

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра прикладної математики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Застосування сучасних методів обробки даних для ідентифікації морських
суден за супутниковими спостереженнями

(тема)

Виконав:

здобувач 2 року навчання, групи САУМ-24-1
Олеся КОСТІКОВА
(Власне ім'я, ПРІЗВИЩЕ)

Спеціальність 124 Системний аналіз

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Системний аналіз і управління

(повна назва освітньої програми)

Керівник асист. Олексій СЛАВІК

(посада, Власне ім'я, ПРІЗВИЩЕ)

Допускається до захисту

Завідувач кафедри ПМ

(підпис)

Максим СИДОРОВ

(Власне ім'я, ПРІЗВИЩЕ)

2025 р.

Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту

Кафедра прикладної математики

Рівень вищої освіти другий (магістерський)

Спеціальність 124 Системний аналіз

(код і повна назва)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Системний аналіз і управління

(повна назва)

ЗАТВЕРДЖУЮ:

Завідувач кафедри _____

(підпис)

“ 10 ” листопада 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Костіковій Олесі Валеріївні

(прізвище, ім'я, по батькові)

1. Тема роботи Застосування сучасних методів обробки даних для ідентифікації морських суден за супутниковими спостереженнями

затверджена наказом по університету від 10 листопада 2025 р. № 1027 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 18 грудня 2025 р.

3. Вихідні дані до роботи супутникові знімки з зображенням морських вод з судами

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Системний аналіз предметної області

2. Вибір і обґрунтування методу розв'язання

3. Програмна реалізація

4. Результати обчислювального експерименту

5. Аналіз можливих застосувань

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій _____

1. Актуальність теми роботи _____

2. Постановка задачі _____

3. Системний аналіз предметної області _____

4. Метод чисельного аналізу _____

5. Результати обчислювального експерименту _____

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Терміни виконання етапів роботи | Примітка |
|---|---|---------------------------------|----------|
| 1 | Підбір та вивчення технічної літератури за темою роботи | 10 – 16 листопада 2025 р. | виконано |
| 2 | Вибір та обґрунтування методу | 17 – 23 листопада 2025 р. | виконано |
| 3 | Розробка алгоритму і програми | 24 – 30 листопада 2025 р. | виконано |
| 4 | Проведення аналітичних досліджень та розрахунків | 01 – 07 грудня 2025 р. | виконано |
| 5 | Робота над текстом пояснювальної записки | 08 – 17 грудня 2025 р. | виконано |
| 6 | Представлення роботи на рецензію в ЕК | 18 грудня 2025 р. | виконано |

Дата видачі завдання 10 листопада 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ асист. Олексій СЛАВІК
(підпис) (посада, Власне ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка: 56 с., 9 рис., 2 дод., 30 джерел.

АНАЛІЗ ДАНИХ, АУГМЕНТАЦІЯ, БІНАРНА КРОСС-ЕНТРОПІЯ, ЗГОРТКА, НЕЙРОННА МЕРЕЖА, СЕГМЕНТАЦІЯ, СИГМОЇДА, СУПУТНИКОВІ ЗНІМКИ, ФУНКЦІЯ ВТРАТ, U-NET АРХІТЕКТУРА.

Об'єкт дослідження – супутникові зображення морських акваторій, на яких можуть бути присутні кораблі різних типів та розмірів.

Мета роботи – розробка ефективної моделі, яка автоматично виявляє всі кораблі на супутникових знімках якомога швидше та точніше, навіть за умов наявності хмар чи серпанку.

Методи дослідження – у роботі використовуються такі методи дослідження як аналіз наукових джерел, методи машинного навчання, методи системного аналізу, методи програмування, оцінки ефективності системи за допомогою показників точності.

У кваліфікаційній роботі розроблено ефективну модель, яка автоматично виявляє кораблі на супутникових знімках, навіть за умов наявності хмар або серпанку. Проведено детальний аналіз предметної області, досліджено сучасні підходи до обробки зображень та сегментації предметів на них.

Основний акцент зроблено на використанні сучасних методів аналізу та обробки супутникових даних для задачі розпізнавання морських суден. Запропонований підхід забезпечує гнучке опрацювання різномірних джерел інформації, скорочення часу ідентифікації та підтримку подальшого розвитку системи. Виконано програмну реалізацію розробленого рішення, проведено його перевірку на практичних прикладах супутникових спостережень і доведено результативність отриманих рішень. Розроблена система може бути інтегрована з існуючими інформаційно-аналітичними платформами та застосовуватися для моніторингу морського простору в різних сферах діяльності.

Результати дослідження можуть бути використані для розроблення та впровадження подібних підходів у галузі моніторингу морського транспорту, систем безпеки судноплавства, контролю руху флоту та аналізу просторових даних. Запропоновані методи забезпечують ефективну обробку супутникової інформації та створюють основу для подальшого розширення функціональних можливостей і підвищення точності ідентифікації морських суден.

ABSTRACT

Introductory note: 56 pages, 9 figures, 2 appendixes, 30 sources.

AUGMENTATION, BINARY CROSS-ENTROPY, CONVOLUTION, DATA ANALYSIS, LOSS FUNCTION, NEURAL NETWORK, SATELLITE IMAGES, SEGMENTATION, SIGMOID, U-NET ARCHITECTURE

Object of research – satellite images of marine areas in which ships of various types and sizes may be present.

Purpose of work – to develop an effective model that automatically detects all ships in satellite images as quickly and accurately as possible, even under conditions of cloud cover or haze.

Methods of research – the work employs research methods such as analysis of scientific sources, machine learning methods, system analysis methods, programming methods, and evaluation of system effectiveness using accuracy metrics.

In the qualification work, an effective model has been developed that automatically detects ships in satellite images, even under conditions of cloud cover or haze. A detailed analysis of the subject area has been conducted, and modern approaches to image processing and object segmentation have been investigated.

The main emphasis is placed on the use of modern methods for the analysis and processing of satellite data for the task of maritime vessel recognition. The proposed approach provides flexible processing of heterogeneous information sources, reduces identification time, and supports further development of the system. A software implementation of the developed solution has been carried out, tested on practical examples of satellite observations, and the effectiveness of the obtained solutions has been demonstrated. The developed system can be integrated with existing information and analytical platforms and applied to maritime space monitoring in various fields of activity.

The research results can be used for the development and implementation of

similar approaches in the field of maritime transport monitoring, navigation safety systems, fleet traffic control, and spatial data analysis. The proposed methods ensure efficient processing of satellite information and form a basis for further expansion of functional capabilities and improvement of the accuracy of maritime vessel identification.

ЗМІСТ

| | |
|--|----|
| | С. |
| Перелік скорочень, умовних познач, одиниць і термінів | 10 |
| Вступ | 11 |
| 1 Системний аналіз предметної області та постановка задач дослідження | 13 |
| 1.1 Системний аналіз задачі застосування сучасних методів обробки даних для ідентифікації морських суден у супутникових спостереженнях | 13 |
| 1.1.1 Загальна характеристика задачі ідентифікації морських суден | 13 |
| 1.1.2 Airbus Ship Detection Challenge як дослідницький кейс | 14 |
| 1.1.3 Сегментація зображень та її використання в машинному навчанні | 14 |
| 1.2 Аналіз сценаріїв вирішення задачі застосування сучасних методів обробки даних для ідентифікації морських суден у супутникових спостереженнях | 16 |
| 1.2.1 Сегментаційні моделі для аналізу супутникових зображень | 16 |
| 1.2.2 Аугментація даних для підвищення точності моделей | 17 |
| 1.2.3 U-Net архітектура нейронної мережі | 18 |
| 1.2.4 Convolution (згортка): принцип та застосування | 21 |
| 1.3 Змістовна та формальна постановка задачі | 23 |
| 1.4 Постановка задач дослідження | 24 |
| 2 Вибір та обґрунтування методу розв’язання | 26 |
| 2.1 Загальний принцип роботи нейронних мереж | 26 |
| 2.2 Loss-функція: | 27 |
| 2.3 Функція активації | 28 |
| 2.4 Оптимізатор | 30 |
| Висновки за розділом 2 | 31 |
| 3 Програмна реалізація | 32 |
| 3.1 Віртуальне середовище та обчислювальні ресурси | 32 |

| | |
|--|----|
| | 9 |
| 3.2 Імплементация коду | 34 |
| 3.2.1 Loss функція: Binary Cross-Entropy | 34 |
| 3.2.2 Функція активації: Сигмоїда..... | 35 |
| 3.2.3 Оптимізатор: «Адам»..... | 35 |
| 3.3 Опис програми | 37 |
| Висновки за розділом 3 | 37 |
| 4 Результати експерименту та їх аналіз | 39 |
| 4.1 Експеримент для фото без кораблів | 39 |
| 4.2 Експеримент для фото з кораблями | 40 |
| Висновки за розділом 4 | 41 |
| Висновки | 42 |
| Перелік джерел посилання | 44 |
| Додаток А Лістинг програми | 47 |
| Додаток Б Визначення датасету | 53 |

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ

BCE – Binary Cross-Entropy;

Adam – Adaptive Moment Estimation;

MSE – Mean Squared Error;

MAE – Mean Absolute Error.

ВСТУП

Актуальність теми. Зараз дуже багато кораблів які знаходяться в світовому океані, кожен з них треба відстежувати. На жаль, зараз у світі дуже нестабільні відносини між країнами, тому охорона власних кордонів має велику вагу в політичних питаннях. В даному випадку детекція кораблів у водах світового океану є актуальною та затребуваною. Крім того, розвиток супутникових технологій та методів обробки великих масивів даних відкриває нові можливості для автоматизованого спостереження за морською обстановкою в режимі близькому до реального часу. Використання сучасних алгоритмів аналізу даних дозволяє підвищити точність і швидкість виявлення суден, зменшити вплив людського фактора та забезпечити своєчасне реагування на потенційні загрози, що є особливо важливим для гарантування безпеки мореплавства та захисту національних інтересів.

Мета і завдання кваліфікаційної роботи. Метою кваліфікаційної роботи є розробка ефективної моделі, яка автоматично виявляє всі кораблі на супутникових знімках якомога швидше та точніше, навіть за умов наявності хмар чи серпанку. Для досягнення поставленої мети необхідно виконати наступні завдання:

– провести огляд і аналіз сучасного стану задачі «застосування сучасних методів обробки даних для ідентифікації об’єктів у супутникових спостереженнях»;

– підготувати дані: завантажити та дослідити датасет Airbus, виконати попередню обробку даних (нормалізація, змінити розмір. Очистити датасет від «пошкоджених» або некоректних зразків);

– реалізувати модель та навчити її на підготовлених даних (обрати підхід та архітектуру, налаштувати гіперпараметри);

– оцінити ефективність реалізованої моделі, перевірити як вона працює з ускладненими зображеннями та оцінити з якою точністю вона визначає на них кораблі;

- провести оптимізацію;
- візуалізувати результати детекції та задокументувати результати.

Об'єктом дослідження є супутникові зображення морських акваторій, на яких можуть бути присутні кораблі різних типів та розмірів.

Предметом дослідження методи та алгоритми автоматичного виявлення кораблів на супутникових знімках із використанням технологій машинного навчання та комп'ютерного зору.

Методи дослідження. У кваліфікаційній роботі використовуються такі методи дослідження як аналіз наукових джерел, методи машинного навчання, методи системного аналізу, методи програмування, оцінки ефективності системи за допомогою показників точності.

Публікації. Результати, отримані у роботі, було представлено на ІХ Міжнародній науково-технічній конференції «Комп'ютерне моделювання та оптимізація складних систем. КМОСС-2025» (м. Дніпро, 5-7 листопада 2025 р.) [1]. Також результати було представлено на XXI Міжнародному молодіжному форумі «Радіоелектроніка та молодь у XXI столітті» (м. Харків, 16-19 квітня 2025 р.) [2].

1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Системний аналіз задачі застосування сучасних методів обробки даних для ідентифікації морських суден у супутникових спостереженнях

1.1.1 Загальна характеристика задачі ідентифікації морських суден

Ідентифікація морських суден на супутникових знімках – це важлива та актуальна задача на даний час. Це потрібно для безпеки судноплавства, для виявлення незаконної діяльності або правопорушень у світовому океані та для моніторингу пересування кораблів. Суден в водах стає все більше та визначати їх на знімках стає складніше. Традиційно такі задачі виконуються за допомогою радарів, оптичного спостереження, але проблема в тому, що надходить дуже великий об'єм даних, які треба обробити. Саме тому можна казати, що мануальний аналіз не є ефективним вирішенням цієї проблеми.

Додатковою складністю є вплив погодних умов та якості супутникових знімків: хмари, туман, відблиски на воді чи низька роздільна здатність можуть значно ускладнювати розпізнавання. Крім того, судна можуть відрізнитися за формою, розміром, кольором, а також частково перекриватися або бути прихованими серед інших об'єктів.

Саме через ці причини використання машинного навчання для автоматичної детекції суден буде спрощувати задачу та надавати ефективне та швидше рішення проблеми. Основним викликом буде велика варіативність суден на зображеннях, наявність суші та портів, що буде збивати модель з вірного шляху.

Отже, автоматична детекція суден за допомогою машинного навчання значно підвищує ефективність аналізу супутникових знімків, зменшує потребу у ручній праці та дозволяє оперативно виявляти потенційні загрози на воді, що робить її важливим інструментом у сфері морської безпеки.

1.1.2 Airbus Ship Detection Challenge як дослідницький кейс

Kaggle – це провідна платформа в галузі аналізу даних і машинного навчання, яка з 2010 року надає спільнотам фахівців можливість змагатися, навчатися та спільно працювати над складними задачами. Компанію заснували у квітні 2010 року, і вже невдовзі вона стала місцем, де компанії, дослідники та аналітики можуть виставляти дані, формулювати завдання і отримувати рішення від світової спільноти.

Одним із основних напрямків діяльності Kaggle є проведення конкурсів-челенджів із машинного навчання. Організації виставляють задачі з реальними наборами даних, а учасники з усього світу створюють моделі, змагаються за найкращі результати і публікують свої рішення. Крім того, платформа надає можливість обміну датасетами, публікації Jupyter ноутбуків і навчання через серію міні-курсів Kaggle Learn.

Airbus Ship Detection Challenge – це змагання, організоване платформою Kaggle спільно з компанією Airbus [3]. Його мета – розробити алгоритм автоматичного виявлення суден на супутникових SAR-зображеннях.

Учасникам надавався набір даних, що містить понад 190 тисяч зображень, на яких судна займають лише малу частину кадру. Дані мають значний дисбаланс – більшість зображень не містять жодного судна. Основною метою є створення моделі, здатної здійснювати піксельну сегментацію: для кожного пікселя визначити, чи належить він судну. Це робить задачу складною, але водночас цінною для подальшого розвитку систем морського моніторингу.

1.1.3 Сегментація зображень та її використання в машинному навчанні

Сегментація в машинному навчанні відноситься до завдання поділу даних на змістовні сегменти або групи [4]. Мета – виявити структуру або закономір-

ності в даних, щоб елементи в одному сегменті були схожі один на одного та відрізнялися від елементів в інших сегментах. Найбільш поширене застосування – розбиття зображення на кілька сегментів (наборів пікселів) для спрощення його представлення чи визначення об'єктів та кордонів.

В машинному навчанні існує декілька типів сегментації:

- clustering-based segmentation (unsupervised) [5]: навчання без вчителя, це означає, що модель самостійно вивчає структуру даних;

- сегментація зображень: вона надає кожному пікселю мітку для того, щоб ідентифікувати об'єкти та області. (підтипи: семантична сегментація – класифікація кожного пікселя, сегментація екземплярів – розділення кожного екземпляра об'єкта, паноптична – поєднує в собі сегментацію екземплярів та семантичну сегментацію);

- сегментація тексту [6]: роздроблення тексту на певні структурні частини (наприклад, речення, теми, абзаци), цей тип використовується в обробці текстів, розпізнання мовлення та в інформаційному пошуку.

Процес включає кілька основних етапів:

- попередня обробка даних – нормалізація зображень, збільшення даних (data augmentation) для покращення стійкості моделі до різних умов освітлення та масштабу;

- навчання моделі сегментації – використання розмічених масок кораблів для тренування нейромережі з метою прогнозування маски для кожного пікселя;

- постобробка результатів – фільтрація шуму, розділення тісно розташованих суден, а також перетворення масок у формат, придатний для оцінки за метрикою IoU (Intersection over Union) [7], що є стандартом для задач сегментації.

Таким чином, сегментація у задачі виявлення кораблів на супутникових знімках є прикладом практичного використання комп'ютерного зору та глибокого навчання для точного виявлення об'єктів у складних умовах.

1.2 Аналіз сценаріїв вирішення задачі застосування сучасних методів обробки даних для ідентифікації морських суден у супутникових спостереженнях

1.2.1 Сегментаційні моделі для аналізу супутникових зображень

Для вирішення даної задачі було обрано використовувати саме сегментаційну модель. По-перше, сегментаційні моделі виконують піксельну класифікацію (визначають до якого класу відноситься кожен піксель). В нашому випадку це критично важливо, тому що відношення корабля до моря/порта дуже мале з супутникових знімків. Він буде займати дуже малу площу на зображенні. Тому, наприклад, детекційні моделі (такі як YOLO/Faster R-CNN) [8] будуть часто пропускати маленькі судна або захоплювати велику площу фону, через те що вони працюють прямокутниками. Сегментаційна модель навпаки буде чітко виділяти контури корабля, навіть якщо на фото зображена тільки його частина, або якщо поруч знаходяться інші об'єкти.

По-друге, датасет, який представляє компанія Kaggle для навчання моделі складається з радарних зображень (SAR), які містять шум та артефактами, які пов'язані з відбиттям хвиль. Сегментаційні моделі (особливо з архітектурою U-net) вміють виділяти структури з різних масштабів завдяки своїй побудові, це забезпечує розділення справжніх об'єктів від шумових утворень.

По-третє, в результаті роботи сегментаційної моделі ми отримуємо маску, яку можна використовувати для побудови карт розташування суден, моніторингу змін у часі та в оцінки щільності руху кораблів. Тобто сегментаційна модель дає нам аналітичний результат, який можна буде використовувати для геоінформаційних систем (GIS), а не просто виявити координати виявленого судна.

Крім того, сегментаційні моделі дозволяють інтегрувати різноманітні техніки покращення результатів, такі як аугментація даних, використання багаторівневих ознак та каскадних підходів. Це забезпечує більш високу стійкість мо-

делі до змін у структурі зображень, різної орієнтації суден або варіацій у рівні шуму. Зокрема, методи аугментації, як-от випадкові повороти, масштабування чи зміна яскравості, допомагають збільшити різноманітність вибірки та покращують узагальнюючу здатність моделі.

Важливим етапом є також правильний підбір функцій втрат. Для задачі, де кораблі займають дуже малу частину зображення, класичні функції, такі як Binary Cross-Entropy [10], можуть бути недостатньо ефективними. Тому доцільно використовувати спеціалізовані метрики, наприклад Dice Loss [9] або комбінацію BCE та Dice Loss, що дозволяє краще враховувати дисбаланс між класами та забезпечувати точніше виділення об'єктів невеликого розміру.

Ще однією перевагою сегментаційних моделей є можливість подальшої постобробки отриманих масок. Наприклад, застосування морфологічних операцій або фільтрації за розміром може допомогти видалити залишкові шумові пікселі або об'єднати фрагментовані області. Це особливо корисно в умовах, коли на зображенні присутні відблиски чи інші артефакти, що можуть спотворювати форму судна.

Таким чином, використання сегментаційних моделей у задачі виявлення суден на супутникових SAR-знімках є виправданим і ефективним рішенням. Вони забезпечують високу точність виділення контурів об'єктів, дозволяють гнучко адаптувати процес навчання до складних умов даних та генерують результати, які можуть бути інтегровані у ширший контекст моніторингу й аналізу морських перевезень.

1.2.2 Аугментація даних для підвищення точності моделей

Аугментація даних [11] використовує наявні дані для створення нових вибірок, які допоможуть покращити оптимізацію моделі. Це штучне збільшення обсягу навчальної вибірки буде застосовано тому що наш набір даних є нерівномірний (у датасеті переважно наявні зображення на яких нема жодного

корабля). Існують такі техніки аугментації даних: геометричне перетворення (відзеркалювання, обератння, масштабування), фотометричні зміни (зміна контрасту) та симуляція умов зйомки (затемнення якоїсь частини зображення, розмиття).

Застосування аугментації дозволяє моделі стати більш стійкою до зміни ракурсів, освітлення, шумів та інших факторів, характерних для супутникових SAR-зображень. Важливо, що ці перетворення виконуються без втрати змістовної інформації: судна та інші об'єкти на зображеннях зберігають свої властивості, проте їхній вигляд модифікується достатньо, щоб збільшити різноманітність даних.

Особливо корисними є методи, що імітують реальні умови зйомки. Наприклад, додавання шуму або часткове затемнення можуть допомогти моделі навчитися визначати судна навіть при несприятливих погодних умовах чи поганій якості знімків. Це критично важливо для задачі виявлення об'єктів у морі, де зовнішні умови часто непередбачувані.

Крім того, аугментація може сприяти зниженню ризику переобучення моделі. Оскільки штучно створені зображення роблять вибірку більш різноманітною, модель стає менш схильною до запам'ятовування окремих прикладів та краще узагальнює дані. Це особливо важливо в ситуаціях, коли кількість позитивних прикладів (з кораблями) значно менша за кількість негативних.

Таким чином, аугментація даних є ключовим етапом у підготовці тренувального набору для задачі виявлення суден. Вона дозволяє покращити якість навчання, підвищити точність моделі та забезпечити її здатність працювати в реальних умовах супутникового моніторингу морської поверхні.

1.2.3 U-Net архітектура нейронної мережі

U-Net архітектуру [12] можна вважати кращим стандартним рішенням задачі сегментації зображень. U-Net має 2 основних шляхи.

Перший шлях – encoder (стискач) – знижує розширення вхідного зображення за допомогою згортки та Max Pulling [13] (рис. 1.1).



Рисунок 1.1 – Схема принципу роботи енкодера

Max Pulling – це операція зменшення розмірності зображення. Тобто зображення ділиться на невеликі блоки (наприклад 3 на 3) і з кожного блоку береться максимальне значення. В результаті ми отримуємо нову карту ознак, але у разі меншу за висотою та шириною. Таким чином відбуваються послідовні згортки шарів, які виділяють високорівневі ознаки.

Чому використовуємо саме Max Pulling у вирішенні нашої задачі. По-перше, це зменшення розмірності та обчислювальних витрат: супутникові знімки дуже великі. Макс-пулінг зменшує розмір карти ознак без втрати важливої інформації про присутність об'єктів, що дозволяє прискорити навчання і зменшити потребу в пам'яті. По-друге, інваріантність до невеликих зсувів та деформацій: кораблі на знімках можуть бути у різних положеннях та під різними кутами. Макс-пулінг дозволяє мережі зберігати ознаки об'єктів незалежно від їх точного розташування у невеликому локальному регіоні, що покращує загальну здатність мережі до узагальнення. Також важливо, що фокус на найсильніші ознаки: Оскільки макс-пулінг вибирає максимальне значення в регіоні, мережа «запам'ятовує» найяскравіші ознаки (наприклад, контури кораблів, виділені пікселі), а слабкі сигнали шуму від води або хмар відкидаються. Також перевагою можна зазначити покращення стабільності навчання: зменшення розмірності також знижує ризик переобучення, оскільки мережа отримує більш стисле

представлення зображення з акцентом на найважливіші характеристики.

Другий шлях – decoder (розжимач) – навпаки збільшує розширення карт ознак до вихідного зображення, що забезпечує точну локалізацію (рис. 1.2). Таким чином відбуваються послідовні «розгортки» шарів, що поступово відновлюють просторову роздільність [14].

По-друге, головною особливістю даної архітектури можна вважати skip connections [15]. Це з'єднання між шарами енкодера та відповідними ним шарами декодера. Це допомагає зберігати навіть невеликі деталі об'єктів, які були втрачені при зменшенні розмірності.

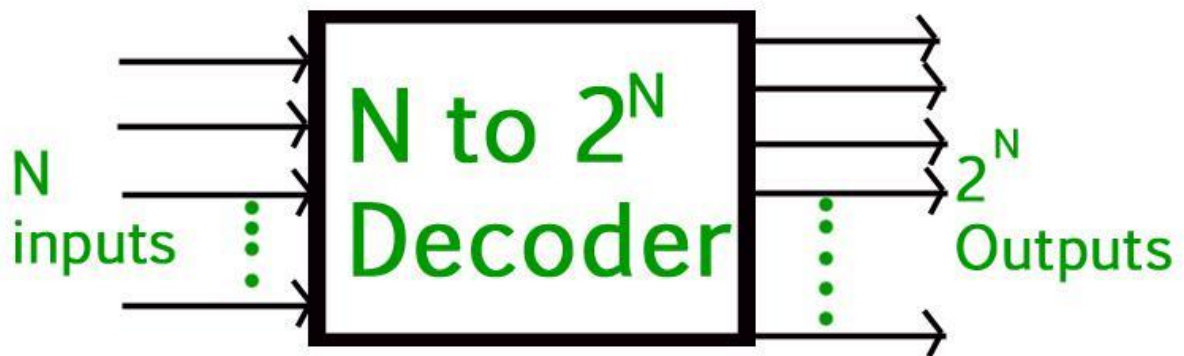


Рисунок 1.2 – Схема принципу роботи декодера

Важливою частиною архітектури U-Net – є "bottleneck" [16]. Bottleneck – це центральний рівень, де зустрічаються шляхи стискання (кодер) та розширення (декодер). Це найглибша частина мережі, що містить найбільш абстрактні ознаки після стиснення зображення до найнижчої роздільної здатності. Вузким місцем зазвичай є згорткові шари, які обробляють ці високорівневі ознаки, перш ніж декодер почне реконструювати остаточну карту сегментації. Окрім цього, bottleneck слугує зоною з максимальною кількістю фільтрів, що дозволяє мережі вловлювати складні патерни та контекст. Саме тут модель може навчитися розпізнавати глобальні структурні взаємозв'язки в зображенні. У цьому шарі часто застосовують додаткові операції, такі як dropout, для запобігання перенавчанню. Дані, отримані на цьому етапі, згодом передаються до декодера

через шляхи розгортання. Під час декодування інформація з bottleneck інтегрується з пропущеними з кодера ознаками, що забезпечує точнішу локалізацію об'єктів. Завдяки цьому U-Net здатний об'єднати глобальний контекст із деталізованими локальними особливостями, що дає високу якість сегментації.

На рисунку 1.3 можна побачити графічну модель архітектури U-Net.

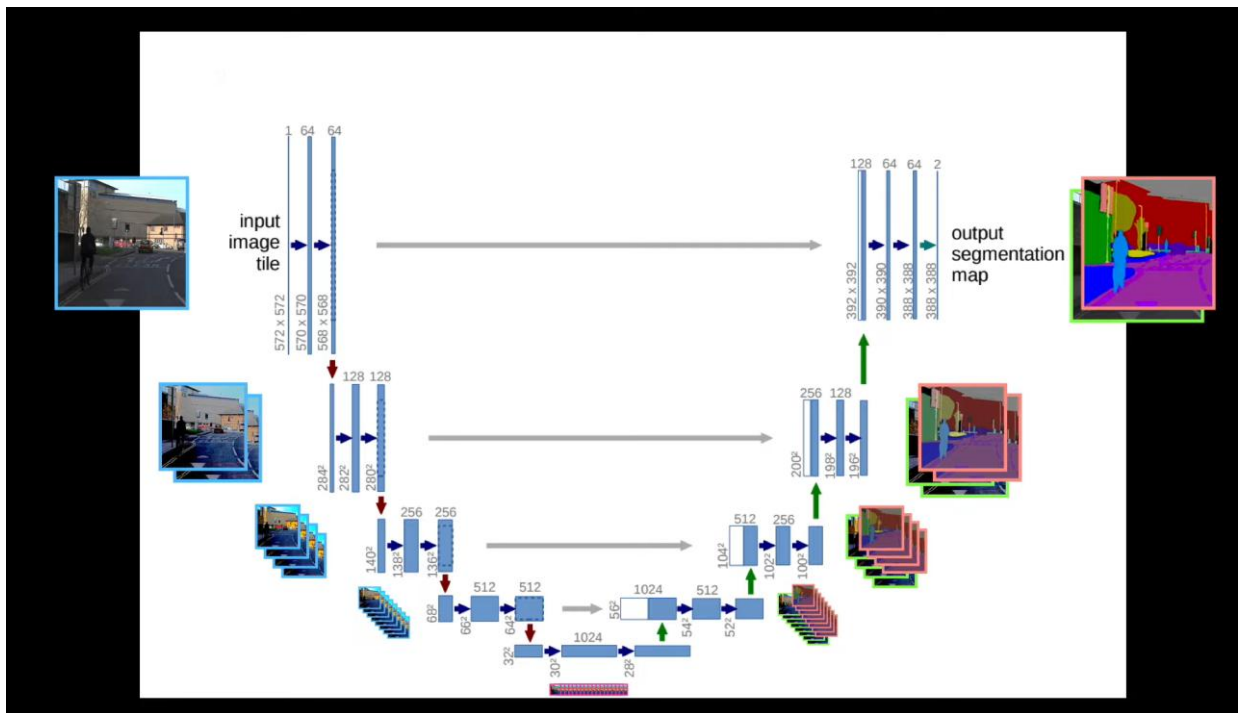


Рисунок 1.3 – Архітектура U-Net

1.2.4 Convolution (згортка): принцип та застосування

Згортка [17] – це математична операція, що дозволяє витягувати локальні особливості зображення. У найпростішому випадку вона виконується шляхом пропускання невеликого фільтра (ядра – kernel) по всьому зображенню і обчислення зваженої суми пікселів в цьому вікні. Результатом є нове зображення (карта ознак), що відбиває наявність певних особливостей у різних частинах вихідного зображення.

Механіка роботи згортки: кожен супутниковий знімок має розмір 768 на 768 пікселів. Фільтр 3 на 3 буде проходитись по зображенню та поступово об-

числювати зважену суму пікселів. Після першого шару згортки модель може виділити контури кораблів і виділити відмінності між фоном та судном. Після кількох початкових згорток ми отримуємо feature maps (карти ознак), які містять більше комбінованої інформації про форму та структуру. Тут фільтри вже можуть «розпізнавати» більш складні елементи: форму корабля, його тінь, відблиски на воді. Завдяки застосуванню pooling – шарів мережа видаляє шум і зменшує розмірність даних, залишаючи лише ключові ознаки. На цьому етапі згортка вже формує ієрархію ознак: від простих до складних. На останніх шарах мережі feature maps комбінуються та обробляються (часто через додаткові згорткові шари або fully connected шари), щоб отримати ймовірність наявності корабля в кожному пікселі або регіоні зображення.

У моделей типу U-Net або Mask R-CNN згортки формують карту ймовірностей маски корабля, а у моделях типу RetinaNet або YOLO згортки створюють ознаки для класифікації та локалізації об'єктів.

Головні переваги згорток у нашій задачі:

- автоматичне виділення ознак: на відміну від класичних методів комп'ютерного зору, де ознаки доводиться визначати вручну (наприклад, контури або текстури), згорткові мережі самостійно навчаються виділяти найважливіші патерни для конкретної задачі, для кораблів це можуть бути контури корпусу, сигнали на воді або специфічні текстури палуби;

- локальна обробка: згортка обробляє невеликі фрагменти зображення за раз, що дозволяє моделі навчитися розпізнавати локальні закономірності незалежно від того, де вони знаходяться на знімку, це робить модель стійкою до зміщень об'єктів;

- масштабування та ієрархія ознак: кілька згорткових шарів, накладених один на одного, дозволяють моделі навчитися спочатку розпізнавати прості патерни (лінії, кути), а потім комбінувати їх у складніші структури (форма корабля, його частини), це особливо важливо для маленьких або частково перекритих кораблів;

- зменшення розмірності та шуму: застосування pooling-операцій після

згорток дозволяє зменшити розміри feature map, відкидаючи менш важливі деталі та залишаючи лише ключові ознаки для детекції, це прискорює навчання та робить модель більш стійкою до шуму.

На рисунку 1.4 представлено схематичний приклад процесу роботи згортки.

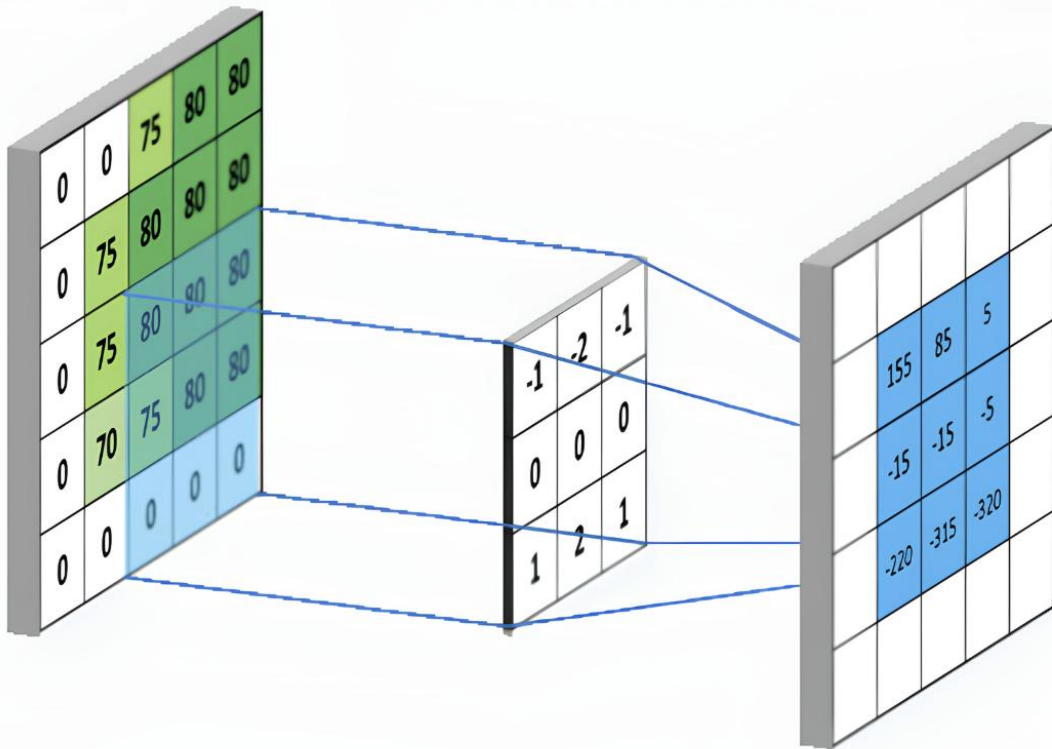


Рисунок 1.4 – Схематичний приклад використання згортки

1.3 Змістовна та формальна постановка задачі

Необхідно розробити нейронну мережу, яка буде здатна обробляти данні для ідентифікації морських суден на супутникових знімках. Основні функції системи:

- застосування випадкових геометричних і фотометричних перетворень (аугментація даних);
- обробка вхідного зображення та сегментація розпізнавання суден;

- післяобробка результатів;
- візуалізація результатів.

Вхідні дані: супутникові SAR-зображення.

Вихідні дані: зображення, на яких буде результат сегментації (фон буде позначено одним кольором, а кораблі на фото – іншим) та оцінка точності моделі.

Обмеження:

- система повинна працювати з SAR-зображеннями у фіксованому форматі;
- навчання моделі потребує значних обчислювальних ресурсів (GPU, оперативна пам'ять, сховище даних).

Головна функція: зробити сегментацію зображення з найвищою точністю та швидкістю.

1.4 Постановка задач дослідження

Отже, метою кваліфікаційної роботи є розробка ефективної моделі, яка автоматично виявляє всі кораблі на супутникових знімках якомога швидше та точніше, навіть за умов наявності хмар чи серпанку. Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести огляд і аналіз сучасного стану задачі «застосування сучасних методів обробки даних для ідентифікації об'єктів у супутникових спостереженнях»;
- підготувати дані: завантажити та дослідити датасет Airbus, виконати попередню обробку даних (нормалізація, змінити розмір, очистити датасет від «пошкоджених» або некоректних зразків);
- реалізувати модель та навчити її на підготовлених даних (обрати підхід та архітектуру, налаштувати гіперпараметри);

- оцінити ефективність реалізованої моделі, перевірити як вона працює з ускладненими зображеннями та оцінити з якою точністю вона визначає на них кораблі;
- провести оптимізацію;
- візуалізувати результати детекції та задокументувати результати.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ'ЯЗАННЯ

2.1 Загальний принцип роботи нейронних мереж

Нейронна мережа – це модель машинного навчання, яка імітує роботу мозку людини. Вона об'єднує нейрони в шари та підбирає певні ваги, що навчаються розпізнавати шаблони з вхідних даних. Нейрон отримує на вхід певні значення, зважує їх коефіцієнтами, підсумовує, додає зсув і пропускає через функцію активації. Нейронна мережа має: вхідний шар, який приймає дані та містить в собі необроблені об'єкти (у нашому випадку це зображення пікселів), приховані шари перетворюють вхідні дані на нові представлення (лінійне перетворення між входом та виходом) та вихідний шар, в якому використовується нелінійна функція активації (у нашому випадку сигмоїда) для отримання остаточного передбачення. Для виконання заданої задачі нейронна мережа буде виконувати *semantic segmentation*, тобто класифікацію кожного пікселя. В нашому випадку вхідними даними будуть супутникові зображення RGB, розміром 768×768 . На виході з нейронної мережі ми будемо отримувати бінарну маску, де 1 – пікселі самого корабля, а 0 – це фон зображення.

Нейронна мережа, як вирішення проблеми детекції кораблів на знімках було обрано тому, що супутникові зображення такої якості і складності дуже складні для класичних алгоритмів: кораблі мають різну форму, розмір та кут нахилу, на зображенні можуть бути ускладнюючі фактори такі як туман, хмари, тощо та фон (море або океан) буде зливатися з кораблем. У нашому випадку набір даних дуже великий, що тільки сприяє використуванню нейронної мережі, бо чим більше даних ми можемо дати їй на вхід, тим точніше вона навчиться розпізнавати судна на зображеннях. Нейронна мережа також здатна автоматично виділяти найважливіші ознаки, що зменшує потребу у ручній обробці даних. Після навчання вона може ефективно узагальнювати інформацію та працювати з новими зображеннями, які не входили до тренувального набору.

Таким чином, використання нейронних мереж значно підвищує точність та надійність задачі виявлення кораблів на супутникових знімках.

2.2 Loss-функція

Лосс функція (або функція втрат) [18] – це функція, яка відстежує ступінь похибки у вихідних даних моделі штучного інтелекту. Вона робить це шляхом кількісної оцінки різниці (втрат) між прогнозованим значенням та фактичним значенням. Якщо прогнози моделі точні, втрати невеликі і навпаки.

Лосс функція перетворює помилку моделі в число, дає сигнал для градієнтного спуску, визначає напрямок і швидкість навчання та напряму впливає на якість та стабільність моделі. Тобто можна сказати, що вона дає моделі інформацію як покращитися.

Існує декілька видів Loss-функцій: Mean Squared Error (MSE), Mean Absolute Error (MAE), Huber Loss, Binary Cross-Entropy (BCE), Categorical Cross-Entropy, Weighted Loss, Focal Loss, Dice Loss, Hinge Loss.

Найпоширенішими лосс функціями є MSE, MAE, Log Loss, Categorical Cross-Entropy та Huber Loss.

Mean Squared Error (MSE) [19] – це функція втрат, що використовується в задачах регресії (коли метою задачі є спрогнозувати певне число). Вона використовується коли важливо сильно карати великі помилки та коли ми маємо «чисті» дані. Особливості цього виду функції в тому, що вона проста та може перетягувати модель через одиничні аномальні точки.

Mean Absolute Error (MAE) [20] – це функція втрат, що використовується для випадків, коли в даних є викиди та всі помилки однаково важливі. Її перевагою є те, що вона стабільніша за MSE та модель менш чутлива до шуму, але навчання може бути більш повільнішим.

Binary Cross-Entropy (BCE) – це функція втрат, яка використовується в задачах бінарної класифікації. Вона кількісно визначає різницю між фактични-

ми мітками класів (0 або 1) та прогнозованими ймовірностями, що видаються моделлю. Чим нижче значення бінарної перехресної ентропії, тим краще прогнози моделі узгоджуються з істинними мітками.

Categorical Cross-Entropy [21] – це функція втрат, що використовується для багатокласової класифікації та коли кожен об'єкт має лише один клас. Вона може працювати з softmax та вирішує задачі типу: «обрати один клас з багатьох».

Huber Loss [22] – це функція втрат, яку можна назвати серединою між MSE та MAE. Вона буде кращим вибором коли важливі і стабільність і чутливість. Якщо наявна мала помилка, то вона поводить себе як MSE. При наявності великої помилки буде реагувати як MAE функція.

Задача проекту – це бінарна піксельна сегментація. Коли кожен піксель належить до певного класу корабля (1), або до фону зображення (0). Отже, кожен піксель можна розглядати як окремий приклад бінарної класифікації. Бінарна крос ентропія безпосередньо вимірює наскільки точно модель вчиться відрізняти корабель від фону на зображенні. Вона може легко оптимізуватися за допомогою класичного оптимізатора Adam.

2.3 Функція активації

Активаційні функції є важливими для будь-якої нейронної мережі, оскільки вони дозволяють генерувати багатовимірні, нелінійні шаблони. Без них вихід нейронної мережі був би обмежений простими лінійними або поліноміальними співвідношеннями першого ступеня.

Активаційні функції застосовуються до нейронів для введення нелінійності, що дозволяє мережі обробляти складніші дані. Ваги в кожному шарі використовуються для обчислення зваженої суми, до якої додається член зміщення. Це зміщення допомагає розрізнити вхідні дані від вихідних і зміщує активаційну функцію за потреби.

Без активаційних функцій нейронні мережі по суті поводитися б як моделі лінійної регресії, що значно знижує їхню продуктивність і репрезентативну здатність. Завдяки включенню активаційних функцій, підходи штучних нейронних мереж, такі як глибоке навчання, здатні моделювати складні, багатовимірні набори даних і вивчати складні шаблони через кілька прихованих шарів.

Історично склалося так, що найпростішими функціями активації були порогова та лінійна. Порогова функція працює за принципом "True or False": якщо сигнал перевищує певний рівень, нейрон активується; в протилежному випадку він залишається неактивним. Такі функції використовували для давніх моделей нейронів, але сьогодні використовуються дуже рідко, оскільки з ними неможливо ефективно навчати мережу. Лінійна функція просто передає той самий однаковий сигнал, тому вона не збільшує здатність моделі до вивчення складних патернів і використовується тільки в окремих випадках, як правило, на виході мережі.

Функція Сигмоїди можна назвати фундаментальною в розвитку нейронних мереж. Вона плавно «стискає» будь-яке вхідне значення до обмеженого діапазону. Тобто сигмоїда перетворює будь-яке число від $-\infty$ до $+\infty$ у значення між 0 і 1, що дозволяє інтерпретувати результат як ймовірність – наскільки сильно нейрон активований або наскільки піксель “належить” до певного класу. Роль Сигмоїди у моделі полягає в тому, щоб на останньому шарі U-Net перетворити вихід моделі на ймовірність приналежності пікселя до класу – корабель.

На рисунку 2.1 можна побачити як виглядає графік функції Сигмоїда.

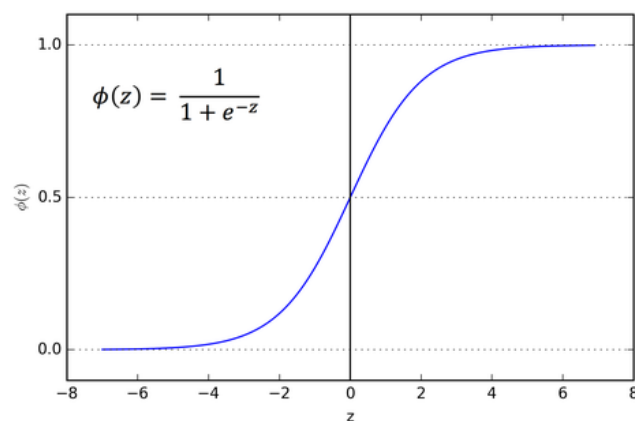


Рисунок 2.1 – Сигмоїдна функція

Існує функція активації типу ReLU. Вона пропускає позитивний сигнал, а негативний ігнорує. Завдяки цьому навчання стає швидшим і стабільнішим, а проблема затухання градієнтів значною мірою зменшується.

2.4 Оптимізатор

Оптимізатор – це алгоритм, який допомагає навчати модель шляхом поступового покращення її параметрів. Ми подаємо дані в модель та отримуємо передбачення. Потім порівнюємо отримане передбачення з правильними відповідями. Таким чином ми отримуємо лосс функцію. Після цього оптимізатор використовує градієнти цієї функції, щоб зрозуміти в якому напрямку та наскільки змінювати параметри моделі (weights). Далі ми повторюємо ці кроки, запускаючи велику кількість епох, доки помилка не стане мінімальною.

Найпростішим оптимізатором є градієнтний спуск. Модель робить маленькі кроки в напрямку, який зменшує помилку. У випадку якщо кроки будуть занадто великі, то навчання буде нестабільним, якщо ж кроки будуть навпаки занадто малі, то процес навчання буде дуже повільним. Саме через ці причини на практиці використовується не класичний градієнтний спуск, а його вид – стохастичний градієнтний спуск. Він оновлює параметри, використовуючи лише тільки частину даних. Через це навчання стає швидшим та дозволяє моделі краще виходити з локальних мінімумів.

Існує такий вид оптимізаторів як адаптивні (AdaGrad, RMSProp і Adam). Вони автоматично підлаштовують швидкість навчання для кожного параметра. Це стає у нагоді, коли різні параметри з різною силою впливають на помилку.

Adam [23] (Adaptive Moment Estimation) – це один із найпопулярніших оптимізаторів для навчання нейронних мереж. Він поєднує ідеї двох методів: Momentum, який використовує усереднений градієнт, щоб рухатися більш плавно та RMSProp, який регулює кроки окремо для кожного параметра залежно

від «історії» градієнтів. Adam адаптує швидкість навчання для кожного параметра окремо, що робить навчання стабільнішим і швидшим.

Висновки за розділом 2

У даному розділі було обґрунтовано вибір методу розв'язання задачі детекції морських суден на супутникових зображеннях. Розглянуто загальний принцип роботи нейронних мереж та показано їхню ефективність для задач піксельної семантичної сегментації, зокрема у випадку складних і неоднорідних зображень морської поверхні. Доведено доцільність використання глибоких нейронних мереж для автоматичного виділення ознак та надійного розпізнавання суден незалежно від їх форми, розміру та умов зйомки.

Обґрунтовано вибір функції втрат Binary Cross-Entropy як оптимальної для задачі бінарної піксельної класифікації, де кожен піксель належить або до класу судна, або до фону. Також розглянуто роль сигмоїдної функції активації, яка дозволяє інтерпретувати вихід моделі як імовірність приналежності пікселя до класу корабля та забезпечує коректну роботу моделі на вихідному шарі.

Окрему увагу приділено вибору оптимізатора Adam, який завдяки адаптивному налаштуванню швидкості навчання, швидкій збіжності та стабільності є ефективним для навчання глибоких нейронних мереж на супутникових зображеннях з високою варіативністю та шумом. У сукупності обрані методи та підходи формують надійну основу для побудови ефективної системи детекції кораблів і забезпечують високу точність та стійкість моделі в реальних умовах застосування.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Віртуальне середовище та обчислювальні ресурси

Сегментаційні моделі висувають високі вимоги до обчислювальних ресурсів, особливо до обсягу відеопам'яті. Моделі комп'ютерного зору, що застосовуються для аналізу супутникових знімків та виявлення морських суден, зазвичай є доволі ресурсомісткими. Хоча існує можливість виконання обчислень виключно на центральному процесорі, такий підхід є малоефективним і значно збільшує час обробки даних.

Під час написання цієї роботи була змога використовувати виключно одну відеокарту NVIDIA GeForce RTX 3080 Ti Laptop GPU. Альтернативним варіантом є застосування хмарних обчислювальних сервісів, однак більшість із них потребують фінансових витрат, що обмежує їх практичне використання в навчальних і дослідницьких цілях.

Важливим аспектом реалізації проєкту є правильна організація програмного середовища. Для цього використовується віртуальне середовище – ізольована програмна оболонка, у межах якої встановлюються всі необхідні бібліотеки та інструменти. Такий підхід дозволяє уникнути конфліктів між різними версіями програмного забезпечення та забезпечує стабільність роботи проєкту. Для ізоляції проєктного середовища та уникнення конфліктів між бібліотеками ми застосовуємо Anaconda [24].

Anaconda – це інструмент в основному для написання програм мовою програмування Python, який дозволяє працювати зокрема з машинним навчанням та великим обсягом даних. Conda може автоматично керувати залежностями бібліотек, що спрощує встановлення складних пакетів (таких як PyTorch [25], torchvision, imageio, pandas, sklearn, та інші). Також Anaconda дозволяє легко встановлювати бібліотеки з підтримкою GPU (PyTorch з CUDA), що дозволяє максимально ефективно використовувати відеокарту. Ще однією важливою перевагою є те, що цей інструмент дозволяє змінювати версії бібліотек для

конкретного середовища, це надає змогу легко та швидко робити оновлення та підтримувати стабільність проекту.

JupyterLab [26] – це інтерактивне середовище розробки, яке є розширенням можливостей класичного Jupyter Notebook [27], який також входить в Anaconda. Воно надає єдиний робочий простір для виконання коду, аналізу даних, візуалізації результатів і роботи з файлами проекту. У контексті нашої задачі JupyterLab відіграє ключову роль, оскільки поєднує зручність експериментів з високою гнучкістю керування проектом.

Основною особливістю JupyterLab є модульна архітектура. Це означає, що користувач може одночасно відкривати декілька ноутбуків, текстових файлів, зображень і скриптів у межах одного інтерфейсу. Це особливо корисно під час роботи з великими наборами зображень, коли необхідно паралельно аналізувати дані, редагувати код та переглядати результати сегментації.

Перевага, яка відіграла значну роль в обиранні середовища, це те, що завдяки інтеграції з `matplotlib` [28] та іншими бібліотеками візуалізації, JupyterLab дозволяє одразу відображати супутникові зображення разом із передбаченими масками суден. Це значно спрощує аналіз якості сегментації та порівняння результатів на різних етапах навчання моделі. Крім того, JupyterLab підтримує розширення, які дають змогу додавати нові функціональні можливості, зокрема інструменти для керування версіями коду та автоматизації робочих процесів. Завдяки цьому середовище є зручним не лише для експериментальних досліджень, а й для повноцінної розробки та супроводу моделей машинного навчання.

У нашому проекті були використані такі бібліотеки та модулі:

- `numpy`, `pandas` [29], `matplotlib` – для роботи з масивами даних та візуалізації;
- `torch`, `torchvision` – для побудови та навчання нейронних мереж на GPU;
- `imageio` – для завантаження супутникових зображень у форматі `numpy`;
- `sklearn.model_selection` [30] – для розподілу даних на навчальні та тестові набори;

– custom dataset та модель UNet – для сегментації суден на супутникових знімках.

3.2 Імплементация коду

3.2.1 Loss функція: Binary Cross-Entropy

Binary Cross-Entropy (BCE) – це одна з найпоширеніших лосс функцій. Вона використовується в задачах бінарної класифікації та бінарної сегментації зображень. Вона визначає різницю між фактичними мітками класів (0 або 1) та прогнозованими ймовірностями, що видаються моделлю після застосування сигмоїдної активаційної функції. Інтуїтивно Binary Cross-Entropy штрафує модель за впевнені, але неправильні прогнози значно сильніше, ніж за невпевнені. Саме такий підхід стимулює мережу швидше навчатися коректним межах між класами. Чим нижче значення лосс функції, тим краще прогнози моделі узгоджуються з істинними мітками.

Задача проекту – це бінарна піксельна сегментація. Коли кожен піксель належить до певного класу корабля (1), або до фону зображення (0). Бінарна крос-ентропія безпосередньо вимірює, наскільки точно модель вчиться відрізняти кораблі від фону на рівні окремих пікселів. Вона добре працює з імовірнісним виходом моделі, дозволяючи інтерпретувати значення кожного пікселя як ступінь впевненості в наявності корабля. Це особливо важливо в задачі виявлення малих об'єктів, де межі кораблів можуть бути розмитими, а частина пікселів містить неоднозначну інформацію.

Крім того, у нашому датасеті присутній значний дисбаланс класів – більшість пікселів належать до фону, а кораблі займають відносно малу площу зображення. Binary Cross-Entropy легко розширюється до зваженої версії (Weighted BCE), що дозволяє компенсувати цей дисбаланс шляхом збільшення ваги помилок на пікселях кораблів. Завдяки цьому модель не схильна ігнорува-

ти рідкісний клас та краще виявляє навіть невеликі кораблі.

3.2.2 Функція активації: Сигмоїда

Функція сигмоїди перетворює будь-яке число від $-\infty$ до $+\infty$ у значення між 0 і 1, що дозволяє інтерпретувати результат як ймовірність – наскільки сильно нейрон активований або наскільки піксель “належить” до певного класу (наприклад, до корабля).

Роль Сигмоїди у моделі полягає в тому, щоб на останньому шарі U-Net перетворити вихід моделі на ймовірність приналежності пікселя до класу – корабель. Завдяки цьому значення, що ближче до 1, можна трактувати як високий ступінь впевненості, що піксель належить до корабля, а значення, близькі до 0, як його відсутність.

По-перше, завдяки Сигмоїді вихідні значення моделі можна трактувати як ймовірність належності пікселя до корабля. Це дозволяє легко комбінувати їх з метриками чи використовувати різні пороги для оптимізації результатів.

По-друге, функція активації сигмоїда добре поєднується з бінарною крос-ентропією (Binary Cross-Entropy Loss), яка є стандартом для бінарної сегментації. Це забезпечує стабільне навчання моделі та правильне масштабування градієнтів.

По-третє, Сигмоїда обмежує вихідні значення між 0 і 1, що запобігає надмірним або негативним прогнозам, які не мають сенсу для ймовірнісної інтерпретації.

3.2.3 Оптимізатор: «Адам»

Adam (Adaptive Moment Estimation) – це один із найпопулярніших оптимізаторів для навчання нейронних мереж. Він поєднує ідеї двох методів:

Momentum, який використовує усереднений градієнт, щоб рухатися більш плавно та RMSProp, який регулює кроки окремо для кожного параметра залежно від «історії» градієнтів. Adam адаптує швидкість навчання для кожного параметра окремо, що робить навчання стабільнішим і швидшим.

Adam підтримує два статистичні значення для кожного параметра:

- перший момент (середній градієнт);
- другий момент (середньоквадратичний градієнт).

Потім він застосовує bias correction (корекцію похибки), щоб компенсувати зміщення на ранніх ітераціях.

Чому був обраний саме оптимізатор Adam:

- висока варіативність даних: супутникові знімки кораблів містять різноманітні умови освітлення, погодні ефекти та масштаби об'єктів, Adam здатен адаптивно налаштувати швидкість навчання для кожного параметра мережі, що допомагає ефективно навчати модель навіть на складних, неоднорідних даних;

- швидка збіжність: завдяки комбінованому використанню першого та другого моментів градієнта Adam швидко сходиться навіть на великих нейронних мережах, що важливо для задачі детекції об'єктів, де моделі, як правило, глибокі (наприклад, U-Net або ResNet-базовані архітектури);

- стабільність навчання: у задачах детекції кораблів зображення часто містять шум або великі порожні області води, Adam зменшує ризик «стрибків» градієнтів і робить процес навчання більш стабільним;

- мінімальна потреба в налаштуванні learning rate: у реальних умовах, коли потрібно швидко експериментувати з архітектурами, Adam дозволяє досягти хороших результатів без тривалого підбору гіперпараметрів.

Таким чином, Adam є ідеальним вибором для Airbus Ship Detection Challenge, оскільки він поєднує швидку збіжність, стабільність та адаптивність до складних, неоднорідних супутникових зображень, що робить його практичним і ефективним для детекції кораблів.

3.3 Опис програми

Програма використовує навчений модельний підхід для передбачення маски, що відображає ймовірне розташування кораблів на зображенні. Ця *predicted mask* створюється за допомогою алгоритмів глибокого навчання, зокрема нейронних мереж, які були натреновані на великій кількості анотованих даних із зазначенням позицій кораблів.

Для перевірки точності роботи моделі передбачувана маска порівнюється з *ground truth mask*, яка є еталонною анотацією та містить реальне розташування кораблів на знімку. Це порівняння дозволяє оцінити ефективність моделі, визначити її сильні та слабкі сторони та виявити випадки, коли модель може пропустити корабель або неправильно визначити його контур.

На виході програма візуалізує три основні компоненти: оригінальне фото, *ground truth mask* та *predicted mask*. Така візуалізація дозволяє користувачам наочно порівняти результати моделі з реальними даними, оцінити точність сегментації та швидко визначити, наскільки добре модель справляється з різними типами зображень, включно з тими, де кораблі частково перекриваються або знаходяться на складному фоні. Крім того, програма може слугувати інструментом для подальшого вдосконалення моделі, наприклад, шляхом аналізу помилок та налаштування архітектури нейронної мережі або методів обробки зображень.

У підсумку, програма не лише демонструє результати роботи моделі, але й є практичним інструментом для аналізу та покращення алгоритмів сегментації кораблів на супутникових знімках, що є важливим для завдань морського моніторингу, контролю судноплавства та виявлення незаконної діяльності на водних просторах.

Висновки за розділом 3

Реалізація проєкту з сегментації кораблів на супутникових знімках потребує значних обчислювальних ресурсів, зокрема потужної відеокарти, оскільки

обробка великих зображень на центральному процесорі є неефективною. Віртуальне середовище, створене за допомогою Anaconda, дозволяє ізолювати програмне середовище, керувати залежностями бібліотек та забезпечує стабільність роботи проєкту, а інтеграція з JupyterLab забезпечує зручний та гнучкий інтерфейс для аналізу даних, експериментів та візуалізації результатів. У процесі розробки були використані ключові бібліотеки для роботи з масивами даних, нейронними мережами та супутниковими зображеннями, зокрема PyTorch, torchvision, numpy, pandas, matplotlib, imageio та sklearn.

Для задачі бінарної сегментації суден застосовано Binary Cross-Entropy як функцію втрат, що дозволяє ефективно навчати модель навіть при дисбалансі класів та точно відрізнити кораблі від фону. Функція активації сигмоїда на вихідному шарі U-Net перетворює результати на ймовірності, забезпечуючи коректну інтерпретацію пікселів як належних або неналежних до класу корабля. Оптимізатор Adam забезпечує стабільне та швидке навчання моделі завдяки адаптивному налаштуванню швидкості навчання для кожного параметра, що особливо важливо для обробки різномірних супутникових зображень.

Програма дозволяє передбачати маску розташування кораблів на зображеннях, порівнювати її з ground truth та візуалізувати результати разом з оригінальним фото. Такий підхід дає змогу оцінювати ефективність моделі, виявляти її слабкі місця та використовувати результати для подальшого вдосконалення алгоритмів сегментації. У підсумку реалізований інструмент є ефективним та практичним засобом для моніторингу морських об'єктів, контролю судноплавства та виявлення потенційної незаконної діяльності на водних просторах.

4 РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТУ ТА ЇХ АНАЛІЗ

4.1 Експеримент для фото без кораблів

В цьому експерименті ми перевіримо як відреагує розроблена система у випадку, коли на фото зовсім немає кораблів.

Нижче наведено приклад виводу програми, коли на вході було фото зовсім без кораблів (див. рис. 4.1).



Рисунок 4.1 – Візуалізація виходу з програми

Під час тестування моделі на зображеннях, де кораблі відсутні, програма коректно відтворює усі три основні компоненти: оригінальне фото, ground truth mask та predicted mask. Ground truth mask у таких випадках порожня, що відповідає відсутності об'єктів, а predicted mask також не містить помилкових виділених ділянок, що свідчить про високу здатність моделі уникати хибнопозитивних прогнозів. Такий результат демонструє, що модель адекватно розпізнає відсутність цільових об'єктів і не генерує артефактів на зображеннях без кораблів, що є важливим показником точності та надійності сегментації. Це підтверджує ефективність обраних функцій втрат, активації та оптимізації для забезпечення стабільної роботи моделі навіть у відсутності об'єктів. Отже, отримані результати свідчать про збалансованість моделі та її здатність коректно узагальнювати дані без перенавчання на наявність кораблів.

4.2 Експеримент для фото з кораблями

В цьому експерименті ми перевіримо чи відтворить розроблена система усі кораблі, які є на фото.

Нижче наведено декілька прикладів виводу програми, коли на вході було фото з різною кількістю кораблів (див. рис. 4.2, 4.3, 4.4).

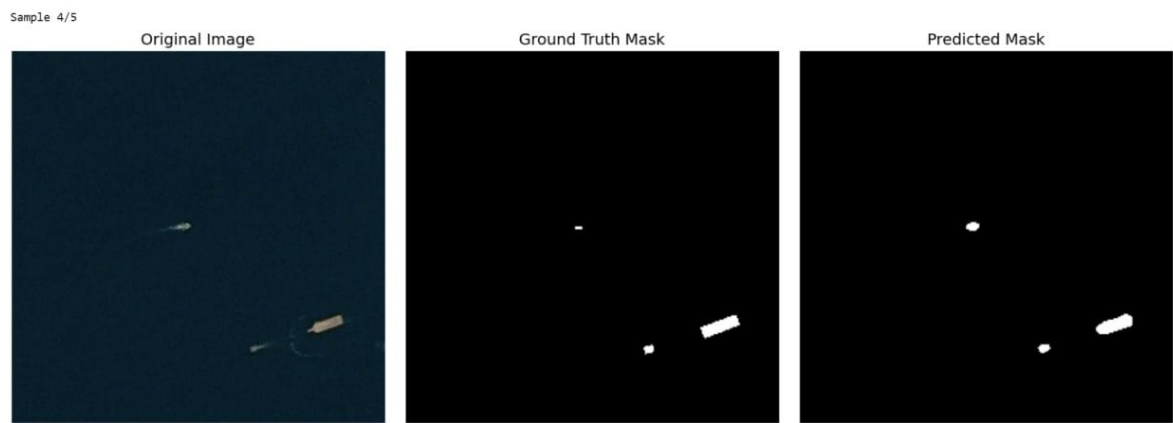


Рисунок 4.2 – Приклад 1 візуалізації виходу з програми

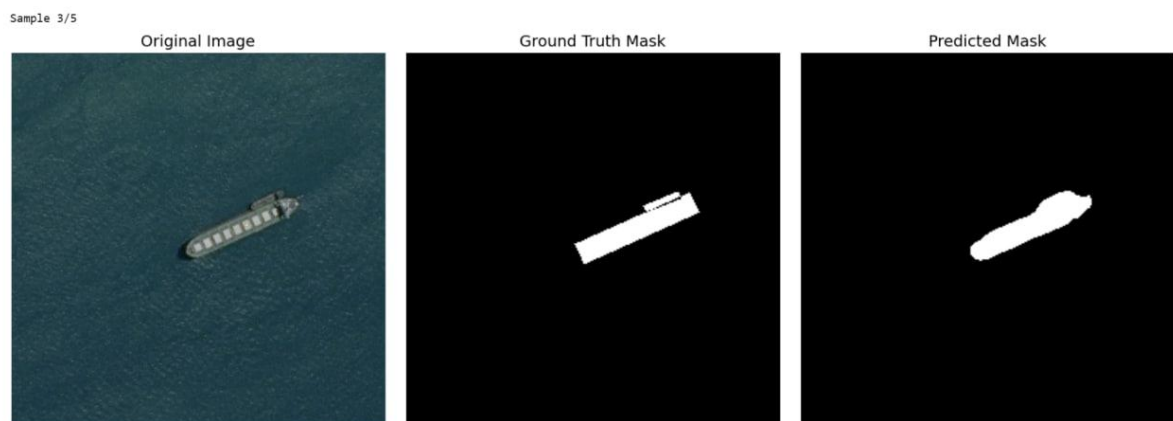


Рисунок 4.3 – Приклад 2 візуалізації виходу з програми

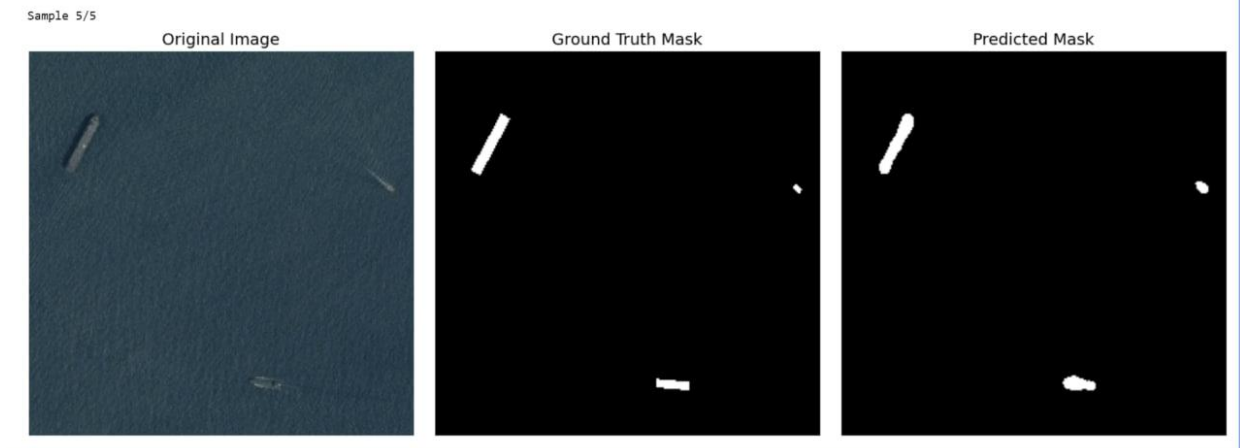


Рисунок 4.4 – Приклад 3 візуалізації виходу з програми

Висновки за розділом 4

Проведений експеримент показав, що модель ефективно справляється з сегментацією супутникових знімків незалежно від кількості кораблів на зображенні. Програма коректно генерує predicted mask для кожного випадку та наочно відображає її разом із оригінальним фото та ground truth mask. Аналіз результатів візуалізації свідчить, що модель здатна точно визначати контури кораблів як у випадках із невеликою кількістю об'єктів, так і на більш завантажених зображеннях, при цьому мінімізуючи помилки пропуску чи невірної класифікації пікселів. Такий підхід дозволяє не лише оцінювати точність сегментації, але й швидко виявляти слабкі місця моделі для подальшого вдосконалення алгоритму.

ВИСНОВКИ

Під час написання кваліфікаційної роботи були виконані поставленні задачі. Такі як розробити ефективну модель, яка автоматично виявляє всі кораблі на супутникових знімках якомога швидше та точніше, навіть за умов наявності хмар чи серпанку.

Системний аналіз показав, що моделі сегментації є найбільш підходящими для вирішення цього завдання, оскільки дозволяють не тільки виявити наявність об'єкта на зображенні, а й точно визначити межі. Описано основні принципи роботи мережі U-Net, зокрема функції згорткових шарів, максимального пулу та декодування, що забезпечують ефективне вилучення ознак та відновлення просторової структури об'єкта.

Для оптимізації процесу обробки даних було застосовано техніки нормалізації та автоматичного вирівнювання яскравості. Було сформовано набір критеріїв для відбору супутникових знімків, що охоплювали різні погодні умови та часові інтервали. Також була проведена підготовка навчального набору, що включала напівавтоматичну розмітку об'єктів із використанням допоміжних інструментів. Для підвищення стійкості моделі застосовано методи аугментації, які враховують повороти, масштабування та випадкові зсуви. Опрацьовано механізм попереднього вилучення текстурних характеристик, що сприяє точнішому розпізнаванню об'єктів на складних фонах. Відзначено важливість балансування класів, оскільки кількість зразків із наявними суднами була значно меншою за кількість фрагментів фону. Структура проєкту була організована таким чином, щоб забезпечити модульність та можливість подальшого розширення функціоналу. Також сформовано підхід до інтеграції отриманої моделі у існуючі серверні рішення, що спростило потенційне впровадження.

В результаті розроблена система на базі архітектури U-Net демонструє високу ефективність у задачах виявлення суден на супутникових знімках, забезпечує можливість масштабування та подальшої інтеграції до систем морського моніторингу. Отримані результати підтверджують перспективність вико-

ристання сучасних методів глибокого навчання для аналізу супутникових даних та розробки автоматизованих систем спостереження.

Отриманні результати можна застосовувати для екологічного моніторингу, моніторингу морських акваторій, підтримки служб безпеки та пошуково-рятувальних операцій, для оптимізації управління морською логістикою та для подальших наукових досліджень.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Костікова О. В. Using AI to identify ships in satellite images. *IX Міжнародна науково-технічна конференція «Комп'ютерне моделювання та оптимізація складних систем. КМОСС-2025»*. зб. матеріалів конференції (м. Дніпро, 5-7 листопада 2025 р.). Дніпро : ННІ УДХТУ, 2025. С. 162–163.
2. Костікова О. В. Detecting all ships in satellite images. *29-й Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті»* : зб. матеріалів форуму (м. Харків, 16-19 квітня 2025 р.) Т. 7. Харків : ХНУРЕ, 2025. С. 197–198.
3. Airbus Ship Detection from Satellite Imagery using Frequency Domain Learning / Mina Al-Saada, Nour Aburaeda, Alavikunhu Panthakkana [et al.]. *Image and Signal Processing for Remote Sensing XXVII, 1186210* : Proc. SPIE (12 September 2021). Vol. 11862. P. 1–7. <https://doi.org/10.1117/12.2600168>.
4. Segmentation Model. URL: <https://www.sciencedirect.com/topics/computer-science/segmentation-model> (дата звернення: 15.11.2025).
5. Clustering-Based Segmentation. URL: <https://fiveable.me/computer-vision-and-image-processing/unit-4/clustering-based-segmentation/study-guide/Yhhmwal095JfAQc5> (дата звернення: 15.11.2025).
6. Text Segmentation - Approaches, Datasets, and Evaluation Metrics. URL: <https://www.assemblyai.com/blog/text-segmentation-approaches-datasets-and-evaluation-metrics> (дата звернення: 15.11.2025).
7. IOU (Intersection over Union). URL: <https://medium.com/analytics-vidhya/iou-intersection-over-union-705a39e7acef> (дата звернення: 15.09.2025).
8. R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms. URL: <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e/> (дата звернення: 15.11.2025).
9. Dice Loss in Image Segmentation. URL: <https://gganbumarketplace.com/machine-learning/dice-loss-in-image-segmentation/> (дата звернення: 15.11.2025).

10. Binary Cross Entropy/Log Loss for Binary Classification. URL: <https://www.geeksforgeeks.org/deep-learning/binary-cross-entropy-log-loss-for-binary-classification/> (дата звернення: 15.11.2025).
11. Image Augmentation. URL: https://d2l.ai/chapter_computer-vision/image-augmentation.html (дата звернення: 15.11.2025).
12. U-Net: Architecture Explained. URL: <https://www.geeksforgeeks.org/machine-learning/u-net-architecture-explained/> (дата звернення: 15.11.2025).
13. What is Max Pooling and Why Do We Need Max Pooling? URL: <https://medium.com/@ilyurek/what-is-max-pooling-and-why-do-we-need-max-pooling-57247a3fbca9> (дата звернення: 15.11.2025).
14. U-Net: Convolutional Networks for Biomedical Image Segmentation. URL: <https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/> (дата звернення: 15.11.2025).
15. What is the role of skip connections in U-Net?. URL: <https://forums.fast.ai/t/what-is-the-role-of-skip-connections-in-u-net/100727> (дата звернення: 15.11.2025).
16. Bottleneck layer in Image Segmentation with U-Net. URL: <https://community.deeplearning.ai/t/bottleneck-layer-in-image-segmentation-with-u-net-w3a2/411326> (дата звернення: 15.11.2025).
17. The U-Net : A Complete Guide. URL: <https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8> (дата звернення: 15.11.2025).
18. What is a loss function?. URL: <https://www.ibm.com/think/topics/loss-function> (дата звернення: 15.11.2025).
19. Mean Squared Error. URL: <https://www.geeksforgeeks.org/maths/mean-squared-error/> (дата звернення: 20.11.2025).
20. Mean Absolute Error. URL: <https://www.sciencedirect.com/topics/engineering/mean-absolute-error> (дата звернення: 15.11.2025).

21. Categorical Cross-Entropy in Multi-Class Classification. URL: <https://www.geeksforgeeks.org/deep-learning/categorical-cross-entropy-in-multi-class-classification/> (дата звернення: 30.11.2025).
22. Huber Loss Function in Machine Learning. URL: <https://www.geeksforgeeks.org/machine-learning/huber-loss-function-in-machine-learning/> (дата звернення: 15.11.2025).
23. What is Adam Optimizer?. URL: <https://www.geeksforgeeks.org/deep-learning/adam-optimizer/> (дата звернення: 15.11.2025).
24. Anaconda. URL: <https://www.anaconda.com/> (дата звернення: 15.11.2025).
25. What is PyTorch? URL: <https://www.ibm.com/think/topics/pytorch> (дата звернення: 15.11.2025).
26. JupyterLab Documentation. URL: <https://jupyterlab.readthedocs.io/en/latest/> (дата звернення: 15.11.2025).
27. Jupyter Notebook Documentation. URL: <https://jupyter-notebook.readthedocs.io/en/stable/> (дата звернення: 15.11.2025).
28. Introduction to Matplotlib. URL: <https://www.geeksforgeeks.org/python/python-introduction-matplotlib/> (дата звернення: 15.11.2025).
29. pandas documentation. URL: <https://pandas.pydata.org/docs/> (дата звернення: 15.11.2025).
30. sklearn.model_selection. URL: https://scikit-learn.org/stable/api/sklearn.model_selection.html (дата звернення: 15.11.2025).