**Міністерство освіти і науки України**

**Національний технічний університет України «КПІ» імені Ігоря Сікорського**

**Кафедра обчислювальної техніки ФІОТ**

**ЗВІТ**

**з виконання завдань заліку**

**з навчальної дисципліни «Технології Computer Vision»**

**Виконав:**

Студент 3 курсу кафедри ІПІ ФІОТ,

Навчальної групи ІП-11

Лошак В.І.

**Перевірив:**

Професор кафедри ОТ ФІОТ

Писарчук О.О.

**Київ 2024**

**І. Білет\_15**

**ІІ. Завдання:**

1. Алгоритми растеризація векторних зображень.

2. Типи фільтрів для цифрових зображень та їх математичні моделі.

3. Розробити програмний скрипт що забезпечує порівняння двох цифрових зображень. Цифрові зображення обрати самостійно.

**ІІІ. Результати виконання модульної контрольної роботи.**

***3.1. Відповідь на теоретичне питання No1.: Алгоритми растеризація векторних зображень.***

Растеризація векторних зображень—це процес перетворення векторних графічних даних, які описують зображення за допомогою геометричних примітивів, таких як точки, лінії, криві та полігони, у піксельний (растровий) формат.

В лабораторних що я виконував протягом цього курсу, застосовувались наступні алгоритми:

1. **Алгоритм Брезенхема для ліній** — використовується для растеризації прямих ліній. Він вирішує, який найближчий піксель до ідеальної лінії слід забарвити, не використовуючи при цьому операції з плаваючою точкою(щоб це було ефективно в плані кількості обчислень).

Основний нахил лінії mmm можна виразити як:

Для спрощення і уникнення операцій з плаваючою комою, алгоритм Брезенхема використовує термін помилки або параметр рішення delta, який коригується на кожному кроці, щоб визначити, чи буде наступний піксель прямо праворуч або праворуч і вгору (або праворуч і вниз, залежно від нахилу лінії).

1. **Алгоритм растеризації кола Брезенхема** це модифікація алгоритму лінії Брезенхема для генерації кола. Він використовує ті ж основні принципи — визначає найближчі пікселі до ідеальної кривої без використання плаваючої точки.
2. **Алгоритм растеризації кривих Безьє. Сам алгоритм** використовуються для гладкого опису кривих шляхів у векторній графіці, і вони базуються на параметричних поліноміальних функціях. В математичному сенсі, крива Безьє задається як лінійна комбінація контрольних точок з використанням поліноміальних базисних функцій, відомих як поліноми Бернштейна.

**Крива безь’є n-ного ступеня визначається як:**

Процес растеризації кривих Безьє включає визначення пікселів на кривій, які повинні бути закрашені на екрані. Це вимагає дискретизації ttt і обчислення відповідних точок на кривій. Алгоритми растеризації можуть використовувати рекурсивне поділ кривої на менші сегменти або адаптивне уточнення, щоб досягти більш високої точності та якості візуалізації.

**Є ще багато інших алгоритмів що використовуються для растеризації, наприклад Алгоритм Ву,** Алгоритм заливки по наскрізному перетину, і т.д. Але оскільки я ніколи з ними не працював, то не вважаю доцільним тут їх висвітлювати.

*Застосування фільтрів* — Використовується щоб позбавитись або додати якісь риси до зображення. Існує багато типів фільтрів. Назву лише ті що я застосовував в лабораторних протягом дисципліни:

* Фільтр Canny для виявлення контурів на зображеннях, використовує градієнти інтенсивності.
* Медіанний фільтр використовував для видалення шумів. Особливо ефективний при видаленні шуму типу «сіль і перець»
* Гаусівський фільтр для згладжування зображень, зменшення шуму на зображенні, ефективний щоб зменшити кількість дрібних деталей.
* Фільтр Собеля також призначений для виділення країв. Використовує дві перпендикулярні маски (одна для горизонтальних, інша для вертикальних країв).

Існує ще багато інших фільтрів які я тут не назвав, але це основні з тих що я згадав

Це були основні методи покращення зображень що я використовував в цьому курсі.

***3.2. Відповідь на теоретичне питання No2: Типи фільтрів для цифрових зображень та їх математичні моделі.***

Фільтри використовується щоб позбавитись або додати якісь риси до зображення. Існує багато типів фільтрів. Назву лише ті що я застосовував в лабораторних протягом дисципліни:

1. Середньозважений фільтр (Averaging Filter) згладжує зображення, замінюючи значення кожного пікселя середнім значенням пікселів у його околиці.
2. Гауссівський фільтр використовує гауссове ядро для згладжування зображення, ефективно зменшуючи шум.
3. Медіанний фільтр замінює кожен піксель медіанним значенням у визначеній околиці, ефективно усуваючи "сіль і перець" шум.
4. Фільтр Собеля використовує два окремі ядра(kernel) для виявлення горизонтальних і вертикальних країв.

A number and numbers in a row

Description automatically generated with medium confidence A number and numbers in a row

Description automatically generated with medium confidence

Рис 1, 2. — Ядра для виявлення горизонтальних та вертикальних країв відповідно.

Величина градієнта обчислюється як:

1. Фільтр Лапласа використовує другу похідну зображення для виявлення країв, шукаючи різкі перепади яскравості що зазвичай характеризуються високими значеннями похідної яскравості.

Ядро фільтра Лапласа має вигляд:

A number and numbers in a row

Description automatically generated with medium confidence

Рис 3. —Ядро фільтра Лапласа

1. Фільтр Canny є комбінацією та певним вдосконаленням над гаусівським та собелевим фільтрами. Він використовується для визначення країв на зображенні. Спочатку він застосовує ці два фільтри у вказаному порядку. Потім кожен піксель перевіряється на те, чи є він максимальним у своєму напрямку градієнту, що забезпечує тонкі краї. Потім використовуються два пороги: низький та високий. Всі крайні пікселі, що перевищують високий поріг, вважаються справжніми краями, пікселі нижче низького порогу відкидаються, а ті, що знаходяться між порогами, вважаються краями лише якщо вони з'єднані з справжніми краями.

Існує ще багато інших фільтрів які я тут не назвав, але це основні з тих що я згадав

***3.3. Відповідь на практичне питання No3: Розробити програмний скрипт що забезпечує порівняння двох цифрових зображень. Цифрові зображення обрати самостійно.***

Індекс структурної подібності (SSIM) — це метод вимірювання подібності між двома зображеннями. Він був розроблений для вдосконалення традиційних методів, таких як середня квадратична помилка (MSE) і пікове співвідношення сигнал/шум (PSNR), які можуть дати оманливі результати, оскільки не враховують візуальне сприйняття.

**Математична модель:**

Індекс SSIM є повною еталонною метрикою; іншими словами, для вимірювання подібності потрібні два зображення: одне з яких є вихідним зображенням, а інше погіршеним зображенням. SSIM розраховується для різних вікнон зображення. Метрика для двох вікон х і y розміру N×N дорівнює:

Насправді ця формула є комбінацією(добутком) декількох інших метрик подібності.

1. Яскравість: Цей компонент вимірює близькість яскравості зображення для двох зображень.
2. Контраст: цей компонент порівнює контраст двох зображень.
3. Структура: вимірює коефіцієнт кореляції між двома зображеннями.

Скрипт що виконує порівняння схожості зображень використовує метрику SSIM. Чим більше показник тим більш схожими є зображення.

A collage of cars

Description automatically generated

Рис 4. — Зображення для порівняння. Були згенеровані за допомогою моделі stable diffusion з використанням запиту: dolorian

Код що виконує порівняння та результат порівняння:

A screen shot of a computer code

Description automatically generated

Перевіримо результати при порівнянні відмінних зображень:

A plane flying in the sky

Description automatically generated

Рис 5. — Порівняння відмінних зображень

Результат порівняння:

A screen shot of a computer code

Description automatically generated

Як видно з результатів порівняння, ми можемо встановити confidence\_interval для порівняння схожості зображень з певною точністю. Результат роботи успішний.