

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Центр післядипломної освіти
(повна назва)
Кафедра _____ Програмної інженерії
(повна назва)

АТЕСТАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти – другий (магістерський)

Дослідження методів та алгоритмів визначення границь зображень
(тема)

Виконав: студент _____ 2 курсу, гр. ПЗмзд-18-1
Романюк О. І.

(прізвище, ініціали)

спеціальності 121 – Інженерія програмного забезпечення
(код і повна назва спеціальності)

освітньо-наукової програми

(тип програми)

Інженерія програмного забезпечення

(повна назва освітньої програми)

Керівник _____ к.т.н., доцент Лановий О. Ф.

(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри, проф.

_____ (підпис)

З.В.Дудар
(прізвище, ініціали)

2020 р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет	Центр післядипломної освіти
Кафедра	Програмної інженерії
Рівень вищої освіти	– другий (магістерський)
Спеціальність	121 – Інженерія програмного забезпечення (код і повна назва)
Тип програми	освітньо-наукова програма
Освітня програма	Інженерія програмного забезпечення (код і повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«_____» _____ 2020 р.

ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

студентові _____ Романюку Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів та алгоритмів визначення границь зображень

затверджена наказом по університету від _____ “_____” _____ 20 _____ р. № _____

2. Термін подання студентом роботи до екзаменаційної комісії 20 травня 20203. Вихідні дані до роботи математичні моделі перетворювання зображень, література. Перелік використовуваних програмних засобів: ОС Microsoft Windows, зображення.

4. Перелік питань, що потрібно опрацювати в роботі:

опис проблеми, існуючі методи, опис існуючих алгоритмів,модифікація існуючих алгоритмів, методи реалізації алгоритма,методи оптимізації, адаптація методів сегментації до умов задачі,структурна схема комплексного алгоритму обробки і аналізу зображень,розробка архітектури та впровадження створеної математичної моделі,програмне застосування алгоритму

5. Консультанти розділів роботи

Найменування розділу	Керівник (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання	23.01.2020	
2	Огляд і аналіз сучасного стану проблеми	28.02.2020	
3	Огляд існуючих методів	15.03.2020	
4	Розробка алгоритмів визначення контурів	08.04.2020	
5	Розробка моделі прототипу системи	15.04.2020	
6	Підготовка пояснювальної записки	01.05.2020	
7	Підготовка презентації та доповіді	12.05.2020	
8	Попередній захист	15.05.2020	
9	Нормоконтроль, рецензування	15.05.2020	
10	Занесення диплома в електронний архів	18.05.2020	
11	Допуск до захисту у зав. кафедри	20.05.2020	

Дата видачі завдання 23.01.2020 р.

Студент _____
(підпис)

Романюк О. І.

(прізвище, ініціали)

Керівник роботи _____
(підпис)

к.т.н., доцент Лановий О. Ф.

(посада, прізвище, ініціали)

РЕФЕРАТ

Атестаційна робота магістра містить: 74 с., 5 рис., 5 табл., 30 джер.

РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ, СИСТЕМА КЕРУВАННЯ, ТЕОРІЯ РОЗПІЗНАВАННЯ ОБРАЗІВ, ТЕХНІЧНИЙ КОНТРОЛЬ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ.

Метою роботи є розробка автоматизованої моделі розпізнавання зображень для систем відеонагляду.

Методи розробки базуються на інструментах розробки веб-застосунків на платформі .Net, бібліотеки WPF та протоколу передачі даних HTTP.

В результаті роботи розглянуто методи трасування контурів, алгоритми визначення контурів зображення та сегментації зображення, розроблено програмну реалізацію системи автоматизованого визначення контурів зображення в системі аналізу відеорядів, яка представляє собою веб-додаток на основі платформи .Net.

IMAGE RECOGNITION, CONTROL SYSTEM, IMAGE IDENTIFICATION THEORY, TECHNICAL CONTROL, SOFTWARE, INTELLECTUAL AND INTELLECTUAL.

The purpose of the work is to develop an automated model of image recognition for video surveillance systems.

Development methods are based on .Net application web development tools, WPF library, HTTP data transfer protocol.

As a result of the work, algorithms of contour tracing and segmentation have been developed together with a program implementation of the above algorithms as part of the system of video analysis. The system is represented by a web application based on the .Net platform.

ЗМІСТ

С.

Перелік умовних скорочень	7
Вступ.....	8
1 Аналіз предметної галузі	11
1.1 Опис проблеми	11
1.2 Існуючі методи	16
1.3 Постановка задачі дослідження	23
2 Аналіз існуючих алгоритмів	26
2.1 Опис існуючих алгоритмів.....	26
2.2 Методи реалізації алгоритма	28
3 Особливості моделювання засобів розпізнавання образів на цифровому відеоряді	32
3.1 Методи оптимізації	32
3.2 Адаптація методів сегментації до умов задачі.....	36
3.3 Алгоритм трасування контурів зображення.....	37
3.4 Реалізація моделі	39
3.5 Модель прототипа системи.....	48
4 Практична реалізація проекту розпізнавання зображень для системи відеонагляду.....	51
4.1 Структурна схема комплексного алгоритму обробки і аналізу зображень ...	51
4.2 Розробка архітектури та впровадження створеної математичної моделі в автоматичний процес відеоспостереження	54
4.3 Програмне застосування алгоритму визнаення контурів у системі відеонагляду за допомогою системи розпізнавання зображень	57
4.4 Рекомендації з використання модулю визначення контурів в системі відеоспостереження	60
Висновки	63
Перелік джерел посилання	65

Додаток А Структурний алгоритм інформаційних потоків	68
Додаток Б Слайди презентації	69

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БЗ – база знань;

АСВ – автоматизована система відеоспостереження;

ТРО – теорія розпізнавання образів;

ІТ – інтелектуальні інформаційні технології;

ІБД – інтегрована база даних;

ЕОМ – електронно-обчислювальна машина;

ЕОД – електронний обмін даними;

ССТV – Система відеоспостереження.

ВСТУП

Актуальність дослідження. Контур визначається як сегмент шириною один піксель і довжиною один або кілька пікселів, а межа визначається як безперервний контур. Контури і межі надають дуже важливу інформацію для подання об'єктів і розпізнавання зображень. Наприклад, вони використовуються для відділення об'єктів від фону, для розрахунку розмірів об'єктів, для класифікації форм і для знаходження характерних точок об'єктів, використовуючи довжину і форму їх контурних пікселів. Крім того, в області графіки та зору можна використовувати інформацію про контур, щоб зберегти форму об'єктів і відновити їх вихідну форму для різних застосувань.

Тому було проведено багато досліджень алгоритмів трасування контурів для вилучення і відстеження контурів об'єкта. Більшість алгоритмів є бінарними алгоритмами трасування контурів зображень, які відстежують контури на оцифрованих чорно-білих зображеннях, отриманих з різних датчиків зображення.

В останні роки, з ростом популярності інтелектуальних / портативних мобільних сенсорних пристроїв, таких як смартфони, розумні годинник і розумні окуляри, різних додатків реального часу, таких як розпізнавання коду зображення, розпізнавання особи, оптичне розпізнавання символів, розпізнавання логотипу, доповнена реальність і змішана реальність стали актуальними для портативних пристроїв. Оскільки портативні мобільні сенсорні пристрої володіють обмеженими апаратними ресурсами, такими як низько продуктивні процесори, пам'ять невеликого розміру, дисплеї з низькою роздільною спроможністю і низька ємність акумулятора, їм потрібні прості і швидкі алгоритми обробки зображень.

Мета дослідження – метою дослідження є розробка автоматизованої моделі розпізнавання зображень для систем відеонагляду.

Для досягнення мети були поставлені наступні завдання:

– провести огляд і аналіз сучасного стану проблеми розпізнавання образів;

- виконати огляд математичних моделей та методів моделювання розпізнавання образів;
- виділити особливості існуючих методів та алгоритмів розпізнавання образів;
- сформулювати постановку мети та задач дослідження роботи;
- обрати та або удосконалити засоби математичного забезпечення для розпізнавання образів;
- розробити модель програмного засобу для розпізнавання зображень в системі відеоспостереження;
- оцінити точність результатів запропонованого алгоритму розпізнавання образів.

Об'єкт дослідження – проектування системи трасування контурів зображень в рамках системи аналізу відеорядів.

Предмет дослідження – особливості застосування методу розпізнавання образів для систем відеонагляду.

Методи дослідження:

- методи системного аналізу;
- аналіз наукової літератури;
- спостереження;
- абстрагування;
- узагальнення.

Теоретично – інформаційну базу дослідження склали праці таких науковців, як Miyatake T., Matsushima H., Ejiri M., Rosenfeld A., Khanykov I. G., Kharinov M. V., Patel C. та інших.

Наукова новизна одержаних результатів. Результати дослідження пропонують альтернативний алгоритм та систему в цілому для трасування контурів зображень в рамках системи аналізу відеорядів.

Практичне значення одержаних результатів полягає в тому, що дослідження ґрунтується на результатах поглибленого вивчення особливостей застосування

системи розпізнання образів, а також запропоновано новий підхід для розпізнавання контурів зображень на пристроях з обмеженими ресурсами.

Структура роботи визначена логікою його дослідження та складається зі вступу, основної частини із чотирьох розділів, висновків, переліку джерел посилань, додатків. Загальний обсяг роботи становить 74 сторінок. Перелік джерел посилання складається із 30 найменувань.

Оформлення пояснювальної записки виконано відповідно рекомендацій та вимог ДСТУ 1.5:2015.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Опис проблеми

Аналіз форми об'єктів відіграє значну роль в багатьох дослідженнях. Зокрема, зміна форми об'єкта може сигналізувати про його перехід з одного стану в інший (наприклад, в процесі розвитку хвороби). У природі форма об'єкта визначається його контури (клітинні стінки, зовнішній шар і т. д.).

Трасування контуру – це етап прийому дискретного сигналу опис меж цифрового об'єкта.

Вимоги до алгоритмів трасування контурів:

- скорочення об'єму пам'яті;
- скорочення часу і складності подальшої обробки;
- отримання інформаційних особливостей про об'єкт.

Окремий контур можна отримати двома способами: підкреслюючи кордон об'єкта шляхом фільтрації вхідних даних зображення або шляхом передачі внутрішнього контуру однорідної області.

Основні алгоритми трасування кордонів об'єкту: алгоритм змії, алгоритм Канни, на основі фільтрації Собеля, Лапласа, Преуїтту і інші.

Вони засновані на підкреслюючих різких перепадах яскравості, які характеризують межі об'єктів. Результатом їх роботи є це набір не пов'язаних між собою областей. Щоб отримати зв'язок контур, необхідно провести додаткову обробку.

У самих різних областях науки і техніки є безліч завдань, пов'язаних з обробкою зображень. Це, наприклад, аналіз і інтерпретація аерофотознімків і космічних знімків в науках про Землю, в метеорології, астрономії, дослідженні природних ресурсів; аналіз оптичних, рентгенівських, теплових, радіографічних та інших зображень в медичній діагностиці, промислової дефектоскопії, наукових

дослідженнях; аналіз зображень і полів, зареєстрованих методами оптичної, радіо- і акустичної голографії і т. д.

До недавнього часу фахівці відповідних областей, вчені, дешифровщики аерофотознімків, геологи, фотограмметрист, лікарі-рентгенологи і інші в рішенні цих завдань могли покладатися тільки на свій зір і працювали вручну. Зростаючий обсяг завдань і підвищення вимог до точності і швидкості їх вирішення викликали інтенсивний розвиток засобів і методів автоматизації обробки зображень.

В даний час цифровою обробкою зображень починає займатися все більше число фахівців з різними підходом, рівнем підготовки, навичками.

Під зображенням будемо розуміти двовимірну функцію, де x і y – координати в просторі, і значення якої в будь-якій точці, що задається парою координат, називається інтенсивністю або рівнем сірого зображення в цій точці. Якщо величини, x і y приймають кінцеве число дискретних значень, то говорять про цифрове зображення. Цифровою обробкою зображень називається обробка цифрових зображень за допомогою цифрових обчислювальних машин (комп'ютерів). Цифрове зображення складається з кінцевого числа елементів, кожен з яких розташований в конкретному місці і приймає певне значення. Ці елементи називаються елементами зображення або пікселями. Найчастіше для елементів цифрового зображення використовується термін «піксель».

Зір є найбільш досконалим з наших органів почуттів, тому не дивно, що зорові образи грають найважливішу роль в людському сприйнятті. Однак, на відміну від людей, здатних сприймати електромагнітне випромінювання лише у видимому діапазоні, машинна обробка зображень охоплює практично весь електромагнітний спектр від гамма-випромінювання до радіохвиль. Оброблювані зображення можуть породжуватися такими джерелами, які для людини незвично пов'язувати з зображеннями. Так, наприклад, ультразвукові зображення; зображення, одержувані в електронній мікроскопії або генеруються комп'ютером. Таким чином, цифрова обробка зображень охоплює широкі і різноманітні області застосування.

Не існує загальноприйнятої точки зору, де закінчується обробка зображень і починаються інші суміжні галузі, наприклад, аналіз зображень і машинний зір. Іноді розмежування робиться тут за тим принципом, що обробка зображень визначається як дисципліна, в якій на вході і на виході процесу присутні зображення. Таке виокремлення є обмеженим і частково штучним. Наприклад, в рамках цього визначення навіть така тривіальна задача, як визначення середньої інтенсивності по полю зображення, не може розглядатися як операція обробки зображення. З іншого боку, існують такі області, як машинний зір, де кінцевою метою є комп'ютерна імітація людського зору, включаючи навчання, здатність до умовиводів і діям на основі спостережуваної інформації. Ця область сама по собі утворює лише один із напрямів штучного інтелекту, метою якого є імітація інтелектуальної діяльності людини. Штучний інтелект знаходиться ще на початковій стадії свого розвитку. Область, пов'язана з аналізом зображень займає проміжне положення між обробкою зображень і машинним зором.

У всьому діапазоні від обробки зображень до машинного зору немає чітких меж, проте, розрізняють в ньому комп'ютеризовані процеси низького, середнього і високого рівня. Процеси низького рівня стосуються тільки примітивних операцій типу передобробки з метою зменшення шуму, підвищення контрасту або поліпшення різкості зображень. Для низькорівневих процесів характерний той факт, що на вході і на виході присутні зображення. Обробка зображень на середньому рівні охоплює такі завдання, як сегментація (розподіл зображення на області або виділення на ньому об'єктів), опис об'єктів і стиснення їх в зручну для комп'ютерної обробки форму, а також класифікація (розпізнавання) окремих об'єктів. Для процесів середнього рівня характерна наявність зображень тільки на вході, на вихід роблять ознаки і атрибути, які добувають із цих зображень (наприклад, межі областей, лінії контурів, відмінні ознаки конкретних об'єктів). Високорівнева обробка включає в себе «осмислення» набору розпізнаних об'єктів, як це робиться в аналізі зображень.

Лінії контуру – це основна та внутрішня інформація, необхідна для представлення форми фігур у таких широких областях застосування, як розпізнавання символів, розпізнавання карт та малюнків, і навіть розпізнавання об'єктів 3D для машинного зору. Було багато дослідницьких робіт щодо алгоритмів контурного відстеження, які генерують набори піксельних координат на межах фігур у цифрових бінарних зображеннях, що скануються до растру. До них відносяться алгоритми, які дозволяють відновити вихідні фігури саме з отриманих наборів координат, а також алгоритми, які ефективно відстежують контур від крайових положень на кожній лінії сканування. Тим не менш, існує небагато алгоритмів, які задовольняють обидві вимоги: точність та ефективність. Звичайні алгоритми час від часу вимагають великої буферної пам'яті зображення як робочої області, що легко знижує швидкість контуру трасування, особливо для кодування високої чіткості великомасштабних зображень, таких, що використовуються в галузі розпізнавання документів та рисунків.

Спочатку треба визначити деякі вирази, необхідні для пояснення звичайних алгоритмів контурного відстеження. У цій роботі розглядаються лише бінарні зображення. Зображення розділене на сітки для формування квадратних пікселів, кожна з одиниці довжини 1 у горизонтальному та вертикальному напрямках. Чорні пікселі на рисунку мають логічне значення 1 на відміну від логічного значення 0 для пікселів білого фону. Точки переходу визначаються як точки, де значення пікселів змінюються від 1 до 0 або 0 до 1, коли зображення сканується в растровому напрямку (зліва направо). Точки переходу з фону до фігури (0 до 1) визначте лівий край (фігури), а точки переходу від фігури до фону (1 до 0) визначте правий край (фігури). Ці краї вважаються посередині між сусідніми пікселями (тобто на межі чорно-білих квадратних пікселів), і таким чином координати точок переходу зміщуються на 0,5 від піксельних центрів. Пара, що складається з лівого краю і правого краю, називається "пробіг". Контурні точки визначаються як чорні точки, що належать фігурі, сусідні пікселі включають принаймні один білий піксель. Набір контурних точок утворює контурну лінію.

Три типових алгоритми контурного відстеження представлені нижче:

а) піксельне центральне трасування. Цей алгоритм є найпопулярнішим і простежує контур певним чином, наприклад, в напрямку проти годинникової стрілки. Трасування починається в першій точці контуру, що виникає при скануванні зображення в растровому режимі, і закінчується, коли початкова точка контуру переглядається після відстеження замкнутого циклу. Під час трасування кожна точка контуру, щойно відвідана, позначається, щоб уникнути подвійного відстеження. Отриманий цикл є контурною лінією, і растрове сканування починається знову, щоб знайти ще одну непомічену початкову точку;

б) піксельне кутове трасування. Цей алгоритм розглядає кожен піксель як квадрат, а кути квадрата кожної точки контуру підлягають відстеженню. Відстеження починається в піксельному куті без позначення та відвідує сусідні кути послідовно, поки знову не досягне початкового кута. Отримана петля являє собою контур, який складається тільки з вертикального та горизонтального векторних сегментів;

в) трасування крайових точок. Цей алгоритм застосовується до зображення, вираженого набором даних про виконання. Дві сусідні горизонтальні лінії сканування зазвичай перевіряються, щоб побачити, чи прогони в обох лініях сканування належним чином пов'язані між собою. Потім крайові точки двох прогонів з'єднуються для опису контуру.

Алгоритми а) і б) вимагають досить великої пам'яті зображення як буферу кадру, щоб впоратися зі складними фігурами на масштабних малюнках. Це означає, що в цих алгоритмах швидкість контурного трасування по суті є повільною. Якщо недостатня пам'ять зображення недоступна, обмін даними зображення відбувається часто, а швидкість ще більше знижується. Крім того, алгоритм а) зазвичай створює помилкову форму при збільшенні зображення, особливо на ділянці, що має ширину одного пікселя.

Крім того, алгоритм в не завжди може відновити початкову форму, і викривлення відбувається на увігнутій частині фігури. Вони вказують на те, що

точність контурів в алгоритмі а) і б) недостатня для того, щоб точно представити вихідні дані.

Вимоги до алгоритмів трасування контуру: зменшення місця для зберігання; скорочення часу та складності подальшої обробки; отримання інформативних особливостей об'єкту. Окремий контур може проходити двома способами: виділяючи межі об'єкта, фільтруючи вхідне зображення або пропускаючи внутрішній контур однорідної області.

Основними алгоритмами вибору границь об'єкту є алгоритм змії, алгоритм Кенні, фільтрація на основі Собеля, Лапласа, Превітта та інших. Вони ґрунтуються на виділенні різких крапель яскравості, характерних для меж предметів. Результатом їх роботи є сукупність непок'єднаних ділянок. Для отримання з'єднаного контуру необхідно провести додаткову обробку.

Алгоритми для вибору областей: порогова сегментація, кластеризація, зростаючий регіон, алгоритм вододілу, сегментація блоків тощо. Вони ґрунтуються на пікселях об'єднання в однорідних регіонах на основі певного критерію однорідності. Результатом їх роботи є сукупність однорідних ділянок. Для отримання опису контуру об'єкта необхідно використовувати алгоритми відстеження контуру.

1.2 Існуючі методи

Процес автоматизації проектування являє собою систему з використання електронно-обчислювальних машин під час створення та обрання способів вирішення різнопланових задач технологічного смислу [7].

Проектування поділяється на три види:

- неавтоматизоване;
- автоматизоване;

– автоматичне.

В основі неавтоматизованого створення лежить такий алгоритм людських дій [2]:

- трансформація опису об'єктів;
- проектування процесів функціональних характеристик об'єкта;
- модельювання алгоритму процесу;
- процес опису на різних мовах.

В процесі автоматизованої розробки вище означений алгоритм відбувається за допомогою застосування обчислювальних машин, тому, очевидно, що процес відбувається без залучення людських ресурсів. За рівнем поглибленості розробок, рівні процесу створення проекту можна поділити на такі:

- створення принципів схем ТП;
- розробка технологічної схеми обробки деталей;
- розробка технологічних механізмів;
- розробка коду для устаткування із числовим програмним управлінням.

Розрахунок оптимальних показників для процесу технології являє собою параметричну оптимізацію.

Потрібно мати на увазі, що рішення таких задач є досить обмеженими.

Отже, процес параметричної оптимізації – це з'ясування характеристик x , де функція $F(x)$ має суто екстремальне значення. Така функція називається цільовою.

На всіх етапах явище технологічного процесу – це сукупність задач, що необхідно розв'язати [2].

Під час вирішення задач відносно технології підготування комунікація інженера-технолога з електронно-обчислювальною машиною являє собою певний алгоритм обміну інформаційними потоками.

Режими обміну інформацією включають в себе два підрозділа: діалоговий та пакетний.

Використання діалогового режиму інженером-технологом дозволяє оперативно використовувати електронно-обчислювальні машини.

Оскільки режим отримує вхідні повідомлення досить оперативно, то цей метод краще застосовувати в якості найбільш ефективного [15].

Діалоговий режим є ефективним при вирішенні таких задач: розпізнавання геометрії образу деталі; проектування їх маршруту обробки; вимір розмірного зв'язку геометричних образів.

Приклад вихідний послідовності розбиттів наведений на рисунку 1.1.

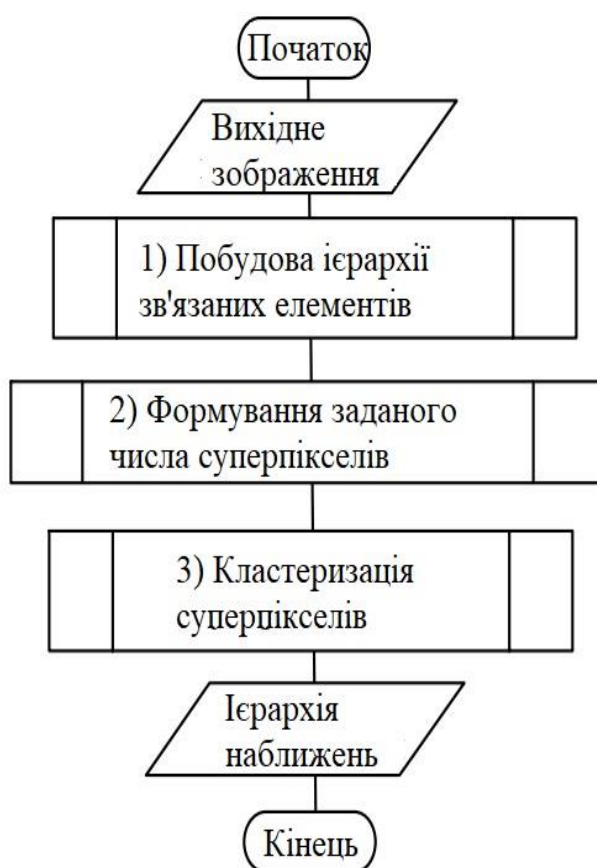


Рисунок 1.1 – Структурна схема алгоритму сегментації цифрових зображень

В основі розглянутого для прикладу алгоритму квазіоптимальної сегментації лежить метод Уорда. Метод полягає в ітеративному поділі пікселів на кластери або, навпаки, їх злиття до одного кластеру. Пікселі, розділені на безлічі, усереднені за значенням яскравості всередині кожного безлічі.

Класичний метод Уорда має ряд істотних недоліків:

– обчислювальна складність методу квадратично зростає з лінійним зростанням числа елементів (пікселів зображення), що перешкоджає безпосередньому застосуванню методу при цифровій обробці зображень;

– метод нестійкий щодо вибору варіанту в разі рівнозначних рішень; на перших кроках укрупнення груп елементів (пікселів, сегментів, кластерів) зустрічаються пари, операції над якими відповідають однаковим значенням цільової показовою функції.

Розглянутий алгоритм квазіоптимальної сегментації забезпечує зниження обрахувальної складності за рахунок поділу процесу побудови ієрархічної послідовності розбиття на три етапи:

а) перший етап – швидка побудова ієрархічної послідовності грубих розбиттів. Швидка побудова здійснюється за рахунок зменшення числа пар пікселів (сегментів, кластерів) на кожній ітерації. Це призводить до неминучого зниження якості. Редукція числа розглянутих пар до тільки суміжних зводить метод Уорда до моделі Мамфорда-Шаха. На цьому етапі будується ієрархія зв'язкових сегментів у спрощеній моделі Мамфорда-Шаха (без урахування довжини кордону сегмента). Для побудови ієрархії застосовується операція Merge злиття двох кластерів.

б) другий етап – проміжне поліпшення якості розбиття при фіксованому числі кластерів пікселів (квітів). Для цього виконуються наступні операції:

- 1) divide – поділ кластера надвоє (до розбиття на попередньому кроці);
- 2) split – виділення частини пікселів в окремий кластер (частина пікселів утворює окремий самостійний кластер);
- 3) correct – рекласифікація частини пікселів кластера (частина пікселів переноситься до іншого існуючого кластеру).

На цьому етапі реалізується спеціально розроблений SI-метод (Segmentation Improvement) поліпшення якості сегментації, який описаний в роботі.

SI-метод полягає в циклічному поділі надвоє сегмента в одному місці зображення і злиття пари суміжних сегментів в іншому місці за критерієм

зменшення значення функціоналу якості для даного розбиття. Результат виконання другого етапу – формування суперпікселів (укрупнених сегментів).

в) третій етап – "добудова" ієрархічної послідовності розбиттів класичним методом Уорда.

У таблиці 1.1 наведені значення класифікаційних ознак, що характеризують розглядуваний алгоритм ієрархічної сегментації, на виході якого формується множина квазіоптимального розбиття вихідного зображення на кластери від 1 до N.

Таблиця 1.1 – Значення класифікаційних ознак, що характеризують розглядуваний алгоритм ієрархічної сегментації

№	Класифікаційний ознака	Значення ознаки
1	Тип операції	Виділення областей зображення на основі властивостей подібності
2	Стратегія обчислень	Послідовне обчислення наближень
3	Наявність критерію	Функціонал якості – сумарна квадратична помилка; критерій якості – мінімум збільшення помилки
4	Тип зображення	Кольорове і півтонове
5	Число розбиття вихідного зображення на виході алгоритму	Безліч розбиття зображення за кількістю розглянутих кластерів, число кластерів змінюється від 1 до N

Слід зауважити, що алгоритми, що генерують безліч розбиття, мають перевагу відносно алгоритмів, що генерують єдине розбиття. З безлічі розбиття обирається одне найбільш підходяще. При цьому вибір здійснює або користувач – «фахівець-предметник», або автоматизована система обробки зображень. Вибір проводиться на етапі постобробки, наступного за етапом сегментації зображення. Відповідне розбиття вибирається в умовах конкретного завдання, при цьому

враховуються апріорна інформація про тематику сцени, особливості зйомки, об'єкти інтересу, їх кількість, особливості та характеристики.

Нижче представимо класифікаційні ознаки у вигляді таблиці 1.2, а також надамо їх стисло характеристику.

Таблиця 1.2 – Класифікаційні ознаки та їхні короткі характеристики

№	Автор	Класифікаційна ознака	Значення ознаки	Примітка
1	Гонзалес - Вудс	1. Спосіб обробки	Знаходження області	Виділення об'єкта за ознаками схожості
2			Знаходження меж	Виділення об'єкта за властивостями відмінності
3	Розенфельд	2. Стратегія виконання	Послідовні обрахування	Послідовне виконання обчислювальних операцій
4			Паралельні обрахування	Паралельне виконання обчислювальних операцій
5	Поршнев	3. Тип зображення	Напівтонове зображення	Алгоритми сегментації зображень з одного складової яскравості
6			Кольорове зображення	Алгоритми сегментації, що враховують всі три колірні складові зображення
7		4. Наявність критерію якості	С критерієм якості	Алгоритми сегментації з використанням встановленого критерію якості
			Без критерію якості	Алгоритми сегментації без використання встановленого критерію якості

Кінець таблиці 1.2

№	Автор	Класифікацій на ознака	Значення ознаки	Примітка
8	Хаников	5. Число розбиттів вихідного зображення на виході алгоритму	Єдине розбиття	Алгоритм генерує на виході одне відсегментоване на однорідні області розбиття вихідного зображення. В ході обчислювального процесу число однорідних за певною характеристиці множин (сегментів, кластерів) фіксоване
			Множина розбиттів	Алгоритм генерує на виході безліч відсегментованих на однорідні області розбиття вихідного зображення. В ході обчислювального процесу число однорідних за певною характеристиці множин (сегментів, кластерів) варіюється в деякому діапазоні

Реалізація поставлених завдань досягається завдяки знанням та творчому мисленню людей під час використання ЕОМ. Однак проектування програм в діалоговому режимі є значно коштовнішим, а значить це впливає на вартість самого процесу розробки.

Таким чином, враховуючи вище викладене, можемо констатувати, що виникає потреба в створенні автоматизованих моделей створення технічного контролю в режимі діалогу, а використовуючи механізми класифікацій, генерування, можливість перейти до програмного режиму з пакетами навчання вищого ступеню.

Проектування автоматизованої системи відеонагляду за допомогою методу розпізнавання образів (руху об'єкта, зображення, облич, символів) мають бути в пакеті прикладних програм, які відносяться до комплексних програм при роботі з головною програмою управління, які мають змогу вирішити схожі технологічні завдання одного класу та виконувати покладену на них функцію відеоспостереження ефективно.

1.3 Постановка задачі дослідження

Незважаючи на високий рівень розвитку сучасної комп'ютерної техніки до теперішнього часу залишається цілий ряд практичних завдань, вирішення яких виявляється досить проблематичним. До числа подібних завдань відноситься задача автоматичного розпізнавання і інтерпретації візуальної інформації. Обумовлено це багато в чому складністю формалізації процесу сприйняття видимих образів. Тому, незважаючи на очевидну легкість, з якою людина вирішує завдання розпізнавання навколишніх його предметів, все ще немає «універсального» математичного або технологічного підходу, що дозволяє конструктивно розробляти методи, алгоритми та автоматичні пристрої, які ефективно здійснюють процес розпізнавання.

Існує широке коло завдань, в яких зображення розглядаються як джерело інформації, на основі якої необхідно винести деяке рішення. Наприклад, такого роду завдання виникають в медичній діагностиці, де зображення того чи іншого людського органа аналізується з метою визначення можливого захворювання. У криміналістиці для встановлення особи людини порівнюють зображення відбитків пальців. За допомогою засобів спектронального дистанційного зондування отримують зображення, за якими з високою вірогідністю знаходять області, що містять поклади корисних копалин. Основою для вирішення такого кола завдань є

теорія розпізнавання образів, яка особливо активно розвивається в зв'язку зі створенням систем штучного інтелекту.

Завдання розпізнавання образів полягає в класифікації зображень на основі певних вимог, причому зображення, що відносяться до одного класу образів, мають відносно високий ступінь близькості.

Прийнятий підхід до розпізнавання образів полягає в класифікації на множині ознак, що обчислюються по спостережуваному зображенню. Можна також сказати, що класифікацію образів полягає в відображенні простору ознак в простір рішень. При такому підході розпізнавання образів включає два завдання: відбір і упорядкування ознак та власне класифікацію.

Критерієм відбору та впорядкування є ступінь важливості ознак для характеристики образів. Завдання класифікації – прийняття рішення про приналежність образу того чи іншого класу на основі аналізу обчислених ознак – має цілий ряд строгих математичних рішень в рамках детерміністичного та імовірнісного підходів. Для того, щоб здійснити класифікацію, необхідно відібрати ознаки [16].

Проведений огляд літературних джерел, присвячених різним аспектам проблеми дослідження методів та алгоритмів визначення границь зображень, показав, що на сьогодні накопичено значний досвід в її вирішенні.

В останні роки, з ростом популярності мобільних сенсорних пристроїв, таких як смартфони, розумні годинник і розумні окуляри, з'вилася необхідність в таких додатках для цих пристроїв, як розпізнавання коду зображення, розпізнавання обличчя, оптичне розпізнавання символів, розпізнавання логотипу, доповнена реальність і змішана реальність. Оскільки мобільні пристрої володіють обмеженими апаратними ресурсами (низько продуктивні процесори, пам'ять невеликого розміру і низька ємність акумулятора), їм потрібні прості і швидкі алгоритми обробки зображень, які споживають невеликий об'єм пам'яті.

При цьому основні основні інсуочі алгоритми суціло фокусуються лише на одному з показників ефективності (швидкість, точність, пам'ять або ресурси

процесора). На час розробки існуючих алгоритмів мобільні пристрої не були такими популярними. Тому потреба в алгоритмах для таких пристроїв ще не була опрацьована.

Як правило, алгоритм трасування контуру може бути оцінений на основі наступних чотирьох критеріїв: точність трасування контуру, час обробки, розмір даних для збереження інформації про контур і можливість точного відновлення і збільшення вихідного контуру з використанням збережених даних.

З огляду на це, метою роботи є розробка нового, удосконаленого або комбінації алгоритмів трасування контурів, заснованих на відстеженні пікселів, які задовольняють вищезгадані проблеми, тобто: забезпечують швидкі і відносно точні результати для трасування контурних пікселів, інформація про контурі може бути стиснута, щоб зменшити обсяг пам'яті і точно відновлюють стислі дані в вихідне контурне зображення.

Для досягнення мети необхідно:

- класифікувати традиційні алгоритми трасування контурів і вивести їх характеристики;
- проаналізувати їх продуктивність на основі точності і швидкості;
- запропонувати і описати алгоритм трасування контуру;
- запропонувати та описати методику стиснення контурних даних і методику відновлення;
- запропонувати алгоритм сегментації зображення як метод оптимізації процесу трасування контуру;
- запропонувати прототип моделі програмного продукту, який реалізує на практиці запропоновану ідею.

2 АНАЛІЗ ІСНУЮЧИХ АЛГОРИТМІВ

2.1 Опис існуючих алгоритмів

Контур визначається як сегмент шириною один піксель і довжиною один або кілька пікселів, а межа визначається як безперервний контур.

Звичайні контурні алгоритми можна розділити на три типи наступним чином: проходження по пікселям, проходження по вершинах і проходження за даними прогону. З них метод відстеження пікселів є найбільш поширеним.

З відомих алгоритмів трасування контурів особливий інтерес привертають наступні:

- алгоритм квадратного відстеження алгоритму, головною перевагою якого є простота. Контурне трасування базується на двох простих правилах: якщо значення активного пікселя дорівнює одиниці (активний піксель знаходиться в точці, що належить об'єкту), то лівий поворот, інакше, коли значення активного пікселя дорівнює нулю (активний піксель знаходиться в точці, яка не належить об'єкту), потім поверніть праворуч). Алгоритм припиняє свою роботу, якщо він повертається до вихідної точки [29];

- модифікований алгоритм квадратного відстеження алгоритму, попередній алгоритм не може відстежити піксель зовнішнього кута, який розташований зліва ззаду, і для відстеження цих пікселів був розроблений модифікований алгоритм. Якщо трасувальник приходиться до лівого заднього зовнішнього кута, ця умова має на увазі, що його лівий задній піксель є чорним, а інші пікселі білими (фон); трасировальник переміститься в лівий задній піксель, а потім його напрямок зміниться в сторону заднього напрямку [11];

- трасування Мура-Сусіда – алгоритм заснований на покроковій верифікації всіх сусідніх точок з метою пошуку наступної точки контуру. Зупинення алгоритму відбувається, коли він повертається до початкової точки [30];

– «радіальне зчитування» – схожий на попередній, але його трасувальник не має інформації про напрямок. Отже, він має дві точки, а саме попередній піксель і поточний піксель для початкового напрямку трасування[13];

– алгоритм Тео Павліді – основна ідея алгоритму полягає у використанні групи з трьох пікселів для визначення наступного піксельного контуру. Перевірка здійснюється шляхом послідовної перевірки сусідніх точок із суворо визначеною послідовністю. Розглядає тільки три суміжних пікселя, наприклад, передній лівий, передній і передній правий. Якщо всі три пікселя білі, трасировщик повертає направо [13];

– алгоритм Змії (активний контур) та алгоритм Амеби – це група алгоритмів, що базуються на знаходженні контурів шляхом послідовності пікселів зображення для пошуку набору крайніх (граничних) пікселів. Алгоритми припиняють свою роботу, якщо всі можливі пікселі є пошуковими або якщо немає пікселів, які б задовольняли певній умові [6];

– топологічно-ієрархічні алгоритми – група алгоритмів, пов'язаних із відстеженням контурів на основі ієрархічних зв'язків між точками. Алгоритми цієї групи замість маркерів використовують морфометричні операції пошуку точок перекриття декількох контурів з метою їх подальшого розділення[6].

Ці алгоритми використовуються для аналізу та опису контуру. Майже в кожній мові програмування є бібліотеки з реалізованими алгоритмами, наприклад, для Matlab, OpenCV.

Основним їх недоліком є залежність від складності ланцюга та критерію зупинки. Алгоритми чутливі до мікроскопічних об'єктів, контур яких містить гілку товщиною в один піксель. Це може призвести як до помилкового закінчення алгоритмів, так і до неправильного вибору контуру. Аналогічна проблема може виникнути, якщо мікрооб'єкт складається з двох або більше частин, які з'єднані лише окремими пікселями. Ще один недолік алгоритмів – недосконалі критерії зупинки (повернення до початкової точки, проходження певної точки кілька разів), що призводять до неправильних результатів роботи.

Деякі алгоритми, такі як алгоритм квадратного відстеження алгоритму і модифікований алгоритм квадратного відстеження алгоритму, виконують непотрібні операції переміщення над білими пікселями. Не всі алгоритми можуть визначати контур розірваних контурних пікселів; отже, вони не можуть бути використані, щоб визначати зв'язність між об'єктами.

Необхідно враховувати розмір даних, які відслідковуються по контуру. Більшість алгоритмів зберігають всі точки пікселів.

2.2 Методи реалізації алгоритма

Побудова опису зображення на основі його уявлення з використанням ознак – найскладніше завдання в процесі побудови будь-якої системи розпізнавання візуальної інформації. При цьому, якщо в рамках деяких математичних моделей вдалося формалізувати процес класифікації, то процес вибору ознак досі залишився процедурою евристичної і залежною як від предметної області, так і від розробника. У той же час певний досвід, накопичений за роки використання засобів розпізнавання образів і обробки зображень для вирішення практичних завдань, дозволяє виділити ряд основних груп ознак, які успішно використовуються для опису і розпізнавання зображень.

До групи геометричних ознак належать ті ознаки, розрахунок яких заснований на використанні геометричних характеристик представлених на зображенні об'єктів. Це можуть бути, наприклад, такі ознаки:

- геометричні розміри зображеного об'єкта по вертикалі або горизонталі;
- відстань між найбільш віддаленими точками на зображеному об'єкті;
- периметр і площа зображеного об'єкта;
- компактність об'єкта (як співвідношення між його площею і периметром);

– числові характеристики описаних або вписаних в зображення об'єкта геометричних фігур, таких як кола, багатокутники, і т.д.

Досить специфічними, але широко використовуються в геометричній групі є ознаки, пов'язані з поданням геометрії контуру об'єкта. Як відомо контур – одна з найбільш важливих характеристик зображеного об'єкта при його сприйнятті людиною. Тому опис контуру – одна з популярних задач, що вирішуються в обробці зображень. До найбільш типових методів опису контуру, використовуваних також і для вирішення завдань розпізнавання, відносяться наступні [8].

Метод ланцюгового кодування для представлення контуру об'єкта був запропонований Фриманом. Він полягає в тому, щоб кордон об'єкта, розташованого на дискретній сітці, представити у вигляді набору елементарних відрізків. Тоді повною характеристикою кордону в кожній точці є напрямок необхідного відрізка. В даному випадку мається на увазі, що точки на кордоні є тільки 4-х зв'язковими (відрізок відкладається лише в 4-х напрямках). Іноді застосовують модифікацію даного методу, що використовує 8-зв'язну модель. Безсумнівним достоїнством представлення кордону зображуваного об'єкта ланцюговим кодом є простота реалізації алгоритму його опису, простота отримання на основі цього опису деяких інших геометричних характеристик об'єкта (наприклад: периметр, площа, лінійні розміри по вертикалі і горизонталі), можливість досягнення інваріантності опису до перетворень подібності – масштабування зображення, його переносу і повороту.

Апарат сплайнів досить широко використовується для практичних застосувань, зокрема, для вирішення завдання опису кордону. Суть методу сплайн апроксимації – представлення кордону об'єкта у вигляді кусочно-поліноміальної функції часто з виконанням ряду обмежень, що накладаються на її гладкість. Найбільш популярними є сплайни третього порядку, оскільки володіють найменшим ступенем, при якій поліноміальна функція може змінювати знак

кривизни. Останнє дозволяє домогтися певної гладкості в точках з'єднання сплайнів за рахунок рівності перших похідних в цих точках [8].

До групи топологічних ознак належать ті ознаки, які характеризують топологічні властивості зображеного об'єкта. Під топологічними властивостями розуміють ті властивості, які залишаються інваріантними щодо топологічних або гомеоморфних відображень. Останні мають на увазі під собою безперервні (прямі і зворотні) відображення. У зв'язку цим топологічні ознаки виявляються апріорно інваріантними до широкого спектру геометричних перетворень: перетворень подібності, афінних перетворень і багатьом іншим. У той же час подібна спільність ознак робить їх досить неінформативними, так як багато з них володіють різною геометрією, об'єкти виявляються невиразними в рамках топологічного опису. Крім того, топологічних ознак відносно небагато, і їх розрахунок, як правило, досить складний. Число зв'язкових компонентів об'єкта – це таке мінімальне число компонент, що складають об'єкт, в кожному з яких будь-які дві точки можуть бути з'єднані лінією, повністю міститься в тому ж компоненті. Число «дірок» в об'єкті – характеризує число зв'язкових компонентів, які не належать об'єкту, але знаходяться всередині нього. Число Ейлера – обчислюється як різниця між числом зв'язкових компонентів об'єкта і числом «дірок» на ньому.

Підсумовуючи все вищесказане, можна стверджувати про те, що топологічні ознаки стосовно до задачі розпізнавання на зображеннях можуть бути використані лише в рідкісних випадках і лише при малому рівні спотворень, пов'язаних зі зникненням або появою частин об'єктів [9].

Ймовірнісними ознаками зображення є числові характеристики випадкового процесу. До найбільш часто використовуваних відносяться: характеристики яскравості, такі як гістограма розподілу значень яскравості на зображенні, текстурні характеристики зображення, до яких відносяться характеристики випадкового процесу, що визначають його кореляційні властивості, такі як коефіцієнт кореляції на зображенні, кореляційну функцію зображення, енергетичні характеристики зображення, до яких відносяться відліки його енергетичного

спектра; ознаки стохастичної геометрії. Дані ознаки характеризують випадкові величини, пов'язані з настанням будь-яких геометричних подій. Найбільш часто респонденти користуються схемою їх генерації, де є випадкове відображення прямої лінії на площину зображення і обчислення деяких характеристик, пов'язаних з подією перетину цією лінією області об'єкта. Наприклад, це може бути кількість перетинів об'єкта лінією, максимальна або мінімальна довжина відрізка лінії, що міститься в об'єкті, сумарна довжина відрізків лінії, що знаходяться в області об'єкта, і т.д. Імовірнісні характеристики таких випадкових величин деяким чином описують форму об'єкта. Причому, в силу випадковості положення лінії, досягається інваріантність до зсувів і поворотів зображення. Більш того, при належній побудові характеристик може бути досягнута інваріантність до масштабу. Безсумнівним достоїнством таких ознак є відносно висока їх стійкість і можливість досить простої автоматизації процесу нарощування числа таких ознак [3].

Отже, в рамках даного розділу було класифіковано традиційні алгоритми трасування контурів і виведені їх характеристики. Було також проаналізовано їх продуктивність на основі точності і швидкості. Було виявлено основні недоліки кожного методу. На основі аналізу існуючих алгоритмів було розроблено та описано запропонований алгоритм, його процедуру трасування (визначення границь) контурів, методу стиснення контурних даних і методу відновлення. Виконана робота закладає основи для подальшого проектування цілісної комплексної системи визначення границь зображень з подальшим їх аналізом в рамках оброблення відеорядів зображень.

3 ОСОБЛИВОСТІ МОДЕЛЮВАННЯ ЗАСОБІВ РОЗПІЗНАВАННЯ ОБРАЗІВ НА ЦИФРОВОМУ ВІДЕОРЯДІ

3.1 Методи оптимізації

Сегментація зображення грає важливу роль в комп'ютерному зорі. Воно спрямоване на виділення значимих об'єктів, що лежать в зображенні.

Як правило, немає унікального методу або підходу до сегментації зображення. Кластеризація є потужною технікою, яка була розвинена в сегментації зображення. Кластерний аналіз полягає в розбитті набору даних зображення на декілька непересічних груп або кластерів. Методи кластеризації, такі як К-метод, покращений К-метод, методи нечіткого середнього значення і поліпшений метод нечіткого середнього значення алгоритм були активно використовували в науці.

Метод К є одним з найпопулярніших методів через свою простоту і обчислювальну ефективність. Кількість ітерацій може бути зменшено при використанні покращеного К методу в порівнянні зі звичайними. Метод нечіткого середнього має додаткову гнучкість для пікселів, що належать кільком класам з різним ступенем членства. Але цей метод є досить повільним і потребує багато часу для розрахунків, яке цей недолік долається покращений за допомогою покращеного методу. Нашим завданням буде запропонувати алгоритми, що дають сегментоване зображення відносно гарної точності, а необхідний комп'ютерний час буде адекватно зменшений.

Якість сегментованого зображення вимірюється статистичними параметрами: індекс RI, глобальна помилка узгодженості, варіації інформації і похибка зміщення кордону.

Сегментація зображення може бути визначена як класифікація всіх елементів зображення або пікселів в зображенні в різні кластери які демонструють подібні риси.

Сегментація включає в себе поділ зображення на групи пікселів, які однорідні відносно до деяких критеріїв [26].

Різні групи не повинні перетинатися одна з одною і суміжні групи повинні бути неоднорідними. Ці групи називаються сегментами. Сегментація зображення розглядається як важлива базова операція для змістовного аналізу і інтерпретації отриманого зображення. Це – важливий компонент аналізу зображень та / або розпізнавання образів системи, і є одним з найскладніших завдань в обробці зображень, яке визначає якість фінального результату. Дослідники багато працювали над цією фундаментальною проблемою і запропонували різні методи для сегментації зображення. Ці методи можуть бути широко класифіковані в сім груп: граничне значення гистограми, кластеризація, зростання регіону, поділ і злиття регіонів, крайова і фізична модель, метод нечітких підходів та метод на основі нейронної мережі [26].

Зображення може бути визначене як двомірна функція $f(x, y)$, де x і y – просторові (плоскі) координати, амплітуда f влюбій парі координат (x, y) називається інтенсивністю сірого зображення в цій точці. Область образів в зображенні може бути неточною з-за впливу неточного інструмента, проблеми з процесором збору даних і взаємодія з природними явищами. Такі дані можуть не підходити для аналізу. Таким чином, система, сегментація зображень часто необхідна і має бути прийнята в якості значного інструменту для скорочення часу і збільшення якості обробки і аналізу.

Проблема складається в виборі підходящих моделей для сегментації, так як сегментація має відбуватися в короткий час. Проблема сегментації має бути вирішена в короткий проміжок часу.

Кластеризацію можна вважати найважливішою структурною невирішеною проблемою, тож, як і будь-яка інша проблема такого роду, вона має справу з структурою невизначених даних. А визначення кластеризації може бути наступним: процес організації об'єктів в групі, члени яких певним чином схожі.

Таким чином, кластер – це сукупність об'єктів, які є "подібними" між собою і є "несхожими" на об'єкти, що належать іншим кластерам.

Коротко розглянемо кожен з методів:

а) К-метод – один з найпростіших алгоритмів який вирішує добре відому проблему кластеризації. Процедура дотримується простого і легкого способу класифікації даних встановлених на певну кількість кластерів (припустимо k кластерів) [27]. Цей алгоритм оснований на квадратичній функції. Середньоквартичним значенням в цій функції є відстань темних пікселів від репрезентативного центру кластеру.

Алгоритм складається з наступних етапів:

- 1) постановка точки K на площині, яка представлена об'єктом, що кластеризується. Ця точка представляє собою центроїд початкової групи;
- 2) кожен об'єкт призначається до групи, яка має найближчий центроїд;
- 3) коли всі об'єкти були віднесені, перераховується положення K центроїдів. Потім повторюємо кроки 2 і 3 поки центроїди більше не рухаються. Це дозволяє поділити об'єкти на групи, а метрику слід мінімізувати.

б) покращений К-метод. Нехай $D = \{d^{(j)} / j = 1, \dots, n\}$ є набором даних, які мають кластери, $C = \{c_i / i = 1, \dots, k\}$ та набори центрів $S_j = \{d^{(j)} \mid d^{(j)} \text{ що є членами в } j\text{-му кластеру}\}$ [27]. Тоді наступна функція визначається як цільова функція:

$$Cost(D, C) = \sum_{j=1}^n dist(d^i, c_k) \quad (3.1)$$

Алгоритм має такі етапи:

- 1) розподіл D на K частин згідно шаблонами даних;

$$D = \bigcup_{k=1}^K S_k \cap S_{k_1} \cap S_{k_2} = \theta, k_1 k_2 \quad (3.2)$$

2) зробити $X_{(k)}^{(0)}, k = 1, \dots, K$ початковими центрами кластеризації, розрахованими по

$$x_{(k)}^{(0)} = \frac{\sum_{d^{(j)} \in S_k} d^{(j)}}{|S_k|}, k = 1, \dots, K \quad (3.3)$$

3) визначають приналежність шаблонів в кожному з K -кластерів відповідно до мінімальною відстанню від критеріїв центру кластера;

4) розраховують нові центри по наступній ітераційній формулі:

$$x_i^{(k+1)} = \sum_{d^{(j)} \in S_k} \frac{d_i^{(j)}}{q_j^k} / \sum_{j=1}^n \frac{1}{q_j^k} \quad (3.4)$$

де $q_j^k = \|x^{(k)} - d^{(j)}\|$.

5) повторюють кроки в і г до тих пір, поки не відбудуться зміни в кластерних центрах;

в) кластеризація нечітких C -середніх [27]. Нечітке середнє – це метод кластеризації, який дозволяє одному фрагменту даних належати двом чи більше кластерів. Тобто дозволяє пікселям належати декільком класам з різним ступенем приналежності. Він засновується на основі наступної цільової функції:

$$J(U, c_1, c_2, \dots, c_c) = \sum_{i=1}^c J_i = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m d_{ij} \quad (3.5)$$

де c_j – центр d -розмірності кластера;

u_{ij} – ступінь участі x_i в кластері j ;

m – будь-яке дійсне число більше 1;

d_{ij} – d -мірні дані.

Метод має обмежене застосування через істотний недолік – неможливість коректного розбиття на кластери, в разі коли кластери мають різну дисперсію по

різним розмірностям елементів (наприклад, кластер має форму еліпса). Даний недолік був виправлений в запропонованому методі.

Алгоритм складається з наступних кроків:

- 1) ініціалізувати $U = [u_{ij}]matrix, U(0)$;
- 2) на кроці k : вирахувати вектори центрів з $c(k) = [c_j] c U(k)$
 $U^{(k)}, U^{k+1}$

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m x_i}{\sum_{i=1}^N u_{ij}^m} \quad (3.6)$$

- 3) оновлення $U^{(k)}, U^{k+1}$

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (3.7)$$

- 4) якщо $\|U^{(k+1)} - U^{(k)}\| < \varepsilon$, то припиняємо ітерацію; в іншому випадку повераємося до кроку 2.

3.2 Адаптація методів сегментації до умов задачі

Для умов задачі було запропоновано вдосконалений алгоритм кластеризація нечітких С-середніх заснований на концепції стиснення даних, коли розмірність вхідних даних значно знижена. Стиснення даних включає в себе два етапи: квантування та його узагальнення.

Квантування простору ознак виконується шляхом маскування молодших бітів « m » значення властивості. Квантований вихідний сигнал призведе до загальних значень інтенсивності для декількох векторів ознак. В процесі агрегування вектори ознак, які мають загальні значення інтенсивності, групуються

разом. Репрезентативне вектор ознак вибирається з кожної групи, і вони наведені в якості вхідних даних для звичайного алгоритму кластеризації нечітких С-середніх. Як тільки кластеризація завершена, значення репрезентаційності в репрезентативній векторній ознаці розподіляються однаково серед всіх членів рівня квантування. Оскільки в модифікованому алгоритмі використовується скорочений набір даних, швидкість збіжності значно поліпшується в порівнянні зі звичайним методом.

У поліпшеному алгоритмі використовуються ті ж етапи, що і в звичайному, за винятком зміни критеріїв поновлення кластера і поновлення значення членства. Модифіковані критерії показані нижче.

$$c_i = \frac{\sum_{j=1}^n u_{ij}^m y_j}{\sum_{j=1}^n u_{ij}^m}, u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}}{d_{kj}}\right)^{2/(m-1)}} \quad (3.8)$$

де y – скорочений набір даних;

$$d_{ij} = y_j - c_i.$$

3.3 Алгоритм трасування контурів зображення

Для вирішення існуючих проблем з відомими алгоритмами було запропоновано модифікований алгоритм трасування контурів адаптований для роботи на пристроях з обмеженими ресурсами.

Спочатку трасувальник досліджує пікселі контуру на основі інтенсивності лівого заднього і лівого пікселів. Після цього трасувальник рухається по пікселям відповідно до інтенсивності переднього і переднього лівого пікселів. Алгоритм складається з восьми ситуацій в яких може опинитись трасувальник.

Для розкриття роботи алгоритму розглянемо всі вісім ситуацій проходження трасувальника по пікселям:

- в першій ситуації задній піксель є білим, а лівий і задній лівий пікселі є чорними. В цій ситуації трасувальник пройде по лівому чорному пікселю і спрямує в нижній лівий, якщо наступний за лівим пікселем є фоном;
- в другій ситуації лише лівий задній піксель є чорним. В цій ситуації трасувальник піде в чорний піксель через лівий білий;
- в третій ситуації лише лівий піксель є чорним. Тепер трасувальник піде в лівий чорний піксель через задній лівий;
- в четвертій ситуації всі пікселі навколо є фоном. Трасувальник в цій ситуації піде вперед;
- в п'ятій ситуації лише передній лівий піксель є чорним. В цій ситуації трасувальник поверне вліво і буде рухатися вперед до чорного пікселя;
- в шостій ситуації передній і передній лівий пікселі є чорними. Тоді трасувальник спрямується вперед, потім наліво і вперед;
- в сьомій ситуації лише передній піксель є чорним, тоді трасувальник піде на чорний піксель і право;
- в останній ситуації всі пікселі є білими. Тоді трасувальник розвернеться.

Всі випадки легко класифікуються з використанням алгоритму. Для автоматизації проходження всю систему можна розцінювати як набір станів. Трасувальник представляє собою вектор. Вектор представляє інтенсивність чотирьох суміжних пікселів трасера. Оновлення стану представляє собою переміщення. Запропонований алгоритм розроблено на основі двох етапів з двома основними цілями.

По-перше, операція перевірки на остановку відбувається тільки в початковому стані; отже, кількість операцій перевірки. Це більш ефективно в порівнянні з перевіркою це відбувається для кожного пікселя контуру. Трасировщик уникає внутрішнього і зовнішнього кутових пікселі в якості початкового і кінцевого пікселів. Більш того, в разі якщо трасировщик не має оновлення, немає необхідності виконувати операцію перевірки.

По-друге, запропонований алгоритм усуває деякі з надлишкових операцій, які використовуються для виявлення білих пікселів. Традиційні алгоритми не враховують білі пікселі в попередньому шляху, тому вони іноді повторно виявляють білі пікселі з попередніх етапів трасування під час поточної трасування.

Стиснення і відновлення даних відбувається за рахунок збереження постпальних наборів станів вектору. Тобто, якщо трасувальник рузався сто пікселів в одному напрямку, то нам потрібно зберегти напрямок і відстань (кількість пікселів), що представляє собою динамічний стан вектору. Зміна напрямку представляє собою новий динамічний стан вектору.

Відтворення послідовності динамічних станів буде трансформовано в структуру даних, де одиницям даних будуть стики чотирьох пікселів. Кожна комбінація стиків буде представлена константою.

Алгоритм відновлення буде складатися з проходження по послідовності станів вектору трасування та послідовним відтворенням контуру шляхом запису послідовності суміжних стиків пікселів.

3.4 Реалізація моделі

Основною вимогою до використовуваної системи ознак аналізованого зображення є вимога ефективності процесу проходження контуру. Ця вимога часто має ряд суперечливих аспектів.

По-перше, вказане пред'являє до ознак вимоги обчислювального характеру. Вони полягають в тому, щоб існував алгоритм розрахунку ознак, і цей алгоритм був обчислювально ефективний. Необхідність першої вимоги обумовлена тим, що розробка будь-якою системою розпізнавання відбувається як правило в рамках деякого ліміту «цінового» ресурсу. Це обмежує можливості використання ряду засобів формування ознак. Наприклад, подібне обмеження в області розпізнавання

зображень може відбитися на можливості використання відеоапаратури, яка реєструє кольорові зображення, або забезпечує підвищену роздільну здатність. Друга вимога випливає з необхідності задоволення певним часовим обмеженням, що накладається на процес розпізнавання в цілому. Ця вимога досить типова для систем розпізнавання в реальному часі, наприклад, для бортових систем дистанційного зондування, систем оперативного контролю та інших.

По-друге, вимога ефективності розпізнавання накладає певні вимоги на значення ознак. А саме, для об'єктів різних класів значення ознак повинні відрізнятися сильніше, ніж для об'єктів одного класу. Ця вимога іноді інтерпретують як вимога компактності опису класу в просторі ознак, коли образи об'єктів одного класу в просторі ознак утворюють компактні області – кластери або таксони. Це не зовсім коректно, тому що на вимогу «відмінності» або «близькості» значень ознак слід дивитися з точки зору використовуваного при розпізнаванні класифікатора – вирішального правила. У той же час компактність образів, в разі її досяжності, дозволяє істотно спростити процедуру класифікації і зробити її більш стійкою.

По-третє, вимога ефективності розпізнавання призводить до необхідності задоволення вимоги стійкості або інваріантності образу (опису) до ряду можливих спотворень об'єкта. Дійсно, об'єкти, що класифікуються на зображенні, в реальному житті піддаються цілому ряду змін. Крім того, процес реєстрації сцени привносить додаткові спотворення, пов'язані з неідеальністю приладів реєстрації, зміною освітленості, шумами і т.д. На практиці це призводить до зміни зображення об'єкта і, отже, зміни його образу в просторі ознак. Останнє в загальному випадку може вплинути на результати класифікації і, отже, ефективність системи в цілому.

Слід зазначити, що, якщо характер вимог ефективності ознак, існування методів їх розрахунку і наявності швидких алгоритмів їх обчислення часто залежить від специфіки розв'язуваної задачі, то вимоги інваріантності для цілого ряду практичних завдань розпізнавання на зображеннях виявляються досить загальними. Зокрема:

- інваріантність до шумових і динамічних спотворень;
- інваріантність до яркісних спотворень (зміни яскравості і контрасту);
- інваріантність до зміни місця розташування об'єкта;
- інваріантність до зміни масштабу об'єкта;
- інваріантність до зміни орієнтації об'єкта (до повороту об'єкта в площині зображення);
- інваріантність до довільних афінних перетворень;
- інваріантність до зміни ракурсу зйомки об'єкта (для тривимірних об'єктів).

Інваріантність до вказаних спотворень в загальному випадку досягається за рахунок попереднього перетворення вихідного зображення. Зокрема, перша група спотворень, пов'язана з шумовими і динамічними змінами зображення, усувається за рахунок використання методів і алгоритмів відновлення. Яскравості спотворення, що відносяться до другої групи, ефективно можуть бути усунені за рахунок приведення зображення до «нормалізовано» виду.

При комп'ютерній обробці вхідні дані графічного зображення зберігаються в цифровому вигляді згідно колірній моделі RGB. Модель базується на поєднанні трьох кольорів – червоного, зеленого і синього. У зображенні точкам належить відповідна пропорція змішування цих кольорів. Для кожного кольору виділяється фіксована пам'ять обсягом 8 біт.

Будь-який колірний простір RGB має бути пов'язаний з еталонним колірним простором CIE XYZ. Для CIE XYZ відома залежність між значенням коефіцієнтів (x, y, z) пікселя і фактичним значенням яскравості еталонних джерел світла, змішенням яких домагаються отримання потрібного сприйманого кольору. Для кожного RGB-зображення перетворення в XYZ має бути явно або неявно визначене, інакше коректне відображення можливе тільки на тому моніторі, на якому створювалося зображення. Дана інформація називається колірним профілем зображення. Якщо монітор має такий профіль, то шляхом перетворення $RGB_{image} \rightarrow XYZ_{image}$ з наступним перетворенням $XYZ_{image} \rightarrow RGB_{display}$

можна отримати на іншому моніторі кольори, що відповідають початковим (з урахуванням обмежень перенесення кольорів).

У просторі XYZ координата Y за визначенням відповідає сприйманій яскравості кольору. Щоб отримати з повнокольорового зображення монохромне, необхідно перетворити кожен піксель в XYZ і взяти компоненту Y як результат. Відомо, що перетворення між будь-якими адитивними колірними системами лінійне (в силу лінійності сприйняття кольору людиною), отже, може бути описане матрицею $M_{3 \times 3}$, такою, що:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = M \begin{pmatrix} R \\ G \\ B \end{pmatrix}. \quad (3.9)$$

Переважає більшість цифрових зображень відповідають стандарту sRGB. Так само необхідно виконувати гамма-перетворення [16]. Таким чином, перетворення з кольорового sRGB- зображення в монохромне здійснюється згідно співвідношень:

$$Y = 255(0.21r + 0.72g + 0.07b)^{1/2.2}, \quad (3.10)$$

$$r = (R/255)^{2.2}, g = (G/255)^{2.2}, b = (B/255)^{2.2} \quad (3.11)$$

де коефіцієнти (0.21, 0.72, 0.07) – округлений до другого знаку рядок із матриці перетворення M sRGB з системи sRGB в XYZ.

Передбачається, що на зображенні з об'єктами, які підлягають розпізнаванню, колір цих об'єктів суттєво відрізняється на фоні іншого кольору: наприклад, на сторінках книги фон – білий, цифра – чорна; на маркуванні вантажного вагону фон – темний, цифра (маркування) – біла. Тому доцільно працювати з монохромним зображенням (кольори зображення знаходяться в діапазоні $[0, 255]$), тобто з градацією сірого «0» – чорний, «255» – білий.

Запропоновано використати «критичне» значення кольору (градації сірого) для об'єкта $Y_{зд}=[0, 255]$, по якому відбуватиметься відсічення фону і виділення тільки тієї частини зображення, де знаходиться об'єкт. Значення $Y_{зд}$ задається залежно від передбачуваного кольору об'єктів і фону. Якщо фон темніший за об'єкти, то відсічення частини зображення з фоном відбуватиметься при $Y < Y_{зд}$, за умови світлішого фону – за цифри $Y > Y_{зд}$, де Y – значення поточного пікселя.

З урахуванням вищезазначеного, розроблено алгоритм виділення об'єкту на монохромному зображенні, який складається з наступних етапів:

- завдання значення $Y_{зд}$ в діапазоні $[0, 255]$, що відповідає градації сірого;
- попіксельне зчитування зображення (див. запропонований алгоритм в розділі 2.2);
- фіксація координати «крайніх» пікселів, значення яких $Y > Y_{зд}$ (фон темніший за об'єкти);
- обрізка зображення по отриманим «крайнім» точкам – вилучення області зображення з об'єктом.

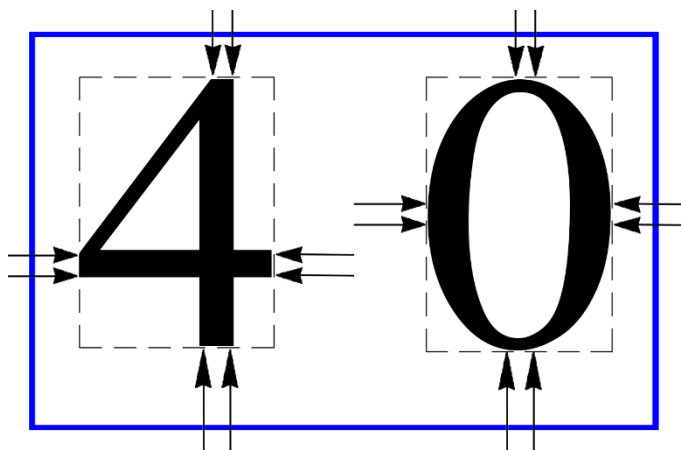


Рисунок 3.1 – Виділення об'єктів на зображенні

На даному прикладі представлена можливість отримання зображення для ідентифікації чисел із зовнішнього пристрою – веб-камери. Проблема ідентифікації такого зображення полягає в тому, що неможливо передбачити якість освітлення, при якому був зроблений знімок, відблиски на знімку та інші дефекти. В процесі

конвертації зображення в монохромне всі ці недоліки можуть призвести до того, що зображення буде занадто темним або світлим. Тобто задане за умовчанням значення $Y_{зд}$, по якому виділяється область з об'єктом, буде неактуальне і програма не зможе відокремити, отже, і розпізнати, об'єкт.

Для вирішення цієї проблеми запропоновано використання медіанної фільтрації зображення [5]. При такому підході відбувається послідовна обробка кожної точки зображення, внаслідок чого утворюється послідовність оцінок. Дана фільтрація дозволяє виключити викиди, тобто значення пікселів, які суттєво відрізняються від інших. При цьому ці викиди замінюються сусідніми значеннями тієї ж послідовності. Кількість значень початкових пікселів, що залучаються до розгляду, називається апертурою фільтру. При апертурі $2m+1$ можна виключити m викидів, що йдуть поспіль. При медіанній фільтрації використовується двомірне вікно (апертура фільтру – рис. 8), що зазвичай має центральну симетрію, при цьому його центр розташовується в поточній точці фільтрації. Точки зображення, які виявилися в межах вікна, утворюють робочу вибірку поточного кроку [16].

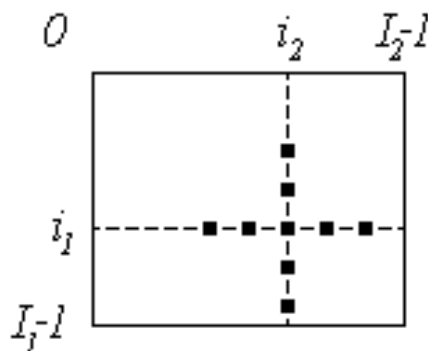


Рисунок 3.2 – Вікно при медіанній фільтрації

Робоча вибірка позначається у вигляді одномірного масиву $Y = \{y_1, y_2, \dots, y_n\}$; число елементів якого дорівнює розміру вікна, а їх розташування є довільним. Якщо упорядкувати послідовність $\{y_i, i = \overline{1, n}\}$ за збільшенням, то її медіаною буде той елемент вибірки, який займає центральне положення в цій

впорядкованій послідовності. Отримане таким чином число є продуктом фільтрації для поточної точки зображення. Медіанна фільтрація позначається як:

$$x^* = med(y_1, y_2, \dots, y_n) \quad (3.12)$$

Для того, щоб зробити алгоритм розпізнавання універсальним – здатним працювати з різними об'єктами, виникає потреба враховувати той факт, що кожне зображення об'єкту, який ідентифікується, може мати свою, відмінну від використовуваної в навчальній вибірці, роздільність. Для вирішення цієї проблеми було запропоновано розділяти зображення з об'єктом на прямокутній області (кількість областей задається програмно) та обчислювати концентрацію пікселів певного кольору в кожній з областей, в результаті чого отримано вхідний вектор параметрів [16].

На цьому етапі також будутьт аналізуватися окремі частини на предмет кількості білих та чорних пікселів. При цьому, якщо буде виявлено сегмент з білими пікселями без чорних, то такий сегмент можна виключити з подальшого аналізу, тобто на цьому сегменті можна не виконувати алгоритми фільтрації, трасування та інші, а якщо сегмент буде складатися з виключно чорних пікселів, то такий сегмент можна не виконувати в алгоритмах трасування та фільтрації, але цей сегмент має бути включений до роботи алгоритмів сегментації, тобто тієї її частини, яка відноється до виділення контурів окремих образів.

Значення (кількість пікселів) ширини w і висоти h оброблюваного зображення може не відповідати заданій кількості частин n і m , на які розподіляється зображення з об'єктом, отже при обчисленні ширини і висоти для областей буде отримано значення дійсного типу, що неприпустимо при роботі з пікселями. У такому разі при розділенні зображення на n по вертикалі і m по горизонталі для «крайніх» областей ширина і висота відрізнятимуться від інших.

Розбиття зображення на $n \times m$ областей відбувається за наступним алгоритмом:

- обчислюється ширина w і висота h обробленого зображення з об'єктом;
- обчислюється ширина wt і висота ht прямокутної області на які ділиться зображення відповідно формулам:

$$\begin{cases} wt = \frac{w}{n}, \\ ht = \frac{h}{m}; \end{cases} \quad (3.13)$$

Якщо ширина w і висота h не кратні кількості частин n і m , то wt і ht округлюють до меншого – це ширина і висота для всіх областей, окрім «крайніх».

- обчислюється остаточно ширина і висота для «крайніх» областей за формулами:

$$\begin{cases} wc = w - wct * n, \\ hc = h - hct * m; \end{cases} \quad (3.14)$$

де wct – закруглене до меншого значення wt ;

hct – закруглене до меншого значення ht .

- обчислюється площа кожної області, яка використовується для подальшого знаходження відсотка пікселів певного кольору градації сірого в області, згідно рівнянь:

$$\begin{cases} st = wt * ht, \\ sc = wc * hc, \\ scw = wc * ht, \\ sch = wt * hc; \end{cases} \quad (3.15)$$

де sc , scw , sch – площа «крайніх» областей;

st – площа інших областей.

Зазвичай розрізняють процедури навчання двох типів: з попереднім навчанням і з суміщенням процесів навчання і розпізнавання. У разі попереднього

навчання до початку розпізнавання для настройки класифікатора висувається ряд об'єктів відомих класів. На основі цієї інформації визначаються параметри класифікатора, і в подальшому саме такий класифікатор застосовується для розпізнавання всіх інших образів.

При розпізнаванні образів з використанням процедури поєднаного навчання і розпізнавання інформація, що доставляється первісної групою образів враховується при побудові початкового правила класифікації. Після цього береться наступна група образів, до якої застосовується наявне правило класифікації. Оцінюється результат класифікації і, при необхідності, правило коригується з урахуванням нової інформації. Первісне правило може бути досить довільним – в результаті подібного послідовного навчання досягається якісна класифікація всіх можливих образів.

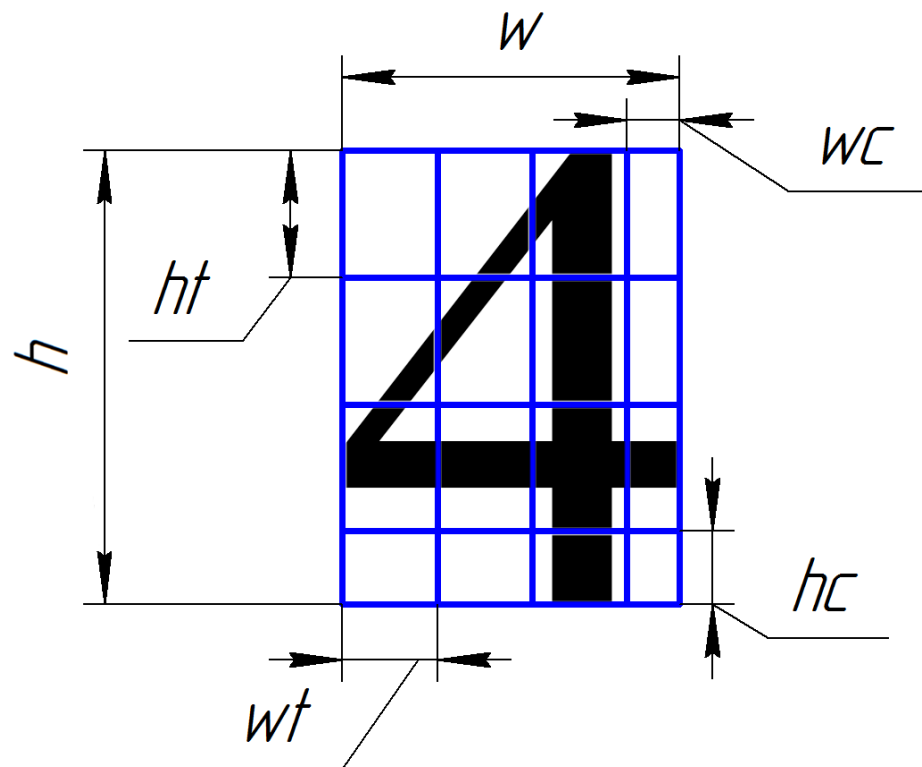


Рисунок 3.3 – Розподілення зображення на області

Розроблений алгоритм та моделі дозволяють робити процес розпізнавання зображень досить ефективним, щоб проводити його на девайсах з малими ресурсами (смарт-камери, смартфони та інші). Даний алгоритм може бути з легкістю модифікований або розширений, що є досить важливим в системах, де використовуються самообучаємі системи.

3.5 Модель прототипа системи

Процес розпізнавання образу включає в себе порядок кластерного методу аналізування, який складається з таких взаємопов'язаних елементів [2]:

- початку;
- зчитування вхідних даних;
- створення матриці координат;
- обчислення значень міри схожості;
- відокремлення ліній на зображенні;
- кінця.

Застосування кластерного аналізу означає такі дії:

- можливість обрати об'єкт для кластеризації;
- здатність аналізу критеріїв, завдяки яким аналізуватимуться об'єкти;
- спроможність розрахувати значення відстані між об'єктами;
- проектування подібних кластерів об'єктів;
- результати аналізу.

Програми розпізнавання об'єкту чи явища, що моделюється людиною завжди мають спеціалізовану направленість на відміну від її власних фізичних можливостей. Відносно загальних підходів до створення будь-якої програми, то тепер у разі, якщо є деяка система об'єктів чи явищ, які потребують процесу розпізнавання або класифікації, на базі узагальнення алгоритмів при моделюванні

систем розпізнавання етапність вирішення конкретних задач виглядає наступним чином [10]:

- відносно обраного принципу набір об'єктів чи явищ поділяється на класи, іншими словами, йому надається абетка класів;
- проектується алгоритм властивостей (словник);
- за словником властивостей характеризується кожен клас;
- моделюються механізми визначення властивостей;
- на механізмах обчислень впроваджується процес зіставлення апостеріорної та апріорної складових та визначаються результати загального процесу розпізнавання.

Проте хоча послідовність дій визначена, залишаються проблеми:

- поділ об'єктів на різноманітні класи;
- способи накопичення та обробки інформації;
- причини вибору ознаки;
- характеристика класів на мові ознак;
- вибір методів для порівняння апріорної та апостеріорної інформації;
- поява системи розпізнавання.

В процесі розпізнання образів груповий аналіз застосовується так, що під час розпізнання креслення деталі переходить до піксельного зображення, а це, в свою чергу, створює матрицю координат. Далі обирається міра відстані пік селів та формується лінія деталі [4].

Порядок дій кластерного аналізу:

- зчитування і обробка вхідних даних. Такими даними можуть бути креслення деталі, а також її зображення;
- проектування матриці координат та виокремлення головних значень. На цьому етапі присвоюється кожному пікселю зображення номер кольорової гами, – потім він відшукується серед інших точок;
- обчислення ступеню подібності. Виокремлення пікселю в групі з найближчою відстанню, це в свою чергу робить зображення ліній зображення;

- кінцеве формування ліній креслення деталі та редакція для більшої якості, засобами видалення шумів;
- фініш роботи алгоритму кластеризації і представлення даних для подальшого виконання програми.

Отже, розроблено математичне та алгоритмічне забезпечення задачі опису запропонованого алгоритму, його процедуру трасування контурів, методику стиснення контурних даних і методику відновлення (див. також розділ 2.2). Запропоновано прототип моделі програмного продукту, який реалізує на практиці запроповану ідею.

Основною метою розробки цієї системи було створення ефективного системного підходу до визначення границь зображень, який оснований на алгоритмах, які не потребують значних ресурсів процесора чи значних об'ємів оперативної пам'яті.

Запропоновані алгоритми не дають можливість визначити границі зображень з дуже високою точністю, але така мета не ставилася, тому що отримані контури будуть використовуватися для розпізнавання типів образів самообучаємою системою. А для цієї мети загального визначення типу образу не потрібна висока точність. А враховуючи той факт, що сучасні камери мають високу роздільну спроможність, це призведе до деякого нівелювання не високої точності алгоритмів, а отже можна стверджувати, що основна мета була досягнута.

Для цієї задачі виконана побудова структурної моделі роботи системи визначення границь (контурів) зображень. Це дозволило виділити основні етапи роботи моделі, їх призначення і взаємозв'язки між ними.

На основі моделі запланована розробка програмного забезпечення, для якої було закладено основні компоненти алгоритму.

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОЕКТУ РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ ДЛЯ СИСТЕМИ ВІДЕОНАГЛЯДУ

4.1 Структурна схема комплексного алгоритму обробки і аналізу зображень

Як правило, системи розпізнавання образів повинні бути деталізовані заздалегідь, щоб забезпечити їх успіх в роботі. Це створює обмеження, які обмежують адаптивність системи і викликають два основних недоліки. По-перше, необхідні додаткові години для програмування. По-друге, система, таким чином, обмежена певними визначеними операціями, що призводить до жорсткої автоматизації.

У динамічному виробничому середовищі гнучка і адаптивна система є затребувана. Інтегруючи обробку зображень з системою автоматизованого аналізу збору та аналізу зображень (відеорядів), система стає більш адаптивною і в той же час відкриває двері для більш широкого кола застосувань. Досить суттєвий вклад в розвиток цього напрямку було зроблено в роботах [26] і [16], які продемонстрували можливість використання роботизованої системи для створення портрета людини на основі захоплення зображення обличчя.

Способи подання вхідної інформації в системі можна розділити на дві групи: одна будує моделі образу, а визначає та аналізує образи на основі захоплених зображень з камери, підключеної системі.

Основна увага приділяється швидкій обробці зображень, побудові контурів і віддаленому моніторингу, аналізу образів та прийняттю рішень.

В роботі, зокрема, представлено веб-систему, яка дозволяє віддаленим операторам контролювати процес автоматизованого аналізу образів в рамках системи відеонагляду. Система націлена на аналіз поведінки людини в повсякденному житті та виявленню заданих ситуацій (злочинів), в той же час розробляючи систему необхідно закласти функціональні можливості системи для майбутнього розширення.

Розроблена система має надавати можливість віддаленим операторам контролювати етапи обробки зображень. Авторизований оператор може навіть отримати віддалений доступ до системи з будь-якого комп'ютера, що працює на будь-якій платформі, через веб-портал системи. У той же час система забезпечує певний рівень контролю для операторів, зумовлений в системі. Це досягається за допомогою C#-API, через який оператор може отримати доступ до системи з аутентифікацією. Крім того, віддалений оператор має можливість виконувати різні онлайн-операції з системою, наприклад повтор запуску, виділення моменту в відоряді, управління камерою.

Система складається з мережевої камери і шина, обидва підключені до сервера додатків через Ethernet. Буде розроблено спеціальний призначений для користувача інтерфейс для управління камерою, етапами обробки зображень і операціями розпізнавання образів. Як показано на рис. 4.1, архітектура системи складається з чотирьох модулів. Центральним є сервер додатків, який встановлює зв'язок з іншими модулями. У той же час, він пропонує необхідні рішення для обробки зображень.

Другий модуль являє собою промислового сервера, який відповідає за аналіз образів та прийняття рішень, і в той же час він отримує команди від іншого сервера і відповідно виконує завдання.

Третій модуль – мережева камера на основі IP, якою можна керувати віддалено через URL; вона використовується в системі для створення знімків людської або інших об'єктів, а потім відправляє зображення на сервер для обробки.

Віддалений оператор, підключений до системи через веб-браузер, представлений четвертим модулем, де оператор має доступ для контролю і управління процесом.

Запропонована система залишається досить гнучкою для її розвертання на обладнанні різної конфігурації та типу, а також для розширення системи, її зміни та підтримки. При цьому структура говорить про те, що для розробки може бути використана мікросервісна архітектура.

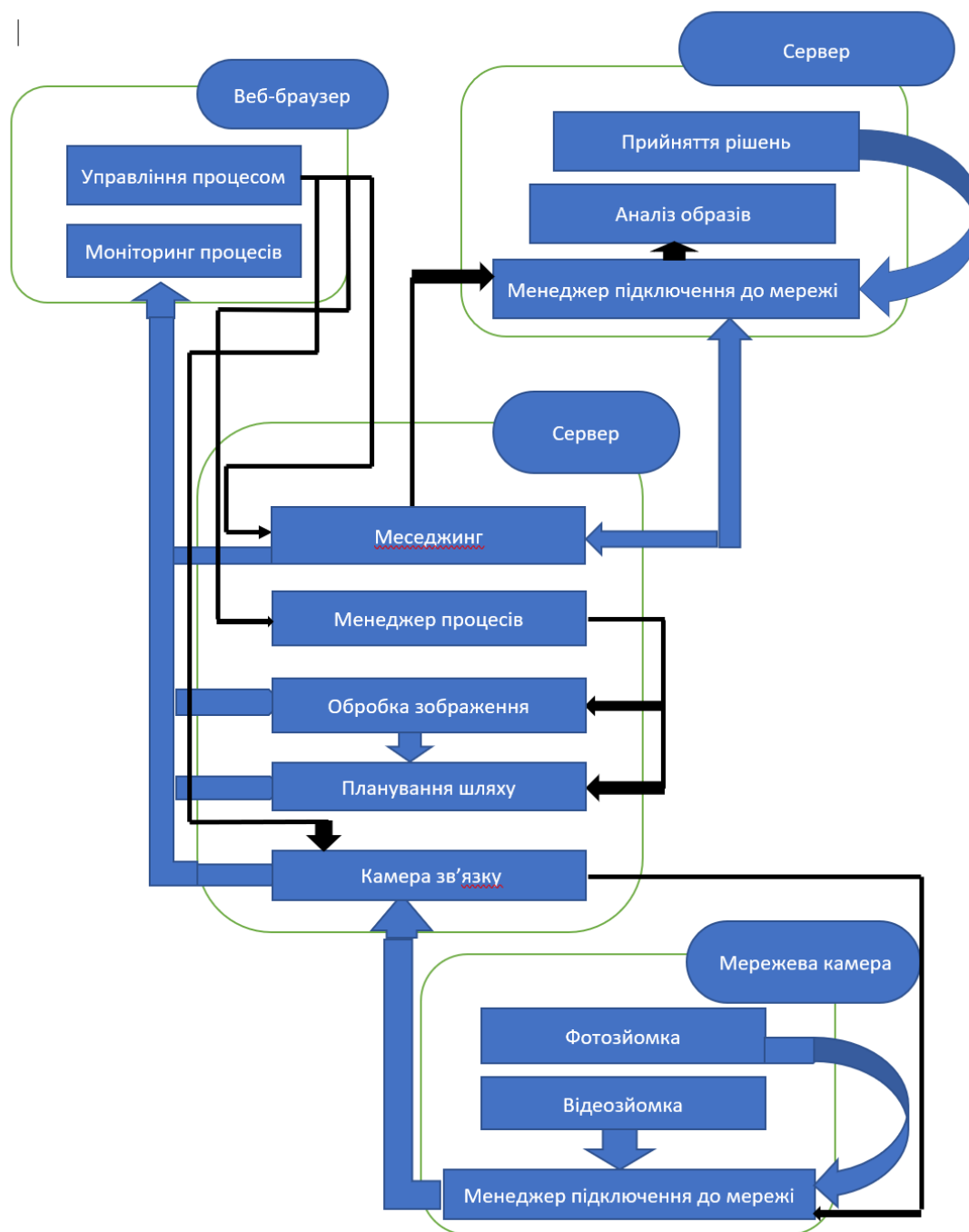


Рисунок 4.1 – Структурна схема системи

Таким чином, система призначена для захоплення зображення об'єктів за допомогою мережевої камери, передачі зображення на сервер, отримання його контурів після обробки зображення сервером і відправки контурів іншому серверу,

пункт за пунктом і контур за контуром. Загальний структурний алгоритм інформаційних потоків наведено в додатку А.

4.2 Розробка архітектури та впровадження створеної математичної моделі в автоматичний процес відеоспостереження

Процедури обробки зображення системою предсавлені в такий спосіб:

– перетворення з кольорового в відтінки сірого. Зображення буде спочатку перетвориться з кольорового в формат відтінків сірого. Цей крок допомагає спростити і прискорити наступні процедури обробки зображень;

– регулювання яскравості і контрастності. Яскравість і контрастність зображення коригуються належним чином для поліпшення результатів майбутніх етапів обробки зображення;

– фільтрація. Однією з основних проблем, що впливають на процес трасування контурів з якість зображення, а саме наявність гауссовського шуму. Отже, фільтр Гаусса з нульовим середнім має бути використаний в процесі видалення шуму;

– початковий поріг. Цей етап – це процес сегментування зображення на два або більше сегмента в залежності від значень інтенсивності його пікселів. Використовується для видалення небажаного шуму, щоб підготувати зображення для наступних кроків;

– виявлення краю. Операція виявлення країв застосовується для виявлення контурів (або країв) на зображенні. Процес виявлення країв виконується двічі: перший раз – для порогового зображення, щоб виявити зовнішні контури зображення, і другий раз – для згладженого зображення, щоб знайти внутрішні контури зображення. Додавання обох зображень разом гарантує виявлення цих найбільш важливих контурів в знімку;

– фінальний поріг. Остаточний поріг повинен видалити шум, який залишається або вводиться в зображення під час обробки на попередніх етапах;

– медіанна фільтрація. Одним з побічних ефектів використання попередніх процедур обробки зображень є шум. Цей шум представлений наявністю областей на зображенні, які складаються з одного або двох пікселів зі значеннями чорного або білого. Ці регіони зазвичай не становлять реального значення і повинні бути усунені;

– тонкі контури. Основною метою процесу проріджування є зменшення надлишкової інформації про виявлені контури зображення при одночасному збереженні важливої інформації про контурах;

– виявлення перетинів контурів. Кілька контурів зображення можуть перетинатися один з одним, утворюючи складний сегмент кривої. Це впливає на маркування і відстеження контурів. Отже, алгоритм маркування кривої може виявляти пересічені контури як одну єдину «криву» з більш ніж двома кінцевими точками. Таким чином, остаточні результати трасування кривої можуть не відповідати меті створення образу з пропущеними ділянками в сегменті кривої. Для правильного відстеження контурів в зображенні спочатку виділяються пересічні пікселі (точки), спільні для більш ніж одного контура. Ці точки поділяють контури на більш короткі, перш ніж всі контури можуть бути виділені;

– маркування контурів. Контури на зображенні виділяються на цьому етапі для того, щоб простежити всі контури по одному незалежно і ціляком. ;

– трасування контурів. Для ефективного створення образів точки в кожному контурі повинні бути розташовані в правильному порядку. Цей порядок гарантує, що контур трасування починається з першої точки і закінчується останньою точкою. Грунтуючись на цьому спостереженні, процес відстеження контурів починається з сканування пікселів на зображенні, щоб знайти той, який має ознаку контуру. Перевіряючи сусідні пікселі поточного пікселя, якщо тільки один з сусіди належать одному контуру, цей піксель є останнім або першим пікселем контуру. Наступним кроком є додавання цього пікселя в новий список, а потім пошук

наступного сусіднього пікселя, який належить контуру. Після того, як нові пікселі ідентифіковані та додані до списку, вони перетворюються в фонові пікселі. Повторюйте процедуру до тих пір, поки не будуть знайдені всі пікселі в контурі і не відскановані всі контури зображення;

– злиття ламаних контурів. Як згадувалося раніше, виділення загальних пікселів пересічних контурів може розділити відрізок контуру на більш короткі. Це добре для унікального маркування контуру, але може бути неефективним для відстеження контуру і проходження. Для вирішення цієї проблеми пропонується наступний метод. Він починається ітеративно, беручи одну кінцеву точку одного контуру, і сканує кінцеві точки інших контурів по одному, порівнюючи відстань між двома кінцевими точками і різницю в напрямках двох кінцевих точок. Якщо результат цього порівняння знаходиться в межах зазначеного порога, два контури (два списки точок) будуть об'єднані в правильному порядку точок.

Для ефективної роботи з контурами необхідно вирішити дві проблеми:

– видалити зайві точки в кожному контурі зі списку, який представляє контур;

– знайти оптимальну або майже оптимальну послідовність для створення контурів, щоб гарантувати найкоротший час проходження контуру.

Вирішення цих проблем дозволить достанько скоротити подальші виділення ресурсів для роботи алгоритмів, а також підвищить точність визначення образів, їх типів.

Ця мета може бути досягнута в такий спосіб:

– видалення точки. Щоб видалити ці зайві або непотрібні точки в кожному контурі, був обраний алгоритм Дугласа-Пекера. Цей алгоритм може зменшити загальну кількість точок в контурах, одночасно зберігаючи форму цих контурів;

– знаходження кращої послідовності. Алгоритм найближчого сусіда попередньо обраний, щоб знайти найкращу послідовність контурів. Цей алгоритм не вимагає важких і дорогих обчислень, але він дає хорошу можливість для системи наблизитися до оптимального рішення.

4.3 Програмне застосування алгоритму визнаення контурів у системі відеонагляду за допомогою системи розпізнавання зображень

Камера та сервери між собою в системі підключені по протоколу HTTP, і цей протокол вимагає, щоб вся інформація відправлялася короткими фрагментами даних. Отже, кожен контур в зображенні представлений декількома пакетами даних. Сервер упаковує ці пакети і відправляє їх по одному. На стороні іншого сервера ці пакети розпаковуюються і опрацьовуються відповідним чином.

Послідовність пакетів, отриманих сервером, має вирішальне значення для плавного створення ескізів. Тому кожен пакет має два додаткових сегмента, що вказують ідентичність контуру і порядок пакетів в контурі, відповідно.

У якості платформи для розробки було обрано .NET з бібліотекою WPF. Мова C# створювалася як мова об'єктно-орієнтованого програмування.

Однією з головних переваг мови є її спрямованість на можливість повторного використання створених компонентів.

Зважаючи на дуже зручний об'єктно-орієнтований дизайн, C# є гарним вибором для швидкого конструювання різних компонентів – від високорівневої бізнес-логіки до системних додатків, що використовують низькорівневий код.

Інструмент розробки – Visual Studio 2019 від компанії Microsoft.

Завдяки Visual Studio 2019 підвищується ефективність роботи як окремих працівників, так і груп спеціалістів в межах всієї організації ніж час розробки рішень для бізнесу і споживчих переваг.

Це комплексне інтегроване середовище з широкими функціональними можливостями має удосконалений інтерфейс і містить нові інструменти для підтримки багатьох процесів.

Розробники можуть створювати інноваційні і якісні додатки з привабливим інтерфейсом, які перевершать очікування користувачів.

Accord.Net – це фреймворк, заснований на C #, що займається розвитком нейромереж, які використовуються для обробки аудіо та зображень.

Компанії можуть використовувати даний фреймворк в комерційних цілях, наприклад, випускаючи додатки з «комп'ютерним зором», додатки для обробки сигналів, а також додатки для статистики.

Переваги Accord.Net [14]:

- перевірений, протестований алгоритм роботи;
- регулярно оновлюється та підтримується розробниками;
- стабільна робота системи;
- проста під час обробки сигналів;
- легкість роботи з числової оптимізацією та штучними нейромережами.

Недоліки Accord.Net:

- маловідома версія, в порівнянні з іншими фреймворками;
- низька продуктивність.

Опис програмних компонентів системи представлений в табл. 4.1-4.3.

Таблиця 4.1 – Програмні модулі VideoCaptureDeviceForm

№	Позначення
1	FilterInfoCollection videoDevices;
2	private string device;
3	public string VideoDevice
4	public VideoCaptureDeviceForm()
5	private void okButton_Click

Таблиця 4.2 – Програмні модулі MotionRegionsForm

№	Позначення
1	DrawingMode currentMode
2	protected override void OnLoad(EventArgs e)

Кінець таблиці 4.2

№	Позначення
3	public MotionRegionsForm()
4	InitializeComponent();
5	public Rectangle[] MotionRectangles
6	public Bitmap VideoFrame

Таблиця 4.3 – Програмні модулі MainForm

№	Позначення
1	private IVideoSource videoSource
2	MotionDetector detector
3	private int motionDetectionType
4	private const int statLength
5	private int statIndex = 0;
6	private int statReady
7	private List<float> motionHistory
8	private int detectedObjectsCount
9	public MainForm()
10	private void MainForm_FormClosing
11	private void OpenVideoSource(IVideoSource source)
12	private void CloseVideoSource()
13	private void videoSourcePlayer_NewFrame(object sender, NewFrameEventArgs args)
14	private void Application_Idle(object sender, EventArgs e)
15	private void timer_Tick(object sender, EventArgs e)
16	private void SetMotionDetectionAlgorithm(IMotionDetector detectionAlgorithm)

Кінець таблиці 4.3

№	Позначення
17	private void SetMotionProcessingAlgorithm(IMotionProcessing processingAlgorithm)
18	private void localVideoCaptureSettingsToolStripMenuItem_Click(object sender, EventArgs e)

Accord.Controls.Vision – компоненти Windows Forms і елементи управління для відстеження рухів голови, обличчя і рук і інших завдань, пов'язаних з комп'ютерним зором.

При запуску програми користувачу буде надана можливість відкрити камеру свого приладу.

Після цього з'явиться зображення на якому рухи об'єкта будуть виділятися червоним кольором.

Користувач може виділяти рухи на камері за допомогою рамки для навчання мережі, після чого наступне відслідковування буде проходити краще ніж попереднє [28].

Таким чином програма демонструє досягнуті результати за допомогою фреймворку машинного навчання Accord.NET.Vision.

4.4 Рекомендації з використання модулю визначення контурів в системі відеоспостереження

Спроектowana система може бути використана за наступними напрямками:

– розпізнавання автомобільних номерів, найпопулярніша функція у сучасних систем відеоспостереження. При правильно побудованій і налагодженій системі можна домогтися розпізнавання автомобільного номеру з ймовірністю до 95%. Але

навіть така висока ймовірність говорить про те, що повністю автономну систему відеоспостереження з розпізнаванням номерів побудувати не можна. На ці 5% потрібен оператор, який буде приймати правильні рішення;

– розпізнавання обличчя – це більш складне завдання, так як тут немає стандартів, як в розпізнаванні автомобільних номерних знаків, до яких можна «прив'язати» систему. Різноманітність обличчя людей величезна, тому і система має невисоку ймовірність в розпізнаванні. Вимоги до проектування та побудови таких систем неймовірно високі. Найпоширеніша помилка, наприклад, при побудові контролю доступу, що камеру можна поставити в будь-якому місці на вході, і вона буде визначати серед вхідних людей своїх і сторонніх. Але навіть при цих складнощах, побудова систем відеоспостереження з розпізнаванням обличчя може мати свої рішення;

– раннє виявлення займань. До виявлення диму і полум'я звичайними димовими і тепловими детекторами, які зазвичай призводять до значного збитку через активацію розбризкувачів, за допомогою інтелектуальних функцій відеоаналізу осередки займання можуть бути виявлені на ранній стадії. Якщо бізнес-процеси зупиняться на кілька днів через пожежі або затоплення, це може коштувати компанії сотні тисяч доларів у вигляді недоотриманих доходів;

– виявлення руху. Будь-яке небажане переміщення людей або транспортних засобів через певний периметр або територію, може бути легко виявлено за допомогою передових функцій відеоаналізу. При виявленні система може автоматично ініціювати дії, серед яких можуть бути оповіщення, сигнали тривоги, автоматичні словесні попередження або блокування; і все це без зазвичай необхідного втручання людини. Можливості в цій області практично безмежні;

– моніторинг та виявлення місцезнаходження загубленого або вкраденого майна. Інтелектуальні функції відеоаналізу можуть знати, які саме активи повинні бути в полі зору камери, і можуть автоматично створювати оповіщення / сигнали, якщо ці активи будуть відсутні на потрібному місці. Крім того, ви можете легко використовувати смарт-функції пошуку аналітики, щоб швидко знайти втрачене,

вкрадене чи пошкоджене обладнання. Просто визначивши область інтересу в полі зору камери, можна використовувати інтелектуальний пошук, щоб знайти останній рух або діяльність в цьому районі;

– захист будівель. Відеоаналітика може бути використана для здійснення спостереження за всіма хто заходить або виходить з кожних дверей у вашому закладі. Захист областей, які мають вирішальне значення для вашого бізнесу, підтримку своїх бізнес-операцій і забезпечення безперебійної роботи, запобігання зайвих перерв в роботі – це додаткові способи, якими аналітика може зменшити втрати і, в кінцевому рахунку, збільшити рентабельність;

– з використанням систем штучного інтелекту, з використання алгоритмів розпізнавання образів система може безперервно моніторити записи з камер відеоспостереження. Потім система аналізує дані на предмет потенційно небезпечних ситуацій, на які потрібно швидко реагувати. Крім цього, система може бути навчена розпізнавати обличчя в потоці, аналізуючи кадри в реальному часі і зіставляє їх з базою пошуку. При збігу йде оповіщення правоохоронних органів. У системи є модуль аналізу масових зібрань, який шукає по відео скупчення людей. Система може за аналізом відеорядів визначати динаміку кожного образу, а отже визначати дії окремих об'єктів та взаємодію або вплив одного об'єкту на інший при цьому визначаючи характер взаємодії визначати чи пристні в окремих випадках сцени злочинів, насильницьких дій. Якщо буде виявлено подібні випадки, то система зможе автоматично сповістити правоохоронні органи про злочин та надати фото ті відео з місця злочину, а також надати результати розпізнавання осіб або номерів автомобілів, які були задіяні в злочині.

ВИСНОВКИ

Згідно з темою роботи та поставленими задачами дослідження, в атестаційній роботі магістра розроблено алгоритм визначення границь зображень та алгоритми оптимізації процесу визначення границь зображень. Запропонований алгоритм може бути застосований на широкому ряді пристроїв – від промислових серверів до мобільних пристроїв з обмеженими ресурсами.

Для вирішення поставленої задачі було проведено огляд сучасного стану проблеми визначення границь зображень, були проаналізовані існуючі алгоритми з вивченням їх основних недоліків та переваг. Додатково був проведений детальний аналіз існуючих алгоритмів сегментації зображень.

На підставі проведеного аналізу було виконано модифікацію стиснення контурів та відновлення контурів для того, щоб зменшити об'єм використовуваної пам'яті пристроєм та скоротити навантаження на мережу при передачі даних.

Запропонований алгоритм базується на восьми паттернах поведінки трасувальника, що спрощує програмну реалізацію, а відсутність складних математичних розрахунків зменшує навантаження на ресурси пристрою. В той же час дана стратегія зменшує точність розрахунку контуру, оскільки обмежена кількість паттернів не може гарантувати коректне зчитування всіх можливих комбінацій пікслів. З іншого боку, сама ідея алгоритму залишається відкритою для розширення та удосконалення шляхом простого додавання та модифікації існуючих паттернів. Подальше додавання та модифікація паттернів може бути виконано самообучаємою системою з використанням штучних нейронних мереж.

Для подальшого вивчення та оцінки ефективності запропонованого алгоритму було виконано розробку моделі прототипу «proof of concept» програмного продукту, в якому було реалізовано модифікований алгоритм. Основною вимогою до програмного рішення була його простота та дешевизна рішення. Тому було обрано .NET з бібліотекою WPF, а використання готових рішень на початкових

етапах розробки продукту дозволило ефективно протестувати алгоритми та зробити остаточні зміни перед фінальним етапом розробки. В свою чергу використання кросс-платформенної мови C# дозволить виконувати один і той же код на різних пристроях.

В результаті виконання атестаційної роботи було закладено теоретико-методологічне обґрунтування розробки модифікованого алгоритму трасування контурів, розроблено алгоритм та модель прототипу програмного забезпечення для розв'язання задачі трасування контурів.

Сфера застосування – системи автоматизованого відеонагляду з розпізнаванням образів та автоматизованого прийняття рішень. Результати можуть бути використані компаніями, що займаються проектуванням і виготовленням систем відеонагляду з автоматизованим розпізнаванням та прийняттям рішень.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Антонюк В. С. Методологія наукових досліджень: навч. посіб./ В.С. Антонюк, Л. Г. Полонський, В. І. Аверченков, Ю. А. Малахов. – К.: НТУУ «КПІ», 2015. – 286 с.
2. Головкин Б. А. Машинное распознавание и линейное программирование. – М.: Советское радио. 1973. – 100 с.
3. Методы компьютерной обработки изображений. Под редакцией В.А.Сойфера. – М.: Физматлит, 2001. – 784 с.
4. Поспелов Г. С Искусственный интеллект – основа новой информационной технологии – М.: Высшая школа, 1988 – 280 с.
5. Фомин Я. А. Распознавание образов: теория и применения / Я. А. Фомин. – М.: ФАЗИС, 2012. – 429 с.
6. Pavlidis, T. Algorithms for Graphics and Image Processing; Springer-Verlag: Berlin, Germany, 2012.
7. Главач В., Шлезингер М. Десять лекций по статистическому и структурному распознаванию образов. К.: Наукова думка, 2004. – Режим доступа: www.irtc.org.ua/image/Files/Schles/esh10_full.pdf.
8. Вудс Р., Гонсалес Р. Цифровая обработка изображений //М.: Техносфера. – 2005. – С. 44-48.
9. Дуда Р., Харт П. Распознавание образов и анализ сцен. – М.: Мир, 1976. – 674 с.
10. Лепский А. Е., Броневиц А. Г. Математические методы распознавания образов. (Курс лекций). Южный федеральный университет : Таганрог, 2009. – Режим доступа: http://www.lepskiy.ucoz.com/lect_Lepskiy_Bronevich_pass.pdf.
11. Малыгина М.П., Частикова В.А. Программирование на языке высокого уровня с#: учеб. пособие /Кубан. гос. технол. ун-т. – Краснодар: Изд. КубГТУ, 2011.

12. Cheong, C.; Han, T.D. Improved simple boundary following algorithm. J. Korea Inf. Sci. Soc. Softw. Appl. 2006, 33, 427–439.
13. Reddy, P.R.; Amarnadh, V.; Bhaskar, M. Evaluation of stopping criterion in contour tracing algorithms. Int. J. Comput. Sci. Inf. Technol. 2012, 3, 3888–3894
14. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньский, Л. Рутковский. – М. Горячая линия-Телеком 2004. – 112 с.
15. Основи теорії розпізнавання образів : навч. посіб. : у 2 ч. / А. С. Довбиш, І. В. Шелехов. – Суми : Сумський державний університет, 2015. – Ч. 1. – 109 с.
16. Maheswari, D., Radha, V.. Noise Removal in compound image using Median filter, International Journal on Computer Science and Engineering, Vol. 2, p. 1359, 2010.
17. Xu X., Xu S., Jin, L., Song, E.. Characteristic analysis of Otsu threshold and its applications, Elsevier B.V., Pattern Recognition Letters, Vol. 32, p. 956, 2010.
18. Егоров С. В. Оптимизация алгоритмов кластерного анализа в задачах распознавания образов. Херсон, Вестник ХНТУ. - 2011. - № 2(41). С. 172-173.
19. Зайченко С. А., Алисейко З. А. Распознавание цифр методом кластерного анализа. Сб. научных трудов по материалам 7-го Международного молодежного форума "Радиоэлектроника и молодежь в XXI веке". Харьков: 2003
20. Збитнева М. В., Комиссаров А. В. Анализ формата PNG для внедрения цифрового водяного знака. Киев, научно-технический сборник №58 «Коммунальное хозяйство городов»: Серия: Технические науки и архитектура, 2004. С.229-235.
21. Зыбина К. В. Обзор типов нейронных систем для анализа медицинских изображений. Информационные системы и технологии (ИСТ-2018): материалы 7-й Международ. Науч.-техн. конф., Коблево-Харьков, 2018, – тезисы докладов. – Х.: МОНУ, ХНУРЭ, 2018.
22. Малыгина М. П. Аспекты практического применения цветового различия для распознавания и выделения границ изображений / М. П. Малыгина, Д. А.

Шичкин // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета. 2013. Т. 89. № 89-89 (09). С. 676-688.

23. Титаренко Д. П. Модели метрического поиска на множествах изображений. Материалы 12-го Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке», 1-3 апреля 2008 г. Часть 2. Харьков, 2008, С.215.

24. A. Rabotiahov, O. Kobylin, V. Lyashenko. Bionic image segmentation of cytology samples method. Telecommunications and Computer Engineering (TCSET), pp. 665–670, 2018.

25. Kirill Smelyakov, Anastasiya Chupryna, Mykyta Hvozdiev, Denys Sandrkin. Gradational Correction Models Efficiency Analysis of Low-Light Digital Image. // 2019 Open Conference of Electrical, Electronic and Information Sciences (eStream), IEEE. Available online: <https://ieeexplore.ieee.org/abstract/document/8732174>

26. Krishna Kant Singh, Akansha Singh. A Study Of Image Segmentation Algorithms For Different Types Of Images IJCSI International Journal of Computer Science Issues 2010.

27. Mario G.C.A. Cimino, Beatrice Lazzarini and Francesco Marcelloni, A novel approach to fuzzy clustering based on a dissimilarity relation extracted from data using a TS system, Pattern Recognition, 39(11)(2006) 2077-2091

28. Accord Framework.NET. URL:<http://accord-framework.net>.

29. Ghuneim, A. Contour Tracing. 2015. Available online: http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/index.html.

30. Toussaint, G. Grids, Connectivity and Contour Tracing. Available online: <http://www-cgri.cs.mcgill.ca/~godfried/teaching/pr-notes/contour.ps>.