

Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

на тему:

**КОМП'ЮТЕРНА СИСТЕМА ВИЗНАЧЕННЯ ТА КОНТРОЛЮ  
РЯДНОСТІ РУХУ**

Виконав: студент 2 курсу, групи КІ-22мз  
спеціальності 123 — «Комп'ютерна інженерія»

О.О. Гринчак Гринчак О.О.

Керівник: к.т.н., доц.каф. ОТ

І.С. Колесник Колесник І.С.

« 14 » 06 2024 р.

Опонент: д.т.н., проф. каф. МБІС

Ю.Є. Яремчук Яремчук Ю.Є.

« 16 » 06 2024 р.

**Допущено до захисту**

Завідувач кафедри ОТ

О.Д. Азаров д.т.н., проф. Азаров О. Д.

« 16 » 06 2024 р.

ВНТУ 2024

## ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

Галузь знань — Інформаційні технології

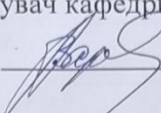
Освітній рівень — магістр

Спеціальність — 123 Комп'ютерна інженерія

Освітня програма — Комп'ютерна інженерія

**ЗАТВЕРДЖУЮ**

Завідувач кафедри обчислювальної техніки

 д.т.н., проф. О.Д. Азаров

"12" 03 2024 р.

### ЗАВДАННЯ

#### НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Гринчак Ользі Олегівні

1 Тема роботи «Комп'ютерна система визначення та контролю рядності руху» керівник роботи Колесник Ірина Сергіївна к.т.н., доцент, затверджено наказом вищого навчального закладу від 11.03.24 року № 81

2 Строк подання студентом роботи 12.06.24р..

3 Вихідні дані до роботи: призначення: призначення системи – виявлення первинних зміни рядності руху, надання сигналу про появу такої зміни, тип системи — розподілена, призначення — організація взаємодії з водіями різних типів транспортних засобів, передача повідомлень про зміну контрольованих параметрів у візуальний блок сповіщення.

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): обґрунтування проблеми, методів та технік застосування комп'ютеризованих систем пожежогасіння, математичні моделі для виявлення джерел загоряння на зображенні, структурна реалізація комп'ютерної системи для пожежної сигналізації із визначенням джерела загоряння, виявлення пожежі за допомогою датчика та камери, економічна частина, висновки.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): структурна схема приймальної частини комп'ютерної пожежної системи пожаротушіння, блок-схема методу пошуку центру осередку

займання, схема електрична структурна системи, схема електрична структурна модулю, кінцевий модуль, лістинг коду програми методу пошуку центру осередку займання.

6 Консультанти розділів роботи приведені в таблиці 1.

Таблиця 1 — Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Колесник Ірина Сергіївна к.т.н., доцент	12.03.2024р	12.03.2024
5	Небава Микола Іванович проф., к.е.н		

7 Дата видачі завдання 12.03.2024р

8 Календарний план виконання МКР приведений в таблиці 2.

Таблиця 2 — Календарний план

№ з/п	Назва етапів МКР	Строк виконання	Підпис
1	Постановка задачі	19.03.24	
2	Аналіз предметної області	20.03.24	
3	Розробка структурної схеми	25.03.24	
5	Розрахунок аналогової частини	1.04.24	
6	Вибір ПЗ для розробки	19.04.24	
7	Розробка роботи нейромережі	20.04.24	
8	Розрахунок економічної частини	10.05.24	
9	Оформлення пояснювальної записки та ілюстративного матеріалу	19.05.24	
10	Виконання МКР	26.05.24	
11	Перевірка якості виконання магістерської кваліфікаційної роботи та усунення недоліків	3.06.24	
12	Підписи супроводжувальних документів у керівника, опонента, нормоконтролера	05.06.24	
13	Перевірка «антиплагіат»	05.06.24	

Студент Гринчак Ольга Олегівна

Керівник

к.т.н., доц. Колесник Ірина Сергіївна

## АНОТАЦІЯ

УДК004

Гринчак О.О. Комп'ютерна система визначення та контролю рядності руху. Магістерська кваліфікаційна робота зі спеціальності 123 — Комп'ютерна Інженерія, Вінниця: ВНТУ, 2024 — 106 с. На укр. мові. Бібліогр.: 45 назв; рис.: 31; табл.6.

Магістерська кваліфікаційна робота аналізує вплив зростаючих транспортних потреб на автомобільний парк та кількість поїздок. Також розроблено системи керування транспортними засобами на основі аналізу зображень, яка виявляє ризиковані маневри та допомагає уникнути аварій. В роботі представлено систему, яка виявляє ризиковані маневри, включаючи неправильні зміни смуги руху з фільтром Калмана та інші алгоритми. Враховуючи особливості доріг, пропонується покращити розташування дорожніх ліній та розпізнавання типів дорожньої розмітки. Ця робота вирішує актуальні проблеми та може покращити безпеку на дорогах.

Ключові слова: лінія дорожньої розмітки, комп'ютерна система, колірна модель, віртуальні сенсори, технічний зір.

## ABSTRACT

UDC 004

Hrynychak O.O. Computer system for determining and controlling the traffic line. Master's qualification work inspecialty 123 – Computer Engineering, Vinnytsia: VNTU, 2024 - 10 p. In Ukrainian. Bibliography: 45 titles; Figures: 31; Table 6.

The master's thesis analyzes the impact of growing transportation needs on the vehicle fleet and the number of trips. It also develops image analysis-based vehicle control systems that detect risky maneuvers and help avoid accidents. This paper presents a system that detects risky maneuvers, including improper lane changes with a Kalman filter and other algorithms. Taking into account the peculiarities of roads, it is proposed to improve the location of road lines and the recognition of road marking types. This work solves current problems and can be of great importance for road safety.

Keywords: road marking line, computer system, color model, virtual sensors, technical vision.

## ЗМІСТ

<b>ВСТУП</b> .....	8
<b>1 ОГЛЯД ЗАДАЧІ РОЗПІЗНАВАННЯ ДОРОЖНЬОЇ РОЗМІТКИ ТА СИСТЕМНИХ ФУНКЦІЙ, ЩО ВИКОНУЮТЬ ЦЮ ЗАДАЧУ</b> .....	10
1.1 Аналіз можливостей систем керування зміною смуги руху .....	10
1.2 Сучасні можливості інтелектуальних комп'ютерних систем та їх можливих рівнів автоматизації.....	11
1.3 Застосування людино-машинних інтерфейсів в комп'ютерних системах керування автомобілем .....	17
1.4 Канали взаємодії людино-машиного інтерфейсу при визначенні рядності руху .....	19
<b>2 МАТЕМАТИЧНА МОДЕЛЬ ДЕТЕКТУВАННЯ ДОРОЖНЬОЇ РОЗМІТКИ ТА АВТОМАТИЧНОГО КЕРУВАННЯ</b> .....	26
2.1 Деталізація блоку обробки відеоінформації .....	26
2.2 Автокалібрування системи стереозображення при визначенні розмітки дороги.....	28
2.3 Краї та їх розташування .....	30
2.4 Диференціальні оператори.....	31
2.5 Виявлення та визначення країв.....	32
2.6 Реєстрація та обробка інформацію з датчиків .....	35
2.7 Комп'ютерна система автоматичного керування .....	37
<b>3 МОДЕЛЮВАННЯ СИСТЕМИ КОНТРОЛЮ РЯДНОСТІ РУХУ</b> .....	43
3.1 Вибір оптимального методу виявлення .....	43
3.2 Модель дорожньої смуги .....	55
3.3 Визначення транспортного засобу в просторі .....	58

					08-54.МКР.002.00.000 ПЗ		
		№ докум.	Підпис				
Розробив	Гринчак О.О.			Комп'ютерна система визначення та контролю рядності руху <i>Пояснювальна записка</i>	Лім.	Арк.	Аркушів
Керівник	Колесник І.С.				6		
Рецензент	Яремчук Ю.Є.				<b>ВНТУ, гр. 1КІ-22мс</b>		
Н. Контроль	Швець С. І.						
Затверджую	Азаров О. Д.						

3.4 Властивості вибраної CCD-матриці .....	60
<b>4 РЕАЛІЗАЦІЯ СЕРЕДОВИЩА ВИКОНАННЯ ПРОГРАМИ .....</b>	<b>63</b>
4.1 Програмні складові .....	63
4.2 Інструкція користувача .....	65
<b>5 ЕКОНОМІЧНА ЧАСТИНА .....</b>	<b>71</b>
5.1 Комерційний та технологічний аудит науково-технічної розробки....	71
5.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи .....	74
5.3 Нарахування на заробітну плату розробників .....	75
5.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності .....	82
<b>ВИСНОВКИ .....</b>	<b>87</b>
<b>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....</b>	<b>89</b>
<b>ДОДАТОК А</b> Технічне завдання .....	<b>94</b>
<b>ДОДАТОК Б</b> Структурна схема комп'ютерної системи контролю рядності руху .....	<b>98</b>
<b>ДОДАТОК В</b> Блок-схема методу визначення смуги руху .....	<b>99</b>
Схема електрична структурна системи .....	100
<b>ДОДАТОК Г</b> Схема електрична структурна блоку обробки зображення .	<b>101</b>
<b>ДОДАТОК Д</b> Модуль захоплення відео .....	<b>102</b>
<b>ДОДАТОК Е</b> Лістинг коду програми методу пошуку віртуальних сенсорів .....	<b>103</b>
<b>ДОДАТОК И</b> Протокол перевірки кваліфікаційної роботи нанаявність текстових запозичень .....	<b>106</b>

## ВСТУП

Сьогодні комп'ютерні системи використовуються в багатьох сферах людської діяльності як зручний і багатофункціональний інструмент для вирішення багатьох завдань. Багато галузей техніки, пов'язаних зі збором, обробкою, зберіганням і передачею інформації, в даний час значною мірою орієнтовані на розвиток систем, в яких інформація базується на зображеннях. Зображення, яке можна вважати двовимірним сигналом, несе інформацію з набагато більшою ємністю, ніж звичайний одновимірний (часовий) сигнал. Водночас, при роботі з візуальними даними, вирішенні наукових та інженерних завдань потрібні зусилля, засновані на знанні спеціальних методів, оскільки традиційна ідеологія одновимірних знаків і систем у цих випадках малоприматна. Особливо це проявляється при створенні нових типів інформаційних систем, які вирішують завдання, які ще не були вирішені в науці і техніці і зараз вирішуються завдяки використанню такої візуальної інформації [1-3].

**Актуальність** теми полягає в необхідності створення високопродуктивних, високоточних засобів, які дозволяють обробляти, аналізувати та розпізнавати інформацію в режимі реального часу, що, в свою чергу, надає можливість розробляти нові технічні рішення, які підвищують якість життя та безпеку.

**Об'єкт дослідження** — процес отримання та оброблення відеоінформації в системі визначення смуги руху.

**Предмет дослідження**— методи та засоби комп'ютеризованих систем технічного зору для аналізу дорожньої розмітки.

**Мета роботи** полягає у розширенні функціональних можливостей комп'ютерної системи для попередження та реєстрації несанкціонованої зміни рядності руху із застосування методу детектування дорожньої розмітки, що дозволяє отримати кращі показники швидкодії та точності детектування.

Для досягнення поставленої мети потрібно виконати такі **задачі**:



—проаналізувати існуючі методи детектування дорожньої розмітки; покращити методи детектування та супутні алгоритми роботи системи;

—запропонувати структурні та функціональні схеми;

—обрати найкращі варіанти програмної реалізації комп'ютерної системи та оптимізувати код для покращення швидкодії комплексу;

—підібрати актуальні елементи системи

**Наукова новизна** — удосконалено метод визначення місцезнаходження полоси дорожньої розмітки за рахунок введення в автомобільну комп'ютерну систему стереовізійних засобів та віртуальних сенсорів, що дозволить підвищити швидкість виявлення засобів гасіння.

**Практичне значення** одержаних результатів полягає в тому, що на основі запропонованих методів та теоретичних положень розроблена комп'ютерна система, яка дозволяє вчасно відслідковувати зміну руху автомобіля та дозволить зменшити кількість ДТП.

Апробація результатів. Основні положення та результати виконані в магістерській роботі досліджень доповідались та обговорювались на конференції:

- КОНФЕРЕНЦІЇ ВНТУ електронні наукові видання, Молодь в науці: дослідження, проблеми, перспективи (МН-2024)

# **1 ОГЛЯД ЗАДАЧІ РОЗПІЗНАВАННЯ ДОРОЖНЬОЇ РОЗМІТКИ ТА СИСТЕМНИХ ФУНКЦІЙ, ЩО ВИКОНУЮТЬ ЦЮ ЗАДАЧУ**

## **1.1 Аналіз можливостей систем керування зміною смуги руху**

Дорожньо-транспортні інциденти є поширеною причиною смертності у світовому масштабі. Автопригоди трапляються за дуже різних обставин і з найрізноманітніших причин. У 1996 році було зафіксовано понад 37 000 дорожньо-транспортних пригод зі смертельними наслідками, в яких жертвами стали 42 000 осіб [5]. Загальний матеріальний збиток від дорожньо-транспортних випадків оцінюється більш ніж у 150 мільярдів американських доларів на рік. Донедавна зусилля, спрямовані на запобігання аваріям або зменшення їх негативних наслідків, обмежувалися такими підходами, як обов'язкове використання ременів безпеки, вимога до автовиробників встановлювати подушки безпеки в стандартній комплектації, зниження швидкісних обмежень та встановлення відбійників [5]. Незважаючи на те, що більшість цих методів мали суттєвий позитивний вплив на кількість ДТП, сьогодні можна досягнути більшого завдяки впровадженню новітніх технологічних досягнень.

У роботі представлено систему, яка покликана зменшити кількість катастроф, спричинених аваріями певного характеру: Зїзд з дороги (ROR), також відомий як виїзд з дороги одним учасником дорожнього руху (SVRD). Причинами цього типу ДТП є зокрема неуважність, алкогольне сп'яніння, сонливість та неумисні рухи рульового колеса [6], які, в свою чергу, призвели до смертельних наслідків на автошляхах.

Згідно з результатами нещодавнього дослідження, люди, які спілкуються по мобільному телефону за кермом, мають більше шансів потрапити в аварію, ніж водії, які були в стані алкогольного сп'яніння [7]. Наприклад, водій, який одночасно розмовляє по мобільному телефону, може не відреагувати на те, що автомобіль повільно з'їжджає з дороги, або не розгледіти поворот. У такому випадку неможливість повністю управляти транспортним засобом може призвести до нещасного випадку.

Наразі найбільш поширеним методом запобігання самовільного виїзду зі смуги руху є організація спеціальних ділянок асфальту на обабіч дороги. Це хвилясті ділянки дорожнього полотна, які зазвичай розміщені на відстані до 6-15 см від краю смуги руху [8]. Коли автівка з'їжджає зі своєї смуги, її передні колеса потрапляють на цю ділянку, спричиняючи вібрацію та різкий шум, який оповіщає водія про з'їзд зі смуги та допомагає йому взяти заходів для усунення недоліків керування автомобілем.

Хоча системи попередження про виїзд за смугу руху (LDW) розробляються вже багато років, їх безпосереднє практичне застосування в конструкціях сучасних автомобілів розпочалося лише нещодавно.



Рисунок 1.1 — Системи попередження про виїзд за смугу руху

1.2 Сучасні можливості інтелектуальних комп'ютерних систем та їх можливих рівнів автоматизації

В даний час дві основні теми досліджень, що стосуються інтелектуальних транспортних засобів, - це системи допомоги водієві Advancend (ADAS) і автономні транспортні засоби. Інтерес до досліджень в

області автономних транспортних засобів був високим з кінця 1990-х років, коли мета створення повністю автономного транспортного засобу здавалася дуже близькою; в той же час ADAS почали сильно приваблювати виробників транспортних засобів, особливо з огляду на те, що масове виробництво було майже здійсненим при обмежених витратах. В останні роки, завдяки імпульсу деяких провідних компаній у своїх галузях і завдяки новим потребам, що виникають у зв'язку з "зеленою економікою", інтерес до автономних транспортних засобів знову став дуже високим.

Особливо в певних чітко визначених галузях, таких як сільське господарство, гірничодобувна промисловість, вибухотехнічні роботи і, нарешті, військова сфера. ADAS в основному призначені для використання в заздалегідь визначених і дуже обмежених сценаріях, наприклад, на автомобільних дорогах, на автостоянках, в ситуаціях руху ззаду і т.д.; їх ефективність повинна бути дуже високою у відповідних сферах застосування з низьким рівнем допустимих помилок. На додаток до цього, вартість і невеликі розміри є важливими характеристиками, які дозволяють легко інтегрувати і згодом комерціалізувати їх на автомобільному ринку [9, 10]. З іншого боку, в автономних транспортних засобах фактори вартості і розміру є менш важливими, оскільки створювані системи не повинні інтегруватися в уже існуючі транспортні засоби, а представляють абсолютно нову лінійку транспортних засобів; з іншого боку, автономні транспортні засоби повинні справлятися з набагато більш різноманітними, менш структурованими ситуаціями і, зокрема, повинні справлятися з усіма погодними умовами, всіма типами місцевості і всіма сценаріями, з якими звичайний транспортний засіб стикається протягом свого життєвого циклу [11, 12, 13].

Наприклад, концепції сприйняття, планування та реалізації ідентичні в обох системах, але різні вимоги з точки зору вартості та інтеграції призводять до різних систем сприйняття та обробки отриманих даних [14]. У двох словах, основна відмінність між двома типами систем полягає в наявності або відсутності водія в транспортному засобі з усіма наслідками,

що з цього випливають, особливо в разі помилки в автоматизованій системі водіння. В даний час основні DAS, такі як ABS, ESP і т.д., втручаються в поведінку автомобіля, не дозволяючи водієві перевизначати команди, оскільки вони характеризуються дуже обмеженим, але дуже точним типом сприйняття, добре деталізованою і добре описаною системою прийняття рішень і застосовуються тільки в чітко визначених сценаріях; крім того, автоматизація, впроваджена DAS, дуже обмежена в часі і триває лише кілька миттєвостей. ADAS, з іншого боку, включає в себе дуже складну систему сприйняття і впровадження, але обмеження витрат не дозволяють створювати надлишкові системи сприйняття, а сфера застосування, як і у випадку з DAS, є дуже обмеженою і чітко визначеною.

Насправді, між DAS, ADAS і автоматичним керуванням є й інші рівні можливого автоматизації, які аналізуються в наступному розділі.

Як зазначалося, дві основні теми досліджень в галузі ІТС можуть бути включені в більш широкий аналіз можливих рівнів автоматизації інтелектуального транспортного засобу. Зокрема, можна виділити п'ять рівнів: помічник з експлуатації; асистент водія; напівавтоматичне водіння; високоавтоматизоване водіння; повністю автоматизоване водіння (рис 1.2).

Рівні перераховані в порядку зростання автоматизації, тобто на першому рівні втручання автоматики найменше, а на останньому – найбільше. На першому рівні цієї шкали ви маєте повністю ручне керування, так само, як керують будь-яким сучасним автомобілем, тобто за допомогою класичних аналогових елементів керування рульовим керуванням, прискоренням і гальмуванням. На другому рівні, відомому як "Асистоване водіння", автомобіль може надсилати водієві попередження, не беручи, однак, на себе безпосередній контроль над транспортним засобом.

	SAE РІВЕНЬ 0	SAE РІВЕНЬ 1	SAE РІВЕНЬ 2	SAE РІВЕНЬ 3	SAE РІВЕНЬ 4	SAE РІВЕНЬ 5
Що повинен робити водій?	Ви керуєте автомобілем, навіть якщо ноги не на педалях, а руки не на рулі.			Ви <u>не</u> керуєте автомобілем, якщо активовані функції автоматичного водіння.		
	Ви маєте слідкувати за електронними помічниками: рулити, гальмувати або прискорюватись, якщо потрібно.			якщо система буде цього вимагати, необхідно керувати самостійно.	Електронні системи не вимагають від водія керувати автомобілем самостійно.	
	Системи допомоги водію			Системи безпілотного водіння		
Як вони працюють?	Помічники лише попереджують і надають короткотривалу допомогу.	Помічники допомагають керувати АБО прискорюватись/гальмувати.	Помічники допомагають керувати ТА прискорюватись/гальмувати.	Система може самостійно керувати автомобілем лише за умови одночасного виконання кількох умов.		Система може керувати автомобілем за будь-яких умов.
Приклади систем	Автоматичне аварійне гальмування Попередження про сліпі зони Попередження про покидання смуги	Утримання на смузі АБО Адаптивний круїз-контроль	Утримання на смузі ТА Адаптивний круїз-контроль одночасно	Помічник піг час руху у дорожніх заторах.	Місцеве безпілотне таксі. Може бути відсутній руль або педаль.	Те саме, що і рівень 4., але можливість автоматизованого пересування зберігається

Рисунок 1.2 — Рівні автоматизації інтелектуального автомобіля

Це стосується системи попередження про виїзд зі смуги руху (LDW) - системи, яка попереджає водія про перетин розділювальної лінії між однією проїжджою частиною та іншою; ця система попереджає користувача про перетин, що відбувається, щоб він міг належним чином відреагувати, якщо перетин не передбачений, але не втручається безпосередньо в керування транспортним засобом.

Напівавтоматичне водіння характеризується обмеженим втручанням автоматики в керування автомобілем.

Попередження від системи навігації, але також і обмежене в часі втручання. Втручання часто мотивовані міркуваннями безпеки або зручності для водія. Прикладами застосувань на цьому рівні є

Асистент утримання в смузі руху: цей додаток допомагає водієві утримувати автомобіль в межах поточної смуги руху, вносячи невеликі корективи, якщо автомобіль збирається перетнути розмітку; в той же час, водій може вирішити, використовуючи кермо, відмінити дії системи і добровільно змінити смугу руху.

Удосконалений круїз-контроль: завданням цієї системи є утримання автомобіля на певній швидкості з дотриманням безпечної дистанції до транспортного засобу, що рухається попереду.

Наступний рівень, який називається "Високоавтоматизоване водіння", характеризується ще більшим рівнем автоматизації. Зокрема, якщо водій обирає бажаний тип автоматизації, то автоматика буде відповідати за керування транспортним засобом навіть на великі відстані. Прикладів такого типу водіння може бути багато, але наразі жоден з них не реалізований у комерційних транспортних засобах. Дуже цікавим прикладом є поєднання систем LKS і ACC: ця нова ADAS зможе утримувати транспортний засіб у межах смуги руху на певній швидкості, водночас дотримуючись безпечної дистанції до транспортного засобу попереду. Таке застосування дуже корисне в таких ситуаціях, як рух по автомагістралі, де автоматизація може взяти на себе керування транспортним засобом навіть на дуже довгі відстані і час, полегшуючи таким чином навантаження на водія транспортного засобу.

Останній рівень за шкалою автоматизації називається "Повністю автоматизоване водіння". На цьому рівні водієві потрібно лише визначити пункт призначення своєї подорожі, оскільки контроль над транспортним засобом повністю знаходиться в руках автоматизованої системи водіння, завданням якої є планування та виконання поїздки від початкової точки до пункту призначення, обраного водієм.

З підвищенням рівня автоматизації змінюється тип взаємодії між системою водіння і водієм: якщо на низьких рівнях автоматизації інтелектуальний транспортний засіб здатний лише пропонувати пропозиції або виправлення, то на високих рівнях автоматизації штучний інтелект може запропонувати водієві серію повних рішень, навіть на різних рівнях складності; це приводить нас до розуміння того, що з підвищенням рівня взаємодії між транспортним засобом і водієм можуть виникати нові і різні конфлікти на різних рівнях; наступний розділ буде присвячений аналізу цих конфліктів і їх можливого вирішення.

В ситуації, коли водій керує транспортним засобом вручну; в певний момент автоматизована система водіння виявляє умови (наявність розмежувальних ліній, правильне виявлення інших транспортних засобів тощо) для високоавтоматизованого водіння і, відповідно, пропонує низку можливих маневрів, наприклад, рух по автомагістралі та обгін транспортного засобу, що їхав попереду. В результаті водій може вирішити увімкнути автопілот і доручити інтелектуальному автомобілю продовжити рух замість нього. З іншого боку, автомобіль може оцінити маневр як небезпечний, якщо, як показано на рисунку 1.4, сприйняття виявить зустрічний транспортний засіб у лівій смузі, і автоматизований автомобіль відмовиться виконувати маневр, а між водієм і автомобілем виникне конфлікт.

З іншого боку, водій повинен мати можливість повністю відмінити команди на рівні управління, щоб можна було вирішити будь-які проблеми, викликані помилкою в сприйнятті інтелектуального транспортного засобу, що може бути вирішено.

Після того, як маршрут визначено, водій може вирішити змінити його, або через зміну пункту призначення, або через те, що водій вирішить їхати іншим маршрутом. Рішення полягає у вирішенні проблеми на двох рівнях: проблеми на рівні навігації повинні бути вирішені за допомогою спеціального інтерфейсу, який дозволяє користувачеві вибрати маршрут, який він вважає найбільш прийнятним.

Доречно, тоді як будь-які помилки, пов'язані з проблемою сприйняття навколишнього середовища або неправильним розрахунком траєкторії руху транспортного засобу, повинні бути вирішені шляхом надання користувачеві можливості перевизначення команд на рівні управління, щоб він або вона могли взяти на себе повний контроль над транспортним засобом у разі необхідності.



### 1.3 Застосування людино-машинних інтерфейсів в комп'ютерних системах керування автомобілем

Принципи людино-машинних інтерфейсів (ЛМІ) застосовні до багатьох аспектів водіння і повинні враховуватися при розробці всіх автомобільних систем, особливо ADAS. Деякі дослідження, наприклад, [16], показують негативний вплив систем допомоги водієві, що відволікають увагу, особливо на людей похилого віку.

ЛМІ для інтелектуальних транспортних засобів повинен передбачати три способи обміну інформацією між водієм і транспортним засобом: надання зворотного зв'язку водієві, отримання вхідних даних від драйвера, виконує дії самостійно.

Надання зворотного зв'язку водієві означає спілкування з ним за допомогою сенсорних каналів, коротше кажучи, це означає надсилання повідомлень, які можуть бути: візуальні ефекти, звукові, тактильні, нюх.

Кожен тип повідомлення вимагає різного часу сприйняття; на час сприйняття також впливає складність повідомлення та кількість завдань, які одержувач виконує в цей момент. Зір - це відчуття, яке актуалізується всіма типами повідомлень, адже стимул, отриманий на будь-яке з чотирьох перерахованих вище почуттів, викликає додатковий візуальний пошук, щоб зрозуміти джерело стимулу. Наприклад, поява візуального або акустичного сигналу в автомобілі змушує водія шукати джерело цього сигналу, і це робиться в першу чергу за допомогою зору, що призводить до збільшення розумового навантаження, яке виконує водій. Загалом, всі подразники, що генеруються всіма сенсорними каналами, сприяють збільшенню навантаження на водія; кожен водій також має здатність тимчасово відволікати увагу від дороги, не відволікаючись, що називається зоровим відволіканням, за умови, що зміна фокусу уваги дуже обмежена в часі. Візуальне відволікання виникає, коли зоровий стимул перевищує зорову здатність, тобто коли людина занадто довго концентрується на завданні,

відмінному від керування автомобілем, або дивиться на об'єкт, відмінний від дороги.

Що стосується акустичних повідомлень, то вони є менш складними для сприйняття, оскільки водії зазвичай виконують такі дії, як прослуховування радіо під час руху, але легко зрозуміти, що особливо складні акустичні повідомлення можуть викликати відволікання уваги водія. Одне з найскладніших для вимірювання перевантажень - це когнітивне перевантаження, яке виникає переважно тоді, коли водій відволікається, думаючи про щось інше, окрім водіння. Ще одним значним перевантаженням є ручне перевантаження, яке виникає, коли водій зайнятий другорядними діями, такими як пиття кави або налаштування радіо; міркування, зроблені для зорового перевантаження, також застосовуються до цього перевантаження в тому сенсі, що користувач має певну здатність виконувати другорядні ручні дії, не відволікаючись від завдання водіння, але якщо ці дії перевищують максимальну здатність користувача, то увага водія відволікається.

Тактильний зворотний зв'язок є частиною стандартної конструкції автомобіля, яка дозволяє керувати ним: зворотний зв'язок по зусиллю на кермі та його реакція на дії користувача є однією з особливостей дизайну автомобіля. Іншим тактильним зворотним зв'язком, хоча і в широкому сенсі, є реакція на педаль газу і "зерно", а також вібрації автомобіля, які в автомобілях класу люкс, безумовно, менші, ніж в автомобілях економ-класу, і які впливають на комфорт водіння, а отже, на вантаження, яким піддається водій. Тактильні системи зворотного зв'язку можуть бути цінними методами попередження водія, окремо або в поєднанні з візуальними чи акустичними системами. Одним із прикладів є використання вібрації на кермі для попередження водія про перетин смуги руху; ця емуляція вібраційних смуг на автомагістралі виявилася дуже ефективною і викликає мінімальне збільшення розумового навантаження.

Нюховий зворотний зв'язок наразі не використовується в людино-машинних інтерфейсах, хоча емпіричні дослідження показали, що нюхова стимуляція може підвищити ефективність тактильного зворотного зв'язку[17].

#### 1.4 Канали взаємодії людино-машинного інтерфейсу при визначенні рядності руху

Ще одним аспектом при розробці ЛМІ для інтелектуальних транспортних систем має бути розподіл, включаючи час, візуальних, слухових і ручних дій, очікуваних від водія, оскільки було показано, що водій може виконувати кілька візуальних завдань одночасно, в той час як ручні завдання є послідовними. Звідси випливає, що попередження, які інтегрують кілька сенсорних модальностей, можуть бути корисними, якщо вони не перевантажують водія; фактично, додавання тактильного зворотного зв'язку до аудіо/візуальних попереджень може підвищити ефективність останніх без надмірного впливу на психічне навантаження водія. Загалом, інтегровані попередження повинні враховувати мануальні, перцептивні та когнітивні потреби водія, і хоча не існує абсолютних правил у розробці попереджень ADAS, врахування цього може призвести до створення більш ефективних ЛМІ з точки зору часу реакції водія.

Людино-машинні інтерфейси також мають завдання виконувати дії автономно, тобто виконувати певні дії, не вимагаючи будь-якого конкретного втручання з боку користувача. Наприклад, це відбувається у випадку повністю автономного водіння, коли після введення даних користувачем транспортний засіб керує собою повністю автономно, не вимагаючи подальшого втручання користувача, або в ADAS, коли команди користувача доповнюються можливостями автоматизації транспортного засобу, як показано в попередньому розділі.



Рисунок 1.3 — Приклад людино-машинного інтерфейсу

Слід пам'ятати, що автоматизована система водіння може потребувати зворотного зв'язку з користувачем, і в ідеалі вона повинна постійно знати, в якому стані знаходиться увага водія. Стан уваги означає, скільки уваги водій приділяє основному завданню - керуванню автомобілем або перевищенню швидкості. Під час нормального водіння стан уваги проявляється насамперед у погляді водія, який повинен бути зафіксований на дорозі; відхилення від цього положення погляду може свідчити про відволікання водія, а також про відкривання/закривання очей або позіхання водія. Багато факторів, таких як втома, сонливість, розслабленість і т.д., можуть впливати на поведінку водія.

Тиша або інші зовнішні перешкоди можуть спричинити погіршення уваги водія. Під час високоавтоматизованих сеансів водіння водій може бути менш уважним під час виконання завдання, але транспортний засіб повинен знати, наскільки уважним є водій, щоб у разі потреби застосувати відповідну політику відкликання.

ЛМІ на рівні управління: як правило, елементи керування рульовим керуванням і прискоренням/гальмуванням в інтелектуальному транспортному засобі можуть бути модифіковані для виявлення наміру водія

втрутитися на рівні управління або за допомогою спеціальних кнопок на кермі, або за допомогою датчика, здатного оцінити силу, з якою водій прикладає зусилля до елементів керування. У той же час, інтегрований зворотний зв'язок і попередження, що поєднують акустичні, візуальні і тактильні повідомлення, можуть бути надіслані через НМІ для полегшення комунікації між автомобілем і водієм.

ЛМІ рівня маневрів: сенсорний екран, за допомогою якого можна вибирати між маневрами, запропонованими штучним інтелектом, або серією команд, розміщених на кермі, щоб вибрати потрібний маневр. На цьому рівні можна інтегрувати системи введення, такі як голосові команди, і використовувати дисплеї для полегшення комунікації та зменшення когнітивного навантаження на водія.

Навігаційний рівень: сенсорний екран, за допомогою якого можна вибрати бажаний маршрут або віртуально змінити маршрут, використовуючи кермо в якості покажчика напрямку, методи голосового введення і дисплеї на лобовому склі також можуть бути включені на цьому рівні для полегшення комунікації і зниження когнітивного навантаження на водія. З огляду на проблеми і конфлікти кожного рівня автоматизації, було вирішено розробити і спроектувати нову архітектуру, здатну підтримувати описані вище рівні автоматизації, і розробити нову парадигму водіння на високоавтоматизованому рівні, яку ми назвемо "Цифрове водіння", що дозволяє керувати транспортним засобом за допомогою дискретних команд на високому рівні абстракції, які називаються маневрами, без безпосередньої взаємодії зі звичайними командами водіння, але залишаючи інтелектуальний транспортний засіб під повним контролем на довгих ділянках маршруту.

Переваги такого типу водіння перед іншими рівнями автоматизації полягають у наступному: провідний у циклі "сприйняття-дія"; відсторонення водія від безпосереднього керування транспортним засобом; групування всіх ADAS в рамках одного ЛМІ; юридична доцільність проекту;

З точки зору повністю автоматизованого водіння, як видно з аналізу різних рівнів автоматизації, взаємодія між транспортним засобом і водієм відбувається на рівні навігації, як вже зазначалося в розділі 1.2; в цьому випадку водій призначає один або кілька пунктів призначення, а автоматизація транспортного засобу має завдання безпечно виконати кожен маневр, що веде від початкової точки до всіх пунктів призначення. Це означає, що автоматизована система водіння повинна приймати кожне рішення, не маючи можливості жодним чином покладатися на водія, який, відповідно до концепції повністю автоматизованого водіння, не є частиною циклу "дія-сприйняття". Отже, існує потреба в системах сприйняття і когнітивних системах, які мають дуже високу продуктивність, особливо з точки зору розуміння середовища, що оточує транспортний засіб.

Як було показано сприйняття зовнішнього світу та інтерпретація різних ситуацій під час водіння, безумовно, є завданням складніше, ніж інтелектуальна система водіння. В інтерпретації дорожніх ситуацій потенційно можливі два типи помилок: помилкові спрацьовування (також звані  $\alpha$ -помилками) і помилкові неспрацьовування (також звані  $\beta$ -помилками). У випадку помилкових спрацьовувань автоматика виявляє перешкоду там, де її насправді немає, тоді як у випадку помилкових відмов об'єкт не виявляється там, де він насправді є. Проблема при розробці архітектури системи полягає в тому, що система сприйняття, яка зменшує один з двох типів помилок, автоматично збільшує інший. Системні архітектури для автоматизованих транспортних засобів зазвичай покладаються на складні системні політики, як описано в [18], і на консервативні стратегічні рішення, що мають тенденцію допускати більше хибних спрацьовувань, ніж хибних відмов; це тому, що хибні спрацьовування за своєю суттю є менш небезпечними з точки зору безпеки для можливих пасажирів. На закінчення, стратегії автоматичного відновлення після помилкових ситуацій є дуже трудомісткими з точки зору обчислювальної складності, оскільки вони повинні оцінювати багато різних

змін з потенціалом ризику, який час від часу зростає, як показано в [19]. Цей процес прийняття рішень можна значно скоротити, ввівши водія в цикл сприйняття дії, оскільки властива будь-якому водієві здатність вирішувати, коли позитивне виявлення є істинним, а коли хибним, перевершує автоматизацію. Крім того, водій здатний швидко розпізнати і проаналізувати будь-яку, навіть складну, ситуацію на дорозі і, за необхідності, знайти альтернативні варіанти дій. Таким чином, інтеграція водія в цикл "сприйняття-дія" є однією з найбільших переваг керування на рівні маневру.

На противагу цьому, візьмемо для прикладу сучасні ADAS, які здатні керувати автомобілем без втручання людини, принаймні в деяких чітко окреслених ситуаціях, таких як рух по автомагістралі. У цьому випадку автомобіль, оснащений системою адаптивного круїз-контролю (ACC) і системою допомоги при утриманні в смузі руху (LKA), здатний здійснювати як поздовжнє, так і поперечне керування транспортним засобом, і, таким чином, вже здатний рухатися в обмеженому середовищі, наприклад, на автомагістралі. Навіть в ADAS, як вже зазначалося, при повністю автоматичному водінні може виникнути неправильна поведінка, а отже, необхідний нагляд водієм: якщо система знає про свої помилки, вона може повідомити про них водія і передати йому керування з необхідними запобіжними заходами; і навпаки, якщо ні, то саме водій повинен усвідомити помилку і відновити контроль над транспортним засобом за допомогою звичайних аналогових засобів керування. З цієї причини активація і контроль базових і вдосконалених DAS відрізняються, як показано в Таблиці 1.1.

Таблиця 1.1 — Відмінності між DAS та ADAS.

	Базова DAS	Просунута DAS
Сприйняття	Мінімальні та дуже зручні в користуванні.	Комплекс
Автоматизація	Автоматизація початкова	Довговічні та активовані користувачем
Нагляд водія	Не обов'язково	Необхідно

Зокрема, для деяких DAS автоматизація є дуже короткою, і тому вони не потребують нагляду, оскільки сприйняття є мінімальним і дуже сфокусованим. З іншого боку, ADAS доводиться мати справу навіть з дуже складними ситуаціями в автоматичному режимі, який також може бути активований протягом тривалого часу, і в цьому випадку водій повинен контролювати систему.

Аналогове керування, характерне для сучасного автомобіля, має дві основні проблеми: потреба в навчанні водіїв, вразливість людського фактору.

По-перше, водіння не є природною діяльністю, тому кожен водій повинен бути навчений керувати транспортним засобом. Як і в будь-якій діяльності, що підлягає навчанню, здібності кожної людини впливають на кінцевий результат, тому існують категорії водіїв, які є менш кваліфікованими, ніж інші, і більш схильні робити помилки та спричиняти дорожньо-транспортні пригоди, коли завдання водіння стає складнішим, наприклад, у випадку нерівного дорожнього покриття або несприятливих погодних умов. Крім того, кожен водій вразливий до особистих та навколишніх середовищ: втому, наприклад, сонливість, або емоційні стани, такі як тривога чи порушення внаслідок вживання алкоголю або наркотиків, можуть суттєво впливати на водійську діяльність і знижувати безпеку. Збільшення кількості ADAS у транспортних засобах наступного покоління створює ще одну проблему: ці системи є гетерогенними і незалежними одна від одної, що може призвести до перевантаження діяльності водія, який повинен контролювати всі ADAS, і, як наслідок, до збільшення складності завдання водія з керування. Крім того, гетерогенні та окремі людино-машинні інтерфейси призводять до значного перевантаження когнітивної діяльності, оскільки вони не враховують фізичні, когнітивні та мануальні потреби, які має кожен водій. Отже, об'єднання всіх СППР в рамках єдиної парадигми водіння на рівні маневру дає велику перевагу у вигляді спрощення людино-машинного інтерфейсу, оскільки всі інтерфейси об'єднуються в єдиний ЛМІ, що призводить до розвантаження когнітивної діяльності водія.



Як останній фактор, слід врахувати, що відповідно до Віденської конвенції про дорожній рух 1968 року [20], яка є чинною європейською конвенцією, на якій ґрунтується законодавство європейських країн, кожен транспортний засіб потребує водія, який несе відповідальність за транспортний засіб у разі аварії, навіть під час автономних сесій; отже, для того, щоб зробити високоавтоматизоване водіння юридично можливим, необхідно розглядати водія як наглядача та особу, відповідальну за автомобіль.

## 2 МАТЕМАТИЧНА МОДЕЛЬ ДЕТЕКТУВАННЯ ДОРОЖНЬОЇ РОЗМІТКИ АВТОМАТИЧНОГО КЕРУВАННЯ

### 2.1 Деталізація блоку обробки відеоінформації

Як відомо, камера системи LDW, розташована в нижній частині лобового скла (можливе також розміщення в бампері або фарі), розпізнає маркери і передає дані про своє положення (кут, відстань) до комп'ютера. Комп'ютер, базуючись на різноманітних даних і впровадженому алгоритму, приймає рішення про необхідність активувати звуковий сигнал попередження для водія [4].

Камера, що використовується в цьому комплексі, має роздільну здатність зображення 1920x1080 пікселів з CMOS сенсором і забезпечує 25 кадрів в секунду, це видно на блоку 1 блок-схеми блоку обробки відеозображення (Додаток В). Оскільки для функціонування системи потрібна певна відеопослідовність, блок 2 блок-схеми (Додаток В) відповідає за перевірку наявності наступних кадрів відеопослідовності для аналізу. Коли відеопослідовність закінчується, система завершує свою роботу (блок 20).

Коли зображення відеопослідовності (блок 2) стає доступним, система переходить до наступного етапу - блоку 4 блок-схеми, а також до блоку 9 блок-схеми блоку обробки відеозображень: нормалізація зображення.

У контексті обробки зображень, нормалізація - це процес, який змінює діапазон значень інтенсивності пікселів. Зазвичай, зображення мають низьку контрастність, наприклад, через відблиски. Іноді нормалізацію також називають розтягуванням гістограми. У більш загальних областях обробки даних, таких як цифрова обробка сигналів, її називають динамічним спектральним розширенням [16].

Нормалізація перетворює n-вимірне зображення у відтінки сірого.

$$I_{norm} = \frac{I - Min}{Max - Min} \times (NEWMAX - newMin) + newMin. \quad (2.1)$$

Зі значенням інтенсивності в діапазоні (Min, Max), на новому зображенні, інтенсивність значення знаходиться в діапазоні (newMin, NEWMAX).

Лінійна нормалізація цифрового напівтонового зображення виконується за наступною формулою:

$$I_{norm} = \frac{I - Min}{Max - Min} \times (NEWMAX - newMin) + newMin. (2.2)$$

Давайте розглянемо це на прикладі. Якщо діапазон інтенсивності зображення варіюється від 50 до 180, а ви бажаєте отримати діапазон від 0 до 255, вам потрібно відняти 50 від кожного пікселя інтенсивності, щоб отримати новий діапазон від 0 до 130. Після цього, кожен піксель інтенсивності множиться на коефіцієнт 130/255, що дає вам бажаний діапазон від 0 до 255 [17]. Цей процес повторюється для кожного пікселя інтенсивності, множачи його на 130/255, що знову дає діапазон від 0 до 255 [17].

Наступним етапом алгоритму (який відповідає блоку 5 програмної блок-схеми та блоку 8 блок-схеми обробки відеозображень) є встановлення колірної моделі дорожньої розмітки. Ця модель використовується для виявлення найбільш подібної ділянки серед кількох потенційних. Колірна модель задається змінною `self.avgRGB = [ 0.8888889, 0.99607843, 0.98823529 ]` і динамічно оновлюється в залежності від виявлених сегментів ліній дорожньої розмітки. Зауважте, що ця модель є гнучкою і може адаптуватися до змін умов освітлення та інших факторів.

Колірна модель дорожньої розмітки визначається в колірному просторі RGB. Однак, в подальшому процесі обробки, модель та весь зображення перетворюються з колірного простору RGB в колірний простір HSV. Це робиться тому, що в колірному просторі HSV колір менш залежний від яскравості, що зменшує негативний вплив тіні та автоекспозиції камери на результат відокремлення маркерів від фону [18]. Цей процес перетворення

кольору відбувається в блоку 6 блок-схеми обробки відеозображення (Додаток В) та блоку 6 блок-схеми програми (Додаток С).

Блок попередньої обробки зображень, який виконує всі ці перетворення, також включає всі ключові інструменти, необхідні для вирішення поставленої задачі. Це включає мову програмування високого рівня Python (блок 6) та допоміжні бібліотеки для підвищення продуктивності, такі як розширення мови PythonNumPy та SciPy (блоки 3, 4 структурної схеми блоку обробки відеозображень) [26, 27].

Основною бібліотекою, яка використовується для реалізації рішення поставленої задачі, є бібліотека OpenCV. Вона містить функції та алгоритми для комп'ютерного зору, обробки зображень та чисельних алгоритмів (блок 5). Ця бібліотека, написана на мові високого рівня (C/C++), включає алгоритми для інтерпретації зображень, калібрування камери за зразками, усунення оптичних спотворень, виявлення схожості, аналізу руху об'єктів, виявлення та відстеження форми об'єктів, 3D-реконструкції, сегментації об'єктів, розпізнавання жестів та інше [22].

## 2.2 Автокалібрування системи стереозображення при визначенні розмітки дороги

Процес калібрування дозволяє системі встановити кореляцію між положенням об'єктів у реальному світі та їх проекцією на зображенні. Таким чином, це важливий процес для отримання відносного положення цільового транспортного засобу щодо елементів, виявлених системою технічного зору, якими є пішоходи, інші транспортні засоби, сигнали заднього ходу та лінії дороги. Параметри калібрування та відповідність між системами відліку, пов'язаними з навколишнім світом і зображенням, визначається двома типами параметрів: еометричні спотворення можуть бути:

- радіальними: кінчик лінзи або рогівка (рис. 2.1) тангенціальними.

 Внутрішні параметри — це параметри, пов'язані з камерою плюс оптика в цілому. Фокусна відстань:

зазвичай проводиться горизонтальне і вертикальне вимірювання, оскільки лінзи зазвичай не є ідеально круглими.

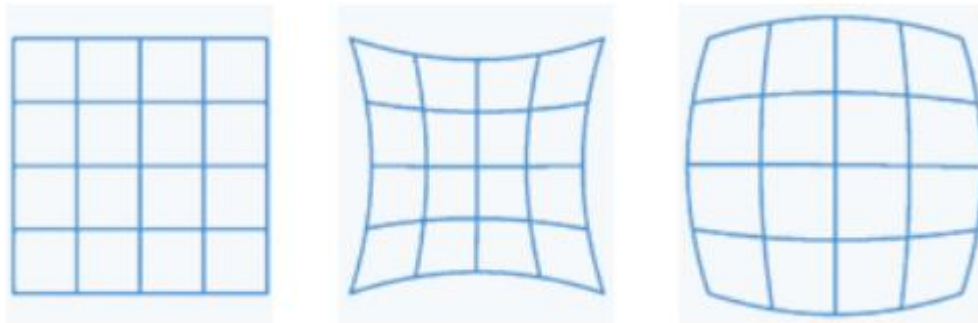


Рисунок 2.1 — Типи радіальних викривлень

Спотворення впливу лінз може вплинути на калагеометрію як основний елемент пошкодження. На рис. 2.1 показано ідеальний вхід, подушковий тип, бочкоподібний тип викривлень. Існує кілька методів калібрування внутрішніх параметрів монокулярних і бінокулярних систем. Ці параметри можна отримати в автономному режимі, без необхідності встановлення системи моніторингу в позитивному положенні. Звичайно, калібрування зовнішніх параметрів можна виконати лише за допомогою системи моніторингу, встановленої на транспортному засобі, в якому вона буде використовуватися. Це має кілька недоліків: з одного боку, велика різноманітність транспортних засобів, а отже, можливих положень і орієнтацій, які може мати приймальна система, не дає можливості використовувати універсальні значення за замовчуванням, які можна було б точно виміряти за допомогою якогось більш курсивного алгоритму. З іншого боку, потрібен видимий шаблон калібрування зсередини транспортного засобу. Більшість існуючих систем вимагають нанесення шаблонів на землю, на дах автомобіля, або на мобільний транспортний засіб.

### 2.3 Краї та їх розташування

Наступним етапом у процесі ідентифікації є використання детектора країв Canny до всієї картинки. Це сприяє зниженню впливу автоматичного налаштування експозиції камери та тіней від дорожніх структур.

Розглянемо завдання виявлення та локалізації країв. Краї - це лінії на зображенні, вздовж яких відбувається інтенсивна зміна яскравості або її похідних відносно просторових змінних. Найбільш цікаві зміни яскравості - це ті, що відображають важливі характеристики поверхні зображення. Це місця, де різко змінюється орієнтація поверхні, де один об'єкт перекриває інший, де проходить межа паралельної тіні, де відбивні властивості поверхні не є неперервними тощо. У кожному випадку необхідно знайти точки розривів яскравості або їхні похідні, щоб знати властивості об'єкта, який фотографується. Розглянемо також використання диференціальних операторів для виділення тих особливостей зображення, які допомагають знайти ділянки, де можна виявити фрагмент краю [19].

Звичайно, шум вимірювань яскравості обмежує можливість вилучення інформації про краї. Ми спостерігаємо протиріччя між чутливістю і точністю і приходимо до висновку, що короткі краї повинні мати вищий контраст, ніж довгі, щоб їх можна було виявити. Виявлення країв можна вважати розширенням сегментації зображення, оскільки краї можна використовувати для поділу зображення на області, що відповідають різним поверхням.

Інтуїтивно, край можна визначити як границю між двома регіонами, кожен з яких має приблизно однакову яскравість. Краї часто виникають на зображеннях через наявність контурних ліній об'єктів. У цьому контексті два регіони, про які йдеться, представляють собою зображення двох різних поверхонь. Краї також виникають через відсутність неперервності в орієнтації поверхні та розриви в її відбивних властивостях. Якщо взяти точку перетину функції освітленості з лінією, що проходить під прямим кутом до краю, то зазвичай спостерігається стрибок значень. Однак на практиці різниця не буде різкою через розмитість та обмеження візуального засобу.

Наступним кроком ми використаємо просту модель, щоб дослідити оператори, які можуть покращувати (загострювати) краї зображення, збільшуючи різницю в яскравості в їх околицях. Зображення з гострими краями вимагають подальшої обробки для виділення ліній та кривих. Досі більшість зусиль було зосереджено на підвищенні різкості країв, а не на виявленні та локалізації їхніх фрагментів, але останнім часом ситуація змінилася. Менше уваги приділяється об'єднанню фрагментів країв у більші одиниці, тобто лінії та криві на зображенні [21].

Однак, важливо зазначити, що виявлення країв - це лише початковий етап обробки зображень. Після виявлення країв, наступним кроком є їх аналіз та інтерпретація. Це може включати в себе визначення форми та розміру об'єктів на зображенні, визначення їх відносного положення та орієнтації, а також визначення їх відносних відстаней. Всі ці інформації можуть бути використані для подальшого аналізу зображення та визначення його семантичного змісту. Зрештою, ціль обробки зображень - це не просто виявлення країв, а отримання корисної інформації з зображення, яка може бути використана для різноманітних цілей, включаючи розпізнавання об'єктів, навігацію, відстеження об'єктів та багато іншого.

#### 2.4 Диференціальні оператори

Спрощена модель краю на зображенні представляє собою лінію, яка розділяє дві області з високим контрастом.

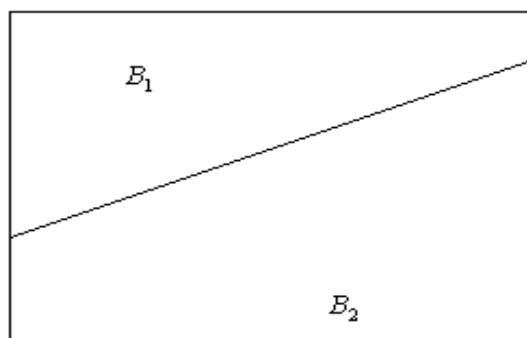


Рисунок 2.2—Ідеальний край у вигляді прямої областями яскравості

Нам потрібна функція  $i(z)$  з одиничним кроком, яка визначається як

$$i(z) = \begin{cases} 0, & z < 0 \\ 1, & z \geq 0 \end{cases} \quad (2.3)$$

Так, це слід враховувати як інтеграл від одновимірного одиничного імпульсу:

$$\delta(z) = \frac{d}{dz} i(z). \quad (2.4)$$

Визначимо що край розташовується вздовж прямої

$$x \sin \theta - y \cos \theta + \rho = 0 ; \quad (2.5)$$

Відповідно яскравість зображення приймає вигляд:

$$E(x, y) = B_1 + (B_2 - B_1) u(x \sin \theta - y \cos \theta + \rho). \quad (2.6)$$

Часткові похідні описуються таким чином:

$$\frac{dE}{dx} = \sin \theta (B_2 - B_1) \delta(x \sin \theta - y \cos \theta + \rho) ; \quad (2.7)$$

Скінченно-різницева апроксимація похідної завжди пов'язана з певною точкою. Звичайна формула кінцевих різниць дає незміщену оцінку відносно проміжної точки між опорними точками. Ці формули оцінюють частинні похідні, бо вони стосуються спільної вершини в центрі чотирьох пікселів.

Обчислення для всього зображення дають великі значення там, де яскравість швидко змінюється. В областях з постійною яскравістю результат дорівнює нулю (для шуму результат малий). Результати можна подати на новий патч з покращеними краями. Аналогічно обробляється група елементів 3x3 [19].

## 2.5 Виявлення та визначення країв

Найважливішим процесом є використання крайового детектора Канні. Його основна функція — зменшення шумів у вигляді автоматичної експозиції та тіней.

Якщо сигнал від посилення країв явно перевищує шум, можна визначити, чи належить точка до краю. Це рішення не абсолютно надійне



через можливий значний шум. Можна зменшити ймовірність помилки, вибравши поріг для мінімізації неправильно класифікованих точок.

Якщо поріг занадто високий, слабкі межі не враховуються. Існує протиріччя між типами помилок. Збільшуючи області усереднення або зменшуючи частоту, можемо зменшити вплив шуму і полегшити виділення слабких країв. Але зі збільшенням розміру областей можуть потрапляти інші ребра, тому для розпізнавання коротких ребер вони повинні бути більш контрастними.

Зображення з гострими краями показує вищі значення яскравості не тільки на краю, але й у сусідніх точках, особливо якщо зображення було згладжене. Якщо зображення без шуму, максимальні значення яскравості будуть на краю. Ці значення можна використовувати для придушення сусідніх високих значень.

Квадрат градієнта кольору як оператор створює гребінь на краю пропорційний квадрату різниці яскравостей.

Оператори Лапласа або квадратичних варіацій створюють два паралельні гребені по обидва боки краю. При використанні оператора Лапласа гребені мають протилежні знаки, і край проходить там, де змінюється знак. Розробки у виявленні країв базуються на побудові оптимальних операторів для виявлення країв і вивченні протиріч між виявленням і локалізацією. Вибір критерію оптимальності та моделі зображення відіграє роль. Модель краю як ідеальної ступінчастої функції з шумом має обмеження, і існують реалістичні альтернативи.

Нижче наведено візуальний огляд Kanni.

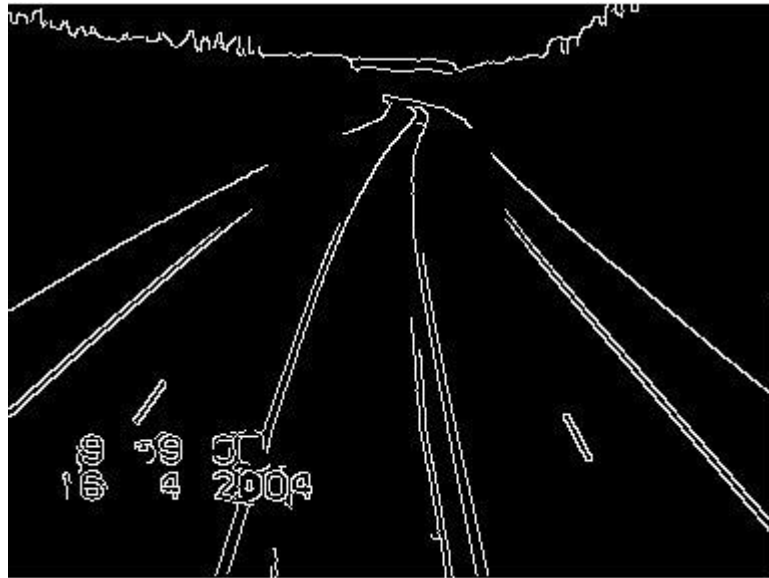


Рисунок 2.3 — Крайові зони дороги із крайовим детектором

При розробці методів виявлення країв припущення часто не застосовні в реальному світі. Хоча багато поверхонь мають постійну відбивну здатність, їхні зображення не мають рівномірної яскравості. Яскравість залежить від багатьох факторів. Елементи зображення різних об'єктів можуть мати однакові растрові значення. Лише іноді зображення можна вважати сукупністю областей з постійною яскравістю [20].

Розробка методів виявлення ребер складна, бо сама проблема недостатньо зрозуміла: як визначити, що ребро було "пропущене" або з'явилося "неправильне" ребро? Відповідь залежить від того, як ми хочемо використовувати результат.

Наведемо результати різних операторів виділення країв для ілюстрації деяких з вищезазначених моментів.



Рисунок 2.4 — Результат обробки

## 2.6 Реєстрація та обробка інформацію з датчиків

Для того, щоб ефективно дослідити стан дорожнього покриття та вірно визначити дорожню розмітку, важливо встановити конкретну робочу зону (зону інтересу). Це специфічна область, де зазвичай розташована дорожня розмітка. Ці критичні дії реалізуються у блоку 4 (створення зображення) та блоку 8 нашої блок-схеми. Наступний етап - це блок віртуальних датчиків. Кожен датчик включає в себе модель дорожньої смуги і намагається виявити її у своїй визначеній робочій області. Модель використовує дві ключові характеристики об'єкта дослідження: колір і ширину дорожньої смуги. Ширина смуги автоматично встановлюється системою шляхом аналізу результатів, отриманих на початковому зображенні, оскільки положення камери на автомобілі може значно варіюватися, що може викликати спотворення перспективи, які, в свою чергу, можуть призвести до неправильної роботи системи. Ці важливі деталі представлені в блоку 7 нашої блок-схеми (Додаток Б) та в блоку 10 Додатку В.

Наступним кроком для кожного з датчиків є виявлення всіх закритих областей (це відображено в блоку 11 нашої блок-схеми), а саме областей, які обмежені з обох боків. Для цього область пошуку зменшується до робочої області віртуальних датчиків. Експериментально було встановлено, що ефективно відкидати сегменти розміром менше 2 пікселів, щоб відфільтрувати шуми (це відображено в блоку 8 Додатку Б). Якщо в датчику знайдено декілька таких сегментів (це перевіряється в блоку 12), то обчислюється середній колір області та її відстань до заданої колірної моделі забороненої зони (це відображено в блоках 13 та 14). Серед всіх виявлених ділянок проводиться пошук ділянок, ширина яких відповідає ширині смуги розділення (це відображено в блоку 9 блок-схеми обробки зображення відеопослідовності та блоку 15 блок-схеми програми).

Після збору даних з кожного датчика, всі ці дані мають бути враховані та представлені в моделі. Для того, щоб врахувати всі дані датчиків, було використано метод найменших квадратів, який дозволяє отримати дані про центри всіх сегментів, розглядаючи віртуальні датчики як розділові смужки. В результаті роботи алгоритму була створена таблиця з певною функціональною залежністю:

Таблиця 2.1 — Таблиця залежності вхідних даних, яка відображає взаємозв'язок між різними параметрами, що впливають на роботу системи

x	x <sub>1</sub>	x <sub>2</sub>	...	x <sub>n</sub>
y	y <sub>1</sub>	y <sub>2</sub>	...	y <sub>n</sub>

Ця таблиця є важливим інструментом для аналізу та оптимізації роботи системи. Вона допомагає зрозуміти, як різні параметри впливають на загальну продуктивність системи, та дозволяє визначити оптимальні параметри для досягнення найкращих результатів.

Основна задача полягає у виявленні аналітичної форми функції, яка найкраще підходить для цієї таблиці з експериментальними даними. Цю функцію можна шукати як інтерполяційний поліном. Однак, інтерполяційні

поліноми не завжди адекватно відображають поведінку табличної функції. Більше того, значення, отримані в результаті експерименту, часто є сумнівними. У такому випадку, задача інтерполяції табличної функції втрачає свою актуальність. Тому, замість цього, шукають функцію, значення якої достатньо близькі до табличних значень. Така формула відома як емпірична формула або рівняння регресії. Емпіричні формули мають велике практичне значення, оскільки вдало підібрана емпірична формула дозволяє не тільки апроксимувати набір експериментальних даних, “згладжуючи” значення, але й екстраполювати знайдену залежність на інші інтервали. Формування моделі границі з індивідуальних даних датчиків з подальшим застосуванням методу найменших квадратів відбувається в блоці 16 блок-схеми.

Після того, як визначено інтервал між смугами руху, наступним кроком є визначення положення транспортного засобу в просторі. Для вирішення цієї задачі було обрано умовну горизонтальну лінію (це відображено в блоку 9 блок-схеми), на яку нанесені спеціальні мітки, що містять інформацію про положення транспортного засобу. Використовуючи інформацію про положення точок перетину відносно горизонтальної лінії, визначається місцезнаходження транспортного засобу, а потім ця інформація аналізується, щоб уникнути несанкціонованої зміни смуги руху. За прийняття рішень системи відповідають блок 10 блок-схеми блоку обробки відеозображень та блоки 17, 18 блок-схеми програми. Якщо система виявляє несанкціоновану зміну послідовності руху, у блоці 17 водієві видається візуальне попередження, а у блоці 19 відеопослідовність записується на носій даних для подальшого аналізу та використання.

## 2.7 Комп’ютерна система автоматичного керування

Як видно з огляду, в літературі є кілька рішень, які намагаються вирішити проблему маневрування. Рішення представлено в [26] TU Darmstad і базується на парадигмі Conduct-by-Wire: для керування транспортним

засобом використовується скінченний набір елементів, які називаються маневрами. Кожен маневр безпосередньо перетворюється на дії водія.



Рисунок 2.5 — Транспортний засіб з цифровим керуванням

Автоматизація без втручання водія в традиційні елементи керування автомобілем. Внесок в керування на рівні маневрів полягає в новому підході до високоавтоматизованого водіння, який називається "Цифрове водіння", що базується на безперервній взаємодії між людиною і транспортним засобом. Частиною цього підходу є каталог дій високого рівня, політика вирішення конфліктів між двома інтелектуальними агентами та відповідний інтерфейс ЛМІ.

Діяльність водія можна розділити на три основні ієрархічні рівні [27]: навігація, рух (маневрування) і стабілізація. На рівні навігації водій обирає відповідний маршрут для досягнення пункту призначення; це завдання зазвичай виконується на початку водіння і може повторюватися у разі виникнення проблем під час поїздки. На рівні водіння водій створює всі необхідні траєкторії для проїзду дорогами, запланованими на вищому рівні; ці дії не впливають на планування, зроблене раніше. Останній, нижчий

рівень, який називається рівнем стабілізації, стосується правильного позиціонування транспортного засобу на проїжджій частині; в кожному мить позиція уточнюється водієм за допомогою традиційних елементів керування прискоренням, гальмуванням і рульовим керуванням.

У звичайних автомобілях, як показано на водій діє на всіх трьох рівнях одночасно: спочатку планується поїздка і будь-які альтернативні маршрути (навігація), потім обирається відповідна траєкторія і швидкість для кожного маршруту (водіння або маневрування), і, нарешті, вони трансформуються в конкретні дії за допомогою рульового керування або прискорення/гальмування (стабілізація).

Метою цифрового водіння є сприяння взаємодії між водієм і транспортним засобом від рівня стабілізації до рівня маневру, що дозволяє вийти за рамки прямого управління транспортним засобом за допомогою звичайних аналітичних інтерфейсів; крім того, передбачається вбудувати цю парадигму водіння в загальну архітектуру, яка може підтримувати багато рівнів автоматизації і яка може отримати вигоду від внутрішньорівневої співпраці. Завдяки дуже невеликому набору маневрів, який буде проаналізовано нижче, кожна ситуація водіння буде добре описана одним маневром, і водієві не доведеться турбуватися про те, як ця дія буде реалізована. Під час розробки та впровадження людський фактор залишався в межах циклу "сприйняття-дія" для того, щоб реалізувати співпрацю між людиною і машиною, залишаючи водія відповідальним за діяльність транспортного засобу; це стало можливим, зокрема, завдяки створенню політики вирішення конфліктів, яка регулювала відносини між інтелектуальними агентами.

Цифрове водіння складається в основному з 3 елементів: Один із них це обгін. Маневр обгону, унікальний для всіх дорожніх ситуацій принаймні по одній смузі в кожному напрямку.

Зміна смуги руху — маневр використовується при зміні смуги руху праворуч або ліворуч у багатосмуговому середовищі в кожному напрямку,

анаприклад, автомагістралі LKS+ACC. Мета маневру "LKS+ACC" - дозволити рух на міських або позаміських дорогах з однією або кількома смугами в кожному напрямку, де наявні лінії розділення, шляхом зміни швидкості відповідно до швидкості транспортних засобів, що рухаються попереду. Таким чином, цей маневр є основним маневром, який використовується у всіх випадках дорожньої навігації, за винятком особливих ситуацій, наприклад, на автостоянці, де немає розмежувальних ліній. Перелік передумов для цього маневру складається лише з одного елемента: наявність дорожньої розмітки.

Це фактично єдиний елемент, необхідний для активації маневру; крім того, маневр не має після-умов, оскільки він не має природного завершення, а триває доти, доки існують межі смуги руху. Перелік умов виходу також складається лише з одного елемента:

2.4 Відсутність розмітки смуг руху для систем автоматичного водіння

На рис.2.6 показано маневр "LKS+ACC"; як видно, кожен світлофор розташований у постійному просторі паралельно до напрямку смуги, якою рухається транспортний засіб, зокрема, дані для кожного маневру

Правильне визначення демаркаційних ліній доріг: Демаркаційні лінії необхідні для розрахунку маневру. Вільна ліва смуга: цю інформацію надає спеціальний додаток для сприйняття (BlindSpotMonitor).

Як зазначалося вище, одночасна поява всього переліку передумов призводить до активації маневру обгону. Перелік передумов складається з наступних елементів:

Це початкове планування маневру обгону, описане вище, на кожному наступному кроці планування (у постійному часі) положення воріт, які ще потрібно проїхати, уточнюється з використанням нових даних сприйняття (зокрема, щодо положення маркерів смуги руху); нарешті, нові ворота додаються в ліву смугу руху відповідно до маневру "LKS+ACC".



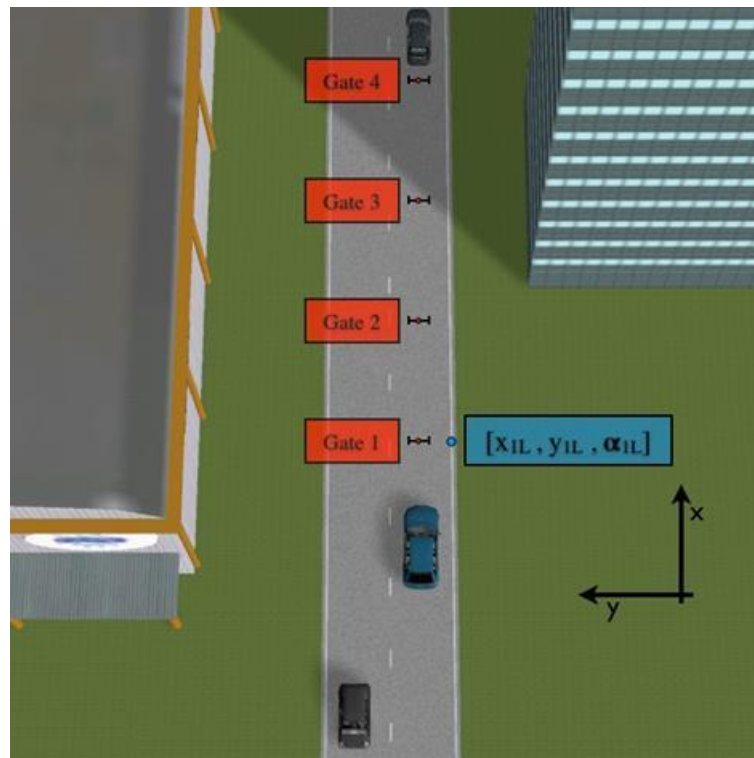


Рисунок 2.6 — Структура маневру LKS+ACC

На завершення цього розділу наведемо приклад дорожньої ситуації, щоб пояснити, як насправді працює механізм обміну маневрами. У випадку, коли ми маємо ситуацію руху на автомагістралі, в якій інтелектуальний транспортний засіб не має жодного транспортного засобу на своєму шляху, а ліва смуга зайнята. З цієї причини маневр обгону не виконується, оскільки не виконуються деякі його передумови. З цієї ж причини відключено маневр "Перестроювання в правий ряд", оскільки транспортний засіб вже перебуває в крайній правій смузі. З іншого боку, маневри з нижчим пріоритетом, починаючи з "LKS+ACC" і закінчуючи "Рухом у вільному просторі", можуть бути активовані одночасно, оскільки всі їхні умови можна перевірити; отже, список воріт, створених кожним активованим маневром, передається планувальнику на рівні управління.

Хмари точок часто вимагають великого обсягу пам'яті, оскільки є дуже великими структурами. Крім того, спосіб їх організації та великий обсяг даних призводить до великих витрат часу та коштів на обчислення. Оскільки

виконання алгоритмів допомоги водієві має відбуватися в режимі реального часу, необхідно зменшити хмару точок, не впливаючи на її просторову однорідність, що дозволить скоротити час пошуку наступних процесів, таких як виділення площини. Для виконання такого спрощення 3D-середовища був використаний метод, відомий як Voxel-метод [21], який полягає в поділі хмари точок на тривимірну сітку кубів з фіксованим розміром. Потім в межах кожного куба обчислюється центроїд всіх точок, що містяться в цьому кубі. Ця точка є спрощенням всіх точок у кубі. Виконуючи цю дію над кожним кубом, ми отримуємо спрощену хмару точок. Як видно з прикладу на рисунку 19, однорідність ПК буде залежати від розміру кубів, що використовуються для сегментації середовища. Основною метою створення цієї хмари точок є досягнення суттєвої ознаки, що дозволяє відрізнити площину дороги від решти сцени, оскільки дорога завжди є площиною з найбільшою кількістю точок або ПК. Розрахована хмара точок публікується вузлом або пакетом "point\_cloud" у темі "/stereo\_camera/points2", і за допомогою програми перегляду, такої як Rviz, її можна побачити, підписавшись на цю тему.

#### 4.1.3 ВИДІЛЕННЯ ПЛОЩИНИ ТА ОТРИМАННЯ ЗОВНІШНІХ ПАРАМЕТРІВ:

Як вказує сама назва, основною функцією цього вузла є виділення площини дороги та відокремлення її від решти для отримання вигляду з висоти пташиного польоту, який буде пояснено пізніше.

Рисунок 19: Приклад еволюції рівномірності ПК. (а) ПК без методу Вокселя; (б) ПК з середнім розміром куба Вокселя; (в) ПК з великим розміром куба Вокселя.

### 3 МОДЕЛЮВАННЯ СИСТЕМИ КОНТРОЛЮ РЯДНОСТІ РУХУ

#### 3.1 Вибір оптимального методу виявлення

Як зазначалося вище, для того, щоб розпізнати розмітку смуги руху, її необхідно спочатку відокремити від фону. Для цього експериментально було вирішено підібрати певну колірну модель смуги та проаналізувати зображення з камери як функцію різниці евклідової відстані до моделі з використанням певного порогу, що призвело до наступного результату (рис. 4.1):



Рисунок 3.1 — Відокремлення лінії від фону

На рисунку видно, що результат був досить непоганим. Однак, якщо на дорозі не було тіні, результат був наступним (рис. 3.2):



Рисунок 3.2 - Визначення лінії від фону із тіней

Однак, пізніше було виявлено найбільший недолік. При зміні яскравості зображення на вихідному зображенні з'являлося занадто багато шуму. Як уже згадувалося, було вирішено використовувати колірний простір HSV, щоб запобігти впливу яскравості на результат розпізнавання. На наступному рисунку представлено два послідовних зроблених зображення з великою різницею в яскравості, і відповідно (рис. 3.4)— результат розпізнавання обох зображень. Величина ухилів повинна бути приблизно однаковою в початковій і кінцевій точках. 2. Напрямок ухилу на кожному кінці повинен бути протилежним. 3. Довжина ділянки повинна бути близькою до ширини дорожньої розмітки. Ця ширина доріжки встановлюється відповідно до висоти віртуальної камери з висоти пташиного польоту. Таким чином, завдяки розрізненню дорожньої розмітки та фону, усуваються слабкі краї сцени. Нарешті, на третьому етапі будується маска дорожньої розмітки,  $RM$ , шляхом накладання виявлених сегментів як частини дорожньої розмітки. Крім того, щоб виділити центр дорожньої розмітки, виконується аналіз кожного виявленого сегмента, в якому передбачається, що кожен сегмент має підсегмент мінімальної величини, що

відповідає білій області дорожньої розмітки. На рисунку показано маску дорожньої розмітки (а) та виділення центральних ліній ширини смуги руху (б).



Рисунок 3.3 — Приклад автоматичної експозиції з камери

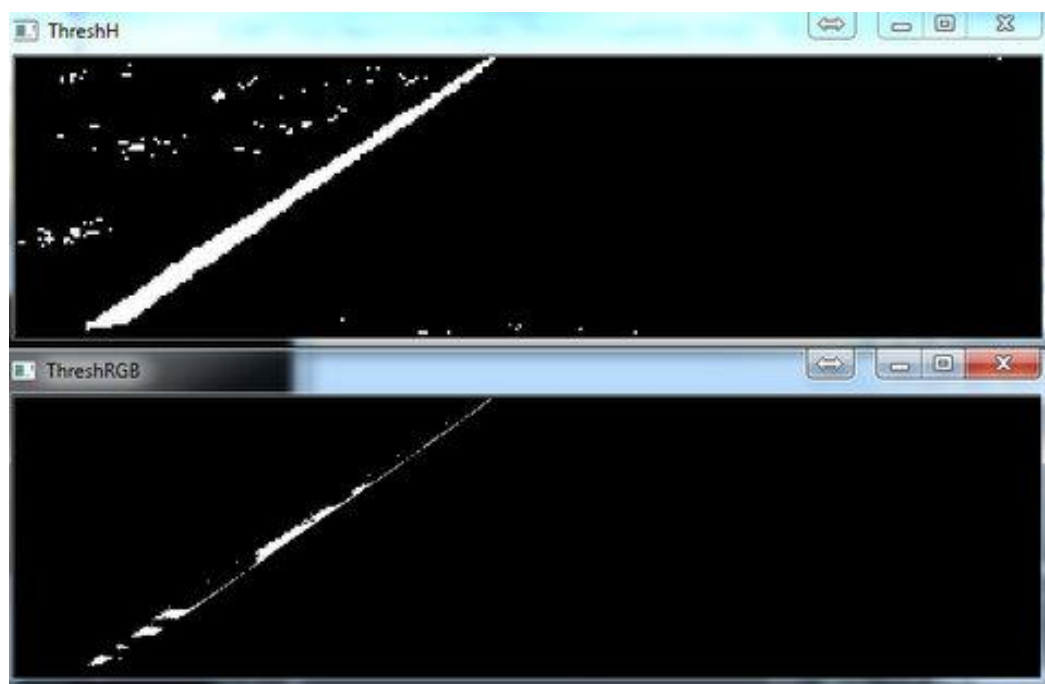


Рисунок 4.4 —Результат залежно від автоматичної експозиції камери

З наведених вище прикладів видно, що при екстремальних значеннях яскравості в оригінальному зображенні з'являється сильний шум.

Інші спроби були зроблені за допомогою різних фільтрів і методів обробки відео, включаючи популярний метод Хафа—це метод пошуку

ліній та інших простих фігур на зображенні [27]. Після побудови маски дорожньої розмітки починається процес виявлення ліній за допомогою алгоритму перетворення Хафа [8]. Як видно на рисунку, виявлено багато ліній доріг, а також інші неважливі лінії. Метод Хафа вперше було застосовано для аналізу зображень камер. Перетворення Хафа використовується для пошуку об'єктів, що підходять до певного класу форми, за допомогою процедури налаштування. З цієї множини  $X$  виділяється підмножина з більшою кількістю елементів, яка також задовольняє умову суміжності та не перетинається. Ця підмножина буде називатися структурою дороги,  $R$ . Класифікація смуг руху: Класифікація смуг руху, разом з іншими етапами, такими як виявлення ліній, є дуже важливим етапом для правильного розуміння дороги, щоб мати можливість здійснювати автономні дії, такі як обгін, зміна смуги руху або прийняття рішень в залежності від типу смуги руху.

Таким чином, знаючи тип смуги руху або сусідніх смуг, можна значно покращити такі процеси, як стеження за смугою руху або візуальна одометрія. Для початку класифікації смуг руху в класифікатор вводиться бінарний профіль  $P_i(5)$ , який витягується з маски дорожньої розмітки  $RM(4)$ . Цей профіль складається з набору точок, що належать до структури дороги,  $R$ . Ці точки визначають кінці кожної лінії, виявленої на  $RM$ . Процедура налаштування застосовується до відповідного простору параметрів, з якого отримують об'єкти певного класу форми за локальним максимумом у так званому просторі, який утворюється під час обчислення перетворення Хафа [27]. Нажаль, цей метод також не видав гарних результатів. Таким чином, було отримано наступні результати (рис. 3.5).

З рисунку добре видно, що розпізнавання смуги руху є недостатньо точним і коректним. Для цього застосовано інший метод розпізнавання розмітки.

Хоча в попередніх розділах було пояснено необхідність виконання певних дій для правильного виявлення та класифікації дорожньої розмітки,

таких як розрахунок карти дисперсії, створення ПК, вилучення плану тощо, ці процеси виконувалися в одній програмі, що не дозволяло повторно використовувати код для інших додатків ADAS.

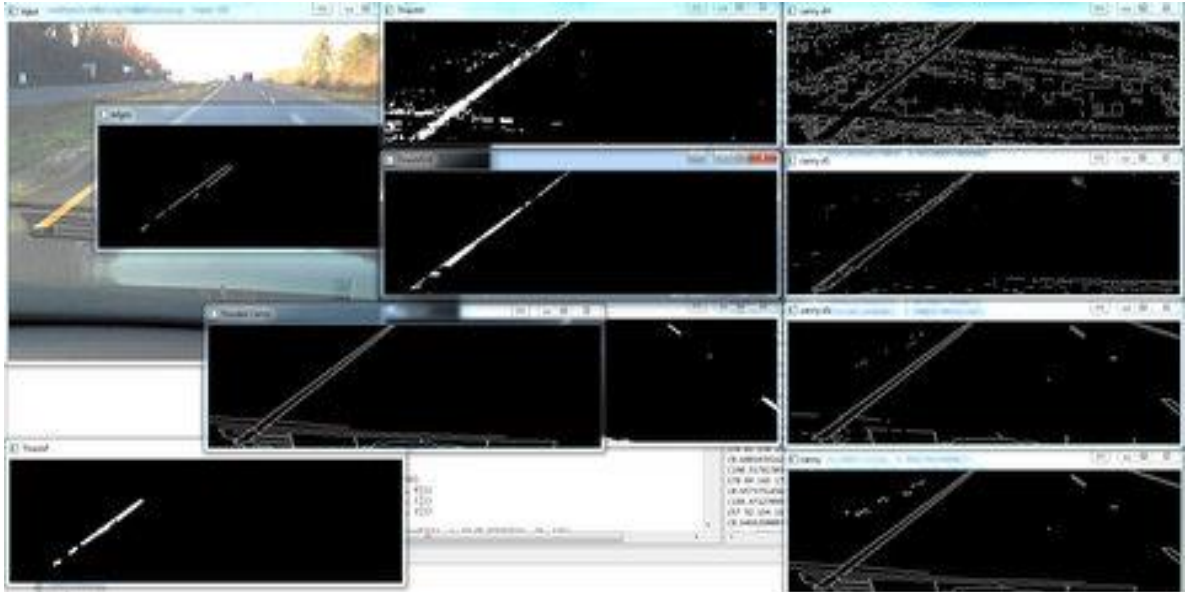


Рисунок 3.5 — Результати застосування різних методів та фільтрів

Саме тому було вирішено змінити архітектуру IVVI 2.0. У цій ЦГЗ повторно використовувалися різні пакети ROS, програма для розрахунку карти нерівномірності, інша програма для створення хмари точок, інша для вилучення та калібрування площини, одна для зміни перспективи на вигляд з висоти пташиного польоту, і, нарешті, програма з рештою методів, необхідних для виявлення та класифікації, і всі вони були інтегровані для коректного функціонування програми.з поточною конфігурацією, якщо інша система ADAS, реалізована в IVVI 2. 0 вимагає, наприклад, використання ПК або BirdView, їй потрібно буде лише підписатися на відповідну тему і зберегти ці кроки, таким чином скоротивши час розробки програми і збільшивши швидкість обчислень. Нижче наведено стислий опис тем та найсуттєвіших змін у найважливіших вузлах:Вид з висоти пташиного польоту:Після отримання виду з висоти пташиного польоту, як описано у розділі 3, буде створено нове вікно, в якому цей вид буде опубліковано у форматі Mat завдяки OpenCv, описаному у пункті 3 попереднього розділу.

Крім того, ця змінна Mat буде опублікована у темі `"/stereo_camera/bird_view"`. Під час цієї роботи цей вузол було модифіковано таким чином, щоб витяг внутрішніх змінних, необхідних для виконання коду, відбувався безпосередньо з теми `"/stereo_camera/left/camera_info"` і з подальшим покращенням коректної функціональності цього вузла для різних розмірів зображень (640x480, 1024x768 і 1280x960).

Пропонується використання "віртуальних датчиків", які дозволяються виявити не всю смугу, а лише ділянку смуги в певній обраній горизонтальній області (рис. 3.6).



Рисунок 4.6 —Вигляд"віртуальних датчиків"

Позиціонування транспортного засобу: Завдяки серії алгоритмів, розроблених в [24], можна розрахувати відносно положення автомобіля по відношенню до ширини виявленої смуги руху і, отже, знати відстань автомобіля до розділювальної лінії, щоб можна було скоригувати напрямок руху автомобіля, якщо він опиниться в ситуації з'їзду зі смуги. Як пояснювалося, можна знати кількість смуг на дорозі, якою рухається автомобіль, а також точну смугу дороги, яку займає транспортний засіб. Маючи ці дві дані, можна показати у вікні зображення виявленої дороги,



розмістивши автомобіль на конкретній смузі, і в межах цієї смуги відносно положення, яке він займає в будь-який момент часу, як показано на рисунку 2.7. Це вікно дуже корисно показувати як для моніторингу результатів, так і для відображення на приладовій панелі автомобіля, щоб запропонувати цю інформацію користувачеві транспортного засобу. Крім того, це зображення опубліковано в темі `"/lane_detection/road"`, тому його можна використовувати в подальшій роботі.

На наступному рисунку показані робочі діапазони датчиків (рис. 3.7):



Рисунок 3.7 — Віртуальні робочі області датчиків

Перевага цього методу полягає в тому, що всі датчики один від одного працюють незалежно, тому ви можете фільтрувати свої результати по відношенню один до одного.

Оскільки функціональність вузла стала можливою для різних розмірів, необхідно було оновити отримання внутрішніх параметрів камери, щоб вони також витягувалися аналогічно до попереднього вузла. Інші оновлення, зроблені в цьому вузлі. Оновлений код для коректної роботи фільтра Калмана, який використовується для згладжування стрибків значень положення транспортного засобу в смузі руху, що відбуваються між різними кадрами відео. \* Оновлення коду OpenCV3 для коректної роботи класифікатора на основі опорно-векторної машини (SVM). Додано новий

вузол, який дозволяє змінювати розмір вхідного зображення до потрібного користувачеві.

Таким чином, розмір можна вибрати, ввівши значення бажаного розміру екрану у `.launch1` з назвою `"scale_stereo.launch"`. Було зроблено й інші незначні покращення, наприклад, додано можливість вводити в програму кольорові зображення, оскільки стара реалізація не підтримувала цей формат, або використання математичних операторів, таких як синуси або косинуси, через бібліотеки, а не через бібліотеки `OpenCV`, оскільки це призводило до помилок через перехід на нову версію `OpenCV3`.

Кількість датчиків є важливим параметром для роботи системи. З одного боку, цей параметр визначає точність розпізнавання міток, а з іншого - пропускну здатність. У дослідженні було використано 70 датчиків, що відповідає швидкості 5,7 кадрів в секунду.

Це цілком достатній для розпізнавання міток в реальному часі. Слід також зазначити, що роздільна здатність відеопотоку становила 1920x1080 пікселів. Для приклада наведено графіки залежності точності розпізнавання міток та швидкості від кількості датчиків (рис.3.8) та (рис. 3.9).

Точність розпізнавання було проаналізовано під час виконання алгоритму на конкретному еталонному відеопослідовності. Проаналізувавши отримані дані, було обрано 70 датчиків як оптимальний робочий варіант системи.

Наприклад, для визначення смуги руху спочатку необхідно обрізати зображення, отримане з камери, до робочої області, яка включає діапазон видимості розмітки смуги руху.

Для того, щоб покращити перетворення Хафа, замість безпосереднього вимірювання параметрів лінії з найбільшою кількістю голосів, що відповідає максимуму накопичувача Хафа, вони обчислюються як середньозважене значення параметрів ліній, що містяться в матриці накопичувача в точці, близькій до максимуму. За основу зважування береться кількість голосів, отриманих лінією, що дещо підвищує точність параметрів виявлених ліній.

Знову ж таки, оскільки лінії практично вертикальні, акумулятор Хафа може бути нахилений під кутом, що дозволяє уникнути інтерференції з іншими об'єктами на зображенні і прискорює обчислення. Діапазон допустимого нахилу ліній - від 15 до 15 мінімальних градусів.

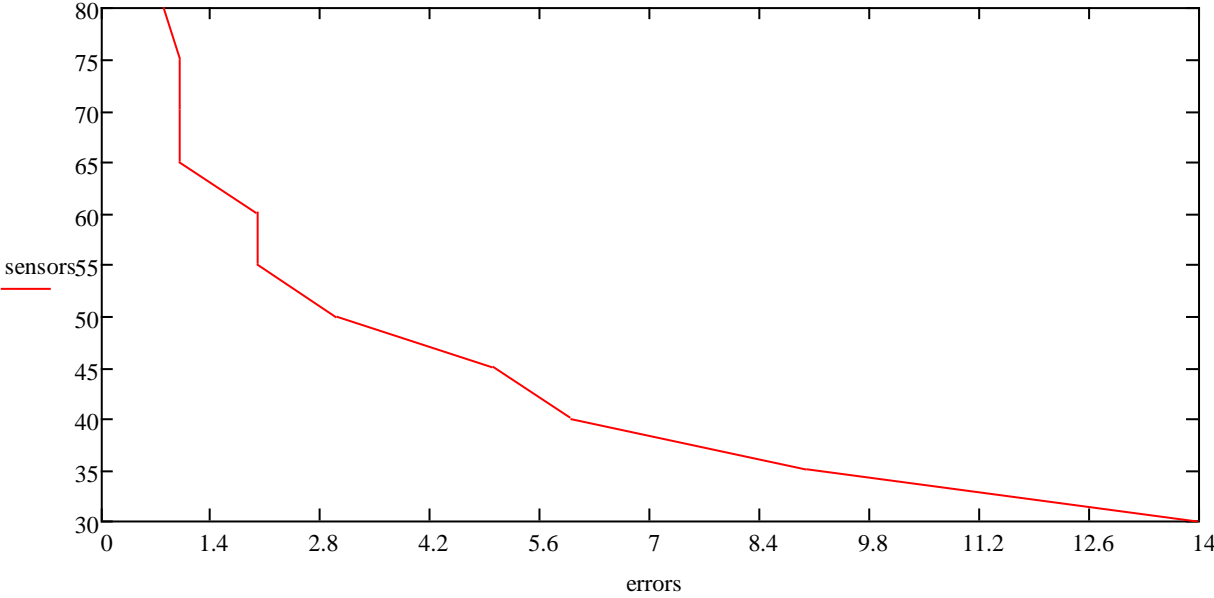


Рисунок 3.8 — Залежність кількості відмов від кількості датчиків

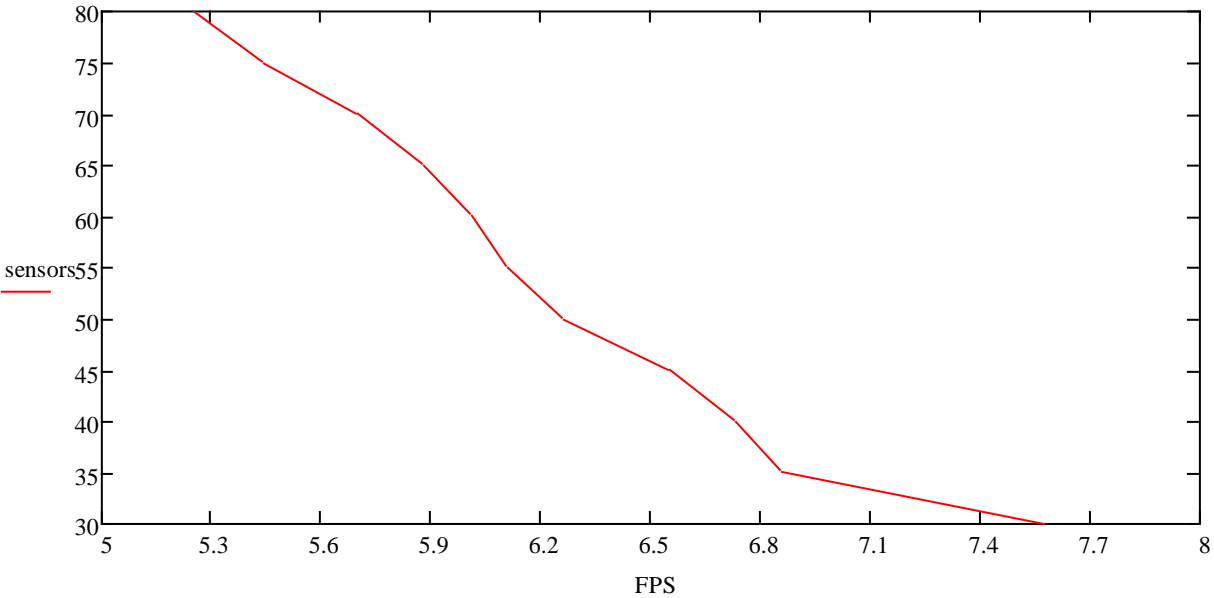


Рисунок 3.9 — Залежність продуктивності від кількості датчиків

Виявлена дорожня розмітка показана жовтим кольором, а знайдені лінії показані знову третім кольором. праворуч представлена матриця акумулятора Хафа, з максимумами каналів, обрамленими кружками.

У наведеному вище випадку виявляються обидві можливі колії, але вони не розрізняють, чи містяться вони в одній полосі, що робить коридор дуже вразливим до шуму, спричиненого структурою. Для того, щоб обмежити кількість помилкових спрацьовувань, які в основному пов'язані з тим, що використовується одна смуга, тепер використовується розмежування смуг як комбінація декількох паралельних смуг, які розділені приблизно однаковою відстанню, тобто шириною смуги. Ця модель включає до чотирьох ліній колії, для яких в акумуляторі Хафа виділяються чотири області інтересу: дві, що відповідають лівій і правій стороні, плюс дві інші відповідні області, що відповідають суміжним сторонам смуги руху (Рис. 3.10). У кожній області інтересу найкраща лінія дороги виділяється за допомогою перетворення Хафа і вважається дійсною, якщо вона набрала достатню кількість балів.<sup>2</sup> Спочатку визначаються сторони смуги, по якій рухається транспортний засіб:

У випадку, якщо смуга класифікується як безперервна лінія проникнення, тоді, і тільки тоді, відповідна область сканується на наявність сусідньої смуги. Таким чином, чутливість до структурних шумів зведена до мінімуму, оскільки в кожній області шукається тільки одна лінія. Крім того, щоб уникнути проблеми виявлення хибних спрацьовувань, які перетинають інші лінії, було реалізовано механізм націлювання на більшість виявлених ліній шляхом проектування акумулятора Хафа на лінію кута параметра (Рис. 3.9). На цій проекції обчислюється максимальне значення, яке відповідає орієнтації більшості ліній на зображенні, і навколо нього окреслюється поворот на 3 градуси, поза цим діапазоном вважається, що лінії на зображенні мають максимальне значення 3 градуси.

Це підвищує продуктивність, оскільки аналізується лише частина зображення, а не все зображення, і дозволяє уникнути помилок в алгоритмі, спричинених, наприклад, відбиттям світла від капота. У програмному коді цей параметр задається змінною  $\text{stopArea} = [300, 670, 1620, 950]$ , де

задаються координати пікселів у висоту і ширину зображення. В результаті отримуємо наступне зображення, яке буде проаналізовано далі (рис. 3.10):



Рисунок 3.10 — Робоча область для подальшого аналізу



Рисунок 3.11 — Початковезображення

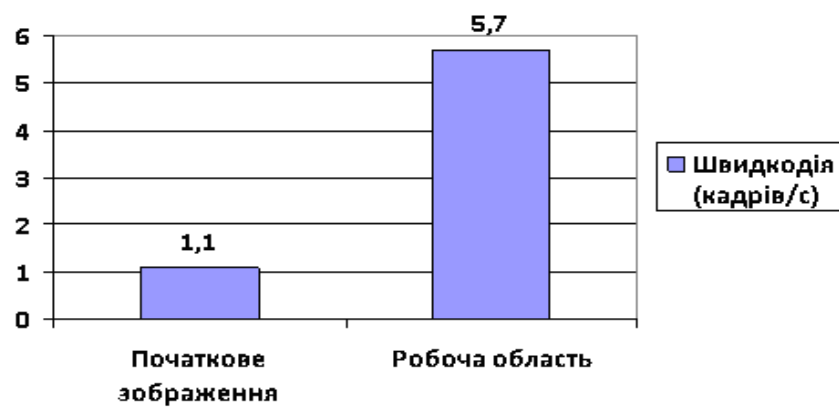


Рисунок 3.12 — Збільшення продуктивності після вибору робочого простору

Для підвищення продуктивності та точності алгоритму було впроваджено фільтр Калмана, який відстежує положення та розмір областей, що цікавлять нас, у часі. На цьому етапі створюється новий режим рейки, який складається з трьох параметрів (рис.3.12). Кут, утворений осями транспортного засобу та рейки, ширина полоси. На основі цієї моделі смуги руху будується модель дороги, яка включає розглянуті типи смуг руху (безперервна, переривчаста та така, що зливається) та кількість смуг руху, яка автоматично обчислюється на основі інформації про тип смуги руху. Слід зазначити, що, хоча кількість смуг не була детально розглянута в цьому дослідженні.

Виявлені лінії, визначені за допомогою перетворення Хафа, будуть прямокутними до ліній дороги, і залежно від видимості та стану ліній, візерунок полоси може мати тенденцію до найкращого збігу в точці, розташованій ближче або далі від транспортного засобу. У випадку, коли лінії найкраще співпадають у найвіддаленішій точці, максимальне відхилення виявлених ліній від дорожньої розмітки буде в найближчій точці, а транспортний засіб може здаватися таким, що знаходиться за межами смуги руху. Однак на зображенні, знятому камерами, кількість пікселів дорожньої розмітки, що з'являються в безпосередній близькості від автомобіля, значно вища через вплив активного пікселя. Тому, коли зображення генерується у зворотному напрямку, нижня (біля автомобіля) область є більш помітною, а також виглядає різкішою, ніж верхня. З іншого боку, коли мова йде про придорожній рух, саме найближча до транспортного засобу область представляє найбільший інтерес, оскільки саме там транспортний засіб негайно починає рух. З цієї причини було використано процедуру перетворення Хофа (HoughTransformedDevotionProcedure), так що висота випромінюваних пікселів зменшується по мірі віддалення від транспортного засобу.

Це дає швидкість просування 1112 кадрів/с. Однак, на кривих ділянках важко стежити за смугою руху. На рис. 3.13 показано еволюцію трьох

параметрів смуги руху вздовж послідовності кадрів. На кадрі 930 видно, що крива розпізнається правильно, але на кадрах 1080 і 1200 показані параметри демонструють деякі коливання, які відповідають втратам рейок на кривій і при зміні смуги руху відповідно. З одного боку, змінні не були повністю незалежними, а з іншого боку, транспортний засіб не мав власних датчиків, таких як акселерометри або GPS, тому не було можливості ввести вхідні змінні в рівняння стану.

Виявлення смуги руху за допомогою послідовності Калмана Отже, необхідно було перепроектувати цей режим, але, враховуючи, що режим ідеального репера не є проміжним етапом для розробки режиму руху по кривій, бажано було б розширити його, включивши до нього кривизну рейки. Це також вимагало переосмислення методів, що використовуються в процесі, оскільки, якщо б ми їх замінили, нам довелося б застосовувати перетворення Хафа для кіл, яке, оскільки вимагає тривимірного акумулятора, є значно повільнішим.

Наступний крок — знайти всі закриті області для кожного з датчиків, а потім відфільтрувати результати відповідно до алгоритму (білий - дорога, чорний - розділювальна лінія)(рис. 3.13).



Рисунок 3.13 — Результати визначення кожного датчика

### 3.2 Модель дорожньої смуги

Для подальшого пошуку та визначення положення лінії розмежування необхідно обрати модель, яка відображає фактичний об'єкт. Ми провели експерименти з представленням кривих другого порядку та простого відрізка прямої лінії. Висновок: представлення розділової лінії простою лінією є

досить ефективним і не набагато поступається кривим другого порядку, хоча криві є невід'ємною частиною доріг.”

Метод базується на дереві частинок, окремим випадком якого є дерево Калмана. Відстеження форм відбувається за допомогою статистичної вибірки, завдяки чому відпадає необхідність у перетворенні Хафа, і, на відміну від дерева Калмана, він допускає будь-який вид статистичного розподілу. Крім того, разом з алгоритмом автори пропонують механізм аналізу механізму автоматичного стану, що робить непотрібним використання обох аналітичних інструментів, таких як дослідження невизначеності. Оскільки пропозиція Блейкера вирішувала, принаймні теоретично, проблеми, викладені вище, було вирішено замінити фільтр Калмана на адаптовану до цього випадку реалізацію фільтра частинок, включити процедуру ідентифікації системи, а також переробити запропонований алгоритм відповідно до нового алгоритму відстежування. Проектування дорожнього полотна з урахуванням кривизни щодо планування доріг передбачає типи кривих: кругові криві, які повторюють окружну кривизну, та перехідні криві, які з'єднують попередні криві з прямими ділянками і повторюють кривизну аклотоїда. Ці криві були апроксимовані в цій трасуванні хвильовою апроксимацією, яка має простіший вираз у простіших декартових координатах. Апроксимація кругової кривої хвильовою апроксимацією наведена в Додатку

Відстеження частинок за допомогою фільтра частинок. На відміну від диференціального фільтра Калмана, так звані фільтри частинок або послідовні методи Монте-Карло не потребують мінімальних гауссових розподілів, як лінійні, і дозволяють здійснювати вибірку і відстеження багатовимірних розподілів за допомогою зменшеної кількості відліків (так званих частинок), оскільки вони мають тенденцію до центрування навколо максимумів розподілу щільності ймовірності. Крім того, фільтри частинок мають вищу здатність відновлюватися після втрат на цілі, ніж фільтр Калмана. Для відстеження смуг було обрано метод SIR



(SamplingImportanceResampling), який називається алгоритм ConDensation, запропонований М. Ізарді А. Блейком (Isardy A. Blake). ІзардіБлейком (Isardy A. Blake, 1998), і який був розроблений для повністю емпіричного відстеження складних форм у відеопослідовностях. Тералінійність була подібна до лінійності треку за допомогою регресійної моделі другого порядку .

Для врахування всіх даних датчиків ми використали метод найменших квадратів, де дані з центрів усіх сегментів, які віртуальні датчики вважають буферними смугами, слугують вхідними даними



Рисунок 3.14 — Модель лінії

Ручне визначення смуги руху і стеження, для виконання деяких простих послідовностей. Потім динамічний режим є просто регресією спостережень, отриманих у цій послідовності. Цей новий делосеом використовується для відстеження більш складної послідовності, і регресія повторюється для отримання більш точного деломоменту. Цей процес виконується кілька разів, поки не буде отримано деломомент, здатний адекватно відслідковувати об'єкти в реальних умовах. Спочатку він використовував простий одиночний режим, в якому всі змінні були незалежними, щоб рухатися по смузі руху в послідовності однієї прямої лінії, за якою один поворот ліворуч, а потім ще один праворуч, без жодних змін смуги руху. Завдяки спостереженням за цією послідовністю помилки динамічного режиму були зведені до мінімуму, а в цьому новому режимі смуга руху рухалася в послідовності, яка включала зміну смуги руху. Якщо перший режим вимагав 10 000 частинок для забезпечення збіжності, то останній навчений режим вимагає лише 1 000 частинок. Результати

Відстеження кривих смуг руху значно покращує результати, отримані при відстеженні прямих смуг, без втрати швидкості. Вся обробка виконується зі швидкістю 12 кадрів в секунду на процесорі з тактовою частотою 2,2 ГГц. Затримка в часі пов'язана із захопленням та повторним захопленням зображень. На рисунку 3.12 показано виявлення алгоритму з 2500 кадрів, включаючи лівий та правий повороти, два кільцеві перехрестя та дві зміни смуги руху, що відповідає дорожньому сполученню між містом та автомагістраллю. Хоча схема була розроблена для міських умов, ця послідовність є більш шумною, ніж добре утримувана дорога, і тому вважається хорошим тестовим прикладом. На графіку показано еволюцію чотирьох параметрів, що залишилися в хронометрі, та отриманий бал. Прогресія параметра під великим кутом пояснює вібрацію камери. Стереосистема кріпиться до автомобіля за допомогою висувного кронштейна до присоски, прикріпленої до лобового скла, як показано на рисунку 4.31. Спосіб кріплення висувного кронштейна призводить до передачі вібрацій на камери, особливо в горизонтальному напрямку. Основним місцем відмови в цих наслідках є гіроскопи, що відповідає двом заштрихованим ділянкам. Цей алгоритм розроблений для роботи на магістральній мережі доріг, тому на кільцевих розв'язках кривизна дорожніх ліній перевищує обмеження, накладені на параметри смуги руху, так що оцінка не може бути коректноспівставлена зі спостереженнями.

### 3.3 Визначення транспортного засобу в просторі

Після визначення розділювачів смуг руху необхідно визначити місцезнаходження автомобіля відносно цих смуг. Для цього ми використовуємо умовну горизонтальну лінію (блок 8 на блок-схемі програми), яка має спеціальні позначки, що перетинають лінії смуг руху.

За допомогою інформації про положення точок перетину відносно горизонтальної лінії ми визначаємо місцезнаходження транспортного засобу та аналізуємо можливість несанкціонованого виїзду зі смуги.

Існують три рівні небезпеки: “зелена зона” вказує на знаходження транспортного засобу на своїй смузі, помаранчевий індикатор показує, що транспортний засіб збирається виїхати зі смуги, а червоне світло вказує на несанкціонований виїзд зі смуги

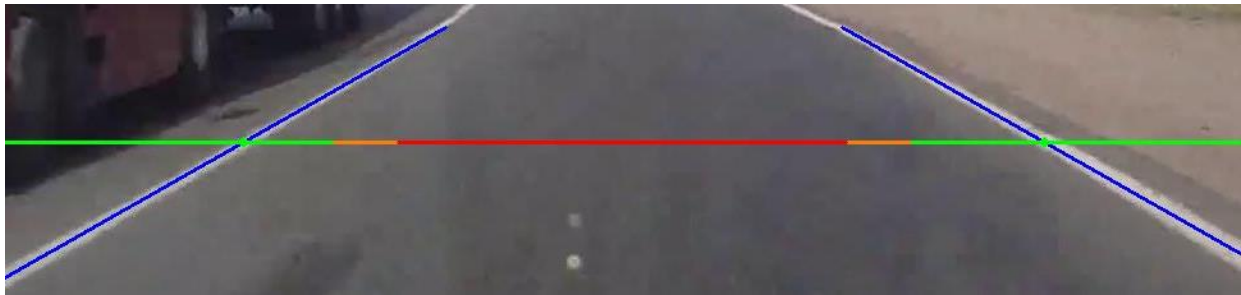


Рисунок 3.15 — Метод визначення місцезнаходження транспортного засобу

Ця система була розроблена з метою зменшення кількості ДТП, які виникають через незаконну зміну смуги руху. Вона досить ефективна при різних сигнальних лініях та погодних умовах.

Також вона має візуальну індикацію виїзду зі смуги руху (див. рис. 3.16) та записує факт незаконної зміни смуги руху. Принцип її роботи наступний: коли транспортний засіб перетинає “червону зону”, програмне забезпечення записує відеоряд на жорсткий диск (блок 11 у Додатку Б та блок 17 у Додатку В). Якщо станеться ДТП, ця функція може бути використана для подальшого аналізу.

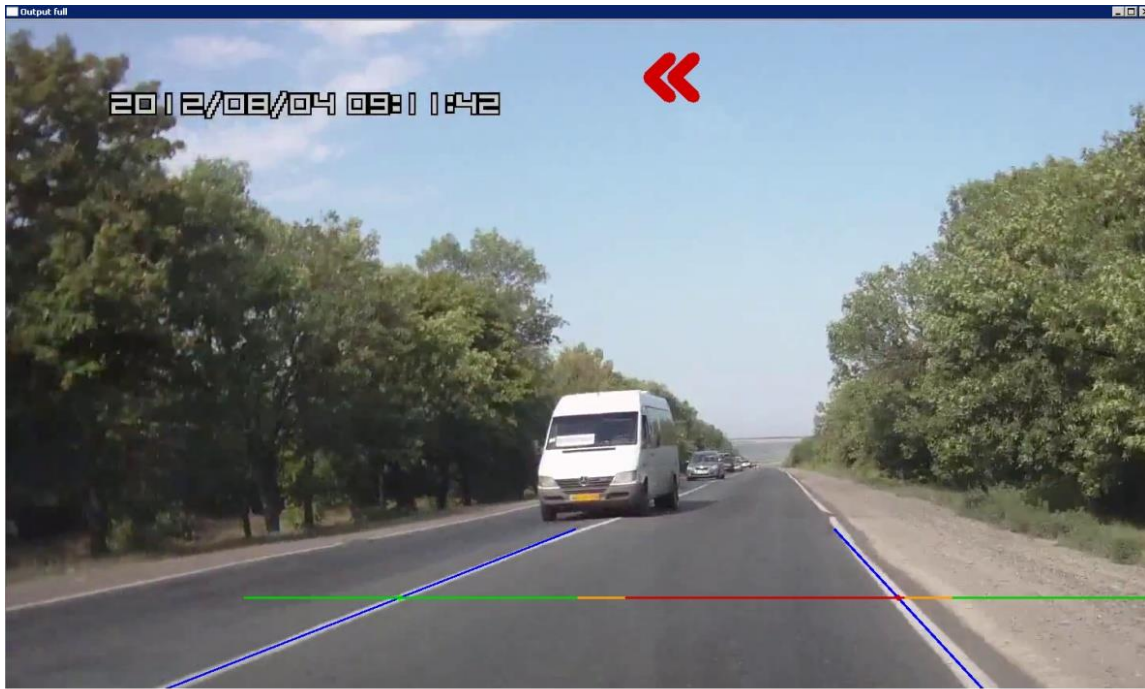


Рисунок 3.16 — Відображення зміни смуги руху

### 3.4 Властивості вибраної CCD-матриці

На сьогоднішній день компанія SonySemiconductor є одним з найбільших виробників CCD та CMOS сенсорів зображення. Давайте розглянемо будову, параметри та характеристики ПЗС-матриці SonySemiconductor, яка використовується в дисертаційній роботі. Прототипи цієї матриці широко застосовуються в серійних відеокамерах компанії.

Мікросхема ICX227AK (DD2 у Додатку Е) є модифікацією попередньої кольорової матриці ICX207AK з мозаїчним фільтром типу E, Su, Mg, G та електронним затвором. Попереднім прототипом була ICX087AK. Цей чіп був створений за технологією Super HAD CCD, яка є зареєстрованою торговою маркою компанії Sony. Відмінність полягає в тому, що проміжки між мікролінзами, розташованими над повністю чутливими елементами матриці, були значно зменшені. Це збільшує світловий потік, що падає на кожен елемент, і загальну чутливість матриці [28]. Ця камера має електронний затвор і забезпечує хороший захист від синього світла. Електронний затвор був розроблений для досягнення експозиції, що змінюється в часі, на ПЗС-матрицях з чересстрочною розгорткою.

Метод полягає в тому, що на підкладку матриці подаються групи імпульсів різної тривалості, що забезпечує швидкий і одночасний розряд усіх комірок матриці. При цьому кожен ряд ділиться на дві групи зон, одну для нормальної роботи, а іншу для швидкого розряду. Змінюючи співвідношення тривалості цих ділянок, час експозиції змінюється пропорційно.

Використання електронного затвора дає можливість записувати події, що швидко рухаються, але водночас знижує чутливість сенсорів. Розмиття зображення - це специфічне спотворення, яке виникає при використанні ПЗЗ з міжрядковою розгорткою. Спотворення з'являється на зображенні у вигляді вертикальних шлейфів, що слідує за яскраво освітленими або блискучими об'єктами. Спотворення видно лише тоді, коли значення експозиції значно перевищує нормальне значення. Антисатуманення— це метод боротьби з явищем оптичного цвітіння, яке викликане переповненням комірок ПЗЗ і перетіканням зарядів у сусідні комірки. Це проявляється на зображенні у вигляді вертикального розмиття яскравих об'єктів [29, 30].

Параметри мікросхеми. Діагональ — 4,5 мм (1/4 дюйма). Загальна кількість пікселів — 537 (H), 597 (V). Загалом близько 320 тис. пікселів. - Розміри чіпа складають 4,34 x 3,69 мм. Розміри пікселів - 7,3 x 4,7 мкм [28]. - Межі затемнення (положення оптичного чорного) становлять 7 пікселів по горизонталі з лівого боку, 30 пікселів з правого боку і по вертикалі 14 пікселів знизу і 1 піксель зверху (рис. 4.17).

Структуру та розташування виводів мікросхеми ICX207AK показано на рисунку 4.18, а розташування виводів наведено у Додатку Е: - 1-4 (Vφ4-Vφ1) - входи тактових імпульсів перенесення вертикального регістра; - 6 - корпус; - 7 (Vout) - аналоговий вихід; - 8 (Vdd) - напруга живлення; - 9 - корпус; - 10 (SUB) - вхід тактового сигналу зміщення підкладки; - 11 (PL) - вхід зсуву транзистора захисту; - 12 (RG) - вхід строб-імпульсів; - 13, 14 (HØ1, HØ2) - вход тактових імпульсів перенесення горизонтального регістра [29, 30].

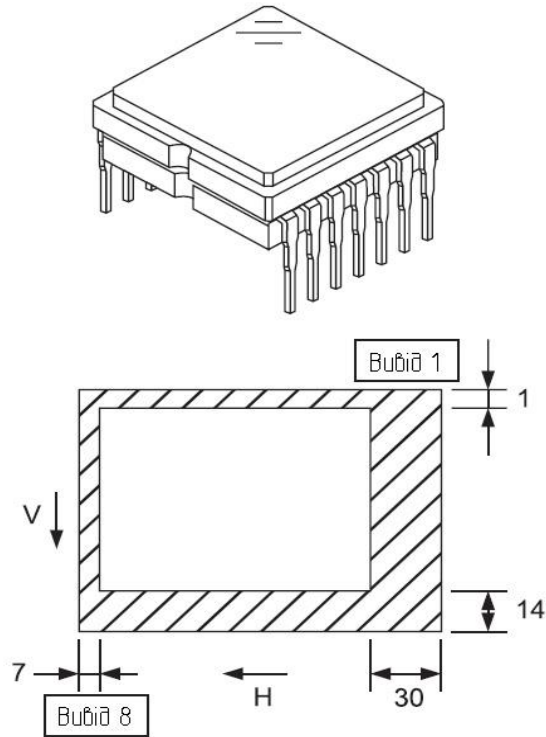


Рисунок 3.17 — Вигляд світлочутливого вікна мікросхеми

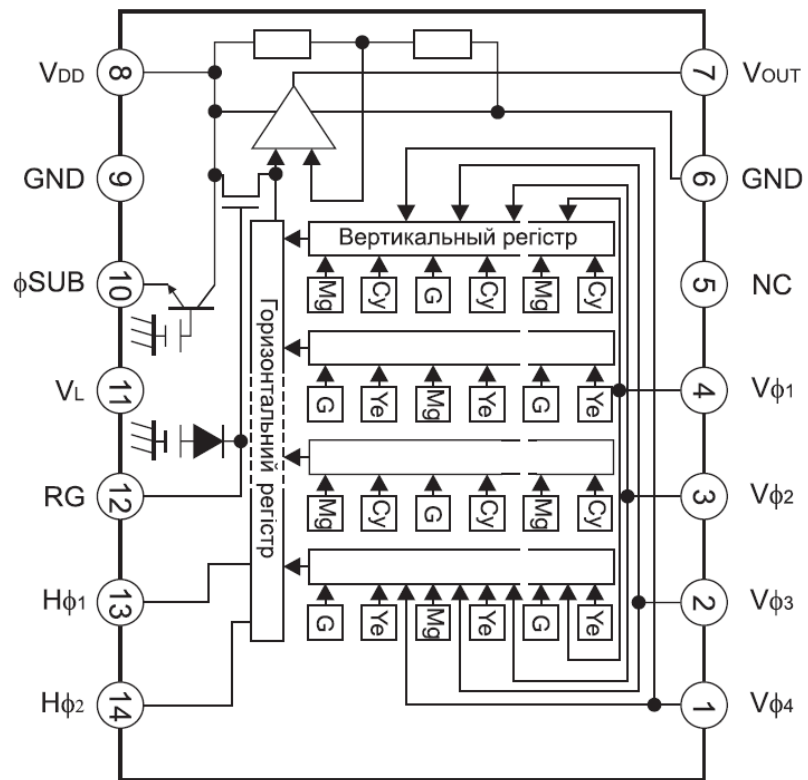


Рисунок 3.18 — Структура ICX207AK та призначення виводів

## 5 РЕАЛІЗАЦІЯ СЕРЕДОВИЩА ВИКОНАННЯ ПРОГРАМИ

### 5.1 Програмні складові

Для комп'ютерної імплементації обраного алгоритму було прийнято рішення використати високорівневу об'єктно-орієнтовану мову програмування Python. Python надає підтримку модулів та пакетів модулів, що значно спрощує процес модульності та повторного використання коду. Важливо відзначити, що оскільки Python є інтерпретованою мовою, математичні алгоритми, як правило, працюють значно повільніше на Python, ніж на компільованих мовах, таких як C або навіть Java. З метою підвищення продуктивності ми використовуємо розширення PythonNumPy [28]. NumPy намагається вирішити проблему продуктивності для великої кількості обчислювальних алгоритмів, надаючи підтримку багатовимірних масивів та багатьох функцій і операторів для роботи з ними. Таким чином, будь-який алгоритм, який можна виразити в першу чергу як послідовність операцій над масивами і матрицями, працює так само швидко, як і еквівалентний код, написаний на мові C.

Використання бібліотеки SciPy також було включено в наш процес. Це бібліотека високоякісних наукових інструментів з відкритим вихідним кодом для мови програмування Python. SciPy містить модулі для оптимізації, інтегрування, спеціальних функцій, обробки сигналів, обробки зображень, генетичних алгоритмів, розв'язання звичайних диференціальних рівнянь та інших задач, що часто вирішуються в науці та інженерії [39].

Основні модулі бібліотеки OpenCV включають:

Core: Цей модуль містить основні функції та класи, які використовуються іншими модулями. Він включає такі елементи, як базові структури даних, алгоритми лінійної алгебри, математичні трансформації та інше.

Imgproc: цей модуль включає функції для обробки зображень, такі як фільтрація, геометричні перетворення, кольорне перетворення, гістограми та інше.

**Highgui:** цей модуль містить функції для створення простих графічних інтерфейсів, введення та виведення зображень та відео.

**Video:** цей модуль включає алгоритми для аналізу та обробки відео, такі як відслідковування об'єктів, виявлення руху, виявлення змін сцени та інше.

**Calib3d:** модуль включає алгоритми для калібрування камери, виявлення шахових дошок, відновлення 3D сцен та інше.

**Features2d:** модуль включає алгоритми для виявлення та опису особливостей зображень, відповідності між зображеннями, відслідковування особливостей та інше.

**Objdetect:** цей модуль включає алгоритми для виявлення об'єктів на зображеннях, такі як обличчя, очі, пішоходи та інше.

**Dnn:** модуль включає алгоритми для підтримки глибокого навчання, включаючи підтримку різних фреймворків глибокого навчання та предтренованих моделей.

**ML:** модуль включає алгоритми машинного навчання, такі як k-найближчих сусідів, наївний Баєс, дерева рішень, SVM, нейронні мережі та інше.

Ці модулі разом забезпечують потужний набір інструментів для розробки програмного забезпечення, що використовує комп'ютерний зір для розуміння та обробки зображень та відео. OpenCV продовжує бути однією з найпопулярніших бібліотек для розробки програмного забезпечення в області комп'ютерного зору

**Sxcore** — центральне ядро. Включає в себе базові структури даних і алгоритми: початкові операції над багатовимірними числовими масивами, матрична алгебра, математичні функції, генератори випадкових чисел, запис/відновлення структур даних в/з XML, початкові функції 2D графіки.

**CV** — це модуль для обробки зображень та застосування комп'ютерного зору: початкові операції над зображеннями (фільтрація, геометричні перетворення, кольірних просторів і т.д.), аналіз зображень (вибір



ознак, пошук контурів, морфологія, гістограми). Аналіз руху та стеження за об'єктами, виявлення об'єктів.

Highgui — модуль введення/виведення для відео зображень, створення інтерфейсу користувача. Виконується захоплення відео з камер і з відео файлів. А також запис статичних зображень, функції для організації простого UI.

Svauх — експериментальні функції.

CvCam — захоплення відео, а саме дозволяє здійснювати захоплення відео з цифрових відеокамер. З опису модулів бібліотеки OpenCV впливає широкі можливості для реалізації поставленої задачі, а саме організації розпізнавання на відео.

## 4.2 Інструкція користувача

Оскільки програмний продукт був спланований як модульний, він вимагав використання безкоштовного інтегрованого модульного середовища розробки програмного забезпечення, яким є Eclipse. Це середовище, написане переважно на мові Java, може бути використане для створення програмних продуктів на Java, а також, за допомогою різноманітних плагінів, на інших мовах програмування, таких як Ada, C, C++, Perl, PHP, Python та інші.

Для середовища Eclipse доступна велика кількість безкоштовних та комерційних модулів. Початково середовище було розроблено спеціально для Java, але сьогодні існує велика кількість розширень для підтримки інших мов програмування. У нашому випадку ми будемо використовувати модуль Python - Pydev, який є середовищем розробки Python.

Для запуску програми необхідно встановити інтегроване середовище розробки Python 2.7.3 та середовище розробки програмного забезпечення Eclipse. Щоб мати можливість програмувати на Python, необхідно встановити модуль Pydev. Його можна встановити наступним чином: у середовищі Eclipse потрібно викликати діалогове вікно “Довідка” та натиснути на кнопку

“Встановити нове програмне забезпечення...” (рис. 4.1).

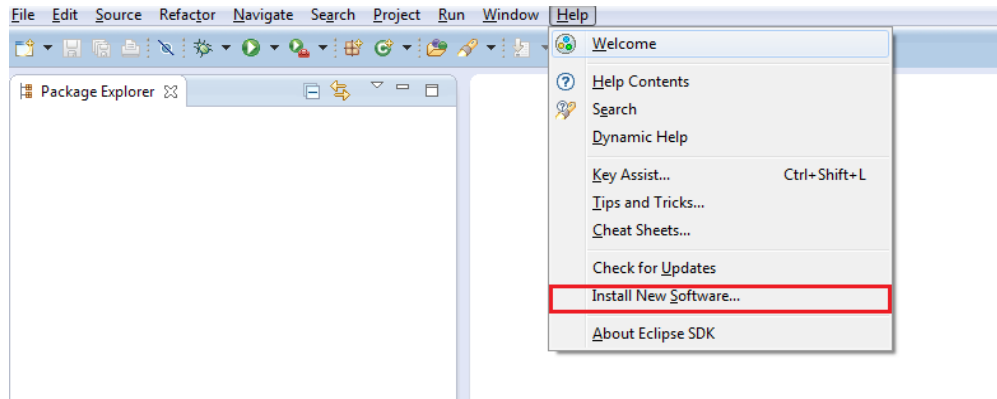


Рисунок 4.1 — Діалогове вікно "Довідка"

У наступному діалоговому вікні натисніть кнопку "Додати" (рис.4.2).

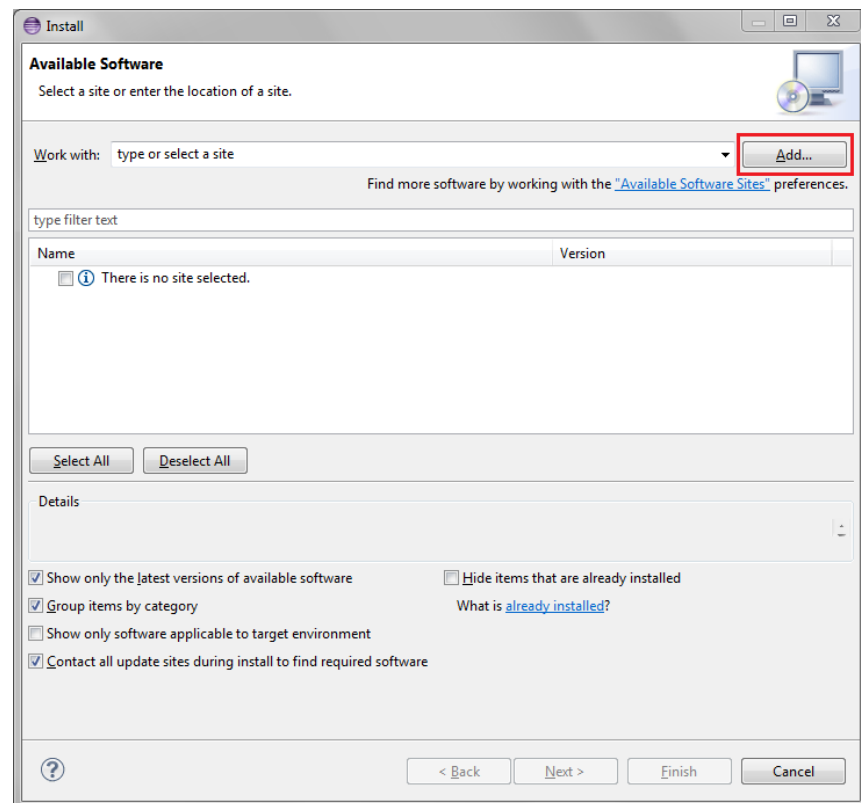


Рисунок 4.2 — Діалогове вікно встановлення нового програмного забезпечення

Тепер у полі "Name:" потрібно ввести PyDev, а в полі "Location:" - адресу завантаження: "http://pydev.org/updates" і натиснути "ОК" (рис. 4.3).

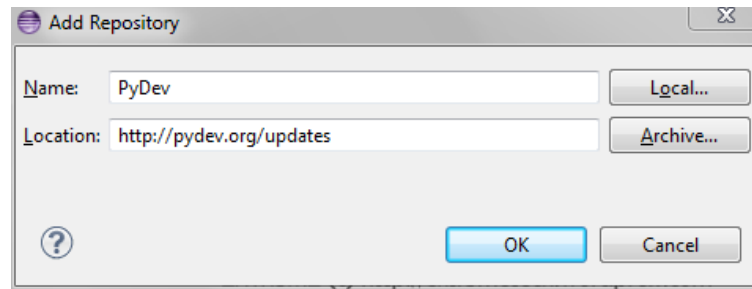


Рисунок 4.3 — Вікно завантаження додаткових бібліотек

Через деякий час ми побачимо вікно, в якому потрібно вибрати елемент "PyDev", щоб завантажити його (рис. 4.4).

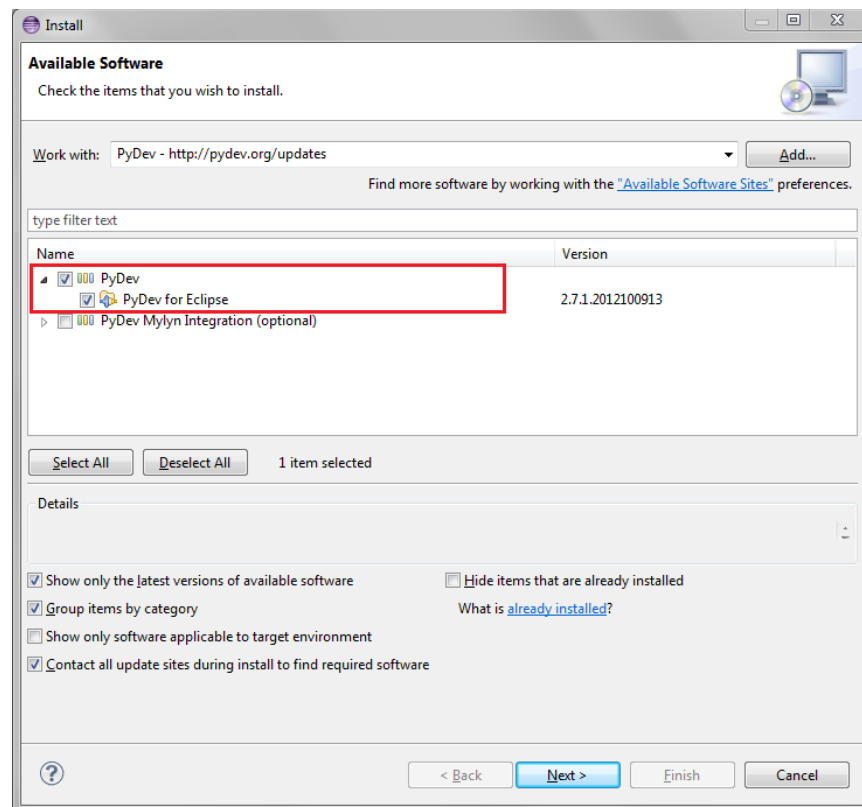


Рисунок 4.4 — Діалогове вікно вибору елементів для завантаження

Після завантаження перейдіть до меню "Вікно" середовища Eclipse і виберіть "Налаштування", потім перейдіть до каталогу Pydev>Interpreter-Python і натисніть на "Новий..." (рис. 4.5).

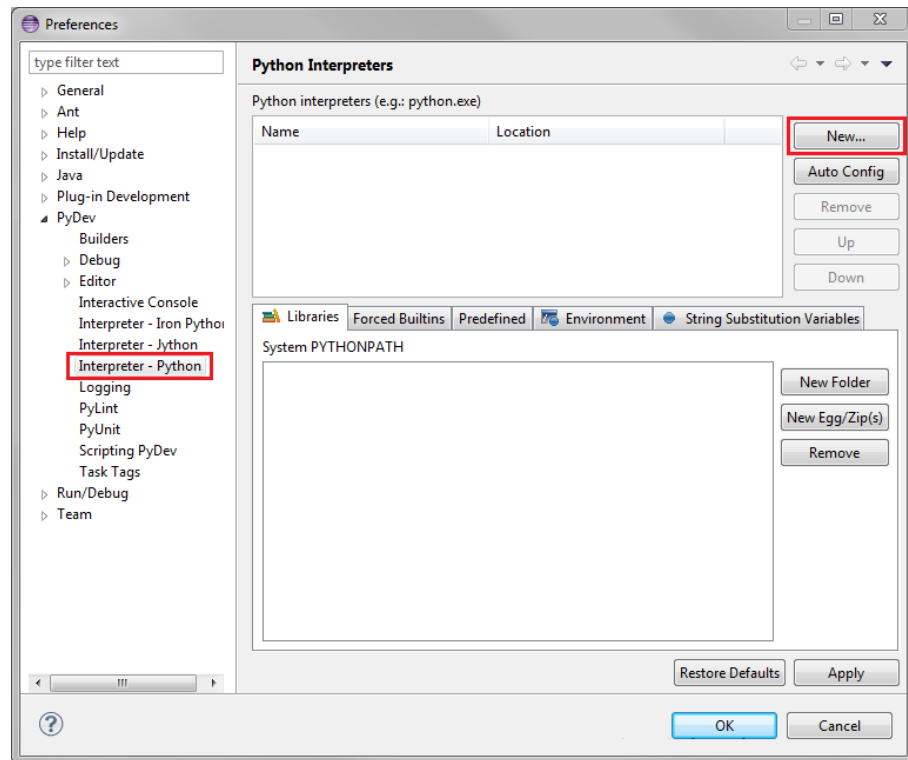


Рисунок 4.5 — Вікно встановлення

У вікні вибору інтерпретатора у полі "Ім'я інтерпретатора:" введіть "Python27", а у полі "Виконавчий файл інтерпретатора:" введіть, куди було встановлено середовище Python, і виберіть його виконавчий файл "python.exe", після чого натисніть "ОК" (Рисунок 4.6).

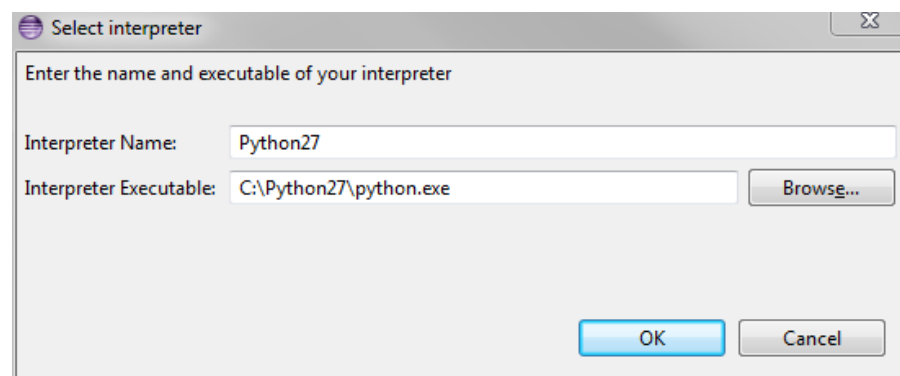


Рисунок 4.6 — Ім'я інтерпретатора

Вам буде запропоновано вибрати компоненти, які буде додано до середовища Eclipse. Виберіть наступні компоненти (Рис. 4.7):

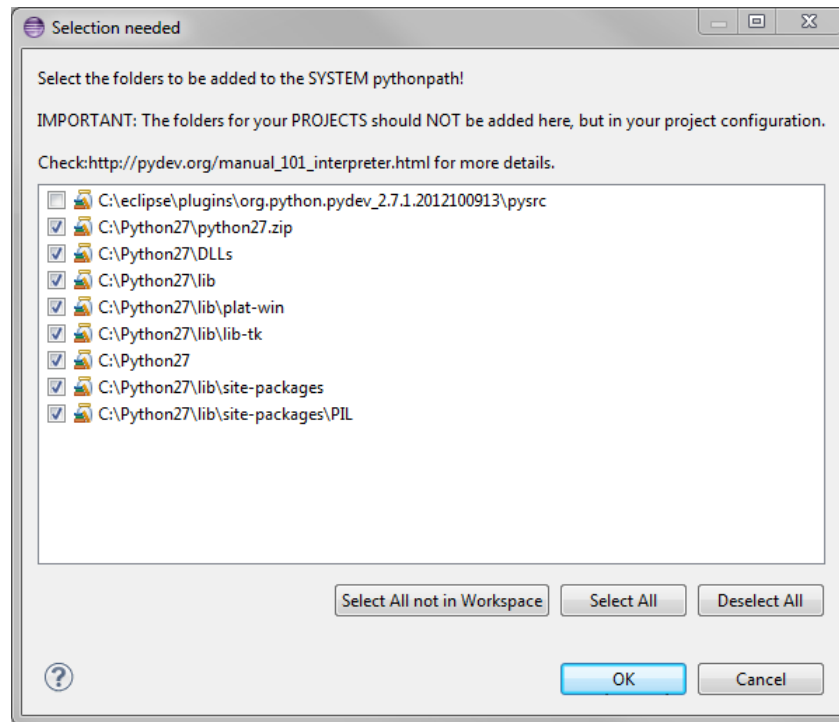


Рисунок 4.7 — Вікно вибору компонентів для використання

Після вибору компонентів збережіть зміни. Тепер середовище Eclipse готове до програмування на Python.

Як зазначалося вище, основним компонентом для розв'язання поставленої задачі є бібліотека функцій та алгоритмів для комп'ютерного зору, обробки зображень та чисельних алгоритмів OpenCV. У цьому прототипі системи LDW ми використали OpenCV версії 2.3.1, а також бібліотеки NumPy 1.6.1 та SciPy 0.9.0 [22]. Для роботи з цими бібліотеками необхідно спочатку встановити їх на комп'ютер, а потім скопіювати всі файли з каталогу "C:\opencv\build\python\2.7" до каталогу "C:\Python27\LibSite-packages".

Тепер у нас є всі необхідні інструменти для запуску та роботи з прототипом. Додаток С містить лістинг додатку. Для його запуску нам потрібно створити відповідні модулі додатку, які буде описано в наступному розділі, а потім імпортувати каталог з модулями в середовище Eclipse (Файл>Імпорт>Існуючі проекти в робочу область) (Рисунок 4.8).

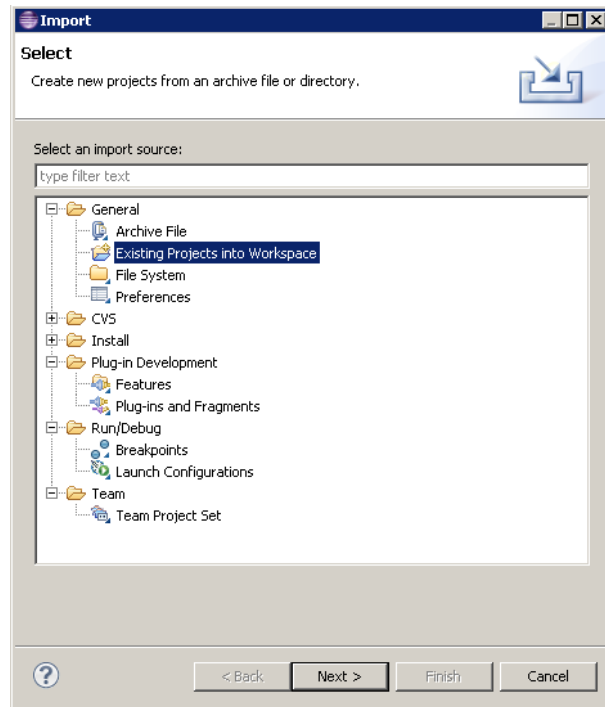


Рисунок 4.8 — Вікно вибору методу завантаження завершених проектів

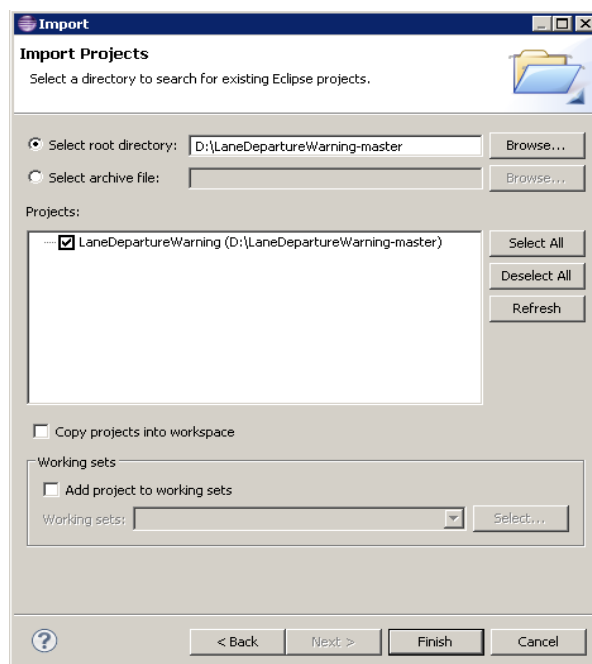


Рисунок 4.9 — Вікно вибору проекту в каталозі

Тому для продовження роботи необхідно в модулі "Відеопроцесор" вказати джерело вхідного відеопослідовності (камера або відеофайл). Ця операція виконується командою "cv.VideoCapture".

## 5 ЕКОНОМІЧНА ЧАСТИНА

### 5.1 Комерційний та технологічний аудит науково-технічної розробки

Метою даного розділу є проведення технологічного аудиту, в даному випадку нової системи реєстрації та розпізнавання на пряму руху за допомогою комп'ютерного зору.

Особливістю розробки є розширення функціональних можливостей методу детектування дорожньої розмітки та вдосконалено алгоритми роботи комплексу, що дозволило покращити структурні та функціональні схеми, апаратні і програмні засоби системи для розпізнавання дорожньої розмітки з допомогою комп'ютерного зору та поліпшення якості і безпеки життя.

Аналогом може бути Система LDWSECU 10000\$ або 400000 грн.

Для проведення комерційного та технологічного аудиту залучають не менше 3-х незалежних експертів. Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п'яти бальної системи оцінювання за 12-ма критеріями, у відповідності із табл. 5.1.

Таблиця 5.1

— Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Бали (за 5-ти бальною шкалою)					
Р	0	1	2	3	4
Технічна здійсненність концепції					
	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевіреною об'єктом з даної продукту

Продовження табл. 5.1

Ринкові переваги					
2	Багатоаналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінама	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненість					
	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значно гроші та час на навчання наявних фахівців	Необхідно значно навчання фахівців та збільшення штату	Необхідно значно навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
1	Необхідно розробити нові матеріали	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні дешеві та недорогі матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві



Продовження табл. 5.1

1	Термін реалізації ідеї більше ніж за 10 років	Термін реалізації ідеї більше ніж за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій
2	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, щовимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Усі дані по кожному параметру занесено в таблиці 5.2

Таблиця 5.2 — Результати оцінювання комерційного потенціалу розробки

Критерії оцінювання	ПБ експертів		
	Експерт 1	Експерт 2	Експерт 3
	Бали		
Технічна здійсненність концепції	3	4	4
Наявність аналогів на ринку	3	4	3
Цінова політика	4	4	4
Технічні та споживчі властивості виробу	4	4	3
Експлуатаційні витрати	4	3	4
Ринок збуту	4	4	3
Конкурентоспроможність	3	3	4
Фахівці з технічної і комерційної реалізації	4	4	3
Фінансування	4	3	4
Матеріально-технічна база	3	3	3
Термін реалізації ідеї	4	4	4
Супровідна документація	4	4	4
Сума	44	44	43
Середньоарифметична сума балів	$(44+44+43) / 3 = 43,67$		

За даними таблиці 5.2 можна зробити висновок щодо рівня комерційного потенціалу даної розробки. Для цього доцільно скористатися рекомендаціями, аведеними в таблиці 5.3.

Таблиця 5.3 — Рівні комерційного потенціалу розробки

Середньоарифметична сума балів, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 - 10	Низький
11 - 20	Нижче середнього
21 - 30	Середній
31 - 40	Вище середнього
41 - 48	Високий

Як видно з таблиці, рівень комерційного потенціалу розроблюваного нового програмного продукту є високим, що досягається зарахунок розширення функціональних можливостей методу детектування дорожньої розмітки автосконалено алгоритми роботического комплексу, що дозволило покращити структурні та функціональні схеми, апаратні і програмні засоби системи для розпізнавання дорожньої розмітки з допомогою комп'ютерного зору та поліпшення якості і безпеки життя.

5.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи

Основна заробітна плата розробників, яка розраховується за формулою:

$$Z_o = \frac{M}{T_p} \cdot t, \quad (5.1)$$

де  $M$  — місячний посадовий оклад конкретного розробника (дослідника), грн.;

$T_p$  — число робочих днів за місяць, 23 днів;

$t$  — число днів роботи розробника (дослідника).

Результати розрахунків зведемо до таблиці 5.5.

Таблиця 5.4 — Основна заробітна плата розробників

Найменування посад и	Місячний посадовий оклад, грн.	О плата за робочий день, грн.	Число днів в роботи	В итрати на заробітну плату, грн.
Керівник проекту	45000	1 956,52	33	64 565,217
Програміст	41500	1 804,35	33	59 543,478
Всього				12 4108,70

Так як в даному випадку розробляється програмний продукт, то розробник виступає одночасно і основним робітником, і тестувальником розроблюваного програмного продукту.

Додаткова заробітна плата розробників, які брати участь в розробці обладнання/програмного продукту.

Додаткову заробітну плату прийнято розраховувати як 14 % від основної заробітної плати розробників та робітників:

$$Z_d = Z_o \cdot 14 \% / 100 \% \quad (5.2)$$

$$Z_d = (124108,70 \cdot 14 \% / 100 \% ) = 17375,22 \text{ (грн.)}$$

### 5.3 Нарахування на заробітну плату розробників

Згідно діючого законодавства нарахування на заробітну плату складають 22 % від суми основної та додаткової заробітної плати.

$$H_z = (Z_o + Z_d) \cdot 22 \% / 100 \% \quad (5.3)$$

$$H_3 = (124108,70 + 17375,22) \cdot 22 \% / 100 \% = 31126,46 \text{ (грн.)}$$

Оскільки для розроблювального пристрою не потрібновитрачатиматеріали та комплектуючі, то витрати на матеріали і комплектуючідорівнюють нулю.

Амортизаціяобладнання, яке використовувалось для проведеннярозробки.

Амортизаціяобладнання, щовикористовувалось для розробки в спрощеномувидлядірозраховується за формулою:

$$A = \frac{Ц}{T_{в}} \cdot \frac{t_{вик}}{12} \text{ [грн.]}. \quad (5.4)$$

де Ц—балансова вартість обладнання, грн.;

T

—термінкорисноговикористанняобладнаннязгідноподатковогозаконодавства, років

$t_{вик}$ —термінвикористанняпід час розробки, місяців

Розрахуємо, для прикладу, амортизаційні витрати на комп'ютербалансовавартість якого становить 21000 грн., термін його корисного використання згідно податкового законодавства – 2 роки, а термін його фактичного використання – 1,43 міс.

$$A_{обл} = \frac{21000}{2} \times \frac{1,43}{12} = 1255,435 \text{ грн.}$$

Аналогічно визначаємо амортизаційні витрати на інше обладнання та приміщення. Розрахунки заносимо до таблиці 5.5. Так як вартість ліцензійної ОС та спеціалізованих ліцензійних нематеріальних ресурсів менше 20000 грн, то даний нематеріальний актив не амортизується, а його вартість включається у вартість розробки повністю,  $B_{нем.ак.} = 8000$  грн.

Таблиця 5.5 — Амортизаційні відрахування на матеріальні та нематеріальні ресурси для розробників

Найменування обладнання	Балансова вартість, грн.	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн.
Комп'ютер та комп'ютерна периферія (ноутбук ASUS, зовнішня карта відеозахоплення HDMI to USB 2.0 VCC-01 Black, CCTV об'єктив для систем відеоспостереження Tamron 13VG308ASIR II)	21000	2	1,43	1255,435
Офісне обладнання (меблі)	35000	4	1,43	1046,196
Приміщення	1500000	20	1,43	8967,391
Всього				11269,02

Тарифи на електроенергію для непобутових споживачів (промислових підприємств) відрізняються від тарифів на електроенергію для населення. При цьому тарифи на розподіл електроенергії у різних постачальників (енергорозподільних компаній), будуть різними. Крім того, розмір тарифу залежить від класу напруги (1-й або 2-й клас). Тарифи на розподіл електроенергії для всіх енергорозподільних компаній встановлює Національна комісія з регулювання енергетики і комунальних послуг (НКРЕКП). Витрати на силову електроенергію розраховуються за формулою:

$$V_e = V \cdot \Pi \cdot \Phi \cdot K_{\Pi}, \quad (5.5)$$

де  $V$  — вартість 1 кВт-години електроенергії для 1 класу підприємства з ПДВ в 2024 році для Вінницької області за даними Енера-Вінниця,  $V = (5635,47/1000) \cdot 1,2 = 6,76$  грн./кВт;

$\Pi$  — встановлена потужність обладнання, кВт.  $\Pi = 0,35$  кВт;

$\Phi$  — фактична кількість годин роботи обладнання, годин.

$K_{\Pi}$  — коефіцієнт використання потужності,  $K_{\Pi} = 0,9$ .

$$V_e = 0,9 \cdot 0,35 \cdot 8 \cdot 33 \cdot 6,76 = 562,1616 \text{ (грн.)}$$

Інші витрати та загальновиробничі витрати.

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками. Витрати за статтею «Інші витрати» розраховуються як 50...100% від суми основної заробітної плати дослідників:

$$I_g = (Z_o + Z_p) \cdot \frac{H_{ib}}{100\%}, \quad (5.6)$$

де  $H_{ib}$  — норма нарахування за статтею «Інші витрати».

$$I_g = 124108,70 \cdot 90\% / 100\% = 111697,8 \text{ (грн.)}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін. Витрати за статтею «Накладні

(загально виробничі) витрати» розраховуються як 100...150% від суми основної заробітної плати дослідників:

$$H_{нзв} = (3_o + 3_p) \cdot \frac{H_{нзв}}{100\%}, \quad (5.7)$$

де  $H_{нзв}$  — норма нарахування за статтею «Накладні (загально виробничі) витрати».

$$H_{нзв} = 124108,70 * 150 \% / 100 \% = 186163 \text{ (грн.)}$$

Витрати на проведення науково-дослідної роботи.

Сума всіх попередніх статей витрат дає загальні витрати на проведення науково-дослідної роботи:

$$B_{заг} = 124108,70 + 17375,22 + 31126,46 + 11269,02 + 8000 + 562,16 + 111697,8 + 186163 = 490302,43 \text{ грн.}$$

Розрахунок загальних витрат на науково-дослідну (науково-технічну) роботу та оформлення її результатів.

Загальні витрати на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{B_{заг}}{\eta} \text{ (грн)}, \quad (5.8)$$

де  $\eta$  — коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи.

Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то  $\eta=0,1$ ; технічного проектування, то  $\eta=0,2$ ; розробки конструкторської документації, то  $\eta=0,3$ ; розробки технологій, то

$\eta=0,4$ ; розробки дослідного зразка, то  $\eta=0,5$ ; розробки промислового зразка, то  $\eta=0,7$ ; впровадження, то  $\eta=0,9$ . Оберемо  $\eta = 0,5$ , так як розробка, на даний момент, знаходиться на стадії дослідного зразка:

$$3B = 490302,43 / 0,5 = 980605 \text{ грн.}$$

### 5.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнювальним позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів цієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку. Саме зростання чистого прибутку забезпечить потенційному інвестору надходження додаткових коштів, дозволить покращити фінансові результати його діяльності, підвищить конкурентоспроможність та може позитивно вплинути на ухвалення рішення щодо комерціалізації цієї розробки.

Для того, щоб розрахувати можливе зростання чистого прибутку у потенційного інвестора від можливого впровадження науково-технічної розробки необхідно:

- а) вказати, з якого часу можуть бути впроваджені результати науково-технічної розробки;
- б) зазначити, протягом скількох років після впровадження цієї науково-технічної розробки очікуються основні позитивні результати для потенційного інвестора (наприклад, протягом 3-х років після її впровадження);
- в) кількісно оцінити величину існуючого та майбутнього попиту на цю або аналогічні чи подібні науково-технічні розробки та назвати основних суб'єктів (зацікавлених осіб) цього попиту;
- г) визначити ціну реалізації на ринку науково-технічних розробок з аналогічними чи подібними функціями.



При розрахунку економічної ефективності потрібно обов'язково враховувати змінність грошей у часі, оскільки від вкладення інвестицій до отримання прибутку минає час. При оцінюванні ефективності інноваційних проєктів передбачається розрахунок таких важливих показників:

- абсолютного економічного ефекту (чистого дисконтованого доходу);
- внутрішньої економічної дохідності (внутрішньої норми дохідності);
- терміну окупності (дисконтованого терміну окупності).

Аналізуючи напрямки проведення науково-технічних розробок, розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором можна об'єднати, враховуючи визначені ситуації з відповідними умовами.

Розробка суттєво вдосконалення програмного засобу (програмного забезпечення, програмного продукту) для використання масовим споживачем.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$$\Delta\Pi_i = (\pm\Delta\Pi_0 \cdot N + \Pi_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\rho}{100}\right), \quad (5.9)$$

де  $\pm\Delta\Pi_0$ —змінна вартості програмного продукту (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу;

$N$ —кількість споживачів, які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки;

$\Pi_0$ —основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки,  $\Pi_0 = \Pi_{0\pm} \Delta\Pi_0$ ;

$C_6$ —вартість програмного продукту у році до впровадження результатів розробки;

$\Delta N$ —збільшення кількості споживачів продукту, в аналізовані періоди часу, від покращення його певних характеристик;

$\lambda$ —коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт  $\lambda = 0,8333$ .

$p$ —коефіцієнт, який враховує рентабельність продукту;

$\vartheta$ —ставка податку на прибуток, у 2024 році  $\vartheta = 18\%$ .

Припустимо, що при прогнозованій ціні 35000 грн. за одиницю виробу, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 5000 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року – на 1000 шт., протягом другого року – на 1500 шт., протягом третього року на 1800 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$$\Delta\Pi_1 = (0 \cdot 5000 + (35000 + 5000) \cdot 1000) \cdot 0,8333 \cdot 0,2 \cdot (1 - 0,18) = 4783333,142 \text{ грн.}$$

$$\Delta\Pi_2 = (0 \cdot 5000 + (35000 + 5000) \cdot (1000 + 1500)) \cdot 0,8333 \cdot 0,2 \cdot (1 - 0,18) = 13666666,120 \text{ грн.}$$

$$\Delta\Pi_3 = (0 \cdot 5000 + (35000 + 5000) \cdot (1000 + 1500 + 1800)) \cdot 0,8333 \cdot 0,2 \cdot (1 - 0,18) = 23506665,726 \text{ грн.}$$

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 41956664,99 грн.

### 5.5 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Розраховуємо приведену вартість збільшення всіх чистих прибутків  $\Pi_3$ , що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (5.10)$$

де  $\Delta\Pi_i$  – збільшення чистого прибутку у кожному ізроків, протягом яких виявляються результати виконаної та впровадженої науково-дослідної (науково-технічної) роботи, грн;

$T$  — період часу, протягом якого виявляються результати впровадженої науково-дослідної (науково-технічної) роботи, роки;

$\tau$  – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні,  $\tau = 0,05 \dots 0,15$ ;

$t$  — період часу (в роках).

Збільшення прибутку ми отримаємо, починаючи з першого року:

$$ПП = (4783333,142/(1+0,1)^1) + (13666666,120/(1+0,1)^2) + (23506665,726/(1+0,1)^3) = 4348484,67 + 11294765,39 + 17660905,88 = 33304155,94 \text{ грн.}$$

Далі розраховують величину початкових інвестицій  $PV$ , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{инв} * ЗВ, \quad (5.11)$$

де  $k_{инв}$  — коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай  $k_{инв} = 2 \dots 5$ , але може бути і більшим;

$ЗВ$  — загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 2 * 980605 = 1961209,71 \text{ грн.}$$

Тоді абсолютний економічний ефект  $E_{abc}$  або чистий приведений дохід ( $NPV$ , *Net Present Value*) для потенційного інвестора від можливо впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{abc} = III - PV, \quad (5.12)$$

$$E_{abc} = 33304155,94 - 1961209,71 = 31342946,24 \text{ грн.}$$

Оскільки  $E_{abc} > 0$  то вкладання коштів на виконання та впровадження результату даної науково-дослідної (науково-технічної) роботи може бути доцільним.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність або показник внутрішньої норми дохідності ( $IRR$ , *Internal Rate of Return*) вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій  $E_{\theta}$ . Для цього використаємо формулу:

$$E_{\theta} = \sqrt[T_{жк}]{1 + \frac{E_{abc}}{PV}} - 1, \quad (5.13)$$

$T_{жк}$  – життєвий цикл наукової розробки, роки.

$$E_{\theta} = \sqrt[3]{1 + 31342946,24/1961209,71} - 1 = 1,570$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (5.14)$$

де  $d$ —середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні  $d = (0,09 \dots 0,14)$ ;

$f$ —показник, що характеризує ризикованість вкладень; зазвичай, величина  $f = (0,05 \dots 0,5)$ .

$$\tau_{\min} = 0,14 + 0,05 = 0,19.$$

Так як  $E_B > \tau_{\min}$ , то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термінокупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_B}, \quad (5.15)$$

$$T_{ок} = 1 / 1,570 = 0,64 \text{ р.}$$

Оскільки  $T_{ок} < 3$ -х років, а сама термінокупність рівний 0,64 роки, то фінансування даної наукової розробки є доцільним.

Висновки до розділу: економічна частина даної роботи містить розрахунок витрат на розробку нового програмного продукту, сума яких складає 980605 гривень.

Було прогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розраховано періодокупності витрат для інвестора та економічний ефект при використанні даної розробки.

В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є висококонкурентоспроможним. Період окупності складе близько 0,64 роки.

:

## ВИСНОВКИ

В роботі було проаналізовано, як зростаючі транспортні потреби призводять до збільшення автомобільного парку та кількості поїздок. Також пояснюється, на основі статистики поведінки водіїв за кермом, як мобільні технології несуть у собі значний ризик для здоров'я людини, що є ціною, яку сьогодні доводиться платити за дедалі швидші та віддаленіші подорожі. Згідно зі статистикою, неуважність та небезпечне водіння є причиною багатьох аварій. Тому в роботі було запропонована розроблена системи керування транспортним засобом у потокці на основі аналізу зображень, яка ґрунтується на наступних критеріях: Здатність виявляти ризиковані або неправильні маневри, зокрема ті, що пов'язані з ненавмисною зміною смуги руху або виїздом з неї, або ті, які, навіть якщо вони ненавмисні, не відповідають правилам або пов'язані з ризиком аварії. Їх легко встановити на транспортний засіб.

Розробка цієї системи також призведе до впровадження комп'ютерної системи виявлення та відстеження прямих смуг руху з фільтром Калмана (з часом буде відкинута); система виявлення та відстеження кривих смуг руху з сажовим фільтром; система класифікації смуг руху; система оповіщення про якість смуги руху.

Вищеописані алгоритми є відповіддю на ряд проблем, які необхідно було вирішити під час вдосконалення цих методів.

Просторова карта, що використовується, придатна для більшості ситуацій, що виникають на дорозі, але її можна покращити таким чином, щоб врахувати особливості, такі як поява та вигляд бічних рейок та перехресть. Це дозволило б уникнути проблеми вібрацій у системі технічного зору, покращити розташування дорожніх ліній і дозволило б аналізувати ширшу демаркацію дороги. Враховувати зникнення або поступове зникнення смуг руху, а також роздвоєння смуг. Розширити діапазон типів дорожньої розмітки, що розпізнаються. Зокрема, здатність розпізнавати подвійні лінії та

розрізняти більше типів переривчастих смуг полегшить розрізнення автомагістралей та звичайних доріг.

Реалізований режим використовує виключно інформацію, надану камерами, а нещодавно в транспортному засобі було встановлено навігаційну систему, яка надає необхідну інформацію про транспортний засіб самостійно. Тому пропонується: Включити зворотний зв'язок з транспортним засобом в обмеження швидкості транспортного засобу, передбачається, що це покращить ефективність дотримання смуги руху та дозволить заздалегідь більш точно прогнозувати напрямок руху.



## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Forsyth D.A. Computer Vision a modern approach / D.A. Forsyth, J. Ponce // Upper Saddle River, New Jersey. – 1993. – 828 p.
2. Longuet-Higgins H.C. Computer A computer algorithm for reconstructing a scene from two projections / H.C. Longuet-Higgins // University of Sussex, Brighton. – 1981. – 6 p.
3. D. Martín Gómez, «IVVI 2.0: An intelligent vehicle based on computational perception,» 2015.
4. J. Rousseeuw, «Least Median of Squares Regression,» Journal of the American Statistical Association, 1985.
5. C. H. P. A. & G. F. Rodriguez-Garavito, COMPUTER VISION APPLIED TO ROAD LINES RECOGNITION USING MACHINE LEARNING, ICIT, 2015.
6. Bose, B. and Grimson, E. (2003). Ground plane rectification by tracking moving objects. In Proceedings of the Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. Dailey, D. J. and Cathey, F. W. (2005).
7. A novel technique to dynamically measure vehicle speed using uncalibrated roadway cameras. In Proceedings IEEE. Fischler, M. A. and Bolles, R. C. (1981).
8. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. In Comm. Assoc. Comp. Mach. Kalman, R. E. and Bucy, R. S. (1960).
9. New results in linear filtering and prediction theory. In Transactions of the ASME, Journal of Basic Engineering (Series D). Lazaros Grammatikopoulos, George Karras, E. P. (2002).
10. Automatic estimation of vehicle speed from uncalibrated video sequences. In International Archives of Photogrammetry and Remote Sensing. Liebowitz, D. and Zisserman, A. (1998).

11. Metric rectification for perspective images of planes. In Proceedings of Computer Vision and Pattern Recognition. Lucas, B. D. and Kanade, T. (1981).
12. An iterative image registration technique with an application to stereo vision. In Proceedings DARPA Image Understanding Workshop. Schoepflin, T. and Dailey, D. J. (2003).
13. Dynamic camera calibration of roadsides traffic management cameras for vehicle speed estimation. In IEEE Transactions on Intelligent Transportation Systems
14. Istrip James Lenz, Donald R. Stauffer Donald R. Stauffer Electronic rumble // Proc. SPIE 2902, Transportation Sensors and Controls: Collision Avoidance, Traffic Management, and ITS, 106 (February 17, 1997);
15. R. M. Haralick, L. G. Shapiro Haralick R. M. Image Segmentation Techniques Computer Vision, Graphics, and Image Processing, Vol 29, No 1, 1985
16. N. R. Pal, S. K. Pal. A Review on Image Segmentation Techniques. Pattern Recognition, Vol. 26, No 9, 1993.
17. National Highway Transportation and Safety Administration. Fatality analysis reporting system. [Электронный ресурс] – Режим доступа: <http://www-fars.nhtsa.dot.gov/main/index.aspx>
18. J.A. Hadden, J.H. Everson, D.B. Pape, V.K. Narendran, D.A. Pomerleau Modeling and analysis of driver/vehicle dynamics with "run-off-road" crash avoidance systems. In Proceedings of the 30th ISATA Conference, 1997.
19. R.J. Tibshirani, Redelmeier D.A. Association between cellular-telephone calls and motor vehicle collisions. New England Journal of Medicine, 1997
20. Amol Borkar. A New Multi-camera Approach for Lane Departure Warning / Amol Borkar, Monson Hayes, Mark T. Smith // In proceeding of: Advances Concepts for Intelligent Vision Systems - 13th International Conference, ACIVS 2011, Ghent, Belgium, August 22-25, 2011.
21. Visvikis C. Study on lane departure warning and lane change assistant systems / C Visvikis, T L

- Smith, M Pitcherand R Smith // Technicalassistanceandeconomicanalysisinthefiledoflegislationpertinenttotheissueofautomotivesafety. TransportResearchLaboratoryNovember 2008
22. Wiley J. AlgorithmsforImageProcessingandComputerVision / JohnWiley&Sons // Inc. NewYork, NY, USA ©1996 ISBN0471140562
23. Davies E. MachineVision: Theory, AlgorithmsandPracticalities / E. Davies // AcademicPress, 1990, Chap. 5.
24. Computervision [Электронный ресурс] – Режим доступа: [http://en.wikipedia.org/wiki/Computer\\_vision](http://en.wikipedia.org/wiki/Computer_vision)
25. IJCA  
ProceedingsonInternationalConferenceonInnovationinCommunication,  
InformationandComputing 2013 IJCA Journal, ICICIC 2013 - Number 1,  
ColorImageSegmentationusing OTSU MethodandColorSpace
26. RafaelGonzález C. DigitalImageProcessing / Rafael C. González,  
RichardEugeneWoods // PrenticeHall. p. 85. ISBN 0-13-168728-X.
27. Normalization (imageprocessing) [Электронный ресурс] – Режим  
доступу: [http://en.wikipedia.org/wiki/Normalization\\_%28image\\_processing%29](http://en.wikipedia.org/wiki/Normalization_%28image_processing%29)
28. 18. HSL and HSV [Электронный ресурс] – Режим доступа:  
[http://en.wikipedia.org/wiki/HSL\\_and\\_HSV](http://en.wikipedia.org/wiki/HSL_and_HSV)
29. Cannyedgedetector [Электронный ресурс] – Режим доступа:  
[http://en.wikipedia.org/wiki/Canny\\_edge\\_detector](http://en.wikipedia.org/wiki/Canny_edge_detector)
30. GaryBradski, AdrianKaehler «LearningOpenCV.  
ComputervisionwiththeOpenCVlibrary».
31. Leastsquares [Электронный ресурс] – Режим доступа:  
[http://en.wikipedia.org/wiki/Least\\_squares](http://en.wikipedia.org/wiki/Least_squares)
32. Houghtransform [Электронный ресурс] – Режим доступа:  
[http://en.wikipedia.org/wiki/Hough\\_transform](http://en.wikipedia.org/wiki/Hough_transform)
33. Huang T.S. SomePropertiesofthe E MatrixinTwo-  
ViewMotionEstimation / T.S. Huang, O.D. Faugeras // Correspondence. – 1989. –  
5 p.

34. Faugeras O.D. Three-Dimensional Computer Vision: a geometric viewpoint / O.D. Faugeras // Massachusetts, Cambridge. – 1993. – 675 p.
35. Hartley R. Multiple View Geometry in Computer Vision: Second Edition / R. Hartley, A. Zisserman // Cambridge University Press. – 2005. – 655 p.
36. Tsai R.Y. Multiple frame Image Restoration and Registration Advances in Computer Vision and Image Processing / T.S. Huang, R.Y. Tsai // JAI Press, Greenwich. – 1985. – 22 p.
37. CCD and CMOS sensors [Электронный ресурс] – режим доступа: <http://www.issibern.ch/forads/sr-009-23.pdf>
38. CCD area image sensor [Электронный ресурс] – режим доступа: <https://www.hamamatsu.com/eu/en/product/optical-sensors/image-sensor/ccd-cmos-nmos-image-sensor/area-sensor/for-scientific-measurement/index.html>.
39. КМОН-сенсор [Электронный ресурс] – режим доступа: <https://uk.wikipedia.org/wiki/КМОН-сенсор>.
40. CMOS area image sensor [Электронный ресурс] – режим доступа: <https://www.hamamatsu.com/eu/en/product/optical-sensors/image-sensor/ccd-cmos-nmos-image-sensor/area-sensor/for-camera/index.html>.
41. Mercury cadmium telluride [Электронный ресурс] – режим доступа: [https://en.wikipedia.org/wiki/Mercury\\_cadmium\\_telluride](https://en.wikipedia.org/wiki/Mercury_cadmium_telluride).
42. Інфрачервоні детектори на основі потрійних напівпровідникових квантових структур [Электронный ресурс] – режим доступа: <http://visnykpb.kpi.ua/article/view/92951>.
43. Kaehler A. Learning OpenCV 3 Computer Vision in C++ / A. Kaehler, G. Bradski // USA. – 2017. – 992 p.
44. PCA9515 I<sup>2</sup>C-bus repeater. [Электронный ресурс]. Режим доступа: <https://www.nxp.com/docs/en/data-sheet/PCA9515.pdf>.
45. Arduino Nano 3.0. [Электронный ресурс] Режим доступа: <https://doc.arduino.ua/ru/hardware/Nano>.
46. Arduino Relay Module 1 relay 5V. [Электронный ресурс]. Режим доступа: <http://www.kosmodrom.com.ua/el.php?name=RelayModule1relay5V>

47. Методичні вказівки до виконання магістерських кваліфікаційних робіт студентами спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти денної та заочної форм навчання. / Укладачі О.Д. Азаров, О.В. Дудник, С.І. Швець – Вінниця : ВНТУ, 2023. – 57 с.

48. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. — Вінниця : ВНТУ, 2021. 42 с.

49. КОНФЕРЕНЦІЇ ВНТУ електронні наукові видання, Молодь в науці: дослідження, проблеми, перспективи (МН-2024)/ <https://conferences.vntu.edu.ua/index.php/mn/mn2024/paper/view/21418>

**ДОДАТОК А**

Технічне завдання

Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ

Завідувачка кафедри ОТ

проф., д.т.н.. Азаров О.Д..

“ 29 ” \_\_\_\_\_ 02 \_\_\_\_\_ 2024 р.**ТЕХНІЧНЕ ЗАВДАННЯ**

на виконання магістерської кваліфікаційної роботи

“Система охоронної пожежної сигналізації з визначенням джерела  
загоряння”

08-55.МКР.002.00.000 ПЗ

Науковий керівник: к.т.н., доцент каф. ОТ

\_\_\_\_\_ Колесник І.С..

Студент групи КІ-22мз

\_\_\_\_\_ Гринчак О.О.

## 1 Підстава для виконання магістерської кваліфікаційної роботи (МКР)

1.1 Основною актуальністю роботи введення амортизованого виводу як способу передбачення з високою ймовірністю, що дозволяє прогнозувати рішення на основі спостережень для розв'язання складних завдань, знижуючи обчислювальні витрати та забезпечуючи більш ефективні підходи.

### 1.2 Наказ про затвердження теми МКР.

## 2 Мета МКР і призначення розробки

2.1 Мета роботи — полягає у розширенні функціональних можливостей комп'ютерної системи для попередження та реєстрації несанкціонованої зміни швидкості руху із застосування методу детектування дорожньої розмітки, що дозволяє отримати кращі показники швидкодії та точності детектування.

2.2 Визначення підходів до побудови кінцевого модуля комп'ютеризованої системи попередження ДТП, що забезпечує збір та аналіз даних від зовнішніх відео датчиків.

## 3 Вихідні дані для виконання МКР

— провести аналіз тенденцій в області автоматизованих систем запобігання ДТП, підходів до побудови сигналізації водію про небезпеку;

— розробити підходи до побудови сигналізації із методом детектування зміни полоси руху;

— робити систему, що буде функціонувати в режимі реального часу;

— визначити функціональну побудову модуля системи сигналізації з її функціональними можливостями;

— дослідити практичні аспекти функціонування такої системи.

## 4 Етапи МКР та очікувані результати

Етапи роботи та очікувані результати приведено в Таблиці А.1.

Таблиця А.1 — Етапи МКР

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		початок	кінець	
1	Обґрунтування проблеми, методів та технік у глибокому навчанні	19.04.2024	25.04.2024	Розділ 1
2	Аналіз представлення фігур та форми	1.05.2024	19.05.2024	Розділ 2
3	Вибір даних для дослідження	20.05.2024	23.05.2024	Розділ 3
4	Експериментування	23.05.2024	30.05.2024	Розділ 4
4	Підготовка економічної частини	05.06.2024	23.06.2024	Розділ 5
6	Оформлення пояснювальної записки, графічного матеріалу і презентації	3.06.2024	06.06.2024	ПЗ, графічний матеріал і презентація
7	Підготовка і підпису супроводжуючих документів, нормоконтроль та тест на плагіат	06.06.2024	06.06.2024	Оформлені документи

## 6 Матеріали, що подаються до захисту МКР

До захисту подаються: пояснювальна записка МКР, графічні і ілюстративні матеріали, протокол попереднього захисту МКР на кафедрі, відгук наукового керівника, відгук опонента, протокол складання державних екзаменів, анотації до МКР українською та іноземною мовами.

## 7 Порядок контролю виконання та захисту МКР

Виконання етапів графічної та розрахункової документації МКР контролюється науковим керівником згідно з встановленими термінами. Захист МКР відбувається на засіданні Екзаменаційної комісії, затвердженої наказом ректора.

## 8 Вимоги до оформлювання та порядок виконання МКР

### 8.1 При оформлюванні МКР використовуються:



— ДСТУ 3008: 2015 «Звіти в сфері науки і техніки. Структура та правила оформлювання»;

— ДСТУ 8302: 2015 «Бібліографічні посилання. Загальні положення та правила складання»;

— ГОСТ 2.104–2006 «Єдина система конструкторської документації. Основні написи»;

— методичні вказівки до виконання магістерських кваліфікаційних робіт зі спеціальності 123 — «Комп'ютерна інженерія»;

— документи на які посилаються у вищевказаних.

8.2 Порядок виконання МКР викладено в «Положення про кваліфікаційні роботи на другому (магістерському) рівні вищої освіти СУЯ ВНТУ–03.02.02 П.001.01:21.

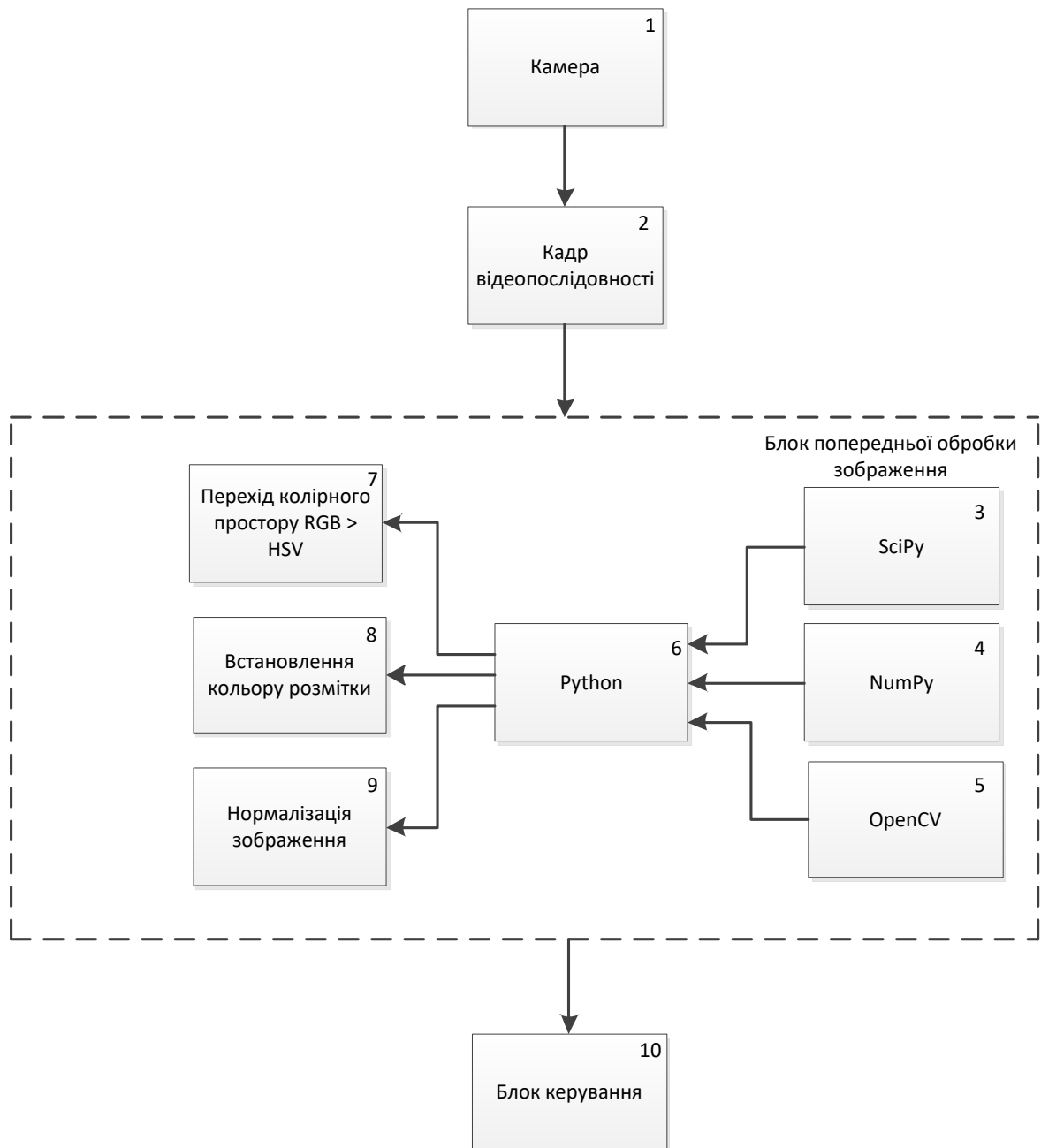
**ДОДАТОК Б****Структурна схема комп'ютерної системи контролю рядності руху**

Рисунок Б1 — Структурна схема комп'ютерної системи контролю рядності руху

## ДОДАТОК В

## Блок-схема методу визначення смуги руху

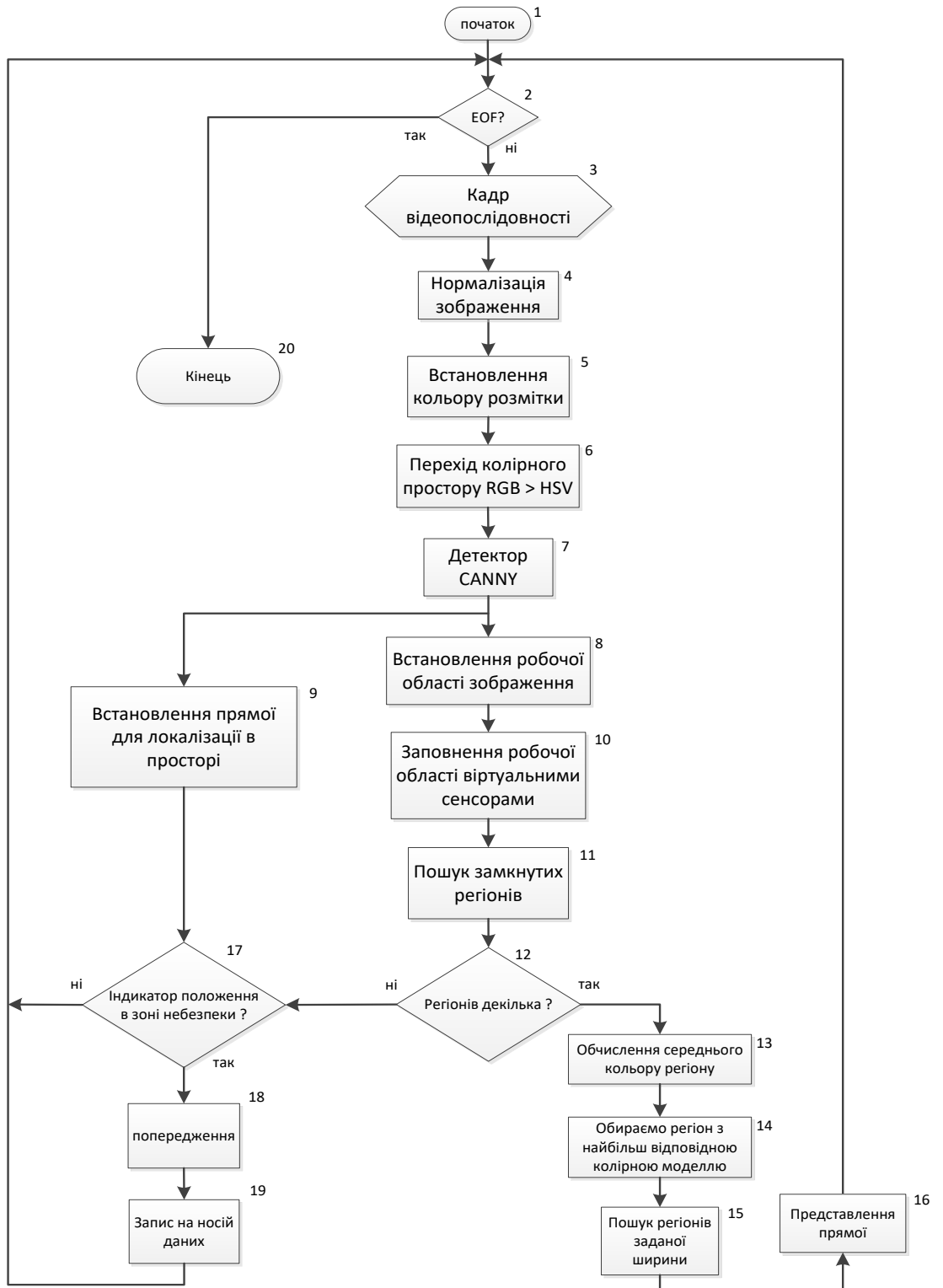


Рисунок В— Блок-схема методу визначення смуги руху

## Додаток Г

## Схема електрична структурна системи



Рисунок Г1 —Схема електрична структурна системи

## ДОДАТОК Д

## Схема електрична структурна блоку обробки зображення

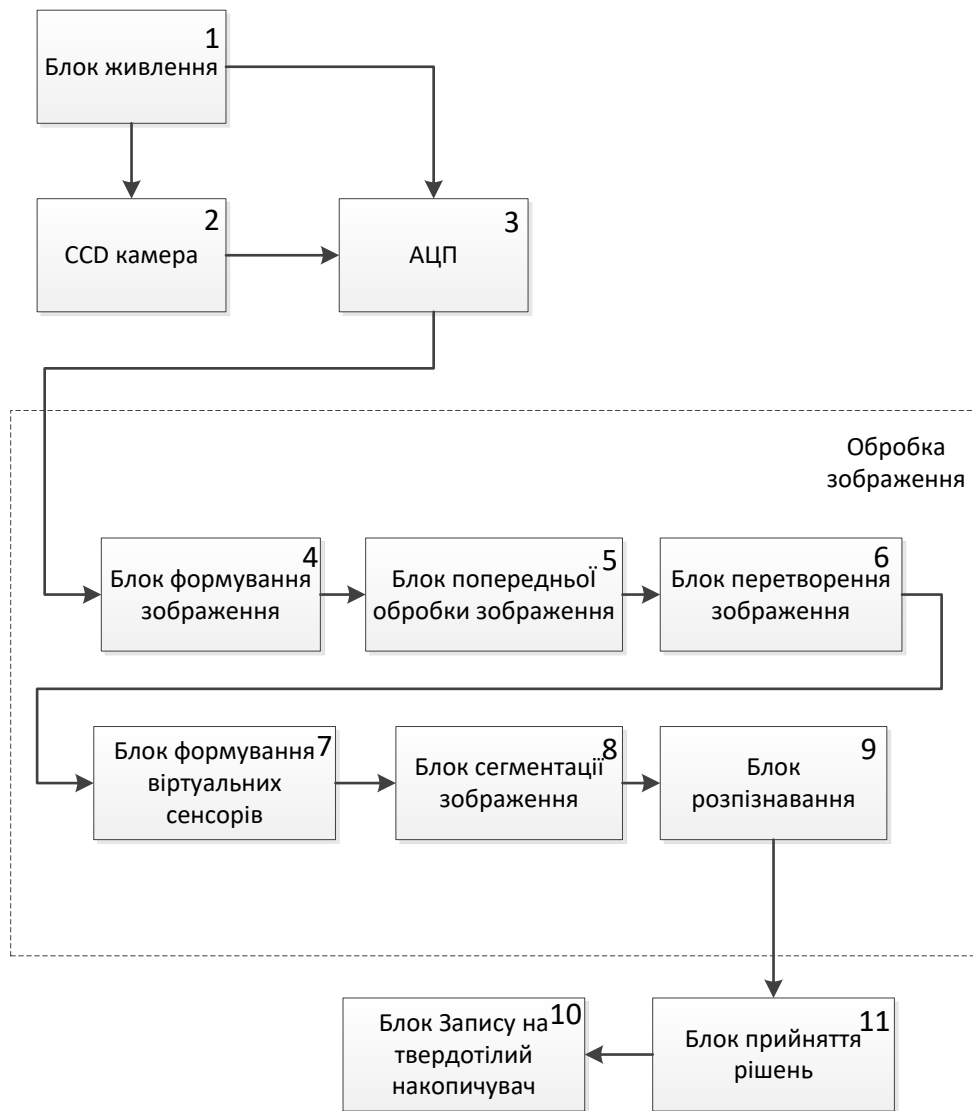


Рисунок Д1 —Схема електрична структурна блоку обробки зображення

## Додаток Е

### Модуль захоплення відео

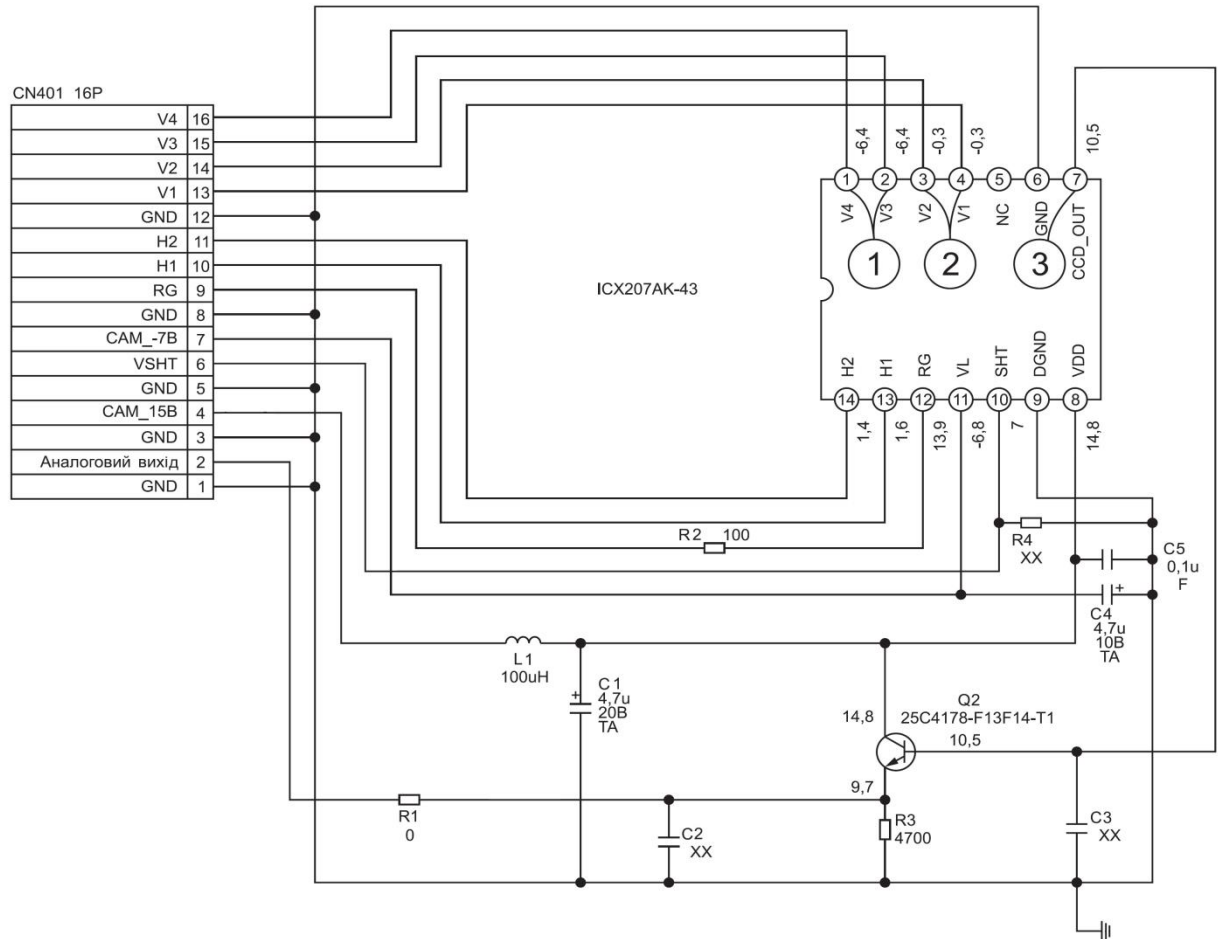


Рисунок Е1 —Модуль захоплення відео

**ДОДАТОК Ж**

Лістинг коду програми методу пошуку віртуальних сенсорів

Sensor.py

```
import cv2 as cv
```

```
import numpy as np
```

```
class LaneSensor():
```

```
    def __init__(self):
```

```
        self.xPos = 0
```

```
        self.yPos = 0
```

```
        self.width = 0
```

```
        self.lineRGB = [0, 0, 0]
```

```
        self.lineHSV = [0, 0, 0]
```

```
        self.lineWidth = [0, 0]
```

```
        self.roadRGB = [0, 0, 0]
```

```
        self.roadHSV = [0, 0, 0]
```

```
    def SetGeometry(self, position, width):
```

```
        self.xPos = max(0, position[0]-width/2)
```

```
        self.yPos = position[1]
```

```
        self.width = width
```

```
    def DrawGeometry(self, img):
```

```
        cv.line(img, (self.xPos, self.yPos), (self.xPos+self.width, self.yPos), [0, 0,  
255])
```

```
    def InitializeModel(self, linergb, linehsv, roadrgb, roadhsv):
```

```
        self.lineRGB = linergb
```

```
        self.lineHSV = linehsv
```

```
        self.roadRGB = roadrgb
```

```

self.roadHSV = roadhsv

    def CalculatePixelsProperties(self, rgb, hsv):
        #simple RGB distance
        rgbLaneError = np.abs(rgb - self.lineRGB)
        rgbLaneDistance =
np.sqrt(np.square(rgbLaneError[:,0])+np.square(rgbLaneError[:,1])+np.square(rgb
LaneError[:,2]))/3.0

        HLaneError = np.abs(hsv - self.lineHSV)[:, 0]/360.0

        #reliability is 1 for now
        probability = 1.0-(1.0*rgbLaneDistance + 0.0*HLaneError)/1.0
        reliability = np.ones_like(probability)

        return (probability, reliability)

    def FindSegments(self, rgbGlobal, hsvGlobal, cannyGlobal, outputImg,
previousLineCenterPosition):
        #crop image to a sensor size
        rgb = rgbGlobal[self.yPos, self.xPos:(self.xPos+self.width)]
        hsv = hsvGlobal[self.yPos, self.xPos:(self.xPos+self.width)]
        canny = cannyGlobal[self.yPos, self.xPos:(self.xPos+self.width)]

        probability, reliability = self.CalculatePixelsProperties(rgb, hsv)

        #check if we have any data
        if canny.shape[0] == 0: return [0, [], []]

        #find start of a first segment

```



```
segStart = 0
    while canny[segStart] == 0: #find start of a first segment
segStart+=1
    if segStart == canny.shape[0]:
        break
```

## ДОДАТОК И

Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень

Назва роботи: Комп'ютерна система визначення та контролю рядності руху

Тип роботи: магістерська кваліфікаційна робота  
(БДР,МКР)

Підрозділ кафедра обчислювальної техніки  
(кафедра, факультет)

### Показники звіту подібності Unichesk

Оригінальність \_\_\_\_\_ Схожість \_\_\_\_\_

Аналіз звіту подібності і (відмітити потрібне):

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку \_\_\_\_\_  
(підпис)

Захарченко С.М.  
(прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unichesk щодо роботи.

Автор роботи \_\_\_\_\_  
(підпис)

Гринчак О.О.  
(прізвище, ініціали)

Керівник роботи \_\_\_\_\_  
(підпис)

Колесник І.С.  
(прізвище, ініціали)