**Міністерство освіти і науки України**

**Луцький національний технічний університет**
(повне найменування закладу вищої освіти)

**Факультет комп'ютерних та інформаційних технологій**
(повне найменування факультету)

**Кафедра комп'ютерної інженерії та безпеки**
(повне найменування кафедри)

# КВАЛІФІКАЦІЙНА РОБОТА ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»

# СИСТЕМА ВІЗУАЛЬНОГО АНАЛІЗУ ДЛЯ АВТОМАТИЗАЦІЇ ВИРОБНИЧОЇ ЛІНІЇ

# VISUAL ANALYSIS SYSTEM FOR AUTOMATION PRODUCTION LINE

спеціальність 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія
(назва освітньої програми)

Виконав: здобувач вищої освіти
групи КІ-41
Верчук Владислав Русланович

(підпис)

Керівник:
к.т.н., доцент
Костючко Сергій Миколайович

(підпис)

Кваліфікаційну роботу
допущено до захисту
« 10 » червня 2025 р.
Гарант освітньої програми:
к.т.н., доцент
Лавренчук Світлана Василівна

(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет *комп'ютерних та інформаційних технологій*
Кафедра *комп'ютерної інженерії та безпеки*
Ступінь вищої освіти: *бакалавр*
Галузь знань: *12 Інформаційні технології*
Спеціальність: *123 Комп'ютерна інженерія*
Освітня програма: *«Комп'ютерна інженерія»*

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ доц. Тарас ТЕРЛЕЦЬКИЙ
« 10 » 01 2025 р.

# ЗАВДАННЯ
# НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Верчуку Владиславу Руслановичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи *Система візуального аналізу для автоматизації виробничої лінії*

Керівник роботи *к.т.н., доц. Костючко Сергій Миколайович*

затверджені наказом закладу вищої освіти від «04» січня 2025 року № 11/01-02
2. Строк подання здобувачем вищої освіти кваліфікаційної роботи 10.06.2025р.

3. Вихідні дані до роботи *Джерелом розробки є науково-технічна література та публікації в періодичних виданнях з даного питання, опубліковані зарубіжні та вітчизняні роботи в даній області та різні інтернет-ресурси технічного спрямування*

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):
*Вступ*
*Розроблення структурної схеми та алгоритму роботи автоматизованої системи*
*Розробка програмного забезпечення автоматизованої системи*
*Висновки*

5. Перелік графічного (ілюстративного) матеріалу:

6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| *Огляд проблеми створення системи та постановка завдань дослідження* | *Костючко С.М., доцент* | | |
| *Теоретичне обгрунтування та технічна реалізація системи* | *Костючко С.М., доцент* | | |
| *Проєктування та реалізація системи* | *Костючко С.М., доцент* | | |
| *Нормоконтроль* | *Багнюк Н.В., доцент* | | |
| *Гарант ОП* | *Лавренчук С.В., доцент* | | |
| *Показник запозичень тексту* | ***___%*** | | |
| *Академічна доброчесність* | *Міскевич О.І.,ст.викладач* | | |

7. Дата видачі завдання 10.01.2025 р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи | Строк виконання етапів роботи | Примітка |
|---|---|---|---|
| 1. | *Огляд літератури із досліджуваної проблеми, аналіз предметної області та наявних рішень* | до 10.02.2025 р. | Виконано |
| 2. | *Розроблення структурної схеми та алгоритму роботи автоматизованої системи* | до 02.03.2025 р. | Виконано |
| 3. | *Розробка програмного забезпечення автоматизованої системи* | до 02.04.2025 р. | Виконано |
| 4. | *Висновки та загальні результати дослідження* | до 10.04.2025 р. | Виконано |
| 5. | *Формування списку використаних джерел* | до 15.04.2025 р. | Виконано |
| 6. | *Формування додатків* | до 02.05.2025 р. | Виконано |
| 7. | *Оформлення ілюстративного матеріалу* | до 10.05.2025 р. | Виконано |
| 8. | *Нормоконтроль* | до 15.05.2025 р. | Виконано |
| 9. | *Інструментальна перевірка на академічний плагіат* | до 30.05.2025 р. | Виконано |
| 10 | *Представлення кваліфікаційної роботи бакалавра до захисту* | до 03.06.2025 р | Виконано |
| 11. | *Здача кваліфікаційної роботи та всіх супровідних документів на кафедру* | до 10.06.2025 р. | Виконано |

**Здобувач вищої освіти** _____ (підпис) Верчук В.Р. (прізвище, ініціали)

**Керівник кваліфікаційної роботи** _____ (підпис) Костючко С.М. (прізвище, ініціали)

# АНОТАЦІЯ

Верчук В.Р. Система візуального аналізу для автоматизації виробничої лінії. Рукопис.

Кваліфікаційна робота бакалавра ОП «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія. Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота складається з вступу, трьох розділів, висновків, списку використаних джерел, додатків.

У першому розділі здійснено аналіз сучасних методів комп'ютерного зору, промислових датчиків, виявлення переваг і недоліків існуючих підходів.

У другому розділі проведено розробку структурної схеми автоматизованої системи, здійснено вибір апаратного забезпечення (Raspberry Pi, камера), описано алгоритму роботи системи, визначено умов використання, побудовано моделі процесів.

У третьому розділі здійснено вибір програмного середовища (Python, PyCharm), розроблено та протестовано програмний код з використанням бібліотек OpenCV для обробки зображень, проведено калібрування системи, здійснено експерименти.

Об'єкт дослідження – система візуального аналізу для автоматизації виробничої лінії.

Предмет дослідження – методи, алгоритми та програмні засоби розробки систем комп'ютерного зору для автоматизованого аналізу та контролю продукції на виробничих лініях.

Метою роботи є проєктування та реалізація системи візуального аналізу для автоматизації виробничої лінії з використанням апаратно-програмних рішень, що дозволяє підвищити ефективність, точність і продуктивність контролю якості продукції.

Ключові слова: комп'ютерний зір, автоматизація виробництва, Raspberry pi, OpenCV, аналіз зображень.

# ABSTRACT

Verchuk V. Visual analysis system for automation production line. Manuscript.

Qualification work of the bachelor of the specialty "Computer Engineering" specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2025.

The qualification work consists of an introduction, three chapters, conclusions, a list of sources used, and appendices.

The first chapter shows the analysis of modern methods of computer vision, industrial sensors, machine vision systems, a review of industry solutions, identification of advantages and disadvantages of existing approaches.

In the second chapter, the structural diagram of an automated system is developed, the hardware is selected (Raspberry Pi, camera), the system operation algorithm is described, the conditions of use are determined, and process models are built.

In the third chapter, the software environment is selected (Python, PyCharm), the program code is developed and tested using OpenCV libraries for image processing, the system is calibrated, and experiments are carried out.

The research object is a visual analysis system for automating a production line.

The research subject is methods, algorithms and software tools for developing computer vision systems for automated analysis and control of products on production lines.

The purpose of the work is to design and implement a visual analysis system for automating a production line using hardware and software solutions (Raspberry Pi, Python, OpenCV), which allows to increase the efficiency, accuracy and productivity of product quality control.

Keywords: computer vision, production automation, raspberry pi, opencv, image analysis.

# ЗМІСТ

ВСТУП ..... 7
РОЗДІЛ 1 АНАЛІТИЧНА ЧАСТИНА ..... 9
1.1 Аналіз існуючих методів, засобів та автоматизованих систем формування бібліотеки зразків готової продукції підприємства ..... 9
1.2 Сучасні промислові датчики та системи технічного зору ..... 18
РОЗДІЛ 2 РОЗРОБЛЕННЯ СТРУКТУРНОЇ СХЕМИ ТА АЛГОРИТМУ РОБОТИ АВТОМАТИЗОВАНОЇ СИСТЕМИ ..... 21
2.1 Розроблення структурної схеми автоматизованої системи формування бібліотеки готової продукції підприємства ..... 21
2.2 Вибір компонентів системи для реалізації формування бібліотеки зразків готової продукції ..... 23
2.3 Алгоритм роботи автоматизованої системи формування бібліотеки зразків готової продукції підприємства ..... 26
2.4 Вибір інструментів для реалізації технології технічного зору ..... 30
РОЗДІЛ 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ ..... 33
3.1 Вибір програмного середовища ..... 33
3.2 Розроблення програмного забезпечення ..... 35
3.3 Проведення експерименту ..... 43
ВИСНОВКИ ..... 48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ..... 50

# ВСТУП

Сучасні системи машинного зору застосовуються для вирішення завдань промислової автоматизації в таких напрямках, як безконтактні вимірювання, виявлення дефектів, контроль якості та безперервний моніторинг у середовищі виробничих ліній.

Інтеграція систем машинного зору в автоматизовані процеси сприяє усуненню впливу так званого «людського чинника» та забезпечує повноцінний контроль кожного виробу на стадії виробництва.

Структурно такі системи включають об’єктиви, промислові відеокамери, освітлювальні модулі, контролери або обчислювальні пристрої з програмним забезпеченням для обробки та аналізу зображень, а також засоби підключення до виконавчих пристроїв і зовнішніх сенсорів.

Машинний зір є міждисциплінарною технологією, що ефективно використовується в сучасному інженерному проєктуванні в різних галузях – від металургії та машинобудування до виготовлення пакувальних матеріалів.

Сучасні високошвидкісні промислові камери з високою роздільною здатністю дають змогу реалізувати автоматизований контроль якості на високопродуктивних пакувальних, складських і сортувальних лініях, здійснювати вимірювання з точністю до кількох мікрон і виявляти дефекти, недоступні для людського зору.

Впровадження автоматизованих систем на основі машинного зору дозволяє істотно підвищити ефективність контролю якості продукції, зменшити витрати, прискорити процеси вимірювання й пошуку дефектів у режимі безперервного спостереження та усунути фактор суб’єктивного впливу.

Досягнення зазначених результатів можливе за умови коректного налаштування системи з урахуванням специфіки середовища та експлуатаційних умов, у яких вона функціонує.

Для досягнення поставленої в роботі мети були сформовані наступні завдання:

– провести аналіз існуючих методів і систем комп’ютерного зору, що використовуються для автоматизованого контролю якості на виробничих лініях;

– вибрати та обґрунтувати апаратно-програмну платформу (зокрема Raspberry Pi, камери, сенсори), яка підходить для реалізації системи візуального аналізу;

– розробити структурну схему системи, визначити основні компоненти та описати їхню взаємодію;

– скласти алгоритм роботи системи, який забезпечить точне розпізнавання об’єктів і автоматичний аналіз зображень у режимі реального часу;

– реалізувати програмне забезпечення на основі Python та бібліотеки OpenCV для обробки й аналізу вхідних візуальних даних;

– провести тестування системи в реальних умовах виробничої лінії, оцінити її точність, продуктивність і стабільність роботи.

# РОЗДІЛ 1
# АНАЛІТИЧНА ЧАСТИНА

## 1.1 Аналіз існуючих методів, засобів та автоматизованих систем формування бібліотеки зразків готової продукції підприємства

Область комп'ютерного зору є відносно молодою, багатогранною та динамічною галуззю досліджень. Лише з початку 1980-х років спостерігається інтенсивний розвиток у цьому напрямі, що стало можливим завдяки здатності комп'ютерів обробляти великі обсяги даних, зокрема візуальну інформацію, хоча перші напрацювання з'явилися ще раніше.

Варто зазначити, що ці дослідження зазвичай розпочинались у рамках суміжних дисциплін, і наразі відсутнє єдине загальноприйняте формулювання задачі комп'ютерного зору. Також немає уніфікованого підходу до визначення способів розв'язання цієї задачі [1].

Замість цього існує значна кількість методів, орієнтованих на вирішення конкретних задач у межах комп'ютерного зору, при цьому вибір методології часто визначається специфікою поставленої задачі та лише зрідка може бути узагальнений для ширшого спектра застосувань.

Хоча значна частина методів і технологій все ще перебуває на етапі базових досліджень, дедалі частіше вони інтегруються в комерційні рішення, де стають складовою великих систем, що застосовуються для вирішення комплексних завдань – наприклад, у медичній візуалізації або контролі якості в промисловому виробництві.

У більшості прикладних реалізацій систем комп'ютерного зору керуючі пристрої заздалегідь запрограмовані на виконання специфічних завдань, хоча підходи до розв'язання таких задач дедалі частіше ґрунтуються на знаннях та демонструють універсальність [2].

Автоматизоване планування та прийняття рішень у середовищах із керованими діями – такими як переміщення роботизованих систем або маніпуляторів – становить важливий компонент штучного інтелекту.

Цей тип обробки зазвичай потребує наявності вхідних даних, що формуються системами машинного зору, які функціонують як відеосенсори й забезпечують комплексну інформацію про середовище та технологічні процеси.

Серед суміжних напрямів варто відзначити розпізнавання образів та методи машинного навчання. Внаслідок цього комп'ютерний зір часто розглядається як складова частина штучного інтелекту або комп'ютерних наук.

Ще одним близьким напрямом є обробка сигналів. Багато методів, що використовуються для аналізу одновимірних (часових) сигналів, можуть бути адаптовані до обробки двовимірних або багатовимірних сигналів, характерних для систем машинного зору.

Проте через специфіку зображень було розроблено низку унікальних методів, властивих виключно галузі комп'ютерного зору, які не мають аналогів серед класичних підходів до аналізу одновимірних сигналів. Особливістю таких методів є їхня виражена нелінійність, яка у поєднанні з багатовимірністю даних, робить відповідну підгалузь обробки сигналів самостійним компонентом у структурі комп'ютерного зору.

Окрім наведених підходів до вирішення задач комп'ютерного зору, значна частина проблем вивчається у межах математичних дисциплін. Зокрема, методи базуються на статистичних підходах, техніках оптимізації або геометричних принципах. Інтенсивні дослідження проводяться у сфері практичного використання машинного зору, зокрема щодо програмної та апаратної реалізації існуючих методів або їх адаптації з метою досягнення високої продуктивності без значного збільшення обчислювальних ресурсів.

Сучасні технології широко застосовуються для обробки зображень у рамках автоматичного контролю, керування й вимірювання.

Промислові автоматизовані системи для отримання візуальної інформації включають сенсорні компоненти, які позначаються як системи машинного зору. Саме вони дозволяють роботизованим комплексам формувати зображення об’єктів, продукції або сцен, здійснювати їх цифрову обробку та інтерпретацію для подальшої передачі на обчислювальні пристрої.

Машинний зір виконує аналіз візуальної інформації з метою подальшого прийняття керувальних рішень.

Результати обробки можуть бути збережені або передані виконавчим пристроям, таким як роботи або маніпулятори, відповідно до поставлених завдань і функціонального призначення системи.

У порівнянні з іншими сенсорними технологіями, системи машинного зору здатні забезпечити до 90 % візуальної інформації, необхідної для обчислювального пристрою, з метою забезпечення коректного функціонування системи та формування керувальних впливів на підключені виконавчі елементи. Процес функціонування включає кілька послідовних етапів:

– отримання інформації;

– оброблення інформації;

– сегментація та опис;

– розпізнавання та інтерпретація.

Автоматизація виробничих процесів стала ключовою передумовою для розвитку промисловості з метою забезпечення високої точності, ефективності та продуктивності.

Центральним елементом цієї трансформації є технологія комп’ютерного зору, яка виконує фізичний контроль із високим рівнем точності (рисунок 1.1). Застосування комп’ютерного зору модернізує процедури контролю якості, адже завдяки автоматизованим інтелектуальним системам досягається вища ефективність порівняно з традиційними методами ручного інспектування.

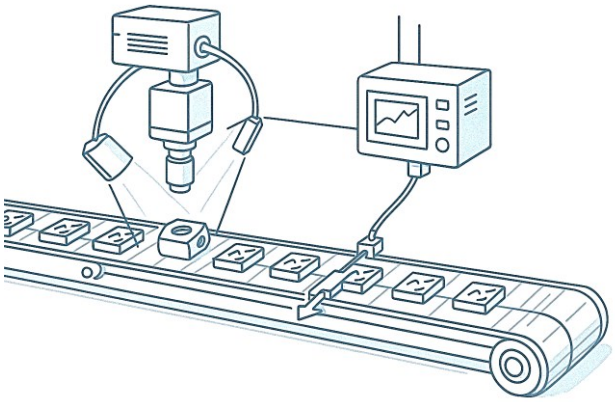

Рисунок 1.1 – Ілюстрація комп’ютерного зору для автоматизованих перевірок складальних ліній

Комп’ютерний зір, як одна з галузей штучного інтелекту, надає можливість комп’ютерам і програмним системам навчатися за зображеннями та відео для подальшого прийняття рішень [3].

У контексті складальних виробничих ліній системи комп’ютерного зору включають в себе камери, сенсори та алгоритми, які забезпечують моніторинг, аналіз і контроль якості продукції в режимі реального часу [4].

Завдяки впровадженню комп’ютерного зору виробники можуть адаптувати власні перевірочні процеси та концентрувати увагу на нещодавно виявлених дефектах виробів (рисунок 1.2).

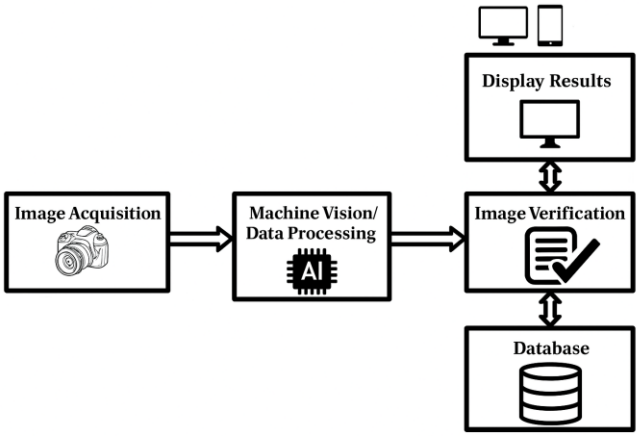


Рисунок 1.2 – Блок-схема системи контролю з використанням комп’ютерного зору на складальній лінії

Упродовж останніх двох років системи комп’ютерного зору зазнали значного прогресу завдяки впровадженню новітніх підходів глибокого навчання.

Згорткові нейронні мережі (ЗНС) та моделі, засновані на трансформерах, зокрема Vision Transformers (ViT), нині широко застосовуються для високоточного виявлення та класифікації виробничих дефектів.

Високий рівень точності: архітектури глибокого навчання дозволяють виявляти дефекти на мікрорівні, що суттєво підвищує ефективність систем контролю якості.

Моделі з можливістю самонавчання: сучасні алгоритми можуть бути адаптовані з використанням обмежених обсягів даних, включно з новими прикладами варіацій продукції або раніше невідомими дефектами.

Використання периферійного штучного інтелекту (Edge AI) забезпечує розміщення обчислювальних потужностей поблизу виробничого обладнання безпосередньо на об’єкті.

Низький рівень затримки: прийняття рішень здійснюється з мінімальною затримкою, без необхідності передачі даних у хмарні сервіси.

Збереження конфіденційності: окремі категорії виробничих даних обробляються локально, що сприяє підвищенню рівня їхнього захисту та мінімізації ризиків витоку інформації.

Сучасні моделі промислових камер характеризуються підвищеною роздільною здатністю, що забезпечує детальне зображення об’єктів та створює умови для поглибленого аналізу за необхідності.

Такі камери є особливо цінними для високотехнологічних галузей, зокрема електронної промисловості, де дефекти можуть мати мінімальні розміри й бути малопомітними.

Використання мультиспектральних і гіперспектральних методів розширює межі візуального контролю за межі видимого спектру, надаючи можливість виявлення внутрішніх або хімічних дефектів у матеріалах, які не можуть бути ідентифіковані неозброєним оком (рисунок 1.3).

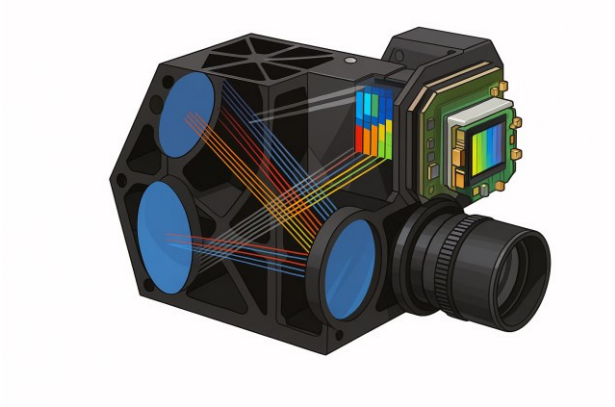

Рисунок 1.3 – Схема гіперспектральної камери зображення

У системах тривимірного комп’ютерного зору для побудови високоточних просторових моделей об'єктів застосовуються стереоскопічні камери та технологія LiDAR [5].

Ці засоби дають змогу ефективно ідентифікувати характерні ознаки структурних пошкоджень або помилок під час складання виробів (рисунок 1.4).

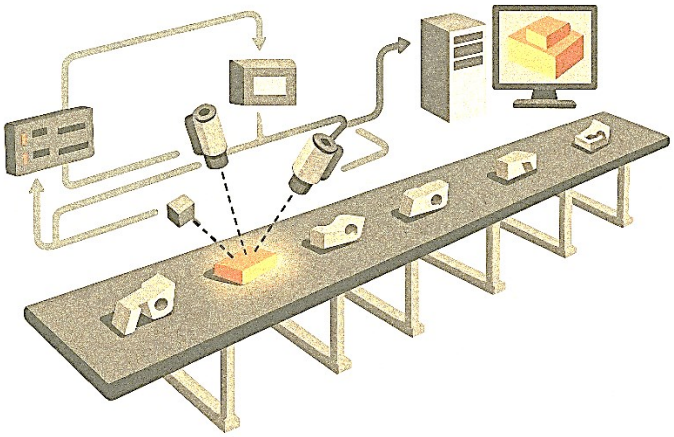

Рисунок 1.4 – Ілюстрація налаштування стереоскопічної камери для системи 3D-зору

Пояснюваний штучний інтелект надає можливість користувачам отримати уявлення про логіку прийняття рішень, яку використовує модель штучного інтелекту при аналізі зображень у системах комп’ютерного зору.

Застосування технологій комп’ютерного зору для задач автоматизованого контролю є ефективним інструментом промислової інспекції.

Виявлення дефектів: комп’ютерний зір демонструє високу результативність у виявленні таких дефектів, як стирання, вм’ятини, зсуви або відсутність окремих елементів конструкції, у режимі реального часу [6].

У сфері виробництва електроніки ця технологія застосовується для виявлення дефектів паяних з’єднань, а також мікротріщин на друкованих платах.

В автомобільній галузі системи комп’ютерного зору використовуються для виявлення дефектів лакофарбового покриття, пошкоджень елементів кузова, а також порушень у встановленні панелей чи вузлів (рисунок 1.5).

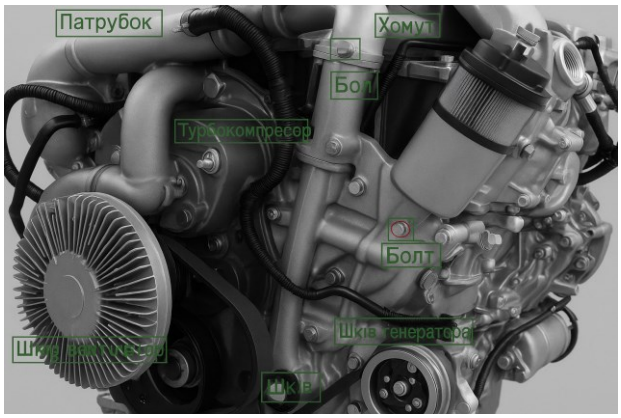


Рисунок 1.5 – Вихідне зображення системи машинного зору, яка виявляє відсутні деталі в автомобільній промисловості

Коректне з’єднання всіх елементів конструкції та дотримання послідовності їх монтажу є обов’язковими умовами для забезпечення надійної роботи обладнання в критичних галузях, таких як аерокосмічна промисловість або медична техніка.

У аерокосмічному секторі системи комп’ютерного зору дозволяють перевірити, чи всі складні вузли правильно встановлені, що є передумовою відповідності пристрою встановленим стандартам безпеки.

У сфері медичних виробів здійснюється контроль відповідності технічних параметрів високоякісним вимогам – зокрема, для таких пристроїв, як кардіостимулятори або шприци.

Системи комп'ютерного зору здійснюють автоматизовану перевірку геометричних параметрів деталей з високою точністю відповідно до проектної документації.

У точному машинобудуванні це дозволяє вимірювати, наприклад, різницю в габаритах між окремими компонентами двигуна або забезпечувати точність інструментів, що застосовуються у хірургічних втручаннях.

Контроль якості поверхні має важливе значення у виробництві побутової електроніки та упаковки, де естетичні та функціональні дефекти неприпустимі.

У виробництві смартфонів комп'ютерний зір використовується для забезпечення бездоганного вигляду екранів та задніх кришок.
У галузі харчового пакування системи виявляють дефекти, такі як неповне запечатування продукції або пошкодження етикеток.

Інтеграція комп'ютерного зору з сенсорними технологіями дає змогу здійснювати моніторинг стану зношуваних компонентів та формувати прогнози щодо їх технічного обслуговування.

У виробничих автоматизованих системах це дозволяє виявляти, наприклад, ранні ознаки зносу конвеєрної стрічки або відхилення положення роботизованих маніпуляторів від заданих координат (рисунок 1.6).

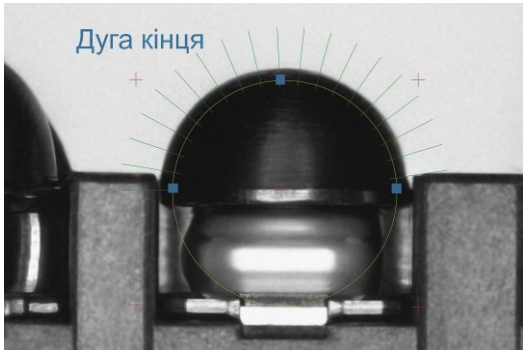


Рисунок 1.6 – Система комп'ютерного зору з виявленням виробничого дефект у кульковому підшипнику

Системи комп’ютерного зору в автомобільній промисловості забезпечують перевірку читабельності та коректності штрих-кодів, QR-кодів і маркувань продукції в режимі реального часу. У сфері роздрібної торгівлі такі рішення допомагають запобігти втратам товарів або помилковому обліку їх кількості на складах і в торгових точках.

Високотехнологічні системи комп’ютерного зору дозволяють здійснювати моніторинг взаємодії працівників з обладнанням з метою контролю дотримання вимог техніки безпеки. У промисловій робототехніці такі системи виявляють присутність персоналу в зонах підвищеної небезпеки поруч із автоматизованими установками, що сприяє запобіганню травматизму.

На відміну від людини, яка схильна до втоми та суб’єктивних помилок, системи комп’ютерного зору характеризуються високою точністю й здатністю виявляти навіть найменші дефекти у продукції. Автоматизовані комплекси контролю адаптовані до роботи на високошвидкісних складальних лініях, забезпечуючи своєчасне виявлення і корекцію помилок у режимі реального часу.

Раннє виявлення виробничих дефектів дозволяє зменшити кількість браку та повернень, усунути необхідність у повторному виробництві, а також скоротити витрати на ручний контроль. Масштабовані системи комп’ютерного зору здатні оперативно навчатися на нових типах продукції, що спрощує їх інтеграцію в галузі з широким асортиментом і частою зміною виробничих ліній.

Комп’ютерний зір дозволяє збирати та аналізувати статистичні дані про типові дефекти, що створює передумови для постійного вдосконалення технологічних процесів і підвищення загального рівня якості продукції.

Хоча комп’ютерний зір має значний трансформаційний потенціал, його впровадження в системи автоматизованого контролю на складальних лініях супроводжується низкою викликів. Однією з ключових проблем є вимоги до даних, оскільки ефективне навчання моделей глибокого навчання потребує великого обсягу маркованих даних, які не завжди доступні. Додаткові труднощі спричиняють умови навколишнього середовища на виробництві, зокрема

коливання освітлення, наявність пилу чи вібрацій, що негативно впливають на стабільність і точність роботи систем.

Ще одним суттєвим бар’єром є складність інтеграції – поєднання систем комп’ютерного зору з іншими виробничими платформами вимагає значних часових та інженерних витрат. Також існує потреба у кваліфікованих кадрах, зокрема інженерах зі штучного інтелекту та фахівцях з обробки зображень, що необхідні для розгортання, налаштування та обслуговування таких систем. Окрім того, високий рівень початкових інвестицій, пов'язаний із придбанням камер, сенсорів, обчислювального обладнання та програмного забезпечення для аналізу, може стримувати впровадження, особливо на підприємствах малого та середнього бізнесу.

Водночас існують шляхи подолання цих викликів. Зокрема, використання відкритих програмних фреймворків, таких як OpenCV, а також розвиток периферійних обчислювальних пристроїв на базі штучного інтелекту сприяють зменшенню вартості систем і роблять їх доступними для невеликих виробничих підприємств. Розширені підходи до навчання моделей, включно з генерацією синтетичних даних і трансферним навчанням, дозволяють частково вирішити проблему обмеженості реальних навчальних вибірок. Завдяки вдосконаленню алгоритмів покращення зображення та створенню надійних механічних рішень системи комп’ютерного зору можуть функціонувати навіть у складних виробничих умовах. Крім того, широке впровадження рішень типу «підключи та працюй», що включають шаблонізовані програмно-апаратні компоненти, значно спрощує процес впровадження та скорочує його тривалість [7].

### 1.2 Сучасні промислові датчики та системи технічного зору

Сучасні технології машинного зору відіграють важливу роль у сфері промислової автоматизації, забезпечуючи контроль якості, точне позиціонування та аналітичний моніторинг процесів. Системи технічного зору поділяються на

двомірні та тривимірні залежно від методів побудови візуального представлення об’єктів. Двомірні датчики забезпечують аналіз форми, розмірів, положення й наявності об’єктів, тоді як тривимірні дозволяють будувати глибокі просторові моделі та працювати в неструктурованому середовищі в реальному часі [8].

Двомірні сенсори, такі як O2D (датчик контуру), забезпечують розпізнавання та сортування об’єктів на основі заздалегідь заданих моделей. Завдяки використанню еталонів "придатних" і "непридатних" зразків, ці пристрої здатні ефективно визначати положення та орієнтацію деталей. Лічильник пікселів O2V, на відміну від контурних сенсорів, здійснює оцінку об’єктів за такими параметрами, як площа, округлість і щільність, що дозволяє застосовувати його в умовах, де контурний аналіз є ускладненим – наприклад, при блискучих або віддзеркалюваних поверхнях.

Серія VG пропонує вдосконалені сенсори з дев’ятьма основними функціями контролю, серед яких: контраст, край, яскравість, кут, довжина, діаметр та підрахунок об’єктів. Ці системи використовуються в різних промислових секторах – від харчової до напівпровідникової галузі. Камери серії VOS мають модульну архітектуру та забезпечують попередню обробку даних безпосередньо на пристрої. Завдяки універсальному інтерфейсу користувача та широким налаштуванням вони підтримують адаптацію під конкретні виробничі умови, що дозволяє ефективно вирішувати задачі позиціонування, навігації, OCR та оптичного вимірювання.

Тривимірні датчики, такі як KUKA_3D Perception, забезпечують точне позиціонування об’єктів у просторі з точністю до міліметра. Вбудована обчислювальна платформа дозволяє сенсору самостійно обробляти глибину зображення та візуальні дані, що робить систему надзвичайно ефективною для мобільної робототехніки. Завдяки своїй точності та здатності працювати при недостатньому освітленні, ці сенсори ідеально підходять для складних динамічних середовищ.

Системи технічного зору використовуються для визначення просторових координат об’єктів (x, y, z), а також кутових орієнтацій (θ, ω, φ). Прості конфігурації дозволяють контролювати лише площинне положення та орієнтацію об’єктів, що є достатнім для більшості прикладних задач. Більш складні комплекси забезпечують повноцінну шестивимірну локалізацію, однак їх застосування виправдане лише у високоточних технологічних процесах. Перевагою таких систем є можливість гнучкої переконфігурації робочого простору без змін у конструкції – достатньо лише оновити програмну логіку.

Однією з найрозвиненіших систем у цій галузі є FANUC iRVision, що забезпечує повну інтеграцію з контролером R-30iB. Система підтримує розпізнавання дво- та тривимірних об’єктів, сортування за кольором, зчитування штрих-кодів, гнучку подачу деталей, а також високошвидкісне лінійне візуальне стеження. Завдяки інтеграції системи безпосередньо в керуючий модуль, усувається необхідність у зовнішньому обладнанні, що дозволяє зменшити витрати та підвищити адаптивність у змінних виробничих умовах.

Таким чином, сучасні промислові датчики технічного зору забезпечують не лише візуальний контроль, а й функції інтелектуального аналізу та прийняття рішень. Вони виступають ключовим елементом в автоматизованих системах, сприяючи підвищенню ефективності, гнучкості та точності у виробничих і логістичних процесах.

# РОЗДІЛ 2
# РОЗРОБЛЕННЯ СТРУКТУРНОЇ СХЕМИ ТА АЛГОРИТМУ РОБОТИ АВТОМАТИЗОВАНОЇ СИСТЕМИ

## 2.1 Розроблення структурної схеми автоматизованої системи формування бібліотеки готової продукції підприємства

У сучасних проектах впроваджуються апаратні й програмні технології, які пройшли апробацію та стандартизацію в галузі промислових застосувань, паралельно з розвитком і вдосконаленням уже наявних засобів автоматизації. Контролери, інтегровані в технічні засоби автоматизації, зазвичай експлуатуються в умовах підвищеної складності, а відмова пристрою в таких системах може спричинити значні втрати, оскільки об’єкт керування працює безперервно та в режимі реального часу. Цифрові системи автоматичного управління, як правило, містять велику кількість сенсорів і перетворювачів, що забезпечують вимірювання фізичних параметрів, таких як температура, витрата рідини, тиск, швидкість та інші. Сенсори здійснюють перетворення фізичних величин у стандартні електричні сигнали, зазвичай у вигляді напруги або струму. При використанні комп’ютеризованих систем виникає потреба у трансформації аналогового сигналу у цифрову форму, що зумовило швидкий розвиток сенсорів нового покоління з вбудованими контролерами, які забезпечують цю функцію. Такий інтелектуальний сенсор виступає як складова обчислювальної системи та підтримує цифрові протоколи передачі даних. Поширене використання мають сенсори, у яких контролер виконує попередню цифрову обробку сигналу, зокрема фільтрацію шумів, корекцію систематичних похибок і моніторинг працездатності. Спостерігається тенденція до впровадження цифрових технічних засобів керування, у яких перетворення фізичних параметрів відбувається безпосередньо у цифрову форму, придатну для подальшої передачі по цифрових каналах зв’язку [9]. Усі сучасні системи технічного зору в процесі функціонування виконують послідовні операції, що

включають зчитування інформації з об’єкта, перетворення оптичного сигналу в електричний, оцифрування та подальшу обробку – дискретизацію, квантування, стиснення, аналіз за допомогою мікропроцесора та формування відповідного рішення. Таким чином, структурну схему розроблюваної системи автоматизації можна представити як комбінацію блоків: блоку захоплення та попередньої обробки зображення (оцифрування, стиснення) та блоку керування, основними функціями якого є збереження й аналіз інформації (рисунок 2.1).

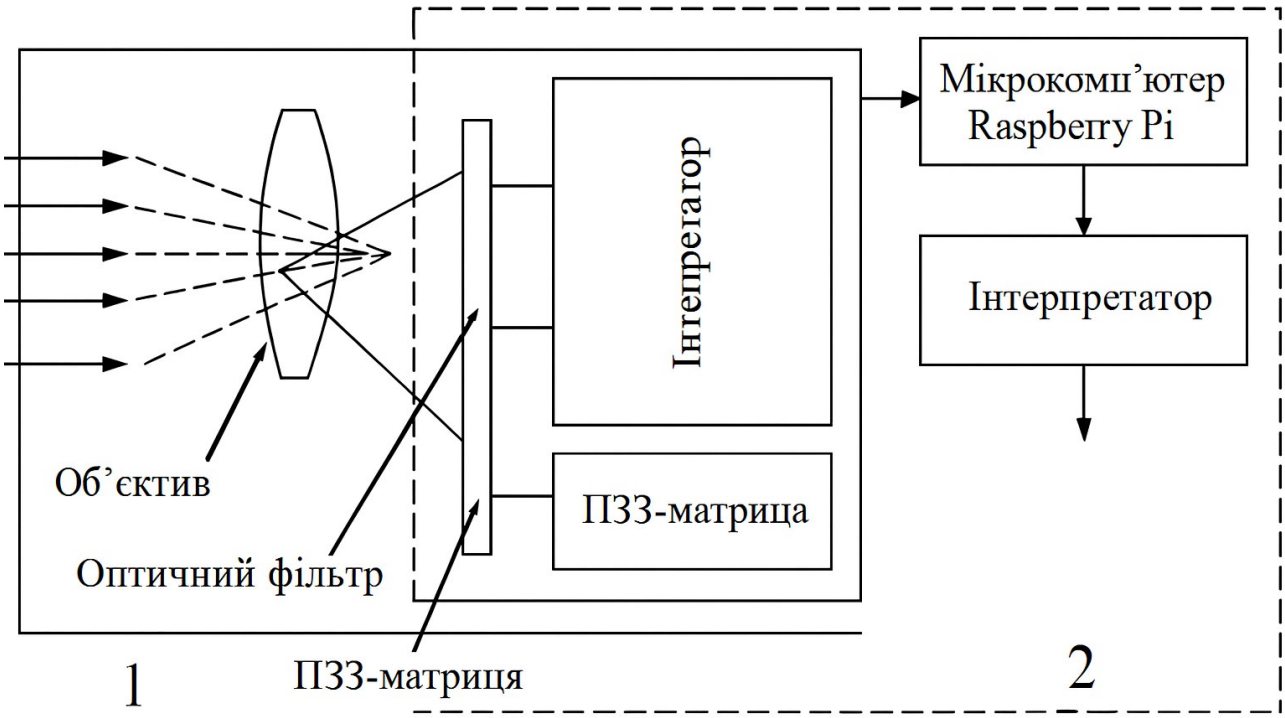


Рисунок 2.1 – Структурна схема автоматизованої системи

Блок отримання та попереднього оброблення зображення складається з об'єктива, оптичного фільтра, ПЗЗ-матриці та мікросхем, які забезпечують оцифрування вхідного сигналу, його дискретизацію, квантування та стиснення.

Сучасні промислові камери за принципом перетворення оптичного сигналу в електричний поділяються здебільшого на ПЗЗ- та КМОН-типи. Основною відмінністю є потреба ПЗЗ-камер у додаткових електронних схемах

для реалізації процесу оцифрування сигналу, що зумовлює збільшення габаритів пристроїв, побудованих на основі ПЗЗ-матриць.

Такі пристрої також потребують використання спеціалізованих кодерів, які здійснюють стискання цифрового відеосигналу до стандартних форматів зображення, таких як JPEG або PNG.

Камери з КМОН-архітектурою були розроблені як удосконалена альтернатива ПЗЗ-камер і відзначаються наявністю окремого аналого-цифрового перетворювача в кожному рядку сенсорної матриці, що реалізується на одному кремнієвому кристалі.

Завдяки простоті виготовлення та мініатюризації елементної бази КМОН-камери отримали широке поширення у різних сферах застосування.

Блок управління включає мікрокомп’ютер, який виконує обробку відеосигналу згідно з алгоритмічними приписами, а також монітор, що використовується для переналаштування системи та візуального контролю оператором результатів її роботи [10].

## 2.2 Вибір компонентів системи для реалізації формування бібліотеки зразків готової продукції

У розроблюваній системі автоматизації доцільним є використання мікрокомп’ютера Raspberry Pi, що обумовлено його низькою вартістю порівняно з персональними комп’ютерами та достатньо високою обчислювальною потужністю.

Головною відмінністю від звичайного ПК є наявність на платі багатофункціональних портів GPIO (General-purpose input/output), які розширюють можливості апаратної взаємодії [11].

Користувач має змогу самостійно керувати підключеннями до портів GPIO, що дає можливість підключати додаткові пристрої, такі як дисплеї, кнопки, реле, сенсори та інші електронні модулі на власний розсуд.

Попри візуальну схожість із платформою Arduino, Raspberry Pi принципово відрізняється за архітектурою функціонування.

Цей мікрокомп’ютер працює під керуванням однієї з адаптованих для нього спеціалізованих операційних систем, що забезпечує гнучкість у налаштуванні та використанні [12].

Вибір операційної системи залежить як від сфери застосування, так і від особистих уподобань користувача; у межах нашої системи використовується дистрибутив Ubuntu, зокрема його модифікація Ubuntu Mate, яка вважається однією з найбільш оптимальних.

Завдяки низьким системним вимогам Ubuntu Mate є придатною як для сучасних робочих станцій, так і для одноплатних комп’ютерів і застарілого обладнання.

Обрана платформа Raspberry Pi (рисунок 2.2) повністю відповідає технічним потребам, демонструє високі показники продуктивності та містить низку важливих функцій, серед яких вбудована підтримка мови програмування Python, підключення двох дисплеїв з роздільною здатністю 4K, порт Ethernet 1Gbps та два порти USB 3.0.

Також мікрокомп’ютер підтримує апаратне декодування H.265/HEVC (до 4Kp60) та H.264 (до 1080p60), оснащений дводіапазонним Wi-Fi модулем стандарту IEEE 802.11ac і Bluetooth версії 5.0 BLE.

На сьогодні Raspberry Pi 4 є одним із найпотужніших одноплатних комп’ютерів на ринку завдяки 4-ядерному 64-бітному процесору SoC Broadcom BCM2711 з архітектурою Cortex-A72 і тактовою частотою 1,5 ГГц [13].

Можливе конфігурування за обсягом оперативної пам’яті: доступні варіанти з 1GB, 2GB, 4GB і 8GB, при цьому для реалізації нашого проєкту достатньо версії з 4GB.

Схема плати Raspberry Pi 4 представлена нижче на рисунку 2.2.

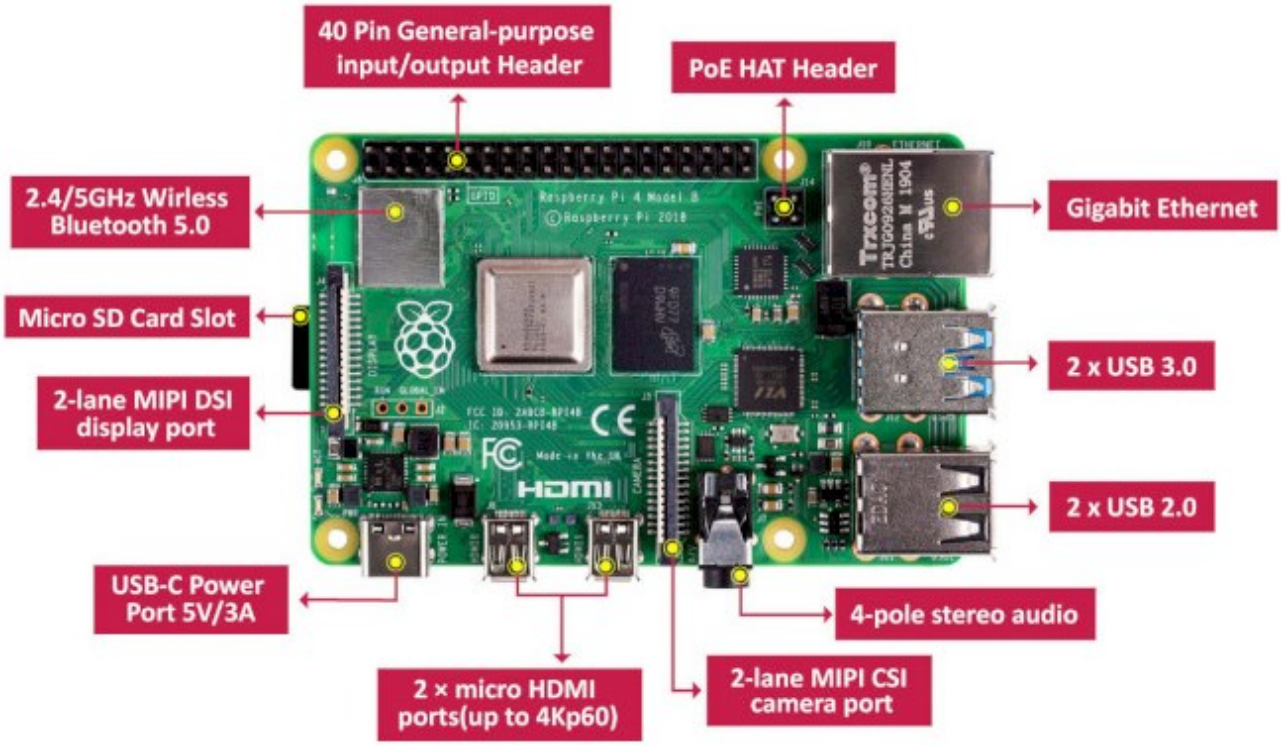


Рисунок 2.2 – Схема плати Raspberry Pi 4 [13]

Плата Raspberry Pi 4 оснащена такими основними апаратними компонентами [14]:

– контакти загального призначення (GPIO);

– Ethernet-порт для проводового мережевого з’єднання;

– два порти USB 3.0 і два порти USB 2.0 для підключення периферійних пристроїв;

– порт для підключення камери;

– два порти HDMI для зовнішніх дисплеїв;

– USB-порт для подачі живлення;

– спеціалізований відеовихід;

– слот для карти пам’яті microSD, розташований на зворотному боці плати.

Таким чином, наявність сучасної та функціонально насиченої платформи дає змогу реалізовувати широкий спектр інженерних задач.

У якості відеокамери для реалізації проєкту було обрано модель OV5647, яка є сумісною з Raspberry Pi та демонструє відповідний рівень якості.

Вона забезпечує зйомку з частотою до 60 кадрів за секунду при роздільній здатності 720p, що цілком відповідає вимогам до розроблюваної системи.

Зображення обраної відеокамери наведено на рисунку 2.3.

Рисунок 2.3 – Камера OV5647

У залежності від умов експлуатації та місця розміщення системи, відеокамера може бути замінена на більш функціональну модель.

## 2.3 Алгоритм роботи автоматизованої системи формування бібліотеки зразків готової продукції підприємства

Передбачається використання стаціонарної базової платформи, яка здійснюватиме зйомку та обробку зображень у режимі реального часу. У зв’язку з цим нами було змодельовано орієнтовну зону розміщення відповідного компонента системи, що наведена на рисунку 2.4.

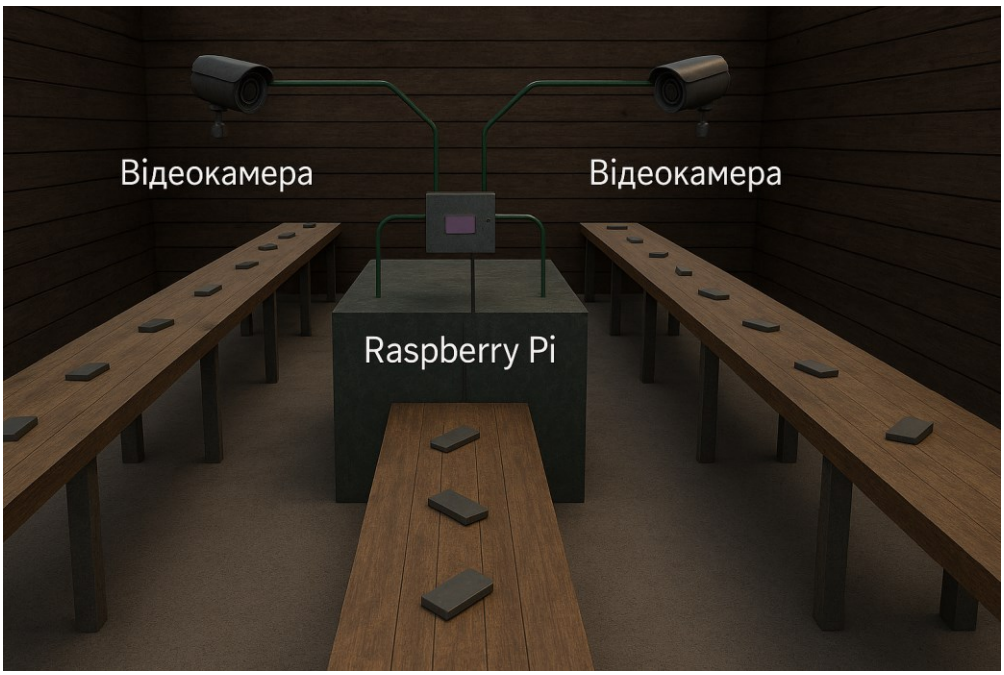


Рисунок 2.4 – Загальний вид зони застосування системи

Запропоновано розміщення системи у виробничому приміщенні з конвеєрною стрічкою. Передбачається встановлення відеокамер (сенсорів) збоку та над конвеєром, з використанням однотонного (світлого) фону позаду в зоні горизонтальної зйомки, а також додаткового освітлення для забезпечення чіткості зображення.

У наведеному прикладі мають місце обмеження, пов’язані з використанням двох конвеєрних ліній і чотирьох відеокамер на одну плату Raspberry Pi, що зумовлено наявністю лише чотирьох USB-портів на самій платі.

Робота системи базується на алгоритмі, що включає отримання цифрового сигналу від камери, передачу його на обробку до мікрокомп’ютера з можливістю візуалізації результатів.

На рисунку 2.5 наведено схему розміщення основних компонентів системи, а розріз конструкції представлено на рисунку 2.6.

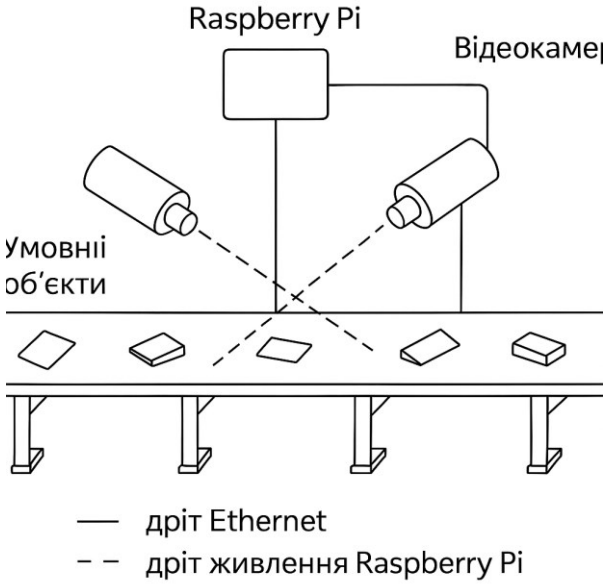


Рисунок 2.5 – Розміщення системи на конвеєрній лінії

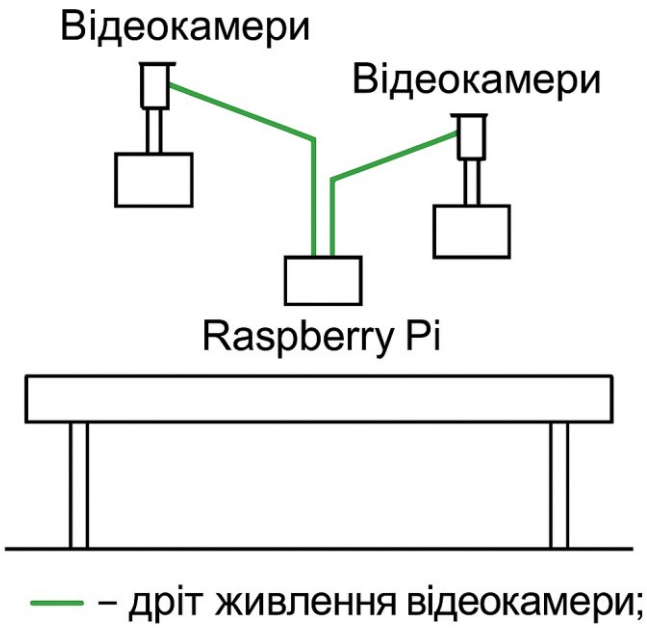


Рисунок 2.6 – Розміщення системи на конвеєрній лінії у розрізі

На першому етапі програмне забезпечення отримує зображення від відеокамери через USB-порт.

Оскільки обробка необробленого зображення є складною, передусім необхідно виконати попередню обробку з метою виокремлення об’єкта, що нас цікавить.

Для цього застосовується набір фільтрів, огляд яких буде представлено в підпункті 3.2, а також використовується оператор Собеля.

На другому етапі з отриманого зображення необхідно виділити інформацію, що може бути інтерпретована в числовому вигляді.

Основними вихідними даними є координати кутових точок об’єкта.

На третьому етапі, використовуючи координати крайніх точок, за допомогою програмних бібліотек здійснюється обчислення габаритних параметрів об’єкта: довжини, ширини, периметра та площі.

Порівнюючи ці геометричні параметри з заданими еталонами, можна визначити, чи об’єкт становить інтерес для подальшої обробки, чи його слід пропустити.

Заключним етапом є перевірка відповідності об’єкта визначеним критеріям, зокрема, за такими параметрами, як кількість граней.

Якщо об’єкт відповідає вимогам, він підлягає вимірюванню, класифікації та фіксації у базі даних, після чого система переходить до аналізу наступного зображення [15].

Алгоритм функціонування системи наведено на рисунку 2.6.

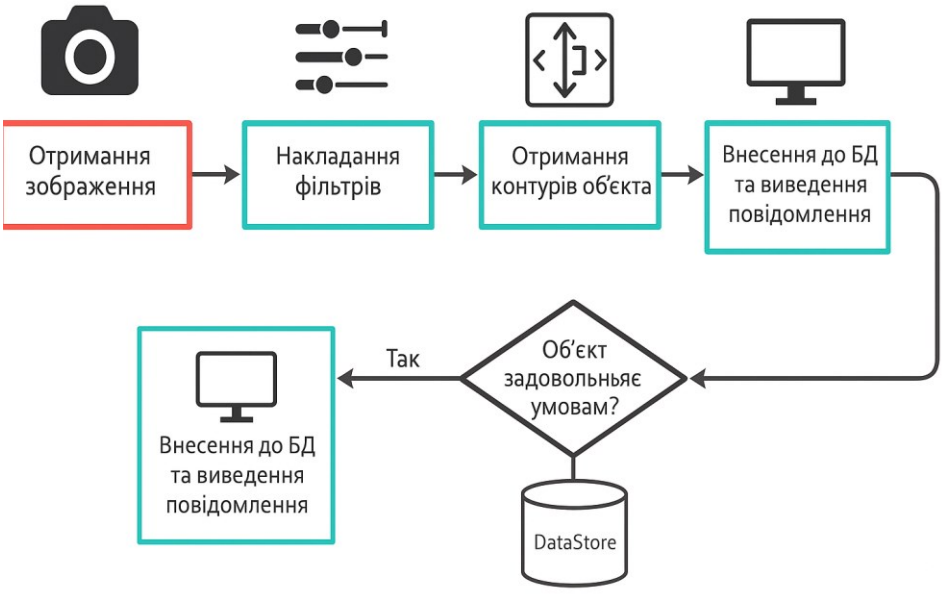


Рисунок 2.6 – Алгоритм формування бібліотеки зразків продукції

## 2.4 Вибір інструментів для реалізації технології технічного зору

Одним із найефективніших інструментів для реалізації систем комп'ютерного зору є бібліотека OpenCV.

OpenCV являє собою бібліотеку алгоритмів з відкритим програмним кодом, призначену для задач комп'ютерного зору, обробки зображень та загальних чисельних обчислень.

Вона реалізована мовами програмування C/C++, однак також доступна версія для Python, що є одним із ключових критеріїв її вибору.

Розробка бібліотеки була започаткована у 1998 році компанією Intel, надалі продовжена в Itseez за активної участі міжнародної спільноти.

OpenCV поширюється на умовах відкритої ліцензії, що дозволяє її безкоштовне використання як у відкритих проєктах, так і в комерційних розробках із закритим кодом.

Інтеграція бібліотеки у програму не обов'язково передбачає використання повного набору функціоналу – можливо вибіркове застосування окремих фрагментів коду.

Єдиною вимогою ліцензії є наявність копії ліцензії OpenCV у супровідній документації до програмного продукту.

Завдяки ліберальним умовам ліцензування бібліотека активно використовується провідними компаніями, навчальними закладами та дослідницькими установами, зокрема: NVidia, Intel, Willow Garage, Google; часткове фінансування її розробки здійснюється компаніями NVidia та Willow Garage [16].

Багаторівнева архітектура бібліотеки представлена на рисунку 2.7.

OpenCV включає 16 функціональних модулів та реалізує понад 1000 алгоритмів обробки зображень.

Платформа підтримує основні операційні системи, такі як Linux, Windows, MacOS, Android та iOS, а також дозволяє інтегрувати сторонні

бібліотеки, наприклад, для роботи з пристроями типу Kinect або реалізації паралельних обчислень (GPU).

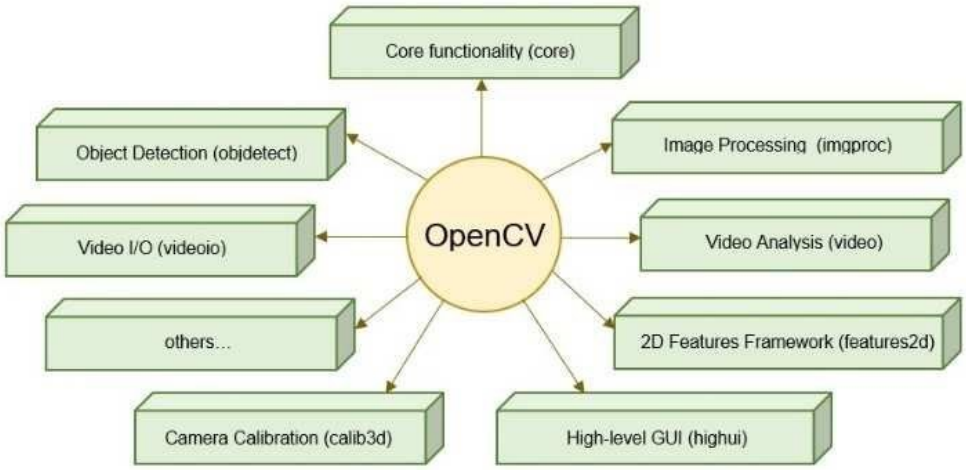


Рисунок 2.7 – Структура OpenCV

Основні модулі бібліотеки OpenCV умовно поділяються на чотири функціональні групи:

– модулі Core та HighGUI, що забезпечують базову функціональність бібліотеки, включаючи реалізацію основних структур даних, математичних операцій, генерацію випадкових чисел, функції лінійної алгебри, швидке перетворення Фур'є, а також введення та виведення зображень і відео, зокрема у форматах XML, YAML та інших;

– модулі ImgProc і Features2D, призначені для обробки зображень, зокрема для фільтрації, геометричних трансформацій, перетворення колірних просторів, сегментації, виявлення ключових точок, ребер і контурного аналізу;

– модулі Video, ObjDetect, Calib3D, що забезпечують калібрування камер, аналіз руху, відстеження об'єктів, просторове позиціонування, побудову карт глибини та виявлення об'єктів за допомогою методів оптичного потоку;

– модуль ML, який реалізує алгоритми машинного навчання, серед яких: метод k-ближчих сусідів, наївний байєсівський класифікатор, дерева рішень,

бустинг, градієнтний бустинг дерев рішень, випадкові ліси, машина опорних векторів, нейронні мережі та інші.

Після попередньої обробки виконується виявлення об’єктів, що становлять інтерес, виділення їх характерних частин та сегментація зображення за допомогою модулів ImgProc і ObjDetect. У випадку фіксованої камери та рухомих об’єктів доцільним є використання алгоритмів віднімання фону. Після цього вирішується ключове завдання: визначення просторового положення об’єкта, реконструкція за структурою, аналіз структури або реєстрація об’єкта в сцені, що реалізується за допомогою модулів Calib3D, Contrib, Video, Stitching, Videostab і ML.

Наприклад, у задачах побудови панорам виконується зіставлення фрагментів зображень із різних кадрів і визначення необхідного геометричного перетворення. У системах відеоспостереження може бути реалізовано відновлення траєкторій руху об’єктів. На завершальному етапі виконується класифікація та прийняття рішень з використанням модулю ML. Наприклад, у системах відеонагляду визначається наявність або відсутність об’єкта, що викликає підозру; у задачах розпізнавання тексту – здійснюється ідентифікація змісту знайденого текстового фрагмента. Таким чином, бібліотека OpenCV є оптимальним рішенням для реалізації поставленого завдання [18].

# РОЗДІЛ 3
# РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ

## 3.1 Вибір програмного середовища

У сучасних умовах, коли обчислювальні ресурси персональних комп'ютерів і серверів досягли високого рівня, значно зросла популярність інтерпретованих мов програмування. Це пов'язано з тим, що для запуску програм, написаних такими мовами, окрім самого коду, необхідно також ініціювати інтерпретатор, що потребує додаткових ресурсів. Прикладом такої мови є Python.

Згідно зі статистичними даними останніх п'яти років, Python стабільно входить до п'ятірки найбільш затребуваних мов програмування, а за результатами опитувань DOU у першому кварталі 2021 року вона посіла четверте місце в Україні, поступившись лише JavaScript, Java та C#. Така позиція свідчить про високу популярність і актуальність цієї мови на сучасному ринку програмних розробок.

Популярність Python пояснюється його універсальністю, оскільки мова здатна ефективно вирішувати широке коло завдань і є придатною для всіх основних операційних систем. Зокрема, за останні роки Python активно використовується в галузях машинного навчання та аналізу даних завдяки своїй гнучкості, масштабованості та розширюваності, що має ключове значення в цих сферах. Окрім цього, Python широко застосовується у веб-розробці, де залишається актуальним із початку 2000-х років [19].

До переваг мови Python належать:

– гнучкість, що є однією з головних причин її популярності серед розробників;

– розширюваність, яка забезпечується за рахунок великої кількості бібліотек і модулів, а також можливості інтеграції з кодом, написаним іншими мовами, зокрема мовою С;

– простота синтаксису, що сприяє зрозумілості та читабельності коду;

– інтерпретованість, що дозволяє запускати програми на більшості сучасних платформ без додаткової компіляції, оскільки інтерпретатор зазвичай попередньо встановлено в дистрибутивах Linux;

– наявність стандарту PEP, який уніфікує стиль програмування, роблячи код читабельним і легким для супроводу;

– відкритий вихідний код інтерпретатора, що забезпечує можливість участі у розвитку мови сторонніх розробників, зокрема в реалізації нових функцій у релізах.

Python є високорівневою мовою програмування загального призначення з динамічною строгою типізацією, синтаксисом на основі відступів і автоматичним управлінням пам’яттю, що не створює проблем в умовах сучасної обчислювальної інфраструктури. Мова орієнтована на підвищення продуктивності розробника, забезпечує легке читання та супровід коду, а також гарантує сумісність створених програм із різними платформами. Обґрунтуванням вибору Python як основної мови програмування в межах цього проєкту є її простота, низький поріг входження, наявність великої кількості навчальних матеріалів, а також підтримка бібліотеки OpenCV [20].

Мова Python не потребує спеціального середовища розробки, оскільки скрипти можуть запускатися безпосередньо з терміналу або у вигляді файлів із розширенням «.py». Водночас застосування інтегрованого середовища розробки (IDE) значно полегшує роботу з кодом, особливо при взаємодії з базами даних. У межах цього проєкту найдоцільнішим вибором є середовище PyCharm – кросплатформна IDE, спеціалізована для мови Python. Вона забезпечує інтегровані засоби відладки, автозаповнення коду, графічні інструменти роботи з

базами даних і файлів, що ідеально відповідає завданням, поставленим у межах кваліфікаційної роботи.

## 3.2 Розроблення програмного забезпечення

Для розв’язання задачі промислової автоматизації у сфері безконтактних вимірювань запропоновано використання розмірів зразка-еталона з метою порівняння його параметрів із параметрами об’єктів, що вимірюються.

Однак у процесі створення програмного забезпечення виникла проблема, пов’язана з необхідністю адаптації системи до різних умов експлуатації, що вимагає щоразу змінювати числові константи залежно від відстані між відеокамерою та об’єктами у випадку вертикальної зйомки.

З огляду на вказане, доцільним видається початкове впровадження калібрувального режиму в рамках програмного забезпечення.

Для реалізації цього підзавдання було обрано підхід порівняння отриманих від системи значень з еталонними параметрами, що дозволяє обчислити коефіцієнт відстані для вимірюваних об’єктів.

У ролі еталонного зразка для калібрування використано об’єкт із квадратною формою та довжиною ребра 45 мм, що дає змогу перевірити правильність встановлення камери та виявити можливі перекоси або нахили.

Режим калібрування передбачає відсутність відображення частоти кадрів та наявність відповідних інформативних написів на екрані оператора.

З метою покращення позиціонування тестового зразка, у центрі екрана було додано червону мітку.

Зображення, що виводиться оператору у цьому режимі, наведено на рисунку 3.1.

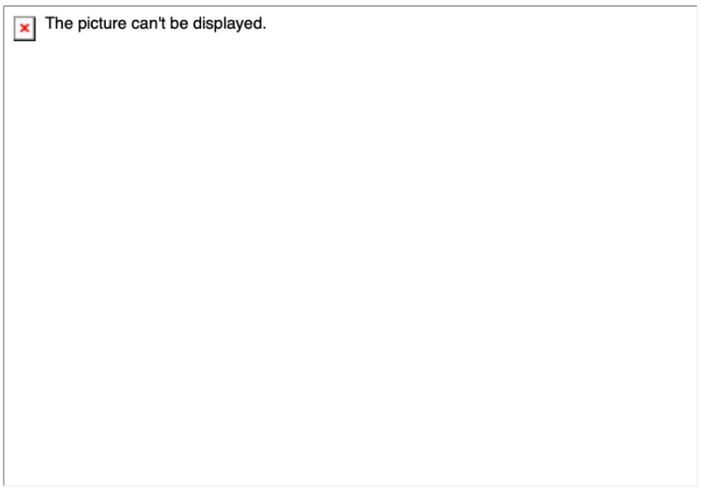


Рисунок 3.1 – Відображення режиму калібрування

У межах реалізації калібрувального режиму було визначено діапазон допустимих значень коефіцієнта відстані в інтервалі [0,5:2].

Це означає, що допустиме відхилення реальної відстані до об’єкта не повинно перевищувати подвоєного або зменшеного вдвічі значення від еталонного.

Для полегшення взаємодії оператора з системою в процесі калібрування до інтерфейсу було додано текстові повідомлення з рекомендаціями щодо правильного позиціонування камери відносно об’єкта.

Візуалізацію повідомлення про надмірно близьке розташування камери наведено на рисунку 3.2.

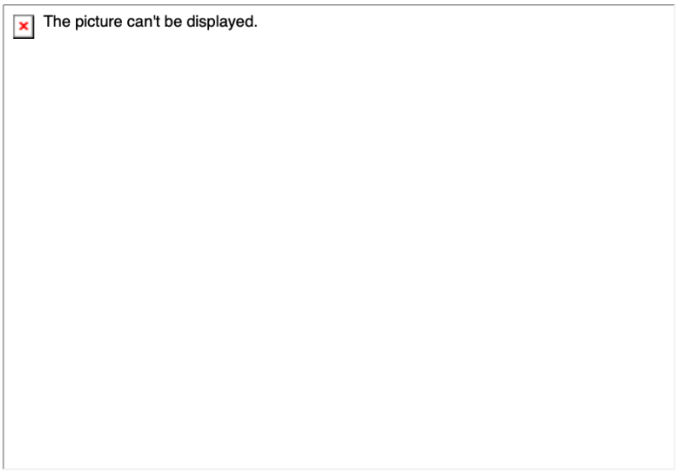


Рисунок 3.2 – Повідомлення про занадто близьке положення

Після розміщення тестового об’єкта у відповідному положенні та його успішного розпізнавання програмним забезпеченням, розраховане значення коефіцієнта віддаленості зберігається в пам’яті системи.

Після цього оператору виводиться повідомлення про успішне завершення процедури вимірювання коефіцієнта.

Після відображення зазначеного повідомлення програма автоматично завершує роботу в режимі калібрування та переходить до основного режиму функціонування.

Візуалізацію завершення калібрувального режиму представлено на рисунку 3.3.

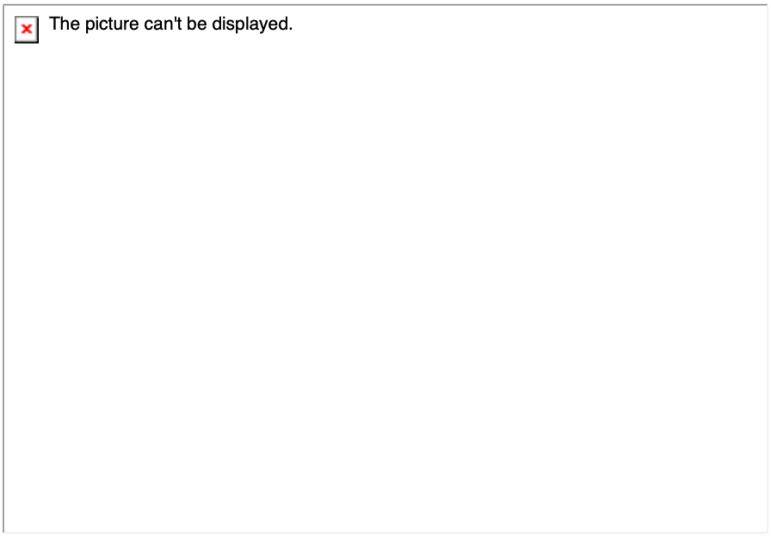


Рисунок 3.3 – Завершення роботи режиму калібрування

Після обчислення необхідного коефіцієнта віддаленості програмне забезпечення автоматично переходить у робочий режим.

У цьому режимі здійснюється безпосередня обробка об’єктів, що надходять у зону огляду, тому доцільно детальніше розглянути приклад обробки прямокутного об’єкта.

На першому етапі до отриманого зображення застосовуються фільтри, починаючи з функції cvtColor з бібліотеки OpenCV.

У рамках цього фільтра використовується параметр COLOR_BGR2GRAY, який здійснює перетворення зображення з кольорового у відтінки сірого, тобто у чорно-білий формат.

Зображення після застосування функції cvtColor наведено на рисунку 3.4.

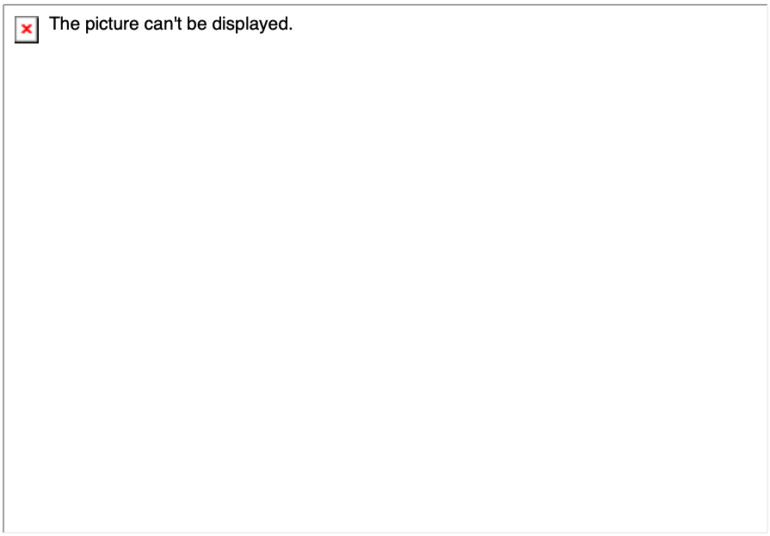


Рисунок 3.4 – Зображення після функції cvtColor

Другою за черговістю, але не менш важливою функцією обробки є функція Canny.

Вона реалізує алгоритм Canny Edge Detection – один із найпоширеніших методів виявлення контурів об’єктів, запропонований Джоном Кенні.

Робота алгоритму складається з декількох етапів:

– придушення шумів;

– обчислення градієнта інтенсивності зображення;

– немаксимальне придушення;

– визначення порогових значень за методом гістерезису.

Розглянемо коротко кожен із цих етапів. Оскільки процедура виявлення країв є надзвичайно чутливою до шумів, на першому етапі здійснюється згладжування зображення з використанням Гаусового фільтра розміром 5×5.

Після згладження отримане зображення обробляється фільтром Собеля як у горизонтальному, так і у вертикальному напрямках для обчислення відповідних похідних – GxG_xGx та GyG_yGy.

На основі цих двох компонент формується градієнт краю та визначається його напрямок для кожного пікселя зображення.

Напрямок градієнта завжди є перпендикулярним до країв об’єкта на зображенні.

Для подальшої обробки цей напрямок округлюється до одного з чотирьох дискретних значень, що відповідають вертикальному, горизонтальному та двом діагональним орієнтаціям.

Після обчислення величини та напрямку градієнта здійснюється сканування зображення з метою усунення нерелевантних пікселів.

Це досягається шляхом перевірки кожного пікселя на предмет того, чи є він локальним максимумом у межах своєї околиці у напрямку градієнта.

Візуалізація цього процесу наведена на рисунку 3.5.

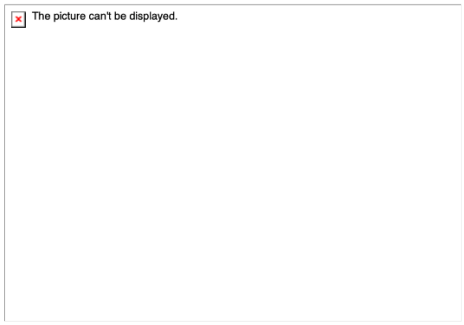


Рисунок 3.5 – Принцип немаксимального придушення

Точка A розташована на краю об’єкта у вертикальному напрямку, при цьому напрямок градієнта є нормальним до цього краю.

Точки B та C розташовані по напрямку градієнта, що дозволяє здійснити порівняння яскравості точки A з її сусідами.

У результаті точка A перевіряється на наявність локального максимуму шляхом порівняння з точками B та C. Якщо A має найбільше значення серед

них, вона зберігається для подальшої обробки, інакше – пригнічується, тобто анулюється.

На наступному етапі – визначення порогових значень гістерезису – вирішується, які контури є значущими, а які слід ігнорувати.

Для реалізації цього підходу необхідно задати два порогових значення: minVal та maxVal.

Усі точки з градієнтом інтенсивності, що перевищує значення maxVal, однозначно класифікуються як частини контуру та розглядаються як істинні ребра.

Пікселі з градієнтом нижчим за minVal вважаються шумом і автоматично відкидаються.

Пікселі з градієнтом, що знаходиться між minVal та maxVal, потребують додаткового аналізу та класифікуються як ребра лише у випадку, якщо вони є зв’язаними з іншими сильними ребрами.

Такий підхід дозволяє ефективно фільтрувати одиничні піксельні шуми, ґрунтуючись на припущенні, що справжні границі об’єктів мають протяжну лінійну структуру.

У результаті обробки формується зображення з чітко окресленими краями, приклад якого представлено на рисунку 3.6.

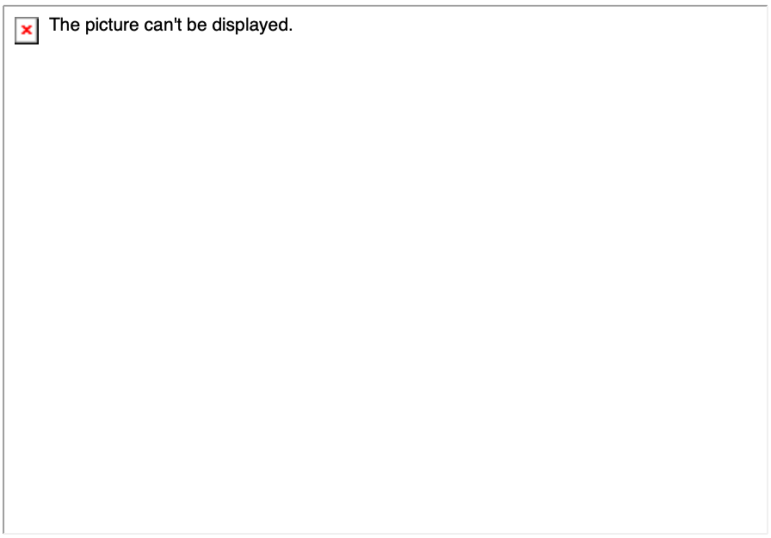


Рисунок 3.6 – Отримане зображення після функції Canny

Наступною функцією, яку буде використано у процесі обробки зображення, є morphologyEx.

Ця функція реалізує низку базових морфологічних операцій, заснованих на геометричних характеристиках зображення.

Зазвичай такі операції виконуються над бінаризованими зображеннями.

Функція приймає два основні аргументи: вихідне зображення та структуруючий елемент (ядро), який визначає форму й масштаб застосовуваної операції.

До двох основних морфологічних трансформацій належать ерозія (Erosion) та розширення (Dilation).

На їх основі реалізуються складніші морфологічні операції, зокрема відкриття, закриття, градієнт тощо.

У нашому випадку використовуються оператори MORPH_CLOSE та MORPH_RECT, які, як видно на рисунку 3.7, забезпечують ефективне заповнення внутрішніх порожнин об’єкта та усунення зайвих артефактів за межами контуру.

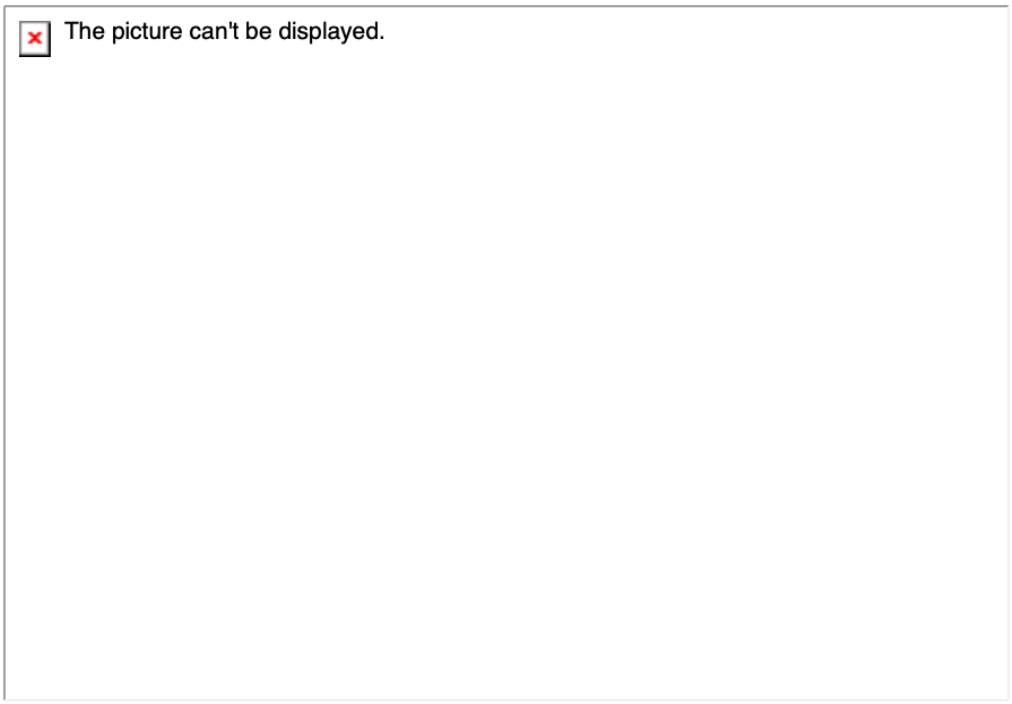


Рисунок 3.7 – Отримане зображення після функції Canny

Отримане після морфологічної обробки зображення використовується як основа для визначення габаритних характеристик об’єкта.

На першому етапі зображення обробляється за допомогою функції findContours для виявлення замкнутих контурів на бінаризованому полі.

Отримані контури передаються до функції grab_contours із бібліотеки imutils для подальшого захоплення та опрацювання.

Після цього здійснюється повернення до бібліотеки OpenCV, де за допомогою функції approxPolyDP виконується апроксимація контурів.

Ця функція дозволяє описати початковий контур наближеним багатокутником з меншою кількістю вершин, при цьому зберігаючи відстань між контуром і апроксимацією не більше за задану величину точності, яка у нашому випадку становить 0,015.

У результаті зазначених операцій система визначає координати кутів об’єкта.

На рисунку 3.8 ці координати позначено рожевим кольором, а лінії, що з’єднують кути – зеленим.

Саме це зображення відображається оператору в режимі візуального контролю.

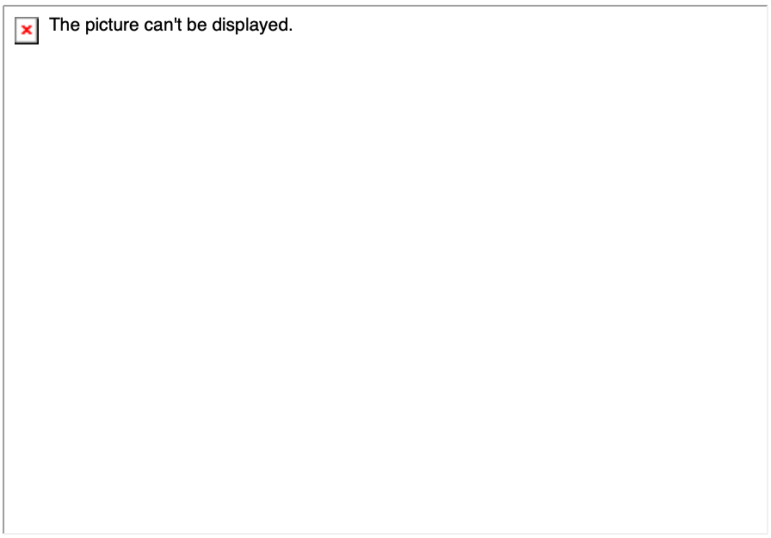


Рисунок 3.8 – Зображення для перегляду оператора

### 3.3 Проведення експерименту

Експериментальні дослідження, які передбачено виконати, ґрунтуються на оцінці якості розпізнавання об’єкта, розміщеного в межах розробленої системи сканування.

Насамперед варто зазначити, що еталонні габарити об’єкта становлять 75 мм по довжині та 45 мм по ширині.

Для оцінювання точності результатів сканування встановлено наступні допуски відносно еталонних значень:

– відхилення, що не перевищує 1 мм – оцінюється як відмінно;

– відхилення в межах від 1 до 2 мм – добре;

– відхилення в межах від 2 до 3 мм – задовільно;

– відхилення понад 3 мм – незадовільно.

Зміна параметрів функції Canny безпосередньо впливає на якість розпізнавання об’єктів, тому доцільно дослідити їх вплив на результати сканування.

Перший і другий аргументи цієї функції визначають нижню та верхню межі порогових значень гістерезису відповідно.

Кожен експеримент буде включати по десять сканувань, під час яких об’єкт незначно змінюватиме своє положення.

Такий підхід дозволяє отримати репрезентативні результати, оскільки незначна вибірка не забезпечує надійної оцінки точності.

Перший експеримент проведено із пороговими значеннями, що становлять 50 та 700 (рисунок 3.9).

Перший експеримент продемонстрував, що на поточному етапі підібрані параметри функції Canny є невдалими – більшість отриманих результатів класифікуються як незадовільні.

Лише в окремих випадках, зокрема у третьому та сьомому скануваннях, програмному забезпеченню вдалося частково коректно виявити контури об’єкта, проте навіть ці результати не можна вважати прийнятними, оскільки за таких параметрів алгоритму результат значною мірою є випадковим.

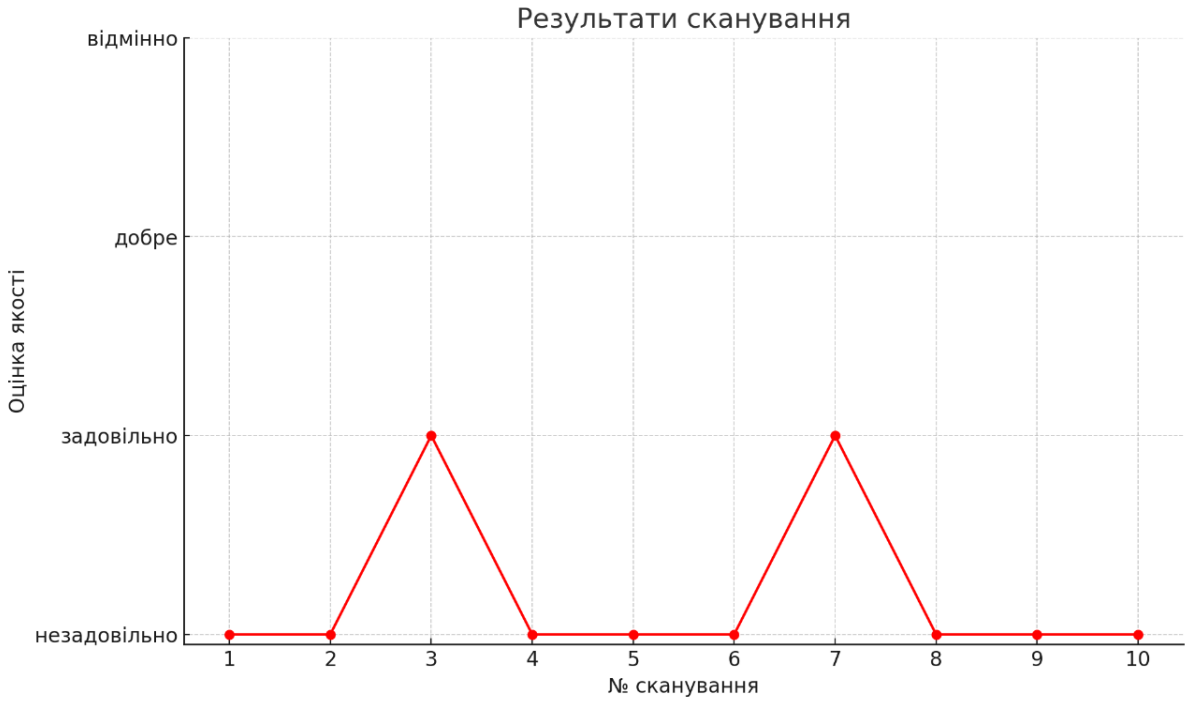


Рисунок 3.9 – Результати першого експерименту

Отримані результати підтверджують необхідність подальших експериментів з метою підвищення точності визначення розмірів об’єкта.

У другому експерименті перший пороговий параметр було збільшено на 50, а другий – зменшено на 200, що дозволить оцінити вплив зменшення діапазону гістерезису.

Підсумкові значення параметрів функції Canny для цього експерименту становлять 100 і 500 відповідно (рисунок 3.10).

Отримані результати другого експерименту свідчать про певне покращення точності – переважна більшість вимірювань оцінені як задовільні.

У десятому вимірюванні вперше було зафіксовано оцінку добре, що свідчить про позитивну динаміку в налаштуванні параметрів обробки.

Разом із тим, загальний рівень якості розпізнавання об’єктів залишається нижчим за бажаний, оскільки оцінки не перевищують порогу «задовільно», а отже, експериментальна робота має бути продовжена.

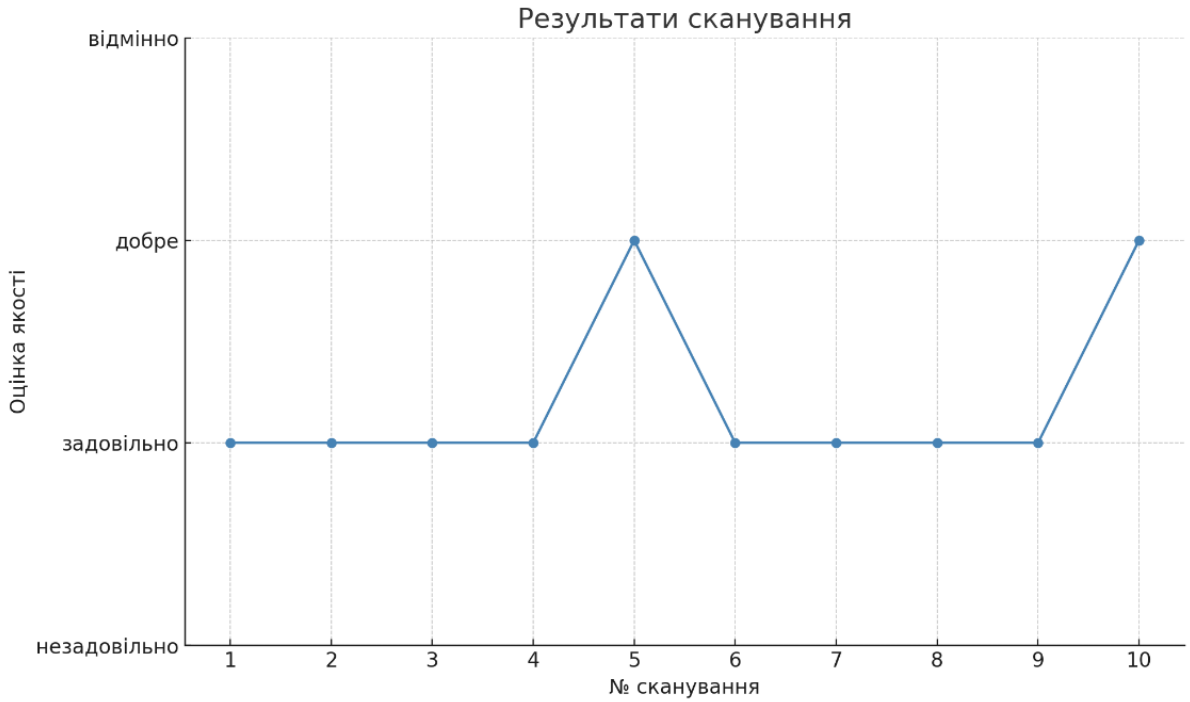


Рисунок 3.10 – Результати другого експерименту

У третьому експерименті перший пороговий параметр функції Canny було збільшено ще на 50, а другий – зменшено на 200.

Таким чином, нові порогові значення становлять 150 і 300 відповідно (рисунок 3.11).

Третій експеримент показав суттєве покращення результатів: усі оцінки якості сканування були на рівні не нижче «добре», а в першому, другому та сьомому скануваннях зафіксовано результат «відмінно».

Загалом, отримані значення відповідають поставленим критеріям точності, і результати можна вважати задовільними з практичної точки зору.

Однак з метою подальшого підвищення точності розпізнавання було прийнято рішення провести додаткові експерименти.

Четвертий експеримент засвідчив найкращі показники якості, яких вдалося досягти завдяки оптимізації аргументів функції Canny.

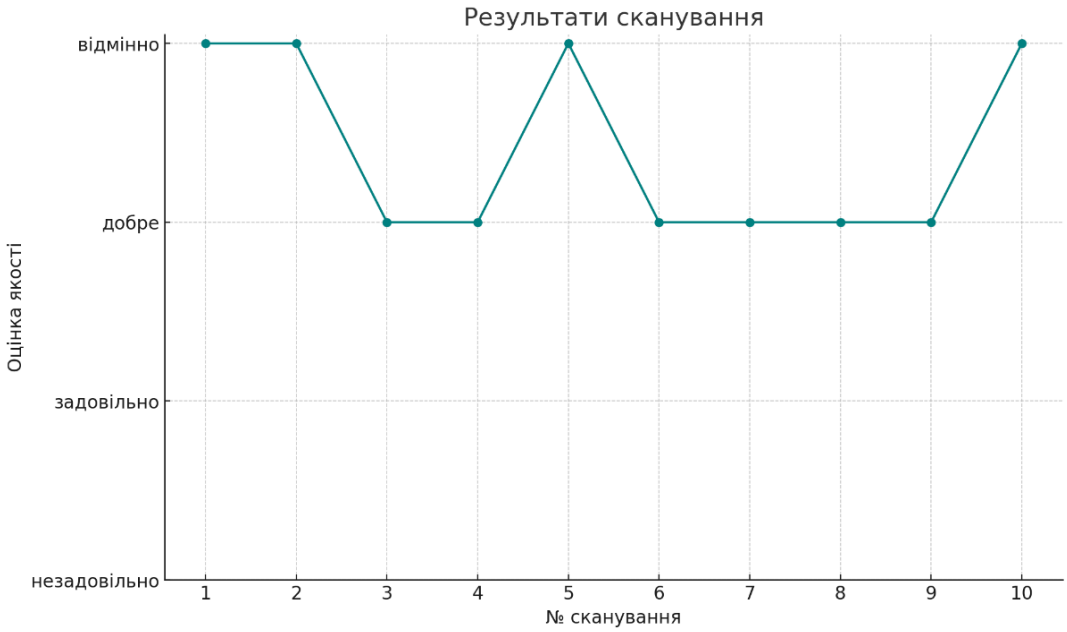


Рисунок 3.11 – Результати третього експерименту

Підібрані значення порогів у цьому випадку становлять 100 і 300 відповідно (рисунок 3.12).

Четвертий експеримент продемонстрував найвищий рівень точності серед усіх проведених серій вимірювань: шість результатів отримали оцінку «відмінно», а решта чотири – «добре».

Це свідчить про досягнення оптимальних параметрів функції обробки зображення, зокрема Canny, для конкретних умов системи.

Подальша модифікація порогових значень функції не призвела до помітного покращення якості, однак уже досягнуті результати можна вважати більш ніж задовільними, оскільки відхилення виміряних розмірів не перевищує одного міліметра від еталонних значень об’єкта.

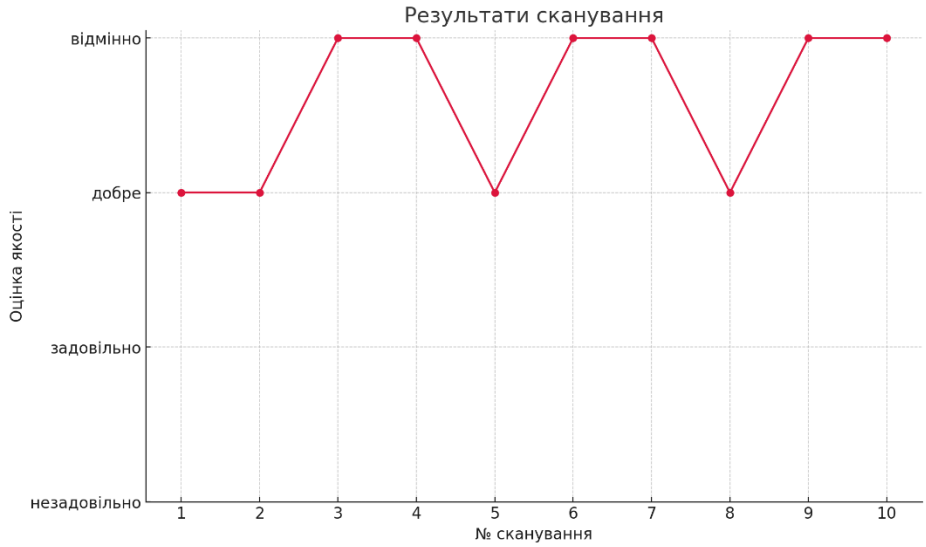


Рисунок 3.12 – Результати четвертого експерименту

Можливості підвищення точності в майбутньому можуть бути реалізовані за рахунок апаратної та програмної модернізації системи.

До таких удосконалень слід віднести використання відеокамери з вищою роздільною здатністю, встановлення додаткового освітлення, вибір фону з високим контрастом до кольору об’єкта сканування, а також оптимізацію самого алгоритму обробки зображення.

Візуалізацію результату успішного сканування наведено на рисунку 3.13.

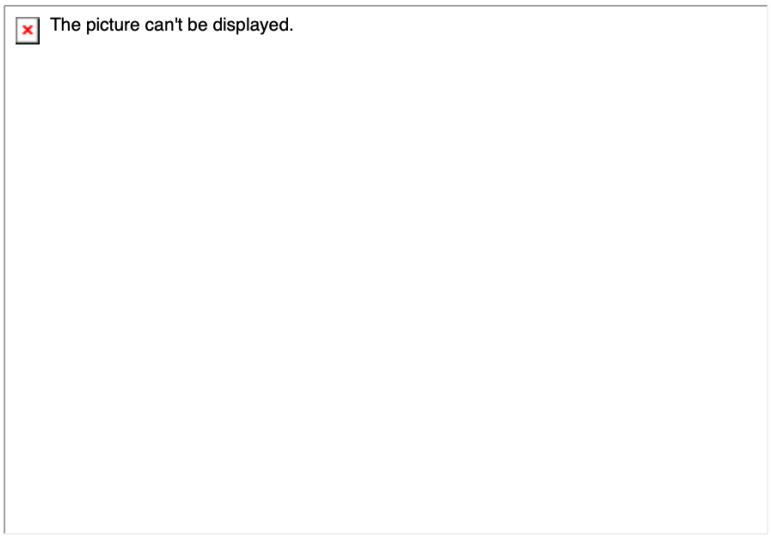


Рисунок 3.13 – Захоплення об’єкта з оцінкою «відмінно»

# ВИСНОВКИ

У межах кваліфікаційної роботи було успішно розроблено систему візуального аналізу для автоматизації виробничої лінії, що базується на апаратно-програмному рішенні Raspberry Pi з використанням алгоритмів комп’ютерного зору. Робота дала змогу підвищити ефективність контролю якості продукції та продемонструвала можливість впровадження недорогих, але потужних систем технічного зору у виробничі процеси.

Було проведено детальний огляд сучасних систем комп’ютерного зору, промислових сенсорів та камер, а також методів автоматичного аналізу зображень. Це дозволило виділити ключові переваги (висока точність, масштабованість) і обмеження (вимоги до ресурсів, вартість) наявних рішень.

Обґрунтовано використання мікрокомп’ютера Raspberry Pi у поєднанні з камерою та Python-орієнтованим середовищем як оптимального варіанту для задач візуального аналізу, враховуючи співвідношення вартість–продуктивність.

Створено структурну схему, що описує взаємодію компонентів (камера, Raspberry Pi, блок живлення, мережевий інтерфейс), визначено функціональні блоки системи та їхні технічні параметри.

Розроблено покроковий алгоритм, що включає етапи захоплення зображення, попередньої обробки, виявлення об’єктів, класифікації й подання результатів користувачу.

Розроблено програмний код на Python з використанням бібліотеки OpenCV, проведено налаштування параметрів обробки зображень (фільтрація, сегментація, контурний аналіз).

Система була протестована на імпровізованій виробничій лінії, результати показали високу точність розпізнавання зразків продукції та стабільність роботи в реальному часі.

Розроблена система візуального аналізу є ефективним рішенням для автоматизації контролю якості на виробничих лініях, забезпечує підвищення продуктивності, зниження людського фактору й мінімізацію браку. Робота підтвердила актуальність використання систем комп’ютерного зору на базі доступних апаратних платформ у промислових задачах.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chris E., Jonhson K., & Allison J. The Evolution of Computer Vision: From Pixels to Perception. ResearchGate. URL: https://www.researchgate.net/publication/390066290_The_Evolution_of_Computer_Vision_From_Pixels_to_Perception (дата звернення: 25.04.2025).

2. GeeksforGeeks. What is Machine Vision? URL: https://www.geeksforgeeks.org/what-is-machine-vision/ (дата звернення: 25.04.2025).

3. TechnoLynx. Computer Vision for Quality Control in Manufacturing. URL: https://www.technolynx.com/post/computer-vision-for-quality-control-in-manufacturing (дата звернення: 25.04.2025).

4. Intellias. Top Computer Vision Applications Across Industries. URL: https://intellias.com/top-computer-vision-applications-for-industries/ (дата звернення: 25.04.2025).

5. Євсеєнко, О. М., Гапон, А. І., & Крилова, В. А. Розробка автоматизованої системи розпізнавання показань приладів за допомогою камери машинного зору. ResearchGate. URL: https://www.researchgate.net/publication/361817195_Rozrobka_avtomatizovanoi_sistemi_rozpiznavanna_pokazan_priladiv_za_dopomogou_kameri_masinnogo_zoru (дата звернення: 25.04.2025).

6. Kibsi. Revolutionizing Quality Control in Manufacturing with Computer Vision & AI. URL: https://www.kibsi.com/manufacturing/revolutionizing-quality-control-manufacturing-computer-vision/ (дата звернення: 25.04.2025).

7. Venkat. OpenCV for Computer Vision: Basics of Image Processing using OpenCV. Medium. URL: https://medium.com/ai-in-plain-english/opencv-for-computer-vision-9dd2c3c35f22 (дата звернення: 25.04.2025).

8. Photoneo. 2D vs 3D Vision System: Finding a Perfect Fit For Your Application. URL: https://www.photoneo.com/2d-vs-3d-vision-system/ (дата звернення: 25.04.2025).

9. Hütten N., Gomes M., Hölken F., Meisen T. Deep Learning for Automated Visual Inspection in Manufacturing and Maintenance: A Survey of Open-Access Papers. URL: https://www.researchgate.net/publication/377620577_Deep_Learning_for_Automated_Visual_Inspection_in_Manufacturing_and_Maintenance_A_Survey_of_Open-_Access_Papers (дата звернення: 25.04.2025).

10. Tokyo Electron. What is a CMOS Image Sensor? URL: https://www.tel.com/museum/exhibition/principle/cmos.html (дата звернення: 25.04.2025).

11. Learn Robotics. Raspberry Pi GPIO: A Beginner's Guide. URL: https://www.learnrobotics.org/blog/raspberry-pi-gpio/ (дата звернення: 25.04.2025).

12. Raspberry Pi Foundation. Camera Software Documentation. URL: https://www.raspberrypi.com/documentation/computers/camera_software.html (дата звернення: 25.04.2025).

13. Sand M. Raspberry Pi 4 ARM Cortex-A72 Processor. URL: https://sandsoftwaresound.net/raspberry-pi-4-arm-cortex-a72-processor/ (дата звернення: 25.04.2025).

14. Raspberry Pi Foundation. Raspberry Pi 4 Datasheet. URL: https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf (дата звернення: 25.04.2025).

15. ebrary.net. Acquisition of Image. URL: https://ebrary.net/201167/education/acquisition_image (дата звернення: 25.04.2025).

16. Kargin, K. Computer Vision Fundamentals and OpenCV Overview. Medium. URL: https://keremkargin.medium.com/computer-vision-fundamentals-and-opencv-overview-9a30fe94f0ce (дата звернення: 25.04.2025).

17. OpenCV. Introduction to Machine Learning module – OpenCV 4.x documentation. URL: https://docs.opencv.org/4.x/dc/dd6/ml_intro.html (дата звернення: 25.04.2025).

18. Viso Suite. OpenCV: The Complete Guide to Computer Vision With OpenCV. URL: https://viso.ai/computer-vision/opencv/ (дата звернення: 25.04.2025).

19. DOU.ua. Рейтинг мов програмування в Україні 2023. URL: https://dou.ua/lenta/articles/language-rating-2023/ (дата звернення: 25.04.2025).

20. GeeksforGeeks. Python Language – Advantages and Applications. URL: https://www.geeksforgeeks.org/python-language-advantages-applications/ (дата звернення: 25.04.2025).