

Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «МАГІСТР»

РОЗРОБКА І ДОСЛІДЖЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ
ОБЛІКУ АВТОТРАНСПОРТУ ЧЕРЕЗ КОНТРОЛЬНО-ПРОПУСКНИЙ
ПУНКТ

DEVELOPMENT AND RESEARCH OF AN AUTOMATED SYSTEM FOR
ACCOUNTING FOR VEHICLES AT CHECKPOINTS

спеціальність 122 Комп'ютерні науки

освітня програма «Комп'ютерні науки»

Виконав: здобувач вищої освіти
групи КНм-21
Кутас Олександр Володимирович

(підпис)

Керівник: д.пед.н, с.н.с.
Козубцов Ігор Миколайович

(підпис)

Кваліфікаційну роботу
допущено до захисту
«___» _____ 2025 р.
Гарант освітньої програми:
к.т.н., доцент
Ліщина Валерій Олександрович

(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет *комп'ютерних та інформаційних технологій*

Кафедра *комп'ютерних наук*

Ступінь вищої освіти: *магістр*

Галузь знань: *12 Інформаційні технології*

Спеціальність: *122 Комп'ютерні науки*

Освітня програма: *«Комп'ютерні науки»*

ЗАТВЕРДЖУЮ

Завідувач кафедри

Валерій ЛІЩИНА

«14» травня 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА ДРУГОГО (МАГІСТЕРСЬКОГО) РІВНЯ ВИЩОЇ ОСВІТИ

Кутас Олександр Володимирович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «Розробка і дослідження автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт»

Керівник д.пед.н, к.т.н., с.н.с. Козубцов Ігор Миколайович

затверджені наказом закладу вищої освіти від «14» травня 2025 р. № 255/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи «05» грудня 2025 р.

3. Вихідні дані до роботи: Сервер має відповідати наступним вимогам:

1) операційна система Ubuntu 16.04 64-розрядна; 2) платформа Node.js не нижче 6.0; 3) процесор із тактовою частотою 2 ГГц або швидший; 4) оперативна пам'ять 2 гігабайт (ГБ); 5) 8 Гб вільного місця на жорсткому диску;

Клієнтський додаток має відповідати таким мінімальним системним та апаратним вимогам: 1) операційна система Windows 7/8/10 32 або 64-розрядна; 2) процесор із тактовою частотою 1 ГГц або швидший – 32-розрядний (x86) або 64-розрядний (x64); 3) оперативна пам'ять 4 гігабайт (ГБ) (для 32-розрядної версії) або 16 ГБ (для 64-розрядної версії); 4) 256 Мб вільного місця на жорсткому диску; 5) браузері Google Chrome 50, Mozilla Firefox 48, Edge 40, Opera 42 вище.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити): Аналіз сучасного стану проблеми, існуючих методів і засобів її розв'язання, аналіз і вибір засобів проектування, опис функціонального наповнення об'єкта проектування, розробка й обґрунтування системного наповнення, експериментальне дослідження результативності предмету дослідження.

5. Перелік графічного матеріалу:

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Аналіз сучасного стану автоматизації обліку автотранспорту</i>	<i>Козубцов І. М.</i>		
<i>Методи і підходи до розпізнавання образів</i>	<i>Козубцов І. М.</i>		
<i>Розробка та дослідження автоматизованої системи обліку автотранспорту через контрольні-пропускні пункти</i>	<i>Козубцов І. М.</i>		
<i>Показник запозичень тексту</i>		_____ %	
<i>Інструментальна перевірка</i>	<i>Кошелюк В. А.</i>		
<i>Нормоконтроль</i>	<i>Сачук В. О.</i>		
<i>Гарант ОПП</i>	<i>Ліщина В. О.</i>		

7. Дата видачі завдання «14» травня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Примітка
1	<i>Провести огляд літературних джерел по темі кваліфікаційної роботи</i>	<i>до 30.06.2025 р</i>	
2	<i>Провести аналіз загальної проблеми і вибір напрямків дослідження</i>	<i>до 01.09.2025 р.</i>	
3	<i>Розробити функціональну схему роботи програмного продукту</i>	<i>до 01.10.2025 р</i>	
4	<i>Описати засоби розробки об'єкта проектування</i>	<i>до 15.10.2025 р.</i>	
5	<i>Практична реалізація об'єкта проектування</i>	<i>до 10.11.2025 р.</i>	
6	<i>Провести експериментальне дослідження результативності предмету дослідження</i>	<i>до 25.11.2025 р.</i>	
7	<i>Здача чистового варіанту кваліфікаційної роботи бакалавра на кафедру</i>	<i>до 05.12.2025 р.</i>	

Здобувач вищої освіти _____ Олександр КУТАС

Керівник роботи _____ Ігор КОЗУБЦОВ

АНОТАЦІЯ

Кутас О. В. Розробка і дослідження автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт. Рукопис.

Кваліфікаційна робота магістра ОП «Комп'ютерні науки». Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота магістра складається з вступу, 3 розділів, висновків, списку використаних джерел та додатків.

У кваліфікаційній магістерській роботі було вирішено науково-технічну задачу, спрямовану на розробку та дослідження роботи є автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт. У дослідженні були розглянуті основні типи автоматизованих систем контролю обліку автотранспорту через контрольні пункти. Наукова новизна одержаного результату полягає в удосконаленні процесу автоматизації обліку автотранспорту через контрольно-пропускний пункт. Практичне значення результату полягає у реалізації можливості автоматизованому обліку автотранспорту через контрольно-пропускний пункт.

Ключові слова: застосунок, автоматизація, автотранспорт, контроль, контрольно-пропускний пункт.

ABSTRACT

Alexander Kutas. Development and research of an automated system for accounting for motor vehicles through a checkpoint. Manuscript.

Master's thesis in Computer Science. Lutsk National Technical University. Lutsk, 2025.

The master's thesis consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The master's thesis addresses a scientific and technical problem aimed at developing and researching the operation of an automated system for accounting for motor vehicles through a checkpoint. The study examined the main types of automated systems for controlling motor vehicle accounting through checkpoints. The scientific novelty of the result obtained lies in the improvement of the process of automating motor vehicle accounting through a checkpoint. The practical significance of the result lies in the implementation of the possibility of automated motor vehicle accounting through a checkpoint.

Keywords: application, automation, motor transport, control, checkpoint.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1 АНАЛІЗ СУЧАСНОГО СТАНУ АВТОМАТИЗАЦІЇ ОБЛІКУ АВТОТРАНСПОРТУ.....	11
1.1 Обґрунтування потреби у розробці автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт	11
1.2 Огляд основних компонентів автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт	13
1.3 Автоматизований пункт збору платежів із самообслуговуванням.....	15
1.4 Постановка завдання на розробку автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт	17
РОЗДІЛ 2 МЕТОДИ І ПІДХОДИ ДО РОЗПІЗНАВАННЯ ОБРАЗІВ	20
2.1 Загальна структура способу розпізнавання державного реєстраційного номера на автотранспорті.....	20
2.2 Огляд методів фільтрації зображення і попереднього пошуку.....	21
2.3 Огляд алгоритмів розпізнавання символів	25
2.4 Огляд сучасних рішень розпізнавання образів	28
2.5 Розпізнавання символів за допомогою CNN мережі.....	32
2.6 Розпізнавання об'єктів у зображенні	39
2.7 Проблема перенавчання та способів вирішення.....	41
РОЗДІЛ 3 РОЗРОБКА ТА ДОСЛІДЖЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ АВТОТРАНСПОРТУ ЧЕРЕЗ КОНТРОЛЬНО-ПРОПУСКНИЙ ПУНКТ	43
3.1 Дослідження залежності вибору методу ініціалізації ваг	43
3.2 Дослідження вплив передавальної функції на CNN мережу.....	46
3.3 Дослідження впливу розміру ядра згортки	48
3.4 Розробка архітектури програмного забезпечення автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт	51

3.5	Опис технічних засобів автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт	53
3.6	Опис інтерфейсу адміністратора системи обліку автотранспорту через контрольно-пропускний пункт	57
3.7	Опис інтерфейсу веб-порталу користувача системи обліку автотранспорту через контрольно-пропускний пункт	61
3.8	Опис структури додатку автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт	63
3.9	Алгоритм створення платних доріг.....	66
	ВИСНОВКИ.....	68
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	71
	ДОДАТКИ.....	73

ВСТУП

Актуальність роботи та підстави для її виконання. За останні 20 років технології, пов'язані з комп'ютерним зором стрімко розвинулись та продовжують розвиватися. Також збільшується частка технологій для автоматизації багатьох процесів у різноманітних сферах.

Значна частка такого програмного забезпечення оброблює саме відеоматеріали. Наразі дані технології вже здатні на такі речі як, наприклад, автопілот для автомобіля або визначення діагнозу на основі знімків.

Ми вже звикли до використання даних технологій у повсякденному житті, адже вони стали доступні широкій аудиторії майже з появою перших смартфонів. Взяти, наприклад, технологію розпізнавання обличчя у відеопотоці у реальному часі, яка вже давно доступна у камері звичайного смартфона. автоматизація не оминув процес обліку автотранспорту через контрольно-пропускний пункт.

Автоматизована система обліку автотранспорту через контрольно-пропускний пункт направлена на усунення затримки при проїзді через:

- платні дороги шляхом збору плати за проїзд в електронному вигляді;
- обліку автотранспорту через контрольно-пропускний пункт блок-постів;
- обліку автотранспорту через контрольно-пропускний пункт автопарків в тому числі у секторі безпеки та оборони скорочення обсягу работ черговому парку.

Система аналізує, чи транспортні засоби, які пересуваються, зареєстровані в базі даних. У разі відсутності реєстрації, вона інформує правоохоронні органи про транспортний засіб, що порушив правила. Наприклад, система може автоматично генерувати електронний рахунок для оплати або здійснювати оплату самостійно, якщо електронна картка прив'язана до зареєстрованого профілю.

Автоматизована система обліку автотранспорту через контрольно-

пропускний пункт базується на чотирьох основних компонентах:

- автоматична ідентифікація автотранспорту;
- автоматичну класифікацію автотранспорту;
- обробка транзакцій;
- сповіщення необхідних служб при порушенні проїзду контрольно-пропускний пункт.

Дана система дозволяє отримати багато переваг: власники автотранспортів економлять свій час; через те що машинам не потрібно зупинятися; економія пального; дозволяє отримати інформацію про трафік на дорогах.

Метою кваліфікаційної роботи є розробка та дослідження автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт.

Для досягнення мети сформуємо такі завдання:

- проаналізувати потреби у розробці автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт;
- проаналізувати відомі методи і підходи до розпізнавання образів;
- обґрунтувати архітектуру програмного забезпечення автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт;
- дослідити моделі автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт;
- розробити практичні рекомендації з реалізації автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт;

Об'єктом дослідження є процес автоматизації обліку автотранспорту через контрольно-пропускний пункт.

Предметом дослідження є процеси та методи проектування автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт.

Умови та обмеження. При розробці автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт обмежимося сформованими вихідними вимогами до сервера та програмного продукту

клієнтського додатку.

Можливі сфери застосування. Застосунок може бути використаний для автоматизації обліку автотранспорту через контрольно-пропускний пункт. Також застосунок може бути застосований для автоматизованого обліку автотранспорту через контрольно-пропускний пункт або блок пости в умовах війни при перетині автомобільної техніки з або в напрямку тимчасово не контрольованої території.

Наукова новизна одержаного результату. Удосконалено процес автоматизації обліку автотранспорту через контрольно-пропускний пункт.

Практичне значення результату полягає у реалізації можливості автоматизованому обліку автотранспорту через контрольно-пропускний пункт.

РОЗДІЛ 1

АНАЛІЗ СУЧАСНОГО СТАНУ АВТОМАТИЗАЦІЇ ОБЛІКУ АВТОТРАНСПОРТУ

1.1 Обґрунтування потреби у розробці автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт

Автоматизована система обліку автотранспорту через контрольний пропускний пункт направлена на усунення затримки при проїзді через:

- плату за проїзд в електронному вигляді;
- облік автотранспорту через контрольно-пропускний пункт блок-постів;
- облік автотранспорту через контрольно-пропускний пункт автопарків в тому числі у секторі безпеки та оборони скорочення обсягу робіт черговому парку;
- визначення інформації про технічний стан військової техніки при перетині контрольного пропускного пунктів [1].

Система перевіряє, чи зареєстровані в базі даних транспортні засоби, що проїжджають повз контрольно-пропускний пункт, і якщо ні, повідомляє правоохоронні органи про транспортні засоби, які проїхали повз без реєстрації. Наприклад, система виставляє електронний рахунок-фактуру на оплату або автоматично обробляє платежі після прив'язки смарт-картки до зареєстрованого профілю.

Автоматизована система обліку автотранспорту через контрольно-пропускний пункт це сукупність пристроїв для зняття оплати за проїзд по платній ділянці дороги без зупинки транспортного засобу.

Ця система являє собою опору у формі літери «П», яку встановлюють біля в'їздів і виїздів з платних доріг. Зазвичай транспортні засоби, які проходять через цю опору, повинні мати RFID-мітки або транспондери, що часто розташовуються під лобовим склом автомобіля. У деяких випадках для проїзду через систему водій може бути змушений знизити швидкість, але повна зупинка зазвичай не потрібна.

Контрольні пункти зазвичай оснащені камерами, які фіксують номери транспортних засобів, що проїхали без ідентифікатора або не зареєструвавшись у системі. Надалі ці знімки можуть використовуватися для притягнення порушників до відповідальності.

Оплата за користування дорогою може здійснюватися як заздалегідь (у формі передоплати), так і на основі подальшого виставлення рахунку власнику автомобіля. У другому випадку водій має заздалегідь укласти договір із компанією-оператором дороги, щоб забезпечити автоматичне списання коштів із банківської картки.

Інші функції застосування автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт за даними [2] (рис. 1.1):

– обліку наявності та роботи автотранспорту, в тому числі надання транспортних послуг;

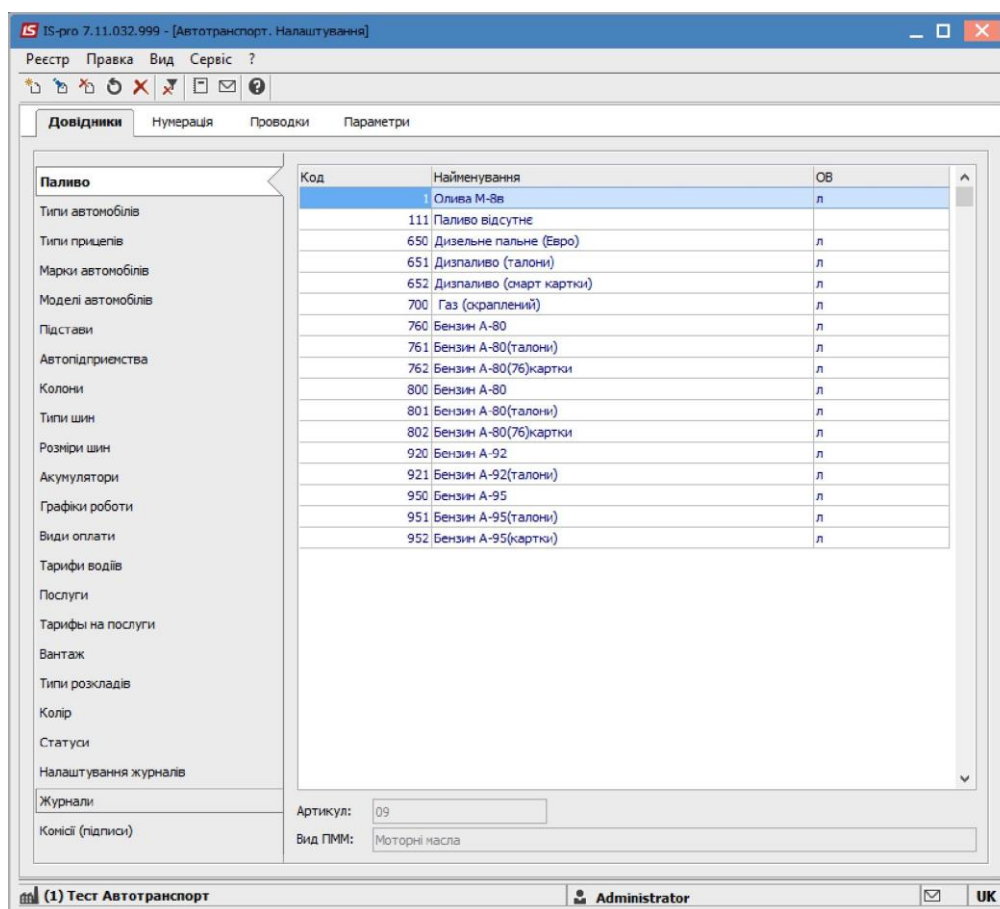


Рисунок 1.1 – Модуль Налаштування підсистеми облік «Автотранспорту» [2]

- обліку палива і паливно-мастильних матеріалів (ПММ);
- облік роботи водіїв;
- обліку пробігу автомобільних шин;
- обліку акумуляторів;
- планування технічного обслуговування автомобілів.

Основним документом відомої системи є Подорожній лист, на підставі якого організовано облік.

Дана система має попри широкі функціональні можливості немає автоматизованої оплати проїзду по платним дорогам.

Крім того, програмне забезпечення має дорогу вартість ліцензії.

1.2 Огляд основних компонентів автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт

Автоматична ідентифікація транспортного засобу. Автоматична ідентифікація транспортного засобу (Automated Vehicle Identification – AVI) – це процес встановлення особистості транспортного засобу, для якого має бути виставлений рахунок [3]. У переважній більшості систем здійснюється реєстрація проїзду транспортних засобів через платні ворота. У таких випадках ключовим завданням стає ідентифікація транспортного засобу в зоні розташування воріт.

У ранніх системах AVI штрих-коди використовувалися для ідентифікації транспортних засобів: вони кріпилися до автомобілів і зчитувалися оптичними пристроями на платіжних пунктах. Однак такі оптичні системи виявилися ненадійними, особливо за несприятливих погодних умов або коли автомобілі були забруднені.

Сучасні системи AVI переважно застосовують радіочастотну ідентифікацію (RFID). Вони функціонують завдяки взаємодії антени на платних воротах із транспондером у транспортному засобі через виділений короткий діапазон зв'язку (DSRC). RFID-мітки демонструють високу точність

зчитування, навіть на великих швидкостях. Головним недоліком таких систем залишається висока вартість встановлення транспондерів на кожен автомобіль.

Щоб уникнути потреби у використанні транспондерів, деякі системи застосовують технологію розпізнавання номерних знаків транспортних засобів. Камери фіксують зображення автомобілів, які проїжджають через контрольні пункти, після чого ці дані обробляються, і номерний знак використовується для ідентифікації транспортного засобу. Це дозволяє користувачам взаємодіяти із системою без попереднього контакту з платіжними агентствами. Раніше головною проблемою таких рішень була висока частота помилок, що спричиняло помилки у виставленні рахунків і збільшення витрат на обробку транзакцій. Для мінімізації цих недоліків запроваджували ручну перевірку, однак це створювало додаткові кадрові витрати. Сьогодні ж завдяки розвитку технологій комп'ютерного зору та систем розпізнавання номерних знаків вдалося значно знизити кількість помилок у їх ідентифікації.

У Німеччині вирішили обрати інший підхід і використовували глобальну систему позиціонування для ідентифікації вантажівок на платних автобанах, відстежуючи їхнє місцезнаходження. Проте впровадження цієї системи виявилось значно дорожчим і тривалішим, ніж попередньо очікувалося.

Автоматизована класифікація транспортних засобів. Автоматизована класифікація транспортних засобів тісно пов'язана з процесом автоматичної ідентифікації [3]. У більшості систем для різних типів транспортних засобів передбачені різні тарифні ставки, що зумовлює потребу у розпізнаванні виду транспорту при проходженні платіжних воріт.

Найпростішим підходом є зберігання інформації про клас транспортного засобу у профілі клієнта та використання даних автоматичної ідентифікації (AVI) для визначення типу авто. Однак, хоча цей спосіб є зручним, він обмежує свободу користувача.

Більш складні системи інтегрують набір датчиків для точнішого аналізу. Наприклад, індуктивні датчики, встановлені в дорожньому покритті, фіксують проміжки між транспортними засобами, надаючи базові дані про їх присутність

та довжину. Датчики тиску допомагають визначати кількість осей машини, яка проїжджає через них, а також ідентифікувати транспорт із подвійними шинами. Крім того, оптичні системи дозволяють детально аналізувати форму автомобіля та відрізнити вантажівки від причепів.

Обробка транзакцій. Обробка транзакцій передбачає управління рахунками клієнтів, виконання платних операцій, зарахування їхніх платежів на рахунок та обробку запитів, що надходять від клієнтів [4]. У деяких системах цей компонент часто називають центром обслуговування клієнтів. Загалом, функції обробки транзакцій мають схожість з банківськими сервісами, через що деякі компанії укладають угоди з банками, передаючи їм цю частину роботи.

Рахунки клієнтів можуть бути сплачені постфактум, коли періодично виставляються рахунки для оплати. Або ж використовувати систему передоплати, де клієнт поповнює баланс на своєму рахунку, і кошти списуються у момент проведення платежів. Система передоплати є більш поширеною, оскільки процес стягнення заборгованості часто вважається економічно недоцільним. Щоб уникнути таких проблем в майбутньому, більшість систем вимагають наявності позитивного балансу на рахунку при створенні облікового запису.

Система порушення правозастосування. Система контролю порушень у сфері правозастосування є ефективним засобом для зменшення кількості неоплачених проїздів, адже автоматизовані процеси без участі людини створюють сприятливі умови для уникнення платежів [4]. З цієї причини, у випадку, коли транспортний засіб не ідентифікується через систему AVI, інформація про його номерний знак надсилається до відповідних правозахисних органів.

1.3 Автоматизований пункт збору платежів із самообслуговуванням

Автоматизований пункт збору платежів з функцією самообслуговування [5]. Ця система дозволяє водіям транспортних засобів здійснювати оплату за проїзд як готівкою, так і банківською карткою. Вартість проїзду

запрограмована заздалегідь, залежно від часу доби, та відображається на комп'ютерній панелі. За потреби вона може бути змінена через систему SCADA у диспетчерському центрі. Після успішного здійснення оплати активується реле, яке опускає бар'єр, забезпечуючи проїзд транспортного засобу. Однак ця система морально застаріла та не передбачає беззупинковий проїзд контрольних пунктів. Це, незважаючи на її автоматизованість без залучення персоналу, створює проблему зростання трафіку.

Автоматизований пункт збору платежів із транспондерами. Автоматизований пункт збору платежів із використанням транспондерів є сучасною системою, яка відрізняється тим, що проїзд через контрольний пункт здійснюється без необхідності зупинки [5]. Транспортні засоби повинні бути обладнані ідентифікаційним пристроєм – транспондером і зареєстровані в системі автоматичного проїзду. На даний момент роль транспондерів виконують RFID-мітки, які розміщуються у вигляді наклейки на лобовому склі або фарах автомобіля. Лазерний детектор здійснює зчитування даних з малим проміжком часу, приблизно 4 мс, передаючи сигнал про наближення транспортного засобу на контролер. Контролер, у свою чергу, активує антену для зчитування інформації та камеру для фіксації зображення автомобіля. Установлені на опорах RFID-зчитувачі забезпечують ідентифікацію менш ніж за 0,02 мс, дозволяючи транспортним засобам проїжджати контрольний пункт без зниження швидкості.

Дані, отримані від контролера та камери, передаються на комутатор із використанням гігабітної лінії Ethernet, після чого надходять на сервер збору даних, який розташовується в придорожньому щиті. Придорожній щит, як правило, не потребує постійного технічного обслуговування. Його функціонування контролює вбудований контролер, який здійснює моніторинг електроживлення пристроїв, температури й вологості, а також стану датчика диму. Крім того, встановлюється система виявлення несанкціонованого проникнення, яка за необхідності передає сигнал тривоги у головний диспетчерський пункт.

Оброблені дані з усіх контрольних пунктів надходять до диспетчерського центру, де відбувається автоматичне списання коштів із електронного рахунку власника транспортного засобу. Вартість проїзду може залежати від довжини маршруту, тарифів конкретної зони та часу доби.

Автоматизований пункт збору платежів. Автоматизований пункт збору платежів характеризується відсутністю потреби у додаткових ідентифікуючих пристроях. Ідентифікація транспортних засобів здійснюється завдяки відеокамерам, що спрощує процес. При наближенні автомобіля лазерний детектор надсилає сигнал до контролера, який активує отримання високоякісного зображення. Це зображення передається на сервер для обробки даних, де виконується розпізнавання номерного знака і його пошук у системі. Якщо в базі даних номер відсутній, інформація про це направляється до правоохоронних органів для встановлення відповідальності порушника.

Серед переваг такої системи варто виділити відсутність необхідності встановлення додаткових пристроїв у транспортний засіб – достатньо лише зареєструватися в автоматизованій системі збору платежів. Недоліком може бути незначне зниження швидкості руху при проходженні через зону розпізнавання для отримання зображення максимальної якості, хоча сучасні технології дозволяють мінімізувати цей фактор. Розвиток систем розпізнавання образів, зокрема автомобільних номерів, досяг такого рівня, що дозволяє забезпечити практично стовідсоткову точність під час обробки даних.

1.4 Постановка завдання на розробку автоматизованої системи обліку автотранспорту через контрольний-пропускний пункт

На підставі розглянутого аналізу в підрозділах 1.1-1.3 встановлено:

– автоматизована система обліку автотранспорту під час проїзду через контрольний пункт спрямована на мінімізацію затримок на платних дорогах шляхом впровадження електронної системи збору коштів. Вона перевіряє, чи транспортні засоби зареєстровані в базі даних, і, у разі відсутності реєстрації,

передає інформацію про порушення до правоохоронних органів. Система автоматично формує електронний рахунок для оплати або здійснює платіж, якщо до зареєстрованого профілю прив'язана електронна картка;

- на сьогодні існує два основні підходи до реалізації автоматизованих пунктів збору платежів: використання транспондерів, які кріпляться на транспортному засобі для ідентифікації власника, та застосування сучасних технологій комп'ютерного зору з розпізнавання номерних знаків автомобілів;

- поряд з перевагами варто зазначити на принциповому недоліку використання засобів дистанційної телекомунікації в навчанні, а саме труднощі виникають при мотивації, координації та контролю результатів навчання;

- сформулюємо мету кваліфікаційної роботи як потреба у розробці та дослідженні автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт.

Для досягнення мети в наступних розділах потрібно вирішити наступні завдання:

- проаналізувати відомі методи і підходи до розпізнавання образів;
- обґрунтувати архітектуру програмного забезпечення автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт;
- дослідити моделі автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт;
- розробити практичні рекомендації з реалізації автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт.

Обмеження вибрані з наявних можливостей здобувача освіти – головного розробника магістерської роботи.

При розробці автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт обмежимося вихідними даними.

Для коректної роботи сервер має відповідати наступним системним та апаратним вимогам (табл. 1.1).

Таблиця 1.1 – Системні та апаратні вимоги до сервера

№ п/п	Параметри	Значення
1	Операційна система	Ubuntu 16.04 64-розрядна
2	Платформа	Node.js не нижче 6.0
3	Процесор із тактовою частотою	2 ГГц або швидший – 64-розрядний (x64)
4	Оперативна пам'ять	16 гігабайт (ГБ)
5	Потреба у вільному місці на жорсткому диску не менше	8 Гб

Для коректної роботи програмного продукту клієнтського додатку персональний комп'ютер має відповідати мінімальним системним та апаратним вимогам (табл. 1.2).

Таблиця 1.2 – Системні та апаратні вимоги до клієнтського додатку

№ п/п	Параметри	Значення
1.	Операційна система	Windows 7/8/10 32 або 64-розрядна
2.	Процесор із тактовою частотою	1 ГГц або швидший – 32-розрядний (x86) або 64-розрядний (x64)
3.	Оперативна пам'ять	4 гігабайт (ГБ) (для 32-розрядної версії) або 16 ГБ (для 64-розрядної версії)
4.	Потреба у вільному місці на жорсткому диску не менше	256 Мб
5.	Браузер	Google Chrome 50, Mozilla Firefox 48, Edge 40, Opera 42 і вище

За результатами постановки завдання в наступних розділах будуть вирішуватися задачі з розробки автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт.

РОЗДІЛ 2

МЕТОДИ І ПІДХОДИ ДО РОЗПІЗНАВАННЯ ОБРАЗІВ

2.1 Загальна структура способу розпізнавання державного реєстраційного номеру на автотранспорті

Загалом алгоритм включає три основні етапи.

Етап 1. Фільтрація зображення та початковий пошук номера – визначення області, де міститься номер.

Етап 2. Нормалізація номера – уточнення меж номера і покращення контрасту.

Етап 3. Розпізнавання тексту – зчитування інформації із нормалізованого зображення.

Це базова структура алгоритму розпізнавання. Якщо номер добре освітлений, чітко розташований лінійно та не забруднений, можна обійтися без перших двох етапів. Окрім того, можливе об'єднання процесів пошуку номера й його нормалізації.

Структурна схема системи розпізнавання номерних знаків зображено на рисунку 2.1.

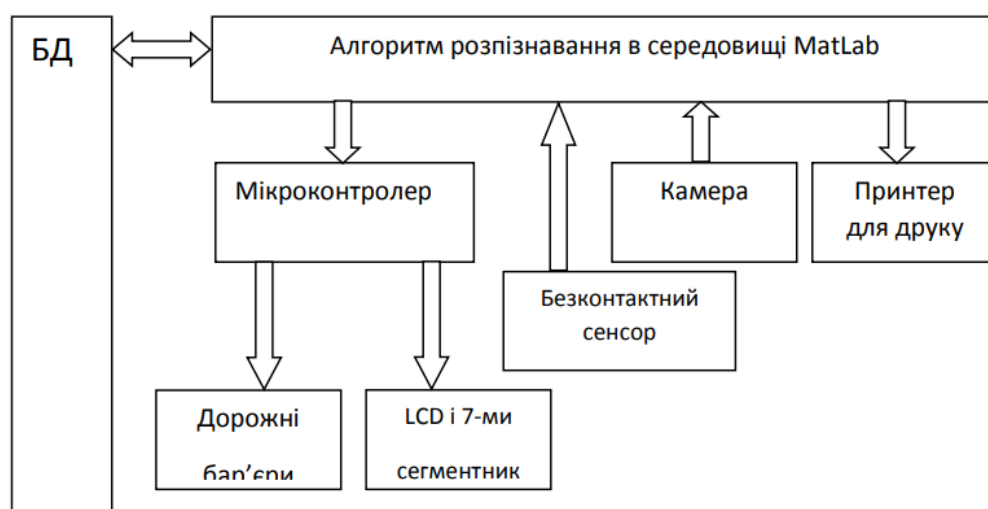


Рисунок 2.1 – Структурна схема системи розпізнавання номерних знаків

Порівняння систем розпізнавання номерних знаків побудована за

результатами аналізу літератури [6-8] подано в таблиці 2.1.

Провівши аналіз існуючих систем розпізнавання номерних знаків, можна зробити висновок, що вони є складними в плані програмної і апаратної реалізації, або потребують спеціалізованих інструментів, наприклад, спеціалізованої камери відео спостереження в системі.

Таблиця 2.1 – Порівняння систем розпізнавання номерних знаків

Характеристики	Системи		
	Система на основі камери Bosch	Vehicle Vision	Система розпізнавання номера автомобіля VNR
Метод розпізнавання	OCR	Нейромережевий	Еталонний метод
Переваги	Розпізнавання в будь-яких умовах освітлення	Створення великої бази даних	Автоматизація процесу видачі штрафу
Недоліки	Спеціалізована і дорого вартісна камера	База даних потребує великої потужності ПК	Потрібна велика база еталонів

2.2 Огляд методів фільтрації зображення і попереднього пошуку

Бінаризація по порогу, вибір області гистограми. Одним із найпростіших перетворень є бінаризація зображення за порогом. Бінаризація передбачає перетворення кольорового або відтінкового зображення на двокольорове, тобто чорно-біле. Ключовим параметром цього процесу виступає поріг t , з яким порівнюється яскравість кожного пікселя. За результатами такого порівняння пікселю присвоюється значення 0 або 1. Методи бінаризації умовно поділяються на дві категорії: глобальні та локальні. У разі глобальної бінаризації порогове значення залишається сталим протягом усього процесу. Натомість у локальній бінаризації зображення поділяється на окремі області, для яких визначається локальний поріг. Основною метою бінаризації є значне скорочення обсягу інформації, з якою доводиться працювати.

Для RGB-зображень та зображень у градаціях сірого порогом виступає певне значення кольору. Є задачі, для яких саме такого перетворення цілком достатньо. Водночас правильний вибір порога суттєво впливає на якість самого процесу бінаризації. Як правило, для цього використовується алгоритм, що

адаптивно підбирає відповідний поріг. Такий алгоритм може базуватися на застосуванні математичного сподівання (середнього значення) або медіани.

Порогова обробка зображень може виконуватися різними методами.

Метод 1. Бінаризація з нижнім порогом – це одна з найпростіших операцій, яка застосовує лише одне порогове значення.

Метод 2. Бінаризація з подвійним обмеженням – використовується для виділення областей, у яких яскравість пікселів змінюється в заданому діапазоні.

Метод 3. Неповна порогова обробка – це перетворення, яке дозволяє отримати зображення без зайвого фону, з усіма деталями вихідного зображення, що робить його більш придатним для подальшого аналізу.

Бінаризація може забезпечити дуже цікаві результати при роботі з гістограмами, зокрема якщо зображення представлено не в RGB, а в HSV-форматі. Аналіз гістограм регіонів є одним із найпоширеніших методів, ілюстрований на рисунку 2.2. Цей підхід базується на припущенні, що частотна характеристика певного регіону відрізняється від характеристик області навколо нього.



Рисунок 2.2 – Приклад аналізу гістограм зображення

На зображенні окреслюються межі, що полягає у виділенні високочастотних просторових компонентів. Створюється проекція зображення на вісь y , а іноді й на вісь x . Максимум отриманої проекції може збігатися з місцем розташування номера.

Однак, цей метод має суттєвий недолік: розміри автотранспортного засобу повинні відповідати розмірам кадру, адже фон може включати написи чи інші деталізовані елементи, що впливають на точність розпізнавання.

Фільтрація контурів. Фільтрація контурів заснована на алгоритмах, що визначають точки цифрового зображення, де спостерігаються різкі зміни яскравості чи інші типи неоднорідностей. Контури відіграють важливу роль у випадках, коли необхідно перейти від роботи зі зображенням до аналізу об'єктів, розміщених на ньому. Особливо це актуально для складних об'єктів, які виразно вирізняються на фоні. У таких випадках виділення контурів може стати єдиним ефективним методом роботи з цими елементами. Для реалізації задачі фільтрації контурів існує цілий спектр різноманітних алгоритмів: Кенні; Собеля; Лапласа; Прюїтта; Робертса.

Найпоширенішим методом для виявлення границь є оператор Кенні – оптимальний алгоритм, який відповідає трьом ключовим критеріям: ефективно виявлення, точна локалізація та однозначний відгук на кожну границю. У його роботі також запроваджено концепцію пригнічення не-максимумів, що передбачає визначення піксельних точок границі як точок, де спостерігається локальний максимум градієнта в напрямі вектору градієнта.

Алгоритм Кані включає п'ять основних етапів обробки зображення.

Етап 1. Згладжування – процес розмиття зображення для зменшення шуму та покращення якості подальшого аналізу.

Етап 2. Визначення градієнтів – межі виділяються в областях, де градієнт яскравості досягає максимальних значень.

Етап 3. Придушення немаксимальних значень – лише локальні максимуми зазначаються як межі, інша інформація відсікається.

Етап 4. Двоетапна порогова фільтрація – потенційні межі класифікуються на основі встановлених порогових значень.

Етап 5. Визначення остаточних меж – завершується шляхом знищення всіх структур, які не зв'язані з достовірно визначеними межами.

Статичний аналіз, класифікатори. Недоліком попередніх методів є те, що

на реальних, забруднених номерах не можна чітко виділити межі. Для таких ситуацій доцільніше використовувати методи, засновані на різних класифікаторах. Одним із цих підходів є каскад Хаара.

Каскад Хаара вважається одним із найбільш ефективних способів швидкого розпізнавання об'єктів. Цей класифікатор базується на ознаках Хаара, які обчислюються шляхом аналізу інтенсивності пікселів у певних областях. Ознака Хаара формується зі суміжних прямокутних ділянок, які накладаються на зображення. Інтенсивності пікселів у цих ділянках підлягають підрахунку: визначається сума пікселів у кожній області, після чого обчислюється різниця між отриманими значеннями. Саме ця різниця виступає як значення ознаки певного розміру і позиції на зображенні.

На етапі виявлення в методі Віоли-Джонса застосовується рухоме вікно сталої величини, яке послідовно сканує зображення. У межах кожної області, захопленої вікном, розраховується ознака Хаара. Визначення наявності або відсутності об'єкта у вікні базується на порівнянні значення ознаки з навчальним пороговим значенням. Проте, оскільки поодинокі ознаки Хаара мають низьку дискримінативну здатність (їхня ефективність дещо перевищує випадковий розподіл), для точного опису об'єкта необхідна значна кількість таких ознак. Щоб вирішити цю проблему, метод Віоли-Джонса організовує ознаки Хаара у каскадний класифікатор.

Найпростішу 2-прямокутну ознаку Хаара можна визначити як різницю між сумами пікселів двох сусідніх областей у прямокутнику, який може змінювати своє положення та масштаб на зображенні. Крім 2-прямокутних ознак, Віола і Джонс ввели також 3-прямокутні та 4-прямокутні ознаки. Кожен тип таких ознак здатний відобразити певні характеристики зображення, наприклад, межі чи текстурні зміни. Зокрема, 2-прямокутна ознака може демонструвати місце переходу між темними і світлими ділянками.

Основною перевагою ознак Хаара є їхня висока швидкість обчислення порівняно з іншими методами. Завдяки використанню інтегрального представлення зображення значення ознак Хаара можна розрахувати за сталий час, що дорівнює

приблизно 60 процесорним інструкціям для 2-прямокутної ознаки.

Процес навчання класифікатора починається із завантаження набору правильних зображень з попередньо виділеними областями. Далі здійснюється перебір примітивів та обчислення значень ознак для створення моделі, здатної ефективно класифікувати об'єкти на основі заданих критеріїв.

2.3 Огляд алгоритмів розпізнавання символів

Задача розпізнавання тексту або окремих символів (optical character recognition, OCR) є складною й водночас традиційною проблемою в області комп'ютерного зору. Загалом, методи розпізнавання тексту поділяються на два основні класи:

- структурні методи, які базуються на аналізі морфології та контуру та застосовуються до бінаризованих зображень;
- растрові методи, що орієнтуються на аналіз самого зображення без попереднього спрощення.

У багатьох випадках доцільно комбінувати структурні та растрові підходи для досягнення найкращих результатів.

K-nearest. Це один із найлегших для розуміння методів розпізнавання символів, який, попри свою простоту, часто може перевершити ненадійні реалізації SVM (метод опорних векторів) або нейронних мереж.

Принцип роботи:

- спочатку створюється база даних, у яку записуються велика кількість зображень реальних символів, заздалегідь правильно класифікованих вручну;
- визначається міра відстані між символами. Якщо зображення бінаризоване, оптимальним варіантом є використання операції XOR;
- під час розпізнавання нового символу поетапно обчислюється відстань між ним і кожним символом у базі даних. Серед k найближчих сусідів можуть бути представники різних класів. Символ зараховується до того класу, представників якого більше серед цих сусідів.

Теоретично, якщо зібрати дуже велику базу прикладів символів, зроблених під різними кутами, при різному освітленні та з різними пошкодженнями, то K-nearest може стати універсальним методом. Однак для цього необхідно забезпечити швидкий розрахунок відстані між зображеннями, що передбачає обов'язкову бінаризацію і використання XOR. Проте це створює проблеми при роботі із забрудненими або пошкодженими символами, адже бінаризація може непередбачувано змінювати їхнє зображення.

Основна перевага цього методу полягає в його простоті та прозорості. Завдяки цьому метод легко налаштовувати й адаптувати для досягнення оптимальних результатів.

Кореляційний. Методи розпізнавання зображень часто базуються на емпіричних підходах, однак доцільно залучати математичний апарат теорії ймовірностей, відшліфований у задачах обробки сигналів у радіолокаційних системах. Наприклад, коли ми працюємо з автомобільними номерами, шрифт їхніх символів нам заздалегідь відомий, а шум фотокамери чи пил на поверхні номеру можна змодельовати як адитивний білий гаусівський шум. Проблема може полягати в невизначеності положення символу чи його нахилу, але ці параметри можна перебирати. У разі, якщо зображення залишається небінаризованим, виникає ще одна змінна – яскравість символу, тобто амплітуда сигналу.

Основне завдання зводиться до розрахунку коваріації вхідного сигналу з гіпотетичним, враховуючи можливі зсуви та повороти. У цьому випадку X позначає вхідний сигнал, Y – гіпотезу, а E – математичне очікування. Якщо необхідно обрати серед кількох можливих символів, для кожного з них створюються окремі гіпотези щодо його нахилу та зміщення. У ситуації, де відомо, що зображення точно містить символ, максимальне значення коваріації дозволяє визначити сам символ, його точне положення та кут нахилу. Проте тут постає проблема схожості зображень різних символів, таких як «р» і «в», або «о» і «с». У цьому випадку найпростішим рішенням є введення вагової матриці коефіцієнтів для кожного символу.

Методи такого типу зазвичай називають «зіставленням із шаблонами» (template matching), що цілком відповідає їх суті: створюються зразки, з якими порівнюється вхідне зображення. Якщо за деякими параметрами виникає невизначеність, можна або перебирати всі можливі варіанти, або застосовувати адаптивні підходи.

Переваги методу:

- результат залишається передбачуваним і достовірним за умови, що шум відповідає обраній моделі;
- метод здатен розпізнавати навіть сильно забруднені, пошкоджені або потерті символи у випадку чітко заданого шрифту.

Недолік: висока обчислювальна складність.

У подальшому в цій роботі буде розглянуто ще один підхід до розпізнавання номерних знаків із використанням згорткових нейронних мереж.

Алгоритми нормалізації. Більшість згаданих вище алгоритмів визначають номер недосконало, тому потребують додаткового уточнення його положення та покращення якості зображення. Наприклад, необхідно зробити поворот і обрізання країв, а також підвищити контрастність.

Одним із важливих кроків є вирівнювання номера в горизонтальній площині. Коли виділяється лише область номера, подальше виділення меж об'єкта працює набагато ефективніше. Це пояснюється тим, що всі довгі горизонтальні прямі, які вдається визначити, переважно є межами самого номера. Для цієї задачі використовується перетворення Хафа – чисельний метод, який застосовують для виявлення об'єктів на зображенні. Він орієнтований на пошук фігур певного класу шляхом застосування механізму голосування.

Ця процедура голосування відбувається в просторі параметрів, де визначаються об'єкти потрібного класу за допомогою знаходження локальних максимумів у так званому накопичувальному просторі (просторі акумуляції), який формується під час обчислення трансформації Хафа. Класичний алгоритм перетворення Хафа спеціалізується на ідентифікації прямих на зображенні. Він

дозволяє швидко виявити дві ключові горизонтальні лінії та обрізати зображення відповідно до їх положення.

2.4 Огляд сучасних рішень розпізнавання образів

Сьогодні існують технології та сервіси, що пропонують готові рішення для розпізнавання зображень. У рамках цієї роботи було проаналізовано функціональність таких сервісів, як Microsoft Azure Cognitive Services, OpenALPR Cloud Stream Services, Google Cloud Vision Services, Amazon Rekognition та OpenCV [9-11], а короткий огляд в таблиці 2.2.

Таблиця 2.2 – Короткий огляд технологій

Технологія	Коротка характеристика
Microsoft Cognitive Services	Microsoft Cognitive Services – це комплекс API, SDK і сервісів, які надають розробникам можливості створювати більш інтелектуальні, інтерактивні та привабливі додатки. Цей інструментарій розширює та вдосконалює портфоліо рішень Microsoft у сфері машинного навчання, забезпечуючи простий спосіб інтеграції інтелектуальних функцій у додатки.
Google Cloud Vision API	Google Cloud Vision API дозволяє розробникам аналізувати зміст зображень, надаючи доступ до потужних моделей машинного навчання через зручний REST API. Цей сервіс здатний швидко класифікувати зображення за тисячами категорій, розпізнавати окремі об'єкти та обличчя, а також ідентифікувати й читати текст, надрукований на зображеннях.
OpenALPR Cloud Stream Services	Сервіс OpenALPR Cloud Stream Services надає функцію автоматичного розпізнавання автомобільних номерних знаків. Його можна використовувати через зовнішній API або інтегрувати з будь-якою IP-камерою для максимальної зручності та ефективності
Amazon Rekognition	Amazon Rekognition – це сервіс, який спрощує інтеграцію функцій аналізу зображень у додатки. Завдяки йому можна визначати об'єкти, сцени, обличчя, а також здійснювати пошук і порівняння облич, або виявляти неприйнятний контент на зображеннях. API Rekognition надає можливість додати до додатка функціонал візуального пошуку та розпізнавання зображень, використовуючи алгоритми глибокого навчання.

Усі технології забезпечують високу точність у визначенні та розпізнаванні образів на зображеннях, пропонуючи готові рішення та зручність у використанні завдяки добре документованому та доступному API. Проте, якщо йдеться про OpenCV, то ця бібліотека надає набір функцій, які

охоплюють приблизно половину задач, що можуть виникати при аналізі зображень. Вона вимагає значно більше зусиль для ефективного використання, але водночас відкриває можливості для створення власних алгоритмів і розширення вже існуючих рішень. Додатково представлена порівняльна таблиця 2.3, яка демонструє можливості цих сервісів.

Таблиця. 2.3 – Порівняльна таблиця можливостей сервісів розпізнавання

Характеристики	Amazon Rekognition	Google Cloud Vision	Microsoft Cognitive Services	OpenCV
Face detection	+	+	+	+
Face recognition (image)	+	-	+	+
Face recognition (video)	-	-	-	+
Emotional Depth (%)	-	-	+	-
Emotions Present (Y/N)	+	+	+	-
Age and Gender	+	-	+	-
Multi-face tracking	+	+	+	+
SDK	-	-	-	+
API	+	+	+	-

З таблиці видно, що кожна технологія пропонує різний набір функцій. Лише OpenCV надає офіційний SDK для роботи з ним. Найширший спектр можливостей забезпечує Microsoft Cognitive Services.

У магістерській роботі для порівняння з власним підходом буде використано OpenALPR, оскільки ця технологія пропонує готове розв'язання для розпізнавання номерних знаків транспортних засобів із високою точністю.

Огляд роботи OpenALPR. OpenALPR функціонує як послідовний ланцюг дій, спрямованих на аналіз зображень з метою отримання номерного знаку транспортного засобу. Вхідними даними є зображення, яке проходить кілька етапів обробки, результатом чого є визначення можливого номеру на ньому.

Основні етапи роботи включають [12].

Етап 1. Розпізнавання – знаходяться потенційні області, які можуть містити номерний знак. Розпізнавання – цей етап виконується один раз для кожного вхідного зображення. Використовується алгоритм LBP (Local Binary Patterns – локальні бінарні шаблони, що часто застосовуються для розпізнавання обличчя), щоб ідентифікувати можливі області номерного знака

(x , y , ширина, висота). Кожна з таких областей надсилається на подальші етапи аналізу. Фаза розпізнавання зазвичай є найбільш ресурсозатратною з точки зору обчислень.

Етап 2. Бінаризація – виділена область перетворюється на чорно-біле зображення, що спрощує подальшу обробку. Бінаризація – цей етап може виконуватися кілька разів, залежно від кількості визначених областей номерного знака. У процесі бінаризації створюється кілька бінарних зображень для кожної ділянки. Це дає змогу підвищити вірогідність розпізнавання всіх символів навіть за складних умов, наприклад, коли зображення занадто темне чи світле. Кожне сформоване бінарне зображення проходить далі на наступні етапи аналізу.

Етап 3. Символьний аналіз – визначається розмір та розташування «згустків» символів у межах номерного знака. Символьний аналіз – на цьому етапі система намагається визначити можливі ділянки з окремими символами в межах області номерного знака. Спершу виявляються всі зв'язані плями в межах цієї ділянки, після чого фільтруються ті, що відповідають ширині та висоті символів і знаходяться в одній лінії з іншими схожими плямами. Якщо відповідних областей не знайдено, ця ділянка виключається з подальшої обробки. Якщо ж потенційні символи виявлено, їхні контури зберігаються для подальшого аналізу.

Етап 4. Знаходження країв – окреслюються межі та форма області номерного знака. Визначення границь – формується список горизонтальних і вертикальних ліній, які належать до області номерного знака. Далі на основі цього списку та висоти символів (визначених під час попереднього етапу) здійснюється пошук можливих меж. Використовуються різні параметричні показники для точного визначення найбільш логічних меж. У разі потреби за базову межу беруть стандартні розміри номерного знака, що дозволяє перевірити правильність результатів.

Етап 5. Усунення спотворень (нормалізація) – перспективне викривлення виправляється, щоб область виглядала рівною, опираючись на стандартні

розміри номерних знаків.

Етап 6. Сегментація символів – окремі символи ізолюються та очищуються для подальшої індивідуальної обробки. Сегментація символів – завданням цієї фази є виокремлення кожного символу, що входить до складу номера. Для цього використовується вертикальна гістограма, яка допомагає знайти проміжки між символами. На цьому етапі вилучаються дрібні плями та артефакти зображення, а також символи, які занадто маленькі або непридатні для аналізу. Така обробка допомагає уникнути помилкового врахування меж пластини як окремих символів, наприклад цифри «1» чи літери «I».

Етап 7. Оптичне розпізнавання символів – кожен ізольований символ аналізується для отримання набору можливих варіантів його значення. Нормалізація – базуючись на визначених межах, область номерного знака перетворюється у стандартний вигляд із правильною орієнтацією та пропорціями. Цей етап усуває перекося або спотворення, щоб отримати максимально точне горизонтальне зображення.

Етап 8. Додаткова обробка – на основі попереднього аналізу формується список можливих комбінацій номерів. Якщо необхідно, виконується порівняння зі зразками регіональних шаблонів за допомогою регулярних виразів. Післяобробка (постобробка) – з усіх отриманих варіантів розпізнаних символів формується список найімовірніших комбінацій. У цьому списку залишаються лише ті символи, які перевищують визначений поріг упевненості. Також допускається «м'який» поріг: символи трохи нижче рівня впевненості можуть залишатися у списку як потенційні варіанти. На цьому етапі також можлива додаткова валідація результатів для конкретного регіону або налаштування параметрів залежно від вимог.

Ця послідовність забезпечує автоматичне й точне розпізнавання транспортних номерних знаків зі зображень.

2.5 Розпізнавання символів за допомогою CNN мережі

Протягом останніх років глибокі нейронні мережі стали основою проривних результатів у різноманітних задачах розпізнавання образів, зокрема у сферах комп'ютерного зору та обробки голосу. Однією з ключових компонент, які сприяли цьому успіху, є згортова нейронна мережа (ЗНМ).

У межах машинного навчання ЗНМ належать до класу глибоких штучних нейронних мереж прямого поширення. Цей вид мереж представляє модифіковану версію багат шарового перцептрона із додатковими шарами елементів, сплановану так, щоб потребувати мінімальної попередньої обробки даних [13]. Вони також відомі своєю інваріантністю щодо зсувів завдяки архітектурі спільних ваг і здатності зберігати просторові характеристики при паралельному перенесенні [14].

Основною перевагою ЗНМ порівняно зі звичайними багат шаровими перцептронами у завданнях обробки зображень є їхня здатність ефективно працювати із великими обсягами вхідних даних, долаючи проблеми, які типово виникають при прямому застосуванні класичних підходів. Наприклад, при застосуванні багат шарового перцептрона для кольорового зображення розміром $32 \times 32 \times 3$, кожен піксель і його канали розглядаються як окремі параметри. У результаті кожен нейрон першого прихованого шару додає близько 3000 нових параметрів. З ростом роздільної здатності картинки кількість параметрів швидко виходить з-під контролю, що робить обчислення непрактичними. Один із популярних підходів до цієї проблеми – зменшення роздільної здатності зображень до рівня, який дозволяє працювати зі стандартними моделями. Проте такий підхід збільшує ризик втрати суттєвих даних та ігнорує просторову структуру, розглядаючи пікселі на великих відстанях як рівнозначні сусіднім.

Рішенням цієї проблеми стала ідея про те, що пікселі, розташовані поруч, відіграють більшу роль у формуванні образу, ніж ті, що віддалені один від одного. Крім цього, якщо ключовий елемент у зображенні є важливим для

класифікації, неважливо, в якій частині зображення він перебуває.

Основні відмінності ЗНМ від багат шарових перцептронів такі:

- тривимірна структура шарів: У шарах ЗНМ нейрони впорядковані у трьох вимірах: ширині, висоті та глибині. Нейрони кожного шару з'єднані лише з невеликою областю попереднього шару, відомою як рецептивне поле. Архітектура будується шляхом комбінування різних типів шарів – як локально-з'єднаних, так і повноз'єднаних;

- локальна зв'язаність: ЗНМ використовують концепцію рецептивних полів для виявлення ознак у межах локальних ділянок вхідного простору. Завдяки цьому фільтри адаптуються до локальних особливостей даних, створюючи сильну відповідь на місцеві входи. Шар за шаром ці фільтри стають більш глобальними: починають із дрібних деталей входу й переходять до більших структур;

- спільні ваги в ЗНМ передбачають, що кожен фільтр повторюється по всьому зоровому полю. Ці повторювані вузли використовують однакову параметризацію, тобто вектор ваг і зсуву, та формують карту ознак. Це забезпечує те, що всі нейрони в конкретному згортковому шарі реагують на одну й ту ж ознаку в межах свого рецептивного поля. Завдяки повторюванню вузлів стає можливим виявлення ознак незалежно від їх розташування в зоровому полі, що гарантує інваріантність до зсувів.

Ці властивості разом суттєво покращують здатність ЗНМ узагальнювати результати на завданнях комп'ютерного зору. Використання спільних ваг значно скорочує кількість параметрів, які потрібно вивчати мережі, тим самим зменшуючи вимоги до пам'яті. Це також дозволяє тренувати більші та потужніші моделі.

Згортковий шар. Розглянемо поняття оператора згортки. Якщо маємо двовимірне зображення I та невелику матрицю K розміром $h \times w$ (яку називають ядром згортки), що графічно кодує певну ознаку, можна обчислити згорнуте зображення $I * K$. Це здійснюється шляхом послідовного накладання ядра на зображення всіма можливими способами та обчисленням суми помножених

відповідних елементів зображення і ядра для кожної позиції (2.1):

$$(I * K) = \sum_{i=1}^h \sum_{j=1}^w K_{ij} \times I_{x+i-1,y+j-1} \quad (2.1)$$

Згортка – це матриця невеликого розміру. Вона використовується в обробці зображень як фільтр для таких цілей, як розмиття, підвищення різкості, виділення контурів тощо. Обробка зображення передбачає обчислення нового значення певного пікселя з урахуванням значень пікселів, що його оточують.

На рисунку 2.3 схематично представлено описану формулу (2.1).

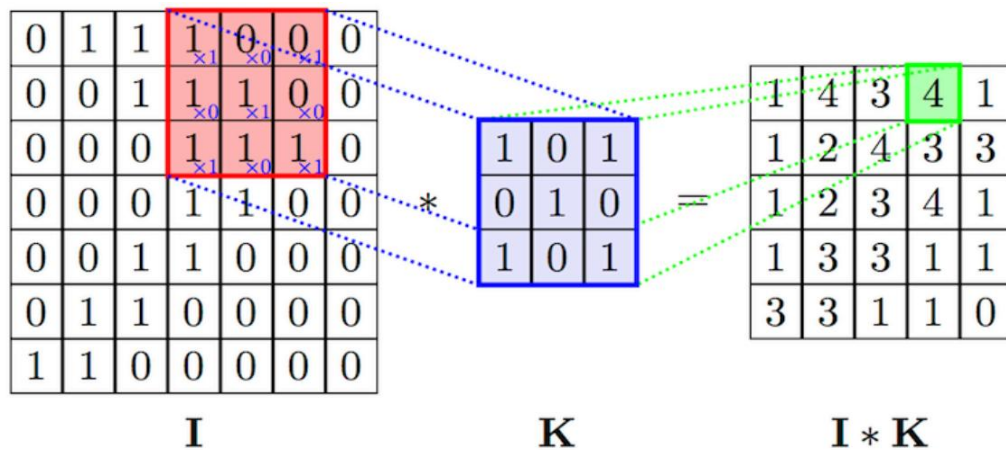


Рисунок 2.3 – Схематичне зображення операції згортки

Згортковий шар є основним елементом у структурі ЗНМ. Його параметри формуються з набору фільтрів (або ядер), що мають невелике рецептивне поле, яке охоплює всю глибину вхідного об'єму. Під час прямого проходу кожен фільтр виконує операцію згортки на ширині та висоті вхідних даних, обчислюючи скалярний добуток між значеннями фільтру та відповідними областями входу. Результатом цієї операції є двовимірна карта активації, що відображає реакцію фільтра на певні ознаки у конкретних просторових локаціях вхідного об'єму. Усі карти активації, створені різними фільтрами, об'єднуються вздовж осі глибини, утворюючи повний об'єм виходу згорткового шару. Кожен елемент цього виходу можна розглядати як результат

роботи нейрона, який аналізує невелику область вхідних даних і використовує ті самі параметри, що й інші нейрони в межах його карти активації.

Просторова організація. Розмір вихідної ємності згорткового шару визначається трьома ключовими гіперпараметрами:

- глибиною;
- кроком;
- нульовим доповненням.

Глибина вихідної ємності визначає кількість нейронів у шарі, які з'єднуються з однією і тією самою областю вхідних даних. Ці нейрони навчаються розпізнавати різні ознаки вхідного сигналу. Наприклад, якщо перший згортковий шар обробляє сире зображення, то нейрони вздовж осі глибини можуть реагувати на різні орієнтовані контури або кольорові структури.

Крок регулює те, як розміщуються стовпчики по ширині та висоті вхідного простору. При кроці, що дорівнює 1, фільтри переміщуються на один піксель за раз, що призводить до значного перекриття рецептивних полів між стовпчиками і збільшення вихідної ємності. Якщо ж крок збільшується до 2 (або навіть до 3 і більше), фільтри пересуваються на кілька пікселів за один раз. У такому разі степінь перекриття зменшується, а просторові розміри вихідної ємності стають меншими.

Додатково часто застосовують нульове доповнення для полів вхідної ємності. Розмір цього доповнення є третім гіперпараметром. Завдяки йому можна контролювати просторовий розмір вихідної ємності. Наприклад, у певних випадках важливо зберегти вихідний розмір таким самим, як і вхідний, що й забезпечується нульовим доповненням.

Агрегувальний шар. До ключових понять ЗНМ відноситься агрегування (pooling), яке представляє собою форму нелінійного зменшення дискретизації. Існує кілька видів функцій для виконання агрегування, серед яких найпоширенішим є максимізаційне агрегування. Цей метод поділяє вхідне зображення на набір прямокутних областей без перекриття і визначає максимум

для кожної з них. Основна ідея полягає в тому, що точне розташування ознаки не є настільки важливим, як її загальне положення відносно інших ознак.

Агрегувальний шар виконує скорочення просторового розміру представлення, що дозволяє зменшити кількість параметрів та обсяг обчислень у мережі й, у такий спосіб, допомагає запобігти перенавчанню. Перенавчання виникає тоді, коли статистична модель замість виявлення реальних закономірностей у даних починає описувати випадкові похибки або шум.

Агрегувальний шар обробляє кожен зріз глибини входу незалежно, зменшуючи його просторові розміри. Найпоширенішим методом є використання шарів із фільтрами розміром 2×2 , які застосовуються з кроком 2. Така схема (рис. 2.4) зменшує дискретизацію кожного окремого зрізу глибини входу вдвічі за шириною і висотою, відкидаючи 75 % збуджень.



Рисунок 2.4 – Максимізаційне агрегування

У цьому випадку кожна операція взяття максимуму проводиться над чотирма значеннями. При цьому розмір за глибиною залишається незмінним.

Окрім максимізаційного агрегування, вузли можуть застосовувати й інші функції, наприклад, усереднювальне агрегування (average pooling) або агрегування на основі L2-норми (L2-norm pooling).

Функції активації. Найбільш популярною функцією активації (передавальною функцією) є ReLU (Rectified Linear Units) – зрізані лінійні вузли. Це ненасичувальна передавальна функція (2.2):

$$\sigma(x) = \max(0, x), \quad (2.2)$$

де x – вхідне значення нейрона.

Вона посилює нелінійні властивості функції ухвалення рішення і мережі в цілому, не зачіпаючи рецептивних полів згорткового шару [3]. Перевагами є: швидкість обчислення, інваріантність відносно масштабування.

Для посилення нелінійності застосовуються й інші функції, наприклад, насичувальні.

Гіперболічний тангенс (2.3):

$$\sigma(x) = \tanh(x). \quad (2.3)$$

Сигмоїдна функція (2.4):

$$\sigma(x) = (1 + e^{-x})^{-1}. \quad (2.4)$$

Графік передавальних функцій: тотожна функція (Identity), сигмоїдальна функція (Logistic), гіперболічний тангенс (Tanh), функція зрізаних лінійних вузлів (ReLU) – при $z \in [-2, 2]$ (рис. 2.5). Але здебільшого зараз надають перевагу зрізаному лінійному вузлові, оскільки він тренує нейронну мережу в декілька разів швидше без значної розплати точністю узагальнення.

Повноз'єднаний шар. У фіналі роботи нейронної мережі, після застосування кількох згорткових і агрегувальних шарів максимізації, високорівневі обчислення виконуються за допомогою повноз'єднаних шарів. У цих шарах кожен нейрон підключений до всіх вихідних сигналів попереднього шару, як і в класичних нейронних мережах. Обчислення їх активацій здійснюється шляхом матричного множення, доповненого додаванням значення зсуву (біаса).

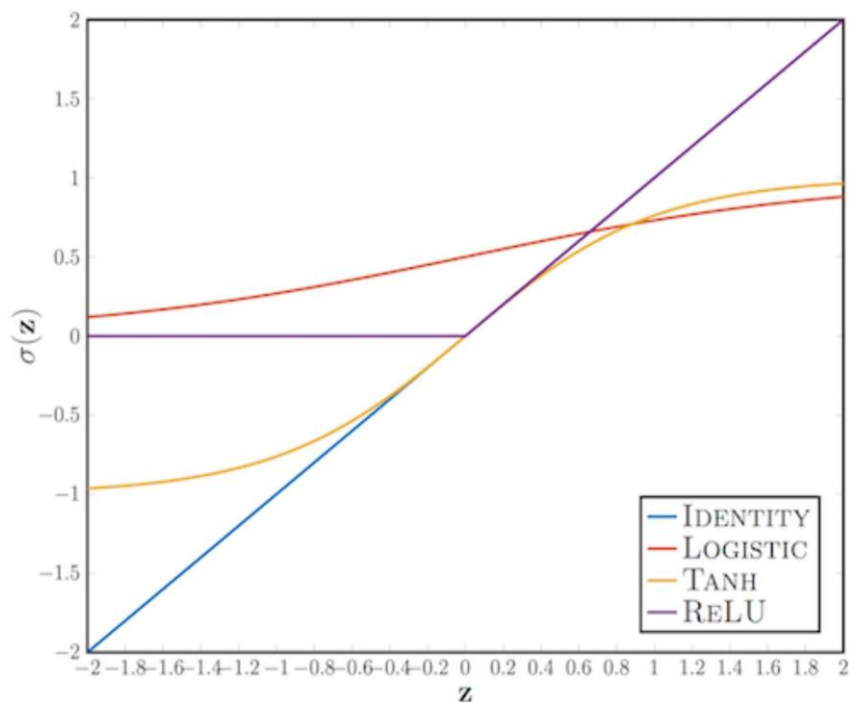


Рисунок 2.5 – Графік передавальних функцій

Повноз'єднаний шар приймає вхідні дані й генерує N -вимірний вектор, де N – це кількість класів для вибору програми. Наприклад, якщо створюють систему розпізнавання цифр, значення $N=10$. Кожен елемент цього N -вимірного вектора описує ймовірність належності до певного класу.

Шар втрат. Шар втрат визначає, як модель під час навчання штрафувє відхилення між передбаченими значеннями і справжніми мітками. Зазвичай цей шар розташований наприкінці архітектури. Для різних типів задач в ньому можуть використовуватись різні функції втрат. Нормовані експоненційні втрати (softmax) використовуються для задач класифікації, коли потрібно передбачити один єдиний клас із K взаємовиключних категорій. Сигмоїдні перехресно-ентропійні втрати застосовуються, коли необхідно передбачити K незалежних ймовірностей у діапазоні від 0 до 1. Евклідові втрати підходять для регресійних задач, де цільові значення є дійсними числами.

Побудова стандартної CNN мережі з подальшим дослідженням та покращенням гіперпараметрів мережі.

Мережа складає з двох згорткових шарів, за кожним із яких розташовуватиметься агрегувальний шар. У кінцевій частині мережі розміщені

два повнозв'язні шари: перший із 512 нейронів, а другий із 10 нейронами, що відповідає кількості класів для розпізнавання. Для запобігання перенавчанню буде використано метод dropout. Регуляризацію застосуємо двічі: після пари згорткових та агрегувальних шарів, а також після передостаннього повнозв'язного шару.

Як функції активації для агрегувальних та повнозв'язних шарів буде використано ReLU (зрізана лінійна функція). Для останнього шару застосуємо softmax, щоб конвертувати вектор числових значень у вектор імовірностей.

Обрані початкові параметри моделі.

Для першого згорткового шару: ядро розміром 3x3, 32 фільтри, крок 1.

Для агрегувальних шарів: ядро розміром 2x2.

Для другого згорткового шару: ядро розміром 3x3, 64 фільтри, крок 1.

Для повнозв'язних шарів: перший має 512 нейронів, а другий – 10.

Для регуляризації dropout: на першому етапі відкидається 25 % параметрів, на другому – 50 %.

Модель була реалізована з використанням бібліотеки TensorFlow (ліст.А.1 у додатку А).

2.6 Розпізнавання об'єктів у зображенні

Поняття виявлення та розпізнавання об'єктів мають багато спільного, але водночас між ними є суттєві відмінності. Виявлення об'єктів на зображенні передбачає визначення їхньої наявності та місця розташування на цифровому зображенні. Розпізнавання об'єктів, своєю чергою, зосереджується на класифікації конкретних об'єктів, які вже виявлено на зображенні.

Людина з легкістю відрізняє та класифікує об'єкти, проте комп'ютери стикаються із певними викликами у цьому процесі. Для виконання таких завдань зазвичай використовують глибинні нейронні мережі. Ці моделі навчаються на великих наборах даних і демонструють високу точність у розпізнаванні об'єктів (рис. 2.6).

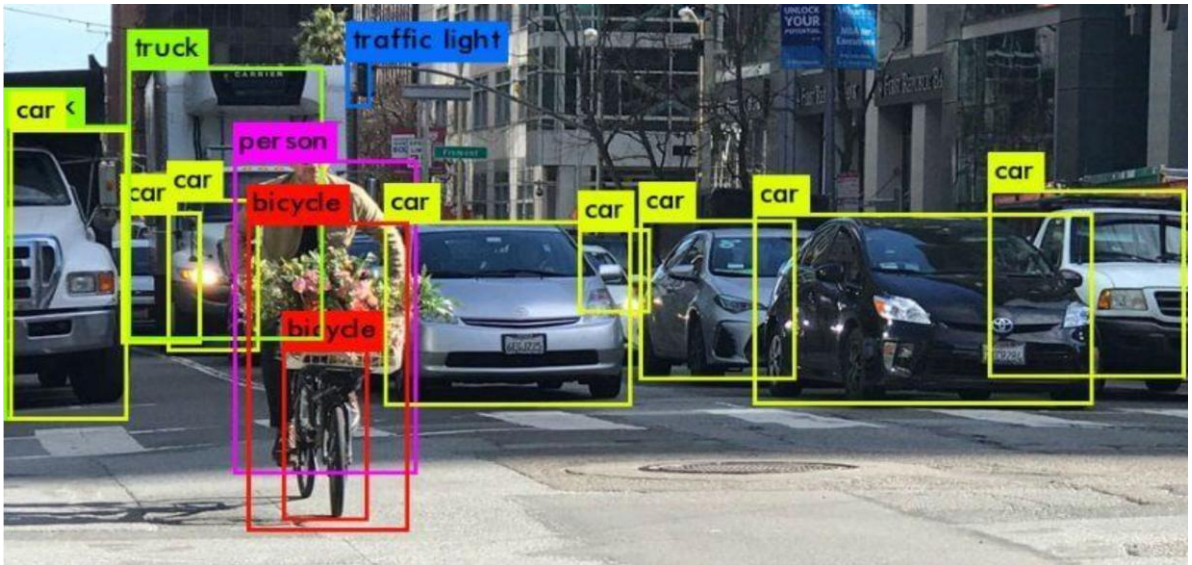


Рисунок 2.6 – Приклад розпізнавання об'єктів

Зазвичай розпізнавання зображень передбачає створення потужних глибоких нейронних мереж, що аналізують кожен піксель зображення. Навчання таких нейронних мереж може здійснюватися із застосуванням традиційних підходів: із вчителем, без вчителя, напівнавчання тощо.

Однак існують і альтернативні методи розпізнавання об'єктів у зображеннях, які не потребують використання нейромереж. Зокрема, метод шаблонів порівнює зображення із заданим набором шаблонів, тоді як метод геометричних ознак виділяє на зображенні базові характеристики: розмір, форму, кольори тощо. Спираючись на ці параметри, об'єкт класифікується.

Етапи обробки зображення для розпізнавання кольору на прикладі методу `isRed()` показано на рисунку 2.7.



Рисунок 2.7 – Етапи обробки зображення при розпізнаванні кольору у методі `isRed()`

Додаток здатний розпізнавати сім кольорів: чорний, білий, сірий, червоний, синій, зелений і жовтий. Для кожного кольору існує відповідний метод, який визначає площу найбільшої області цього кольору в зображенні.

2.7 Проблема перенавчання та спосіб вирішення

Для тренування моделей часто використовують базу даних Modified National Institute of Standards and Technology (MNIST) – обширну колекцію зразків рукописних цифр [15]. Ця база була створена як стандарт для калібрування та порівняння методів розпізнавання зображень за допомогою алгоритмів машинного навчання, особливо на основі штучних нейронних мереж. База включає 60 тисяч зображень для навчання та 10 тисяч зображень для тестування. Кожне зображення представлено у форматі 28 x 28 пікселів.

Перенавчання відбувається, коли модель є надміру складною, з надто великою кількістю параметрів у порівнянні з обсягом наявних даних. Така модель демонструє низьку передбачувальну ефективність, оскільки надто чутливо реагує на несуттєві відхилення в тренувальних даних. Для подолання цієї проблеми застосовуються методи регуляризації. Вони передбачають введення додаткової інформації з метою вирішення некоректно сформульованої задачі або запобігання перенавчанню.

Оскільки повноз'єднані шари містять найбільшу кількість параметрів, вони найбільш схильні до перенавчання. Одним із методів його зменшення є виключення (dropout). У цьому процесі на кожному етапі тренування певні вузли випадково «вимикаються» з мережі з імовірністю $1-W$, тоді як інші залишаються активними з імовірністю W . У результаті формується зменшена версія мережі, в якій також видаляються вхідні й вихідні зв'язки неактивних вузлів.

На наступному етапі тренування проводиться вже на зменшеній мережі. Після цього всі вимкнені вузли повертаються до системи разом зі своїми початковими вагами.

Інший простий спосіб запобігти перенавчанню – це скоротити кількість

параметрів. Це можна зробити, зменшивши кількість прихованих вузлів у кожному шарі або обмеживши глибину мережі. У згорткових мережах на кількість параметрів також впливає розмір фільтрів. Зменшення кількості параметрів безпосередньо скорочує передбачувальну складність моделі, що обмежує її здатність надмірно підлаштовуватися до тренувальних даних і, відповідно, знижує ризик перенавчання.

За результатами виконання другого розділу магістерській роботі резюмуємо:

- проведено аналіз сучасних сервісів для розпізнавання образів дозволив встановити особливості. Для порівняння з розробленим підходом використано платформу OpenALPR, яка є готовим рішенням для розпізнавання номерних знаків транспортних засобів із високою точністю. Цей сервіс обрано через його зручність у використанні та високу ефективність. Подано детальний розгляд принципу роботи сервісу з розпізнавання державного реєстраційного номеру на автотранспорті;

- обґрунтовано загальну структуру алгоритму розпізнавання символів, детально описано основні етапи роботи алгоритму, такі як методи фільтрації зображень, алгоритми нормалізації та розгляд алгоритмів розпізнавання.

РОЗДІЛ 3

РОЗРОБКА ТА ДОСЛІДЖЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ АВТОТРАНСПОРТУ ЧЕРЕЗ КОНТРОЛЬНО-ПРОПУСКНИЙ ПУНКТ

3.1 Дослідження залежності вибору методу ініціалізації ваг

Одним із ключових етапів побудови моделі є ініціалізація нейронної мережі, що передбачає вибір початкових значень ваг для шарів. Грамотно обраний спосіб ініціалізації ваг може суттєво вплинути на ефективність навчання моделі, оскільки він встановлює стартові параметри, враховуючи специфіку функції втрат.

Метод ініціалізації Зав'єра (Xavier Glorot) базується на ідеї забезпечення легкого проходження сигналу крізь шари як у прямому напрямку, так і під час зворотного поширення помилки [16]. Цей підхід ідеально підходить для лінійної функції активації, а також добре працює для сигмоїдної функції, оскільки в області, де функція ненасичена, вона має лінійний характер. Для обчислення ваг цей метод використовує ймовірнісні розподіли (рівномірний або нормальний) із дисперсією, що визначається формулою (3.1):

$$\text{Var}(W) = \frac{2}{n_{in} + n_{out}}, \quad (3.1)$$

де n_{in} і n_{out} – кількість нейронів в попередньому та наступних шарах відповідно того як в роботі.

Метод ініціалізації Хе (Kaiming He) є модифікацією методу Зав'єра, спеціально адаптованою для активаційної функції ReLU [17]. Ця модифікація враховує, що функція ReLU повертає нуль для половини визначеної області. У такій ситуації (3.2):

$$\text{Var}(W) = \frac{2}{n_{in}}, \quad (3.2)$$

Щоб визначити дисперсію для ініціалізації Зав'єра, слід проаналізувати, як поводить дисперсія вихідних значень лінійного нейрона (без урахування зсуву). При цьому передбачається, що вагові коефіцієнти та вхідні значення не корелюють між собою та мають нульове математичне сподівання (3.3):

$$\text{Var}(\sum_{i=1}^{n_{in}} w_i x_i) = \sum_{i=1}^{n_{in}} \text{Var}(w_i x_i) = \sum_{i=1}^{n_{in}} \text{Var}(W) \text{Var}(X) = n_{in} \text{Var}(W) \text{Var}(X) \quad (3.3)$$

Для того щоб зберегти дисперсію вхідних даних після проходження через шар, необхідно дотримуватися умови (3.4):

$$\text{Var}(W) = \frac{1}{n_{in}}. \quad (3.4)$$

Можна застосувати цей же аргумент при зворотньому поширенні помилки, щоб отримати значення (3.5):

$$\text{Var}(W) = \frac{1}{n_{out}}. \quad (3.5)$$

Так як не можна задовільнити обидві потреби, вибираємо дисперсію вагів як середнє за формулою 3.1.

Проведемо дослідження впливу цих обох методів на нашу модель. За результатами дослідження побудовано для наочності два графіки (рис. 3.1). Варто зазначити, що 15 % даних були виділені для моніторингу й оцінки перенавчання моделі протягом процесу тренування мережі.

Під час використання методу Зав'єра модель досягла точності 97.4 %, що є досить високим результатом. Коливання точності пояснюються застосуванням методу регуляризації, який відключає частини нейронів для запобігання перенавчання.

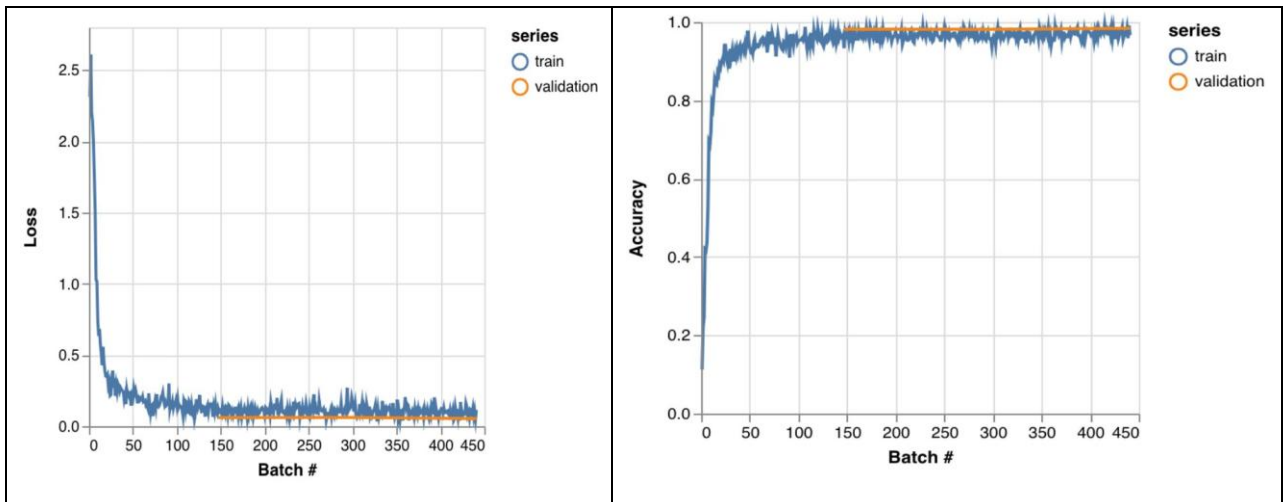


Рисунок 3.1 – Графіки функції втрат та точності навчання CNN моделі з ініціалізацією вагів за допомогою методу Зав'єра

На рисунку 3.2 представлені графіки, отримані при використанні методу ініціалізації He. Цей метод демонструє менші коливання на початкових етапах, однак його точність є помітно нижчою – остаточний показник 96.5 %. З графіку видно, що на останніх ітераціях метод характеризується більшою варіативністю у визначенні точності. Водночас, цей метод проявляє себе значно краще при вирішенні задач оптимізації, зокрема в мінімізації функції втрат.

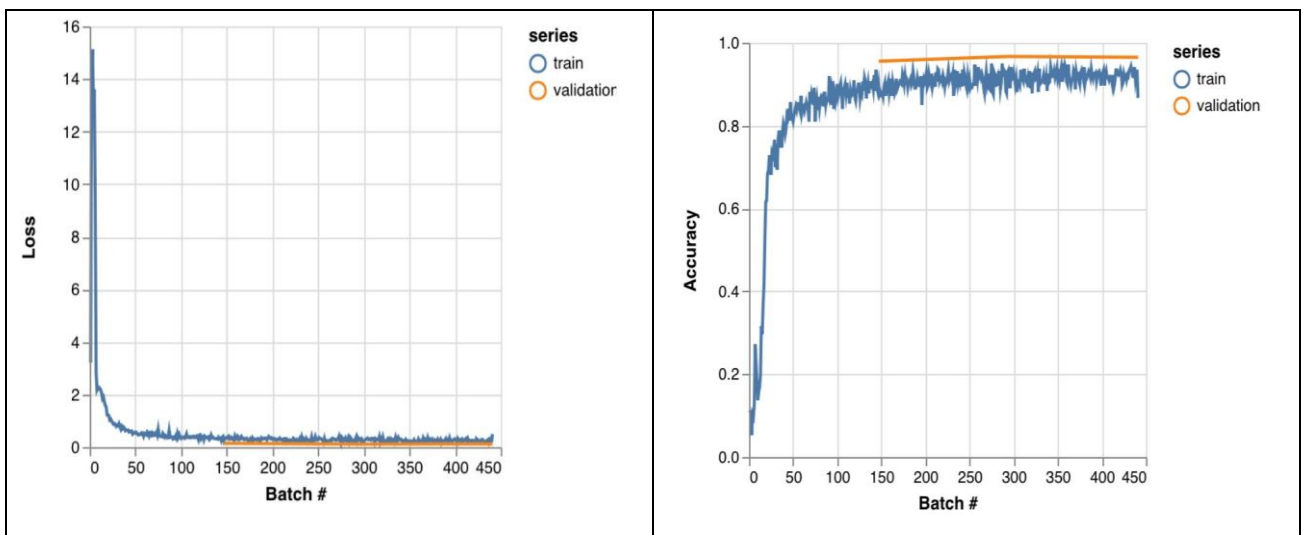


Рисунок 3.2 – Графіки функції втрат та точності навчання CNN моделі з ініціалізацією вагів за допомогою методу He

Слід також звернути увагу на параметр часу навчання мережі, оскільки при використанні реальних даних і збільшенні кількості ітерацій цей процес може тривати дні. Це робить час навчання важливим показником продуктивності в дослідженнях. У рамках проведеного експерименту час навчання за методом Не становив 272.267 секунд, а за методом Зав'єра – 288.489 секунд. Під час тренування було виконано лише три ітерації, проте різниця в часі склала 5.95 %. Така диспропорція може мати відчутний вплив за умов значно більшої кількості ітерацій і великого обсягу тренувальних даних.

Отримані результати дослідження свідчать про те, що метод ініціалізації вагів Не забезпечує швидше навчання CNN-мереж порівняно з популярним методом Зав'єра, демонструючи різницю приблизно у 6 %. Крім того, цей метод впливає на ефективність мінімізації функції втрат. Менша точність результатів може бути пов'язана з обмеженою кількістю ітерацій, виконаних у ході дослідження, і, ймовірно, буде менш помітною при збільшенні їх кількості.

3.2 Дослідження вплив передавальної функції на CNN мережу

Дослідимо вплив вибору функції активації на роботу CNN. Для аналізу обрано такі функції активації: сигмоїдна функція, гіперболічний тангенс та функція відсічених лінійних вузлів (ReLU). На рисунку 3.3 наведено графіки процесу навчання моделі з використанням сигмоїдної функції активації у згорткових шарах.

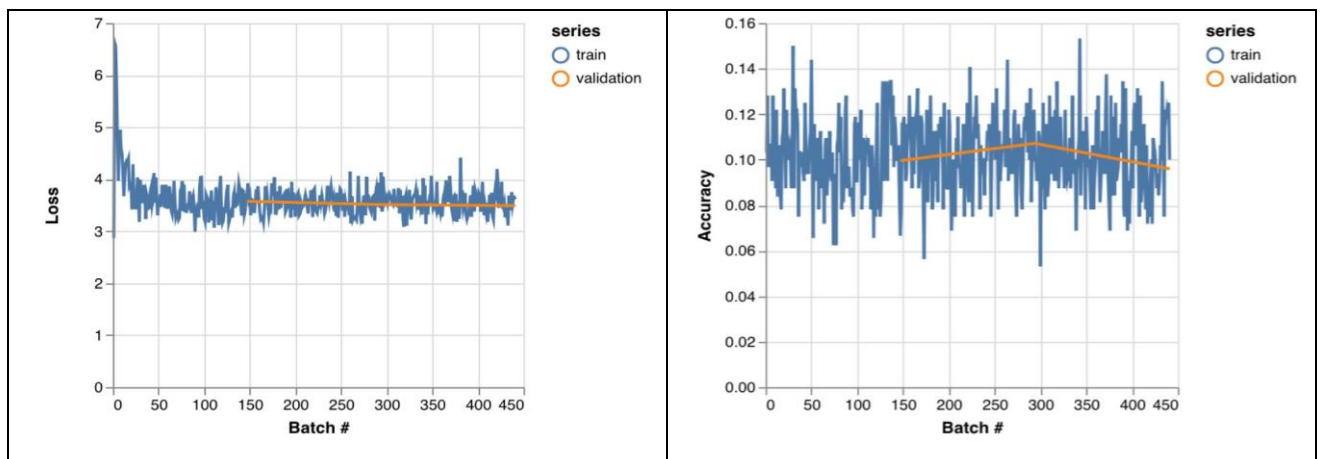


Рисунок 3.3 – Графіки функцій втрат та точності навчання CNN мережі з сигмоїдною функцією активації

З аналізу графіків видно, що модель стикається з проблемою зникнення градієнта. Це зумовлено тим, що сигмоїдна функція має градієнт в діапазоні $[0, 1/4]$, а під час зворотного поширення градієнти обчислюються за допомогою ланцюгового правила. Як наслідок, відбувається перемноження n малих значень при розрахунку градієнтів для «передніх» шарів у мережі з n шарами. Це призводить до експоненційного зменшення градієнта (або сигналу похибки) залежно від кількості шарів, через що передні шари навчаються вкрай повільно.

З аналізу графіків можна зробити висновок, що завдяки тому, що функція має градієнти в інтервалі $[0,1]$, ймовірність виникнення проблеми зникнення градієнта є меншою. На початку спостерігаються значні коливання, а загальний час навчання становив 210.669 с. При цьому досягнута точність складає 94.2 %, а мінімальне значення функції втрат становить 0.316.

На рисунку 3.4 представлено графіки процесу навчання мережі з використанням функції активації гіперболічний тангенс.

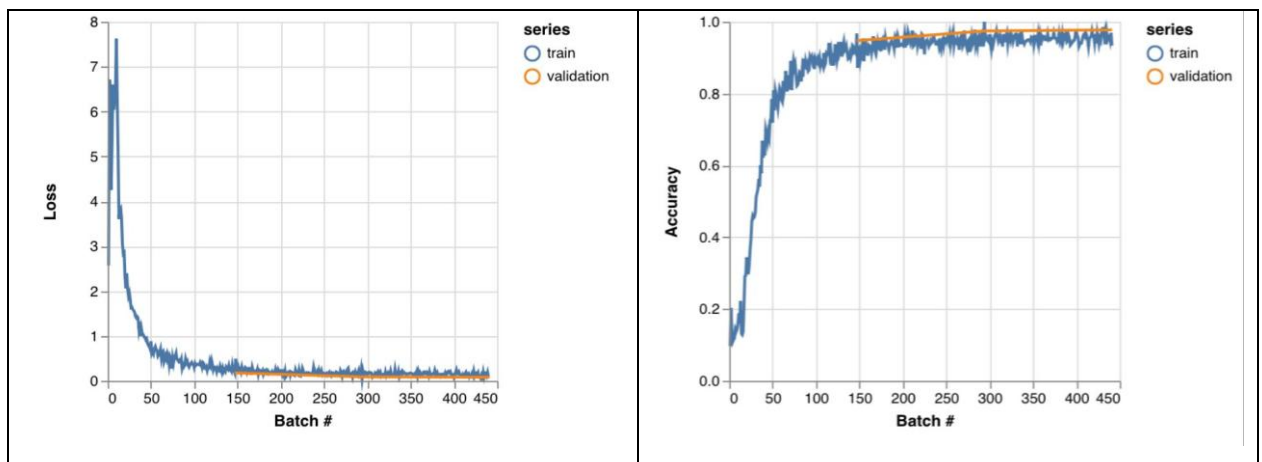


Рисунок 3.4 – Графіки функцій втрат та точності навчання CNN мережі з функцією активації гіперболічний тангенс

У свою чергу, графіки на рисунку 3.5 демонструють, що при використанні функції зрізаних лінійних вузлів покращується ефективність процесу навчання: точність досягає 97.9 %, а значення функції втрат знижується до 0.073. Однак сам процес навчання тривав значно довше – 259.368 с.

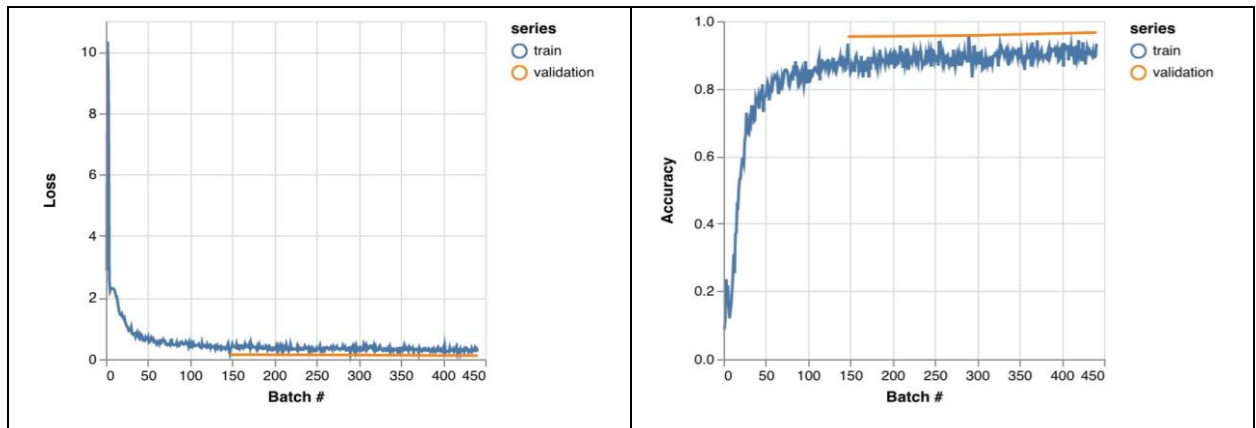


Рисунок 3.5 – Графіки функцій втрат та точності навчання CNN мережі з функцією активації зрізаних лінійних вузлів (ReLU)

З дослідження можна впевнено зробити висновок, що передавальна функція зрізаних лінійних вузлів є значно більш ефективною для згорткових нейронних мереж. Вона забезпечує помітно кращі результати точності, хоч і процес навчання займає більше часу порівняно з гіперболічним тангенсом.

3.3 Дослідження впливу розміру ядра згортки

Проаналізуємо розміри ядра згортки, що використовуються в згорткових шарах нейронної мережі. Для дослідження обрано ядра розмірів 3x3, 5x5 та 9x9, оскільки при використанні більших розмірів існує ризик втрати важливих деталей зображення. Оцінимо результати навчання для трьох різновидів матриць (табл. 3.1).

Таблиця 3.1 Вихідні результати навчання

Розмір матриці/ Характеристики	Точність	Функція втрат	Час (с)
3x3	97.9%	0.073	259.368
5x5	97.5%	0.076	239.210
9x9	97.0%	0.109	193.129

Зі збільшенням розміру матриці спостерігається значне скорочення часу навчання. Для матриць розміром 3x3 та 9x9 ця різниця становить приблизно 66 с. Водночас помітне зниження точності моделі на 0,9%. Мінімальне

значення функції втрат змінилося незначно.

3.3. Дослідження метода для покращення навчання мережі

Ще один спосіб покращення процесу навчання нейронної мережі – використання методу *batch normalization* [18]. Цей підхід був розроблений для вирішення проблеми внутрішнього коваріантного зсуву. Проблема виникає через постійну зміну розподілу вихідних значень функцій активації, що уповільнює навчання, оскільки кожен шар має адаптуватися до нових розподілів під час кожної ітерації тренування. *Batch normalization* нормалізує вхідні дані для кожного шару, допомагаючи усунути цю проблему. Окрім того, метод суттєво прискорює навчання параметрів мережі.

Варто зауважити, що параметри ρ та Γ навчаються одночасно з іншими параметрами мережі в рамках загального процесу тренування. Додамо два шари *batch normalization* до архітектури мережі та проаналізуємо їхній вплив. Для цього буде використано модель із попереднього експерименту з ядром згортки розміром 9×9 (рис. 3.6).

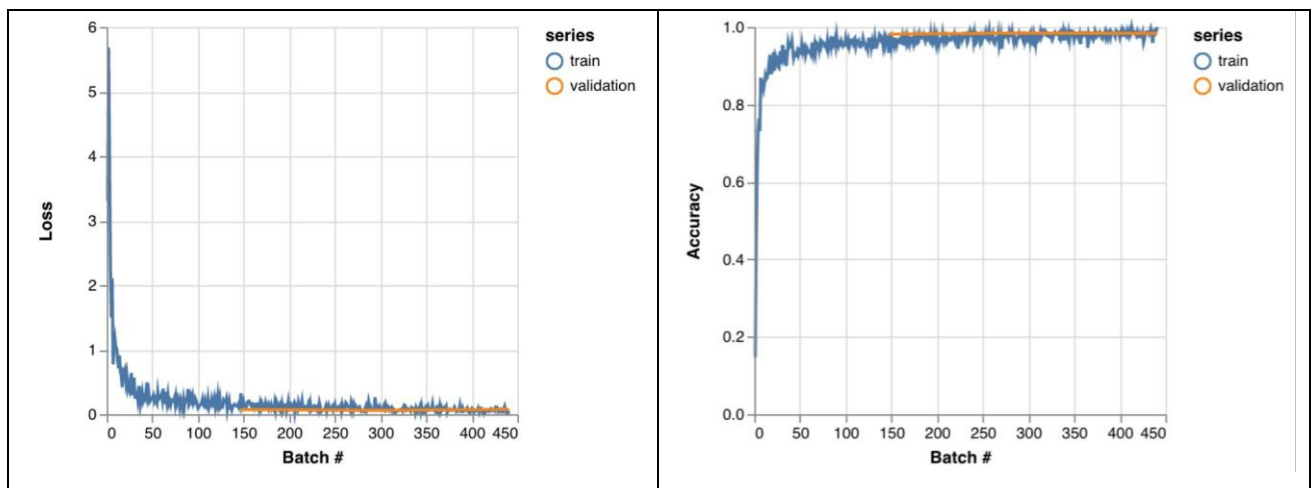


Рисунок 3.6 – Графіки функцій втрат та точності навчання CNN мережі

З графіка видно, що запропонований метод дозволяє значно швидше досягати оптимальних параметрів. Коливання на початкових етапах навчання

практично відсутні. Завдяки використанню batch normalization вдалося підвищити точність моделі до 98.6% і знизити значення функції втрат до 0.057. При цьому тривалість навчального процесу зросла до 303 с.

При цьому автори методу зазначають, що застосування batch normalization може усунути потребу у використанні методів регуляризації. Тому дослідимо процес навчання мережі з використанням batch normalization без регуляризації (рис. 3.7).

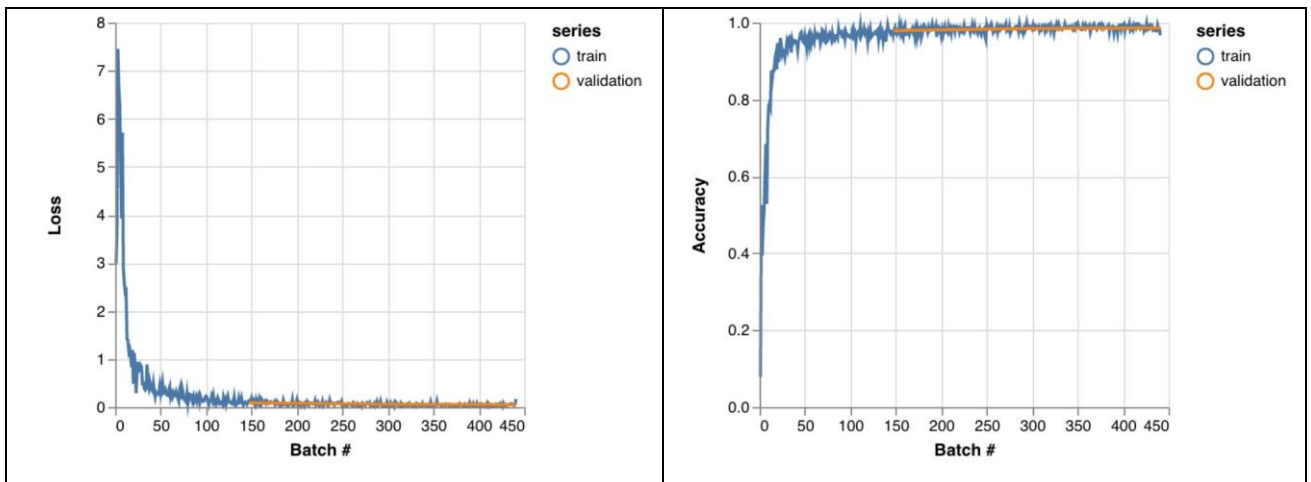


Рисунок 3.7 – Графіки функцій втрат та точності навчання CNN мережі з методом batch normalization без регуляризації

З графіків видно, що використання batch normalization дійсно дозволяє обходитися без методів регуляризації, не викликаючи значних змін у характеристиках процесу навчання. Точність досягає 98.6 %, а функція втрат становить 0.054. Помітно лише незначне скорочення часу навчання, яке склало 294 с.

Таким чином, за результатами проведеного дослідження CNN-мережі з використанням основних методів оптимізації навчального процесу встановлено:

- що від вибору розміру ядра згортки суттєво впливає на скорочення часу навчання, тоді як початкова ініціалізація параметрів відіграє дещо меншу роль;

- покращення точності моделі можливе завдяки застосуванню методів регуляризації, batch normalization або зміні розміру ядра згортки;
- серед усіх функцій активації найкращі результати продемонструвала функція ReLU;
- що за розробленого рішення досягнуто таких характеристик: точність – 98.6 %, значення функції втрат – 0.054, час навчання – 294 с.

3.4 Розробка архітектури програмного забезпечення автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт

В магістерській роботі використовується триярусна веб-додатку. Дана модель передбачає наявність трьох компонентів (рис. 3.8): клієнта; серверної частини; сервер бази даних.

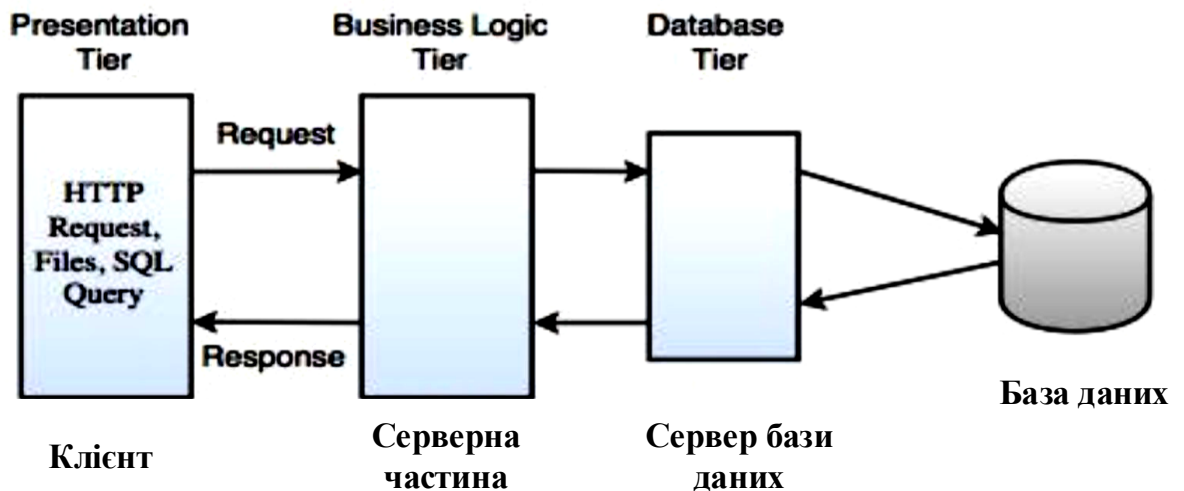


Рисунок 3.8 – Триярусна архітектура веб-додатку [18]

Користувач – це графічний інтерфейс, який надається кінцевому споживачу. На цьому рівні не повинно бути жодних прямих зв’язків із базою даних. На цей рівень виносять лише найпростішу бізнес-логіку: інтерфейс авторизації, алгоритми шифрування, перевірка введених значень, нескладні операції з відомостями.

Серверна частина (сервер додатків) – розміщений на другому рівні, на якому зосереджена найбільша частина бізнес-логіки. Поза лишаються лише фрагменти, що експортуються до клієнта, а також складники логіки, винесені у базу даних, наприклад процедури та тригери. Сервер додатків проектується так, аби додавання до нього додаткових примірників забезпечувало горизонтальне масштабування потужності програмного комплексу та не вимагало внесення змін у програмний код застосунку.

Сервер сховища даних – забезпечує збереження відомостей та виноситься на окремий рівень, реалізується зазвичай системами керування базою даних, з'єднання до цього елемента гарантується лише з рівня серверів застосунків.

У магістерській роботі клієнтська частина реалізована за допомогою Javascript фреймворка AngularJS. У ній втілена найпростіша логіка показу даних та незначне оперування ними:

- формування матриці шляхів;
- автентифікація особи;
- застосування google карт.

Сервери застосунків – реалізовані на базі платформи Nodejs. У них сконцентрована головна бізнес-логіка роботи веб-додатку:

- формування подорожей;
- розпізнавання номерних знаків автотранспорту;
- контакт із платіжною системою.

Сервери сховища даних реалізовані на MongoDB. Дана технологія дозволяє створити з простотою реплікаційну множину баз даних. При виході з ладу однієї з них, інші сховища голосують хто стане головною базою, а хто буде копіювати відомості для можливості подальшого поновлення при збої ще одного вузла.

Порівняно з дворівневою клієнт-серверною архітектурою, даний вид архітектури надає кращу масштабованість, завдяки горизонтальному масштабуванню серверів застосунків. Також дає змогу кращого налаштування системи завдяки відокремленості рівнів один від одного.

3.5 Опис технічних засобів автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт

Для забезпечення функціонування проектованої автоматизованої системи обліку автотранспорту через контрольні пункти, необхідні наступні технічні засоби: відеокамера; промисловий комп'ютер; радіолокаційний датчик; комутатор.

Промисловий комп'ютер. Промисловий комп'ютер UNO-3483G від компанії Advantech розроблений для монтажу у компактні щити автоматизації [19]. Він підтримує можливість додавання модулів розширення, а також встановлення жорстких дисків і твердотільних накопичувачів.

Цей комп'ютер вирізняється багатофункціональністю та універсальністю, завдяки наступним особливостям:

- подвійний вхід живлення, що забезпечує стабільну роботу навіть при відключенні одного з джерел енергії;
- широкий діапазон робочих температур від -20°C до $+60^{\circ}\text{C}$;
- високий рівень захисту корпусу IP67, що дозволяє використовувати пристрій у суворих умовах навколишнього середовища;
- можливість віддаленого управління, що дає змогу контролювати пристрій з операторського пункту.

Комп'ютер також має такі порти: два гігабітних GbE, два USB 3.0, один RS-232 та один RS-422/485.

Системні вимоги. Для коректної роботи сервер має відповідати наступним системним та апаратним вимогам (табл. 3.2).

Таблиця 3.2 – Системні та апаратні вимоги до сервера

№ п/п	Параметри	Значення
1	Операційна система	Ubuntu 16.04 64-розрядна
2	Платформа	Node.js не нижче 6.0
3	Процесор із тактовою частотою	2 ГГц або швидший – 64-розрядний (x64)
4	Оперативна пам'ять	16 гігабайт (ГБ)
5	Потреба у вільному місці на жорсткому диску не менше	8 Гб

Для коректної роботи програмного продукту клієнтського додатку персональний комп'ютер має відповідати таким мінімальним системним та апаратним вимогам (табл. 3.3).

Таблиця 3.3 – Системні та апаратні вимоги до клієнтського додатку

№ п/п	Параметри	Значення
6.	Операційна система	Windows 7/8/10 32 або 64-розрядна
7.	Процесор із тактовою частотою	1 ГГц або швидший – 32-розрядний (x86) або 64-розрядний (x64)
8.	Оперативна пам'ять	4 гігабайт (ГБ) (для 32-розрядної версії) або 16 ГБ (для 64-розрядної версії)
9.	Потреба у вільному місці на жорсткому диску не менше	256 Мб
10.	Браузер	Google Chrome 50, Mozilla Firefox 48, Edge 40, Opera 42 і вище

Відеокамера Для отримання високо якісних зображень використовується стаціонарна мережева відеокамера SNC-VB630 [20] і зображена на рисунку 3.9, а технічні характеристики в таблиці 3.4.



Рисунок 3.9 – Камера SNC-VB630

Вона забезпечує відеоспостереження Full HD з максимальною роздільною здатністю 1920 x 1080 пікселів і підвищеною чутливістю завдяки новій CMOS-матриці Exmor.

Таблиця 3.4 – Технічні характеристики радіолокаційного датчика

№п/п	Параметри	Значення
1.	Датчик зображення	1/2,9-дюймовий CMOS-датчик Exmor з прогресивною розгорткою
2.	Число ефективних пікселів	Близько 2.14 мегапікселів
3.	Система сигналів	NTSC/PAL (з перемиканням)
4.	Система синхронізації	Внутрішня синхронізація
5.	Мінімальна освітленість (50 IRE)	Колір: 0,1 лк (F1.2, View-DR вимк., VE вимк., АРУ вкл., 1/30 с, 30 кадрів/с) Ч/Б: 0,07 лк (F1.2, View-DR вимк., VE вимк., АРУ вкл., 1/30 с, 30
6.	Динамічний діапазон	Еквівалент 90 дБ із технологією View-DR
7.	Відношення сигнал/шум (посилення 0 дБ)	Більше 50 дБ
8.	Швидкість затвора	1/1 - 1/10 000 с
9.	Баланс білого	Автоматичне стеження за балансом білого (ATW, ATW-PRO); режими: люмінесцентна лампа, ртутна лампа, натрієва лампа, металогалогенідна лампа, білий світлодіод, баланс білого одним натисканням, ручний
10.	Об'єктив	Варіооб'єктив із вузлом кріплення CS Mount
11.	Діапазон масштабування	Оптичне масштабування, 2,9-кратне Цифрове масштабування 4x 11,6-кратне загальне масштабування
12.	Функція налаштування Easy Focus	Так
13.	Горизонтальний кут огляду	від 114,2° до 40,0°
14.	Вертикальний кут огляду	від 60,6° до 22,5°
15.	Фокусна відстань	f = від 2,8 мм до 8,0 мм
16.	Діафрагмова кількість	Від F1,2 (межа Wide) до F1,95 (межа Tele)
17.	Мінімальна відстань до об'єкту	300 мм
18.	Формат стиснення	H.264 (High/Main/Baseline Profile), JPEG
19.	Максимальна частота кадрів	H.264: JPEG: 60 кадр/с (1920 x 1080) JPEG: 60 кадр/с (1280 x 720)

Камера SNC-VB630 обладнана передовою системою обробки сигналів IPELA ENGINE EX, яка забезпечує високий рівень продуктивності завдяки сучасним можливостям, включаючи функції View-DR, XDNR та DEPA Advanced. Підвищена частота кадрів дозволяє досягти плавного та природного відтворення руху у відео, а технологія View-DR гарантує чіткість зображень навіть за умов низького освітлення, сильного контрового світла чи високого контрасту.

Вбудована система аналізу зображень вдосконалена за рахунок

додаткових можливостей, таких як розпізнавання облич, а також ідентифікація зниклих чи залишених об'єктів.

Камера SNC-VB630 є відмінним вибором для тих, хто потребує надійного та ефективного рішення для відеоспостереження з високою якістю зображення та розширеними функціями аналізу [20].

Радіолокаційний датчик RAS407 зображено на рисунку 3.10 з сайту [20].



Рисунок 3.10 – Радіолокаційний датчик RAS407 [20]

Параметри радіолокаційного датчика RAS407 наведені в таблиці 3.5 за даними виробника [20].

Таблиця 3.5 – Технічні характеристики радіолокаційного датчика

№п/п	Параметри	Значення
1.	Принцип вимірювання	FMCW HF
2.	Діапазон частот	24 ГГц ... 24,25 ГГц
3.	Робоча область	0,2 m ... 20 m
4.	Вигляд підключення	1 x Кабель з відкритим кінцем
5.	Напруга живлення	10 V DC ... 30 V DC
6.	Потужність	8 W
7.	Вага	580 г
8.	Розміри (Д x Ш x В):	101 мм x 151 мм x 60 mm
9.	Оцінка	< 0,5 s
10.	USB:	✓
11.	Цифрові виходи	3 (Реле) </i>
12.	Діапазон робочих температур	-20 °C ... +60 °C

Стиійка система спрацьовування навіть при забрудненні та в поганих погодних умовах.

Комутатор ЕКІ-55281 виробника Advantech є 8-ми портовим Fast Ethernet зображено на рисунку 3.11 з сайту [20].



Рисунок 3.11 – Комутатор ЕКІ-55281 [20]

Параметри Комутатор ЕКІ-55281 наведені в таблиці 3.6 за даними виробника [20].

Таблиця 3.6 – Технічні характеристики радіолокаційного датчика

№п/п	Параметри	Значення
1.	Стандарт Ethernet	IEEE 802.3, 802.3u, 802.3x, 802.1D, 802.1w, 802.1p, 802.1Q, 802.1x, 802.3ad, 802.3az
2.	Тип роз'єму порту	RJ-45
3.	відстань передачі	100 м
4.	Живлення	24 В
5.	Розміри (Ш х В х Г)	43 x 120 x 84 мм
6.	Споживання енергії	5,2 Вт (система)

Окремо комплектується схема з'єднувальними провідниками.

3.6 Опис інтерфейсу адміністратора системи обліку автотранспорту через контрольно-пропускний пункт

Реєстрація адміністратора в системі виконується під час запуску сервера. Адміністратор автоматично додається в базу даних із заздалегідь визначеним логіном і паролем. Подальший доступ до панелі адміністратора здійснюється через логін-форму (рис. 3.12).

Рисунок 3.12 – Логін форма в панель адміністратора

Після входу в систему користувачу відображається список доріг із зазначенням кількості контрольних пунктів, які необхідно попередньо внести в систему (рис. 3.13). На цій сторінці також доступна кнопка «Створити нову платну дорогу», натискання якої відкриває модальне вікно. У цьому вікні користувач може ввести назву нової платної дороги.

Платна дорога	Контрольні пункти		
E 40	4		
H-03	5		
E 95	3		

Кількість елементів на сторінку: 5 ▾ 1-3 з 3 < >

Рисунок 3.13 – Список створених доріг

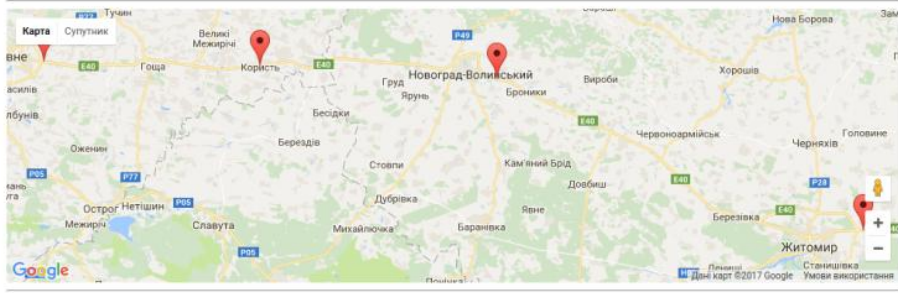
Після створення платної дороги, при натисканні на її назву, користувач бачить вікно, у якому відображається назва дороги, перелік контрольних пунктів із зазначеними координатами, а також карта, на якій розміщені ці пункти (рис. 3.14).

Для створення нового контрольного пункту потрібно натиснути кнопку «Створити новий контрольний пункт».

ДО СПИСКУ ДОРІГ

Дорога : Е 40 СТВОРИТИ НОВИЙ КОНТРОЛЬНИЙ ПУНКТ

Створений: Jun 10, 2017
Оновлений: Jun 11, 2017



Ім'я контрольного пункту	Широта	Довгота		
Рівне	50.60933428547427	26.351051330566406		
Житомир	50.28610369138736	28.818941116333008		

Рисунок 3.14 – Сторінка списку контрольних пунктів

Після цього з'явиться модальне вікно створення контрольного пункту (рис. 3.15). У цьому вікні слід вказати основні дані: назву дороги та координати розташування.


Створити новий Контрольний пункт

Name
Київ

Latitude
49.56211099999999

Longitude
34.5065889

Київське шосе, Полтава, Полтавська область, Україна



ВІДМІНИТИ ЗБЕРЕГТИ

Рисунок 3.15 – Сторінка створення контрольного пункту

Дані можна внести через поле введення з автодоповненням і пошуком через Google або шляхом розміщення точки за допомогою миші на карті. Такий підхід забезпечує зручний і швидкий пошук необхідного контрольного пункту.

Після завершення налаштування всіх необхідних маршрутів і контрольно-пропускних пунктів, на сторінці створення тарифних матриць при натисканні кнопки «Додати матрицю» відкривається модальне вікно. У цьому вікні користувач може обрати зі списку початкові та кінцеві точки за допомогою полів введення з автозаповненням, яке охоплює всі контрольні пункти (рис. 3.16). Крім того, у відповідному полі вводиться назва тарифної матриці.

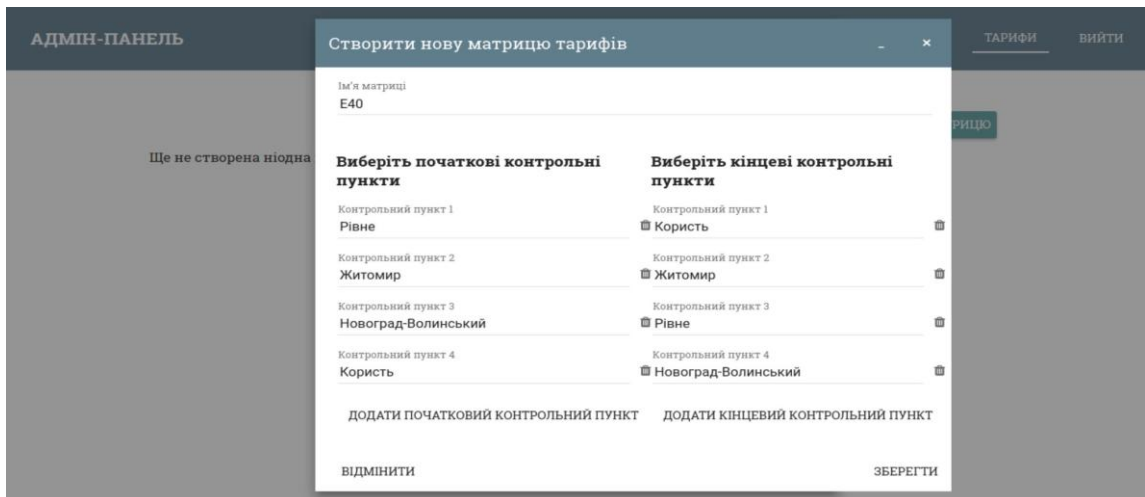


Рисунок 3.16 – Модальне вікно створення матриці тарифів

Після створення цієї матриці потрібно ввести значення тарифів для кожного можливого маршруту через відповідну платну дорогу (рис. 3.17).

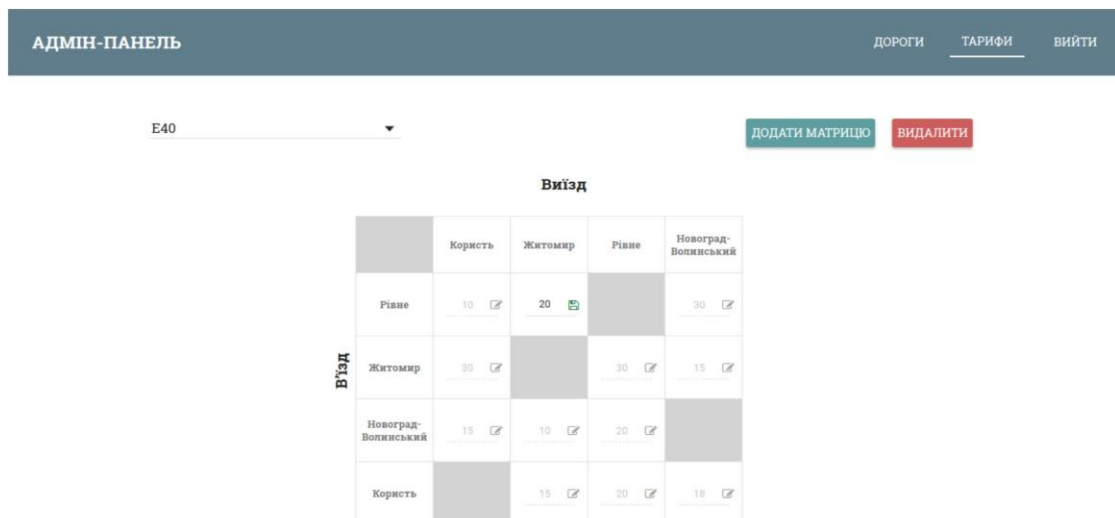


Рисунок 3.17 – Матриця тарифів, редагування значення тарифу

Для введення даних у кожен комірок слід натиснути на іконку редагування, що відкриває доступ до введення інформації у це поле.

3.7 Опис інтерфейсу веб-порталу користувача системи обліку автотранспорту через контрольно-пропускний пункт

Для доступу до веб-порталу користувачу необхідно спочатку заповнити реєстраційну форму, зазначену на рисунку 3.18. У цій формі потрібно вказати всі необхідні дані для реєстрації: ім'я, прізвище, електронну пошту, номер телефону та пароль.

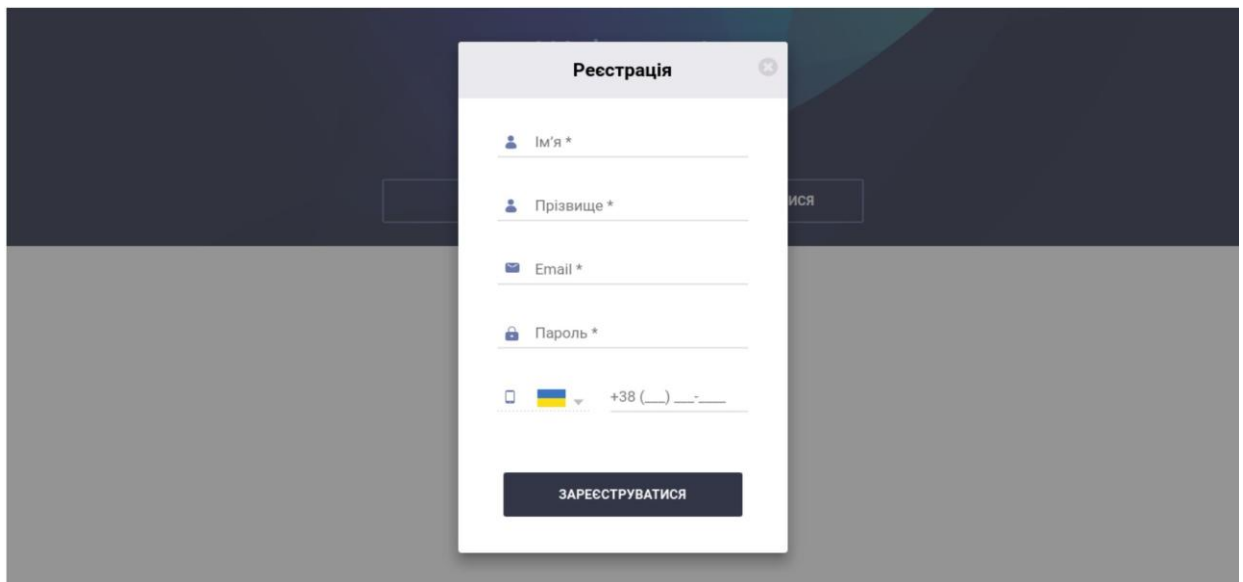


Рисунок 3.18 – Модальне вікно реєстрації користувача

Для подальшого заповнення профілю необхідно перейти до розділу налаштувань (рис. 3.19), де можна заповнити всі обов'язкові поля для активації функції автоматичної ідентифікації користувача на дорозі. На цій сторінці слід вказати інформацію про свій транспортний засіб.

Крім цього, доступна опція введення даних кредитної картки для створення платіжного профілю, що забезпечить можливість оплати проїзду через нашу систему.

Ім'я: Микола Василів

Місто: Київ

Клас автотранспорту: Клас В

Номер автотранспорту: 1218

ЗБЕРЕГТИ

Email: koliavasyliv@gmail.com

Область: Київська

Телефон: +380000000000

Адрес: Янгеля 5

Ввести свої платіжні дані

КАРТКИ ЯКІ ПРИЙМАЮТЬСЯ: VISA AMEX MasterCard DISCOVER

НОМЕР КАРТКИ: 4242 - 4242 - 4242 - 4242

EXP. DATE: 8 2018

CVV2: ...

АВТОМАТИЧНА ОПЛАТА:

ЗБЕРЕГТИ КРЕДИТНУ КАРТКУ

Рисунок 3.19 – Сторінка налаштування профіля користувача

На веб-порталі також передбачена функція перегляду неоплачених поїздок. За допомогою прапорців користувач має можливість обрати необхідні поїздки для оплати (рис. 3.20).

<input type="checkbox"/>	НАЗВА ДОРОГИ	КОНТРОЛЬНІ ПУНКТИ	ДАТА	ЦІНА
<input type="checkbox"/>	E40	Користь - Рівне	11.11.2017	15.00
<input checked="" type="checkbox"/>	E95	Новоград-Волинський - Рівне	11.11.2017	5.00
<input checked="" type="checkbox"/>	E40	Житомир - Київ	11.11.2017	10.00
<input type="checkbox"/>	H-02	Житомир - Користь	11.11.2017	15.00
<input checked="" type="checkbox"/>	E40	Рівне - Новоград-Волинський	11.11.2017	16.00

Елементів на сторінці: 5 1 - 5 з 20 < >

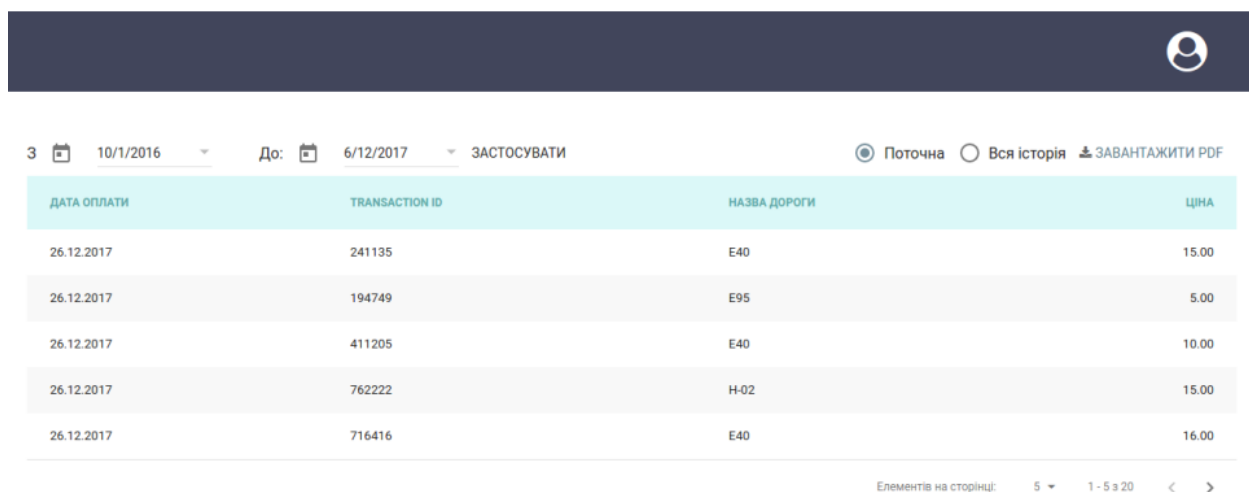
ЗАГАЛЬНА СУМА: 31

ОПЛАТИТИ ВИБРАНІ

Рисунок 3.20 – Список неоплачених поїздок

Процедура оплати здійснюється через кнопку «Оплатити вибрані». На сторінці відображається перелік усіх неоплачених поїздок із зазначенням назви дороги, контрольних пунктів, дати та вартості.

Користувач також може переглядати історію поїздок і фільтрувати її за потрібним часовим періодом (рис. 3.21).



The screenshot shows a user interface for viewing payment history. At the top, there is a dark navigation bar with a user profile icon. Below it, a filter section shows the date range from 10/1/2016 to 6/12/2017, with a 'ЗАСТОСУВАТИ' button. To the right, there are radio buttons for 'Поточна' (selected) and 'Вся історія', along with a 'ЗАВАНТАЖИТИ PDF' button. The main content is a table with the following data:

ДАТА ОПЛАТИ	TRANSACTION ID	НАЗВА ДОРОГИ	ЦІНА
26.12.2017	241135	E40	15.00
26.12.2017	194749	E95	5.00
26.12.2017	411205	E40	10.00
26.12.2017	762222	H-02	15.00
26.12.2017	716416	E40	16.00

At the bottom right of the table, there is a pagination control: 'Елементів на сторінці: 5' and '1 - 5 з 20'.

Рисунок 3.21 – Історія оплачених поїздок

На цій же сторінці доступна функція експорту даних у форматі PDF. Експорт можна здійснити як для всієї історії, так і для обраного періоду за допомогою інструментів вибору дати.

3.8 Опис структури додатку автоматизованої системи обліку автотранспорту через контрольньо-пропускний пункт

Основними архітектурними підходами до створення веб-додатків є монолітна структура та мікросервісна. Монолітна архітектура передбачає, що весь функціонал реалізований в одному проекті. Недоліком такої структури є уразливість: у разі збою сервера, припиняє роботу весь веб-додаток. Це відбувається через тісну інтеграцію всіх сервісів, модулів роботи з базою даних і взаємодії з клієнтом, які функціонують як єдиний вузол.

На відміну від цього, мікросервісна архітектура дозволяє додатку

продовжувати функціонувати навіть при відмові одного з її компонентів. Непрацездатність окремого сервісу вплине лише на той функціонал, за який він відповідає, за умови слабкої залежності між сервісами. Однак таку структуру доцільно використовувати лише для складних систем, оскільки її реалізація супроводжується додатковими труднощами. Важливо ретельно визначити чіткі межі між сервісами, щоб зміни в одному з них не зумовлювали необхідності модифікацій в інших.

У рамках цієї роботи обрано монолітну структуру проекту, оскільки обсяг функціоналу є достатньо компактним і не потребує поділу на окремі вузли зі строго окресленими зонами відповідальності.

Інтерфейс користувача реалізовано за допомогою бібліотеки QT, а бекенд написаний на C++ із використанням бібліотек OpenCV та Tesseract. Для розробки застосунку було застосовано шаблон проектування «Посередник» (Mediator) – поведінковий патерн, який знижує рівень зв'язаності між численними класами завдяки передачі управління в один клас-посередник [9]. Очікувано, використання цього шаблону спростило загальну структуру програми та централізувало керування в одному класі. Проте сам клас AppMediator став досить об'ємним. Додатковою перевагою є дотримання принципу відкритості/закритості, що полегшує розширення функціоналу системи. Загальна структура «Посередника» зображена на рисунку Б.1, а спрощена структура застосунку – на рисунку Б.2 додатку Б. У застосунку представлені такі класи (табл. 3.7).

Таблиця 3.7 – Класи та характеристики «Посередника»

№	Найменування класу	Опис функціоналу
1	AppMediator	забезпечує взаємодію між усіма класами застосунку
2	PlateDetector	відповідає за визначення номерного знака автомобіля на зображенні
3	PlateRecogniser	займається розпізнаванням тексту номерного знака автомобіля
4	ColorDetector	виконує функцію ідентифікації кольору автомобіля
5	MainWindow	відповідає за користувацький інтерфейс і обробляє різні події, наприклад, натискання кнопок
6	CarsDao	реалізує операції з базою даних; деякі методи цього класу використовують клас Car

Продовження таблиці 3.7

7	ErrorWindow	не має ключового функціоналу, але сприяє покращенню взаємодії з програмою. Сповіщає користувача про успішність чи невдачу операцій з базою даних
8	ImageDemonstrator	не має ключового функціоналу, але сприяє покращенню взаємодії з програмою. Відображає вибране зображення в окремому вікні
9	PositiveResultWindow	не має ключового функціоналу, але сприяє покращенню взаємодії з програмою. Сповіщає користувача про успішність чи невдачу операцій з базою даних
10	Mediator	Необхідний для реалізації шаблону «Посередник»
11	Component	Необхідний для реалізації шаблону «Посередник»

Для зберігання даних було обрано SQLite завдяки його простоті у використанні. У класі CarsDao реалізовані методи для додавання, видалення, оновлення та отримання даних із бази. Структура бази відносно мінімалістична: вона включає таблицю CAR, що містить записи формату (номер автомобіля, опис, марка, колір). Номер автомобіля визначено як унікальне значення, тому цей атрибут встановлено як primary key.

Для взаємодії з базою застосовується бібліотека SQLite3, яка використовує методи `sqlite3_exec()` або `sqlite3_step()`. Основна різниця між ними полягає у вхідних параметрах: метод `sqlite3_step()` потребує передачі об'єкта `sqlite3_stmt`.

Особливістю роботи з SQLite у C++ є обробка SELECT-запитів, які можуть потребувати додаткової уваги під час налаштування.

Клас AppMediator вийшов доволі об'ємним, як і очікувалось, проте, на мою думку, його код залишився достатньо структурованим і зрозумілим. Основне завдання цього класу полягає у забезпеченні зв'язку між інтерфейсом (клас MainWindow), модулями обробки зображень та класами доступу до бази даних.

Ідеальною реалізацією патерна «Посередник» було б використання лише одного універсального методу `notify()`, але на практиці довелося враховувати різні сигнатури методів для вирішення конкретних завдань.

3.9 Алгоритм створення платних доріг

Спершу через адмін-панель створюються платні дороги. До них додаються місця розташування пунктів збору мита. Локація обирається на Google-картах, звідки автоматично завантажуються координати місцевості.

Після налаштування основних елементів необхідно створити матрицю тарифів. Вона визначає розмір тарифу залежно від в'їзду та виїзду через встановлені контрольні точки на дорозі.

Під час проїзду користувач вказує початкову та кінцеву точку, після чого система автоматично визначає відповідний тариф для цих двох точок. По завершенні подорожі статус поїздки змінюється на завершений, але не оплачений.

Оплата поїздок. У даній роботі використовується платіжна система Authorize.Net, яка забезпечує можливість оплат за допомогою будь-якого типу платіжних карток. Крім того, цю систему можна перевіряти в тестовому режимі (пісочниці), використовуючи набір номерів тестових кредитних карток для оцінки її коректності.

Authorize.Net підтримує збереження даних кредитних карток, що дозволяє створювати профілі для підписок. Для кожного користувача створюється один загальний профіль – «customer». У рамках цього профілю користувач може додавати необмежену кількість платіжних профілів, кожен із яких відповідає окремій кредитній картці. Таким чином, користувач має змогу вибирати, якою саме картокою здійснити оплату.

Також у системі реалізовано механізм щомісячних підписок, за яким оплата проводиться автоматично першого числа кожного місяця, зі сповіщенням користувача через додаток.

При успішному виконанні платежу статус поїздки змінюється на оплачений. У колекції транзакцій створюється новий запис, який містить дату проведення операції, її статус (отриманий від Authorize.Net), а також ідентифікатор користувача.

Реалізація бази даних. У магістерській роботі пропонується використання

NoSQL бази даних MongoDB, яка працює з поняттями колекцій і документів. Колекції слугують аналогом таблиць, а документи є еквівалентом окремих рядків цих таблиць. Атомарність операцій у MongoDB забезпечується на рівні одного документа.

Дані в базі зберігаються у форматі BSON, що є бінарним представленням JSON-документів. Структура JSON збігається зі структурою об'єктів у JavaScript, що спрощує процес запису та зчитування даних із бази без необхідності використання проміжних бібліотек для конвертації форматів даних.

Документи MongoDB складаються з пар «поле-значення» і мають гнучку структуру. Значеннями полів можуть бути будь-які типи даних BSON, включаючи вкладені документи, масиви та навіть масиви документів.

У базі даних основними сутностями є користувачі, поїздки, платні дороги, контрольні пункти, матриці, тарифи й транзакції. Приклади структури документа і даних можна побачити на відповідних ілюстраціях у додатку (ліст. А.2 додатку А).

Таким чином, в магістерській роботі розглянуті ключові аспекти реалізації системи в цілому. Детально описано алгоритми створення поїздок та їх оплати, а також базу даних, яка забезпечує функціонування системи.

Визначені технічні вимоги до сервера і персонального комп'ютера користувача, необхідного для стабільної роботи автоматизованої системи контролю та обліку автотранспорту через контрольні пункти. Усі компоненти обладнання мають високий рівень захисту, адаптацію до широкого температурного діапазону та можливість працювати у різноманітних погодних умовах.

Сформовано опис основного інтерфейсу, що забезпечує взаємодію адміністратора системи з користувачами веб-порталу. Представлено цикл створення платних доріг із контрольними пунктами та налаштування тарифів. Також на веб-порталі демонструються такі елементи, як сторінка персональних налаштувань, модуль реєстрації користувачів і список нових поїздок.

ВИСНОВКИ

У кваліфікаційній роботі було вирішено науково-технічну задачу, спрямовану на розробку та дослідження роботи є автоматизованої системи обліку автотранспорту через контрольно-пропускний пункт.

У дослідженні були розглянуті основні типи автоматизованих систем контролю обліку автотранспорту через контрольні пункти. Такі системи орієнтовані на усунення затримок при проїзді через платні дороги завдяки впровадженню електронної оплати проїзду.

Детально розглянуто сучасні методи обробки зображень. Проаналізовано ключові методи розпізнавання та етапи, необхідні для попередньої обробки зображень. Проведений огляд існуючих сервісів у галузі комп'ютерного зору.

У цій роботі розглядається застосування мереж типу CNN на прикладі розпізнавання номерних знаків, хоча такі мережі також широко використовуються для візуального контролю якості продукції, виявлення дефектів під час лиття та зварювання, а також перевірки точності і правильності розміщення елементів на платах. Автоматизований контроль якості є необхідністю сучасності, що обумовлено потребою у швидкому отриманні контрольних показників, високими вимогами до якості продукції та необхідністю більш точної й кількісної оцінки, яка усуває суб'єктивність людського фактора.

У ході дослідження також проаналізовано модель нейронної мережі CNN для завдання розпізнавання номерних знаків. Під час роботи вдалося покращити характеристики навчання моделі, досягнувши точності 98.6%. Визначено оптимальні параметри та методи для використання в CNN: функція активації ReLU, ініціалізація параметрів методом He та застосування методу нормалізації пакетів (batch normalization) із більшим розміром ядра згортки.

У межах роботи було розроблено автоматизовану систему обліку автотранспорту через контрольні пункти, засновану на технології розпізнавання номерних знаків. Такий підхід ідентифікації є інноваційним і базується на

сучасному стеку комп'ютерних технологій. Основною перевагою системи є те, що не потрібно встановлювати додаткове обладнання на транспортний засіб, достатньо лише зареєструватись у системі автоматизованого збору платежів.

Запропоноване рішення передбачає створення автоматизованої системи обліку автотранспорту, яка працює через контрольні пункти без залучення додаткових пристроїв, таких як транспондери, зі сторони клієнтів. Ця система реалізована з використанням сучасних комп'ютерних технологій, оскільки ідентифікація транспортного засобу відбуватиметься за допомогою комп'ютерного зору. Обробка даних здійснюватиметься на сервері, побудованому на платформі Node.js із залученням зовнішніх сервісів та використанням розробленої моделі глибокої нейронної мережі для розпізнавання номерних знаків автомобілів.

Передача зображень здійснюється за допомогою промислових комп'ютерів, підключених до комутатора. До кожного комп'ютера приєднана камера та радіолокаційний датчик руху, який надсилає сигнал у момент, коли потрібно зафіксувати зображення. Камера виконує серію знімків, і на промисловому комп'ютері проводиться їх попередня обробка. Під час цієї обробки обирається зображення найвищої якості за певними критеріями, наприклад, за рівнем освітленості. Лише після цього вибране зображення передається на сервер для подальшої обробки, розпізнавання та ідентифікації користувача. Після завершення обробки користувач отримає сповіщення на телефон або в особистому кабінеті на вебпорталі. У сповіщенні міститимуться дані про поїздку: назва дороги, дата, час в'їзду та виїзду, а також сума оплати.

У процесі роботи були враховані всі необхідні технічні засоби для забезпечення функціонування системи. Загалом у розробці було реалізовано такі функції:

- серверна частина для ідентифікації користувача за номерним знаком транспортного засобу;
- сервіс здійснення оплати за допомогою кредитної картки;
- клієнтська частина, що включає адміністративну панель та вебпортал

для користувачів;

– модель нейронної мережі (CNN), створена для розпізнавання номерних знаків.

Наукова новизна одержаного результату. Удосконалено процес автоматизації обліку автотранспорту через контрольню-пропускний пункт.

Практичне значення результату полягає у реалізації можливості автоматизованому обліку автотранспорту через контрольню-пропускний пункт.

Прогноз подальших наукових досліджень. Подальші дослідження доцільно зорієнтувати на розв'язання проблеми автоматизації процесу обліку автомобілів при перетині блок-постів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Козубцова Л. М., Козубцов І. М., Бескровний О. І., Палагута А. М., Сухомлинова О.В. Перспективна технологія збору інформації про технічний стан військової техніки зв'язку. Міжнародна науково-практична конференція «Застосування інформаційних технологій у підготовці та діяльності сил охорони правопорядку» (Харків, 15 березня 2023). Харків: НАНГ України. С. 22-23.
2. Інструкція користувача. Розділ 12 Облік автотранспорту. Версія 7.11.032. 2019. 67 с.
3. Toll roads take cashless route. URL: http://usatoday30.usatoday.com/news/nation/2008-07-27-cashlesstoll-roads_N.htm (дата звернення: 5.09.2025).
4. Road tolling in Norway. URL: https://www.vegvesen.no/_attachment/109072/binary/187602. (дата звернення 10.09.2025).
5. Система розпізнавання номерних знаків. URL: <https://leater.com/en/services/car-number-plate-recognition-system.html> (дата звернення: 15.09.2025).
6. Система розпізнавання номера автомобіля. Загальна кількість спеціалістів із технічного обслуговування Neetia. URL: <http://enature.co.jp/product/vehicle> (дата звернення: 15.10.2025).
7. Google cloud Platform. URL: <https://cloud.google.com/vision/docs> (дата звернення: 15.09.2025).
8. Amazok Rekognition. URL: <https://aws.amazon.com/ru/rekognition> (дата звернення: 18.09.2025).
9. OpenCV Documentation. URL: <http://docs.opencv.org/3.2.0>. (дата звернення: 25.09.2025).
10. OpenALPR. URL: <https://github.com/openalpr/openalpr> (дата звернення: 26.09.2025).
11. LeCun Y. LeNet-5, Convolutional neural networks. URL: <http://yann.lecun.com/exdb/lenet> (дата звернення: 26.09.2025).
12. Shift Invariant Pattern Recognition Neural Network and Its Optical Architecture. URL: <https://www.scirp.org/reference/referencespapers?referenceid=3583>

056 (дата звернення: 10.10.2025).

13. LeCun Y., Cortes C. The MNIST database of handwritten digits. URL: https://www.lri.fr/~marc/Master2/MNIST_doc.pdf. (дата звернення 15.10.2025).

14. Glorot X., Bengio Y. Understanding the difficulty of training deep feedforward neural networks. URL: <http://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf>. (дата звернення: 15.10.2025).

15. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. URL: <https://arxiv.org/pdf/1502.01852.pdf> (дата звернення: 15.10.2025).

16. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. URL: <https://arxiv.org/pdf/1502.03167.pdf> (дата звернення: 15.10.2025).

17. Промисловий ПК UNO-3483G. URL: https://www.advantech.com/ru-ru/products/1-2mljdc/uno-3483g/mod_b9471e67-653c-4681-8b11-aa54a2d5c8e0 (дата звернення: 15.10.2025).

18. Стационарна мережева відеокамера SNC-VB630. URL: <https://www.ipteam.com.ua/ip-cams/ip-cameras-sony/fixed-sony/vb630.html> (дата звернення: 15.10.2025).

19. Радарний датчик Sick RAS407-2801000. ТОВ «Трейд Контрол». URL: <https://trade-control.com.ua/ua/products/sick-ras407-2801000> (дата звернення 15.10.2025).

20. Комутатор ЕКІ-55281. Advantech. URL: https://www.advantech.com/ru-ru/products/ca824526-87c9-4fa4-9fc0-c635a85ad2e5/eki-5528i-mb/mod_74c6696a-ca2e-4c0e-a587-75639e92f721 (дата звернення: 15.10.2025).

ДОДАТКИ

ДОДАТОК А

Лістинг програмного забезпечення

Лістинг А.1 – Фрагмент коду опису моделі за допомогою бібліотеки TensorFlow

```
const model = tf.sequential();
  model.add(tf.layers.conv2d({ inputShape: [28, 28, 1], kernelSize: 3, filters: 32, strides: 1,
activation: 'relu', kernelInitializer: 'heNormal' }));
  model.add(tf.layers.maxPooling2d({ poolSize: [2, 2], strides: [2, 2] }));
  model.add(tf.layers.conv2d({ kernelSize: 3, filters: 64, strides: 1, activation: 'relu',
kernelInitializer: 'heNormal' }));
  model.add(tf.layers.maxPooling2d({ poolSize: [2, 2], strides: [2, 2] }));
  model.add(tf.layers.flatten()); model.add(tf.layers.dropout({rate: 0.25}));
  model.add(tf.layers.dense({ units: 512, kernelInitializer: 'heNormal', activation: 'relu' }));
  model.add(tf.layers.dropout({rate: 0.5})); model.add(tf.layers.dense({units: 10, activation:
'softmax'}));
  // Defining the optimizer
  const LEARNING_RATE = 0.015;
  const optimizer = tf.train.adam(LEARNING_RATE);
  model.compile({ optimizer: optimizer, loss: 'categoricalCrossentropy', metrics: ['accuracy'],
});
```

кінець лістингу А.1

Лістинг А.2– Приклад частини документа користувача в базі даних

```
{
  firstName: "testl",
  lastName: "testl",
  updatedAt: Date("Jun 12, 1963"),
  customerProfileId: "21da41das23SQ4",
  subscriptions: [
    {
      paymentProfileId: "42fd12da24",
      interval: 2,
      unit: "month", enabled: true
    }
  ],
  paymentProfiles: [
    { paymentProfileId: "42fd12da24", cardNum: "2345"
    }
  ]
}
```

кінець лістингу А.2

ДОДАТОК Б

Графічний матеріал

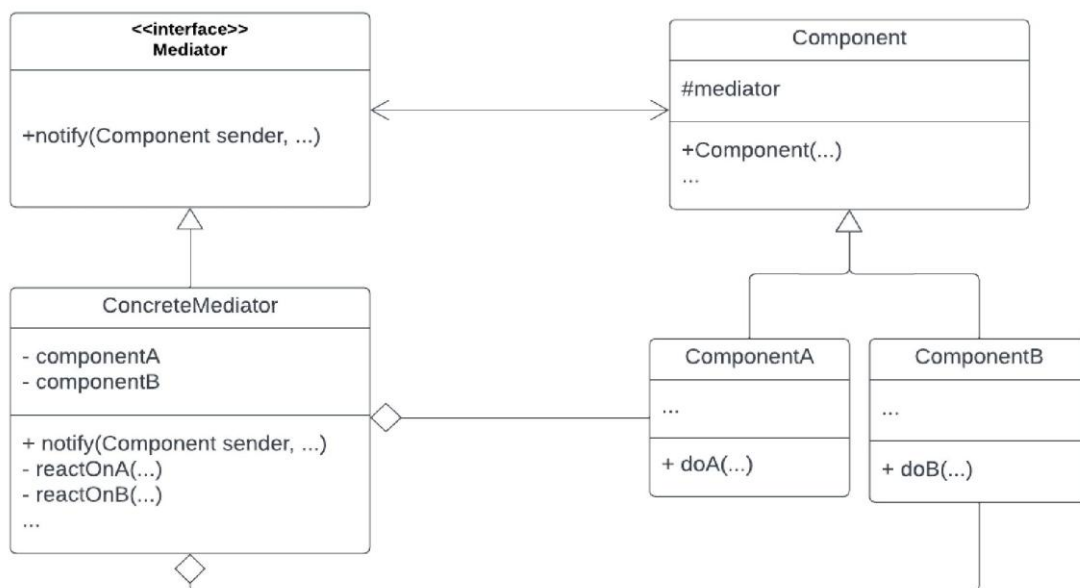


Рисунок Б.1 – Загальна структура «Посередника»

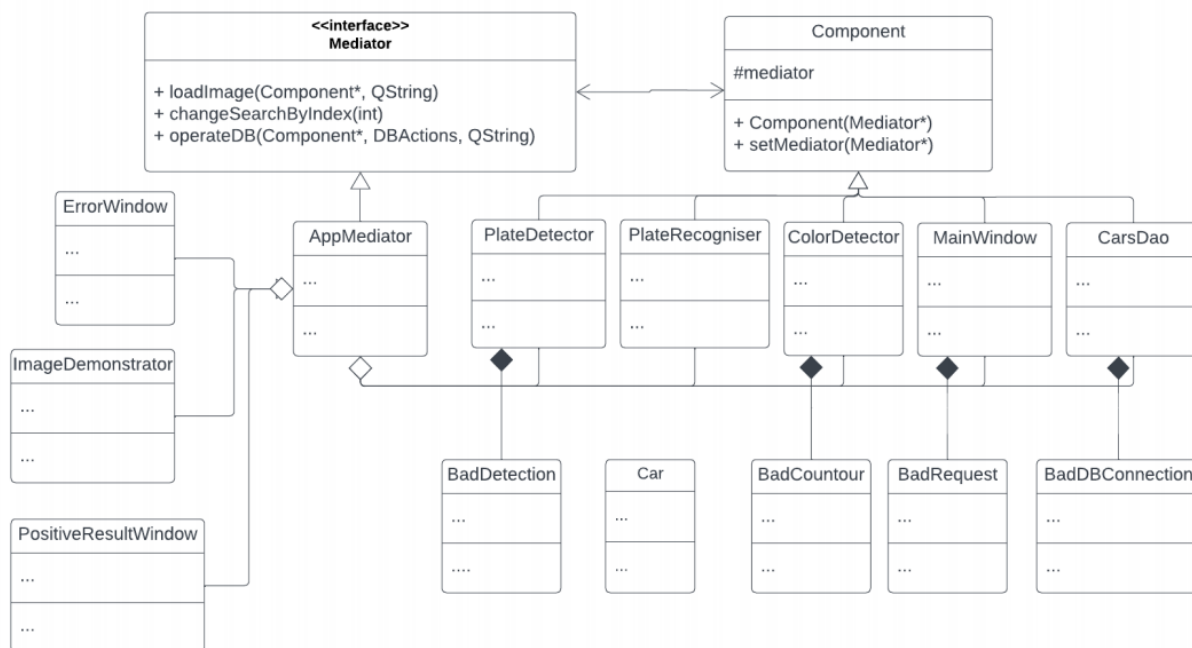


Рисунок Б.2 – Структура застосунку (спрощена)