

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Охорона (назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

Магістр

(назва освітнього ступеня)

на тему: Математичне та програмне забезпечення розпізнавання
написів на зображеннях на основі нейромережевого алгоритму
в комп'ютеризованих системах

Виконав: студент 6 курсу групи СІм-61
спеціальності

123 «Комп'ютерна інженерія»

(шифр і назва спеціальності (напряму підготовки))

Чарковський Д.Р.

(підпис)

(прізвище та ініціали)

Керівник

Стадник Н.Б.

(підпис)

(прізвище та ініціали)

Нормоконтроль

Тиш Є.В.

(підпис)

(прізвище та ініціали)

Завідувач

Осухівська Г.М.

кафедри

(підпис)

(прізвище та ініціали)

Рецензент

Цуприк Г.Б.

(підпис)

(прізвище та ініціали)

м. Тернопіль – 2023

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

доц. Осухівська Г.М.

(підпис)

(прізвище та ініціали)

« »

20__ р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня

магістр

(назва освітнього ступеня)

за спеціальністю

123 Комп'ютерна інженерія

студенту

Чарковському Дмитру Руслановичу

(прізвище, ім'я, по батькові)

1. Тема роботи

Математичне та програмне забезпечення розпізнавання

написів на зображеннях на основі нейромережевого алгоритму

в комп'ютеризованих системах

Керівник роботи

Стадник Наталія Богданівна., к.т.н.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом по університету від « 01 » грудня 2023 року № 4/7-1132

2. Термін подання студентом роботи 28.12.2023

3. Вихідні дані до роботи

наукові літературні джерела

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1 Аналіз предметної області дослідження. 2 Теоретична частина.

3. Практична частина. 4 Охорона праці та безпека в надзвичайних ситуаціях

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Тема, мета, задачі, об'єкт, предмет, новизна дослідження. 2. Актуальність дослідження.

3. Порівняння методів детектування текстових областей на зображеннях.

4. Алгоритм роботи ПЗ. 5. Структура нейромережі.

6. Математичний апарат. 7. Тестування роботи алгоритму

8. Порівняння результатів тестування методів.

9. Результати роботи алгоритму. 10. Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Охорона праці</i>	<i>Осухівська Г.М., доцент</i>		
<i>Безпека в НС</i>	<i>Стадник. І.Я., проф. каф. ОХ</i>		

7. Дата видачі завдання _____ 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	<i>Затвердження теми кваліфікаційної роботи</i>	<i>06.12.23</i>	<i>Виконано</i>
2	<i>Аналіз літературних джерел</i>	<i>02.12-05.12.23</i>	<i>Виконано</i>
3	<i>Обґрунтування актуальності дослідження</i>	<i>07.12-09.12.23</i>	<i>Виконано</i>
4	<i>Аналіз предмету дослідження та предметної області</i>	<i>10.12-11.12.23</i>	<i>Виконано</i>
5	<i>Проведення дослідження методів та засобів аналітичного опрацювання даних</i>	<i>11.12-12.12.23</i>	<i>Виконано</i>
6	<i>Оформлення розділу «Аналіз предметної області дослідження»</i>	<i>12.12-13.12.23</i>	<i>Виконано</i>
7	<i>Оформлення розділу «Теоретична частина»</i>	<i>13.12-14.12.23</i>	<i>Виконано</i>
8	<i>Оформлення розділу «Практична частина»</i>	<i>14.12-15.12.23</i>	<i>Виконано</i>
9	<i>Оформлення розділу «Охорона праці та безпека в надзвичайних ситуаціях»</i>	<i>07.12-12.12.23</i>	<i>Виконано</i>
10	<i>Нормоконтроль</i>	<i>11.12-14.12.23</i>	
11	<i>Попередній захист роботи</i>	<i>20.12.23</i>	<i>Виконано</i>
12	<i>Захист кваліфікаційної роботи</i>	<i>27.12.23</i>	

Студент

(підпис)

Чарковський Д.Р.

_____ (прізвище та ініціали)

Керівник роботи

(підпис)

Стадник Н.Б.

_____ (прізвище та ініціали)

АНОТАЦІЯ

Математичне та програмне забезпечення розпізнавання написів на зображеннях на основі нейромережевого алгоритму в комп'ютеризованих системах // Кваліфікаційна робота за освітнім рівнем «магістр» // Чарковський Дмитро Русланович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних систем та мереж, група СІМ-62 // Тернопіль, 2023 // с. – 74, рис. – 33, табл. – 6, аркушів А1 – 10 , бібліогр. – 34.

Ключові слова: НЕЙРОННІ МЕРЕЖІ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, МАШИННЕ НАВЧАННЯ, ОБРОБКА ЗОБРАЖЕНЬ, ТЕКСТОВІ ОБЛАСТІ

Кваліфікаційна робота присвячена дослідженню нейромережевого розпізнавання написів на зображеннях. У процесі дослідження було проведено аналіз існуючих моделей виявлення текстових областей на зображеннях реальних сцен, виходячи з якого було зроблено вибір моделі для подальшої реалізації.

В результаті було запропоновано нейромережевий алгоритм розпізнавання написів на зображення реальних сцен. Реалізація алгоритму передбачає побудову згорткової нейронної мережі U-Net. Для навчання мережі використовується база зображення KAIST, що знаходиться у відкритому доступі. Результатом роботи алгоритму для кожного зображення є маска даного зображення, де відзначені текстові області. Було проведено порівняння з роботами, які проводили тестування на аналогічній базі зображень, під час якого запропонований алгоритм показав найкращі результати.

В результаті кінцевого навчання нейромережевого алгоритму вдалося досягти високої точності: 91,1% для навчальної вибірки, 90,0% для валідаційної та 88,0% для тестової.

ANNOTATION

Mathematics and software for image recognition of inscriptions based on neural network algorithm in computerized systems // Master thesis // Charkovskyi Dmytro // Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Systems and Nets, group CIm - 61 // Ternopil, 2023 // p. – 74, fig. – 33 , table. – 6, Sheets A1 - 10 , Ref. - 34.

Keywords: NEURAL NETWORKS, CONVOLUTIONAL NEURAL NETWORK, MACHINE LEARNING, IMAGE PROCESSING, TEXT AREAS

The thesis deals with the study of neural network recognition of inscriptions on images. In the process of research, an analysis of existing models for detecting text areas on images of real scenes was carried out, based on which a model was selected for further implementation.

As a result, a neural network algorithm for recognizing inscriptions on images of real scenes was proposed. The implementation of the algorithm involves the construction of a U-Net convolutional neural network. The publicly available KAIST image database is used to train the network. The result of the algorithm for each image is a mask of this image, where the text areas are marked. A comparison was made with works that conducted testing on a similar image database, during which the proposed algorithm showed the best results.

As a result of the final training of the neural network algorithm, it was possible to achieve high accuracy: 91.1% for the training sample, 90.0% for the validation sample, and 88.0% for the test sample.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

DCT (Discrete Cosine Transform) – дискретно-косинусне перетворення

HOG (Histogram of Oriented Gradients) – гістограма спрямованих градієнтів

ML (machine learning) – машинне навчання

MSER (Maximally Stable Extremal Regions) – метод максимально стабільних екстремальних областей

SWT (Stroke Width Transform) – метод перетворення за товщиною штриха символу тексту

БД – база даних

ЗНМ – згорткова нейронна мережа

ПЗ – програмне забезпечення

ПФ – перетворення Фур'є

ПНМ – повнозгорткові нейронні мережі

ЦП – центральний процесор

ШІ – штучний інтелект

ШНМ – штучна нейронна мережа

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ	12
1.1 Методи зв'язаних компонентів	12
1.2 Текстурні методи	14
1.3 Методи глибокого навчання	18
1.4 Порівняння методів детектування текстових областей на зображеннях	21
1.5. Висновки до розділу	23
РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА	24
2.1. Використовувані інструменти	24
2.2. Модулі програми для навчання згорткової нейронної мережі	26
2.3. Архітектура програмного забезпечення	27
2.4. Опис структури мережі	28
2.5. Параметри оцінки під час навчання мережі	30
2.6. Фільтри для обробки зображень	31
2.7. Перетворення Фур'є	33
2.8. Частотні фільтри, що згладжують	34
2.9. Висновки до розділу	35
РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА.	36
3.1. Підбір розмірів вхідних зображень	36
3.2. Підбір кількості шарів	39
3.3. Підбір типів згорткових шарів	40
3.4. Передобробка вхідних зображень	41
3.5. Частотний аналіз з використанням перетворення Фур'є	46
3.6. Розбиття та стиснення зображень перед подачею на вхід мережі	50
3.7. Порівняння з іншими роботами	55
3.8. Висновки до розділу	58

РОЗДІЛ 4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	59
4.1. Охорона праці.....	59
4.2. Комп'ютерне забезпечення процесу оцінки радіаційної та хімічної обстановки.....	62
4.3. Висновки до розділу	64
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	66
ДОДАТОК А. Тези конференції	
ДОДАТОК Б. Фрагмент коду навчання мережі	

ВСТУП

Актуальність теми. В даний час наявна гігантська кількість інформації, котра зберігається як зображення, зміст яких є певною цінністю. Детектування та подальше розпізнавання тексту на зображеннях може бути застосоване в таких областях як:

- переведення фотографій документів у текстову форму [1];
- автоматичне визначення номерних знаків автомобілів [2];
- геолокація об'єкта за назвами вулиць [3];
- покращення якості детектування та розпізнавання об'єктів на зображеннях.

Однак обсяги інформації, що зберігається у вигляді зображень, великі, що унеможлиблює її обробку вручну. Тим не менш, автоматизовані методи обробки зображень дозволяють успішно справлятися із цим завданням.

Незважаючи на широку область можливого застосування та прогрес у сфері ML [4, 5], виявлення та сегментація текстових областей на зображеннях все ще є проблемою. Текст, розташований на зображеннях реальних сцен, може бути різним за розміром, стилем, кольором, поворотом щодо горизонту.

Також можливе його перекриття іншими об'єктами на зображенні або низька контрастність із фоном. Застосування фільтрів для згладжування шумів і збільшення контрастності на межах об'єктів не є універсальним вирішенням проблеми, оскільки може викликати помилкові спрацьовування, так і заважати виявленню значущих областей.

Мета дослідження: розробка нейромережевого алгоритму розпізнавання текстових областей зображення реальних сцен.

В роботі поставлено та розв'язано **наступні задачі:**

- проаналізувати існуючі моделі виявлення текстових областей на зображення реальних сцен;
- вибрати модель для подальшої реалізації;

- розробити алгоритм розпізнавання написів;
- реалізувати програму з урахуванням розробленого алгоритму;
- провести навчання та тестування розробленого алгоритму.

Об'єкт дослідження: нейромережевий алгоритм розпізнавання написів на зображеннях реальних сцен.

Предмет дослідження: нейромережеві моделі, призначені для розпізнавання текстових областей.

Методи дослідження: Методологічною основою дослідження є моделювання фізичного середовища, дослідження методів навчання, програмна реалізація обраного методу та проведення чисельних експериментів щодо навчання моделі розпізнавання написів.

Наукова новизна отриманих результатів:

- досліджено основні характеристики методів, котрі використовуються для розпізнавання областей тексту (методи пов'язаних компонентів, текстурні методи, методи глибокого навчання);
- створено нейромережевий алгоритм розпізнавання написів на зображення реальних сцен (на основі ЗНМ U-Net);
- при тестуванні алгоритм досяг високої точності для різних видів вибірок (для навчальної 91,1%, для валідаційної 90,0% , для тестової 88,0%).

Практичне значення одержаних результатів. Розроблене ПЗ може бути використане в дослідницьких установах, підприємствах або незалежними дослідниками як передобробка зображень реальних сцен для подальшого вилучення з сегментованих областей текстової інформації.

Публікації. Результати дослідження апробовано на XI науково-технічній конференції «Інформаційні моделі, системи та технології» у вигляді опублікованих тез [23].

Структура роботи. Робота складається з пояснювальної записки та графічної частини. Пояснювальна записка складається з вступу, 4 розділів, висновків, списку використаної літератури та додатків. Обсяг роботи:

пояснювальна записка – 74 арк. формату А4, графічна частина – 10 аркушів формату А1.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ

Методи розпізнавання текстових областей використовуються для визначення наявності та виділення розташування текстових областей на зображеннях. Однак, точність цього визначення може бути знижена через різні розміри, стилі та напрями написів. Крім того, низький контраст із фоном або складні фони можуть викликати додаткові труднощі. Усі існуючі методи розпізнавання текстових областей можна розділити на такі групи:

- методи пов'язаних компонентів;
- текстурні методи;
- методи глибокого навчання.

1.1. Методи пов'язаних компонент

Принцип роботи цих методів полягає в пошуку та об'єднанні малих компонентів у великі на підставі певних характеристик пікселів: яскравість, колір, товщина контуру елемента. Потім отриманих компонентів видобуваються ознаки для подальшої класифікації на текстові та нетекстові компоненти. Знайдені текстові компоненти витягуються з зображень та об'єднуються у текстові області.

Головними перевагами методів зв'язаних компонентів є простота обчислень і висока точність. Однак ці методи погано працюють при зміні повороту або масштабу зображень, обробці складних фонів та деяких інших випадках.

Для розгляду було обрано MSER та SWT.

Перший з них використовує екстремальні області для розпізнавання текстових областей. Екстремальні області на зображенні визначаються двома умовами щодо безлічі пікселів усередині області [6]:

- область залишається замкненою при перетворенні координат (поворот, розтяг);

– область залишається замкненою при зміні яскравості зображення.

Застосування методу MSER для різних порогових значень представлено на рис. 1.1.

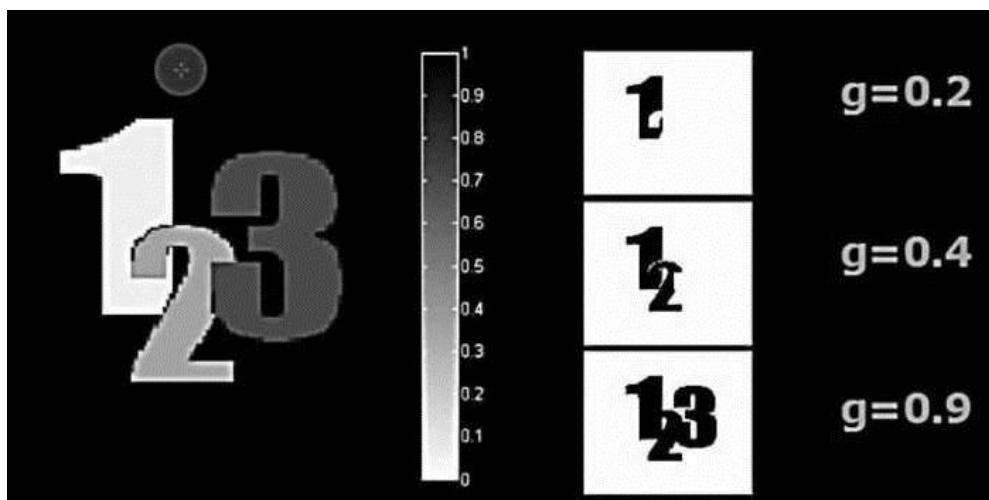


Рисунок 1.1 – Приклад застосування методу MSER

Дані методи вимагають мало пам'яті для реалізації, можуть працювати в режимі реального часу, але погано розпізнають текстові області на розмитих або неконтрастних зображеннях [7].

Метод SWT будується на припущенні, що літери та символи на текстових областях мають певні геометричні особливості: однакова ширина кожного символу і однакова товщині штриха символу тексту. Інакше кажучи, найчастіше текстові області мають невелике зміна товщині штриха від символу до символу, на відміну нетекстових областей [8].

SWT перетворює зображення на масив, де кожен елемент містить значення товщини штриха для відповідного пікселя, використовуючи для цього оператор Кенні. Далі пікселі групуються у передбачувані текстові області. Два сусідні пікселі об'єднуються, якщо вони мають рівні значення товщини штриха. Відокремити текстові області від нетекстових дозволяє однакове значення товщини тексту. Потім із виділених текстових областей формується передбачене текстове поле.

На рис. 1.2 показано послідовність застосування методу SWT.

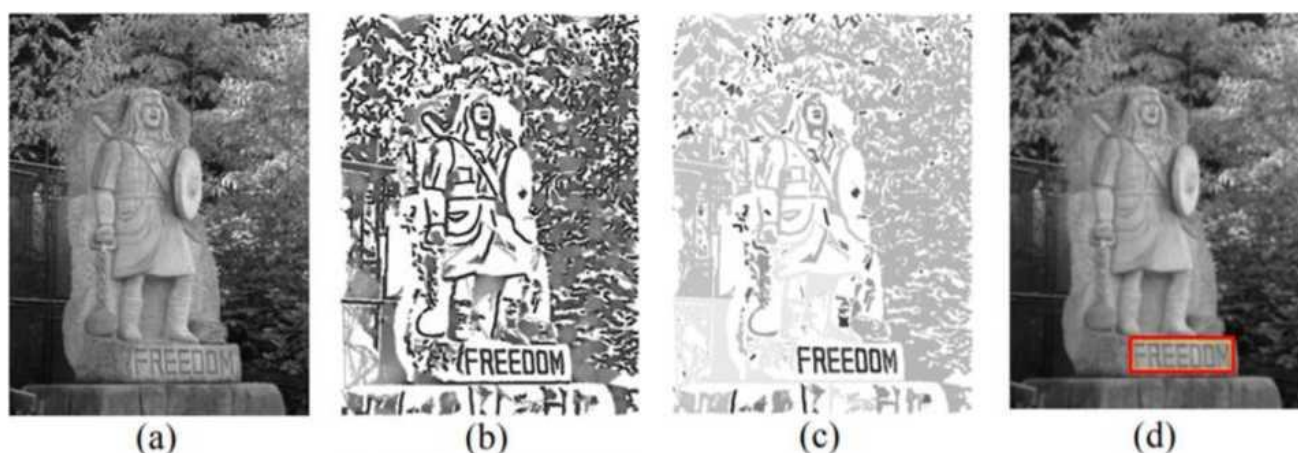


Рисунок 1.2 – Приклад застосування методу SWT (а – вихідне зображення, б – зображення зі значеннями ширини штриха для кожного пікселя, с – зображення з виділеними текстовими областями, d – зображення з передбаченим текстовим полем)

Експерименти показують, що метод SWT дуже ефективний для виявлення тексту. Даний метод може бути застосований до різних шрифтів та мов, а також до різних розмірів та поворотів написів. Однак метод SWT не є повністю автоматичним і вимагає попереднього підбору параметрів, що може призвести до помилкових спрацьовувань для складних випадків.

1.2. Текsturні методи

Грунтуються на ідеї про те, що текстові області відрізняються від фону зображень високочастотними та регулярними текстурними ознаками. У цих методах відбувається застосування класифікатора до окремих областей зображення визначення наявності чи відсутності тексту. Текsturні методи добре показують себе на зашумлених та спотворених зображеннях, а також за наявності складного тексту або тла. Однак, така класифікація складна у обчислювальному відношенні, що позначається на часі роботи методів. Додаткові труднощі можуть бути

викликані тим, що нарівні з тестовими областями на зображеннях вулиць регулярність мають такі об'єкти, як цегляна кладка, вікна будинків, листя дерев.

Для розгляду було обрано такі текстурні методи:

- DCT;
- HOG;
- ознаки Хаара.

DCT було створено для стиснення зображень, що дозволяє трансформувати простір зображення у простір властивостей з нижчою розмірністю [9].

Однак цей метод можна використовувати і для вилучення текстових областей із зображень. Перетворення застосовується кожного окремого фрагмента зображення незалежно друг від друга, після чого відбувається об'єднання значень. Значення отриманих частотних коефіцієнтів відбивають локальну періодичність області зображення. Неявна періодичність означає, що у межах виникають розриви.

Алгоритм, запропонований Zhong, використовує DCT і складається з двох основних етапів [10]:

- виявлення передбачуваних текстових областей у стиснутій області частотних коефіцієнтів;
- постобробка областей.

Застосування DCT показано на рис. 1.3.

Запропонований алгоритм має високу швидкість за рахунок роботи з простором властивостей з більш низькою розмірністю і постобробкою зображень зі зниженою роздільною здатністю. Однак цей алгоритм не надає високої точності при розподілі областей на текстові та нетекстові.

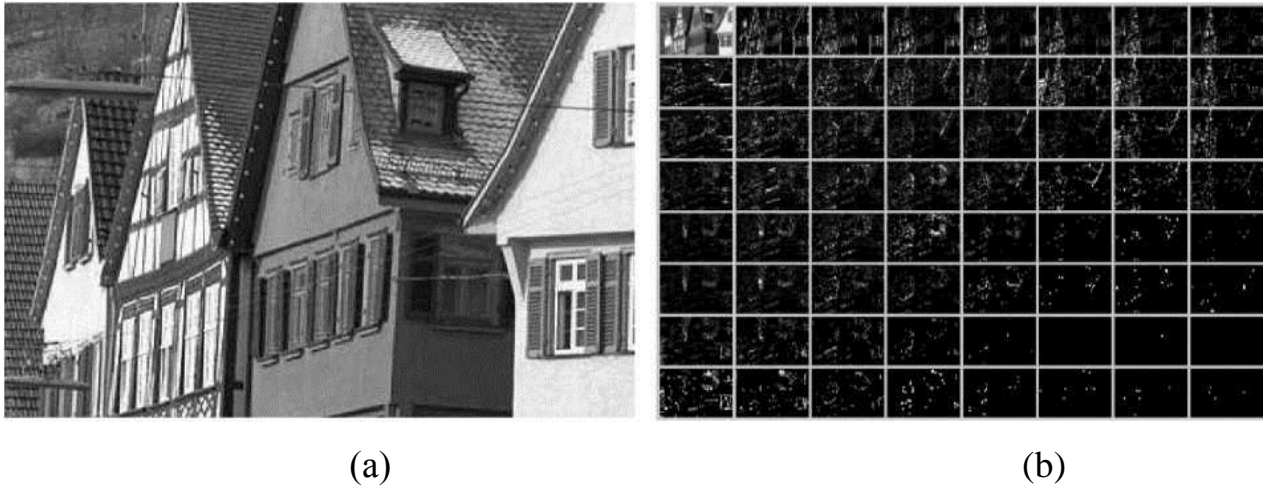


Рисунок 1.3 – Застосування DCT (a – вихідне зображення, b – зображення з виділеними частотними коефіцієнтами)

Ще одним текстурним методом, що розглядається, є використання HOG - дескрипторів. Спочатку відбувається поділ зображення на комірки, потім кожного пікселя всередині комірки розраховується гістограма напрямів градієнтів. Об'єднання отриманих значень для кожного осередку називається HOG-дескриптором. Отримане градієнтне зображення виділяє контури на зображенні та виключає з розгляду несуттєву інформацію [11].

На рис. 1.4 показано приклад застосування цього методу для розпізнавання цифр. Для кожної цифри представлені гістограми, що показують перевагу кожного градієнта [12].

У цьому прикладі розглядаються окремі цифри, а чи не повноцінний текст. Такі дані легко візуалізувати та інтерпретувати людиною на відміну від зображень реальних сцен, на яких крім тексту є й інші об'єкти. Тому метод HOG найбільше підходить для попередньої обробки зображень, але не для самостійного застосування.

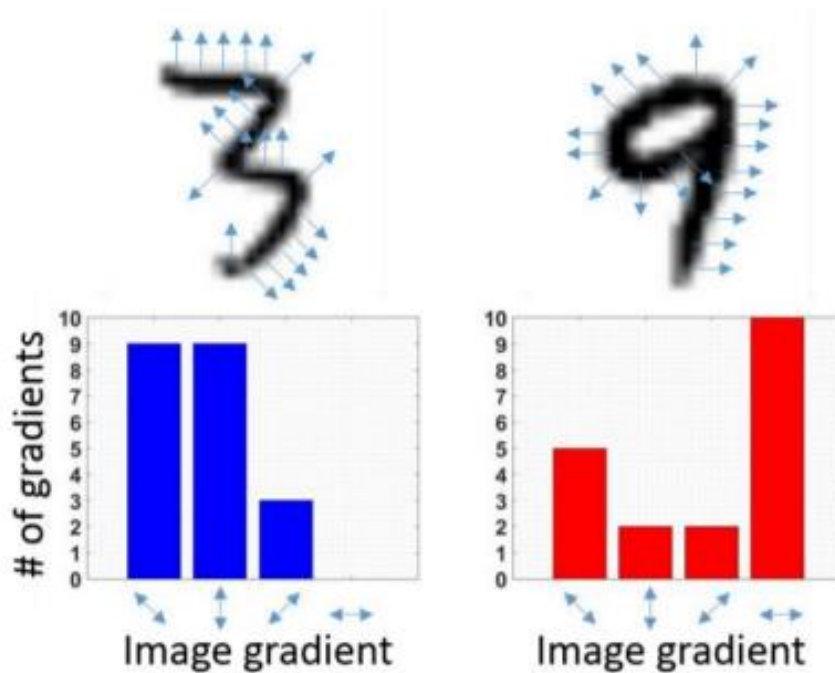


Рисунок 1.4 – Приклад застосування методу HOG

Ознаки Хаара використовують вейвлет-перетворення Хаара. Даний метод використовує як ознаки прямокутні області, розділені на контрастні частини. Дані ознаки застосовуються шляхом накладення на фрагменти зображення, що розглядається, при накладенні відбувається обчислення суми інтенсивностей пікселів для кожної області ознаки. Відмінність між отриманими значеннями дозволяє розподілити за категоріями фрагменти зображення [13].

У більшості випадків текст являє собою кілька частин, що добре контрастують по горизонталі або вертикалі на зображенні незалежно від кольору тексту або фону. Цей факт надає можливість застосування ознак Хаара виявлення текстових областей [14,15].

На рис. 1.5 представлені приклади набору ознак Хаара.

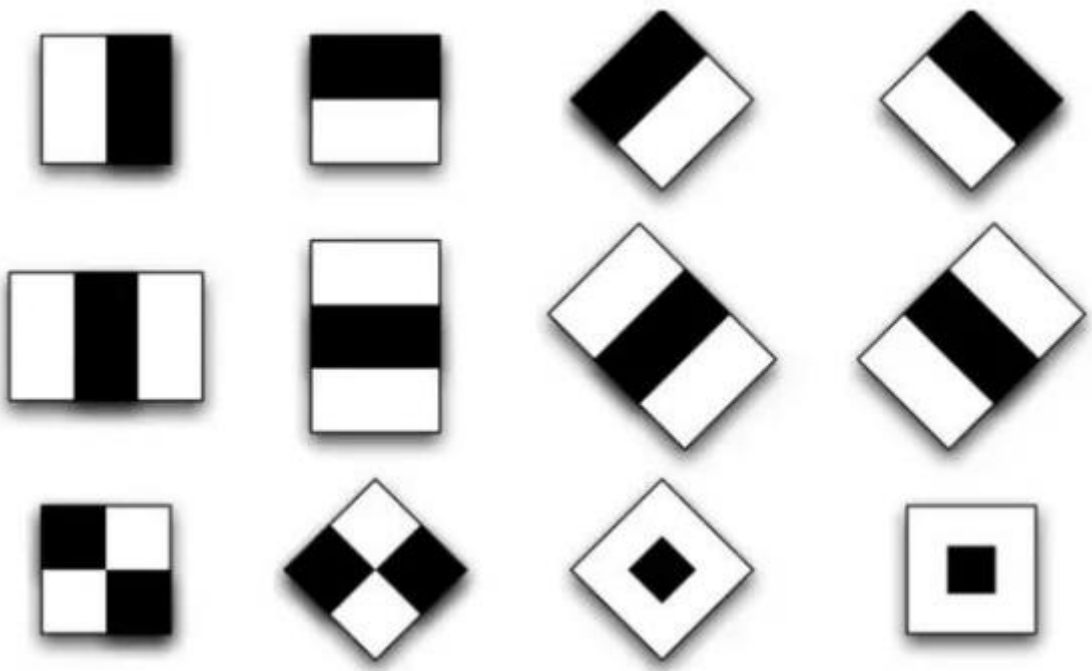


Рисунок 1.5 - Ознаки Хаара

Головним негативним моментом застосування цього методу виявлення текстових областей є залежність результату від початкового вибору набору застосовуваних ознак. Позиції літер у тексті та їх форми можуть відрізнятися для різних зображень, що може викликати додаткову проблему при підготовці набору ознак.

1.3. Методи глибокого навчання

Належать до сфери ML [16, 17]. Найбільш поширеними методами є різні типи ШНМ. Повнозв'язна нейронна мережа складається з шарів нейронів. Властиво нейрони попереднього шару пов'язані з нейронами наступного шару. У процесі навчання та роботи мережі нейрони послідовно активуються шар за шаром. Процес активації є формування вихідного сигналу нейрона у відповідь отриманий вхідний сигнал. Навчання такої нейронної мережі полягає у підборі коефіцієнтів ваги зв'язків між нейронами таким чином, щоб вихідні значення мережі відповідали еталонним [18].

ЗНМ є класом ШНМ. На відміну від повнозв'язних нейронних мереж, у ЗНМ сусідні нейрони одного шару пов'язані з локальною областю нейронів наступного шару, що значно скорочує кількість ваг, що настраюються під час навчання.

Це робить ЗНМ більш обчислювально ефективними, дозволяючи використовувати як вхідні значення зображення більшого розміру. ЗНМ можуть включати підвбіркові шари, використовувані зменшення розмірів даних рахунок об'єднання виходів кластерів нейронів в один вхідний нейрон [19].

Структура ЗНМ представлена на рис. 1.6.

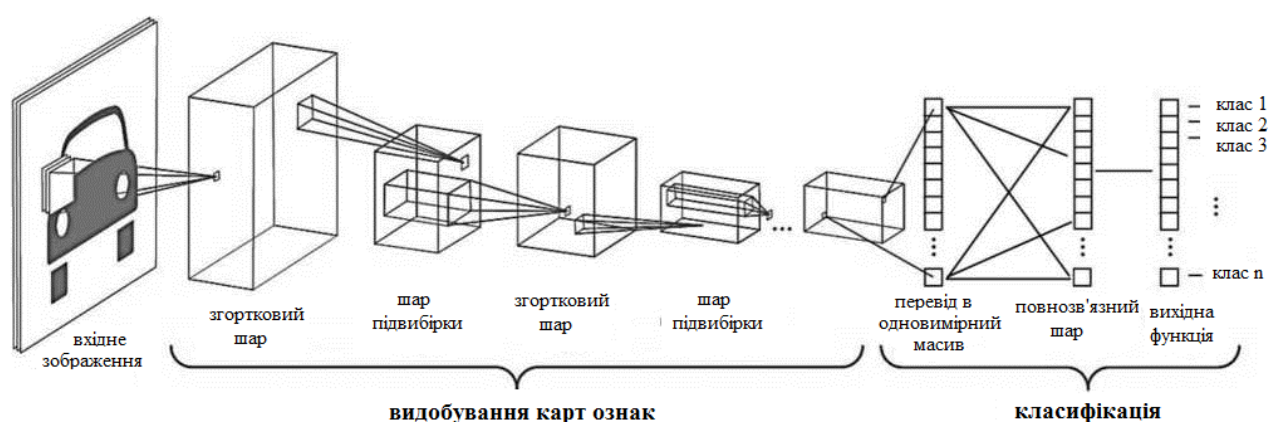


Рисунок 1.6 – Структура ЗНМ

Основною перевагою ЗНМ є можливість роботи зі складними зображеннями, оскільки вона має інваріантність до поворотів, розтягувань, змін яскравості чи контрастності [19]. Така мережа може отримувати інформацію із зображення з невеликими порівняно з повнозв'язними мережами обчислювальними витратами. Недоліком ЗНМ є її вихідні значення, які у вигляді вектора значень: ця мережа може бути використана виявлення наявності тексту на зображенні, але з визначення його местоположення.

Структура ПНМ схожа на структуру ЗНМ. Головна відмінність полягає у вихідному шарі мережі. Даний шар здійснює об'єднання всіх карт ознак з попередніх шарів для створення зображення у вигляді сегментованої карти, кожен сегмент якої являє собою область певного класу. Отримана картка може бути

використана для визначення положення текстових областей на зображенні [20].

Приклад структури ПНМ показаний на рис. 1.7.

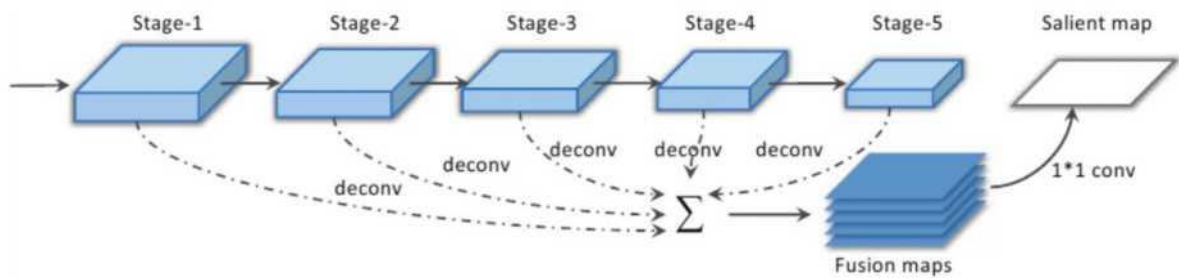


Рисунок 1.7 - Приклад структури ПНМ

При використанні ПНМ можуть з'являтися помилкові спрацьовування або пропущені символи за дуже низької контрастності, кривизни, сильного відображення світла, щільного розташування рядків тексту або великих проміжків між символами. Ще одним недоліком цього типу мережі є швидкодія.

Архітектура U-Net. Вихідне зображення ПНМ архітектури U-Net збігається за розмірами із вхідним зображенням. Відмінною рисою U-Net є її симетрична структура. Вона складається з звужувальної та розширюючої частин, які створюють U-подібну структуру. Розширювальна частина складається з шарів дискретизації, що підвищує, які дозволяють обробляти більшу кількість карт ознак. У цій структурі відсутні пов'язані шари, в результаті вихідне зображення мережі містить передбачення тільки для тих пікселів, які відповідають вхідному зображенню [21].

Приклад ПНМ архітектури U-Net наведено на рис. 1.8.

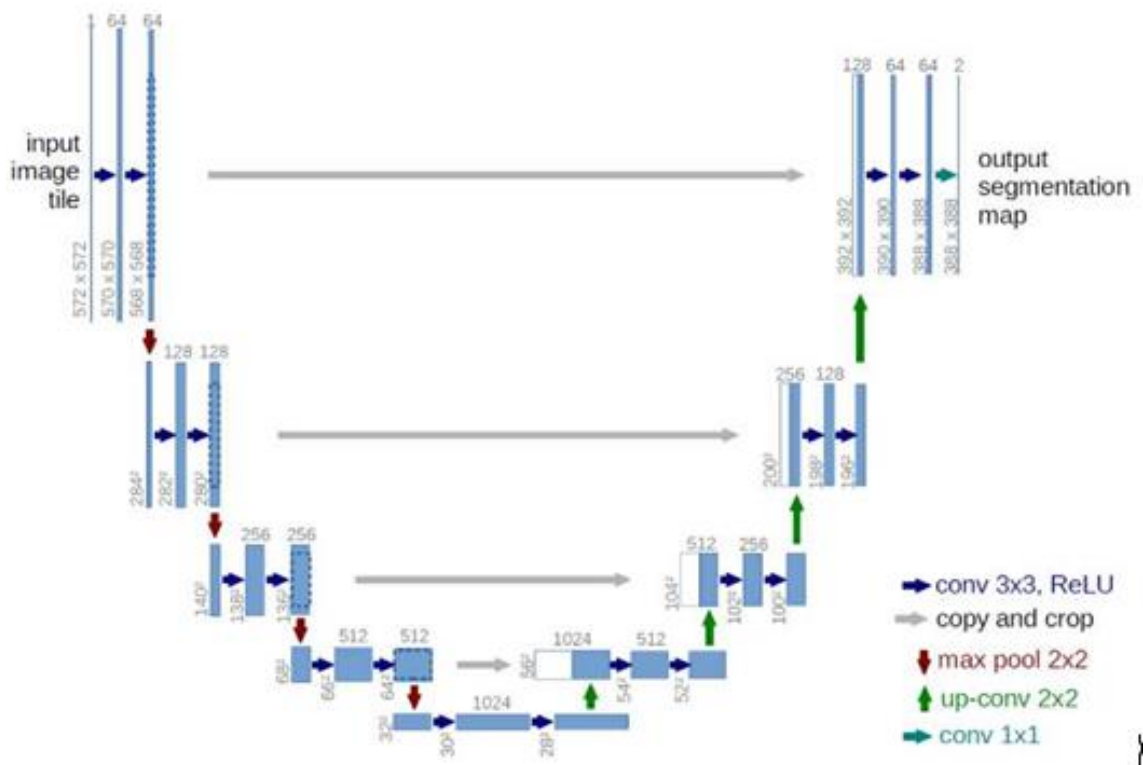


Рисунок 1.8 - Приклад ШНМ архітектури U-Net

Основною перевагою архітектури U-Net є наявність більшої кількості шарів підвищує дискретизації, які дозволяють обробляти більшу кількість карт ознак. Крім того, U-Net забезпечує попіксельну точність завдяки однаковому розміру вхідних та вихідних зображень [22].

1.4. Порівняння методів детектування текстових областей на зображеннях

Для порівняння було розглянуто такі характеристики:

- швидкість роботи (кількість часу, необхідна для навчання цього методу);
- ресурсна вартість обчислень (вимоги до обчислювальних ресурсів обладнання);
- робота зі складними зображеннями: геометричні перетворення (повороти, розтягування); текстурні спотворення (шум, низька контрастність, складні фони);

– ефективність методу (точність чи інші заходи оцінки).

Результати порівняння представлено у таблиці 1.1.

Таблиця 1.1

Порівняння методів

Характеристика		Швидкість роботи	Ресурсна вартість обчислень	Геометричні перетворення	Текстурні спотворення	Ефективність методу
Метод						
Методи пов'язаних компонент	Метод максимально стабільних екстремальних областей	Висока	Низька	Не працює	Не працює	Середня
	Метод перетворення за товщиною штриха символу тексту	Середня	Низька	Працює	Не працює	Середня
Текстурні методи	Дискретно косинусне перетворення	Висока	Низька	Не працює	Не працює	Низька
	NOG-дескриптори	Середня	Низька	Не працює	Не працює	Низька
	Ознаки Хаара	Середня	Низька	Не працює	Не працює	Низька
Методи глибокого навчання	Згорткова нейронна мережа	Середня	Низька	Працює	Працює	Середня
	Повнозгорткова нейронна мережа	Низька	Середня	Працює	Працює	Висока
	Мережа архітектури U-Net	Низька	Середня	Працює	Працює	Висока

Як видно з таблиці, методи зв'язаних компонентів мають високу швидкість роботи, проте погано працюють з різними спотвореннями зображень та мають середню ефективність методу [23]. Текстурні методи мають середню швидкість роботи, погано працюють з різними спотвореннями і мають низьку ефективність.

Методи глибокого навчання є найповільнішою групою через тривалий час навчання, але добре працюють з різними спотвореннями зображень, а також мають високу продуктивність. Отже, методи, що ґрунтуються на глибокому навчанні, становлять велику цінність для розгляду.

ЗНМ пред'являють менше вимог до обладнання порівняно з іншими мережами, але є недостатніми для визначення розташування текстових областей на зображеннях. Повнозгорткові нейронні мережі з несиметричною структурою та мережі архітектури U-Net знаходяться на одному рівні. Однак симетрична архітектура мережі U-Net надає більше можливостей для розпізнавання текстових областей через наявність більшої кількості шарів дискретизації, що підвищує, для обробки карт ознак. Тому архітектура U-Net була обрана для реалізації алгоритму в даній роботі.

1.5. Висновки до розділу

Розглянуто особливості предметної області, зокрема описано основні методи розпізнавання текстових областей, котрі застосовуються для детектування та подальше розпізнавання тексту на зображеннях (методи пов'язаних компонентів, текстурні методи, методи глибокого навчання).

Проведено порівняльний аналіз характеристик методів, котрий дозволив виявити їх позитивні та негативні сторони.

За результатами виконаного аналізу вибрана архітектура U-Net для побудови алгоритму.

РОЗДІЛ 2

ТЕОРЕТИЧНА ЧАСТИНА

2.1. Інструменти, використані для проектування алгоритму

Для реалізації ПЗ було обрано високорівневу мову програмування Python [24, 25], оскільки для нього реалізовані бібліотеки для роботи з ЗНМ. Були використані функції відкритої нейромережевої бібліотеки Keras [26], що є надбудовою над бібліотекою ML TensorFlow. Також для роботи з багатовимірними масивами зображень задіяно розширення мови NumPy [27]. Для обробки зображень було задіяно бібліотеку scikit-image.

Написання програмного коду проводилося в інтегрованому середовищі розробки PyCharm. Дане середовище розробки включає інтегрований відлагоджувач і запуск тестування, надає зручний інтерфейс для роботи з програмним кодом.

Для навчання ЗНМ був використаний набір хмарних служб Google Cloud Platform, що надає такі служби, як хмарні обчислення, зберігання даних, аналіз даних та ML. Було створено віртуальну машину для навчання ЗНМ.

Як базу зображень було обрано базу KAIST Scene Text Database, котра містить 3000 зображень, знятих у різних середовищах, включаючи сцени на вулиці та в приміщенні за різних умов освітлення (ясний день, ніч, сильне штучне освітлення тощо). База даних сцени KAIST класифікується відповідно до мови записаного тексту сцени: корейська, англійська (номер) і змішана (корейська + англійська + номер). Текст сцени на зображеннях є репрезентативним для звичайного тексту на корейських вулицях або в магазинах.

З бази KAIST було вибрано 1215 фотографій різних вивісок, що містять текст англійською та корейською мовою. Зображення були отримані за допомогою цифрової камери високої роздільної здатності або мобільного телефону з низькою роздільною здатністю. Усі зображення були приведені до розміру 640×480 пікселів.

Для визначення розташування тексту для кожного зображення представлено зображення-маска, на якому чорним кольором виділено тло, а червоним – текст.

Приклади зображень із бази зображень KAIST Scene Text Database представлені на рис. 2.1.



Рисунок 2.1 - Приклади зображень із бази зображень KAIST Scene Text Database

Було проведено штучне збільшення бази зображень у 7 разів за рахунок повороту та обрізання зображень. Повороти проводилися в обидві сторони від -18° до 18° з кроком 6° . Також усі маски були переведені у чорно-білий формат. Результат такої обробки на одному зображенні представлений на рис. 2.2.



Рисунок 2.2 - Повертання одного зображення з бази зображень

Додатковий спосіб збільшення бази зображень представлений на рис. 2.3, де до одного зображення застосовується як стиск до розмірів, що використовуються для структури мережі, так і поділ зображення на 4 частини з перекриттям.



Рисунок 2.3 - Стиснення та поділ з перекриттям одного зображення з бази зображень

2.2. Модулі програми для навчання згорткової нейронної мережі

Програма містить дві складові частини, оформлені у виді модулів:

- попереднього опрацювання бази зображень;
- відповідальний за роботу зі ЗНМ.

При запуску роботи першого модуля спочатку відбувається завантаження даних, застосування фільтрів до вхідних зображень, а також завантаження зображень-масок, з якими далі відбуватиметься порівняння виходів мережі, що навчається. Для бази зображень формуються масиви, до яких надалі відбуватиметься звернення.

При запуску роботи модуля, відповідального за роботу зі ЗНМ, зчитуються масиви БД зображень. База зображень ділиться на навчальну, валідаційну та тестову вибірки у співвідношенні 72%, 18% та 10% відповідно. Далі відбувається створення нейронної мережі із заздалегідь прописаними параметрами шарів. Після цього можливий виклик навчання створеної ШНМ на переданих класу вибірках. Після завершення навчання здійснюється збереження параметрів мережі. Потім

мережа запускається для прогнозів на тестових значеннях, кожне з яких зберігається в окремий файл.

2.3. Архітектура програмного забезпечення

При запуску програми попередньої обробки зображень з БД відбувається завантаження зображень їх попередня фільтрація та завантаження зображень-масок.

При запуску розробленого ПЗ, пов'язаного з ШНМ, відбувається поділ раніше завантаженої та обробленої бази зображень показано у п. 2.2. Далі створюється нейронна мережа, відбувається її навчання, валідація та тестування на сформованих раніше вибірках. Після закінчення навчання програма зберігає параметри ЗНМ.

Об'єднаний алгоритм роботи програм як блок-схеми представлений на рис. 2.4.

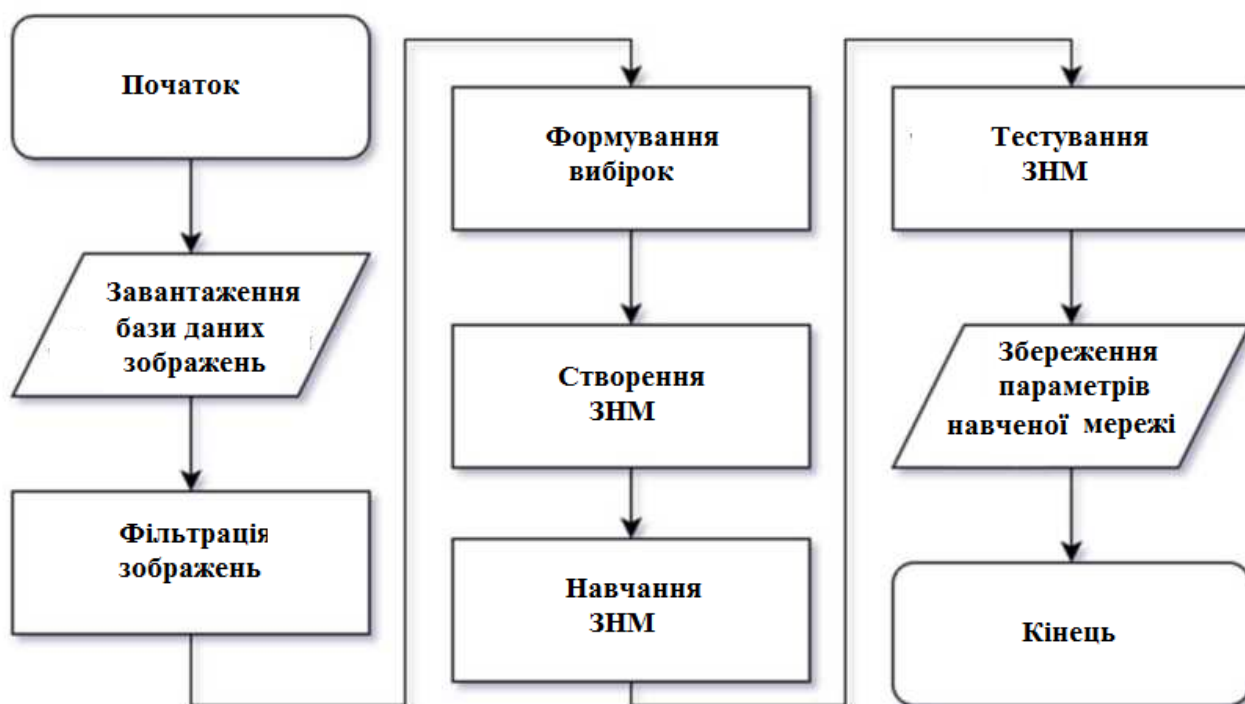


Рисунок 2.4 – Об'єднаний алгоритм роботи програм

2.4. Опис структури мережі

Для навчання використовуються дві конфігурації мережі: перша представлена на рис. 2.5 та буде застосовуватися для кінцевого навчання мережі U-Net.

Вхідними значеннями є зображення розміром 384×384 пікселів. Перший згортковий шар містить 32 карти згорткового шару однакового розміру, одержуваних за допомогою ядра розміром 3×3 . Перший підвибірковий шар є операцією вибірки локальних максимальних значень за допомогою ядра розміром 2×2 і кроком, що відповідає розміру ядра. З другого по п'ятий згортковий шари відбувається збільшення кількості карт: 64, 128, 256, 512 відповідно. З другого до четвертого шару підвибірки жодних змін не відбувається.

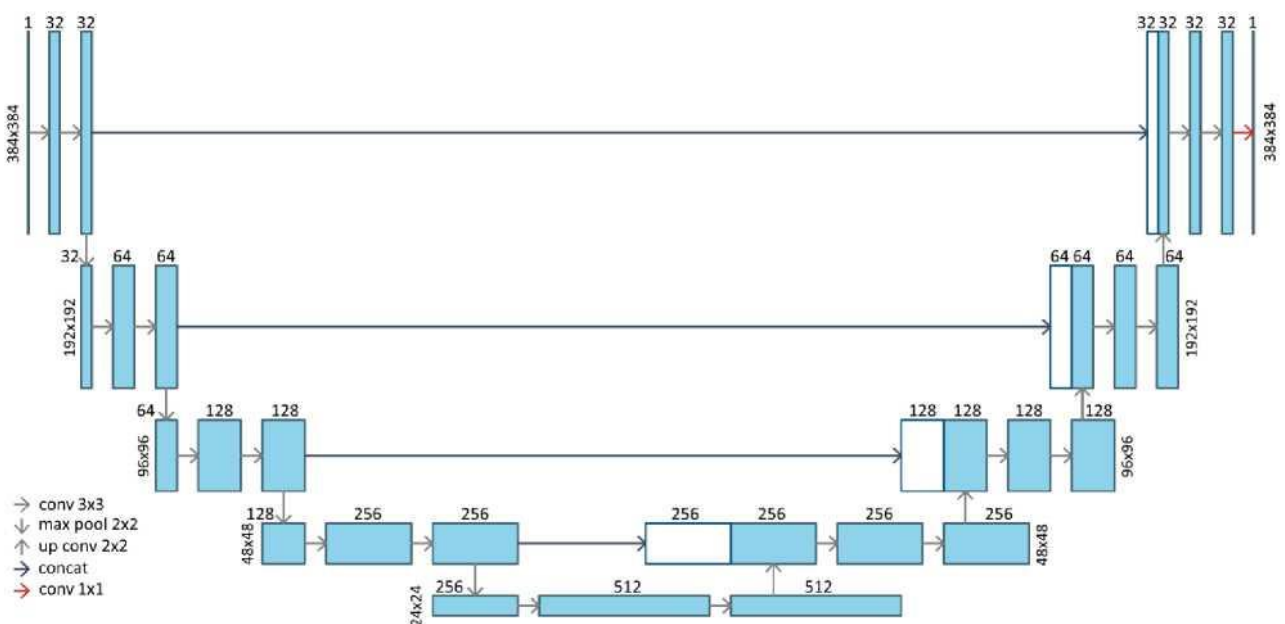


Рисунок 2.5 - Схема кінцевої ЗНМ U-Net

Далі додаються шари конкатенації карт ознак кодууючої частини. І замість згорткових шарів використовуються шари зворотної згортки, що дає змогу збільшити розмір зображення вдвічі за обома осями. І з шару 6 по шар 9 власне зворотної згортки відбувається зменшення кількості карток: 256, 128, 64, 32

Рисунок 2.6 - Схема перевірконої ЗНМ U-Net

Ця мережа відрізняється від кінцевої мережі розмірами вхідних зображень, а також кількістю шарів.

2.5. Параметри оцінки під час навчання мережі

Для обчислення помилки мережі формулою (2.2) розраховується коефіцієнт Дайса (Dice coefficient), який показує міру подібності зображень:

$$dice\ coefficient = \frac{2 * intersection(y_{true}, y_{pred}) + smooth}{y_{true} + y_{pred} + smooth}, \quad (2.2)$$

де y_{true} - дійсне значення пікселів зображень, y_{pred} - передбачене значення пікселів зображень, $intersection$ - перетин/множення значень пікселів зображень, $smooth$ - коефіцієнт згладжування.

Чим вище значення коефіцієнта Дайса, тим більше справжніх і передбачуваних значень пікселів зображень збігається, відповідно тим краще детектуються текстові області.

Як додаткові параметри оцінювання роботи мережі було вирішено ввести такі значення, представлені у формулах (2.3) – (2.5):

$$Precision = \frac{TP}{TP + FP}, \quad (2.3)$$

де $Precision$ - точність, TP - істинно-позитивне рішення, FP – хибно- позитивне рішення.

$$Recall = \frac{TP}{TP + FN}, \quad (2.4)$$

де $Recall$ – повнота, TP - істино-позитивне рішення, FN – хибно- негативне рішення.

$$F = (\beta^2 + 1) \frac{Precision * Recall}{\beta^2 * Precision + Recall}, \quad (2.5)$$

де F - F -міра, β - коефіцієнт від 0 до ∞ .

2.6. Фільтри для обробки зображень

2.6.1. Двосторонній фільтр згладжування

Визначається за формулою (2.6):

$$I^{filtered}(x, y) = \frac{\sum_{j=-N}^N \sum_{i=-N}^N I(i, j) * f_r(I(x, y), I(i, j)) * g_s(x, y, i, j)}{\sum_{j=-N}^N \sum_{i=-N}^N f_r(I(x, y), I(i, j)) * g_s(x, y, i, j)}, \quad (2.6)$$

де $I^{filtered}$ - відфільтроване зображення; I – вхідне зображення для фільтрації; (x, y) - координати поточного пікселя, що підлягає фільтрації; (i, j) – координати поточного пікселя з маски; N - розмір маски; f_r - ядро діапазону для згладжування різниць інтенсивностей (функція Гауса); g_s – просторове ядро для згладжування різниць координат (функція Гауса).

2.6.2 Фільтр різкості

Фільтрація проводиться кореляцією вихідного зображення з обраною маскою, що розраховується за формулою (2.7):

$$I^{filtered}(x, y) = \sum_{j=-N}^N \sum_{i=-N}^N kernel(i, j) * I(x + i, y + j), \quad (2.7)$$

де $I^{filtered}$ - відфільтроване зображення; I – вхідне зображення для фільтрації; $kernel$ - маска; N - розмір маски.

Маска для фільтра, що підвищує різкість, і маска для фільтра різкості, що виділяє краї, представлені на рис. 2.7.

0	-1	0
-1	5	-1
0	-1	0

а)

0	-1	0
-1	4	-1
0	-1	0

б)

Рисунок 2.7 - Маски для фільтра різкості (а - підвищує різкість, б - виділяє краї)

2.6.3. Фільтр, що виділяє краї

Спочатку за формулою (2.8) виділяється заднє тло шляхом послідовного застосування ерозії та розширення:

$$Bg = (I \ominus kernel) \oplus kernel, \quad (2.8)$$

де I – вхідне зображення; $kernel$ - маска; Bg - зображення заднього фону.

Потім за формулою (2.9) кожен піксель вихідного зображення поділяється на відповідний піксель заднього фону:

$$D(x, y) = \frac{I(x, y)}{Bg(x, y)}, \quad (2.9)$$

де (x, y) - координати пікселя; D - зображення, отримане при розподілі.

Після чого за формулою (2.10) отримане зображення проходить через граничну функцію:

$$I^{filtered}(x, y) = \begin{cases} 0, & \text{if } D(x, y) \geq \text{threshold;} \\ 255, & \text{if } D(x, y) < \text{threshold,} \end{cases} \quad (2.10)$$

де $threshold$ - порогове значення; $I^{filtered}$ - відфільтроване зображення.

2.7. Перетворення Фур'є

Як функцію для частотного аналізу було обрано ПФ. Результат застосування цієї функції полягає в тому, що яка-небудь функція, котра періодично продукує свої величини, можна записати як суму синусів чи/або косинусів різних частот, що помножені на якісь коефіцієнти [28, 29].

Формула (2.11) є ПФ двох змінних:

$$F(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) e^{-i2\pi(ux+vy)} dx dy, \quad (2.11)$$

Для зворотного перетворення використовується формула (2.12):

$$f(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} F(u, v) e^{i2\pi(ux+vy)} du dv, \quad (2.12)$$

де i - уявна одиниця.

2.8. Частотні фільтри, що згладжують

Контури та інші різкі перепади яскравості на зображенні (наприклад, пов'язані з шумом) роблять значний внесок у високочастотну частину його ПФ [28]. Отже, згладжування досягається в частотній області ослабленням високочастотних компонентів певного діапазону Фур'є-образу даного зображення. Фільтрація в частотній ділянці задається рівністю, поданою у формулі (2.13):

$$G(u, v) = H(u, v)F(u, v), \quad (2.13)$$

де $H(u, v)$ – функція згладжування.

2.8.1. Ідеальний фільтр низьких частот

Задається формулою (2.14):

$$H(u, v) = \begin{cases} 1 & \text{при } D(u, v) < D_0, \\ 0 & \text{при } D(u, v) \geq D_0, \end{cases} \quad (2.14)$$

де $D(u, v)$ - відстань від точки (u, v) до початку координат, D_0 - задана дистанція від початку координат.

2.8.2. Фільтр низьких частот Гауса

Визначається формулою (2.15):

$$H(u, v) = e^{-D^2(u,v)/2D_0^2}, \quad (2.15)$$

де $D(u, v)$ - відстань від точки (u, v) до початку координат, D_0 - задана частота зрізу.

2.8.3. Фільтр низьких частот Баттерворта

Задається формулою (2.16):

$$H(u, v) = \frac{1}{1 + (D(u, v)/D_0)^{2n}}, \quad (2.16)$$

де $D(u, v)$ - відстань від точки (u, v) до початку координат, D_0 - задана частота зрізу, n - заданий порядок.

2.9. Висновки до розділу

Наведена коротка інформація про програмні засоби, котрі використані для розробки продукту (Python, Keras, TensorFlow, NumPy, scikit-image), а також середовище розробки PyCharm, набір хмарних служб Google Cloud Platform та база зображень KAIST Scene Text Database.

Описані модулі для навчання ЗНМ, представлена архітектура ПЗ, наведено алгоритм роботи розробки. Описано структуру мережі.

Визначені параметри оцінки під час навчання мережі. Приведено математичний апарат фільтрів для обробки зображень, частотних фільтрів для згладження, виконання ПФ.

РОЗДІЛ 3

ПРАКТИЧНА ЧАСТИНА

Для навчання мережі на сервісі Google Cloud Platform було створено віртуальну машину з 8 віртуальними ЦП та 104 ГБ пам'яті. Платформа ЦП не вказується та вибирається системою автоматично.

3.1 Вибір розмірів вхідних зображень

Підбір розмірів вхідних зображень проводився на базі зображень KAIST Scene Text Database. Для згорткових шарів використовується розмір ядра 33 пікселя, а за допомогою підвбіркових шарів проводиться зменшення зображення в 2 рази. Ці параметри враховувалися для вибору розмірів вхідних зображень для порівняння. Порівняльне навчання проводилося для мережі з 10 шарами, тому розміри вхідних зображень мали без залишку ділитися як на 3, так і на 2^4 . Тому для порівняння були вибрані наступні розміри вхідних зображень: 192x192 пікселя і 384x384 пікселя.

На рис. 3.1 показані графіки залежності власне коефіцієнта Дайса на вибірках (навчальній та валідаційній) від кількості епох. Для зображень розміром 192x192 пікселів було проведено навчання у 110 епох, для зображень розміром 384x384 пікселів – 60 епох.

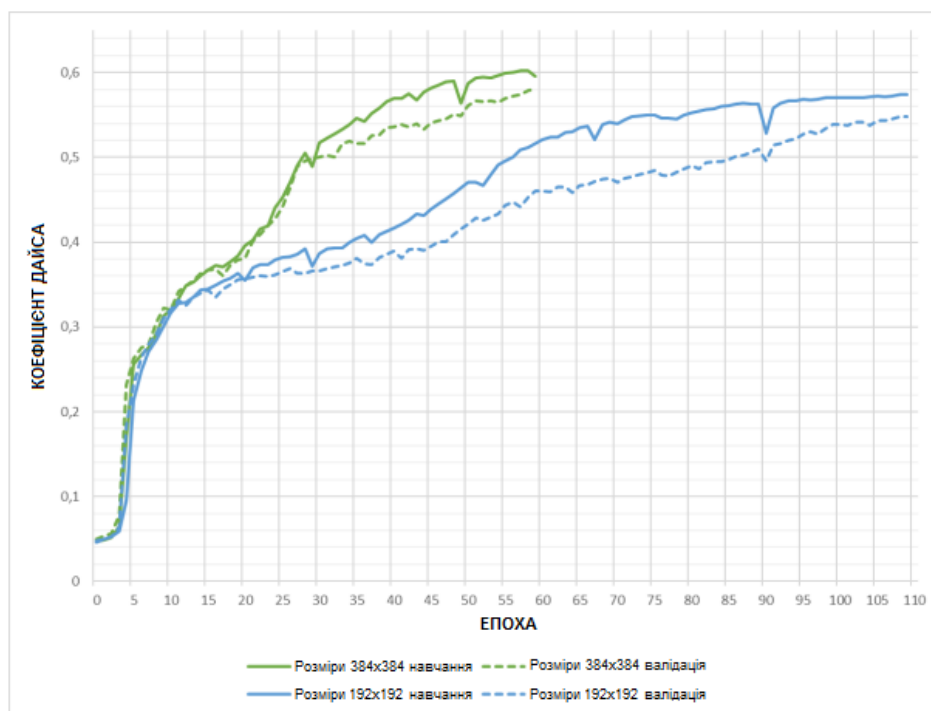


Рисунок 3.1 - Порівняльне навчання фінальної мережі щодо різних розмірів зображень на вході

Подальше навчання не проводилося через те, що середня зміна точності для останніх 10 епох не перевищувала 0,06% і 0,02% відповідно. Отже, такі значення є непорівняними щодо властиво витраченого часу. У таблиці 3.1 подано кінцеві параметри навчання.

Таблиця 3.1

Порівняльне навчання кінцевої мережі різних розмірів вхідних зображень

Розмір вхідного зображення, піксель	192*192	384*384
Кількість епох	110	60
Час навчання однієї епохи, хвилинка	6,93	10,67
Загальний час навчання, година	12,7	10,7
Коефіцієнт Дайса для навчання	0,5744	0,6025
Коефіцієнт Дайса для валідації	0,5483	0,5804
Середня зміна точності останніх 10 епох, %	0,06	0,02

На рис. 3.2 представлені найкращі маски для тестових образів, одержані із мережі максимальної точності. Зверху представлений оригінал зображення, у другому рядку – результат навчання розмірів 192*192 пікселів, знизу – результат навчання розмірів 384*384 пікселів.



Рисунок 3.2 - Найкращі маски для тестових зображень, отримані з мережі максимальної точності (зверху - оригінал зображення, в центрі - результат навчання розмірів 192x192, знизу - результат навчання розмірів 384x384)

Видно, що навчання на зображеннях розмірами 192x192 пікселів краще справляється з детектуванням великих текстових областей, відсіваючи зайві елементи, але не знаходить маленькі області. У той час як навчання на зображеннях розміром 384x384 пікселів помилково детектує елементи фону на зображеннях з великими текстовими областями, але коректно розпізнаються малі області.

На рис. 3.3 представлені найгірші маски для тестових зображень, одержані з мережі максимальної точності.

Видно, що навчання на зображеннях розмірами 192x192 пікселів детектує багато нетекстових областей (наприклад, вікна будівель, просвіти між деревами,

вивіски з малюнками). Навчання на зображеннях розмірами 384x384 пікселів також детектує нетекстові області, однак у меншій кількості.



Рисунок 3.3 - Найгірші маски для тестових зображень, отримані з мережі максимальної точності (зверху - оригінал зображення, у центрі - результат навчання розмірів 192x192, знизу - результат навчання розмірів 384x384)

Таким чином, було прийнято рішення надалі подавати на вхід мережі зображення розміром 384×384 пікселів.

3.2 Підбір кількості шарів

Наступним кроком формування структури кінцевої ЗНМ був вибір кількості шарів. Було проведено порівняльне навчання для мережі з 6, 8 і 10 шарами. Більшість верств не бралось на розгляд через збільшення часу навчання. У всіх наведених випадках на вхід мережі подавалися зображення, приведені до розмірів 384x384 пікселя. Навчання проходило упродовж 10 епох. Графіки на рис. 3.4 відображають залежності коефіцієнта Дайса навчання та валідації власне від кількості епох.

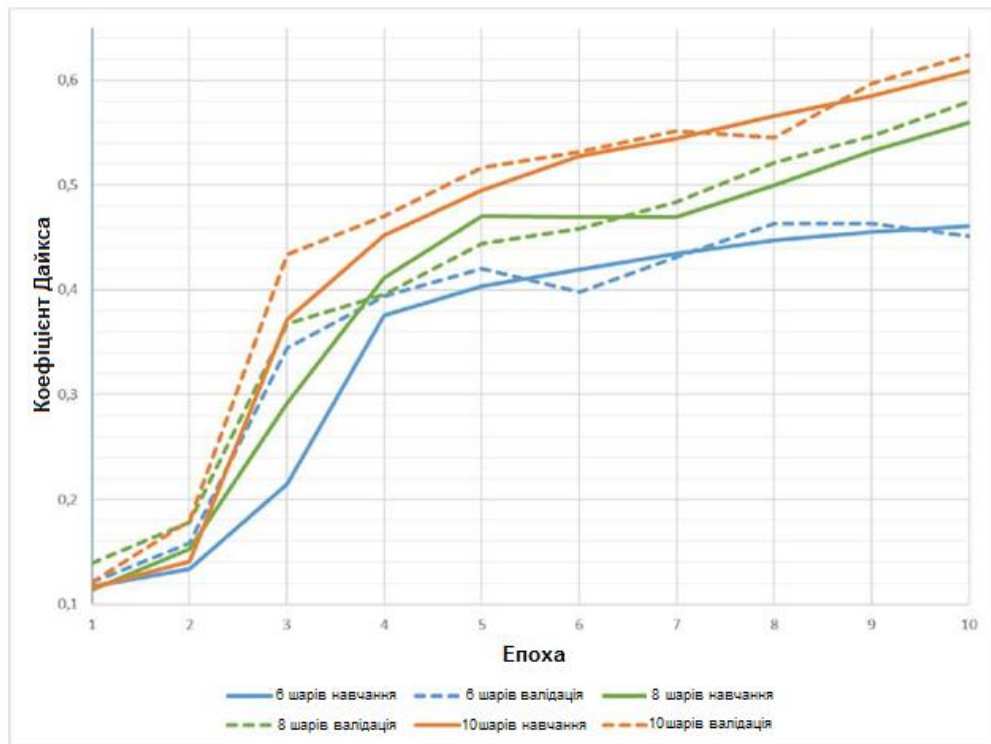


Рисунок 3.4 - Графіки залежності коефіцієнта Дайса навчання та валідації від кількості епох для 6, 8 та 10 шарів

На момент останньої епохи коефіцієнта Дайса для навчання з 6-ю, 8-ю і 10-ма шарами досягає 46,1%, 55,9% і 60,9% відповідно. З рисунка видно, що найкращі результати дає мережа з використанням 10 шарів. Подальше перевірочне навчання не потрібно, так як протягом останніх 5 епох навчання значення коефіцієнта Дайса для мережі з 10 шарами в середньому перевищують значення для мережі з 6 і 8 шарами на 27,7% і 12,1% відповідно.

3.3 Підбір типів згорткових шарів

Було проведено порівняння конфігурацій мережі з використанням шарів Max Pooling і без них. У другому випадку згортка шарів відбувалася за рахунок шарів Convolution2D. Навчання проводилося протягом 10 епох. Графіки на рис. 3.5 демонструють як залежить коефіцієнт Дайса навчання та валідації властиво від кількості епох.

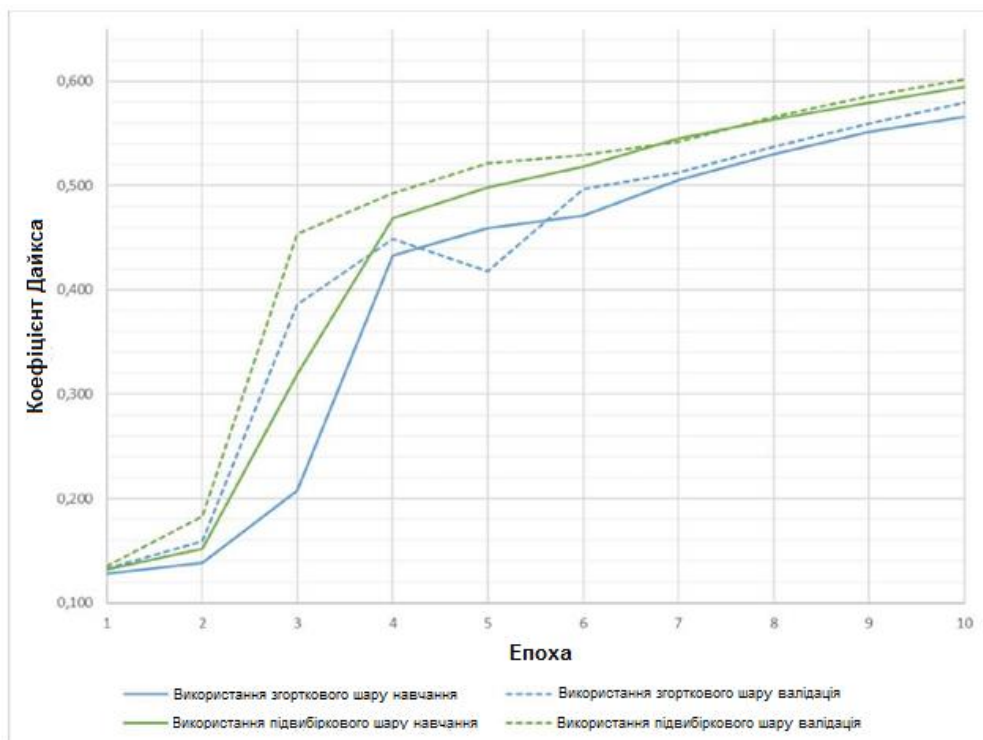


Рисунок 3.5 - Графіки залежності коефіцієнта Дайса навчання та валідації від кількості епох при використанні шарів Max Pooling та без даних шарів

На момент останньої епохи коефіцієнт Дайса навчання при використанні шарів Max Pooling і без цих шарів досягає 59,5% та 56,6% відповідно. Видно, що найкращі результати навчання досягаються під час використання шарів Max Pooling.

3.4 Передобробка вхідних зображень

Було вирішено використовувати передобробку перед подачею зображень як вхідні дані для ЗНМ. Фільтри, що використовуються, описані в пункті 2.6. На рис. 3.6 подано варіанти використання фільтрів для випадково вибраного зображення з бази зображень.



Рисунок 3.6 - Передобробка зображення: а - без фільтрів, б - двосторонній фільтр згладжування з коефіцієнтом 50; фільтр різкості, що виділяє краї, е - двосторонній фільтр згладжування з коефіцієнтом 50 та фільтр виділення країв

Для порівняння передобробки вхідних зображень було здійснено навчання на перевіірочній мережі, описаній у пункті 2.4, на вихідній базі зображень завдовжки 20 епох. Графіки на рис. 3.7 та 3.8 відображають як залежить коефіцієнта Дайса від кількості епох відповідно.

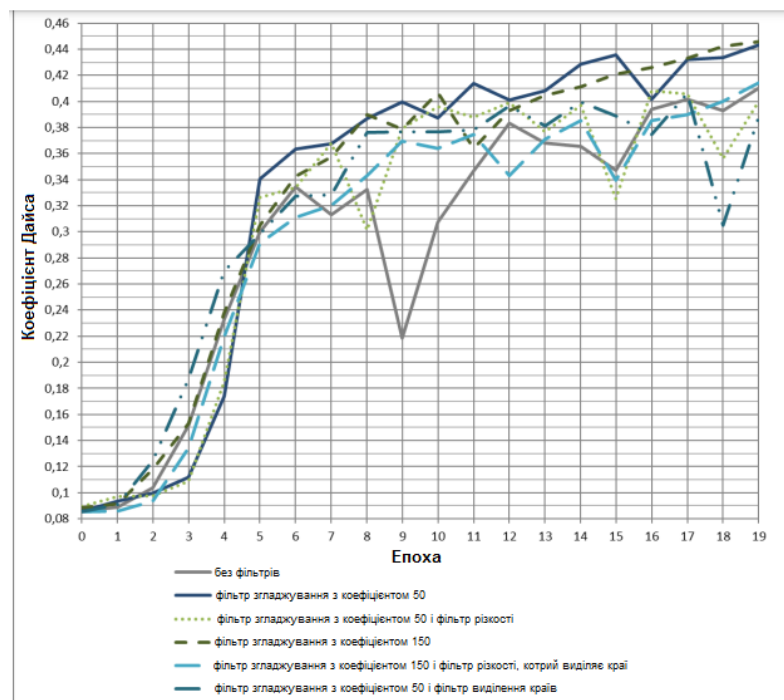


Рисунок 3.7 - Графіки залежності коефіцієнта Дайса навчання від кількості епох для різних фільтрів

Видно, що для навчальної вибірки найбільш точними та стабільними є двосторонній фільтр згладжування з коефіцієнтом 50 і коефіцієнтом 150.

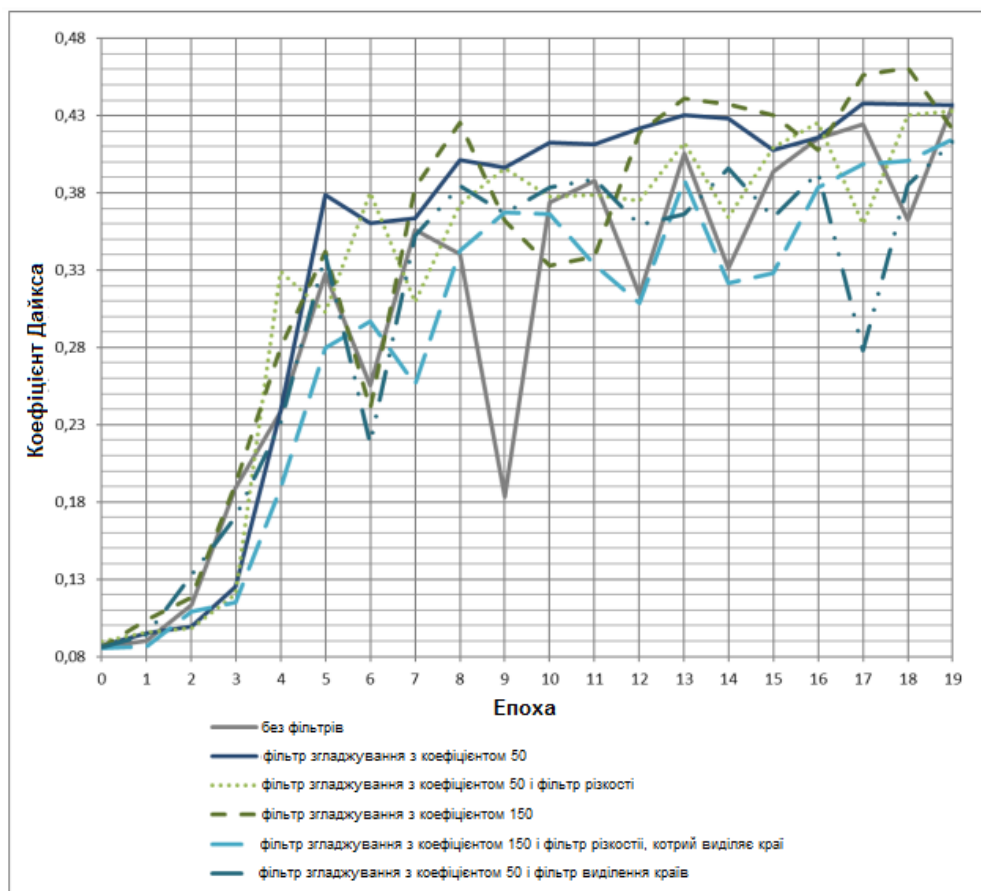


Рисунок 3.8 - Графіки залежності коефіцієнта Дайса валідації від кількості епох для різних фільтрів

Для валідаційної вибірки більшість фільтрів досить непостійні, найстабільнішим є двосторонній фільтр згладжування з коефіцієнтом 50.

Таким чином, було вирішено використовувати двосторонній фільтр згладжування з коефіцієнтом 50 для навчання кінцевої мережі та порівняння його результатів з навчанням без фільтрів. Для цього було використано базу зображень, збільшену за допомогою обертань. На рис. 3.9 представлені графіки залежності коефіцієнта Дайса від кількості епох під час використання фільтра згладжування без нього.

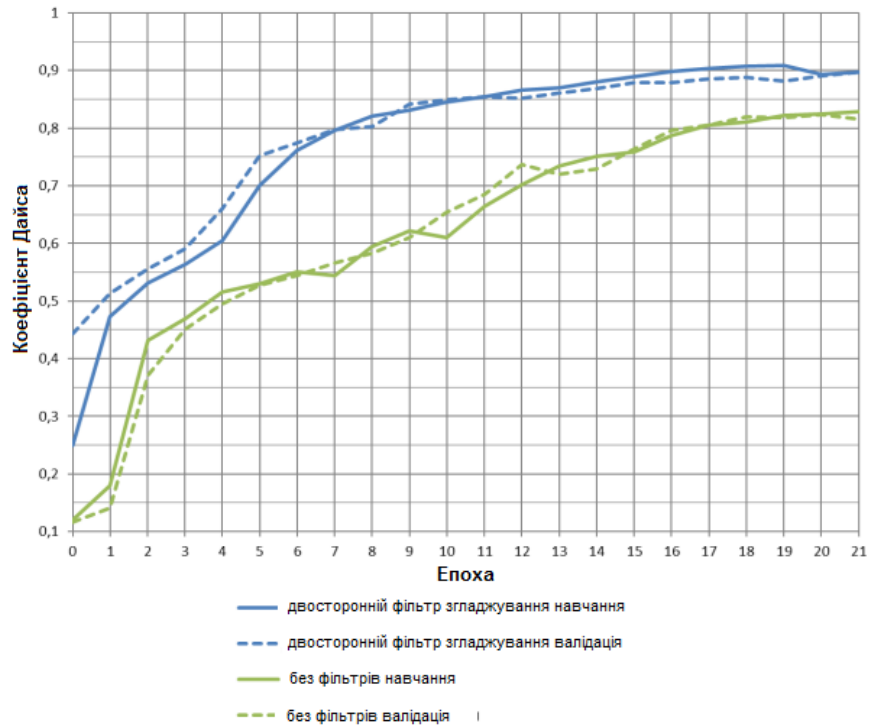


Рисунок 3.9 - Графіки залежності коефіцієнта Дайса навчання та валідації від кількості епох при використанні фільтра згладжування та без нього

На момент останньої епохи коефіцієнти Дайса на навчання, валідації і тестування представлені у таблиці 3.2.

Таблиця 3.2

Порівняльне навчання кінцевої мережі з використанням двостороннього фільтра згладжування для вхідних зображень та без нього

Параметр		З фільтром згладжування	Без фільтру
Коефіцієнт Дайса	Навчання	0,897	0,829
	Валідація	0,897	0,816
	Тестування	0,894	-

Фільтрація розмиттям дозволило прибрати шуми на зображеннях, що полегшило мережі виділення контурів текстових областей.

На рис. 3.10 та 3.11 представлені найкращі та гірші маски для тестових фрагментів, які одержані з мережі найбільшої точності навчання з фільтруванням.



Рисунок 3.10 - Найкращі маски для тестових зображень, отримані з мережі максимальної точності (зверху – оригінал зображення, знизу – результат навчання з фільтрацією)



Рисунок 3.11 - Найгірші маски для тестових зображень, отримані з мережі максимальної точності (зверху – оригінал зображення, знизу – результат навчання з фільтрацією)

Незважаючи на те, що на більшій частині тестових зображень текстові області детектуються правильно, в деяких випадках мережа виділяє такі області, як вікна будівель, просвіти між деревами, вивіски з малюнками. Деякі текстові написи

з тонким шрифтом також погано визначаються за рахунок стиснення зображень. Неоднорідні текстові написи (градієнт, контурні шрифти) також є проблемою.

3.5 Частотний аналіз із використанням перетворення Фур'є

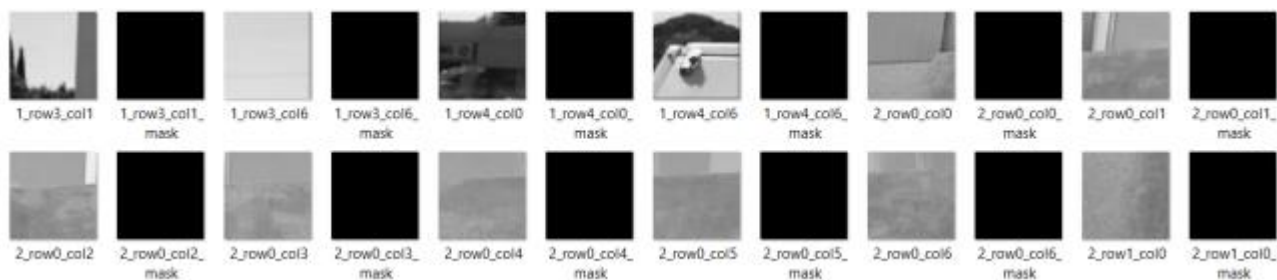
Як альтернативний варіант передобробки вхідних зображень було розглянуто частотний аналіз з ПФ.

3.5.1. Подача на вхід мережі фрагментів зображення після перетворення Фур'є

Для використання ПФ було взято вихідну базу зображень, і зроблено її розрізання на фрагменти розмірами 100×100 пікселів. Отриманий набір зображень був поділений на фрагменти з текстом, що представлені на рис. 3.12, та фрагменти без тексту, наведені на рис. 3.13. Кількість фрагментів обох груп, що використовуються, було приведено до рівного значення для балансування вибірок.



Рисунок 3.12- Фрагменти зображень із текстом



Малюнок 3.13 - Фрагменти зображень без тексту

Для подачі на вхід мережі до всіх зображень з отриманої бази було застосовано ПФ. Для порівняння навчань також були сформовані вибірки із центруванням отриманих перетворень. Еталонним вихідним значенням кожного вхідного зображення є значення 0 чи 1, залежно від відсутності чи наявності текстових областей відповідно.

Приклад результатів навчання мережі завдовжки 10 епох при подачі на вхід мережі фрагментів зображень з ПФ представлений на рис. 3.14.

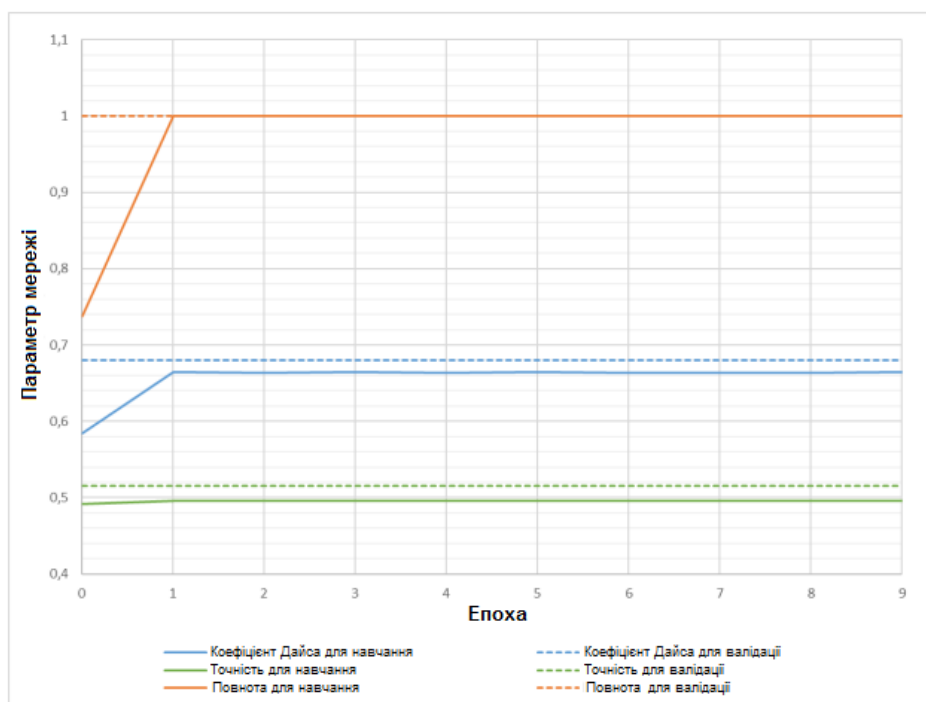


Рисунок 3.14 - Результат навчання мережі під час подачі на вхід мережі фрагментів зображень з ПФ

Видно, що повнота як для навчання, так і для валідації набула значення, що дорівнює 1, що означає передбачення всіх вихідних значень мережі як 1, тобто містять текстові області.

Повторні спроби навчання, а також спроби навчання із центрованими версіями ПФ показували аналогічні результати. Також були показані результати, при яких повнота приймала значення 0, що означає передбачення всіх вихідних значень мережі як 0, тобто текстові області, що не містять.

3.5.2 Додавання шарів із перетворенням Фур'є

Для реалізації додавання шарів з ПФ на початок мережі використовувалися функції бібліотеки tensorflow. На рис. 3.15 показані результати перевірного навчання нейронної мережі для 2 спроб застосування шарів із ПФ. Для порівняння додані результати навчання без цих шарів.

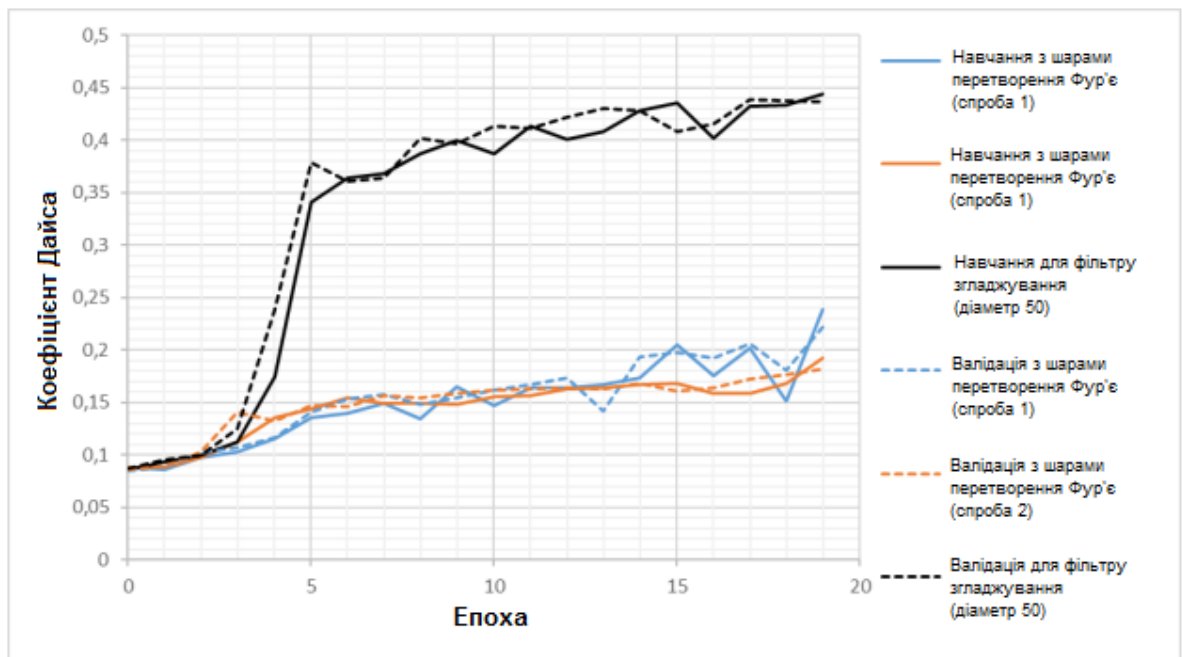


Рисунок 3.15 - Результати перевірного навчання нейронної мережі для 2 спроб застосування шарів із ПФ

Видно, що додавання цих шарів значно понизило значення коефіцієнта Дайса. Тобто навчання мережі не проводиться.

3.5.3 Частотні фільтри, що згладжують, з перетворенням Фур'є

Для використання ПФ як частини фільтрів, що згладжують, були обрані фільтри, перераховані в пункті 2.8. Їхні параметри, подані в табл. 3.3.

Фільтри розмиття та їх параметри

Назва фільтра низьких частот	Значення параметра D_0	Значення параметра n
Ідеальний фільтр	80, 120, 160, 200	-
Фільтр Гаусса	120, 160, 200	-
Фільтр Баттерворта	120, 160, 200	2, 5, 8

На рис. 3.16 представлені приклади фільтрів значення параметра D_0 рівного 120.

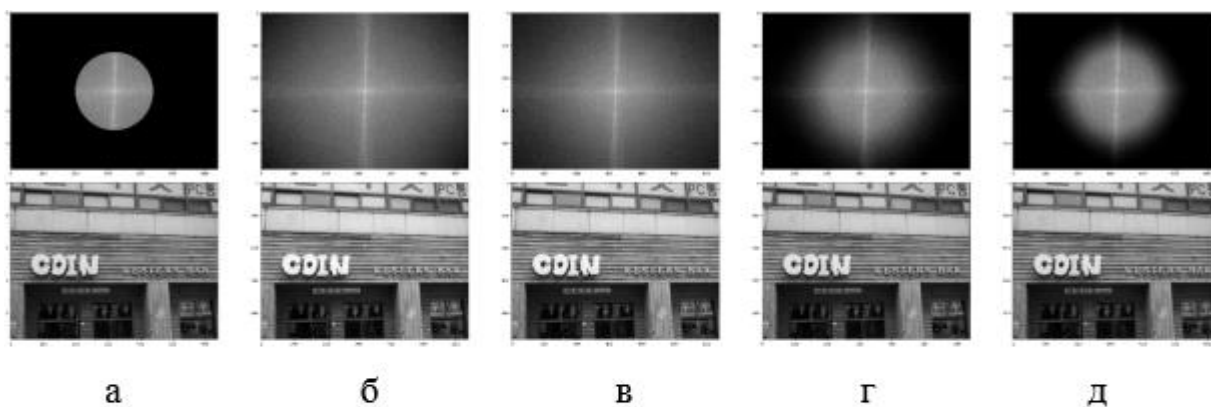


Рисунок 3.16 - Приклади фільтрів та їх застосування для параметра $D_0=120$ (а - ідеальний фільтр низьких частот; б – фільтр низьких частот Гауса, в - фільтр низьких частот Баттерворта ($n=2$), г - фільтр низьких частот Баттерворта ($n=5$), д - фільтр низьких частот Баттерворта ($n=8$))

У результаті було обрано найкращі значення параметрів та проведено їх порівняння з фільтром згладжування. На рис. 3.17 показано порівняння результатів перевірного навчання нейронної мережі для кращих фільтрів із ПФ та фільтра згладжування.

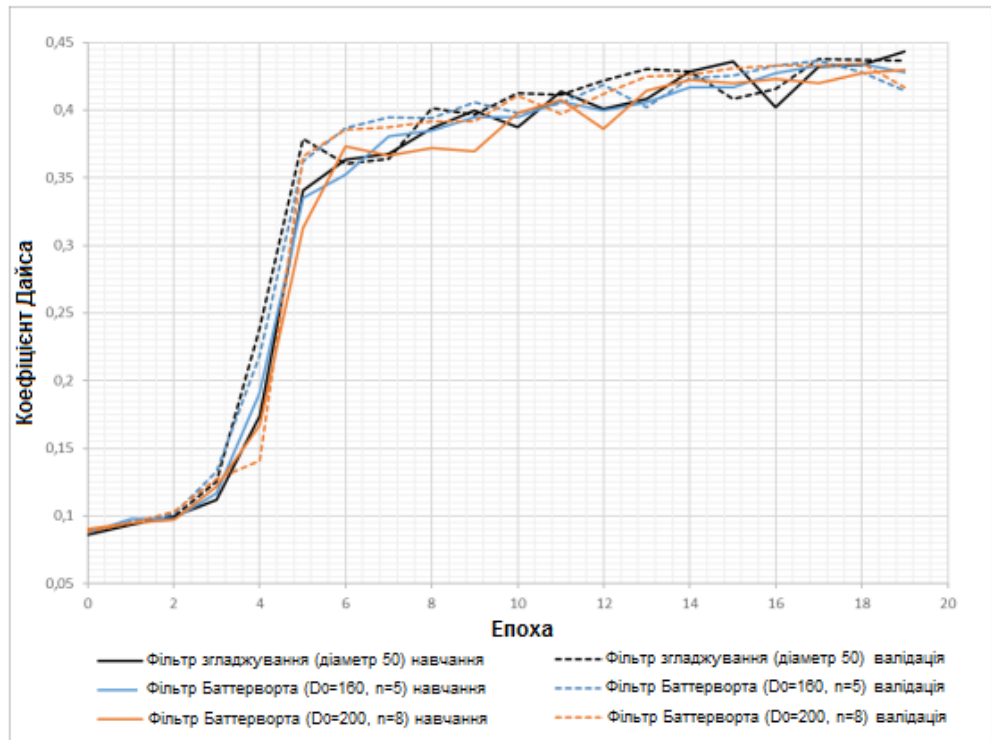


Рисунок 3.17 - Порівняння результатів перевірки мережі для кращих фільтрів з ПФ і фільтра згладжування

Видно, що коефіцієнти Дайса навіть кращих фільтрів з ПФ (фільтр Баттерворта з $D_0=160$ та $n=5$, фільтр Баттерворта з $D_0=200$ та $n=8$) не перевищують значення коефіцієнтів для згладжування фільтра.

3.6 Розбиття та стиснення зображень перед подачею на вхід мережі

Оскільки вихідний розмір зображень дорівнює 640×480 пікселів, а на вхід мережі подаються зображення розміром 384×384 пікселя, було вирішено спробувати розбиття зображень на 4 частини замість зміни розмірів. На рис. 3.18 показаний приклад стиснення та розбиття на 4 частини одного й того ж зображення.



Рисунок 3.18 - Приклад попередньої обробки зображення (ліворуч - зображення, стиснуте до розмірів 384*384 пікселя, праворуч - частини зображення розміром 384*384 пікселя)

Проведено перевіряюче навчання довжиною в 10 епох для 3 варіантів подачі вхідних зображень: стиск, поділ, комбінація стиску та поділ. На рис. 3.19 представлені графіки коефіцієнта Дайса кожному за варіанта.

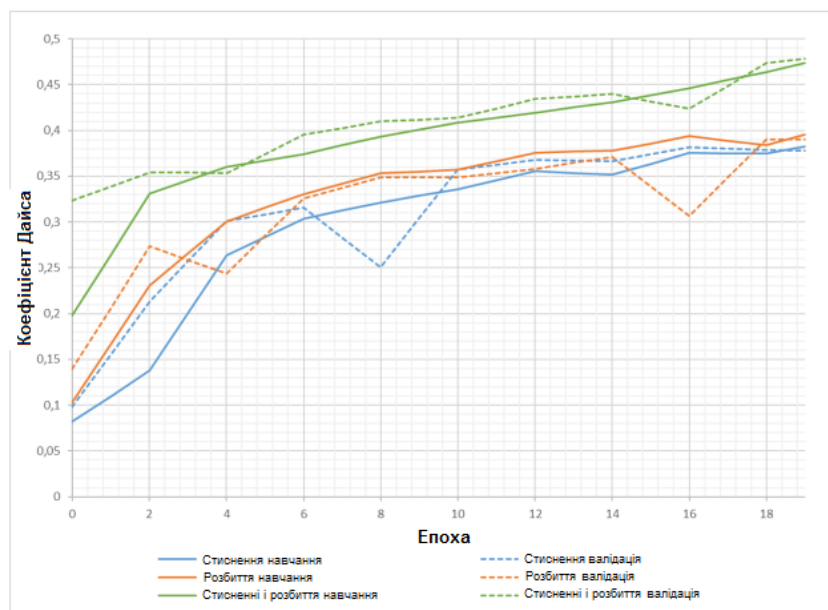


Рисунок 3.19 - Порівняння результатів перевірки мережі для 3 варіантів подачі вхідних зображень: стиснення, розбиття, комбінація стиснення та розбиття

Видно, що найкращий результат показує комбінація стиснення та розбиття, тому для даного варіанту було проведено навчання кінцевої структури мережі на повній базі зображень завдовжки 5 епох. На малюнку 3.20 представлені графіки залежності коефіцієнта Дайса від кількості епох за такого формату вхідних даних.

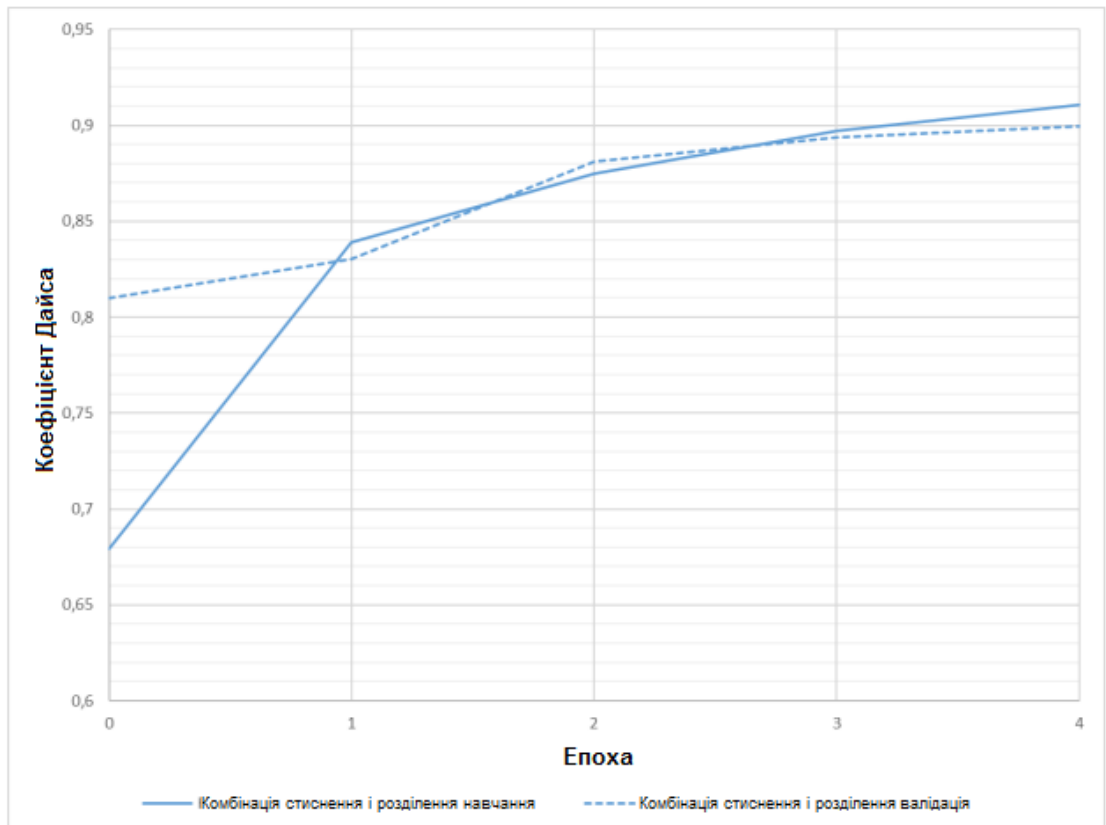


Рисунок 3.20 - Графіки залежності коефіцієнта Дайса навчання та валідації від кількості епох для комбінації стиснення та розбиття вхідних зображень

У таблиці 3.4 представлені параметри оцінки навчання, валідації та тестування. Порівняння здійснюється навчанням на кінцевій мережі при подачі лише стислих зображень.

**Порівняння навчання кінцевої мережі зі стисненням вхідних зображень
та з комбінацією стиснення та розбиття**

Параметр	Тип	Кінцева мережа, база зображень з поворотами та стисненням		Кінцева мережа, база зображень з поворотами, стисненням та розбиттям
Кількість епох навчання		22	5	5
Повний час навчання, година		59	13,5	60
Коефіцієнт Дайса	Навчання	0,897	0,605	0,911
	Валідація	0,897	0,660	0,900
	Тестування	0,894	-	0,880
Точність	Навчання	0,948	0,584	0,915
	Валідація	0,945	0,813	0,917
	Тестування	0,946	-	0,902
Повнота	Навчання	0,807	0,653	0,902
	Валідація	0,823	0,681	0,882
	Тестування	0,822	-	0,864
F-міра ($\beta=1$)	Навчання	0,8718	0,631	0,908
	Валідація	0,8798	0,662	0,900
	Тестування	0,918	-	0,883

Результати показують, що за однакову кількість епох комбінація стиснення та поділу суттєво збільшує значення параметрів оцінки мережі як для навчальної, так і для валідаційної вибірки. Порівняння з результатами передбачень для тестової вибірки не може бути здійснене через проведення тестування після завершення навчання мережі.

При порівнянні з результатами навчання за рівного часу навчання мереж видно, що комбінація стиснення та розбиття дозволяє отримати кращі результати для коефіцієнта Дайса, повноти та F-міри. Точність поступається на 3,3%, 2,9% і 4,4% для навчальної, валідаційної та тестової вибірок відповідно.

Високі значення повноти вказують на те, що більшість текстових областей дійсно визначається алгоритмом; в той час як високі значення точності вказують на те, що більшість виділених областей дійсно є текстовими. Розгляд точності та

повноти для результатів навчання за рівного часу показує, що з навчання на стислих зображеннях дані значення різняться у середньому 12,9%, а навчання на комбінації стислих і розділених - на 2,9%. Це означає, що під час навчання на стислих зображеннях мережа не здатна виділити всі текстові області.

На рис. 3.21 та 3.22 представлені кращі та найгірші варіанти передбачень відповідно.



Рисунок 3.21 – Найкращі маски для тестових зображень, отримані з мережі максимальної точності



Рисунок 3.22 – Найгірші маски для тестових зображень, отримані з мережі максимальної точності

На більшості тестових зображень текстові області детектуються правильно.

Мережа добре справляється з різними типами та розмірами шрифтів, а також з поворотами. Складними випадками визначення є великі написи: виділяються додаткові області навколо чи, навпаки, опускаються області всередині символів.

3.7 Порівняння з іншими роботами

Проведено порівняння результатів тестування поточного алгоритму з результатами, отриманими іншими авторами під час використання бази зображень KAIST Scene Text Database. Зважаючи на відсутність результатів робіт інших авторів, які використовують нейромережеві підходи, були обрані роботи з іншими методами.

Список методів, наведених у цих роботах, вказаний нижче:

1. Метод, що ґрунтується на квадратичній дискримінантній функції [29];
2. Метод найменшого розрізу графа [29];
3. Комбінація методу квадратичної дискримінантної функції та найменшого розрізу графа [29];
4. Метод визначення порогових значень Otsu [29];
5. Метод визначення порогових значень Niblack [29];
6. Метод на основі кластеризації К-середніх [29];
7. Метод MSER [30];
8. Модифікований метод на основі кластеризації К-середніх [30];
9. Метод детектування спеціальних областей (Blob) [30];
10. Метод, що ґрунтується на Марківській моделі випадкового поля [30];
11. Комбінація методу MSER та алгоритму дендрограм одинарної кластеризації [31].

Тестування робіт 1-6 проводилося лише з англійської частини бази зображень, тоді як роботи 7-11 розглядають повний набір зображень. Подальше звернення до робіт провадиться за назвами їх методів.

У табл. 3.5 представлені параметри оцінки тестування всім перерахованих

методів, і навіть для методу, поданого у цій роботі. Як параметри порівняння були використані точність, повнота і F- міра.

Таблиця 3.5

Порівняння результатів тестування методів на базі зображень KAIST Scene Text Database

База зображень	Метод	Точність	Повнота	F -міра
Англійська частина	Метод, заснований на квадратичній дискримінантній функції	0,789	0,925	0,851
	Метод найменшого розрізу графа	0,786	0,918	0,847
	Комбінація методу квадратичної дискримінантної функції та найменшого розрізу графа	0,828	0,893	0,860
	Метод визначення порогових значень Otsu	0,747	0,905	0,818
	Метод визначення порога Niblack	0,689	0,909	0,784
	Метод на основі кластеризації K-середніх	0,763	0,914	0,821
Повна	Метод максимально стабільних екстремальних областей	0,924	0,341	0,480
	Модифікований метод на основі кластеризації K -середніх	0,690	0,600	0,640
	Метод детектування спеціальних областей (Blob)	0,879	0,489	0,593
	Метод, що ґрунтується на Марківській моделі випадкового	0,697	0,291	0,376
	Комбінація методу максимально стабільних екстремальних областей та алгоритму дендрограм одинарної	0,670	0,890	0,760
	Поточний метод	0,902	0,864	0,883

Як видно з таблиці вище максимальне значення точності має робота із

використанням методу MSER. Отримане значення перевищує точність методу, поданого у цій роботі, на 2,5%. Однак повнота і F-міра даного методу показують дуже низькі результати, отже, велика кількість текстових областей не виявляється. Решта інших методів показують точність тестування нижче, ніж поточний метод.

Всі методи, тестування яких проходить на англійській частині бази зображень, мають більш високі значення повноти. При цьому різниця з поточним методом не перевищує 7% для найкращого результату. Це може бути обумовлено більш простим набором даних для тестування. Тим не менш, ці ж методи показують нижчі варіанти для точності, що означає наявність більшої кількості помилкових спрацьовувань на нетекстових областях.

Серед методів, тестування яких відбувається на повній базі зображень, лише метод комбінації використання MSER та алгоритму дендрограм одинарної кластеризації показує значення вище поточного. Але в той же час, точність даного методу є найнижчою зі всіх представлених значень.

Поточний спосіб показує найбільше значення F-міри. У середньому методи, що проводять тестування на англійській частині бази зображень, показують значення на 6% менше; для повної бази кількість відсотків збільшується до 35,5%.

Таким чином, метод, представлений у цій роботі, не має найкращих значень точності та повноти при їх роздільному розгляді, але при цьому перевершує всі інші за значенням F-міри. Це означає, що жоден спосіб з інших робіт не перевищує поточний за всіма параметрами.

Високі значення відразу всіх параметрів показують, що навчання відбувалося більш збалансовано, тобто більшість текстових областей дійсно визначається алгоритмом, і більшість виділених областей дійсно є текстовими. Таким чином, запропонований алгоритм перевершує алгоритми інших авторів за F-мірою, що свідчить про кращу його збалансованість між вірними та помилковими спрацьовуваннями, і є порівнянним з кращими результатами алгоритмів за точністю та повнотою.

3.8. Висновок до розділу

З метою навчання мережі на Google Cloud Platform було побудовано віртуальну машину. Проведено підбір розмірів вхідних зображень із бази KAIST Scene Text Database, виконано порівняльне навчання кінцевої мережі різних розмірів вхідних зображень, в результаті оптимальним визначено розмір 384 x 384. Було проведено порівняльне навчання для мережі з 6, 8 і 10 шарами. Оптимальним є застосування 10 шарів.

Для передобробки вхідних зображень прийнято рішення застосовувати двосторонній фільтр згладжування з коефіцієнтом 50 для навчання кінцевої мережі.

Проведено частотний аналіз з використанням ПФ, виконано розбиття та стиснення зображень перед подачею на вхід мережі.

Проведено порівняння розробленого нейромережевого алгоритму із результатами, наведеними в роботах інших авторів.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

Метою кваліфікаційної роботи магістра є розробка нейромережевого алгоритму розпізнавання текстових областей зображення реальних сцен. Оскільки, проведення робіт з розробки та використання алгоритму передбачає застосування комп'ютерної техніки, зокрема ПК та периферійних пристроїв, то обов'язковим є дотримання вимог з охорони праці і техніки безпеки.

Для ефективної і безпечної роботи колективу працівників з розробки ПЗ комп'ютерних систем, в тому числі і фахівців з підвищення ефективності контролю доступу в приміщення, необхідно організувати безпечні умови праці. При цьому керівник організації несе безпосередню відповідальність за порушення нормативно-правових актів з охорони праці [33]. Окрім цього, на робочих місцях працівників необхідно забезпечити дотримання вимог, затверджених Наказом Мінсоцполітики від 14.02.2018 за № 207 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Згідно Вимог приміщення, де розміщені робочі місця операторів, крім приміщень, у яких розміщені робочі місця операторів великих ЕОМ загального призначення (сервер), мають бути оснащені системою автоматичної пожежної сигналізації відповідно до цих вимог;

– переліку однотипних за призначенням об'єктів, які підлягають обладнанню автоматичними установками пожежогасіння та пожежної сигналізації, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 22.08.2005 N 161, зареєстрованого в Міністерстві юстиції України 05.09.2005 за N 990/11270 (НАПБ Б.06.004-2005);

– Державних будівельних норм "Інженерне обладнання будинків і споруд. Пожежна автоматика будинків і споруд", затверджених наказом Держбуду України від 28.10.98 N 247 (далі - ДБН В.2.5-56:2014, з димовими пожежними сповіщувачами та переносними вуглекислотними вогнегасниками.

В інших приміщеннях допускається встановлювати теплові пожежні сповіщувачі. Приміщення, де розміщені робочі місця операторів, мають бути оснащені вогнегасниками, кількість яких визначається згідно з вимогами ДСТУ 4297:2004 «Пожежна техніка. Технічне обслуговування вогнегасників». Загальні технічні вимоги і з урахуванням граничнодопустимих концентрацій вогнегасної рідини відповідно до вимог НАПБ А.01.001-2014. Приміщення, в яких розміщуються робочі місця операторів сервера загального призначення, обладнуються системою автоматичної пожежної сигналізації та засобами пожежогасіння відповідно до вимог ДБН В.2.5-56:2014, ДБН В.2.5-56:2010, НАПБ А.01.001-2014 і вимог нормативно-технічної та експлуатаційної документації виробника. Проходи до засобів пожежогасіння мають бути вільними.

Лінія електромережі для живлення комп'ютера та периферійних пристроїв повинні бути виконаними як окрема групова трипровідна мережа шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Не допускається використовувати нульовий робочий провідник як нульовий захисний провідник. Нульовий захисний провідник прокладається від стійки групового розподільного щита, розподільного пункту до розеток електроживлення. Не допускається підключати на щиті до одного контактного затискача нульовий робочий та нульовий захисний провідники.

Площа перерізу нульового робочого та нульового захисного провідника в груповій трипровідній мережі має бути не менше площі перерізу фазового провідника. Усі провідники мають відповідати номінальним параметрам мережі та навантаження, умовам навколишнього середовища, умовам розподілу провідників,

температурному режиму та типам апаратури захисту, вимогам НПАОП 40.1-1.01-97.

У приміщенні, де одночасно експлуатуються понад п'ять комп'ютерів, на помітному, доступному місці встановлюється аварійний резервний вимикач, який може повністю вимкнути електричне живлення приміщення, крім освітлення. Комп'ютери повинні підключатися до електромережі тільки за допомогою справних штепсельних з'єднань і електророзеток заводського виготовлення.

У штепсельних з'єднаннях та електророзетках, крім контактів фазового та нульового робочого провідників, мають бути спеціальні контакти для підключення нульового захисного провідника. Їхня конструкція має бути такою, щоб приєднання нульового захисного провідника відбувалося раніше, ніж приєднання фазового та нульового робочого провідників. Порядок роз'єднання при відключенні має бути зворотним. Не допускається підключати комп'ютери до звичайної двопровідної електромережі, в тому числі – з використанням перехідних пристроїв. Електромережі штепсельних з'єднань та електророзеток для живлення комп'ютерної техніки повинні бути виконаними за магістральною схемою, по 3-6 з'єднань або електророзеток в одному колі. Штепсельні з'єднання та електророзетки для напруги 12 В та 42 В за своєю конструкцією мають відрізнятися від штепсельних з'єднань для напруги 127 В та 220 В. Штепсельні з'єднання та електророзетки, розраховані на напругу 12 В та 42 В, мають візуально (за кольором) відрізнятися від кольору штепсельних з'єднань, розрахованих на напругу 127 В та 220 В.

При підвищенні ефективності контролю доступу в приміщення, де для забезпечення безпеки мешканців, співробітників і збереження майна використовуються ДС, важливим, з точки зору охорони праці, є забезпечення достатньої величини природного та штучного освітлення, які визначені у НПАОП 0.00-7.15-18. Організація робочого місця фахівця із дослідження методів та програмно-апаратних засобів оптимізаційних процесів на основі ГА повинна забезпечувати відповідність усіх елементів робочого місця та їх розташування

ергономічним вимогам ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги». Відстань від екрана до ока фахівців, які працюють за комп'ютером визначається згідно з вимогами ДСанПіН 3.3.2.007-98.

Розміщення принтера або іншого пристрою введення-виведення інформації на робочому місці має забезпечувати добру видимість екрана комп'ютера, зручність ручного керування пристроєм введення-виведення інформації в зоні досяжності моторного поля згідно з вимогами ДСанПіН 3.3.2.007-98.

Таким чином, у результаті аналізу вимог щодо охорони праці користувачів комп'ютерів, визначено особливості організації робочих місць, вимог з електробезпеки, природного та штучного освітлення для ефективної і безпечної роботи фахівців з дослідження нейромережових алгоритмів розпізнавання текстових областей зображення реальних сцен.

4.2. Комп'ютерне забезпечення процесу оцінки радіаційної та хімічної обстановки

Екологічне співтовариство розробило сімейство інструментів комплексної екологічної оцінки. Програмне забезпечення і послуги (ESS), комерційна група ПАСА, включаючи AirWare (для повітряних проблеми якості), WaterWare (для якості води), CityWare (якість повітря і води в контексті великих міст) і EIAxpert (для надання допомоги із загальним впливом на навколишнє середовище). Функціональність в цілому схожа на RAISON, хоча з великим акцентом на моделювання і меншим акцентом на керування даними. Знову ж таки, інструменти ESS розроблені як модульні набори інструментів (доступні спеціальні системи для вирішення конкретних завдань). Компоненти включають стандартні імітаційні моделі, включаючи моделі ISC і PBM Агентства з охорони навколишнього середовища США, управління даними, в тому числі ГІС, аналіз даних (наприклад, аналіз часових рядів даних спостережень), візуалізація, а також оптимізація [34].

Іноді немає готових моделей, придатних для конкретного застосування, але тягар розробки нової програми на Фортрані або С / С ++ є надмірним. Розробка моделі оточення може відносно легко реалізувати власні моделі комп'ютерів і не турбуватися про включення процедур для вирішення рівнянь, візуалізації і т. д. Як правило, за допомогою цих інструментів користувач просто повинен вказати свою модель, використовуючи або математичні рівняння, або спеціальні графічні символи або значки, які безпосередньо представляють поведінку системи.

На даний момент є розроблені моделі комп'ютерного забезпечення процесу для оцінки радіаційної та хімічної обстановки.

GEMS – це система на основі моделей, яка підтримує оцінки схильності і ризику, надаючи доступ до одиночних і мультимедійних моделям експозиції, фізико-хімічні властивості методи оцінки, статистичний аналіз, графічні та картографічні програми з відповідними даними на навколишнє середовище, джерела, рецептори і популяції. У розробці з 1981 року, GEMS надає аналітикам 84 84 інтерактивний, легко досліджуваний інтерфейс для різних моделей, програм і даних, які необхідні для оцінки хімічного впливу і ризику [34].

HSPF – це комплексний пакет для моделювання кількості і якості стоків з багатоцільових водозборів і процесів радіації, що відбуваються в потоках або повністю змішаних озерах. Це дозволяє інтегроване моделювання землі і ґрунту, процесів забруднення при гідравлічній і осадово-хімічній взаємодії. Результатом моделювання є тимчасові дані витрати стоку, концентрація поживних речовин і пестицидів, а також дані кількості і якості води в будь-якій точці водозбору. Алгоритми якості води включають динаміку BOD / DO, вуглець, азот і фосфор. Процеси трансформації, які включені в модель це: гідроліз, фотоліз, окислення, випаровування, сорбція і біодеградація. Вторинні або «дочірні» хімічні речовини також моделюються.

Вимоги до даних для моделі можуть бути досить широкими в залежності від конкретного застосування.

Модель MMSOILS – це методологія оцінки впливу на людину і ризику для здоров'я, пов'язаних з викидами забруднень з небезпечних відходів. Мультимедійна модель, що стосується перенесення хімічної речовини в ґрунтові води, поверхневі води, атмосферу і накопичення в їжі. Шляхи впливу на людину, які розглянуті в методології включають: потрапляння в ґрунт, вдихання летких речовин в повітря і тверді частинки, шкірний контакт, прийом питної води і т.д. Ризик, пов'язаний із загальною дозою опромінення, розраховується на основі хімічної токсичності [34].

4.3. Висновки до розділу

В цьому розділі проаналізовано важливі питання охорони праці та безпеки в надзвичайних ситуаціях, висвітлено питання комп'ютерного забезпечення процесу оцінки радіаційної та хімічної обстановки.

ВИСНОВКИ

У ході виконання випускної кваліфікаційної роботи був проведений аналіз предметної області, що включає розгляд існуючих методів виявлення текстових областей на зображеннях реальних сцен. В результаті опрацювання аналітичної частини в якості класифікатора була обрана нейронна мережа архітектури U-Net .

Як базу зображень було обрано набір даних KAIST Scene Text Database, для якого було проведено збільшення кількості зображень за рахунок застосування поворотів, стиснення та поділу.

Було розроблено нейромережевий алгоритм розпізнавання написів на зображеннях реальних сцен, для якого проводився підбір таких параметрів ШНМ як розміри вхідних зображень, кількість і типи шарів, що використовуються. Було проведено проектування ПЗ на підставі розробленого алгоритму, а також його подальша реалізація та тестування.

В результаті кінцевого навчання нейромережевого алгоритму вдалося досягти високої точності: 91,1% для навчальної вибірки, 90,0% для валідаційної та 88,0% для тестової.

Було проведено порівняння з роботами, що проводили тестування на базі зображень KAIST Scene Text Database: повний набір даних та лише англійська частина. У результаті порівняння запропонований у роботі алгоритм показав найкращі результати.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Text line segmentation in historical document images using an adaptive U-Net architecture / O. Mechi, M. Mehri, R. Ingold, and N. Essoukri Ben Amara // *2019 International Conference on Document Analysis and Recognition (ICDAR)*. 2019. P.369-374.
2. A new U-Net based license plate enhancement model in night and day images / P. N. Chowdhury, P. Shivakumara, R. Raghavendra, U. Pal, T. Lu and M. Blumenstein // *The 5th Asian Conference on Pattern Recognition*. 2019. P.749-763.
3. Дуже просто про геокодування. URL: <http://www.50northspatial.org/ua/introduction-to-geocoding/> (дата звертання 07.12.2023).
4. Басюк Т.М. та ін. Машинне навчання: Навчальний посібник Львів: Видавництво «Новий Світ - 2000», 2021. 315 с.
5. Кононова К. Ю. Машинне навчання: методи та моделі. Харків: ХНУ імені В. Н. Каразіна, 2020. 301 с.
6. Robust wide-baseline stereo from maximally stable extremal regions / J. Matas, O. Chum, M. Urban, and T. Pajdla // *Image and Vision Computing*. 2004. Vol. 22, №10. P.761-767.
7. Neumann L. Real-time scene text localization and recognition / L. Neumann, J. Matas // *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2012. P.3538-3545.
8. Epshtein B. Detecting text in natural scenes with stroke width transform / B. Epshtein, E. Ofek, Y. Wexler // *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2010. P.2963-2970.
9. Ahmed N. Discrete Cosine Transform / N. Ahmed, T. Natarajan, K. R. Rao // *IEEE Transactions on Computers*. 1974. Vol. C-23, №1. P.90-93.
10. Zhong Y. Automatic caption localization in compressed video / Y. Zhong, H. Zhang and A. K. Jain // *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 2000. Vol. 22, №4. P.385-392.

11. Dalal N. Histograms of Oriented Gradients for Human Detection / N. Dalal, B. Triggs // *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* 2005. Vol. 1. P.886-893.
12. Czarnek N. Physically Motivated Feature Development for Machine Learning Applications // Department of Electrical and Computer Engineering Duke University. 2017.
13. Viola P. Rapid Object Detection using a Boosted Cascade of Simple Features / P. Viola, M. Jones // *Proceedings of the 2001 IEEE Computer Society 88 Conference on Computer Vision and Pattern Recognition*. 2001. Vol. 1. P.511- 518.
14. Ghorbel A. Generalized Haar-like filters for document analysis: application to word spotting and text extraction from comics // Document and Text Processing. Université de La Rochelle. 2016.
15. Chen X. Detecting and reading text in natural scenes / X. Chen and A. L. Yuille // *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2004. Vol. 2. P.366-373.
16. Schmidhuber J. Deep learning in neural networks: An overview // *Neural Networks*. 2015. Vol. 61. P.85-117.
17. Bengio, Y. Deep Learning / Y. Bengio, Y. LeCun, G. Hinton // *Nature*. 2015. Vol. 521. P. 436–444.
18. Goodfellow I. J. Multi-digit number recognition from street view imagery using deep convolutional neural networks / I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, V. Shet // *Proceedings of the International Conference on Learning Representations (ICLR)*. 2014.
19. Zhang, Z. Multi-oriented text detection with fully convolutional networks. / Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, X. Bai // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016. P.4159-4167.
20. Ronneberger O. U-net: Convolutional networks for biomedical image segmentation / O. Ronneberger, P. Fischer, T. Brox // *Medical Image Computing and Computer-Assisted Intervention*. 2015. Vol. 9351. P.234-241.

21. Bezmaternykh P.V. U-Net-bin: hacking the document image binarization contest / P.V. Bezmaternykh, D.A. Ilin, D.P. Nikolaev // *Computer Optics*. 2019. Vol. 43 №5. P.825-832.

22. Нейроподібні методи, алгоритми та структури обробки зображень у реальному часі: монографія / Ю. М. Рашкевич, Р. О. Ткаченко, І. Г. Цмоць, Д. Д. Пелешко ; НУ Львівська політехніка. Львів: Вид-во Львів. політехніки, 2014. 256 с. : іл.

23. Чарковський Д.Р., Стадник Н.Б. Методи детектування текстових областей на зображеннях. *Інформаційні моделі, системи та технології: Праці XI наук.-техн. конф.* (Тернопіль, 13-14 грудня 2023 р.), Тернопіль, 2023. С. 192.

24. Machine Learning with Python. URL: <https://www.freecodecamp.org/learn/machine-learning-with-python/> (дата звертання 12.12.2023).

25. Васильєв О. Програмування мовою Python. К. : Навчальна книга Богдан, 2019. 418 с.

26. Keras: The Python Deep Learning library URL: <https://keras.io/> (дата звертання 24.11.23).

27. NumPy. URL: <http://www.numpy.org/> (дата звертання 24.11.23).

28. Bai B. A Seed-Based Segmentation Method for Scene Text Extraction / B. Bai, F. Yin and C. L. Liu // *11th IAPR International Workshop on Document Analysis Systems*. 2014. P.262-266.

29. Agrawal A. Enhanced characterness for text detection in the wild / A. Agrawal, P. Mukherjee, S. Srivastava, B. Lall // *Proceedings of 2nd International Conference on Computer Vision & Image Processing*. 2018. P. 359-369.

30. Gomez L. A fast hierarchical method for multi-script and arbitrary oriented scene text extraction / L. Gomez, D. Karatzas // *International Journal on Document Analysis and Recognition*. 2016. Vol. 19, №4. P.335-349.

31. Lee J. Page segmentation using a convolutional neural network with trainable co-occurrence features. / J. Lee, H. Hayashi, W. Ohyama, S. Uchida // *2019 International Conference on Document Analysis and Recognition (ICDAR)*. 2019. P.1023-1028.

32. Wick C. Fully convolutional neural networks for page segmentation of historical document images / C. Wick, F. Puppe // *13th IAPR International Workshop on Document Analysis Systems (DAS)*. 2008. P. 287-292.

33. Толок А.О. Крюковська О.А. Безпека життєдіяльності: Навч. посібник. 2011. 215 с.

34. Зеркалов Д.В. Охорона праці в галузі: Загальні вимоги. Навчальний посібник. К.: Основа. 2011. 551 с.

ДОДАТОК А

Тези конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ

МАТЕРІАЛИ

XI НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

**«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



13-14 грудня 2023 року

ТЕРНОПІЛЬ
2023

Б.С.Таранія, О.Р.Тєбрії РОЗРОБКА СКРИПТУ ДЛЯ ІНТЕГРАЦІЇ СКЛАДОВИХ ЕЛЕМЕНТІВ ІНЕРЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ «ARDUPILOT» B.S.Tarania, O.R.Tsibriv DEVELOPMENT OF A SCRIPT FOR INTEGRATION COMPONENT ELEMENTS OF THE INERTIAL SYSTEM USING «ARDUPILOT» TECHNOLOGIES	182
М.В.Томонський ПОРІВНЯННЯ БАЗ ДАНИХ MONODB ТА POSTGRESQL В КОНЕКТІ РОЗРОБКИ СУЧАСНИХ ВЕБ-ЗАСТОСУНКІВ M.V.Tomonyski COMPARISON OF MONGODB AND POSTGRESQL DATABASES IN THE CONTEXT OF MODER WEB APPLICATIONS DEVELOPMENT	184
В.Тумошчук, Т.Чеж, А.Фіалка, Н.Луцьк МЕТОДИ ВІРТУАЛІЗАЦІЇ В КЛАСТЕРАХ ВИСОКОЇ ДОСТУПНОСТІ V. Tymoshchuk, T. Chezh, A. Fialka, N. Lutsyk METHODS OF VIRTUALIZATION IN HIGH AVAILABILITY CLUSTERS	186
В.Тумошчук, В.Васильшин, І.Мудрий, Н.Луцьк ОГЛЯД ТА ПОРІВНЯННЯ ПРОТОКОЛІВ ПЕРЕДАЧІ ІНФОРМАЦІЇ В ІОТ V. Tymoshchuk, V. Vasylyshyn, I. Mudryi, N. Lutsyk OVERVIEW AND COMPARISON OF INFORMATION TRANSFER PROTOCOLS IN IOT	188
Ткачук Р.М., Ткачук Р.А. ЗАБЕЗПЕЧЕННЯ ІНДИВІДУАЛЬНОГО ПІДБОРУ КЛАПАНІВ ДЛЯ ВІВІВІДУ ВНУТРІШНЬООЧНОЇ РІДНИНИ ПРИ ЛІКУВАННІ ГЛАУКОМИ Tkachuk R.M., Tkachuk R.A. PROVISION OF INDIVIDUAL SELECTION OF VALVES FOR THE REMOVAL OF INTRAOCULAR FLUID IN THE TREATMENT OF GLAUCOMA	189
Д.А.Урбан СУЧАСНІ МОДЕЛІ ТА АЛГОРИТМИ ДЛЯ АВТОМАТИЗОВАНОГО АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ДИНАМІКИ СОЦІАЛЬНИХ МЕДІА D. A. Urban MODERN MODELS AND ALGORITHMS FOR AUTOMATED ANALYSIS AND FORECASTING OF SOCIAL MEDIA DYNAMICS	190
Лілія Кхвостівська, Назар Паламар, Сергій Сторозь ПРОГРАМНИЙ ЗАСІБ ВЕЙВЛЕТ-ВІЯВЛЕННЯ РАДІОСИГНАЛІВ В МАТЕРИНЬСЬКОМУ БАЗИСІ МЕКСИКАНСЬКОГО КАПЕЛЮХА Lilija Khvostivska, Nazar Palamar, Serhii Storozh SOFTWARE WAVELET DETECTION OF RADIO SIGNALS IN THE MEXICAN HAT MOTHER BASE	191
Д.Р.Чарковський, Н.В.Студник МЕТОДИ ДЕТЕКТУВАННЯ ТЕКСТОВИХ ОБЛАСТЕЙ НА ЗОБРАЖЕННЯХ D.R. Charkovskyi, N.B. Studnyk METHODS FOR DETECTION OF TEXT REGIONS IN IMAGES	192
Євгенія Тіш, Руслан Шалапай ІєРАРХІЧНА КЛАСТЕРИЗАЦІЯ ДЛЯ ВИЗНАЧЕННЯ СУКУПНОСТІ ФУНКЦІОНАЛЬНИХ ТА НЕФУНКЦІОНАЛЬНИХ ВІМОГ КОМП'ЮТЕРНИХ СИСТЕМ Ievhenia Tysh, Ruslan Shalapai HIERARCHICAL CLUSTERIZATION FOR DETERMINING FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS OF COMPUTER SYSTEMS	193

МЕТОДИ ДЕТЕКТУВАННЯ ТЕКСТОВИХ ОБЛАСТЕЙ НА ЗОБРАЖЕННЯХ

D.R. Charkovskiy, N.B. Stadnyk, Ph.D.

METHODS FOR DETECTION OF TEXT REGIONS IN IMAGES

Кількість інформації, котра зберігається як зображення, велика, що робить неможливим її опрацювання вручну. Попри те, що область можливого застосування достатньо широка та наявний значний прогрес у машинному навчанні, детектування та поділ на сегменти текстових областей на зображеннях наразі ще є проблемою. Текст, котрий розміщений на зображеннях реальних сцен, може мати різний розмір, колір, стиль, поворот щодо горизонту тощо. Також ймовірним є його перекриття додатковими об'єктами на зображенні чи присутність низької контрастності із основним фоном.

Для детектування та виділення розташування текстових областей на зображеннях застосовуються стандартні методи розпізнавання текстових областей. Усі наявні методи розпізнавання можна поділити на такі три групи:

- методи пов'язаних компонентів;
- текстурні методи;
- методи глибокого навчання.

Для порівняння методів детектування текстових областей на зображеннях було розглянуто такі характеристики:

- швидкість роботи (кількість часу, необхідна для навчання цього методу);
- ресурсна вартість обчислень (вимоги до обчислювальних ресурсів обладнання);
- робота зі складними зображеннями: геометричні перетворення (повороти, розтягування); текстурні спотворення (шум, низька контрастність, складні фони);
- ефективність методу (точність чи інші засоби оцінки).

Методи, які порівнювалися: пов'язаних компонентів - метод максимально стабільних екстремальних областей (MSER), метод перетворення за товщиною штриха символу тексту (SWT); текстурні - дискретно-косинусне перетворення (DCT), гістограма спрямованих градієнтів (HOG), ознаки Хаара; глибокого навчання - згортова нейронна мережа (CNN), повнозгортова нейронна мережа (FCN), мережа архітектури U-Net.

Результати порівняння свідчать, що методи пов'язаних компонентів володіють високою швидкістю роботи, але недостатньо добре працюють із різноманітними спотвореннями зображень та володіють середньою ефективністю. Текстурні методи мають середню швидкість роботи, погано працюють з різними спотвореннями і мають низьку ефективність. Методи глибокого навчання є найповільнішою групою через тривалий час навчання, але добре працюють з різними спотвореннями зображень, а також мають високу продуктивність. Отже, методи, що ґрунтуються на глибокому навчанні, становлять велику цінність для розгляду.

CNN пред'являють менше вимог до обладнання порівняно з іншими мережами, проте є недостатніми для визначення розташування текстових областей на зображеннях. FCN з несиметричною структурою та мережі архітектури U-Net знаходяться на одному рівні. Однак симетрична архітектура мережі U-Net надає більше можливостей для розпізнавання текстових областей через наявність більшої кількості шарів підвищуючої дискретизації для опрацювання карт ознак. Тому архітектура U-Net була обрана для реалізації алгоритму розпізнавання написів на зображеннях.

Додаток Б

Фрагмент коду навчання мережі з файлу Training.ipynb

```
def get_img_id(img_path):
    return img_path[:len(img_path)-4]
#img_ids = list(map(get_img_id,
list(train_masks_df.img.values)))
def load_image_disk(img_id,
folder=TRAIN_PATH):
    img = imageio.imread(os.path.join(folder, img_id + ".png"))
return img

def load_mask_disk(img_id, folder=TRAIN_MASKS_PATH):
    img = imageio.imread(os.path.join(folder, img_id + ".png"))
return img

# Helper functions to plot car, mask, masked_car
def plot_image(img_id):
    img = imageio.imread(os.path.join(TRAIN_PATH, img_id + ".png"))
imgplot = plt.imshow(img)
plt.axis('off') plt.show()

def get_raw_img(img_id):
    return mpimg.imread(os.path.join(RAW_PATH, img_id + ".png"))

def plot_image_from_array(img_array):
imgplot = plt.imshow(img_array)
plt.axis('off')
plt.show()

def plot_mask(img_id):
    mask = mpimg.imread(os.path.join(TRAIN_MASKS_PATH, img_id +
".png"))
imgplot = plt.imshow(mask)
plt.axis('off') plt.show()

def plot_masked_image(img_id):
```

```

img = imageio.imread(os.path.join(TRAIN_PATH, img_id + ".png"))
mask = imageio.imread(os.path.join(TRAIN_MASKS_PATH, img_id +
".png"))
mask = mask[:, :, 0:3]

mask[mask == 255] = 1 masked_img = img * mask
imgplot = plt.imshow(masked_img) plt.axis('off')
plt.show()

def get_mask(img_id, num_channels=-1): if num_channels!=-1:
mask = imageio.imread(os.path.join(TRAIN_MASKS_PATH, img_id +
".png"), flatten=True)
mask[mask > 128] = 1
if len(mask.shape) == 2:
mask = mask.reshape(mask.shape[0], mask.shape[1], 1) return
mask
else:
mask = imageio.imread(os.path.join(TRAIN_MASKS_PATH, img_id +
".png"))
mask = mask[:, :, 0:3] return mask

def resize_img(img, new_s = new_shape): return
trans.resize(img, new_s)

def load_imgs(img_id, folder=TRAIN_PATH):
img = imageio.imread(os.path.join(folder, img_id + ".png"))
return img

def gray2rgb(img):
img = np.squeeze(img) w, h = img.shape
ret = np.empty((w, h, 3), dtype=np.uint8) ret[:, :, 0] = img
ret[:, :, 1] = img
ret[:, :, 2] = img return ret

.....

```