

Сторінка 0: None

Сторінка 1: Тема моєї бакалаврської дипломної роботи – ‘Математичне та програмне забезпечення чат боту з покращення якості зображення на межах кольорів’

Сторінка 2: Щодо актуальності даної теми, думаю, ніхто не буде заперечувати, що кожного з нас є фотографії. І було б непогано мати інструмент для покращення якості цих фото.

Сторінка 3: Метою цього дослідження є покращення якості зображень на межах кольорів за допомогою відповідних алгоритмів машинного навчання. Покращення деталізації та різкості на межах кольорових переходів. Мінімальне збільшення PSNR має бути не менше за 0.3 dB.

Сторінка 4: Були розглянуті різні альтернативи інтерфейсу для взаємодії з користувачем. Я обрав чат бота, оскільки це сучасне, зручне та доступне рішення

Сторінка 5: Огляд існуючих комерційних рішень покращення зображень та їх функціоналу.

Сторінка 6: На слайді наведена порівняльна таблиця з 5-ти комерційних рішень. Тут варто звернути увагу, на 2 критерії – це вартість, тобто кожне рішення має платні інструменти, що не дуже добре, якщо дивитись із позиції користувача, і те що жодне рішення не створено у вигляді чат боту. Стосовно цього, можу сказати лише те, що кожна платформа орієнтовна на свою вікову групу, на свого користувача, і вгодити кожному можна лише створивши застосунок для всіх платформ.

Сторінка 7: Огляд обраних алгоритмів машинного навчання

Сторінка 8: Основний вибраний мною алгоритм – це згорткова нейронна мережа. Суть її в тому, що ми маємо зображення, яке представляємо у вигляді матриці, де кожен елемент є пікселем. І накладаємо на цю матрицю фільтр, який поступово рухаємо з певним кроком для того, щоб витягати певні ознаки. Ці ознаки пропускаємо через функцію активації, в нашому випадку це ReLU для розпізнавання деталей.

Сторінка 9: Також використовується залишково-щільний блок, який має шари згортки та функцію активації ReLU. Але тут кожен шар отримує вхід не тільки від попереднього шару, але й від усіх попередніх шарів блоку. Це допомагає уникнути проблеми згасання градієнта, сприяючи кращій передачі інформації через мережу та дозволяє ефективно тренувати глибокі нейронні мережі.

Сторінка 10: Функція активації в нейронній мережі визначає вихід нейрона на основі його вхідних даних. Вона додає нелінійність до моделі

ReLU було обрано, тому що функція простіша в обчислюванні, від того швидша та ще допомагає уникнути проблеми згасання градієнту.

Для додатнього входу похідна рівна 1, тому градієнт не згасає, а є постійним.

Сторінка 11: Архітектура моделі складається з комбінації згорткових шарів та залишково-щільних блоків.

Сторінка 12: Набір даних

Сторінка 13: У роботі використано частину з великого набору даних ImageNet, а саме, ту частину яка викладена на платформі Kaggle. Кількість зображень можете побачити на цьому графіку.

Сторінка 14: Щоб нормалізувати дані до діапазону [0; 1] ми представляємо зображення у вигляді матриці і ділимо кожен елемент на 255.

Сторінка 15: Для навчання моделі було відібрано зображення, які мають 3 канали кольору, ширину від 500 до 800 пікселів та висоту від 300 до 800 пікселів.

Сторінка 16: Передобробка являє собою перетворення зображення з RGB в YUV та виділення лише каналу яскравості Y. Це відбувається за такими формулами.

Сторінка 17: None

Сторінка 18: Модель була навчена на такій кількості даних.

Сторінка 19: Зменшення навчального набору даних викликано обмеженнями в обчислювальних потужностях. Локально модель навчалась дуже довго. Тому було прийнято рішення навчати в хмарному середовищі Google Colab. Використовуючи відеокарту Nvidia Tesla T4

Сторінка 20: None

Сторінка 21:

$$PSNR = 20 * \log_{10}(\frac{MAX_I}{\sqrt{MSE}})$$

де MAX_I – максимальне можливе значення пікселя, а MSE – середньоквадратична помилка.

Вищий $PSNR$ [17] означає краще покращення, оскільки вказує на вище відношення сигналу до шуму.

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (I(i, j) - K(i, j))^2 \quad (4.1.2)$$

де $I(i, j)$ – значення пікселя в позиції (i, j) оригінального зображення, $K(i, j)$ – значення пікселя в позиції (i, j) покращеного зображення, а M і N – розміри зображення.

Нижче значення MSE означає, що покращене зображення більш схоже на оригінальне, оскільки показує меншу квадратну помилку між ними.

$$MAE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |I(i, j) - K(i, j)|$$

де $I(i, j)$ – значення пікселя в позиції (i, j) оригінального зображення, $K(i, j)$ – значення пікселя в позиції (i, j) покращеного зображення, а M і N – розміри зображення.

Чим нижче значення MAE тим краще результат, оскільки це означає меншу різницю між оригінальним та покращеним зображеннями.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

де μ_x і μ_y – середні значення пікселів у вікнах x та y , σ_x^2 і σ_y^2 – дисперсії в цих вікнах, σ_{xy} – коваріація між x та y , а C_1 та C_2 – стабілізуючі константи для уникнення ділення на нуль.

Чим вищий SSIM [18], тим краще результат, оскільки це вказує на вищу структурну подібність між оригінальним та покращеним зображеннями.

Сторінка 22: Далі йдуть декілька слайдів із результатами. Пропоную не зупинятись на них довго. Основна ідея – це з оригінального зображення зробити погіршене, а потім це погіршене спробувати покращити передавши на вхід до моделі. Зверніть увагу на контури об'єкта

Сторінка 23: Це результат роботи на старих фотографіях

Сторінка 24: Результат роботи на рентгенівських знімках

Сторінка 25: Результат роботи на текстових елементах

Сторінка 26: На слайді представлено як виглядає інтерфейс чат боту. Також було знайдено середнє значення PSNR на тестовому набору даних і різниця склала 0.647 dB, що задовольняє поставлену мету.

Сторінка 27: Висновки

Сторінка 28: *Зачитати висновки*

Сторінка 29: На цьому в мене все, готовий відповісти на ваші запитання.