

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційних радіотехнологій та технічного захисту інформації
(повна назва)

Кафедра медіаінженерії та інформаційних радіоелектронних систем
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)
(позначення документа)

Методи використання штучного інтелекту в системах відеоспостереження

(тема)

Виконав:

студент 2 курсу, групи МІм-22-2
Віктор ГЕРБІН

(прізвище, ініціали)

Спеціальність 172 Телекомунікації та радіотехніка

(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Медіаінженерія
(повна назва освітньої програми)

Керівник доц. Юрій ЛИКОВ
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Володимир КАРТАШОВ
(прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій та технічного захисту інформації

Кафедра Медіаінженерії та інформаційних радіоелектронних систем

Рівень вищої освіти другий (магістерський)

Спеціальність 172 Телекомунікації та радіотехніка

(код і повна назва)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма "Медіаінженерія"

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« ____ » _____ 20 ____ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

Студентові Гербіну Віктору Анатолійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Методи використання штучного інтелекту в системах відеоспостереження

затверджена наказом по університету від " 20 " 10 2023 р. № 1224 СТ

2. Термін подання студентом роботи 08.01.2024 р.

3. Вихідні дані до проекту (роботи) _____

1. Аналіз алгоритмів і підходів для забезпечення необхідного рівня надійності результуючої системи

2. Визначення особливостей використання систем розпізнавання зображень у камерах відеоспостереження в задачах, пов'язаних із захистом інформації

3. Аналіз можливості використання камер відеоспостереження на основі штучного інтелекту

4. Перелік питань, що потрібно опрацювати в роботі

ВСТУП

1. Основні напрями застосування технологій штучного інтелекту у системах відеоспостереження

2. Особливості використання штучного інтелекту у системах відеоспостереження

3. Обґрунтування вибору програмних засобів для вирішення задачі

4. Реалізація системи відеоспостереження з використанням штучного інтелекту

ВИСНОВКИ

ПЕРЕЛІК ПОСИЛАНЬ

ДОДАТКИ

5. Перелік графічного матеріалу із зазначенням обов'язкових креслеників, схем, плакатів, комп'ютерних ілюстрацій:

1. Цілі та задачі кваліфікаційної роботи; 2. Аналіз предметної області; 3. Приклади нейронних мереж; 4. Особливості використання штучного інтелекту у системах відеоспостереження; 5. Використання штучного інтелекту у системах відеоспостереження для підвищення рівня інформаційної безпеки; 6. Використання штучного інтелекту у системах відеоспостереження для підвищення рівня інформаційної безпеки; 7. Фактори впливу на рівень інформаційної безпеки; 8. Реалізація системи відеоспостереження з використанням штучного інтелекту; 9. Результат роботи програми. Помірна кількість людей, динамічна камера, наявність перекриття; 10. Результат роботи програми. Велика

кількість людей, динамічна камера, наявність перекриття; 11. Результат роботи програми. Статична вулична камера; 12. Аналіз показників роботи алгоритму; 13. Висновки.

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Термин виконання етапів роботи | Примітка |
|----|--|--------------------------------|----------|
| 1. | Основні напрями застосування технологій штучного інтелекту у системах відеоспостереження | 20.10.23–28.11.23 | |
| 2. | Особливості використання штучного інтелекту у системах відеоспостереження | 21.11.23–28.11.23 | |
| 3. | Обґрунтування вибору програмних засобів для вирішення задачі | 23.11.23–02.12.23 | |
| 4. | Реалізація системи відеоспостереження з використанням штучного інтелекту | 01.12.23–15.12.23 | |
| 5. | Графічна частина роботи | 15.12.23–20.12.23 | |
| 6. | Перевірка керівником | 20.12.23–24.12.23 | |
| 7. | Перевірка на академічний плагіат | 24.12.23–26.12.23 | |
| 8. | Перевірка завідувачем кафедри, рецензування | 27.12.23–31.12.23 | |

Дата видачі завдання 20.10.2023 р.

Студент Віктор ГЕРБІН
(підпис)

Керівник роботи _____
(підпис) доц. Юрій ЛИКОВ
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи має: 88 с., 43 рис., 3 табл., 2 додатки, 52 джерела.

СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ, **МОБІЛЬНІ РОБОТИ, НАВІГАЦІЯ, КОМП'ЮТЕРНИЙ ЗІР, ІНФОРМАЦІЙНА БЕЗПЕКА, , АЛГОРИТМ DEEP SORT ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ**

Об'єкт дослідження – системи відеоспостереження з використанням штучного інтелекту.

Предмет дослідження – особливості та перспективи використання систем відеоспостереження з використанням комп'ютерного зору та систем штучного інтелекту.

Мета кваліфікаційної роботи – дослідити особливості та перспективи використання систем відеоспостереження з використанням комп'ютерного зору для підвищення рівня фізичного захисту інформації в межах захищеного периметра шляхом організації спостереження за переміщеннями осіб.

Методи дослідження – **теоретичний аналіз, числові розрахунки, математичне моделювання, статистична обробка даних, аналіз геометрії антропогенних сцен.**

У даній роботі спроектована система відслідковування переміщень, з використанням алгоритму Deep Sort. Для цього, мовою UML, були система була описана за допомогою діаграм класів та діаграми пакетів, що відображають логіку роботи та взаємодії компонентів між собою. Спроектована система була реалізована мовою програмування Python з урахуванням специфіки задачі та наявності необхідного інструментарію.

Отримана реалізація системи з використанням алгоритму Deep Sort була перевірено у ряді ситуацій. Навіть при великій кількості об'єктів, що одночасно відстежуються, алгоритм показав задовільну швидкодію і точність.

ABSTRACT

The explanatory note of the qualification work has: 88 pages, 43 figures, 3 tables, 2 appendices, 52 sources.

VIDEO SURVEILLANCE SYSTEMS, MOBILE ROBOTS, NAVIGATION, COMPUTER VISION, INFORMATION SECURITY, , DEEP SORT ALGORITHM, CONVOLUTIONAL NEURAL NETWORKS

The object of research is video surveillance systems using artificial intelligence.

The subject of the study is the features and prospects of using video surveillance systems using computer vision and artificial intelligence systems.

The purpose of the qualification work is to investigate the features and prospects of using video surveillance systems using computer vision to increase the level of physical protection of information within the protected perimeter by organizing surveillance of people's movements.

Research methods – theoretical analysis, numerical calculations, mathematical modeling, statistical data processing, analysis of the geometry of anthropogenic scenes.

In this work, a movement tracking system is designed using the Deep Sort algorithm. For this, in the UML language, the system was described using class diagrams and package diagrams, which reflect the logic of work and the interaction of components with each other. The designed system was implemented in the Python programming language, taking into account the specifics of the task and the availability of the necessary tools.

The resulting implementation of the system using the Deep Sort algorithm was tested in a number of situations. Even with a large number of simultaneously tracked objects, the algorithm showed satisfactory speed and accuracy.

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CNN – Convolutional Neural Network;

DSSS (Direct Sequence Spread Spectrum) - метод розширення спектра методом прямої послідовності;

OFDM (Orthogonal Frequency Division Multiplexing) – метод модуляції з ортогональним частотним розділенням;

PSK (Phase Shift Keying), BPSK, QPSK - метод модуляції з стрибкоподібною зміною фази;

QAM (Quadrature Amplitude Modulation) - метод амплітудно фазової

ЗЗ - зворотний зв'язок;

КВА - контрольно-вимірювальна апаратура;

МЕК - міжнародна електротехнічна комісія;

модуляції

НКІ ЕМВ - надкороткоімпульсне електромагнітне випромінювання;

ПК - персональний комп'ютер;

РЕБ - радіоелектронна боротьба;

РЕП - радіоелектронне подавлення;

РРЛ - радіорелейні лінії;

СПО - спеціальне програмне забезпечення;

ШПС - широкосмугові сигнали;

ШШС - шумоподібний широкосмуговий сигнал;

ЗМІСТ

| | |
|--|----|
| Перелік умовних скорочень | 6 |
| Вступ..... | 9 |
| 1 Основні напрями застосування технологій штучного інтелекту у системах відеоспостереження | 11 |
| 1.1 Аналіз предметної області..... | 13 |
| 1.2 Штучні нейронні мережі та глибоке навчання | 16 |
| 1.2.1 Багатошарові перцептрони..... | 17 |
| 1.2.2 Передавальна функція | 19 |
| 1.3 Згорткова нейронна мережа | 22 |
| 1.3.1 Процес згортки | 22 |
| 1.3.2 Згорткові шари..... | 24 |
| 1.4 Фільтр Калмана | 26 |
| 1.5 Угорський алгоритм..... | 34 |
| 1.6 Висновки до розділу | 35 |
| 2 Особливості використання штучного інтелекту у системах відеоспостереження | 37 |
| 2.1 Канали витоку інформації | 37 |
| 2.2 Аналіз роботи систем відеоспостереження у задачах розпізнавання аномальної поведінки | 38 |
| 2.3 Модель взаємодії людини і об'єкта з точки зору інформаційної безпеки | 40 |
| 2.4 Використання штучного інтелекту у системах відеоспостереження для підвищення рівня інформаційної безпеки | 42 |
| 2.4.1 Побудова теплових карт переміщень на основі зібраних даних..... | 49 |

| | |
|---|----|
| | 8 |
| 2.5 Висновки до розділу | 50 |
| 3 Обґрунтування вибору програмних засобів для вирішення задачі..... | 53 |
| 3.1 Обґрунтування вибору програмних засобів | 53 |
| 3.2 Аналіз мови програмування Python | 59 |
| 3.3 Аналіз бібліотек для роботи з растровою графікою..... | 59 |
| 3.4 Аналіз можливостей даної бібліотеки | 60 |
| 3.4.1 Завантаження зображення | 60 |
| 3.4.2 Збереження зображення | 61 |
| 3.4.4 Накладання фільтрів | 61 |
| Висновки до розділу | 61 |
| 4 Реалізація системи відеоспостереження з використанням штучного інтелекту | 62 |
| 4.1 Модель оцінки | 62 |
| 4.2 Проектування системи..... | 63 |
| 4.3 Функціональні особливості..... | 69 |
| 4.4 Результати роботи програми..... | 70 |
| 4.5 Аналіз показників роботи алгоритму..... | 72 |
| 4.6 Висновки до розділу | 74 |
| ВИСНОВКИ..... | 76 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ | 80 |
| Додатки..... | 89 |
| Додаток А..... | 90 |
| Додаток Б | 98 |

ВСТУП

Актуальність роботи. Системи відеоспостереження з використанням штучного інтелекту сьогодні активно розвиваються і успішно використовуються у великих масштабах. Удосконалюється як наукова база штучних нейронних мереж, так і обчислювальна потужність технічних пристроїв. Результати цього дослідження пропонують новий погляд на використання системи відеоспостереження комп'ютерного зору, а саме для підвищення фізичного рівня захисту інформації.

Мета і завдання дослідження. Дослідити особливості та перспективи використання систем відеоспостереження з використанням комп'ютерного зору для підвищення рівня фізичного захисту інформації в охоронюваному периметрі шляхом організації моніторингу пересування людей.

Наукова новизна отриманих результатів. Результати роботи пропонують альтернативний шлях використання камер відеоспостереження на основі штучного інтелекту як додаткового методу контролю персоналу з метою підвищення рівня безпеки системи від витоку інформації через фізичні канали.

Практичне значення отриманих результатів. З розвитком галузі штучного інтелекту та контекстного аналізу зображень результати цієї роботи можуть бути використані в подальших дослідженнях впровадження більш складних систем, наприклад, в аналізі підозрілої поведінки з точки зору інформаційної безпеки.

Задачі дослідження:

1. Огляд досягнень у галузі комп'ютерного зору та огляд існуючих методів вирішення.
2. Аналіз алгоритмів і підходів для забезпечення необхідного рівня надійності результуючої системи.

3. Визначення особливостей використання систем розпізнавання зображень у камерах відеоспостереження в задачах, пов'язаних із захистом інформації.
4. Аналіз можливості використання камер відеоспостереження на основі штучного інтелекту, крім інших організаційно-технічних методів забезпечення захисту інформації.

1 ОСНОВНІ НАПРЯМИ ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ ШТУЧНОГО ІНТЕЛЕКТУ У СИСТЕМАХ ВІДЕОСПОСТЕРЕЖЕННЯ

Сьогодні термін штучний інтелект (ШІ) міцно увійшов у повсякденне життя. І хоча пристроям з елементами штучного інтелекту ще не вистачає здатності розуміти проблему та знаходити її вирішення, коли справа доходить до зменшення помилок у оперативних завданнях та пошуку аномалій у різних процесах, штучний інтелект випереджає людські можливості та компетенції.

Штучний інтелект — це набір теоретичних і практичних підходів в інформаційних технологіях, які передбачають створення систем, здатних функціонувати інтелектуально й незалежно, подібно до механізму прийняття рішень у людському мозку. Штучний інтелект відіграє важливу роль в оцінці помилок, які може зробити людина. Завдяки штучному інтелекту машина зможе вчитися на досвіді, обробляючи великі обсяги даних і розпізнаючи шаблони, що містяться в них. Наприклад, Siri від Apple, розпізнавання обличчя та безпілотний автомобіль покладаються на машинне навчання та обробку природної мови, які є окремою гілкою штучного інтелекту. Крім того, штучний інтелект включає багато суміжних областей і технологій, таких як машинне навчання, глибоке навчання, нейронні мережі, обробка природної мови та інші [1].

Для глибшого розуміння систематизуємо основні напрями досліджень у сфері штучного інтелекту, які ми практично впроваджуємо та використовуємо практично щодня.

Машинне навчання — це набір технологій, які дозволяють комп'ютерам мислити за допомогою математичних алгоритмів на основі зібраних даних і конкретних інструкцій і правил. Замість того, щоб програмувати комп'ютер на кожному кроці, цей підхід реалізує технології, які дозволяють йому поступово навчатися з даних без вказівок програміста. Прикладом машинного навчання

є віртуальні персональні помічники, які допомагають вам знаходити інформацію та надають вказівки щодо конкретних завдань, коли їх запитують голосом. Інші приклади включають: відеоспостереження для відстеження незвичайної поведінки, соціальні медіа-сервіси, з якими ви можете взаємодіяти – люди, яких ви можете знати, на основі постійної самоосвіти відповідно до вподобань людей.

Нейронні мережі (Deep Learning) є подальшим розвитком машинного навчання та є комп'ютерною моделлю, яка виконує завдання класифікації безпосередньо з зображень, тексту чи аудіо. Ця модель реалізована з використанням широкого набору багатовимірних даних і багаторівневої архітектури нейронної мережі. Глибоке навчання дозволяє досягти точності розпізнавання при використанні більш складних моделей. Прикладами застосування цієї моделі в повсякденному житті є автоматизоване водіння, розпізнавання обличчя та символів, промислова автоматизація тощо [2]. Обробка природної мови, набір методів, які дозволяють машині аналізувати людську мову, зокрема в чат-ботах для програм обслуговування клієнтів, використовує машинне навчання та обробку природної мови. Штучний інтелект використовує технології, які можна застосовувати в багатьох галузях, таких як фінансові установи, освіта та охорона здоров'я.

Зважаючи на вищевикладене цілком доцільним є потенційне використання методів ШІ для забезпечення розпізнавання образів. Кількість даних, що генеруються в сучасному світі, постійно збільшується, при цьому інформація зберігається та передається у різній формі з використанням мережі Інтернет.

1.1 Аналіз предметної області

Зоровий апарат людини здатний легко сприймати навколишній світ. Наприклад, він може відокремити головний об'єкт від його оточення та врахувати тонкі шаблони затінення (рис. 1.1).



Рисунок 1.1 – Приклад виділення об'єкту з навколишнього фону, зором людини

Крім того, подивившись на фотографію групи людей (рис. 1.2), можна легко порахувати точну кількість учасників і навіть за виразом обличчя зробити висновок про їхній настрій і стан. Перцептивні психологи витратили десятиліття, намагаючись зрозуміти, як працює зорова система, і хоча вони можуть використовувати певні оптичні ілюзії, щоб розкрити деякі основні принципи, наразі немає остаточних відповідей на всі поставлені запитання.



Рисунок 1.2 - Приклад застосування комп'ютерного зору у розпізнаванні
обличчя

Дослідники комп'ютерного зору розробили паралельні математичні методи для розпізнавання та відтворення тривимірної форми об'єктів на зображенні. Наразі існують досить надійні методи для точного обчислення часткової 3D-моделі середовища з використанням тисяч зображень, які частково перекриваються. Використовуючи досить великий набір даних, можна створювати точні щільні моделі 3D поверхонь за допомогою стереопокрощення (рис. 1.3).

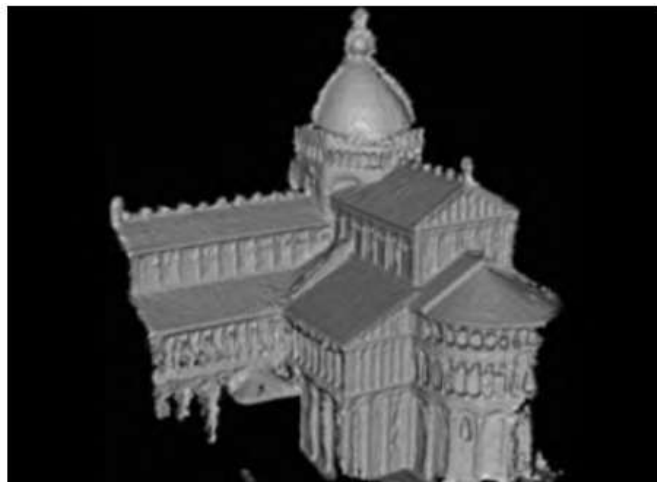


Рисунок 1.3 - Стереосистемні алгоритми зіставлення можуть створювати
детальну 3D-модель фасаду будівлі

Сучасні алгоритми відстеження руху здатні відстежувати рух людини або навіть групи людей в досить складних умовах зйомки. (рис. 1.4).



Рисунок 1.4 - Приклад роботи алгоритму відстеження людини, що рухається

Однак, незважаючи на ці досягнення, рівень розпізнавання образів і вилучення контексту із зображень все ще залишається досить низьким, не досягаючи рівня дитини дошкільного віку. Причина, чому розпізнавання образів є таким складним завданням, полягає в тому, що сама проблема має протилежний характер, вимагаючи відновлення невідомих даних із недостатньої інформації. Тому необхідно використовувати фізичні та імовірнісні моделі, щоб уникнути розбіжностей між потенційними рішеннями. Однак моделювання візуального світу в реальному часі з усіма його функціями наразі є неможливим завданням.

У задачах комп'ютерного зору робляться спроби зворотного відновлення прихованої інформації [3], тобто описати світ, представлений одним або серією зображень, і відтворити його властивості, такі як форма, освітлення та колірні переходи. Люди, які не працювали в цій сфері, часто недооцінюють складність проблеми. Це помилкове уявлення про те, що бачення має бути легким, сягає корінням у ранні дні штучного інтелекту, коли

спочатку вважалося, що когнітивні (міркування та планування) частини інтелекту за своєю суттю складніші за елементи сприйняття.

В даний час комп'ютерний зір використовується в різних галузях промисловості, зокрема:

- розпізнавання обличчя - для покращення фокусування камери, а також для більш релевантного пошуку зображень;
- візуальна аутентифікація - тобто перевірка особистості на основі певних рис обличчя;
- відслідковування руху транспорту - допоміжний метод контролю дорожнього трафіку та розпізнавання автомобільних номерів;
- оптичне розпізнавання символів - зчитування рукописних поштових кодів на листах та автоматичне розпізнавання номерного знаку;
- технічна інспекція - перевірка деталей для забезпечення якості, використовуючи стерео бачення з спеціалізованим освітленням для вимірювання деформацій крил літальних апаратів або пошук дефектів сталевих деталей з використанням рентгенівського зору;
- 3D-модельювання - повністю автоматизована побудова 3D-моделей з аерофотознімків;
- автомобільна безпека - виявлення непередбачених перешкод, таких як пішоходи на вулиці, в умовах, коли активні методи зору, такі як радар, не працюють;
- об'єднання комп'ютерних зображень (COI) з реальними відеозаписами.

1.2 Штучні нейронні мережі та глибоке навчання

Штучна нейронна мережа містить в собі набір взаємопов'язаних пристроїв обробки даних [3]. Дано вхідні значення w_0, x_1, \dots, x_D , де являє собою зовнішній вхід, а x_1, \dots, x_D - входи, з інших блоків обробки (в межах мережі). Пристрій обробки обчислює його вихід як $y = f(z)$. Тут f - передавальна

функція, а z отримується шляхом застосування правила розповсюдження, яке відображає всі входи на вхід до z . Ця модель єдиного пристрою обробки включає визначення нейрона, де замість правила поширення використовується суматор для обчислення z як зваженої суми всіх вхідних даних.

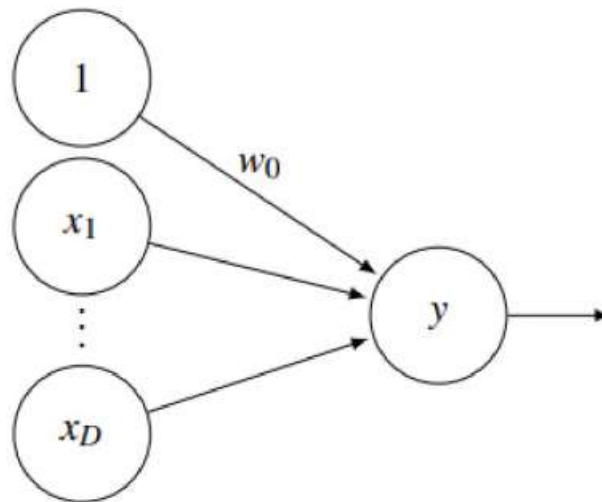


Рисунок 1.5 - Графічне представлення нейронної мережі

Нейронні мережі можуть бути візуально представлені у вигляді спрямованого графа, що називається мережевий граф. Кожен елемент представлена вершиною, позначеною відповідно до виходу, а зв'язок між елементами позначається у вигляді однонапрямлених ребр (рисунок 1.5).

Для зручності розрізняють вхідні та вихідні елементи. Вхідний елемент обчислює вихід $y := x$, де x - одиничне вхідне значення. Вихідні елементи можуть приймати довільну кількість вхідних значень. В цілому мережа являє собою функцію $y(x)$, розміри якої залежать від кількості вхідних та вихідних елементів, що означає, що вхід мережі приймається вхідними елементами а вихідні елементи формують вихід мережі.

1.2.1 Багатошарові перцептрони

Перцептрон, що представлений на рис. 1.6,

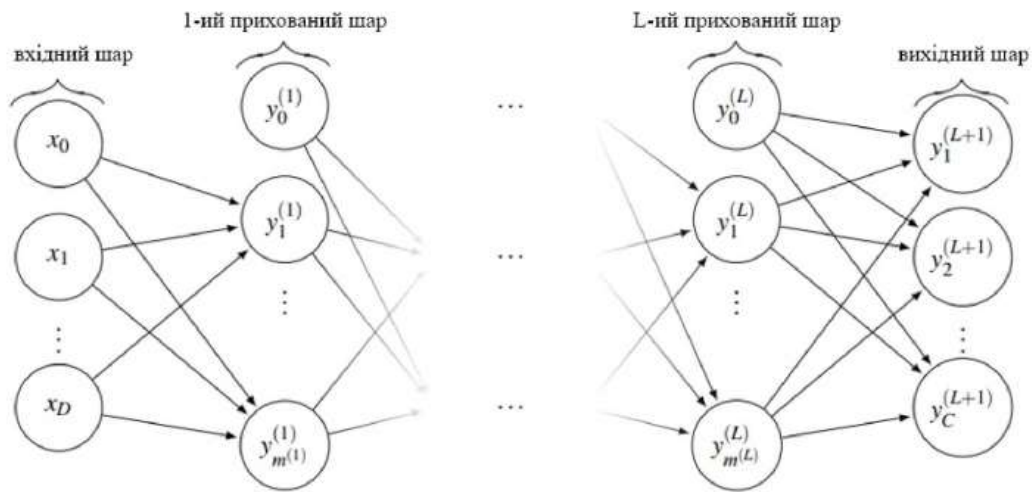


Рисунок 1.6 - Мережевий граф перцептрона з $L + 1$ шарів, складається з D входів, C виходів, а також містить I прихованих шарів

має $L + 1$ шарів, складається з D входів, C виходів, а також містить I прихованих шарів. Кожний i -тий елемент з шару l обчислює вихід:

$$y_i^l = f(z_i^{(l)}), \quad (1.1)$$

де

$$z_i^{(l)} = \sum_{k=0}^{m^{(l-1)}} w_{i,k}^{(l)} y_{i,k}^{(l-1)} + w_{i,0}^{(l)}, \quad (1.2)$$

де $w_{i,k}^l$ позначає зважене з'єднання k -го елемента у шарі $l - 1$ з i -тим елементом у шарі l , а $w_{i,0}^l$ являє собою зовнішній вхід до пристрою і називається зміщенням. Тут, $m^{(l)}$ позначає кількість елементів у шарі l , так, що $D = m^{(0)}$ та $C = m^{(L+1)}$. Для простоти, зміщення можна розглядати як ваг при введенні фіктивного блоку $y_0^l := 1$ у кожен з шарів:

$$z_i^{(l)} = \sum_{k=0}^{m^{(l-1)}} w_{i,k}^{(l)} y_{i,k}^{(l-1)}, \quad (1.3)$$

або

$$z^{(l)} = w^{(l)} y^{(l-1)}, \quad (1.4)$$

де $z^{(l)}, w^{(l)}$ та $y^{(l-1)}$ позначають відповідні векторні та матричні вигляди про фактичні входи $z_i^{(l)}$, ваги $w_{i,k}^{(l)}$ та виходи $y_{i,k}^{(l-1)}$ відповідно.

У загальному вигляді, функціональне представлення багат шарового перцептрона виглядає наступним чином:

$$Y(\bullet, w): R^D \rightarrow R^C, x \Rightarrow y(x, w), \quad (1.5)$$

де вихід вектору (x, w) містить вихідні значення $y_i(x, w) := y_i^{L+1}$ та w є вектором усіх вагових коефіцієнтів в мережі.

Нейронна мережа називається глибокою, якщо вона містить більш ніж три прихованих шара. Тренування глибоких нейронних мереж, розглядається як непросте завдання.

1.2.2 Передавальна функція

Передавальна функція - залежність вхідного сингала елемента нейронної мережі до вхідного:

$$y = f(z) \quad (1.6)$$

Найбільш часто [3] використовуваними типами передавальних функцій є:

- порогові функції;
- кусково-лінійна функції;
- сигмоїдні функції.

У загальному випадку порогова функція визначається як функція Гевісайда (рис 1.7):

$$h(z) = \begin{cases} 1, & \text{якщо } z \geq 0 \\ 0, & \text{якщо } z < 0 \end{cases} \quad (1.7)$$

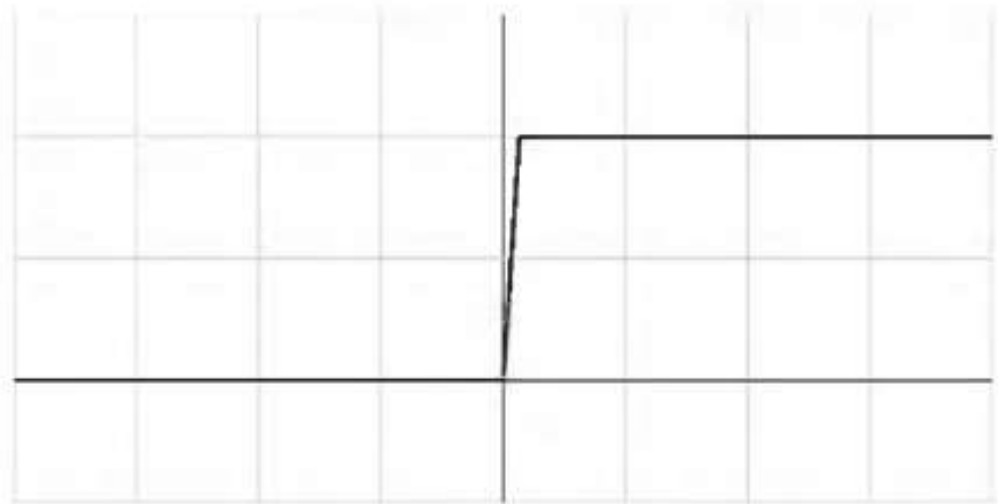


Рисунок 1.7 - Графік функції Гевісайда

Проте як порогові функції, так і кусково-лінійні функції мають деякі недоліки. По-перше, для навчання мережі може знадобитися, щоб функція активації була диференційованою. По-друге, функції нелінійної активації показують кращий результат за рахунок додаткової обчислювальної потужності, яку вони спричиняють (за рахунок нелінійності).

Розповсюдженою передавальною функцією є сигмоїдна функція. Наприклад, логістична сигмоїдна функція (рис. 1.8) має вигляд:

$$\sigma(z) = \frac{1}{1+\exp(-z)} \quad (1.9)$$

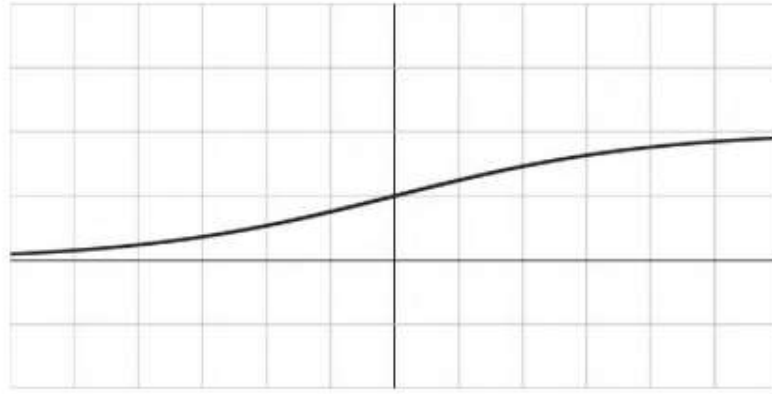


Рисунок 1.8 - Графік логістичної сигмоїдної функції

Графік даної функції має *s*-подібну форму, а сама функція є диференційованою та монотонною. Функція гіперболічного тангенса $th(z)$ може розглядатися як лінійне перетворення логістичної сигмоїди на інтервал $[-1,1]$. Варто зауважити, що обидві дані передавальні функції є функціями з насиченням (тобто є обмеженими).

При використанні нейронних мереж у задачах класифікації, використовується передавальна функція *softmax* для елементів виведення, що інтерпретують вихідні значення як апостеріорні ймовірності. В такому випадку виходом *i*-го блоку у вихідному шарі є:

$$\sigma(z^{L+1}, i) = \frac{\exp(z_i^{(L+1)})}{\sum_{k=1}^C \exp(z_k^{(L+1)})} \quad (1.10)$$

Досліди показують, що функція логістичної сигмоїди, а також гіперболічного тангенса, мають погані показники у задачах глибокого навчання [3]. Повідомляється про досягнення найкращих показників за допомогою передавальної функції *softsign* (рис. 1.9):

$$s(z) = \frac{1}{1+|z|} \quad (1.11)$$

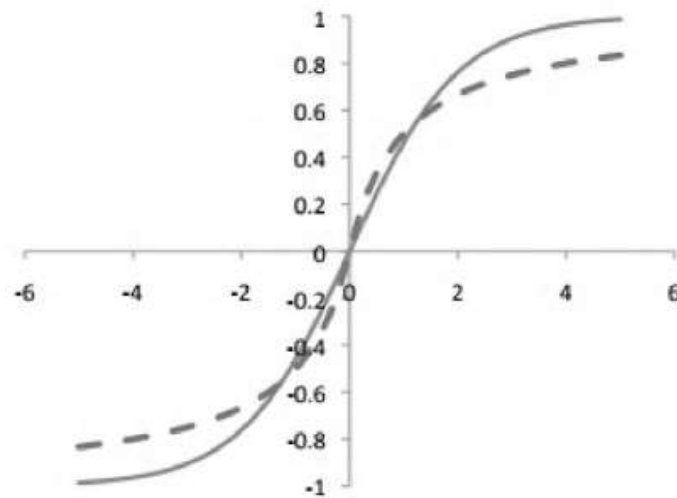


Рисунок 1.9 - Співставлення графіків функцій *th* (світла лінія) та *softsign* (темна пунктирна лінія)

1.3 Згорткова нейронна мережа

Згорткові нейронні мережі (CNN - Convolutional Neural Network) є класом глибоких штучних нейронних мереж, який успішно застосовувався до аналізу візуальних зображень [3]

1.3.1 Процес згортки

Для простоти представимо зображення, як градації сірого, яке визначається функцією:

$$I: \{1, \dots, n_1\} \times \{1, \dots, n_2\} \rightarrow W \subseteq \mathbb{R}, (i, j) \Rightarrow I_{i,j}, \quad (1.12)$$

так, що зображення I може бути представлено за допомогою матриці розміром $n_1 \times n_2$. У більшості випадків W представлений набором $\{0, \dots, 255\}$, що являє собою 8-бітний канал. Тоді кольорове зображення може бути представлено масивом розміру $n_1 \times n_2 \times 3$, якщо взяти три кольорові канали, наприклад схему КОВ. Виходячи з того, що $K \in R^{2h_1+1 \times 2h_2+1}$, дискретна згортка зображення I з фільтром K можна представити формулою:

$$(I * K)_{r,s} := \sum_{u=-h_1}^{h_1} \sum_{v=-h_2}^{h_2} K_{u,v} I_{r+u,s+v}, \quad (1.13)$$

де фільтр має вигляд:

$$K = \begin{pmatrix} K_{-h_1,-h_2} & \cdots & K_{-h_1,h_2} \\ \vdots & K_{0,0} & \vdots \\ K_{h_1,-h_2} & \cdots & K_{h_1,h_2} \end{pmatrix}, \quad (1.14)$$

Варто зауважити, що поведінка цієї операції на границях зображення повинна бути визначена належним чином. Одним із шляхів вирішення даної проблеми є застосування фільтра лише для тих місць зображення, де він поводить себе коректно. Це, в свою чергу, призводить до того, що вихідний масив буде меншим за початкове зображення.

Розповсюдженим фільтром для згладжування є дискретний фільтр Гауса:

$$(K_{G(\sigma)})_{r,s} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{r^2+s^2}{2\sigma^2}\right) \quad (1.15)$$

де σ є середньо квадратичним відхиленням розподілу Гауса.

1.3.2 Згорткові шари

Позначимо згортковий шар як l . Тоді вхід на шар l складається з $m_1^{(l-1)}$ відображень ознак, отриманих з попереднього шару, кожне має розмір $m_2^{(l-1)} \times m_3^{(l-1)}$. У випадку, коли $l = 1$, на вхід подається власно необроблене зображення I , що складається з одного чи більше складових каналів. Вихід згорткового шару l складається з $m_1^{(l)}$ відображень ознак, що має вигляд матриці розмірності $m_2^{(l)} \times m_3^{(l)}$. Кожна i -та матриця відображення ознак $Y_i^{(l)}$ з шару l має наступний вигляд:

$$Y_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{m_1^{(l-1)}} K_{i,j}^{(l)} * Y_j^{(l-1)} \quad (1.16)$$

де $B_i^{(l)}$ є матрицею зсуву (англ. – *bias matrix*), а $K_{i,j}^{(l)}$ є фільтром $2h_1^{(l)} + 1 \times 2h_2^{(l)} + 1$, що поєднує j -те відображення ознак у шарі $(l-1)$ з i -тим відображенням з шару l (рис. 1.10).

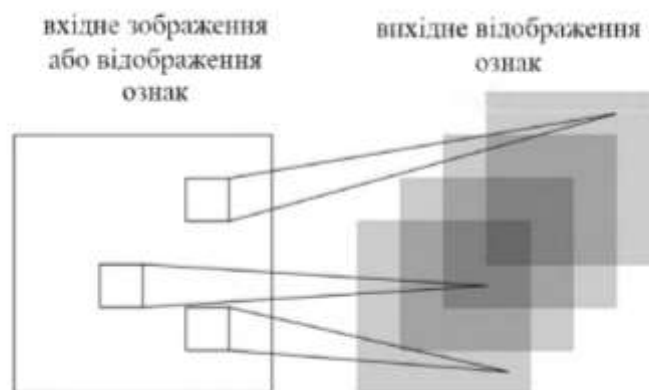


Рисунок 1.10 - Ілюстрація одного зі згорткових шарів

На вхід подається вхідне зображення (у випадку $l = 1$) чи відображення ознак з попереднього шару, що згортається за допомогою набору фільтрів для отримання вихідного відображення.

При цьому $m_1^{(l)}$ та $m_2^{(l)}$ знаходяться під впливом ефекту границі, а отже, при використанні дискретної згортки у так званій коректній області, розмір вихідного відображення буде наступним:

$$m_2^{(l)} = m_2^{(l-1)} - 2h_1^{(l)} \quad (1.17)$$

та

$$m_3^{(l)} = m_3^{(l-1)} - 2h_2^{(l)} \quad (1.18)$$

Часто фільтри, що використовують для обчислення певного відображення ознак $Y_i^{(l)}$, є однаковими, тобто $K_{i,j}^{(l)} = K_{i,k}^{(l)}$, для $j \neq k$. Окрім цього, сума $\sum_{j=1}^{m_1^{(l-1)}} K_{i,j}^{(l)} * Y_j^{(l-1)}$ може вийти за межі підмножини вхідних відображень ознак.

Для того, щоб співвіднести згортковий шар та його операції, з багатошаровим перцептроном, необхідно переписати рівняння:

$$Y_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{m_1^{(l-1)}} K_{i,j}^{(l)} * Y_j^{(l-1)} \quad (1.19)$$

Кожне відображення ознак $Y_i^{(l)}$ у шарі l складається з $m_1^{(l)} \cdot m_3^{(l)}$ елементів, що розташовані у двовимірному масиві. При цьому елемент в положенні (r,s) обчислює вихід наступним чином:

$$\left(Y_i^{(l)}\right)_{r,s} = \left(B_i^{(l)}\right)_{r,s} + \sum_{j=1}^{m_1^{(l-1)}} \left(K_{i,j}^{(l)} * Y_j^{(l-1)}\right)_{r,s}, \quad (1.20)$$

$$\left(Y_i^{(l)}\right)_{r,s} = \left(B_i^{(l)}\right)_{r,s} + \sum_{j=1}^{m_1^{(l-1)}} \sum_{u=-h_1^{(l)}}^{h_1^{(l)}} \sum_{v=-h_2^{(l)}}^{h_2^{(l)}} \left(K_{i,j}^{(l)}\right)_{u,v} * \left(Y_j^{(l-1)}\right)_{r+u,s+v}, \quad (1.21)$$

Тренувальні вагові коефіцієнти знаходяться у фільтрах $K_{i,j}^{(l)}$ та матриці зміщень $B_i^{(l)}$.

Також варто зауважити використання процесу відбору виборки (англ. subsampling) для зменшення впливу шумів та спотворень. Це може бути зроблено за рахунок так званих факторів пропуску (англ. skipping factors) $s_1^{(l)}$ та $s_2^{(l)}$. Підхід заключається у пропусканні певної кількості пікселів зображення у горизонтальному та вертикальному напрямках перед повторним приміненням фільтру. При цьому розміри вихідного відображення ознак будуть наступними:

$$m_2^{(l)} = \frac{m_2^{(l-1)} - 2h_1^{(l)}}{s_1^{(l)} + 1} \quad (1.22)$$

та

$$m_3^{(l)} = \frac{m_3^{(l-1)} - 2h_2^{(l)}}{s_2^{(l)} + 1} \quad (1.23)$$

1.4 Фільтр Калмана

Фільтр Калмана давно вважається оптимальним рішенням для багатьох задач відстеження та прогнозування даних [8].

Багато сигналів можна описати наступним чином:

$$y_k = a_k x_k + n_k, \quad (1.24)$$

де y_k - залежний від часу сигнал, a_k - це коефіцієнт посилення, x_k - сигнал, що переносить інформацію і n_k - адитивний шум.

Мета - оцінити x_k . Похибкою вважається різниця між очікуваним \hat{x}_k та реальним x_k :

$$f(e_k) = f(x_k - \hat{x}_k). \quad (1.25)$$

Особлива форма $f(e_k)$ залежить від застосування, однак відомо, що ця функція повинна бути позитивною та при цьому монотонно зростати. Функція помилки, яка має ці характеристики являє собою квадрат функції помилки. Оскільки необхідно враховувати здатність фільтра прогнозувати багато даних протягом певного періоду часу, необхідно ввести більш значущу метрику, а саме очікуване значення функції помилки;

$$f_{\text{втрати}} = E(f(e_k)) \quad (1.26)$$

З цього випливає функція середнього квадратичного відхилення.

$$\epsilon_t = E(f(e_k^2)). \quad (1.27)$$

Більш строгі виведення може бути виконані з використанням статистики максимальної вірогідності. Це досягається переоцінкою мети фільтра на визначення значення \hat{x}_k такого, що максимізує вірогідність y . А саме,

$$\max[P(y|\hat{x}_k)]. \quad (1.28)$$

Якщо припустити, що адитивний випадковий шум є розподіленим гауссовим шумом зі стандартним відхиленням k , то

$$P(y|\hat{x}_k) = K_k \exp - \left(\frac{(y_k - a_k x_k)^2}{2\sigma_k^2} \right), \quad (1.29)$$

де K_k – константа нормалізації. Тоді максимальна функція правдоподібності є наступною:

$$P(y|\hat{x}_k) = \prod_k K_k \exp - \left(\frac{(y_k - a_k x_k)^2}{2\sigma_k^2} \right), \quad (1.30)$$

З цього випливає, що

$$\log P(y|\hat{x}_k) = K_k \exp - \frac{1}{2} \sum_n \left(\frac{(y_k - a_k x_k)^2}{2\sigma_k^2} \right), \quad (1.31)$$

Керуюча функція являє собою середньо квадратичне відхилення і може бути максимізована змінною x_k . Тому вона застосовується у разі очікуваного відхилення y_k , як гауссового розподілу. У такому випадку середньо квадратичне відхилення служить для забезпечення значення x_k , яке максимізує вірогідність сигналу y_k .

У даному випадку оптимальний фільтр визначається як той, з набору всіх можливих, що мінімізує середньо квадратичне відхилення.

Припустимо, що ми хочемо знати значення змінної в процесі з формою

$$x_{k+1} = \Phi x_k + \omega_k, \quad (1.32)$$

де x_k - вектор станів процесу в момент часу k (розмірність $n \times 1$); Φ - це матриця (розмірність $n \times m$) переходу станів в процесі від стану у момент часу k до стану у момент часу $k + 1$, що вважається стаціонарною з плином часу; ω_k - пов'язаний з процесом білий шум з відомою коваріацією (розмірність $n \times 1$).

Спостереження за цією змінною можна показати у формі

$$z_k = Hx_k + v_k. \quad (1.33)$$

де z_k - фактичне вимірювання величини x у момент часу k (розмірність $m \times 1$); H - це зв'язок, без шуму, між вектором стану та вектором вимірювання, що припускається стаціонарним з плином часу (розмірність $m \times n$); v_k - пов'язана з цим помилка вимірювання, тобто білий шум, що пов'язаний з вимірювання, що має відому коваріацію та нульову перехресну кореляцію з власне шумом процесу (розмірність $m \times 1$).

Як було показано, для мінімізації середньо квадратичного відхилення, для отримання оптимального фільтру можна правильно моделювати системні похибки за допомогою гауссових розподілів. Коваріації двох шумових моделей вважаються стаціонарними з плином часу і виглядають наступним чином

$$Q = E[\omega_k \omega_k^T], \quad (1.34)$$

$$R = E[v_k v_k^T], \quad (1.35)$$

З функції середнього квадратичного відхилення, впливає:

$$E[e_k e_k^T] = P_k, \quad (1.36)$$

де P_k - матриця коваріації похибки в момент часу k (розмірність $n \times n$).

Рівняння може бути розширене наступним чином:

$$P_k = E[e_k e_k^T] = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] \quad (1.37)$$

Приймаючи попередню оцінку \hat{x}_k за \hat{x}'_k , що була отримана зі знання системи, можна отримати рівняння для нової оцінки, поєднавши попередню оцінку з даними вимірювань

$$\hat{x}_k = \hat{x}'_k + K_k(z_k - H\hat{x}'_k), \quad (1.38)$$

де K_k - передавальний коефіцієнт Калмана. В свою чергу, $z_k - H\hat{x}'_k$ нововведення (відхилення) вимірювання

$$i_k = z_k - H\hat{x}'_k, \quad (1.39)$$

За допомогою підстановок отримаємо

$$\begin{aligned} \hat{x}_k &= \hat{x}'_k + K_k(H\hat{x}'_k + v_k - H\hat{x}'_k) \\ P_k &= E\{[(I - K_k H)(x_k - \hat{x}'_k) - K_k v_k][(I - K_k H)(x_k - \hat{x}'_k) - K_k v_k]^T\} \end{aligned} \quad (1.40)$$

Варто зауважити, що, що $x_k - \hat{x}'_k$ є похибкою попередньою оцінки. Вона не корелює з похибкою вимірювання, отже рівняння може бути перезаписаним у вигляді

$$P_k = (I - K_k H)E[(x_k - \hat{x}'_k)(x_k - \hat{x}'_k)^T](I - K_k H) + K_k E[v_k v_k^T] K_k^T \quad (1.41)$$

Після підстановки маємо

$$P_k = (I - K_k H)P'_k(I - K_k H)^T + K_k R K_k^T \quad (1.42)$$

де P'_k є попередньою оцінкою P'_k ,

Отримане рівняння є коваріацією оновленої оцінки. Діагональ матриці коваріації містить середні квадратичні відхилення

$$P_{kk} = \begin{bmatrix} E[e_{k-1}e_{k-1}^T] & E[e_k e_{k-1}^T] & E[e_{k+1} e_{k-1}^T] \\ E[e_{k-1}e_k^T] & E[e_k e_k^T] & E[e_{k+1} e_k^T] \\ E[e_{k-1}e_{k+1}^T] & E[e_k e_{k+1}^T] & E[e_{k+1} e_{k+1}^T] \end{bmatrix} \quad (1.43)$$

Сума елементів діагоналі матриці є слідом матриці. У випадку матриці коваріації похибок, слідом буде сума середньо квадратичних відхилень. Тому дане відхилення можна мінімізувати за допомогою мінімізації сліду P_k , що призведе до зменшення сліду P_{kk} .

Слід P_k є першою похідною по K_k і результат прирівнюється до нуля для знаходження умови мінімуму.

$$P_k = P'_k - K_k H P'_k - P'_k H^T K_k^T + K_k (H P'_k H^T + R) K_k^T \quad (1.44)$$

Варто зауважити, що слід матриці дорівнює сліду її транспозиції, отже можемо записати рівняння як

$$T[P_k] = T[P'_k] - 2T[K_k H P'_k] + T[K_k (H P'_k H^T + R) K_k^T] \quad (1.45)$$

де $T[P_k]$ - слід матриці P_k .

Після диференціювання по K_k отримуємо

$$\frac{dT[P_k]}{dK_k} = -2(H P'_k)^T + 2K_k (H P'_k H^T + R) \quad (1.46)$$

Після прирівнювання з нулем та перестановки доданків

$$(H P'_k)^T = K_k (H P'_k H^T + R) \quad (1.47)$$

Знаходимо коефіцієнт Калмана K_k

$$K_k = HP'_kH^T + R \quad (1.48)$$

Після остаточної підстановки отримуємо

$$P_k = P'_k - P'_kH^T(HP'_kH^T + R)^{-1}HP'_k \quad (1.49)$$

$$P_k = P'_k - K_kHP'_k \quad (1.50)$$

$$P_k = (I - K_kH)P'_k \quad (1.51)$$

Отримане рівняння - це оновлена матриця коваріації похибок з оптимальним передавальним коефіцієнтом (формула коваріації апостеріорної похибки). Отримані рівняння використовують для оцінки значення x_k . Проекція стану досягається за рахунок

$$\hat{x}'_{k+1} = \Phi x_k \quad (1.52)$$

Для завершення рекурсії необхідно знайти рівняння, що проектує матрицю коваріації похибок у наступний інтервал часу $k + 1$. Для початку сформуємо вираз для попередньої похибки

$$e'_{k+1} = x_{k+1} - \hat{x}'_{k+1}$$

$$e'_{k+1} = (\Phi x_k + \omega_k) - \Phi \hat{x}'_k$$

$$e'_{k+1} = \Phi e_k + \omega_k$$

Продовжуючи рівняння для $k + 1$.

$$P'_{k+1} = E[e'_{k+1}e_{k+1}^T] = E[(\Phi e_k + \omega_k)(\Phi e_k + \omega_k)^T] \quad (1.53)$$

Варто зауважити, що e_k та ω_k мають нульову перехресну кореляцію, оскільки шум ω_k накопичується у момент часу $\overline{k, k + 1}$, тоді як e_k - це похибка до моменту k .

$$P'_{k+1} = E[e'_{k+1}e_{k+1}^T]$$

$$P'_{k+1} = E[\Phi e_k(\Phi e_k)^T] + E[\omega_k\omega_k^T]$$

$$P'_{k+1} = \Phi P_k \Phi^T + Q$$

На рис. 1.11 представлений остаточний алгоритм фільтру Калмана.

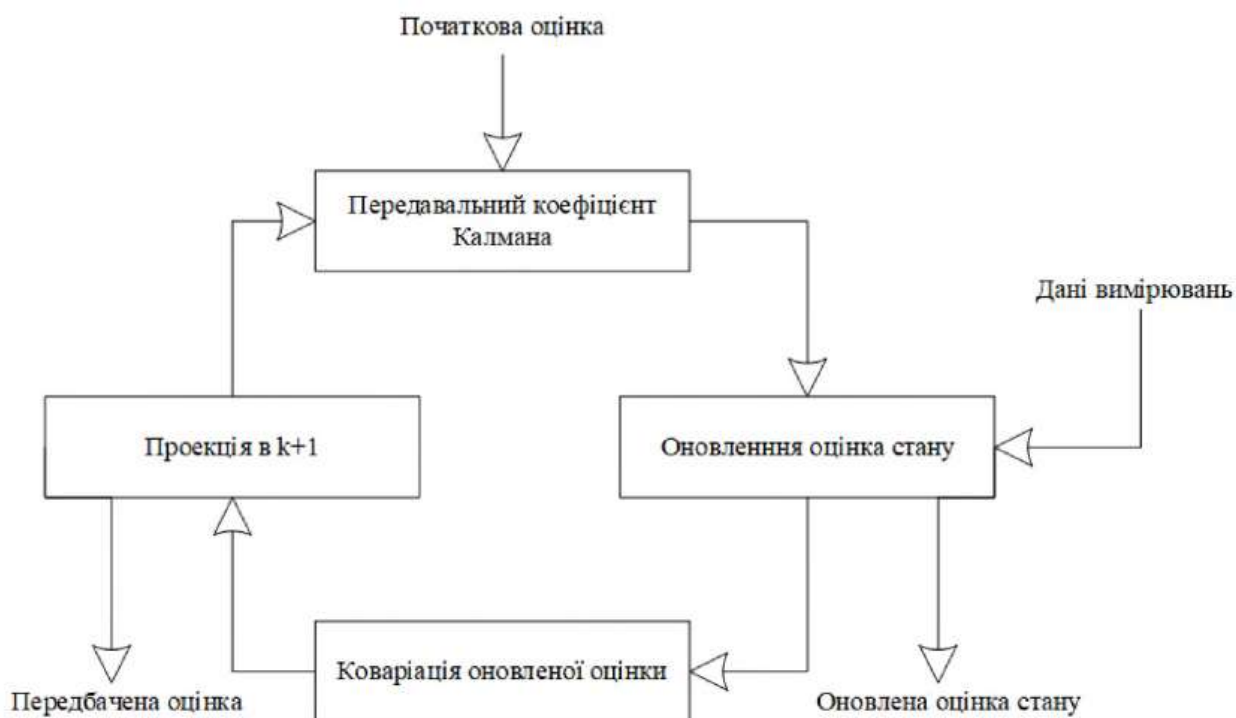


Рисунок 1.11 - Діаграма фільтру Калмана

1.5 Угорський алгоритм

Угорський алгоритм — алгоритм комбінаторної оптимізації, що розв'язує задачу про призначення за поліноміальний час $O(n^4)$.

Постановка завдання: Дано n ресурсів, якими необхідно розпорядитись для вирішення n задач, причому на один ресурс має приходитись лише одне завдання. Також, відома вартість вирішення кожної з задач на кожному ресурсі. Необхідно знайти оптимальне призначення завдань ресурсам. У матричному вигляді отримаємо матрицю розмірності $n \times n$, де елемент в i -тому рядку та j -тому стовпці показує вартість призначення j -тої роботи i -тому ресурсу

$$C = \begin{bmatrix} c_{1,1} & \cdots & c_{1,n} \\ \vdots & \ddots & \vdots \\ c_{n,1} & \cdots & c_{n,n} \end{bmatrix} \quad (1.54)$$

Алгоритм [7]:

1. Впорядкувати інформацію в матриці таким чином, щоб рядки матриці представляли «ресурси», а колонки — «завдання», тоді як кожен елемент матриці представляє витрати на виконання певним ресурсом певного завдання.

2. Переконатися в тому, що матриця є квадратною; в протилежному випадку слід додати фіктивний рядок (ресурс) чи колонку (завдання), де кожен елемент буде дорівнювати найбільшому елементу початкової матриці.

3. В кожному рядку від кожного елемента відняти найменше значення для даного рядка.

4. В кожному стовпці від кожного елемента відняти найменше значення для даного стовпця.

5. Викреслити всі нульові елементи з найменш можливою кількістю ліній (якщо кількість ліній дорівнює розмірності матриці, то слід перейти до кроку 9).

6. Додати мінімальний з не викреслених елементів до кожного викресленого елементу (якщо елемент викреслено двома лініями, то додавати слід теж двічі) 7. Від кожного елементу матриці відняти мінімальний елемент.

8. Знову викреслити всі нульові елементи використовуючи найменшу кількість ліній (якщо кількість використаних ліній не дорівнює розмірності матриці, то слід повернутись до кроку 6).

9. Вибрати розподіл «завдань» між «ресурсами» таким чином, щоб в кожному рядкові та стовпці був вибраний лише один нуль.

10. Перенести розподіл на початкову матрицю, ігноруючи фіктивні колонки і рядки. Цей розподіл покаже який «ресурс» яке «завдання» має виконати, а сума виділених елементів покаже загальну вартість виконання робіт.

1.6 Висновки до розділу

Людина і вищі тварини буквально повинні розпізнавати, приймати рішення і вчитися на кожному кроці. Наука про штучні нейронні мережі виникла з бажання зрозуміти, як мозок вирішує такі складні завдання, і реалізувати ці принципи в автоматизованих системах.

Хоча штучні нейронні мережі є лише дуже спрощеними еквівалентами природних нейронних мереж, нервові системи тварин і людей набагато складніші, ніж пристрої, які можна створити за допомогою сучасних технологій. Проте навіть цього рівня достатньо для успішного вирішення багатьох практичних завдань.

Поки що серед таких завдань – завдання з розпізнавання тексту, моніторинг автомобільного руху, моделювання об'єктів 3-0 на основі набору статичних зображень, а також виявлення та відстеження переміщень об'єктів.

Саме методи відстеження об'єктів, тобто людей, засновані на відеозаписах, починають активно використовувати різні організації державного та приватного секторів. Для громадських організацій метою є забезпечення громадського порядку, тоді як приватні організації прагнуть забезпечити промислову безпеку та безпеку бізнесу. Однак у часи підвищення цінності комерційної таємниці мало уваги приділяється саме безпеці в інформаційному сенсі.

У цій главі обговорюються основні терміни та принципи розпізнавання образів, які розвинулися за останні півстоліття. Допоміжні алгоритми, такі як фільтр Калмана та угорський алгоритм, також вважалися важливими елементами майбутньої системи.

2 ОСОБЛИВОСТІ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ У СИСТЕМАХ ВІДЕОСПОСТЕРЕЖЕННЯ

2.1 Канали витоку інформації

Витік конфіденційної інформації - це неконтрольований вихід конфіденційної інформації за межі ІБ або кола осіб, яким вона довірена службою, відомих у процесі роботи. Цей витік може бути наслідком:

- розголошення конфіденційної інформації;
- відходу інформації по різним, головним чином технічним, каналам;
- несанкціонованого доступу до конфіденційної інформації різними способами.

Розголошення інформації її власником або іншими володільцями - це умисне або необережне вчинення посадовими особами та користувачами, яким під час виконання службових обов'язків чи роботи в установленому порядку довірено відповідну інформацію, що призвело до її розголошення особам, які не має права доступу до інформації.

Можливий неконтрольований витік конфіденційної інформації з візуально-оптичних, акустичних, електромагнітних та інших каналів.

Несанкціонований доступ — це протиправне, навмисне володіння конфіденційною інформацією особою, яка не має права доступу до інформації, що захищається.

Ця робота присвячена фізичним каналам витоку інформації та фізичного несанкціонованого доступу до захищених інформаційних ресурсів:

- викрадення/копіювання/знищення фізичних носіїв інформації;
- несанкціонований доступ персоналу та сторонніх осіб до об'єкта;
- перехоплення інформації по акустичним/візуальним каналам витоку (підслуховування, підглядання).

2.2 Аналіз роботи систем відеоспостереження у задачах розпізнавання аномальної поведінки

Автоматизовані системи відеоспостереження складаються з мережі відеодатчиків, які відстежують людей, враховуючи їх поведінкові особливості та виокремлюючи серед них ті, які можуть становити загрозу.

Цей підхід використовується в системах, де є якісь універсальні, явні (візуальні) метрики, які дозволяють визначити стан безпеки/небезпеки.

Прикладом таких індикаторів може бути виявлення певних предметів, які вказують на небезпеку, наприклад зброї (рис. 2.1)



Рисунок 2.1 - Розпізнавання збройного нападу

Іншою аномалією, що може бути свідченням незвичності подій, що відбуваються, є поза людини та швидкість руху (рис. 2.2).



Рисунок 2.2 - Розпізнавання незвичної поведінки

Системи комп'ютерного зору, навчені розпізнавати рух людини, розпізнають рух інших транспортних засобів, наприклад, велосипеда (рис. 2.3), як ненормальний і повідомляють про це оператора.



Рисунок 2.3 - Аномальний рухаючийся об'єкт (велосипедист)

Однак у сфері інформаційної безпеки сам факт складності критеріїв інформаційної безпеки є проблематичним. Наприклад, нормою є фізичний доступ однієї особи до певного об'єкта інформаційної безпеки, а та сама дія

іншої особи може бути порушенням рівня доступу. Крім того, більш-менш достовірне виявлення самого факту фізичного доступу до певного інформаційного ресурсу є складним завданням.

2.3 Модель взаємодії людини і об'єкта з точки зору інформаційної безпеки

Можливим рішенням проблеми може виступати модель взаємодії людини з об'єктами (human-object interaction) (рис. 2.4). Дана модель була досліджена у роботі.



Рисунок 2.4 - Результат роботи iCAN

У цій роботі запропоновано модуль, який використовує контекстно-залежний підхід, а також навчається виділяти інформаційні області на основі зовнішнього вигляду людини чи об'єкта. Контекст дає вам підказки щодо того, на які ділянки зображення слід звернути увагу. Наприклад, щоб визначити, чи тримає людина якийсь предмет у руці, потрібно звернути увагу на область навколо руки людини. Однак якщо на фото є велосипед, варто звернути увагу на позу людини, яка знаходиться поруч. Це також допоможе вам уникнути можливих помилок, оскільки в цьому випадку їзда верхи або велосипед – це різні види діяльності. Система також фокусується на очах людини, щоб визначити їх гостроту.

Однак існує ряд недоліків, які на даний момент не дозволяють використовувати цей підхід у системах відеоспостереження:

1. Швидкодія. Системи відеоспостереження з використанням алгоритмів розпізнавання образів мають бути достатньо швидкими для одночасного відстеження десятків людей за допомогою кожної з наявних камер. І хоча iCAN дозволяє відслідковувати взаємодію людини і об'єкта у реальному часі з задовільною частотою кадрів, її масштабованість не є достатньою.
2. Ракурс. Система потребує прямого ракурсу для оптимальної роботи. Камери відеоспостереження загалом знаходяться вище рівня людини, що обмежують відстеження зорового центру уваги людини, що є найбільш критичним аспектом фізичного захисту інформаційної системи.
3. Аналіз дій людини. Не зважаючи на приголомшуючу точність опису певних ситуацій, iCAN фокусується на процесі дії (наприклад, людина тримає документ), а не на результаті (людина сховала документ у валізу), що не дозволяє на даному етапі аналізувати дії, що потенційно призводять до витоку інформації.

Описаний ряд недоліків частково буде виправлений у майбутньому зростаючими потужностями обчислювальної техніки, і в більшій мірі, подальшими активними дослідженнями в даній галузі.

2.4 Використання штучного інтелекту у системах відеоспостереження для підвищення рівня інформаційної безпеки

І хоча, системи розпізнавання образів на даний момент не можуть бути використані для розпізнавання факту фізичної інформаційної загрози. Вони можуть бути використані як допоміжний механізм вже існуючих рішень в завданнях інформаційної безпеки.

Головним завданням камер відеоспостереження є передача візуальної інформації обслуговуючому персоналу про стан зони спостереження. Оскільки об'єкти, що підлягають методам інформаційного захисту, є по своїй суті ізольованими, в більшості випадків, єдиними динамічними об'єктами є люди, що знаходяться на його території. Отже, витягування суті зображення в цьому сенсі буде виявлення факту наявності і положення людей у певній області, обмеженій полем зору камери.

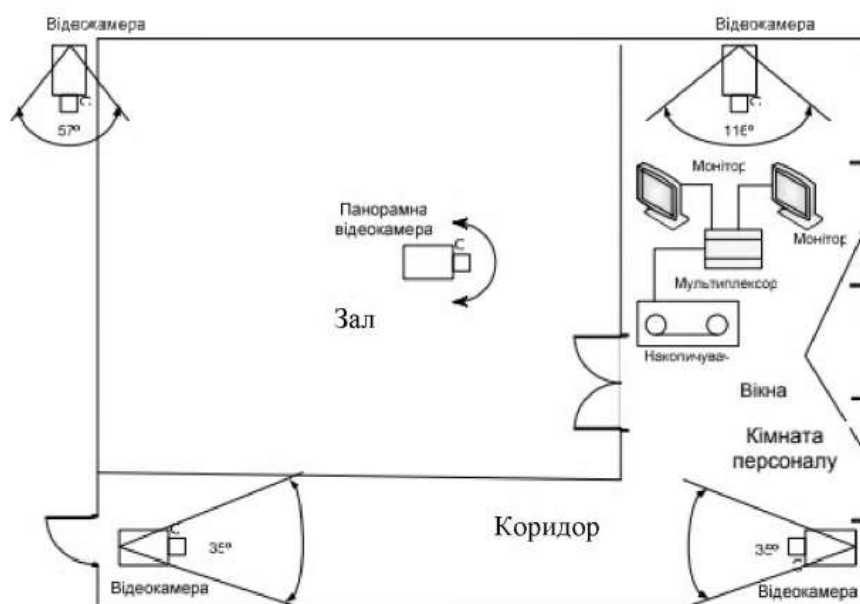


Рисунок 2.5 - Схема розташування відеокамер на захищеному об'єкті

Комп'ютерний зір дозволяє даний процес визначення описаних ознак. Більш того, такі алгоритми показали свою відносну надійність та швидкодію. Для максимізації результату роботи даного підходу необхідні наступні припущення:

1. Камера є статичною. У більшості випадків це не є проблемою, оскільки при охороні закритих приміщень перевага надається певному напрямку (рис. 2.5).
2. Дані з камер відеоспостереження зберігаються деякий час на носіях. Даний сценарій є ймовірним, оскільки у захищених об'єктів величина ризику переважає ціну закупівлі і обслуговування носіїв даних.
3. Кількість камер є достатньо великою для поставання питання про термін зберігання даних. У випадку ретроспективи успішного витoku інформації, дані з камер за відповідний період часу можуть бути знищені, що ускладнює розслідування інциденту.
4. Система відеоспостереження є доповненням до певної системи аутентифікації, наприклад смарт-карт (рис. 2.6).



Рисунок 2.6 - Система з використанням смарт-карти

У даній роботі, камери відеоспостереження також розглядаються як об'єкт інформаційної безпеки.

Для підвищення рівня інформаційної безпеки було запропоновано використання комп'ютерного зору, що виконує функції слідкування за переміщенням людей, та відповідає наступним критеріям:

1. Кожний об'єкт, що відстежується, має унікальний ідентифікатор, який отримується у момент детектування.
2. Система має коректно обробляти тимчасовий вихід об'єкта з полю зору камери, при чому час має бути регульованим.
3. Шанс втрати об'єкта, в результаті якого необхідна його повторна ідентифікація, є мінімальною.
4. Швидкодія дозволить робити обчислення у режимі реального часу та при цьому має прийнятну масштабованість.
5. Система не порушує попередні вимоги до безпеки та конфіденційності даних на різних етапах відеоспостереження.
6. Система має візуалізувати результати обробки для поліпшення роботи оператора.

У запропонованій системі можна виділити наступні ключові задачі:

1) Отримання ідентифікатора особи при її реєстрації камерами відеоспостереження у давньому підході є ключовою проблемою, оскільки його коректність та унікальність сильно впливає на роботу системи в цілому. Найкращим підходом, у цьому випадку, буде служити пункт контролю з використанням певних аутентифікаторів, наприклад зчитувача смарт-карток. Також є можливим використання засобів біометрії, проте ціна готових рішень є вищою, у порівнянні з альтернативами.

2) На території об'єктів, де наявність сторонніх осіб (наприклад тих, перебувають на території об'єкту за гостьовим посвідченням) є невідворотною, а вартість впровадження комплексних мір захисту периметру є необґрунтовано високою, є сенс у виділенні даних сторонніх осіб на моніторі оператора. Для поліпшення роботи операторів необхідним є виокремлення важливих з точки зору інформаційної безпеки подій. Даний підхід, за умови надійного алгоритму розпізнавання ознак людини та відслідковування

переміщень може забезпечити підвищення рівня використання системи відеоспостереження, оскільки постійна увага оператора буде одночасно сконцентрована на меншій кількості об'єктів. Їх переміщенням необхідно приділяти більшу увагу і приймати необхідні міри у разі їх довгострокової відсутності у полі зору засобів спостереження.

3) Також, доступ до даних аутентифікації у описаній системі є обмеженим, з причин конфіденційності та факту того, що програми-аутентифікатори в основному є сторонніми, а отже не мають чітко врегульованої структури. Вирішити це питання можна за рахунок використання API зі сторони запропонованої системи, що дозволяє використання її функціоналу у парі з даними з інших джерел шляхом використання конекторів, написаних під певний набір загальноживаних систем.

4) Для поліпшення стану інформаційної захищеності об'єкту, також було запропоновано використання даних про переміщення осіб, отриманих в результаті роботи алгоритму відстеження, як статистичних. Суть полягає у використанні метрик, що дозволяють стверджувати про потенційні вразливості певних зон захищеного об'єкту на основі наявності у даній зоні людей, що не мають необхідного рівня доступу. Дані про переміщення осіб, будучи по своїй суті, сукупністю координат точки, ідентифікатора особи та ідентифікатора камери у кожний момент часу, займають менше місця на фізичних носіях інформації, а отже підлягають можливості їх довгострокового накопичення.

5) Забезпечення можливості ручного контролю системи відслідковування переміщень. У разі втрати об'єкта спостереження, оператор повинен мати можливість власноруч вказати системі втрачений об'єкт, що підвищить рівень уваги оператора у довгостроковій перспективі а також компенсує можливі помилки самої системи.

У результаті можна сформулювати наступні фактори впливу на рівень інформаційної безпеки, описані у табл. 2.1. Окремо варто зауважити пункт 5,

оскільки збереження додаткових даних та їх витік може призвести до порушення приватності. В якості рішення даної проблеми пропонується розмежування ідентифікатора особи з камер відеоспостереження та ідентифікатора особи з системи аутентифікації.

Таблиця 2.1 - Фактори впливу на рівень інформаційної безпеки

| № | Чинник | Фактор впливу |
|----|--|---|
| 1. | Використання ідентифікатора, отриманого під час допуску особи на територію захищеного об'єкта для відстеження її подальших переміщень. | Підвищення рівня безпеки за рахунок використання додаткових засобів аутентифікації. Рівень впливу визначається ефективністю алгоритму відстеження та власне системи аутентифікації. |
| 2. | Виокремлення осіб з фону на відеозаписі, підсвічення з урахуванням певної встановленої легенди, наприклад рівня допуску. | Підвищення рівня контрольованості переміщень осіб, що становлять потенційну загрозу для витоку інформації, проте наявність яких на території об'єкту є неминучою (наприклад, особи, що знаходяться за гостьовим посвідченням). |
| 3. | Сповіщення оператора про тимчасову втрату сліду певної особи, що становить потенційну небезпеку (наприклад особи з низьким рівнем допуску) | Покращення спостережності ситуації на території об'єкта за рахунок наголошення уваги оператора на окремому секторі відеоспостереження. Рівень впливу визначається здатністю алгоритму до довгострокового відслідковування особи з низьким шансом її втрати. |

| | | |
|----|--|---|
| 4. | Сповіщення оператора про остаточну втрату сліду певної особи з можливістю власноруч відновити слід. | Покращення спостережності ситуації на території об'єкта за рахунок наголошення уваги оператора на окремому секторі відеоспостереження. Рівень впливу визначається здатністю алгоритму до довгострокового відслідковування особи з низьким шансом її втрати. |
| 5. | Збереження даних про переміщення осіб на території об'єкту у форматі вихідних даних системи комп'ютерного зору подальшого використання у статистичних цілях та при розслідуванні інцидентів. | Підвищення рівня захисту за рахунок додаткових записів про стан об'єкту. Можливість використання статистичних для їх даних для виявлення аномалій та закономірностей переміщень для аргументованого внесення змін у вже існуючу систему безпеки. |

Схема, що описує процес виконання запропонованих вимог у ході функціонування системи на території захищеного об'єкту, наведена на рис. 2.7.

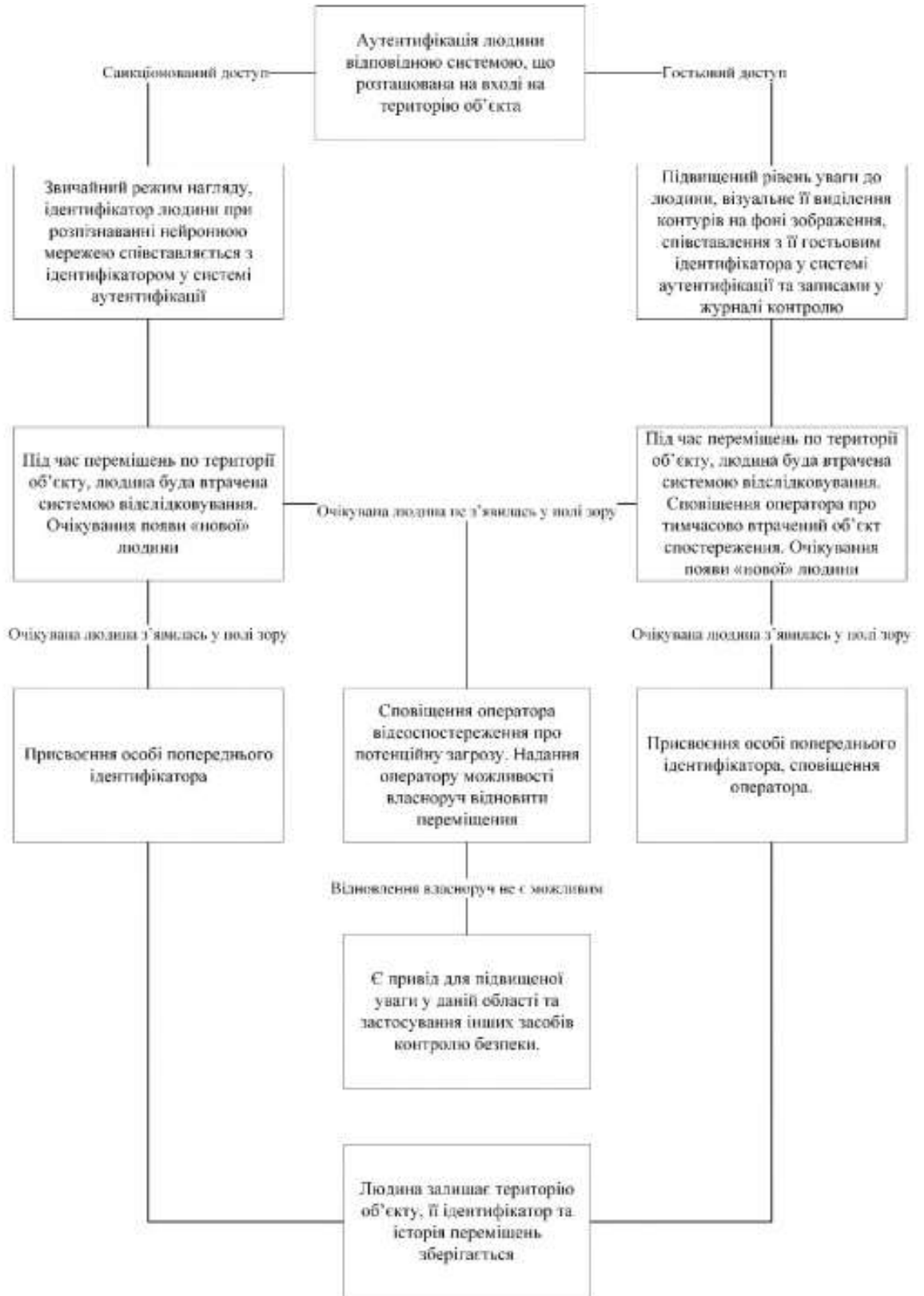


Рисунок 2.7 - Схема роботи системи відеоспостереження з використанням штучного інтелекту

2.4.1 Побудова теплових карт переміщень на основі зібраних даних

Як було зазначено, отримані дані можна використовувати у статистичних цілях. Одним із таких методів є отримання так званих теплових карт (англ. - heatmap). Теплова карта являє собою графічне зображення даних, де окремі значення, що містяться в матриці даних, представлені у вигляді відповідних градієнтів кольорів. У цьому випадку в якості даних використовується інформація з камер відеоспостереження про переміщення осіб на території захищеного об'єкта. На рис. 2.8 можна бачити приклад використання даної технології. Інтенсивність переміщень позначається більш яскравим кольором (або більш світлим відтінком у випадку чорно-білого зображення). Як можна бачити на даній теплової карті, алгоритм не розрізняє переміщення окремих людей, оскільки не має задачі до відслідковування переміщень кожної особи.



Рисунок 2.8 - Приклад теплової карти переміщень людей (відтінки сірого позначають частоту проходження)

У випадку, якщо дану технологію впроваджувати, як доповнюючу, у системи захисту, має сенс виокремлювати переміщення кожної окремої особи. У представлені запропоновані шляхи використання отриманих даних у сенсі підвищення рівня інформаційної безпеки.

Таблиця 2.2 - Шляхи використання теплових карт переміщень персоналу у сенсі системи захисту

| Шлях використання | Вплив на системи захисту |
|---|---|
| Виявлення закономірностей переміщень | <ul style="list-style-type: none"> - можливість реорганізації системи відеоспостереження з метою підвищення її продуктивності; - аналіз ефективності використання компонентів контролю доступу, з якими особи напряду взаємодіють; - можливість реорганізації персоналу з метою мінімізації переміщень та взаємодії осіб з різним рівнем доступу; - більш ефективний розподіл витрат на КСЗІ з урахуванням відмінності у рівні та джерелі загроз в залежності від часу. |
| Виявлення аномалій переміщень (окремих осіб відповідно до їх ріння доступу) | - можливість своєчасно реагувати на потенційно несанкціонований доступ. |

2.5 Висновки до розділу

У розділі була проаналізована можливість використання камер відеоспостереження, як самостійного методу захисту інформації. Для

перелічених у розділі каналів витоку інформації, на сьогодні самостійне застосування алгоритмів комп'ютерного зору в камерах в КСЗІ є недостатньо ефективним по причині відсутності як теоретичної бази з розпізнавання аномальної поведінки в термінах фізичного ЗІ, так і недостатньої потужності обчислювальних засобів.

Незважаючи на це, комп'ютерний зір з успіхом може бути використаний як допоміжний засіб для підвищення рівня захисту ІС. При поєднанні з системою аутентифікації, наприклад, смарт-картами, система відеоспостереження зі штучним інтелектом дозволить відслідковувати переміщення людей, при цьому однозначно ідентифікуючи їх, що відкриває можливості для створення нових правил безпеки а також поліпшує роботу операторів. Сценарій використання представляє собою наступні кроки:

1. Ідентифікація особи камерами відеоспостереження та надання особі ідентифікатора.
2. З використанням засобів аутентифікації, таких як смарт-карта чи біометрія, особистість людини співставляється з ідентифікатором особи, отриманим з камер відеоспостереження. Використання сторонніх засобів для аутентифікації дозволяє налаштувати різні рівні контролю, основується на рівні допуску особи, що пройшла аутентицикацію.
3. При переміщенні людини по території захищеного об'єкту, система намагається не втратити її слід, таким чином, зберігаючи ідентифікатор особи незмінним впродовж усього часу перебування.
4. При втраті сліду, система намагатиметься його відновити, виходячи з припущення про незмінну кількість осіб на території об'єкту у короткому часовому діапазоні. При неможливості відновити слід, система намагається сповістити оператора відеоспостереження та пропонує можливість власноруч виправити помилку.
5. Усі дані про переміщення зберігаються у компактній формі та можуть бути використані для збору статистики переміщення, оскільки без

додаткових ідентифікаторів по своїй суті є анонімними. У разі використання додаткових ідентифікаторів, збережені дані можна використовувати як допоміжні засоби у процесі розслідування інцидентів.

У якості основного шляху використання збережених даних про переміщення осіб, було запропоновано побудова теплових карт, що дозволять аргументоване внесення змін у вже існуючу систему. На основі теплових карт, можна робити висновки про міру ефективності використання інших систем захисту інформації і контролю доступу. Також теплові карти дають можливість до детектування аномальних переміщень конкретних осіб, тобто може виступати у ролі превентивних засобу з виявлення потенційних загроз.

Запропоновані рішення органічно доповнюють вже існуючі засоби із забезпечення безпеки, в тому числі і інформаційної. Описана система не порушує попередні вимоги до безпеки та конфіденційності даних на різних етапах відеоспостереження.

3 ОБҐРУНТУВАННЯ ВИБОРУ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ

3.1 Обґрунтування вибору програмних засобів

Питання обробки та передачі зображень, що використовуються в даній роботі, широко обговорюються в публікаціях, присвячених науковим напрямкам кафедри МІРЕС [12-29]. Серед мов програмування беззаперечними лідерами в обробці графічних зображень є Python і Java. Розглянемо коротку історію їх створення та основні бібліотеки для роботи з графічними об'єктами.

Java — це мова програмування високого рівня, яка широко використовується в сучасному світі. Він може ефективно підтримувати та обробляти цифрову обробку зображень з різними функціями [11].

Обробка цифрових зображень (DIP) стосується обробки цифрових зображень за допомогою цифрового комп'ютера. Це підполе сигналів і систем, але з особливим акцентом на зображеннях. DIP зосереджується на розробці комп'ютерної системи, здатної обробляти зображення. Входом такої системи є цифрове зображення. Система обробляє зображення за допомогою ефективних алгоритмів і генерує зображення як запит.

Java `BufferedImage` classes є підкласом `Image` classes. Він використовується для обробки та управління даними зображеннями. `BufferedImage`, створений із зображенням даних `ColorModel`. Усі об'єкти `BufferedImage` мають координату верхнього лівого кута (0, 0) [11].

За допомогою класу Клас Java `BufferedImage` можна виконати такі операції:

- завантаження і завантаження зображення;
- пікселі зображення;
- підвищення контрастності зображення;
- підвищення яскравості зображення;

- підвищення різкості зображення;
- метод спотворення зображення;
- додавання кордонів;
- базовий поріг;
- перетворення форми зображення;
- застосування фільтра Гаусса;
- застосування Box Filter;
- ерозія і розширення;
- застосування водяних знаків;
- середньозважений фільтр;
- створити ефект масштабування [11].

Python (найчастіше вживане звучання «Python», назва запозичена з британської програми Monty Python) — це інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією. Розроблено в 1990 році Гвідо ван Россумом. Високорівневі структури даних разом із динамічною семантикою та динамічним зв'язуванням роблять їх привабливими для швидкої розробки додатків і як спосіб підключення існуючих компонентів. Python підтримує модулі та пакети модулів, що сприяє модульності та повторному використанню коду. Інтерпретатор Python і стандартні бібліотеки доступні як у скомпільованій формі, так і у вихідному вигляді на всіх основних платформах. Мова програмування Python підтримує декілька парадигм програмування, зокрема: об'єктно-орієнтоване, процедурне, функціональне та аспектно-орієнтоване [11-12].

Основні переваги:

- чистий синтаксис (для виділення блоків потрібно використовувати відступи);
- портативність програми (яка характерна для більшості інтерпретованих мов);

- стандартний дистрибутив має велику кількість корисних модулів (включаючи модуль для створення графічного інтерфейсу);
- вміння використовувати Python у діалоговому режимі (дуже корисно для експериментів та вирішення простих задач);
- стандартний дистрибутив має просте, але досить ефективне середовище програмування під назвою IDLE і написане на Python;
- зручний для розв'язування математичних задач (має інструменти для роботи з комплексними числами, може оперувати цілими числами будь-якого розміру, може використовуватися як потужний калькулятор у режимі діалогу);
- з відкритим кодом (може редагуватися іншими користувачами).

Python має потужні високорівневі структури даних і простий, але ефективний підхід до об'єктно-орієнтованого програмування. Елегантний синтаксис Python, підтримка динамічних типів і той факт, що це інтерпретована мова, роблять його ідеальним для написання сценаріїв і швидкого створення програм у багатьох галузях промисловості на більшості платформ [12].

Інтерпретатор Python можна розширити функціями та типами даних, розробленими на C або C++ (або будь-якій іншій мові, яку можна викликати з C). Python також є зручною мовою розширення для програм, які потребують подальшого налагодження.

Обробка зображення – це процес аналізу та роботи з цифровим зображенням, метою якого є покращення якості зображення або вилучення інформації для подальшого використання.

Типові завдання включають відображення зображення та виконання основних операцій (обрізання, відображення, обертання, сегментація, класифікація, символи, відновлення та розпізнавання) [12].

Бібліотеки Python для роботи з зображеннями:

1. scikit-image

scikit-image — це пакет Python з відкритим кодом, який працює з масивами NumPy. Він реалізує алгоритми та інструменти для використання в наукових дослідженнях, освіті та промислових цілях. Це дуже проста і зрозуміла бібліотека навіть новачкам в екосистемі Python. Ця бібліотека містить високоякісний перевірений код, написаний активною спільнотою волонтерів [5].

2. NumPy

NumPy — одна з основних бібліотек Python для масивів. Зображення — це стандартний масив NumPy, що містить пікселі точок даних. Таким чином, виконуючи основні операції NumPy (зрізи, маски, примхливе індексування), ми можемо змінювати значення пікселів зображення. Саме зображення можна завантажити за допомогою skimage та відобразити за допомогою Matplotlib. ресурси Усі ресурси та документація доступні на офіційному веб-сайті NumPy [5].

3. SciPy

SciPy є таким же важливим науковим модулем у Python, як і NumPy. Підходить для вирішення основних завдань обробки та іншої роботи з зображеннями. Зокрема, субмодуль scipy.ndimage надає функції, які працюють з n-вимірними масивами NumPy. Поточний пакет містить функції лінійної та нелінійної фільтрації, бінарної морфології, інтерполяції B-сплайнів та вимірювання об'єктів [6].

4. PIL/Pillow

PIL (Python Imaging Library) — це безкоштовна бібліотека Python для відкриття, роботи та збереження різних форматів зображень. На жаль, його розробка згодом припинилася, а останнє оновлення вийшло в 2009 році. На щастя, є Pillow - активно розвивається вилка PIL з простою установкою. Він працює на всіх основних операційних системах і підтримує Python 3. Бібліотека містить основні функції обробки зображень, включаючи точкові операції,

фільтри з набором вбудованих ядер згортки та перетворення простору кольорів [5].

5. OpenCV-Python

OpenCV (Open Source Computer Vision Library) — одна з найпопулярніших бібліотек для програм комп'ютерного зору. OpenCV-Python — це версія інтерфейсу OpenCV на Python. Наявність коду C/C++ у серверній частині гарантує швидкість бібліотеки, а оболонка Python у зовнішній частині забезпечує легкість налаштування та впровадження. З цієї причини OpenCV-Python є чудовим рішенням для інтенсивних обчислювальних програм у сфері комп'ютерного зору [5].

6. SimpleCV

SimpleCV — ще одна платформа з відкритим вихідним кодом для створення додатків бачення. З ним ви маєте доступ до кількох потужних бібліотек комп'ютерного бачення (наприклад, OpenCV) без необхідності вивчати глибину кольорів, формати файлів, колірні простори тощо. Крива навчання набагато коротша, ніж з OpenCV, і, як вони кажуть у своєму слогані «комп'ютерний зір створений легше».

Плюси SimpleCV:

- навіть програмісти-початківці можуть написати прості тести машинного зору;
- функціональна сумісність з камерами, відеофайлами, зображеннями та відеопотоками [5].

7. Mahotas

Mahotas також є бібліотекою Python для комп'ютерного зору та обробки зображень. Він містить стандартні функції обробки зображень (фільтри та морфологічні операції), а також сучасні можливості комп'ютерного зору для обчислення характеристик (виявлення спеціальних точок і локальних дескрипторів). Швидкість програмування забезпечується інтерфейсом Python, а алгоритми C++ є плюсом для швидкості. Mahotas — це швидка бібліотека з

мінімальною кількістю коду та залежностей. Більш детально це описано в документації [6].

8. SimpleITK

ITK, або Insight Segmentation and Registration Toolkit, — це кросплатформна платформа з відкритим кодом, яка надає повний набір інструментів аналізу зображень. Подібно до SimpleITK - спрощений шар, "накладений" на ITK. Цей рівень полегшує роботу з бібліотекою для швидкого створення прототипів, вивчення та інтерпретованих мов. SimpleITK — це набір інструментів аналізу зображень із великою кількістю компонентів, які підтримують загальну фільтрацію, сегментацію та реєстрацію зображень. Сам SimpleITK написаний на C++, але доступний для багатьох мов програмування, включаючи Python [5].

9. pgmagick

pgmagick — це оболонка Python для бібліотеки GraphicsMagick. Систему GraphicsMagick іноді називають швейцарським армійським ножом обробки зображень. Він пропонує набір потужних інструментів і бібліотек, які підтримують читання, запис і обробку зображень у більш ніж 88 основних форматах, включаючи DPX, GIF, JPEG, JPEG-2000, PNG, PDF, PNM і TIFF. [5].

10. PyCairo

PyCairo — це набір прив'язок коду Python для графічної бібліотеки Cairo. Cairo — це бібліотека візуалізації двовимірної векторної графіки. Векторна графіка цікава тим, що не втрачає чіткості при зміні розміру або трансформації. PyCairo — це набір прив'язок Cairo, які можна використовувати для виклику команд Cairo з Python [5].

Для цієї роботи підходять як Python, так і Java. Ці дві мови програмування працюють з растровою графікою, але для цієї програми ми вибрали мову програмування Python. Виконувати це завдання набагато

зрозуміліше і зручніше. Має ще багато бібліотек для роботи з растровою графікою.

Python є відмінним інструментом для вирішення таких завдань. З доступністю та зростаючою популярністю Python як наукової та технічної мови програмування в екосистемі з'явилося багато найкращих інструментів обробки зображень, тому мова програмування Python буде більш підходящою.

3.2 Аналіз мови програмування Python

У світі програмування завжди є гіки, які намагаються спростити та вдосконалити популярні мови. Кращі з них змінюють хід історії. Так сталося, наприклад, з Гвідо Ван Россумом, творцем Python.

Для складних розрахунків, моделювання та побудови діаграм програміст повинен використовувати сторонні бібліотеки або, частіше, створювати власні. Це створювало багато труднощів, вивчити іншу мову було набагато легше.

Гвідо вирішив усі ці проблеми за допомогою Python. Синтаксис став дійсно високорівневим, щоб навіть не програміст міг зрозуміти код. Це в поєднанні з поширенням Інтернету дало потужний поштовх для створення вузькоспеціалізованих бібліотек, за допомогою яких можна вирішувати необмежену кількість завдань. Якщо у 1980-х кожен міг програмувати для себе, то в наступне десятиліття це стало елементом багатьох професій.

3.3 Аналіз бібліотек для роботи з растровою графікою

Для роботи із зображеннями в Python найбільш часто використовують бібліотеку PIL (Python Image Library).

Python Imaging Library (скорочено PIL) - бібліотека мови Python, призначена для роботи з растровою графікою.

Можливості бібліотеки:

- підтримка бінарних, напівтонових, індексованих, повнокольорових и СМУК зображення;
- підтримка форматів BMP, EPS, GIF, JPEG, PDF, PNG, PNM, TIFF;
- підтримка форматів (ICO, MPEG, PCX, PSD, WMF та других) тільки для читання;
- перетворення зображення з одного формату у інший.

Редагування зображень (використання різнокольорових фільтрів, масштабування, малювання, операції з матрицею тощо);
використовуючи бібліотеки з Tkinter і PyQt.

Щоб встановити бібліотеку, завантажте файл з офіційного сайту бібліотеки або встановіть з командного рядка:

```
>>> pip install pillow
```

Щоб перевірити працездатність бібліотеки і дізнатися чи вдало вона встановилася, у вікні Python Shell редактора IDLE набираємо наступний код:

```
>>> from PIL import Image  
>>> Image.VERSION
```

Якщо ви отримали відповідь, не має значення, чи інша версія бібліотеки, головне отримати відповідь, а не помилку. Отримавши версію бібліотеки, ви можете з нею працювати, тобто її успішно встановлено.

3.4 Аналіз можливостей даної бібліотеки

3.4.1 Завантаження зображення

Функція `open()` використовується для відкриття файлу зображення. Функція повертає об'єкт, через який виконується подальша робота з зображенням.

```
img = Image.open('Sunflower.jpg')
```

3.4.2 Збереження зображення

Для збереження файлу на диску з готовим зображенням використовується функція `save()`:

```
img_out = img.save('.\Images\sunflower_1_orig.jpg')
```

3.3.3 Конвертирование зображення

```
img_out = img.convert('L')
```

3.4.4 Накладання фільтрів

Якщо ми хочемо накласти якийсь фільтр або додати різкості нам потрібно залучити бібліотеку `ImageFilter`.

```
>>> from PIL import ImageFilter
```

Список можливих фільтрів:

```
from PIL.ImageFilter import (
    BLUR, CONTOUR, DETAIL, EDGE_ENHANCE, EDGE_ENHANCE_MORE,
    EMBOSS, FIND_EDGES, SMOOTH, SMOOTH_MORE, SHARPEN)
```

Висновки до розділу

У цій частині кваліфікаційної роботи було розглянуто та обґрунтовано питання вибору програмного забезпечення для обробки графічної інформації. Вибір програмних засобів обґрунтований, враховуючи мову програмування Python та бібліотеки, пов'язані з растровою графікою, конфігураціями збереження та завантаження та ін. Описано процес створення програми та реалізації графічного інтерфейсу.

4 РЕАЛІЗАЦІЯ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ

Для вирішення поставленої задачі, був використаний алгоритм Deep Sort (Deep Simpl Online and RealTime Traking algorithm), що використовує бібліотеку машинного навчання Tensorflow. В якості вхідних параметрів для нейронної мережі були використані дані, отримані F.Yu та ін. у своїй роботі [9].

Користуючись швидким розвитком розпізнавання зображень на основі CNN, алгоритм використовує його модифікацію під назвою Faster REGION CNN (FrRCNN). FrRCNN являє собою повноцінний фреймворк, що складається з двох етапів. Перший етап витягує особливі ознаки зображення і пропонує регіони для другого етапу, який потім класифікує об'єкт у запропонованому регіоні. Використання даного підходу підвищує модульність рішення, створюючи більш ефективну систему розпізнавання зображень.

4.1 Модель оцінки

У даному розділі описано об'єктну модель, тобто представлення і модель руху, що використовується для передачі рис цілі в наступний кадр. Відсутність об'єкта між кадрами розцінюється як його лінійний рівномірний рух, який не залежить від інших об'єктів та руху камери. Стан кожною цілі має наступний вигляд

$$X = [u, v, s, r, \dot{u}, \dot{v}, \dot{s}]^T \quad (4.1)$$

де u та v представляють горизонтальне та вертикальне положення центрального пікселю цілі, s та r представляють масштаб і відношення сторін

прямокутника, що обмежує ціль. Важливо зауважити, що відношення сторін приймається як константне значення. У момент детектування цілі, обмежуючий її прямокутник використовується для оновлення її стану і положення, в той час як компонент швидкості вирішується за допомогою фільтру Калмана. При відсутності відстежуваної цілі у певний момент часу, її стан оновлюється, виходячи з припущення про рівномірний прямолінійний рух.

4.2 Проектування системи

У даному розділі описана структура системи. Загалом, система складається з модулів детектування нових об'єктів, відслідковування попередньо виявлених об'єктів, візуалізатора результатів роботи та головного модуля управління. Структура системи у даному розділі описана мовою ЦМЬ за допомогою діаграми класів та діаграми компонентів.

Діаграма класів представляє собою схему статичної структури, яка описує структуру системи, показуючи класи системи, їх атрибути, операції (або методи) та взаємозв'язок між об'єктами.

Діаграма пакетів, в свою чергу, відображає залежності між пакетами, що складають модель системи. Пакет є групою елементів системи, що є згрупованими за певної логікою та ділять спільний простір імен.

На рис. 4.1 представлений клас, що відповідає за зберігання даних про виявлений об'єкт. Він містить наступні атрибути:

- коефіцієнт впевненості детектування об'єкту;
- вектор ознак, що описує об'єкт;
- координати прямокутника, що описують виявлений об'єкт.

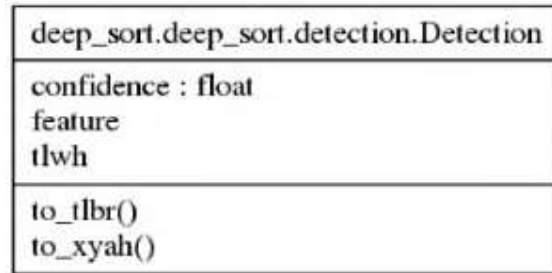


Рисунок 4.1 - Діаграма класів модуля виявлених об'єктів

На рис. 4.2 міститься клас, що описує поточний стан об'єкта. Щойно створені сліди класифікуються як «гіпотетичні», доки не знайдено достатніх доказів. Після цього, стан сліду змінюється на «підтверджено». Слід, що був знищений, позначається як «видалений» і є кандидатом на видалення.

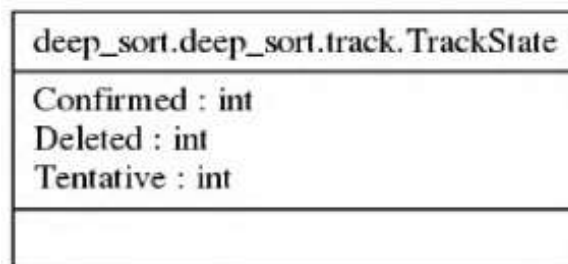


Рисунок 4.2 - Діаграма класів модуля контролю стану сліду цілі

На рис. 4.3 представлений клас, що відповідає за збереження даних про слід об'єкта спостереження та містить наступні атрибути:

- загальна кількість кадрів після першого входження об'єкта у поле зору;
- матриця коваріації початкового розподілу стану;
- набір ознак об'єкта, що оновлюється з кожним новим обчисленням • загальна кількість оновлень вимірювань;
- вектор середніх величин розподілу поточний стану;
- поточний стан сліду;
- загальна кількість кадрів після останнього оновлення вимірювань;
- унікальний ідентифікатор сліду, а отже, і самого об'єкта.



Рисунок 4.3 - Діаграма класів модуля по роботі зі слідами цілі

Модуль, зображений на рис. 4.4, відповідає за одночасне відслідковування відразу декількох об'єктів. Складовою частиною модуля є клас, відповідальний за припущення про стан об'єкта на поточний момент часу, що використовує описаний у розділі 1.4 фільтр Калмана. Власне клас обробки слідів містить наступні атрибути:

- фільтр Калмана для фільтрації цільових траєкторій у просторі зображення;
- максимальна кількість кадрів для втраченого об'єкта, перш ніж його слід буде видалено;
- максимальне значення відношення площі перетину двох прямокутників, що описують об'єкти, до їх об'єднання. Необхідна для коректної обробки перекриття одного об'єкта іншим;
- кількість кадрів, на яких слід залишається на етапі ініціалізації;
- список активних слідів на поточний момент часу

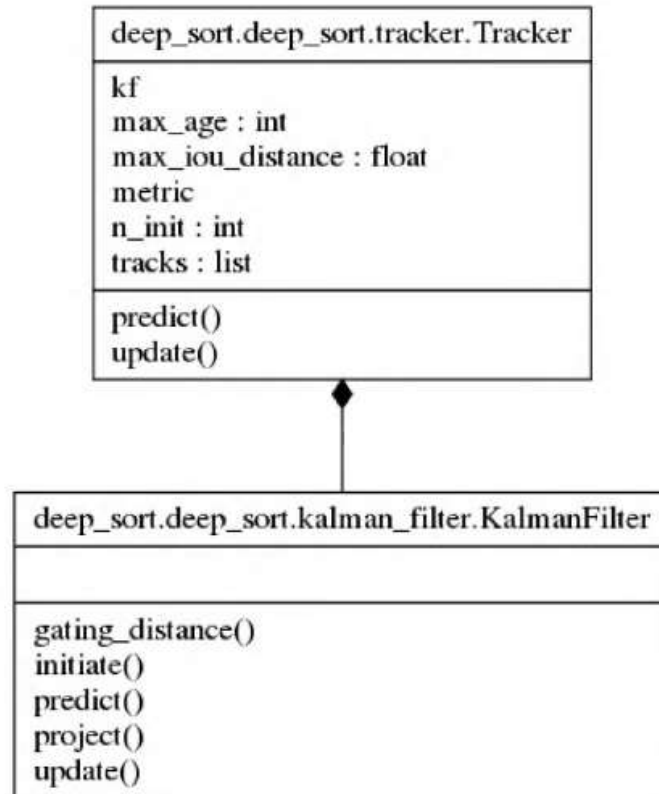


Рисунок 4.4 - Діаграма класів модуля відслідковування декількох об'єктів

Модуль візуалізації (рис. 4.5) представлений у вигляді взаємодії двох класів. Він включає в себе атрибути з налаштування відображення результату роботи. До характеристик, які відображаються поверх відеозапису належать прямокутник, що описує об'єкт та має певний колір, в залежності від налаштувань системи, а також унікальний ідентифікатор об'єкта.

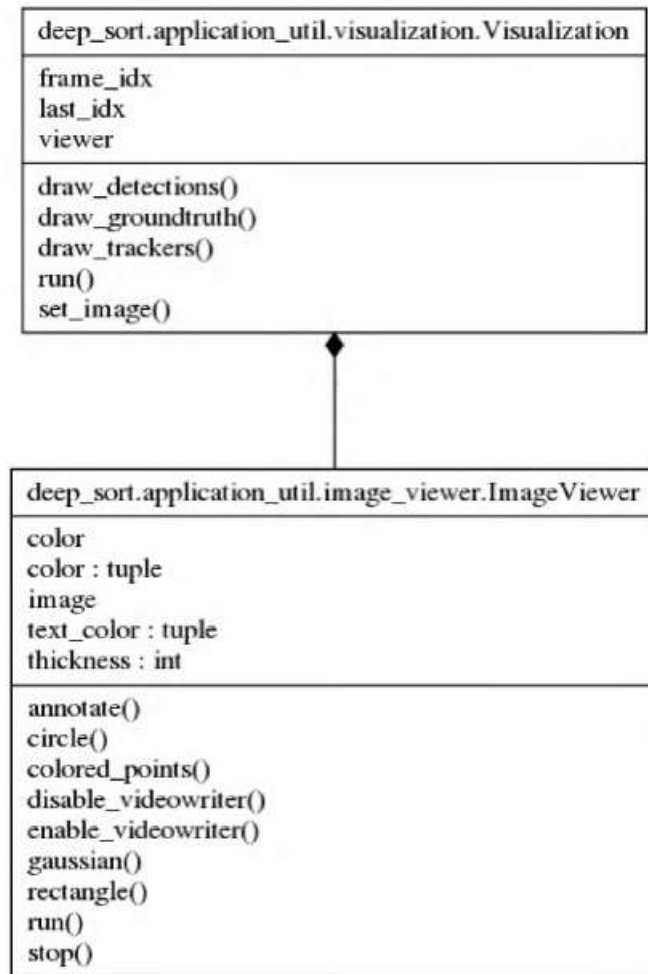


Рисунок 4.5 - Діаграма класів модуля візуалізації виявлення об'єктів

У разі, коли візуалізація результату роботи не є необхідною, існує модуль-заглушка, представлений на рис. 4.6.

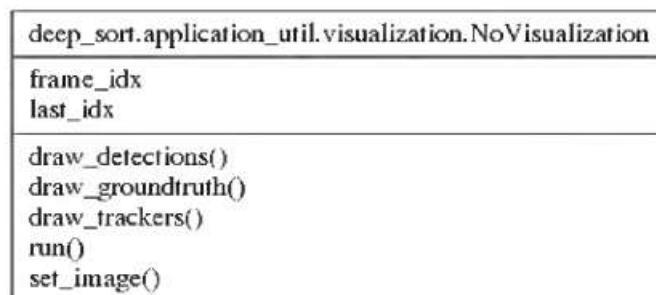


Рисунок 4.6 - Діаграма класів модуля-заглушки візуалізації виявлення об'єктів

Спроектвана діаграма пакетів, представлена на рис. 4.7, показує взаємозв'язок модулів у системі. З діаграми видно, поєднання основних компонентів системи у головний виконавчий модуль.



Рисунок 4.7 - Діаграма пакетів системи відслідковування переміщень з використанням алгоритма DeepSort

Для головного виконавчого модулю був спроектований консольний інтерфейс взаємодії. Даний інтерфейс має наступні можливості:

- зчитування відеопослідовності за вказаним місцезнаходженням;
- запису результату роботи на носій;
- виводу результату роботи на монітор користувача;
- налаштування порогового значення коефіцієнту впевненості, який впливає на чутливість системи до потенційного виявлення нових об'єктів;
- налаштування максимального часу, за який об'єкт вважається

- втраченим назавжди, а його слід видаляється;
- налаштування максимального коефіцієнту перекриття, після якого об'єкт вважається тимчасово зниклими, а для його подальше місцеположення вираховується тільки за рахунок фільтру Калмана;
 - налаштування мінімальної висоти об'єкта, що дозволяє зменшити кількість похибок першого роду у процесі детектування.

Даний інтерфейс дозволяє гнучке налаштування системи відповідно до умов експлуатації.

4.3 Функціональні особливості

У процесі детектування існуючих цілей, розміри обмежуючого прямокутника оцінюються за рахунок припущення про його нового положення у наступному кадрі. Матриця призначень вираховується як відстань Жаккара між кожним виявленим та кожним припущеним обмежуючим прямокутником для існуючих цілей. Задача оптимально вирішується за допомогою Угорського алгоритму, що був описаний у розділі 1.5. Більш того, мінімальна відстань Жаккара застосовується для відхилення припущень, де виявлене перекриття менше певного значення. Це використовується для коректної обробки сценаріїв, коли один об'єкт короткочасно перекриває інший. У такому випадку, лише об'єкт, що перекриває, буде відстежуватись, в той час як для перекритого об'єкту не буде робитись припущень.

Коли об'єкт вперше з'являється або остаточно зникає з поля зору, унікальні ідентифікатори відповідно створюються та знищуються. При створенні слідів, будь яке детектування з перекриттям, меншим за мінімальне, сприймається як вірогідна присутність невістеженого об'єкта. Власне слід створюється згідно з розмірами відповідного обмежуючого прямокутника та початковою швидкістю, рівною нулю. Оскільки об'єкт раніше не

спостерігався, коваріація його швидкості встановлюється як велике значення, відображаючи таким чином невпевненість. Більш того, нові сліди мають пройти випробний термін, для зменшення кількості похибок першого роду.

Сліди знищуються, коли відповідний об'єкт не спостерігається більше ніж T кадрів. Це запобігає неконтрольованій ріст кількості слідів та похибок, пов'язаних з припущеннями про місцезнаходження об'єкта у довгостроковій перспективі без належних виправлень зі сторони детектора. Даний підхід дозволяє калібрувати час T , основуючись на особливостях експлуатації, таких як рівень візуального покриття камерами відеоспостереження, критичність втрати сліду об'єкта спостереження та швидкодія.

4.4 Результати роботи програми

Була проведена перевірка швидкодії та ефективності з використанням згенерованого масиву даних. В нього входили відеозаписи, зняті під різним кутом, з різною роздільною здатністю та масштабом. Окрім цього, деякі з відео були записані у русі, що ускладнювало системі роботу. Результат можна бачити на рис. 4.8. Алгоритм успішно виявив усіх людей та присвоїв їм унікальні ідентифікатори.



Рисунок 4.8 - Результат роботи програми. Помірна кількість людей, динамічна камера, наявність перекриття

У наборі даних присутні моменти з великими скупченнями людей. І хоча такі ситуації далекі від реальності, коли річ йде про захищений об'єкт, даний уривок дозволяє перевірити масштабованість системи. На рис. 4.9 алгоритм намагається ідентифікувати кожну людину на платформі. Результати задовільні. При цьому швидкодія залишилася на необхідному рівні.



Рисунок 4.9 - Результат роботи програми. Велика кількість людей, динамічна камера, наявність перекриття

Також алгоритм перевірено на даних з вуличної камери, що найбільше схоже по сценарію на спостереження за певним приміщенням (рис. 4.10). Алгоритм показав задовільний результат. Варто зауважити, що люди, котрі були лише частково в полі зору камери, не були виявлені, однак це вирішується за рахунок правильного проектування системи відеоспостереження.



Рисунок 4.10 - Результат роботи програми. Статична вулична камера

4.5 Аналіз показників роботи алгоритму

Оцінка та порівняння методів багатоцільового відстеження не є тривіальною з багатьох причин. По-перше, на відміну від інших завдань, таких як зменшення рівня шуму у зображенні, у даній області важко досягти ідеального результату. Частково видимі та перекриті цілі, відображення у дзеркальних поверхнях, створюють проблеми для однозначного констатування наявності цілі спостереження. В деяких випадках, при перегляді відеозапису, навіть люди не можуть прийти згоди, а це, в свою чергу, ускладнює можливість створення даних для порівняння. По-друге, кількість різних оціночних показників з вільними параметрами та неоднозначними визначеннями часто призводить до непослідовності кількісних результатів в науковій літературі. Нарешті, відсутність заздалегідь визначених даних тесту та тренувань ускладнює неупереджене порівняння різних методів.

Одночасне відстеження великої цілей є важливою проблемою у задачах опису та розуміння контексту сцени, у якій, на відміну від інших дослідницьких областей, до нещодавна були відсутні бенчмарки та інші

методи порівняння показників роботи. З цією метою, був розроблений тест Mot Banchmark, що складається з трьох основних компонентів:

1. Публічно доступні набори даних;
2. Загальноприйнятий методи оцінки;
3. Інфраструктура, що дозволяє додавати нові методи оцінки та набори даних.

Результати роботи алгоритму Deep Sort у порівнянні з іншими алгоритмами, представленими на Mot Banchmark, наведені у табл. 4.1.

Таблиця 4.1 - Результати роботи алгоритмів у Mot Banchmark

| | MotA | MotP | MT | ML | ID | FM | FP | FN | Runtime |
|-----------|------|------|-------|-------|------|------|-------|-------|---------|
| EAMTT | 52.5 | 78.8 | 19.0% | 34.9% | 910 | 1321 | 4407 | 81223 | 12 Гц |
| POI | 66.1 | 79.5 | 34.0% | 20.8% | 805 | 3093 | 5061 | 55914 | 10 Гц |
| SORT | 59.8 | 79.6 | 25.4% | 22.7% | 1423 | 1835 | 8698 | 63245 | 60 Гц |
| Deep Sort | 61.4 | 79.1 | 32.8% | 18.2% | 781 | 2008 | 12852 | 56668 | 40 Гц |

- MotA - Загальна точність відстеження, що враховує FM, FP, ID;
- MotP - Загальна точність визначення положення цілі;
- MT - кількість в більшості вистежених траєкторій. Тобто ціль має один і той же ідентифікатор принаймні протягом 80% її тривалості життя;
- ML - кількість в більшості втрачених траєкторій. Тобто ціль не відслідковується принаймні протягом 20% її тривалості життя;
- ID - кількість разів, коли ідентифікатор об'єкта помилково переходив іншому об'єкту;
- FM - кількість разів, коли слід було перервано через відсутність виявлення;
- FP - кількість помилкових виявлень;
- FN - кількість пропущених виявлень.

Як можна бачити з табл. 4.1, алгоритм Deep Sort відповідає сучасним критеріям ефективної системи розпізнавання образів, при чому зберігає можливість його використання у режимі реального часу.

4.6 Висновки до розділу

У ході роботи була спроектована система відслідковування переміщень, з використанням алгоритму Deep Sort. Для цього, мовою UML, були система була описана за допомогою діаграм класів та діаграми пакетів, що відображають логіку роботи та взаємодії компонентів між собою. Спроектована система була реалізована мовою програмування Python з урахуванням специфіки задачі та наявності необхідного інструментарію.

Отримана реалізація системи з використанням алгоритму Deep Sort була перевірено у ряді ситуацій. Навіть при великій кількості об'єктів, що одночасно відстежуються, алгоритм показав задовільну швидкість і точність.

В якості даних для тестування були вибрані відеозаписи тривалістю від однієї до тридцяти хвилин. У цьому часовому діапазоні програма показала стабільність роботи. Жодних слідів ефектів витоку пам'яті не було виявлено. В якості об'єктивного показника ефективності алгоритму був вибраний Mot Vanchmark, система для тестування алгоритмів з одночасного відслідковування великої кількості об'єктів. Даний бенчмарк на сьогодні є єдиною масштабною платформою для тестування та порівняння роботи алгоритмів розпізнавання образів. В даному бенчмарку, використаний алгоритм продемонстрував хороші результати навіть у порівнянні з більш повільними алгоритмами, при цьому зберігаючи частоту обробки кадрів на прийнятному рівні. Найважливішим показником є порівняно низький відсоток втрат слідів об'єктів, що позитивно впливає на якість роботи системи у задачах інформаційної безпеки.

Результатом роботи став програмний продукт, що може бути застосований для описаної у розділі 2 системи. Додавання системи

розпізнавання образів не знижує, а навпаки, підвищує рівень захисту інформаційної системи, доповнюючи інші засоби ТЗІ. Точно відслідковуючи положення об'єктів та маючи низький відсоток втрат їх сліду, програма здатна записувати та зберігати ці дані у доступній для подальшого відтворення формі.

ВИСНОВКИ

Людина і вищі тварини буквально повинні розпізнавати, приймати рішення і вчитися на кожному кроці. Наука про штучні нейронні мережі виникла з бажання зрозуміти, як мозок вирішує такі складні завдання, і реалізувати ці принципи в автоматизованих системах.

Хоча штучні нейронні мережі є лише дуже спрощеними еквівалентами природних нейронних мереж, нервові системи тварин і людей набагато складніші, ніж пристрої, які можна створити за допомогою сучасних технологій. Проте навіть цього рівня достатньо для успішного вирішення багатьох практичних завдань.

Поки що серед таких завдань – завдання з розпізнавання тексту, моніторинг автомобільного руху, моделювання об'єктів 3-0 на основі набору статичних зображень, а також виявлення та відстеження переміщень об'єктів.

Саме методи відстеження об'єктів, тобто людей, засновані на відеозаписах, починають активно використовувати різні організації державного та приватного секторів. Для громадських організацій метою є забезпечення громадського порядку, тоді як приватні організації прагнуть забезпечити промислову безпеку та безпеку бізнесу. Однак у часи підвищення цінності комерційної таємниці мало уваги приділяється саме безпеці в інформаційному сенсі.

У цій главі обговорюються основні терміни та принципи розпізнавання образів, які розвинулися за останні півстоліття. Допоміжні алгоритми, такі як фільтр Калмана та угорський алгоритм, також вважалися важливими елементами майбутньої системи.

У роботі була проаналізована можливість використання камер відеоспостереження, як самостійного методу захисту інформації. Для перелічених у розділі каналів витоку інформації, на сьогодні самостійне застосування алгоритмів комп'ютерного зору в камерах в КСЗІ є недостатньо

ефективним по причині відсутності як теоретичної бази з розпізнавання аномальної поведінки в термінах фізичного ЗІ, так і недостатньої потужності обчислювальних засобів.

Незважаючи на це, комп'ютерний зір з успіхом може бути використаний як допоміжний засіб для підвищення рівня захисту ІС. При поєднанні з системою аутентифікації, наприклад, смарт-картами, система відеоспостереження зі штучним інтелектом дозволить відслідковувати переміщення людей, при цьому однозначно ідентифікуючи їх, що відкриває можливості для створення нових правил безпеки а також поліпшує роботу операторів. Сценарій використання представляє собою наступні кроки:

6. Ідентифікація особи камерами відеоспостереження та надання особі ідентифікатора;
7. З використанням засобів аутентифікації, таких як смарт-карта чи біометрія, особистість людини співставляється з ідентифікатором особи, отриманим з камер відеоспостереження. Використання сторонніх засобів для аутентифікації дозволяє налаштувати різні рівні контролю, основується на рівні допуску особи, що пройшла аутентицикацію;
8. При переміщенні людини по території захищеного об'єкту, система намагається не втратити її слід, таким чином, зберігаючи ідентифікатор особи незмінним впродовж усього часу перебування;
9. При втраті сліду, система намагатиметься його відновити, виходячи з припущення про незмінну кількість осіб на території об'єкту у короткому часовому діапазоні. При неможливості відновити слід, система намагається сповістити оператора відеоспостереження та пропонує можливість власноруч виправити помилку.
10. Усі дані про переміщення зберігаються у компактній формі та можуть бути використані для збору статистики переміщення, оскільки без додаткових ідентифікаторів по своїй суті є анонімними. У разі використання додаткових ідентифікаторів, збережені дані можна

використовувати як допоміжні засоби у процесі розслідування інцидентів.

У якості основного шляху використання збережених даних про переміщення осіб, було запропоновано побудова теплових карт, що дозволять аргументоване внесення змін у вже існуючу систему. На основі теплових карт, можна робити висновки про міру ефективності використання інших систем захисту інформації і контролю доступу. Також теплові карти дають можливість до детектування аномальних переміщень конкретних осіб, тобто може виступати у ролі превентивних засобу з виявлення потенційних загроз.

Запропоновані рішення органічно доповнюють вже існуючі засоби із забезпечення безпеки, в тому числі і інформаційної. Описана система не порушує попередні вимоги до безпеки та конфіденційності даних на різних етапах відеоспостереження.

У цій частині кваліфікаційної роботи було розглянуто та обґрунтовано питання вибору програмного забезпечення для обробки графічної інформації. Вибір програмних засобів обґрунтований, враховуючи мову програмування Python та бібліотеки, пов'язані з растровою графікою, конфігураціями збереження та завантаження та ін. Описано процес створення програми та реалізації графічного інтерфейсу.

У ході роботи була спроектована система відслідковування переміщень, з використанням алгоритму Deep Sort. Для цього, мовою UML, були система була описана за допомогою діаграм класів та діаграми пакетів, що відображають логіку роботи та взаємодії компонентів між собою. Спроектована система була реалізована мовою програмування Python з урахуванням специфіки задачі та наявності необхідного інструментарію.

Отримана реалізація системи з використанням алгоритму Deep Sort була перевірено у ряді ситуацій. Навіть при великій кількості об'єктів, що одночасно відстежуються, алгоритм показав задовільну швидкодію і точність.

В якості даних для тестування були вибрані відеозаписи тривалістю від однієї до тридцяти хвилин. У цьому часовому діапазоні програма показала

стабільність роботи. Жодних слідів ефектів витоку пам'яті не було виявлено. В якості об'єктивного показника ефективності алгоритму був вибраний Mot Banchmark, система для тестування алгоритмів з одночасного відслідковування великої кількості об'єктів. Даний бенчмарк на сьогодні є єдиною масштабною платформою для тестування та порівняння роботи алгоритмів розпізнавання образів. В даному бенчмарку, використаний алгоритм продемонстрував хороші результати навіть у порівнянні з більш повільними алгоритмами, при цьому зберігаючи частоту обробки кадрів на прийнятному рівні. Найважливішим показником є порівняно низький відсоток втрат слідів об'єктів, що позитивно впливає на якість роботи системи у задачах інформаційної безпеки.

Результатом роботи став програмний продукт, що може бути застосований для описаної у розділі 2 системи. Додавання системи розпізнавання образів не знижує, а навпаки, підвищує рівень захисту інформаційної системи, доповнюючи інші засоби ТЗІ. Точно відслідковуючи положення об'єктів та маючи низький відсоток втрат їх сліду, програма здатна записувати та зберігати ці дані у доступній для подальшого відтворення формі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Scherer, Dominik; Müller, Andreas C.; Behnke, Sven (2010). "[Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition](#)" (PDF). Artificial Neural Networks (ICANN), 20th International Conference on. Thessaloniki, Greece: Springer. pp. 92–101. [Archived](#) (PDF) from the original on 2018-04-03. Retrieved 2016-12-28.
2. Graham, Benjamin (2014-12-18). "Fractional Max-Pooling". [arXiv:1412.6071](#) [cs.CV].
3. Springenberg, Jost Tobias; Dosovitskiy, Alexey; Brox, Thomas; Riedmiller, Martin (2014-12-21). "Striving for Simplicity: The All Convolutional Net". [arXiv:1412.6806](#) [cs.LG].
4. Grel, Tomasz (2017-02-28). "[Region of interest pooling explained](#)". deepsense.io. Archived from [the original](#) on 2017-06-02. Retrieved 5 April 2017.
5. Girshick, Ross (2015-09-27). "Fast R-CNN". [arXiv:1504.08083](#) [cs.CV].
6. Ma, Zhanyu; Chang, Dongliang; Xie, Jiyang; Ding, Yifeng; Wen, Shaoguo; Li, Xiaoxu; Si, Zhongwei; Guo, Jun (2019). "Fine-Grained Vehicle Classification With Channel Max Pooling Modified CNNs". IEEE Transactions on Vehicular Technology. Institute of Electrical and Electronics Engineers (IEEE). 68 (4): 3224–3233. [doi:10.1109/tvt.2019.2899972](#). [ISSN 0018-9545](#). [S2CID 86674074](#)
7. Графічний редактор Paint [Електроний ресурс] URL: <http://www.kolomna-school7-ict.narod.ru/st30103.htm> (дата звернення: 20.12.2023).
8. Microsoft PowerPoint [Електроний ресурс] URL: https://ru.m.wikipedia.org/wiki/Microsoft_PowerPoint (дата звернення: 20.12.2023).

9. Adobe Photoshop [Електроний ресурс] URL: <http://asadmin.ru/content/view/144/>(дата звернення: 20.12.2023).
10. Інструменти Python для роботи з зображеннями [Електроний ресурс] URL: <https://medium.com/nuances-of-programming/10-%D0%B8%D0%BD%D1%81%D1%82%D1%80%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%BE%D0%B2-python-%D0%B4%D0%BB%D1%8F-%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D1%8B-%D1%81-%D0%B8%D0%B7%D0%BE%D0%B1%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%B8%D1%8F%D0%BC%D0%B8-1478612e7ade> (дата звернення: 20.12.2023).
11. Характеристики Ванана пі [Електроний ресурс] URL: <https://diytimes.ru/post/66> (дата звернення: 20.12.2023).
12. Raspberry Pi [Електроний ресурс] URL: <http://f1-it.ru/plyus-i-minus-raspberry-pi-400.html> (дата звернення: 20.12.2023).
13. Orange Pi [Електроний ресурс] URL: <http://edurobots.ru/dictionary/orange-pi-analog-raspberry/> (дата звернення: 20.12.2023).
14. Orange Pi [Електроний ресурс]: <https://myraspberry.ru/analogi-raspberry-pi.html#%D0%9F%D1%80%D0%B5%D0%B8%D0%BC%D1%83%D1%89%D0%B5%D1%81%D1%82%D0%B2%D0%B0-Orange-Pi> (дата звернення: 20.12.2023).
15. Java [Електроний ресурс] URL: <https://coderlessons.com/tutorials/akademicheskii/tsifrovaia-obrabotka-izobrazhenii-s-ispolzovaniem-java/java-dip-vvedenie-v-opencv>(дата звернення: 20.12.2023).
16. Python [Електроний ресурс] URL: <https://uk.wikipedia.org/wiki/Python> (дата звернення: 20.12.2023).
17. Фокусна відстань [Електроний ресурс] URL: <https://www.sony.ua/electronics/fokusna-vidstan-kut-ohlyadu-perspektyva>

(дата звернення: 20.12.2023).

18. A Comparative Example Between The Use Of Pca And Mds For Image Classification / Hernandez, W., Mendez, A., Flor-Unda, O., Camejo, I.M., Kolendovska, M. // IEEE International Symposium on Industrial Electronics, 29th IEEE International Symposium on Industrial Electronics, ISIE 2020; Delft; Netherlands; 17 June 2020 до 19 June 2020; Volume 2020-June, June 2020, № 9152565, Pages 1353-1358
19. Algorithm For Generating Refined Frequency Estimates In Atmospheric Radio Sounding Systems / Kartashov V., Hernandez W., Hernandez-Balbuena D., M. Kolendovska, Konovalenko O., Melnyk V. // IEEE International Symposium on Industrial Electronics, 29th IEEE International Symposium on Industrial Electronics, ISIE 2020; Delft; Netherlands; 17 June 2020 до 19 June 2020; Volume 2020-June, June 2020, № 9152562, Pages 79-82
20. Application of Fast Frequency Shift Measurement Method for INS in Navigation of Drones / D. Avalos-Gonzalez, D.H. Balbuena, V. Tyrsa, V.M. Kartashov, M. Kolendovska, S. Sheiko, O. Sergiyenko, V. Melnyk, F.N. Murrieta-Rico // IECON 2018 – 44th Annual Conference of the IEEE Industrial Electronics Society. – P. 3159–3164.
21. Avalos-Gonzalez, D., Sergiyenko, O., Hernandez-Balbuena, D., Tyrsa, V., Kartashov V.M., V., Rivas-Lopes, M., Murrieta-Rico, F.N. Constraints definition and application optimization based on geometric analysis of the frequency measurement method by pulse coincidence // Measurement: Journal of the International Measurement Confederation (USA). 2018, V.126. P. 184-193.
22. Book “Control and Signal Processing Applications for Mobile and Aerial Robotic Systems”, Hardback - Advances in Computational Intelligence and Robotics English. Edited by Oleg Sergiyenko, Moises Rivas-Lopez, Wendy Flores-Fuentes, Julio Cesar Rodríguez-Quíñonez, Lars Lindner. Editorial IGI

- Global, Hershey, United States, January 2020, 340 páginas. ISBN10 152259924X, ISBN13 9781522599241
23. Cesar Sepulveda-Valdez ; Oleg Sergiyenko ; Vera Tyrsa ; Wendy Flores-Fuentes ; Julio César Rodríguez-Quiñonez ; Fabian Natanael Murrienta-Rico ; Jesús Elías Miranda-Vega ; Paolo Mercorelli ; Marina Kolendovska. "Geometric analysis of a laser scanner functioning based on dynamic triangulation," 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 1398-1403, doi: 10.1109/ISIE45063.2020.9152268.
<https://ieeexplore.ieee.org/abstract/document/9152268>
24. Cuauhtémoc Mariscal-García; Wendy Flores-Fuentes; Daniel Hernández-Balbuena; Julio C. Rodríguez-Quiñonez ; Oleg Sergiyenko. "Classification of Vehicle Images through Deep Neural Networks for Camera View Position Selection," 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 1376-1380, doi: 10.1109/ISIE45063.2020.9152440.
<https://ieeexplore.ieee.org/abstract/document/9152440>
25. Developing and Applying Optoelectronics in Machine Vision/ O. Sergiyenko, J.C. Rodriguez-Quiñonez, IGI Global, 2016; 341p.
26. Experimental estimation of direction finding to unmanned air vehicles algorithms efficiency by their acoustic emission, /Oleynikov, V., Zubkov, O., Kartashov, V., ...Sheiko, S., Babkin, S.//2019 IEEE International Scientific-Practical Conference: Problems of Infocommunications Science and Technology, PIC S and T 2019 - Proceedings, 2019, стр. 175-178, 9061337
27. Features of acoustic noise of small unmanned aerial vehicles / Semenets, V.V., Kartashov, V.M., Leonidov, V.I. //Telecommunications and Radio Engineering (English translation of Elektrosvyaz and Radiotekhnika), 2020, 79(11), стр. 985-995

28. Geometric Analysis Of A Laser Scanner Functioning Based On Dynamic Triangulation / Sepulveda-Valdez, C., Sergiyenko, O., Tyrsa, V, Mercorelli, P., Kolendovska, M. // IEEE International Symposium on Industrial Electronics, 29th IEEE International Symposium on Industrial Electronics, ISIE 2020; Delft; Netherlands; 17 June 2020 до 19 June 2020; Volume 2020-June, June 2020, № 9152268, Pages 1398-1403
<https://ieeexplore.ieee.org/abstract/document/9152255>
<https://ieeexplore.ieee.org/document/9161870>
29. I. Y. A. Corpus, L. Lindner, O. Sergiyenko. "Transimpedance Amplifier for Laser Scanning System Range Extension," 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 1421-1426, doi: 10.1109/ISIE45063.2020.9152487.
<https://ieeexplore.ieee.org/abstract/document/9152487>
30. Ivanov, M., Sergiyenko, O., Mercorelli, P., Hernandez, W.c, Rodriguez Quinonez, J.C.d, Katashov V., Kolendovska, M., Iryna, T. Effective informational entropy reduction in multi-robot systems based on real-time TVS. IEEE International Symposium on Industrial Electronics, 2019-June, 8781209, c. 1162-1167.
31. Jonathan J. Sanchez-Castro ; Julio C. Rodríguez-Quiñonez ; Luis R. Ramírez-Hernández ; Guillermo Galaviz ; Daniel Hernández-Balbuena ; Gabriel Trujillo-Hernández ; Wendy Flores-Fuentes ; Paolo Mercorelli ; Wilmar Hernández-Perdomo ; Oleg Sergiyenko ; Félix Fernando González-Navarro. "A Lean Convolutional Neural Network for Vehicle Classification," 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 1365-1369, doi: 10.1109/ISIE45063.2020.9152274.
<https://ieeexplore.ieee.org/abstract/document/9152274>
32. Lindner, L., Sergiyenko, O., Rivas-López, M., (...), Gurko, A., Kartashov, V.M. Machine vision system for UAV navigation; IEEE, 2016 International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and

- Road Vehicles and International Transportation Electrification Conference, ESARS-ITEC, 2016; pp.1–6. DOI: 10.1109/ESARS-ITEC.2016.7841356.
- 33.M. Ivanov, O. Sergiyenko, V. Tyrsa, P. Mercorelli, V. Kartashov, W. Hernandez, S. Sheiko, M. Kolendovska. Individual scans fusion in virtual knowledge base for navigation of mobile robotic group with 3D TVS // Proceedings of 44th Annual Conference of IEEE Industrial Electronics Society (IECON).. -2018. – Washington DC, USA. -S. 3187-3192. . ISBN 978-1-5090-6683-4/18/.
- 34.Murrieta-Rico, F.N., Petranovskii, V., Galvan, D.H., Sergiyenko, O., Yocupicio-Gaxiola, R.I., De Dios Sanchez-Lopez, J. Phase effect in frequency measurements of a quartz crystal using the pulse coincidence principle. 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 185-190, 9152255, DOI: 10.1109/ISIE45063.2020.9152255
- 35.Oleksandr Sotnikov, Vladimir Kartashov, Oleksandr Tymochko, Oleg Sergiyenko, Vera Tyrsa, Paolo Mercorelli, Wendy Flores-Fuentes. Methods for Ensuring the Accuracy of Radiometric and Optoelectronic Navigation Systems of Flying Robots in a Developed Infrastructure. Chapter 16// Machine Vision and Navigation; Springer, Cham. pp.537–578. Editors: Sergiyenko, Oleg, Flores-Fuentes, Wendy, Mercorelli, Paolo. DOI: 10.1007/978-3-030-22587-2_16.
- 36.Optical detection of unmanned air vehicles on a video stream in a real-time/Kartashov, V., Oleynikov, V., Zubkov, O., Sheiko, S.// 2019 International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019 - Proceedings, 2019, 9165362/
- 37.Principles Of Construction And Assessment Of Technical Characteristics Of Multi-Frequency Atmospheric Sodar In The Humidity Measurement Mode / Kartashov, V.M., Sidorov, G.I., Sheiko, S.A., Kolendovskaya, M.M., Sergienko, O.Yu. // Telecommunications And Radio Engineering

- (English Translation Of *Elektrosvyaz And Radiotekhnika*), 2020, ISSN Print: 0040-2508, ISSN Online: 1943-6009, DOI: 10.1615/TelecomRadEng.v79.i4.50, p. 323-333/
38. Research Of The Uncertainty Of Measurement Frequencies And Definitions Of The Frequency Signal In The Waveguide With Respect To Power / Semenets, V.Zakharov, I. Serhienko, M., Kartashov, V.M, , Kolendovska, M., Hernandez, W., Hipolito, J.I.N., , Tyrsa, V.// 45th Annual Conference of the IEEE Industrial Electronics Society, IECON 2019; Lisbon Congress CenterLisbon; Portugal; 14 October 2019 до 17 October 2019; CFP19IEC-ART; Код 155980, Volume 2019-October, October 2019, № 8927203, Pages 4674-4679
39. Spatial-Temporal Processing Of Acoustic Signals Of Unmanned Aerial Vehicles /Kartashov V.M., Oleinikov V.N., Zubkov O.V., Sheiko S.A., Kolendovska M.M.// *Telecommunications And Radio Engineering (English Translation Of Elektrosvyaz And Radiotekhnika)*, 2020, ISSN Print: 0040-2508, ISSN Online: 1943-6009, DOI: 10.1615/Telecomradeng.v79.i9.40, p. 769-780
40. Stereoscopic Vision Systems In Machine Vision, Models, And Applications (Book Chapter)/ Ramírez-Hernández, L.R., Rodríguez-Quiñonez, J.C., Castro-Toscano, M.J., Kolendovska, M., Murrieta-Rico, F.N.// *Machine Vision And Navigation*, 2019 *Machine Vision and Navigation* 30 September 2019, Pages 241-265
41. Strelkova T., Kartashov V., Lytyuga A., Strelkov A. Theoretical Methods of Images Processing in Optoelectronic Systems. Chapter 16. // *Biometrics: Concepts, Methodologies, Tools, and Applications*; Oleg Sergiyenko and Julio C. Rodriguez-Quiñonez. (341p.), IGI Global, 2017; pp. 361-381. DOI: 10.4018/978-1-5225-0983-7.ch016.
42. Strelkova T., Kartashov V., Lytyuga A., Strelkov A. Theoretical Methods of Images Processing in Optoelectronic Systems. Chapter 6// *Developing and*

- Applying Optoelectronics in Machine Vision; Oleg Sergiyenko and Julio C. Rodriguez-Quiñonez. (341p.) – USA, Herhey, IGI Global, 2016; pp.180-205.
43. Sytnik O., Kartashov V. Methods and Algorithms for Technical Vision in Radar Introscopy. Chapter 13// Optoelectronics in Machine Vision-Based Theories and Applications. IGI Global, 2019; pp. 373-391.
44. The Use of Factorization and Multimode Parametric Spectra in Estimating Frequency and Spectral Parameters of Signal/Semenets, V., Kartashov, V., Sergiyenko, O., ...Rodriguez-Quinonez, J.C., Flores-Fuentes, W.//IEEE International Symposium on Industrial Electronics, 2020, 2020-June, p. 215-219
45. Unda, O.F., Hernandez, W., Vargas, O., Mendez, A., Sergiyenko, O., Tyrsa, V. Construction of a robotic platform of differential type for first-year students of electronic engineering, 2020 International Symposium on Power Electronics, Electrical Drives, Automation and Motion, SPEEDAM 2020, 24-26 de junio de 2020, Sorrento, Italia, pp. 538-543, 9161870, DOI: 10.1109/SPEEDAM48782.2020.9161870
46. Use of Acoustic Signature for Detection, Recognition and Direction Finding of Small Unmanned Aerial Vehicles/Kartashov, V., Oleynikov, V., Koryttsev, I., ...Babkin, S., Selieznov, I.//Proceedings - 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2020, 2020, p. 377-380/
47. V. Semenets; Vladimir Kartashov ; Oleg Sergiyenko; Vyacheslav Tikhonov ; Paolo Mercorelli ; Sergiy Sheiko ; Nataliya Chmelarova. "The Use of Factorization and Multimode Parametric Spectra in Estimating Frequency and Spectral Parameters of Signal," 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 215-219, doi: 10.1109/ISIE45063.2020.9152238.
<https://ieeexplore.ieee.org/abstract/document/9152238>
48. Wilmar Hernandez ; Alfredo Mendez ; Omar Flor-Unda ; Vicente Gonzalez-Posada ; Jose Luis Jimenez ; Oleg Sergiyenko ; Julio C. Rodriguez-Quiñonez

- ; Mykhailo Ivanov ; Ivan Menes Camejo ; Marina Kolendovska. "A comparative example between the use of PCA and MDS for image classification," 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, Netherlands, 17-19 of June 2020, pp. 1353-1358, doi: 10.1109/ISIE45063.2020.9152565.
<https://ieeexplore.ieee.org/abstract/document/9152565>
- 49.Карташов В.М. и др. Обработка сигналов в радиоэлектронных системах дистанционного мониторинга атмосферы; Харьков: ХНУРЭ, 2014. 312 с.
- 50.Ситнік О.В., Карташов В.М. Радіотехнічні системи. Навч. посібник. Х.: Сміт, 2009. 448 с.
- 51.Офіційний сайт бази даних візуальних новел. [Електронний ресурс]
URL: <https://vndb.org/v?f=&o=a&p=12&s=rel> (дата звернення: 17.05.2022).