

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

перший (бакалаврський)
(рівень вищої освіти)

Розробка системи автоматизації для управління
мобільною платформою підвищеної прохідності
(тема)

Виконав:
студент 4 курсу, групи АКТАКІТ-20-1
Кусков А. В.
(прізвище, ініціали)

Спеціальності 151 Автоматизація та
комп'ютерно-інтегровані технології
(код і повна назва спеціальності)

Тип програми Освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Автоматизація та
комп'ютерно-інтегровані технології
(повна назва освітньої програми)

Керівник доц. Аллахверанов Р. Ю.
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри КІТАР

(підпис)

Невлюдов І. Ш.
(прізвище, ініціали)

2024р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Автоматики і комп'ютеризованих технологій
Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
Рівень вищої освіти перший (бакалаврський)
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
Тип програми освітньо-професійна
Освітня програма Автоматизація та комп'ютерно-інтегровані технології
(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри КІТАР _____
(підпис)

«_____» _____ 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Кускову Андрію Віталійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка системи автоматизації для управління мобільною платформою підвищеної прохідності

затверджена наказом по університету від “ 03 ” червня 2024р. № 544 Ст.

2. Термін подання студентом роботи до екзаменаційної комісії 27.06.2024р.

3. Вихідні дані до роботи 3.1 Мобільна платформа LEGO;

3.2 Середовище розробки Microsoft Visual Studio;

3.3 Бібліотека OpenCV;

3.4 Мова програмування C++;

3.5 Операційна система Windows 10.

4. Перелік питань, що потрібно опрацювати в роботі

4.1 Вступ;

4.2 Аналіз предметної області;

4.3 Візуальне управління мобільними платформами;

4.4 Розробка програмного забезпечення для управління мобільною платформою за допомогою візуальної цілевказівки;

4.5 Охорона праці;

4.6 Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

Демонстраційний матеріал представлений у форматі презентації

Power Point (.ppt) – 12 с. формату А4*

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Аналіз предметної області</i>	<i>13.05 – 20.05.24</i>	<i>виконано</i>
2	<i>Візуальне управління мобільними платформами</i>	<i>21.05 – 29.05.24</i>	<i>виконано</i>
3	<i>Розробка програмного забезпечення для управління мобільною платформою за допомогою візуальної цілевказівки</i>	<i>30.05 – 09.06.24</i>	<i>виконано</i>
4	<i>Охорона праці</i>	<i>10.06 – 12.06.24</i>	<i>виконано</i>
5	<i>Оформлення пояснювальної записки</i>	<i>13.06 – 15.06.24</i>	<i>виконано</i>
6	<i>Подання роботи на перевірку Інтернет-системою StrikePlagiarism</i>	<i>16.06 – 19.06.24</i>	<i>виконано</i>
7	<i>Подання роботи на рецензію</i>	<i>20.06 – 22.06.24</i>	<i>виконано</i>
8	<i>Подання роботи на підпис зав. кафедри</i>	<i>23.06 – 25.06.24</i>	<i>виконано</i>
9	<i>Подання кваліфікаційної роботи в ЕК</i>	<i>26.06.24</i>	<i>виконано</i>

Дата видачі завдання 13.05.2024 р.

Студент

(підпис)

Кусков А. В.

(прізвище, ініціали)

Керівник роботи

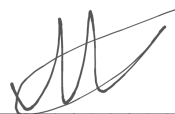
(підпис)

доц. Аллахверанов Р. Ю.

(посада, прізвище, ініціали)

Я, як студент ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

«20» червня 2024 р.



Кусков А. В.

РЕФЕРАТ

Пояснювальна записка: 70 с., 3 табл., 22 рис., 2 дод., 27 джерел.

NXT MINDSTORMS, OPENCV, C++, MICROSOFT VISUAL STUDIO,
МОБІЛЬНА ПЛАТФОРМА, УПРАВЛІННЯ, ПРОГРАМНЕ
ЗАБЕЗПЕЧЕННЯ, ВІЗУАЛЬНЕ УПРАВЛІННЯ.

Об'єкт розробки – управління мобільною платформою.

Предмет розробки – програмне забезпечення.

Мета роботи – розроблення програмного забезпечення для управління мобільною платформою підвищеної прохідності за допомогою візуальної цілевказівки.

У кваліфікаційній роботі опрацьовані мобільні платформи, їхня класифікація, особливості систем візуального управління. За допомогою середовища розроблення Microsoft Visual Studio і мови програмування C++ створено програмне забезпечення.

Результатом роботи є програмне забезпечення для управління мобільною платформою у робочому просторі гнучкої інтегрованої системи.

Область функціонування – системи розпізнавання команд для дистанційного управління мобільною платформою в умовах неповної визначеності технологічних ситуацій.

ABSTRACT

Explanatory note: 70 pp., 3 tables, 22 figs., 2 appendices, 27 sources.

NXT MINDSTORMS, OPENCV, C++, MICROSOFT VISUAL STUDIO,
MOBILE PLATFORM, CONTROL, SOFTWARE, VISUAL CONTROL.

The object of development is mobile platform management.

The subject of development is software.

The purpose of the work is to develop software for managing a mobile platform with increased traffic using visual targeting.

Mobile platforms, their classification, features of visual control systems are elaborated in the qualification work. The software was created using the Microsoft Visual Studio development environment and the C++ programming language.

The result of the work is software for managing a mobile platform in the workspace of a flexible integrated system.

The field of operation is command recognition systems for remote control of a mobile platform in conditions of incomplete certainty of technological situations.

ЗМІСТ

Перелік скорочень	7
Вступ	8
1 Аналіз предметної області	10
1.1 Мобільні платформи та області їхнього застосування	10
1.2 Пристрої управління мобільними платформами	14
1.3 Реалізація системи управління мобільними платформами	20
2 Візуальне управління мобільними платформами	28
2.1 Системи машинного зору мобільних платформ	28
2.2 Програмне забезпечення систем машинного зору	34
2.3 Системи візуального управління	42
3 Розроблення програмного забезпечення для управління мобільною платформою за допомогою візуальної цілевказівки	45
3.1 Опис алгоритмів системи візуального управління	45
3.2 Опис функцій програмного забезпечення	48
3.3 Порядок роботи програмного забезпечення. Тестовий приклад	51
4 Охорона праці	58
4.1 Аналіз умов праці в лабораторії	58
4.2 Промислова безпека в лабораторії	58
4.3 Виробнича санітарія і гігієна праці	59
4.4 Пожежна безпека лабораторії	61
Висновки	63
Перелік джерел посилання	64
Додаток А Текст програми	66
Додаток А Демонстраційний графічний матеріал	67

ПЕРЕЛІК СКОРОЧЕНЬ

ВП – виконавчий пристрій;

МП – мобільна платформа;

ПДУ – пульт дистанційного управління;

РМП – робототехнічна мобільна платформа;

САП – система автоматизованого проектування;

СМЗ – система машинного зору;

СУ – система управління;

ТЗ – транспортний засіб;

MFC – Microsoft Foundation Classes;

OpenCV – Open Computer Vision library.

ВСТУП

Із початком активного обороту сучасної мікропроцесорної техніки з'явилися сприятливі умови для розвитку малогабаритних мобільних керованих дистанційно платформ. Поява промислових платформ, застосування яких дозволяє заміщати людину на багатьох ділянках сучасного виробництва, разом із автоматичними системами, котрі можуть бути придатними в небезпечних для людини умовах, стають актуальною як науковою, так і технічною проблемою. Епоха робототехніки розпочинає своє втілення з галузей, в яких діє найбільш жорстке детерміноване середовище та процеси, як от у машинобудуванні. Зі збільшенням потреби в застосуванні сенсорних засобів, систем управління, робототехнічні системи почали розроблятися і для задіяння у немашинобудівних галузях. Передусім зазначені системи потрібні в тих видах діяльності людини, в яких її знаходження чи участь можуть стати небезпечними для життя чи здоров'я.

Програмне забезпечення, рекомендоване для управління мобільною платформою із залученням візуальної цілевказівки ґрунтується на інтегрованому середовищі Visual Studio. За його допомогою розробляють як консольні програми, так і програми з графічним інтерфейсом.

Діапазон використання мобільних платформ (МП) надзвичайно поширений:

- МП послуговуються під час вивчення космічного простору та глибин океану;
- на базі МП проводяться найскладніші хірургічні операції;
- техніка стає більш розумною та самостійною
- управління рухом, контроль середовища, прицілювання здійснює машина, а людині переадресується розв'язання тактичних задач, технічне обслуговування, вказівки об'єктів діяльності платформ.

Зважаючи на все зазначене вище, можемо стверджувати про актуальність науково-практичних робіт у галузі систем управління мобільних платформ, що окреслює спрямованість даної кваліфікаційної роботи рівня бакалавр.

Об'єкт розробки – управління мобільною платформою.

Предмет розробки – програмне забезпечення.

Мета роботи – розроблення програмного забезпечення для управління мобільною платформою підвищеної прохідності за допомогою візуальної цілевказівки.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- проаналізувати технічне завдання;
- дослідити системи управління мобільними платформами;
- створити програмне забезпечення для управління мобільною платформою із залученням візуальної цілевказівки.

Пояснювальну записку з кваліфікаційної роботи оформлено згідно з ДСТУ 3008:2015 [1], а також з рекомендаціями з підготовки і оформлення кваліфікаційної роботи здобувачами першого (бакалаврського) рівня вищої освіти [2-3].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Мобільні платформи та області їхнього застосування

З огляду на те, що в різних джерелах інформації відсутнє однозначне трактування термінів, пов'язаних із мобільними платформами, а разом з тим і загальноприйнятої класифікації, перш за все, опрацюємо кілька визначень.

Оскільки загальне поняття «платформа» сприймається окремими авторами по-різному, отже, найбільш доцільним уважатимемо таке трактування, в якому це машина з автоматичним або інтерактивним управлінням; перепрограмована чи самонавчальна; мобільна чи маніпуляційна; з кількома ступенями рухливості; придатна для детермінованих, так і недетермінованих середовищ, а також процесах [4].

Під поняттям мобільних платформ рекомендується розглядати лише пристрої, переміщення яких активізується невизначеною (не прокладеною попередньо), необмеженою траєкторією.

До таких платформ пропонується записувати тільки машини з декількома ступенями рухливості. Бо машини з одним ступенем рухливості давно сприймають просто як транспортні засоби.

Платформи вважаються механічними помічниками людини (рисунок 1.1), оскільки вони спроможні реалізовувати операції за інстальованою до них програмою, до того ж реагувати на середовище.

Зважаючи на те, що мобільні платформи можуть реалізовуватись як у детермінованих, так і недетермінованих середовищах та процесах, таким чином, область їхніх залучень практично необмежена (на відміну від промислових). МП є актуальними для промисловості, невиробничої сфери, залучаються під час проведення вибухотехнічних робіт та операцій, до того ж, при охороні важливих об'єктів.



Рисунок 1.1 – Мобільні платформи

До мобільних платформ висувають загальні вимоги, зокрема:

- активна рухливість і прохідність з урахуванням міських умов, усередині будівель та споруд, у зонах руйнувань, на пересіченій місцевості, як на твердих гладких покриттях, так і на деформованих ґрунтових;
- впевнено функціонувати як у невідготовлених природних умовах, так і середовищах, спеціально пристосованих для проживання людини (всередині будинків, у транспортних комунікаціях), вписуватися в міські транспортні потоки чи рухатися серед транспортних колон;
- конструкція платформи має сприяти її високій мобільності.

Для реалізації поставлених задач, виділимо такі основні групи МП:

- роботехнічна мобільна платформа;
- універсальні наземні платформи, призначені для функціонування на об'єктах транспорту, промисловості, міської інфраструктури тощо, а також відкритій слабкопересіченій місцевості;
- спеціальні робототехнічні комплекси, де МП здатні переміщатися

вертикальними та похилими поверхнями промислових об'єктів і транспортних засобів, до того ж у трубопроводах та вузьких місцях [2].

За масою (відповідно, і мобільністю) та головним призначенням МП можна розподілити на 4 групи:

- надлегкі (до 35 кг), (рисунок 1.2);
- легкі (до 150 кг);
- середні (до 800 кг);
- важкі (вагою понад 800 кг).



Рисунок 1.2 – Мобільна платформа МРК-01

Мобільна платформа МРК-01 (рисунок 1.2) спрямована для виконання інспекційних перевірок, пошуку та знищення вибухонебезпечних предметів. Оскільки її вага дорівнює 20 кг, вона належить до класу надлегких МРП з такими технічними характеристиками: габаритні розміри – 0,57 м × 0,48 м × 0,21 м; швидкість пересування – 0-2,5 км/г; трансмісія – електромеханічна; рух – колісний.

У світовій практиці найбільше уваги приділяється розробленню надлегких, легких і робототехнічних комплексів перших трьох груп. Такий попит характеризується не лише їхньою маневреністю, а й можливістю швидкої технічної адаптації щодо конкретного виду операції чи виконуваних робіт. Крім того, значною перевагою назвемо відносно невеликі матеріальні та економічні витрати на їхнє виробництво та експлуатацію [4].

Першопочатково закладений до конструкції більшості платформ модульний принцип дозволяє проєктувати багатофункціональні комплекси із залученням єдиної транспортної системи як ключової та формувати робочу систему під час встановлення робочого обладнання та необхідної системи управління.

Зважаючи на рівні незалежності у плані управління від участі людини-оператора, всі МП можна розподілити на великі групи відповідно до:

- автоматичного управління (автономні платформи);
- інтерактивного управління (телеоператори, телеплатформи).

З огляду на специфіку операцій, умови експлуатації, а також функціональне призначення мобільної платформи окреслюють її конструктивні особливості, ступінь складності системи управління (СУ), вагогабаритні критерії та найменування спеціального обладнання (таблиця 1.1).

Таблиця 1.1 – Класифікація мобільних платформ за призначенням

Група	Базові вимоги	Базовий варіант використання
Спеціального призначення	Компактність, безшумність	Автономне, один канал передачі відеоінформації
Військового призначення	Надійність, простота в управлінні, стандартне корисне навантаження	Як складова розвідувальних, ударних та охоронних комплексів, багатоканальні системи передачі різноманітних даних
Для екстремальних ситуацій, наукових досліджень, кінематографічного користування	Стійкість до несприятливих зовнішніх впливів, універсальність по відношенню до корисного бортового навантаження	Автономне, багатоваріантність реалізації каналів передачі даних
Для спортивних, промислових та побутових призначень	Простота в управлінні, економічність, надійність	Автономне, один канал передачі відеоданих

З метою реалізації процесів без участі людини складовою платформ має бути штучний інтелект. Зважаючи на це, МП із автоматичним управлінням, на відміну від інших, у джерелах інформації здебільшого трактуються як автономні. Автономні МП – це платформи, котрі можуть самостійно (з врахуванням сигналів від датчиків) приймати рішення в середовищі, що змінюється. Зрідка їх також іменують самонавчальними мобільними платформами.

МП з інтерактивним управлінням – це платформи, що здатні працювати як в автоматичному режимі, так і під управлінням людини-оператора. На противагу біотехнічним системам (із ручним управлінням), інтерактивні системи облаштовані пристроями пам'яті для автоматичного виконання певних дій [4].

Для промислових та сільськогосподарських призначень більш доцільно застосувати термін «моніторинг». Для завдань такого класу характерними будуть відносна незмінність району, а також залучення штатного складу корисного навантаження. Разом з цим слід забезпечити зберігання та опрацювання великих обсягів різноманітної інформації, що надходить. Це, власне, реалізується стаціонарними пультами управління, що виконують перелічені функції.

1.2 Пристрої управління мобільними платформами

Умови навколишнього середовища не завжди є сприятливими для виконання тієї чи іншої дії безпосередньо людиною. У такому разі для застосування у надзвичайних ситуаціях для допомоги людині задіюють мобільні платформи.

МП налічує низку сенсорів для сприйняття довкілля, низку виконавчих пристроїв (ВП) для впливу на середовище та систему управління, що наділяє платформу змогою реалізовувати цілеспрямовані та корисні дії. На рисунку 1.3 зображено функціональну схему платформи.

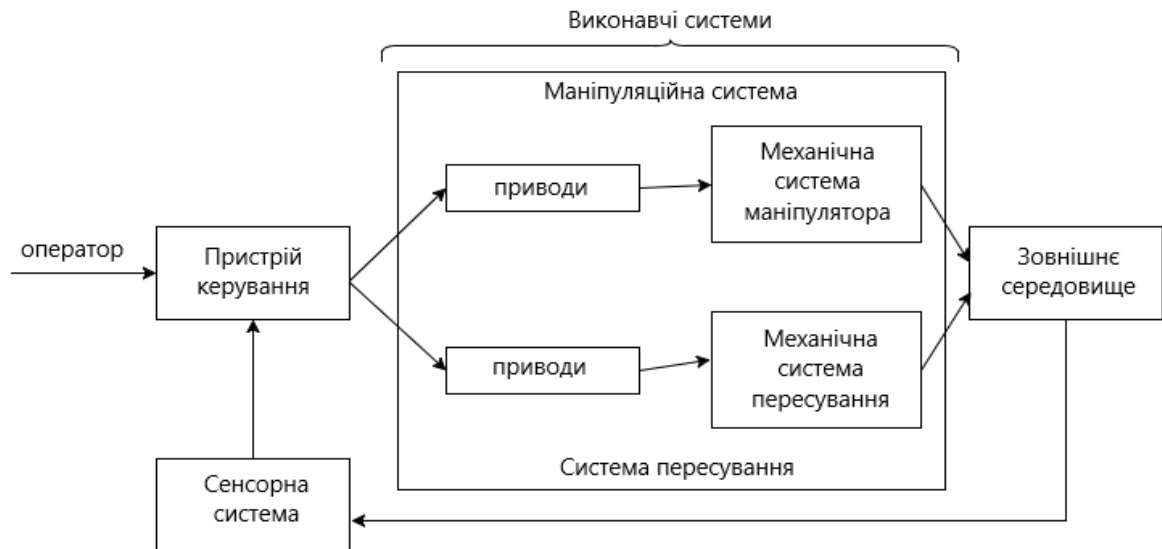


Рисунок 1.3 – Функціональна схема платформи

Її складниками є виконавчі системи – маніпуляційна (один або кілька маніпуляторів) та пересування, за умови, що платформа рухається, сенсорну систему, що забезпечує інформацією про довкілля, та пристрій управління. Зі свого боку до складу виконавчих систем належать механічні та системи приводів. Механічна система маніпулятора це, як правило, кінематичний ланцюг, який містить рухомі ланки з кутовим або поступальним переміщенням, у кінці котрого розташований робочий орган у вигляді захватного пристрою чи якого-небудь інструменту.

Залучення МП дозволяє виключити загрозу для життя та здоров'я людини-оператора. Зважаючи на це, проблема щодо створення мобільних платформ, які здібні до самостійного пересування та автоматичної реалізації поставлених завдань, є актуальною. Крім того, важливу роль відіграє розроблення системи навігації, завдяки якій можна скласти карту навколишнього середовища, в якому задіяна МП, планувати маршрут, котрий приведе до цілі та обходу перешкод, які з'являються на шляху.

На сьогодні управління платформою на рівні рухів, здебільшого, виконує людина-оператор, до того ж від фахівця вимагається безперервне спостереження за платформою та, водночас, оперативне управління її діями. Такий підхід характеризується нездатністю приймати самостійні рішення

платформою. Крім того, має низку недоліків, серед яких: необхідність організувати та постійно підтримувати канал зв'язку з людиною-оператором (на кшталт кабельного чи радіозв'язку). Це суттєво обмежує область використання платформи. [5].

Під час реалізації технологічних операцій оператор, отримавши від системи машинного зору інформацію про об'єкт та робочий процес, безперервно відбувається ручне управління виконавчими механізмами маніпулятора та транспортного засобу. Нелегкий процес управління, синтезуючись із характером виконуваних робіт, які потребують зосередженої уваги та обережності, спонукає до швидкої втоми оператора і, як наслідок, ймовірність помилкових дій може збільшуватись. До того ж, людина не може завжди правильно оцінити середовище за даними телеметрії та проводити адекватне управління. Перелічені недоліки можна усунути, якщо управління виконуватиметься людиною-оператором не лише на рівні завдання окремих рухів, але й на рівні окреслення мети. За таких умов платформа повинна самостійно (чи за незначної участі людини) реалізовувати окреслені завдання.

На рисунку 1.4 продемонстровано узагальнену структуру системи управління чутливих платформ, до яких належать і досліджувані платформи з адаптивним управлінням. Така СУ містить п'ять рівнів.

Координація дій між людиною-оператором і платформою виконується за допомогою пульта. Оператор окреслює платформі завдання, контролює процес виконання, а також здійснює загальний контроль за функціонуванням платформи в цілому.

П'ятий (верхній) рівень автоматичного управління аналізує завдання, внесені людиною-оператором, і визначає послідовність дій платформи для виконання завдання, отже, планує дії платформи. Іншими словами, спершу здійснюється аналіз інформації про зовнішнє середовище, що надійшла від сенсорної системи, а потім поєднуються моделі, на базі яких реалізується планування дій платформою.

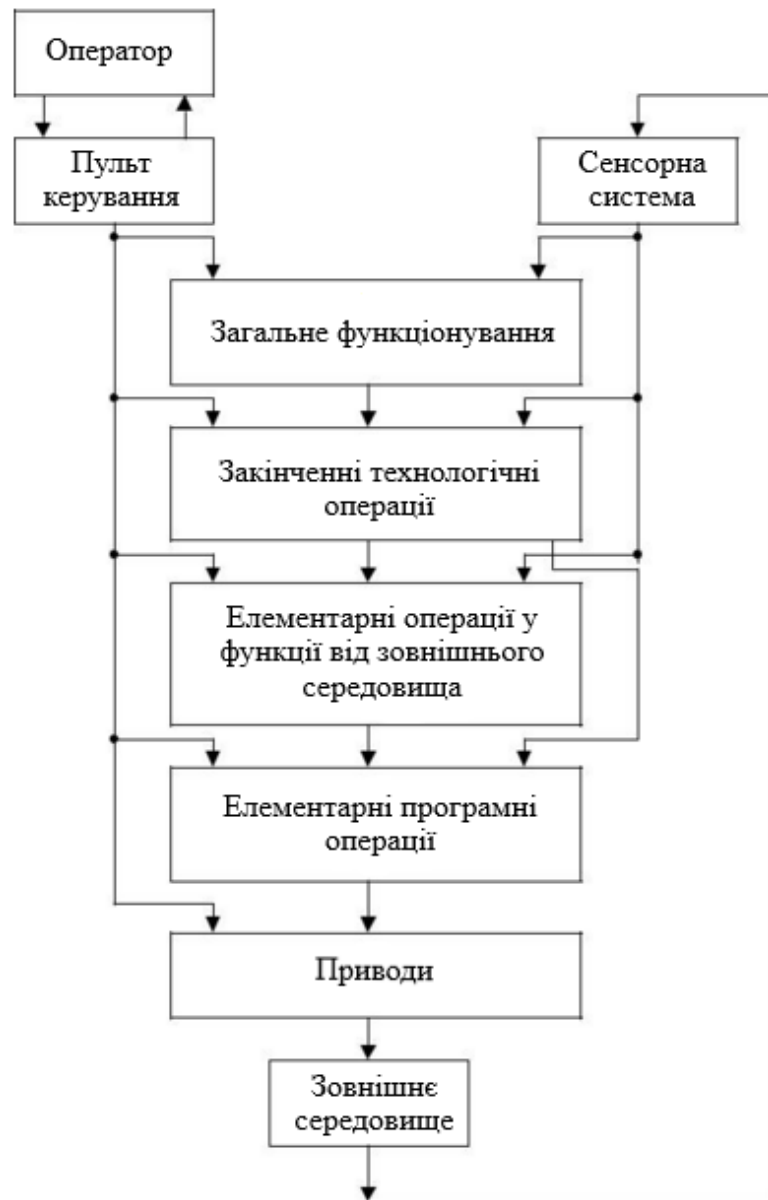


Рисунок 1.4 – Узагальнена структура системи управління платформою

Четвертий рівень управління полягає у сполученні функціонально закінчених складних дій, за підсумками яких розв’язується конкретна задача, на кшталт збирання якого-небудь виробу. Згідно з планом, попередньо опрацьованому на вищому рівні, здійснюється його розбиття на послідовність елементарних характерних операцій, які виконуються нижніми рівнями управління.

Третій та другий рівні управління визначаються як рівні функціонування елементарних операцій, на які можуть спрямовуватись

закінчені дії платформи. Проте відмінність між ними полягає в тому, що на 3 рівні поєднуються адаптивні управління у функції від інформації про зовнішнє середовище, а от на 2 рівні – найпростіші управління за програмою. Зважаючи на це, під час синтезу управлінь на 3 рівні разом із типовими програмами рівня 2 задіяні команди на вхід 1 рівня паралельно з впливами управління з виходу 2 рівня.

Таким чином, завдання, котре надійшло на вхід 3 рівня, виконується, по-перше, як послідовність типових програм другого рівня, а по-друге, як сукупність впливів управління безпосередньо на окремі приводи 1 рівня. Всі ці дії загалом задаються та узгоджуються з рівнем у залежності від поточної інформації про зовнішнє середовище та власне стан платформи.

На 2 рівні окреслюються впливи управління, що потім надходять на 1 рівень, який реалізує програмне управління приводами. Нижній, 1, рівень управління функціонує за окремими ступенями рухливості платформи і трактується як система управління приводами [6].

До пристроїв, які відповідають за реалізацію ключових функцій управління належать механізми управління переміщеннями робочих органів, перемикання, гальмування, синхронізація, включення, відключення, регулювання частоти та швидкостей.

Перші системи забезпечення навігації платформ були розроблені на базі сканувальних датчиків, зокрема, телебачення та локаційних номерів. Спеціальна обчислювальна схема платформи зводила електричні сигнали до аналогів різних перешкод і робила висновок про доцільність того чи іншого руху. Стандартними критеріями перешкод, які сприймаються платформою, назвемо стіну, навіс, яму-обрив, нахил, небезпеку для далекоміра та інші спрощені або збільшені деталі площини.

Інтелектуальними системами управління йменується підкатегорія СУ, що є варіантом ще більшої модернізації. У даній категорії з'явилась можливість зворотного спілкування з людиною, самонавчання, планування та перепланування поведінки, навігація, взаємодія з іншими платформами,

інструментами та обладнанням. На рисунку 1.5 продемонстровано узагальнену схему системи інтелектуального управління платформою (СІУП). Підкреслимо, що вона є конкретизацією загальної схеми управління чутливої платформи на рисунку 1.4.

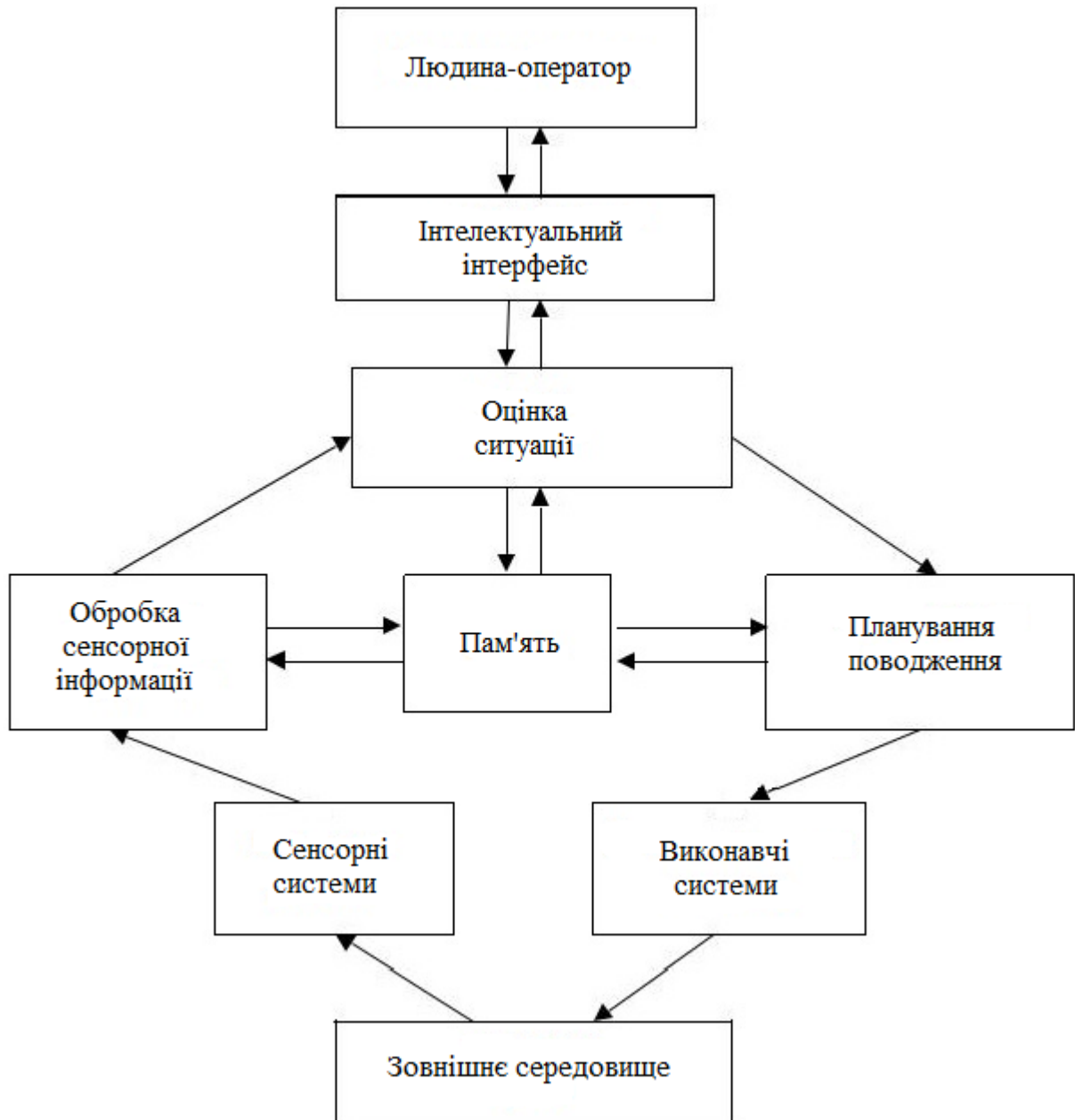


Рисунок 1.5 – Схема системи інтелектуального управління платформою

У центрі схеми розташовано блок пам'яті, котрий двосторонньо пов'язаний з іншими системами, що опрацьовують інформацію. Крім того, складовою цього блоку є база знань щодо зовнішнього середовища, тобто

ієрархічна модель зовнішнього середовища, а також база даних про власне платформу й операції, що вона має реалізовувати. Разом з тим спеціалізовані оперативні бази знань та даних, які пов'язані з центральною пам'яттю, можуть існувати і в окремих системах платформи [7].

Затребуваний рівень адаптації та штучного інтелекту системи окреслюється ступенем невизначеності та складності місцевості та характером розв'язання завдань, вншими словами, призначенням платформи. В останньому випадку головне значення має мінімально необхідний ступінь автономності управління платформою, що окреслюється переліком її дій, які повинні реалізовуватись без участі людини-оператора. До того ж, враховується і перспектива тимчасової втрати зв'язку з оператором, а також обмежена пропускна спроможність каналів зв'язку, неповна та обмежена достовірність інформації, що надходить оператору від платформи, разом із її затримкою в часі.

Так, через великий рівень перешкод у каналі передачі оператору іноді не видно зображення панорами місцевості перед платформою. Зважаючи на те, що оператор лише епізодично отримує цю інформацію, він має можливість втручатися в автономний рух платформи лише з метою контролювання та коригування маршруту руху, задіюючи цілевказівки окремих проміжних точок траси в моменти її достовірного огляду. [8].

1.3 Реалізація системи управління мобільними платформами

Система управління мобільної платформи взаємодіє з людиною-оператором шляхом активного діалогу, під час якого людина виконує такі дії щодо:

- створення робочої програми, котру можна представити як набір даних, який описує точки позиціонування захватного пристрою платформи та сигнали управління на технологічні пристрої чи як набір інструкцій проблемно-орієнтованою мовою;

- редагування робочої програми послуговуючись програми-редактора даних або редактора тексту, оскільки, як зазначали вище, програма може мати вигляд як даних, так і інструкції;
- створення об'єктного та завантажувального модулів робочої програми, забезпечення видалення старих файлів, додавання нових, перейменування та зберігання програм у бібліотеці;
- налагодження робочої програми, іншими словами, за допомогою програмного забезпечення реалізує її покрокове застосування, аналізує результати налагодження, а також у разі задовільної якості програми подає команду на її виконання;
- реалізації функції контролю справності обладнання, тобто перевіряє канали зв'язку з технологічними пристроями, калібрує вимірювальні системи платформи та здійснює інші операції діагностування [9].

Одним із найпопулярніших робототехнічних комплектів у світі вважається LEGO Mindstorms NXT, який було розроблено спільно зі співробітниками Массачусетського технологічного інституту. LEGO Mindstorms NXT – це великий набір конструктивних деталей (більше 600 од), налічує 3 сервоприводи, набір датчиків та мікрокомп'ютер NXT Brick. У світі комплект застосовують під час проведення багатьох робототехнічних заходів, у тому числі змагань WRO (World Robotic Olympiad). Разом із комплектом поставляється середовище візуального програмування NXT-G, розроблене компанією National Instruments на базі власного продукту LabView.

Отже, комплект NXT – це зручний полігон для реалізації на його основі комплексу всіляких робіт під час дослідження робототехніки. [10].

Зовнішній вигляд мікрокомп'ютера NXT Brick зображено на рисунку 1.6. Живлення надходить від 6 акумуляторів формату AA. Блок обладнаний РК-дисплеєм з можливістю відображення тексту та графіки. Для переміщення розділами меню застосовують клавіші на лицьовій панелі: помаранчева – ОК, темно-сіра – скасування. Крім того, блок може

відтворювати звуки не лише із заздалегідь записаних файлів, але і різної тональності. Під'єднання до комп'ютера реалізується інтерфейсом USB.

До того ж наявний модуль зв'язку Bluetooth, який дозволяє як управляти блоком із комп'ютера, так і поєднувати блоки один з одним. NXT Brick обладнаний трьома портами для приводів, які позначені літерами А, В і С (для руху, здебільшого застосовують В і С, тоді як А – для маніпулятора), а також чотирма портами для датчиків, позначених цифрами.

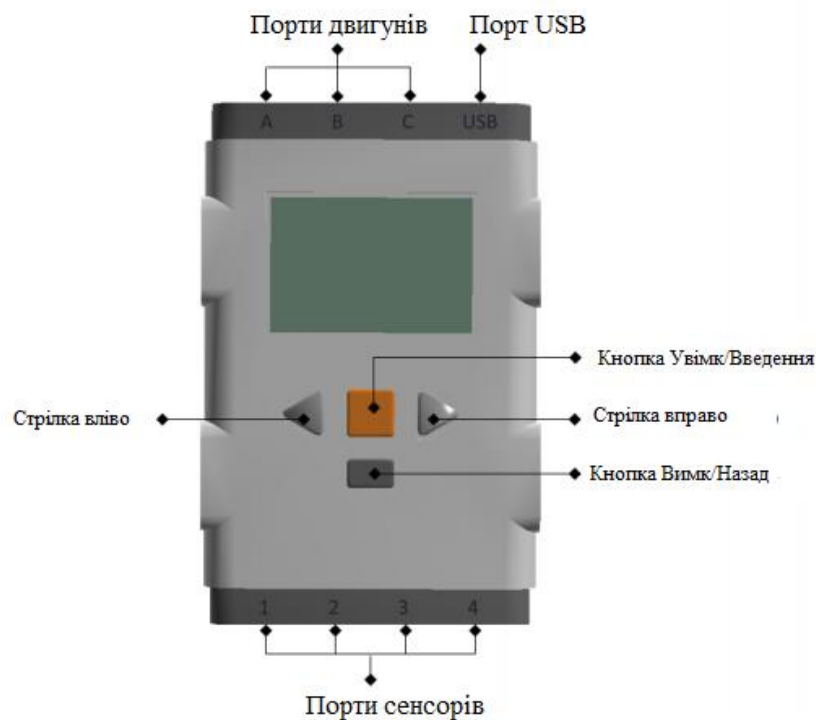


Рисунок 1.6 – Зовнішній вигляд мікрокомп'ютера NXT Brick

За основу конструктора Lego Mindstorms NXT 2.0 було взято програмований блок NXT Brick (ще його називають «цеглою»), інтерактивні сервомотори, а також кілька датчиків. Крім того, іншими складовими комплекту є:

- порт 1,2 – датчик торкання;
- порт 3 – датчик кольору;
- порт 4 – ультразвуковий сенсор;
- порт А – мотор для додаткових функцій;

– порт В,С – мотор для руху.

У версії конструктора для навчання також наявний датчик звуку, що вимірює рівень гучності звуку, а RGB-датчик замінено на датчик освітленості. До того ж є інші датчики, що випускаються окремо від комплекту, на кшталт гіроскопічного, датчиків для вимірювання температури, компас тощо.

Усі датчики та мотори під'єднуються до NXT-блоку через порти входу та виходу чорними шестипровідними кабелями. Сенсори під'єднуються до вхідних портів під номерами 1-4, а мотори – до вихідних портів з назвами А, В, С на блоці NXT. Виробник рекомендує застосовувати стандартні порти для під'єднання сенсорів та моторів.

Для програмування NXT Brick можна застосовувати середовище візуального програмування NXT-G. На рисунку 1.7 зображено, що вікно програми розбито на кілька частин.

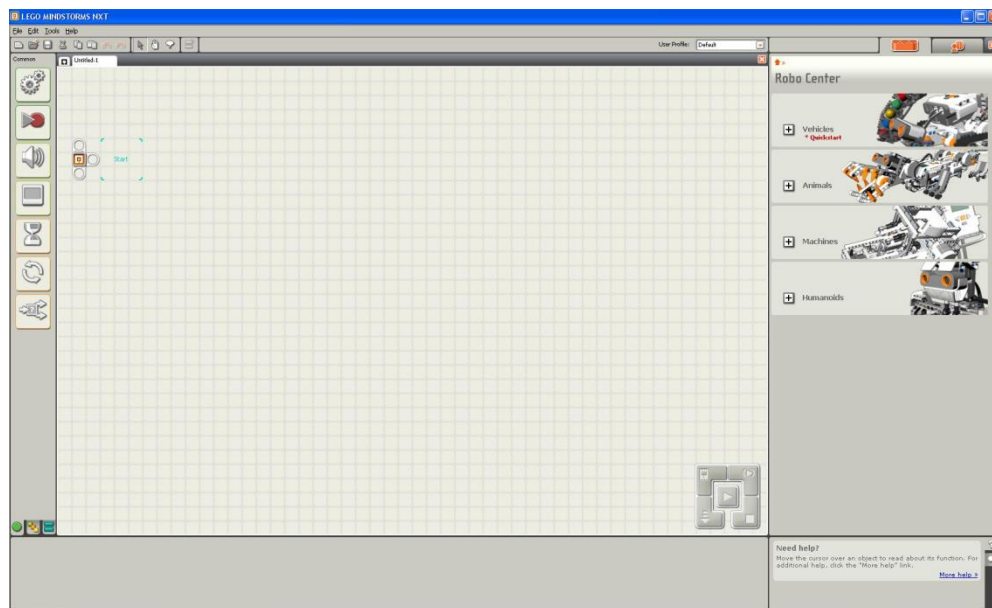


Рисунок 1.7 – Інтерфейс середовища програмування NXT-G

Центральною, найбільшою частиною, називається робоче поле програми. Ліворуч від неї розташована панель інструментів або команд, яким можна задати роботу. Внизу, тимчасово порожня, частина програми є рядком

критеріїв, де можна змінювати параметри команд. Праворуч розташовані вже готові інструкції для збирання стандартних конструкцій, які можна залучати для навчання азам конструювання та програмування. Крім того, у верхній частині програми наявне звичайне контекстне меню [11].

З метою ефективної участі людини для управління платформою, а також для спільного виконання робіт важливе значення приділяється організації взаємодії між платформою та людиною-оператором. Пристрої зв'язку розподіляються на дві групи: пристрої оператора, що задаються, а також пристрої відображення інформації, що надходить від платформи (рисунок 1.8).

До пристроїв, які задаються, належать командні пульти, набірні панелі, механічні аналоги виконавчих органів та їхні елементи, мовні командні пристрої, пристрої цілевказівки, стандартні термінали ЕОМ.

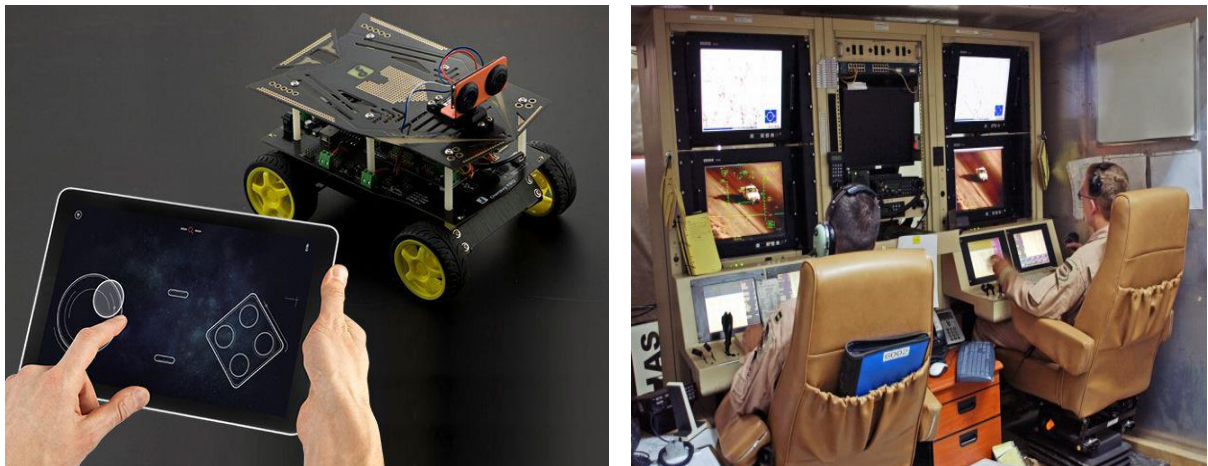


Рисунок 1.8 – Пристрої зв'язку

Пристрої цілевказівки застосовують для визначення в робочій зоні платформи об'єктів-цілей та вимірювання їхніх просторових координат. З такою метою можуть застосовуватись різного роду далекоміри (ультразвукові та лазерні), втім обов'язково забезпечені пристроями, що дозволяють оператору контролювати візуально наведення далекоміра на вибраний об'єкт.

Для переміщення нерівними поверхнями, кам'янистою місцевістю чи по траві розробляються шестиколісні платформи зі значно більшим зчепленням на відміну від чотириколісних. Ще більшому зчепленню сприяє застосування г гусениць. Багато сучасних платформ військового призначення, а також платформи для переміщення грубими поверхнями проєктуються як гусеничні. Втім використання подібних платформ у приміщеннях, на гладких покриттях та килимах ускладнюється. [9].

На сьогодні проведено значну кількість досліджень, які пов'язані з розробкою алгоритмів управління. Вони забезпечують виконання простих операцій шляхом застосування мобільних платформ. Приміром, це може бути планування траєкторій для обходу перешкод, уточнення карти місцевості, проникнення до важкодоступних зон, прецизійний рух складними криволінійними траєкторіями із дотриманням заданої орієнтації елементів платформи в просторі тощо. Подальші дослідження щодо нових типів мобільних платформ умотивовані численними застосунками в різних галузях людської діяльності, зокрема, автоматизація управління рухом транспортних засобів, пожежогасіння, телехірургія, розмінування підозрілих предметів тощо [9].

Для управління мобільними платформами все частіше послуговуються дистанційними приладами, як от пульти управління. Пульт дистанційного управління (ПДУ) позначає електронний пристрій для дистанційного управління іншим пристроєм на відстані людиною. Іншими словами, ПДУ – це частина дистанційного управління об'єкта, апаратами та механізмами на мобільних об'єктах (літаки, космічні кораблі, судна тощо). Крім того, ПДУ застосовують для управління системами зв'язку, виробничими процесами, технікою підвищеної небезпеки. Конструктивно пульт, здебільшого, має вид невеликої коробка, в середині якої розташовані електронна схема, кнопки управління та (як правило) джерело автономного живлення (рисунок 1.9).

З метою ефективного функціонування інтелектуальні платформи облаштовані системою сприйняття зовнішнього середовища, засобами

аналізу ситуацій та прийняття рішень. До того ж втілюють планування руху (зокрема й побудову траси).

Підкреслимо, що доволі вагомою є теоретико-механічна складова задачі, що ґрунтується на дослідженнях загальних глобальних властивостей системи управління (наявність, стійкість та розгалуження рішень диференціальних рівнянь руху платформи, їхні геометричні властивості тощо). Зважаючи на це, завдання управління та конструювання координуються із власними «натуральними» властивостями системи, коли можна застосовувати особливості природи, а не йти проти них. Усвідомлення механічної природи, стійкості чи нестійкості того чи іншого стаціонарного руху платформи є вагомим для вирішення завдань стабілізації таких рухів.



Рисунок 1.9 – Пристрої управління платформами

Теоретико-механічні дослідження ґрунтуються, як правило, на знаннях інтегралів некерованого руху та властивостей, вільних рухів мобільної платформи. Це сприяє можливості побудови «природних» режимів управління, що забезпечуватимуть оптимізацію енергетичних витрат під час руху системи. До того ж, мобільні платформи розглядаються як мехатронні системи, в яких електромеханічні процеси інтегруються з інформаційними. Проблеми управління та динаміки мехатронних систем потребують розробки нових підходів, які ґрунтуються на ефективному застосуванні комп'ютерних

методів (систем аналітичних обчислень, чисельних експериментів зі складними математичними моделями, комп'ютерної анімації, візуального моделювання). Окреслені підходи значно розширили можливості дослідження та призвели до появи результатів, які фактично отримати неможливо, якщо залучати тільки аналітичні методи. [9].

Мобільна платформа рухається з метою розв'язання тих чи інших завдань, опрацьовує дані із зовнішніх датчиків, і повинна регулярно обробляти інформацію, щоб управляти власним рухом.

2 ВІЗУАЛЬНЕ УПРАВЛІННЯ МОБІЛЬНИМИ ПЛАТФОРМАМИ

2.1 Системи машинного зору мобільних платформ

У всьому світі й досі є актуальною діяльність із створення промислових роботів та спеціальних мобільних робототехнічних платформ. Утім, системи управління цими платформами наразі перебувають на попередньому рівні: головною складовою системи управління так і залишається людина-оператор з пультом у руках, водночас пульт й досі складається з традиційного монітора, тумблерів і джойстиків. Отже, всі переваги та недоліки цих систем – звідси. Дана проблематика пов'язується з відсутністю коштів, які могли б значно полегшити роботу оператору, а в певних ситуаціях і зовсім замінити його.

Системами машинного зору здебільшого визначають сукупність апаратних і програмних засобів надходження, передавання, зберігання й оброблення зображень разом зі шлейфами даних. Крім того, сюди належить автоматичний аналіз зображень та формування рішень (зокрема, вимірювання параметрів об'єктів на зображенні) чи продукування сигналів управління, що подаються на виконавчі механізми (наприклад, під час організації руху мобільних платформ і маніпулюванні об'єктами, що відстежуються). Шлейф даних трактується як структурований безперервний потік даних від усіх бортових сенсорів мобільної платформи та команди управління, що організуються за певними правилами в деяку послідовність [10].

Машинним зором називають науковий напрямок у галузі штучного інтелекту, до того ж, і робототехніки, та пов'язані з ним технології щодо отримання зображень об'єктів реального світу, їхнього оброблення та застосування отриманих даних для розв'язання різноманітних прикладних задач без участі людини (повної або часткової).

Машинний зір тісно взаємодіє з такими напрямками, як оброблення та аналіз зображень, розпізнавання образів тощо. До того ж, відсутнє стандартне трактування того, яким чином має розв'язуватись проблема даної області. Разом з тим іноді буває складно однозначно розподілити задачі та методи, котрими послуговуються для розв'язання, до однієї з цих областей. Якщо проаналізувати технічні прийоми, алгоритми, методи оброблення зображень, які застосовуються та розробляються в зазначених областях, можна стверджувати, що вони – більш менш ідентичні.

Машинний зір, здебільшого, зосереджується на промисловому використанні, на кшталт автономних платформ і систем візуальної перевірки та вимірювань. Мова йде про те, що технології датчиків зображення та теорії управління, пов'язані з обробленням відеоданих для управління платформою, а також оброблення отриманих даних у реальному часі, реалізуються програмно чи апаратно.

Оброблення й аналіз зображень, здебільшого, спрямовано на роботу з 2D-зображеннями. Іншими словами, перетворення одного зображення на інше за допомогою попіксельних операцій збільшення контрастності, операцій з окреслення країв, усунення шумів або геометричні перетворення, наприклад, обертання зображення. Згадані операції припускають, що оброблення й аналіз зображення реалізуються незалежно від самих зображень [10].

Машинний зір зосереджується на обробленні тривимірних площин, спроектованих на одне чи декілька зображень. Як от відновлення структури чи іншої інформації про 3D-площину за одним або декількома зображеннями. Машинний зір, здебільшого, залежить від більш-менш складних припущень з огляду на те, що представлено на зображеннях.

До того ж, є область, іншими словами, візуалізація, що спершу належала до процесу створення зображень, проте іноді залучалась для оброблення й аналізу. Так, рентгенографія взаємодіє з аналізом відеоданих медичного застосування.

На додаток, розпізнавання образів вважається областю, що послуговується різними методами отримання інформації з відеоданих, як правило, закладених на статистичному підході. Значна частина даної галузі спрямована на практичне застосування таких методів.

Таким чином, отримуємо невтішний висновок, що поняття «машинний зір» наразі складається з: комп'ютерного зору, розпізнавання зорових образів, аналізу й обробки зображень тощо (рисунок 2.1).



Рисунок 2.1 – Аналіз машинним зором навколишнього середовища

Системи машинного зору (СМЗ) знайшли своє призначення в авіаційних, космічних, наземних, надводних та підводних мобільних пристроях, іншими словами, там, де необхідний аналіз навколишнього середовища чи оточення в режимі реального часу. Активний розвиток сучасних технологій у мікроелектроніці посприяв появі мініатюрних та енергетично економічних матричних і лінійних приймачів оптичного випромінювання. Прилади, що наділені високою чутливістю та достатньо

високий рівень просторової і колірної роздільної здатності (до того ж, в інфрачервоній області спектру), СМЗ, створені на базі таких твердотілих датчиків відеоінформації. Крім того, є незамінними джерелами інформації у разі створення мобільних систем із супервізорним управлінням [10].

З метою класифікації методів та підходів, які застосовують у системах машинного зору, власне зір розподілили на три базові підкласи: низького, середнього та високого рівнів. Системи машинного зору низького рівня придатні для оброблення інформації з датчиків відчуття.

Ці системи можна класифікувати як «інтелектуальні» машини, за умови, що їм притаманні такі ознаки (ознаки інтелектуальної поведінки):

- спроможність визначати суттєву інформації з безлічі незалежних ознак;
- здатність формулювати цілі та окреслювати плани для досягнення цих цілей;
- здібність до навчання на прикладах та узагальнення таких знань з можливістю відновлювати події, ґрунтуючись на неповній інформації.

Створення систем машинного зору з такими характеристиками для обмежених видів робочого простору, фактично, є реальним, однак параметри таких систем є далекими від можливостей зору людини. Так, за основу машинного зору було обрано аналітичну формалізацію, що спрямована на розв'язання конкретних задач. Машини з сенсорними параметрами, наближеними до можливостей людини, напевно, з'являться не скоро. Втім підкреслимо, що копіювання природи – це не єдиний шлях для розв'язання такої проблеми.

Дійсно, масове використання методів машинного зору відбулось менше десяти років тому, цьому посприяло досягнення відповідного рівня продуктивності процесорів у персональних та мобільних комп'ютерах. Зважаючи на це, у плані практичного використання системи машинного зору подолали низку етапів: етап індивідуального розв'язання (як у питанні апаратного забезпечення, так і алгоритмів) конкретних задач; етап

використання у професійних галузях (перш за все у промисловості та оборонній сфері) за допомогою спецпроцесорів, спеціалізованих систем створення зображень, а також алгоритмів, призначених для роботи в умовах низької апріорної невизначеності. Втім такі рішення допускали масштабування та етап масового використання [10].

Система машинного зору складається з таких головних компонентів:

- підсистеми створення зображень, яка сама може містити різні компоненти, зокрема, об'єктив і прилад із зарядним зв'язком;
- обчислювача;
- алгоритмів аналізу зображень, які можуть програмно функціонувати на процесорах загального призначення, апаратно як складова обчислювача і навіть апаратно в межах підсистеми створення зображень.

Найпоширенішого використання набувають системи машинного зору, що задіюють стандартні камери (рисунок 2.2) та комп'ютери як перші два компоненти (саме до таких систем більш доцільно застосовувати термін «комп'ютерний зір», хоча чіткого поділу між поняттями машинного та комп'ютерного зору немає).



Рисунок 2.2 – Прилади машинного (комп'ютерного) зору

Втім, певна річ, інші системи машинного зору теж менш важливі. Тому вибір «нестандартних» способів створення зображень значно збільшує можливості систем машинного зору. Проте за можливостями алгоритмічного

забезпечення СМЗ значно поступаються зору людини, хоча за можливостями отримання інформації про об'єкти, котрі відстежуються, вони значно перевершують його.

Одним серед вагомих застосувань вважається оброблення зображень у медицині. Дана область визначається отриманням інформації з відеоданих з метою встановлення медичного діагнозу пацієнтам.

Так, відеодані, здебільшого, отримують шляхом застосування мікроскопії, ангиографії, рентгенографії, томографії й ультразвукових досліджень. Наприклад, за інформацією, котра може надходити з таких відео, виявляють пухлини, атеросклероз або інших злоякісні зміни. До того ж, за приклад візьмемо вимірювання розмірів органів, кровотоку тощо. Зазначена прикладна область сприяє, до того ж, медичним дослідженням, подаючи нову інформацію, на кшталт про будову мозку чи якість медичного лікування [10].

Іншою прикладною областю машинного зору назвемо промисловість, для котрої отримують інформацію з метою підтримки виробничого процесу. Як приклад наведемо контроль якості, тобто деталі або кінцевий продукт автоматично перевіряються на наявність дефектів або вимірювання положення та орієнтації деталей, що піднімаються рукою робота.

Одними серед новітніх галузей використання стали автономні транспортні засоби (ТЗ), зокрема, наземні, підводні та повітряні. Рівень автономності змінюється від повністю автономних (безпілотних) (рисунки 2.3) до транспортних засобів, у яких системи, що ґрунтуються на машинному зорі, підтримують водія чи пілота в різних ситуаціях. Повністю автономні ТЗ послуговуються машинним зором для навігації з метою отримання інформації про місцезнаходження, для створення карти навколишнього середовища чи виявлення перешкод.

Їх можна застосовувати для певних задач, зокрема, виявлення лісових пожеж. Так, прикладами таких систем можемо навести систему попереджувальної сигналізації про перешкоди на машинах, а також системи автономної посадки літаків.



Рисунок 2.3 – Безпілотні засоби бачення місцевості

Функціонування системи машинного зору залежить від області її застосування, апаратної платформи та вимог щодо продуктивності. Деякі системи є автономними і розв’язують специфічні проблеми детектування та вимірювання, водночас інші системи є складовими підсистеми більших систем, які вже можуть містити підсистеми контролю механічних маніпуляторів (платформи), інформаційні бази даних (пошук схожих зображень), інтерфейси людина-машина (комп’ютерні ігри) тощо. Втім є функції, характерні для багатьох систем машинного зору [10].

Цифрові зображення надходять від одного чи декількох датчиків зображення, котрі крім різних типів світлочутливих камер облаштовані датчиками відстані, радарами, ультразвуковими камерами тощо. Зважаючи на тип датчика, дані, що виходять, можуть бути звичайним 2D або 3D-зображенням, або послідовністю зображень. Значення пікселів, здебільшого, відповідають інтенсивності світла в одній або декількох спектральних смугах (кольорові чи зображення у відтінках сірого), втім можуть пов’язуватись із різними фізичними вимірюваннями, зокрема, глибиною, поглинанням або відображенням звукових або електромагнітних хвиль чи ядерним магнітним резонансом.

Перед тим, як методи машинного зору можуть бути задіяні до відеоданих з метою витягування певної частки інформації, відеодані

доцільно обробити, щоб вони задовольняли деяким умовам, в залежності від застосовуваного методу [10]. Наприклад:

- проведення повторної вибірки з метою переконання, що координатна система зображення – правильна;
- видалення шуму з метою позбавлення спотворень, внесених датчиком;
- поліпшення контрастності з метою виявлення потрібної інформації;
- масштабування з метою кращого розрізнення структур на зображенні.

Деталі зображення різного рівня складності виокремлюються із відеоданих. Стандартними прикладами таких деталей є:

- лінії, межі та кромки;
- локалізовані точки інтересу: кути, краплі або точки (складніші деталі можуть стосуватися структури, форми або руху).

На певному етапі оброблення приймається рішення щодо точок або ділянок зображення, що є важливими для подальшого оброблення, зокрема:

- виділення певного цікавлячого набору точок;
- сегментація однієї чи декількох ділянок зображення, що містять характерний об'єкт.

На етапі високорівневого оброблення вхідні дані, здебільшого, трактуються як невеликий набір даних, на кшталт набору точок або ділянки зображення, де, ймовірно, розташований певний об'єкт. Наприклад:

- перевірка, чи задовільняють дані умовам, які залежать від методу та використання;
- оцінка типових параметрів (положення чи розмір об'єкта);
- класифікація встановленого об'єкта за різними категоріями [10].

2.2 Програмне забезпечення систем машинного зору

Програмне забезпечення, зазвичай, робить кілька кроків для оброблення зображень. Нерідко спершу зображення обробляється з метою зменшення шуму чи конвертації безлічі відтінків сірого в просторі поєднання чорного і білого (бінаризації). Після початкового оброблення програма обчислюватиме, проводитиме вимірювання, а також визначатиме об'єкти, розміри, дефекти та інші характеристики зображення. Наприкінці, програма пропускає чи забраковує деталь відповідно до заданих параметрів. Якщо в деталі наявний брак, програмне забезпечення подає сигнал механічному пристрою для відхилення деталі; в протилежному разі, система може зупинити виробничу лінію та попередити працівника (людину) з метою розв'язання цієї проблеми, що призвела до невдачі. Зважаючи на те, що більшість систем машинного зору покладаються на «чорно-білі» камери, застосування кольорових камер – все більш поширене явище. До того ж, все частіше системи машинного зору послуговуються цифровими камерами прямого підключення, а не камери з окремим вихопленням кадрів, що скорочує витрати та спрощує систему [10].

Компонентами програмного забезпечення називають програми, подані документально або на машинних-носіях, а також експлуатаційні документи, на кшталт інструкцій для програмістів. Програмне забезпечення здійснює на САПР визначальний вплив, з огляду на те, що воно пов'язує в єдине ціле більшість інших видів забезпечення (технічного, лінгвістичного, математичного та інформаційного). Розроблення програмного забезпечення потребує основної частки матеріальних та тимчасових витрат для створення САПР. Зважаючи на це, особливо важливо керуватися загальними вимогами та принципами побудови програмного забезпечення.

Найкраща конструктивна та схематична побудова інформаційно-управляючих систем залишиться лише складною оболонкою, якщо не наділити її внутрішнім змістом – алгоритмічним та програмним

наповненням, спроможним забезпечити розв'язання необхідного класу задач управління, у тому числі й інтелектуальних. Шматок кремнію, оброблений на найсучасніших підприємствах електронної промисловості, що в результаті став мікропроцесором, у цілому є субстанцією досить інертною, а його чудове перетворення на «мозок» робота реалізується завдяки програмному забезпеченню, що і є рушійною силою управління. Органічною складовою алгоритмічного та програмного забезпечення вважаються мови програмування, котрі є не тільки і не стільки засобом представлення інформації та спілкування робота з оператором, скільки методом логічного аналізу інтелектуальних задач. Звичайні людські мови для цієї мети підходять погано, у зв'язку з тим, що не забезпечують суворості формалізації задач, точності та надійності їхнього логічного аналізу. До того ж, спроможність робота розпізнавати окремі слова для опису задачі доволі обмежена, тому потребує великого обсягу оперативної пам'яті для зберігання мовних даних. Зважаючи на це, мовою робота повинна стати спеціально створена формалізована мова. До того ж, мова програмування має відповідати тим задачам управління, що мають розв'язуватися за її допомогою [10].

Загалом можна визначити чотири рівні мови програмування платформ, відповідно до завдань управління – виконавчих приводів, маніпулятора, операцій та завдання. Застосовуючи мову нижчого виконавчого рівня, оператор може активувати рух окремих ступенів рухливості маніпулятора на кшталт певних значень лінійних або кутових переміщень ланок.

Мова рівня маніпулятора дає можливість управляти сукупним рухом у робочому просторі відносно довільної системи узгодження для здійснення необхідного переміщення та орієнтації робочого органу, не зважаючи на стан окремих ступенів рухливості та координацію їхнього переміщення.

Мова рівня операцій сприяє створенню робочої програми за допомогою вказівки лише послідовності операцій, які платформа повинна виконати над об'єктом маніпулювання.

Мовою найвищого рівня завдань стає програма дій платформи, що складається з номінацій «що зробити», тобто, у загальному виді без деталізації.

Наразі сформовано два підходи до програмування мобільних платформ:

- програмування, що орієнтоване на платформу;
- проблемно-орієнтоване програмування чи програмування на рівні задачі.

У роботоорієнтованих мовах, які зараз є ключовими в сучасних пристроях управління, за платформу описується послідовність її рухів. Платформою управляє програма протягом реалізації всієї задачі, до того ж, кожен крок програми приблизно відповідає одній дії платформи [11].

NXT-G вважається офіційною мовою, що входить до в пакета Lego Mindstorms, який надходить разом із конструктором. Вона є графічною мовою програмування, в якій програми можна реалізовувати за допомогою натискання клавіш миші, а також перетягування блоків коду на екрані. Разом з тим, платформа NXT-G містить докладні інструкції зі складання чималого ряду моделей роботів LEGO; додаткові інструменти для побудови зображень, редагування звуків та калібрування сенсорів; контролер для передачі, запуску чи зупинки створених програм, а також перегляду довідкової інформації про мікрокомп'ютер (ресурси пам'яті, характеристики зв'язку тощо) (рисунок 2.4).

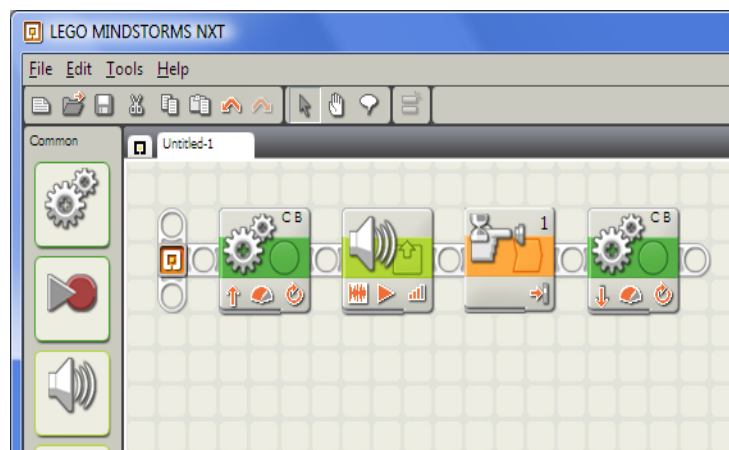


Рисунок 2.4 – Інструменти для створення дій мобільної платформи

RoboLab є багатофункціональним графічним середовищем програмування, що реалізується на базі LabView 7.0. Середовище програмування RoboLab дає можливість легко опанувати основи робототехніки та навчитися конструювати керовані машини. У RoboLab застосовується створена спеціальна версія середовища графічного програмування LabView фірми National Instruments з набагато простішими функціями та спрощеним інтерфейсом, що робить її доступною навіть для дітей (рисунок 2.5).

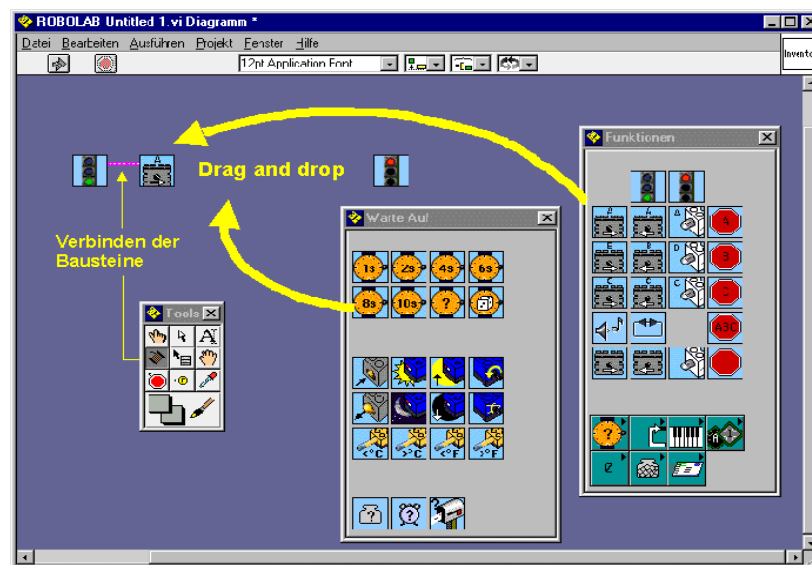


Рисунок 2.5 – Інтерфейс середовища програмування RoboLab

Зважаючи на те, що LabView разом з тим є потужним інструментом для вимірювання та контролю сигналів, а також сприяє отримуванню результатів різних вимірювань у реальному часі з кількох приладів, аналізуючи їх, то комплекс RoboLab може застосовуватись як засіб моделювання для розробників автоматизованих систем та роботів [11].

Microsoft Robotics Developer Studio (Microsoft RDS, MRDS) – Windows вважається орієнтованим середовищем для управління мобільними платформами чи роботами та їхніми симуляціями. Такий продукт корисний для академічного, аматорського чи комерційного розроблення та підтримує чималу кількість різноманітного апаратного забезпечення роботів.

Утім існують й особливості:

- мова візуального програмування Microsoft Visual Programming Language призначена для створення та налагодження програмних застосунків роботів;
- Web- та Windows-орієнтовані інтерфейси (рисунок 2.6);
- симуляція 3D (у тому числі апаратне прискорення);
- полегшений доступ до датчиків і виконавчих механізмів робота;
- підтримування кількох мов, зокрема, C#, Visual Basic.NET, JScript та IronPython [11].



Рисунок 2.6 – Інтерфейс середовища Microsoft Robotics Developer Studio

LEGO Digital Designer є програмою для створення різноманітних 3D-об'єктів, які ґрунтуються на віртуальних частинах конструктора LEGO. Тут є чималий набір найрізноманітніших деталей, у тому числі конструкції тонкого призначення, на кшталт залізничних рейок. LEGO Digital Designer складається з простого та зручного інтерфейсу, що дає можливість розібратися в управлінні створення моделей без особливих труднощів.

У даній роботі застосовано бібліотеку OpenCV (Open Source Computer Vision Library), яка до 1-ї версії створювалась в центрі розроблення програмного забезпечення Intel. OpenCV написана мовою високого рівня (C/C++), до того ж, складається з алгоритмів щодо: інтерпретації зображень, калібрування камери за зразком, усунення оптичних спотворень, визначення подібностей, аналізу переміщення об'єкта, визначення форми об'єкта, а також стеження за об'єктом, 3D-реконструкції, сегментації об'єкта, розпізнавання

жестів тощо. (рисунок 2.7). Розглянута бібліотека дуже популярна завдяки своїй відкритості та можливості безкоштовного користування як із навчальною, так і комерційною метою.

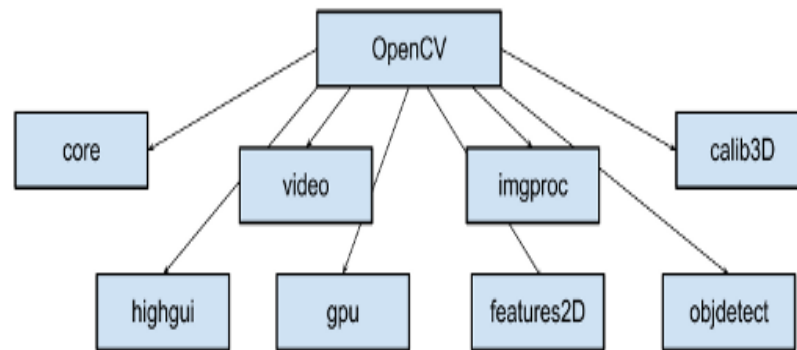


Рисунок 2.7 – Ключові модулі бібліотеки OpenCV

Ключові модулі бібліотеки можна розподілити між 3 групами (розділами):

- модулі `core`, `highgui`, задіюють базову функціональність (основні структури, математичні функції, генератори випадкових чисел, лінійну алгебру, швидке перетворення Фур'є, введення/виведення зображень та відео);

- модулі `imgproc`, `features2d` придатні для оброблення зображень (фільтрації, геометричних перетворень, перетворень колірних просторів, сегментації, виявленні особливих точок та ребер, контурного аналізу тощо);

- модулі `video`, `objdetect`, `calib3d` сприяють калібруванню камери, аналізу руху та відстеженню об'єктів, обчисленню положення у просторі, побудові карти глибини, детектуванні об'єктів, оптичного потоку) [12].

2.3 Системи візуального управління

Візуальні системи управління спрямовані на організацію зворотного зв'язку щодо відносного положення робочого органу платформи. Такі системи трактуються як засоби управління позиціонуванням, які отримують

й опрацьовують візуальну інформацію про робоче середовище.

Одним із вагомих напрямків розвитку робототехнічних систем завжди вважалась організація взаємодії маніпулятора та сенсорних аналізаторів. Застосування сенсорів наділяє систему гнучкістю, розширює її функціональні можливості, а також збільшує коло розв'язуваних нею задач. Більшість наразі існуючих систем, які реалізують взаємодію маніпулятора та сенсорів, побудовані за допомогою зв'язування автономних сенсорної та робототехнічної підсистем [12].

Експериментальною платформою для розроблення програмного забезпечення в нашій роботі стали WebCamera, виробничий об'єкт, який функціонує на базі мікрокомп'ютера NXT Brick, а також безпосередньо комп'ютер, до якого передається та зчитується інформація користувачем.

Отже, схема організації взаємодії пристрою управління (комп'ютера), пристрою, що забезпечує візуальне управління (WebCamera) та виробничого агента (мобільної платформи), подано на рисунку 2.8.

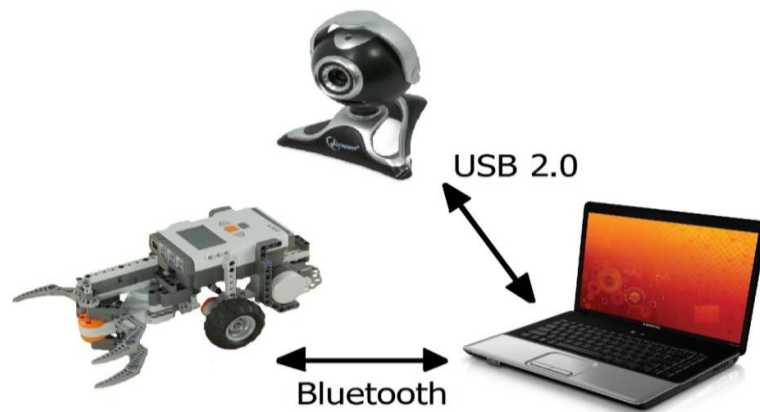


Рисунок 2.8 – Схема організації візуального управління

За допомогою системи машинного зору виконується аналіз дискретизованого робочого простору, аналізуються перешкоди та генерація маршруту автономного виробничого агента. Після ухвалення рішення починається його реалізація за допомогою системи програмного управління мобільною платформою.

Вказівки задач переміщення виробничого агента (мобільної платформи) реалізується візуальним способом, іншими словами, вказівками цільових об'єктів. Так, користувач за допомогою маніпулятора «миша» може вказувати початкову та кінцеву точки маршруту виробничого агента (платформи). До тогож, згідно з алгоритмами пошуку рішення агент рухається в дискретному робочому просторі, інформацію про який отримує за допомогою глобальної системи машинного зору. Аналіз інформації щодо стану робочого простору виконується за допомогою методу Кенні, а також завдяки аналізу контурів, а поточне положення виробничого агента в робочому просторі окреслюється шляхом застосування розробленого класифікатора типу Хаара [13].

Зважаючи на візуальне управління процеси обчислювального оброблення образу можна трактувати як вимірювання, що функціонує в кожному з положень робочого органу (рисунок 2.9).

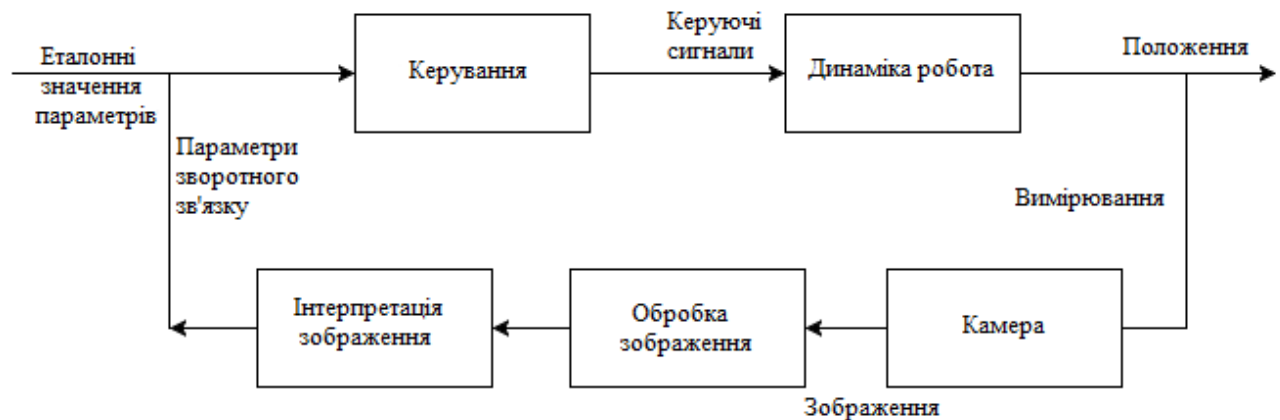


Рисунок 2.9 – Структурна схема системи візуального управління платформою

Вплив зазначених вимірювань на роботу системи управління із замкнутим циклом визначається таким чином:

- вимірювання долучають додаткові перешкоди в роботу системи;
- проведення вимірювань сприяє тимчасовим затримкам;

– до системи слід вводити еталонні значення критеріїв зворотного зв'язку, що використовуються.

Візуальні системи управління платформами забезпечують зворотний зв'язок щодо стану робочого органу. Вони послуговуються механізмом інтерактивного позиціонування, що ґрунтується на формуванні та обробленні візуальної інформації про навколишнє середовище. Позитивне розв'язання задач адаптивного візуального управління може суттєво покращити особливості системи управління мобільною платформою [12].

3 РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ УПРАВЛІННЯ МОБІЛЬНОЮ ПЛАТФОРМОЮ ЗА ДОПОМОГОЮ ВІЗУАЛЬНОГО ЦІЛКОВАННЯ

3.1 Опис алгоритмів системи візуального управління

Опрацюємо роботу алгоритму візуального управління мобільною платформою. Зазначений алгоритм реалізує функцію руху платформи на задану використовуючи засіб візуальної цілевказівки точки відповідно до розробленого плану переміщення. У разі розроблення алгоритму доцільно:

- розрахувати найкоротший шлях до цільової точки;
- зважати на розміри платформи за умов планування переміщень між перешкодами та об'єктами робочого простору;
- враховувати, що шлях повинен складатися з найменшої кількості змін напрямку, тому що поворот для платформи є окремою задачею, що споживає деяку кількість його «життєвих ресурсів» (тобто заряду акумуляторів), і займає багато часу;
- порівнювати масштаб карти з командами, котрі подаються на плату управління двигунами.

Першопочатково робота алгоритму полягає у перевірці, чи є працездатною камера системи машинного зору. Якщо камеру не знайдено, активізація програми закінчується. У разі її наявності, система отримує поточне зображення робочого простору. Далі з метою визначення характеристик об'єкта робочого простору слід використати пороговий метод та визначити межі об'єктів, залучивши метод Кенні.

Після окреслення меж об'єктів робочого простору програма визначає їхній контур, а також знаходить низку критеріїв об'єкта, наприклад, центри мас, периметри, площі.

Наявність інформації про робочий простір дає можливість візуальної

цілевказівки. Програма перевіряє умову розташування платформи у початковій точці (X_0, Y_0) . У разі відсутності платформи, реалізується трансформація системи координат щодо поточного положення платформи та перехід до наступної дії.

Якщо платформа наявна у початковій точці систем координат у робочому просторі, користувачеві рекомендується задати цільову точку. Якщо цільова точка не потрібна (чи її досягнуто), програма завершує своє функціонування (переривання циклу роботи системи машинного зору активується шляхом натисканням кнопки «X»). Якщо цільову точку (X_g, Y_g) необхідно задати, користувач повинен ввести її шляхом натискання правою клавішею миші. Далі проводиться перевірка умови – чи збігаються точки (X_0, Y_0) з точками (X_g, Y_g) . У разі збігу таких точок програма повертається до перевірки умови, чи потрібна цільова точка. У разі неспівпадіння виконується побудова маршруту в точках (X_g, Y_g) та його активізація за допомогою руху платформи. Після реалізації перелічених дій програма повертається до умови визначення, чи потрібна цільова точка (рисунк 3.1).

Областю функціонування алгоритму візуального управління є організація взаємодії системи управління мобільної платформи, маніпулятора та сенсорної системи. Застосування сенсорів наділяє систему гнучкістю, розширює її функціональні можливості, а також збільшує коло розв'язуваних нею задач. Більшість наявних наразі систем, які послуговуються взаємодією маніпулятора із сенсорами, створені шляхом зв'язування автономних сенсорної та робототехнічної підсистем. Дані системи продемонстрували актуальність концепцій взаємодії системи управління платформою із сенсорною підсистемою, втім у процесі їхнього розроблення були порушені ключові питання аналізу, проєктування та оцінки швидкодії таких систем. Структури, що діють за принципом «огляд – переміщення», базуються на внутрішньому замкненому циклі управління. Системи візуального стеження не наділені таким циклом. Зважаючи на це, управління маніпулятором застосовує лише інформацію сенсорного зв'язку [11-12].

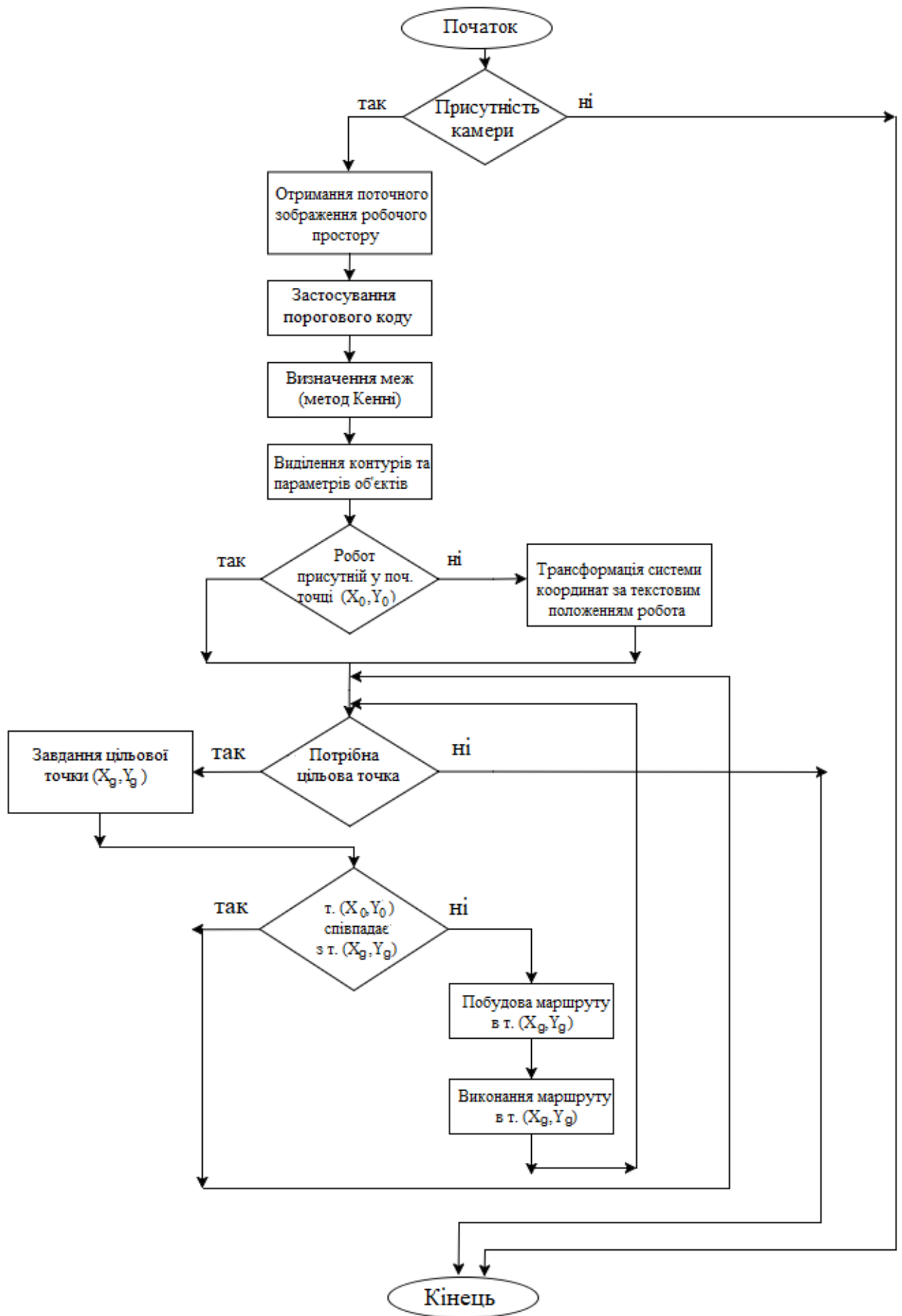


Рисунок 3.1 – Алгоритм візуального управління платформою

3.2 Опис функцій програмного забезпечення

Програмне забезпечення для візуального управління мобільною платформою функціонує за допомогою програмної бібліотеки OpenCV. Використання OpenCV у мові програмування Visual C++ не вважається надмірно складною, проте вимагає виконання певних стандартних кроків.

Перш за все, доцільно інсталювати бібліотеку OpenCV. До того ж, функції OpenCV потребують визначення розміщення заголовних (*.h) та бібліотечних (*.lib) файлів. З цією метою потрібно за допомогою меню Visual C++: Tools/Options/VC++ Directories визначити шлях до *.h та *.lib- файлів. Підкреслимо, що з'являється необхідність зазначити шляхи до всіх каталогів, у яких присутні *.h-файли, тобто до каталогів:

```

\OpenCV\otherlibs\highgui;
\OpenCV\otherlibs\cvCam\include;
\OpenCV\cxcore\include;
\OpenCV\cvaux\include;
\OpenCV\cv\include, так само \opencv\ Lib для бібліотечних
файлів.

```

Звертаємо увагу, що розроблення програми візуального управління ґрунтується на MFC-проекті, що має свої особливості. Так, MFC-проект складатиметься з двох класів: CApp (застосунок) і CMainWin (головне вікно) з відповідною першою функцією InitInstance(); конструктором і картою повідомлень для класу вікна.

Опрацюємо призначення ключових функцій програми. Функція OnCamera() трактується як обробник, який містить цикл обробки зображень, відстежувані системою машинного зору. Функція активізується з конструктора головного вікна, що модифікується у вигляді:

```

CMainWin::CMainWin(void)
{Create(NULL,"Adaptive decision-
maker",WS_OVERLAPPEDWINDOW);
.....
int resp=MessageBox("Do you want camera to use",NULL, MB_YESNO |
MB_ICONQUESTION);
if(resp==IDYES)
{
if(capture=cvCaptureFromCAM(0))
{ cvNamedWindow("mainWin",CV_WINDOW_AUTOSIZE);
cvNamedWindow("Contour",CV_WINDOW_AUTOSIZE);
cvSetMouseCallback("mainWin", on_mouse,this);//
OnCamera();
}
}
}
}

```

Власне функція OnCamera() трактується як нескінченний цикл, мета якого полягає в послідовному отриманні кадрів з камери системи машинного зору (СМЗ) та їхньому обробленні:

```

void CMainWin::OnCamera()
{.....
bool firstFrame=true;
while(1)
{CvMemStorage *storage=cvCreateMemStorage(0);
img=cvQueryFrame(capture);
cvCvtColor(img,dst2,CV_RGB2GRAY);

cvThreshold(dst2,dst3,1,255,CV_THRESH_BINARY_INV|CV_THRESH_
OTSU);
cvCanny(dst3,dst3,128,255);
cvShowImage("Canny",dst3);
x=cvWaitKey(10);
if(x=='x'){RobotCommand(2);break;}
}
cvReleaseCapture(&capture);
cvDestroyWindow("mainWin");
cvDestroyWindow("Canny");
}

```

Підкреслимо, що вхідне зображення, що створюється камерою на прикладі програми, виводиться на головне вікно проєкту, окремі результати перетворення – до вікон, які визначаються користувачем.

Крім того, зауважимо, що питання очищення пам'яті набуває критичного значення для роботи програми. Не можна не згадати, що функціонування СМЗ повинно діяти в режимі реального часу. З огляду на це, кожна зайва змінна пам'ять застосовуватиметься кілька разів на секунду. Неможна гарантувати, що вона буде коректно застосовуватись у процесі реалізації проєкту.

Закриття головного вікна програми повинно забезпечувати закриття всіх активних у програмі вікон (у протилежному випадку вони можуть залишитися в пам'яті). Таким чином, доцільно змінити функцію OnClose ():

```
void CMainWin::OnClose()// WM_CLOSE додається до карти
повідомлень
{
    cvcamStop(); // зупинка камери
    cvDestroyAllWindows(); // закриття всіх вікон
    cvcamExit(); // вихід із режиму камер
}
```

Заголовні файли, необхідні для роботи об'єкта, подано нижче:

```
#pragma comment(lib,"cv210.lib")
#pragma comment(lib,"cvaux210.lib")
#pragma comment(lib,"cxcore210.lib")
#pragma comment(lib,"highgui210.lib")

#include "afxwin.h"
#include <afxcmn.h>
#include <afxmt.h>
#include <math.h>

#include <highgui.h>
#include <cv.h>
```

```
#include "LegoNXT.h"
```

3.3 Порядок роботи програмного забезпечення

На першому етапі функціонування СМЗ платформи ставиться задача обробки (фільтрації) зображень шляхом залучення різних програмних методів, реалізованих у бібліотеці OpenCV, наприклад, визначення меж об'єктів, створення контурів об'єктів та визначення геометричних критеріїв (центр контуру, площу тощо).

На рисунках продемонстровано ключові результати роботи програми, котра забезпечує оброблення відеоінформації.

З рисунка 3.2 видно, що інформацією для системи машинного зору платформи може бути кадр із зображенням об'єктів різного типу, що надходить з відеокамери. Окрема задача для СМЗ полягає у визначенні меж та крайових точок об'єктів. З цією метою можна послуговуватись кількома методами, зокрема методами Харріса та Кенні.



Рисунок 3.2 – Первинне зображення об'єктів, знятих камерою

Для останнього методу за вхідну інформацію подається чорно-біле зображення, тому для реалізації методу Кенні можна застосувати такий програмний код:

```
cvCvtColor (src, gray, CV_RGB2GRAY)
cvCanny (gray, dst3,25,100 + cannyt, 3),
```

де *src* – вихідне зображення;

gray – вихідне зображення, що перетворено на чорно-білий колір;

dst3 – вихідний масив перетворення Кенні.

Результат методу з'ясування меж об'єктів продемонстровано на рисунку 3.3. Третій та четвертий критерії функції *cvCanny()* збігаються з порогами перетворення. Зміна значень може контролюватись регулятором, також зображеному на рисунку 3.3 (змінна *cannyt*).



Рисунок 3.3 – Застосування методу Кенні
для визначення меж об'єктів

Результати застосування методів визначення меж об'єктів є вагомими для отримання інформації про контури об'єктів. Для їхнього окреслення

застосовується ще одна OpenCV-функція:

```
contrNum = cvFindContours (dst3, storage, & contour, sizeof (CvContour),
CV_RETR_CCOMP, CV_CHAIN_APPROX_SIMPLE),
```

де `dst3` – масив зображення, отриманий перетворенням Кенні;

`storage` – вихідний масив функції отримання контурів;

`contour` – пов'язаний список, у якому міститься опис послідовностей контурів та окремих елементів.

Заключним завданням етапу розроблення є виведення до вікна програми контурів, знайдених на зображенні, визначення їхніх центрів та позначення у спеціальний спосіб. У подальшому отримана інформація може бути використана системою підтримки та прийняття рішень платформою у процесі планування дій маніпулятора платформи. До того ж, пов'язаних із маніпуляціями знайденими у робочому полі платформи об'єктами.

У процесі оброблення контурів передбачається робота з їхнього зв'язаного списку. Центри контурів вираховуються за допомогою функції обчислення моментів і позначаються колами у вихідному вікні:

```
for (; contour! = 0; contour = contour-> h_next)
{
    cvDrawContours (dst7, contour, color, color, -1,1, 8);
    cvMoments (contour, & moments, 0);
    centre_of_mass.x = moments.m10 / moments.m00;
    centre_of_mass.y = moments.m01 / moments.m00;
    cvCircle (dst7, centre_of_mass, 5, CV_RGB (255,255,255), 1, 8,0)
}
```

Заключний результат продемонстровано на рисунку 3.4, на якому зазначені сформовані контури об'єктів та зафіксовано (за допомогою кіл малого радіусу) їхні центри. З огляду на це, програма спроможна визначати

контури об'єктів та їхні окремі властивості у режимі реального часу.

Наступний крок, в якому, за умов реалізації, можна побудувати карти робочих зон платформи. Проте слід враховувати, що площина огляду WEB-камери є доволі невеликою, до того ж має прямокутну форму. Зважаючи на це, виникає потреба в інсталяції спеціальної процедури для побудови карти робочої зони. Дана процедура планується шляхом дискретної фіксації окремих ділянок робочої зони, а також обробленням інформації у кожній області. Разом з тим розглядаються варіанти застосування як циліндричної, так і прямокутної системи координат.

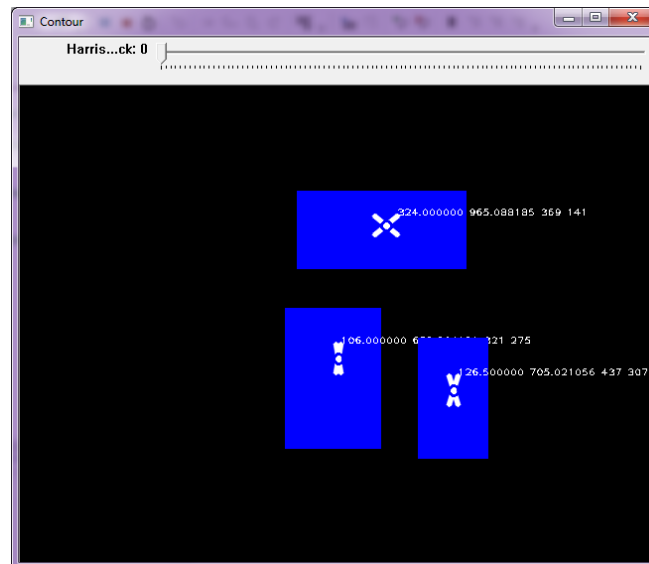


Рисунок 3.4 – Застосування функції виявлення контурів і центрів об'єктів

Функція `OnMouse()` реагує на натискання лівої та правої клавiш миші. Зауважимо, що у разі натискання лівої клавiші задається поточна позиція МП, а також виконується її переведення із системи координат Windows-програми до системи координат платформи. Реалізується так звана релокалізація, тобто, завдання нових координат мобільної платформи. У разі натискання правої клавiші задається так звана нульова точка (waypoint) як елемент траєкторій переміщення. Крім того, за допомогою процедури пошуку шляху з урахуванням виявлених перешкод будується шлях руху платформи `pl=pathCv(ac2)`.

Побудований шлях прямує $r_pl=reversCv(pl)$, з нього видаляються повторювані цикли і послідовності $r_pl=roadripper(r_pl)$.

Для функціонування шляху за допомогою системи управління платформою запускається потік управління RobotRouter.

Функція OnMouse() забезпечує опрацювання повідомлень шляхом натискання лівої та правої клавіш миші.

Натискання лівої клавіші активізує візуальну вказівку поточного положення платформи у вікні програми:

```

case CV_EVENT_LBUTTONDOWN:
if(bStart)
{ sp=FindCellCv(CPoint(x,y));
  InitPoint.x=1.47-sp.x/32.0*1.47; InitPoint.y=sp.y/24.0*1.1;
  cvCurPoint.x = (float)x; cvCurPoint.y = (float)y;
  RobotCommand(1);
  if(sp==CPoint(0,0))AfxMessageBox("StartPoint isn't set");
  else {curPoint=sp;
        vp[sp.x][sp.y]=true;
        setpoint=true; }
}

```

Натискання правої клавіші миші активізує послідовне візуальне цілезазначення точок маршруту МП:

```

case CV_EVENT_RBUTTONDOWN:
  sp.x = (1.47 - cmd.currentPose[0])*32.0/1.47;
  sp.y = cmd.currentPose[1] * 24.0/1.1
  if(setpoint)
  { ep=FindCellCv(CPoint(x,y));
    if(ep==CPoint(0,0))AfxMessageBox("EndPoint isn't set");
    else {cell *ac2=new cell(curPoint,NULL);

```



```
pl=pathCv(ac2); }}
```

До того ж, у даному режимі реалізується запуск функції pathCv(), побудова маршруту платформи, функція обігу маршруту reversCv() та його оптимізації roadripper():

```
pl=pathCv(ac2);
r_pl=reversCv(pl);
r_pl=roadripper(r_pl);
```

Після завершення процесу побудови маршруту активується потік RobotRouter для функціонування запропонованого маршруту переміщення:

```
while(pl)
{FillCellCv(pl->curpos.x,pl->curpos.y,2);pl=pl->next;}
if(r_init)AfxBeginThread(RobotRouter,NULL);
}
```

Наступною функцією, що заповнює зайняту площу робочого простору, стає setBusycell():

```
void setBusycell()
{for(int i=BusyArea.x/25;i<=(BusyArea.x+BusyArea.width)/25+1;i++)
for(int j=BusyArea.y/25;j<=(BusyArea.y+BusyArea.height)/25+1;j++)
FillCellCv(i,j,3);}
```

Функція robotItsels() вираховує поточну позицію платформи:

```
bool robotItself(CvPoint centre_of_mass)
{
bool res=false;
for(int i=curPoint.x-1;i<curPoint.x+1;i++)
for(int j=curPoint.y-1;j<curPoint.y+1;j++)
```

```

if(CPoint(i,j)==FindCellCv(CPoint(centre_of_mass.x,centre_of_mass.y)))
    res=true;
    return res;
}

```

У процесі роботи програми враховується, що робоча область платформи – дискретна, тобто є масивом 33×25 , що пов'язано з роздільною здатністю системи машинного зору платформи: CPoint cp[33][25] (cp – current point). Зайнятість точок простору описується за допомогою масиву bool bp[33][25] (bp – busy point). Відвідані платформою точки позначаються масивом vp[33][25] (vp – visited point).

```

CPoint cp[33][25],sp,ep,curPoint,prevPoint;
bool bp[33][25],d=true,not_subrout=true;
bool vp[33][25]

```

Мінімальною вимогою стає залучення датчика системи машинного зору з роздільною здатністю 320×240 (додаткова одиниця у розмірності із системою координат вікна програми).

У процесі планування переміщення платформи програма буде траєкторію її переміщення між центрами дискретних осередків. До того ж, дискретними вважаються і перешкоди у робочому просторі. Іншими словами, часткове заняття осередку перешкодою, з позиції програми, позначає її повну зайнятість. Такий підхід може погіршити роботу програми з низькою роздільною здатністю камери, втім не впливає на роботу за умови високої роздільної здатності, для якої розмір дискретного масиву робочого простору мобільної платформи пропорційно збільшено.

Повний текст програми наведено у Додатку А.

4 ОХОРОНА ПРАЦІ

4.1 Аналіз умов праці в лабораторії

Дослідження з кваліфікаційної роботи проводилися в лабораторії розміром: 4,2 м × 3,5 м × 3,2 м. У приміщенні працює 1 людина. Площа приміщення становить 14,7 м², об'єм – 47,04 м³. На одне робоче місце припадає вся площа і об'єм приміщення, відповідає нормативному документу НПАОП 0.00-1.28-10 [14].

Було визначено, відповідно до ДСТУ 2293-99, що в лабораторії існують небезпечні та шкідливі чинники:

- рухливість повітря знижено;
- недостатня освітленість робочої зони;
- підвищений показник напруги в електричному ланцюзі, у разі чого може виникнути замикання через тіло людини (небезпечний чинник).

Домінуючий шкідливий виробничий чинник – недостатньо штучного освітлення.

4.2 Промислова безпека в лабораторії

У лабораторії застосовується трифазна чотирьохдротова мережа змінного струму напругою 220/380 В, частота 50 Гц, режим нейтралі – глухозаземлена. Згідно з НПАОП 40.1-1.21-98, приміщення належить до класу приміщень без підвищеної небезпеки. Умови, що сприяють підвищеній та особливій небезпеці (підвищена вологість, струмопровідний пил, струмопровідні підлоги, можливість одночасного дотику до заземлених металоконструкцій будівлі та металевих поверхонь електроприладів) – відсутні.

Для захисту працівників від ураження електричним струмом у лабораторії, відповідно до НПАОП 40.1-1.32-01, застосовується система заземлення TN-C-S типу та захисне відключення.

Необхідно проводити контроль ізоляції згідно з вимогами ПУЕ-2011. Здійснювати контроль між нульовим та фазним провідниками, а також між фазами. Опір ізоляції не менше 500 кОм на фазу. Контроль проводити не рідше 1 разу на рік за умови відключеного електроживлення.

Електропроводка в приміщенні виконана з можливістю заміни: прихована в каналах будівельних конструкцій, що відповідає вимогам НПАОП 40.1-1.32-01.

4.3 Виробнича санітарія і гігієна праці

Роботи, що проводяться в приміщенні лабораторії, виконуються сидячи та не потребують систематичних важких фізичних навантажень, підняття та перенесення важких речей. За таких умов енерговитрати працівників складають не більше 120 ккал/год. Отже, виконувані роботи належать до категорії 1а.

Відповідно до ДСН 3.3.6.042-99 та категорії виконуваних робіт, для приміщення встановлені наступні оптимальні норми мікроклімату: температура в літній період 23 – 25 °С, в зимовий період 22 – 24 °С, відносна вологість повітря 40 – 60 %, швидкість руху повітря $\leq 0,1$ м/с.

Щоб забезпечити необхідні норми мікрокліматичних критеріїв та чистоти повітря в лабораторії застосовується кондиціонування. Підтримка перелічених критеріїв у холодний період реалізується системою опалення відповідно до СНиП 2.04.05-91.

Ще один шкідливий чинник – це розумове перенапруження. Основні заходи щодо захисту від розумового перенапруження людини:

– регулярно, через кожні 40 – 50 хв. робочого часу робити технологічні перерви, для розрядки розумового напруження;

–проведення фізичних вправ (зарядки, розминки) під час перерви.

Рівень загального штучного освітлення приміщення можна перевірити за допомогою методу питомої потужності.

Розрахунок штучного освітлення проводиться методом коефіцієнта використання світлового потоку. Мета перевірного розрахунку – визначення фактичної освітленості в приміщенні. Основна розрахункова формула методу коефіцієнта використання світлового потоку:

$$F = \frac{E_f k_z S_z}{n \eta \gamma N}, \quad (4.1)$$

де E_f – фактична освітленість, лк;

S – площа освітлюваного приміщення, що залежить від розмірів приміщення;

z – коефіцієнт нерівномірності освітленості ($z=1,1$);

k_z – коефіцієнт запасу, що враховує запилення світильників та знос джерел запасу світла в процесі експлуатації;

N – кількість світильників в ряду;

η – коефіцієнт використання світлового потоку ламп;

γ – коефіцієнт затінення ($\gamma=0,8$);

n – кількість рядів світильників;

У таблиці 4.1 наведено вихідні дані.

Таблиця 4.1 – Вихідні дані

F_l	n_l	N	η	N	k_z
1600	2	2	0.5	4	1.5

Фактична освітленість складає:

$$E_f = \frac{Fm\eta\gamma N}{k_z S_z}, \quad (4.2)$$

$$F = F_i n_i, \quad (4.3)$$

$$F = 1600 \cdot 2 = 3200 \text{ Лк},$$

$$E_f = \frac{3200 \cdot 2 \cdot 0,5 \cdot 4 \cdot 0,88}{1,5 \cdot 14,07 \cdot 1,1} = 441 \text{ Лк}.$$

За результатами розрахунку отримали освітленість складає 441лк, що належить допустимому діапазону від 200 лк до 500 лк. У приміщенні нестачі штучного освітлення немає.

4.4 Пожежна безпека лабораторії

Приміщення лабораторії за вибухо- та пожежною небезпекою відноситься до категорії В за НАПБ Б.03.002-2007, оскільки в приміщеннях лабораторії знаходяться тверді горючі матеріали [14]. Основними причинами виникнення пожежі можна вважати:

- випадкове пошкодження ізоляції струмоведучих провідників;
- незадовільний стан вилок, розеток;
- перенавантаження дротів живлення.

Лабораторія розташована в будівлі II ступеня вогнестійкості відповідно до ДБН В.1.1.7-2016 .

За пожежонебезпекою, згідно з ПУЕ-2011, лабораторія належить зоні П – Па [14]. У ній присутні тверді горючі речовини, папір, меблі, одяг.

Заходи пожежної профілактики відповідно до ГОСТ 12.1.004-91. ССБТ, до яких входять: застосування системи запобігання пожежі, протипожежного захисту, виконання організаційних заходів.

Згідно НАПБ Б03.001-2004 в приміщеннях розміщені первинні засоби пожежогасіння – вуглекислотний вогнегасник ВВК-1,4 з розрахунку 1 вогнегасник на 3 ПК, але не менше 1 на приміщення.

Згідно ДБН В.2.5.56-2010 в приміщеннях встановлено точковий димової-пожежний сповіщувач ДП-1, який контролює площу до 86 м² [14].

ВИСНОВКИ

У процесі функціонування автоматизованих систем управління інтегрованих виробничих систем доцільно відстежувати розміщення платформ та технологічного обладнання, особливо за умов динаміки робочого простору та окреслених виробничих задач. Все це говорить про необхідність нових підходів у галузі розроблення інформаційного та програмного забезпечення інтелектуальних роботизованих систем.

У поданій кваліфікаційній роботі за допомогою інтегрованого середовища розроблення Microsoft Visual Studio та мови програмування C++ було створено програмне забезпечення для управління мобільною платформою із залученням візуальної цілевказівки. Такий продукт може функціонувати в автоматизованих системах управління гнучкого інтегрованого виробництва, а також у робочому просторі інтегрованої системи.

Для досягнення поставленої мети було реалізовано низку задач:

- досліджено категорії та використання мобільних платформ;
- вивчено системи візуального управління;
- створено програмне забезпечення для управління мобільною платформою;

Виконана робота дає можливість зробити висновок, що на сьогодні візуальне управління мобільними платформами є актуальною науково-практичною задачею, розв'язання якої може бути корисним для різних сфер життєдіяльності.

При виконанні розділу «Охорона праці» були визначені небезпечні і шкідливі виробничі фактори. Таким чином, були розроблені заходи і технічні засоби щодо забезпечення безпеки праці працюючого персоналу. Застосування цих заходів дозволить максимально знизити ймовірність отримання травм при роботі, а також поліпшити умови роботи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008:2015 Інформація та документація «Звіти у сфері науки і техніки». Структура та правила оформлювання. / В. Земцева; Ю. Поліщук, канд. фіз.-мат. наук; Р. Санченко, канд. техн. наук; Л. Шрамко; А. Ямчук (науковий керівник) ДП «УкрНДНЦ» від 22 червня 2015р. № 61 з 2017- 07-01.
2. Методичні вказівки з підготовки кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти денної і заочної форми навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / Упоряд.: І.Ш. Невлюдов, О.І. Филипенко, О.В. Токарева, С.П. Новоселов, О.В Сичова. – Харків: ХНУРЕ, 2023. – 64 с.
3. Навчальний посібник з підготовки кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форм навчання спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія» [Електронний ресурс] : навчальний посібник / І. Ш. Невлюдов, О. В. Токарева, О. М. Цимбал, А. І. Бронніков ; М-во освіти і науки України, ХНУРЕ. – Харків : Видавництво Іванченка І. С., 2023. – 218 с.
4. Siegwart, Roland. Introduction to Autonomous Mobile Robots / Roland Siegwart, Illah R. Nourbakhsh. – A Bradford Book. MIT Press. – 2014. – 336 p.
5. Yong L. Robotics and Autonomous Systems / L. Yong, J. Jim Zhu, – U: Ohio Design Center. – 2018. – 480 с.
6. Костюк В. І. Робототехніка і мехатроніка / В. І. Костюк, Г. О. Спину. – К.: Вища школа, 1994. – 446 с.

7. Платов І. М. Використання лазерного датчика відстані VL53L0X для систем зору мобільних роботів / І. М. Платов, О. М. Павловський // XIII Науково-практична конференція студентів, аспірантів та молодих вчених «Погляд у майбутнє приладобудування», 13-14 травня 2020 р., м. Київ, Україна: збірник праць конференції. – Київ: КПІ ім. Ігоря Сікорського, 2020. – С. 59-62.

8. Невлюдов І.Ш. Людино-машинний інтерфейс в технічних засобах автоматизації: Навчальний посібник / І.Ш. Невлюдов, О.І. Филипенко, Б.О. Шостак. – Харків : «ХТМТ», 2019. – 244 с.

9. Субботін С.О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: Навчальний посібник / С.О. Субботін. – Запоріжжя: ЗНТУ, 2008. – 341 с.

10. Невлюдов І.Ш. Технічні засоби автоматизації: Підручник / І.Ш. Невлюдов, А.О. Андрусевич, О.І. Филипенко, Н.П. Демська, С.П. Новоселов. – Кривий Ріг : Криворізький коледж НАУ, 2019. – 366 с.

11. Савченко, А. С. Синельников О.О. Методи та системи штучного інтелекту: навч. посібник– Київ : НАУ, 2017. – 176 с.

12. Лубко Д.В., Шаров С.В. Методи та системи штучного інтелекту: навч. посібник – Мелітополь: ФОП Однорог Т.В., 2019. – 264с

13. Діагностика та контроль робочих процесів: навч. посіб. для студентів спеціальності «Прикладна механіка» денної та дистанційної форм навчання / В. М. Доля – Харків: НТУ «ХПІ», 2019. – 129 с.

14. Організація керування умовами праці» підготовки освітнього рівня бакалавр усіх спеціальностей та усіх напрямів університету / ХНУРЕ; розроб.: Т.Є. Стиценко, Г.В. Пронюк, Н.М. Сердюк. – Харків, 2017. – 108 с.