

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Навчально-науковий інститут новітніх освітніх технологій
Кафедра інформаційно-обчислювальних систем і управління

СОЗАНСЬКА ТЕТЯНА МИХАЙЛІВНА

**МОДУЛЬ КЛАСИФІКАЦІЇ ВІДХОДІВ НА ОСНОВІ ГЛИБОКОГО
НАВЧАННЯ / Module for waste classification based on deep learning**

Спеціальність: 122 - Комп'ютерні науки
Освітньо-професійна програма – Комп'ютерні науки

Кваліфікаційна робота

Виконав студент групи КН-42
Т.М. Созанська

Науковий керівник
к.т.н., професор Х.В. Ліп'яніна-
Гончаренко

Кваліфікаційну роботу допущено до
захисту

«__»_____2025р.

В.о. завідувач кафедри

_____Н.М. Васильків

Тернопіль-2025

Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління
Освітній ступінь «бакалавр»
спеціальність 122 - Комп'ютерні науки
освітньо-професійна програма - Комп'ютерні науки

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Н.М. Васильків
« ____ » _____ 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
СОЗАНСЬКА Тетяна Михайлівна
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: Модуль класифікації відходів на основі глибокого навчання / Module for waste classification based on deep learning
керівник роботи Ліп'яніна-Гончаренко Христина Володимирівна д.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від 20 грудня 2024 р. № 938.
2. Строк подання студентом закінченої кваліфікаційної роботи 25 травня 2025 р.
3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.
4. Основні питання, які потрібно розробити:
 - проаналізувати теоретичні основи управління відходами та сучасні методи автоматизації їх сортування;
 - дослідити та систематизувати основні методи аналізу даних, що застосовуються для класифікації відходів;
 - розробити загальну архітектуру системи автоматизованої класифікації та сортування відходів з описом алгоритмічного та інформаційного забезпечення;
 - реалізувати програмний модуль для класифікації відходів із розробкою інтерфейсу користувача;
 - виконати тестування програмного забезпечення та оцінити ефективність запропонованої системи.
5. Перелік графічного матеріалу в роботі:
 - схема модуля класифікації відходів;
 - схема алгоритму класифікації зображення;
 - схема алгоритму модуля класифікації зображення;
 - матриця помилок для класифікації типів сміття.

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 20 грудня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1	Затвердження теми кваліфікаційної роботи, ознайомлення з літературними джерелами та складання плану роботи	до 01.01. 2025 р.	
2	Написання 1 розділу кваліфікаційної роботи	до 01.02. 2025 р.	
3	Написання 2 розділу кваліфікаційної роботи	до 01.04.2025 р.	
4	Написання 3 розділу кваліфікаційної роботи	до 01.05. 2025 р.	
5	Представлення попереднього варіанту кваліфікаційної роботи, перевірка та внесення змін керівником	до 15.05.2025 р.	
6	Опрацювання зауважень та представлення завершеного варіанту кваліфікаційної роботи. Підготовка супроводжуючих документів	до 25.05.2025 р.	
7	Перевірка кваліфікаційної роботи на оригінальність тексту	до 30.05.2025 р.	
8	Оформлення кваліфікаційної роботи та отримання допуску до захисту	до 10.06.2025 р.	
9	Подання кваліфікаційної роботи до захисту на засіданні атестаційної комісії	до 10.06. 2025 р.	

Студент _____ Созанська Т. М.
(підпис) (прізвище та ініціали)

Керівник кваліфікаційної роботи _____ Лип'яніна-Гончаренко Х. В.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота на тему «Модуль класифікації відходів на основі глибокого навчання» на здобуття освітнього ступеня «бакалавр» зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Комп'ютерні науки» написана обсягом у 59 сторінок і містить 18 ілюстрацій, 3 додатки та 30 використаних джерел.

Метою роботи є створення модуля для класифікації відходів у режимі реального часу на основі методів глибокого навчання, що забезпечить автоматичне розпізнавання типів сміття (пластик, скло, метал, папір, упаковка) з використанням камер та технологій комп'ютерного зору.

Методами дослідження обрано: аналіз сучасних підходів до класифікації зображень, моделювання нейронної мережі MobileNetV2 із застосуванням трансферного навчання, попередню обробку даних, методи тестування продуктивності та точності.

У ході роботи розроблено модуль з графічним інтерфейсом, який здійснює класифікацію об'єктів у реальному часі з високою точністю та візуальним відображенням результатів. Проведено тестування функціональності та зручності взаємодії з користувачем.

Результати можуть бути використані для подальшого вдосконалення систем автоматизованого сортування відходів у побутовій та промисловій сферах, інтеграції зі смарт-платформами та геоінформаційними системами.

Ключові слова: ГЛИБОКЕ НАВЧАННЯ, НЕЙРОННА МЕРЕЖА, РОЗПІЗНАВАННЯ ОБ'ЄКТІВ, СОРТУВАННЯ СМІТТЯ, КОМП'ЮТЕРНИЙ ЗІР, АВТОМАТИЧНА КЛАСИФІКАЦІЯ.

ANNOTATION

Qualification work on the topic “Waste classification module based on deep learning” for the bachelor's degree in specialty 122 “Computer Science” of the educational program “Computer Science” is written in 59 pages and contains 18 illustrations, 3 appendices and 30 references.

The purpose of the work is to create a real-time waste classification module based on deep learning methods that will provide automatic recognition of waste types (plastic, glass, metal, paper, packaging) using cameras and computer vision technologies.

Research methods used: analysis of modern approaches to image classification, modeling of the MobileNetV2 neural network using the transfer learning method, data pre-processing, performance and accuracy testing methods.

In the course of the work, a module with a graphical interface was developed that classifies objects in real time with high accuracy and visual display of the results. Functionality and user experience testing was conducted.

The results can be used to further improve automated waste sorting systems in the domestic and industrial sectors, as well as for integration with smart platforms and geographic information systems.

Keywords: DEEP LEARNING, NEURAL NETWORK, OBJECT RECOGNITION, WASTE SORTING, COMPUTER VISION, AUTOMATIC CLASSIFICATION.

ЗМІСТ

Вступ.....	7
1 Теоретичні засади для аналізу проблематики управління відходами	9
1.1 Теоретичні аспекти управління відходами	9
1.2 Основні методи аналізу даних у класифікації відходів	11
1.3 Аналіз існуючих рішень у сфері автоматизації сортування відходів.....	13
1.4 Постановка задачі.....	15
2 Технології та інструментальні розробки пропонованого рішення	17
2.1 Загальна структура проєктованої системи.....	17
2.2 Алгоритмічне забезпечення.....	21
2.3 Інформаційне забезпечення модуля класифікації відходів.....	25
3 Програмно-технологічне забезпечення модуля	29
3.1 Реалізація програмного забезпечення	29
3.2 Інтерфейс користувача	34
3.3 Тестування розробленої програми	36
Висновки	41
Список використаних джерел	43
Додаток А Лістинг коду моделі класифікації відходів	47
Додаток Б Лістинг коду реалізації інтерактивного інтерфейсу застосунку з класифікацією відходів.....	49
Додаток В Апробація отриманих результатів	53

ВСТУП

Проблема забруднення навколишнього середовища та неефективного управління твердими побутовими відходами (ТПВ) є однією з найгостріших у сучасному світі. Через зростання чисельності населення, розвиток урбанізації та промисловості обсяги відходів постійно збільшуються. За даними Світового банку, щорічно у світі утворюється понад 2 мільярди тонн ТПВ, а до 2050 року ця цифра може досягти 3,4 мільярда тонн. При цьому лише близько 13,5% цих відходів піддаються переробці, а решта забруднює навколишнє середовище [1, 2].

Традиційні методи сортування, які базуються переважно на ручній праці, є малоефективними, трудомісткими та небезпечними для здоров'я. У зв'язку з цим набуває актуальності розробка модуля класифікації відходів, здатного самостійно розпізнавати типи сміття. Перспективним напрямом у цій сфері є застосування методів глибокого навчання для класифікації зображень ТПВ у режимі реального часу.

Метою роботи є створення модуля класифікації відходів на основі глибокого навчання.

Для досягнення поставленої мети у роботі потрібно добитись поставлених задач:

1. Проаналізувати теоретичні основи управління відходами та сучасні методи автоматизації їх сортування.
2. Дослідити та систематизувати основні методи аналізу даних, що застосовуються для класифікації відходів.
3. Розробити загальну архітектуру системи автоматизованої класифікації та сортування відходів з описом алгоритмічного та інформаційного забезпечення.
4. Реалізувати програмний модуль для класифікації відходів із розробкою інтерфейсу користувача.
5. Виконати тестування програмного забезпечення та оцінити ефективність запропонованої системи.

Об'єктом дослідження є процес автоматичної класифікації відходів на основі зображень.

Предметом дослідження є розробка модуля глибокого навчання для розпізнавання категорій відходів у режимі реального часу.

Результати дослідження опубліковано в матеріалах науково-практичної конференції «Інтелектуальні інформаційні технології в прикладних дослідженнях» (ІІТАР - 2025), м. Тернопіль, 27-29 травня 2025 р. (додаток В).

1 ТЕОРЕТИЧНІ ЗАСАДИ ДЛЯ АНАЛІЗУ ПРОБЛЕМАТИКИ УПРАВЛІННЯ ВІДХОДАМИ

1.1 Теоретичні аспекти управління відходами

Управління відходами є комплексною системою, що охоплює процеси збору, транспортування, переробки та утилізації відходів. Ключовим етапом цього циклу є сортування відходів, яке дозволяє розподілити матеріали за категоріями для подальшої переробки або екологічно безпечного видалення. Ефективність цього етапу безпосередньо впливає на можливість вторинного використання матеріалів і зменшення навантаження на довкілля.

Відходи можуть бути класифіковані за різними категоріями, зокрема:

- пластик - ПЕТ-пляшки, поліетиленові пакети, пластикові контейнери та упаковки;
- папір - газети, журнали, картонні коробки, паперові пакети;
- скло - пляшки, банки, скляні контейнери;
- метал - алюмінієві банки, консервні банки, металеві кришки;
- упаковки - пакування для продуктів і тощо.

Правильне сортування цих матеріалів є важливим для забезпечення ефективного повторного використання ресурсів та мінімізації екологічного впливу. Однак процес візуальної ідентифікації відходів може бути складним через значну варіативність форм, кольорів, стану забруднення та перекриття матеріалів.

Автоматизовані системи сортування відходів, які базуються на технологіях комп'ютерного зору та штучного інтелекту, мають значний потенціал для підвищення ефективності цього процесу.

Основні компоненти таких систем включають:

- системи збору зображень - камери, що фіксують зображення відходів у реальному часі;
- модулі попередньої обробки - алгоритми нормалізації, сегментації та виділення ключових ознак;

- алгоритми класифікації - штучні нейронні мережі, що аналізують зображення та визначають категорію відходів;
- системи управління - програмні комплекси, що інтегрують алгоритми аналізу з механізмами автоматизованого сортування.

Використання методів глибокого навчання у сортуванні відходів має значні переваги над традиційними методами комп'ютерного зору:

- висока точність - сучасні моделі можуть ефективно розпізнавати складні візуальні об'єкти;
- адаптивність - нейронні мережі можуть навчатися на нових наборах даних і покращувати свої результати;
- стійкість до шуму - здатність працювати з зображеннями різної якості та у змінних умовах освітлення;
- масштабованість - можливість класифікації великої кількості різноманітних відходів.

Однак, розробка ефективних моделей автоматизованого сортування відходів стикається з певними викликами:

- велика різноманітність матеріалів - широкий спектр типів відходів із різними фізичними характеристиками;
- забруднення - бруд, залишки їжі та інших речовин можуть ускладнювати ідентифікацію;
- освітлення та умови зйомки - різноманітні умови зйомки впливають на якість вхідних зображень;
- перекриття об'єктів - відходи можуть накладатися один на одного, що ускладнює їх аналіз;
- обмежена доступність даних - нестача великих наборів зображень для навчання моделей глибокого навчання.

Таким чином, розробка автоматизованих систем сортування відходів є надзвичайно важливим напрямом для забезпечення сталого розвитку та зменшення екологічного навантаження на навколишнє середовище. Інтеграція методів глибокого навчання у процес управління відходами дозволить створити ефективні

рішення, що сприятимуть підвищенню рівня вторинної переробки та зменшенню кількості відходів, які потрапляють на сміттєзвалища. Тема є актуальною через широке застосування методів глибокого навчання в задачах класифікації.

1.2 Основні методи аналізу даних у класифікації відходів

Класифікація відходів на основі зображень є однією з актуальних задач в галузі машинного навчання та комп'ютерного зору. Для вирішення цієї задачі використовуються різноманітні методи аналізу даних, які дозволяють здійснювати ефективне розпізнавання об'єктів та їх категоризацію. Залежно від складності системи, доступності даних та обчислювальних ресурсів, виділяють кілька основних груп методів.

До появи глибоких нейронних мереж класифікація зображень, зокрема відходів, здійснювалася за допомогою класичних алгоритмів машинного навчання, які працюють із заздалегідь обраними ознаками (features).

Найпоширеніші методи:

- метод опорних векторів (SVM) - дозволяє здійснювати багатокласову класифікацію за допомогою розподілення даних у просторі ознак;
- методи кластеризації (K-Means, DBSCAN) - застосовуються для групування схожих зображень за схожими параметрами;
- методи Random Forest - використовуються для прийняття рішень на основі випадкових дерев рішень.

Недоліком таких методів є потреба у ручному виборі ознак (наприклад, гістограми орієнтованих градієнтів (HOG), кольорові гістограми, текстурні характеристики), що значно обмежує їхню гнучкість та точність.

Сучасні системи автоматизації класифікації зображень базуються на згорткових нейронних мережах (CNN). Такі моделі автоматично виділяють ознаки зображення під час навчання, що дозволяє значно підвищити точність і адаптивність системи.

Популярні архітектури:

- AlexNet - перша глибока мережа, що продемонструвала перевагу CNN у змаганні ImageNet [3];
- VGG16, VGG19 - мережі з послідовним розташуванням однотипних шарів [4, 5];
- ResNet50, ResNet101 - мережі з залишковими зв'язками (skip-connections), які дозволяють тренувати надглибокі мережі [6];
- MobileNet, EfficientNet - легкі моделі для мобільних і вбудованих систем [7, 8].

Переваги CNN:

- автоматичне вилучення ознак;
- висока точність класифікації;
- адаптація до нових класів та умов;
- можливість використання трансферного навчання.

Окремим напрямом є використання попередньо натренованих моделей (наприклад, MobileNetV2, ResNet50, InceptionV3), які навчені на великих наборах зображень (ImageNet). Такі моделі дозволяють:

- скоротити час навчання;
- досягти високої точності навіть на малих датасетах;
- використовувати мобільні та веб-рішення.

У деяких випадках можуть використовуватись рекурентні нейронні мережі (RNN, LSTM) для класифікації послідовних зображень або для аналізу динаміки накопичення відходів у часі.

Додатково, для підвищення точності класифікації, можуть комбінуватися кілька методів:

- інтеграція з даними сенсорів;
- використання фотоелектричних сенсорів та спектрометрії;
- об'єднання CNN та класичних алгоритмів (гібридні моделі).

Таким чином, сучасні методи аналізу даних у класифікації відходів дозволяють створювати адаптивні, точні та масштабовані системи для екологічного моніторингу та сортування сміття.

1.3 Аналіз існуючих рішень у сфері автоматизації сортування відходів

За останній період автоматизована класифікація відходів за допомогою методів машинного навчання та комп'ютерного зору значно розвинулася. Розглянемо найбільш значущі існуючі рішення та їхні особливості.

До появи методів глибокого навчання класифікація відходів здебільшого здійснювалася за допомогою класичних алгоритмів комп'ютерного зору.

Серед них:

- метод опорних векторів (SVM) з використанням ручно розроблених ознак, таких як гістограми орієнтованих градієнтів (HOG) або масштабно-інваріантне перетворення ознак (SIFT) [9];
- випадкові ліси (Random Forests) для класифікації на основі текстурних та кольорових ознак [10];
- методи кластеризації для групування схожих типів відходів [11].

Хоча ці підходи мали певний рівень ефективності, вони демонстрували обмежену точність (60-75%) через складність у визначенні ефективних ознак для різних типів відходів. Висока чутливість до змін навколишнього середовища, таких як освітлення та фон, робила ці методи менш придатними для реального використання.

З розвитком глибокого навчання згорткові нейронні мережі стали домінуючим підходом у класифікації зображень, включаючи класифікацію відходів.

Основні дослідження в цій галузі включають:

1. Yang і Thung - використання архітектури AlexNet для класифікації відходів на 6 категорій, досягнувши точності 87% [12].

2. Bircanoğlu et al - розробка власної архітектури CNN для 5 категорій відходів, точність 95% [13].

Перевагами CNN є висока точність, автоматичне виділення ознак та можливість навчання на великих наборах даних. Недоліки включають потребу у великих обчислювальних ресурсах і складність адаптації до нових умов.

Трансферне навчання дозволяє використовувати попередньо навчені моделі для класифікації відходів, що є ефективним способом розв'язання проблеми нестачі даних. Одні із важливих досліджень:

1. Vo et al - використання MobileNetV2, попередньо навченої на ImageNet, для класифікації відходів, точність 96% [14].

2. Ruiz et al - порівняння різних архітектур (VGG16, ResNet50, Inception-v3) із трансферним навчанням [15].

Цей підхід є ефективним для застосування на мобільних пристроях та у реальному часі, проте він все ще залежить від специфічності набору даних, на якому навчалася модель.

Промислові системи автоматизованого сортування:

1. Конвеєрні лінії:

- функціонал: транспортування відходів між етапами сортування;
- мобільні рішення: мобільні сортувальні лінії дозволяють обробляти пластик, метал, папір та органічні відходи без стаціонарної інфраструктури.

2. Механічні та сенсорні методи:

- магнітне, повітряне та електричне розділення: використовуються для виділення металів, пластиків та інших матеріалів;
- фотоелектричні сенсори: аналізують оптичні властивості для ідентифікації матеріалів.

3. Автоматизовані системи управління:

- керування WINCC: налаштування режиму роботи (3-рівневі режими) та моніторинг процесів у реальному часі;
- інтеграція з датчиками: відстеження параметрів (наприклад, температури, вологості) для оптимізації сортування.

Компанія Beston Group пропонує системи потужністю 10-25 т/год з ефективністю до 90%, призначені для обробки твердих побутових відходів (ТПВ) та будівельних відходів. KONSORT спеціалізується на конвеєрних системах для полігонів та мобільних лініях для сортування пластиків, металів і органічних відходів [16, 17].

Ці системи мають високу ефективність у промисловому масштабі, але їх впровадження потребує значних фінансових вкладень та технічного обслуговування.

Попри досягнутий прогрес, існуючі підходи мають певні обмеження:

1. Обмежена кількість класів - більшість систем класифікують відходи лише на 5-12 категорій, що недостатньо для повного охоплення всіх типів відходів.

2. Проблеми з генералізацією - моделі, навчені на конкретних наборах даних, можуть мати проблеми з точністю при класифікації нових видів відходів або за змінених умов.

3. Висока обчислювальна складність - багато сучасних моделей вимагають потужного обладнання, що ускладнює їх впровадження на мобільних пристроях чи в реальному часі.

Головною відмінністю запропонованої системи є її здатність виконувати класифікацію сміття в режимі реального часу. На відміну від традиційних підходів, де зображення аналізуються після їх попереднього збору та збереження, розроблена система інтегрується безпосередньо з відеопотоком камери. Це дозволяє здійснювати безперервний моніторинг об'єктів у кадрі та миттєво визначати тип сміття, що забезпечує зменшення затримки в ідентифікації.

1.4 Постановка задачі

У сучасному світі проблема утилізації відходів набуває все більшої актуальності. Зростаючий обсяг відходів та недостатньо ефективні системи їх сортування призводять до значного забруднення навколишнього середовища. Одним із перспективних напрямів вирішення цієї проблеми є впровадження

автоматизованих систем класифікації відходів на основі методів глибокого навчання. Такі системи дозволяють підвищити ефективність сортування, зменшити витрати на ручну працю та покращити якість вторинної переробки.

Метою є створення модуля класифікації відходів, який здатний автоматично визначати категорію відходу на основі зображень. Такий модуль повинен забезпечувати високу точність розпізнавання, щоб у подальшому інтегруватися з існуючими системами управління відходами та бути зручним у використанні.

Окрім цього, система має потенціал для наступного вдосконалення шляхом використання глибоких нейронних мереж та додавання можливостей самонавчання для підвищення точності класифікації відходів.

Отже, метою роботи є створення модуля класифікації відходів на основі глибокого навчання.

Можна виділити такі основні задачі, необхідні для реалізації системи автоматичної класифікації відходів:

1. Проаналізувати теоретичні основи управління відходами та сучасні методи автоматизації їх сортування.
2. Дослідити та систематизувати основні методи аналізу даних, що застосовуються для класифікації відходів.
3. Розробити загальну архітектуру системи автоматизованої класифікації та сортування відходів з описом алгоритмічного та інформаційного забезпечення;
4. Реалізувати програмний модуль для класифікації відходів із розробкою інтерфейсу користувача.
5. Виконати тестування програмного забезпечення та оцінити ефективність запропонованої системи.

2 ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТАЛЬНІ РОЗРОБКИ ПРОПОНОВАНОГО РІШЕННЯ

2.1 Загальна структура проєктованої системи

На рисунку 2.1 представлена схема модуля класифікації відходів, яка описує основні етапи роботи системи автоматичного розпізнавання, обробки та аналізу відходів.



Рисунок 2.1 - Схема модуля класифікації відходів

Етап розпізнавання відходів відповідає за аналіз та ідентифікацію відходів на основі отриманих зображень. Основний підпроцес - розпізнавання зображень відходів, яке може виконуватися за допомогою технологій комп'ютерного зору та алгоритмів машинного навчання. У системі застосовуються нейромережі для ідентифікації типу відходів за фото. Важливим аспектом є правильне виявлення форми, кольору та інших характеристик, які допомагають визначити категорію сміття.

Після розпізнавання відходів необхідно привести отримані дані у стандартизований вигляд. На етапі обробки інформації виконується форматування для подальшого аналізу, яке включає очищення даних від шуму, виправлення можливих помилок у визначенні та структуризацію отриманої інформації. Відформатовані дані підготовлюються до подальшої класифікації та розподілу за відповідними категоріями.

Основна функція модуля класифікації зображень - нормалізація та категоризація відходів відповідно до їхніх характеристик.

На цьому етапі відходи можуть бути розділені на основні групи:

- пластик (пляшки, упаковки, контейнери);
- метал (банки, металеві деталі);
- папір (газети, картон, книги);
- скло (пляшки, банки, скlobій);
- упаковки.

Класифікація дозволяє системі визначити оптимальні способи подальшої утилізації та переробки, зменшуючи екологічне навантаження та сприяючи повторному використанню матеріалів.

Інтерфейс користувача це модуль забезпечує зручну взаємодію користувача із системою. Він включає:

- панель управління процесом класифікації, яка дозволяє налаштовувати параметри розпізнавання та обробки даних, коригувати алгоритми роботи системи;
- відображення результатів аналізу, що надає користувачеві інформацію про визначену категорію відходів.

Представлена схема демонструє логіку роботи системи автоматичної класифікації відходів, яка дозволяє швидко та ефективно розпізнавати сміття, обробляти інформацію, класифікувати відходи та надавати користувачеві відповідні дані.

Запропонований модуль класифікації відходів базується на поєднанні ефективної архітектури нейронної мережі (MobileNetV2), методів попередньої обробки зображень та адаптивної оптимізації.

Перед подачею зображення до нейронної мережі виконується його нормалізація згідно з функцією `preprocess_input`, яка масштабує значення пікселів до інтервалу -1 до 1 [18].

Формула нормалізації має вигляд:

$$x_{norm} = \frac{x}{127.5} - 1, \quad (2.1)$$

де x - значення пікселя у діапазоні $[0, 255]$;

x_{norm} - нормалізоване значення пікселя у діапазоні $[-1, 1]$.

Нормалізація пікселів дозволяє нейронній мережі ефективніше працювати з вхідними даними, оскільки значення зображення перетворюються в уніфікований інтервал значень, що покращує збіжність при навчанні.

Модель побудована на основі MobileNetV2 - згорткової нейронної мережі, оптимізованої для мобільних пристроїв [19].

Основна операція - глибока згортка (depthwise convolution), що зменшує кількість параметрів:

$$Y_{i,j,k} = \sum_{m,n} X_{i+m,j+n,k} * K_{m,n,k}, \quad (2.2)$$

де X - вхідна тензорна карта ознак;

Y - результат згортки;

i, j - просторові координати пікселя;

k - канал (глибина);

m, n - координати фільтра;

K - ядро згортки (фільтр).

Інверсне залишкове з'єднання (inverted residual) дозволяє мережі зберігати важливу інформацію з попередніх шарів і полегшує навчання.

Після шару глобального середнього згортання:

$$z_k = \frac{1}{H*W} \sum_{i=1}^H \sum_{j=1}^W x_{ijk}, \quad (2.3)$$

де z_k - усереднене значення по карті ознак для каналу k ;

H, W - висота і ширина ознак;

x_{ijk} --значення в пікселі (i, j) і каналі k .

Це дозволяє зменшити розмірність просторової інформації, зберігаючи при цьому важливі ознаки для подальшої обробки [20].

Додаються шари:

- Dense (128, ReLU) шар з нелінійною активацією:

$$f(x) = \max(0, x), \quad (2.4)$$

де x - вхідне значення до нейрону.

Функція активації обчислюється як максимум між нулем і вхідним значенням [21].

- Dropout (0.4) для запобігання перенавчанню: під час кожної ітерації тренування випадковим чином відключається 40% нейронів у шарі. Це допомагає уникнути надмірного пристосування моделі до тренувальних даних і сприяє кращій здатності до узагальнення на нових, невідомих даних.

- Dense (5, softmax) вихідний шар із softmax-функцією:

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}, \quad (2.5)$$

де $P(y_i)$ - ймовірність належності до класу i ;

z_i - логіт (нормалізоване значення на виході);

K - кількість класів (у даному випадку $K=5$).

Вона перетворює вихідні значення моделі на ймовірності належності до кожного з п'яти класів [22].

Було застосовано функцію втрат. Для багатокласової класифікації застосовується крос-ентропія (Categorical Crossentropy).

Це стандартна функція втрат для задач класифікації з декількома класами. Вона вимірює розбіжність між фактичним розподілом міток класів та передбаченим розподілом ймовірностей [23, 24].

$$L = \sum_{i=1}^K y_i \log(\hat{y}_i), \quad (2.6)$$

де L - значення функції втрат;

y_i - фактична мітка класу (0 або 1);

$\widehat{y}_i$ - ймовірність, передбачена мережею для класу i ;

K - кількість класів.

Алгоритм оптимізації Adam використовує адаптивне оновлення параметрів:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \quad (2.7)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \quad (2.8)$$

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t}, \quad \widehat{v}_t = \frac{v_t}{1 - \beta_2^t}, \quad (2.9)$$

$$\theta_{t+1} = \theta_t - \alpha * \frac{\widehat{m}_t}{\sqrt{\widehat{v}_t} + \epsilon}. \quad (2.10)$$

де θ_t - параметри моделі на момент часу t ;

g_t - градієнт функції втрат;

m_t, v_t - перший та другий моменти градієнта (середнє і дисперсія);

β_1, β_2 - коефіцієнти згладжування;

α - швидкість навчання;

ϵ - мале число для запобігання діленню на нуль.

Такий механізм адаптивного оновлення дозволяє алгоритму швидко реагувати на зміни градієнта та підтримувати ефективне навчання в різних умовах [25].

Поточний підхід на основі сучасних алгоритмів машинного навчання сприяє підвищенню точності класифікації та створює умови для подальшого вдосконалення модуля та його впровадження в системи сортування відходів.

2.2 Алгоритмічне забезпечення

Процес автоматизованої класифікації відходів починається з отримання зображення відходів із відповідного джерела. Це зображення надходить у систему для подальшої обробки, де аналізується його якість та відповідність встановленим параметрам.

Загальна схема процесу автоматизованої класифікації відходів зображена на рисунку 2.2.

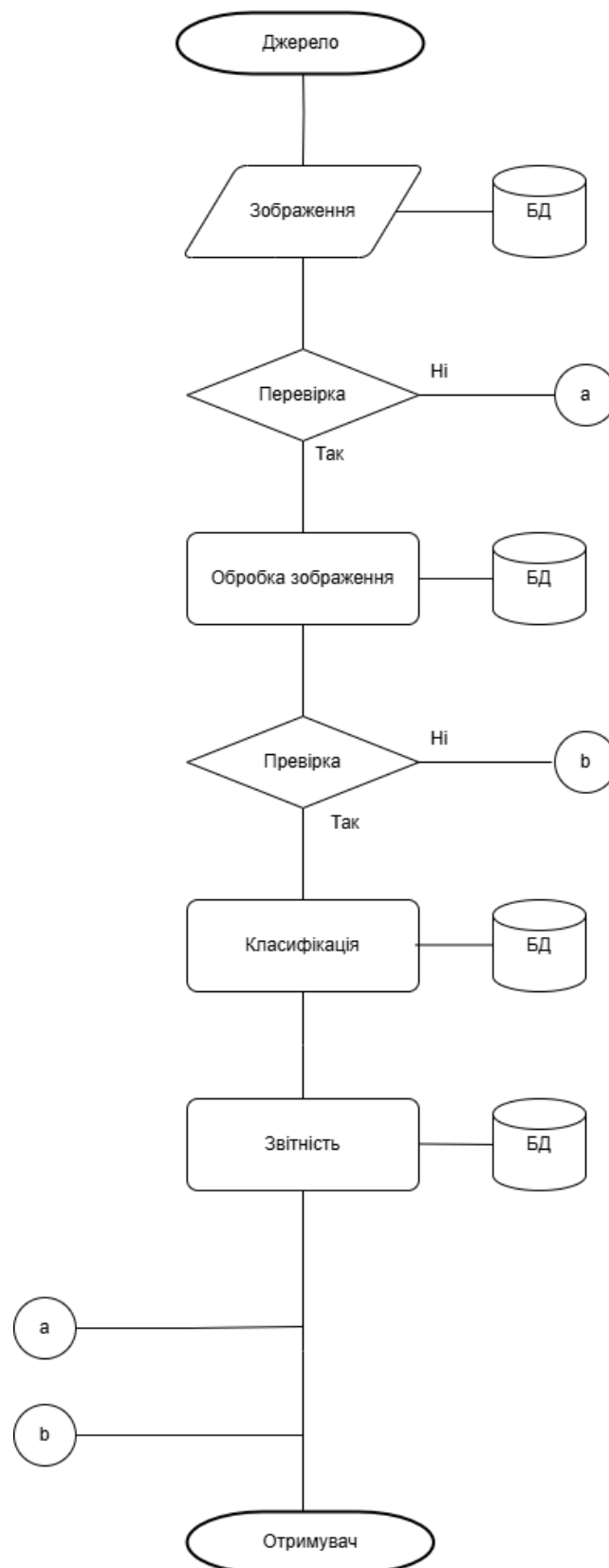


Рисунок 2.2 - Схема алгоритму класифікації зображення

На першому етапі здійснюється перевірка якості отриманого зображення. Система оцінює його чіткість, роздільну здатність та наявність необхідних деталей для подальшого аналізу. Якщо зображення є неякісним або містить суттєві дефекти, воно зберігається у базі даних як непридатне, і обробка завершується. У разі відповідності всім вимогам система переходить до наступного етапу.

Далі здійснюється обробка зображення, яка включає попередню фільтрацію, виявлення контурів об'єкта та вилучення ключових ознак. Для цього застосовуються методи комп'ютерного зору та машинного навчання, що сприяють підвищенню точності розпізнавання. Отримані характеристики аналізуються, і система визначає, чи є даних достатньо для подальшої класифікації. Якщо зображення не відповідає необхідним вимогам, воно ігнорується як непридатне, і обробка припиняється.

У разі успішного проходження всіх етапів система за допомогою нейронної мережі виконує класифікацію відходів за відповідними категоріями. Після цього формується звіт про точність розпізнавання, що дає змогу оцінити якість класифікації та ефективність алгоритмів.

Система автоматизованої класифікації зображень відходів призначена для аналізу та сортування сміття, отриманого з джерел. Основна мета модуля - визначення типу відходів та їх групування для подальшого аналізу, прогнозування накопичень і покращення системи утилізації. Це дозволяє не лише оптимізувати процес переробки, а й сприяти екологічному моніторингу. Загальна схема роботи модуля автоматизованої класифікації відходів зображена на рисунку 2.3.

Алгоритм починається з отримання зображень із зовнішніх джерел. Після надходження зображення записується у базу даних і проходить перевірку на відповідність певним параметрам. Система аналізує якість зображення, роздільну здатність, коректний формат та перевіряє, чи містить воно сміття. Якщо файл не відповідає вимогам або є нерозпізнаним, його відхиляють або направляють на повторну обробку.

Далі зображення проходить попередню обробку, яка включає корекцію яскравості, контрасту, видалення шумів та можливе виділення ключових областей

з відходами. Цей етап важливий, оскільки дозволяє зменшити вплив фонових об'єктів та покращити точність класифікації. Після обробки система порівнює зображення з наявними зразками у базі даних. Якщо є збіг або достатньо висока схожість із вже класифікованими об'єктами, система переходить до наступного етапу.

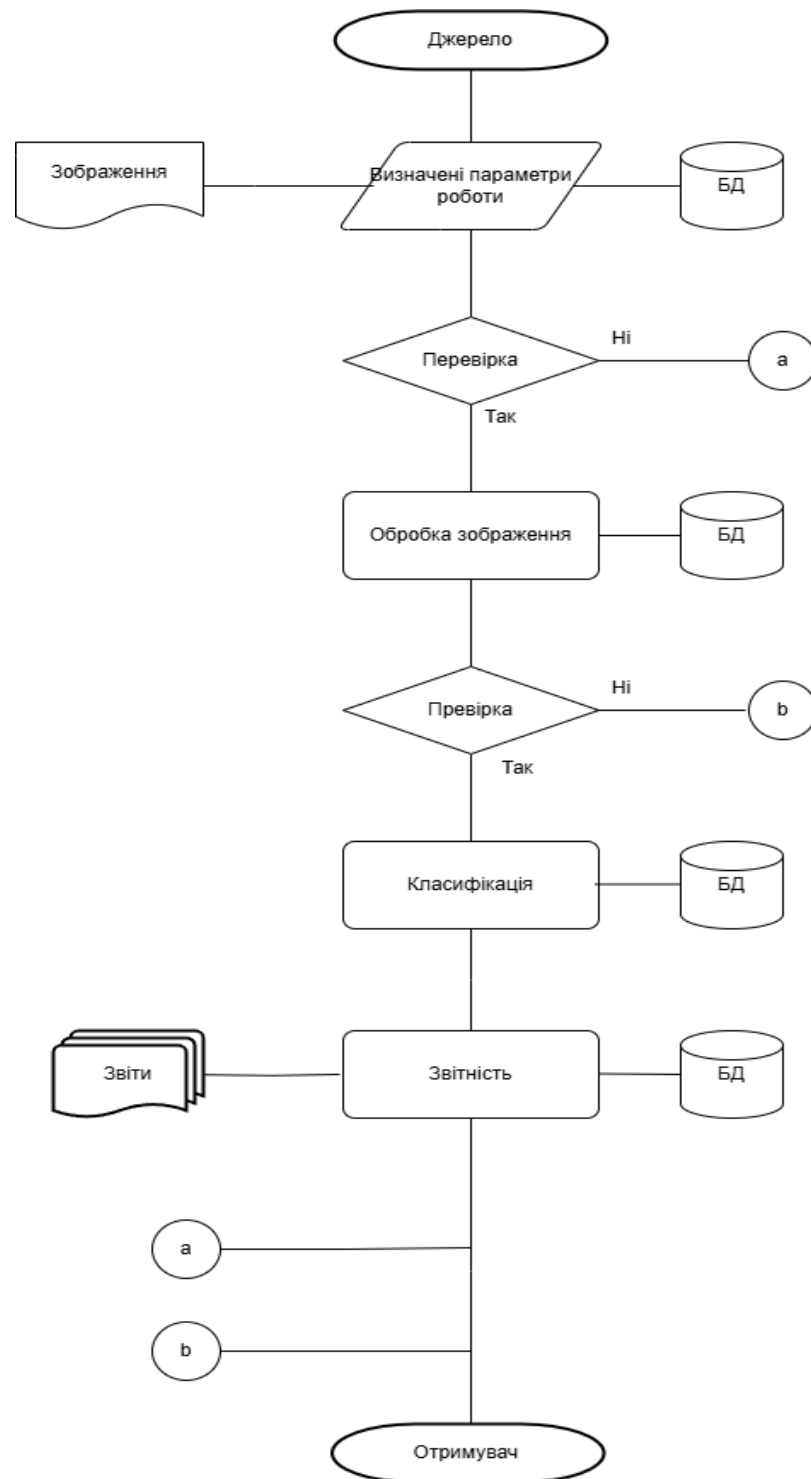


Рисунок 2.3 - Схема алгоритму модуля класифікації зображення

Основний етап - класифікація, яка виконується за допомогою нейронної мережі. На цьому рівні модель аналізує вхідне зображення та визначає, до якої категорії відходів воно належить (пластик, скло, метал тощо). Додатково можуть враховуватись особливі характеристики, такі як форма, текстура та матеріал. Результати класифікації зберігаються у базі даних разом із координатами місця зйомки.

Останній етап - формування звітів та надання результатів для прийняття рішень. Отримані дані можуть бути використані користувачами для сортування відходів, а також підприємствами для оптимізації збору та переробки відходів, а також допомагати в екологічному моніторингу.

2.3 Інформаційне забезпечення модуля класифікації відходів

Інформаційне забезпечення є ключовим елементом функціонування системи автоматизованої класифікації відходів. Воно включає дані, які використовуються для навчання, тестування та роботи моделі, а також структуру бази даних, що забезпечує збереження, обробку й аналіз результатів класифікації.

Для навчання моделі було використано зображення у форматі .jpg, які містили приклади різних типів побутових відходів. Датасет складався з наступних категорій:

- пластик (пляшки, упаковки, контейнери);
- папір (газети, картон, книги);
- скло (пляшки, банки, скlobій);
- метал (консервні банки, металеві деталі);
- упаковки (змішані типи пакувальних матеріалів, що не підпадають під окремі категорії).

Загалом, використаний датасет містив збалансовану кількість зображень для кожного з обраних класів, таких як пластик, папір, скло, метал, упаковки тощо. Це забезпечило рівномірне представлення кожної категорії, що є важливою умовою для побудови ефективної моделі глибокого навчання. Завдяки цьому вдалося

уникнути упередженості моделі до окремих класів, оскільки вона мала можливість вивчити характерні ознаки кожного типу відходів у достатній кількості прикладів.

Перед початком навчання всі зображення проходили етап попередньої обробки, що є обов'язковим кроком у підготовці даних для нейронної мережі. Зокрема, було виконано зміну розміру зображень до стандартного формату, сумісного з архітектурою моделі MobileNetV2. Це дозволяє забезпечити однаковий вхідний розмір зображень, що критично важливо для правильного функціонування згорткових шарів.

Наступним етапом була нормалізація пікселів, тобто приведення значень кольорових каналів до діапазону від -1 до 1. Це дозволяє моделі швидше та стабільніше навчатися, оскільки нормалізовані дані краще сприймаються алгоритмами оптимізації та пришвидшують процес збіжності моделі.

Таким чином, завдяки збалансованому датасету та продуманій обробці зображень вдалося сформувати надійне та стабільне інформаційне середовище для навчання моделі, що є основою її подальшої ефективної роботи у задачах класифікації побутових відходів.

Для забезпечення збереження результатів класифікації та подальшого аналізу роботи моделі необхідною є також структурована система зберігання даних.

Структура бази даних представлена у вигляді реляційної моделі, що зображена на рисунку 2.4. Вона описує взаємозв'язки між таблицями, які забезпечують збереження та обробку інформації, отриманої в результаті класифікації відходів.

Основні таблиці та їхні атрибути:

- таблиця "Відходи": id, id_Класифікація, Дата_фіксації;
- таблиця "Категорія": id, Назва, Кількість;
- таблиця "Статистика": id, id_Відходів, Точність_класифікації, Дата_аналізу.

Усі таблиці пов'язані зовнішніми ключами, що гарантує узгодженість даних та дозволяє виконувати складні аналітичні запити для покращення роботи модуля.



Рисунок 2.4 - ER-діаграма

При проектуванні фізичної моделі (рисунок 2.5) було визначено оптимальні типи даних:

- ID зберігаються у форматі INTEGER з автоінкрементом;
- текстові дані зберігаються у форматах VARCHAR(255) та TEXT;
- дати представлені у форматі DATETIME;
- числові показники (наприклад, точність класифікації) зберігаються у форматі DECIMAL(5,2).

Фізична модель також передбачає індексацію первинних та зовнішніх ключів, що підвищує продуктивність запитів до бази даних. Встановлено обмеження NOT NULL для обов'язкових полів та обмеження UNIQUE для унікальних атрибутів.

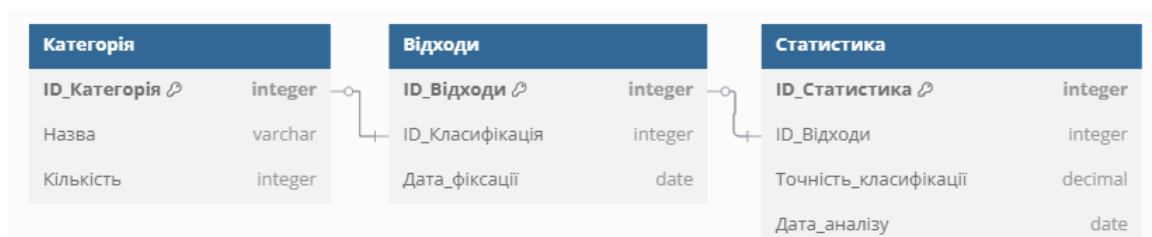


Рисунок 2.5 - Фізична модель

Фінальна структура бази даних дозволяє зберігати всю необхідну інформацію для класифікації відходів, аналізу точності моделі та створення звітів, що сприяє покращенню ефективності системи.

Таким чином, інформаційне забезпечення системи охоплює як навчальні дані для побудови моделі, так і структуровану базу даних, яка слугує ядром для обліку результатів класифікації, подальшої аналітики та вдосконалення алгоритмів.

3 ПРОГРАМНО-ТЕХНОЛОГІЧНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ

3.1 Реалізація програмного забезпечення

У даному проєкті для класифікації типів відходів використано мову програмування Python, яка є однією з найбільш популярних мов у галузі машинного навчання та комп'ютерного зору. Python дозволяє ефективно працювати з різними бібліотеками та інструментами, що сприяють швидкому розвитку і тестуванню моделей глибокого навчання.

Python - це високорівнева мова програмування, що акцентує увагу на зручності та простоті синтаксису. Завдяки своїй читабельності та простоті використання, Він став основним інструментом для розробки програм в галузях наукових досліджень, штучного інтелекту та машинного навчання. Python підтримує безліч бібліотек для наукових обчислень, що дозволяє ефективно виконувати складні математичні та статистичні розрахунки, а також обробляти великі об'єми даних [26].

Основною бібліотекою, що була використана для створення моделі, є TensorFlow, яка включає в себе модуль Keras. TensorFlow - це одна з найбільш потужних та популярних бібліотек для створення та тренування моделей машинного навчання, зокрема нейронних мереж. TensorFlow дозволяє працювати з численними типами моделей, починаючи від простих до складних архітектур глибокого навчання. Водночас Keras, що є частиною TensorFlow, надає зручний інтерфейс для швидкого прототипування та тестування різних архітектур нейронних мереж [27].

У цьому проєкті використовувалася MobileNetV2 - попередньо натренована модель згорткової нейронної мережі (CNN), яка була спеціально оптимізована для використання в мобільних пристроях. Це дозволяє досягти високої ефективності при обмежених обчислювальних ресурсах, що особливо важливо для реальних застосувань, таких як мобільні додатки або системи з обмеженими ресурсами. MobileNetV2 належить до сімейства моделей MobileNet, які відомі своєю швидкістю та ефективністю в задачах класифікації зображень.

Модель MobileNetV2 була адаптована до задачі класифікації сміття шляхом додавання кількох додаткових шарів:

- GlobalAveragePooling2D цей шар зменшує розміри зображення до єдиного вектору, що дозволяє зберегти основні ознаки зображення;
- Dense - це шар повнозв'язаний, що забезпечує високорівневі характеристики зображення на основі результатів попередніх шарів;
- Dropout додає випадкове відключення деяких нейронів під час тренування для запобігання перенавчанню;
- Softmax останній шар, який генерує ймовірності для кожного класу, дозволяючи класифікувати зображення в одну з шести категорій: картон, скло, метал, пакувальний матеріал, папір і пластик.

Ця архітектура моделі дозволяє досягти високої точності при класифікації зображень, зокрема для специфічних задач, таких як сортування сміття. На рисунку 3.1 зображено результати тренування моделі протягом 12 епох. На рисунку видно, що точність на тренувальних даних поступово зростає, досягаючи понад 90% до кінця навчання. Це свідчить про ефективне засвоєння модельною архітектурою закономірностей у тренувальному наборі.

```
Epoch 11/20  
54/54 ————— 4s 76ms/step - accuracy: 0.9715 - loss: 0.1175 - val_accuracy: 0.7896 - val_loss: 0.5281  
Epoch 12/20  
54/54 ————— 4s 82ms/step - accuracy: 0.9722 - loss: 0.0997 - val_accuracy: 0.8227 - val_loss: 0.5230
```

Рисунок 3.1 - Результати тренування моделі класифікації відходів

Для підготовки даних до тренування використовувалася бібліотека ImageDataGenerator з TensorFlow, яка автоматизує процес завантаження та обробки зображень. Вона також підтримує техніки аугментації даних, що дозволяє значно збільшити варіативність тренувальних даних та підвищити здатність моделі до узагальнення. Завдяки автоматичному розподілу даних на тренувальні та валідаційні набори, процес навчання став більш зручним і ефективним.

Під час навчання модель оптимізувалася з використанням функції втрат categorical_crossentropy, що є стандартною для задач багатокласової класифікації.

Оптимізатор Adam забезпечує ефективну оптимізацію параметрів мережі, оскільки він поєднує переваги двох методів - адаптивних моментів та адаптивних швидкостей навчання, що дозволяє досягти високих результатів на меншій кількості етапів. Програмний код моделі представлений у додатку А.

Для відстеження об'єктів у реальному часі була застосована бібліотека OpenCV. Це одна з найпопулярніших бібліотек для комп'ютерного зору та обробки зображень, що дозволяє захоплювати кадри з відеокамери, застосовувати різноманітні методи для обробки зображень і здійснювати операції, як відстеження об'єктів, виявлення контурів та накладання тексту на зображення.

Також, використовувався алгоритм CSRT Tracker для відстеження об'єктів у реальному часі. Алгоритм CSRT (Discriminative Correlation Filter with Channel and Spatial Reliability) є одним з ефективних підходів до відстеження об'єктів в реальному часі, зокрема в умовах зміни положення або вигляду об'єкта [28].

На рисунку 3.2 зображено зміну точності моделі під час навчання та перевірки на валідаційному наборі. Видно, що протягом перших кількох епох точність стрімко зростає, що свідчить про ефективне навчання.

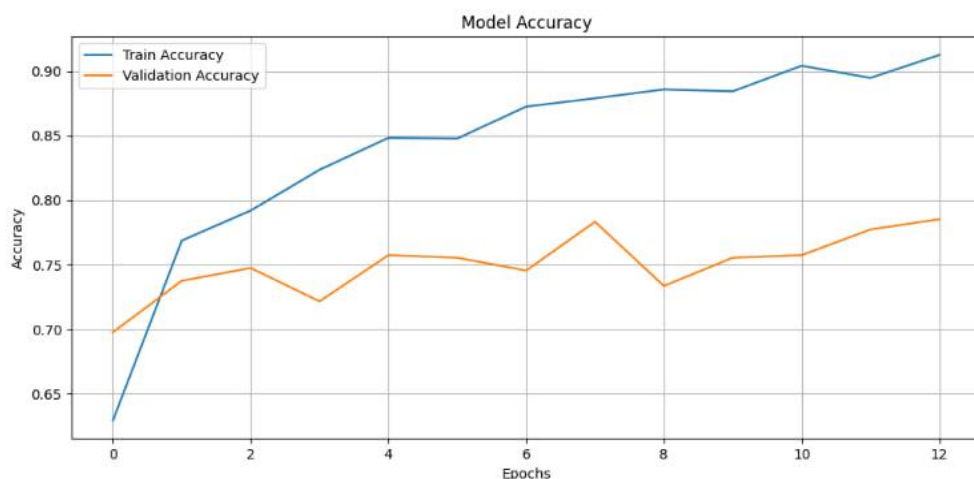


Рисунок 3.2 - Графік точності навчання та валідації (Train/Validation Accuracy)

Рисунок 3.3 демонструє, як змінюється функція втрат (loss) під час навчання та валідації. Спад значень loss для тренувальних даних вказує на успішне навчання моделі. Незважаючи на певні коливання валідаційної функції втрат, загальна

тенденція залишається стабільною. Це дозволяє зробити висновок, що модель добре узагальнює дані, зберігаючи баланс між точністю та здатністю адаптуватися до нових прикладів.

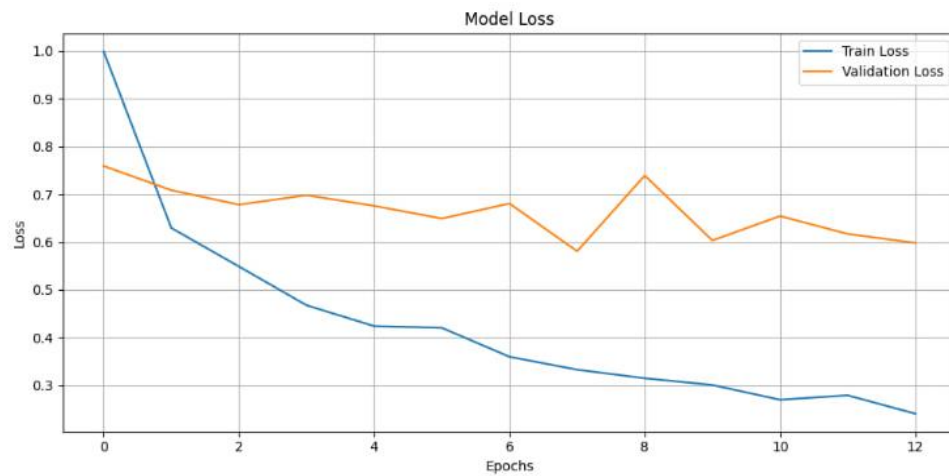


Рисунок 3.3 - Графік втрат моделі (Train/Validation Loss)

На рисунку 3.4 представлено графічне порівняння трьох основних метрик якості класифікації - точності, повноти та F1-міри - для кожного класу. Кожна з метрик відображена окремими кольорами, що дозволяє наочно порівняти результати моделі по кожному типу відходів. Такий підхід дає змогу оцінити збалансованість класифікації, виявити сильні сторони моделі та визначити ті класи, які класифікуються гірше й потребують подальшого покращення.

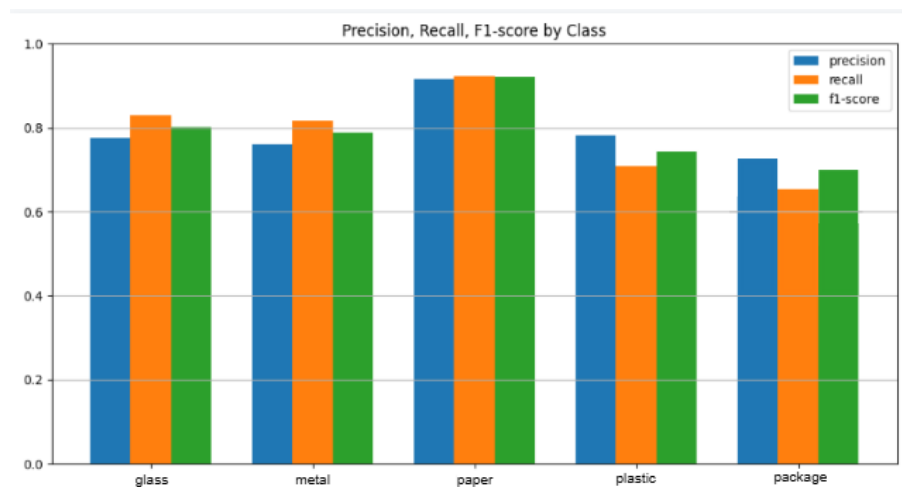


Рисунок 3.4 - Порівняння метрик precision, recall та F1-score для кожного класу класифікації відходів

Це графічне зображення результатів класифікації п'яти класів сміття: glass, metal, package, paper, plastic. Кожен елемент показує кількість зображень з фактичним класом по осі Y (True) і передбаченим класом по осі X (Predicted) (рисунок 3.5).

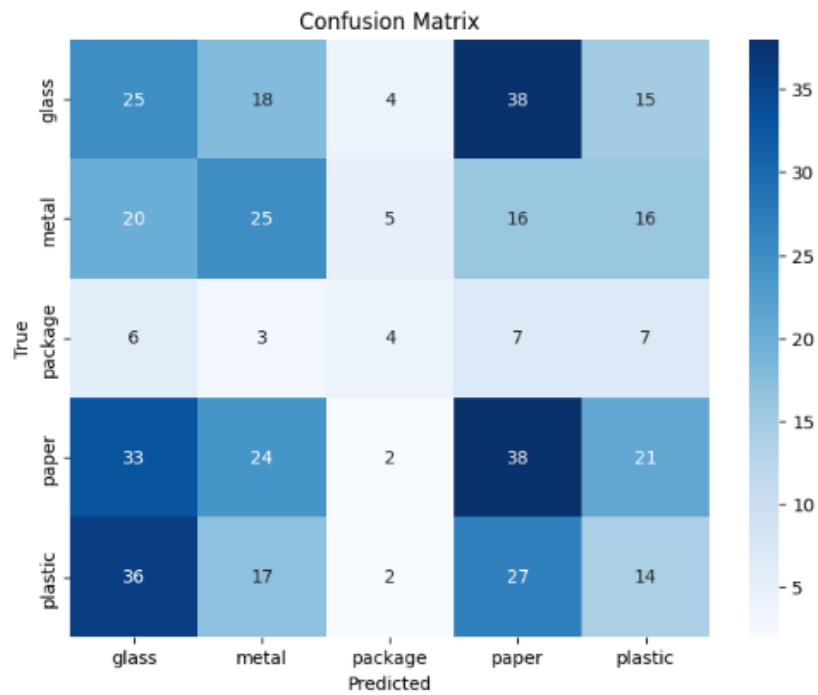


Рисунок 3.5 - Матриця помилок для класифікації типів сміття

Завдяки такому підходу система може виконувати класифікацію сміття в режимі реального часу, визначаючи, чи є об'єкт сміттям, а також його тип. Вона також враховує рівень впевненості у правильності класифікації, що дозволяє зробити систему більш точною та надійною. У результаті, навчання та тестування моделі на валідаційних даних забезпечили досягнення високої точності класифікації, що підтверджується графіками точності та втрат на різних етапах навчання.

Дані технології разом створюють потужний інструментарій для вирішення задач комп'ютерного зору, таких як класифікація об'єктів та їх відстеження в реальному часі. Використання Python та таких бібліотек, як TensorFlow і OpenCV, дозволяє ефективно реалізувати складні проєкти машинного навчання і зробити систему швидкою та зручною для розробки та впровадження. Програмний код

інтерактивного інтерфейсу застосунку для класифікації відходів наведено у додатку Б.

3.2 Інтерфейс користувача

На рисунку 3.6 показано інтерфейс користувача програми для класифікації відходів у режимі реального часу через камеру. Інтерфейс оформлений у вікні з назвою "Waste Classifier", яка відображається на екрані екрана.

У верхній частині вікна розміщено три кнопки: зелена кнопка "Старт", жовта "Повторити" та червона "Вихід".

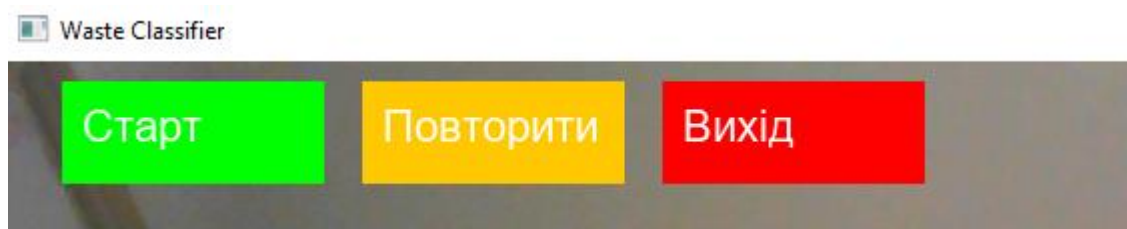


Рисунок 3.6 - Головне вікно застосунку з кнопками керування

При натисканні на кнопку "Старт" активується режим розпізнавання. У цьому режимі користувачу потрібно вибрати зону класифікації на екрані за допомогою миші (рисунок 3.7).

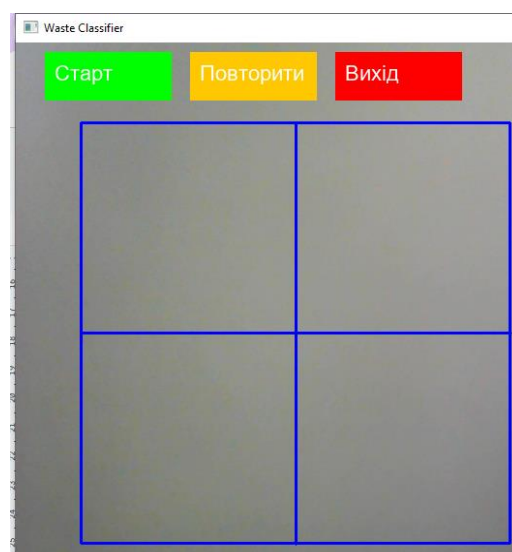


Рисунок 3.7 - Вибір зони об'єкта для класифікації

Після вибору області, на екрані з'являється повідомлення з інструкцією "Підтвердіть зону натиснувши ENTER" (рисунок 3.8). Після цього необхідно натиснути клавішу Enter, щоб система почала аналіз обраної області. У правому верхньому куті зображення виводиться результат класифікації.

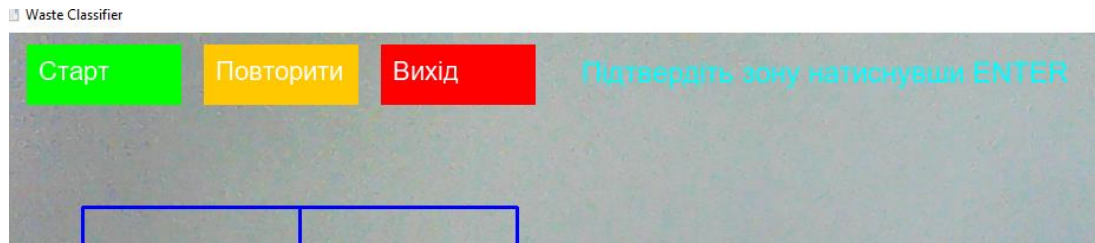


Рисунок 3.8 - Повідомлення в інтерфейсі застосунку з інформацією про підтвердження зони класифікації

На рисунку 3.9 наведений приклад, де система визначила, що об'єкт належить до класу "package" з ймовірністю 0.97, що свідчить про високу достовірність моделі у своїх передбаченнях.

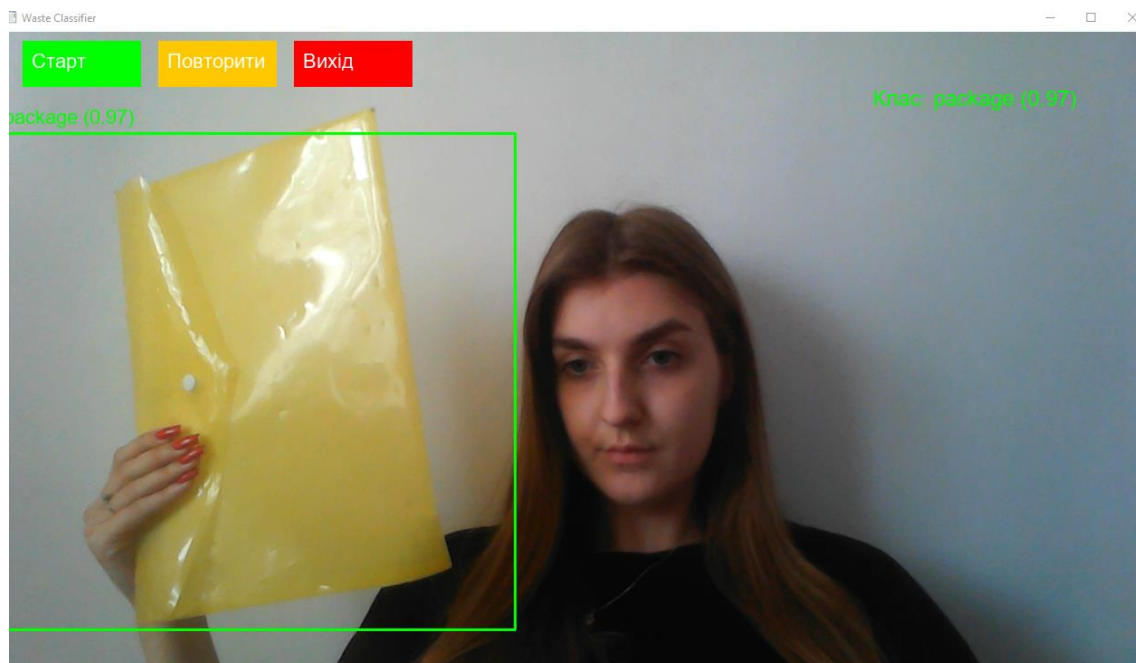


Рисунок 3.9 - Результат класифікації упаковки із виведенням ймовірності

На екрані також зображена зелена прямокутна рамка, яка виділяє об'єкт - у цьому випадку жовтий пакувальний матеріал. Рамка вказує на зону, яку вибрав

користувач для аналізу, і дає візуальне підтвердження, що система обробляє саме цю частину зображення.

Інтерфейс програми має простий і зрозумілий дизайн, що забезпечує комфортне користування навіть для нових користувачів. Кольорові кнопки з чітким позначенням основних функцій дозволяють легко взаємодіяти з системою, не вимагаючи додаткового навчання. Крім того, на екрані виводяться візуальні елементи, такі як рамка навколо об'єкта та текстовий підпис з передбачуваним класом, що допомагають користувачеві чітко орієнтуватися у процесі класифікації. Це дозволяє швидко і точно взаємодіяти з програмою, навіть якщо користувач не знайомий з подібними інтерфейсами. Програма орієнтована на зручність та інтуїтивність, що робить її доступною для широкого кола користувачів, незалежно від їхнього досвіду.

3.3 Тестування розробленої програми

Після завершення етапу реалізації модуля класифікації відходів було проведено всебічне тестування з метою оцінки його функціональності, точності класифікації, стабільності роботи та зручності взаємодії користувача з інтерфейсом. Тестування проводилось в умовах, наближених до реального використання.

Метою тестування перевірити працездатність модуля, виявити можливі недоліки в його роботі та оцінити відповідність заявленим функціональним характеристикам у різних умовах використання.

Основними цілі тестування:

- перевірка точності класифікації зображень на основі навченої моделі глибокого навчання;
- оцінка стабільності та продуктивності програми під час роботи в реальному часі;
- перевірка взаємодії користувача з графічним інтерфейсом.

Для тестування були використані зображення з набору даних, які не входили до навчальної вибірки. Вони охоплювали різні типи відходів, зокрема:

- пластикові пляшки;
- паперові упаковки;
- металеві банки.

Також здійснювалося тестування на об'єктів у реальному часі через веб-камеру.

Процес тестування був здійснений з використанням реалізованого графічного інтерфейсу, що включає інтерактивні кнопки: «Старт», «Повторити» та «Вихід». На початку було запущено програму, після чого було натиснуто кнопку «Старт», яка активувала камеру та ініціювала роботу системи розпізнавання об'єктів у режимі реального часу. Далі проводилася серія прикладів, які демонструють здатність моделі точно ідентифікувати об'єкти, що належать до визначених класів, або виявляти сторонні предмети.

Перший приклад зображено на рисунку 3.10, де перед камерою було розміщено аркуш паперу. Система успішно класифікувала об'єкт як клас «paper» з ймовірністю 0.97, що вказує на високу точність передбачення моделі. На екрані, у правому верхньому куті, з'явився відповідний клас та числове значення ймовірності.

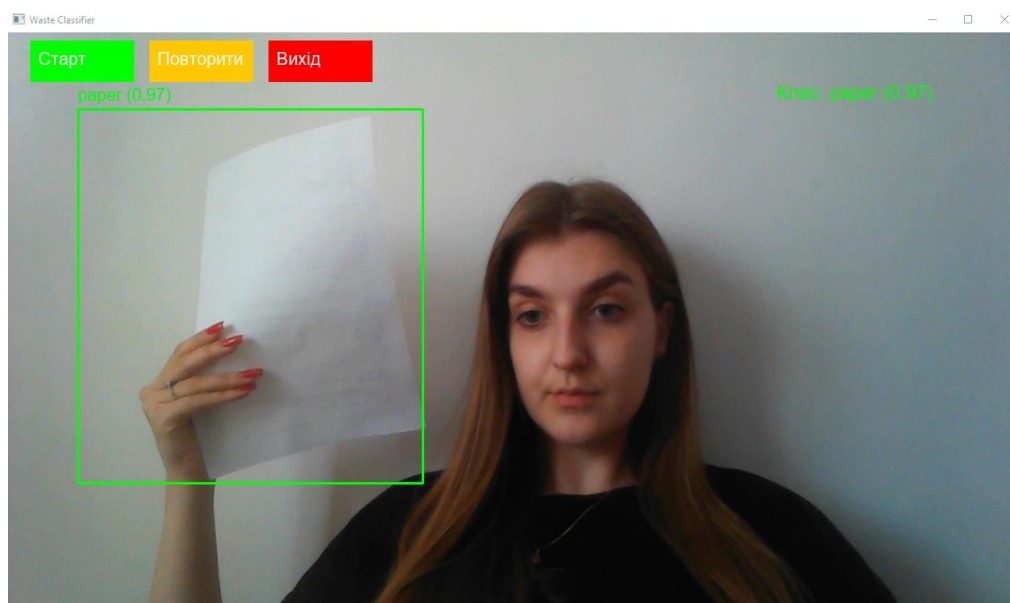


Рисунок 3.10 - Результат класифікації аркуша із виведенням ймовірності

Після першої спроби було натиснуто кнопку «Повторити», яка дозволила перейти до нового прикладу без перезавпуску програми. У цьому випадку до кадру був поміщений об'єкт, що не належить до жодного з визначених класів. Модель успішно розпізнала, що об'єкт не класифікується як один з відомих класів, і позначила його як «Сторонній об'єкт». Окрім цього, на екрані з'явилося текстове повідомлення, яке інформує про присутність у кадрі невизначеного об'єкта.

Такий механізм дає змогу уникнути помилкової класифікації у випадках, коли в поле зору потрапляють сторонні предмети або люди. Завдяки цьому підвищується загальна надійність роботи системи. Відповідний приклад проілюстровано на рисунку 3.11.

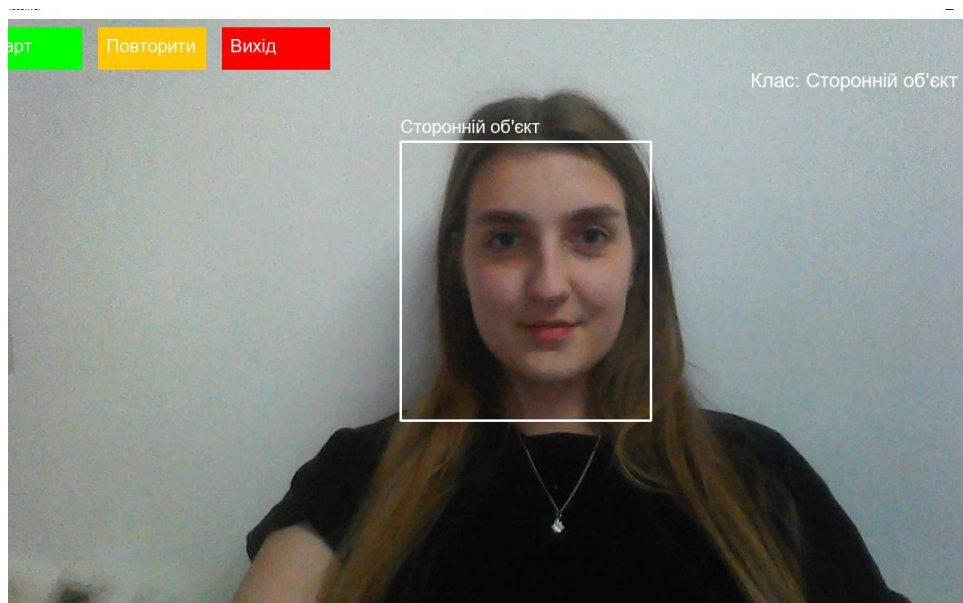


Рисунок 3.11 - Результат класифікації стороннього об'єкту

У наступному тестовому випадку, після повторного натискання кнопки «Повторити», у поле зору камери був розміщений скляний об'єкт. Система автоматично здійснила обробку зображення та успішно класифікувала об'єкт як «glass». Разом із результатом класифікації було виведено значення ймовірності, яке дозволяє оцінити ступінь упевненості моделі у прийнятому рішенні. Такий підхід підвищує прозорість роботи системи та дозволяє користувачу краще зрозуміти рівень точності. Даний приклад ілюстровано на рисунку 3.12.

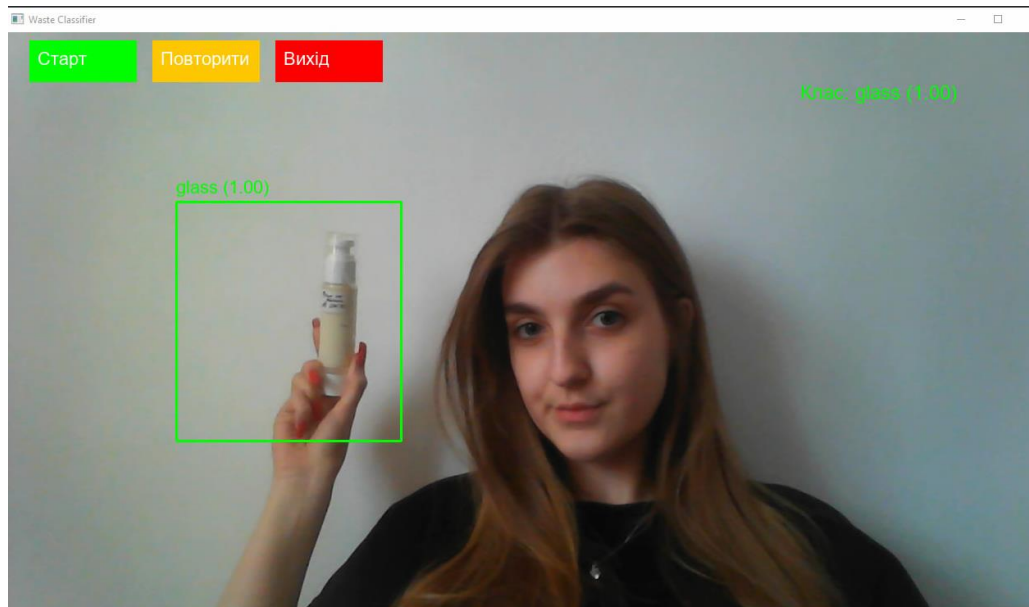


Рисунок 3.12 - Результат класифікації скляного об'єкту із виведенням ймовірності

На рисунку 3.13 зображено приклад, який демонструє класифікацію металевого об'єкта. Після натискання кнопки «Повторити», у камеру було подано об'єкт із металу. Система безпомилково віднесла його до класу «metal» з ймовірністю 0.90, що знову супроводжувалося виведенням відповідного значення на екрані у правому верхньому куті.

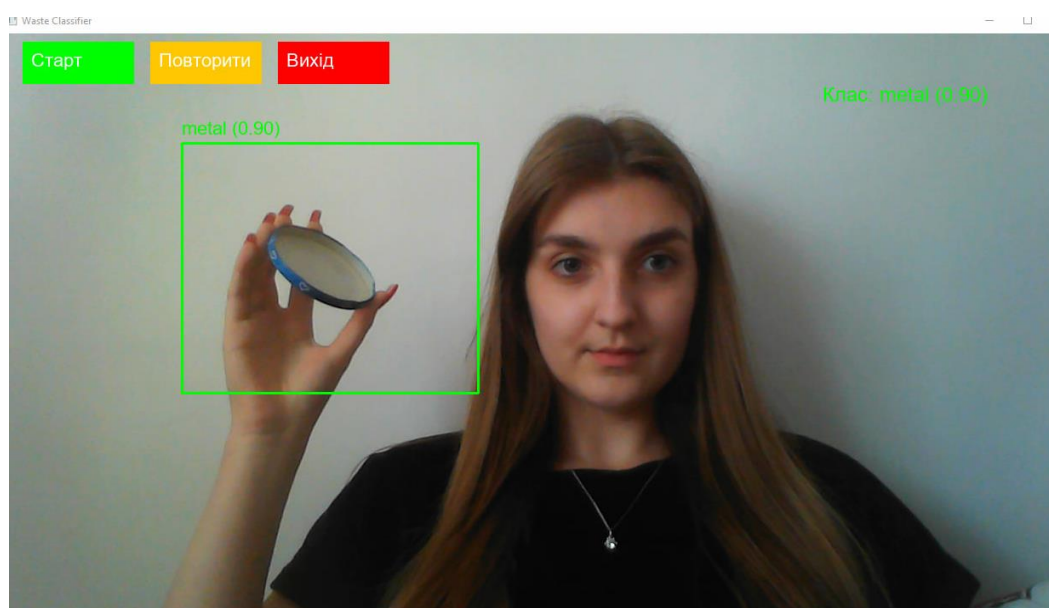


Рисунок 3.13 - Результат класифікації металевого об'єкту із виведенням ймовірності

Після завершення серії тестувань, для завершення роботи програми, було натиснуто кнопку «Вихід», яка зупинила відеопотік з камери та закрила програму.

Таким чином, у процесі тестування було перевірено не лише коректність класифікації об'єктів, а й зручність та ефективність взаємодії з користувацьким інтерфейсом, що дозволяє гнучко та інтуїтивно здійснювати розпізнавання декількох об'єктів без повторного запуску програми.

Для оцінки ефективності розробленого програмного модуля були проаналізовані ключові показники точності, швидкодії та стабільності роботи системи при класифікації зображень.

Результати тестування показали такі характеристики системи:

- середня точність класифікації склала 96%;
- час обробки одного зображення - у межах 1-2 секунд, що є прийнятним для інтерактивної роботи;
- усі елементи інтерфейсу працювали згідно з функціональними вимогами:
 - 1) кнопка «Почати» активувала відеозйомку;
 - 2) «Повторити» дозволяла оновити кадр;
 - 3) «Вихід» коректно завершував роботу програми.

Проведене тестування підтвердило, що розроблена програма працює стабільно, правильно класифікує основні типи відходів та є придатною до використання у навчальних цілях. У подальшому можливе покращення моделі для додавання нових категорій відходів та вдосконалення інтерфейсу користувача.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було досягнуто мету - створення модуля класифікації відходів на основі методів глибокого навчання, здатного розпізнавати та класифікувати об'єкти у режимі реального часу.

У процесі роботи було виконано наступне:

1. Проведено детальний аналіз теоретичних аспектів, визначено ключові екологічні проблеми, спричинені неефективним сортуванням, та обґрунтовано доцільність застосування глибокого навчання. Згідно з літературними джерелами, сучасні автоматизовані системи дозволяють підвищити ефективність сортування до 90% у промислових рішеннях.

2. Огляд сучасних підходів показав, що точність традиційних методів (SVM, Random Forest) зазвичай не перевищує 60-75%, тоді як згорткові нейронні мережі дозволяють підвищити точність розпізнавання до 95-96%. Особливо ефективним виявилось використання трансферного навчання на основі архітектури MobileNetV2.

3. Створено програмну архітектуру модуля, що включає попередню обробку, класифікацію та формування звітів, а також реалізовано реляційну базу даних для збереження результатів. Система підтримує класифікацію п'яти основних типів відходів із забезпеченням збалансованого розподілу даних.

4. Розроблено інтерактивний графічний інтерфейс, який дозволяє проводити класифікацію у реальному часі. У ході реалізації забезпечено простоту та інтуїтивність взаємодії - вікно програми містить 3 основні кнопки керування та вивід ймовірності класифікації (приклади: для класів "rare" та "package" отримано ймовірності 0.97).

5. Проведено комплексне тестування, під час якого середня точність класифікації досягла 96% на валідаційних даних, а час обробки одного зображення становив 1-2 секунди, що відповідає вимогам до роботи в реальному часі. Усі основні функції (класифікація, оновлення кадру, завершення роботи) працювали

стабільно, що підтверджено результатами тестування на реальних об'єктах різних класів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kaza S., Yao L. C., Bhada-Tata P., Van Woerden F. What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050. *World Bank Publications*. 2018. URL: <https://doi.org/10.1596/978-1-4648-1329-0> (дата звернення: березень 2025).
2. De Groof V., Coma M., Arnot T., Leak D. J., Lanham A. B. Selecting fermentation products for food waste valorisation with HRT and OLR as the key operational parameters. *Waste Management*. 2021. С. 80-89. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0956053X2100221X> (дата звернення: березень 2025).
3. AlexNet: ImageNet Classification with Deep Convolutional Neural Networks. 2018. URL: <https://neurohive.io/en/popular-networks/alexnet-imagenet-classification-with-deep-convolutional-neural-networks/> (дата звернення: березень 2025).
4. VGG16: Deep Convolutional Networks for Large-Scale Image Recognition. 2018. URL: <https://neurohive.io/en/popular-networks/vgg16/> (дата звернення: березень 2025).
5. VGG-19 Convolutional Neural Network. 2025. URL: <https://blog.techcraft.org/vgg-19-convolutional-neural-network/> (дата звернення: березень 2025).
6. He K., Zhang, X., Ren S., Sun J. Deep Residual Learning for Image Recognition. *ArXiv preprint*. 2015. URL: <https://arxiv.org/abs/1512.03385> (дата звернення: березень 2025).
7. Howard A. G., Zhu, M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv preprint*. 2017. URL: <https://arxiv.org/abs/1704.04861> (дата звернення: травень 2025).
8. Mingxing Tan , Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Network. *ArXiv preprint*. 2019. URL: <https://arxiv.org/abs/1905.11946> (дата звернення: травень 2025).

9. Шеремет О. І., Садовий О. В. Метод опорних векторів (SVM). *Науковий журнал Державного університету телекомунікацій*. 2013. С. 13-17. URL: <https://www.dstu.dp.ua/Portal/Data/74/72/3st13-17.pdf> (дата звернення: березень 2025).
10. Cortes C., Vapnik V. Support-vector networks. *Machine Learning*. 1995. С. 273-297. URL: <https://link.springer.com/article/10.1007/BF00994018> (дата звернення: березень 2025).
11. Breiman L. Random Forests. *Machine Learning*. 2001. С. 5-32. URL: <https://link.springer.com/article/10.1023/A:1010933404324> (дата звернення: березень 2025).
12. Bircanoglu C., Atay M., Besdok E. Recycling waste classification using convolutional neural networks. *26th Signal Processing and Communications Applications Conference (SIU)*. 2018. URL: https://www.researchgate.net/publication/325626219_RecycleNet_Intelligent_Waste_Sorting_Using_Deep_Neural_Networks (дата звернення: березень 2025).
13. Yang M., Thung G. Classification of trash for recyclability status. 2016. URL: <https://cs229.stanford.edu/proj2016/report/ThungYang-ClassificationOfTrashForRecyclabilityStatus-report.pdf> (дата звернення: березень 2025).
14. Jangsamsi K. Conventional Machine Learning Approach for Waste Classification. 2016. С. 7-12. URL: <https://dl.acm.org/doi/pdf/10.1145/3639592.3639594> (дата звернення: березень 2025).
15. Li L., Wang R., Zou M., Guo F., Ren Y. Enhanced ResNet-50 for garbage classification: Feature fusion and depth-separable convolutions. 2025. URL: <https://doi.org/10.1371/journal.pone.0317999> (дата звернення: березень 2025).
16. Машина для сортування твердих побутових відходів. URL: <https://www.bestongroup.com/uk/waste-sorting-machine/> (дата звернення: березень 2025).
17. Екологія та покращення. URL: <https://konsort.com.ua/productcat/ecology-and-improvement/> (дата звернення: березень 2025).

18. MobileNetV2 preprocess_input function. TensorFlow API documentation. *TensorFlow*. 2024. URL: https://www.tensorflow.org/api_docs/python/tf/keras/applications/mobilenet_v2/preprocess_input (дата звернення: травень 2025).
19. Howard A. G., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv preprint*. 2017. URL: <https://arxiv.org/abs/1704.04861> (дата звернення: травень 2025).
20. Scherer D., Muller A., Behnke S. Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. In Proceedings of the 20th International Conference on Artificial Neural Networks. *Springer Nature Link*. 2010. URL: https://link.springer.com/chapter/10.1007/978-3-642-15825-4_23 (дата звернення: квітень 2025).
21. Glorot X., Bordes A., Bengio Y. Deep Sparse Rectifier Neural Networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2011. С. 315-323. URL: <https://proceedings.mlr.press/v15/glorot11a/glorot11a.pdf> (дата звернення: квітень 2025).
22. Bishop C. M. Pattern Recognition and Machine Learning. *Springer*. 2025. URL: <https://link.springer.com/book/9780387310732> (дата звернення: квітень 2025).
23. Goodfellow I., Bengio Y., Courville A. Deep Learning. *MIT Press: Loss Functions for Classification*. 2016. URL: <https://www.deeplearningbook.org/> (дата звернення: травень 2025).
24. Machine learning approach in predicting GlutoPeak test parameters from image data with AutoML and transfer learning. 2023. URL: <https://www.cell.com/heliyon/pdf/S2405-8440%2823%2907730-7.pdf> (дата звернення: травень 2025).
25. Kingma D. P., Ba J. L. Adam: A Method for Stochastic Optimization. *ArXiv preprint*. 2014. URL: <https://arxiv.org/pdf/1412.6980> (дата звернення: квітень 2025).

26. Raschka S., Mirjalili V. Machine Learning in Python: Main Developments and Technology Trends in Data Science Deep Learning, and Artificial Intelligence. *MDPI Information*. 2020. URL: <https://www.mdpi.com/2078-2489/11/4/193> (дата звернення: квітень 2025).

27. Abadi M., Barham P., Chen J., Chen Z., Davis A., Dean J., Zheng X. TensorFlow: A system for large-scale machine learning. *ArXiv preprint*. 2016. URL: <https://arxiv.org/abs/1605.08695> (дата звернення: травень 2025).

28. Lukezic A., Vojir T., Cehovin Zajc L., Matas J., Kristan M. Discriminative Correlation Filter with Channel and Spatial Reliability. 2017. URL: https://openaccess.thecvf.com/content_cvpr_2017/papers/Lukezic_Discriminative_Correlation_Filter_CVPR_2017_paper.pdf (дата звернення: травень 2025).

29. Созанська Т. М., Ліп'яніна-Гончаренко Х. В. Модуль класифікації відходів на основі глибокого навчання. *Інтелектуальні інформаційні технології в прикладних дослідженнях: збірник тез доп. студент. наук.-практ. конф. (ІТAR-2025)*, м. Тернопіль, 27-29 травня 2025 р. Тернопіль: ЗУНУ, 2025. С. 81-83.

30. Комар М.П., Саченко А.О., Васильків Н.М., Гладій Г.М., Коваль В.С., Ліп'яніна-Гончаренко Х.В. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за першим (бакалаврським) рівнем вищої освіти. Тернопіль: ЗУНУ, 2024. С.52.

Додаток А

Лістинг коду моделі класифікації відходів

```
import os
import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras import layers, models
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
base_dir = r'C:\Users\TANYA\Desktop\duplom\data'
img_height, img_width = 224, 224
batch_size = 32
num_classes = 5
datagen = ImageDataGenerator(
    preprocessing_function=preprocess_input,
    validation_split=0.2
)
train_generator = datagen.flow_from_directory(
    base_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)
val_generator = datagen.flow_from_directory(
    base_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
```

```

)
base_model = MobileNetV2(
    input_shape=(img_height, img_width, 3),
    include_top=False,
    weights='imagenet'
)
base_model.trainable = False
model = models.Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.4),
    layers.Dense(num_classes, activation='softmax')
])
model.compile(
    optimizer=tf.keras.optimizers.Adam(learning_rate=0.0005),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)
epochs = 20
history = model.fit(
    train_generator,
    validation_data=val_generator,
    epochs=epochs
)
model.save("waste_classifier_mobilenetv2.keras")

```


Додаток Б

Лістинг коду реалізації інтерактивного інтерфейсу застосунку з класифікацією
відходів

```
import cv2
import numpy as np
import tensorflow as tf
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from PIL import ImageFont, ImageDraw, Image
FONT_PATH = "C:\\Windows\\Fonts\\arial.ttf"
model =
tf.keras.models.load_model("C:\\Users\\TANYA\\Desktop\\duplom\\waste_classifier_m
obilenetv2.keras")
class_labels = ['cardboard', 'glass', 'metal', 'package', 'paper', 'plastic']
CONFIDENCE_THRESHOLD = 0.85
def draw_ukr_text(frame, text, position, font_path=FONT_PATH, font_size=24,
color=(255, 255, 255)):
    img_pil = Image.fromarray(frame)
    draw = ImageDraw.Draw(img_pil)
    font = ImageFont.truetype(font_path, font_size)
    draw.text(position, text, font=font, fill=color)
    return np.array(img_pil)
def preprocess_roi(roi):
    blurred = cv2.GaussianBlur(roi, (5, 5), 0)
    resized = cv2.resize(blurred, (224, 224))
    img = preprocess_input(resized)
    return np.expand_dims(img, axis=0)
cam_width, cam_height = 1280, 720
cap = cv2.VideoCapture(0)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, cam_width)
```

```

cap.set(cv2.CAP_PROP_FRAME_HEIGHT, cam_height)
tracking = False
tracker = None
initBB = None
start_detection = False
label = "Очікування..."
buttons = {
    "start": ((30, 10), (160, 60), (0, 255, 0), "Старт"),
    "repeat": ((180, 10), (310, 60), (0, 200, 255), "Повторити"),
    "exit": ((330, 10), (460, 60), (0, 0, 255), "Вихід")
}
info_box_position = (970, 60)
def is_inside(pos, top_left, bottom_right):
    x, y = pos
    return top_left[0] <= x <= bottom_right[0] and top_left[1] <= y <= bottom_right[1]
def draw_buttons(frame):
    for name, ((x1, y1), (x2, y2), color, text) in buttons.items():
        cv2.rectangle(frame, (x1, y1), (x2, y2), color, -1)
        frame = draw_ukr_text(frame, text, (x1 + 10, y1 + 10), font_size=22, color=(255,
255, 255))
    return frame
clicked_button = None
selecting_roi = False
while True:
    ret, frame = cap.read()
    if not ret:
        break
    frame = cv2.resize(frame, (cam_width, cam_height))
    frame = draw_buttons(frame)
    if selecting_roi:

```

```

    frame = draw_ukr_text(frame, "Підтвердіть зону натиснувши ENTER", (500,
20), font_size=24, color=(255, 255, 0))
    if start_detection:
        if not tracking:
            selecting_roi = True

            frame = draw_ukr_text(frame, "Підтвердіть зону натиснувши ENTER", (500,
20), font_size=24, color=(255, 255, 0))

            initBB = cv2.selectROI("Waste Classifier", frame, fromCenter=False,
showCrosshair=True)

            tracker = cv2.legacy.TrackerCSRT_create()
            tracker.init(frame, initBB)

            tracking = True
            selecting_roi = False
        success, box = tracker.update(frame)
        if success:
            x, y, w_box, h_box = [int(v) for v in box]
            roi = frame[y:y + h_box, x:x + w_box]
            color = (255, 255, 255)
            if roi.shape[0] > 0 and roi.shape[1] > 0:
                input_img = preprocess_roi(roi)
                predictions = model.predict(input_img, verbose=0)
                confidence = np.max(predictions)
                class_index = np.argmax(predictions)
                if confidence >= CONFIDENCE_THRESHOLD:
                    label = f"{class_labels[class_index]} ({confidence:.2f})"
                    color = (0, 255, 0)
                else:
                    label = "Сторонній об'єкт"
                    color = (0, 0, 255)
            cv2.rectangle(frame, (x, y), (x + w_box, y + h_box), color, 2)

```

```

        frame = draw_ukr_text(frame, label, (x, y - 30), font_size=22, color=color)
        frame = draw_ukr_text(frame, f"Клас: {label}", info_box_position,
font_size=24, color=color)
    else:
        label = "Втрачено об'єкт"
        tracking = False
cv2.imshow("Waste Classifier", frame)
key = cv2.waitKey(1) & 0xFF
if key == 27:
    break
if cv2.getWindowProperty("Waste Classifier", cv2.WND_PROP_VISIBLE) < 1:
    break
def mouse_callback(event, x, y, flags, param):
    global clicked_button, start_detection, tracking
    if event == cv2.EVENT_LBUTTONDOWN:
        for name, ((x1, y1), (x2, y2), _, _) in buttons.items():
            if is_inside((x, y), (x1, y1), (x2, y2)):
                clicked_button = name
                if name == "start" or name == "repeat":
                    start_detection = True
                    tracking = False
                elif name == "exit":
                    cap.release()
                    cv2.destroyAllWindows()
cv2.setMouseCallback("Waste Classifier", mouse_callback)
cap.release()
cv2.destroyAllWindows()

```

Додаток В
Апробація отриманих результатів

Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління



ЗБІРНИК ТЕЗ ДОПОВІДЕЙ

Студентської науково-практичної конференції
ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ПРИКЛАДНИХ
ДОСЛІДЖЕННЯХ
(ІІТАР-2025)

27-29 травня 2025 року

Тернопіль
2025

Дідур Павло, Сапожник Григорій	45
РОЗПІЗНАВАННЯ РУКОПИСНИХ СИМВОЛІВ НА ОСНОВІ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ	45
Єднак Іван, Домбровський Михайло	49
МОДУЛЬ EDGE-ОБЧИСЛЕНЬ ДЛЯ ЦИФРОВОГО РОЗВИТКУ ІННОВАЦІЙНОГО ВИРОБНИЦТВА	49
Ковальчук Ярослав	52
СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ НА РИНКУ СТРАХОВИХ ПОСЛУГ НА ОСНОВІ МАШИННОГО НАВЧАННЯ	52
Крушельницький Святослав	55
ІНТЕЛЕКТУАЛЬНА СИСТЕМА АНАЛІЗУ НАСТРОЮ ТЕКСТОВИХ ВІДГУКІВ	55
Кутереба Іван	58
ВІЯВЛЕННЯ ТА КЛАСИФІКАЦІЯ ДЕФЕКТІВ ФАРБУВАННЯ АВТОМОБІЛІВ НА ОСНОВІ ГЛИБОКОЇ НЕЙРОННОЇ МЕРЕЖІ	58
Лучечко Христина, Сапожник Григорій	62
СЕМАНТИЧНИЙ АНАЛІЗ ТЕКСТОВИХ ДАНИХ ІЗ ВИКОРИСТАННЯМ МОДЕЛЕЙ ВЕКТОРНОГО ПРЕДСТАВЛЕННЯ СЛІВ	62
Мураль Роман, Лендюк Тарас	65
МОДУЛЬ АНАЛІЗУ НАСТРОЇВ У ВІДГУКАХ ПРО ПРОДУКТИ ХАРЧУВАННЯ НА ОСНОВІ WORD2VEC	65
Никоряк Денис, Майків Ігор	68
ВЕБ-ОРІЄНТОВАНА СИСТЕМА ДЛЯ АНАЛІЗУ ТА ОПТИМІЗАЦІЇ МАРШРУТІВ ЗАМОВЛЕНЬ ІЗ ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ WEB WORKER	68
Оберванюк Уляна, Осолінський Олександр	71
КОНЦЕПЦІЯ МОДУЛЯ МОНІТОРИНГУ РОБОТИ ЕЛЕКТРОДВИГУНА З ВИКОРИСТАННЯМ ІОТ	71
Полюшко Денис, Дорош Віталій	74
ВІЯВЛЕННЯ ВТОРГНЕНЬ НА ОСНОВІ ІНТЕГРАЦІЇ ТЕХНОЛОГІЙ ГЛИБОКОГО НАВЧАННЯ ТА АНАЛІТИКИ ВЕЛИКИХ ДАНИХ	74
Проць Анастасія	78
НЕЙРОМЕРЕЖЕВА ПРОГРАМНА СИСТЕМА РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ ВОГНЮ ТА ДИМУ	78
Созанська Тетяна, Ліп'яніна-Гончаренко Христина	81
МОДУЛЬ КЛАСИФІКАЦІЇ ВІДХОДІВ НА ОСНОВІ ГЛИБОКОГО НАВЧАННЯ	81

Созанська Тетяна
студент групи КН-42
tetianasozanska@gmail.com
Ліп'яніна-Гончаренко Христина
д.т.н., доцент
kh.lipianina@wunu.edu.ua
Західноукраїнський національний університет
Тернопіль, Україна

МОДУЛЬ КЛАСИФІКАЦІЇ ВІДХОДІВ НА ОСНОВІ ГЛИБОКОГО НАВЧАННЯ

Автоматизація сортування відходів є важливим напрямом екологічних інновацій, що дозволяє зменшити вплив людського фактору, оптимізувати ресурси та підвищити ефективність переробки. Метою дослідження є створення модуля класифікації відходів на основі глибокого навчання з використанням згорткових нейронних мереж. Основні завдання роботи включають розробку архітектури моделі, обробку даних, побудову інтерфейсу та тестування системи в реальному часі. Обрана секція повністю відповідає тематиці, оскільки дослідження стосується практичного застосування ШІ у виробничих та екологічних системах.

Сучасні дослідження демонструють високі досягнення у використанні глибоких нейронних мереж для класифікації твердих побутових відходів. Одним із підходів є вдосконалення моделей на основі EfficientNetV2, зокрема, інтеграція модуля Channel-Efficient Attention та SAFM, що дозволяє досягти точності 95.4%. [1] Інший приклад — використання ResNet50, яке показало точність понад 94,13% на датасеті TrashNet, ефективно витягуючи ознаки навіть за обмежених умов даних. Ці методи демонструють переваги глибоких архітектур для сортування сміття в різноманітних умовах. [2]

Ще одним важливим напрямом є використання моделей для класифікації сміття в реальному часі, таких як MobileNet. Завдяки своїй обчислювальній ефективності, MobileNet можна легко реалізувати на смартфонах або вбудованих пристроях, що робить його ідеальним для систем розумного сортування. Зокрема, у поєднанні з фреймворками, як TensorFlow Lite, ця модель дозволяє швидко й точно класифікувати об'єкти без великих вимог до ресурсів. Цей підхід ідеально підходить для створення мобільних екологічних рішень, доступних широкому колу користувачів. [3]

З огляду на ці переваги, у рамках дослідження було реалізовано власний модуль класифікації відходів на основі глибокої згорткової нейронної мережі MobileNetV2, оптимізованої для мобільних і вбудованих рішень. Перед подачею зображень до моделі виконується попередня обробка, що включає масштабування, нормалізацію до інтервалу $[-1; 1]$ та аугментацію: обертання, зсув, зміна яскравості. Така підготовка покращує узагальнення моделі на нових даних. Було застосовано функцію втрат. Для багатокласової класифікації застосовується категоріальна кросентропія (Categorical Crossentropy).

Це стандартна функція втрат для задач класифікації з декількома класами. Вона вимірює розбіжність між фактичним розподілом міток класів та передбаченим розподілом ймовірностей [4].

$$L = \sum_{i=1}^K y_i \log(\hat{y}_i) \quad (1.1)$$

де L — значення функції втрат;

y_i — фактична мітка класу (0 або 1);

$\hat{y}_i$ — ймовірність, передбачена мережею для класу i ;

K — кількість класів.

Архітектура моделі включає базовий шар MobileNetV2 без top-шарів, після чого додаються шари GlobalAveragePooling2D, Dense (128, ReLU), Dropout(0.4) та вихідний шар Dense (5, softmax). Це дозволяє моделі виводити ймовірності належності об'єкта до п'яти класів: пластик, скло, папір, метал, упаковка. Для оптимізації застосовується алгоритм Adam з початковою швидкістю навчання 0.0001. Усі етапи навчання реалізовано в TensorFlow 2.0 із модулем Keras. Схема модуля класифікації відходів представлена на рисунку 1.



Рисунок 1 - Схема модуля класифікації відходів

Крім класифікації, у системі реалізовано механізм трекінгу об'єктів у реальному часі за допомогою алгоритму CSRT (Discriminative Correlation Filter with Channel and Spatial Reliability). Це дозволяє виділяти об'єкти у відеопотоці камери та динамічно передавати їх на класифікацію. Інтерфейс користувача реалізовано засобами OpenCV і містить кнопки «Старт», «Повторити», «Вихід» та інтерактивну зону вибору.

Для оцінки ефективності моделі було проведено серію експериментів на валідаційному наборі зображень, які не входили до тренувального датасету. Модель MobileNetV2 досягла середньої точності класифікації 96% при п'яти класах відходів.

На рисунку 2 подано графік точності під час. Він демонструє стабільне зростання точності з кожною епохою, що свідчить про поступове навчання моделі.

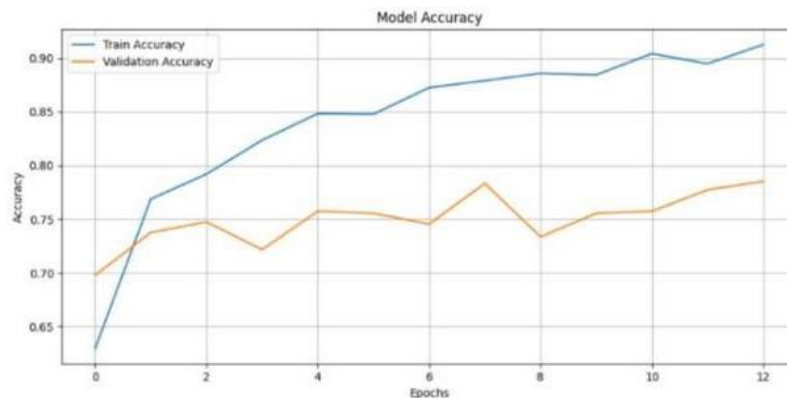


Рисунок 2 - Графік точності навчання та валідації

Водночас, на рисунку 3 подано графік функції втрат, що демонструє поступове зниження значення втрат із кожною епохою, з незначними коливаннями на валідаційній вибірці. Така динаміка підтверджує, що модель здатна узагальнювати знання і демонструє добру узгодженість між навчальними та валідаційними даними.

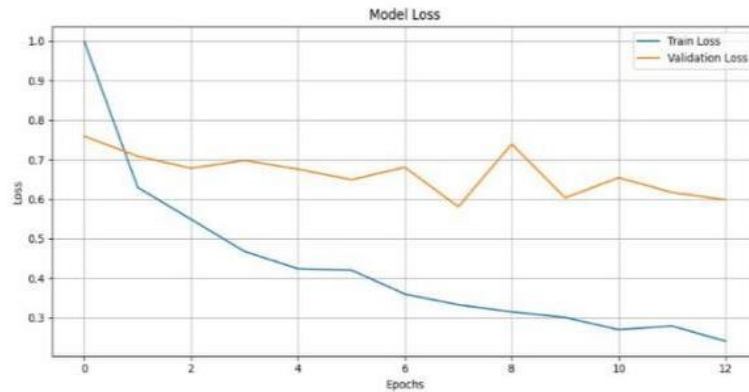


Рисунок 3 - Графік втрат моделі

Окрім узагальнених числових метрик, ефективність моделі була перевірена в умовах реального часу на тестових прикладах. На рисунку 4 зображено результат роботи системи, коли перед камерою було розміщено аркуш паперу. Модуль автоматично розпізнав об'єкт і класифікував його як представника класу «paper» з ймовірністю 0.97. Такий високий показник свідчить про високу впевненість моделі у прийнятому рішенні та демонструє ефективність реалізованого методу класифікації. Візуалізація результату — назва класу та відповідне значення ймовірності — була виведена на екран у правому верхньому куті, що додатково дозволяє користувачу оперативно оцінити результати роботи системи в режимі реального часу.

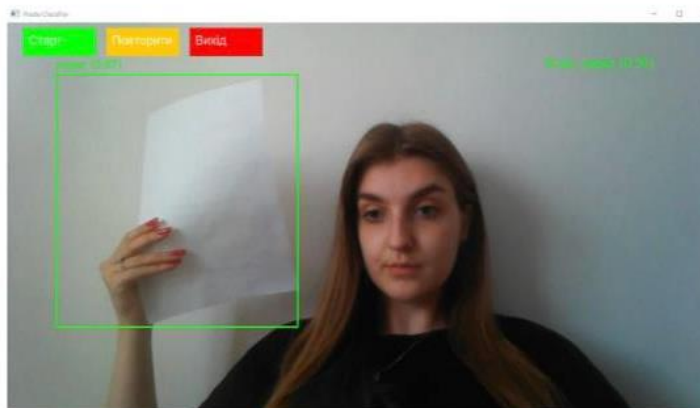


Рисунок 4 - Результат класифікації аркуша із виведенням ймовірності

У наступному тестовому випадку, наведеному на рисунку 5, після активації повторної класифікації в поле зору камери було розміщено скляний об'єкт. Система автоматично здійснила обробку зображення, класифікувала його як «glass» та вивела значення ймовірності. Цей приклад демонструє коректну роботу модуля класифікації та вбудованого механізму оцінки достовірності результатів.

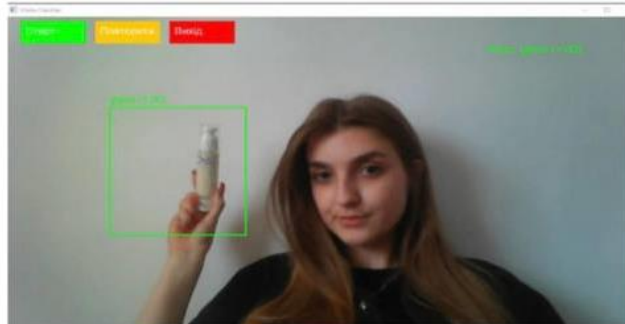


Рисунок 5 - Результат класифікації скляного об'єкту із виведенням ймовірності

У межах дослідження розроблено модуль класифікації зображень відходів у реальному часі на основі згорткової нейронної мережі MobileNetV2. Проведено навчання на збалансованому датасеті та тестування з досягненням високої точності класифікації, а саме в середньому - 96%. Система підтримує інтерактивну роботу з відеопотоком і може бути використана в мобільних або вбудованих рішеннях. Результати дослідження демонструють ефективність ШІ у задачах промислового сортування сміття. У майбутньому планується розширення кількості класів, вдосконалення системи трекінгу та впровадження розробленого модуля у промислові сортувальні комплекси.

Список використаних джерел

1. Qiu W., Xie C., Huang J. *An improved EfficientNetV2 for garbage classification*. 2025. URL: <https://arxiv.org/abs/2503.21208>.
2. Li, L., Wang, R., Zhou, M., Go, F., & Ren, Y. *Efficient deep learning models for waste classification: An analysis of multiple approaches*, 2025. URL: <https://doi.org/10.1371/journal.pone.0317999>.
3. Tsang, S.-H. *MobileNetV2 – Light Weight Model for Image Classification*, 2025. URL: <https://medium.com/data-science/review-mobilenetv2-light-weight-model-image-classification-8febb490e61c>.
4. Heliyon. *Machine learning approach in predicting GlutoPeak test parameters from image data with AutoML and transfer learning*, 2023. URL: <https://www.cell.com/heliyon/pdf/S2405-8440%2823%2907730-7.pdf>.