

SECTION 8. INFORMATICS AND CYBERNETICS

DOI: 10.46299/ISG.2024.MONO.TECH.4.8.1

8.1 Проєктування та дослідження інтелектуальної системи виявлення та класифікації рухомих об'єктів у відеопотоці

Вступ. Виявлення об'єктів у відеопотоці -одна з найбільш фундаментальних і складних задач по виявленню об'єктів на зображеннях і відео. У минулому велику увагу приділяли проведенню додаткових досліджень щодо задач комп'ютерного зору, таких як класифікація об'єктів, підрахунок об'єктів та моніторинг об'єктів.

У реальному середовищі присутні різні типи об'єктів. Ідентифікація цих об'єктів за допомогою машинного навчання є складним завданням. Комп'ютерне бачення - це розділ інформатики, який дозволяє машині бачити, ідентифікувати та обробляти об'єкти на нерухомих зображеннях та відеозаписах у візуальному світі. Відео - це послідовність безперервних зображень (так званих відеокадрів), що відображаються з певною частотою кадрів. Люди негайно виявляють або розпізнають об'єкти за зображеннями або відео і визначають їх місце розташування завдяки взаємозв'язку нейронів мозку. Виявлення об'єктів на відео - це завдання штучного інтелекту, яке використовується для того самого процесу виявлення об'єктів.

Виявлення об'єктів є фундаментальним, і давня проблема комп'ютерного зору була основною активною сферою досліджень протягом декількох десятиліть [279-288]. Різні методи виявлення об'єктів використовувалися у різних додатках комп'ютерного зору, таких як розпізнавання осіб, підрахунок пішоходів, системи безпеки, виявлення транспортних засобів, самокеровані автомобілі і т.ін. деякі терміни комп'ютерного зору, такі як локалізація, класифікація і розпізнавання об'єктів, пов'язані з обробкою виявлення об'єктів.

Класифікація об'єктів визначає клас одного або декількох об'єктів, що існують на зображенні, і присвоює об'єктам мітки [280]. Локалізація об'єкта - це

процес, який визначає положення одного або декількох об'єктів на зображенні або відео за допомогою обмежувальної рамки [281].

Поєднання процесу локалізації об'єкта та класифікації відоме як виявлення об'єкта [282]. Повний процес розпізнавання об'єктів приймає зображення в якості вхідних даних, ідентифікує об'єкти, присвоює мітки об'єкту відповідного класу і видає ймовірність класу розпізнаного об'єкта [283, 284]. Кожен об'єкт має унікальні ознаки, які допомагають ідентифікувати клас.

Наприклад, квадрат з усіма рівними сторонами допомагає виявити об'єкт квадратного розміру. Велика частина дослідницької роботи з виявлення об'єктів поділяється на три категорії: виявлення помітних об'єктів, виявлення предметності та виявлення об'єктів, специфічних для конкретної категорії. Основною метою виявлення та класифікації об'єктів є ідентифікація категорій об'єктів за зображеннями, і вона має справу з міжкласовими та внутрішньокласовими варіаціями та подібністю [285].

Метою даної роботи є підвищення швидкості і точності розпізнавання об'єктів у відеопотоці за рахунок дослідження, проектування і розробки програмних компонентів для підсистеми виявлення та класифікації об'єктів у відеопотоці.

Основна частина. Комп'ютерний зір та штучний інтелект мають підмножину, відому як виявлення об'єктів. Де його метою є розробка моделі машинного навчання, здатної виявляти певні об'єкти на зображенні [279, 280]. Саме виявлення об'єктів можна розглядати як одне з фундаментальних понять, що використовуються у багатьох реальних програмах машинного навчання, таких як розпізнавання зображень, розпізнавання обличчя та багато інших [279, 281]. Для того щоб модель машинного навчання розпізнавала об'єкти, їй повинен бути наданий величезний обсяг даних у вигляді зображень, що містять вибрані об'єкти, які модель повинна виявити [282]. Конкретні об'єкти на зображенні, яких вона ніколи раніше не бачила [281, 283].

Об'єкти інформативної області, присутні на зображенні, можуть відрізнятися за розміром і співвідношенням сторін. Модель виявлення об'єктів

сканує все зображення в різних масштабах, щоб виявити об'єкти і знайти впізнаваний візерунок. Можливі положення об'єктів можна визначити за допомогою цієї стратегії, але є деякі недоліки. Ця стратегія є дорогою, оскільки під час обробки створюється кілька вікон-кандидатів, що вимагає високої обчислювальної потужності. Незважаючи на це, якщо в цьому методі застосовується фіксована кількість розсувних вікон, то можуть вийти задовільні області.

Виділення ознак ознаки є особливо важливим параметром зображення, яке використовувалося для класифікації та розпізнавання об'єктів. На першому етапі розпізнається шаблон, а потім цей шаблон надалі використовується для вилучення відмінних ознак, пов'язаних з об'єктом.

Для вилучення ознак використовуються різні методи, такі як HOG, SIFT та Нагг. Однак розробка хорошого та надійного описувача функцій є складним завданням через великі проблеми виявлення об'єктів на рівні вулиці є одним з варіантів використання виявлення об'єктів. В основному використовується в самокерованих автомобілях, безпілотних літальних апаратах та камерах спостереження. Виявляючи ці об'єкти, можна зменшити кількість нещасних випадків, таких як автокатастрофи, зіткнення та багато іншого. Це також дозволяє органам влади відстежувати можливі порушення щодо безпеки дорожнього руху.

Класифікація використовується для прогнозування класу за заданими точками даних. У цьому процесі відповідні ознаки об'єднуються для представлення об'єкта порівняно з навченою моделлю. Візуальне розпізнавання вимагає класифікації, щоб відрізнити цільовий об'єкт від інших об'єктів, роблячи представлення об'єкта більш семантичним та інформативним. SVM, DPM, AdaBoost і т.ін. зазвичай використовуються для класифікації.

Зі зростанням технічного розвитку біометрія стає важливою для розпізнавання особистості людини в цілях безпеки. Біометрична аутентифікація є більш надійним методом ідентифікації особи. Аутентифікація здійснюється на основі різних біологічних особливостей кожної людини, таких як відбитки

пальців, ДНК, сітківка ока, вухо і т.ін. Існують різні типи методів виявлення об'єктів, які використовувалися для біометричного аналізу в минулих дослідницьких роботах.

Більшість житлових зон (метро, парки, школи, торгові центри і т.ін.) контролюються різними технічними засобами відеоспостереження, оскільки люди не можуть постійно відстежувати відеокліпи. Виявлення об'єктів відіграє важливу роль в відеоспостереженні, дозволяючи відразу ідентифікувати і відстежувати екземпляри конкретного об'єкта на місці події, наприклад, відстежувати підозрювану людину або транспортний засіб по відеозапису.

В останні роки було доведено, що автономні роботи є однією з найцікавіших областей досліджень. Виявлення об'єкта - це основне завдання, яке виконує робот для ідентифікації прилеглих об'єктів і виконувати деякі операції, такі як надання інформації, відкриття-закриття дверей, сигналізація і т.ін.

Виявлення людини також є складним завданням в комп'ютерному зорі, тому що люди як об'єкти мають різний зовнішній вигляд і приймають широкий спектр поз.

Для ідентифікації людей за зображеннями або відеозаписами було запропоновано різні архітектури виявлення об'єктів, така як виявлення пішоходів.

Виявлення об'єктів використовувалося для ідентифікації індивідуальної особи, і це перша область застосування у виявленні об'єктів людини. Функція розпізнавання осіб забезпечує високу точність виявлення при мінімізації часу обчислень. Розпізнавання обличчя допомогло адаптувати виявлення об'єктів до різних областей застосування. Зараз багато додатків використовують цю ідею для виявлення посмішок у режимі реального часу за допомогою камер, макіяжу обличчя, розрахунку віку тощо. З розвитком технологій Інтелектуальних Транспортних систем, таких як безпілотний автомобіль, збільшилась сфера досліджень. Ці інтелектуальні системи необхідні для ідентифікації, визначення місця розташування або відстеження прилеглих об'єктів і контролю швидкості транспортного засобу. Система виявлення об'єктів також працює з більш

дрібнозернистими зображеннями та зображеннями регіонального рівня, такими як виявлення та розпізнавання світлофорів.

Обмеження виявлення об'єктів:

– Мультиклас: різні програми вимагають одночасного виявлення більш ніж одного класу об'єктів. Таким чином, швидкість обробки виявлення стає важливою проблемою, а також можливість багаторазової класифікації без будь-якої втрати точності. Деякі методи виявлення об'єктів або навчені моделі обмежуються виявленням об'єктів одного класу з певним видом. Вони не можуть виявити один і той же об'єкт з різними видами, позами або варіаціями ракурсу.

– Об'єкт на зображенні представлений з різними співвідношеннями сторін і просторовим розташуванням. Припустимо, що розмір об'єкта дуже малий, наприклад, менше 5% зображення, тоді система не зможе виявити ці менші об'єкти. Іноді об'єкти розташовані дуже близько один до одного, як стопка тарілок, тому виявити ці об'єкти дуже складно.

– Ефективність - важливий параметр, вимірюваний для кожної системи виявлення об'єктів. Деякі з розроблених систем виявлення об'єктів відрізняються надійністю і швидкою обробкою, але вони вимагають високоефективних ресурсів (центрального і графічного процесорів). Мікросистеми мають дуже обмежені ресурси для обробки, тому це складне завдання.

– Моделі виявлення об'єктів можуть виявляти лише ті, для яких навчена модель. Наприклад, модель виявлення м'яча виявляє м'яч, і іноді вона розпізнає апельсин як м'яч, тому що апельсин має приблизно таку ж форму. Основною причиною несправності виявлення є те, що об'єкт з м'ячем не вмикається під час тренування.

У процесі створення моделі машинного навчання інші роботи зазвичай використовують різні алгоритми та архітектури, такі як згорткові нейронні мережі (CNN) [284, 285], опорні векторні машини (SVM) [282, 286, 287], вибір функцій [288] та багато інших. Результати цих різних алгоритмів і архітектур також можуть відрізнятися один від одного, що робить кожен з них унікальним і має свій власний найкращий варіант використання. Що стосується самого

виявлення об'єктів, то деякі з найбільш часто використовуваних архітектур, що використовуються за останні кілька років, складаються з однократних мультибоксів детекторів (SSD) [289], You Only Look Once (YOLO) [290], а також Region Based Convolutional Neural Networks (RCNN) та Faster -RCNN [291, 292, 293].

За останні кілька років у багатьох дослідженнях порівнювалися продуктивність і результативність алгоритмів виявлення об'єктів в конкретному випадку і навколишньому середовищу. Однак дослідження недостатньо вивчили новий алгоритм виявлення об'єктів, такий як YOLOv5. В даний час існує дуже мало досліджень, пов'язаних з YOLOv5 та його попередниками або іншими алгоритмами виявлення об'єктів, особливо для виявлення декількох об'єктів на дорозі.

Кілька статей порівнювали алгоритми виявлення об'єктів, специфічні для одного типу об'єктів на дорозі. Стаття С. Чоял та А.К. Сінгх [294] провели порівняння між FASTER-RCNN та SSD MobileNet на дорожніх знаках.

Результати показали, що FASTER-RCNN є більш точним, але вимагає більше часу на навчання, ніж SSD MobileNet. Крім того, подібне дослідження було проведено М. Шахудом та ін. [295] порівняли підтипи YOLOv3 на дорожніх знаках і виявили, що YOLOv3-tiny на 13% менш точний, ніж YOLOv3, але YOLOv3-tiny дуже корисний у програмах реального часу через високу частоту кадрів-до 200 кадрів в секунду (FPS).

Одним з важливих об'єктів для виявлення на дорозі є люди. С. Е. Кім та ін. [296] провели порівняння багатьох алгоритмів виявлення об'єктів, порівнювали алгоритми FASTER-RCNN, YOLO, SSD та R-FCN. Було виявлено, що YOLOv3 є найбільш ідеальним для виявлення людей, оскільки він дає відносно точні результати за розумний час. Мао та ін. [297] запропонували вдосконалення YOLOv3 з використанням шару просторового пірамідального об'єднання (SPP), щоб YOLOv3 краще працював для виявлення транспортних засобів. Його дослідження дозволили підвищити точність і частоту помилок YOLOv3 при виявленні транспортних засобів.

Інший об'єкт, який часто з'являється на дорозі, а саме вибоїни, був піднятий в якості теми виявлення об'єктів Пінг та ін [298]. В цій роботі було виявлено, що YOLOv3 є найефективнішим алгоритмом, оскільки його швидкість та результати виявлення є більш надійними, ніж у HOG, SSD та Faster-RCNN. Крім того, на узбіччі дороги також часто з'являються інші об'єкти, такі як дерева. Порівняння YOLOv3-SPP та YOLOv3 було проведено Інхуа та ін. [299]. В результаті стверджується, що YOLOv3-SPP є найбільш підходящим для виявлення дерев.

З цих досліджень YOLOv3 був обраний ідеальним алгоритмом виявлення об'єктів або основною темою дослідження. Не тільки для виявлення об'єктів на дорозі, але і для виявлення інших об'єктів, таких як виявлення маски, виконане Лю і Рен [300]. Де YOLOv3 може досягти хорошої продуктивності та меншого часу виведення, ніж Faster-RCNN.

Було проведено не так багато досліджень нового покоління YOLO для виявлення об'єктів на дорозі. Коли це відбувається, результати порівняння не настільки задовільні, як, наприклад, дослідження С. Kumar та ін. [301] щодо виявлення кількох об'єктів на камерах спостереження. Оскільки пояснення експериментального методу є неповним, а показники, що використовуються при оцінці, не стандартизовані. Однак при виявленні інших об'єктів нове покоління YOLO було добре порівняне, наприклад, порівняння YOLOv4 і YOLOv5 при виявленні ізоляторів Рахманом та ін [302].

Було проведено багато досліджень у порівнянні виявлення об'єктів YOLO. Можна сказати, що порівняльне дослідження виявлення об'єктів, яке часто з'являється в дорожньо-транспортних пригодах, проводиться на YOLOv3, FASTER-RCNN і SSD MobileNet. Хоча в кількох дослідженнях використовується алгоритм YOLOv5, лише деякі з них були зосереджені на порівнянні алгоритму YOLOv5 з його попередниками, явно виявляючи безліч об'єктів на дорозі.

Метрики оцінювання

Для оцінки алгоритмів використовуються кілька показників продуктивності. У цьому дослідженні було обрано Precision, recall, F1-score, mAP

та час виведення. Показники продуктивності будуть оцінюватися на основі тестового набору даних.

$$Precision = \frac{TruePositive}{TruePositive+FalsePositive} \quad (1)$$

$$Recall = \frac{TruePositive}{TruePositive+FalseNegative} \quad (2)$$

$$F1 - Score = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad (3)$$

$$mAP = \frac{\sum_{q=1}^Q AveP(q)}{Q} \quad (4)$$

$$Energy\ efficiency = \frac{Average\ power\ (Joule/Second)}{Inference\ speed\ (Frame/Second)} \quad (5)$$

Дослідники в своїх роботах часто використовують таку метрику, як чутливість (sensitivity). Насправді чутливість і повнота оцінюють одне і те ж, відмінність в найменуванні виникло через приналежність цих термінів до різних галузей науки. Так само зустрічаються назви True positive rate або TPR (оцінка вірно позитивних) і Probability of detection (ймовірність виявлення).

Рівняння (1), Precision визначає відсоток правильних прогнозів порівняно з хибнопозитивними та істинно позитивними, які вимірюють наскільки точні результати прогнозування. Рівняння (2) визначає, наскільки добре алгоритм знаходить усі позитивні випадки.

Рівняння (3), Оцінка F1 - це баланс між точністю і повнотою, значення знаходиться в діапазоні від 0 до 1. Більш високий бал F1 - означає кращий баланс точності та повноти. F-міра-зважене гармонійне середнє повноти і точності. Цей показник демонструє, як багато випадків прогнозується моделлю правильно, і скільки справжніх екземплярів модель не пропустить. F-міра об'єднує в собі інформацію про повноту і точності використовуваної моделі.

Специфічність - це показник, що відображає точність роботи моделі бінарної класифікації. Специфічність визначається як відношення числа істинно-негативних класифікацій TN (true negative) до загальної кількості негативних

класифікацій, тобто сумі істинно-негативних і хибнопозитивних FP (false positive, помилок I роду) класифікацій:

$$Specificity = \frac{TrueNegative}{TrueNegative + FalsePositive}$$

Якщо число помилково-позитивних класифікацій велике, тобто модель допустила велику кількість помилок, розпізнавши негативні приклади як позитивні, то специфічність прагне до 0. Навпаки, якщо ж число помилково-позитивних спостережень прагне до 0, то специфічність прагне до 1.

Модель, що володіє високою специфічністю (тобто допускає мале число помилок I роду), забезпечує більшу ймовірність правильного розпізнавання для негативних спостережень.

Для вимірювання точності детекторів об'єктів використовується рівняння (4) - середнє значення точності по всіх класах. mAP вважається мірою для аналізу загальної продуктивності алгоритму виявлення об'єктів. У цьому дослідженні також була використана IoU (Intersection-over-Union) – метрика ступеня перетину між двома обмежувальними рамками. Відсоток збігу площі перетину між прогнозованим прямокутником і прямокутником з істинним підставою вимірюється IoU. mAP@.5 означає, що поріг IoU для середньої точності дорівнює 0,5.

Для перевірки обчислювальної ефективності моделі обчислюється кількість операцій з плаваючою комою в секунду (FLOPs) шляхом ділення часу обчислень GPU на загальну кількість FLOPs. Розмір моделі обчислюється на основі загальної кількості параметрів з проміжними картами активації. Для об'єктивного порівняння, FLOPs і обсяг пам'яті (або використання основної пам'яті) розраховуються для пакета розміром в одиницю. Як правило, енергоспоживання графічного процесора вимірюється за допомогою інтерфейсу системного монітора Nvidia (nvidia-smi). Значення потужності вибирається з фіксованим інтервалом в 0,1 секунди і обчислюється середня потужність

графічного процесора. Споживання енергії на зображення вимірюється як відношення середньої потужності до швидкості виводу, як визначено у (5). Енергоефективність визначається як загальна кількість енергії, споживаної для обробки вхідного зображення; більш низьке значення енергоспоживання означає більшу енергоефективність.

Опис наборів даних. Вибір набору даних вважається дуже необхідним для виконання різних завдань розпізнавання об'єктів в режимі реального часу. У таблиці 1 показано порівняння декількох різних наборів даних, доступних для додатків автономного водіння, в першу чергу для виявлення дорожніх об'єктів. Анотації включають лише чотири класи (залежно від наявності), а саме транспортний засіб, пішохід, дорожній знак та світлофор. Порівняно з іншими наборами даних для експериментальної оцінки вибрано набір даних BDD100K, що містить 100 тис. зображень (70 тис. навчальних зображень, 20 тис. тестових зображень, і 10 тис. перевірочних зображень) з різних географічних, екологічних і погодних сценаріїв, таких як вулиці, тунелі, заправні станції, автостоянки, житлові будинки і шосе в різних умовах в різний час дня і ночі на міських вулицях з реальним рухом; до 90 об'єктів на зображенні. Кожен об'єкт має логічні значення для перекриття та усічення. У цій роботі вибрано набір даних BDD100K через величезну кількість автономних зображень при складних дорожніх сценаріях і різних погодних умовах. Крім того, набір даних надає зображення в реальному часі для тестування алгоритмів, заснованих на глибокому навчанні, в умовах обмежень, що виконують завдання різної складності. Знімок декількох зображень з набору даних BDD100K показаний на рис. 1

Набір даних BDD100K спочатку містить об'єкти, а саме людину, вершника, велосипед, автомобіль, автобус, вантажівку, поїзд, світлофор і дорожній знак. У цій оцінці нерелевантні вибірки виключаються через дисбаланс класів і, зокрема, чотирьох класів об'єктів, які вважаються найбільш важливими, а саме транспортний засіб (велосипед, легковий автомобіль, автобус, вантажівка), пішохід (людина і вершник), дорожній знак і світлофор зберігаються. Варто зазначити, що кілька зображень у наборі даних визнані недійсними (неправильна

мітка або координати). Після визначення кількості анотованих об'єктів, які розглядаються в цій роботі, вони перераховані в таблиці 2.

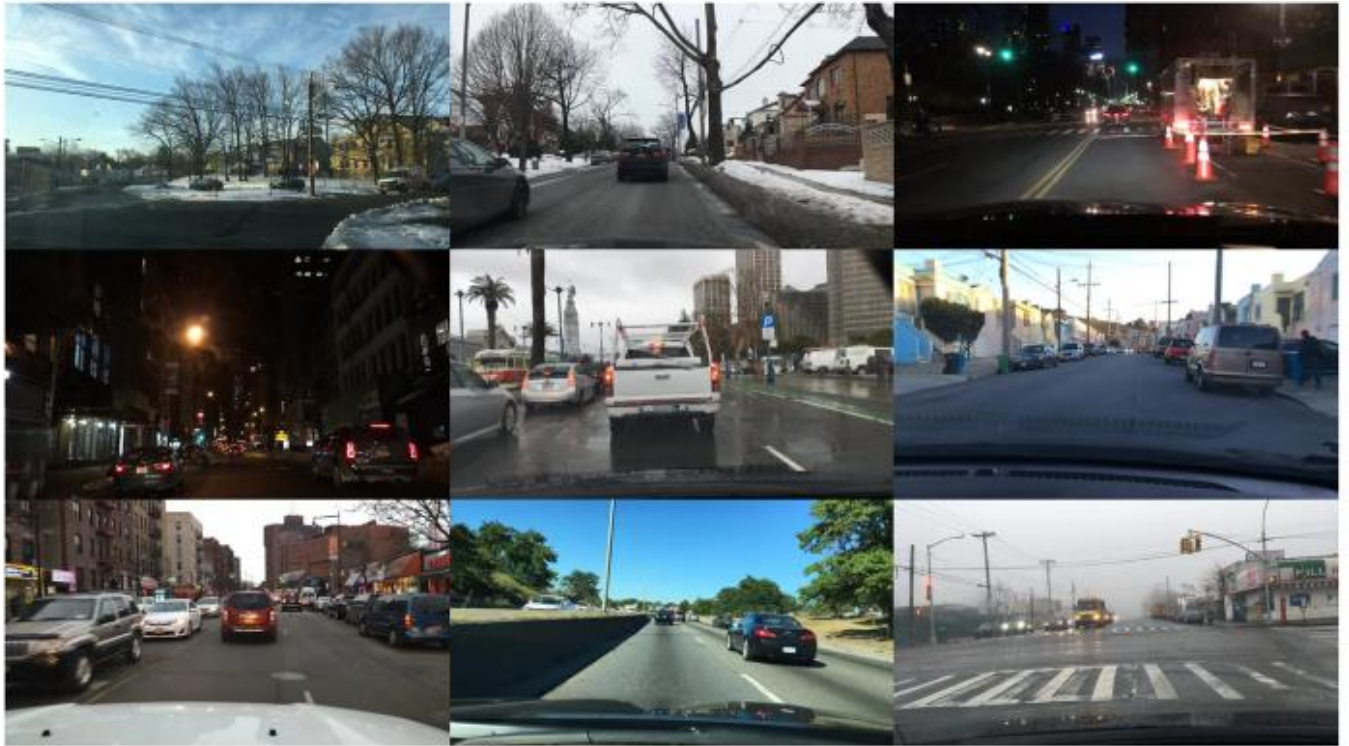


Рисунок 1. Знімок відео з набору даних BDD100K

Таблиця 1.

Наборів даних для виявлення дорожніх об'єктів в режимі онлайн

Набір даних	рік	зображень (x1000)	Класи	Анотації (М)	Різноманітність погоди
Kitti	2012	15	8	0,2	низька
BDD100K	2018	100	10	1,8	висока
ApolloScape	2018	144	27	2.1	низька
EuroCity	2019	47	30	0,8	середня
nuScenes	2019	1386	23	224	середня

Кількість анотованих об'єктів у наборі даних BDD100K

Набір даних	транспортні засоби	пішоходи	дорожні знаки	світлофори
Навчання	765,066	95,866	239,686	186,117
Тестування	218,587	27,524	68,984	53,291
Валідація	109,807	13,911	34,908	26,885

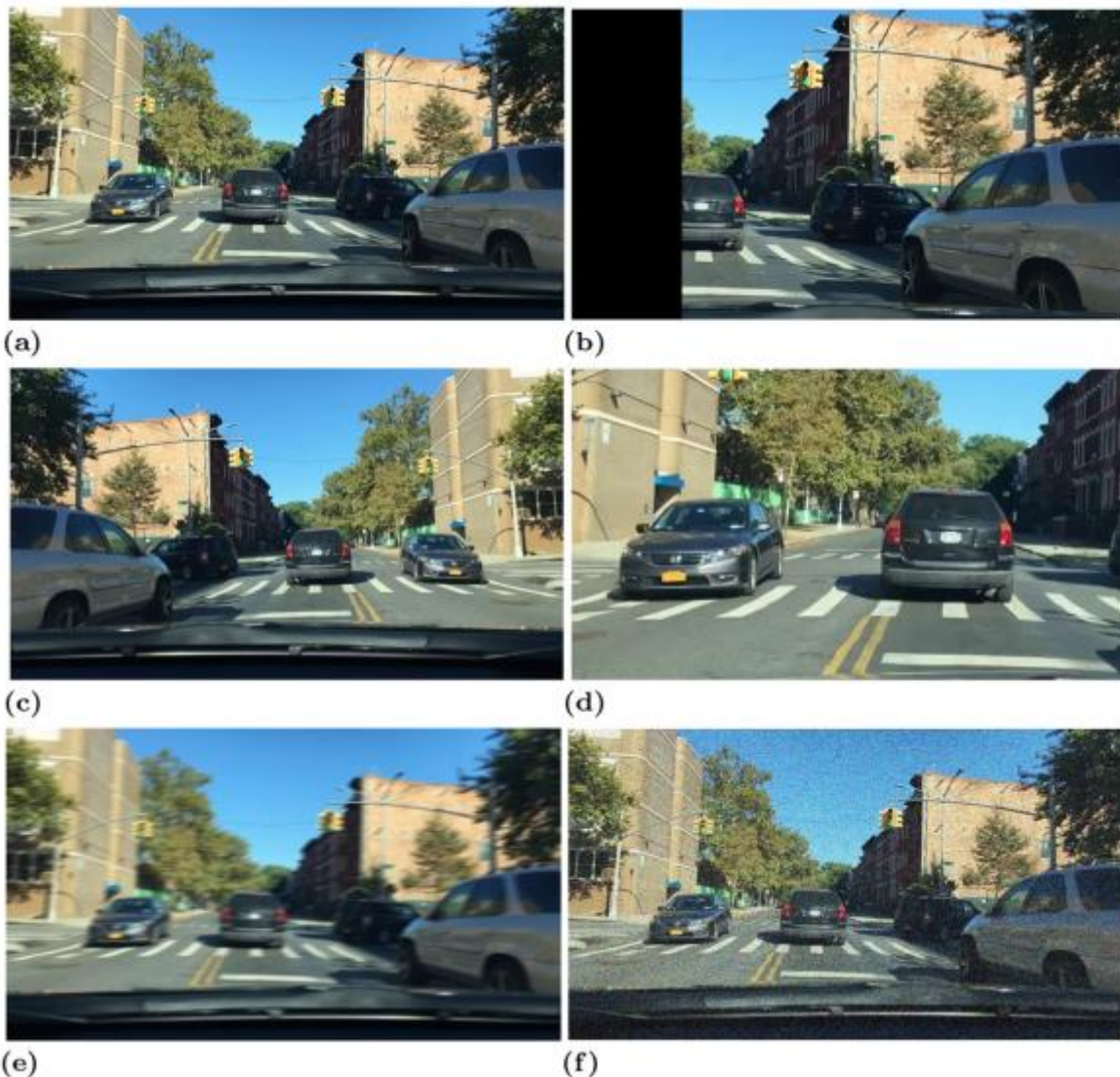


Рисунок 2. Методи збільшення даних (а) оригінальне зображення (б) зміщення
(с) горизонтальне перевертання (д) випадкова обрізка (е) розмиття руху (ф)
додавання випадкового шуму

Збільшення обсягу даних-це набір різних методів, що використовуються
для збільшення розміру та якості навчальних моделей. Мета полягає в тому, щоб

збільшити здатність до узагальнення без зміни категорії зображень. У цій роботі виконується кілька основних операцій зі збільшення набір даних BDD100K, що включає зміщення зображення, горизонтальне перевертання, випадкове масштабування, випадкове обрізання, розмиття руху та додавання випадкового шуму, як показано на рис. 2. На додаток до основних методів збільшення, у цій роботі використовується Мозаїчне збільшення даних.

Мозаїка - це новий метод збільшення даних, запропонований Гленном Йохером у 2020 році, який використовує чотири навчальні зображення замість одного зображення. Цей метод має дві основні переваги.

По-перше, модель вчиться розпізнавати об'єкти поза звичайними контекстами, знайденими в наборі даних. По-друге, це покращує статистику пакетної нормалізації за допомогою невеликих міні-пакетів, що робить можливим навчання на одному графічному процесорі.

Експериментальна установка. Популярні фреймворки глибокого навчання Tensorflow та Caffe використовуються для навчання моделей відповідно до офіційної документації. Експериментальне середовище складається з CUDA v10.1, cuDNN v7.6.4, OpenCV v4.5.0, ОС Ubuntu 18.04.5, 16 ГБ оперативної пам'яті, Intel i7-8750H центральний процесор і графічний процесор Nvidia GeForce RTX2080ti. Збільшення застосовується до навчальних та перевірочних зображень для підвищення здатності до навчання. Тому для прискорення цієї оцінки використовуються оптимальні розміри пакетів 2 (для Mask-rcnn), 4 (для R-FCN і RetinaNet) і 8 (для SSD і YOLOv4) . Стохастичний градієнтний спуск (SGD) використовується як оптимізатор під час навчання. Початкове значення швидкості навчання встановлено на 0,1, швидкість імпульсу встановлена на 0,949, а швидкість загасання встановлена на 0,0001, щоб зменшити втрати навчання та запобігти вибуху градієнта. Швидкість навчання знижується на 0,5 на кожну 50-ту епоху. Жодні інші параметри не були змінені та прийняті фреймворками за замовчуванням. Навчання тривало 200 епох.

Після навчання обчислюються ваги мережі для перевірки експериментальних результатів на зображеннях з тестового набору даних BDD100K.

У таблиці 3 наведена продуктивність алгоритмів виявлення об'єктів з урахуванням часу виведення на зображення і швидкості в кадрах в секунду на CPU і GPU-процесорі. У таблиці 4 наведені чутливість і специфічність для кожного класу, А також середні чутливість і специфічність моделей виявлення об'єктів. У таблиці 5 наведено порівняння з точки зору архітектури, використаної в цьому експерименті, розміру моделі, обсягу пам'яті, збоїв та енергоефективності. Всі результати отримані на зображеннях з тестового набору даних BDD100K. найкращі значення в таблицях виділені жирним шрифтом.

Таблиця 3.

Час виведення та швидкість моделей виявлення об'єктів на наборі даних
BDD 100k

Модель	процесорний час (мс)	Час роботи графічного процесора (мс)	швидкість процесора (кадр / сек)	швидкість графічного процесора (кадр / сек)
R-FCN	2967,36	178,96	3,37	5,58
Mask R-CNN	3412,57	326,78	2,93	3,06
SSD	72.98	19.12	13.70	52.31
RetinaNet	3686.12	97.66	2.71	10.24
YOLOv4	121.53	20.19	8.23	49.52

Таблиця 4.

Продуктивність моделей виявлення об'єктів на наборі даних BDD 100k

Модель	Клас	Чутливість (%)	специфічність (%)
R-FCN	транспортний засіб	68.20	70.24
	Пішохід	62.69	63.16
	Дорожній знак	58.43	60.87
	Світлофор	55.17	56.46
	середнє значення	61,14	62,68

Продовження таблиці 4

Mask R-CNN	транспортний засіб	89.21	83.26
	Пішохід	86,58	80,32
	Дорожній знак	82.64	75.26
	Світлофор	79.76	73.15
	середнє значення	84,58	77,80
SSD	транспортний засіб	63.21	66.21
	Пішохід	57,82	61,84
	Дорожній знак	46.81	50.36
	Світлофор	42.49	44.27
	середнє значення	54,59	55,69
RetinaNet	транспортний засіб	81.19	85.12
	Пішохід	80.27	82.16
	Дорожній знак	79.81	81.49
	Світлофор	79.38	80.62
	середнє значення	80,17	82,35
YOLOv4	транспортний засіб	91.24	92.49
	Пішохід	88.19	89.14
	Дорожній знак	85.65	86.20
	Світлофор	83.14	84.16
	середнє значення	87,06	87,90

Таблиця 5.

Порівняння моделей виявлення об'єктів з точки зору енергоефективності
та інших аспектів

Модель	Розмір базової моделі (МБ)	Обсяг оперативної пам'яті (ГБ)	FLOPs (G)	Енергоефективність (J/image)
R-FCN	436.12	3.89	306.81	127.64
Mask R-CNN	573.48	5.34	614.23	168.19
SSD	136.04	0.73	95.86	5.38
RetinaNet	471.74	3.32	359.32	47.31
YOLOv4	269.53	1.89	179.65	7.16

Двоступеневі детектори, як правило, демонстрували кращу точність виявлення порівняно з одноступінчатыми детекторами. Згідно з наведеними вище експериментальними результатами, спостерігається, що одноступінчатий алгоритм виявлення YOLOv4 має видатну продуктивність з точки зору швидкості і точності, а також досить високу енергоефективність в порівнянні з іншими одноступінчатыми і двоступеневими алгоритмами виявлення дорожніх об'єктів, оскільки він використовує CSPDarkNet-53, яка підвищує точність класифікатора і детектора. Більше того, додавання блоку SPP за допомогою PANet збільшує поле сприйняття, яке відокремлює найважливіші контекстні функції без зменшення швидкості мережі. Завдяки цьому точність виявлення YOLOv4 перевершує двоступеневі детектори, а перевага полягає у високій швидкості виявлення в наборі даних BDD100K. двоступеневий алгоритм виявлення Mask R-CNN демонструє хорошу точність виявлення, оскільки він використовує RoIAlign для отримання більш точного вирівнювання меж від пікселя до пікселя.

SSD демонструє низьку точність і чуйність, оскільки модель виконує виявлення середніх і малих цілей з дуже невеликої кількості шарів; отже, вона не має достатньо семантичної та крайової інформації для виявлення більшості дорожніх об'єктів. Тоді як R-FCN демонструє низьку точність та чуйність, оскільки не використовує глобальний середній пул. Таким чином, остаточні прогнози базуються на картах об'єктів, чутливих до місця розташування, що зменшує загальну точність та чуйність. Mask R-CNN демонструє низьку точність, оскільки вона ігнорує просторову інформацію між сприймаючими полями, а це означає, що вона не враховує деталі екземпляра між крайовими пікселями. Крім того, Mask R-CNN використовує багатомасштабний FPN, а модуль RoIAlign зменшує невідповідність між достовірністю класифікації та прогнозованими блоками, покращуючи виявлення середніх та малих цілей. Частота відгуків Mask R-CNN близька до YOLOv4 для виявлення об'єктів із закриттям і усіченням, оскільки кількість помилкових виявлень значно перевищує кількість правильних виявлень; тому його не можна вважати

збалансованим алгоритмом. У цій роботі виявлено, що точність двоступеневих моделей нижча, ніж одноступеневих. Зі зміною рівня виявлення продуктивність знижується при перетворенні перекриття та усічення, особливо для SSD, оскільки він почав пропускати цілі. SSD демонструє високу точність і низький рівень відкликання, тим самим збільшуючи частоту виявлення промахів, що вказує на те, що він був більш неконтрольованим при виявленні середніх і дрібних цілей; отже, його не можна вважати збалансованим алгоритмом. Навпаки, в RetinaNet володіє високою адаптивністю до багатомасштабних зображень завдяки використанню FPN і Focal; таким чином, він забезпечує хороші результати при виявленні середніх і великих цілей. RetinaNet можна розглядати як збалансований алгоритм.

Тим не менш, YOLOv4 є більш збалансованим алгоритмом, ніж інші моделі, оскільки він забезпечує високу точність при швидкому відтворенні на всіх рівнях категорії для всіх дорожніх об'єктів.

З таблиці 5 зрозуміло, що SSD можна вважати швидким алгоритмом виявлення об'єктів у двоступеневих та одноступеневих моделях виявлення без будь-яких додаткових вимог до обладнання.

Швидкість виявлення YOLOv4 майже така ж, як SSD з вимогами до графічного процесора. Швидкість виявлення Mask R-CNN низька через використання додаткової гілки модулем RoIAlign, що збільшує час обчислень для остаточної класифікації та регресії обмежувального поля. YOLOv4 демонструє високу чутливість і специфічність у всіх класах. Тим часом двоетапний алгоритм виявлення Mask R-CNN показує низьку середню специфічність, оскільки RPN не може виконати регресію обмежувального поля для невеликих об'єктів, що вказує на велику кількість помилкових виявлень. RetinaNet демонструє більш високу середню специфічність, ніж Mask R-CNN, через жорсткий відбір негативних класів під час навчання.

Mask детектора R-CNN має великий розмір моделі та великий обсяг пам'яті. Крім того, маска R-CNN показує велику кількість провалів оскільки вони використовують надлишкові сприйнятливості поля для виявлення об'єктів, що

перекриваються, що призводить до збільшення обчислень та збільшення споживання енергії. У R-FCN використовується велика кількість обчислень RPN і відсутня оптимізація шкал і коефіцієнтів прив'язки, що призводить до неефективного споживання енергії. SSD має невеликий розмір моделі і меншу кількість збоїв, що забезпечує високу енергоефективність. Хоча YOLOv4 використовує трохи більше енергії, ніж SSD, завдяки більшій кількості параметрів, він дає кращі результати, ніж інші моделі виявлення об'єктів. RetinaNet використовує RPN для фільтрації багатьох негативних пропозицій-кандидатів, що призводить до більшої кількості збоїв і енергоспоживання, ніж в інших моделях одноступінчастого виявлення.

Для подальшого аналізу ефективності виявлення в цій роботі використовуються конкретні тестові зразки, аналогічні реальному дорожньому сценарію з набору тестових даних BDD100K.

Таблиця 6.

Короткий опис різних характеристик моделей виявлення об'єктів

Модель	Точність	швидкість	складність	ефективність
R-FCN	низька	повільна	середня	низька
Mask R-CNN	середня	повільна	висока	низька
SSD	низька	швидкий	низька	висока
RetinaNet	середня	помірна	висока	середня
YOLOv4	висока	швидкий	середня	висока

Згідно з оцінкою наведених вище експериментальних результатів, у таблиці 6 узагальнено порівняння різних алгоритмів виявлення об'єктів з точки зору загальної точності виявлення (середнє значення mAP), швидкості виведення (швидкість графічного процесора), обчислювальної складності (кількість збоїв) та ефективності (енергоефективність). Критерії загальної точності виявлення встановлені як низькі (<50), середні (≥ 50 і < 75) і високі (≥ 75). Критерії швидкості виведення встановлюються як швидка (>30), помірна (≤ 30 і > 10) і повільна (≤ 10). Критерії обчислювальної складності встановлені як низька (<100), середня (≥ 100).

$i < 350$) і висока (≥ 350). Критерії енергоефективності встановлені як низькі (> 50), середні (≤ 50 і > 10) і високі (≤ 10).

Таким чином, модель одноступінчастого виявлення YOLOv4 продемонструвала видатну точність при високій швидкості виявлення та високій енергоефективності, задовольняючи сучасним стандартам виявлення об'єктів у режимі реального часу.

Виконаємо порівняльний аналіз моделей одноступінчастого виявлення YOLO.

З метою визначення найшвидших моделей YOLO для кожної GPU, враховуючи швидкість та продуктивність, базові моделі YOLO перевіряються на GPU від NVIDIA, таких як TESLA P100, TESLA V100, GTX 1080Ti та RTX 4090.

Різні версії YOLO, такі як YOLOv4, YOLOv5, YOLOv6 та YOLOv7, пропонують компроміси між швидкістю та точністю. Наприклад, варіанти Nano та Tiny акцентують легкість та швидку продуктивність, тоді як наступні версії, наприклад YOLOv7, забезпечують вищу точність, виходячи за межі невеликого зниження швидкості.

Як показано на рисунку 4, графік надає інформацію про продуктивність різних моделей YOLO на різних пристроях GPU.

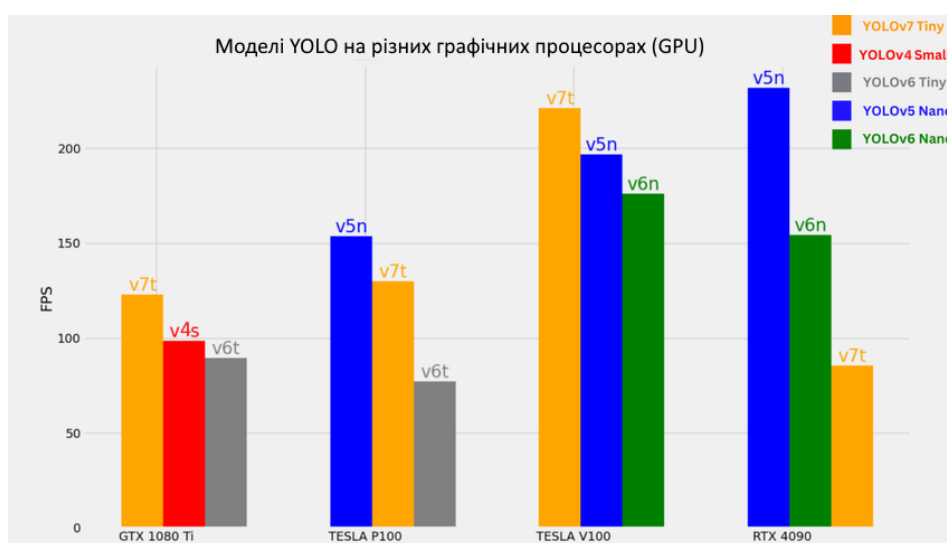


Рисунок 4. Моделі YOLO на різних графічних платформах

По-перше, з графіка видно, що YOLOv5 видає найкращі результати за швидкістю на GPU RTX 4090 та TESLA P100. Це свідчить про те, що, якщо основний критерій - досягнення найвищої кількості кадрів на секунду (FPS) для точного виявлення в реальному часі, YOLOv5 Nano стане оптимальним вибором. По-друге, YOLOv7 Tiny виділяється як модель, яка надає найвищу пропускну здатність на GTX 1080 Ti та TESLA V100. Пропускна здатність визначається кількістю об'єктів, виявлених за одиницю часу, і YOLOv7 Tiny в цьому виявляється найкращим для цих конкретних пристроїв GPU. Це особливо корисно в сценаріях, коли точне виявлення більшої кількості об'єктів важливіше, ніж досягнення максимальної кількості кадрів на секунду.

На противагу, моделі YOLOv6 Nano та Tiny, хоч і не досягають такої ж швидкості, що YOLOv5 та YOLOv7, але все ж не вважаються дуже повільними. Хоча графік не містить точних даних про їхню продуктивність, він вказує на те, що ці моделі забезпечують баланс між швидкістю та точністю. Вони можуть бути чудовим вибором, коли потрібна помірна швидкість, при цьому отримуючи задовільні результати об'єктивного виявлення.

Узагальнюючи, кращий вибір залежить від конкретних вимог завдання. YOLOv5 Nano ідеально підходить для реальних застосувань, де важлива максимальна кількість кадрів на секунду. YOLOv7 Tiny видається корисним у сценаріях, де важлива висока пропускна здатність більше, ніж продуктивність в реальному часі. Тим часом моделі YOLOv6 Nano і Tiny пропонують компроміс між швидкістю та точністю, що робить їх прийнятними у випадках, коли потрібна помірна швидкість без значних втрат у якості виявлення.

Як зображено на рисунку 5, на платформі процесорів, модель YOLOv5 отримує найвищий показник швидкості. Ця модель може працювати в реальному часі, перевищуючи 30 кадрів на секунду. Це означає, що вона може обробляти та аналізувати зображення або відеопотоки в реальному часі, що надає швидкі результати об'єктивного виявлення. Модель YOLOv5 добре оптимізована для ефективності та швидкості, що робить її добре пристосованою для процесорів споживчого класу, де реальний час має велике значення.

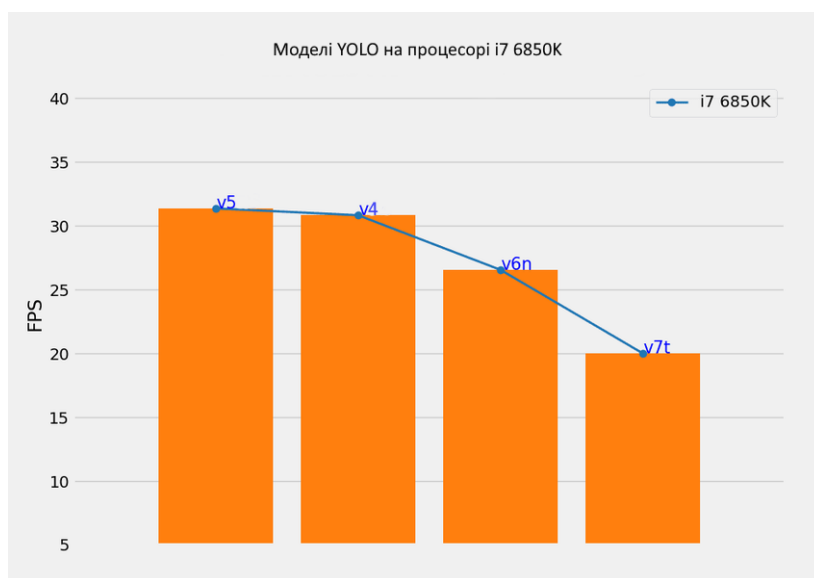


Рисунок 5. Чотири найшвидші моделі виявлення об'єктів YOLO на процесорі i7 6850K

З іншого боку, модель YOLOv7 Tiny, хоч і здатна до об'єктивного виявлення в реальному часі, працює трохи повільніше в порівнянні з моделлю YOLOv5. Зазвичай вона працює на рівні близько 20 кадрів на секунду, що все ж вражає і підходить для багатьох реальних застосувань. Хоча вона може не досягати такої швидкості, як YOLOv5, модель YOLOv7 Tiny забезпечує баланс між швидкістю та точністю, забезпечуючи задовільні результати при ефективній обробці на загальнодоступних процесорах.

Згідно з наданими результатами виявлено, що менші моделі, як правило, проявляють вищу продуктивність. Це виявлено у тестах, де моделі YOLOv5 Nano та YOLOv6 Nano виявилися найшвидшими альтернативами. Важливо зазначити,

що навіть на застарілій версії процесора i7 ці моделі здатні досягти вражаючої швидкості понад 30 кадрів на секунду (FPS). Це підтверджує ефективність та оптимізацію моделей YOLOv5, v6 Nano для обробки на процесорах масового споживання, роблячи їх чудовими виборами для об'єктивного виявлення в реальному часі на таких процесорах.

YOLOv5 збалансований алгоритм, надаючи високу точність при швидкому виявленні об'єктів на всіх рівнях категорій дорожніх об'єктів.

З урахуванням проведених досліджень, YOLOv5 виявляє найкращу швидкість роботи, що робить її найбільш оптимальним варіантом для сценаріїв, де вирішальне значення має максимальна кількість кадрів на секунду для точного виявлення в реальному часі.

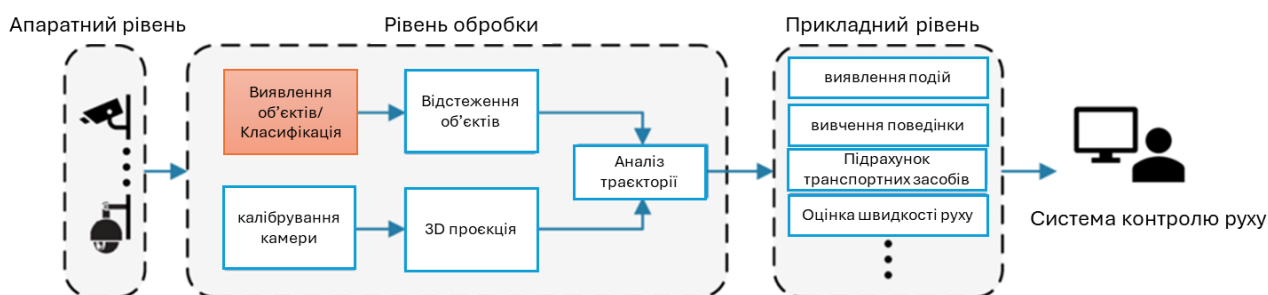


Рисунок 6. Загальна архітектура інтелектуальних систем моніторингу дорожнього руху.

У цьому дослідженні ми пропонуємо використовувати модель YOLOV5 як потенційне рішення для підсистеми виявлення та класифікації об'єктів (рис.6), яка є складовою більш складних інтелектуальних систем моніторингу дорожнього руху. Виявлення об'єктів є ключовим компонентом у розробці цих систем, і виконання подальших кроків значною мірою залежить від виконання цього завдання.

Ідентифікація об'єктів часто розглядається як дуже важливий аспект комп'ютерного зору, насамперед через його широкий спектр практичних застосувань. Термін "ідентифікація об'єктів" відноситься до здатності програмних систем точно визначати просторові координати об'єктів всередині

зображення або сцени, а також класифікувати і маркувати кожен ідентифікований об'єкт.

Ідентифікація осіб, виявлення транспортних засобів, підрахунок пішоходів, онлайн-фотографії, системи безпеки та автомобілі без водія - все це отримало широке застосування. Виявлення об'єктів має широкий спектр застосувань у кількох областях практики. Як і у випадку з іншими комп'ютерними технологіями, очікується, що зусилля програмістів та розробників програмного забезпечення забезпечать широкий спектр інноваційних та чудових програм для виявлення об'єктів.

Використання сучасних методик виявлення об'єктів в різних додатках і системах, а також створення нових додатків, заснованих на цих методологіях, являє собою нетривіальну задачу. На початкових етапах виявлення об'єктів використовувалися класичні методи, зокрема, бібліотеки комп'ютерного зору. Проте, традиційні алгоритми демонстрували недостатню продуктивність при адаптації до різних ситуацій. Пропонована система має ряд переваг: досягнення високого рівня точності та швидкості.

Архітектура системи для інтелектуального виявлення об'єктів у реальному часі на базі YOLOv5 включає кілька ключових компонентів, які взаємодіють між собою для оптимальної функціональності (рис. 7):

Джерела вхідних даних. Цей компонент включає в себе стрічки з камер в реальному часі та відеопотоки з попередньо записаних джерел. Він забезпечує надходження вхідних даних у систему.

Модуль виявлення об'єктів. Основний модуль, що використовує YOLOv5 для точного виявлення та класифікації об'єктів у реальному часі.

Підтримка настоюваних об'єктів. Цей компонент надає можливість навчання та додавання користувацьких класів об'єктів для більш широкого розпізнавання.

Модуль попередньої обробки відповідає за покращення зображень для оптимізації процесу виявлення об'єктів шляхом налаштування розміру, зменшення шуму та підвищення контрастності.

Синхронізація кадрів забезпечує синхронізацію кадрів з різних камер для вирівнювання даних та забезпечення їхньої правильної обробки.

Модуль відстеження об'єктів відстежує та моніторить виявлені об'єкти за допомогою алгоритмів відстеження, наприклад, DeepSORT, що допомагає відслідковувати об'єкти в часі.

Інтерфейс користувача забезпечує зручний спосіб налаштування системи, моніторингу та перегляду результатів виявлення об'єктів через веб-панель.

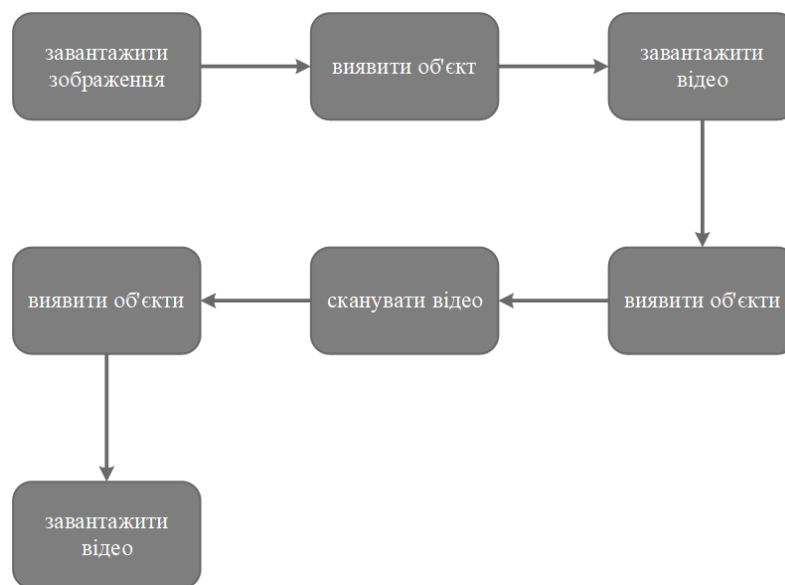


Рисунок 7. Схема робочого процесу системи розпізнавання об'єктів

Ця архітектура надає основу для розробки системи інтелектуального виявлення об'єктів у реальному часі з використанням YOLOv5. Вона може бути налаштована та розширена відповідно до конкретних вимог та потреб для досягнення максимально ефективних результатів.

На рисунках 8 та 9 представлені DFD діаграми для системи розпізнавання об'єктів, які ілюструють потік інформації та функціональні елементи цієї системи на відповідних рівнях представлення.

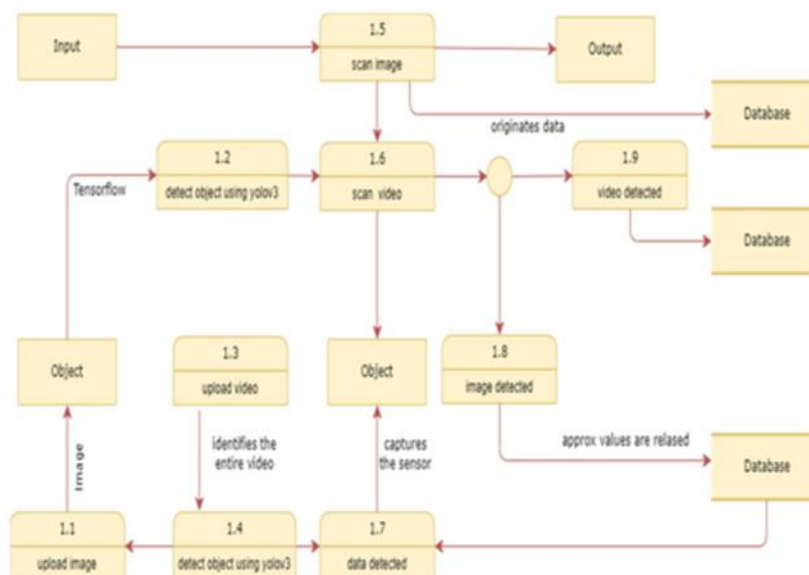


Рисунок 8. DFD діаграма

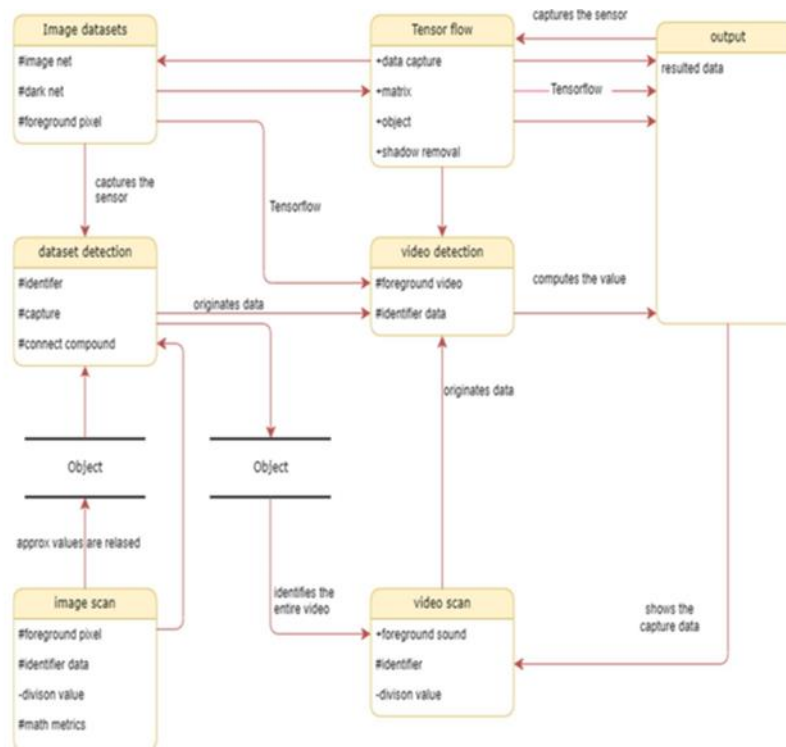


Рисунок 9. Деталізація DFD діаграми

На рисунку 11 наведена діаграма варіантів використання для системи розпізнавання об'єктів, що ілюструє, як система взаємодіє з її користувачами або зовнішніми агентами через різні випадки використання. Це сприяє кращому розумінню того, як система використовується та які функції вона здійснює для своїх користувачів.

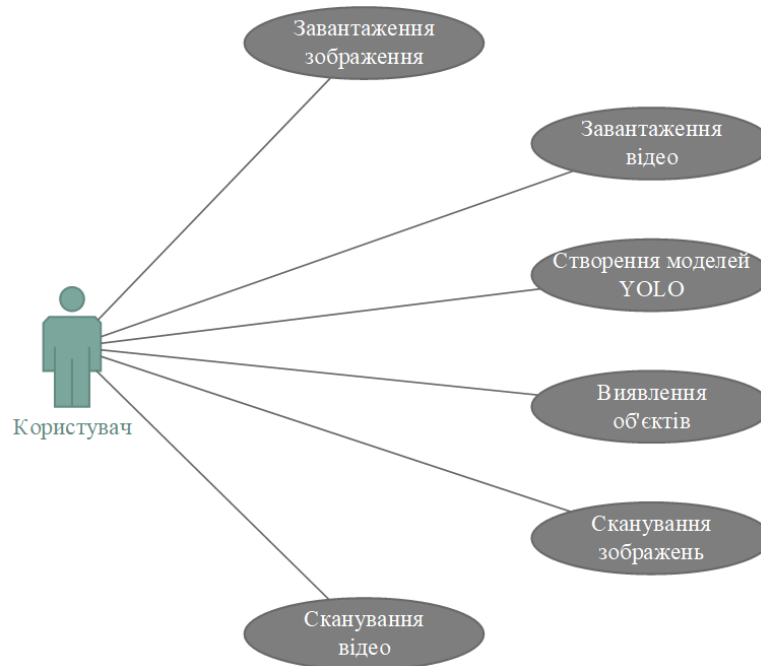


Рисунок 11. Діаграма use-case

На рисунку 12 надана діаграма класів для системи розпізнавання об'єктів, яка ілюструє класи системи, їхні методи та атрибути, а також взаємозв'язки між цими класами. Це допомагає виявити структуру системи та взаємодію між її складовими елементами для більш детального розуміння роботи системи.

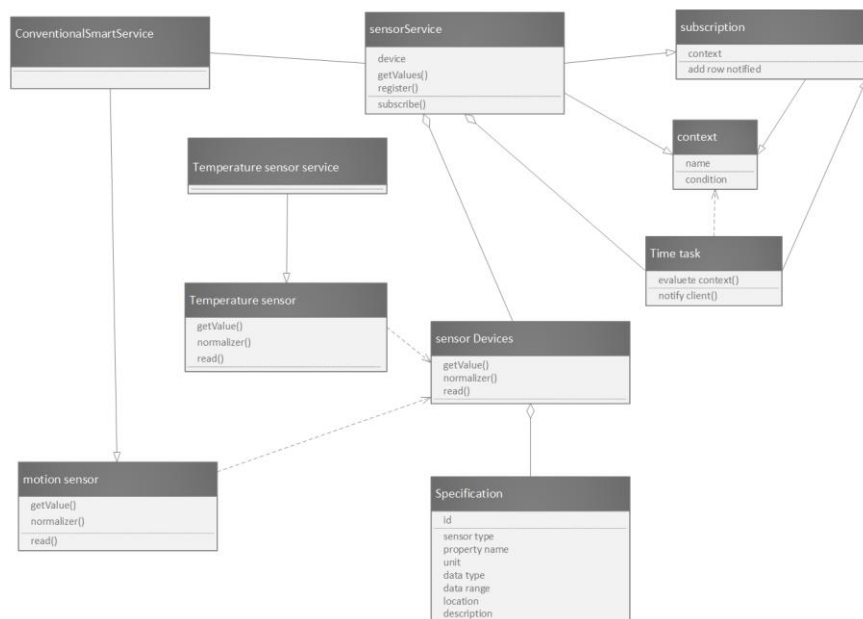


Рисунок 12. Діаграма класів

На рисунку 13 представлена діаграма послідовності для системи розпізнавання об'єктів, які ілюструють взаємодію між різними компонентами системи. Лінії життя представляють об'єкти, які беруть участь у взаємодії, і показують їх активність відносно часу. Повідомлення між об'єктами відображаються стрілками та показують порядок передачі повідомлень.

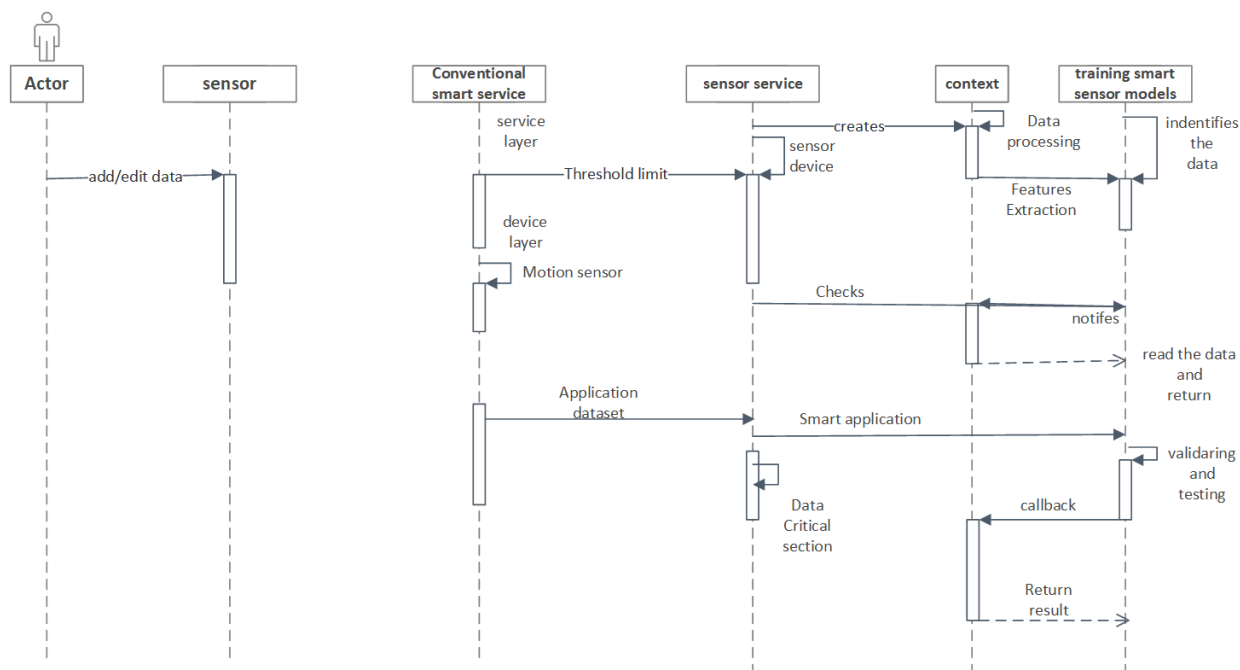


Рисунок 13. Діаграма послідовності

На діаграмі компонентів показано, як компоненти спілкуються один з одним через їхні інтерфейси та порти. Це дозволяє відобразити взаємодію між компонентами та їхню структуру на високому рівні.

Ця діаграма може служити основою для подальшого розроблення деталей архітектури системи та визначення, як компоненти реалізують функціональність системи на більш низькому рівні.

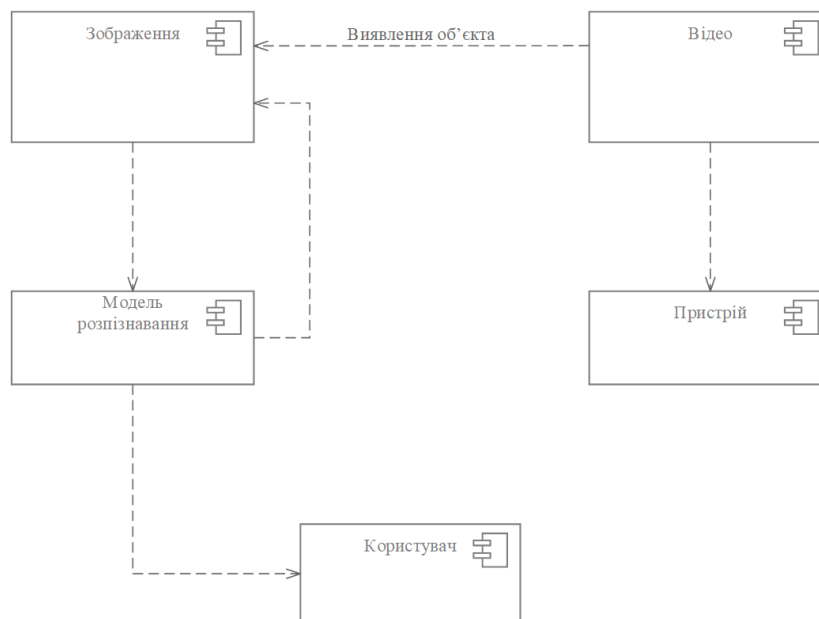


Рисунок 14 – Діаграма компонентів

На діаграмі компонентів для системи розпізнавання об'єктів (рис.14) видно, що дані спочатку передаються великому "прямокутнику", який може представляти собою або саму систему в цілому, або підсистеми та компоненти всієї системи. Це визначається контекстом і рівнем деталізації діаграми.

Аналіз отриманих результатів моделювання. Як зазначається в літературних роботах, тіні від транспортних засобів, щільний рух і умови низької освітленості можуть створювати проблеми при виявленні транспортних засобів в денному режимі. Пропонований метод усуває обмеження виявлення транспортних засобів в умовах щільного руху і наявності довгих тіней, як показано на рис. 15 (а) і (б) відповідно. На ньому зображено точне виявлення транспортних засобів, при цьому йому не заважають довгі тіні через сонячне світло. Крім того, система зарекомендувала себе краще при виявленні транспортних засобів в умовах заторів на дорогах.

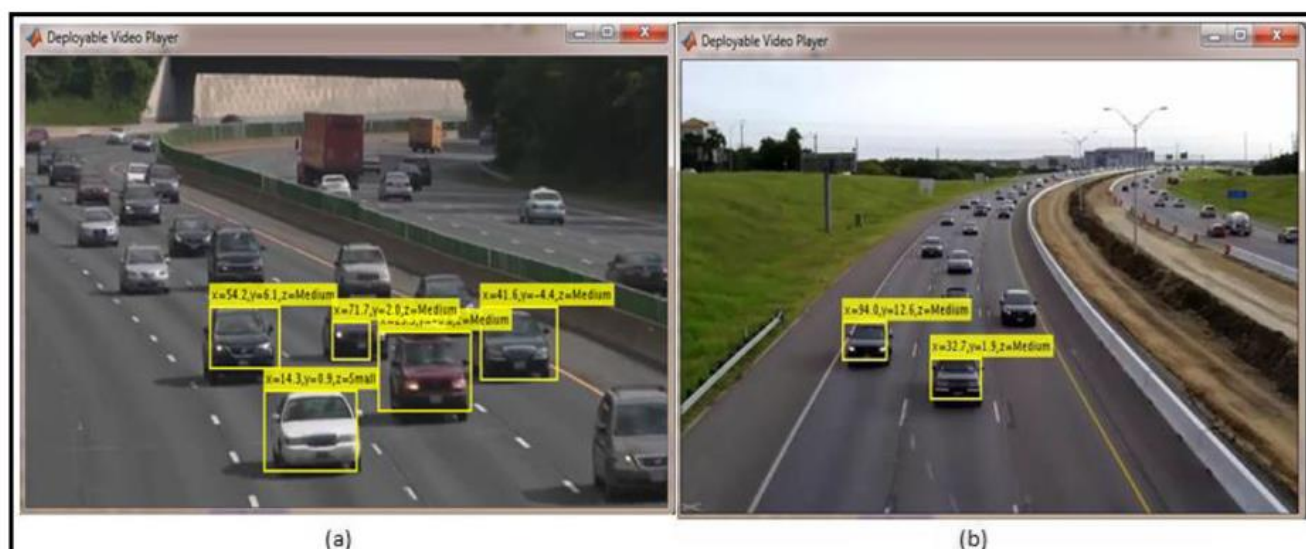


Рисунок 15. Помічені результати виявлення: (а) при наявності пробок на дорогах, (б) виявлення транспортних засобів при наявності довгих тіней.

Запропонована в роботі модель має хорошу надійність порівняно з іншими методами. Запропонована система також включає виявлення транспортних засобів у похмуру погоду. У порівнянні зі звичайним днем, в похмуру погоду освітленість менше. В роботі також розглянуто приклади розпізнавання об'єктів в областях з низькою освітленістю. Проведене дослідження показало хороші результати у виявленні транспортних засобів, як показано на рис. 15. Жовта обмежувальна рамка позначає виявлені транспортні засоби та класифікаційні знаки як малі, середні та великі. На обчислення результатів дослідження пішло 0,62 секунди. Також досягнута точність 94% виявлення транспортних засобів на відео, що містить 345 кадрів розміром 640×360 , як показано на рис. 16. Система отримала значення 98% повноти (recall) і 90% precision при виявленні транспортних засобів.



Рисунок 16. Результати виявлення транспортного засобу в похмуру погоду (при слабкому освітленні)

YOLOv5 славиться своїми важливими функціями, такими як швидке отримання точних результатів. Це час, що витрачається системою на обчислення результатів. Час обробки повинен бути невеликим, щоб система могла видавати бажані результати. Представлений аналіз лінійного графіка, який визначає час обробки, витрачений YOLOv5 на обчислення бажаних результатів у денний час. Графік складається з аналізу всіх трьох випадків. Із графіку видно, що модель точно розпізнає багатоцільові транспортні засоби за менший час обробки (рис. 17).



Рисунок 17. Порівняння часу обчислень при різних умовах

У таблиці нижче представлена загальна продуктивність, досягнута при виявленні транспортних засобів в денному режимі з використанням чотирьох показників, зазначених вище. В роботі розраховані показники оцінки для кожного випадку в денному режимі, і результати наведені в таблиці 7.

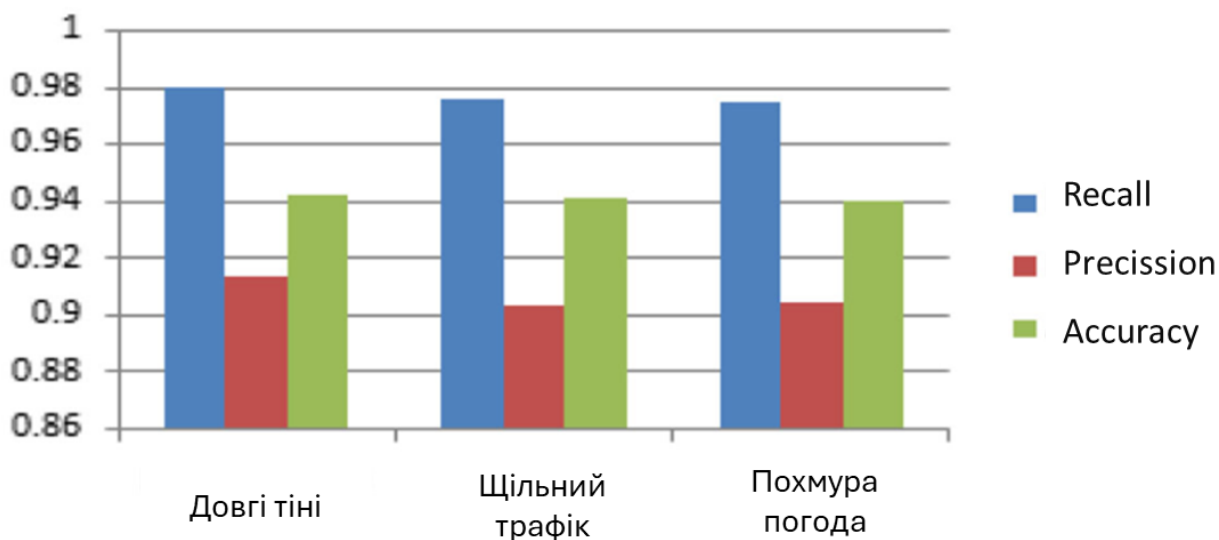


Рисунок 18. Аналіз результатів показників продуктивності при денному режимі освітлення

У таблиці 7 показано оцінку продуктивності за допомогою моделі класифікації YOLOv5 у денному режимі.

Таблиця 7.

Аналіз ефективності різних оціночних показників, що використовуються
при розпізнаванні денного зору.

Денний режим виявлення	час обробки (сек)	Точність (Accuracy)	Повнота (recall)	Точність (Precision)
Виявлення при наявності довгих тіней	0,87	0,94	0,97	0,90
виявлення в умовах щільного руху	0,79	0,93	0,98	0,91
Виявлення в похмуру погоду	0,62	0,94	0,98	0,90
Загальна (середня) продуктивність в денному режимі	0,76	0,94	0,97	0,90

Дані в таблиці є результатом використовуваних показників продуктивності. Дуже мало систем враховують такі фактори, як низька освітленість, затори та тіні від транспортних засобів. Це чітко показує, що запропонований підхід може виявити транспортні засоби і в несприятливих ситуаціях. Щоб система була ефективною і надійною, час обробки повинен бути невеликим, а точність повинна бути високою. Представлення гістограми на рис.4.4 показника продуктивності в денному режимі вказує на те, що час обробки виявився меншим для кожного випадку в пропонуваній системі, а всі інші показники високі.

Системи нічного бачення як і раніше відчують труднощі з виявленням транспортних засобів.

Запропонована методологія є новою, оскільки вона також розроблена та навчена виявляти рухомі транспортні засоби в нічний режим. Низька освітленість викликає ряд проблем при локалізації транспортних засобів, таких як неправильна класифікація, збільшення часу обчислення результатів і багато іншого. В роботі для оцінки продуктивності використовується нічне відео, що містить 1232 кадру роздільною здатністю 426×240 . Візуальне уявлення показано на рис. 19.

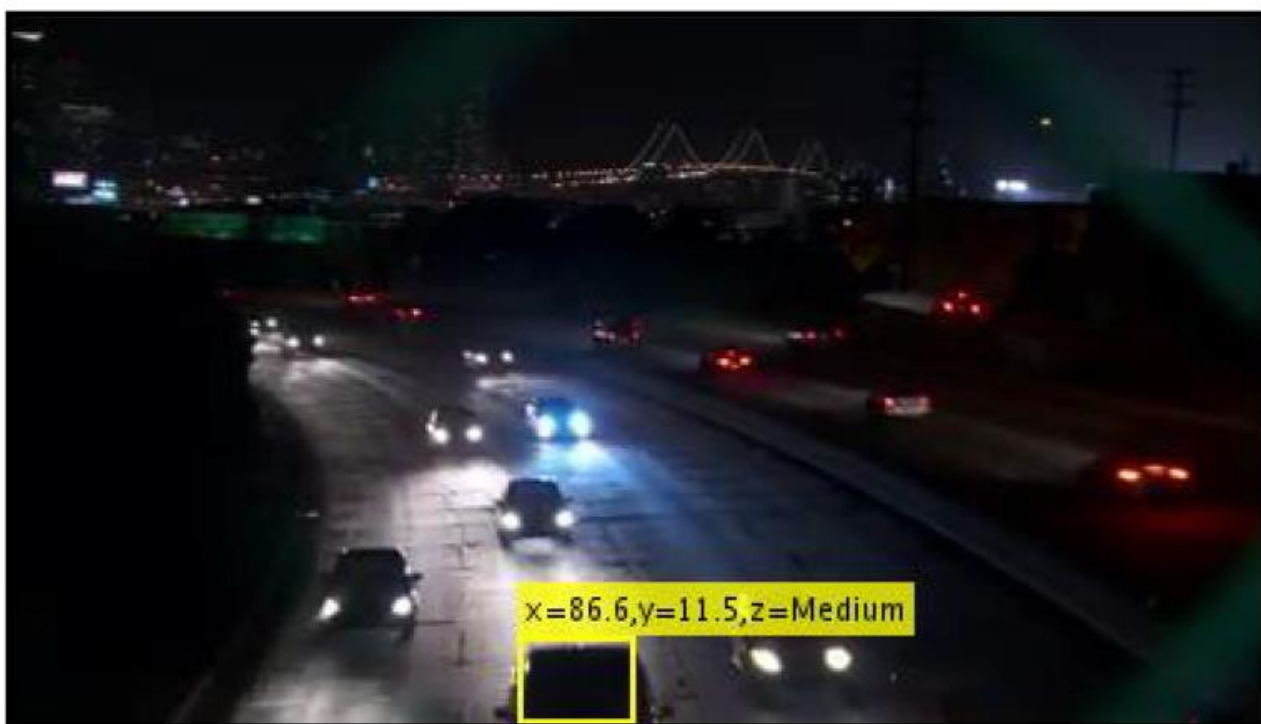


Рисунок 19. Результати виявлення транспортного засобу в нічному режимі

На рисунку 19 показано виявлення транспортних засобів в умовах крайньої темряви, що підтверджує застосовність запропонованого методу. В ході дослідження була досягнута точність виявлення транспортних засобів на рівні 94,35%. Крім того, повнота, точність та вимірний час обчислень становили 99,24%, 90% та 0,39 сек відповідно.

Результати показників ефективності денного режиму були усереднені для всіх трьох випадків.

У таблиці 8 показано відсоток показників ефективності, що використовуються у запропонованій методології.

Для вирішення більшості проблем програмні компоненти системи були розроблені головним чином для денного режиму. Порівняно легко проводити дослідження при хорошому освітленні.

Виключно деякі системи були розроблені для роботи в декількох режимах (вдень і вночі).

Таблиця 8.

Загальний порівняльний аналіз показників продуктивності в денному і нічному режимах

Режим виявлення	час обробки (сек)	Точність (Accuracy)	Повнота (recall)	Точність (Precision)
День	0,76	94,06%	97,64%	90,01%
ніч	0,39	94,35%	99,24%	90%
загальна (середня) продуктивність моделі	0,59	94,20%	98,44%	90%

Запропоноване дослідження об'єднало методологію як для денного, так і для нічного бачення. Система загального призначення була сконструйована з використанням моделі YOLOv5, яка рідко використовується для виявлення транспортних засобів. Дані у всіх наведених вище таблицях показують, що продуктивність запропонованої системи не тільки краще для денного режиму, але і для умов низької освітленості, таких як ніч і похмура погода. Гістограма, що показує загальний сценарій, чітко пояснює, що для кожного режиму виявлення транспортних засобів застосована роботі модель дає найкращі результати. Загальна середня точність, досягнута в денному режимі, становить 94,06%, а в нічному режимі - 94,35%. Середня точність всієї моделі становить 94,20%. Таким чином, були досягнуті цілі щодо підвищення ефективності та швидкості виявлення об'єктів у відеопотоці в різних умовах освітленості, при щільному дорожньому русі і при наявності довгих тіней і інше (рис.20).

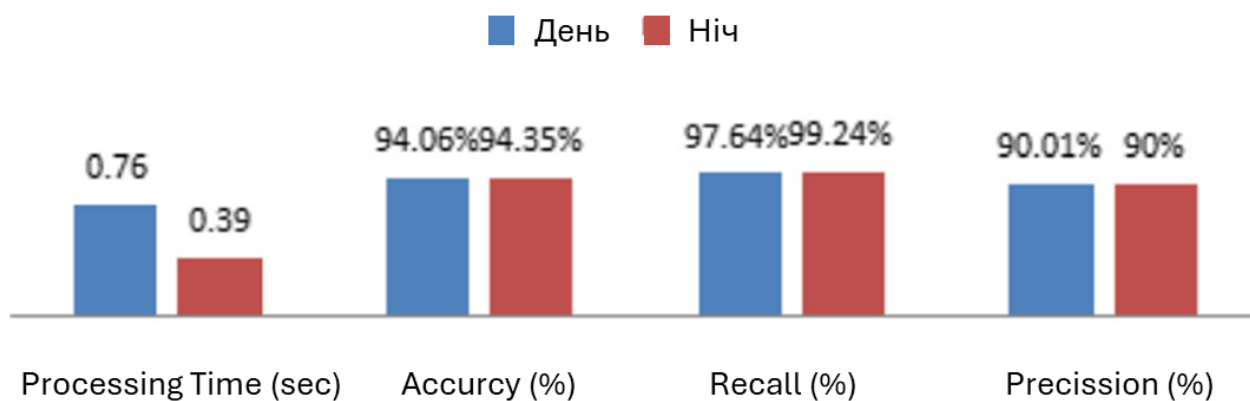


Рисунок 20. Порівняння продуктивності системи в денному і нічному режимах

Автоматичне виявлення транспортних засобів є найважливішим додатком інтелектуальної транспортної системи (ITS).

Впровадження її технологій на дорогах і магістралях стало надзвичайно необхідним у зв'язку зі збільшенням інтенсивності дорожнього руху. Ця система набула величезного значення в області планування і органів безпеки. Основна мета дослідження полягала в тому, щоб провести загальний огляд існуючої роботи в різних режимах, щоб поліпшити результати шляхом усунення недоліків. В роботі запропоновано використання моделі глибокого навчання YOLOv5, відому тим, що дозволяє класифікувати об'єкти за менший час обчислень. В роботі проведено дослідження запропонованого методу за допомогою набору інструментів глибокого навчання на різних відео, знятих у різних умовах дорожнього руху, таких як хмарна погода, яскраве сонце та нічний режим. Описана вище техніка добре працює зі складними сценами, такими як щільний рух, довгі тіні та умови слабого освітлення, такі як хмарна погода та виявлення нічного режиму. Продуктивність автоматичної системи виявлення транспортних засобів оцінюється з точки зору точності, точності, відгуку та часу обробки. В майбутньому можуть з'явитися більш складні і різноманітні набори даних, які доступні публічно і містять різні типи транспортних засобів на зображенні. Таким чином, майбутні дослідження повинні бути спрямовані на пошук всіх можливих підходів, які можуть забезпечити виявлення в режимі реального часу кожного типу транспортного засобу в складних умовах.

Висновки. Дослідження інтелектуальної системи виявлення об'єктів в реальному часі на базі YOLOv5 являє собою значний прогрес в області комп'ютерного зору і штучного інтелекту. YOLOv5, що володіє найсучаснішими можливостями виявлення об'єктів, відкрив численні можливості для додатків в режимі реального часу в різних областях, включаючи безпеку, спостереження, автономні транспортні засоби та багато іншого.

У ході цього дослідження ми вивчили функціональні та нефункціональні вимоги, а також системну архітектуру, необхідні для створення ефективної інтелектуальної системи виявлення об'єктів у режимі реального часу. В роботі підкреслено важливість досягнення високої точності, низької затримки та масштабованості.

Проаналізувавши результати, можна зробити висновок, що інтеграція новітніх технологій глибокого навчання у системи комп'ютерного зору відкриває нові можливості для підвищення ефективності виявлення об'єктів. Також було ідентифіковано, що ключовими параметрами оцінки ефективності таких систем є Recall, Accuracy, Precision та час обробки, що підтверджується виконаними експериментами.

З огляду на отримані дані, ми визначили, що подальші дослідження в області комп'ютерного зору та виявлення об'єктів повинні бути спрямовані на оптимізацію алгоритмів для забезпечення ще більшої швидкості та точності розпізнавання, особливо в складних умовах. Це зокрема важливо для розширення застосування таких систем в різних галузях, від контролю трафіку до розширених систем безпеки.

У підсумку, наше дослідження підкреслює значний потенціал використання алгоритмів комп'ютерного зору та глибокого навчання для вдосконалення систем виявлення об'єктів у відеопотоках, надаючи важливий внесок у розвиток технологій спостереження та аналітики.