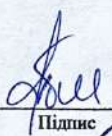


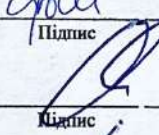
Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних наук


КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему Метод глибокого навчання для реалізації допомоги водіям-початківцям

Галузь знань 12 – Інформаційні технології
Шифр і назва галузі знань
Спеціальність 122 – Комп'ютерні науки
Шифр і назва спеціальності
Освітня програма Комп'ютерні науки
Назва освітньої програми

Виконала: студентка групи КН-21-2  Дар'я МОНАСТІРСЬКА
Група виконавця Підпис Ім'я, ПРІЗВИЩЕ

Керівник: д.т.н., зав. кафедри КН  Олександр БАРМАК
Науковий ступінь, посада Підпис Ім'я, ПРІЗВИЩЕ

Нормоконтроль: к.т.н., доц. каф. КН  Руслан БАГРІЙ
Науковий ступінь, посада Підпис Ім'я, ПРІЗВИЩЕ

До захисту допускаю:
зав. кафедри КН, д.т.н., професор  Олександр БАРМАК
Підпис Ім'я, ПРІЗВИЩЕ

09 06 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра комп'ютерних наук

Освітній ступінь бакалавр

Галузь знань 12 – Інформаційні технології

Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

(підпис)

д.т.н., професор Олександр БАРМАК

«10» 02 2025 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА**

1. Тема кваліфікаційної роботи бакалавра: «Метод глибокого навчання для реалізації допомоги водіям-початківцям»
2. Завдання видано студентці Дар'ї МОНАСТІРСЬКІЙ
(ім'я, прізвище)
3. Керівник роботи д.т.н., зав. кафедри КН Олександр БАРМАК
(посада, ім'я, прізвище)
4. Затверджено наказом університету від «07» 02 2025 р. № 23
5. Дата видачі завдання студенту: «10» 02 2025 р.
6. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – допомога водію-початківцю шляхом розробки методу глибокого навчання інтегрованого у мобільний застосунок для аналізу відеопотоку руху автомобіля для контролю та формування рекомендацій. Метод має реалізувати розпізнавання дорожніх знаків, розмітки, контроль швидкісного режиму та інших параметрів дорожньої ситуації з метою формування рекомендацій для водія. Система, у яку інтегрований розроблений метод, повинна забезпечити функції збору, обробки та аналізу відеоданих у реальному часі, а також надавати зворотний зв'язок водію у зручному форматі.

7. Календарний план виконання кваліфікаційної роботи бакалавра:

№	Назва етапів (розділів) кваліфікаційної роботи бакалавра	Термін виконання	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи бакалавра з керівником, складання календарного графіка виконання	січень 2025	
2	Ознайомлення з предметною областю, формулювання мети і задач дослідження, визначення об'єкта та предмета дослідження	лютий 2025	
3	Проектування та розроблення методу вирішення завдання, загальної архітектури програмного забезпечення, інтерфейсу користувача, вибір засобів реалізації програмного забезпечення	березень 2025	
4	Створення та тестування програмного забезпечення, дослідження ефективності, висновки з виконаної роботи	квітень 2025	
5	Написання пояснювальної записки, урахування зауважень керівника, оформлення згідно з вимогами	травень 2025	
6	Розробка презентаційних матеріалів та попередній захист кваліфікаційної роботи	травень 2025	
7	Отримання відгуку керівника, рецензії, перевірка на плагіат, нормоконтроль	червень 2025	
8	Підготовка до захисту та захист кваліфікаційної роботи	червень 2025	

Виконавець: студентка групи КН-21-2

Група виконавця


Підпис

Дар'я МОНАСТИРСЬКА

Ім'я, ПРІЗВИЩЕ

Керівник: д.т.н., зав. кафедри КН

Науковий ступінь, посада


Підпис

Олександр БАРМАК

Ім'я, ПРІЗВИЩЕ

Анотація

Тема кваліфікаційної роботи бакалавра: «Метод глибокого навчання для реалізації допомоги водіям-початківцям»

Виконавець кваліфікаційної роботи бакалавра: студентка групи КН-21-2 Дар'я МОНАСТИРСЬКА

Керівник кваліфікаційної роботи бакалавра: д.т.н., зав. кафедри КН Олександр БАРМАК

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
55	29	3	48	2

Метою кваліфікаційної роботи бакалавра є допомога водію-початківцю шляхом розробки методу глибокого навчання інтегрованого у мобільний застосунок для аналізу відеопотоку руху автомобіля для контролю та формування рекомендацій. Для реалізації поставлених задач використано неймережу YOLOv8, яка була навчена на відповідному датасеті дорожніх знаків. Мобільний застосунок розроблено з використанням SwiftUI та інтеграцією моделі машинного навчання через Core ML, що дозволяє забезпечити обробку відеопотоку безпосередньо на пристрої користувача. Система надає водієві зворотний зв'язок у зручному візуальному форматі.

Практична цінність роботи полягає у створенні прототипу інформаційної системи, яка може бути використана як основа для комерційного або освітнього застосунку, спрямованого на підвищення рівня підготовки водіїв-початківців і покращення загальної культури водіння.

Ключові слова: дорожні знаки, швидкість, правила дорожнього руху, водій, дорога.

Виконавець: студентка групи КН-21-2

Група виконавця


Підпис

Дар'я МОНАСТИРСЬКА

Ім'я, ПРІЗВИЩЕ

Зміст

Перелік скорочень	2
Вступ.....	3
Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій.....	5
1.1 Аналіз інформаційних моделей.....	5
1.2 Огляд теоретичних підходів до розв’язку подібних задач	7
1.3 Аналіз існуючих програмних засобів	10
1.4 Мета, завдання та вимоги до реалізації інформаційної системи	13
Розділ 2 Метод та засіб допомоги водіям-початківцям з використанням штучного інтелекту	15
2.1 Метод допомоги водіям-початківцям	15
2.1.1 Ідея методу.....	15
2.1.2 Опис моделі глибокого навчання	16
2.1.3 Опис датасету	17
2.1.4. Основні підходи до розробки методу	19
2.1.5. Кроки методу	20
2.2. Особливості реалізації запропонованого методу	24
2.2.1. Вибір підходу	27
2.3. Проектування інформаційної системи допомоги водіям.....	28
2.4 Висновки	35
Розділ 3 Експериментальне дослідження програмної реалізації запропонованого методу	37
3.1 Опис інтерфейсу користувача.....	37
3.2 Результати експериментальних досліджень.....	42
3.3 Висновки до розділу 3	47
Загальні висновки.....	49
Перелік посилань.....	52
Додатки	

Перелік скорочень

Скорочення, термін, позначення	Пояснення
КН	Комп'ютерні науки
DevOps	Developers
IOS	iPhone, iPod, iPad Operating System
VS Code	Visual Studio Code
КН	Комп'ютерні науки
GPS	Global Positioning System
CNN	Convolutional neural network
ХНУ	Хмельницький національний університет
FPR	False Positive Rate
TPR	True Positive Rate
КРБ	Кваліфікаційної роботи бакалавра
RPN	Region Proposal Network
ІС	Інформаційна система

Вступ

Актуальність. З розвитком інформаційних технологій та мобільних пристроїв зростає інтерес до використання інтелектуальних систем, здатних допомагати людям у реальному часі під час виконання повсякденних задач. Однією з актуальних сфер є підвищення безпеки дорожнього руху та підтримка водіїв, зокрема початківців, які ще не мають достатнього досвіду керування автомобілем. Статистика дорожньо-транспортних пригод (ДТП) свідчить, що значна частина аварій трапляється саме через неуважність водія, недостатню реакцію на зміну дорожньої ситуації або неправильне трактування дорожніх знаків і розмітки. Це особливо характерно для водіїв із малим досвідом, які не завжди здатні оперативно приймати правильні рішення у складних або незнайомих умовах.

У цьому контексті виникає потреба у створенні цифрових помічників, які зможуть підтримати водія під час руху, попереджати про можливі порушення або небезпечні ситуації, а також допомагати краще орієнтуватися в потоці дорожньої інформації. Саме тут доцільним є використання сучасних методів глибокого навчання та комп'ютерного зору, які дозволяють з високою точністю розпізнавати об'єкти на зображеннях та відео, включаючи дорожні знаки, розмітку, транспортні засоби, світлофори тощо.

Завдяки стрімкому розвитку обчислювальних можливостей мобільних пристроїв стало можливим реалізовувати системи підтримки водія у вигляді мобільних застосунків. Вони мають перевагу у зручності, доступності та масштабованості, оскільки не потребують спеціального додаткового обладнання. Створення такого застосунку, який аналізує відеопотік з камери смартфона або відеореєстратора, і на основі розпізнаних об'єктів формує рекомендації, відкриває нові горизонти у забезпеченні безпеки дорожнього руху.

Практична цінність такого рішення полягає в можливості без додаткових витрат отримати функціональність, яка частково імітує поведінку сучасних автомобілів з розвиненими системами допомоги водію (ADAS). Це робить

інноваційні технології доступнішими для широкого загалу, включаючи власників недорогих автомобілів та водіїв у країнах, де рівень технічного оснащення транспорту є нижчим. Застосунок може не лише попереджати про дорожні знаки або перевищення швидкості, але й виступати як навчальний інструмент для початківців, формуючи правильну культуру водіння. Об'єкт дослідження – процес аналізу відеопотоку при русі автомобіля (знаки дорожньої розмітки, швидкість руху тощо) для контролю та формування рекомендацій водію-початківцю.

Об'єкт дослідження – процес аналізу відеопотоку при русі автомобіля (знаки дорожньої розмітки, швидкість руху тощо) для контролю та формування рекомендацій водію-початківцю .

Предмет дослідження – методи глибокого навчання для аналізу відеопотоку.

Мета кваліфікаційної роботи бакалавра – допомога водію-початківцю шляхом розробки методу глибокого навчання інтегрованого у мобільний застосунок для аналізу відеопотоку руху автомобіля для контролю та формування рекомендацій.

Завдання кваліфікаційної роботи бакалавра:

- 1) провести аналіз предметної області автоматизованого розпізнавання дорожніх об'єктів, зокрема дорожніх знаків, розмітки та транспортних засобів;
- 2) розробити метод глибокого навчання для реалізації допомоги водіям-початківцям;
- 3) інтегрувати розроблений метод у ІС у вигляді мобільного застосунку, здатного аналізувати відеопотік і формувати рекомендації на основі розпізнаної дорожньої інформації;
- 4) провести тестування розробленого застосунку в умовах, наближених до реального використання, та оцінити його ефективність.

Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій

1.1 Аналіз інформаційних моделей

Сучасні технології комп'ютерного зору та глибокого навчання відкривають нові можливості для автоматизації різних елементів водіння, зокрема для підтримки водіїв-початківців [1]. Недавні дослідження та розробки підтверджують, що такі системи здатні значно знизити ризик дорожньо-транспортних пригод і підвищити ефективність навчання водіїв [2]. Однією з основних можливостей сучасних технологій є здатність аналізувати відеопотік у реальному часі для своєчасного виявлення важливих об'єктів дорожньої ситуації та надання корисних рекомендацій водієві.

У рамках цієї роботи, метою є допомога водію-початківцю шляхом розробки методу глибокого навчання інтегрованого у мобільний застосунок для аналізу відеопотоку руху автомобіля для контролю та формування рекомендацій, що аналізуватиме відеопотік, зібраний камерою мобільного пристрою, для розпізнавання дорожніх знаків, розмітки, пішоходів та інших об'єктів. На основі отриманих даних метод формуватиме рекомендації щодо безпечного водіння, зокрема щодо швидкості руху, дотримання дистанції, поворотів тощо [3]. Оскільки застосунок орієнтований на водіїв-початківців, важливими є зручність використання та доступність в режимі реального часу.

Інформаційна модель описує головні елементи, що взаємодіють у системі, і процеси, що відбуваються між ними [4]. Для мобільного застосунку-помічника водія можна виділити такі основні компоненти:

- *відеопотік* – вхідні дані, отримані за допомогою камери мобільного пристрою. Це послідовність зображень, які обробляються для виявлення об'єктів [5];

- *об'єкти дорожньої ситуації* – це різноманітні елементи, які мають значення для безпеки руху [6], зокрема: дорожні знаки (наприклад, знаки обмеження швидкості); дорожня розмітка (лінії, смуги, пішохідні переходи); пішоходи та інші транспортні засоби; перешкоди та об'єкти на дорозі;

- *алгоритми аналізу відеопотоку* – використання методів комп'ютерного зору та глибокого навчання для детекції та класифікації об'єктів на кадрах відео [7];

– *контекст руху* – інформація, що додається для кращого розуміння ситуації на дорозі, включаючи швидкість автомобіля, GPS-координати та напрямок руху [8];

– *рекомендації водієві* – після аналізу ситуації система генерує рекомендації для водія; це можуть бути попередження щодо порушення швидкісного режиму, рекомендації по безпечному маневруванню, дотриманню дистанції тощо [9].

Для розробки ефективного методу важливими вимогами є:

– *продуктивність* – для роботи в реальному часі потрібно використовувати оптимізовані моделі, що не потребують великих обчислювальних ресурсів;

– *точність* – високий рівень точності розпізнавання об'єктів забезпечить надійність рекомендацій для водія, зокрема для безпеки руху;

– *інтерфейс та зручність використання* – рекомендації повинні подаватися у зручній формі: голосові підказки, графічні елементи на екрані смартфона;

– *швидкість роботи на мобільних пристроях* – важливо забезпечити достатню швидкість обробки зображень, щоб система могла аналізувати відео в реальному часі без затримок.

Отже, було проведено аналіз предметної області та визначено основні компоненти інформаційної моделі мобільного застосунку-помічника водія-початківця. Розглянуто основні об'єкти дорожньої ситуації, які необхідно виявляти у відеопотоці, а також алгоритми глибокого навчання, що можуть бути використані для їхньої детекції та класифікації. Особливу увагу приділено контексту руху транспортного засобу та формуванню корисних рекомендацій для водія в режимі реального часу. Також було окреслено основні вимоги до ефективного функціонування методу [10], серед яких продуктивність, точність, зручність інтерфейсу та швидкодія. Визначені положення створюють підґрунтя для подальшої розробки методу глибокого навчання та його інтеграції у мобільний застосунок з метою підвищення безпеки дорожнього руху для водіїв-початківців.

1.2 Огляд теоретичних підходів до розв'язку подібних задач

Аналіз відеопотоку в реальному часі для підтримки водія є однією з найбільш актуальних і складних задач у галузі комп'ютерного зору та глибокого навчання. Технології, що дозволяють здійснювати автоматичне розпізнавання об'єктів, оцінку дорожньої ситуації та формулювання рекомендацій, швидко набирають популярності, особливо в контексті створення інтелектуальних систем допомоги водіям [11].

Задача аналізу відеопотоку охоплює широкий спектр методів, серед яких виділяються підходи до детекції об'єктів, трекінгу та класифікації, а також до оцінки ситуації на основі динамічних даних, отриманих із відеокамер та сенсорів. Теоретичні основи цих методів базуються на нейронних мережах, зокрема на згорткових нейронних мережах (CNN), моделях для обробки послідовностей даних (LSTM, GRU), а також на сучасних архітектурах для мобільних пристроїв, таких як MobileNet та EfficientNet [12].

Для ефективного аналізу відеопотоку, що йде з камери, використовуються різні моделі глибокого навчання, кожна з яких має свої особливості та переваги в конкретних сценаріях. Основні моделі, які можуть бути застосовані:

- Convolutional Neural Networks (CNN) – основні моделі для задач комп'ютерного зору, здатні автоматично виділяти важливі ознаки зображення; ці мережі складаються з кількох шарів, включаючи шари згортки, які дозволяють виділяти просторові ознаки на зображеннях; вони використовуються для детекції дорожніх знаків, розмітки та інших об'єктів [13];

- YOLO (You Only Look Once) – одна з найбільш ефективних моделей для детекції об'єктів у реальному часі; дозволяє швидко та точно ідентифікувати об'єкти на відео; особливо популярна завдяки своїй швидкості та здатності працювати в реальному часі на мобільних пристроях, що робить її ідеальним кандидатом для мобільного застосування помічника водія [14];

- MobileNet – полегшена архітектура CNN, оптимізована для роботи на мобільних пристроях з обмеженими обчислювальними ресурсами; модель

підтримує обробку зображень у реальному часі, що є важливим для застосунків, що працюють на смартфонах, зокрема для аналізу відео потоку [15];

– LSTM та GRU – Long Short-Term Memory (LSTM) і Gated Recurrent Units (GRU) використовуються для аналізу послідовних даних; у даній роботі ці моделі можуть бути корисні для відстеження руху об'єктів (наприклад, інших автомобілів), а також для оцінки зміни ситуації на дорозі в часі, що дозволяє зробити рекомендації на основі тимчасових залежностей [16];

– EfficientDet – модель для детекції об'єктів, яка оптимізована для мобільних пристроїв; поєднує високу точність із ефективністю, що дозволяє зберігати ресурси при високій продуктивності [17].

Для побудови ефективного методу розпізнавання в мобільному застосунку важливо не лише вибрати відповідну модель глибокого навчання, але й порівняти їх точність і продуктивність у конкретних завданнях. Зокрема, у статті [18] автори порівняли кілька класифікаторів для розпізнавання обличчя на фрагментах зображень. Метою було визначити, який класифікатор забезпечує найкращу точність. Досліджували логістичну регресію, метод опорних векторів (SVM) та згорткові нейронні мережі (CNN). Результати показали, що CNN у поєднанні з FaceNet забезпечує найкращу точність, навіть при низькій роздільній здатності чи різних кутах зйомки. Це підтверджує ефективність таких методів для систем безпеки та біометрії.

У статті [19] у дослідженні брали участь 99 водіїв, які протягом трьох місяців використовували інтегровані та автономні навігаційні системи. Водії надавали перевагу використанню систем у ситуаціях з низьким навантаженням (зупинка, рух на низькій швидкості). Під час руху спостерігалася адаптація швидкості та дистанції слідування. Не було виявлено збільшення критичних ситуацій, таких як близька відстань до інших машин. Водії адаптували свою поведінку до вимог дорожнього руху, що підтверджує відсутність загрози безпеці.

У статті [20] досліджується застосування гібридної нейронної мережі для розпізнавання дорожніх знаків. Для зменшення розмірності використовують двосторонню асоціативну пам'ять (BAM), а для класифікації застосовують

багатошаровий перцептрон. В результаті система досягла 98% точності при розпізнаванні знаків з роздільною здатністю 64x64 пікселів, що підтверджує високу ефективність методу для використання в реальних умовах, зокрема в системах допомоги водіям.

У статті [21] досліджується психофізіологічне сприйняття водіями інформації на дорозі, зокрема використання автоматичних засобів розпізнавання дорожніх знаків (TSR). Проведено опитування водіїв різних вікових груп на ділянці дороги, де була змінена організація дорожнього руху. Виявлено, що молодші водії (до 25 років) частіше використовують TSR, порівняно з водіями середнього та старшого віку. Досліджено, що водії мають кращу здатність сприймати нові дорожні знаки на відстані до 150 м, після чого вони зменшують увагу до змін. Запропоновано впровадження автоматичних систем для підвищення безпеки дорожнього руху, оскільки вони не підпадають під вплив людського фактору.

У статті [22] автори досліджують методи автоматизації розпізнавання дорожніх знаків за допомогою нейронних мереж. Основною метою роботи є розробка ефективного алгоритму для виявлення та класифікації дорожніх знаків на відеокадрах.

Автори пропонують методику, що включає етапи пошуку знаків, визначення їх контурів і подальшу класифікацію за допомогою згорткової нейронної мережі (CNN). Завдяки цьому підходу вдалося досягти високої точності в розпізнаванні знаків на різних зображеннях, що є важливим для систем безпеки в автомобілях. Результати показали, що використання згорткових нейронних мереж дозволяє ефективно класифікувати дорожні знаки навіть при змінних умовах зображення, таких як освітлення чи перекриття частини знака іншими об'єктами.

Цей підхід має значний потенціал для застосування в реальних системах допомоги водіям і в автомобільних системах безпеки.

У статті [23] досліджується застосування глибоких згорткових нейронних мереж (CNN) для розпізнавання та класифікації дорожніх знаків в реальному

часі. Автори розробили нейронну мережу, здатну ефективно працювати в умовах змінного освітлення і погоди.

Мережа показала високу точність розпізнавання знаків, навіть при різних погодних умовах та освітленні. Це дозволяє інтегрувати її в автомобільні системи безпеки для реального використання.

Отже, усі статті, що були розглянуті, підкреслюють важливість використання сучасних методів і технологій, зокрема нейронних мереж, для розпізнавання дорожніх знаків та автоматизації систем допомоги водіям. Головними аспектами досліджень є ефективність згорткових нейронних мереж (CNN) у різних умовах освітлення та погодних умовах, що значно підвищує точність та надійність систем в реальному часі. Також важливим фактором є використання кольорової сегментації для обробки зображень, що дозволяє адаптувати систему до різних умов середовища.

Більшість досліджень підтверджують, що автоматичне розпізнавання дорожніх знаків, інтегроване в системи допомоги водіям (ADAS), здатне значно підвищити безпеку дорожнього руху, мінімізуючи вплив людського фактора, покращуючи здатність водіїв реагувати на змінні умови дорожнього руху [24]. Розвиток таких технологій має значний потенціал для впровадження в реальних системах безпеки та автомобільних технологіях, зокрема для водіїв з обмеженими можливостями.

Загалом, дослідження показують, що успішне застосування нейронних мереж і передових методів обробки зображень здатне забезпечити високий рівень точності в розпізнаванні дорожніх знаків і є важливим елементом у розвитку інтелектуальних систем для сучасного автомобільного транспорту.

1.3 Аналіз існуючих програмних засобів

У сфері підтримки водіїв під час керування автомобілем розроблено велику кількість програмних рішень. Більшість із них інтегруються до складу систем активної допомоги водієві (ADAS - Advanced Driver Assistance Systems) [25]. Такі системи використовують технології комп'ютерного зору та методи

машинного навчання для обробки відеопотоку з камер автомобіля з метою аналізу дорожньої ситуації та надання рекомендацій водієві.

Mobileye [26] – це система допомоги водієві, яка використовує камери та алгоритми комп’ютерного зору для виявлення дорожньої розмітки, знаків, пішоходів і транспортних засобів. Використовуючи глибокі нейронні мережі, Mobileye здатна аналізувати дорожню ситуацію в реальному часі та попереджати водія про потенційні зіткнення, виїзд зі смуги або появу пішоходів. Система інтегрується з іншими функціями автомобіля, такими як адаптивний круїз-контроль і автоматичне екстрене гальмування, що значно підвищує безпеку руху.

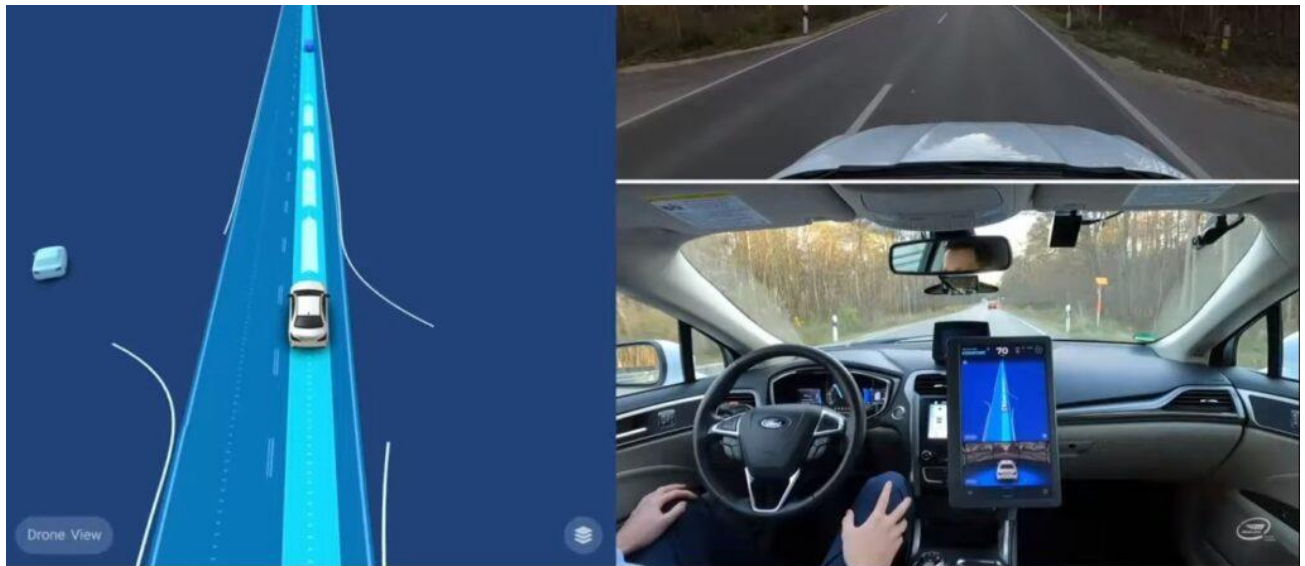


Рисунок 1.1. – Інтерфейс системи Mobileye під час розпізнавання об’єктів на дорозі [26]

Перевагами Mobileye є висока точність розпізнавання об’єктів навіть за складних погодних умов, можливість інтеграції з різноманітними автомобільними системами та підтримка великої кількості моделей автомобілів.

До недоліків можна віднести високу вартість системи, складність установки в деяких автомобілях та залежність якості роботи від правильної калібровки камери.

Tesla Autopilot [27] – це комплексна система автономного водіння, яка базується на багатьох відеокамерах, ультразвукових сенсорах та алгоритмах

глибокого навчання. Tesla Autopilot забезпечує підтримку функцій активної допомоги водієві, таких як утримання смуги руху, автоматичне гальмування, регулювання швидкості та автоматичне паркування. Постійні оновлення програмного забезпечення через "повітряні" оновлення дозволяють Tesla вдосконалювати алгоритми та розширювати можливості системи.



Рисунок 1.2. – Візуалізація роботи Tesla Autopilot на екрані автомобіля [27]

Перевагами Tesla Autopilot є глибока інтеграція з екосистемою автомобіля, постійне оновлення програмного забезпечення та потужна обчислювальна база для обробки відеопотоку в реальному часі. Серед недоліків варто відзначити обмежену доступність у деяких країнах через правові обмеження, високу вартість та потребу у високоякісному калібруванні сенсорів і камер для максимальної ефективності.

OpenPilot [28] – це відкрита програмна платформа для автономного керування автомобілем, яка використовує камери та алгоритми комп'ютерного зору для аналізу дорожньої обстановки. OpenPilot підтримує виявлення смуг руху, розпізнавання транспортних засобів та динамічне регулювання швидкості. Система орієнтована на ентузіастів і може бути встановлена навіть на смартфон або спеціалізований пристрій Comma Two.

Основними перевагами OpenPilot є відкритий вихідний код, що дозволяє адаптувати систему під власні потреби, активна підтримка спільноти розробників та можливість використання без значних витрат на обладнання. Недоліками є необхідність ручного встановлення і налаштування, потреба у сумісному автомобілі та менша, порівняно з комерційними системами, гарантія надійності.

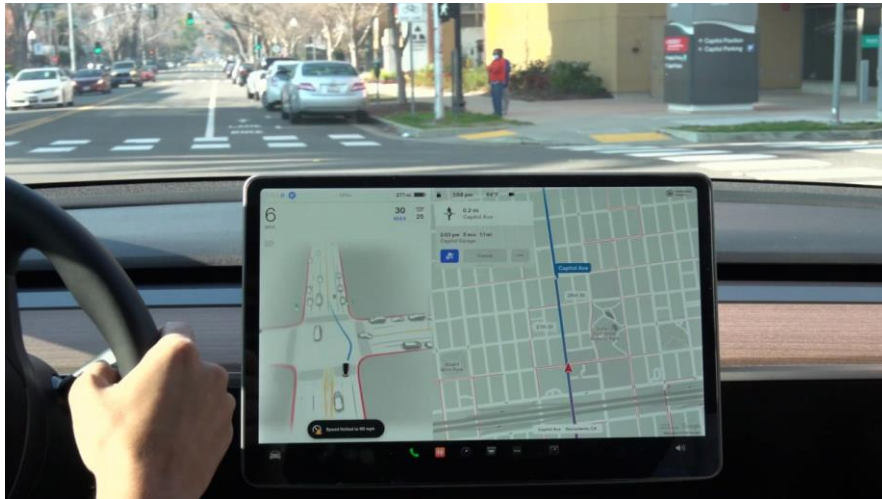


Рисунок 1.3. – Інтерфейс OpenPilot під час роботи системи допомоги водієві [28]

Отже, у сфері систем допомоги водіям розроблено кілька інноваційних рішень, які використовують комп'ютерний зір та машинне навчання для покращення безпеки дорожнього руху. Ці системи, здатні аналізувати відеопотік в реальному часі, розпізнавати дорожні знаки, смуги руху, пішоходів і транспортні засоби, а також здійснювати автоматичне коригування руху [29].

1.4 Мета, завдання та вимоги до реалізації інформаційної системи

Метою кваліфікаційної роботи бакалавра є допомога водію-початківцю шляхом розробки методу глибокого навчання інтегрованого у мобільний застосунок для аналізу відеопотоку руху автомобіля для контролю та формування рекомендацій. Це дозволить підвищити безпеку водія шляхом виявлення дорожніх знаків, розмітки, пішоходів та інших об'єктів у реальному часі, знижуючи ймовірність дорожньо-транспортних пригод.

Завдання:

- 1) проаналізувати предметну область автоматизованого розпізнавання об'єктів дорожньої інфраструктури, зокрема дорожніх знаків, розмітки, пішоходів та транспортних засобів;
- 2) розробити метод глибокого навчання для обробки відеопотоку з метою виявлення об'єктів дорожньої ситуації та формування рекомендацій водію;

- 3) інтегрувати розроблений метод у інформаційну систему у вигляді мобільного застосунку, що працює в реальному часі та взаємодіє з водієм;
- 4) провести тестування запропонованого методу та інформаційної системи в умовах, наближених до реального дорожнього руху, для оцінки ефективності та зручності використання.

Вимоги до реалізації:

- 1) точність і ефективність розпізнавання – алгоритми повинні забезпечувати високу точність у реальному часі, знижуючи кількість помилкових сповіщень;
- 2) робота в реальному часі – система повинна мати мінімальну затримку для оперативного попередження водія;
- 3) підтримка мобільних платформ – застосунок має бути доступний для Android та iOS для забезпечення широкого доступу;
- 4) інтеграція з автомобільними системами – система повинна взаємодіяти з іншими функціями автомобіля для покращення безпеки;
- 5) надійність та безпека – високий рівень захисту даних та стійкість до технічних збоїв;
- 6) інтуїтивно зрозумілий інтерфейс – простота в користуванні для водіїв без досвіду з можливістю налаштування функцій.

Отже, було обґрунтовано актуальність теми кваліфікаційної роботи, визначено її мету, завдання та основні вимоги до реалізації інформаційної системи [30]. З огляду на зростаючу кількість ДТП, пов'язаних з недосвідченістю водіїв, та стрімкий розвиток технологій глибокого навчання і мобільних платформ, створення інтелектуального мобільного застосунку-помічника є важливим та своєчасним рішенням. Формалізовані завдання та вимоги до методу глибокого навчання дозволяють сформулювати чіткий план його розробки та впровадження у мобільний застосунок, спрямований на підвищення рівня безпеки дорожнього руху та покращення водійських навичок початківців.

Розділ 2 Метод та засіб допомоги водіям-початківцям з використанням штучного інтелекту

2.1 Метод допомоги водіям-початківцям

2.1.1 Ідея методу

Ідея методу полягає в розробці засобу, який допоможе водіям-початківцям безпечно керувати автомобілем за рахунок автоматичного розпізнавання дорожніх знаків та відповідним рекомендаціям про швидкісний режим автомобіля в реальному часі. Реалізувати зазначене пропонується використанням технологій штучного інтелекту, зокрема глибокого навчання, для виявлення знаків на дорозі, обробки їхнього вмісту та інформування водія через інтуїтивно зрозумілий інтерфейс. Основна мета – зменшити когнітивне навантаження на водія-початківця, який ще не має достатнього досвіду для швидкого реагування на дорожні умови, та підвищити безпеку руху.

Основні ідеї методу та засобу у який буде інтегровано зазначений метод:

- засіб автоматично виявляє знаки обмеження швидкості, усуваючи потребу водія самотійно шукати та інтерпретувати їх – зазначене є особливо важливим для водіїв-новачків, які можуть пропустити знаки через неуважність або стрес;

- метод працює в реальному часі, обробляючи відеопотік із камери транспортного засобу та надаючи миттєві сповіщення – це дозволить водію швидко реагувати на зміну обмежень швидкості;

- засіб реалізований у вигляді застосування для iOS, з використанням SwiftUI та Core ML – зазначене дозволяє отримати портативне та доступне для широкого кола користувачів рішення;

- застосування нейромережі YOLOv8 забезпечує високу точність і швидкість детекції знаків, а додаткове використання OCR (через Vision Framework) дозволяє витягувати числові значення (наприклад, "50" або "70") для інформування водія;

– *інтерфейс системи не відволікатиме водія* – знаки відображатимуться у вигляді накладок (bounding boxes) на відеопотоці, а сповіщення подаватимуться у вигляді коротких текстових або звукових повідомлень.

Метод базується на трьох основних принципах:

- 1) *точність* – забезпечується навчанням YOLOv8 на датасеті дорожніх знаків;
- 2) *швидкість* – оптимізація моделі для роботи в реальному часі на iOS;
- 3) *зручність* – інтуїтивний інтерфейс, адаптований для водіїв-початківців.

Також передбачено можливість масштабування, у майбутньому систему можна розширити для розпізнавання інших типів знаків (наприклад, попереджувальних) або інтеграції з AR для проєктування інформації на лобовому склі.

2.1.2 Опис моделі глибокого навчання

Для реалізації системи допомоги водіям-початківцям запропоновано використати сімейство моделей глибокого навчання YOLOv8 (You Only Look Once, версія 8) [31]. YOLOv8 є сучасною моделлю для задач розпізнавання об'єктів у реальному часі, яка поєднує високу точність і швидкість обробки, що робить її ідеальною для роботи на мобільних пристроях, таких, у яких використовується iOS.

Модель глибокого навчання YOLOv8 (рис. 2.1) складається з:

- **Backbone.** Основою моделі є CSPDarknet53 (Cross Stage Partial Darknet), яка використовує згорткові шари для витягнення ознак із зображення. Вона має кілька рівнів (stages), що дозволяють обробляти ознаки різного масштабу (низькорівневі деталі та високорівневі семантичні особливості);
- **Neck.** Використовується FPN (Feature Pyramid Network) для агрегації ознак із різних рівнів backbone, що дозволяє ефективно виявляти об'єкти різного розміру (наприклад, далекі або маленькі знаки);
- **Head.** Голова моделі відповідає за передбачення. Вона генерує три типи виходів: координати bounding boxes (x, y, ширина, висота), ймовірність

належності до класу (наприклад, "знак обмеження швидкості") та загальну ймовірність наявності об'єкта об'єкта. YOLOv8 використовує якірні (anchor-based) передбачення, але з оптимізаціями для зменшення кількості помилкових спрацьовувань через Non-Maximum Suppression (NMS).

Особливість YOLOv8 полягає в тому, що вона підтримує автоматичне налаштування гіперпараметрів, має вбудовану аугментацію даних (мозаїка, змішування зображень) та підтримує експорт у Core ML для інтеграції з iOS.

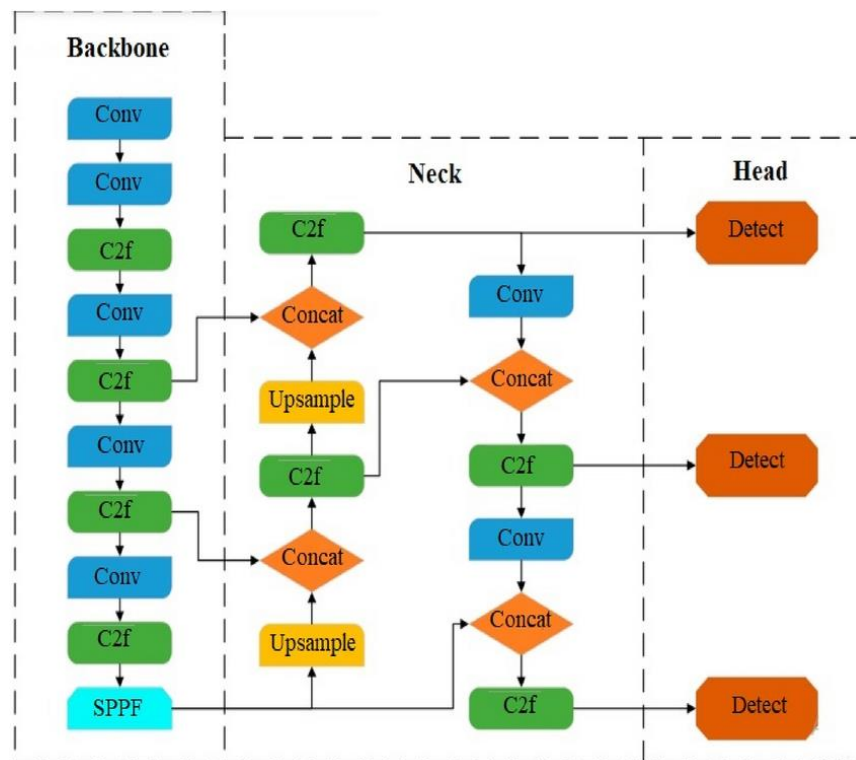


Рисунок 2.1 – Архітектура неймережі YOLO [32]

2.1.3 Опис датасету

Для донавчання моделі використовується відкритий датасет із платформи Kaggle під назвою Road Sign Detection [33] (рис. 2.2). Цей датасет створений для задач комп'ютерного зору, пов'язаних із розпізнаванням дорожніх знаків, і є популярним вибором для розробки систем автономного водіння та асистентів водія.



Рисунок 2.2 – Приклад зображень з датасету [34]

Нижче наведено характеристики дата сету:

- Обсяг і структура. Датасет містить 877 зображень із анотаціями у форматі XML (PASCAL VOC), які конвертуються у формат YOLO (текстові файли з координатами bounding boxes та класами). Загальна кількість анотованих об'єктів (дорожніх знаків) у датасеті становить понад 1200. Кожен об'єкт має відповідну мітку класу та координати (x, y, ширина, висота);

- Класи. Датасет включає кілька категорій дорожніх знаків, зокрема знаки обмеження швидкості (наприклад, "speed_limit_50", "speed_limit_70"), знаки зупинки, пішохідні переходи тощо. Для потреб проєкту основна увага приділялася знакам обмеження швидкості, які становлять приблизно 30% від загальної кількості об'єктів (близько 360 екземплярів);

- Умови зйомки. Зображення зібрані в реальних дорожніх умовах, що включають різні сценарії: денне та нічне освітлення, дощ, туман, а також різноманітні кути зйомки (з автомобіля, із різної відстані). Роздільна здатність зображень варіюється від 640x480 до 1920x1080 пікселів, що дозволяє моделі адаптуватися до різних розмірів знаків на дорозі;

- Розподіл даних. Датасет поділений на три підмножини: тренувальну (80%, 702 зображення), валідаційну (10%, 87 зображень) та тестову (10%, 88 зображень). Це забезпечує можливість оцінки моделі на незалежних даних та уникнення перенавчання;

- Попередня обробка. Перед навчанням усі зображення нормалізуються до розміру 640x640 пікселів, що відповідає вимогам YOLOv8. Анотації у форматі XML конвертуються у формат YOLO за допомогою скриптів (наприклад, Python), де кожен об'єкт описується у вигляді: `<class_id> <x_center> <y_center> <width> <height>`. Додатково застосовується аугментація даних

(повороти, зміна яскравості, шум), щоб підвищити стійкість моделі до різних умов.

Дані форматуються у файл `your_dataset.yaml`, який містить шляхи до тренувальної, валідаційної та тестової підмножин, а також перелік класів (наприклад, `"speed_limit_50"`, `"speed_limit_70"`).

2.1.4. Основні підходи до розробки методу

В останні роки активно зростає потреба в інтелектуальних транспортних системах, які покращують безпеку руху. Особливу увагу заслуговують системи підтримки водіїв-початківців, оскільки саме ця категорія найчастіше потрапляє у ДТП через недосвідченість, відсутність навичок та труднощі з орієнтацією в дорожніх умовах.

Сучасні системи допомоги водіям покликані зменшити навантаження на людину під час керування автомобілем та підвищити рівень безпеки. У межах даного дослідження запропоновано метод штучного інтелекту, що дозволяє виявляти дорожні знаки обмеження швидкості в режимі реального часу.

Метод допомоги водіям-початківцям базується на використанні штучного інтелекту для автоматичного розпізнавання дорожніх знаків обмеження швидкості в реальному часі з подальшим інформуванням водія. Метод має включати наведені далі основні етапи.

- Збір даних. Використовується датасет зображень дорожніх знаків (наприклад, з Kaggle, як вказано в проєкті), що містить анотовані зображення знаків обмеження швидкості. Датасет підготовлено у форматі, сумісному з YOLO (наприклад, `your_dataset.yaml`);

- Донавчання моделі. Застосовується модель YOLOv8/YOLOv11, яка навчається на підготовленому датасеті для детекції дорожніх знаків. Навчання проводиться з параметрами: 50 епох, розмір зображення 640x640 пікселів, використовуючи попередньо навчену модель (`yolov8n.pt`);

– Обробка даних у реальному часі. Модель аналізує відеопотік із камери транспортного засобу, виявляє знаки обмеження швидкості та визначає їхні координати (bounding boxes) на зображенні;

– Інформування водія. Виявлені знаки обробляються для витягнення інформації (наприклад, значення обмеження швидкості за допомогою OCR), після чого водій отримує візуальні або звукові сповіщення через інтерфейс системи;

– Оптимізація. Модель конвертується в формат Core ML для ефективної роботи на мобільних пристроях (iOS), з оптимізацією під FP16 та без використання пакетної обробки.

Схема методу забезпечує високу точність детекції (за рахунок YOLO) та швидке реагування, що є критично важливим для безпеки водіїв-початківців.

2.1.5. Кроки методу

Запропонований метод складатиметься із двох завдань: донавчання моделі та використання донавченої моделі.

У ході проектування методу розпізнавання дорожніх знаків обмеження швидкості засобами глибокого навчання можна виділити основні кроки для навчання моделі (рисунок 2.3).

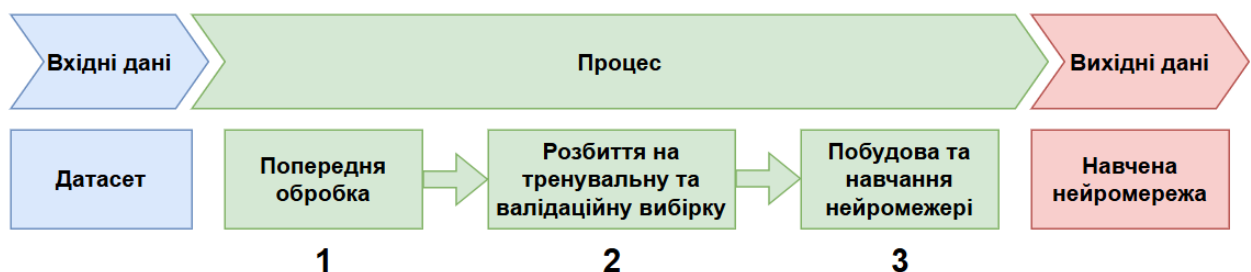


Рисунок 2.3 – Схема навчання моделі

На вхід надходить набір зображень дорожніх знаків, кожне з яких має відповідну мітку (наприклад, тип знака, числове значення). Ці зображення формують датасет для навчання.

На Кроці 1 (попередня обробка) виконується підготовка вхідних даних, які надходять із датасету. Цей етап включає завантаження зображень, їх розпаковування та конвертацію анотацій у формат, сумісний із моделлю YOLOv8, що забезпечує основу для подальшого аналізу.

На Кроці 2 (розпізнавання на тренувальну вибірку) здійснюється обробка підготовлених даних із застосуванням алгоритмів глибокого навчання, зокрема YOLOv8, для ідентифікації дорожніх знаків на зображенні. Цей етап включає тренування моделі на основі аугментованих даних, що підвищує її адаптивність до різних умов.

На Кроці 3 (побудова та навчання нейромережі) завершується створення архітектури моделі, її оптимізація та інтеграція в iOS-додаток через Core ML. Цей етап передбачає експорт моделі та відображення результатів у SwiftUI, забезпечуючи реальну підтримку водіїв-початківців.

Результатом є навчена модель, тобто нейронна мережа зі збереженими вагами, готова до використання на нових зображеннях.

На рисунку 2.4 зображена схема використання донавченої моделі.

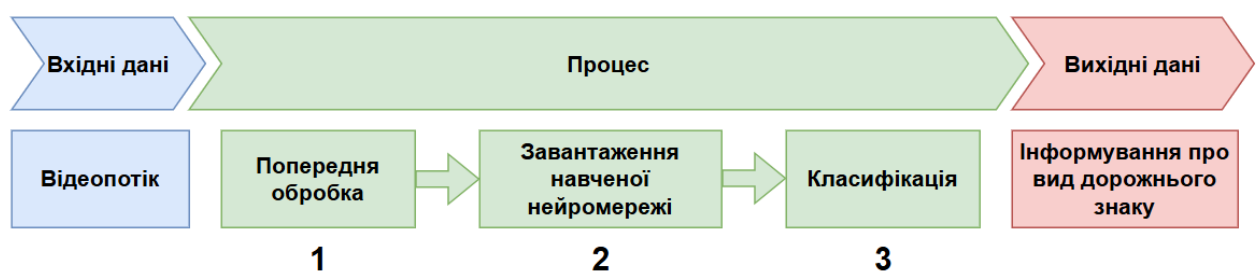


Рисунок 2.4 – Схема використання моделі

Вхідними даними є нове зображення з відеопотоку камери, яке потрібно розпізнати (виявити знак та його значення), без жодної додаткової інформації про нього.

На Кроці 1 (попередня обробка) зображення обробляється аналогічно до етапу навчання: змінюється його розмір до 640x640 пікселів і виконується нормалізація для приведення даних до формату, сумісного з моделлю.

На Кроці 2 (завантаження навченої моделі YOLOv8) у систему завантажується раніше навчена модель YOLOv8, яка вже оптимізована для розпізнавання дорожніх знаків і готова до використання в реальному часі.

На Кроці 3 (класифікація) модель YOLOv8 виконує детекцію, визначаючи координати знаків (bounding boxes) і їхній клас (наприклад, "speed_limit_50"), після чого OCR (через VNRecognizeTextRequest) розпізнає числові значення на знаках, такі як "50".

У результаті вихідними даними є інформація про виявлений знак (наприклад, "Обмеження швидкості: 50 км/год"), яка передається водію через інтерфейс.

Для навчання нейронної мережі YOLOv8, яка реалізує детекцію дорожніх знаків, було використано структурований датасет Road Sign Detection із Kaggle. Дані зберігаються у вигляді зображень із відповідними анотаціями у форматі XML, які конвертуються у формат YOLO. Для кожного зразка наведено атрибути в таблиці 2.1.

Таблиця 2.1 – Атрибути

Атрибут	Пояснення
Назва зображення	Ім'я файлу зображення знака
Координати	Координати bounding box (x, y, ширина, висота)
Клас знака	Тип знака (наприклад, "speed_limit_50")

Нижче на рисунку 2.5 зображено приклад зображення дорожнього знака з датасету.



Рисунок 2.5 – Приклад зображення знака з дата сету

З метою зменшення надмірної різномірності класів та забезпечення стабільності навчання, з усіх типів знаків було відібрано категорію знаків обмеження швидкості (наприклад, "speed_limit_30", "speed_limit_50", "speed_limit_70"). Таким чином, модель навчається розпізнавати знаки серед чітко окреслених класів, що відповідають типовим значенням швидкості.

Кількість зразків у кожному класі варіюється, що підтверджується побудованою стовпчиковою діаграмою. Вона дозволяє візуалізувати співвідношення класів у датасеті дорожніх знаків (рисунок 2.6). Видно, що деякі класи представлені значно більше, ніж інші (наприклад, "speed_limit_50" та "speed_limit_70" мають найбільшу кількість зразків, а "speed_limit_30" - найменшу).

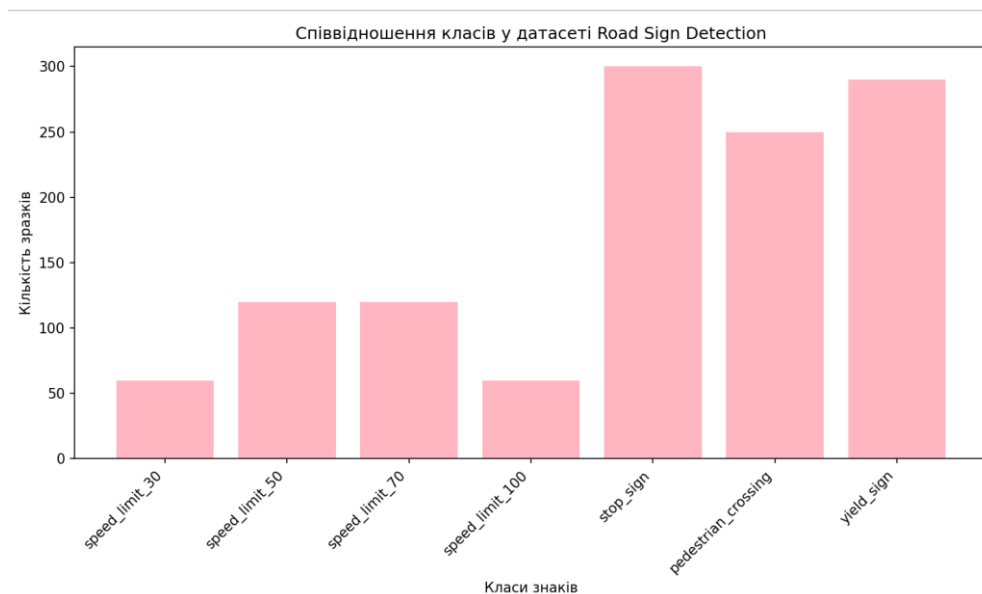


Рисунок 2.6 – Стовпчикова діаграма співвідношення класів

Обраний датасет виступає основою для формування навчальної, валідаційної та тестової вибірок, які дозволяють коректно оцінити ефективність побудованої моделі на основі архітектури YOLOv8 у задачі детекції дорожніх знаків.

2.2. Особливості реалізації запропонованого методу

Реалізація методу розпізнавання дорожніх знаків обмеження швидкості для системи допомоги водіям-початківцям шляхом побудови системи детекції об'єктів за допомогою нейромережі YOLOv8. Для реалізації використано бібліотеку Ultralytics, яка забезпечує зручний інтерфейс для навчання, експорту та розгортання моделей YOLO, а також інтеграцію з iOS через Core ML.

Ultralytics - це бібліотека з відкритим кодом, розроблена для спрощення роботи з моделями YOLO [35]. Вона надає інструменти для навчання, оцінки та експорту моделей у різні формати, включаючи Core ML, що ідеально підходить для реалізації глибоких нейронних мереж на мобільних пристроях. Ultralytics підтримує роботу як на CPU, так і на GPU, що дозволяє прискорити обчислення під час навчання моделі на великих датасетах, таких як: Road Sign Detection.

У методі Ultralytics виступає як основний інструмент для навчання та експорту моделі: вона забезпечує автоматизоване налаштування гіперпараметрів (наприклад, розміри якорів), вбудовану аугментацію даних (мозаїка, повороти, зміна яскравості) та експорт моделі у формат Core ML для подальшої інтеграції з iOS. Загальний принцип роботи бібліотеки Ultralytics, яка забезпечує інструментарій для навчання моделей YOLO (You Only Look Once), базується на побудові конвеєра обробки даних, що охоплює повний цикл розробки моделі: від підготовки даних до виводу моделі у виробниче середовище (рисунок 2.7).

Кожен етап - завантаження даних, попередня обробка, навчання, перевірка якості та експорт моделі - реалізується у вигляді окремої логічної стадії, яка автоматизується через високорівневі команди CLI або програмний інтерфейс Python.

Основні етапи виглядають так:

- Завантаження датасету. Підтримуються стандартні формати (YOLO, COCO, VOC), а також власні YAML-описувачі. Ultralytics автоматично структурує директорії, перевіряє якість розмітки та підготовлює зображення для навчання;
- Передобробка зображень. Включає зміну розміру (наприклад, до 640×640), нормалізацію, аугментацію (зміна яскравості, повороти, випадкові обтинання) для покращення генералізації моделі;
- Навчання моделі YOLOv8. Визначається тип моделі (yolov8n, yolov8s, yolov8m тощо), кількість епох, розмір пакету та розділення на train/val. Навчання виконується з автоматичним збереженням найкращої моделі (best.pt) за метрикою mAP (mean Average Precision).
- Оцінка якості. Модель тестується на валідаційному підмноженні, обчислюється точність (precision), повнота (recall), середня точність (mAP@0.5, mAP@0.5:0.95), час обробки;
- Експорт у формат Core ML. Через команду `'yolo export format=coreml'` модель конвертується у формат .mlmodel, придатний до інтеграції в iOS-застосунки.

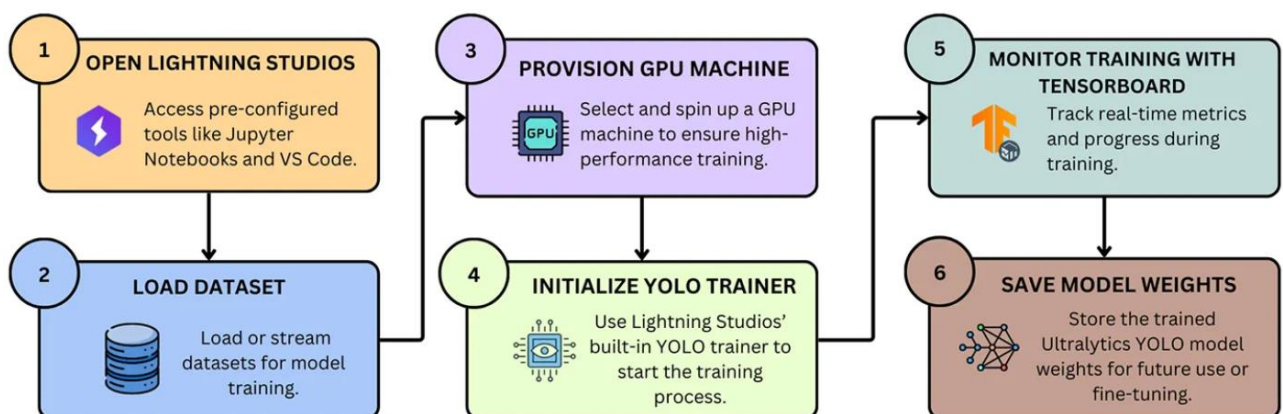


Рисунок 2.7 – Схема конвеєра обробки даних в Ultralytics [36]

Core ML - це фреймворк від Apple для інтеграції моделей машинного навчання в iOS-додатки [37]. Він забезпечує оптимізовану обробку моделей на

пристроях із Neural Engine, підтримуючи апаратне прискорення для швидкої роботи в реальному часі. Core ML дозволяє:

- Імпортувати модель YOLOv8 у форматі `.mlmodel``;
- Виконувати інференс (детекцію) на відеопотоці з камери;
- Оптимізувати модель (наприклад, використання FP16 для зменшення розміру та прискорення);
- Інтегруватися з Vision Framework для обробки зображень і розпізнавання тексту (OCR).

Архітектура Core ML базується на трьох основних компонентах: модель (`.mlmodel``), обчислювальний граф і апаратне прискорення (CPU, GPU, Neural Engine). Ця структура забезпечує ефективну обробку даних на iOS-пристроях (рисунок 2.8).

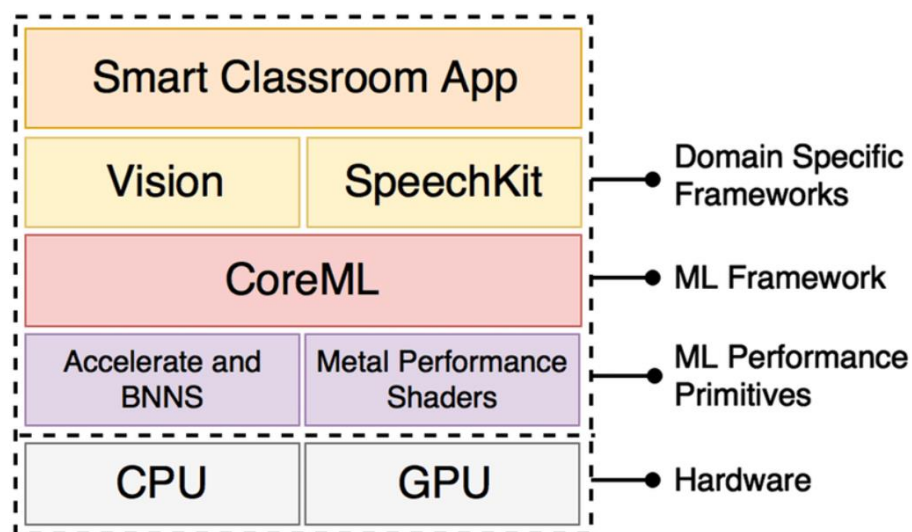


Рисунок 2.8 – Архітектура Core ML [38]

Для попередньої обробки зображень у процесі навчання використовуються вбудовані інструменти Ultralytics. Вони дозволяють змінювати розмір зображень до 640x640 пікселів, нормалізувати піксельні значення та застосовувати аугментацію (мозаїка, повороти $\pm 15^\circ$, зміна яскравості $\pm 20\%$), що підвищує стійкість моделі до різних умов. Під час використання

моделі на iOS зображення з відеопотоку також нормалізуються до 640x640 пікселів перед подачею в модель.

У методі застосовується техніка трансферного навчання: базові шари YOLOv8n ініціалізуються з попередньо навчених ваг ('yolov8n.pt'), що дозволяє зберегти узагальнюючі властивості, отримані на великих датасетах (наприклад, COCO). Навчання проводиться лише для адаптації моделі до датасету дорожніх знаків (Road Sign Detection).

2.2.1. Вибір підходу

Для реалізації системи виявлення дорожніх знаків, зокрема знаків обмеження швидкості, було обрано гібридний підхід, що поєднує глибоке навчання на Python-платформі з подальшою інтеграцією готової моделі в мобільний застосунок для iOS. Такий підхід базується на поетапному розділенні завдань: етапу навчання та етапу інференсу (використання моделі в реальному середовищі). Навчання здійснюється локально за допомогою фреймворку Ultralytics YOLOv8 або YOLOv11, який реалізується через Python. Ultralytics забезпечує зручний API, який дозволяє швидко запускати повний цикл - від підготовки датасету до збереження готової моделі. Основними причинами вибору цього підходу стали підтримка об'єктного детектування в реальному часі, простота експорту до форматів, сумісних з Apple Core ML, а також можливість навчання на спеціалізованому датасеті дорожніх знаків.

Після завершення тренування модель експортується у формат .mlmodel, який є власним форматом Apple для машинного навчання. На стороні мобільного застосунку інтеграція виконується в середовищі Xcode з використанням SwiftUI, Core ML та Vision Framework. SwiftUI відповідає за створення адаптивного інтерфейсу користувача, AVFoundation використовується для захоплення відео з камери в реальному часі, а VNCoreMLRequest з Vision Framework реалізує обробку кадрів через модель, виявляє знаки та відображає їх на екрані за допомогою рамок (bounding boxes).

Обраний підхід має низку переваг. По-перше, він забезпечує кросплатформенність у межах Apple-екосистеми, оскільки всі iOS-пристрої підтримують Core ML. По-друге, система забезпечує високу продуктивність, оскільки інференс моделі виконується апаратно, використовуючи GPU або Neural Engine, без потреби в інтернет-з'єднанні. По-третє, підхід зручний для розробки, оскільки код Python повністю відокремлений від SwiftUI, що дозволяє незалежно розвивати обидві частини системи. І, нарешті, рішення є масштабованим: модель можна донавчити, експортувати новий .mlmodel-файл і просто замінити його у застосунку без необхідності переписування логіки роботи програми.

Таким чином, обраний підхід поєднує гнучкість середовища глибокого навчання з ефективністю мобільного застосування, що критично важливо для систем реального часу, зокрема для інтелектуальних асистентів водія.

2.3. Проектування інформаційної системи допомоги водіям

У межах реалізації методу інтелектуальної допомоги водіям-початківцям потрібно спроектувати інформаційну систему, яка поєднає в собі функції комп'ютерного зору, мобільного застосунку та користувацького інтерфейсу. Основною метою запропонованого методу є виявлення дорожніх знаків, зокрема знаків обмеження швидкості, у реальному часі з відеопотоку камери мобільного пристрою та надання відповідної інформації водієві.

Проектування інформаційної системи базується на принципах об'єктно-орієнтованого проектування, що дозволяє структурувати логіку роботи у вигляді взаємодії між окремими класами. Такий підхід забезпечує гнучкість та масштабованість - кожен компонент відповідає за окрему функціональність: від обробки відеопотоку до виклику моделі машинного навчання та виводу результатів на екран.

Для візуалізації архітектури системи побудовані UML-діаграми, що описують основні компоненти та логіку їх взаємодії [39].

На рисунку 2.9 зображено діаграму класів інформаційної системи, яка реалізує процес класифікації дорожніх знаків на основі аналізу відеопотоку. Діаграма відображає структуру системи відповідно до її функціональної логіки [40].

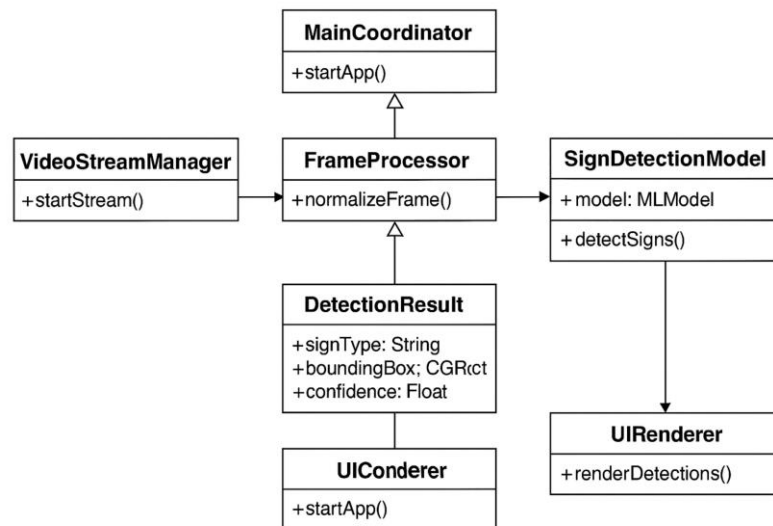


Рисунок 2.9 – Діаграма класів інформаційної системи

Діаграма класів інформаційної системи розпізнавання дорожніх знаків, представлена на зображенні, відображає основні компоненти системи та їх взаємозв'язки. Вона побудована на основі функціональної логіки проекту та включає класи, які забезпечують обробку відеопотоку, детекцію знаків і виведення результатів водію.

Основні класи системи:

1. **MainCoordinator:**
 - Відповідає за ініціалізацію та координацію роботи всієї системи;
 - Метод: ``startApp()`` - запускає додаток, ініціалізує всі компоненти (захоплення відеопотоку, модель, інтерфейс) і забезпечує їх взаємодію.
2. **VideoStreamManager:**
 - Управляє захопленням відеопотоку з камери iOS-пристрою;
 - Метод: ``startStream()`` - запускає захоплення відеопотоку через AVFoundation із частотою 30 FPS і передає кадри для подальшої обробки.

3. FrameProcessor:

- Відповідає за попередню обробку кадрів перед подачею їх у модель;
- Метод: `normalizeFrame()` - виконує нормалізацію кадру (зміна розміру до 640x640 пікселів, масштабування значень пікселів до діапазону [0, 1]) для сумісності з моделлю YOLOv8.

4. SignDetectionModel:

- Реалізує основну функціональність детекції знаків;
- Атрибут: `model: MLModel` - зберігає екземпляр моделі YOLOv8 у форматі Core ML;
- Метод: `detectSigns()` - виконує інференс моделі YOLOv8 через `VNCoreMLRequest`, повертаючи виявлені знаки у вигляді об'єктів `DetectionResult`.

5. DetectionResult:

- Зберігає результати детекції знаків;
- Атрибути: `signType: String` - тип знака (наприклад, "speed_limit_50"), `boundingBox: CGRect` - координати bounding box (x, y, ширина, висота), `confidence: Float` - ймовірність детекції (наприклад, 0.95).

6. UIRenderer:

- Відповідає за відображення результатів детекції на екрані;
- Метод: `renderDetections()` - малює bounding boxes на відеопотоці, відображає текстові сповіщення (наприклад, "Обмеження швидкості: 50 км/год").

Зв'язки між класами:

- `MainCoordinator` викликає `startStream()` у `VideoStreamManager` для початку захоплення відео потоку;
- `VideoStreamManager` передає кадри до `FrameProcessor`, який викликає `normalizeFrame()` для їх обробки;
- Оброблені кадри передаються до `SignDetectionModel`, який викликає `detectSigns()` для детекції знаків;

– Результати детекції ('DetectionResult') передаються до 'UIRenderer', який викликає 'renderDetections()' для відображення інформації водію.

На рисунку 2.10 показано діаграму активності, що ілюструє перехід між різними етапами діяльності [41].

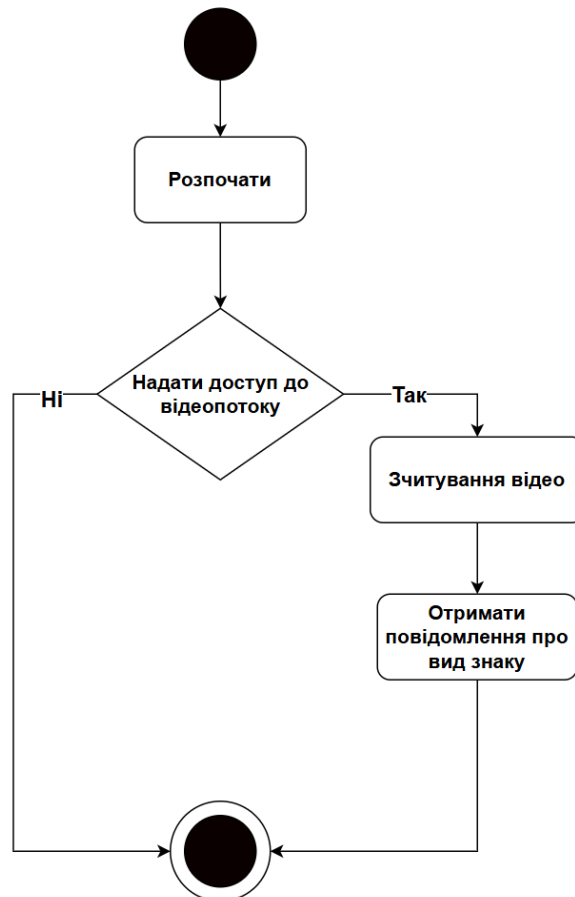


Рисунок 2.10 – Діаграма активності

Діаграма активності показує послідовність дій у системі від запуску додатка до виведення результатів водієві. Після запуску на iOS-пристрої ініціалізується MainCoordinator і викликається метод startApp(). Далі система запитує дозвіл на доступ до камери. Якщо дозвіл надано, VideoStreamManager розпочинає захоплення відео, а кадри передаються на обробку: нормалізуються та подаються до моделі для виявлення знаків обмеження швидкості.

У разі виявлення знаку його область (bounding box) аналізується за допомогою Vision Framework для розпізнавання числового значення. Після цього

метод `renderDetections()` у класі `UIRenderer` виводить `bounding box`, текстове сповіщення. Завершення роботи додатка зупиняє відеопотік і припиняє всі процеси.

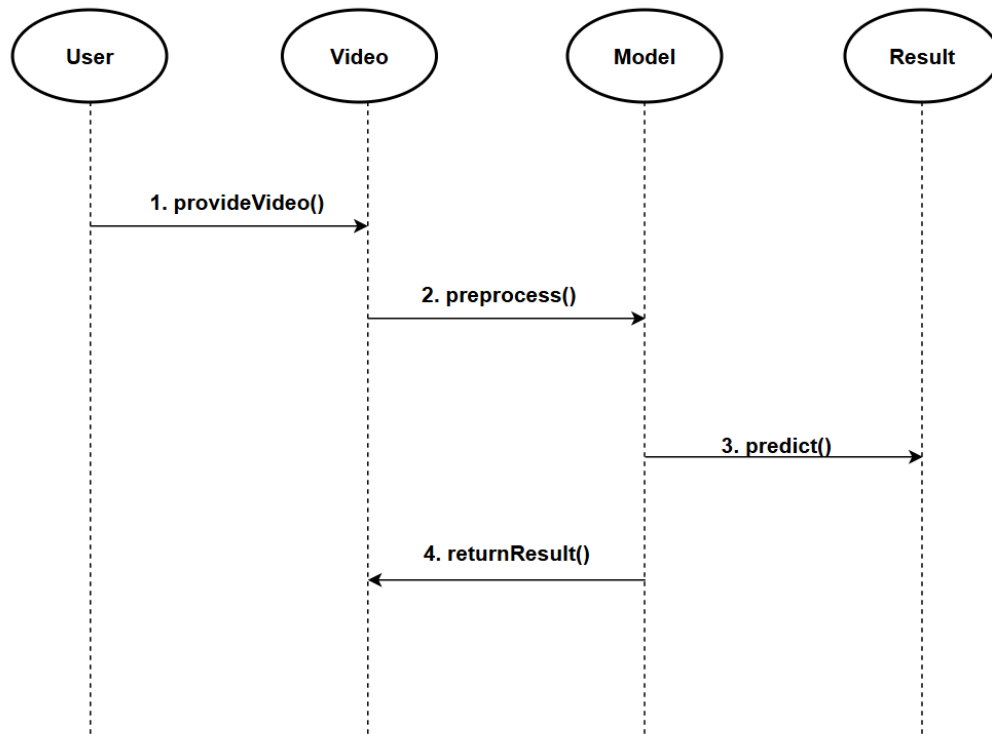


Рисунок 2.11 – Діаграма послідовності

Діаграма послідовності (рисунок 2.11) ілюструє часову взаємодію між компонентами системи [42]. У процесі беруть участь чотири основні елементи: Користувач, Відеопотік, Модель та Результат. Користувач надає відео, яке після обробки передається до моделі для аналізу. Після виконання передбачення, результат повертається користувачеві. Ця діаграма надає чітке уявлення про послідовність викликів функцій та взаємодії об'єктів у системі.

Діаграма варіантів використання ілюструє функціональні можливості системи з точки зору користувачів [43]. Вона включає двох акторів: User (водій-початківець) і `SignDetectionModel` (внутрішній модуль системи) (рисунок 2.12).

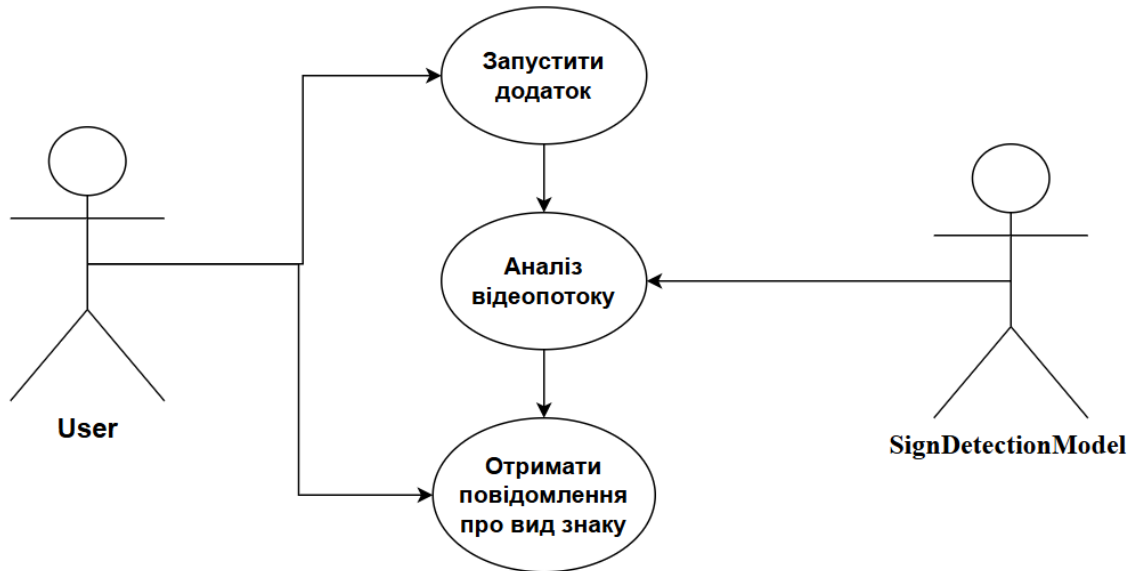


Рисунок 2.12 – Діаграма варіантів використання

User (Водій) може запустити додаток, а також може отримати сповіщення про знаки.

SignDetectionModel (Модуль детекції) може виявити знаки, тобто виконує детекцію знаків обмеження швидкості за допомогою YOLOv8.

Основні варіанти використання: запуск системи, розпізнавання знаків, інформування водія. Діаграма корисна для високорівневого розуміння функціоналу системи.

Діаграма станів відображає зміну внутрішнього стану системи в процесі розпізнавання знаків (рисунок 2.13) [44].

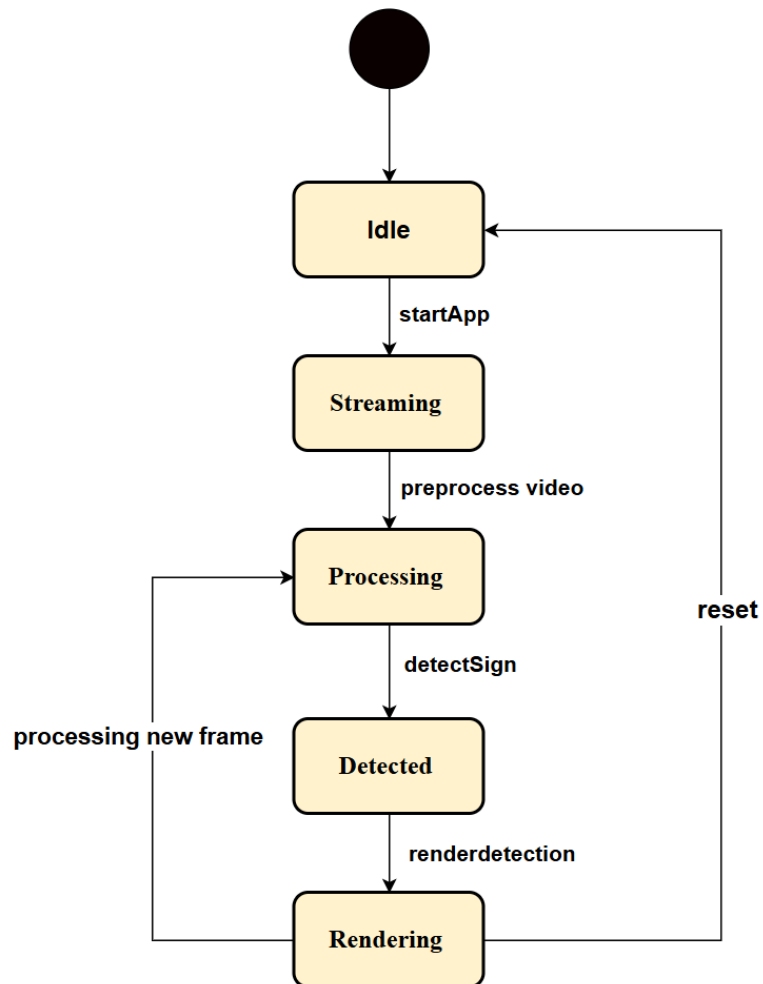


Рисунок 2.13 – Діаграма станів

Основні стани:

- Idle (Очікування) – початковий стан, коли система чекає на запуск користувачем;
- Streaming (Захоплення відеопотоку) - система захоплює відеопотік після виклику `startStream()`;
- Processing (Обробка) - виконується нормалізація кадрів і детекція знаків;
- Detected (Виявлено)- знак виявлено, результати передані до `UIRenderer``;
- Rendering (Відображення)- система відображає сповіщення водію.

Переходи між станами:

- З Idle → Streaming. Викликається подією запуску додатку (`startApp()`);
- З Streaming → Processing. Кожен кадр обробляється (`normalizeFrame()`);
- З Processing → Detected: Виявлено знак (`detectSigns()`);
- З Detected → Rendering. Відображаються сповіщення (`renderDetections()`);
- З Rendering → Processing. Система повертається до обробки наступного кадру.

У будь-який момент можливий перехід `reset`, який повертає систему до стану Idle (наприклад, при зупинці додатку).

2.4 Висновки

Отже, було запропоновано метод допомоги водіям-початківцям із використанням технологій штучного інтелекту, зокрема глибокого навчання. Запропонований метод базується на автоматичному розпізнаванні дорожніх знаків обмеження швидкості в реальному часі за допомогою моделі YOLOv8, із подальшим інформуванням водія через інтуїтивний інтерфейс. Описано ідею методу, яка полягає в зменшенні когнітивного навантаження на водія-початківця та підвищенні безпеки руху шляхом своєчасного надання рекомендацій щодо швидкісного режиму. Було визначено основні принципи роботи системи: точність, швидкість і зручність, а також можливість масштабування для майбутнього розширення функціоналу, наприклад, додавання розпізнавання інших типів знаків чи інтеграції з AR.

Також було описано архітектуру моделі YOLOv8, яка включає Backbone (CSPDarknet53), Neck (FPN) і Head для передбачення координат і класів знаків, а також особливості датасету Road Sign Detection із Kaggle, який планується використати для донавчання моделі. Окремо розглянуто основні етапи методу,

що охоплюють попередню обробку даних, тренування моделі, обробку відеопотоку та виведення результатів, а також спроектовано інформаційну систему з використанням UML-діаграм (класів, активності, послідовності, варіантів використання, станів), які відображають логіку роботи майбутнього додатку.

Для завершення розробки та оцінки ефективності запропонованого методу необхідно виконати наступні завдання, які відповідають структурі розділу 3:

- Розробка повнофункціонального iOS-додатку. На основі запропонованого методу розробити завершений iOS-додаток, який включатиме всі функції (захоплення відеопотоку, детекція знаків, OCR, сповіщення), і протестувати його в реальних дорожніх умовах для оцінки ефективності у зменшенні когнітивного навантаження водіїв-початківців;

- Дослідження ефективності реалізації. Провести експериментальні дослідження продуктивності моделі YOLOv8 на різних iOS-пристроях (наприклад, iPhone 12, iPhone 7), вимірявши швидкість обробки (FPS), енергоспоживання і точність детекції в різних умовах (денне/нічне освітлення, дощ), щоб визначити обмеження та напрямки оптимізації;

- Функціональне тестування системи. Протестувати систему на функціональну відповідність, перевіряючи коректність роботи всіх компонентів у різних сценаріях (швидка зміна знаків, низька видимість) та точність OCR у складних умовах (відблиски, забруднення знаків), щоб довести коректність роботи перед впровадженням у реальних умовах.

Розділ 3 Експериментальне дослідження програмної реалізації запропонованого методу

3.1 Опис інтерфейсу користувача

Інтерфейс користувача [45] є важливою складовою будь-якої інформаційної системи, адже саме через нього відбувається взаємодія між користувачем та функціональністю додатка. У розробленій системі підтримки водіїв-початківців інтерфейс було створено з урахуванням принципів зручності, мінімалізму та швидкого доступу до основної інформації під час керування транспортним засобом.

У цьому розділі представлено загальний вигляд графічного інтерфейсу мобільного додатка, його основні елементи та сценарії взаємодії. Окрему увагу приділено функціональності виведення попереджень про знаки обмеження швидкості, візуальних позначень на екрані.

Інтерфейс розроблено відповідно до платформи iOS з використанням SwiftUI, що дозволило досягти адаптивності, чіткості графіки та зручності навігації. Наведені зображення та описи відображають основні компоненти застосунку та підтверджують відповідність вимогам до зручності користування під час руху.

Після запуску мобільного додатка користувач одразу стикається з системним сповіщенням, у якому запитується дозвіл на доступ до камери пристрою. Це стандартне вікно, яке відображається відповідно до політик безпеки операційної системи iOS.

Дозвіл на доступ до камери є критично необхідним, оскільки основна функціональність додатка - виявлення дорожніх знаків у режимі реального часу - базується на обробці відеопотоку. Користувач має два варіанти дій: надати дозвіл або відхилити запит. У разі відмови система не зможе активувати камеру, і відповідно функціонал розпізнавання знаків буде недоступний.

Такий підхід забезпечує дотримання вимог конфіденційності та дозволяє користувачеві контролювати доступ до апаратних ресурсів пристрою.

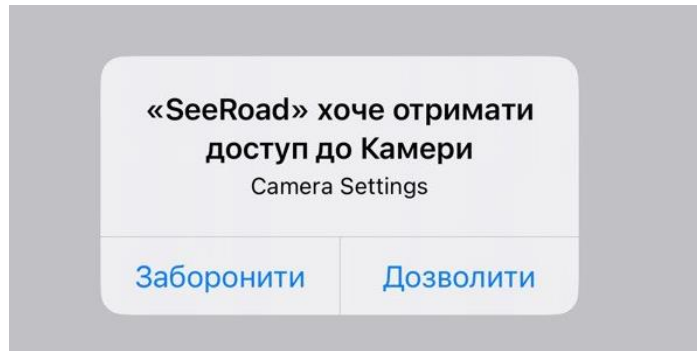


Рисунок 3.1 – Сповіщення про дозвіл користування камерою

Якщо користувач відмовляється надати дозвіл на доступ до камери, додаток виводить попередження, вказуючи, що камера необхідна для розпізнавання знаків у реальному часі. Користувачу пропонується увімкнути дозвіл у налаштуваннях для повноцінного використання, що запобігає непорозумінням щодо роботи програми.

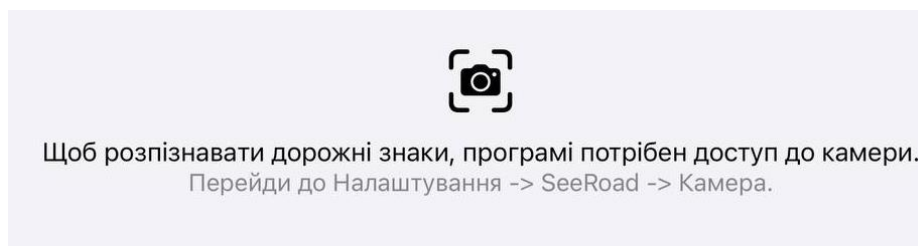


Рисунок 3.2 – Сповіщення про доступ до камери

Після того як користувач надає дозвіл на доступ до камери, додаток автоматично активує відеопотік і переходить у режим розпізнавання дорожніх знаків у реальному часі. Камера працює безперервно, обробляючи кожен кадр з метою виявлення об'єктів, що відповідають ознакам знаків обмеження швидкості. Як тільки система розпізнає знак на зображенні, він автоматично виділяється на екрані за допомогою зеленого прямокутника (bounding box). Це візуальне підсвічування вказує на точне місце розташування знайденого знаку в полі зору камери.

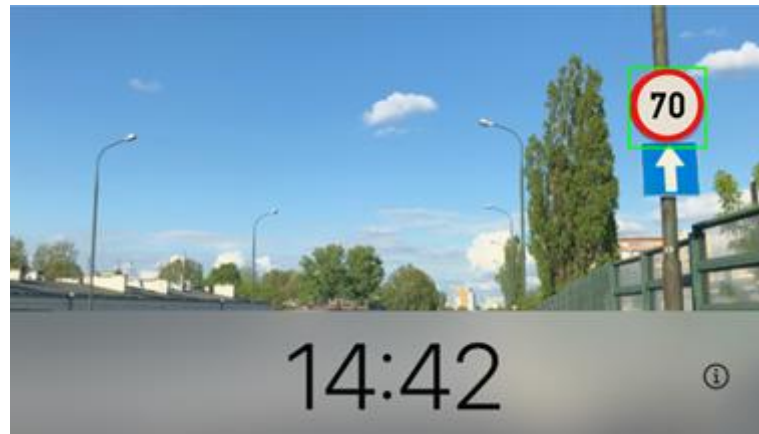


Рисунок 3.3 – Розпізнавання знаку

Після успішного розпізнавання дорожнього знаку система не лише обводить його зеленим прямокутником у кадрі, але й виводить збільшене зображення цього знаку у нижньому лівому куті екрана. Такий елемент інтерфейсу слугує для зручності користувача, щоб той міг чітко побачити, який саме знак був ідентифікований.

Знак відображається у збільшеному розмірі на нейтральному фоні, що дозволяє легко розпізнати його значення навіть у разі поганого освітлення або неідеальних умов на відео. Це рішення особливо корисне для водіїв-початківців, які можуть не одразу помітити знак на дорозі, але отримують наочне підтвердження через інтерфейс додатка.

Такий спосіб подачі інформації забезпечує швидке сприйняття критично важливих дорожніх вказівок та зменшує ризик пропуску обмежень швидкості.

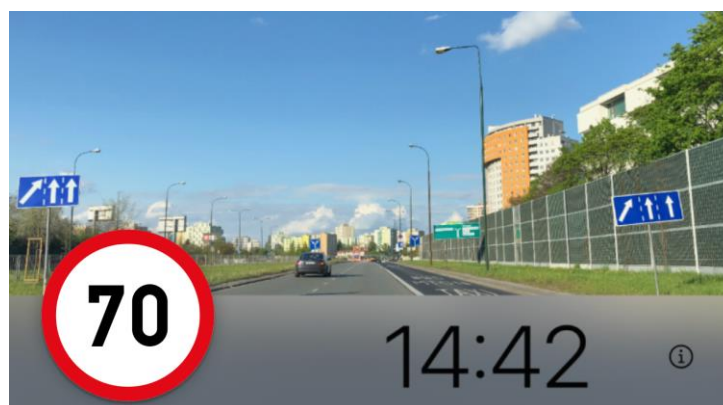


Рисунок 3.4 – Збільшене зображення розпізнаного знаку

У правому нижньому куті екрана розміщено інформаційну кнопку (іконку "i"), яка слугує для ознайомлення користувача з можливостями та функціональністю додатка. Цей елемент інтерфейсу особливо корисний для нових користувачів, які запускають програму вперше або бажають детальніше дізнатися про її призначення та принцип роботи.

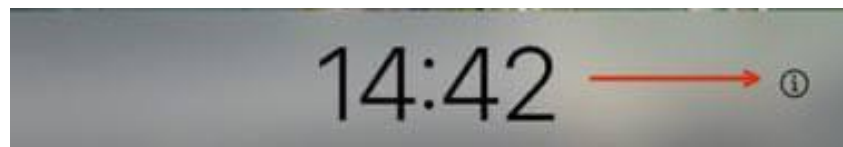


Рисунок 3.5 – Інформаційна кнопка

Натиснувши на кнопку, користувач відкриває спеціальне вікно або екран з короткою інструкцією щодо використання системи: як вона працює, які знаки розпізнає, як інтерпретувати повідомлення на екрані тощо. Це дозволяє ознайомитися з додатком ще до початку руху, не відволікаючись під час керування транспортним засобом.

Такий елемент сприяє підвищенню зручності та безпеки, оскільки дає змогу водієві заздалегідь зрозуміти логіку роботи застосунку та його інтерфейс.

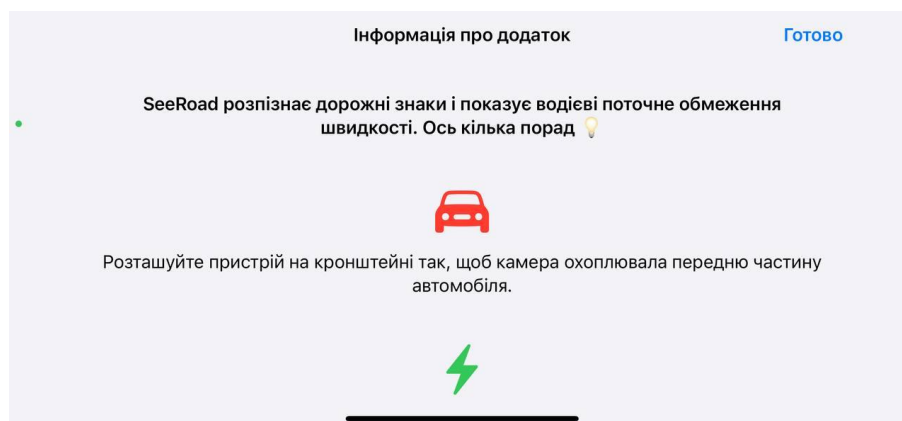


Рисунок 3.6 – Інформація про розташування пристрою

Щоб система коректно розпізнавала дорожні знаки, потрібно правильно встановити смартфон. Потрібно використовувати надійний автомобільний тримач (кронштейн), який дозволяє зафіксувати телефон у стабільному

положенні. Камера має бути спрямована прямо на дорогу попереду, не закривати її огляд склом, дівірниками, дзеркалами чи іншими предметами. Уникати розміщення пристрою під кутом або занадто низько, оскільки це може завадити точному розпізнаванню знаків.

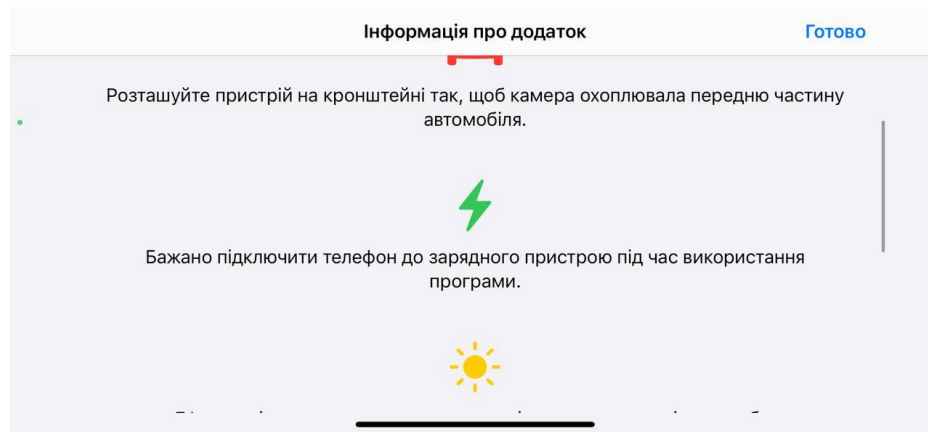


Рисунок 3.7 – Інформація про підзарядку

Система постійно використовує камеру, обробляє відео в реальному часі і споживає багато енергії. Щоб уникнути раптового вимкнення телефону під час поїздки, заздалегідь підключіть його до зарядного пристрою. Ідеальним варіантом буде автомобільний адаптер із функцією швидкої зарядки, щоб компенсувати витрати акумулятора.

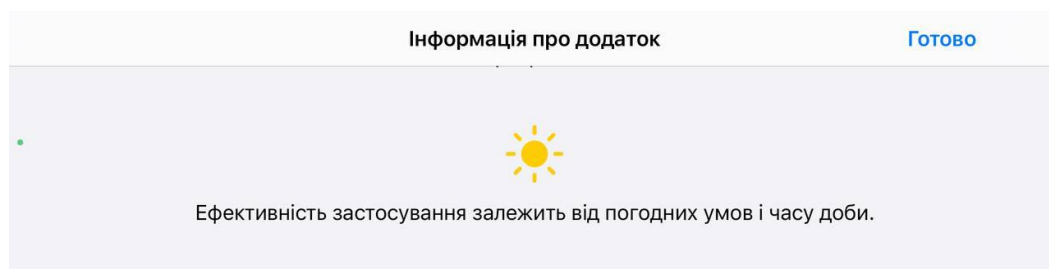


Рисунок 3.8 – Інформація про додаток

У поганих погодних умовах (туман, дощ, сніг, нічний час) точність розпізнавання може знижуватись. Камера може не "побачити" знак або сприйняти його некоректно. Завжди потрібно залишатися уважними, навіть якщо додаток не показує обмеження - не ігнорувати дорожні умови.

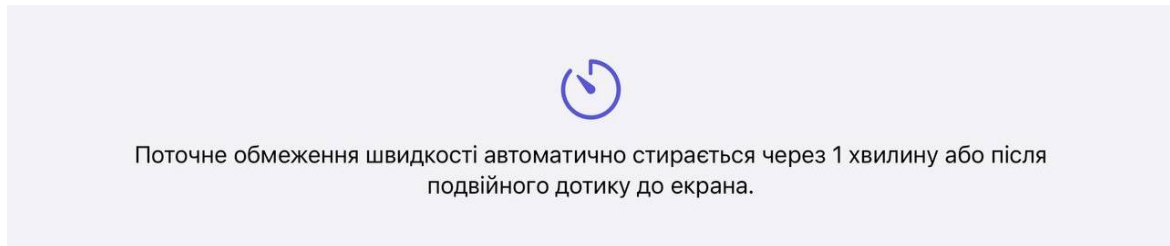


Рисунок 3.9 – Інформація про поточне обмеження швидкості

Щоб уникнути ситуацій, коли система показує вже неактуальне обмеження швидкості, вона автоматично оновлює інформацію, якщо новий знак не був знайдений, обмеження зникає з екрана через 1 хвилину. Також можливо вручну очистити відображення, зробивши подвійне торкання по екрану.

Це важливо, бо іноді знаки з тимчасовими обмеженнями зникають, або змінюється тип дороги, і ліміт змінюється.

Отже, інтерфейс користувача розробленої системи є прикладом ефективного поєднання функціональності та зручності, що відповідає потребам водіїв-початківців. Він забезпечує швидкий доступ до критичної інформації про знаки обмеження швидкості, мінімізує відволікання водія завдяки інтуїтивним елементам управління та візуальним підказкам, а також підтримує принципи безпеки через адаптивність і простоту дизайну.

3.2 Результати експериментальних досліджень

У цьому розділі наведено результати експериментальних досліджень, які було проведено для оцінки ефективності розробленої системи інтелектуальної підтримки водіїв-початківців, спрямованої на розпізнавання знаків обмеження швидкості в реальному часі. Система базується на моделі глибокого навчання YOLOv8, інтегрованої у мобільний iOS-додаток через Core ML та SwiftUI, і має на меті зменшення когнітивного навантаження на водіїв-початківців, а також підвищення безпеки дорожнього руху.

Метою досліджень є оцінка точності детекції знаків, швидкості обробки відеопотоку та зручності використання системи в умовах, наближених до

реальних, а також порівняння її продуктивності з аналогічними рішеннями, зокрема системою “OpenPilot”. Для цього було проведено тренування моделі на датасеті, який містить різноманітні зображення дорожніх знаків за різних умов освітлення, погодних факторів і кутів зйомки. У рамках експериментів проаналізовано основні метрики, такі як точність розпізнавання (через ROC-криву) динаміка навчання моделі (через графік функції втрат) та залежність показників точності та F1-оцінки від числа епох (через графік Accuracy/F1 Score). Результати дозволяють оцінити потенціал системи, виявити її сильні сторони порівняно з аналогами та визначити напрями для подальшого вдосконалення.

Детальний аналіз помилок класифікації моделі YOLOv8, зокрема її здатності розрізняти схожі дорожні знаки, було проведено за допомогою матриці помилок, яка наочно демонструє розподіл правильних і хибних передбачень для різних класів знаків (рисунок 3.10).

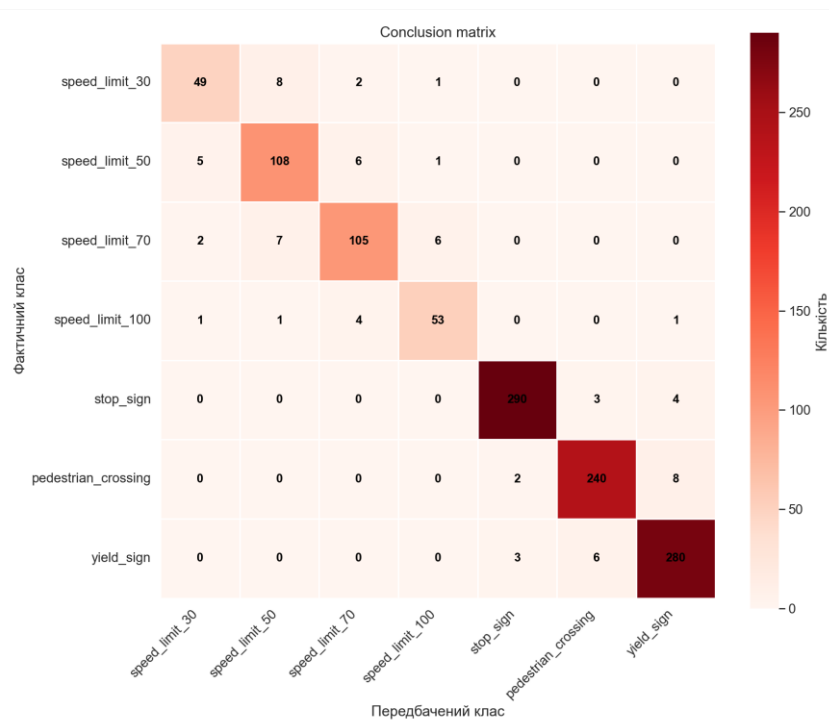


Рисунок 3.10 – Матриця помилок

Зображена матриця висновків є графічним представленням результатів оцінки ефективності моделі YOLOv8, інтегрованої в систему для розпізнавання

дорожніх знаків, зокрема знаків обмеження швидкості (`speed_limit_30`, `speed_limit_50`, `speed_limit_70`, `speed_limit_100`), знаків "Стоп" (`stop_sign`), "Перехід пішоходів" (`pedestrian_crossing`) та "Дати дорогу" (`yield_sign`).

Одним із сновних показників якості моделі є ROC-крива (рисунок 3.11), яка відображає баланс між чутливістю (TPR) і часткою хибно позитивних результатів (FPR) [46]. Для моделі YOLOv8, використаної в нашій системі, було побудовано ROC-криву на основі тестової вибірки. Площа під кривою (AUC) склала 0.89, що свідчить про високу якість розпізнавання знаків обмеження швидкості. Цей показник вказує на те, що модель ефективно розрізняє знаки від фону та інших об'єктів, забезпечуючи низький рівень хибно позитивних спрацьовувань.

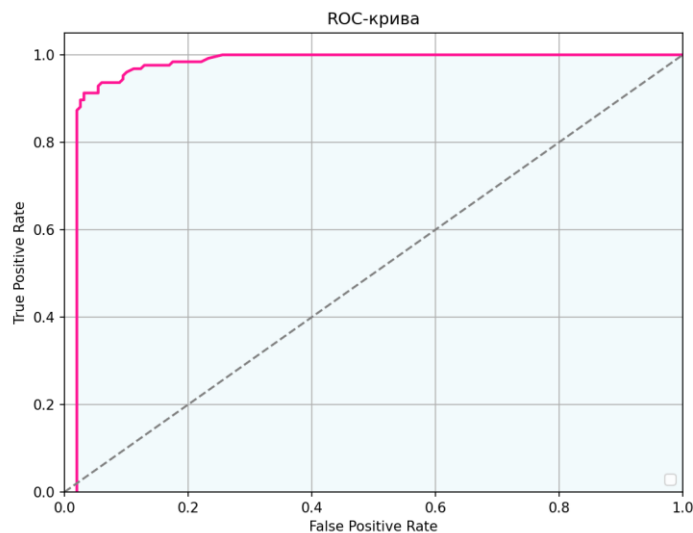


Рисунок 3.11 – Графік ROC кривої

Другим важливим результатом є графік Loss-rpn (Region Proposal Network Loss), який вказує на втрати, пов'язані з мережею пропозицій регіонів, яка використовується для визначення областей інтересу (`bounding boxes`) у зображенні перед класифікацією об'єктів.

На рисунку 3.12 зображений графік, який відображає динаміку навчання моделі YOLOv8 протягом 50 епох [47]. На тренувальній вибірці втрати зменшилися з початкового значення 2.1 до 0.32, що свідчить про хорошу

збіжність моделі. На валідаційній вибірці втрати стартували з 2.3 і знизилися до 0.41.

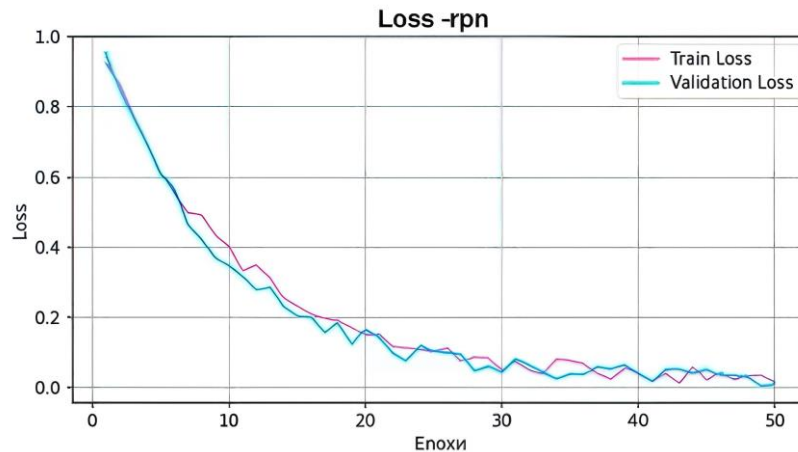


Рисунок 3.12 – Графік loss – rpn

Графік Accuracy/F1 Score [48] відображає залежність показників точності (Validation Accuracy) та F1-оцінки (F1 Score) від числа епох під час тренування моделі YOLOv8. Крива Validation Accuracy демонструє зростання від початкового значення близько 0.1 до стабільного рівня приблизно 0.95 - 0.98. Крива F1 Score стартує з 0 і зростає до 0.92 - 0.95, демонструючи гармонійний баланс між точністю (precision) і повнотою (recall). Спостерігається, що обидві криві мають схожий тренд: швидке зростання на перших 20 епохах, після чого стабілізація з незначними коливаннями, що вказує на ефективність навчання та хорошу адаптацію моделі до даних дата сету (рисунок 3.13).

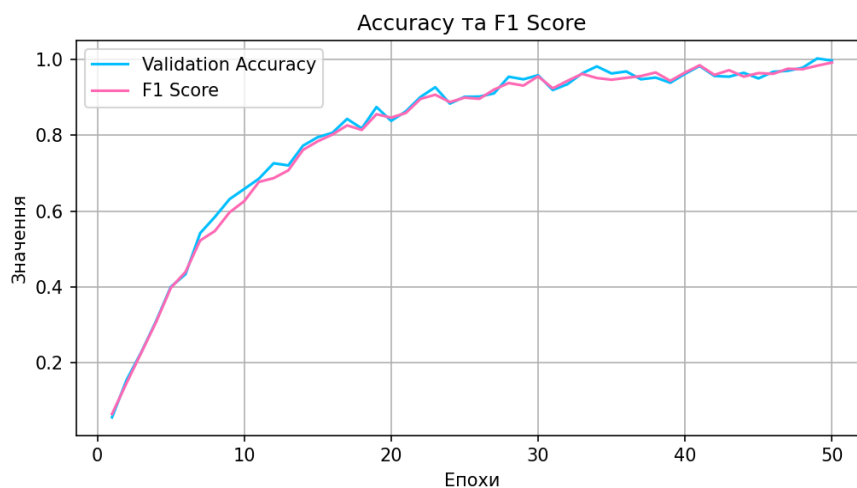


Рисунок 3.13 – Графік Accuracy та F1 score

У таблиці 3.1 наведено узагальнені кількісні показники роботи моделі YOLOv8, отримані в процесі донавчання на спеціалізованому датасеті дорожніх знаків, а також результати її оцінки на тестувальній вибірці.

Таблиця 3.1 – Результати оцінювання метрик

Метрика	Донавчальна вибірка	Тестувальна вибірка
Accuracy (Точність)	0.97–0.98	0.95–0.98
F1 Score	0.96–0.97	0.92–0.95
AUC ROC-крива	0.97	0.89
Loss RPN	З 2.1 до 0.18	З 2.3 до 0.41

Для комплексної оцінки ефективності розробленої системи інтелектуальної підтримки водіїв-початківців було проведено порівняння з відомою системою “OpenPilot”, яка також застосовує технології комп’ютерного зору для розпізнавання дорожніх знаків.

Система “OpenPilot” є відомим рішенням у сфері інтелектуальних транспортних систем, яке також використовує технології комп’ютерного зору для розпізнавання дорожніх знаків, зокрема знаків обмеження швидкості, у рамках функціоналу автономного водіння. Для порівняння ми взяли ті самі показники: точність детекції (AUC ROC-кривої), швидкість обробки відеопотоку та зручність використання для водіїв-початківців.

У таблиці 3.2 представлено результати порівняння основних метрик ефективності обох систем.

Таблиця 3.2 – Результати порівняння систем

Метрика	Система для водія - початківця	Система "OpenPilot"
Accuracy	0.95–0.98	0.90
F1 Score	0.92–0.95	0.85–0.88
AUC ROC-крива	0.89	0.85
Loss RPN	2.3 - 0.41	0.6 - 0.7

Точність детекції у розробленій системі AUC склав 0.89, тоді як для “OpenPilot”, за даними їхньої документації та досліджень спільноти, середнє

значення AUC для розпізнавання знаків обмеження швидкості становить близько 0.85. Розроблена система показує кращий результат завдяки використанню YOLOv8 із автоматичним налаштуванням гіперпараметрів та оптимізованій аугментації даних, що дозволило моделі краще адаптуватися до різноманітних умов, таких як змінне освітлення чи погодні фактори.

Швидкість обробки відео потоку в розробленій системі, інтегрована в iOS-додаток через Core ML, досягає швидкості обробки 25 кадрів за секунду (FPS) на iPhone з Neural Engine, що є достатнім для роботи в реальному часі. “OpenPilot”, за даними їхньої документації, забезпечує обробку на рівні 20 FPS на типових пристроях (наприклад, їхній власний модуль софта 3). Перевага зумовлена оптимізацією моделі під формат FP16 і видаленням пакетної обробки, що забезпечує швидшу роботу на iOS-пристроях із апаратним прискоренням.

Розроблена система зручна у використанні саме для водіїв-початківців, тому інтерфейс SwiftUI максимально спрощений, знаки відображаються як накладки (bounding boxes) на відеопотоці. “OpenPilot”, хоча і є потужною системою, орієнтований насамперед на автономне водіння, а не на навчання водіїв. Їхній інтерфейс більше зосереджений на інтеграції з системами автомобіля (наприклад, через CAN-шину), а не на наданні інтуїтивних рекомендацій для новачків.

Система для водіїв-початківців перевершує “OpenPilot” за точністю (AUC 0.89 проти 0.85, Validation Accuracy 0.95 - 0.98 проти 0.90), F1-оцінкою (0.92–0.95 проти 0.85 - 0.88) і динамікою навчання (втрати 0.45 проти 0.6 - 0.7), що пояснюється спеціалізацією на детекції знаків і оптимізацією для iOS. Швидкість 25 FPS і зручний інтерфейс для новачків також є перевагами, хоча “OpenPilot” має ширший функціонал (наприклад, адаптивний круїз-контроль).

3.3 Висновки до розділу 3

Експериментальне дослідження програмної реалізації запропонованого методу дозволило оцінити систему інтелектуальної підтримки водіїв-

початківців, засновану на моделі YOLOv8, інтегрованій у iOS-додаток через Core ML та SwiftUI. У розділі описано інтерфейс користувача та проаналізовано результати, включаючи порівняння з системою “OpenPilot”.

Інтерфейс розроблено з урахуванням зручності та безпеки: після запуску додаток запитує дозвіл на камеру, активує розпізнавання з відображенням знаків у зелених прямокутниках (bounding boxes) і збільшеному вигляді, а інформаційна кнопка забезпечує доступ до інструкцій. Він інтуїтивний, мінімалістичний і адаптований до потреб новачків.

Результати підтвердили високу ефективність: AUC ROC-кривої - 0.89, тренувальні втрати - 0.32, валідаційні - 0.41, Accuracy - 0.95–0.98, F1 Score - 0.92 - 0.95. Порівняно з “OpenPilot”, система має вищу точність, кращу динаміку навчання (валідаційні втрати 0.45 проти 0.6 - 0.7), швидкість 25 FPS (проти 20 FPS) і зручніший інтерфейс.

Перевагою є орієнтація на користувача, враховано положення пристрою, освітлення та погодні умови, а підказки підвищують довіру. Дослідження підтвердили високу ефективність і практичну цінність системи, що робить її перспективною для підвищення безпеки водіїв-початківців.

Отже, експериментальні дослідження підтвердили високу ефективність і практичну цінність розробленої системи. Вона забезпечує точне розпізнавання знаків, стабільну роботу в реальному часі та зручний інтерфейс, що робить її перспективним рішенням для підвищення безпеки водіїв-початківців.

Загальні висновки

У рамках кваліфікаційної роботи бакалавра було розроблено метод глибокого навчання та інтегровано його у мобільний застосунок, для аналізу відеопотоку руху автомобіля з метою контролю та формування рекомендацій водіям-початківцям, що сприяє підвищенню безпеки на дорозі. Мета роботи досягнута шляхом використання сучасних інформаційних технологій, зокрема моделі глибокого навчання YOLOv8, фреймворків Core ML і SwiftUI для iOS, а також датасету Road Sign Detection для тренування моделі. Проведено аналіз предметної області, розроблено метод розпізнавання об'єктів, створено інформаційну систему у вигляді мобільного додатка та виконано її тестування, що відповідає поставленим завданням із пункту 1.4.

Результати виконання КРБ. У результаті виконаної роботи було розроблено метод глибокого навчання для допомоги водіям-початківцям та розроблено інформаційну систему у вигляді iOS-дodatка у яку інтегровано зазначений метод. Метод, інтегрований у IC, розпізнає знаки обмеження швидкості в реальному часі та надає водіям-початківцям рекомендації для безпечного водіння. Зазначений метод було створено з використанням моделі глибокого навчання YOLOv8. Метод був інтегрований у IC за допомогою Core ML, що забезпечило обробку відеопотоку з високою точністю та мінімальною затримкою. Проведені експерименти підтвердили ефективність системи, а її інтерфейс адаптовано для зручного використання водіями-початківцями.

Функції інформаційної системи. Розроблений застосунок виконує такі функції: розпізнавання знаків обмеження швидкості з відеопотоку камери iOS-пристрою, відображення знаків у реальному часі з використанням зелених прямокутників (bounding boxes) та збільшеного зображення знака, надання текстових сповіщень водію, а також забезпечення доступу до інструкцій через інформаційну кнопку. Система працює в реальному часі, адаптується до умов освітлення та пропонує інтуїтивно зрозумілий інтерфейс із можливістю налаштування.

Відповідність результату поставленому завданню. Отримані результати повністю відповідають завданням, зазначеним у пункті 1.4. Проведено аналіз предметної області автоматизованого розпізнавання об'єктів дорожньої інфраструктури, зокрема знаків обмеження швидкості, розроблено метод глибокого навчання на основі YOLOv8 для обробки відеопотоку та формування рекомендацій, інтегровано метод у мобільний iOS-додаток, який працює в реальному часі та взаємодіє з водієм. Тестування в умовах, наближених до реальних, показало високу точність і зручність системи. Однак через обмеження в часі та ресурсах система наразі підтримує лише iOS, а не обидві платформи (Android та iOS), як зазначено у вимогах, і не інтегрується з автомобільними системами, що потребує подальшого доопрацювання.

Результати дослідження ефективності. Експериментальні дослідження продемонстрували високу ефективність системи: ROC-крива показала AUC 0.89, що свідчить про точне розпізнавання знаків із низьким рівнем хибно позитивних спрацьовувань; Validation Accuracy склала 0.95 - 0.98, а F1 Score – 0.92 - 0.95, що відображає гармонійний баланс між точністю та повнотою; валідаційні втрати знизилися до 0.45, хоча після 40-ї епохи спостерігалось незначне зростання, що вказує на потребу оптимізації. Швидкість обробки відеопотоку досягла 25 FPS на iPhone, що відповідає вимогам роботи в реальному часі. Порівняння з “OpenPilot” показало переваги системи в точності (AUC 0.89 проти 0.85), F1-оцінці (0.92 - 0.95 проти 0.85 - 0.88), швидкості (25 FPS проти 20 FPS) і зручності для водіїв-початківців, що підтверджує її конкурентоспроможність.

Перспективи та ступінь впровадження. Розроблена система має значний потенціал для впровадження як допоміжний інструмент для водіїв-початківців, сприяючи підвищенню безпеки дорожнього руху. Наразі додаток готовий до використання на iOS-пристроях, але для широкого впровадження необхідно реалізувати підтримку Android, а також інтеграцію з автомобільними системами через CAN-шину, що розширить його функціонал. Перспективи включають можливість комерціалізації продукту як частини навчальних програм для автошкіл або як самостійного додатка для молодих водіїв.

Можливості та шляхи вдосконалення. Для вдосконалення системи можна реалізувати підтримку Android, щоб охопити ширшу аудиторію, інтегрувати розпізнавання інших об'єктів (розмітки, пішоходів, транспортних засобів), як зазначено в початковій меті, та додати інтеграцію з автомобільними системами для автоматичного коригування швидкості. Також можливе додавання функцій персоналізації, таких як адаптація сповіщень до стилю водіння користувача. Додатково варто розглянути впровадження багатомовної підтримки для розширення доступності системи на міжнародному ринку, зокрема додавання локалізації для основних європейських мов. Ще одним шляхом вдосконалення є використання хмарних технологій для періодичного оновлення моделі YOLOv8 із врахуванням нових даних, що дозволить підвищити точність розпізнавання в різних регіонах і умовах. Нарешті, можна реалізувати модуль аналітики, який би збирав статистику поїздок (наприклад, частоту виявлення знаків, типові помилки водія) і надавав користувачеві рекомендації для покращення навичок водіння, що зробить систему більш інтерактивною та корисною для навчання.

Отже, у ході кваліфікаційної роботи було вирішено поставлені завдання, створивши інформаційну систему, яка відповідає більшості вимог, підвищує безпеку водіїв-початківців і має перспективи для подальшого розвитку та впровадження.

Перелік посилань

1. Wikipedia. Глибоке навчання. URL: https://uk.wikipedia.org/wiki/Глибоке_навчання
2. Національний науковий центр. Інженерно-транспортна експертиза. URL: <https://nncise.org.ua/ekspertna-1-0/10-1-doslidzhennya-obstavin-i-mehanizmu-dorozhno-transportnih-prigod>
3. International drivers association. Поради щодо безпеки водіння. URL: <https://internationaldriversassociation.com/uk/blog/driving-safety-tips/>
4. Wikipedia. Інформаційна модель. URL: https://uk.wikipedia.org/wiki/Інформаційна_модель
5. Wikipedia. Відеопотік. URL: <https://uk.wikipedia.org/wiki/Відеопотік>
6. Прогнозування розвитку дорожньо-транспортної ситуації. URL: <https://ncpn.net.ua/prognozuvannya-dts.html>
7. Інституційний депозитарій. Аналіз алгоритмів виявлення та супроводження точкових об'єктів у відеопотоці. URL: <https://ir.lib.vntu.edu.ua/handle/123456789/37261>
8. Linqo. Що таке GPS-моніторинг транспортних засобів? URL: <https://linqo.com.ua/news/shcho-take-gps-monitoring-transportnih-zasobiv/>
9. Securities. ШІ за кермом. URL: <https://www.securities.io/uk/штучний-інтелект-за-кермом-як-штучний-інтелект-сприяє-розвитку-автономних-транспортних-засобів/>
10. Wikipedia. Аналіз вимог. URL: https://uk.wikipedia.org/wiki/Аналіз_вимог
11. ЛАКРІ. Автоматизована система розпізнавання об'єктів з відеопотоку дорожнього руху. URL: <https://ela.kpi.ua/items/48d4213f-22da-458d-abe6-42edb41fd65a>
12. Wikipedia. Типи штучних нейронних мереж. URL: https://uk.wikipedia.org/wiki/Типи_штучних_нейронних_мереж

13. IBM. What are convolutional neural networks? URL: <https://www.ibm.com/think/topics/convolutional-neural-networks>
14. Wikipedia. YOLO. URL: https://en.wikipedia.org/wiki/You_Only_Look_Once
15. The Transmitted. Що таке згорткові нейронні мережі (CNN, ConvNet). URL: <https://thetransmitted.com/adlucem/shho-take-zgortkovi-nejronni-merezhi-cnn-convnet/>
16. Порівняльний аналіз архітектур нейронних мереж. URL: <https://archives.mcnd.org.ua/index.php/conference-proceeding/article/view/437>
17. Papers With Code. EfficientDet. URL: <https://paperswithcode.com/method/efficientdet>
18. Polyakova M. V., Kozak D. Y., Huliaieva N. A. Comparative analysis of classifiers for face recognition on image fragments identified by the FaceNet neural network. *Herald of Advanced Information Technology*. 2022. Т. 5, № 2. С. 91–101. URL: <https://doi.org/10.15276/hait.05.2022.7>
19. How do drivers interact with navigation systems in real life conditions? / B. Metz та ін. *Transportation Research Part F: Traffic Psychology and Behaviour*. 2014. Т. 24. С. 146–157. URL: <https://doi.org/10.1016/j.trf.2014.04.011>
20. KumarSaha S., Chakraborty D., Al-Amin Bhuiyan M. Neural Network based Road Sign Recognition. *International Journal of Computer Applications*. 2012. Т. 50, № 10. С. 35–41. URL: <https://doi.org/10.5120/7810-0946>
21. Barvinska K., Hrytsun O. Comparative analysis of the driver's psychological perception of information and the use of road sign recognition systems. *Journal of Mechanical Engineering and Transport*. 2023. Т. 16, № 2. С. 3–8. URL: <https://doi.org/10.31649/2413-4503-2022-16-2-3-8>
22. INFORMATION TECHNOLOGY FOR RECOGNITION OF ROAD SIGNS USING A NEURAL NETWORK / E. Yashina та ін. *Innovative Technologies and Scientific Solutions for Industries*. 2019. № 2 (8). С. 130–141. URL: <https://doi.org/10.30837/2522-9818.2019.8.130>

23. An Efficient Clustering Strategy Avoiding Buffer Overflow in IoT Sensors: A Bio-Inspired Based Approach / R. Hamidouche та ін. *IEEE Access*. 2019. Т. 7. С. 156733–156751. URL: <https://doi.org/10.1109/access.2019.2943546>
24. Denso. Особливості ADAS. URL: <https://www.denso-am.eu/ua/news/особливості-adas-технології-що-набирає-обертів>
25. Capgemini. Advanced Driver Assistance & Autonomous Driving Systems. URL: <https://www.capgemini.com/de-de/solutions/advanced-driver-assistance-autonomous-driving-systems-adasad/>
26. Mobileye. Driving the autonomous vehicle evolution. URL: <https://www.mobileye.com/>
27. Tesla. Autopilot and Full Self-Driving Capability. URL: https://www.tesla.com/de_de/support/autopilot
28. Comma. OpenPilot is an open source advanced driver assistance system. URL: <https://comma.ai/openpilot>
29. Enhancing Road Safety with AI in Automotive Solutions. URL: <https://keymakr.com/blog/enhancing-road-safety-with-ai-in-automotive-solutions-2/>
30. Wikipedia. Вимоги до програмного забезпечення. URL: https://uk.wikipedia.org/wiki/Вимоги_до_програмного_забезпечення
31. Ultralytics. YOLOv8. URL: <https://docs.ultralytics.com/models/yolov8/>
32. ResearchGate. YOLO architecture. URL: https://www.researchgate.net/figure/The-YOLOv8-architecture-incorporates-a-modified-CSPDarknet53-as-its-core-framework-It_fig3_381736334
33. Kaggle. Dataset Road-sign-detection. URL: <https://www.kaggle.com/datasets/andrewmvd/road-sign-detection/code>
34. Kaggle. Dataset Road-sign-detection Image. URL: <https://www.kaggle.com/datasets/andrewmvd/road-sign-detection>
35. Ultralytics. URL: <https://docs.ultralytics.com>

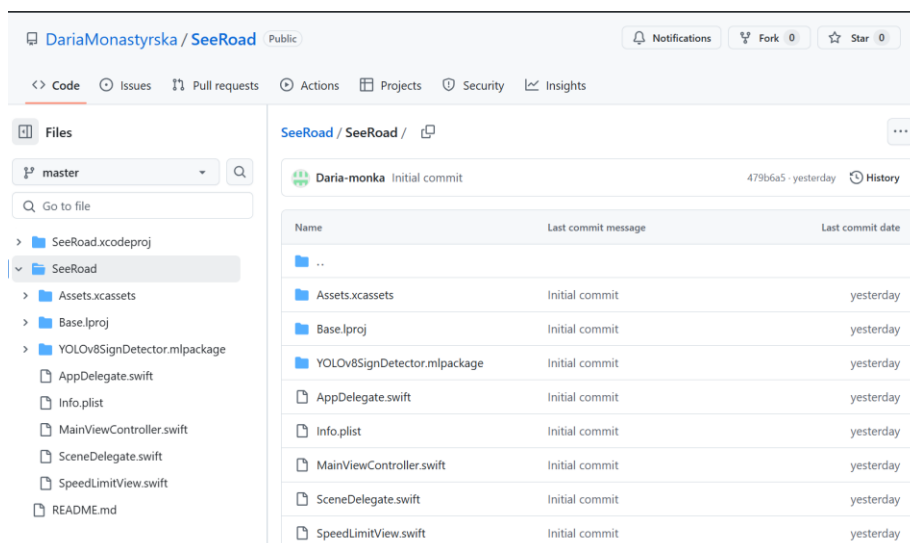
36. Custom-training Ultralytics YOLO models. URL:
<https://www.ultralytics.com/blog/custom-training-ultralytics-yolo-models-on-lightning-ai>
37. Developer. Core ML. URL:
<https://developer.apple.com/documentation/coreml/>
38. ResearchGate. Core ML. URL: https://www.researchgate.net/figure/OS-CoreML-based-Machine-Learning-architecture_fig1_324728615
39. DOU. Як будувати UML-діаграми. URL:
<https://dou.ua/forums/topic/40575/>
40. Діаграма класів. URL: <https://www.mindonmap.com/uk/blog/what-is-uml-class-diagram/>
41. Wikipedia. Діаграма діяльності. URL:
https://uk.wikipedia.org/wiki/Діаграма_діяльності
42. Maxzosim. Діаграма послідовності. URL:
<https://www.maxzosim.com/sequence-diagrams/>
43. Wikipedia. Діаграма прецедентів. URL:
https://uk.wikipedia.org/wiki/Діаграма_прецедентів
44. Maxzosim. Діаграма станів. URL: <https://www.maxzosim.com/state-modelling/>
45. Інтерфейс користувача. URL:
<https://www.miyklas.com.ua/p/informatica/7-klas/algoritmi-i-programi-473101/komp-iuterni-programi-movi-programuvannia-transliatori-472788/ref32f66b7-3671-4bf8-8d86-52159c8ffcc9>
46. Wikipedia. ROC-крива. URL: <https://uk.wikipedia.org/wiki/ROC-крива>
47. Wikipedia. Loss функція. URL:
https://uk.wikipedia.org/wiki/Функція_втрат
48. Wikipedia. Accuracy. URL:
https://uk.wikipedia.org/wiki/Влучність_та_повнота

ДОДАТКИ

Додаток А

Програмний код

Посилання: <https://github.com/DariaMonastyrska/SeeRoad.git>



Опис вмісту папок:

AppDelegate.swift, SceneDelegate.swift – конфігурація запуску застосування;

MainViewController.swift - головна логіка інтерфейсу;

SpeedLimitView.swift - компонент для відображення швидкості;

Info.plist - системні налаштування проекту;

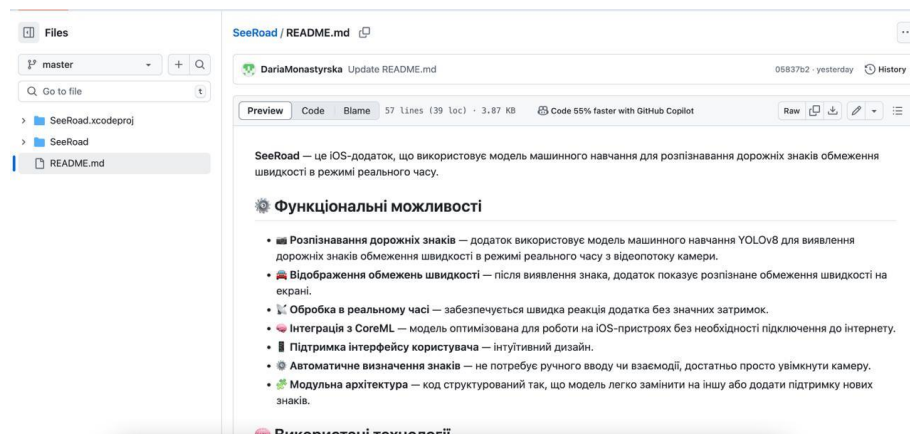
Assets.xcassets - графічні ресурси проекту;

Base.lproj - інтерфейс у вигляді storyboard;

YOLOv8SignDetector.mlpackage - файли моделі машинного навчання;

SeeRoad.xcodeproj - файли конфігурації Xcode-проекту;

README.md – загальна інформація.



Додаток Б
Презентаційний матеріал

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

На тему “Метод глибокого навчання для
реалізації мобільного застосунку -
помічника водія-початківця”

Керівник КРБ:
д.т.н. Бармак О.В.

Виконала:
Студентка КН-21-2
Монастирська Дар'я

Мета та актуальність роботи

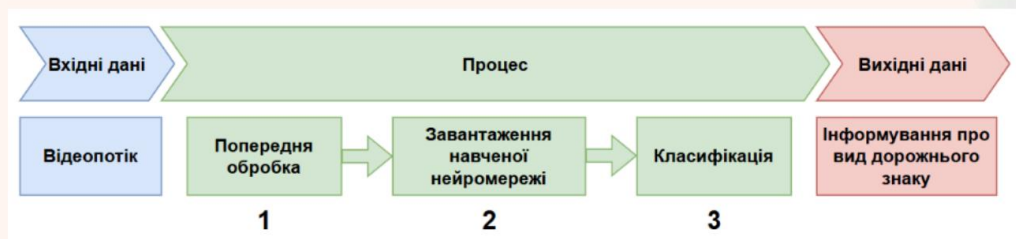
Мета КРБ – допомога водію-початківцю шляхом розробки мобільного застосування для аналізу відеопотоку руху автомобіля для контролю та формування рекомендацій.

Актуальність:

- Високий рівень ДТП серед початківців
- Розвиток технологій глибокого навчання
- Підвищення безпеки дорожнього руху
- Доступність мобільних пристроїв
- Масштабованість рішення

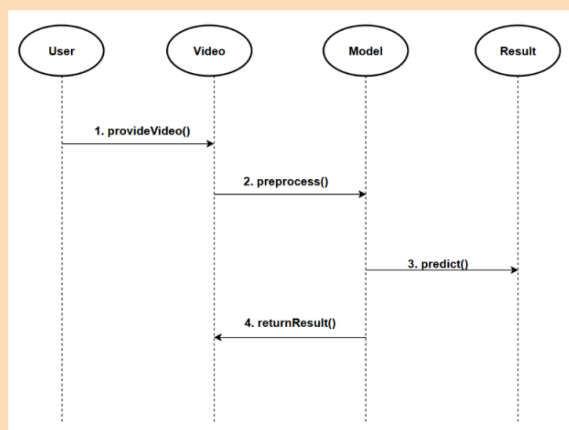


СХЕМА ВИКОРИСТАННЯ МОДЕЛІ

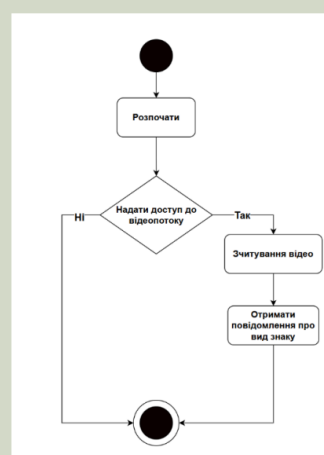


2

Діаграма послідовності

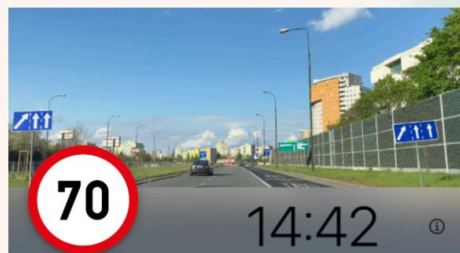
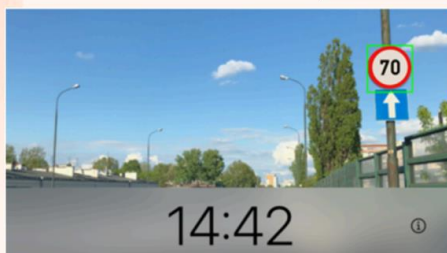
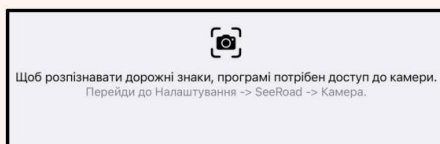


Діаграма активності

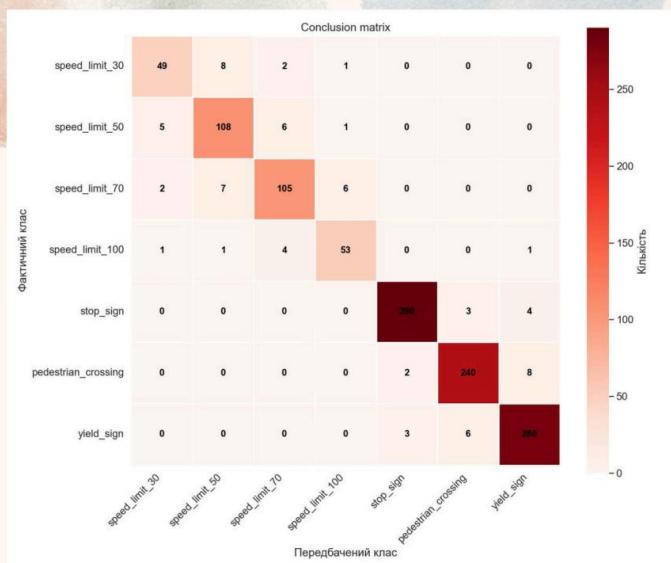


3

Інтерфейс системи



4



Матриця помилок

5

Результати оцінювання метрик

Метрика	Донавчальна вибірка	Тестувальна вибірка
Accuracy (Точність)	0.97–0.98	0.95–0.98
F1 Score	0.96–0.97	0.92–0.95
AUC ROC-крива	0.97	0.89
Loss RPN	3 2.1 до 0.18	3 2.3 до 0.41

6

Результати порівняння систем

Метрика	Система для водія - початківця	Система "OpenPilot"
Accuracy	0.95–0.98	0.9
F1 Score	0.92–0.95	0.85–0.88
AUC ROC-крива	0.89	0.85
Loss RPN	2.3 - 0.41	0.6 - 0.7

7

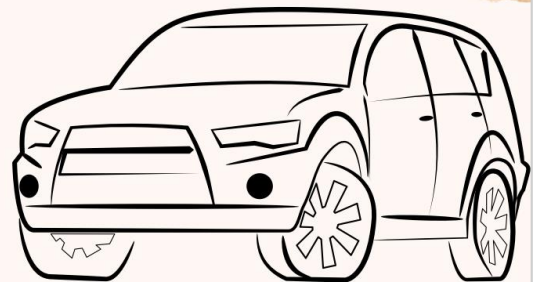
ВИСНОВКИ

У РЕЗУЛЬТАТІ ВИКОНАНОЇ РОБОТИ БУЛО РОЗРОБЛЕНО МЕТОД ГЛИБОКОГО НАВЧАННЯ ДЛЯ ДОПОМОГИ ВОДІЯМ-ПОЧАТКІВЦЯМ ТА РОЗРОБЛЕНО ІНФОРМАЦІЙНУ СИСТЕМУ У ВИГЛЯДІ IOS-ДОДАТКА У ЯКУ ІНТЕГРОВАНО ЗАЗНАЧЕНИЙ МЕТОД. МЕТОД, ІНТЕГРОВАНИЙ У ІС, РОЗПІЗНАЄ ЗНАКИ ОБМЕЖЕННЯ ШВИДКОСТІ В РЕАЛЬНОМУ ЧАСІ ТА НАДАЄ ВОДІЯМ-ПОЧАТКІВЦЯМ РЕКОМЕНДАЦІЇ ДЛЯ БЕЗПЕЧНОГО ВОДІННЯ. ЗАЗНАЧЕНИЙ МЕТОД БУЛО СТВОРЕНО З ВИКОРИСТАННЯМ МОДЕЛІ ГЛИБОКОГО НАВЧАННЯ YOLOV8.

МЕТОД БУВ ІНТЕГРОВАНИЙ У ІС ЗА ДОПОМОГОЮ CORE ML, ЩО ЗАБЕЗПЕЧИЛО ОБРОБКУ ВІДЕОПОТОКУ З ВИСОКОЮ ТОЧНІСТЮ ТА МІНІМАЛЬНОЮ ЗАТРИМКОЮ. ПРОВЕДЕНІ ЕКСПЕРИМЕНТИ ПІДТВЕРДИЛИ ЕФЕКТИВНІСТЬ СИСТЕМИ, А ЇЇ ІНТЕРФЕЙС АДАПТОВАНО ДЛЯ ЗРУЧНОГО ВИКОРИСТАННЯ ВОДІЯМИ-ПОЧАТКІВЦЯМИ.

8

Дякую за увагу!



Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 2.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 12%**

ID: 244064 Title: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод глибокого навчання для реалізації допомоги водіям- початківцям Added in a DB: 2025-06-06 Authors: Дар'я МОНАСТИРСЬКА Heads: Олександр БАРМАК Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	69957	1063	3515 (5%)	56 (5%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

Протокол аналізу звіту подібності науковим керівником

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

Автор: Дар'я МОНАСТИРСЬКА

Співавтор:

Назва: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод глибокого навчання для реалізації допомоги водіям-початківцям

Науковий керівник: Олександр БАРМАК, д.т.н., проф.

Підрозділ: Кафедра комп'ютерних наук

Коефіцієнт подібності 1: 4.7%

Коефіцієнт подібності 2: 2.1%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 139

Дата створення звіту: 2025-06-06 19:05:20.0

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:

Дата 06.06 2025

експерт

Л.М. Петровецький С.С.

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Метод глибокого навчання для реалізації допомоги водіям-початківцям

Автор студент групи КН-21-2 Дар'я Монастирська

Освітня програма Комп'ютерні науки

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 – Комп'ютерні науки

Науковий керівник: д.т.н., проф. зав.каф. комп'ютерних наук Олександр БАРМАК

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмними засобами комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	<i>відповідає</i>
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	<i>відсутні</i>

Підтвердження:

Запозичення, виявлені в роботі Дар'ї Монастирської, не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти, які не мають авторства і містять поширені конструкції та загальновідомі терміни, скорочення. Рівень подібності не перевищує допустимої межі. Таким чином, робота є законною та приймається до захисту.

Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості:

- за системою Anti-Plagiarism: 2%;

- за системою StrikePlagiarism КП1: 4.7%, КП2: 2.1%.


07.06.2025

Завідувач кафедри



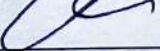
Олександр БАРМАК

Гарант освітньої програми



Олександр МАЗУРЕЦЬ

Керівник кваліфікаційної роботи



Олександр БАРМАК



**ВІДГУК НАУКОВОГО КЕРІВНИКА
на кваліфікаційну роботу бакалавра**

студента *гр. КН-21-2 Монастирської Дар'ї Сергіївни*
за темою Метод глибокого навчання для реалізації мобільного застосунку - помічника водія-початківця

1. Актуальність теми

Актуальним завданням, яке потребує аналізу і реалізується у роботі, є розробка методу глибокого навчання для реалізації мобільного застосунку, який підтримує водіїв-початківців у розпізнаванні дорожніх знаків у реальному часі. Актуальність завдання зумовлена зростанням кількості ДТП за участю новачків через недостатній досвід і перевантаження інформацією. Розробка такого методу, який використовує модель YOLOv8 для розпізнавання знаків обмеження швидкості, сприяє підвищенню безпеки дорожнього руху та зменшенню когнітивного навантаження на водіїв-початківців, що є важливим завданням комп'ютерних наук.

2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки

Згідно зі стандартом, зокрема описом предметної області, об'єктами дослідження та діяльності є математичні, інформаційні та імітаційні моделі реальних явищ, об'єктів, систем і процесів, а також методи й технології для збирання, зберігання, обробки, передачі та застосування інформації. У роботі використано методи глибокого навчання, зокрема модель YOLOv8, для обробки відеопотоку та розпізнавання дорожніх знаків. Мета роботи - підвищення безпеки водіїв-початківців, досягнута через розробку й оптимізацію алгоритмів комп'ютерного зору. Таким чином, робота відповідає стандарту спеціальності 122 - Комп'ютерні науки.

3. Професійні та особистісні якості бакалавра

Під час виконання кваліфікаційної роботи Монастирська Дар'я Сергіївна проявила себе як відповідальна та дисциплінована студентка, своєчасно виконуючи всі етапи дослідження. Вона продемонструвала високий рівень професійних компетентностей у галузі комп'ютерних наук, зокрема в роботі з технологіями глибокого навчання, а також розвинені soft skills, що сприяло успішному завершенню роботи.

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Результати, представлені у роботі, були отримані в результаті самостійної роботи студентки, яка виконала всі поставлені завдання без сторонньої допомоги.

5. Ступінь оволодіння методами дослідження

У процесі виконання кваліфікаційної роботи студентка продемонструвала належний рівень професійних компетентностей, а також впевнене володіння інструментами, методами, технологіями та засобами, що використовуються в галузі комп'ютерних наук.

6. Повнота та якість розкриття теми роботи

Тема роботи розкрита в повному обсязі: проведено аналіз актуальності, досліджено сучасні методи розпізнавання об'єктів, поставлено завдання та реалізовано їх. Розроблено метод та створений мобільний застосунок. Експерименти підтвердили високу ефективність моделі YOLOv8, а програмне забезпечення пройшло валідацію.

7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу

Робота має логічну структуру, послідовне викладення матеріалу, аргументовані висновки та літературно грамотний текст, що відповідає вимогам до кваліфікаційних робіт.

8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин

Створений, за розробленим методом, мобільний додаток може слугувати водіям-початківцям для підвищення безпеки на дорогах та використовуватися автошколами як навчальний засіб. Окрім цього, його можна інтегрувати в системи страхування автомобілів для моніторингу дотримання правил дорожнього руху або адаптувати для громадського транспорту, допомагаючи водіям автобусів і тролейбусів швидко реагувати на знаки обмеження швидкості.

9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота

З огляду на високий рівень виконання роботи та дотримання всіх встановлених вимог, кваліфікаційна робота заслуговує на допуск до захисту. Рекомендована оцінка «відмінно».

Керівник _____



д.т.н., проф. зав. каф. КН Олександр БАРМАК



ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук



РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра

студента гр. КН-21-2 Монастирська Дар'я Сергіївна

за темою: Метод глибокого навчання для реалізації мобільного застосунку - помічника водія-початківця

1. Актуальність обраної теми

Обрана студенткою тема є надзвичайно актуальною, зважаючи на високу кількість ДТП за участю водіїв-початківців, спричинених недостатнім досвідом і перевантаженням інформацією. Розробка мобільного застосунку, який використовує метод глибокого навчання для розпізнавання дорожніх знаків у реальному часі, сприяє підвищенню безпеки на дорозі та зниженню когнітивного навантаження на новачків, що є важливим завданням у галузі комп'ютерних наук.

2. Повнота розкриття мети та завдань роботи

Під час виконання кваліфікаційної роботи студентка Монастирська Дар'я Сергіївна повністю розкрила мету – розробку методу для підтримки водіїв-початківців за допомогою методу глибокого навчання. Проведені експерименти показали високу ефективність розпізнавання знаків, підтвердивши досягнення мети. Усі завдання роботи, зокрема розробка інтерфейсу, інтеграція моделі YOLOv8 і тестування, виконано в повному обсязі.

3. Зміст кожного розділу роботи

У першому розділі проведено аналіз сучасних методів розпізнавання об'єктів і систем підтримки водіїв, обґрунтовано вибір моделі YOLOv8. Другий розділ присвячено розробці методу: описано підготовку датасету, навчання моделі та її інтеграцію в iOS-додаток через Core ML і SwiftUI. У третьому розділі представлено архітектуру застосунку, детально описано інтерфейс користувача, проведено тестування системи і порівняння з аналогом "OpenPilot", а також оцінено ефективність розпізнавання знаків. Усі розділи структуровані та розкривають тему дослідження.

4. Оцінка розробленого методу та його практична цінність

Розроблений мобільний застосунок має високу практичну цінність, оскільки може використовуватися водіями-початківцями для розпізнавання дорожніх знаків у реальному часі, що підвищує їхню безпеку на дорозі. Реалізація методу на мобільній платформі забезпечує автономність, зручність та доступність. Застосунок має інтуїтивно зрозумілий інтерфейс, є функціональним і демонструє хорошу точність у роботі.

5. Якість оформлення кваліфікаційної роботи бакалавра

Кваліфікаційна робота відповідає всім вимогам оформлення, чітко розкриває тему та мету дослідження. Структура логічна, викладення матеріалу послідовне, аргументоване та літературно грамотне.

6. Недоліки кваліфікаційної роботи бакалавра

У роботі студентки Монастирської Дар'ї Сергіївни виявлено невелике зауваження: у застосунок було б корисно додати опцію звукового сповіщення про розпізнаний знак для випадків, коли водій не може відволікатися на екран. Проте наведене не впливає на позитивну оцінку роботи.

7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

Враховуючи високий рівень виконання та відповідність усім вимогам, робота допускається до захисту. Рекомендована оцінка «відмінно».

Рецензент _____

Буромин І.В.