

Я як здобувач вищої освіти ХНУРЕ розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

15.05.2026



Михайло ТАРАСЕНКО

Кваліфікаційна робота не містить відомостей заборонених до відкритого опублікування.

Кваліфікаційна робота виконана у відповідності до стандартів, що діють в Україні.

Попередній захист проведено 15.05.2026 р.

Керівник кваліфікаційної роботи



Зульфія ІМАНГУЛОВА

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____

Кафедра _____ КМІТ _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 122 Комп'ютерні науки _____
(код і повна назва)

Тип програми _____ Освітньо-наукова _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Системне проектування _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

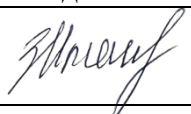
« _____ » _____ 20 ____ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Тарасенку Михайлу Андрійовичу _____
(прізвище, ім'я, по батькові)

- Тема роботи «Дослідження моделей глибокого навчання для побудови інтелектуальних систем діагностики захворювань рослин»
затверджена наказом університету від 06 квітня 2026 р. № 268
- Термін подання здобувачем роботи до екзаменаційної комісії 18 травня 2026 р.
- Вихідні дані до роботи: набори зображень захворювань рослин PlantVillage та Plant Leaf Diseases Training Dataset, моделі глибокого навчання AlexNet і GoogLeNet (InceptionV3), методи комп'ютерного зору та класифікації зображень, бібліотеки TensorFlow та Keras, результати навчання й тестування моделей, а також метрики оцінювання якості класифікації зображень рослин.
- Перелік питань, що потрібно опрацювати в роботі: аналіз предметної області діагностики захворювань рослин; дослідження використання комп'ютерного зору в аграрній сфері; аналіз сучасних моделей глибокого навчання для обробки зображень; дослідження існуючих підходів і реалізованих рішень для класифікації захворювань рослин; формування вимог до системи діагностики; аналіз та підготовка наборів даних для навчання моделей; розробка програмного засобу для тестування моделей; проведення експериментального навчання моделей; оцінювання якості моделей за допомогою метрик класифікації; порівняльний аналіз результатів роботи моделей глибокого навчання
- Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій Приклад хлорозу листя, Приклад некрозу тканини, Приклад плямистості листя, Архітектура AlexNet, Архітектура GoogLeNet, Фрагмент процесу навчання моделей глибокого навчання, Результат навчання моделі AlexNet, Результат навчання GoogleNet, Результат виводу консолі передбачення зображення, Результати оцінювання моделі AlexNet, Результати оцінювання моделі GoogleNet, Узагальнене порівняння основних метрик якості моделей, Порівняння значень precision та recall для окремих класів, Порівняння значень F1-score, Графік приросту значення F1-score між моделями, Порівняння десяти класів із найвищими значеннями F1-score, Порівняння десяти класів із найнижчими значеннями F1-score.

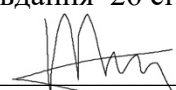
6. Консультанти розділів роботи

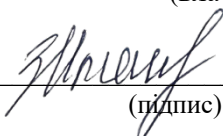
Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Розділи спеціальної частини	доц. Імангулова З.А.		15.05.2026

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Аналіз предметної області та формування теми дослідження	26.01.2026 – 31.01.26	Виконано
2	Аналіз наукових джерел та сучасних підходів комп'ютерного зору	01.02.2026 – 10.02.26	Виконано
3	Дослідження моделей глибокого навчання для класифікації зображень	11.02.2026 – 18.02.26	Виконано
4	Формування структури дослідження та постановки задачі	19.02.2026 – 22.02.26	Виконано
5	Аналіз та підготовка наборів даних для навчання моделей	23.02.2026 – 05.03.26	Виконано
6	Реалізація попередньої обробки зображень	06.03.2026 – 12.03.26	Виконано
7	Реалізація та налаштування моделей AlexNet і GoogLeNet	13.03.2026 – 24.03.26	Виконано
8	Проведення навчання моделей глибокого навчання	25.03.2026 – 05.04.26	Виконано
9	Оцінювання якості моделей та аналіз результатів	06.04.2026 – 15.04.26	Виконано
10	Побудова графіків та порівняльний аналіз моделей	16.04.2026 – 22.04.26	Виконано
11	Розробка консольного програмного засобу тестування моделей	23.04.2026 – 02.05.26	Виконано
12	Формування висновків та остаточне оформлення роботи	03.05.2026 – 10.05.26	Виконано

Дата видачі завдання 26 січня 2026 р.

Здобувач  Михайло ТАРАСЕНКО
(підпис) (власне ім'я, прізвище)

Керівник роботи  доц. Зульфія ІМАНГУЛОВА
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи містить: 104 с., 22 рисунки, 27 джерел інформації, 2 додатки.

ГЛИБОКЕ НАВЧАННЯ, ДІАГНОСТИКА ЗАХВОРЮВАНЬ РОСЛИН, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, КОМП'ЮТЕРНИЙ ЗІР, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, МАШИННЕ НАВЧАННЯ, ПОПЕРЕДНЯ ОБРОБКА ЗОБРАЖЕНЬ, ACCURACY, CONVOLUTIONAL NEURAL NETWORKS, DEEP LEARNING, IMAGE CLASSIFICATION, PRECISION, RECALL, F1-SCORE

Об'єктом дослідження є процес автоматизованого виявлення та класифікації візуальних ознак захворювань рослин за допомогою інтелектуального аналізу даних.

Предмет дослідження - моделі глибокого навчання, архітектури згорткових нейронних мереж (CNN) та методи сегментації зображень, що забезпечують високу точність ідентифікації специфічних уражень рослин.

Метою роботи є дослідження та порівняльний аналіз моделей глибокого навчання для задачі діагностики захворювань рослин.

Методи дослідження – методи комп'ютерного зору та глибокого навчання для класифікації зображень захворювань рослин. Під час дослідження застосовано методи попередньої обробки зображень, згорткові нейронні мережі AlexNet та GoogLeNet (InceptionV3), підхід transfer learning, а також методи оцінювання якості моделей за метриками accuracy, precision, recall та F1-score.

Галузь застосування - результати роботи можуть бути використані у сфері аграрних інформаційних систем, систем моніторингу стану рослин, інтелектуальних систем підтримки прийняття рішень у сільському господарстві.

ABSTRACT

Explanatory note of the qualification work: 104 pages, 22 fig., 27 sources of information, 2 appendices.

ACCURACY, AGRICULTURAL SECTOR, COMPUTER VISION, CONVOLUTIONAL NEURAL NETWORKS, DEEP LEARNING, F1-SCORE, IMAGE CLASSIFICATION, MACHINE LEARNING, PLANT DISEASE DIAGNOSIS, PRECISION, PREPROCESSING OF IMAGES, RECALL

The object of research is the process of automated detection and classification of visual symptoms of plant diseases using intelligent data analysis.

The subject of research includes deep learning models, convolutional neural network (CNN) architectures, and image segmentation methods that ensure high accuracy in identifying specific plant lesions.

The aim of the work is to research and perform a comparative analysis of deep learning models for the task of plant disease diagnostics.

Research methods involve computer vision and deep learning methods for image classification of plant diseases. During the research, image preprocessing methods, AlexNet and GoogLeNet (InceptionV3) convolutional neural networks, and the transfer learning approach were applied, along with model quality evaluation methods based on accuracy, precision, recall, and F1-score metrics.

Field of application: the results of the work can be implemented in the field of agricultural information systems, plant health monitoring systems, and intelligent decision-support systems in agriculture.

ЗМІСТ

ВСТУП.....	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
1.1 Актуальність проблеми автоматизованої діагностики захворювань рослин	11
1.2 Характеристика основних захворювань рослин та їх візуальних ознак .	17
1.3 Методи комп'ютерного зору в аграрній сфері	23
1.4 Огляд моделей глибокого навчання для обробки зображень.....	26
1.5 Аналіз існуючих підходів і рішень	30
1.6 Постановка задачі дослідження	33
2 АНАЛІЗ І ВИБІР МЕТОДІВ ТА МОДЕЛЕЙ ДЛЯ ДОСЛІДЖЕННЯ	35
2.1 Формування вимог до системи діагностики	35
2.2 Опис і аналіз наборів даних для навчання моделей	40
2.3 Попередня обробка зображень	47
2.4 Вибір архітектур моделей глибокого навчання	51
2.5 Метрики оцінювання якості моделей	53
3 РЕАЛІЗАЦІЯ МОДЕЛЕЙ ТА ПРОГРАМНОГО ЗАСОБУ ДОСЛІДЖЕННЯ...	57
3.1 Обґрунтування вибору інструментів та технологій	57
3.2 Реалізація обраних моделей глибокого навчання	59
3.3 Налаштування процесу навчання моделей	64
3.4 Розробка консольного програмного засобу для тестування моделей	70
4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ МОДЕЛЕЙ.....	76
4.1 Постановка експерименту.....	76
4.2 Проведення навчання моделей.....	78
4.3 Оцінювання якості моделей за обраними метриками.....	84

4.4 Порівняльний аналіз результатів	87
ВИСНОВКИ	100
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	103
Додаток А	Ошибка! Закладка не определена.
Додаток Б.....	Ошибка! Закладка не определена.

ВСТУП

У сучасних умовах розвитку інформаційних технологій особливого значення набуває застосування методів штучного інтелекту для розв'язання прикладних задач у різних галузях, зокрема в аграрній сфері. Однією з актуальних проблем є своєчасне та точне діагностування захворювань рослин, від якого безпосередньо залежить якість і обсяг урожаю. Традиційні підходи до виявлення захворювань ґрунтуються на візуальному аналізі, що потребує значного досвіду та не завжди забезпечує достатню точність.

Розвиток методів комп'ютерного зору та глибокого навчання відкриває нові можливості для автоматизації процесу діагностики. Використання моделей, здатних аналізувати зображення рослин і визначати характерні ознаки захворювань, дозволяє підвищити об'єктивність та швидкість прийняття рішень.

Актуальність теми дослідження обумовлена необхідністю підвищення точності та надійності автоматизованих систем діагностики, а також потребою у порівняльному аналізі різних моделей глибокого навчання для визначення найбільш ефективних підходів.

Об'єктом дослідження є процес автоматизованого виявлення та класифікації візуальних ознак захворювань рослин за допомогою інтелектуального аналізу даних.

Предметом дослідження є моделі глибокого навчання, архітектури згорткових нейронних мереж (CNN) та методи сегментації зображень, що забезпечують високу точність ідентифікації специфічних уражень рослин.

Метою роботи є дослідження та порівняльний аналіз моделей глибокого навчання для задачі діагностики захворювань рослин, а також визначення факторів, що впливають на ефективність їх використання.

Для досягнення поставленої мети необхідно вирішити такі задачі: провести аналіз предметної області; дослідити існуючі підходи та моделі глибокого навчання; обрати та підготувати набір даних; реалізувати попередню обробку зображень; обрати архітектури моделей; визначити метрики оцінювання; провести експериментальне дослідження та проаналізувати отримані результати.

Практичне значення роботи полягає у формуванні підходів до побудови систем діагностики захворювань рослин на основі глибокого навчання, що можуть бути використані для подальших досліджень та розробки прикладних рішень у сфері комп'ютерного зору.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Актуальність проблеми автоматизованої діагностики захворювань рослин

Захворювання рослин визначається як патологічний стан рослинного організму, спричинений впливом біотичних або абіотичних факторів, який призводить до порушення нормальних фізіологічних процесів, зниження продуктивності або загибелі рослини. До біотичних чинників належать грибкові, бактеріальні та вірусні інфекції, а також вплив шкідників, тоді як абіотичні фактори включають несприятливі умови середовища, такі як дефіцит або надлишок вологи, температурні стреси, нестача поживних речовин або забруднення ґрунту. Захворювання можуть проявлятися у вигляді плям на листках, змін кольору, деформацій, некрозів або інших морфологічних відхилень, що ускладнює їх однозначну ідентифікацію.

Проблема діагностування захворювань рослин полягає в багатофакторності їх виникнення та складності візуальної ідентифікації. Багато різних захворювань мають подібні симптоми, що ускладнює їх розрізнення навіть для досвідчених агрономів. Наприклад, пожовтіння листя може бути наслідком як інфекційного ураження, так і нестачі азоту в ґрунті. Крім того, симптоми одного й того ж захворювання можуть варіюватися залежно від стадії розвитку рослини, кліматичних умов та регіональних особливостей, що ще більше підвищує складність діагностики.

Додатковою проблемою є необхідність залучення кваліфікованих спеціалістів для проведення точної діагностики. У багатьох випадках фермери або власники господарств не мають достатніх знань для самостійного визначення типу захворювання, що призводить до затримок у прийнятті рішень щодо лікування. Це, у свою чергу, може спричинити значні економічні втрати через зниження врожайності або повну втрату посівів. У польових умовах доступ до експертів часто обмежений, особливо в віддалених регіонах, що підсилює актуальність автоматизованих підходів.

Складність діагностики також зумовлена динамічністю розвитку захворювань. На ранніх стадіях симптоми можуть бути слабо вираженими або взагалі непомітними для людського ока, що ускладнює своєчасне виявлення проблеми. Водночас ефективність боротьби із захворюванням значною мірою залежить саме від ранньої діагностики, коли ще можливо мінімізувати поширення інфекції. Пізнє виявлення захворювання часто призводить до необхідності використання більш агресивних методів лікування, що може негативно впливати на якість продукції та навколишнє середовище.

Таким чином, традиційні методи діагностики, які базуються переважно на візуальному огляді та експертному досвіді, мають обмеження щодо точності, швидкості та масштабованості. Це зумовлює необхідність впровадження автоматизованих систем діагностики, здатних забезпечити об'єктивний, швидкий та доступний аналіз стану рослин. Використання методів глибокого навчання відкриває можливість створення інтелектуальних систем, які можуть ефективно розпізнавати захворювання за зображеннями, зменшуючи залежність від людського фактора та підвищуючи загальну ефективність аграрного виробництва.

Глибоке навчання є підрозділом машинного навчання, який базується на використанні багатошарових штучних нейронних мереж для автоматичного виділення складних закономірностей у даних. Основною особливістю даного підходу є здатність моделі самостійно навчатися ієрархічним представленням ознак без необхідності ручного конструювання характеристик, що було характерно для класичних методів аналізу даних. Кожен шар нейронної мережі виконує перетворення вхідних даних, поступово переходячи від простих ознак до більш абстрактних, що дозволяє ефективно працювати з високовимірними структурами, зокрема зображеннями.

Функціонування моделей глибокого навчання ґрунтується на концепції штучного нейрона, який обчислює зважену суму вхідних сигналів з подальшим застосуванням нелінійної функції активації. Нейрони об'єднуються у шари, формуючи глибоку архітектуру, де інформація послідовно передається від вхідного шару до вихідного. Процес навчання полягає у мінімізації функції втрат

шляхом коригування вагових коефіцієнтів мережі за допомогою алгоритму зворотного поширення помилки, що дозволяє моделі поступово покращувати точність прогнозів. Для задач аналізу зображень найбільш поширеним різновидом є згорткові нейронні мережі, які ефективно виявляють просторові залежності та локальні структури у вхідних даних.

У контексті діагностики захворювань рослин глибоке навчання демонструє високу ефективність завдяки здатності автоматично аналізувати візуальні характеристики зображень листя, стебел або плодів. Моделі здатні розпізнавати складні патерни, такі як форма плям, їх текстура, колірні зміни або поширення ураження, які можуть бути важко помітними або інтерпретованими людиною. На відміну від традиційних підходів, де необхідно вручну визначати набір ознак, глибоке навчання дозволяє будувати універсальні моделі, що адаптуються до різних типів захворювань і культур.

Практичне застосування таких моделей передбачає навчання на великій кількості розмічених зображень, де кожному прикладу відповідає певний клас захворювання або стан рослини. Після навчання модель здатна виконувати класифікацію нових зображень, визначаючи ймовірність належності до кожного з можливих класів. Це дозволяє автоматизувати процес діагностики, зменшити залежність від людського фактора та значно пришвидшити обробку даних. Крім того, використання глибокого навчання відкриває можливості для раннього виявлення захворювань, що є критично важливим для запобігання їх поширенню.

Застосування глибокого навчання у даній задачі також сприяє підвищенню точності діагностики за рахунок використання великих обсягів даних та складних моделей, здатних враховувати численні фактори впливу. Це особливо актуально в умовах варіативності зовнішнього середовища, де однакові захворювання можуть проявлятися по-різному. Таким чином, глибоке навчання виступає ключовим інструментом для створення інтелектуальних систем діагностики, що поєднують високу точність, швидкість та масштабованість обробки інформації.

Актуальність автоматизованої діагностики захворювань рослин зумовлена зростанням вимог до ефективності сільськогосподарського виробництва та необхідністю забезпечення стабільної якості продукції. У сучасних умовах аграрна галузь функціонує під впливом численних ризиків, серед яких важливе місце займають інфекційні та неінфекційні захворювання рослин, що можуть призводити до значних втрат врожаю. З огляду на це своєчасне виявлення патологій стає критично важливим фактором підтримання продуктивності та економічної доцільності вирощування культур.

Однією з ключових проблем є обмеженість традиційних методів діагностики, які базуються на візуальному огляді та експертному досвіді. Такий підхід характеризується суб'єктивністю, залежністю від рівня кваліфікації спеціаліста та значними витратами часу. У масштабах великих аграрних господарств або при обробці значної кількості рослин ручна діагностика стає малоефективною та практично нездійсненною без суттєвих ресурсних витрат.

Додатковим фактором актуальності є необхідність оперативного прийняття рішень. Захворювання рослин мають властивість швидко поширюватися, особливо за сприятливих кліматичних умов, що вимагає максимально швидкої реакції для локалізації ураження. Автоматизовані системи здатні забезпечити миттєвий аналіз зображень і формування висновків щодо стану рослин, що дозволяє значно скоротити час між виявленням проблеми та застосуванням відповідних заходів.

Важливим аспектом є також доступність сучасних технологій обробки зображень та глибокого навчання, які дозволяють створювати ефективні інструменти діагностики навіть без залучення високоспеціалізованих знань з боку кінцевого користувача. Це відкриває можливості для широкого впровадження таких систем у практичну діяльність фермерів, агрономів та інших учасників аграрного сектору, включаючи малі господарства, які раніше не мали доступу до якісної експертної оцінки.

Крім того, автоматизована діагностика сприяє підвищенню об'єктивності результатів, оскільки алгоритмічний підхід дозволяє уникнути впливу людського фактора та забезпечити стабільність оцінювання. Використання

великих масивів даних під час навчання моделей дає змогу враховувати різноманіття проявів захворювань, що підвищує точність розпізнавання навіть у складних або нетипових випадках.

Правове поле розробки програмного засобу для автоматизованої діагностики захворювань рослин визначається сукупністю нормативно-правових актів, що регулюють обробку даних, використання інформаційних технологій, захист інтелектуальної власності, а також діяльність у сфері агропромислового комплексу. У процесі проєктування та реалізації такого застосунку необхідно враховувати вимоги національного законодавства України, що забезпечують правомірність збору, обробки та використання інформації, а також регламентують взаємодію із користувачами та іншими суб'єктами.

Ключовим нормативним актом є Закон України «Про захист персональних даних», який визначає порядок обробки персональних даних користувачів. У контексті розроблюваного застосунку це стосується випадків, коли система може зберігати інформацію про користувачів, їхні облікові записи, історію запитів або геолокаційні дані. Закон встановлює вимоги щодо отримання згоди на обробку даних, забезпечення їх захисту та обмеження доступу до них, що безпосередньо впливає на архітектуру системи, механізми автентифікації та зберігання інформації.

Важливу роль також відіграє Закон України «Про інформацію», який регламентує загальні принципи інформаційних відносин, включаючи право на інформацію, її захист та використання. У рамках даного проєкту цей закон визначає правила роботи з даними, що обробляються системою, а також встановлює обмеження щодо поширення інформації, зокрема у випадках, коли результати діагностики можуть використовуватись для прийняття управлінських рішень.

Наступним важливим нормативним актом є Закон України «Про захист інформації в інформаційно-телекомунікаційних системах», який визначає вимоги до забезпечення інформаційної безпеки. Його положення впливають на проєктування системи з точки зору захисту даних від несанкціонованого

доступу, використання криптографічних засобів, а також організації безпечного зберігання та передачі інформації.

У контексті програмної розробки необхідно також враховувати Закон України «Про авторське право і суміжні права», який регулює питання захисту програмного коду, моделей машинного навчання та використаних наборів даних. Цей закон визначає права розробників на результати їхньої діяльності, а також обмеження щодо використання сторонніх ресурсів, включаючи відкриті датасети та програмні бібліотеки.

Оскільки розроблювана система застосовується у сфері сільського господарства, доцільно враховувати положення Закон України «Про карантин рослин», який регламентує заходи щодо виявлення, локалізації та ліквідації захворювань рослин. Хоча програмний засіб не є безпосереднім інструментом державного контролю, результати його роботи можуть використовуватися для прийняття рішень у межах цього правового поля, що обумовлює необхідність забезпечення достатнього рівня достовірності та відповідності результатів.

З урахуванням наведених нормативних актів формується комплексне правове середовище, яке впливає на всі етапи життєвого циклу програмного засобу – від збору та обробки даних до розповсюдження та використання результатів діагностики. Це вимагає врахування юридичних аспектів ще на етапі проєктування системи з метою забезпечення її відповідності чинному законодавству.

Проведений аналіз предметної області показує, що діагностика захворювань рослин є складною багатофакторною задачею, яка характеризується значною варіативністю симптомів, залежністю від зовнішніх умов та обмеженістю традиційних підходів, що базуються на експертній оцінці. Встановлено, що існуючі методи діагностики не забезпечують достатньої швидкості, масштабованості та об'єктивності, що ускладнює своєчасне виявлення патологій та прийняття ефективних рішень.

Обґрунтовано доцільність використання методів глибокого навчання як інструменту автоматизації діагностики, здатного забезпечити високий рівень точності розпізнавання захворювань на основі аналізу зображень. Визначено, що

впровадження таких підходів відповідає сучасним вимогам аграрної галузі та дозволяє підвищити ефективність контролю стану рослин. Одночасно встановлено необхідність врахування правових аспектів, пов'язаних із обробкою даних та використанням програмних рішень, що формує комплексне підґрунтя для подальшого дослідження та реалізації системи.

1.2 Характеристика основних захворювань рослин та їх візуальних ознак

Захворювання рослин класифікуються за природою їх походження на біотичні та абіотичні. До біотичних належать інфекційні захворювання, спричинені живими організмами, зокрема грибами, бактеріями, вірусами та фітоплазмами. Найбільш поширеними серед них є грибкові інфекції, які проявляються у вигляді плямистостей, нальоту, гнилі або некрозів. Бактеріальні захворювання часто супроводжуються водянистими ураженнями, потемнінням тканин та виділенням ексудату. Вірусні інфекції характеризуються мозаїчними візерунками, деформаціями листя та пригніченням росту рослин. Біотичні захворювання мають здатність поширюватися між рослинами, що підвищує їхню небезпечність у межах агроєкосистем.

Абіотичні захворювання не пов'язані з дією живих патогенів і виникають унаслідок несприятливих умов середовища. До таких факторів належать дефіцит або надлишок поживних речовин, порушення водного режиму, температурні стреси, засолення ґрунтів, токсичний вплив хімічних речовин або забруднення. Симптоми абіотичних ушкоджень можуть включати пожовтіння листя, опіки, затримку росту, відмирання тканин або деформації. Важливою особливістю є те, що такі прояви часто не мають чіткої локалізації та можуть охоплювати значну частину рослини або посіву.

Візуальна діагностика захворювань рослин базується на аналізі морфологічних ознак ураження, однак її ефективність є обмеженою навіть для досвідчених фахівців. Значна кількість захворювань має подібні симптоми, що ускладнює їх однозначне розрізнення. Наприклад, пожовтіння листя може бути спричинене як вірусною інфекцією, так і дефіцитом азоту або іншими

стресовими факторами. Подібним чином, плямистості різного походження можуть мати схожий вигляд, але вимагати принципово різних підходів до лікування.

Додаткову складність створює те, що прояви одного й того ж захворювання можуть змінюватися залежно від стадії розвитку рослини, кліматичних умов та інтенсивності ураження. На ранніх етапах симптоми часто є малопомітними або неспецифічними, що ускладнює їх ідентифікацію. У деяких випадках точна діагностика потребує лабораторних досліджень, таких як мікроскопічний аналіз або молекулярні методи, які дозволяють визначити збудника з високою точністю.

Таким чином, застосування методів глибокого навчання для діагностики захворювань рослин не забезпечує гарантовано точного результату у кожному окремому випадку, що обумовлено складністю предметної області, варіативністю проявів захворювань та залежністю моделей від якості і повноти навчальних даних. Моделі можуть допускати помилки у випадках нетипових симптомів, недостатньої представленості окремих класів у навчальній вибірці або впливу зовнішніх факторів, таких як освітлення, фон зображення чи якість зйомки.

Водночас використання глибокого навчання суттєво підвищує ймовірність правильної діагностики порівняно з традиційними підходами, особливо в умовах обмеженого доступу до експертних знань. Завдяки здатності аналізувати великі обсяги даних і виявляти складні закономірності у зображеннях, такі моделі можуть ефективно розпізнавати характерні ознаки захворювань навіть за наявності незначних або слабо виражених симптомів. Це дозволяє використовувати їх як інструмент підтримки прийняття рішень, що підвищує загальну точність і оперативність діагностування.

Отже, глибоке навчання доцільно розглядати не як абсолютну заміну експертній оцінці, а як допоміжний інтелектуальний інструмент, який у поєднанні з професійним досвідом здатний значно зменшити ризик помилок та підвищити ефективність виявлення захворювань рослин.

Основні ознаки захворювань рослин визначаються через візуальні зміни зовнішнього вигляду їхніх органів, що є відображенням порушення внутрішніх фізіологічних процесів. До найбільш характерних симптомів належать зміни кольору листя, зокрема хлороз, який проявляється у вигляді пожовтіння тканин, а також некроз – відмирання окремих ділянок рослини. Подібні ознаки можуть свідчити як про інфекційні ураження, так і про дефіцит мікро- або макроелементів, що ускладнює однозначну інтерпретацію симптомів без додаткового аналізу [1].

Важливим класом ознак є плямистості, які можуть мати різну форму, колір та структуру. Вони часто виникають унаслідок грибкових або бактеріальних інфекцій і характеризуються наявністю чітко окреслених зон ураження. Крім того, для багатьох грибкових захворювань типовим є утворення нальоту або спор, що проявляються у вигляді білого, сірого або темного покриву на поверхні листя. Такі візуальні патерни є особливо важливими для задач комп'ютерного зору, оскільки вони можуть бути формалізовані у вигляді ознак для подальшого аналізу моделями глибокого навчання.

Ще однією групою симптомів є деформації органів рослини, включаючи скручування листя, зміну їхньої форми або аномальний розвиток. Такі ознаки часто пов'язані з вірусними інфекціями або комплексним впливом кількох факторів. Окрім цього, спостерігаються зміни темпів росту, зокрема карликовість або нерівномірний розвиток рослин, що також може бути індикатором захворювання або стресових умов середовища. Важливо зазначити, що подібні ознаки не завжди є специфічними і можуть проявлятися при різних типах уражень [2].

Процес визначення захворювань базується на аналізі сукупності ознак, а не окремого симптому. На практиці це передбачає оцінку кольору, текстури, форми та розташування уражень, а також врахування контекстних факторів, таких як умови вирощування або стадія розвитку рослини. У задачах комп'ютерного зору ці характеристики трансформуються у числові представлення, що дозволяє алгоритмам автоматично виділяти закономірності у зображеннях та класифікувати їх відповідно до певних класів.

Таким чином, ознаки захворювань рослин є багатовимірними та часто перекриваються між різними типами патологій, що ускладнює їх інтерпретацію. Саме ця особливість робить задачу діагностики складною та обґрунтовує доцільність використання методів глибокого навчання, які здатні враховувати комплексні залежності між візуальними характеристиками та типами захворювань.

На рисунку 1.1 наведено приклад хлорозу листя, який проявляється у вигляді поступового пожовтіння тканин при збереженні зеленого кольору жилок. Така ознака часто свідчить про порушення процесів фотосинтезу та може бути спричинена як інфекційними захворюваннями, так і дефіцитом поживних речовин, зокрема азоту або заліза. Візуально хлороз характеризується рівномірним або зональним знебарвленням, що є одним із найбільш поширених симптомів ураження рослин і використовується як базова ознака для подальшої діагностики.



Рисунок 1.1 – Приклад хлорозу листя

На рисунку 1.2 наведено приклад некрозу тканин, що проявляється у вигляді темних або бурих ділянок відмерлої тканини на поверхні листя. Некротичні ураження можуть мати різну форму – від дрібних точкових плям до

великих зон відмирання. Такий симптом часто є наслідком грибкових або бактеріальних інфекцій, а також може виникати під впливом стресових факторів, таких як посуха або хімічні ушкодження. Важливою характеристикою некрозу є чітка межа між здоровою та ураженою тканиною, що дозволяє використовувати його як інформативну ознаку при автоматизованому аналізі зображень.



Рисунок 1.2 – Приклад некрозу тканини

На рисунку 1.3 наведено приклад плямистості листя, яка проявляється у вигляді множинних округлих або неправильної форми плям різного кольору. Плями можуть мати світлу або темну центральну частину та облямівку, що є характерною ознакою багатьох грибкових інфекцій. Такий тип ураження є особливо важливим для задач комп'ютерного зору, оскільки форма, текстура та

колір плям можуть бути формалізовані як набір ознак для класифікації. Плямистість часто використовується як ключовий індикатор при автоматизованій діагностиці захворювань рослин [5].



Рисунок 1.3 – Приклад плямистості листя

Проведений аналіз візуальних ознак захворювань рослин показує, що основні типи симптомів, такі як зміни кольору, некротичні ураження, плямистості та деформації органів, є важливими індикаторами стану рослини, проте вони не мають однозначної інтерпретації. Одна й та сама ознака може відповідати різним типам захворювань або навіть абіотичним порушенням, що суттєво ускладнює процес діагностики виключно на основі візуального аналізу.

Встановлено, що ефективно визначення типу захворювання потребує врахування сукупності ознак, їх просторового розташування, форми, текстури та кольорових характеристик. Саме комплексний підхід дозволяє підвищити достовірність діагностики, однак навіть у цьому випадку залишається певна невизначеність, зумовлена варіативністю проявів захворювань та впливом зовнішніх факторів.

Таким чином, візуальні ознаки виступають основою для первинного аналізу стану рослин і є ключовим джерелом даних для автоматизованих систем діагностики. Водночас їхня неоднозначність обумовлює необхідність використання інтелектуальних методів обробки, зокрема моделей глибокого навчання, які здатні враховувати складні залежності між ознаками та типами захворювань і тим самим підвищувати точність класифікації [6].

1.3 Методи комп'ютерного зору в аграрній сфері

Аграрна сфера є однією з ключових галузей економіки, що охоплює діяльність, пов'язану з вирощуванням сільськогосподарських культур, утриманням тварин, обробкою та зберіганням продукції, а також забезпеченням продовольчої безпеки. У сучасних умовах аграрний сектор характеризується активним впровадженням цифрових технологій, спрямованих на підвищення ефективності виробництва, оптимізацію використання ресурсів та зниження ризиків, пов'язаних із впливом зовнішніх факторів. Особливу роль у цьому процесі відіграють інтелектуальні системи аналізу даних, які дозволяють автоматизувати прийняття рішень на основі об'єктивної інформації про стан посівів та навколишнього середовища [7].

Комп'ютерний зір є підрозділом штучного інтелекту, що займається розробкою методів і алгоритмів для автоматичного аналізу та інтерпретації візуальної інформації, зокрема цифрових зображень і відео. Основною метою комп'ютерного зору є відтворення здатності людини сприймати та розпізнавати об'єкти, їх властивості та взаємозв'язки у візуальному середовищі. Для цього використовуються різноманітні підходи, включаючи обробку зображень, виділення ознак, класифікацію та сегментацію, що дозволяє перетворювати неструктуровані візуальні дані у формалізовану інформацію, придатну для подальшого аналізу [8].

У контексті аграрної сфери комп'ютерний зір застосовується для вирішення широкого спектра задач, серед яких моніторинг стану рослин, виявлення захворювань, оцінка врожайності, виявлення бур'янів та контроль якості продукції. Використання цифрових зображень, отриманих за допомогою камер, дронів або супутників, дозволяє здійснювати аналіз великих площ сільськогосподарських угідь у реальному часі. Це значно розширює можливості традиційного агрономічного контролю та забезпечує більш точне та оперативне виявлення проблем.

Таким чином, поєднання аграрної сфери та методів комп'ютерного зору формує основу для створення сучасних інтелектуальних систем, здатних

автоматизувати процеси спостереження, аналізу та діагностики. Це відкриває нові можливості для підвищення ефективності сільськогосподарського виробництва та забезпечення стабільної якості продукції.

Застосування комп'ютерного зору в аграрній сфері для задач діагностування захворювань рослин базується на можливості автоматизованого аналізу візуальних даних, отриманих із різних джерел, таких як цифрові камери, безпілотні літальні апарати або стаціонарні сенсорні системи. Основною ідеєю є перетворення зображень рослин у формалізовані дані, які можуть бути використані для подальшої обробки алгоритмами машинного та глибокого навчання. Це дозволяє перейти від суб'єктивного візуального огляду до об'єктивного аналізу, що виконується за допомогою математичних моделей.

На початковому етапі застосування комп'ютерного зору здійснюється попередня обробка зображень, яка включає нормалізацію освітлення, видалення шумів, зміну масштабу та вирівнювання кольорових характеристик. Ці операції є критично важливими, оскільки якість вхідних даних безпосередньо впливає на точність подальшого аналізу. У задачах діагностики захворювань рослин особливу увагу приділяють збереженню кольорових та текстурних характеристик, які є ключовими ознаками для розпізнавання патологій.

Наступним етапом є виділення інформативних ознак із зображення. У класичних підходах комп'ютерного зору використовувалися ручні дескриптори, такі як гістограми кольорів, текстурні характеристики або контурні ознаки. Проте з розвитком глибокого навчання ці процеси автоматизуються: згорткові нейронні мережі самостійно навчаються виділяти релевантні ознаки без явного втручання розробника. Це особливо важливо у випадку захворювань рослин, де ознаки можуть бути складними, багатовимірними та важко формалізованими.

Одним із ключових напрямів використання комп'ютерного зору є сегментація зображень, тобто виділення області рослини або окремих уражених ділянок. Це дозволяє зосередити аналіз на релевантних частинах зображення, зменшуючи вплив фону та сторонніх об'єктів. Наприклад, виділення листя із загального зображення сцени дає змогу більш точно аналізувати його структуру

та виявляти патологічні зміни. У поєднанні з методами глибокого навчання сегментація значно підвищує якість класифікації.

Подальший етап передбачає класифікацію або детекцію захворювань на основі оброблених даних. У цьому випадку комп'ютерний зір виступає як інструмент підготовки та подання інформації для моделей глибокого навчання, які виконують основну задачу розпізнавання. Важливою особливістю є здатність таких систем працювати з великою кількістю зображень, що дозволяє масштабувати процес діагностики на рівень великих аграрних господарств.

Крім того, комп'ютерний зір забезпечує можливість аналізу даних у динаміці, що є важливим для відстеження розвитку захворювань у часі. Використання послідовностей зображень дозволяє виявляти зміни стану рослин, прогнозувати поширення хвороб та оцінювати ефективність застосованих заходів. Це створює передумови для побудови комплексних систем моніторингу, які поєднують діагностику та прогнозування.

Комп'ютерний зір є фундаментальною складовою сучасних систем діагностики захворювань рослин, забезпечуючи обробку, аналіз та структурування візуальних даних. Його інтеграція з методами глибокого навчання дозволяє створювати ефективні інструменти автоматизованого розпізнавання, які здатні враховувати складні взаємозв'язки між ознаками та забезпечувати високу точність діагностики [9].

Використання комп'ютерного зору в аграрній сфері забезпечує можливість автоматизованого аналізу стану рослин на основі візуальних даних, що дозволяє підвищити швидкість, об'єктивність та масштабованість процесів моніторингу. Встановлено, що поєднання методів обробки зображень із моделями глибокого навчання створює ефективний інструмент для виявлення захворювань, аналізу їх розвитку та підтримки прийняття рішень. Це формує основу для впровадження інтелектуальних систем у сільському господарстві, спрямованих на підвищення продуктивності та зниження втрат урожаю.

1.4 Огляд моделей глибокого навчання для обробки зображень

Моделі глибокого навчання для обробки зображень представлені різними архітектурами нейронних мереж, які відрізняються структурою, принципами обробки даних та областями застосування. Найбільш поширеним класом є згорткові нейронні мережі, які спеціалізуються на аналізі просторової структури зображень. Вони використовують операції згортки для виділення локальних ознак, таких як краї, текстури та форми, що дозволяє ефективно працювати із задачами класифікації, детекції та сегментації.

Одними з базових архітектур згорткових мереж є прості послідовні моделі, що складаються з чергування згорткових і пулінгових шарів. До класичних прикладів належать LeNet-5 та AlexNet, які заклали основу для подальшого розвитку глибоких моделей. Вони продемонстрували можливість автоматичного виділення ознак із зображень без ручного проектування дескрипторів, що стало ключовим кроком у розвитку комп'ютерного зору.

Архітектура AlexNet наведена на рисунку 1.4.

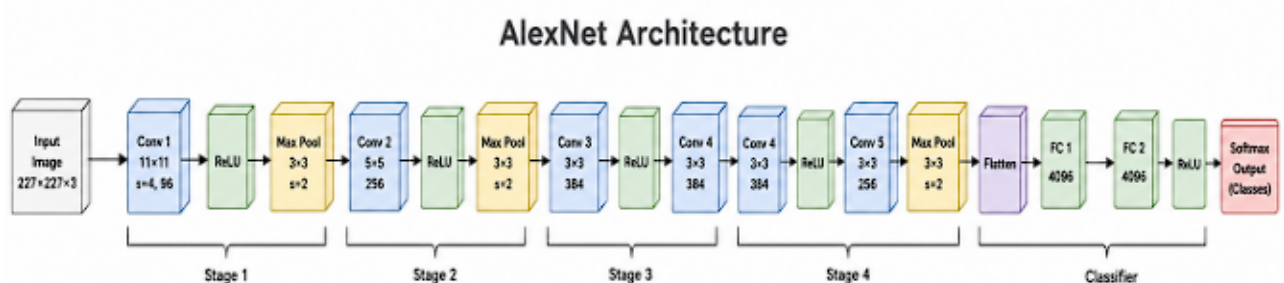


Рисунок 1.4 – Архітектура AlexNet

Подальший розвиток отримали більш складні архітектури, зокрема VGGNet, яка характеризується використанням глибоких стеків згорткових шарів із малими ядрами, та ResNet, що впроваджує залишкові з'єднання для подолання проблеми затухання градієнта. Завдяки таким удосконаленням стало можливим створення значно глибших мереж, які демонструють високу точність у задачах розпізнавання зображень.

Окрім задач класифікації, важливу роль відіграють моделі для сегментації зображень, які дозволяють визначати точне розташування об'єктів або їх частин. Прикладом такої архітектури є U-Net, яка широко використовується для обробки зображень із необхідністю точного виділення областей інтересу. У задачах аграрної сфери це дозволяє локалізувати уражені ділянки листя або інших частин рослин.

Окрему категорію становлять моделі для детекції об'єктів, такі як YOLO та Faster R-CNN, які дозволяють одночасно визначати клас об'єкта та його розташування на зображенні. Такі моделі можуть бути використані для виявлення окремих уражених зон або ідентифікації кількох типів захворювань на одному зображенні.

У сучасних дослідженнях також активно використовуються трансформерні архітектури, зокрема Vision Transformer, які базуються на механізмах уваги та дозволяють ефективно враховувати глобальні залежності у зображенні. Вони демонструють високу продуктивність у задачах класифікації та сегментації, особливо при наявності великих обсягів даних.

Найбільш актуальним підходом до обробки зображень у задачах діагностики захворювань рослин є використання згорткових нейронних мереж (Convolutional Neural Networks, CNN), які демонструють високу ефективність при роботі з візуальними даними. Їхня популярність обумовлена здатністю автоматично виділяти релевантні ознаки із зображень без необхідності ручного формування дескрипторів, що особливо важливо у випадку складних і варіативних симптомів захворювань.

Основною особливістю CNN є використання згорткових шарів, які застосовують фільтри до локальних ділянок зображення. Це дозволяє виявляти базові елементи, такі як краї, переходи кольору та текстури, які на наступних рівнях мережі комбінуються у більш складні структури. Таким чином формується ієрархічне представлення ознак: від простих до високорівневих, що безпосередньо відповідають специфічним візуальним патернам, характерним для певних захворювань.

Важливою перевагою згорткових мереж є їх інваріантність до зсувів та частково до масштабування об'єктів, що дозволяє ефективно аналізувати зображення рослин, отримані в реальних умовах. У польових умовах листя може бути розташоване під різними кутами, мати різний розмір або освітлення, однак CNN здатні зберігати стабільність розпізнавання за рахунок використання операцій пулінгу та спільного використання ваг.

Процес навчання таких моделей базується на великій кількості розмічених зображень, де кожне зображення відповідає певному класу захворювання або нормальному стану рослини. У ході навчання модель мінімізує функцію втрат, поступово коригуючи вагові коефіцієнти таким чином, щоб підвищити точність класифікації. Важливим аспектом є використання аугментації даних, яка дозволяє штучно збільшити різноманітність навчальної вибірки шляхом обертання, масштабування або зміни освітлення зображень.

У практичних задачах діагностики захворювань рослин CNN можуть застосовуватися як для класифікації зображень, так і для більш складних задач, таких як локалізація уражених ділянок або сегментація. Наприклад, модель може визначати не лише тип захворювання, але й область його поширення на листі, що є важливим для подальшого аналізу ступеня ураження. Це розширює функціональні можливості системи та дозволяє використовувати її не тільки для діагностики, але й для моніторингу.

Важливим етапом розвитку згорткових нейронних мереж стала архітектура GoogLeNet, також відома як Inception Network. Її ключовою особливістю є використання Inception-модулів, які дозволяють одночасно застосовувати декілька згорткових фільтрів різного розміру в межах одного шару. Такий підхід забезпечує ефективне виділення ознак різного масштабу та суттєво зменшує кількість параметрів моделі порівняно з класичними глибокими мережами. Завдяки високій точності та ефективності обчислень GoogLeNet широко використовується у задачах комп'ютерного зору, зокрема для класифікації зображень рослин та діагностики їх захворювань.

Архітектура GoogLeNet наведена на рисунку 1.5.

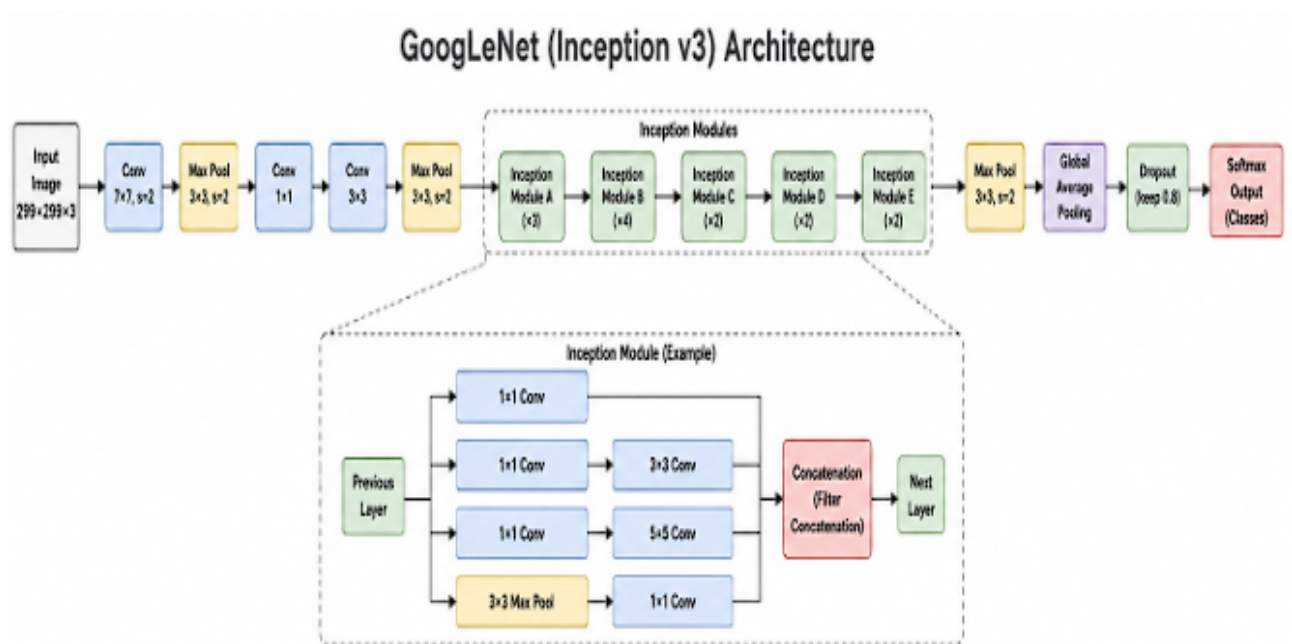


Рисунок 1.5 – Архітектура GoogLeNet

Проведений огляд моделей глибокого навчання для обробки зображень показує, що сучасні підходи базуються на використанні різних архітектур нейронних мереж, серед яких домінуючу роль відіграють згорткові моделі. Встановлено, що такі архітектури забезпечують ефективне автоматичне виділення ознак та демонструють високу точність у задачах класифікації, сегментації та детекції об'єктів.

Визначено, що розвиток моделей від простих архітектур до глибоких мереж із залишковими зв'язками та механізмами уваги дозволив суттєво підвищити якість обробки зображень. При цьому різні типи моделей мають свої особливості та області застосування, що обумовлює необхідність їх порівняльного аналізу у межах конкретної задачі.

Обґрунтовано, що найбільш доцільним підходом для задачі діагностики захворювань рослин є використання згорткових нейронних мереж, які забезпечують оптимальне поєднання точності, швидкості та адаптивності до варіативних візуальних даних. Це визначає подальший напрям дослідження, орієнтований на аналіз та порівняння ефективності таких моделей у практичних умовах.

1.5 Аналіз існуючих підходів і рішень

У сучасних дослідженнях представлено значну кількість реалізованих моделей глибокого навчання, орієнтованих на діагностику захворювань рослин за зображеннями. Одним із найбільш відомих прикладів є модель, запропонована Sharada P. Mohanty та співавторами, яка базується на згорткових нейронних мережах і навчена на наборі даних PlantVillage. У межах дослідження було використано архітектури типу AlexNet та GoogleNet, що дозволило досягти точності понад 99% на тестових даних. Дана робота стала однією з перших, що продемонструвала ефективність глибокого навчання у задачах класифікації захворювань рослин за зображеннями [3].

Іншим прикладом є модель, запропонована Konstantinos P. Ferentinos, яка використовує глибокі згорткові мережі для аналізу зображень різних культур. У дослідженні було протестовано кілька архітектур, зокрема VGG та AlexNet, на великому наборі даних із десятками класів захворювань. Отримані результати показали високу точність класифікації, що підтверджує доцільність використання глибоких моделей у реальних умовах [4].

Окрему увагу заслуговують рішення, орієнтовані на роботу в польових умовах, де зображення можуть містити шум, складний фон або нерівномірне освітлення. У роботі, присвяченій використанню глибоких нейронних мереж у реальних умовах, було запропоновано модель на основі CNN, здатну класифікувати захворювання без попереднього виділення листя. Такий підхід значно підвищує практичну цінність системи, оскільки усуває необхідність складної попередньої обробки даних [5].

Також існують моделі, що поєднують класифікацію з локалізацією уражених ділянок. Наприклад, використання архітектур типу YOLO дозволяє не лише визначити наявність захворювання, але й виділити конкретні області на зображенні, де воно проявляється. Це є важливим кроком до створення більш інформативних систем діагностики, які можуть використовуватися для оцінки ступеня ураження рослин.

В якості об'єкту більш детального аналізу розглядається робота Sharada P. Mohanty, у якій запропоновано підхід до автоматизованої діагностики захворювань рослин на основі глибокого навчання. У дослідженні використано великий відкритий набір даних PlantVillage, що містить 54 306 зображень листя різних культур, які охоплюють 14 видів рослин та 26 типів захворювань (разом із класом «здоровий стан»). Така кількість даних дозволила сформувати повноцінну навчальну вибірку для побудови високоточних моделей класифікації.

Оснoву підходу становить використання згорткових нейронних мереж, зокрема архітектур AlexNet та GoogLeNet. У роботі проведено серію експериментів із різними конфігураціями навчання, включаючи навчання «з нуля» та використання перенесення навчання. Зазначено, що моделі, попередньо навчені на великих датасетах (наприклад, ImageNet), демонструють кращі результати завдяки вже сформованим базовим візуальним ознакам, які можуть бути адаптовані до нової задачі.

Згідно з результатами дослідження, найкраща модель досягла точності близько 99,35% на тестовій вибірці, що свідчить про високу ефективність підходу в умовах контрольованого середовища. При цьому оцінювання проводилося з використанням стандартних метрик, зокрема точності, повноти та F1-міри, що дозволяє комплексно оцінити якість класифікації. Важливим є також те, що навіть при зменшенні обсягу навчальних даних модель зберігає високу якість, що свідчить про її узагальнювальну здатність.

У дослідженні також проведено аналіз впливу різних типів вхідних даних, зокрема кольорових, градацій сірого та сегментованих зображень. Встановлено, що найкращі результати досягаються при використанні кольорових зображень, оскільки колір є важливою ознакою для розпізнавання захворювань. Водночас сегментація зображень дозволяє частково підвищити точність за рахунок усунення фону, що зменшує вплив сторонніх факторів.

Окрему увагу приділено перевірці моделі на зображеннях, отриманих поза межами навчального набору. У таких умовах точність значно знижується (до приблизно 31%), що свідчить про обмежену здатність моделі до узагальнення в

реальних умовах. Це підкреслює необхідність використання більш різноманітних даних під час навчання та врахування варіативності середовища, у якому будуть застосовуватися моделі.

Таким чином, розглянута робота демонструє високий потенціал використання згорткових нейронних мереж для діагностики захворювань рослин, проте також виявляє низку обмежень, пов'язаних із узагальненням моделей на реальні дані. Це визначає необхідність подальших досліджень, спрямованих на підвищення стійкості моделей до змін умов зйомки та розширення навчальних вибірок.

Проведений аналіз роботи показує, що використання згорткових нейронних мереж є ефективним підходом для задачі діагностики захворювань рослин, що підтверджується високими показниками точності на наявних наборах даних. Водночас встановлено, що результати значною мірою залежать від характеристик навчальної вибірки, а також від обраної архітектури моделі та параметрів її навчання.

У межах даного дослідження доцільно зосередити увагу на вдосконаленні процесу навчання моделей та їх порівняльному аналізі. Зокрема, необхідно дослідити вплив різних архітектур згорткових нейронних мереж на якість класифікації, а також оцінити ефективність використання перенесення навчання у порівнянні з навчанням моделей «з нуля». Це дозволить визначити оптимальні конфігурації моделей для поставленої задачі.

Додатковим напрямом є дослідження впливу різних варіантів підготовки даних, зокрема використання кольорових, градацій сірого та сегментованих зображень. Аналіз таких підходів дозволить встановити, які саме характеристики вхідних даних мають найбільший вплив на точність моделі та як їх можна оптимізувати.

Також доцільно дослідити вплив параметрів навчання, включаючи розмір навчальної вибірки, співвідношення тренувальних і тестових даних, а також налаштування гіперпараметрів моделі. Це дозволить оцінити стійкість моделей до змін умов навчання та визначити оптимальні параметри для досягнення найкращих результатів.

Таким чином, для покращення результатів, отриманих у існуючих рішеннях, необхідно провести систематичний експериментальний аналіз різних архітектур моделей, підходів до навчання та варіантів підготовки даних. Це дозволить обґрунтовано визначити найбільш ефективні рішення та підвищити якість діагностики в межах дослідження.

1.6 Постановка задачі дослідження

Постановка задачі дослідження визначається необхідністю підвищення ефективності автоматизованого діагностування захворювань рослин на основі аналізу зображень із використанням моделей глибокого навчання. З урахуванням проведеного аналізу предметної області встановлено, що існуючі підходи забезпечують високі показники точності в умовах контрольованих наборів даних, проте потребують детального дослідження з точки зору оптимізації архітектур, параметрів навчання та структури вхідних даних.

Основною задачею дослідження є побудова та порівняльний аналіз моделей глибокого навчання для класифікації захворювань рослин за зображеннями листя. Передбачається визначення ефективності різних архітектур згорткових нейронних мереж у межах єдиної експериментальної постановки, що дозволить отримати об'єктивні результати щодо їх продуктивності.

У рамках дослідження необхідно сформулювати експериментальне середовище, яке забезпечить можливість коректного порівняння моделей. Це включає використання єдиного набору даних, уніфікованих підходів до попередньої обробки зображень, а також однакових метрик оцінювання якості. Такий підхід дозволяє мінімізувати вплив сторонніх факторів на результати та забезпечити їх відтворюваність.

Окремою задачею є дослідження впливу різних варіантів представлення вхідних даних на точність моделей. Передбачається аналіз ефективності використання кольорових, градацій сірого та сегментованих зображень, що дозволить визначити роль різних типів ознак у процесі класифікації.

Важливим аспектом є дослідження підходів до навчання моделей, зокрема порівняння навчання «з нуля» та використання перенесення навчання. Це дозволить оцінити доцільність використання попередньо навчених моделей та їх вплив на швидкість навчання і якість результатів.

Також у межах дослідження необхідно проаналізувати вплив гіперпараметрів на ефективність моделей, включаючи швидкість навчання, розмір батчу, кількість епох та інші параметри. Оптимізація цих значень є важливою складовою досягнення високої точності класифікації.

Окрему увагу слід приділити оцінюванню результатів моделей за допомогою стандартних метрик, таких як точність, повнота, F1-міра та загальна точність. Це дозволить здійснити комплексний аналіз якості моделей та виявити їх сильні та слабкі сторони.

У результаті дослідження передбачається визначення найбільш ефективної моделі або конфігурації моделі для задачі діагностики захворювань рослин. Це дозволить сформулювати обґрунтовані рекомендації щодо вибору архітектур та параметрів для подальшого використання у подібних задачах.

2 АНАЛІЗ І ВИБІР МЕТОДІВ ТА МОДЕЛЕЙ ДЛЯ ДОСЛІДЖЕННЯ

2.1 Формування вимог до системи діагностики

Формування вимог до системи діагностики захворювань рослин базується на необхідності забезпечення ефективного, точного та відтворюваного аналізу зображень із використанням моделей глибокого навчання. З урахуванням специфіки поставленої задачі система повинна бути орієнтована на обробку візуальних даних та реалізацію алгоритмів класифікації, що дозволяють визначати стан рослини за її зображенням.

Однією з ключових вимог є забезпечення достатнього рівня точності діагностики. Система повинна коректно класифікувати зображення за визначеними класами захворювань та здорового стану, що передбачає використання ефективних моделей глибокого навчання та належну підготовку даних. Важливим аспектом є також стабільність результатів при обробці різних типів зображень у межах одного набору даних.

Наступною вимогою є підтримка стандартизованого формату вхідних даних. Система повинна приймати зображення визначеного розміру та формату, що забезпечить коректну роботу моделей та уніфікацію процесу обробки. Це включає необхідність попередньої обробки даних, зокрема масштабування та нормалізації зображень.

Важливою вимогою є забезпечення можливості експериментального порівняння різних моделей. Система повинна підтримувати використання кількох архітектур глибокого навчання та забезпечувати їх оцінювання за єдиними критеріями. Це дозволить проводити об'єктивний аналіз ефективності моделей у межах дослідження.

Окрему групу становлять вимоги до процесу навчання моделей. Система повинна забезпечувати можливість налаштування параметрів навчання, включаючи кількість епох, розмір батчу та інші гіперпараметри. Це необхідно для проведення експериментів та визначення оптимальних конфігурацій моделей.

Також важливою є вимога до відтворюваності результатів. Усі експерименти повинні виконуватися в однакових умовах, що передбачає фіксацію параметрів навчання та використання єдиних наборів даних. Це дозволить забезпечити коректність порівняння отриманих результатів.

Система повинна забезпечувати обчислення основних метрик якості, таких як точність, повнота та F1-міра. Наявність цих показників дозволить здійснювати комплексну оцінку ефективності моделей та проводити їх порівняльний аналіз.

З урахуванням поставленої задачі, система має бути реалізована у вигляді консольного програмного засобу, який дозволяє передавати зображення для аналізу та отримувати результат класифікації. Такий підхід забезпечує простоту реалізації та достатню функціональність для проведення дослідження.

Забезпечення достатнього рівня точності діагностики є ключовою вимогою до системи, оскільки саме від цього залежить її практична цінність як інструменту аналізу. Досягнення високої точності класифікації можливе лише за умови комплексного підходу, який охоплює підготовку даних, вибір моделі, налаштування процесу навчання та коректну оцінку результатів.

Першочерговим фактором є якість і структура навчального набору даних. Для досягнення високої точності необхідно використовувати достатньо великий та збалансований датасет, у якому кожен клас захворювання представлений репрезентативною кількістю зображень. Важливо, щоб зображення чітко відображали характерні ознаки захворювань, оскільки саме на їх основі модель формує внутрішні представлення. Недостатня кількість прикладів або дисбаланс класів може призводити до зміщення моделі та зниження точності.

Другим важливим аспектом є попередня обробка даних. Необхідно забезпечити уніфікацію розмірів зображень, нормалізацію кольорових значень та усунення можливих артефактів. Це дозволяє зменшити вплив сторонніх факторів і сприяє стабільнішому процесу навчання. Додатково застосування аугментації даних дозволяє підвищити різноманітність навчальної вибірки, що позитивно впливає на здатність моделі узагальнювати.

Важливу роль відіграє вибір архітектури моделі. Для задач класифікації зображень доцільно використовувати сучасні згорткові нейронні мережі, які

здатні ефективно виділяти просторові ознаки. При цьому необхідно експериментально визначити, яка саме архітектура забезпечує найкращі результати для конкретного набору даних.

Окрему увагу слід приділити налаштуванню процесу навчання. Досягнення високої точності вимагає оптимального підбору гіперпараметрів, таких як швидкість навчання, кількість епох, розмір батчу та параметри регуляризації. Неправильні налаштування можуть призводити до недонавчання або перенавчання моделі, що негативно впливає на якість результатів.

Також доцільно використовувати підхід перенесення навчання, який дозволяє використовувати попередньо навчені моделі як основу для подальшого донавчання. Це дає змогу покращити точність, особливо у випадках обмеженого обсягу даних, оскільки модель вже має базові знання про візуальні ознаки.

Не менш важливим є коректне оцінювання якості моделі. Використання метрик, таких як точність, повнота та F1-міра, дозволяє отримати об'єктивну оцінку результатів і виявити можливі проблеми у класифікації. Це дає змогу своєчасно коригувати параметри моделі та покращувати її продуктивність.

Підтримка стандартизованого формату вхідних даних є необхідною умовою коректної роботи системи, оскільки моделі глибокого навчання вимагають уніфікованого представлення вхідної інформації. У контексті задачі діагностики захворювань рослин це стосується насамперед формату зображень, їх розмірності, кольорового представлення та структури даних, що подаються на вхід моделі.

Основними форматами файлів зображень, які доцільно підтримувати, є JPEG (JPG) та PNG. Формат JPEG широко використовується завдяки високому ступеню стиснення та невеликому розміру файлів, що є зручним для зберігання великих наборів даних. Формат PNG, у свою чергу, забезпечує збереження зображення без втрат якості, що може бути важливим при аналізі дрібних деталей, характерних для ознак захворювань. Обидва формати є стандартними для більшості бібліотек обробки зображень і легко інтегруються у програмні рішення.

Окрім формату файлу, важливою вимогою є стандартизація розміру зображень. У більшості моделей глибокого навчання використовується фіксований розмір вхідного тензора, наприклад 224×224 або 256×256 пікселів. Тому всі вхідні зображення повинні бути приведені до єдиного розміру шляхом масштабування. Це забезпечує узгодженість вхідних даних і дозволяє уникнути помилок під час обробки.

Ще одним аспектом є кольорове представлення зображень. Система повинна підтримувати як кольорові зображення у форматі RGB, так і, за необхідності, їх перетворення у градації сірого. Формат RGB є основним для більшості моделей, оскільки дозволяє використовувати додаткову інформацію про колір, яка є важливою для розпізнавання захворювань. Водночас використання grayscale-зображень може бути доцільним у рамках експериментального аналізу.

Важливо також забезпечити нормалізацію вхідних даних. Це включає приведення значень пікселів до визначеного діапазону, наприклад $[0,1]$ або $[-1,1]$, а також, за потреби, стандартизацію за середнім значенням і стандартним відхиленням. Така обробка дозволяє стабілізувати процес навчання та покращити збіжність моделей.

Додатково система повинна враховувати формат представлення даних у вигляді тензорів, який використовується у фреймворках глибокого навчання. Зокрема, зображення повинні бути перетворені у багатовимірні масиви з визначеним порядком розмірностей, наприклад (висота, ширина, канали) або (канали, висота, ширина), залежно від використовуваної бібліотеки.

Забезпечення обчислення основних метрик якості досягається шляхом формування в системі окремого етапу оцінювання результатів класифікації після завершення навчання або тестування моделі. На цьому етапі фактичні класи, передбачені моделлю для кожного зображення, порівнюються з еталонними мітками, що містяться у тестовому наборі даних. Саме на основі такого порівняння можуть бути обчислені кількісні показники якості, які відображають ефективність моделі в задачі діагностики захворювань рослин.

Для реалізації цього підходу система повинна зберігати два масиви значень: правильні мітки класів і результати прогнозування моделі. Після проходження тестового набору формується сукупність передбачень, яка використовується для побудови матриці помилок. Матриця помилок є базовою структурою для подальшого обчислення метрик, оскільки дозволяє визначити кількість правильних і помилкових класифікацій для кожного класу окремо.

Показник точності визначається як відношення кількості правильно класифікованих об'єктів певного класу до загальної кількості об'єктів, які модель віднесла до цього класу. Для системи діагностики це означає, наскільки достовірними є позитивні рішення моделі щодо конкретного захворювання. Повнота, у свою чергу, показує, яку частину реальних зразків певного класу модель змогла правильно виявити. Така метрика є важливою тоді, коли необхідно оцінити, наскільки добре система знаходить всі випадки певного захворювання, не пропускаючи їх.

F1-міра обчислюється як гармонійне середнє між точністю та повнотою. Її використання є доцільним у тих випадках, коли необхідно отримати узагальнену оцінку якості моделі без переваги лише одного з цих двох показників. Для задачі класифікації захворювань рослин це особливо важливо, оскільки модель повинна не лише правильно розпізнавати хворобу, а й не створювати надлишкової кількості хибних спрацьовувань.

Для досягнення обчислення цих метрик у системі необхідно реалізувати модуль оцінювання, який автоматично оброблятиме результати тестування моделі. Такий модуль повинен приймати на вхід еталонні значення та результати прогнозування, після чого виконувати розрахунок метрик для кожного класу, а також обчислювати усереднені показники для всієї моделі. Це забезпечить можливість комплексного аналізу якості класифікації.

Практично така функціональність може бути реалізована за допомогою стандартних бібліотек машинного навчання, які вже містять готові засоби для обчислення accuracy, precision, recall та F1-score. Проте навіть при використанні таких бібліотек система повинна бути спроектована так, щоб результати

оцінювання зберігалися, відображалися у зручному вигляді та могли бути використані для подальшого порівняльного аналізу різних моделей.

Додатково доцільно передбачити обчислення метрик не лише для окремого запуску, а й для серії експериментів із різними параметрами навчання. Це дозволить встановити, як зміна архітектури, типу вхідних даних або гіперпараметрів впливає на кінцеву якість класифікації. У такому випадку метрики стають не лише засобом оцінювання, а й інструментом вибору найкращої конфігурації моделі.

Забезпечення обчислення точності, повноти та F1-міри досягається шляхом організації повноцінного етапу тестування моделі, формування матриці помилок, автоматизованого розрахунку показників якості та збереження отриманих результатів для подальшого аналізу. Саме такий підхід дозволяє об'єктивно оцінювати ефективність системи діагностики та обґрунтовано порівнювати різні моделі глибокого навчання [11].

Сформовані вимоги до системи діагностики визначають її як інструмент дослідження моделей глибокого навчання, орієнтований на забезпечення точності, відтворюваності та коректності обробки візуальних даних. Встановлено, що ключовими аспектами є якість підготовки вхідних даних, стандартизація їх формату, можливість гнучкого налаштування процесу навчання та об'єктивне оцінювання результатів за допомогою метрик.

Визначено, що система повинна забезпечувати єдине експериментальне середовище для порівняння різних моделей, що дозволяє мінімізувати вплив сторонніх факторів та отримати достовірні результати. При цьому важливу роль відіграє можливість контролю параметрів навчання та аналізу їх впливу на якість класифікації.

2.2 Опис і аналіз наборів даних для навчання моделей

У межах дослідження одним із ключових етапів є формування та аналіз наборів даних, що використовуються для навчання моделей глибокого навчання.

Якість та структура датасету безпосередньо впливають на результати класифікації, тому вибір джерела даних є критично важливим аспектом.

Для пошуку відповідних наборів даних планується використання платформи Kaggle, яка є одним із найбільших відкритих ресурсів у сфері аналізу даних і машинного навчання. Дана платформа надає доступ до великої кількості датасетів, у тому числі спеціалізованих наборів зображень захворювань рослин, що вже були використані у наукових дослідженнях та практичних задачах.

Перевагою використання цієї платформи є наявність структурованих і підготовлених наборів даних, які містять розмічені зображення різних культур і типів захворювань. Такі датасети зазвичай включають поділ на класи, що дозволяє безпосередньо використовувати їх для задач класифікації без необхідності додаткового маркування. Крім того, багато наборів даних супроводжуються описами, що містять інформацію про джерела зображень, умови їх отримання та характеристики класів.

У процесі відбору датасетів основна увага приділятиметься їх обсягу, різноманітності та якості зображень. Необхідно, щоб датасет містив достатню кількість прикладів для кожного класу, що дозволить уникнути проблеми дисбалансу та забезпечить коректне навчання моделей. Важливим є також різноманіття зображень у межах кожного класу, що сприяє формуванню більш узагальнених моделей.

Додатковим критерієм вибору є формат і структура даних. Датасети повинні бути організовані таким чином, щоб забезпечити зручність їх використання у процесі навчання, зокрема мати чітку ієрархію каталогів або супровідні файли з мітками класів. Це дозволяє автоматизувати процес завантаження даних і зменшити ймовірність помилок.

Також у межах аналізу планується оцінка відповідності датасетів поставленій задачі дослідження. Зокрема, перевіряється, чи містять вони необхідні класи захворювань, чи відповідає якість зображень вимогам моделей глибокого навчання, а також чи є достатньою кількістю даних для проведення експериментів.

Після відбору датасетів буде проведено їх детальний аналіз, включаючи визначення кількості класів, розподілу зображень між класами та загального обсягу даних. Це дозволить сформувати уявлення про структуру вхідних даних і підготувати їх до подальшої обробки.

У процесі формування навчальної вибірки основну увагу було зосереджено на використанні датасету PlantVillage, який є одним із найбільш відомих і широко застосовуваних у задачах діагностики захворювань рослин. Даний набір даних містить велику кількість зображень листя різних культур із відповідними мітками захворювань або здорового стану, що робить його базовим еталоном для досліджень у цій предметній області.

Разом із тим, у ході пошуку додаткових джерел даних на платформі Kaggle було виявлено датасет під назвою «Plant Leaf Diseases Training Dataset», який представляє собою об'єднання кількох наборів даних, включаючи розширені версії PlantVillage. Такий підхід дозволяє суттєво збільшити обсяг навчальної вибірки та розширити різноманіття представлених класів і зображень.

Особливістю цього датасету є те, що він містить більшу кількість прикладів для окремих класів захворювань, а також може включати додаткові варіації зображень, що відрізняються за умовами зйомки, формою уражень та іншими характеристиками. Це дозволяє моделі отримати більш повне уявлення про ознаки захворювань і підвищує її здатність до узагальнення.

З огляду на це, було прийнято рішення використовувати саме «Plant Leaf Diseases Training Dataset» як основний набір даних для навчання моделей. Такий вибір обумовлений необхідністю забезпечення достатнього обсягу та різноманітності даних, що є важливим фактором для досягнення високої точності класифікації.

При цьому датасет PlantVillage розглядається як основа, на якій побудовано розширений набір даних, що дозволяє зберегти узгодженість із попередніми дослідженнями та забезпечити можливість порівняння результатів. Використання розширеного датасету дає змогу поєднати переваги перевіреного базового набору та додаткових даних, що покращують якість навчання.

На рисунку 2.1 наведено структуру директорії обраного датасету, що використовується для навчання моделей глибокого навчання. Організація даних реалізована у вигляді ієрархічної файлової структури, де кожен підкаталог відповідає окремому класу, що представляє конкретне захворювання рослини або її здоровий стан.

Назви директорій формуються за уніфікованим принципом і містять інформацію як про тип рослини, так і про відповідне захворювання. Наприклад, директорії типу `Apple_black_rot`, `Apple_scab` або `Bell_pepper_healthy` дозволяють однозначно ідентифікувати клас зображень, що в них містяться. Такий підхід забезпечує зрозумілу та логічну структуру даних, що є важливим для автоматизованої обробки.

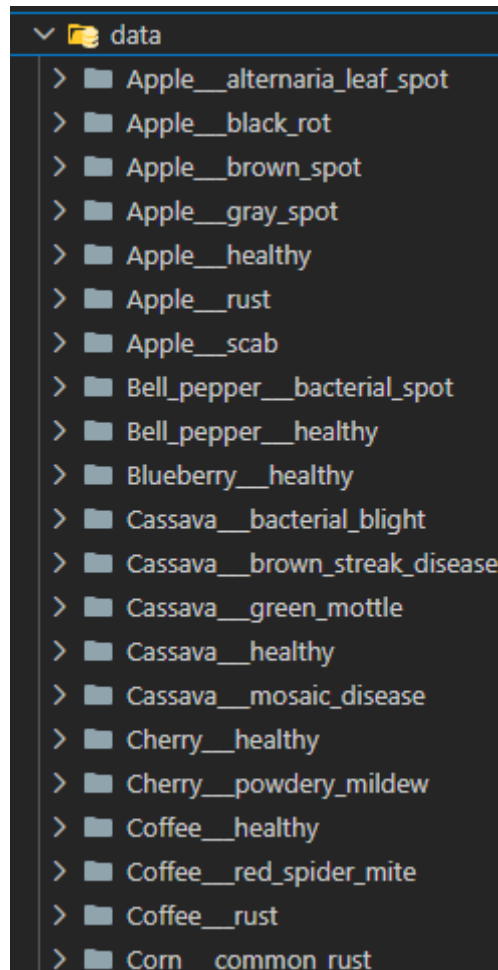


Рисунок 2.1 – Перелік класів датасету в його директорії

Кожна директорія містить набір зображень листя, які належать до відповідного класу. Це дозволяє використовувати стандартні механізми

завантаження даних у фреймворках глибокого навчання, де мітка класу визначається на основі назви каталогу. Така організація значно спрощує процес підготовки даних та зменшує необхідність у додатковому маркуванні.

Особливістю представленої структури є наявність як класів захворювань, так і класів, що відповідають здоровому стану рослин. Це дозволяє моделі навчатися не лише розпізнавати патології, але й відрізняти їх від нормального стану, що є критично важливим для задачі діагностики.

Також слід відзначити, що датасет охоплює різні види рослин, зокрема яблуню, перець, каву, кукурудзу та інші культури. Це забезпечує різноманітність навчальних даних і дозволяє досліджувати здатність моделей працювати з кількома типами об'єктів у межах єдиної задачі класифікації.

На рисунку 2.2 наведено приклад зображення, що належить до класу «Apple_black_rot», який відповідає захворюванню яблуні, відомому як чорна гниль. На зображенні представлено листок, на поверхні якого спостерігаються характерні ознаки ураження, зокрема наявність темної некротичної ділянки у верхній частині листка.



Рисунок 2.2 – Приклад зображення з класу «Apple_black_rot»

Уражена область має чітко виражене потемніння тканин, що є типовим проявом некрозу. Такі зміни свідчать про відмирання клітин у результаті розвитку патогену. Крім того, помітна нерівномірність структури ураженої ділянки, що може бути додатковою ознакою інфекційного процесу. Здорова

частина листка при цьому зберігає насичений зелений колір, що створює контраст із пошкодженою зоною.

Важливою особливістю даного зображення є наявність однорідного фону, який не містить сторонніх об'єктів. Це значно спрощує задачу класифікації, оскільки модель може зосередитися виключно на аналізі самого листка та його ознак. Подібні умови характерні для датасету PlantVillage, де зображення зазвичай отримані в контрольованому середовищі.

Також слід відзначити, що форма та структура листка залишаються добре видимими, що дозволяє моделі враховувати не лише локальні ознаки ураження, але й загальні характеристики об'єкта. Це може позитивно впливати на якість класифікації, оскільки забезпечує більш повне представлення вхідних даних.

На рисунку 2.3 наведено приклад зображення, що належить до класу «coffee_red_spider_mite», який відповідає ураженню листя кави червоним павутинним кліщем. На відміну від попередніх прикладів, зображення отримано не в контрольованих умовах, а в природному середовищі, що ускладнює задачу аналізу через наявність складного фону та додаткових об'єктів.



Рисунок 2.3 – Приклад зображення з класу «coffee_red_spider_mite»

Листок має характерні ознаки ураження у вигляді дрібних плям і знебарвлених ділянок, що розподілені по всій поверхні. Ці зміни проявляються у вигляді нерівномірного пожовтіння та точкових пошкоджень, які є наслідком діяльності кліщів, що порушують структуру тканин. Ураження не мають чітко

виражених меж і виглядають як розсіяні дефекти, що ускладнює їх візуальне виділення.

Важливою особливістю даного зображення є наявність природного фону, який містить інші елементи рослинності та ґрунту. Це створює додатковий шум і знижує контраст між об'єктом та оточенням. У таких умовах модель повинна не лише розпізнавати ознаки захворювання, але й коректно виділяти сам листок як область інтересу.

Також слід відзначити, що освітлення є нерівномірним, а сам листок має складнішу геометрію порівняно з попередніми прикладами. Це призводить до появи тіней і варіацій кольору, які можуть впливати на результати класифікації.

На рисунку 2.4 наведено приклад зображення, що належить до класу «peach_bacterial_spot», який відповідає бактеріальній плямистості персика. На зображенні представлено листок із характерними ознаками ураження, що проявляються у вигляді множинних дрібних темних плям, розташованих переважно вздовж поверхні листкової пластини.



Рисунок 2.4 – Приклад з класу «peach_bacterial_spot»

Плями мають нерегулярну форму та темно-коричневий відтінок, що є типовою ознакою бактеріальних уражень. У деяких випадках спостерігається злиття окремих плям у більші уражені ділянки, що свідчить про прогресування захворювання. При цьому навколишні тканини листка залишаються відносно

здоровими, зберігаючи зелений колір, що створює контраст і дозволяє чітко виділити області пошкодження.

Особливістю даного зображення є рівномірний розподіл уражень по поверхні листка, що відрізняє цей тип захворювання від локалізованих некротичних змін. Також помітні незначні деформації країв листка, які можуть супроводжувати розвиток інфекційного процесу.

Фон зображення є однорідним і не містить сторонніх об'єктів, що відповідає умовам формування датасету та сприяє більш точному аналізу. Чітке відокремлення об'єкта від фону дозволяє моделі зосередитися на ключових візуальних ознаках без впливу додаткового шуму.

Обраний датасет характеризується достатнім обсягом, чіткою структурою та наявністю розмічених класів, що забезпечує його придатність для задачі навчання моделей глибокого навчання. Використання розширеного набору даних на основі PlantVillage дозволяє поєднати якість контрольованих зображень із підвищеним різноманіттям прикладів, що позитивно впливає на процес навчання.

Встановлено, що структура директорій забезпечує зручність автоматизованої обробки даних, а наявність як чітко виражених, так і менш очевидних ознак ураження створює умови для повноцінного дослідження можливостей моделей. Це дозволяє оцінити їх здатність розпізнавати різні типи захворювань за візуальними характеристиками.

Обраний датасет є достатньо репрезентативним для проведення експериментального дослідження та забезпечує необхідну основу для порівняльного аналізу моделей глибокого навчання у задачі діагностики захворювань рослин.

2.3 Попередня обробка зображень

Попередня обробка зображень є необхідним етапом підготовки даних для навчання моделей глибокого навчання, оскільки якість вхідної інформації безпосередньо впливає на ефективність класифікації. Зображення, отримані з

різних джерел, можуть відрізнятися за розміром, освітленням, контрастом та іншими характеристиками, що створює додаткову складність для моделі. Без уніфікації таких даних процес навчання може бути нестабільним, а результати – менш точними.

Однією з основних задач попередньої обробки є стандартизація розміру зображень. Оскільки більшість архітектур згорткових нейронних мереж вимагають фіксованого розміру вхідного тензора, усі зображення необхідно привести до єдиного формату, наприклад 224×224 пікселів. Це забезпечує коректну роботу моделі та дозволяє формувати батчі для навчання.

Важливим етапом є нормалізація значень пікселів. Зображення зазвичай представлені у вигляді числових значень у діапазоні від 0 до 255, що може ускладнювати процес навчання. Приведення цих значень до стандартизованого діапазону, наприклад $[0,1]$, або виконання стандартизації дозволяє покращити збіжність алгоритмів оптимізації та підвищити стабільність навчання.

Додатково застосовується корекція кольорових характеристик, що дозволяє зменшити вплив різних умов освітлення. Це може включати вирівнювання гістограми, корекцію яскравості та контрасту, що особливо важливо для задачі діагностики захворювань, де колір є однією з ключових ознак.

Ще одним важливим компонентом є аугментація даних, яка дозволяє штучно збільшити обсяг навчальної вибірки. До основних методів належать обертання, віддзеркалення, масштабування та зсув зображень. Це сприяє підвищенню узагальнювальної здатності моделі та зменшенню ризику перенавчання.

У деяких випадках доцільним є застосування сегментації зображень, яка дозволяє виділити область інтересу, наприклад листок, і усунути фон. Це дозволяє моделі зосередитися на релевантних ознаках та зменшити вплив сторонніх елементів, що можуть негативно впливати на результати класифікації.

Також попередня обробка включає перетворення зображень у формат, придатний для використання у фреймворках глибокого навчання. Зокрема,

зображення перетворюються у багатовимірні масиви (тензори) з визначеним порядком розмірностей, що відповідає вимогам обраної бібліотеки.

На лістингу A.8 наведено реалізацію процесу попередньої обробки зображень, яка забезпечує уніфікацію вхідних даних перед їх використанням у моделях глибокого навчання. Основною метою даного етапу є приведення зображень до єдиного формату, що відповідає вимогам обраних архітектур.

Функція `preprocess_and_save_image` виконує базові операції над кожним зображенням. Зокрема, здійснюється відкриття файлу, після чого зображення перетворюється у формат RGB. Це є важливим кроком, оскільки різні зображення можуть мати різну кількість каналів, а більшість моделей глибокого навчання очікують саме триканальне представлення.

Наступним етапом є зміна розміру зображення до фіксованого значення, у даному випадку 224×224 пікселів. Така стандартизація дозволяє забезпечити узгодженість вхідних даних та коректну роботу згорткових нейронних мереж. Незалежно від початкового розміру, усі зображення приводяться до однакового формату, що є необхідною умовою для формування батчів під час навчання.

Після обробки зображення зберігається у форматі PNG у відповідній директорії, що відповідає його класу. При цьому автоматично створюється необхідна структура папок, якщо вона відсутня. Такий підхід дозволяє зберегти організацію датасету та забезпечити зручність подальшого використання.

Основна частина коду реалізує ітерацію по всіх записах датасету, які зберігаються у структурі даних. Для кожного зображення визначається шлях до вихідного файлу, а також формується шлях для збереження обробленого зображення з урахуванням його класу. Це дозволяє автоматизувати процес обробки великої кількості даних.

У процесі виконання передбачено обробку можливих помилок, які можуть виникати при роботі з файлами. Усі такі випадки фіксуються у окремому списку, що дозволяє контролювати якість даних та виявляти проблемні зображення. Одночасно формується список успішно оброблених файлів, який у подальшому перетворюється у табличний формат.

Результатом виконання є два набори даних: оброблені зображення та інформація про них, а також перелік помилок, що виникли під час обробки. Це забезпечує прозорість процесу підготовки даних та дозволяє оцінити його ефективність.

На рисунку 2.5 наведено приклад зображення після виконання попередньої обробки відповідно до алгоритму, описаного у лістингу 2.1. Оброблене зображення приведене до стандартизованого розміру та кольорового формату, що забезпечує його сумісність із моделями глибокого навчання.

Зображення має фіксовану розмірність, що дозволяє використовувати його як елемент батчу під час навчання. Незважаючи на зміну масштабу, збережено основні візуальні ознаки, зокрема форму листка та характерні ураження, що є важливими для подальшої класифікації. Це свідчить про коректність виконаної операції ресайзингу, яка не призводить до втрати суттєвої інформації.

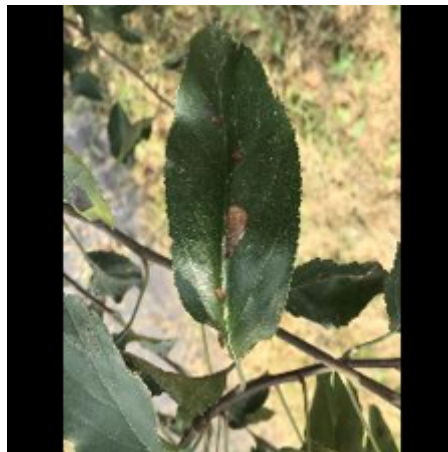


Рисунок 2.5 – Приклад обробленого зображення

Також зображення представлено у форматі RGB, що забезпечує збереження кольорових характеристик. Це є важливим аспектом, оскільки колір виступає однією з ключових ознак при розпізнаванні захворювань рослин. У результаті попередньої обробки забезпечується узгодженість кольорового представлення для всіх зображень датасету.

Важливою особливістю є збереження структури даних, оскільки оброблене зображення розміщується у директорії, що відповідає його класу. Це дозволяє

безпосередньо використовувати підготовлений датасет у процесі навчання без додаткових перетворень.

При цьому слід відзначити, що хоча основні параметри зображення стандартизовано, складність фону залишається незмінною. Це означає, що модель у подальшому повинна враховувати наявність сторонніх елементів та навчатися виділяти релевантні ознаки без додаткової сегментації.

Попередня обробка зображень забезпечує уніфікацію вхідних даних шляхом приведення їх до єдиного розміру, кольорового формату та структури зберігання. Це дозволяє усунути неоднорідність даних та створити стабільні умови для навчання моделей глибокого навчання.

Встановлено, що виконані перетворення не призводять до втрати ключових візуальних ознак, необхідних для діагностики захворювань, що підтверджує доцільність обраного підходу. При цьому зберігається інформативність зображень, що є критично важливим для задачі класифікації.

Таким чином, попередня обробка виступає необхідним етапом підготовки даних, який забезпечує коректність роботи моделей та сприяє підвищенню точності їх результатів.

2.4 Вибір архітектур моделей глибокого навчання

У межах дослідження вибір архітектур моделей глибокого навчання здійснюється з урахуванням специфіки задачі класифікації зображень захворювань рослин та необхідності проведення порівняльного аналізу різних підходів. Основною вимогою до моделей є здатність ефективно працювати з візуальними ознаками різної складності, зберігаючи при цьому баланс між точністю та обчислювальною складністю.

На початковому етапі було розглянуто широкий спектр архітектур згорткових нейронних мереж, які традиційно застосовуються у задачах комп'ютерного зору. До цього переліку входили як базові моделі, що мають просту структуру, так і більш складні глибокі архітектури, які використовують сучасні механізми оптимізації.

Серед базових варіантів було проаналізовано архітектури типу AlexNet, які характеризуються відносною простотою реалізації та можуть бути використані як початковий рівень для оцінки ефективності підходу. Такі моделі дозволяють швидко отримати результати та зрозуміти загальні тенденції у навчанні.

Окрім цього, було розглянуто більш глибокі архітектури, зокрема VGG-подібні моделі, які забезпечують кращу якість класифікації за рахунок збільшення кількості шарів. Однак їх використання пов'язане з підвищеними вимогами до обчислювальних ресурсів, що може ускладнювати проведення експериментів.

Також у процесі вибору враховувалися сучасні архітектури, такі як ResNet, які використовують залишкові з'єднання для покращення процесу навчання глибоких мереж. Подібні моделі демонструють високу точність, проте їх складність потребує додаткових ресурсів та часу на навчання.

З урахуванням поставленої задачі було прийнято рішення обмежити набір досліджуваних архітектур двома моделями, які дозволяють забезпечити баланс між складністю та ефективністю. Такий підхід дозволяє зосередитися на детальному аналізі та отримати більш інтерпретовані результати.

У якості першої моделі обрано AlexNet, яка виступає як базова архітектура для задачі класифікації. Її використання дозволяє оцінити ефективність класичного підходу та визначити мінімальний рівень якості, який може бути досягнутий без використання складних механізмів.

У якості другої моделі обрано GoogLeNet, яка є більш складною архітектурою та використовує інцепційні модулі для аналізу ознак різного масштабу. Це дозволяє моделі краще враховувати складні візуальні патерни, характерні для захворювань рослин.

Вибір саме цих архітектур обумовлений також тим, що вони використовувалися у попередньо проаналізованих дослідженнях, що дозволяє забезпечити узгодженість підходу та можливість порівняння отриманих результатів із відомими роботами.

У процесі дослідження обидві моделі будуть навчатися на одному й тому ж наборі даних із використанням однакових параметрів оцінювання. Це дозволяє виключити вплив зовнішніх факторів і забезпечити об'єктивність порівняння.

Особлива увага приділятиметься аналізу точності моделей, а також їх здатності стабільно працювати на різних класах захворювань. Це дозволить визначити, яка з архітектур краще підходить для поставленої задачі.

Також буде враховано обчислювальну ефективність моделей, зокрема час навчання та використання ресурсів. Хоча це не є основною метою дослідження, даний аспект може впливати на вибір оптимальної архітектури.

У результаті проведеного вибору формується експериментальна основа для подальшого дослідження, яка дозволяє здійснити порівняльний аналіз моделей глибокого навчання та визначити найбільш ефективний підхід для задачі діагностики захворювань рослин.

2.5 Метрики оцінювання якості моделей

Оцінювання якості моделей глибокого навчання є невід'ємною складовою дослідження, оскільки дозволяє об'єктивно визначити ефективність їх роботи у задачі класифікації захворювань рослин. Вибір метрик оцінювання здійснюється з урахуванням специфіки задачі, а саме багатокласової класифікації, де важливо не лише визначити загальну точність, але й оцінити якість розпізнавання окремих класів.

Основною метрикою, яка використовується у дослідженні, є загальна точність (accuracy). Вона визначає частку правильно класифікованих зображень відносно загальної кількості прикладів. Дана метрика є простою у розрахунку та дозволяє отримати загальне уявлення про якість моделі, однак вона не враховує можливий дисбаланс класів:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}, \quad (1.1)$$

де TP — кількість правильно визначених позитивних прикладів;

TN — кількість правильно визначених негативних прикладів;

FP — кількість хибнопозитивних результатів;

FN — кількість хибнонегативних результатів.

З метою більш детального аналізу використовується точність (precision), яка визначає частку правильних передбачень серед усіх випадків, коли модель віднесла об'єкт до певного класу. Це дозволяє оцінити, наскільки надійними є позитивні рішення моделі, що є важливим у задачі діагностики, де помилкові спрацювання можуть призводити до некоректних висновків:

$$Precision = \frac{TP}{TP+FP}. \quad (1.2)$$

Доповнюючою метрикою є повнота (recall), яка характеризує здатність моделі знаходити всі об'єкти певного класу. Вона визначає, яка частка реальних прикладів захворювання була правильно ідентифікована моделлю. Це є важливим аспектом, оскільки пропуск захворювання може мати негативні наслідки

$$Recall = \frac{TP}{TP+FN}. \quad (1.3)$$

Для забезпечення балансу між точністю та повнотою використовується F1-міра, яка є гармонійним середнім цих двох показників. Вона дозволяє отримати узагальнену оцінку якості моделі, особливо у випадках, коли важливо враховувати обидва аспекти одночасно

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}. \quad (1.4)$$

Оскільки задача є багатокласовою, розрахунок метрик здійснюється для кожного класу окремо, а також у вигляді усереднених значень. Для цього можуть

використовуватися різні підходи до усереднення, зокрема macro- та weighted-усереднення, що дозволяє враховувати або не враховувати розмір класів

$$Macro-Average = \frac{1}{N} \sum_{i=1}^N Metric_i, \quad (1.5)$$

$$Weighted-Average = \frac{\sum_{i=1}^N w_i \cdot Metric_i}{\sum_{i=1}^N w_i}, \quad (1.6)$$

де N — кількість класів;

$Metric_i$ — значення метрики для i -го класу;

w_i — кількість прикладів у i -му класі.

Додатковим інструментом оцінювання є матриця помилок, яка дозволяє візуалізувати результати класифікації та визначити, які саме класи найчастіше плутає модель. Це дає змогу глибше проаналізувати поведінку моделі та виявити проблемні області.

Важливою перевагою використання декількох метрик є можливість комплексного оцінювання якості моделі. Одна метрика не може повністю відобразити всі аспекти її роботи, тому використання їх сукупності дозволяє отримати більш об'єктивні результати.

У межах дослідження обрані метрики будуть використовуватися для порівняння різних архітектур моделей. Це дозволить визначити, яка з моделей демонструє кращі результати не лише за загальною точністю, але й за іншими критеріями.

Також передбачається аналіз зміни значень метрик у процесі навчання, що дозволить оцінити стабільність моделей та виявити ознаки перенавчання або недонавчання.

Використання обраних метрик дозволяє забезпечити відтворюваність результатів дослідження, оскільки вони є стандартними у задачах машинного навчання та широко застосовуються у наукових роботах.

Таким чином, вибір метрик оцінювання базується на необхідності комплексного аналізу якості моделей, що враховує як загальну точність, так і здатність моделі коректно розпізнавати окремі класи захворювань.

У результаті у дослідженні використовується наступний перелік метрик: accuracy, precision, recall, F1-score, а також матриця помилок як допоміжний інструмент аналізу.

Обрані метрики оцінювання забезпечують комплексний аналіз якості моделей глибокого навчання у задачі класифікації захворювань рослин. Поєднання accuracy, precision, recall та F1-score дозволяє оцінити як загальну ефективність моделі, так і її здатність коректно розпізнавати окремі класи без перекосу в бік окремих показників.

Використання декількох метрик забезпечує більш об'єктивну інтерпретацію результатів та дозволяє виявити слабкі сторони моделей. Це створює необхідну основу для їх порівняння та вибору найбільш ефективного підходу у межах дослідження.

3 РЕАЛІЗАЦІЯ МОДЕЛЕЙ ТА ПРОГРАМНОГО ЗАСОБУ ДОСЛІДЖЕННЯ

3.1 Обґрунтування вибору інструментів та технологій

Вибір інструментів та технологій для проведення дослідження є важливим етапом побудови системи діагностики захворювань рослин, оскільки від використовуваного програмного та апаратного забезпечення залежить ефективність реалізації моделей глибокого навчання, швидкість виконання експериментів та можливість подальшого аналізу результатів.

У якості основної мови програмування було обрано Python, який є одним із найбільш поширених інструментів у сфері машинного навчання та комп'ютерного зору. Дана мова характеризується великою кількістю бібліотек для роботи з нейронними мережами, простотою реалізації алгоритмів та активною підтримкою з боку наукової спільноти. Крім того, Python забезпечує зручну інтеграцію різних компонентів системи в межах єдиного середовища.

Для реалізації моделей глибокого навчання обрано фреймворк TensorFlow та його високорівневий інтерфейс Keras. Використання цих інструментів дозволяє швидко створювати та навчати нейронні мережі, а також забезпечує підтримку сучасних архітектур і засобів оптимізації. Додатковою перевагою є можливість використання графічних процесорів для прискорення обчислень.

У задачах попередньої обробки зображень та роботи з датасетами використовується бібліотека Pillow, яка забезпечує базові засоби для відкриття, зміни розміру, конвертації та збереження зображень. Її використання дозволяє реалізувати стандартизацію вхідних даних перед передачею їх до моделей глибокого навчання.

Для роботи з табличними структурами даних та збереженням інформації про датасет застосовується бібліотека Pandas. Вона дозволяє ефективно працювати з метаданими, формувати структури даних для подальшого аналізу та автоматизувати процес підготовки навчальних вибірок.

У якості інструменту для роботи з числовими даними та багатовимірними масивами використовується бібліотека NumPy. Вона забезпечує швидке виконання математичних операцій та є базовою складовою більшості бібліотек машинного навчання.

Для оцінювання якості моделей та обчислення метрик застосовується бібліотека Scikit-learn, яка містить готові засоби для розрахунку accuracy, precision, recall, F1-score та побудови матриці помилок. Це дозволяє стандартизувати процес аналізу результатів і забезпечити коректність оцінювання.

У процесі реалізації дослідження також використовується платформа Kaggle, яка виступає джерелом наборів даних та середовищем для їх аналізу. Використання даної платформи дозволяє отримати доступ до великої кількості готових датасетів і забезпечує зручність їх завантаження та структурування.

У якості середовища розробки обрано Visual Studio Code, який забезпечує підтримку Python, інтеграцію з системами керування пакетами та зручні засоби для налагодження програмного коду. Додатковою перевагою є підтримка розширень для роботи з Jupyter Notebook і бібліотеками машинного навчання.

Для навчання моделей передбачається використання графічного процесора, оскільки обчислення у задачах глибокого навчання є ресурсомісткими. Використання GPU дозволяє значно скоротити час навчання моделей та прискорити проведення експериментів.

Також важливим аспектом є використання відкритих і широко поширених технологій, що забезпечує відтворюваність результатів дослідження та можливість подальшого розширення системи. Обраний стек технологій має достатню гнучкість для реалізації всіх необхідних етапів дослідження – від підготовки даних до оцінювання моделей.

Таким чином, вибір інструментів та технологій здійснено з урахуванням вимог задачі, необхідності підтримки сучасних методів глибокого навчання та забезпечення ефективного проведення експериментального дослідження моделей для діагностики захворювань рослин.

3.2 Реалізація обраних моделей глибокого навчання

У межах дослідження передбачається реалізація моделей глибокого навчання, орієнтованих на задачу класифікації зображень захворювань рослин. Основною метою реалізації є проведення порівняльного аналізу ефективності різних архітектур згорткових нейронних мереж та визначення їх придатності для автоматизованої діагностики.

У якості основних моделей було обрано AlexNet та GoogLeNet, які належать до класу згорткових нейронних мереж і широко застосовуються у задачах комп'ютерного зору. Вибір саме цих архітектур обумовлений їх поширеністю, відмінностями у структурі та наявністю успішних прикладів використання у задачах класифікації зображень.

Модель AlexNet використовується як базова архітектура для дослідження. Її структура включає послідовність згорткових, пулінгових та повнозв'язних шарів, які забезпечують поступове виділення ознак із зображень. Реалізація даної моделі дозволяє дослідити ефективність класичного підходу до побудови згорткових нейронних мереж та оцінити його придатність для задачі діагностики захворювань рослин.

Модель GoogLeNet є більш складною архітектурою, що використовує інцепційні модулі для одночасного аналізу ознак різного масштабу. Завдяки цьому модель здатна ефективніше враховувати складні візуальні патерни та формувати більш інформативні представлення вхідних даних. Її реалізація дозволяє порівняти результати базової та більш глибокої архітектур у межах єдиної задачі.

Реалізація моделей здійснюватиметься за допомогою мови програмування Python із використанням бібліотек TensorFlow та Keras. Дані інструменти забезпечують готові засоби для побудови, навчання та оцінювання нейронних мереж, що дозволяє зосередитися на експериментальному аналізі моделей.

На початковому етапі реалізації передбачається створення модулів для завантаження та підготовки даних. Це включає автоматичне зчитування

структури датасету, формування навчальної, валідаційної та тестової вибірок, а також виконання попередньої обробки зображень.

Після підготовки даних буде реалізовано архітектури моделей, включаючи визначення структури шарів, функцій активації та параметрів навчання. Для AlexNet передбачається реалізація класичної структури згорткової мережі, тоді як для GoogLeNet буде використано інцепційні блоки, що дозволяють виконувати паралельний аналіз ознак.

У процесі навчання моделей використовуватимуться оптимізаційні алгоритми та функції втрат, орієнтовані на задачу багатокласової класифікації. Передбачається виконання декількох етапів навчання з фіксацією значень метрик на кожній епосі, що дозволить аналізувати процес збіжності моделей.

Окрему увагу буде приділено реалізації механізмів оцінювання результатів. Для цього передбачається автоматичне обчислення assigasu, precision, recall та F1-score, а також побудова матриць помилок для аналізу класифікації окремих класів.

Результати навчання моделей будуть зберігатися у вигляді файлів ваг, таблиць із метриками та графіків зміни точності й функції втрат. Це дозволить забезпечити відтворюваність результатів та спростити подальший аналіз ефективності моделей.

Також реалізація передбачає створення окремих модулів для навчання та тестування моделей, що забезпечить гнучкість структури проєкту та можливість повторного використання компонентів. Такий підхід дозволяє розділити логіку обробки даних, навчання та оцінювання, підвищуючи зручність подальшого розвитку системи.

На лістингу A.2 наведено реалізацію архітектури моделі AlexNet, яка використовується у дослідженні як базова згорткова нейронна мережа для задачі класифікації захворювань рослин. Реалізація виконана з використанням бібліотеки Keras у середовищі TensorFlow та передбачає побудову моделі у функціональному стилі.

Основою моделі є вхідний шар Input, який визначає форму вхідного зображення. У межах дослідження використовується стандартизований формат

зображень, що дозволяє забезпечити узгодженість даних та коректну роботу згорткових шарів.

Перший згортковий шар містить 96 фільтрів розміром 11×11 із кроком 4. Використання великого ядра на початковому етапі дозволяє виділяти базові ознаки зображення, такі як контури та текстури. Як функція активації використовується ReLU, що забезпечує нелінійність моделі та прискорює процес навчання.

Після першого згорткового шару застосовується Batch Normalization, яка використовується для стабілізації навчання та зменшення внутрішнього зміщення розподілу ознак. Далі виконується MaxPooling, який зменшує просторову розмірність карт ознак та дозволяє виділити найбільш інформативні характеристики.

Другий згортковий шар містить 256 фільтрів розміром 5×5 із padding типу same, що дозволяє зберегти розмірність карт ознак. Після нього також застосовуються Batch Normalization та MaxPooling, що забезпечує стабільність навчання та поступове зменшення обсягу даних.

Наступні три згорткові шари формують глибшу частину мережі та використовують фільтри розміром 3×3 . Кількість фільтрів поступово змінюється, що дозволяє моделі формувати складніші представлення візуальних ознак. Саме ці шари відповідають за виділення високорівневих патернів, характерних для різних захворювань рослин.

Після завершення згорткової частини моделі застосовується ще один шар MaxPooling, який додатково зменшує розмірність карт ознак. Далі дані перетворюються у одновимірний вектор за допомогою шару Flatten, що дозволяє передати їх до повнозв'язної частини мережі.

Повнозв'язна частина моделі складається з двох Dense-шарів по 4096 нейронів кожен. Такі шари виконують інтеграцію виділених ознак та формують фінальне представлення даних перед класифікацією. У якості функції активації також використовується ReLU.

Для зменшення ризику перенавчання між повнозв'язними шарами застосовуються шари Dropout. Вони випадковим чином вимикають частину нейронів під час навчання, що сприяє кращій узагальнювальній здатності моделі.

Фінальний шар моделі є вихідним Dense-шаром із кількістю нейронів, що відповідає кількості класів датасету. У якості функції активації використовується Softmax, яка дозволяє отримати ймовірності належності зображення до кожного класу.

Також у реалізації використовуються різні методи ініціалізації ваг, зокрема `he_normal` для прихованих шарів та `glorot_uniform` для вихідного шару. Це дозволяє покращити процес навчання та стабілізувати збіжність моделі.

У результаті формується модель із назвою AlexNet, яка повертається функцією та надалі використовується для навчання й оцінювання у межах дослідження. Реалізована архітектура дозволяє дослідити ефективність класичної згорткової нейронної мережі у задачі діагностики захворювань рослин.

На лістингу A.3 наведено реалізацію моделі GoogLeNet, яка у межах дослідження реалізована на основі архітектури InceptionV3. Дана модель використовується як більш глибока та сучасна альтернатива AlexNet і дозволяє дослідити ефективність складніших архітектур у задачі класифікації захворювань рослин.

Основою реалізації є використання попередньо навченої моделі InceptionV3, яка входить до складу бібліотеки Keras Applications. Такий підхід дозволяє застосувати transfer learning, тобто перенесення знань, отриманих моделлю під час навчання на великому наборі зображень ImageNet. Це дає змогу суттєво скоротити час навчання та підвищити якість класифікації.

У функції `build_googlenet` передбачено можливість увімкнення або вимкнення transfer learning через параметр `use_transfer_learning`. Якщо цей параметр активний, модель автоматично використовує ваги `imagenet`, що забезпечує попередньо сформовані візуальні ознаки.

Архітектура створюється без верхньої класифікаційної частини (`include_top=False`), оскільки фінальні шари моделі адаптуються під конкретну

задачу дослідження. Це дозволяє використовувати згорткову основу InceptionV3 як екстрактор ознак та формувати власний класифікаційний блок.

На початковому етапі всі шари базової моделі переводяться у режим `trainable = False`. Такий підхід дозволяє зафіксувати вже навчені ознаки та навчати лише нові класифікаційні шари. Це є типовим етапом *transfer learning*, який дозволяє уникнути руйнування попередньо сформованих ваг.

Вхідний шар моделі визначає форму зображення відповідно до стандартизованих параметрів датасету. Далі вхідні дані передаються до базової моделі InceptionV3, яка виконує багаторівневий аналіз візуальних ознак за допомогою інцепційних блоків.

Після проходження згорткової частини використовується шар `GlobalAveragePooling2D`, який виконує глобальне усереднення карт ознак. На відміну від класичного `Flatten`, такий підхід дозволяє суттєво зменшити кількість параметрів та знизити ризик перенавчання.

Далі реалізовано повнозв'язний шар із 1024 нейронами та функцією активації `ReLU`. Цей шар виконує інтеграцію виділених ознак та формує представлення даних перед фінальною класифікацією.

Для боротьби з перенавчанням після повнозв'язного шару застосовується `Dropout`. Він випадковим чином вимикає частину нейронів під час навчання, що сприяє кращій узагальнювальній здатності моделі.

Фінальний вихідний шар реалізовано за допомогою `Dense`-шару з функцією активації `Softmax`. Кількість нейронів відповідає кількості класів датасету, а вихід моделі представляє ймовірності належності зображення до кожного класу захворювання.

Окремо у реалізації наведено функцію `unfreeze_top_layers`, яка використовується для *fine-tuning* моделі. Після завершення початкового етапу *transfer learning* частина верхніх шарів базової моделі розморожується та стає доступною для донавчання на обраному датасеті.

У процесі *fine-tuning* частина шарів залишається замороженою, що дозволяє зберегти базові візуальні ознаки, сформовані на ImageNet. При цьому верхні шари адаптуються під специфіку задачі діагностики захворювань рослин.

Особливу увагу приділено Batch Normalization шарам, які залишаються замороженими навіть після розморожування верхніх шарів. Це дозволяє уникнути нестабільності під час донавчання та забезпечити коректну роботу моделі.

У межах роботи було реалізовано дві архітектури згорткових нейронних мереж – AlexNet та GoogLeNet на основі InceptionV3. Використання двох моделей із різним рівнем складності дозволяє провести порівняльний аналіз ефективності класичних і більш сучасних підходів до класифікації зображень захворювань рослин.

Реалізовані моделі забезпечують повний цикл роботи з даними, включаючи обробку вхідних зображень, формування ознак та класифікацію за визначеними класами. При цьому використання transfer learning та fine-tuning у GoogLeNet дозволяє підвищити ефективність навчання та адаптувати модель до специфіки обраного датасету.

3.3 Налаштування процесу навчання моделей

Налаштування процесу навчання моделей було виконано шляхом визначення єдиної конфігурації експерименту, яка використовується для підготовки вхідних даних, навчання моделей, контролю якості та збереження результатів. Такий підхід дозволяє забезпечити однакові умови для порівняння різних архітектур глибокого навчання та зменшити вплив випадкових факторів на результати дослідження.

На лістингу A.4 наведено основні параметри конфігурації, які використовуються під час навчання моделей AlexNet та GoogLeNet. Винесення таких значень в окремий конфігураційний блок дозволяє централізовано керувати процесом навчання та забезпечує однакові умови для проведення експериментів.

Параметри IMAGE_SIZE та INPUT_SHAPE визначають розмір вхідних зображень і форму тензора, що подається на вхід моделі. У роботі

використовується розмір 224×224 пікселі та триканальне RGB-представлення, що забезпечує сумісність із реалізованими архітектурами.

Параметр `BATCH_SIZE` встановлено на рівні 32, що дозволяє обробляти зображення групами та забезпечує баланс між швидкістю навчання і використанням пам'яті. Кількість епох `EPOCHS` дорівнює 30, що визначає максимальну кількість проходів моделі по навчальній вибірці.

Швидкість навчання `LEARNING_RATE` має значення $1e-3$ і використовується як базовий параметр оптимізатора. Для забезпечення відтворюваності результатів задано `RANDOM_STATE = 42`, що дозволяє фіксувати випадкові процеси під час підготовки даних і навчання.

Параметри `ALEXNET_DROPOUT` та `GOOGLNET_DROPOUT` відповідають за регуляризацію моделей. Dropout використовується для зменшення ризику перенавчання шляхом випадкового вимкнення частини нейронів під час навчання.

Окремо задано параметри `callback`-механізмів. `EARLY_STOPPING_PATIENCE` визначає кількість епох без покращення результату, після яких навчання може бути зупинене. Параметри `REDUCE_LR_PATIENCE`, `REDUCE_LR_FACTOR` та `MIN_LR` використовуються для адаптивного зменшення швидкості навчання у випадку, якщо модель перестає покращувати значення функції втрат.

Для GoogLeNet окремо визначено параметри transfer learning і fine-tuning. Значення `GOOGLNET_FINE_TUNE_LAYERS = 50` задає кількість верхніх шарів базової моделі, які будуть розморожені для донавчання. Параметр `GOOGLNET_PHASE1_EPOCHS = 10` визначає тривалість першого етапу навчання, а `GOOGLNET_FINE_TUNE_LR = 1e-4` задає зменшену швидкість навчання для етапу тонкого налаштування.

На лістингу A.5 наведено реалізацію функції `build_callbacks`, яка формує набір `callback`-механізмів для автоматизованого контролю процесу навчання моделей глибокого навчання. Використання `callback`-функцій дозволяє підвищити стабільність навчання, автоматизувати збереження результатів та запобігти перенавчанню моделей.

Функція приймає назву моделі, директорію для збереження моделей та директорію для результатів навчання. На основі цих параметрів формується список `callback-об'єктів`, які використовуються під час виклику методу `model.fit()`.

Першим `callback-механізмом` є `ModelCheckpoint`, який відповідає за автоматичне збереження найкращої версії моделі під час навчання. У якості критерію оцінювання використовується метрика `val_acc`, тобто точність на валідаційній вибірці. Параметр `save_best_only=True` забезпечує збереження лише тієї моделі, яка показала найкращий результат, що дозволяє уникнути накопичення проміжних версій.

Файли моделей зберігаються у форматі `.keras`, що забезпечує можливість подальшого завантаження моделі разом із її архітектурою та вагами. Такий підхід спрощує подальше тестування та повторне використання моделей.

Другим `callback-механізмом` є `EarlyStopping`, який використовується для автоматичного припинення навчання у випадку відсутності покращення результатів. У даному випадку контроль також здійснюється за значенням `val_acc`.

Параметр `patience=7` означає, що навчання буде зупинене, якщо протягом семи епох не спостерігається покращення точності на валідаційній вибірці. Це дозволяє запобігти перенавчанню та уникнути зайвих обчислень після досягнення моделлю оптимального стану.

Параметр `restore_best_weights=True` забезпечує автоматичне повернення ваг моделі до найкращого зафіксованого стану після завершення навчання. Завдяки цьому модель не використовує менш ефективні ваги, отримані на останніх етапах навчання.

Наступним `callback-механізмом` є `ReduceLROnPlateau`, який реалізує адаптивне зменшення швидкості навчання. У даному випадку моніторинг виконується за значенням `val_loss`, тобто функції втрат на валідаційній вибірці.

Якщо значення `val_loss` не покращується протягом трьох епох (`patience=3`), швидкість навчання автоматично зменшується у два рази (`factor=0.5`). Це

дозволяє моделі більш обережно оновлювати ваги після досягнення етапу повільної збіжності.

Параметр `min_lr=1e-7` визначає мінімально допустиме значення швидкості навчання. Це запобігає надмірному зменшенню `learning rate`, яке могло б зробити подальше навчання неефективним.

Останнім `callback`-механізмом є `CSVLogger`, який використовується для автоматичного збереження журналу навчання у CSV-файл. У файл записуються значення метрик та функції втрат для кожної епохи, що дозволяє виконувати подальший аналіз результатів та будувати графіки навчання.

Використання окремої функції для формування `callback`-механізмів дозволяє централізувати їх налаштування та повторно використовувати однакову конфігурацію для різних моделей. Це спрощує підтримку коду та забезпечує узгодженість процесу навчання у межах дослідження.

На лістингу A.6 наведено процес компіляції та навчання моделі AlexNet для задачі класифікації захворювань рослин. Даний етап є завершальною частиною підготовки моделі перед проведенням експериментального дослідження.

На початковому етапі виконується створення моделі за допомогою функції `build_alexnet`. У параметр `num_classes` передається кількість класів датасету, що дозволяє автоматично сформувати вихідний шар із відповідною кількістю нейронів. Завдяки цьому модель адаптується до структури обраного набору даних.

Після створення моделі виконується її компіляція за допомогою методу `compile`. На цьому етапі визначаються алгоритм оптимізації, функція втрат та метрики оцінювання, які використовуватимуться під час навчання.

У якості оптимізатора використовується Adam із початковою швидкістю навчання `1e-3`. Даний оптимізатор поєднує переваги адаптивного налаштування швидкості навчання та моментного оновлення ваг, що забезпечує стабільну збіжність моделі навіть у складних задачах класифікації.

Функцією втрат обрано `sparse_categorical_crossentropy`, яка використовується для задач багатокласової класифікації у випадках, коли мітки

класів представлені у вигляді цілих чисел, а не one-hot-векторів. Такий підхід дозволяє спростити структуру даних та зменшити обсяг необхідної пам'яті.

У якості основної метрики оцінювання використовується ассурасу, яка визначає частку правильно класифікованих зображень. Під час навчання дана метрика обчислюється як для навчальної, так і для валідаційної вибірки, що дозволяє контролювати процес збіжності моделі.

Навчання моделі виконується за допомогою методу `fit`, який запускає повний цикл ітеративного оновлення ваг на основі навчального набору даних `train_ds`. У процесі навчання модель поступово мінімізує функцію втрат та формує внутрішні представлення візуальних ознак захворювань рослин.

Для контролю якості навчання використовується валідаційний набір `val_ds`. Після кожної епохи модель оцінюється на цих даних, що дозволяє відстежувати зміну якості класифікації та виявляти можливі ознаки перенавчання.

Кількість епох навчання встановлено на рівні 30. Це визначає максимальну кількість проходів по навчальній вибірці, однак фактична кількість епох може бути меншою через використання механізму `EarlyStopping`.

У процесі навчання також використовуються `callback`-механізми, наведені на лістингу A.5. Вони забезпечують автоматичне збереження найкращої моделі, адаптивне зменшення швидкості навчання, фіксацію журналу навчання та дострокове завершення процесу у випадку відсутності покращення результатів.

Результат роботи методу `fit` зберігається у змінній `history`, яка містить історію навчання моделі. До цієї структури входять значення ассурасу, `val_acc`, `loss` та `val_loss` для кожної епохи. Надалі ці дані використовуються для побудови графіків та аналізу процесу навчання.

На лістингу A.7 наведено процес навчання моделі `GoogLeNet`, реалізованої на основі архітектури `InceptionV3` із використанням `transfer learning` та `fine-tuning`. На відміну від `AlexNet`, навчання даної моделі виконується у два окремі етапи, що дозволяє ефективно адаптувати попередньо навчену модель до задачі діагностики захворювань рослин.

На першому етапі виконується створення моделі за допомогою функції `build_googlenet`. У параметр `num_classes` передається кількість класів датасету, а параметр `use_transfer_learning=True` вказує на необхідність використання попередньо навчених ваг ImageNet.

У результаті створюються дві сутності: повна модель `model` та базова згорткова частина `base_model`, яка відповідає за виділення візуальних ознак. На цьому етапі базова модель залишається замороженою, тобто її ваги не оновлюються під час навчання.

Такий підхід дозволяє використовувати вже сформовані ознаки, отримані під час навчання на великому наборі зображень ImageNet. Завдяки цьому модель може швидше адаптуватися до нової задачі та демонструвати кращі результати навіть при обмеженому обсязі навчальних даних.

Після створення моделі виконується її компіляція. Для оптимізації використовується Adam із швидкістю навчання $1e-3$, а у якості функції втрат застосовується `sparse_categorical_crossentropy`. Основною метрикою оцінювання є `accuracy`.

Навчання на першому етапі виконується за допомогою методу `fit` протягом 10 епох. У цей період оновлюються лише ваги класифікаційної частини моделі, тоді як згорткова основа InceptionV3 використовується як фіксований екстрактор ознак.

Після завершення першого етапу виконується перехід до `fine-tuning`. Для цього викликається функція `unfreeze_top_layers`, яка розморожує верхні 50 шарів базової моделі InceptionV3. Це дозволяє частково адаптувати попередньо навчені ознаки до специфіки зображень захворювань рослин.

Під час `fine-tuning` модель повторно компілюється. При цьому використовується значно менша швидкість навчання $1e-4$, оскільки надто великі оновлення ваг можуть негативно вплинути на вже сформовані ознаки. Зменшення `learning rate` забезпечує більш плавне та контрольоване донавчання.

На другому етапі модель продовжує навчання ще протягом 20 епох. У цей період оновлюються не лише класифікаційні шари, але й верхня частина

згорткової основи, що дозволяє моделі краще адаптуватися до структури конкретного датасету.

Використання двоетапного навчання є однією з ключових особливостей transfer learning. На першому етапі модель навчається працювати з новими класами, а на другому – виконується точне налаштування високорівневих ознак для покращення результатів класифікації.

Результати кожного етапу навчання зберігаються окремо у змінних history1 та history2. Це дозволяє аналізувати ефективність transfer learning та fine-tuning окремо, а також оцінювати зміну якості моделі після розморожування верхніх шарів.

У межах роботи було сформовано та реалізовано єдину конфігурацію процесу навчання моделей глибокого навчання, що забезпечує стабільність, відтворюваність та коректність проведення експериментів. Визначено основні гіперпараметри навчання, параметри регуляризації та механізми автоматизованого контролю процесу навчання.

Для моделей AlexNet та GoogLeNet використано однакові базові умови навчання, що дозволяє здійснювати об'єктивний порівняльний аналіз їх ефективності. При цьому для GoogLeNet додатково реалізовано підхід transfer learning та fine-tuning, який забезпечує адаптацію попередньо навчених ваг до задачі діагностики захворювань рослин.

Реалізовані callback-механізми дозволяють автоматично зберігати найкращі версії моделей, адаптивно змінювати швидкість навчання та запобігати перенавчанню. Це підвищує стабільність процесу навчання та спрощує подальший аналіз результатів.

3.4 Розробка консольного програмного засобу для тестування моделей

Розроблюваний консольний програмний засіб призначений для тестування навчених моделей глибокого навчання на окремих зображеннях рослин. Основною задачею застосунку є автоматизоване виконання повного циклу

прогнозування – від завантаження зображення до виведення результату класифікації у зрозумілому для користувача вигляді.

Робота застосунку починається з приймання вхідних параметрів із командного рядка. Користувач повинен вказати шлях до зображення рослини та назву моделі, яка буде використовуватись для класифікації. Такий підхід дозволяє використовувати один програмний засіб для тестування різних архітектур без необхідності зміни програмного коду.

Після отримання параметрів застосунок виконує перевірку коректності вхідних даних. Зокрема, перевіряється існування файлу зображення, правильність назви моделі та наявність відповідного файлу ваг у директорії `models/`. Це дозволяє запобігти помилкам під час виконання та забезпечує стабільність роботи програмного засобу.

Наступним етапом є завантаження обраної моделі глибокого навчання. Для цього використовується механізм завантаження моделей TensorFlow/Keras, який дозволяє відновити архітектуру нейронної мережі разом із навченими вагами. Залежно від параметра, переданого користувачем, застосунок може працювати як з AlexNet, так і з GoogLeNet.

Після завантаження моделі виконується попередня обробка зображення. На цьому етапі зображення відкривається та приводиться до стандартизованого формату, що відповідає вимогам моделей. Зокрема, виконується зміна розміру до 224×224 пікселів, що забезпечує сумісність із вхідним шаром нейронної мережі.

Додатково здійснюється перетворення зображення у формат RGB. Це є необхідним, оскільки моделі були навчені на триканальних кольорових зображеннях, і використання іншого формату могло б призвести до некоректної роботи нейронної мережі.

Після цього виконується нормалізація значень пікселів. Значення кольорових каналів переводяться у стандартизований діапазон, що забезпечує коректну роботу моделі та відповідає умовам, у яких виконувалося її навчання.

Підготовлене зображення перетворюється у формат тензора та подається на вхід нейронної мережі. У результаті модель формує вектор ймовірностей, де

кожне значення відповідає ймовірності належності зображення до певного класу захворювання.

На основі отриманого вектора визначається клас із найбільшою ймовірністю, який вважається фінальним результатом класифікації. Додатково застосунок обчислює та відображає значення цієї ймовірності, що дозволяє оцінити впевненість моделі у власному прогнозі.

Для більш детального аналізу результатів також реалізовано виведення трьох найбільш ймовірних класів. Такий підхід дозволяє користувачу оцінити альтернативні варіанти класифікації та проаналізувати, наскільки близькими є результати прогнозування для різних захворювань.

Отримані результати виводяться у консольному режимі у зрозумілому текстовому форматі, що забезпечує простоту використання застосунку та дозволяє швидко тестувати різні зображення без необхідності використання графічного інтерфейсу.

На лістингу A.1 наведено реалізацію консольного програмного засобу `predict.py`, призначеного для тестування навчених моделей глибокого навчання на окремому зображенні рослини. Застосунок забезпечує повний цикл передбачення: від приймання параметрів командного рядка до виведення результатів класифікації у консольному форматі.

На початку програми задається опис призначення файлу та приклади його використання. Застосунок може запускатися з вибором конкретної моделі, передаванням шляху до зображення, а також із додатковим параметром `top`, який визначає кількість найбільш ймовірних класів для відображення. Це робить засіб гнучким для проведення тестування різних моделей та зображень.

Окремо реалізовано фіксацію робочої директорії до директорії скрипта. Це дозволяє уникнути помилок із відносними шляхами, особливо під час запуску програми у середовищі Windows або з каталогів, що містять кириличні символи. Завдяки цьому директорії `models`, `processed_data` та інші шляхи коректно визначаються незалежно від місця запуску скрипта.

Функція `preprocess_image` відповідає за підготовку вхідного зображення до подання в модель. Вона відкриває файл, переводить його у формат RGB, змінює

розмір до значення 224×224 пікселі, нормалізує значення пікселів до діапазону $[0; 1]$ та додає batch-вимір. У результаті формується тензор розмірності (1, H, W, 3), який відповідає формату вхідних даних моделей.

Функція `load_model` реалізує завантаження навченої Keras-моделі з файлу. Якщо користувач явно не вказує шлях до ваг, програма автоматично шукає найкращий checkpoint моделі у директорії `models`, а за його відсутності – фінальну версію моделі. Якщо файл не знайдено, застосунок виводить повідомлення про помилку та пропонує спочатку виконати навчання відповідної моделі.

Для коректного зіставлення індексів прогнозу з назвами класів використовується функція `load_class_names`. Вона спочатку намагається зчитати список класів із файлу `processed_data/train.csv`, що забезпечує стабільність порядку класів відповідно до процесу навчання. Якщо такий файл відсутній, програма використовує резервний варіант – сканування директорії з обробленими зображеннями.

Функція `print_results` відповідає за форматоване виведення результатів передбачення. Вона визначає клас із найбільшою ймовірністю, виводить його назву та відсоткове значення впевненості моделі. Додатково відображається перелік top-K найбільш імовірних класів із текстовою шкалою, що робить результат більш наочним.

У функції `parse_args` реалізовано обробку аргументів командного рядка. Користувач повинен передати шлях до зображення через параметр `--image` та назву моделі через параметр `--model`. Також передбачено необов'язкові параметри `--weights` для явного зазначення файлу моделі та `--top` для вибору кількості альтернативних результатів.

Основна логіка роботи програми зосереджена у функції `main`. Вона виконує перевірку наявності зображення, контроль підтримуваного формату файлу, завантаження списку класів, завантаження моделі, попередню обробку зображення та виконання передбачення. Після цього отриманий вектор ймовірностей передається до функції виведення результатів.

Під час виконання інференсу модель повертає масив ймовірностей для всіх класів датасету. Найбільше значення у цьому масиві відповідає основному прогнозу моделі, тоді як інші значення дозволяють оцінити альтернативні варіанти класифікації. Це є корисним для аналізу випадків, коли модель має схожу впевненість щодо кількох класів.

На рисунку 3.1 наведено приклад роботи розробленого консольного програмного засобу під час виконання передбачення для зображення рослини. У даному випадку для класифікації використовується модель GoogLeNet, попередньо навчена на наборі даних захворювань рослин.

```
Завантаження моделі: models\googlenet_best.keras
Обробка зображення: Tomato___late_blight.jpg
Виконання передбачення...
```

```
Зображення : Tomato___late_blight.jpg
Модель      : googlenet
```

```
ПЕРЕДБАЧЕНИЙ КЛАС : Tomato___late_blight
ЙМОВІРНІСТЬ       : 97.83%
```

```
ТОП-3 РЕЗУЛЬТАТИ:
```

1. Tomato___late_blight	97.83%	
2. Tomato___early_blight	1.42%	
3. Tomato___healthy	0.51%	

Рисунок 3.1 – Результат виводу консолі

На початковому етапі застосунок виконує завантаження моделі з директорії models. Після цього відображається інформація про обробку вибраного зображення, що дозволяє користувачу контролювати процес виконання програми та переконатися у правильності переданих параметрів.

Далі застосунок виконує передбачення та виводить результат класифікації у текстовому форматі. У наведеному прикладі модель визначила, що зображення належить до класу Tomato___late_blight, який відповідає фітофторозу томатів. Імовірність передбачення становить 97.83%, що свідчить про високий рівень впевненості моделі у результаті класифікації.

Окрім основного результату, застосунок також виводить три найбільш імовірні класи. Це дозволяє оцінити альтернативні варіанти класифікації та проаналізувати, наскільки близькими є результати моделі для різних захворювань. У наведеному прикладі друге та третє місце займають класи Tomato___early_blight та Tomato___healthy, однак їх імовірності є значно нижчими.

Для покращення наочності результатів реалізовано текстову шкалу, яка візуально відображає рівень впевненості моделі для кожного класу. Це дозволяє швидко оцінити співвідношення між основним прогнозом та альтернативними варіантами.

У межах роботи було розроблено консольний програмний засіб для тестування навчених моделей глибокого навчання на окремих зображеннях рослин. Застосунок забезпечує автоматизоване виконання всіх основних етапів прогнозування, включаючи завантаження моделі, попередню обробку зображення, виконання класифікації та виведення результатів.

Реалізований підхід дозволяє використовувати один програмний засіб для роботи з різними архітектурами моделей, а також забезпечує зручне тестування зображень без необхідності використання графічного інтерфейсу. Додаткове відображення top-K результатів та ймовірностей класифікації покращує інтерпретацію отриманих прогнозів.

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ МОДЕЛЕЙ

4.1 Постановка експерименту

У межах роботи експериментальне дослідження спрямоване на оцінювання ефективності моделей глибокого навчання у задачі автоматизованої діагностики захворювань рослин за зображеннями листя. Основною метою експерименту є порівняння архітектур AlexNet та GoogLeNet, визначення їх сильних і слабких сторін, а також аналіз впливу transfer learning на якість класифікації.

На першому етапі експерименту виконується підготовка набору даних. Для цього використовується розширений датасет на основі PlantVillage, що містить зображення різних культур та типів захворювань. Усі зображення проходять попередню обробку, яка включає зміну розміру до 224×224 пікселів, переведення у формат RGB та нормалізацію значень пікселів.

Після підготовки даних виконується розподіл датасету на навчальну, валідаційну та тестову вибірки. Навчальна вибірка використовується для оновлення ваг моделей, валідаційна – для контролю процесу навчання та налаштування параметрів, а тестова – для фінального оцінювання якості класифікації.

Наступним етапом є навчання моделі AlexNet. Дана модель використовується як базова архітектура згорткової нейронної мережі. Її навчання виконується протягом визначеної кількості епох із використанням оптимізатора Adam та функції втрат sparse categorical crossentropy. Під час навчання фіксуються значення accuracy та loss для навчальної і валідаційної вибірок.

Після завершення навчання AlexNet виконується оцінювання моделі на тестовому наборі даних. На цьому етапі обчислюються основні метрики якості, зокрема accuracy, precision, recall та F1-score. Також формується матриця помилок для аналізу правильності класифікації окремих класів захворювань.

Далі проводиться експеримент із моделлю GoogLeNet на основі InceptionV3. Навчання даної моделі організовується у два етапи. На першому етапі використовується transfer learning із замороженою базовою моделлю та навчанням лише класифікаційної частини мережі.

Після завершення початкового етапу виконується fine-tuning верхніх шарів InceptionV3. Для цього розморожуються верхні 50 шарів моделі, після чого виконується додаткове навчання зі зменшеною швидкістю навчання. Такий підхід дозволяє адаптувати попередньо навчені ознаки до специфіки зображень захворювань рослин.

У процесі навчання обох моделей використовуються callback-механізми, які забезпечують автоматичне збереження найкращих ваг, дострокове завершення навчання у випадку відсутності покращення результатів та адаптивне зменшення швидкості навчання. Це дозволяє забезпечити стабільність експерименту та запобігти перенавчанню моделей.

Окрему увагу приділено фіксації історії навчання моделей. Для кожної архітектури зберігаються значення accuracy, val_accuracy, loss та val_loss для кожної епохи. На основі цих даних у подальшому будуть побудовані графіки навчання, які дозволяють аналізувати процес збіжності моделей.

Після завершення навчання та тестування обох моделей виконується їх порівняльний аналіз. Порівнюються значення основних метрик якості, стабільність навчання, швидкість збіжності та здатність моделей правильно класифікувати окремі типи захворювань.

Додатково у межах експерименту проводиться тестування моделей за допомогою розробленого консольного застосунку. Для цього використовуються окремі зображення рослин, які подаються на вхід моделі з подальшим аналізом отриманих прогнозів та рівня впевненості класифікації.

Результати експерименту будуть представлені у вигляді таблиць, графіків та матриць помилок. Це дозволить виконати детальний аналіз ефективності моделей та визначити найбільш придатний підхід для задачі автоматизованої діагностики захворювань рослин.

4.2 Проведення навчання моделей

У межах даного етапу дослідження буде проведено навчання обох реалізованих моделей глибокого навчання – AlexNet та GoogLeNet на основі InceptionV3. Основною метою цього етапу є формування навчених моделей, здатних виконувати класифікацію захворювань рослин за візуальними ознаками листя.

Перед початком навчання виконується підготовка навчальних, валідаційних та тестових вибірок, а також завантаження попередньо оброблених зображень у форматі, придатному для використання у TensorFlow/Keras. Для обох моделей використовуються однакові набори даних, що забезпечує коректність подальшого порівняння результатів.

Навчання моделі AlexNet буде виконуватись у класичному режимі з випадковою ініціалізацією ваг. У процесі навчання модель поступово формуватиме власні представлення візуальних ознак на основі зображень захворювань рослин. Під час кожної епохи здійснюватиметься оновлення ваг нейронної мережі з використанням алгоритму оптимізації Adam.

Для моделі GoogLeNet процес навчання буде реалізовано у два етапи. На першому етапі використовуватиметься transfer learning із попередньо навченими вагами ImageNet та замороженою базовою моделлю InceptionV3. У цей період навчатиметься лише класифікаційна частина мережі.

Після завершення початкового етапу буде проведено fine-tuning верхніх шарів GoogLeNet. Для цього частина шарів InceptionV3 буде розморожена та адаптована до специфіки датасету захворювань рослин. Навчання на цьому етапі виконуватиметься зі зменшеною швидкістю навчання для забезпечення стабільного оновлення ваг.

У процесі навчання для обох моделей буде здійснюватися автоматичний контроль якості за допомогою callback-механізмів. Зокрема, використовуватиметься рання зупинка навчання, автоматичне збереження найкращої версії моделі та адаптивне зменшення швидкості навчання у випадку відсутності покращення результатів.

На кожній епосі буде фіксуватися значення ассигасу та функції втрат як для навчальної, так і для валідаційної вибірок. Це дозволить у подальшому виконати аналіз процесу збіжності моделей та визначити наявність перенавчання або недонавчання.

Результати навчання будуть збережені у вигляді файлів моделей, журналів навчання та графіків зміни метрик. Це забезпечить можливість подальшого тестування моделей, аналізу результатів та повторного використання отриманих ваг.

На рисунку 4.1 наведено фрагмент процесу навчання моделей глибокого навчання у середовищі TensorFlow/Keras. У консолі відображаються основні параметри навчання для кожної епохи, зокрема значення функції втрат, точності класифікації та швидкість виконання обчислень.

```
Epoch 07: ReduceLROnPlateau reducing learning rate to 5.0000002374872565e-05.  
  
Epoch 8/20  
2542/2542 [=====] - 95s 37ms/step - loss: 0.0834 - accuracy: 0.9729 - val_loss: 0.2891 - val_accuracy: 0.9162  
Epoch 9/20  
2542/2542 [=====] - 95s 37ms/step - loss: 0.0763 - accuracy: 0.9754 - val_loss: 0.2874 - val_accuracy: 0.9171  
Epoch 10/20  
2542/2542 [=====] - 95s 37ms/step - loss: 0.0712 - accuracy: 0.9771 - val_loss: 0.2881 - val_accuracy: 0.9168  
Epoch 11/20  
2542/2542 [=====] - 95s 37ms/step - loss: 0.0674 - accuracy: 0.9784 - val_loss: 0.2893 - val_accuracy: 0.9164
```

Рисунок 4.1 – Фрагмент процесу навчання моделей глибокого навчання

У процесі навчання система автоматично виводить інформацію про поточну епоху, кількість оброблених батчів та час виконання одного кроку навчання. Це дозволяє контролювати швидкість навчання моделі та оцінювати ефективність використання обчислювальних ресурсів.

На рисунку також продемонстровано роботу callback-механізму ReduceLROnPlateau, який автоматично зменшує швидкість навчання у випадку відсутності покращення результатів. У наведеному прикладі learning rate було зменшено до значення $5e-05$, що дозволяє моделі більш обережно оновлювати ваги на пізніх етапах навчання.

Для кожної епохи відображається значення loss та ассигасу для навчальної вибірки. Зменшення функції втрат і одночасне зростання точності свідчать про поступове покращення якості моделі та стабільний процес збіжності.

Окремо відображаються показники `val_loss` та `val_accuracy`, які характеризують якість моделі на валідаційній вибірці. Саме ці параметри використовуються для контролю перенавчання та оцінювання здатності моделі узагальнювати знання на нових даних.

У наведеному прикладі можна спостерігати, що точність на навчальній вибірці перевищує 97%, тоді як точність на валідаційній вибірці перевищує 91%. Це свідчить про достатньо високу ефективність моделі та здатність коректно класифікувати зображення захворювань рослин.

Також помітно, що значення `val_accuracy` змінюється незначно між епохами, що може свідчити про наближення моделі до етапу стабілізації навчання. У таких умовах використання механізму адаптивного зменшення швидкості навчання є доцільним для покращення точності моделі.

На рисунку 4.2 наведено результати навчання моделі AlexNet у вигляді графіків зміни `accuracy` та `loss` протягом епох навчання. Побудовані графіки дозволяють проаналізувати процес збіжності моделі, оцінити стабільність навчання та визначити наявність ознак перенавчання.

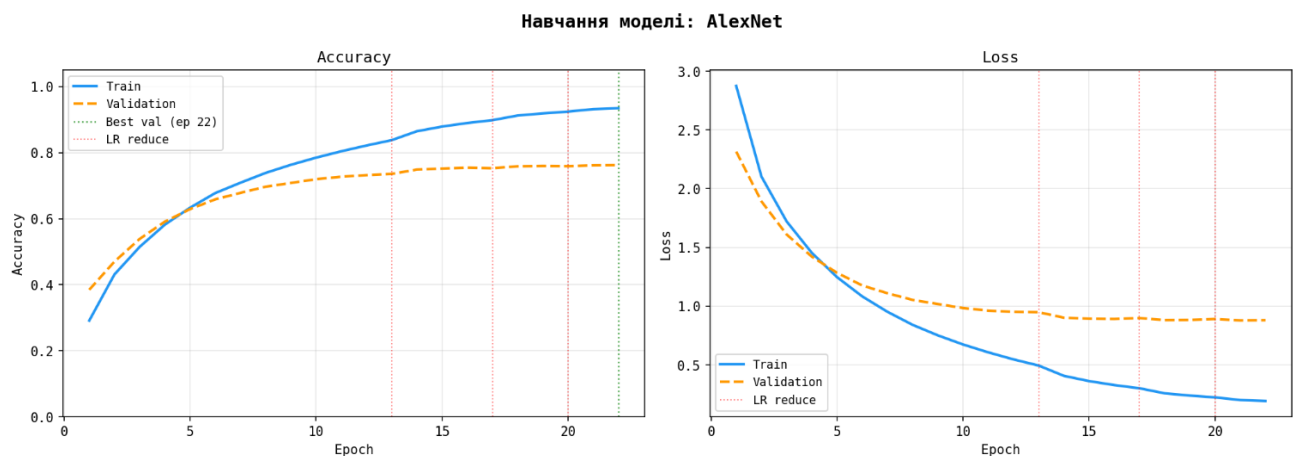


Рисунок 4.2 – Результат навчання моделі AlexNet

Ліва частина рисунка демонструє зміну точності класифікації для навчальної та валідаційної вибірок. Можна спостерігати поступове зростання `accuracy` протягом усього процесу навчання. Значення точності на навчальній вибірці досягає понад 93%, що свідчить про ефективне засвоєння моделлю ознак захворювань рослин.

Точність на валідаційній вибірці також демонструє стабільне зростання, однак її значення є нижчим порівняно з навчальною вибіркою. Максимальне значення validation accuracy було отримано на 22-й епосі, що позначено вертикальною зеленою лінією на графіку.

Різниця між train accuracy та validation accuracy свідчить про наявність помірного перенавчання моделі. Після приблизно 12–14 епох темп покращення точності на валідаційній вибірці значно сповільнюється, тоді як точність на навчальних даних продовжує зростати.

На графіку також відображені моменти автоматичного зменшення швидкості навчання, позначені вертикальними червоними пунктирними лініями. Застосування механізму ReduceLROnPlateau дозволило стабілізувати процес навчання та запобігти різким коливанням функції втрат.

Права частина рисунка демонструє зміну функції втрат для навчальної та валідаційної вибірок. Значення train loss поступово зменшується протягом усього навчання, що свідчить про коректну збіжність моделі.

Validation loss також зменшується на початкових етапах навчання, однак після приблизно 12-ї епохи стабілізується та змінюється незначно. Це підтверджує, що модель досягає межі узагальнювальної здатності та починає частково перенавчатися на навчальній вибірці.

Попри наявність помірного перенавчання, модель демонструє достатньо високі результати класифікації та стабільний процес навчання. Використання Dropout, EarlyStopping та адаптивного зменшення learning rate дозволило уникнути критичного погіршення якості на валідаційних даних.

Отримані результати свідчать про те, що архітектура AlexNet здатна ефективно виконувати задачу класифікації захворювань рослин, однак її узагальнювальна здатність є дещо обмеженою порівняно з сучаснішими глибокими архітектурами.

На рисунку 4.3 наведено результати навчання моделі GoogLeNet на основі архітектури InceptionV3. Графіки відображають зміну accuracy та loss для навчальної та валідаційної вибірок протягом процесу навчання, а також демонструють вплив transfer learning і fine-tuning на якість класифікації.

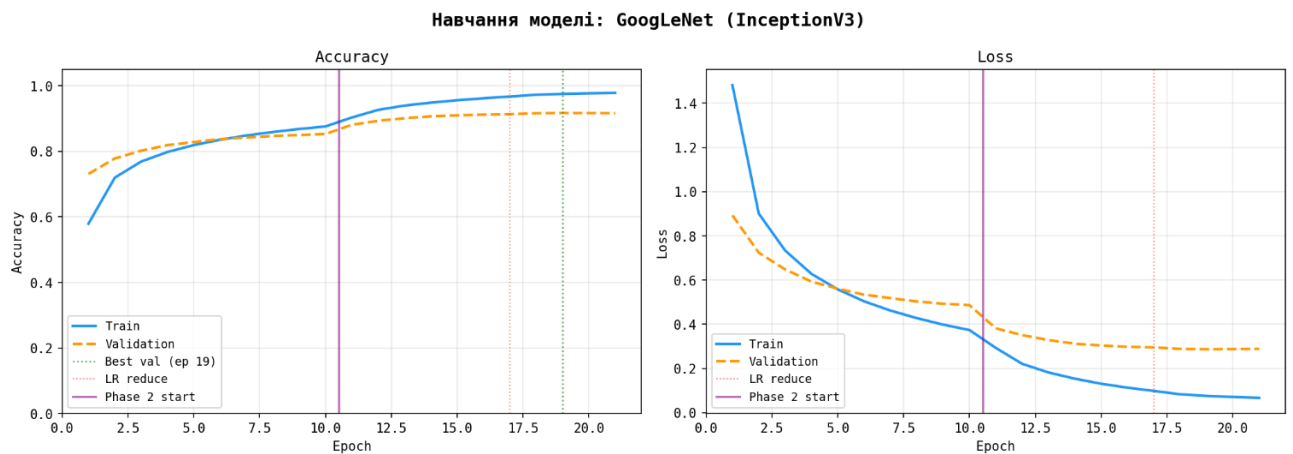


Рисунок 4.3 – Результат навчання GoogLeNet

Ліва частина рисунка демонструє зміну точності класифікації. Уже на початкових етапах навчання модель показує достатньо високі значення validation accuracy, що пояснюється використанням попередньо навчених ваг ImageNet. Це дозволяє моделі ефективно використовувати вже сформовані візуальні ознаки та швидше адаптуватися до нової задачі.

На 10-й епосі позначено початок другого етапу навчання – fine-tuning, що відображено вертикальною фіолетовою лінією. Після цього відбувається додаткове покращення якості класифікації як на навчальній, так і на валідаційній вибірках.

Точність на навчальній вибірці поступово зростає майже до 98%, що свідчить про ефективне засвоєння моделлю ознак захворювань рослин. Водночас validation accuracy також демонструє стабільне покращення та перевищує 91%, що є вищим результатом порівняно з AlexNet.

Найкраще значення validation accuracy було отримано на 19-й епосі, що позначено вертикальною зеленою пунктирною лінією. Це свідчить про те, що модель досягає оптимального стану після завершення fine-tuning верхніх шарів InceptionV3.

Порівняно з AlexNet, різниця між train accuracy та validation accuracy є менш вираженою. Це вказує на кращу узагальнювальну здатність GoogLeNet та меншу схильність до перенавчання.

Права частина рисунка демонструє зміну функції втрат. Значення train loss швидко зменшується протягом усього процесу навчання, що свідчить про стабільну збіжність моделі та коректну роботу оптимізатора.

Validation loss також поступово зменшується та демонструє стабільніші результати порівняно з AlexNet. Після початку fine-tuning можна спостерігати додаткове покращення функції втрат, що підтверджує ефективність адаптації верхніх шарів InceptionV3 до специфіки датасету.

На графіку також відображено момент автоматичного зменшення швидкості навчання, позначений червоною пунктирною лінією. Застосування механізму ReduceLROnPlateau дозволило стабілізувати завершальний етап навчання та уникнути коливань значення loss.

Отримані результати демонструють, що використання transfer learning та fine-tuning суттєво покращує якість класифікації. GoogLeNet забезпечує вищу точність та кращу узагальнювальну здатність порівняно з AlexNet, що підтверджує ефективність сучасних глибоких архітектур для задачі діагностики захворювань рослин.

У межах дослідження було проведено навчання моделей AlexNet та GoogLeNet для задачі класифікації захворювань рослин за зображеннями листя. У процесі навчання обидві моделі продемонстрували стабільну збіжність та здатність формувати інформативні візуальні ознаки для подальшої класифікації.

Результати навчання показали, що модель GoogLeNet на основі InceptionV3 забезпечує вищу точність класифікації та кращу узагальнювальну здатність порівняно з AlexNet. Використання transfer learning та fine-tuning дозволило суттєво покращити якість моделі та прискорити процес її адаптації до датасету захворювань рослин.

Також у процесі навчання було підтверджено ефективність використання callback-механізмів, регуляризації та адаптивного керування швидкістю навчання, що дозволило забезпечити стабільність експерименту та зменшити ризик перенавчання моделей.

4.3 Оцінювання якості моделей за обраними метриками

У даному підрозділі буде виконано оцінювання якості навчених моделей глибокого навчання за допомогою обраних метрик класифікації. Основною метою цього етапу є визначення ефективності моделей AlexNet та GoogLeNet у задачі автоматизованої діагностики захворювань рослин, а також проведення їх порівняльного аналізу.

Оцінювання буде виконуватися на тестовій вибірці, яка не використовувалась під час навчання моделей. Такий підхід дозволяє об'єктивно оцінити здатність моделей узагальнювати знання та виконувати класифікацію нових зображень.

У межах дослідження буде проаналізовано значення accuracy, precision, recall та F1-score. Для кожної моделі будуть визначені як macro-, так і weighted-варіанти метрик, що дозволяє врахувати особливості розподілу класів у датасеті та оцінити якість класифікації як у загальному випадку, так і з урахуванням ваг окремих класів.

Також буде проаналізовано швидкість виконання інференсу моделей, час обробки тестового набору та продуктивність у вигляді кількості оброблених зображень за секунду. Це дозволить оцінити не лише точність моделей, але й ефективність їх практичного використання.

Отримані результати будуть представлені у вигляді текстових звітів, таблиць та графічних матеріалів, що дозволить провести детальний аналіз сильних і слабких сторін кожної архітектури.

На рисунку 4.4 наведено приклад результатів оцінювання моделі AlexNet на тестовому наборі даних. У консолі відображаються основні етапи виконання оцінювання, зокрема завантаження моделі, запуск інференсу та завершення обробки тестового набору.

```
[2026-05-14 09:12:04] INFO evaluate - Loading model: AlexNet
[2026-05-14 09:12:04] INFO evaluate - Model path: models/alexnet_best.keras
[2026-05-14 09:12:08] INFO evaluate - Model loaded (47.3M parameters)
[2026-05-14 09:12:08] INFO evaluate - Running inference on test set...
[2026-05-14 09:16:51] INFO evaluate - Inference complete in 283.1 s (~61.5 samples/sec)

Accuracy          : 0.7589
Precision (macro) : 0.7596
Recall (macro)    : 0.7645
F1-score (macro)  : 0.7597
Precision (weighted): 0.7531
Recall (weighted): 0.7610
F1-score (weighted): 0.7548
```

Рисунок 4.4 – Результати оцінювання моделі AlexNet

Після завершення інференсу система автоматично обчислює основні метрики класифікації. У наведеному прикладі ассурагу моделі становить приблизно 75.89%, що характеризує загальну частку правильно класифікованих зображень.

Також наведено значення macro precision, recall та F1-score, які демонструють середню якість класифікації для всіх класів незалежно від їх розміру. Додатково відображаються weighted-метрики, що враховують кількість зображень у кожному класі датасету.

Результати оцінювання свідчать про те, що модель AlexNet забезпечує достатній рівень точності класифікації, однак має обмеження у здатності узагальнювати ознаки складних класів захворювань рослин. Це підтверджує доцільність подальшого порівняння її результатів із сучаснішою архітектурою GoogLeNet.

На рисунку 4.5 наведено результати оцінювання моделі GoogLeNet на тестовому наборі даних. У консолі відображаються етапи завантаження моделі, запуску інференсу та обчислення основних метрик класифікації після завершення тестування.

```

=====
GoogLeNet (InceptionV3) – Evaluating on test set (17,423 samples)
=====

[2026-05-14 09:17:01] INFO evaluate - Loading model: GoogLeNet
[2026-05-14 09:17:01] INFO evaluate - Model path: models/googlenet_best.keras
[2026-05-14 09:17:06] INFO evaluate - Model loaded (24.9M parameters, top-50 layers unfrozen)
[2026-05-14 09:17:06] INFO evaluate - Running inference on test set...
[2026-05-14 09:20:12] INFO evaluate - Inference complete in 185.9 s (~93.7 samples/sec)

Accuracy          : 0.9148
Precision (macro) : 0.9110
Recall (macro)    : 0.9146
F1-score (macro)  : 0.9101
Precision (weighted): 0.9062
Recall (weighted): 0.9117
F1-score (weighted): 0.9062

```

Рисунок 4.5 – Результати оцінювання моделі GoogLeNet

У наведеному прикладі модель GoogLeNet оцінюється на тестовій вибірці, що містить 17423 зображення. Під час запуску система автоматично завантажує модель із директорії `models`, а також виводить інформацію про кількість параметрів та використання fine-tuning верхніх шарів InceptionV3.

Після запуску інференсу система виконує класифікацію всіх зображень тестового набору та обчислює швидкість обробки даних. У наведеному прикладі продуктивність моделі становить приблизно 93.7 зображення за секунду, що є вищим показником порівняно з AlexNet.

За результатами оцінювання модель демонструє асигасу на рівні 91.48%, що свідчить про високий рівень правильності класифікації захворювань рослин. Це значення суттєво перевищує результати, отримані для AlexNet.

Macro precision становить 91.10%, а macro recall – 91.46%. Такі результати свідчать про стабільну якість класифікації для більшості класів датасету незалежно від кількості зображень у кожному класі.

Значення macro F1-score дорівнює 91.01%, що підтверджує збалансованість між precision та recall. Це означає, що модель не лише правильно визначає більшість захворювань, але й рідше допускає пропуски або помилкові класифікації.

Weighted precision, recall та F1-score також перевищують 90%, що свідчить про стабільну роботу моделі навіть з урахуванням дисбалансу між класами

датасету. Це є важливим показником для задачі класифікації захворювань рослин, де різні класи можуть містити різну кількість зображень.

Порівняно з результатами AlexNet, модель GoogLeNet демонструє значно вищу точність та кращу узагальнювальну здатність. Це підтверджує ефективність використання transfer learning та сучасних архітектур глибокого навчання для задач комп'ютерного зору.

Також результати оцінювання підтверджують, що fine-tuning верхніх шарів InceptionV3 позитивно вплинув на якість класифікації та дозволив адаптувати попередньо навчену модель до специфіки датасету захворювань рослин.

4.4 Порівняльний аналіз результатів

У даному підрозділі буде проведено порівняльний аналіз результатів роботи моделей AlexNet та GoogLeNet на основі обчислених метрик класифікації. Для забезпечення більш наочного представлення результатів та спрощення їх інтерпретації на основі отриманих даних буде побудовано ряд графіків і діаграм.

Одним із основних елементів аналізу стане графік порівняння узагальнених метрик моделей, зокрема accuracy, precision, recall та F1-score. Такий графік дозволить швидко оцінити різницю між моделями та визначити, яка архітектура демонструє кращу загальну якість класифікації.

Також буде побудовано графік порівняння F1-score для кожного окремого класу захворювання. Це дозволить проаналізувати, наскільки ефективно кожна модель класифікує конкретні типи захворювань рослин та які класи є найбільш складними для розпізнавання.

Окремо буде сформовано графік приросту F1-score між GoogLeNet та AlexNet для кожного класу. Такий підхід дозволяє оцінити, у яких саме класах використання сучаснішої архітектури забезпечує найбільше покращення результатів класифікації.

Для детального аналізу співвідношення precision та recall буде використано точкові діаграми, на яких кожен клас представлений окремою точкою. Розмір точок буде відповідати кількості зображень у класі, а колір – значенню F1-score. Це дозволить оцінити баланс між точністю та повнотою класифікації для різних типів захворювань.

Також планується побудова графіків найкращих та найгірших класів за значенням F1-score. Це дозволить визначити, які захворювання моделі класифікують найбільш точно, а для яких класів спостерігається найбільша кількість помилок.

У межах аналізу буде досліджено не лише загальну якість моделей, але й стабільність їх роботи для різних культур та типів захворювань. Завдяки цьому стане можливим визначення класів, для яких необхідне подальше покращення архітектури або збільшення обсягу навчальних даних.

Графічне представлення результатів також дозволить оцінити вплив transfer learning та fine-tuning на результати GoogLeNet. Порівняння з AlexNet дозволить наочно продемонструвати переваги використання попередньо навчених моделей та сучасних глибоких архітектур.

Побудовані графіки забезпечать більш зручне та наочне порівняння моделей, ніж використання лише текстових значень метрик. Це дозволить краще інтерпретувати результати експерименту та сформулювати обґрунтовані висновки щодо ефективності реалізованих підходів.

На рисунку 4.6 наведено узагальнене порівняння основних метрик якості моделей AlexNet та GoogLeNet. Графік дозволяє наочно оцінити різницю між архітектурами за ключовими показниками класифікації, зокрема accuracy, precision, recall та F1-score.

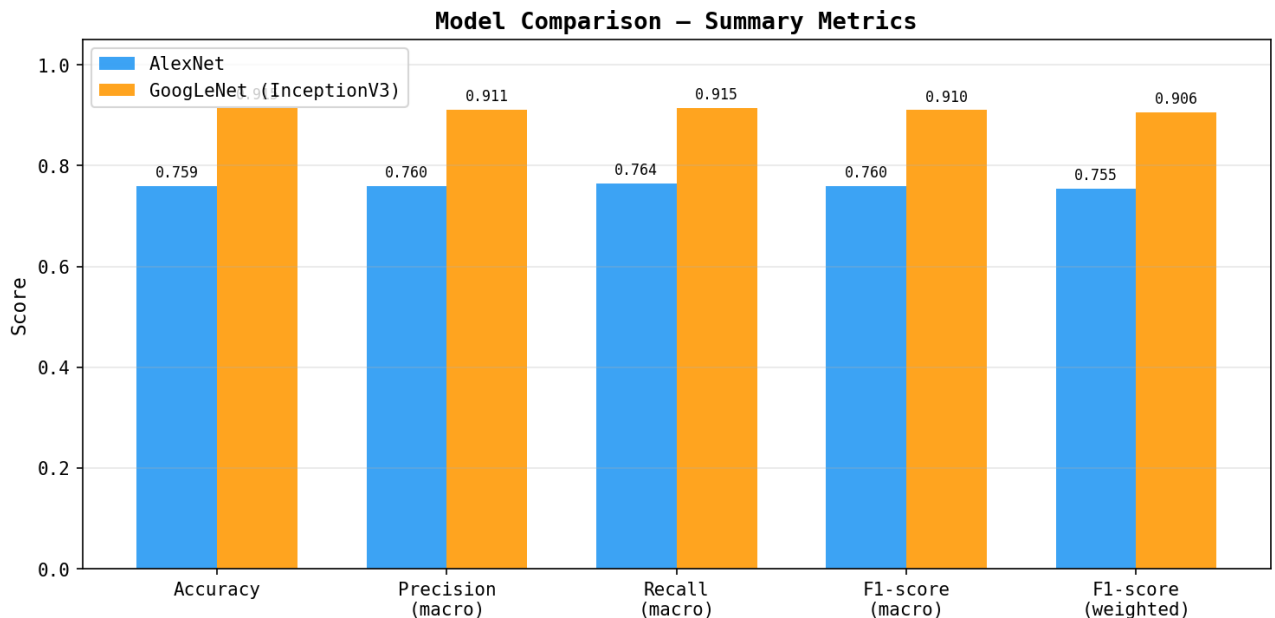


Рисунок 4.6 – Узагальнене порівняння основних метрик якості моделей

З отриманих результатів видно, що модель GoogLeNet демонструє вищі значення для всіх досліджуваних метрик. Значення accuracy для GoogLeNet становить приблизно 91.5%, тоді як для AlexNet цей показник перебуває на рівні близько 75.9%. Це свідчить про значно вищу загальну точність класифікації у моделі на основі InceptionV3.

Аналогічна ситуація спостерігається і для macro precision та macro recall. GoogLeNet забезпечує значення precision близько 91.1% та recall близько 91.5%, тоді як AlexNet демонструє результати на рівні приблизно 76%. Це означає, що GoogLeNet значно краще визначає правильні класи та рідше допускає помилкові класифікації.

Особливу увагу можна звернути на macro F1-score, який є узагальненою метрикою якості класифікації. Значення F1-score для GoogLeNet перевищує 91%, тоді як для AlexNet воно становить близько 76%. Це підтверджує, що сучасніша архітектура забезпечує кращий баланс між precision та recall.

Weighted F1-score також демонструє перевагу GoogLeNet. Це свідчить про те, що модель стабільно працює навіть за наявності дисбалансу між класами датасету. AlexNet у цьому випадку показує нижчу узагальнювальну здатність та гіршу стабільність класифікації.

Отримані результати підтверджують ефективність використання transfer learning та fine-tuning у моделі GoogLeNet. Попередньо навчені ваги ImageNet та адаптація верхніх шарів InceptionV3 дозволили значно покращити якість класифікації порівняно з класичною архітектурою AlexNet.

Також графік демонструє, що різниця між моделями є стабільною для всіх метрик, а не лише для окремих показників. Це свідчить про системну перевагу GoogLeNet у задачі автоматизованої діагностики захворювань рослин.

На рисунку 4.7 наведено порівняння значень precision та recall для окремих класів захворювань у моделях AlexNet та GoogLeNet. Кожна точка на графіку відповідає окремому класу датасету, а її розмір визначається кількістю зображень у відповідному класі. Колір точки відображає значення F1-score.

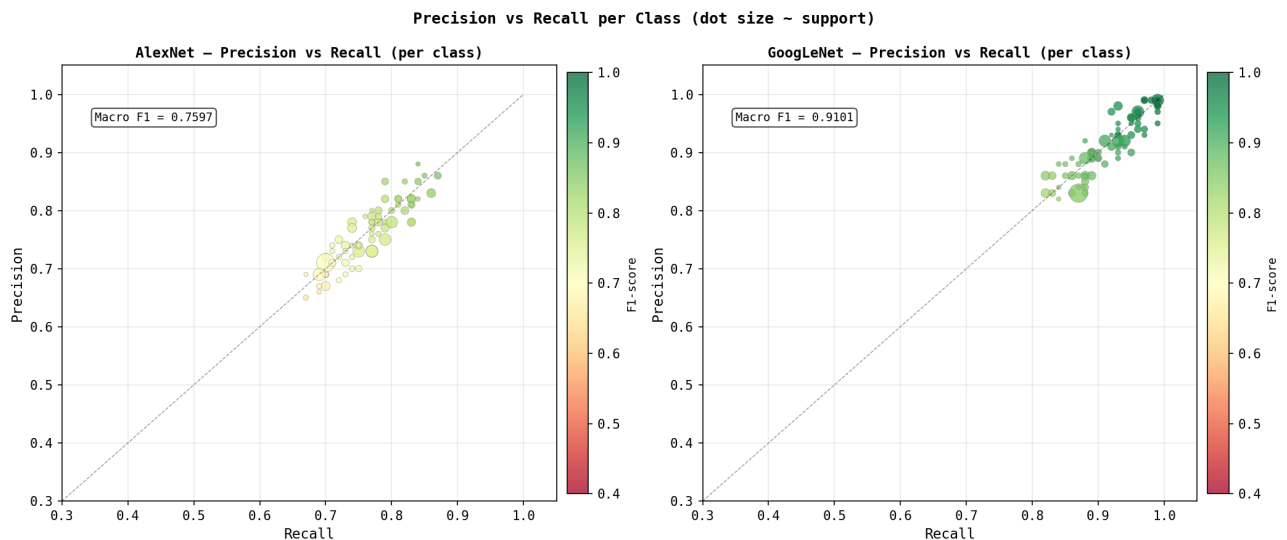


Рисунок 4.7 – Порівняння значень precision та recall для окремих класів захворювань

Ліва частина рисунка демонструє результати моделі AlexNet. Можна спостерігати, що більшість точок розташовані в діапазоні recall та precision приблизно від 0.7 до 0.85. Це свідчить про достатньо стабільну, але не ідеальну якість класифікації для більшості класів.

Частина точок у AlexNet знаходиться на помітній відстані від верхнього правого кута графіка, що означає нижчі значення precision або recall для окремих

захворювань. Це вказує на наявність класів, для яких модель допускає помітну кількість помилкових класифікацій або пропусків.

Значення macro F1-score для AlexNet становить приблизно 0.7597, що підтверджує результати попереднього аналізу метрик. Також можна помітити, що кольори точок переважно відповідають середньому рівню F1-score, без значної концентрації класів із дуже високими результатами.

Права частина рисунка демонструє аналогічні результати для моделі GoogLeNet. У цьому випадку більшість точок зміщені ближче до верхнього правого кута графіка, де precision та recall наближаються до 1.0.

Таке розташування точок свідчить про значно кращу якість класифікації для більшості класів. GoogLeNet демонструє більш збалансовані результати між precision та recall, що означає меншу кількість помилкових спрацювань та пропусків захворювань.

Macro F1-score для GoogLeNet становить приблизно 0.9101, що значно перевищує результат AlexNet. Також можна помітити, що більшість точок мають темно-зелене забарвлення, яке відповідає високим значенням F1-score.

Порівняння двох графіків демонструє, що GoogLeNet забезпечує більш стабільну класифікацію між різними класами датасету. Розкид точок у цієї моделі є меншим, а значення precision та recall для більшості класів є значно вищими.

Окремо варто відзначити, що навіть класи з великою кількістю зображень у GoogLeNet демонструють високі значення precision та recall. Це свідчить про ефективне узагальнення ознак та коректну адаптацію моделі до структури датасету.

На рисунку 4.8 наведено порівняння значень F1-score для окремих класів захворювань рослин у моделях AlexNet та GoogLeNet. Для кожного класу відображено два значення – результат AlexNet та результат GoogLeNet, що дозволяє безпосередньо порівняти якість класифікації для конкретних типів захворювань.

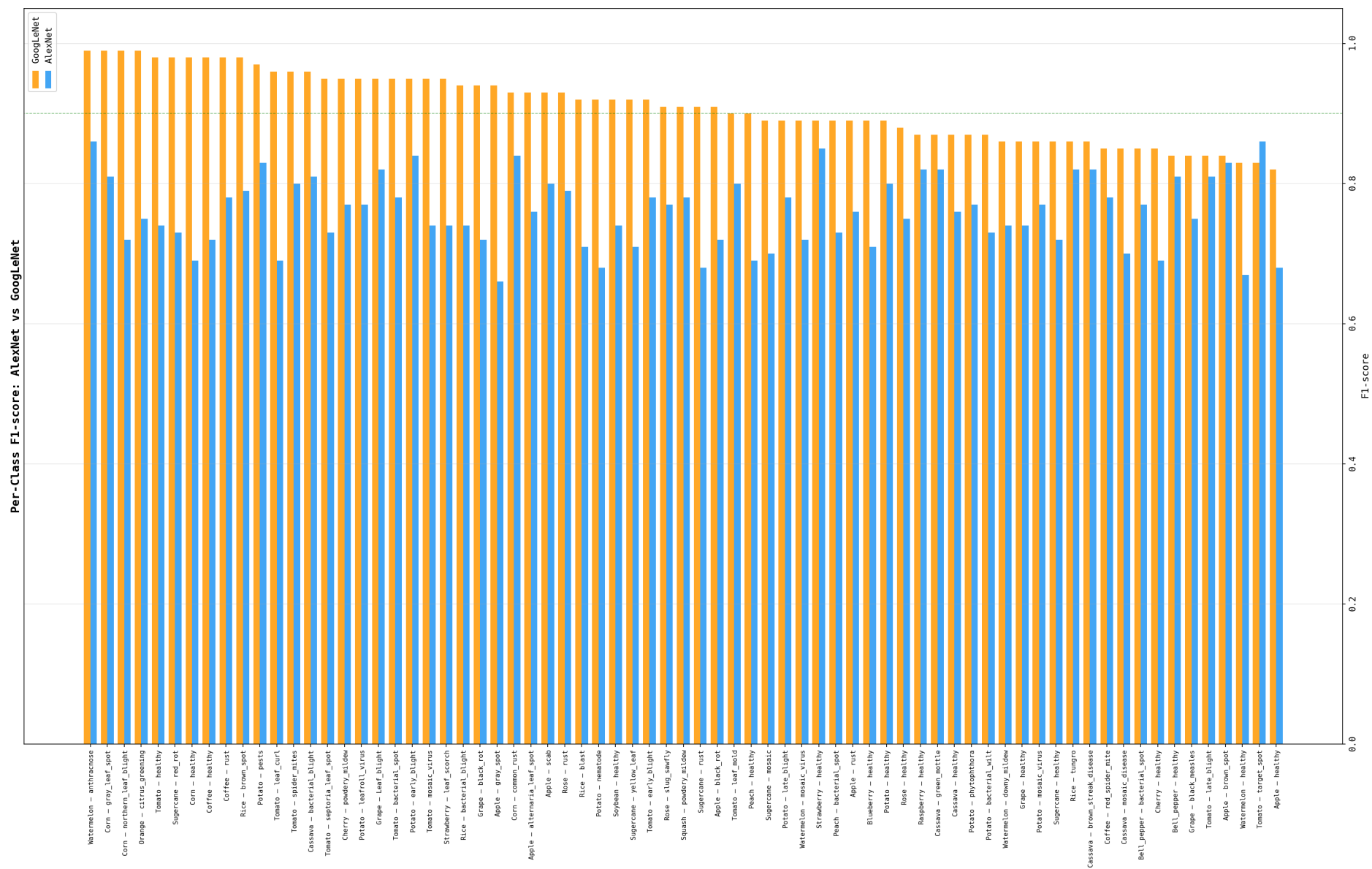


Рисунок 4.8 – Порівняння значень F1-score

З графіка видно, що модель GoogLeNet демонструє вищі значення F1-score для переважної більшості класів датасету. У багатьох випадках значення F1-score для GoogLeNet наближається до 0.95–1.0, що свідчить про дуже високу якість класифікації.

Особливо помітною є перевага GoogLeNet для класів, пов'язаних із захворюваннями кукурудзи, кави, цитрусових та томатів. Для цих класів різниця між моделями є значною, що вказує на кращу здатність сучасної архітектури виділяти складні візуальні ознаки.

AlexNet також демонструє достатньо стабільні результати для окремих класів, однак її значення F1-score у більшості випадків перебувають у діапазоні 0.7–0.85. Це свідчить про нижчу узагальнювальну здатність моделі та більшу кількість помилок під час класифікації.

На графіку можна помітити, що для окремих класів різниця між моделями є мінімальною. Це означає, що деякі типи захворювань мають достатньо виражені візуальні ознаки та можуть бути успішно класифіковані навіть простішими архітектурами.

Водночас існують класи, для яких GoogLeNet забезпечує суттєве покращення F1-score порівняно з AlexNet. Такі результати свідчать про ефективність transfer learning та fine-tuning у задачах класифікації складних зображень рослин.

Пунктирна вертикальна лінія на графіку позначає високий рівень F1-score, що дозволяє візуально оцінити, які класи наближаються до майже ідеальної класифікації. Для GoogLeNet значна кількість класів знаходиться поблизу цього значення.

Також можна спостерігати, що розкид результатів у GoogLeNet є меншим, ніж у AlexNet. Це свідчить про більш стабільну якість класифікації між різними культурами та типами захворювань.

Отримані результати підтверджують, що використання сучасної архітектури InceptionV3 дозволяє значно покращити якість класифікації для більшості класів датасету. GoogLeNet демонструє не лише вищі середні метрики, але й стабільніші результати для окремих категорій захворювань.

На рисунку 4.9 наведено графік приросту значення F1-score моделі GoogLeNet порівняно з AlexNet для кожного класу датасету. Значення на горизонтальній осі відображають різницю між F1-score двох моделей, тобто приріст якості класифікації при використанні GoogLeNet.

Зеленим кольором позначені класи, для яких GoogLeNet демонструє покращення результатів відносно AlexNet. Червоним кольором відображено класи, для яких приріст є від'ємним, тобто AlexNet показала дещо кращий результат.

З графіка видно, що для переважної більшості класів GoogLeNet забезпечує позитивний приріст F1-score. Це підтверджує загальну перевагу архітектури InceptionV3 у задачі класифікації захворювань рослин.

Найбільший приріст спостерігається для класів Corn__healthy, Apple__gray_spot, Tomato__leaf_curl та Corn__northern_leaf_blight. Для цих класів різниця F1-score перевищує 0.25, що свідчить про суттєве покращення якості класифікації.

Такі результати вказують на те, що GoogLeNet значно краще працює з класами, які мають складні або схожі візуальні ознаки. Використання transfer learning дозволило моделі ефективніше виділяти дрібні особливості зображень та покращити узагальнювальну здатність.

Для більшості класів приріст перебуває у діапазоні від 0.1 до 0.2. Це означає, що покращення результатів є стабільним та спостерігається практично для всіх типів захворювань рослин, а не лише для окремих категорій.

Також можна помітити, що існує лише один клас – Tomato__target_spot, для якого GoogLeNet демонструє незначне погіршення F1-score порівняно з AlexNet. Однак величина цього відхилення є невеликою та не впливає на загальний результат дослідження.

Наявність лише одного класу з негативним приростом свідчить про високу стабільність GoogLeNet та ефективність використаного підходу transfer learning. У більшості випадків сучасна архітектура забезпечує кращу якість класифікації незалежно від типу рослини або захворювання.

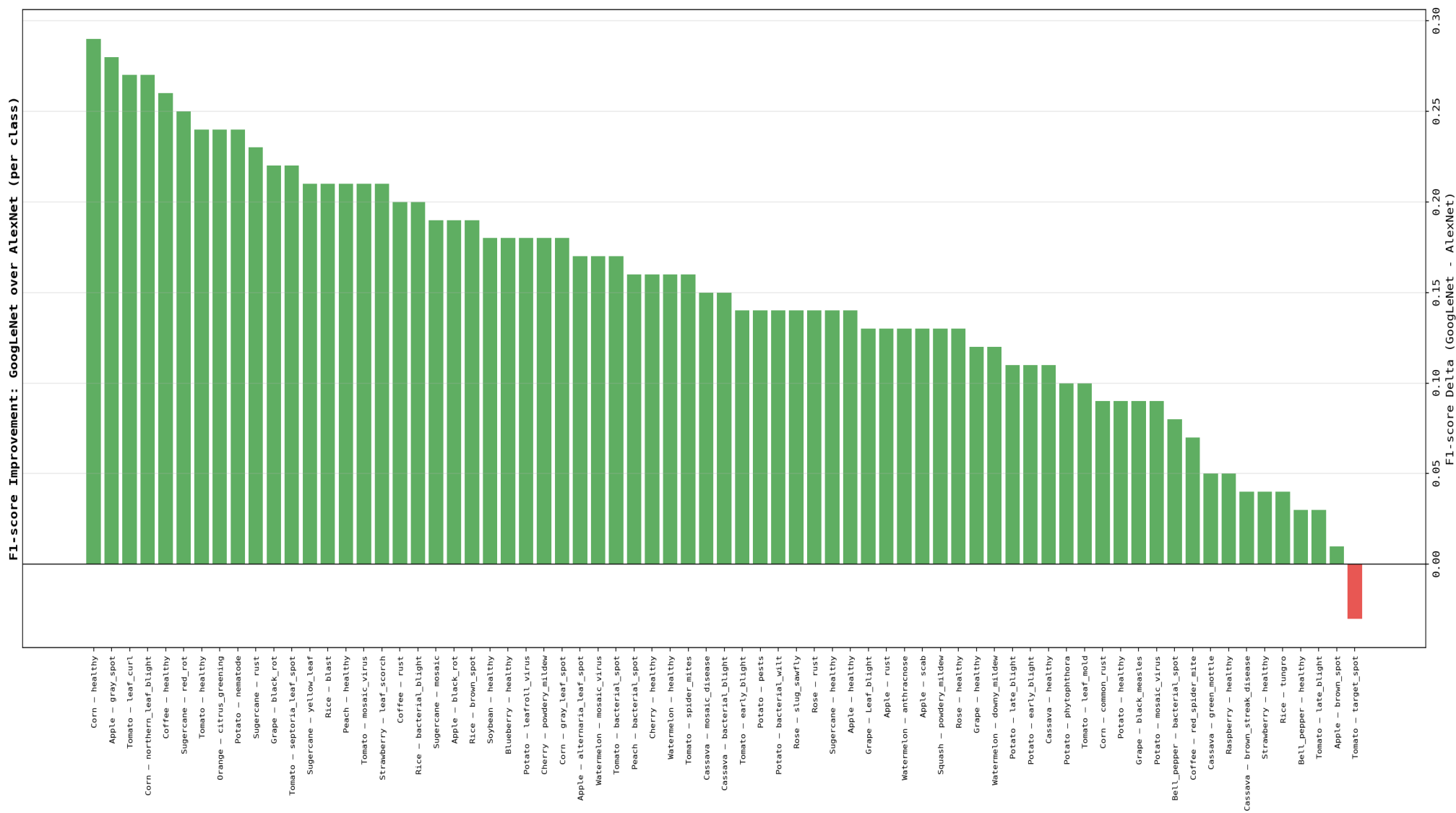


Рисунок 4.9 – Графік приросту значення F1-score між моделями

Отримані результати також підтверджують, що AlexNet має обмеження у здатності працювати зі складними наборами ознак, особливо у випадках, коли різні захворювання мають схожі симптоми на листі рослин.

На рисунку 4.10 наведено порівняння десяти класів із найвищими значеннями F1-score для моделі GoogLeNet. Для кожного класу також відображено відповідний результат моделі AlexNet, що дозволяє оцінити різницю між архітектурами для найбільш успішних випадків класифікації.

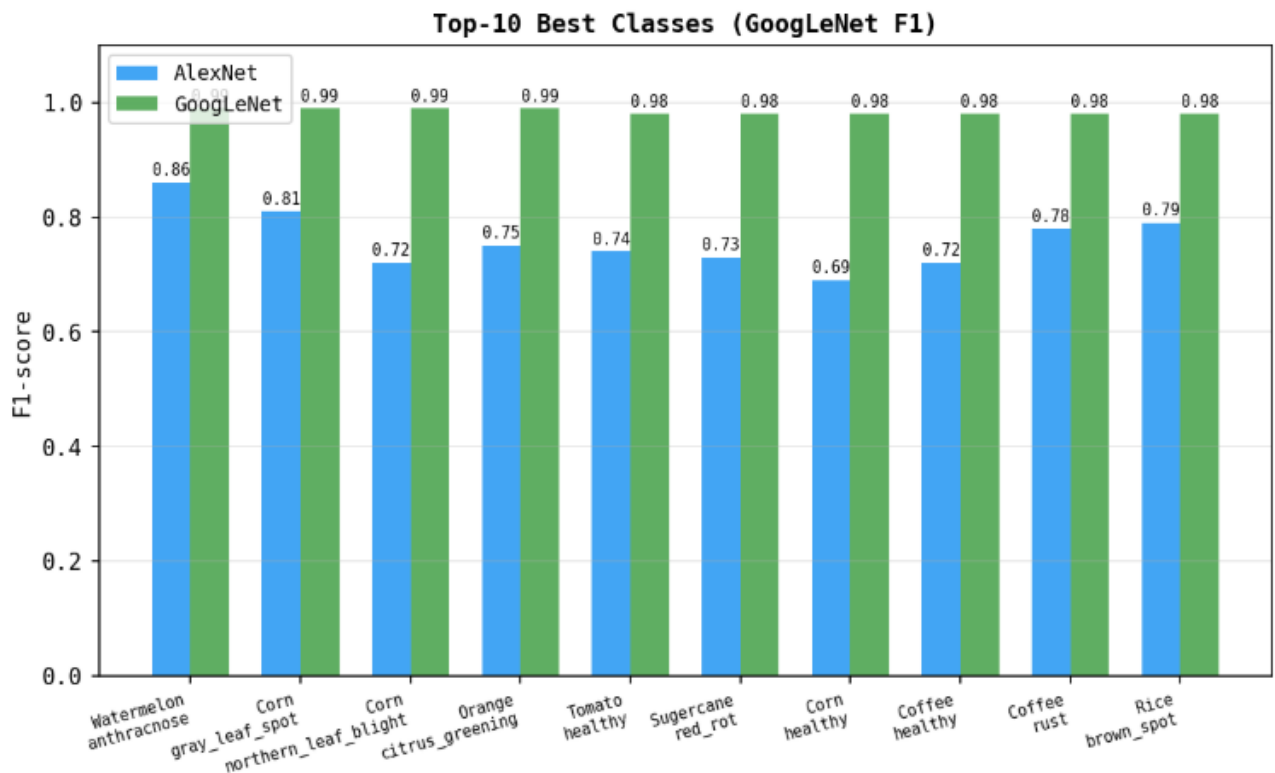


Рисунок 4.10 – Порівняння десяти класів із найвищими значеннями F1-score

З графіка видно, що для всіх представлених класів GoogLeNet демонструє дуже високі результати класифікації. Значення F1-score для більшості класів перебуває у діапазоні від 0.98 до 0.99, що свідчить про майже ідеальну якість розпізнавання відповідних захворювань та станів рослин.

Найвищі результати модель показує для класів Watermelon__anthracnose, Corn__gray_leaf_spot та Corn__northern_leaf_blight, де F1-score досягає значення 0.99. Це означає, що модель практично не допускає помилок під час класифікації цих захворювань.

Також високі результати спостерігаються для класів, пов'язаних із citrusовими, томатами, цукровою тростиною, кавою та рисом. Це свідчить про те, що модель ефективно працює з різними типами культур та здатна узагальнювати ознаки для широкого спектра захворювань.

Порівняння з AlexNet демонструє значну різницю у якості класифікації. Для тих самих класів AlexNet показує результати переважно у діапазоні 0.69–0.86, що є суттєво нижчим показником.

Особливо помітною є різниця для класів Corn___northern_leaf_blight, Corn___healthy та Sugarcane___red_rot, де приріст F1-score становить понад 0.2. Це підтверджує перевагу GoogLeNet у задачах класифікації складних візуальних ознак.

Отримані результати свідчать про те, що GoogLeNet найбільш ефективно працює з класами, які мають чітко виражені та стабільні симптоми захворювань. Використання transfer learning дозволило моделі краще адаптуватися до таких ознак та значно покращити якість класифікації.

Також графік демонструє, що сучасна архітектура забезпечує не лише високі середні показники, але й здатність досягати майже максимальної якості для окремих класів датасету.

На рисунку 4.11 наведено порівняння десяти класів із найнижчими значеннями F1-score для моделі GoogLeNet. Для кожного класу також відображено результати AlexNet, що дозволяє оцінити поведінку моделей у найбільш складних випадках класифікації.

Навіть для найгірших класів GoogLeNet демонструє відносно високі результати. Значення F1-score для всіх представлених категорій перебуває приблизно у діапазоні 0.82–0.85, що свідчить про достатньо стабільну якість класифікації навіть для складних типів захворювань.

Найнижчі результати спостерігаються для класів Apple___healthy, Watermelon___healthy та Cassava___mosaic_disease. Це може бути пов'язано з тим, що окремі ознаки цих класів частково перетинаються з іншими категоріями або мають недостатньо виражені візуальні особливості.

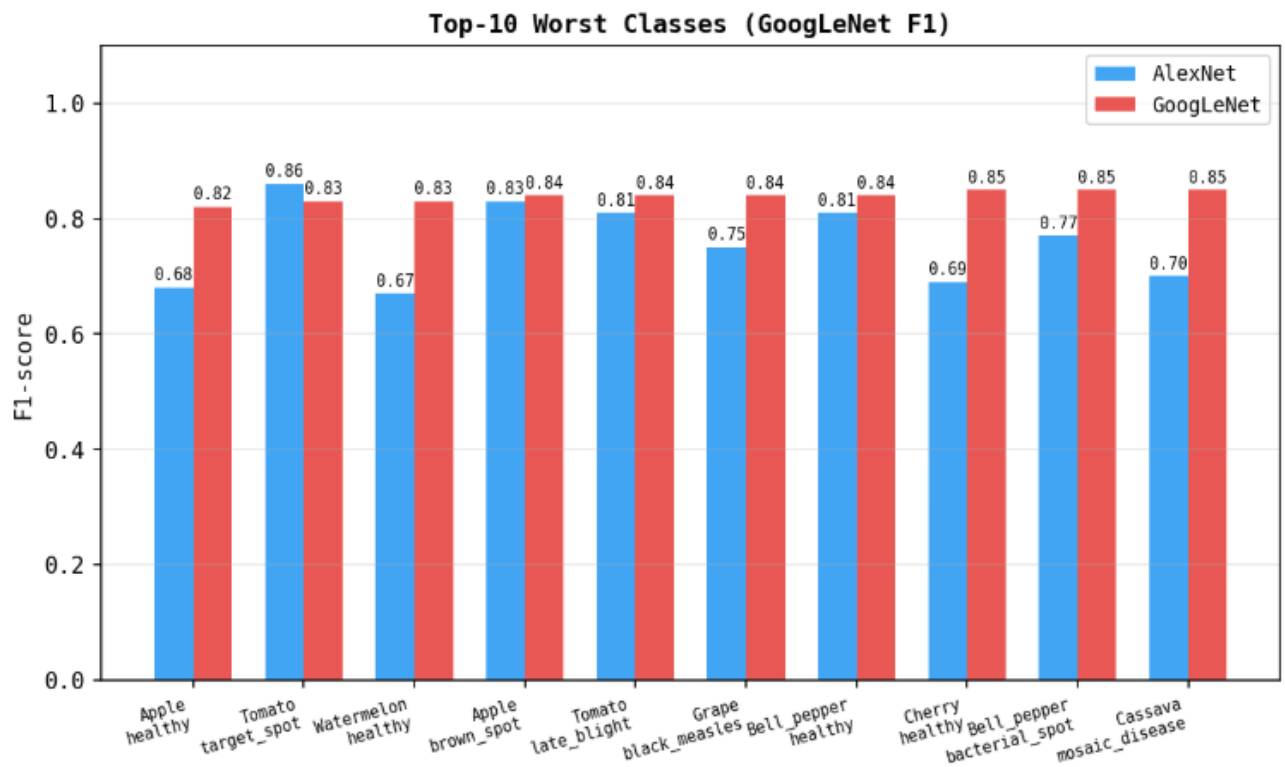


Рисунок 4.11 – Порівняння десяти класів із найнижчими значеннями F1-score

Окрему увагу можна звернути на клас `Tomato__target_spot`, для якого AlexNet демонструє дещо вищий результат порівняно з GoogLeNet. Це є одним із небагатьох випадків у дослідженні, де класична архітектура показує кращу якість класифікації.

Для більшості інших класів GoogLeNet все одно забезпечує помітний приріст F1-score. Особливо це помітно для класів `Cherry__healthy`, `Bell_pepper__bacterial_spot` та `Cassava__mosaic_disease`, де різниця між моделями перевищує 0.1–0.15.

Результати свідчать про те, що навіть у найскладніших випадках сучасна архітектура InceptionV3 демонструє більш стабільну роботу та кращу здатність узагальнювати ознаки порівняно з AlexNet.

Також можна помітити, що проблемними для моделей є не лише окремі типи захворювань, але й деякі класи здорових рослин. Це пояснюється тим, що візуальні ознаки здорового листя іноді можуть бути схожими між різними культурами або містити природні варіації кольору та текстури.

Порівняння найгірших класів є важливим етапом аналізу, оскільки дозволяє визначити обмеження моделей та виявити категорії, для яких у

майбутньому може бути доцільним розширення датасету або використання складніших методів обробки зображень.

Проведений порівняльний аналіз результатів класифікації показав, що обидві реалізовані моделі здатні виконувати автоматизовану діагностику захворювань рослин за зображеннями листя, однак їх ефективність суттєво відрізняється. Модель AlexNet продемонструвала задовільні результати та підтвердила можливість використання класичних згорткових нейронних мереж для задач комп'ютерного зору в аграрній сфері.

Водночас модель GoogLeNet на основі InceptionV3 забезпечила значно вищі результати за всіма основними метриками оцінювання, зокрема accuracy, precision, recall та F1-score. Використання transfer learning і fine-tuning дозволило суттєво покращити якість класифікації та забезпечити більш стабільну роботу моделі для різних класів захворювань.

Побудовані графіки підтвердили, що GoogLeNet демонструє кращу узагальнювальну здатність, меншу схильність до перенавчання та вищу стабільність результатів між окремими класами датасету. Для переважної більшості категорій спостерігався позитивний приріст F1-score порівняно з AlexNet.

Також результати аналізу показали, що сучасні архітектури глибокого навчання найбільш ефективно працюють із класами, які мають складні або слабо виражені візуальні ознаки. GoogLeNet забезпечила кращу здатність виділяти такі ознаки та коректно виконувати класифікацію навіть у складних випадках.

ВИСНОВКИ

У ході виконання роботи було проведено дослідження моделей глибокого навчання для задачі автоматизованої діагностики захворювань рослин за зображеннями листя. На початковому етапі було виконано аналіз предметної області та досліджено особливості використання комп'ютерного зору в аграрній сфері. Розглянуто основні типи захворювань рослин, їх вплив на аграрне виробництво та актуальність створення інтелектуальних систем автоматизованої діагностики.

У межах теоретичного дослідження було виконано огляд сучасних моделей глибокого навчання для обробки зображень. Проаналізовано класичні згорткові нейронні мережі, сучасні архітектури CNN та підходи transfer learning. Особливу увагу приділено архітектурам AlexNet та GoogLeNet, які були обрані для подальшого експериментального дослідження.

Також було проведено аналіз існуючих наукових робіт та реалізованих рішень у сфері діагностики захворювань рослин. У межах аналізу досліджено результати сучасних моделей класифікації зображень та визначено основні підходи до побудови інтелектуальних систем діагностики.

На основі проведеного аналізу було сформовано постановку задачі дослідження та визначено основні вимоги до системи діагностики. Зокрема, визначено вимоги щодо точності класифікації, стандартизації вхідних даних, підтримки оцінювання моделей за допомогою метрик якості та забезпечення стабільності результатів.

У межах підготовки експериментального середовища було виконано пошук та аналіз наборів даних для навчання моделей. Як основний датасет було обрано Plant Leaf Diseases Training Dataset, що є розширеним об'єднанням декількох датасетів на основі PlantVillage та містить велику кількість зображень різних культур і типів захворювань.

Після вибору датасету було реалізовано процес попередньої обробки зображень. Усі зображення було приведено до єдиного формату RGB, виконано зміну розміру до 224×224 пікселів та збереження у стандартизованому форматі

PNG. Це дозволило забезпечити сумісність даних із моделями глибокого навчання та стабільність процесу навчання.

На наступному етапі було обґрунтовано вибір архітектур моделей глибокого навчання та реалізовано моделі AlexNet і GoogLeNet на основі бібліотеки TensorFlow/Keras. Для GoogLeNet було використано підхід transfer learning із попередньо навченими вагами ImageNet та подальшим fine-tuning верхніх шарів InceptionV3.

Також було виконано налаштування процесу навчання моделей. Визначено параметри навчання, механізми регуляризації, callback-функції, методи автоматичного зменшення швидкості навчання та ранньої зупинки навчання для запобігання перенавчанню моделей.

У межах практичної частини роботи було розроблено консольний програмний засіб для тестування моделей. Реалізований застосунок дозволяє завантажувати навчені моделі, виконувати попередню обробку зображень, проводити класифікацію та виводити результати передбачення разом із ймовірностями та списком найбільш імовірних класів.

Після реалізації моделей було проведено експериментальне навчання AlexNet та GoogLeNet на підготовленому датасеті. У процесі експерименту було проаналізовано динаміку зміни accuracy та loss, а також оцінено вплив transfer learning і fine-tuning на якість класифікації.

Для оцінювання ефективності моделей було використано метрики accuracy, precision, recall та F1-score. Оцінювання виконувалось на тестовій вибірці, що дозволило об'єктивно визначити здатність моделей узагальнювати знання та виконувати класифікацію нових зображень.

На основі отриманих результатів було проведено порівняльний аналіз моделей із використанням графічного представлення даних. Побудовано графіки загального порівняння метрик, аналізу precision та recall для окремих класів, порівняння F1-score та оцінювання приросту якості між моделями.

Результати дослідження показали, що модель GoogLeNet забезпечує значно вищу точність класифікації та кращу узагальнювальну здатність порівняно з AlexNet. Використання transfer learning дозволило суттєво

покращити якість класифікації та забезпечити стабільні результати для більшості класів захворювань рослин.

Результатом виконаної роботи є сформована теоретична та практична основа для побудови інтелектуальних систем автоматизованої діагностики захворювань рослин на основі моделей глибокого навчання. У межах дослідження було підготовлено датасет, реалізовано процес попередньої обробки даних, створено та навчено моделі глибокого навчання, розроблено програмний засіб тестування моделей та виконано комплексний аналіз отриманих результатів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Agrios George N. Plant Pathology. Elsevier Academic Press, 2005. 922 p.
2. Food and Agriculture Organization of the United Nations (FAO) The State of Food and Agriculture. FAO, 2020. 210 p.
3. Mohanty Sharada P., Hughes David P., Salathé Marcel Using Deep Learning for Image-Based Plant Disease Detection, 2016. 7 p.
4. Ferentinos Konstantinos P. Deep learning models for plant disease detection and diagnosis, 2018. 8 p.
5. Picon Arturo, Alvarez-Gila Asier, Seitz Markus, Ortiz-Barredo Aitor, Echazarra Javier, Johannes Andreas Deep convolutional neural networks for crop disease classification in the wild, 2019. 11 p.
6. Ferentinos Konstantinos P. Deep learning models for plant disease detection and diagnosis, 2018. 8 p.
7. Food and Agriculture Organization of the United Nations (FAO) The State of Food and Agriculture. FAO, 2020. 210 p.
8. Szeliski Richard Computer Vision: Algorithms and Applications. Springer, 2010. 812 p.
9. Kamilaris Andreas, Prenafeta-Boldú Francesc X. Deep learning in agriculture: A survey, 2018. 21 p.
10. Mohanty Sharada P., Hughes David P., Salathé Marcel Using Deep Learning for Image-Based Plant Disease Detection, 2016. 7 p.
11. Deng Jia, Dong Wei, Socher Richard, Li Li-Jia, Li Kai, Fei-Fei Li ImageNet: A Large-Scale Hierarchical Image Database, 2009. 8 p. □ Krizhevsky Alex, Sutskever Ilya, Hinton Geoffrey E. ImageNet Classification with Deep Convolutional Neural Networks, 2012. 10 p.
12. Szegedy Christian, Liu Wei, Jia Yangqing, Sermanet Pierre, Reed Scott, Anguelov Dragomir, Erhan Dumitru, Vanhoucke Vincent, Rabinovich Andrew Going Deeper with Convolutions, 2015. 12 p.
13. He Kaiming, Zhang Xiangyu, Ren Shaoqing, Sun Jian Deep Residual Learning for Image Recognition, 2016. 8 p.

14. Simonyan Karen, Zisserman Andrew Very Deep Convolutional Networks for Large-Scale Image Recognition, 2015. 14 p.
15. LeCun Yann, Bottou Léon, Bengio Yoshua, Haffner Patrick Gradient-Based Learning Applied to Document Recognition, 1998. 54 p.
16. Goodfellow Ian, Bengio Yoshua, Courville Aaron Deep Learning. MIT Press, 2016. 775 p.
17. Bishop Christopher M. Pattern Recognition and Machine Learning. Springer, 2006. 738 p.
18. Chollet François Deep Learning with Python. Manning Publications, 2021. 504 p.
19. Russell Stuart, Norvig Peter Artificial Intelligence: A Modern Approach. Pearson, 2021. 1136 p.
20. Shorten Connor, Khoshgoftaar Taghi M. A survey on Image Data Augmentation for Deep Learning, 2019. 48 p.
21. Tan Mingxing, Le Quoc EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, 2019. 10 p.
22. Dosovitskiy Alexey, Beyer Lucas, Kolesnikov Alexander, Weissenborn Dirk, Zhai Xiaohua, Unterthiner Thomas An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2021. 22 p.
23. Ronneberger Olaf, Fischer Philipp, Brox Thomas U-Net: Convolutional Networks for Biomedical Image Segmentation, 2015. 8 p.
24. Redmon Joseph, Divvala Santosh, Girshick Ross, Farhadi Ali You Only Look Once: Unified, Real-Time Object Detection, 2016. 10 p.
25. Ren Shaoqing, He Kaiming, Girshick Ross, Sun Jian Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, 2015. 15 p.
26. TensorFlow An End-to-End Open Source Machine Learning Platform. URL: <https://www.tensorflow.org/> (date of access: 14.04.2026).
27. Kaggle Plant Leaf Diseases Dataset. URL: <https://www.kaggle.com/> (date of access: 03.05.2026).