

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління
(повна назва)

Кафедра _____ електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти _____ другий (магістерський)

Система розпізнавання номерів контейнерів за
допомогою веб-камери, встановленої на мобільному
колісному роботі
(тема)

Виконав:

студент _____ II курсу, групи _____ СПМ-21-2
Сергеев Д.А.
(прізвище, ініціали)

Спеціальність _____
123 «Комп'ютерна інженерія»
(код і повна назва спеціальності)

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

Освітня програма _____
Системне програмування
(повна назва освітньої програми)

Керівник: _____ проф. Каргін А.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

_____ Коваленко А. А.
(підпис) (прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 123 «Комп'ютерна інженерія» _____
(код і повна назва)

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

Освітня програма _____ Системне програмування _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту _____ Сергєєву Данилу Андрійовичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи Система розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі

затверджена наказом по університету від “ 03 ” квітня 2023 р. № 318 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 17 травня 2023 р

3. Вхідні дані до роботи _____

4. Перелік питань, що потрібно опрацювати у роботі _____

Огляд існуючих систем

Вибір методів розробки та вибір робота

Концептуальна модель системи

Фізична модель системи

Створення нейромережі

Реалізація системи

Тестування системи

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) _____

Презентація доповіді – 10 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

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	Огляд існуючих систем	03.04.2023-10.04.2023	
2	Вибір методів розробки та вибір робота	11.04.2023-16.04.2023	
3	Концептуальна модель системи	16.04.2023-19.04.2023	
4	Фізична модель системи	19.04.2023-23.04.2023	
6	Реалізація системи	23.04.2023-6.05.2023	
7	Тестування системи	06.05.2023-10.05.2023	
8	Подання кваліфікаційної роботи керівнику на попередній захист	11.05.2023-14.05.2023	
9	Подання кваліфікаційної роботи на рецензування	14.05.2023-17.05.2023	

Дата видачі завдання 03 квітня 2023 р

Студент _____
(підпис)

Керівник роботи _____
(підпис)

проф. Каргін А.О.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 66 с., 23 рис., 3 дод., 16 джерел.

СИСТЕМА РОЗПІЗНАВАННЯ, МОБІЛЬНИЙ КОЛІСНИЙ РОБОТ, АЛГОРИТМ, ВЕБ-КАМЕРА, СЕРВЕР, МОДЕЛЬ

Метою кваліфікаційної роботи є розробка системи, яка може автоматично розпізнавати номери контейнерів за допомогою веб-камери, що встановлена на мобільному колісному роботі.

У ході виконання кваліфікаційної роботи було зібрано та опрацьовано набір даних для розробки системи розпізнавання номерів контейнерів, розроблені нейромережі, які забезпечують, знаходження номеру та його розпізнавання. Було перевірено ефективність системи на тестовому наборі даних.

ABSTRACT

Master's thesis: 66 pages, 23 figures, 3 appendices, 16 sources.

RECOGNITION SYSTEM, MOBILE WHEEL ROBOT, ALGORITHM,
WEBCAM, SERVER, MODEL

The major goal of this thesis is to develop a system that can automatically recognize container numbers using a web camera installed on a mobile wheeled robot.

In order to achieve this goal, a set of data was collected and processed during the execution of the qualification work to develop a container number recognition system. Neural networks were developed to facilitate the identification and recognition of the container numbers. The efficiency of the system was verified on a test dataset.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	11
1.1 Аналіз мобільних роботів та галузей їх застосування	11
1.2 Огляд методів розпізнавання номерів за допомогою веб-камери	18
1.3 Огляд систем розпізнавання зображень	22
1.4 Постановка задачі.....	26
2 ДОЛІДЖЕННЯ ТА ПРОЕКТУВАННЯ СИСТЕМИ РОЗПІЗНАВАННЯ НОМЕРІЙ КОНТЕЙНЕРІВ ЗА ДОПОМОГОЮ ВЕБ-КАМЕРИ, ВСТАНОВЛЕНОЇ НА МОБІЛЬНОМУ КОЛІСНОМУ РОБОТІ.....	28
2.1 Концептуальна модель системи	28
2.2 Логічна модель системи	34
2.3 Моделювання в середовищі MatLab.....	30
3 РЕАЛІЗАЦІЯ СИСТЕМИ.....	43
3.1 Реалізація системи	43
3.2 Опис використаних нейромереж.....	48
3.3 Опис результатів тестування	51
ВИСНОВКИ	55
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	56
ДОДАТОК А ГРАФІЧНИЙ МАТЕРІАЛ КВАЛІКАЦІЙНОЇ РОБОТИ	58
ДОДАТОК Б СТРУКТУРА НЕЙРОМЕРЕЖІ ДЛЯ ЗНАХОДЖЕННЯ НОМЕРУ КОНТЕЙНЕРУ НА ЗОБРАЖЕНІ.....	63

ДОДАТОК В СТРУКТУРА НЕЙРОМЕРЕЖІ РОЗПІЗНАВАННЯ	
СИМВОЛІВ НОМЕР КОНТЕЙНЕРА.....	66

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І
ТЕРМІНІВ

ПЗ – програмне забезпечення

БС – безлюдні системи

МР – Мобільний робот

ШІ – штучний інтелект

ШНМ – штучн нейронна мережа

ВСТУП

В сучасному світі, де все швидко розвивається і наука досягла значних успіхів, необхідність в автоматизації процесів стала дедалі вищою. Через це з'являється потреба в нових інноваційних технологіях, які можуть спростити і полегшити роботу людей та організацій.

Важливою складовою бізнесу в багатьох галузях промисловості є товари, що транспортуються у контейнерах. Технологія транспортування передбачає проміжні етапи тимчасового зберігання контейнерів на складах, обслуговування яких автоматизовано. Більшість сучасних складів відносяться до безлюдних систем (БС) [1, 2]. Автоматичні склади, як БС, передбачають автономне функціонування у тому числі й в непередбачених ситуаціях, коли системи керування різноманітним обладнанням складу повинні приймати рішення без втручання людини. Такі склади в залежності від спеціалізації обладнанні мобільними роботами, які виконують, як завдання по транспортуванню контейнерів, так й допоміжні функції, наприклад, додатковий облік й контроль наявності певних контейнерів на певних місцях й таке інше [2]. Мобільні роботи (МР) мають переміщуватися до вказаної позиції складу по певному маршруту й використовуючи зорову камеру ідентифікувати місце, наявність контейнеру певного типу й його номер для подальшої обробки системою обліку. За для того, щоб не порушувати цикли основних технологічних процесів МР повинні орієнтуватися в поточній ситуації, уникати аварійних ситуацій й виконувати своє завдання в автономному режимі.

Система розпізнавання номерів контейнерів є важливим елементом в логістиці та транспорті. Вона дозволяє швидко та ефективно ідентифікувати контейнери та відслідковувати їх рух. Однак, традиційні методи розпізнавання номерів контейнерів, такі як ручне введення даних або використання стаціонарних систем, можуть бути складними та затратними.

Мобільний колісний робот є ідеальним засобом для автоматизації

процесу розпізнавання номерів контейнерів. Він може бути ефективно використаний на складах, в портах та на інших місцях з великою кількістю контейнерів, де потрібно швидко та точно ідентифікувати їх.

У даній роботі розглядається проблема розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі. Ця система може бути використана в багатьох сферах, таких як логістика, складське господарство, контроль за вантажопотоками тощо.

Розробка такої системи є актуальною задачею, оскільки вона може значно спростити та прискорити процес розпізнавання номерів контейнерів, що рухаються на території підприємств, а також зменшити витрати на використання дороговартісних систем розпізнавання, які вимагають додаткового обладнання та приміщення для його розміщення.

Також, система розпізнавання номерів контейнерів може бути використана для контролю за вантажопотоками на складах та у логістичних центрах, що дозволить автоматизувати процес складання звітів та підвищити ефективність роботи персоналу.

Отже, розробка системи розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі є важливою задачею, яка може мати широкі можливості застосування в різних галузях транспортного бізнесу та логістики.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз мобільних роботів та галузей їх застосування

Слово "робот", як відомо, слов'янського походження. Його ввів відомий письменник К. Чапек в 1920 р. у своїй фантастичній п'єсі "R.U.R", в якій так названі механічні робітники, призначені для заміни людей на важких фізичних роботах. Назва "робот" утворено від чеського слова robot, що означає важку підневільну працю [3].

Умови функціонування роботів визначаються типом середовища експлуатації і характером робочого процесу, їх можна розділити на дві категорії: детерміновані (визначені) і недетерміновані (невизначені).

До детермінованих середовищ відносяться середовища, спроектовані і створені людиною. Відповідно, детермінованим процесом є будь-який процес, протікання якого повністю залежить від цілеспрямованої діяльності людини.

У детермінованих середовищах вже є або висока, або необхідна міра організації, яка може бути досягнута при порівняно невеликих витратах. Визначеність середовища обумовлена апріорним знанням точного положення усіх об'єктів, з якими може взаємодіяти робот. Для маніпуляційного робота це означає точне знання місця розташування і орієнтації об'єктів, розташованих в його робочій зоні. Для транспортного робота детермінованим середовищем є, наприклад, рейкова траса в цеху. До першої категорії відносяться також середовища, які можна організувати відповідно до вимог, хоча і ціною значних витрат (не повністю організовані середовища). В цьому випадку окремі об'єкти можуть мати заздалегідь невідомі відхилення від еталону.

У середовищах другої категорії практично неможливо здійснити їх організацію. Такі середовища називаються повністю неорганізованими (недетермінованими). До них відносяться, зокрема, природні середовища і середовища, що створюються аварійними ситуаціями як в природних умовах,

так і при руйнуванні середовищ, спроектованих і створених людиною, тобто при руйнуваннях будівель і споруд. До дій робота в природних середовищах відносяться: розвідка на місцевості, військові дії, розмінування і патрулювання, підводні і підземні роботи і тому подібне (у тому числі у випадках радіоактивного, хімічного і бактеріологічного зараження місцевості).

До недетермінованих процесів відноситься будь-який процес, протікання і результат якого повністю не залежить від цілеспрямованої діяльності людини. Недетермінованими процесами є ведення бойових дій, усі природні процеси (землетруси, виверження вулканів і тому подібне), пожежі, вибухи (як результати техногенних аварій) і тому подібне [4].

Класифікація мобільних роботів за:

- сферами застосування можуть бути виробничі (промислові), військові (бойові, такі, що забезпечують), дослідницькі, медичні;
- наземні, підземні, надводні, підводні, повітря, космічні за місцем існування (експлуатації);
- програмні, адаптивні, інтелектуальні за типом системи управління;
- маніпуляційні, транспортні, інформаційні, комбіновані за функціональним призначенню::;
- спеціальні, спеціалізовані, універсальні за універсальністю::;
- електричні, гідравлічні, пневматичні за типом приводів::;
- гусеничні, колісні, колісно-гусеничні, крокуючі, колісно-крокуючі, роторні, з петлевим, гвинтовим, водометним і реактивним за типом переміщення::;
- за типом джерел первинних сигналів управління бувають електричні, біоелектричні, акустичні;
- автоматичні, дистанційно керовані (що копіюють, командні, інтерактивні, супервізорні, діалогові), ручні за способом управління.

Узагальнена структура мобільного робототехнічного комплексу.

Конструктивно універсальні мобільні роботи є самохідними засобами, що оснащуються розвідувальною апаратурою, набором змінного робочого устаткування і інструменту. Вони розраховані на дистанційне керування

оператором, що веде спостереження безпосередньо або за допомогою телевізійної камери.

До складу встановлених на роботах комплексів приладів і устаткування входять (рисунок 1.1) :

- навісне устаткування;
- система освітлення;
- рушій;
- приводи;
- система зв'язку;
- пост управління;
- система відчуття;
- система управління;
- енергоустановка.

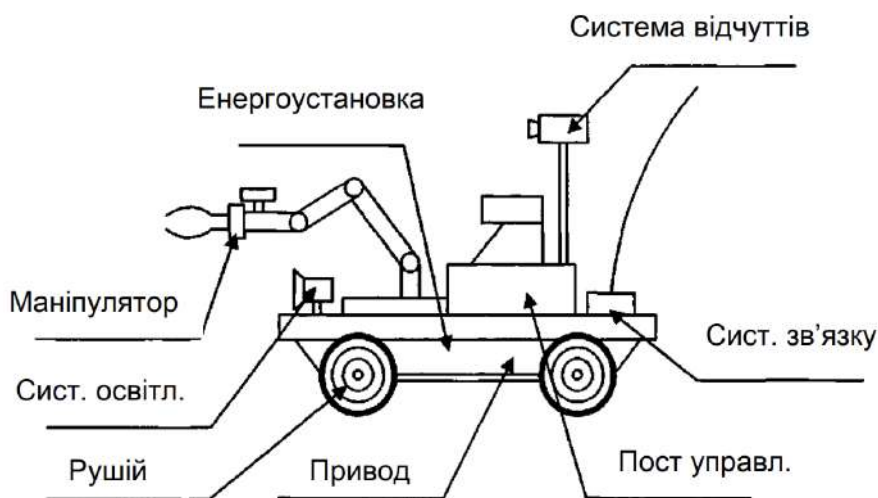


Рисунок 1.1 – Узагальнена структура мобільного робототехнічного комплексу

Будь-який мобільний робот може бути представлений у вигляді сукупності трьох основних систем - транспортної, спеціальної і системи управління (рисунок 1.2) [5]

Транспортна система є транспортним засобом, призначеним для доставки спеціального і технологічного устаткування до місця виконання поставленого

завдання.

Спеціальні системи служать для безпосереднього виконання поставлених завдань. Спеціальна система складається з необхідного набору технологічного устаткування, склад якого визначається видом вирішуваної задачі і призначенням мобільного робота.

Наприклад, при рішенні розвідувальних завдань технологічним устаткуванням є комплект сенсорів і засобу первинної обробки інформації. Виконання технологічних завдань може забезпечуватися маніпулятором і набором змінного інструменту до нього. При проведенні взривотехнічних робіт необхідним устаткуванням є засоби діагностики вибухових пристроїв і гідроруйнівники.

Система управління забезпечує управління рухом і роботою технологічного устаткування робота, а також адаптивне управління ходовою частиною і енергетичною установкою з урахуванням взаємодії транспортної системи з довкіллям. Система управління включає частину (апаратура управління роботом, датчики, система технічного зору і мікропроцесори попередньої обробки інформації), що інформаційно-управляє, розташовану на мобільному роботі; пост оператора мобільного робота (пульт управління, відеопереглядові пристрої; ЕОМ для обробки інформації) і комплект прийомо-передаючої апаратури, що забезпечує передачу інформації від робота на пост оператора і команд, що управляють, від поста оператора на МР.

Система управління рухом повинна також забезпечувати планування руху в недетермінованих умовах на основі картографічної бази з урахуванням інформації, що безперервно поступає, в систему управління від технічних органів чуття і навігаційної системи.

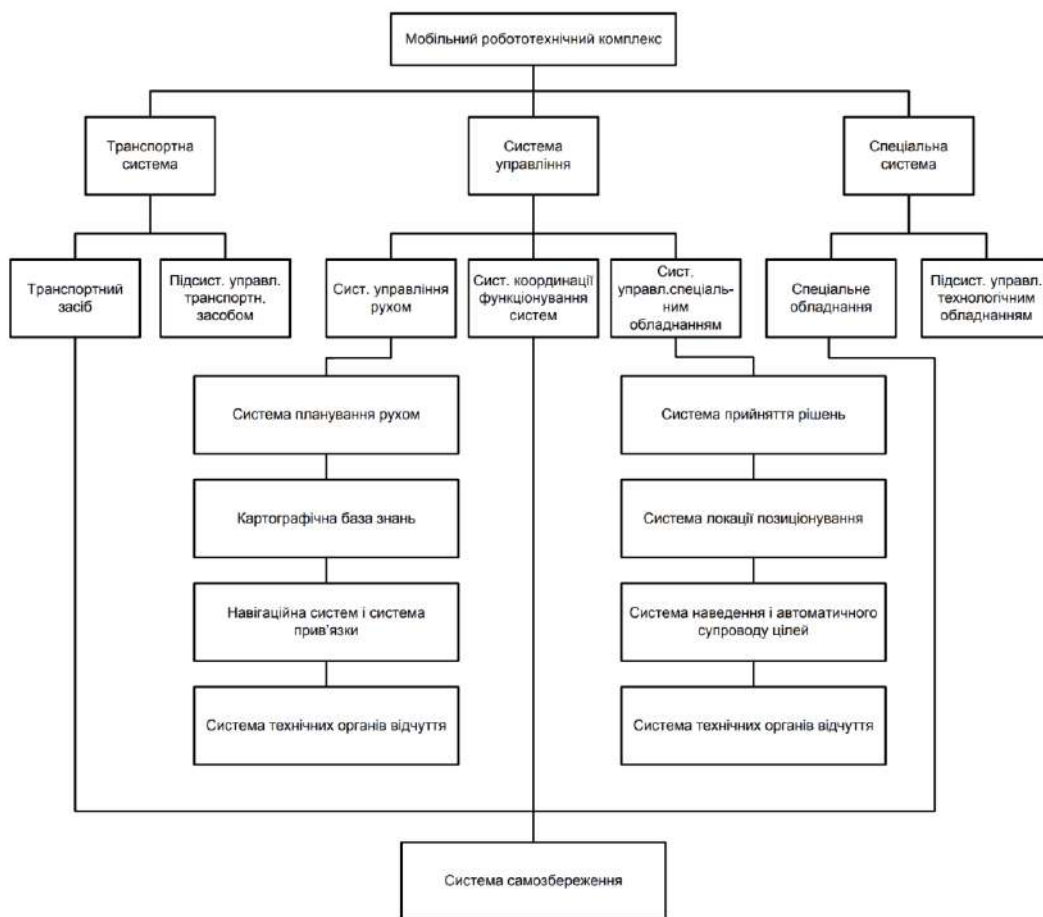


Рисунок 1.2 – Узагальнена структура мобільного робототехнічного комплексу

На рисунку 1.3 приведена типова схема системи управління рухом мобільного робота. Блок управління рухом здійснює управління тяговими приводами і приводами повороту шасі.

Блок побудови маршруту синтезує траєкторію руху, оптимізуючи її, по мінімуму витрати енергії, що особливо важливо для роботів з автономним енергоживленням, або по мінімуму часу виконання завдання, коли в постановку завдання входить умова забезпечення максимальної швидкодії.

Блок побудови карти (плану) місцевості представляє її у формі, зручній для вирішення завдання вибору маршруту, зокрема, з виділенням непереборних перешкод і небезпечних або неясних ділянок. Якщо виконання завдання вимагає знання більшої ділянки місцевості, чим дають сенсорні системи в початковому положенні робота, тобто до початку руху, карта місцевості

формується і передається у блок побудови маршруту фрагментами у міру пересування робота. Такими ж ділянками послідовно здійснюється і синтез траєкторії руху. В цьому випадку перше наближення для усієї траєкторії робиться за наявною апріорною інформацією про місцевість або, як мінімум, визначається загальний напрям руху. На нижньому загальносистемному рівні системи управління знаходиться канал управління рухом у ближній зоні, що включає найбільш детальну модель цієї зони і що реалізує алгоритми забезпечення безпеки руху. Необхідний рівень адаптації і штучного інтелекту системи визначається мірою невизначеності і складності місцевості і характером що підлягають виконанню завдань, тобто призначенням робота.

В останньому відношенні основне значення має мінімально необхідна міра автономності управління роботом, що визначається переліком його дій, які повинні виконуватися без участі людини-оператора. При цьому враховується і можливість тимчасової втрати зв'язку з оператором, обмежена пропускна спроможність каналів зв'язку, неповнота і обмежена достовірність отримуваної оператором від робота інформації і її затримка в часі.

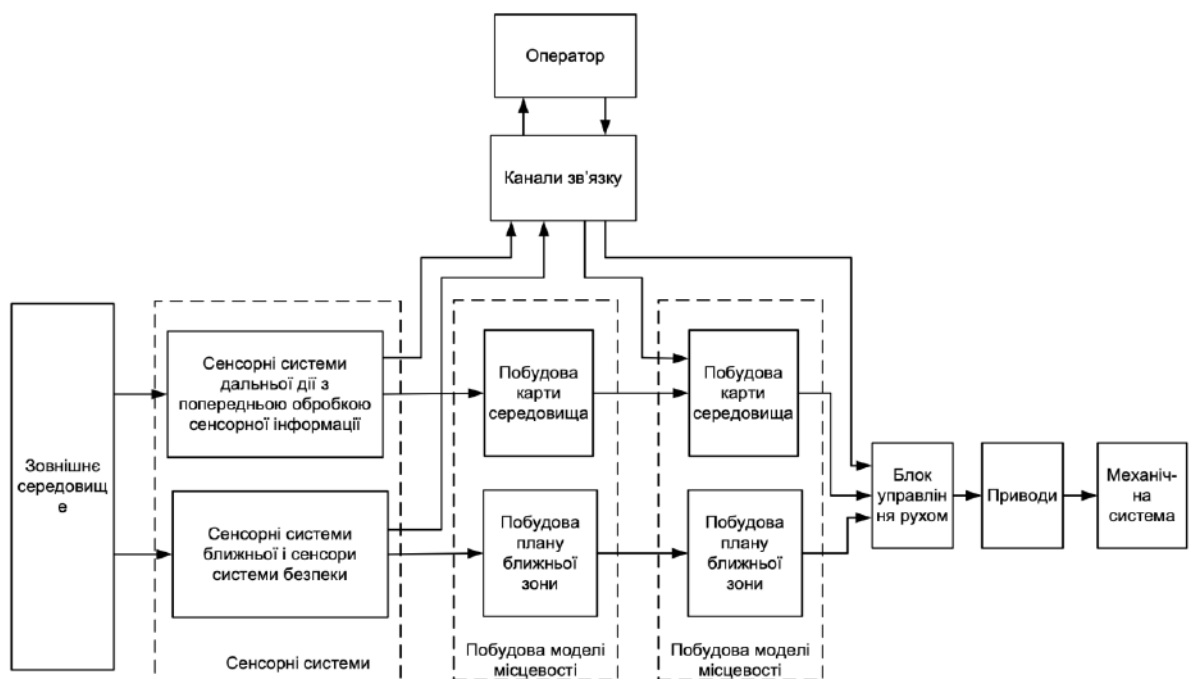


Рисунок 1.3 – Функціональна схема системи управління рухом мобільного робота

Мобільні роботи універсальні і тому можуть бути використані в різних галузях. Розглянемо деякі з них, де мобільні роботи використовуються:

- у логістика та складське господарство використовуються на складах та у логістичних компаніях для автоматизованого переміщення товарів з одного місця на інше. Застосування мобільних роботів дозволяє зменшити кількість ручної роботи, збільшити точність та швидкість переміщення товарів та скоротити час на обробку вантажів;
- у виробництві потрібні для автоматизації рутинних операцій, наприклад, для збору та переміщення комплектуючих. Застосування мобільних роботів дозволяє зменшити кількість помилок при виробництві та підвищити ефективність виробництва;
- у медицині використовуються у закладах для автоматизації рутинних процесів, наприклад, для доставки медикаментів до пацієнтів. Застосування мобільних роботів дозволяє зменшити кількість ручної роботи, збільшити точність та швидкість процесу та забезпечити більш ефективну роботу медичного персоналу;
- задля безпеки мобільні роботи використовуються для забезпечення безпеки в різних галузях, наприклад, на фабриках, у банках та на публічних місцях;
- у сферах контролю, моніторингу та захисту об'єктів безпеки здатні виявляти небезпеки, рухатись у важкодоступних місцях та збирати дані про потенційні загрози;
- в освіті використовуються в освітніх закладах для навчання студентів робототехніки та програмування. Застосування мобільних роботів дозволяє студентам отримати практичні навички у робототехніці та розвивати свої творчі та проблемно-орієнтовані навички;
- у сільське господарство для автоматизованої обробки полів, наприклад, для розкидання добрив та збирання врожаю. Застосування мобільних роботів дозволяє збільшити виробничі потужності та зменшити витрати на робочу силу;

– у ресторанний бізнес для доставки їжі до столів, тощо. Застосування мобільних роботів дозволяє зменшити час очікування замовлень та забезпечити більш точне та ефективне обслуговування клієнтів.

1.2 Огляд методів розпізнавання номерів за допомогою веб-камери, встановленої на мобільному колісному роботі

Незважаючи на всі відмінності програмних і апаратних систем розпізнавання, їх об'єднує те, що всі вони працюють з алгоритмами і використовують певні методи обробки зображень. Розглянемо різні методи і алгоритми розпізнавання номерів котейнерів. Процес ідентифікації номера можна розділити на два етапи: попередній пошук номера (локалізація номерний пластины) та розпізнавання символів

Існує кілька способів попереднього пошуку номера. Найбільш очевидний спосіб виділення номера - пошук прямокутного контуру. Даний спосіб працює тільки при наявності ясної читабельності контуру, з досить високою роздільною здатністю і з рівною границею, при відсутності перешкод між камерою і номерний пластиною.

Проводиться фільтрація зображення для знаходження кордонів, потім проводиться виділення всіх знайдених контурів і їх аналіз. Даний спосіб непрактичний, оскільки контур номеру може бути нечітким (брудним, зливатися з кольором машини, наявність перешкод і т.д.).

Більш цікавим, стабільним і практичним представляється підхід, де аналізується тільки частина рамки номерного знаку. Спочатку виділяються контури, після чого знаходяться всі вертикальні прямі. Для будь-яких двох прямих, розташованих недалеко один від одного, з невеликим зрушенням по осі Y, з правильним відношенням відстані між ними до їх довжини, розглядається гіпотеза того, що номер розташовується між ними [6].

Недоліки попередніх методів полягають в тому, що на реальних номерах, забруднених пилом, немає ні виражених кордонів, ні вираженої статистики,

обов'язкових для їх коректної роботи. Кращі, але рідко використовувані методи – це методи, які спираються на різні класифікатори. Ці методи дають можливість аналізувати задану область на предмет наявності в ній характерних для номера відношень, точок або градієнтів і дозволяють знаходити номер в складних і нетипових умовах. Наприклад, навчений каскад Хаара, робота якого проводиться за методом Віоли-Джонса, в основі якого лежать примітиви Хаара, що представляють собою розбивку заданої прямокутної області на набори різнотипних прямокутних областей. [7]

У реальних алгоритмах багато методів побічно або безпосередньо базуються на наявності кордонів номера. Так само якщо при виявленні номера кордони не використовуються, вони можуть використатися в майбутньому аналізі.

Існують ситуації, в яких даний метод буде неефективний. Наприклад, при ситуації з досить чистим номером, так як такі контейнерів зустрічаються дуже мало, в порівнянні з брудними номерами, то і їх число може виявитися недостатнім для якісного навчання. У таких випадках метод буде не ефективним.

Відкрите програмне забезпечення, яке виконує автоматичне розпізнавання символів (букв) і тексту цілком має назву Tesseract OCR. Переваги полягають в тому, що ця програма існує для будь-яких ОС, легко навчається і стабільно працює. Недоліком же є те, що алгоритм дуже погано працює з битим, брудним і деформованим текстом. При розпізнаванні номерів за допомогою даного ПЗ ймовірність правильного розпізнавання лежить в діапазоні близько 20-30% з існуючої бази: найчистіші і прямі.

Дуже простий для розуміння метод розпізнавання символів, який, незважаючи на свою примітивність, часто дає гарний результат.

Принцип роботи:

- попередньо записується певна кількість зображень реальних символів, коректно розбитих на класи;
- вводиться міра відстані між символами (якщо зображення бінарного типу, то

- операція XOR буде оптимальною);
- при спробі розпізнавання символу, по черзі розраховується дистанція між розпізнаваним символом і всіма символами в базі. Серед k найближчих сусідів, можливо, будуть представники різних класів. Символ отримує такий же клас, як і у більшості сусідніх символів.

Недоліком даного методу може послужити лише необхідність швидкого розрахунку дистанції між зображеннями, а, отже, переведення їх до бінарного типу і використання XOR. Бінаризація зовсім непередбачувано змінює символ, і тому у випадку з забрудненими або потертими номерами будуть проблеми. Незважаючи на це при наявності великої бази з прикладами символів, зафіксованих в різних умовах, теоретично k -nearest - це все, що потрібно [8].

У багатьох випадках дуже важливо розуміти, як працює алгоритм. У даного методу є одне дуже важливе в цьому плані перевага: через простоту і «прозорість» алгоритму налагодження та налаштування на оптимальний результат проводиться без затруднень.

Кореляційний метод розпізнавання - метод розпізнавання образів, при якому для кожного класу об'єктів в декартовому просторі ознак задається еталонна область і будь-який розпізнаний об'єкт відноситься до класу, що відповідає найближчій еталонній області; остання формується шляхом допустимих перетворень одного або декількох еталонних векторів класу. Якщо зображення не приведено до бінарного типу, амплітуда сигналу невідома, отже, немає можливості визначити яскравість символу.

Розпізнавання зводиться до операції розрахунку коваріації вхідного сигналу з гіпотетичним (з урахуванням заданих зсувів і поворотів):

$$\text{cov}(X,Y) = E[(X - EX)(Y - EY)]$$
, де X - вхідний сигнал, Y - гіпотеза.
Позначення E – математичне очікування.

Іноді такі методи називаються «template - matching» (зіставлення шаблонів). Дана назва повністю відображає їх суть, яка полягає у виборі зразків і наступному порівнянні вхідного зображення із зразками. У разі виникнення невизначеності щодо параметрів, проводиться перебір всіх можливих варіантів,

або застосовуються адаптивні підходи.

Переваги методу:

- передбачуваний і добре вивчений результат, якщо шум, навіть в малому ступені, відповідає обраній моделі;
- можливість розпізнати навіть сильно запошований/брудний/потертий символ при строго заданому шрифті.

Недоліки:

- Чималі витрати на процедури обчислення

Адаптивне розпізнавання. Існує два підходи. Шрифтовий підхід. Програма проводить аналіз і виміри різних характеристик шрифту, заносючи їх в свою базу еталонних характеристик [10]. Даний процес можна назвати навчанням програми. Після цього процесу шрифтова програма розпізнавання символів може розпізнати даний конкретний шрифт. Далі навчання повторюється для необхідних шрифтів, типи яких будуть залежати від області застосування програми [11].

До недоліків даного підходу можна віднести наступні фактори:

- алгоритму необхідно заздалегідь знати шрифт;
- необхідність наявності блоку налаштування на конкретний шрифт.

З іншого боку, у шрифтового підходу є істотна перевага: при наявності детальної апріорної інформації про символи, існує можливість побудови досить точних і надійних алгоритмів розпізнавання.

Безшрифтовий підхід. Алгоритми, засновані на даному підході, вимірюють і аналізують різні характеристики (ознаки), властиві буквам будь-якого шрифту і розміру (кегля).

Недоліки цього підходу:

- якість розпізнавання суттєво нижче, в порівнянні з шрифтовими алгоритмами;
- Низька значення коефіцієнта надійності розпізнавання. Переваги: • універсальність (робота алгоритму не залежить від типу шрифту і інших його характеристик);

- технологічність (простота навчання, можливість реалізувати автоматизоване навчання).

7 Нейромережі. Штучна нейронна мережа (ШНМ, нейронна мережа) - це набір нейронів, з'єднаних між собою. Робота нейронної мережі полягає в перетворенні вхідного вектору даних в вихідний вектор [11]. Практично будь-яке завдання можна звести до задачі, розв'язуваної нейронною мережею. Для того, щоб нейронна мережа почала працювати, її необхідно навчити. Для цього в нейрони записується певний еталон, потім подається вхідний сигнал, який порівнюється з еталоном і відмінність між еталоном і вхідним впливом фіксується (додається до еталону). Чим більше буде вхідних впливів, тим краще навчиться нейромережа [12]

Переваги:

- при правильному налаштуванні і навчанні може працювати краще за інших методів;
- при великому навчальному масиві даних з'являється стійкість до спотворень символів [13]

Недоліки:

- складність реалізації;
- в багатошарових мережах неможлива діагностика аномальної поведінки.

1.3 Огляд систем розпізнавання зображень отриманих від веб-камери

Системи розпізнавання зображень, отриманих від веб-камери, можуть бути корисні для багатьох застосувань, таких як розпізнавання облич, детекція руху, відстеження об'єктів та інше. Такі системи можуть бути встановлені на мобільних колесах або роботах для виконання різних завдань.

OpenCV - це відкрите програмне забезпечення для комп'ютерного зору, яке дозволяє розпізнавати обличчя, відстежувати об'єкти, визначати кольори та багато іншого. OpenCV має бібліотеки для різних мов програмування, таких як C++, Python та Java.

Метод на основі бібліотек OpenALPR (Automatic License Plate Recognition), алгоритму Haar Cascade та бібліотеки OpenCV. Бібліотека OpenALPR використовується в системі розпізнавання різного типу номерів, тому числі контейнерів. OpenALPR використовує навчальні дані, що містять інформацію про різні шрифти та символи знаків з різних країн. Бібліотека використовує алгоритми комп'ютерного зору для локалізації номерних знаків на зображенні та витягування з них текстової інформації. Вона налаштована для використання у режимі реального часу, що дозволяє розпізнавати номерні знаки на відео з використанням веб-камер. OpenCV - бібліотека відкритого коду для комп'ютерного зору та обробки зображень. Її функціонал дуже великий. Використовується для розв'язання різноманітних задач, пов'язаних з обробкою зображень, таких як детекція облич, розпізнавання тексту, відслідковування об'єктів та інші[14].

TensorFlow - це відкрите програмне забезпечення для машинного навчання, яке дозволяє створювати системи розпізнавання зображень, які можуть відстежувати об'єкти та розпізнавати обличчя. TensorFlow підтримує мови програмування Python та C++.

Один з головних компонентів TensorFlow - це бібліотека для роботи з тензорами (даними у вигляді багатовимірних масивів) та графами обчислень. TensorFlow дозволяє створювати складні графи обчислень з різноманітними операціями, які можна оптимізувати та виконувати на різних пристроях, таких як CPU, GPU або TPU. Програмне забезпечення має вбудовану підтримку для глибокого навчання, що дозволяє створювати складні нейронні мережі для класифікації, сегментації та розпізнавання образів. Також містить багато попередньо навчених моделей, які можна використовувати в різних задачах.

YOLO (You Only Look Once) - це алгоритм обробки зображень для об'єктного визначення, який був розроблений в 2015 році компанією Darknet.

Він працює на основі нейронної мережі, що використовує глибоке навчання. За допомогою алгоритму обробки зображень можна швидко та точно визначати об'єкти на зображеннях та відео.

Основним принципом є те, що він обробляє зображення одним проходом через нейронну мережу, визначаючи об'єкти та їхні координати в один раз. Це відрізняє його від інших алгоритмів, які розділяють обробку зображення на кілька етапів та вимагають більше обчислювальних ресурсів. YOLO використовує нейронну мережу з декількома шарами, які працюють з даними зображення. Вона розбиває зображення на сітку (grid) та кожен елемент сітки відповідає області зображення. Для кожної області мережа визначає ймовірності того, що там знаходиться об'єкт різних класів, а також координати цього об'єкта. Інформація про всі об'єкти на зображенні формує вихідний результат.

Один з головних переваг YOLO - це швидкість обробки зображень, оскільки він працює швидше за більшість інших алгоритмів об'єктного визначення. Крім того, YOLO може обробляти зображення з високою роздільною здатністю, що дозволяє визначати об'єкти з високою точністю.

Azure Cognitive Services - це хмарна платформа, яка надає API для розпізнавання зображень, яка може розпізнавати обличчя, відстежувати об'єкти та класифікувати зображення. Azure Cognitive Services підтримує різні мови програмування та може бути інтегрована з різними додатками та платформами.

Ці системи можуть бути використані для розпізнавання різних об'єктів та виконання різних завдань. Вибір конкретної системи залежить від конкретної задачі, яку потрібно вирішити. Окрім зазначених вище систем, існують також інші системи розпізнавання зображень, які можуть бути використані для розпізнавання зображень отриманих від веб-камери. Наприклад, Amazon Rekognition, Google Cloud Vision API, IBM Watson Visual Recognition та багато інших.

Як правило, такі системи використовують глибинне навчання та штучні нейронні мережі для розпізнавання зображень. Для розпізнавання облич, наприклад, такі системи можуть використовувати навчальні набори зображень, щоб навчити модель розпізнавати особливості різних облич та здійснювати ідентифікацію.

Такі системи можуть бути дуже корисними для автоматизації різних завдань, що потребують розпізнавання зображень, таких як відстеження об'єктів на заводах, визначення стану автотранспорту на дорогах та багато іншого.

Отже, вибір системи залежить від конкретної задачі, яку потрібно вирішити та від технологічних можливостей, які доступні.

Для розробки системи був обраний варіант – OpenCV.

Він має багато вбудованих алгоритмів та бібліотек для роботи з зображеннями, що робить його ідеальним вибором для системи розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі.

Для розпізнавання номерів контейнерів, система може використовувати алгоритми, такі як "Хаар-каскад", який використовується для виявлення облич на зображенні.

Крім того, OpenCV містить інші алгоритми, які можуть використовуватись для розпізнавання та класифікації об'єктів, такі як алгоритм виявлення кольору, який може бути корисним для визначення кольору контейнера та фільтрування зображень, що не містять контейнерів певного кольору.

Крім того, OpenCV є переносимим та може бути використаний на різних платформах, включаючи мобільні пристрої та міні-комп'ютери, такі як Raspberry Pi, що є важливим фактором для системи, встановленої на мобільному роботі.

Нарешті, OpenCV є відкритим програмним забезпеченням з відкритим кодом, що дає можливість створювати власні рішення та адаптувати систему та робить її більш універсальною. Це може бути корисним для покращення точності та ефективності системи розпізнавання номерів контейнерів.

1.4 Постановка задачі

Метою кваліфікаційної роботи є розробка системи розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на поворотній платформі на мобільному роботі. Для досягнення поставленої мети потрібно вирішити наступні задачі:

- розробити алгоритм виявлення номерів контейнерів на зображеннях, отриманих з веб-камери, встановленої на поворотній платформі;
- розробити алгоритм розпізнавання номерів на зображеннях, отриманих з веб-камери, встановленої на поворотній платформі роботу;
- розробити систему, яка буде здатна працювати в режимі реального часу та відображати результати розпізнавання на відеопотоці;
- виконати тестування системи;
- провести аналіз результатів.

Для вирішення поставлених задач будуть використовуватися такі методи та інструменти:

- бібліотеки комп'ютерного зору OpenCV для реалізації алгоритмів виявлення та розпізнавання номерів контейнерів на зображеннях;
- штучні нейронні мережі глибокого навчання для розпізнавання номерів контейнерів на зображеннях.

Мовою програмування обрано Python для реалізації алгоритмів та системи в цілому.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- зібрати базу даних зображень контейнерів з різними номерами і розмірами для навчання та тестування системи розпізнавання;
- обрати та налаштувати алгоритм розпізнавання за допомогою веб-камери, встановленої на поворотній платформі роботу на основі аналізу різних підходів та технологій;
- розробити програмне забезпечення для роботи системи розпізнавання на поворотній платформі, що забезпечує зчитування та обробку зображень, та

- передачу отриманих даних на сервер;
- провести експериментальні дослідження системи розпізнавання на реальних умовах та оцінити її точність та швидкість роботи;
 - порівняти результати з іншими існуючими системами розпізнавання та зробити висновки про ефективність та перспективність запропонованої системи.

Отже, основною метою роботи є розробка системи розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на поворотній платформі робота.

2 ДОСЛІДЖЕННЯ ТА ПРОЕКТУВАННЯ СИСТЕМИ РОЗПІЗНАВАННЯ НОМЕРІЙ КОНТЕЙНЕРІВ ЗА ДОПОМОГОЮ ВЕБ-КАМЕРИ, ВСТАНОВЛЕНОЇ НА МОБІЛЬНОМУ КОЛІСНОМУ РОБОТІ

2.1 Концептуальна модель системи

Концептуальна модель – абстрактна модель, яка виявляє причинно-наслідкові зв'язки, властиві досліджуваному об'єктові в межах, визначених цілями дослідження. По суті, це формальний опис об'єкта моделювання, який відображає концепцію (погляд) дослідника на проблему. Модель складається з переліку взаємопов'язаних понять, що використовуються для опису цієї області, разом з властивостями й характеристиками, класифікацією цих понять, за типами, ситуацій, ознаками в даній області і законів протікання процесів в ній.

Змістовна модель — це абстрактна модель, що визначає структуру модельованої системи, властивості її елементів і причинно-наслідкові зв'язки, властиві системі і суттєві для досягнення мети моделювання.

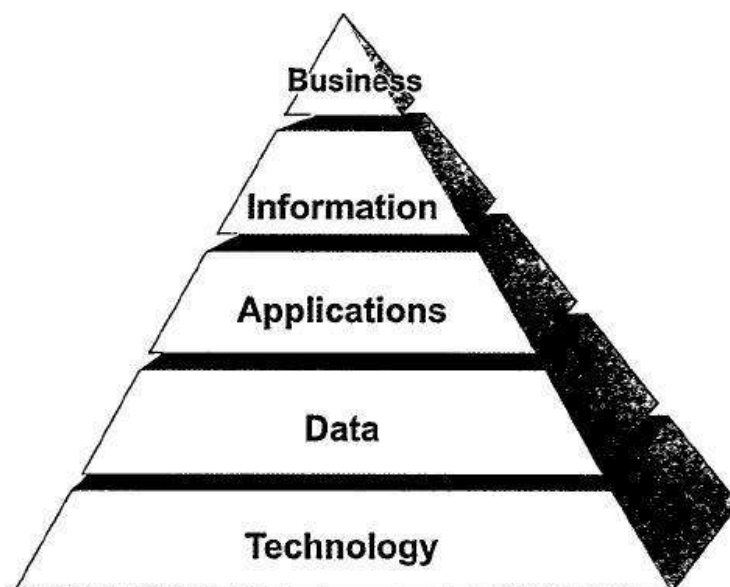


Рисунок 2.1 – Концептуальна модель

Концептуальна модель системи розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі може бути описана наступним чином.

Існують наступні варіанти використання (use case) системи:

- розпізнавати за допомогою веб-камери, яка встановлена на мобільному роботі та зчитує номери контейнерів;
- розпізнані номери будуть збережені та оброблені системою для наступних кроків;

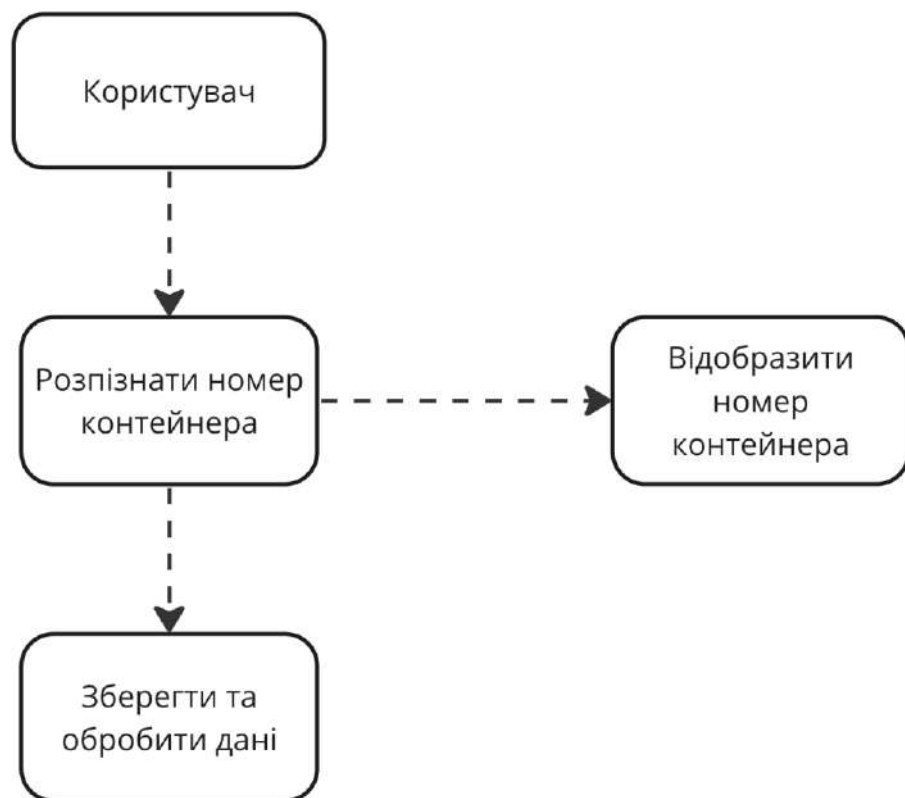


Рисунок 2.2 – Діаграма варіантів використання системи

Для досягнення цієї мети, система може використовувати компоненти, такі як:

- веб-камера, що встановлена на мобільному роботі, буде використовуватися для зчитування номерів контейнерів. Камера може бути підключена до системи за допомогою проводу або бездротової технології;
- система розпізнавання номерів контейнерів буде використовуватися для

- інформацію, що зчитуються з веб-камери. Для розпізнавання можуть використовуватися алгоритми машинного навчання або комп'ютерного зору;
- номери контейнерів можуть бути збережені та оброблені системою для подальшої обробки. Цей компонент може включати в себе базу даних та програмне забезпечення для обробки даних;
 - може мати інтерфейс користувача, що дозволяє користувачу взаємодіяти з системою та переглядати розпізнані номери контейнерів.

У цій системі веб-камера зчитує номери контейнерів, які потім розпізнаються системою розпізнавання. Після розпізнавання вони зберігаються в базі даних та можуть бути оброблені іншими компонентами системи, наприклад, для подальшого використання в логістичних процесах.

Для того, щоб краще зрозуміти, як система буде взаємодіяти з користувачем та які дії він може виконувати, можна скласти діаграму варіантів використання (use case diagram). Вона містить список можливих дій користувача та системи.



Рисунок 2.3 – Можливі дії користувача та системи

Користувач може взаємодіяти з системою за допомогою веб-камери, де

зчитуються номери контейнерів. Система потім розпізнає та зберігає їх для подальшої обробки. Результати розпізнавання можуть бути відображені користувачеві через інтерфейс системи. Таким чином, концептуальна модель системи розпізнавання номерів контейнерів за допомогою веб-камери може включати в себе компоненти, такі як веб-камера, система розпізнавання, база даних та інтерфейс користувача.

З використанням такої системи можна значно спростити та автоматизувати процес розпізнавання та обробки номерів контейнерів, що може допомогти у підвищенні ефективності та точності логістичних процесів. Для того, щоб краще зрозуміти, як система буде працювати, можна також скласти діаграму використання системи (system use case diagram), яка відображає взаємодію між компонентами системи.

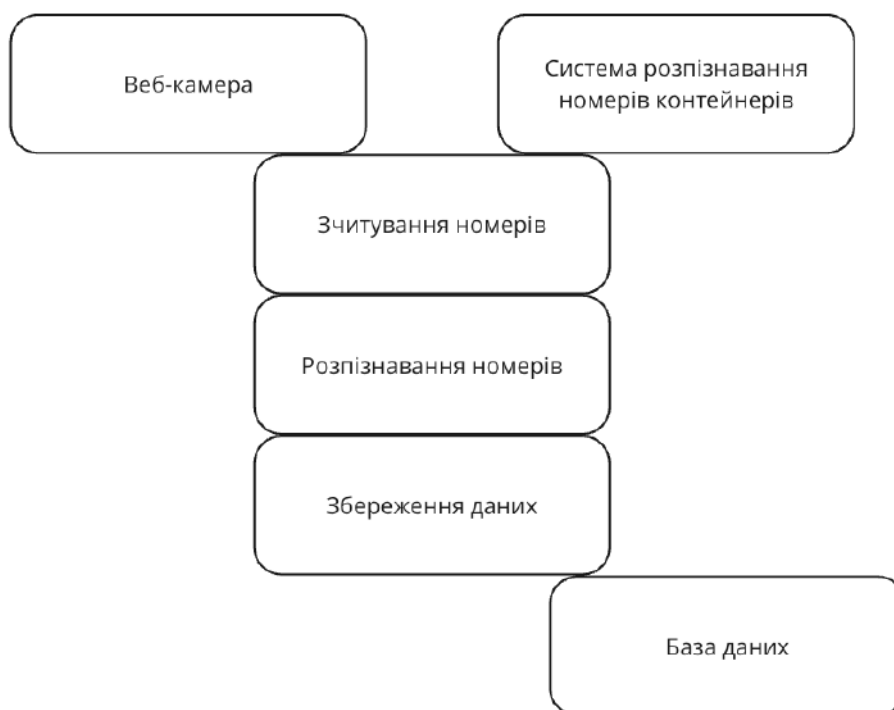


Рисунок 2.4 – Взаємодію між компонентами системи

Веб-камера зчитує номери контейнерів та передає їх на систему розпізнавання. Система розпізнавання контейнерів обробляє та зберігає їх у базі даних. Таким чином, використовуючи таку систему, можна автоматизувати процес розпізнавання та обробки номерів контейнерів, що може допомогти у

підвищенні ефективності логістичних процесів.

З використанням такої системи можна значно спростити та автоматизувати процес розпізнавання та обробки, що може допомогти у підвищенні ефективності та точності логістичних процесів. Для кращого розуміння системи, можна також скласти діаграму варіантів використання (use case diagram), яка допоможе відобразити функціональність системи та взаємодію її компонентів з користувачами.

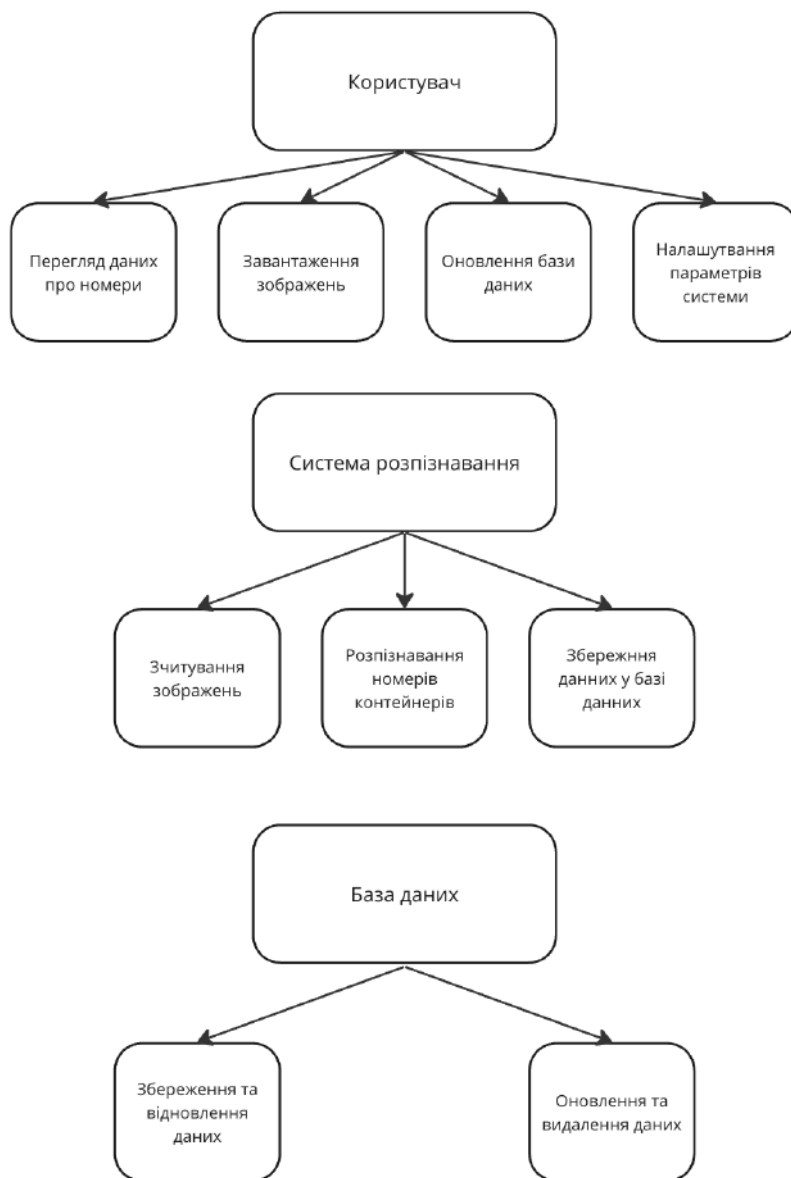


Рисунок 2.5 – Функціональність системи та взаємодію її компонентів з користувачами

З діаграми варіантів використання видно, що користувач може здійснювати різні дії, такі як перегляд даних про номери контейнерів,

завантаження зображень, оновлення бази даних та налаштування параметрів системи. Система розпізнавання номерів контейнерів зі свого боку може здійснювати зчитування зображень, розпізнавання номерів контейнерів та збереження даних у базі даних. База даних відповідає за збереження та відновлення даних, а також за їх оновлення та видалення.

Діаграма варіантів використання допомагає відобразити функціональність системи та взаємодію її компонентів. Окрім того, можна скласти діаграму використання системи (system use case diagram), яка допоможе уявити, як саме система працює та взаємодіє з іншими системами та компонентами.

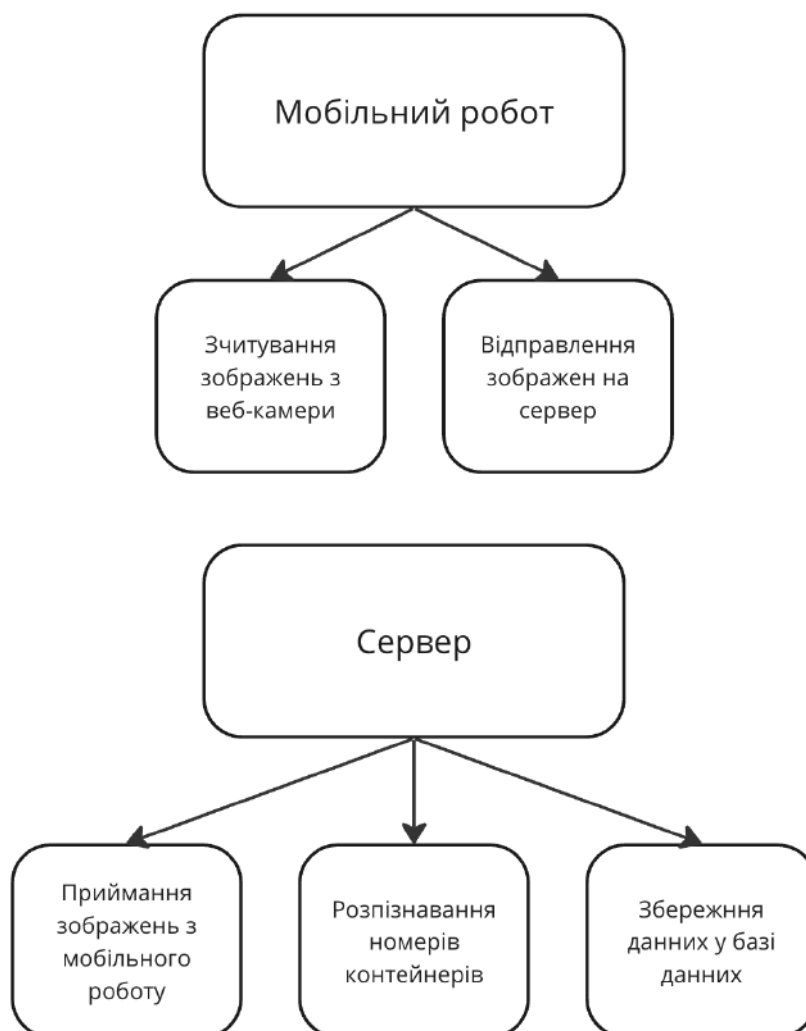


Рисунок 2.6 – Взаємодіє з іншими системами та компонентами

З діаграми використання системи видно, що мобільне колесо здійснює зчитування зображень з веб-камери та відправляє їх на сервер для подальшої обробки. Сервер в свою чергу приймає зображення, розпізнає номери контейнерів та зберігає дані у базі даних. База даних відповідає за збереження, відновлення, оновлення та видалення даних. Таким чином, діаграма використання системи допомагає краще зрозуміти, як саме система працює та взаємодіє з іншими системами та компонентами. Комбінація діаграм використання та варіантів використання допомагає зрозуміти функціональність та взаємодію систем.

Діаграма використання системи (system use case diagram) доповнює діаграму варіантів використання, показуючи більш детальну картину взаємодії системи з зовнішніми компонентами та користувачами.

Діаграма використання системи показує, що веб-камера розпізнає номер контейнера на зображенні та передає результати розпізнавання в систему. Система оброблює дані та зберігає інформацію про розпізнані контейнери в базі даних. Водій може запустити процес розпізнавання та переглянути результат, оператор може переглядати всю інформацію про розпізнані контейнери та оновлювати базу даних.

2.2 Логічна модель системи

Логічна модель системи розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі має бути розбита на декілька основних кроків.



Рисунок 2.7 – Логічна модель системи розпізнавання номерів контейнерів за допомогою веб-камери.

Для захоплення відеопотоку з веб-камери можна використовувати OpenCV бібліотеку. Операції захоплення відео будуть виконуватись в циклі.

Зображення, отримане з веб-камери, буде піддане обробці. Починаючи з простих методів, таких як зменшення шуму, розмиття і збільшення контрастності, та закінчуючи складнішими алгоритмами, такими як видалення перешкод, корекція рівномірності та зменшення дисторсії.

Після обробки зображення, система визначить область, в якій повинен бути розпізнаний номер. Це можна зробити за допомогою нейромержі, яка знаходить область інтересу.

Після виділення області інтересу, система проводить процес розпізнавання.

Далі після успішного розпізнавання, система повідомляє результат.

Можливо просто відображення на екрані, або надіслання інформації про контейнер на сервер, або в базу даних.

У системі використовуються різні математичні моделі для обробки зображень і розпізнавання номерів контейнерів.

Моделі фільтрації шуму: використовуються для видалення шуму зі зображення і покращення якості зображення. До таких моделей можна віднести фільтр Гаусса, медіанний фільтр, фільтр Білатеральної фільтрації тощо.

Моделі обробки зображень: використовуються для вирішення різних задач обробки зображень, наприклад, для збільшення контрастності, зменшення шуму, розмиття зображення, видалення перешкод тощо. До таких моделей можна віднести фільтр Собеля, алгоритм Canny, метод розмиття Гауса, адаптивне програмування та інші.

Моделі машинного навчання: використовуються для класифікації зображень і розпізнавання номерів контейнерів. До таких моделей можна віднести нейронні мережі, наприклад, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), а також класифікатори, такі як Support Vector Machines (SVM) і Random Forest.

Моделі обробки даних використовуються для обробки та аналізу результатів розпізнавання. До таких моделей можна віднести алгоритми обробки тексту, аналізу даних та статистичний аналіз.

Крім того, в системі використовуються нейронні мережі для розпізнавання символів. Зображення номера контейнера подається на вхід нейронній мережі, яка використовує алгоритм зворотного поширення помилки для розпізнавання символів. Для вирішення задачі встановлення номера контейнера на відео з веб-камери використовуються алгоритми комп'ютерного зору, такі як алгоритм Люкаса-Канаде для відстеження об'єктів та алгоритми розпізнавання об'єктів на зображенні.

Фізична модель розпізнавання за допомогою веб-камери, встановленої на поворотній платформі роботу, може бути описана наступним чином:

- мобільна платформа з встановленою Raspberry Pi, веб-камерою та іншими необхідними компонентами для збору даних та обробки зображень. поворотна платформа може бути оснащена додатковими пристроями, такими як GPS-приймач, щоб забезпечити точність визначення місцезнаходження;
- цифрова веб-камера, яка встановлюється на мобільну платформу та призначена для захоплення зображення з контейнерів. Вона може бути налаштована на автоматичну фокусування та регулювання експозиції для отримання якісних зображень;
- одноплатний комп'ютер, який використовується для обробки даних з веб-камери та виконання алгоритмів розпізнавання. Raspberry Pi має вбудований модуль Wi-Fi, що дозволяє передавати дані з колеса на сервер для подальшої обробки;
- джерело живлення для мобільної платформи та всіх її компонентів. Акумулятор повинен забезпечувати достатню енергію для тривалої роботи системи на полі;
- захисний корпус для мобільної платформи, щоб захистити її компоненти від пилу, вологи та ударів. Корпус може бути зроблений з міцних матеріалів, таких як метал або пластик;
- комп'ютерний сервер, який використовується для збереження даних та обробки результатів розпізнавання. Сервер може бути підключений до Інтернету, щоб забезпечити можливість віддаленого доступу до даних та результатів розпізнавання. Дані, отримані від веб-камери можуть бути передані на сервер для подальшої обробки. Це дозволяє зберігати дані в централізованому місці та забезпечує легкий доступ до них з будь-якого місця з підключенням до Інтернету. Для забезпечення безпеки даних та обробки результатів розпізнавання на сервері можна використовувати різні методи захисту, такі як шифрування, аутентифікацію, авторизацію, аудит, контроль доступу та інші. Також важливо забезпечити резервне копіювання даних на сервері.

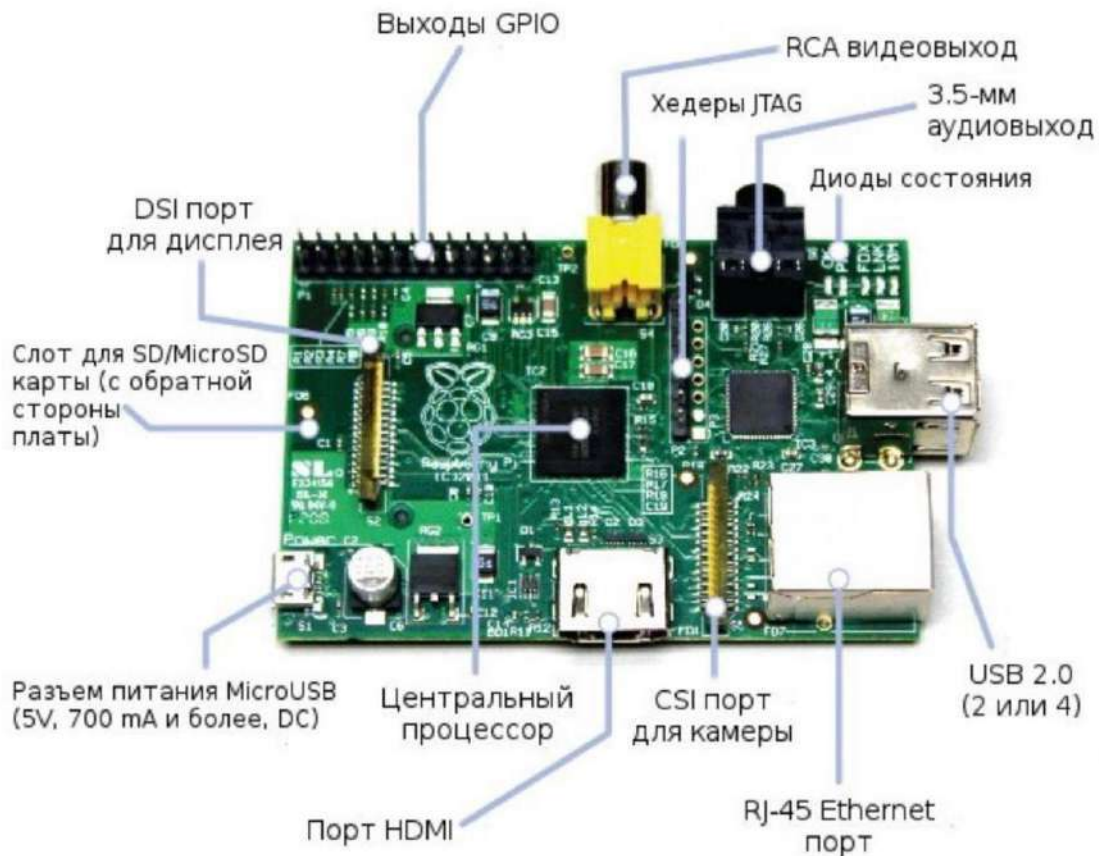


Рисунок 2.8 – Пристрій Raspberry Pi

Raspberry Pi – це одноплатний комп'ютер, який створений з метою навчання програмуванню та створення електронних проектів. Для зменшення розміру плати, вона була спроектована без клавіатури, миші, екрану та інших периферійних пристроїв.

Плата має розміри близько 85 мм на 56 мм, і на ній розміщені всі компоненти необхідні для роботи комп'ютера, включаючи процесор, оперативну пам'ять, зберігання на флеш-пам'яті, адаптер бездротового зв'язку Wi-Fi та Bluetooth, порти USB, HDMI, Ethernet, аудіо виходи та інші.

Основним процесором на Raspberry Pi є ARM-процесор, який може працювати на швидкості від 700 МГц до 1,4 ГГц в залежності від моделі. В залежності від версії, плата може мати від 512 МБ до 8 ГБ оперативної пам'яті, що дозволяє виконувати багатозадачні операції та працювати з великою кількістю даних.

Операційна система Raspberry Pi, яка зазвичай встановлюється на SD-

карту, є Linux-подібною, що дозволяє використовувати різноманітний програмний забезпечення, включаючи Python, Scratch та інші мови програмування.

2.3 Моделювання в середовищі MatLab

MatLab — пакет прикладних програм для числового аналізу, а також мова програмування, що використовується в даному пакеті. Система створена компанією The MathWorks і є зручним засобом для роботи з математичними матрицями, малювання функцій, роботи з алгоритмами, створення робочих оболонок (user interfaces) з програмами в інших мовах програмування. Хоча цей продукт спеціалізується на чисельному обчисленні, спеціальні інструментальні засоби працюють з програмним забезпеченням Maple, що робить його повноцінною системою для роботи з алгеброю. MATLAB має більше, ніж мільйон користувачів на виробництвах і науковців.

Розпізнавання номерів контейнеру є легким завданням, якщо він розташований горизонтально, має рівномірне освітлення, помітні межі чи чисту поверхню. Але на практиці такі умови зустрічаються рідко. Найчастіше освітлення нерівномірне, на зображенні з камери є різні шуми, а сам номер може бути розташований під кутом до горизонту, або покритий брудом, що ускладнює не лише розпізнавання символів, а й визначення його місцезнаходження на зображенні.

Для реалізації використовуватимемо можливості математичного пакету MatLab.

Моделювання процесу розпізнавання номерів контейнерів можна провести за допомогою наступних етапів:

- завантаження зображення з веб-камери на комп'ютер з встановленим MatLab;
- передача зображення на обробку в нейронну мережу;
- розпізнавання номеру на зображенні з допомогою нейронної мережі.

– відображення результатів розпізнавання номеру на екрані.

У MatLab можна використовувати готові бібліотеки для роботи з нейронними мережами, наприклад, Deep Learning Toolbox. Для навчання нейронної мережі необхідно мати набір зображень з відомими номерами контейнерів. Такий набір можна скласти самостійно або використати готові набори даних, наприклад, з відкритих джерел, таких як Open Images Dataset або COCO Dataset.

У процесі моделювання можна виконувати настройку параметрів нейронної мережі, таких як кількість шарів, кількість нейронів у кожному шарі, кількість епох навчання тощо. Також можна застосовувати різні методи покращення результатів, такі як data augmentation, dropout і batch normalization.

Нехай у систему розпізнавання надходить скориговане за допомогою алгоритму Харріса зображення номера контейнера, яке перетворимо на бінарний вигляд. Однією з важливих завдань під час проведення нейромережевого моделювання є формування навчальної вибірки для нейронної мережі. Для цього підготуємо шаблони літер та цифр виду (рис 2.9), перетворимо їх у бінарний вигляд та підготуємо матрицю шаблонів.



Рисунок 2.9 – Шаблон літери «В» та цифри «2»

Реалізуємо функцію сегментації літери на зображенні. На цьому етапі відбувається поділ знайденого номера на окремі символи за допомогою побудови роздільників.

Для класифікації та розпізнавання використовуватимемо нейронну мережу. На вхід нейронної мережі надходить сегмент зображення, отриманий

на етапі вилучення об'єктів.

Основний етап роботи з нейромережею – це її навчання. При навчанні відбувається прогін безлічі прикладів через мережу з корекцією ваг нейронів та із зазначенням значень, що подаються на вхід, та значень, які мають бути отримані на виході нейромережі.

Для навчання нейронної мережі використовуємо Matlab Neural Network (рисунок 2.9), де вхідними даними буде набір шаблонів, таргет дата – сегмент зображення. Вихідними даними буде символ.

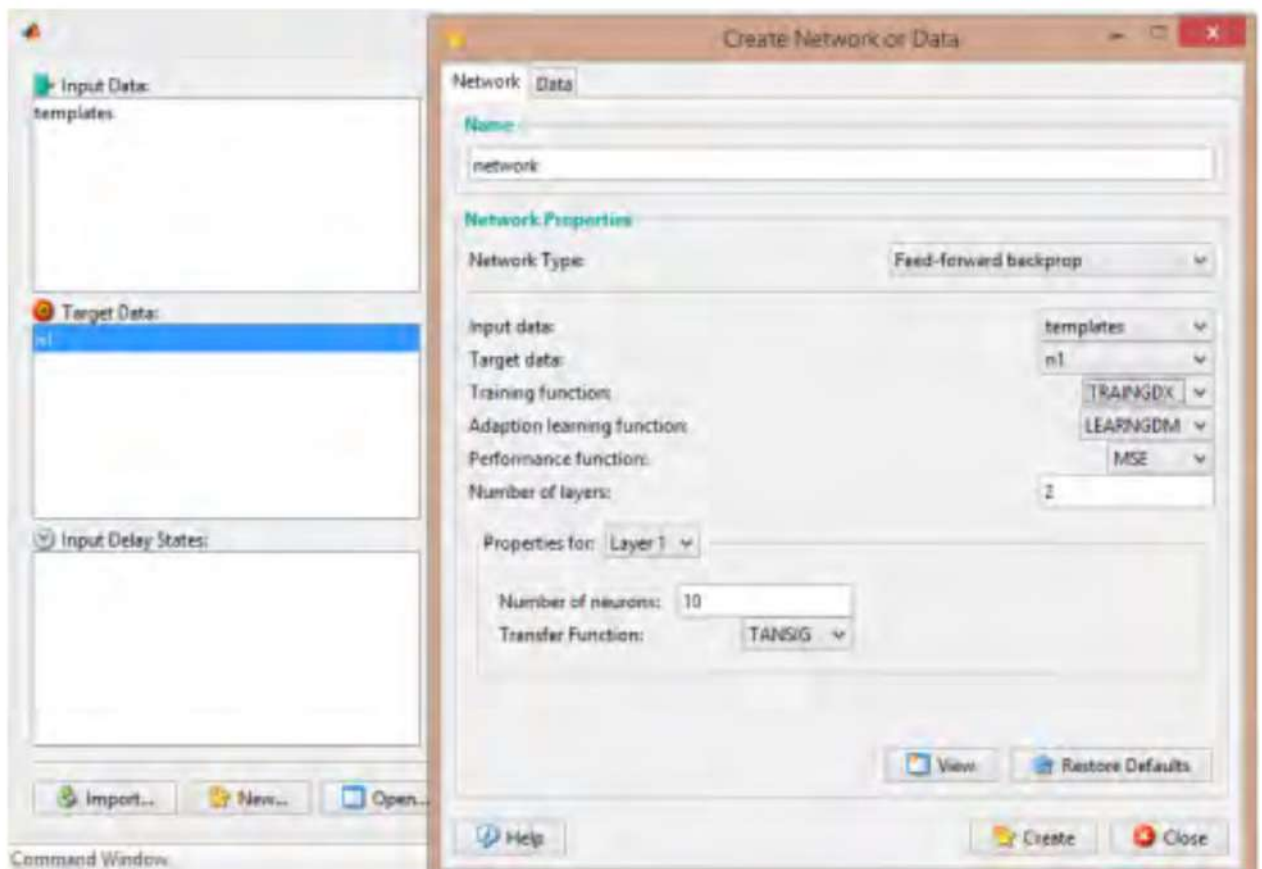


Рисунок 2.10 – Matlab Neural Network

Розробимо процес навчання мережі (рисунок 2.11)

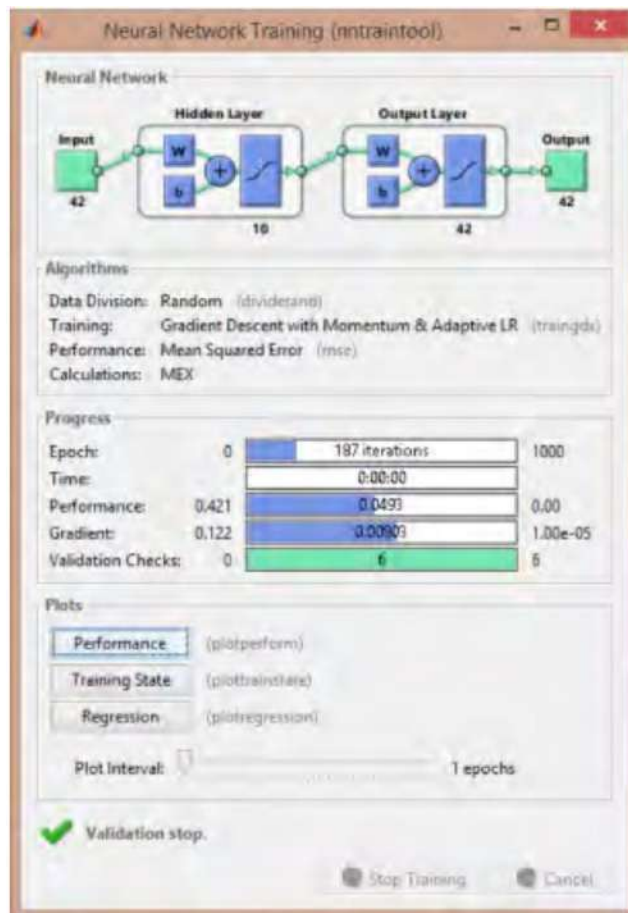


Рисунок 2.11 – Навчання нейронної мережі

У результаті отримаємо навчену нейронну мережу. Перевіримо роботу мережі.

Подаючи на вхід зображення номера контейнера, отримаємо розпізнане зображення (рисунок 2.12).

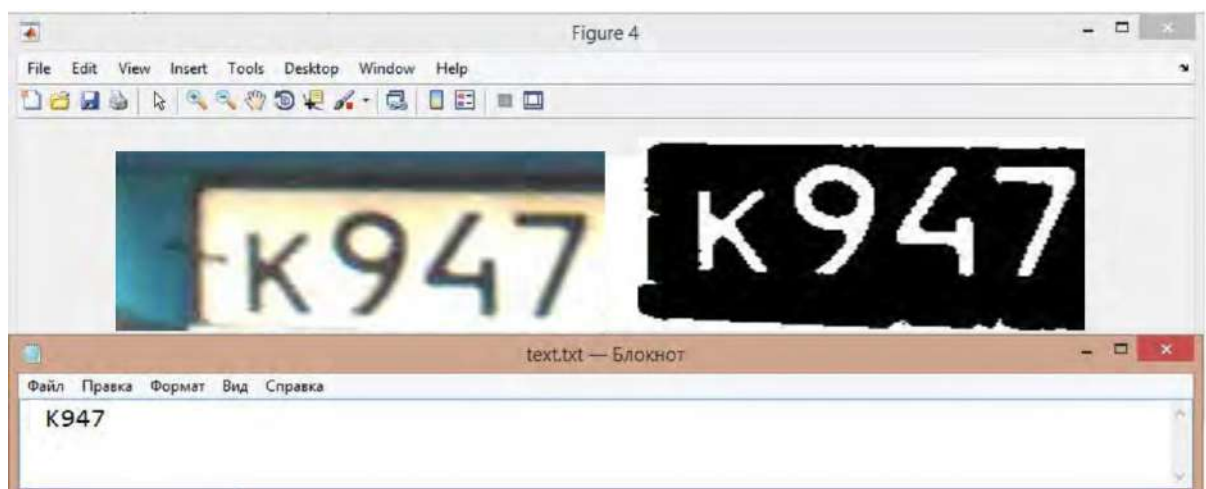


Рисунок 2.12 – Робота нейронної мережі

3 РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 Реалізація системи

Архітектура штучного оточення «Склад контейнерів». Створюється штучне оточення на прикладі автоматизованого сховища з різними секціями для вантажних контейнерів. Загальні вимоги до штучного оточення наступні.

У секціях сховища встановлені маркери відповідних номерів вантажних місць, де зберігаються контейнери (ідентифікатори місця секції) та системи освітлення для забезпечення якісного зображення номеру контейнера у будь-який час доби [15]. Для забезпечення безперебійної роботи МР, необхідно мати доступ до електричної мережі (позиції зарядки акумуляторних батарей, зарядної станції), а також бездротову мережу зв'язку, щоб передавати отримані дані для подальшої обробки та аналізу. Важливо також щоб МР мав права доступу до системи управління сховищем, щоб забезпечити швидкий й безпечний доступ к хмарним сховищам даних.

Для виконання основних функцій МР створюється штучна секція сховища для вантажних контейнерів. Приміщення секції поділяється на 3 основні зони з маршрутами «А-а», «А-б» та «А-с» та нумеровані вантажні місця від нуля до шести. Робот може курсувати між рядами, до будь-якого з контейнерів за маршрутом зображеним на рисунку 1. Підлога має розмітку для орієнтування МР. Робот, позначений на рисунку літерою «MR», має доступ до кожного вантажного місця. Вони позначені пунктирною лінією – це є місце де, як вважається, контейнери мають стояти. Надписом «СН» позначена зарядна станція робота, яка винесена у окрему зону у приміщенні. Секція має одну зарядну станцію. МР може обслуговувати декілька секцій упродовж дня, періодично заряджаюсь між патрулюванням та перевітками. Для експериментів обрано два типу МР з різним апаратним забезпеченням. Перший тип МР на базі чотирьох колісного повнопривідного шасі Multi Chassis-4WD Robot Kit ATV,

мікроконтролеру.

Другий тип МР реалізований на основі Raspberry Pi навчальний набір (PiCar-V для побудови роботів з відкритим вихідним кодом, оснащений ширококутною USB веб-камерою та укомплектований трьома новими платами мікроконтролеру Raspberry Pi, яка має вбудований порт для камери з високою роздільною здатністю що дозволяє працювати з відео та фото без додаткових інструментів або пакетного програмного забезпечення). Плата має достатньо велику оперативну пам'ять для складних обчислень, такі як зчитування інформації з зображення у режимі реального часу. Операційна система Linux прискорює процес розробки окремих компонентів системи на більш релевантних для типу задачі мовах.

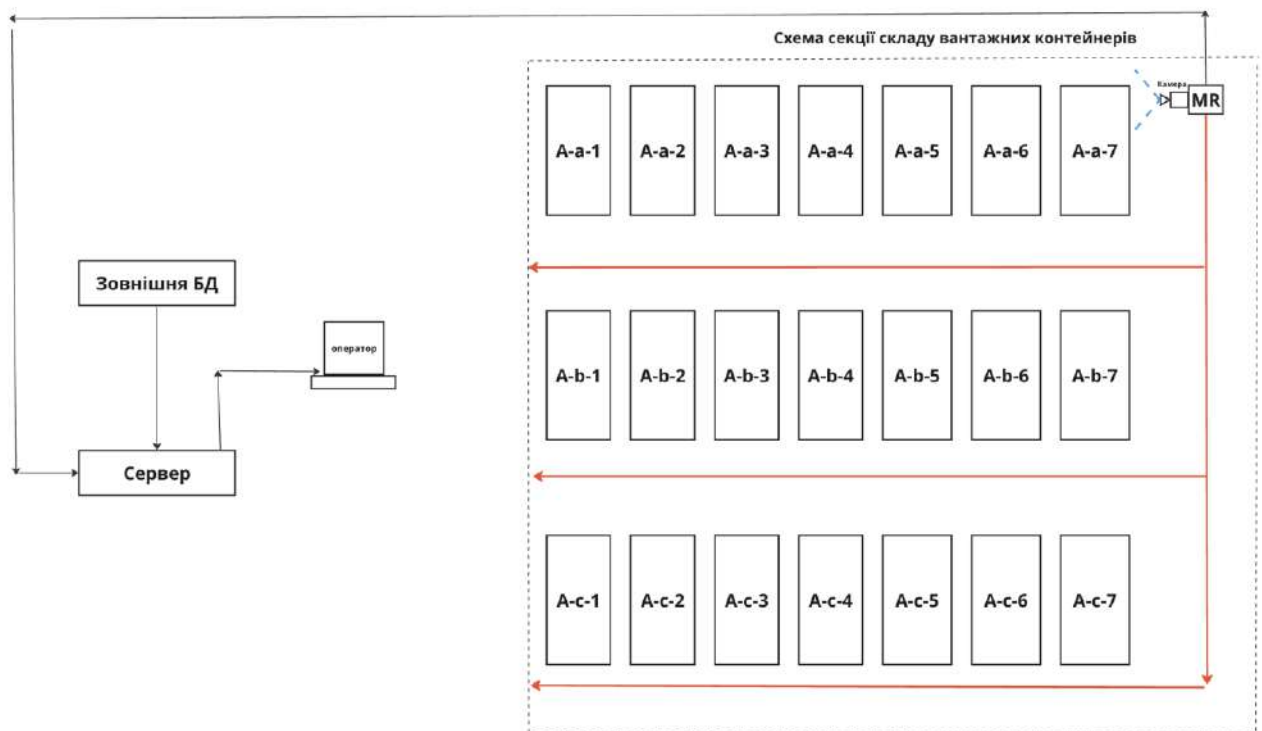


Рисунок 3.1 – Схема секції складу вантажних контейнерів

Найбільш оптимальним вибором є плата Raspberry Pi. Вона має багато переваг, таких як:

- можливість виконання складних розрахунків, наприклад зчитування інформації з зображення в реальному часі, забезпечується наявністю значної кількості оперативної пам'яті;

- наявність вбудованого Wi-Fi та Bluetooth: Raspberry Pi 4 має вбудований Wi-Fi та Bluetooth, що дозволяє підключати його до бездротових мереж та інших пристроїв без необхідності використання додаткових адаптерів.
- підтримка розробки на будь-якій мові програмування, яку можна відтворити на Linux, що дозволяє розробляти окремі компоненти до конкретної задачі, що сприяє швидшому процесу розробки;
- наявність GPIO портів: Raspberry Pi 4 має GPIO порти, які дозволяють підключати до нього датчики та інші пристрої з різними інтерфейсами, такими як I2C, SPI та UART;
- вбудований роз'єм для камери, що дозволяє підключати до нього камеру без необхідності використання додаткових пристроїв.

Фізичну модель системи наведено нижче (рисунок 3.2).

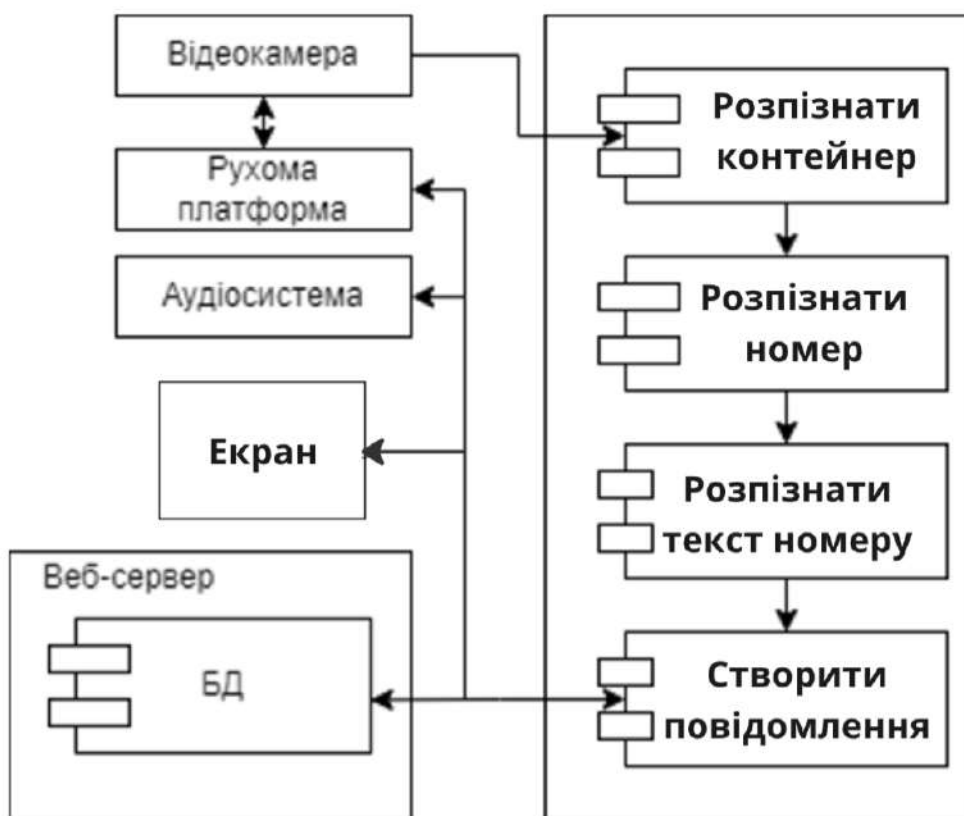


Рисунок 3.2 – Фізична модель системи

Камера, знімає відео, яке потім обробляється на Raspberry Pi.

Використовуючи OpenCV, програма знаходить зображення номера контейнера на відео та вирізає цей фрагмент з відео, зберігаючи як окремий файл зображення.

Далі, за допомогою Haar Cascade класифікатора, OpenCV може використовувати навчальну модель, щоб розпізнати номер на зображенні. Якщо він був розпізнаний успішно, програма може зберегти цю інформацію для подальшої обробки, наприклад, відправки на сервер або виводу на екран.

Використання OpenALPR дозволяє більш точно розпізнавати номери контейнерів, оскільки ця бібліотека містить навчальні моделі та алгоритми розпізнавання, що дозволяє отримати більш точні результати

Після отримання зображення з камери, виконується детекція номера контейнера за допомогою Haar Cascade Classifier. Якщо номер контейнера був успішно визначений, він передається до OpenALPR для розпізнавання. OpenALPR використовує нейронну мережу та алгоритми комп'ютерного зору, щоб перетворити його в текстовий формат.

Після цього результат розпізнавання номеру контейнера відображається на екрані і передається до програмного забезпечення Raspberry Pi для подальшої обробки або збереження.

Для реалізації системи розпізнавання номерів контейнерів за допомогою веб-камери на мобільному колісному роботі можуть використовуватися такі програмні компоненти:

- зчитування відеопотоку з веб-камери за допомогою OpenCV та TensorFlow;
- обробка зображення та виявлення номерів контейнерів на зображенні, реалізована за допомогою нейронної мережі;
- компонент для передачі даних з веб-камери на сервер. Для надсилання REST-запиту з Raspberry Pi 4 можна використати Python - бібліотеку requests. Перед початком використання цієї бібліотеки необхідно її встановити. Для цього у терміналі Raspberry Pi виконується команда «pip install requests». Після встановлення бібліотеки можна робити надсилання запиту REST;

- серверна частина, яка відповідає за збереження та обробку даних. За допомогою Django ORM (Object-Relational Mapping), можна взаємодіяти з базою даних та зберігати результати розпізнавання номерів контейнерів у відповідній таблиці. Додаток Django може також забезпечувати доступ до цих даних через веб-інтерфейс або API, що дозволить іншим додаткам та користувачам звертатися до цих даних за допомогою стандартних HTTP-запитів. Для цього можна використовувати функції Django, такі як Django REST framework, який надає готові інструменти для створення API, які дозволяють працювати з базою даних та обмінюватися даними з іншими додатками;
- база даних, в якій зберігаються результати розпізнавання номерів контейнерів;
- компонент для відображення результатів розпізнавання на веб-інтерфейсі для користувача;
- модуль керування мобільним колісним роботом, який відповідає за рух робота до контейнерів та збір даних з камери. Він складається з декількох компонентів, які забезпечують його функціональність. Одним з основних компонентів є мікроконтролер, який відповідає за керування рухом робота. Рух робота здійснений за допомогою різних методів, таких як дистанційне керування або автономне керування на основі алгоритмів маршрутизації. Для збору даних з камери використовується веб-камера, яка підключена до Raspberry Pi 4. Для керування веб-камерою та збору зображень використовується бібліотека OpenCV. Модуль також може містити різні сенсори, такі як датчики відстані, які допомагають роботу уникати перешкод та збирати додаткові дані про контейнери. Всі зібрані дані з камери та сенсорів можуть бути оброблені та збережені в базі даних, що забезпечується фреймворком Django.

3.2 Опис використаних нейромереж

Штучна нейронна мережа – математична модель, принцип роботи якої нагадує роботу мережі біологічних нейронів. Кожен шар представлений множиною вузлів – штучних нейронів, які видобувають ознаки все більшого рівня, поки останній шар не скомбінує ці ознаки, щоб зробити передбачення:

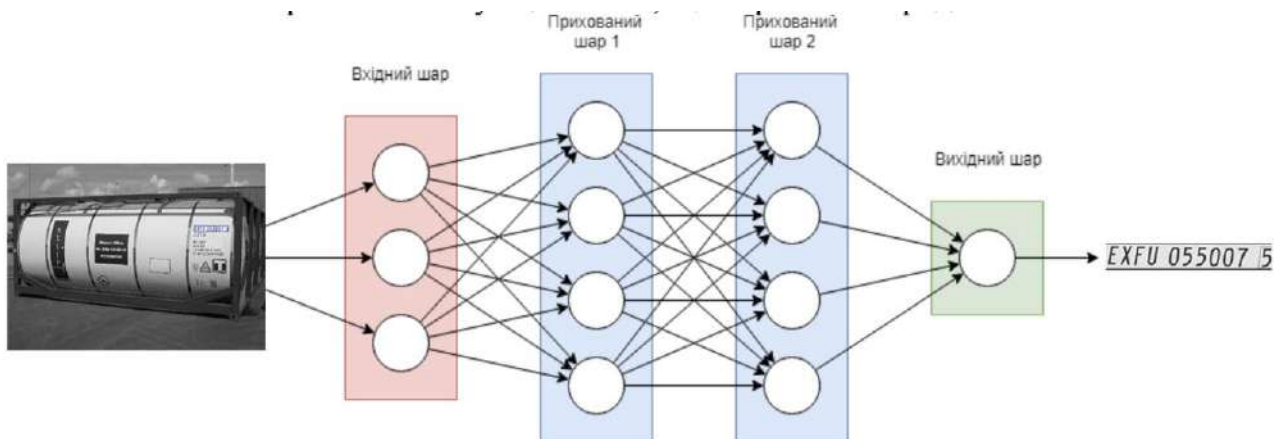


Рисунок 3.3 – Побудова багатошарової нейронної мережі

Штучний нейрон – це модель, яка складається із шару вхідних сигналів, кожен із яких має вагу. Ваги відображають значимість кожного входу штучного нейрону. На основі вхідних активацій та їх ваг у вихідний шар обчислюється сумуюча функція, яка враховуючи вхідні сигнали та їх ваги обчислює деяке значення. Далі результат сумуючої функції передається в активаційну функцію, яка в залежності від значення сумуючої функції обчислює вихідний результат нейрона

У додатку А представлений фрагмент коду, який відповідає за знаходження номера контейнера на зображенні.

Цей код - фрагмент програмного коду на мові програмування Python з використанням бібліотеки Keras, яка є фреймворком для глибокого навчання.

Представлена реалізація нейронної мережі з архітектурою U-Net для задачі розпізнавання номерів контейнерів на зображеннях. У загальному вигляді, U-Net складається з двох частин: енкодера та декодера.

Енкодер складається з кількох блоків, кожен з яких зазвичай містить згортковий шар, функцію активації та шар з підсумовуванням (зазвичай MaxPooling), який зменшує розмір зображення.

Декодер також складається з кількох блоків, кожен з яких містить шар зворотного відтворення збільшення зображення (транспоновані згорткові шари), функцію активації та шар конкатенації, який об'єднує вихідні дані з енкодера та декодера.

В даному кодї, енкодер складається з чотирьох блоків, кожен з яких містить два згорткових шари та функцію активації ReLU. У кожному блоку також використовується шар BatchNormalization для нормалізації виходу з попереднього шару. Після останнього блоку енкодера виконується згортковий шар, що зменшує кількість каналів. Вихідні дані цього шару передаються в якості входу у декодер.

Декодер використовує шар зворотного відтворення збільшення зображення та шар BatchNormalization для нормалізації виходу. Після кожного такого шару виконується операція конкатенації, яка об'єднує вихідні дані з відповідного блоку енкодера та відповідного блоку декодера.

Для кожного блоку використовується два згорткові шари, з максимальним розміром ядра 4x4. Після кожного згорткового шару застосовується функція активації ReLU, а потім застосовується операція BatchNormalization. Потім використовується MaxPooling2D для скорочення розміру зображення та збільшення його резолюції. Після четвертого блоку використовується операція Conv2DTranspose для збільшення розміру зображення.

Після цього застосовується операція конкатенації для об'єднання попереднього блоку з відповідним блоком внизу UNet. Після цього застосовуються додаткові згорткові шари, а також операції BatchNormalization та ReLU.

На останньому шарі використовується функція активації softmax для визначення ймовірності належності зображення до кожного класу. Цей шар визначає кількість класів, в даному випадку - num_classes, що передається в якості вхідного параметра у функцію unet_carnum().

На даному етапі ми отримали тензор о відповідного розміру (зазвичай

(`batch_size`, `num_classes`, `h`, `w`)). Для кожного зображення в батчі `o[i]` містить ймовірності належності пікселів до різних класів. Остаточний результат - індекс класу з найбільшою ймовірністю для кожного пікселя - можна отримати за допомогою методу `argmax()`.

Функція `UNET_CARNUM()` призначена для створення моделі мережі U-Net для розпізнавання номерів контейнерів на зображеннях. У моделі використовуються чотири блоки конволюційних шарів з підрівнями `MaxPooling2D`, два блоки виконують зменшення розміру входу вдвічі, а два інші - вчетверо. Потім виконується процес розширення зображення за допомогою `Conv2DTranspose` шарів, щоб отримати остаточний результат того ж самого розміру, що й вхідне зображення.

Ця модель призначена для розпізнавання номерів контейнерів на зображеннях, які можуть бути взяті з камер спостереження або в інших умовах, де зображення може бути перекрито будь-якими об'єктами, в тому числі іншими контейнерами.

Отже, описана архітектура моделі UNet, яка використовується для сегментації зображень, зокрема, для знаходження номерів контейнерів".

Після тренування моделі, ми отримаємо на виході зображення з класифікованими пікселями, які мають відповідну ймовірність належності до класу "номер".

У додатку Б представлений фрагмент коду, який показує структуру нейромережі розпізнавання символів номера контейнеру

Фрагмент програмного коду на мові програмування Python призначеною, для розпізнавання символів номера контейнеру. Нейромережа приймає зображення з розміром, вказаним у змінній `input_shape`, та виконує операції над ними з метою зменшення розмірності та витягнення важливих ознак. На вході нейромережі зображення проходить через шари з операціями згортки (`Conv2D`) та зворотньої згортки (`Conv2DTranspose`).

У першому шарі `Conv2D` задано параметри згортки з 16 фільтрами, розміром ядра 4 та кроком 2. Функція активації задана як "linear", тобто без

зміни значення ваг та зсувів. Далі використовується шар зворотньої свертки Conv2DTranspose з 16 фільтрами, розміром ядра 4 та кроком 1. Функція активації також задана як "linear". Далі використовується ще один шар зворотньої шгортки Conv2DTranspose з 16 фільтрами, розміром ядра 4 та кроком 1. У цьому шарі використовується гіперболічний тангенс як функція активації.

Після цього ще декілька шарів Conv2D з 16 фільтрами та функцією активації "linear". На останньому шарі використовується функція активації "relu", а також BatchNormalization для нормалізації даних.

Нарешті, Dense шар з функцією активації "softmax", яка використовується для отримання вихідного значення у вигляді розподілу ймовірностей між всіма можливими класами (символами номера контейнеру). Результатом роботи нейромережі є вихідний тензор з розміром (cls,).

Отриманий вихідний тензор представляє собою вектор імовірностей для кожного з класів, тобто для кожного символу номеру контейнера. Припустимо, що класи відповідають цифрам та літерам, які можуть з'явитися в номері контейнера. Тоді розмір вихідного тензору буде дорівнювати загальній кількості цих символів. Наприклад, для латинських літер та цифр загальна кількість символів становить 36

3.3 Опис результатів тестування

Тренування системи відбувається наступним чином. Після запуску відповідної команди програма автоматично перевіряє у поточній директорії наявність директорії data/, що має містити тестові файли з тренувальним набором даних. Якщо вони є, то починається навчання системи. Після старту навчання програма починає виводити дані про поточний крок навчання з інформацією про похибку і затрачений час на певний рок, тощо (рисунок 3.4).

```

Run id: YGABLZ
Log directory: /tmp/tflearn_logs/

Training samples: 64
Validation samples: 64
---
Training Step: 207191 | total loss: 0.03896 | time: 2.469s
| Adam | epoch: 97191 | loss: 0.03896 - acc: 0.9905 | val_loss: 10.61265 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207192 | total loss: 0.04998 | time: 1.318s
| Adam | epoch: 97192 | loss: 0.04998 - acc: 0.9899 | val_loss: 10.59973 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207193 | total loss: 0.04510 | time: 1.319s
| Adam | epoch: 97193 | loss: 0.04510 - acc: 0.9909 | val_loss: 10.56708 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207194 | total loss: 0.04161 | time: 1.323s
| Adam | epoch: 97194 | loss: 0.04161 - acc: 0.9918 | val_loss: 10.55289 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207195 | total loss: 0.03909 | time: 1.315s
| Adam | epoch: 97195 | loss: 0.03909 - acc: 0.9911 | val_loss: 10.54458 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207196 | total loss: 0.04123 | time: 1.321s
| Adam | epoch: 97196 | loss: 0.04123 - acc: 0.9904 | val_loss: 10.54430 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207197 | total loss: 0.03826 | time: 1.321s
| Adam | epoch: 97197 | loss: 0.03826 - acc: 0.9914 | val_loss: 10.54394 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207198 | total loss: 0.03813 | time: 1.321s
| Adam | epoch: 97198 | loss: 0.03813 - acc: 0.9907 | val_loss: 10.54339 - val_acc: 0.2656
- iter: 64/64
---
Training Step: 207199 | total loss: 0.03444 | time: 1.318s
| Adam | epoch: 97199 | loss: 0.03444 - acc: 0.9916 | val_loss: 10.54625 - val_acc: 0.2656
- iter: 64/64
---
Training Step: 207200 | total loss: 0.03151 | time: 1.324s
| Adam | epoch: 97200 | loss: 0.03151 - acc: 0.9924 | val_loss: 10.52518 - val_acc: 0.2656
- iter: 64/64
---
Run id: N1Y94X
Log directory: /tmp/tflearn_logs/

Training samples: 64
Validation samples: 64
---
Training Step: 207201 | total loss: 0.03787 | time: 2.492s
| Adam | epoch: 97201 | loss: 0.03787 - acc: 0.9916 | val_loss: 10.51123 - val_acc: 0.2500
- iter: 64/64
---
Training Step: 207202 | total loss: 0.03478 | time: 1.319s
| Adam | epoch: 97202 | loss: 0.03478 - acc: 0.9925 | val_loss: 10.48448 - val_acc: 0.2500
- iter: 64/64
---
[learn] 0:python* "ip-172-31-47-121" 14:22 02-Jun-17

```

Рисунок 3.4 – Вивід програми про поточний стан навчання нейромережевої системи

Усі дані, про навчання автоматично записуються в теку /tmp/rflearn_log. Використовуючи ці дані, за допомогою CLI додатку Tensorboard можна проаналізувати, як працює система (рисунок 3.5).

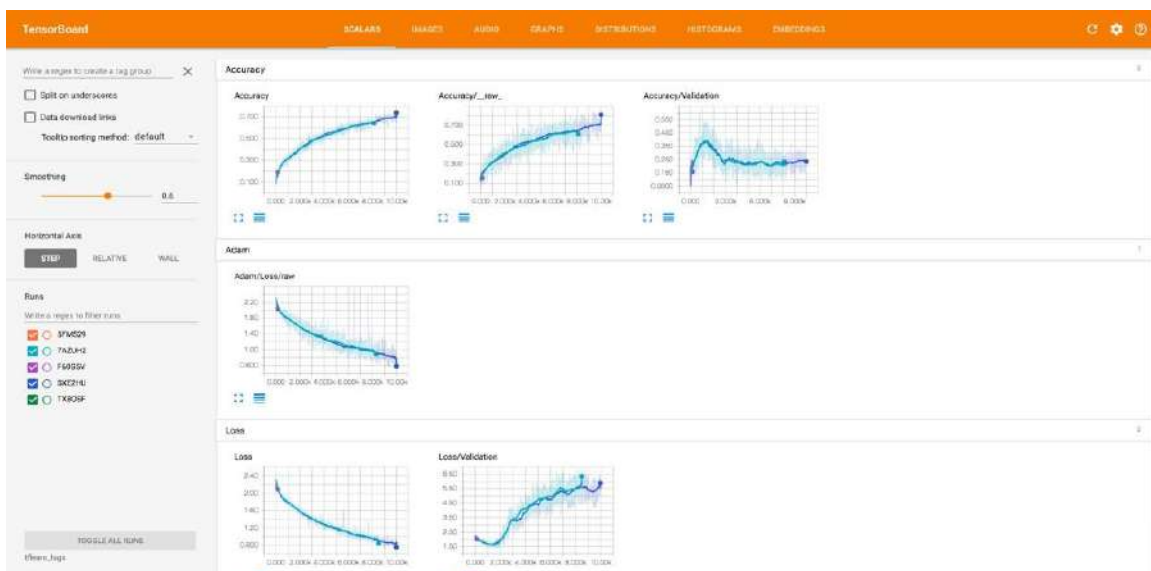


Рисунок 3.5 – Інформація про роботу нейромережевої системи

Також можна детально роздивитися структуру нейромережевої системи, що була створена (рисунок 3.6). Структура зображена у вигляді інтерактивного графу із згрупованими вершинами.

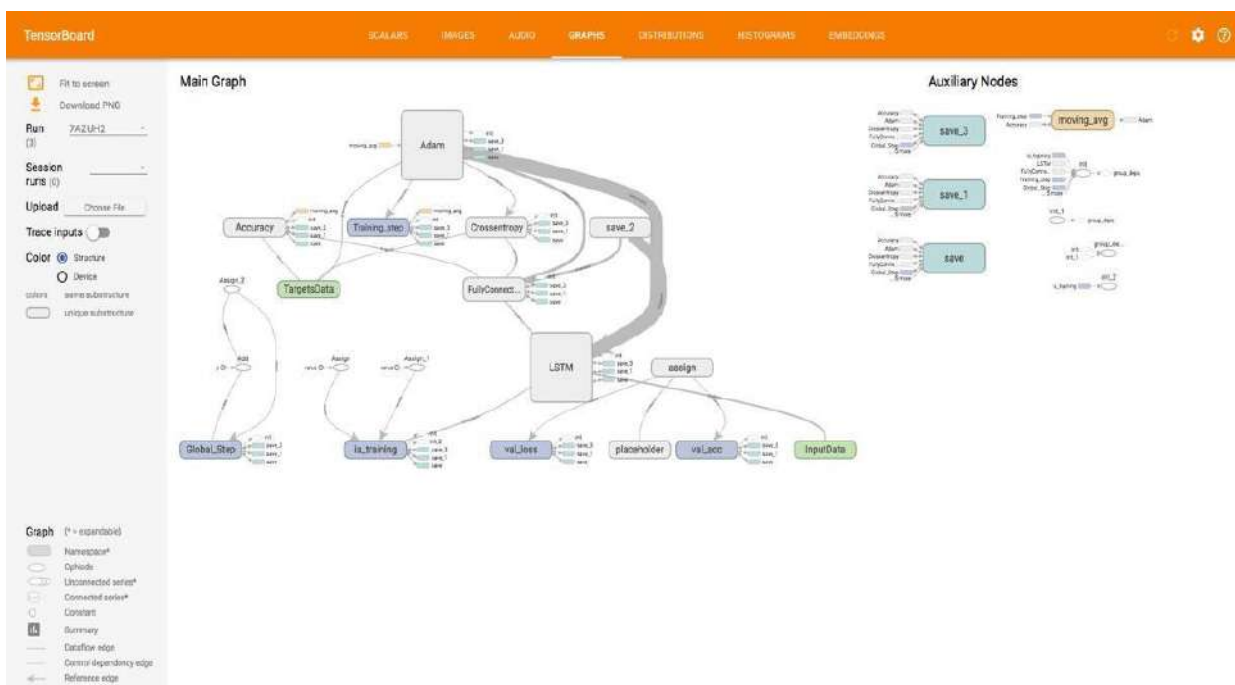


Рисунок 3.6 – Структура нейромережевої системи

В якості вхідного тестового набору візьмемо фотографії контейнерів із номерами (рисунок 3.7)



Рисунок 3.7 – Приклад фотографії з тестового набору

Після обробки фотографії системою отримуємо наступний результат зображений на рисунку 3.8.



Рисунок 3.8 – Результат роботи системи

ВИСНОВКИ

У ході кваліфікаційній роботі була розроблена система розпізнавання номерів контейнерів за допомогою веб-камери, встановленої на мобільному колісному роботі. Створено автоматизовану систему, яка здатна виявляти та розпізнавати номери у режимі реального часу.

У процесі роботи було проведено збір та обробку набору даних для розробки системи розпізнавання. Застосування нейронних мереж дало змогу досягти високої точності виявлення та розпізнавання номерів контейнерів.

Ефективність системи була перевірена на тестовому наборі даних, де вона продемонструвала задовільні результати. Результати свідчать про успішність розробленої системи у розпізнаванні номерів контейнерів за допомогою веб-камери на мобільному колісному роботі.

Отримані результати відкривають перспективи для подальшого вдосконалення системи та її застосування у реальних умовах, зокрема в логістичних та транспортних галузях.

Розроблена система розпізнавання номерів контейнерів є потужним інструментом для автоматизації процесу ідентифікації та відстеження контейнерів у реальному часі, що може покращити ефективність та безпеку в логістичних операціях. Вона має потенціал для використання в різних сферах, включаючи транспортну логістику, складське господарство, контроль за вантажем та безпеку. Вона може сприяти автоматизації та оптимізації процесів, що пов'язані з ідентифікацією та відстеженням контейнерів.

Також є потужний потенціал для розширення та покращення системи. Наприклад, можна розглянути можливість інтеграції з додатковими сенсорами або використання більш потужних обчислювальних ресурсів для поліпшення швидкості та точності розпізнавання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Функціональне призначення та класифікація складів в логістиці // WareTeKa News, 13.11.2020, <https://wareteka.com.ua/uk/blog/klasyfikaciya-skladiv-v-logistyci/>
2. Rasmussen S, Kingston D, Humphrey L (2018) Brief Introduction to Unmanned Systems Autonomy Services (UxAS). Int. Conf. on Unmanned Aircraft Sys. (ICUAS). <https://doi.org/10.1109/ICUAS.2018.8453287>
3. Ярушкіна Н.Г. Основы теории нечетких и гибридных систем. Учебное пособие / Ярушкіна Н.Г. – Финансы и статистика, 2004.- 320 с
4. Бурдаков С.Ф. Системы управления движением колесных роботов / Бурдаков С.Ф., Мирошник И.В., Стельмаков Р.Э. – СПб.:Наука, 2001. – 229с.
5. Батанов А. Ф., Грицынин С. Н., Муркин С. В. Робототехнические комплексы для обеспечения специальных операций // Специальная Техника. №6. – 1999. – С. 10–17.
6. Стругайло В.В. — Обзор методов фильтрации и сегментации цифровых изображений / В.В. Стругайло. — 2012.
7. Работа каскада Хаара в OpenCV: теория и практика [Электронный ресурс] — Режим доступа до ресурсу: <https://habr.com/ru/company/recognitor/blog/228195/>.
8. Shakhnarovich, Darrell, and Indyk — Nearest-Neighbour Methods in Learning and Vision / Shakhnarovich, Darrell, and Indyk // MIT Press. — 2005.
9. Арлазаров В.Л., Троянкер В.В., Котович Н.В. — Адаптивное распознавание Символов [Электронный ресурс] / В.Л. Арлазаров, В.В. Троянкер, Н.В. Котович. — 2000. — Режим доступа до ресурсу: <http://ocrai.narod.ru/adaptive.html>.
10. Ясницький Л. Н. — Введення в штучний інтелект/ Л. Н. Ясницький. — 2005.
11. Котович Н.В., Славин О.А. — Розпізнавання скелетних образів

[Електронний ресурс] / Н.В.Котович, О.А.Славин. — 2003. — Режим доступу до ресурсу: <http://ocrai.narod.ru/skeletrecognize.html>

12. Хайкин Саймон — Нейронные сети / Саймон Хайкин. — 2008.

13. A. Kargin, T. Petrenko, “Feeling Artificial Intelligence for AI-Enabled Autonomous Systems” in Conf. Proc. of 2022 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT) Alamein New City, Egypt, 18-21 December 2022, P.88-93..

14. Келер А., Брэдски Г. Изучаем OpenCV 3: навч. посіб. ДМК Пресс, 2017. 826 с.

15. Функціональне призначення та класифікація складів в логістиці // WareTeKa News, 13.11.2020, <https://wareteka.com.ua/uk/blog/klasyfikaciya-skladiv-v-logistyci/>

16. A. Kargin and T. Petrenko, “Spatio-Temporal Data Interpretation Based on Perceptual Model,” in Advances in Spatio-Temporal Segmentation of Visual Data. Studies in Computational Intelligence, V. Mashtalir, I. Ruban, V. Levashenko, Eds., vol. 876, Springer, Cham, 2020, pp. 101-159.