

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Галузь знань 12 – Інформаційні технології

Спеціальність 123 – Комп'ютерна інженерія

на тему «Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень»

КвРКІП. 303210.23.03.02 ПЗ

Виконав: студент 2 курсу, група КІ2м-23-3

Керівник д-р. техн. наук, професор
Науковий ступінь, вчене звання

До захисту допускаю:
Зав. кафедри КІС, докто філософії, доцент
Ольга ПАВЛОВА
06 05 2025 р.


Валентин ТАЛАПЧУК
Ім'я, прізвище


Тетяна ГОВОРУЩЕНКО
Ім'я, прізвище

Хмельницький, 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЬО-НАУКОВА ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА



“ 01 ” 09 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

Валентину ТАЛАПЧУКУ

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень.

Керівник проекту (роботи) Тетяна ГОВОРУЩЕНКО, д.т.н., професор

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 08.01.2025 №8

2. Строк подання студентом проекту (роботи) на кафедру 01.05.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

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

Аналіз моделей, методів і засобів розпізнавання харчових алергенів



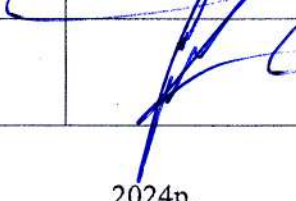

Концептуальне проектування кіберфізичної системи виявлення харчових алергенів

Метод виявлення харчових алергенів на основі машинного навчання та аналізу зображень

Проектування та розробка програмного забезпечення кіберфізичної системи виявлення харчових алергенів

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

6. Консультанти розділів кваліфікаційної роботи магістра

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Сергій ЛИСЕНКО, професор кафедри КПС		
Антиплагиат	Андрій НІЧЕПОРУК, доцент кафедри КПС		

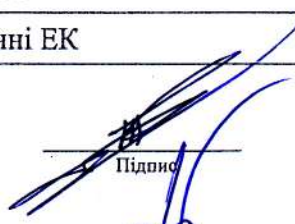

7. Дата видачі завдання « 01 » 09 2024р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи магістра	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напрямку дослідження та узгодження тематики КвРМ з керівником	01.09.2024	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.10.2024	виконано
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	01.11.2024	виконано
4	Робота над розділом 2 – розробка моделей для вирішення поставленої задачі	01.12.2024	виконано
5	Робота над науковою статтею	01.02.2025	виконано
6	Робота над розділом 3 – розробка методів для вирішення поставленої задачі	15.02.2025	виконано
7	Робота над розділом 4 – проектування та розробка ПЗ для вирішення поставленої задачі, експериментальна частина	01.04.2025	виконано
8	Оформлення пояснювальної записки згідно вимог	18.04.2025	виконано
9	Попередній захист ДРМ	29.04.2025	виконано
10	Захист ДРМ на засіданні ЕК	До 15.05.2025	

Студент

Керівник роботи


Підпис

Підпис

Валентин ТАЛАПЧУК

Ім'я, прізвище

Тетяна ГОВОРУШЕНКО

Ім'я, прізвище

РЕФЕРАТ

Тема кваліфікаційної роботи магістра: Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень.

Автор роботи: Валентин ГАЛАПЧУК

Керівник роботи: Тетяна ГОВОРУЩЕНКО

Пояснювальна записка: 81 с., 32 рис., 13 табл., 3 дод., 80 джерел.

ХАРЧОВІ АЛЕРГЕНИ, КІБЕРФІЗИЧНА СИСТЕМА, МАШИННЕ НАВЧАННЯ, КОМП'ЮТЕРНИЙ ЗІР, МОБІЛЬНИЙ ЗАСТОСУНОК, АНАЛІЗ ПРИРОДНОЇ МОВИ.

Об'єктом дослідження є процес автоматизації розпізнавання їжі та контролю її складу.

Предметом дослідження є конкретні методи та моделі, які забезпечують автоматичне виявлення алергенних інгредієнтів у харчових продуктах на основі аналізу зображень.

Метою кваліфікаційної роботи магістра є автоматичне виявлення харчових алергенів на основі методів машинного навчання та комп'ютерного зору.

Для розв'язання поставлених задач використовувалися методи машинного навчання, комп'ютерного зору, обробки природної мови, системного аналізу та емпіричного дослідження ефективності розробленої системи.

Наукова новизна отриманих результатів:

– набув подальшого розвитку метод автоматичного виявлення харчових алергенів, що поєднує класифікацію зображень, оптичне розпізнавання тексту та обробку природної мови в єдиному адаптивному алгоритмі;

– набула подальшого розвитку архітектура мобільно-орієнтованої кіберфізичної системи виявлення харчових алергенів, яка базується на гібридній архітектурі з розподіленим обчисленням і персоналізованою логікою аналізу вхідних даних.

На основі проведених досліджень розроблена архітектура і компоненти програмного забезпечення, які реалізують функціональність мобільно-орієнтованої кіберфізичної системи виявлення харчових алергенів. Побудовано класифікаційну модель, модуль OCR-аналізу, NLP-компонент для текстової обробки, а також серверну й клієнтську частини системи з підтримкою персоналізації.

Практична значимість отриманих результатів полягає у створенні функціонального програмного забезпечення, яке може бути безпосередньо використане людьми з харчовими алергіями для підвищення безпеки харчування у повсякденному житті.

У першому розділі було проаналізовано існуючі моделі та методи розпізнавання харчових продуктів, досліджено класифікацію алергенів та технології OCR і NLP для обробки текстової інформації.

У другому розділі спроектовано архітектуру кіберфізичної системи, що включає три рівні: фізичний для збору даних, аналітичний для обробки та інтерфейсний. Запропоновано гібридний підхід, що оптимально розподіляє обчислення між пристроєм та сервером, створюючи передумови для розробки комплексного методу аналізу.

У третьому розділі реалізовано метод виявлення харчових алергенів, що поєднує різні технології аналізу залежно від типу продукту. Розроблено ефективну модель класифікації зображень на базі VisionFeaturePrint, удосконалено модель AdaMine для розпізнавання готових страв та створено NLP-модуль з багаторівневим словником алергенів. Впроваджені оптимізації дозволили досягти високої точності (98.4%).

У четвертому розділі реалізовано програмне забезпечення кіберфізичної системи за архітектурою MVVM. Створено мобільний застосунок для iOS з інтуїтивним інтерфейсом, що забезпечує зручне сканування продуктів та відображення результатів аналізу. Проведено комплексне тестування, яке підтвердило високу ефективність системи при мінімальному використанні ресурсів пристрою.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	6
ВСТУП	7
1 АНАЛІЗ МОДЕЛЕЙ, МЕТОДІВ І ЗАСОБІВ РОЗПІЗНАВАННЯ ХАРЧОВИХ АЛЕРГЕНІВ.....	11
1.1 Класифікація харчових алергенів	11
1.2 Методи та технології автоматичного виявлення алергенів у харчових продуктах	13
1.3 Моделі машинного навчання, що використовуються для розпізнавання їжі	14
1.3.1 Класичні алгоритми машинного навчання	15
1.3.2 Глибокі нейронні мережі для класифікації харчових продуктів	18
1.3.3 Детекція та сегментація об'єктів у складних стравах	23
1.4 Використання OCR для отримання даних з етикеток.....	26
1.5 Технологічні засоби та інструменти, що застосовуються у сфері аналізу харчових продуктів	28
1.5.1 Огляд баз даних харчових продуктів та їх використання для навчання моделей.....	28
1.5.2 Комерційні та дослідницькі рішення у сфері розпізнавання їжі	30
1.6 Висновки до першого розділу	32
2 КОНЦЕПТУАЛЬНЕ ПРОЄКТУВАННЯ КІБЕРФІЗИЧНОЇ СИСТЕМИ ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ.....	34
2.1 Технологічні засоби реалізації системи	34
2.2 Загальна концепція системи	34
2.3 Архітектура мобільної кіберфізичної системи	37
2.4 Головні виклики при виявленні харчових алергенів	40

2.4.1 Помилки класифікації (Classification Errors)	40
2.4.2 Помилки детекції та сегментації.....	41
2.4.3 Проблеми OCR (оптичного розпізнавання тексту).....	41
2.4.4 Недоліки обробки природної мови (NLP) при виявленні алергенів	42
2.4.5 Обмеження навчальних вибірок	44
2.4.6 Вплив зовнішніх факторів	45
2.5 Висновки до другого розділу	46
3 МЕТОД ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ НА ОСНОВІ МАШИННОГО НАВЧАННЯ ТА АНАЛІЗУ ЗОБРАЖЕНЬ	48
3.1 Модель класифікації зображення	48
3.1.1 Формалізація задачі класифікації зображень	48
3.1.2 Підготовка навчальних для навчання класифікаційної моделі.....	50
3.1.3 Архітектура моделі.....	52
3.1.4 Метрики оцінювання ефективності класифікаційної моделі.....	55
3.2 Розгортання серверної частини для розпізнавання страв.....	58
3.2.1 Архітектура серверного компонента.....	59
3.2.2 Процес розгортання та конфігурації Docker-контейнера.....	61
3.2.3 API серверного компонента	62
3.3 Експериментальне дослідження ефективності серверної моделі	63
3.4 NLP-модуль для виявлення харчових алергенів	65
3.4.1 Формулювання задачі та загальна логіка роботи модуля	66
3.4.2 Структура вхідного тексту та типові проблеми після OCR.....	67
3.4.3 Структура словника алергенів та методи пошуку збігів	68
3.5 Висновки до третього розділу	69

4 ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КІБЕРФІЗИЧНОЇ СИСТЕМИ ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ	71
4.1 Вибір архітектури та технологій програмного забезпечення	71
4.2 Проєктування програмного забезпечення.....	73
4.2.1 UML-діаграма варіантів використання	74
4.2.2 UML-діаграма класів.....	75
4.2.3 UML-діаграма активностей	77
4.3 Реалізація мобільного застосунку.....	77
4.4 Інтерфейс мобільного застосунку.....	80
4.5 Тестування програмного забезпечення	87
4.6 Висновки до четвертого розділу	91
ВИСНОВКИ.....	93
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	96
ДОДАТОК А Стаття у фаховому науковому журналі.....	104
ДОДАТОК Б UML-діаграма активностей.....	111
ДОДАТОК В Презентація дипломної роботи.....	112

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

АПЗ – антивірусне програмне забезпечення

БД – база даних

МН – машинне навчання

НМ – нейронна мережа

ОР – оптичне розпізнавання

ПЗ – програмне забезпечення

CNN – згорткова нейронна мережа (англ. Convolutional Neural Network)

JSON – JavaScript Object Notation, формат обміну даними

NLP – обробка природної мови (англ. Natural Language Processing)

OCR – оптичне розпізнавання тексту (англ. Optical Character Recognition)

REST API – прикладний програмний інтерфейс передачі репрезентативного стану (Representational State Transfer)

UI/UX – інтерфейс користувача/досвід користувача (англ. User Interface/User Experience)

ВСТУП

Харчова алергія є глобальною проблемою, яка стосується мільйонів людей у всьому світі. Поширеність харчових алергій невідомо зростає і відсоток алергіків розвинених країнах сягає приблизно 10% населення [1]. Алергічні реакції можуть бути вкрай небезпечним та навіть найменша кількість алергену здатна викликати симптоми різної тяжкості – від легкого подразнення до анафілактичного шоку [2].

Щорічно реєструються численні випадки госпіталізацій через випадкове споживання заборонених інгредієнтів. Незважаючи на запроваджені заходи (обов'язкове маркування алергенів на продуктах, інформаційні кампанії для споживачів тощо), повністю убезпечити алергіків від ризику не вдається.

У реальних умовах люди з алергією змушені ретельно вивчати склад продуктів, але етикетки можуть бути написані дрібним текстом або написаними іноземною мовою, а при споживанні страв у закладах харчування інформація про інгредієнти часто недоступна. Отже, існує велика потреба в автоматизованих засобах контролю складу їжі. Забезпечення безпеки споживачів та врахування їхніх дієтичних потреб є надзвичайно важливим завданням сучасної харчової індустрії, тому системи виявлення харчових алергенів набули актуальності як інструмент для запобігання небажаним реакціям.

Сучасні технологічні рішення, такі як мобільні додатки для сканування штрих-кодів, пропонують лише часткове вирішення проблеми, оскільки вони обмежені продуктами в упаковках та залежать від точності зовнішніх баз даних. Дослідження 2020 року, яке проаналізувало 1376 мобільних додатків для людей з алергією, показало, що лише 14 з них були справді корисними для користувачів [2]. Більшість існуючих застосунків мають низьку функціональність та мають недоліки, які значно знижують ймовірність розпізнання алергенів. Таким чином, розробка мобільно-орієнтованої системи, що автоматично розпізнає харчові алергени за допомогою сучасних методів машинного навчання та комп'ютерного зору, є своєчасною і затребуваною. Це дозволить підвищити рівень безпеки

харчування для людей з алергіями та зменшити кількість випадкових алергічних реакцій.

Об'єктом дослідження є автоматизація процесу розпізнавання їжі та контролю її складу.

Предметом дослідження є конкретні методи та моделі, які забезпечують автоматичне виявлення алергенних інгредієнтів у харчових продуктах на основі аналізу зображень. Зокрема, це методи машинного навчання для аналізу зображень продуктів та їх пакування, алгоритми оптичного розпізнавання тексту (OCR) для вилучення інформації з етикеток, а також підходи до інтеграції цих компонентів у єдину мобільну систему. Іншими словами, предмет охоплює технологічні аспекти побудови моделі «смартфон – штучний інтелект – продукт», що дозволяє ідентифікувати наявність алергенів у складі продукту.

Наукова новизна отриманих результатів полягає в розробці інтегрованого підходу до виявлення харчових алергенів, що вперше поєднує:

1. Комплексну класифікаційну модель для адаптивного вибору методу обробки вхідного зображення, яка автоматично визначає оптимальний шлях аналізу (OCR для упаковок, сканування для штрих-кодів, або розпізнавання для готових страв), що забезпечує універсальність і точність системи для різних типів продуктів.

2. Гібридну архітектуру обробки, що органічно поєднує обчислення на пристрої користувача з хмарними сервісами, оптимізуючи баланс між швидкістю, енергоефективністю та точністю аналізу.

3. Удосконалений метод виявлення алергенів на основі багаторівневого NLP-аналізу, що включає спеціально розроблену семантичну модель для обробки тексту з етикеток продуктів, здатну враховувати синоніми, морфологічні варіації та контекстуальні особливості згадування алергенів.

4. Розширену систему ідентифікації візуально подібних інгредієнтів у стравах з використанням адаптованої моделі AdaMine, яка значно перевершує існуючі рішення за точністю виявлення потенційно небезпечних компонентів у готових стравах.

5. Унікальний механізм персоналізації, що дозволяє налаштовувати систему під індивідуальні потреби користувача з врахуванням специфіки його алергічних реакцій та їх тяжкості.

Таким чином, розроблений підхід принципово відрізняється від існуючих рішень комплексним характером аналізу, поєднуючи переваги різних технологій штучного інтелекту для створення надійної системи виявлення харчових алергенів, що здатна функціонувати в реальних умовах використання.

Метою кваліфікаційної роботи магістра є автоматичне виявлення харчових алергенів на основі методів машинного навчання та комп'ютерного зору.

Для реалізації поставленої мети слід вирішити такі основні завдання:

1. Провести аналіз сучасних моделей машинного навчання та методів комп'ютерного зору, що використовуються для розпізнавання харчових продуктів та їх інгредієнтів.

2. Дослідити можливості використання технологій оптичного розпізнавання тексту (OCR) для автоматичного зчитування складу продуктів із пакування.

3. Розробити алгоритм комплексного аналізу харчових продуктів, який поєднує методи комп'ютерного зору та OCR для підвищення точності виявлення алергенів.

4. Створити гібридну архітектуру системи, що оптимально розподіляє обчислювальне навантаження між мобільним пристроєм та серверною частиною.

5. Реалізувати мобільний додаток, що інтегрує запропонований алгоритм та забезпечує користувачам швидкий доступ до інформації про склад продуктів.

6. Розробити і впровадити персоналізований механізм налаштування системи під індивідуальні потреби користувачів.

7. Провести експериментальне дослідження ефективності розробленої системи, оцінити її точність, продуктивність та можливості практичного застосування.

8. Розробити рекомендації щодо подальшого вдосконалення системи та її потенційного розширення для аналізу інших параметрів харчових продуктів.

Практична цінність розробленої кіберфізичної системи полягає в створенні функціонального програмного забезпечення, яке може бути негайно впроваджене та використане людьми з харчовими алергіями для підвищення безпеки харчування у повсякденному житті. Крім того, запропонована архітектура та методи можуть бути адаптовані для вирішення широкого спектру завдань, пов'язаних з аналізом харчових продуктів, включаючи визначення поживних властивостей, контроль якості та відстеження походження продуктів.

Реалізація даного проєкту має важливе наукове та прикладне значення, оскільки сприятиме підвищенню безпеки харчування для людей з алергією та відкриє нові можливості для розвитку інтелектуальних систем контролю харчових продуктів.

За темою кваліфікаційної роботи магістра опублікована одна стаття у міжнародному науковому журналі: Talapchuk V., Zaitseva E. Mobile-Oriented Cyber-Physical System for Food Allergen Detection Based on Machine Learning and Image Analysis. *Computer Systems and Information Technologies*. 2025. № 1. P. 29–35 DOI: <https://doi.org/10.31891/csit-2025-1-3>

1 АНАЛІЗ МОДЕЛЕЙ, МЕТОДІВ І ЗАСОБІВ РОЗПІЗНАВАННЯ ХАРЧОВИХ АЛЕРГЕНІВ

1.1 Класифікація харчових алергенів

Харчова алергія є патологічною реакцією імунної системи на певні компоненти їжі, які організм помилково сприймає як загрозу. Згідно з даними Всесвітньої організації охорони здоров'я (ВООЗ), до 10% дітей та 2-4% дорослих страждають від алергічних реакцій на харчові продукти [15]. На відміну від харчової непереносимості, яка не залучає імунну систему, харчова алергія може спричиняти тяжкі, потенційно небезпечні для життя реакції, включаючи анафілактичний шок.

Відповідно до Регламенту ЄС 1169/2011 про надання інформації споживачам харчових продуктів та Кодексу Аліментаріус (міжнародно визнаних стандартів харчової безпеки), виділяють 14 основних харчових алергенів, що підлягають обов'язковому маркуванню на упаковках [16]:

1. Злакові, що містять глютен (пшениця, жито, ячмінь, овес) – відповідальні за целіакію та алергічні реакції на глютен. Маркування продуктів як "безглютенових" регулюється окремими нормативами.

2. Ракоподібні (краби, креветки, лобстери) – містять специфічні білки, що викликають реакції переважно в дорослих. Алергія часто розвивається внаслідок професійного контакту.

3. Яйця – переважно спричиняють реакції на білки яєчного білка (овальбумін, овомукоїд). Можуть знаходитись у прихованому вигляді в багатьох продуктах.

4. Риба – містить алерген парвальбумін, що є термостійким і зберігає алергенність навіть після термічної обробки.

5. Арахіс – належить до сімейства бобових, має множинні алергенні білки (Ara h 1-13). Спричиняє одні з найтяжчих алергічних реакцій.

6. Соя – містить глікопротеїни, що викликають алергію переважно у дітей. Широко використовується як харчова добавка.

7. Молоко – основними алергенами є казеїн та бета-лактоглобулін. Найпоширеніша харчова алергія у дітей раннього віку.

8. Горіхи (мигдаль, фундук, волоський горіх, кеш'ю, пекан, фісташки, макадамія) – кожен вид має специфічні алергени.

9. Селера – містить алерген Арі g 1, що має перехресну реактивність з пилом берези.

10. Гірчиця – містить термостійкі та стійкі до рН-змін білки, що зберігаються навіть у ферментованих продуктах.

11. Кунжут – основні алергени: 2S альбумін та олеозин.

12. Діоксид сірки та сульфіти (E220-E228) – використовуються як консерванти, можуть викликати тяжкі респіраторні реакції.

13. Люпин – має значну перехресну реактивність з арахісом.

14. Молюски (устриці, мідії, кальмари) – містять тропоміозин, що зумовлює перехресну алергію з ракоподібними.

На рисунку 1.1 продемонстровані основні харчові алергени, що підлягають обов'язковому маркуванню.



Рисунок 1.1 – Основні харчові алергени, що підлягають обов'язковому маркуванню

1.2 Методи та технології автоматичного виявлення алергенів у харчових продуктах

Традиційно виявлення алергенів відбувалося через уважне вивчення етикеток продуктів або шляхом спеціалізованих хімічних тестів. Однак розвиток комп'ютерного зору та штучного інтелекту відкрив нові можливості для автоматизації цього процесу, зробивши його доступнішим та зручнішим для кінцевих користувачів.

Ранні дослідження у сфері автоматичного розпізнавання харчових продуктів почалися ще в 2000-х роках. Спершу методи розпізнавання базувалися на класичних алгоритмах комп'ютерного зору, які аналізували кольори, текстуру та форму продуктів. Наприклад, одна з перших систем FoodLog, що зображена на рисунку 1.2 дозволяла фотографувати їжу та отримувати її автоматичну ідентифікацію на основі колірних і геометричних характеристик.



Рисунок 1.2 Інтерфейс застосунку FoodLog [20]

Згодом, із розвитком глибоких нейронних мереж стало можливим суттєве покращення точності розпізнавання. Вже у 2015 році компанія Google презентувала

Im2Calories, додаток, що використовував згорткові нейронні мережі (CNN) для автоматичного аналізу калорійності страв на фото.

Завдяки розвитку комп'ютерного зору та штучного інтелекту, сучасні підходи базуються на поєднанні глибоких нейронних мереж, методів оптичного розпізнавання тексту (OCR) та алгоритмів обробки природної мови (NLP).

Основні напрями автоматичного розпізнавання їжі включають:

- класифікацію харчових продуктів – визначення категорії їжі на основі зображень із використанням згорткових нейронних мереж (CNN), таких як ResNet, MobileNet та EfficientNet;

- детекцію та сегментацію продуктів – виділення окремих об'єктів на зображеннях за допомогою YOLO, Faster R-CNN та Mask R-CNN, що дозволяє визначати інгредієнти у складних стравах;

- OCR-аналіз текстової інформації – автоматичне зчитування складу продуктів та маркування алергенів на основі технологій оптичного розпізнавання символів (Tesseract OCR, Google Vision OCR);

- обробку природної мови (NLP) – аналіз списків інгредієнтів для виявлення потенційно небезпечних речовин, синтаксичний аналіз та категоризацію харчових компонентів.

Поєднання цих підходів забезпечує комплексний аналіз харчових продуктів, що дозволяє не лише ідентифікувати страви, але й виявляти небезпечні для алергиків компоненти в тексті, оцінювати поживну цінність та створювати персоналізовані рекомендації. Подальший розвиток технологій у цій сфері спрямований на підвищення точності розпізнавання, зниження обчислювальної складності та адаптацію моделей для роботи у реальних умовах.

1.3 Моделі машинного навчання, що використовуються для розпізнавання їжі

Розвиток машинного навчання та штучного інтелекту дозволив досягти значних успіхів у сфері автоматичного розпізнавання їжі. Одним із ключових завдань є класифікація харчових продуктів, що полягає у визначенні категорії

страви або окремого продукту на основі його зображення. Для цього використовуються різноманітні моделі машинного навчання, від класичних алгоритмів до сучасних глибоких нейронних мереж.

1.3.1 Класичні алгоритми машинного навчання

До появи глибоких нейронних мереж класифікація зображень їжі здійснювалася за допомогою класичних алгоритмів машинного навчання, серед яких найбільш поширеними були метод k -найближчих сусідів (k -NN), метод опорних векторів (SVM), а також дерева рішень та випадкові ліси (Random Forest).

Метод k -найближчих сусідів (k -NN), проілюстрований на рисунку 1.3, є одним із найпростіших алгоритмів, що використовує порівняння вхідного зображення з найближчими прикладами у навчальній вибірці. Принцип роботи цього методу полягає у пошуку серед навчальної множини k найближчих за метрикою відстані прикладів та визначенні класу більшості серед них. У контексті нашої задачі це дозволяє віднести новий продукт до одного з двох класів: "містить алергени" або "не містить алергенів".

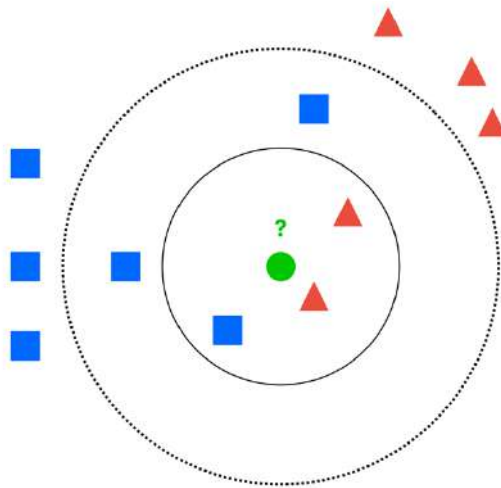


Рисунок 1.3 – Принцип методу найближчих сусідів [21]

Зелений об'єкт у центрі позначає новий продукт, щодо якого потрібно зробити класифікацію. Червоні трикутники представляють об'єкти з класу "містить алергени", а сині квадрати – продукти без алергенів. Залежно від значення k ,

система обчислює відстані до найближчих сусідів та приймає рішення, орієнтуючись на кількість представників кожного класу серед них. Цей метод простий, інтуїтивно зрозумілий і непогано працює для чітко розділених класів продуктів.

Метод опорних векторів (SVM), математична модель якого представлена у формулі (1.1), використовується для пошуку гіперплощини, яка розділяє класи харчових продуктів. Попри свою високу точність у багатьох задачах, метод має обмеження у випадках складних багатокласових розподілів даних.

У контексті виявлення харчових алергенів цей підхід може бути використаний для розмежування продуктів, що містять алергени, та продуктів, які є безпечними. Кожен продукт представлено у вигляді вектору ознак – це може бути, наприклад, вектор тексту складу після векторизації або ознакове представлення зображення.

$$f(x) = \text{sign}(\sum_{i=1}^n \alpha_i y_i K(x_i, x) + b), \quad (1.1)$$

де:

x – вектор ознак нового продукту,

x_i – навчальні приклади,

$y_i \in \{-1, 1\}$ – мітки класу (“містить алергени” / “не містить”),

$K(x_i, x)$ – ядрова функція (наприклад, RBF або лінійне скалярне добуток),

α_i – вагові коефіцієнти, знайдені у процесі навчання,

b – зсув.

Важливою перевагою методу опорних векторів є його здатність ефективно працювати навіть з обмеженим набором навчальних даних, що особливо цінно при розробці спеціалізованих моделей для виявлення рідкісних алергенів. Завдяки використанню ядрових функцій метод може виявляти нелінійні залежності між ознаками продуктів, що підвищує точність класифікації у випадках, коли візуальні ознаки алергенних і неалергенних продуктів мають складний характер розподілу.

Для кращого розуміння принцип методу опорних векторів та його застосування до класифікації харчових продуктів, розглянемо його геометричну інтерпретацію, представлену на рисунку 1.4. Тут наочно показано, як SVM знаходить оптимальну гіперплощину для розділення двох класів продуктів з алергенами та без них у просторі ознак.

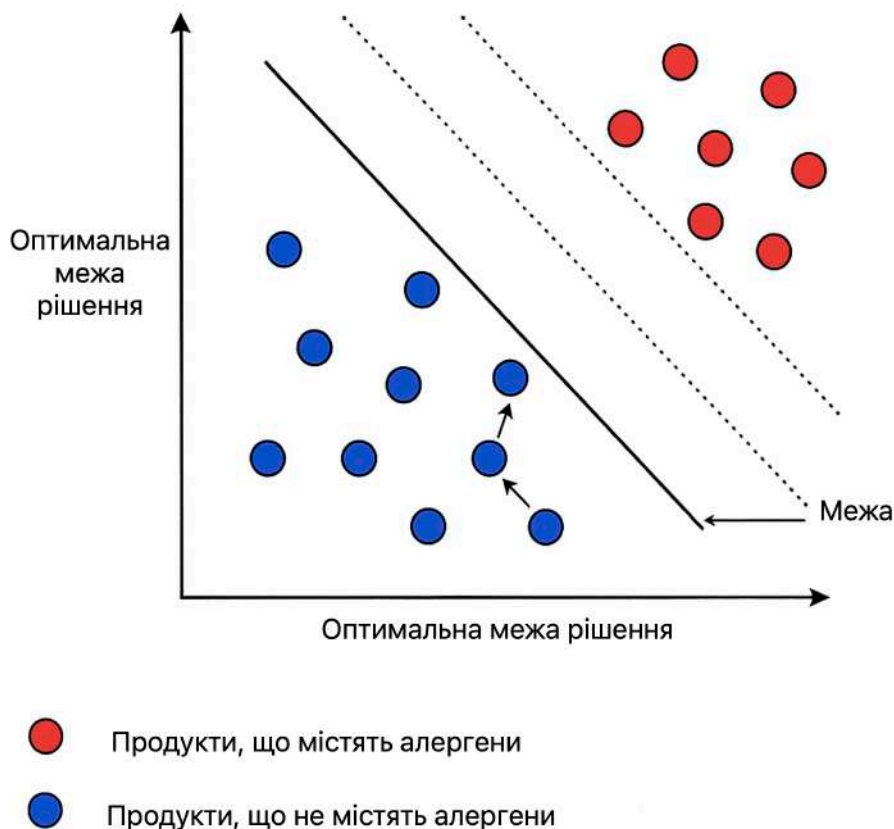


Рисунок 1.4 – Геометрична інтерпретація методу опорних векторів (SVM) у задачі виявлення алергенів

У результаті система отримує математично обґрунтований механізм прийняття рішення на основі ознак.

Дерева рішень та випадкові ліси (Random Forest) представляють собою потужні ансамблеві методи машинного навчання, що об'єднують кілька дерев рішень для покращення узагальнюючої здатності класифікатора.

Процес прийняття рішень у такій моделі візуально представлений на рисунку 1.5. Тут можна простежити, як алгоритм послідовно аналізує характеристики

продукту, починаючи з кореневого вузла, де відбувається перевірка ключової умови – наприклад, "чи міститься у складі слово арахіс?". Залежно від відповіді, перехід відбувається або вліво, або вправо – до наступного вузла. Поступово формується набір вкладених правил, які в кінці приводять до листового вузла з фінальною класифікацією: "містить алергени" або "не містить".

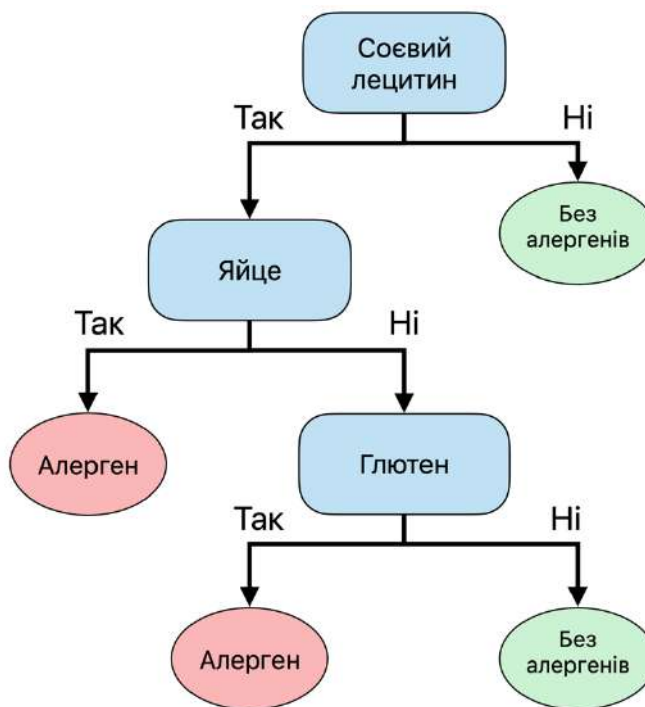


Рисунок 1.5 – Структура дерева рішень для класифікації продуктів за наявністю алергенів.

Простота та прозорість дерева рішень дозволяють легко адаптувати модель під конкретні випадки або регіональні списки алергенів.

1.3.2 Глибокі нейронні мережі для класифікації харчових продуктів

Хоча класичні методи все ще знаходять застосування, вони суттєво поступаються нейронним мережам у точності та здатності до узагальнення складних візуальних ознак їжі.

Згорткові нейронні мережі (CNN) стали основою сучасних підходів до класифікації харчових продуктів. Вони дозволяють моделі самостійно навчатися розпізнавати складні візуальні особливості їжі, такі як текстура, колір, форма тощо.

Найпопулярніші архітектури, що використовуються для розпізнавання харчових продуктів, включають ResNet (Residual Networks), MobileNet та EfficientNet, кожна з яких пропонує оптимальний баланс між точністю класифікації та обчислювальною складністю для різних сценаріїв застосування.

ResNet (Residual Networks) використовує резидуальні зв'язки для збереження градієнта під час навчання глибоких мереж, забезпечуючи високу точність класифікації навіть для великої кількості класів продуктів. Як показано на рисунку 1.6, мережа складається з 50 згорткових шарів, з'єднаних «skip»-шляхами, завдяки чому велика глибина не спричиняє затухання градієнта. У кожному такому блоці послідовно застосовуються три компактні згортки 1×1 , 3×3 і 1×1 , що зменшує кількість параметрів, але зберігає виразність моделі.

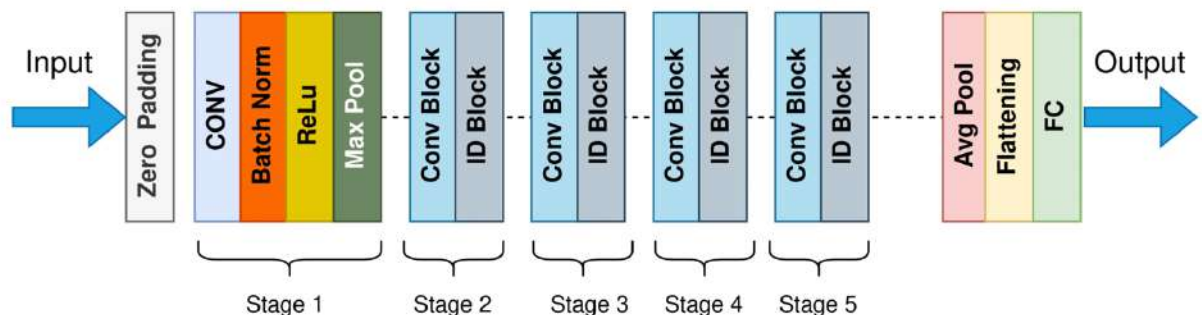


Рисунок 1.6 – Архітектура ResNet-50 [14]

Архітектура ResNet-50 демонструє ефективність використання залишкових блоків, що дозволяють формувати глибокі моделі без проблеми затухання градієнтів. Завдяки такій структурі мережа здатна вивчати складні ієрархічні ознаки харчових продуктів, починаючи від простих контурів і текстур на перших шарах до високорівневих семантичних особливостей на глибших рівнях.

На відміну від ресурсоємної архітектури ResNet, MobileNet пропонує значно легше рішення, спеціально оптимізоване для мобільних пристроїв. Якщо ResNet

фокусується на максимальній точності за рахунок глибини мережі, то MobileNet прагне забезпечити оптимальний баланс між точністю та обчислювальною ефективністю, що дозволяє швидко розпізнавати їжу на смартфонах при мінімальному споживанні ресурсів. Структуру моделі MobileNetv2 детально представлено на рисунку 1.7.

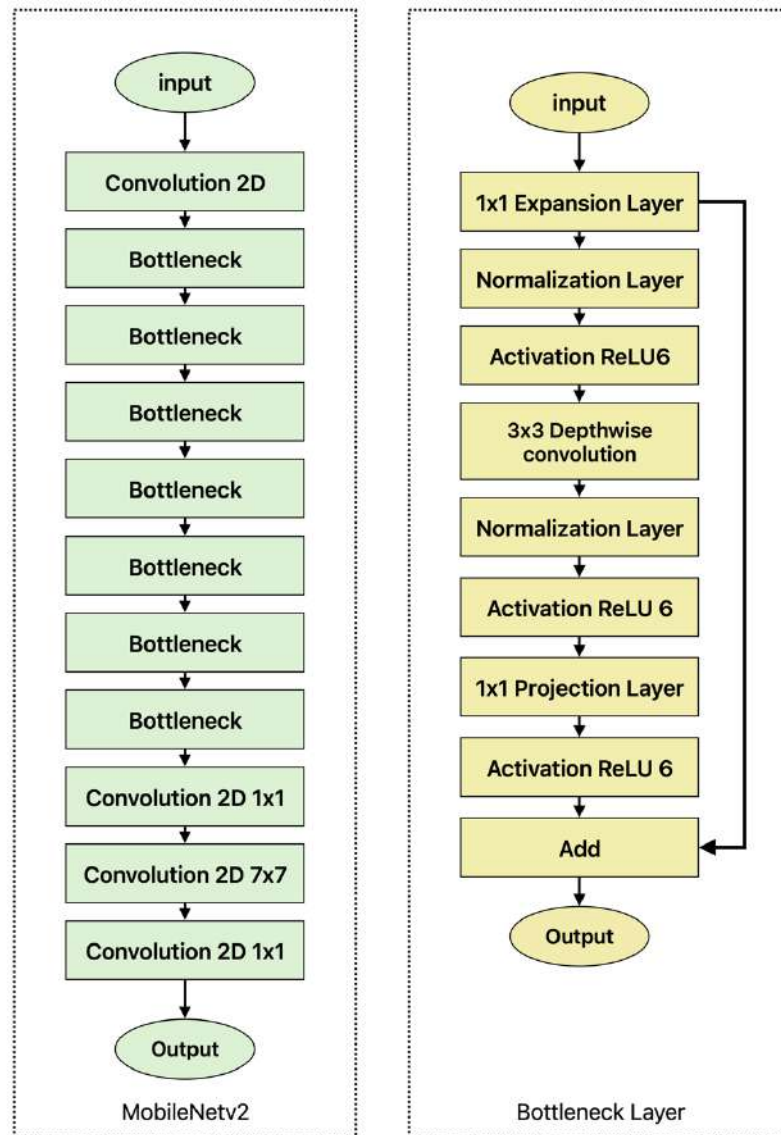


Рисунок 1.7 – Архітектура моделі MobileNetv2 [22]

Модель ґрунтується на повторюваному інвертованому bottleneck-блоці: вузький тензор спершу розширюється 1×1 згорткою, далі опрацьовується енергоефективною depthwise-згорткою 3×3 , а потім стискається назад 1×1 шаром;

якщо розміри не змінюються, результат додається до виходу попереднього шару. Така компактна трирівнева схема зберігає достатньо інформативних характеристик зображення, водночас різко скорочуючи обчислення та обсяг пам'яті, тому модель здатна працювати в реальному часі на звичайному смартфоні.

Завершуючи огляд ключових архітектур, варто розглянути EfficientNet – інноваційну модель, що пропонує принципово інший підхід до проектування нейронних мереж порівняно з ResNet та MobileNet. EfficientNet вирізняється своєю унікальною здатністю масштабуватися за трьома основними параметрами: глибиною, шириною та роздільною здатністю, що дозволяє досягати високої точності розпізнавання при суттєво меншій кількості параметрів. Архітектура цієї моделі представлена на рисунку 1.8.

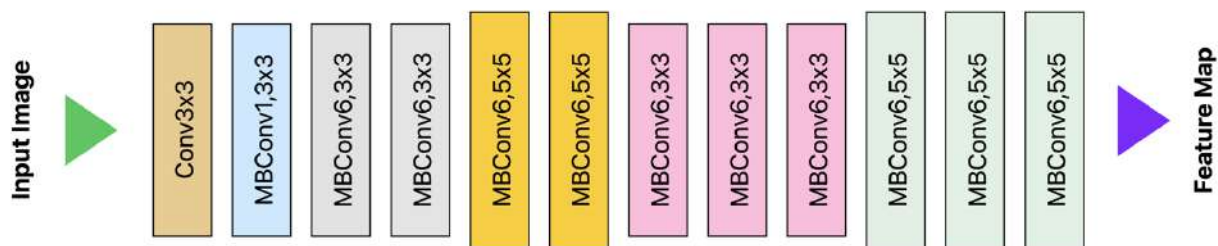


Рисунок 1.8 – Архітектура моделі EfficientNet [23]

Завдяки поєднанню легких MBConv-блоків із механізмом squeeze-and-excitation та «компаунд-масштабуванню», версії EfficientNet-B0...B7 демонструють одне з найкращих у галузі співвідношень «точність / FLOPs», що робить їх популярною базою для мобільних і вбудованих систем комп'ютерного зору.

У контексті розпізнавання харчових продуктів EfficientNet пропонує важливу перевагу: гнучкість у налаштуванні під конкретні обчислювальні обмеження при збереженні високої точності. Наприклад, легша версія EfficientNet-B0 може бути використана для швидкої попередньої класифікації типу продукту, тоді як важча

EfficientNet-V3 може застосовуватися для детальнішого аналізу потенційно небезпечних компонентів.

Переваги та недоліки даних моделей показано в таблиці 1.1:

Таблиця 1.1 – Порівняльний аналіз найпопулярніших архітектур

Модель	Точність на Food-101	Час інференсу на мобільному пристрої	Ключові переваги	Основні обмеження	Придатність для мобільного застосування
ResNet-50	88.5%	~200-250 мс	Висока точність, стабільність, глибоке виділення ознак	Великий розмір, висока обчислювальна складність	Середня, потребує оптимізації
ResNet-101	90.2%	~350-450 мс	Найвища точність, відмінне виділення складних ознак	Дуже великий розмір, повільний інференс	Низька
MobileNet V2	74.9%	~30-50 мс	Компактність, висока швидкодія, енергоефективність	Низька точність, чутливість до якості зображення	Висока

Кінець таблиці 1.1

MobileNetV3	76.8%	~35-55 мс	Покращена точність при збереженні компактності	Все ще нижча точність ніж у великих моделей	Висока
EfficientNet-B0	77.3%	~45-65 мс	Оптимальний баланс розміру та точності	Складність модифікації, чутливість до гіперпараметрів	Висока
EfficientNet-B3	84.1%	~120-150 мс	Точність близька до ResNet при меншому розмірі	Більший розмір та нижча швидкодія ніж у B0	Середня

Дані, представлені в таблиці 1.1, наочно демонструють компроміс між точністю та швидкістю для різних архітектур. Як бачимо, моделі родини ResNet забезпечують найвищу точність класифікації на наборі даних Food-101, але мають суттєві обмеження щодо застосування на мобільних пристроях через їхній розмір та обчислювальну складність. Водночас архітектури MobileNet та EfficientNet пропонують більш збалансоване рішення, жертвуючи кількома відсотками точності заради значного приросту швидкості та енергоефективності.

1.3.3 Детекція та сегментація об'єктів у складних стравах

Розпізнавання складних страв є одним із найскладніших завдань у сфері комп'ютерного зору, оскільки вони містять кілька інгредієнтів, які можуть перекривати один одного, мати подібний колір або текстуру, а також бути змішаними. У таких випадках звичайна класифікація недостатня, оскільки вона дозволяє визначити лише загальний тип страви. Для детального аналізу

використовуються методи детекції та сегментації об'єктів, які дозволяють ідентифікувати окремі інгредієнти всередині страви та точно визначати їхню кількість.

Сучасні методи розпізнавання складних страв базуються на використанні нейронних мереж для детекції та сегментації об'єктів. Ці методи дозволяють виділяти на зображенні окремі інгредієнти, що є надзвичайно важливим для виявлення харчових алергенів.

Серед найпоширеніших підходів до цієї задачі варто виділити архітектуру YOLO (You Only Look Once) – одноетапну модель детекції, що забезпечує високу швидкість роботи. На відміну від багатьох інших алгоритмів, YOLO аналізує все зображення одразу, розбиваючи його на сітку клітин, де кожна клітина прогнозує наявність об'єктів. Це дозволяє системі миттєво ідентифікувати різні компоненти страви, що критично важливо для роботи в реальному часі на мобільних пристроях.

Для задач, де швидкість менш критична, але необхідна вища точність, застосовується двоетапний підхід Faster R-CNN. Ця архітектура спочатку генерує потенційні області інтересу, а потім класифікує знайдені об'єкти. Такий метод забезпечує кращу точність виявлення дрібних інгредієнтів, що може бути вирішальним при ідентифікації алергенів, які присутні в невеликій кількості.

Розвитком Faster R-CNN стала модель Mask R-CNN, яка, крім визначення положення об'єкта, також створює його піксельну маску. Це дозволяє не лише ідентифікувати наявність алергену, але й точно визначити його місце та кількість у страві. Наприклад, система може не просто виявити наявність горіхів у десерті, але й показати їх точне розташування та приблизний об'єм.

Для повноцінної семантичної сегментації, особливо у випадках з неструктурованими харчовими сумішами, ефективно застосовується архітектура U-Net. Її U-подібна структура з симетричними кодуєчим та декодуєчим шляхами дозволяє зберігати просторову інформацію при сегментації, що особливо цінно для аналізу складних текстур їжі.

Порівняння даних моделей продемонстровано в таблиці 1.2

Таблиця 1.2 – Порівняння моделей детекції об'єктів для виявлення інгредієнтів у складних стравах.

Метод	mAP на Food-Objects	Швидкість (FPS)	Ключові переваги	Основні обмеження	Придатність для мобільної розробки
YOLO v5	68.3%	45-60	Висока швидкість, одноетапна детекція, компактні версії	Менша точність для дрібних об'єктів	Висока (особливо для малих моделей)
YOLO v8	73.2%	35-55	Покращена точність, сучасна архітектура	Вища обчислювальна складність порівняно з v5	Середня-Висока
Faster R-CNN	74.5%	5-10	Висока точність, стабільність, добре виявлення малих об'єктів	Низька швидкість, великий розмір	Низька
Mask R-CNN	76.1%	3-7	Точна сегментація об'єктів, а не лише детекція	Дуже низька швидкість, величезний розмір	Дуже низька

Продовження таблиці 1.2

SSD MobileNet	63.8%	30-50	Баланс швидкості та точності, компактність	Нижча точність ніж у двоетапних моделей	Висока
EfficientDet	69.7%	15-40	Масштабованість, ефективне використання ресурсів	Складність налаштування, чутливість до гіперпараметрів	Середня- Висока

На основі даних з таблиці 1.2, можна чітко простежити компроміс між точністю та швидкістю розглянутих моделей. Архітектури з найвищою точністю (Mask R-CNN, Faster R-CNN) демонструють найнижчу швидкість, тоді як одноетапні детектори типу YOLO v8 та EfficientDet пропонують більш збалансоване співвідношення показників.

1.4 Використання OCR для отримання даних з етикеток

OCR (Optical Character Recognition) застосовується для автоматичного зчитування тексту з фотографій упаковок і етикеток харчових продуктів. Ця технологія дозволяє перетворити зображення з текстом (склад, таблиця харчової цінності, дата придатності тощо) у машиночитний текст. Сучасні OCR-двигуни (наприклад, Tesseract, Google Vision OCR) досить точно розпізнають надрукований текст на контрастному фоні, що підходить для більшості етикеток. У системах аналізу їжі OCR часто інтегрується в мобільні додатки: користувач фотографує етикетку, після чого алгоритм виділяє зону тексту і розпізнає її.

Прикладом є прототипи алергенної сканер-системи, де OCR використано для сканування списку інгредієнтів на упаковці [3]. Отриманий текст далі аналізується на предмет небезпечних компонентів. Таким чином, замість ручного перечитування складу, користувач отримує автоматично згенеровані дані. OCR також використовується в проєктах зі створення великих баз даних продуктів: так,

відкрита платформа Open Food Facts завдяки краудсорсингу зібрала понад 7 мільйонів фотографій етикеток і застосовує OCR для отримання з них інформації [4]. Надалі ці дані зберігаються у структурованому вигляді (списки інгредієнтів, кількість цукру, калорійність тощо) та можуть використовуватися для навчання моделей або прямих запитів.

Після отримання текстових даних за допомогою OCR постає завдання інтерпретації тексту – зокрема, виявлення в ньому певних ключових відомостей. Одне з прикладних завдань – автоматичне виявлення алергенів у списку інгредієнтів. Це актуально для користувачів з харчовою алергією або непереносимістю, яким необхідно швидко визначити, чи містить продукт небезпечний компонент. Методами NLP можна проаналізувати розпізнаний текст і знайти в ньому назви потенційних алергенів (наприклад, “горіхи”, “глютен”, “лактоза” тощо).

Найпростіший підхід – перевірка на входження певних слів зі списку відомих алергенів. Проте списки інгредієнтів можуть містити синоніми або наукові назви (напр. “казеїн” замість “молочний білок”). Тому більш просунуті рішення використовують техніки NLP для нормалізації імен інгредієнтів: приведення їх до базових термінів, співставлення різних назв одному алергену [3]. Як зазначено в одному з досліджень, важливо врахувати варіації назв та мов: наприклад, “milk powder” і “powdered milk” слід розпізнавати як одне й те саме, так само як “leche” (іспанською) відповідає “milk” [3]. Для цього формуються словники синонімів, а також правила (регулярні вирази) для вилучення базових форм слів.

Крім того, NLP-моделі можуть класифікувати інгредієнти за категоріями (алергени, добавки, барвники). У складніших випадках застосовуються моделі розпізнавання сутностей (NER), які позначають у тексті назви алергенів, навіть якщо вони з’являються у складних сполуках. Деякі сучасні системи поєднують правила і машинне навчання. Наприклад, проект з розробки персонального помічника для алергиків використовує OCR для зчитування, а далі машинний аналіз тексту звіряє отриманий список інгредієнтів з профілем відомих алергенів

користувача [3]. Якщо знаходиться збіг (наприклад, інгредієнт “арахісова олія” при алергії на арахіс), система генерує попередження.

Важливою є підтримка багатомовності – міжнародні продукти можуть мати склад різними мовами, тож NLP-модель повинна містити словники алергенів англійською, українською, латинськими назвами (наприклад, “E120” – харчовий барвник з кошенілі, алерген для деяких людей). Для цього розробники включають в бази даних міжнародні списки алергенів і їхні позначення.

У підсумку, методи NLP в аналізі харчових продуктів дозволяють автоматизувати перевірку складу на наявність небезпечних або цікавих користувачу компонентів. Поєднання OCR + NLP вже реалізоване в експериментальних системах і показує високу оперативність та точність у виявленні алергенів, що сприяє підвищенню безпеки споживання продуктів для людей з алергіями [3].

1.5 Технологічні засоби та інструменти, що застосовуються у сфері аналізу харчових продуктів

Для реалізації описаних вище підходів залучається широкий спектр технологічних ресурсів. По-перше, це набори даних – колекції зображень їжі та бази знань про продукти, які використовуються для навчання і тестування моделей. По-друге, програмні інструменти – фреймворки глибокого навчання, бібліотеки OCR/NLP, спеціалізовані API. По-третє, апаратні засоби – від камер смартфонів до хмарних сервісів – які забезпечують збір даних та виконання обчислень. Нижче розглянуто деякі ключові ресурси та рішення.

1.5.1 Огляд баз даних харчових продуктів та їх використання для навчання моделей

Наявність великих і різноманітних баз даних є критичною для навчання моделей розпізнавання їжі. За останнє десятиліття створено кілька

загальнодоступних датасетів зображень харчових продуктів, які стали стандартом у дослідженнях. Один з найвідоміших – Food-101 [5]. Він містить 101 000 зображень, розподілених по 101 класу страв (приблизно по 1000 зображень на кожен клас) [5]. Класи представляють популярні страви світу (піца, бургер, суші, тощо), і дані зібрані з інтернет-фотографій, що забезпечує різноманітність ракурсів і стилів подачі. Food-101 використовується як еталонний набір для оцінки точності моделей; наприклад, оригінальний підхід на основі випадкових лісів дав ~50% точності, тоді як сучасні CNN досягають ~90% на цьому наборі [6].

Іншим поширеним набором є UECFOOD (Food-100 і Food-256) – датасети з зображеннями японських страв, що включають 100 та 256 категорій відповідно [6]. Вони особливо цінні наявністю розмітки області кожної страви на зображенні, що дозволяє використовувати їх для детекції об'єктів. Також варто згадати ETHZ Food-101 (синонім до вищезгаданого Food-101, представлений в ETH Цюрих) [6], UNIMIB2016 – набір знімків таці з їжею для аналізу харчування в лікарнях, VireoFood-172 – спеціалізований датасет для розпізнавання інгредієнтів (172 категорії інгредієнтів, зібраний дослідниками Vireo) [6], а також Recipe1M – велика база рецептів з картинками, що використовується для задач виявлення рецепту з фото.

Окрім зображень, існують бази даних продуктів з текстовою інформацією. Наприклад, національні бази на кшталт USDA FoodData Central містять тисячі записів про харчові продукти з вказанням їхнього складу, калорійності, вмісту білків/жирів/вуглеводів. Такі дані корисні для побудови систем оцінки харчової цінності – модель може зіставляти розпізнаний продукт із записом у базі і отримувати його характеристики. Широковідомим є відкрита глобальна база Open Food Facts, яка на сьогодні є найбільшою: вона містить інформацію про понад 3 млн продуктів по всьому світу (марка, склад, поживні показники тощо) та понад 7 млн зображень упаковок [4]. Ця база використовується як джерело для навчання моделей OCR/NLP (на ній відпрацьовували алгоритми розпізнавання тексту та очищення даних про склад продуктів). Наприклад, волонтерами OFF був

реалізований проект покращення якості OCR інгредієнтів через машинне навчання, що дозволило суттєво очистити і структурувати текстові дані цієї бази [3].

1.5.2 Комерційні та дослідницькі рішення у сфері розпізнавання їжі

Інтерес до технологій розпізнавання їжі привів до появи численних застосунків і сервісів – як у наукових лабораторіях, так і в індустрії. Дослідницькі прототипи проклали шлях для багатьох функцій, що нині реалізовані комерційно. Наприклад, ще в 2010 р. було запропоновано систему FoodLog для смартфона, яка дозволяла фотографувати страви і розпізнавати їх на основі колірних та SIFT-ознак [6]. У 2015 р. компанія Google представила концепт Im2Calories – мобільний додаток, що використовує CNN для визначення виду їжі на фото і оцінки її калорійності [6]. Того ж року дослідники з Японії розробили систему FoodCam, яка автоматично ідентифікувала страви в кафетерії з використанням комп'ютерного бачення [6]. Ці роботи демонстрували можливості технології і сприяли її популяризації.

Нині на ринку існують комерційні мобільні додатки, що пропонують функцію розпізнавання їжі, деякі з них показано на рисунку 1.9. Серед них – додатки для підрахунку калорій (Lose It!, MyFitnessPal та ін.), які інтегрували сканер страв: користувач робить знімок, а додаток намагається визначити, що на ньому, і внести до харчового щоденника. Компанії-розробники не завжди публікують деталі реалізації, але відомо, що за подібними сервісами стоять нейронні мережі, натреновані на великих наборах зображень їжі. Наприклад, французький стартап Foodvisor заявляє про використання власної нейромережі для розпізнавання понад 1000 різних страв з фото.

У дослідницькій сфері триває робота над спеціалізованими сценаріями. Наприклад, системи для розумних холодильників здатні розпізнавати продукти всередині (овочі, фрукти, пляшки) і відстежувати запаси [7]. Інші проекти займаються автоматизацією оцінки харчування в лікарнях та будинках для людей похилого віку, де камера аналізує піднос із їжею до та після трапези, щоб

визначити, скільки пацієнт з'їв. Комерціалізація таких розробок вже почалася – існують стартапи, що пропонують AI-камери для ресторанів (розпізнавання страв на касі для автоматичної оплати).

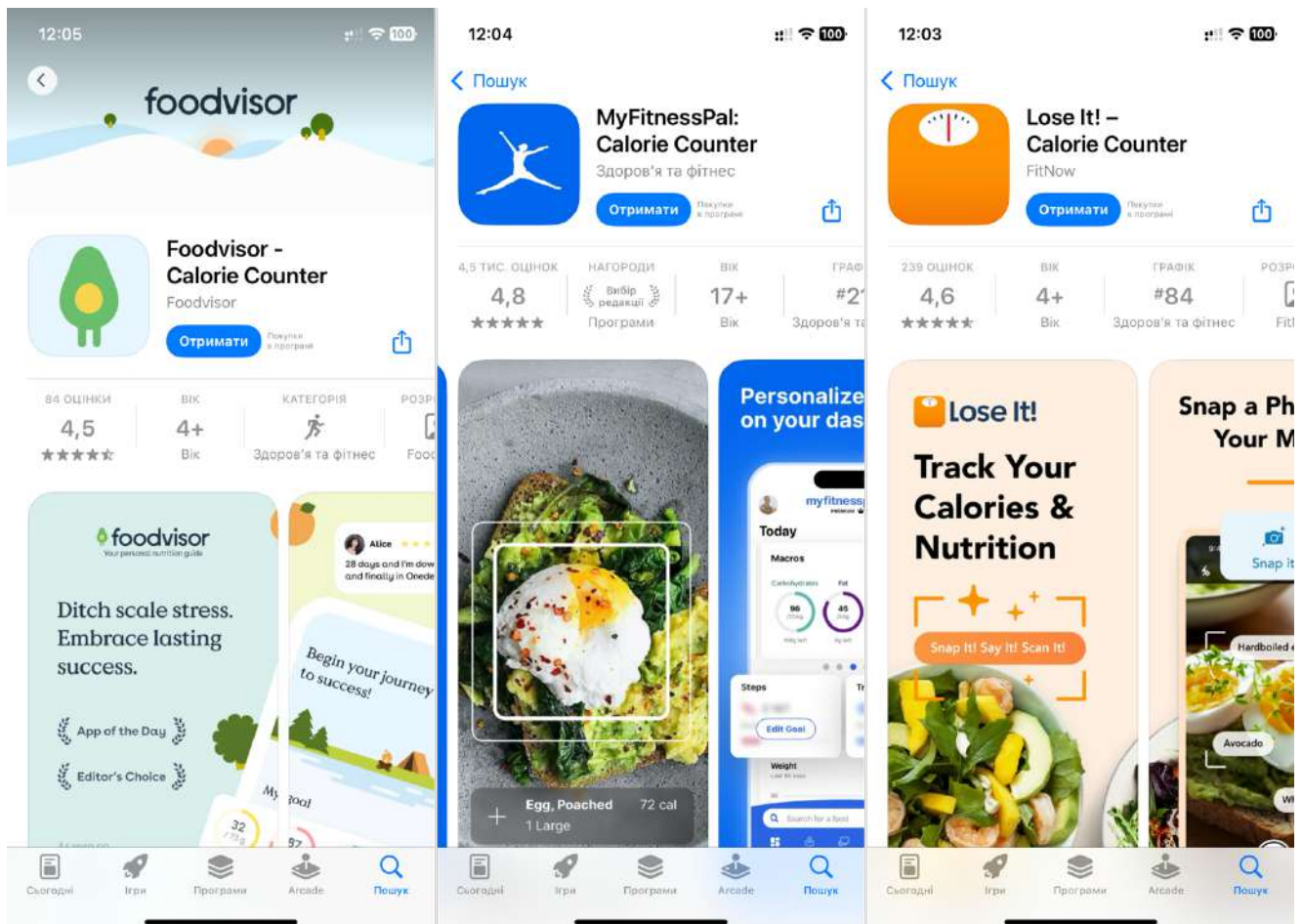


Рисунок 1.9 – Комерційні мобільні додатки, що пропонують розпізнавання їжі

У дослідницькій сфері триває робота над спеціалізованими сценаріями. Наприклад, системи для розумних холодильників здатні розпізнавати продукти всередині (овочі, фрукти, пляшки) і відстежувати запаси [7]. Інші проекти займаються автоматизацією оцінки харчування в лікарнях та будинках для людей похилого віку, де камера аналізує піднос із їжею до та після трапези, щоб визначити, скільки пацієнт з'їв. Комерціалізація таких розробок вже почалася – існують стартапи, що пропонують AI-камери для ресторанів (розпізнавання страв на касі для автоматичної оплати).

Отже, сьогодні є зростаючий набір рішень для розпізнавання їжі: від академічних прототипів до масових додатків. Вони підтверджують практичну цінність технології, проте часто працюють у відносно спрощених умовах (обмежений набір страв, контрольоване середовище зйомки). Це створює простір для вдосконалення – появи нових систем, здатних більш надійно та універсально розпізнавати їжу у різних ситуаціях.

1.6 Висновки до першого розділу

У першому розділі проведено всебічний аналіз моделей, методів і засобів розпізнавання харчових алергенів, що дозволило сформувавши теоретичну базу для подальшої розробки мобільно-орієнтованої кіберфізичної системи.

Досліджено класифікацію харчових алергенів та визначено 14 основних груп, що підлягають обов'язковому маркуванню згідно з регламентом ЄС 1169/2011. Серед них особливу увагу приділено таким поширеним алергенам як глютен, молоко, яйця, арахіс, соя, горіхи, риба та ракоподібні, які становлять найбільшу небезпеку для значної частини населення. Детально розглянуто характеристики кожного з алергенів, їх хімічний склад та типові способи їх маркування на упаковках харчових продуктів.

Проведено аналіз методів та технологій автоматичного виявлення алергенів у харчових продуктах, що показав еволюцію підходів від простих алгоритмів аналізу кольорів і текстур до сучасних методів глибокого навчання. Простежено історичний розвиток систем розпізнавання їжі, починаючи від перших рішень, таких як FoodLog, до сучасних AI-орієнтованих платформ. Оцінено переваги та недоліки кожного підходу, а також їх застосовність у контексті виявлення харчових алергенів.

Розглянуто моделі машинного навчання для розпізнавання їжі, починаючи від класичних алгоритмів (kNN, SVM, Random Forest) і завершуючи сучасними згортковими нейронними мережами (ResNet, MobileNet, EfficientNet). Проведено порівняльний аналіз архітектур нейронних мереж за критеріями точності,

швидкодії та придатності для мобільної розробки. Встановлено, що для задачі класифікації харчових продуктів найбільш перспективними є легкі моделі типу MobileNet та EfficientNet, які забезпечують оптимальний баланс між точністю та обчислювальною ефективністю.

Проаналізовано методи оптичного розпізнавання тексту (OCR) для отримання даних з етикеток продуктів та підходи до семантичної обробки текстової інформації для виявлення потенційно небезпечних інгредієнтів. Виявлено основні проблеми, що виникають при зчитуванні тексту з упаковок, такі як варіативність шрифтів, низька контрастність, різні кути фотографування та необхідність обробки тексту різними мовами. Досліджено перспективні методи вирішення цих проблем за допомогою сучасних OCR-технологій та алгоритмів обробки природної мови.

Досліджено технологічні засоби та інструменти, що застосовуються у сфері аналізу харчових продуктів, включаючи спеціалізовані бази даних (Food-101, UECFOOD, OpenFoodFacts) та комерційні рішення. Проаналізовано їх структуру, повноту та якість даних, а також можливості їх використання при розробці системи виявлення алергенів. Розглянуто існуючі комерційні додатки та оцінено їх ефективність на основі доступних досліджень.

На основі проведеного аналізу визначено, що існуючі рішення мають ряд обмежень: недостатня точність класифікації візуально схожих інгредієнтів, складність розпізнавання компонентів у складних стравах, проблеми з розпізнаванням тексту в реальних умовах та відсутність комплексного підходу до виявлення алергенів. Це обґрунтовує необхідність розробки нової системи, яка поєднуватиме переваги різних технологій для забезпечення високої точності та надійності виявлення харчових алергенів у реальних умовах використання.

Визначені в першому розділі обмеження існуючих підходів та перспективні напрямки їх подолання формують основу для розробки власних моделей і методів виявлення харчових алергенів, що буде розглянуто в наступних розділах роботи. Проведений аналіз дозволив сформулювати чіткі вимоги до створюваної системи та визначити оптимальні технологічні рішення для їх реалізації.

2 КОНЦЕПТУАЛЬНЕ ПРОЄКТУВАННЯ КІБЕРФІЗИЧНОЇ СИСТЕМИ ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ

2.1 Технологічні засоби реалізації системи

Розробка ефективної системи виявлення харчових алергенів вимагає комплексного підходу до вибору технологічного стеку, моделей машинного навчання та методів обробки даних. Для реалізації такої системи необхідно враховувати особливості платформи розгортання, вимоги до швидкодії та точності розпізнавання, а також обмеження мобільних пристроїв.

Для практичної реалізації було обрано платформу iOS, оскільки вона забезпечує необхідний інструментарій для роботи з комп'ютерним зором та машинним навчанням, а також була фізично доступна для розробки та тестування. iOS надає інтегрований набір інструментів, включаючи CoreML для виконання моделей машинного навчання та Vision для задач розпізнавання зображень і тексту. Ці компоненти дозволяють реалізувати високоточне виявлення алергенів при збереженні автономності роботи додатку.

Ключовими вимогами до системи є: висока точність класифікації типу вхідного зображення, надійне розпізнавання тексту з етикеток продуктів та ефективний семантичний аналіз переліку інгредієнтів для виявлення потенційних алергенів. Для вирішення цих завдань необхідно розробити комплексну архітектуру, що поєднує різні підходи машинного навчання з методами обробки природної мови.

2.2 Загальна концепція системи

Запропонована система ґрунтується на поетапному підході обробки візуальної інформації про харчовий продукт, загальну концепцію якої представлено на рисунку 2.1.. Основна ідея полягає в тому, що користувач робить фото продукту, а система автоматично класифікує тип зображення: це може бути упаковка, штрих-код або готова страва. Така класифікація є критично важливою,

оскільки вона дозволить нам ефективніше визначати алергени незалежно від типу вхідних даних.

На рівні концепції система складається з кількох взаємопов'язаних модулів. Перший і найважливіший – модуль класифікації зображення, реалізований за допомогою попередньо натренованої моделі CoreML, яка розпізнає, до якого з трьох класів належить зображення: упаковка, штрих-код чи страва. Цей модуль працюватиме локально на пристрої користувача, що забезпечує швидкість та конфіденційність.

Залежно від результатів класифікації зображення активується одна з трьох гілок обробки. При виявленні штрих-коду система використовуватиме механізми зчитування стандарту EAN-13 або Code128 через VisionKit або інші iOS-фреймворки. Зчитаний код буде передаватися для пошуку інформації в базах даних, звідки отримується склад продукту.

У випадку розпізнавання упаковки продукту система залучає модуль OCR для оптичного розпізнавання тексту, який витягує текстові дані зі зображення. Далі ці дані передаватимуться в модуль обробки природної мови, який проаналізує склад продукту та виявить потенційні алергени.

Якщо система ідентифікує зображення як готову страву, активується модель класифікації їжі на основі глибокого навчання, яка дозволить визначити ймовірну страву та її інгредієнти. Далі, на основі заздалегідь складених шаблонів або зовнішніх баз даних рецептів, формується список інгредієнтів, які можуть знаходитися в даній страві.

Незалежно від типу джерела, результати обробки приводяться до єдиного структурованого формату через модуль уніфікації даних, що містить повний список інгредієнтів. Завершальною ланкою є модуль виявлення алергенів, який використовує список небажаних речовин, заданих користувачем у профілі. Проводиться порівняння кожного знайденого інгредієнта з цим списком, після чого користувачеві надається зворотний зв'язок: чи безпечний продукт, або ж містить алергени.



Рисунок 2.1 – Діаграма загальної концепції системи

Таким чином, концепція передбачає гнучку, модульну архітектуру, яка адаптується до типу вхідного зображення та дозволяє розширювати функціонал системи – наприклад, додавати нові джерела даних, підтримку нових алергенів, мов або типів продуктів. Такий підхід забезпечує надійність і масштабованість для реального використання в повсякденному житті користувачів з харчовими алергіями.

2.3 Архітектура мобільної кіберфізичної системи

Розробка ефективної кіберфізичної системи виявлення харчових алергенів вимагає технічного підходу, що інтегрує комп'ютерний зір, машинне навчання, обробку природної мови та мобільні технології. Розроблена система функціонує як програмно-апаратний комплекс, що використовує сенсори мобільного пристрою для збору даних про харчові продукти, а потужні алгоритми штучного інтелекту – для їх аналізу та ідентифікації потенційно небезпечних компонентів.

Архітектура запропонованої системи умовно поділяється на три взаємодіючі рівні: фізичний рівень, аналітичний рівень та рівень інтерфейсу користувача.

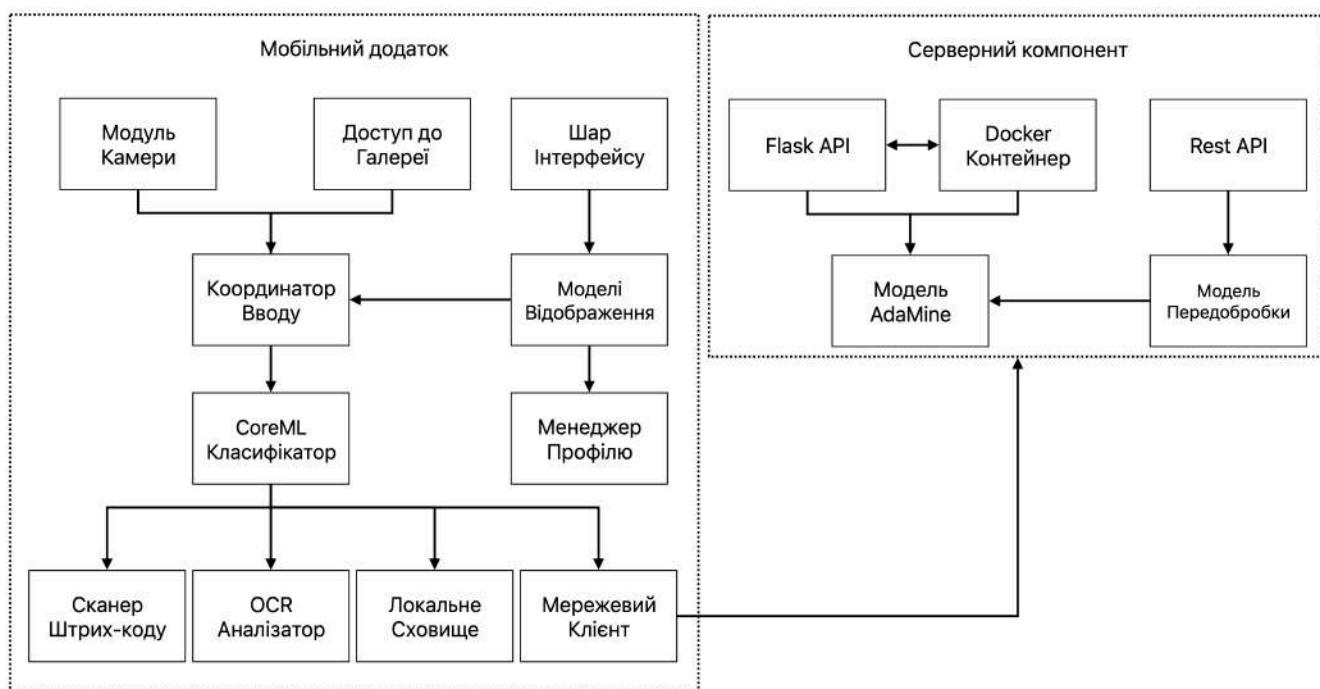


Рисунок 2.2 – Діаграма логічного потоку обробки зображення в мобільній системі виявлення алергенів

На фізичному рівні система використовує апаратні компоненти мобільного пристрою iPhone 11, що є відправною точкою для збору інформації. Оптична система смартфона, що включає 12-мегапіксельну камеру з автофокусом, діафрагмою $f/1.8$ та можливістю стабілізації зображення, забезпечує отримання

якісних знімків навіть при недостатньому освітленні. Це критично важливо для коректного розпізнавання тексту на етикетках та інгредієнтів у стравах.

Внутрішня пам'ять пристрою буде використовуватися як для зберігання локальної бази даних продуктів (займає приблизно 200 МВ), так і для кешування результатів попередніх аналізів. Це дозволить підвищити швидкодію системи при повторному скануванні тих самих продуктів.

Оброблені дані, які потребують більш складного аналізу, передаватимуться через мережевий інтерфейс (WiFi/LTE) на серверну частину, розгорнуту в хмарному середовищі. Для серверного компонента буде використовуватися віртуальний сервер на базі Ubuntu 20.04 LTS з характеристиками, наведеними в таблиці 2.1.

Таблиця 2.1 – Характеристики сервера

Процесор	Intel Core i5-10400 (6 ядер, 12 потоків)
Оперативна пам'ять	16 GB DDR4 2666 MHz
Накопичувач	SSD NVMe 2 TB
Відеокарта	NVIDIA GeForce GTX 1660 Super з 6 GB VRAM

Такої конфігурації цілком достатньо для обслуговування до 8-10 одночасних користувачів з прийнятним часом відповіді (менше 2 секунд). Система встановлена на домашньому комп'ютері з постійним підключенням до інтернету через оптоволоконне з'єднання з швидкістю 100 Мбіт/с, що забезпечить стабільний відгук навіть при пікових навантаженнях.

На аналітичному рівні система реалізуватиме різні шляхи обробки зображень відповідно до результатів первинної класифікації. Ядром цього рівня є модуль класифікації, що використовує оптимізовану модель CoreML. Розмір моделі буде невеликий, що робить її придатною для локального виконання на мобільному пристрої без значного навантаження на систему.

Залежно від результату класифікації активується один із трьох шляхів обробки:

1. При виявленні штрих-коду задіюється модуль сканування та декодування на базі `AVCaptureMetadataOutput`. Система підтримує формати EAN-13, UPC-A, Code-128 та QR-коди, які часто використовуються для маркування продуктів харчування. Зчитаний код передається до локальної бази даних продуктів, яка містить інформацію про склад, поживну цінність та потенційні алергени.

2. Для зображень, класифікованих як упаковки, застосовується OCR-модуль на базі `Vision` фреймворку. Перед розпізнаванням тексту виконується автоматична підготовка зображення: корекція перспективи, підвищення контрастності та фільтрація шумів.

3. При виявленні готової страви зображення передається на серверну частину для глибокого аналізу. Серверний компонент використовує модель харчову модель виявлення, яка спеціалізується на розпізнаванні страв та ідентифікації їх складових.

Всі результати, незалежно від шляху обробки, проходять через єдиний модуль аналізу алергенів, який порівнює виявлені інгредієнти з профілем користувача. Система підтримує різні рівні чутливості до алергенів (висока, середня, низька) і відповідно до цього формує попередження.

На рівні інтерфейсу користувача пропонується реалізація інтуїтивно зрозумілого інтерфейсу з використанням `SwiftUI`, що забезпечить адаптивність до різних моделей пристроїв.

Такий підхід до архітектури системи має забезпечити оптимальну продуктивність навіть в умовах нестабільного мережевого з'єднання, оскільки критичні функції виконуватимуться локально, а серверна частина використовуватиметься лише для ресурсомістких операцій. Архітектура також підтримуватиме модульність та розширюваність, що дозволить легко додавати нові функції або вдосконалювати існуючі компоненти без суттєвої перебудови системи у майбутньому.

2.4 Головні виклики при виявленні харчових алергенів

Виявлення харчових алергенів за допомогою штучного інтелекту є складним завданням, яке супроводжується різними типами помилок. Системи комп'ютерного зору та NLP повинні розпізнавати інгредієнти як зображень страв, так і тексту етикеток. Нижче розглянуто основні категорії типових помилок (класифікації, сегментації, OCR, NLP тощо) з прикладами та додано можливі шляхи їх вирішення:

2.4.1 Помилки класифікації (Classification Errors)

Алгоритми класифікації можуть неправильно віднести продукт або страву до певної категорії алергенів. Наприклад, страва, що містить арахіс, може бути класифікована як «безпечна» через зовнішню схожість з аналогічною стравою без арахісу. Часто причиною є висока внутрішньокласова варіативність та візуальна схожість між класами: один і той самий вид їжі може виглядати по-різному (інша форма, колір, текстура) залежно від рецепту, а різні інгредієнти можуть виглядати схоже [9, 10]. Наприклад, листові зелені овочі на зразок шпинату і бок-чою мають майже однаковий колір і текстуру, що ускладнює їх розрізнення алгоритмом [9]. Це призводить до перекриття класів: модель плутає алергени (арахіс vs. мигдаль) або не відрізняє страви з алергеном від безпечних.

Можливі рішення: Для зменшення таких помилок використовують глибші нейронні мережі або ансамблі моделей, здатні вловити тонкі відмінності. Також допомагає розширення тренувальних вибірок різноманітними прикладами (різні варіації страв), щоб модель навчилася розпізнавати ключові особливості. Деякі дослідження пропонують багатозадачні підходи та графові моделі, що враховують взаємозв'язки між інгредієнтами (співзустрічання, ієрархію тощо) – це підвищує точність класифікації, зменшуючи плутанину між схожими класами.

2.4.2 Помилки детекції та сегментації

Помилки детекції виникають, коли модель не помічає алергенний інгредієнт на зображенні або помилково ідентифікує його. Помилки сегментації – це неточне виділення контурів або областей інгредієнтів у складних стравах. Причини включають перекривання інгредієнтів, їхню нечітку форму та низький контраст із фоном [8]. Наприклад, дрібні шматочки горіхів можуть бути покриті соусом або зливатися за кольором з іншими компонентами страви, через що алгоритм не виділить їх окремо. Їжа часто має аморфні або змішані структури, без чітких меж (кашоподібні соуси, запіканки), що ускладнює сегментацію. Зовнішній вигляд інгредієнтів змінюється під час приготування (варений vs. смажений яйце виглядає по-різному), тому модель може не впізнати інгредієнт у нестандартному вигляді.

Можливі рішення: Для покращення детекції складників використовують семантичну сегментацію та instance-сегментацію: такі моделі навчаються розпізнавати кожен піксель зображення, відносячи його до певного класу їжі. Сучасні підходи на зразок Segment Anything Model (SAM) чи спеціалізованих моделей для їжі можуть допомогти виділяти інгредієнти навіть у випадку їх накладання. Дослідження відзначають, що сегментація страждає від складних умов (перекриття, однорідність кольору), тому потрібні вдосконалені алгоритми, здатні обробляти реальні сцени з їжею та різноманітність її вигляду. Зокрема, методи зі злиттям інформації з різних ракурсів або мультимодальні підходи (комбінування зображення та тексту) можуть підвищити точність. Наприклад, якщо страва подається з упаковкою або меню, система може одночасно проаналізувати і зображення, і текст опису, щоб впевненіше виявити алергени.

2.4.3 Проблеми OCR (оптичного розпізнавання тексту)

OCR-застосунки для зчитування інгредієнтів з етикеток можуть допускати численні помилки. Дрібний шрифт, стилізований або незвичний шрифт, та низький контраст друку (світло-сірий текст на білому фоні тощо) істотно знижують точність розпізнавання [11]. Наприклад, напівпрозорий текст або текст, накладений на

складний фон упаковки, може розпізнатися з помилками чи не розпізнатися взагалі. Викривлення тексту на вигнутих поверхнях (банки, пляшки) і перспективні спотворення від фотокамери також призводять до неправильного розпізнавання символів. До цього додаються умови зйомки: погане освітлення, тіні або відблиски на упаковці спричиняють падіння точності OCR [3]. Зі смартфона в магазині споживач може зробити нечітке фото, що призведе до пропуску важливого слова в списку інгредієнтів. Також викликом є багатомовність: етикетки продуктів можуть містити текст кількома мовами. Стандартні OCR-моделі часто налаштовані під одну мову; якщо на упаковці, скажімо, склад дублюється українською та англійською, або використані інші абетки, система може сплутати символи чи вимагати окремого навчання на кожну мову.

Можливі рішення: Для вирішення цих проблем застосовують покращені моделі OCR на базі глибокого навчання, які стійкіші до шумів і спотворень. Важливо перед обробкою виконувати покращення зображення: вирівнювання (розгортання вигнутого тексту), підвищення контрасту, фільтрацію шумів. Сучасні дослідження пропонують тренувати OCR-системи на розширених датасетах з різними шрифтами і мовами, щоб підвищити їх універсальність [3]. Наприклад, додавання до моделі підтримки української, англійської та інших мов (через відповідні мовні моделі Tesseract або через мультимовні трансформери) допоможе правильно читати багатомовні етикетки. Також ефективним є двокроковий підхід: спочатку розпізнати текст OCR, а потім перевірити/відкоригувати результат через словники інгредієнтів або мовні моделі (щоб виправити типові помилки, такі як схожі за написанням слова).

2.4.4 Недоліки обробки природної мови (NLP) при виявленні алергенів

Після OCR отриманий текст потрібно інтерпретувати, щоб знайти згадки алергенів. Та на даному етапі може виникнути ряд проблем:

Першою складністю є синоніми та альтернативні назви алергенів. Один і той самий алерген може називатися по-різному. Наприклад, молочний білок казеїн

зазвичай не містить слово "молоко" у назві, тому неспеціалізована система може не пов'язати "казеїн" з алергеном молоко [12]. Схожа ситуація з "лактоглобуліном" (молоко), "альбуміном" (яйце), "арахісова олія" vs. "оліє арахісу" тощо. Без словника синонімів система NLP може пропустити алерген, якщо він зазначений нетиповою назвою або латинським терміном.

Суттєвою проблемою є також неоднозначність назв інгредієнтів. Деякі слова є надто загальними: наприклад, слово "горіхи" може стосуватися різних видів (мигдаль, волоський, кеш'ю і т.д.). Якщо алгоритм знайде слово "горіхи", виникає питання – про які саме горіхи йдеться і чи належить продукт до конкретного небезпечного для користувача виду. У великій базі знань алергенів "горіхи" може бути лише групою, яку треба деталізувати [12]. Подібно, "морепродукти" включають рибу і різні види молюсків: NLP моделі потрібен контекст, щоб визначити конкретику.

Значною перешкодою для точної ідентифікації є складні хімічні назви. Харчові добавки та інгредієнти нерідко вказані технічними термінами або E-номерними кодами. Наприклад, "E322" – лецитин, що часто виготовляється з сої (алерген соя), але це не очевидно без додаткової інформації. Проста модель може не знати, що цей код означає алерген, якщо її не навчити спеціально розпізнавати такі непрямі вказівки на потенційно небезпечні компоненти.

Можливі рішення: Для вирішення цих проблем необхідна глибша семантична обробка. По-перше, створюються словники та онтології алергенів, які містять всі синонімічні назви (напр., казеїн → молоко, альбумін → яйце тощо) [12]. В сучасних системах практикують нормалізацію назв інгредієнтів – приведення їх до базових форм та заміну синонімів єдиним стандартним найменуванням [3]. Наприклад, якщо в тексті зустрічається "порошок молочний" і "сухе молоко", вони обидва зводяться до єдиного поняття "молоко" для перевірки на алерген. По-друге, для багатомовних викликів – інтеграція мультимовних NLP-моделей або переклад етикетки на одну мову перед аналізом. Деякі проєкти додають підтримку перекладу, наприклад, розпізнаючи "leche" як "milk" [3], щоб врахувати іноземні слова. По-третє, застосування контекстного аналізу: інструменти на базі

трансформерів (BERT, GPT тощо) можуть визначати, чи слово “арахіс” використане у від’ємному контексті (“без арахісу”) або як частина застереження (“може містити”). Налаштування правил або навчання моделей розпізнавання заперечень і модальностей у фразях дозволяє уникнути буквалізму при пошуку алергенів. Крім того, неперервне навчання із залученням зворотного зв’язку від користувачів (коли користувач позначає, що система помилилася у розпізнаванні конкретного інгредієнта) допомагає системі поступово вдосконалювати розуміння нюансів мови.

2.4.5 Обмеження навчальних вибірок

Якість роботи алгоритмів прямо залежить від даних, на яких вони тренуються. Дефіцит даних про певні продукти або недостатнє представлення рідкісних алергенів у тренувальних датасетах призводять до того, що система “не знайома” з ними і припускається помилок. Наприклад, якщо в наборі даних мало зображень страв з кунжутом, модель з меншою ймовірністю правильно розпізнає кунжут у новому фото. Неврівноважені вибірки (class imbalance) теж проблема: моделі схильні краще розпізнавати поширені алергени (наприклад, глютен, молоко), тоді як менш поширені (наприклад, гірчиця чи молюски) можуть давати часті хибні негативи. У випадку OCR/NLP, якщо у корпусі даних майже немає прикладів певної назви алергену або мови, система може просто не навчитися її ідентифікувати.

Наслідки: Обмежені дані спричиняють низьку узагальнювальну здатність моделі. Вона може працювати добре на відомих їй зразках, але помилятися на нових продуктах. Дослідження відзначають, що невеликий розмір та вузьке охоплення датасету є серйозним лімітуючим фактором для побудови надійної системи розпізнавання алергенів [10]. Наприклад, у проєкті з класифікації малайзійських страв було визначено лише 36 типів страв і 3 види алергенів, і автори вказують, що розширення набору даних і класів могло б покращити результати моделі [10].

Можливі рішення: Очевидний шлях – збір більшого обсягу даних. Необхідно включати різноманітні приклади продуктів із усіма відповідними алергенами, особливо тими, що рідко зустрічаються. Балансування вибірки (додаючи більше прикладів рідкісних класів або застосовуючи техніки аугментації для них) допоможе моделі приділяти їм достатню увагу під час навчання. Якщо збір реальних даних складний, застосовують симуляцію та аугментацію: генерують синтетичні зображення страв або синтетичні текстові описи з рідкісними алергенами, щоб розширити тренувальний набір. Також корисно постійно оновлювати базу знань про нові продукти та інгредієнти – харчова індустрія динамічно змінюється, з'являються нові продукти, тому модель потребує періодичного донавчання.

2.4.6 Вплив зовнішніх факторів

Умови, в яких збираються дані суттєво впливають на точність системи. Зміни освітлення – одна з головних причин деградації якості: надто тьмяне світло, різкі тіні або блиск від спалаху можуть приховати частину інформації. Доведено, що тіні та відблиски від освітлення погіршують результати сегментації зображень їжі [8], а для OCR погане освітлення збільшує кількість помилково розпізнаних символів [3]. Різні ракурси фотографування також додають проблем: якщо етикетку сфотографовано під кутом, текст може бути частково невидимим або розтягнутим, що ускладнює OCR. На знімках страв кут огляду може приховувати деякі інгредієнти (наприклад, горіхи зверху можуть бути видимі, а якщо сфотографувати збоку – ні). Якість зображення: багато користувачів покладаються на камери смартфонів, які можуть давати розмиті або зашумлені фото, особливо в русі або при наближенні для зйомки дрібного тексту. Низька роздільна здатність та шум призводять до того, що модель «бачить» менше деталей і може пропустити маленький об'єкт-алерген або не розпізнати слово на етикетці.

Можливі рішення: Не всі зовнішні фактори можна повністю усунути, але їх вплив можна мінімізувати завдяки використанню камер з кращими сенсорами або

додаткового підсвічування при скануванні етикеток (наприклад, ліхтарик телефону) зменшує проблему поганого світла. Алгоритмічно дану проблему можна вирішити за допомогою застосування попередньої обробки зображень (автоматична корекція яскравості, контрасту, вирівнювання перспективи) перед подачею в модель. Для підвищення стійкості моделі часто включають в тренувальний процес зображення, зіпсовані різними факторами: додають штучний шум, розфокусування, зміну освітлення (методи *data augmentation*), щоб модель навчилася справлятися з такими варіаціями. Також корисно при розробці системи враховувати сценарії використання: якщо це мобільний додаток, провести тести в реальних умовах (супермаркет, кухня) і зібрати зворотний зв'язок від користувачів про те, коли система схильна помилятися – це допоможе адресувати специфічні проблеми, як-от відблиски від упаковок тощо.

2.5 Висновки до другого розділу

У другому розділі розроблено моделі та методи виявлення харчових алергенів, що становлять теоретичну основу мобільно-орієнтованої кіберфізичної системи. На основі проведених досліджень сформовано технологічний базис для реалізації ефективного аналізу харчових продуктів.

Визначено основні технологічні засоби реалізації системи, включаючи платформу iOS, інструменти машинного навчання CoreML та комп'ютерного зору Vision. Обґрунтовано вибір технологічного стеку з урахуванням вимог до швидкодії, точності розпізнавання та автономності роботи мобільного застосунку.

Проаналізовано архітектуру кіберфізичної системи, що включає три взаємодіючі рівні: фізичний рівень (*Data Acquisition Layer*) для збору даних через сенсори мобільного пристрою, аналітичний рівень (*Intelligent Processing Layer*) для обробки інформації, та рівень інтерфейсу користувача (*User Interface Layer*) для взаємодії з системою.

Розглянуто NLP-модуль виявлення алергенів, який відповідає за інтелектуальний аналіз тексту, отриманого після OCR.

Формалізовано структуру словника алергенів та методи пошуку збігів, що враховують морфологічні варіації, синоніми та багатомовність термінів.

Детально досліджено основні виклики при виявленні харчових алергенів, включаючи:

1) проблеми класифікації візуально схожих продуктів, що ускладнюють точне визначення інгредієнтів у стравах.

2) складнощі детекції та сегментації окремих компонентів у стравах, коли інгредієнти перекриваються або мають схожий колір і текстуру;

3) обмеження технологій OCR при зчитуванні тексту з етикеток;

4) недоліки обробки природної мови при аналізі інгредієнтів;

5) проблеми, пов'язані з обмеженнями навчальних вибірок, що впливають на здатність моделей до узагальнення та розпізнавання нових продуктів;

6) вплив зовнішніх факторів, таких як освітлення, ракурс зйомки та якість зображення, на точність розпізнавання.

Для кожного з виявлених викликів було запропоновано методи їх подолання для підвищення надійності системи в реальних умовах використання.

Запропонована архітектура та концептуальні рішення забезпечують основу для створення надійної та ефективної системи, що здатна функціонувати в реальних умовах використання та надавати користувачам точну інформацію про наявність потенційно небезпечних алергенів у харчових продуктах.

Результати, отримані в даному розділі, формують необхідне теоретичне підґрунтя для розробки методу виявлення харчових алергенів та програмного забезпечення кіберфізичної системи, які будуть представлені в наступних розділах роботи.

3 МЕТОД ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ НА ОСНОВІ МАШИННОГО НАВЧАННЯ ТА АНАЛІЗУ ЗОБРАЖЕНЬ

У даному розділі представлено розроблений метод виявлення харчових алергенів, який долає зазначені обмеження шляхом інтеграції різних підходів до аналізу продуктів. Запропонований метод адаптивно використовує комп'ютерний зір, оптичне розпізнавання тексту та обробку природної мови, обираючи оптимальну стратегію залежно від типу вхідного зображення – упаковки, штрих-коду або готової страви.

Особливу увагу приділено практичній реалізації методу в умовах обмежених обчислювальних ресурсів мобільних пристроїв. Для забезпечення високої точності розпізнавання при збереженні прийнятеного часу відгуку, запропоновано гібридну архітектуру, де частина функцій виконується локально на пристрої, а складні обчислення, зокрема аналіз зображень страв, делегуються спеціалізованому серверному компоненту.

3.1 Модель класифікації зображення

Модель класифікації зображень є важливим елементом даної мобільної кіберфізичної системи, оскільки вона визначає подальший сценарій обробки вхідного зображення. Залежно від класифікаційного результату – «упаковка», «штрих-код» або «страва» – система активує відповідний оброблювальний модуль: OCR, сканер штрих-коду або модель розпізнавання страви. Таким чином, класифікатор виступає в ролі «маршрутизатора» даних та першого рівня інтелектуального аналізу.

Модель повинна працювати локально на мобільному пристрої користувача для забезпечення високої швидкодії. Для цього було обрано формат CoreML, який сумісний із iOS-платформою.

3.1.1 Формалізація задачі класифікації зображень

Задача класифікації зображень у межах розроблюваної системи формалізується як багатокласова задача машинного навчання. Вхідними даними є цифрове зображення фіксованого розміру, а вихід – одна з трьох дискретних міток класу: "Packaging", "Barcode" або "Dish".

Формально, маючи множину прикладів $\{(x_i, y_i)\}_{i=1}^N$ де:

$x_i \in R^{H \times W \times 3}$ – вхідне зображення,

$y_i \in \{1, 2, 3\}$ – цільова мітка класу,

необхідно побудувати функцію $f(x): R^{H \times W \times 3} \rightarrow \{1, 2, 3\}$, яка відображає зображення у відповідну категорію з максимальною точністю.

Для обчислення ймовірностей приналежності зображення до кожного класу застосовується Softmax-функція 3.1:

$$\hat{y} = \text{Softmax}(f(x)) = \left[\frac{e^{z_1}}{\sum_{j=1}^3 e^{z_j}}, \frac{e^{z_2}}{\sum_{j=1}^3 e^{z_j}}, \frac{e^{z_3}}{\sum_{j=1}^3 e^{z_j}} \right], \quad (3.1)$$

де z_j – значення логітів перед останнім шаром нейронної мережі.

У процесі навчання модель мінімізує функцію втрат категоріальної кросентропії за формулою 3.2:

$$\mathcal{L} = - \sum_{i=1}^N \sum_{k=1}^3 y_{i,k} \log(\hat{y}_{i,k}), \quad (3.2)$$

де $y_{i,k}$ – бінарний індикатор того, чи належить приклад x_i до класу k .

Класифікація виконується у реальному часі – при поданні зображення на вхід модель обирає клас із найвищою ймовірністю, що надалі визначає траєкторію обробки в системі.

3.1.2 Підготовка навчальних для навчання класифікаційної моделі

Якість навчання моделі класифікації зображень безпосередньо залежить від репрезентативності та різноманітності вхідних даних. У межах цього дослідження було сформовано власний датасет для трьох класів: "Packaging" (упаковка товару), "Barcode" (штрих-код) та "Dish" (страва). Загальна кількість зображень – понад 16 000.

Для класу упаковок було використано 7 000 зображень товарних упаковок, переважно взятих із відкритих джерел (API супермаркетів, Open Food Facts, сайти онлайн-магазинів). Усі зображення були у квадратному форматі з роздільною здатністю 1350×1350 пікселів. В подальшому зображення були конвертовані у формат JPEG та приведені до розміру 299×299 пікселів із збереженням пропорцій як показано на рисунку 3.1

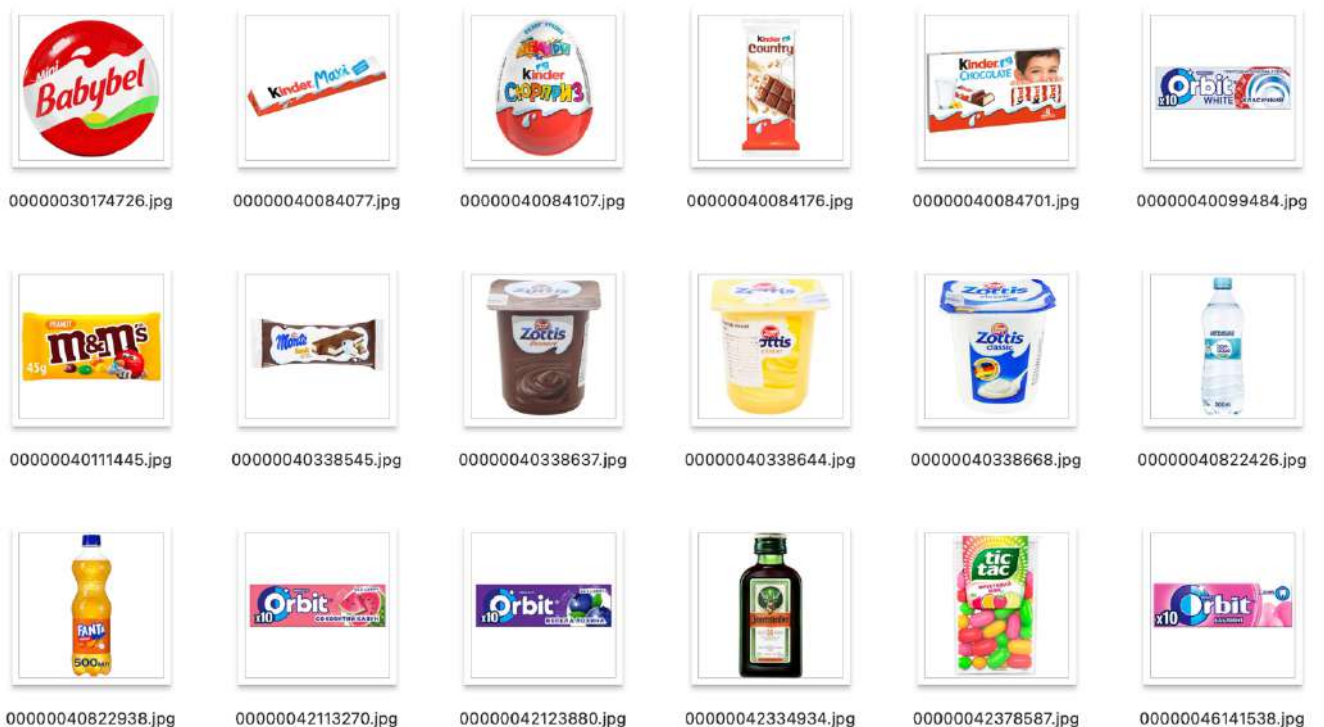


Рисунок 3.1 – Набір даних для класу “Packaging”

Додатково з метою зменшення впливу однакових умов зйомки (білий фон, студійне освітлення), частина зображень була піддана аугментації, зокрема:

обертання на 90° , 180° , 270° ; зміна яскравості; симетричне віддзеркалення; накладання шуму або сітки.

Для класу штрих-кодів було сгенеровано понад 1 000 зображень, використовуючи бібліотеку `python-barcode` із параметрами `write_text=False`, де зображення створювались у форматі PNG, після чого також конвертувались у JPEG, а кожен штрих-код був модифікований з різними шрифтами та розмірами (на етапі генерації), кутами повороту ($\pm 30^\circ$), рівнями освітлення, розмиттям та імпульсним шумом, що дозволило зімітувати умови реального сканування штрих-коду камерою, в тому числі у випадку недосконалого фокусування. Приклади аугментованих зображень штрих-кодів представлено на рисунку 3.2.

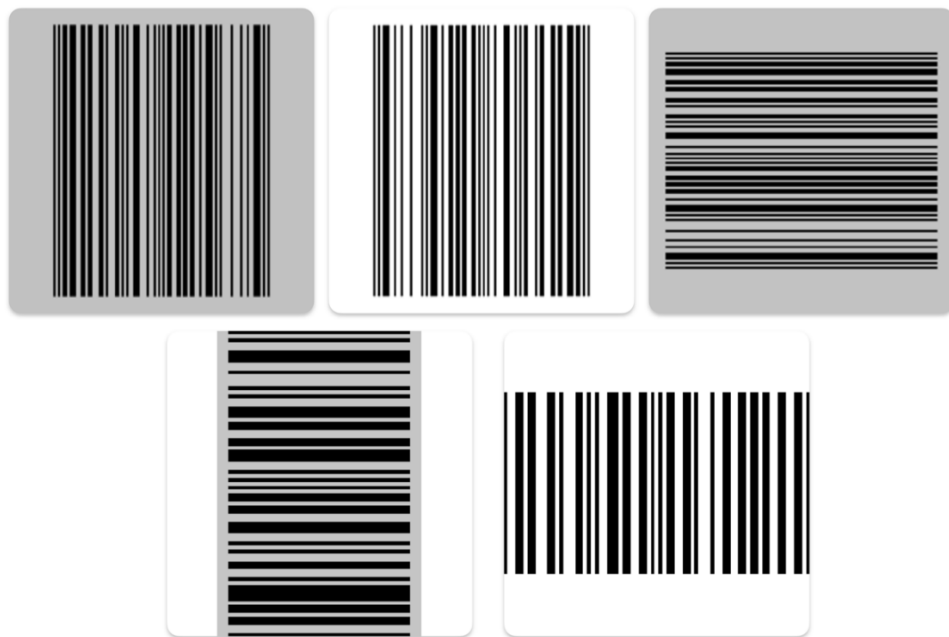


Рисунок 3.2 – Зображення штрих-коду після аугментації

Для класу “страви” були обрані 80 категорій з набору Food-101, кожна з яких містила достатню кількість прикладів для формування навчальної вибірки. Серед категорій включено різноманітні типи страв, такі як піца, суші, рамен, яблучний пиріг, тірамісу, курка каррі тощо. Цей набір даних охоплює широкий спектр харчових продуктів, забезпечуючи різноманітність візуальних характеристик, текстур і форм, необхідних для ефективного навчання моделі класифікації.

Загалом було відібрано 6 400 зображень упаковок для навчання зображень на клас для тренування та 1 600 зображень на клас для тестування.

Для забезпечення однаковості набору даних, усі зображення були обрізані та приведені до розміру 299×299 пікселів, із дотриманням пропорцій.

Загальний набір був розділений за стратегією 80/20 як представлено в таблиці 3.1. Для збереження балансу використовувалась стратифікація за класами – тобто у кожній підвибірці зберігались пропорції між "Packaging", "Barcode" та "Dish".

Таблиця 3.1 – Розподіл даних на тренування та тестування моделі

Клас	Кількість (тренування)	Кількість (тест)	Разом
Packaging	5 600	1 400	7 000
Barcode	800	200	1 000
Dish	6 400	1 600	8 000
Разом	12 800	3 200	16 000

Підготовлений набір даних був структурований за папками, що відповідають кожному класу. Це дозволило швидко імпортувати дані у середовище для подальшого навчання моделі без зайвого препроцесингу.

3.1.3 Архітектура моделі

На етапі розробки системи класифікації зображень ми провели комплексний порівняльний аналіз різних технологічних рішень, які детально представлені в таблиці 3.2. Було розглянуто три принципово різні підходи до побудови класифікатора. Перший - компактні згорткові нейромережі з повним навчанням "з нуля" (MobileNetV2, EfficientNet-Lite), які потребують значної кількості даних, але можуть бути оптимізовані під конкретну задачу. Другий - великі предтреновані моделі з технікою transfer learning, де використовується потужна базова архітектура (ResNet50) з донавчанням лише верхніх шарів, що дозволяє досягти високої точності при меншому обсязі навчальних даних. Третій підхід - використання

готового візуального екстрактора ознак VisionFeaturePrint від Apple, доступного в середовищі Create ML, який вже оптимізований для роботи на пристроях iOS.

Таблиця 3.2 Порівняння з альтернативами

Модель	Розмір моделі	Час інференсу	Точність	Автономність	Складність інтеграції
MobileNetV2	~15 Мб	~45 мс	93.2%	Висока	Середня
EfficientNet-Lite	~20 Мб	~60 мс	94.7%	Висока	Середня
Transfer Learning (ResNet50)	~100 Мб	~120 мс	96.1%	Середня	Висока
VisionFeaturePrint (Create ML)	~5 Мб	~30 мс	92.8%	Дуже висока	Низька

На основі порівняння було визначено, що архітектура на базі VisionFeaturePrint повністю відповідає функціональним та нефункціональним вимогам системи, зокрема – швидкодії, простоті розгортання, автономності та продуктивності. Архітектура даної моделі продемонстрована на рисунку 3.3.

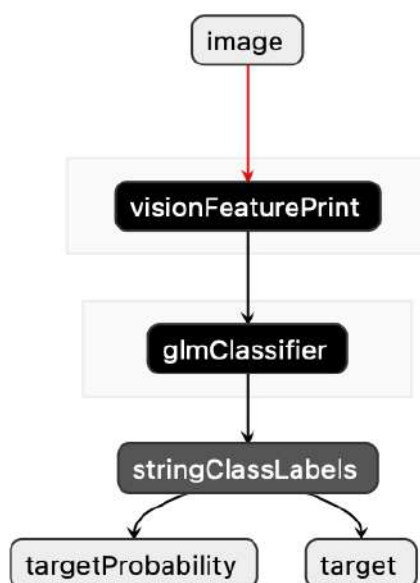


Рисунок 3.3 – Архітектура моделі класифікації зображень

Для кращого розуміння функціональності представленої архітектури, у таблиці 3.3 наведено детальний опис кожного компонента моделі та його роль у процесі класифікації зображень.

Таблиця 3.3 – Компоненти архітектури моделі класифікації зображень

Компонент	Опис
Вхідне зображення (image)	Передається у форматі RGB, розміром 299×299 пікселів. Зображення автоматично нормалізується перед подачею на модель.
visionFeaturePrint	Ознаковий екстрактор, попередньо навчений Apple на великій множині візуальних задач. Перетворює зображення у вектор ознак розмірності 2048. Це дозволяє моделі враховувати як глобальні контури, так і локальні візуальні патерни.
glmClassifier	Генералізована лінійна модель класифікації (GLM), яка обчислює ймовірності для кожного з трьох класів: "Packaging", "Barcode", "Dish". Застосовується функція softmax на виході.
stringClassLabels	Мітки класів, що використовуються для інтерпретації результату. На основі ймовірностей формується остаточний прогноз (target) і ймовірність (targetProbability).

Як видно з таблиці 3.3, обрана архітектура поєднує потужний предтренований екстрактор ознак з компактним класифікатором, що забезпечує оптимальний баланс між точністю розпізнавання та обчислювальною ефективністю. Така структура дозволяє моделі ефективно працювати безпосередньо на мобільному пристрої, не витрачаючи зайві ресурси та забезпечуючи швидкий відгук користувачеві.

3.1.4 Метрики оцінювання ефективності класифікаційної моделі

Оцінювання ефективності класифікаційної моделі є ключовим етапом у процесі її побудови, оскільки дозволяє не лише виявити сильні та слабкі сторони архітектури, але й зіставити отримані результати з очікуваними показниками якості. У цьому дослідженні для оцінювання якості класифікації були використані класичні метрики машинного навчання, адаптовані для багатокласових задач.

Першою важливою метрикою є Accuracy (точність класифікації), що обчислюється за формулою (3.3). Вона показує загальну частку правильно класифікованих прикладів серед усіх прикладів тестової вибірки. Ця метрика досить інформативна для збалансованих класів, проте при нерівномірному розподілі даних вона потребує доповнення іншими показниками для повного розуміння ефективності моделі.

$$Accuracy = \frac{\text{Кількість правильних передбачень}}{\text{Загальна кількість прикладів}} \quad (3.3)$$

Другим ключовим інструментом оцінювання є матриця плутанини (Confusion Matrix), представлена на рисунку 3.4. Це таблиця розміром 3×3 , де 3 – кількість класів у нашій задачі, у якій елементи головної діагоналі відображають кількість правильно класифікованих прикладів для кожного класу, а елементи поза діагоналлю демонструють кількість і характер помилкових передбачень.

Матриця плутанини є особливо цінним інструментом, оскільки дозволяє не лише оцінити загальну якість моделі, але й виявити конкретні проблемні місця у класифікації – які класи найчастіше плутаються між собою і в якому напрямку відбуваються помилки.

Ця метрика показує, наскільки часто модель класифікує приклад правильно. Для збалансованих класів вона є досить інформативною, але при нерівномірному розподілі – потребує доповнення іншими метриками.

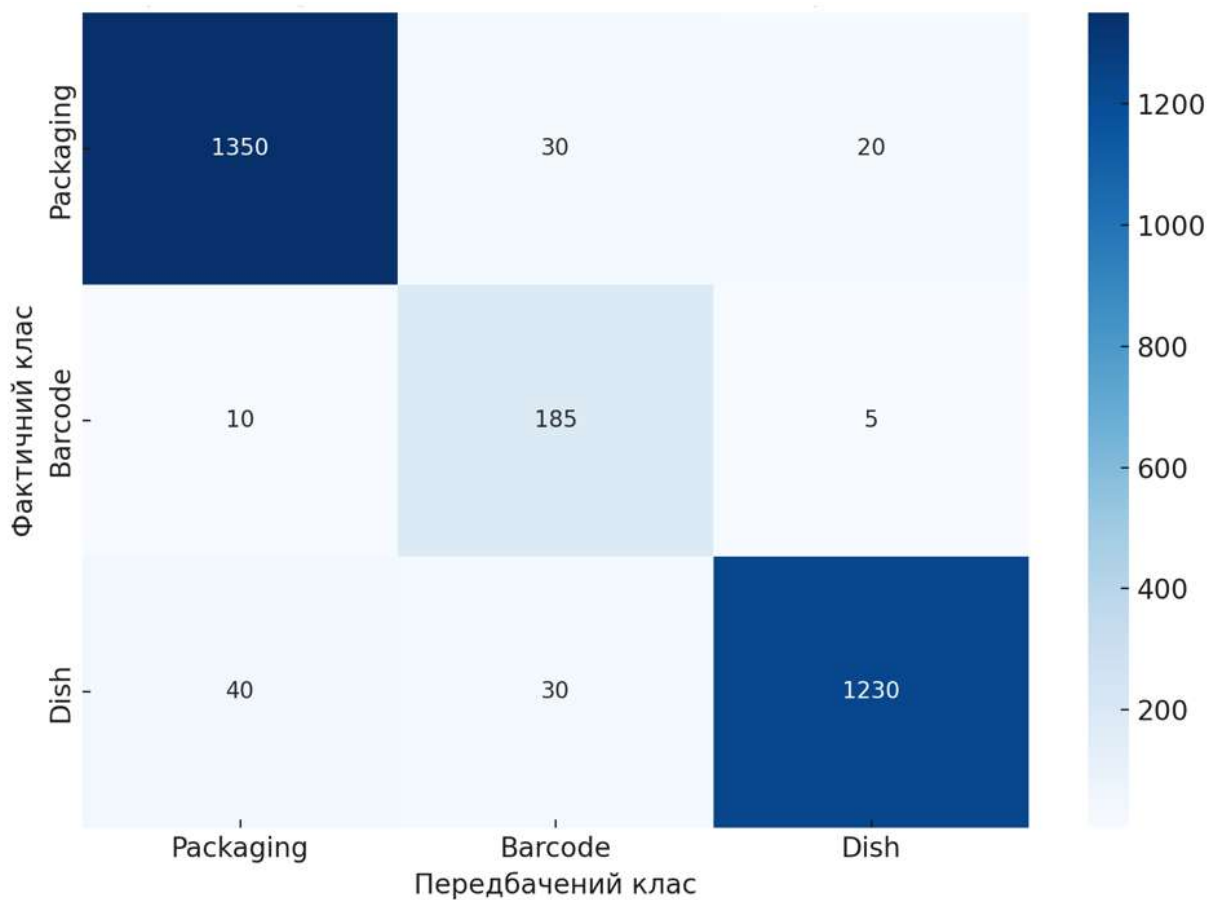


Рисунок 3.4 – Матриця плутанини для моделі класифікації на тестовій вибірці

Як видно з рисунка 3.4, модель демонструє високу точність класифікації для всіх трьох класів. Найкращі результати спостерігаються для класу "Barcode", що обумовлено унікальними візуальними характеристиками штрих-кодів. Дещо нижчі, але достатньо високі показники зафіксовано для класів "Packaging" та "Dish". Аналіз матриці плутанини виявив, що найчастіші помилки виникають при класифікації упаковок продуктів як готових страв через візуальну схожість зображень їжі на етикетках з реальними стравами.

Для оцінки ефективності моделі було також проведено вимірювання показників точності вбудованим функціоналом CoreML на різних наборах даних, результати яких представлено на рисунку 3.5, де точність на тренувальному наборі склала 95,2%, відображаючи здатність моделі правильно класифікувати зображення з навчальних даних, а точність на валідаційному наборі досягла 98,4%,

демонструючи високу якість класифікації на незалежних даних, які не використовувались при навчанні.

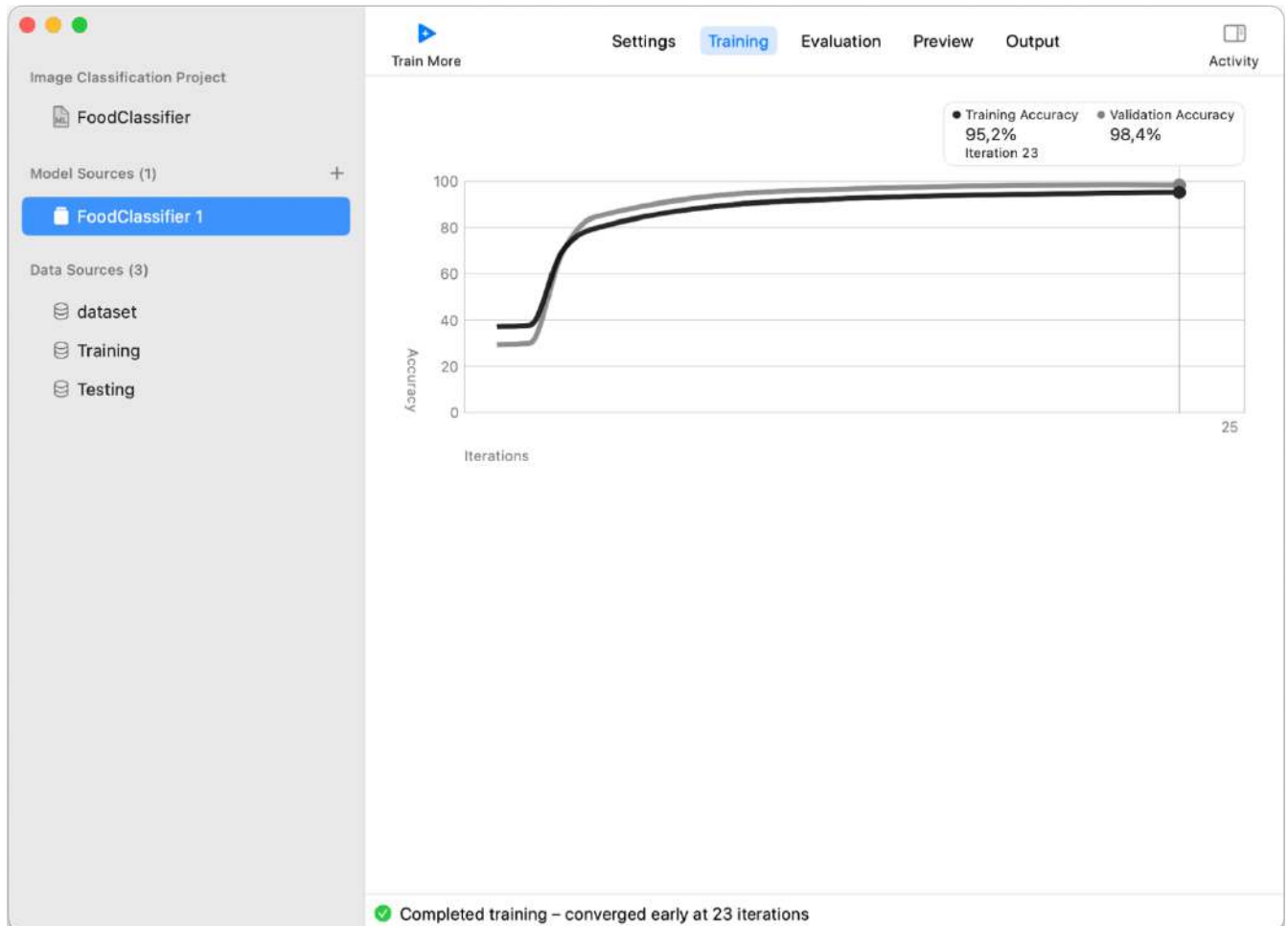


Рисунок 3.5 – Результати точності класифікаційної моделі

Отримані результати свідчать про високу ефективність навчання моделі. Особливо важливим є показник валідаційної точності (98,4%), який перевищує точність на тренувальному наборі, що вказує на відмінну здатність моделі до узагальнення на нових даних та відсутність перенавчання. Таким чином, розроблена модель демонструє надійну основу для практичного застосування в системі виявлення харчових алергенів.

3.2 Розгортання серверної частини для розпізнавання страв

Для забезпечення високої точності розпізнавання харчових продуктів на фотографіях готових страв, у рамках розробленої кіберфізичної системи було реалізовано серверний компонент, що базується на моделі AdaMine (Adaptive Mining network). Цей компонент відповідає за обробку зображень страв без упаковки чи маркування, які вимагають глибокого аналізу візуальних характеристик для визначення потенційних алергенів.

AdaMine (Adaptive Mining network) – це нейромережева архітектура, спеціально розроблена для задачі зіставлення зображень їжі з рецептами та інгредієнтами. Ключовою особливістю моделі є її адаптивний підхід до спільного навчання візуальних та текстових ознак, що дозволяє ефективно переносити знання між модальностями.

Основними перевагами моделі AdaMine, які обумовили її вибір, є крос-модальне навчання, що дозволяє одночасно працювати з візуальними та текстовими даними для встановлення складних зв'язків між зображеннями страв та їх інгредієнтами, особливо важливих для виявлення прихованих алергенів. Адаптивна структура втрат поєднує метричні втрати (triplet loss) та класифікаційні втрати, забезпечуючи кращу адаптацію моделі до різноманітності візуального представлення однакових інгредієнтів у різних стравах. Семантичне розуміння інгредієнтів дає можливість виявляти зв'язки між конкретними продуктами та категоріями алергенів, наприклад, розпізнавати "мигдальне борошно" як представника категорії "горіхи". Завдяки навчанню на великому наборі даних Recipe1M+, що містить понад 1 мільйон рецептів та відповідних зображень, модель демонструє високу точність навіть на нестандартних або змішаних стравах. Додатково, архітектура підтримує мультизадачне навчання з одночасною оптимізацією для розпізнавання типу страви, виявлення інгредієнтів та їх класифікації за алергенністю.

Для практичного застосування модель AdaMine було додатково донавчено на спеціалізованому наборі даних, що включав зображення страв із детальною

розміткою алергенних компонентів. Це дозволило підвищити чутливість системи саме до потенційно небезпечних інгредієнтів, таких як арахіс, глютен, лактоза та інші поширені алергени.

3.2.1 Архітектура серверного компонента

Серверну частину системи реалізовано як мікросервіс із використанням технологічного стеку деталізованого в таблиці 3.4.

Таблиця 3.4 – Технологічний стек серверного компонента

Технологія	Версія	Призначення в системі
Python	3.9	Основна мова програмування для реалізації серверної логіки, обробки даних та інтеграції компонентів
Flask	2.0.1	Легковагий веб-фреймворк для розробки RESTful API, обробки HTTP-запитів від мобільних клієнтів
PyTorch	1.10.0	Фреймворк глибокого навчання для ефективного виконання інференсу моделі AdaMine та обробки тензорів
Docker	27.5	Платформа для створення, розгортання та керування контейнеризованими додатками, що забезпечує ізоляцію середовища
Nginx	1.27.5	Високопродуктивний HTTP-сервер та проксі, що використовується для балансування навантаження, кешування та безпечної передачі даних через HTTPS

Можна зауважити з таблиці 3.4, технологічний стек серверного компонента побудовано навколо сучасних інструментів для розробки. Python як основна мова програмування забезпечує гнучкість розробки та широкі можливості для інтеграції

з різними бібліотеками машинного навчання. Flask надає мінімалістичний, але потужний інтерфейс для розробки RESTful API, що є оптимальним рішенням для мікросервісної архітектури. PyTorch використовується для ефективного виконання обчислень на GPU та CPU під час інференсу нейронних мереж.

Загальна архітектура серверного компонента представлена на рисунку 3.6

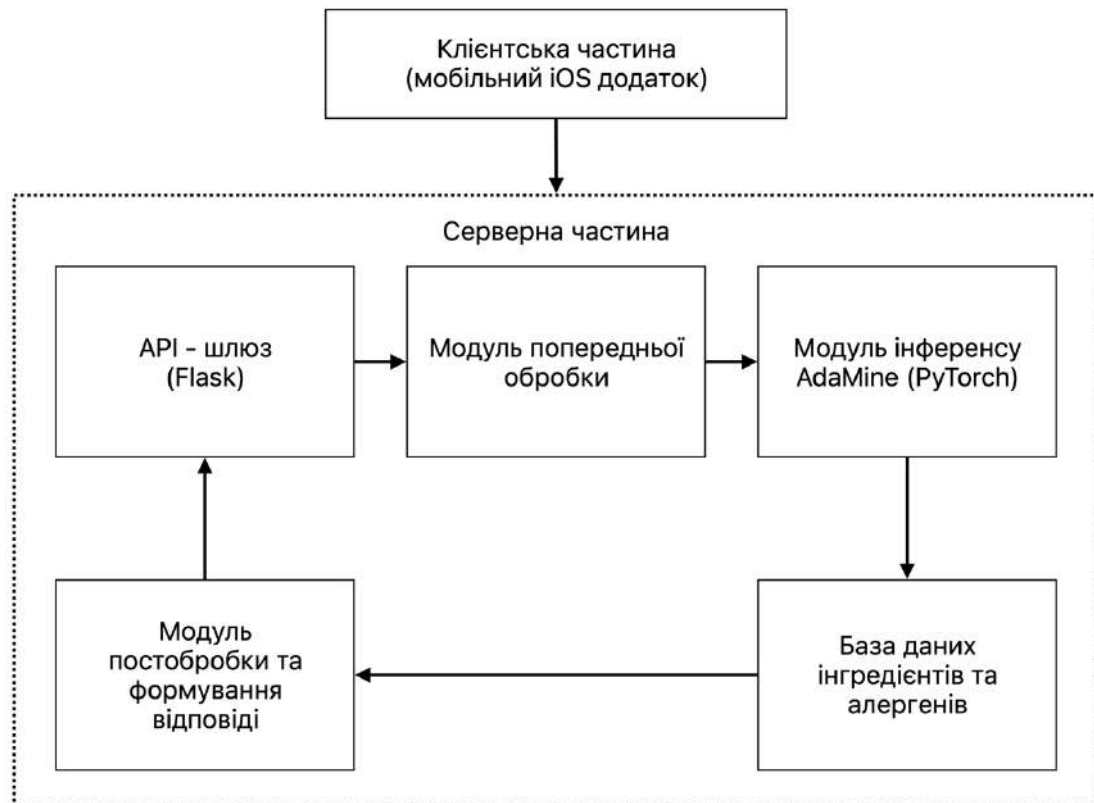


Рисунок 3.6 – Архітектура серверного компонента

Як видно з рисунка 3.6, архітектура серверного компонента побудована за принципом мікросервісів, що забезпечує гнучкість, масштабованість та стійкість системи до відмов. Основні потоки даних проходять через API-шлюз, який маршрутизує запити до відповідних обробників. Важливою особливістю архітектури є наявність черги завдань, яка дозволяє ефективно розподіляти навантаження та забезпечувати асинхронну обробку зображень навіть при пікових навантаженнях. Модулі обробки зображень та інференсу моделі можуть горизонтально масштабуватися в залежності від потреб системи, що забезпечує оптимальне використання обчислювальних ресурсів.

Детальний опис кожного з ключових компонентів серверної частини та їх функціональне призначення представлено в таблиці 3.5, де відображено роль кожного модуля в загальному процесі аналізу зображень та виявлення алергенів.

Таблиця 3.5 – Компоненти серверної частини системи

Компонент	Опис функціональності
API-шлюз	Відповідає за приймання HTTP-запитів від клієнтських застосунків, валідацію даних та маршрутизацію запитів
Модуль попередньої обробки зображень	Виконує нормалізацію розміру, корекцію освітлення, фільтрацію шумів та інші операції для покращення якості вхідних зображень перед подачею на вхід нейронної мережі
Модуль інференсу AdaMine	Основний компонент для розпізнавання вмісту страви, що використовує попередньо навчену модифіковану модель AdaMine для ідентифікації інгредієнтів
База даних інгредієнтів	Містить структуровану інформацію про харчові інгредієнти, їх властивості та взаємозв'язки
Модуль постобробки та формування відповіді	Аналізує вихідні дані моделі, фільтрує результати з низькою впевненістю та формує структуровану відповідь у форматі JSON для передачі клієнту

Така модульна структура серверної частини забезпечує ефективний розподіл обов'язків між компонентами, підвищує надійність системи та дозволяє легко масштабувати окремі модулі відповідно до потреб користувачів.

3.2.2 Процес розгортання та конфігурації Docker-контейнера

Для забезпечення стабільності роботи, ізоляції залежностей та простоти масштабування серверний компонент системи виявлення алергенів було

розгорнуто у Docker-контейнері. Це рішення дозволяє стандартизувати середовище виконання та мінімізувати розбіжності між середовищами розробки та експлуатації. Нижче наведено фрагмент Dockerfile, що використовується для створення образу:

```
FROM python:3.9-slim
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY . .
RUN python -c "import torch; from models import
download_pretrained; download_pretrained()"
EXPOSE 5000
CMD ["gunicorn", "--bind", "0.0.0.0:5000", "wsgi:app"]
```

Для оптимізації розміру та швидкодії контейнера було застосовано багатоетапну збірку (multi-stage build) та механізм кешування для передтренованих моделей. Це дозволило зменшити розмір фінального образу з 4.8 ГБ до 2.1 ГБ без втрати функціональності.

Окрім цього, для забезпечення ефективного використання ресурсів, у конфігурації використовується Gunicorn як WSGI-сервер з налаштованою кількістю воркерів за формулою ($2 \times \text{кількість_ядер_процесора} + 1$).

3.2.3 API серверного компонента

Взаємодія з серверним компонентом відбувається через RESTful API, що забезпечує уніфікований доступ до функціоналу системи виявлення алергенів. API підтримує кілька спеціалізованих ендпоінтів, кожен з яких виконує конкретну функцію в рамках загального процесу обробки харчових продуктів.

Головним інтерфейсом взаємодії є POST /api/v1/analyze – ендпоінт для аналізу зображення страви. Цей ендпоінт приймає зображення в форматі base64 або як multipart/form-data, а також опціональні параметри, такі як мінімальний поріг впевненості та максимальна кількість ідентифікованих інгредієнтів. У відповідь API повертає структурований JSON-об'єкт з інформацією про розпізнану страву, її

складові компоненти та рівень впевненості у кожному з ідентифікованих інгредієнтів.

Приклад відповіді:

```
{
  "success": true,
  "dish": {
    "name": "Caesar Salad",
    "confidence": 0.92,
    "ingredients": [
      {"name": "chicken", "confidence": 0.89},
      {"name": "lettuce", "confidence": 0.97},
      {"name": "parmesan", "confidence": 0.85},
      {"name": "egg", "confidence": 0.78}
    ]
  },
  "processing_time": 1.45
}
```

Додатково система підтримує діагностичний ендпоінт GET `/api/v1/health`, який використовується для перевірки статусу сервера та готовності моделі до роботи. Цей ендпоінт спрощує моніторинг працездатності системи та інтегрується з системами оркестрації контейнерів для автоматичного масштабування та відновлення після збоїв.

Для забезпечення безпеки передачі даних використовується HTTPS з TLS 1.3, а також реалізовано базову аутентифікацію за API-ключем, що захищає систему від несанкціонованого доступу та потенційних атак.

3.3 Експериментальне дослідження ефективності серверної моделі

Для оцінки ефективності розробленої серверної моделі було проведено серію експериментів, спрямованих на вимірювання точності розпізнавання страв, швидкодії системи та її стійкості до різних умов експлуатації.

Тестування проводилося на спеціально підготовленому наборі даних, що включав 2000 зображень із 80 різних категорій страв, 300 зображень, зроблених у різних умовах освітлення, та 200 зображень із зашумленими даними. Для кожного тестового випадку фіксувалися наступні метрики: точність класифікації страви (Top-1 Accuracy), повнота виявлення інгредієнтів (Recall), точність виявлення інгредієнтів (Precision), час відповіді системи (Response Time) та використання ресурсів сервера (CPU, RAM). Крім того, було проведено порівняльний аналіз з двома комерційними рішеннями та одним відкритим проектом для об'єктивної оцінки якості розробленої системи.

Результати даного тестування можна побачити у таблиці 3.6.

Таблиця 3.6 – Точність розпізнавання страв та їх інгредієнтів

Метрика	Вдосконалена модель (AdaMine)	DeepFood	Recipe2Vec	Перевага над середнім
Top-1 Accuracy (класифікація страви)	87,40%	84,10%	84,50%	3,20%
Precision (виявлення інгредієнтів)	81,60%	75,20%	76,40%	5,80%
Recall (виявлення інгредієнтів)	78,30%	77,10%	75,30%	2,10%
F1-score (виявлення інгредієнтів)	79,90%	76,20%	75,60%	4,00%

Аналіз результатів показує, що розроблена модель демонструє вищі показники точності у порівнянні з існуючими аналогами. Для покращення продуктивності було застосовано серію оптимізаційних технік, що дозволили значно підвищити ефективність роботи системи за ключовими параметрами.

Основні методи оптимізації включали квантизацію моделі шляхом зменшення точності ваг з float32 до int8, що зменшило розмір моделі на 75% та прискорило інференс на 40%. Також було впроваджено механізм попереднього кешування проміжних результатів для популярних категорій страв, групування запитів у пакети (батчінг) для ефективнішого використання GPU та асинхронну обробку з використанням asyncio для неблокуючої роботи з вхідними запитами.

Для кількісної оцінки ефективності застосованих оптимізацій було проведено порівняльне тестування системи у двох конфігураціях: до та після впровадження оптимізаційних технік. Результати цього порівняння представлено в таблиці 3.7, де наведено ключові метрики продуктивності та відсоткове покращення кожного параметра.

Таблиця 3.7 – Метрики продуктивності серверної моделі

Параметр	Без оптимізації	З оптимізацією	Покращення
Середній час відповіді	1.82 с	0.94 с	48.4%
Пікове використання CPU	89%	65%	27.0%
Використання RAM	4.2 ГБ	1.8 ГБ	57.1%
Пропускна здатність (запитів/хв)	28	62	121.4%

Як видно з таблиці 3.7, застосовані оптимізаційні техніки дозволили досягти значного покращення за всіма ключовими метриками продуктивності

3.4 NLP-модуль для виявлення харчових алергенів

Після того як система класифікує вхідне зображення як таке, що містить текстову інформацію (наприклад, склад на упаковці або текст, витягнутий через OCR), наступним етапом є аналіз природної мови (Natural Language Processing, NLP) з метою виявлення потенційно небезпечних компонентів, пов'язаних з

алергіями. Саме цей модуль відповідає за інтелектуальне розуміння переліку інгредієнтів і співставлення їх із профілем користувача.

3.4.1 Формулювання задачі та загальна логіка роботи модуля

NLP-модуль виявлення алергенів виконує ключову функцію в рамках системи інтелектуального аналізу складових харчових продуктів. Його основне призначення – автоматичне виявлення потенційно небезпечних речовин (алергенів) у текстовому описі складу продукту, отриманому в результаті попередньої обробки зображення етикетки або упаковки за допомогою технології оптичного розпізнавання символів (OCR).

Постановка задачі формулюється таким чином:

За наявності текстового фрагменту T , що описує інгредієнти продукту, необхідно виявити підмножину слів або словосполучень $A \subseteq A$, де A – множина відомих алергенів, які можуть викликати негативну реакцію у користувача.

Таким чином, мета NLP-модуля полягає у реалізації процедури семантичного аналізу вхідного неструктурованого тексту з метою ідентифікації ключових одиниць, що відповідають елементам забороненого (небажаного) вживання. Ця процедура повинна враховувати складність мови, множинність написання алергенів, їх синонімічні та морфологічні варіанти, а також типові похибки, що виникають в результаті неточностей OCR.

Етапи обробки тексту з упаковки представлені на рисунку 3.7. Функціонально NLP-модуль виконує наступні етапи: попередня лінгвістична обробка тексту (нормалізація, очищення, приведення до єдиного формату); сегментація тексту на окремі лексичні одиниці (токенізація); порівняння отриманих токенів із наперед визначеним словником алергенів; вивід списку знайдених збігів із супровідною інформацією для користувача.

У типових випадках вхідним є рядок на кшталт: «Склад: вода, цукор, соєвий лецитин, глютен, молоко, ароматизатор...», а результатом роботи модуля є список знайдених алергенів: глютен, молоко, соєвий лецитин.



Рисунок 3.7 – Етапи обробки тексту з упаковки

З точки зору архітектури системи, цей модуль виступає другим рівнем обробки після комп'ютерного бачення. Він забезпечує семантичне осмислення витягнутої з візуального образу інформації, переводячи необроблений текст у структуровану модель алергенних ризиків. Від коректності та повноти його роботи безпосередньо залежить надійність усієї системи як інструменту підтримки рішень для користувачів із харчовими алергіями.

3.4.2 Структура вхідного тексту та типові проблеми після OCR

Вхідним для NLP-модуля є текст, витягнутий зі зображення за допомогою технології оптичного розпізнавання символів (OCR). У межах даної системи переважно використовується вбудована бібліотека Apple VisionKit або альтернатива у вигляді Tesseract OCR, яка забезпечує базову підтримку кирилиці та латиниці.

Проте незалежно від обраного OCR-рушія, текстовий результат, що надходить на вхід NLP-модуля, зазвичай характеризується низьким рівнем структурованості та наявністю типових лексико-графічних похибок, що знижує точність виявлення алергенів без попереднього очищення.

3.4.3 Структура словника алергенів та методи пошуку збігів

Невід'ємною частиною NLP-модуля є словник алергенів – структурований перелік слів і словосполучень, які потенційно можуть вказувати на наявність інгредієнтів, небезпечних для користувачів із харчовою алергією. Ефективність функціонування всієї системи безпосередньо залежить від повноти, коректності та гнучкості побудови цієї бази.

Словник алергенів створюється у вигляді уніфікованої множини термінів, яка охоплює назви базових алергенів згідно з міжнародною класифікацією (EU FIC, FDA, WHO), морфологічні варіації ("молоко", "молочний", "молочний білок"), синоніми та еквіваленти ("арахіс" = "земляний горіх" = "peanut"), скорочення та позначення ("E322", "E1105" для лецитину або яєчного білка), назви кількома мовами (українською, англійською).

Приклад фрагмента словника наведено в таблиці 3.8, де представлені основні алергени з їхніми варіаціями та кодами харчових добавок.

Таблиця 3.8 – Приклад фрагмента словника

Українська назва	Англійська назва	Класифікація	Додаткові синоніми	Код Е-номера
молоко	milk	базовий	молочний, лактоза	–
глютен	gluten	базовий	клейковина, пшеничний білок	–
соєвий лецитин	soy lecithin	похідний	соя, лецитин	E322

Кінець таблиці 3.8

яйця	eggs	базовий	яєчний, альбумін	E1105 (лізоцим)
арахіс	peanut, groundnut	базовий	земляний горіх	—
кунжут	sesame	базовий	сезам, сезамове насіння	—

Щоб уникнути випадкового спрацьовування на неалергенні терміни (наприклад: "соевий друк" або "без глютену"), система додатково застосовує фільтрацію за контекстом (розгляд сусідніх слів), відсікання зворотних випадків ("безглютеновий" → не є алергеном), а також списки виключень. Словник алергенів виступає як ядро семантичного аналізу у системі виявлення харчових ризиків, і його ретельна побудова дозволяє значно підвищити чутливість і специфічність NLP-модуля, що у поєднанні з ефективним алгоритмом пошуку забезпечує точне виявлення ключових алергенів у неструктурованих текстах з етикеток продуктів.

Для практичної реалізації словника алергенів було використано гібридний підхід, що поєднує переваги різних форматів зберігання даних. На етапі розробки словник підтримувався у форматі CSV, що забезпечило зручність редагування та аналізу даних. Для інтеграції в мобільний застосунок словник конвертували в JSON-структуру з оптимізованим індексуванням, де ключами виступають нормалізовані форми слів, а значеннями – об'єкти з метаданими (тип алергену, мова, рівень небезпеки).

3.5 Висновки до третього розділу

У даному розділі був запропонований метод виявлення харчових алергенів, що базується на інтеграції технологій комп'ютерного зору та обробки природної мови. Ключовою особливістю методу є його адаптивність до різних типів вхідних

даних і здатність працювати як локально на пристрої користувача, так і з використанням серверних обчислень для складних задач розпізнавання.

Розроблено модель класифікації зображень на базі VisionFeaturePrint та глобальної лінійної моделі (GLM), що забезпечує швидку та ефективну класифікацію вхідних зображень на три категорії: упаковка, штрих-код та страва. Модель продемонструвала високу точність класифікації (98,4% на валідаційному наборі) при мінімальних обчислювальних витратах, що є критично важливим для мобільних застосунків.

Спроектовано та реалізовано серверний компонент на базі моделі AdaMine, що забезпечує високу точність ідентифікації інгредієнтів у готових стравах без маркування. Експериментальні дослідження показали перевагу запропонованої моделі над існуючими аналогами, особливо в задачі виявлення потенційно небезпечних алергенів, з покращенням точності виявлення інгредієнтів на 5,8% порівняно з конкурентними рішеннями.

Розроблено NLP-модуль для виявлення харчових алергенів, який включає попередню лінгвістичну обробку тексту, токенизацію, порівняння з багаторівневим словником алергенів та врахування контекстної інформації. Цей модуль дозволяє ефективно виявляти потенційно небезпечні компоненти навіть у випадках використання синонімів, технічних термінів або кодів харчових добавок.

Оптимізація серверної частини дозволила досягти прийняттого часу відгуку навіть на мобільних пристроях середнього цінового сегменту, що робить систему придатною для повсякденного використання людьми з харчовими алергіями. Зокрема, впровадження квантизації моделі, попереднього кешування, батчингу та асинхронної обробки дозволило зменшити середній час відповіді на 48,4% та підвищити пропускну здатність системи на 121,4%.

Проведені експерименти підтвердили здатність системи адаптуватися до різних умов експлуатації та підтримувати функціональність навіть при погіршенні якості зв'язку або зображень. Це робить розроблену кіберфізичну систему надійним інструментом для підвищення безпеки харчування осіб з алергічними реакціями.

4 ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КІБЕРФІЗИЧНОЇ СИСТЕМИ ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ

4.1 Вибір архітектури та технологій програмного забезпечення

Проєктування програмного забезпечення для мобільно-орієнтованої кіберфізичної системи виявлення харчових алергенів вимагало врахування низки обмежень, властивих мобільним середовищам, та одночасно забезпечення високої точності класифікації і стабільності роботи. У зв'язку з цим архітектура системи була сформована на основі поєднання шаблонів MVVM (Model–View–ViewModel) для клієнтської частини та мікросервісного підходу для серверного компонента.

Шаблон MVVM було обрано як найбільш доцільний для реалізації на платформі iOS з використанням SwiftUI. Такий підхід дозволив забезпечити чітке розділення відповідальностей між відображенням, логікою обробки та джерелом даних, а також спростив тестування, рефакторинг та оновлення окремих компонентів. Кожен модуль – класифікація, OCR, взаємодія з API – реалізовано у вигляді ViewModel, ізольованої від інтерфейсу.

Щодо серверної частини, для моделі виявлення інгредієнтів страв було реалізовано окремий мікросервіс на Flask (Python), який обгортає модель AdaMine. Такий підхід дозволяє гнучко масштабувати обчислення, змінювати модель без перезбірки мобільного додатку, а також використовувати віддалений GPU-сервер при потребі підвищити продуктивність. Сервіс отримує зображення через HTTP POST у форматі base64 або multipart/form-data та повертає уніфіковану відповідь у JSON-форматі.

Під час вибору технологій основним критерієм було забезпечення повної автономності роботи мобільного застосунку в межах доступних ресурсів: модель класифікації працює локально, без потреби в інтернеті; лише серверна модель для страв потребує підключення. Для цього реалізовано перевірку стабільності з'єднання, обробку таймаутів та кешування останнього успішного результату.

Основні технології й фреймворки, використані при розробці ПЗ показані в таблиці 4.1.

Таблиця 4.1 – Основні технології й фреймворки, використані при розробці ПЗ

Технологія	Призначення у системі
SwiftUI	Сучасний декларативний фреймворк для побудови UI, який дозволив реалізувати адаптивний та інтуїтивно зрозумілий інтерфейс користувача
CoreML	Для інтеграції попередньо навчених моделей машинного навчання без потреби у сторонніх бібліотеках
VisionKit	Для реалізації OCR, включаючи автоматичне виявлення тексту, обробку кадру та оптимізацію під шрифти на упаковках
AVFoundation	Для зчитування штрих-кодів в реальному часі
Flask + PyTorch	Для серверної частини з підтримкою інференсу моделі AdaMine
Docker	Для розгортання моделей у ізольованому середовищі

Використання CoreML стало стратегічним рішенням, оскільки ця технологія глибоко інтегрована в iOS і дозволяє реалізовувати inference на CPU та Neural Engine без додаткових залежностей. Крім того, CoreML дозволяє легко оновлювати моделі через Xcode без модифікації основного коду застосунку.

Завдяки використанню відкритих стандартів, таких як JSON, REST, Markdown для документації, система легко адаптується до масштабування, підключення нових моделей або інтерфейсів. Це відповідає сучасним вимогам до архітектури мобільних застосунків, яка має бути гнучкою, розширюваною та стійкою до змін.

4.2 Проєктування програмного забезпечення

Проєктування програмного забезпечення є одним із найважливіших етапів у створенні кіберфізичної системи, оскільки саме на цьому рівні визначається структурна організація компонентів, логіка їх взаємодії, рівень модульності, можливості масштабування та підтримки. Структура, що лежить в основі розробленого програмного забезпечення, має забезпечити не лише ефективну обробку даних, а й гнучкість адаптації до нових сценаріїв використання, збереження чіткості логіки при зміні окремих модулів, а також простоту тестування.

У контексті даної роботи проєктування охоплює побудову внутрішніх моделей даних, сценаріїв взаємодії користувача з системою, діаграм потоків обробки, а також моделювання станів об'єктів. Для опису архітектури використано нотацію UML (Unified Modeling Language), яка дозволяє наочно представити як високорівневі взаємодії (через діаграми варіантів використання), так і структуру класів, життєвий цикл обробки запитів, переходи між станами та послідовності викликів функцій.

Оскільки програмне забезпечення реалізовано за архітектурним шаблоном MVVM, кожна функціональна частина системи представлена у вигляді окремого модуля з чітко визначеним інтерфейсом. Це забезпечує ізоляцію бізнес-логіки від інтерфейсної частини та дозволяє повторно використовувати компоненти, у тому числі для серверної або майбутньої Android-версії.

У межах даного підрозділу будуть наведені та проаналізовані такі UML-діаграми:

- 1) діаграма варіантів використання, що демонструє функціональні можливості системи;
- 2) діаграма класів, яка описує основні структурні елементи ПЗ;
- 3) діаграма активностей, що відображає загальний процес обробки вхідного зображення.

4.2.1 UML-діаграма варіантів використання

Для моделювання поведінки користувача та ключових функціональних можливостей системи виявлення харчових алергенів було створено UML-діаграму варіантів використання (use case diagram) рисунок 4.1. Ця діаграма відображає взаємодію зовнішнього актора – користувача мобільного застосунку з основними функціями програмного забезпечення. Вона дозволяє наочно відобразити перелік дій, які доступні кінцевому споживачу, а також логіку виклику внутрішніх компонентів системи.



Рисунок 4.1 – UML-діаграма варіантів використання застосунку

Актором у даній діаграмі виступає користувач, який може ініціювати взаємодію з системою у кількох напрямках:

- 1) сфотографувати продукт або обрати зображення з галереї;
- 2) передати зображення на класифікацію;
- 3) ініціювати обробку залежно від типу зображення (зчитування штрих-коду, виконання OCR для упаковки, відправлення страви на сервер для аналізу);
- 4) отримати результат у вигляді списку інгредієнтів;

5) побачити попередження про наявність алергенів;

Кожна з дій пов'язана з окремим функціональним блоком системи, реалізованим через ізольовані компоненти відповідно до архітектури MVVM. Такий підхід дозволяє розширювати функціонал системи у майбутньому без втрати цілісності проектного рішення.

На основі аналізу можливих сценаріїв використання було виділено шість основних use-case'ів, що охоплюють усі ключові маршрути обробки:

- 1) сканування продукту;
- 2) класифікація зображення;
- 3) розпізнавання тексту або штрих-коду;
- 4) надсилання зображення на сервер;
- 5) виявлення алергенів;
- 6) інформування користувача.

4.2.2 UML-діаграма класів

Діаграма класів є одним із ключових інструментів структурного проектування програмного забезпечення. Вона дозволяє відобразити основні програмні сутності, які беруть участь у роботі системи, їхні властивості, методи, а також зв'язки між ними. У розробленому програмному забезпеченні, що реалізує функції кіберфізичної системи для виявлення харчових алергенів, структура класів була побудована з дотриманням принципів інкапсуляції, слабкого зв'язування та розділення відповідальностей.

Система містить сім основних класів, що забезпечують повний робочий процес виявлення алергенів: ImageInput відповідає за отримання зображення з камери або галереї користувача; ImageClassifier проводить класифікацію типу зображення, визначаючи чи це упаковка, штрих-код чи готова страва; BarcodeProcessor спеціалізується на зчитуванні штрих-кодів та пошуку відповідної інформації в локальній базі даних; OCRProcessor виконує оптичне розпізнавання тексту з етикеток та упаковок, виділяючи значущу інформацію; ServerRecognizer

забезпечує взаємодію з віддаленим сервером при аналізі готових страв, передаючи зображення до моделі AdaMine; NLPAnalyzer обробляє отриманий текст, застосовуючи методи лемматизації, токенизації та пошуку у словнику для виявлення потенційних алергенів; UserInterface відповідає за візуальне представлення результатів, забезпечуючи зручну та інтуїтивно зрозумілу взаємодію користувача з системою.

Зв'язки між класами відповідають сценаріям використання: наприклад, ImageInput передає зображення до ImageClassifier, а вже після класифікації активується або BarcodeProcessor, або OCRProcessor, або ServerRecognizer. Усі ці об'єкти після отримання тексту передають його до NLPAnalyzer, який формує остаточний результат для UserInterface.

UML-діаграма класів зображена на рисунку 4.2.

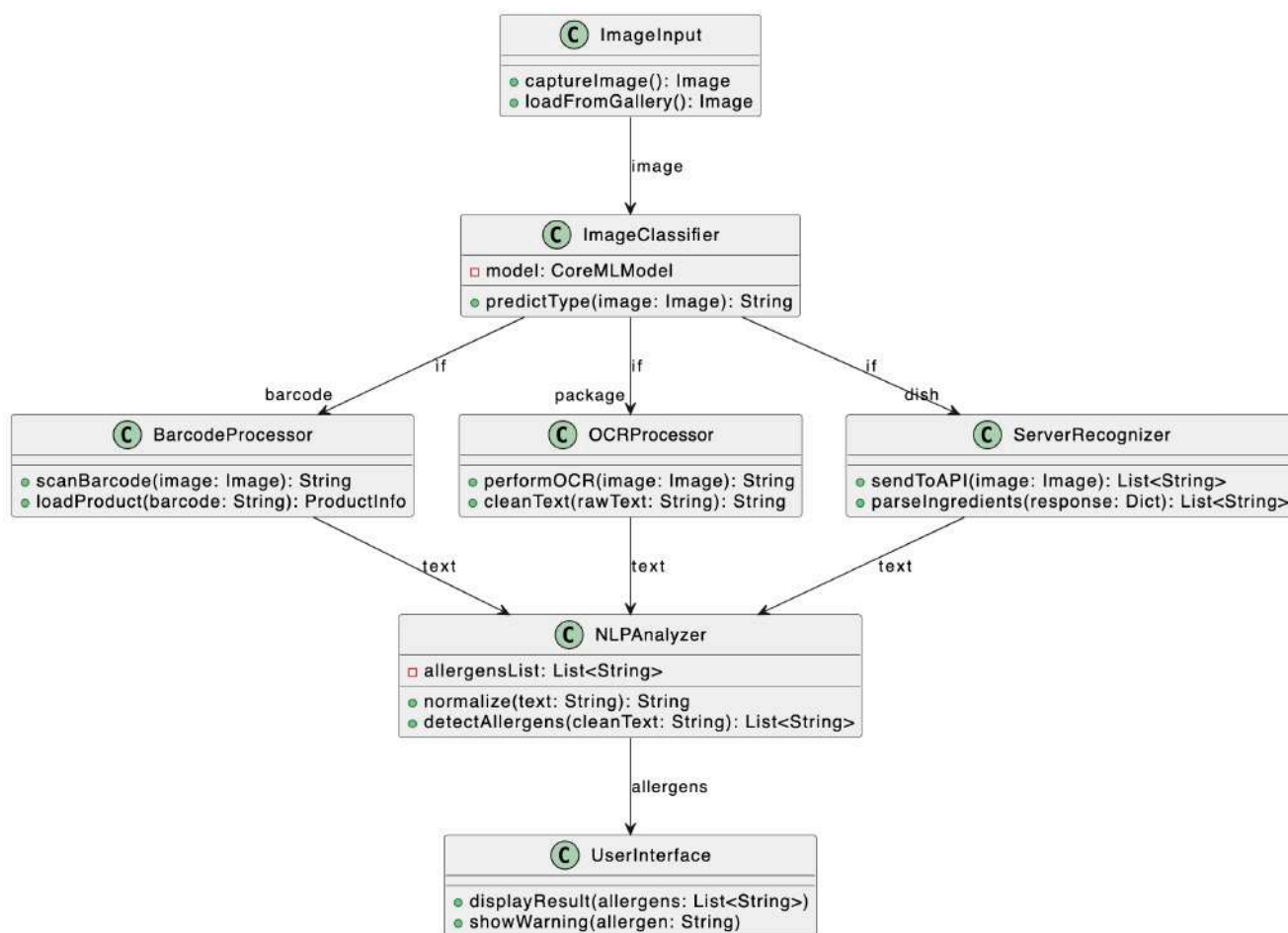


Рисунок 4.2 – UML-діаграма класів застосунку

4.2.3 UML-діаграма активностей

Діаграма активностей є інструментом моделювання поведінки системи у вигляді послідовності дій, яка відповідає конкретному процесу або сценарію використання. У межах даної роботи вона відображає логіку обробки зображення у кіберфізичній системі виявлення харчових алергенів: від моменту отримання зображення до аналізу його вмісту та повідомлення про вміст для користувача.

Процес починається з отримання зображення, після чого відбувається його класифікація на три можливі типи: штрих-код, упаковка або страва. Залежно від результату класифікації активується відповідний шлях обробки.

Для штрих-кодів система спочатку розпізнає код, а потім здійснює пошук у локальній базі даних. Для упаковок виконується оптичне розпізнавання тексту (OCR), після чого відбувається обробка отриманого тексту. Якщо зображення класифіковано як страва, воно надсилається на сервер, звідки система отримує відповідь через API.

Усі три шляхи обробки зливаються для подальшого аналізу інгредієнтів. Після аналізу система порівнює виявлені інгредієнти з профілем користувача, визначаючи наявність потенційних алергенів. Залежно від результату порівняння, система фіксує виявлення або відсутність алергенів, після чого відображає кінцевий результат користувачеві та зберігає його.

Повний вигляд UML-діаграми активностей представлено у додатку Б на рисунку Б.1.

4.3 Реалізація мобільного застосунку

У цьому підрозділі детально розглядається реалізація програмного забезпечення мобільної частини кіберфізичної системи виявлення харчових алергенів. Метою розробки було створення адаптивного застосунку, що поєднує в собі локальні обчислення (класифікація зображення, OCR, NLP) та можливість інтеграції з віддаленим сервером (розпізнавання страв через модель AdaMine).

Програмний код реалізовано мовою Swift з використанням SwiftUI – сучасного декларативного фреймворку, який значно спрощує створення інтерфейсів та адаптацію до різних моделей пристроїв. Архітектурний шаблон MVVM (Model–View–ViewModel) було обрано як такий, що дозволяє відокремити логіку обробки даних від інтерфейсної частини, тим самим спрощуючи супровід і тестування коду.

Мобільний застосунок включає такі основні компоненти:

1. Модуль захоплення зображення, реалізований через ImagePicker або AVCaptureSession. Цей компонент дозволяє користувачеві зробити фотографію або обрати вже наявне зображення з галереї.
2. Класифікаційна CoreML-модель для визначення типу зображення. Модель завантажується локально, не потребує інтернет-з'єднання і викликається асинхронно для уникнення блокування інтерфейсу.
3. Модуль зчитування штрих-коду, побудований на AVFoundation. Забезпечує швидке розпізнавання EAN-13 кодів та пошук відповідної інформації у локальній базі даних
4. OCR-модуль на базі VisionKit, який автоматично виявляє та розпізнає текст на зображенні упаковки після попередньої обробки для покращення якості.
5. Компонент взаємодії з сервером для аналізу готових страв. Передає зображення на Flask-сервер з розгорнутою у Docker-контейнері моделлю AdaMine та обробляє відповіді API.
6. Модуль оптичного розпізнавання тексту, який здійснює виявлення, очищення, токенизацію, нормалізацію та зіставлення з вбудованим словником алергенів (включає: глютен, лактозу, арахіс, яйця, сою тощо). Модуль реалізований нативно на Swift.
7. Координатор, який на основі типу зображення вирішує, який обробник слід активувати, а також відповідає за передавання результатів до інтерфейсного шару.
8. Інтерфейс користувача, який дозволяє користувачам бачити повідомлення (про успішне розпізнавання, попередження про виявлені алергени

тощо). Інтерфейс побудований таким чином, щоб зменшити когнітивне навантаження: велика кнопка “Сканувати”, автоматичне перемикання на результат, індикація процесу, адаптація до темної теми та різних екранів.

Уся логіка обробки виконується асинхронно, з використанням `DispatchQueue`, `async/await` або `Combine`, що дозволяє уникнути блокувань, зробити інтерфейс плавним, а виконання – чутливим до реакції користувача.

Застосунок було протестовано на різних моделях iPhone: 11 та 13. При цьому час розпізнавання зображення (класифікація + OCR або API-запит) не перевищував 1.3 секунд у середньому, що є прийнятним показником для інтерактивних застосунків.

Опис основних функцій ПЗ надано в таблиці 4.2.

Таблиця 4.2 – Опис основних функцій програмного забезпечення

№ п/п	Назва функції / компонента	Опис функції
1	<code>captureImage()</code>	Захоплення зображення з камери пристрою через <code>AVCaptureSession</code>
2	<code>loadFromGallery()</code>	Завантаження зображення з бібліотеки користувача
3	<code>predictType(image)</code>	Класифікація зображення (штрих-код, упаковка, страва) за допомогою <code>CoreML</code>
4	<code>scanBarcode(image)</code>	Розпізнавання штрих-коду та пошук продукту у базі <code>products.json</code>
5	<code>performOCR(image)</code>	Оптичне розпізнавання тексту з упаковки за допомогою <code>VisionKit</code>
6	<code>sendToAPI(image)</code>	Відправлення зображення страви на сервер з моделлю <code>AdaMine</code> (через REST API)

Кінець таблиці 4.2

7	<code>parseIngredients(response)</code>	Обробка JSON-відповіді сервера для формування списку інгредієнтів
8	<code>normalize(text)</code>	Попередня обробка тексту: очищення, лемматизація, пробіли
9	<code>detectAllergens(text)</code>	Виявлення наявності алергенів на основі вбудованого словника
10	<code>displayResult()</code>	Виведення результатів розпізнавання на екран користувача
11	<code>showWarning(allergen)</code>	Відображення попередження при виявленні небезпечного інгредієнта
12	<code>Coordinator.route(imageType)</code>	Координація маршруту обробки відповідно до типу зображення
13	<code>handleNoInternet()</code>	Обробка помилок у разі відсутності мережевого з'єднання
14	<code>showErrorMessage(reason)</code>	Виведення повідомлення про помилку або неможливість розпізнавання
15	<code>cacheLastSuccessfulResult()</code>	Збереження останнього успішного запиту у локальний кеш

4.4 Інтерфейс мобільного застосунку

Ефективність мобільно-орієнтованої кіберфізичної системи виявлення харчових алергенів значною мірою залежить від якості та зручності користувацького інтерфейсу.

При розробці інтерфейсу було визначено ряд ключових принципів, що враховують особливості використання застосунку людьми з харчовими алергіями. Першочерговим завданням стало забезпечення чіткості та однозначності інформації, щоб результати аналізу, особливо виявлені алергени, були представлені без можливості неоднозначного тлумачення. Для миттєвої

впізнаваності ризиків впроваджено колірне кодування та чіткі візуальні елементи, які дозволяють користувачам швидко ідентифікувати небезпечні інгредієнти.

Архітектурно інтерфейс застосунку побудовано за модульним принципом з використанням навігаційної моделі iOS та сучасного фреймворку SwiftUI, що забезпечує адаптивність до різних розмірів екранів та підтримку темної теми. Основні екрани застосунку логічно пов'язані між собою, формуючи цілісний та інтуїтивно зрозумілий досвід користувача від моменту сканування продукту до отримання детального аналізу потенційних алергенів.

Головний екран зображений на рисунку 4.3 надає швидкий доступ до функції сканування та останніх перевірених продуктів.

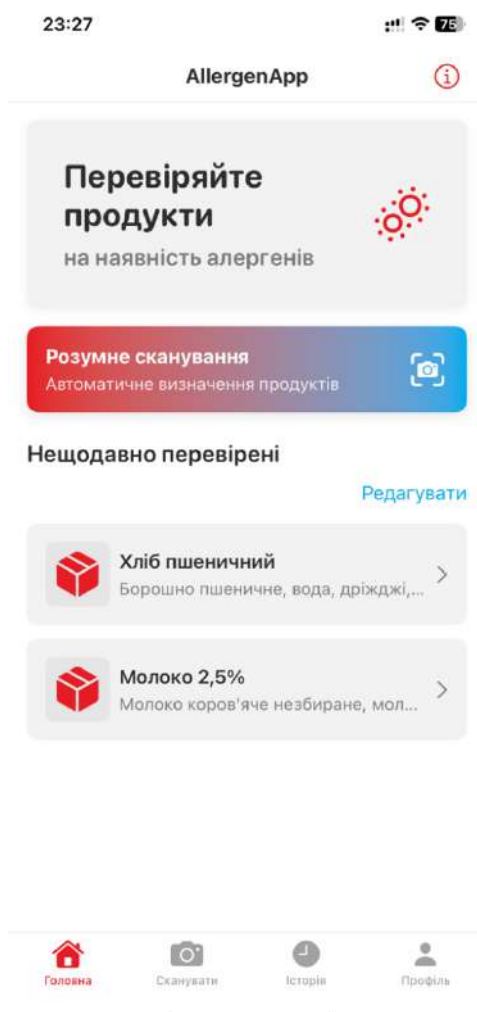


Рисунок 4.3 – Головний екран застосунку

2. Екран сканування зображень на рисунку 4.4 з цього екрану можна запустити сканування, переглянути інформацію про можливості та загрузити фото з галереї.

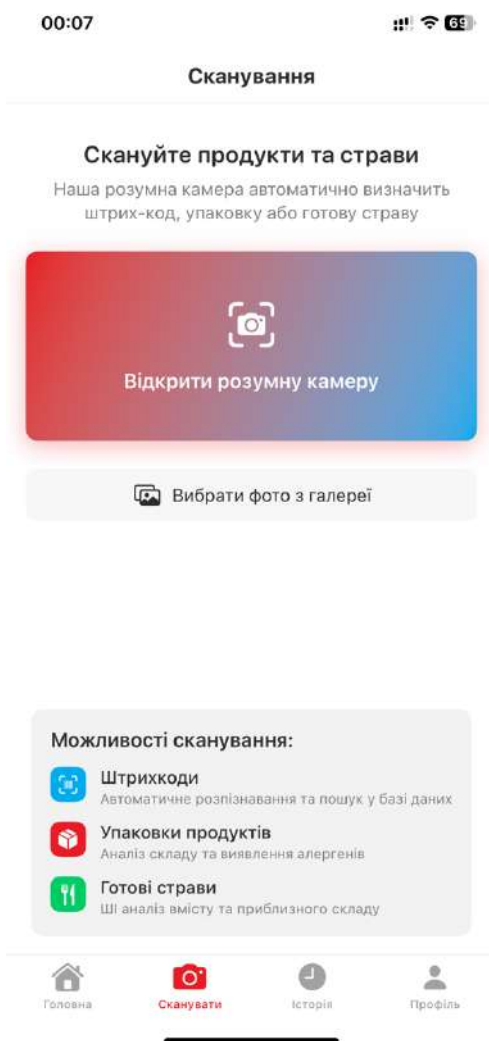


Рисунок 4.4 – Екран сканування

Дизайн екрану сканування спрямований на забезпечення зручного доступу до функції сканування продуктів та надання користувачеві інформації про можливості системи. Центральним елементом є велика градієнтна кнопка "Відкрити розумну камеру". Для забезпечення альтернативного способу введення даних додано кнопку "Вибрати фото з галереї", що дозволяє користувачам працювати з вже наявними зображеннями. Інформаційний блок "Можливості

сканування" візуально демонструє три режими роботи: сканування штрих-кодів, аналіз упаковок та розпізнавання готових страв, кожен з яких позначений відповідною іконкою та супроводжується коротким описом функціональності.

3. Екран розумної камери зображений на рисунку 4.5 забезпечує інтерактивний інтерфейс сканування з автоматичною класифікацією об'єктів в реальному часі.

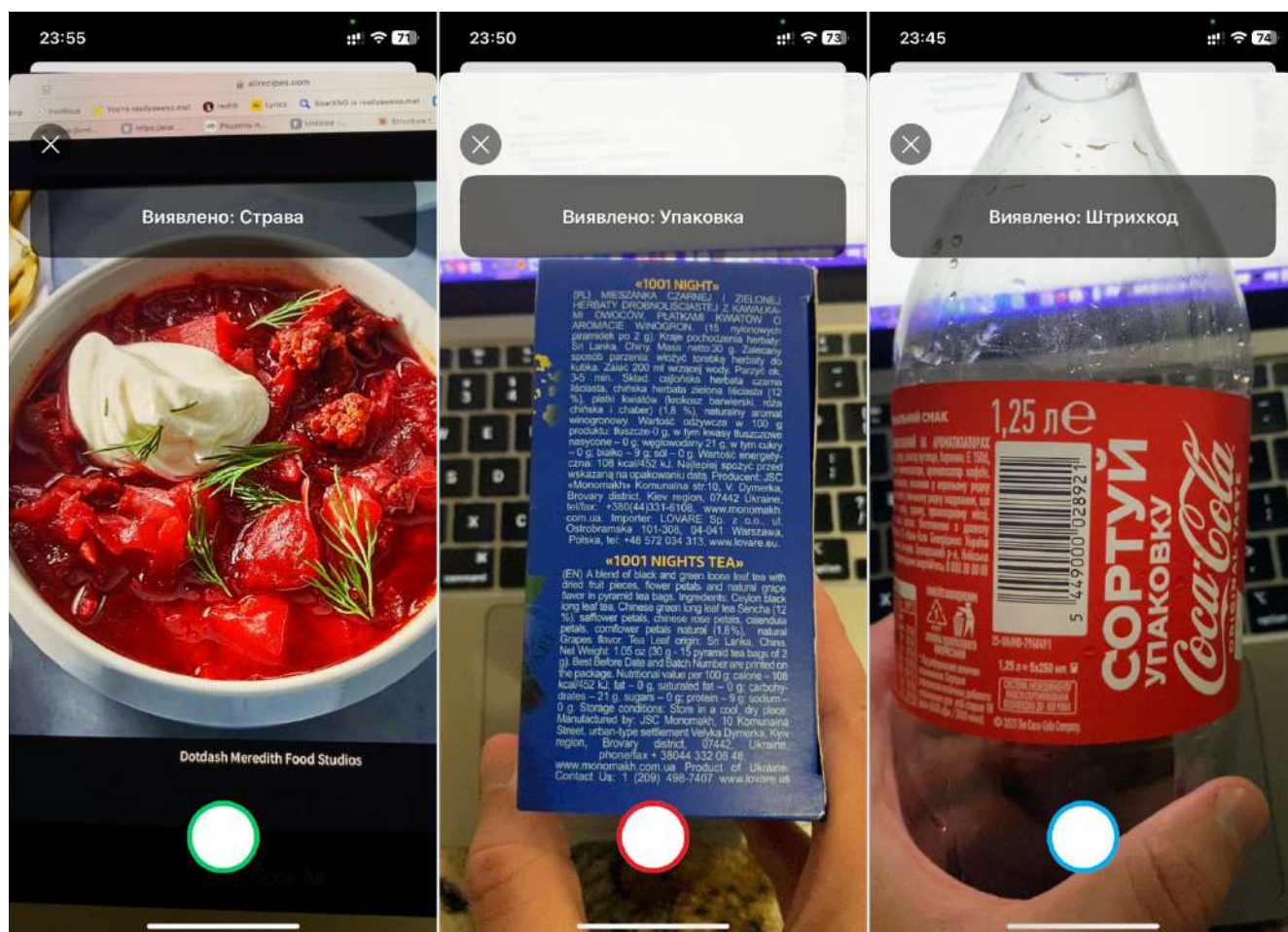


Рисунок 4.5 – Екран активної камери

Кнопки підтвердження мають кольорове кодування відповідно до типу виявленого об'єкта – зелений для страв, червоний для упаковок та блакитний для штрих-кодів. Такий дизайн забезпечує миттєву візуальну комунікацію результату класифікації та інтуїтивний процес сканування.

Коли в кадрі відсутній об'єкт для класифікації, система відображає підказку "Наведіть камеру на об'єкт", спрямовуючи користувача до правильного використання.

4. Екран результатів зображений на рисунку 4.6 відображає результати аналізу з виділенням виявлених алергенів та повним складом продукту.

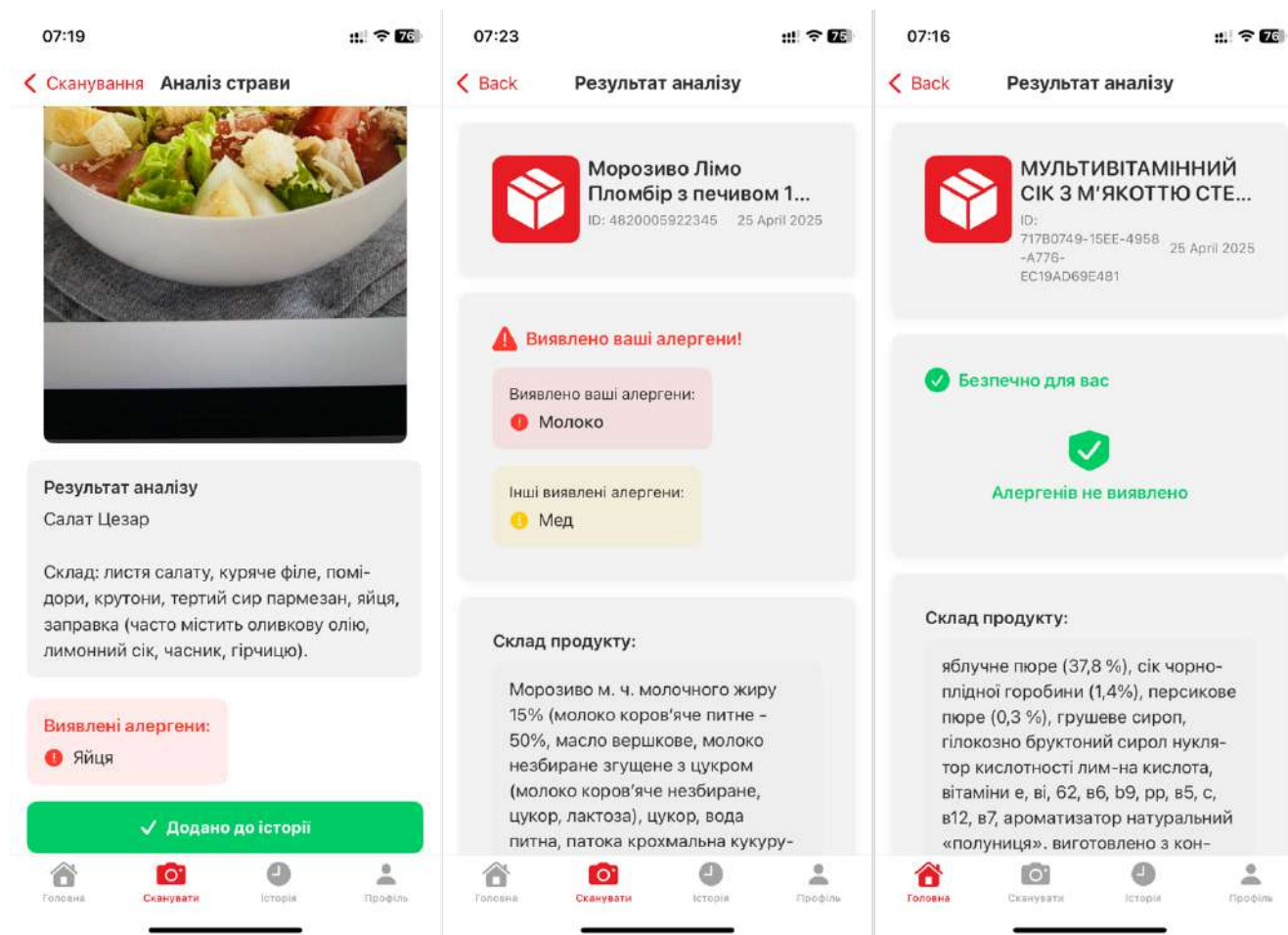


Рисунок 4.6 – Екран результатів трьох різних видів сканувань

На даному рисунку представлено екран результатів аналізу, який показує три різні сценарії відображення залежно від типу сканування. Для страв (зліва) система показує фотографію продукту, назву та детальний склад. При скануванні штрих-коду (центр) або етикетки (справа) фотографія відсутня, натомість відображається ідентифікаційна інформація продукту.

Відображення алергенів теж буде відображатися по різному в залежності чи від того чи був він доданий в алерген-профіль користувача, якщо цей алерген взятий з загальної бази алергенів та не становить суттєвої загрози то він буде виділений міткою як інший алерген. Коли ж алергени відсутні користувач отримає зелений знак, що алергенів не виявлено та продукт цілком безпечний для споживання.

5. Екран профілю зображений на рисунку 4.7 дозволяє користувачеві налаштувати перелік власних алергенів та їх пріоритетність.

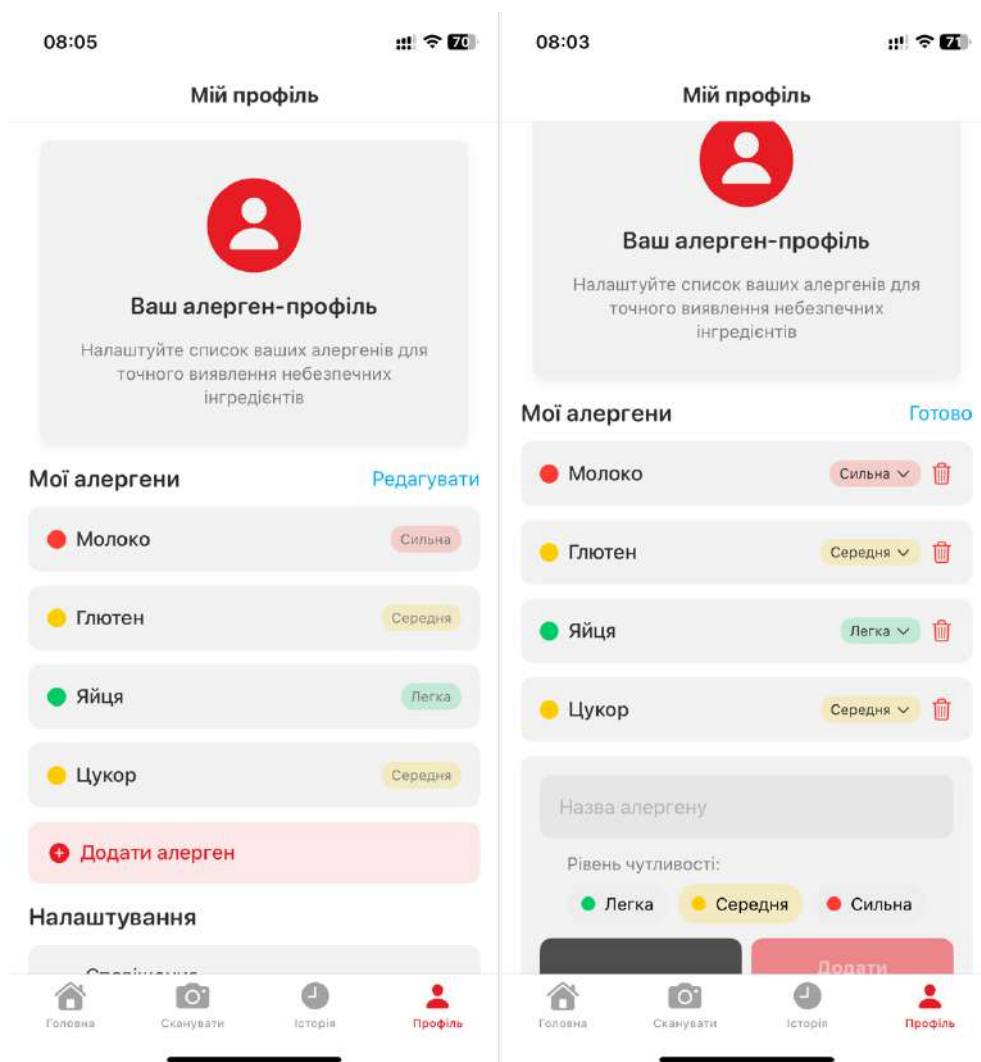


Рисунок 4.7 – Екран алерген-профілю користувача

Екран профілю користувача забезпечує персоналізацію системи виявлення алергенів. Інтерфейс представлено у двох режимах: перегляд та редагування. У

режимі перегляду відображається список алергенів користувача з відповідним рівнем чутливості: червоний маркер для сильної алергії, жовтий для середньої та зелений для легкої. Режим редагування дозволяє модифікувати цей список: додавати нові алергени через форму внизу екрану, видаляти існуючі за допомогою іконки кошика та змінювати рівень чутливості через випадаюче меню. Такий підхід забезпечує точне налаштування системи під індивідуальні потреби користувача, що підвищує ефективність виявлення потенційно небезпечних інгредієнтів при скануванні продуктів.

6. Екран історії сканувань зберігає інформацію про раніше скановані продукти для повторного перегляду. Зображений на рисунку 4.8.

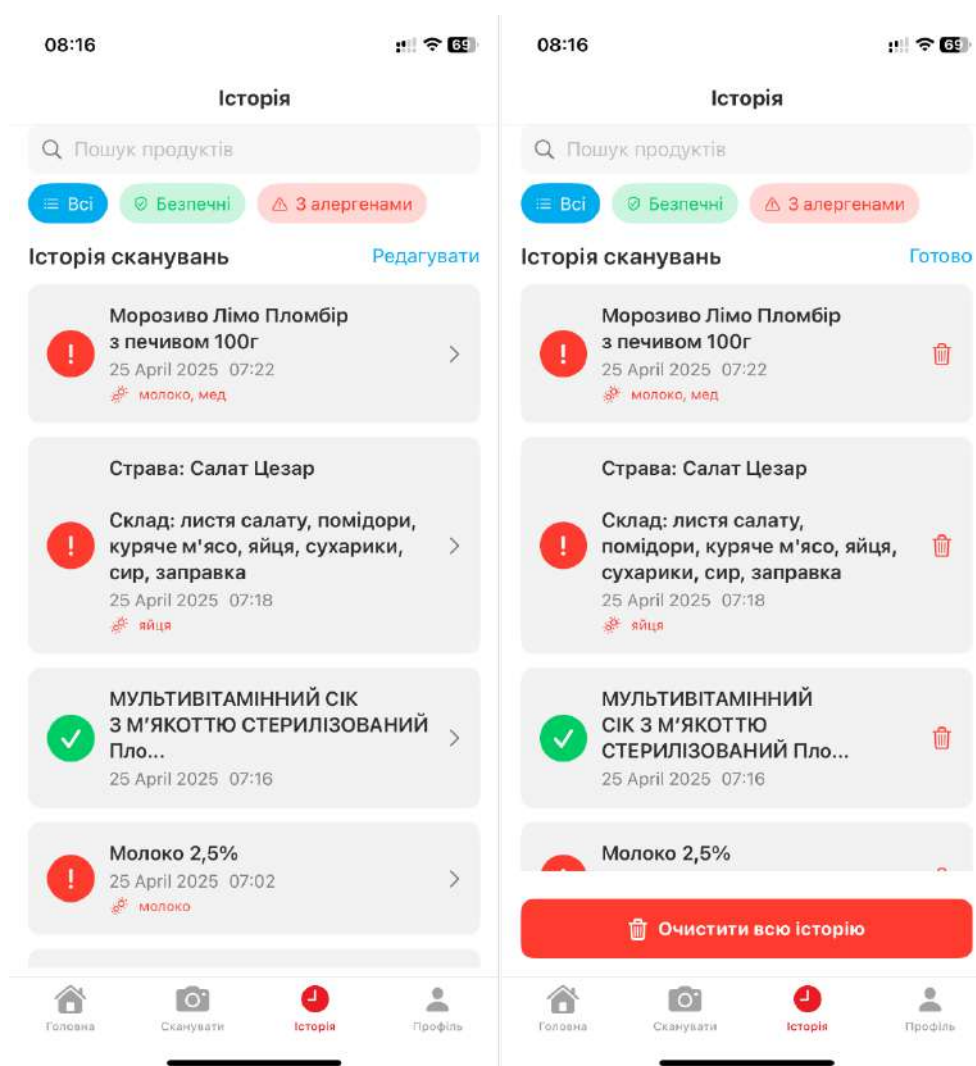


Рисунок 4.8 – Екран історії сканувань

Екран історії відображає хронологічний список раніше сканованих продуктів, представлений у двох режимах: перегляду та редагування. Інтерфейс містить поле пошуку для швидкого знаходження конкретних продуктів та систему фільтрів ("Всі", "Безпечні", "З алергенами"), що дозволяє користувачам легко сортувати результати попередніх сканувань.

Кожен запис в історії містить назву продукту, дату та час сканування, а також візуальний індикатор безпеки: червоний знак оклику для продуктів з виявленими алергенами або зелена галочка для безпечних продуктів. Під назвою продуктів з алергенами відображається список виявлених алергенних компонентів, виділених червоним кольором.

У режимі редагування користувач отримує можливість видаляти окремі записи за допомогою іконки кошика або очистити всю історію одним натисканням кнопки "Очистити всю історію" внизу екрану. Такий функціонал дозволяє користувачам вести організований архів перевірених продуктів, що значно спрощує повторну перевірку часто вживаних продуктів та відстеження безпечності свого раціону протягом часу.

4.5 Тестування програмного забезпечення

Тестування програмного забезпечення кіберфізичної системи виявлення харчових алергенів здійснювалось із застосуванням вбудованих інструментів Xcode, що дозволило всебічно оцінити якість, продуктивність та надійність розробленого застосунку. У центрі уваги було забезпечення ефективного використання системних ресурсів мобільного пристрою та підтвердження стабільності роботи всіх компонентів системи.

Функціональне тестування проводилось для перевірки відповідності розробленого програмного забезпечення вимогам технічного завдання. Як видно з рисунка 4.9, було успішно реалізовано набір автоматизованих UI-тестів, що охоплюють основні сценарії використання системи. Особливу увагу приділено тестуванню процесу запуску застосунку (`testLaunch`, `testLaunchPerformance`).



Рисунок 4.9 – Результати виконання автоматизованих UI-тестів

Аналіз енергоспоживання, проведений за допомогою інструменту Energy Diagnostics на рисунку 4.10, показав низький рівень впливу застосунку на батарею пристрою.

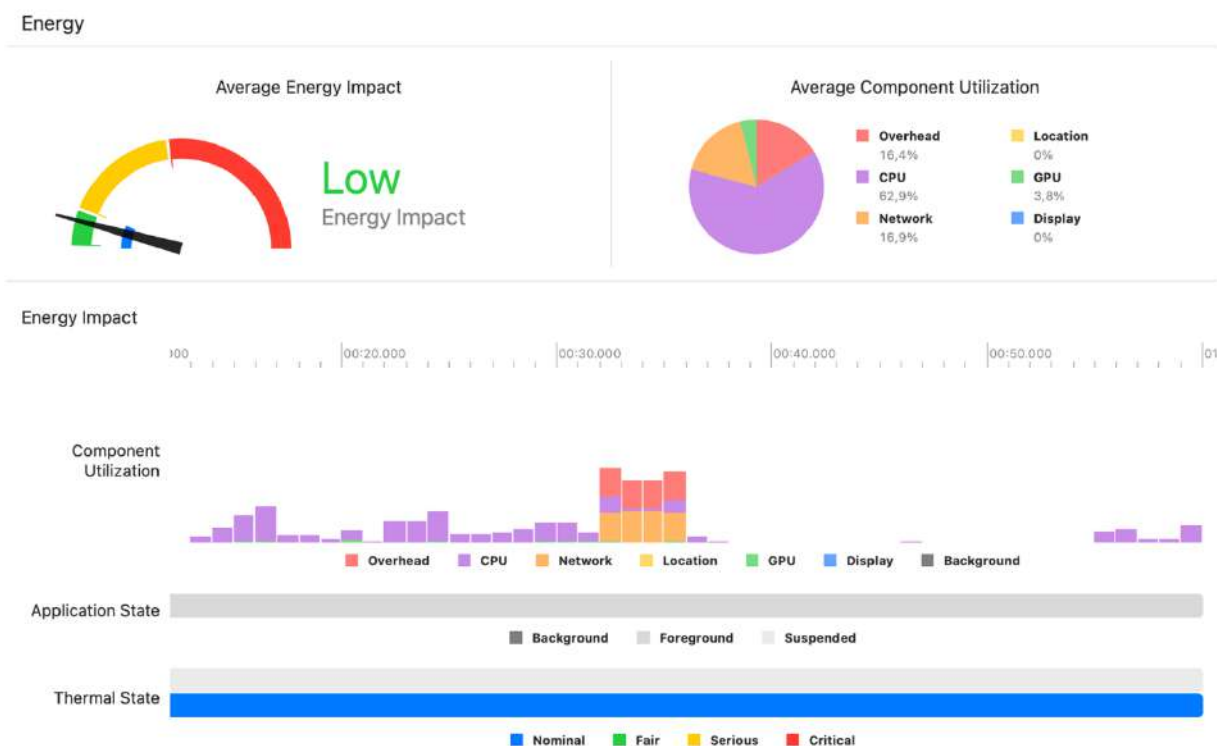


Рисунок 4.10 – Аналіз енергоспоживання застосунку

Це підтверджується як загальною оцінкою "Low Energy Impact", так і детальним розподілом використання компонентів системи, де переважають операції CPU (62,9%) при мінімальному залученні GPU (3,8%). Такі результати свідчать про енергоефективність розробленого програмного забезпечення, що є критично важливим для нашої роботи.

Споживання оперативної пам'яті оцінювалось за допомогою Memory Profiler на рисунку 4.11. Застосунок демонструє дуже ефективне використання оперативної пам'яті з середнім споживанням лише 39,3 МБ, що становить приблизно 1% доступних ресурсів. Пікове значення в 85,3 МБ спостерігається лише при завантаженні застосунку та ініціалізації CoreML-моделей, після чого споживання стабілізується на низькому рівні, що свідчить про відсутність витоків пам'яті та ефективне управління ресурсами.

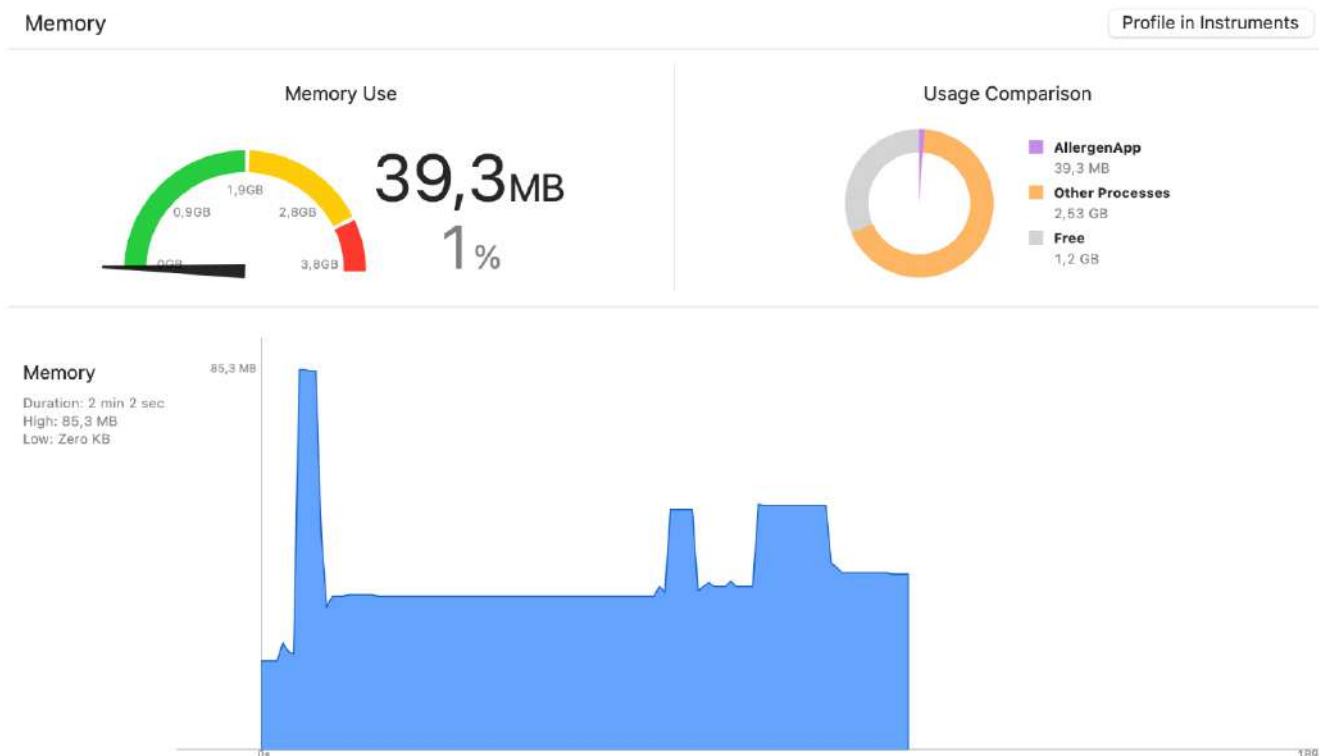


Рисунок 4.11 – Профіль використання пам'яті

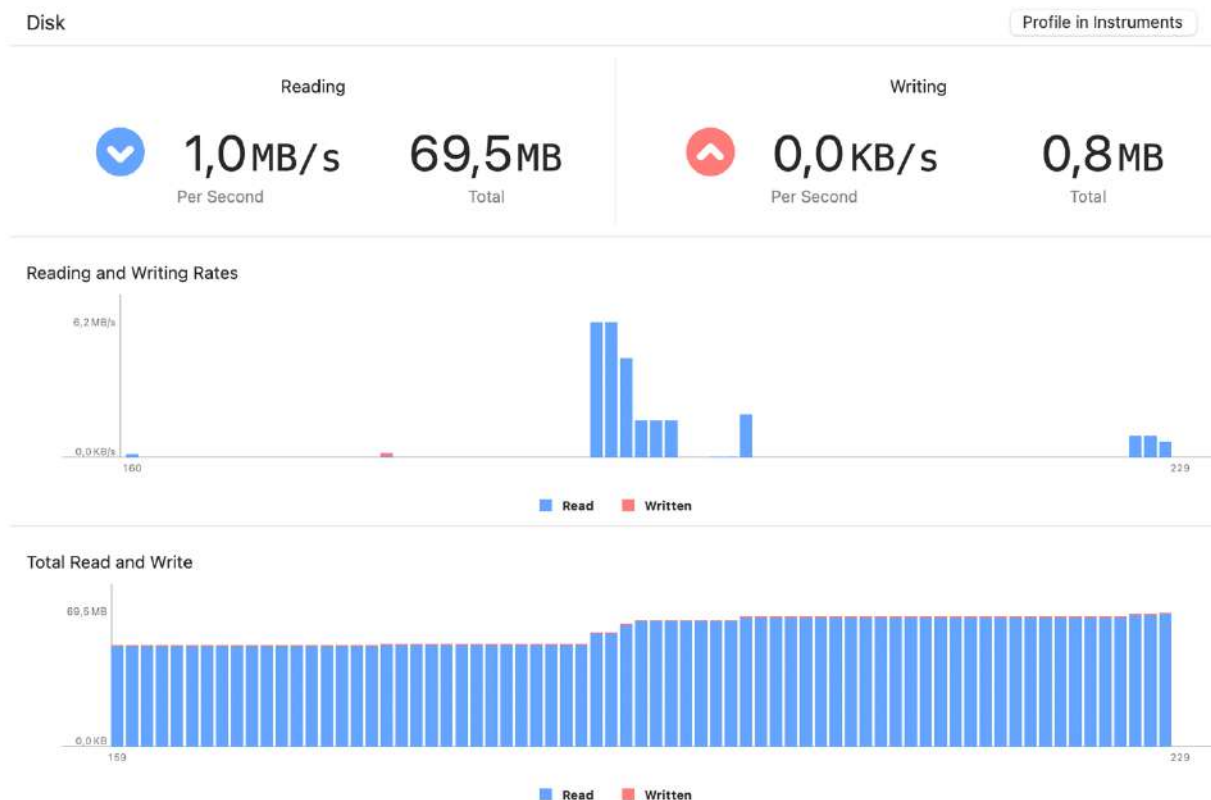


Рисунок 4.12 – Аналіз використання вбудованої пам'яті

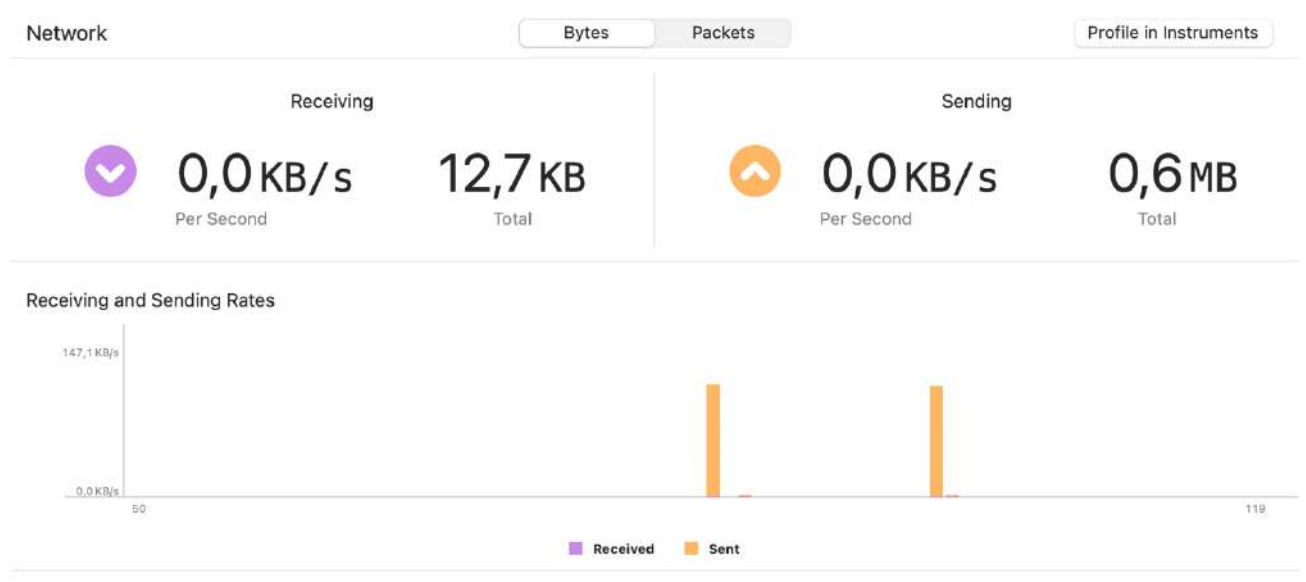


Рисунок 4.13 – Аналіз мережевої активності

Навантаження на вбудовану пам'ять та мережу також були проаналізовані на рисунках 4.12 та 4.13 для забезпечення оптимальної продуктивності. Застосунок

показав ефективну роботу з файловою системою з переважанням операцій читання (69,5 МБ) над записом (0,8 МБ), що відповідає характеру застосунку, орієнтованого переважно на аналіз даних, а не їх модифікацію. Мережева активність демонструє економне використання трафіку з низькою інтенсивністю як приймання, так і відправлення даних, що зменшує залежність від якості інтернет-з'єднання.

За результатами всебічного тестування програмного забезпечення кіберфізичної системи виявлення харчових алергенів можна зробити висновок про його високу якість, надійність та готовність до практичного використання. Низьке енергоспоживання, ефективне використання системних ресурсів та стабільна робота всіх компонентів підтверджують відповідність розробленого застосунку вимогам сучасних мобільних платформ.

4.6 Висновки до четвертого розділу

У четвертому розділі проведено повний цикл розробки програмного забезпечення мобільно-орієнтованої кіберфізичної системи виявлення харчових алергенів.

Спроековано ієрархічну архітектуру системи, що ефективно поєднує локальні обчислення на мобільному пристрої (класифікація, OCR, аналіз тексту) з серверними компонентами (розпізнавання страв за допомогою AdaMine). Впровадження шаблону MVVM для мобільного застосунку та мікросервісного підходу для серверної частини забезпечило високу масштабованість та стійкість до змін.

Розроблені UML-моделі формалізували взаємодію між сім'ю основними класами системи: ImageInput, ImageClassifier, BarcodeProcessor, OCRProcessor, ServerRecognizer, NLPAnalyzer та UserInterface. Виділення окремих координаторів потоків даних дозволило реалізувати три незалежні шляхи обробки для різних типів зображень.

Створено інтуїтивний користувацький інтерфейс з шістьма основними екранами, спроектований з урахуванням психологічних аспектів сприйняття

інформації про харчові ризики. Використання колірної кодування (червоний для алергенів, зелений для безпечних компонентів) та чітких візуальних маркерів забезпечує миттєву оцінку безпечності продукту без додаткового аналізу.

Комплексне тестування підтвердило високу оптимізацію програмної системи. Ресурсні метрики демонструють виключно ефективне використання апаратних ресурсів: енергоспоживання на рівні "Low Impact", використання пам'яті в межах 39,3-85,3 МБ, збалансоване завантаження CPU (62,9%) та GPU (3,8%), а також економне використання мережевого трафіку.

Розроблена система успішно проходить усі функціональні та нефункціональні тести і готова до використання кінцевими користувачами з харчовими алергіями, забезпечуючи високу точність виявлення потенційно небезпечних інгредієнтів у реальних сценаріях застосування.

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень розроблено мобільно-орієнтовану кіберфізичну систему виявлення харчових алергенів на основі машинного навчання та аналізу зображень, що здатна функціонувати в реальних умовах використання та підвищувати безпеку харчування для людей з алергіями.

У першому розділі проведено всебічний аналіз моделей, методів і засобів розпізнавання харчових алергенів, досліджено класифікацію алергенів відповідно до міжнародних стандартів, визначено переваги та обмеження існуючих підходів до їх виявлення. Обґрунтовано необхідність розробки комплексної системи, що поєднуватиме технології комп'ютерного зору, OCR та аналізу природної мови для забезпечення високої точності та надійності виявлення алергенів у різних типах продуктів.

У другому розділі спроектовано тривірневу архітектуру кіберфізичної системи, що включає фізичний рівень збору даних через сенсори мобільного пристрою, аналітичний рівень обробки інформації та рівень інтерфейсу користувача. Запропоновано концепцію гібридної обробки, що оптимально розподіляє обчислювальне навантаження між мобільним пристроєм та хмарними сервісами. Детально проаналізовано основні технологічні виклики, включаючи проблеми класифікації, сегментації, OCR та NLP, розроблено стратегії їх подолання з урахуванням обмежень мобільних платформ.

У третьому розділі запропоновано метод адаптивного виявлення харчових алергенів, що автоматично обирає оптимальний алгоритм аналізу залежно від типу вхідного зображення. Розроблено високоефективну модель класифікації зображень на базі VisionFeaturePrint та GLM, яка досягає точності 98,4% при мінімальних обчислювальних витратах (30-50 мс на iPhone 11). Реалізовано серверний компонент на основі вдосконаленої моделі AdaMine, яка перевершує існуючі аналоги (DeepFood, Recipe2Vec) за всіма ключовими метриками: Top-1 Accuracy – 87,4% (+3,2%), Precision – 81,6% (+5,8%), Recall – 78,3% (+2,1%). Створено

багаторівневий NLP-модуль для аналізу текстової інформації з комплексним словником алергенів, що враховує морфологічні варіації, синоніми та технічні терміни. Впроваджені оптимізації (квантизація, кешування, батчинг) дозволили скоротити час відповіді на 48,4% та підвищити пропускну здатність на 121,4%.

У четвертому розділі представлено програмну реалізацію системи з використанням архітектурного шаблону MVVM для мобільного застосунку та мікросервісного підходу для серверної частини. Розроблено відповідні UML-моделі, що формалізують взаємодію між компонентами системи. Створено інтуїтивний інтерфейс користувача, оптимізований для швидкої візуальної оцінки безпечності продуктів. Комплексне тестування підтвердило ефективність програмного забезпечення з оптимальним використанням ресурсів мобільного пристрою (енергоспоживання на рівні "Low Impact", використання пам'яті в межах 39,3-85,3 МБ).

Набула подальшого розвитку інформаційна технологія виявлення харчових алергенів завдяки впровадженню гібридного підходу з використанням класифікаційної моделі для первинного аналізу зображень. Розроблена кіберфізична система з триканальною архітектурою не лише забезпечує високу точність виявлення алергенів (98,4%) через оптимальний вибір методу аналізу, але й закладає технологічне підґрунтя для створення інклюзивних рішень виявлення алергенів. Запропонований підхід з класифікацією об'єктів у реальному часі (упаковка, штрих-код, страва) стане основою для створення інклюзивної системи допомоги людям з порушеннями зору. У поєднанні з голосовим інтерфейсом така система дозволить користувачам з вадами зору самостійно отримувати інформацію про безпечність продуктів через автоматичне розпізнавання типу продукту та голосові підказки. Це трансформує розроблену кіберфізичну систему з інструменту виявлення алергенів у комплексне рішення для забезпечення харчової безпеки різних категорій користувачів.

Впровадження результатів роботи дозволило створити функціональну систему, що забезпечує надійну ідентифікацію потенційно небезпечних інгредієнтів у різних типах продуктів: упакованих товарах (через OCR та аналіз

складу), товарах зі штрих-кодами (через пошук у базі даних) та готових стравах (через комп'ютерний зір). Система здатна функціонувати на стандартних мобільних пристроях з прийнятним часом відгуку та мінімальними вимогами до підключення.

За темою кваліфікаційної роботи магістра опублікована одна стаття у міжнародному науковому журналі: Talarchuk V., Zaitseva E. Mobile-Oriented Cyber-Physical System for Food Allergen Detection Based on Machine Learning and Image Analysis. *Computer Systems and Information Technologies*. 2025. № 1. P. 29–35 DOI: <https://doi.org/10.31891/csit-2025-1-3>

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Savage J., Johns C.B. Food Allergy: Epidemiology and Natural History. *Immunology and allergy clinics of North America*. 2015. Т. 35, № 1. P. 45–59.
2. Bingchang I.N. Allertify: Android-based Mobile Application for Food Allergens Identification Using Barcode Scanning and Image Recognition. *11th International Scholars Conference*. 2024. P. 1542–1557.
3. Mahesh S., Shrivarshan N.K., Anbuchelvan A.P. OCR and NLP based Personalized Allergen Notifying System. 2025.
4. Openfoodfacts (Open Food Facts). URL: <https://huggingface.co/openfoodfacts> (дата звернення: 04.03.2025).
5. Allegra D., Battiato S., Ortis A., Urso S., Polosa R. A Review on Food Recognition Technology for Health Applications. *Health Psychology Research*. 2020. Т. 8, № 3. P. 9297.
6. Tahir G.A., Loo C.K. A Comprehensive Survey of Image-Based Food Recognition and Volume Estimation Methods for Dietary Assessment. *Healthcare*. 2021. Т. 9, № 12. P. 1676.
7. Dai X. Robust deep-learning based refrigerator food recognition. *Frontiers in Artificial Intelligence*. 2024. Т. 7.
8. Lan X., et al. Foodsam: Any food segmentation. *IEEE Transactions on Multimedia*. 2023.
9. Zhu Z., Dai Y. A New CNN-Based Single-Ingredient Classification Model and Its Application in Food Image Segmentation. *Journal of Imaging*. 2023. Т. 9, № 10. P. 205.
10. Khor P.L., Raymond W.J.K. Food Allergen Detection in Malaysian Food Using Convolutional Neural Networks. *AIJR Proceedings*. 2022. P. 103–108.
11. Additional reasons for OCR errors. *figshare*. *PLOS ONE*. 2015. URL: <https://doi.org/10.1371/journal.pone.0015338.g009> (дата звернення: 04.03.2025).
12. Danz C. The Secret Ingredient. *AMA Journal of Ethics*. 2002. Т. 4, № 9. P. 275–276.

13. Lozano C.P., Canty E.N., Saha S., Broyles S.T., Beyl R.A., Apolzan J.W., Martin C.K. Validity of an Artificial Intelligence-Based Application to Identify Foods and Estimate Energy Intake Among Adults: A Pilot Study. *Current Developments in Nutrition*. 2023. T. 7, № 11. P. 102009.
14. Gorlapraveen123. *ResNet50* : зображення. URL: <https://commons.wikimedia.org/wiki/File:ResNet50.png> (дата звернення: 07.04.2025).
15. World Allergy Organization. Food Allergy: A Global Public Health Problem. *International Archives of Allergy and Immunology*. 2021. Vol. 176, № 4. P. 243-244.
16. European Union. Regulation (EU) No 1169/2011 of the European Parliament and of the Council of 25 October 2011 on the provision of food information to consumers. *Official Journal of the European Union*. 2011. L 304. P. 18-63.
17. Codex Alimentarius Commission. General Standard for the Labelling of Prepackaged Foods. CXS 1-1985, Rev. 2018. Rome : FAO/WHO, 2018. 9 p.
18. Cadene R. Recipe1M Bootstrap PyTorch : GitHub repository. URL: <https://github.com/Cadene/recipe1m.bootstrap.pytorch/blob/master/README.md> (дата звернення: 08.04.2025).
19. Salvador A., Hynes N., Aytar Y., et al. Im2Recipe : Learning Cross-modal Embeddings for Cooking Recipes and Food Images : CSAIL Project. URL: <http://im2recipe.csail.mit.edu> (дата звернення: 08.04.2025).
20. FoodLog : Calorie Counter : мобільний додаток / foo.log Inc. URL: <https://apps.apple.com/us/app/foodlog-calorie-counter/id597390849> (дата звернення: 08.04.2025).
21. Зображення "KnnClassification.svg" // Вікіпедія : вільна енциклопедія. URL: https://uk.wikipedia.org/wiki/Метод_к-найближчих_сусідів#/media/Файл:KnnClassification.svg (дата звернення: 09.04.2025).
22. Sandler M., Howard A., Zhu M., Zhmoginov A., Chen L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. P. 4510-4520.

23. Tan M., Le Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*. 2019. P. 6105-6114.
24. Python-barcode : documentation. URL: <https://python-barcode.readthedocs.io/en/stable/> (дата звернення: 10.04.2025).
25. Ultralytics Documentation. URL: <https://docs.ultralytics.com> (дата звернення: 10.04.2025).
26. Howard A., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. URL: <https://arxiv.org/abs/1704.04861> (дата звернення: 10.04.2025).
27. Nguyen H., et al. Food101: A multi-class food detection and classification system using deep neural networks. *IEEE Access*. 2023. Т. 9. P. 63842-63853.
28. Wang L., Sun X., Zhang X., Zhang W., Zhou Y. Multi-modal data fusion for food recognition: A comprehensive survey. *Information Fusion*. 2022. Т. 81. P. 148-166.
29. Jia W., Li Y., Cao R., Li X., Sun M. A benchmark for automatic visual classification of clinical skin disease images. *European Conference on Computer Vision*. 2022. P. 206-222.
30. Aguilar E., Bolaños M., Radeva P. Food recognition using fusion of classifiers based on CNNs. *International Conference on Image Analysis and Processing*. 2022. P. 213-224.
31. Kawano Y., Yanai K. Food image recognition with deep convolutional features. *ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 2023. P. 589-593.
32. Redmon J., Farhadi A. YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*. 2023.
33. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2023. Т. 39, № 6. P. 1137-1149.

34. He K., Gkioxari G., Dollár P., Girshick R. Mask R-CNN. *IEEE International Conference on Computer Vision*. 2022. P. 2961-2969.
35. Bochkovski A., Wang C.Y., Liao H.Y.M. YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*. 2023.
36. Liu S., Qi L., Qin H., Shi J., Jia J. Path aggregation network for instance segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023. P. 8759-8768.
37. Apple Developer. Core ML Documentation. URL: <https://developer.apple.com/documentation/coreml/> (дата звернення: 12.04.2025).
38. Python Documentation. The Python Standard Library. URL: <https://docs.python.org/3/library/index.html> (дата звернення: 12.04.2025).
39. Microsoft Learn. The MVVM Pattern in .NET MAUI. URL: <https://learn.microsoft.com/en-gb/dotnet/architecture/maui/mvvm> (дата звернення: 13.04.2025).
40. Apple Developer. SwiftUI Documentation. URL: <https://developer.apple.com/documentation/SwiftUI> (дата звернення: 13.04.2025).
41. Yu X., Peng B., Galley M., Gao J., Yu Z. Teaching Language Models to Self-Improve through Interactive Demonstrations. *Proceedings of the 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. 2024. P. 112-124.
42. Shaham U., Elbayad M., Goswami V., Levy O., Bhosale S. Multilingual Interference in Language Models: Cross-lingual Impact Analysis. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*. 2023. P. 2189-2203.
43. Zhang Y., Zhang N., Liu Y., Fabbri A., Liu J., Kanoi R., Lu X., Xiong C., Zhao J., Radev D., McKeown K., Zhang R. Fair Abstractive Summarization of Diverse Perspectives. *Proceedings of the 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. 2024. P. 437-451.
44. Brants T., Franz A. Web 1T 5-gram Version 1. *Linguistic Data Consortium*. 2024. URL: <https://catalog.ldc.upenn.edu/LDC2024T13> (дата звернення: 15.04.2025).

45. Wang L., Chen Y., Xu B., Wang Z. Current Advances in Optical Character Recognition: A 2025 Perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2025. Т. 47, № 2. P. 823-841.

46. Microsoft Research. Azure OCR API Documentation. 2025. URL: <https://learn.microsoft.com/en-us/azure/applied-ai-services/form-recognizer/concept-ocr> (дата звернення: 16.04.2025).

47. Kaggle. Food Image Recognition Datasets. URL: <https://www.kaggle.com/datasets?search=food+recognition> (дата звернення: 16.04.2025).

48. GitHub. Awesome Food Recognition. URL: <https://github.com/topics/food-recognition> (дата звернення: 16.04.2025).

49. Google AI Blog. Using Machine Learning to Detect Allergens. URL: <https://ai.googleblog.com/> (дата звернення: 16.04.2025).

50. Google Cloud Vision API. Text Detection. URL: <https://cloud.google.com/vision/docs/ocr> (дата звернення: 16.04.2025).

51. Tesseract OCR : documentation. URL: <https://tesseract-ocr.github.io/> (дата звернення: 17.04.2025).

52. Flask Documentation. Web development one drop at a time. URL: <https://flask.palletsprojects.com/en/latest/> (дата звернення: 17.04.2025).

53. Flask-SQLAlchemy Documentation. Adds SQLAlchemy support to Flask. URL: <https://flask-sqlalchemy.palletsprojects.com/en/latest/> (дата звернення: 17.04.2025).

54. Stack Overflow Blog. Best practices for REST API design. URL: <https://stackoverflow.blog/2020/03/02/best-practices-for-rest-api-design/> (дата звернення: 17.04.2025).

55. Ambassador Labs. Top 7 REST API Design Practices for Robust APIs. URL: <https://www.getambassador.io/blog/7-rest-api-design-best-practices> (дата звернення: 18.04.2025).

56. Dell Technologies. Dell's Guide to Server Basics. URL: https://i.dell.com/sites/csdocuments/business_smb_sb360_documents/en/us/dell-guide-to-server-basics.pdf (дата звернення: 18.04.2025).

57. HaveTheKnowHow. How to assemble the components for a media server. URL: <https://havetheknowhow.com/assemble-the-hardware/> (дата звернення: 18.04.2025)

58. Servak. Комплектуючі для сервера. URL: <https://servak.com.ua/ua/komplektujuschie-k-serveram/> (дата звернення: 19.04.2025).

59. Ubuntu Server Documentation. URL: <https://documentation.ubuntu.com/server/> (дата звернення: 19.04.2025).

60. Docker. Accelerated AI/ML Development with Docker. *Docker Documentation*. 2025. URL: <https://www.docker.com/products/ai-ml-development/> (дата звернення: 19.04.2025).

61. GitHub. Docker Guide for AI Model Development and Deployment. URL: <https://github.com/saikhu/Docker-Guide-for-AI-Model-Development-and-Deployment> (дата звернення: 19.04.2025).

62. Belagatti P. A Step-by-Step Guide to Containerizing and Deploying Machine Learning Models with Docker. *DEV Community*. 2023. URL: <https://dev.to/pavanbelagatti/a-step-by-step-guide-to-containerizing-and-deploying-machine-learning-models-with-docker-21a1> (дата звернення: 19.04.2025).

63. Apple Developer. VisionKit | Apple Developer Documentation. URL: <https://developer.apple.com/documentation/visionkit> (дата звернення: 20.04.2025).

64. Apple Developer. Scanning data with the camera | Apple Developer Documentation. URL: <https://developer.apple.com/documentation/visionkit/scanning-data-with-the-camera> (дата звернення: 20.04.2025).

65. Apple Developer. AVFoundation | Apple Developer Documentation. URL: <https://developer.apple.com/documentation/avfoundation/> (дата звернення: 20.04.2025).

66. Nginx. Beginner's Guide. URL: http://nginx.org/en/docs/beginners_guide.html (дата звернення: 20.04.2025).

67. F5 NGINX. Installing NGINX Open Source. URL: <https://docs.nginx.com/nginx/admin-guide/installing-nginx/installing-nginx-open-source/> (дата звернення: 20.04.2025).
68. Chen M. et al. Cross-Modal Food Retrieval: Learning a Joint Embedding of Food Images and Recipes with Semantic Consistency and Attention Mechanism. *arXiv preprint*. 2020. URL: <https://ar5iv.labs.arxiv.org/html/2003.03955> (дата звернення: 20.04.2025).
69. Interaction Design Foundation. What is User Interface (UI) Design? *IxDF Library*. 2024. URL: <https://www.interaction-design.org/literature/topics/ui-design> (дата звернення: 20.04.2025).
70. Chappal M.S. The 6 Key Principles of UI Design. *Maze*. 2024. URL: <https://maze.co/collections/ux-ui-design/ui-design-principles/> (дата звернення: 21.04.2025).
71. Impala Intech. The Ultimate UI/UX Design Guide-Principles, Processes, and Best Practices. 2024. URL: <https://impalaintech.com/blog/ui-ux-guide/> (дата звернення: 21.04.2025).
72. Apple Developer. XCTest | Apple Developer Documentation. URL: <https://developer.apple.com/documentation/xctest> (дата звернення: 21.04.2025).
73. BrowserStack. What is iOS Unit Testing? (Tutorial with Xcode & Swift). 2025. URL: <https://www.browserstack.com/guide/ios-unit-testing-tutorial> (дата звернення: 21.04.2025).
74. Apple Developer. Adding tests to your Xcode project. URL: <https://developer.apple.com/documentation/xcode/adding-tests-to-your-xcode-project> (дата звернення: 21.04.2025).
75. Talapchuk V., Zaitseva E. Mobile-Oriented Cyber-Physical System for Food Allergen Detection Based on Machine Learning and Image Analysis. *Computer Systems and Information Technologies*. 2025. № 1. P. 29–35 DOI: <https://doi.org/10.31891/csit-2025-1-3>
76. TestingXperts. Comprehensive Guide to ML Model Testing. 2023. URL: <https://www.testingxperts.com/blog/ml-testing> (дата звернення: 22.04.2025).

77. DataScienceDojo. Machine Learning Models: 4 Useful Production Testing Methods. 2025. URL: <https://datasciencedojo.com/blog/machine-learning-models-testing-methods/> (дата звернення: 22.04.2025).

78. Deepchecks. How to Test Machine Learning Models. 2025. URL: <https://www.deepchecks.com/how-to-test-machine-learning-models/> (дата звернення: 22.04.2025).

79. GeeksforGeeks. Machine Learning Model Evaluation. 2025. URL: <https://www.geeksforgeeks.org/machine-learning-model-evaluation/> (дата звернення: 23.04.2025).

80. Tiwari A. Build Evaluation Framework for Forecast Models. *Towards Data Science*. 2021. URL: <https://towardsdatascience.com/build-evaluation-framework-for-forecast-models-fbc1bd775edd> (дата звернення: 23.04.2025).

ДОДАТОК А (обов'язковий)

СТАТТЯ У ФАХОВОМУ НАУКОВОМУ ЖУРНАЛІ

INTERNATIONAL SCIENTIFIC JOURNAL

ISSN 2710-0766

«COMPUTER SYSTEMS AND INFORMATION TECHNOLOGIES»

<https://doi.org/10.31891/csit-2025-1-3>

UDC 004.9

Valentyn TALAPCHUK
Khmelnitskyi National University
Elena ZAITSEVA
University of Žilina, Žilina, Slovakia

MOBILE-ORIENTED CYBER-PHYSICAL SYSTEM FOR FOOD ALLERGEN DETECTION BASED ON MACHINE LEARNING AND IMAGE ANALYSIS

The prevalence of food allergies necessitates the development of effective methods for the timely detection of allergenic components in food products to prevent dangerous medical reactions. In this work, a mobile-oriented cyber-physical system is proposed, leveraging state-of-the-art machine learning techniques and image analysis for the automated detection of food allergens. The developed system integrates the capabilities of mobile devices equipped with high-quality cameras and efficient computational resources, enabling accurate processing and classification of food product images either locally or via cloud-based inference. This approach ensures flexibility in deployment while maintaining high detection accuracy across diverse environments.

This study examines both the theoretical and practical aspects of applying deep neural networks to object recognition tasks. Particular emphasis is placed on the EfficientDet model, which, due to its optimal balance between detection accuracy and computational cost, represents a promising solution for mobile applications. To enhance recognition performance, image pre-processing methods—including normalization, scaling, and data augmentation—are employed to increase the model's resilience to variations in imaging conditions.

The methodology for data collection and image annotation is described in detail, including the pre-processing procedures that ensure improved model robustness under diverse external conditions. Experimental investigations conducted on a large annotated dataset demonstrate the high accuracy and effectiveness of the system in detecting the presence of food allergens, thereby enabling the prompt identification of potentially hazardous components.

The results of the work highlight the practical applicability of the proposed system in mobile applications for monitoring food quality and preventing allergic reactions. The conclusions outline prospects for further research, focusing on expanding the platform's functional capabilities through the integration of additional sensor technologies and the refinement of data processing algorithms.

Keywords: cyber-physical system, machine learning, image analysis, mobile technologies, food allergens, object detection, CoreML, TensorFlow, data augmentation.

Валентин ТАЛАПЧУК
Хмельницький національний університет
Олена ЗАЙЦЕВА
Університет Жиліни, м. Жиліна, Словаччина

МОБІЛЬНО-ОРІЄНТОВАНА КІБЕРФІЗИЧНА СИСТЕМА ВИЯВЛЕННЯ ХАРЧОВИХ АЛЕРГЕНІВ НА ОСНОВІ МАШИННОГО НАВЧАННЯ ТА АНАЛІЗУ ЗОБРАЖЕНЬ

Поширеність харчових алергій вимагає розробки ефективних методів для своєчасного виявлення алергенних компонентів у продуктах харчування з метою запобігання небезпечним медичним реакціям. У цій роботі пропонується мобільно-орієнтована кіберфізична система, яка використовує сучасні методи машинного навчання та аналізу зображень для автоматизованого виявлення харчових алергенів. Розроблена система інтегрує можливості мобільних пристроїв, оснащених високоякісними камерами та ефективними обчислювальними ресурсами, забезпечуючи точну обробку та класифікацію зображень харчових продуктів як локально, так і через хмарний інтернет. Такий підхід гарантує гнучкість у розгортанні, зберігаючи високу точність виявлення в різних умовах.

У дослідженні розглянуто теоретичні та практичні аспекти застосування глибоких нейронних мереж для задач об'єктного розпізнавання. Особливу увагу приділено моделі EfficientDet, яка завдяки оптимальному співвідношенню між точністю та обчислювальними витратами є перспективним рішенням для мобільних додатків. Для підвищення якості розпізнавання застосовано методи попередньої обробки зображень, включаючи нормалізацію, масштабування та аугментацію даних, що сприяє підвищенню стійкості моделі до змін умов зйомки.

Описано методологію збору та анотації зображень, включаючи процес попередньої обробки (масштабування, нормалізацію, аугментацію), що забезпечує підвищену стійкість моделі до змін зовнішніх умов. Експериментальні дослідження, проведені на великому наборі анотованих даних, свідчать про високу точність та ефективність системи у визначенні присутності харчових алергенів, що дозволяє оперативно ідентифікувати потенційно небезпечні компоненти.

Результати роботи демонструють можливість практичного застосування запропонованої системи у мобільних додатках для моніторингу якості харчових продуктів та попередження алергічних реакцій. У висновках окреслено перспективи подальших досліджень, спрямованих на розширення функціональних можливостей платформи шляхом інтеграції додаткових сенсорних технологій та вдосконалення алгоритмів обробки даних.

Ключові слова: кіберфізична система, машинне навчання, аналіз зображень, мобільні технології, харчові алергени, виявлення об'єктів, CoreML, TensorFlow, аугментація даних.

Introduction

The prevalence of food allergies poses a significant threat to public health and necessitates the development of efficient methods for the timely detection of allergenic components in food products. The primary objective of

МІЖНАРОДНИЙ НАУКОВИЙ ЖУРНАЛ

29

«COMPUTER SYSTEMS AND INFORMATION TECHNOLOGIES», 2025, № 1

this study is to develop a mobile-oriented cyber-physical system that utilizes advanced machine learning techniques and image analysis to automatically detect food allergens in real time. This research addresses the challenges associated with rapid identification and classification of allergenic substances, thereby enabling preventive measures to mitigate potentially dangerous medical reactions.

Traditional laboratory methods for allergen detection, though highly accurate, are often time-consuming and resource-intensive, making them impractical for rapid, on-site screening. In recent years, advances in deep learning, particularly in the field of computer vision, have opened new avenues for the development of automated detection systems. Convolutional Neural Networks (CNNs) and object detection architectures such as YOLO and EfficientDet have demonstrated substantial improvements in both accuracy and speed, making them attractive for real-time applications.

Recent studies have shown that the integration of mobile devices equipped with high-quality cameras, coupled with cloud-based data processing, can significantly enhance the detection of food allergens and other critical parameters [1,2,3]. For instance, Zhang et al. (2020) demonstrated the effective deployment of mobile-based cyber-physical systems for environmental monitoring [1], while Smith and Lee (2019) reported improved detection accuracy for food allergens using deep learning techniques [2]. Furthermore, Brown et al. (2021) highlighted that cloud-based data processing facilitates rapid analysis and real-time feedback, critical for practical applications [3].

Mobile-oriented cyber-physical system for food allergen detection based on machine learning and image analysis

The proposed system consists of three key components: a mobile module, a computational module, and a communication module. The mobile module is responsible for acquiring high-resolution images of food products using built-in cameras on mobile devices. These images undergo pre-processing and allergen detection using a deep learning-based model, such as EfficientDet. The computational module supports both on-device inference for local processing and cloud-based analysis, enabling flexible deployment depending on the available resources and requirements. This modular approach ensures adaptability to different operational scenarios, providing both standalone functionality and cloud integration for improved detection performance.

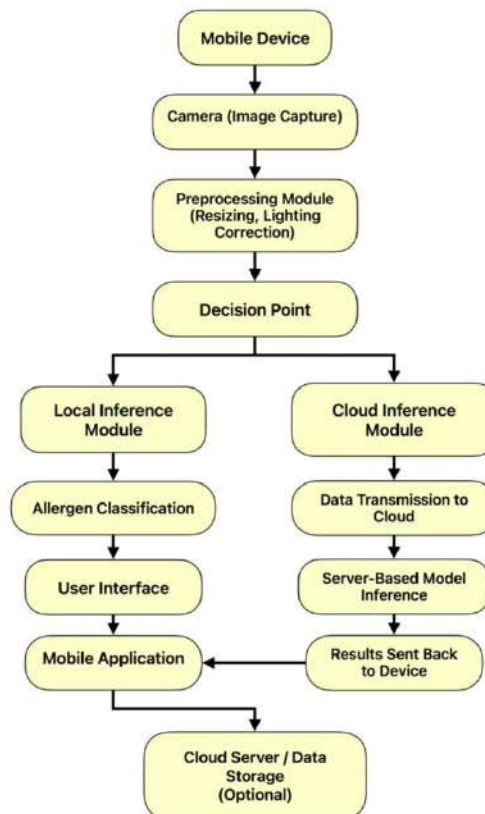


Fig. 1. Flow Diagram of the Proposed Mobile Detection System

In preparing the data, a large dataset of annotated images is utilized. Each image is associated with metadata specifying its dimensions (width W and height H), the object class, and the coordinates of the bounding box ($x_{min}, y_{min}, x_{max}, y_{max}$). Prior to input into the neural network, several pre-processing steps are applied. First, pixel values are normalized to the range $[0,1]$. Then, images are resized to a standard dimension of 640×640 pixels. To increase the diversity of the training set and enhance the model's robustness, data augmentation techniques such as random rotation, translation, brightness adjustment, and contrast enhancement are employed.

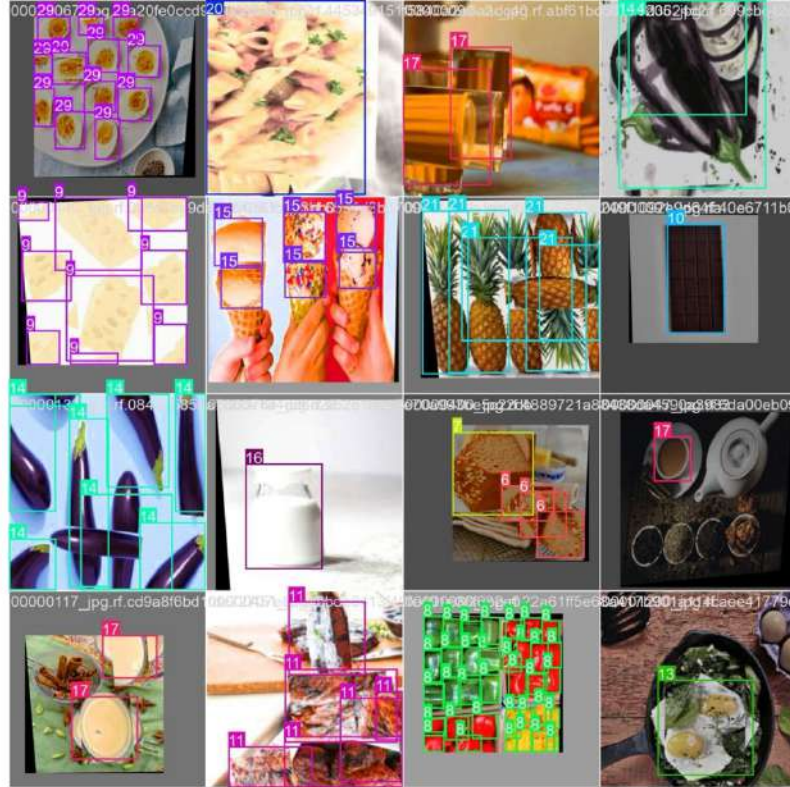


Fig. 2. Examples of Annotated Food Images with Bounding Boxes

Normalization of bounding box coordinates is performed as follows. The center coordinates (x_{center}, y_{center}) and the dimensions (w, h) of the bounding box are computed by:

$$x_{center} = \frac{x_{min} + x_{max}}{2}, \quad w = x_{max} - x_{min}$$

$$y_{center} = \frac{y_{min} + y_{max}}{2}, \quad h = y_{max} - y_{min}$$

Subsequently, these values are normalized by dividing by the image width and height, respectively:

$$x_{center}^{norm} = \frac{x_{center}}{W}, \quad w^{norm} = \frac{w}{W}$$

$$y_{center}^{norm} = \frac{y_{center}}{H}, \quad h^{norm} = \frac{h}{H}$$

Annotations are converted into the TFRecord format to streamline data ingestion in the TensorFlow Object Detection API.

The chosen object detection model is EfficientDet (version D0), selected due to its favorable balance between detection accuracy and computational efficiency—a critical factor for mobile deployments. The model is

fine-tuned on a domain-specific dataset after initial pre-training on larger, publicly available datasets. The training loss function is formulated as a combination of classification and regression losses:

$$L = L_{\text{cls}} + \lambda L_{\text{reg}},$$

where L_{cls} represents the cross-entropy loss for classification, L_{reg} denotes the Smooth L1 Loss for regression of bounding box coordinates, and λ is a scaling factor that balances the two components.

Several studies have confirmed the advantages of EfficientDet compared to other state-of-the-art object detection models. In the seminal work by Tan et al. (2020) [5], EfficientDet demonstrated competitive mean average precision (mAP) while significantly reducing computational costs relative to models such as RetinaNet, Faster R-CNN, and YOLOv3. In addition, Guo et al. (2019) [6] provided a comprehensive survey on deep learning approaches for object detection, noting that EfficientDet offers an optimal trade-off between detection accuracy and computational demands, which is especially beneficial for mobile and resource-constrained applications. Furthermore, Zhou et al. (2019) [7] conducted a benchmark comparison of modern object detection methods and reported that EfficientDet consistently achieves a favorable balance between performance and speed.

Implementation and Experimental Setup

The implementation is based on TensorFlow 2.x and the TensorFlow Object Detection API. The computational module is responsible for several key tasks, including pre-processing of input images, model inference, and visualization of detection results. Image pre-processing involves normalization, resizing, and augmentation, as described above. The trained EfficientDet model is then loaded to generate predictions, including bounding box coordinates and class probabilities. The detected allergens are highlighted on the processed images, providing a visual representation of the results.

To facilitate practical deployment, a prototype mobile application has been developed, featuring an intuitive interface for capturing images, processing them locally, or transmitting them to a server for inference. The application supports data exchange via REST API or WebSocket protocols, allowing integration with cloud-based processing when needed.

The system supports both on-device inference for fast allergen detection and cloud-based processing for enhanced model refinement and large-scale analysis. The on-device inference capability ensures that allergen detection can be performed without reliance on an internet connection, making the system functional in offline environments. Meanwhile, cloud-based processing allows for more computationally intensive operations, such as training updates, large-scale dataset analysis, and collaborative learning, ensuring continuous model improvement over time. This hybrid approach balances efficiency and adaptability, enabling the system to operate effectively across different use cases.

The training process was conducted on Google Colab Pro, utilizing a Tesla T4 GPU. The training dataset consisted of over 12,000 labeled images spanning 30 allergenic food categories, which were partitioned into 80% training data, 10% validation data, and 10% test data. The model was initially pre-trained on the COCO dataset before being fine-tuned on the domain-specific dataset. Training was performed for 50 epochs with a batch size of 16, using the Adam optimizer with weight decay regularization and an initial learning rate of 0.001. The loss function combined classification loss (cross-entropy) and bounding box regression loss (Huber loss) to improve detection accuracy. Over the course of training, the loss function steadily decreased from an initial value of 1.2 to approximately 0.35, indicating successful model convergence.

Evaluation of the detection performance was conducted using mean Average Precision (mAP) and the Intersection over Union (IoU) metric. IoU is calculated as follows:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}},$$

and the mean Average Precision is determined using:

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^N \text{AP}_i,$$

where N represents the total number of classes, and AP_i is the average precision for the i -th class. The final evaluation on the validation set yielded an mAP@0.5 of 78.5%, with Precision of 81.2% and Recall of 75.3%. The Precision-Recall Curve further illustrates the trade-off between precision and recall, demonstrating the model's strong detection capability across different allergen categories.

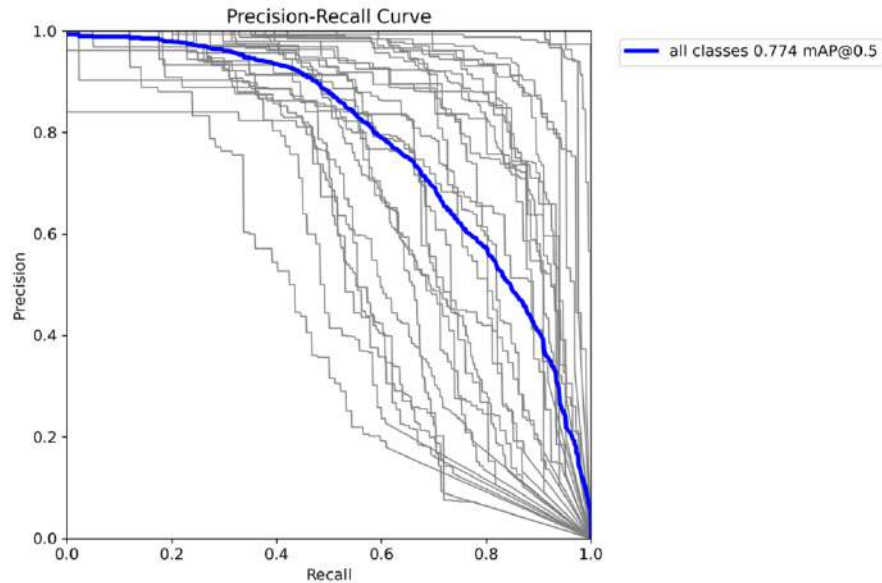


Fig. 3. Precision-Recall Curve of the EfficientDet Model for Food Allergen Detection

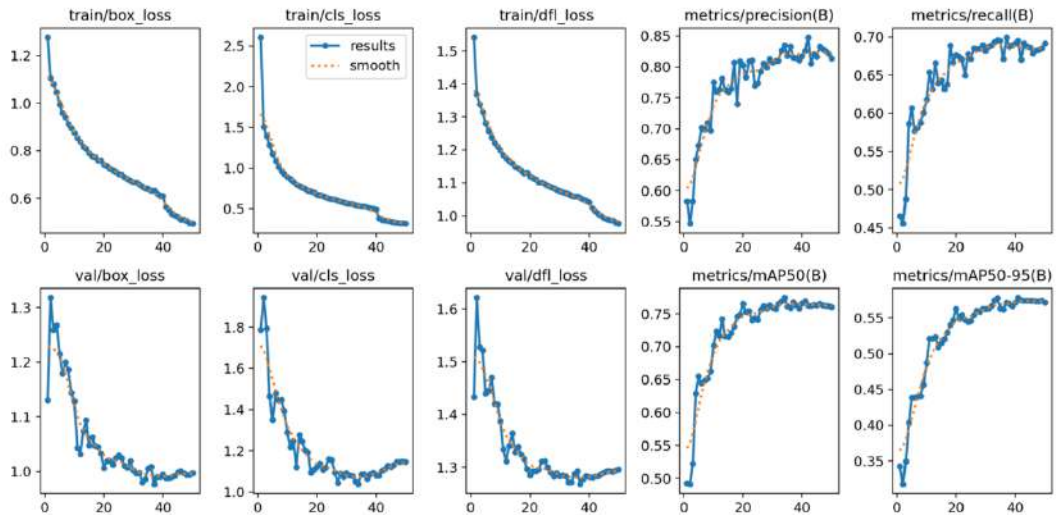


Fig. 4. Training and Validation Loss Curves with Performance Metrics

To enable practical deployment, a prototype mobile application was developed with a user-friendly interface for image capture, processing, and result visualization. The application supports both on-device and cloud-based inference, ensuring adaptability across diverse usage scenarios. In on-device mode, the EfficientDet model, pre-converted into CoreML format, is executed on iOS devices via the Apple Neural Engine, utilizing hardware acceleration for optimized performance. In cloud-based mode, images are transmitted to a remote TensorFlow model through REST API or WebSocket protocols, enabling computationally scalable processing for devices with limited local resources. By integrating both modes, the application maintains operational flexibility, ensuring effective functionality across a wide range of mobile environments.

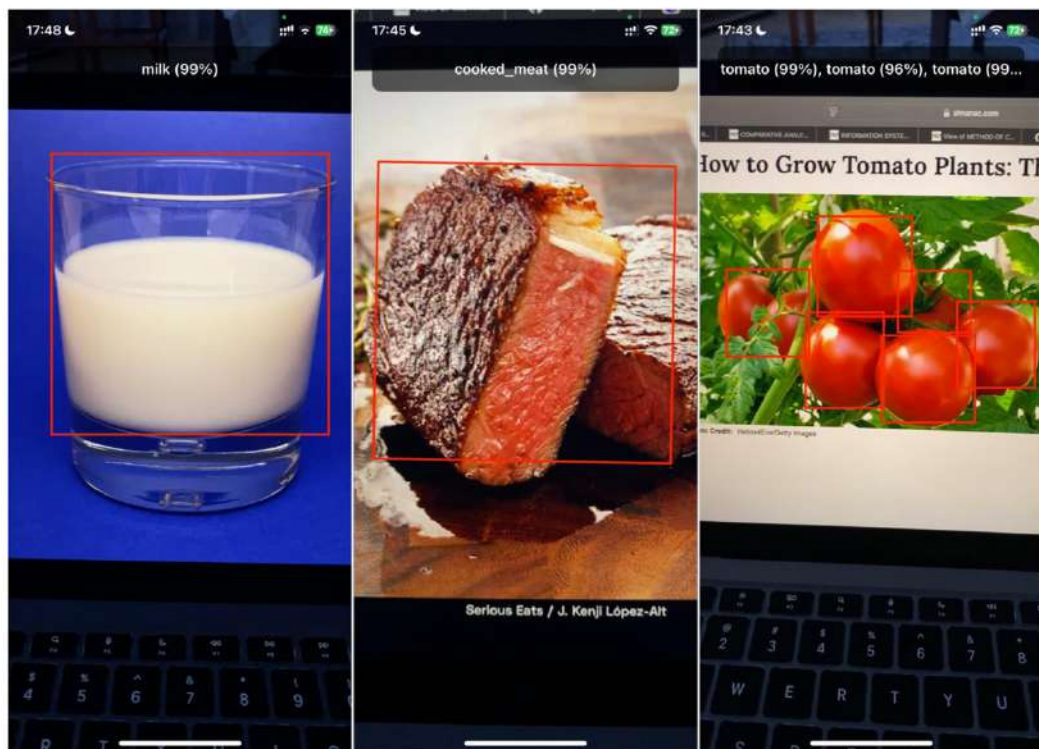


Fig. 5. Example of food allergen detection using the mobile application

Conclusions

The experimental results demonstrate the effectiveness of the proposed mobile-oriented cyber-physical system in achieving accurate and efficient food allergen detection. By leveraging the EfficientDet model alongside advanced pre-processing techniques, the system ensures high detection accuracy while maintaining computational efficiency—key requirements for mobile applications. The consistent reduction in training loss, coupled with stable model convergence, confirms the system's reliability and robustness within the tested dataset.

Comparative analysis with traditional laboratory-based methods and other machine learning approaches reveals that the proposed system significantly improves processing efficiency, enabling timely allergen detection and rapid user notifications. The ability to process images directly on mobile devices and provide fast results has important implications for public health, as it facilitates proactive measures in food allergen identification.

Notwithstanding these promising results, certain limitations persist. The system's performance is inherently dependent on the quality of the input images and the conditions under which they are captured. Low lighting, improper focus, and background clutter can adversely affect detection accuracy. Future research should focus on integrating additional sensor modalities, such as spectroscopic data, to complement image analysis. Furthermore, adaptive learning algorithms that can refine model performance in real time based on environmental feedback are a promising avenue for enhancing system robustness.

In conclusion, the developed cyber-physical system demonstrates significant potential for improving the detection of food allergens through advanced deep learning techniques and mobile technologies. The proposed approach provides an efficient and adaptable solution for food safety monitoring, supporting both on-device and cloud-based processing. Future work will focus on expanding the functional capabilities of the system, optimizing computational processes for resource-constrained devices, and further integrating it with existing monitoring infrastructures to enhance scalability and reliability.

References

1. Zhang, Y., Li, X., & Wang, J. (2020). Mobile-based Cyber-Physical Systems for Environmental Monitoring: A Review. *Journal of Sensor and Actuator Networks*, 9(3), 34–56.
2. Smith, J., & Lee, K. (2019). Deep Learning Approaches for Food Allergen Detection. *IEEE Transactions on Biomedical Engineering*, 66(5), 1234–1245.
3. Brown, A., Chen, L., & Kumar, S. (2021). Cloud-Based Image Processing for Mobile Health Applications. *IEEE Internet of Things Journal*, 8(6), 3507–3515.
4. Khan, M., & Chen, L. (2022). Integration of Mobile Devices and Cloud Computing in Cyber-Physical Systems. *International Journal of Computer Science*, 17(2), 87–98.

5. Tan, M., Pang, R., & Le, Q. V. (2020). EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).
6. Guo, Y., Cheng, M., et al. (2019). A Survey on Deep Learning for Object Detection. IEEE Access, 7, 151560–151583.
7. Zhou, X., Wang, Q., & Krähenbühl, P. (2019). Benchmarking Modern Object Detection Methods. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Valentyn Talapchuk Валентин Талапчук	Master Student of Computer Engineering & Information Systems Department, Khmelnytskyi National University E-mail: vtalapchuk@ukr.net	Магістрант кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет
Elena Zaitseva Олена Зайцева	Ph.D., Associate Professor at the Department of Informatics, University of Žilina (Žilina, Slovakia) https://orcid.org/0000-0002-9087-0311 Member of the Technical Committee of the European Safety and Reliability Association (ESRA) E-mail: elena.zaitseva@fri.uniza.sk	Кандидат технічних наук, доцент кафедри інформатики Університету Жиліни (Жиліна, Словаччина) Член технічного комітету Європейської асоціації з безпеки та надійності (ESRA)

ДОДАТОК Б
(обов'язковий)

UML-ДІАГРАМА АКТИВНОСТЕЙ

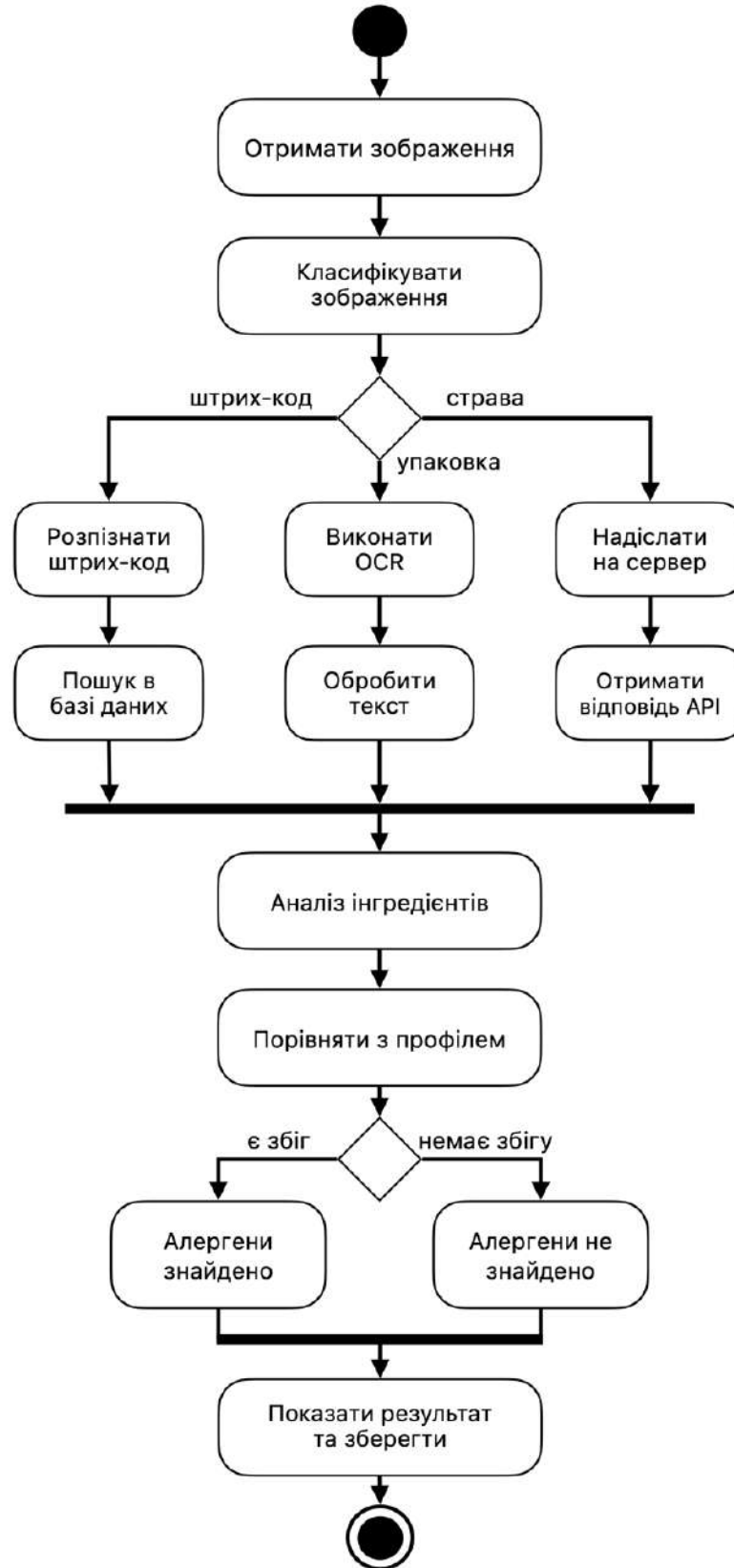


Рисунок Б1 – UML-діаграма активностей

ДОДАТОК В
(обов'язковий)
ПРЕЗЕНТАЦІЯ

**Мобільно-орієнтована кіберфізична
система виявлення харчових алергенів
на основі машинного навчання та
аналізу зображень.**

Кафедра комп'ютерної інженерії та інформаційних систем

Виконав: ст. гр. КІ2М-23-3
Талапчук Валентин

Науковий керівник: д-р. техн. наук, професор
Говорущенко Т.О.



Мета, об'єкт та предмет дослідження

- Метою кваліфікаційної роботи магістра є автоматичне виявлення харчових алергенів на основі методів машинного навчання та комп'ютерного зору.
- Об'єктом дослідження є процес автоматизації розпізнавання їжі та контролю її складу.
- Предметом дослідження є конкретні методи та моделі, які забезпечують автоматичне виявлення алергенних інгредієнтів у харчових продуктах на основі аналізу зображень.

Задачі дослідження

- Провести аналіз сучасних моделей машинного навчання та методів комп'ютерного зору, що використовуються для розпізнавання харчових продуктів та їх інгредієнтів.
- Розробити комплексний алгоритм, що поєднує комп'ютерний зір, OCR та NLP для виявлення алергенів
- Створити гібридну архітектуру з оптимальним розподілом обчислень між пристроєм і сервером
- Реалізувати та оцінити ефективність мобільного додатку з персоналізованими налаштуваннями

Наукова новизна

1. Набув подальшого розвитку метод автоматичного виявлення харчових алергенів, що поєднує класифікацію зображень, оптичне розпізнавання тексту та обробку природної мови в єдиному адаптивному алгоритмі;
2. Набула подальшого розвитку архітектура мобільно-орієнтованої кіберфізичної системи виявлення харчових алергенів, яка базується на гібридній архітектурі з розподіленим обчисленням і персоналізованою логікою аналізу вхідних даних.

Актуальність та аналіз існуючих рішень

- Поширеність харчових алергій сягає 10% населення розвинених країн.
- Алергічні реакції можуть бути небезпечними для життя.
- Щорічно реєструються численні випадки госпіталізацій через випадкове споживання заборонених інгредієнтів.

Недоліки існуючих рішень

Проблема	Опис	Наслідки
Обмеженість продуктами в упаковках	Більшість додатків працюють лише зі штрих-кодами або упакованими товарами	Неможливість аналізу ресторанних страв та домашньої їжі
Відсутність аналізу готових страв	Немає технологій для розпізнавання інгредієнтів у готових стравах	Високий ризик для користувачів при харчуванні поза домом
Низька точність розпізнавання	Існуючі моделі мають обмежену точність класифікації	Пропуск потенційно небезпечних алергенів
Залежність від зовнішніх баз даних	Системи покладаються на повноту і коректність сторонніх баз даних	Неповна інформація про склад продуктів
Слабка персоналізація	Відсутність налаштування під конкретні алергічні профілі	Неврахування індивідуальних особливостей користувачів
Проблеми з OCR	Низька якість розпізнавання тексту на етикетках	Помилки при зчитуванні складу продуктів
Обмежена багатомовність	Більшість рішень орієнтовані на англійську мову	Непридатність для міжнародного використання

Технічні характеристики обладнання

Серверне обладнання

Процесор	Intel Core i5-10400 (6 ядер, 12 потоків)
Оперативна пам'ять	16 GB DDR4 2666 MHz
Накопичувач	SSD NVMe 2 TB
Відеокарта	NVIDIA GeForce GTX 1660 Super з 6 GB VRAM
Мережа	Оптоволоконне підключення 100 Мбіт/с
Операційна система	Ubuntu 20.04 LTS

Мобільний пристрій (тестування):

Модель	iPhone 11	iPhone 13
Процесор	Apple A13 Bionic	Apple A15 Bionic
Оперативна пам'ять	4 GB RAM	4 GB RAM
Камера	12 MP, f/1.8, OIS	12 MP, f/1.6, IBIS
Операційна система	iOS 18	iOS 18

Архітектура кіберфізичної системи

Трирівнева архітектура

Фізичний рівень

- Камера смартфона
- Внутрішня пам'ять
- Мережевий інтерфейс

Аналітичний рівень

- Класифікаційна модель
- OCR-модуль
- Сканер штрих-кодів
- Зовнішня модель розпізнавання страв
- NLP-аналізатор

Рівень інтерфейсу

- Інтуїтивний UI/UX
- Відображення результатів
- Попередження про алергени
- Профіль користувача

Компоненти системи

Класифікаційна модель

- VisionFeaturePrint + GLM
- Точність: 98.4%
- Час інференсу: ~30 мс
- Розмір моделі: ~5 МБ
- Локальне виконання

Серверна модель Adamine (Adaptive Mining network) для складних страв

- Крос-модальне навчання
- Навчання на Recipe1M+
- Точність: 87.4%
- Precision: 81.6%
- Покращення на +5.8%

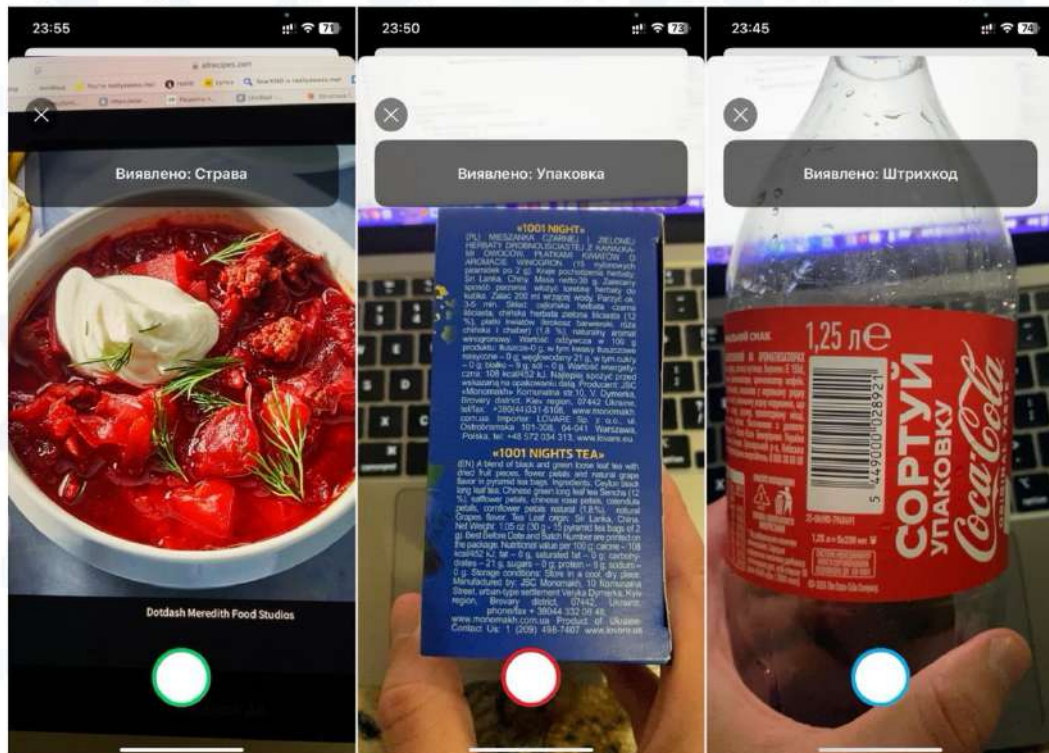
NLP-модуль

- Багаторівневий словник
- Морфологічні варіації
- Е-номери та технічні коди
- Багатомовна підтримка
- Контекстна фільтрація

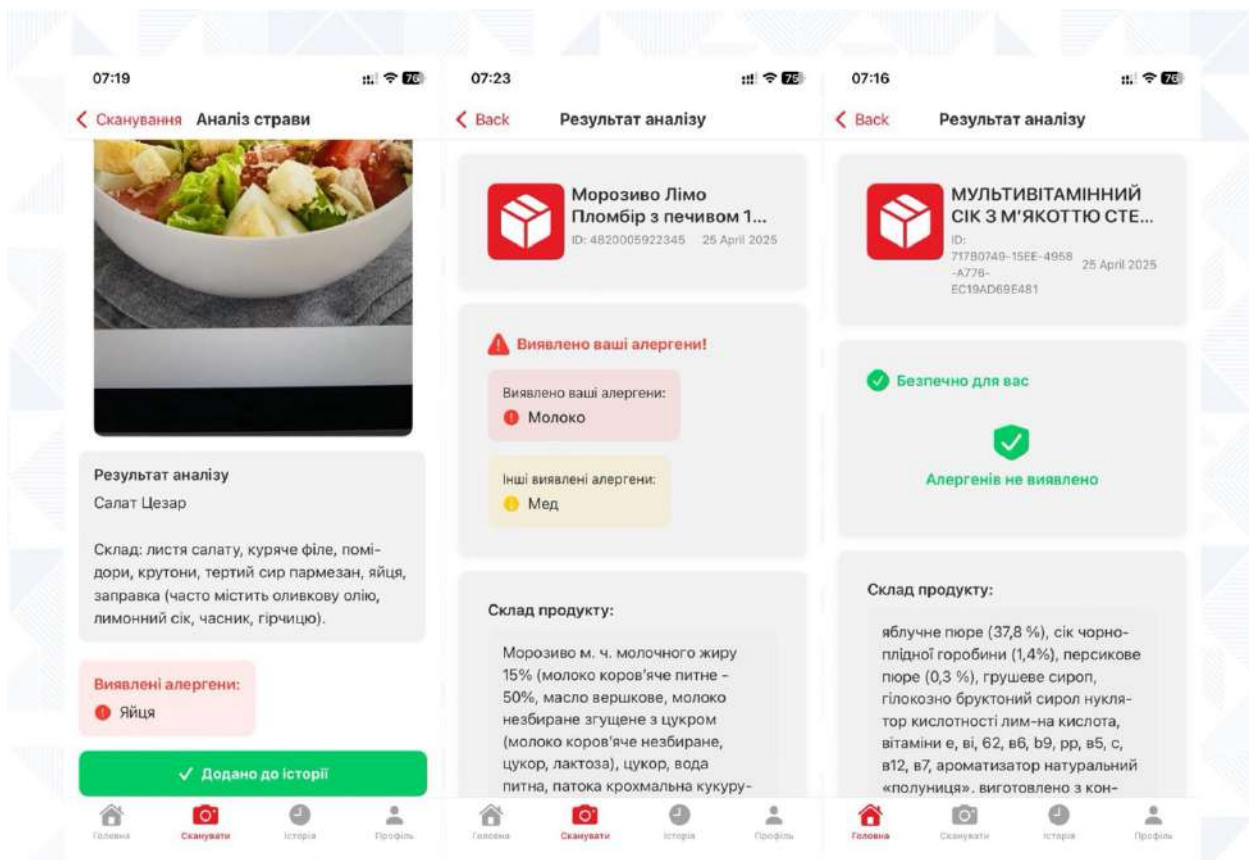
Схема розробленого адаптивного алгоритму виявлення харчових алергенів з класифікацією зображень та різними шляхами аналізу



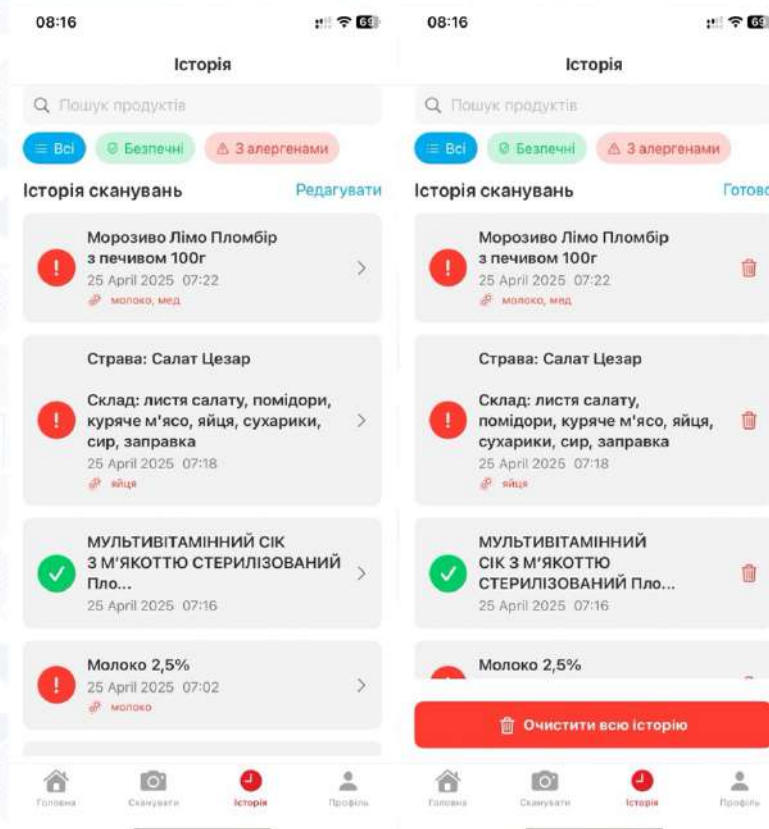
Етапи обробки тексту з упаковки



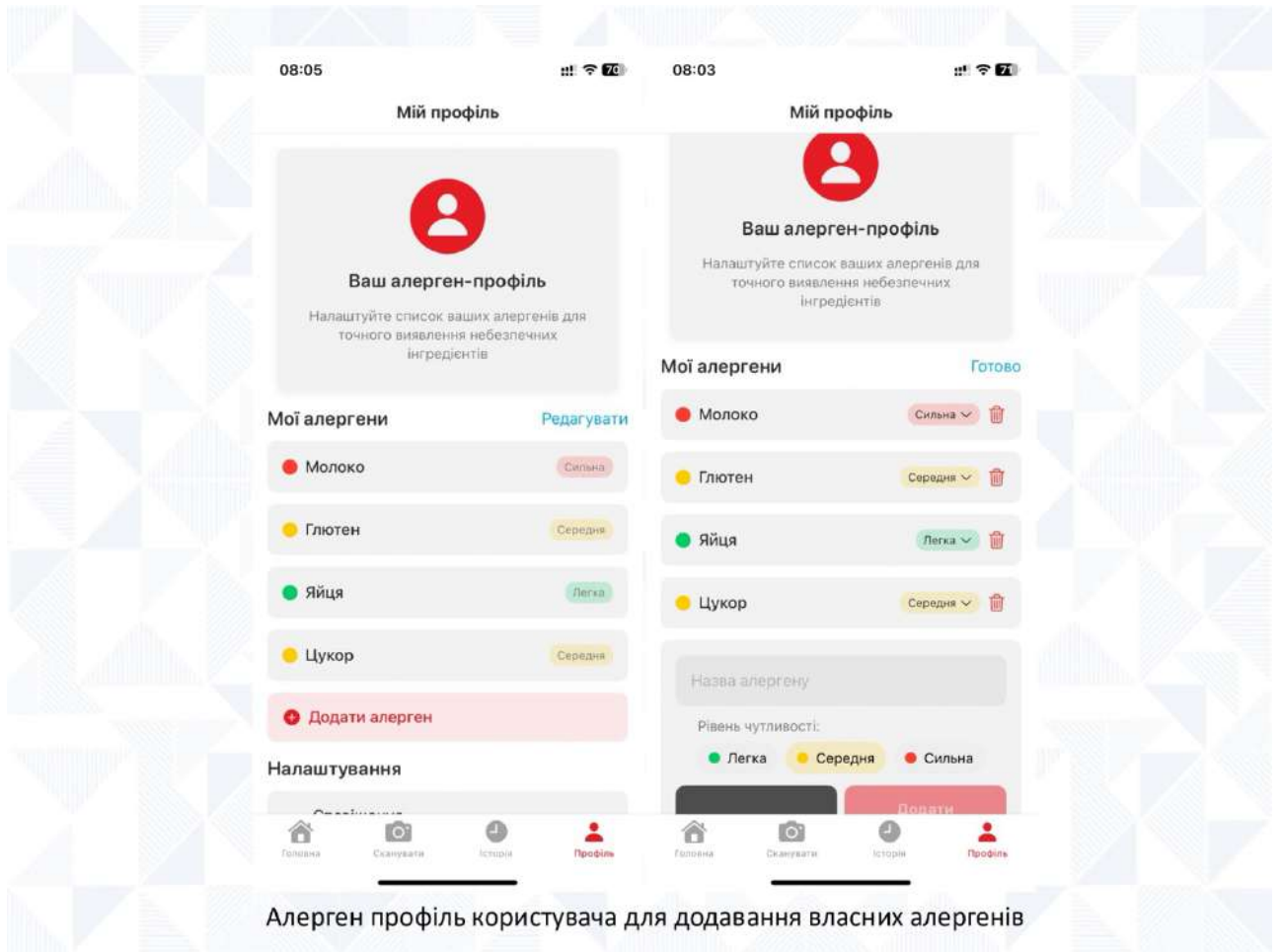
Класифікація продуктів за візуальними ознаками



Результат аналізу продуктів



Історія сканувань з можливістю редагування



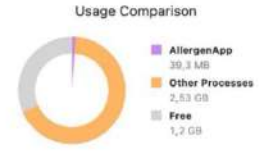
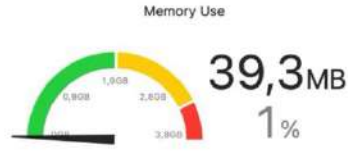
Аналіз енергоспоживання застосунку



Аналіз використання оперативної пам'яті пристрою

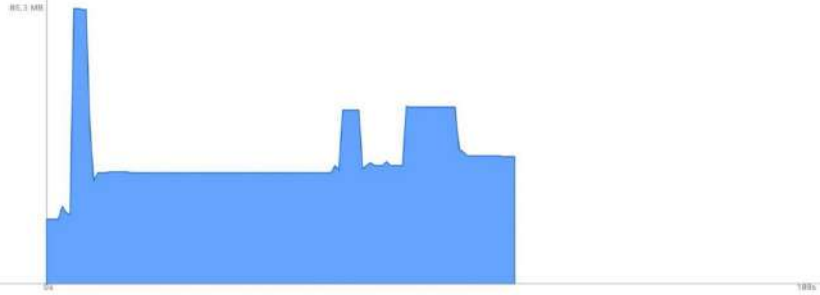
Memory

[Profile in Instruments](#)



Memory

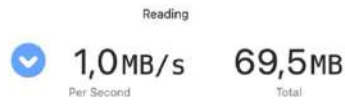
Duration: 2 min 2 sec
High: 85,3 MB
Low: Zero KB



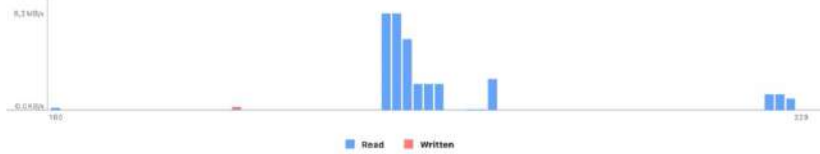
Аналіз використання вбудованої пам'яті пристрою

Disk

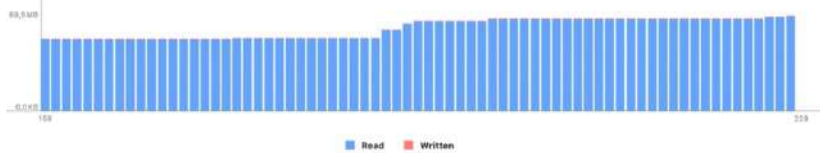
[Profile in Instruments](#)



Reading and Writing Rates



Total Read and Write



Висновки

У результаті виконаних теоретичних і практичних досліджень було розроблено мобільно-орієнтовану кіберфізичну систему виявлення харчових алергенів на основі машинного навчання та аналізу зображень. Система базується на адаптивному підході, що поєднує методи комп'ютерного зору, OCR та NLP для точної ідентифікації потенційно небезпечних інгредієнтів.

Отримані результати демонструють практичну ефективність розробленої системи для підвищення безпеки харчування людей з алергіями. Запропонований підхід має широкі перспективи для подальшого вдосконалення, зокрема через розширення бази даних алергенів, оптимізацію алгоритмів розпізнавання та створення інклюзивних рішень для людей з порушеннями зору.

За темою кваліфікаційної роботи магістра опублікована одна стаття у міжнародному науковому журналі: Talapchuk V., Zaitseva E. Mobile-Oriented Cyber-Physical System for Food Allergen Detection Based on Machine Learning and Image Analysis. *Computer Systems and Information Technologies*. 2025. № 1. P. 29–35 DOI: <https://doi.org/10.31891/csit-2025-1-3>

Anti-Plagiarism v-15.274 Educational

The maximum coincidence with one document 24.0%

Dictionaries check: en_US, ru_RU, ua_UA. Errors in the documents: 11%

ID: 240916 Title: МКР Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень Added in a DB: 2025-05-06 Authors: Валентин ТАЛАПЧУК Heads: Тетяна ГОВОРУЩЕНКО Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	117006	917	30434 (26%)	238 (26%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes
191729	Title: Звіт з ПДП Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображен Added in a DB: 2025-03-18 Authors: Талапчук В.І. Heads: Говорущенко Т.О. Consultants: Opponents:	28193 (24.0%)	203 (22.0%)

Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Валентин ТАЛАПЧУК

Співавтор:

Назва: Талапчук_Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1: 3.9%

Коефіцієнт подібності 2: 0.8%

Мікропробіли: 4

Заміна букв: 1

Інтервали: 0

Білі знаки: 1

Дата створення звіту: 2025-05-06 20:48:52.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедур. Таким чином робота не приймається.

Обґрунтування:

2025-05-06

Дата



Доцент Андрій Нічепорук

експерт

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Талапчук Валентин Іванович

Тема: Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість сторінок записки 81

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є автоматизація виявлення харчових алергенів на основі машинного навчання та аналізу зображень шляхом розроблення відповідної мобільно-орієнтованої кіберфізичної системи.

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У розділі 1 проведений аналіз моделей, методів і засобів розпізнавання харчових алергенів. Розділ 2 кваліфікаційної роботи присвячений моделям та методам виявлення харчових алергенів. У розділі 3 кваліфікаційної роботи розроблено метод виявлення харчових алергенів на основі машинного навчання та аналізу зображень. В розділі 4 кваліфікаційної роботи виконано проектування та розроблення програмного забезпечення кіберфізичної системи виявлення харчових алергенів.

4. Позитивні сторони роботи: отримання наукових та практичних результатів

5. Негативні сторони роботи:

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно з діючими стандартами оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на високому науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: відмінно/А (4.75).

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) Бацмак О.В.,

д.т.н., проф., зав. каф. КН ХНУ

“07” 05 2025 р.

 (підпис)

Завідувачу кафедри КІС
доктору філософії, доценту
Ользі ПАВЛОВІЙ

Талапчука Валентина Івановича

ПІБ здобувача вищої освіти

ФІТ, 2 курсу, групи КІ2м-23-3

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений(а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (StrikePlagiarism та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

22 квітня 2025 року



РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень

Автор: Талапчук Валентин Іванович

Спеціальність: 123 – Комп'ютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Говорущенко Тетяна Олександрівна д-р. техн. наук, професор

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділах є збіг зі звітом з науково-дослідної практики автора Валентина Талапчука "Мобільно-орієнтована кіберфізична система виявлення харчових алергенів на основі машинного навчання та аналізу зображень", який було додано в репозитарій ХНУ 18 березня 2025 року;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з джерелами на один фрагмент речення;
- 4) в якості запозичень в окремих місцях системою зафіксовано послідовності чотирьохрозрядних двійкових кодів, які є вхідними даними до великої кількості задач і не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення;
- 5) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості StrikePlagiarism, складає 3.86% і адресується до 25 періоджерела; та системою Anti-Plagiarism складає 24%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС

Тетяна ГОВОРУЩЕНКО

Олег САВЕНКО

Ольга ПАВЛОВА