

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА  
Факультет інформаційних технологій  
Кафедра інтелектуальних технологій**

**ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА  
БАКАЛАВРА**

на тему:

**«Система розпізнавання місця пожежі  
на основі супутникових даних методами машинного  
навчання»**

Галузь знань **12 «Інформаційні технології»**

Спеціальність **122 «Комп'ютерні науки»**

Освітня програма **«Комп'ютерні науки»**

Освітній рівень: **бакалавр**

Виконала:  
студентка 4 курсу групи КН-41

Шпирук Є.С. 

Керівник:  
Сорока Петро Миколайович,  
кандидат фізико-математичних наук, доцент



Випускна кваліфікаційна робота бакалавра допущена до захисту  
рішенням кафедри *інтелектуальних технологій*  
Протокол № 11 від 06.06.2022 р.  
Зав. кафедри \_\_\_\_\_ доц. Іларіонов О.Є.

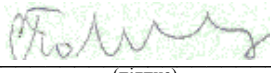
Київ - 2022




6. Консультанти з випускної кваліфікаційної роботи із зазначенням розділів випускної кваліфікаційної роботи, що їх стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 15 лютого 2022 року

Керівник  / Сорока П. М. /  
(підпис) (ПІБ)

Завдання прийняв до виконання  / Шпирук Є. С. /  
(підпис) (ПІБ)

### КАЛЕНДАРНИЙ ПЛАН

Пор. №	Назва етапів випускної кваліфікаційної роботи	Термін виконання етапів випускної кваліфікаційної роботи	Примітка
1	Дослідження джерел та збір інформації для випускної кваліфікаційної роботи.	15.02.2022 - 22.02.2021	
2	Робота над розділом 1. Аналіз предметної області.	23.02.22 – 01.03.22	
3	Робота над розділом 2. Пошук даних та аналіз інструментів та методів для реалізації поставленої задачі	01.03.22– 07.03.22	
4	Робота над розділом 3. Попередня обробка зображень, створення датасету для навчання.	08.03.22 – 15.04.22	
5	Робота над розділом 3. Програмна реалізація нейронних мереж для класифікації	16.04.2022 - 8.05.2021	
6	Робота над розділом 3. Тестування програмного модулю, доповнення датасету. Аналіз отриманих результатів. Написання висновків.	9.05.2022 – 17.05.2022	
7	Оформлення пояснювальної записки, підготовка презентаційних матеріалів.	18.05.2022 – 29.05.2022	

Студент-дипломник \_\_\_\_\_ / Шпирук Є. С. /  
(підпис) (ПІБ)

Керівник випускної кваліфікаційної роботи \_\_\_\_\_ / Сорока П. М. /  
(підпис) (ПІБ)

**Шпирук Євгенія Сергіївна** виконала випускню кваліфікаційну роботу на тему «Система розпізнавання місця пожежі на основі супутникових даних методами машинного навчання» за спеціальністю 122 – «Комп’ютерні науки».

У цій випускній кваліфікаційній роботі проведено аналіз існуючих розв’язків класифікації місця розгортання пожежі на основі даних дистанційного зондування Землі. Створено дві згорткові нейронні мережі для бінарної класифікації супутникових знімків з використанням різних функцій активації. Проведено порівняльний аналіз отриманих результатів.

**Ключові слова:** згорткова нейронна мережа, дистанційне зондування Землі, бінарна класифікація, геоінформаційна система, функція активації.

### Summary

The degree project: «Fire scene recognition system based on satellite data using machine learning methods» has been completed by **Yevheniia Shpyruk** by specialty 122 – «Computer Science».

In this graduation thesis the analysis of existing solutions of fire classification based on satellite data is conducted. Two convolutional neural networks are created for the implementation of binary classification using different activation functions. Comparative analysis of received results is conducted.

**Keywords:** convolutional neural network, remote sensing data, binary classification, geographic information system, activation function.

ВСТУП.....	7
РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД КЛАСИФІКАЦІЇ ДАНИХ ЗА ДОПОМОГОЮ МЕТОДІВ МАШИННОГО НАВЧАННЯ.....	8
1.1 Область застосування дистанційного зондування землі (ДЗЗ).....	8
1.2 Методи класифікації.....	10
1.3 Огляд існуючих рішень.....	14
1.4 Аналіз зацікавлених сторін.....	17
1.5 Постановка задачі.....	18
РОЗДІЛ 2. РОЗРОБКА АРХІТЕКТУРИ КЛАСИФІКАЦІЇ ПОЖЕЖІ ЗА ДОПОМОГОЮ МЕТОДІВ МАШИННОГО НАВЧАННЯ.....	20
2.1 Узагальнена структура системи класифікації даних ДЗЗ.....	20
2.2 Функціональний аналіз класифікації даних ДЗЗ.....	21
2.3 Вибір нейронної мережі.....	24
2.3.1. Нейронна мережа U-Net.....	24
2.3.2. Нейронна мережа Link-Net.....	26
2.4 Вибір функції активації.....	27
РОЗДІЛ 3 СТВОРЕННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ КЛАСИФІКАЦІЇ ТА ЇХ ПОРІВНЯЛЬНИЙ АНАЛІЗ.....	31
3.1 Попередня обробка супутникових знімків.....	31
3.2 Формування маски для навчання нейронних мереж.....	35
3.3 Створення нейронних мереж для класифікації супутникових даних .....	39
3.4 Порівняльний аналіз отриманих результатів.....	45
ВИСНОВКИ.....	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	51
ДОДАТКИ.....	53

## ВСТУП

Космічна ера людства та стрімкий розвиток дистанційного зондування Землі зробили можливими дослідження багатьох глобальних процесів та явищ, зокрема, розпізнавання місць розгортання пожеж та оцінку їх масштабів.

Зміна клімату, видового складу лісів, землекористування та зростання антропогенного навантаження на ліси зумовили суттєве загострення проблеми лісових пожеж і їхніх наслідків для суспільства та довкілля, як у глобальному, так і регіональному масштабі. Згідно статистичних даних щорічно на Землі відбувається від 50 до 120 тисяч природних пожеж. Про загострення цієї ситуації в світі свідчать великі лісові пожежі, які відбулися впродовж останнього десятиріччя, зокрема дві потужні пожежі в Чорнобильській зоні відчуження, які в 2015 р. охопили загальну площу близько 15 тис. га, а також катастрофічні пожежі, які трапились у 2016-2018 рр. в Австралії, Греції, Іспанії, Каліфорнії, Німеччині, Португалії, Швеції.

Окрім економічних збитків лісові пожежі призводять до тяжких екологічних наслідків: забруднення атмосфери, погіршення якості питної води, пришвидшення глобального потепління, збільшується ризик ураганів та тайфунів, втрачається біологічне різноманіття. Саме тому проблема лісових пожеж є однією з найважливіших національних задач.

## РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД КЛАСИФІКАЦІЇ ДАНИХ ЗА ДОПОМОГОЮ МЕТОДІВ МАШИННОГО НАВЧАННЯ

### 1.1. Область застосування дистанційного зондування Землі (ДЗЗ)

Дистанційне зондування Землі в Україні почали активно використовувати у 1970-х роках з метою моніторингу сільськогосподарських територій. На сьогоднішній день області застосування ДЗЗ значно розширилася завдяки неперервному розвитку цього наукового напрямку.

Дистанційне зондування землі активно застосовують для:

- створення та оновлення карт;
- управління територіями;
- екологічний та природоохоронний моніторинг;
- оцінка стану сільськогосподарських культур, прогнозування врожаю;
- контроль стану лісів, спостереження за вирубкою та оцінка наслідків лісових пожеж;
- контроль кліматичних змін;
- прогнозування та спостереження за стихійними лихами, оцінка масштабів та наслідків;
- геологічні дослідження, розвідка корисних копалин;
- дослідження атмосфери.

На рис. 1.1 зображена класифікація лісу та озеленених ділянок засобами ДЗЗ.

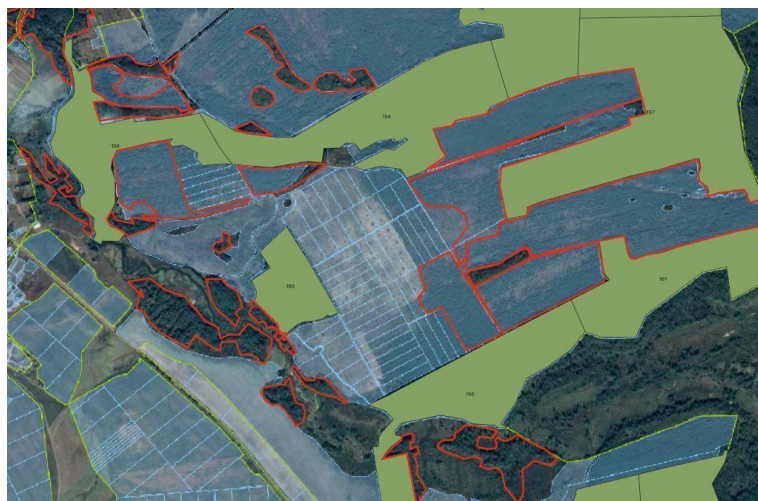


Рисунок 1.1. Класифікація лісу та озеленених ділянок засобами ДЗЗ

Для ДЗЗ використовують оптичні і радіолокаційні дані. Їх забезпечують супутники останнього покоління Landsat та Sentinel. Sentinel забезпечує найвищу роздільну здатність космознімків (з-поміж тих, що перебувають у відкритому доступі) – 10 м.

Landsat – найстаріша програма багатоспектральної супутникової зйомки з постійним спостереженням за планетою, розроблена NASA і працююча з 1970-х років. Оптичні знімки Landsat робилися з роздільною здатністю 30 м з початку 1980-х років. Починаючи з Landsat 5, почалось використання температурної інфрачервоної зйомки (з меншою роздільною здатністю, ніж оптична). 21 вересня 2021 року NASA відправили в космос найпотужніший супутник серії Landsat – Landsat 9. Як і всі попередні серії, Landsat 9 продовжить виконувати роль програми по моніторингу земельних ресурсів. Landsat 9 замінить застарілий Landsat 7, запущений в 1999 році і працюватиме в парі з Landsat 8, запущеним в 2013 році. На рис. 1.2 зображений Landsat 9.



Рисунок 1.2. Супутник Landsat 9

На рис. 1.3 зображений супутник Sentinel-2B, який був запущений в 2017 році.



Рисунок 1.3. Супутник Sentinel-2B

## 1.2. Методи класифікації

На сьогоднішній день існує багато різноманітних методів класифікації. Наприклад, можна виділити наступні: SVM, Bag of words. Також широкого поширення набуло використання штучних нейронних мереж для розв'язання задач класифікації.

В основі алгоритму SVM лежить принцип знаходження такої гіперплощини, яка б розділяла об'єкти на декілька класів. Для виконання якісної класифікації необхідне знаходження даної гіперплощини на достатній відстані від об'єктів. Використання такого алгоритму не потребує наявності великої навчальної вибірки, так як використовується не вся множина, а лише деяка частина об'єктів на межах областей, що може бути причиною помилок при класифікації. Принцип роботи алгоритму SVM зображений на рис. 1.4.

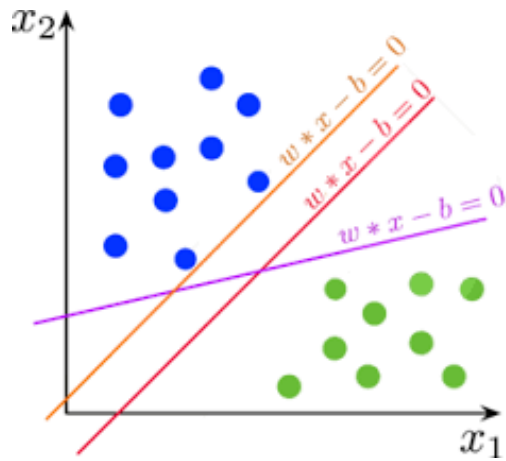


Рисунок 1.4. Алгоритм SVM

В основі роботи алгоритма Bag of words лежить знаходження ключових точок зображення і побудова нормованої гістограми, яка б описувала зображення. Недолік цього алгоритму полягає в тому, що він не враховує просторову інформацію об'єктів, тому можуть з'являтися помилки класифікації при наявності подібних ключових точок об'єктів. На рис. 1.5 зображений принцип роботи алгоритму Bag of words.

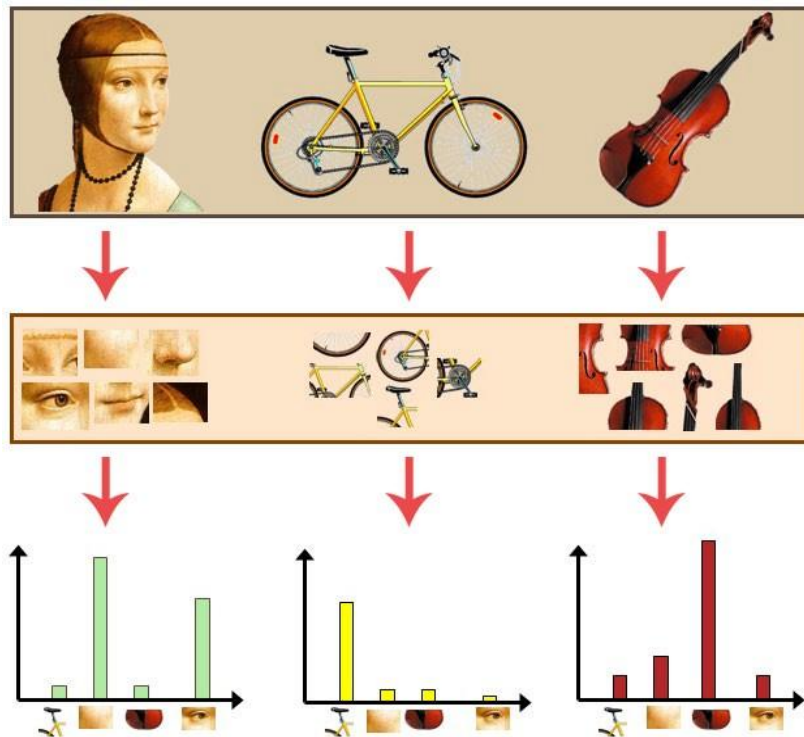


Рисунок 1.5. Принцип роботи алгоритму Bag of words

Дослідження застосовності нейронних мереж для класифікації ведуться з початку 1990-х років. Наукове співтовариство звернуло увагу на цей метод, тому що його використання, як правило, не вимагає апріорного знання про статистичний розподіл даних. Перші спроби застосування нейронних мереж для розпізнавання даних ДЗЗ показували обмежені можливості і вкрай низькі результати, однак моделювання нових архітектур дало поштовх використанню нейронних мереж.

Класичні повнозв'язні штучні нейронні мережі (багатошаровий перцептрон (рис. 1.6)), не дуже підходять для обробки зображень. Це пов'язано з тим, що зображення, як правило, мають великий розмір і складаються з великої кількості пікселів, тому необхідно, щоб нейронна мережа складалася з великої кількості нейронів та зв'язків. Ще один недолік полягає в тому, що класичні нейронні мережі не враховують двовимірну структуру зображень і погано запам'ятовують просторовий взаємозв'язок об'єктів та складових частин.

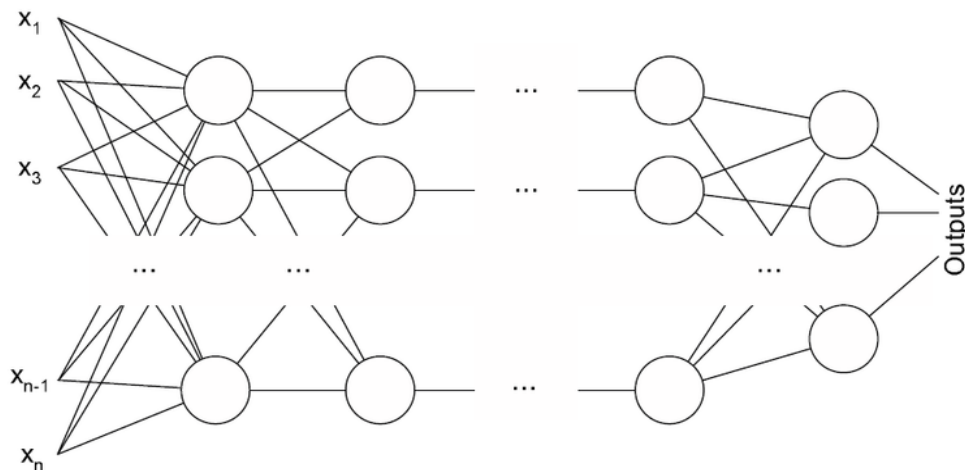


Рисунок 1.6. Багатошаровий перцептрон

В 1998 році був розроблений такий тип нейронних мереж, який добре підходить до обробки зображень. Такі нейронні мережі називають згортковими. Вони забезпечують високу стійкість до спотворення об'єктів, наявності шумів, зсувів і зміні ракурса знімання.

Згорткові нейронні мережі (ЗНМ) – це клас глибоких штучних нейронних мереж прямого поширення. ЗНМ використовують різновид багатошарових перцептронів, розроблений так, щоб вимагати використання мінімальної обробки. Це означає, що мережа навчається за допомогою фільтрів, що в традиційних алгоритмах доводиться виконувати вручну. Ця незалежність у конструюванні ознак від апріорних знань та людських зусиль є великою перевагою. Структура згорткових нейронних мереж зображена на рис. 1.7.

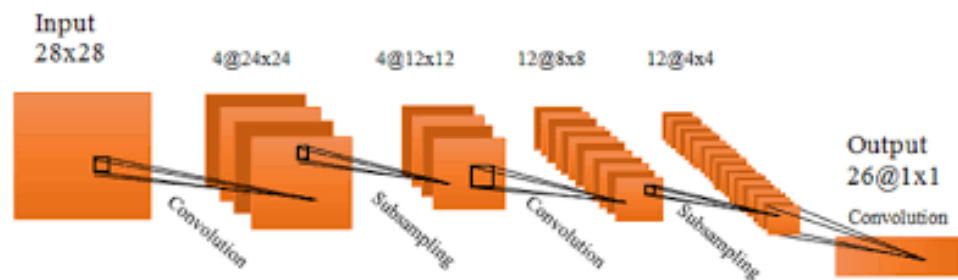


Рисунок 1.7. Структура згорткових нейронних мереж

Зображення, що подається на вхід, виступає в якості вхідного шару (рис. 1.8). Значення вхідних пікселів зображення нормалізується з метою пришвидшення результатів навчання. Нормалізовані значення пікселів визначаються залежно від типу використовуваної функції активації.

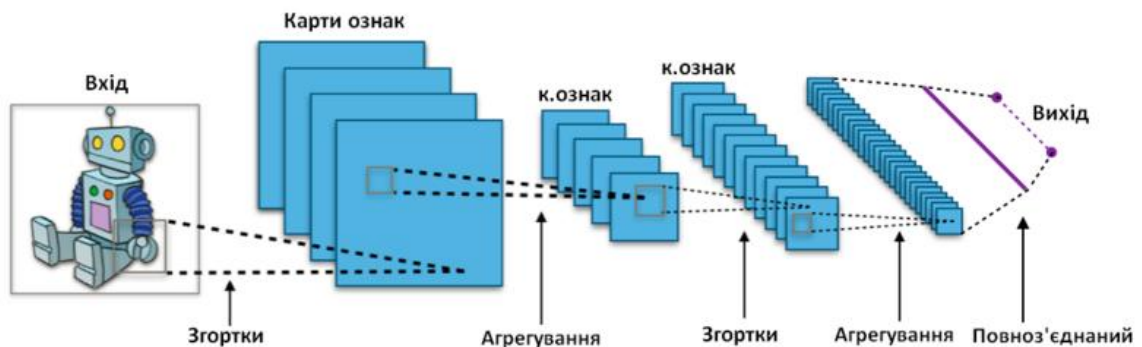


Рисунок 1.8. Згорткові нейронні мережі

Згортковий шар представляється у вигляді карт ознак. Операція згортки (рис. 1.9) – це обрахунок нового значення обраного пікселя, що є ключовим етапом в даній структурі. Для цього обчислення використовується матриця, що має назву ядро згортки. Ядро згортки – ковзне вікно, яке

проходить по всій області попередньої карти для виявлення певних ознак об'єкту.

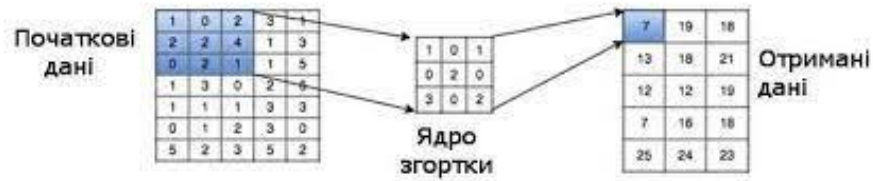


Рисунок 1.9. Операція згортки

Переваги ЗНМ є:

- порівняно з повнозв'язною нейронною мережею (наприклад, перцептрона) – набагато менше ваг, які необхідно налаштовувати, так як одне ядро використовується для всього зображення замість того, щоб обраховувати для кожного пікселя вхідного зображення власні вагові коефіцієнти;
- зручне розпаралелювання обчислень, а, отже, можливість реалізації алгоритмів роботи і навчання мережі на графічних процесорах;
- відносна стійкість до повороту та зрушення зображення, яке необхідно розпізнати;

### 1.3. Огляд існуючих рішень

«FireWatch» – це автоматизована система створена для виявлення лісових пожеж. Система функціонує шляхом розпізнавання диму. Кожні 6 хвилин сенсор розташований на висоті 10 м над лісом здійснює поворот на 360° та фіксує наявність диму. У випадку надходження ознак задимлення, сигнал тривоги надходить в центр управління, де оператор приймає рішення щодо подальшого реагування. Така система має ряд суттєвих недоліків: обмежена видимість, наявність сліпих зон, залежність від погодних умов тощо. Система «FireWatch» зображена на рис. 1.10.



Рисунок 1.10. Система «FireWatch»

На рис. 1.11 зображено порівняння системи «FireWatch» з класичними пристроями спостереження. Наприклад, IQ Firewatch фіксує незначне задимлення як вдень, так і вночі на великих відстанях (до 60 км), при цьому відеокамера здатна фіксувати лише густий дим переважно в світлу частину доби. Тепловізор здатний фіксувати пожежу лише на стадії відкритого вогню.

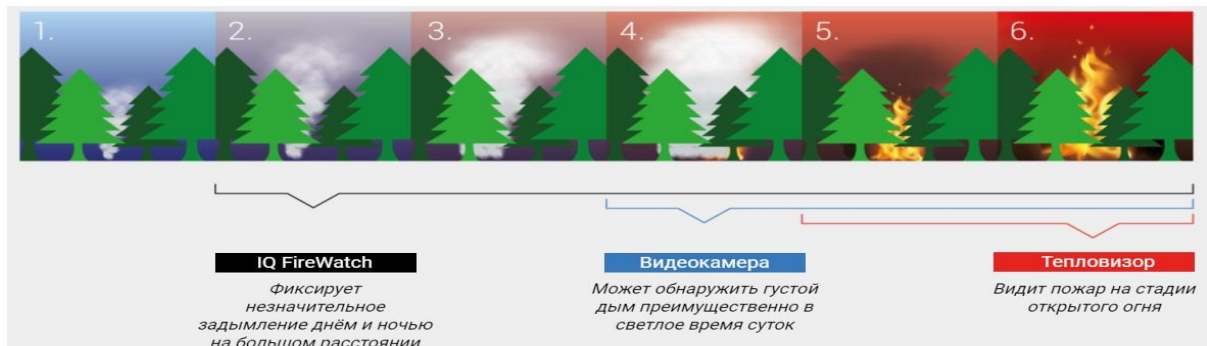


Рисунок 1.11. Порівняння системи «FireWatch» та класичних пристроїв спостереження

Досить ефективною виявилася система знаходження лісових пожеж, яка базується на штучній нейронній мережі і реалізована за допомогою бездротової сенсорної мережі (рис. 1.12). Вчені університету науки і технологій Китаю припустили, що використання системи, яка залежить лише від одного критерію лісової пожежі є малоефективним та збільшує

ймовірність хибних спрацювань. Саме тому її алгоритм враховує 11 різноманітних атрибутів. Хибні тривоги відбувалися через те, що фотоелектричний блок знаходження диму був чутливий до білих частинок від тліючого вогню і пилу. Модуль знаходження пожежі відповідає за прийняття рішення і надсилання сигналу тривоги. Він містить п'ять сенсорних вузлів, чотири з яких збирають дані про можливе розгортання пожежі. Нейронна мережа реалізована на кожному окремому сенсорному вузлі. П'ятий вузол датчика працює як базова станція і збирає дані зондування і пожежну тривогу від інших чотирьох вузлів. Для того, щоб безперервно використовувати систему датчиків було прийняте рішення про установку сонячних батарей, оскільки це був найдешевший та найекологічніший варіант. Оскільки в лісі доступне лише переривчасте сонячне світло, вченими була розроблена спеціальна модель сонячної батареї. Енергія сонячної панелі накопичується суперконденсатором. Модуль користувацького інтерфейсу відповідає за відображення даних, зчитаних з датчиків. Дані зондування і пожежна сигналізація передаються від базової станції до користувача.

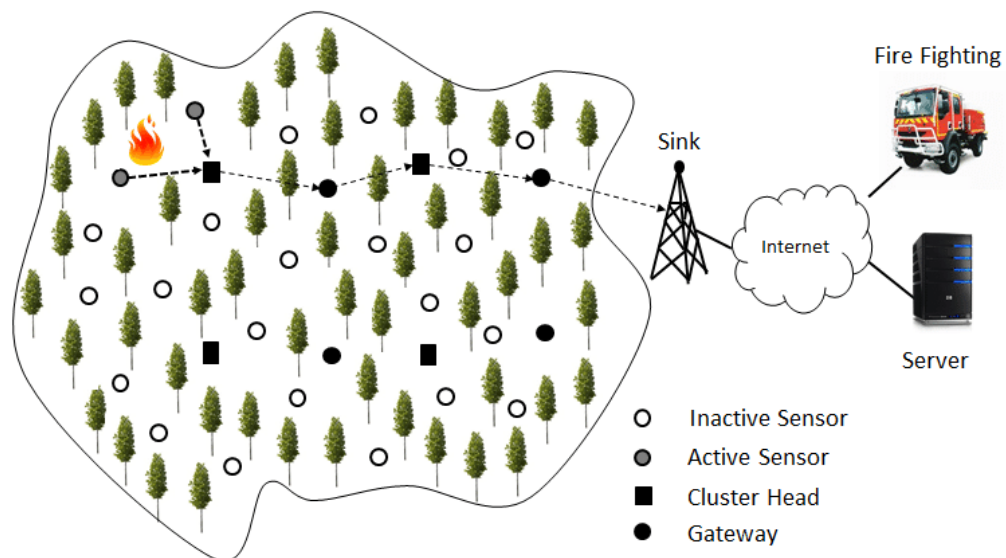


Рисунок 1.12. Система розпізнавання місця розгортання пожежі

Вчені з Тайванського університету запропонували алгоритм знаходження вогню на основі зображень з використанням нейронних мереж зворотного поширення. Цей метод складається з трьох етапів: сегментація моделі кольорового полум'я, видалення хибних областей та оцінка небезпеки. На першому етапі три компоненти HSI моделі кожного пікселя, а саме тон, насиченість та світло, подаються на навчання нейронної мережі. Якщо піксель не відноситься до знайденої області полум'я, він зафарбовується чорним кольором, в іншому випадку колір залишається незмінним. Хибними полями може бути відображення полум'я та об'єкти подібних кольорів. В першій ситуації область можна відфільтрувати шляхом видалення пікселів з найменшою інтенсивністю, а в другій враховується фон об'єкту: якщо він аналогічний або дуже подібний, то метод різниці видалить об'єкт із зображення. Обробка зображення нейронною мережею представлена на рис. 1.13.



Рисунок 1.13. Зображення полум'я до та після обробки нейронною мережею

#### 1.4. Аналіз зацікавлених сторін

##### 1. Внутрішні зацікавлені сторони (особи, що приймають участь у функціонуванні системи):

- розробник системи;
- землевласники та керуючі лісовими ділянками;
- пожежна охорона.

2. Зовнішні зацікавлені сторони (особи, які не приймають участь у безпосередньому функціонуванні системи):

- конкуренти;
- партнери;
- екологи;
- міністерство захисту довкілля та природних ресурсів України.

### 1.5. Постановка задачі

*Мета роботи:* проведення бінарної класифікації даних дистанційного зондування Землі за допомогою згорткових нейронних мереж для визначення місця розгортання пожежі.

*Об'єкт роботи:* класифікація даних дистанційного зондування Землі.

*Предмет роботи:* класифікація даних дистанційного зондування Землі за допомогою згорткових нейронних мереж.

*Завданнями кваліфікаційної роботи є:*

- проведення аналізу літературних та інших джерел щодо сучасного стану дешифрування даних ДЗЗ та використання нейронних мереж для їх класифікації;
- опанування процесу отримання даних ДЗЗ, опис їх характеристик;
- розробка та тестування нейронної мережі для розв'язання задачі класифікації;
- аналіз отриманих результатів.

*Функціональні вимоги:*

- тематична обробка даних ДЗЗ;
  - модифікація яскравості та контрастності зображення;
  - зменшення кількості шумів;
  - збільшення роздільної здатності;
- класифікація місця пожежі на знімку;
  - розпізнавання місця розгортання пожежі;

- оцінка масштабу пожежі.

*Нефункціональні вимоги:*

- визначення швидкості розпізнавання пожежі на знімках;
- визначення точності розпізнавання.

Схема для відображення функціональних та нефункціональних вимог зображена на рис. 1.14.

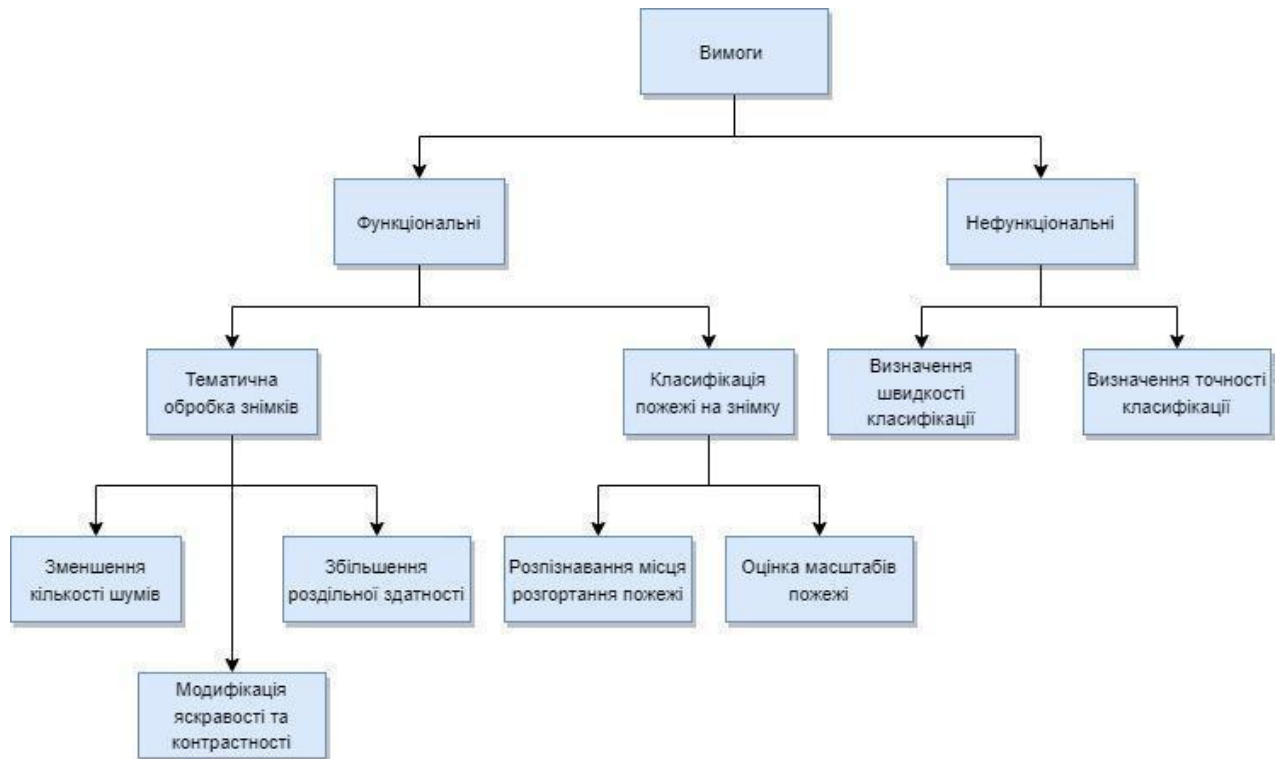


Рисунок 1.14. Функціональні та нефункціональні вимоги

Застосування нейронних мереж під час розробки даної системи дає змогу значно підвищити рівень автоматизації в умовах великої кількості даних.

## РОЗДІЛ 2. РОЗРОБКА АРХІТЕКТУРИ КЛАСИФІКАЦІЇ ПОЖЕЖІ ЗА ДОПОМОГОЮ МЕТОДІВ МАШИННОГО НАВЧАННЯ

### 2. 1. Узагальнена структура системи класифікації даних ДЗЗ

Після отримання супутникових знімків в першу чергу проводяться операції по покращанню зображень та приведення їх до єдиного вигляду. Дані операції виконуються з метою покращання якості зображень, що, в свою чергу, призводить до підвищення точності класифікації. Існує три основних етапи покращання даних ДЗЗ:

1. Радіометрична корекція. Даний етап передбачає приведення знімку (або ж серії знімків) до єдиної яскравості та контрастності з метою кращої візуалізації.
2. Атмосферна корекція. Етап передбачає поліпшення характеристик знімку шляхом усунення перешкод, що завдає атмосфера Землі (збільшення роздільної здатності).
3. Геометрична корекція. Даний етап передбачає трансформацію зображення в необхідну систему координат.

Наступним етапом є безпосередньо класифікація отриманих знімків і завершальним етапом є проведення порівняльного аналізу отриманих результатів.

На рис. 2.1 представлена узагальнена схема процесів системи.

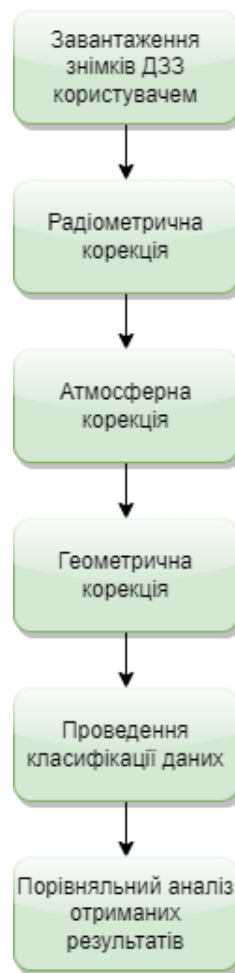


Рисунок 2.1. Узагальнена схема процесів системи

## 2.2. Функціональний аналіз класифікації даних ДЗЗ

Для отримання та завантаження даних ДЗЗ використовується геопортал геологічної служби США. Портал даних USGS GloVis (рис. 2.2) – це потужний ресурс для отримання геопросторових даних.

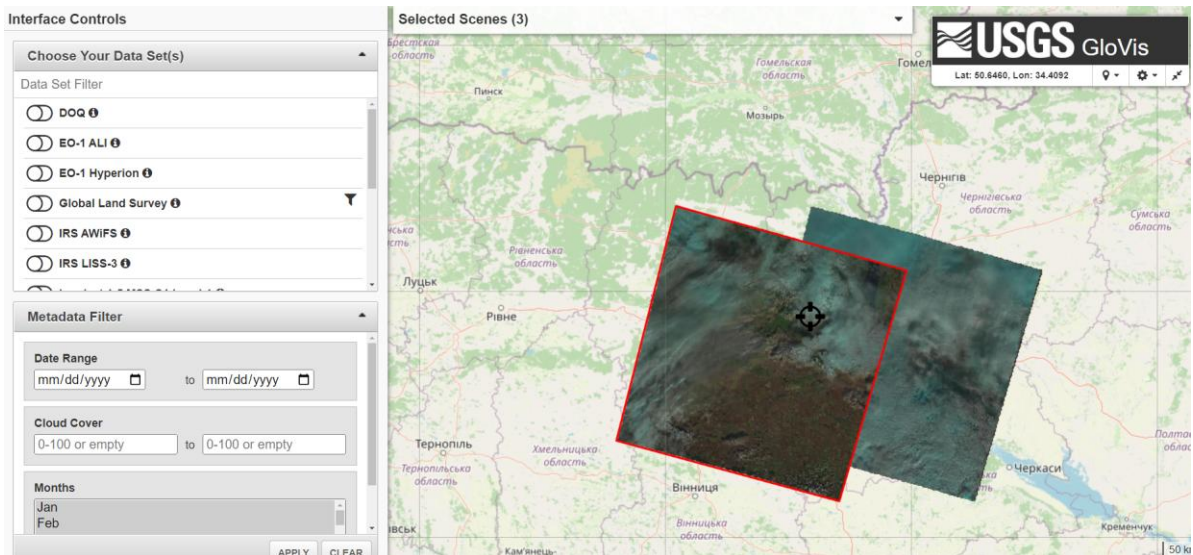


Рисунок 2.2. Портал даних USGS GloVis

Для завантаження архіву з обраними знімками необхідно обрати полігон, який буде входити в знімок, додатково можна вказати часовий проміжок за яким фільтрувати знімки. Також користувач має змогу вибрати необхідний супутник зі списку. В даній кваліфікаційній роботі використовуватимуться знімки, зроблені за допомогою супутника Landsat-8. Додатково користувач має змогу фільтрувати знімки за рівнем хмарності, обирати пору року впродовж якої були зроблені знімки, час зйомки впродовж доби тощо. На рис. 2.3 зображений знімок Київської області з порталу GloVis, зроблений за допомогою супутника Landsat 8.

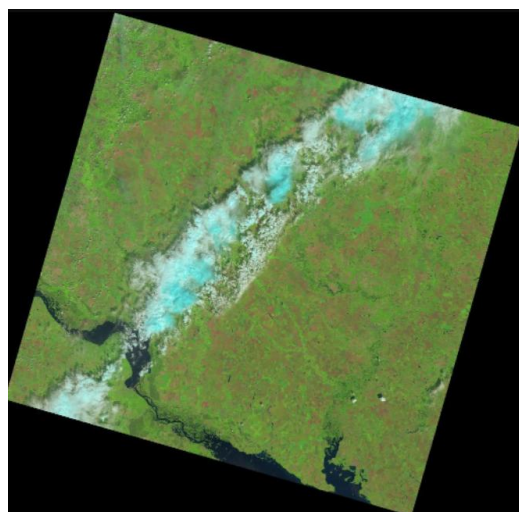


Рис. 2.3. Знімок Київської області з Landsat 8

Для тематичної обробки завантажених зображень доцільно використовувати геоінформаційні системи.

Геоінформаційні системи (ГІС) – це інформаційні системи, призначені для збирання, зберігання, аналізу та візуалізації просторових даних. Для виконання задач тематичної обробки в даній роботі була обрана система QGIS. QGIS здатна вирішувати необхідні задачі тематичної обробки. За потреби її функціонал може бути розширений за допомогою додаткової архітектури модулів. В кваліфікаційній роботі використовуватиметься додатковий модуль Semi-Automatic Classification (SCP).

Отже, функції системи класифікації даних ДЗЗ:

- завантаження знімків з GloVis;
- тематична обробка даних в QGIS:
  - радіометрична корекція;
  - атмосферна корекція (збільшення роздільної здатності);
  - геометрична корекція;
- класифікація зображень за допомогою згорткових нейронних мереж:
  - навчання мережі;
  - налаштування вагових коефіцієнтів;
- аналіз та порівняння отриманих результатів;
  - аналіз точності проведеної класифікації;
  - аналіз швидкості проведеної класифікації.

Узагальнене дерево функцій системи класифікації даних ДЗЗ представлено на рис. 2.4.

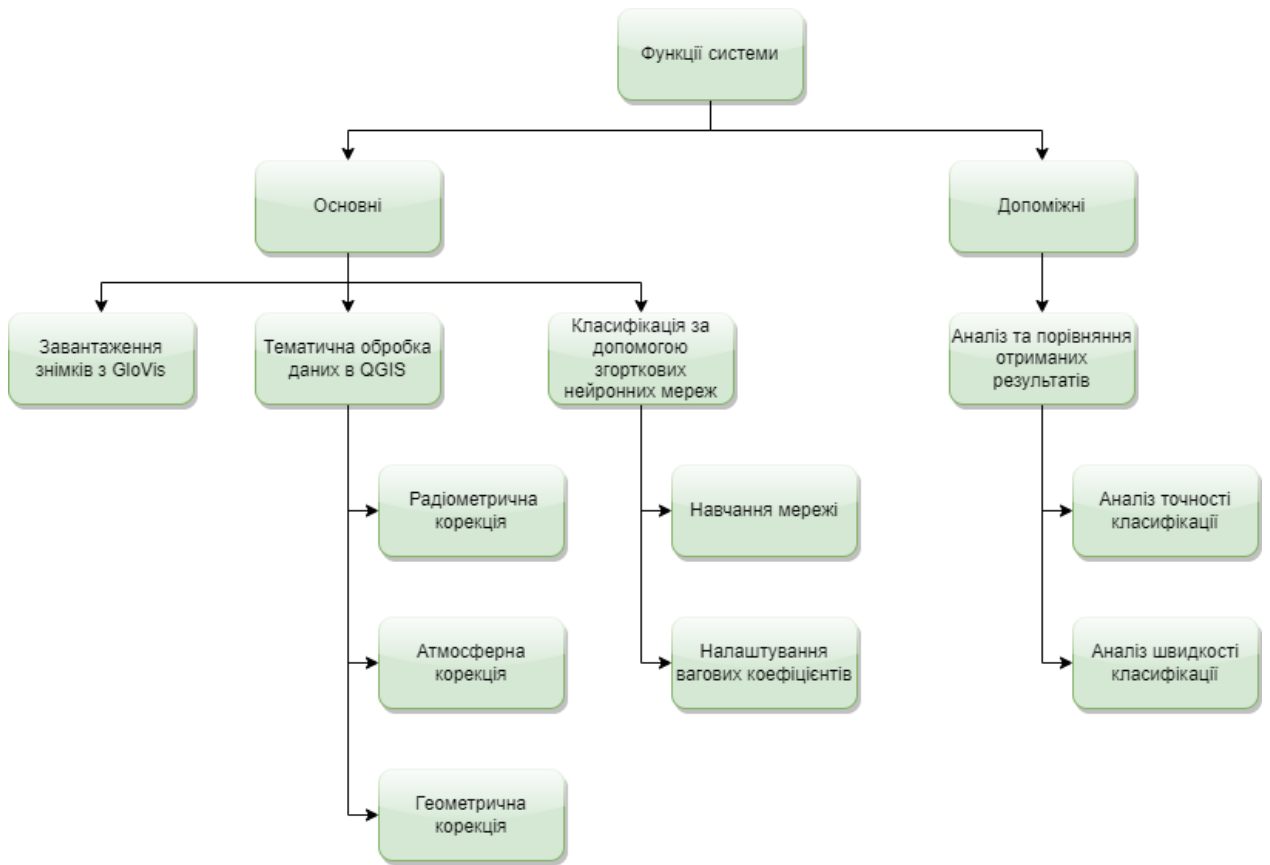


Рисунок 2.4. Узагальнене дерево функцій системи

Архітектура системи класифікації даних зображена на рис. 2.5.

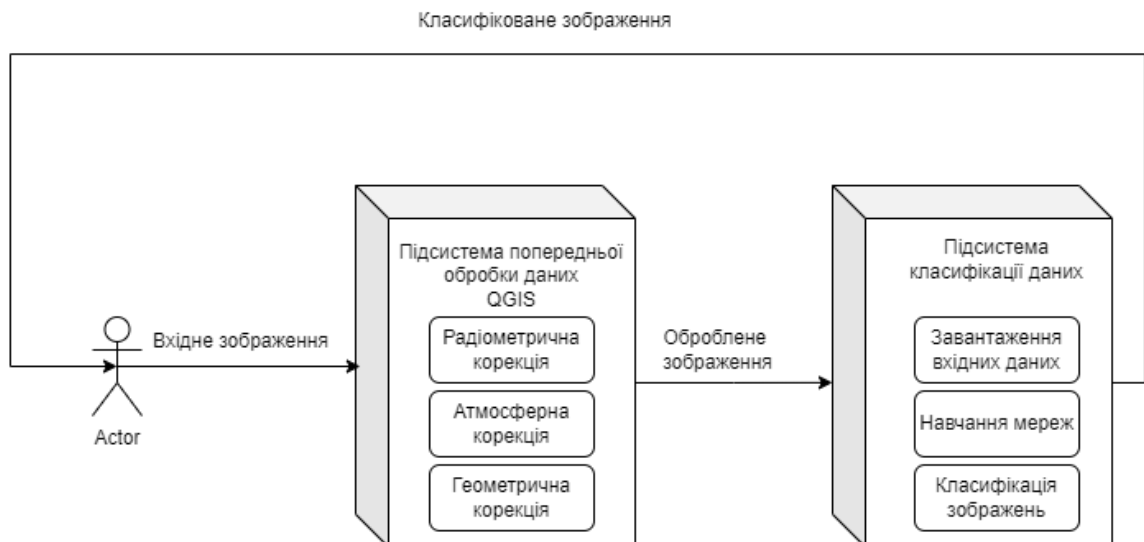


Рисунок 2.5. Архітектура системи класифікації даних

## 2.3 Вибір нейронної мережі

Для виконання задачі класифікації використовуються дві згорткові нейронні мережі для проведення подальшого аналізу та порівняння їх ефективності, а саме згорткові нейронні мережі U-Net та LinkNet.

### 2.3.1 Нейронна мережа U-Net

U-Net – згорткова нейронна мережа, яка представляє собою послідовність шарів згортки та пулінгу. Спершу U-Net була створена для сегментації біомедичних зображень. Архітектура представляє собою повнозв'язну згорткову мережу, модифіковану так, щоб вона могла працювати з меншою кількістю прикладів і давала більш точну сегментацію. Навчається мережа ковзним методом.

Характерні риси U-Net:

- високі результати для різного типу задач;
- використання невеликої кількості даних для отримання високих результатів.

Архітектура нейронної мережі U-Net наведена на рис. 2.6.

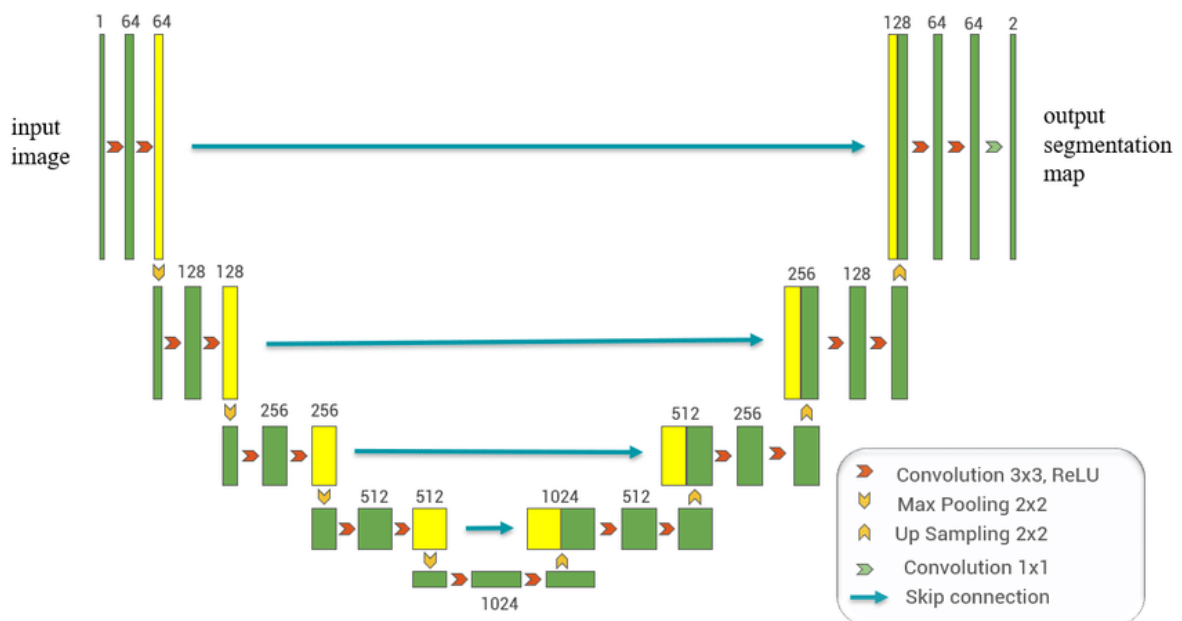


Рисунок 2.6 Архітектура U-Net

Дана нейронна мережа складається з кодувальника (encoder) та декодувальника (decoder). Кодувальник має стандартну архітектуру згорткової нейронної мережі. Натомість, декодувальник складається з операцій підвищення дискредитації карти властивостей, за якою слідує:

- згортка;
- об'єднання з відповідним образом;
- дві згортки;
- функція активації ReLu.

Нижче на рис. 2.7 наведено приклад семантичної сегментації зображень за допомогою нейронної мережі U-Net.

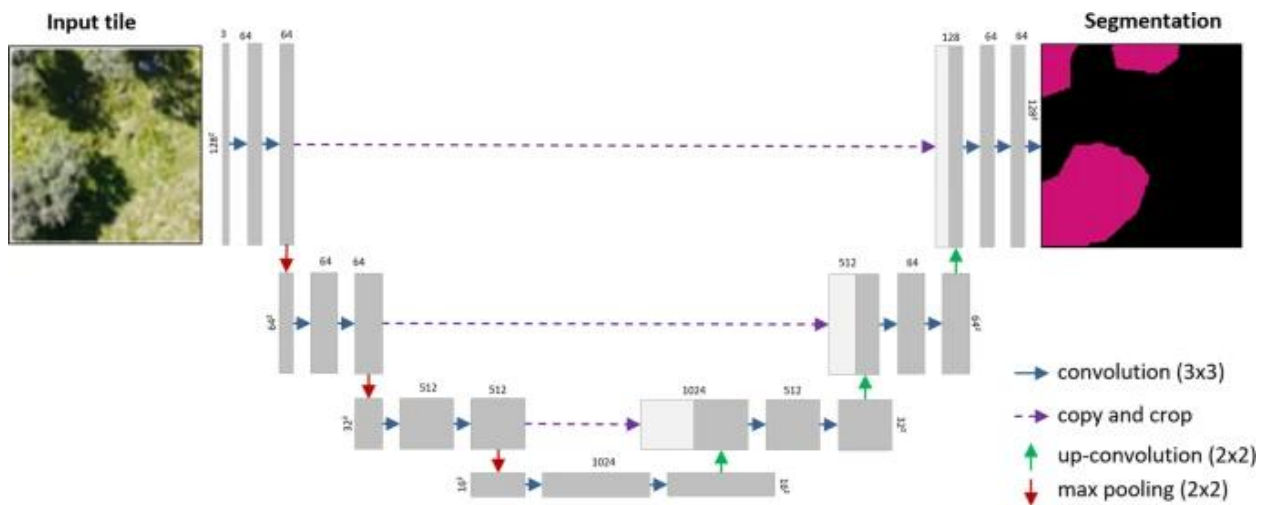


Рисунок 2.7 Сегментоване зображення за допомогою U-Net

### 2.3.2 Нейронна мережа Link-Net

Link-Net складається з кодувальника (encoder) та декодувальника (decoder), за допомогою яких відбувається розбиття зображення. В ході роботи даної нейронної мережі також відбувається створення резервного копіювання зображення перед проходженням декількох остаточних шарів.

Архітектура Link-Net наведена на рис. 2.8.

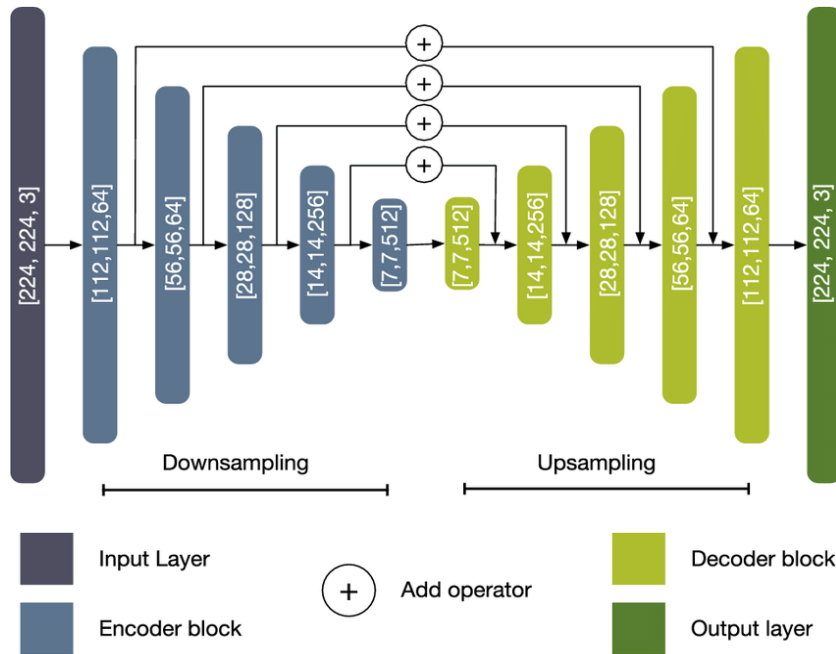


Рисунок 2.8 Архітектура Link-Net

Ліва частина мережі – є кодувальником, права – декодувальником. Початок мережі – блок, що реалізує функцію згортки. Додаючи вихідні дані кодувальника до декодувальника, продуктивність мережі зростає, оскільки це дозволяє декодувальнику швидше відновлювати інформацію шарів блоку кодувальника.

Головною відмінністю даної мережі від U-Net є те, що використовується операція додавання значень відповідних каналів. При тому, U-Net використовує операцію конкатенації, що призводить до збільшення каналів. На рис. 2.9 зображено схеми U-Net та LinkNet, та їх відмінність.

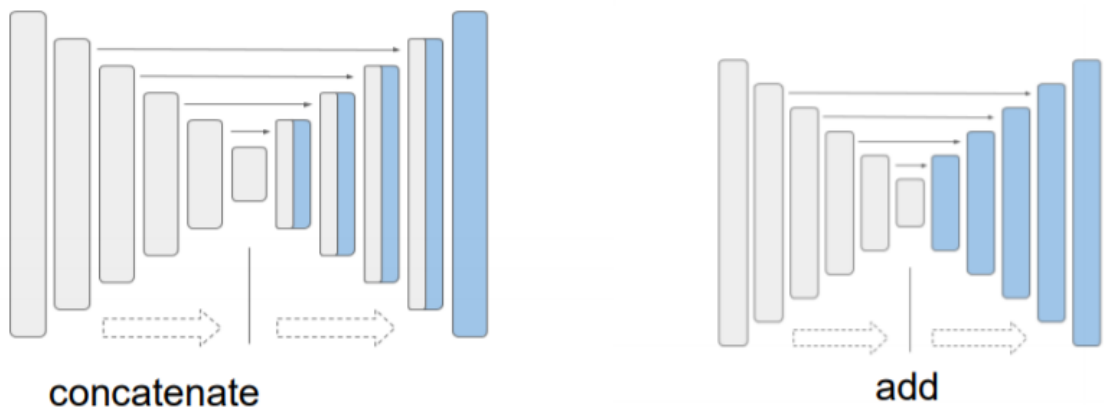


Рисунок 2.9 Відмінність мережі U-Net (зліва) та LinkNet (справа)

## 2.4. Вибір функції активації

Одним із важливих етапів розробки нейронної мережі є вибір функції активації нейронів шару. Функція активації – це нелінійне перетворення вхідного сигналу. Цей перетворений вихід потім надсилається наступному шару нейронів на вхід. Вибір функції активації в основному визначає метод навчання та функціональні можливості системи. Класичний алгоритм зворотного розповсюдження помилки добре працює на нейронних мережах з двома і трьома шарами, але при збільшенні глибини карти виникають труднощі (наприклад, загасання градієнтів). Тому по мірі поширення похибки від вихідного шару до вхідного на кожному шарі відбувається додаткове множення поточного результату на похідну функції активації. На рис. 2.10. представлені деякі найбільш поширені функції активації.

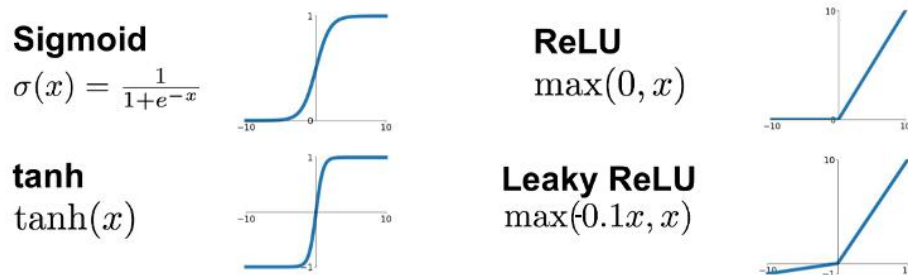


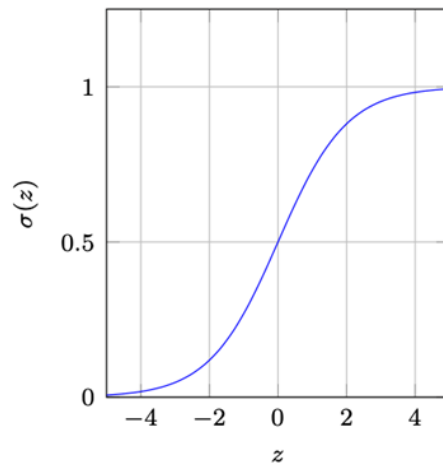
Рисунок 2.10. Функції активації

Для заданої задачі класифікації були обрані функції активації Sigmoid та Softmax.

Введення функцій сигмоїдального типу було обумовлене обмеженістю нейронних мереж із пороговою функцією активації нейронів - за такої функції активації будь-який із виходів мережі дорівнює або нулю, або одиниці, що обмежує використання мереж не в задачах класифікації. Використання сигмоїдальних функцій дозволило перейти від бінарних виходів нейрона до аналогових. Сигмоїдна функція наступну формулу:

$$f(x) = \frac{L}{1 + e^{-kx}} \quad (2.1)$$

Константа  $L$  визначає значення максимуму кривої, а константа  $k$  впливає на крутість переходу. Графік функції активації Sigmoid зображений на рис. 2.11.



(a) Sigmoid activation function.

Рисунок 2.11. Графік функції активації Sigmoid

Переваги використання Sigmoid:

- гладка функція, зручна для виведення;
- дані можуть бути стиснуті, щоб гарантувати відсутність проблем із амплітудою даних;

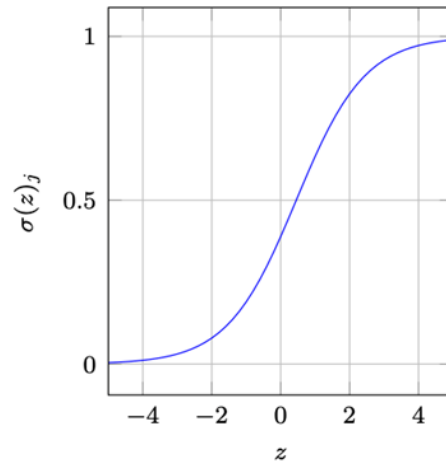
Softmax – узагальнена логістична функція, яка складається з декількох класів. Для подібної класифікації нейромережа будується таким чином, що на останньому шарі кількість нейронів є рівною кількості шуканих класів. Кожний нейрон видає значення ймовірності приналежності об'єкта до класу, тобто значення між нулем та одиницею, а всі нейрони в сумі дають одиницю.

Формула Softmax виглядає наступним чином:

$$f(x)_i = \frac{e^{x_i}}{\sum_{k=1}^K e^{z_k}}, \quad (2.2)$$

де  $K$  – кількість класів.

Графік функції активації Softmax зображений на рис. 2.12.



(b) Softmax activation function.

Рисунок 2.12. Графік функції активації Softmax

Переваги використання Softmax:

- здатність обробляти декілька класів в моделях логістичної регресії;
- корисність для вихідних нейронів.

## РОЗДІЛ 3. СТВОРЕННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ КЛАСИФІКАЦІЇ ТА ЇХ ПОРІВНЯЛЬНИЙ АНАЛІЗ

### 3.1. Попередня обробка супутникових знімків

За допомогою порталу даних USGS GloVis був обраний та завантажений супутниковий знімок, що містить місце розгортання пожежі. Для дослідження був обраний знімок території на сході Австралії, а саме територія Національного парку Вулемі, де в 2019 році спалахнули пожежі катастрофічних масштабів (рис. 3.1). Зміст завантаженого архіву зображений на рис. 3.2.

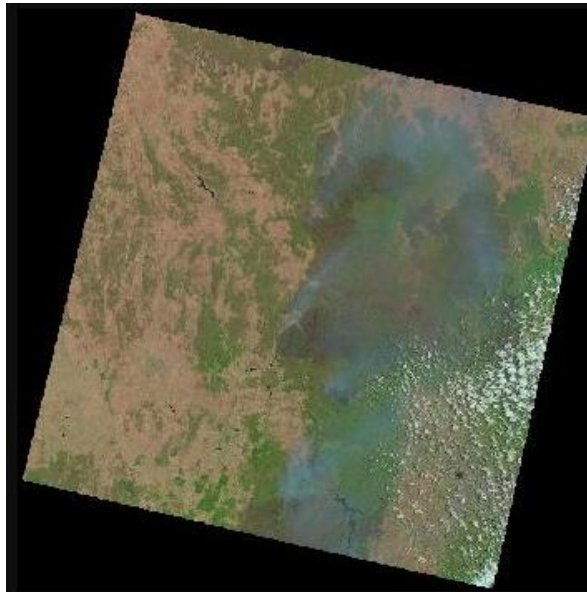


Рисунок 3.1. Знімок пожежі за допомогою Landsat-8

```
LC08_L1TP_090083_20191215_20201023_01_T1_ANG  
LC08_L1TP_090083_20191215_20201023_01_T1_B1  
LC08_L1TP_090083_20191215_20201023_01_T1_B2  
LC08_L1TP_090083_20191215_20201023_01_T1_B3  
LC08_L1TP_090083_20191215_20201023_01_T1_B4  
LC08_L1TP_090083_20191215_20201023_01_T1_B5  
LC08_L1TP_090083_20191215_20201023_01_T1_B6  
LC08_L1TP_090083_20191215_20201023_01_T1_B7  
LC08_L1TP_090083_20191215_20201023_01_T1_B8  
LC08_L1TP_090083_20191215_20201023_01_T1_B9  
LC08_L1TP_090083_20191215_20201023_01_T1_B10  
LC08_L1TP_090083_20191215_20201023_01_T1_B11  
LC08_L1TP_090083_20191215_20201023_01_T1_BQA  
LC08_L1TP_090083_20191215_20201023_01_T1_MTL
```

Рисунок 3.2. Зміст завантаженого архіву

Завантажений архів містить 11 одноканальних знімків, кожен з яких відповідає спектральному каналу супутника Landsat-8. Характеристики спектральних каналів (довжина хвилі, роздільна здатність) супутника Landsat-8 представлена в табл. 3.1.

*Таблиця 3.1* Характеристики спектральних каналів супутника Landsat-8

Канали	Довжина хвилі [мікрометри]	Роздільна здатність [метри]
1 - Узбережний аерозоль	0,43 – 0,45	30
2 - Синій	0,45 – 0,51	30
3 - Зелений	0,53 – 0,59	30
4 - Червоний	0,64 – 0,67	30
5 - Близький інфрачервоний (NIR)	0,85 – 0,88	30
6 - SWIR 1	1,57 – 1,65	30
7 - SWIR 2	2,11 – 2,29	30
8 - Панхроматичний	0,50 – 0,68	15
9 - Пір'їсті хмари	1,36 – 1,38	30
10 - Тепловий інфрачервоний (TIRS) 1	10,60 – 11,19	100
11 - Тепловий інфрачервоний (TIRS) 2	11,50 – 12,51	100

Одноканальні знімки з архіву завантажуються у геоінформаційну систему QGIS з метою їх подальшої обробки. Приклад знімків представлений на рис. 3.3.

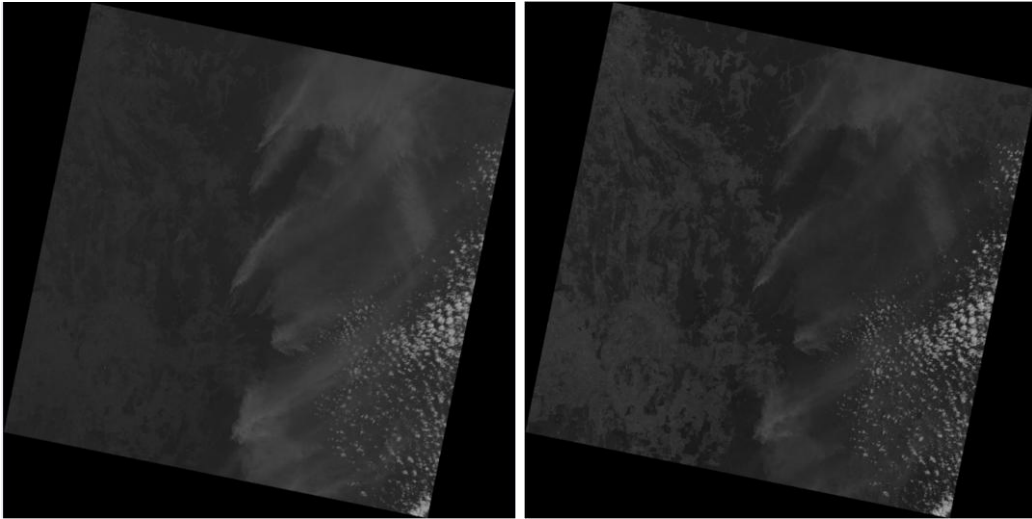


Рисунок 3.3. Знімки синього (зліва) та червоного (справа) спектральних каналів

З використанням встановленого модулю SCP та за допомогою додаткової процедури атмосферної корекції було збільшено роздільну здатність знімків з 30 м до 15 м (рис. 3.4).

Landsat conversion to TOA reflectance and brightness temperature

Directory containing Landsat bands: C:/pr1

Select MTL file: C:/pr1/LC08\_L1TP\_090083\_20191215\_20201023\_01\_T1\_MTL.txt

Brightness temperature in Celsius

Apply DOS1 atmospheric correction

Perform pansharpening (Landsat 7 or 8)

Create Band set and use Band set tools  Add bands in a new Band set

Metadata

Satellite: LANDSAT\_8 Date (YYYY-MM-DD): Sun elevation:

	Band	RADIANCE_MULT	RADIANCE_ADD	REFLECTANCE_MULT	REFLECTANCE_ADD
1	LC08_L1TP_090083_20191215_20201023_01_T...	1.3274E-02			
2	LC08_L1TP_090083_20191215_20201023_01_T...	1.2232E-02			
3	LC08_L1TP_090083_20191215_20201023_01_T...	1.0315E-02			
4	LC08_L1TP_090083_20191215_20201023_01_T...	6.3121E-03			
5	LC08_L1TP_090083_20191215_20201023_01_T...	1.5698E-03			
6	LC08_L1TP_090083_20191215_20201023_01_T...	5.2909E-04			

Рисунок 3.4. Покращання роздільної здатності знімків

Підготовлені растри зі збільшеною роздільною здатністю об'єднуються в один багатоспектральний растр. Кількість каналів, які об'єднуються в растр, залежить від того, для яких цілей буде в подальшому використано цей багатоканальний знімок. Найчастіше об'єднуються канали 1-7 або 2-7. Залежно від досліджуваних об'єктів або явищ, необхідно обрати оптимальну комбінацію каналів для найкращого відображення. Для цього обирають три зональних зображення, кожне з яких розглядають як червону, зелену та синю

складові простору RGB. Застосування різних комбінацій каналів дозволяє створювати зображення, які здатні підкреслювати ті чи інші особливості об'єктів.

Для класифікації пожежі найчастіше використовують комбінації 7-5-3 та 7-6-4.

Комбінації 7-5-3 та 7-6-4 відображають знімки в «природних кольорах», тобто об'єкти земної поверхні виглядають подібними на те, як вони сприймаються людським оком, але при цьому ці комбінації дозволяють аналізувати стан атмосфери та дим. В основному вони застосовуються для моніторингу пожеж, вивчення їх динаміки та для пост-пожежного аналізу.

На рис. 3.5 зображені комбінації каналів 7-6-4 (зліва) та 7-5-3 (справа) багатоканального знімку.

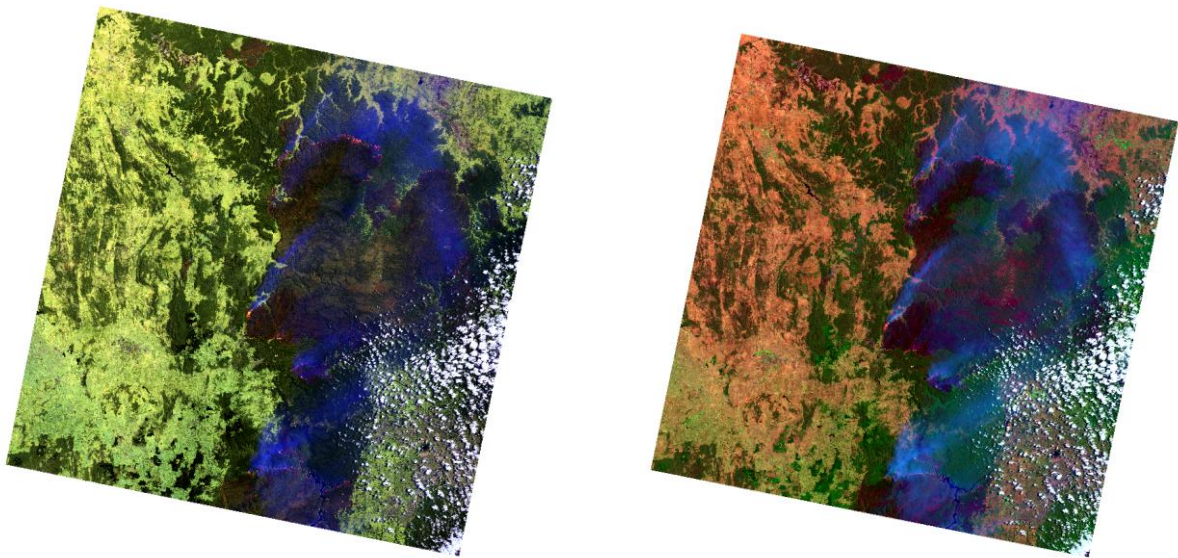


Рисунок 3.5. Комбінації каналів 7-6-4 (зліва) та 7-5-3 (справа)

Для подальшої роботи використовується багатоканальний знімок з комбінацією каналів 7-5-3. Для отриманого знімку необхідно також провести процедуру радіометричної корекції (налаштування яскравості, контрастності). В результаті виконаних перетворень отримаємо знімок, зображений на рис. 3.6.

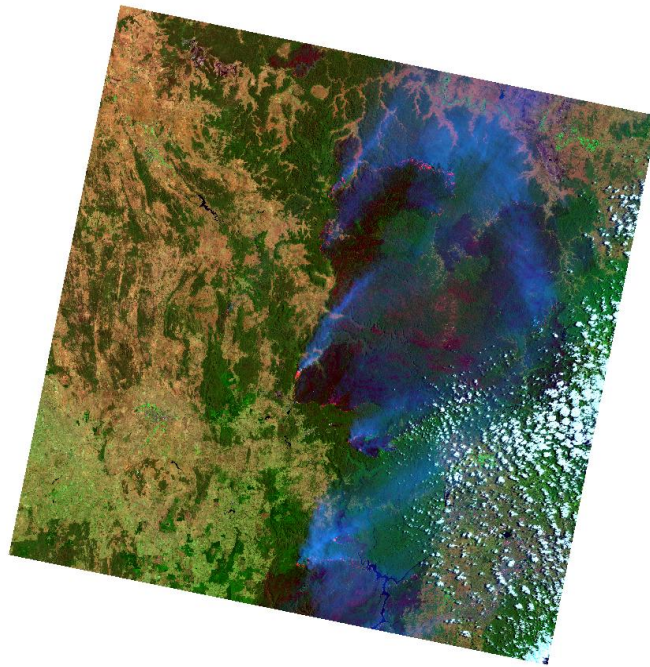


Рисунок 3.6. Багатоканальний знімок

### 3.2. Формування маски для навчання нейронних мереж

Для формування маски для навчання нейронних мереж необхідно на багатоканальному знімку визначити еталонні ділянки, які найкраще репрезентують пожежі (або задимлення).

Для створення маски було проведено класифікацію знімка за допомогою модуля SCP. Також була створена карта індексу NDVI, яка матиме допоміжну роль при визначенні еталонних ділянок в процесі класифікації об'єктів на знімку.

Індекс NDVI (Normalized Difference Vegetation Index) – нормалізований відносний вегетаційний індекс – простий кількісний показник кількості здатної до фотосинтезу біомаси. NDVI обчислюється за наступною формулою:

$$NDVI = \frac{NIR - RED}{NIR + RED} \quad (3.1)$$

де NIR – відображення в ближній інфрачервоній області спектра, а RED – відображення в червоній області спектра.

Можливі значення NDVI лежать в діапазоні від -1 до 1. Значення NDVI збільшується у випадку наявності великої кількості біомаси та зменшується у випадку її висихання, вирубки або ж внаслідок вигорання.

Карта індексу NDVI створюється засобами QGIS через вбудований калькулятор растрів. Процес обчислення індексу NDVI зображено на рис. 3.7.

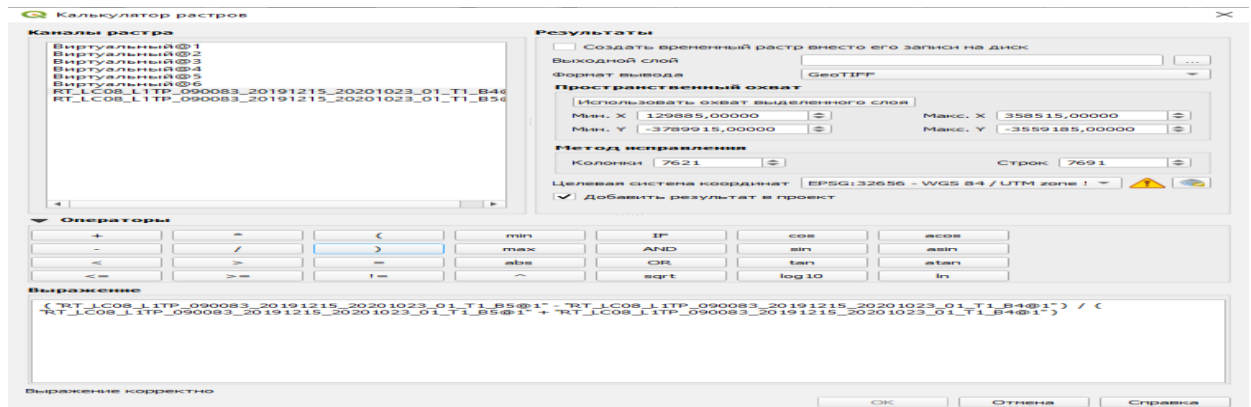
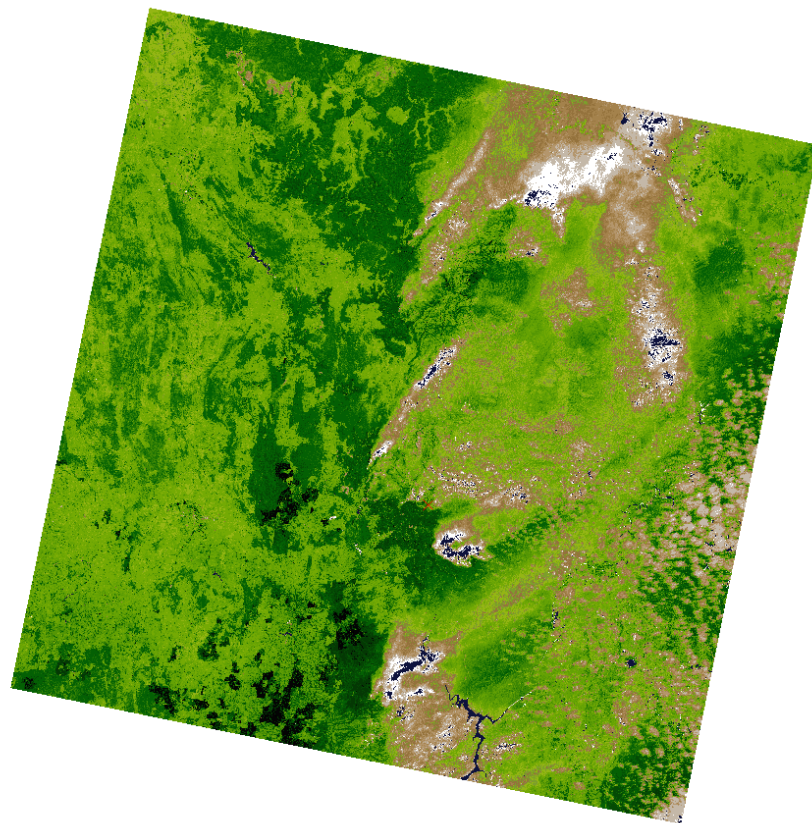


Рисунок 3.7. Обчислення індексу NDVI через вбудований калькулятор растрів

В результаті виконаних обчислень була отримана карта індексу NDVI, яка представлена на рис. 3.8.



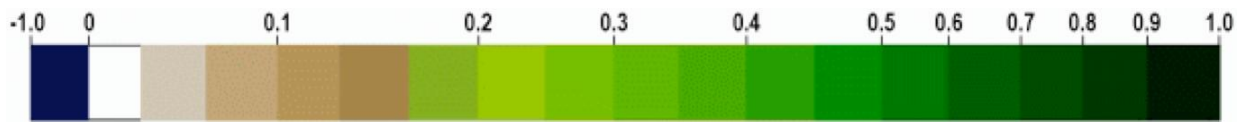


Рисунок 3.8. Карта індексу NDVI

Для отримання маски для навчання НМ необхідно виконати контрольовану класифікацію багатоканального знімку методом мінімальних відстаней за допомогою модуля SCP.

Контрольована класифікація (також відома як класифікація з навчанням) передбачає віднесення пікселя на знімку до відомого класу земних об'єктів, якому відповідає попередньо визначений спектральний образ в багатовимірному просторі спектральних ознак.

Обрано 7 тематичних класів. Три класи описані безпосередньо для класифікації пожежі (дим, місця займання пожежі, уражені довготривалою пожежею ділянки). Для кожного класу потрібно обрати еталони (рис. 3.8). Еталони визначають якість навчальної вибірки та, як наслідок, точність контрольованої класифікації.



Рисунок 3.8. Вибір еталонних ділянок для репрезентації лісу

Тематичні класи представлені на рис. 3.9.

SCP &Dock

Home C:/diploma/main.scp

ROI & Signature list

Filter

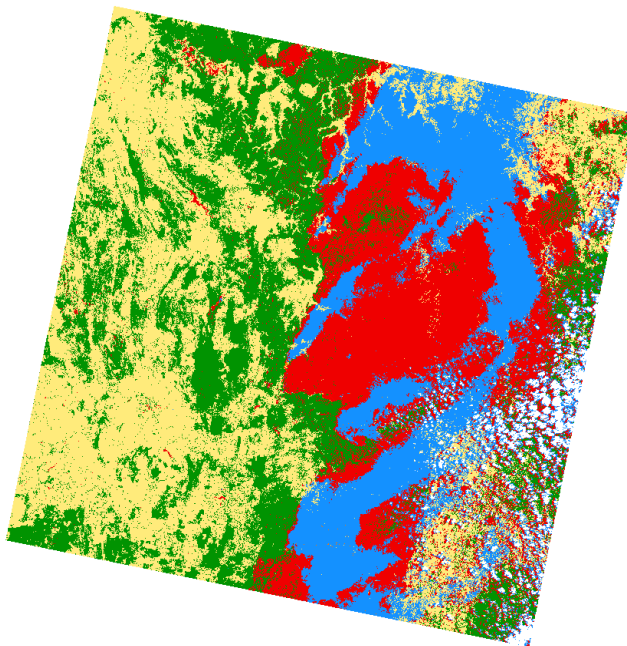
MC ID	C ID	Name	Type	Color
- 1		<b>MC 1</b>		
<input checked="" type="checkbox"/>	1	smoke	R&S	Blue
<input checked="" type="checkbox"/>	2	land1	R&S	Yellow
<input checked="" type="checkbox"/>	3	forest	R&S	Green
<input checked="" type="checkbox"/>	4	land2	R&S	Yellow
<input checked="" type="checkbox"/>	5	cloud	R&S	White
<input checked="" type="checkbox"/>	6	fire	R&S	Orange
<input checked="" type="checkbox"/>	7	burning area	R&S	Red

Training input

ROI options

### Рисунок 3.9. Описані тематичні класи

Результат проведеної класифікації представлений на рис. 3.10.



### Рисунок 3.10. Результат проведеної класифікації

Отриманий знімок перетворюється на бінарний растр таким чином, що всі класи, що відносяться до пожежі (дим, місця займання пожежі, уражені довготривалою пожежею ділянки), позначені білим кольором, а інші – чорним (рис. 3.11). Таким чином, отримується бінарне зображення – маска, яка буде використовуватись для навчання нейронних мереж.

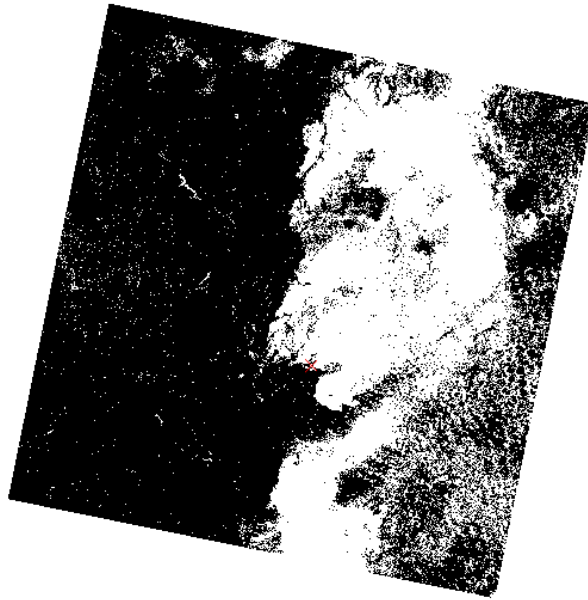


Рисунок 3.11. Маска для навчальної вибірки

Для формування вхідних даних необхідно поділити попередньо оброблений багатоканальний знімок та його бінарну маску на фрагменти. Створена маска та багатоканальний знімок попарно поділяються на фрагменти для формування вхідних даних (рис. 3.12).

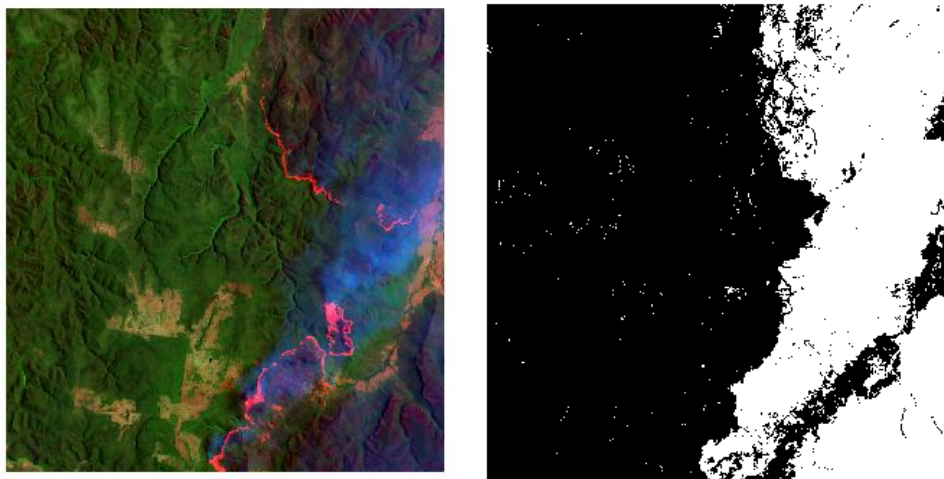


Рис. 3.12. Приклад вхідного зображення та його маска

### 3.3 Створення нейронних мереж для класифікації супутникових даних

Для створення програмного додатку була обрана мова програмування Python. Основні підключені модулі:

- tensorflow – бібліотека для машинного навчання, розроблена для побудови та тренування нейронних мереж;

- keras – бібліотека, що є надбудовою над TensorFlow; містить багато реалізацій широко застосовуваних блоків нейронних мереж, таких як оптимізатори, цільові функції тощо;
- segmentation\_models – бібліотека, яка містить в собі моделі мереж для класифікації та сегментації; використовується для зменшення часу навчання за рахунок налаштованих вагових коефіцієнтів;
- opencv – це бібліотека алгоритмів комп'ютерного зору, яка призначена для аналізу, класифікації та обробки зображень.

Наступним етапом після підключення необхідних бібліотек є зчитування вхідних зображень та створених бінарних масок, а також зміна розмірів вхідних зображень до  $128 \times 128$  пікселів. Обрізані вхідні зображення та маски записуються в масиви `images` та `masks` відповідно.

```
images = []
for path in glob.glob(os.path.join("C:/diploma_input/imagess/*.tif")):
    img = cv2.imread(path, 1)
    img = cv2.resize(img, (128, 128))
    images.append(img)
images = np.array(images)
masks = []
for path1 in glob.glob("C:/diploma_input/masks/*.tif"):
    mask = cv2.imread(path1, 0)
    mask = cv2.resize(mask, (128, 128), interpolation = cv2.INTER_NEAREST)
    masks.append(mask)
masks = np.array(masks)
```

Вхідні дані поділяються у відношенні 50% тренувальних на 50% тестувальних за допомогою функції `train_test_split`:

```
xTrain, xTest, yTrain, yTest = train_test_split(images, train_masks_input, test_size = 0.5, random_state = 0)
```

Також необхідно задати параметри, що будуть використовуватися для обох нейронних мереж, а саме: кількість класів, кількість епох, коефіцієнт швидкості навчання та оптимізатор.

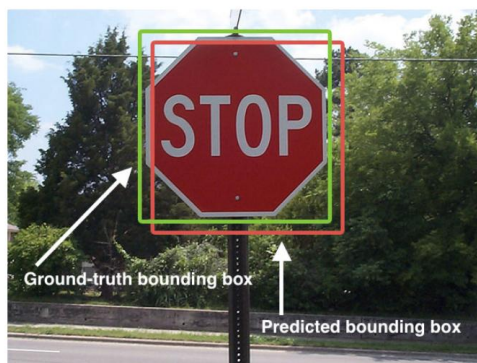
Коефіцієнт швидкості навчання – це гіперпараметр, що визначає порядок того, як ми коригуватимемо наші ваги з урахуванням функції втрат у градієнтному спуску. Чим нижче величина, тим повільніше ми рухаємось

похилою. При використанні низького коефіцієнта швидкості навчання ми можемо отримати позитивний ефект у тому сенсі, щоб не пропустити жодного локального мінімуму.

Оптимізатори – важливий компонент архітектури нейронних мереж. Вони визначають оптимальний набір параметрів моделі, таких як ваги та зсув, щоб при розв’язанні конкретної задачі модель видавала найкращі результати. Найпоширенішою технікою оптимізації, яка використовується більшістю нейронних мереж, є алгоритм градієнтного спуску. Більшість популярних бібліотек глибокого навчання, наприклад PyTorch та Keras, мають багато вбудованих оптимізаторів, що базуються на використанні алгоритму градієнтного спуску, наприклад SGD, Adadelata, Adagrad, RMSProp, Adam тощо. В даному випадку використовується оптимізатор Adam.

Отже, кількість описаних класів - 2, кількість епох - 50 , коефіцієнт швидкості навчання = 0,0001, оптимізатор - Adam.

Для оцінки ефективності роботи нейронної мережі застосовується IoU score (Intersection over Union) та F1-score. IoU також відомий як Jaccard index, - число від 0 до 1, яке демонструє, наскільки у двох об’єктів (еталонного (ground-truth bounding box) і поточного (predicted bounding box)) співпадає внутрішній об’єм (рис. 3.13). Значення IoU обчислюється за допомогою значень recall та precision, де recall - повнота, precision - точність. Precision можна інтерпретувати як частку об’єктів, названих класифікатором позитивними і які при цьому дійсно є позитивними, а recall показує, яку частку об’єктів позитивного класу з усіх його об’єктів знайшов алгоритм.



$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Рисунок 3.13. Intersection over Union score

Якщо значення IoU score  $> 0,5$ , то ефективність роботи нейронної мережі вважається високою (рис. 3.14).



Рисунок 3.14. Приклади IoU scores

Також для створення обох нейронних мереж додатково застосовується основна (базова) нейронна мережа, яка служить для отримання ознак з вхідного зображення. Це дозволяє використовувати архітектуру класифікації з попередньо навченими вагами. В якості такої базової нейронної мережі була обрана ResNet.

Після задання необхідних параметрів наступним кроком є безпосередньо створення та навчання кожної мережі.

```
#МОДЕЛЬ UNET
model1 = sm.Unet(BACKBONE1, encoder_weights='imagenet', classes=n_classes, activation='softmax')
model1.compile(optim, total_loss, metrics=metrics)
print(model1.summary())
start1 = datetime.now()
history1=model1.fit(X_train1,
                    y_train_cat,
                    batch_size=5,
                    epochs=50,
                    verbose=1,
                    validation_data=(X_test1, y_test_cat))

#обрахунок часу роботи U-Net
stop1 = datetime.now()
execution_time_unet = stop1-start1
print("U-Net - час навчання - ", execution_time_unet)
model2 = sm.Unet(BACKBONE1, encoder_weights='imagenet', classes=n_classes, activation='softmax')
```

```

model2.compile(optim, total_loss, metrics=metrics)
print(model2.summary())
start2 = datetime.now()
history2=model2.fit(X_train1,
                    y_train_cat,
                    batch_size=7,
                    epochs=50,
                    verbose=1,
                    validation_data=(X_test1, y_test_cat))
stop2 = datetime.now()
execution_time_linknet = stop2-start2
print("Linknet - час навчання - ", execution_time_linknet)

```

Після навчання обох мереж дані про навчання необхідно зберегти для кожної епохи для подальшого використання під час побудови графіків та тестування точності розпізнавання пожежі на знімку (рис. 3.14).

	A	B	C	D	E	F	G	H	I	J	K	L
1	,	loss,iou_score,f1-score,	val_loss,	val_iou_score,	val_f1-score							
2												
3	0,	0.7863698601722717,	0.3516576588153839,	0.5188992619514465,	0.7817566990852356,	0.32258474826812744,	0.4845009446144104					
4												
5	1,	0.7475121021270752,	0.4607662856578827,	0.629605770111084,	0.7811737060546875,	0.3259534239768982,	0.48985669016838074					
6												
7	2,	0.7595100998878479,	0.45681920647621155,	0.6127014756202698,	0.773339569568634,	0.4010440707206726,	0.5611087083816528					
8												
9	3,	0.6991337537765503,	0.6294914484024048,	0.7688266038894653,	0.7656655311584473,	0.4256784915924072,	0.5729619860649109					

Рисунок 3.15. Збережені дані про навчання

Для візуалізації процесу навчання виводяться графіки, що відображають точність (IoU score) та втрати (loss) на навчальних та тестових даних для мереж U-Net (рис. 3.16-3.17) та Linknet (рис. 3.18-3.19).

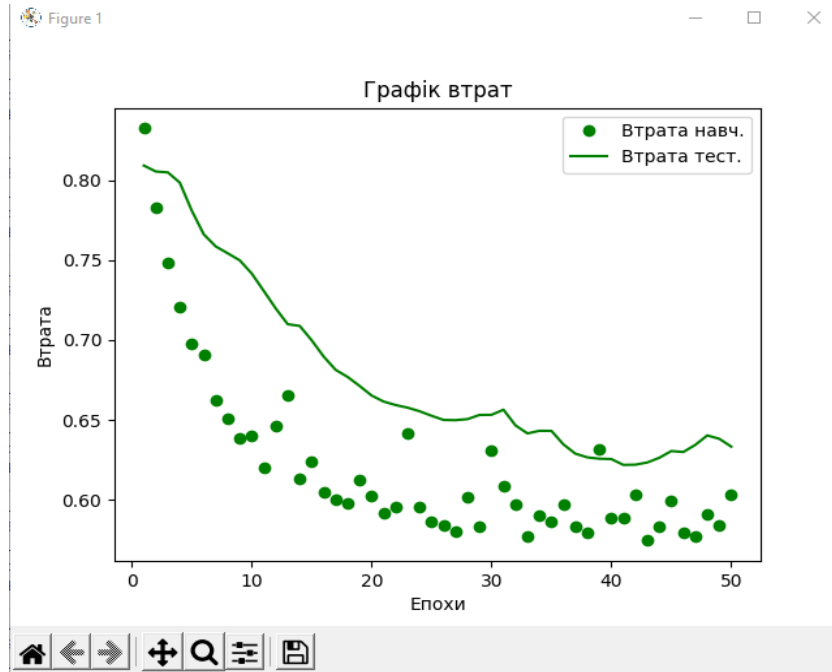


Рисунок 3.16. Графік втрат для мережі U-Net

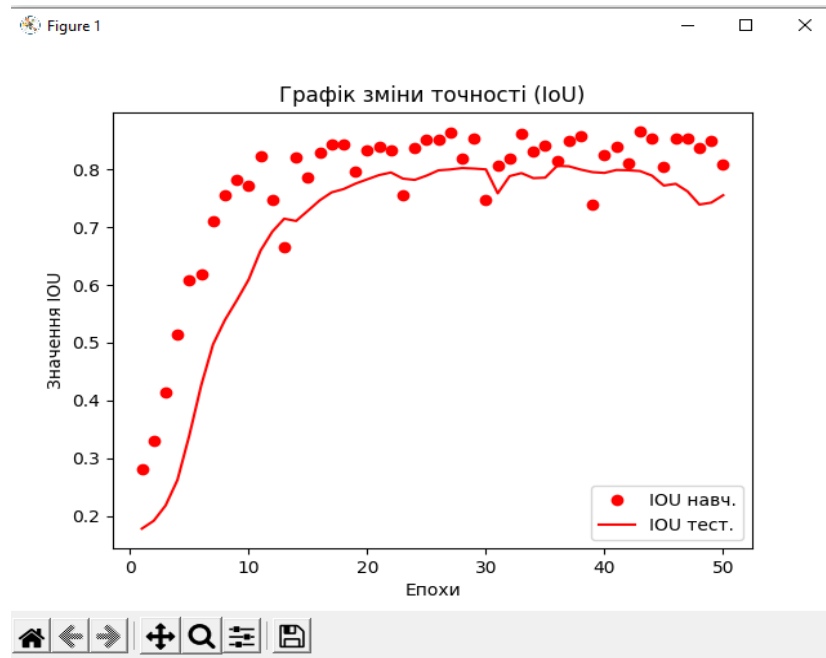


Рисунок 3.17. Графік зміни точності для мережі U-Net

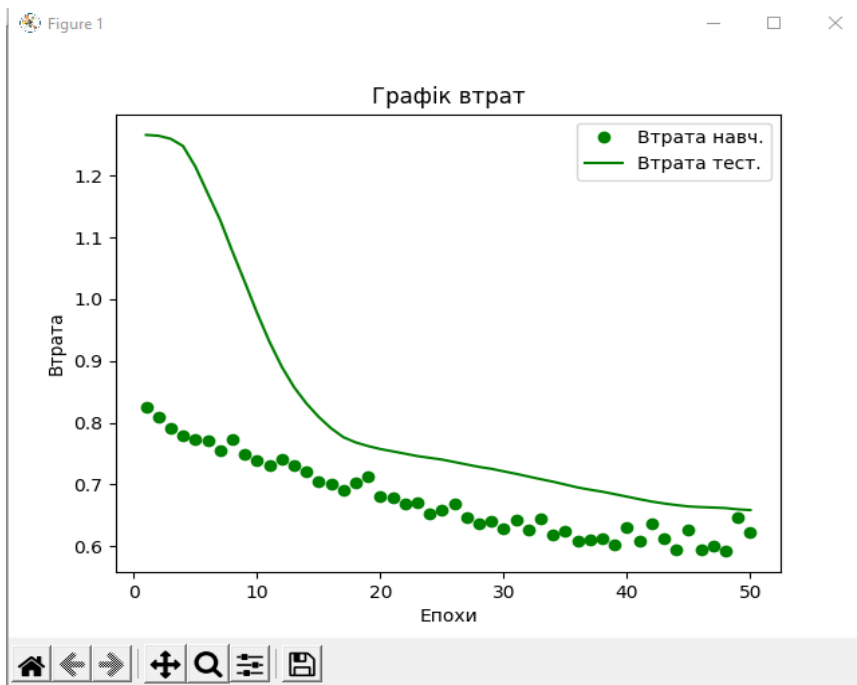


Рисунок 3.18. Графік втрат для мережі Linknet

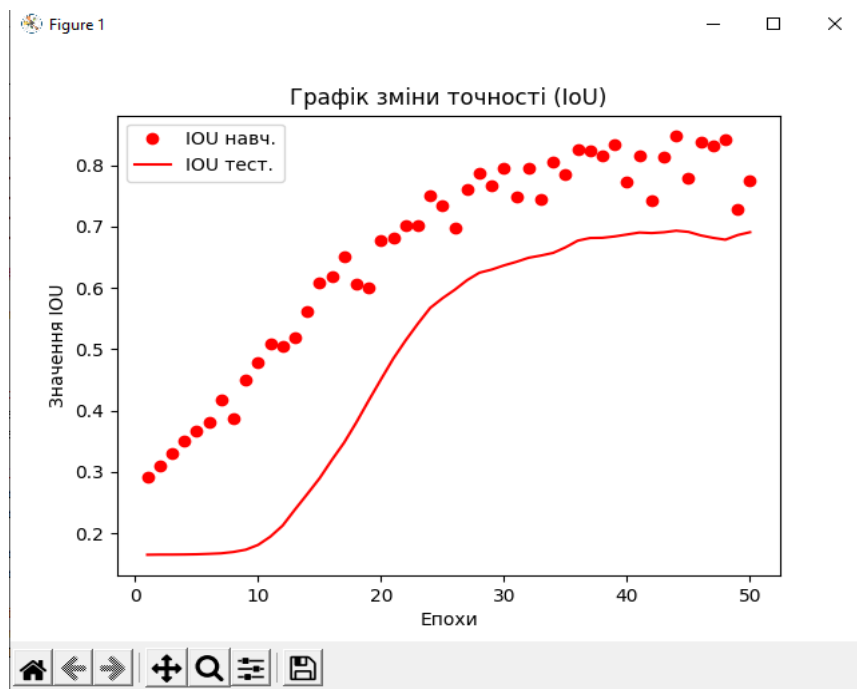


Рисунок 3.19. Графік зміни точності (IoU)

З отриманих графіків бачимо, що значення втрат зменшується з кожною епохою, в той час як значення точності, навпаки, зростає. Дані зміни означають, що мережа навчається успішно. Оскільки значення IoU перевищує 0,5, можна зробити висновок, що мережа класифікує з високою точністю.

### 3.4 Порівняльний аналіз отриманих результатів

Для здійснення порівняльного аналізу роботи нейронних мереж необхідно по чергово перевірити роботу кожної мережі з використанням функцій активації Sigmoid та Softmax.

На рис. 3.20 зображений результат навчання нейронної мережі U-Net з використанням функції активації Sigmoid на останній епосі.

```
loss: 0.3773 - iou score: 0.6969 - f1-score: 0.7851
```

Рисунок 3.20. Результат навчання U-Net з Sigmoid

Час витрачений на навчання мережі зображений на рис. 3.21.

```
U-Net - час навчання - 0:13:32.381850
```

Рисунок 3.21. Час навчання U-Net з Sigmoid

На рис. 3.22 зображений результат навчання нейронної мережі Linknet з використанням функції активації Sigmoid.

```
loss: 0.5942 - iou_score: 0.8217 - f1-score: 0.8986
```

Рисунок 3.22. Результат навчання Linknet з Sigmoid

Час витрачений на навчання мережі зображений на рис. 3.23.

```
Linknet - час навчання - 0:12:37.880894
```

Рисунок 3.23. Час навчання Linknet з Sigmoid

Порівняльні результати роботи програми з використанням функції активації Sigmoid представлені в табл. 3.2.

Таблиця 3.2 Використання функції активації Sigmoid

Нейронна мережа	Час виконання	Значення loss	IoU-score	F1-score
U-Net	13:32	0,3773	0,6969	0,7851
LinkNet	12:38	0,5942	0,8217	0,8986

Середнє значення IoU для мереж U-Net та Linknet, розраховане для всіх епох, зображене на рис. 3.24.

Середнє значення IoU використовуючи U-Net: 0.7676878  
 Середнє значення IoU використовуючи Linknet: 0.8274969

Рисунок 3.24. Середнє значення IoU з функцією активації Sigmoid

Вхідне зображення, що використовувалося для тестування представлено на рис. 3.25. Результат розпізнавання пожежі на зображенні з використанням функції активації Sigmoid представлений на рис. 3.26.

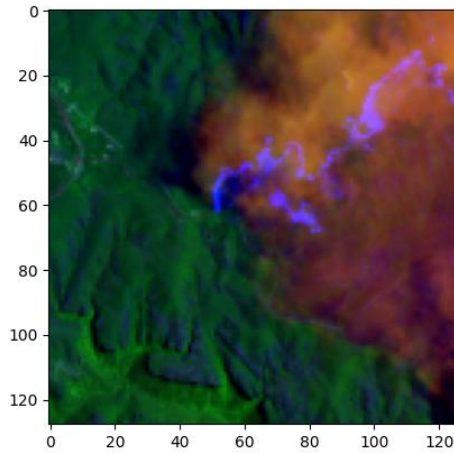


Рисунок 3.25. Тестове зображення для класифікації

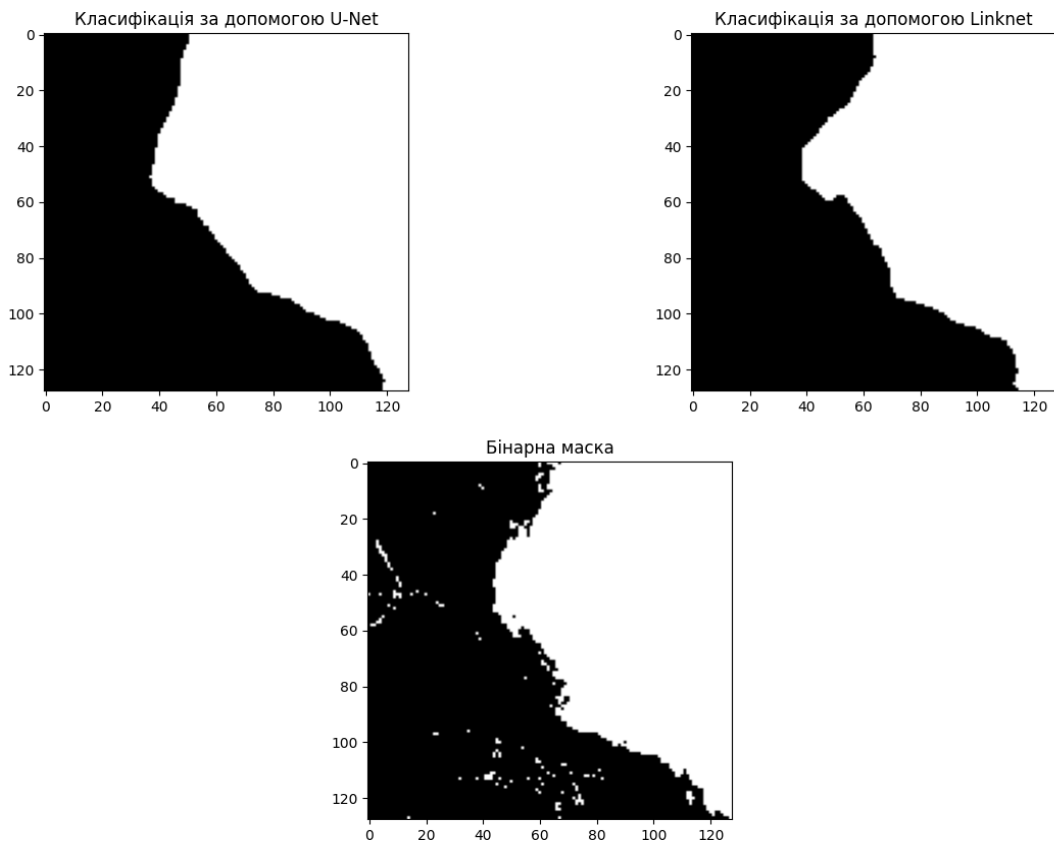


Рисунок 3.26. Результат проведеної класифікації вхідного зображення

На рис. 3.27 зображений результат навчання нейронної мережі U-Net з використанням функції активації Softmax на останній епосі.

`loss: 0.5725 - iou_score: 0.8527 - f1-score: 0.9196`

Рисунок 3.27. Результат навчання U-Net з Softmax

Час витрачений на навчання мережі зображений на рис. 3.28.

`U-Net - час навчання - 0:13:38.032794`

Рисунок 3.28. Час навчання U-Net з Softmax

На рис. 3.29 зображений результат навчання нейронної мережі Linknet з використанням функції активації Softmax на останній епосі.

`loss: 0.5742 - iou_score: 0.8567 - f1-score: 0.9216`

Рисунок 3.29. Результат навчання Linknet з Softmax

Час витрачений на навчання мережі зображений на рис. 3.30.

`Linknet - час навчання - 0:08:02.382044`

Рисунок 3.30. Час навчання Linknet з Softmax

Порівняльні результати роботи програми з використанням функції активації Softmax представлені в табл. 3.3.

Таблиця 3.3 Використання функції активації Softmax

Нейронна мережа	Час виконання	Значення loss	IoU-score	F1-score
U-Net	13:38	0,5725	0,8527	0,9196
LinkNet	08:02	0,5742	0,8567	0,9216

Середнє значення IoU для мереж U-Net та Linknet, розраховане для всіх епох, зображене на рис. 3.31.

`Середнє значення IoU використовуючи U-Net: 0.79312325`  
`Середнє значення IoU використовуючи Linknet: 0.8251804`

Рисунок 3.31. Середнє значення IoU з використанням Softmax

Результат розпізнавання пожежі на зображенні з використанням функції активації Softmax представлений на рис. 3.32.

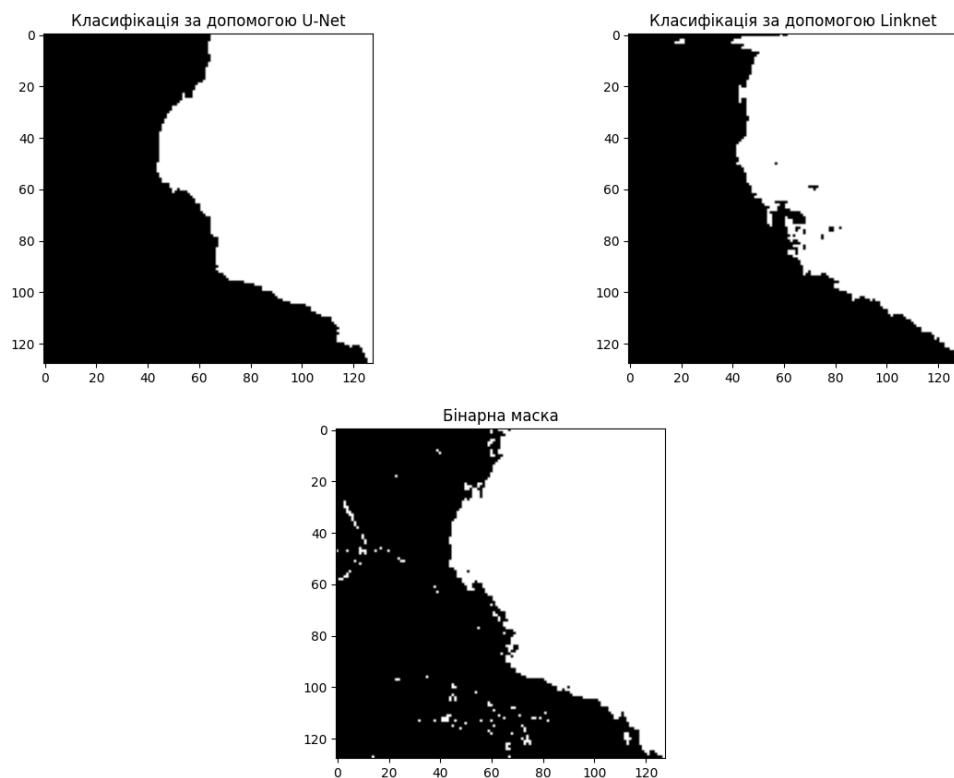


Рисунок 3.32. Результат проведеної класифікації вхідного зображення

Отримані результати внаслідок навчання обох мереж з різними функціями активації вносяться в таблицю для проведення порівняльного аналізу (табл. 3.4).

Таблиця 3.4 Порівняльний аналіз для мереж U-Net та Linknet з функціями активації Sigmoid і Softmax

	U-Net		Linknet	
	Sigmoid	Softmax	Sigmoid	Softmax
Час виконання (хв)	13:32	13:38	12:38	08:02
Значення loss	0,3773	0,5725	0,5942	0,5742
IoU-score	0,6969	0,8527	0,8217	0,8567
F1-score	0,7851	0,9196	0,8986	0,9216

В табл. 3.5 відображені середні значення точності класифікації (IoU) для мереж U-Net та Linknet з використанням функцій активації Sigmoid та Softmax.

Таблиця 3.5 Середні значення точності класифікації (IoU)

	Sigmoid	Softmax
U-Net	0,7676	0,7931
Linknet	0,8275	0,8251

Мережа Linknet продемонструвала вищу точність класифікації ( $\approx 82\%$ ) з обома активаційними функціями (табл. 3.5) за менший час, ніж мережа U-Net ( $\approx 78\%$ ). Кращої точності класифікації вдалося досягти, використовуючи функцію активації Softmax.

## ВИСНОВКИ

В результаті виконання випускної кваліфікаційної роботи проаналізовано існуючі рішення класифікації даних дистанційного зондування Землі. Створено дві згорткові нейронні мережі (U-Net та LinkNet) для бінарної класифікації пожежі на основі космічних знімків з супутника Landsat-8 з використанням функцій активації Sigmoid та Softmax. Проведено порівняльний аналіз отриманих результатів.

Мережа Linknet продемонструвала вищу точність класифікації ( $\approx 82\%$ ) з обома активаційними функціями Sigmoid та Softmax за менший час, ніж мережа U-Net ( $\approx 78\%$ ). Кращої точності класифікації вдалося досягти, використовуючи функцію активації Softmax.

Результати даного дослідження можна в подальшому використовувати і вдосконалювати. Розроблені мережі можуть використовуватися для часткової автоматизації ручної праці під час проведення контролю пожежної безпеки.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Сафонова А.Н. Методы машинного обучения при обработке изображений сверхвысокого пространственного разрешения на примере задач классификации растительности [Электронный ресурс] : диссертация ... кандидата технических наук / А.Н. Сафонова. - Красноярск: СФУ, 2019. [Электронный ресурс]. URL: <http://elib.sfukras.ru/handle/2311/127137>
2. Prakash, Nikhil & Manconi, Andrea & Loew, Simon. (2020). Mapping Landslides on EO Data: Performance of Deep Learning Models vs. Traditional Machine Learning Models. *Remote Sensing*. 12. 346. 10.3390/rs12030346.[Электронный ресурс]. URL:[https://www.researchgate.net/figure/U-net-architecture-with-ResNet34-blocks-in-the-down-sampling-path\\_fig2\\_338723610](https://www.researchgate.net/figure/U-net-architecture-with-ResNet34-blocks-in-the-down-sampling-path_fig2_338723610)
3. LeCu Y., Kavukcuoglu K. and Farabet C. (2010), Convolutional Networks and Applications in Vision, Proceedings of 2010 IEEE International Symposium on Circuits and Systems.
4. Романов А.А., Рубанов К.А. Эффективность непараметрических классификаторов в условиях лимитированной обучающей выборки //Journal of Siberian Federal University. Engineering & Technologies 5 (2012, 5) 495-506 [Электронный ресурс]. URL: <http://elib.sfu-kras.ru/handle/2311/3194>
5. Анисенко О.В. Розвиток дистанційного зондування Землі в Україні, 2017. [Электронный ресурс] – URL: [http://www.agrosvit.info/pdf/7\\_2017/10.pdf](http://www.agrosvit.info/pdf/7_2017/10.pdf)
6. Свідзінська, Д.В. Методи геоєкологічних досліджень: геоінформаційний практикум на основі відкритої ГІС SAGA: навчальний посібник/ Д.В. Свідзінська. – К.: Логос, 2014. – 402 с. ISBN 978-966-171-830-1

7. Стаття «NDVI – теория и практика» [Електронний ресурс]. URL: <https://gis-lab.info/qa/ndvi.html>
8. Стаття «Which activation function suits better to your Deep Learning scenario?» [Електронний ресурс] . URL: <https://datascience.aero/aviation-function-deep-learning/>
9. Стаття «U-Nets with ResNet Encoders and cross connections». [Електронний ресурс]. URL: <https://towardsdatascience.com/u-nets-with-resnet-encoders-and-cross-connections-d8ba94125a2c>
10. Стаття «Интерпретация комбинаций каналов данных Landsat». [Електронний ресурс]. URL: <https://gis-lab.info/qa/landsat-bandcomb.html>
- 11.Стаття «Which activation function suits better to your Deep Learning scenario?» [Електронний ресурс] – URL: <https://datascience.aero/aviation-function-deep-learning/>
12. Стаття «Сигмоидная функция активации: активация в нейронных сетях многослойный перцептрон» [Електронний ресурс] - URL: <https://russianblogs.com/article/9445941336/>
- 13.Архів Геологічної служби США. URL: <https://earthexplorer.usgs.gov/>

## ДОДАТКИ

## Додаток А

Програмний код для класи  
фікації пожежі за допомогою мереж  
U-Net та Linknet

```

#підключення бібліотек
import keras
from keras.utils import to_categorical
from keras.models import load_model
from keras.metrics import MeanIoU
import pandas as pd
import tensorflow as tf
import segmentation_models as sm
import cv2
import os
from datetime import datetime
import numpy as np
from matplotlib import pyplot as plt
from sklearn.preprocessing import LabelEncoder
from keras.utils import to_categorical
import random
from keras.metrics import MeanIoU
import glob
from sklearn.model_selection import train_test_split
#зчитування вхідних зображень та запис у масив
images = []
for directory_path in glob.glob("C:/diploma_input/images/"):
    for img_path in glob.glob(os.path.join(directory_path, "*.tif")):
        img = cv2.imread(img_path, 1)
        img = cv2.resize(img, (128, 128))
        images.append(img)
images = np.array(images)
#зчитування створених масок та запис у масив
masks = []
for directory_path in glob.glob("C:/diploma_input/masks/"):
    for mask_path in glob.glob(os.path.join(directory_path, "*.tif")):
        mask = cv2.imread(mask_path, 0)
        mask = cv2.resize(mask, (128, 128), interpolation = cv2.INTER_NEAREST)
        masks.append(mask)
masks = np.array(masks)

labelencoder = LabelEncoder()
n, h, w = masks.shape
train_masks_reshaped = masks.reshape(-1,1)
train_masks_reshaped_encoded = labelencoder.fit_transform(train_masks_reshaped)
train_masks_encoded_original_shape = train_masks_reshaped_encoded.reshape(n, h, w)

np.unique(train_masks_encoded_original_shape)
train_masks_input = np.expand_dims(train_masks_encoded_original_shape, axis=3)

n_classes=2
xTrain, xTest, yTrain, yTest = train_test_split(images, train_masks_input, test_size = 0.5, random_state = 0)

print("значення класів, що використовуються в датасеті - ", np.unique(yTrain))

train_masks_cat = to_categorical(yTrain, num_classes=n_classes)

```

```

y_train_cat = train_masks_cat.reshape((yTrain.shape[0], yTrain.shape[1], yTrain.shape[2], n_classes))

test_masks_cat = to_categorical(yTest, num_classes=n_classes)
y_test_cat = test_masks_cat.reshape((yTest.shape[0], yTest.shape[1], yTest.shape[2], n_classes))

#параметри, що використовуються в обох мережах
LR = 0.0001
optim = keras.optimizers.Adam(LR)
n_classes=2

dice_loss = sm.losses.DiceLoss(class_weights=np.array([0.5, 0.5]))
focal_loss = sm.losses.CategoricalFocalLoss()
total_loss = dice_loss + (1 * focal_loss)

metrics = [sm.metrics.IOUScore(threshold=0.5), sm.metrics.FScore(threshold=0.5)]

BACKBONE1 = 'resnet34'
preprocess_input1 = sm.get_preprocessing(BACKBONE1)

X_train1 = preprocess_input1(xTrain)
X_test1 = preprocess_input1(xTest)

#МОДЕЛЬ UNET
model1 = sm.Unet(BACKBONE1, encoder_weights='imagenet', classes=n_classes, activation='softmax')
model1.compile(optim, total_loss, metrics=metrics)

print(model1.summary())
start1 = datetime.now()
history1=model1.fit(X_train1,
                    y_train_cat,
                    batch_size=5,
                    epochs=50,
                    verbose=1,
                    validation_data=(X_test1, y_test_cat))
#обрахунок часу роботи U-Net
stop1 = datetime.now()
execution_time_unet = stop1-start1
print("U-Net - час навчання - ", execution_time_unet)
model1.save('unet.hdf5')
hist1_df = pd.DataFrame(history1.history)
hist1_csv_file = 'unet.csv'
with open(hist1_csv_file, mode='w') as f:
    hist1_df.to_csv(f)

#побудова графіку втрат для U-Net
loss = history1.history['loss']
val_loss = history1.history['val_loss']
epochs = range(1, len(loss) + 1)
plt.plot(epochs, loss, 'go', label='Втрати навч.')
plt.plot(epochs, val_loss, 'g', label='Втрати тест.')
plt.title('Графік втрат')
plt.xlabel('Епохи')
plt.ylabel('Втрати')
plt.legend()
plt.show()
#побудова графіку зміни точності
acc = history1.history['iou_score']
val_acc = history1.history['val_iou_score']
plt.plot(epochs, acc, 'ro',
label='IOU навч.')
plt.plot(epochs, val_acc, 'r', label='IOU тест.')

```

```

plt.title('Графік зміни точності')
plt.xlabel('Епохи')
plt.ylabel('IOU')
plt.legend()
plt.show()

#МОДЕЛЬ LINKNET

model2 = sm.Unet(BACKBONE1, encoder_weights='imagenet', classes=n_classes, activation='softmax')
model2.compile(optim, total_loss, metrics=metrics)
print(model2.summary())
start2 = datetime.now()

history2=model2.fit(X_train1,
                    y_train_cat,
                    batch_size=7,
                    epochs=50,
                    verbose=1,
                    validation_data=(X_test1, y_test_cat))

stop2 = datetime.now()
execution_time_linknet = stop2-start2
print("Linknet - час навчання - ", execution_time_linknet)
model2.save('linknet.hdf5')

hist2_df = pd.DataFrame(history2.history)
hist2_csv_file = 'linknet.csv'
with open(hist2_csv_file, mode='w') as f:
    hist2_df.to_csv(f)

#побудова графіку втрат для LinkNet
loss = history2.history['loss']
val_loss = history2.history['val_loss']
epochs = range(1, len(loss) + 1)
plt.plot(epochs, loss, 'go', label='Втрати навч.')
plt.plot(epochs, val_loss, 'g', label='Втрати тест.')
plt.title('Графік втрат')
plt.xlabel('Епохи')
plt.ylabel('Втрати')
plt.legend()
plt.show()
#побудова графіку зміни точності
acc = history2.history['iou_score']
val_acc = history2.history['val_iou_score']
plt.plot(epochs, acc, 'ro', label='IOU навч.')
plt.plot(epochs, val_acc, 'r', label='IOU тест.')
plt.title('Графік зміни точності')
plt.xlabel('Епохи')
plt.ylabel('IOU')
plt.legend()
plt.show()

model_unet = load_model('unet.hdf5', compile=False)
model_linknet = load_model('linknet.hdf5', compile=False)

#IoU
y_pred_unet=model_unet.predict(X_test1)
y_pred_unet_argmax=np.argmax(y_pred_unet, axis=3)

y_pred_linknet=model_linknet.predict(X_test1)

```

```

y_pred_linknet_argmax=np.argmax(y_pred_linknet, axis=3)

IOU_unet = MeanIoU(num_classes=n_classes)
IOU_unet.update_state(yTest[:, :, 0], y_pred_unet_argmax)

IOU_linknet = MeanIoU(num_classes=n_classes)
IOU_linknet.update_state(yTest[:, :, 0], y_pred_linknet_argmax)

print("Середнє значення IoU використовуючи U-Net:", IOU_unet.result().numpy())
print("Середнє значення IoU використовуючи Linknet:", IOU_linknet.result().numpy())

#тестування
test_img_number = random.randint(0, len(X_test1))
test_img = X_test1[test_img_number]
ground_truth=yTest[test_img_number]
test_img_input=np.expand_dims(test_img, 0)
test_img_input1 = preprocess_input1(test_img_input)
test_pred_unet = model_unet.predict(test_img_input1)
test_prediction_unet = np.argmax(test_pred_unet, axis=3)[0, :, :]
test_pred_linknet = model_linknet.predict(test_img_input1)
test_prediction_linknet = np.argmax(test_pred_linknet, axis=3)[0, :, :]
plt.subplot(111)
plt.imshow(test_img[:, :, :])
plt.show()
plt.figure(figsize=(15, 15))
plt.subplot(212)
plt.title('Бінарна маска')
plt.imshow(ground_truth[:, :, :], cmap='gray')
plt.subplot(221)
plt.title('Класифікація за допомогою U-Net')
plt.imshow(test_prediction_unet, cmap='gray')
plt.subplot(222)
plt.title('Класифікація за допомогою Linknet')
plt.imshow(test_prediction_linknet, cmap='gray')
plt.show()

```