зібрані дані обробляються, і генерується детальна двовимірна карта приміщення. Дані картографування передаються на віддалений сервер через Python, що дозволяє візуалізацію та аналіз у реальному часі.

1.2 Погодження, що ухвалені в документації

З новими досягненнями в технології самостійної мобільності для мобільних роботів, було запропоновано різноманітні сервісні роботи для допомоги різним галузям, включаючи виробництво, складське господарство, медичний сектор, сільське господарство та ресторани [2]. Сприйняття та розуміння штучного інтелекту постійно покращуються з появою та використанням продуктів штучного інтелекту, таких як автоматизовані роботизовані сутності [3].

Крім того, дефіцит персоналу, постійний попит на поліпшення продукції та зміни в секторі обслуговування змусили галузь ресторанного бізнесу прийняти роботів. Для підвищення продуктивності праці в ресторанах було впроваджено сервісних роботів, таких як автоматизована система керування транспортних засобів (AGV), в японських ресторанах [4,5]. Системи AGV перетворилися на автономні мобільні роботи (AMR), щоб забезпечити більшу

функціональність та покращену ефективність завдяки розвитку технологій штучного інтелекту, потужних процесорів на борту, всюдисущих датчиків та технологій одночасного визначення місця та планування. Традиційні недорогі сервісні роботи рухаються патрулюючи лінію. Радарно-засновані автономні навігаційні роботи широко вивчалися, але їх вартість складає навіть приблизно \$17 000 [6]. Використання штучного інтелекту та роботів у ресторанах все ще перебуває на етапі початкового розвитку, менеджери шукають допомоги у використанні сучасних технологій для поліпшення обслуговування [7]. COVID-19 змінив бізнес, оскільки безконтактна парадигма може забезпечити комфорт клієнтам та персоналу під час відпочинку та роботи [8].

Багато менеджерів обслуговування розглядають впровадження сервісних AMR для забезпечення безперервного обслуговування та полегшення дефіциту робочої сили. Чонг та ін. [5] створили прототип рухомого робота-офіціанта на колесах Меканум для тестування в ресторані. Ю та ін. [9] створили робота для обслуговування в ресторані, який може приймати замовлення, видаляти та доставляти страви. У відповідь на проблеми, пов'язані з COVID-19, у Китаї та Сполучених Штатах з'явилася зростаюча кількість ресторанів, які впроваджують роботів [9]. Янг та ін. [10] запропонували систему автентифікації роботизованої доставки, яка складається з клієнта, сервера та робота. Для задоволення вимог різних закладів харчування було розроблено багато роботів для різних використань, таких як гарячий казан, китайська кухня та кава [11]. Однак різні університети працюють над розробкою універсальної, злагодженої та практичної системи роботів, яка може бути застосована в різних сценаріях застосування в ресторанах. Одна з найефективніших реалізацій обробки роботами тарілок була здійснена на Олімпійських іграх 2022 року в Пекіні. Цей обладнаний приклад очікується збільшити попит на ресторанні роботи ще більше. Однак вартість роботів є важливим чинником, який ресторани враховують перед впровадженням технологій. Сервісні роботи можуть бути побудовані на

загальній рамці замість зазначених вище методик розвитку, щоб мінімізувати витрати на розробку. Чітта та ін. [12] запропонували загальну та просту систему керування для впровадження та керування контролерами роботів з метою підвищення як реального часу, так і спільного використання контролера. [13] Ладос та ін. представили систему керування роботом (ROS) для розробки та тестування функцій автономного керування. Ця ROS може працювати з широким спектром роботизованих систем і є низько-вартісною, базовою, високо варіативною та переконфігуровувальною системою. Чиваров та ін. [14] досліджували взаємодію та співпрацю між роботами та пристроями Інтернету речей (IoT) для створення вартісного автономного гуманоїдного робота для допомоги людям у повсякденних справах. Ткачик та ін. [15] показали прототип модульної транспортабельної робототехнічної платформи, яку можна використовувати як для досліджень, так і для освіти. Нох та ін. [16] представили систему автономної навігації на базі відкритих джерел для внутрішніх мобільних роботів, досягнувши трьох основних модулів: картографування, локалізація та планування. Встановлення відповідного пристосування до різних сервісних завдань залежно від конструкції робота, характеристик клієнта та особливостей завдання обслуговування [18].

Зі зростанням технологій та робототехніки мобільні роботи стали однією з найактивніших галузей, які є не лише високо гнучкими, але й широко використовуваними у військовій справі, сільському господарстві, промисловості, сфері обслуговування та інших галузях. Мобільні роботи можуть замінити людей у невідомих середовищах для виконання складних завдань, зменшити робоче навантаження персоналу та забезпечити їхню безпеку. Здатність робота точно визначити своє місцезнаходження та ідентифікувати оточення під час побудови точної карти є передумовою для виконання цих завдань. Робот створює карту свого оточення на основі інформації, отриманої від датчиків, які він несе, а також оцінює позицію робота на основі деяких створених карт, але існують проблеми, такі як низька

точність та неповне виявлення перешкод при побудові карти за допомогою одного датчика, і технологія фузії даних від кількох датчиків стає ключем для вирішення цієї проблеми.

Одноканальний сенсор LiDAR може сканувати та виявляти перешкоди в площині під час виконання оновлень в реальному часі, що має переваги високої точності та швидкого побудови карти, але одноканальний LiDAR може сканувати лише на певній горизонтальній площині і не може сканувати та виявляти перешкоди вище або нижче горизонтальної площини; датчик глибинної камери збирає безпосередньо інформацію про глибину пікселів, що є вирішення проблем одночасного витоку інформації від одного датчика, низької точності виявлення та вразливості до зовнішніх факторів довкілля, у цій роботі запропонована схема картографування мобільного робота з використанням алгоритму для злиття одноканального LiDAR та датчика глибини, щоб досягти ефекту, та побудувати експериментальну платформу робота для перевірки того, що карта, побудована за допомогою злиття даних від кількох датчиків, є точнішою та повнішою, ефективно уникати проблем, що виникають при картографуванні за допомогою одного датчика.

1.3 Межі використання проєкту

Метою цього дослідження є вивчення можливостей автономного мобільного робота лише на основі технології LIDAR та його здатності сканувати приміщення в тактичному середовищі. Більшість представників промисловості та академії в даний момент зосереджені на розробці високоточних, детальних, великомасштабних та напіваавтономних можливостей LLAM. У свою чергу, це дослідження спрямоване на розробку швидкої, ефективною, обмеженої за масштабом та повністю автономної можливості. Це включає вивчення не лише методів вирівнювання сканів LIDAR, планування маршруту, уникнення перешкод та дослідження фронтального дослідження, а також їх інтеграцію. Також необхідно вивчити

роздільну здатність карти, яка потрібна для тактичного планування, та найбільш ефективний метод візуального відображення тривимірного внутрішнього простору приміщень.

Метою є побудова робота, який зможе успішно картографувати внутрішні приміщення автономно та в реальному часі. Для цього проекту використовувався наземний транспортний засіб для експериментів. Однак оскільки майбутнє цієї технології, ймовірно, пов'язане з повітряним роботом, таким як безпілотною з обладнанням LIDAR, мета включала розробку моделі з шістьма ступенями вільності, яка може легко адаптуватися до повітряного робота.

Багато тривимірних моделей створюються за допомогою процесів тривимірного сканування, таких як LIDAR; однак у багатьох ситуаціях можливо зібрати лише часткове сканування з будь-якої однієї перспективи. Для створення повної тривимірної моделі часткові сканування потрібно об'єднати в одне сканування. Цей процес називається реєстрацією і застосовується як до хмари точок, так і до меш-базованих тривимірних моделей. Зазвичай він включає три етапи. По-перше, знаходять відповідності між точками в послідовних сканах. Далі, на основі цих відповідей точок обчислюється трансформація, яка мінімізує відстань між відповідними точками. Нарешті, трансформовані хмари точок об'єднуються в одну хмару точок, що представляє повну модель. Існує багато методів реєстрації хмар точок, і для кожного методу було розроблено багато варіацій, щоб вони відповідали певним застосуванням. У цій дисертації ми розглянемо три загальні методи реєстрації хмар точок: ітеративне найближче співставлення (ICP), згуртоване співставлення точок (CPD) та нормальне розподілення трансформації (NDT).

Реєстрація повертає або двовимірні, або тривимірні трансформації. У двовимірній трансформації використовуються три параметри: трансляційні

параметри t_x і t_y та кутовий параметр повороту ϕ . Всі три параметри зберігаються у векторі p як

$$p = [t_x t_y \phi] \quad (1)$$

Двовимірне перетворення задано

$$T(p, x) = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} x + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (2)$$

Трансформації можуть бути жорсткими, нежорсткими або афінними. Жорстка трансформація зберігає форму та розмір об'єктів у сцені, застосовуючи ту саму трансформацію до всіх точок. Афінна трансформація дозволяє викривлення, зміни масштабу, а також трансляцію та обертання точок відносно жорсткої трансформації. Має короткий діапазон і високу швидкість знімання. При низьких швидкостях, таких як у цій роботі, в кожному кадрі мало рухового розмиву, і тому переважно використовуються жорсткі трансформації. Нежорсткі трансформації дозволяють змінювати форму об'єктів, точки перетворюються по-різному за допомогою поля зміщення. Нежорсткі трансформації не використовувалися в цьому алгоритмі LLAM.

Сприйняття навколишнього середовища в основному залежить від різноманітних датчиків для досягнення картографування та локалізації, і в багатьох випадках в 3D дослідках SLAM з'явилися чудові результати. Відомий алгоритм пропонує новий спосіб виділення ознак хмари точок, одночасно комбінуючи високочастотну низьку точність одометрії та низькочастотну високу точність одометрії, збалансованою точністю та ефективністю. Lego-loam [покращує точність та ефективність зіставлення шляхом сегментації хмари точок перед виділенням ознак і додає виявлення замикання для виконання завдань картографування землі. Використовується метод відповідання ключового кадру з презуванням вікна та об'єднує IMU, GPS та лідарні фактори для побудови одометрії у щільно зв'язаному способі. Поєднує візуально-інерціальну систему та лідарну інерціальну систему у щільно

зв'язаному способі та досягає стійкої оцінки стану та побудови карти з високою точністю.

Картографування середовища може бути використане для вузьких печер, низькоокисневих підземних проходів, невідомих середовищ, до яких люди не можуть дістатися. Крім того, велика кількість виробничих робіт, гірничих робіт та робіт у промисловості, які все ще виконуються вручну людьми, дуже схильні до виникнення аварій. Тому потрібен пристрій, який може автоматично створювати карти та навігувати, щоб робот міг замінити людських робітників.

Для створення карти оточуючого середовища потрібні датчики з високим рівнем точності, які можуть досягати певної точки на великій відстані. LIDAR є однією з технологій дистанційного зондування, яка має потенціал допомогти (картографувати, моніторити та оцінювати місця просторових елементів) у багатьох галузях / застосуваннях, пов'язаних із наданням геопросторових баз даних. Завдяки щільності даних та високій точності, датчик LIDAR дуже підходить для впровадження в картографування та навігацію роботів. Багато дослідників повідомляли про використання лазерних дальномерів у своїх системах, вони використовують датчик RP Lidar для створення картографування з мобільним роботом. Rusdinar використовував лазерні дальномери для виправлення похибок положень мобільного робота за допомогою фільтру частинок та методу внутрішньої локалізації на базі зорового спостереження для автономних транспортних засобів [4]. А. Rubinstein і Т. Erez створили робота під назвою LiTank, який є роботом, призначеним для картографування тунелів за допомогою лідарних датчиків.

Автоматична навігація робота також потрібна в умовах картографування, щоб робот міг правильно працювати. Дані з LIDAR можуть бути відображені і використані роботами для визначення місцеположення в середовищі. Потім дані з картографування використовуються роботами для

навігації та планування руху. Крім того, дані з LIDAR також використовуються для оцінки параметрів робота.

Позиціонування, необхідне під час картографування оточуючого середовища. Локалізація та планування маршруту є фундаментальними для проблеми роботизованої навігації. Для досягнення успішної навігації робот повинен мати можливість локалізувати себе та створювати одночасну карту середовища

Основною метою цього дослідження є створення обертового пристрою для датчиків LIDAR з метою сканування оточуючого середовища з діапазоном 360 градусів за допомогою лише одного датчика. Потім обертовий пристрій перевіряється за допомогою створення картографування на основі отриманих даних. Потім результати двох сканувань порівнюються, щоб передбачити рух робота. Останній обертовий пристрій перевіряється шляхом створення навігаційного шляху на попередньо створеній карті.

Тоді очікується, що пристрій може бути розвинутий таким чином, щоб його можна було використовувати на роботі для проведення картографування та навігації. Більшість пристроїв для обертових датчиків LIDAR на ринку настільки дорогі, що дослідники очікують розробки кращих та дешевших сенсорних пристроїв для дослідження картографування та навігації роботів.

Для здійснення картографування та навігації потрібні кілька етапів збору даних з датчика, оцінювання позиції робота, створення карти та навігація. Датчик повертається на 360 градусів за допомогою крокового двигуна. Raspberry Pi отримує дані з датчика LiDAR TFmini-S за допомогою комунікації PWM. Дані з датчика отримуються на кожному кроці обертання крокового двигуна. Дані з датчика передаються за допомогою Bluetooth, а дані отримуються за допомогою послідовної комунікації через програмне забезпечення Python. Після отримання даних Python перетворює дані для подальшої обробки у карту та навігацію. Кожний рух крокового двигуна

контролюється командою з Python, яка передається за допомогою Bluetooth за допомогою послідовної комунікації.

Сітка зайнятості відображає багатовимірну карту навколишнього середовища (зазвичай 2D або 3D) у комірку, де кожна комірка зберігає своє ймовірнісне значення стану. Інформацію про середовище можна отримати з даних датчиків відстані, камери та датчиків ударів, які зазвичай використовуються для визначення перешкод у середовищі робота. Існують 2 представлення сітки зайнятості: бінарна сітка зайнятості та ймовірнісна сітка зайнятості.

Бінарна сітка зайнятості використовує значення True для представлення зайнятого робочого простору (перешкоди) та значення False для представлення вільного робочого простору. Ця сітка показує, де розташовані перешкоди та чи може рухатися робот через цей простір. Тоді як ймовірнісна сітка зайнятості використовує ймовірнісні значення для створення більш деталізованих картографічних представлень. Це представлення є переважним методом використання Сітки зайнятості. Цю сітку зазвичай просто називають житловою. Кожна комірка в житловій сітці має значення, яке представляє ймовірність зайнятості комірки. Значення, близьке до 1, представляє високу впевненість у тому, що комірка містить перешкоди. Значення, близьке до 0, представляє впевненість у тому, що комірка не зайнята та є безперешодною.

Висновки до розділу 1

У першому розділі було здійснено аналіз технологічних аспектів навігації роботів в приміщеннях на основі LiDAR та камери. Оглянуті можливості та переваги використання LiDAR, що дозволяє здійснювати точне та детальне картографування навколишнього середовища за допомогою лазерних променів. Також розглянуто важливість використання камери для додаткового збору інформації про навколишнє середовище.

Основною метою цього розділу було встановлення технічної можливості створення програмно-апаратного комплексу, що базується на LiDAR та камері, для навігації роботів у внутрішніх приміщеннях. Аналізуючи наявні технології та методи, було визначено, що використання LiDAR та камери в парі може значно покращити точність та надійність навігації роботів в приміщеннях.

Також у розділі було встановлено важливі завдання, які необхідно вирішити для досягнення поставленої мети. Ці завдання включають інтеграцію LiDAR та камери з апаратним і програмним забезпеченням робота, розробку алгоритмів картографування та навігації, а також оптимізацію системи для роботи в реальному часі та в різних умовах середовища.

Загалом, результати першого розділу підтверджують перспективність використання LiDAR та камери для навігації роботів у внутрішніх приміщеннях. На основі цих висновків буде розроблено подальші кроки для втілення концепції програмно-апаратного комплексу та проведення досліджень для його вдосконалення

2. ЗАГАЛЬНИЙ ОПИС

Використання виявлення світла та визначення дальності (LIDAR) для картографування та уникнення перешкод в автономних наземних транспортних засобах є активною сферою досліджень. Точність, точність, оперативність і постійне зниження вартості LIDAR робить його привабливим як рішення для створення детальних, високоточних карт. Використання LIDAR в автономній локалізації є новою, менш розвиненою сферою дослідження, але багатообіцяюче для майбутнього автономних транспортних засобів.

Останнім часом комерційно доступні системи LIDAR значно знизили свою вартість і підвищили роздільну здатність. Наприклад, компанія Velodyne випускає комерційно доступні LIDAR, які можуть створювати до 4,8 мільйонів тривимірних точок за секунду з дальністю повернення до 245 метрів. Цей рівень роздільної здатності може створювати дуже деталізовані тривимірні моделі; однак обробка комп'ютером такого великого хмари точок може створювати значні проблеми. Величезний обсяг даних, що генерується системами 3D LIDAR, створює затор, що робить локалізацію та картографування в реальному часі складними. Локалізація та картографування на основі LIDAR має широкий спектр застосувань сьогодні. У більшості застосувань LIDAR використовується в основному для побудови карти. Локалізація на основі LIDAR зазвичай поєднується з іншими сенсорами для оцінки положення і орієнтації, або позиції, робота. Часто локалізація та картографування на основі LIDAR виконується у післязборочній обробці. Під час збору даних LIDAR робот або безпілотний транспортний засіб локалізується за допомогою набору інших сенсорів. Після того, як він виконав свою призначену траєкторію, дані обробляються для створення тривимірної карти. Ця модель працює добре для багатьох промислових застосувань; однак вона потребує більших, важчих, більш складних роботів, які не дуже підходять для військових розвідувальних цілей на землі.

2.1 Сфера застосування

Для реалізації автономної навігації роботи потрібно побудувати модель оточуючого середовища, спланувати глобальний маршрут, розрахувати місцеві інструкції руху і, нарешті, дійти до цільової позиції за допомогою виконання керування. Навігаційні карти відіграють важливу роль у сфері мобільних роботів. SLAM на основі LiDAR може використовуватися для отримання хмари точок з багатою інформацією. Цей вид карти може здійснити тривимірну (3D) реконструкцію навколишнього середовища, але не надає роботу інформації, де пройти, тому її не можна безпосередньо використовувати для навігації. Тому, зазвичай потрібно розділити область, яку можна пройти на навколишній карті, тобто точно ідентифікувати перешкоди і області, які важко пройти, щоб робот міг повністю розуміти інформацію про навколишнє середовище і проводити планування маршрутів. Поточні навігаційні карти можуть бути розділені на карті ознак, топологічні карти та сіткові карти. Карті ознак будуються згідно з геометричною інформацією об'єктів ознак у сцені, що широко використовуються в закритих приміщеннях, оскільки такий тип карт лише зберігає геометричну інформацію про ознаки, а його обсяг даних невеликий, часто використовується в ранніх алгоритмах SLAM. Топологічні карти використовують концепцію вузла плюс напрямок, щоб представити відносні взаємини між місцями ознак, що має високу абстракцію і не може правильно виражати форму, об'єм та іншу інформацію про об'єкти, тому вона не підходить для високоточних складних сцен. Сіткові карти використовуються для представлення стану цієї позиції, і значення, що зберігається кожною коміркою, представляє, чи зайнята ця позиція. Вони краще виражають форму та напрям об'єктів у сцені та широко використовуються в області навігації.

Залежно від типу сенсора, SLAM може бути розділений на візуальний SLAM і LIDAR SLAM. На відміну від візуальних сенсорів, які легко піддаються впливу освітлення, погоди, змін точки зору камери та інших

факторів, LIDAR може безпосередньо отримувати інформацію про відстань і є більш стійким до змін у середовищі, що робить систему LIDAR SLAM широко використовуваною на практиці. Початково мобільні роботи широко використовувалися в закритих приміщеннях. Завдяки точності одноканального вимірювання відстані LIDAR та низькому обсягу даних, технологія двовимірного SLAM на основі цього стає зрілим. Однак із зростанням попиту на відкриті сцени одноканальний LIDAR став важким для використання в більших відкритих середовищах.

Багатоканальний LIDAR має сильнішу здатність описувати оточення, але водночас він супроводжується проблемами високої ціни та великої кількості даних. З масовим виробництвом та популяризацією багатоканального LIDAR, зниженням споживання енергії вбудованими процесорами та збільшенням обчислювальної потужності, технологія 3D SLAM на основі багатоканального LIDAR стрімко розвивається. У системах SLAM оцінка орієнтації та траєкторії робота часто супроводжується необхідним дрейфом, що призводить до невдалої локалізації. Рішення SLAM, що використовує лише сенсори LIDAR, досягає оцінки положення в реальному часі за допомогою збігу послідовних ключових кадрів, але похибка накопичується зі збільшенням пройденої відстані. Тому виявлення замкненого циклу є вирішальним для коригування помилок дрейфу та побудови глобально послідовної карти. Наразі більшість систем SLAM на основі LIDAR передбачають, що середовище повністю статичне. Однак у сценах автоматичного водіння, обслуговування в приміщеннях та інших сценах, широко використовуваних у системах SLAM, динамічні об'єкти, такі як транспортні засоби та пішоходи, неминуче будуть заважати. Оскільки 3D хмару точок побудовано послідовним накопиченням кадрів сканування, динамічні об'єкти призводять до появи артефактів на карті середовища, і ці артефакти будуть неправильно ідентифікуватися як перешкоди у наступних застосуваннях, знижуючи навігаційну та позиційну продуктивність роботів.

Тому важливо фільтрувати динамічні об'єкти у навігації у динамічному середовищі. Залежно від різних часів виконання, алгоритми фільтрації динамічних об'єктів можуть бути розділені на дві категорії: перше - фільтрація динамічних хмар точок в реальному часі в процесі побудови малюнка. Такі методи потребують забезпечення реального часу та можуть порівнювати лише сусідні кадри за часом або простором для виявлення динамічних частин. Друге - спочатку будується повна карта, а потім фільтрується динамічна хмара точок. Цей метод може реалізувати більш точне та комплексне фільтрування динамічних частин, використовуючи інформацію всіх кадрів у всьому процесі SLAM, і він має низькі вимоги до реального часу, тому поточні дослідження більше нахиляються до останнього виду алгоритму.

Систему картографування на основі LiDAR, яка використовує багатоканальні сенсори LiDAR як вхідні дані і може генерувати карти, придатні для навігації роботів в динамічних середовищах. Алгоритм SLAM на основі LiDAR з використанням виявлення замикання петель ISC. Використовується одночасно геометричну та інтенсивну інформацію з хмари точок LiDAR для покращення точності позиціонування алгоритму і проводимо компенсацію спотворень для хмари точок у два етапи, щоб забезпечити швидкість роботи алгоритму.

2.2 Характеристики користувачів

Для проведення дослідження використовувалися такі компоненти обладнання: внутрішній двоколісний мобільний робот, 2D система LIDAR, 3D камера з часом прольоту (ToF), вбудований комп'ютер та ноутбук або настільний комп'ютер. Цей робот вже використовувався у попередніх дослідженнях у цій роботі він був модифікований для встановлення 3D камери ToF.

У робота є кілька ключових компонентів, які сприяють його рухові та взаємодії з оточенням. Перш за все, ми маємо мотори, які приводять колеса

робота. Ці мотори дозволяють роботу переміщатися вперед, назад, повертатися та обходити перешкоди. Крім того, у робота є сенсори, які допомагають визначати його положення та навколишнє середовище. Наприклад, ультразвукові сенсори допомагають визначити відстань до об'єктів, інфрачервоні сенсори виявляють тепло, а датчики світла визначають рівень освітлення. Ці сенсори забезпечують роботу інформацією про оточуючий світ, що дозволяє йому приймати рішення щодо навігації та взаємодії з оточенням.

До того ж, у робота є вбудований або міні-комп'ютер, який відповідає за керування всією цією апаратною частиною. Він отримує дані від сенсорів, обробляє їх і видає команди моторам для руху. Крім того, цей мікроконтролер може мати підключення до бездротової мережі, що дозволяє йому комунікувати з іншими пристроями або навіть з хмарними сервісами для обробки даних або отримання інструкцій.

Окрім цього, робот оснащений камерами для візуального сприйняття оточення або навіть розпізнавання обличчя та об'єктів. Це дозволяє роботу більш ефективно взаємодіяти з людьми та іншими предметами навколо нього.

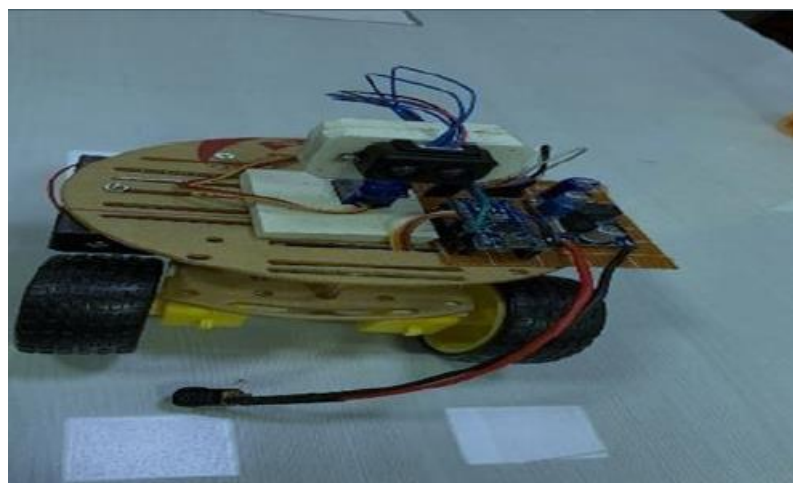


Рисунок 2.1 – Складова для взаємодії з LiDAR TF mini

TFmini LiDAR Sensor є інноваційним проектом, що базується на популярному модулі i, пропонуючи більші можливості та поліпшення для вимірювання відстаней у мобільних роботах та інших пристроях. Цей продукт

відзначається не лише своєю низькою вартістю та компактними розмірами, але й високою точністю, надійністю та швидкістю вимірювання відстаней.

Однією з ключових переваг TFmini є його висока точність вимірювання відстаней, що дозволяє мобільним роботам точно визначати своє положення та відстань до об'єктів у навколишньому середовищі. Це надає можливість роботам ефективно навігувати навіть у складних умовах, таких як яскраве сонячне світло, різні температури та різна поверхнева відбиваюча здатність.

Крім того, TFmini має низьке споживання енергії, що робить його ідеальним вибором для мобільних роботів, які працюють на акумуляторах або в обмежених умовах живлення. Це дозволяє роботам працювати тривалий час без необхідності частої зарядки або заміни батарей. TFmini підтримує різні інтерфейси зв'язку, включаючи UART та I2C, що дозволяє йому легко інтегруватися з різними контролерами та системами управління. Це забезпечує максимальну гнучкість у використанні і розширює його застосування у широкому спектрі мобільних роботів та автономних систем.

TFmini LiDAR Sensor відкриває нові можливості для розвитку та вдосконалення мобільних роботів, забезпечуючи їм надійні та точні засоби навігації та взаємодії з оточенням. Його висока якість та універсальність роблять його ідеальним вибором для широкого спектру застосувань у робототехніці та автономних системах.



Рисунок 2.2 – TF mini LiDAR.

TFmini LiDAR Sensor - це ідеальне рішення для навігації мобільних роботів у внутрішніх приміщеннях завдяки його унікальним характеристикам та функціональності.

Однією з ключових переваг TFmini є його висока точність та швидкість вимірювання відстаней. Це дозволяє роботам точно визначати своє розташування в приміщенні та виявляти перешкоди на своєму шляху. Завдяки цьому, мобільні роботи можуть ефективно уникати зіткнень та навігувати в окремих приміщеннях з високою точністю та надійністю. Також важливою особливістю TFmini є його низьке споживання енергії, що дозволяє тривалий час автономної роботи без необхідності зарядки або заміни батарей. Це особливо корисно для мобільних роботів, які працюють у великих приміщеннях або на значній відстані від джерела живлення, що дозволяє легко інтегрувати його з різними системами управління та контролерами. Це робить його універсальним рішенням для різних типів мобільних роботів та автономних систем, які вимагають точної навігації у внутрішніх приміщеннях.

TFmini LiDAR Sensor допомагає мобільним роботам ефективно навігувати в приміщеннях, забезпечуючи їм високу точність та надійність у вимірюванні відстаней, низьке споживання енергії та легку інтеграцію з іншими компонентами системи управління.

2.3 Загальна структура і склад

Роботний операційний стандарт (ROS) є ключовим фреймворком у робототехніці, який надає роботам потужні інструменти для розвитку, управління та взаємодії з оточенням. Він широко використовується у різних типах мобільних роботів, включаючи ті, що базуються на технології LIDAR.

ROS забезпечує інтеграцію різноманітних сенсорів та компонентів обладнання, включаючи модулі LIDAR, такі як TFmini. Він дозволяє роботам отримувати дані від цих сенсорів, обробляти їх та приймати рішення на основі отриманих відомостей. Крім того, ROS забезпечує можливість візуалізації даних та результатів роботи за допомогою інструментів, таких як Royale Viewer Point Cloud Visualization Software, що допомагає розробникам та інженерам аналізувати та візуалізувати дані з LIDAR у зручному форматі.

Завдяки інтеграції з ROS, роботи можуть ефективно навігувати в приміщеннях, використовуючи дані з LIDAR для створення 3D моделей оточення, визначення місцеположення та планування маршруту. Це дозволяє їм уникати перешкод, знаходити оптимальний шлях і взаємодіяти з оточенням у реальному часі. Також, ROS надає широкий набір інструментів та бібліотек, що сприяють розробці різноманітних функціональностей для мобільних роботів, від планування шляху до взаємодії зі зовнішніми сервісами аналізу даних чи обробки відеоінформації. Таким чином, ROS створює потужний інструментарій для розробки та впровадження мобільних роботів, що базуються на технології LIDAR, забезпечуючи їм потужні можливості навігації та взаємодії з навколишнім середовищем.

Перш ніж ROS став широко використовуваним, виробники та дослідники розробляли власні робототехнічні системи, тому знання про розробку не було узагальненим, що призвело до високих витрат на розробку кожного окремо. Тепер ви можете використовувати ROS та скористатися ним, щоб прискорити розробку високорозвинених робототехнічних систем.

ROS використовує підхід "розподіленої обробки", щоб сприяти спільній розробці програмного забезпечення для роботів на всесвітньому рівні. Одна компанія може мати спеціалістів у картографуванні внутрішніх просторів на світовому рівні. Інша лабораторія може мати фахівців з використання карт для навігації, а ще інша група може навчити модель комп'ютерного зору, яка добре працює для розпізнавання малих об'єктів. ROS розроблено спеціально для співпраці груп, подібних до цих, для спільної роботи та побудови на основі робіт одне на одному.

ROS надає багато готових до використання пакетів, а його бібліотеки забезпечують повне середовище розробки для промислових робототехнічних систем, з програмами, розробленими по всьому світу, та підтримкою для кількох мов програмування, таких як C++, Python та LISP. ROS також пропонує потужні засоби відлагодження, можливості реєстрації та аналізу даних, а також відкрите 3D-симулятор робототехніки під назвою Gazebo, для розробки різних промислових роботів. ROS має децентралізовану архітектуру, яка реалізує розподілену обробку для збільшення повторного використання програмного забезпечення. Вона працює як одна велика система, де процеси, які називаються "вузлами", відправляють та отримують результати операцій. Процеси в ROS утворюють мережу типу рівень до рівня, яку можна представити у вигляді обчислювального графа.

Royale, розробник технологій для вимірювання глибини та створення тривимірних моделей, не лише постачає передове обладнання, але й забезпечує комплексні рішення для аналізу та візуалізації отриманих даних. Одним із ключових інструментів у цьому пакеті є їхній власний інструмент візуалізації, побудований на основі Royale API.

Цей інструмент не тільки простий у використанні, але й дуже потужний. Він автоматично встановлюється разом зі стандартним пакетом установки Royale, що робить його доступним для всіх користувачів. Головна його функція - відображення даних глибини, отриманих з будь-якого каліброваного

лідара. Такий підхід дозволяє оперативно та зручно аналізувати та візуалізувати інформацію про оточуюче середовище, що є надзвичайно корисним для робототехнічних досліджень, навігації та інших областей, пов'язаних з аналізом глибини.

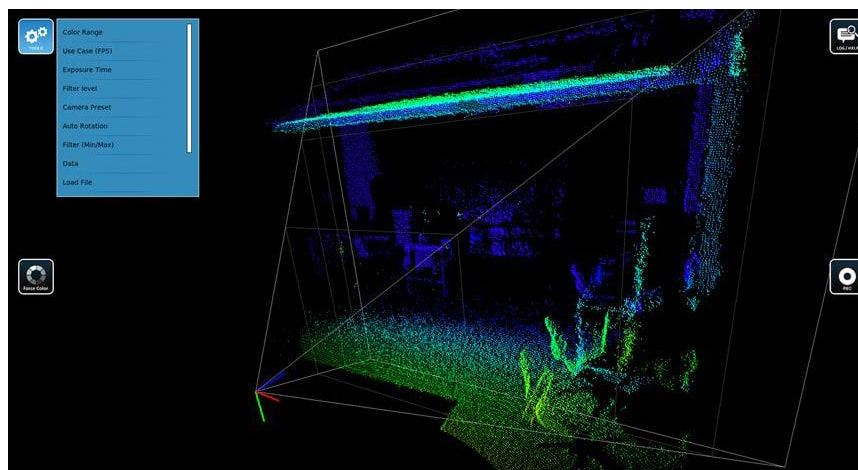


Рисунок 2.3 - Візуалізація хмари точок Royale Viewer

Цей інструмент візуалізації глибинних даних, розроблений компанією Royale, відіграє ключову роль у покращенні навігаційних можливостей роботів. Він забезпечує детальне візуальне відображення тривимірної структури навколишнього середовища, яке робот може використовувати для ефективної навігації та уникнення перешкод.

По-перше, цей інструмент дозволяє роботам отримувати точні дані про глибину навколишнього середовища, що допомагає їм уникати перешкод та об'єктів, знаходячись в їхній близькості. Використовуючи ці дані, робот може розрахувати оптимальний шлях для досягнення мети, уникнути потенційно небезпечних місць та забезпечити безпечний рух у будь-яких умовах.

По-друге, інструмент візуалізації надає можливість роботам побудувати детальну тривимірну модель оточуючого середовища, включаючи структури, перешкоди, об'єкти та їхні розміри. Ця модель може бути використана для розробки точних карт, що допомагають роботу визначити своє місцезнаходження та орієнтацію у просторі, а також планувати оптимальний маршрут для досягнення цілей.

По-третє, інструмент візуалізації дозволяє роботам в реальному часі відстежувати зміни в навколишньому середовищі, такі як рух об'єктів або зміни у конфігурації простору. Це дозволяє роботам адаптуватися до змінних умов та ефективно реагувати на непередбачені ситуації під час навігації, інструмент візуалізації глибинних даних від Royale є необхідною складовою для покращення навігаційних можливостей роботів, забезпечуючи їм точну інформацію про навколишнє середовище та допомагаючи їм безпечно та ефективно переміщатися в різних умовах.

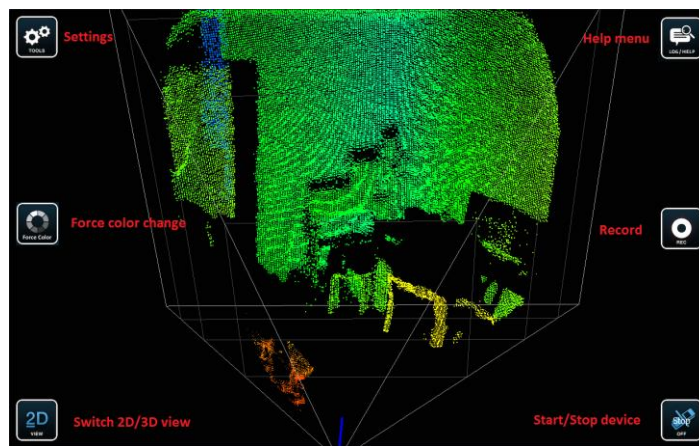


Рисунок 2.4 – Програмне середовище Royale Viewer.

Взаємодіє з лідаром та допомагає навігації роботів в приміщеннях через використання вбудованих функцій та опцій, спрямованих на оптимізацію процесу навігації:

- інтеграція з лідаром: Картинка з меню може включати опції для підключення та налаштування лідара. Користувач може вибрати необхідні параметри, такі як зони сканування, частоту оновлення даних, розмір області сканування тощо. Це дозволяє роботу отримувати актуальні дані про середовище в реальному часі і використовувати їх для навігації;
- обробка даних лідара: може містити опції для обробки даних, отриманих від лідара. Це включає в себе фільтрацію шуму, виявлення перешкод та побудову карти середовища. Після

- обробки ці дані можуть використовуватися для побудови маршрутів та планування переміщень робота в приміщенні;
- використання глибинних даних: може допомагати роботам використовувати глибинні дані, отримані від лідара, для розрахунку оптимального маршруту та уникнення перешкод. Це дозволяє роботу ефективно маневрувати в просторі, уникати перешкод та досягати своєї мети з найменшими зусиллями;
 - відслідковування змін у середовищі: може надавати можливість відслідковувати зміни в навколишньому середовищі, включаючи рух об'єктів або зміни у конфігурації простору. Це дозволяє роботам адаптуватися до змінених умов та ефективно реагувати на непередбачені ситуації.

Взаємодіє з лідаром, надаючи користувачу можливість налаштувати його параметри та обробляти отримані дані, щоб ефективно навігувати робота в приміщенні. Це допомагає роботам уникати перешкод, планувати оптимальні маршрути та забезпечувати безпечну та ефективну навігацію у будь-яких умовах.

2.4 Загальні обмеження

SLAM - це процес, за допомогою якого робот здатний створювати карту навколишнього середовища та одночасно визначати своє місцезнаходження в цьому середовищі. Цей процес вимагає використання різних типів сенсорів, таких як лідар, камери або акселерометри, для збору інформації про оточуюче середовище та рух робота.

У SLAM існують різні підходи, і один з них - LIDAR-базований SLAM. В цьому методі робот здійснює послідовне сканування оточуючого середовища за допомогою лазерних сканерів, які можуть бути двовимірними або тривимірними. Отримані скани, часто називаються "хмарами точок", потім порівнюються між собою у процесі реєстрації. Це порівняння дозволяє

визначити геометричні трансформації між сканами, які використовуються для визначення руху робота в просторі.

У процесі SLAM нові отримані скани трансформуються за рахунок попередніх реєстрацій та додаються до вже існуючої карти середовища. Цей процес дозволяє роботу побудувати докладну карту свого навколишнього середовища та одночасно визначати своє місцезнаходження на цій карті.

Робот використовує кілька відлікових систем для локалізації та картографування, які забезпечують точність і простоту визначення його положення та оточуючого середовища. Деякі з цих відлікових систем є статичними, наприклад, система WORLD, яка визначається позицією обертового центру робота при ініціалізації. Інші відлікові системи, такі як система ROBOT, є рухомими та змінюються разом із рухом робота. Основна фіксована відлікова система - це відлікова система WORLD, яка визначає абсолютне положення робота в середовищі. Вона встановлюється при ініціалізації робота і має праворучну декартову координатну систему з напрямком X вперед від обертового центру робота.

Основна рухома відлікова система - це відлікова система ROBOT, яка пов'язана з самим роботом і рухається разом з ним. Вона також має праворучну декартову координатну систему, з початком у центрі обертової вісі робота (вісь Z) та напрямком X вперед. Ця відлікова система дозволяє роботу визначати своє власне положення та орієнтацію в середовищі під час руху.

Коли робот ініціалізується, відлікові системи ROBOT та WORLD співпадають, поки робот не почне рухатися. У цій роботі розглядалися шість ступенів свободи Six DoF та три ступені свободи Three DoF відлікових систем ROBOT. Протягом експериментів початкове положення відлікової системи ROBOT переносилося та оберталося лише в площині X-Y, оскільки P3-DX - це наземний мобільний робот, який не може подолати сходи, і внутрішні простори для тестування були плоскими без нахилів. Фактично, всі експерименти мали три ступені свободи. Проте під час цієї роботи, коли

відлікова система ROBOT позначалася як три ступені свободи, це означало, що вона була налаштована в програмному забезпеченні таким чином, що враховувалися лише три ступені свободи: X-Y трансляція та обертання по Z Yaw.

Коли відлікова система має шість ступенів свободи, код LLAM був написаний таким чином, щоб враховувати трансляцію у всіх трьох основних напрямках, а також інші дві компоненти обертання (риск і кочення). Це було зроблено як для вивчення збільшеного обчислювального завдання шістьох ступенів свободи, так і для того, щоб наступні дослідження спробували застосувати LLAM до повітряного робота з шістьма ступенями свободи.

Відлікова система є рухомою системою, специфічною для сенсора TF mini. Хмари точок, повернені з сенсора TF mini, знаходяться в праворучній декартовій координатній системі з осі Z вперед, осі Y вниз та осі X праворуч відносно об'єктиву камери, коли він встановлений вертикально. Початок координат знаходиться в центрі об'єктива камери.

Відлікова система двовимірна декартова система координат з початком координат у центрі сенсора. Однак, оскільки перетворення між цими двома форматами просте і відбувається в локальній відліковій системі сенсора, для забезпечення послідовності вимірів відлікова система використовує лише декартові значення відлікова система позначається ведучим верхнім індексом перед змінною. У випадку перетворень між відліковими системами початкова відлікова система, що перетворюється, позначається ведучим нижнім індексом. Наприклад, перетворення T, яке відображає точку x (XYZ координати) з відлікової системи у відлікову систему, позначається.

$$x^w = {}^w_R T x^R \quad (3)$$

Багато 3D-моделей створюються за допомогою процесів тривимірного сканування, таких як LIDAR; проте, у багатьох ситуаціях можливо зібрати лише частковий скан з будь-якої перспективи. Щоб створити повну 3D-

модель, необхідно об'єднати зібрані часткові скани в один скан [18]. Цей процес називається реєстрацією і застосовується як до точкових хмар, так і до мережових 3D-моделей. Зазвичай він включає три кроки. По-перше, знаходять відповідність між точками в послідовних сканах. Далі, на основі цих відповідностей точок обчислюється трансформація, яка мінімізує відстань між відповідними точками. Нарешті, трансформовані точкові хмари об'єднуються в одну точкову хмару, що представляє повну модель [19]. Існує безліч методів реєстрації точкових хмар, і багато варіацій кожного методу розроблено для відповідних застосувань. Застосуємо три загальні методи реєстрації точкових хмар: Iterative Closest Point (ICP), Coherent Point Drift (CPD) та Normal Distribution Transform (NDT).

Реєстрація повертає або 2D, або 3D трансформації. У 2D трансформації використовуються три параметри: трансляційні параметри t_x та t_y та параметр кутового обертання ϕ . Колективно ці три параметри записані у вектор p як

$$p = [t_x t_y \phi] \quad (4)$$

де x - це двовимірна точка, яка трансформується.

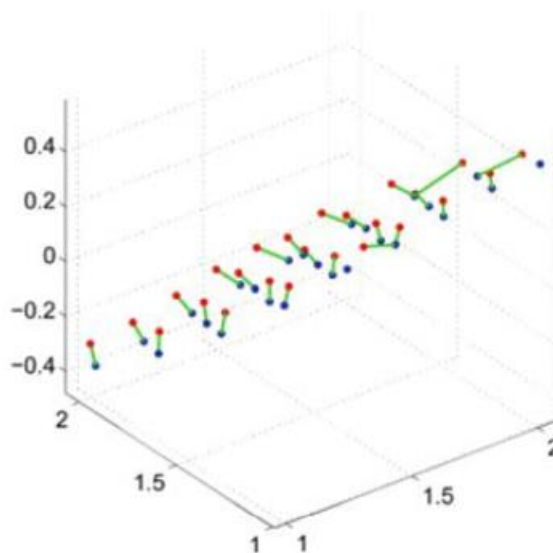


Рисунок 2.5 - Ітерація алгоритму ICP

Візуальне зображення алгоритму ICP можна побачити на рис 2.5. На першій ітерації встановлюється відповідність між точками у червоному наборі та найближчими точками у синьому наборі. Обчислюється трансформація, яка

відображає сині точки ближче до червоних точок. На другій ітерації багато відповідностей між точками не змінюються, однак встановлюються кілька різних відповідностей. Процес трансформації повторюється. Досліджуючи третю та четверту ітерації алгоритму, можна побачити, що жодна з відповідностей між точками не змінюється, це зазвичай розглядається як збіжність і припиняє роботу алгоритму.

KD-дерево починається з кореневої комірки, яка містить усі дані у наборі. Потім воно створюється шляхом знаходження медіанного значення вздовж одного з розмірностей дерева та розбиття точкових даних на частини за допомогою гіперплощини, перпендикулярної до розглянутої вісі. Точки по обидва боки гіперплощини розділяються значенням розглянутої розмірності таким чином, що більші значення потрапляють на один бік площини (наприклад, праворуч), менші значення - на інший (ліворуч). Розділення повторюється вздовж інших розмірностей, створюючи нові рівні дерева. Коли розглянута остання розмірність, алгоритм повертається до першої розмірності та подальше розділяє дані у межах кожної з наявних частин. Це продовжується рекурсивно до тих пір, поки у кожній частині не залишаться лише одиночні точки. Побудова робить пошук KD-дерева для точок у визначеному діапазоні або найближчих сусідів простим та ефективним.

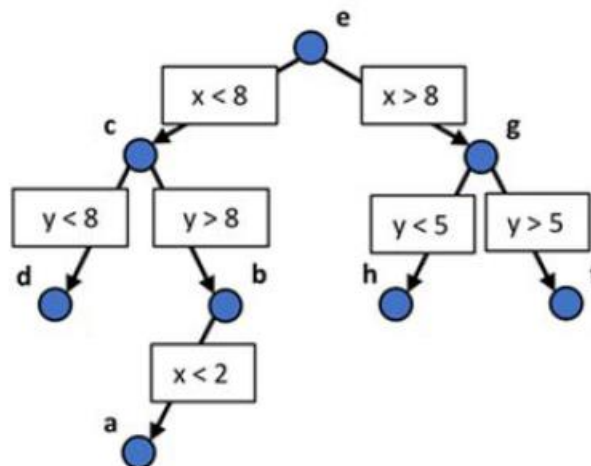


Рисунок 2.6 - Приклад двовимірного KD-дерева.

Побудова простого двовимірного KD-дерева показана на рис 2.6. Спочатку кореневий вузол встановлюється на основі медіанного значення

першої координати. Точки з меншими першими координатами (горизонтальна вісь) сортуються ліворуч, більші - праворуч, створюючи два дочірні вузли. Потім кожен з цих дочірніх вузлів розділяється вздовж другої координати (вертикальна вісь) відповідно до медіанного значення другої координати в кожному з дочірніх. У цьому прикладі за дві розділові операції отримуємо одну залишаючися точку, лист, і тим самим завершуємо побудову KD-дерева. KD-дерево є зручною структурою даних як для пошуку за діапазоном, так і для найближчого сусіда. Пошук за діапазоном корисний для сегментації точкових хмар LIDAR. Пошук найближчих сусідів у KD-дереві корисний для швидкого знаходження найближчих відповідностей точок для ICP.

Висновки до розділу 2

Удосконалення систем LIDAR і постійний розвиток алгоритмів обробки даних відкривають безліч можливостей для застосування робототехніки в різних галузях, зокрема військових операціях. Зменшення витрат на системи LIDAR та підвищення їхньої роздільної здатності робить їх доступнішими та ефективнішими для використання в різних сценаріях. Крім того, широкий спектр методів локалізації, картографування та навігації, що базуються на технології LIDAR, дозволяє розробникам створювати рішення, які відповідають різним вимогам і сценаріям застосування.

Використання KD-дерев, алгоритмів реєстрації точкових хмар та інших технологій в поєднанні з LIDAR дозволяє швидко та точно аналізувати отримані дані. Це важливо для вимірювання, визначення маршрутів та уникнення перешкод в реальному часі. Постійне дослідження і вдосконалення алгоритмів та методів обробки даних дозволяє розвивати все більш ефективні та надійні рішення для військового та цивільного застосування.

Завдяки поєднанню передових технологій і розвитку алгоритмів, ми можемо очікувати подальше розширення можливостей робототехніки і її застосування в різних галузях. Використання систем LIDAR, спільно з іншими датчиками та алгоритмами, є ключовим чинником у подальшому розвитку автономних систем та роботів, які здатні працювати в різних умовах.

3. АПАРАТНО-ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

3.1 Вибір апаратної платформи

Модель Raspberry Pi 4 Model B пропонує потужний 64-бітний чотирьохядерний процесор, який забезпечує високу продуктивність. Здатність до подвійного відображення зображень з роздільною здатністю до 4K за допомогою двох мікро HDMI-портів, а також апаратний декодер відео з роздільною здатністю до 4Kp60 роблять його ідеальним вибором для використання у проектах, де важлива якість зображення. За допомогою до 8 ГБ оперативної пам'яті, можливості бездротового зв'язку на двох діапазонах, Bluetooth 5.0 та гігабітного Ethernet, цей пристрій готовий до різноманітних завдань збору та обробки даних.

Ця модель не тільки зберігає сумісність з попередніми версіями Raspberry Pi, але й відзначається значними покращеннями у продуктивності, з'єднанні та можливостях мультимедіа. Двосмутовий бездротовий зв'язок і Bluetooth мають сертифікацію модульної відповідності, що робить їх ідеальними для використання у проектах з мінімальними витратами на випробування відповідності.



Рисунок 3.1 - Raspberry PI 4 Model B[19].

Крім того, Raspberry Pi 4 Model B є ідеальним вибором для проектів, які потребують великої кількості обчислень, таких як обробка великих обсягів даних з камер або лідарів. Завдяки можливостям підключення до двох моніторів з роздільною здатністю 4K, цей пристрій може бути використаний

для відображення реального часу аналітики або візуалізації даних у великому масштабі.

Також, наявність гігабітного Ethernet та USB 3.0 дозволяє швидко передавати великі обсяги даних між пристроями, що робить Raspberry Pi 4 Model B ідеальним варіантом для систем, які вимагають великої пропускної здатності мережі та швидкого зберігання та обробки даних.

Переваги Raspberry Pi 4 Model B включають:

- висока продуктивність: чотирьохядерний 64-бітний процесор забезпечує швидку та ефективну обробку даних;
- підтримка високих роздільних здатностей: можливість підключення до двох моніторів з роздільною здатністю до 4K забезпечує високоякісне відображення даних;
- розширені можливості зберігання: підтримка карт пам'яті MicroSD та USB 3.0 дозволяє зберігати та передавати великі обсяги даних;
- бездротовий зв'язок: підтримка двосмугового бездротового зв'язку і Bluetooth 5.0 забезпечує швидке та надійне підключення до мережі та пристроїв;
- сумісність та зворотна сумісність: зберігає сумісність з попередніми версіями Raspberry Pi, що полегшує перехід на новий пристрій без необхідності зміни програмного забезпечення;
- енергоефективність: прийняття енергоефективних технологій дозволяє знизити витрати електроенергії та тривалість роботи пристрою;
- розширені можливості: інтеграція модулів PoE (Power over Ethernet) за допомогою окремого модуля HAT дозволяє спростити підключення до мережі Ethernet і живлення пристрою.

Модель Raspberry Pi 3 Model B+ є покращеною версією попередньої моделі і базується на системі на кристалі BCM2837B0 (SoC). Основні переваги

цієї моделі включають у себе потужний 64-бітовий чотирьохядерний процесор ARMv8 та високопродуктивний графічний процесор VideoCore IV.

Оновлена версія Raspberry Pi 3 Model B+ має кілька покращень у продуктивності порівняно з попередньою моделлю, включаючи вищу частоту процесора (1,4 ГГц проти 1,2 ГГц), підвищену пропускну здатність Ethernet та підтримку двосмугового WiFi. Крім того, вона підтримує Power over Ethernet за допомогою окремого модуля.

Бездротовий LAN пристрій обладнаний сертифікацією модульної відповідності, що спрощує інтеграцію плати в кінцеві пристрої та значно зменшує витрати та час на тестування відповідності стандартам.

Raspberry Pi спроектований Фондом Raspberry Pi з метою надання доступної платформи для експериментів та освіти в галузі комп'ютерного програмування. Цей пристрій може використовуватися для великої кількості завдань, включаючи обробку текстових даних, роботу з електронними таблицями, відтворення відео високої якості, гри та програмування. Підключення периферійних пристроїв, таких як клавіатура та миша, здійснюється за допомогою чотирьох USB-портів, що розташовані на платі.

Завдяки заголовку GPIO та компактному розміру, Raspberry Pi може використовуватися як програмований контролер у широкому спектрі застосувань в робототехніці та електроніці. Крім того, він сумісний з різними додатковими модулями та платами, що дозволяє розширювати його можливості та створювати різноманітні проекти у сфері робототехніки та автоматизації.



Рисунок 3.2 - Raspberry Pi 3 B+

Переваги Raspberry Pi 3 Model B+:

- покращена продуктивність: завдяки вдосконаленій апаратній архітектурі, включаючи вищу частоту процесора (1,4 ГГц проти 1,2 ГГц), ця модель пропонує покращену продуктивність порівняно з попередніми версіями;
- підтримка двосмугового WiFi: Raspberry Pi 3 Model B+ має підтримку двосмугового бездротового зв'язку, що дозволяє пристрою працювати в двох частотних діапазонах (2,4 ГГц та 5 ГГц), що забезпечує кращу стабільність та швидкість зв'язку;
- підтримка Ethernet: ця модель має підтримку Gigabit Ethernet, що забезпечує швидкий та надійний провідний зв'язок;
- сумісність з Power over Ethernet (PoE): Raspberry Pi 3 Model B+ підтримує Power over Ethernet за допомогою додаткового модуля (PoE HAT), що дозволяє живити пристрій через мережевий кабель;
- низька вартість: як і інші продукти Raspberry Pi, ця модель доступна за помірну ціну, що робить її доступною для широкого кола користувачів та проектів.

Недоліки Raspberry Pi 3 Model B+:

- обмежена обробка графіки: незважаючи на покращення, вбудований графічний процесор Raspberry Pi 3 Model B+ може бути обмеженим для деяких вимог, таких як відтворення важких 3D-графічних сцен або високоякісного відео;
- обмежені можливості для великих обсягів даних: захоплення та обробка великих обсягів даних може бути викликаною для Raspberry Pi, оскільки він має обмежені ресурси оперативної пам'яті та обробки;
- потреба в додаткових аксесуарах: деякі функції, такі як PoE, вимагають додаткових аксесуарів, що може підвищити загальну вартість використання пристрою;

- залежність від операційної системи: хоча Raspberry Pi підтримує різноманітні операційні системи, деякі програми або драйвери можуть бути несумісними або неоптимізованими для цієї платформи.

Розглянувши Raspberry Pi Zero, слід відзначити, що його основною перевагою є доступність та вигідна ціна, що робить його привабливим для різних користувачів, зокрема студентів, аматорів та ентузіастів. Ця мініатюрна платформа відкриває широкі можливості для проведення різноманітних експериментів та проектів в області робототехніки, автоматизації та комп'ютерної техніки.

Також слід відзначити компактні розміри Raspberry Pi Zero, що робить його досить зручним у використанні у вузьких просторах, таких як корпуси роботів чи портативні пристрої. Це робить його ідеальним вибором для проектів з обмеженим обсягом чи місцем для розміщення.

Незважаючи на свою компактність, Raspberry Pi Zero має достатньо потужності для виконання багатьох завдань, включаючи обробку даних, управління роботами та взаємодію з різними сенсорами та пристроями. Це робить його високопродуктивним та універсальним інструментом для розробки різноманітних проектів.

Проте, слід враховувати обмежені можливості щодо обсягу пам'яті та обробки, що може бути перешкодою для великих обчислювальних завдань чи проектів, які потребують значних ресурсів. Також, у порівнянні з більш потужними моделями Raspberry Pi, Raspberry Pi Zero має обмежені можливості розширення функціональності.



Рисунок 3.3 - Raspberry Pi Zero

Переваги Raspberry Pi Zero:

- доступність та вартість: Raspberry Pi Zero доступний за доступну ціну, що робить його привабливим для широкого кола користувачів, зокрема студентів, аматорів та ентузіастів;
- компактні розміри: мініатюрний розмір Raspberry Pi Zero дозволяє легко використовувати його в обмежених просторових умовах, що робить його ідеальним вибором для різноманітних проектів з обмеженим обсягом або місцем для розміщення;
- потужність: незважаючи на свої компактні розміри, Raspberry Pi Zero має достатньо потужності для виконання багатьох завдань, включаючи обробку даних, управління роботами та взаємодію з різними сенсорами та пристроями.

Недоліки Raspberry Pi Zero:

- обмежені можливості: Raspberry Pi Zero має обмежені можливості щодо обсягу пам'яті та обробки, що може бути перешкодою для великих обчислювальних завдань чи проектів, які потребують значних ресурсів;
- обмеженість розширення функціональності: у порівнянні з іншими моделями Raspberry Pi, Raspberry Pi Zero має обмежені можливості розширення функціональності через обмежену кількість роз'ємів та інтерфейсів.

Raspberry Pi 4 Model B є ідеальним вибором для навігації роботів на основі лідару та камер. Завдяки своїм потужним характеристикам, включаючи високопродуктивний 64-бітовий чотирьохядерний процесор, підтримку відображення на двох дисплеях з роздільною здатністю до 4К, можливість обробки відео з роздільною здатністю до 4Кр60 та до 8 ГБ оперативної пам'яті, Raspberry Pi 4 Model B здатний ефективно обробляти дані з лідару та камери для точної навігації роботів. Крім того, наявність різноманітних інтерфейсів, таких як USB 3.0, Gigabit Ethernet та бездротова мережа з підтримкою Bluetooth 5.0, дозволяє легко підключати зовнішні пристрої, необхідні для роботи з лідаром та камерою

Таблиця 3.1 - Порівняльна характеристика між Raspberry Pi 3 Model B+, Raspberry Pi Zero W та Raspberry Pi 4 Model B

	Частота процесора	Кількість ядер процесора	Архітектура	Оперативна пам'ять	Інтерфейси	Бездротова мережа
Raspberry Pi 3 Model B+:	1.4 ГГц	4	ARMv8	1 ГБ LPD DR2 SDR AM	4 x USB 2.0, HDMI, Ethernet, GPIO	2,4 / 5 ГГц 802.11b/g/n/ac, Bluetooth 4.2
Raspberry Pi Zero W:	1 ГГц	1	ARM11	512 МБ LPD DR2 SDR AM	Mini HDMI, Micro USB, GPIO	2,4 ГГц 802.11n, Bluetooth 4.1
Raspberry Pi 4 Model B:	1.5 ГГц	4	ARM Cortex-A72	2, 4 або 8 ГБ LPD DR4 SDR AM	2 x USB 3.0, 2 x USB 2.0, 2 x micro HDMI, Gigabit Ethernet, GPIO	2,4 / 5 ГГц 802.11b/g/n/ac, Bluetooth 5.0

Кожна модель має свої унікальні характеристики і відповідає різним потребам та бюджетам. Raspberry Pi 3 Model B+ ідеально підходить для більшості проектів, Raspberry Pi Zero W - для вбудованих систем з обмеженим бюджетом та місцем, а Raspberry Pi 4 Model B - для завдань, що вимагають більшої продуктивності та можливостей підключення.

3.2 Опис апаратної платформи та компонентів

Для взаємодії з різноманітними сенсорами та актуаторами у системі обрано міні-комп'ютери Raspberry Pi, зокрема моделі Raspberry Pi 4 Model B. Ці комп'ютери забезпечують потужність та гнучкість для виконання обчислень та управління роботом. Апаратною платформою обрано Raspberry Pi 4 Model B. Для живлення платформи використовується Raspberry Pi UPS НАТ. Також використовується TF Mini для вимірювання відстаней. Для візуального спостереження використовується Raspberry Pi Camera Module 3

На рис 3.4 зображен Raspberry Pi 4 Model B



Рисунок 3.4 - Raspberry Pi 4 Model B зовнішній вигляд.

Переваги Raspberry Pi 4 Model B:

- висока продуктивність: оснащений потужним 64-бітним чотирьохядерним процесором ARM Cortex-A72 з тактовою частотою 1,5 ГГц, який забезпечує значну швидкість роботи;
- підтримка відображення на двох моніторах: можливість підключення двох моніторів з роздільною здатністю до 4К, що

- робить його ідеальним для використання у сферах мультимедіа та розробки;
- розширені можливості пам'яті: версії з різним обсягом оперативної пам'яті (від 2 до 8 ГБ) для вибору відповідно до потреб користувача;
 - покращена мережева підтримка: вбудований двосмуговий Wi-Fi 5 (802.11ac) та Bluetooth 5.0 забезпечують швидку та надійну бездротову зв'язок;
 - розширені можливості зовнішніх пристроїв: має два порти USB 3.0 та два порти USB 2.0 для підключення зовнішніх пристроїв;
 - підтримка Ethernet: вбудований гігабітний Ethernet для стабільного з'єднання з мережею;
 - підтримка Power over Ethernet (PoE): можливість живлення через Ethernet за допомогою відповідного додаткового модуля;
 - сумісність з попередніми версіями: зберігає сумісність з попередніми моделями Raspberry Pi, що дозволяє легко переносити проекти та розробки.

Raspberry Pi 4 Model B - це потужна та універсальна платформа, яка ідеально підходить для використання з лідаром, камерою та системами навігації в приміщенні. Його висока продуктивність забезпечує швидку обробку даних в реальному часі, що є важливим для взаємодії з датчиками та камерами. Крім того, Raspberry Pi 4 підтримує високороздільний відеовихід до 4K, що дозволяє отримувати високоякісні зображення від камер та лідара. Завдяки багатофункціональним інтерфейсам, таким як USB, Ethernet та HDMI, ви з легкістю підключите різноманітні пристрої. Вбудована підтримка Wi-Fi та Bluetooth дозволяє бездротово підключати Raspberry Pi до мережі інтернету та інших пристроїв, що забезпечує зручність у використанні та обміні даними. І, нарешті, велика спільнота користувачів Raspberry Pi забезпечує доступ до

багатої документації та підтримки, що спрощує розробку та впровадження проектів навігації в приміщенні.

TF mini є оптимальним рішенням для навігації в приміщенні, завдяки своїм високим технічним можливостям та широкому спектру функціональності.



Рисунок 3.5 – Дальномір TF mini[20]

TF mini - це відмінний вибір для проектів навігації в приміщенні з декількох причин. По-перше, цей датчик є надзвичайно компактним і легким, що дозволяє легко інтегрувати його в будь-яку систему навігації. Він має низьке енергоспоживання, що робить його ідеальним для проектів, які працюють на батареях або потужності обмежені.

Друга причина - це висока точність і швидкість вимірювань. TF mini здатний точно визначати відстань до об'єктів на значну відстань, що робить його ідеальним для навігації в просторі з високим рівнем точності. Він також має швидку частоту оновлення, що дозволяє отримувати актуальні дані в реальному часі.

Таблиця 3.2 – Технічні характеристики TF mini

	TF mini
Зовнішній розмір	35 x 21 x 18 мм
Вага	4.5г
Інтерфейс	UART (Universal Asynchronous Receiver-Transmitter)
Діапазон робочої температури	Від -10°C до +60°C
Робоча напруга	5 В

Максимальна відстань вимірювання	12 метрів
Точність вимірювання	± 6 сантиметрів

Крім того, TF mini має простий інтерфейс з підтримкою різних протоколів зв'язку, що спрощує його інтеграцію з Raspberry Pi та іншими контролерами. Це дозволяє швидко і легко включити його в проєкт і почати отримувати дані.

Загалом, TF mini є надійним і точним датчиком відстані, який ідеально підходить для проєктів навігації в приміщенні завдяки своїм компактним розмірам, високій точності і швидкості вимірювань, а також простому інтерфейсу.

Модуль камери Raspberry Pi 3 є важливим компонентом для проєктів навігації з декількох причин. По-перше, він забезпечує можливість збирати відеодані, що є ключовим для визначення місцезнаходження та орієнтації в просторі. Завдяки високій роздільній здатності і якості зображення, камера дозволяє роботі точно визначати своє оточення і реагувати на нього.

Друга причина - це можливість використовувати комп'ютерне зорове сприйняття для виявлення об'єктів та перешкод у навколишньому середовищі. Це дозволяє роботу взаємодіяти з оточенням, виявляти перешкоди та побудовувати маршрути для уникнення них.

Крім того, модуль камери Raspberry Pi 3 простий у використанні та має гнучкий програмний інтерфейс, що дозволяє легко інтегрувати його з іншими компонентами системи та обробляти отримані відеодані.



Рисунок 3.6 – Модуль камери Raspberry Pi 3[21]

Загалом, модуль камери Raspberry Pi 3 є важливим інструментом для навігації в приміщенні, оскільки він забезпечує можливість отримувати відеодані, використовувати комп'ютерне зорове сприйняття для визначення оточення та виявлення перешкод, і має простий інтерфейс для інтеграції з іншими компонентами системи.

Таблиця 3.3 – Технічні характеристики камери Raspberry Pi 3

	Raspberry Pi Camera Module 3
Роздільна здатність	
Зовнішній розмір	25 x 20 x 9 мм
Вага	3 г
Зображення	Фіксований фокус
Інтерфейс	CSI (Camera Serial Interface)
Діапазон робочої температури	Від -20°C до +60°C
Робоча напруга	3,3 В
Максимальна відстань вимірювання	Н / Д
Точність вимірювання	Н / Д

Переваги камери Raspberry Pi Camera Module 3 порівняно з іншими камерами включають:

- сумісність з Raspberry Pi: ця камера спеціально розроблена для сумісності з Raspberry Pi, що забезпечує зручну і легку інтеграцію та взаємодію з платформою;
- висока якість зображення: Raspberry Pi Camera Module 3 забезпечує високоякісні зображення та відео завдяки вбудованому об'єктиву та сенсору високої роздільної здатності;
- підтримка HD-відео: вона здатна записувати відео у форматі високої роздільної здатності HD (1080p), що робить її ідеальною для проєктів, які потребують високоякісних відеозаписів;
- низька вартість: вона доступна за помірну ціну, що робить її доступною для широкого кола користувачів та проєктів з обмеженим бюджетом;
- простота використання: завдяки простому підключенню та налаштуванню через інтерфейс Raspberry Pi, ця камера легко використовується навіть користувачами з мінімальним досвідом;
- розширені можливості: Raspberry Pi Camera Module 3 підтримує широкий спектр функцій та налаштувань, таких як автофокус, регулювання експозиції та баланс білого, що дозволяє користувачам налаштовувати зображення під свої потреби.

Камера Raspberry Pi Camera Module 3 є надійним і ефективним засобом для захоплення високоякісних зображень та відео, і вона чудово підходить для багатьох проєктів, включаючи навігацію в приміщенні.

Розширення Raspberry Pi UPS HAT - це надійне рішення для живлення Raspberry Pi, яке забезпечує безперебійну роботу пристрою навіть у випадку відключення основного джерела електроживлення. Воно забезпечує стабільне живлення, автоматично відключає пристрій в разі низького заряду акумулятора або відключення основного джерела живлення, має простий

процес встановлення та налаштування, і має індикатори стану живлення для зручного моніторингу. Це рішення також забезпечує захист від перевантажень, що дозволяє уникнути пошкоджень пристрою в разі виникнення перенапруги.



Рисунок 3.7 - Плата розширення Raspberry Pi UPS HAT[22]

Переваги Raspberry Pi UPS HAT:

- надійне живлення: забезпечує безперебійну роботу Raspberry Pi, навіть при відключенні основного джерела електроживлення;
- автоматичне відключення: відключає пристрій в разі низького заряду акумулятора або відключення основного джерела живлення, щоб уникнути пошкоджень;
- просте встановлення та налаштування: має простий процес встановлення та налаштування, що робить його легким у використанні;
- індикатори стану живлення: має індикатори стану живлення для зручного моніторингу роботи пристрою.

Недоліки Raspberry Pi UPS HAT:

- додаткова вартість: використання Raspberry Pi UPS HAT вимагає додаткових витрат на придбання цього пристрою, що може підвищити загальну вартість проєкту;

- обмежена потужність: UPS HAT може мати обмежену потужність, що може вплинути на можливості живлення підключених пристроїв.

Таблиця 3.4 – Технічні характеристики Raspberry Pi UPS HAT

Вбудований акумулятор	Літій-полімерний акумулятор є
Вбудований зарядний пристрій	Є
Вихідна напруга для Raspberry Pi	5 В
Максимальний вихідний струм	3 А
Інтерфейс зв'язку	I2C
Захист від перенапруги	Є
Захист від зворотного струму	Є
LED індикатори	Є
Підтримувані моделі Raspberry Pi	Raspberry Pi 3, 3B+, 4
Передавання даних через GPIO	Є
Автономний час життя	Залежить від потужності акумулятора та споживання Raspberry Pi

Загалом у проєкті Raspberry Pi 4 Model B виступає як головний контролер, який керує всіма іншими компонентами системи. Він отримує дані від сенсорів та камери, обробляє їх і приймає рішення щодо навігації та керування рухом платформи.

Сенсор TF mini відстежує оточуюче середовище, надаючи дані про відстань до об'єктів. Ця інформація використовується Raspberry Pi для побудови мапи навколишнього середовища та планування шляху руху робота.

Камера Raspberry Pi Camera Module 3 забезпечує систему зображеннями навколишнього середовища. Ці зображення можуть використовуватися для

виявлення об'єктів, розпізнавання маркерів або навігації за допомогою візуальних відомих точок.

UPS НАТ забезпечує постійне живлення Raspberry Pi, що дозволяє уникнути втрати даних або переривань у роботі через відключення живлення. Він гарантує безперебійну роботу системи навіть у випадку перебоїв у живленні.

Взаємодія між цими компонентами дозволяє створити повноцінну систему навігації та керування рухом робота в приміщенні, що враховує як зовнішнє середовище, так і внутрішні умови.

3.3 Розробка програмної частини

Після створення схеми та прототипу плати комплексу, був розроблений алгоритм, за яким функціонує програмний компонент системи.

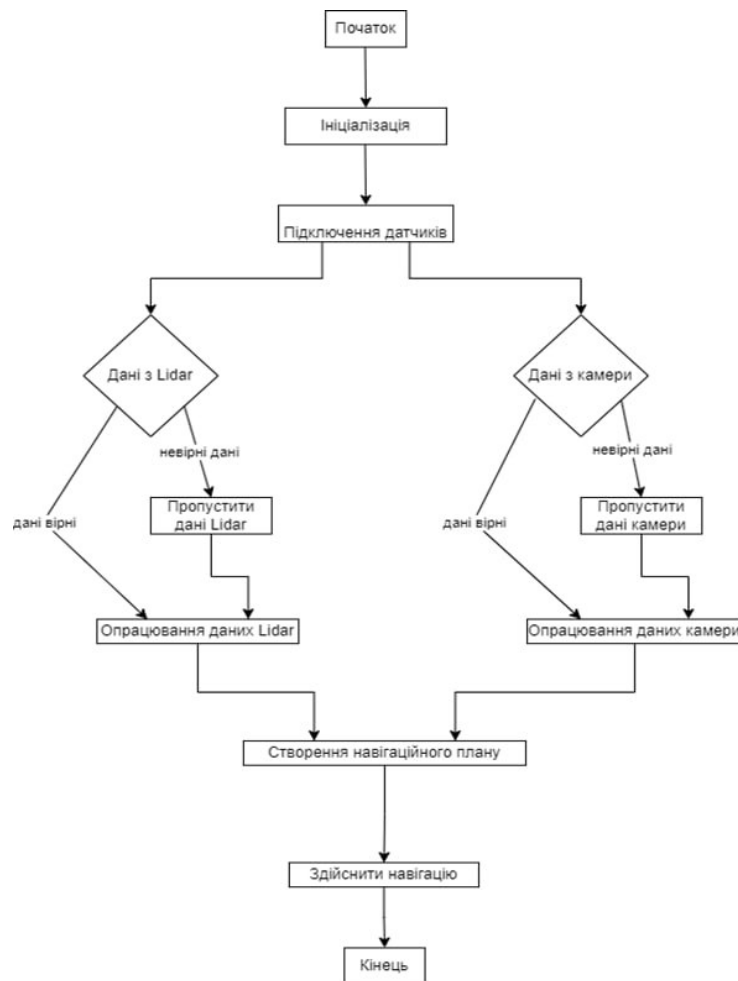


Рисунок 3.8 – Алгоритм взаємодії

Спочатку йде ініціалізація компонентів, перевірка чи всі компоненти підключені, потім йде підключення датчиків, далі збирається спочатку дані з лідару потім дані з камер обробляються та будують навігаційний план та робот переміщується.

Для програмної частини системи вибрано Python як мову програмування. Використання цієї мови для Raspberry Pi виявляється найбільш оптимальним і ефективним підходом для розв'язання поставлених задач.

Python - це широко використовувана мова програмування, яка пропонує багато переваг для розробки роботів, таких як простота, зрозумілість, гнучкість та розширений набір бібліотек і фреймворків. У цій статті ми розглянемо, як Python може бути використаний для створення інтелектуальних машин з використанням двох найпопулярніших фреймворків для робототехніки: ROS і PyRobot

Python має кілька потужних бібліотек для роботи з сенсорами, камерами та іншими пристроями, які можуть бути використані для навігації. Наприклад, OpenCV дозволяє обробляти відеопотік з камери для визначення положення робота в просторі, а бібліотека RPi.GPIO дозволяє керувати підключеними до GPIO портів сенсорами та моторами.

Завдяки цим бібліотекам, Python легко інтегрується з сенсорами та пристроями, що використовуються для навігації, що дозволяє реалізувати такі функції, як визначення відстані до об'єктів, виявлення перешкод та управління рухом.

Крім того, Python має широкий вибір алгоритмів штучного інтелекту та машинного навчання, які можуть бути використані для створення розумних стратегій навігації. Наприклад, алгоритми машинного навчання можуть навчити робота уникати перешкод або вибирати оптимальний маршрут.

Отже, поєднання Python та Raspberry Pi з його бібліотеками створює потужну та гнучку платформу для навігації роботів в приміщенні. Вони

дозволяють легко розробляти та налагоджувати програми, які забезпечують точну та ефективну навігацію роботів у реальному часі.

PyRobot[23] - це відкритий фреймворк, який надає високорівневий інтерфейс для роботизації та навігації. PyRobot базується на ROS і надає апаратно-незалежні API для керування роботами, такими як LoCoBot - недорога мобільна маніпуляційна платформа. Крім того, PyRobot[] також забезпечує функціонал у сприйнятті, плануванні, навчанні та мисленні.



Рисунок 3.9 – Відкрита бібліотека PythonRobotics[24].

PythonRobotics - це відкрите програмне забезпечення, яке надає набір алгоритмів та інструментів для реалізації різноманітних завдань в області робототехніки за допомогою мови програмування Python. Цей проект включає в себе реалізації алгоритмів з таких областей, як локалізація, картографування, планування маршрутів, слідування, обробка даних з датчиків і багато інших. PythonRobotics спрощує розробку програмного забезпечення для роботів шляхом надання готових реалізацій основних алгоритмів, що дозволяє розробникам швидко створювати та тестувати нові робототехнічні програми.

Для розробки програмного забезпечення на мові Python[25] в проекті було обрано середовище розробки JupyterLab[26].

JupyterLab - це інтерактивне середовище розробки, яке надає зручний інтерфейс для написання коду, виконання обчислень, візуалізації даних та досліджень. Воно базується на концепції блокнотів (notebooks), де користувач

може об'єднувати текст, код, графіки та інші елементи в одному документі. JupyterLab підтримує велику кількість мов програмування, включаючи Python, і надає розширювану архітектуру для інтеграції різноманітних інструментів та бібліотек. Його інтерактивний та гнучкий характер робить його ідеальним вибором для розробки робототехнічних програм, де важливо проводити експерименти, візуалізувати дані та аналізувати результати.

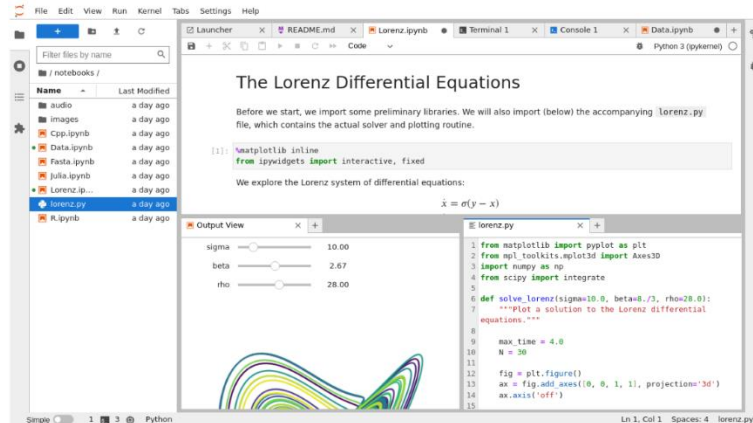


Рисунок 3.10 - Інтерактивне середовище розробки.

Нотатка - це документ, який можна поділитися, що поєднує комп'ютерний код, звичайні описи мовою, дані, багатофункціональні візуалізації, такі як 3D-моделі, діаграми, графіки та малюнки, а також інтерактивні елементи управління. Нотатка, разом з редактором (наприклад, JupyterLab), надає швидке інтерактивне середовище для прототипування та пояснення коду, дослідження та візуалізації даних, а також обміну ідеями з іншими користувачами.

3.4 Бібліотеки

ROS (Robot Operating System) широко використовується платформа для роботи, яка надає набір інструментів та бібліотек для розробки програмного забезпечення для роботів. Вона має різноманітні пакети для навігації, включаючи SLAM (Simultaneous Localization and Mapping), планування шляху і керування рухом.

```
import rospy
```

```
# Ініціалізуємо вузол ROS
rospy.init_node('my_node_name', anonymous=True)

# Додаткові інструкції можна додавати тут
import rospy
from sensor_msgs.msg import LaserScan
from geometry_msgs.msg import Twist

def laser_callback(msg):
    # Отримання даних з лідару
    ranges = msg.ranges
    min_distance = min(ranges)

    # Логіка руху на основі даних з лідару
    if min_distance > 0.5:
        # Якщо відстань до найближчого об'єкта більша за 0.5 метра,
        рухаємось вперед
        move_forward()
    else:
        # Інакше повертаємось
        turn()

def move_forward():
    # Логіка руху вперед
    pass

def turn():
    # Логіка повороту
    pass

def main():
    rospy.init_node('simple_scanning')

    rospy.Subscriber('/scan', LaserScan, laser_callback)
```

```
rospy.spin()

if __name__ == '__main__':
    main()
import rospy
from sensor_msgs.msg import LaserScan
import numpy as np
import matplotlib.pyplot as plt

# Функція обробки даних з лідару
def laser_callback(msg):
    # Отримуємо дані з лідару про відстані та кути
    ranges = msg.ranges # Масив відстаней
    angles = np.linspace(msg.angle_min, msg.angle_max,
len(ranges)) # Масив кутів

    # Перетворюємо відстані та кути в координати (x, y)
    xs = np.multiply(ranges, np.cos(angles)) # Координати x
    ys = np.multiply(ranges, np.sin(angles)) # Координати y

    # Візуалізація мапи
    plt.figure(figsize=(8, 6)) # Розмір вікна
    plt.scatter(xs, ys, s=1) # Відображення точок на графіку
    plt.xlabel('X Coordinate') # Підпис осі x
    plt.ylabel('Y Coordinate') # Підпис осі y
    plt.title('Laser Scan Map') # Заголовок графіку
    plt.gca().set_aspect('equal', adjustable='box') # Зробити
вісі однакового масштабу
    plt.show() # Показати графік

# Головна функція
def main():
    rospy.init_node('mapping') # Ініціалізація вузла ROS
```

```
# Підпис на топик з даними лідару
rospy.Subscriber('/scan', LaserScan, laser_callback)

rospy.spin() # Передача управління ROS

# Виклик головної функції при запуску програми
if __name__ == '__main__':
    main()
```

Спочатку робот обертає свій лазерний датчик в різних напрямках, вимірюючи відстані до об'єктів у своєму оточенні. Кожне вимірювання зберігається разом з відповідним кутом обертання датчика.

Після отримання даних лазерного сканера вони обробляються, щоб виокремити точки, які відображаються на поверхні об'єктів. Ці точки представляють собою відстані від робота до цих об'єктів у різних напрямках.

За допомогою цих точок і відомого положення робота у просторі можна побудувати тривимірну модель оточення. Ця модель може бути представлена у вигляді хмари точок або сітки, що відображає контури об'єктів у просторі.

Далі ця модель може використовуватися для навігації робота в просторі, визначення маршрутів для переміщення, уникнення перешкод та виконання інших завдань в залежності від конкретної задачі робота.

Висновки до розділу 3

У цьому розділі було розглянуто різноманітні аспекти навігації та сканування поверхні для роботів.

Перш за все, було представлено ряд компонентів, які можна використовувати для збору і обробки даних про оточення робота. Мікроконтролер Raspberry Pi 4 Model B, лазерний датчик відстані TF mini та камера Raspberry Pi Camera Module 3 - це основні складові, які забезпечують збір даних для навігації та сканування.

Далі було розглянуто принципи роботи цих компонентів та їх взаємодію в системі. Робота лазерного датчика полягає в вимірюванні відстаней до об'єктів у різних напрямках, після чого ці дані можуть бути використані для побудови карт оточення. Камера використовується для збору візуальної інформації про навколишнє середовище та аналізу цих даних для розпізнавання об'єктів та маркерів.

Нарешті, було представлено простий алгоритм сканування та побудови поверхонь на основі даних з лазерного датчика та камери. Цей алгоритм дозволяє роботу побудувати тривимірну модель оточення на основі зібраних даних і використовувати цю модель для навігації в просторі.

У цілому, розглянуті компоненти та алгоритми навігації та сканування поверхні надають роботам можливість ефективно орієнтуватися в навколишньому середовищі та виконувати різноманітні завдання з автономності.

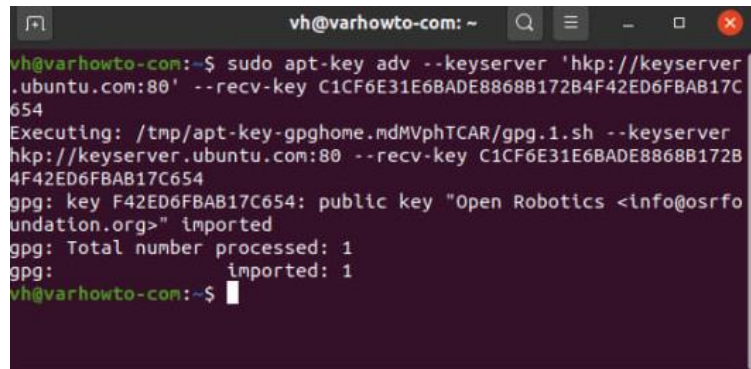
4. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ. АНАЛІЗ РЕЗУЛЬТАТІВ ОБЧИСЛЕНЬ

4.1 Виконання налаштувань

Ініціалізація ROS на Raspberry Pi передбачає наступні кроки:

Встановлення ROS на Raspberry Pi, використовуючи офіційну інструкцію для актуальної версії Raspberry Pi та операційної системи. Для операційних систем, таких як Debian, можна встановити ROS за допомогою команди apt:

```
sudo apt install ros-<distro>-desktop-full
```



```
vh@varhowto-com:~$ sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
Executing: /tmp/apt-key-gpghome.mDMVphtCAR/gpg.1.sh --keyserver hkp://keyserver.ubuntu.com:80 --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
gpg: key F42ED6FBAB17C654: public key "Open Robotics <info@osrfoundation.org>" imported
gpg: Total number processed: 1
gpg:      imported: 1
vh@varhowto-com:~$
```

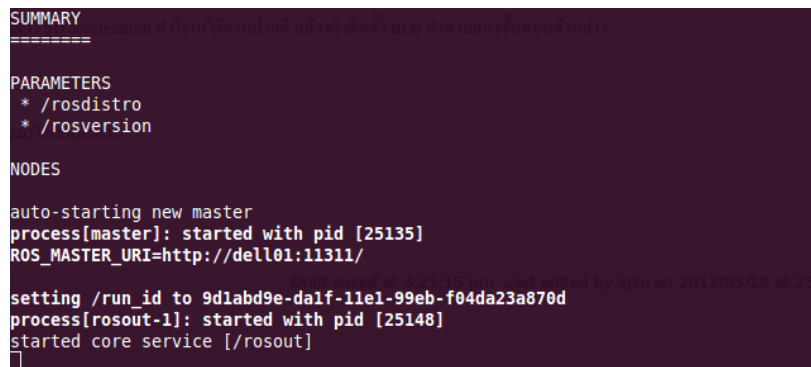
Рисунок 4.1 – Ініціалізація ROS

Де <distro> - це кодова назва версії ROS.

Ініціалізація ROS майстра після успішної установки ROS, запустить ROS майстра за допомогою команди:

```
roscore
```

Ця команда запустить основний процес ROS, який керує комунікацією між вузлами ROS.



```
SUMMARY
=====
PARAMETERS
* /rostdistro
* /rosversion

NODES

auto-starting new master
process[master]: started with pid [25135]
ROS_MASTER_URI=http://dell01:11311/

setting /run_id to 9d1abd9e-dalf-11e1-99eb-f04da23a870d
process[rosout-1]: started with pid [25148]
started core service [/rosout]
```

Рисунок 4.2 – Запуск основного процесу ROS

Перевіряємо працездатність після запуску майстра ROS переконайтеся, що він працює належним чином. Треба відкрити новий термінал і виконати команду для перевірки доступності майстра ROS:

```
rosnode list
```

Налаштування програми навігації робота за допомогою Python. В цьому коді має викликатись функції для отримання даних зі сенсорів, обробки цих даних та прийняття рішень щодо навігації

Треба імпортувати необхідні бібліотеки для роботи з навігацією.

```
import rospy
```

```
from sensor_msgs.msg import LaserScan
```

далі треба створити функцію-обробник для отримання даних зі сенсорів. Підписуємось на потрібну функцію, щоб отримувати дані з лазерного дальноміру.

```
def laser_callback(data):
```

```
    rospy.init_node('navigation_node')
```

```
    rospy.Subscriber('/laser_scan_topic', LaserScan, laser_callback)
```

Запускаємо цикл виконання програми, який буде очікувати отримання нових даних зі сенсорів та викликати функцію обробки для навігації.

```
    rospy.spin()
```

Це загальні налаштування програми навігації на Python для робота на Raspberry Pi. До цього коду можна додавати різноманітні алгоритми для обробки даних та прийняття рішень щодо навігації в приміщенні.

4.2 Побудова графіків навігації

Під час навігації за методом чистого переслідування роботу відомо, який є метою або список мет, і він намагається взяти найбільш прямий маршрут до цілі, уникаючи проміжних перешкод. Цей метод має недоліки, коли поєднаний з навігацією на основі потенційного поля. Під час спроби досягти мети робот може потрапити в локальний мінімум в потенційному полі і застрягнути. Якщо робот потрапляє в локальний мінімум, потрібен додатковий набір логіки

управління, щоб допомогти йому вибратися замкнутого мінімуму та продовжити рухатися до мети. Раніше в дисертаційній роботі, такі як слідування за стіною та слідування за рельєфом. Для цієї роботи основною метою було визначити, чи може робот проводити 3D SLAM в реальному часі. Під час експерименту робот був розміщений у лабораторному просторі таким чином, щоб в ньому були перешкоди, але не такі, що створюють локальні мінімуми в потенційному полі. Наприклад, перешкоди, які мають форму конкавного у, з точки зору робота, або внутрішні кути між роботом та метою можуть створити локальні мінімуми. Замість цього, під час навігації за методом чистого переслідування, перешкоди обмежувалися об'єктами, такими як дорожні знаки, відро з фарбою, стільці, зовнішні кути і т. д. Під час експерименту використовував виключно навігацію за методом чистого переслідування.

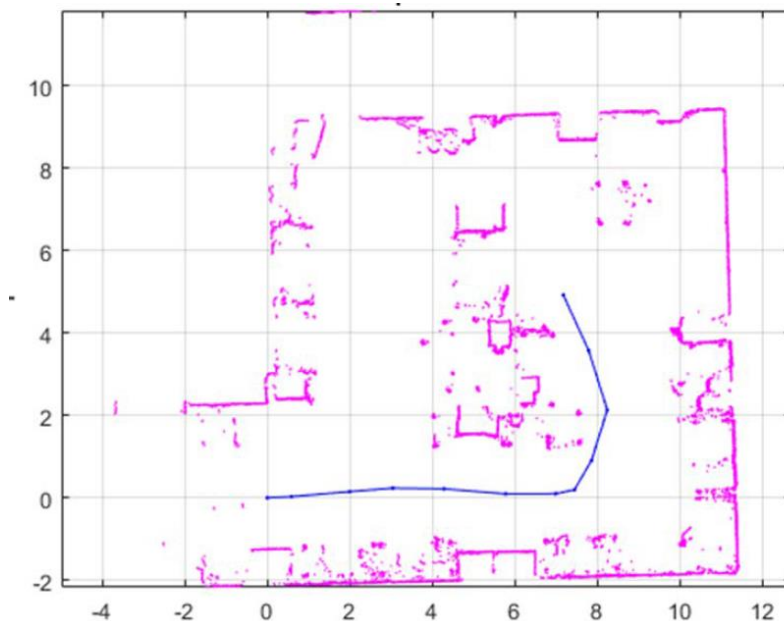


Рисунок 4.3 – Навігація робота у приміщенні.

Рожеві точки на рис 4.3 є окремими лазерними відгуками від сенсора, а синя доріжка - траєкторія руху робота, обчислена з трансформацій послідовних лазерних сканів. Карта зайнятості лабораторного середовища, розрахована за допомогою цієї карти SLAM, показана на рис 4.4.



Рисунок 4.4 – Карта зайнятості

Дані з лідару SLAM потім розділяються на комірки та позначаються відповідно. Білі комірки на рис 4.4 відповідають простору, який відомий як незайнятий перешкодами. Чорні комірки відображають відомий зайнятий простір, а сірі комірки представляють невідомий простір, який робот ще не дослідив або не бачить.

Для повноцінного дослідження нового середовища робот повинен визначити, які частини навколишнього середовища ще не досліджені. Після ідентифікації недосліджених областей карти слід вибрати одну з них на підставі визначених критеріїв для дослідження. Потім робот повинен використовувати вже наявні знання про перешкоди, а також інформацію про поточне положення для планування шляху для дослідження цієї фронти. Існує багато способів досягнення цього та багато критеріїв, за якими робот може вибирати фронти.

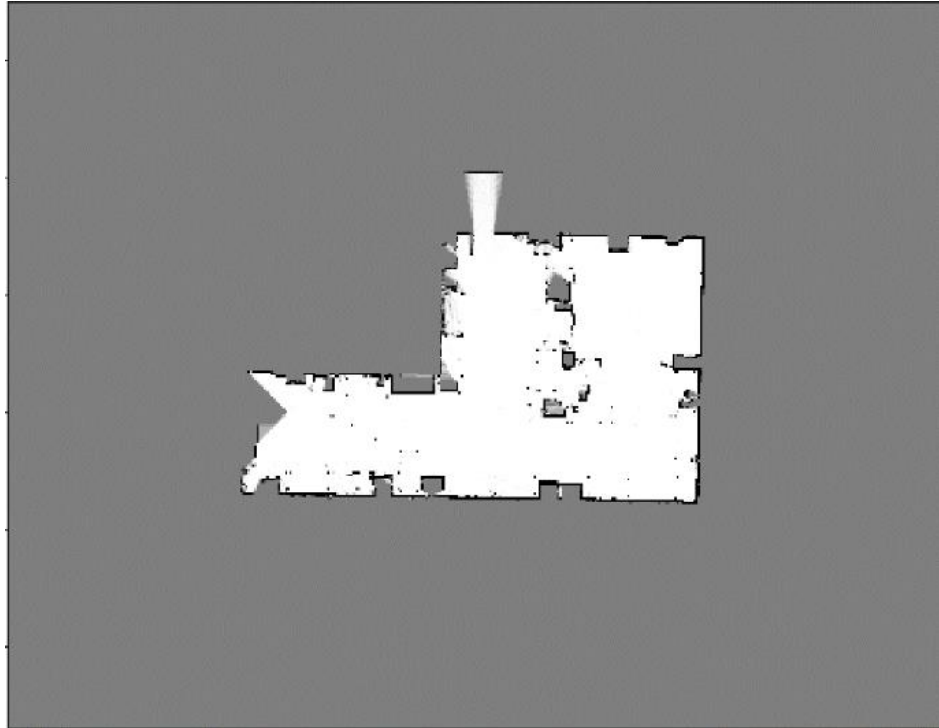


Рисунок 4.5 - Приклад початкової трикомпонентної карти зайнятості.

Тоді решта країв зберігається у бінарному зображенні як граничні пікселі. Якщо кілька граничних пікселів розташовані поруч, ці пікселі групуються разом як з'єднані, і група зберігається як межа. Приклад підключених граничних пікселів, які зберігаються у двійковому зображенні, зображено на рис 4.6.

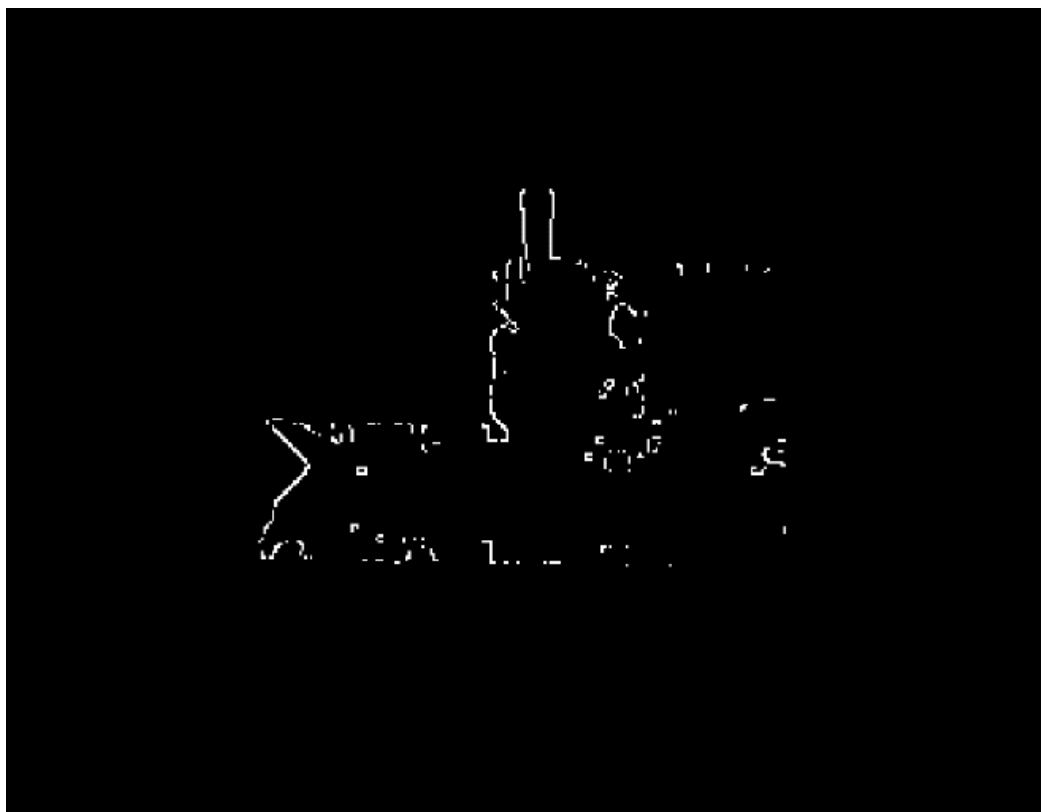


Рисунок 4.6 – Приклад граничних пікселів, що залишилися.

В структурі даних зберігаються параметри кожного еліпса, такі як його положення, розміри та орієнтація. Ця інформація корисна для подальшого аналізу та використання в алгоритмах обробки даних. Сортування цих параметрів за зменшенням площі допомагає визначити найбільш важливі еліпси в контексті досліджуваної області. Такий підхід дозволяє спростити подальшу обробку даних та робить її більш ефективною.

4.3 Розробка робота

Робот був зібраний на базі компонентів, які були надані раніше. Він складався з Raspberry Pi 4 Model B, Raspberry Pi UPS HAT, TF міні лідару та Raspberry Pi Camera Module 3. Цей робот є класним завдяки своїм вражаючим можливостям та високій продуктивності. За допомогою Raspberry Pi 4 Model B як основи, робот може виконувати різноманітні завдання з навігації та сприймання оточуючого середовища завдяки лідару та камері. Використання Raspberry Pi UPS HAT забезпечує надійне живлення робота, що дозволяє йому працювати безперервно. Загалом, цей робот представляє собою потужний і

універсальний інструмент для досліджень у галузі робототехніки та автоматизації.

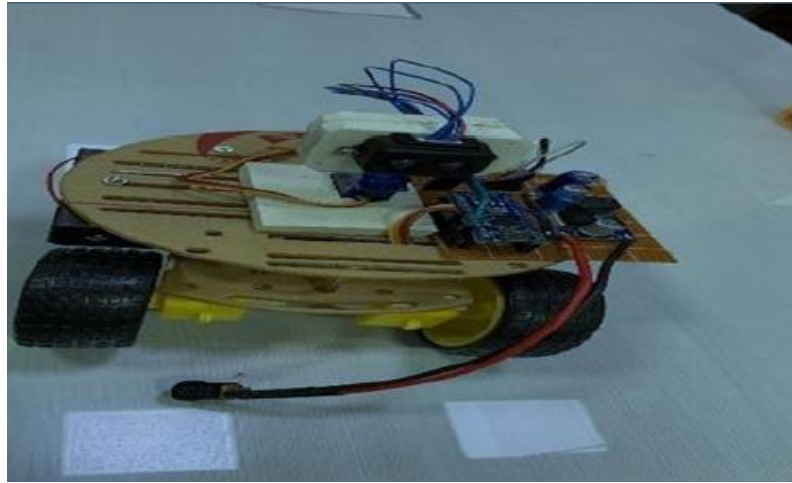


Рисунок 4.7 – Макет робота з лідаром.

Ці компоненти дозволили створити робота з високою функціональністю та надійністю. Він може виконувати різноманітні завдання з навігації, дослідження оточуючого середовища та взаємодії з ним. Використання цих компонентів у поєднанні з програмним забезпеченням дозволило створити потужний та універсальний робот для різноманітних досліджень у галузі робототехніки та автоматизації.

Висновки до розділу 4

У цьому розділі було проведено розробку робота на базі обраних компонентів, включаючи Raspberry Pi 4 Model B, Raspberry Pi UPS HAT, TF mini LiDAR та Raspberry Pi Camera Module 3. Ці компоненти були успішно інтегровані для створення потужного та функціонального робота, який може виконувати різноманітні завдання навігації та взаємодії з оточуючим середовищем.

Розроблений робот виявився досить ефективним у роботі, забезпечуючи надійну навігацію у приміщенні, а також здатність сприймати та аналізувати своє оточення за допомогою LiDAR та камери. Спроможність робота сканувати поверхні дозволяє йому створювати карту оточення та використовувати цю інформацію для оптимальної навігації в просторі.

Використання Raspberry Pi як основи робота дозволило ефективно керувати його функціоналом та взаємодіяти з іншими системами.

Отже, розроблений робот є потужним інструментом для дослідження та реалізації різноманітних завдань у галузі робототехніки та автоматизації. Його можливості забезпечують широкий спектр застосувань, включаючи автономну навігацію, дослідження середовища та виконання завдань у різних областях.

ВИСНОВКИ

У ході дослідження було вивчено та впроваджено сучасні технології в галузі робототехніки, що відкриває широкі перспективи для подальшого розвитку автономних систем та розумних роботів. Використання Raspberry Pi 4 Model B як основи для робота дозволило поєднати високу продуктивність та ефективність з доступністю та простотою використання. Багатофункціональність та гнучкість цієї платформи дозволили створити робота, який може виконувати різноманітні завдання в реальному часі, від сканування оточення до навігації в незнайомих середовищах.

Наявність додаткових модулів, таких як Raspberry Pi UPS HAT, забезпечує стабільне та безперервне живлення робота, що є критичним для його надійної роботи. TF mini LiDAR та Raspberry Pi Camera Module 3 забезпечують високоточну навігацію та здатність взаємодії з оточуючим середовищем, що робить робота відмінним інструментом для виконання завдань у різних умовах.

Під час експериментальних досліджень робот продемонстрував свою ефективність у вирішенні різноманітних завдань, таких як картографування невідомого середовища, навігація в просторі та взаємодія з об'єктами навколишнього середовища. Велика швидкість обробки даних та точність результатів зробили робота незамінним помічником у вирішенні складних завдань.

Недоліками проєкту є недостатня потужність живлення, існуюча батарея має дуже низький коефіцієнт роботи, що обмежує час автономної роботи. Заміна на потужнішу батарею є необхідною для збільшення часу роботи апаратно-програмного комплексу.

Для подолання можливих проблем, таких як уникнення попадання в локальні мінімуми, може знадобитися розробка додаткових алгоритмів або функцій для покращення навігації та уникнення перешкод.

На основі результатів дослідження можна зробити висновок, що розроблений робот є перспективним інструментом для використання у різних галузях, включаючи промисловість, медицину, аграрний сектор та багато інших. Його можливості дозволять ефективно виконувати різноманітні завдання і вносити значний внесок у розвиток сучасних технологій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Плохой О. С., Савінов В. Ю. Розробка методів виявлення та уникнення перешкод для мобільних роботів на основі датчиків LiDAR та камер. Інформаційні технології і автоматизація – 2023 : тези доп. XVI міжнародної науково-практичної конференції «Інформаційні технології і автоматизація - 2023» , 19–20 жовтня. 2023 р. Одеса : ОНТУ. ім. П. Н. Платонова, 2023. С. 368-370.
2. Fragapane, G.; de Koster, R.; Sgarbossa, F.; Strandhagen, J.O. Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda. Eur. J. Oper. Res. 2021, 292, 405–426. (дата звернення: 15.02.2024)
3. Sindermann, C.; Sha, P.; Zhou, M.; Wernicke, J.; Schmitt, H.S.; Sariyska, R.; Stavrou, M.; Becker, B.; Montag, C. Assessing the attitude towards artificial intelligence: Introduction of a short measure in German, Chinese, and English language. KI—Künstliche Intell. 2021, 35, 109–118. (дата звернення: 15.02.2024)
4. Shimmura, T.; Ichikari, R.; Okuma, T. Human-Robot Hybrid Service System Introduction for Enhancing Labor and Robot Productivity. In Advances in Production Management Systems. Towards Smart and Digital Manufacturing; Lalic, B., Majstorovic, V., Marjanovic, U., von Cieminski, G., Romero, D., Eds.; IFIP Advances in Information and Communication Technology; Springer International Publishing: Cham, Switzerland, 2020; pp. 661–669. (дата звернення: 15.02.2024)
5. Shimmura, T.; Ichikari, R.; Okuma, T.; Ito, H.; Okada, K.; Nonaka, T. Service robot introduction to a restaurant enhances both labor productivity and service quality. Procedia CIRP 2020, 88, 589–594. (дата звернення: 16.02.2024)

6. Cheong, A.; Lau, M.; Foo, E.; Hedley, J.; Bo, J.W. Development of a Robotic Waiter System. IFAC-PapersOnLine 2016, 49, 681–686. (дата звернення: 16.02.2024)
7. Blöcher, K.; Alt, R. AI and robotics in the European restaurant sector: Assessing potentials for process innovation in a Eng. Proc. 2023, 5, x FOR PEER REVIEW 10 of 4 high-contact service industry. Electron. Mark. 2021, 31, 529–551. (дата звернення: 16.02.2024)
8. Jeong, M.; Kim, K.; Ma, F.; DiPietro, R. Key factors driving customers' restaurant dining behavior during the COVID-19 pandemic. Int. J. Contemp. Hosp. Manag. 2022, 34, 836–858. (дата звернення: 16.02.2024)
9. Yu, Q.; Yuan, C.; Fu, Z.; Zhao, Y. An autonomous restaurant service robot with high positioning accuracy. Ind. Robot. 2019, 39, 271–281. (дата звернення: 16.02.2024)
10. Ma, E.; Bao, Y.; Huang, L.; Wang, D.; Kim, M.S. When a Robot Makes Your Dinner: A Comparative Analysis of Product Level and Customer Experience between the U.S. and Chinese Robotic Restaurants. Cornell Hosp. Q. 2021. (дата звернення: 17.02.2024)
11. Yang, J.; Gope, P.; Cheng, Y.; Sun, L. Design, analysis and implementation of a smart next-generation secure shipping infrastructure using autonomous robot. Comput. Netw. 2021, 187, 107779. (дата звернення: 17.02.2024)
12. Pudu. Smart Delivery Robot-Pudu Robotics. 2021. (дата звернення: 17.02.2024)
13. Chitta, S.; Marder-Eppstein, E.; Meeussen, W.; Pradeep, V.; Rodríguez Tsouroukdissian, A.; Bohren, J.; Coleman, D.; Magyar, B.; Raiola, G.; Lüdtke, M.; et al. ros_control: A generic and simple control framework for ROS. J. Open Source Software. 2017, 2, 456. (дата звернення: 17.02.2024)
14. Ladosz, P.; Coombes, M.; Smith, J.; Hutchinson, M. A Generic ROS Based System for Rapid Development and Testing of Algorithms for Autonomous Ground and Aerial Vehicles. In Robot Operating System (ROS): The Complete

- Reference; Studies in Computational Intelligence; Springer International Publishing: Cham, Switzerland, 2019; Volume 3, pp. 113–153. (дата звернення: 17.02.2024)
- 15.Chivarov, S.; Kopacek, P.; Chivarov, N. Cost oriented humanoid robot communication with iot devices via mqtt and interaction with a smart home hub connected devices. IFAC- PapersOnLine 2019, 52, 104–109. (дата звернення: 19.02.2024)
- 16.Tkáčik, M.; Březina, A.; Jadlovska, S. Design of a Prototype for a Modular Mobile Robotic Platform. IFAC PapersOnLine 2019, 52, 192–197. (дата звернення: 19.02.2024)
17. Noh, S.; Park, J.; Park, J. Autonomous Mobile Robot Navigation in Indoor Environments: Mapping, Localization, and Planning. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 21–23 October 2020; pp. 908–913. (дата звернення: 19.02.2024)
- 18.Belanche, D.; Casalo, L.V.; Flavián, C.; Schepers, J. Service robot implementation: A theoretical framework and research agenda. Serv. Ind. J. 2020, 40, 203–225. (дата звернення: 19.02.2024)
- 19.Raspberry Pi [Електронний ресурс]. – Режим доступу: <https://www.raspberrypi.com/products/>
- 20.TF mini [Електронний ресурс]. – Режим доступу: <https://mwf-metaldetectors.com/product/tfmini/>
- 21.Raspberry Pi Camera Module 3 [Електронний ресурс]. – Режим доступу: <https://www.raspberrypi.com/products/camera-module-3/>
- 22.Raspberry Pi UPS HAT [Електронний ресурс]. – Режим доступу: https://evo.net.ua/plata-rasshireniya-raspberry-pi-ups-hat-18160/?gad_source=1&gclid=CjwKCAiA_tuuBhAUEiwAvxkgTkBZVK-Ofqeex6crN7xYIdtzTELRIgMajbB42tpSmep2KmXRHPR1xBoCc-oQAvD_BwE

-
23. PyRobot [Електронний ресурс]. – Режим доступу: <https://github.com/facebookresearch/pyrobot>
24. PythonRobotics [Електронний ресурс]. – Режим доступу: <https://github.com/AtsushiSakai/PythonRobotics>
25. Python [Електронний ресурс]. – Режим доступу: <https://www.python.org/>
26. JupyterLab [Електронний ресурс]. – Режим доступу: <https://docs.jupyter.org/en/latest/>
27. Пат. на корисну модель US. Pat. № US9715030B1 Object detection and removal apparatus / Michael Zullo, заявл. 07.03.2016, опубл. 25.07.2017
28. Пат. на корисну модель US. Pat. № System and method for detecting bats and their impact on wind facilities / Michael J. Adler, Christian M. Newman, заявл. 17.06.2009, опубл. 13.02.2013
29. Пат. на корисну модель US. Pat. № US20150286340A1 * Optical detector/ Basf Se; заявл. 19.08.2013, опубл. 08.10.2015

Додаток А

Код програми

```
from functools import total_ordering
import matplotlib.pyplot as plt
import numpy as np

@total_ordering
class FloatGrid:
    """
    Представляє об'єкт, який містить одне значення типу float.
    Підтримує операції порівняння.
    """

    def __init__(self, init_val=0.0):
        self.data = init_val

    def get_float_data(self):
        return self.data

    def __eq__(self, other):
        if not isinstance(other, FloatGrid):
            return NotImplemented
        return self.get_float_data() == other.get_float_data()

    def __lt__(self, other):
        if not isinstance(other, FloatGrid):
            return NotImplemented
        return self.get_float_data() < other.get_float_data()

class GridMap:
    """
```

Представляє сіткову карту для управління та маніпулювання сітковими даними.

"""

```
def __init__(self, width, height, resolution,  
             center_x, center_y, init_val=FloatGrid(0.0)):
```

"""

Ініціалізує сіткову карту з вказаними розмірами та роздільною здатністю.

```
:param width: кількість сіткових клітинок по ширині  
:param height: кількість сіткових клітинок по висоті  
:param resolution: роздільна здатність сітки [м]  
:param center_x: центральна позиція x [м]  
:param center_y: центральна позиція y [м]  
:param init_val: початкове значення для всіх сіткових клітинок  
"""
```

```
self.width = width  
self.height = height  
self.resolution = resolution  
self.center_x = center_x  
self.center_y = center_y
```

```
self.left_lower_x = self.center_x - self.width / 2.0 *  
self.resolution
```

```
self.left_lower_y = self.center_y - self.height / 2.0 *  
self.resolution
```

```
self.n_data = self.width * self.height  
self.data = [init_val] * self.n_data  
self.data_type = type(init_val)
```

Інші методи опущено для кратності

```
def plot_grid_map(self, ax=None):
    """
    Відображає сіткову карту у вигляді теплової карти.

    :param ax: необов'язкова вісь matplotlib для відображення
    :return: об'єкт теплової карти
    """
    float_data_array = np.array([d.get_float_data() for d in
self.data])
    grid_data = np.reshape(float_data_array, (self.height,
self.width))
    if not ax:
        fig, ax = plt.subplots()
        heat_map = ax.pcolor(grid_data, cmap="Blues",
vmin=0.0, vmax=1.0)
        plt.axis("equal")

    return heat_map
```

Додаток Б

Алгоритм роботи програми

