

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

«Інтелектуальна система керування автомобільною стоянкою»

Студента групи ІН.м-81н

Голінки А.Ю.

Завідувач випускаючої кафедри

Довбиш А.С.

Керівник роботи

Шелехов І.В.

Суми 2020

Сумський державний університет

(назва вузу)

Факультет Електроніки та інформаційних технологій

Кафедра Комп'ютерних наук

Спеціальність Комп'ютерні науки

Затверджую:

зав.кафедрою _____

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ

Голінки Артема Юрійовича

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Інтелектуальна система керування автомобільною стоянкою

затверджую наказом по інституту від “ _____ ” _____ 20__ р. № _____

2. Термін задачі студентом закінченого проекту (роботи) _____

3. Вхідні данні до проекту (роботи) зображення автомобільної парковки з камер відео-спостереження.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) Аналітичний огляд компонентів та різних типів автоматизованих систем для автостоянок. Аналіз порівнянь існуючих методів виявлення контуру об'єкту. Опис методу розв'язання поставленої задачі. Опис вхідних даних та алгоритму попередньої обробки зображення. Тестування системи за різних природних умов. Аналіз результатів тестування.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

14 схематичних зображення для пояснення; 3 фото прикладів автоматизованих систем; 15 фото з різних етапів роботи алгоритму; 1 графік з результатами тестування;

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
<i>1.3 Постановка задачі</i>	<i>Шелехов І.В.</i>		

7. Дата видачі завдання 02.03.2020

Керівник

(підпис)

Завдання прийняв до виконання

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1	<i>Пошук та вибір літератури для виконання дипломного проекту. Складення бібліографії.</i>	<i>02.03.2020 – 13.03.2020</i>	
2	<i>Написання першого розділу дипломного проекту: Аналіз проблеми на постановка задачі</i>	<i>16.03.2020 – 20.03.2020</i>	
3	<i>Написання другого розділу дипломного проекту: Технологія інтелектуального управління автопарками</i>	<i>23.03.2020 – 27.03.2020</i>	
4	<i>Вибір та налаштування середовища розробки. Аналіз технологій реалізації подібних рішень. Вибір методу визначення контуру об'єкта</i>	<i>30.03.2020 – 03.04.2020</i>	
5	<i>Пошук вхідних даних, проведення дослідницького аналізу даних, попередньої обробки даних.</i>	<i>06.04.2020 – 17.04.2020</i>	
6	<i>Програмна реалізація інтелектуальної системи</i>	<i>20.04.2020 – 24.04.2020</i>	
7	<i>Проведення тестування програмної реалізації за різних умов</i>	<i>27.04.2020 – 01.05.2020</i>	
8	<i>Написання третього розділу дипломного проекту: Інформаційне та програмне забезпечення системи інтелектуального управління автопарками</i>	<i>04.05.2020 – 08.05.2020</i>	
9	<i>Написання розділів: Вступ, Висновки, Додатки</i>	<i>11.05.2020 – 15.05.2020</i>	

Студент – дипломник

(підпис)

Керівник проекту

(підпис)

РЕФЕРАТ

Записка: 54 стор., 33 рис., 3 табл., 1 додаток, 28 джерел.

Об'єкт дослідження — зображення автомобільної парковки з камер відео-спостереження.

Мета роботи — розробка інтелектуальної системи для автоматичного розпізнаванню заповнення автомобільної парковки

Методи дослідження — логіко-аналітичний, вимірювання, порівняння, емпіричний, візуальний, системний

Результати — проведено аналіз типів автоматичних систем автостоянок, порівняльний аналіз існуючих методів виявлення контуру об'єкту, розроблений функціонал автоматичного виявлення заповнення автомобільної парковки, проведено тестування даної реалізації, зроблені висновки по результатам тестування, дана оцінка актуальності та ефективності даної системи

ЗОБРАЖЕННЯ АВТОСТОЯНОК, МЕТОД ОПЕРАТОРА CANNY, МЕТОД ВИЯВЛЕННЯ РЕБЕР, АВТОМАТИЗАЦІЯ, РОЗПІЗНАВАННЯ ОБ'ЄКТІВ, СИСТЕМИ УПРАВЛІННЯ АВТОСТОЯНКОЮ

ЗМІСТ

ВСТУП	7
1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ	8
1.1 Системи управління автостоянкою	8
1.2 Категорії системи управління автостоянкою	11
1.2.1 Система на основі лічильника	12
1.2.2 Система на основі зображень.....	12
1.2.3 Система на основі ультразвукового датчика.....	13
1.2.4 Бездротова система	14
1.3 Апаратні засоби виявлення транспортних засобів.....	15
1.3.1 Активні (інтрузивні) датчики	16
1.3.2 Пасивні (неінтрузивні) датчики.....	20
1.4 Програмні засоби обробки сигналів і даних.....	25
1.5 Постановка задачі.....	26
2 ТЕХНОЛОГІЯ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ АВТОПАРКАМИ.....	27
2.1 Попередня обробка зображень.....	27
2.2 Порівняння методів виділення контуру на зображенні.....	30
2.3 Метод оператора Canny	32
2.3.1 Гауссовий фільтр.....	33
2.3.2 Пошук градієнта інтенсивності зображення	34
2.3.3 Немаксимальне придушення	34
2.3.4 Подвійний поріг	36
2.3.5 Відстеження країв за допомогою гістерезису	36
3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ АВТОПАРКАМИ.....	38
3.1 Опис вхідних даних.....	38
3.2 Короткий опис програми	41
3.3 Тестування системи.....	43
ВИСНОВОК	46
СПИСОК ЛІТЕРАТУРИ	48
ДОДАТОК	52

ВСТУП

Через збільшення кількості транспортних засобів на дорогах проблеми з дорожнім рухом завжди присутні. В деяких містах переважно відсутні вільні місця на узбіччі. Саме тому водії залишають автомобілі на тротуарах, при цьому заважаючи перехожим. Щоб позбутися таких ситуацій, потрібно вводити платне паркування.

Великі міста не мають змоги будувати нові сучасні автостоянки в центрі міста, тому потрібно використовувати вже існуючі і модернізувати їх. Нинішні некеровані автостоянки та транспортні засоби ускладнюють ефективному розміщенню більшої кількості транспортних засобів в умовах зменшення територій для забудови стоянок. Саме тому необхідно мати систему управління на великих автостоянках щоб забезпечити максимальну ефективність використання вільного місця та швидкості роботи.

Це дослідження спрямоване на аналіз типів систем управління автостоянкою, опис реалізації одного з таких типів системи та тестування на реальних даних. Дана система буде виявляти вільне місце на парковці, коли автомобіль виїжджає, щоб давати змогу паркувати водієві свій транспортний засіб.

1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Системи управління автостоянкою

Існує два типи систем управління парковкою. Перша – це звичайна система паркування, а друга – система автоматичного паркування. Автоматична система паркування використовується для того, щоб зробити весь процес паркування автомобілів більш ефективним і менш складним як для водіїв, так і для адміністраторів. Це можна зробити за допомогою датчиків біля входу та виходу з парку, комп'ютерної системи, яка керує всім процесом та різних дисплейних панелей та ліхтарів, які допомагають водієві стояти свій автомобіль. Приклад такої системи, яка використовує автоматичну систему паркування [1], показана на рис 1.1

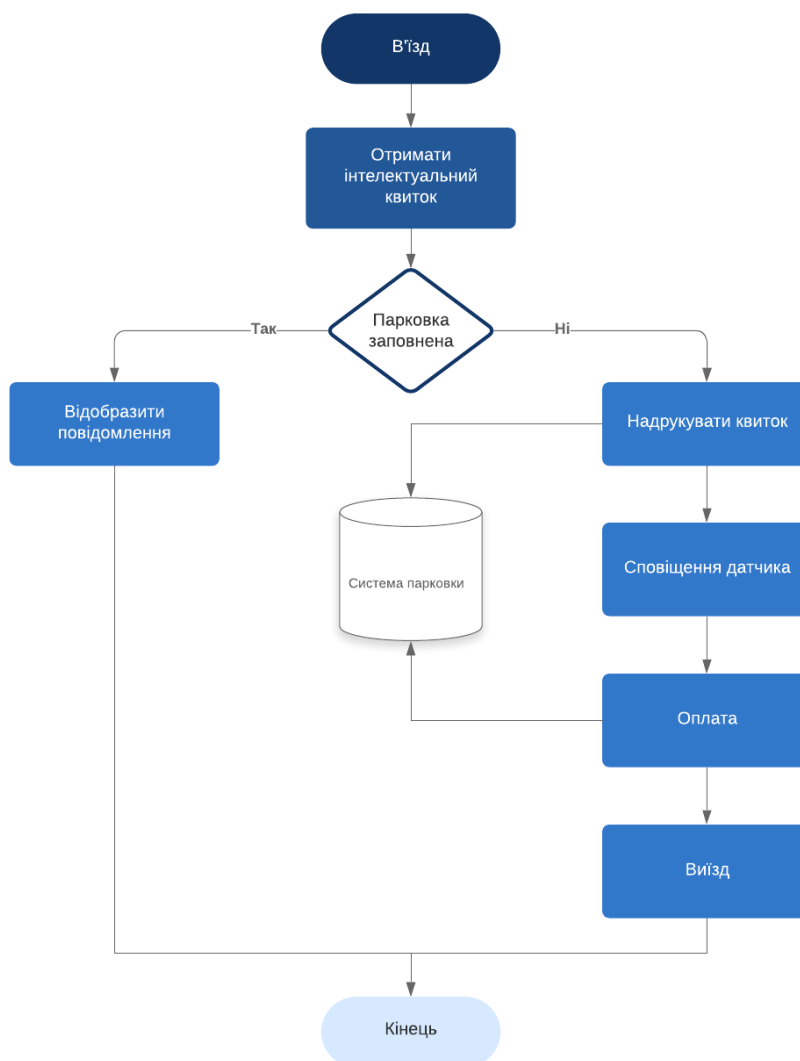


Рисунок 1.1 – Інтелектуальне управління системою паркування автомобілів

Наявність місця для паркування транспортного засобу повинно визначатися в режимі реального часу. Крім того, система автоматичного паркування зберігає дані про час, коли автомобіль був на парковці та обчислює суму, яку потрібно буде оплатити. Для впровадження інтелектуальної системи управління парковкою автомобіля необхідно застосувати одну з сучасних технологій обробки зображень. Основним елементом такої системи буде відеокамера, за допомогою якої формуватимуться зображення парковки і автомобілів. Маючи зображення, інтелектуальна система здатна визначити конкретні вільні для паркування автомобілів і направляти водія до них.

Перш ніж впровадити таку систему, потрібно врахувати її ефективність порівняно з вартістю. Основні переваги при використанні системи управління автостоянкою (СУАС) з автостоянкою без автоматизацій та систем управління перераховані нижче:

- 1) СУАС дозволяє підвищити економність та ефективність у використанні простору, тобто паркувати більше автомобілів на невеликій площі.[6]
- 2) СУАС зменшує затори та аварії на автостоянках.
- 3) СУАС підвищить якість обслуговування клієнтів парковки.
- 4) СУАС здатна підвищити гнучкість дизайну для ділянок, для яких потрібні автостоянки.
- 5) СУАС надає можливість введення нових видів розміщення паркувальних місць та оперативного реагування на нові вимоги щодо парковок.
- 6) СУАС зменшує згубний вплив на навколишнє середовище [10].

Також у СУАС є деякі інші переваги щодо пов'язані з безпекою автомобілів, що вже знаходяться на парковці, безпекою руху автомобілів, які заїжджають або виїжджають з парковки. Використовуючи систему управління автостоянкою, автомобілі витратять менше часу на переміщення навколо частин автостоянки, у яких немає місця та зможуть їхати безпосередньо до

вільного для паркування місця. Рух на автостоянці також може бути небезпечним, оскільки водію перешкоджають припарковані автомобілі, а також автомобілі або пішоходів, що рухаються між припаркованими автомобілями.[5, 7].

СУАС здатні наблизити ефективність звичайної автостоянки до сучасних автоматизованих, що здатні переміщати машини до визначеного місця на парковці самостійно, використовуючи, наприклад, конвеєрні стрічки. В порівнянні зі звичайною такі автостоянки розміщують щонайменше на 30% більше автомобілів. СУАС дозволяє звичайній автомобільній парковці розмістити більше автомобілів в межах однієї зони, що особливо корисно у великих містах, де вартість площа землі досить значна.

Крім того, СУАС здатна оптимізувати витрати на очищення, вентиляцію, освітлення та протипожежний захист, що будуть забезпечуватися лише для зайнятих автомобілями паркувальних місць [14, 16].

Завдяки способу керування транспортними засобами на автостоянці зменшиться кількість автомобілів, які будуть їздити в межах автостоянки. Це зменшить витрачене паливо та кількість токсичних газів, таких як чадний газ, що виділяється в атмосферу [17].

При проектуванні систему управління автостоянкою враховуються такі основні вимоги і загальні принципи:

1. Водії повинні мати можливість вибору найбільш зручної парковки, яка є в наявності.
2. Інформація про доступність автостоянки може бути надана користувачам через Інтернет або через стільникову мережу.
3. Автоматична система повинна обслуговувати одночасно декілька водіїв.
4. СУАС має накопичувати статистичні дані про наповненість автостоянки.

5. Гнучкість – плани паркування повинні враховувати невизначеність та зміни.

6. Пріоритетність – найбільш бажаними місцями слід керувати, щоб надавати перевагу користувачам з вищим пріоритетом.

7. Ціноутворення – наскільки це можливо, користувачі повинні платити безпосередньо за паркування.

8. Пікове управління – слід докласти особливих зусиль для керування автостоянкою у час пік.

9. Якість паркінгів слід вважати такою ж важливою, як кількість, включаючи естетику, безпеку та доступність та інформацію про користувачів.

В даній роботі буде розглянуто реалізацію лише однієї з частин СУАС – автоматизація ведення підрахунку доступних парко-місць за допомогою зображення з камери.

1.2 Категорії системи управління автостоянкою

Чотири основні категорії системи управління автостоянкою, що використовують різні технології [2], показані на рис 1.2.



Рисунок 1.2 – Категорії системи управління автостоянкою

1.2.1 Система на основі лічильника

Система, що базується на лічильниках, використовує датчики, розташовані біля входу та виходу, щоб підрахувати кількість автомобілів, які в'їжджають та виїжджають з автостоянки. Ця система обчислює кількість доступних місць для паркування на ділянці автостоянки, віднімаючи від загальної кількості місць кількість машин, які в'їхали на автостоянку і додаючи кількість машин, які виїхали з неї. Таким чином, ця категорія САУС не може вказати водію, де слід паркуватися [21, 23]. На рис. 1.3 показано, як система, що базується на лічильниках, може відображати кількість доступних місць на кожному рівні автостоянки [3].

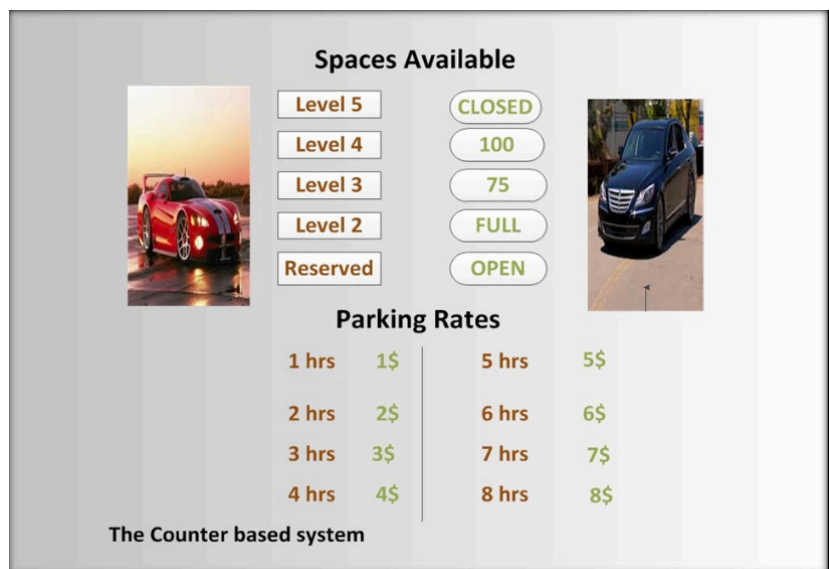


Рисунок 1.3 – Система на основі лічильника

1.2.2 Система на основі зображень

Техніка на основі зображень використовує камеру, щоб зробити знімок автомобільної стоянки. Це корисно, оскільки за допомогою цього зображення можуть бути відомі конкретні місця, які є вільними, що потім дозволить направити туди водія. Однак обсяг даних, знятих з камер, дуже великий, і датчики можуть бути досить дорогими [12, 16]. Приклад цієї системи показаний на рис. 1.4



Рисунок 1.4 - Система на основі зображень

На рис. 1.4 ліворуч показано наявність вільних місць, середня колонка – це також елемент графічного інтерфейсу, що підтверджує інформацію про вільні місця, та права колонки – це фото відповідних дисплеїв, що відображають таку інформацію водію.

1.2.3 Система на основі ультразвукового датчика

Система, що базується на ультразвукових датчиках, що встановлені на кожному паркувальному місці і з'єднані з центральним комп'ютером. Ця система також надає дані про наявність вільних місць, але значно дорожча за попередню, оскільки до її складу входить багато датчиків і окрема електрична система [27, 29]. Рис. 1.5 ілюструє приклад системи на основі ультразвукового датчика.

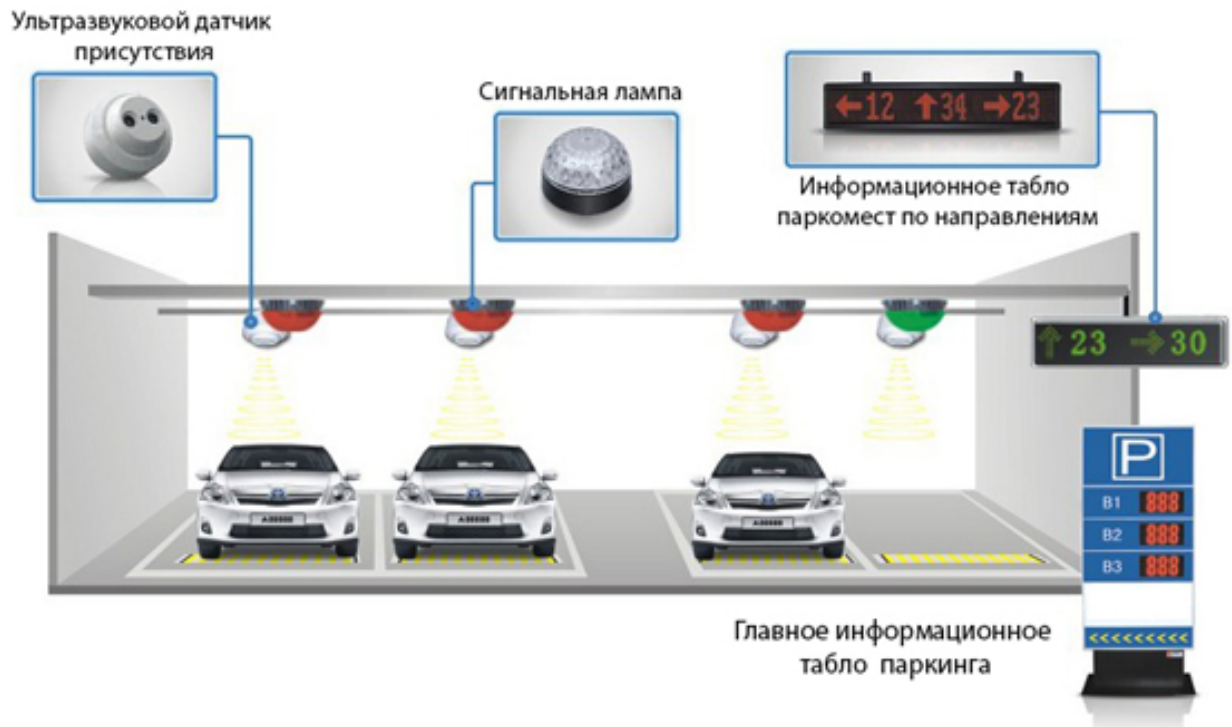


Рисунок 1.5 - Система на основі ультразвукового датчика

1.2.4 Бездротова система

Система на основі бездротової мережі використовує вузли, розташовані навколо автостоянки, які передають свої дані бездротово. Це зменшує необхідність витрат на встановлення та обслуговування та робить систему більш гнучкою. На рис. 1.6 показаний приклад бездротової системи. Ця система використовує вузли, які мають світловий, акустичний та температурний датчик і передають свої дані по Інтернету в базу даних, де можна проводити обробку, а також водіям, які перевіряють наявність вільних місць паркування через Інтернеті [30, 31].

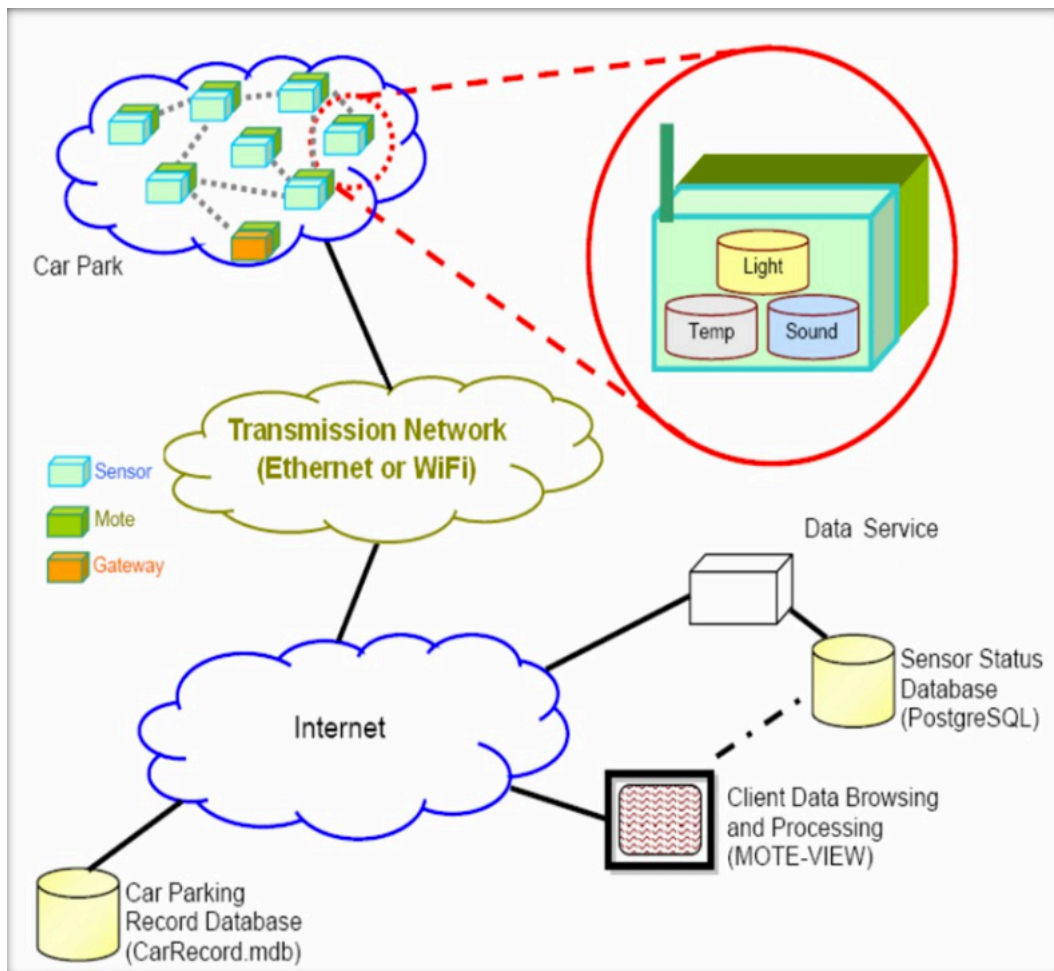


Рисунок 1.6 - Схема бездротової системи

Для впровадження системи управління автостоянкою потрібні датчики на вході та виїзді з автостоянки. Крім того, датчики також розміщуються в певних зонах автостоянки так, щоб можна було збирати інформацію про вільні і зайняті паркувальні місця. Оскільки дані з датчиків мають бути доступні комп'ютеру в режимі реального часу, то необхідно їх сигнали перетворювати у цифрову форму. Інформацію, отриману від усіх датчиків, потрібно збирати та обробляти комп'ютером, а потім відображати не тільки на екрані комп'ютера, але і на моніторах, розташованих на території стоянки для оперативного інформування водіїв [4, 6].

1.3 Апаратні засоби виявлення транспортних засобів

Існує два основних типи датчиків виявлення транспортних засобів, які: пасивні (неінтрузивні) та активні (інтрузивні) датчики. Активні датчики

детектують по одному транспортному засобу за раз. Їх необхідно розмістити безпосередньо в зонах руху автомобілів. Пасивні датчики можуть одночасно виявляти декілька транспортних засобів. Вони розміщуються на певній відстані від зон руху так, щоб можна побачити багато транспортних засобів.

1.3.1 Активні (інтрузивні) датчики

Для детектування транспортних засобів інтрузивними датчиками необхідно виконати модифікацію основної дороги, якою рухаються автомобілі, або доріжки для водіїв біля неї. Ця модифікація може включати такі речі, як вирізання ями на дорозі, тунель під дорогою або прикріплення датчика збоку від дороги. Такі зміни потребують припинення руху по дорозі, на час виконання відповідних робіт. Оскільки датчики розміщені на дорозі, вони будуть загрозувати пошкодженням транспортних засобів. В теперішній час активно використовуються індуктивні, пневматичні, магнітометричні, магнітні та вагові інтрузивні датчики [3, 9].

Індуктивні датчики – це найпоширеніший тип інтрузивних датчиків. Індуктивна котушка розміщується під тим місцем дороги, де проїжджає машина. Через котушка проходить сигнал з частотою від 10 до 50 Гц. Коли машина рухається над котушкою, метал у ній зменшує індуктивність в котушці, що змінює частоту коливань. При використанні двох таких датчиків можна не тільки підтвердити факт проїзду транспортному засобу, але і визначити швидкість його руху за формулою:

$$v = \frac{l}{t_2 - t_1},$$

де l = Відстань між двома датчиками в метрах, t_1 – час в'їзду автомобіля на перший датчик, t_2 – час в'їзду автомобіля на другий датчик [8, 4]

На рис. 1.7 показаний цей принцип роботи індуктивного детектора

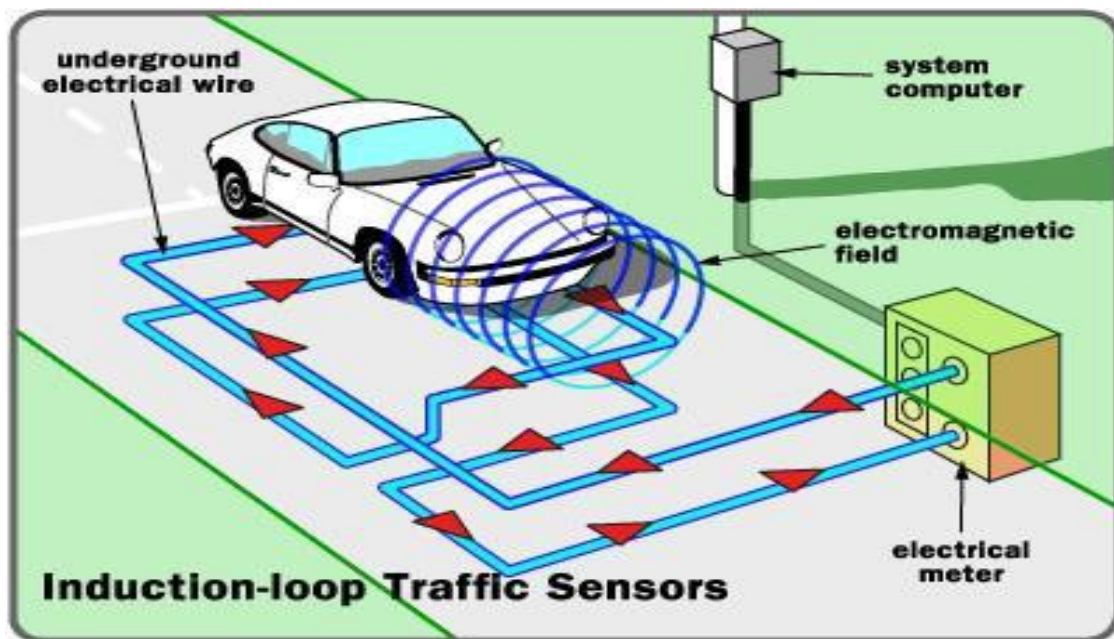


Рисунок 1.7 – Індуктивний детектор

Пневматичні датчики - це простий і недорогий метод підрахунку автомобілів, які проїздять повз. Довга гумова трубка, розміщена перпендикулярно до дороги, з'єднана з перемикачем, який відсилає сигнал на комп'ютер. Коли автомобіль проходить його, тиск повітря в трубці закриває вимикач, тим самим підтверджуючи цей факт. Недоліком цієї системи є те, що вона чутлива до температури і ситуацій, коли декілька транспортних засобів одночасно активізували пневмодатчик. На рис. 1.8 подано приклад пневматичної системи трубок [42].

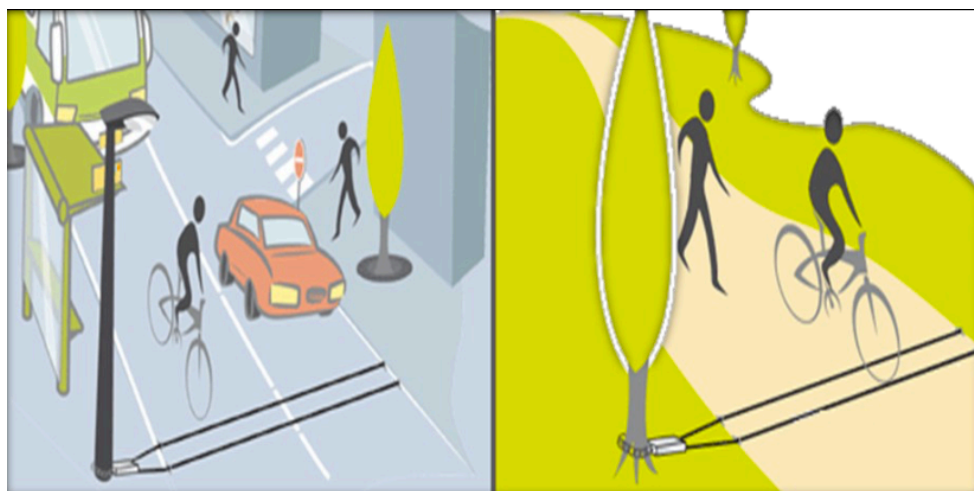


Рисунок 1.8 – Пневматичні трубки

У магнітних датчиках виявлення автомобілів вимірюються зміни в природньому магнітному полі, що викликані автомобілями. Це показано на рис 1.9, посередині показане магнітне поле, вироблене автомобілем з металу, а права сторона рисунку показує вплив цього на природнє магнітне поле. Рівняння для нього наведено у наступному:

$$B_z = B_{z^s} + \Delta B_z$$

де B_{z^s} - поле, що утворюється ізольованою котушкою, і ΔB_z – це зміна поля, викликана автомобілем [17].

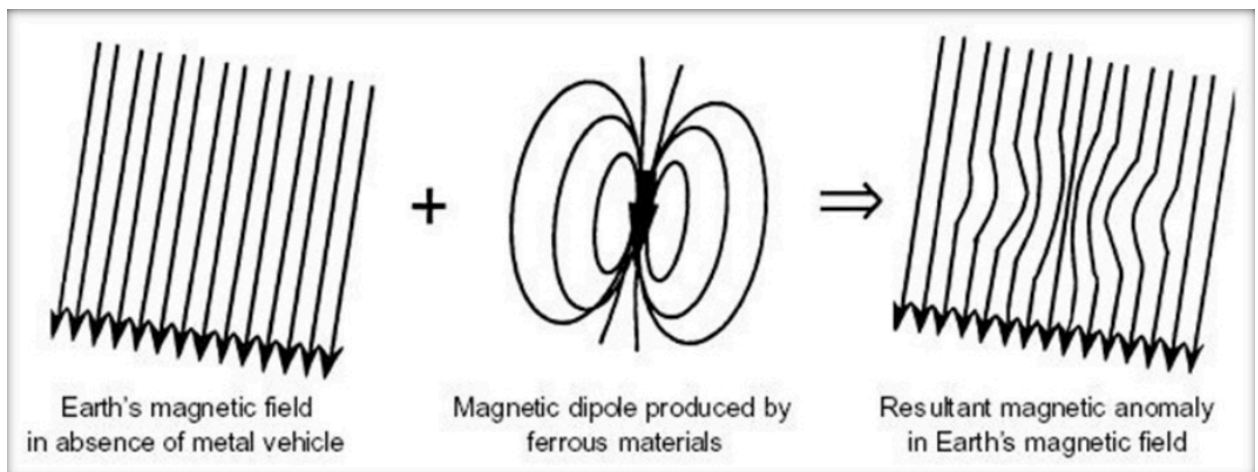


Рисунок 1.9 – Зміна природного поля металевим об'єктом

Магнітометричний датчик видає сигнал, що пропорційний силі магнітного поля навколо нього, як показано на рис. 1.10. Інший тип магнітометра використовує прилад, який називається датчиком флюсових вольт. Флюсовий накопичувач складається з магнітного ядра, яке періодично насичується першою котушкою, а друга котушка перебуває під дією сформованого електромагнітного поля.

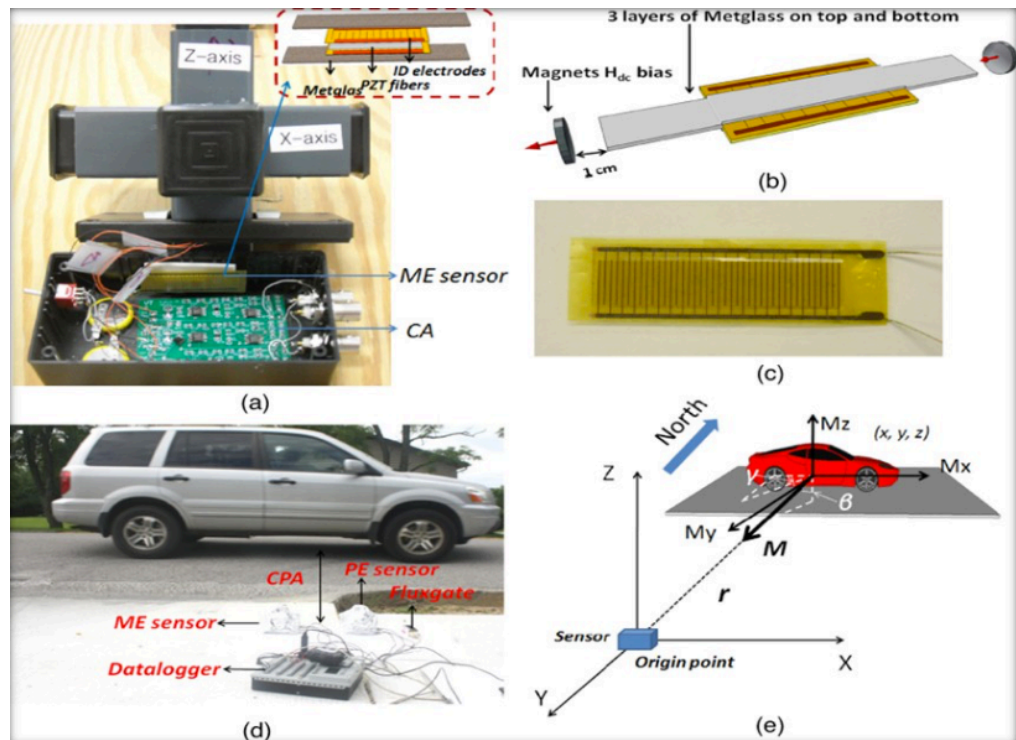


Рисунок 1.10 – Магнітометричні датчики

Вагові датчики використовуються в тому числі в системах зважування авто в русі. Такі системи складаються з п'єзоелектричного датчика, що вбудовано в спеціальний канал в дорожньому покритті. Ваговий датчик завжди поєднується з іншими системами, активними чи пасивними, що надають додаткові дані про характеристики авто [5].

На рис. 1.11 показано, що ваговий датчик складається з згинальної пластини, що тисне на п'єзоелемент, та системи передачі результатів вимірювання).

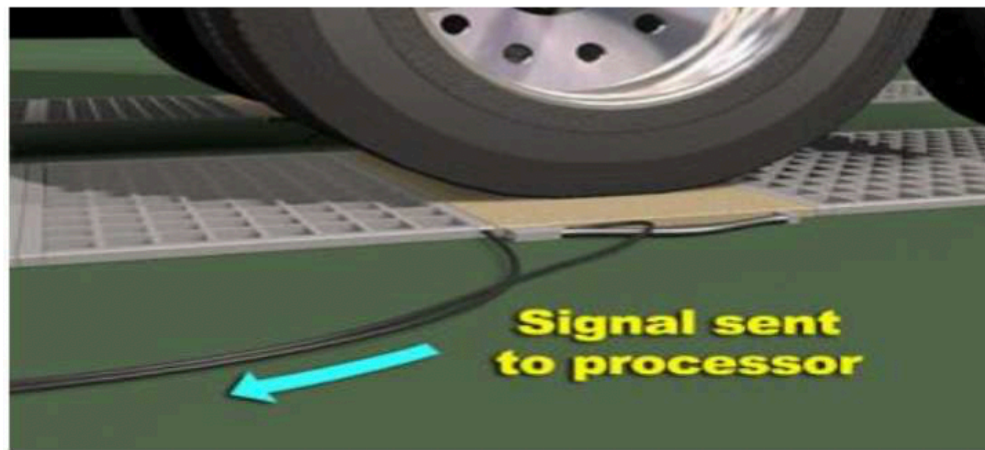


Рисунок 1.11 – Система детектора руху ваги [9]

1.3.2 Пасивні (неінтрузивні) датчики

Встановлення та обслуговування неінтрузивних датчиків не впливає на дорогу, за якою вони стежать. Це означає, що немає необхідності зупиняти рух автомобілів під час встановлення чи модифікації датчика. Типи неінтрузивних датчиків включають мікрохвильовий радіолокатор, пасивний акустичний масив, пасивний інфрачервоний, RFID та ультразвукові датчики. Зображення з відеокамери також може використовуватися для виявлення автомобілів [6].

Мікрохвильові радіолокаційні датчики випромінюють електромагнітний сигнал, що має частоту між 1–30 ГГц, і виявляє об'єкти через їх відбиття сигналу, як показано на рис. 1.12. Результати його вимірювань набагато точніші, ніж інших датчиків. При цьому на них не впливає дощ настільки, як інші датчики. Він може виявити транспортні засоби, що проїжджають по декількох смугах, та обчислити швидкість, з якою вони рухаються. Однак без додаткового датчика мікрохвильовий радіолокаційний датчик не може виявити нерухомі транспортні засоби [7].

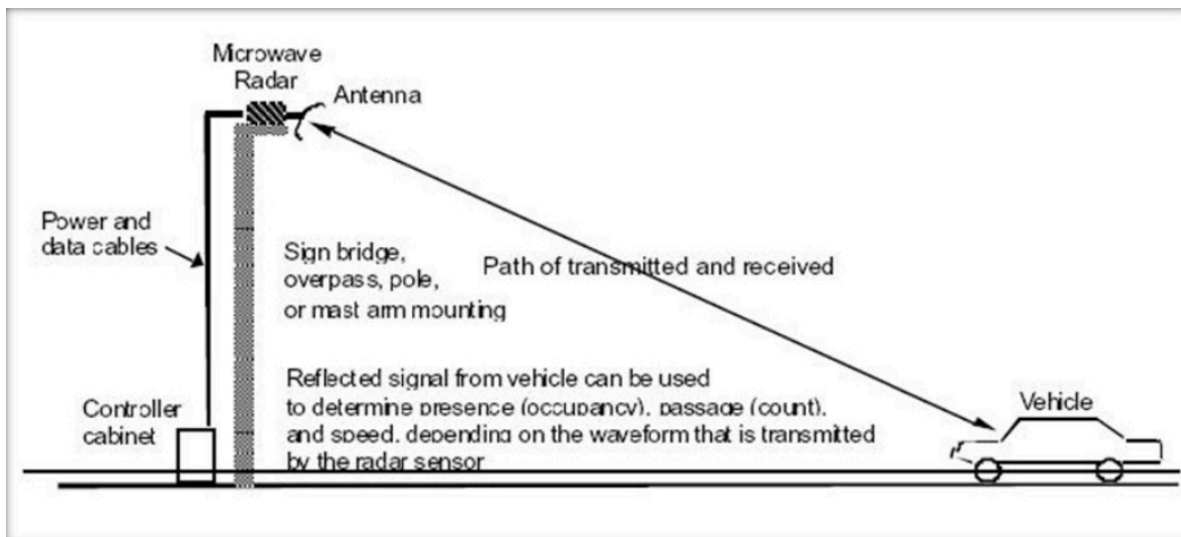


Рисунок 1.12 – Пояснення роботи мікрохвильового радіолокаційного датчика

Транспортні засоби виявляються за допомогою активних інфрачервоних датчиків так само, як і в мікрохвильового радіолокатора, через відбиття електромагнітного сигналу. Інфрачервоні сигнали мають значно нижчу частоту, ніж мікрохвильова піч, і більш чутливі до умов навколишнього середовища, таких як туман або сніг. Інфрачервоний датчик може випромінювати кілька променів, щоб більш точно визначити положення та швидкість руху автомобілів [8].

Ультразвукові датчики працюють за тим же принципом, що і радіолокаційні датчики НВЧ. Їх частота перевищує діапазон слуху людини, між 25 і 50 кГц.

$$p(x,w) = \frac{-i_w p}{2\pi} \int_S^1 v_z(y,w) \frac{\exp(i_k r)}{r} d_S(y)$$

де p - щільність середовища, w - кутова частота, v - швидкість в напрямку осі Z і $r = |x - y|$ - відстань між точкою оцінювання x та джерелом y на поверхні S . На рис. 1.13 показано зображення типової установки ультразвукової системи [9].

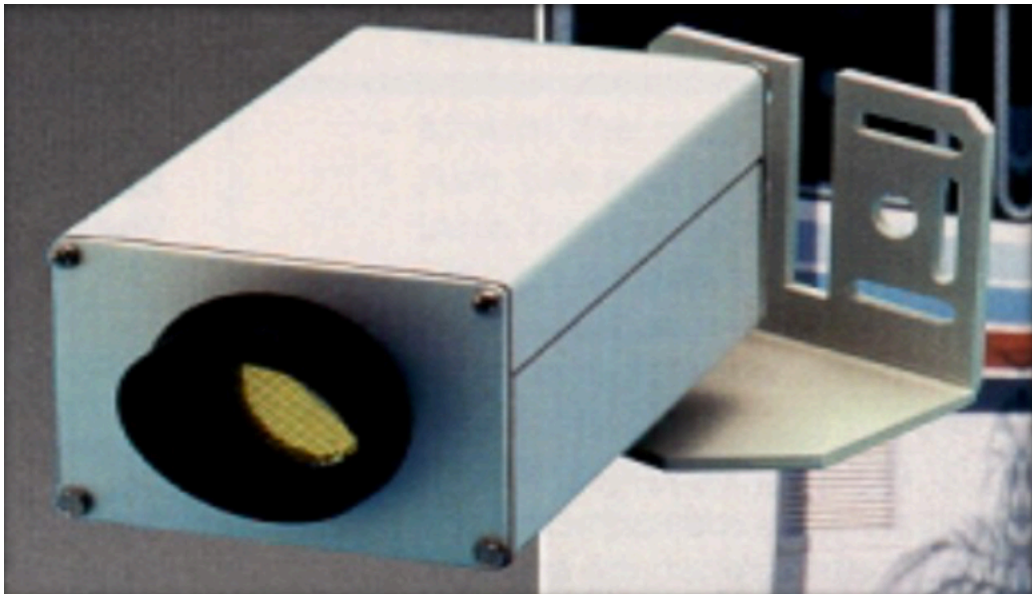


Рисунок 1.13 - Ультразвукові датчики

Для формування відеозображень в системах виявлення транспортних засобів використовуються одна або кілька камер. При цьому для обробки зображень, що зроблені камерами, необхідне спеціальне програмне забезпечення. Ця обробка зазвичай проводиться шляхом вивчення різниці між послідовними кадрами відео. На рис. 1.14 показано відстань виявлення, на якій система відео-виявлення автомобілю може розрізнити два і більше близько розташовані транспортні засоби на поверхні дороги. Ця відстань залежить від висоти кріплення камери, відстані між транспортними засобами або зазором та висоти транспортного засобу.

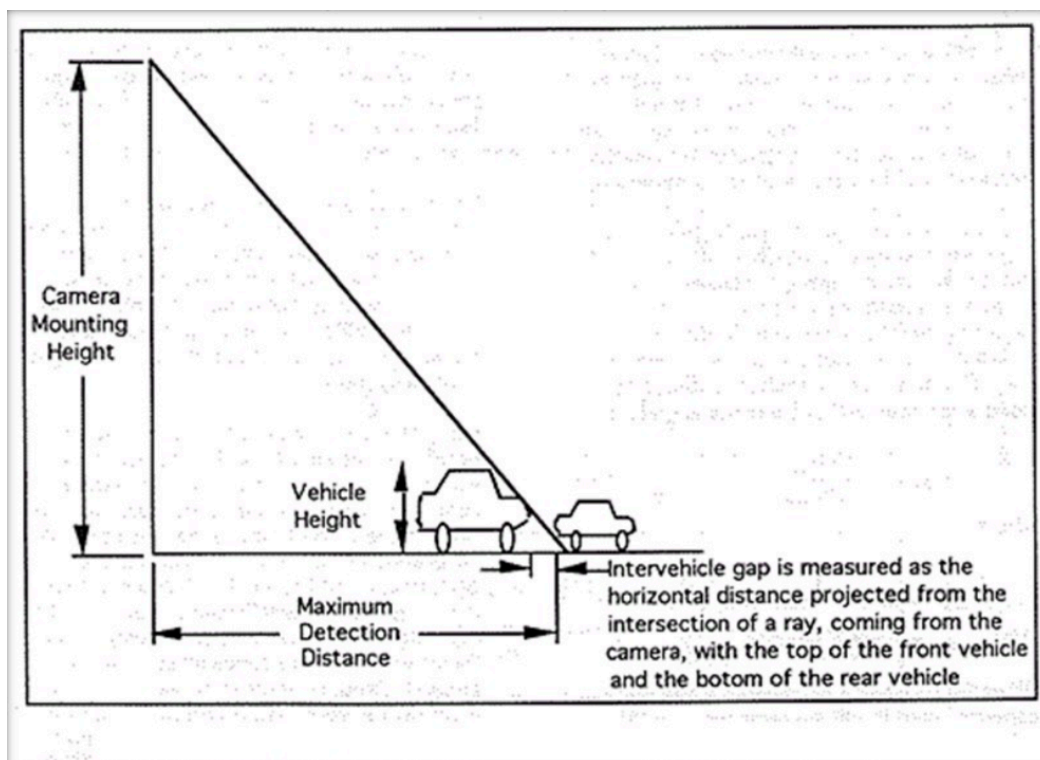


Рисунок 1.14 - Лінія геометрії виявлення процесора відеозображення [7]

Інші фактори, які слід враховувати при встановленні камер, що використовуються у системах система відео-виявлення, включають:

- вертикальний та бічний кути огляду,
- кількість спостережених смуг,
- стійкість щодо вітру та вібрації;
- якість зображення.

Переваги та недоліки активних і пасивних датчиків наведено в табл. 1.1

Таблиця 1.1 – Сильні та слабкі сторони інтрузивних та неінтрузивних сенсорних технологій [19]

Тип датчика	Переваги	Недоліки
Магнітометричні	Деякі типи підтримують бездротове радіо, роблячи їх портативними Витримує складні умови руху	Датчики потрібно перевстановлювати щоразу при ремонті дорожнього покриття. При встановленні в існуюче покриття робиться його надріз, що зменшує термін експлуатації.

Продовження табл. 1.1

Тип датчика	Переваги	Недоліки
Магнітні	Деякі типи не потребують надрізу в дорожньому покритті і можуть розміщуватися там, де проблематично встановити інші види датчиків Витримує складні умови руху	Не вдається виявити транспортні засоби, які є нерухомими. В більшості випадків при встановленні в існуюче покриття робиться його надріз, або прокладається тунель
Індуктивні	Можна обчислити щільність руху, відокремлювати автомобілі з незначною відстанню між ними та наявність транспортних засобів, що не рухаються. Технологія використовується давно, тому виробники мають великий досвід. Завдяки цьому існує багато конструкцій, які можна використовувати в різних додатках	Часто необхідно встановити більше одного пристрою для ефективного виявлення автомобілів Проблеми з дорожнім рухом і температура можуть вплинути на показники Для встановлення потрібно надрізати тротуар (така ж проблема, як і магнітометр)
Мікрохвильові	Можна виявити транспортні засоби, що рухаються по кількох смугах руху, та їх швидкість Зазвичай стійкі до складних природних умов	Транспортний засіб, що не рухається, неможливо виявити
Ультразвукові	Здатний виявляти транспортні засоби на більш ніж одній смузі	Великі періоди повторення імпульсів можуть погіршити результати вимірювання на автострадах із транспортними засобами, які рухаються із середньою та високою швидкістю. Деякі умови навколишнього середовища, такі як зміна температури та надзвичайна турбулентність повітря, можуть вплинути на ефективність. Компенсація температури вбудована в деякі моделі.

Продовження табл. 1.1

Тип датчика	Переваги	Недоліки
Інфрачервоні	<p>Доступна робота з кількома смугами. Багатозонні пасивні датчики вимірюють швидкість.</p> <p>Активний датчик передає кілька променів для точного вимірювання положення, швидкості та класу автомобіля.</p>	<p>Пасивний датчик, можливо, знижує чутливість до транспортних засобів у своєму полі зору у дощ та туман.</p>
Відео	<p>Легко додавати та змінювати зони виявлення.</p> <p>Забезпечує аналіз значної площі, коли інформація, зібрана в одному місці камери, може бути пов'язана з іншою. Доступний багатий масив даних. Здатен працювати з декількома смугами руху.</p>	<p>Погода, тіні, проекція транспортного засобу на сусідні смуги, оклюзія, перехід на день до ночі, контраст транспортного засобу / дороги, а також вода, соляний бруд, бурульки та павутинки на об'єктиві камери можуть впливати на продуктивність.</p> <p>економічно вигідний, якщо потрібно обробляти дані з декількох зон виявлення.</p>

1.4 Програмні засоби обробки сигналів і даних

Сигнали, що були виміряні датчиками, необхідно перетворити в цифрову форму, яку можуть використовувати комп'ютери. Багато сучасних датчиків мають компоненти, що виконують аналогово-цифрові перетворення. Часто декілька датчиків можуть бути підключені до мікроконтролера, міні-комп'ютера, який приймає свої дані та передає їх на комп'ютер, який виконує необхідні перетворення [21, 23].

Дані, зібрані з усіх датчиків, повинні бути оброблені центральним комп'ютером і потім перетворені в рекомендації водіям. Ці рекомендації включатимуть таку інформацію, як доступні автостоянки на ділянці автостоянки та маршрути проїзду транспортного засобу до наявної автостоянки [16, 18].

1.5 Постановка задачі

Мета цього дослідження - розробити інтелектуальну систему управління автостоянкою, що оптимізувати як процес паркування, так і спостереження за припаркованими автомобілями. Основним інтелектуальним компонентом такої системи буде підсистема визначення вільних паркувальних місць за зображеннями, що формуються підсистемою відеоспостереження автостоянки. Для досягнення поставлено мети необхідно:

- 1) Сформулювати вхідний математичний опис інтелектуальної системи управління автостоянкою.
- 2) Обрати математичну модель процесу визначення наявності автомобілів на паркувальних місцях.
- 3) Обрати обладнання, яке потрібно для фотографування автостоянки та відправлення зображень на комп'ютер за допомогою бездротової системи зв'язку.
- 4) Розробити та програмно реалізувати алгоритм визначення наявності автомобілів на паркувальних місцях
- 5) Провести тестування розробленої системи

2 ТЕХНОЛОГІЯ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ АВТОПАРКАМИ

2.1 Попередня обробка зображень

Перед обробкою та виявленням вільних паркувальних місць, потрібно попередньо підготувати шаблонне зображення на якому розмістимо зелені кола на позиції парко-місця. Підготовка зображення має п'ять кроків, які відображені на рис. 2.1

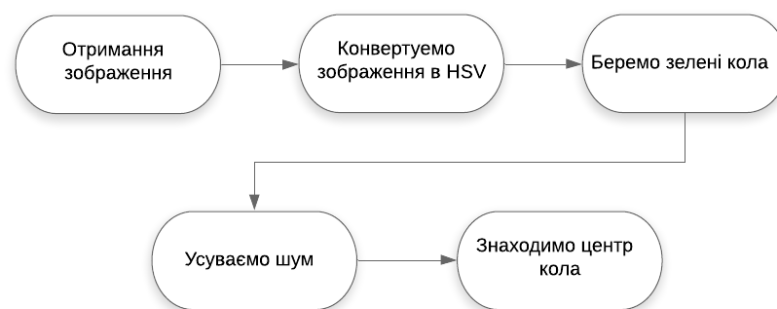


Рисунок 2.1 – Попередня обробка зображення

По-перше, зображення автостоянки можна зробити, коли на автостоянці немає автомобілів, як показано на рис 2.2. Це було б використано для того, щоб система записувала місце розташування всіх автостоянок. Значення RGB можна використовувати для визначення місця зелених кіл, що представляють порожні місця для автомобілів. Завдяки цьому система буде знати, де шукати машини в майбутньому.



Рисунок 2.2 – Шаблонне зображення

На порожній автостоянці для отримання місця розташування зелених крапок зображення перетворюється у зображення HSV, як показано на рис. 2.3. Це робиться за допомогою команди 'rgb2hsv' в Matlab.

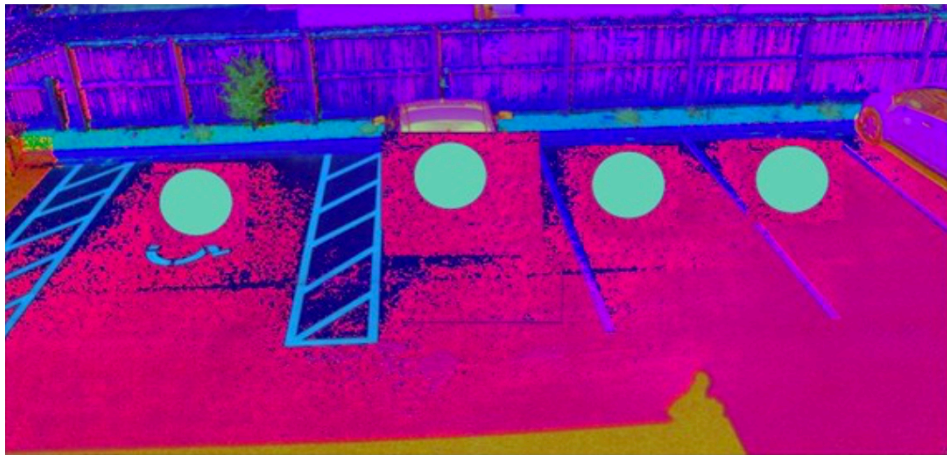


Рисунок 2.3 – Шаблонне зображення перетворене у зображення HSV
Зелені кола можна дістати, вибравши певні значення пікселів H, S та V.

У цьому випадку діапазон пікселів встановлюється таким чином, що тільки зелені пікселі будуть вилучені з зображення даного формату.

Процеси виявлення кола вимагають перетворення Хоффа. Перетворення Хоффа - це техніка вилучення особливостей, яка буде використовуватися для вилучення кіл, їх радіусів разом із їх центрами.

Перш за все, червоний сегмент зображення витягується, оскільки, як ми знаємо, червоні пікселі - це найпотужніший сегмент зображення, як показано на рис 2.4.



Рисунок 2.4 – Червоний сегмент зображення

Якщо в середовищі є об'єкти, які мають однакове значення червоних пікселів, які відобразатимуться і на витягнутому зображенні. Це буде шумом для витягнутого зображення і може створити неточний результат, який ви отримаєте при використанні трансформації Хоффа. Тому деталі, які не відповідають формі крапки або кола, на який розглядаються, слід обрізати.

Процес виявлення кола показаний у наступних пунктах:

- Створіть простір акумулятора відносно пікселів і встановіть ці значення на нуль
- Для кожного краю збільшуйте всі значення відповідно до рівняння кола $((i-a)^2 + (j-b)^2 = r^2)$
- Це дасть змінну "а" в рівнянні
- Тепер для всіх можливих значень "а" знайдіть значення "b", які відповідають рівнянню
- Пошук локальних максимумів. Це точки, які мають більшу ймовірність, ніж інші
- Ці максимуми - це розташування кіл
- Намалюйте коло, використовуючи команду `rect` у Matlab, щоб отримати результат, як на рис 2.5.

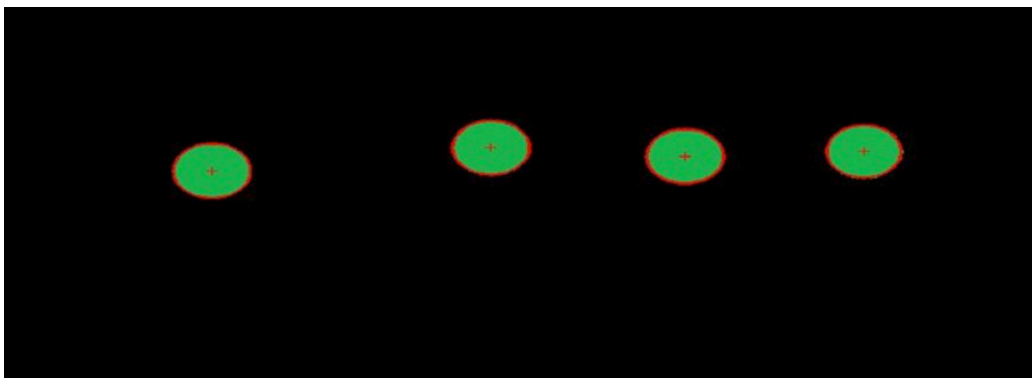


Рисунок 2.5 – Зелені кола, отримані при виявленні, обведені червоним, показуючи їх центр і середину

2.2 Порівняння методів виділення контуру на зображенні

Для реалізації системи комп'ютерного зору, яка дозволить розпізнавати наявність автомобіля на парко-місці, потрібно застосовувати один з методів виявлення контуру на зображенні. Таких методів реалізовано декілька, тому потрібно порівняти їх та обрати найбільш підходящий.

Досить часто аналіз контуру об'єкта на зображенні виконується в медичних дослідженнях. Так в роботі [8] наведені результати виділення контурів на медичних зображеннях. Автор порівняв основні сучасні методи (алгоритми): Canny, Prewitt, Roberts, Sobel, LoG ((Laplacian Of Gaussian)), К-середніх. Для обраних зображень найкращі результати показали алгоритми Canny, LoG і К-середніх, тому що вони оконтурити лінію всередині об'єкта. Також при обробці зображення алгоритмом Canny автор спостерігав найменшу ступенчатість контурів об'єкта, в порівнянні з іншими алгоритмами.

У публікації [9] автори досліджують такі ж методи як в попередній роботі, за винятком методу К-середніх. Основне завдання роботи - отримання бінарного зображення, що містить замкнуті структурні області об'єктів на зображенні, якими на медичному зображенні можуть бути кордону органів, вени, а також пухлини. Автори встановили, що для використовуваних зображень при низькому рівні шуму алгоритм Canny зі стандартною шириною порогових рівнів має найкращий результат. Зі збільшенням порогу чутливості і відповідно рівня шуму найменші показники значення відносного числа помилково певних контурних пікселів показали алгоритми Roberts і Prewitt, причому при рівні шуму понад 8% це значення у них приблизно однаково. Найгірший результат показав алгоритм LoG.

Також виділення контуру актуально для інших прикладних задач. В роботі [11] розглянута проблема розпізнавання контуру долоні на складних зображеннях. Для цієї мети досліджуються методи Sobel і Canny, але виходячи з представлених результатів, в пріоритеті задача сегментації. Для досліджуваних

зображень автор визначив, що метод Canny працює швидше, ніж метод Sobel, але більш чіткі межі об'єктів на зображенні виходять

при обробці на основі застосування оператора Sobel.

Розглянемо приклад з порівнянням візуального виявлення межі каплі метала на рис 2.6. На зображенні видно на скільки деталізований результат дає метод Canny. В деяких випадках потрібно отримати лише конкур об'єкту і з цим добре справляються всі інші методи, хоча щоб отримати такий же результат з методом Canny потрібно лише зменшити контрастність зображення. В протилежному випадку, коли потрібно отримати максимальну кількість деталей інші методи з цим не впораються.

Исходное изображение			
Метод Canny			
Метод Prewitt			
Метод Roberts			
Метод Sobel			
Метод Laplacian of Gaussian			

Рисунок 2.6 – Приклад з порівнянням візуального виявлення межі каплі метала

Опираючись на проведений аналіз в інших роботах можна сказати, що ідеального методу не існує і метод потрібно обирати в залежності від цілі обробки та мети. Саме для реалізації інтелектуальної системи підрахунку заповнення автостоянки був обраний метод Canny тому, що фокус ставився на швидкість отримання результатів та порівняно висока ефективність даного методу.

2.3 Метод оператора Canny

Виявлення ребра Canny широко застосовується в різних системах комп'ютерного зору. Canny встановив, що вимоги до застосування виявлення країв у різних системах зору відносно однакові. Таким чином, рішення для виявлення краю для вирішення цих вимог може бути реалізовано в широкому діапазоні ситуацій. Загальні критерії виявлення ребер включають:

- Виявлення повинно точно вловлювати якомога більше ребер, зображених на зображенні
- Крайова точка, виявлена у оператора, повинна точно локалізуватися в центрі краю.
- Даний край на зображенні повинен бути позначений лише один раз, і коли це можливо, шум зображення не повинен створювати помилкові краї.

Серед розроблених до цього часу методів виявлення ребер алгоритм виявлення ребра Canny є одним із найбільш строго визначених методів, що забезпечує хороше і надійне виявлення. Завдяки своїй оптимальності відповідати трьом критеріям виявлення ребер та простоті процесу впровадження, він став одним із найпопулярніших алгоритмів виявлення ребер. Алгоритм виявлення ребра Canny можна розділити на 5 різних етапів:

1. Застосувати фільтр Гаусса, щоб згладити зображення, щоб зняти шум
2. Знайти градієнти інтенсивності зображення

3. Застосовати не максимальне придушення, щоб позбутися хибної реакції на виявлення ребер
4. Застосовати подвійний поріг для визначення потенційних ребер
5. Відстеження краю за допомогою гістерезису: Завершити виявлення країв, придушивши всі інші ребра, які слабкі та не з'єднані із сильними краями.

2.3.1 Гауссовий фільтр

Оскільки на результати виявлення ребер легко впливає шум на зображенні, важливо відфільтрувати шум, щоб запобігти помилковому виявленню, викликаному ним. Щоб згладити зображення, ядро фільтра Гаусса поєднується із зображенням. Цей крок трохи згладить зображення, щоб зменшити вплив очевидного шуму на крайовий детектор. Рівняння для ядра фільтра Гаусса розміром $(2k + 1) \times (2k + 1)$ задається:

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i - (k + 1))^2 + (j - (k + 1))^2}{2\sigma^2}\right); 1 \leq i, j \leq (2k + 1)$$

Ось приклад фільтра Гаусса 5×5 , який використовується для створення сусіднього зображення, з $\sigma = 1$.

$$\mathbf{B} = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * \mathbf{A}.$$

Важливо розуміти, що вибір розміру ядра Гаусса вплине на продуктивність детектора. Чим більший розмір, тим менша чутливість детектора до шуму. Крім того, помилка локалізації для виявлення краю трохи збільшиться зі збільшенням розміру ядра фільтра Гаусса. 5×5 - це хороший розмір для більшості випадків, але це також буде залежати від конкретних ситуацій.

2.3.2 Пошук градієнта інтенсивності зображення

Край на зображенні може вказувати в різних напрямках, тому алгоритм Canny використовує чотири фільтри для виявлення горизонтальних, вертикальних та діагональних ребер у розмитому зображенні. Оператор виявлення ребер повертає значення для першої похідної в горизонтальному напрямку (G_x) і вертикальному напрямку (G_y). З цього можна визначити градієнт ребра та напрямок:

$$G = \sqrt{G_x^2 + G_y^2} \quad \Theta = \arctg\left(\frac{G_y}{G_x}\right).$$

де G можна обчислити за допомогою функції гіпота, а atan2 - функцію арктангенту з двома аргументами. Кут напрямку ребра округлюється до одного з чотирьох кутів, що представляють вертикальний, горизонтальний та два діагоналі (0° , 45° , 90° і 135°). Напрямок краю, що падає у кожен кольорову область, буде встановлено на певні значення кута, наприклад, θ у $[0^\circ, 22,5^\circ]$ або $[157,5^\circ, 180^\circ]$ картах до 0° .

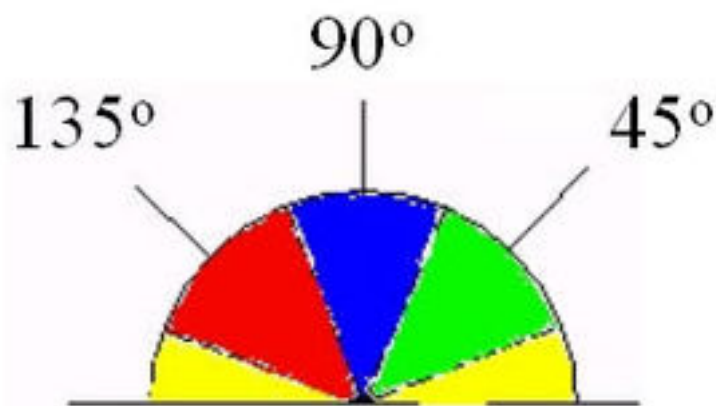


Рисунок 2.7 – Направлення градієнта

2.3.3 Немаксимальне придушення

Немаксимальне придушення - це техніка розрідження краю. Немаксимальне придушення застосовується для пошуку "найбільшого" краю. Після застосування розрахунку градієнта край, витягнутий із значення градієнта,

все ще досить розмитий. Щодо критерію 3, то має бути лише одна точна відповідь на край. Таким чином, не максимальне придушення може допомогти придушити всі значення градієнта (встановивши їх на 0), крім локальних максимумів, які вказують місця з найбільш різкою зміною значення інтенсивності. Алгоритм кожного пікселя в градієнтному зображенні:

- Порівняйте силу краю поточного пікселя з силою краю пікселя у позитивному та негативному напрямках градієнта.
- Якщо сила краю поточного пікселя найбільша порівняно з іншими пікселями у масці з тим же напрямком (наприклад, піксель, який спрямований у напрямку u , буде порівняно з пікселем над і під ним у вертикальній осі v), значення буде збережено. Інакше значення буде придушене.

У деяких реалізаціях алгоритм класифікує безперервні градієнтні напрямки на невеликий набір дискретних напрямків, а потім переміщує фільтр 3×3 над виходом попереднього кроку (тобто міцність краю та напрямки градієнта). На кожному пікселі він придушує силу краю центрального пікселя (встановивши його значення 0), якщо його величина не більша за величину двох сусідів у напрямку градієнта. Наприклад,

- якщо кут закругленого градієнта дорівнює 0° (тобто край знаходиться в напрямку північ-південь), то точка вважатиметься на краю, якщо її величина градієнта більша за величини пікселів у східному та західному напрямках,
- якщо кут закругленого градієнта дорівнює 90° (тобто край знаходиться у напрямку схід-захід), то точка вважатиметься на краю, якщо його величина градієнта більша за величини пікселів у північному та південному напрямках,
- якщо округлений кут градієнта становить 135° (тобто край знаходиться в північно - східному-південно - західному напрямку) точка буде

- вважатися на краю , якщо її градієнт величина більше величин в пікселях в північно - заході і південно-східних напрямках
- якщо кут закругленого градієнта 45° (тобто край знаходиться в напрямку північ захід - південь) точку вважатимуть на краю, якщо його величина градієнта більша за величини пікселів на північному сході та південному заході напрямки.

2.3.4 Подвійний поріг

Після застосування не максимального придушення решта крайових пікселів забезпечують більш точне зображення реальних країв на зображенні. Однак залишаються деякі крайові пікселі, які спричинені шумом та різницею кольорів. Для обліку цих помилкових відповідей важливо відфільтрувати крайові пікселі зі слабким значенням градієнта та зберегти крайові пікселі з високим значенням градієнта. Це досягається шляхом вибору високих та низьких порогових значень. Якщо значення градієнта крайового пікселя вище високого порогового значення, воно позначається як сильний крайовий піксель. Якщо значення градієнта крайового пікселя менше високого порогового значення і більше низького порогового значення, воно позначається як слабкий крайовий піксель. Якщо значення крайового пікселя менше, ніж значення низького порогового значення, воно буде придушене.

2.3.5 Відстеження країв за допомогою гістерезису

Поки сильні крайові пікселі, безумовно, повинні бути залучені до кінцевого зображення краю, оскільки вони витягуються з справжніх ребер зображення. Однак, будуть пізні дискусії щодо слабких пікселів, оскільки ці пікселі можуть бути витягнуті з справжнього краю, або відхилення від шуму / кольору. Щоб досягти точного результату, слід видалити слабкі краї, викликані останніми причинами. Зазвичай слабкий піксельний край, викликаний

справжніми краями, буде підключений до сильного крайового пікселя, коли відповіді на шум не підключені. Для відстеження крайового з'єднання застосовується аналіз блобу, дивлячись на слабкий крайовий піксель та його 8-з'єднані пікселі сусідства. Поки є один сильний крайовий піксель, який бере участь у краплі, ця слабка крайова точка може бути ідентифікована як та, яку слід зберегти.

3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ АВТОПАРКАМИ

3.1 Опис вхідних даних

Вилучимо зображення к камери відеоспостереження однієї з парковок. Загалом автостоянки мають схожу дорогу та розмітку, будемо вважати її як фон. Фон буде програмно вилучатись задля легкого виявлення лише автомобілів. Техніка обробки зображень по краях має п'ять кроків [13], як вони показані на рис 3.1.

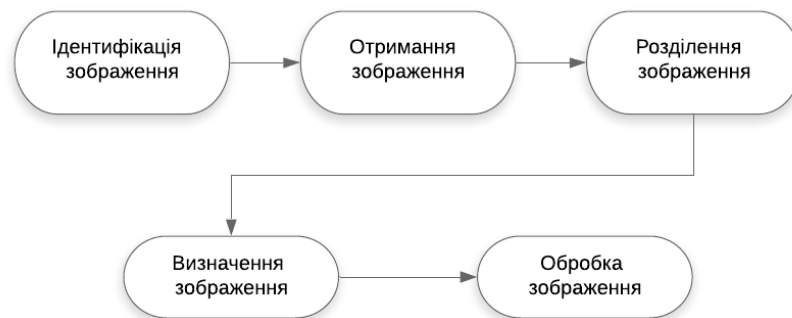


Рисунок 3.1 – Попередня обробка зображення

За зразок тестового зображення візьмемо те, яке зображене на рис. 3.2



Рисунок 3.2 – Обробка зображення

По-перше, зображення перетворюється на HSV, оскільки в просторі HSV легко диференціювати різні кольорові пікселі. На рис 3.3 добре видно, що червоні кольори є фоном.



Рисунок 3.3 – Зображення паркування, перетворене у зображення HSV

Переглядаючи пікселі на рис 3.3, стає зрозуміло, що пікселі для фону потрапляють у визначений діапазон кольорів. Наступним кроком потрібно сегментувати зображення щоб чітко відрізнити автомобіль та фон, тобто дорогу, як на рис 3.4



Рисунок 3.4 – Сегментоване зображення

Наступним кроком перетворюємо на сіре зображення, щоб отримати результат, як показано на рис 3.5

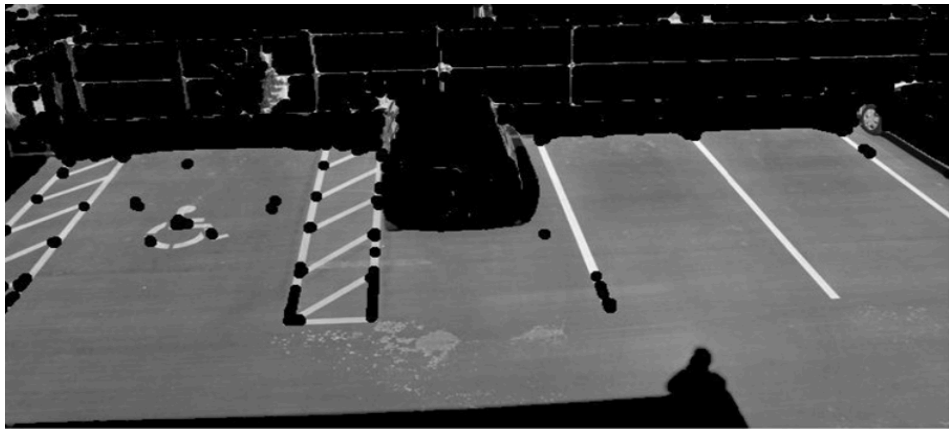


Рисунок 3.5 – Сіре сегментоване зображення

Це нелінійний процес, пов'язаний із формою та морфологією зображення. Це відносно впорядкування значення пікселів, а не їх числових значень, як показано на рис. 3.6

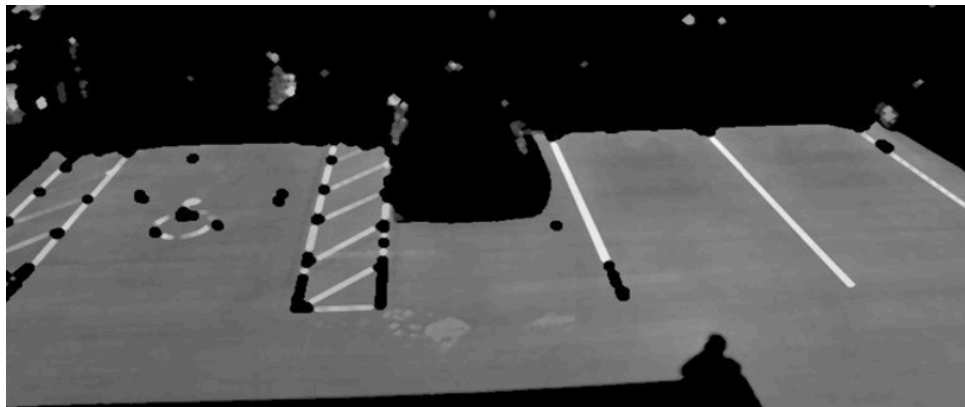


Рисунок 3.6 – Зображення після морфологічного процесу

Потім зображення перетворюється на бінарне зображення, як показано на рис 3.7



Рисунок 3.7 – Бінарне зображення

3.2 Короткий опис програми

За рахунок аналізу шаблонного зображення та зображення отримане із камери, система здатна виявляти зайняті та вільні місця. Це відбувається наступним чином: шаблонне зображення містить в собі кола з радіусом, які розташовані на місцях, де повинні знаходитися автомобілі. Після попередньої обробки зображення з камери тепер містить зображення дороги та усунуло будь-які зайві об'єкти. Далі алгоритм порівнює значення пікселів у обробленому зображенні, на місцях розташування кіл у шаблонного. Якщо друге зображення містить в загальній масі білі пікселі, тоді місце порожнє, в протилежному випадку, коли містить чорні пікселі - місце зайняте. На рис. 2.8, бачимо приклад такого порівняння:

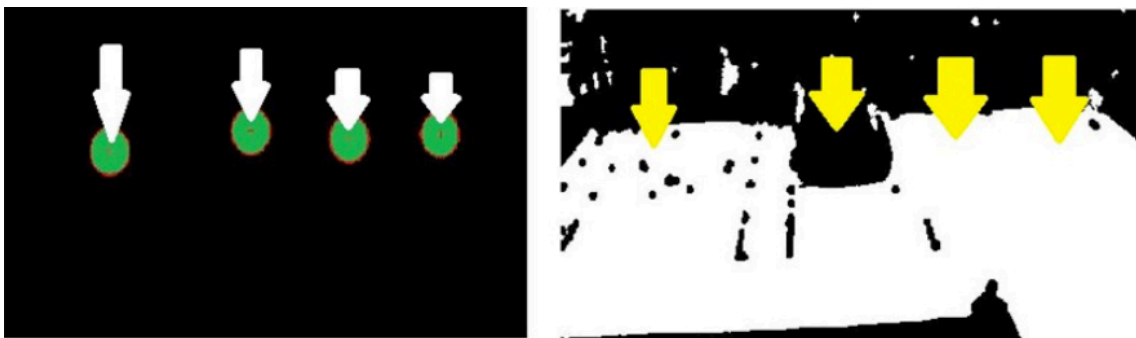


Рисунок 3.8 – Порівняння значень пікселів на місцях розташування кіл

Після порівняння зображень виведемо фігуру синього кольору для відображення вільного місця. Приклад такого відображення можна побачити на рис. 3.9:

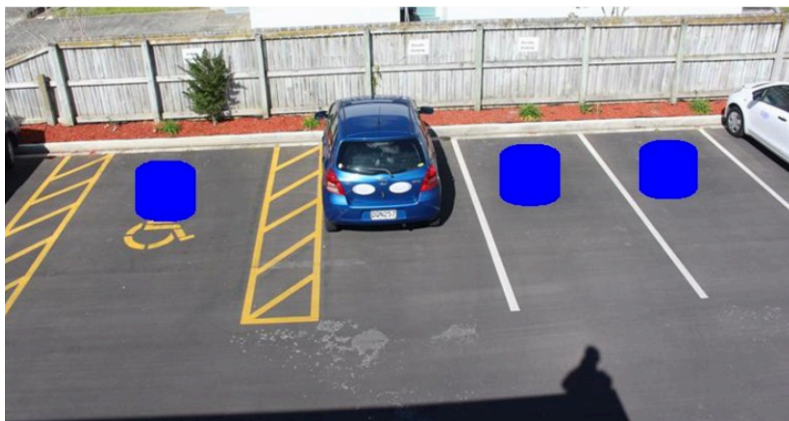


Рисунок 3.9 – Відображення вільних місць

Для реалізації даної системи Matlab використовувався для написання функцій обробки зображень. Оглянемо функції, які використовуються в реалізації даної системи.

Таблиця 3.1 – Огляд функцій використаних при реалізації

Назва функції	Опис
strel	Створює структурируючий елемент у формі диска, з можливістю вказати радіус кількість елементів лінійної структуризації, використовуваних для апроксимації форми диска.
Cat	Об'єднання масивів
plot	Побудова лінійних графіків. Використовувалась для відображення проміжних результатів
imfill	Виконує операцію заливки на фонових пікселях вхідного бінарного зображення, починаючи з точок, зазначених у місцях розташування.
uint8	Перетворює OPC HDA масив об'єктів даних DObj в матрицю uint8.
round	Округлення до найближчого десятка чи цілого числа
rectangle	Створення прямокутник з гострими або вигнутими кутами
flipud	Переставляє рядки масиву A симетричні відносно горизонтальної осі. Якщо масив A має непарне число рядків, то середній рядок залишається на своєму місці
pcolor	Створює псевдокольоровий графік, використовуючи значення в матриці
rgb2hsv	Перетворює червоні, зелені та сині значення RGB-зображення у відтінки, значення зображення HSV.
rect	Використовується для створення кіл

Зразок коду з обробки зображення знаходиться в додатку.

3.3 Тестування системи

Точність системи вимірюється зображеннями, зробленими на різні інтервали часу. Продуктивність обчислюється за порівнюючи результати зайнятості з основною істиною після кожні 5 сек. Продуктивність запропонованої системи є вимірюється за допомогою рівняння:

ЗКПМ = Загальна кількість паркувальних місць

ФКА = фактична кількість автомобілів

ПКА = Прогнозована кількість автомобілів

Продуктивність = $1 - (| \text{ФКА} - \text{ПКА} |) / \text{ЗКПМ} * 100$

Відсоток помилок у запропонованій системі складе

знайти за допомогою рівняння:

Відсоткова помилка = $(| \text{ФКА} - \text{ПКА} |) / \text{ЗКПМ} * 100$

Результат одного з тестових запусків можна побачити на наступних зображеннях. Рис. 3.10 є тестовим зображенням взятим з камери в хмарних день з гарною видимістю, а рис. 3.11 – результат аналізу зображення, на якому відмічене вільне місце.



Рисунок 3.10 – Приклад зображення взятим з камери в хмарних день з гарною видимістю

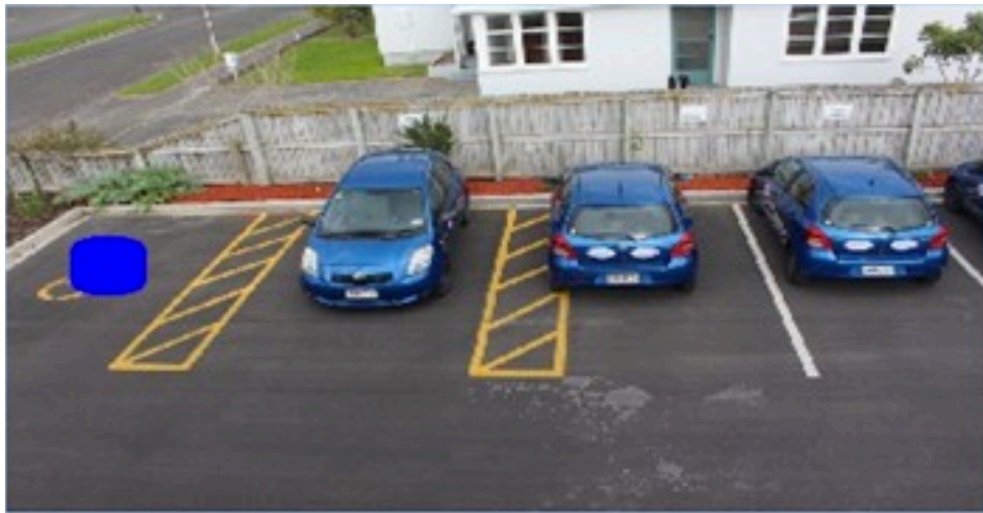


Рисунок 3.11 – Результат аналізу зображення

Проводиться тестування запропонованої методики на модельній парковці, що має загальні місця для паркування ЗКПМ = 4. Тест проводиться в різних погодних умовах як показано в таблиці 3.2.

Таблиця 3.2 – Результати тестування на знаходження вільних місць

	Видимість	Номер тестів	Правильно виявлення	Помилковий виявлення	Точність
Сонячний день	Гарна	50	50	0	100%
	Погана	50	48	2	96%
Хмарний день	Гарна	50	49	1	98%
	Погана	50	46	4	94%

Виявлено точність запропонованого алгоритму 100%, 98% 96% і 94%. Результати показують, що при захопленні зображення парковки незрозумілі через менше освітлення або оклюзій, ефективність знижується і точність виявлення знижується. Помічено, що середня продуктивність становить 99,5% і дуже висока порівняно з іншими програмами виявлення місць на стоянці. Продуктивність запропонованого методу відображається на рис. 3.12, на якому синім кольором відображається кількість тестів, а помаранчевим – кількість

успішних підрахунків. На графіку бачимо залежність результатів від погодних умов: на сонячну погоду с гарною видимістю система дає максимальну результативність, а прі хмарній з поганою видимістю – найнижчу. В умовах підземної парковки за умов наявності гарного освітлення система даватиме стабільно гарну результативність.

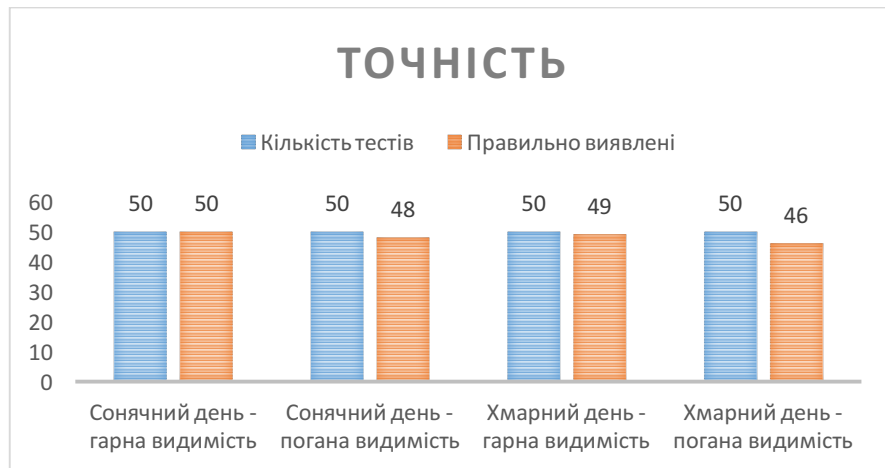


Рисунок 3.12 – Результати тестування точності методу

ВИСНОВОК

В даній роботі було проведено аналіз існуючих типів автоматизації автомобільної парковки: описано принцип роботи, переваги та недоліки кожного з них, а також пристроїв, які використовуються при обладнанні цих типів парковок.

Узагальнюючи, можна сказати що існує чотири основні типи систем, що використовуються при керуванні автостоянкою:

- система, що базується на лічильниках, в якій підраховуються машини, що входять і виходять з будівлі;
- система на основі зображень, де зроблено зображення автомобілів, що знаходяться на автомобільних стоянках;
- ультразвукова система, що має датчики на кожному місці стоянки автомобіля, підключені до центрального процесорного блоку;
- бездротові датчики, які мають вузли, які беруть дані від декількох датчиків і передають їх процесору.

Для виявлення автомобілів, які в'їжджають і залишають автостоянку, можна використовувати датчики виявлення вторгнення (IDS). Дані з цих датчиків необхідно обробити та надіслати на дисплеї, які надають водіям необхідну інформацію (наприклад, кількість доступних паркувальних місць на рівні автостоянки).

Кожна технологія детектора та конкретний пристрій має свої обмеження та індивідуальні можливості. Успішне застосування детекторних технологій багато в чому залежить від правильного вибору пристрою. На вибір детекторів впливає багато факторів, такі як тип даних, точність даних, простота установки, вартість та надійність.

Для виявлення присутності автомобілів на парковці будуть використовуватись зображення камери на автостоянці. В даній реалізації

системи підрахунку кількості доступних місць був застосований метод оператора Canny. Для реалізації програмного забезпечення було використовується середовище для наукових та інженерних розрахунків Matlab. Дана реалізація є чудовим варіантом для автостоянок без жодних автоматизацій, яка не буде потребувати додаткових інвестицій оскільки на більшості автостоянках присутнє відеоспостереження. З плином часу дану систему можна буде доповнити іншими датчиками, які були оглянуті в даної роботі. Вони зможуть доповнити функціонал та ще більше зменшити відсоток людино-часів затрачених в обслуговуванні клієнтів автостоянки.

СПИСОК ЛІТЕРАТУРИ

1. Л. Д. Ярошук., Інтелектуальні системи управління. Експертні системи – основи проектування та застосування в системах автоматизації, 2019.
2. П'ятикоп О.Є., Левицька Т.О., Тельних Л.В. «Порівняння методів виділення контуру для зображення краплі металу», Вісник приазовського державного технічного університету, Вип.33 2016 р.
3. H. Al-kharusi and I. Al-Bahadly, Intelligent Parking Management System Based on Image Processing,” World Journal of Engineering and Technology, Vol.2, Issue 2, 2014.
4. Кваско, М. З., Жураковський Я. Ю., Комп'ютерні методи проектування систем автоматизації, 2014.
5. Power D. J. Web-based and model-driven decision support systems: concepts and issues. Americas Conference on Information Systems, Long Beach, California, 2000.
6. Lin, S., Chen, Y. and Liu, S, A Vision-Based Parking Lot Management System. IEEE International Conference on Systems, Man, and Cybernetics, Taipei, 2006, 2897-2902.
7. Вінцюк Т.К., Сажок М.М., Селюх Р.А., Федорин Д.Я., Юхименко О.А., Робейко В.В. Автоматичне розпізнавання, розуміння та синтез мовленнєвих сигналів в Україні. Управляющие системы и машины. 2018. № 6. С. 7-24.
8. Ель-Хатіб С.А. Порівняльний аналіз алгоритмів виділення контурів медичних зображень / С.А. Ель-Хатіб, Ю.А. Скобцов // Тези доповідей II-ї міжнародній науково-технічній конференції студентів, аспірантів і молодих вчених «Інформаційні управляючі системи та технології і комп'ютерний моніторинг». - Донецьк, 2011. - Т. 2. - С. 52-56.

9. Методи аналізу медичних зображень / А.В. Доронічева, Н.Е. Косих, С.К. Полумієнко, С.З. Савін // Екологічна безпека та природокористування. - 2014. - Вип. 16. - С. 148 - 170.
10. Бритик В.І. Дослідження можливостей різних фільтрів і їх застосування в задачах розпізнавання образів / В.І. Бритик, Є.Ю. Жиліна // Біоніка інтелекту: наукотехнічний журнал. - 2014. - № 2 (83). - С. 88-95.
11. Шевченко Е.А. Завдання розпізнавання контуру долоні на складних зображеннях / Е.А. Шевченко // Штучний інтелект. - 2013. - № 4. - С. 244-251.
12. Riza, N.A., Marraccini, P.J. and Baxley, C.R. Data Efficient Digital Micromirror Device-Based Image EdgeDetection Sensor Using Space-Time Processing. IEEE Sensors Journal, 2012, 1043-1047
13. Zilan, R., Barceló-Ordinas, J.M. and Tavli, B. (2008) Image Recognition Traffic Patterns for Wireless Multimedia Mobility in Next Generation Internet, 4th International Workshop of the EuroNGI/EuroFGI Network of Excellence, Barcelona, 16-18 January 2008, 49-59.
http://link.springer.com/chapter/10.1007%2F978-3-540-89183-3_5
14. И. Свирин, А. Ханин, Некоторые аспекты автоматического распознавания автомобильных номеров, "Алгоритм Безопасности" № 3, 2010
15. Панкратов А.А. Анисимова Г.Б, Проектирование информационной системы оптимизации работы автостоянки, «Компьютерные и информационные науки», 2018
16. Souissi, R., Cheikhrouhou, O., Kammoun, I. and Abid, M. A Parking Management System Using Wireless Sensor Networks. 2011 International Conference on Microelectronics (ICM), Hammamet, 2011, 1-7
17. Верещагина Е.А., Рудниченко А.К., Разработка клиент-серверной информационной системы учёта автомобильных пропусков с

- использованием информации из устаревшей базы данных, «Компьютерные и информационные науки», 2018
18. Gu, J.; Zhang, Z.; Yu, F.; Liu, Q. Design and implementation of a street parking system using wireless sensor networks. In Proceedings of the 10th IEEE International Conference on Industrial Informatics (INDIN), Beijing, China, 25–27 July 2012; pp. 1212–1217.
 19. Nawaz, S.; Efstratiou, C.; Mascolo, C. Parksense: A smartphone based sensing system for on-street parking. In Proceedings of the 19th Annual International Conference on Mobile Computing and Networking, Miami, FL, USA, 30 September– 4 October 2013; pp. 75–86.
 20. Rose, R. A smart technique for determining base-station locations in an urban environment. *IEEE Trans. Veh. Technol.* 2001, 50, 43–47.
 21. Han, J.K.; Park, B.S.; Choi, Y.S.; Park, H.K. Genetic approach with a new representation for base station placement in mobile communications. In Proceedings of the 2001 IEEE Vehicular Technology Conference, Atlantic City, NJ, USA, 7–11 October 2001; Volume 4, pp. 2703–2707.
 22. Meunier, H.; Talbi, E.; Reininger, P. A multiobjective genetic algorithm for radio network optimization. In Proceedings of the 2000 Congress on Evolutionary Computation, La Jolla, CA, USA, 16–19 July 2000; Volume 1, pp. 317–324.
 23. Amaldi, E.; Capone, A.; Malucelli, F.; Signori, F. UMTS radio planning: Optimizing base station configuration. In Proceedings of the 2002 IEEE Vehicular Technology Conference, Vancouver, BC, Canada, 24–28 September 2002; Volume 2, pp. 768– 772
 24. Barton, J., Buckley, J., O’Flynn, B., O’Mathuna, S.C., Benson, J.P., O’Donovan, T., Roedig, U. and Sreenan, C. (2007) The D-Systems Project—Wireless Sensor Networks for Car-Park Management. *IEEE 65th Vehicular Technology Conference*, Dublin, 22-25 April 2007, 170-173.
 25. Zou, Y.; Chakrabarty, K. Sensor deployment and target localization based on virtual forces. In Proceedings of the Twenty-Second Annual Joint Conference of

- the IEEE Computer and Communications Societies (INFOCOM 2003), San Francisco, CA, USA, 30 March–3 April 2003 ; Volume 2, pp. 1293–1303.
26. Fonseca, C.M.; Fleming, P.J. Genetic algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In Proceedings of the Fifth International Conference on Genetic Algorithms, Urbana-Champaign, IL, USA, 17–22 July 1993; Volume 93, pp. 416–423.
27. Jourdan, D.B.; de Weck, O.L. Layout Optimization for a Wireless Sensor Network Using a Multi-Objective Genetic Algorithm. In Proceedings of the 2004 59th IEEE Vehicular Technology Conference, Milan, Italy, 17–19 May 2004; Volume 5, pp. 2466–2470.
28. Jourdan, D.B.; de Weck, O.L. Multi-Objective Genetic Algorithm for the Automated Planning of a Wireless Sensor Network to Monitor a Critical Facility. In Proceedings of the SPIE, Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense III, Orlando, FL, USA, 12–16 April 2004; Volume 5403, pp. 565–575.

ДОДАТОК

Частина коду по конвертації зображення в HSV формат

```
hue = (Img_HSV(:,:,1) >= 0) & (Img_HSV(:,:,1) <= .9);  
saturation = (Img_HSV(:,:,2) >= .01) & (Img_HSV(:,:,2) <= .1);  
value = (Img_HSV(:,:,3) >= .2) & (Img_HSV(:,:,3) <= 0.55);  
Temp = uint8(hue & saturation & value);
```

```
Temp = imclose(Temp, strel('disk', 5));  
Red_pixel = Temp .* Org_Img(:,:,1);  
Green_pixel = Temp .* Org_Img(:,:,2);  
Blue_pixel = Temp .* Org_Img(:,:,3);  
Color_Img = cat(3, Red_pixel, Green_pixel, Blue_pixel);  
figure,imshow(Color_Img)  
title('color segmented image')
```

```
se=strel('disk',2);  
I=imopen(Gray_img,se);  
disp('Morphological process completed')  
figure,imshow(I)  
title('Image after Morphological process')  
I=imfill(I,'holes');  
se=strel('disk',5); % I=imopen(I,se);
```


Відображення вільних місць

```

figure,imshow(Org_Img);

hold on;

plot(circen(:,1), circen(:,2), 'r+');

i=1;
for ii = 1 : size(circen, 1)

    if (I(round(circen(i,2)),round(circen(i,1))))~=0)

        rectangle('Position',[circen(ii,1) - cirrad(ii), circen(ii,2) - cirrad(ii),
2*cirrad(ii), 2*cirrad(ii)],...

            'Curvature', [0.8,0.4], 'edgecolor', 'b', 'linewidth', 1,'FaceColor','b'); end

        i=i+1;

    end

hold off;

disp('Displayed free parking lot')

% Матриця обробки, яка відобразатиметься pcolor
% 'm' зберігає кількість рядків і 'n' зберігає кількість стовпців
[m, n] = розмір (новоматриця);
% новий рядок і стовпець потрібні, оскільки вони не використовуються
pcolor

% створити порожній рядок для даних
newrow = zeros(1, n);

% створити порожній стовпець для даних
newcolumn = zeros (m + 1, 1);

```

```
% нові дані мають додатковий рядок  
newdata = [newrow; newmatrix];  
  
% нові дані мають додатковий стовпчик  
newdata = [newdata newcolumn];  
  
% перевернути дані вертикально (за допомогою фліппуду), щоб вони  
бачились так само  
  
% як відображено в електронній таблиці  
flipdata = flipud(newdata);  
  
% відображення перевернутих даних  
h = pcolor(flipdata);
```