

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 охоронних систем

(повне найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»
РОБОТ ДЛЯ СОРТУВАННЯ ОБ'ЄКТІВ ЗА КОЛЬОРОМ
ROBOT FOR SORTING OBJECTS BY COLOR**

спеціальність 123 Комп'ютерна інженерія

(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія

(назва освітньої програми)

Виконав: здобувач вищої освіти
групи КІ-41

Веремчук Дарина Павлівна

(підпис)

Керівник:

к.т.н., доцент

Мельник Катерина Вікторівна

(підпис)

Кваліфікаційну роботу

допущено до захисту

« _____ » червня 2026 р.

Гарант освітньої програми:

к.т.н., доцент

Лавренчук Світлана Василівна

(підпис)

Луцьк – 2026 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

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

ЗАТВЕРДЖУЮ

В.О. завідувача кафедри

доц. Т. Терлецький

« 06 » 05 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Веремчук Дарині Павлівні

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи *Робот для сортування об'єктів за кольором*

Керівник роботи *к.т.н., доцент Мельник Катерина Вікторівна*

затверджені наказом закладу вищої освіти від « 06 » травня 2026 року № 242/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи 28.05.2026 р.

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, різні інтернет-ресурси технічного спрямування*

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

Вступ

Теоритичні основи комп'ютерного зору

Проектування роботизованої системи сортування

Реалізація та дослідження системи сортування

Висновки

5. Перелік графічного (ілюстративного) матеріалу:

Теоритичні основи комп'ютерного зору

Моделі представлення кольору

Принцип роботи сенсора кольору EV3

Алгоритм розпізнавання кольору

АНОТАЦІЯ

Веремчук Д.П. Робот для сортування об'єктів за кольором. Рукопис.

Кваліфікаційна робота бакалавра ОП «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія. Луцький національний технічний університет. Луцьк, 2026.

Кваліфікаційна робота присвячена дослідженню та практичній реалізації роботизованої системи сортування об'єктів за кольоровими ознаками із використанням сенсорних технологій та елементів комп'ютерного зору. Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

У першому розділі розглянуто теоретичні основи комп'ютерного зору та методи обробки кольорових характеристик об'єктів. Проаналізовано принципи цифрової обробки зображень, сегментації, класифікації об'єктів за кольором, особливості кольорових моделей RGB та HSV, а також підходи до використання сенсорів кольору в робототехнічних системах.

Другий розділ присвячено проектуванню роботизованої системи сортування. Описано конструкцію роботизованої платформи на базі LEGO Mindstorms EV3, обґрунтовано вибір апаратних компонентів, сенсора кольору та середовища програмування LEGO Mindstorms EV3 Software. Розглянуто структуру програмного забезпечення та алгоритм функціонування системи.

У третьому розділі наведено практичну реалізацію алгоритму розпізнавання кольору, результати тестування роботизованої системи та оцінювання ефективності її роботи. Проведено аналіз точності розпізнавання кольорів, швидкодії системи та впливу зовнішніх факторів на стабільність функціонування.

Ключові слова: комп'ютерний зір, робототехніка, LEGO Mindstorms EV3, сенсор кольору, сортування об'єктів, розпізнавання кольору, алгоритм, роботизована система.

ANNOTATION

Veremchuk D.P. Robot for sorting objects by color. Manuscript.

Qualifying bachelor's thesis of the Educational Program "Computer Engineering", specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2026.

The qualification work is devoted to the study and practical implementation of a robotic system for sorting objects by color characteristics using sensor technologies and elements of computer vision. The thesis consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The first chapter presents the theoretical foundations of computer vision and methods for processing color characteristics of objects. The principles of digital image processing, segmentation, color-based object classification, features of RGB and HSV color models, and approaches to the use of color sensors in robotic systems are analyzed.

The second chapter is devoted to the design of the robotic sorting system. The structure of the LEGO Mindstorms EV3 robotic platform is described, the choice of hardware components, color sensor, and LEGO Mindstorms EV3 Software development environment is substantiated. The structure of the software and the operating algorithm of the system are considered.

The third chapter presents the practical implementation of the color recognition algorithm, the results of testing the robotic system, and the evaluation of its efficiency. The accuracy of color recognition, system response speed, and the influence of external factors on the stability of operation are analyzed.

Keywords: computer vision, robotics, LEGO Mindstorms EV3, color sensor, object sorting, color recognition, algorithm, robotic system.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ КОМП'ЮТЕРНОГО ЗОРУ	10
1.1 Сучасні напрями застосування комп'ютерного зору	10
1.2 Структура та етапи обробки цифрових зображень	13
1.3 Методи обробки та класифікації об'єктів за кольоровими ознаками	19
РОЗДІЛ 2 ПРОЄКТУВАННЯ РОБОТИЗОВАНОЇ СИСТЕМИ СОРТУВАННЯ.....	25
2.1 Конструкція сортувального робота	25
2.2 Вибір апаратної платформи та сенсорів	29
2.3 Програмне забезпечення системи	32
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ СИСТЕМИ СОРТУВАННЯ.....	36
3.1 Реалізація алгоритму розпізнавання об'єктів за кольором	36
3.2 Моделювання та тестування роботизованої системи.....	40
3.3 Оцінка ефективності роботи системи	44
ВИСНОВКИ	47
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	48

ВСТУП

На сучасному етапі розвитку автоматизації ключову роль відіграють системи, призначені для сортування об'єктів за заданими характеристиками. Вони знаходять широке застосування у виробничих процесах, сфері логістики та системах контролю якості, забезпечуючи оперативне та точне виявлення й розподіл об'єктів. Однією з основних ознак класифікації виступає колір, що дає змогу ефективно проводити первинне сортування без необхідності аналізувати складні геометричні параметри.

У сучасних автоматизованих системах аналіз кольорових характеристик можна здійснювати двома основними підходами: через застосування систем комп'ютерного зору або за допомогою спеціалізованих сенсорів кольору. У свою чергу, методи комп'ютерного зору забезпечують високу точність аналізу, однак мають певні обмеження, пов'язані з необхідністю використання значних обчислювальних ресурсів, цифрових камер та спеціалізованих алгоритмів обробки зображень. Такі вимоги істотно ускладнюють інтеграцію цих систем у роботизовані механізми з обмеженою продуктивністю чи мінімальною апаратною базою.

Як альтернатива, широко застосовуються інтегральні сенсори кольору, які функціонують на основі аналізу відбитого світлового потоку. Цей підхід не лише знижує вимоги до програмного забезпечення та апаратної інфраструктури, а й дозволяє працювати в режимі реального часу завдяки меншій потребі в обчислювальних ресурсах. Проте результати роботи сенсорів кольору можуть значно залежати від зовнішніх умов, таких як рівень освітлення, дистанція між сенсором і об'єктом, а також оптичні властивості поверхні матеріалу. Така залежність обумовлює необхідність детального дослідження функціональних особливостей сенсорів кольору та розробки спеціалізованих алгоритмів для забезпечення їхньої стабільної роботи в різноманітних експлуатаційних умовах.

Автоматизоване сортування об'єктів за заданими ознаками є важливою складовою багатьох технічних процесів, зокрема у виробництві, логістиці та системах контролю якості. Одним із параметрів, який широко використовується

для класифікації, є колір об'єкта. Його визначення може здійснюватися як шляхом аналізу зображень, так і за допомогою спеціалізованих сенсорів, що фіксують характеристики відбитого світла.

Використання сенсорів кольору дозволяє реалізувати задачі розпізнавання без обробки повноцінних цифрових зображень. Такий підхід зменшує обсяг обчислень і спрощує програмну реалізацію, що є доцільним для роботизованих платформ із обмеженими ресурсами. Разом із тим точність визначення кольору залежить від умов освітлення, відстані до об'єкта та властивостей його поверхні, що потребує врахування при розробці алгоритмів.

Робототехнічні платформи навчального типу, зокрема LEGO Mindstorms EV3, надають можливість реалізувати подібні задачі з урахуванням інтеграції сенсорів, виконавчих механізмів та програмного забезпечення. Це дозволяє дослідити принципи побудови автоматизованих систем та перевірити їх роботу в експериментальних умовах.

Метою роботи є розробка та дослідження роботизованої системи сортування об'єктів за кольоровими ознаками на основі сенсорного визначення кольору.

Об'єктом дослідження є процес сортування об'єктів у роботизованих системах.

Предметом дослідження є методи розпізнавання кольору об'єктів із використанням сенсорних технологій.

Для досягнення поставленої мети визначено такі завдання:

- проаналізувати підходи до визначення кольорових характеристик об'єктів;
- дослідити принцип роботи сенсора кольору та його обмеження;
- розробити конструкцію роботизованої платформи та обґрунтувати вибір апаратних компонентів;
- реалізувати алгоритм розпізнавання кольору;
- провести тестування системи в різних умовах та оцінити точність та ефективність її роботи.

Дослідження виконувалося із застосуванням поєднання теоретичних та експериментальних методів. Теоретичний етап включав аналіз наукових і технічних джерел, що стосуються розпізнавання кольору, сенсорних систем та принципів побудови роботизованих комплексів. Це дало змогу визначити доцільний підхід до реалізації задачі з урахуванням обмежень обчислювальних ресурсів.

На етапі проєктування використовувалися методи алгоритмічного моделювання, що дозволили сформувати логічну структуру роботи системи та визначити послідовність обробки вхідних даних. Реалізація алгоритму здійснювалася у візуальному середовищі програмування LEGO Mindstorms EV3, яке забезпечує можливість побудови програм у вигляді блокових схем та їх безпосереднього виконання на роботизованій платформі.

Експериментальна частина роботи базувалася на використанні робототехнічної платформи LEGO Mindstorms EV3, яка включає програмований контролер, сенсор кольору та виконавчі механізми. Застосування цієї платформи дозволило перевірити працездатність алгоритму в умовах фізичної взаємодії з об'єктами, а не лише в межах абстрактної моделі. Для проведення досліджень було організовано серію тестових випробувань із використанням об'єктів різних кольорів.

Оцінювання результатів здійснювалося на основі порівняння фактичних і очікуваних результатів розпізнавання, а також аналізу стабільності роботи системи за різних умов. Для цього використовувалися елементи кількісного аналізу, зокрема визначення точності розпізнавання та часу реакції системи. Отримані дані були узагальнені у вигляді таблиць, що дозволило провести їх подальший аналіз.

У роботі використано методи аналізу, алгоритмічного проєктування, моделювання та експериментальної перевірки.

Практичне значення отриманих результатів полягає у створенні роботизованої системи, здатної виконувати сортування об'єктів за кольором, а також у можливості використання розробленого підходу для подальшого ускладнення алгоритмів і розширення функціональності системи.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ОСНОВИ КОМП'ЮТЕРНОГО ЗОРУ

1.1 Сучасні напрями застосування комп'ютерного зору

Комп'ютерний зір є одним із основних напрямів штучного інтелекту, спрямованим на те, щоб технічні системи могли імітувати здатність людського зору. Сьогодні комп'ютерний зір є важливою основою для розробки «розумних» систем, які здатні працювати в реальному часі і адаптуватися до змін у навколишньому середовищі. Комп'ютерний зір можна розглядати як доповнення до біологічного зору, не обов'язково протиставляючи їх. У біології досліджується зорове сприйняття людини й тварин, формуючи моделі цих процесів на основі фізіологічних досліджень. Зі свого боку, комп'ютерний зір займається вивченням і розробкою систем, які реалізуються апаратно чи програмно. Співпраця та обмін знаннями між цими двома сферами виявилися дуже плідними для обох наукових напрямків. Галузь комп'ютерного зору можна описати як молодий і різноманітний напрямок. Хоча існують дослідження, що розпочалися ще раніше, активне вивчення цієї теми набрало обертів лише з кінця 1970-х років, коли з'явилися можливості обробляти великі обсяги даних, такі як зображення. Ці дослідження зазвичай стартували з інших сфер, тому немає стандартного визначення проблеми комп'ютерного зору. Що ще важливіше, не існує й узгодженого підходу до вирішення цієї проблеми. Натомість є безліч методів для вирішення різних чітко визначених завдань комп'ютерного зору, де вибір методів залежить від конкретного завдання і рідко може бути використаний для широкого спектра застосувань. Багато методів і застосувань досі перебувають на етапі ґрунтовного вивчення, проте все більше з них знаходять застосування в комерційних продуктах у складі складніших систем, здатних вирішувати комплексні завдання, як-от у медичних зображеннях або в контролі якості у виробничих процесах [20].

У багатьох випадках використання комп'ютерного зору комп'ютери попередньо налаштовані для вирішення конкретних завдань, але методи на основі знань стають усе більш універсальними.

Найбільш прийнятна нейропсихологічна модель описує початковий етап розпізнавання об'єктів як процес, в якому мозок спершу інтерпретує основні характеристики об'єктів, такі як форма, колір і глибина. Сигнали від ока, які надходять у мозок, аналізуються для визначення контурів об'єкта, і ці контури з'єднуються у більш складне уявлення, що доповнює форму. Системи комп'ютерного зору працюють схожим чином: спершу вони визначають контури об'єкта, а тоді об'єднують ці контури у форму. Головна відмінність полягає у тому, що комп'ютери оперують зображеннями у числовому форматі, тому система комп'ютерного зору потребує специфічного методу для обробки пікселів, з яких складається зображення. Комп'ютер присвоює значення пікселям на зображенні, і, аналізуючи різницю між значеннями в різних ділянках, система може виявити контури. Наприклад, у відтінках сірого значення змінюються від чорного (0) до білого (255). Різка зміна значень між суміжними пікселями вказує на наявність краю. Цей принцип порівняння значень пікселів також застосовний до кольорових зображень, де комп'ютер порівнює різниці між каналами RGB [21].

Одним із найважливіших підходів, використовуваних у галузі комп'ютерного зору, є нейронні мережі. Ці мережі складаються з численних шарів нейронів, що функціонують спільно задля ідентифікації та класифікації об'єктів на візуальних даних. Вони мають властивість автоматичного розпізнавання шаблонів і характеристик на зображеннях, що забезпечує їхню високу ефективність у вирішенні завдань комп'ютерного зору. Комп'ютерний зір використовує алгоритми, які дозволяють комп'ютеру аналізувати зображення і відео, виокремлюючи з них корисну інформацію шляхом обробки великих обсягів даних. Основні алгоритми, застосовувані у цій сфері, включають виокремлення ознак та класифікацію. Алгоритми для виокремлення ознак допомагають виявляти важливі елементи зображень, такі як грані, кути та кольорові області, завдяки чому комп'ютер здатен розпізнавати об'єкти.

Класифікаційні алгоритми використовуються для визначення категорії об'єкта на зображенні. Серед вад існуючих алгоритмів, що використовуються у комп'ютерному зорі для обробки зображень, можна виділити такі аспекти:

- під час розпізнавання об'єкта один або кілька об'єктів можуть бути ідентифіковані разом із їхнім двовимірним положенням на зображенні або тривимірним розташуванням у просторі сцени;

- недостатня точність в ідентифікації та порівнянні індивідуальних біометричних характеристик об'єкта є суттєвим недоліком;

- виклики у правильній інтерпретації, що виникають при виявленні, наприклад, потенційних змін клітин або тканин та їх властивостей під час медичних досліджень, а також обмежень у системах управління робототехнікою та іншими подібними сферами [24].

Комп'ютерний зір знаходить численні застосування в різноманітних промислових секторах. У сфері охорони здоров'я ця технологія може стати вагомим інструментом для рентгенологів, істотно полегшуючи їм завдання точного інтерпретування медичних знімків. У роздрібній торгівлі технології комп'ютерного зору використовуються для детального аналізу поведінки клієнтів і контролю за асортиментом на полицях, тоді як у виробничій галузі вони служать засобом контролю якості та здійснення прогнозованого обслуговування обладнання. Надійна робота автономних транспортних засобів тісно пов'язана з комп'ютерним зором, який виконує критичну функцію розпізнавання об'єктів, забезпечуючи тим самим їхню безпечну навігацію на дорогах. Ще однією сферою застосування комп'ютерного зору є індустрія розваг, особливо в контексті віртуальної реальності (VR), де ключову роль відіграє візуальне занурення користувача. Завдяки використанню камер для високоточного відстеження реальних рухів користувачів у навколишньому середовищі, комп'ютери здатні створювати переконливий ефект занурення, накладаючи цифрові зображення на реальний світ з мінімальною затримкою. Це включає встановлення завдань, наприклад, класифікацію зображень. Серед популярних бібліотек для завдань комп'ютерного зору виділяються PyTorch і TensorFlow.

Алгоритми комп'ютерного зору розроблені для аналізу зображень і виявлення шаблонів. Для цього існує безліч інструментів, таких як OpenCV, TensorFlow, CUDA, MATLAB, Keras і SimpleCV. Окрім цього, Microsoft Azure Computer Vision API надає можливість використовувати попередньо навчені моделі для аналізу зображень [25].

Основні функції комп'ютерного зору:

- розпізнавання об'єктів;
- локалізація об'єктів;
- класифікація зображення;
- сегментація зображень;
- відстеження об'єктів;
- розпізнавання облич;
- оптичне розпізнавання символів;
- оцінка глибин 3D рекомендація;
- системи автономного управління.

Комп'ютерний зір є не просто окремою технологією, а потужним інструментом, що надає звичайним технічним системам «свідомість», роблячи їх спроможними самостійно аналізувати навколишнє середовище. Його ефективність заснована на інтеграції програмних алгоритмів та апаратного забезпечення, що дозволяє впроваджувати комп'ютерний зір у реальному часі з високою точністю і можливістю масштабування.

1.2 Структура та етапи обробки цифрових зображень

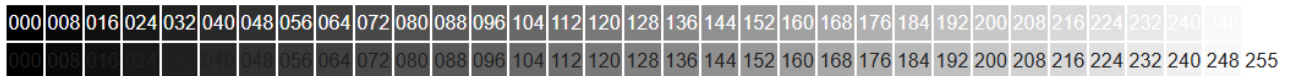
Обробка цифрових зображень охоплює декілька ключових етапів:

- цифрове подання зображень;
- попередня обробка;
- сегментація;
- виділення характеристик;
- інтерпретація.

Цифрове зображення являє собою масив даних, створений шляхом перетворення оригіналу в цифровий формат за допомогою дискретизації. Після застосування певного алгоритму для кодування та запису на носій інформації, цей масив даних перетворюється у файл. У сучасному поліграфічному виробництві всі ілюстрації та елементи оформлення представлені різними типами цифрових зображень. За методом перетворення оригіналу вони розподіляються на растрові, векторні та змішані типи.

Растрові зображення становлять собою двовимірні масиви даних, відомі як матриці пікселів, де кожен елемент відображає фрагмент оригінального зображення з усередненим значенням кольору. Растрові зображення можна отримати двома основними способами. Перший спосіб полягає в скануванні оригіналу за допомогою спеціального пристрою – сканера, у якому кожен оптичний елемент ПЗЗ-лінійки або ПЗЗ-матриці зчитує яскравіші та кольорові характеристики оригіналу. Ці дані перетворюються в двійковий код кольору і зберігаються в осередках двовимірного масиву даних, який називається матрицею пікселів. Другий спосіб отримання растрового зображення полягає у проектуванні оригіналу на ПЗЗ-матрицю через систему лінз, тобто об'єктив. Такий метод аналогово-цифрового перетворення характерний для цифрових фотоапаратів та відеокамер. Основні особливості растрового зображення – це його розмір і глибина кольору. Під розміром зображення мається на увазі кількість рядків і стовпців у матриці пікселів, що використовуються для його збереження. Цифрове зображення можна змінити, регулюючи його фізичний розмір під час друку, при цьому кількість пікселів у матриці залишатиметься незмінною. Глибина кольору вказує на якість передачі кольорів і кількість відтінків, які можуть бути відображені пікселями матриці. Кожен елемент матриці є числом у двійковій системі числення, розмір якого визначається в бітах. Глибина кольору – це кількість біт на піксель. Наприклад, зображення з глибиною кольору 16 біт на піксель може відтворювати 65 536 відтінків, тоді як 24 біт на піксель дозволяють отримати вже 16 777 216 кольорів, що повністю задовольняє потреби поліграфічного виробництва [27].

Одним байтом (8 біт) можна закодувати 256 відтінків (наприклад, сірого). У цьому випадку 0 представляє абсолютно чорний колір, а 255 – абсолютно білий (рис. 1.1).



Рисинок 1.1 – Глибина кольору від чорного до білого [26]

Колір пікселя визначається, як комбінація трьох кольорів: червоного (R), зеленого (G) та синього (B) в різних пропорціях (рис. 1.2).

R	G	B	Couleur
0	0	0	чорний
0	0	1	майже чорний
255	0	0	червоний
0	255	0	зелений
0	0	255	синій
128	128	128	сірий
255	255	255	білий

Рисунок 1.2 – Комбінація трьох кольорів [26]

У контексті інтернет-видань для оформлення веб-сайтів нерідко використовуються зображення з глибиною кольору 4, 2 або навіть 1 біт. Така низька кольорова точність часто є достатньою для відтворення технічної графіки, зокрема чорно-білих схем та діаграм. Процес аналого-цифрового перетворення завжди супроводжується втратою певної кількості інформації. Це відбувається через дискретизацію, яка здійснюється шляхом усереднення та узагальнення вхідної аналогової інформації.

Їх основне призначення полягає у представленні фотографічних ілюстрацій. У медіа індустрії растрові зображення отримують за допомогою сканерів та цифрових камер. Використання растрових зображень є оптимальним

у тих випадках, коли необхідно відтворити аналоговий оригінал, такий як фотографія, малюнок чи складний елемент оформлення, які недоцільно переводити у векторну форму. Сам ефірний відео потік складається з послідовності кадрів, кожен з яких є матрицею пікселів.

Якість сканованого зображення визначається роздільною здатністю, з якою воно було отримане. На ілюстрації показано фрагменти цифрового зображення, утворені при поступовому зниженні роздільності (рис. 1.3).



Рисунок 1.3 – Цифрове зображення при поступовому зниженні роздільності [26]

Іншим типом цифрових зображень є векторні зображення. Найменшою складовою частиною векторного зображення є вектор та крива Безьє. У контексті комп'ютерної графіки, вектор визначається як відрізок, що з'єднує дві точки з конкретними координатами. Основною одиницею управління кривою Безьє виступає вузол, який також називають контрольною точкою або контрольним вектором. Ступінь кривизни лінії залежить від координат вузла, а також від координат двох контрольних точок. Цифрове представлення контуру елемента векторного зображення складається з масиву даних, що включає координати контрольних та керуючих точок разом із характеристиками кривої, такими як її товщина, колір, напрямок, а для замкнених кривих – також інформацію про колір і тип заливки.

Примітиви в цьому контексті характеризуються як базові геометричні форми, що в даних зберігаються у вигляді цілісних записів. Вони не розділяються на криві Безьє й вектори, а кодуються за допомогою певних

умовних позначень, які вказують на геометричну форму, її розміри, координати розташування, а також тип і колір заливки, товщину і колір контуру та інші характеристики. Інформація про текстову частину зображення переважно подається у вигляді кодів символів, супроводжуваних цифровими даними про візуальні параметри тексту, зокрема шрифтові гарнітури, накреслення, кольорову палітру контуру і заливки, методи заливки та способи вирівнювання в межах блоку [27].

Створення векторних зображень відбувається двома основними способами: ручним та автоматичним трасуванням. При ручному трасуванні художник або дизайнер самостійно генерує зображення, використовуючи графічний редактор для формування векторів, кривих Безьє і графічних примітивів, ніби окреслюючи контури уявним малюнком. Автоматичне трасування виконується за допомогою програмного забезпечення, яке завдяки інтелектуальним алгоритмам розпізнає контури оригіналу, представленого растровим зображенням. Це дозволяє створювати лінії та заливки на основі отриманої інформації, що утворюють векторне зображення максимально наближене до оригіналу. Вагомою перевагою векторних зображень є їх здатність до масштабування без погіршення якості. Основним недоліком векторних зображень є те, що вони зазвичай відтворюють оригінал у спрощеній формі. Через це певні деталі оригіналу можуть бути втрачені у векторному форматі.

Цифрові зображення змішаного типу представляють собою складні масиви даних, що складаються з інформації у формі матриці пікселів, а також векторних описів, кривих Безьє, примітивів і текстових блоків. Одна з основних концепцій, що лежить в основі вертикальної структури векторно-растрових зображень, є поняття шару. Шар являє собою область даних, яка містить інформацію про окремі елементи вертикальної структури зображення. Векторно-растрові зображення формуються з вихідних векторних і растрових елементів через обробку в графічних редакторах.

Додатково, до цього типу зображень можна віднести результати діяльності програм комп'ютерної верстки, де головними векторними елементами є текстові блоки. Зображення змішаного типу інтегрують в собі переваги й недоліки тих

типів зображень, які наявні у них у формі окремих елементів або шарів. Крім того, сучасні графічні редактори надають можливість створення багатосторінкових зображень.

Головною перевагою зображень змішаного типу є здатність до автономного редагування кожного шару, тоді як суттєвим недоліком виступає значний обсяг даних масиву і відповідно кінцевого файлу. У процесі підготовки макетів використовують векторні шари для оформлення тексту і графічних символів (логотипів, товарних знаків, лого-груп, лінійок та інших), тоді як растрові шари застосовуються для підкладок і фотознімків.

У MATLAB передбачено використання спеціальних функціональних бібліотек (toolbox), які значно розширюють можливості програми. Серед них можна виділити Digital Signal Processing Toolbox, Image Processing Toolbox, Wavelet Toolbox та Filter Design Toolbox. Ці інструменти активно використовуються для цифрової обробки сигналів та зображень, забезпечуючи гнучкість і ефективність у виконанні складних завдань [22].

Сегментація – це процес поділу цифрового зображення на кілька частин, які часто називають суперпікселями. Основна мета цього процесу полягає в спрощенні або зміні представлення зображення для легшого аналізу або передавання каналами зв'язку. Зазвичай сегментація використовується для виділення об'єктів та окреслення меж, таких як лінії або криві, на зображеннях. Фактично, цей процес полягає у присвоєнні міток кожному пікселю, щоб пікселі з однаковими мітками мали спільні візуальні характеристики. Сегментація зображення призводить до утворення набору сегментів, які повністю покривають зображення, або набору контурів, виділених з нього. Усі пікселі в межах одного сегмента мають подібні характеристики або властивості, такі як колір, яскравість чи текстура. У той же час сусідні сегменти суттєво різняться за цими ознаками.

Одним з найпростіших підходів до сегментації зображень є метод порогового визначення. Цей метод заснований на використанні певного рівня порогу для перетворення зображення в сірій шкалі у бінарне зображення. Важливим аспектом цього методу є вибір відповідного порогового значення (або кількох значень при необхідності множинного порогу). У промисловості

застосовуються кілька популярних методів визначення порогів, серед яких метод максимальної ентропії, метод Отсу (максимального відхилення) та кластерний аналіз методом к-середніх.

Відносно нещодавно були розроблені новітні методи порогового оброблення для комп'ютерної томографії. Основною відмінністю цих методів від підходу Отсу є те, що пороги отримуються безпосередньо з радіографій, а не з зображень, утворених на їх основі. Нещодавні розробки пропонують впровадження багатовимірних нелінійних порогів, заснованих на правилах нечіткої логіки. У цих дослідженнях рішення щодо входження кожного пікселя до певного сегмента приймається на основі багатовимірних правил, отриманих з використанням нечіткої логіки та еволюційних алгоритмів, що враховують освітленість зображення та мету його подальшого використання.

Інтерпретація є процесом коли система робить висновки про природу об'єкта на основі обчислених ознак. Це може бути:

- класифікація об'єктів;
- розпізнавання образів;
- оцінка стану об'єкта.

Кожен етап, починаючи з цифрового представлення до інтерпретації, відіграє важливу роль в ефективності систем комп'ютерного зору. Послідовне виконання цих етапів формує основу для автоматизованих систем аналізу, зокрема у таких завданнях, як сортування об'єктів за кольором, контроль якості та медична діагностика [17].

1.3 Методи обробки та класифікації об'єктів за кольоровими ознаками

Колірна модель – це метод кодування кольорів шляхом упорядкування числових значень відповідно до певних базових компонентів.

Модель RGB представляє собою апаратно-орієнтовану концепцію, в якій кольори визначаються шляхом поєднання трьох основних кольорів: червоного,

зеленого та синього. Ці кольори комбінуються у різних співвідношеннях для досягнення широкого спектра відтінків.

Саме через здатність додавання кольорів модель RGB називають адитивною.

Назва походить від перших літер англійських слів: R (RED) – червоний, G (GREEN) – зелений, B (BLUE) – синій. Кожен базовий колір може мати інтенсивність у діапазоні від 0 до 255. Загальна кількість кольорів, що ця модель може відтворити, становить $256 \cdot 256 \cdot 256 = 16\,777\,216$.

За допомогою моделі RGB описують кольори, які отримують змішуванням світлових променів. Цю модель використовують у моніторах, телевізорах, сканерах, слайд-проекторах, в рекламних кольорових лампах та інших пристроях, де колір формується шляхом змішування світлових пучків. Вона також застосовується для опису кольорів у мережі Інтернет у спеціальному шістнадцятковому форматі (#RRGGBB). Змінюючи інтенсивність свічення кольорових пікселів, можна створити безліч відтінків. Якщо інтенсивність кожного з них максимальна (255), утворюється білий колір (рис. 1.4) [16].



Рисунок 1.4 – Комбінації базових кольорів моделі RGB[23]

Колірна модель RGB є однією з базових моделей подання кольору в цифрових системах. Її назва походить від англійських слів Red, Green, Blue, що відповідають червоному, зеленому та синьому кольорам.

Принцип роботи цієї моделі ґрунтується на адитивному змішуванні світла: різні кольори утворюються шляхом поєднання трьох основних компонентів із різною інтенсивністю.

Кожен із каналів RGB може набувати значення в діапазоні від 0 до 255. Значення (0, 0, 0) відповідає чорному кольору, оскільки всі канали мають мінімальну інтенсивність. Значення (255, 255, 255) відповідає білому кольору, адже всі три канали мають максимальну інтенсивність.

Наприклад, червоний колір задається комбінацією (255, 0, 0), зелений - (0, 255, 0), синій - (0, 0, 255). Якщо змінювати співвідношення інтенсивності каналів, можна отримати широкий спектр відтінків.

У системах комп'ютерного зору модель RGB використовується для аналізу зображень, розпізнавання кольорових об'єктів і виділення потрібних ділянок сцени. Для роботизованої системи сортування ця модель є зручною, оскільки дає змогу визначати колір об'єкта на основі числових значень, отриманих від сенсора кольору або камери. Саме тому RGB-модель може бути використана як основа для розроблення алгоритму класифікації об'єктів за кольоровими ознаками.

Модель CMY(або ж CMYK) є концептуальною основою для створення друкованих зображень, призначених для пристроїв, які функціонують на принципі віднімання кольорів шляхом поглинання.

Друковані зображення суттєво відрізняються від екранних тим, що вони сприймаються у відбитому світлі, адже папір не генерує власного світла. Основу колірної моделі CMY складають кольори, що утворюються шляхом віднімання основних кольорів RGB від білого. Звідси походить назва цієї субтрактивної моделі – від англійського слова «to subtract», що означає «віднімати»[16]. Основні кольори моделі CMY такі (рис. 1.5):

- C (CYAN) – блакитний = білий – червоний = зелений + синій;
- M (MAGENTA) – пурпуровий = білий – зелений = червоний + синій;
- Y (YELLOW) – жовтий = білий – синій = червоний + зелений.

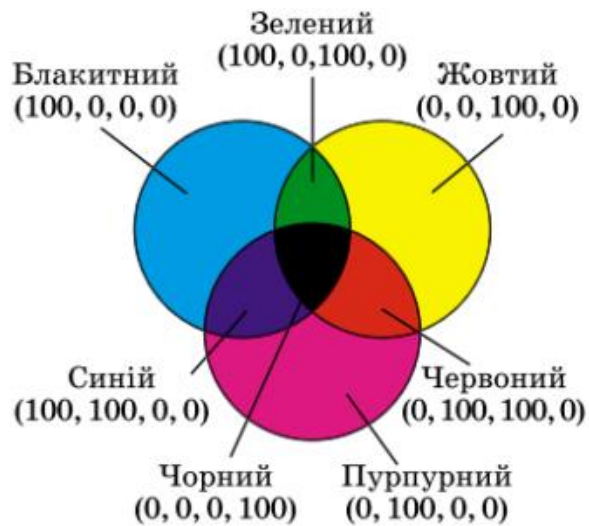


Рисунок 1.5 – Комбінації базових кольорів моделі CMY [23]

У колірних системах RGB і CMYK враховуються обмеження, що виникають через апаратне забезпечення – екрани комп'ютерів для RGB і друкарські фарби для CMYK. Проте більш природний підхід до опису кольору пропонує система HSB. Модель HSB являє собою систему кодування, що використовує цілі числа для представлення таких характеристик кольору, як відтінок (Hue), насиченість (Saturation) та яскравість (Brightness).

Перевагою моделі HSB над іншими колірними моделями є її відповідність природній характеристиці кольору, а також її здатність відображати модель сприйняття кольорів людиною. У цій моделі відтінок асоціюється з довжиною хвилі світла, насиченість визначає інтенсивність кольорової хвилі, а яскравість вказує на загальну кількість світла.

Яскравість позначає рівень освітленості кольору. При нульовій яскравості колір стає чорним. Найвища яскравість у поєднанні з максимальною насиченістю забезпечує найбільш виразний варіант кольору. По суті, яскравість показує введення чорного відтінку в колір. Яскравість чорного кольору є 0, тоді як білого – 1.

Графічна репрезентація моделі HSB може бути виконана у формі кольорового круга, на якому розташовані відтінки. Кожен відтінок відповідає певному градусу, загалом існує 360 варіантів (наприклад, червоний – 0 градусів,

жовтий – 60, зелений – 120 і так далі). По зовнішньому обводу круга представлені чисті спектральні тони (параметр H вимірюється в кутових градусах від 0 до 360).

Чим ближче колір до центру круга, тим менша його насиченість і тим більш пастельний він стає (параметр S вимірюється у відсотках). Яскравість (або освітленість) відображається через шкалу, перпендикулярно розташовану до площини колірного круга (параметр V вимірюється у відсотках). Усі кольори на зовнішньому колі мають максимальну яскравість [16] (рис. 1.6).

Модель HSB характеризується апаратною незалежністю, оскільки вона не орієнтована на технічні засоби відтворення кольорів.

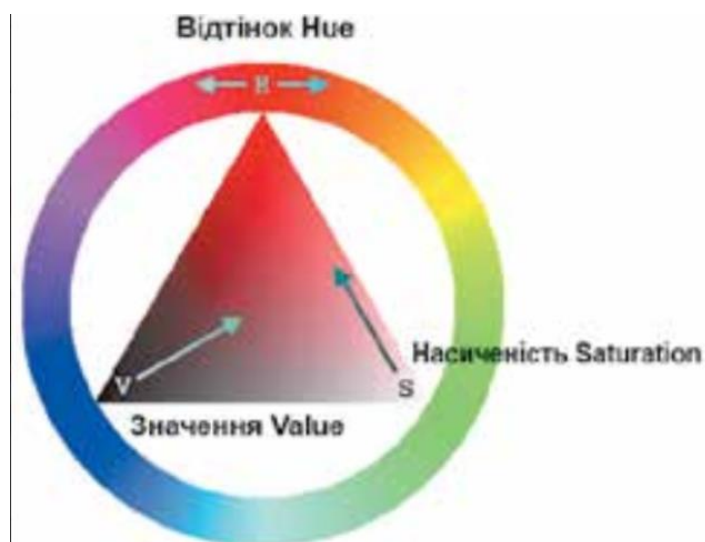


Рисунок 1.6 – Графічна репрезентація моделі HSB [23]

Центральним недоліком HSB є необхідність її конвертації в модель RGB для роботи з комп'ютерними моніторами чи в модель CMYK для реалізації чотириколірного друку. Отримати інформацію про код обраного кольору можна, використовуючи різноманітні програмні засоби.

Приміром, під час вибору кольору для заливки об'єктів у таких програмах, як Word та Excel, зазвичай надається код кольору у моделях RGB або HSB (рис. 1.7).



Рисунок 1.7 – Вибір кольору для заливки об’єктів у програмі Word/Excel
[23]

У програмі LibreOffice Draw доступне відображення кодів кольорів у моделях RGB, HSB та CMYK (рис. 1.8) [16].

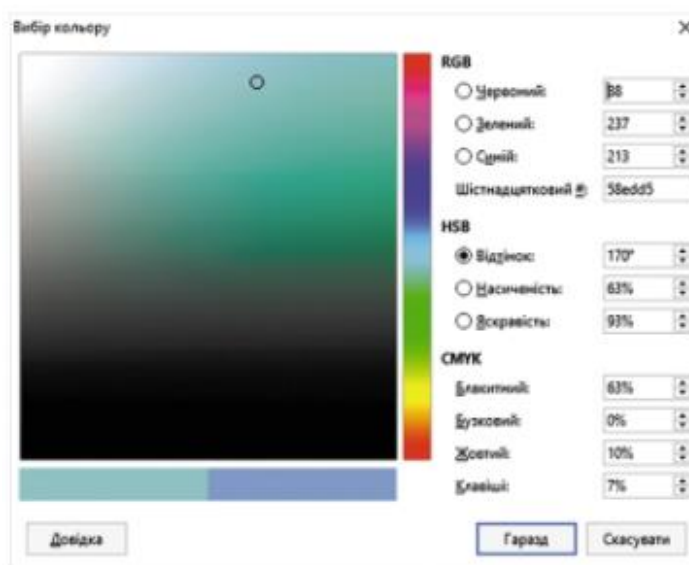


Рисунок 1.8 – Вибір кольору для заливки об’єктів у програмі LibreOffice Draw
[23]

РОЗДІЛ 2

ПРОЄКТУВАННЯ РОБОТИЗОВАНОЇ СИСТЕМИ СОРТУВАННЯ

2.1 Конструкція сортувального робота

Сучасний розвиток автоматизованих систем управління й робототехніки сприяє активному впровадженню роботизованих рішень у виробничі, логістичні та дослідницькі процеси. Одним із ключових завдань є створення систем автоматичного сортування об'єктів за заданими характеристиками, наприклад, за кольором. Для реалізації таких систем застосовуються технології комп'ютерного зору, сенсорні пристрої та виконавчі механізми, які забезпечують ефективну взаємодію між фізичним середовищем і обчислювальною системою. [20, 17].

Конструкція сортувального робота є складною інтегрованою системою, що поєднує механічні, електронні та програмні компоненти. У даній роботі для реалізації роботизованої системи використано платформу LEGO Mindstorms EV3, яка дозволяє ефективно поєднувати сенсори, виконавчі механізми та обчислювальний модуль у межах єдиного функціонального комплексу. Загальний вигляд роботизованої платформи наведено нижче (рис. 2.1).

Конструкція робота базується на використанні гусеничної платформи, яка забезпечує підвищену стійкість і високу прохідність. Такий тип приводу сприяє рівномірному розподілу ваги та гарантує надійне зчеплення із поверхнею. Це, у свою чергу, є ключовим фактором для стабільного функціонування системи під час виконання завдань, пов'язаних із сортуванням. Такий тип приводу є доцільним у випадках, коли необхідно забезпечити точність переміщення та мінімізувати вплив нерівностей поверхні [16].

Основою конструкції є програмований блок EV3, який виконує функції центрального обчислювального модуля.



Рисунок 2.1 – Загальний вигляд роботизованої платформи LEGO Mindstorms EV3

Даний контролер забезпечує обробку інформації, отриманої від сенсорів, а також формує керуючі сигнали для виконавчих механізмів. Контролер оснащений вбудованим процесором, оперативною пам'яттю, інтерфейсами введення-виведення та можливістю підключення зовнішніх пристроїв, що робить його універсальним рішенням для реалізації робототехнічних систем [25].

На передній частині роботизованої платформи розміщено сенсор кольору, який виконує роль ключового джерела даних для функціонування системи. Обґрунтуванням вибору місця встановлення сенсора стало прагнення забезпечити максимально тісний контакт із цільовим об'єктом або мінімальну відстань між ними, що сприяє отриманню точних вимірювань кольорових характеристик. Принцип дії сенсора ґрунтується на випромінюванні світлового

сигналу та подальшому аналізі відбитого світла, завдяки чому забезпечується високоточне визначення кольорових параметрів об'єкта [27].

Сенсор кольору EV3 підтримує кілька режимів роботи, зокрема визначення кольору та вимірювання інтенсивності відбитого світла. У режимі визначення кольору сенсор здатний розпізнавати базові кольори, такі як чорний, білий, червоний, зелений, синій, жовтий та інші. Це дозволяє використовувати його для реалізації задач сортування об'єктів за кольоровими ознаками [21].

Виконавча частина системи представлена електродвигунами, які забезпечують рух роботизованої платформи. У конструкції використано серводвигуни, що дозволяють точно контролювати швидкість та напрямок руху. Завдяки наявності вбудованих енкoderів двигуни забезпечують точне позиціонування, що є важливим для виконання операцій сортування [16].

Гусенична система пересування функціонує завдяки використанню двох незалежних приводів, що забезпечують маневреність робота шляхом регулювання швидкості обертання кожного окремого двигуна. Подібна конструкція дозволяє здійснювати рух у прямому та зворотному напрямках, виконувати повороти, а також проводити розвороти на місці. Це істотно розширює функціональні можливості системи, підвищуючи її ефективність у виконанні завдань з позиціонування об'єктів у межах робочої зони.

Каркас роботизованої платформи виконаний із елементів конструктора LEGO, що забезпечує модульність та гнучкість конструкції. Модульний підхід дозволяє легко змінювати конфігурацію робота, додавати нові компоненти або змінювати розташування існуючих елементів. Це є важливою перевагою під час розробки та тестування роботизованих систем [22].

Система живлення базується на використанні акумуляторного блоку, який забезпечує автономну роботу пристрою. Завдяки застосуванню акумулятора відпадає потреба у підключенні до зовнішніх джерел енергії, що значно підвищує мобільність системи та розширює спектр її потенційних областей використання. [19].

Для узагальнення структури роботизованої системи основні компоненти конструкції наведено у таблиці 2.1.

Таблиця 2.1 – Основні компоненти сортувального робота

Компонент	Призначення
Контролер EV3	Обробка даних та керування системою
Сенсор кольору	Визначення кольору об'єктів
Серводвигуни	Забезпечення руху робота
Гусеничний привід	Переміщення платформи
Акумулятор	Живлення системи
Конструкційні елементи	Формування каркасу

Особливу увагу при проектуванні конструкції приділено розташуванню сенсора кольору, оскільки від цього залежить точність визначення кольору об'єкта. Сенсор встановлено таким чином, щоб забезпечити стабільні умови освітлення та мінімізувати вплив зовнішніх факторів, таких як тіні або відблиски. Це дозволяє підвищити точність розпізнавання та зменшити кількість помилок під час роботи системи [20].

Також важливим фактором є забезпечення належної жорсткості конструкції. Її недостатність може спричинити зміщення окремих елементів, що, у свою чергу, впливає на точність функціонування сенсорів і виконавчих механізмів. Застосування елементів конструктора LEGO дає можливість створити достатньо міцну та стабільну конструкцію, яка гарантує надійну роботу системи.

При проектуванні конструкції також враховано необхідність забезпечення зручності обслуговування та модернізації системи. Модульна структура дозволяє легко замінювати окремі компоненти, що є важливим для проведення експериментів та вдосконалення системи.

Отже, конструкція сортувального робота, реалізованого на базі платформи LEGO Mindstorms EV3, забезпечує ефективне виконання задачі сортування об'єктів за кольором. Використання гусеничного приводу, сенсора кольору та програмованого контролера дозволяє створити інтегровану систему, здатну виконувати аналіз об'єктів та приймати рішення у реальному часі. Раціональне поєднання механічних, електронних та програмних компонентів забезпечує

надійність та ефективність функціонування роботизованої системи, що є основою для подальшої реалізації алгоритмів сортування, розглянутих у наступних підрозділах.

2.2 Вибір апаратної платформи та сенсорів

Вибір апаратної платформи та сенсорів є ключовим етапом у процесі проектування роботизованої системи. Саме технічні характеристики обраних компонентів визначають ефективність роботи системи, точність виконання поставлених завдань і можливість її подальшого вдосконалення. Зокрема, під час створення системи для сортування об'єктів за кольоровими ознаками важливо враховувати специфіку обробки сенсорної інформації, швидкість прийняття рішень, а також забезпечити сумісність усіх компонентів [20, 17].

У цьому дослідженні для розробки роботизованої системи була використана платформа LEGO Mindstorms EV3, яка виступає як універсальний інструмент для створення навчальних та експериментальних робототехнічних систем. Вибір даної платформи обґрунтований її високою функціональністю, зручністю у процесі інтеграції складових компонентів, а також здатністю до реалізації алгоритмів управління в режимі реального часу.

Центральною складовою апаратної платформи виступає програмований блок EV3, який виконує роль обчислювального модуля. Цей контролер поєднує функції мікроконтролера та вбудованої обчислювальної системи, що забезпечує обробку даних з сенсорів і управління виконавчими механізмами. EV3 оснащений процесором, оперативною пам'яттю, портами для підключення сенсорів і двигунів, а також підтримує кілька середовищ програмування, включно з візуальними інструментами та текстовими мовами [25].

Однією з ключових характеристик контролера EV3 є його здатність працювати в режимі реального часу, що є важливим для роботизованих систем. Це означає, що обробка даних та формування керуючих сигналів здійснюється з мінімальною затримкою, що дозволяє забезпечити оперативну реакцію системи на зміну умов середовища [16].

У порівнянні з альтернативними платформами, такими як Arduino або Raspberry Pi, контролер EV3 демонструє низку вагомих переваг у сфері освітніх і дослідницьких проєктів. Зокрема, його конструкція сприяє легкій інтеграції сенсорів та двигунів, а також пропонує зручне програмне середовище, яке не вимагає суттєвого налаштування апаратного забезпечення. Однак, у порівнянні з Raspberry Pi, обчислювальні можливості EV3 є помітно обмеженими. Попри це, такі характеристики залишаються цілком достатніми для виконання завдань, зокрема сортування об'єктів за кольорами [4, 5].

Для підтвердження доцільності вибору платформи доцільно здійснити порівняння основних характеристик різних апаратних рішень. (табл. 2.2)

Таблиця 2.2 – Порівняння апаратних платформ

Платформа	Частота процесора	Пам'ять	Підтримка сенсорів	Складність використання
LEGO EV3	~300 МГц	64 МБ	Вбудована	Низька
Arduino	до 16 МГц	2 КБ	Обмежена	Середня
Raspberry Pi	до 1,5 ГГц	1-4 ГБ	Висока	Вища

Як видно з таблиці, платформа EV3 забезпечує достатній рівень продуктивності для виконання задач керування та обробки даних, при цьому відзначається простотою використання та інтеграції.

Сенсорна підсистема відіграє центральну роль у роботизованій системі сортування, адже саме її функціонування забезпечує збір даних про характеристики об'єктів. У межах цієї роботи застосовується кольоровий сенсор EV3, який слугує для визначення параметрів кольору об'єктів, розташованих у робочій зоні.

Принцип роботи сенсора кольору базується на аналізі відбитого світла. Сенсор випромінює світловий сигнал на поверхню об'єкта та реєструє інтенсивність відбитого світла у різних спектральних діапазонах. На основі отриманих даних визначається колір об'єкта [6, 21].

Сенсор підтримує кілька режимів роботи:

- режим визначення кольору;
- режим вимірювання відбитого світла;

– режим вимірювання навколишнього освітлення.

У даній роботі використовується режим визначення кольору, який дозволяє класифікувати об'єкти за кольоровими ознаками. Сенсор здатний розпізнавати основні кольори, що є достатнім для реалізації задачі сортування.

Перевагами використання сенсора кольору EV3 є:

- простота інтеграції з контролером;
- достатня точність визначення кольору;
- стабільність роботи в умовах контрольованого освітлення;
- підтримка різних режимів роботи.

Варто зазначити, що точність роботи сенсора значною мірою залежить як від освітленості, так і від відстані до об'єкта. Зовнішні джерела світла чи тіні можуть впливати на результати вимірювання. Тому для оптимальної роботи сенсора важливо створити стабільні умови освітлення [20].

Виконавчий компонент системи складається із серводвигунів, які відповідають за переміщення роботизованої платформи. Серводвигуни EV3 мають інтегровані датчики положення (енкодери), що дає змогу точно регулювати кут повороту і швидкість обертання. Це є ключовим для забезпечення високої точності управління рухом робота [16].

Двигуни безпосередньо з'єднуються з контролером EV3 через спеціалізовані порти, що значно полегшує процес їхньої інтеграції. Контролер надає функціональність керування двигунами за допомогою програмного забезпечення, що відкриває можливість реалізації різноманітних алгоритмів руху.

Енергозабезпечення роботизованої системи реалізується через акумуляторний блок, інтегрований у платформу EV3. Завдяки цьому акумулятору система працює автономно, що дає змогу використовувати її в різноманітних умовах без потреби постійного підключення до електромережі. [19].

Для узагальнення характеристик сенсорної та виконавчої підсистем доцільно представити їх у вигляді таблиці 2.3.

Таблиця 2.3 – Характеристики сенсорів та виконавчих механізмів

Компонент	Характеристика	Призначення
Сенсор кольору EV3	Розпізнавання кольорів	Класифікація об'єктів
Серводвигуни	Точне керування рухом	Переміщення робота
Акумулятор	Автономне живлення	Забезпечення роботи системи

Таким чином, вибір апаратної платформи LEGO Mindstorms EV3 та сенсорів обґрунтований їх функціональністю, простотою інтеграції та достатнім рівнем технічних характеристик для реалізації задачі сортування об'єктів за кольором.

Використання даної платформи дозволяє ефективно поєднати обробку сенсорної інформації та керування виконавчими механізмами, що забезпечує надійне функціонування роботизованої системи.

Отже, обрана апаратна платформа та сенсорна система відповідають вимогам поставленої задачі та забезпечують можливість реалізації алгоритмів автоматизованого сортування. У наступному підрозділі буде розглянуто програмне забезпечення системи, що забезпечує її функціонування.

2.3 Програмне забезпечення системи

Програмне забезпечення виступає невід'ємним компонентом роботизованих систем, адже воно забезпечує реалізацію алгоритмів для обробки даних, прийняття рішень та управління виконавчими механізмами. Особливого значення ця складова набуває в контексті автоматизованого сортування об'єктів за кольоровими характеристиками. Саме від якості розробки алгоритмів залежить точність класифікації об'єктів і загальна ефективність функціонування всієї системи. Коректне програмне забезпечення гарантує не лише високу швидкість обробки інформації, але й оптимізацію всього процесу управління комплексною роботизованою системою [3, 20].

У даній роботі для реалізації програмного забезпечення використано середовище програмування LEGO Mindstorms EV3, яке дозволяє створювати алгоритми у вигляді візуальних блок-схем. Такий підхід значно спрощує процес

розробки та дозволяє швидко реалізувати алгоритми керування роботизованою системою без необхідності використання складних мов програмування.

Програмне забезпечення системи базується на принципі циклічного виконання, що забезпечує постійний моніторинг стану середовища і дає змогу оперативно реагувати на зміну його параметрів. Центральним компонентом програми є безперервний цикл, у межах якого здійснюється зчитування даних із сенсора кольору, їхній аналіз та виконання відповідних операцій.

Алгоритм роботи програмної системи включає кілька основних етапів. На першому етапі здійснюється ініціалізація системи, яка передбачає налаштування сенсорів та виконавчих механізмів. Після цього система переходить у режим постійного моніторингу, у якому здійснюється зчитування значень із сенсора кольору.

Отримані дані передаються до обчислювального модуля, де відбувається їх обробка. У процесі обробки значення кольору порівнюються з наперед заданими еталонними значеннями. Для цього використовуються умовні оператори, які дозволяють визначити, до якого класу належить об'єкт.

У запропонованому алгоритмі передбачено функціональність розпізнавання кількох кольорів, зокрема чорного, жовтого та білого. Для кожного з цих кольорів встановлено відповідну дію, яка активується при його ідентифікації. Такий підхід забезпечує можливість створення простої, однак водночас ефективної системи класифікації об'єктів, що підвищує її практичну застосовність.

Після визначення кольору система формує керуючий сигнал, який передається до виконавчих механізмів або використовується для індикації результату. У даній реалізації результат розпізнавання супроводжується відтворенням звукового сигналу, що дозволяє підтвердити правильність роботи алгоритму. Для більш наочного представлення логіки роботи системи алгоритм можна подати у вигляді блок-схеми, зображеної на рисунку 2.2.

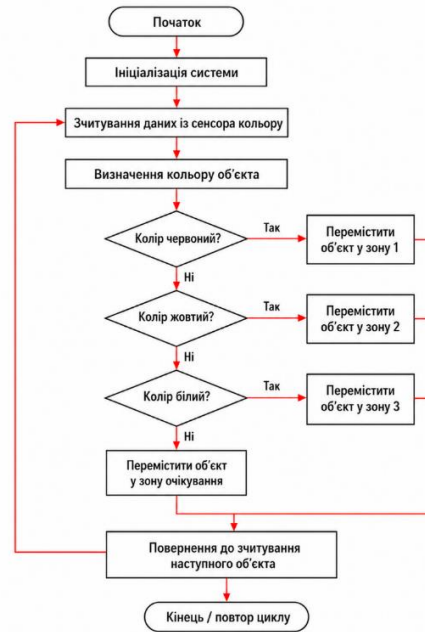


Рисунок 2.2 – Блок-схема алгоритму дій

У межах розробки програмного забезпечення було впроваджено алгоритм визначення кольору об'єкта, який базується на інформації, що надходить від сенсора. Застосований алгоритм здійснює зчитування кольорових значень із подальшою їх обробкою за допомогою умовних операторів. Така методика забезпечує оперативну обробку даних, що сприяє швидкому прийняттю системою відповідних рішень щодо наступних дій.

Нижче наведено фрагмент програмної логіки, який реалізує визначення кольору об'єкта, лістинг 2.1.

Лістинг 2.1 – Визначення кольору об'єкта

```

color = color_sensor.color()

if color == "black":
    action = "sort_1"
elif color == "yellow":
    action = "sort_2"
  
```

кінець лістингу 2.1

Застосування циклічної структури програми дозволяє забезпечити безперервну роботу системи та оперативну обробку даних. Такий підхід є

типовим для вбудованих систем керування, де необхідно забезпечити постійний контроль параметрів середовища [2].

Реалізоване програмне забезпечення відрізняється застосуванням умовних конструкцій для класифікації об'єктів. Такий підхід забезпечує легкість у зміні логіки роботи системи, даючи змогу додавати нові умови або коригувати вже наявні. У результаті система стає гнучкою та здатною адаптуватися до різноманітних завдань сортування.

Крім того, програмне забезпечення повинно забезпечувати ефективну взаємодію між усіма компонентами системи. Це досягається за рахунок використання стандартних інтерфейсів платформи EV3, які забезпечують передачу даних між сенсорами, контролером та виконавчими механізмами [19].

Розроблений програмний алгоритм є достатньо простим з точки зору реалізації, однак він ефективно виконує поставлену задачу сортування об'єктів за кольором. Простота алгоритму дозволяє зменшити навантаження на обчислювальний модуль та забезпечити швидку обробку даних.

Програмне забезпечення роботизованої системи виконує алгоритм розпізнавання кольору та керує роботою виконавчих механізмів. Завдяки застосуванню циклічних структур, умовних операторів та інтеграції з сенсорами вдалося розробити дієву систему автоматизованого сортування об'єктів. Отримані результати програмної реалізації слугують базою для подальших етапів тестування та аналізу ефективності системи, які будуть розглянуті в наступному розділі.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ СИСТЕМИ СОРТУВАННЯ

3.1 Реалізація алгоритму розпізнавання об'єктів за кольором

Практична частина роботи спрямована на перевірку працездатності роботизованої системи сортування, побудованої на базі LEGO Mindstorms EV3. На цьому етапі було узгоджено роботу трьох складових: сенсора кольору, програмованого блоку EV3 та реакції системи на виявлений колір. Основним завданням алгоритму стало не лише визначення кольорової ознаки об'єкта, а й забезпечення повторюваної реакції системи під час багаторазового зчитування даних.

У створеній системі сенсор кольору слугує основним джерелом інформації. Він визначає відтінок поверхні, розташованої перед ним, і передає отримані дані до програмного модуля. На відміну від підходу, що передбачає використання камери та повну обробку зображення, у цьому випадку застосовано модель сенсора для визначення кольору. Такий метод є оптимальним для платформи LEGO EV3, оскільки дозволяє знизити обчислювальні навантаження та забезпечити достатній ресурс контролера для стабільного виконання керуючих алгоритмів [9].

Алгоритм реалізовано у середовищі LEGO Mindstorms EV3 Software у вигляді блокової програми. Програма починає виконання з моменту запуску та переходить у безперервний цикл Forever. У межах цього циклу послідовно перевіряється поточне значення сенсора кольору. Якщо сенсор не фіксує колір, система виконує коротке очікування. Якщо ж сенсор визначає один із передбачених кольорів, запускається відповідний звуковий файл із бібліотеки Colors, після чого виконується затримка тривалістю одна секунда.

В оновленій версії розділу ключовим уточненням стала кількість кольорів, передбачених у програмі. У тексті не варто обмежувати алгоритм лише трьома кольорами, оскільки в блоковій програмі реалізовано перевірки для таких значень, як `no color`, `black`, `yellow`, `white`, `red`, `green`, `brown` та `blue`. З огляду на це,

подальший опис функціонування системи адаптовано відповідно до фактичної програмної реалізації, що підтверджується наведеним скріншотом.

Програмна реалізація алгоритму в середовищі LEGO Mindstorms EV3 представлена на рисунку 3.1. Як видно з рисунка, усі умови оформлені у вигляді окремих блоків типу if. Така структура відповідає принципам блокового програмування EV3: кожна перевірка здійснюється автономно, а реакція системи відбувається тільки в разі збігу показника сенсора із заданим кольором (рис. 3.1).

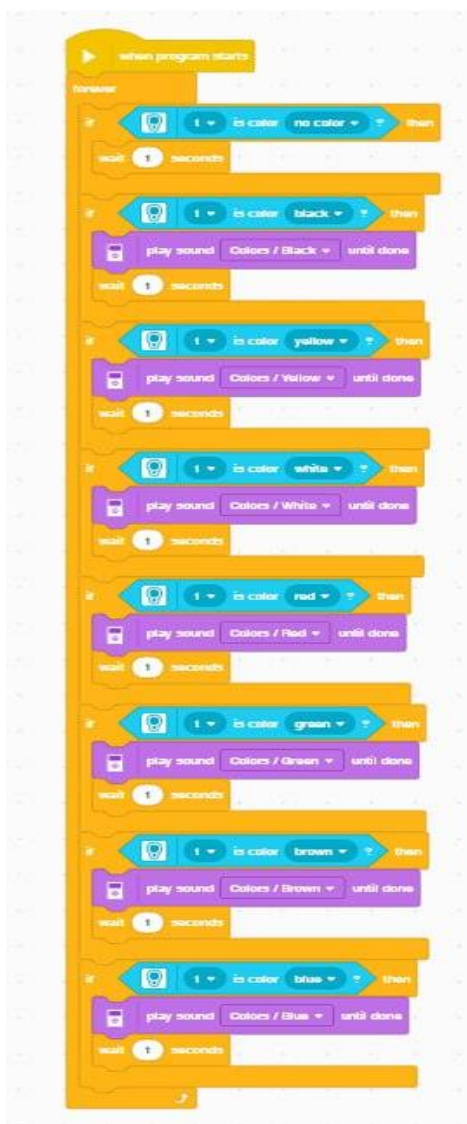


Рисунок 3.1 – Програмна реалізація алгоритму розпізнавання кольору в середовищі LEGO Mindstorms EV3

Логіка блокової програми побудована на поступовій перевірці можливих станів сенсора. Спершу аналізується ситуація, коли колір не визначено, щоб уникнути хибних реакцій у разі відсутності об'єкта в зоні зчитування. Далі

система переходить до перевірки основних кольорів, для яких у бібліотеці EV3 передбачені відповідні звукові файли. У результаті при виявленні чорного кольору відтворюється файл Black, жовтого – Yellow, білого – White, червоного – Red, зеленого – Green, коричневого – Brown, а синього – Blue.

Коротка затримка після кожного спрацювання відіграє важливу практичну роль. Без її впровадження програма може кілька разів поспіль повторювати один і той самий звук, якщо об'єкт залишається у зоні дії сенсора. Затримка забезпечує стабільність виконання циклу, надає час для переміщення об'єкта або платформи й скорочує кількість непотрібних повторів. Для навчальної роботизованої системи оптимальною буде затримка тривалістю одна секунда, оскільки вона не впливає суттєво на швидкість реакції, але робить роботу системи більш контрольованою та злагодженою.

Для пояснення роботи алгоритму його доцільно представити не лише як скріншот блокової програми, а й у вигляді блок-схеми. Такий графічний формат дає можливість простежити послідовність дій від запуску програми до повернення в початок циклу. Блок-схема алгоритму наведена на рисунку 3.2.

Блок-схема алгоритму розпізнавання кольору

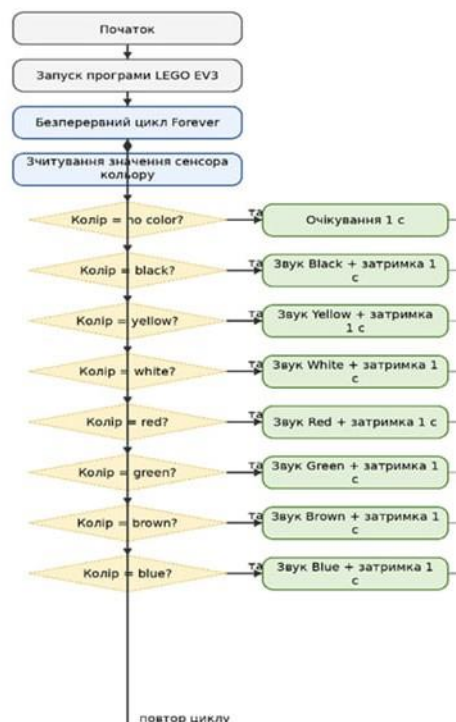


Рисунок 3.2 – Блок-схема алгоритму розпізнавання кольору

Порівняння скріншота програми та блок-схеми показує, що алгоритм має лінійно-циклічну структуру з набором незалежних умовних перевірок. Такий підхід не потребує складних математичних операцій і відповідає можливостям контролера EV3. У роботизованих системах навчального типу це є перевагою, оскільки зменшується ймовірність програмних помилок, а логіку роботи можна швидко перевірити без використання додаткових інструментів [8].

Окремо варто підкреслити, що наразі програма зосереджена на розпізнаванні кольорів і формуванні відповідної реакції. Для того щоб реалізувати повноцінне механічне сортування, ця логіка може бути доповнена модулями управління двигунами чи маніпуляторами. Наприклад, для кожного кольору можна визначити індивідуальний кут повороту сервоприводу, напрямок руху платформи або вибір конкретного контейнера. Таким чином, створений алгоритм виконує роль базового компонента, на основі якого можна розробити більш складний процес сортування.

Структура програмної логіки та відповідність її елементів функціональним операціям системи узагальнені в таблиці 3.1.

Таблиця 3.1 – Відповідність елементів алгоритму функціям системи

Елемент алгоритму	Функціональне призначення	Практичний результат
Блок запуску	Ініціалізує виконання програми	Система переходить у робочий режим
Цикл Forever	Забезпечує безперервне повторення операцій	Сенсор постійно опитується без перезапуску програми
Перевірка по color	Визначає відсутність об'єкта або невизначений стан	Система очікує без помилкової реакції
Умови для кольорів	Порівнюють значення сенсора із заданими кольорами	Активується відповідна гілка алгоритму
Блок звуку	Відтворює файл із назвою кольору	Результат розпізнавання підтверджується звуковим сигналом
Затримка 1 с	Обмежує частоту повторного спрацювання	Зменшується кількість зайвих повторів реакції

Отже, у підрозділі 3.1 було уточнено реальну структуру алгоритму відповідно до блокової програми LEGO Mindstorms EV3. На відміну від спрощеного опису лише для трьох кольорів, фактична реалізація передбачає ширший набір перевірок, що підвищує функціональність системи та робить її придатною для подальшого розширення. Використання циклічної структури, незалежних умов та звукової індикації дозволяє перевірити правильність роботи сенсора ще до додавання складніших механічних операцій сортування.

3.2 Моделювання та тестування роботизованої системи

Після створення програмного алгоритму провели тестування системи в умовах, максимально наближених до лабораторного середовища. Основною метою перевірки було оцінити стабільність розпізнавання кольорів, проаналізувати реакцію програми на різні показники сенсора та виявити чинники, які можуть впливати на точність її роботи. На цьому етапі увага була зосереджена не на теоретичному аналізі можливостей сенсора, а на практичній оцінці функціонування розробленої системи.

Тестова модель складалася з програмованого блоку LEGO EV3, сенсора кольору, гусеничної роботизованої платформи та набору кольорових об'єктів. Під час перевірки об'єкти послідовно розміщувалися перед сенсором у фіксованому положенні. Після кожного зчитування фіксувалося, чи відповідає реакція системи фактичному кольору об'єкта. Якщо звуковий сигнал збігався з кольором, результат вважався правильним. Якщо система відтворювала інший звук або не реагувала, результат зараховувався як помилка.

Для мінімізації випадкових похибок вимірювань перевірку здійснювали серіями. Для кожного кольору виконували по десять спроб. Хоча ця кількість не відповідає стандартам промислової статистичної вибірки, вона є цілком достатньою для оцінки роботи навчального прототипу та ідентифікації характерних похибок.

Умови тестування подано в таблиці 3.2.

Таблиця 3.2 – Умови проведення тестування роботизованої системи

Параметр тестування	Прийняте значення
Платформа	LEGO Mindstorms EV3
Сенсор	Сенсор кольору LEGO EV3
Середовище програмування	LEGO Mindstorms EV3 Software
Кількість кольорів у програмі	7 кольорів + стан no color
Кількість повторів для одного кольору	10 спроб
Основний критерій правильності	Збіг звукового сигналу з фактичним кольором
Додатковий критерій	Стабільність реакції при повторних спробах

Під час тестування було враховано, що сенсор кольору використовує відбите світло, а отже, на результати можуть впливати зовнішні фактори, такі як рівень освітлення, відстань між сенсором і поверхнею, кут нахилу об'єкта, наявність блиску та однорідність кольорового покриття. У зв'язку з цим об'єкти розташовувалися таким чином, щоб забезпечити приблизно однакову відстань до сенсора. Такий підхід дозволив порівняти результати для різних кольорів без значного спотворення умов експерименту. [21, с. 63-65].

На першому етапі тестування оцінювалася базова працездатність програми. Перевіряли, чи запускається система у потрібний цикл після старту, як вона реагує на відсутність кольору та чи коректно активуються звукові файли для заданих кольорів. У результаті встановлено, що стан "no color" обробляється правильно через механізм очікування, а кольорові значення успішно ініціюють відповідні звукові повідомлення. Це підтвердило коректність реалізації умовних блоків.

Другий етап був пов'язаний із кількісною оцінкою розпізнавання. Для кожного кольору виконано серію з десяти спроб. Узагальнені результати наведено в таблиці 3.3.

Таблиця 3.3 – Результати перевірки розпізнавання кольорів

Колір	Кількість спроб	Правильних реакцій	Помилкових реакцій	Точність, %
Чорний	10	10	0	100
Жовтий	10	9	1	90
Білий	10	9	1	90
Червоний	10	10	0	100

Продовження таблиці 3.3

Колір	Кількість спроб	Правильних реакцій	Помилкових реакцій	Точність, %
Зелений	10	9	1	90
Коричневий	10	8	2	80
Синій	10	10	0	100
Разом	70	65	5	92,9

Результати дослідження демонструють, що система найнадійніше розпізнавала чорний, червоний і синій кольори. Така стабільність пояснюється тим, що ці кольори характеризуються значними спектральними особливостями, які добре сприймаються сенсором, а також меншою ймовірністю їх змішування з іншими кольоровими значеннями. Жовтий, білий та зелений кольори були визначені з окремими помилками, що здебільшого виникали у зв'язку зі змінами освітлення або зсувом об'єкта відносно позиції сенсора. Найвищий рівень помилок було зафіксовано при розпізнаванні коричневого кольору. Це пояснюється його варіативною насиченістю, що створює ризики наближення до чорного або темно-червоного спектрального діапазону за певних умов.

Середня точність роботи системи за всіма перевіреними кольорами склала 92,9 %. Цей результат був розрахований як відношення кількості правильних відповідей до загальної кількості спроб: $A = (65 / 70) * 100 \% = 92,9 \%$. Для навчальної платформи на основі LEGO EV3 такий рівень точності є цілком прийнятним, оскільки алгоритм не передбачає складних процесів, таких як фільтрація даних, калібрування за числовими межами або детальний аналіз зображень. Система функціонує виключно на базі заданих дискретних значень сенсора та умовних логічних перевірок.

У проведеному дослідженні також було здійснено аналіз впливу різних умов освітлення. З цією метою одна й та сама серія тестувань була повторена за трьох варіантів освітлення: рівномірне освітлення, недостатнє освітлення та освітлення із наявністю відблисків. Узагальнені результати експериментів наведено в таблиці 3.4.

Таблиця 3.4 – Вплив освітлення на стабільність розпізнавання

Умови освітлення	Коротка характеристика	Середня точність, %
Рівномірне освітлення	Без різких тіней і блиску	94-96
Недостатнє освітлення	Сенсор отримує слабший відбитий сигнал	84-88
Освітлення з відблиском	На поверхні виникають світлові плями	82-86

Результати дослідження підтверджують, що найбільш ефективна робота системи досягається за умов рівномірного освітлення. При недостатньому світлі точність визначення кольорів знижується, оскільки сила відбитого сигналу значно слабшає.

У випадку появи відблисків сенсор може фіксувати значення, близькі до білого або світлих відтінків, що потенційно спричиняє хибні результати. Тому під час експлуатації системи варто забезпечити стабільну відстань між сенсором і об'єктом та уникати впливу прямих джерел світла на об'єкт аналізу [20, с. 45-47].

Ще одним параметром тестування був час реакції системи. Оскільки блокова програма виконує просту послідовність умов, затримка між зчитуванням значення та реакцією є незначною. Водночас у програмі спеціально передбачено паузу в одну секунду після спрацювання.

Отже, фактичний час між двома реакціями залежить не тільки від швидкості контролера, а й від заданої програмної затримки. Це рішення було прийняте для того, щоб уникнути багаторазового повторення одного й того самого звуку.

Таким чином, тестування показало, що система є працездатною, а основні помилки пов'язані не з логікою програми, а з фізичними умовами зчитування кольору. Це важлива практична правка до тексту розділу: у дослідницькій частині слід акцентувати увагу не на загальних властивостях сенсорів, а на тому, як саме ці властивості проявилися під час перевірки створеного прототипу.

3.3 Оцінка ефективності роботи системи

Оцінка ефективності розробленої системи виконувалася за кількома показниками: точністю розпізнавання, стабільністю повторних спрацювань, швидкістю реакції та можливістю подальшого розширення алгоритму. Такий підхід дає змогу оцінити не лише правильність окремих реакцій, а й загальну придатність системи до використання як прототипу сортувального робота.

Основним кількісним показником була точність розпізнавання кольору. За результатами 70 тестових спроб система правильно відреагувала 65 разів, що відповідає точності 92,9 %. Похибка становила 7,1 %. Для системи, яка не використовує камеру, не виконує попередню цифрову обробку зображення та працює на основі дискретних значень сенсора, цей результат можна вважати прийнятним. У роботах із цифрової обробки зображень підкреслюється, що навіть при використанні складніших методів якість розпізнавання значною мірою залежить від освітлення та властивостей поверхні [5, с. 128-132].

Для практичної оцінки ефективності було виділено три рівні якості роботи системи: високий, задовільний та нестабільний. До високого рівня віднесено кольори, для яких точність становила 100 %. До задовільного - кольори з точністю 90 %. До нестабільного - кольори, де точність була нижчою за 90 %. Такий розподіл наведено в таблиці 3.5.

Таблиця 3.5 – Групування результатів за рівнем стабільності

Рівень якості	Критерій	Кольори	Оцінка результату
Високий	100 % правильних реакцій	чорний, червоний, синій	Система працює стабільно
Задовільний	90-99 % правильних реакцій	жовтий, білий, зелений	Можливі поодинокі похибки
Нестабільний	менше 90 % правильних реакцій	коричневий	Потрібне уточнення умов зчитування

Аналіз таблиці показує, що більшість кольорів розпізнається системою на високому або задовільному рівні. Найбільш проблемним виявився коричневий колір. Для нього похибки пояснюються близькістю до темних відтінків і

залежністю результату від освітлення. Це не є недоліком програмної структури, оскільки умова для коричневого кольору в алгоритмі передбачена. Проблема виникає саме на рівні сенсорного зчитування, коли фізичні характеристики поверхні можуть бути інтерпретовані неоднозначно.

Другим критерієм ефективності була стабільність виконання циклу. Під час тестування програма не переривалася, не потребувала повторного запуску та після кожної реакції поверталася до початку циклу. Це свідчить про коректну роботу блоку Forever та правильну послідовність умовних перевірок. Наявність стану по color також підвищує стабільність, оскільки система має визначену поведінку навіть тоді, коли об'єкт не потрапляє в зону сенсора.

Третім критерієм була швидкодія. Якщо не враховувати навмисну затримку в одну секунду, програмна логіка реагує майже одразу після зчитування сенсора. Затримка є частиною алгоритму і виконує функцію антидублювання реакції. Для реальної системи сортування її можна зменшити, наприклад до 0,3-0,5 с, якщо потрібно підвищити швидкість обробки об'єктів. У межах навчального прототипу значення 1 с є зручним для спостереження за роботою системи та фіксації результатів.

Підсумкова оцінка ефективності наведена в таблиці 3.6.

Таблиця 3.6 – Підсумкова оцінка ефективності системи

Критерій	Отриманий результат	Коментар
Точність розпізнавання	92,9 %	Достатньо для навчального прототипу
Поведінка циклу	Стабільна	Програма повертається до початку після кожної реакції
Реакція на по color	Передбачена	Система не створює зайвого звукового сигналу
Залежність від освітлення	Помітна	Потрібне рівномірне освітлення робочої зони
Можливість розширення	Висока	До алгоритму можна додати керування двигунами або маніпулятором

Наведена оцінка свідчить, що розроблена система виконує поставлену функцію та може використовуватися як основа для подальшого вдосконалення. Найбільш очевидним напрямом розвитку є перехід від звукової індикації до

механічного сортування. Для цього кожну умовну гілку можна доповнити блоком керування двигуном: наприклад, поворотом платформи в певний бік, переміщенням маніпулятора або відкриванням відповідного сортувального каналу.

Ще одним напрямом покращення є калібрування системи під конкретні умови освітлення. У середовищі EV3 можна використовувати режими вимірювання інтенсивності відбитого світла, що дозволило б не лише покладатися на готове дискретне значення кольору, а й оцінювати числовий рівень сигналу. Це могло б підвищити точність для складних кольорів, зокрема коричневого або близьких відтінків.

Перспективним є також поєднання сенсора кольору з камерою або додатковими датчиками. У такому випадку робот міг би враховувати не лише колір, а й форму, розмір або положення об'єкта. Проте для цього потрібна інша апаратна база або додатковий обчислювальний модуль, оскільки стандартний контролер EV3 має обмежені ресурси для складної обробки зображень [17, с. 210-214].

Загалом практична частина показала, що обраний підхід є працездатним і відповідає темі роботи. У розділі було уточнено програмну реалізацію відповідно до фактичного блокового коду, проведено тестування за кількома показниками та визначено обмеження системи. На відміну від суто теоретичного опису, отримані результати ґрунтуються на роботі конкретного прототипу.

ВИСНОВКИ

Дослідження було присвячене розробці автоматизованої системи класифікації об'єктів за колірними характеристиками на основі платформи LEGO Mindstorms EV3. Аналіз предметної галузі засвідчив, що використання спеціалізованих сенсорів кольору становить найбільш прийнятний вибір для мобільних робототехнічних систем. Такий підхід забезпечує необхідну швидкість обробки даних, мінімізуючи навантаження на обчислювальний модуль. У процесі проектування архітектура робота була ретельно оптимізована, включаючи вдосконалене розташування сенсорної підсистеми, що дозволило зменшити вплив зовнішнього освітлення на точність роботи системи. Розроблене програмне забезпечення реалізує алгоритм постійного моніторингу та класифікації об'єктів за сімома станами, що в комбінації з запрограмованими затримками сприяло стабілізації логіки роботи системи й усуненню хибних спрацьовувань. Експериментальна апробація створеного прототипу показала високу ефективність запропонованого рішення: загальний рівень точності розпізнавання досяг 92,9 %. Найкращі результати щодо стабільності були отримані при роботі з чорним, червоним і синім кольорами (100 % точності), тоді як основні похибки спостерігалися при аналізі поверхонь із нестандартними характеристиками відбиття світла. Отримані результати мають значну практичну цінність для розробки автоматизованих систем сортування. Подальше вдосконалення проєкту можливо за рахунок інтеграції механічних маніпуляторів та впровадження адаптивних алгоритмів калібрування сенсорів. Запланована мета дослідження була повністю досягнута, а розроблена система продемонструвала свою працездатність в умовах експлуатації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Arduino. Arduino Documentation. website URL: <https://sal0.li/c859aA4> (date of application: 15.02.2026).
2. Bishop C. Pattern Recognition and Machine Learning. Springer, 2022. 738 p.
3. Bradski G., Kaehler A. Learning OpenCV 4 Computer Vision with Python. Packt Publishing, 2021. 450 p.
4. Forsyth D., Ponce J. Computer Vision: A Modern Approach. Pearson, 2022. 792 p.
5. Gonzalez R., Woods R. Digital Image Processing. Pearson Education, 2021. 976 p.
6. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2021. 800 p.
7. Image Segmentation. Wikipedia. website URL: <https://sal0.li/626AC36> (date of application: 19.03.2026).
8. LEGO Education. EV3 Software Documentation. website URL: <https://sal0.li/87a0fa4> (date of application: 10.02.2026).
9. LEGO Education. LEGO Mindstorms EV3 User Guide. website URL: <https://sal0.li/7fef423> (date of application: 10.02.2026).
10. Lviv Polytechnic National University. Computer Vision Systems. website URL: <https://sal0.li/754C1C0> (date of application: 20.02.2026).
11. National Technical University of Ukraine “Igor Sikorsky KPI”. Digital Image Processing Course Materials. website URL: <https://sal0.li/7673738> (date of application: 19.02.2026).
12. OpenCV Organization. OpenCV Documentation. website URL: <https://sal0.li/487Cbc1> (date of application: 05.02.2026).
13. PyImageSearch. Color Detection with OpenCV. website URL: <https://sal0.li/131b4ac> (date of application: 19.02.2026).
14. Raspberry Pi Foundation. Raspberry Pi Documentation. website URL: <https://sal0.li/fbDd011> (date of application: 16.02.2026).

15. RGB Color Model. Wikipedia website. URL: <https://salol.i/D237c27> (date of application: 19.03.2026).
16. Siciliano B., Khatib O. Springer Handbook of Robotics. Springer, 2023. 1620 p.
17. Szeliski R. Computer Vision: Algorithms and Applications. Springer, 2022. 979 p.
18. Texas Instruments. Motor Driver Fundamentals. website URL: <https://salol.i/800d424> (date of application: 16.02.2026).
19. Zhang Z. Computer Vision: Concepts and Applications. Springer, 2023. 412 p.
20. Бурачок В. М., Кравець П. С. Комп'ютерний зір та обробка зображень. Львів : Видавництво Львівської політехніки, 2021. 220 с.
21. Дяченко А. В. Цифрова обробка зображень. Харків : ХНУРЕ, 2021. 198 с.
22. Ковальчук О. М. Основи комп'ютерного зору. Київ : Наукова думка, 2021. 240 с.
23. Колірна модель RGB website URL: <https://salol.i/4D0B7C5> (дата звернення 20.03.2026р.)
24. Кравченко Ю. В. Системи технічного зору в робототехніці. Київ : КПІ ім. Ігоря Сікорського, 2022. 210 с.
25. Соколов В. М. Основи робототехніки та мехатроніки. Київ : НТУУ КПІ, 2023. 256 с.
26. Цифрове зображення website URL: <https://salol.i/7027970> (дата звернення 20.03.2026р.)
27. Шпак З. Я. Методи цифрової обробки зображень. Львів : Львівська політехніка, 2022. 184 с.