РЕФЕРАТ

Дипломна робота: 75 сторінок, 37 рисунків, 14 таблиць, 14 джерел.

Ціль роботи – захист роботи нейронних мереж від зовнішнього впливу у вигляді спотворень за допомогою фільтрації.

В роботі приведені існуючі методи оцінювання візуальної якості зображень та їх реалізація. Представлені кілька моделей шуму. Розглянуто та реалізовано згорткову нейронну мережу для класифікації об’єктів на зображеннях. Проведено класифікація тестових зображень з шумом та після фільтрації. Проведено оцінку якості фільтрації за допомогою метрик.

МЕТРИКИ ВІЗУАЛЬНОЇ ЯКОСТІ, ШУМИ НА ЗОБРАЖЕННЯХ ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, ФІЛЬТРАЦІЯ ЗОБРАЖЕНЬ.

ABSTRACT

Thesis: 75 pages, 37 figures, 14 tables, 14 sources.

The purpose of the work - to protect the neural networks from external influences in the form of distortion through filtering.

In robotic methods of evaluation of visual images and their implementation. Presented several models of noise. The neural network for images is displayed and converted. The classification of tests of images with noise and after filtering is carried out. The filtering rate was determined using metrics.

VISUAL QUALITY METRICS, IMAGE NOISE ROLLED NEURAL NETWORKS, IMAGE FILTRATION,

Зміст

ВСТУП 6

1 НЕЙРОННІ МЕРЕЖІ 7

1.1.2 Дата сет Coco (Common Objects in Context) 8

1.2 Спотворення і завади 8

1.2.4 Ерланг (гамма) шум 8

1.3 Критерії ефективності придушення перешкод 8

1.3.4 DSSIM (структурна несхожість) 8

2 ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДІВ ОБРОБКИ ЗОБРАЖЕНЬ 10

2.1 Завади 10

2.1.1 Реалізація моделі шуму AWGN 10

3 РЕЗУЛЬТАТИ КЛАСИФІКАЦІЇ 11

4 ЕКОНОМІЧНА ЧАСТИНА 12

4.1 Розрахунок витрат на проведення дослідження 12

Висновки 18

ПЕРЕЛІК ПОСИЛАНЬ 20

# ВСТУП

Однією з фундаментальних проблем сучасності є проблема зорового сприйняття. Виникнувши дуже давно, вона є актуальною і нині, бо зображення є природним засобом спілкування людини і машини в будь-яких системах обробки, аналізу і контролю. Комп'ютерна графіка поділяється на три основних напрямки: візуалізація, обробка зображень і розпізнавання образів. Візуалізація- це створення зображення на основі якогось опису (моделі). Основне завдання розпізнавання образів - отримання семантичного опису зображених об'єктів. Обробка зображень відповідає за перетворення (фільтрацію) зображень. Розвиток сучасних засобів комп'ютерної техніки та інформаційних технологій сприяє широкому впровадженню в практику систем автоматичної обробки зображень. Першочерговим завданням такої системи є поліпшення якості зображення. Проблема шумозаглушення є однією з найактуальніших і поширених проблем в області обробки зображень.

Для визначення і оцінювання візуальної якості зображень застосовуються спеціальні методи і критерії - метрики. За їх величинам можна поставити діагноз візуальної якості зображення, мати поняття про параметри і походження шуму в них, а також яким чином їх можна поліпшити і домогтися кращої якості в порівнянні з вихідним. Метрики поділяються на ті, які залежать від сприйняття людини, і на ті що розраховуються автоматично. Процедура поліпшення якості зображень зводиться до виконання комплексу заходів для поліпшення візуального сприйняття зображення, або перетворення його в більш зручну форму для подальшого аналізу.

# 1 НЕЙРОННІ МЕРЕЖІ

Згорткова нейронна мережа (також CNN або ConvNet) (рис. 1.1) є одним з найбільш популярних алгоритмів в глибокому навчанні, це такий вид машинного навчання, при якому модель вчиться виконувати завдання класифікації безпосередньо на зображенні, відео, тексті або звуці.

Згорткові нейронні мережі особливо корисні для пошуку шаблонів (патернів) на зображеннях для розпізнавання об'єктів, осіб і сцен. Вони вчяться безпосередньо на зображеннях, використовуючи шаблони для класифікації зображень і усуваючи потребу в ручному витяганні ознак.

Додатки, що вимагають розпізнавання об'єктів і комп'ютерного зору, такі як безпілотні автомобілі і додатки для розпізнавання осіб, багато в чому залежать від CNN. Залежно від вашої програми ви можете побудувати свою CNN з нуля або використовувати готову модель на ваших даних.

* Що робить згорткові нейронні мережі такими корисними? Використання CNN для глибокого навчання стало популярнішим завдяки трьом важливим факторам:
* CNN усувають необхідність ручного вилучення ознак - CNN сама витягує ознаки.
* CNN видають найсучасніші результати розпізнавання.
* CNN можуть бути перенавчитися для виконання нових завдань розпізнавання, що дозволить вам використовувати існуючі мережі.



Рисунок 1.1 – Робота згорткової мережі

* 1. Класифікація на основі нейромереж

Класифікація є однією з найважливіших задач інтелектуального аналізу даних. Вона вирішується за допомогою аналітичних моделей, званих класифікаторами. Затребуваність класифікації обумовлена ​​порівняльною простотою алгоритмів і методів її реалізації, і високою інтерпретованістю результатів в порівнянні з іншими технологіями аналізу даних.

### 1.1.2 Дата сет Coco (Common Objects in Context)

# 1.2 Спотворення і завади

### 1.2.4 Ерланг (гамма) шум

 (1.1)

Середнє значення та дисперсія цієї щільності наведені як

 (1.2)

# 1.3 Критерії ефективності придушення перешкод

### 1.3.4 DSSIM (структурна несхожість)

Існує ще одна метрика відстані, яка називається структурною несхожістю (DSSIM), виведена із структурної подібності (SSIM), і може бути виражена як:

 (1.9)

Метод індексу SSIM, показник вимірювання якості, обчислюється на основі обчислення трьох основних аспектів, які називаються яскравістю, контрастом та структурним терміном або терміном кореляції. Цей індекс є поєднанням множення цих трьох аспектів.

Метод індексу структурної подібності можна виразити за допомогою цих трьох термінів як:

 (1.10)

* 1. Фільтри
		1. BM3D

# 2 ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДІВ ОБРОБКИ ЗОБРАЖЕНЬ

## 2.1 Завади

### 2.1.1 Реалізація моделі шуму AWGN

 На рис. 2.1 показано лістинг функції додавання до зображення адитивного білого гауссового шуму. Приклад роботи показано на рис. 2.2.



Рисунок 2.1 – Реалізація AWGN



Рисунок 2.2 – Приклад роботи AWGN

# 3 РЕЗУЛЬТАТИ КЛАСИФІКАЦІЇ

 Класифікація зображень проводилася за допомогою нейромережі YOLOv5 та моделі YOLOv5s. Зображення були взяті з датасету Coco. Були використані метрики PSNR, FSIM, PSNR-HVS-M, IoU та відсоток правильно класифікованих зображень. На гістограмах (рис.3.2, 3.4, 3.6, 3.8) показано відносну зміну значення кожної з метрик.

# 4 ЕКОНОМІЧНА ЧАСТИНА

 У даному розділі необхідно скласти перелік етапів роботи, які необхідні при розробці та розрахувати трудомісткість та собівартість продукту. Це дозволить визначити оптимальний організаційно-технологічний варіант виконання роботи.

##  4.1 Розрахунок витрат на проведення дослідження

Роботу буде виконувати один студент та використовувати для розробки комп’ютер з середніми показниками ефективності. Розрахуємо середню часову ставку виконавця роботи: заробітна плата 37397 грн./місяць. Враховуємо, що молодший системний інженер працює по 8 годин у день 5 днів у тиждень, тобто 176 годин в місяць. Таким чином, можна розрахувати погодинний оклад працівника. Формула для розрахунку:

 $L=\frac{Р}{Н}$, (4.1)

де $L$– погодинний оклад робітника, грн/год;

$P$ – місячна заробітна плата робітника, грн;

$Н$– кількість відпрацьованих годин.

Спираючись на формулу (4.1), розрахуємо погодинний оклад працівника:

$$L=\frac{37397}{176}=212.48(грн)$$

Додаткова зарплата у даному випадку встановлена на рівні 20% від основної. Розрахунок додаткової заробітної плати за 1 годину:

$L\_{ADD}=L⋅К\_{зпдод,}(4.2)$

,де Кзпдод – коефіцієнт додаткової заробітної плати.

$$L\_{ADD}=212.48∙0,2=42,5(грн)$$

 Опишемо перелік робіт по проекту для визначення загальної кількості годин необхідних на виконання дослідження – див. табл 4.1.

Таблиця 4.1 – Перелік робіт по дослідженню та трудомісткість

|  |  |  |
| --- | --- | --- |
| Номер етапу | Назва стадій та етапів | Трудомісткість, люд./год. |
| І | Аналіз технічного завдання | 50 |
| ІІ | Розробка веб-додатку  | 240 |
| ІІІ | Тестування веб-додатку | 40 |
| Разом: | 320 |

Відповідно до табл. 4.1, трудомісткість дослідження складає 320
люд. /год. Таким чином, повна заробітна плата виконавця роботи (Lповна) розраховується по формулі (4.3):

 $L\_{повна}=L⋅T+L\_{ADD}⋅T,$ (4.3)

де T – кількість відпрацьованих годин.

$$L\_{повна}=212.48⋅320+42,45⋅320=81577.59(грн.)$$

Нарахування на заробітну плату (єдиний соціальний внесок складає 22%) та розраховується за формулою:

 $L\_{соц}=L\_{повна}∙Н\_{соц}$,(4.4)

де Нсоц  = 22%.

$$L\_{соц}=81577.59∙0,22=17947.0698(грн.)$$

 Для підготовки необхідних документів і звітів тестування використовувалися різні канцтовари. Перелік і вартість канцелярських інструментів і витратних матеріалів наведено в таблиці 4.2.

Таблиця 4.2 – Вартість канцтоварів

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Найменування | Кіл-сть, шт | Ціна,грн | Сума,грн |
| 1 | Тека для паперів  | 1 | 14 | 14 |
| 2 | Канцтовари у асортименті | 1 | 85 | 85 |
| 3 | Упаковка матових файлів формату А4 (100 шт/уп) | 1 | 42 | 42 |
| 4 | Степлер  | 1 | 18 | 18 |
| 5 | Бумага формату А4 упаковка | 1 | 100 | 100 |
| Разом: | 259 |

Вартість канцтоварів для оформлення, зберігання проектної документації склала 259 грн.

В рамках виконання проекту треба використати обладнання та програмне забезпечення. Перелік і вартість цього обладнання та ліцензійного забезпечення наведено в таблиці 4.3

Таблиця 4.3 – Розрахунок ціни обладнання та програмного забезпечення.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Найменування | Кіл-сть, шт | Ціна,грн | Сума,грн |
| 1 | Стаціонарний комп’ютер | 1 | 13179 | 13179 |
| 2 | Програмне забезпечення Pycharm  | 1 | 5 449,56 | 5 449,56 |
| Разом: | 18628.56 |

Річна сума амортизаційних відрахувань приведена у табл. 4.4.

Таблиця 4.4 – Розрахунок річної суми амортизаційних відрахувань.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № п/п | Елементи основних фондів | Кіл-сть | Вартість, грн. | Сума грн. | Норма амортизації, % | Річні амортизаційні відрахування, грн. |
| 1 | Стаціонарний комп’ютер | 1 | 13179 | 13179 | 20% | 2635.8 |

Розрахуємо величини амортизаційних відрахувань з розрахунку на одну годину роботи по кожному виду обладнання. Формула для розрахунку:

 $А\_{H}=\frac{A\_{T}}{T\_{Y}}$, (4.5)

,де $T\_{Y}$– кількість робочих годин за рік ‒ 1993 години у 2021 році.

 Величини амортизаційних відрахувань з розрахунку на одну годину роботи обладнання наведені у табл. 4.5.

Таблиця 4.5 – Розрахунок величини амортизаційних відрахувань з розрахунку на одну годину роботи обладнання.

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Елементи основних фондів | Річні амортизаційні відрахування, грн. | Амортизаційні відрахування за одну годину роботи, грн. |
| 1 | Стаціонарний комп’ютер | 2635.8 | 1,32 |

Спираючись на розрахунки з табл. 4.5, отримаємо суму амортизації по кожному виду обладнання відносно кількості відпрацьованих годин. Формула для розрахунку суми амортизації:

 $A\_{F}=A\_{H}⋅T$, (4.6)

,де $T$‒ кількість відпрацьованого обладнанням часу, у год.

Розрахуємо суму амортизації обладнання виходячи з кількості відпрацьованих годин за формулою (4.6). Результати приведені у табл. 4.6

Таблиця 4.6 – Розрахунок величини амортизаційних відрахувань з розрахунку кількості відпрацьованих годин.

Таблиця 4.6 – Собівартість розробки додатку за весь час

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Елементи основних фондів | Амортизаційні відрахування за одну годину роботи, грн. | Кількість відпрацьованого часу, год. | Амортизаційні відрахування за відпрацьований час, грн. |
| 1 | Стаціонарний комп’ютер | 1,32 | 320 | 422.4 |

В результаті обліку всіх перерахованих вище статей витрат можна визначити собівартість розробки додатку (таблиця 4.7).

Таблиця 4.7 – Собівартість розробки додатку

|  |  |  |
| --- | --- | --- |
| № п/п | Стаття калькуляції | Сума,грн |
| 1 | Амортизація | 422.4 |
| 2 | Заробітна плата інженера | $$81577.59$$ |
| 3 | Канцелярське обладнання | 259 |
| 4 | Витрати на програмне забезпечення | 5 449,56 |
| 5 | Єдиний соціальний внесок | $$17947.07$$ |
| Собівартість : | 105655.6 |

Таким чином, у результаті виконаних розрахунків собівартість розробки веб-додатку молодшим програмним інженером складає 105655.6 грн.

# Висновки

Метою даної дипломної роботи було дослідження якості фільтрації різними фільтрами. Для цього був розроблений інтерфейс, для нейронної мережі. Була проведена фільтрація зображень, які були зашумлені адитивним просторово-корельованим шумом, адитивним білим гаусовим шумом, мультиплікативним та спеклом. Проведений аналіз впливу шуму на класифікацію зображень.

Для класифікації зображень було вибрано згорткову нейронну мережу YOLOv5 та навчену модель YOLOv5s. В результаті нейронна мережа використовувалась для класифікації об’єктів на зображеннях з датасету COCO.

В якості шумів біди обрані адитивний білий гаусів шум, адитивний просторово-корельований шум, спекл та мультиплікативний шум. Після обробки фотографій цими шумами було проведено класифікацію зображень та вимір метрик PSNR, FSIM, PSNR-HVS-M та IoU. Після цього зображення були відфільтровані Фільтрами Лі, Фроста, медіанним, ДКП та BM3D. Далі знову проведено класифікацію та вимір метрик.

 В результаті було визначено, що при фільтрації деякі об’єкти можуть класифікуватися гірше ніж на зашумленних зображенняхнезалежно від типу шуму та фільтра. Для адитивного просторово-корельованого шуму. Якщо брати до уваги загальну кількість правильно визначених об’єктів, фільтр Лі не покращує, а навіть погіршує результат, на відміну від нього фільтр BM3D та фільтр Фроста показали кращий результат збільшивши кількість правильно визначених об’єктів на 5,7 та 3.9 відсотка відповідно. Для зображень з адитивним білим гаусовим шумом можна зробити висновок, що показники метрик після обробки ДКП та медіанним фільтрами не покращилися. Після фільтрації зображень з спекл шумом можна зробити висновок, що фільтр Фроста зберігає більше деталей на зображеннях ніж фільтр Лі. Результат класифікації після обробки зображення з спекл шумом фільтром Фроста покращився на 10.3 відсотка, натомість, після обробки фільтром Лі показник значно погіршився відносно зображень без фільтрації. Зображення з мультиплікативним шумом були оброблені фільтрами Лі та BM3D, BM3D показав кращі результати метрик після фільтрації ніж фільтр Лі.

# ПЕРЕЛІК ПОСИЛАНЬ

1. DIFFERENT TYPES OF NOISES IN IMAGES AND NOISE REMOVING TECHNIQUE [Текст] / Mr. Mandar D. Sontakke, Dr. Mrs. Meghana S. Kulkarni, - 2015, 14 p.
2. Improved BM3D image denoising using SSIM-optimized Wiener filter [Електронний ресурс] // Mahmud Hasan & Mahmoud R. El-Sakka: сайт. – Режим доступу: https://link.springer.com/article/10.1186/s13640-018-0264-z
3. Подавление шума на изображениях с использованием больших кодовых книг участков изображений [Текст] / Н.Н. – 2012. – 6 p.
4. An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm [Текст] Marc Lebrun, Arthur Leclaire. – 2012. – 38p.
5. Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study [Текст] / Umme Sara, Morium Akter, Mohammad Shorif Uddin. – 2019. – 11 p.
6. Noise in Digital Image Processing [Електронний ресурс] / Anisha Swain. сайт. – Режим доступу: https://medium.com/image-vision/noise-in-digital-image-processing-55357c9fab71
7. onvolutional Neural Network 3 things you need to know [Електронний ресурс] – 2021: сайт.–Режим доступу: <https://www.mathworks>. com/discovery/convolutional-neural-network-matlab.html
8. Image Denoising Techniques: A Review [Текст] Inderjeet Singh, Lal Chand – 2013. - 5 p.
9. Wiener discrete cosine transform-based image filtering [Текст] / Oleksiy Pogrebnyak, Vladimir V. Lukin KhAI - Aerospace university -2012. - 17 p.
10. Using fuzzy filtering to reduce impulse and Gaussian noise in black and white and color images [Текст] / Arunkumar S., Tej Akula R., Gupta R., Vimala Devi M.R. – 2009. – 4 p.
11. Метрика визуального качества изображений для корректного учета искажений яркости и контраста [Текст]/ Н.Н. Пономаренко, О.И. Еремеев, В.В. Лукин– 2009 6p.
12. FSIM: A Feature Similarity Index for Image Quality Assessment [Текст] / Lin Zhanga, Lei Zhanga, Xuanqin Moub, David - 20013. - 21 p.
13. Microsoft COCO: Common Objects in Context [Електронний ресурс]/ Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, Piotr Dollár сайт.–Режим доступу: <https://arxiv.org/abs/1405.0312>
14. Understanding IU and IoU (Intersection over Union) concepts in deep learning and Python program implementation [Електронний ресурс]/ сайт.–Режим доступу: https://blog.csdn.net/iamoldpan/article/details/78799857