Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему *Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання*

Галузь знань *12 – Інформаційні технології*
Шифр і назва галузі знань

Спеціальність *122 – Комп'ютерні науки*
Шифр і назва спеціальності

Освітня програма *Комп'ютерні науки*
Назва освітньої програми

Виконала: *студент групи КН-21-1* — Група виконавця | Підпис | *Ігор ШВАЧКА* — Ім'я, ПРІЗВИЩЕ

Керівник: *д.т.н., проф. каф. КН* — Науковий ступінь, посада | Підпис | *Едуард МАНЗЮК* — Ім'я, ПРІЗВИЩЕ

Нормоконтроль: *к.т.н., доц. каф. КН* — Науковий ступінь, посада | Підпис | *Руслан БАГРІЙ* — Ім'я, ПРІЗВИЩЕ

До захисту допускаю:
зав. кафедри КН, д.т.н., професор — Підпис | *Олександр БАРМАК* — Ім'я, ПРІЗВИЩЕ

18 06 2025 р.

Хмельницький 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет *інформаційних технологій*

Кафедра *комп'ютерних наук*

Освітній ступінь *бакалавр*

Галузь знань *12 – Інформаційні технології*

Спеціальність *122 – Комп'ютерні науки*

ЗАТВЕРДЖУЮ

Завідувач кафедри *комп'ютерних наук*

_____

(підпис)

д.т.н., професор Олександр БАРМАК

«*10*» *02* *2025* року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

1. Тема кваліфікаційної роботи бакалавра: *«Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання»*

2. Завдання видано студенту *Ігорю Швачці*
(Ім'я, прізвище)

3. Керівник роботи *д.т.н., проф. каф. КН Едуард МАНЗЮК*
(посада, ім'я, прізвище)

4. Затверджено наказом університету від «*07*» *02* *2025* р. №*23*

5. Дата видачі завдання студенту: «*10*» *02* *2025* р.

6. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета кваліфікаційної роботи бакалавра – підвищення точності виявлення фейкових зображень на основі аналізу зображень засобами штучного інтелекту. Задачі: провести аналіз актуальної розробки методу класифікації зображень; провести аналіз теоретичних та практичних підходів які дозволяють визначити зображення; розробити метод для визначення фейкових зображень; розробити інформаційну систему методу визначення згенеровоних зображень; провести тестування та оцінення точності системи класифікації на реальних і згенерованих зображеннях.

7. Календарний план виконання кваліфікаційної роботи бакалавра:

№	Назва етапів (розділів) кваліфікаційної роботи бакалавра	Термін виконання	Примітка
1	*Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи бакалавра з керівником, складання календарного графіка виконання роботи*	*січень 2025*	Виконано
2	*Ознайомлення з предметною областю, формулювання мети та задач дослідження, визначення об'єкта та предмета дослідження*	*лютий 2025*	Виконано
3	*Проектування та розробка загальної архітектури методу класифікації фейкових зображень, інтерфейсу користувача, вибір засобів реалізації*	*березень 2025*	Виконано
4	*Створення та дослідження точності методу класифікації фейкових зображень*	*квітень 2025*	Виконано
5	*Написання пояснювальної записки, урахування зауважень керівника, оформлення згідно вимог*	*травень 2025*	Виконано
6	*Розробка презентаційних матеріалів та попередній захист кваліфікаційної роботи*	*травень 2025*	Виконано
7	*Отримання відгуку керівника, рецензії, перевірка на плагіат, нормоконтроль*	*червень 2025*	Виконано
8	*Підготовка до захисту та захист кваліфікаційної роботи бакалавра*	*червень 2025*	Виконано

Виконавець: *студент групи КН-21-1* _____ *Ігор Швачка*
Група виконавця — Підпис — Ім'я, ПРІЗВИЩЕ

Керівник: *д.т.н., проф. каф. КН* _____ *Едуард МАНЗЮК*
Науковий ступінь, посада — Підпис — Ім'я, ПРІЗВИЩЕ

Анотація

Тема кваліфікаційної роботи бакалавра: *«Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання»*

Виконавець кваліфікаційної роботи бакалавра: *студент групи КН-21-1 Ігор Швачка*

Керівник кваліфікаційної роботи бакалавра: *д.т.н., проф. каф. КН Едуард МАНЗЮК*

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
46	17	2	43	2

Мета кваліфікаційної роботи бакалавра – підвищення точності виявлення фейкових зображень на основі аналізу зображень засобами штучного інтелекту.

Розроблений метод надає можливість локально використовувати штучний інтелект, який здатний розпізнавати фейкові зображення від реальних, що надає можливість підвищити точність виявлення згенерованих зображень на основі аналізу зображень.

Ключові слова: генерація зображеня, згорткова нейронна мережа, зображення, виявлення, навчальна модель.

Виконавець: *студент групи КН-21-1* _____ *Ігор ШВАЧКА*

Група виконавця | Підпис | Ім'я, ПРІЗВИЩЕ

Зміст

Перелік скорочень

Скорочення, термін, позначення	Пояснення
ШІ	Штучний інтелект
ЗНМ	Згорткова нейронна мережа
GAN	Generative Adversarial Network
MVC	Model-View-Controller
ПП	Програмний продукт
ІС	Інформаційна система

Вступ

Кваліфікаційна робота присвячена розробці методу класифікації фейкових зображень із використанням комп'ютерного зору та глибинного навчання. Через зростання обсягів візуального контенту зростає потреба в автоматичних засобах виявлення підроблених зображень. Використання згорткових нейронних мереж у поєднанні з комп’ютерним зором дає нові можливості для розв’язання цієї задачі.

Актуальність теми зумовлена поширенням фейкових зображень у соціальних мережах, засобах масової інформації та інших цифрових платформах. Такі зображення можуть мати значний вплив на суспільну думку, політичні процеси та безпеку. Тому питання розробки ефективних методів виявлення та класифікації фейкових візуальних даних є надзвичайно важливим у сучасних умовах.

Об’єкт дослідження – процес класифікації фейкових зображень з використанням комп'ютерного зору на основі глибинного навчання.

Предмет дослідження – методи генеративного ШІ, засоби та підходи до виявлення фейкових зображень.

Мета кваліфікаційної роботи бакалавра – підвищення точності виявлення фейкових зображень на основі аналізу зображень засобами штучного інтелекту.

Для досягнення поставленої мети необхідно реалізувати наступні задачі:

– Провести аналіз актуальної розробки методу класифікації зображень;

– провести аналіз теоретичних та практичних підходів які дозволяють визначити зображення;

– розробити метод для визначення фейкових зображень;

– розробити інформаційну систему методу визначення згенеровоних зображень;

– провести тестування та оцінення точності системи класифікації на реальних і згенерованих зображеннях.

Розділ 1 Огляд моделей, методів і підходів до виявлення фейкових зображень на основі комп'ютерного зору та глибинного навчання

1.1 Аналіз інформаційних моделей визначення фейкових зображень

Зображення - важливий засіб комунікації, а людське сприйняття базується на зорових даних, що робить аналіз графіки ключовим у багатьох сферах [1]. Сучасні дослідження підтверджують ефективність механізмів уваги в комп'ютерному зорі для покращення обробки зображень [2]. Цифрове зображення - двовимірна матриця пікселів з інформацією про колір і яскравість, що є основою для аналізу [3].

Цифрові технології дозволяють створювати або змінювати зображення, породжуючи проблему ідентифікації фейків [4]. Цифровий аналіз медіа стає ключовим інструментом у боротьбі з такими підробками, особливо з поширенням DeepFake [5]. Фейкові зображення модифіковані або створені для введення в оману та шахрайства [6].

Традиційні методи виявляють підробки через аналіз піксельних аномалій і метаданих [7], але глибокі нейронні мережі показують кращі результати [8]. З ростом складності маніпуляцій ці методи стають менш ефективними [9].

За останнє десятиліття значно зросла роль комп'ютерного зору та глибинного навчання у виявленні фейків [10]. Нейронні мережі автоматично виявляють тонкі закономірності, що відрізняють оригінали від підробок [11]. GAN служать як для створення, так і для виявлення аномалій [12]. Компактні мережі, як MesoNet, ефективні для виявлення підробок облич у відео [13].

Методи класифікації фейків застосовуються у криміналістиці, рекламі, журналістиці та політичних технологіях для боротьби з маніпуляціями [14]. Основні сутності - цифрове зображення, метадані, ознаки, модель класифікації та користувач [15]. Процеси включають збір, обробку, витяг ознак, навчання і класифікацію [16]. Аналіз предметної області виділяє завдання автоматизації виявлення підробок із застосуванням ШІ для підвищення точності й швидкості рішень [17].

1.2 Огляд теоретичних підходів до розв’язку подібних задач по виявленню зображень створених нейромережами

Для розв’язання задачі виявлення підроблених зображень широко застосовують методи штучного інтелекту, зокрема глибинне навчання [18], [19]. Основним інструментом у цій галузі є згорткові нейронні мережі (CNN), які забезпечують ефективний автоматичний аналіз візуальної інформації, здатні виділяти складні ознаки зображень без необхідності ручного конструювання алгоритмів [20].

CNN складаються зі згорткових шарів, що фільтрують вхідні дані та виявляють локальні патерни, шарів підвибірки (пулингу), які зменшують розмірність даних, а також повнозв’язних шарів для класифікації [21], [22]. Така структура дозволяє досягти високої точності у задачах розпізнавання та класифікації зображень [23], [24].

Для навчання CNN використовують набори маркованих даних, де мережа навчається розрізняти справжні та підроблені зображення [25]. Навчання з учителем передбачає оптимізацію ваг мережі шляхом мінімізації функції втрат, що відображає різницю між передбаченнями моделі та реальними мітками [26].

Одним із викликів у цій сфері є необхідність великої кількості різноманітних даних для навчання, що забезпечує узагальнювальну здатність моделі [27]. Техніки збільшення даних (data augmentation), такі як повороти, масштабування, змінення яскравості, допомагають розширити доступні тренувальні набори, покращуючи стійкість мережі до варіацій зображень [28], [29].

Окрім класичних CNN, у сучасних дослідженнях використовують глибинні ансамблі моделей, а також методи трансферного навчання [30], [31], що дозволяють адаптувати попередньо навчені моделі для нових задач з меншими витратами ресурсів [32], [33].

Бібліотеки TensorFlow та Keras у мові програмування Python надають гнучкі засоби для побудови, навчання та оцінювання нейронних мереж [34]. Вони підтримують різноманітні архітектури CNN, полегшують роботу з великими наборами даних і дозволяють експериментувати з параметрами моделей, що є важливим для підвищення їхньої продуктивності у задачах детекції підробок [35].

В результаті, використання глибинного навчання на базі TensorFlow та Keras дає змогу створити ефективні системи автоматичного виявлення фальшивих зображень, що є актуальним напрямком у боротьбі з дезінформацією та забезпеченні достовірності візуального контенту [36], [37].

1.3 Аналіз існуючих програмних засобів та наукових рішень детекції фейкових зображень

Через стрімкий розвиток цифрових технологій візуальний контент став повсякденним явищем - від особистих зображень до ілюстрацій у новинах. Зі зростанням ролі штучного інтелекту та глибинного навчання постала нова проблема: створення зображень, що майже не відрізняються від реальних. Через що з’явилась необхідність у появі спеціалізованих програм, здатних надійно визначати, чи є зображення оригінальним, зміненим чи повністю згенерованим.

У цьому порівняльному огляді будуть представлені кілька програм, які мають на меті виявлення підроблених зображень або перевірку їх автентичності. Аналіз спрямований на висвітлення функціональних можливостей кожного інструменту, їхніх переваг і недоліків, з метою визначити найбільш ефективне програмне забезпечення для подібних завдань. Для кожного з додатків буде окремо розглянута специфіка роботи та потенціал у практичному використанні.

Слід зауважити, що технології постійно оновлюються, а отже, з часом кожен із розглянутих інструментів може отримати нові функції. Регулярна перевірка оновлень дозволяє підтримувати актуальність використання програмного забезпечення.

Одним із таких інструментів є Perceptual Image Diff - програма, розроблена для глибокого порівняння двох зображень з метою виявлення розбіжностей між ними. Вона аналізує зображення на рівні пікселів, орієнтуючись на візуальні структури, що дозволяє виявляти навіть мінімальні зміни. Завдяки складним методам аналізу, цей інструмент підходить для детальної перевірки автентичності графічного контенту. Perceptual Image Diff використовує моделі людського зору для виявлення відмінностей, які відображаються у вигляді карти змін, що полегшує інтерпретацію результатів при візуальному порівнянні оригіналу та підозрілого зображення.

Серед сильних сторін Perceptual Image Diff:

– Здатність до глибокого аналізу структури зображення, що дозволяє виявляти зміни, не видимі на перший погляд;

– відкритий вихідний код, що забезпечує можливість адаптації під індивідуальні потреби користувача;

– застосування розвинених алгоритмів порівняння, які забезпечують високу точність результатів.

Недоліки Perceptual Image Diff:

– Для ефективного використання потрібні знання у сфері комп’ютерного бачення, що може ускладнити роботу для новачків;

– високі обчислювальні вимоги можуть впливати на швидкодію при аналізі великих або складних зображень.

Отже, Perceptual Image Diff є технічно потужним інструментом для дослідження графічних змін, здатним виявляти навіть незначні відхилення між зображеннями шляхом моделювання особливостей людського зору, але вимагає певного рівня компетентності й ресурсів для повноцінного використання [38].

Рисунок 1.1 – Інтерфейс Perceptual Image Diff [38]

Beyond Compare (див. рисунок 1.2) - це універсальна програма для зіставлення файлів і каталогів, яка також може бути використана для порівняння зображень. Вона дозволяє знаходити відмінності між графічними файлами, а також синхронізувати великі обсяги даних, що робить її корисною не лише для роботи з візуальним контентом, а й для управління файлами загалом.

Переваги Beyond Compare:

– Програма підтримує порівняння на рівні пікселів, що дозволяє точно визначати навіть найдрібніші зміни в зображеннях;

– інтуїтивно зрозумілий інтерфейс дає змогу швидко візуалізувати відмінності без залучення складного аналізу;

– можливість синхронізувати дані між каталогами значно розширює спектр її застосування, особливо в проєктній роботі.

Недоліки Beyond Compare:

– Beyond Compare не призначена для роботи з контентом, створеним нейронними мережами, тому точність при виявленні фальшивих зображень на основі AI обмежена;

– у порівнянні з вузькоспеціалізованими рішеннями, вона не здатна виявляти глибші або приховані аномалії в штучно створених зображеннях.

У підсумку, Beyond Compare є зручним рішенням для базового аналізу і загального порівняння зображень, однак у сфері детектування AI-контенту її можливості залишаються недостатніми [39].

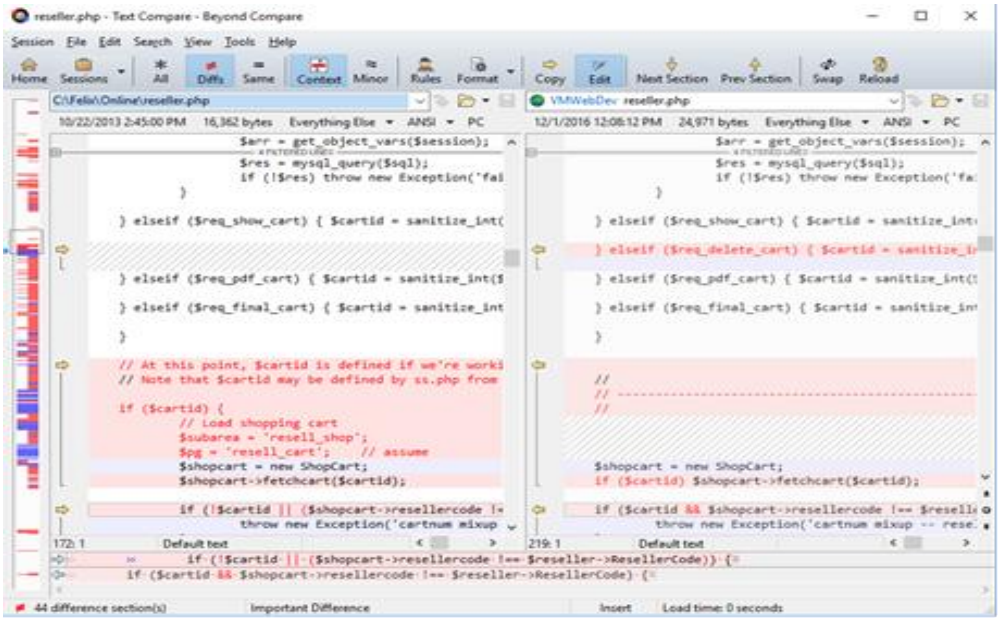


Рисунок 1.2 – Інтерфейс Beyond Compare [39]

DiffImg (див. рисунок 1.3) - це легкий у використанні застосунок для швидкого порівняння зображень. Його головне призначення - виявлення візуальних відмінностей між двома файлами за допомогою простих маркерів, які одразу показують, де були внесені зміни.

Основні переваги DiffImg:

– Простий інтерфейс і відсутність потреби в спеціалізованих знаннях роблять його доступним для широкої аудиторії;

– програма невимоглива до системних ресурсів, що дозволяє запускати її навіть на малопотужних пристроях;

– підходить для швидкої перевірки базових змін, зокрема при порівнянні оригінального зображення і потенційної підробки.

Недоліки DiffImg:

– Точність програми є обмеженою: вона аналізує лише піксельні відмінності, що ускладнює виявлення складніших змін, властивих AI-генерації;

– для задач, що потребують глибокого розпізнавання маніпуляцій, DiffImg не є достатньо потужним інструментом, і його варто доповнювати іншими програмами.

Таким чином, DiffImg - це швидкий і зручний варіант для базового аналізу зображень, але для серйозного розпізнавання фальсифікацій краще використовувати більш розвинене програмне забезпечення [40].

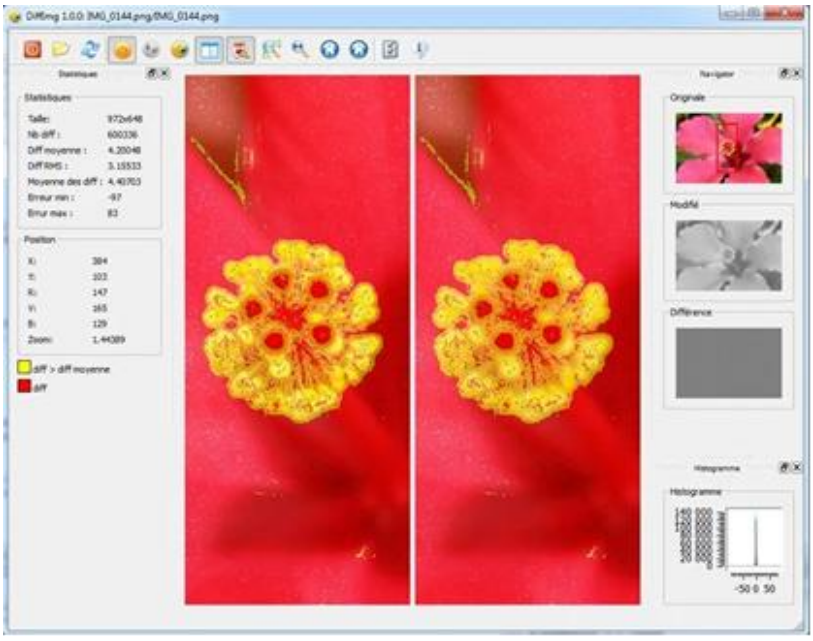


Рисунок 1.3. – Інтерфейс DiffImg [40]

Використання CNN у задачах розпізнавання фальшивих зображень відкриває нові горизонти в автоматизації цього процесу. Завдяки здатності самонавчатися на великих наборах даних, CNN забезпечують високу точність аналізу й здатні виявляти найдрібніші маніпуляції, які залишаються непомітними для класичних алгоритмів. Мережева структура легко адаптується до нових типів фальсифікацій, зберігаючи при цьому ефективність. Окрім цього, CNN підходять для роботи з різноманітними типами зображень та можуть масштабуватися відповідно до складності задачі. Попри те, що інші інструменти можуть бути простішими у використанні або швидшими для базових перевірок, саме нейромережеві методи забезпечують найглибший і найнадійніший аналіз, що

критично важливо в умовах зростаючої кількості фальшивих зображень, згенерованих за допомогою штучного інтелекту.

1.4 Мета, задачі та вимоги до реалізації інформаційної системи класифікації фейкових зображень

Мета кваліфікаційної роботи бакалавра – підвищення точності виявлення фейкових зображень на основі аналізу зображень засобами штучного інтелекту.

Для досягнення поставленої мети необхідно реалізувати наступні задачі:

– Провести аналіз актуальної розробки методу класифікації зображень;

– провести аналіз теоретичних та практичних підходів які дозволяють визначити зображення;

– розробити метод для визначення фейкових зображень;

– розробити інформаційну систему методу визначення згенеровоних зображень;

– провести тестування та оцінення точності системи класифікації на реальних і згенерованих зображеннях.

Розділ 2 Розробка методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання

2.1 Метод класифікації фейкових зображень

Вхідними даними для методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання є масив вхідних даних, які зображені на малюнку нижче (рисунок 2.1).

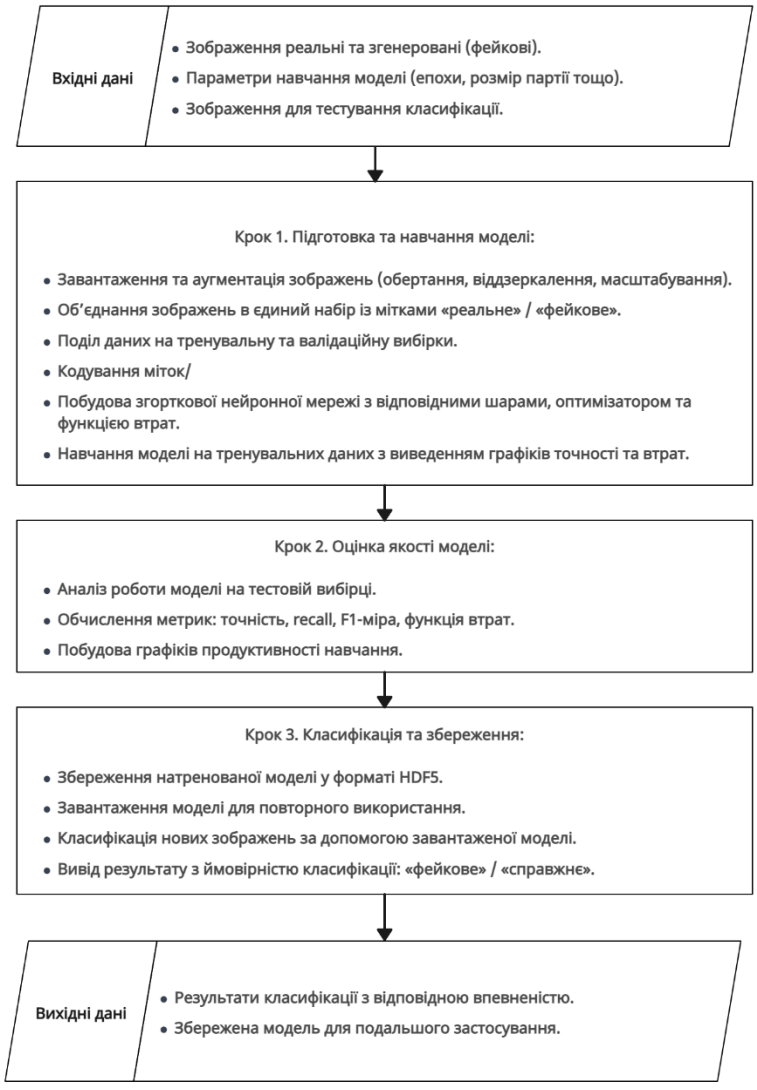


Рисунок 2.1 – Схема методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання

Схема методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання складається з наступних частин:

– Вхідні дані: Використовується набір реальних і фейкових зображень для тренування моделі. Задаються параметри навчання, такі як кількість епох, розмір батчу та максимальна кількість зображень на клас. Окремо готується зображення для перевірки класифікації.

– Крок 1: завантажуються зображення з папок REAL і FAKE. Виконується стандартизація: зміна розміру (задається моделлю) і нормалізація значень пікселів до [0, 1]. Кожному зображенню присвоюється мітка: 0 (реальне) або 1 (фейк). Дані змішуються і розбиваються на тренувальну та валідаційну вибірки. Використовується архітектура CNN із шарами Conv2D, MaxPooling2D, Flatten, Dense та Dropout для уникнення перенавчання. Модель компілюється з оптимізатором Adam, функцією втрат binary_crossentropy та метрикою accuracy. Навчання виконується з використанням ранньої зупинки та збереженням найкращої моделі;

– Крок 2: модель оцінюється на валідаційній вибірці. Використовуються метрики точності, втрат, матриця сплутаності та показники (precision, recall, F1-score), які відображаються у графічних діаграмах;

– Крок 3: після тренування модель зберігається у файл .h5. Для класифікації нових зображень модель завантажується, зображення проходять підготовку (зміна розміру, нормалізація), а метод predict визначає клас (реальне чи фейк) із відображенням результату у графічному інтерфейсі;

– Вихідні дані: результати класифікації показують клас зображення та рівень впевненості. Модель зберігається для повторного використання, економлячи час на подальше тренування.

Створена схема методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання максимально детально показує усі ключові етапи обробки даних - від початкового завантаження і підготовки зображень до навчання, оцінки, збереження моделі та застосування її

для нових передбачень. Така структура дозволяє чітко та наочно відобразити всі кроки побудови та використання моделі, що сприяє кращому розумінню роботи алгоритму та його практичного застосування.

2.2 Функціональна структура інформаційної системи визначення згенерованих зображень

Функціональна структура розробленої інформаційної системи передбачає реалізацію двох основних режимів роботи: навчання моделі класифікації зображень та розпізнавання зображень за допомогою попередньо натренованої моделі. Кожен із режимів реалізується у вигляді окремого функціонального модуля з відповідним графічним інтерфейсом, що забезпечує просту та логічну взаємодію користувача із системою.

Основні функціональні можливості системи:

– Користувач має можливість обирати каталоги із зображеннями двох класів - реальними (REAL) та згенерованими (FAKE). Що дає змогу системі сформувати навчальний і валідаційний набори;

– виконання нормалізації зображень, перевірка обсягу доступної оперативної пам’яті, а також автоматичний поділ даних на тренувальну (80%) та валідаційну (20%) вибірки;

– налаштування основних параметрів навчання моделі, зокрема кількість епох, розмір партії (batch size) та обмеження кількості зображень у кожному класі;

– тренування згорткової нейронної мережі з аугментацією даних. У процесі відображаються графіки точності та функції втрат по епохах, що дозволяє оцінити результат навчання;

– після навчання модель зберігається у форматі HDF5 для подальшого використання;

– підтримується імпорту існуючої моделі без повторного навчання;

– можливість обрати окреме зображення у форматі PNG, JPG, JPEG або BMP. Система масштабує зображення для перегляду, обробляє його та видає

результат класифікації - «Справжнє» або «Фейкове» з відповідним відсотком упевненості.

Графічний інтерфейс реалізує зрозумілу навігаційну структуру (рисунок 2.3), яка дозволяє користувачу переходити між основними режимами:

1. Головне меню:
– Перехід до модуля навчання моделі;
– перехід до модуля класифікації зображень;
– завершення роботи програми.
2. Форма навчання:
– Введення параметрів;
– запуск процесу навчання;
– повернення до головного меню.
3. Форма розпізнавання:
– Вибір зображення;
– класифікація зображення;
– повернення до головного меню.

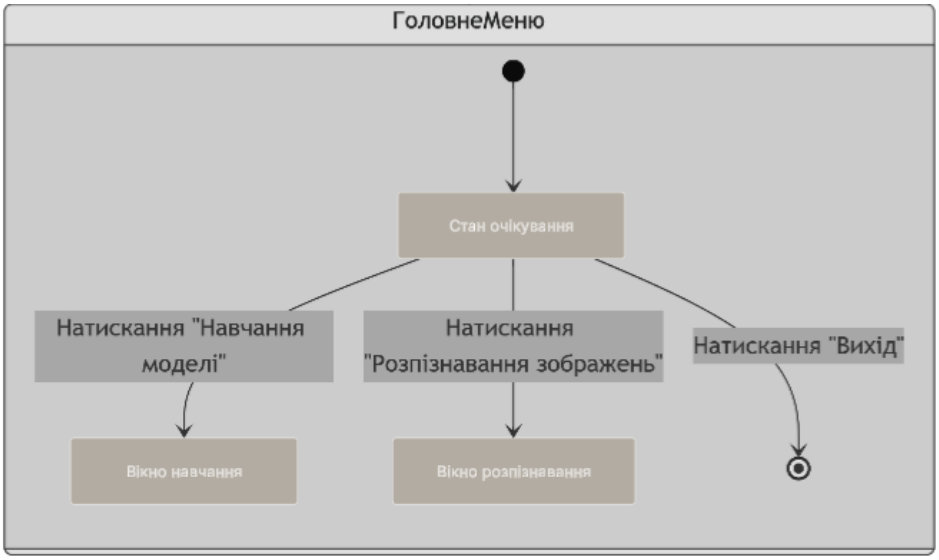


Рисунок 2.3 – Схема навігації між формами системи

На рисунку 2.4 представлено DFD-діаграму першого рівня, що ілюструє обмін даними між основними функціональними компонентами, зовнішніми сутностями та сховищами.

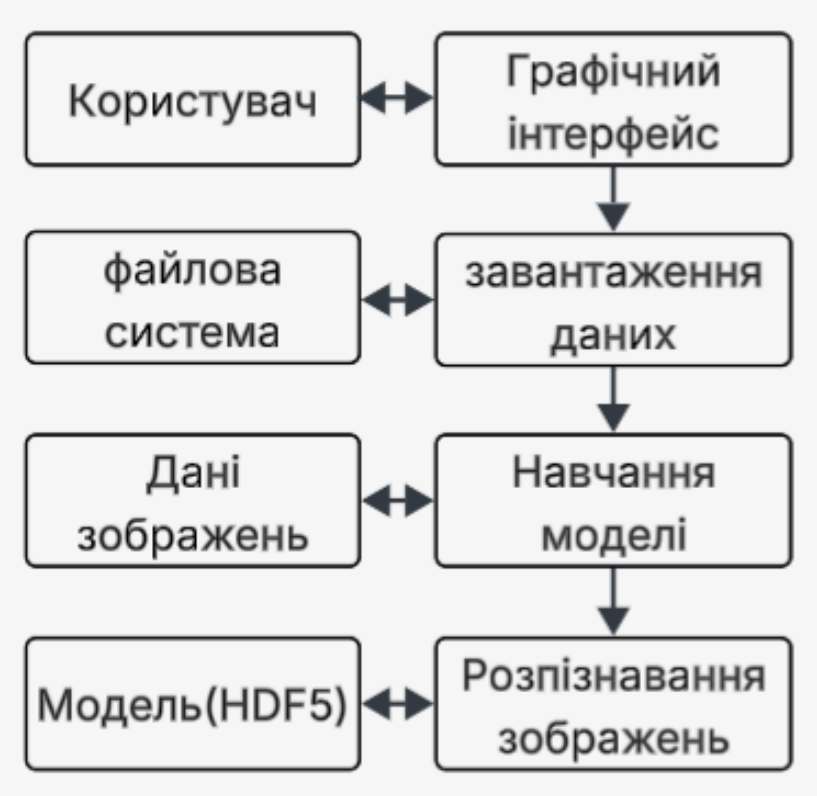


Рисунок 2.4 – DFD-діаграма системи "Детектор фейкових зображень"

Зовнішні сутності:

– Користувач - джерело вхідних даних (зображення, параметри), одержувач результатів;

– файлова система -джерело даних і місце збереження моделей.

Основні процеси:

1.Завантаження даних:

– Вхід: шляхи до папок REAL/FAKE;

– процес: перевірка пам’яті, нормалізація, поділ на вибірки;

– вихід: X_train, X_val, y_train, y_val.

2. Навчання моделі:

– Вхід: тренувальні дані, параметри навчання;

– процес: аугментація, тренування, збереження;

– вихід: модель у HDF5, графіки.

3. Розпізнавання зображень:

– Вхід: окреме зображення;

– процес: обробка, класифікація;

– вихід: ймовірність фейковості, упевненість.

Сховища:

– Набори даних - тренувальні та валідаційні;

– модель - зберігається у форматі .h5.

Інформаційна система складається з наступних логічних модулів:

1. Модуль завантаження даних – реалізує вибір, обробку та розподіл зображень.

2. Модуль навчання моделі – здійснює конфігурацію, запуск та збереження навчання.

3. Модуль розпізнавання – забезпечує класифікацію окремих зображень.

4. Модуль користувацького інтерфейсу – керує вікнами, подіями та переходами.

Модульна архітектура сприяє підтримці, повторному використанню коду та масштабуванню системи.

2.3 Розробка архітектури нейронної мережі виявлення згенерованих зображень

Незважаючи на використання високорівневих бібліотек, таких як TensorFlow та Keras, структура моделі створюється вручну відповідно до специфіки задачі бінарної класифікації зображень. Модель призначена для обробки зображень, зменшених до розміру 64×64 пікселів із трьома кольоровими каналами (RGB), що дає змогу ефективно виявляти характерні артефакти, притаманні синтетичним або фейковим зображенням, створеним за допомогою генеративних моделей.

Архітектура моделі реалізована як послідовна (Sequential) структура шарів, де кожен компонент виконує специфічну функцію в процесі вилучення ознак та класифікації. Основу моделі становлять згорткові шари (Conv2D), кожен із яких вилучає дедалі складніші ознаки з вхідного зображення. Перший згортковий шар містить 32 фільтри розміром 3×3 з активацією ReLU. Далі йдуть шари з 64, 128 і 256 фільтрами відповідно, усі також із ядром 3×3 та функцією активації ReLU. Після кожного згорткового шару використовується шар нормалізації BatchNormalization для стабілізації розподілу активацій, що покращує збіжність та пришвидшує навчання.

Зменшення просторових розмірів здійснюється за допомогою шарів MaxPooling2D із розміром вікна 2×2, що дозволяє зберігати суттєві ознаки при зменшенні розміру вхідних даних та зниженні обчислювального навантаження. Вихід останнього згорткового шару сплощується через Flatten для переходу до щільного (Dense) шару. Перший повнозв’язний шар містить 256 нейронів із активацією ReLU, після чого використовується шар Dropout із імовірністю 0.5 для зменшення ризику перенавчання. Завершує модель вихідний шар із одним нейроном та активацією sigmoid, який генерує ймовірність належності зображення до класу "фейкове".

Розмірність вхідних даних становить (batch_size, 64, 64, 3), де batch_size - кількість зображень, оброблюваних за раз (наприклад, 32). Дані мають тип float32 та нормалізуються до діапазону [0, 1]. Вихід моделі має форму (batch_size, 1) і містить ймовірність для кожного зображення щодо його класифікації як фейкове (1) або справжнє (0), також у форматі float32.

Модель компілюється з використанням оптимізатора Adam зі швидкістю навчання 0.0001. Як функція втрат використовується бінарна крос-ентропія (binary_crossentropy), що дозволяє ефективно навчатись на задачах двокласової класифікації. Основна метрика оцінки - точність (accuracy), яка вказує на частку правильних передбачень.

Розроблена архітектура забезпечує баланс між обчислювальною ефективністю та точністю, що робить її придатною для інтеграції в програмну

систему класифікації фейкових зображень. За результатами експериментального налаштування, така структура дозволяє досягти високих показників точності при помірних ресурсних вимогах, з можливістю масштабування на складніші набори даних.

2.4 Проектна архітектура системи та взаємозв'язок компонентів методу класифікації зображень

З урахуванням реалізованого підходу до виявлення фейкових зображень за допомогою глибинного навчання та інструментів комп'ютерного зору, було розроблено логічну структуру інформаційної системи, яка відображає основні етапи обробки, аналізу та класифікації зображень з метою оцінки їх автентичності. Ця структура враховує послідовність проходження даних через ключові компоненти системи - від попередньої обробки зображення до прийняття рішення на основі результатів моделі. Візуальне представлення цієї структури наведено на (рисунку 2.5).

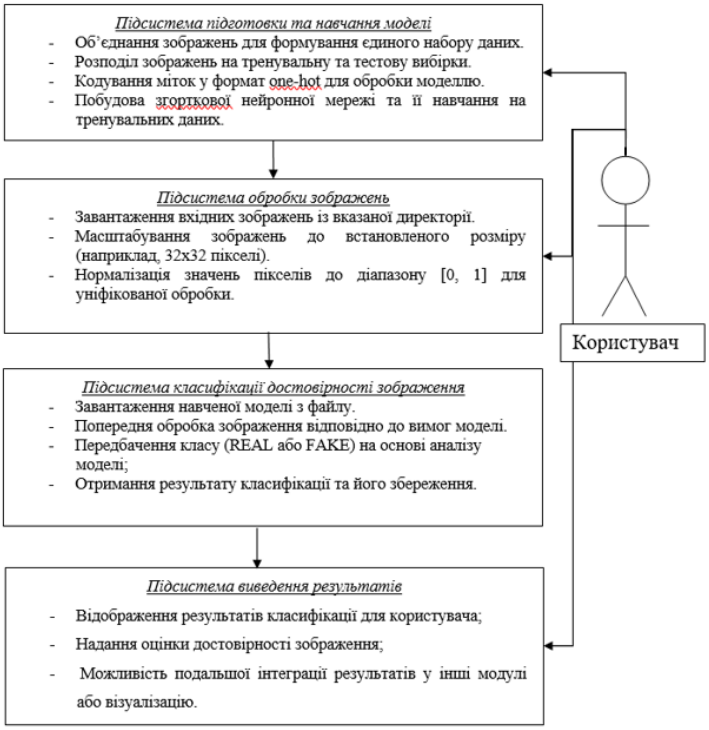


Рисунок 2.5 – Інформаційна схема методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання

Інформаційна система методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання реалізує низку етапів, що виконуються за участю користувача.

На першому етапі здійснюється підготовка даних. Зображення зчитуються з папок REAL і FAKE, змінюється їхній розмір (заданий моделлю) і нормалізуються значення пікселів до діапазону [0, 1]. Кожному зображенню присвоюється мітка (0 для реального, 1 для фейкового), після чого дані об'єднуються, перетасовуються та розбиваються на тренувальну і валідаційну вибірки.

Після цього створюється архітектура згорткової нейронної мережі, яка включає основні шари для обробки зображень. Модель тренується на підготовленому наборі даних і зберігається для подальшого використання.

На етапі класифікації користувач завантажує тестове зображення через графічний інтерфейс. Зображення проходить попередню обробку (зміна розміру, нормалізація), а навчена модель виконує прогноз, повертаючи ймовірність належності до класу REAL або FAKE.

Результат відображається у зручному форматі: користувач отримує інформацію про передбачений клас та ймовірність. Ці дані можуть бути використані для подальшої обробки або інтеграції в інші системи.

Таким чином, система забезпечує повний цикл обробки – від підготовки даних до класифікації зображень та виведення результату.

2.5 Підготовка робочих вхідних даних для системи розпізнавання зображень згенерованих нейромережами

Для ефективної роботи системи класифікації фейкових зображень необхідно попередньо сформувати навчальні та тестові дані у відповідному форматі. Підготовка даних здійснюється як із локальних джерел, так і з використанням публічно доступних датасетів. Система підтримує можливість інтерактивного завантаження зображень користувачем через графічний інтерфейс

або використання наперед сформованих масивів даних, адаптованих під вимоги моделі. Усі зображення проходять уніфіковану попередню обробку та розмітку.

Основними джерелами даних є:

1. Локальні зображення: Користувач може завантажувати власні зображення, розділені на дві категорії - реальні (REAL) та фейкові (FAKE). Наприклад, реальні фотографії можуть бути зроблені камерою смартфона, а фейкові - згенеровані за допомогою моделей типу StyleGAN, Midjourney або DALL·E.

2. Відкриті датасети: Для забезпечення об'єктивного тренування й тестування моделі використовуються відомі набори зображень з відкритим доступом:

– Fake and Real Face Detection Dataset - Містить 2000 фотографій облич із високою роздільною здатніст. Вибірка вважається достовірною для формування класу реальних чи фейкових зображень зображень.Формат [41]: PNG.Мітки: додаються вручну - 0 (реальні зображення), 1 (фейкові зображення).

Попередня обробка даних включає такі етапи:

1. Завантаження: Користувач через графічний інтерфейс обирає дві папки зображень. Програма автоматично зчитує всі файли у форматах PNG, JPG, JPEG та проводить базову валідацію.

2. Фільтрація та нормалізація:Зображення приводяться до розміру 64×64 пікселів із трьома кольоровими каналами (RGB). Дані масштабуються до діапазону [0, 1], тип - float32.

3. Розмітка:Залежно від розташування (у папці REAL або FAKE), кожному зображенню присвоюється мітка - 0 або 1.

4. Розподіл даних:Всі зображення випадковим чином поділяються на навчальну (80%) і валідаційну (20%) вибірки. Наприклад, при 10 000 зображеннях - 8 000 ідуть на тренування, 2 000 - на перевірку.

5. Аугментація:Для підвищення якості навчання застосовуються методи аугментації: випадкове горизонтальне відображення, зсуви, обертання до 20°,

масштабування й зум. Це дозволяє розширити тренувальний датасет без генерації нових зображень.

Використання цієї функції у системі передбачає завантаження тензорів під час початку тренування моделі. Для цього вбудовано автоматизований модуль формування датасету, який дозволяє швидко змінювати джерела, перевіряти якість даних, а також масштабувати підготовлений набір відповідно до доступної обчислювальної потужності. За потреби користувач може задати максимальну кількість зображень на клас - наприклад, по 2 000 реальних і фейкових для швидшого експериментального тестування.

Розроблена система має гнучкий механізм формування навчального набору даних, що дозволяє адаптувати її до різних сценаріїв - від ручного додавання власних зображень до повноцінного використання потужних відкритих датасетів. Це забезпечує як точність, так і універсальність нейронної мережі в задачі класифікації фейкових зображень.

2.6 Особливості використання спеціалізованих програмних компонентів методу виявлення згенерованих зображень

У процесі розробки програмної системи для класифікації зображень на фейкові та справжні було використано низку спеціалізованих бібліотек і програмних компонентів, що забезпечили ефективну реалізацію як етапів машинного навчання, так і зручної взаємодії з користувачем.

Основною бібліотекою для реалізації згорткової нейронної мережі є TensorFlow. Цей фреймворк надає зручні інструменти для побудови, компіляції та тренування моделі. У роботі використано послідовну модель з шарами Conv2D, MaxPooling2D, Dropout, Flatten, Dense. Компіляція виконувалась за допомогою оптимізатора Adam, функції втрат binary_crossentropy, а точність (accuracy) використовувалася як метрика ефективності. Для попередньої обробки та подачі зображень застосовувався модуль ImageDataGenerator.

Для підготовки вхідних зображень використовувалась бібліотека OpenCV - з її допомогою здійснювалося зчитування файлів, зміна розміру до необхідного формату (наприклад, 64×64), а також приведення зображень до відповідного колірного простору (RGB). Додатково використовувалась бібліотека Pillow (PIL) для масштабування зображень при відображенні в інтерфейсі користувача - зокрема, до розміру 800×600 пікселів у вікні перегляду.

Для поділу даних на тренувальні та валідаційні набори використовувалась бібліотека scikit-learn, зокрема функція train_test_split, яка дозволяє випадковим чином формувати збалансовані вибірки (з параметрами test_size=0.2, random_state=42). Цей етап є критично важливим для уникнення перенавчання моделі та забезпечення її здатності до генералізації.

У процесі навчання було важливо контролювати точність і прогрес моделі, для чого використовувалась бібліотека matplotlib. Вона дала змогу візуалізувати зміни точності (accuracy) і функції втрат (loss) на кожній епосі, що допомогло вчасно виявити можливе перенавчання або недотренованість моделі.

Для забезпечення стабільності роботи системи було інтегровано бібліотеку psutil, яка дозволяла моніторити використання оперативної пам’яті та уникати аварійних завершень роботи при обробці великих обсягів зображень. Бібліотека numpy застосовувалась для роботи з масивами зображень, їхньої нормалізації, обчислень середніх значень.

Крім того, для реалізації графічного інтерфейсу користувача було використано стандартну бібліотеку tkinter, яка забезпечила зручну побудову вікон, кнопок, меню та елементів навігації між сторінками. Інтерфейс забезпечує завантаження зображення, передачу його до нейронної мережі, виведення результатів класифікації та додаткову інформацію про оброблене зображення.

Спроектована система є інтегрованим середовищем, що поєднує в собі функціонал штучного інтелекту, комп’ютерного зору та зручний інтерфейс користувача. Поєднання бібліотек TensorFlow, OpenCV, Pillow, scikit-learn, matplotlib, psutil, numpy та tkinter дозволило реалізувати повноцінну

функціональну інформаційну систему, здатну ефективно виконувати завдання з класифікації зображень.

2.7 Висновки до розділу 2

В другому розділі було виконано наступні задічі:

– Створено схему методу класифікації фейкових зображень з використанням технологій комп'ютерного зору;

– визначено функціональної системи методу визначення фейкових зображень;

– розроблено архітектуру методу визначення зображень за допомогою комп’ютерного зору;

– побудована інформаційна схема методу класифікації фейкових зображень;

– визначено навчальний датасет для визначення штучно створених зображень.

Наведені пункти дозволяють чітко визначити який саме функцірнал має мати розроблюваний метод визначення згенерованих зображень.

Розділ 3 Програмна реалізація системи класифікації фейкових зображень із використанням технологій комп'ютерного зору

3.1 Визначення шляхів дослідження та засобів створення методу класифікації фейкових зображень

У межах даного дослідження передбачено створення програмної системи, призначеної для вивчення та оцінки точності методу класифікації фейкових зображень, що був детально розглянутий і обґрунтований у теоретичному розділі роботи. Розробка цієї системи здійснюється із застосуванням мови програмування Python, що є одним із найпоширеніших інструментів у сфері машинного навчання, а також бібліотеки TensorFlow, яка забезпечує широкі можливості для побудови та тренування глибоких нейронних мереж.

Програмна система методу реалізує повний цикл обробки візуальних даних - від завантаження та підготовки зображень до навчання моделі та подальшої класифікації. Вона включає функціонал для імпорту локальних датасетів, автоматизовану попередню обробку вхідних зображень, процедури аугментації даних з метою покращення узагальнюючої здатності моделі, а також побудову згорткової нейронної мережі, адаптованої для розв'язання поставленого завдання. Крім того, реалізовано можливість візуалізації основних метрик ефективності, таких як точність, повнота, F1-міра, що дозволяє здійснювати глибокий аналіз результатів і робити висновки щодо продуктивності обраного підходу до класифікації.

3.2 Засоби розробки інформаційної системи визначення згенерованих зображень

Для реалізації програмної системи класифікації фейкових зображень виникла потреба у ретельному виборі відповідних засобів розробки, які б забезпечували ефективну та узгоджену роботу системи на всіх етапах її створення та функціонування. Це стосується не лише розробки логіки обробки зображень та

побудови моделі класифікації, а й ефективного управління даними, налаштування програмного середовища, а також забезпечення зручності написання, відлагодження та підтримки коду. Обрані інструменти мають відповідати сучасним вимогам до продуктивності, масштабованості та підтримки бібліотек глибокого навчання, що є критично важливим для реалізації поставленої задачі.

Мова програмування - Python 3.8+, яка є стандартом де-факто у сфері аналізу зображень і глибокого навчання. Її синтаксис забезпечує швидке створення прототипів і читаємий код, що важливо для експериментального етапу [42].

Основним середовищем розробки обрано PyCharm Community Edition [43] - воно забезпечує зручне автодоповнення, візуалізацію структури коду та ефективне налагодження, що критично при роботі з великою кількістю експериментальних конфігурацій.

3.3 Структура та функціональне призначення програмних складових системи класифікації зображень

Архітектура програмної системи побудована за модульним принципом із застосуванням шаблону проєктування Model-View-Controller (MVC), який дозволяє чітко розмежувати функціональні обов’язки між окремими компонентами, забезпечити логічну структуру програмного коду та суттєво спростити процеси тестування, відлагодження і масштабування системи. Такий підхід сприяє підвищенню гнучкості при внесенні змін, полегшує супровід програмного забезпечення, а також покращує загальну організацію коду та взаємодію між елементами системи.

Основні функціональні компоненти програмного продукту охоплюють всі ключові етапи роботи з візуальними даними, а саме: завантаження вхідних зображень підготовку та попередню обробку даних, тренування нейронної мережі за допомогою методів глибокого навчання, а також виконання класифікації нових зображень і виведення результатів. Реалізація зазначених етапів здійснена у

вигляді окремих модулів, кожен з яких відповідає за певну частину функціоналу, що сприяє структурованості та забезпечує зручність при реалізації експериментальних досліджень.

До складу системи входить п’ять основних модулів: App, MainMenu, TrainingWindow, RecognitionWindow, CNNModel. Вони реалізовані у вигляді окремих класів, функціонально пов’язаних між собою через об’єктно-орієнтовані зв’язки, подані на діаграмі модулів (рис. 3.1).

Модуль App виконує роль головного координатора: він ініціалізує вікна, забезпечує навігацію між формами та зберігає об’єкт нейронної мережі для передачі в інші модулі. Це дозволяє централізовано керувати станом застосунку та уникати дублювання коду. Крім того, модуль відповідає за виклик функцій завантаження зображень, запуск процесу навчання та класифікації, а також обробку виняткових ситуацій, забезпечуючи узгоджену та стабільну роботу всієї системи.

Модуль MainMenu слугує стартовим інтерфейсом системи. Він дозволяє користувачу перейти до навчання або розпізнавання, а також завершити роботу програми. Цей модуль реалізовано як графічне представлення (View), що взаємодіє з користувачем через кнопки, обробляючи події натискання та викликаючи відповідні функції переходу, ініційовані головним модулем.

Модуль TrainingWindow надає функціонал для завантаження датасету, налаштування параметрів нейромережі та запуску тренування. Він реалізований як інтерфейс з вбудованою логікою взаємодії з моделлю. Прогрес навчання відображається в режимі реального часу завдяки використанню потоків (threading) і черг (queue.Queue).

Модуль RecognitionWindow відповідає за процес розпізнавання окремих зображень. Користувач має можливість обрати файл з локальної системи, переглянути його у вікні застосунку та запустити процес класифікації. Прогноз здійснюється на основі попередньо навченої моделі, а результат - ймовірність належності зображення до фейкових або справжніх - миттєво відображається у цьому ж вікні.

Модуль CNNModel реалізує всю логіку побудови, навчання та застосування згорткової нейронної мережі. Його структура включає шари Conv2D, Dense, MaxPooling, а також параметри оптимізації (кількість епох, швидкість навчання тощо). Методи train() і predict() є ключовими для взаємодії з іншими модулями. Модель зберігається у форматі .h5, що дозволяє повторно її використовувати без повторного навчання.

Передача даних між модулями реалізується через посилання на об’єкти Python: масиви NumPy для зображень, словники з параметрами навчання та черги для передачі статусу виконання. Механізм асинхронної обробки забезпечує стабільну роботу навіть при обробці великих обсягів вхідної інформації. Стан моделі, шляхи до файлів, результати прогнозу та параметри зберігаються як атрибути відповідних класів.

На рисунку 3.1 подано модульну діаграму системи, яка демонструє взаємозв’язки між основними компонентами. У свою чергу, на рисунку 3.2 наведено діаграму класів, що описує структуру та методи кожного модуля.

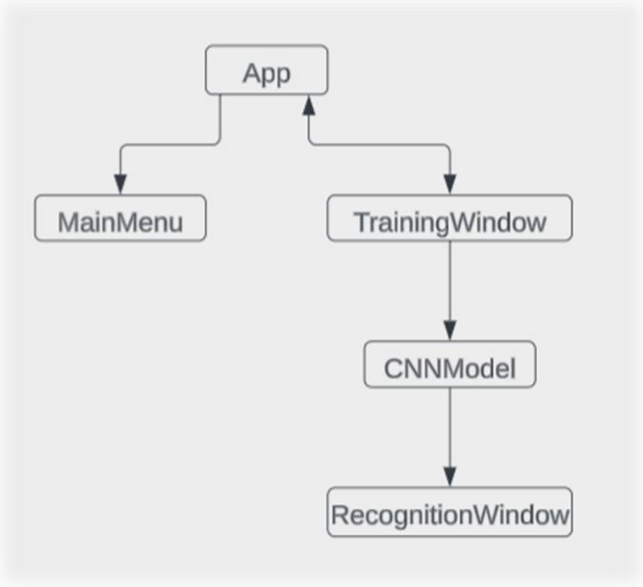


Рисунок 3.1 – Діаграма модулів системи

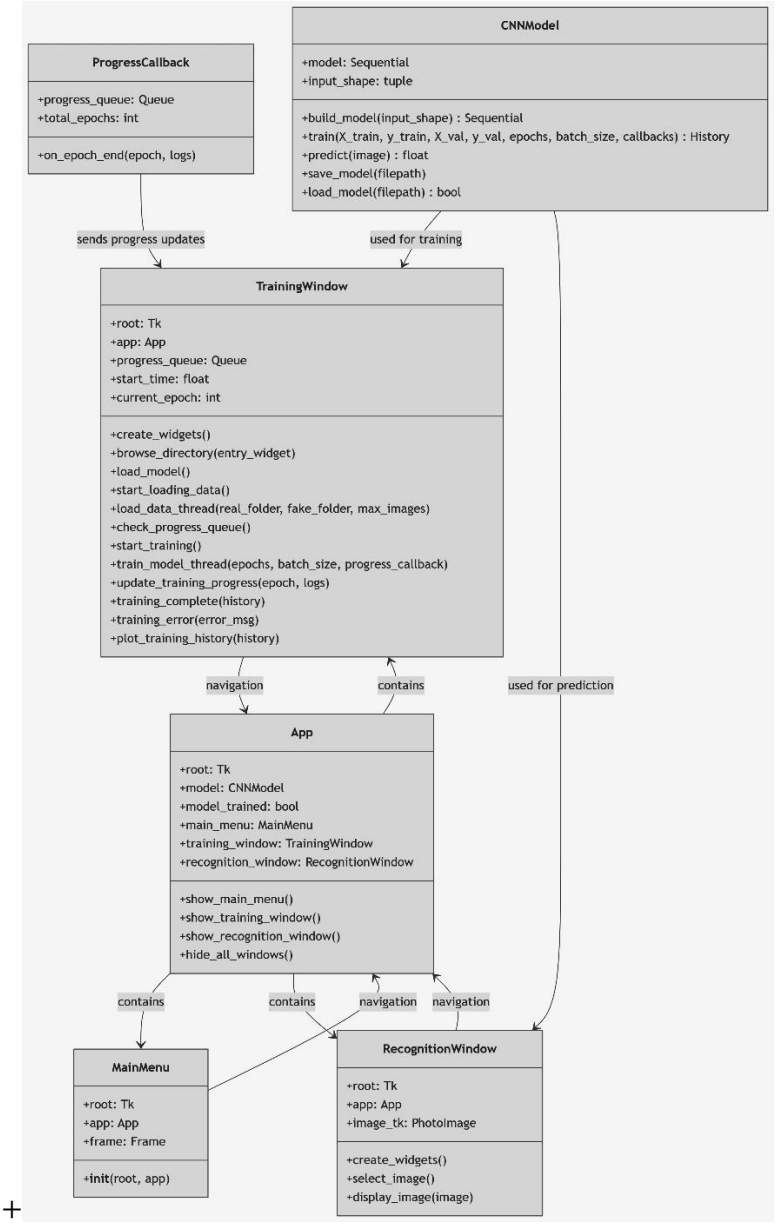


Рисунок 3.2 – Діаграма класів системи

Реалізована архітектура програмної системи демонструє високий рівень модульності, що забезпечує чітке розділення обов’язків між компонентами, полегшує супровід коду та дозволяє ефективно управляти складністю проєкту. Завдяки використанню шаблону Model-View-Controller досягнуто ізоляції бізнес-логіки від елементів інтерфейсу користувача, що значно спрощує процес тестування, налагодження та подальшої розробки. Крім того, структура системи підтримує повторне використання коду, що зменшує обсяг дублювання і підвищує загальну ефективність розробки.

Обрана архітектурна модель сприяє гнучкості та масштабованості рішення - за потреби можна легко додавати нові функціональні модулі, інтегрувати додаткові алгоритми класифікації або змінювати існуючі компоненти без істотного впливу на загальну структуру. Це особливо важливо в контексті досліджень, пов’язаних із фейковими зображеннями, оскільки ця галузь активно розвивається, і система має бути готовою до адаптації. Побудована архітектура забезпечує надійний фундамент для довготривалого використання, експериментування та розширення функціональних можливостей програмної системи.

3.4 Особливості реалізації програмних складових системи розпізнавання зображень

Програмна реалізація системи розпізнавання фейкових зображень охоплює низку ключових етапів: завантаження та попередню обробку даних, навчання моделі згорткової нейронної мережі, а також класифікацію нових зображень за допомогою отриманої моделі. Ці етапи реалізовано у вигляді окремих функціональних модулів, які взаємодіють між собою через спільну модельну архітектуру.

Після запуску програми користувач через інтерфейс обирає директорії зображень двох класів – «REAL» і «FAKE». Система сканує обрані папки, фільтруючи файли за підтримуваними форматами (PNG, JPG, JPEG), після чого зображення масштабується до розміру 64×64 пікселів. Для подальшої обробки значення пікселів нормалізуються до діапазону [0,1], що забезпечує коректне подання на вхід нейронної мережі. При завантаженні великої кількості зображень система оцінює доступний обсяг оперативної пам’яті. Якщо вільна пам’ять нижча за встановлений поріг (наприклад, 1000 МБ), виводиться попередження, щоб уникнути аварійного завершення процесу.

Для підвищення зручності користувача весь процес реалізовано з використанням асинхронного підходу та мультипотоковості. Це дозволяє не

блокувати графічний інтерфейс під час інтенсивних операцій. Візуальний прогрес завантаження відображається у вигляді числового індикатора. На скріншоті нижче показано інтерфейс під час завантаження 2000 зображень, по 1000 на кожен клас - на екрані видно, що завантажено 651 з 1000 реальних зображень.

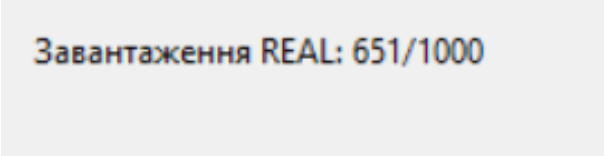


Рисунок 3.3 – Процес завантаження датасету для навчання

Після завершення етапу попередньої обробки зображень користувач має можливість перейти до наступного кроку - тренування згорткової нейронної мережі. Детальний опис архітектури моделі наведено у попередньому розділі, а реалізація забезпечує гнучке налаштування основних параметрів процесу навчання, зокрема кількості епох, розміру партії та інших технічних характеристик. Перед подачею до моделі вхідні дані проходять додаткову стадію обробки - аугментацію, яка включає випадкові повороти, горизонтальне віддзеркалення, масштабування та інші трансформації, що покращують узагальнюючу здатність мережі. У ході навчання візуалізуються графіки зміни точності та втрат на кожній епосі, що дає змогу в реальному часі оцінювати прогрес моделі та своєчасно виявляти ознаки перенавчання чи недостатнього навчання.

Для запобігання перенавчанню використано механізми автоматичного збереження найкращої моделі (ModelCheckpoint) та дострокового завершення навчання у разі відсутності покращень (EarlyStopping). Завдяки використанню потокової обробки інтерфейс залишається активним під час тренування, а користувач може спостерігати за ходом процесу. Як видно на рисунку 3.4, під час 22-ї з 30 епох на екрані відображено поточну точність (93%) та втрати (0.14).

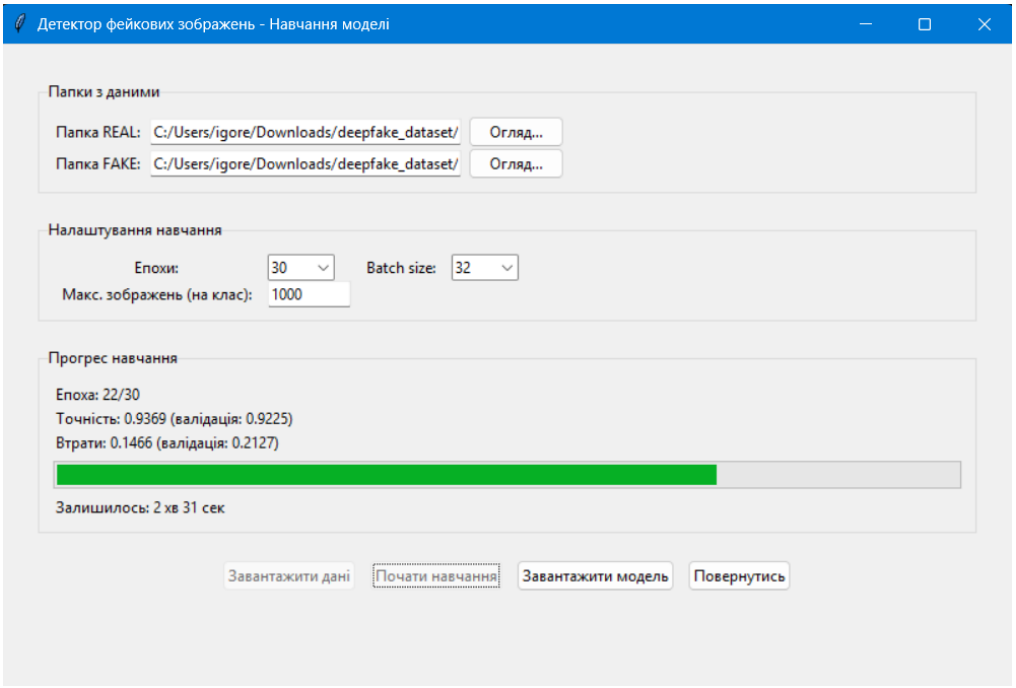


Рисунок. 3.4– Процес навчання моделі

Після навчання система переходить до режиму розпізнавання. Користувач завантажує окреме зображення для аналізу, яке автоматично масштабується до розміру, зручного для перегляду, і відображається на канвасі з підтримкою прокрутки. Після обробки модель повертає ймовірність належності зображення до фейкових. Цей результат виводиться у вигляді короткого текстового повідомлення – наприклад, «Фейкове, 92% впевненості». При цьому реалізовано перевірку коректності обраного файлу, наявності моделі, а також обробку типових помилок із відповідними повідомленнями у вікні програми.

Інтеграція окремих модулів реалізована через спільні об’єкти та черги (queue.Queue), що дозволяє обмінюватися даними між частинами програми без блокувань. Наприклад, результати попередньої обробки одразу передаються в навчальний модуль, а збережена модель миттєво доступна для модуля класифікації. Завдяки цьому програма працює як єдине цілісне середовище.

На скріншоті головного меню показано інтерфейс після завершення навчання: статус «Модель готова», активні кнопки «Навчання» та «Розпізнавання», а також заголовок «Детектор фейкових зображень», що підкреслює основну мету системи.

Реалізовані алгоритми та функціональні компоненти забезпечують стабільну, зручну та наочну роботу із системою. Комбінація нейронного моделювання, ефективної обробки зображень та зрозумілого графічного інтерфейсу робить програмний продукт готовим до реального використання як прикладний інструмент.

3.5 Аналіз функціональності системи розпізнавання зображень

Щоб користувач міг користуватися даною програмою він повинен запустити її через ярлик, після чого йому відкриється головне вікно де він зможе обрати між навчанням моделі і розпізнавання зображення. Натиснувши кнопку «Навчання моделі» відкриється вікно в якому вказавши всі необхідні параметри можна почати навчання моделі, деякі параметри встановленні за замовчуванням але користувач може їх змінити під свої потреби (рисунок 3.5).

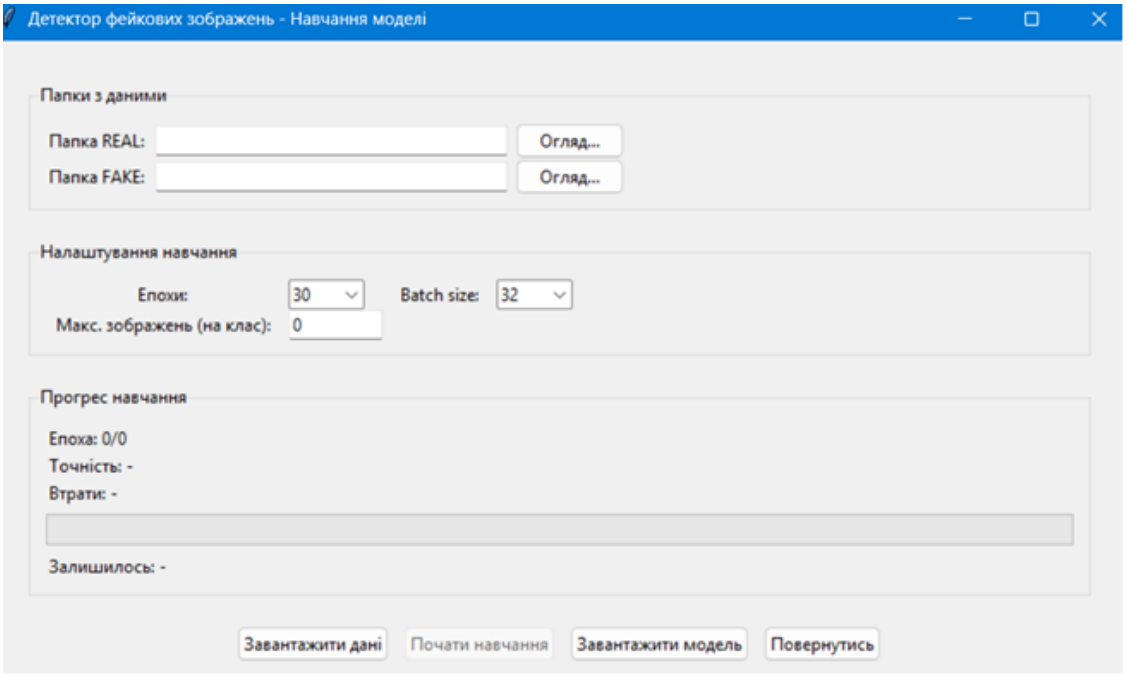


Рисунок 3.5 – Вікно для навчання нейронної моделі

Натиснувши кнопку «Розпізнавання зображень» відкриється вікно в якому можна обрати зображення, після чого метод відразу класифікує зображення і ймовірність (рисунок 3.6).

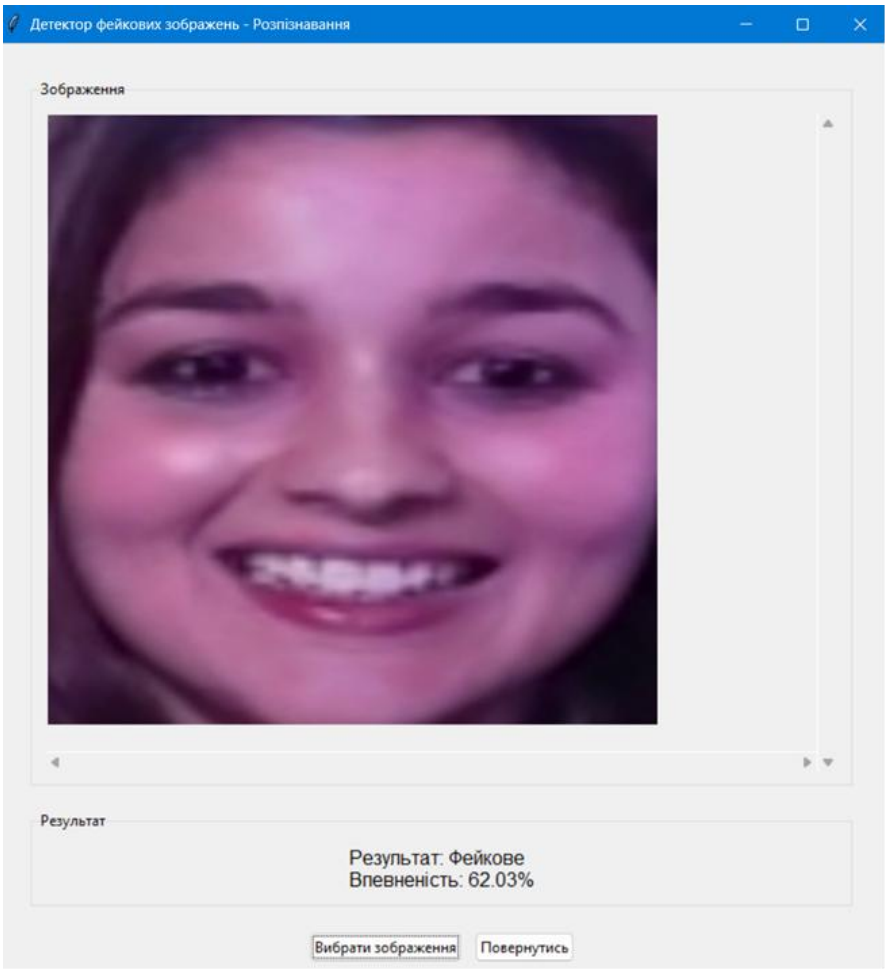


Рисунок 3.6 – Вікно класифікації зображення

Таким чином можна впевнитися що метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання є простим у використані, що дозволяє новим користувачам дуже легко використовувати створений метод який дозволяє заощадити час для вирішення проблем з зображеннями.

3.6 Результати досліджень методу класифікації зображень

Після завершення етапу створення функціональної складової розробленого методу було здійснено серію комплексних експериментальних досліджень. Основною метою цих досліджень є перевірка та об’єктивне оцінювання якості класифікації зображень на основі побудованої та реалізованої моделі глибинного

навчання. З цією метою було використано набір кількісних метрик, які забезпечують формалізований підхід до визначення точності роботи моделі. Серед ключових показників, що враховувалися при аналізі - загальна точність класифікації, значення функції втрат (loss), показник повноти (recall), а також інші інформативні характеристики, що відображають здатність моделі коректно відрізняти справжні зображення від згенеровани.

Окрему увагу було приділено всебічному тестуванню точності функціонування розробленого методу в цілому. У процесі валідації перевірено коректність роботи всіх основних складових системи, зокрема модулів, відповідальних за завантаження зображень, попередню обробку вхідних даних, запуск процесу навчання та тестування моделі, а також генерацію та відображення результатів класифікації. Крім того, здійснено оцінку стабільності та надійності функціонування програмної системи за умов змінного рівня навантаження, що дало змогу переконатися в її здатності працювати без збоїв у різних сценаріях використання.

Для оцінки точності розробленої системи детекції фейкових зображень було проведено комплексне дослідження на збалансованому датасеті Fake and Real Face Detection Dataset, що включає 2000 зображень, з яких по 1000 на кожен клас. Такий підхід дозволяє оцінити систему в умовах, максимально наближених до реальних сценаріїв застосування.

Етапи досліджень включали:

– проведення навчання моделі;

– збір та аналіз метрик якості класифікації;

– оцінку впливу методів попередньої обробки даних (аугментація, нормалізація);

– побудову графіків динаміки тренування;

– створення матриці сплутаності для аналізу помилок;

– порівняння різних конфігурацій (із аугментацією та без).

Основні показники якості роботи наведені на рисунку 3.7.

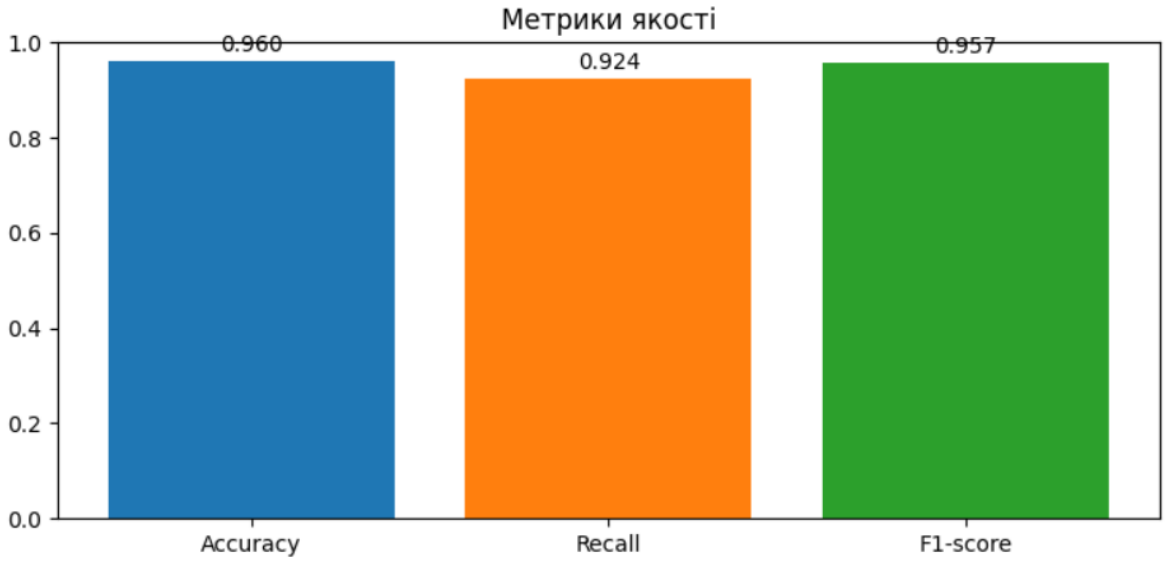


Рисунок 3.6 – Метрики якості навчання

Отримані результати демонструють високий рівень якості класифікації зображень як на тренувальній, так і на тестовій вибірках. З поміж усіх оцінюваних метрик найбільш інформативним та збалансованим показником виступає F1-міра, яка поєднує у собі як точність класифікації, так і повноту виявлення, тим самим забезпечуючи комплексну оцінку точності роботи моделі.

Зміна точності (рисунок 3.7) й втрат (рисунок 3.6) на етапі тренування та валідації представлені на діаграмах:

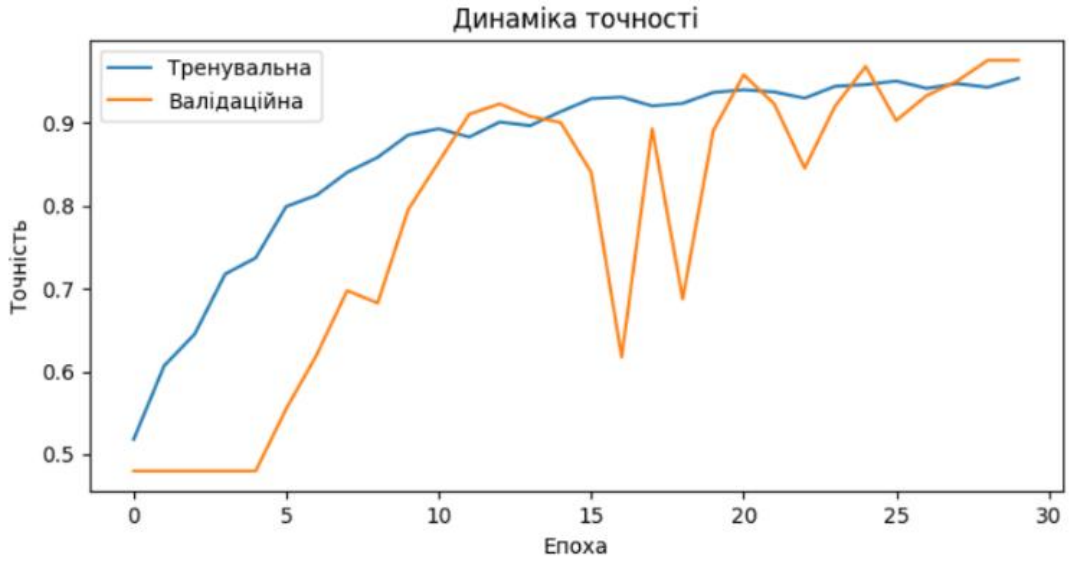


Рисунок 3.7 – Діаграма динаміки точності навчання моделі

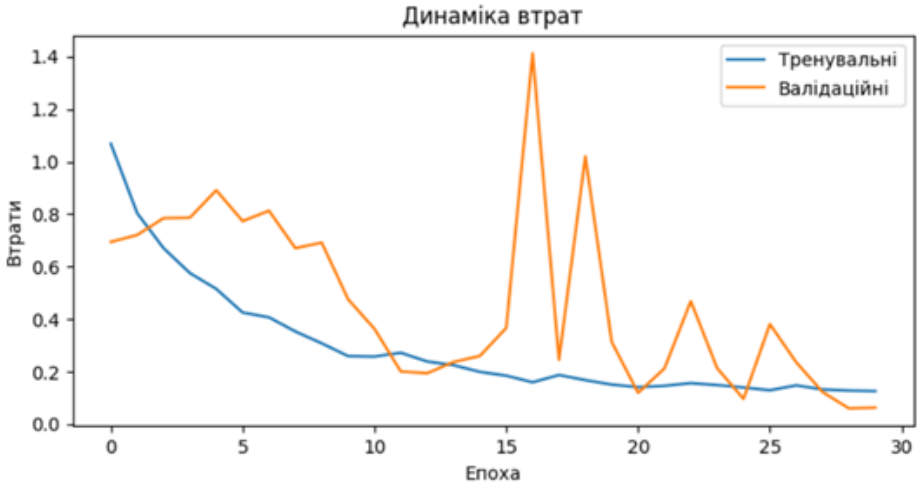


Рисунок 3.8 – Діаграма динаміки втрат при навчанні моделі

Як видно з графіку, модель швидко навчається у перші 10 епох, далі відбувається поступова стабілізація. Втрати зменшуються, точність зростає, що вказує на відсутність перенавчання.

Матриця помилок (рисунок 3.9) дозволила детальніше оцінити помилки моделі:

Рисунок 3.9 – Матриця помилок

Аналіз отриманої матриці помилок свідчить про те, що модель має тенденцію до дещо вищого рівня хибно-негативних класифікацій у випадках фейкових зображень. Це означає, що окремі підроблен і зображення помилково ідентифікуються як справжні. Така поведінка моделі переважно зумовлена

наявністю високоякісних прикладів, згенерованих за допомогою сучасних генеративних алгоритмів (наприклад GAN), які здатні відтворювати дуже правдоподібні візуальні характеристики. Цей фактор ускладнює процес виявлення ознак штучності, оскільки візуальні артефакти стають менш вираженими або повністю відсутніми. Водночас виявлена особливість моделі підкреслює необхідність подальшого вдосконалення архітектури та використання додаткових ознак чи комбінованих підходів до аналізу, зокрема тих, що враховують контекст зображення або метадані.

Ще одним важливим аспектом дослідження стало використання аугментації, яка включала обертання, зміну яскравості та контрастності, додавання шуму (таблиця 3.1).

Таблиця 3.1 – Вплив аугментації на навчання

Конфігурація	**Точність**	**F1-міра**
Без аугментації	0.912	0.89
З аугментацією	0.96	0.95

Як видно з таблиці аугментація позитивно вплинула на загальні показники.

Таблиця 3.2 – Вплив кількості епох на точність

Кількість епох	**Точність (%)**
10	85.2
20	91.4
30	96.5

Таблиця демонструє, що зі збільшенням кількості епох точність покращуються. Оптимальний результат досягається після 20 епохи.

Результати проведених експериментальних досліджень підтверджують ефективність запропонованого підходу до детекції фейкових зображень. Розроблена система продемонструвала високий рівень точності та надійності при

класифікації зображень різного типу та походження, що вказує на її універсальність і практичну цінність. Досягнуті значення основних метричних показників - зокрема, точність класифікації на рівні 96% та F1-міра 0.95 - свідчать про збалансовану роботу моделі з урахуванням як точності, так і повноти розпізнавання.

Особливо важливою є роль аугментації в підвищенні якості класифікації: застосування методів штучного розширення навчального набору дозволило підвищити загальну узагальнювальну здатність моделі та зменшити ризик перенавчання. Крім того, розроблена модифікована архітектура нейронної мережі забезпечила стабільну роботу системи під час тренування та прогнозування, а також створила передумови для подальшого масштабування та адаптації до більш складних задач. Отримані результати підтверджують перспективність використання розробленої системи в реальних умовах для автоматизованого виявлення підроблених або штучно згенерованих зображень.

3.7 Висновки до розділу 3

У результаті реалізації, тестування та дослідження процесу класифікації фейкових зображень із застосуванням технологій комп'ютерного зору та глибинного навчання було підтверджено як функціональну працездатність, так і точність розробленого методу. Створена система вирізняється модульною структурою з чітко визначеним розподілом функціональних компонентів, що охоплюють інтерфейси для навчання моделі та розпізнавання зображень, а також засоби управління процесом роботи згорткової нейронної мережі. Використання мови програмування Python у поєднанні з бібліотеками для глибокого навчання надало можливість забезпечити високу гнучкість, розширюваність та масштабованість розробленої інформаційної системи.

У процесі тестування розробленої системи було досягнуто високого рівня точності класифікації фейкових зображень - до 96% при застосуванні методів аугментації даних, що сприяють покращенню узагальнюючої здатності моделі.

Функціональне тестування підтвердило коректність і стабільність роботи всіх основних компонентів системи, зокрема при обробці некоректних або частково відсутніх вхідних даних. Результати експериментальних досліджень засвідчили, що обрана архітектура нейронної мережі демонструє вищу точність у порівнянні з базовими моделями, зберігаючи оптимальний баланс між точністю класифікації та швидкістю процесу навчання.

Інтерфейс розробленої інформаційної системи вирізняється зручністю використання та інтуїтивною зрозумілістю, що дозволяє користувачеві легко здійснювати всі необхідні дії відповідно до закладених сценаріїв взаємодії. Система забезпечує візуалізацію процесу класифікації та відображення результатів у наочній формі, що сприяє кращому сприйняттю і контролю за виконанням задач. Передбачено функціонал для збереження обчислених метрик і графічних результатів. Додатково було проведено експериментальні дослідження, спрямовані на вивчення впливу різних конфігурацій параметрів і підходів до обробки даних. Це дозволило оцінити точність системи за критеріями точності, продуктивності та стійкості до помилкових або непередбачених сценаріїв роботи.

Розроблена інформаційна система повною мірою відповідає визначеним вимогам щодо функціональності, надійності та точності класифікації. Її архітектура і реалізовані модулі забезпечують ефективне виявлення згенерованих зображень. Завдяки цьому система може бути використана не лише як практичний інструмент, але й як основа для подальших наукових досліджень і вдосконалення методів виявлення фейкових візуальних даних.

Загальні висновки

У результаті виконання кваліфікаційної роботи бакалавра було розроблено метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання. Метод був реалізований за допомогою мови програмування Python з використанням бібліотек TensorFlow, OpenCV і tkinter, що забезпечують навчання моделі, обробку зображень та зручний графічний інтерфейс. Метод включає функціонал для завантаження даних, навчання моделі, збереження/завантаження моделі та аналізу зображень на класи REAL (реальні) і FAKE (фейкові).

Поставлені задачі та вимоги щодо створення методу автоматичної класифікації зображень виконані повністю, а саме:

– Проведено аналіз актуальної розробки методу класифікації зображень.

– Проведено аналіз теоретичних та практичних підходів які дозволяють визначити зображення.

– Розроблено метод для визначення фейкових зображень.

– Створено інформаційну систему методу визначення згенеровоних зображень.

– Проведено тестування та оцінення точності системи класифікації на реальних і згенерованих зображеннях.

Розроблений метод дозволяє користувачам легко навчати модель і перевіряти достовірність зображень, що робить його доступним для широкого кола користувачів.

Розроблений метод має потенціал для використання в реальних задачах, таких як перевірка достовірності зображень у медіа чи соціальних мережах. У майбутньому планується вдосконалення програми шляхом збільшення набору даних, покращення архітектури моделі та розширення функціоналу інтерфейсу, що зробить метод предметом подальшої роботи.

Перелік посилань

1. The Science of Visual Data Communication: What Works / S. L. Franconeri et al. *Psychological Science in the Public Interest*. 2021. Vol. 22, no. 3. P. 110–161.

2. Reinforced Neighborhood Selection Guided Multi-Relational Graph Neural Networks / H. Peng et al. ACM Transactions on Information Systems. 2022. Vol. 40, no. 4. P. 1–46.

3. AQUiD: Automated Quality Assessment Using Digital Image Processing. International Journal for Research in Engineering Application & Management. 2020. P. 295–300..

4. V D., D P.-T., L R. Comprehensive digital image analysis to detect manipulation. Artificial Intelligence. 2025. Vol. 30, AI.2025.30(1). P. 77–83.

5. Liu C., others. A survey of deep learning techniques for image forgery detection. Journal of Visual Communication and Image Representation. 2021. Vol. 82. P. 103289.

6. Mrs Supriya Shree, Riddhi Arya, Saket Kumar Roy. Investigating the Evolving Landscape of Deepfake Technology: Generative AI's Role in it's Generation and Detection. International Research Journal on Advanced Engineering Hub (IRJAEH). 2024. Vol. 2, no. 05. P. 1489–1511.

7. How to identify fake images? : Multiscale methods vs. Sherlock Holmes / M. Park et al. Communications for Statistical Applications and Methods. 2021. Vol. 28, no. 6. P. 583–594.

8. Abdelhamed M., El-Rabie E.-S., Baumy A. A Comprehensive Survey on Passive Techniques for Digital Image Forgery Detection. Menoufia Journal of Electronic Engineering Research. 2024. P. 9–21.

9. Pathak M. V. Processing-In-Memory Techniques: Survey, Advances, and Challenges. International Journal for Research in Applied Science and Engineering Technology. 2024. Vol. 12, no. 5. P. 4956–4970.

10. IEEE Proceedings. IEEE Industry Applications Magazine. 2022. Vol. 28, no. 5. P. 92.

11. Li T., Dong X., Lin H. Guided Depth Map Super-Resolution Using Recumbent Y Network. IEEE Access. 2020. Vol. 8. P. 122695–122708.

12. IEEE Transactions on Information Forensics and Security publication information. IEEE Transactions on Information Forensics and Security. 2021. Vol. 16. P. C2.

13. A survey of machine learning techniques in adversarial image forensics / E. Nowroozi et al. Computers & Security. 2021. Vol. 100. P. 102092.

14. Xu B. Improved convolutional neural network in remote sensing image classification. Neural Computing and Applications. 2020.

15. Kaur N., Jindal N., Singh K. Passive Image Forgery Detection Techniques: A Review, Challenges, and Future Directions. Wireless Personal Communications. 2024.

16. Singh S. P. Image Forgery Detection using Deep Learning. International Journal for Research in Applied Science and Engineering Technology. 2024. Vol. 12, no. 11. P. 149–154.

17. Zhu W., Wang X., Gao W. Multimedia Intelligence: When Multimedia Meets Artificial Intelligence. IEEE Transactions on Multimedia. 2020. Vol. 22, no. 7. P. 1823–1835.

18. Identifying Fake Images Through CNN Based Classification Using FIDAC / S. Pawar et al. 2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP), Hyderabad, India, 21–23 July 2022.

19. SMALED2 with BICD2 gene mutations: Report of two cases and portrayal of a classical phenotype / V. Picher-Martel et al. Neuromuscular Disorders. 2020. Vol. 30, no. 8. P. 669–673.

20. Zhao L., Zhang Z. A improved pooling method for convolutional neural networks. Scientific Reports. 2024. Vol. 14, no. 1.

21. Varma P. R., Malathi K. Fake image detection using convolutional neural network and compare the accuracy with Naive Bayes classifier. INTERNATIONAL

CONFERENCE ON NEWER ENGINEERING CONCEPTS AND TECHNOLOGY: ICONNECT-2024, Trichy, India. 2025. P. 020113.

22. Deep Learning-Based Digital Image Forgery Detection Using Transfer Learning / E. U. H. Qazi et al. Intelligent Automation & Soft Computing. 2023. P. 1–10.

23. Şafak E., Barışçı N. Detection of fake face images using lightweight convolutional neural networks with stacking ensemble learning method. PeerJ Computer Science. 2024. Vol. 10. P. e2103.

24. Zhao L., Zhang Z. A improved pooling method for convolutional neural networks. Scientific Reports. 2024. Vol. 14, no. 1.

25. Varma P. R., Malathi K. Fake image detection using convolutional neural network and compare the accuracy with Naive Bayes classifier. INTERNATIONAL CONFERENCE ON NEWER ENGINEERING CONCEPTS AND TECHNOLOGY: ICONNECT-2024, Trichy, India. 2025. P. 020113.

26. Zhao L., Zhang Z. A improved pooling method for convolutional neural networks. Scientific Reports. 2024. Vol. 14, no. 1.

27. Robust forgery detection for compressed images using CNN supervision / B. Diallo et al. Forensic Science International: Reports. 2020. Vol. 2. P. 100112.

28. Şafak E., Barışçı N. Detection of fake face images using lightweight convolutional neural networks with stacking ensemble learning method. PeerJ Computer Science. 2024. Vol. 10. P. e2103.

29. Deep Learning-Based Digital Image Forgery Detection Using Transfer Learning / E. U. H. Qazi et al. Intelligent Automation & Soft Computing. 2023. P. 1–10.

30. Pavithra A., Geetha B. T. Detection of skin cancer using support vector machine classifier compare with convolutional neural network classifier based on accuracy. CONTEMPORARY INNOVATIONS IN ENGINEERING AND MANAGEMENT, Nandyal, India. 2023.

31. Zhao L., Zhang Z. A improved pooling method for convolutional neural networks. Scientific Reports. 2024. Vol. 14, no. 1.

32. CNN-based Approach for Robust Detection of Copy-Move Forgery in Images / A. S et al. Inteligencia Artificial. 2024. Vol. 27, no. 73. P. 80–91.

33. Deepfake detection using convolutional vision transformers and convolutional neural networks / A. H. Soudy et al. Neural Computing and Applications. 2024.

34. Convolutional Neural Network (CNN). Encyclopedia of Computer Graphics and Games. Cham, 2024. P. 483.

35. Classification. Digital Image Processing. Berlin/Heidelberg. P. 533–549.

36. Enhanced Image Forgery Detection using a Hybrid Approach: Integration of ELA, CNN, and XGBoost / K. Sukhmani et al. International Journal of Performability Engineering. 2024. Vol. 20, no. 6. P. 367.

37. Deepfake detection using convolutional vision transformers and convolutional neural networks / A. H. Soudy et al. Neural Computing and Applications. 2024.

38. Perceptual Image Difference. URL: https://stackoverflow.com/questions/29464174/perceptual-image-comparison.

39. Beyond Compare. URL: https://www.scootersoftware.com.

40. Unknown. Diffimg. URL: https://rsload.net/soft/graphics/15383-diffimg.html.

41. Real and Fake Face Detection. URL: https://www.kaggle.com/datasets/ciplab/real-and-fake-face-detection.

42. Python Software Foundation. Python. URL: https://www.python.org.

43. JetBrains. PyCharm. URL: https://www.jetbrains.com/pycharm.

ДОДАТКИ

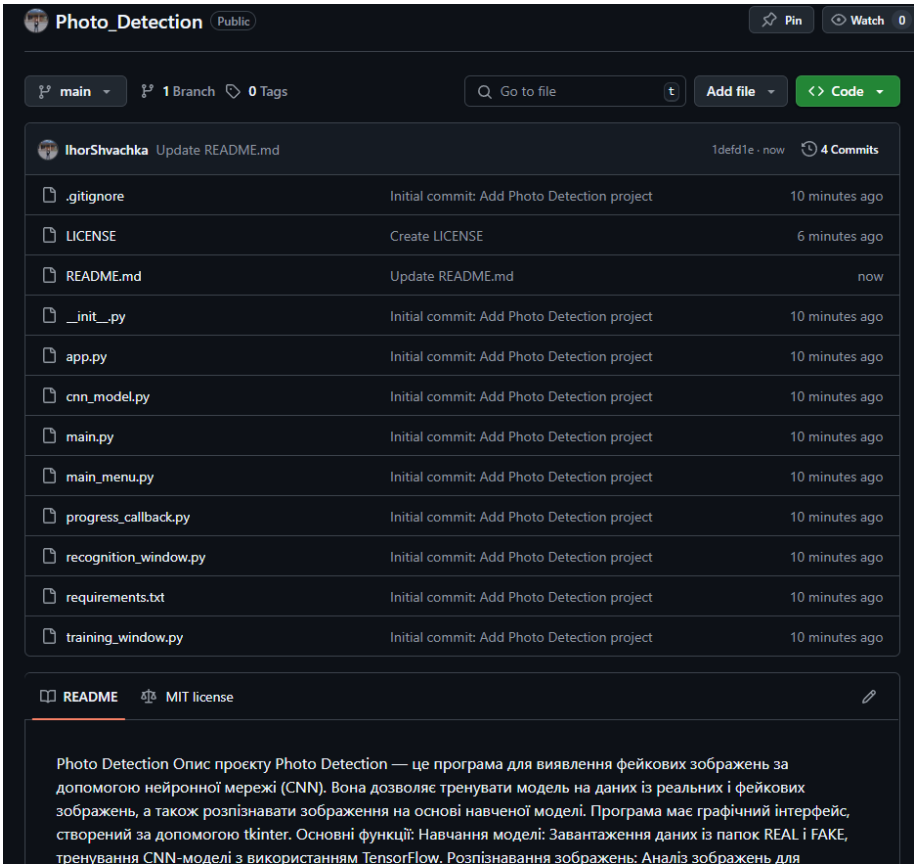
Додаток А

Програмний код

Посилання на репозиторій на GitHub

https://github.com/IhorShvachka/Photo_Detection

Вигляд сторінки репозиторію



Опис вмісту файлу

У створеному репозиторію розміщені:

– main.py - точка входу програми.

– app.py - клас App для управління вікнами.

– cnn_model.py - клас CNNModel для створення та тренування моделі.

– main_menu.py - клас MainMenu для головного меню.

– training_window.py - клас TrainingWindow для вікна тренування.

– recognition_window.py - клас RecognitionWindow для вікна розпізнавання.

– progress_callback.py - клас ProgressCallback для відстеження прогресу тренування.

– README.md – документація проєкту.

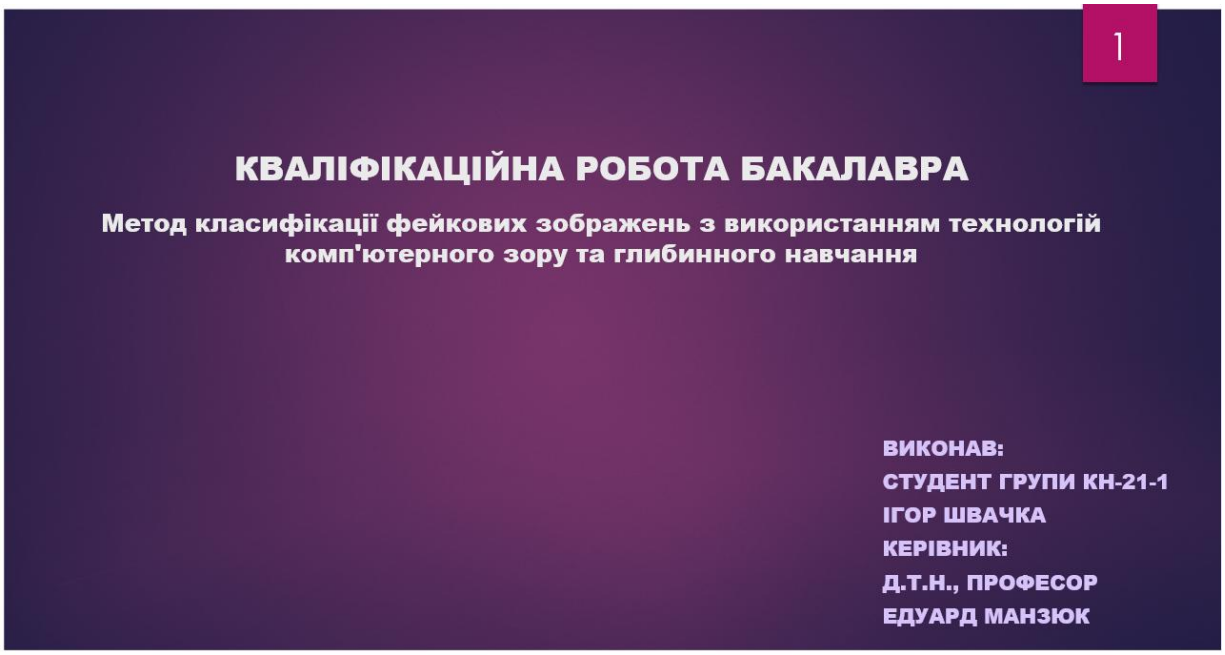
– .gitignore - виключає з репозиторію віртуальне оточення, кеш, IDE-файли та моделі для чистоти проєкту.

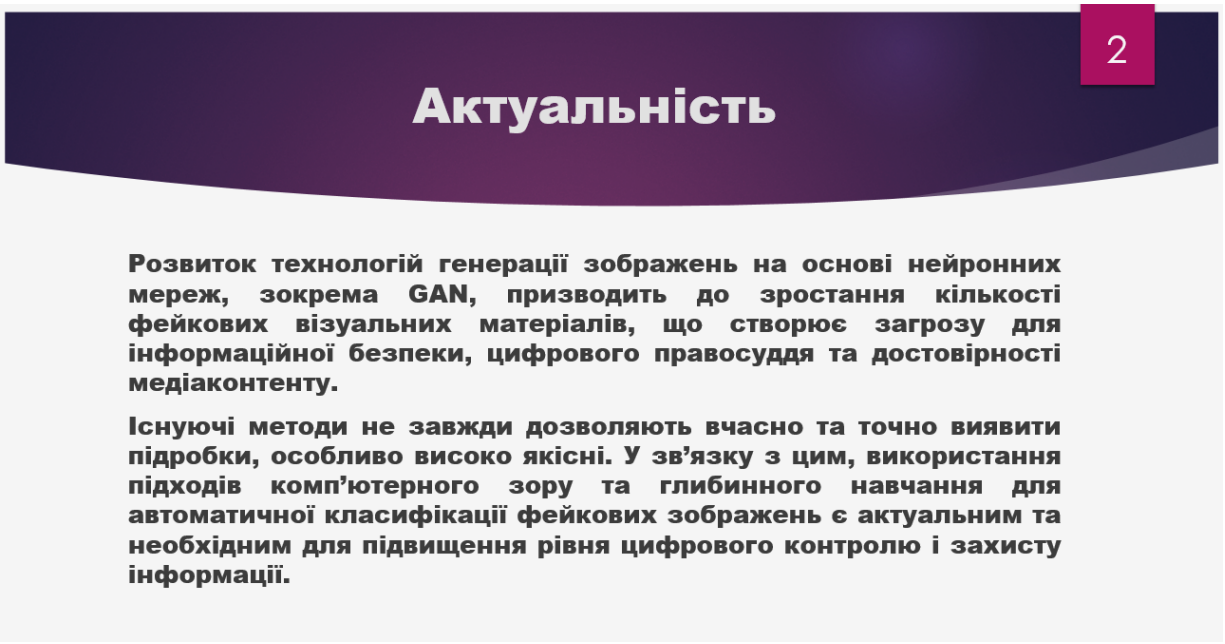
– __init__.py - Позначає Photo_Detection як пакет Python для імпортів.

– requirements.txt - список залежностей.

Додаток В

Презентаційний матеріал





3

Мета та завдання кваліфікаційної роботи

Об'єктом дослідження кваліфікаційної роботи є процес класифікації фейкових зображень з використанням комп'ютерного зору на основі глибинного навчання.

Предмет дослідження – методи генеративного ШІ, засоби та підходи до виявлення фейкових зображень.

Метою кваліфікаційної роботи бакалавра є підвищення ефективності виявлення фейкових зображень на основі аналізу зображень засобами штучного інтелекту.

Для досягнення поставленої мети виконано наступні задачі:

- Проведення аналізу актуальної розробки методу класифікації зображень.
- Проведення аналізу теоретичних та практичних підходів які дозволяють визначити зображення.
- Розроблення методу для визначення фейкових зображень.
- Розроблення інформаційної системи методу визначення згенеровоних зображень.
- Виконання дослідження та оцінення точності системи класифікації на реальних і згенерованих зображеннях.

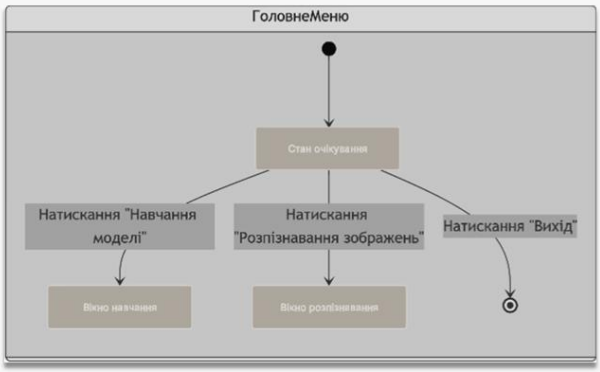
4

Схема методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання

5

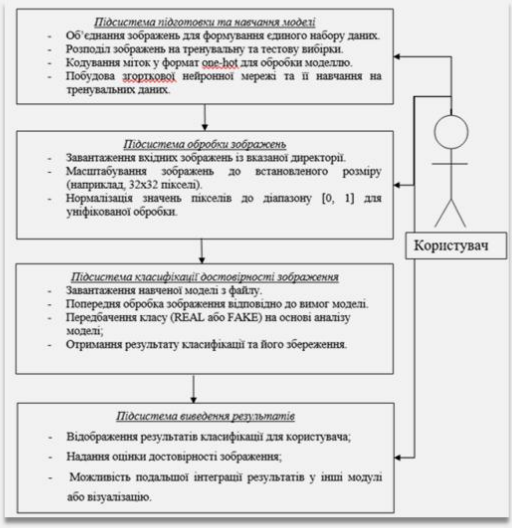
DFD-діаграма системи

Схема навігації між формами системи



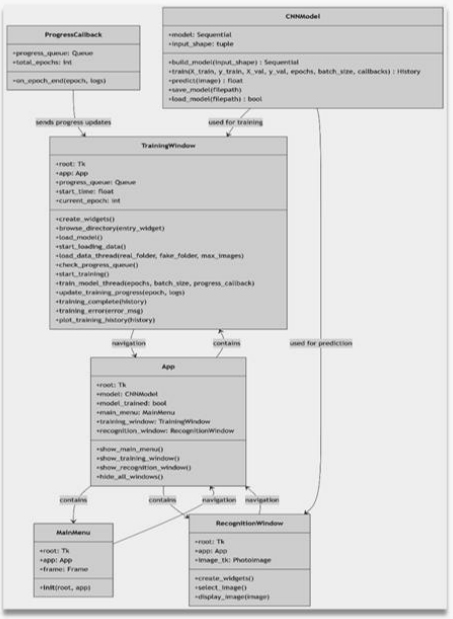
6

Інформаційна схема методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання



7

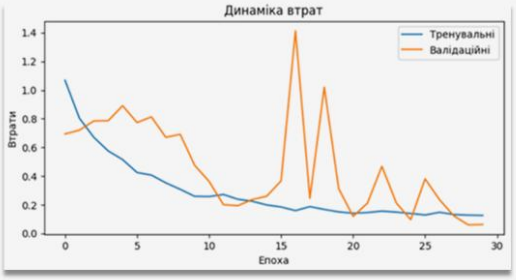
Діаграма класів методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання



8

Діаграма динаміки точності навчання моделі

Діаграма динаміки втрат при навчанні моделі



9

Матриця сплутаності методу класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання

10

ВИСНОВОК

У РЕЗУЛЬТАТІ ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ РОЗРОБЛЕНО МЕТОД КЛАСИФІКАЦІЇ ФЕЙКОВИХ ЗОБРАЖЕНЬ НА ОСНОВІ КОМП’ЮТЕРНОГО ЗОРУ ТА ГЛИБИННОГО НАВЧАННЯ. БУЛО ПРОВЕДЕНО АНАЛІЗ ПІДХОДІВ ДО КЛАСИФІКАЦІЇ, СТВОРЕНО ІНФОРМАЦІЙНУ СИСТЕМУ ДЛЯ ВИЯВЛЕННЯ ЗГЕНЕРОВАНИХ СВІТЛИН, А ТАКОЖ ПРОВЕДЕНО ДОСЛІДЖЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ЗОБРАЖЕННЯХ РІЗНОГО ТИПУ, ЩО ДОЗВОЛЯЄ ОЦІНИТИ ЙОГО ПРАЦЕЗДАТНІСТЬ І ВІДПОВІДНІСТЬ ПОСТАВЛЕНІЙ МЕТІ.

11

Дякую за увагу!

Wed Jun 18 21:58:00 EEST 2025, Петровський Сергій Степанович, Хмельницький національний університет, ХНУ

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 3.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 11%**

	Document		Sum coincidence on the DB	
ID: 246828 Title: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання Added in a DB: 2025-06-18 Authors: Ігор ШВАЧКА Heads: Едуард МАНЗЮК Consultants: Opponents:	Symbols	Lexemes	Symbols	Lexemes
	54189	795	3279 (6%)	50 (6%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Ігор ШВАЧКА

Співавтор:

Назва: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання

Науковий керівник: Едуард МАНЗЮК, д.т.н., доцент

Підрозділ: Кафедра комп'ютерних наук

Коефіцієнт подібності 1:6.8%

Коефіцієнт подібності 2:2.5%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 4

Дата створення звіту: 2025-06-18 20:42:56.0

Після аналізу Звіту подібності констатую наступне:

☒ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обґрунтування:

2025-06-18

Дата експерт Перковський С.С.

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи *Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання*
Автор *студент групи КН-21-1 Ігор ШВАЧКА*
Освітня програма *Комп'ютерні науки*
Рівень вищої освіти *перший (бакалаврський)*
Спеціальність *122 – Комп'ютерні науки*
Науковий керівник: *д.т.н., проф. каф. комп'ютерних наук Едуард МАНЗЮК*

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмними засобами комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	***відповідає***
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	***відсутні***

Підтвердження:

Запозичення, виявлені в роботі Ігора Швачки, не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти, які не мають авторства і містять поширені конструкції та загальновідомі терміни, скорочення. Рівень подібності не перевищує допустимої межі. Таким чином, робота є законною та приймається до захисту.

Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості:
- за системою Anti-Plagiarism: 3%;
- за системою StrikePlagiarism КП1: 6,8%, КП2: 2,5%.

18.06.2025

Завідувач кафедри _____ *Олександр БАРМАК*

Гарант освітньої програми _____ *Олександр МАЗУРЕЦЬ*

Керівник кваліфікаційної роботи _____ *Едуард МАНЗЮК*

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук

РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра

студента *гр. КН-21-1 Швачка Ігор Сергійович*
за темою: *Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання*

1. Актуальність обраної теми

Актуальність теми зумовлена потребою в ефективному виявленні фейкових зображень, що активно створюються за допомогою нейромереж. Глибинне навчання й комп'ютерний зір забезпечують сучасні підходи для вирішення цієї проблеми.

2. Повнота розкриття мети та завдань роботи

Мета роботи чітко сформульована - розробка методу класифікації фейкових зображень. Усі завдання реалізовані: проведено аналіз підходів, реалізовано інформаційну систему, виконано тестування та досліджено ефективність розробленого методу.

3. Зміст кожного розділу роботи

Перший розділ аналізує сучасні підходи до виявлення фейкових зображень, теоретичні основи глибинного навчання та комп'ютерного зору, а також визначає мету, завдання й вимоги до розробки методу. У другому розділі описано запропонований метод, його функціональну структуру, архітектуру нейронної мережі, підготовку вхідних даних. Третій розділ містить опис реалізації програмної системи, структуру модулів, результати тестування, дослідження ефективності моделі та висновки.

4. Оцінка розробленої інформаційної системи, її практична цінність

Інформаційна система демонструє високу точність класифікації фейкових зображень, стабільність роботи та зручний інтерфейс. Практична цінність полягає в автоматизованому аналізі цифрових зображень для виявлення підробок без додаткових ресурсів або зовнішнього втручання.

5. Якість оформлення кваліфікаційної роботи бакалавра

Робота відповідає встановленим вимогам до оформлення, має чітку структуру, включає всі обов'язкові розділи, наукові посилання, таблиці, діаграми та графічні матеріали, що ілюструють результати дослідження.

6. Недоліки кваліфікаційної роботи бакалавра

Незважаючи на високу точність і стабільність, система поки що не адаптована під інші платформи, що обмежує її гнучкість. Також відсутні механізми пояснень розпізнавальних рішень моделі, що ускладнює розуміння результатів користувачами.

7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка відмінно.

Рецензент к.т.н. доц. каф. КІС Нікіпорець Д.О. _____

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук

ВІДГУК НАУКОВОГО КЕРІВНИКА

на кваліфікаційну роботу бакалавра

студента *гр. КН-21-1 Ігоря ШВАЧКИ*

за темою *Метод класифікації фейкових зображень з використанням технологій комп'ютерного зору та глибинного навчання*

1. Актуальність теми

Тема кваліфікаційної роботи є надзвичайно актуальною у сучасних умовах широкого поширення технологій штучного інтелекту та генеративних моделей. З розвитком алгоритмів створення синтетичного контенту виникла гостра потреба у розробці ефективних методів виявлення фейкових зображень. Проблема детекції підроблених візуальних даних має критичне значення для забезпечення інформаційної безпеки,.

2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки

Виконана робота повністю відповідає вимогам стандарту спеціальності 122 "Комп'ютерні науки". Дослідження охоплює розробку інформаційних моделей, алгоритмів машинного навчання та програмних систем для автоматизованої класифікації зображень. У роботі застосовано сучасні методи глибинного навчання, зокрема згорткові нейронні мережі,

3. Професійні та особистісні якості бакалавра

Під час виконання кваліфікаційної роботи студент Швачка Ігор продемонстрував високий рівень професійної підготовки та відповідальне ставлення до навчального процесу. Студент продемонстрував глибокі компетенції у галузі машинного навчання, комп'ютерного зору та розробки програмних систем.

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Усі результати, представлені в роботі, отримані завдяки самостійній роботі студента. Він особисто розробив реалізував алгоритми попередньої обробки зображень, створив та протестував програмну систему класифікації. Студент самостійно провів експериментальні дослідження та аналіз ефективності розробленого методу.

5. Ступінь оволодіння методами дослідження

Студент продемонстрував впевнене володіння сучасними методами дослідження в галузі комп'ютерних наук та машинного навчання. Він ефективно застосував методи глибинного навчання, технології комп'ютерного зору, алгоритми попередньої обробки зображень та аугментації даних.

6. Повнота та якість розкриття теми роботи

Тема роботи розкрита повністю. Проведено аналіз існуючих підходів до виявлення фейкових зображень та сучасних методів комп'ютерного зору. Розроблено функціональну систему для розпізнавання справжніх та згенерованих зображень. Експериментальні дослідження підтвердили ефективність запропонованого методу та його переваги над базовими підходами.

7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу

Робота має логічну структуру та послідовний виклад матеріалу. Усі розділи пов'язані між собою та спрямовані на досягнення поставленої мети. Теоретичні положення підкріплені експериментальними результатами та відповідними графічними матеріалами. Стиль викладу аргументований, літературно грамотний.

8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин

Розроблений метод має потенціал для практичного застосування у сфері інформаційної безпеки та соціальних мереж. Система класифікації може бути використана для автоматичної перевірки достовірності візуального контенту в новинних ресурсах, соціальних платформах та системах контент-модерації.

9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота

*Враховуючи актуальність теми, якість виконання, відповідність усім вимогам до кваліфікаційних робіт бакалавра, робота може бути допущена до захисту. Рекомендована оцінка «**відмінно**» .*

Керівник _____ *д.т.н., проф. каф. КН Едуард МАНЗЮК*