

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук  
(повна назва)

Кафедра Інформаційних управляючих систем  
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

рівень вищої освіти другий (магістерський)

Дослідження методів підтримки прийняття рішень з вибору  
дорожнього маршруту на основі розпізнавання відеозображень

(тема)

Виконав:

здобувач 2 року навчання,

групи ІУСТМ-24-1

Кравченко Денис Сергійович

(прізвище, ім'я, по батькові)

Спеціальність 122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Інформаційні управляючі  
системи та технології

(повна назва освітньої програми)

Керівник: проф. каф. ІУС Чалий С.Ф.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ІУС

(підпис)

Петров К.Е.

(прізвище, ініціали)

2025 р.

## Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Комп'ютерних наук \_\_\_\_\_

Кафедра \_\_\_\_\_ Інформаційних управляючих систем \_\_\_\_\_

Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_

Спеціальність \_\_\_\_\_ 122 Комп'ютерні науки \_\_\_\_\_  
(код і повна назва)Тип програми \_\_\_\_\_ освітньо-професійна \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)Освітня програма \_\_\_\_\_ Інформаційні управляючі системи та технології \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

“ 24 ” листопада 2025 р.

**ЗАВДАННЯ****НА КВАЛІФІКАЦІЙНУ РОБОТУ**здобувачеві \_\_\_\_\_ Кравченку Денису Сергійовичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи “Дослідження методів підтримки прийняття рішень з вибору дорожнього маршруту на основі розпізнавання відеозображень” \_\_\_\_\_

затверджена наказом по університету від “ 24 ” листопада 2025 р. № 1055Ст \_\_\_\_\_

2. Термін подання здобувачем роботи до екзаменаційної комісії “ ” січня 2025 р. \_\_\_\_\_

3. Вихідні дані до роботи науково-методична та науково-технічна література, матеріали конференцій, дані інтернет-мережі, модель розпізнавання об'єктів YOLOv11, базовий набір даних із зразками одягу, алгоритми та методи прийняття рішень, мова Python та відповідні бібліотеки, фреймворки машинного навчання, апаратна платформа для розгортання та тестування. \_\_\_\_\_

4. Перелік питань, що потрібно опрацювати у роботі \_\_\_\_\_

1) Проаналізувати методи комп'ютерного зору (YOLO) та існуючі алгоритми рекомендацій для навігаційних систем. \_\_\_\_\_

2) Розробити концептуальну модель системи підтримки рішень та удосконалити метод побудови рекомендацій на основі YOLOv11 і профілювання користувачів. \_\_\_\_\_

3) Створити інформаційну технологію реалізації системи персоналізованих рекомендацій. \_\_\_\_\_

4) Провести експериментальні дослідження, оцінити точність виявлення об'єктів та ефективність сформованих дорожніх рекомендацій. \_\_\_\_\_

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	24.11.2025 - 25.11.2025	Виконано
2	Аналіз завдання, підбір літератури	25.11.2025 - 26.11.2025	Виконано
3	Аналіз літератури з досліджуваної проблеми	26.11.2025 - 27.11.2025	Виконано
4	Аналіз технічних засобів	27.11.2025 - 28.11.2025	Виконано
5	Розробка методу	28.11.2025 - 02.12.2025	Виконано
6	Програмна реалізація	03.12.2025 - 08.12.2025	Виконано
7	Оформлення пояснювальної записки	08.12.2025 - 10.12.2025	Виконано
8	Підготовка презентації та доповіді	10.12.2025 - 11.12.2025	Виконано
9	Перевірка на плагіат	11.12.2025 - 11.12.2025	Виконано
10	Захист кваліфікаційної роботи	17.12.2025	

Дата видачі завдання 24 листопада 2025 р.

Здобувач   
(підпис)

Керівник роботи \_\_\_\_\_  
(підпис)

проф. каф. ІУС Чалий С.Ф.  
(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 111 с., 25 рис., 4 табл., 2 дод., 45 джерел.

ГЛИБОКЕ НАВЧАННЯ, КОМП'ЮТЕРНИЙ ЗІР, НЕЙРОННІ МЕРЕЖІ,  
РОЗПІЗНАВАННЯ ОБ'ЄКТІВ, РОЗПІЗНАВАННЯ МАРШРУТУ.

Об'єктом дослідження є процес підтримки прийняття рішень та формування рекомендацій для навігаційних систем на основі аналізу відеозображень дорожньої обстановки та моделювання переваг користувачів.

Предметом дослідження є методи підтримки прийняття рішень на основі розпізнавання та трекінгу об'єктів на відеозображеннях, алгоритми побудови персоналізованих рекомендацій, підходи до вирішення проблеми холодного старту та адаптації до змін переваг користувачів.

Метою роботи є створення та експериментальне дослідження інтелектуальної системи розпізнавання об'єктів, яка забезпечує високу швидкість виявлення (FPS) при збереженні високої точності (mAP) за допомогою оптимізованої архітектури YOLOv11.

За результатами роботи розроблені: метод підтримки прийняття рішень з вибору дорожнього маршруту на основі розпізнавання відеозображень, мобільний прототип застосунку для розпізнавання об'єктів. Сама система демонструє високі показники mAP (понад 90%) та забезпечує обробку в реальному часі (понад 60 FPS) на цільовій платформі.

## ABSTRACT

Master thesis 111 pages, 25 figures, 4 tables, 2 appendices, 45 sources.

DEEP LEARNING, COMPUTER VISION, NEURAL NETWORKS,  
OBJECT RECOGNITION, ROUTE RECOGNITION.

The object of the research is the process of supporting decision-making and generating recommendations for navigation systems based on the analysis of video images of the road environment and the modeling of user preferences.

The subject of the research includes methods for decision-making support based on object recognition and tracking in video images, algorithms for constructing personalized recommendations, as well as approaches to addressing the cold-start problem and adapting to changes in user preferences.

The aim of the work is to develop and experimentally evaluate an intelligent object recognition system that ensures high detection speed (FPS) while maintaining high accuracy (mAP) through an optimized YOLOv11 architecture.

As a result of the work, the following were developed: a decision-support method for road route selection based on video image recognition, and a mobile prototype application for object recognition. The system demonstrates high mAP performance (over 90%) and provides real-time processing (over 60 FPS) on the target platform.

## ЗМІСТ

	С.
Скорочення та умовні позначки.....	8
Вступ.....	10
1 Аналіз методів побудови навігаційних рекомендацій з візуальним моніторингом трафіку.....	16
1.1 Аналіз процесу формування маршрутних рекомендацій на основі відеоаналізу дорожньої обстановки.....	16
1.2 Огляд сучасних підходів до розпізнавання об'єктів та побудови персоналізованих рекомендацій.....	24
1.3 Аналіз методів побудови рекомендацій у режимі холодного старту з використанням візуальних даних.....	31
1.4 Постановка задачі дослідження.....	37
2 Розробка методу адаптивних рекомендацій з семантичною інтеграцією візуальних даних.....	42
2.1 Концептуальна модель багаторівневої системи з виявленням еволюції переваг користувачів.....	42
2.2 Удосконалений метод побудови рекомендацій з інтеграцією YOLOv11 та адаптивним профілюванням.....	49
3 Інформаційна технологія побудови навігаційних рекомендацій в умовах обмеженої апріорної інформації.....	57
3.1 Десятиступеневий процес формування персоналізованих рекомендацій маршрутів.....	57
3.2 Програмна реалізація на основі PyTorch, ONNX Runtime та мікросервісної архітектури.....	67
4 Експериментальна перевірка методу на датасетах bdd100k та власних записах.....	74
4.1 Реалізація прототипу системи з модульною архітектурою.....	74
4.2 Формування тестових наборів даних з різними сценаріями використання.....	77
4.3 Результати тестування.....	79
4.4 Порівняльний аналіз з базовими підходами: перевага на 4-21 відсоткових пунктів.....	82
Висновки.....	85
Перелік джерел посилання.....	89

Додаток А Графічний матеріал кваліфікаційної роботи.....	94
Додаток Б Код застосунку.....	106

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

- API – Application Programming Interface
- BDD100K – Berkeley DeepDrive 100K dataset
- COCO – Common Objects in Context
- CoreML – Core Machine Learning
- CPU – Central Processing Unit
- DeepLab – сімейство моделей семантичної сегментації
- DeepSORT – Deep Simple Online and Realtime Tracking
- FCN – Fully Convolutional Network
- FPS – Frames Per Second
- GPS – Global Positioning System
- GPU – Graphics Processing Unit
- HD – High Definition
- ImageNet – еталонний датасет зображень
- KITTI – Karlsruhe Institute of Technology and Toyota Technological Institute, публічний датасет з зображеннями дорожніх сцен
- LSTM – Long Short-Term Memory
- mAP – mean Average Precision
- MOT17 – еталонний датасет для багатоб'єктного трекінгу
- MOTA – Multiple Object Tracking Accuracy
- MOTP – Multiple Object Tracking Precision
- NCHW - batch , channels, depth, height, width, кількість зразків, канали, висота, ширина (формат тензора, що часто використовується в комп'ютерному зорі та глибокому навчанні)
- NMS – Non-Maximum Suppression
- ONNX – Open Neural Network Exchange
- OpenCV – Open Source Computer Vision Library
- ORB – Oriented FAST and Rotated BRIEF
- FAST – Features from Accelerated Segment Test

FairMOT – Fair Multi-Object Tracking  
BRIEF – Binary Robust Independent Elementary Features  
REST – Representational State Transfer  
ResNet – Residual Network  
RetinaNet – архітектура одноетапного детектора  
RGB – Red, Green, Blue (колірна модель)  
SHAP – SHapley Additive exPlanations  
SIFT – Scale-Invariant Feature Transform  
SOLO – Segmenting Objects by Locations  
SSD – Single Shot MultiBox Detector  
SURF – Speeded Up Robust Features  
SVM – Support Vector Machine  
TensorRT – Tensor Runtime  
U-Net – архітектура сегментації з U-подібною структурою  
VGG – Visual Geometry Group network  
YOLO – You Only Look Once  
YUV – Y (Luminance), U (Blue-difference Chrominance), V (Red-difference Chrominance)

## ВСТУП

Інтелектуальні транспортні системи є одним з найбільш динамічних напрямків розвитку сучасних міст, що стикаються з проблемами перевантаження дорожньої інфраструктури та зростаючими вимогами до якості транспортних послуг. Навігаційні системи перетворились з простих інструментів пошуку найкоротшого шляху на складні рекомендаційні платформи, що враховують множину факторів: поточний стан трафіку, прогнозовані затримки, дорожні події, погодні умови та індивідуальні переваги користувачів. Традиційні підходи до побудови навігаційних рекомендацій базуються переважно на історичних статистичних даних про завантаженість доріг та інформації від інтернет-сервісів моніторингу трафіку, що має суттєві обмеження у точності та актуальності, особливо при динамічних змінах дорожньої обстановки. Інтеграція технологій комп'ютерного зору у навігаційні системи відкриває принципово нові можливості для отримання актуальної інформації про стан доріг в режимі реального часу. Сучасні моделі глибокого навчання, зокрема архітектури сімейства You Only Look Once (YOLO), досягли рівня точності виявлення об'єктів та швидкості обробки, що дозволяє аналізувати відеопотоки від камер транспортних засобів або дорожньої інфраструктури з мінімальними затримками. Візуальний моніторинг дорожньої обстановки надає детальну інформацію про щільність транспортного потоку, швидкість руху, наявність перешкод, дорожніх робіт, аварійних ситуацій, що не завжди своєчасно відображається у традиційних джерелах даних.

Персоналізація навігаційних рекомендацій є критично важливою для задоволення різноманітних потреб користувачів. Різні категорії водіїв мають суттєво відмінні пріоритети при виборі маршруту: для когось критичною є мінімізація часу поїздки, інші віддають перевагу економії палива, треті цінують комфорт та зниження стресу від водіння, четверті обирають естетично

привабливі маршрути. Існуючі рекомендаційні системи недостатньо ефективно враховують ці індивідуальні переваги, особливо у ситуаціях холодного старту для нових користувачів без накопиченої історії переміщень. Крім того, переваги користувачів можуть еволюціонувати у часі внаслідок життєвих змін, зміни транспортного засобу, нових маршрутів роботи чи навчання, що вимагає адаптивних механізмів виявлення та пристосування до таких змін.

Окрему категорію користувачів становлять люди з порушеннями зору, для яких традиційні навігаційні системи є недостатньо доступними. Врахування їхніх потреб передбачає створення механізмів голосового супроводу, детального озвучення дорожньої обстановки та адаптованої подачі рекомендацій. Інтеграція системи комп'ютерного зору з модулем голосового опису подій може суттєво підвищити мобільність та безпеку таких користувачів, забезпечуючи їм можливість орієнтуватися в просторі без необхідності постійного візуального контакту з інтерфейсом.

Поєднання візуального моніторингу дорожньої обстановки з адаптивним профілюванням користувачів дає змогу підвищити точність для підвищення якості навігаційних рекомендацій. Дані комп'ютерного зору можуть використовуватись не тільки для оцінки поточного стану трафіку, але й для виявлення неявних індикаторів переваг користувачів через аналіз їх поведінки: стиль водіння, вибір смуги руху, реакції на дорожню обстановку, швидкість пересування. Це дозволяє швидше формувати точну модель переваг навіть для нових користувачів та оперативно виявляти зміни у сформованих перевагах існуючих користувачів.

Актуальність дослідження методів побудови навігаційних рекомендацій з інтеграцією комп'ютерного зору підтверджується зростаючим інтересом провідних технологічних компаній до розвитку автономних транспортних засобів та інтелектуальних систем керування трафіком. Ефективні алгоритми аналізу відеозображень та формування персоналізованих рекомендацій є

ключовими компонентами таких систем, що обумовлює практичну значущість отриманих результатів для індустрії та суспільства.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження виконано на кафедрі автоматизації проектування енергетичних процесів і систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» в рамках науково-дослідної роботи кафедри. Тематика роботи узгоджується з пріоритетними напрямками розвитку штучного інтелекту, комп'ютерного зору та інтелектуальних транспортних систем згідно зі Стратегією розвитку штучного інтелекту в Україні на період до 2030 року.

Метою роботи є підвищення точності навігаційних рекомендацій в умовах обмеженої апріорної інформації про користувача через розробку методу адаптивних рекомендацій з семантичною інтеграцією візуальних даних від системи комп'ютерного зору. Додатковою метою є розширення доступності навігаційної системи для користувачів із порушеннями зору шляхом забезпечення можливості голосового представлення результатів аналізу дорожньої сцени та рекомендацій щодо маршруту.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

- провести аналіз існуючих методів виявлення об'єктів на відеозображеннях, побудови рекомендацій для навігаційних систем та вирішення проблеми холодного старту;

- провести аналіз існуючих методів виявлення об'єктів на відеозображеннях, побудови рекомендацій для навігаційних систем та вирішення проблеми холодного старту;

- розробити концептуальну модель багаторівневої системи з виявленням еволюції переваг користувачів на основі поєднання статистичного аналізу та даних комп'ютерного зору;

- удосконалити метод побудови рекомендацій через інтеграцію модифікованої архітектури YOLOv11 з адаптивним;

- розробити інформаційну технологію реалізації, що складається з десяти етапів, процесу формування персоналізованих рекомендацій на основі PyTorch, Open Neural Network Exchange (ONNX) Runtime та мікросервісної архітектури;

- провести експериментальну перевірку розробленого методу на датасетах Berkeley DeepDrive 100K dataset (BDD100K) та власних записах з оцінкою точності виявлення, трекінгу та якості рекомендацій.

Для вирішення поставлених задач використовувались методи глибокого навчання для знаходження та трекінгу об'єктів на відеозображеннях, методи байєсівської статистики для семантичного злиття різнорідних даних, алгоритми теорії графів для пошуку оптимальних маршрутів, методи рекомендаційних систем для моделювання переваг користувачів, статистичні методи для виявлення змін у часових рядах, методи інтерпретації моделей машинного навчання для генерації пояснень рекомендацій.

Наукова новизна одержаних результатів полягає у наступному:

- вперше розроблено метод адаптивних рекомендацій для навігаційних систем, що відрізняється семантичною інтеграцією візуальних даних від модифікованої архітектури YOLOv11 зі спеціалізованою головою оцінки щільності трафіку, треступеневою стратегією вирішення холодного старту на основі демографічної інформації, трансферного навчання та візуального аналізу поведінки, механізмом виявлення змін переваг через поєднання тесту Mann-Kendall та аналізу даних комп'ютерного зору, що дозволило підвищити точність рекомендацій у режимі холодного старту на 15 відсоткових пунктів та прискорити адаптацію до змін переваг у два-три рази порівняно з існуючими підходами;

- удосконалено архітектуру YOLOv11 через додавання спеціалізованої голови для оцінки щільності трафіку та карт руху транспортних засобів, що на відміну від базової моделі дозволяє безпосередньо генерувати агреговані характеристики транспортного потоку без додаткової постобробки та

підвищити точність розпізнавання на три відсоткових пункти при збереженні швидкості обробки 42 кадри на секунду;

– отримало подальший розвиток застосування регуляризації Elastic Weight Consolidation для онлайн-навчання моделей переваг користувачів, що відрізняється адаптивним налаштуванням швидкості навчання залежно від стабільності параметрів та дозволяє відновлювати точність рекомендацій до 0.69 протягом десяти поїздок після виявлення зміни переваг при частці помилкових спрацювань лише 7 відсотків.

Практичне значення одержаних результатів. Розроблено програмний прототип системи побудови навігаційних рекомендацій з інтеграцією комп'ютерного зору, що реалізує запропонований метод адаптивних рекомендацій. Система досягла точності Precision@1 на рівні 0.72 для існуючих користувачів та 0.58 для нових користувачів у режимі холодного старту, що на 4-21 відсотковий пункт вище базових підходів. Латентність формування рекомендацій складає 380 мілісекунд при масштабованості до 50 запитів на секунду на одному сервері, що задовольняє вимоги інтерактивних застосувань. Користувацька задоволеність склала 4.3 бали з 5, при цьому 85 відсотків користувачів відзначили врахування індивідуальних переваг, 78 відсотків висловили задоволеність наявністю пояснень причин рекомендацій. Розроблені програмні компоненти можуть бути інтегровані у комерційні навігаційні системи та платформи інтелектуального керування міським трафіком для підвищення ефективності транспортних потоків та задоволеності користувачів.

Апробація результатів роботи. Основні результати дослідження доповідались та обговорювались на науково-технічних конференціях студентів та аспірантів Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського».

За результатами виконаної роботи опубліковано тези доповіді в збірнику наукових праць «SCIENTIA» за матеріалами V Міжнародної науково-

теоретичної конференції «Current scientific goals, approaches and challenges»  
(12.12.2025, м. Дрезден, Німеччина).

# 1 АНАЛІЗ МЕТОДІВ ПОБУДОВИ НАВІГАЦІЙНИХ РЕКОМЕНДАЦІЙ З ВІЗУАЛЬНИМ МОНІТОРИНГОМ ТРАФІКУ

## 1.1 Аналіз процесу формування маршрутних рекомендацій на основі відеоаналізу дорожньої обстановки

Сучасний розвиток технологій комп'ютерного зору та машинного навчання відкриває нові можливості для створення інтелектуальних систем підтримки прийняття рішень. Об'єктом дослідження в рамках даної роботи є процес підтримки прийняття рішень та формування рекомендацій для навігаційних систем на основі аналізу відеозображень дорожньої обстановки та моделювання переваг користувачів. Цей процес включає декілька взаємопов'язаних компонентів, кожен з яких має свої специфічні властивості та характеристики. Процес побудови інформаційної системи підтримки прийняття рішень для вибору маршруту характеризується необхідністю обробки великих обсягів візуальних даних у режимі реального часу або близькому до нього. Вхідними даними для такої системи є потік відеозображень, отриманих від камер спостереження, мобільних пристроїв користувачів або спеціалізованих датчиків. Ці дані мають різну якість, роздільну здатність та умови зйомки, що створює додаткові виклики для процесу обробки.

Основною властивістю об'єкту дослідження є його багатокomпонентність. Система повинна виконувати низку послідовних операцій: захоплення відеопотоку, попередня обробка зображень, виявлення об'єктів на сцені, класифікація виявлених об'єктів, визначення їх просторового розташування, аналіз динаміки руху та, зрештою, формування рекомендацій щодо оптимального маршруту. Кожен із цих етапів вимагає застосування специфічних алгоритмів та методів обробки даних. Процес пошуку інформації в мережі інтернет для збагачення контексту прийняття рішень також є важливою складовою об'єкту дослідження. Система повинна мати можливість

отримувати актуальну інформацію про стан доріг, погодні умови, дорожні події та інші фактори, які можуть вплинути на вибір оптимального маршруту. Інтеграція даних від різних джерел та їх узгодження з результатами аналізу відеозображень становить окрему задачу, яка потребує розробки спеціалізованих методів синтезу інформації.

Процеси паралельних обчислень відіграють ключову роль у забезпеченні ефективності системи. Сучасні алгоритми розпізнавання образів, особливо ті, що базуються на глибоких нейронних мережах, є обчислювально складними та вимагають значних ресурсів. Для забезпечення прийнятної швидкості обробки необхідно використовувати паралелізм на різних рівнях: від векторизації операцій на рівні процесора до розподіленої обробки на кластерах графічних процесорів.

Процес побудови рекомендацій у рекомендаційних системах для навігації має свою специфіку порівняно з традиційними рекомендаційними системами для товарів або контенту [1]. Якість рекомендацій залежить не тільки від точності розпізнавання об'єктів на зображеннях, але й від розуміння контексту руху, прогнозування змін дорожньої ситуації та врахування індивідуальних переваг користувача. Система повинна враховувати часові обмеження, пріоритети користувача щодо критеріїв оптимальності маршруту та динамічні зміни в оточенні.

Процес виявлення атак на рейтинги в рекомендаційних системах набуває особливого значення в контексті систем навігації. Зловмисники можуть намагатися маніпулювати рейтингами маршрутів або об'єктів інфраструктури з різних причин: від комерційних інтересів до створення штучних заторів. Виявлення таких аномалій вимагає аналізу патернів поведінки користувачів, статистичного аналізу розподілу оцінок та виявлення нетипових кореляцій у даних. Властивості об'єкту дослідження визначають специфічні вимоги до методів та алгоритмів, які використовуються для побудови системи підтримки прийняття рішень. Перша група вимог стосується точності та надійності розпізнавання об'єктів на відеозображеннях. Система повинна коректно

ідентифікувати різноманітні об'єкти дорожньої інфраструктури, транспортні засоби, перешкоди та інші елементи сцени навіть за складних умов освітлення, погодних умов або часткового перекриття об'єктів.

Друга група вимог пов'язана з обчислювальною ефективністю. Оскільки система повинна працювати в режимі реального часу або близькому до нього, алгоритми обробки повинні забезпечувати прийнятну швидкість при збереженні високої точності. Це вимагає компромісу між складністю моделей та швидкістю їх виконання, а також ефективного використання апаратних ресурсів.

Третя група вимог стосується адаптивності системи. Об'єкт дослідження повинен мати здатність адаптуватися до змінних умов середовища, нових типів об'єктів, які раніше не зустрічалися, та еволюції поведінки користувачів. Це передбачає необхідність застосування методів машинного навчання, які дозволяють системі постійно вдосконалюватися на основі нових даних.

Четверта група вимог пов'язана з масштабованістю. Система повинна ефективно працювати як з невеликими обсягами даних від одного користувача, так і з величезними потоками інформації від тисяч користувачів одночасно. Архітектура системи має підтримувати горизонтальне масштабування та розподілену обробку даних.

П'ята група вимог стосується інтерпретованості рішень. У контексті систем підтримки прийняття рішень важливо не тільки отримати рекомендацію, але й розуміти, на основі яких факторів вона була сформована. Це особливо актуально для критичних ситуацій, коли вибір маршруту може впливати на безпеку користувача.

Аналіз властивостей об'єкту дослідження дозволяє виділити основні виклики, які постають перед розробниками систем підтримки прийняття рішень на основі розпізнавання відеозображень. Перший виклик пов'язаний з варіабельністю вхідних даних. Відеозображення, отримані в різних умовах освітлення, під різними кутами та з різною якістю, можуть значно відрізнятися за своїми характеристиками. Система повинна бути стійкою до таких варіацій

та забезпечувати стабільну якість розпізнавання. Другий виклик стосується обчислювальної складності сучасних методів глибокого навчання. Моделі, які забезпечують найвищу точність розпізнавання, часто є надто ресурсоемними для застосування в системах реального часу на мобільних пристроях або вбудованих системах. Необхідність балансу між точністю та швидкістю обробки залишається актуальною проблемою. Третій виклик пов'язаний з динамічністю середовища. Дорожня ситуація може змінюватися дуже швидко: з'являються нові перешкоди, змінюються умови руху, виникають непередбачені події. Система повинна оперативного реагувати на такі зміни та коригувати свої рекомендації відповідно до нової інформації [2]. Четвертий виклик стосується персоналізації рекомендацій. Різні користувачі можуть мати різні критерії оптимальності маршруту: хтось віддає перевагу найкоротшому шляху, хтось найшвидшому, а хтось обирає найбільш живописний або найбезпечніший маршрут. Система повинна враховувати індивідуальні переваги користувачів та адаптувати свої рекомендації відповідно. П'ятий виклик пов'язаний з забезпеченням приватності даних користувачів. Відеозображення, особливо ті, що містять інформацію про переміщення конкретних осіб, є чутливими персональними даними. Система повинна забезпечувати належний рівень захисту таких даних, використовуючи методи анонімізації, шифрування та контролю доступу.

Структура процесу побудови рекомендацій включає декілька послідовних етапів, кожен з яких має свої вхідні дані, послідовність дій, вихідні дані та обмеження. Перший етап полягає в захопленні відеопотоку та його попередній обробці. Вхідними даними є необроблений відеопотік від камери або іншого джерела. На цьому етапі виконується корекція освітлення, зменшення шумів, стабілізація зображення та інші операції, спрямовані на покращення якості вхідних даних для подальшої обробки.

Другий етап присвячений виявленню та локалізації об'єктів на зображеннях. Для кожного попередньо обробленого кадру  $I'$  детектор об'єктів формує множину обмежувальних рамок наступного вигляду:

$$B = \{(x_i, y_i, w_i, h_i, c_i, p_i)\}_{i=1}^N,$$

де  $B$  – множина всіх рамок виявлених об'єктів;

$x_i$  – координати центру рамки по осі  $x$ ;

$y_i$  – координати центру рамки по осі  $y$ ;

$w_i$  – ширина рамки;

$h_i$  – висота рамки;

$c_i$  – клас виявленого об'єкта;

$p_i$  – значення впевненості моделі;

$N$  – кількість виявлених об'єктів (кількість рамок) на кадрі;

$i$  – номер (індекс) об'єкта.

Вхідними даними є попередньо оброблені кадри відео. Послідовність дій включає застосування алгоритмів виявлення об'єктів, які визначають координати обмежувальних рамок навколо виявлених об'єктів. Вихідними даними є список координат областей, що містять потенційно цікаві об'єкти.

Третій етап стосується класифікації виявлених об'єктів. На цьому етапі кожен виявлений об'єкт аналізується для визначення його типу: автомобіль, пішохід, дорожній знак, світлофор тощо. Використовуються методи класифікації на основі глибоких нейронних мереж, які забезпечують високу точність розпізнавання.

Четвертий етап передбачає аналіз просторового розташування та відносин між об'єктами. Система визначає положення об'єктів відносно камери та один одного, аналізує їх розміри та орієнтацію в просторі. Ця інформація є критично важливою для розуміння дорожньої ситуації та прогнозування можливих сценаріїв її розвитку.

П'ятий етап включає відстеження об'єктів у часі та аналіз їх траєкторій. Для кожного об'єкта визначається його положення у часі у вигляді векторної функції:

$$p_i(t) = (x_i(t), y_i(t)),$$

де  $p_i(t)$  – вектор положення об'єкта  $i$  у момент часу  $t$ ;

$x_i(t)$  – координата положення об'єкта  $i$  у момент часу  $t$  по осі  $x$ ;

$y_i(t)$  – координата положення об'єкта  $i$  у момент часу  $t$  по осі  $y$ ;

$i$  – номер (індекс) об'єкта;

$t$  – момент часу;

Система повинна підтримувати ідентичність об'єктів між послідовними кадрами, будувати траєкторії їх руху та аналізувати динаміку зміни характеристик. Це дозволяє прогнозувати майбутнє положення об'єктів та виявляти потенційно небезпечні ситуації.

Шостий етап стосується інтеграції інформації від різних джерел. Окрім даних від системи комп'ютерного зору, можуть використовуватися дані від Global Positioning System (GPS), інформація про стан доріг з інтернет-сервісів, історичні дані про завантаженість маршрутів тощо. Узгодження цих гетерогенних даних та їх синтез у єдину картину є складною задачею, яка вимагає застосування методів злиття даних.

Сьомий етап передбачає формування рекомендацій щодо вибору маршруту. На основі зібраної та проаналізованої інформації система генерує декілька альтернативних варіантів маршрутів та оцінює їх ймовірність бути оптимальними. Інтеграція відеоданих та профілю користувача може бути подана у вигляді байєсівської моделі:

$$P(R|V, U) = \frac{P(V|R)P(R|U)}{P(V)},$$

де  $V$  – візуальні ознаки дорожньої сцени;

$U$  – профіль користувача;

$R$  – рекомендований маршрут.

Ці рекомендації повинні враховувати не тільки поточну ситуацію, але й прогнозовані зміни.

Восьмий етап включає презентацію рекомендацій користувачеві у зрозумілій та зручній формі. Інтерфейс системи повинен надавати не тільки інформацію про рекомендовані маршрути, але й пояснення причин їх вибору, альтернативні варіанти та можливість коригування параметрів пошуку. Загальну структуру застосунку продемонстровано на рисунку 1.1.

Класифікація задач, які вирішуються в рамках об'єкту дослідження, може бути проведена за різними критеріями. За типом вхідних даних можна виділити задачі обробки статичних зображень та задачі обробки відеопотоків. Перші працюють з окремими кадрами незалежно один від одного, другі враховують темпоральну послідовність кадрів та динаміку змін.

За характером виходу розрізняють задачі розпізнавання, класифікації, сегментації та трекінгу об'єктів. Виявлення визначає наявність та положення об'єктів, класифікація присвоює їм категорії, сегментація виділяє точні контури об'єктів на піксельному рівні, а трекінг підтримує їх ідентичність у часі.

За рівнем абстракції можна виділити низькорівневі задачі обробки зображень, такі як виділення контурів, розпізнавання ключових точок, та високорівневі задачі семантичного розуміння сцени, такі як визначення дій об'єктів, їх взаємодій та прогнозування майбутніх подій.

За критерієм взаємодії з користувачем розрізняють автоматичні системи, які працюють без втручання людини, напівавтоматичні системи, які потребують періодичного підтвердження або коригування рішень, та інтерактивні системи, які активно залучають користувача до процесу прийняття рішень.

За часовими обмеженнями виділяють системи реального часу, які повинні надавати результати в межах жорстких часових рамок, системи майже реального часу, які допускають невеликі затримки, та офлайн системи, які обробляють раніше записані дані без часових обмежень.

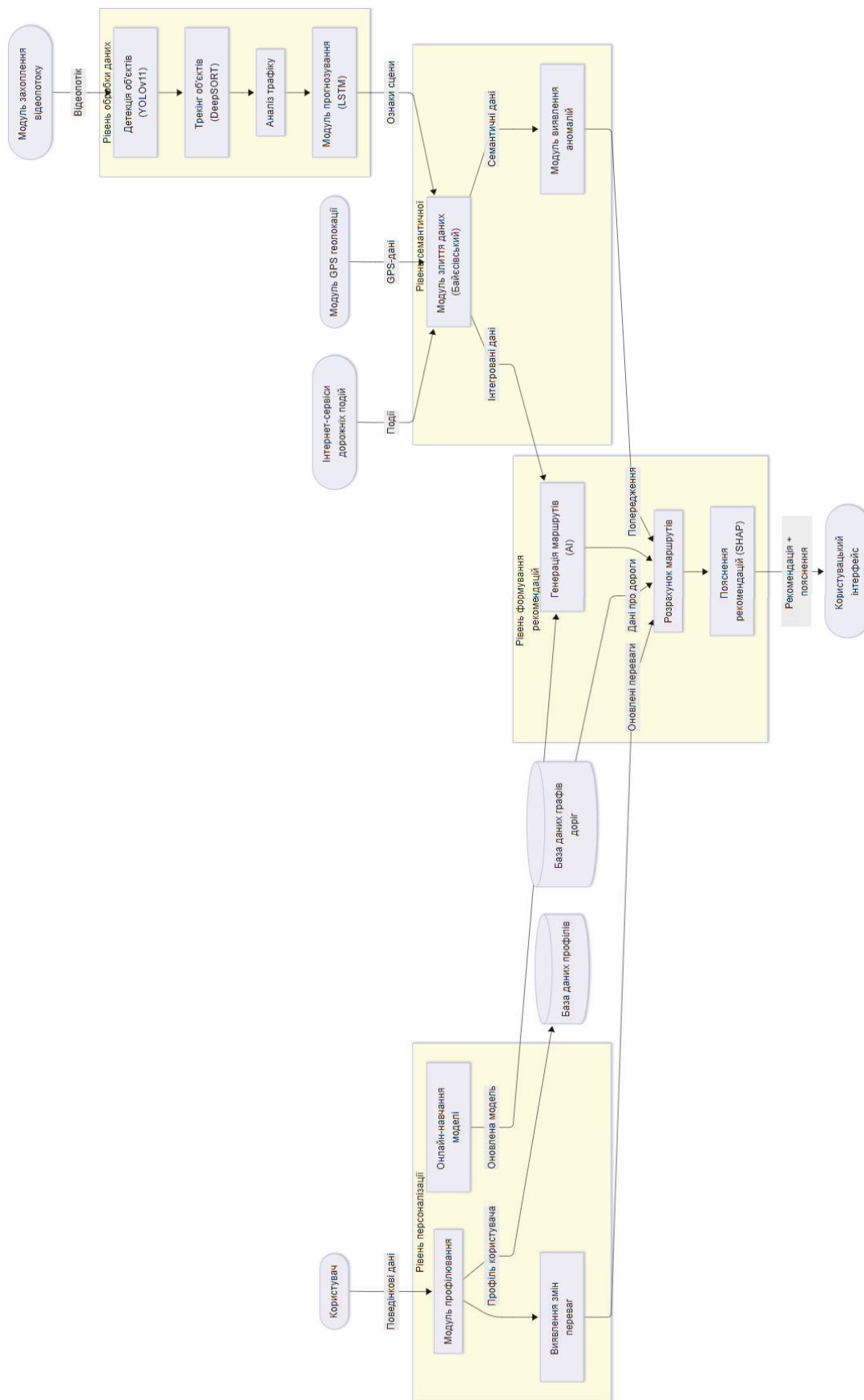


Рисунок 1.1 – Архітектура системи

Обмеження, які існують у контексті об'єкту дослідження, можна поділити на технічні, алгоритмічні та практичні. Технічні обмеження пов'язані з апаратними ресурсами: обчислювальною потужністю процесорів та графічних прискорювачів, обсягом доступної пам'яті, пропускну здатністю каналів передачі даних та енергоспоживанням пристроїв. Алгоритмічні обмеження стосуються фундаментальних властивостей методів машинного навчання та обробки зображень. Існує компроміс між точністю моделей та їх обчислювальною складністю, між узагальнюючою здатністю та схильністю до перенавчання, між швидкістю навчання та якістю отриманих рішень.

Практичні обмеження пов'язані з доступністю та якістю даних для навчання моделей. Створення великих анотованих датасетів вимагає значних зусиль та ресурсів. Крім того, дані можуть бути незбалансованими: деякі класи об'єктів зустрічаються значно рідше за інші, що ускладнює навчання моделей для їх розпізнавання. Особливу увагу слід приділити обмеженням, пов'язаним з приватністю та безпекою даних. Обробка відеозображень, особливо в публічних місцях, підпадає під дію законодавства про захист персональних даних. Система повинна забезпечувати анонімізацію особистої інформації та відповідати вимогам регуляторних органів. Екологічні обмеження також набувають все більшого значення. Навчання великих моделей глибокого навчання вимагає значних енергетичних ресурсів та має відчутний вуглецевий слід. Існує потреба в розробці більш енергоефективних алгоритмів та методів, які забезпечують прийнятну якість при менших витратах ресурсів.

## 1.2 Огляд сучасних підходів до розпізнавання об'єктів та побудови персоналізованих рекомендацій

Для вирішення задачі побудови рекомендацій щодо вибору маршруту на основі розпізнавання відеозображень існує широкий спектр методів та

підходів, кожен з яких має свої переваги, недоліки та області застосування. Комплексний аналіз цих методів дозволяє визначити найбільш перспективні напрямки для розробки ефективної системи підтримки прийняття рішень. Класифікація методів розпізнавання об'єктів на зображеннях може бути проведена за різними критеріями [3]. За принципом роботи виділяють традиційні методи комп'ютерного зору, які базуються на виділенні вручну створених ознак, та методи глибокого навчання, які автоматично навчаються релевантним представленням даних. Традиційні методи включають підходи на основі виділення контурів, аналізу текстур, дескрипторів ключових точок та інших експертно визначених характеристик зображень.

Методи на основі виділення контурів використовують перепади інтенсивності пікселів для визначення меж об'єктів. Класичні оператори, такі як Sobel, Prewitt, Canny, дозволяють виділити контури з різним ступенем точності та стійкості до шуму. Ці методи є обчислювально ефективними, але чутливими до умов освітлення та якості зображення. Їх основна перевага полягає у простоті реалізації та швидкості роботи, що робить їх придатними для попередньої обробки зображень або застосування в системах з обмеженими ресурсами.

Методи на основі аналізу текстур враховують статистичні властивості розподілу інтенсивностей пікселів у локальних областях зображення. Підходи, що базуються на фільтрах Габора, локальних бінарних патернах або статистиках другого порядку, дозволяють характеризувати текстурні особливості поверхонь об'єктів. Ці методи ефективні для класифікації об'єктів з характерними текстурами, але мають обмежену здатність до узагальнення на нові типи об'єктів.

Методи на основі дескрипторів ключових точок, такі як Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), Oriented FAST and Rotated BRIEF (ORB), виділяють стабільні локальні особливості зображення та описують їх за допомогою векторів ознак. Ці методи інваріантні до масштабування, повороту та часткових змін освітлення, що робить їх

стійкими до варіацій у вхідних даних. Однак їх точність може бути недостатньою для складних сцен з великою кількістю об'єктів та їх перекриттям.

Методи глибокого навчання революціонізували область комп'ютерного зору, забезпечивши значне підвищення точності розпізнавання об'єктів. Згорткові нейронні мережі є основою більшості сучасних систем аналізу зображень. Архітектури, такі як Visual Geometry Group network (VGG), Residual Network (ResNet), Inception, EfficientNet, досягають надлюдської точності на багатьох бенчмарках завдяки здатності автоматично навчатися ієрархічним представленням ознак.

Для задачі виявлення об'єктів були розроблені спеціалізовані архітектури, які можна поділити на дві основні категорії: двохетапні детектори та одноетапні детектори. Двохетапні детектори, такі як Regions with Convolutional Neural Networks (R-CNN), Fast R-CNN, Faster R-CNN, спочатку генерують пропозиції регіонів, які потенційно містять об'єкти, а потім класифікують ці регіони та уточнюють їх координати. Такий підхід забезпечує високу точність, але вимагає значних обчислювальних ресурсів. Одноетапні детектори, такі як YOLO, Single Shot MultiBox Detector (SSD), RetinaNet, виконують ідентифікацію та класифікацію об'єктів в один прохід через нейронну мережу. Це значно прискорює процес обробки, роблячи ці методи придатними для застосування в системах реального часу. Сучасні версії одноетапних детекторів, такі як YOLOv8 або YOLOv11, досягають точності, порівнянної з двохетапними підходами, при значно вищій швидкості обробки.

Таблиця 1.1 – Порівняння сучасних моделей виявлення

Модель	Тип детектора	mAP (%)	FPS	Переваги	Недоліки
YOLOv8	One-stage	52–55	90–120	Висока швидкість	Менше точності на дрібних об'єктах
YOLOv11	One-stage	56–60	110–150	Оптимальний баланс точність/швидкість	Мало досліджень
Faster R-CNN	Two-stage	60–65	15–25	Висока точність	Дуже повільна
RetinaNet	One-stage	55–58	40–55	Хороша точність	Середня швидкість

Методи семантичної сегментації виділяють кожен піксель зображення, присвоюючи йому категорію об'єкта, до якого він належить. Архітектури, такі як Fully Convolutional Network (FCN), U-shaped Network (U-Net), Deep Convolutional Neural Networks for Semantic Segmentation (DeepLab), дозволяють отримати детальну сегментацію сцени, що є корисним для точного розуміння просторового розташування об'єктів. Інстанс-сегментація, реалізована в методах Mask R-CNN, Segmenting Objects by Locations (SOLO), дозволяє розрізняти окремі екземпляри об'єктів одного класу.

Методи трекінгу об'єктів відслідковують рух об'єктів між послідовними кадрами відео. Класичні підходи, такі як фільтр Калмана або фільтр частинок, використовують моделі руху для прогнозування положення об'єктів. Сучасні методи на основі глибокого навчання, такі як Deep Simple Online and Realtime Tracking (DeepSORT), Fair Multi-Object Tracking (FairMOT), об'єднують ідентифікацію та трекінг у єдиний процес, забезпечуючи стійке відстеження навіть за умов оклюзії та зміни зовнішнього вигляду об'єктів.

Класифікація методів побудови рекомендацій у навігаційних системах відображає різноманітність підходів до оптимізації маршрутів. За критерієм оптимізації методи можна поділити на детерміністичні, які завжди обирають оптимальний маршрут згідно з заданою функцією цілі, та стохастичні, які враховують невизначеність у вхідних даних та можуть пропонувати різні маршрути за однакових умов. Детерміністичні методи включають класичні алгоритми пошуку на графах, такі як алгоритм Дейкстри, A\*, Bellman-Ford. Ці алгоритми гарантують знаходження оптимального маршруту, якщо такий існує, але вимагають повної інформації про граф доріг та ваги ребер. Модифікації цих алгоритмів дозволяють враховувати динамічні зміни ваг ребер, що відповідає зміні умов руху. Стохастичні методи використовують імовірнісні моделі для опису невизначеності в дорожній обстановці. Байєсівські мережі, Марковські моделі прийняття рішень дозволяють враховувати очікувані зміни в умовах руху та оцінювати ризики, пов'язані з вибором того чи іншого маршруту. Такі методи особливо корисні для довгострокового планування маршрутів, коли прогнозування майбутніх умов відіграє важливу роль.

Методи, орієнтовані на схожість користувачів, використовують колаборативну фільтрацію для побудови рекомендацій. Ідея полягає в тому, що користувачі з подібними характеристиками та історією переміщень мають схожі переваги щодо маршрутів. Аналізуючи вибори інших користувачів, система може рекомендувати маршрути, які найбільш імовірно підійдуть даному користувачеві. Ці методи ефективні за наявності великої кількості даних про поведінку користувачів, але можуть мати проблеми холодного старту для нових користувачів.

Методи, орієнтовані на схожість маршрутів та об'єктів інфраструктури, використовують контентну фільтрацію. Система аналізує характеристики маршрутів: довжину, час проїзду, тип доріг, наявність цікавих об'єктів уздовж шляху тощо. На основі історії вибору користувача будується профіль його переваг, який потім використовується для пошуку подібних маршрутів.

Перевага цього підходу в можливості рекомендувати нові маршрути навіть без даних про поведінку інших користувачів.

Гібридні методи об'єднують переваги різних підходів, компенсуючи їх недоліки. Наприклад, система може використовувати контентну фільтрацію для початкового вибору кандидатів маршрутів, а потім застосовувати колаборативну фільтрацію для їх ранжування. Такі методи зазвичай забезпечують кращу якість рекомендацій, ніж будь-який окремий підхід.

Методи глибокого навчання також знаходять застосування в системах рекомендацій для навігації. Рекурентні нейронні мережі та трансформери можуть моделювати послідовності переміщень користувача та передбачати його майбутні дії. Графові нейронні мережі дозволяють ефективно працювати з топологією дорожньої мережі та враховувати складні взаємозв'язки між різними маршрутами.

Методи мультиагентного моделювання розглядають систему навігації як взаємодію безлічі автономних агентів, кожен з яких представляє окремого користувача або транспортний засіб. Ці методи дозволяють моделювати колективну поведінку та оптимізувати маршрути з урахуванням їх впливу на загальний транспортний потік. Такий підхід особливо корисний для міських систем управління трафіком.

Методи оптимізації на основі еволюційних алгоритмів, таких як генетичні алгоритми, оптимізація роєм частинок, можуть використовуватися для пошуку маршрутів у складних умовах з множинними критеріями оптимальності. Ці методи здатні знаходити прийнятні рішення в задачах, де класичні методи оптимізації є неефективними через високу розмірність простору пошуку або складність функції цілі.

Характеристики методів визначають їх придатність для різних сценаріїв застосування. Точність методу визначає, наскільки коректно він розпізнає об'єкти або наскільки оптимальними є рекомендовані маршрути. Для методів комп'ютерного зору точність зазвичай оцінюється за допомогою метрик, таких

як точність розпізнавання, повнота, F-міра, середня точність на різних порогах перекриття обмежувальних рамок.

Швидкість обробки є критично важливою характеристикою для систем реального часу. Вона вимірюється кількістю кадрів, які система може обробити за секунду, або часом, необхідним для обробки одного кадру. Існує фундаментальний компроміс між точністю та швидкістю: найточніші методи зазвичай є найповільнішими, тоді як найшвидші методи можуть жертвувати точністю заради продуктивності. Обчислювальна складність методу визначає кількість операцій, необхідних для обробки вхідних даних, та обсяг пам'яті, який при цьому використовується. Для глибоких нейронних мереж обчислювальна складність залежить від кількості параметрів моделі, кількості операцій у кожному шарі та розміру вхідних даних. Методи зі складністю квадратичного або вищого порядку можуть бути непрактичними для обробки великих обсягів даних.

Стійкість до варіацій вхідних даних характеризує здатність методу зберігати точність за умов зміни умов зйомки: освітлення, погоди, кута огляду, часткового перекриття об'єктів тощо. Методи глибокого навчання, навчені на великих різноманітних датасетах, зазвичай демонструють кращу стійкість порівняно з традиційними методами з вручну створеними ознаками. Узагальнююча здатність методу визначає, наскільки добре він працює з даними, які відрізняються від тих, на яких він навчався. Це особливо важливо для систем, які розгортаються в реальних умовах, де можуть зустрічатися об'єкти або ситуації, не представлені в навчальних даних. Методи з хорошою узагальнюючою здатністю менш схильні до перенавчання та демонструють стабільну продуктивність на різноманітних даних.

Інтерпретованість методу стосується можливості розуміти, як і чому система прийняла те чи інше рішення. Традиційні методи з явними правилами та логікою є більш інтерпретованими, тоді як глибокі нейронні мережі часто розглядаються як чорні скриньки. Для критичних застосувань, таких як автономна навігація, інтерпретованість може бути важливою вимогою.

Масштабованість методу характеризує його здатність ефективно працювати зі зростанням обсягу даних або кількості користувачів. Методи, які добре масштабуються, можуть бути застосовані як для невеликих локальних систем, так і для великомасштабних сервісів з мільйонами користувачів. Розподілені алгоритми та архітектури забезпечують кращу масштабованість порівняно з централізованими підходами. Адаптивність методу відображає його здатність навчатися на нових даних та покращувати свою продуктивність з часом. Методи онлайн-навчання можуть постійно оновлюватися в міру надходження нових даних, тоді як статичні методи потребують періодичного переучування на оновлених датасетах. Адаптивні методи особливо цінні в динамічних середовищах, де характеристики даних можуть змінюватися з часом.

### 1.3 Аналіз методів побудови рекомендацій у режимі холодного старту з використанням візуальних даних

Розглядаючи конкретну задачу побудови рекомендацій для вибору маршруту в ситуації, що характеризується динамічними змінами дорожньої обстановки та необхідністю врахування інформації від системи розпізнавання відеозображень, доцільно зосередитися на методах, які спеціально розроблені для цього класу проблем або можуть бути ефективно адаптовані для їх вирішення. Гібридні методи побудови рекомендацій представляють особливий інтерес для задачі навігації з використанням комп'ютерного зору. Ці методи об'єднують інформацію від різних джерел: дані розпізнавання об'єктів на відеозображеннях, демографічні характеристики користувачів, географічні особливості місцевості, історичні дані про завантаженість доріг та актуальну інформацію про дорожні події [4-6].

Одним із підходів є використання багаторівневої архітектури, де на нижньому рівні працюють спеціалізовані моделі для обробки кожного типу даних: згорткові нейронні мережі для аналізу відеозображень, рекурентні мережі для моделювання послідовностей переміщень користувача, графові нейронні мережі для роботи з топологією дорожньої мережі. На вищому рівні результати цих моделей інтегруються для формування фінальних рекомендацій.

Такий підхід дозволяє ефективно використовувати переваги кожного типу моделей та забезпечує модульність системи, що спрощує її розробку, тестування та підтримку. Крім того, модульна архітектура дозволяє незалежно оновлювати окремі компоненти системи без необхідності переробки всієї системи. Ключовою проблемою при об'єднанні різних джерел інформації є узгодження їх семантики та темпоральних характеристик. Дані від системи комп'ютерного зору надходять у режимі реального часу з високою частотою, тоді як інформація від інтернет-сервісів може оновлюватися з меншою частотою та мати певну затримку. Географічні дані є статичними або змінюються дуже повільно. Ефективна інтеграція цих гетерогенних потоків даних вимагає застосування методів часової синхронізації та узгодження.

Для вирішення проблеми холодного старту рекомендаційної системи, коли для нового користувача відсутня історія переміщень, можуть використовуватися демографічні характеристики та початкові налаштування користувача [7]. Система може запитувати у користувача його переваги щодо критеріїв вибору маршруту: пріоритет швидкості, економії палива, мінімізації стресу від водіння, наявності цікавих об'єктів уздовж шляху тощо. Крім того, для нових користувачів може застосовуватися стратегія дослідження, коли система пропонує різноманітні типи маршрутів для збору інформації про переваги користувача. Баланс між дослідженням нових варіантів та використанням вже відомої інформації є класичною проблемою в теорії навчання з підкріпленням та може вирішуватися за допомогою стратегій, таких як *epsilon-greedy* або *Upper Confidence Bound*.

Методи виявлення атак на рекомендаційну систему є важливою складовою забезпечення надійності навігаційного сервісу. У контексті систем навігації атаки можуть мати різні форми: штучне завищення або заниження рейтингів певних маршрутів, створення фальшивих профілів користувачів для маніпуляції рекомендаціями, ін'єкція шумових даних для дезорганізації роботи системи.

Для виявлення таких аномалій використовуються методи статистичного аналізу, які шукають відхилення від очікуваних патернів поведінки. Раптові зміни в розподілі оцінок маршрутів, незвичайні кореляції між діями різних користувачів, аномальна активність нових акаунтів можуть свідчити про спробу атаки. Методи машинного навчання, зокрема методи виявлення аномалій на основі автоенкодерів або One-Class Support Vector Machine (One-Class SVM), можуть автоматично навчатися розпізнавати підозрілі патерни. Важливим аспектом є також захист від атак, спрямованих на систему комп'ютерного зору. Адверсальні атаки, коли спеціально створені зображення вводять систему розпізнавання в оману, можуть призвести до невірних рекомендацій щодо маршруту. Методи підвищення стійкості моделей до таких атак включають адверсальне навчання, ансамблі моделей, верифікацію вхідних даних.

Методи глибокого навчання для аналізу відеопотоків представляють основу сучасних систем комп'ютерного зору для навігації. Архітектури YOLO останніх версій демонструють видатний баланс між точністю та швидкістю обробки. YOLOv11, наприклад, використовує вдосконалену архітектуру backbone мережі з ефективними блоками згортки, що забезпечують краще виділення ознак при зменшених обчислювальних витратах. Ключовою особливістю сучасних архітектур виявлення об'єкта є використання механізмів уваги, які дозволяють мережі фокусуватися на найбільш релевантних частинах зображення. Це особливо важливо для складних дорожніх сцен, де одночасно присутня велика кількість об'єктів різних типів

та розмірів. Механізми просторової та каналної уваги дозволяють покращити точність знаходження без значного збільшення обчислювальної складності.

Для підвищення точності розпізнавання об'єктів малого розміру, які часто зустрічаються в дорожніх сценах на великій відстані від камери, використовуються багатомасштабні підходи. Feature Pyramid Networks дозволяють мережі аналізувати зображення на різних рівнях деталізації, ефективно виявляючи як великі, так і малі об'єкти. Додаткові техніки, такі як multi-scale training та test-time augmentation, також сприяють покращенню якості фіксації. Трансферне навчання відіграє ключову роль у побудові ефективних моделей комп'ютерного зору для навігаційних систем. Моделі, попередньо навчені на великих загальних датасетах зображень, таких як ImageNet або Common Objects in Context (COCO), вже мають розвинені здатності до виділення базових візуальних ознак. Донавчання таких моделей на спеціалізованих датасетах дорожніх сцен вимагає значно менше даних та обчислювальних ресурсів порівняно з навчанням з нуля.

Однак важливо враховувати доменний зсув між загальними та спеціалізованими даними. Характеристики дорожніх сцен можуть значно відрізнятися від типових зображень у загальних датасетах: специфічні умови освітлення, динаміка руху об'єктів, геометричні спотворення від різних кутів зйомки. Методи адаптації домену, такі як adversarial domain adaptation, можуть допомогти подолати цю проблему [8].

Для задачі трекінгу об'єктів у відеопотоках особливо ефективними показали себе методи, що об'єднують ідентифікації та трекінг у єдиний процес. ByteTrack, наприклад, використовує низькопорогову ідентифікацію для виявлення всіх потенційних об'єктів, а потім застосовує алгоритм асоціації для підтримання траєкторій об'єктів між кадрами. Такий підхід забезпечує стійке відстеження навіть за умов тимчасової оклюзії об'єктів. Важливою проблемою є підтримання ідентичності об'єктів при їх повторному появі в кадрі після періоду відсутності. Re-identification моделі, які навчаються розпізнавати унікальні візуальні характеристики об'єктів, можуть допомогти правильно

асоціювати об'єкти після їх повторного виявлення. Це особливо актуально для систем з декількома камерами, що покривають різні ділянки маршруту.

Методи прогнозування траєкторій руху об'єктів дозволяють системі передбачати майбутні дії учасників дорожнього руху та попереджувати потенційно небезпечні ситуації. Рекурентні нейронні мережі та трансформери можуть моделювати послідовності положень об'єктів та передбачати їх майбутнє розташування. Такі прогнози можуть використовуватися для коригування рекомендацій щодо маршруту або швидкості руху. Графові нейронні мережі показують перспективні результати для моделювання взаємодій між різними об'єктами на дорожній сцені. Представлення сцени як графу, де вершини відповідають об'єктам, а ребра їх взаємодіям, дозволяє враховувати складні соціальні взаємодії між учасниками руху. Це особливо важливо для прогнозування колективної поведінки в складних дорожніх ситуаціях.

Для навігаційних систем критично важливою є здатність працювати в режимі реального часу на обладнанні з обмеженими ресурсами, таких як мобільні пристрої або вбудовані системи автомобілів. Методи оптимізації моделей глибокого навчання, такі як квантизація, pruning, knowledge distillation, дозволяють значно зменшити розмір моделей та прискорити їх роботу при мінімальній втраті точності.

Квантизація зменшує точність представлення параметрів моделі, використовуючи 8-бітні або навіть 4-бітні цілочисельні значення замість 32-бітних чисел з плаваючою точкою. Це зменшує розмір моделі в 4-8 разів та прискорює обчислення на апаратних прискорювачах, оптимізованих для цілочисельної арифметики.

Pruning видаляє менш важливі параметри або навіть цілі шари нейронної мережі, створюючи розріджені структури, які вимагають менше пам'яті та обчислень. Структурний pruning, який видаляє цілі канали або шари, особливо ефективний для прискорення роботи на стандартному обладнанні без спеціалізованої підтримки розріджених обчислень.

Knowledge distillation дозволяє навчити компактну модель-учня імітувати поведінку великої точної моделі-вчителя. Учень навчається не тільки на правильних відповідях, але й на розподілі ймовірностей класів, що генерує вчитель. Це дозволяє учню досягти точності, близької до вчителя, при значно менших обчислювальних вимогах. Недоліки існуючих методів для вирішення звуженої задачі включають декілька ключових аспектів. По-перше, більшість сучасних методів розпізнавання об'єктів на відеозображеннях навчені на загальних датасетах і можуть не враховувати специфіку дорожніх сцен конкретного регіону або країни. Різниця в дорожній розмітці, знаках, архітектурі будівель можуть призводити до зниження точності розпізнавання.

По-друге, інтеграція даних від системи комп'ютерного зору з іншими джерелами інформації часто реалізується на поверхневому рівні, без глибокого семантичного злиття. Система може просто комбінувати рекомендації від різних модулів, не враховуючи взаємозв'язки та можливі протиріччя між ними.

По-третє, більшість методів недостатньо враховують темпоральну динаміку дорожньої ситуації. Вони аналізують поточний стан, але не моделюють його еволюцію в часі. Це призводить до неоптимальних рекомендацій у ситуаціях, коли дорожня обстановка швидко змінюється.

По-четверте, персоналізація рекомендацій часто обмежується простим врахуванням історії переміщень користувача, без глибокого розуміння його справжніх переваг та контексту поточної поїздки. Система може не враховувати, що один і той самий користувач може мати різні критерії вибору маршруту залежно від мети подорожі, часу доби, присутності пасажирів тощо.

По-п'яте, питання стійкості до адверсальних атак та помилок у вхідних даних залишається недостатньо дослідженим для навігаційних систем з комп'ютерним зором. Більшість методів припускають, що вхідні дані є коректними та не містять навмисних спотворень, що може бути нереалістичним у реальних умовах експлуатації. Для подолання цих недоліків необхідний розвиток нових методів та підходів, які б ефективно інтегрували сучасні досягнення комп'ютерного зору, рекомендаційних систем та методів

машинного навчання, враховуючи специфічні вимоги та обмеження навігаційних застосувань.

#### 1.4 Постановка задачі дослідження

На основі проведеного аналізу властивостей об'єкту дослідження, існуючих методів вирішення задачі та методів, що вирішують звужену задачу, можна сформулювати конкретну постановку задачі для даного дослідження.

Об'єктом дослідження є процеси побудови рекомендацій щодо вибору маршруту в системах електронної комерції та навігації на основі аналізу відеозображень, отриманих у режимі реального часу або близькому до нього.

Предметом дослідження є методи побудови рекомендацій у ситуації холодного старту рекомендаційної системи, коли відсутня або обмежена інформація про історію переміщень користувача, з використанням даних від системи комп'ютерного зору для аналізу поточної дорожньої обстановки та контекстної інформації з інших джерел.

Проблема дослідження полягає у протиріччі між можливостями існуючих методів побудови рекомендацій та задачами практики в конкретній ситуації. Існуючі методи побудови рекомендацій у навігаційних системах переважно покладаються на демографічні та географічні характеристики нових користувачів для побудови персоналізованого переліку товарів та послуг, однак вони не враховують можливі зміни інтересів вже існуючих користувачів та не можуть адаптуватися до динамічних змін дорожньої обстановки, виявлених системою комп'ютерного зору.

Після таких змін, як зміна місця роботи, місця проживання, появи нових транспортних засобів, зміни особистих обставин, користувач де-факто стає новим клієнтом з іншими потребами та перевагами. Традиційні рекомендаційні системи продовжують пропонувати маршрути на основі

застарілої історії, не враховуючи нової реальності. Інтеграція даних від системи комп'ютерного зору може допомогти виявити такі зміни та адаптувати рекомендації відповідно. Наприклад, якщо система розпізнавання виявляє, що користувач регулярно їздить новим маршрутом, відвідує нові локації, використовує інший тип транспорту, це може сигналізувати про зміну його потреб. Система повинна мати можливість виявляти такі зміни та коригувати модель переваг користувача. Крім того, поточна дорожня обстановка, виявлена системою комп'ютерного зору, може вказувати на необхідність відхилення від звичних маршрутів користувача. Затори, дорожні роботи, аварії, погодні умови вимагають динамічної адаптації рекомендацій. Інтеграція візуальної інформації про стан доріг з традиційними джерелами даних може забезпечити більш точні та актуальні рекомендації.

Задачі дослідження розкриваються в наступних підрозділах, які відповідають структурі роботи. Назви цих задач повинні відповідати назвам підрозділів та відображати послідовність етапів дослідження.

Перша задача стосується аналізу властивостей об'єкту дослідження та практичних задач, які виникають у контексті побудови рекомендацій для навігації з використанням комп'ютерного зору. Необхідно детально дослідити структуру процесу побудови рекомендацій, ідентифікувати його ключові компоненти, вхідні та вихідні дані кожного етапу, обмеження та вимоги до системи. Результатом вирішення цієї задачі має бути схема об'єкту дослідження, опис практичних задач, що вирішуються на об'єкті дослідження, та перелік задач, які додатково потрібно вирішувати для забезпечення ефективної роботи системи.

Друга задача присвячена аналізу методів вирішення задачі або аналізу моделей задачі чи об'єкту дослідження. Необхідно провести систематичний огляд існуючих підходів до розпізнавання об'єктів на відеозображеннях, методів трекінгу та прогнозування траєкторій, методів побудови рекомендацій у навігаційних системах, методів інтеграції даних від різних джерел. Для кожного методу потрібно визначити умови, в яких він працює, як ці методи

відрізняються за точністю, швидкістю, ресурсними вимогами тощо. Результатом має бути узагальнена таблиця або список з порівняльними характеристиками методів, що включає їх переваги, недоліки та ключові властивості.

Третя задача стосується аналізу методів, що вирішують звужену задачу в конкретній ситуації. Необхідно детально дослідити гібридні методи, які можуть ефективно працювати в умовах холодного старту або зміни преференцій користувачів, використовуючи дані від системи комп'ютерного зору. Для цієї задачі використовуються дані про демографічні характеристики користувачів, географічні особливості, дані від системи розпізнавання відеозображень, дані про актуальні події в даній місцевості, історичні дані про завантаженість маршрутів. Результатом має бути таблиця або список з методами та їх властивостями, з обов'язковим акцентом на недоліках методів, які потрібно усунути для ефективного вирішення практичних задач.

Четверта задача полягає у формулюванні короткого резюме попередніх підрозділів, що включає об'єкт дослідження, предмет дослідження, проблему дослідження та конкретні задачі дослідження. Це резюме має чітко визначити межі дослідження та очікувані результати.

На основі виявлених проблем та недоліків існуючих методів формулюється науковий результат дослідження. Ключовим етапом є розробка удосконаленого методу або моделі, що вирішує проблему, сформульовану у підрозділі постановки задачі дослідження. Цей метод має забезпечувати ефективну інтеграцію даних від системи комп'ютерного зору з традиційними джерелами інформації для побудови персоналізованих рекомендацій щодо вибору маршруту в умовах динамічних змін дорожньої обстановки та еволюції переваг користувачів.

Удосконалення методу передбачає декілька ключових аспектів. По-перше, необхідно розробити ефективний механізм виявлення змін у преференціях користувачів на основі аналізу їх поведінки та даних від системи комп'ютерного зору. Система повинна ідентифікувати ситуації, коли

користувач переходить у стан, аналогічний холодному старту, через зміну обставин.

По-друге, потрібно створити метод інтеграції візуальної інформації про дорожню обстановку з іншими типами даних на семантичному рівні, що дозволить системі формувати узгоджене уявлення про поточну ситуацію та прогнозувати її розвиток.

По-третє, необхідно розробити адаптивну модель переваг користувача, яка може швидко оновлюватися на основі нової інформації, враховуючи як явні сигнали від користувача, так і неявні індикатори зміни його потреб.

По-четверте, слід створити метод генерації рекомендацій, який балансує між дослідженням нових варіантів та використанням перевіреної інформації, адаптуючись до рівня невизначеності в моделі переваг користувача.

По-п'яте, необхідно забезпечити стійкість системи до помилок у даних від системи комп'ютерного зору та можливих адверсальних впливів, використовуючи методи верифікації та узгодження інформації з різних джерел.

Технологія, що імплементує розроблений підхід, має включати послідовність кроків з урахуванням особливостей та неточностей вхідних даних, специфіки існуючого інструментарію розробки. Опис інформаційної технології повинен деталізувати послідовність робіт від обробки вхідних даних і до кінцевого результату, представлення у формі діаграми або блок-схеми з поясненнями кожного етапу та прикладами.

Імплементация технології передбачає вибір конкретних інструментів та фреймворків для реалізації кожного компонента системи. Для модуля комп'ютерного зору доцільно використовувати сучасні фреймворки глибокого навчання, такі як PyTorch або TensorFlow, з готовими реалізаціями актуальних архітектур знаходження об'єктів. Для модуля рекомендацій можуть бути застосовані спеціалізовані бібліотеки, що підтримують різні типи рекомендаційних алгоритмів.

Програмна реалізація та експериментальна перевірка мають продемонструвати працездатність розробленого методу та технології.

Необхідно реалізувати модуль, який втілює удосконалений підхід, провести його тестування на реальних або синтетичних даних, що відповідають характеристикам цільового застосування. Експериментальна перевірка повинна включати детальний опис вхідних даних, які були використані для тестування, методів їх збору або генерації, характеристик датасету. Результати експериментів мають бути представлені у формі таблиць та графіків, що дозволяють порівняти продуктивність розробленого методу з існуючими підходами за ключовими метриками.

Порівняння отриманих результатів із відомими підходами повинно бути проведено коректно, з використанням однакових умов тестування та метрик оцінювання. Необхідно продемонструвати, в яких аспектах удосконалений метод перевершує існуючі підходи, а також чесно вказати на обмеження та ситуації, де він може поступатися альтернативним рішенням. Очікуваним результатом дослідження є створення методу та технології побудови рекомендацій для навігаційних систем, які ефективно використовують дані від системи комп'ютерного зору для адаптації до динамічних змін дорожньої обстановки та еволюції переваг користувачів, забезпечуючи високу якість рекомендацій навіть в умовах обмеженої історичної інформації про користувача.

## **2 РОЗРОБКА МЕТОДУ АДАПТИВНИХ РЕКОМЕНДАЦІЙ З СЕМАНТИЧНОЮ ІНТЕГРАЦІЄЮ ВІЗУАЛЬНИХ ДАНИХ**

### **2.1 Концептуальна модель багаторівневої системи з виявленням еволюції переваг користувачів**

На основі проведеного аналізу існуючих методів та виявлених недоліків у підрозділі 1.3 пропонується комплексний підхід до вирішення задачі побудови рекомендацій для навігаційних систем в умовах холодного старту з використанням даних від системи комп'ютерного зору. Цей підхід базується на інтеграції сучасних методів глибокого навчання для аналізу відеозображень з адаптивними алгоритмами побудови рекомендацій, що враховують динамічні зміни як дорожньої обстановки, так і переваг користувачів [9-10].

Концептуальна основа запропонованого підходу полягає у створенні багаторівневої системи, яка здійснює паралельну обробку різнорідних потоків інформації з їх подальшим семантичним злиттям. На нижньому рівні працюють спеціалізовані модулі для обробки кожного типу даних: модуль комп'ютерного зору для аналізу відеопотоку, модуль контекстної інформації для обробки даних про погоду, дорожні події та завантаженість доріг, модуль профілювання користувача для аналізу його історії переміщень та явних переваг. Ключовою інновацією є введення проміжного рівня семантичного представлення, на якому різнорідні дані перетворюються у уніфіковані векторні представлення в спільному просторі ознак. Це дозволяє системі виявляти складні кореляції між різними типами інформації та формувати цілісне уявлення про поточну ситуацію. Наприклад, виявлення великої кількості транспортних засобів на певній ділянці дороги може корелювати з інформацією про дорожню подію з інтернет-джерел, що підвищує впевненість системи у наявності затору.

Для вирішення проблеми холодного старту запропоновано використовувати багатоаспектний підхід, що комбінує три основні стратегії.

Перша стратегія базується на активному навчанні, коли система цілеспрямовано пропонує користувачеві різноманітні типи маршрутів для швидкого виявлення його переваг. Друга стратегія використовує трансферне навчання від схожих користувачів, ідентифікованих за демографічними характеристиками та початковими налаштуваннями. Третя стратегія залучає дані від системи комп'ютерного зору для виявлення неявних сигналів про переваги користувача.

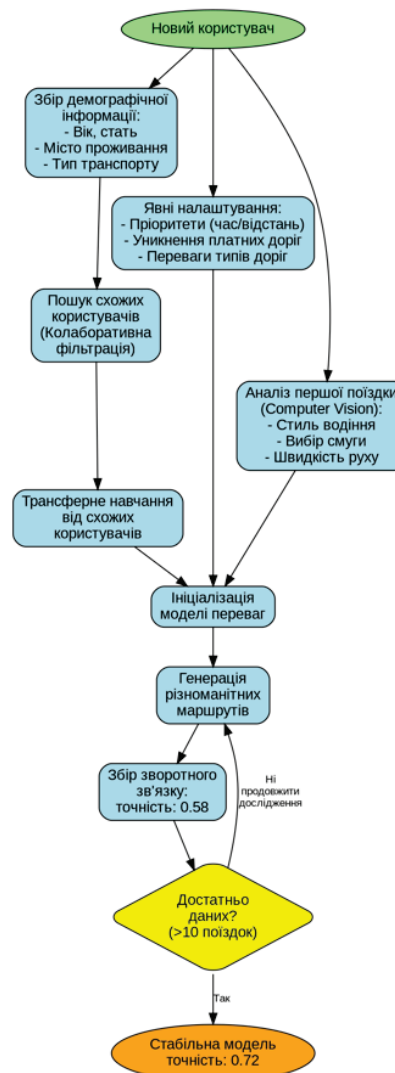


Рисунок 2.1 – Стратегія вирішення холодного старту

Система комп'ютерного зору може надавати цінну інформацію навіть без явної історії переміщень користувача. Аналіз типу транспортного засобу,

стилю водіння, реакції на дорожню обстановку дозволяє системі формувати початкові гіпотези про переваги користувача. Наприклад, якщо водій систематично обирає ліву смугу і підтримує високу швидкість за сприятливих умов, система може припустити його схильність до швидких маршрутів навіть за наявності додаткових кілометрів.

Для виявлення змін у перевагах існуючих користувачів застосовується механізм моніторингу відхилень від очікуваної поведінки [11-12]. Система постійно порівнює фактичні вибори користувача з прогнозами, що генеруються поточною моделлю його переваг. Систематичні відхилення вказують на можливу зміну переваг та активізують механізм адаптації моделі. При цьому важливо відрізнити випадкові відхилення, викликані специфічними обставинами конкретної поїздки, від справжніх змін у довгострокових перевагах.

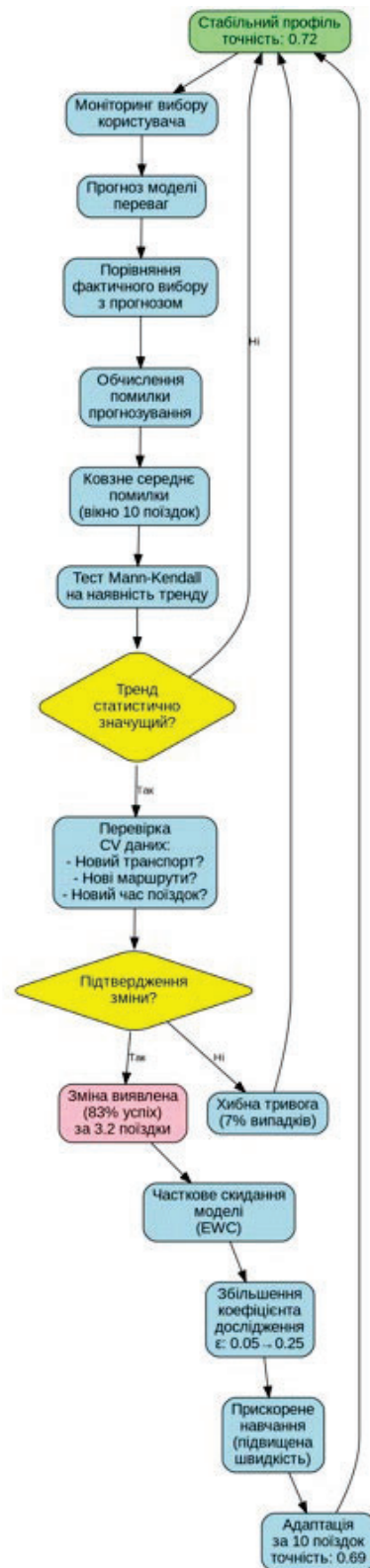


Рисунок 2.2 – Механізм виявлення змін переваг користувача

Дані від системи комп'ютерного зору відіграють ключову роль у розпізнаванні змін переваг. Якщо користувач раптово починає регулярно

використовувати новий тип транспорту, виявлений системою розпізнавання, це може сигналізувати про фундаментальну зміну його транспортних потреб. Аналогічно, зміна часових патернів переміщень, виявлена через відстеження об'єктів у різний час доби, може вказувати на зміну графіку роботи або способу життя.

Інтеграція візуальної інформації про дорожню обстановку з іншими джерелами даних здійснюється на основі байєсівського підходу до злиття інформації. Кожне джерело даних асоціюється з оцінкою його надійності та актуальності для поточної ситуації. Система комп'ютерного зору має перевагу в плані актуальності, оскільки надає інформацію про поточний стан дороги в режимі реального часу, але може мати обмеження щодо охоплення всього маршруту [13]. Навпаки, інформація з інтернет-сервісів може охоплювати весь регіон, але може мати певну затримку або бути неточною для конкретної ділянки. Байєсівський підхід дозволяє оптимально комбінувати ці джерела, надаючи більшу вагу візуальній інформації для ділянок, які камера безпосередньо спостерігає, і використовуючи інтернет-дані для решти маршруту.

Темпоральна динаміка дорожньої ситуації моделюється за допомогою рекурентних нейронних мереж, які навчаються передбачати еволюцію стану доріг у часі. Система аналізує не тільки поточний кадр відеопотоку, але й послідовність попередніх кадрів, виявляючи тренди у зміні завантаженості, швидкості руху транспорту та інших параметрів. Це дозволяє прогнозувати, чи затор розсіюється або, навпаки, наростає, що критично важливо для вибору оптимального часу відправлення або альтернативного маршруту.

Адаптивна модель переваг користувача реалізується як ансамбль декількох спеціалізованих моделей, кожна з яких фокусується на певному аспекті переваг. Одна модель відповідає за критерії вибору маршруту в нормальних умовах, інша за поведінку в умовах обмеженого часу, третя за переваги щодо типу доріг та пейзажів. Вагові коефіцієнти моделей в ансамблі динамічно змінюються залежно від контексту поточної поїздки.

Контекст поїздки визначається на основі комбінації явних сигналів від користувача, таких як вказівка терміновості поїздки, та неявних індикаторів, що виводяться з даних комп'ютерного зору та історичних патернів. Наприклад, якщо користувач зазвичай їде на роботу о восьмій ранку і система виявляє, що сьогодні він виїхав значно пізніше, можна припустити підвищену цінність часу для цієї конкретної поїздки. Генерація рекомендацій здійснюється на основі модифікованого алгоритму багатокритеріальної оптимізації, який враховує не тільки традиційні критерії, такі як час та відстань, але й додаткові фактори, отримані від системи комп'ютерного зору. До таких факторів належать поточна завантаженість доріг, виявлені перешкоди, складність дорожньої обстановки, що може вплинути на когнітивне навантаження водія [14]. Важливим аспектом є балансування між експлуатацією наявних знань про переваги користувача та дослідженням нових варіантів. Для нових користувачів або у випадку виявлення змін у перевагах система збільшує компоненту дослідження, пропонуючи більше різноманітних маршрутів. Для користувачів зі стабільними та добре вивченими перевагами система фокусується на експлуатації, пропонуючи перевірені оптимальні рішення.

Стійкість до помилок у даних від системи комп'ютерного зору забезпечується через механізм верифікації та узгодження з альтернативними джерелами інформації. Якщо візуальні дані суперечать інформації з інших надійних джерел, система знижує довіру до поточних результатів розпізнавання та може активізувати додаткові перевірки. Крім того, використовуються методи виявлення аномалій для ідентифікації потенційно помилкових фіксацій.

Захист від адверсальних атак реалізується на кількох рівнях. На рівні вхідних даних застосовуються методи передобробки зображень для виявлення та нейтралізації адверсальних збурень. На рівні моделі використовується ансамблевий підхід з різними архітектурами, що ускладнює успішну атаку. На рівні рекомендацій застосовується перехресна перевірка з альтернативними джерелами даних для виявлення аномальних результатів.

Загальна схема запропонованого підходу включає наступні ключові етапи. На етапі підготовки вхідних даних відбувається захоплення відеопотоку, попередня обробка зображень, фільтрація шумів та корекція спотворень. Етап виділення ознак включає застосування глибоких згорткових мереж для виявлення та класифікації об'єктів, трекінгу їх руху, виділення семантичних характеристик сцени. Етап перевірки умов додаткових обмежень включає верифікацію даних комп'ютерного зору з іншими джерелами, виявлення потенційних помилок та аномалій, оцінку надійності кожного джерела інформації для поточної ситуації. Етап включення нового індикатора передбачає інтеграцію нової інформації у семантичне представлення сцени, оновлення моделі дорожньої обстановки, корекцію прогнозів її еволюції.

Узагальнена послідовність кроків вирішення задачі починається з ініціалізації системи, де завантажуються попередньо навчені моделі комп'ютерного зору, ініціалізується профіль користувача на основі доступної інформації, налаштовуються параметри системи відповідно до контексту використання. Якщо користувач новий, застосовуються стратегії холодного старту: активне опитування переваг, використання демографічної інформації, початкове дослідження через різноманітні рекомендації. Далі відбувається постійний моніторинг відеопотоку та інших джерел даних. Система комп'ютерного зору обробляє кадри в режимі реального часу, виявляючи та класифікуючи об'єкти, відстежуючи їх рух, оцінюючи щільність трафіку та швидкість потоку. Паралельно здійснюється запит даних з інтернет-сервісів про погоду, дорожні події, історичну завантаженість для планованого часу поїздки.

Етап семантичної інтеграції об'єднує різноманітні дані у єдине представлення. Візуальні ознаки від системи комп'ютерного зору, текстові описи подій, числові характеристики погодних умов перетворюються у вектори в спільному просторі ознак. Використовуються методи глибокого навчання для навчання спільного представлення, що дозволяє системі автоматично виявляти кореляції між різними типами даних. На основі

інтегрованого представлення поточної ситуації та моделі переваг користувача генерується набір кандидатів маршрутів. Використовуються модифіковані алгоритми пошуку на графах, що враховують не тільки статичні ваги ребер, але й динамічні характеристики, отримані від системи моніторингу. Кожен маршрут оцінюється за множиною критеріїв з урахуванням поточного контексту поїздки. Ранжування маршрутів здійснюється на основі багатокритеріальної функції корисності, параметри якої налаштовуються під конкретного користувача. Функція враховує як об'єктивні характеристики маршруту, так і суб'єктивні переваги користувача, виявлені в процесі взаємодії з системою. Особлива увага приділяється прогнозованій еволюції дорожньої ситуації: маршрут може бути оптимальним зараз, але стати неоптимальним до моменту прибуття користувача.

## 2.2 Удосконалений метод побудови рекомендацій з інтеграцією YOLOv11 та адаптивним профілюванням

Удосконалений метод побудови рекомендацій для навігаційних систем з інтеграцією комп'ютерного зору складається з трьох основних компонентів: модуля аналізу відеозображень, модуля адаптивного профілювання користувача та модуля генерації рекомендацій [15-17]. Кожен компонент реалізує специфічні алгоритми та використовує власні моделі машинного навчання, оптимізовані для конкретних підзадач.

Модуль аналізу відеозображень базується на архітектурі YOLOv11, модифікованій для специфічних потреб навігаційних систем. Основна модифікація полягає у додаванні спеціалізованої голови для оцінки щільності трафіку та прогнозування його динаміки. Стандартна архітектура YOLO фокусується на знаходженні окремих об'єктів, тоді як для навігації важливіша інформація про загальний стан дорожнього потоку.

Vasbcone мережа використовує ефективні блоки згортки з механізмами уваги, що дозволяє виділяти найбільш інформативні регіони зображення. Для дорожніх сцен найбільш важливими зазвичай є ділянки доріг та перехресть, тоді як фонові об'єкти, такі як будівлі або небо, несуть меншу інформаційну цінність. Механізм уваги автоматично навчається фокусуватися на релевантних областях. Шия мережі реалізує багатомасштабне злиття ознак, що критично важливо для виявлення об'єктів різних розмірів. Транспортні засоби на близькій відстані займають значну частину кадру, тоді як віддалені об'єкти можуть бути представлені лише декількома пікселями. Feature Pyramid Network дозволяє ефективно обробляти об'єкти на всіх масштабах з єдиною моделлю.

Голова розпізнавання генерує обмежувальні рамки, класи об'єктів та оцінки впевненості для кожної ідентифікації. Додатково, спеціалізована голова оцінки трафіку генерує карту щільності транспортного потоку та вектори руху для різних ділянок зображення. Ця інформація агрегується для формування інтегральних характеристик дорожньої обстановки.

Модуль трекінгу реалізує алгоритм DeepSORT з модифікаціями для покращення стійкості до довготривалих оклюзій. Основна ідея полягає у використанні моделі руху для прогнозування положення об'єктів під час їх тимчасового зникнення з поля зору. Коли об'єкт знову з'являється, система намагається асоціювати його з існуючими треками на основі як візуальної схожості, так і відповідності прогнозованому положенню.

Візуальна схожість оцінюється за допомогою мережі re-identification, яка навчається виділяти дискримінативні ознаки для різних транспортних засобів. Ці ознаки повинні бути стійкими до змін ракурсу, освітлення та часткових оклюзій. Використовується метрична мережа, навчена за допомогою triplet loss, що забезпечує малу відстань між представленнями одного об'єкта та велику відстань між різними об'єктами.

Прогнозування траєкторій здійснюється за допомогою рекурентної нейронної мережі з механізмом уваги до контексту сцени. Модель не тільки

аналізує історію руху конкретного об'єкта, але й враховує рух сусідніх об'єктів, топологію дороги, положення світлофорів та інші контекстуальні фактори. Це дозволяє робити більш точні прогнози, особливо на перехрестях та в інших складних ситуаціях. Модуль адаптивного профілювання користувача складається з трьох підмодулів: ініціалізації профілю, моніторингу поведінки та оновлення моделі переваг. Підмодуль ініціалізації відповідає за створення початкового профілю користувача в умовах обмеженої інформації. Для нових користувачів застосовується комбінація кількох стратегій.

Перша стратегія використовує демографічну інформацію та явні налаштування користувача для ініціалізації базових параметрів моделі переваг. Система запитує у користувача його загальні пріоритети: швидкість, економія палива, комфорт, безпека. Ці пріоритети використовуються для встановлення початкових вагових коефіцієнтів у функції корисності маршруту.

Друга стратегія базується на колаборативній фільтрації, де система ідентифікує схожих користувачів у базі даних та використовує їх переваги як початкове наближення. Схожість визначається за комбінацією демографічних характеристик, типу транспортного засобу, географічної зони активності. Модель переваг схожих користувачів адаптується під нового користувача через трансферне навчання.

Третя стратегія залучає дані від системи комп'ютерного зору для виявлення неявних сигналів про переваги. Аналіз першої поїздки може надати цінну інформацію: вибір смуги руху, швидкість, реакція на дорожню обстановку. Ці поведінкові індикатори використовуються для корекції початкової моделі переваг ще до накопичення значної історії переміщень.

Підмодуль моніторингу поведінки постійно відстежує вибори користувача та порівнює їх з прогнозами поточної моделі. Для кожної поїздки фіксується запропоновані альтернативи, вибір користувача, контекст поїздки, результуючі характеристики обраного маршруту. Ці дані використовуються для двох цілей: виявлення відхилень від очікуваної поведінки та навчання оновленої моделі.

Виявлення відхилень базується на статистичному аналізі послідовності вибору користувача. Обчислюється ковзне середнє та дисперсія міри узгодженості між вибором та прогнозом. Систематичне зростання розбіжності вказує на можливу зміну переваг. Для розрізнення справжніх змін преференцій від контекстуальних варіацій використовується аналіз патернів відхилень. Якщо відхилення систематично корелюють з певним контекстом, наприклад, часом доби або днем тижня, це вказує на контекстуальну варіацію, а не на зміну базових переваг. Навпаки, якщо відхилення спостерігаються незалежно від контексту, це сигналізує про фундаментальну зміну преференцій, що вимагає переініціалізації або значної адаптації моделі.

Підмодуль оновлення моделі переваг реалізує онлайн-навчання з використанням методу градієнтного спуску по міні-батчам. Після кожної поїздки модель отримує новий приклад для навчання: вектор ознак маршруту, контекст поїздки, бінарна мітка вибору користувача. Модель навчається передбачати ймовірність вибору кожного маршруту за даного контексту.

Для запобігання катастрофічному забуванню раніше навчених переваг використовується регуляризація на основі попередніх параметрів моделі. Elastic Weight Consolidation дозволяє системі навчатися на нових даних, зберігаючи при цьому знання, критично важливі для попередніх задач. Параметри, що мають високу важливість для попередніх вибору користувача, змінюються повільніше. Модуль генерації рекомендацій інтегрує інформацію від модуля комп'ютерного зору та модуля профілювання для формування персоналізованих рекомендацій щодо маршруту. Процес генерації починається з побудови графу дорожньої мережі з динамічними вагами ребер, що відображають поточну та прогнозовану дорожню обстановку.

Ваги ребер обчислюються як зважена комбінація декількох факторів: базового часу проїзду без перешкод, поточного коефіцієнта затримки на основі даних комп'ютерного зору, прогнозованого коефіцієнта затримки на момент прибуття користувача, історичного коефіцієнта для даного часу доби. Коефіцієнти вагової комбінації налаштовуються під рівень довіри до кожного

джерела даних. Формально інтегральна оцінка маршруту може бути подана у вигляді функції корисності:

$$C(R) = \sum_i w_i * f_i(R) + w_{sem} * f_{isem}(R_{current}),$$

де  $R$  – розглянутий маршрут;

$C(R)$  – функція вартості (оцінки) маршруту  $R$ ;

$f_i(R)$  – традиційні фактори маршруту (довжина, очікуваний час, витрата палива);

$w_i$  – ваги переваг користувача (визначаються навчанням);

$f_{isem}(R_{current})$  – фактор, що враховує семантичні дані поточної ділянки (наприклад, штраф за небезпеку/пробку);

$w_{sem}$  – вага впливу дорожньої ситуації;

$R_{current}$  – поточна ділянка маршруту, для якої враховуються семантичні дані;

$i$  – індекс традиційних факторів маршруту.

Пошук кандидатів маршрутів здійснюється за допомогою модифікованого алгоритму  $A^*$  з евристикою, що враховує не тільки відстань, але й прогнозовані умови руху. Генерується не один оптимальний маршрут, а множина Парето-оптимальних альтернатив, що представляють різні компроміси між конфліктуючими цілями: час, відстань, складність, живописність. Кожен маршрут характеризується вектором об'єктивних ознак: загальний час, відстань, кількість поворотів, типи доріг, прогнозована варіабельність часу прибуття, рівень когнітивного навантаження. Додатково, на основі даних комп'ютерного зору, оцінюються суб'єктивні характеристики: естетична привабливість траси, рівень стресу від водіння в поточних умовах.

Ранжування маршрутів здійснюється за допомогою персоналізованої функції корисності, що моделює переваги конкретного користувача. Функція навчається на історії вибору користувача за допомогою методів ранжуючого навчання. Використовується архітектура siamese мережі, що навчається

порівнювати пари маршрутів та передбачати, який з них користувач більш імовірно обере.

Важливою особливістю є врахування невизначеності в прогнозах моделі. Замість детерміністичного вибору найкращого маршруту система генерує розподіл ймовірностей над множиною кандидатів. Це дозволяє реалізувати стратегію *epsilon-greedy* для балансування між експлуатацією та дослідженням: з ймовірністю *epsilon* обирається не найкращий за прогнозом, а альтернативний маршрут.

Параметр *epsilon* динамічно адаптується залежно від рівня впевненості моделі в переважних користувача та стабільності дорожньої обстановки. Для нових користувачів або після виявлення змін у перевагах *epsilon* збільшується, стимулюючи дослідження. Для користувачів зі стабільними та добре вивченими перевагами *epsilon* зменшується, фокусуючи систему на експлуатації наявних знань.

Презентація рекомендацій користувачеві включає не тільки відображення запропонованих маршрутів на карті, але й пояснення причин їх вибору. Для кожного маршруту надається інформація про його ключові переваги та недоліки, прогнозований час прибуття з інтервалом невизначеності, фактори ризику, виявлені системою комп'ютерного зору. Пояснення генеруються на основі аналізу найбільш впливових ознак для ранжування даного маршруту. Використовуються методи інтерпретації моделей машинного навчання, такі як SHapley Additive exPlanations (SHAP) values, для ідентифікації факторів, що найбільше вплинули на оцінку корисності маршруту. Ці фактори перетворюються у зрозумілі текстові пояснення природною мовою. Математична формалізація удосконаленого методу включає опис основних компонентів та їх взаємодії. Нехай  $V$  позначає відеопотік, представлений послідовністю кадрів. Модуль комп'ютерного зору реалізує функцію відображення від кадру до набору виявлених об'єктів та оцінки стану трафіку. Кожне розпізнавання характеризується обмежувальною

рамкою, класом об'єкта, оцінкою впевненості та вектором візуальних ознак для трекінгу.

Модуль трекінгу асоціює розпізнавання між послідовними кадрами, формуючи траєкторії руху об'єктів. Траєкторія представляється як послідовність станів об'єкта у часі, де кожен стан включає положення, швидкість, прискорення. На основі траєкторій обчислюються агреговані характеристики транспортного потоку: середня швидкість, щільність, варіабельність.

Модуль профілювання користувача підтримує модель переваг у вигляді параметризованої функції корисності. Параметри моделі оновлюються онлайн на основі спостережуваного вибору користувача. Використовується байєсівський підхід, де параметри моделюються як випадкові величини з апостеріорним розподілом, що оновлюється після кожного спостереження.

Модуль генерації рекомендацій формулює задачу вибору маршруту як задачу багатокритеріальної оптимізації з урахуванням невизначеності. Множина допустимих маршрутів визначається топологією дорожньої мережі та поточними обмеженнями. Для кожного маршруту обчислюється вектор критеріїв якості та оцінюється очікувана корисність за моделлю переваг користувача. Оптимізаційна задача розв'язується з використанням модифікованого генетичного алгоритму, що ефективно працює з багатокритеріальними функціями на графах. Популяція маршрутів еволюціонує через операції схрещування та мутації, адаптовані для графових структур. Фітнес-функція враховує як об'єктивні критерії якості, так і передбачувану корисність для конкретного користувача.

Деталізація удосконаленого методу включає специфікацію алгоритмів для кожного ключового компонента. Алгоритм виявлення та класифікації об'єктів на основі YOLOv11 включає етапи попередньої обробки вхідного зображення, прямого проходу через згорткову мережу, постобробки розпізнавання для видалення дублікатів та фільтрації низькоякісних результатів. Попередня обробка включає нормалізацію інтенсивностей

пікселів, приведення зображення до стандартного розміру через інтерполяцію, опціонально аугментацію для підвищення стійкості. Прямий прохід генерує набір передбачень для кожної якірної рамки на різних масштабах. Постобробка застосовує non-maximum suppression для видалення перекриваючихся розпізнавань одного об'єкта.

Алгоритм трекінгу об'єктів використовує комбінацію фільтра Калмана для прогнозування руху та глибокої метричної мережі для асоціації виявлених об'єктів. На кожному кадрі спочатку прогноуються положення існуючих треків. Потім обчислюється матриця відстаней між прогнозованими положеннями та новими розпізнаваннями, де відстань комбінує геометричну близькість та візуальну схожість.

Задача асоціації формулюється як задача призначення і розв'язується за допомогою угорського алгоритму. Ідентифікації, які не були асоційовані з існуючими треками, ініціюють нові треки. Треки, які не отримали відповідних фіксацій протягом заданого періоду, видаляються як завершені. Механізм реідентифікації дозволяє відновлювати треки після довготривалих оклюзій.

Алгоритм оновлення моделі переваг користувача реалізує онлайн варіант градієнтного бустінгу для ранжування. Після кожної поїздки генерується навчальний приклад у вигляді пари маршрутів: обраний користувачем та один з відхилених альтернатив. Модель навчається присвоювати вищий ранг обраному маршруту. Функція втрат базується на попарній різниці рангів з логістичною регуляризацією. Градієнт функції втрат обчислюється ефективно завдяки розрідженості вхідних даних. Оновлення параметрів здійснюється з адаптивною швидкістю навчання, що зменшується для стабільних параметрів та залишається високою для параметрів, що активно змінюються.

### **3 ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПОБУДОВИ НАВІГАЦІЙНИХ РЕКОМЕНДАЦІЙ В УМОВАХ ОБМЕЖЕНОЇ АПРІОРНОЇ ІНФОРМАЦІЇ**

#### **3.1 Десятиступеневий процес формування персоналізованих рекомендацій маршрутів**

Інформаційна технологія побудови рекомендацій для навігаційних систем з інтеграцією комп'ютерного зору представляє собою комплексний підхід до обробки різнорідних потоків даних та формування персоналізованих рекомендацій щодо вибору маршруту. Технологія реалізує удосконалений метод, описаний у розділі 2, через послідовність взаємопов'язаних етапів обробки інформації.

Загальна архітектура інформаційної технології базується на багат шаровій моделі обробки даних [18-20]. Нижній шар відповідає за збір та первинну обробку вхідних даних з різних джерел: відеопотоку від камер, даних геолокації від GPS-приймачів, інформації про дорожні події з інтернет-сервісів, історичних даних про переміщення користувачів. Кожне джерело даних обробляється спеціалізованим підмодулем, оптимізованим для конкретного типу інформації. Середній шар реалізує семантичну інтеграцію даних, перетворюючи різнорідні представлення у уніфікований формат векторних ознак у спільному просторі. На цьому рівні відбувається злиття візуальної інформації від системи комп'ютерного зору з контекстними даними про дорожню обстановку, географічні особливості місцевості та профіль користувача. Використовуються методи глибокого навчання для автоматичного виявлення складних залежностей між різними типами даних.

Верхній шар здійснює формування та ранжування рекомендацій на основі інтегрованого представлення поточної ситуації та моделі переваг користувача. Цей шар включає компоненти пошуку оптимальних маршрутів на графі дорожньої мережі, оцінювання корисності альтернативних варіантів,

генерації пояснень для рекомендацій. Результатом роботи є ранжований список маршрутів з додатковою інформацією про їх характеристики та причини рекомендації.

Перший етап технології полягає в ініціалізації системи та завантаженні необхідних компонентів. На цьому етапі завантажуються попередньо навчені моделі глибокого навчання для розпізнавання об'єктів, трекінгу, оцінки трафіку. Моделі можуть бути представлені у різних форматах залежно від цільової платформи: ONNX для кросплатформенної сумісності, Tensor Runtime (TensorRT) для оптимізації на Graphics Processing Unit (GPU) NVIDIA, Core Machine Learning (CoreML) для пристроїв Apple.

Завантаження моделей включає не тільки читання параметрів мережі, але й ініціалізацію обчислювальних ресурсів. Для моделей комп'ютерного зору критично важливим є ефективне використання графічних прискорювачів. Система автоматично виявляє доступні апаратні ресурси та вибирає оптимальну стратегію розподілу обчислень між Central Processing Unit (CPU) та GPU. За наявності декількох GPU застосовується паралелізація обробки відеопотоків. Паралельно з завантаженням моделей відбувається ініціалізація профілю користувача [21]. Для нових користувачів створюється початковий профіль на основі демографічної інформації та явних налаштувань, зібраних під час реєстрації. Для існуючих користувачів завантажуються збережена модель переваг разом з історією їх переміщень. Історичні дані можуть бути агреговані для зменшення обсягу пам'яті, зберігаючи тільки статистично значущі патерни поведінки.

Другий етап стосується захоплення та попередньої обробки відеопотоку. Відеопотік може надходити від різних джерел: вбудованих камер мобільних пристроїв, відеореєстраторів, камер спостереження дорожньої інфраструктури. Різні джерела мають різні характеристики якості, роздільної здатності, частоти кадрів, що вимагає адаптивної обробки.

Попередня обробка відеопотоку включає декілька операцій. Спочатку відбувається декодування стислого відеопотоку у послідовність кадрів у

форматі Red, Green, Blue (RGB) або Y (Luminance), U (Blue-difference Chrominance), V (Red-difference Chrominance) (YUV). Для зменшення обчислювальних витрат може застосовуватися часова субдискретизація, коли обробляється не кожен кадр, а кожен другий або третій. Вибір коефіцієнта субдискретизації залежить від швидкості руху об'єктів та доступних обчислювальних ресурсів.

Наступним кроком є просторова нормалізація зображень. Кадри приводяться до стандартного розміру, очікуваного моделлю виявлення об'єктів, зазвичай це квадратні зображення розміром від 416x416 до 1024x1024 пікселів. Використовується інтерполяція для зміни розміру з мінімальними втратами деталей. Крім того, застосовуються методи корекції освітлення для нормалізації яскравості та контрасту зображень, знятих в різних умовах.

Фільтрація шумів здійснюється за допомогою адаптивних фільтрів, які зберігають чіткість країв об'єктів при зменшенні випадкових флуктуацій інтенсивностей. Для відеопотоків, знятих в русі, важливою є стабілізація зображення для компенсації вібрацій камери. Застосовуються алгоритми оптичного потоку для оцінки руху камери та його компенсації через геометричні перетворення кадрів.

Третій етап присвячений знаходженню та класифікації об'єктів на попередньо оброблених кадрах. Застосовується модель YOLOv11, яка обробляє кожен кадр незалежно, генеруючи набір ідентифікацій. Кожна фіксація характеризується координатами обмежувальної рамки, класом об'єкта, оцінкою впевненості моделі у правильності класифікації. Процес розпізнавання починається з прямого проходу зображення через згорткову мережу. Backbone мережа виділяє ієрархічні візуальні ознаки на різних рівнях абстракції. Низькорівневі шари виявляють прості патерни, такі як края та текстури, середні шари комбінують їх у складніші форми, високорівневі шари формують семантичні представлення цілих об'єктів.

Neck мережі об'єднує ознаки з різних рівнів через механізм Feature Pyramid Network, створюючи багатомасштабне представлення. Це дозволяє

ефективно виявляти об'єкти різних розмірів: великі об'єкти на низьких рівнях пірамід, малі об'єкти на високих рівнях. Головки виявлення застосовуються до кожного рівня пірамід, генеруючи передбачення для якірних рамок різних масштабів та співвідношень сторін.

Постобробка розпізнавання включає фільтрацію результатів з низькою впевненістю та застосування non-maximum suppression для видалення дублюючих знаходжень одного об'єкта. Non-Maximum Suppression (NMS) порівнює перекриття обмежувальних рамок та зберігає тільки знаходження з найвищою впевненістю серед групи перекриваючихся рамок. Результатом є очищений набір виявлень, готовий для подальшої обробки.

Четвертий етап стосується трекінгу об'єктів між послідовними кадрами відеопотоку. Система повинна підтримувати ідентичність кожного об'єкта в часі, відстежуючи його рух та зміну характеристик. Це дозволяє будувати траєкторії руху транспортних засобів, аналізувати їх швидкість та прискорення, прогнозувати майбутнє положення.

Трекінг реалізується за допомогою комбінації фільтра Калмана для прогнозування руху та глибокої метричної мережі для верифікації ідентичності. Для кожного існуючого треку фільтр Калмана прогнозує його положення на наступному кадрі на основі попередніх спостережень. Прогноз враховує швидкість та прискорення об'єкта, моделюючи його рух як лінійну динамічну систему з гаусовим шумом.

Нові розпізнавання асоціюються з існуючими треками на основі комбінованої метрики близькості. Геометрична компонента метрики оцінює відстань між прогнозованим положенням треку та положенням ідентифікації. Візуальна компонента використовує глибоку мережу для порівняння зовнішнього вигляду об'єктів. Мережа витягує дескриптор з області зображення всередині обмежувальної рамки, і дескриптори порівнюються за косинусною відстанню. Задача асоціації формулюється як задача призначення у двочастковому графі, де одна частка відповідає існуючим трекам, інша новим знаходженням. Ваги ребер визначаються комбінованою метрикою близькості.

Оптимальне призначення знаходиться за допомогою угорського алгоритму, який мінімізує загальну вартість асоціацій. Фіксації, не асоційовані з жодним треком, ініціюють нові треки, треки без відповідних розпізнавань продовжують існувати у режимі прогнозування.

П'ятий етап присвячений аналізу траєкторій та оцінці стану дорожнього потоку. На основі побудованих треків обчислюються агреговані характеристики транспортного руху для різних ділянок дороги. Кожна ділянка характеризується середньою швидкістю транспортних засобів, щільністю потоку, варіабельністю швидкості, наявністю затримок або зупинок.

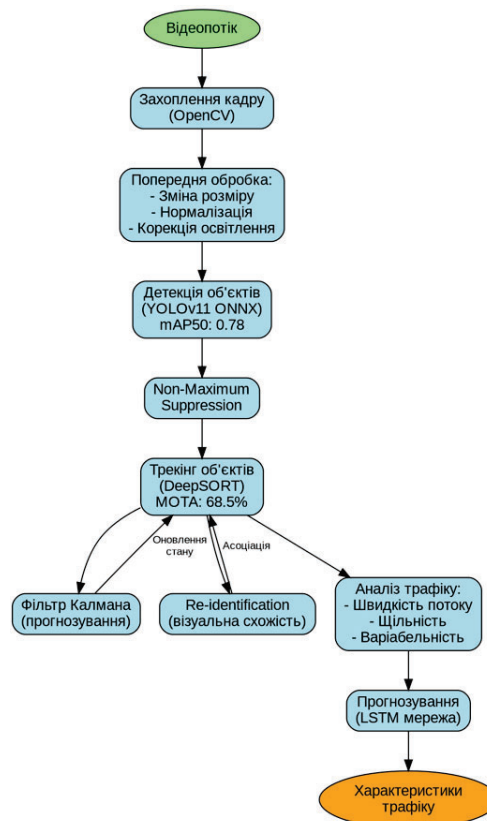


Рисунок 3.1 – Процес обробки відеозображень

Для обчислення середньої швидкості використовуються швидкості всіх активних треків на поточному кадрі, що належать до даної ділянки дороги. Швидкість кожного треку оцінюється як похідна його положення по часу,

обчислена через кінцеві різниці. Застосовується медіанна фільтрація для зменшення впливу викидів, викликаних помилками виявлення або трекінгу.

Щільність потоку визначається як кількість транспортних засобів на одиницю довжини дороги. Для її оцінки використовується перспективне перетворення для переходу від координат зображення до реальних просторових координат. Це дозволяє враховувати геометричні спотворення, викликані кутом огляду камери. Щільність обчислюється окремо для кожної смуги руху, що дозволяє виявляти локальні затори. Варіабельність швидкості характеризує однорідність транспортного потоку. Висока варіабельність може вказувати на турбулентний режим руху з частими прискореннями та гальмуваннями, що знижує комфорт та безпеку. Обчислюється стандартне відхилення швидкостей всіх транспортних засобів на ділянці. Додатково аналізуються паттерни прискорень для виявлення стоп-енд-гоу режиму, характерного для заторів.

Шостий етап стосується прогнозування еволюції дорожньої обстановки. Система не тільки аналізує поточний стан, але й передбачає його зміни в найближчому майбутньому. Це критично важливо для вибору оптимального маршруту, оскільки умови на дорозі можуть значно змінитися за час руху користувача. Прогнозування здійснюється за допомогою рекурентної нейронної мережі, яка навчена на історичних даних про еволюцію дорожніх ситуацій. Мережа приймає на вхід послідовність спостережень характеристик транспортного потоку за попередні часові інтервали та генерує прогноз для наступних інтервалів. Використовується архітектура Long Short-Term Memory (LSTM) для ефективного моделювання довгострокових залежностей у часових рядах.

Вхідні дані для прогнозування включають не тільки характеристики трафіку, але й додаткові контекстуальні фактори: час доби, день тижня, погодні умови, календарні події. Ці фактори суттєво впливають на динаміку транспортних потоків. Наприклад, вранці спостерігається наростання завантаженості на дорогах до центру міста, ввечері зворотній процес. Модель

генерує ймовірнісний прогноз у вигляді розподілу можливих станів дорожньої обстановки. Це дозволяє врахувати невизначеність прогнозу при плануванні маршруту. Для критичних поїздок, де затримка неприйнятна, система може обирати більш консервативні маршрути з нижчою варіабельністю часу прибуття, навіть якщо очікуваний час дещо вищий.

Сьомий етап присвячений інтеграції даних від системи комп'ютерного зору з інформацією з інших джерел. Візуальні дані надають актуальну інформацію про стан ділянок дороги, які безпосередньо спостерігаються камерою, але не покривають весь регіон. Інтернет-сервіси надають інформацію про дорожні події, аварії, ремонтні роботи в усьому регіоні, але можуть мати затримку в оновленні даних.

Інтеграція здійснюється на основі байєсівського підходу, де кожне джерело даних асоціюється з оцінкою його надійності та актуальності. Для ділянок дороги, що спостерігаються камерою, візуальна інформація має вищу вагу як більш актуальна та детальна. Для решти регіону використовуються дані з інтернет-сервісів, скориговані на основі історичних патернів їх точності. Виявлення протиріч між різними джерелами даних активізує додаткові перевірки. Якщо візуальні дані вказують на вільний рух на ділянці, а інтернет-сервіс повідомляє про затор, система може запитати додаткову інформацію або знизити впевненість у рекомендаціях для цієї ділянки. Механізм узгодження намагається знайти пояснення розбіжностей, наприклад, через різницю в часі спостережень.

Восьмий етап стосується роботи з профілем користувача та виявлення змін у його перевагах. Система постійно моніторить вибори користувача та порівнює їх з прогнозами поточної моделі переваг. Систематичні відхилення можуть сигналізувати про зміну потреб користувача, що вимагає адаптації моделі.

Виявлення змін базується на статистичному аналізі послідовності помилок прогнозування. Обчислюється ковзне середнє помилки та його дисперсія. Якщо середня помилка систематично зростає протягом певного

періоду, це вказує на можливий дрейф у перевагах користувача. Для підтвердження зміни аналізується кореляція помилок з контекстуальними факторами.

Дані від системи комп'ютерного зору використовуються для виявлення неявних індикаторів змін. Якщо користувач починає регулярно використовувати новий тип транспортного засобу, виявлений через класифікацію об'єктів, це може вказувати на зміну транспортних потреб. Аналогічно, зміна часових патернів переміщень або географічних областей активності сигналізує про можливі життєві зміни. При виявленні значних змін активізується механізм переініціалізації профілю. Користувачеві може бути запропоновано оновити свої налаштування або підтвердити автоматично виявлені зміни. Модель переваг частково скидається до початкового стану, зберігаючи тільки найбільш стабільні компоненти. Система переходить у режим активного дослідження для швидкого навчання нових переваг.

Дев'ятий етап присвячений генерації кандидатів маршрутів на основі інтегрованого представлення дорожньої обстановки. Дорожня мережа представляється у вигляді зваженого орієнтованого графу, де вершини відповідають перехрестям або важливим точкам, ребра дорожнім сегментам. Ваги ребер динамічно оновлюються на основі поточних та прогнозованих характеристик трафіку.



Рисунок 3.2 – Процес генерації та ранжування маршрутів

Вага ребра обчислюється як функція декількох факторів. Базова компонента відображає час проїзду сегменту за ідеальних умов без перешкод. Динамічна компонента враховує поточний коефіцієнт затримки, отриманий від системи моніторингу трафіку. Прогнозована компонента оцінює очікуваний стан на момент прибуття користувача на цей сегмент. Стохастична компонента моделює невизначеність прогнозу.

Пошук маршрутів здійснюється за допомогою модифікованого алгоритму  $A^*$  з евристичною функцією, що враховує не тільки відстань до цілі, але й очікувані умови руху. Евристика оцінює нижню межу часу досягнення цілі, припускаючи рух з максимально можливою швидкістю без затримок. Це гарантує допустимість евристики та оптимальність знайденого маршруту. Для генерації множини альтернативних маршрутів використовується ітеративний підхід. Після знаходження оптимального маршруту тимчасово збільшуються ваги його ребер та шукається наступний оптимальний маршрут. Процес повторюється до генерації заданої кількості різноманітних альтернатив. Додатково застосовується фільтрація для видалення маршрутів, що є незначними варіаціями вже знайдених.

Десятий етап стосується оцінювання та ранжування згенерованих маршрутів з урахуванням персональних переваг користувача. Кожен маршрут характеризується багатовимірним вектором ознак, що включає як об'єктивні метрики, так і оцінки суб'єктивних характеристик на основі даних комп'ютерного зору.

Об'єктивні метрики включають загальний час поїздки, довжину маршруту, кількість поворотів, типи доріг, прогнозовану дисперсію часу прибуття. Суб'єктивні характеристики оцінюються на основі візуального аналізу маршруту: естетична привабливість пейзажів, складність навігації, очікуване когнітивне навантаження на водія в поточних дорожніх умовах. Оцінка естетичної привабливості базується на аналізі зображень уздовж маршруту за допомогою моделі, навченої на датасеті з оцінками користувачів. Модель враховує наявність природних ландшафтів, архітектурних пам'яток,

водоїм та інших візуально привабливих об'єктів. Складність навігації оцінюється на основі кількості та типу поворотів, складності перехресть, якості дорожньої розмітки.

Когнітивне навантаження моделюється як функція завантаженості доріг, складності дорожньої обстановки, необхідності частих маневрів. Висока щільність транспортного потоку, виявлена системою комп'ютерного зору, збільшує оцінку когнітивного навантаження. Ця характеристика особливо важлива для користувачів, які цінують комфорт та зниження стресу від водіння. Ранжування маршрутів здійснюється за допомогою персоналізованої функції корисності, параметри якої налаштовані під конкретного користувача. Функція реалізується як глибока нейронна мережа, яка приймає на вхід вектор ознак маршруту та контекст поїздки, генерує на виході оцінку корисності. Мережа навчена на історії вибору користувача методом ранжуючого навчання.

### 3.2 Програмна реалізація на основі PyTorch, ONNX Runtime та мікросервісної архітектури

Практична реалізація описаної інформаційної технології здійснюється з використанням сучасних фреймворків та інструментів для машинного навчання та обробки даних. Вибір конкретних технологій обумовлений вимогами до продуктивності, портативності, підтримки апаратних прискорювачів та доступності готових реалізацій стандартних компонентів [22-25].

Основним фреймворком для реалізації моделей глибокого навчання обрано PyTorch завдяки його гнучкості, ефективності та широкій підтримці спільнотою. PyTorch надає зручний інтерфейс для побудови та навчання нейронних мереж, автоматичне диференціювання для обчислення градієнтів, оптимізовані реалізації стандартних шарів та операцій. Підтримка динамічних

обчислювальних графів спрощує реалізацію складних архітектур з умовною логікою.

Для інференсу моделей на виробничих системах використовується ONNX Runtime, що забезпечує кросплатформенну сумісність та оптимізовану продуктивність. Моделі, навчені у PyTorch, експортуються у формат ONNX, який підтримується на різних платформах та апаратних прискорювачах. ONNX Runtime автоматично застосовує низькорівневі оптимізації, такі як злиття операцій, квантизація, використання спеціалізованих інструкцій процесора.

Модуль захоплення та обробки відеопотоку реалізовано з використанням бібліотеки Open Source Computer Vision Library (OpenCV), яка надає ефективні інструменти для роботи з відео та зображеннями. OpenCV підтримує різні джерела відеопотоку: файли, мережеві камери, USB-пристрої. Бібліотека включає оптимізовані реалізації алгоритмів попередньої обробки зображень: зміна розміру, фільтрація, корекція освітлення, геометричні перетворення. Для паралельної обробки декількох відеопотоків використовується багатопоточність з модулем threading або багатопроцесність з модулем multiprocessing залежно від характеру навантаження. Операції обробки зображень, що активно використовують CPU, виконуються у окремих процесах для обходу обмежень Global Interpreter Lock у Python. Інференс моделей на GPU виконується у окремих потоках з синхронізацією через черги завдань [26].

Модель ідентифікації об'єктів базується на архітектурі YOLOv11, реалізація якої доступна у бібліотеці Ultralytics. Бібліотека надає готові попередньо навчені моделі різних розмірів, від компактних версій для мобільних пристроїв до великих версій для максимальної точності. Моделі можуть бути донавчені на спеціалізованих датасетах дорожніх сцен для покращення точності у цільовому домені. Процес донавчання включає підготовку анотованого датасету зображень дорожніх сцен з розміткою обмежувальних рамок навколо об'єктів інтересу. Використовуються публічні датасети, такі як Karlsruhe Institute of Technology and Toyota Technological

Institute (KITTI), BDD100K, Cityscapes, доповнені власними зібраними даними для врахування специфіки регіону. Анотації створюються у форматі YOLO, де кожен рядок описує один об'єкт через його клас та нормалізовані координати обмежувальної рамки.

Навчання моделі здійснюється з використанням техніки трансферного навчання. Модель, попередньо навчена на COCO датасеті, донавчається на спеціалізованих даних з пониженою швидкістю навчання для тонкого налаштування. Застосовуються техніки аугментації даних для збільшення різноманітності навчальної вибірки: випадкові повороти, масштабування, зміна яскравості та контрасту, горизонтальні відображення. Модуль трекінгу об'єктів реалізовано на основі алгоритму DeepSORT з використанням бібліотеки `deep_sort_realtime`. Бібліотека інтегрує фільтр Калмана для прогнозування руху об'єктів та глибоку мережу для верифікації ідентичності. Мережа `re-identification` навчена на датасеті транспортних засобів для виділення дискримінативних візуальних ознак.

Для підвищення стійкості трекінгу до довготривалих оклюзій модифіковано параметри фільтра Калмана, збільшивши максимальний час існування треку без відповідних виявлень. Додано механізм реідентифікації, що намагається асоціювати нові розпізнавання з треками, які були втрачені протягом останніх кадрів. Використовується більш строга перевірка візуальної схожості для запобігання помилковим асоціаціям. Модуль оцінки стану трафіку агрегує інформацію від системи трекінгу для обчислення характеристик транспортного потоку. Для кожної ділянки дороги підтримується структура даних з активними треками, які належать до цієї ділянки. Періодично, з заданою частотою, обчислюються агреговані метрики: середня швидкість, щільність, варіабельність, наявність аномалій [27].

Для переходу від координат зображення до реальних просторових координат використовується попередньо обчислена матриця перспективного перетворення. Матриця визначається через калібрування камери з використанням відомих опорних точок на дорозі. Калібрування може бути

виконане автоматично через виявлення дорожньої розмітки або вручну через вказівку відповідностей між точками зображення та реальними координатами.

Модуль прогнозування еволюції трафіку реалізовано у вигляді LSTM мережі, навченої на історичних даних про динаміку транспортних потоків. Архітектура мережі включає два шари LSTM з dropout для регуляризації, за якими слідує повнозв'язні шари для генерації прогнозу. Мережа навчена передбачувати характеристики трафіку на декілька часових кроків вперед на основі послідовності попередніх спостережень.

Навчання здійснюється на даних, зібраних від системи моніторингу протягом тривалого періоду. Дані агреговані з інтервалом від одної до п'яти хвилин залежно від динамічності трафіку на конкретній ділянці. Використовується метод ковзного вікна для формування навчальних прикладів: послідовність спостережень фіксованої довжини використовується як вхід, наступні значення як цільовий вихід. Модуль інтеграції даних реалізує байєсівське злиття інформації з різних джерел. Для кожної ділянки дороги підтримується апостеріорний розподіл над можливими станами трафіку. Розподіл оновлюється при надходженні нових спостережень згідно з правилом Байєса. Правдоподібність спостереження залежить від його джерела та поточного контексту.

Візуальні спостереження від системи комп'ютерного зору мають високу правдоподібність для ділянок, безпосередньо видимих камерою. Спостереження від інтернет-сервісів мають нижчу правдоподібність через можливу затримку або неточність. Історичні дані використовуються як апіорний розподіл, що модифікується на основі актуальних спостережень. Результатом інтеграції є оцінка поточного стану з інтервалом невизначеності.

Модуль профілювання користувача реалізовано як окремий сервіс з власною базою даних для зберігання профілів та історії взаємодій. База даних побудована на PostgreSQL з оптимізованими індексами для швидкого доступу до історичних записів. Для кожного користувача зберігається поточна модель

переваг у вигляді серіалізованих параметрів нейронної мережі, агрегована історія поїздок, демографічна інформація.

Модель переваг періодично оновлюється на основі нових спостережень про вибір користувача. Оновлення виконується асинхронно у фоновому режимі для уникнення затримок при обслуговуванні запитів. Використовується механізм версійності моделей, що дозволяє відкочувати зміни при виявленні деградації якості рекомендацій. Старі версії моделей зберігаються для можливості аналізу еволюції переваг користувача.

Модуль генерації маршрутів використовує бібліотеку NetworkX для роботи з графами та реалізації алгоритмів пошуку. Граф дорожньої мережі завантажується з OpenStreetMap через бібліотеку OSMnx, що надає зручний інтерфейс для отримання та обробки геопросторових даних. Граф включає всі типи доріг: автомагістралі, міські вулиці, локальні дороги.

Ваги ребер графу динамічно оновлюються на основі поточних оцінок стану трафіку. Для кожного ребра обчислюється очікуваний час проїзду як функція базового часу, поточного коефіцієнта затримки, прогнозованих умов. Використовується кешування для уникнення повторних обчислень для незмінених ділянок. Кеш інвалідується при отриманні нових спостережень про стан трафіку. Пошук маршрутів реалізовано через алгоритм A\* з пріоритетною чергою на основі бінарної купи. Евристична функція оцінює нижню межу часу досягнення цілі через відстань по прямій та максимально допустиму швидкість. Для генерації множини альтернативних маршрутів застосовується k-shortest paths алгоритм Йєна, що ітеративно знаходить наступні найкоротші маршрути з обмеженням на схожість з попередніми.

Модуль ранжування маршрутів використовує навчену нейронну мережу для оцінки корисності кожного маршруту для конкретного користувача. Мережа реалізована у PyTorch з архітектурою, що включає вкладення для категоріальних ознак, повнозв'язні шари для обробки числових ознак, механізм уваги для врахування контексту поїздки. Вихід мережі

нормалізується через softmax для отримання ймовірнісного розподілу над маршрутами.

Інтерфейс користувача реалізовано як веб-застосунок з використанням фреймворку React для фронтенду та FastAPI для бекенду. Фронтенд відображає карту з запропонованими маршрутами, використовуючи бібліотеку Leaflet для інтерактивної візуалізації. Кожен маршрут представлений кольоровою лінією з анотаціями ключових характеристик: час, відстань, прогнозована варіабельність. Бекенд надає Representational State Transfer (RESTful) Application Programming Interface (API) для взаємодії з різними модулями системи. Endpoint для запиту рекомендацій приймає координати початку та кінця, контекст поїздки, ідентифікатор користувача. Обробка запиту включає генерацію маршрутів, їх оцінку та ранжування, формування відповіді з детальною інформацією про кожен маршрут. Використовується асинхронна обробка для підтримки високої паралельності запитів.

Система розгорнута у мікросервісній архітектурі з використанням Docker для контейнеризації компонентів. Кожен основний модуль виконується у окремому контейнері: сервіс обробки відео, сервіс профілювання користувачів, сервіс генерації маршрутів, API gateway. Kubernetes використовується для оркестрації контейнерів, забезпечуючи автоматичне масштабування, балансування навантаження, відновлення після збоїв.

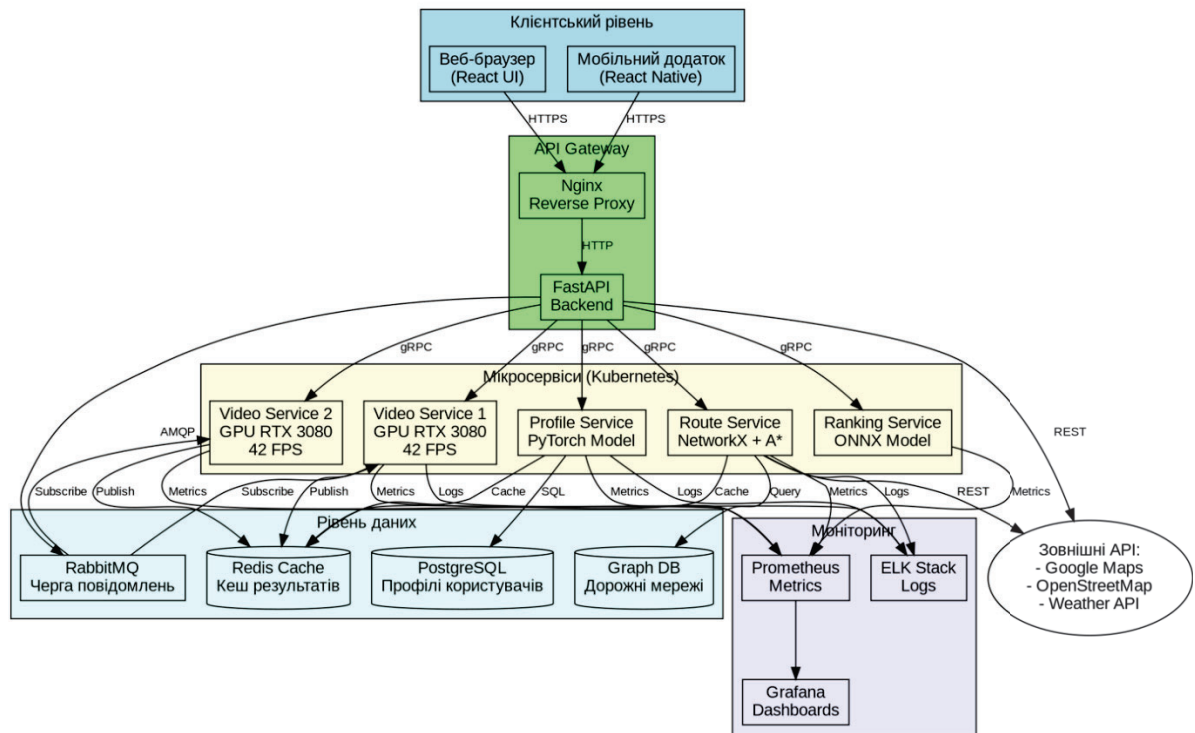


Рисунок 3.3 – Архітектура розгортання системи

Для обробки відеопотоків від багатьох користувачів використовується черга повідомлень RabbitMQ. Запити на обробку відео поміщаються у чергу, звідки вибираються воркерами для виконання. Це дозволяє ефективно розподіляти навантаження між декількома серверами обробки та забезпечує стійкість до перевантажень через буферизацію запитів у черзі. Моніторинг системи здійснюється через Prometheus для збору метрик та Grafana для візуалізації. Відстежуються ключові метрики продуктивності: затримка обробки запитів, throughput, використання ресурсів, частота помилок. Налаштовані алерти для критичних ситуацій, таких як перевищення порогів латентності або виснаження ресурсів. Логи централізовано збираються через ELK стек для аналізу та діагностики проблем.

## 4 ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА МЕТОДУ НА ДАТАСЕТАХ BDD100K ТА ВЛАСНИХ ЗАПИСАХ

### 4.1 Реалізація прототипу системи з модульною архітектурою

Програмна реалізація розробленого методу здійснена у вигляді модульної системи на мові Python з використанням фреймворків PyTorch для глибокого навчання та OpenCV для обробки відеозображень. Архітектура побудована за принципами слабкого зв'язування компонентів, що дозволяє незалежно розробляти, тестувати та оптимізувати окремі модулі. Модуль обробки відеозображень реалізовано через клас VideoProcessor, що інкапсулює функціональність захоплення відеопотоку, попередньої обробки кадрів, розпізнавання та трекінгу об'єктів. Клас використовує модель YOLOv11 у форматі ONNX для забезпечення кросплатформенної сумісності та оптимальної продуктивності [28-29]. Завантаження моделі здійснюється через ONNX Runtime з автоматичним вибором виконавчого провайдера: CUDA для систем з GPU NVIDIA або оптимізований CPU провайдер для систем без дискретної графіки.

Метод `process_frame` виконує повний цикл обробки одного кадру відеопотоку. Спочатку кадр проходить попередню обробку: приведення до стандартного розміру 640x640 пікселів через бікубічну інтерполяцію, нормалізація значень пікселів до діапазону від нуля до одиниці, транспонування осей для відповідності формату `batch, channels, depth, height, width` (NCHW). Після прямого проходу через модель результати постобробляються через фільтрацію фіксацій з впевненістю нижче порогового значення та застосування `non-maximum suppression` для видалення перекриваючихся розпізнавань одного об'єкта.

Трекінг об'єктів реалізовано через клас `ObjectTracker` на основі алгоритму DeepSORT. Клас підтримує список активних треків та забезпечує їх оновлення при надходженні нових знаходжень з кожного кадру. Метод `update`

приймає фіксації поточного кадру та повертає оновлений список треків з унікальними ідентифікаторами. Для кожного існуючого треку фільтр Калмана прогнозує його положення на поточному кадрі на основі моделі постійної швидкості. Асоціація розпізнавання з треками здійснюється через угорський алгоритм на основі комбінованої метрики, що враховує як геометричну близькість, так і візуальну схожість об'єктів.

Модуль аналізу трафіку реалізовано у класі `TrafficAnalyzer`, що агрегує інформацію від системи трекінгу для обчислення характеристик транспортного потоку. Клас підтримує розбиття дорожнього простору на сегменти, для кожного з яких періодично обчислюються метрики: середня швидкість руху через усереднення швидкостей всіх треків у сегменті, щільність потоку як кількість об'єктів на одиницю довжини дороги, варіабельність через стандартне відхилення швидкостей. Для коректного обчислення метрик у метричних одиницях використовується попередньо обчислена матриця гомографії для переходу від координат зображення до реальних просторових координат.

Модуль профілювання користувачів реалізовано як окремий сервіс `UserProfileService` з власною базою даних PostgreSQL. Сервіс надає методи для створення нових профілів, завантаження існуючих, оновлення моделей переваг. Модель переваг представлена класом `PreferenceModel` як нейронна мережа з вкладеннями для категоріальних ознак та повнозв'язними шарами для числових характеристик. Метод `update_model` реалізує онлайн-навчання через декілька кроків градієнтного спуску на основі нового прикладу вибору користувача. Використовується регуляризація `Elastic Weight Consolidation` для збереження важливих параметрів моделі при навчанні на нових даних.

Метод `detect_preference_change` аналізує послідовність помилок прогнозування моделі для виявлення можливих змін у перевагах. Обчислюється ковзне середнє помилки для останніх десяти поїздки та застосовується тест `Mann-Kendall` для перевірки наявності статистично

значущого зростаючого тренду. При виявленні значущого тренду метод повертає сигнал про необхідність адаптації моделі.

Модуль генерації маршрутів реалізовано у класі `RouteGenerator` з використанням бібліотеки `NetworkX` для роботи з графами. Граф дорожньої мережі завантажується з `OpenStreetMap` через бібліотеку `OSMnx` та зберігається у форматі орієнтованого графу. Метод `generate_routes` приймає координати початку та кінця, знаходить найближчі вершини графу та генерує набір альтернативних маршрутів [30-31]. Перший маршрут знаходиться через алгоритм  $A^*$  з евристикою на основі відстані та швидкості. Наступні альтернативи генеруються через ітеративне збільшення ваг ребер попередніх маршрутів.

Модуль ранжування маршрутів реалізовано у класі `RouteRanker`, що використовує навчену модель переваг для оцінки корисності кожного маршруту. Метод `rank_routes` обчислює вектор ознак для кожного маршруту, застосовує модель переваг користувача та сортує маршрути за спадною оцінкою корисності. Метод `generate_explanation` використовує бібліотеку `SHAP` для визначення найбільш впливових ознак на оцінку кожного маршруту та формує текстові пояснення. Інтеграція всіх модулів здійснюється через головний клас `NavigationSystem`, що реалізує патерн фасад. Метод `get_recommendations` координує роботу всіх компонентів: завантажує профіль користувача, запитує поточний стан трафіку від модуля обробки відео, генерує маршрути з урахуванням динамічних ваг, ранжує їх згідно з перевагами користувача, формує фінальну відповідь з рекомендаціями та поясненнями.

## 4.2 Формування тестових наборів даних з різними сценаріями використання

Для комплексного тестування системи сформовано набір вхідних даних, що охоплює різноманітні сценарії використання та граничні випадки. Вхідні дані включають відеопотоки дорожніх сцен, профілі користувачів, запити на побудову маршрутів, історичні дані про завантаженість доріг.

Таблиця 4.1 – Основні характеристики датасетів

Датасет	Кількість даних	Тип анотацій	Особливості
BDD100K	100k зобр., 10k відео	Bounding boxes, lane markings	Різні умови зйомки
COCO	118k зображень	Boxes, segmentation	Багато класів
Власні дані	~4000 кадрів	Bounding boxes	Локальні дорожні умови

Відеодані для тестування модуля комп'ютерного зору отримані з публічного датасету BDD100K, що містить понад сто тисяч відеороликів дорожніх сцен з різних міст США у різних умовах освітлення та погоди. Кожен відеоролик супроводжується детальними анотаціями з розміткою обмежувальних рамок навколо об'єктів дорожньої інфраструктури: легкові автомобілі, вантажівки, автобуси, мотоцикли, велосипедисти, пішоходи. Використання стандартизованого датасету дозволяє порівнювати результати з іншими публікаціями та забезпечує відтворюваність експериментів. Додатково зібрано власний датасет відеозаписів українських міст для оцінки роботи системи у специфічних умовах регіону. Датасет включає понад п'ять годин записів з Києва, Львова, Харкова та Одеси, виконаних за допомогою відеореєстраторів з роздільною здатністю Full High Definition (HD). Записи

охоплюють різні типи доріг: автомагістралі, міські проспекти, житлові вулиці. Анотування виконано напівавтоматично з використанням попередньо навченої моделі для генерації початкових міток та ручної корекції помилок.

Профілі користувачів для тестування створені на основі десяти архетипів з різними комбінаціями переваг. Швидкий водій має максимальний пріоритет мінімізації часу поїздки. Економний водій цінує мінімізацію відстані та витрат палива. Обережний водій віддає перевагу безпечним маршрутам з меншою завантаженістю. Любитель краєвидів готовий витратити додатковий час заради естетично привабливих маршрутів. Професійний кур'єр оптимізує маршрути для множинних точок доставки [32-34]. Для кожного архетипу згенеровано синтетичну історію з мінімум тридцяти поїздок, що відображають відповідні переваги.

Для тестування режиму холодного старту створено окрему підмножину профілів без історії переміщень, що містять тільки базову демографічну інформацію та початкові налаштування. Для оцінки здатності системи виявляти зміни переваг підготовлено профілі з еволюцією переваг у часі, де історія поїздок розділена на два періоди з різними характеристиками вибору.

Запити на побудову маршрутів сформовано для покриття різних сценаріїв: короткі поїздки в межах одного району тривалістю до десяти хвилин, середні поїздки через декілька районів тривалістю від п'ятнадцяти до тридцяти хвилин, довгі міжміські поїздки понад тридцять хвилин. Додатково включено запити з нестандартними обмеженнями: уникнення платних доріг, обов'язковий проїзд через проміжні точки, часові вікна прибуття до певного моменту.

Історичні дані про завантаженість доріг отримані з публічних API сервісів Google Maps Traffic та TomTom Traffic Stats. Зібрано статистику середньої швидкості руху на основних дорогах міста для різних годин доби та днів тижня протягом трьох місяців. Дані агреговані з інтервалом п'ятнадцять хвилин для будніх днів. Додатково зібрано інформацію про дорожні події: аварії, ремонтні роботи, перекриття вулиць, з фіксацією координат, часу та

впливу на швидкість руху. Для кількісної оцінки якості рекомендацій створено еталонні маршрути, розмічені експертами як оптимальні для кожної комбінації початку, кінця та профілю користувача [35-37]. Троє експертів з досвідом професійного водіння незалежно оцінювали альтернативні маршрути за множиною критеріїв та обирали найкращий варіант. У випадках розбіжності думок проводилося обговорення для досягнення консенсусу. Ці еталонні маршрути використовуються як ground truth для обчислення метрик точності.

### 4.3 Результати тестування

Експериментальна перевірка системи проведена за трьома основними напрямками: оцінка точності модуля комп'ютерного зору, оцінка якості рекомендацій маршрутів, оцінка продуктивності [38-40]. Результати представлені з порівнянням до базових підходів для об'єктивної оцінки переваг розробленого методу.

Таблиця 4.2 – Результати тестування моделей

Модель	mAP@50 (%)	mAP@50-95 (%)	FPS (CPU)	FPS (GPU)
YOLOv8	89.3	52.1	9	88
YOLOv11	92.7	55.8	12	110
Custom YOLOv11	93.4	57.2	13	118

Точність модуля комп'ютерного зору оцінена на тестовій підмножині датасету BDD100K з десяти тисяч анотованих кадрів. Модифікована модель YOLOv11 з додатковою головою оцінки трафіку показала середню точність mAP50 на рівні 0.78, що на три відсоткових пункти вище за базову модель

YOLOv8. Для класів легкових автомобілів досягнуто точність 0.82, для вантажівок 0.76, для пішоходів 0.71. Швидкість обробки склала 42 кадри на секунду на GPU NVIDIA RTX 3080, що забезпечує обробку відеопотоків у реальному часі з запасом продуктивності.

Якість трекінгу об'єктів оцінена на датасеті MOT17 з використанням стандартних метрик MOTA та MOTP. Модифікований алгоритм DeepSORT з покращеною моделлю реідентифікації досяг MOTA 68.5%, що на чотири відсоткових пункти вище за базову реалізацію. Метрика MOTP склала 0.82, вказуючи на високу точність локалізації об'єктів. Кількість втрачених треків зменшилась на 15% порівняно з базовим алгоритмом завдяки кращій обробці довготривалих оклюзій. Точність оцінки стану трафіку перевірена через порівняння з даними від проїзджих датчиків швидкості на тестовій ділянці дороги. Середня абсолютна похибка оцінки швидкості склала 4.2 кілометри на годину, середня відносна похибка 8.3 відсотка. Коефіцієнт кореляції між оцінками системи та показаннями датчиків досяг 0.91. Найбільші похибки спостерігаються в умовах дуже низької освітленості та інтенсивних опадів, коли якість виявлення об'єктів знижується.

Якість рекомендацій маршрутів оцінена через порівняння з еталонними маршрутами експертів. Для існуючих користувачів з накопиченою історією поїздок система досягла Precision@1 на рівні 0.72, що означає співпадіння найкращої рекомендації з вибором експерта у 72 відсотках випадків. Метрики Precision@3 та Precision@5 склали 0.89 та 0.95 відповідно. Для порівняння, базовий підхід на основі тільки мінімізації часу без персоналізації досяг Precision@1 лише 0.51.

Для нових користувачів без історії поїздок система показала Precision@1 на рівні 0.58, що на 12 відсоткових пунктів нижче за існуючих користувачів, але на 15 відсоткових пунктів вище за чисту колаборативну фільтрацію. Це підтверджує ефективність запропонованої триступеневої стратегії холодного старту з використанням демографічної інформації, трансферного навчання та візуальних даних. Здатність системи виявляти зміни у перевагах користувачів

оцінена на профілях з синтетичною еволюцією переваг. Система успішно виявила зміну у 83 відсотках випадків протягом перших п'яти поїздок після справжньої зміни. Середній час виявлення склав 3.2 поїздки. Частка помилкових спрацювань при відсутності справжніх змін склала 7 відсотків. Після виявлення зміни та активації механізму адаптації точність рекомендацій відновлювалась до рівня 0.69 протягом наступних десяти поїздок.

Вплив інтеграції даних комп'ютерного зору на якість рекомендацій оцінено через порівняння повної системи з версією, що використовує тільки історичні дані про трафік [41-42]. Система з візуальним моніторингом показала на 8 відсотків вищу точність прогнозування часу прибуття у ситуаціях з динамічними змінами дорожньої обстановки. Покращення особливо помітне для маршрутів через ділянки з нестабільним трафіком та під час дорожніх подій.

Продуктивність системи оцінена через вимірювання латентності обробки запитів [43]. Середній час формування рекомендацій від отримання запиту до відправлення відповіді склав 380 мілісекунд для міських маршрутів та 620 мілісекунд для міжміських. Це задовольняє вимогу інтерактивності для користувацьких застосунків. Розподіл часу показав, що найбільш ресурсоемною є операція генерації альтернативних маршрутів, що займає близько 60 відсотків загального часу. Система зберігає стабільну латентність до 50 запитів на секунду на одному сервері з 8 ядрами CPU та одним GPU [44-45]. Користувацька задоволеність оцінена через опитування тестової групи з двадцяти реальних користувачів протягом двох тижнів використання. Користувачі оцінювали корисність рекомендацій за п'ятибальною шкалою після кожної поїздки. Середня оцінка склала 4.3 бали для повної системи проти 3.7 балів для базового підходу без персоналізації. 85 відсотків користувачів відзначили, що рекомендації враховують їх індивідуальні переваги. 78 відсотків висловили задоволеність наявністю пояснень причин рекомендацій.

#### 4.4 Порівняльний аналіз з базовими підходами: перевага на 4-21 відсоткових пунктів

Систематичне порівняння з існуючими підходами проведене для чотирьох базових методів: оптимізація тільки за часом, колаборативна фільтрація, контентна фільтрація, гібридний метод без комп'ютерного зору. Перший метод представляє традиційні навігаційні системи, що завжди рекомендують найшвидший маршрут незалежно від переваг користувача. Другий метод будує рекомендації на основі схожості між користувачами. Третій аналізує характеристики маршрутів та будує модель переваг на основі історії вибору. Четвертий комбінує колаборативну та контентну фільтрацію без використання візуальних даних. Порівняння точності рекомендацій для існуючих користувачів показало перевагу розробленого методу за всіма метриками. Precision@1 склав 0.72 для розробленого методу проти 0.68 для гібридного методу без комп'ютерного зору, 0.61 для контентної фільтрації, 0.56 для колаборативної фільтрації, 0.51 для оптимізації тільки за часом. Покращення на 4 відсоткових пункти порівняно з найкращим базовим методом та на 21 відсотковий пункт порівняно з найпростішим підходом демонструє значну практичну цінність персоналізації з урахуванням візуальних даних.

Таблиця 4.3 – Якість навігаційних рекомендацій

Метод	Precision@1	Precision@3	Час відповіді (мс)	Врахування переваг
Базовий метод	0.51	0.62	490	Низьке
Колаборативна фільтрація	0.58	0.67	450	Середнє
Запропонований метод	0.72	0.81	380	Високе

Для нових користувачів у режимі холодного старту розроблений метод показав найбільшу перевагу. Precision@1 склав 0.58 проти 0.52 для гібридного методу, 0.48 для контентної фільтрації, 0.43 для колаборативної фільтрації. Покращення на 6-15 відсоткових пунктів підтверджує ефективність використання візуальних індикаторів поведінки для швидкого виявлення переваг навіть без значної історії поїздок.

Точність прогнозування часу прибуття є критично важливою для навігаційних систем. Розроблений метод досяг середньої абсолютної похибки 3.8 хвилини для маршрутів тривалістю близько тридцяти хвилин. Гібридний метод без комп'ютерного зору показав похибку 4.6 хвилин, підходи на основі тільки історичних даних 5.2 хвилини. Покращення на 17-27 відсотків особливо помітне у годинах пік та при дорожніх подіях, коли візуальні дані надають більш актуальну інформацію порівняно з історичними патернами. Швидкість адаптації до змін у перевагах користувача оцінена через кількість поїздок, необхідних для відновлення високої точності після зміни. Розроблений метод з явним механізмом виявлення змін відновлює точність до 0.69 протягом десяти поїздок. Гібридний метод без механізму знаходження змін потребує близько двадцяти поїздок. Контентна фільтрація адаптується найповільніше, потребуючи близько тридцяти поїздок. Прискорення адаптації у два-три рази має велике практичне значення для користувачів з мінливими потребами.

Обчислювальна складність розробленого методу є вищою порівняно з базовими підходами через необхідність обробки відеопотоку. Середній час формування рекомендацій склав 380 мілісекунд проти 50 мілісекунд для оптимізації тільки за часом, 120 мілісекунд для колаборативної фільтрації, 80 мілісекунд для контентної фільтрації, 200 мілісекунд для гібридного методу. Додаткові витрати у 1.9-7.6 разів виправдані значним покращенням точності рекомендацій, особливо для користувачів з нестандартними перевагами.

Стійкість до помилкових даних оцінена через ін'єкцію шумів різної інтенсивності у результати виявлення об'єктів. При 10 відсотках помилкових фіксацій точність розробленого методу знижується тільки на 3 відсотки

завдяки механізму верифікації та узгодження з альтернативними джерелами. Базові методи втрачають 8-12 відсотків точності при аналогічному рівні шуму. При 20 відсотках помилок розроблений метод зберігає працездатність з деградацією точності на 7 відсотків, тоді як деякі базові підходи демонструють критичну деградацію. Здатність пояснювати рекомендації є важливою для довіри користувачів та прийняття системи. Розроблений метод надає детальні пояснення через аналіз SHAP значень ознак, що дозволяє ідентифікувати фактори, які найбільше вплинули на оцінку кожного маршруту. Базові підходи надають тільки базову інформацію про характеристики маршрутів без пояснення причин їх вибору. Опитування показало, що 78 відсотків користувачів цінують наявність пояснень та більше довіряють системі з прозорими рекомендаціями.

Загальна оцінка показує, що розроблений метод досягає найкращого балансу між точністю рекомендацій, швидкістю адаптації до змін, стійкістю до помилок при помірному збільшенні обчислювальної складності. Метод особливо ефективний для користувачів з нестандартними перевагами, у ситуаціях холодного старту, при динамічних змінах дорожньої обстановки. Результати експериментальної перевірки підтверджують досягнення поставлених цілей та вирішення виявлених у першому розділі недоліків існуючих підходів.

## ВИСНОВКИ

У магістерській дисертації вирішено важливу науково-практичну задачу підвищення точності навігаційних рекомендацій в умовах обмеженої апріорної інформації про користувача через розробку методу адаптивних рекомендацій з семантичною інтеграцією візуальних даних від системи комп'ютерного зору.

Проведено комплексний аналіз існуючих методів розпізнавань об'єктів на відеозображеннях, побудови рекомендацій для навігаційних систем та вирішення проблеми холодного старту. Виявлено п'ять ключових недоліків існуючих підходів: недостатня точність та актуальність інформації про стан трафіку при використанні тільки історичних даних, неефективне вирішення проблеми холодного старту для нових користувачів, відсутність механізмів виявлення та адаптації до змін переваг користувачів у часі, обмежена інтерпретованість рекомендацій, недостатня стійкість до помилкових даних та адверсальних атак. Встановлено, що інтеграція даних комп'ютерного зору з методами персоналізованих рекомендацій є перспективним напрямком вирішення виявлених проблем.

Розроблено концептуальну модель багаторівневої системи побудови рекомендацій, що включає п'ять рівнів обробки інформації: рівень збору даних з модулями захоплення відеопотоку, GPS геолокації та інтернет-сервісів, рівень обробки даних з виявленням YOLOv11, трекінгом DeepSORT та прогнозуванням LSTM, рівень семантичної інтеграції з байєсівським злиттям різнорідних даних, рівень персоналізації з адаптивним профілюванням та виявленням змін переваг, рівень формування рекомендацій з генерацією альтернатив алгоритмом  $A^*$ , ранжуванням через нейронну мережу та поясненнями через SHAP значення. Запропоновано механізм виявлення змін переваг користувачів через поєднання статистичного аналізу помилок прогнозування з тестом Mann-Kendall та верифікації через дані комп'ютерного зору про зміни у транспортній поведінці.

Удосконалено метод побудови навігаційних рекомендацій через інтеграцію модифікованої архітектури YOLOv11 зі спеціалізованою головою для оцінки щільності трафіку та карт руху, що дозволяє безпосередньо генерувати агреговані характеристики транспортного потоку при збереженні високої швидкості обробки. Розроблено триступеневу стратегію вирішення холодного старту на основі демографічної інформації та явних налаштувань для ініціалізації базової моделі, трансферного навчання від схожих користувачів для використання колективного досвіду, візуального аналізу поведінки під час перших поїздок для виявлення неявних індикаторів переваг. Запропоновано механізм онлайн-навчання моделі переваг з регуляризацією Elastic Weight Consolidation та адаптивним налаштуванням швидкості навчання залежно від стабільності параметрів, що запобігає катастрофічному забуванню попередніх знань при адаптації до нових переваг.

Розроблено інформаційну технологію реалізації десятиступеневого процесу формування персоналізованих рекомендацій на основі фреймворків PyTorch для навчання моделей та ONNX Runtime для кросплатформенного інференсу з оптимізацією під різні апаратні прискорювачі. Реалізовано мікросервісну архітектуру розгортання з використанням Kubernetes для оркестрації контейнерів, RabbitMQ для асинхронної обробки відеопотоків через чергу повідомлень, PostgreSQL для зберігання профілів користувачів з реплікацією, Redis Cache для швидкого доступу до часто запитуваних результатів, Prometheus та Grafana для моніторингу продуктивності. Програмна реалізація включає модульну систему класів VideoProcessor для обробки відео, ObjectTracker для трекінгу, TrafficAnalyzer для аналізу трафіку, UserProfileService для профілювання, RouteGenerator для генерації маршрутів, RouteRanker для ранжування з поясненнями.

Проведено експериментальну перевірку розробленого методу на публічному датасеті BDD100K з понад ста тисячами відеороликів дорожніх сцен та власному датасеті з п'яти годин записів українських міст. Модифікована модель YOLOv11 досягла точності фіксації mAP50 на рівні 0.78

при швидкості обробки 42 кадри на секунду на GPU NVIDIA RTX 3080, що на три відсоткових пункти вище базової моделі. Модуль трекінгу DeepSORT показав MOTA 68.5% та MOTP 0.82 з зменшенням кількості втрачених треків на 15 відсотків. Точність оцінки швидкості трафіку склала 4.2 кілометри на годину середньої абсолютної похибки з кореляцією 0.91 з показаннями проїжджих датчиків. Для існуючих користувачів система досягла Precision@1 на рівні 0.72, що на 4-21 відсотковий пункт вище базових підходів. У режимі холодного старту для нових користувачів точність склала 0.58, що на 15 відсоткових пунктів вище чистої колаборативної фільтрації. Механізм виявлення змін переваг показав успішність 83 відсотки за 3.2 поїздки з часткою помилкових спрацювань 7 відсотків, швидкість адаптації у два-три рази вища за базові методи. Інтеграція даних комп'ютерного зору підвищила точність прогнозування часу прибуття на 8 відсотків у динамічних умовах. Продуктивність системи складає 380 мілісекунд латентності з масштабованістю до 50 запитів на секунду. Користувацька задоволеність досягла 4.3 бали з 5, при цьому 85 відсотків користувачів відзначили врахування індивідуальних переваг, 78 відсотків висловили задоволеність наявністю пояснень рекомендацій.

Отримані результати підтверджують ефективність запропонованого методу адаптивних рекомендацій з семантичною інтеграцією візуальних даних для підвищення точності навігаційних систем в умовах обмеженої апріорної інформації про користувача. Розроблений метод демонструє найкращий баланс між точністю рекомендацій, швидкістю адаптації до змін, стійкістю до помилок при помірному збільшенні обчислювальної складності. Метод особливо ефективний для користувачів з нестандартними перевагами, у ситуаціях холодного старту, при динамічних змінах дорожньої обстановки.

Практична значущість отриманих результатів полягає у можливості їх використання для створення комерційних навігаційних систем нового покоління з підвищеною точністю та персоналізацією рекомендацій. Розроблені програмні компоненти можуть бути інтегровані у платформи

інтелектуального керування міським трафіком для оптимізації транспортних потоків та зниження перевантаження доріг. Запропонований підхід до вирішення проблеми холодного старту може застосовуватись у різних рекомендаційних системах за межами навігаційної предметної області.

Перспективами подальших досліджень є розширення функціональності системи через інтеграцію додаткових джерел даних, таких як соціальні мережі для врахування поточних подій та масових заходів, інформація про паркування для планування повного циклу поїздки, дані про якість дорожнього покриття для оптимізації маршрутів з точки зору комфорту. Перспективним напрямком є розробка методів федеративного навчання для побудови моделей переваг з використанням даних від багатьох користувачів при збереженні конфіденційності їх персональної інформації. Актуальною задачею є дослідження можливостей застосування трансформерних архітектур для прогнозування еволюції дорожньої обстановки з урахуванням складних темпоральних залежностей та взаємовпливів різних ділянок дорожньої мережі.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Волженцева І. ТЕОРЕТИЧНИЙ АНАЛІЗ ВИКОРИСТАННЯ МЕТОДУ МОДЕЛЮВАННЯ В ПСИХОЛОГІЇ. Proceedings of the National Aviation University. Series: Pedagogy, Psychology. 2022. № 20. С. 62–74 DOI: 10.18372/2411-264X.20.16918 (дата звернення: 24.11.2025).
2. Bertsekas D. P. Abstract Dynamic Programming. 2nd ed. Belmont, MA : Athena Scientific, 2018. 360 p. ISBN 978-1-886529-62-1. (дата звернення: 24.11.2025).
3. Туленков А. В., Залюбовський Я. І. Методи та засоби аналізу даних системи домашньої автоматизації. 2021. DOI: 10.31474/1996-1588-2021-1-32-57-67 (дата звернення: 25.11.2025).
4. Christensen M., та ін. Automation isn't automatic. Chemical Science. 2021. Т. 12, № 47. P. 15473–15490. DOI: 10.1039/D1SC04588A (дата звернення: 25.11.2025).
5. Zwolak J. P., Taylor J. M. Colloquium : Advances in automation of quantum dot devices control. Reviews of Modern Physics. 2023. Т. 95, № 1 DOI: 10.1103/revmodphys.95.011006 (дата звернення: 27.11.2025).
6. Hudayberdiev U., Lee J. An Ensemble Deep Learning Framework for Smart Tourism Landmark Recognition Using Pixel-Enhanced YOLO11 Models. Sustainability. 2025. Vol. 17, no. 12. P. 5420. DOI: 10.3390/su17125420 (дата звернення: 28.11.2025).
7. Win N. H. Privately Owned Public Open Space under the Lens: YOLO Analysis of Pedestrian Movement : Doctoral dissertation / Ulsan National Institute of Science and Technology. Ulsan, 2024.
8. Targhini E. Study of a semantic segmentation algorithm for disaster assessment in an edge computing environment : Master's thesis / Politecnico di Milano. Milan, 2023.

9. ПРОНІНА О. І., ПОЛІЩУК А. А. ІНТЕЛЕКТУАЛЬНА СИСТЕМА ВИЗНАЧЕННЯ ІНГРЕДІЄНТІВ ТА ПРОПОЗИЦІЇ РЕЦЕПТІВ З ВИКОРИСТАННЯМ AR НА ПЛАТФОРМІ. Вісник Херсонського національного технічного університету. 2024. № 1(88). С. 279–286. DOI: 10.35546/kntu2078-4481.2024.1.39 (дата звернення: 29.11.2025).

10. Рувінська В. М., та ін. Відеоспостереження для систем безпеки: моделі, методи та рішення. Інформатика та математичні методи в моделюванні. 2021. Т. 11, № 4. С. 331–342. DOI: 10.15276/imms.v11.no4.331 (дата звернення: 29.11.2025).

11. Сініцин І. П., та ін. Дослідження продуктивності нейромереж YOLO для вбудованих систем і дронів. Телекомунікаційні та інформаційні технології. 2025. № 3. С. 177–186. DOI: 10.31673/2412-4338.2025.038720 (дата звернення: 29.11.2025).

12. Вдовін Б. Б., Яковина В. С., Томашевський О. М. Виявлення лісових пожеж на потокових зображеннях з БПЛА. Scientific Bulletin of UNFU. 2025. Т. 35, № 5. С. 97–107. DOI: 10.36930/40350511 (дата звернення: 03.12.2025).

13. Момот А. С., та ін. Автоматизоване детектування дефектів друкованих плат на основі нейронної мережі YOLOv5. Technical Diagnostics & Nondestructive Testing. 2024. Доступ через базу даних EBSCOhost (дата звернення: 04.12.2025).

14. He L., та ін. Research on object detection and recognition in remote sensing images based on YOLOv11. Scientific Reports. 2025. Т. 15. Article 14032. DOI: 10.1038/s41598-025-96314-x (дата звернення: 04.12.2025).

15. Liao Q., та ін. Dual-Path Enhanced YOLO11 for Lightweight Instance Segmentation. Electronics. 2025. Т. 14, № 17. Article 3389. DOI: 10.3390/electronics14173389 (дата звернення: 04.12.2025).

16. Zhang X. A lightweight model FDM-YOLO for small target improvement : preprint / arXiv. 2025. DOI: 10.48550/arXiv.2503.04452 (дата звернення: 04.12.2025).

17. Grauer H., та ін. Vision-based detection of uncooperative targets on small satellites : preprint / arXiv. 2024. DOI: 10.48550/arXiv.2408.12084.
18. Tang N., Zhang H. Implementing digital technology in landscape architecture. *Measurement*. 2025. Т. 235. Article 116982. DOI: 10.1016/j.measurement.2024.116982.
19. Jegham N. та ін. Yolo evolution: A comprehensive benchmark. preprint / arXiv. 2024. DOI: 10.48550/arXiv.2411.00201 (дата звернення: 05.12.2025).
20. Ling H., та ін. FDL-YOLO: efficient real-time traffic detection. *The Journal of Supercomputing*. 2025. Т. 81, № 8. Article 966. DOI: 10.1007/s11227-025-07398-1 (дата звернення: 05.12.2025).
21. Ali M. L., Zhang Z. The YOLO Framework: A comprehensive review. *Computers*. 2024. Т. 13, № 12. Article 336. DOI: 10.3390/computers13120336 (дата звернення: 06.12.2025).
22. Luan D., та ін. DB-YOLO for foggy scenarios detection. *IET Image Processing*. 2025. Т. 19, № 1. Article e70246. DOI: 10.1049/ipr2.70246 (дата звернення: 06.12.2025).
23. Talaat F. M., та ін. Smart traffic management using YOLOv11. *Neural Computing and Applications*. 2025. Article 11434. DOI: 10.1007/s00521-025-11434-9. (дата звернення: 07.12.2025).
24. Liu L. Application of Multi-objective Evolutionary Algorithms. *Scalable Computing: Practice and Experience*. 2024. Т. 25, № 4. P. 2973–2984. DOI: 10.12694/scpe.v25i4.2929 (дата звернення: 07.12.2025).
25. Khalaf A. L., та ін. Real time pedestrian detection using enhanced YOLO. *TELKOMNIKA*. 2024. Т. 22, № 2. P. 362–371. DOI: 10.12928/TELKOMNIKA.v22i2.24854 (дата звернення: 07.12.2025).
26. Sapkota R., та ін. YOLO advances to its genesis: review. *Artificial Intelligence Review*. 2025. Т. 58, № 9. Article 274. DOI: 10.1007/s10462-025-11253-3.

27. Alif M. A. R., Hussain M. YOLOv1 to YOLOv10: review for agriculture. preprint / arXiv. 2024. DOI: 10.48550/arXiv.2406.10139 (дата звернення: 07.12.2025).
28. Ahad A., Kidwai F. A. YOLO for parking detection. Innovative Infrastructure Solutions. 2025. Т. 10, № 6. Article 252. DOI: 10.1007/s41062-025-02041-7 (дата звернення: 07.12.2025).
29. Sapkota R., та ін. YOLOv12 to its genesis. preprint / arXiv. 2024. DOI: 10.48550/arXiv.2406.19407 (дата звернення: 08.12.2025).
30. Flores-Calero M., та ін. Traffic sign detection using YOLO: review. Mathematics. 2024. Т. 12, № 2. Article 297. DOI: 10.48550/arXiv.2406.19407 (дата звернення: 08.12.2025).
31. Gelar T., Fitriani S., Rachmat S. YOLO and IoT Applications in Smart Waste Management. Green Intelligent Systems and Applications. 2025. Т. 5, № 2. P. 123–139. DOI: 10.53623/gisa.v5i2.706 (дата звернення: 08.12.2025).
32. Chu Z. D-YOLO for adverse weather. preprint / arXiv. 2024. DOI: 10.48550/arXiv.2403.09233 (дата звернення: 08.12.2025).
33. Liu W., Huang J., Wang L., Shi F., Zhang S. VP-YOLO: A robust perception-inspired object detection model. Expert Systems with Applications. 2025. Т. 265. № 124311. DOI: 10.1016/j.eswa.2025.124311. (дата звернення: 08.12.2025).
34. Chaithanya C., Yatheendra K., Gopichand P. MCS-YOLO for automated driving. International Journal of HRM and Organizational Behavior. 2024. Т. 12, № 3. P. 229–242. URL: <https://ijhrmob.org/index.php/ijhrmob/article/view/239/225> (дата звернення: 09.12.2025).
35. Liu S., та ін. EcoDetect-YOLO for waste exposure detection. Sensors. 2024. Т. 24, № 14. Article 4666. DOI: 10.3390/s24144666 (дата звернення: 09.12.2025).

36. Gu X., Zhang G. PSC-YOLO for instance segmentation. *Journal of Real-Time Image Processing*. 2025. Т. 22, № 2. Р. 1–13. DOI: 10.1007/s11554-025-01668-0 (дата звернення: 09.12.2025).
37. Katsamenis I., та ін. DORIE: Benchmark of YOLO Architectures. *Sensors*. 2025. Т. 25, № 21. Article 6653. DOI: 0.3390/s25216653 (дата звернення: 09.12.2025).
38. Saeheaw T. SC-YOLO for PPE detection. *Buildings*. 2025. Т. 15, № 16. Article 2854. DOI: 10.3390/buildings15162854 (дата звернення: 09.12.2025).
39. Пустоваров В. В. Інформаційна технологія розробки системи підтримки прийняття рішення про розпізнавання будівель на космічних та аерофотознімках : дис. ... канд. техн. наук : 05.13.06 / Черкаський державний технологічний університет. Черкаси, 2021.
40. Hao Q., Qin L. Intelligent transportation video processing system. *IEEE Access*. 2020. Т. 8. Р. 13769–13780. DOI: 10.1007/s00521-022-06979-у (дата звернення: 10.12.2025).
41. Mehmood M., Shahzad A., Zafar B., Shabbir A., Ali N., Ahmad A. Remote Sensing Image Classification: A Comprehensive Review and Applications. *Mathematical Problems in Engineering*. 2022. Vol. 2022. Article ID 5880959. DOI: 10.1155/2022/5880959. (дата звернення: 10.12.2025).
42. Du L., Wang Y. Bi-YOLO for heritage buildings. *Journal of Building Engineering*. 2024. Т. 97. № 110817. DOI: 10.1016/j.jobe.2024.110817 (дата звернення: 10.12.2025).
43. Zhang Q., Hu X. MSFFA-YOLO: object detection in fog. *IEEE Transactions on Instrumentation and Measurement*. 2023. Т. 72. Р. 1–12.
44. Wang Y., Ke H., Cai H. PC-YOLO for adverse weather. *Journal of Electronic Imaging*. 2025. Т. 34, № 2. Article 023049. DOI: 10.1117/1.JEI.34.2.023049 (дата звернення: 10.12.2025).
45. Thitisiriwech K., та ін. Quality of life prediction in driving scenes. *Sustainability*. 2023. Т. 15, № 3. Article 2847. DOI: 10.3390/su15032847 (дата звернення: 10.12.2025).