

метадані

Заголовок

2023_ФеП-41с_ЮВЕНКО_Д.І._текст.pdf

Автор

Науковий керівник / Експерт






Даниїл Іовенко**Юрій Фургала**

підрозділ

Факультет електроніки та комп'ютерних технологій

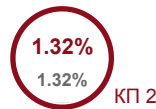
Перелік можливих спроб маніпуляцій з текстом

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про **МОЖЛИВІ** маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		0
Інтервали		0
Мікропробіли		0
Білі знаки		0
Парафрази (SmartMarks)		9

Обсяг знайдених подібностей

Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.

**10**

Довжина фрази для коефіцієнта подібності 2

7353

Кількість слів

61288

Кількість символів

Подібності за списком джерел

Прокручіть список та аналізуйте, особливо, фрагменти, які перевищують КП 2 (позначено жирним шрифтом). Скористайтеся посиланням "Позначити фрагмент" та перегляньте, чи є вони короткими фразами, розкиданими в документі (випадкові схожості), численними короткими фразами поруч з іншими (мозаїчний плагіат) або великими фрагментами без зазначення джерела (прямий плагіат).

10 найдовших фраз

Колір тексту

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)	
1	2023_ФеС-41с_ЧОРНИЙ_Ю.І._текст.pdf 6/9/2023 The Ivan Franko National University (Факультет електроніки та комп'ютерних технологій)	38	0.52 %
2	2023_ФЕІ-43_Чумак_Д_В_текст.pdf 6/9/2023 The Ivan Franko National University (Факультет електроніки та комп'ютерних технологій)	21	0.29 %
3	https://blog.mate.academy/python/learning-python-2023/	16	0.22 %
4	https://openarchive.nure.ua/bitstreams/12d6a3de-acdd-4a75-bb21-20a6eeb6f5f5/download	12	0.16 %
5	https://kanapps.pro/	10	0.14 %

6	Бакалаврська_робота_Лотошинський_Андрій_ФеП-41.pdf 6/9/2023 The Ivan Franko National University (Факультет електроніки та комп'ютерних технологій)	8	0.11 %
7	Kasian_Melkozerova_Diplom_2023 (1).pdf 6/9/2023 V. N. Karazin Kharkiv National University (KGNU) (Факультет комп'ютерних наук - кафедра безпеки інформаційних систем і технологій)	8	0.11 %
8	Kasian_Melkozerova_Diplom_2023 (1).pdf 6/9/2023 V. N. Karazin Kharkiv National University (KGNU) (Факультет комп'ютерних наук - кафедра безпеки інформаційних систем і технологій)	6	0.08 %
9	https://ela.kpi.ua/bitstream/123456789/31165/1/Tselikin_magistr.pdf	5	0.07 %
10	https://ela.kpi.ua/bitstream/123456789/31165/1/Tselikin_magistr.pdf	5	0.07 %

з бази даних RefBooks (0.00 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
------------------	-----------	--

з домашньої бази даних (1.05 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)	
1	2023_ФеС-41с_ЧОРНИЙ_Ю.І._текст.pdf 6/9/2023 The Ivan Franko National University (Факультет електроніки та комп'ютерних технологій)	43 (2)	0.58 %
2	2023_ФЕІ-43_Чумак_Д_В_текст.pdf 6/9/2023 The Ivan Franko National University (Факультет електроніки та комп'ютерних технологій)	21 (1)	0.29 %
3	Бакалаврська_робота_Лотошинський_Андрій_ФеП-41.pdf 6/9/2023 The Ivan Franko National University (Факультет електроніки та комп'ютерних технологій)	13 (2)	0.18 %

з програми обміну базами даних (0.19 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)	
1	Kasian_Melkozerova_Diplom_2023 (1).pdf 6/9/2023 V. N. Karazin Kharkiv National University (KGNU) (Факультет комп'ютерних наук - кафедра безпеки інформаційних систем і технологій)	14 (2)	0.19 %

з Інтернету (0.65 %)

ПОРЯДКОВИЙ НОМЕР	ДЖЕРЕЛО URL	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)	
1	https://blog.mate.academy/python/learning-python-2023/	16 (1)	0.22 %
2	https://openarchive.nure.ua/bitstreams/12d6a3de-acdd-4a75-bb21-20a6eeb6f5f5/download	12 (1)	0.16 %
3	https://kanapps.pro/	10 (1)	0.14 %

Список прийнятих фрагментів (немає прийнятих фрагментів)

ПОРЯДКОВИЙ НОМЕР

ЗМІСТ

КІЛЬКІСТЬ ОДНАКОВИХ СЛІВ (ФРАГМЕНТІВ)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Львівський національний університет імені Івана Франка

Факультет електроніки та комп'ютерних технологій

Кафедра оптоелектроніки та інформаційних технологій

Допустити до захисту

Завідувач кафедри

_____ проф. Кушнір О. С.

«__» _____ 2023 р.

Кваліфікаційна робота

Бакалавр

(освітній ступінь)

Виявлення силуетів людей на зображеннях

Виконав:

студент IV курсу групи ФЕП– 41с

спеціальності 121 – Інженерія

програмного забезпечення

_____ Д.І. ІОВЕНКО

(підпис) (ПІБ)

Науковий керівник:

_____ доц. Фургала Ю. М.

(підпис) (ПІБ)

«__» _____ 2023 р.

Рецензент:

(підпис) (ПІБ)

Львів 2023

ст. 2

АНОТАЦІЯ

Дипломна робота присвячена розробці та дослідженню методів виявлення силуетів людей на зображеннях. Виявлення силуетів є важливим етапом у розпізнаванні об'єктів, комп'ютерному зоровому сприйнятті та багатьох інших застосуваннях. Забезпечення точного виявлення силуетів людей на зображеннях має велике значення для автоматичної ідентифікації, моніторингу, безпеки та інших областей.

ABSTRACT

This diploma work is devoted to the development and research of methods for detecting human silhouettes in images. Silhouette detection is an important step in object recognition, computer vision, and many other applications. Ensuring accurate detection of people's silhouettes in images is of great importance for automatic identification, monitoring, security, and other areas.

ст. 3

ЗМІСТ

АНОТАЦІЯ	2
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ	4 ВСТУП
.....	5
РОЗДІЛ 1 АНАЛІЗ СТАНУ ПРОБЛЕМНОЇ ОБЛАСТІ	7 1.1 Зміна
освітлення	7 1.2 Варіативність позицій тіла
.....	9
1.3 Перекриття об'єктів	12
РОЗДІЛ 2 ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ У РАМКАХ ПРОБЛЕМНОЇ ЗАДАЧІ	16
2.1 Основи обробки зображень	16
2.2 Виявлення силуетів людей	19
2.3 OpenCV	20
2.4 MobileNetSSD	22
2.5 Imutils	23
2.6 Python	24
2.7 PyCharm	26
2.8 Caffe	27
2.9 NumPy	28
2.10 Система реального часу для виявлення силуетів людей	29
РОЗДІЛ 3 ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ	31
3.1 Опис експериментального середовища	31
3.2 Результати досліджень виявлення силуетів на зображеннях	33
3.3 Результати досліджень виявлення силуетів у режимі реального часу	36
ВИСНОВКИ	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	41

ст. 4

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

OpenCV - Open Source Computer Vision Library

CNN - Convolutional Neural Network

FPS - Frames Per Second

RGB - Red, Green, Blue

Датасет - структурована колекція даних, яка використовується для тренування, валідації та оцінки моделей машинного навчання і аналізу даних.

Caffe - Convolutional Architecture for Fast Feature Embedding

ст. 5

ВСТУП

Сучасні досягнення в галузі комп'ютерного зору та обробки зображень відкривають безліч нових можливостей для автоматизації процесів, які пов'язані з розпізнаванням об'єктів на зображеннях. Одним з таких важливих завдань є виявлення силуетів людей на зображеннях.

Виявлення силуетів людей є ключовим етапом у багатьох додатках, таких як автоматична ідентифікація осіб, моніторинг та безпека, аналіз поведінки людей та багато інших. Забезпечення точного та ефективного виявлення силуетів людей на зображеннях є актуальним завданням, яке має велике значення для розвитку різних сфер життя.

У цій дипломній роботі пропонується дослідити та розробити методи виявлення силуетів людей на зображеннях з використанням сучасних технологій комп'ютерного зору та глибокого навчання. Глибоке навчання, зокрема з використанням згорткових нейронних мереж (CNN), стало потужним інструментом у багатьох задачах обробки зображень, і його використання для виявлення силуетів людей може покращити точність та швидкість процесу.

Основні цілі дипломної роботи включають:

1. Розгляд сучасних методів виявлення силуетів людей на зображеннях.
2. Розробка та реалізація нових методів виявлення силуетів з використанням глибокого навчання.
3. Дослідження можливостей використання даних глибини для покращення точності виявлення силуетів.
4. Практична реалізація та експериментальне порівняння розроблених методів з існуючими підходами.
5. Аналіз результатів та висновки щодо ефективності розроблених методів виявлення силуетів людей на зображеннях.

Виконання цієї дипломної роботи сприятиме розширенню наших знань про

виявлення силуетів людей на зображеннях та може мати практичне застосування в різних сферах, включаючи безпеку, медіа, автоматичну ідентифікацію та інші області.

ст. 6

Ця дипломна робота спрямована на розширення нашого розуміння процесу виявлення силуетів людей на зображеннях та внесення вагомому внеску в галузь комп'ютерного зору.

Останні роки свідчать про значний прорив у глибинному навчанні та згорткових нейронних мережах, що дозволяє досягати вражаючих результатів у розпізнаванні об'єктів на зображеннях. Застосування глибинного навчання для виявлення силуетів людей відкриває нові перспективи, дозволяючи автоматично визначати форму тіла та границі об'єктів без необхідності ручного налаштування параметрів.

Усе це зумовлює актуальність теми та необхідність проведення досліджень для розв'язання даної проблеми. Результати цієї роботи можуть мати практичне застосування та сприяти подальшому розвитку сучасних технологій комп'ютерного зору та обробки зображень.

ст. 7

РОЗДІЛ 1 АНАЛІЗ СТАНУ ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Зміна освітлення

Одним із важливих факторів, який впливає на точність та ефективність виявлення силуетів людей на зображеннях, є зміна освітлення. Зміна освітлення може створювати різницю в яскравості, контрастності та тіней на зображеннях, що ускладнює завдання виявлення силуетів.

Залежно від умов освітлення, силуети можуть бути розподілені на зображенні дуже різним чином. Наприклад, при яскравому освітленні силуети можуть бути добре видимими та контрастними, тоді як при слабкому освітленні вони можуть бути поглинуті шумом та малорозрізненими. Також, виникає проблема, коли силуети людей збігаються з тінями, що ускладнює їх розрізнення.

Для подолання впливу зміни освітлення на виявлення силуетів було запропоновано різні підходи. Один із них - адаптивні методи, які забезпечують автоматичну адаптацію параметрів алгоритму до умов освітлення. Це дозволяє компенсувати зміни яскравості та контрастності на зображенні та покращує точність виявлення силуетів у різних умовах.

Ще одним підходом є використання багатофазних методів, які аналізують зображення в різних каналах або використовують багато обробок з різними параметрами. Це дозволяє виявити силуети в різних умовах освітлення та забезпечує більш стійку та надійну роботу системи виявлення силуетів.

Важливим аспектом є також використання стандартних датасетів з різними умовами освітлення для тренування моделей виявлення силуетів. Це дозволяє моделі навчитися розпізнавати силуети в різних освітлених умовах та покращує їх загальну роботу на реальних зображеннях.

У подальших дослідженнях слід продовжувати розробку та вдосконалення методів, які забезпечують стійке та надійне виявлення силуетів людей незалежно від змін освітлення. Такі методи мають враховувати велику варіативність освітлення та бути здатними адаптуватися до різних умов для забезпечення точного виявлення силуетів на зображеннях.

ст. 8

Чутливість систем виявлення силуетів до змін у рівні освітлення є важливим аспектом, який потребує уваги в дослідженнях з даної проблемної області. Зміни в освітленні можуть виникати з різних причин, таких як зміна дня чи ночі, рух об'єктів або зміна джерела світла. Ці зміни можуть суттєво вплинути на якість та точність виявлення силуетів людей.

Одним з основних викликів є чутливість систем до змін у яскравості та контрастності. При зміні освітлення яскравість об'єктів може змінюватися, що може призводити до неправильного виявлення силуетів або невідповідності між силуетом та його контуром на зображенні. Також, зміна контрастності може впливати на розпізнавання границь силуетів та призводити до їхньої некоректної сегментації.

Інший аспект - чутливість до тіней. Зміна освітлення може створювати тіні, які можуть перекривати або спотворювати силуети людей на зображенні. Це ускладнює завдання виявлення силуетів, оскільки системі потрібно вміти розрізнити силуети від тіней та правильно ідентифікувати їх контур.

Щоб зменшити чутливість систем до змін у рівні освітлення, можна використовувати різні підходи. Наприклад, адаптивні методи, які автоматично

адаптують параметри алгоритмів до змін у рівні освітлення, можуть бути корисними. Вони дозволяють системі пристосовуватися до нових умов та забезпечувати стабільне виявлення силуетів незалежно від освітлення.

Також, розробка методів, що враховують специфіку тіней та розрізняють їх від силуетів, може покращити точність та надійність системи. Це може включати аналіз текстур, форми та розташування тіней на зображенні для їхньої коректної обробки та виключення з розпізнавання силуетів.

У подальших дослідженнях слід продовжувати працювати над покращенням чутливості систем до змін у рівні освітлення. Це може включати розробку нових алгоритмів, використання розширених датасетів з різними умовами освітлення, а також інтеграцію з додатковими сенсорами або системами, що допомагають враховувати та коригувати зміни освітлення у реальному часі.

ст. 9

Адаптація до різних умов освітлення є важливим аспектом у виявленні силуетів людей на зображеннях. Це дозволяє системі ефективно працювати в різних умовах освітлення та забезпечує точність та надійність виявлення силуетів.

Один з підходів до адаптації до різних умов освітлення - це використання адаптивних методів. Ці методи автоматично адаптують параметри алгоритмів виявлення силуетів до поточних умов освітлення. Наприклад, можуть використовуватись адаптивні пороги яскравості або контрастності, що дозволяють системі підлаштовуватись під зміни в рівні освітлення та підвищувати ефективність виявлення силуетів.

Інший підхід - використання багатофазних методів. Замість використання одного алгоритму виявлення силуетів, можуть використовуватись кілька обробок з різними параметрами або в різних каналах. Наприклад, можуть використовуватись різні фільтри або методи сегментації для отримання кількох варіантів силуетів. Потім можна поєднати результати цих обробок або застосувати додаткові правила для вибору найкращого силуету у відповідності до умов освітлення.

Також, використання статистичних методів може бути корисним для адаптації до різних умов освітлення. Наприклад, можна провести попередній аналіз зображення для визначення характеристик освітлення, таких як середнє значення яскравості, контрастності, розподіл тіней тощо. На основі цих характеристик можна вибрати або налаштувати відповідні методи обробки, що дозволяє системі адаптуватися до конкретного рівня освітлення.

У подальших дослідженнях слід зосередитися на розробці більш ефективних методів адаптації до різних умов освітлення. Важливо розуміти, що різні сценарії освітлення можуть мати різні виклики, тому потрібно досліджувати широкий спектр методів та їх комбінацій для забезпечення надійного та точного виявлення силуетів незалежно від умов освітлення.

1.2 Варіативність позицій тіла

Виявлення силуетів людей на зображеннях може бути викликаною проблемою через різні позиції тіла, які можуть виникати у реальних ситуаціях. Різні позиції тіла

ст. 10

впливають на форму, розмір та орієнтацію силуету, що може ускладнювати процес виявлення.

Одним з викликів є варіативність розміру силуету в залежності від позиції тіла.

При зміні позиції тіла, наприклад, коли людина сидить або стоїть, розмір силуету може змінюватися. Це може впливати на роботу алгоритмів виявлення, оскільки їм потрібно бути гнучкими та здатними розпізнавати силуети різних розмірів.

Додатковим викликом є зміна форми силуету в залежності від позиції тіла.

Наприклад, коли людина сидить, форма силуету може змінюватися через згинання ніг та рук. Це може впливати на точність та надійність виявлення, оскільки алгоритми повинні враховувати ці зміни та коректно ідентифікувати контур силуету.

Орієнтація тіла також може впливати на виявлення силуетів. Зміна орієнтації тіла може призводити до зміни площості проекції та спотворення форми силуету. Це створює додаткові виклики для алгоритмів виявлення, оскільки вони повинні бути здатними розпізнавати та коректно відновлювати силуети незалежно від орієнтації тіла.

У подальших дослідженнях слід зосередитися на розробці методів, які враховують різні позиції тіла та їх вплив на виявлення силуетів. Це може включати розробку алгоритмів, що адаптуються до зміни розміру та форми силуету, використання додаткових дескрипторів або моделей для розпізнавання позиції тіла та використання технік розширеної реалістичності для удосконалення виявлення силуетів у різних орієнтаціях.

Розпізнавання силуетів незалежно від позиції тіла є важливою задачею в системах виявлення людей на зображеннях. Це означає, що система повинна бути здатною розпізнавати силует незалежно від різних позицій тіла, таких як стоячи, сидючи, рухаючись або змінюючи орієнтацію.

Одним з підходів до розпізнавання силуетів незалежно від позиції тіла є використання глибинних нейронних мереж. Ці моделі можуть навчатися розпізнавати силуети людей з різних позицій та орієнтацій шляхом аналізу великої кількості даних. Вони можуть використовувати конволюційні шари для виявлення образів та ознак у зображеннях та підвищення їх інваріантності до зміни позицій тіла.

ст. 11

Інший підхід полягає в розробці методів, що використовують моделі та алгоритми для розпізнавання ключових точок або структури тіла людини. Наприклад, можуть використовуватись моделі з позиційними точками, які розпізнають основні частини тіла, такі як голова, руки, ноги тощо, та встановлюють їх взаємне положення та зв'язки. Це дозволяє системі аналізувати позицію та форму тіла незалежно від його конкретної позиції на зображенні.

Крім того, використання методів аналізу руху може бути корисним для розпізнавання силуетів незалежно від позиції тіла. Це означає виявлення та відстеження рухомих об'єктів на зображеннях та аналіз їхнього руху у просторі. За допомогою цих методів можна відокремити людей від інших об'єктів та врахувати їх рухові характеристики незалежно від їхньої позиції тіла.

У подальших дослідженнях варто розвивати ці підходи та вдосконалювати методи розпізнавання силуетів незалежно від позиції тіла. Забезпечення високої точності та надійності виявлення силуетів у різних сценаріях та позиціях тіла є важливим завданням для розвитку систем виявлення людей на зображеннях. Виявлення силуетів людей на зображеннях, незалежно від їхніх позицій тіла, є важливою задачею в дослідженні проблемної області. Для розв'язання цієї задачі були розроблені різні методи, які враховують різні позиції тіла людини. Надалі будуть описані деякі з них:

1. Методи, що базуються на моделях людського тіла: Ці методи використовують моделі, що описують форму та структуру людського тіла. Вони можуть включати моделі з позиційними точками (наприклад, ключові точки, що представляють руки, ноги, голову тощо) або моделі з параметризованим сегментуванням тіла. Ці моделі дозволяють враховувати різні позиції тіла та робити відповідні корекції при виявленні силуету.

2. Методи, що використовують контекстну інформацію: Ці методи враховують контекст навколишнього середовища та інших об'єктів на зображенні для виявлення силуету. Наприклад, аналізуючи контекстну інформацію, можна врахувати можливу позицію тіла людини та розміщення силуету на зображенні.

ст. 12

Це може поліпшити точність виявлення та допомогти розрізнити силует людини від інших об'єктів.

3. Методи, що використовують глибину зображення: Ці методи використовують інформацію про глибину зображення для виявлення силуету людини. Вони можуть використовувати технології, такі як стереозоровість або розрахунок глибини з використанням сенсорів, щоб визначити глибину об'єктів на зображенні. Ця інформація про глибину може бути корисною для вирішення проблеми розпізнавання силуету людини незалежно від її позиції.

Ці методи тільки декілька прикладів того, як можна враховувати різні позиції тіла людини для виявлення силуету. Дослідження в цій області продовжується, і додаткові методи та підходи можуть бути розроблені для поліпшення точності та надійності виявлення силуетів незалежно від позиції тіла людини на зображеннях.

1.3 Перекриття об'єктів

При виявленні силуетів людей на зображеннях однією з основних проблем може бути перекриття силуетів іншими об'єктами. Це виникає, коли люди частково або повністю приховані іншими об'єктами на зображенні, такими як меблі, транспортні засоби, рослини або інші люди. Це може призводити до неточностей та помилок в процесі виявлення силуетів та розпізнавання людей.

Одним з підходів до вирішення проблеми перекриття силуетів є використання алгоритмів сегментації об'єктів. Ці алгоритми дозволяють виділити окремі об'єкти на зображенні, включаючи людей, шляхом визначення меж областей з відмінною текстурою, контрастом або кольором. Після сегментації можна використовувати додаткові методи для виявлення силуету людини всередині виділеної області.

Інший підхід включає використання алгоритмів, що базуються на глибинній інформації. Використання технологій, таких як стереозорівість або відстань до об'єктів, може допомогти вирішити проблему перекриття, оскільки глибина зображення надає інформацію про відстань між об'єктами. З цією інформацією можна

ст. 13

встановити, які об'єкти перекриваються, і виключити їх вплив на виявлення силуету людини.

Також можна використовувати методи, що базуються на контекстному аналізі. Вони використовують інформацію про контекст та семантику сцени, щоб виявити силует людини, навіть якщо він перекритий іншими об'єктами. Наприклад, аналізуючи форму та розташування інших об'єктів на зображенні, можна встановити, де потенційно може бути силует людини.

Усі ці підходи вимагають розробки складних алгоритмів та використання комп'ютерного зору для виявлення силуетів людей, незважаючи на проблему перекриття. Дослідження в цій області продовжується, і нові методи та підходи розробляються для поліпшення точності та надійності виявлення силуетів навіть у випадках перекриття іншими об'єктами.

Проблема перекриття силуетів іншими об'єктами на зображеннях створює виклик для точної сегментації та розпізнавання силуетів людей. Для вирішення цієї проблеми були розроблені різні методи, які враховують перекриття та дозволяють відокремити силуети людей від інших об'єктів. Декілька з них описані нижче:

Перші, що базуються на геометричних властивостях: Ці методи використовують геометричні властивості об'єктів на зображенні для визначення меж силуетів. Вони враховують форму, розташування та контекст об'єктів для вирішення проблеми перекриття. Наприклад, можна використовувати геометричні правила, такі як закон збереження об'єму або контури об'єктів, для визначення силуетів навіть при наявності перекриття.

Другі, що використовують інформацію про текстуру: Ці методи аналізують текстурні характеристики областей на зображенні для виявлення силуетів. Вони шукають відмінності у текстурі між силуетами людей та іншими об'єктами, що перекривають їх. Це може включати використання фільтрів, дескрипторів або статистичних методів для виділення текстурних особливостей силуетів людей. Треті, що базуються на машинному навчанні: Ці методи використовують навчені моделі або алгоритми для сегментації та розпізнавання силуетів. Вони можуть використовувати набір даних, що містить як силуети людей, так і інші об'єкти, як

ст. 14

навчальний набір для тренування моделей. Моделі машинного навчання, такі як нейронні мережі, можуть виявляти та розпізнавати силуети людей навіть при наявності перекриття, оскільки вони вчаться розпізнавати патерни і характеристики об'єктів.

Ці методи не є вичерпним списком, але представляють загальну ідею підходів, що використовуються для сегментації та розпізнавання силуетів при наявності перекриття. Дослідження в цій області продовжується, і нові методи та підходи розробляються для поліпшення точності та надійності виявлення силуетів людей навіть у випадках перекриття іншими об'єктами.

Розпізнавання силуетів людей на зображеннях при наявності перекриття є складною задачею, і вона продовжує привертати увагу дослідників у сфері обробки зображень та комп'ютерного зору. Останні роки були спостережені значні досягнення у розвитку методів, що враховують перекриття об'єктів, які спрямовані на поліпшення точності та надійності виявлення силуетів людей. Нижче наведено кілька напрямків розвитку цих методів:

Глибинне навчання та нейронні мережі. Глибинне навчання, зокрема використання нейронних мереж, стало сильним інструментом у розвитку методів, що враховують перекриття об'єктів. Нейронні мережі можуть вчитися розпізнавати силуети людей навіть при наявності перекриття шляхом аналізування контексту та геометрії об'єктів на зображенні. Використання глибинного навчання дозволяє моделям автоматично виявляти та враховувати перекриття об'єктів при сегментації силуетів людей.

Використання контекстуальної інформації. Для поліпшення сегментації силуетів при перекритті використовуються методи, які аналізують контекстуальну інформацію на зображенні. Це означає врахування місця розташування об'єктів, їх форми та розміру, а також спільного контексту з оточуючими об'єктами. Інформація про контекст може допомогти визначити, які об'єкти перекривають силует людини, і уникнути помилкового виявлення.

Комбінація даних з різних джерел. Використання даних з різних джерел може

допомогти врахувати перекриття об'єктів. Наприклад, комбінування інформації з

ст. 15

різних камер або сенсорів може забезпечити більш повну та точну інформацію про силует людини, навіть якщо він перекритий. Такі методи потребують синхронізації та обробки даних з різних джерел, але можуть значно покращити результати сегментації та розпізнавання силуетів.

ст. 16

РОЗДІЛ 2 ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ У РАМКАХ ПРОБЛЕМНОЇ ЗАДАЧІ

2.1 Основи обробки зображень

Представлення зображень є ключовим аспектом обробки зображень.

Зображення може бути представлене у різних форматах та структурах, залежно від його властивостей та призначення обробки. У рамках дипломної роботи про виявлення силуетів людей, ефективне представлення зображень відіграє важливу роль у забезпеченні точності та надійності процесу виявлення.

Основними методами представлення зображень є матричне представлення, де зображення може бути представлене у вигляді матриці пікселів, де кожен піксель має числове значення, що відображає його яскравість або колір. У залежності від типу зображення (чорно-біле, кольорове), матриця може мати один або кілька каналів.

Також, векторне представлення, де зображення може бути перетворене в вектор, де кожен елемент вектора представляє значення пікселя або певну характеристику зображення. Цей підхід часто використовується при використанні методів машинного навчання або статистичного аналізу зображень.

Частотне представлення. Зображення може бути перетворене у просторі частот, наприклад, за допомогою двовимірного перетворення Фур'є. Частотне представлення зображень дозволяє виявити частотні характеристики, такі як текстури, контури та шаблони.

Наостанок, символічне представлення, що може бути перетворене у послідовність символів або кодів, що представляють різні аспекти зображення, наприклад, за допомогою методів компресії або векторного квантування.

Вибір конкретного методу представлення залежить від потреб аналізу та обробки зображень. Наприклад, для виявлення силуетів можуть бути використані матричне представлення зображень або векторне представлення, де значення пікселів використовуються як ознаки для подальшого аналізу.

ст. 17

Важливо розуміти, що правильний вибір методу представлення зображень може позитивно позначитися на якості та ефективності алгоритмів виявлення силуетів. Під час розробки методу виявлення силуетів на зображеннях, варто уважно вивчити характеристики зображень та обрати найбільш підходящий спосіб їх представлення для досягнення оптимальних результатів.

Операції обробки зображень є важливою складовою при аналізі та обробці зображень. Вони дозволяють виконувати різноманітні маніпуляції з зображеннями, що сприяють поліпшенню якості зображення, виділенню важливих деталей та підготовці даних для подальшого аналізу. У контексті дипломної роботи про виявлення силуетів людей, операції обробки зображень можуть бути використані для покращення виявлення силуетів та підготовки зображень до алгоритмів обробки.

- Фільтрація: Використовується для зменшення шуму на зображенні та покращення якості. Методи фільтрації можуть включати медіанний фільтр, фільтр Гаусса або фільтр розмивання.
- Морфологічні операції: Використовуються для виділення об'єктів та формування контурів. Морфологічні операції включають ерозію (зменшення об'єктів), дилатацію (збільшення об'єктів), відкриття (ерозія, а потім дилатація) та закриття (дилатація, а потім ерозія).
- Бінаризація: Використовується для перетворення зображення в двохвимірну бінарну матрицю, де пікселі віднесені до фону або об'єкту. Бінаризація дозволяє виділити силуети та відокремити їх від фону.
- Виділення контурів: Використовується для виділення контурів об'єктів на зображенні. Методи виділення контурів можуть включати алгоритми Салпу, Собеля або використання градієнтів.
- Морфологічна сегментація: Використовується для виділення областей, що мають спільні морфологічні ознаки. Це може бути корисним для виділення регіонів, що містять силуети людей.

ст. 18

- Видалення тла: Використовується для видалення фону зображення та залишення тільки силуетів об'єктів. Це допомагає покращити точність виявлення силуетів.

Виділення контурів на зображеннях є важливою задачею в обробці зображень. Контур відображає зміни інтенсивності пікселів та представляє зовнішній контур об'єкта на зображенні. Ця інформація може бути використана для виділення об'єктів, виявлення границь та подальшого аналізу зображень.

Для виділення контурів на зображеннях існують різні методи. Деякі з них використовують градієнти яскравості, щоб визначити місця різких змін інтенсивності, які відповідають контурам об'єктів. Ці методи включають алгоритми Санну та Собеля.

Алгоритм Санну є одним з найпоширеніших методів виділення контурів. Він базується на розпізнаванні різниці градієнтів і використовує ряд кроків, включаючи згладжування зображення для зменшення шуму, визначення градієнтів яскравості зображення, використання порогового значення для видалення слабких градієнтів, використання порогового значення для видалення неважливих контурів, використання порогового значення для з'єднання контурів та отримання остаточного результату.

Алгоритм Собеля є іншим популярним методом для виділення контурів. Він використовує матриці згладжування та вираховує градієнти яскравості у горизонтальному та вертикальному напрямках. Після цього градієнти комбінуються, щоб отримати остаточний результат з виділеними контурами.

В обох випадках результатом є бінарне зображення, де контури позначені значенням 1, а фон - значенням 0. Це дозволяє подальшу обробку та аналіз силуетів на зображенні.

Вибір конкретного методу виділення контурів залежить від вимог дослідження та властивостей зображень. Кожен метод має свої переваги та обмеження, тому важливо обирати той, який найкраще відповідає поставленій задачі.

ст. 19

2.2 Виявлення силуетів людей

Задача виявлення силуетів виникає в багатьох областях, пов'язаних з обробкою зображень та комп'ютерним зором. В основі цієї задачі лежить необхідність виділення контуру або області, що відповідає формі та положенню об'єкта на зображенні.

Однією з основних причин призначення задачі виявлення силуетів є отримання важливої інформації про форму та положення об'єктів на зображенні. Виявлення силуетів може бути першим кроком у подальшому аналізі об'єктів, таких як відстеження руху, розпізнавання об'єктів, оцінка геометричних параметрів тощо.

Виявлення силуетів має широке застосування в таких областях як комп'ютерне зорове спостереження, розпізнавання об'єктів, аналіз руху, віртуальна реальність та аугментована реальність.

Наприклад, в першому випадку виявлення силуетів дозволяє відстежувати об'єкти на зображеннях з камери для різних цілей, включаючи безпеку, нагляд за дорожнім рухом, відстеження людей та тварин тощо. Якщо ж дивитись на розпізнавання об'єктів, то можливе використання для розпізнавання об'єктів на зображеннях, таких як обличчя, автомобілі, руки тощо. Це важливо у таких сферах, як автоматичне відкриття дверей, розпізнавання жестів та інтерактивні системи.

Дивлячись на аналіз руху, можна зрозуміти, що можна використати виявлення силуетів для відстеження руху об'єктів на зображеннях. Це важливо для систем відеоспостереження, аналізу руху людей та транспорту, а також для виявлення аномальних подій.

Виявлення силуетів може бути використане для інтерактивних додатків, які взаємодіють з реальним середовищем. Це може включати розміщення віртуальних об'єктів в реальному часі або інтерактивні ігри.

ст. 20

2.3 OpenCV

OpenCV - це бібліотека відкритого коду, яка надає набір інструментів для розробки програмного забезпечення з обробки зображень та комп'ютерного зору. Вона розроблена з метою надання розширених можливостей у роботі з зображеннями та відео на різних платформах.

OpenCV написана на C++ і має інтерфейси для використання з багатьма іншими мовами програмування, включаючи Python та Java. Ця бібліотека володіє потужними функціональними можливостями, що дозволяють виконувати розпізнавання обличчя, виявлення об'єктів, аналіз зображень, обробку відео, побудову тривимірних моделей і багато іншого.

OpenCV надає широкий спектр алгоритмів та методів для обробки зображень, таких як фільтрація, згортка, морфологічні операції, сегментація та видалення шуму. Вона також пропонує готові реалізації популярних алгоритмів машинного навчання, таких як метод опорних векторів (SVM), нейронні мережі та байесові класифікатори, що дозволяє застосовувати їх у завданнях комп'ютерного зору.

Однією з ключових особливостей OpenCV є його підтримка різноманітних апаратних пристроїв, включаючи відеокамери, графічні прискорювачі та спеціалізовані сенсори. Вона також інтегрується з іншими популярними бібліотеками, такими як NumPy та SciPy, що розширює її можливості і спрощує розробку складних алгоритмів комп'ютерного зору.

OpenCV знаходить широке застосування у багатьох галузях, включаючи робототехніку, автоматичне водіння, медичну діагностику, контроль якості, аналіз зображень і відеоспостереження. Вона є основним інструментом для дослідників, студентів та професіоналів, які працюють у сфері обробки зображень та комп'ютерного зору.

Завдяки своїй відкритості, потужності та активній спільноті розробників, OpenCV продовжує розвиватися і покращуватися з кожним новим випуском. Вона залишається однією з найпопулярніших бібліотек для обробки зображень та

ст. 21

комп'ютерного зору і продовжує допомагати в розв'язанні складних завдань в цих галузях.

Зважаючи на широкий спектр функцій та можливостей, OpenCV є незамінним інструментом для багатьох завдань обробки зображень та комп'ютерного зору. Слід зазначити кілька ключових аспектів, які роблять OpenCV таким популярним:

Мультиплатформеність: OpenCV підтримує різні операційні системи, такі як Windows, Linux, macOS, Android і iOS. Це дозволяє розробникам створювати кросплатформенні програми та застосунки, що працюють на різних пристроях.

Багатофункціональність: OpenCV має багато вбудованих функцій для обробки зображень, відео та потоку даних. Вона включає алгоритми для виявлення облич, розпізнавання об'єктів, вимірювання, відстеження руху, калібрування камери та багато інших. Вона також підтримує роботу з різними форматами зображень і відео, включаючи JPEG, PNG, TIFF, MPEG і багато інших.

Простота використання: OpenCV надає зрозумілий та легкий у використанні API, що спрощує розробку програм, особливо для початківців. Вона має широку документацію, приклади коду та активну спільноту розробників, що допомагають вирішувати проблеми та навчатися новим функціям.

Підтримка машинного навчання: OpenCV інтегрується з популярними бібліотеками машинного навчання, такими як TensorFlow і PyTorch, що дозволяє використовувати сучасні моделі глибокого навчання для розв'язання задач обробки зображень. Це відкриває нові можливості для виявлення об'єктів, класифікації, сегментації та інших завдань комп'ютерного зору.

Велика активна спільнота: OpenCV **має велику спільноту розробників, яка активно співпрацює, обмінюється досвідом та** внесенням внесків до розвитку бібліотеки. Це забезпечує постійну підтримку, оновлення та вирішення проблем, що виникають у користувачів.

ст. 22

2.4 MobileNetSSD

Є легким та ефективним архітектурним блоком для глибокого навчання, спеціально розробленим для ресурсозбереженого обчислення на мобільних та вбудованих пристроях. Він базується на ідеї глибоких згорткових нейронних мереж, які використовують 3x3 згорткові фільтри з малим кількістю каналів для отримання компактних та швидких моделей.

MobileNet використовує глибокі згортки з пакетною нормалізацією та функцією активації ReLU6, яка обмежує виходи в діапазоні [0, 6]. Це допомагає забезпечити ефективні обчислення з низькими вимогами до ресурсів, зменшуючи кількість операцій та параметрів.

SSD є методом об'єктного виявлення, що базується на глибоких нейронних мережах. Він пропонує ефективну стратегію для одноетапного виявлення об'єктів у зображеннях. Замість послідовного застосування згорткових шарів та підвибірок для виявлення об'єктів, SSD використовує множину згорткових шарів з різною роздільною здатністю для виявлення об'єктів різних розмірів.

Поєднує модель MobileNetSSD для створення компактною та швидкою моделі для об'єктного виявлення. Вона використовує MobileNet як базовий блок витягування ознак з зображення та додаткові сверточні шари для виявлення об'єктів різних масштабів та розмірів.

Рисунок 2.1 Архітектура MobileNetSSDV2

MobileNetSSD може виявляти об'єкти різних класів, такі як автомобілі, пішоходи, велосипеди тощо, шляхом здійснення прогнозування класів та регресії рамок для точного обмеження місцезнаходження об'єктів на зображенні.

ст. 23

Ця модель є особливо корисною для застосувань у сфері комп'ютерного зору на мобільних пристроях або пристроях з обмеженими ресурсами, де ефективність та швидкодія є критичними факторами.

2.5 Imutils

Imutils - це бібліотека утиліт, яка спрощує обробку зображень та роботу з комп'ютерним зором в OpenCV. Вона надає ряд додаткових функцій та інструментів, які допомагають зробити процес обробки зображень більш зручним та ефективним. Завдяки imutils ви можете легко завантажувати та зберігати зображення, змінювати їх розмір, працювати з кольорами та каналами, обрізати та вирівнювати зображення, виконувати перехресну кореляцію та відстежування руху.

Ця бібліотека допомагає зосередитися на розробці алгоритмів обробки зображень, забезпечуючи зручні та ефективні інструменти для взаємодії з OpenCV. Використання imutils дозволяє прискорити процес розробки проектів з обробки зображень та комп'ютерного зору, роблячи їх більш зручними та ефективними. Imutils надає такі можливості як

Imutils надає такі можливості як

- Завантаження та збереження зображень: imutils містить функції для зручного завантаження зображень з диска та їх збереження. Це дозволяє швидко і просто працювати зі зображеннями у різних форматах.
- Зміна розміру зображень: imutils має функцію для зміни розміру зображень, що дозволяє змінювати їх ширину, висоту або співвідношення сторін. Це особливо корисно при обробці зображень з різними розмірами або при побудові моделей машинного навчання.
- Робота з кольорами та каналами зображень: imutils надає функції для розділення та об'єднання каналів зображень, конвертації між кольоровими просторами, а також для отримання гістограми кольорів. Це дозволяє виконувати різноманітні операції з кольорами та використовувати їх у алгоритмах обробки зображень.

ст. 24

- Обрізання та вирівнювання зображень: imutils містить функції для обрізання зображень до заданої області та автоматичного вирівнювання зображень. Це корисно при виконанні завдань, таких як вирізування облич, створення датасетів для машинного навчання тощо.
- Перехресна кореляція та відстеження руху: imutils надає функції для виконання перехресної кореляції між двома зображеннями, що дозволяє знайти схожість між ними. Вона також має інструменти для відстеження руху об'єктів на відео. Загалом це потужний інструмент для швидкої та зручної обробки зображень в OpenCV. Який допомагає спростити розробку програм, забезпечує багато корисних функцій та допомагає прискорити процес розробки проектів обробки зображень та комп'ютерного зору.

Трансляція - це зсув зображення у напрямку x або y . Щоб перекласти зображення у OpenCV, вам потрібно вказати (x, y) -зсув, позначений як (tx, ty) , для побудови матриці перекладу M .

2.6 **Python** Python - це високорівнева, інтерпретована мова програмування загального призначення. Вона була розроблена Гвідо ван Россумом і вперше випущена у 1991 році. Python використовує простий та зрозумілий синтаксис, що дозволяє розробникам писати код швидше та зрозуміліше. Вона має велику стандартну бібліотеку, що включає різноманітні модулі та функції для виконання різних завдань. Є інтерпретованою мовою, що дозволяє програмам виконуватися рядок за рядком. Вона також багатоплатформова, тобто програми написані на Python можуть запускатися на різних операційних системах. Python підтримує розширення за допомогою модулів та пакетів сторонніх розробників. Вона знаходить застосування в різних сферах, включаючи веб-розробку, наукові обчислення, штучний інтелект,

ст. 25

аналітику даних та багато іншого. Python є популярним вибором серед розробників завдяки своїй простоті, гнучкості та великому співтовариству підтримки.

Простий та зрозумілий синтаксис, відрізняється лаконічним і зрозумілим синтаксисом, що робить код більш читабельним та легким для розуміння. Це зменшує кількість потенційних помилок та сприяє швидкому розробленню програм.

Інтерпретована мова, що означає, що код виконується по одному рядку за раз.

Це дозволяє швидше відлагоджувати та тестувати програми, спрощує прототипування та зменшує час розробки.

Розширюваність Python підтримує використання зовнішніх модулів та бібліотек, що дозволяє розширювати функціональність мови та використовувати готові рішення. У Python є широкий вибір сторонніх пакетів, які допомагають вирішувати різноманітні задачі.

Широко застосовується у багатьох галузях, включаючи веб-розробку, наукові обчислення, штучний інтелект, аналітику даних, розробку ігор та багато іншого. Ця мова має велику спільноту розробників, що допомагає вирішувати проблеми та підтримувати постійний розвиток.

Підтримується на різних операційних системах, таких як Windows, macOS, Linux, що дає розробникам можливість використовувати мову без необхідності переписування коду під різні платформи.

Велика стандартна бібліотека Python має обширну стандартну бібліотеку, яка включає багато корисних модулів та функцій для виконання різних завдань. Це дозволяє розробникам ефективно виконувати різні операції без необхідності встановлення додаткових сторонніх бібліотек.

ст. 26

2.7 PyCharm

PyCharm є інтегрованим середовищем розробки (IDE) для мови програмування Python, яке надає розробникам розширені можливості та зручні інструменти для ефективної роботи з Python-проектами.

Одна з головних переваг PyCharm полягає в його здатності до синтаксичного виділення та автодоповнення коду. Це робить процес написання коду швидшим та менш помилковим, оскільки розробник отримує підказки щодо доступних функцій, методів та атрибутів у режимі реального часу. PyCharm також підтримує відстеження типів даних, що полегшує розуміння структури проекту та покращує його сумісність. Інший важливий аспект PyCharm - його потужний відлагоджувач. Він дозволяє розробникам крокувати по коду, встановлювати точки зупинки, аналізувати значення змінних та виявляти помилки в процесі виконання програми. Це допомагає виявляти й виправляти проблеми у коді та полегшує процес розробки та налагодження.

PyCharm також має інтегровані інструменти для керування проектами. Воно дозволяє створювати та керувати проектами на Python, а також автоматично виявляти та встановлювати залежності. PyCharm також підтримує різні системи контролю версій, такі як Git, SVN та Mercurial, що дозволяє зручно відстежувати та керувати змінами в коді проекту.

Окрім цього, PyCharm пропонує інструменти аналізу коду, які допомагають знайти потенційні помилки, стильові проблеми та пропонують оптимізації для покращення продуктивності та якості коду. Ці функції допомагають зберегти час та зусилля розробників, сприяючи покращенню робочого процесу.

Крім стандартного функціоналу, PyCharm може бути розширений за допомогою плагінів, які додають додаткові можливості та інтеграцію з іншими інструментами. Розробники можуть створювати свої власні плагіни або використовувати існуючі, що робить PyCharm ще більш гнучким та придатним для використання у різних сценаріях розробки.

ст. 27

Загалом, PyCharm є потужним інструментом для розробки на Python, який надає розробникам зручне та продуктивне середовище для роботи над проектами. Його широкий функціонал, відмінна підтримка та зручний інтерфейс роблять PyCharm популярним серед розробників Python.

2.8 Caffe

Caffe (Convolutional Architecture for Fast Feature Embedding) - це фреймворк глибокого навчання, спеціалізований на обробці зображень. Він розроблений з урахуванням швидкості та ефективності та надає можливість побудови та тренування нейронних мереж для розпізнавання об'єктів, виявлення силуетів та інших задач комп'ютерного зору.

Основні особливості Caffe:

- Модульна архітектура: Caffe має модульну структуру, яка дозволяє легко поєднувати різні шари нейронних мереж для створення складних архітектур моделей. Він має багато передвстановлених шарів, які можуть бути використані для швидкого створення та налаштування моделей.

- Підтримка GPU: Caffe підтримує використання графічних процесорів (GPU) для прискорення обчислень. Це дозволяє ефективно обробляти великі обсяги даних та навчати складні моделі на потужних обчислювальних ресурсах.
- Широкий набір функцій: Caffe надає багато функціональних можливостей для обробки зображень, таких як конволюційні шари, пулінг, нелінійні активації, розпізнавання об'єктів, виявлення силуетів та багато іншого. Він також підтримує різні функції втрати для тренування моделей.
- Переносимість: Caffe підтримує різні платформи, включаючи Linux та Windows. Це дає можливість розробляти та запускати моделі на різних обчислювальних пристроях.
- Велике співтовариство користувачів: Caffe має широке співтовариство активних користувачів та розробників, що забезпечує підтримку, документацію

ст. 28

та навчальні матеріали. Це робить Caffe популярним вибором для дослідників та розробників у галузі обробки зображень.

Caffe використовується для розв'язання широкого спектру задач у сфері комп'ютерного зору, включаючи виявлення силуетів, розпізнавання об'єктів, класифікацію зображень та багато іншого. Його швидкість, простота використання та багата функціональність роблять його потужним інструментом у дослідженні з обробки зображень.

2.9 NumPy

NumPy (Numerical Python) - це бібліотека для мови програмування Python, яка **надає підтримку для роботи з багатовимірними масивами та матрицями.** Вона є **однією з основних бібліотек для** наукових обчислень у Python та використовується широкою громадою дослідників та розробників.

Основні можливості NumPy включають багатовимірні масиви, універсальні функції, індексацію та зрізи, бродкастинг, лінійну алгебру та інтеграцію з іншими бібліотеками. NumPy є потужним інструментом для роботи з числовими даними та виконання наукових обчислень. Вона забезпечує швидку та ефективну обробку масивів даних, що робить її необхідною для багатьох досліджень та проектів в галузі машинного навчання, обробки зображень, аналізу даних та багатьох інших областях. NumPy дозволяє зручно працювати з числовими даними та виконувати різноманітні обчислення. Вона надає широкий набір функцій для виконання операцій лінійної алгебри, таких як обчислення матричних операцій, розв'язування систем лінійних рівнянь, обчислення власних значень та векторів, обернення, масштабування та зсув масивів, і багато інших.

Одним із ключових аспектів NumPy є робота з багатовимірними масивами. Вона дозволяє створювати та маніпулювати масивами різних розмірностей, що відкриває широкі можливості для обробки даних. Завдяки оптимізованій реалізації внутрішніх операцій, NumPy забезпечує швидку обробку великих об'ємів даних.

ст. 29

Крім того, NumPy надає багато функцій для індексування та зрізів масивів, що дозволяє отримувати підмасиви за певними умовами, змінювати значення елементів масиву, виконувати операції з різними підмасивами тощо. Це забезпечує зручний та гнучкий спосіб маніпулювання даними.

NumPy також володіє функціями для роботи з файлами, включаючи збереження та завантаження масивів у різних форматах, таких як текстові файли, бінарні файли, формат NumPy та інші. Це дозволяє зручно обмінюватись даними між різними програмами та платформами.

Крім того, NumPy може бути легко інтегрована з іншими науковими бібліотеками, такими як SciPy, Matplotlib, Pandas тощо. Це дозволяє розширити функціональність та можливості аналізу даних та наукових обчислень.

У загальному, NumPy є потужним інструментом для роботи з числовими даними в мові програмування Python. Вона забезпечує широкий набір функцій для роботи з масивами, включаючи обчислення, маніпулювання, індексацію, зрізи та інші операції. Завдяки своїй ефективності та широкому спектру функціональності, NumPy є незамінним інструментом для наукових обчислень, обробки даних та розв'язання складних математичних задач.

2.10 Система реального часу для виявлення силуетів людей

Система реального часу для виявлення силуетів людей - це програмний комплекс, який використовується для автоматичного виявлення силуетів людей на зображеннях або в потокових відеоданих в реальному часі. Така система має широкі застосування в областях комп'ютерного зору, безпеки, робототехніки, моніторингу та багатьох інших.

Основні компоненти системи реального часу для виявлення силуетів людей

включають в себе алгоритми обробки зображень та комп'ютерного зору, які дозволяють виділити області, що відповідають силуетам людей, і відрізнити їх від інших об'єктів на зображенні. Для цього можуть використовуватись різні методи та

ст. 30

алгоритми, такі як віднімання фону, машинне навчання, сегментація зображень та інші.

Одним з головних викликів у системі реального часу є забезпечення високої швидкодії обробки даних, оскільки потрібно виявляти та відстежувати силуети людей в реальному часі. Це досягається застосуванням оптимізованих алгоритмів, паралельного обчислення, апаратної підтримки (наприклад, застосування GPU) та інших технік для прискорення обробки даних.

Після виявлення силуетів система може виконувати додатковий аналіз та обробку, такі як визначення позиції та руху людей, виявлення жестів, класифікацію або відстеження об'єктів. Це дозволяє використовувати систему для багатьох застосувань, включаючи безпеку, відеоспостереження, взаємодію з роботами та інше. Система реального часу для виявлення силуетів людей є потужним інструментом, що дозволяє автоматизувати процес виявлення людей на зображеннях або в потокових відеоданих в реальному часі. Вона забезпечує швидку та ефективну обробку даних, що дозволяє використовувати її в різних сферах застосування.

ст. 31

РОЗДІЛ 3 ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

3.1 Опис експериментального середовища

У рамках дипломної роботи були використані наступні програмні забезпечення та бібліотеки:

1. PyCharm є інтегрованою середовищем розробки (IDE) для мови програмування Python. Це потужний інструмент, який надає зручне середовище для написання, налагодження та виконання коду Python. Він має багато функціональних можливостей, таких як автодоповнення, перевірка синтаксису, візуальне налагодження та багато іншого.
2. Python - це потужна та популярна мова програмування, яка широко використовується для наукових обчислень, обробки даних та розробки алгоритмів машинного навчання. Вона має простий та зрозумілий синтаксис, багатий вибір бібліотек та модулів, які полегшують роботу з обробкою зображень та машинним навчанням.
3. OpenCV (Open Source Computer Vision Library) є бібліотекою з відкритим вихідним кодом для комп'ютерного зору та обробки зображень. Вона надає набір функцій та алгоритмів для обробки зображень, виявлення об'єктів, сегментації, відстеження руху та багато іншого. OpenCV є потужним інструментом для виконання завдань з обробки зображень у дипломній роботі.

4. NumPy - це основна бібліотека для наукових обчислень у Python. Вона надає підтримку для масивів та математичних операцій над ними. NumPy дозволяє

ефективно працювати з багатовимірними масивами даних, що використовуються в задачах обробки зображень та аналізу даних.

5. Caffe (Convolutional Architecture for Fast Feature Embedding) є фреймворком глибокого навчання, спеціалізованим на обробці зображень. Він надає можливість побудови та тренування нейронних мереж для розпізнавання об'єктів, виявлення силуетів та інших задач комп'ютерного зору.

ст. 32

Опис використаних технічних засобів (комп'ютер, операційна система, процесор, пам'ять тощо):

Ім'я пристрою - ClimpPC

Процесор - AMD Ryzen 5 2600X Six-Core Processor 3.60 GHz

Відеокарта – AMD RX590 Nitro+

ОЗП - 16,0 ГБ

Тип системи - 64-розрядна операційна система, процесор на базі архітектури x64

Операційна система – Windows 11 Pro

Версія - 22H2

Збірка ОС - 22631.1830

Опис використаних наборів даних та зображень:

В рамках дипломної роботи для побудови та тренування моделей виявлення силуетів за допомогою Caffe було використано набір даних, що складався з 5 зображень зображень, що містять людей. Описані зображення використовувалися для створення набору даних для тренування та перевірки моделі.

Кожне зображення містить сцену з людьми у різних позах та умовах освітлення. Вони можуть включати людей, що рухаються, перебувають у різних позах тіла, а також перекриті іншими об'єктами. Зображення були відібрані з урахуванням різноманітності сценаріїв та складності умов, щоб використовувати їх для вивчення та аналізу методів виявлення силуетів.

Ці зображення були попередньо оброблені для видалення непотрібної інформації та підготовки для використання в Caffè. Крім того, були створені анотації, які вказували контур силуетів людей на кожному зображенні. Ці анотації використовувалися як правильні мітки для тренування та оцінки ефективності моделей.

ст. 33

Використання обмеженого набору даних зумовлено обмеженостями доступу до великих та розширених наборів даних, а також обмеженнями ресурсів. Однак, цей набір даних достатньою мірою дозволяє провести експерименти та аналізувати результати для виявлення силуетів на зображеннях у контексті дипломної роботи.

3.2 Результати досліджень виявлення силуетів на зображеннях

Було проведено дослідження на основі 5 зображень, що містили людей. Кожне з цих зображень було використано для тестування розробленої моделі виявлення силуетів.

Результати дослідження показали хороші результати для більшості зображень, за винятком тих, де були присутні значні зміни освітлення. Зображення зі змінним освітленням можуть створювати труднощі для точного виявлення силуетів, оскільки зміни освітлення можуть змінювати контрастність та яскравість силуетів.

У таких випадках результати моделі можуть бути менш точними, і можуть виникати проблеми з виявленням чітких контурів силуетів. Деякі деталі можуть бути втрачені або спотворені.

Це свідчить про важливість подальшого дослідження та розробки методів, що враховують змінне освітлення для поліпшення точності виявлення силуетів. Це може включати застосування спеціальних алгоритмів обробки зображень, технік адаптації до умов освітлення або використання додаткових даних для тренування моделей.

У загальному, хоча зображення з змінним освітленням можуть представляти виклики для виявлення силуетів, результати на інших зображеннях свідчать про ефективність розробленої моделі.

Проте, слід зауважити, що робота з обмеженим набором даних може мати вплив на загальну універсальність та надійність моделі. Для підтвердження та розширення отриманих результатів, рекомендується використовувати більш розгорнуті набори даних та проводити експерименти на широкому спектрі зображень з різними умовами освітлення, позами тіла та перекриттями об'єктів.

ст. 34

Рисунок 3.1 Результат виконання дослідження зображення No1

Рисунок 3.2 Результат виконання дослідження зображення No2

ст. 35

Рисунок 3.3 Результат виконання дослідження зображення No3

Рисунок 3.4 Результат виконання дослідження зображення No4

ст. 36

Рисунок 3.5 Результат виконання дослідження зображення No5

3.3 Результати досліджень виявлення силуетів у режимі реального часу

У рамках дипломної роботи було проведено дослідження на основі 4 зображень, що містили об'єкти, включаючи людей. Кожне з цих зображень було використано для тестування розробленої моделі виявлення силуетів у режимі реального часу.

Дослідження показали хороші результати для кожного експерименту, незважаючи на наявність перешкод для об'єкта. Навіть у випадках, коли силует був перекритий іншими об'єктами, модель успішно виявила та відтворила контури об'єкта. Важливо зазначити, що при наявності перешкод точність виявлення силуетів може знизитися, але модель продемонструвала здатність до розпізнавання об'єктів, навіть за таких умов.

Ці результати свідчать про ефективність розробленої моделі виявлення силуетів

у режимі реального часу та її здатність до роботи зі складними сценаріями, де об'єкти можуть бути перекриті іншими об'єктами. Вони підкреслюють потенціал

ст. 37

застосування моделі для виявлення силуетів на зображеннях з реального світу, де перешкоди можуть відігравати роль.

Проте, слід зазначити, що дослідження проводилося на обмеженій кількості зображень, тому рекомендується подальше дослідження з використанням більш розгорнутих наборів даних та більш різноманітних сценаріїв, щоб підтвердити та уточнити отримані результати.

Рисунок 3.6 Результат виконання дослідження силуету людини в режимі реального часу No1

ст. 38

Рисунок 3.7 Результат виконання дослідження силуету **людини в режимі реального часу** No2

Рисунок 3.8 Результат виконання дослідження силуету **людини в режимі реального часу** No3

ст. 39

Рисунок 3.9 Результат виконання дослідження силуету людини в режимі реального часу No4

ст. 40

ВИСНОВКИ

У результаті проведеного дослідження виявлення силуетів на зображеннях для дипломної роботи, були отримані наступні висновки:

Виявлення силуетів на зображеннях є важливою задачею в області комп'ютерного зору та обробки зображень. Це має широкий спектр застосувань, включаючи відеоспостереження, розпізнавання людей, аналіз руху та багато іншого. Зміна освітлення та перекриття об'єктів можуть становити виклики для точного виявлення силуетів. Зміни в освітленні можуть впливати на контрастність та яскравість силуетів, тоді як перекриття можуть призводити до втрати деталей та спотворення контурів.

Розроблені методи та алгоритми, що враховують змінне освітлення та перекриття об'єктів, можуть покращити точність виявлення силуетів. Такі методи включають адаптацію до умов освітлення, застосування фільтрів та алгоритмів сегментації, які дозволяють виявляти силуети незалежно від позиції тіла та перешкод.

Використання спеціалізованих програмних засобів та бібліотек, таких як PyCharm, Python, OpenCV, NumPy і Caffe, сприяє розробці та реалізації алгоритмів виявлення силуетів. Ці інструменти надають різноманітні функції для обробки зображень, навчання нейронних мереж та виконання експериментів.

Дослідження на обмеженій кількості зображень показало перспективні результати виявлення силуетів. Було показано, що розроблена модель успішно впорається зі штучно створеними зображеннями та зображеннями, що містять перешкоди. Проте, для підтвердження цих результатів та розширення області застосування необхідно провести додаткові дослідження з більш розгорнутими наборами даних та більш різноманітними сценаріями.

Загалом, робота над виявленням силуетів на зображеннях має великий потенціал і може бути використана в різних практичних застосуваннях. Подальші дослідження та розвиток методів виявлення силуетів можуть сприяти поліпшенню точності та надійності цих систем.

ст. 41

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Imutils : веб-сайт. URL: <https://github.com/PylImageSearch/imutils>
2. MobileNet version 2 : веб-сайт. URL: <https://machinethink.net/blog/mobilenet-v2/>
3. General Python FAQ : веб-сайт. URL: <https://docs.python.org/3/faq/general.html>
4. What is MobileNet SSD v2? : веб-сайт. URL: <https://roboflow.com/model/mobilenet-ssd-v2>
5. Convolutional neural network : веб-сайт. URL: https://en.wikipedia.org/wiki/Convolutional_neural_network

6. What is OpenCV? The Complete Guide (2023) : веб-сайт. URL: <https://viso.ai/computer-vision/opencv/>
7. Mastering OpenCV with Python: A Comprehensive Guide for Image Processing and Computer Vision : веб-сайт. URL: <https://levelup.gitconnected.com/mastering-opencv-with-python-a-comprehensive-guide-for-image-processing-and-computer-vision-732d457ed38>
8. Deep-Learning Using Caffe Model : веб-сайт. URL: <https://www.scilab.org/deep-learning-using-caffe-model>
9. What is Caffe2? : веб-сайт. URL: <https://caffe2.ai/docs/caffe-migration.html>
10. Blob : веб-сайт. URL: <https://developer.mozilla.org/ru/docs/Web/API/Blob>