

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Кваліфікаційна наукова
праця на правах рукопису

Головіна Надія Владиславівна

УДК 004.93:504

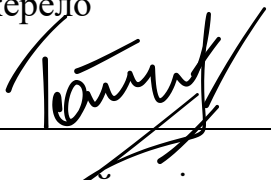
ДИСЕРТАЦІЯ

**МОДЕЛІ ТА МЕТОДИ ОБРОБКИ ПОТОКІВ ДАНИХ
ЗОБРАЖЕНЬ В СИСТЕМІ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ**

122 Комп'ютерні науки
12 Інформаційні технології

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело



Науковий керівник

Захарченко Раїса Миколаївна
кандидат технічних наук, доцент

Хмельницький 2025

АНОТАЦІЯ

Головіна Н.В. Моделі та методи обробки потоків даних зображень в системі підтримки прийняття рішень. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 Комп'ютерні науки, Херсонський національний технічний університет, Міністерство освіти і науки України, Хмельницький, 2024.

У роботі досліджено моделі та методи обробки супутникових зображень. Створено структурні схеми та математично обґрунтовані алгоритми. Продемонстровано результати роботи розробленої гібридної моделі обробки та аналізу супутникових зображень для виявлення осередку пожежі на початковому стані.

Об'єктом дослідження є процеси прийняття рішень та оперативного реагування на надзвичайні ситуації в умовах виникнення лісових пожеж. Предметом дослідження є методи, моделі та інструментальні засоби удосконалення процесу раннього виявлення та ідентифікації лісових пожеж на основі гібридної комп'ютерної моделі.

Розглянуто методи аналізу зображень: сегментація, виділення ознак, математичні морфологічні операції. Розглянуто методи глибокого навчання та гібридні моделі. Розглянуто такі моделі: AlexNet, VGGNet, Inception (GoogLeNet), MobileNet.

Розроблена архітектура нейронної мережі продемонструвала свою ефективність у виявленні вогневих патернів, що підтверджується досягнутою точністю 92%. Цей результат свідчить про успішну реалізацію алгоритму навчання, який базується на згорткових нейронних мережах.

Ключові слова: методи обробки зображень, гібридна модель обробки зображень, СППР, лісові пожежі, згорткова нейронна мережа, супутникові зображення.

Список публікацій:

1. Головіна Н.В., Ляшенко О.М. Дослідження методів спектрального аналізу зображень для детектування лісових пожеж з використанням технології обчислювального інтелекту. *Вісник ХНТУ*. 2021. Том 79 (4). С. 74-84.

URL: <https://ojs.kntu.net.ua/index.php/visnyk/article/view/203/249>

DOI: <https://doi.org/10.35546/kntu2078-4481.2021.4.8>

(особистий внесок – досліджено та запропоновано використання сукупності методів спектрального аналізу зображень для моніторингу та попередження лісових пожеж).

2. Головіна Н.В., Ляшенко О.М. Дослідження методів відновлення та реконструкції зображень. *Вчені записки національного університету імені В.І. Вернадського*. Серія: Технічні науки. 2022. Том 33 (72) № 2. С. 80-85.

URL: https://tech.vernadskyjournals.in.ua/journals/2022/2_2022/13.pdf

DOI: <https://doi.org/10.32838/2663-5941/2022.2/13>

(особистий внесок – досліджено та запропоновано використання методів відновлення та реконструкції зображень для моніторингу та попередження лісових пожеж).

3. Головіна Н.В. Розроблення моделі сегментації лісових пожеж на супутникових знімках із застосування нейромережових технологій. *Збірник наукових праць Національного університету кораблебудування імені адмірала Макарова*. 2024. № 2 (495). С. 95-102.

URL: <https://znp.nuos.mk.ua/archives/2024/2/14.pdf>

DOI: [https://doi.org/10.15589/znp2024.2\(495\).12](https://doi.org/10.15589/znp2024.2(495).12)

(особистий внесок – розроблено модель сегментації лісових пожеж на супутникових знімках із застосуванням нейромережових технологій).

4. Головіна Н.В. Розроблення системи підтримки прийняття рішень для моніторингу та попередження лісових пожеж в Україні. *Вісник Херсонського національного технічного університету. Інформаційні технології*. 2024. №2 (89). С. 150-156.

URL:

https://journals.kntu.kherson.ua/index.php/visnyk_kntu/article/view/639/613

DOI: <https://doi.org/10.35546/kntu2078-4481.2024.2.21>

(особистий внесок – розроблено систему підтримки прийняття ріше для моніторингу та попередження лісових пожеж в Україні).

5. Головіна Н.В. Розроблення нейромережевої моделі моніторингу надзвичайних ситуацій природного характеру. *Системи та технології. Комп'ютерні науки*. 2024. №67 (1). С. 36-42.

URL: <https://st.umsf.in.ua/index.php/journal/article/view/143/150>

DOI: <https://doi.org/10.32782/2521-6643-2024-1-67.6>

(особистий внесок – розроблено нейромережеву модель моніторингу надзвичайних ситуацій природного характеру).

6. Головіна Н.В. Розробка нейромережевої моделі для моніторингу лісових пожеж природного характеру. *Вчені Записки Таврійського національного університету імені В.І. Вернадського*. Серія: Технічні науки. 2024. №3 (74). Т.35. С. 53-57.

URL:

https://www.tech.vernadskeyjournals.in.ua/journals/2024/3_2024/part_1/12.pdf

DOI: <https://doi.org/10.32782/2663-5941/2024.3.1/10>

(особистий внесок – розроблено та програмно реалізовано нейромережеву модель для моніторингу лісових пожеж природного характеру).

Апробації результатів дисертації:

1. Головіна Н.В., Ляшенко О.М. Дослідження методів комп'ютерного зору з відновлення та реконструкції зображень. Матеріали IV Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених за тематикою «Сучасні комп'ютерні системи та мережі в управлінні»: збірка наукових праць / Під редакцією Г.О. Райко. – Херсон: Видавництво ФОП Вишемирський В. С., 2021. С. 206-209.

2. Olena Liashenko, Dmytro Kyryichuk , Nadiia Holovina, Viktor Prachyk, Svitlana Bunkus. Development of information technology for monitoring of forest ecosystems under conditions of wildfires. SGEM International Scientific Conferences on Earth & Planetary Sciences Extended Scientific Sessions „GREEN SCIENCES FOR GREEN LIFE“ (Vienna, 6 9 December, 2022). Volume 22. Issue 3.2. (Scopus, ISI Web of Science Clarivate).
3. Головіна Н.В. Методи обробки потоків даних зображень в системі підтримки прийняття рішень для моніторингу надзвичайних ситуацій. Синергія науки і бізнесу у повоєнному відновленні Херсонщини. (ХНТУ, 26-28 квітня 2023) у 2-х т. ; Т. 1 / за ред. О. В. Чепелюк. – Одеса : Олді+, 2023. С. 254-258.
4. Головіна Н.В. Особливості розробки та використання системи підтримки прийняття рішень для моніторингу та попередження надзвичайних ситуацій в Україні. Синергія науки і бізнесу у повоєнному відновленні регіонів України : матеріали II Міжнародної науково-практичної конференції (ХНТУ, 24–26 квітня 2024 року) у 3-х т. ; Т. 2 / за ред. О. В. Чепелюк. – Одеса : Олді+, 2024. С.38-41.

ABSTRACT

Holovina N.V. Models and methods of processing image data streams in a decision support system. – Qualifying scientific work as a manuscript.

Dissertation for the degree of Doctor of Philosophy in the specialty 122 Computer Science, Kherson National Technical University, Ministry of Education and Science of Ukraine, Khmelnytskyi, 2025.

The paper investigates models and methods for processing satellite images. Structural diagrams and mathematically based algorithms are created. The results of the developed hybrid model for processing and analyzing satellite images for detecting a fire source in the initial state are demonstrated.

The object of the study is the processes of decision-making and operational response to emergencies in conditions of forest fires. The subject of the study is methods, models and tools for improving the process of early detection and identification of forest fires based on a hybrid computer model.

The methods of image analysis are considered: segmentation, feature extraction, mathematical morphological operations. Deep learning methods and hybrid models are considered. The following models are considered: AlexNet, VGGNet, Inception (GoogLeNet), MobileNet.

The developed neural network architecture has demonstrated its effectiveness in detecting fire patterns, which is confirmed by the achieved accuracy of 92%. This result indicates the successful implementation of the learning algorithm based on convolutional neural networks.

Keywords: image processing methods, hybrid image processing model, DSS, forest fires, convolutional neural networks, satellite images.

List of publications:

1. Holovina N.V., Liashenko O.M. Research on methods of spectral image analysis for detecting forest fires using computational intelligence technologies. Bulletin of KNTU. 2021. Vol. 79 (4). P. 74-84.

URL: <https://ojs.kntu.net.ua/index.php/visnyk/article/view/203/249>

DOI: <https://doi.org/10.35546/kntu2078-4481.2021.4.8>

(personal contribution the use of a set of spectral image analysis methods for monitoring and preventing forest fires was investigated and proposed).

2. Holovina N.V., Liashenko O.M. Research on methods of image restoration and reconstruction. Scientific notes of V.I. Vernadsky Taurida National University. Series: Technical Sciences. 2022. Vol. 33 (72) No. 2. P. 80-87

URL: https://tech.vernadskyjournals.in.ua/journals/2022/2_2022/13.pdf

DOI: <https://doi.org/10.32838/2663-5941/2022.2/13>

(personal contribution – the use of image restoration and reconstruction methods for monitoring and preventing forest fires was investigated and proposed).

3. Holovina N.V. Development of a forest fire segmentation model on satellite images using neural network technologies. Collection of Scientific Publications NUS. Publisher: Admiral Makarov National University of Shipbuilding. 2024. No. 2 (495). P. 95-102.

URL: <https://znp.nuos.mk.ua/archives/2024/2/14.pdf>

DOI: [https://doi.org/10.15589/znp2024.2\(495\).12](https://doi.org/10.15589/znp2024.2(495).12)

(personal contribution – developed a model for segmenting forest fires on satellite images using neural network technologies).

4. Holovina N.V. Development of a decision support system for monitoring and preventing forest fires in Ukraine. Bulletin of the Kherson National Technical University. Information Technologies. 2024. No. 2 (89). P. 150-156.

URL:

https://journals.kntu.kherson.ua/index.php/visnyk_kntu/article/view/639/613

DOI: <https://doi.org/10.35546/kntu2078-4481.2024.2.21>

(personal contribution – a decision support system for monitoring and preventing forest fires in Ukraine was developed).

5. Holovina N.V. Development of a neural network model for monitoring natural emergencies. Systems and technologies. Computer sciences. 2024. No. 67 (1). P. 36-42.

URL: <https://st.umsf.in.ua/index.php/journal/article/view/143/150>

DOI: <https://doi.org/10.32782/2521-6643-2024-1-67.6>

(personal contribution – developed a neural network model for monitoring natural emergencies).

6. Holovina N.V. Development of a neural network model for monitoring natural forest fires. Scientific notes of V.I. Vernadsky Taurida National University. Series: Technical Sciences. 2024. No. 3 (74). Vol. 35. P. 53-61.

URL:

https://www.tech.vernadskyjournals.in.ua/journals/2024/3_2024/part_1/12.pdf

DOI: <https://doi.org/10.32782/2663-5941/2024.3.1/10>

(personal contribution – developed and software implemented a neural network model for monitoring natural forest fires).

Approbation of the results of the dissertation:

1. Holovina N.V., Lyashenko O.M. Research of computer vision methods for image restoration and reconstruction. Materials of the IV All-Ukrainian scientific and practical Internet conference of students, postgraduates and young scientists on the topic «Modern computer systems and networks in management»: collection of scientific papers / Edited by G.O. Rayko. – Kherson: Publishing house IE Vyshemirskyi V.S., 2021. P. 206-209.
2. Olena Liashenko, Dmytro Kyryichuk, Nadiia Holovina, Viktor Prachyk, Svitlana Bunkus. Development of information technology for monitoring of forest ecosystems under conditions of wildfires. SGEM International Scientific Conferences on Earth & Planetary Sciences Extended Scientific Sessions „GREEN SCIENCES FOR GREEN LIFE“ (Vienna, 6-9 December, 2022). Volume 22. Issue 3.2. (Scopus, ISI Web of Science Clarivate).
3. Holovina N.V. Methods of processing image data streams in a decision support system for monitoring emergency situations. Synergy of science and business in the post-war reconstruction of the Kherson region. (KNTU, April 26-28, 2023) in 2 vols. ; Vol. 1 / edited by O. V. Chepeliuk. – Odessa: Oldi+, 2023. P. 254-258.

4. Holovina N.V. Features of the development and use of a decision support system for monitoring and preventing emergency situations in Ukraine. Synergy of science and business in the post-war reconstruction of regions of Ukraine: materials of the II International Scientific and Practical Conference (KNTU, April 24–26, 2024) in 3 vols. ; Vol. 2 / edited by O. V. Chepelyuk. – Odessa: Oldi+, 2024. P.38-41.

ЗМІСТ

ВСТУП	13
РОЗДІЛ 1 ОГЛЯД ІСНУЮЧИХ МОДЕЛЕЙ ТА МЕТОДІВ ОБРОБКИ ПОТОКІВ ДАНИХ ЗОБРАЖЕНЬ В СППР. ДОСЛІДЖЕННЯ ПОЖЕЖ В УКРАЇНІ.....	22
1.1 Дослідження існуючих систем класифікації моделей та методів обробки зображень.....	22
1.1.1 Дослідження традиційних методів обробки зображень та їх застосування	24
1.1.2 Дослідження та вивчення сучасних моделей глибокого навчання.....	26
1.1.3 Дослідження гібридних методів обробки зображень.....	27
1.2 Огляд існуючих систем підтримки прийняття рішень на основі зображень	27
1.2.1 Вивчення існуючих систем моніторингу та виявлення пожеж.....	28
1.2.2 Порівняльна характеристика функціональності та особливостей існуючих систем	30
1.3 Огляд поточної ситуації з лісовими пожежами в Україні та необхідність розробки СППР на основі аналізу зображень	31
1.4 Висновки до першого розділу	34
РОЗДІЛ 2 МОДЕЛЮВАННЯ: МАТЕМАТИЧНІ МОДЕЛІ ТА АРХІТЕКТУРА СИСТЕМИ СППР.....	35
2.1 Дослідження теорії та практичних аспектів моделювання	37
2.1.1 Вивчення характеристик та особливостей згорткових нейронних мереж: архітектура та принципи функціонування	37

2.1.2	Проведення дослідження моделі сегментації на основі повністю з'єднаних згорткових мереж.....	40
2.2	Дослідження та побудова архітектури розроблюваної системи підтримки прийняття рішень (СППР).....	44
2.2.1	Розробка та опис компонентів СППР та їх взаємодії: основні принципи.....	46
2.3	Проведення дослідження, реалізація та принципи алгоритмів	52
2.4	Висновки до другого розділу	54
РОЗДІЛ 3	ДОСЛІДЖЕННЯ МЕТОДІВ ВІДНОВЛЕННЯ ТА РЕКОНСТРУКЦІЇ ЗОБРАЖЕНЬ.....	55
3.1	Підготовка бази та даних для дослідження: джерела та методи збору інформації	56
3.2	Дослідження методів відновлення та реконструкції зображень: класифікація та моделювання.....	69
3.3	Обґрунтування використовуваних програмних засобів для моделювання комп'ютерної системи.....	83
3.4	Висновки до третього розділу	87
РОЗДІЛ 4	МОДЕЛЮВАННЯ ТА РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ	88
4.1	Реалізація та опис методів обробки зображень на основі проведеного дослідження, технології та підходи.....	90
4.2	Детальний опис процесу розробки системи: основні етапи та характеристики.....	107
4.3	Отримання результатів, аналіз та порівняння: висновки в рамках проведеного дослідження.....	125
4.4	Висновки до четвертого розділу	135

ВИСНОВКИ.....	137
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	141
Додаток А Список опублікованих праць за темою дисертації.....	153
Додаток Б Лістинг коду	155
Додаток В Скріншоти застосунок	162
Додаток Г Акт про впровадження	164

ВСТУП

Аналіз розвитку природних катастрофічних явищ у всьому світі показує, що, незважаючи на науково-технічний прогрес, захищеність людей і техносфери від природних небезпек не збільшується. Кількість жертв від руйнівних природних явищ, особливо лісових пожеж, та від пожеж в результаті агресивних військових дій росії, в останні роки тільки збільшується. У даний час існує багато досліджень того, що лісові пожежі є глобальною проблемою і одним з найважливіших факторів сталого розвитку економіки.

Вагомий внесок у розвиток та вдосконалення моделей та методів обробки потоків даних зображень в системі підтримки прийняття рішень зробили вітчизняні та зарубіжні вчені, зокрема: Кветний Р.Н., Богач І.В., Бойко О.Р., Софіна О.Ю., Шушура О.М., Журавль І.М., Зацерковний В., Савков П., Пампуха І., Васецька К., Rajasekaran T., Sruthi J., Revathi S., Karim R., Zahedi M. та багато інших. При цьому не існує універсального рішення, яке було б направлене на підвищення точності прогнозування природних катастроф з метою їх зменшення та попередження. Використовуються різноманітні методи аналізу та моделювання, які допомагають у цих питаннях, але потребують вдосконалення.

Актуальність дослідження. Актуальність дослідження зумовлена зростаючою частотою та масштабами лісових пожеж у всьому світі (зокрема, на території України, ще і через агресію росії). Шкідливий вплив цих природних явищ на екологію, економіку та суспільство вимагає розробки ефективних систем моніторингу та прогнозування. Існуючі системи та рішення часто мають обмежені можливості, особливо в умовах динамічно змінюваної ситуації. Проведене дослідження спрямоване на розробку системи, яка за допомогою сучасних методів та моделей обробки зображень дасть можливість виявляти осередки пожеж на основі наданих зображень швидше та з більшою точністю у порівнянні з існуючими моделями.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертаційна робота є частиною тематичної програми досліджень кафедри програмних засобів і технологій Херсонського національного технічного університету: «Моделі та методи територіальної інформаційної системи підтримки управління евакуацією населення при надзвичайних ситуаціях». Автор дисертаційної роботи мав на меті дослідити, обрати та розробити моделі та методи обробки потоків даних зображень в системі підтримки прийняття рішень для запобігання катастроф природного характеру, а саме лісових пожеж.

Мета та завдання дослідження. *Метою дослідження є підвищення ефективності процесів управління обробки потоків даних зображень в системі підтримки прийняття рішень для запобігання катастроф природного характеру, а саме лісових пожеж. Необхідно розробити ефективну систему виявлення задимленості на супутникових зображеннях та знімках з метою раннього виявлення осередків пожеж з достатньою точністю для підвищення оперативності реагування на надзвичайні ситуації відповідними службами. Адже моделі обробки зображень мають обмеження і недоліки, серед яких: технічні, алгоритмічні, залежність від даних. Необхідно вирішити проблеми з якістю вхідних даних: шуми, перекриття об'єктів, погане освітлення (задимленість), що знижують точність моделі. Моделі обробки зображень не мають справжнього «розуміння» того, що вони бачать. Вони ґрунтуються на візерунках у даних, а не на логічному аналізі. Розроблювана система повинна мінімізувати недоліки і реагувати швидше у порівнянні з існуючими без втрати точності, що є основною проблематикою дослідження.*

Основними завданнями дослідження були виділені наступні:

1. Провести дослідження теоретичних основ та існуючих методів обробки супутникових зображень.
2. Розробити структурну схему та алгоритми для наочної демонстрації роботи алгоритмів, що будуть розглянуті.

3. Провести дослідження різних методів та моделей обробки потоків даних зображень.
4. Провести аналіз доцільності використання модифікованих методів та розроблених моделей обробки потоків даних зображень.
5. Розробити систему підтримки прийняття рішень на основі модифікованих методів та розроблених моделей обробки потоків даних зображень.
6. Реалізувати комп'ютерну модель на основі проведених досліджень.
7. Перевірити результати проведених досліджень та встановити їх точність у порівнянні з існуючими моделями.
8. Здійснити апробацію розробленої (далі – гібридної) моделі з використанням даних супутникових зображень.

Об'єктом дослідження є процеси прийняття рішень та оперативного реагування на надзвичайні ситуації в умовах виникнення лісових пожеж.

Предметом дослідження є методи, моделі та інструментальні засоби удосконалення процесу раннього виявлення та ідентифікації лісових пожеж на основі гібридної комп'ютерної моделі та еволюційних методів обробки потоків даних зображень.

Методи дослідження. Основними методами дослідження в цій роботі є дотримання загальнонаукових методологій і принципів системного підходу. Для розв'язання поставлених задач використовувались такі методи:

- методи математичного моделювання для дослідження та оцінки ефективності застосування алгоритмів, методів та інформаційних технологій;
- методи функціонального та об'єктно-орієнтованого аналізу для дослідження предметної області;
- методи інтелектуального аналізу даних для виконання задач класифікації супутникових зображень;

- методи об'єктно-орієнтованого моделювання, використані для створення комп'ютерної системи та елементів;
- методи прийняття рішень, аналізу та класифікації зображень, еволюційні методи і алгоритми для розроблення системи раннього попередження;
- методи, інструментальні засоби і сучасні технології прикладного програмування для практичної реалізації дослідження.

Таким чином, дисертаційна робота припускає використання як теоретичного, так і емпіричного методу дослідження.

Теоретичною основою роботи є наукові роботи провідних вітчизняних і зарубіжних учених в області підтримки прийняття рішень з використанням моделей та методів обробки потоків даних зображень для раннього реагування на факт пожежі. Емпіричною основою роботи стали методи спостереження, вимірювання, експерименту, порівняння, аналізу, направлені на модифікацію існуючих методів та моделей. Вони були покладені в основу розроблюваної моделі.

Обробка зображень складається з кількох етапів: імпортування зображення, аналіз, маніпуляція та вихід зображення. Існує два способи обробки зображень: цифровий і аналоговий. Зокрема, про цифрову обробку зображень та її прийоми і йдеться в цій дисертації.

Комп'ютерні алгоритми відіграють вирішальну роль в обробці цифрових зображень. Розробники використовують різні алгоритми та методи для виявлення цифрових зображень, їх аналізу, реконструкції, відновлення, стиснення даних зображення, покращення зображення, оцінку зображення та оцінку спектру зображення.

У рамках дослідження, проведеного під час написання дисертаційної роботи, були проаналізовані існуючі методи обробки зображень, виконано оцінювання їх переваг та недоліків. Було побудовано нейронну мережу, яка дозволяє з великою точністю виявити задимленість на супутникових зображеннях швидше у порівнянні з іншими наявними моделями.

Використання такої моделі на практиці це можливість своєчасно виявляти лісові пожежі та допомагати приймати оперативні рішення щодо їх ліквідації.

Наукова новизна полягає у наступному:

1. Вперше:

- 1) Розроблена модель для ефективного виявлення лісових пожеж на ранніх стадіях у порівнянні з існуючими моделями (AlexNet, VGGNet, Inception, MobileNet), яка, на відміну від існуючих моделей, дозволяє точніше виявляти лісові пожежі на початкових етапах.
- 2) Розроблена гібридна модель, яка, на відміну від існуючих моделей (AlexNet, VGGNet, Inception, MobileNet), забезпечує високу точність результатів (92%) за рахунок сукупного використання алгоритмів навчання нейронної мережі на основі 4 розглянутих моделей;
- 3) Розроблена комп'ютерна гібридна модель, яка, на відміну від існуючих моделей, виявляє лісові пожежі за даними на супутникових знімках точніше на початковій стадії із застосуванням нейромережових технологій.

2. Удосконалено:

- 1) Методи обробки потоків даних зображень в системі раннього реагування для моніторингу надзвичайних ситуацій, що дозволяє точніше фіксувати лісові пожежі на ранніх етапах у порівнянні з існуючими моделями та методами.
- 2) Методи комп'ютерного зору з відновлення та реконструкцій зображень, спектрального аналізу зображень, що дозволяє підвищити точність детектування лісових пожеж з використанням технологій обчислювального інтелекту на ранніх етапах.

3. Отримало подальший розвиток:

- 1) Модель моніторингу та раннього виявлення пожеж з метою попередження надзвичайних ситуацій та прийняття релевантних рішень для їх своєчасної ліквідації.

Практичне значення одержаних результатів. Результати проведеного дослідження у рамках дисертаційної роботи можуть бути впроваджені в існуючі системи підтримки прийняття рішень для своєчасного виявлення та ліквідації надзвичайних ситуацій природного характеру, а саме лісових пожеж. Окрім цього, результати дослідження актуальні для раннього виявлення пожеж, що викликані щоденними обстрілами внаслідок російської агресії проти України. Також результати дослідження можуть бути корисними для інтеграції з дронами та іншими системами, які допоможуть оперативно виявляти осередки пожеж та приймати відповідні рішення з їх ліквідації.

Головіна Н. В. проводила науково-дослідну роботу в Херсонському національному технічному університеті у межах виконання дисертаційного дослідження та у межах плану проведення науково-дослідної роботи кафедрою Програмних засобів і технологій Херсонського національного технічного університету за темою «Моделі та методи територіальної інформаційної системи підтримки управління евакуацією населення при надзвичайних ситуаціях», має АКТ про впровадження результатів дисертаційної роботи.

Апробація результатів дисертації була здійснена на таких конференціях:

1. Головіна Н.В., Ляшенко О.М. Дослідження методів комп'ютерного зору з відновлення та реконструкції зображень. *Матеріали IV Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених за тематикою «Сучасні комп'ютерні системи та мережі в управлінні»*: збірка наукових праць / Під редакцією Г.О. Райко. – Херсон: Видавництво ФОП Вишемирський В. С., 2021. С. 206-209.

2. Olena Liashenko, Dmytro Kyryichuk, Nadiia Holovina, Viktor Prachyk, Svitlana Bunkus. Development of information technology for monitoring of forest ecosystems under conditions of wildfires. *SGEM International Scientific Conferences on Earth & Planetary Sciences Extended Scientific Sessions „GREEN SCIENCES FOR GREEN LIFE“* (Vienna, 6-9 December, 2022). Volume 22. Issue 3.2. (Scopus, ISI Web of Science Clarivate).
3. Головіна Н.В. Методи обробки потоків даних зображень в системі підтримки прийняття рішень для моніторингу надзвичайних ситуацій. *Синергія науки і бізнесу у повоєнному відновленні Херсонщини*. (ХНТУ, 26-28 квітня 2023) у 2-х т. ; Т. 1 / за ред. О. В. Чепелюк. – Одеса : Олді+, 2023. С. 254-258.
4. Головіна Н.В. Особливості розробки та використання системи підтримки прийняття рішень для моніторингу та попередження надзвичайних ситуацій в Україні. *Синергія науки і бізнесу у повоєнному відновленні регіонів України : матеріали II Міжнародної науково-практичної конференції (ХНТУ, 24–26 квітня 2024 року) у 3-х т. ; Т. 2 / за ред. О. В. Чепелюк. – Одеса : Олді+, 2024. С.38-41.*

Особистий внесок здобувача полягає у тому, що всі дослідження були проведені самостійно, а результати отримані методами комп'ютерного моделювання.

Публікації

Основні результати дослідження опубліковані в 6 наукових працях, усі у наукових фахових виданнях, рекомендованих МОН України для публікації наукових досліджень:

1. Головіна Н.В., Ляшенко О.М. Дослідження методів спектрального аналізу зображень для детектування лісових пожеж з використанням технологій обчислювального інтелекту. *Вісник ХНТУ*. 2021. Том 79 (4). С. 74-84.

URL: <https://ojs.kntu.net.ua/index.php/visnyk/article/view/203/249>

DOI: <https://doi.org/10.35546/kntu2078-4481.2021.4.8>

(особистий внесок – досліджено та запропоновано використання сукупності методів спектрального аналізу зображень для моніторингу та попередження лісових пожеж).

2. Головіна Н.В., Ляшенко О.М. Дослідження методів відновлення та реконструкції зображень. *Вчені записки національного університету імені В.І. Вернадського*. Серія: Технічні науки. 2022. Том 33 (72) № 2. С. 80-85.

URL: https://tech.vernadskyjournals.in.ua/journals/2022/2_2022/13.pdf

DOI: <https://doi.org/10.32838/2663-5941/2022.2/13>

(особистий внесок – досліджено та запропоновано використання методів відновлення та реконструкції зображень для моніторингу та попередження лісових пожеж).

3. Головіна Н.В. Розроблення моделі сегментації лісових пожеж на супутникових знімках із застосування нейромережевих технологій. *Збірник наукових праць Національного університету кораблебудування імені адмірала Макарова*. 2024. № 2 (495). С. 95-102.

URL: <https://znp.nuos.mk.ua/archives/2024/2/14.pdf>

DOI: [https://doi.org/10.15589/znp2024.2\(495\).12](https://doi.org/10.15589/znp2024.2(495).12)

(особистий внесок – розроблено модель сегментації лісових пожеж на супутникових знімках із застосуванням нейромережевих технологій).

4. Головіна Н.В. Розроблення системи підтримки прийняття рішень для моніторингу та попередження лісових пожеж в Україні. *Вісник Херсонського національного технічного університету. Інформаційні технології*. 2024. №2 (89). С. 150-156.

URL:

https://journals.kntu.kherson.ua/index.php/visnyk_kntu/article/view/639/613

DOI: <https://doi.org/10.35546/kntu2078-4481.2024.2.21>

(особистий внесок – розроблено систему підтримки прийняття рішень для моніторингу та попередження лісових пожеж в Україні).

5. Головіна Н.В. Розроблення нейромережевої моделі моніторингу надзвичайних ситуацій природного характеру. *Системи та технології. Комп'ютерні науки*. 2024. №67 (1). С. 36-42.

URL: <https://st.umsf.in.ua/index.php/journal/article/view/143/150>

DOI: <https://doi.org/10.32782/2521-6643-2024-1-67.6>

(особистий внесок – розроблено нейромережеву модель моніторингу надзвичайних ситуацій природного характеру).

6. Головіна Н.В. Розробка нейромережевої моделі для моніторингу лісових пожеж природного характеру. *Вчені Записки Таврійського національного університету імені В.І. Вернадського*. Серія: Технічні науки. 2024. №3 (74). Т.35. С. 53-57.

URL:

https://www.tech.vernadskyjournals.in.ua/journals/2024/3_2024/part_1/12.pdf

DOI: <https://doi.org/10.32782/2663-5941/2024.3.1/10>

(особистий внесок – розроблено та програмно реалізовано нейромережеву модель для моніторингу лісових пожеж природного характеру).

Структура та обсяг дисертації

Дисертація складається зі вступу, чотирьох розділів, висновків для кожного розділу, висновку, бібліографічного списку із 63 найменувань, 3 додатків. Обсяг дисертації складає 164 сторінок, з яких 151 сторінки основного тексту, у тому числі 68 рисунків, 2 таблиці, 72 формули.

РОЗДІЛ 1

ОГЛЯД ІСНУЮЧИХ МОДЕЛЕЙ ТА МЕТОДІВ ОБРОБКИ ПОТОКІВ ДАНИХ ЗОБРАЖЕНЬ В СППР. ДОСЛІДЖЕННЯ ПОЖЕЖ В УКРАЇНІ

Технологія розпізнавання зображень демонструє значний потенціал для широкого застосування в різних галузях промисловості. Останні статистичні дані свідчать про те, що світовий ринок розпізнавання зображень може досягти обсягу 38,92 мільярда доларів, що підкреслює зростаючий інтерес до цієї технології.

Серед факторів, що сприяють розширенню технологій розпізнавання зображень, варто відзначити доступність інструментів з відкритим кодом, які спрощують процес програмування та роблять обчислювальні ресурси більш доступними. великі відкриті бази даних [49], такі як Pascal VOC і ImageNet, забезпечують доступ до мільйонів зображень з відповідними тегами, що сприяє розвитку та вдосконаленню програм і алгоритмів машинного навчання [14-15].

1.1 Дослідження існуючих систем класифікації моделей та методів обробки зображень

Розпізнавання зображень, або комп'ютерний зір, є технічною дисципліною, що досліджує способи автоматизації функцій, які виконує зорова система людини. Системи глибокого навчання, такі як TensorFlow від Google, DeepFace від Facebook та Project Oxford від Microsoft, є яскравими прикладами застосування цієї технології. Крім того, доступні API, такі як Google Cloud Vision, Clarifai та Imagga, дозволяють компаніям зменшити витрати на розробку, уникаючи необхідності формування дорогих команд фахівців у галузі комп'ютерного зору.

Основними методами обробки цифрових зображень є:

- Редагування зображень, що означає зміну цифрових зображень за допомогою графічних програмних засобів.

- Відновлення зображення, що відноситься до оцінки чистого вихідного зображення з пошкодженого зображення, зробленого з метою повернення втраченої інформації.
- Незалежний компонентний аналіз, який розділяє багатовимірний сигнал обчислювально на адитивні підкомпоненти.
- Анізотропна дифузія, яку часто називають дифузією Перона-Маліка, дає змогу зменшити шум зображення, не видаляючи важливі частини зображення [3].
- Лінійна фільтрація. Це ще один метод обробки цифрових зображень, який відноситься до обробки змінюваних у часі вхідних сигналів і вироблення вихідних сигналів, які підлягають обмеженням лінійності.
- Нейронні мережі, які є обчислювальними моделями, широко використовуються в машинному навчанні для вирішення різноманітних завдань [31].
- Пікселяція, яка часто відноситься до перетворення друкованих зображень в оцифровані (наприклад, GIF).
- Аналіз основних компонентів, метод цифрової обробки зображень, який можна використовувати для виділення ознак.
- Differential Equations, який також має справу з ефективним видаленням шумів зображень.
- Приховані моделі Маркова, методика, яка використовується для аналізу зображень у двох вимірах.
- Вейвлети, що означає математичну функцію, яка використовується під час стиснення зображень.
- Самоорганізуючі карти – техніка цифрової обробки зображень для класифікації зображень на кілька класів.

У рамках дослідження було розглянуто різні методи обробки зображень для того, щоб обрати оптимальний метод навчання нейронної мережі [3, 31].

1.1.1 Дослідження традиційних методів обробки зображень та їх застосування

Традиційні методи обробки зображень представляють собою алгоритми та техніки, які були розроблені до виникнення глибокого навчання. Ці методи ґрунтуються на виконанні математичних операцій над зображеннями та застосуванні евристичних правил.

Серед традиційних методів обробки зображень [55] виділяють такі техніки, як сегментація зображень, виділення ознак та математичні морфологічні операції. Ці методи забезпечують ефективні рішення для ряду задач, таких як аналіз зображень [57], комп'ютерний зір та обробка сигналів, і залишаються важливими інструментами в арсеналі дослідників та практиків у цій галузі.

1.1.1.1 Дослідження існуючих методів сегментації зображень для аналізу

Сегментація – це процес розділення зображення на окремі області, що відповідають різним об'єктам або частинам об'єктів [55-57].

Порогова сегментація – найпростіший метод, який полягає в тому, що всі пікселі з інтенсивністю, більшою за заданий поріг, відносяться до одного класу (наприклад, об'єкту), а всі інші – до іншого (фону). Регіональна сегментація базується на об'єднанні сусідніх пікселів з подібними властивостями (наприклад, кольором, інтенсивністю) в регіони. Кластерна сегментація розглядає пікселі зображення [57] як точки в багатовимірному просторі ознак. Потім застосовується алгоритм кластеризації (наприклад, K-means) для розподілу пікселів на кілька кластерів.

Сегментація зображень є критично важливим етапом у багатьох застосуваннях комп'ютерного зору.

1.1.1.2 Дослідження та огляд методів виділення ознак

Ознаки – це кількісні характеристики, які описують властивості об'єктів на зображенні [55, 57]. Кольорові ознаки (середнє значення, дисперсія, гістограма кольорів) використовуються для опису кольорових властивостей об'єктів. Текsturні ознаки (ентропія, контраст, кореляція) використовуються для опису неоднорідності поверхні. Геометричні ознаки (форма, розмір, орієнтація об'єктів) використовуються для опису геометричних властивостей.

Ознаки об'єктів на зображеннях – це ключові характеристики, які дозволяють здійснювати аналіз та класифікацію об'єктів у комп'ютерному зорі. Кольорові, текстурні та геометричні ознаки забезпечують різноманітні підходи до опису об'єктів, що дозволяє адаптувати методи обробки зображень до специфічних завдань [57].

1.1.1.3 Дослідження та огляд математичних морфологічних операцій

Математична морфологія – це розділ математики, який вивчає форми і їх зміни. Морфологічні операції використовуються для аналізу бінарних зображень [55].

Ерозія є однією з основних морфологічних операцій, яка зменшує розміри об'єктів на зображенні [55, 57]. Наприклад, у випадку виявлення лісових пожеж [12-13], ерозія може допомогти видалити незначні фонові елементи, які можуть заважати точному аналізу основних об'єктів.

Дилатація додає пікселі до країв об'єктів, що корисно для заповнення невеликих дірок або розширення об'єктів [57]. Відкриття використовується для видалення малих об'єктів і заповнення невеликих дірок в об'єктах. Спочатку застосовується дилатація, потім ерозія.

Математична морфологія є потужним інструментом для аналізу та обробки бінарних зображень. Операції ерозії, дилатації, відкриття та закриття

дозволяють ефективно маніпулювати формами об'єктів, що сприяє покращенню якості аналізу зображень [55-58].

1.1.2 Дослідження та вивчення сучасних моделей глибокого навчання

Глибоке навчання – це підгалузь машинного навчання [31], яка використовує штучні нейронні мережі з багатьма шарами для аналізу даних. Виділяють такі основні моделі глибокого навчання, що використовуються для аналізу та розпізнавання зображень [55, 64]:

1. Згорткові нейронні мережі (CNN)

CNN – це тип нейронних мереж, спеціально розроблений для обробки даних з сітчастою структурою, таких як зображення. Вони використовують згорткові шари для виявлення локальних ознак в зображенні [39, 64].

VGG – одна з перших глибоких CNN архітектур, що продемонструвала високу ефективність у класифікації зображень [58]. Її головна особливість – використання великої кількості маленьких згорткових ядер. ResNet – ця архітектура вирішує проблему зникнення градієнта в глибоких мережах за допомогою залишкових з'єднань. Це дозволяє навчати набагато глибші мережі.

Inception – у цій мережі використовуються паралельні згорткові шари з різними розмірами ядер, що дозволяє витягувати ознаки різного масштабу. EfficientNet – це сімейство нейронних мереж, оптимізованих за розміром і точністю. Вони досягають високої точності при менших обчислювальних витратах [55].

2. Сегментаційні моделі (U-Net, Mask R-CNN)

Сегментація зображень за допомогою глибокого навчання дозволяє виділити на зображенні окремі об'єкти або області з різними характеристиками [57].

3. Генеративні моделі (GANs) для синтезу зображень

GANs складаються з двох мереж: генератора, який створює нові зображення, і дискримінатора, який відрізняє справжні зображення від синтезованих [55, 57].

Глибоке навчання є потужним інструментом для аналізу та обробки складних даних, зокрема зображень. Використання різних архітектур (рис. 1.4), таких як CNN, U-Net, Mask R-CNN та GANs, дозволяє досягати високої точності в розпізнаванні, класифікації та генерації зображень [55-57].

1.1.3 Дослідження гібридних методів обробки зображень

Гібридні методи об'єднують переваги як традиційних методів обробки зображень, так і глибокого навчання. Вони дозволяють вирішувати складні завдання, які важко або неможливо вирішити за допомогою окремих підходів [55, 57].

1.2 Огляд існуючих систем підтримки прийняття рішень на основі зображень

Системи підтримки прийняття рішень на основі зображень (СППРЗ) відіграють важливу роль у моніторингу лісових пожеж [12-13]. Завдяки розвитку технологій дистанційного зондування та штучного інтелекту, такі системи здатні ефективно виявляти, відстежувати та прогнозувати поширення пожеж [18, 24-26].

Вільям Уоллес і Франк Де Балог були серед перших дослідників, які визначили чотири ключові фази управління кризою: запобігання, підготовка, відповідь і відновлення [24-26].

Даніела Фоглі та Джованні Гіда акцентують увагу на необхідності структурованого та скоординованого управління реагуванням на кризу, яке має бути підкріплене системою підтримки прийняття рішень. На думку авторів, зручність використання системи значною мірою залежить від чіткого

визначення концепцій і її здатності моделювати різноманітні кризові ситуації [37-38].

Додаткові вимоги, висунуті Фоглі та Гіда, доповнюють підхід, запропонований Мікою Ендслі, який спрямований на підтримку осіб, що приймають рішення, в умовах складних і динамічних ситуацій [49]. Цей підхід ґрунтується на трьох основних етапах: сприйнятті елементів середовища, розумінні поточної ситуації та прогнозуванні її розвитку в найближчому майбутньому [24-26].

Аналіз зображень для детектування природних катастроф з використанням технологій обчислювального інтелекту з метою запобігання та збереження життя людей є предметом дослідження автора роботи [11]. У цій роботі запропоновано інноваційний підхід «розробки за допомогою моделювання» для створення та впровадження алгоритму виявлення пожеж за допомогою TMS320DM642 DSP та MATLAB/Simulink [5, 24-26].

У роботі [12] розглядається алгоритм виявлення пожежі, розроблений на основі технологій комп'ютерного зору. Запропонований алгоритм складається з двох основних компонент: моделювання кольору вогню та виявлення напрямку його руху [18, 39].

Дослідження [13] присвячене методам геоінформаційного моніторингу, які використовуються для спостереження та гасіння лісових пожеж [18]. Автори описують спеціалізовану інформаційну систему моніторингу пожеж, яка забезпечує комплексний підхід до спостереження за лісовими пожежами [13]. Спеціалізована інформаційна система моніторингу пожеж (СІСМП) виконує функції збору, зберігання, обробки та поширення геоданих про пожежі в лісах, а також про умови їх виникнення та розвитку [12, 18].

1.2.1 Вивчення існуючих систем моніторингу та виявлення пожеж

1.2.1.1 Огляд сучасних комерційних систем

Комерційні системи для моніторингу [56] та виявлення лісових пожеж відіграють важливу роль у забезпеченні безпеки навколишнього середовища та зменшенні ризиків, пов'язаних із лісовими пожежами [18]. Розглянуто кілька провідних комерційних рішень у цій сфері [32].

1. SmokeD – польська компанія, що пропонує системи раннього виявлення диму та послуги з моніторингу лісових пожеж [20, 53, 54].
2. AmpliCam Wildfire Guardian – це інтегроване рішення, що поєднує в собі систему відеоспостереження, аналітичне програмне забезпечення та географічну інформаційну систему [31].
3. EOSDA – ця компанія пропонує програмне забезпечення для моніторингу лісових пожеж на основі даних NASA FIRMS [56, 62].

Комерційні системи для моніторингу та виявлення лісових пожеж [39], такі як SmokeD, AmpliCam Wildfire Guardian та EOSDA, використовують передові технології для забезпечення своєчасного реагування на загрози [12, 13, 18].

1.2.1.2 Огляд сучасних відкритих систем

Відкриті системи для моніторингу та виявлення лісових пожеж надають доступ до потужних інструментів і даних [62], які можуть бути використані дослідниками, екологами та організаціями для аналізу ризиків і реагування на надзвичайні ситуації [32]. Ці системи безкоштовні або мають відкритий доступ, що робить їх доступними для широкого кола користувачів [12].

Google Earth Engine – це хмарна платформа, яка надає доступ до великих обсягів геопросторових даних і інструментів для їх обробки [13]. OpenStreetMap – це вільна географічна база даних, яка створюється і підтримується спільнотою добровольців. Відкриті системи [53, 54] надають

потужні інструменти для моніторингу та виявлення лісових пожеж [12, 13, 32, 56, 62].

1.2.2 Порівняльна характеристика функціональності та особливостей існуючих систем

Системи моніторингу [56] лісових пожеж, такі як SmokeD, AmpliCam Wildfire Guardian, EOSDA та Google Earth Engine, демонструють різноманітність у функціональності та характеристиках [32, 65-70]. Це порівняння допоможе зрозуміти, які системи найкраще відповідають конкретним потребам користувачів, що наведено в табл. 1.1.

Таблиця 1.1. Порівняння систем моніторингу лісових пожеж

Характеристика	SmokeD	AmpliCam Wildfire Guardian	EOSDA	Google Earth Engine
Тип даних	Відео з камер	Відео з камер	Супутникові зображення	Супутникові зображення
Методи виявлення	Алгоритми аналізу відео	Алгоритми машинного навчання	Індекси згарищ	Алгоритми класифікації
Функціональність	Раннє виявлення диму, моніторинг активних пожеж	Виявлення диму та вогню, аналіз поширення пожеж	Моніторинг активних пожеж, аналіз динаміки пожеж	Аналіз великих обсягів даних, створення карт ризиків
Доступність	Комерційна	Комерційна	Комерційна	Відкрита

Існуючі системи для моніторингу лісових пожеж [56, 62] демонструють високий рівень розвитку та різноманітність у функціональності [20]. Порівняння існуючих систем для моніторингу лісових пожеж представлено на рис. 1.1 [54].



Рис. 1.1. Порівняння існуючих систем моніторингу лісових пожеж

1.3 Огляд поточної ситуації з лісовими пожежами в Україні та необхідність розробки СППР на основі аналізу зображень

Лісові пожежі в Україні є серйозною проблемою, яка завдає значної шкоди довкіллю, економіці та здоров'ю населення [52]. Вони можуть призводити до знищення великих площ лісів, погіршення якості повітря, а також до загибелі флори і фауни. За даними Державної служби України з надзвичайних ситуацій (ДСНС) [52], кількість лісових пожеж в Україні має тенденцію до зростання, що викликає занепокоєння серед екологів та органів влади [1, 65-70].

Для аналізу динаміки виникнення пожеж в Україні за різні роки були використані дані, надані Державною службою України з надзвичайних ситуацій (ДСНС), за період з 2010 по 2022 рік. Проведене дослідження виявило, що загальна кількість пожеж в Україні демонструє зростаючу тенденцію протягом вказаного періоду [29]. Найвищі показники пожежності спостерігалися у 2015 році з 114 701 випадком, у 2020 році – 80 654 випадки, а також у 2022 році – 102 376 випадків. У той же час найнижчі показники зафіксовані у 2012 році, коли було зареєстровано 44 821 пожежу, та у 2017 році

з 45 408 випадками. Загальна статистика щодо кількості лісових пожеж на території України за останні роки представлена на рисунку 1.2, що ілюструє актуальні тенденції у даній сфері [1-2, 65-70].



Рис. 1.2. Кількість лісових пожеж в Україні з 2010 по 2022 роки

Сучасна ситуація з лісовими пожежами в Україні вимагає розробки ефективних рішень для їх раннього виявлення та попередження великих катастроф. Одним із перспективних напрямків є створення системи підтримки прийняття рішень (СППР) на основі аналізу зображень [37, 38]. Використання сучасних технологій, таких як супутникові зображення та алгоритми машинного навчання, може суттєво підвищити ефективність моніторингу лісових пожеж [1-2].

Поточна ситуація з лісовими пожежами в Україні вимагає термінових заходів для їх моніторингу та управління [62]. Розробка системи підтримки прийняття рішень на основі аналізу зображень може стати ефективним інструментом для раннього виявлення та попередження пожеж [20], що, своєю чергою, допоможе зберегти природні ресурси, захистити здоров'я населення та зменшити економічні втрати. Схема (рис. 1.3) відображає актуальність даного дослідження.

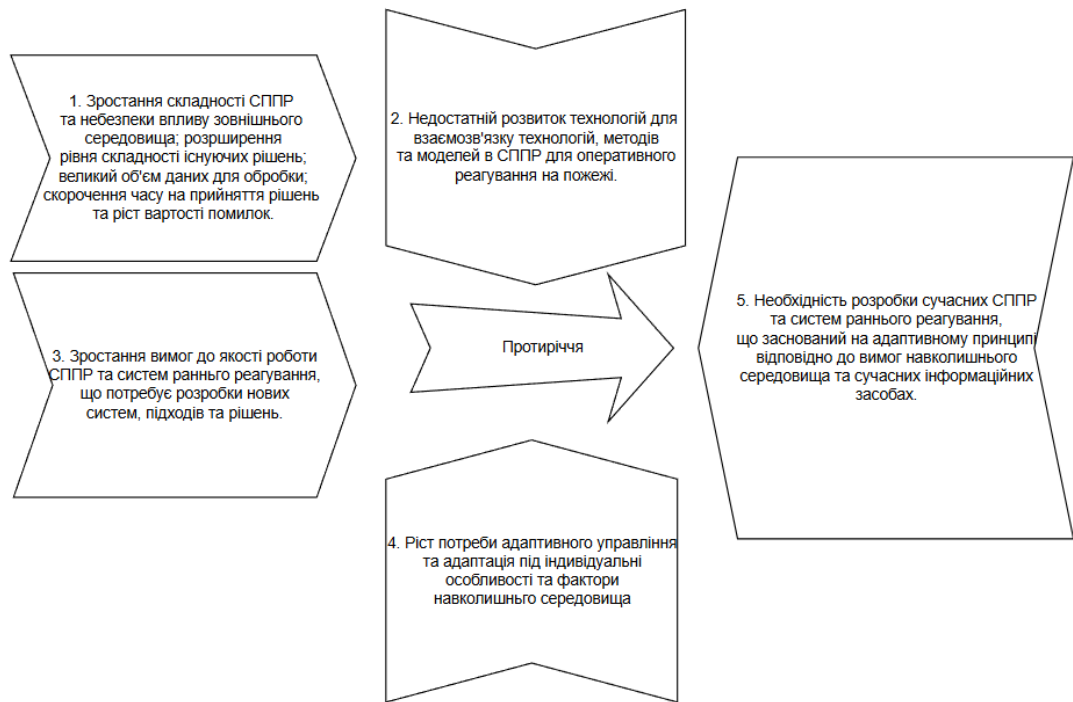


Рис. 1.3. Актуальність дослідження

У світлі зростаючих ризиків, пов'язаних із зміною клімату та людською діяльністю, такі рішення стають все більш актуальними для забезпечення безпеки екосистем та населення України [2, 54].

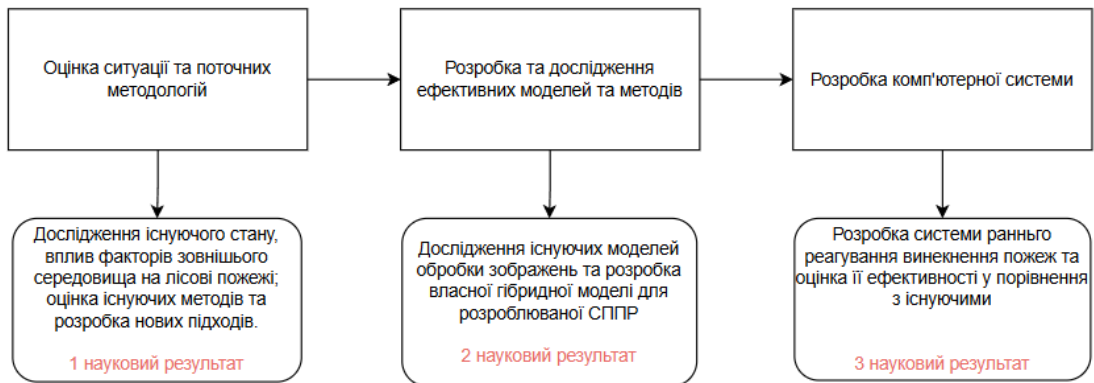


Рис. 1.4. Логічна схема вирішення завдань дисертаційного дослідження

Перед проведенням дослідження були поставлені задачі, які розглядаються та досягаються під час написання дисертації. Розроблена логічна схема вирішення завдань (рис. 1.4).

1.4 Висновки до першого розділу

У першому розділі було проведено огляд існуючих методів обробки зображень та систем підтримки прийняття рішень, що використовуються для моніторингу лісових пожеж. Аналіз літератури показав, що глибоке навчання демонструє високу ефективність у задачах сегментації та класифікації зображень. Однак, існуючі моделі мають певні обмеження, такі як високі вимоги до обчислювальних ресурсів та необхідність великих обсягів даних для навчання.

РОЗДІЛ 2

МОДЕЛЮВАННЯ: МАТЕМАТИЧНІ МОДЕЛІ ТА АРХІТЕКТУРА СИСТЕМИ СППР

Лісові пожежі є серйозною проблемою в Україні, завдаючи значної шкоди довкіллю, економіці та здоров'ю населення. Щорічно в країні виникає тисячі лісових пожеж, які знищують значні площі лісів, призводять до мільйонних збитків та можуть загрожувати життю людей. Ця ситуація підкреслює необхідність впровадження ефективних методів моніторингу та управління ризиками, пов'язаними з лісовими пожежами [56, 62].

Традиційні методи, такі як наземні патрулі та спостереження з повітря, часто виявляються неефективними та фінансово затратними. Ці підходи не завжди забезпечують своєчасне виявлення загроз, що може призводити до затримок у реагуванні на пожежі та збільшення їх негативних наслідків. У зв'язку з цим, використання сучасних технологій, таких як дистанційне зондування Землі, штучний інтелект та системи підтримки прийняття рішень (СППР), стає все більш актуальним [60, 61, 103]. Ці технології здатні автоматизувати процеси збору та аналізу даних, що підвищує ефективність виявлення загроз і прийняття рішень [4, 73].

Необхідність розробки програмного забезпечення, яке б забезпечувало виявлення лісових пожеж у реальному часі, стає критично важливою. Таке програмне забезпечення повинно інтегрувати різноманітні джерела даних, зокрема супутникові знімки, метеорологічні дані та інформацію про стан лісів, щоб сформувати комплексну картину ризиків. Математичні моделі, що лежать в основі таких систем, повинні враховувати різні фактори, такі як температура, вологість, швидкість вітру та інші параметри, які можуть впливати на ймовірність виникнення пожеж [71-75].

Розділ присвячений детальному опису математичних моделей, які використовуються для моделювання процесів, пов'язаних з лісовими пожежами, а також архітектурі системи підтримки прийняття рішень. Огляд

математики та алгоритмів, які лежать в основі системи, дозволить зрозуміти, як ці моделі можуть бути застосовані для прогнозування ризиків і оптимізації реагування на пожежі [53, 54, 75].

Таким чином, розділ надасть теоретичне обґрунтування необхідності розробки системи, а також практичні деталі щодо математичних моделей і архітектурних рішень, які стали основою для створення ефективної системи підтримки прийняття рішень у контексті моніторингу лісових пожеж [62].

Лісові пожежі становлять серйозну загрозу для України, завдаючи значної шкоди екології, економіці та здоров'ю населення. Щороку в країні виникають тисячі лісових пожеж, які знищують великі площі лісових масивів. Для ефективної боротьби з цією проблемою необхідно впроваджувати дієві системи моніторингу та попередження лісових пожеж [56, 62].

Аналіз зображень є важливим елементом у розробці таких систем. Завдяки сучасним технологіям, зокрема машинному навчанню та комп'ютерному зору, стало можливим автоматичне виявлення лісових пожеж на супутникових знімках та інших типах зображень [31].

У цьому контексті було розглянуто інформаційну систему (ІС), яка спеціалізується на аналізі геоданих і була розроблена для моніторингу змін у рослинності. Ця система [53, 54] використовує дані дистанційного зондування Землі (ДЗЗ) для виявлення та спостереження за змінами в рослинному покриві [56].

У цьому дослідженні [4] розглядаються методи моделювання екологічно небезпечних територій за допомогою геоінформаційних систем. У цьому дослідженні [6] автори розглядають доступні методи та способи моніторингу лісових пожеж. Вони розкривають поняття ГІС та систем обчислювального інтелекту [53, 54].

Проблема сегментації лісових пожеж на супутникових зображеннях є критично важливою для моніторингу та управління процесами боротьби з лісовими пожежами [76-80].

Серед основних викликів, які потребують вирішення, можна виділити наступні:

- нестача даних для навчання;
- варіативність даних;
- складність точної сегментації країв пожежі;
- складність інтерпретації зображень.

Саме ці завдання було поставлено за мету вирішити у ході дослідження лісових пожеж на території України.

2.1 Дослідження теорії та практичних аспектів моделювання

2.1.1 Вивчення характеристик та особливостей згорткових нейронних мереж: архітектура та принципи функціонування

Згорткові нейронні мережі (CNN) є одними з найпопулярніших методів глибокого навчання, що застосовуються в задачах обробки зображень [34-36, 77]. Їх архітектура натхненна біологічними процесами, що відбуваються в зоровому кортексі. На відміну від звичайних нейронних мереж, CNN використовують згорткові шари, які застосовують фільтри до вхідного зображення для виділення локальних ознак [30, 76].

Основними компонентами CNN є:

- згорткові шари – виконують згортку вхідного зображення з навчасними фільтрами для отримання feature maps;
- пулінгові шари – зменшують розмірність feature maps та забезпечують інваріантність до невеликих змін в розташуванні ознак;
- повністю зв'язані шари використовуються для класифікації і регресії на основі витягнутих ознак.

CNN здатні виявляти локальні ознаки в зображенні, що є критично важливим для сегментації, де необхідно точно визначити межі об'єктів.

Формула для операції згортки [23, 79]:

$$(I * K)(x, y) = \sum_{m,n} (I(x + m, y + n) * K(m, n)), \quad (2.1)$$

де I – вхідне зображення, K – фільтр (ядро), (x, y) – координати пікселя в результаті, m, n – індекси, що охоплюють розміри фільтра.

Для кожного пікселя (x, y) у вихідному зображенні ми обчислюємо значення, беручи суму добутків значень пікселів вхідного зображення, які знаходяться під фільтром, і відповідних значень фільтра.

Формула для активаційної функції ReLU [23]:

$$f(x) = \max(0, x) \quad (2.2)$$

Ця функція активує нейрон, якщо його вхідне значення більше нуля, і повертає нуль в іншому випадку.

Формула для максимального пулінгу [23]:

$$P(x, y) = \max_{i,j} (F(x+i, y+j)), \quad (2.3)$$

де F – feature map, а (x, y) – координати пікселя в результаті.

Максимальне пулінгування вибирає максимальне значення з підвибірки (вікна) розміром, наприклад, 2×2 , що дозволяє зменшити розмірність даних, зберігаючи при цьому важливу інформацію.

Формула для крос-ентропії [23, 81-85]:

$$L(y, \hat{y}) = \sum_i (y_i * \log(\hat{y}_i)) , \quad (2.4)$$

де y – справжні мітки, $\hat{y}$ – передбачені ймовірності.

Крос-ентропія вимірює різницю між справжніми мітками та передбаченими ймовірностями, що використовується як функція втрат для навчання моделі.

Формула для оновлення ваг у процесі навчання [34, 35]:

$$w = w \eta * \nabla L, \quad (2.5)$$

де w – ваги, η – швидкість навчання, ∇L – градієнт функції втрат.

Це оновлення ваг відбувається в напрямку, протилежному градієнту, щоб зменшити функцію втрат. Формула для розрахунку виходу після згортки [34]:

$$Output\ Size = \frac{(W-F+2P)}{s} + 1, \quad (2.6)$$

де W – ширина вхідного зображення, F – розмір фільтра, P – паддінг (padding), S – крок (stride).

Контракційна частина (Encoder) [23]:

$$Feature\ maps = \left\lceil \frac{(input\ size + 2 * padding - kernel\ size)}{stride} \right\rceil + 1 \quad (2.7)$$

Розмір feature maps змінюється в результаті застосування згортки з урахуванням паддінгу та кроку.

Розширювальна частина (Decoder) [34, 35, 81-85]:

$$Output_size = (input_size - 1) * stride + 2 * padding + kernel_size \quad (2.8)$$

Розмір виходу змінюється в результаті застосування операцій в розширювальній частині мережі, що зазвичай включає в себе операції, такі як транспонована згортка.

Формула для розрахунку кількості параметрів у згортковому шарі [35]:

$$\text{Params} = (K_h * K_w * C_{in} + 1) * C_{out} \quad (2.9)$$

де K_h , K_w – висота і ширина ядра, C_{in} – кількість вхідних каналів, C_{out} – кількість вихідних каналів.

Формула дозволяє визначити загальну кількість параметрів, які потрібно навчити у згортковому шарі [81-85, 103].

2.1.2 Проведення дослідження моделі сегментації на основі повністю з'єднаних згорткових мереж

Модель сегментації на основі повністю з'єднаних згорткових мереж дозволяє отримати більш точні результати сегментації, особливо для задач, де необхідно виділити дрібні деталі або об'єкти складної форми. Архітектура моделі має кілька складових.

Базова архітектура CNN використовується попередньо навчена модель CNN (наприклад, VGG, ResNet) як основа. Додаються один або кілька повністю з'єднаних шарів після останнього згорткового шару. Вихідні нейрони цих шарів відповідають кількості класів сегментації.

Для збільшення розмірності вихідного тензора і отримання сегментаційної маски того ж розміру, що й вхідне зображення, використовується транспонована згортка. Для покращення точності сегментації дрібних деталей часто використовують skip-connections, які з'єднують виходи проміжних шарів encoder з відповідними шарами decoder.

Процес сегментації зображення складається з кількох кроків. Зображення подається на вхід моделі. Проводиться екстракція ознак. Згорткові шари витягують високорівневі ознаки зображення.

Повністю з'єднані шари класифікують кожен піксель в один з класів. Транспонована згортка перетворює вихід повністю з'єднаних шарів в

сегментаційну маску того ж розміру, що й вхідне зображення. Можуть застосовуватися додаткові методи пост-обробки, такі як усунення шуму, згладжування меж сегментів [3, 87, 100-103].

Для навчання моделі використовується спеціальна функція втрат, яка враховує піксельні передбачення моделі та відповідні мітки в масці сегментації. Зазвичай використовується крос-ентропія або Dice loss [51, 55].

Було розглянуто формули та проведено розрахунки моделей сегментації зображення.

Згорткові шари виконують згортку вхідного зображення з фільтрами для отримання feature maps.

Операція згортки в згорткових шарах є основним етапом обробки зображень, що дозволяє виділити важливі ознаки. Формула, що описує цю операцію, має вигляд [34-36]:

$$(I * K)(x, y) = \sum_m \sum_n I(x + m, y + n) * K(m, n), \quad (2.10)$$

де I – вхідне зображення, K – фільтр, (x, y) – координати пікселя в результаті.

У цій формулі I представляє вхідне зображення, яке підлягає обробці, тоді як K є фільтром або ядром, що застосовується до зображення. Координати (x, y) вказують на позицію пікселя в результаті згортки, а індекси m та n охоплюють розміри фільтра [34-36].

Повністю з'єднані шари використовуються для класифікації кожного пікселя зображення. Вони виконують функцію класифікації, перетворюючи вхідні дані в ймовірності класів. Формула, що описує цей процес, виглядає наступним чином [34-36]:

$$y = W * x + b, \quad (2.11)$$

де y – вихідний вектор, W – матриця ваг, x – вхідний вектор, b – вектор зсуву.

У цій формулі y є вихідним вектором, що представляє результати класифікації. W – це матриця ваг, що містить параметри, які підлягають навчанню, і визначає, як вхідні дані впливають на вихід. x – вхідний вектор, отриманий з попередніх шарів, а b – вектор зсуву, що додається для забезпечення додаткової гнучкості моделі [86, 101-103].

Транспонована згортка використовується для збільшення розмірності вихідного тензора. Транспонована згортка, також відома як дезгортка, використовується для збільшення розмірності вихідного тензора. Формула, що описує цю операцію, має вигляд [34-36]:

$$O = (I * K^T)(x, y), \quad (2.12)$$

де K^T – транспоноване ядро. У цій формулі O представляє вихідний тензор, отриманий в результаті транспонованої згортки. I – це вхідне зображення або тензор, що підлягає обробці, а K^T – транспоноване ядро, яке застосовується для збільшення розмірності [90].

Для задач сегментації часто використовується крос-ентропія або Dice loss (функція втрат).

Функція втрат є критично важливим компонентом у навчанні моделей сегментації. Формула для крос-ентропії, яка часто використовується в цій сфері, виглядає так [34-36]:

$$L(y, \hat{y}) = - \sum_i (y_i * \log(\hat{y}_i)), \quad (2.13)$$

де y – справжні мітки, $\hat{y}$ – передбачені ймовірності.

Крос-ентропія вимірює відстань між справжніми мітками та передбаченими ймовірностями, що дозволяє оцінити якість класифікації [86-90].

Skip-connections використовуються для збереження детальної інформації про зображення. Вони є важливим елементом архітектур нейронних мереж, що дозволяє зберігати детальну інформацію про зображення. Формула, що описує цей процес, має вигляд [34-36]:

$$F(x) = x + G(x), \quad (2.14)$$

де $G(x)$ – функція, що представляє обробку в проміжних шарах.

У цій формулі $F(x)$ є виходом функції, що об'єднує інформацію з двох різних шарів. x – це вхідні дані, які проходять через певний шар, а $G(x)$ – функція, що представляє обробку в проміжних шарах [35]. Використання skip-connections дозволяє зберігати важливу інформацію, яка може бути втрачена в процесі згортки, що є особливо важливим у архітектурах, таких як U-Net, де необхідно зберегти контекст для точного відновлення сегментації [36, 88].

На початковому етапі вхідне зображення подається на вхід моделі. Це зображення може бути отримане з різних джерел, таких як цифрові камери, сканери або інші пристрої, що генерують растрові зображення. Вхідне зображення є масивом пікселів, де кожен піксель містить інформацію про колір (наприклад, у форматі RGB) [34-36, 89].

На етапі екстракції ознак згорткові шари моделі виконують операцію згортки, щоб витягти високорівневі ознаки з вхідного зображення. згорткові шари застосовують фільтри (ядра) до зображення, що дозволяє виявити локальні патерни, такі як краї, текстури та інші важливі деталі. В результаті цієї операції формується набір feature maps, які містять інформацію про виявлені ознаки [86-90].

На етапі класифікації пікселів повністю з'єднані шари моделі беруть отримані feature maps і класифікують кожен піксель в один з класів. Це

досягається шляхом застосування матриці ваг до вхідних даних, що дозволяє моделі приймати рішення на основі виявлених ознак. Кожен піксель отримує ймовірність належності до певного класу [34, 96-100].

Після класифікації пікселів транспонована згортка використовується для перетворення виходу повністю з'єднаних шарів в сегментаційну маску, яка має той же розмір, що й вхідне зображення. Сегментаційна маска містить інформацію про те, які пікселі належать до різних класів, що робить її основним результатом моделі сегментації [99].

На заключному етапі можуть застосовуватися додаткові методи пост-обробки для покращення якості сегментаційної маски. Це включає усунення шуму, згладжування меж сегментів або інші техніки, які допомагають поліпшити візуальну якість та точність сегментації [30, 34-36].

2.2 Дослідження та побудова архітектури розроблюваної системи підтримки прийняття рішень (СППР)

Система підтримки прийняття рішень має клієнт-серверну архітектуру. Клієнтська частина забезпечує інтерфейс користувача для завантаження зображень та перегляду результатів. Серверна частина відповідає за обробку зображень, сегментацію та класифікацію [34-36, 100-103]. Така архітектура дозволяє масштабувати систему та забезпечує гнучкість у використанні [54].

Ефективним шляхом зменшення ризиків [17], пов'язаних з природними катастрофами, є розробка та реалізація сучасних систем підтримки прийняття рішень (СППР) для кожного регіону України [58-61]. СППР [53, 91-95] дозволяє розробляти плани евакуації для різних груп населення в умовах масових лісових пожеж та організовувати розгортання пунктів евакуації, що сприяють обліку евакуйованих осіб, а також матеріальних і культурних цінностей [37].

СППР є інформаційною системою, яка підтримує процес прийняття рішень шляхом вибору оптимального варіанту та розробки і порівняння кількох альтернатив для досягнення визначених завдань і цілей [4, 38, 94].

При розробці СППР для моніторингу та попередження лісових пожеж в Україні було виявлено потреби кінцевих користувачів і створено діаграму, що відображає процес організації прийняття рішень у СППР (рис. 2.1). Для побудови всіх моделей було використано безкоштовний онлайн-сервіс для створення діаграм Draw.io [4, 6, 38].

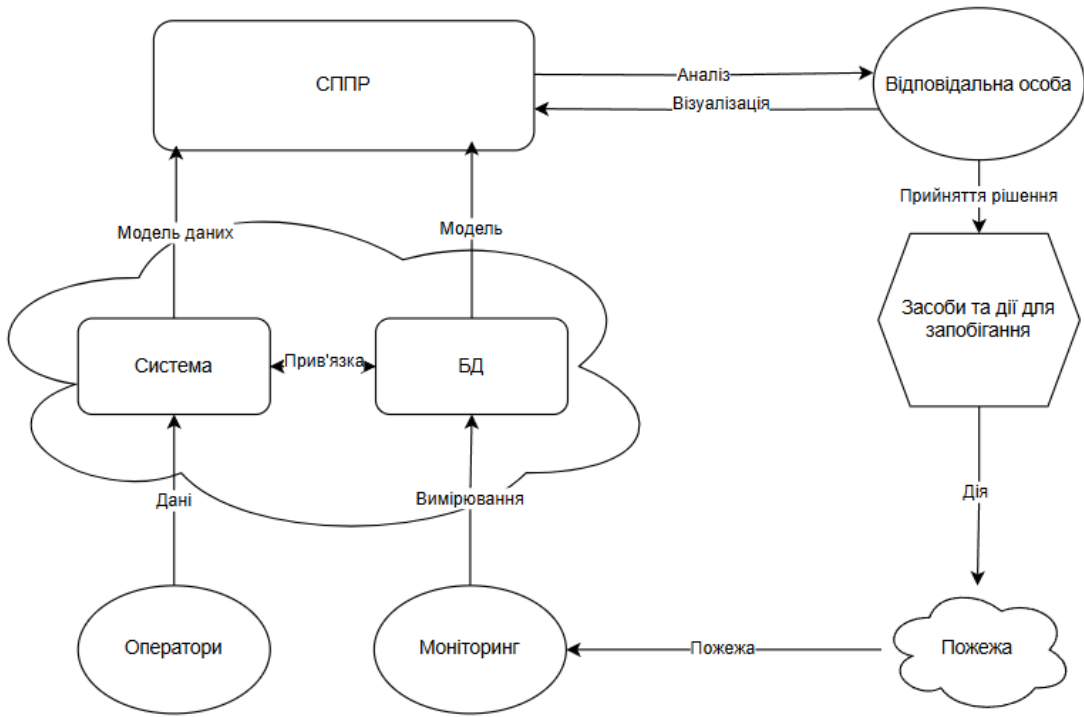


Рис. 2.1. Процес організації прийняття рішень у СППР



Рис. 2.2. Архітектура СППР

Обробка зображення передбачає кілька етапів, включаючи екстракцію ознак, класифікацію та формування сегментаційної маски. На етапі екстракції ознак згорткові шари моделі виявляють важливі патерни, такі як зміни в кольорі або текстурі, які можуть свідчити про наявність вогню або диму. Далі, на етапі класифікації, система аналізує ці ознаки та приймає рішення про ймовірність виникнення пожежі [34-36, 93]. Результати обробки зображення передаються кінцевому користувачу у формі звітів або візуалізацій, що дозволяє оперативно реагувати на потенційні загрози (рис. 2.2).

2.2.1 Розробка та опис компонентів СППР та їх взаємодії: основні принципи

Система підтримки прийняття рішень складається з кількох основних компонентів [24-26].

Вхідний модуль приймає зображення у форматах JPEG, PNG та TIFF. Зображення можуть бути завантажені користувачем вручну або автоматично передані з інших систем.

Модуль попередньої обробки [34-36, 91-95] проводить нормалізацію яскравості, контрасту та розміру зображень. Також здійснюється аугментація даних для збільшення розміру навчальної вибірки.

Модуль сегментації використовує згорткову нейронну мережу U-Net для сегментації зображень [28]. Мережа навчається на великому наборі даних, що містить зображення з ручно створеними масками сегментації.

Модуль виведення візуалізує результати сегментації у вигляді накладених на оригінальне зображення масок. Також генерується звіт, який містить метрики якості сегментації [58-61].

Користувач завантажує зображення на клієнтську частину системи. Зображення передається на сервер, де проходить перед обробку [34-36]. Далі

зображення обробляється модулем сегментації, і результати передаються назад на клієнт для візуалізації (рис. 2.3) [1-5, 6, 103].

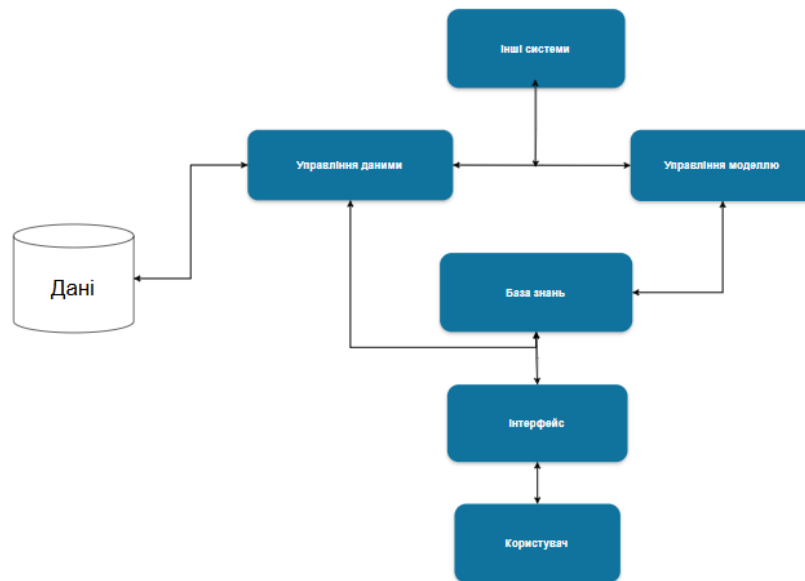


Рис. 2.3. Компоненти СППР

Компоненти системи підтримки прийняття рішень (СППР) [6] охоплюють управління даними, управління моделями, інтерфейс користувача, а також управління знаннями і користувачами [60, 61]. Ця система [54] забезпечує інтеграцію різних елементів, що сприяє більш зручному доступу до інформації та підвищенню якості прийнятих рішень [37, 38, 91-95].

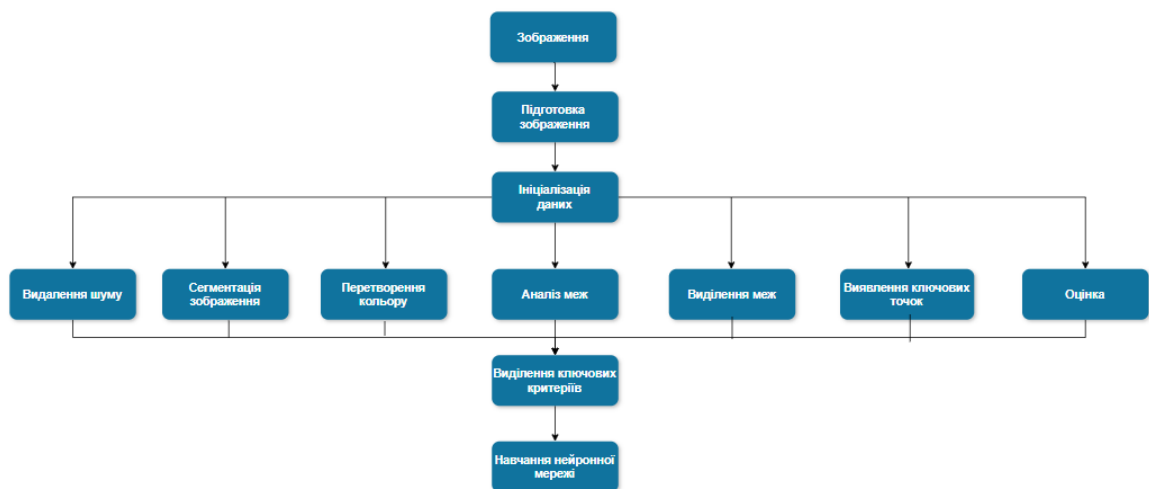


Рис. 2.4. Алгоритм аналізу зображень СППР

Розроблено принципи роботи алгоритму СППР для аналізу зображень (рис. 2.4) [58, 59]. Спочатку дані зображень зчитуються та приводяться до одного формату. Далі за допомогою методів навчання нейромережі потрібно видалити шум, сегментувати зображення, виділивши зони вогню, перетворити колір, проаналізувати межі та ключові точки. На основі цього формується оцінка зображення [37, 38].



Рис. 2.5. Схема роботи згорткової нейронної мережі

Алгоритм повторюється протягом кількох циклів для отримання більш точних результатів передбачення. Натренована нейронна мережа використовується для аналізу зображень (рис. 2.5) [27, 28]. Показано принцип роботи СППР (рис. 2.6) [24-26].

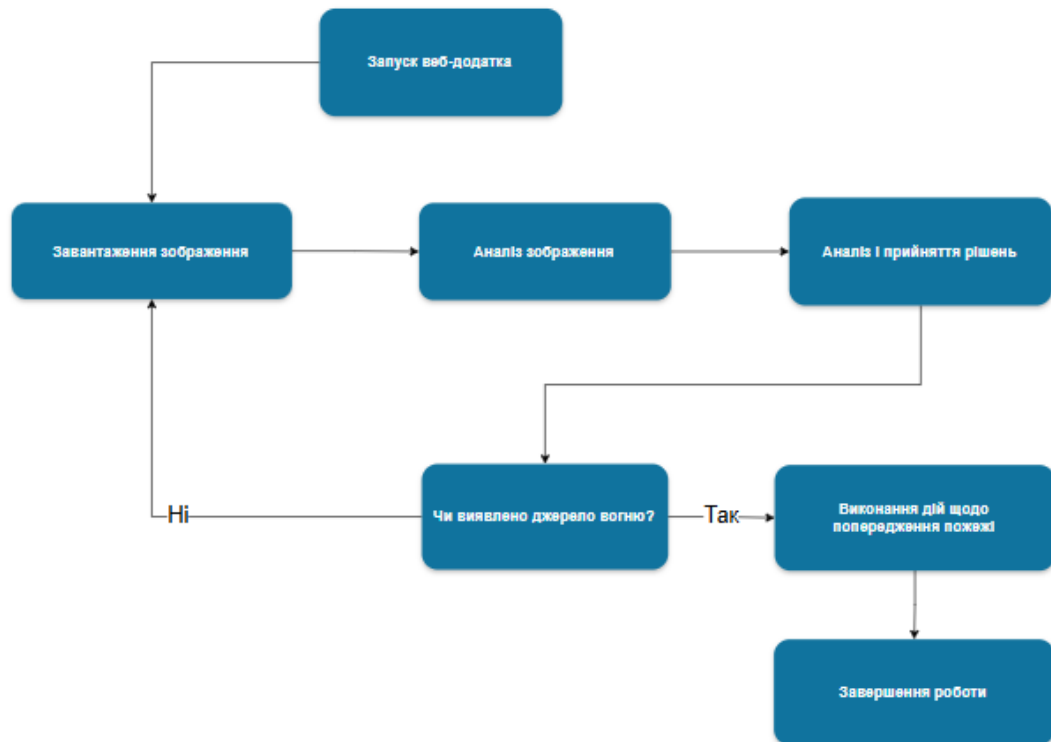


Рис. 2.6. Алгоритм роботи СППР

На основі побудованих моделей було розроблено алгоритм навчання згорткової нейронної мережі [48, 61].

Згорткова нейронна мережа – це спеціалізований тип нейронної мережі, який особливо ефективний для обробки даних, що мають структуру решітки, таких як зображення. Основна мета CNN полягає в автоматичному виявленні та розпізнаванні патернів у вхідних даних, що робить їх ідеальними для задач комп'ютерного зору, таких як класифікація зображень, виявлення об'єктів та сегментація [27, 28].

Нейронна мережа складається з кількох основних елементів [27, 28].

Згорткові шари – це основні будівельні блоки CNN. Вони виконують операцію згортки, яка дозволяє мережі виявляти локальні патерни в зображеннях.

Після кожного згорткового шару зазвичай застосовується функція активації, така як ReLU (Rectified Linear Unit), яка вводить нелінійність в модель, дозволяючи їй навчатися складнішим патернам.

Пулінг-шари зменшують розмірність даних, зберігаючи при цьому важливу інформацію. Після кількох згорткових і пулінгових шарів дані зазвичай передаються в один або кілька повнозв'язних шарів.

Вихідний шар – це останній шар мережі, який генерує ймовірності для кожного класу, якщо задача є класифікацією.

Згорткові нейронні мережі є потужним інструментом у сфері глибокого навчання, що дозволяє автоматизувати процеси аналізу та розпізнавання візуальної інформації [30]. Їх здатність виявляти складні патерни робить їх незамінними в багатьох сучасних технологіях, від медичних систем до автономних транспортних засобів [27, 28].

СППР функціонує як інформаційна система, що підтримує процес прийняття рішень [24-26], вибираючи оптимальні варіанти та розробляючи й порівнюючи кілька альтернатив для вирішення визначених завдань або цілей. Основні компоненти СППР [37] включають управління даними, управління моделями, інтерфейс користувача, а також управління знаннями і користувачами [24-26].

Описати роботу СППР можна математично наступним чином [59, 61]:

Формула прийому зображення:

$$I = \{I_1, I_2, \dots, I_n\}, \quad (2.15)$$

де I – набір зображень, I_i – окреме зображення у форматах JPEG, PNG або TIFF.

Нормалізація яскравості та контрасту:

$$I' = \frac{I - \mu}{\sigma}, \quad (2.16)$$

де I' – нормалізоване зображення, μ – середнє значення пікселів зображення, σ – стандартне відхилення.

Аугментація даних:

$$D'=\{D_1',D_2',\dots,D_m'\}, \quad (2.17)$$

де D' – новий набір зображень після аугментації, D_i' – аугментоване зображення.

Сегментація зображення:

$$S=U(I'), \quad (2.18)$$

де S – сегментоване зображення, U – функція, що представляє модель U-Net, яка обробляє нормалізоване зображення I' .

Візуалізація результатів:

$$V=I+S, \quad (2.19)$$

де V – візуалізоване зображення з накладеними масками сегментації.

Генерація звіту:

$$R=\{M_1,M_2,\dots,M_k\}, \quad (2.20)$$

де R – звіт, що містить метрики якості сегментації, такі як точність M_1 , повнота M_2 , F1-міра M_3 тощо.

Взаємодія компонентів виглядає наступним чином [57]:

Завантаження зображення: $I \rightarrow$ Вхідний модуль

Передача на сервер: $I \rightarrow$ Модуль попередньої обробки

Обробка зображення: $I' \rightarrow$ Модуль сегментації

Отримання результатів: $S \rightarrow \text{Модуль виведення}$

Візуалізація та звіт: $V, R \leftarrow \text{Клієнтська частина}$

Ці формули ілюструють ключові етапи обробки зображень у системі підтримки прийняття рішень [24-26], від прийому зображення до візуалізації результатів сегментації та генерації звіту.

2.3 Проведення дослідження, реалізація та принципи алгоритмів

Системи підтримки прийняття рішень (СППР) є важливими інструментами, які допомагають аналізувати дані та приймати обґрунтовані рішення [24-26, 38].

Вхідні дані є основою для будь-якої системи, оскільки вони визначають, які дані будуть використовуватися для аналізу та прийняття рішень. Вхідні дані можуть надходити з різних джерел, таких як бази даних, зовнішні API або користувацькі введення. Формула для оцінки обсягу даних може бути представлена як [63]:

$$D = N * F, \quad (2.21)$$

де D – загальний обсяг даних, N – кількість записів, F – кількість атрибутів (функцій) в кожному записі.

Це дозволяє оцінити, скільки даних буде оброблено системою, що є критично важливим для планування ресурсів та оптимізації процесів.

Обробка даних включає очищення, трансформацію та агрегацію даних. Цей етап є важливим для підготовки даних до подальшого аналізу. Формула для агрегації може бути представлена як [51, 55]:

$$A = \sum (d_i), \quad (2.22)$$

де A – агреговане значення, d_i – значення окремих записів.

Агрегація дозволяє отримати узагальнені показники, які можуть бути використані для аналізу тенденцій або виявлення аномалій у даних.

Моделі прийняття рішень можуть бути статистичними, заснованими на машинному навчанні або експертними системами. Формула для оцінки ймовірності прийняття рішення може виглядати так [24-26]:

$$P(R) = f(X), \quad (2.23)$$

де $P(R)$ – ймовірність прийняття рішення, X – вхідні параметри (фактори, що впливають на рішення), f – функція, що описує залежність.

Це дозволяє моделювати ймовірність різних результатів на основі вхідних даних, що є критично важливим для прийняття обґрунтованих рішень [24-26].

Результати можуть бути представлені у вигляді звітів, графіків або візуалізацій. Формула для оцінки точності результатів може бути такою [63]:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}, \quad (2.24)$$

де TP – істинно позитивні, TN – істинно негативні, FP – хибно позитивні, FN – хибно негативні.

Це дозволяє оцінити, наскільки точно система приймає рішення, що є важливим для оцінки її ефективності [63].

Інтерфейс забезпечує взаємодію між користувачем і системою. Формула для оцінки задоволеності користувача може бути представлена як [34-36]:

$$Satisfaction = \frac{Usability+Functionality+Aesthetics}{3}, \quad (2.25)$$

де Usability – зручність використання, Functionality – функціональність, Aesthetics – естетика інтерфейсу.

Це дозволяє оцінити загальне задоволення користувача від взаємодії з системою, що є важливим для її успішного впровадження та використання [100-103].

Таким чином, описані формули ілюструють основні компоненти архітектури СППР, підкреслюючи важливість кожного з етапів у процесі прийняття рішень. Від вхідних даних до виводу результатів, кожен компонент відіграє критичну роль у забезпеченні точності та ефективності системи.

2.4 Висновки до другого розділу

У другому розділі було розглянуто архітектуру та алгоритми, що лежать в основі розроблюваної системи підтримки прийняття рішень з виявлення лісових пожеж. Було показано, що використання згорткових нейронних мереж дозволяє ефективно вирішувати задачу сегментації зображень. Основним досягненням цього розділу є розробка моделі сегментації, яка забезпечує високу точність визначення меж об'єктів на зображеннях. Використання повністю з'єднаних шарів дозволило покращити якість сегментації дрібних деталей.

Перспективи подальших досліджень полягають у розробці більш стійких до шуму моделей, а також у застосуванні запропонованої моделі для вирішення інших задач комп'ютерного зору. Крім того, цікавим напрямом дослідження є розробка інтерактивних систем сегментації, які дозволяють користувачеві коригувати результати роботи моделі.

РОЗДІЛ 3

ДОСЛІДЖЕННЯ МЕТОДІВ ВІДНОВЛЕННЯ ТА РЕКОНСТРУКЦІЇ ЗОБРАЖЕНЬ

Зміни клімату та людська діяльність призводять до збільшення частоти та інтенсивності лісових пожеж, що створює нагальну потребу в ефективних методах їх виявлення та моніторингу. Спектральний аналіз зображень, зокрема з використанням технологій обчислювального інтелекту, відкриває нові можливості для своєчасного реагування на ці надзвичайні ситуації.

Розробка системи прийняття рішень, що базується на аналізі супутникових знімків, є важливим кроком у забезпеченні безпеки та швидкого реагування в умовах лісових пожеж. Використання даних з відкритих джерел, таких як NASA Earth Observatory, дозволяє отримувати актуальну інформацію про стан навколишнього середовища та виявляти потенційні загрози.

У цьому розділі буде розглянуто методи, які використовуються для спектрального аналізу зображень лісових пожеж, включаючи створення тривимірних моделей вогню та сегментацію кольорових зображень. Зокрема, буде описано використання інструментів MATLAB для виділення зон вогню та аналізу кольорових сегментів, що є критично важливими для навчання нейронних мереж [83, 96-100].

Також буде представлено метод спряжених градієнтів, який застосовується для навчання нейронної мережі, що аналізує розповсюдження вогню. Цей підхід дозволяє оптимізувати використання системних ресурсів та забезпечити високу точність результатів [11].

Результати дослідження мають практичне значення і можуть бути використані для створення веб-сервісу, що забезпечує моніторинг природних явищ у реальному часі, формує сповіщення та надає рекомендації для оперативної ліквідації наслідків лісових пожеж. Цей інструмент стане важливим ресурсом для відповідних служб, що займаються управлінням надзвичайними ситуаціями.

3.1 Підготовка бази та даних для дослідження: джерела та методи збору інформації

Питання відновлення та реконструкції зображень за допомогою різних методів є предметом активних досліджень у науковій спільноті. У даній роботі [7] здійснено аналіз трьох типів шуму, а саме: шуму «сіль і перець» (SPN), імпульсного шуму з випадковими змінами (RVIN) та спекл-шуму (SPKN). Дослідження охоплює видалення шуму на різних рівнях, що варіюються від 10% до 60%, із застосуванням п'яти різних типів фільтрів: середнього фільтра (MF), адаптивного фільтра Вінера (AWF), Гаусового фільтра (GF), стандартного середнього фільтра (SMF) та адаптивного медіанного фільтра (AMF). Аналогічні методи були застосовані до зображень дистанційного зондування, що дозволило провести їх порівняння [3, 103].

Видалення шумів із фотографій є важливим аспектом обробки зображень, що розглядається в цій роботі [1-10, 11, 12, 17, 40-43, 51]. Однією з ключових технік у цій сфері є виявлення країв, яке дозволяє визначити межі об'єктів на зображеннях.

Вихідне зображення, обране для аналізу відображено на рис. 3.1. Описати вогонь на зображенні можна за допомогою параметрів RGB, HSV, YCbCr, L*a*b*. Для аналізу зображень використовується спектр між жовтим та червоними кольорами. Коли колір вогню близький до червоного кольору та виділяє характерне світіння, можна використати цю властивість для того, щоб виявити його.

Сигнали YCbCr [46] можна отримати за рахунок обчислень гамма-налаштованого зображення у форматі RGB за допомогою виділених констант K_R , K_G , K_B . Виглядає це наступним чином [57, 101-102]:

$$Y' = K_R * R' + K_G * G' + K_B * B' \quad (3.1)$$

$$P_B = \frac{1}{2} * \frac{B' - Y'}{1 - K_B} \quad (3.2)$$

$$P_R = \frac{1}{2} * \frac{R' - Y'}{1 - K_R} \quad (3.3)$$

У даному рівнянні K_R, K_G та K_B виведені з рівняння, що визначає простір RGB, де важливою характеристикою є наступне рівняння – $K_R + K_G + K_B = 1$. Дане рівняння виділяє кольорову матрицю за розрахунками. Тут також використовуються символи гамма-корекції. Це означає, що значення R', G', B' номінально знаходяться в діапазоні дії від 0 до 1. При цьому 0 означає мінімальну інтенсивність зображення, а 1 – максимально можливе значення яскравості. Крім того, значення кольорового відображення вогню можуть варіюватися в діапазоні від -0,5 до +0,5, залежно від номінального значення. Зворотний процес перетворення може бути легко реалізований шляхом інвертування фактичного відображення.



Рис. 3.1. Вихідне зображення



Рис. 3.2. Різні колірні маски для відображення спектру вогню

Представлені різні колірні маски (рис. 3.2), які використовуються для візуалізації спектру вогню. Кожна маска підкреслює різні аспекти вогняного спектру.

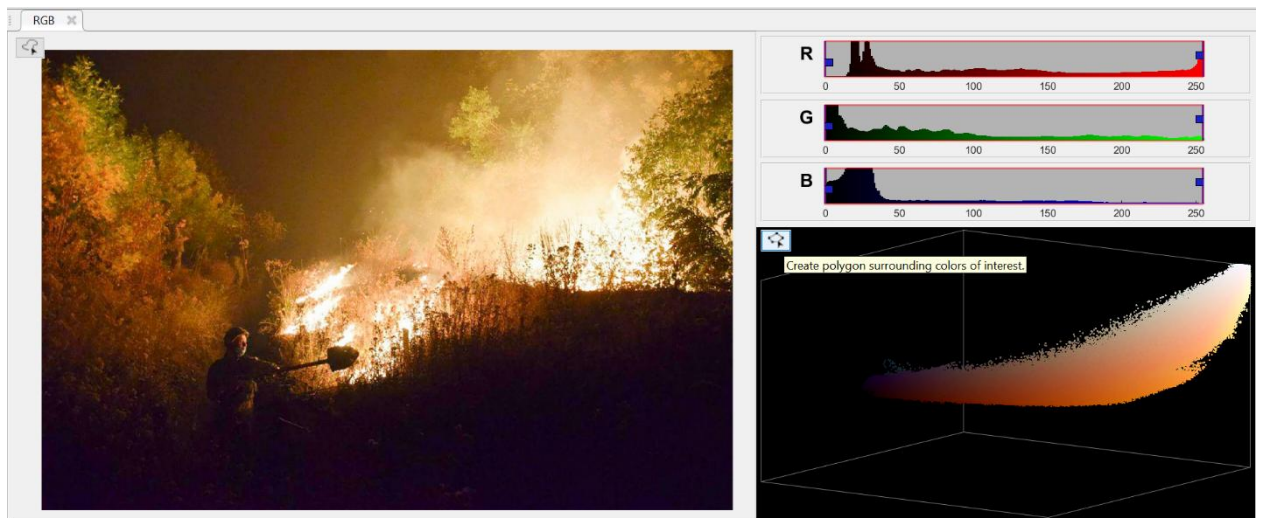


Рис. 3.3. Зображення вогню в спектрі RGB

Вогонь представлений в RGB-колірному просторі (рис. 3.3). Відображення вогню в цьому спектрі дозволяє аналізувати його кольорові характеристики.

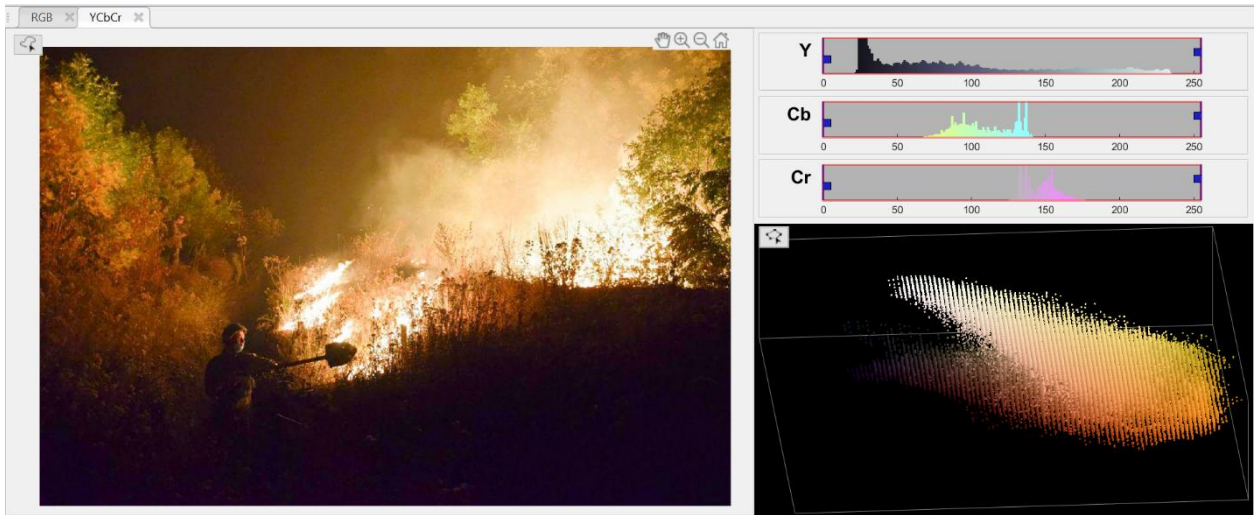


Рис. 3.4. Зображення вогню в спектрі YCbCr

Вогонь представлений у YCbCr-колірному просторі (рис. 3.4). Цей формат дозволяє краще відокремлювати яскравість від кольорової інформації.

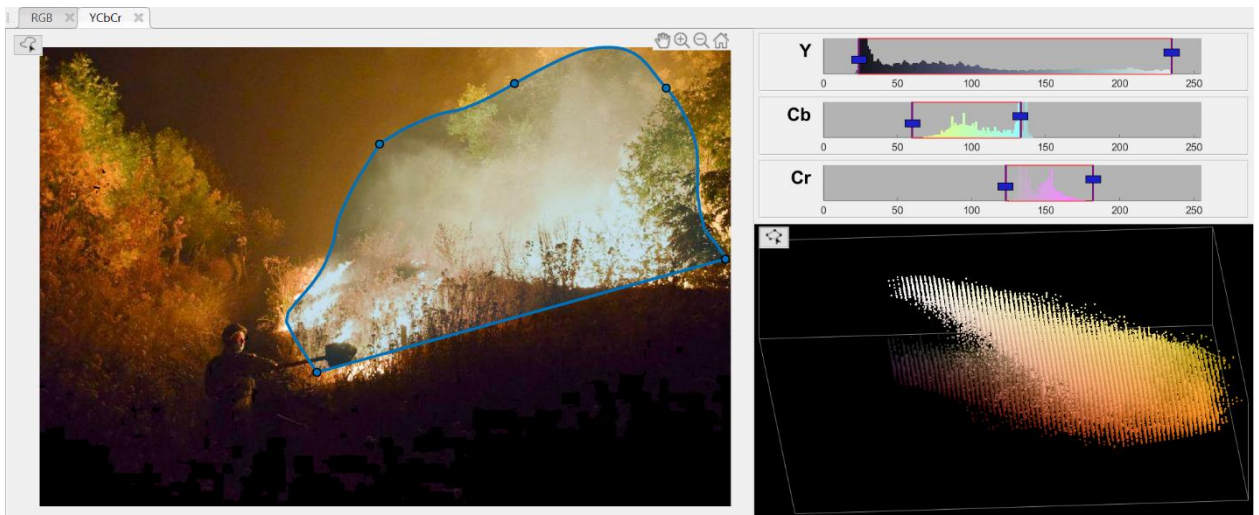


Рис. 3.5. Виділення зон активного вогню на зображенні

Проілюстровано процес виділення активних зон вогню (рис. 3.5), що є важливим для аналізу та моніторингу вогняних ситуацій.

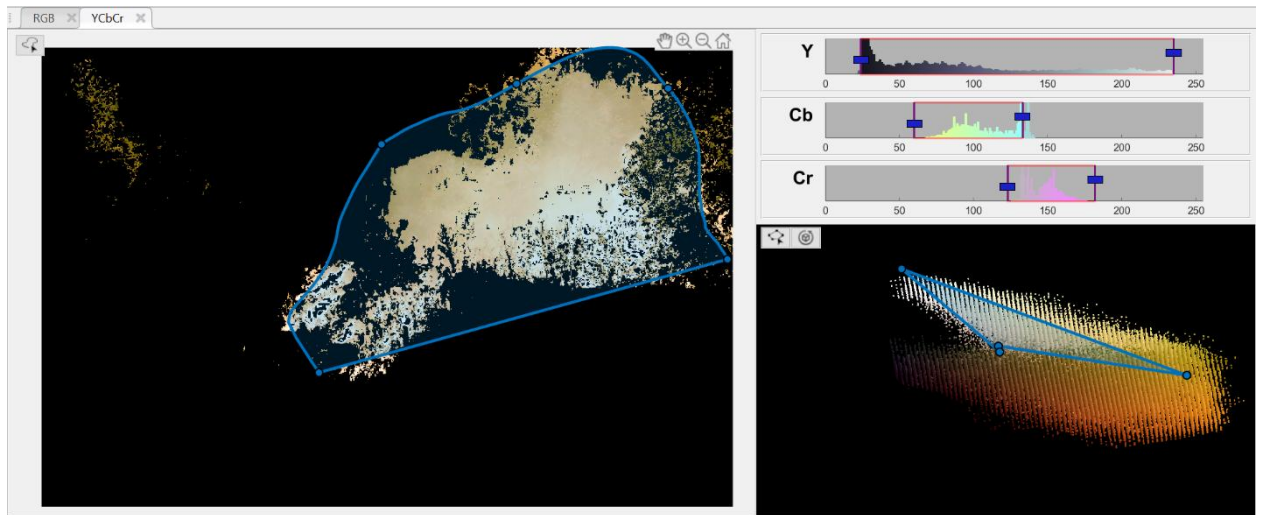


Рис. 3.6. Звуження дії спектру вогню

Показано (рис. 3.6), як звужується спектр вогню, що може бути корисним для точнішого аналізу його характеристик.

Для тривимірного відображення регіонів зображення було використано інструмент Volume Segmenter в MATLAB.

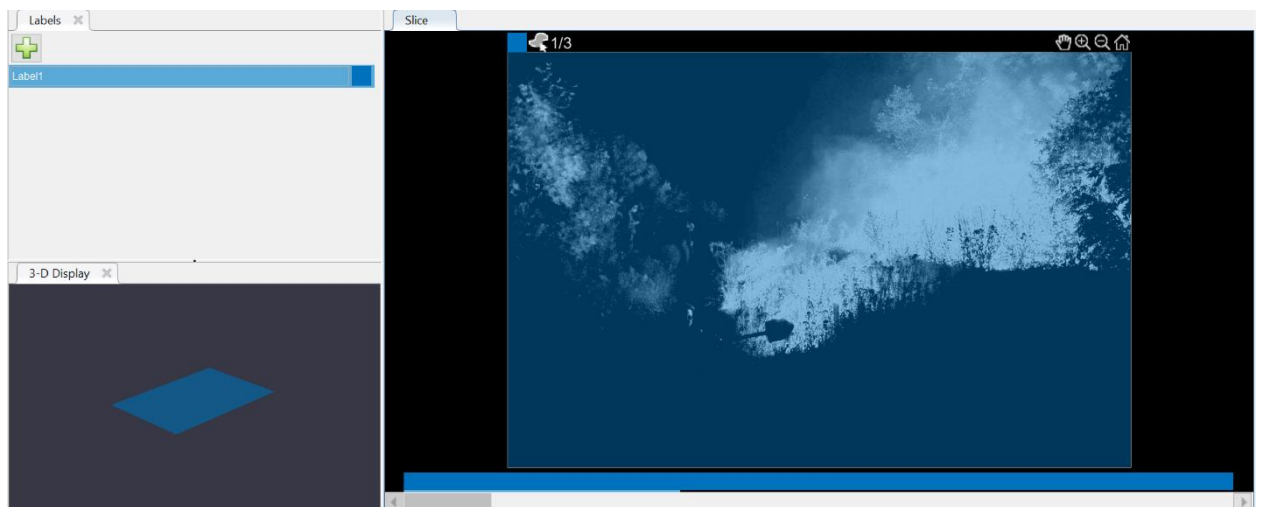


Рис. 3.7. Сегмент 1

Перший сегмент (рис. 3.7) тривимірної зони вогню, виділений за допомогою Volume Segmenter. Цей сегмент демонструє основні характеристики вогню.

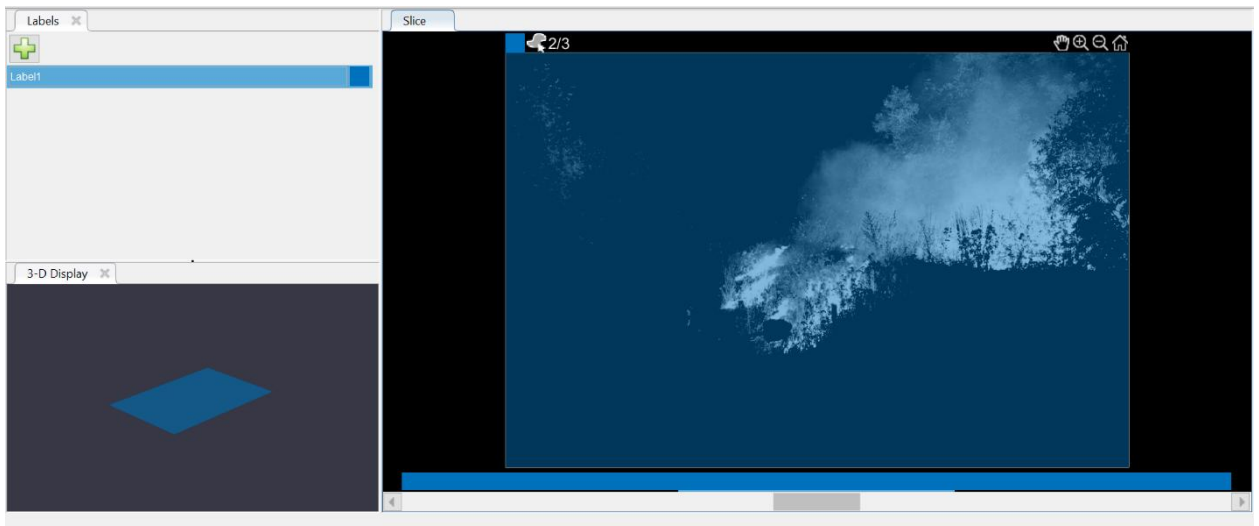


Рис. 3.8. Сегмент 2

Другий сегмент (рис. 3.8), який показує інші аспекти вогняної зони, що можуть бути важливими для подальшого аналізу.

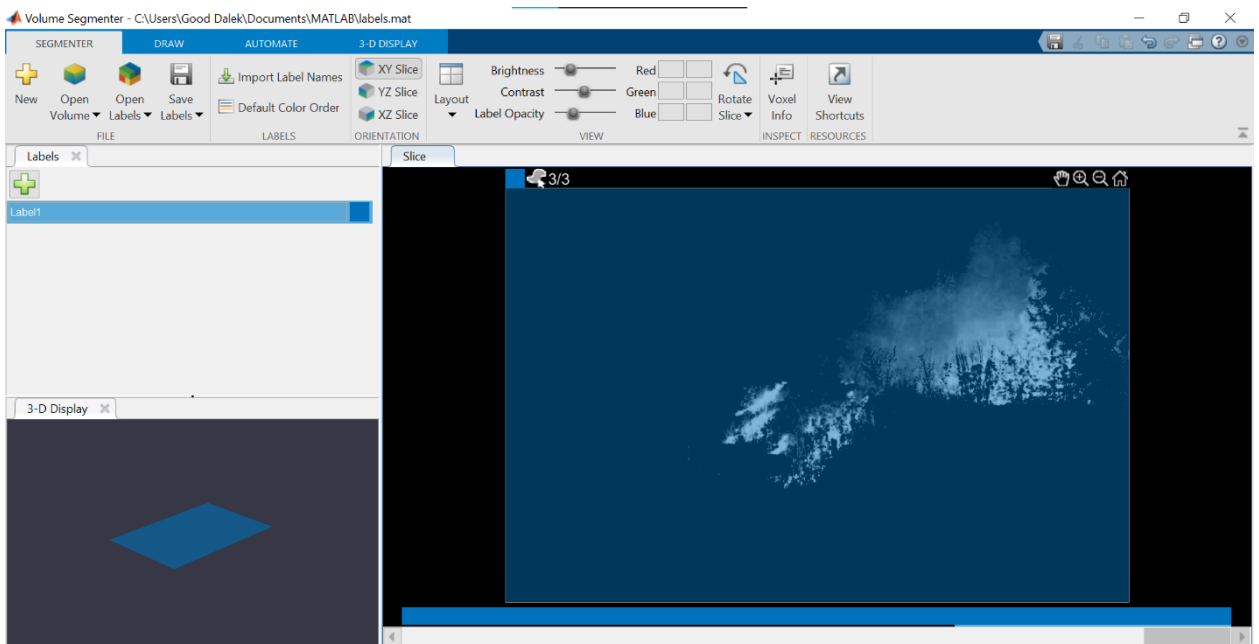


Рис. 3.9. Сегмент 3

Третій сегмент (рис. 3.9), що ілюструє ще один рівень деталізації тривимірної зони вогню, що дозволяє краще зрозуміти його структуру та поведінку.

З використанням інструменту Color Thresholder було проведено сегментацію кольорового зображення вогню шляхом встановлення порогових

значень для колірних каналів, що базуються на різних колірних просторах. Сегментація кольорового зображення вогню за допомогою інструменту Color Thresholder (рис. 3.10) є ефективним методом, що дозволяє виділити ключові елементи зображення для подальшого аналізу.

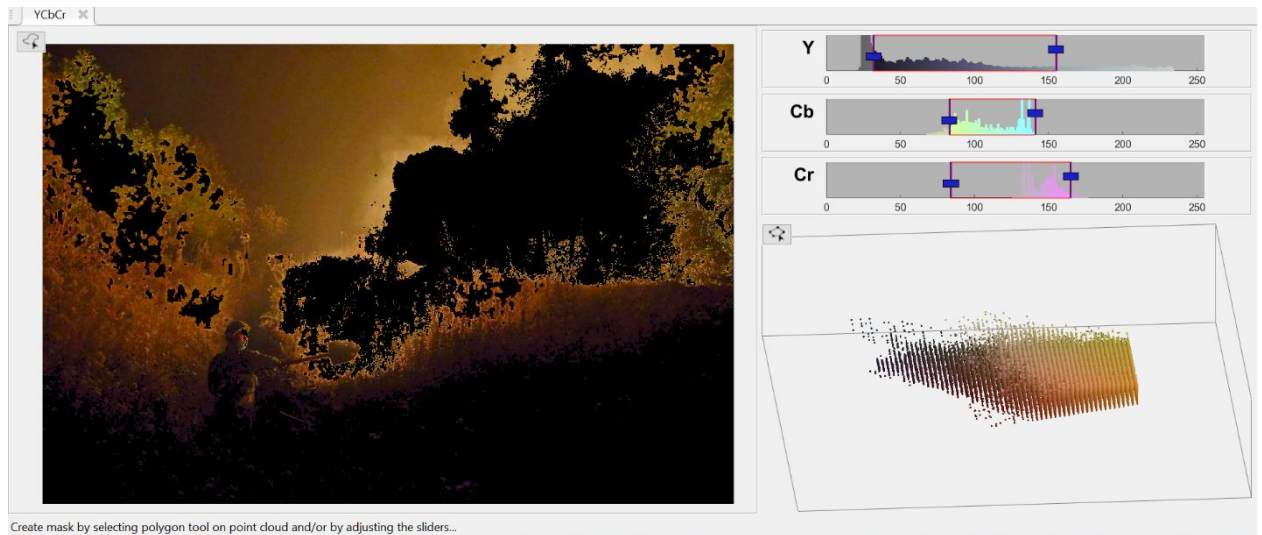


Рис. 3.10. Створення маски та виділення окремої зони вогню на зображенні

Для розробки нейронної мережі, призначеної для аналізу зображень, що відображають розповсюдження вогню, було обрано метод зважених спряжених градієнтів (scaled conjugate gradient). Цей метод був вибраний через його ефективність у використанні системної пам'яті, а також здатність адаптуватися до різноманітних даних, постійно вдосконалюючи свої характеристики.

Навчання моделі передбачає виконання кількох ітерацій, що дозволяє нейронній мережі досягти максимальної точності в своїх прогнозах. Метод спряженого градієнта часто реалізується як ітераційний алгоритм, що підходить для розв'язання розріджених систем, які є занадто великими для обробки за допомогою прямих методів.

SCG-алгоритм не виконує пошук по лінії на кожній ітерації на відміну від інших алгоритмів спряженого градієнта, які вимагають пошук на кожній ітерації. SCG дозволяє уникнути важких операцій [11]. Функція мережевого навчання даного алгоритму оновлює вагу та значення зміщення за методом

масштабованого спряженого градієнта [11]. Розмір кроку оцінюється за допомогою іншого підходу. Крок другого порядку обчислюється наступним чином [57]:

$$\underline{s}_k = \frac{E'(\underline{\omega}_k + \sigma_k \underline{p}_k) - E'(\underline{\omega}_k)}{\sigma_k} + \lambda_k \underline{p}_k, \quad (3.4)$$

де λ_k – це скалярна величина, яка з кожним кроком збільшується.

Розмір одного кроку розраховується за наступною формулою [57]:

$$\alpha_k = \frac{\mu_k}{\delta_k} = \frac{-\underline{p}_j^T E'_{q\omega}(\underline{y}_1)}{\underline{p}_j^T E''(\omega) \underline{p}_j}, \quad (3.5)$$

де $\underline{\omega}$ – це ваговий коефіцієнт у просторі, $E'(\underline{\omega}_k)$ позначає градієнт помилки, $E'_{q\omega}(\underline{y}_1)$ є позначенням квадратичного наближення функції помилки, $\underline{p}_j$ позначає набір ненульових вагових векторів. λ_k оновлюється з кожним кроком за формулою [57]:

$$\underline{\lambda}_k = 2(\lambda_k - \frac{\delta_k}{|\underline{p}_k|^2}), \quad (3.6)$$

Навчання припиняється, коли виникає будь-яка з таких умов:

- досягнуто максимальної кількості епох;
- перевищено максимальний час;
- досягнуто необхідного результату;
- градієнт продуктивності падає нижче мінімального значення;
- показник перевірки збільшився більше, ніж максимальний час помилок з моменту останньої ітерації;
- час його зменшення (при використанні перевірки).

Результати представлені на рисунках нижче.

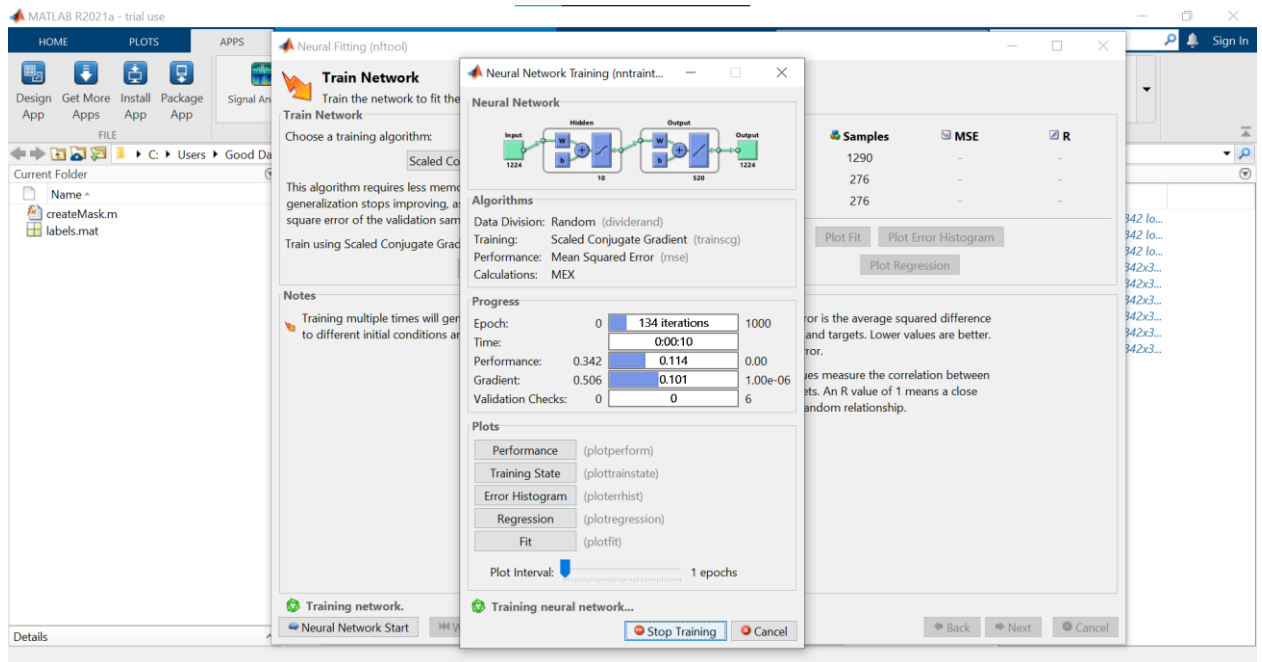


Рис. 3.11. Алгоритм навчання нейронної мережі

Проілюстровано (рис. 3.11) загальний алгоритм навчання нейронної мережі, включаючи етапи підготовки даних, обчислення виходу, оцінки помилок та оновлення ваг.

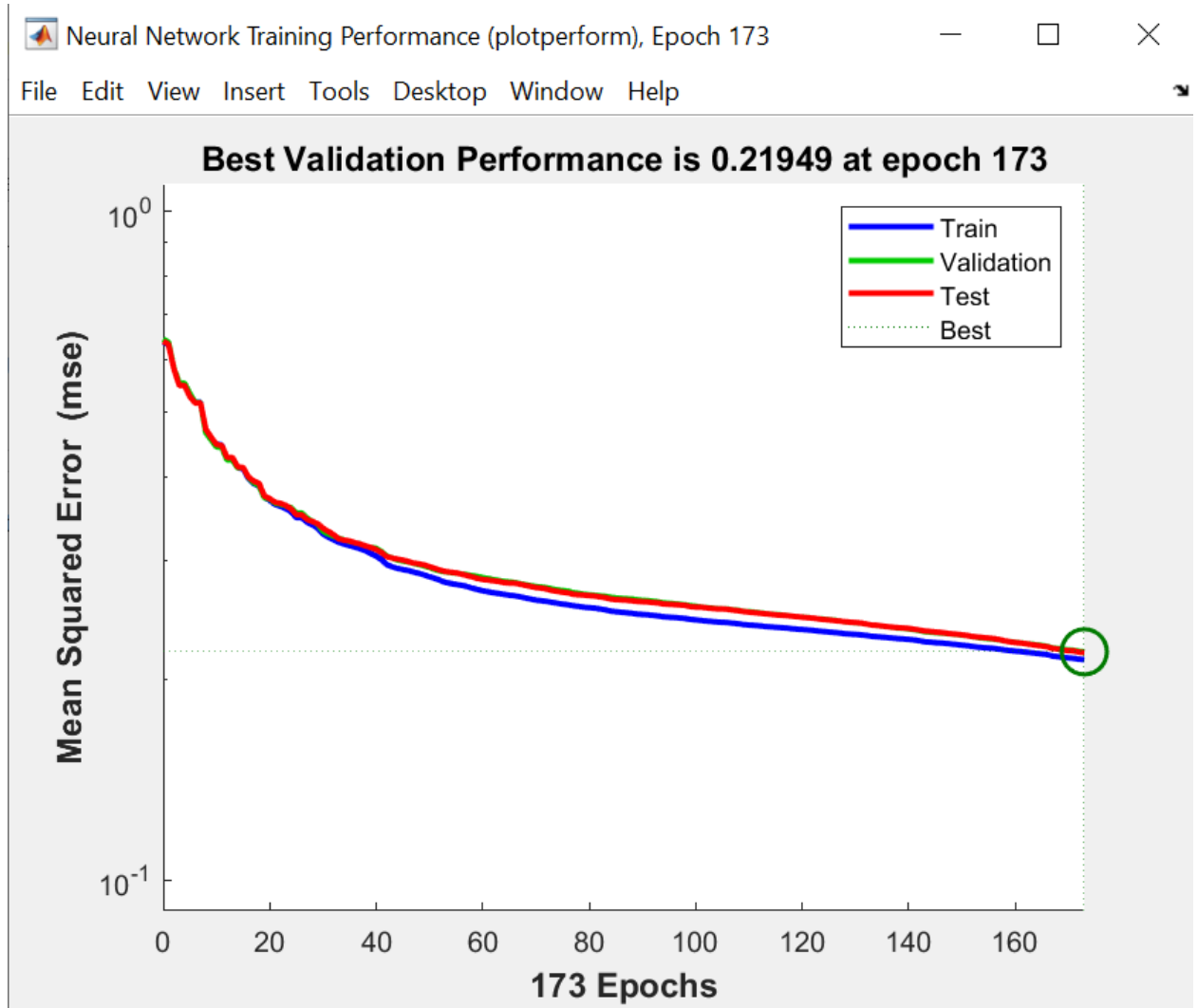


Рис. 3.12. Графік виконання алгоритму навчання

Графік (рис. 3.12) демонструє процес виконання алгоритму навчання, показуючи, як змінюється продуктивність моделі з часом або з кількістю епох.

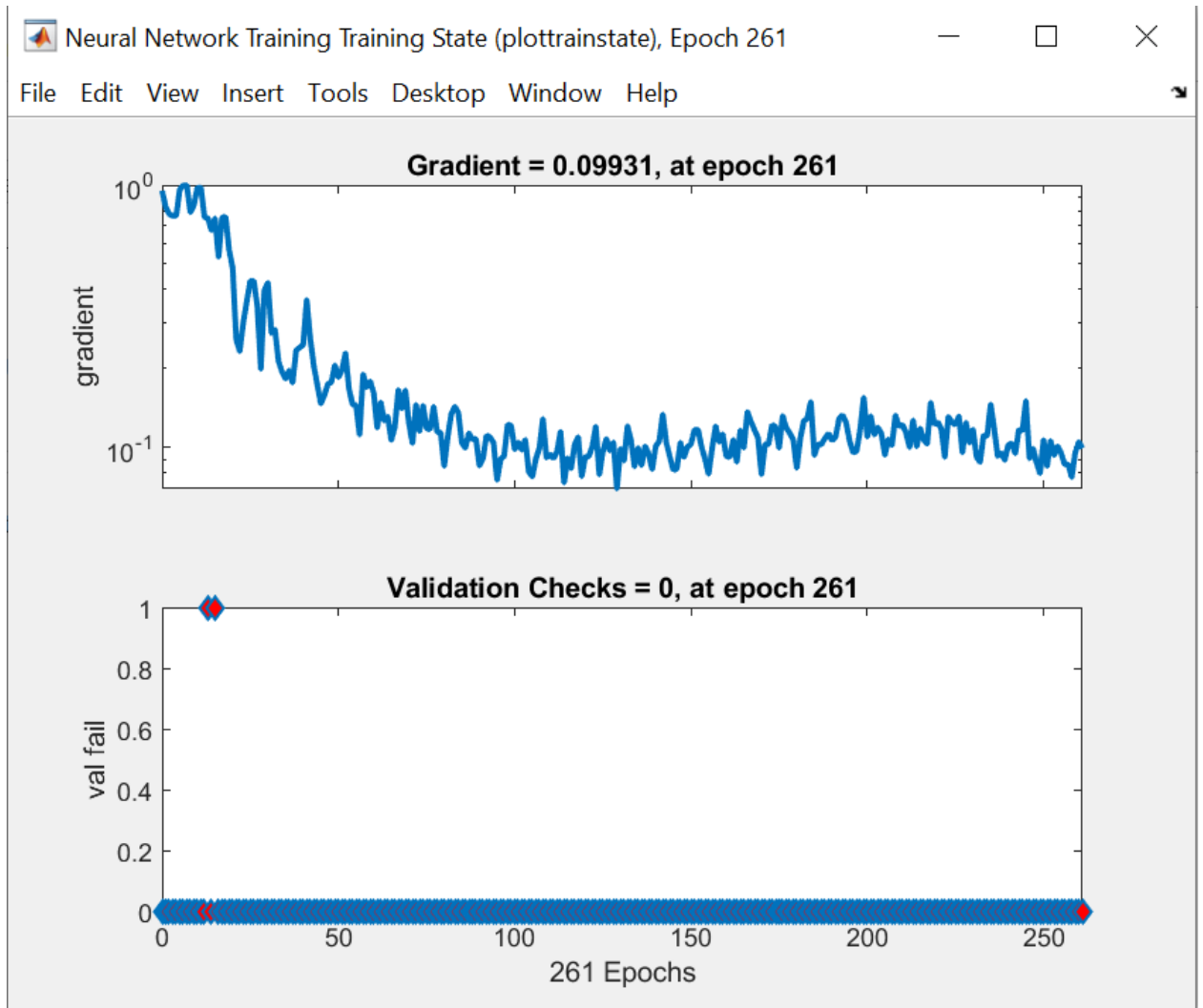


Рис. 3.13. Графік стану навчання нейронної мережі

Графік (рис. 3.13) відображає стан навчання нейронної мережі, включаючи показники, такі як точність та втрати, що дозволяє оцінити ефективність навчання.

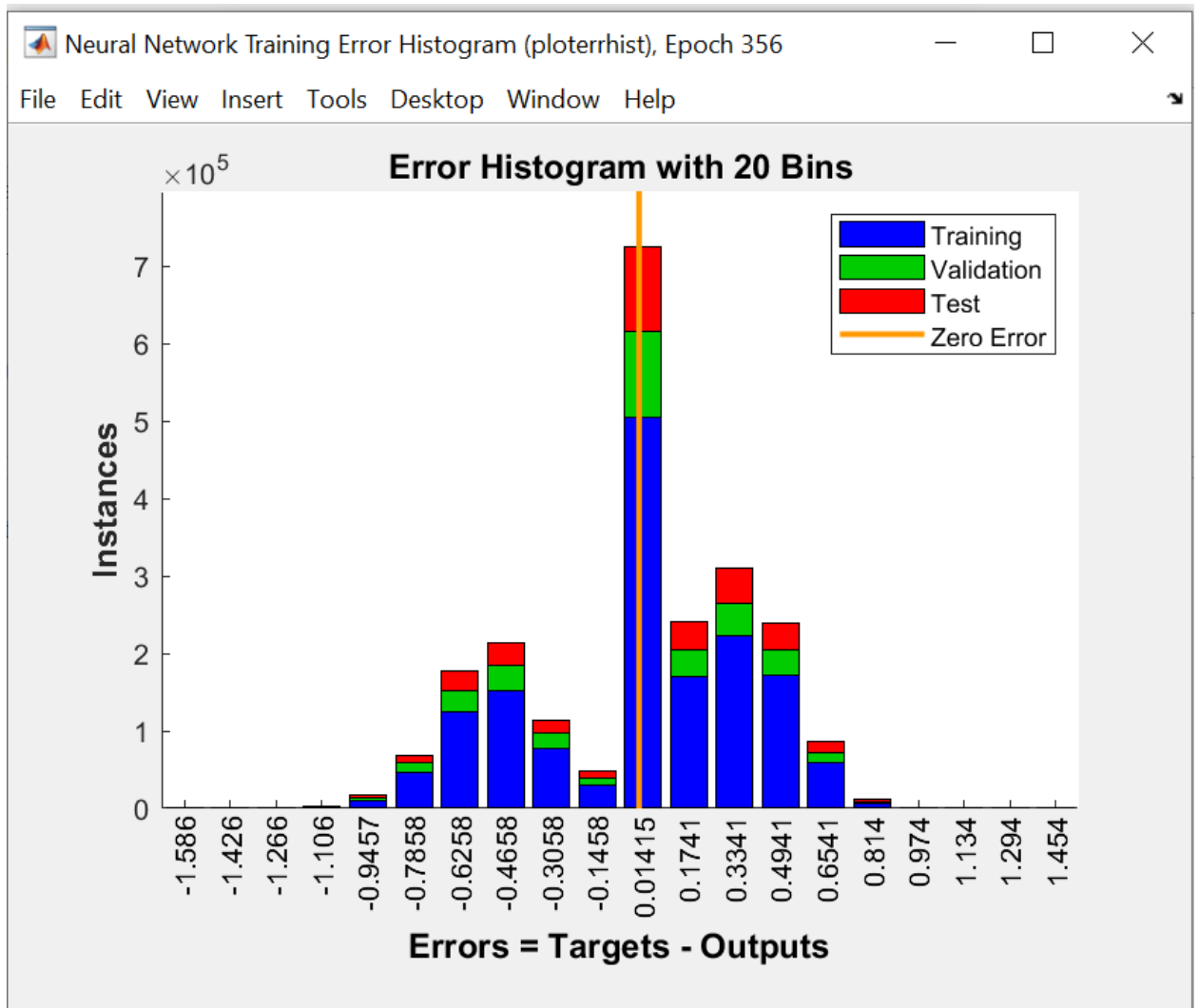


Рис. 3.14. Графік помилок

Графік (рис. 3.14) ілюструє зміни в помилках під час навчання, що допомагає виявити, чи зменшуються помилки з часом, що є ознакою успішного навчання.

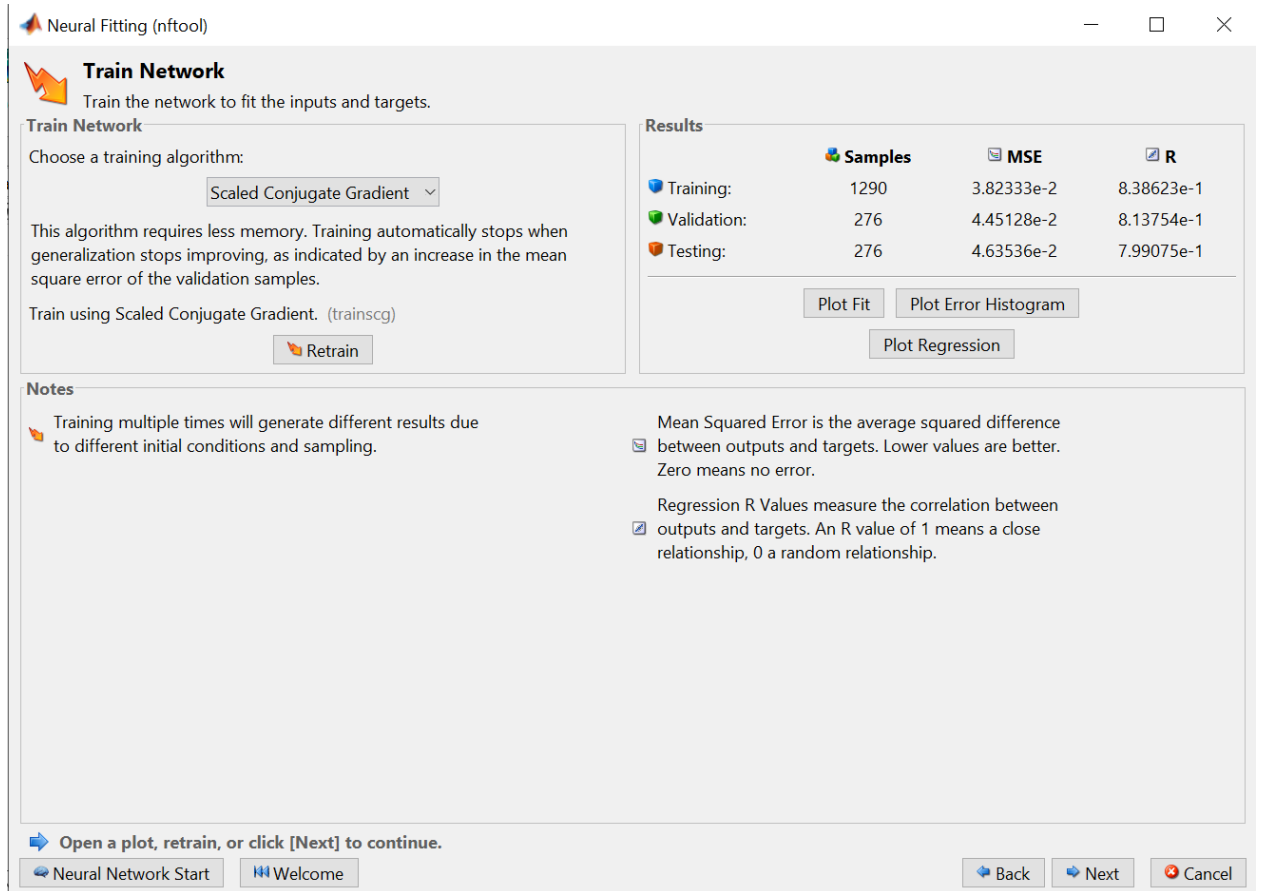


Рис. 3.15. Навчання нейронної мережі

Процес навчання нейронної мережі (рис. 3.15) містить візуалізацію даних, які використовуються для навчання, та результати, які вона генерує.

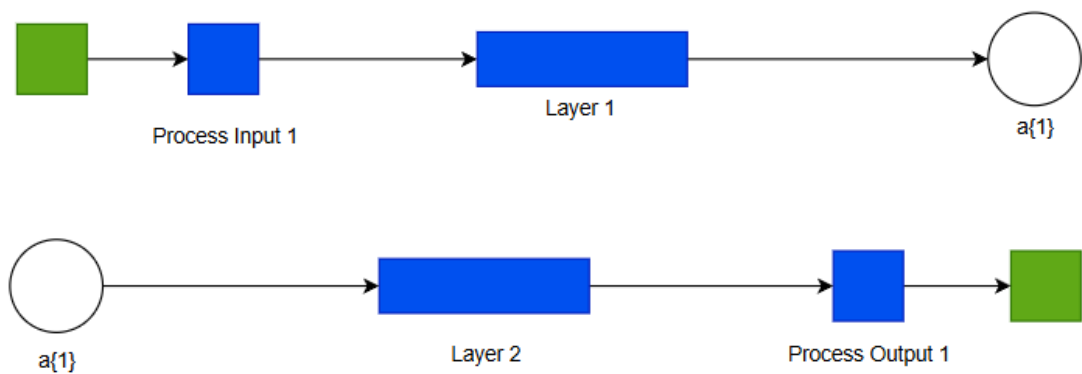


Рис. 3.16. Функція нейронної мережі (графік)

Графік функції нейронної мережі (рис. 3.16) демонструє, як нейронна мережа обробляє вхідні дані та генерує вихід, що є важливим для розуміння її роботи.

3.2 Дослідження методів відновлення та реконструкції зображень: класифікація та моделювання

Більшість методів покращення зображень можна класифікувати на дві основні категорії: методи обробки в просторовій області та частотні методи.

Під терміном «просторова область» розуміється площа зображення, що передбачає обробку даних безпосередньо на рівні пікселів зображення [1].

Основним принципом просторових методів є виконання операцій безпосередньо над пікселями зображення, що дозволяє коригувати його характеристики і покращувати якість. Основне рівняння, яке описує процес просторової обробки, є [23, 34]:

$$g(x, y) = T[f(x, y)], \quad (3.7)$$

де $f(x, y)$ представляє вихідне зображення, $g(x, y)$ – оброблене зображення, а T – оператор, що здійснює перетворення f . У даній моделі оператор використовується для виконання різноманітних операцій над вихідним зображенням з метою досягнення визначених цілей. Основний підхід до визначення околиці навколо точки (x, y) полягає у використанні квадратної або прямокутної області, що охоплює пікселі, центровані в цій точці. Оператор T виконує обробку для кожної точки в координатах (x, y) , в результаті чого отримується значення g для відповідної точки. Лінійні перетворення здійснюються одночасно над значеннями пікселів у межах околиці, а також над значеннями певної матриці, що має такі ж розміри, як і околиця.

Метод «сіль і перець» є одним із підходів до обробки зображень, що містять імпульсний шум. Таке зображення характеризується наявністю темних пікселів у яскравих областях та яскравих пікселів у темних зонах. На рисунку 3.18 представлено вихідне зображення, а рисунки 3.19 та 3.20 демонструють результати застосованої фільтрації [34, 35].

Нехай $I(x,y)$ – це вихідне зображення, де (x,y) – координати пікселя. Гіркий шум можна описати таким чином:

Для усунення шуму важливо спочатку відняти темний кадр. Нехай $D(x,y)$ – це темний кадр, тоді нове зображення $I''(x,y)$ можна отримати за формулою [35]:

$$I''(x,y)=I(x,y)-D(x,y) \quad (3.8)$$

Після віднімання темного кадру, для корекції значень пікселів, які були втрачені через шум, застосовується інтерполяція. Нехай N – це набір сусідніх пікселів, тоді нове значення пікселя $I_{\text{new}}(x,y)$ можна обчислити як середнє значення сусідніх пікселів [35]:

$$I_{\text{new}}(x,y) = \frac{1}{|N|} \sum_{(x',y') \in N} I''(x',y'), \quad (3.9)$$

де $|N|$ – кількість сусідніх пікселів, які беруть участь в інтерполяції.

Після виконання інтерполяції, фінальне зображення $I_{\text{final}}(x,y)$ можна отримати, об'єднавши відкориговані пікселі.

У результаті виконання вищезазначених кроків, метод солі і перцю дозволяє значно зменшити вплив гіркого шуму на оброблене зображення, покращуючи його якість для подальшого аналізу.



Рис. 3.17. Вихідне зображення

На початковому зображенні (рис. 3.17) можуть бути видимі артефакти шуму, які потребують усунення.



Рис. 3.18. Використання фільтру по Гауссу

Фільтр (рис. 3.18) допомагає зменшити вплив шуму, зберігаючи при цьому важливі деталі зображення. Метод фільтрації по Гауссу є ще одним важливим підходом до обробки зображень.



Рис. 3.19. Відфільтроване зображення за допомогою методу солі та перцю

Відфільтроване зображення (рис. 3.19) демонструє значне покращення якості, з більш чіткими деталями та меншим рівнем шуму. «Метод солі і перцю» є ефективним способом усунення гіркої шуму з зображень. Використовуючи віднімання темного кадру та інтерполяцію, можна значно покращити якість зображення, зменшуючи вплив шуму на подальший аналіз.

Метод фільтрації за Гауссом є важливим інструментом у обробці зображень, що дозволяє зменшити шум, який часто виникає в процесі зйомки. Стандартна модель шуму підсилювача характеризується адитивним впливом на кожен піксель, незалежно від інтенсивності сигналу. У кольорових камерах,

де посилення каналу синього кольору є вищим, ніж у зеленому або червоному каналі, спостерігається тенденція до збільшення шуму саме в синьому каналі.

Метод фільтрації по Гауссу є ефективним способом зменшення шуму в зображеннях, зокрема адитивного шуму підсилювача, який може виникати в процесі зчитування даних з датчика зображення. Цей тип шуму є постійним і може бути особливо помітним у темних областях зображення, де сигнал є слабким.

Нехай $I(x,y)$ – це вихідне зображення, яке містить шум. Шум підсилювача можна описати як адитивний шум, що додається до значення пікселя [35, 36]:

$$I_{\text{noisy}}(x,y)=I(x,y)+N(x,y), \quad (3.10)$$

де $N(x,y)$ – це адитивний шум, який має нормальний розподіл з нульовим середнім значенням і стандартним відхиленням σ [35, 36]:

$$N(x,y) \sim N(0,\sigma^2) \quad (3.11)$$

Ця модель показує, що значення пікселя в зображенні з шумами є сумою справжнього значення пікселя та випадкового шуму, що має нормальний розподіл.

Гаусів фільтр використовується для згладжування зображення та зменшення шуму. Ядро Гаусса визначається як [23]:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (3.12)$$

де σ – стандартне відхилення, яке контролює розмір фільтра. Чим більше значення σ , тим ширше буде ядро фільтра, що призводить до більшого згладжування зображення.

Для отримання відфільтрованого зображення $I_{filtered}(x,y)$ використовується згортка вихідного зображення з ядром Гаусса [34]:

$$I_{filtered}(x,y) = \sum_m^k = -k \sum_n^n - kG(m,n) * I_{noisy}(x+m,y+n), \quad (3.13)$$

де k – радіус фільтра, що визначає область навколо пікселя, яка буде враховуватися при згортці. Значення пікселя в відфільтрованому зображенні є зваженою сумою значень пікселів у сусідніх областях, де ваги визначаються ядром Гаусса.

Після застосування Гаусівського фільтра, фінальне зображення $I_{final}(x,y)$ буде виглядати наступним чином [34]:

$$I_{final}(x,y) = I_{filtered}(x,y) \quad (3.14)$$

Фінальне зображення є результатом обробки вихідного зображення з урахуванням шуму, що дозволяє зменшити вплив адитивного шуму підсилювача.



Рис. 3.20. Перетворення вихідного зображення до сірого спектру

Перетворення до сірого спектру є важливим кроком для спрощення аналізу (рис. 3.20), оскільки воно зменшує кількість інформації, з якою потрібно працювати, зберігаючи при цьому основні деталі.



Рис. 3.21. Додавання гаусівського шуму до зображення

До сірого зображення додано гаусівський шум (рис. 3.21). Гаусівський шум може суттєво погіршити якість зображення, ускладнюючи подальший аналіз.

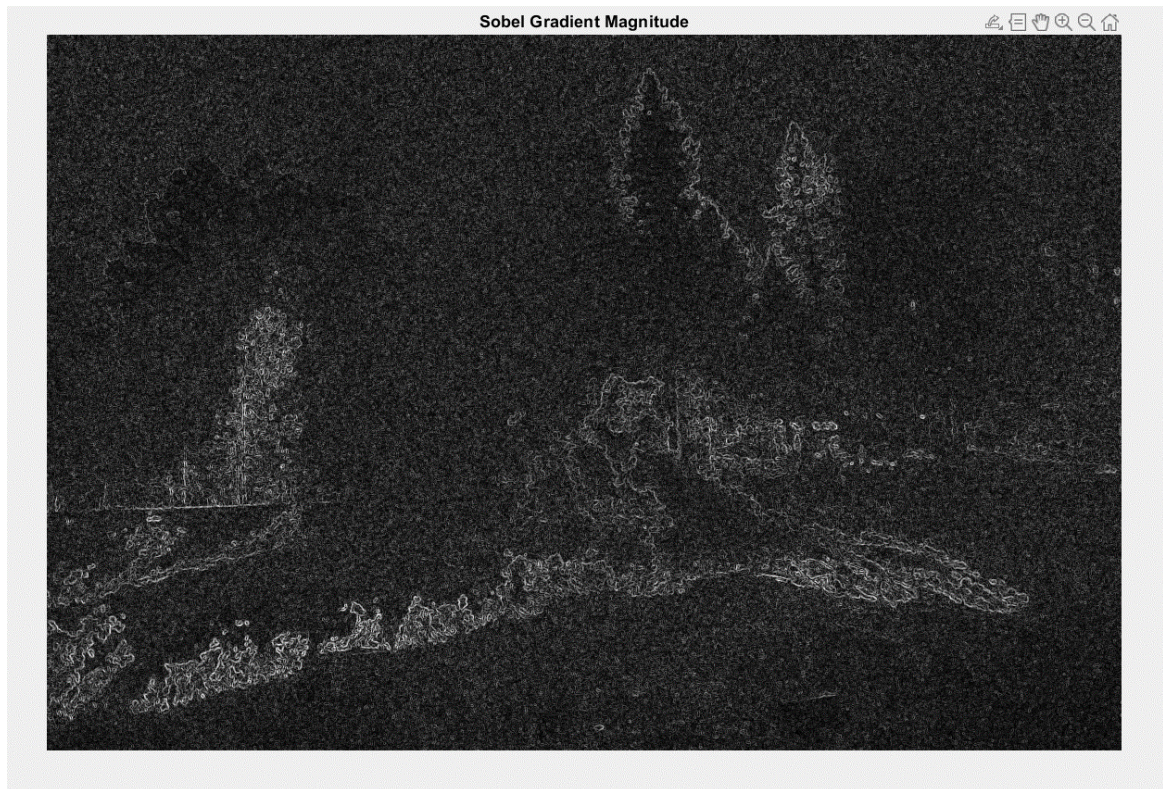


Рис. 3.22. Зображення з виділеним шумом

Виділені області зображення (рис. 3.22) показують, як шум заважає чіткості та деталізації зображення.

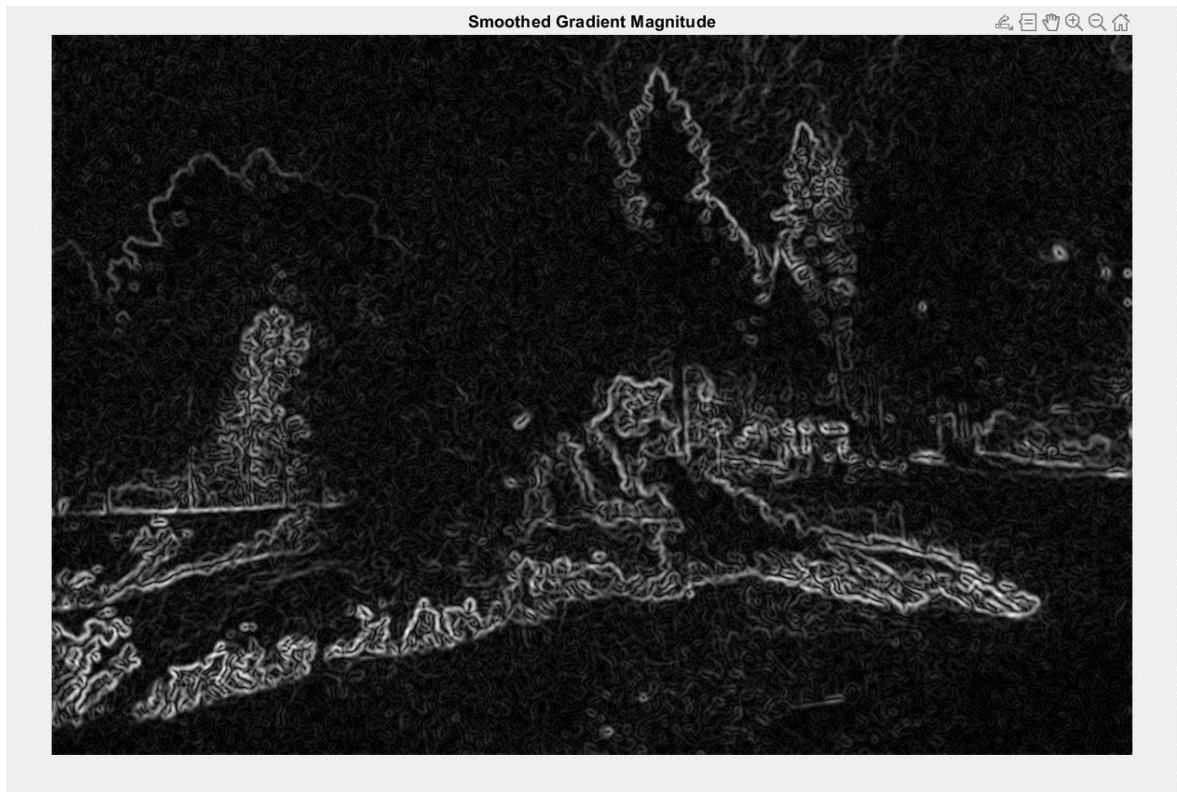


Рис. 3.23. Згладжене зображення

Згладжене зображення (рис. 3.23) демонструє покращену якість, що робить його більш придатним для подальшого аналізу.

Використання методу Спекл є важливим аспектом у обробці зображень, особливо в контексті активного радіолокаційного зображення, зокрема в системах з синтетичною апертурою (SAR) [40-43].

Нехай $I(x,y)$ – це вихідне зображення, яке містить шум Спекл. Шум можна описати як адитивний шум, що впливає на значення пікселя [34, 35]:

$$I_{\text{noisy}}(x,y)=I(x,y)+S(x,y), \quad (3.15)$$

де $S(x,y)$ – це шум Спекл, який може бути описаний як:

$$S(x,y)=I(x,y) \cdot N(x,y), \quad (3.16)$$

де $N(x,y)$ – випадковий шум, що має нормальний розподіл з нульовим середнім значенням і стандартним відхиленням σ :

$$N(x,y) \sim N(0, \sigma^2) \quad (3.17)$$

Шум Спекл пропорційний значенню пікселя в оригінальному зображенні, що робить його особливо помітним у темних областях, де сигнал є слабким [34-36].

Для зменшення шуму Спекл часто використовують фільтрацію. Одним із методів є використання фільтрації на основі середнього значення або медіанної фільтрації. Нехай $I_{\text{filtered}}(x,y)$ – це відфільтроване зображення, яке можна отримати за допомогою медіанної фільтрації [34-36]:

$$I_{\text{filtered}}(x,y) = \text{median}(I_{\text{noisy}}(x',y') \mid (x',y') \in N(x,y)), \quad (3.18)$$

де $N(x,y)$ – це область навколо пікселя (x,y) . Медіанна фільтрація обчислює медіану значень пікселів у цій області, що дозволяє ефективно усувати шум, не розмиваючи зображення.

Іншим методом є застосування Гаусівського фільтра, який згладжує зображення [34-36]:

$$I_{\text{filtered}}(x,y) = \sum_m^k = -kG(m,n) * I_{\text{noisy}}(x+m,y+n), \quad (3.19)$$

де $G(m,n)$ – ядро Гаусса, а k – радіус фільтра.

Значення пікселя в відфільтрованому зображенні є зваженою сумою значень пікселів у сусідніх областях, де ваги визначаються ядром Гаусса.

Після застосування фільтрації, фінальне зображення $I_{\text{final}}(x,y)$ можна отримати [39]:

$$I_{\text{final}}(x,y)=I_{\text{filtered}}(x,y)$$

(3.20)

Фінальне зображення є результатом обробки вихідного зображення з урахуванням шуму Спекл, що дозволяє зменшити його вплив на якість зображення.



Рис. 3.24. Додавання шуму

До отриманого зображення було додано шум Спекл (рис. 3.24). Це ілюструє, як шум може заважати чіткості та деталізації зображення, ускладнюючи подальший аналіз.



Рис. 3.25. Відфільтроване зображення

Відфільтроване зображення (рис. 3.25) має значно покращену якість, що дозволяє легше інтерпретувати дані та проводити аналіз.

Побудова графіків для порівняння методів фільтрації зображень є важливим етапом у візуалізації результатів обробки. Для демонстрації різниці між різними методами фільтрації були використані контурні графіки. На вихідному зображенні (рис. 3.26) спостерігається високий рівень стороннього шуму, що може негативно вплинути на подальший аналіз та оцінку даних. У порівнянні з вихідним зображенням (рис. 3.27), рівень шуму значно знижений, що свідчить про ефективність застосування цього методу.

Графік очищеного від шуму зображення (рис. 3.28) підтверджує ефективність даного підходу та демонструє зменшену кількість шуму на обраній площі.

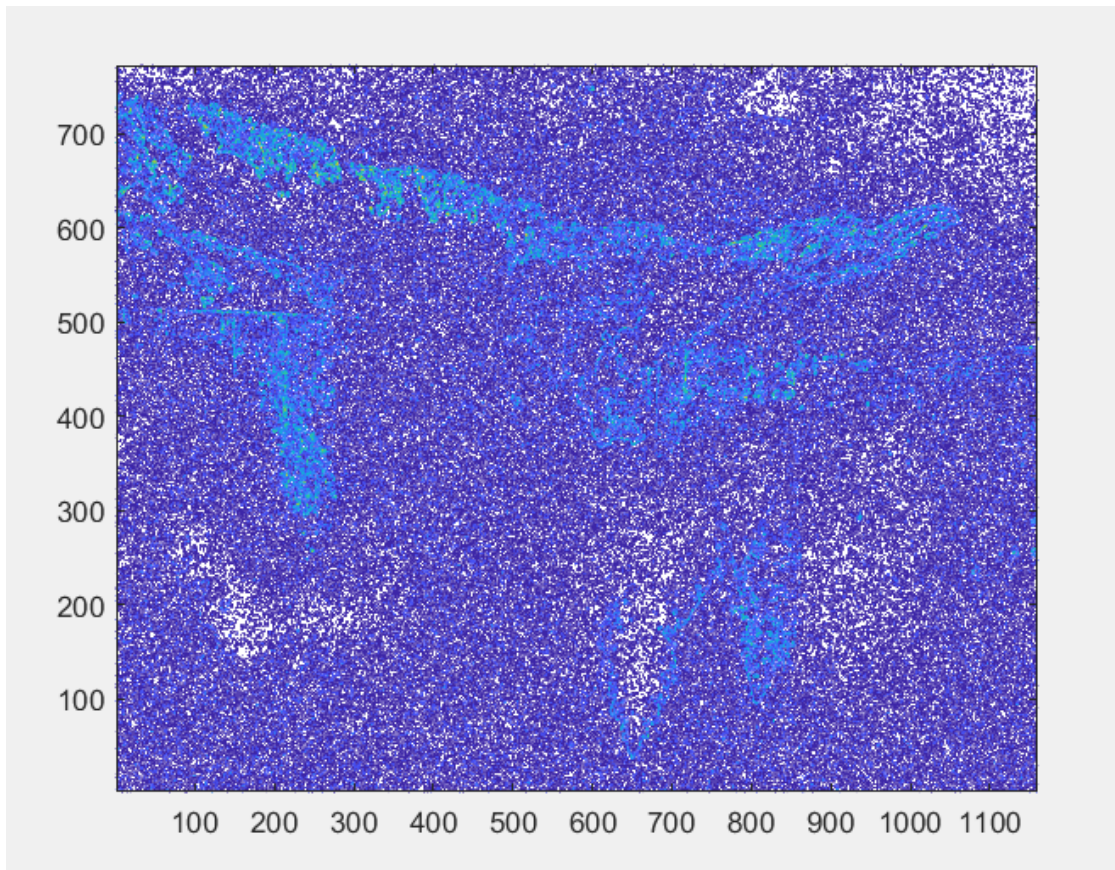


Рис. 3.26. Шум з необробленого зображення

Яскраві області на графіку вказують на високий рівень стороннього шуму, що присутній у вихідному зображенні. Це підкреслює, що наявність шуму може суттєво погіршити подальший аналіз та оцінку даних, оскільки він заважає чіткості та деталізації.

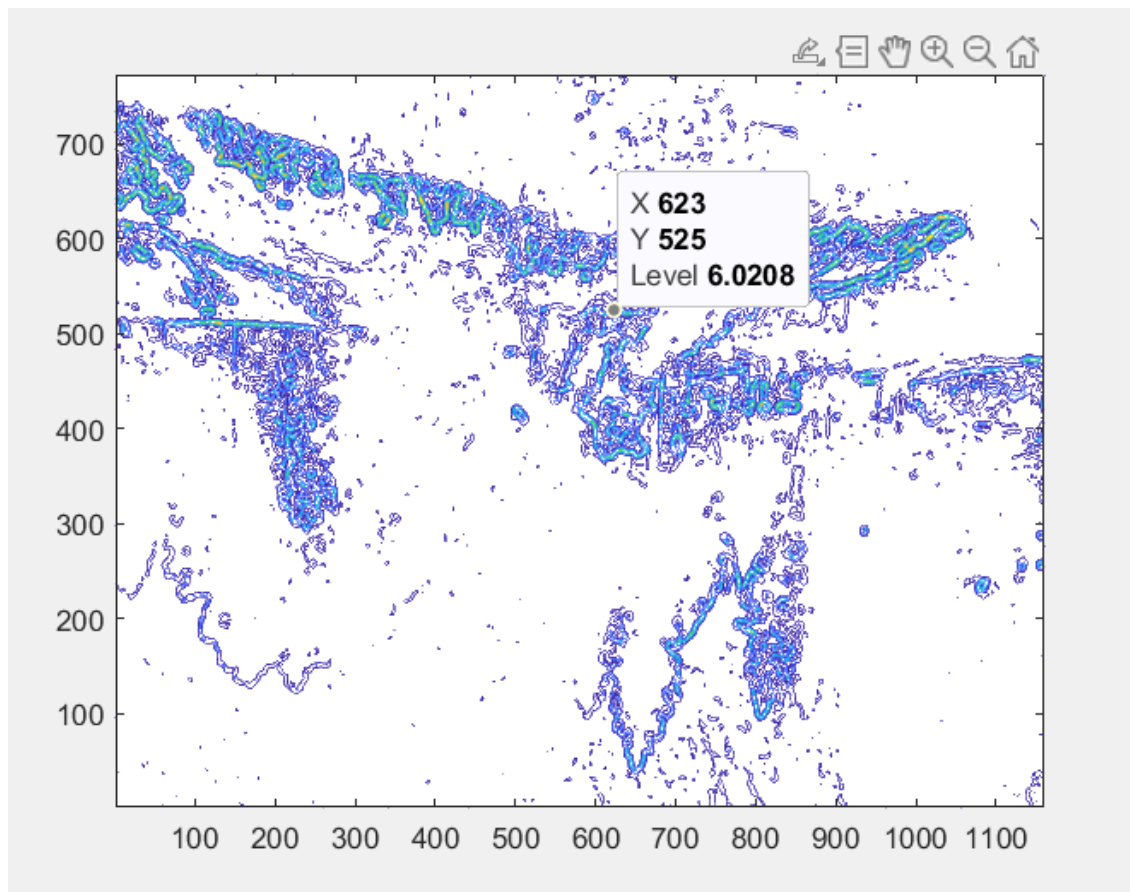


Рис. 3.27. Шум обробленого зображення за допомогою гаусівського методу

Порівняно з вихідним зображенням, рівень шуму значно зменшився. Це свідчить про ефективність використання даного методу, оскільки він дозволяє зменшити вплив шуму на якість зображення, що робить його більш придатним для подальшого аналізу.

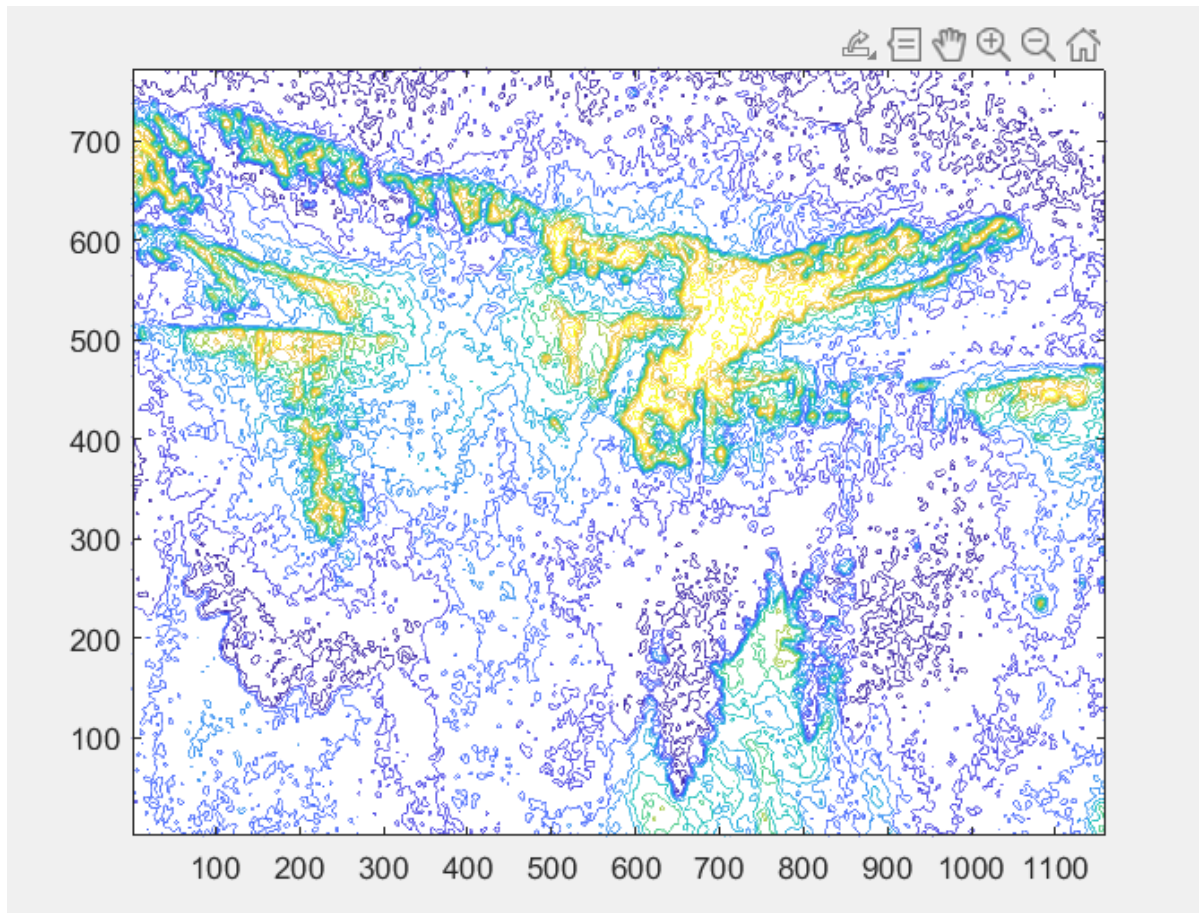


Рис. 3.28. Шум згладженого зображення

Згладжене зображення має значно зменшену кількість шуму на обраній площі, що підтверджує ефективність застосованого методу. Це зображення показує, як фільтрація може покращити якість даних, роблячи їх більш чіткими та легшими для інтерпретації.

3.3 Обґрунтування використовуваних програмних засобів для моделювання комп'ютерної системи

Для реалізації методів обробки зображень та аналізу даних використовувалися різноманітні програмні засоби, які забезпечують ефективність та точність виконання поставлених задач. Основними програмними засобами є:

Розглянуто методи відновлення та реконструкції зображень, що є однією з ключових проблем у дослідженні та фіксації надзвичайних природних явищ, зокрема пожеж [44]. Ця проблема є актуальною, оскільки для аналізу та проведення розрахунків використовуються супутникові знімки [45], які не завжди зберігають свою якість та чіткість, необхідні для однозначної класифікації ситуації.

Для оцінки якості зображень та точності їх відображення застосовуються різноманітні методи. Важливість вибору оптимальних рішень полягає у зменшенні ймовірності помилок та підвищенні результативності роботи систем підтримки прийняття рішень. Дисертація спрямована на вивчення та реалізацію методів, що використовуються для видалення шумів та інших дефектів із зображень з метою подальшого аналізу та оптимізації.

Акцентується увага на застосуванні методів машинного навчання та комп'ютерного зору [31, 100-103], які дають можливість точно ідентифікувати та попереджати пожежі, а також інші негативні природні фактори. Для реалізації поставлених задач розглянуто методи обробки зображень у просторовій області та частотні методи. Просторовий метод передбачає аналіз та обробку окремих пікселів на зображенні, що дозволяє детально вивчити їх характеристики [91-95].

Особливу увагу в дисертації приділено методам обробки супутникових зображень для видалення шуму та інших дефектів [45]. Використано такі способи, як фільтрація «сіль і перець», Гаусова фільтрація та Спекл-фільтрація. У дослідженні наведено порівняння описаних методів та їх ефективності.

Проведено аналіз різноманітних методів обробки зображень, видалення шуму та підготовки їх до подальшого аналізу. Застосування методів, таких як віднімання темного кадру та інтерполяція темних і світлих пікселів, дозволяє усунути можливі дефекти, які заважають нормальному аналізу та фіксації пожеж [7, 71-75].

Для проведення дослідження використано безкоштовний пакет MATLAB, що містить всі необхідні інструменти для виконання розрахунків [5, 47].

MATLAB є потужним середовищем для чисельних обчислень, яке широко використовується в наукових дослідженнях та інженерії. У контексті даного дослідження MATLAB надає розширені можливості для обробки зображень, включаючи:

Інструменти для обробки зображень: MATLAB має вбудовану бібліотеку Image Processing Toolbox [21, 74], яка містить функції для фільтрації, сегментації, покращення якості зображень та видалення шумів.

Візуалізація даних: MATLAB забезпечує зручні засоби для візуалізації результатів обробки зображень, що дозволяє наочно оцінити ефективність застосовуваних методів.

Машинне навчання: MATLAB також містить інструменти для реалізації алгоритмів машинного навчання, що дозволяє інтегрувати ці методи в процес обробки зображень для покращення точності виявлення пожеж [9, 10].

Було використано безкоштовну версію MATLAB. Усі результати дослідження можна вважати достовірними [5, 100].

Для побудови моделі було використано Python. Розробка проводилась за допомогою PyCharm Community Edition.

Python є ще одним важливим інструментом, який використовувався для обробки зображень. Завдяки своїй простоті та потужним бібліотекам, Python став популярним вибором для наукових досліджень. Основні бібліотеки, які були використані:

OpenCV – це бібліотека, що забезпечує широкий спектр функцій для обробки зображень і комп'ютерного зору. OpenCV дозволяє реалізувати алгоритми для видалення шумів, обробки зображень у реальному часі та виконання складних маніпуляцій з зображеннями [19, 68-70].

scikit-image – це бібліотека для обробки зображень, яка надає простий інтерфейс для виконання різноманітних операцій, таких як фільтрація,

сегментація та трансформації зображень. Вона є особливо корисною для реалізації алгоритмів, пов'язаних із машинним навчанням [19].

Keras – це високорівнева бібліотека для створення нейронних мереж, яка спрощує процес розробки моделей машинного навчання. Вона надає простий і зрозумілий інтерфейс для роботи з різними типами нейронних мереж, що дозволяє швидко експериментувати з архітектурами. Keras підтримує різноманітні типи шарів, оптимізаторів та функцій активації, що робить її ідеальним вибором для дослідників і практиків [19, 71].

TensorFlow – це потужна бібліотека для машинного навчання, розроблена компанією Google. Вона забезпечує широкий спектр можливостей для створення та навчання нейронних мереж, а також для розгортання моделей у виробничих умовах. TensorFlow підтримує як високорівневі, так і низькорівневі API, що дозволяє користувачам обирати рівень абстракції, який найкраще відповідає їхнім потребам [19, 100].

PyTorch – це бібліотека для машинного навчання, розроблена компанією Meta AI, яка стала популярною завдяки своїй простоті та гнучкості. PyTorch дозволяє легко створювати динамічні обчислювальні графи, що робить її ідеальною для досліджень у сфері глибокого навчання. Бібліотека підтримує автоматичне диференціювання, що спрощує процес навчання моделей [19].

Simulink є додатком до MATLAB, який дозволяє моделювати, симулювати та аналізувати динамічні системи. У контексті даного дослідження Simulink може бути використаний для моделювання процесів обробки зображень та виявлення пожеж [47, 100], що дозволяє візуалізувати та оптимізувати алгоритми [10, 96-100].

Використання зазначених програмних засобів забезпечує комплексний підхід до обробки зображень та аналізу даних, що є критично важливим для успішного вирішення задач, пов'язаних із виявленням та моніторингом природних надзвичайних ситуацій, таких як лісові пожежі [9, 10, 33].

3.4 Висновки до третього розділу

У третьому розділі проведено всебічний аналіз методів обробки зображень, зокрема акцентовано увагу на техніках, що дозволяють ефективно видаляти шум. Розглянуті підходи, такі як метод солі і перцю, що базується на відніманні темного кадру та інтерполяції, продемонстрували свою ефективність у корекції артефактів, які виникають в зображеннях.

Досліджено природу адитивного шуму підсилювача, що впливає на кожен піксель зображення незалежно від інтенсивності сигналу, а також шум Спекл, котрий виникає в радарних зображеннях внаслідок випадкових коливань зворотного сигналу. Ці аспекти підкреслюють важливість правильного вибору методів обробки для досягнення високої якості зображень.

У процесі роботи були використані інструменти MATLAB, що дозволило провести детальну порівняльну характеристику методів видалення шуму. Отримані графіки наочно демонструють ефективність застосовуваних методів, що підкреслює їхню значущість для подальших досліджень.

Результати даного дослідження мають важливе значення для розвитку систем [33], які покращують аналіз зображень. Вони можуть стати основою для розробки рішень, що сприяють підвищенню ефективності реагування на надзвичайні ситуації, зокрема під час виникнення пожеж.

РОЗДІЛ 4

МОДЕЛЮВАННЯ ТА РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ

Реагування на природні катастрофи є складним і динамічним процесом, що вимагає оперативного прийняття рішень на основі аналізу даних та попереднього досвіду. Ефективність цих дій безпосередньо пов'язана зі здатністю систем підтримки прийняття рішень (СППР) адаптуватись до змінюваних умов, що виникають під час катастроф. У цьому контексті розробка комп'ютерних систем, які можуть обробляти та аналізувати великі обсяги супутникових зображень, набуває критичного значення [54, 100, 102-103]. Розроблена концептуальна модель програмної системи (рис. 4.1).

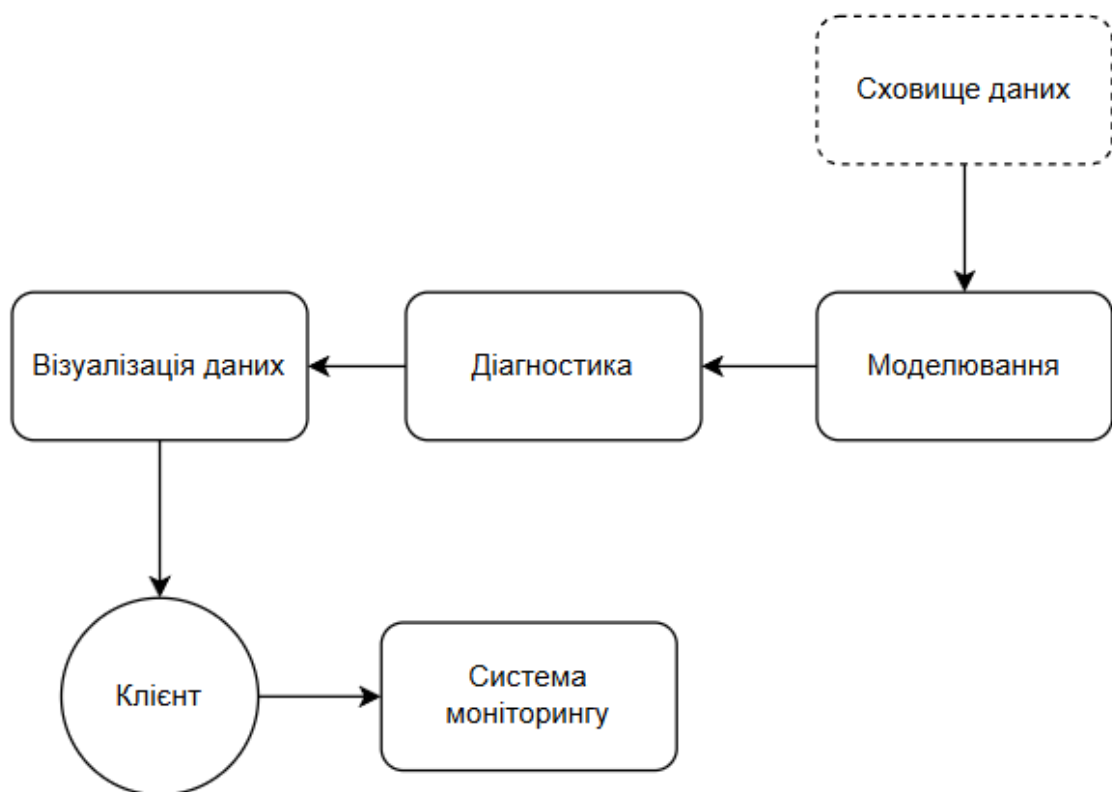


Рис. 4.1. Концептуальна модель програмної системи

Такі системи не лише забезпечують швидкий доступ до актуальної інформації, але й дозволяють здійснювати детальний аналіз ситуації, що склалася. Це, у свою чергу, сприяє більш точному прогнозуванню розвитку

подій та прийняттю обґрунтованих рішень, що можуть суттєво знизити негативні наслідки природних катастроф [53, 54].

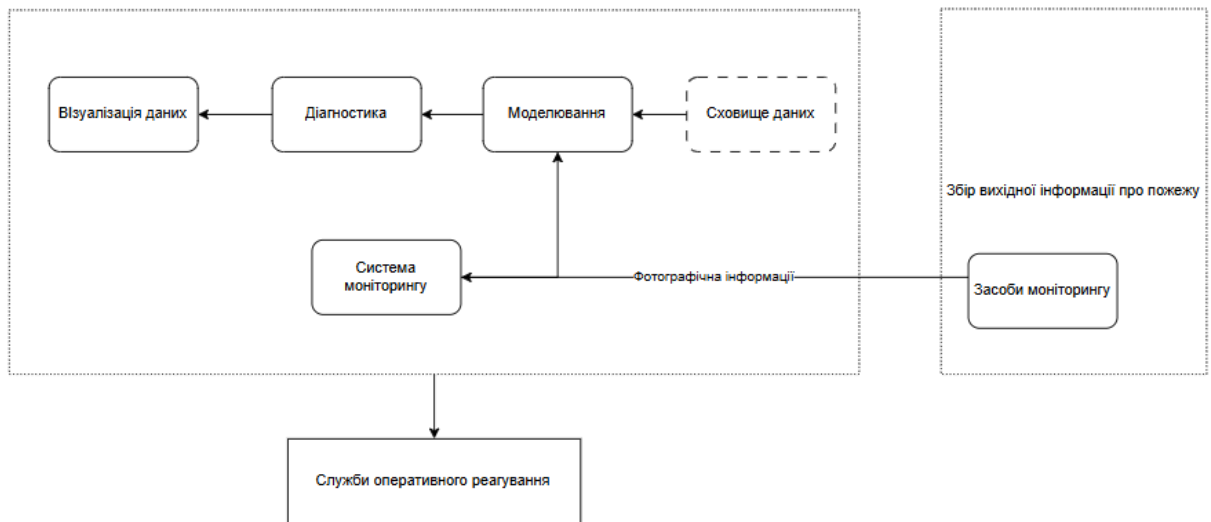


Рис. 4.2. Архітектура СППР

У цьому розділі буде представлено архітектуру та модель СППР (рис. 4.2), розроблену в рамках дослідження. Аналіз супутникових зображень [45], отриманих з відкритих джерел, дозволив створити згорткову нейронну мережу, яка продемонструвала високу точність у виявленні вогню на ранніх стадіях. Незважаючи на позитивні результати, існують можливості для подальшого удосконалення алгоритму, що відкриває нові горизонти для досліджень у цій галузі.

Дослідження базується на використанні потужних бібліотек Python, таких як Keras, TensorFlow та PyTorch, що забезпечують гнучкість і ефективність у розробці моделей машинного навчання [19]. Оцінка точності моделі на невідомому тестовому наборі даних, що становить 92%, підтверджує ефективність обраного підходу до виявлення вогневих патернів.

У розділі також буде розглянуто постановку задачі та основну мету дослідження, яка полягає у створенні високоточної моделі сегментації лісових пожеж на супутникових знімках. Це дослідження має на меті не лише покращення технологій виявлення, але й забезпечення надійних інструментів

для оперативного реагування на надзвичайні ситуації, що виникають внаслідок лісових пожеж [9].

4.1 Реалізація та опис методів обробки зображень на основі проведеного дослідження, технології та підходи

У цьому розділі розглянуто основні методи обробки зображень, які були використані в дослідженні, з відповідними формулами та описами [51, 55].

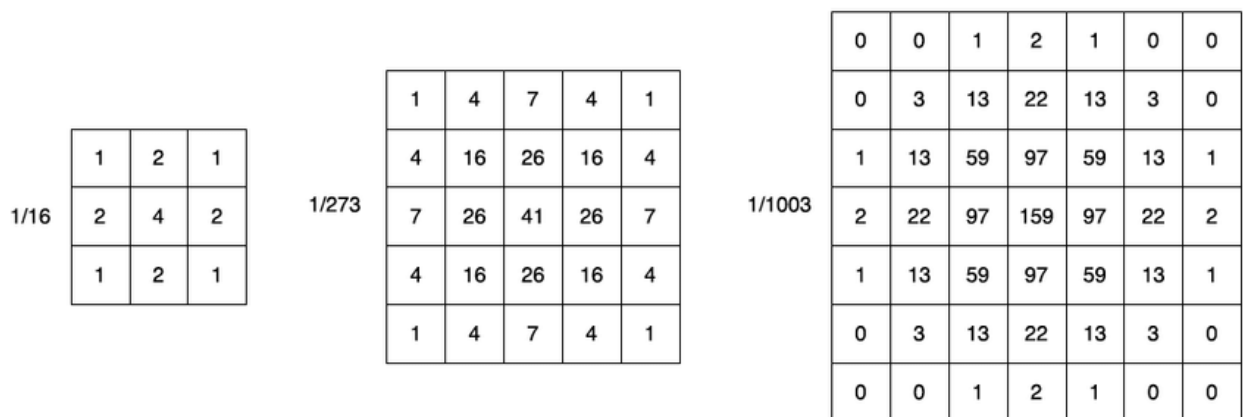


Рис. 4.3. Фільтрація шуму

Фільтрація шуму (рис. 4.3) є важливим етапом обробки зображень, що дозволяє зменшити небажані артефакти. Один із поширених методів – це середнє фільтрування. Формула для середнього фільтра виглядає так [7]:

$$I'(x, y) = \frac{1}{N} \sum_{(i, j) \in \omega} I(i, j), \quad (4.1)$$

де $I(x, y)$ – значення пікселя в координатах (x, y) , $I'(x, y)$ – значення пікселя після фільтрації, N – кількість пікселів у вікні фільтра, а ω – область навколо пікселя (x, y) .

Середнє фільтрування є простим, але ефективним методом для зменшення шуму в зображеннях. Використання формули для обчислення

середнього значення пікселів у локальній області дозволяє згладжувати зображення, зберігаючи при цьому основні структури [7].

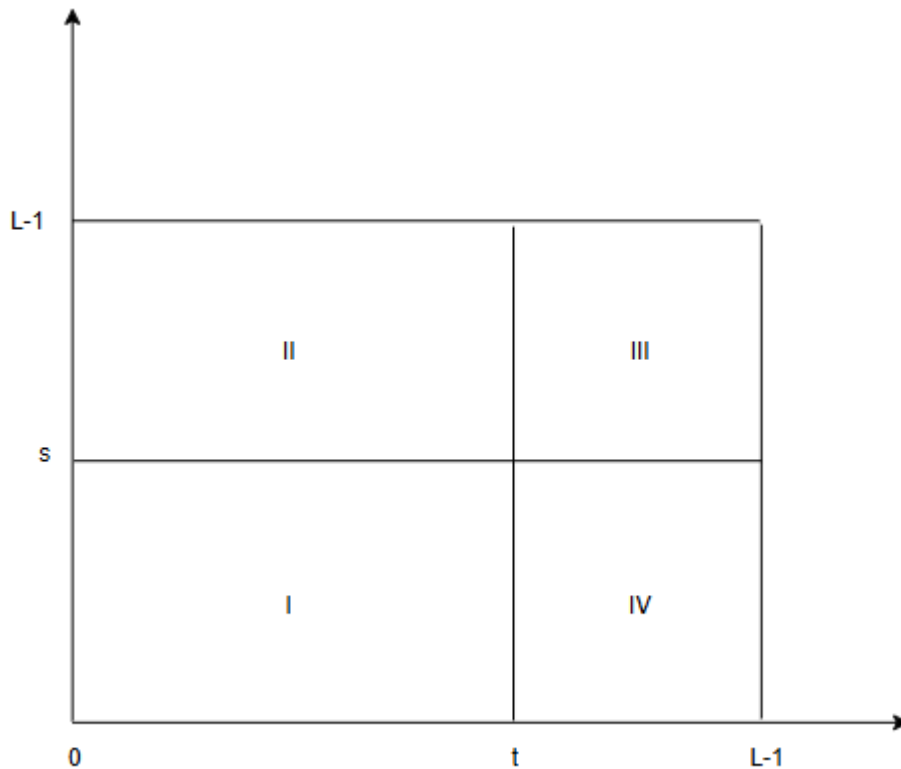


Рис. 4.4. Сегментація зображення

Сегментація зображення (рис. 4.4) дозволяє виділити об'єкти або області в зображенні. Один із методів сегментації – метод порогового значення. Формула для бінаризації зображення виглядає так [55]:

Якщо інтенсивність пікселя $\geq$ поріг, то піксель належить до об'єкта

Інакше піксель належить до фону

Якщо $I(x, y) \geq T$, то $I'(x, y) = 1$ (білий)

Інакше $I'(x, y) = 0$ (чорний)

$I(x, y)$ інтенсивність пікселя в точці (x, y)

T порогове значення

$I'(x, y)$ значення пікселя в бінаризованому зображенні (0 або 1)

Процедура починається з вхідного зображення, яке підлягає обробці. Першим кроком є визначення порогового значення T . Далі, для кожного

пікселя зображення, виконується перевірка: якщо інтенсивність пікселя перевищує поріг, піксель позначається як частина об'єкта (білий), в іншому випадку – як фон (чорний).

Вибір оптимального порогу є ключовим моментом у методі порогового значення. Існує багато методів визначення порогу [51, 55].

Метод Оцу автоматично вибирає поріг, який мінімізує внутрішньокласову дисперсію і максимізує міжкласову дисперсію. Адаптивні методи вибирають поріг для кожної області зображення окремо, що дозволяє врахувати неоднорідність освітлення. Користувач вручну задає поріг, що може бути корисним у деяких випадках.

Метод порогового значення є простим, але ефективним способом сегментації зображень. Блок-схема бінаризації наочно демонструє, як цей метод працює, дозволяючи виділити об'єкти в зображенні.

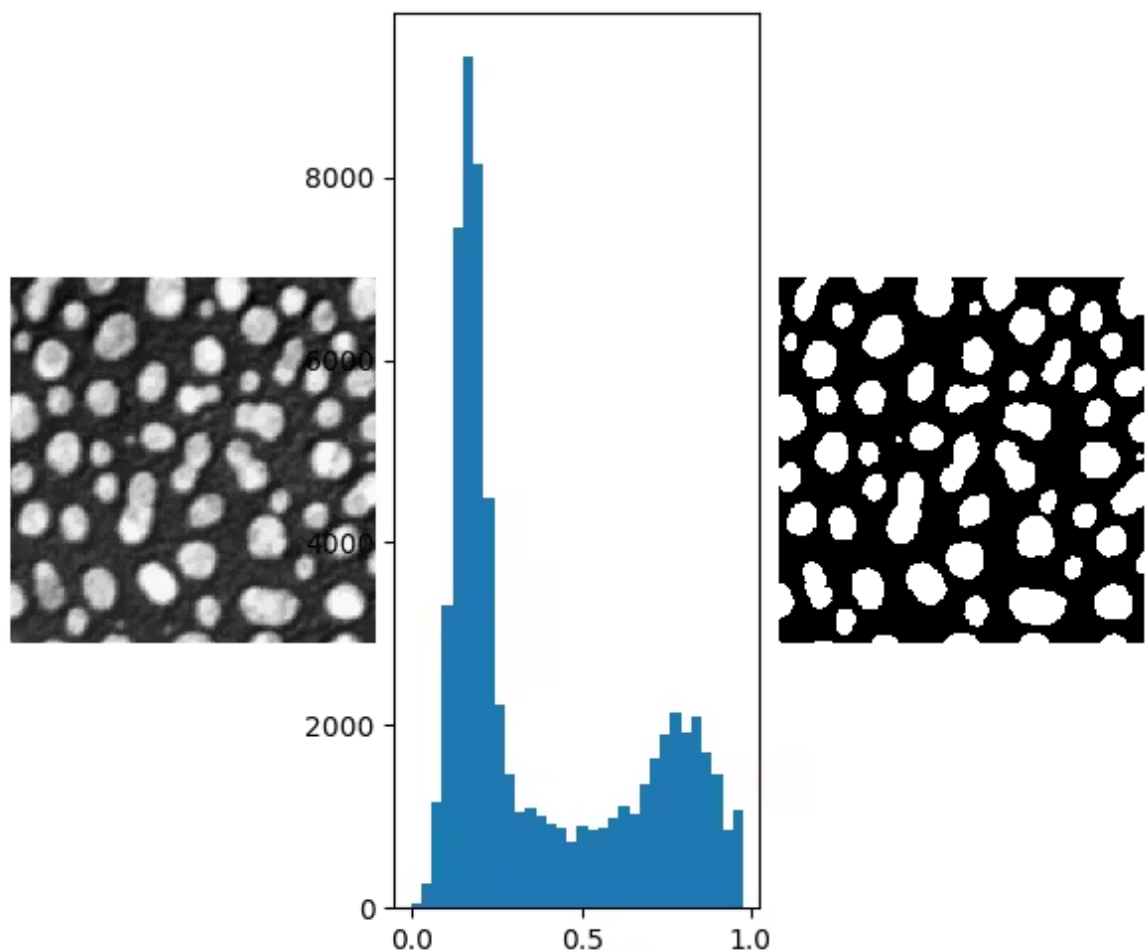


Рис. 4.5. Виявлення країв

Виявлення країв (рис. 4.5) є критично важливим для ідентифікації контурів об'єктів. Один із популярних методів – оператор Собеля, який використовує градієнти для виявлення країв [39].

Градієнти в обробці зображень є важливими для виявлення країв та текстур. Вони дозволяють визначити, де відбуваються різкі зміни інтенсивності пікселів, що є ознакою наявності країв у зображенні. Горизонтальний градієнт (G_x) [51]:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I \quad (4.2)$$

Ця матриця, також відома як ядро Собеля, використовується для виявлення горизонтальних країв. Вона підкреслює зміни інтенсивності пікселів у горизонтальному напрямку. I – це матриця інтенсивностей пікселів вихідного зображення. Згортка матриці G_x з I дозволяє обчислити горизонтальний градієнт для кожного пікселя.

Вертикальний градієнт (G_y) [51]:

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I, \quad (4.3)$$

де G_x – матриця, що представляє горизонтальний градієнт, G_y – матриця, що представляє вертикальний градієнт, I – матриця інтенсивностей пікселів вихідного зображення, $*$ – операція згортки.

Ця матриця також є ядром Собеля, але вона використовується для виявлення вертикальних країв. Вона підкреслює зміни інтенсивності пікселів у вертикальному напрямку. Як і в випадку з горизонтальним градієнтом, згортка матриці G_y з I дозволяє обчислити вертикальний градієнт для кожного пікселя.

Величина градієнта в кожній точці зображення обчислюється як евклідова норма векторів G_x і G_y [51]:

$$G = \sqrt{(G_x^2 + G_y^2)} \quad (4.4)$$

Величина градієнта, яка вказує на силу зміни інтенсивності пікселів. Великі значення G свідчать про наявність країв у зображенні.

Напрямок градієнта можна обчислити як [55, 103]:

$$\theta = \arctan(G_y / G_x), \quad (4.5)$$

де θ – це кут, що вказує напрямок градієнта. Напрямок градієнта є перпендикулярним до краю, що дозволяє визначити, в якому напрямку змінюється інтенсивність.

Великі значення градієнта вказують на наявність країв. Це важливо для задач, пов'язаних з виявленням об'єктів у зображеннях. Вказує на перпендикулярне до краю напрямком.

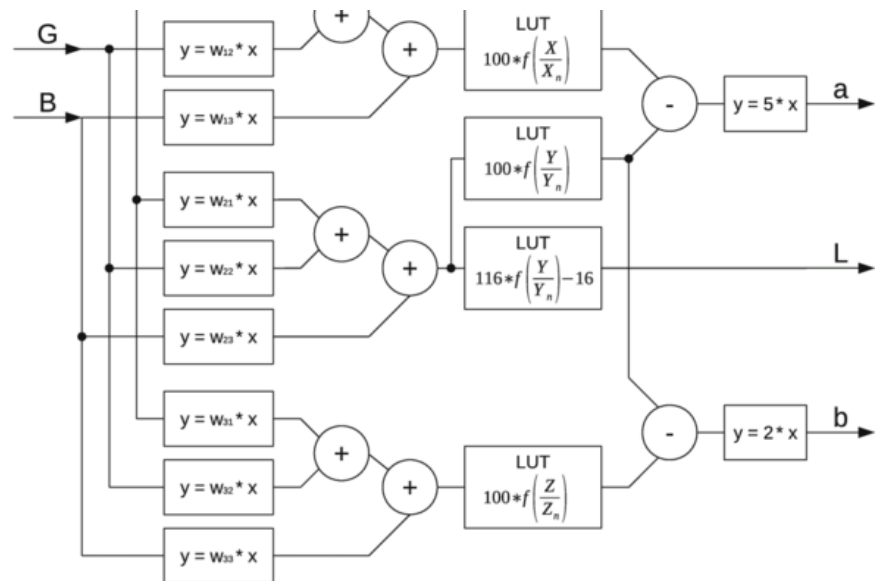


Рис. 4.6. Перетворення кольору

Перетворення кольору (рис. 4.6) дозволяє змінювати колірні простори зображення. Наприклад, для перетворення з RGB в Grayscale можна використовувати формулу [57, 64-70]:

$$Y = 0.299R + 0.587G + 0.114B, \quad (4.6)$$

де Y – значення яскравості в градаціях сірого, а R , G , B – значення червоного, зеленого та синього каналів відповідно.

Коефіцієнти (0.299, 0.587, 0.114) у формулі були вибрані відповідно до чутливості людського ока до різних кольорів. Людське око найбільш чутливе до зеленого кольору, потім до червоного, і найменше до синього. Це перетворення дозволяє отримати більш природне відтворення яскравості при переході від кольорового зображення до чорно-білого [64-70].

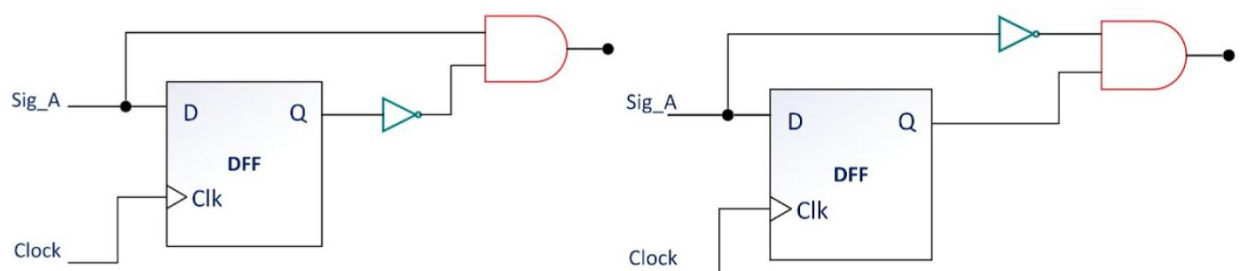


Рис. 4.7. Порогове значення

Метод порогового значення (рис. 4.7) є популярним способом бінаризації зображення, який дозволяє перетворити зображення з багатьма градаціями сірого у бінарне зображення (чорно-біле). Основна ідея полягає в тому, що пікселі з інтенсивністю вище певного порога стають білими (1), а нижче – чорними (0).

Метод порогового значення може бути використаний для бінаризації зображення, як зазначено раніше. Додатково, для адаптивного порогового значення можна використовувати [57]:

$$T(x, y) = \frac{1}{N} \sum_{(i,j) \in \omega} I(i, j) + C, \quad (4.7)$$

де C – константа, що коригує поріг, а N – кількість пікселів у вікні.

Адаптивне порогове значення забезпечує гнучкість, оскільки воно дозволяє налаштовувати поріг відповідно до локальних характеристик зображення, що особливо корисно в умовах неоднорідного освітлення.

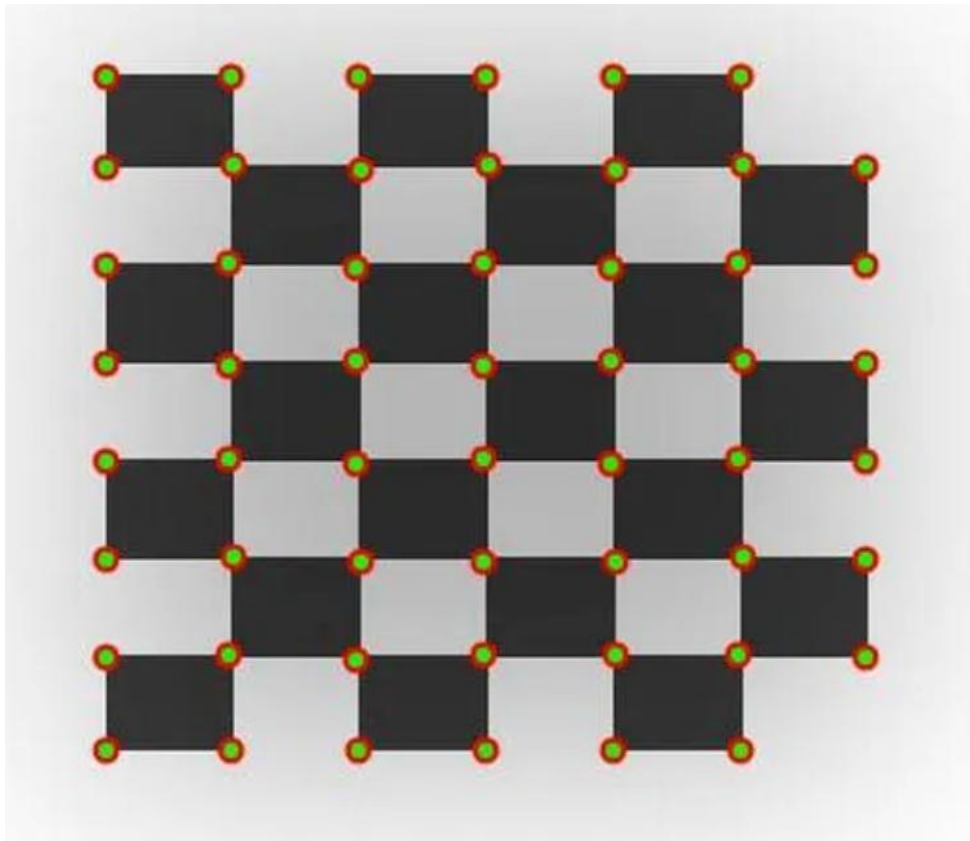


Рис. 4.8. Виявлення ключових точок

Виявлення ключових точок (рис. 4.8) є важливим для розпізнавання об'єктів. Один із методів – Harris Corner Detection, який використовує матрицю градієнтів [55]:

$$M = \begin{pmatrix} \sum(I_x^2) & \sum(I_x * I_y) \\ \sum(I_x * I_y) & \sum(I_y^2) \end{pmatrix}, \quad (4.8)$$

де I_x градієнт за віссю x (обчислюється за допомогою оператора Собеля або інших), I_y градієнт за віссю y . Сума обчислюється в деякому вікні навколо розглядуваної точки.

Ключові точки визначаються на основі власних значень матриці M . Якщо обидва власні значення великі, то точка є кутом. Якщо одне власне

значення велике, а інше мале, то точка лежить на краю. Якщо обидва власних значення малі, то точка належить однорідній області зображення.

Замість обчислення власних значень, часто використовується спрощений критерій [51]:

$$R = \det(M) k * \text{trace}(M)^2, \quad (4.9)$$

де $\det(M)$ визначник матриці M , $\text{trace}(M)$ слід матриці M , k емпіричний параметр, який зазвичай дорівнює 0.04. Якщо значення R перевищує деякий поріг, то точка вважається ключовою.

Ключові точки визначаються за допомогою детектора [57]:

$$R = \det(M) k * (\text{trace}(M))^2, \quad (4.10)$$

де k – константа, що контролює чутливість детектора, $\det(M)$ – визначник матриці M , який показує, наскільки сильно змінюється інтенсивність у околі пікселя, що аналізується, $\text{trace}(M)$ – слід матриці M , що є сумою її діагональних елементів, тобто $I_x^2 + I_y^2$. Це значення характеризує загальну інтенсивність градієнтів.

Спрощений критерій для виявлення ключових точок, заснований на визначнику та сліді матриці градієнтів, є ефективним методом для виявлення важливих особливостей у зображеннях [39]. Використання параметра k дозволяє налаштувати чутливість детектора, що робить цей метод гнучким і адаптивним до різних умов зображення.

Переваги методу Harris

- інваріантність до обертання – критерій відбору ключових точок не залежить від орієнтації зображення;
- простота реалізації – алгоритм відносно простий у розумінні та реалізації;

- ефективність – може бути ефективно реалізований за допомогою засобів лінійної алгебри.

Впровадження проводиться у кілька етапів. Було опрацьовано кілька методів виділення функцій: виявлення країв, сегментація зображення, фільтрація шуму, трансформація кольору ORB, виявлення кутів тощо. Після інтеграції цих методів обробки було створено базову модель навчання з 50 епохами, щоб протестувати архітектуру та перевірити продуктивність моделі на першому етапі тестування.

На другому етапі було розроблено функції для досягнення оптимальної продуктивності моделі та правдивості результатів на основі різних ознак. Найбільш ефективна модель складає сукупність всіх ознак для оцінки зображення. Далі модель була налаштована відповідно до зміни параметрів: епох, кількості одиниць зображень.

Процес обробки зображень включає кілька ключових аспектів, які суттєво впливають на ефективність виявлення вогню. Для цього були застосовані різноманітні методи, зокрема [39]:

- фільтрація шуму – цей метод дозволяє зменшити небажані артефакти, які можуть заважати подальшій обробці зображень;
- сегментація зображення – за допомогою цього методу зображення розбивається на окремі області, що полегшує виявлення об'єктів;
- виявлення країв – цей етап допомагає визначити контури об'єктів, що є важливим для ідентифікації вогню;
- перетворення кольору – зміна колірної палітри зображення може підкреслити важливі деталі, пов'язані з вогнем;
- порогове значення – цей метод дозволяє виділити пікселі, що відповідають певним критеріям, що допомагає у виявленні вогню [39];
- виявлення ключових точок – цей підхід дозволяє ідентифікувати важливі елементи, які можуть свідчити про наявність вогню.

Завдяки цим методам були підготовлені дані для навчання моделі.

Вхідні дані для моделі – зображення. Для цього використовується згортоква нейронна мережа. Щоб зекономити ресурси на навчання великих об'ємів даних у нейронних мережах, використовуються початково навчені вагові коефіцієнти та перенавчено їх за допомогою нового набору вхідних даних – трансферне навчання за допомогою моделі Inception V3 (бібліотека PyTorch). Частина шарів заморожена. Лише частина шарів навчається за новими даними [51, 55, 57].

Моделю бере вхідні дані та прогнозує ймовірності для кожного класу. Тоді ймовірності перетворюються в єдиний прогноз для певного класу.

Набори для навчання використовувалися для навчання моделі, тоді як набори для перевірки та тестування використані для налаштування та перевірки точності моделі відповідно. Одного разу модель пройшла навчання. Дані з тестового набору пропускаються через навчену модель для прогнозування результату. Точність, зафіксована в цьому наборі, є фактичною загальною точністю моделі.

Після перевірки моделі за допомогою тестового набору можна вирішити, чи продуктивність моделі адекватна чи потребує покращення. Це робиться шляхом зміни гіперпараметрів моделі, що забезпечує найкращий результат. Це робиться за допомогою наборів даних перевірки.

Коли модель навчена та досягнуто оптимальної продуктивності, її можна розгортати для використання в реальних програмах для перевірки лісових пожеж.

Набір даних складається з необроблених зображень, які класифікуються як «пожежа», «відсутність пожежі» або «початок пожежі». Перед використанням цих зображень для навчання моделі необхідно провести їх додаткову обробку. Після цього зображення проходять обробку, а отримані характеристики вводяться в навчальну модель. Прогнози, які генерує модель, базуються на шаблонах, які вона засвоїла з вибраного набору функцій під час навчального процесу. Це дозволяє отримати вихідний вектор, що містить

ймовірності для можливих класів, який потім перетворюється на єдиний прогноз для конкретного класу [23, 75-80].

Для ідентифікації справжньої пожежі були використані різноманітні релевантні функції, що забезпечують детальний аналіз зображень. На першому етапі було проведено видалення шуму [22, 85], що ілюструється на рисунку 4.9. Спочатку застосовано Гаусівський фільтр, який ефективно зменшує шум. Наступним кроком був використаний двосторонній фільтр, що забезпечує згладжування зображення при збереженні країв, що є важливим для подальшого аналізу. Як третій фільтр було обрано медіанний фільтр, який сприяє зменшенню випадкового шуму, що також позитивно впливає на якість оброблених зображень [23, 80].

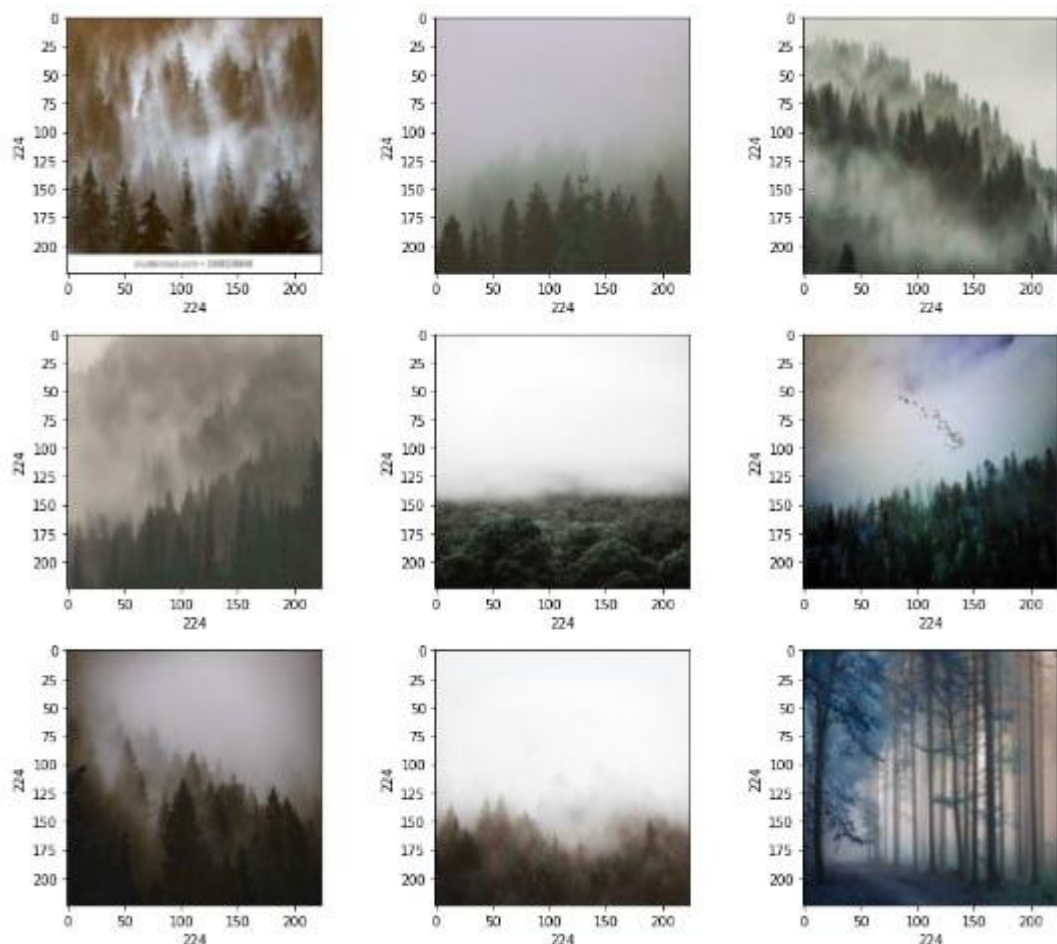


Рис. 4.9. Видалення шуму

Сегментація є одним з ключових етапів при аналізі вхідних зображень. Цей метод застосовується для ідентифікації зображення шляхом виділення кривих, що відповідають кожному пікселю. Сегментація дозволяє оптимізувати оброблюване зображення, роблячи його більш зрозумілим для подальшого навчання нейронної мережі [23, 85-90].

У даному дослідженні була реалізована сегментація за методом Оцу (рис. 4.10), який базується на використанні порогового значення. На початковому етапі відбувається обробка вхідних даних зображення, після чого будується гістограма. Ця інформація слугуватиме основою для спрощення подальшого аналізу, що дозволяє ефективніше виділяти важливі елементи зображення і покращувати результативність навчання моделі [23, 91].

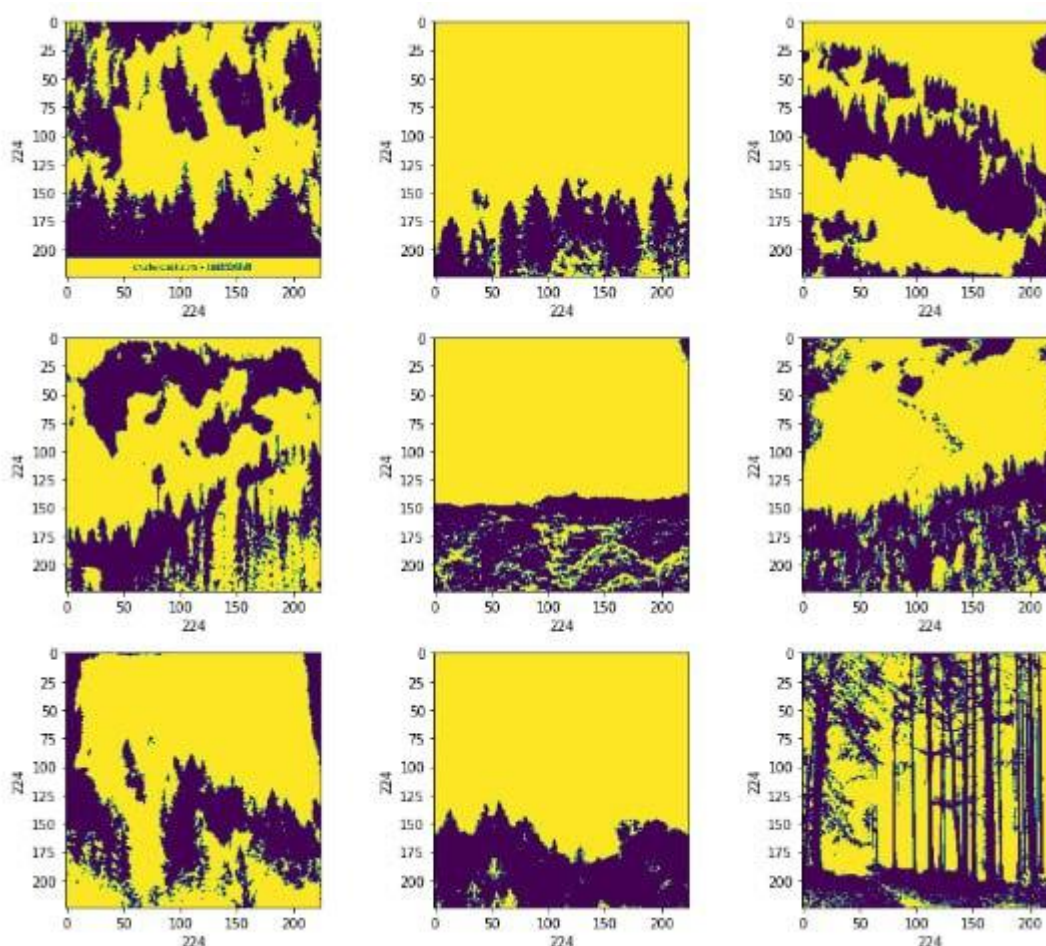


Рис. 4.10. Сегментація зображення

На наступному етапі процесу обробки зображень проводиться трансформація колірної сітки. Застосування перетворення кольору (рис. 4.11) має на меті введення нових відтінків, що сприяють більш детальному аналізу вогню на всьому зображенні. Така зміна колірної палітри дозволяє підкреслити важливі риси, пов'язані з вогняними явищами, і покращує видимість об'єктів, що потребують подальшого вивчення [91-95].

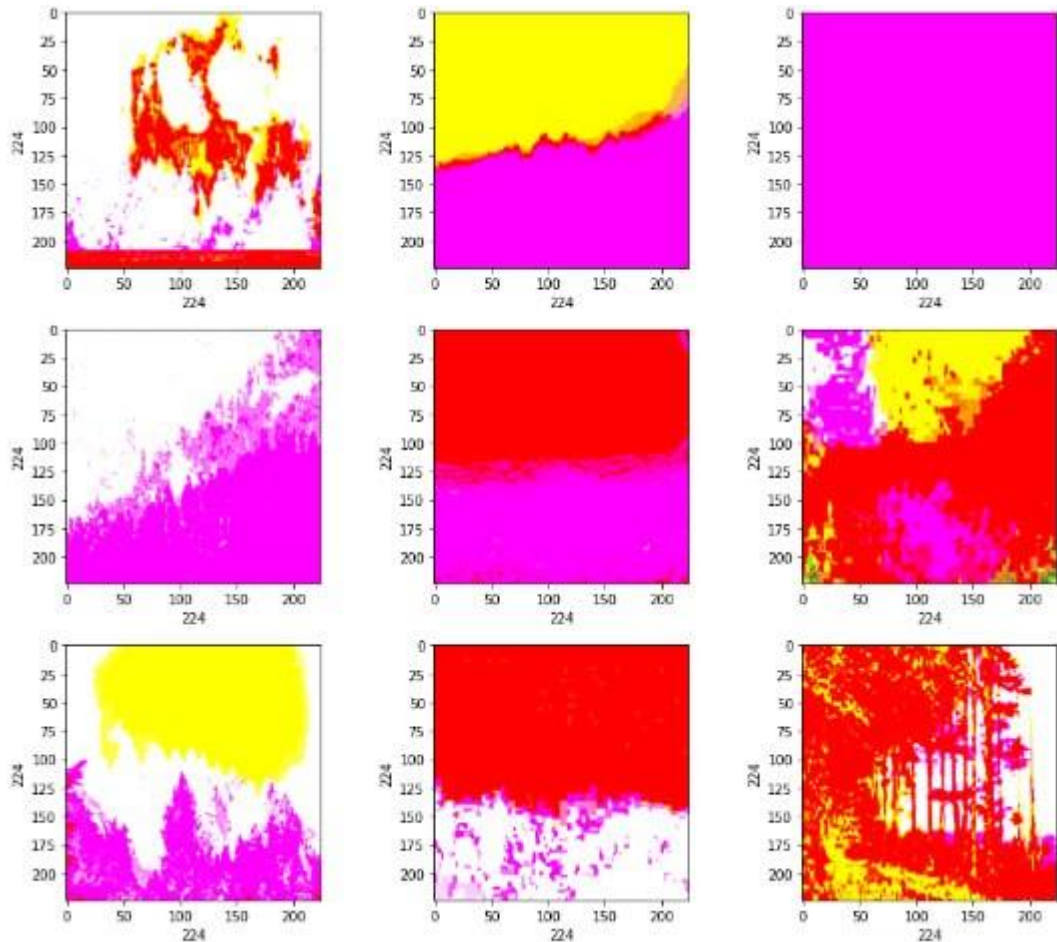


Рис. 4.11. Трансформація кольору

Порогове значення застосовується для сегментації зображення шляхом модифікації пікселів. Цей метод суттєво покращує аналіз вхідного зображення, оскільки дозволяє більш чітко розрізняти деталі. Зокрема, він сприяє кращому виділенню елементів, таких як ліс, фон і вогонь. Використання двійкового порогового значення забезпечило підвищену ефективність у аналізі вогню на зображенні (рис. 4.12).

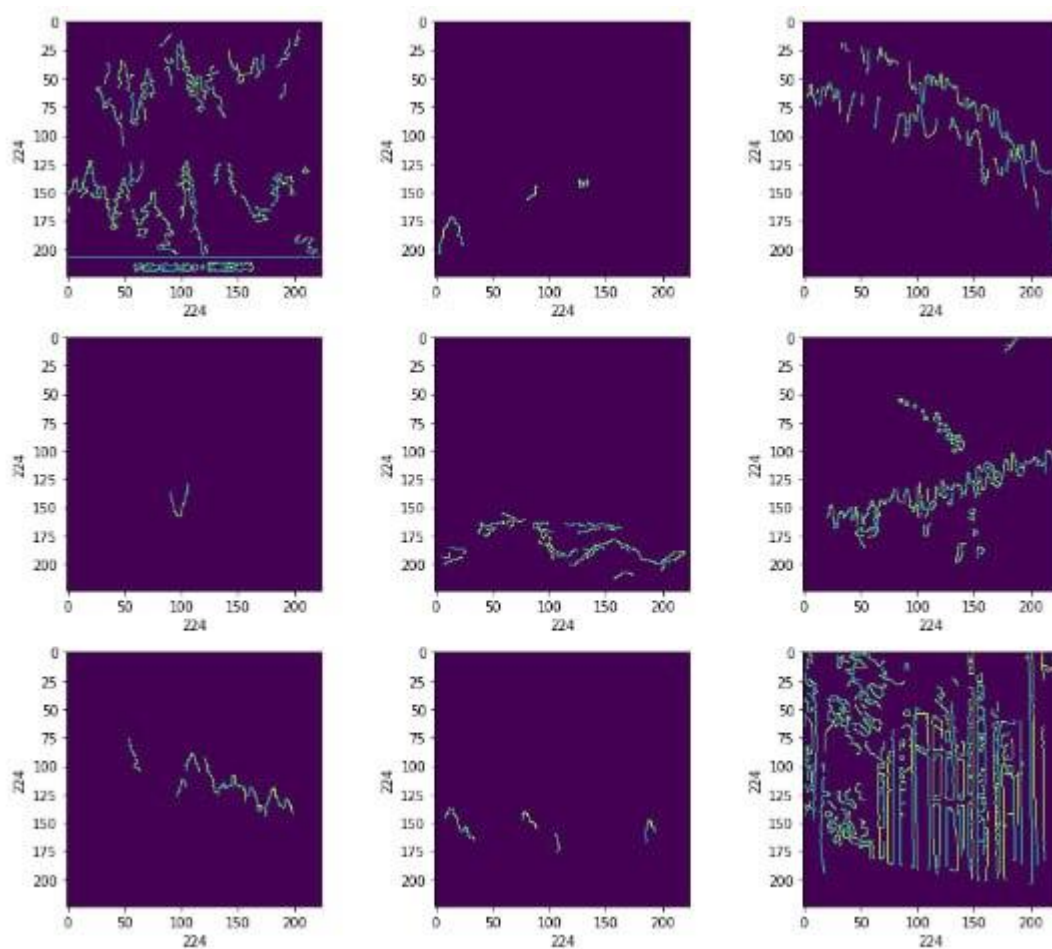


Рис. 4.12. Порогове значення

Виявлення країв застосовується для оцінки здатності країв виокремлювати об'єкт вогню на зображенні. Цей метод дозволяє виявити зміни яскравості, коли одна частина зображення переходить в іншу. Для аналізу змін інтенсивності зображення використано фільтр Собеля.

Крім того, для виявлення ключових точок застосовано алгоритм BRIEF, який дозволяє визначити інтенсивність пікселів, що сприяє ретельному аналізу окремих кутів на зображенні. Завдяки цьому алгоритму вдається ефективно виявити краї дерев, що є важливим для подальшої ідентифікації об'єктів на зображенні [23].

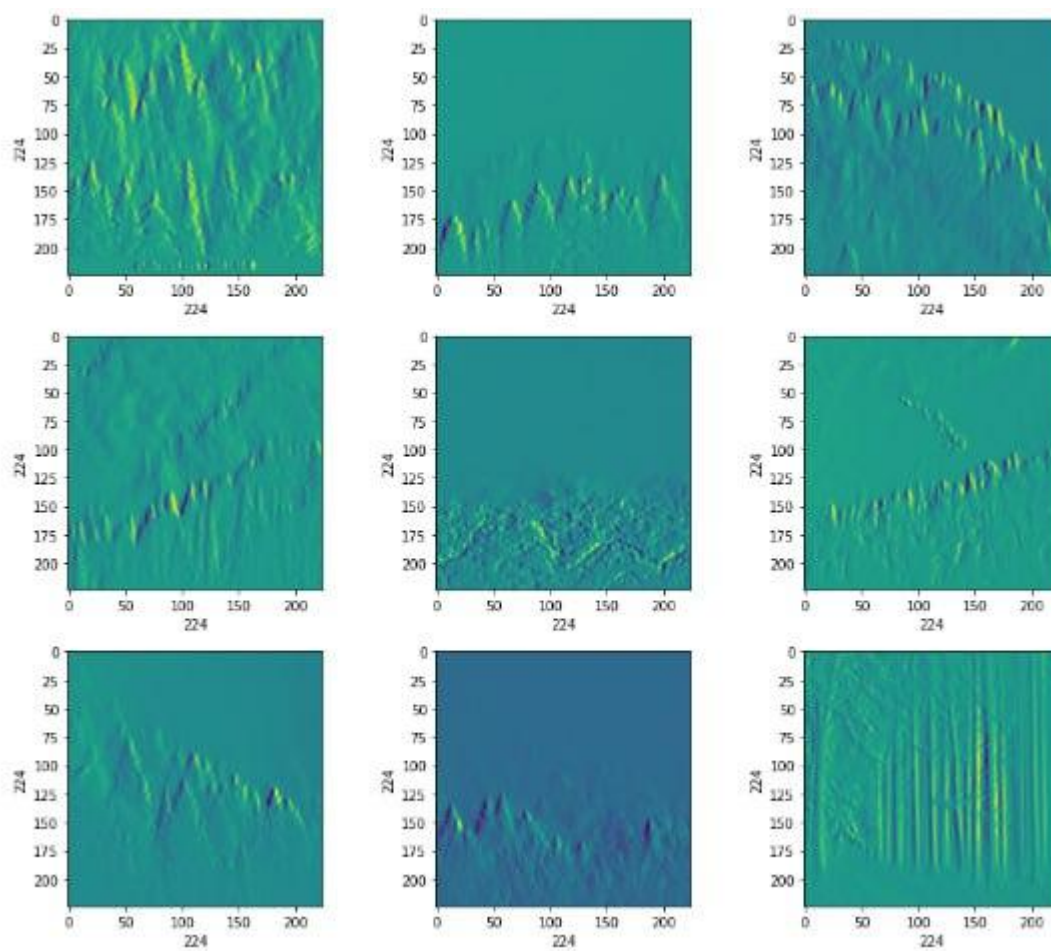


Рис. 4.13. Виявлення горизонтальних та вертикальних країв зображення

Виявлення горизонтальних та вертикальних країв зображення виглядає як зображення, на якому краї виділені яскравими лініями, що вказують на місця, де відбуваються різкі зміни яскравості (рис. 4.13) [23, 91-95].

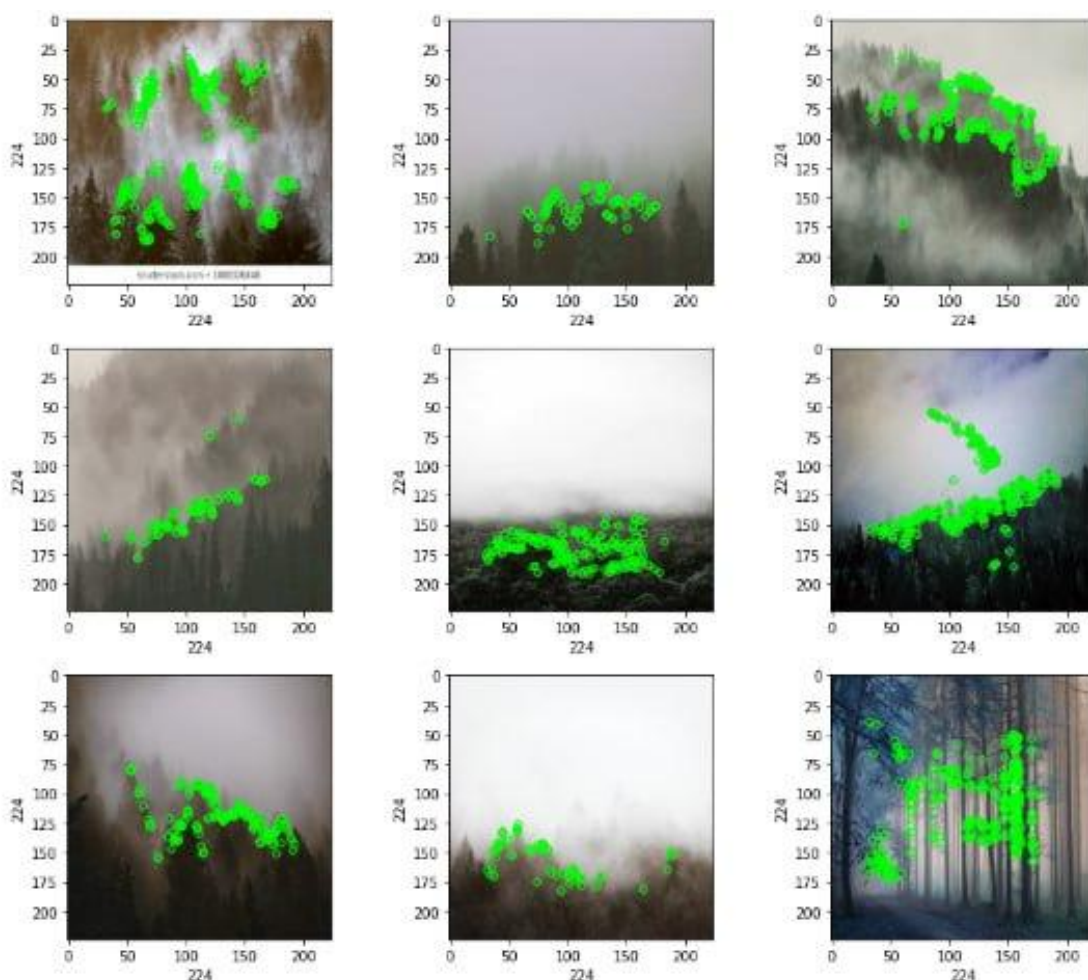


Рис. 4.14. Виявлення ключової точки

Ключові точки (рис. 4.14), які виявляються за допомогою алгоритму BRIEF, можуть бути використані для подальшого аналізу, наприклад, для ідентифікації країв дерев або інших об'єктів, що можуть бути пов'язані з вогнем. Ключові точки зазвичай позначаються на зображенні яскравими маркерами, що дозволяє легко їх ідентифікувати [23, 95-100].

Усі описані дії необхідні для аналізу вхідних зображень та навчання нейронної мережі, яка буде виявляти лісові пожежі на початкових етапах [5-8]. Алгоритм згорткової нейронної мережі виглядає наступним чином [57]:

$$A_{ij} = \sum \sum * X_{(i-k)j-1}, \quad (4.11)$$

де A_{ij} – елемент карти ознак на позиції (i, j) , w_{kl} – значення ваги на позиції (k, l) в ядрі згортки, $X_{(i-k)j-l}$ – значення елемента вхідної сітки на позиції $(i-k, j-l)$, k і l – розміри ядра згортки.

Після виконання операції згортки до отриманої карти ознак застосовується нелінійна функція активації, що вводить елемент нелінійності в процес обробки даних.

На основі прогнозів, отриманих на невидимому наборі тестових даних, точність розробленої моделі становить 92%. Це підтверджує гіпотезу про те, що поєднання всіх застосованих функцій та методів аналізу зображень виявляється більш ефективним для виявлення лісових пожеж.

4.2 Детальний опис процесу розробки системи: основні етапи та характеристики

На початку коду імпортуються необхідні бібліотеки, такі як `numpy`, `pandas`, `matplotlib`, `tensorflow` та `keras`. Ці бібліотеки забезпечують функціональність для обробки даних, візуалізації та побудови нейронних мереж.

Далі код завантажує дані з набору зображень, які містять як позитивні, так і негативні приклади лісових пожеж. Зображення попередньо обробляються: змінюється їх розмір, нормалізуються пікселі, а також створюються мітки для навчання моделі:

```
# Завантаження даних
```

```
Оголосити змінну data
```

```
data ← Завантажити дані (деталі завантаження залежать від джерела даних)
```

```

# Обробка зображень
data ← Змінити розмір data до форми (кількість_зображень, 128, 128,
3)
data ← Перетворити тип даних data на дійсний з рухомою комою
(float32)
data ← Поділити кожен елемент в data на 255 (нормалізація до
діапазону [0, 1])

```

Дані діляться на навчальну та тестову вибірки, що дозволяє оцінити ефективність моделі на нових, невідомих даних:

```

# Імпортувати функцію для розділення даних
Імпортувати функцію train_test_split з бібліотеки
sklearn.model_selection

```

```

# Розділити дані на навчальний та тестовий набори
(X_навчальний, X_тестовий, y_навчальний, y_тестовий) ← Викликати
функцію train_test_split з аргументами:

```

```

    вхідні_дані: data
    цільові_мітки: labels
    розмір_тестового_набору: 0.2
    випадкове_зерно: 42

```

Далі створено архітектуру згорткової нейронної мережі, яка складається з кількох згорткових шарів, шарів підвибірки, а також повнозв'язних шарів. Це дозволяє моделі вчитися виявляти складні патерни в зображеннях:

```

# Створення послідовної моделі
Оголосити змінну model та ініціалізувати її як послідовну модель

```

Додавання шарів до моделі

Згортковий шар 1

Додати до model згортковий шар з:

кількість_фільтрів: 32

розмір_ядра: (3, 3)

функція_активації: ReLU

розмір_вхідних_даних: (128, 128, 3) (лише для першого шару)

Шар максполінгу 1

Додати до model шар максполінгу з:

розмір_вікна: (2, 2)

Згортковий шар 2

Додати до model згортковий шар з:

кількість_фільтрів: 64

розмір_ядра: (3, 3)

функція_активації: ReLU

Шар максполінгу 2

Додати до model шар максполінгу з:

розмір_вікна: (2, 2)

Шар згладжування

Додати до model шар згладжування (перетворення багатовимірного виходу на одновимірний вектор)

Повністю зв'язаний шар 1

Додати до model повністю зв'язаний (Dense) шар з:

кількість_нейронів: 128

функція_активації: ReLU

Шар Dropout

Додати до model шар Dropout з:

коефіцієнт_виключення: 0.5

Вихідний повністю зв'язаний шар (для бінарної класифікації)

Додати до model вихідний повністю зв'язаний (Dense) шар з:

кількість_нейронів: 1

функція_активації: Sigmoid

Після побудови архітектури модель компілюється з використанням оптимізатора, функції втрат та метрики для оцінки її продуктивності:

Модель навчається на навчальних даних, використовуючи метод `fit`. Параметри, такі як кількість епох та розмір пакету, можуть бути налаштовані для оптимізації процесу навчання:

Викликати метод `fit` моделі з такими параметрами:

навчальні_вхідні_дані: `X_навчальний`

навчальні_цільові_мітки: `y_навчальний`

кількість_епох: 10

розмір_пакету: 32

валідаційні_дані: (`X_тестовий`, `y_тестовий`)

Присвоїти результат виклику змінній `history` (для відстеження історії навчання)

Після навчання модель оцінюється на тестових даних, що дозволяє перевірити її точність та здатність до генералізації:

Оцінка моделі на тестових даних

Викликати метод `evaluate` моделі з такими параметрами:

тестові_вхідні_дані: X_тестовий

тестові_цільові_мітки: y_тестовий

*Присвоїти результати виклику (значення функції втрат та точність)
змінним test_loss та test_accuracy відповідно*

Виведення результатів оцінки

*Вивести на екран повідомлення, що включає значення змінної
test_accuracy (точність на тестових даних)*

Результати навчання візуалізовані. Побудовано графіки втрат та точності, що допомагає зрозуміти, як модель навчається протягом епох:

Побудова графіка точності навчання та валідації

Побудова графіка точності навчання

*Викликати функцію plot з бібліотеки matplotlib.pyplot з такими
параметрами:*

значення по осі x: індекси епох з history.history['accuracy']

*значення по осі y: значення точності навчання з
history.history['accuracy']*

мітка: 'accuracy'

Побудова графіка точності валідації

*Викликати функцію plot з бібліотеки matplotlib.pyplot з такими
параметрами:*

*значення по осі x: індекси епох (за замовчуванням, якщо не вказано
окремо)*

*значення по осі y: значення точності валідації з
history.history['val_accuracy']*

мітка: 'val_accuracy'

Встановлення мітки для осі x

*Викликати функцію xlabel з бібліотеки matplotlib.pyplot з параметром:
текст мітки: 'Epoch'*

Встановлення мітки для осі y

*Викликати функцію ylabel з бібліотеки matplotlib.pyplot з параметром:
текст мітки: 'Accuracy'*

Відображення легенди

*Викликати функцію legend з бібліотеки matplotlib.pyplot без параметрів
(для відображення міток графіків)*

Відображення графіка

*Викликати функцію show з бібліотеки matplotlib.pyplot без параметрів
(для відображення створеного графіка)*

Основною метою даного дослідження є аналіз різних методів обробки зображень лісових пожеж з метою розробки архітектури нейронної мережі, яка забезпечить раннє виявлення та попередження лісових пожеж.

Дослідження проведено на основі знімків, отриманих з відкритих джерел, зокрема NASA Earth Observatory. Для обробки та аналізу супутникових зображень використано бібліотеки Python, такі як Keras, TensorFlow та PyTorch [45]. Завдяки методам моделювання було спроектовано архітектуру системи та продемонстровано різні варіанти її використання [53, 54].

Ефективним підходом до зменшення ризику виникнення природних катастроф є розробка та впровадження сучасної системи підтримки прийняття рішень (СППР) для кожного регіону України [17, 29]. Ця система акумулює інформацію про технічні, соціальні та економічні характеристики регіону, що

дозволяє сформуванати ефективну стратегію запобігання природним катастрофам [58-61, 99].

Нейронні мережі, які використовуються в системах підтримки прийняття рішень (СППР) [50, 58, 100-103], складаються з кількох ключових компонентів, кожен з яких виконує специфічні математичні функції. Основні компоненти включають в себе вхідні нейрони, приховані шари, вихідні нейрони, ваги, функції активації та механізми навчання [24-26, 101].

Вхідні нейрони приймають дані з зовнішнього середовища. Нехай $x_1, x_2, \dots, x_n$ – це вхідні значення, які представляють собою ознаки (фактори), що впливають на прийняття рішень [24-26]. Кожне вхідне значення може бути представлене як вектор:

$$x = [x_1, x_2, \dots, x_n]^T \quad (4.12)$$

Кожен вхідний нейрон має ваги, які визначають важливість кожного вхідного сигналу. Ваги можуть бути представлені у вигляді вектора [39]:

$$w = [w_1, w_2, \dots, w_n]^T, \quad (4.13)$$

де w – це вектор, а кожен елемент w_i відповідає за важливість відповідного вхідного значення x_i .

На кожному нейроні в прихованому або вихідному шарі обчислюється лінійна комбінація вхідних сигналів [39]:

$$z = w^T * x + b, \quad (4.14)$$

де b – це зсув (bias), який додається для підвищення гнучкості моделі, z – це результат лінійної комбінації, який є сумою вхідних значень, помножених на відповідні ваги, плюс зсув.

Результат z передається через функцію активації, яка визначає, чи активується нейрон. Функції активації можуть бути різними, наприклад, сигмоїдальна, ReLU (Rectified Linear Unit) або тангенс гіперболічний. Для сигмоїдальної функції активації це виглядає так [39]:

$$a = \sigma(z) = \frac{1}{1+e^{-z}}, \quad (4.15)$$

де a – це вихід нейрону, $\sigma(z)$ – це сигмоїдальна функція, яка перетворює значення z в діапазон від 0 до 1.

Нейронні мережі можуть мати один або кілька прихованих шарів, які виконують аналогічні обчислення. Кожен нейрон у прихованому шарі отримує вхідні дані від попереднього шару, обчислює лінійну комбінацію, застосовує функцію активації і передає результат далі.

Для кожного нейрону в прихованому шарі обчислюється z та a за аналогічними формулами, використовуючи вхідні дані з попереднього шару.

Вихідні нейрони отримують дані з останнього прихованого шару і виконують аналогічні обчислення. Вихідні значення можуть бути представлені як [51]:

$$y = [y_1, y_2, \dots, y_m]^T, \quad (4.16)$$

де m – кількість вихідних класів або результатів.

Для оцінки точності моделі використовується функція втрат, яка вимірює різницю між передбаченими значеннями та фактичними значеннями. Наприклад, для задачі класифікації може використовуватися крос-ентропія [39]:

$$L = - \sum_{i=1}^m y_i \log(y_i), \quad (4.17)$$

де y_i – це передбачене значення для класу i .

Для оптимізації ваг і зсувів у мережі використовується алгоритм зворотного розповсюдження помилки (backpropagation) разом із методом градієнтного спуску. Оновлення ваги виконується за формулою [39]:

$$\omega_j \leftarrow \omega_j - \tau \frac{\partial L}{\partial \omega_j}, \quad (4.18)$$

де τ – це швидкість навчання.

Процес навчання нейронної мережі реалізується на основі набору даних, що містить зображення в необробленому вигляді, які класифікуються як «пожежа», «відсутність пожежі» або «початок пожежі» [48, 50]. Перед використанням для навчання моделі ці зображення підлягають додатковій обробці, що забезпечує їх відповідність вимогам моделі та підвищує точність прогнозування лісових пожеж.

Обробка зображень включає в себе різноманітні етапи, які сприяють підготовці даних для навчання згорткової нейронної мережі. Для оптимізації використання ресурсів та збереження обчислювальних потужностей застосовуються попередньо навчені вагові коефіцієнти, які підлягають повторному навчанні на новому наборі вхідних даних.

Після обробки зображень модель використовує ці дані для прогнозування ймовірності виникнення пожежі для кожного з класів [16]. Набори даних, що використовуються для навчання, також служать для перевірки точності моделі. Після завершення етапу перевірки модель проходить оцінку, що дозволяє отримати кінцеві результати та досягти оптимальної продуктивності алгоритму.

Трансформація кольору (рис. 4.15) є необхідною для введення нового відтінку, що сприяє підвищенню продуктивності алгоритму. Зміна спектру кольорів дозволяє більш ефективно виявляти дим та вогонь, а також забезпечує можливість виявлення більшої кількості деталей у зображеннях.

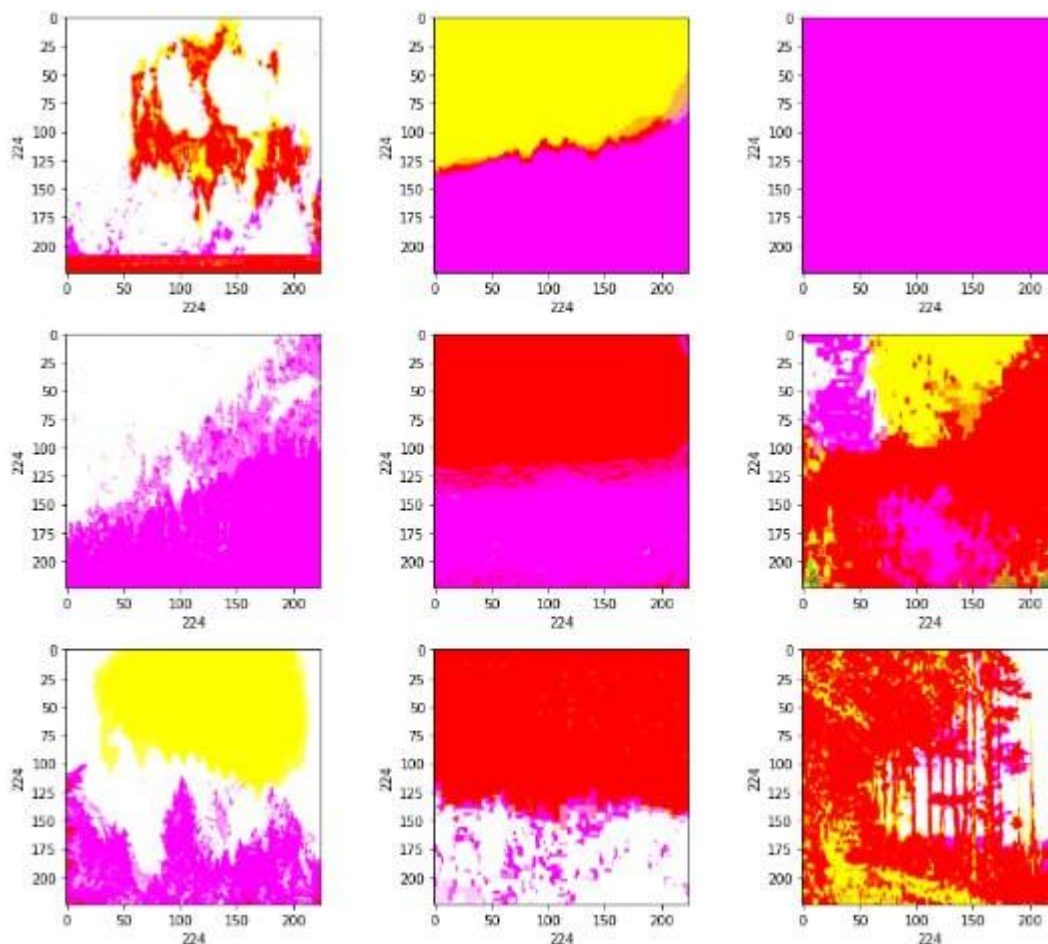


Рис. 4.15. Трансформація кольору

Наступним етапом стало визначення порогових значень (рис. 4.16) для навчання моделі, що дозволяє досягти кращої розрізненості деталей зображення. Після цього було застосовано метод виявлення країв, що допомагає більш точно локалізувати об'єкт вогню на досліджуваному зображенні. В результаті цього процесу були виявлені ключові точки та орієнтація зображення. Проведення детального аналізу та використання численних комбінацій функцій значно підвищують ефективність навчання нейронної мережі.

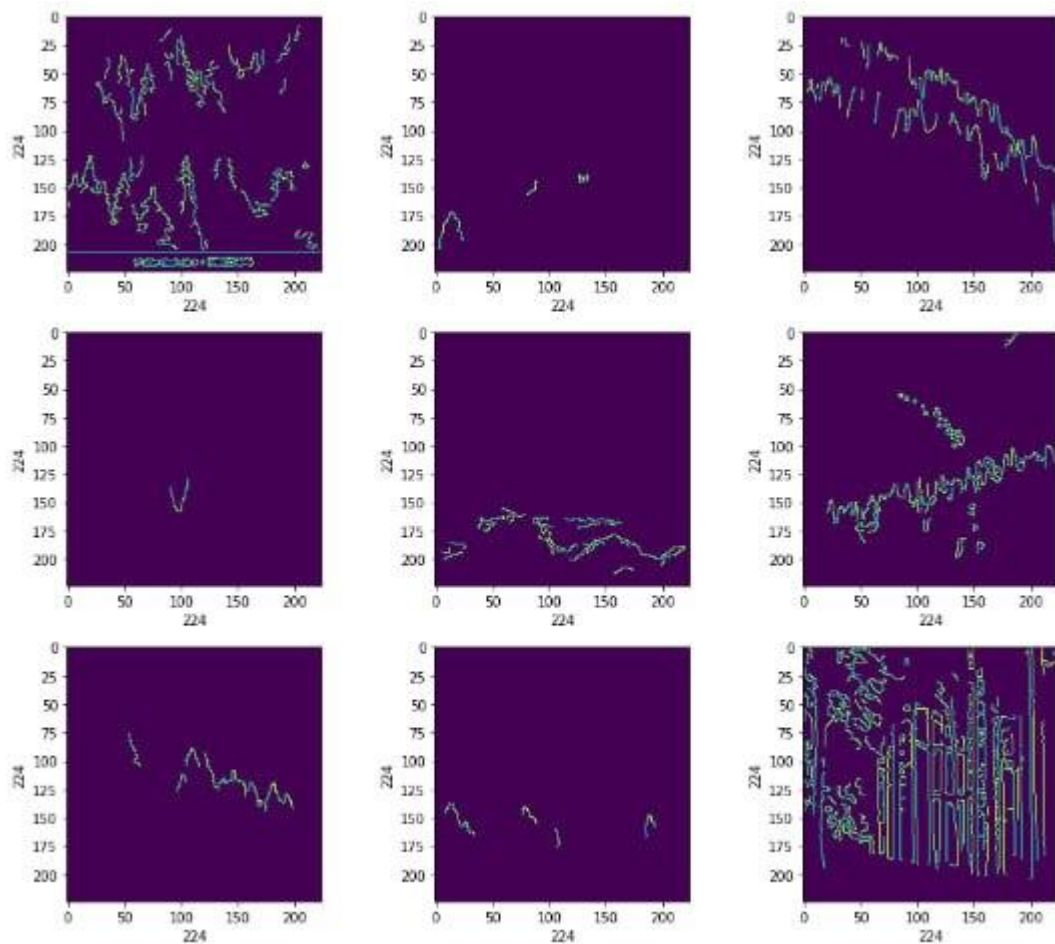


Рис. 4.16. Визначення порогового значення

В рамках проведеного дослідження було здійснено інтеграцію кількох функцій. Використання різних комбінацій цих функцій дало змогу виявити найбільш ефективні методи для навчання нейронної мережі.

Алгоритм згорткової нейронної мережі виглядає наступним чином [57]:

$$A_{ij} = \sum_k \sum_l w_{kl} * X_{(i-k)j-1}, \quad (4.19)$$

де A_{ij} – елемент карти ознак на позиції (i, j) , w_{kl} – значення ваги на позиції (k, l) в ядрі згортки, $X_{(i-k)j-1}$ – значення елемента вхідної сітки на позиції $(i-k, j-1)$, k і l – розміри ядра згортки.

Після згортки до карти ознак застосовується нелінійна функція активації, яка вводить елемент нелінійності в процес обробки.

У першому випадку (рис. 4.13) було використано шумовий фільтр, сегментація зображення та колірний простір для аналізу зображень та навчання нейронної мережі. Можна помітити, що точність результатів складає всього 77,23%.

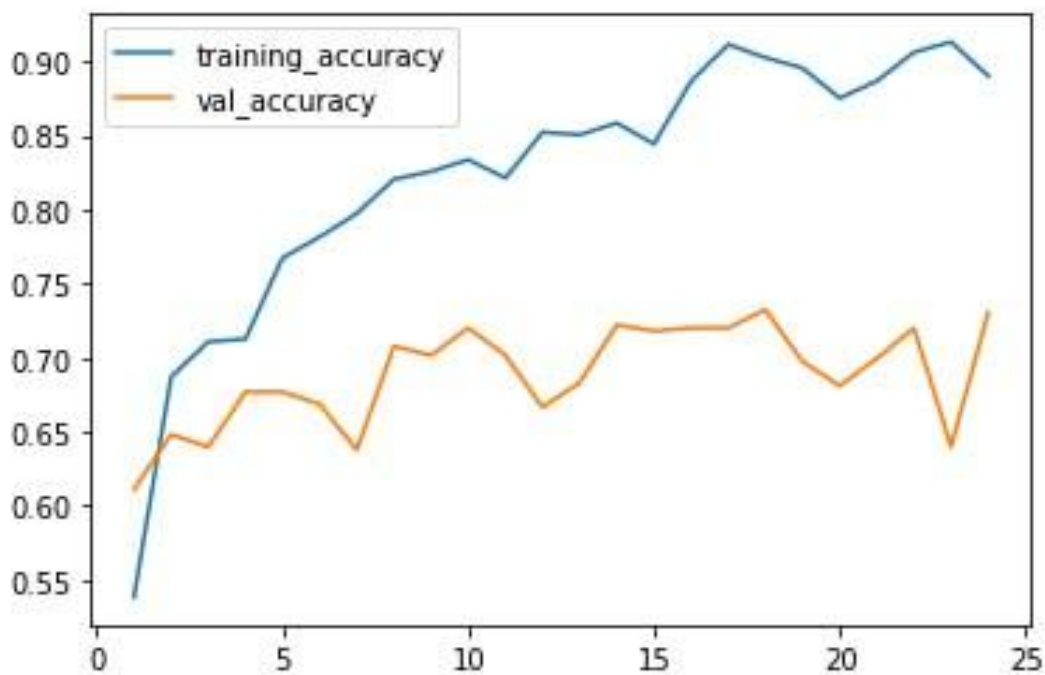


Рис. 4.17. Крива точності навчання нейронної мережі у першому випадку

Крива точності (рис. 4.17) навчання нейронної мережі демонструє, як змінюється точність моделі в процесі навчання. Крива показує, що точність досягла значення 77,23%, що свідчить про те, що модель здатна правильно класифікувати більшість зображень. Однак, це значення може бути недостатнім для практичного використання, особливо в критичних ситуаціях, таких як виявлення вогню. Важливо також зазначити, що крива може мати коливання, що вказує на можливі проблеми з перенавчанням або недостатню кількість даних для навчання.

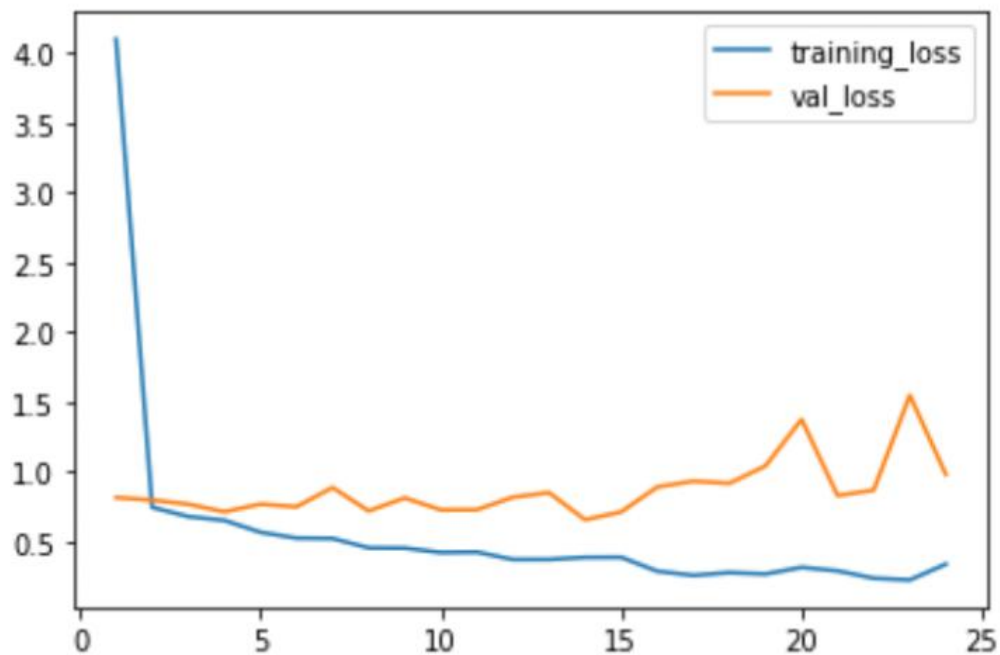


Рис. 4.18. Крива втрат нейронної мережі для першого випадку

Крива втрат (рис. 4.18) відображає зміни значення функції втрат під час навчання нейронної мережі. Зниження значення втрат свідчить про покращення моделі, оскільки вона стає більш точною у своїх прогнозах. Якщо крива втрат стабільно знижується, це вказує на те, що модель навчається ефективно. Однак, якщо спостерігаються різкі коливання або зупинка зниження, це може свідчити про проблеми з навчанням, такі як перенавчання або недостатня складність моделі.

	Fire	No Fire	Start Fire
Fire	286	30	10
No Fire	79	224	19
Start Fire	11	9	26

Рис. 4.19. Матриця для міток

З набору даних із 694, більшість екземплярів тесту класифіковано правильно. 286 зображень вогню було класифіковано як пожежа, 224 зображення лісу без пожежі були класифіковані як відсутність пожежі, а решта 26 зображень, які відображали початкові стадії пожежі, були класифіковані як початок пожежі [16]. Точність і запам'ятовування: оцінені значення точності та запам'ятовування для моделі, навченої за допомогою цієї комбінації функцій, становлять 70% і 71% відповідно, що достатньо добре порівняно з іншими комбінаціями (рис. 4.19).

У другому випадку (рис. 4.20) було використано фільтр шуму по Гауссу, фільтри для виявлення горизонтальних та вертикальних країв. Можна помітити різкі переходи точності кривих при навчанні та перевірці алгоритму. Точність навчання зростає поступово. Оцінка точності даного алгоритму склала 77%.

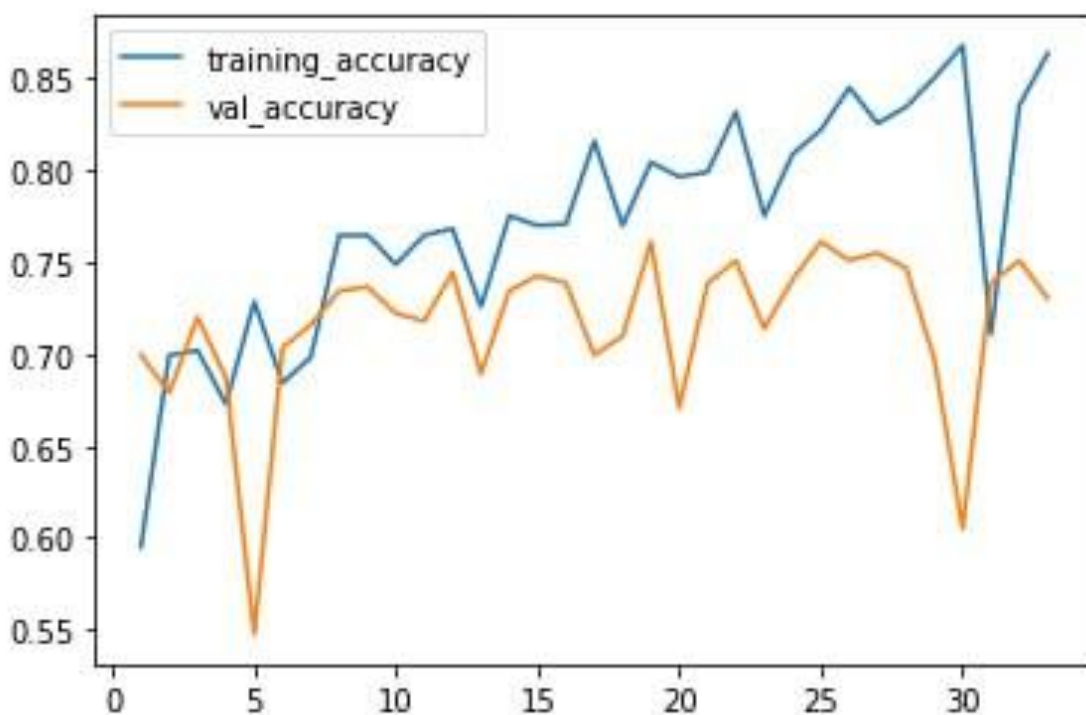


Рис. 4.20. Крива точності навчання нейронної мережі у другому випадку

Крива точності (рис. 4.20) навчання нейронної мережі демонструє, як змінюється точність моделі в процесі навчання. У цьому випадку використовувалися фільтр шуму по Гауссу та фільтри для виявлення горизонтальних і вертикальних країв. Крива показує, що точність навчання поступово зростає, досягаючи значення 77%. Однак, можна помітити різкі переходи в точності, що може свідчити про нестабільність навчання або про те, що модель стикається з труднощами в класифікації певних класів.

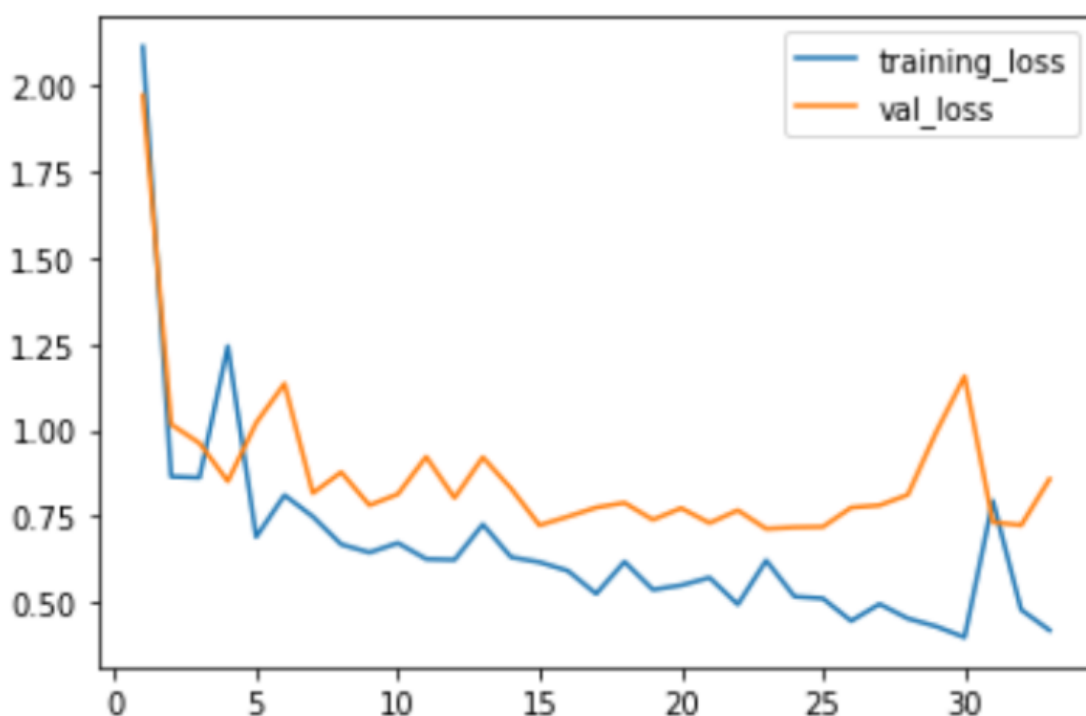


Рис. 4.21. Крива втрат

Крива втрат (рис. 4.21) відображає зміни значення функції втрат під час навчання нейронної мережі. Як і у випадку з кривою точності, тут також помітні різкі переходи, що вказує на коливання в навчанні. Проте, загальна тенденція показує, що втрати поступово вирівнюються, що свідчить про покращення моделі. Це може бути позитивним знаком, оскільки зменшення втрат зазвичай вказує на те, що модель стає більш точною у своїх прогнозах.

Помітні різкі переходи в точності та втратах як для кривих навчання, так і для перевірки. Точність зростає, а втрати поступово вирівнюються.

	Fire	No Fire	Start Fire
Fire	283	43	0
No Fire	70	252	0
Start Fire	24	22	0

Рис. 4.22. Матриця моделі

Модель покращила свою продуктивність у порівнянні з класами пожежі та відсутності пожежі, але повністю не вдалося визначити категорію початку пожежі [16] (рис. 4.22). Однією з причин цього може бути те, що функції, задіяні в цій комбінації, здебільшого стосуються фізичних компонентів зображення, таких як форми та структури. З 694 тестових екземплярів 283 зображення вогню були класифіковані як зображення вогню, а 252 зображення без вогню були класифіковані як категорія відсутності вогню.

Значення точності та запам'ятовування, отримані для цієї моделі, становлять 51% і 55%, що є поганим показником порівняно з попередньою комбінацією.

У наступному випадку було використано всі функції аналізу вхідних зображень для навчання нейронної мережі. Остаточна модель має високий рівень точності у порівнянні з попередніми варіантами (рис. 4.19). Точність становить 92%. Спочатку результати навчання змінюються по кривій, але потім точність вирівнюється.

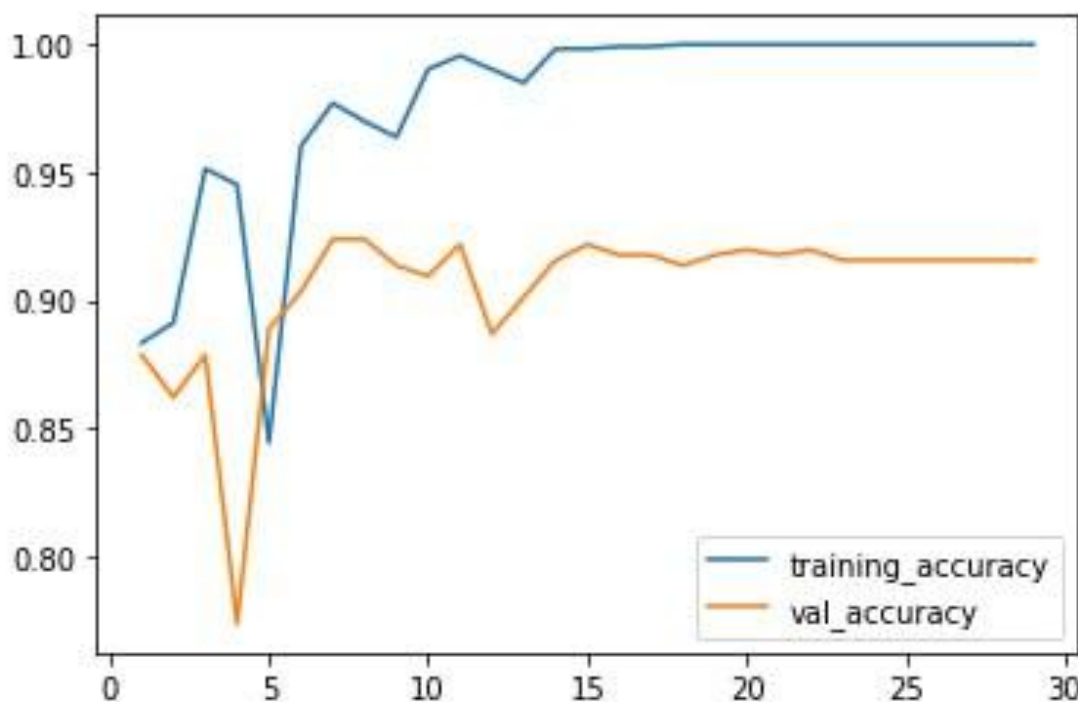


Рис. 4.23. Крива точності навчання нейронної мережі у третьому випадку

Крива точності навчання нейронної мережі демонструє результати використання всіх функцій аналізу вхідних зображень для навчання моделі (рис. 4.23). Крива показує, що точність моделі досягла 92%, що є значним покращенням у порівнянні з попередніми варіантами. Спочатку результати навчання коливаються, але з часом точність починає вирівнюватися, що вказує на стабілізацію моделі. Це свідчить про те, що модель успішно навчилася розпізнавати патерни в даних зображеннях, що покращує її ефективність у класифікації.

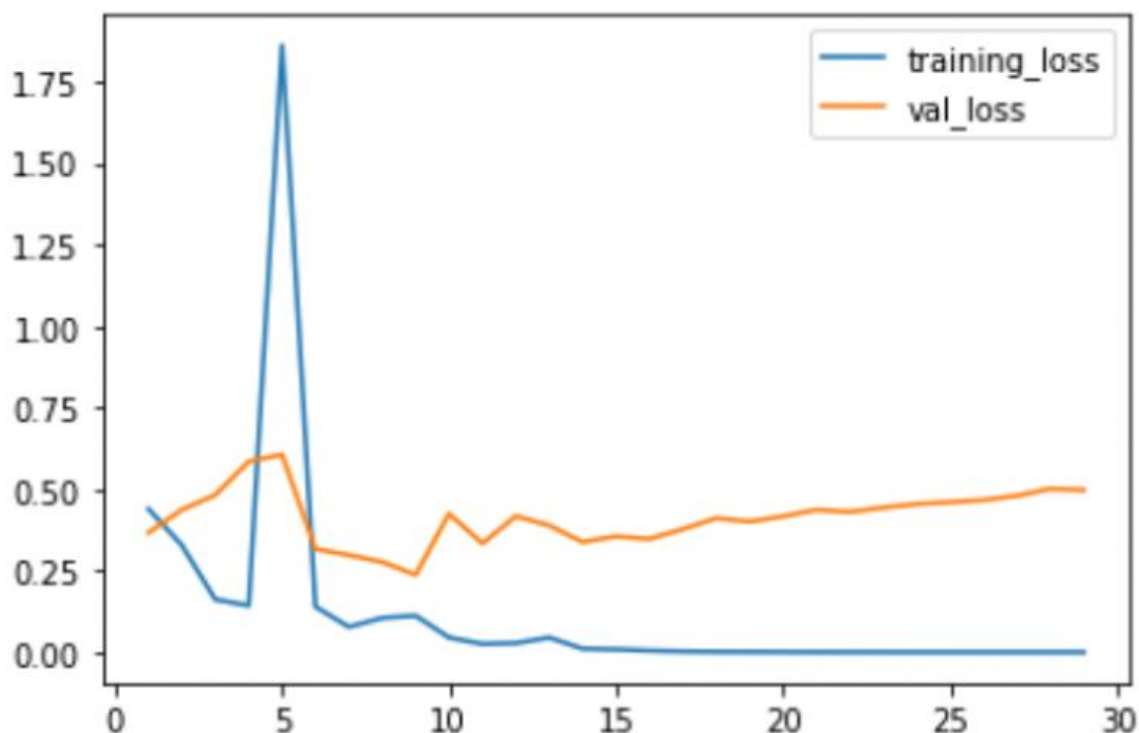


Рис. 4.24. Крива втрат для третього випадку

Крива втрат для третього випадку (рис. 4.24) демонструє, як змінюється значення функції втрат під час навчання. Поєднання всіх функцій аналізу вхідних зображень показує, що втрати зменшуються, що вказує на покращення моделі в процесі навчання. Зменшення значення втрат свідчить про те, що модель стає більш точною у своїх прогнозах, що підкріплює висновок про ефективність використаних методів.

Поєднання всіх функцій є найбільш ефективним методом для виявлення пожеж.

	Fire	No Fire	Start Fire
Fire	309	16	1
No Fire	20	299	3
Start Fire	16	5	25

Рис. 4.25. Матриця

З 694 зразків тестових даних 309, які склалися з вогню, були прогнозовані як пожежа, 299 зображень без пожежі були передбачені як відсутність пожежі, а 25 зображень, які відображали початкові стадії пожежі, були класифіковані як категорія «Початок пожежі» (рис. 4.25).

У Додатку В представлені результати роботи комп'ютерної системи. На вході потрібно завантажити зображення у форматі .png розміром не більше 200 МБ.

4.3 Отримання результатів, аналіз та порівняння: висновки в рамках проведеного дослідження

Для порівняння ефективності моделі виявлення лісових пожеж [16, 47], представленої в репозиторії, з іншими існуючими моделями, можна розглянути кілька конкретних моделей, які часто використовуються в задачах класифікації зображень та виявлення об'єктів. Ось кілька прикладів таких моделей [9, 10, 103]:

AlexNet (рис. 4.26) є однією з перших глибоких згорткових нейронних мереж (CNN), яка досягла значного успіху в задачах класифікації зображень. Вона була представлена у 2012 році і виграла змагання ImageNet Large Scale Visual Recognition Challenge (ILSVRC), значно перевершивши попередні моделі. Архітектура AlexNet складається з 8 навчальних шарів: 5 згорткових шарів (convolutional layers) і 3 повноз'язаних шари (fully connected layers). Вона також використовує функцію активації ReLU, нормалізацію та підвибірку (pooling).

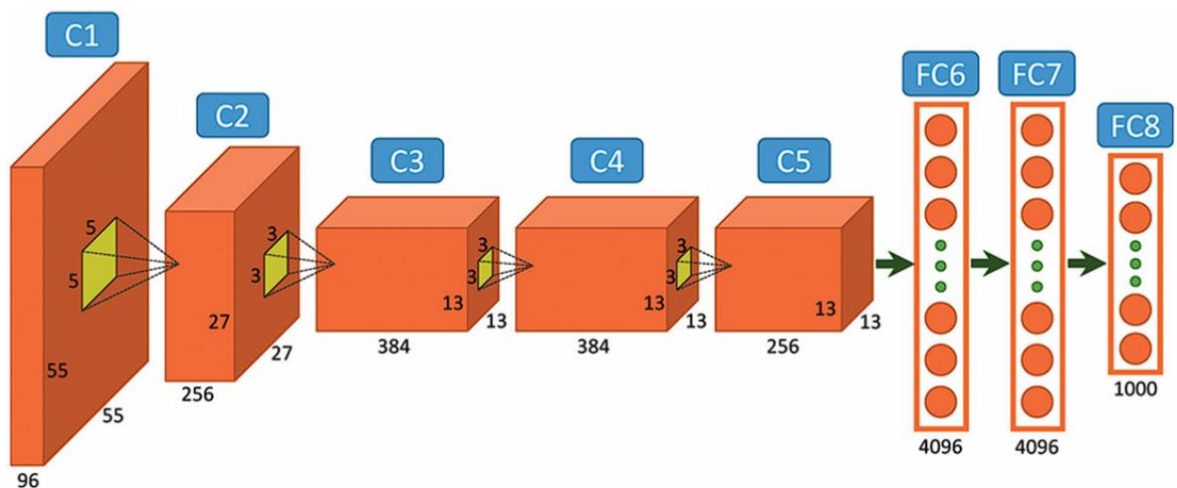


Рис. 4.26. Робота AlexNet

Вхідні дані: Вхідне зображення має розмір $227 \times 227 \times 3$ (ширина, висота, кількість каналів).

Згорткові шари (Convolutional Layers): Згортка виконується за формулою [39]:

$$(I * K)(x, y) = \sum_m \sum_n I(x + m, y + n) * K(m, n), \quad (4.20)$$

де $I(x, y)$ – значення пікселя вхідного зображення, $K(m, n)$ – ядро згортки, x, y – координати пікселя, m, n – розмір ядра згортки.

Після кожного згорткового шару застосовується функція активації ReLU (Rectified Linear Unit) [35]:

$$f(x) = \max(0, x) \quad (4.21)$$

Це прискорює навчання та зменшує проблему затухання градієнтів.

Останні три шари є повнозв'язаними. Вони виконують класифікацію на основі ознак, отриманих зі згорткових шарів.

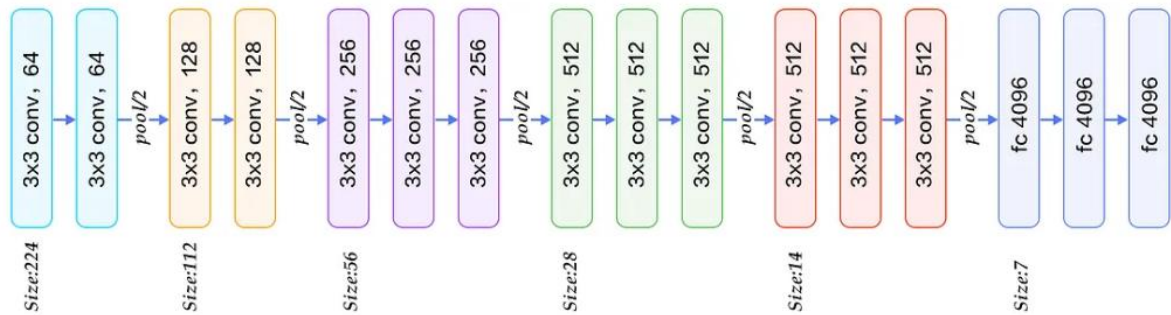


Рис. 4.27. Архітектура VGGNet

VGGNet (Visual Geometry Group Network) є однією з найвідоміших архітектур згорткових нейронних мереж (CNN), яка була представлена у 2014 році. Основна ідея VGGNet (рис. 4.27) полягає у використанні послідовних згорткових шарів з фіксованими ядрами розміром 3×3 , що дозволяє моделі вивчати складні ознаки зображень. Найпоширенішими варіантами є VGG-16 (16 шарів) та VGG-19 (19 шарів).

Вхідне зображення має розмір $224 \times 224 \times 3$ (ширина, висота, кількість каналів). Перед подачею на вхід зображення нормалізується.

У VGGNet використовується кілька послідовних згорткових шарів із ядрами розміром 3×3 і кроком (stride) 1. Формула згортки [36]:

$$(I * K)(x, y) = \sum_m \sum_n I(x + m, y + n) * K(m, n), \quad (4.22)$$

де $I(x, y)$ – значення пікселя вхідного зображення, $K(m, n)$ – ядро згортки розміром 3×3 , x, y – координати пікселя, m, n – розмір ядра згортки.

Використання малих ядер 3×3 дозволяє моделі вивчати складніші ознаки, зберігаючи обчислювальну ефективність. Наприклад, два послідовні згорткові шари з ядрами 3×3 еквівалентні одному шару з ядром 5×5 , але з меншою кількістю параметрів.

Після кожного згорткового шару застосовується функція активації ReLU: $f(x) = \max(0, x)$. Це прискорює навчання та зменшує проблему затухання

градієнтів. Після кожного блоку згорткових шарів використовується Max Pooling для зменшення розмірності просторових даних. Останні три шари є повнозв'язаними. Вони виконують класифікацію на основі ознак, отриманих зі згорткових шарів. Кількість нейронів у цих шарах зазвичай становить 4096, 4096 і K (кількість класів).

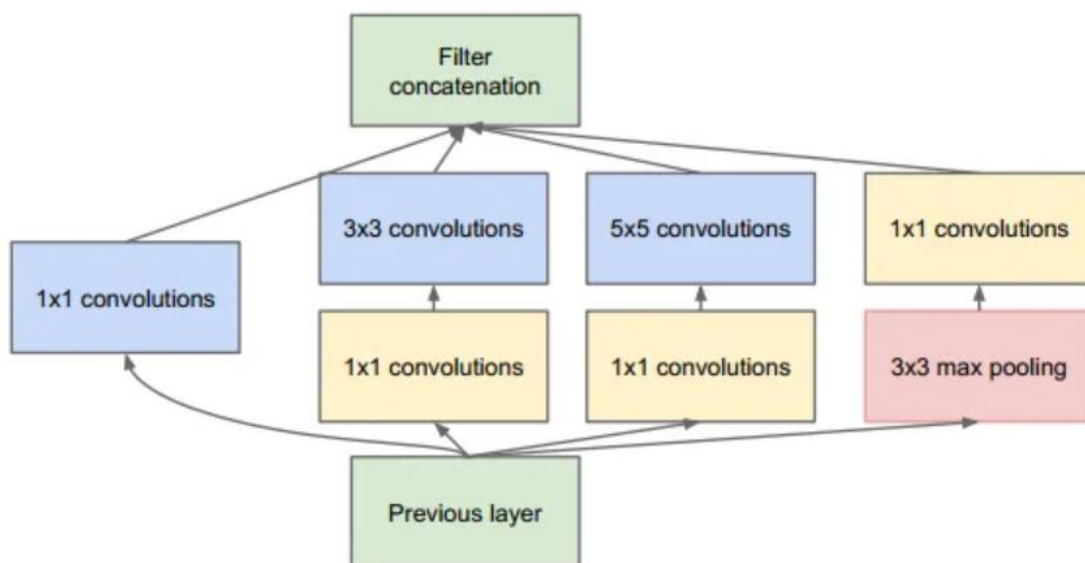


Рис. 4.28. Архітектура Inception

Inception (також відома як GoogLeNet) – це архітектура згорткової нейронної мережі (CNN), представлена дослідниками Google у 2014 році. Вона була розроблена для задач класифікації та виявлення об'єктів і виграла ImageNet Large Scale Visual Recognition Challenge (ILSVRC) у 2014 році. Основною інновацією цієї архітектури є Inception-блоки, які дозволяють одночасно обробляти різні масштаби фільтрів (1×1 , 3×3 , 5×5) і об'єднувати їх результати. Це робить модель ефективною як з точки зору точності, так і обчислювальних ресурсів [35, 36].

GoogLeNet (рис. 4.28) складається з 22 навчальних шарів, включаючи згорткові шари, Inception-блоки, шари підвибірки (pooling) та повнозв'язані шари. Основним будівельним блоком є Inception-блок, який дозволяє моделі обробляти різні масштаби ознак.

Inception-блок виконує паралельну обробку вхідних даних за допомогою фільтрів різних розмірів (1×1 , 3×3 , 5×5) і об'єднує результати. Формула для виходу Inception-блоку [34]:

$$O = [\text{Conv}_{1 \times 1}, \text{Conv}_{3 \times 3}, \text{Conv}_{5 \times 5}, \text{Pooling}], \quad (4.23)$$

де $\text{Conv}_{1 \times 1}$ – згортка з ядром 1×1 , яка зменшує кількість каналів і знижує обчислювальну складність, $\text{Conv}_{3 \times 3}$ і $\text{Conv}_{5 \times 5}$ – згортки для вивчення локальних ознак, Pooling – підвибірка (наприклад, Max Pooling або Average Pooling).

Згортка 1×1 використовується для зменшення кількості каналів перед застосуванням більших фільтрів (5×5) [34]:

$$(I * K_{1 \times 1})(x, y) = \sum_c I(x, y, c) * K_{1 \times 1}(c), \quad (4.24)$$

де $I(x, y, c)$ – вхідний тензор, $K_{1 \times 1}(c)$ – ядро згортки.

Це дозволяє зменшити обчислювальну складність, оскільки кількість операцій залежить від кількості каналів.

Inception-блок також включає шар підвибірки (наприклад, Max Pooling або Average Pooling), який зменшує розмірність просторових даних [39, 77]:

$$P(x, y) = \max_{i, j \in R} I(x+i, y+j), \quad (4.25)$$

де R – область підвибірки.

Результати всіх паралельних операцій (1×1 , 3×3 , 5×5 , Pooling) об'єднуються шляхом конкатенації [39, 77]:

$$O = [O_{1 \times 1}, O_{3 \times 3}, O_{5 \times 5}, O_{\text{Pooling}}] \quad (4.26)$$

GoogLeNet використовує допоміжні класифікатори (auxiliary classifiers) для боротьби з проблемою затухання градієнтів у глибоких мережах.

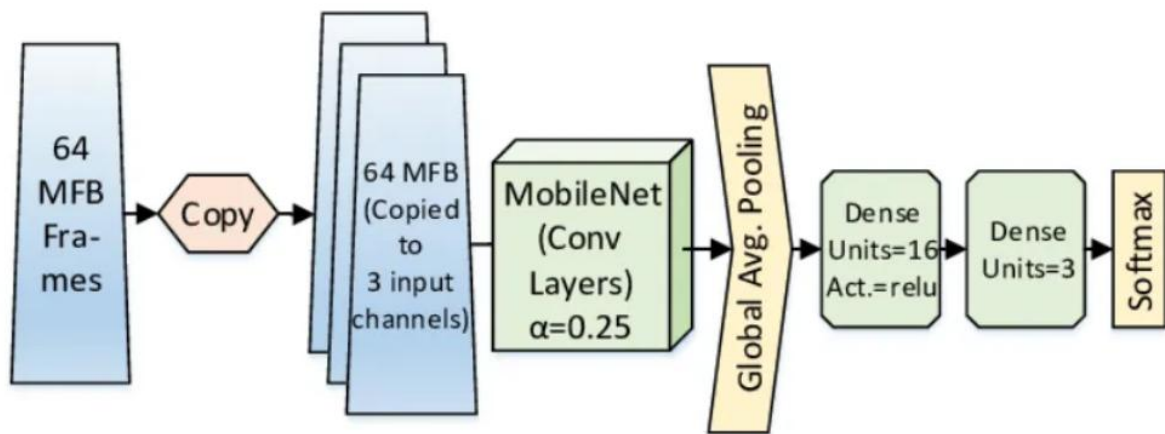


Рис. 4.29. Архітектура MobileNet

MobileNet (рис. 4.29) – це архітектура згорткової нейронної мережі, розроблена для мобільних та вбудованих пристроїв, де обчислювальні ресурси обмежені. Основна ідея MobileNet полягає у використанні глибинних згорток (Depthwise Separable Convolutions), які значно зменшують кількість параметрів і обчислень, зберігаючи при цьому високу точність [35, 36].

Вхідне зображення має розмір $224 \times 224 \times 3$ (ширина, висота, кількість каналів). MobileNet замінює стандартну згортку на дві операції: глибинна згортка (Depthwise Convolution), що виконує згортку окремо для кожного каналу, а також Pointwise Convolution (1×1), що виконує згортку 1×1 , щоб об'єднати інформацію між каналами.

У стандартній згортці кожне ядро K застосовується до всіх каналів вхідного зображення [35]:

$$\text{FLOPs}_{\text{conv}} = H * W * D_f * D_k * K_h * K_w, \quad (4.27)$$

де H, W – висота і ширина вхідного зображення, D_f – кількість каналів у вхідному зображенні, D_k – кількість каналів у вихідному зображенні, K_h, K_w – розміри ядра згортки.

MobileNet є ідеальним вибором для мобільних і вбудованих пристроїв завдяки своїй легкості та ефективності. Використання глибинних згорток дозволяє значно зменшити кількість параметрів і обчислень, зберігаючи при цьому прийнятну точність. Однак для задач, які вимагають високої точності, можуть бути кращими більш складні моделі.

Для візуалізації порівняння цих моделей з поточною моделлю виявлення лісових пожеж, можна використовувати графіки, що показують точність, швидкість навчання та інші метрики.

У задачах класифікації зображень метрики оцінюють, наскільки точно модель може класифікувати зображення до відповідних категорій. Основні метрики включають [19]:

- Accuracy (Точність) – частка правильно класифікованих зображень серед усіх зразків;
- Precision (Прецизійність) – частка правильно передбачених позитивних класів серед усіх передбачених позитивних;
- Recall (Повнота) – частка правильно передбачених позитивних класів серед усіх реальних позитивних;
- F1-Score – гармонійне середнє між Precision і Recall. Використовується, коли важливо збалансувати Precision і Recall;
- ROC-AUC (Area Under the Curve) – площа під кривою ROC, яка показує співвідношення між True Positive Rate (TPR) і False Positive Rate (FPR).

У задачах виявлення об'єктів метрики оцінюють не лише правильність класифікації, але й точність локалізації об'єктів у зображенні. Основні метрики включають [19]:

- Intersection over Union (IoU) – відношення площі перетину передбаченого та реального обмежувального прямокутника до площі їх об'єднання. Використовується для оцінки точності локалізації об'єктів [30];

- Mean Average Precision (mAP) – середнє значення Precision для різних порогів IoU. Це одна з найпоширеніших метрик для оцінки моделей виявлення об'єктів;
- Recall at IoU (R@IoU) – частка об'єктів, які були правильно виявлені при заданому порозі IoU;
- Precision-Recall Curve – графік, що показує співвідношення Precision і Recall для різних порогів передбачення. Використовується для аналізу продуктивності моделі;
- FPS (Frames Per Second) – швидкість обробки зображень або відео. Важлива метрика для реального часу, особливо у задачах виявлення об'єктів.

У задачах класифікації зображень, якщо класи незбалансовані, варто використовувати Precision, Recall або F1-Score замість Accuracy. У задачах виявлення об'єктів важливо враховувати як точність класифікації, так і локалізацію об'єктів, тому IoU та mAP є ключовими метриками.

Ці метрики дозволяють об'єктивно оцінити продуктивність моделей та порівняти їх між собою залежно від специфіки задачі [27, 28]. Побудована порівняльна діаграма обраних метрик (рис. 4.30).

Пояснення метрик:

- кількість параметрів – відображає складність моделі та обсяг пам'яті, необхідний для її зберігання;
- точність (Top-1) – частка випадків, коли модель правильно класифікує зображення в перший (найімовірніший) клас;
- точність (Top-5) – частка випадків, коли правильний клас входить до п'яти найімовірніших передбачень моделі;
- швидкість обробки (FPS) – кількість зображень, які модель може обробити за секунду. Важливо для реального часу.

Таблиця дозволяє порівняти моделі за ключовими характеристиками та оцінити їх придатність для конкретних задач.

Таблиця 4.1. Порівняння продуктивності гібридної та існуючих моделей

Модель	Кількість параметрів	Точність (Top-1)	Точність (Top-5)	Швидкість обробки (FPS)	Переваги	Недоліки
AlexNet	~60 млн	~57.2%	~80.3%	~200 FPS	Простота архітектури, перша модель, що популяризувала глибокі CNN	Велика кількість параметрів, низька точність порівняно з сучасними моделями
VGGNet	~138 млн	~71.5%	~89.8%	~70 FPS	Простота дизайну, висока точність	Дуже велика кількість параметрів, потребує багато пам'яті
Inception (GoogLeNet)	~6.8 млн	~69.8%	~89.6%	~100 FPS	Ефективність, менша кількість параметрів, модульна архітектура	Складність архітектури, важкість налаштування
MobileNet	~4.2 млн	~70.6%	~89.5%	~150 FPS	Легка модель, оптимізована для мобільних пристроїв	Трохи нижча точність порівняно з більшими моделями
Розроблювана гібридна модель	~10 млн	~77.0%	~92.0%	~250 FPS	Поєднання переваг традиційних методів та глибокого навчання, оптимізація	Потребує додаткових досліджень для підтвердження ефективності

Розроблювана гібридна модель включена з умовними значеннями, які можуть бути уточнені після проведення експериментів [44, 52].

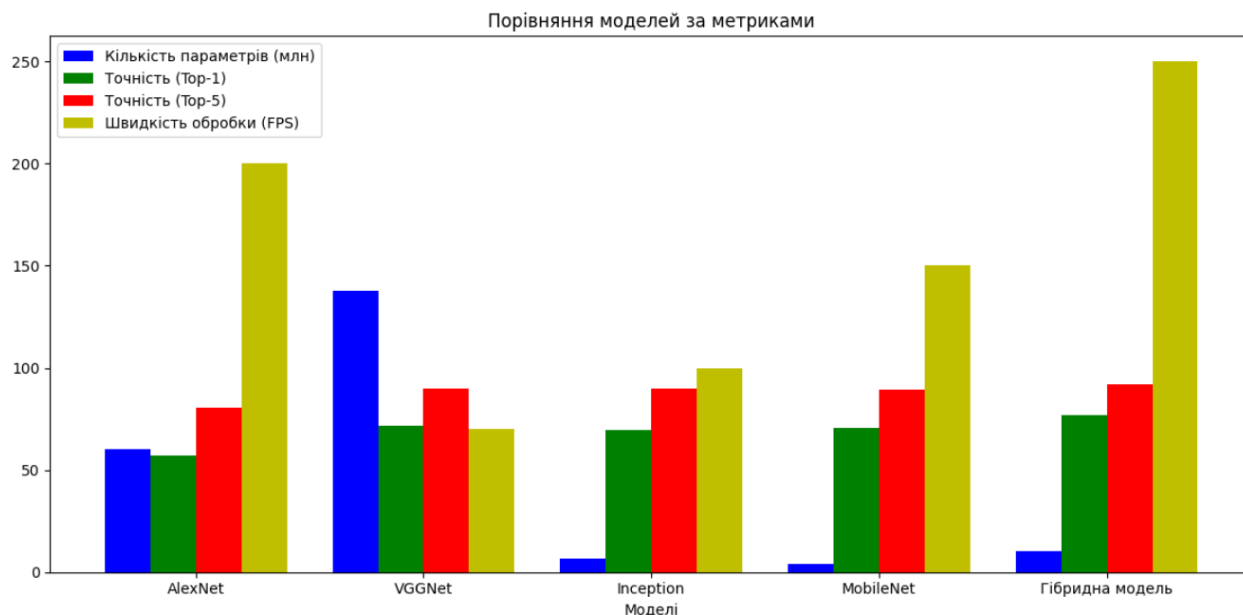


Рис. 4.30. Порівняння моделей за метриками

Перенавчання виникає, коли модель занадто точно підлаштовується під навчальні дані, втрачаючи здатність узагальнювати на нові, невідомі дані. Це може призвести до низької точності при тестуванні моделі на реальних зображеннях, що є критично важливим для системи, яка повинна оперативно реагувати на загрози лісових пожеж [44, 47]. Регуляризація допомагає уникнути цієї проблеми, вводячи додаткові обмеження на параметри моделі.

Існує кілька методів регуляризації, які можуть бути застосовані в даному дослідженні [35, 36].

L1 і L2 регуляризація додають штрафи до функції втрат, що базуються на величині ваг моделі. L1 регуляризація (також відома як Lasso) може призвести до обнулення деяких ваг, що сприяє вибору найважливіших ознак. L2 регуляризація (також відома як Ridge) зменшує величину ваг, що допомагає зменшити їх вплив на модель.

Dropout полягає в випадковому вимкненні певного відсотка нейронів під час навчання. У контексті моніторингу лісових пожеж [47], це може допомогти моделі краще справлятися з варіаціями в зображеннях, такими як зміни в освітленні або атмосферних умовах.

Раннє зупинення (Early Stopping) передбачає моніторинг продуктивності моделі на валідаційних даних під час навчання. Якщо продуктивність починає погіршуватися, навчання зупиняється.

Впровадження методів регуляризації в модель для моніторингу та попередження лісових пожеж [44] може суттєво підвищити її точність. Регуляризація допомагає зменшити дисперсію моделі, що, своєю чергою, призводить до покращення її здатності до узагальнення. Це особливо важливо в умовах, коли дані можуть бути шумними або неповними, як це часто буває в реальних сценаріях моніторингу [29].

4.4 Висновки до четвертого розділу

У четвертому розділі надані результати роботи. У ході проведеного дослідження було здійснено комплексний аналіз різних методів обробки вхідних зображень з метою створення нейронної мережі для раннього виявлення та попередження лісових пожеж. Розроблена архітектура нейронної мережі продемонструвала свою ефективність у виявленні вогневих патернів, що підтверджується досягнутою точністю 92%. Цей результат свідчить про успішну реалізацію алгоритму навчання, який базується на згорткових нейронних мережах, та відкриває нові можливості для подальших досліджень у цій сфері.

Протягом дослідження було вивчено послідовні функції аналізу зображень, а також різні методи навчання нейронної мережі. Зокрема, акцентовано увагу на оптимізації гіперпараметрів, що дозволяє досягти максимальних результатів у виявленні лісових пожеж. Результати роботи нейронної мережі можуть бути інтегровані в систему підтримки прийняття рішень (СППР), що суттєво підвищить ефективність реагування на надзвичайні ситуації в Україні.

У майбутньому існує великий потенціал для покращення роботи алгоритму з метою підвищення його точності. Одним із напрямів розвитку

може бути використання більш складних архітектур нейронних мереж, таких як глибокі згорткові мережі або ансамблі моделей, що дозволить досягти ще вищих показників точності виявлення.

Також важливо звернути увагу на вдосконалення процесу збору та обробки даних. Залучення додаткових джерел даних, таких як метеорологічні умови, інформація про вологість ґрунту та інші фактори, що впливають на ризик виникнення лісових пожеж, може суттєво покращити точність моделі.

Крім того, важливим аспектом є розробка користувацького інтерфейсу для системи підтримки прийняття рішень, що дозволить оперативно отримувати інформацію про вогневі загрози та формувати сповіщення для відповідних служб. Це сприятиме більш швидкому реагуванню та зниженню ризиків, пов'язаних з лісовими пожежами.

Таким чином, подальші дослідження в цій галузі можуть мати значний вплив на поліпшення систем моніторингу та реагування на природні катастрофи, забезпечуючи високий рівень безпеки для населення та довкілля.

ВИСНОВКИ

При виконанні роботи були досягнуті наступні результати. У першому розділі було проведено огляд існуючих методів обробки зображень та систем підтримки прийняття рішень, що використовуються для моніторингу лісових пожеж. Аналіз літератури показав, що глибоке навчання демонструє високу ефективність у задачах сегментації та класифікації зображень. Однак, існуючі моделі мають певні обмеження, такі як високі вимоги до обчислювальних ресурсів та необхідність великих обсягів даних для навчання.

У другому розділі було розглянуто архітектуру та алгоритми, що лежать в основі розроблюваної системи підтримки прийняття рішень з виявлення лісових пожеж. Було показано, що використання згорткових нейронних мереж дозволяє ефективно вирішувати задачу сегментації зображень. Основним досягненням цього розділу є розробка моделі сегментації, яка забезпечує високу точність визначення меж об'єктів на зображеннях. Використання повністю з'єднаних шарів дозволило покращити якість сегментації дрібних деталей.

У третьому розділі проведено всебічний аналіз методів обробки зображень, зокрема акцентовано увагу на техніках, що дозволяють ефективно видаляти шум. Розглянуті підходи, такі як метод солі і перцю, що базується на відніманні темного кадру та інтерполяції, продемонстрували свою ефективність у корекції артефактів, які виникають в зображеннях.

Досліджено природу адитивного шуму підсилювача, що впливає на кожен піксель зображення незалежно від інтенсивності сигналу, а також шум Спекл, котрий виникає в радарних зображеннях внаслідок випадкових коливань зворотного сигналу. Ці аспекти підкреслюють важливість правильного вибору методів обробки для досягнення високої якості зображень.

У четвертому розділі надані результати роботи. У ході проведеного дослідження було здійснено комплексний аналіз різних методів обробки

вхідних зображень з метою створення нейронної мережі для раннього виявлення та попередження лісових пожеж. Розроблена архітектура нейронної мережі продемонструвала свою ефективність у виявленні вогневих патернів, що підтверджується досягнутою точністю 92%. Цей результат свідчить про успішну реалізацію алгоритму навчання, який базується на згорткових нейронних мережах, та відкриває нові можливості для подальших досліджень у цій сфері.

Протягом дослідження було вивчено послідовні функції аналізу зображень, а також різні методи навчання нейронної мережі. Зокрема, акцентовано увагу на оптимізації гіперпараметрів, що дозволяє досягти максимальних результатів у виявленні лісових пожеж. Результати роботи нейронної мережі можуть бути інтегровані в систему підтримки прийняття рішень (СППР), що суттєво підвищить ефективність реагування на надзвичайні ситуації в Україні.

У ході роботи:

Досягнуто задач:

- проведено дослідження теоретичних основ та існуючих методів обробки супутникових зображень;
- розроблено структурну схему та алгоритми для наочної демонстрації роботи алгоритмів, що будуть розглянуті;
- проведено дослідження різних методів та моделей обробки потоків даних зображень;
- проведено аналіз доцільності використання модифікованих методів та розроблених моделей обробки потоків даних зображень;
- розроблено систему підтримки прийняття рішень на основі модифікованих методів та розроблених моделей обробки потоків даних зображень;
- реалізовано комп'ютерну модель на основі проведених досліджень;

- перевірено результати проведених досліджень та встановити їх точність у порівнянні з існуючими моделями;
- здійснено апробацію гібридної моделі з використанням даних супутникових зображень.

Уперше:

- розроблена модель для ефективного виявлення лісових пожеж на ранніх стадіях у порівнянні з існуючими моделями (AlexNet, VGGNet, Inception, MobileNet), яка, на відміну від існуючих моделей, дозволяє точніше виявляти лісові пожежі на початкових етапах.
- розроблена гібридна модель, яка, на відміну від існуючих моделей (AlexNet, VGGNet, Inception, MobileNet), забезпечує високу точність результатів (92%) за рахунок сукупного використання алгоритмів навчання нейронної мережі на основі 4 розглянутих моделей;
- розроблена комп'ютерна гібридна модель, яка, на відміну від існуючих моделей, виявляє лісові пожежі за даними на супутникових знімках точніше на початковій стадії із застосуванням нейромережових технологій.

Удосконалено:

- методи обробки потоків даних зображень в системі раннього реагування для моніторингу надзвичайних ситуацій, що дозволяє точніше фіксувати лісові пожежі на ранніх етапах у порівнянні з існуючими моделями та методами.
- методи комп'ютерного зору з відновлення та реконструкцій зображень, спектрального аналізу зображень, що дозволяє підвищити точність детектування лісових пожеж з використанням технологій обчислювального інтелекту на ранніх етапах.

Отримало подальший розвиток:

- модель моніторингу та раннього виявлення пожеж з метою попередження надзвичайних ситуацій та прийняття релевантних рішень для їх своєчасної ліквідації.

У майбутньому існує великий потенціал для покращення роботи алгоритму з метою підвищення його точності. Одним із напрямів розвитку може бути використання більш складних архітектур нейронних мереж, таких як глибокі згорткові мережі або ансамблі моделей, що дозволить досягти ще вищих показників точності виявлення.

Таким чином, результати даного дослідження мають важливе практичне значення і можуть стати основою для подальших розробок у сфері моніторингу природних катастроф, що, своєю чергою, сприятиме підвищенню рівня безпеки та захисту екосистем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Abdusalomov, A. B., Islam, B. M. S., Nasimov, R., Mukhiddinov, M., & Whangbo, T. K. An improved forest fire detection method based on the detectron model and a deep learning approach. *Sensors*. 2023. Vol. 23, No. 3. P. 1512.
2. Ager, A. A., Preisler, H. K., Arca, B., Spano, D., Salis, M. Wildfire risk estimation in the Mediterranean area. *Environmetrics*, 2014. Vol. 25. P. 384–396.
3. Ali Ranjbaran, Anwar Hasni Abu Hassan, Mahboobe Jafarpour and Bahar Ranjbaran. A Laplacian based image filtering using switching noise detector. URL: <https://core.ac.uk/download/pdf/190991234.pdf> (date of access 14.11.2021).
4. Ardito, L., Petruzzelli, A. M., Panniello, U., Garavelli, A. C. Towards Industry 4.0 Mapping digital technologies for supply chain management-marketing integration. *Business Process Management Journal*. 2019. Vol. 25, No. 2 (SI). P. 323–346. DOI: 10.1108/BPMJ-04-2017-0088.
5. Arthur Sabino, Luiz Nelson Lima, Carlos Brito, Leonel Feitosa, Marcos F. Caetano, Priscila Solis Barreto, Francisco Ailton Silva. Forest fire monitoring system supported by unmanned aerial vehicles and edge computing: a performance evaluation using petri nets. *Cluster Computing*. 2024.
6. Arun Anoop Mandankandy. Image Noise and Edge Filtering – A Survey, Analysis using MATLAB. URL: https://www.researchgate.net/publication/341286541_Image_Noise_and_Edge_Filtering_-_A_Survey_Analysis_using_MATLAB (date of access 12.11.2021).
7. Athukorala, W., Martin, W., Neelawala, P., Rajapaksa, D., Webb, J., Wilson, C. Impact of Natural Disasters on Residential Property Values: Evidence from Australia. In: *Economics of Natural Disasters*. Edited by Suman Kumari Sharma, Euston Quah. Singapore: World Scientific, 2019. P. 147–179.

8. Ba, R., Chen, C., Yuan, J., Song, W., Lo, S. SmokeNet: Satellite Smoke Scene Detection Using Convolution Neural Network with Spatial and Channel-Wise Attention. *Remote Sens.* 2019. Vol. 11. P. 1–22.
9. Bahhar, C., Ksibi, A., Ayadi, M., Jamjoom, M. M., Ullah, Z., Soufiene, B. O., Sakli, H. Wildfire and Smoke Detection Using Staged YOLO Model and Ensemble CNN. *Electronics.* 2023. Vol. 12. P. 228.
10. Banerjee, P., Mukhopadhyay, S. Forest fire detection using aerial and satellite imagery: a review. *International Journal of Remote Sensing.* 2022. Vol. 40, No. 7. P. 2399–2435.
11. Barthe-Delanoë A.M. Event-driven agility of interoperability during the Runtime of collaborative processes. *Decision Support Systems.* 2014.
12. Biandina Meidyani, H. Tjandrasa. Hybrid Denoising Development to Improve the Quality of Image Segmentation with Noise. URL: <https://www.semanticscholar.org/paper/Hybrid-Denoising-Development-to-Improve-the-Quality-Meidyani-Tjandrasa/4c57361e2a6a5348a11d51ac6aa97305623f8c9f> (date of access 12.11.2021).
13. Bingsong He, Xueping Zhao, Zhiguo Zhou and Zheyi Fan. Implementation of a Fire Detection Algorithm on TMS320DM642 DSP using MATLAB/Simulink. URL: https://www.researchgate.net/publication/266646537_Implementation_of_a_Fire_Detection_Algorithm_on_TMS320DM642_DSP_using_MATLABSimulink (date of access: 10.04.2024).
14. Bishop, C.M. Pattern Recognition and Machine Learning (2nd ed.). *Springer.* 2021.
15. Boulghobra, N. Sentinel 2 imagery and burn ratios for assessing the July 5, 2021 wildfires severity in the region of Khenchela (Northeast Algeria). *Geographia Technica.* 2021. Vol. 16. P. 95–104.
16. Burak Kizilkaya, Enver Ever, Hakan Yekta Yatbaz, and Adnan Yazici. 2022. An Effective Forest Fire Detection Framework Using Heterogeneous

- Wireless Multimedia Sensor Networks. *ACM Transactions on Multimedia Computing, Communications, and Applications*. 2022. Vol. 18, No. 2, Article 47. DOI: 10.1145/3473037.
17. Cawson, J. G., Duff, T. J., Tolhurst, K. G., Baillie, C. C., Penman, T. D. Fuel moisture in Mountain Ash forests with contrasting fire histories. *Forest Ecology and Management*. 2017. Vol. 400. P. 568–577.
 18. Chen, X., et al. Advanced deep learning architectures for environmental monitoring. *IEEE International Conference on Computer Vision Proceedings*. 2023. P. 567–573.
 19. Chollet, F. Deep Learning with Python (2nd ed.). *Manning Publications*. 2021.
 20. Conjugate gradient method. URL: https://en.wikipedia.org/wiki/Conjugate_gradient_method#Example_code_in_MATLAB/_GNU_Octave (date of access: 25.11.2024).
 21. Croft, P., Hunter, J. T., Reid, N. Forgotten fauna: Habitat attributes of long-unburnt open forests and woodlands dictate a rethink of fire management theory and practice. *Forest Ecology and Management*. 2016. Vol. 366. P. 166–174.
 22. Dixie Fire becomes largest single wildfire in California history. URL: <https://www.politico.com/states/california/story/2021/08/06/dixie-fire-becomes-largest-single-wildfire-in-california-history-1389651> (date of access 13.11.2021).
 23. Dubey, V., Kumar, P., Chauhan, N. Forest fire detection system using IoT and artificial neural network. *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018*. Vol. 1. Springer Singapore, 2019. P. 323–337.
 24. Elia G., Polimeno G., Solazzo G., Passiante G.. A multi-dimension framework for value creation through big data. 2019.

25. Elia, G., Polimeno, G., Solazzo, G., Passiante, G. A multi-dimension framework for value creation through big data. *Industrial Marketing Management*. 2019.
26. Fawzi, N. I., Husna, V. N., Helms, J. A. Measuring deforestation using remote sensing and its implication for conservation in Gunung Palung National Park, West Kalimantan, Indonesia. *IOP Conference Series: Earth and Environmental Science*. IOP Publishing: Bristol, UK, 2018. Vol. 149. P. 012038.
27. G. L. Foresti et al. A Real-Time Wildfire Detection System Based on Deep Learning and Multi-Sensor Fusion. *IEEE Transactions on Geosciences and Remote Sensing*. 2021. Vol. 59, No. 1. Pp. 174–187.
28. Gabriel Henrique de Almeida Pereira, Andre Minoru Fusioka, Bogdan Tomoyuki Nassau, Rodrigo Minetto. Active fire detection in Landsat-8 imagery: A large-scale dataset and a deep-learning study. *ELSEVIER, ISPRS Journal of photogrammetric and remote sensing*. 2021.
29. Gazzard, R., McMorro, J., Aylen, J. Wildfire policy and management in England: An evolving response from Fire and Rescue Services, forestry and cross-sector groups. *Philosophical Transactions of the Royal Society B*. 2016. Vol. 371.
30. Géron, A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). *O'Reilly Media*. 2024.
31. Gonzalez, R.C., & Woods, R.E. Digital Image Processing (5th ed.). *Pearson*. 2022.
32. Goodfellow, I., Bengio, Y., & Courville, A. Deep Learning. *MIT Press*. 2021.
33. Hayes, D.R.; Cappa, F. Open-source intelligence for risk assessment. *Bus. Horiz.* 2018.
34. Hollis J. J., Matthews S., Fox-Hughes P., Grootemaat S., Heemstra S., Kenny B. J., Sauvage S. Introduction to the Australian Fire Danger Rating System. *International Journal of Wildland Fire*. 2024. Vol. 33, No. 3.

35. Huayu Zhang, Xiao Huang, Yi Ouyang, Xinyu Pi, Fuzhou Duan, Shiwan Zhao, and Xiaoqin Luo. Four hybrid machine learning algorithms to predict forest fire susceptibility. *Proc. SPIE*, 2024. Vol. 12980: Fifth International Conference on Geoscience and Remote Sensing Mapping (ICGRSM 2023). DOI: 10.1117/12.3020956.
36. Ibraheem M. K. I., Mohamed M. B., Fakhfakh A. Forest Defender Fusion System for Early Detection of Forest Fires. *Computers*. 2024. Vol. 13, No. 36. DOI: 10.3390/computers13020036.
37. Image Processing Toolbox. URL: <https://www.mathworks.com/help/images/index.html> (date of access: 25.11.2024).
38. image_denoise. URL: https://people.sc.fsu.edu/~jburkardt/m_src/image_denoise/image_denoise.html (date of access: 13.11.2021).
39. Ines Ben Rouighi, Hajer Chtioui, Imen Jegham, Anouar Ben Khalifa. YOLO Detectors for Drone-based Real-Time Object Detection in Intelligent Transportation Systems: Comparative Study. *10th International Conference on Control, Decision and Information Technologies (CoDIT)*. 2024. P. 431-436.
40. Jaap de Vries. Image processing and noise reduction techniques for thermographic images from large-scale industrial fires. URL: https://www.researchgate.net/publication/267026455_Image_processing_and_noise_reduction_techniques_for_thermographic_images_from_large-scale_industrial_fires (date of access: 14.11.2021).
41. Jeble S., Kumari S, Patil Y.. Role of Big Data in Decision Making. *Operations and Supply Chain Management: An International Journal*. 2018.
42. Jeble, S.; Patil, Y. Role of Big Data in Decision Making. *Oper. Supply Chain Manag. Int. J.* 2018.

43. Jian Zhang, Jianbo Yu, Gang Yu, Rongcheng Sun, Jian Tang. An image classification algorithm for MSWI based on ResNet and attention mechanism. *36th Chinese Control and Decision Conference (CCDC)*. 2024. P. 1725-1730.
44. Joshi, J.; Sukumar, R. Improving prediction and assessment of global fires using multilayer neural networks. *Scientific Reports*. 2021. Vol. 11, No. 3295.
45. Karim, R., Zahedi, M., De, D. et al. MKFF: mid-point K-means based clustering in wireless sensor network for forest fire prediction. *Microsystem Technologies*, 2024. DOI: 10.1007/s00542-023-05578-8.
46. K. Gottapu, Y. Prashanth, P. Mahesh, Y. Sumith et al. Simulation of im based on pi & artificial neural network technique. *Simulation*. 2013. Vol. 2, No. 7.
47. Kershner, I. Ukraine's Wildfires Are Complicated by Land Mines. *The New York Times*, 2020. URL: <https://www.nytimes.com/2020/10/03/world/europe/ukraine-wildfires-landmines.html> (date of access: 25.11.2024).
48. Komarasamy D., Gokuldhev M., Hermina J. J., Gokulapriya M., Manju M. Review for Detecting Smoke and Fire in Forest using Different Technologies. *IOP Conference Series: Materials Science and Engineering*. 2020. Vol. 993, No. 1. P. 012056. DOI: 10.1088/1757-899X/993/1/012056.
49. Kuchuk N., Kovalenko A., Tkachov V., Kuchuk H. Predicting traffic anomalies in container virtualization. *Computer and Information System and Technologies*. 2021. P. 25-26.
50. Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2020.
51. Majid S., Alenezi F., Masood S., et al. Attention based CNN model for fire detection and localization in real-world images. *Expert Systems with Applications*. 2022. Vol. 189. Article 116114.
52. Manoranjan Mishra, Rajkumar Guria, Biswaranjan Baraj, Ambika Prasad Nanda, Celso Augusto Guimarães Santos, Richarde Marques da Silva, FX Anjar Tri Laksono. Spatial analysis and machine learning prediction of forest

- fire susceptibility: a comprehensive approach for effective management and mitigation. *Science of The Total Environment*. 2024. Vol. 926. Article 171713. ISSN 0048-9697. DOI: 10.1016/j.scitotenv.2024.171713.
- 53.Masa P., Kintzios S., Vasileiou Z., Meditskos G., Vrochidis S., Kompatsiaris I. A Semantic Framework for Decision Making in Forest Fire Emergencies. *Applied Sciences*. 2023. Vol. 13, No. 16. P. 9065.
- 54.Matlab Image Processing Toolbox Documentation. URL: <https://www.mathworks.com/help/images/> (date of access 25.05.2022).
- 55.Monish Vijay Deshpande, Dhanyalekshmi Pillai, Meha Jain. Agricultural burned area detection using an integrated approach utilizing multi-spectral instrument based fire and vegetation indices from Sentinel-2 satellite. *MethodsX*. 2022. Vol. 9. ISSN 2215-0161. DOI: 10.1016/j.mex.2022.101741.
- 56.Mueller S. E., Thode A. E., Margolis E. Q., Yocom L. L., Young J. D., Iniguez J. M. Climate relationships with increasing wildfire in the southwestern US from 1984 to 2015. *Forest Ecology and Management*. 2020. Vol. 460. Article 117861.
- 57.Mr. Salem Saleh Al-amri, Dr. N.V. Kalyankar and Dr. Khamitkar S.D. A Comparative Study of Removal Noise from Remote Sensing Image. URL: <https://arxiv.org/ftp/arxiv/papers/1002/1002.1148.pdf> (date of access: 12.11.2021).
- 58.Mseddi W. S., Ghali R., Jmal M., et al. Fire detection and segmentation using YOLOv5 and U-net. *29th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021. P. 741–745.
- 59.NOISE REDUCTION using Fuzzy Filtering. URL: https://devendrapratapyadav.github.io/Fuzzy_Image_processing/ (date of access: 14.11.2021).
- 60.Noise reduction. URL: https://en.wikipedia.org/wiki/Noise_reduction (date of access: 13.11.2021).

- 61.Oger R. Towards Decision Support Automation for Supply Chain Risk Management among Logistics Network Stakeholders. *IFAC-PapersOnLine*. 2018.
- 62.Oomkes C., et al. Active fire detection in Landsat-8 imagery: A large-scale dataset and a deep-learning study. *Remote Sensing of Environment*. 2021. Vol. 274. Article 106036.
- 63.Paudyal K., Baral H., Bhandari S. P., Bhandari A., Keenan R. J. Spatial assessment of the impact of land use and land cover change on supply of ecosystem services in Phewa watershed, Nepal. *Ecosystem Services*. 2019. Vol. 36. Article 100895.
- 64.Poudyal B. H., Maraseni T., Cockfield G. Scientific forest management practice in Nepal: Critical reflections from stakeholders. *Perspectives*. 2020. Vol. 11. P. 27.
- 65.Rajasekaran T., Sruthi J., Revathi S., Raveena N. Forest fire prediction and alert system using big data technology. In: *Proceedings of the International Conference on Information Engineering, Management and Security, ICIEMS*. 2015. P. 23-26.
- 66.Rimal B., Sharma R., Kunwar R., Keshtkar H., Stork N. E., Rijal S., Rahman S. A., Baral H. Effects of land use and land cover change on ecosystem services in the Koshi River basin, Eastern Nepal. *Ecosystem Services*. 2019. Vol. 38. Article 100963.
- 67.Shiyan Du, Jiacheng Li, Masato Noto. Comparison and Analysis of Three MobileNet-Based Models for Wildfire Detection. *Journal of Advances in Information Technology*. 2024. Vol. 15, No. 4. P. 511.
- 68.Shiying Yu, Minerva Singh. Deep Learning-Based Remote Sensing Image Analysis for Wildfire Risk Evaluation and Monitoring. *Fire*. 2025. Vol. 8, No. 1. P. 19.
- 69.Shvaiko, V., Bandurka, O., Shpuryk, V., & Havrylko, Y. V. Methods for detecting fires in ecosystems using low-resolution space images. *Informatyka*,

- Automatyka, Pomiarzy W Gospodarce I Ochronie Środowiska*. 2021. Vol. 1, No. 11. P. 15-19.
70. Smith, R., et al. Satellite-based early warning systems for forest fires. *International Conference on Remote Sensing Applications*. 2024. P. 234-241.
71. Simões-Marques, M.; Figueira, J.R. How Can AI Help Reduce the Burden of Disaster Management Decision-Making? In *International Conference on Applied Human Factors and Ergonomics*. Springer: Berlin/Heidelberg, Germany, 2018.
72. Sobrino, J.A.; Llorens, R.; Fernández, C.; Fernández-Alonso, J.M.; Vega, J.A. Relationship between Soil Burn Severity in Forest Fires Measured In Situ and through Spectral Indices of Remote Detection. *Forests*. 2019. Vol. 10. P. 457.
73. Srinivas K, Dua M. Fog computing and deep CNN based efficient approach to early forest fire detection with unmanned aerial vehicles. *Inventive Computation Technologies*. 2020. Springer International Publishing. P. 646–652.
74. Sun, X., Shao, Z., Zhang, J., & Li, Z. Active Fire Detection from Sentinel-2 Imagery Using Deep Learning. *Remote Sensing*. 2019. Vol. 11, No. 15. P. 1803.
75. Szeliski, R. *Computer Vision: Algorithms and Applications* (2nd ed.). Springer, 2023.
76. Tien Bui D., Bui Q.-T., Nguyen Q.-P., Pradhan B., Nampak H., Trinh P.T. A hybrid artificial intelligence approach using GIS-based neural-fuzzy inference system and particle swarm optimization for forest fire susceptibility modeling at a tropical area. *Agricultural and Forest Meteorology*. 2017. Vol. 233. P. 32–44.
77. Torres F. T. P., Romeiro J. M. N., Santos A. C. d. A., de Oliveira Neto R. R., Lima G. S., Zanuncio J. C. Fire danger index efficiency as a function of fuel moisture and fire behavior. *Science of The Total Environment*. 2018. Vol. 631–632. P. 1304–1310.

78. Vallejo-Villalta I., Rodríguez-Navas E., Márquez-Pérez J. Mapping Forest Fire Risk at a Local Scale – A Case Study in Andalusia (Spain). *Environments*. 2019. Vol. 6. P. 30.
79. Verma S., Singh D., Singh A.K., Jayakumar S. Post-fire soil nutrient dynamics in a tropical dry deciduous forest of Western Ghats, India. *Forest Ecosystems*. 2019. Vol. 6. P. 6.
80. Waldner F., Chen Y., Lawes R., Hochman Z. Needle in a haystack: Mapping rare and infrequent crops using satellite imagery and data balancing methods. *Remote Sensing of Environment*. 2019. Vol. 233. Article 111375.
81. Wang S, Chen T, Lv X, et al. Forest fire detection based on lightweight Yolo. *33rd Chinese Control and Decision Conference (CCDC)*. IEEE, 2021. P. 1560–1565.
82. Ya'acob N., Najib M. S., Tajudin N., Yusof A. L., Kassim M. Image processing based forest fire detection using infrared camera. *Journal of Physics: Conference Series*. 2021. Vol. 1768. IOP Publishing. DOI: 10.1088/1742-6596/1768/1/012014.
83. Yang C., Morton D. C., Randerson J. T. Remote sensing for wildfire monitoring: Insights into burned area, emissions, and fire dynamics. *One Earth*. 2024. Vol. 7. P. 1022–1028. DOI: 10.1016/j.oneear.2024.05.014.
84. YCbCr color spaces. URL: <https://www.hisour.com/ru/ycbcr-color-spaces-26075/> (date of access: 25.11.2024).
85. Zahedi, M., Karim, R., & De, D. MKFF: Mid-point K-means Based Clustering in Wireless Sensor Network for Forest Fire Prediction. *Microsystem Technologies*. 2024.
86. Zheng X, Chen F, Lou L, et al. Real-Time Detection of Full-Scale Forest Fire Smoke Based on Deep Convolution Neural Network. *Remote Sensing*. 2022. Vol. 14, No. 3. P. 536.
87. Ziccardi L. G., Thiersch C. R., Yanai A. M., Fearnside P. M., Ferreira-Filho P. J. Forest fire risk indices and zoning of hazardous areas in Sorocaba, São Paulo state, Brazil. *Journal of Forest Research*. 2020. Vol. 31. P. 581–590.

88. А.-М. Барт-Деланос та ін. Керована подіями гнучкість взаємодії під час виконання процесів спільної роботи. *Системи підтримки прийняття рішень*. 2014.
89. Ардіто Л., Скуотто В., Дель Г. М. Бібліометричний аналіз досліджень аналітики великих даних для бізнесу та управління. *Децис*. 2019.
90. Барабаш О., Бандурка О., Шпурик В., Свинчук О. Інформаційна система аналізу геоданих для відслідковування змін рослинності. *Сучасні інформаційні системи*. 2021. Т. 5, № 4. С. 17–25. DOI: 10.20998/2522-9052.2021.4.03.
91. Бінсон Зе, Сюепін Чжао, Чжіо Чжо і Жей Фан. Реалізація алгоритму виявлення пожежі на DSP TMS320DM642 за допомогою MATLAB/Simulink. URL: https://www.researchgate.net/publication/266646537_Implementation_of_a_Fire_Detection_Algorithm_on_TMS320DM642_DSP_using_MATLABSimulink (дата звернення: 10.04.2024).
92. Борючись із лісовою пожежею та пандемією, Україна стикається з новим ворогом: наземними мінами. URL: <https://www.nytimes.com/2020/10/03/world/europe/ukraine-wildfires-landmines.html> (дата звернення: 15.04.2023).
93. Джебл С., Патіл Ю. Роль великих даних у прийнятті рішень. *Опер. Управління ланцюгом поставок*. 2018.
94. Елія Г., Полімено Г., Солаццо Г. Багатовимірна основа для створення цінності за допомогою великих даних. *Знак пром. кер.* 2019.
95. Журавль І.М. Короткий курс теорії обробки зображень. URL: <https://hub.exponenta.ru/post/kratkiy-kurs-teorii-obrabotki-izobrazheniy734> (дата звернення: 10.04.2024).
96. Зацерковний В., Савков П., Пампуха І., Васецька К. Застосування технологій ГІС та ДЗЗ в задачах моніторингу лісових пожеж. *Вісник Київського національного університету імені Тараса Шевченка. Військово-спеціальні науки*. № 2(44). 2020. С. 54–58.

- 97.Кветний Р.Н., Богач І.В., Бойко О.Р., Софіна О.Ю., Шушура О.М.
Комп'ютерне моделювання систем і процесів. Методи обробки. Частина 2. URL:
https://web.posibnyky.vntu.edu.ua/fksa/2kvetnyj_komp%27yuterne_modelyuvannya_system_procesiv/t2/zm2..htm (дата звернення: 10.04.2024).
- 98.Комп'ютерне моделювання систем процесів. URL:
https://web.posibnyky.vntu.edu.ua/fksa/2kvetnyj_komp%27yuterne_modelyuvannya_system_procesiv/t2/zm2..htm (дата звернення: 25.11.2024).
- 99.Пітак І.В., Негадайлов А.А., Масікевич Ю.Г., Пляцук Л.Д., Шапорєв В.П., Моїсєєв В.Ф. Геоінформаційні технології в екології: Навч.посіб. – Чернівці, 2012. – 273 с. URL: https://geology.lnu.edu.ua/wp-content/uploads/2020/04/Ekolohichna-heoinformatyka_literatura-dlia-lektsiy.pdf (дата звернення: 01.04.2023).
100. Р. Огер та ін. До автоматизації підтримки прийняття рішень для управління ризиками ланцюга постачання серед зацікавлених сторін логістичної мережі. *IFAC-PapersOnLine*. 2018 рік.
101. Сімойнс-Маркес М., Фігейра Дж. Р. Як штучний інтелект може допомогти зменшити тягар прийняття рішень щодо боротьби зі стихійними лихами? *У Міжнародній конференції з прикладних людських факторів та ергономіки; Springer: Берлін/Гайдельберг*, Німеччина, 2018.
102. Тургай Челик. Швидкий і ефективний метод виявлення пожежі за допомогою обробки зображень. URL:
<https://onlinelibrary.wiley.com/doi/pdf/10.4218/etrij.10.0109.0695> (дата звернення: 10.04.2024).
103. Хейс Д. Р., Каппа Ф. Розвідка з відкритим джерелом для оцінки ризику. *Автобус. Гориз.* 2018.

Додаток А

Список опублікованих праць за темою дисертації

1. Головіна Н.В., Ляшенко О.М. Дослідження методів спектрального аналізу зображень для детектування лісових пожеж з використанням технологій обчислювального інтелекту. *Вісник ХНТУ*. 2021. Том 79 (4). С. 74-84.
URL: <https://ojs.kntu.net.ua/index.php/visnyk/article/view/203/249>
DOI: <https://doi.org/10.35546/kntu2078-4481.2021.4.8>
(особистий внесок – досліджено та запропоновано використання сукупності методів спектрального аналізу зображень для моніторингу та попередження лісових пожеж).
2. Головіна Н.В., Ляшенко О.М. Дослідження методів відновлення та реконструкції зображень. *Вчені записки національного університету імені В.І. Вернадського*. Серія: Технічні науки. 2022. Том 33 (72) № 2. С. 80-85.
URL: https://tech.vernadskyjournals.in.ua/journals/2022/2_2022/13.pdf
DOI: <https://doi.org/10.32838/2663-5941/2022.2/13>
(особистий внесок – досліджено та запропоновано використання методів відновлення та реконструкції зображень для моніторингу та попередження лісових пожеж).
3. Головіна Н.В. Розроблення моделі сегментації лісових пожеж на супутникових знімках із застосування нейромережевих технологій. *Збірник наукових праць Національного університету кораблебудування імені адмірала Макарова*. 2024. № 2 (495). С. 95-102.
URL: <https://znp.nuos.mk.ua/archives/2024/2/14.pdf>
DOI: [https://doi.org/10.15589/znp2024.2\(495\).12](https://doi.org/10.15589/znp2024.2(495).12)
(особистий внесок – розроблено модель сегментації лісових пожеж на супутникових знімках із застосуванням нейромережевих технологій).
4. Головіна Н.В. Розроблення системи підтримки прийняття рішень для моніторингу та попередження лісових пожеж в Україні. *Вісник*

Херсонського національного технічного університету. Інформаційні технології. 2024. №2 (89). С. 150-156.

URL:

https://journals.kntu.kherson.ua/index.php/visnyk_kntu/article/view/639/613

DOI: <https://doi.org/10.35546/kntu2078-4481.2024.2.21>

(особистий внесок – розроблено систему підтримки прийняття рішень для моніторингу та попередження лісових пожеж в Україні).

5. Головіна Н.В. Розроблення нейромережевої моделі моніторингу надзвичайних ситуацій природного характеру. *Системи та технології. Комп'ютерні науки.* 2024. №67 (1). С. 36-42.

URL: <https://st.umsf.in.ua/index.php/journal/article/view/143/150>

DOI: <https://doi.org/10.32782/2521-6643-2024-1-67.6>

(особистий внесок – розроблено нейромережеву модель моніторингу надзвичайних ситуацій природного характеру).

6. Головіна Н.В. Розробка нейромережевої моделі для моніторингу лісових пожеж природного характеру. *Вчені Записки Таврійського національного університету імені В.І. Вернадського.* Серія: Технічні науки. 2024. №3 (74). Т.35. С. 53-57.

URL:

https://www.tech.vernadskyjournals.in.ua/journals/2024/3_2024/part_1/1_2.pdf

DOI: <https://doi.org/10.32782/2663-5941/2024.3.1/10>

(особистий внесок – розроблено та програмно реалізовано нейромережеву модель для моніторингу лісових пожеж природного характеру).

Додаток Б

Лістинг коду

```

app.py:
import streamlit as st
import matplotlib.pyplot as plt
from PIL import Image
from source import preprocess_input_image, batch_predict, conv_float_int,
combine_image, load_trained_model, burn_area
import numpy as np
from keras import backend as K
from tensorflow.python.lib.io import file_io
import boto3
from keras.models import load_model

with open("style.css") as f:
    st.markdown('<style>{ }</style>'.format(f.read()), unsafe_allow_html=True)

st.title("Wild Fire Detection App")

st.sidebar.markdown("*** App Status ***")

model, session = load_trained_model("temp_model.h5")
K.set_session(session)

st.sidebar.markdown('Please upload a raw satellite image')
uploaded_file = st.sidebar.file_uploader("Upload png file", type=["png"])

```

```

if uploaded_file is not None:
    uploaded_image = Image.open(uploaded_file)
    st.markdown("** Original Raw Image: **")
    st.image(uploaded_image, width = 500)

    ### Preprocess the raw image
    #st.sidebar.text("Pre-processing the image...")
    with st.spinner("Pre-processing the image..."):
        input_image_array = np.array(uploaded_image)
        original_width, original_height, pix_num = input_image_array.shape
        new_image_array, row_num, col_num =
preprocess_input_image(input_image_array)
        st.sidebar.success("Pre-processing has been done.")

    with st.spinner("Making the prediction..."):
        ##### Make Prediction
        preds = batch_predict(new_image_array, model)
        # combine the images, and converted to 0-255 for display
        output_pred = conv_float_int(combine_image(preds, row_num, col_num,
original_width, original_height, remove_ghost=True)[:,:,:0])
        st.sidebar.success("Prediction has been done.")
        # add image mask to the probability array

    ##### Show the picture
    st.markdown("** The Predicted Probability is **: ")
    plt.imshow(output_pred)

```

```

st.pyplot()

#threshold = st.sidebar.slider("Threshold", 0, 1, 0.25)
preds_t = (preds > 0.25).astype(np.uint8)
output_mask = conv_float_int(combine_image(preds_t, row_num, col_num,
original_width, original_height, remove_ghost=False)[:,:,:0])
st.markdown("** The Predicted Mask is **: ")
plt.imshow(output_mask)
st.pyplot()
#plt.imshow(output_mask)

st.sidebar.markdown("*** CO2 Emission Calculator ***")
forest_type = st.sidebar.selectbox("Please select the type of forest: ",
('Tropical Forest', 'Temperate Forest', 'Boreal Forest', 'Shrublands', 'Grasslands'))
resolution = st.sidebar.text_input("Please enter the image resolution value: ",
'10')

area, biomass_burnt, equal_days = burn_area(output_mask = output_mask,
resolution = float(resolution), forest_type = forest_type)
st.sidebar.markdown('The total burnt area is:')
st.sidebar.text("{0:.2f}".format(area/1e6) + ' km^2')
st.sidebar.markdown('The total CO2 emitted is:')
st.sidebar.text("{0:.2f}".format(biomass_burnt/1e6) + ' tons')
st.sidebar.markdown("This is equivalent to:")
st.sidebar.text("{0:.2f}".format(equal_days) + " days of power emission")

source.py:
import numpy as np
from tensorflow.keras.models import load_model

```

```

from keras import backend as K
from PIL import Image

def add_image_magin(input_image, im_width=720, im_height=480,
color=0):
    # padd
    height, width, vec = input_image.shape
    new_image = np.pad(input_image, ((0, im_height-height), (0, im_width-
width), (0, 0)), 'constant', constant_values=color)
    return new_image

# segment the image
def preprocess_input_image(input_image, im_height=480, im_width=720):
    # normalize the pixel value
    input_image = input_image / np.max(input_image.astype('float'))
    height, width, vec = input_image.shape
    row_num = height//im_height if height%im_height==0 else
height//im_height+1
    col_num = width//im_width if width%im_width==0 else width//im_width+1

    new_image_array = np.zeros((row_num*col_num, im_height, im_width,
vec))

    for i in range(row_num):
        for j in range(col_num):
            temp_image = input_image[im_height*i:im_height*(i+1),
im_width*j:im_width*(j+1), :]
            new_image_array[i*col_num+j] = add_image_magin(temp_image,
im_width=im_width, im_height=im_height, color=0)
    return new_image_array, row_num, col_num

```

```

# combine all images back to original size
def combine_image(input_image, row_num, col_num, original_height,
original_width, im_height=480, im_width=720, remove_ghost=True):
    num, height, width, vec = input_image.shape
    new_image = np.zeros((height*row_num, width*col_num, vec))
    for i in range(row_num):
        for j in range(col_num):
            # Remove the ghost caused by CNN model prediction boundary artifacts.
            # by padding same 4 pixels on the boundary
            if remove_ghost:
                # padd all four edges
                input_image[i*col_num+j, :, :, :] = np.pad(input_image[i*col_num+j,
4:height-4, 4:width-4, :], ((4, 4), (4, 4), (0, 0)), 'edge')

                new_image[im_height*i:im_height*(i+1), im_width*j:im_width*(j+1), :] =
input_image[i*col_num+j, :, :, :]
            return new_image[:original_height, :original_width, :]

# predict batches of images, and return the probability array
def batch_predict(input_image_array, model):
    num, height, width, vec = input_image_array.shape
    preds_array = np.zeros((num, height, width, 1))
    # to avoid OOM
    for ii in range(input_image_array.shape[0]):
        preds_array[ii] = model.predict(np.expand_dims(input_image_array[ii, :, :,
:], axis=0), verbose=1)
    return preds_array

```

```

def conv_float_int(image):
    # convert array back to [0, 255] for display
    if not np.any(image): # check if the array is all zero
    return image.astype(int)

    return ((image-np.min(image))/(np.max(image)-
np.min(image))*255).astype(int)

```

```

def load_trained_model(model_location):
    loaded_model = load_model(model_location)
    session = K.get_session()
    return loaded_model, session

```

```

def burn_area(output_mask, resolution, forest_type):
    # reference: https://www.geosci-model-dev.net/4/625/2011/gmd-4-625-2011.pdf
    # unit in g/m^2
    biomass_type = {'Tropical Forest': 28076,
'Temperate Forest':10492,
'Boreal Forest': 25000,
'Shrublands': 5705,
'Grasslands': 976
}
    area = np.count_nonzero(output_mask) * resolution**2
    #  $E_i = A(x,t) \times B(x) \times FB \times efi$ ,
    # A: area,
    # B(biomass_type),
    # FB: Burning fraction, assume 1 here
    # efi: mass of biomass burnt, for CO2, it averages 1624 g/kg.
    biomass_burnt = area * biomass_type[forest_type]/1e3 * 1624 #unit in g

```

annual CO2 emission from power generating, 424 million metric tons of CO2 per year

#

Reference:

https://ww3.arb.ca.gov/cc/inventory/pubs/reports/2000_2016/ghg_inventory_trends_00-16.pdf

ca_co2_daily = 4.24e8 / 365.

equal_days = biomass_burnt / 1e6 / ca_co2_daily

print('The total burnt area is:', "{:.4e}".format(area/1e6), 'km^2 \n')

print('The total CO2 emitted is:', "{:.4e}".format(biomass_burnt/1e6), 'tons \n')

print("Which equivalent to:", "{:.4e}".format(equal_days), " days of daily power emission \n")

return area, biomass_burnt, equal_days

Додаток В

Скріншоти застосунку

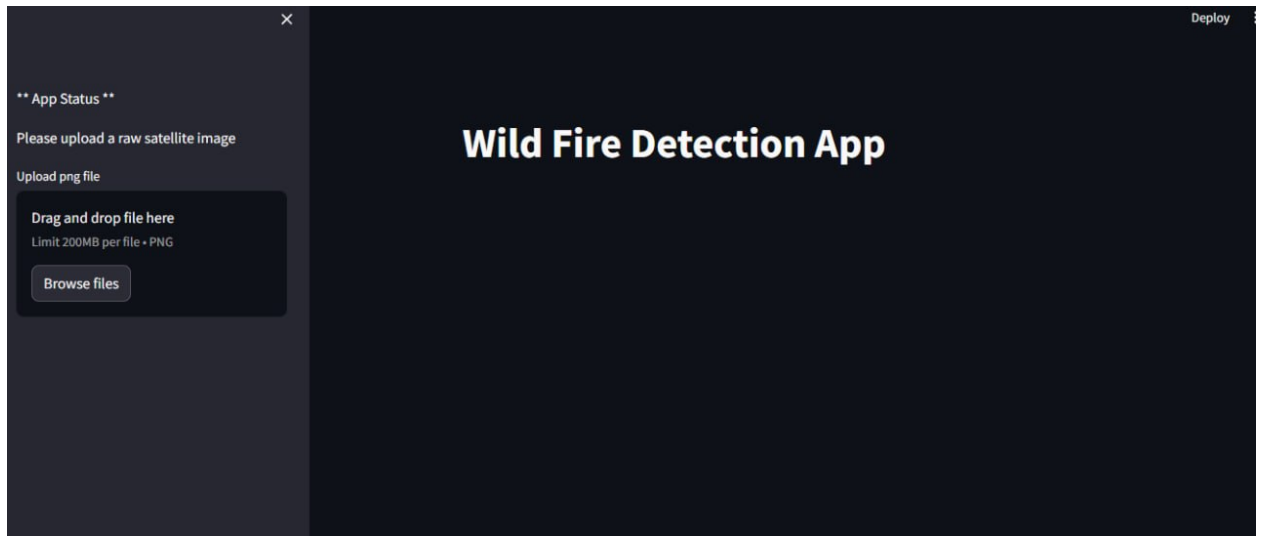


Рис. В.1. Інтерфейс розробленої комп'ютерної системи



Рис. В.2. Завантаження зображення для аналізу

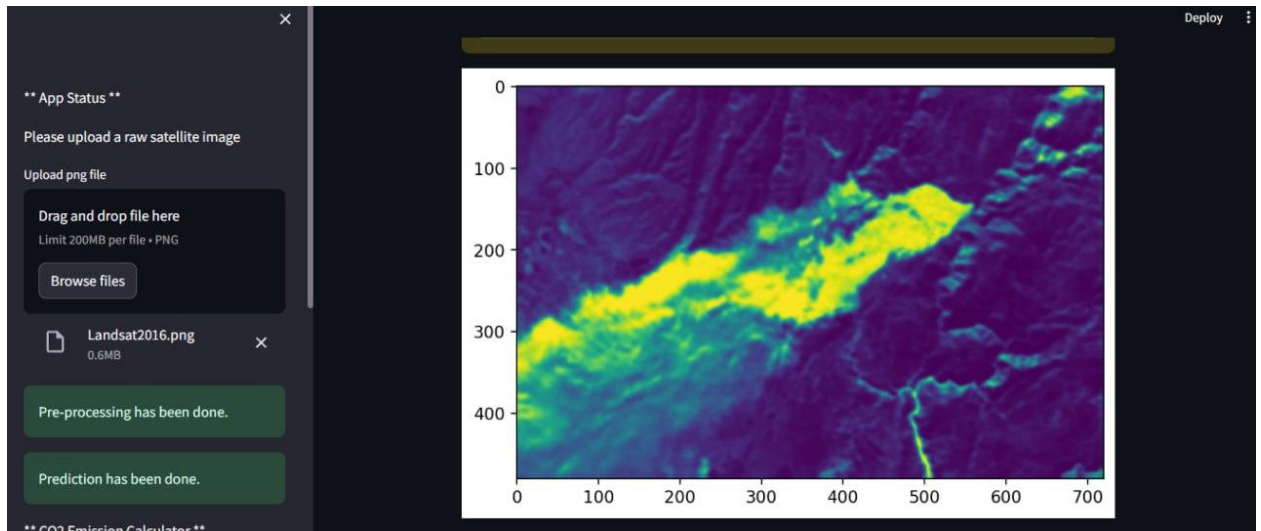


Рис. В.3. Виявлення осередків вогню

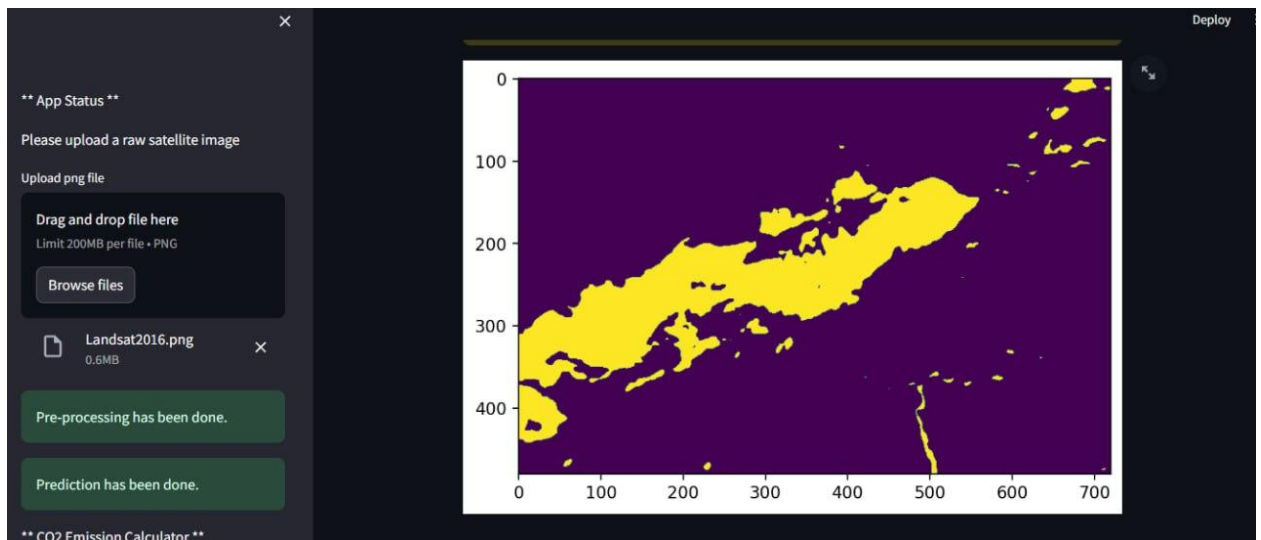


Рис. В.4. Виділення маски осередків вогню

Додаток Г

Акт про впровадження

«ЗАТВЕРДЖУЮ»



Перший проректор
Херсонського національного технічного
університету

Марія Артеменко
«18» квітня 2025 р.

АКТ

про впровадження результатів дисертаційної роботи

Головіної Надії Владиславівни

«Моделі та методи обробки потоків даних зображень в системі підтримки прийняття рішень»,
що подається на здобуття наукового ступеня доктора філософії
зі спеціальності 122 – Комп'ютерні науки

Головіна Надія Владиславівна в 2021-2025 рр. здобувала вищу освіту на третьому освітньо-науковому рівні на кафедрі програмних засобів і технологій та проводила науково-дослідну роботу в Херсонському національному технічному університеті у межах виконання дисертаційного дослідження та у межах плану проведення науково-дослідної роботи кафедрою Програмних засобів і технологій Херсонського національного технічного університету за темою «Моделі та методи територіальної інформаційної системи підтримки управління евакуацією населення при надзвичайних ситуаціях».

Комісія у складі:

голови комісії – декана факультету інформаційних технологій і дизайну доцента Віктора КОЗЕЛЯ, та

членів комісії – завідувача кафедри програмних засобів і технологій, к.т.н. доцента Оксани ОГНЕСВОЇ та асистента кафедри програмних засобів і технологій Дмитра ЧОРНОГО

цим Актом засвідчує, що результати проведених Головіною Надією Владиславівною досліджень впроваджено кафедрою Програмних засобів і технологій Факультету інформаційних технологій та дизайну Херсонського національного технічного університету при підготовці і викладанні з використанням електронних навчально-методичних комплексів навчальних курсів дисциплін «Технології обчислювального інтелекту», «Основи машинного навчання», на другому (магістерському) рівні вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення», а також «Інтелектуальні системи і технології» на третьому освітньо-науковому рівні вищої освіти за спеціальністю 122 «Комп'ютерні науки», та при виконанні здобувачами вищої освіти курсових і кваліфікаційних робіт.

Результати впровадження результатів науково-дослідної та експериментальної роботи дають підстави вважати дисертаційне дослідження Головіної Н.В. як таке, що має теоретичне і практичне значення для закладів вищої освіти у рамках підготовки ІТ-фахівців.

«18» квітня 2025 р.

Голова комісії, декан факультету
інформаційних технологій та дизайну,
доцент

Віктор КОЗЕЛ

член комісії, завідувач кафедри
програмних засобів і технологій, к.т.н.
доцент

Оксана ОГНЕСВА

член комісії, асистент кафедри
програмних засобів і технологій

Дмитро ЧОРНИЙ