

УДК 004.056

DOI [https://doi.org/10.15589/znp2025.2\(500\).35](https://doi.org/10.15589/znp2025.2(500).35)

## CNN-BASED INTELLIGENT SYSTEM FOR PLANT STATE RECOGNITION

ІНТЕЛЕКТУАЛЬНА СИСТЕМА РОЗПІЗНАВАННЯ СТАНУ РОСЛИН  
НА ОСНОВІ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ**Olena M. Haitan**

azalie@ukr.net

ORCID: 0000-0002-7228-9937

**Serhii P. Aleshyn**

aleshsp@ukr.net

ORCID: 0009-0002-4239-4194

**I. Louattar**

samuelmohamedovic@gmail.com

**О. М. Гайтан,**

старший викладач

**С. П. Альшин,**

канд. техн. наук, доцент

**І. Луаттар,**

студент

*National University "Yuri Kondratyuk Poltava Polytechnic", Poltava**Національний університет «Полтавська політехніка імені Юрія Кондратюка», м. Полтава*

**Abstract.** Plant diseases represent a significant challenge to global food security, resulting in annual crop losses ranging from 20% to 40%. There is a high demand for the implementation of intelligent plant health monitoring systems that can provide timely detection of disease symptoms.

*Purpose.* This study aims to develop and evaluate a deep learning model and an intelligent system for the automated classification of crop diseases, with a focus on potato crops as a representative case.

*Method.* The proposed system is based on a Convolutional Neural Network (CNN), which has demonstrated high efficiency in image analysis tasks. The model was trained using a dataset from the Kaggle platform. The dataset underwent preprocessing, including data cleaning, pixel normalization, and augmentation techniques such as rotation and contrast adjustment, to enhance the robustness and generalization capacity of the model. The standardized image dimension is 256×256 pixels in RGB format. The dataset was segmented into training, validation, and test subsets, with the option of random permutation of elements to prevent overfitting. The model architecture comprises a CNN with a Softmax activation function in the output layer for multi-class classification. The Adam optimizer was employed for efficient parameter tuning. Furthermore, the trained model has been integrated into a web application using the FastAPI framework, thereby enabling users to upload images of plant leaves and receive real-time classification results with corresponding probabilities.

*Results.* The model demonstrated an accuracy of 95% on the test sample, indicating its high reliability in identifying and classifying potato leaf diseases. The findings of the study confirm the effectiveness of employing computer vision and deep learning methodologies for early disease diagnosis.

*Scientific novelty.* This study presents a comprehensive pipeline from model training to real-world deployment, emphasizing the application of modern CNN architectures for agricultural diagnostics. The novelty lies in the integration of a classification model into a web-based platform, offering a practical tool for field use.

*Practical importance.* The developed system enables early and accurate disease detection, supporting agricultural decision-making processes. It has the potential for scalability to other types of crops and can serve as the basis for the development of a real-time plant health monitoring platform, which would assist agricultural producers in increasing yields and reducing economic losses. The potential integration with drone or smartphone imaging systems makes it a promising tool for modern precision agriculture.

**Key words:** deep learning; convolutional neural networks; potato; plant leaf; classification; recognition; Adam optimizer.

**Анотація.** Хвороби рослин становлять серйозну загрозу для глобальної продовольчої безпеки, спричиняючи щорічні втрати врожаю в межах від 20 % до 40 %. Це зумовлює високий попит на впровадження інтелектуальних систем моніторингу стану рослин, здатних своєчасно виявляти симптоми захворювань.

*Мета.* Метою дослідження є розробка та оцінка моделі глибокого навчання та інтелектуальної системи для автоматизованої класифікації хвороб сільськогосподарських культур, з акцентом на картоплю в якості репрезентативного прикладу.

*Метод.* Запропонована система базується на згортковій нейронній мережі (CNN), яка продемонструвала високу ефективність в задачах аналізу зображень. Модель навчалась на основі набору даних з платформи Kaggle. Перед обробкою дані проходили етапи очищення, нормалізації пікселів та розширення (аугментації), включно з обертанням зображень та зміною контрасту, що підвищило здатність моделі до узагальнення. Розмір зображень було стандартизовано до 256×256 пікселів у форматі RGB. Набір даних було поділено на тренувальну, валідаційну та тестову вибірки з можливістю випадкової перестановки елементів для уникнення перенавчання. Архітектура моделі включає CNN із функцією активації Softmax на вихідному шарі для багатокласової класифікації. Для оптимізації навчання використовувався алгоритм Adam. Крім того, навчена модель була інтегрована у веб-додаток на основі фреймворку FastAPI, що дозволяє користувачам завантажувати зображення листя рослин і отримувати результати класифікації з відповідними ймовірностями в реальному часі.

*Результати.* Модель продемонструвала точність 95% на тестовій вибірці, що свідчить про її високу надійність у виявленні та класифікації хвороб листя картоплі. Отримані результати підтверджують ефективність використання методів комп'ютерного зору та глибокого навчання для ранньої діагностики захворювань.

*Наукова новизна.* У роботі представлено повний цикл – від навчання моделі до її впровадження у практичний додаток, з акцентом на використання сучасних архітектур CNN для аграрної діагностики. Новизна полягає в інтеграції моделі класифікації у вебплатформу, що забезпечує практичну цінність для використання в польових умовах.

*Практичне значення.* Розроблена система забезпечує раннє та точне виявлення хвороб, що сприяє прийняттю ефективних агрономічних рішень. Система має потенціал до масштабування на інші культури й може стати основою для створення платформи моніторингу здоров'я рослин у реальному часі. Можливість інтеграції з дронами або камерами смартфонів робить її перспективним інструментом для сучасного точного землеробства.

**Ключові слова:** глибинне навчання; згорткова нейронна мережа; картопля; листок рослини; класифікація; розпізнавання; оптимізатор Adam.

## ПОСТАНОВКА ЗАДАЧІ

Стрімке зростання чисельності населення планети становить серйозну загрозу глобальній продовольчій безпеці. За прогнозами ООН, щоб забезпечити зростаючий попит, світове виробництво продуктів харчування виробництво має зрости щонайменше на 60 % протягом наступних 30 років. У зв'язку з цим актуальним є підвищення продуктивності сільськогосподарських культур та зменшення втрат урожаю, значну частину яких спричиняють захворювання рослин. Згідно з оцінками, захворювання рослин є причиною 20–40 % глобальних втрат урожаю щороку, що становить приблизно 220 млрд. доларів США.

Захворювання рослини – це порушення нормального стану рослини, спричинене впливом біотичних або інфекційних агентів, що призводить до розладу або зміни її життєво важливих функцій. Усі види рослин є вразливими до захворювань. Хоча кожен вид має характерні захворювання, їх загальна кількість є відносно обмеженою. Захворювання рослин негативно впливають на якість продукції, терміни її зберігання, поживну цінність та товарну привабливість, що робить значну частину врожаю непридатною до реалізації.

Особливої уваги потребують культури, що мають вирішальне значення для продовольчої безпеки, зокрема картопля. Значні втрати урожаю цієї культури можуть бути викликані пізньою або некоректною ідентифікацією таких хвороб, як фітофтороз та альтернаріоз. Своєчасне і точне виявлення захворювань рослин є ключовим для вчасного застосування відповідних заходів захисту і, відповідно, зменшення втрат.

Діагностика хвороб рослин здебільшого здійснюється вручну, що є трудомістким, суб'єктивним та схильним до помилок процесом та потребує експертизи у галузі фітопатології. Значний потенціал у цьому контексті демонструють системи автоматизованого розпізнавання хвороб рослин, що базуються на підходах штучного інтелекту та глибинного навчання. Одним із найбільш ефективних інструментів є згорткові нейронні мережі (Convolutional Neural Networks, CNN), які демонструють високу точність у задачах класифікації зображень, зокрема при ідентифікації хвороб культур.

Робота присвячена розробці автоматизованої системи розпізнавання стану рослин на основі CNN на прикладі листя картоплі. Для створення такої системи побудована поетапна модель на основі зображень здорових та уражених листків картоплі, яка здійснює послідовну класифікацію культури, виявлення хвороби та її подальшу ідентифікацію. Такий поступовий підхід покликаний підвищити ефективність і точність систем виявлення хвороб рослин у реальних умовах.

## АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ

Питання розпізнавання та класифікації рослин за допомогою машинного навчання та методів комп'ютерного зору розглядається у роботах багатьох вітчизняних та зарубіжних науковців.

Кумар Н. та ін. [1] представили систему комп'ютерного зору Leafsnap, яка ідентифікує види дерев за фотографіями їхніх листків і вважається однією з найкращих систем розпізнавання рослин. Система забезпечує фільтрацію нерелевантних зображень,

сегментацію листка, вилучення ознак кривизни контуру з бінарзованого зображення та класифікацію методом найближчих сусідів з використанням перетину гістограм як метрики відстані. Завдяки використанню простору HSV та гістограм кривизни на кількох масштабах досягнута висока точність розпізнавання на реальних даних.

Грабар О.І. та ін. [2] провели загальний огляд та аналіз алгоритмів розпізнавання рослин та відмітили переваги поєднання декількох мереж.

Піцек Л. та ін. [3] розглянули та порівняли методи машинного навчання для автоматичного розпізнавання видів рослин на основі зображень, а також запропонували метод розпізнавання на основі пошуку (retrieval-based), який здійснює класифікацію за методом найближчого сусіда в глибоко вбудованому (embedding) просторі. Метод пошуку зображень базується на моделі, навченої за допомогою функції втрат Recall@k.

Як зазначено у [3], передовими підходами до класифікації зображень є системи на основі згорткових нейронних мереж (CNN) та трансформерів для зору (ViT). Порівняння глибоких нейронних класифікаторів демонструє покращення точності класифікації, досягнуте сучасними архітектурами CNN.

Lee S.H. та ін. [4] запропонували систему ідентифікації рослин за ознаками виду та органу на основі вдосконаленої архітектури згорткової нейронної мережі VGG. Передбачено використання спільних шарів для обох типів ознак, а також шарів злиття для їх комбінованого представлення. Навчання здійснюється поетапно: спочатку на органах, потім – на видах, із подальшим об'єднанням для злиття ознак. Автори оцінили ефективність чотирьох підходів до класифікації рослин: попереднє навчання VGG-16 на наборі даних ImageNet2012, навчання з нуля, використання архітектури з високорівневим злиттям ознак та донавчання із застосуванням валідаційного набору. Результати експериментів на наборі даних LifeClef2016 демонструють, що найкращих результатів вдалося досягти при поєднанні попереднього навчання VGG-мережі з донавчанням з використанням валідаційного набору.

Сутаджі Д. та Рошид Х. [5] досліджували класифікацію хвороб сільськогосподарських культур із використанням десяти попередньо навчених моделей CNN. Для кожної моделі було проведено тонке налаштування (fine-tuning) на основі набору даних PlantVillage. Результати експериментів показують, що найкращою моделлю CNN виявилася DenseNet121.

Поєднання декількох універсальних CNN-моделей дозволяє підвищити точність класифікації. Сутаджі Д. та ін. [6] розглянули метод ансамблювання глибинного навчання з ваговими коефіцієнтами для покращення точності класифікації хвороб зернових. Автори комбінують дві та три попередньо навчені

CNN-моделі (зокрема MobileNetV2, DenseNet121 і Xception) з використанням трансферного навчання та оптимізації ваг за допомогою grid search. Результати на наборі PlantVillage показали покращену класифікацію 38 хвороб, а найкращі ансамблі досягли точності 99,49 % і 99,56 %.

Гуменюк Р. та Попович І. [7] провели загальний огляд архітектур глибокого навчання для діагностики захворювань рослин, виділили проблеми використання існуючих методів та доступних наборів даних.

## ВІДОКРЕМЛЕННЯ НЕВИРІШЕНИХ РАНІШЕ ЧАСТИН ЗАГАЛЬНОЇ ПРОБЛЕМИ

Більшість представлених робіт досліджують розпізнавання виду рослин, а не їх стану. Крім того, розглянуті роботи зосереджені на класифікації лише обмеженої кількості типів захворювань, що не дозволяє масштабувати системи для комплексного моніторингу.

Ще одним аспектом, який потребує доопрацювання, є ефективна інтеграція таких систем у прикладні платформи для агровиробників з урахуванням потреб користувача, доступності інтерфейсу та можливості роботи в умовах обмежених обчислювальних ресурсів.

## МЕТА ДОСЛІДЖЕННЯ

Мета статті – розробка та впровадження моделі розпізнавання стану рослин за допомогою згорткових нейронних мереж для автоматичного виявлення захворювань та класифікації їх типу (на прикладі листя картоплі).

## МЕТОДИ, ОБ'ЄКТ, ПРЕДМЕТ ТА МЕТОДИ ДОСЛІДЖЕННЯ

**Методи.** У роботі використовуються методи комп'ютерного зору для попередньої обробки зображень (нормалізація, масштабування, аугментація); глибинне навчання з використанням згорткових нейронних мереж; методи машинного навчання для класифікації зображень; емпіричні методи оцінювання точності моделі на тестових даних.

**Об'єкт дослідження** – виявлення та розпізнавання захворювань рослин на основі аналізу зображень листя.

**Предмет дослідження** – використання методів глибинного навчання, зокрема згорткових нейронних мереж (CNN) для автоматизованої ідентифікації хвороб сільськогосподарських культур.

**Метою дослідження** є побудова моделі глибинного навчання для ранньої діагностики захворювань листя картоплі, а також її інтеграція у веб-додаток для практичного застосування.

## ОСНОВНИЙ МАТЕРІАЛ

### Глибинне навчання для розпізнавання стану.

Глибинне навчання, як напрям машинного навчання, імітує здатність людини навчатися на основі прикладів, що робить його ефективним у завданнях

розпізнавання зображень. В аграрних технологіях ці методи дозволяють точно і швидко розпізнавати стан рослин із зображень, виявляючи ознаки захворювань або нестачі поживних речовин. Глибокі нейронні мережі забезпечують високу точність класифікації, іноді навіть перевершуючи результати експертної оцінки.

Незважаючи на те, що теоретичні засади глибокого навчання були сформовані ще у 1980-х роках, його практичне застосування в агросекторі почало активно розвиватися лише нещодавно. Причини цього – поява великих масивів зображень рослин із точною розміткою і розвиток апаратного забезпечення. Наприклад, для навчання моделі, що виявляє ознаки фітохвороб, потрібно багато зображень різних культур у різних фазах розвитку. Важливу роль відіграє й обчислювальна інфраструктура. Використання графічних процесорів та хмарних сервісів дозволяє значно прискорити процес навчання нейронної мережі – з кількох тижнів до кількох годин або навіть хвилин.

Вибір моделі глибокого навчання залежить від специфіки завдання. Кожна архітектура має свої переваги, що зумовлює доцільність її застосування для певних завдань. RNN демонструють високу ефективність в обробці послідовних даних, DBN – у задачах навчання без вчителя, GAN – у генерації даних, CNN – в обробці зображень, ANN вирізняються універсальністю тощо.

У сфері обробки зображень найкращі результати демонструють згорткові нейронні мережі (CNN). Їх здатність ефективно працювати з сітчастими структурами даних, такими як зображення, пояснює широку популярність CNN у задачах класифікації та розпізнавання об'єктів.

**Згорткова нейронна мережа** – це тип вдосконаленої нейронної мережі, спеціально розробленої для обробки та аналізу візуальних даних, зокрема зображень. Особливість CNN полягає в її здатності автоматично виділяти релевантні ознаки зображень завдяки процесу згортки, що дозволяє уникнути ручного виділення ознак та підвищити точність моделі.

Ключові переваги CNN: інваріантність до трансляцій, що дозволяє розпізнавати об'єкти незалежно від їхнього положення на зображенні; ієрархічне

навчання, завдяки якому модель поступово формує уявлення про складні ознаки; ефективне масштабування завдяки можливості паралельного навчання. Недоліками є обмежені можливості обробки послідовних даних та значні обчислювальні витрати через велику кількість параметрів. Незважаючи на вказані недоліки, CNN є провідним інструментом розпізнавання та класифікації зображень.

Зазвичай CNN складається з послідовного розташування шарів згортки, шарів об'єднання та функцій активації, які завершуються звичайними шарами (рис. 1). Включення згорткових шарів і шарів об'єднання допомагає досягти підвищеної точності на відміну від використання лише звичайних шарів повного зв'язку. По мірі того, як дані проходять через кожен шар, CNN удосконалюється у розпізнаванні різних частин зображення. Перші шари помічають основні риси, і в міру просування вони починають об'єднувати ці риси, щоб ідентифікувати більші форми або об'єкти, поки нарешті не з'ясують, що це за зображення в цілому. Послідовна структура цих шарів у CNN полегшує вилучення складних ознак із зображень, що робить її особливо корисною для класифікації зображень.

Серед прихованих шарів, призначених для вилучення значущої інформації з зображень, вирішальну роль відіграють чотири ключові компоненти:

1. Шар згортки (Convolution) відповідає за виявлення закономірностей та особливостей вхідного зображення за допомогою операцій згортки. Фільтри, або невеликі матриці, ковзають по вхідному зображенню для виявлення специфічних характеристик, таких як краї або текстури.

2. Шар ReLU (Rectified Linear Unit – випрямлена лінійна одиниця) вносить нелінійність у мережу. Він застосовує функцію активації, що перетворює від'ємні значення пікселів на нуль, а додатні значення залишає незмінними. Ця нелінійна активація допомагає мережі виявляти складні взаємозв'язки в даних.

3. Шар об'єднання (Pooling) зменшує просторові розміри даних, зберігаючи важливу інформацію. Він передбачає зменшення вибірки шляхом вибору максимального або середнього значення з групи пікселів.

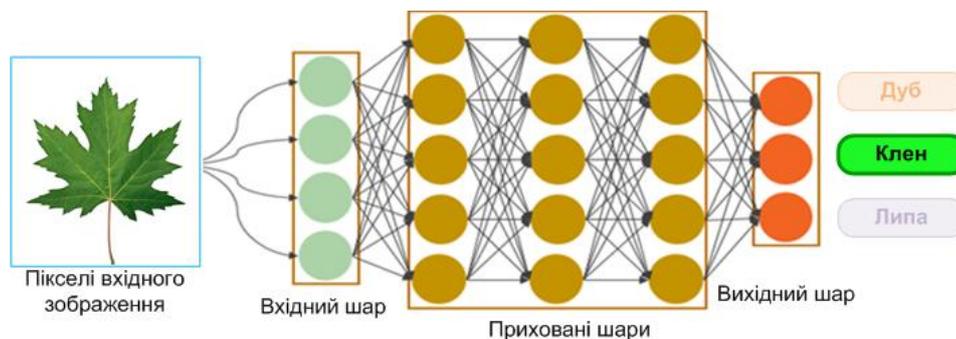


Рис. 1. Структура згорткової нейронної мережі [8]

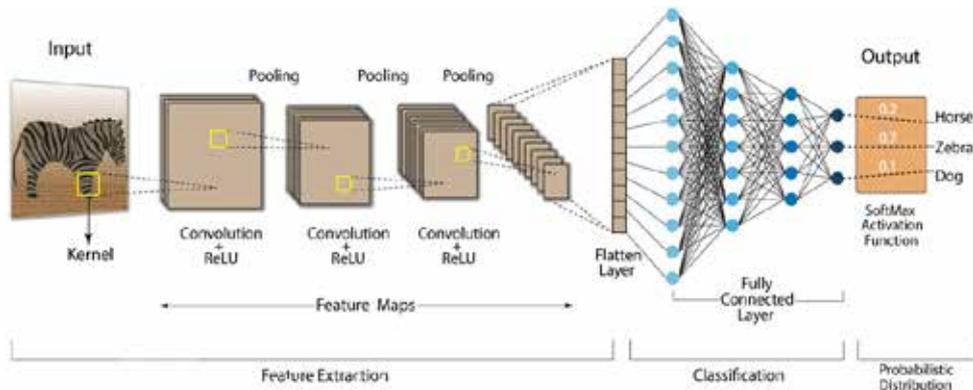


Рис. 2. Згорткові нейронної мережі [8]

Це допомагає зменшити обчислювальну складність і підвищити здатність мережі розпізнавати ознаки, інваріантні до невеликих змін в позиції об'єктів.

4. Шар повного зв'язку (Fully Connected Layer) інтегрує інформацію, отриману з попередніх шарів. Нейрони цього шару з'єднані з усіма нейронами попереднього шару, що дозволяє мережі врахувати всі вивчені ознаки для класифікації.

Ці чотири шари працюють спільно, кожен вносячи свій внесок у ієрархічне вилучення ознак із вхідного зображення. Їх поєднання дозволяє згортковим нейронним мережам ефективно обробляти та інтерпретувати візуальну інформацію завдань класифікації зображень.

Класифікація – це процес у контрольованому машинному навчанні, коли заданий набір вхідних даних сортується за категоріями на основі однієї або декількох змінних. У машинному навчанні існують різні типи класифікації залежно від кількості заданих категорій:

1. Бінарна класифікація – фундаментальне завдання, мета якого – віднести дані до одного з двох класів, наприклад, є хвороба чи немає.

2. Багатокласова класифікація передбачає класифікацію даних на більш ніж два класи. Передбачається, що об'єкт може належати лише до одного класу, наприклад, класифікація хвороб рослин (ранній фітофтороз, пізній фітофтороз тощо).

3. Багатоміткова класифікація дозволяє призначити декілька невиключних міток кожному екземпляру. Наприклад, рослина може мати декілька хвороб одночасно.

4. Незбалансована класифікація виконується для набору даних, в якому порушені пропорції класів. Такий сценарій є поширеним при роботі з рідкісними подіями або аномаліями.

У даному проекті при виконанні класифікації алгоритм генерує оцінку ймовірності віднесення даних до певної категорії.

#### Вибір ключових технологій.

Для розробки системи були обрані такі основні технології:

1. Python та Node.js: При розробці проекту використовуються переваги як Python, так і Node.js. Python, завдяки своїй читабельності та широкому набору бібліотек, є основною мовою для розробки бекенду, аналізу даних і завдань машинного навчання. Node.js, у свою чергу, забезпечує роботу серверного додатку, надаючи масштабоване та подіє-орієнтоване середовище для обробки даних в реальному часі та відповідне API.

2. npm (Node Package Manager): Для спрощення управління залежностями в проекті на Node.js та ефективного керування пакетами використовуємо npm; npm дозволяє знаходити, встановлювати та управляти сторонніми бібліотеками та інструментами, що особливо корисно для забезпечення узгодженості та надійності компонентів Node.js проекту.

3. FastAPI та uvicorn: FastAPI, побудований на Python, відіграє важливу роль у швидкому розробленні API з акцентом на високу продуктивність. Завдяки автоматичному генеруванню OpenAPI та JSON Schema, FastAPI спрощує процес розробки API. Uvicorn, обраний ASGI сервер, доповнює FastAPI, забезпечуючи ефективне асинхронне оброблення веб-запитів.

4. numpy та TensorFlow: Можливості numpy для маніпулювання масивами покращують процеси обробки даних, тоді як TensorFlow надає змогу створювати, навчати і розгортати моделі машинного навчання.

5. io та BytesIO: Модуль io разом із BytesIO є незамінними для обробки операцій введення/виведення та управління бінарними даними. BytesIO, зокрема, дозволяє ефективно працювати з бінарними даними в пам'яті, що критично важливо для оптимізації операцій з файлами в межах проекту.

6. Pillow та Matplotlib: Використання Pillow дозволяє обробляти зображення для таких завдань, як зміна розміру зображень, фільтрація та робота з різними форматами зображень. Крім того, Matplotlib надає можливості візуалізації даних, дозволяючи створювати інформативні та візуально привабливі графіки та діаграми.

7. Jupyter: Jupyter notebooks є невід’ємною частиною проекту, забезпечуючи інтерактивне середовище для спільного кодування, дослідження даних та експериментів.

Інтеграція цих технологій покращує процеси розробки, аналізу та візуалізації в проекті, забезпечуючи надійне і масштабоване рішення.

### Основні етапи розпізнавання стану рослин:

**1. Збір даних.** Будь-який проект машинного навчання починається з процесу збору даних, які використовуються як навчальний набір даних. Для отримання даних необхідно купувати їх у сторонніх постачальників, збирати самостійно та класифікувати, створювати скрипти для збору даних або використовувати дані з відкритих джерел, таких як PlantVillage, iNaturalist, Flora Incognita тощо [9, 10]. У роботі [3] наводиться перелік доступних наборів даних для проведення класифікації рослин.

У цьому проекті використовуються дані з сайту Kaggle [11].

**2. Попередня обробка даних.** Стан даних змінюється в процесі їх проходження через модель. Після збору даних перший етап включає очищення та попередню обробку даних, де сирі дані готуються до аналізу. Під час цього етапу використовується функціональність набору даних у TensorFlow та застосовуються техніки доповнення даних.

Кожен елемент набору даних є кортежем. Перша складова містить пакет з 32 елементів зображень, а друга – пакет з 32 елементів міток класів.

Розмір пакету визначає кількість зображень, що обробляються в кожній ітерації. Стандартне зображення має розмір 256x256 пікселів та формат RGB, що складається з трьох кольорових каналів.

Обробка зображень пакетами є ресурсоефективним підходом. Це особливо корисно при роботі з великими наборами даних, які можуть перевищувати доступну оперативну пам’ять системи. Розбиваючи набір даних на керовані пакети, система може ефективно використовувати ресурси, не перевантажуючи пам’ять. Така стратегія пакетної обробки



Рис. 3. Візуальне зображення зображень з відповідними мітками класів

сприяє більш ефективній обробці всього набору даних.

Для налаштування пакетної обробки вхідних даних використовується Keras. Зображення організовано в окремі директорії, кожна з яких відповідає певному класу, назва директорії слугує міткою класу для зображень у ній. Візуалізація зображень разом з відповідними мітками класів створюється за допомогою Matplotlib (див. рис. 3).

Перед введенням зображення в модель його розмір автоматично змінюється до потрібних параметрів. Зміна розміру зображення до (256, 256) є важливою, оскільки при використанні моделі для класифікації може бути надано зображення іншого розміру.

Для підвищення продуктивності моделі нормалізуються значення пікселів зображень. Нормалізація повинна здійснюватися під час навчання і під час класифікації. Тому ці кроки включаються як шари в модель.

Після нормалізації виконується доповнення даних, яке дозволяє створювати додаткові зображення шляхом виконання операцій, таких як обертання або коригування контрасту. Ці доповнені зображення потім додаються до вхідного набору. Доповнення даних є особливо корисним при роботі з невеликими наборами даних, оскільки воно підвищує точність моделі, розширюючи та урізноманітнюючи наявний набір даних.

Після підготовки даних набір даних ділиться на три окремі підмножини: навчальну, валідаційну та тестову. Навчальна множина використовується для безпосереднього навчання моделі, валідаційна – під час зворотного розповсюдження для налаштування її параметрів, тестова – для оцінки роботи моделі після завершення навчання (рис. 4).

Для поділу даних на відповідні набори створено функцію. Ця функція приймає первинний набір даних як вхідний параметр і встановлює значення за замовчуванням для параметрів, що стосуються розподілу на навчальний, валідаційний і тестовий

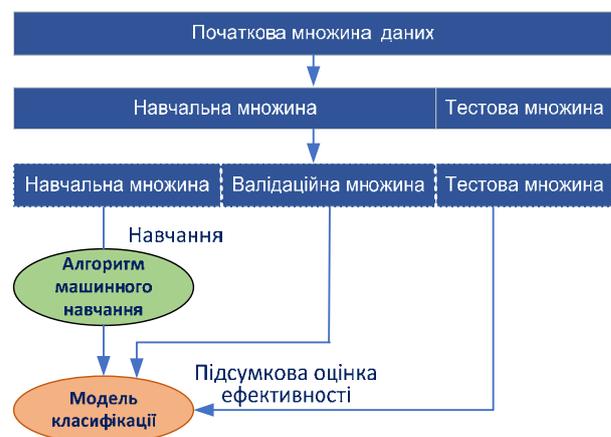
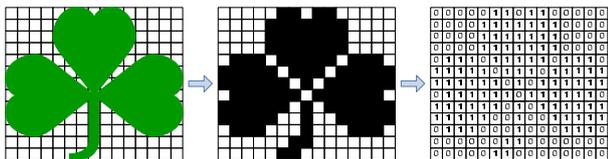


Рис. 4. Поділ даних на підмножини в моделі

набори. У функцію додано параметр перемішування, який дозволяє рандомізувати зображення в наборі даних.

**3. Моделювання.** Після попередньої обробки даних виконується побудова моделі на основі цих даних.

Кожне зображення представляється у вигляді масиву значень пікселів:



**Рис. 5.** Представлення зображення у згортковій нейронній мережі

Для класифікації зображень використовується згорткова нейронна мережа (CNN) з активацією Softmax у вихідному шарі. Модель складається з кількох шарів. Початкові шари призначені для зміни розміру, нормалізації та доповнення даних. Шари для зміни розміру забезпечують однорідність вхідних даних приведенням їх до стандартних розмірів, нормалізація підвищує стабільність моделі, а доповнення даних сприяє підвищенню надійності та покращенню узагальнення за рахунок

диверсифікації набору навчальних даних. Для вихідного шару обирається функція активації Softmax, оскільки вона найкраще підходить для багатокласової класифікації, забезпечуючи розподіл ймовірностей між різними класами.

Архітектура розробленої мережі у TensorFlow/Keras, яка використана для класифікації стану рослин, представлена на рис. 6.

**4. Оптимізація та навчання моделі.** Оптимізатори – це алгоритми, що використовуються для зменшення втрат шляхом налаштування ваг та зміщень.

Для ефективного навчання моделі використовується оптимізатор Адам, один з найпоширеніших алгоритмів тренування нейронних мереж. Він значно прискорює процес навчання без великого навантаження на комп'ютерну потужність, особливо при роботі з великими масивами даних. Адам враховує не тільки поточні, а й попередні градієнти, тобто вчиться на власному досвіді. Крім того, він має можливість змінювати швидкість навчання в процесі роботи, що дозволяє йому адаптуватися до ситуації. Поєднання двох ключових аспектів – навчання на основі попереднього досвіду та адаптація на льоту – робить Адама найкращим вибором для навчання моделей у машинному навчанні.

Для оцінки ефективності моделі використовуються такі метрики, як точність і втрата. Для обчислення втрат у багатокласовій класифікації застосовується функція SparseCategoricalCrossentropy.

**5. Оцінка та валідація моделі.** Після навчання моделі її точність оцінюється за допомогою тестового набору даних для перевірки її здатності до узагальнення на нових зображеннях. Валідаційна множина використовується під час навчання для налаштування параметрів моделі та забезпечення її ефективності.

**6. Розгортання та застосування моделі.** Після навчання, оцінки та доведення ефективності модель може бути використана в реальних додатках.

Для роботи з моделлю був розроблений веб-додаток з використанням FastAPI для бекенду. FastAPI є перспективним Python-фреймворком для створення ефективних та масштабованих REST API веб-додатків.

У рамках проекту було створено REST API інтерфейс, який дозволяє користувачу завантажувати зображення та забезпечує їх обробку моделлю. Система класифікує зображення, надаючи ймовірності для кожного класу, а результат класифікації відображається разом із асоційованими ймовірностями.

Фрагменти інтерфейсу веб-додатку для розпізнавання стану листя картоплі для здорового та хворих листків наведено на рис. 7.

FastAPI полегшує завдання класифікації зображень завдяки своїй простоті та зручності інтеграції функціоналу завантаження файлів.

```
model.summary()
Model: "sequential_2"
Layer (type)                Output Shape                Param #
-----
sequential (Sequential)     (32, 256, 256, 3)          0
conv2d (Conv2D)              (32, 254, 254, 32)         896
max_pooling2d (MaxPooling2D) (32, 127, 127, 32)        0
conv2d_1 (Conv2D)            (32, 125, 125, 64)         18496
max_pooling2d_1 (MaxPooling2 (32, 62, 62, 64)          0
conv2d_2 (Conv2D)            (32, 60, 60, 64)           36928
max_pooling2d_2 (MaxPooling2 (32, 30, 30, 64)          0
conv2d_3 (Conv2D)            (32, 28, 28, 64)           36928
max_pooling2d_3 (MaxPooling2 (32, 14, 14, 64)          0
conv2d_4 (Conv2D)            (32, 12, 12, 64)           36928
max_pooling2d_4 (MaxPooling2 (32, 6, 6, 64)            0
conv2d_5 (Conv2D)            (32, 4, 4, 64)              36928
max_pooling2d_5 (MaxPooling2 (32, 2, 2, 64)            0
flatten (Flatten)            (32, 256)                    0
dense (Dense)                 (32, 64)                      16448
dense_1 (Dense)               (32, 3)                         195
-----
Total params: 183,747
Trainable params: 183,747
Non-trainable params: 0
```

**Рис. 6.** Архітектура нейронної мережі для класифікації стану рослин

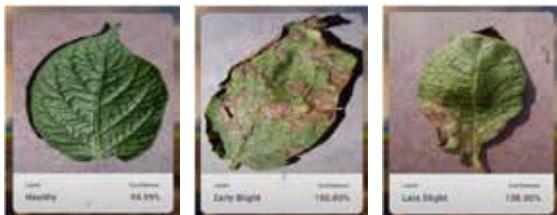


Рис. 7. Приклади результатів розпізнавання стану листя картоплі

**Продуктивність моделі.** Розглянемо результати навчання, оцінювання та передбачення моделі розпізнавання стану рослин на прикладі листя картоплі.

Фрагмент коду, що ілюструє процес навчання моделі, представлено на рис. 8. Модель навчалась протягом 50 епох, з використанням зворотного поширення помилки для оновлення параметрів. На кожній епосі виконувались валідаційні кроки з використанням матриці точності для оцінки ефективності.

```
history = model.fit(
    train_ds,
    batch_size=BATCH_SIZE,
    validation_data=val_ds,
    verbose=1,
    epochs=50,
)

Epoch 1/50
54/54 [#####] - 265s 4s/step - loss: 0.8145 - accuracy: 0.4093 - val_loss: 0.8283 - val_accuracy: 0.5184
Epoch 2/50
33/34 [#####] - 374s 3s/step - loss: 0.7984 - accuracy: 0.5487
...
Epoch 46/50
54/54 [#####] - 385s 3s/step - loss: 0.8087 - accuracy: 0.8965 - val_loss: 0.8968 - val_accuracy: 0.8948
Epoch 47/50
54/54 [#####] - 374s 3s/step - loss: 0.8508 - accuracy: 0.8792 - val_loss: 0.8865 - val_accuracy: 0.8748
Epoch 48/50
54/54 [#####] - 369s 3s/step - loss: 0.8394 - accuracy: 0.8858 - val_loss: 0.8132 - val_accuracy: 0.8934
Epoch 49/50
54/54 [#####] - 376s 3s/step - loss: 0.8253 - accuracy: 0.8923 - val_loss: 0.8132 - val_accuracy: 0.8936
Epoch 50/50
54/54 [#####] - 383s 3s/step - loss: 0.8139 - accuracy: 0.8878 - val_loss: 0.8474 - val_accuracy: 0.8844
```

Рис. 8. Фрагмент процесу навчання моделі

Після завершення навчання було проведено оцінювання моделі, результати якого наведено на рис. 9. Модель досягла точності 95 %, що свідчить про її здатність класифікувати вхідні зображення.

```
test_loss, test_acc = model.evaluate(test_ds)
0/0 [#####] - 6s 758ms/step - loss: 0.1183 - accuracy: 0.9533
```

Рис. 9. Результати оцінювання моделі (точність 95 %)

На рис. 10 представлено динаміку зміни точності та втрат під час навчання і валідації моделі. Модель демонструє поступове покращення обох показників, що свідчить про стабільне навчання без перенавчання.

Для здійснення передбачень реалізована функція, яка приймає модель та зображення як вхідні параметри. Результатом є масив ймовірностей для кожного класу, з якого обирається клас із найвищою

ймовірністю. На рис. 11 представлено результати передбачення для кількох зразків.

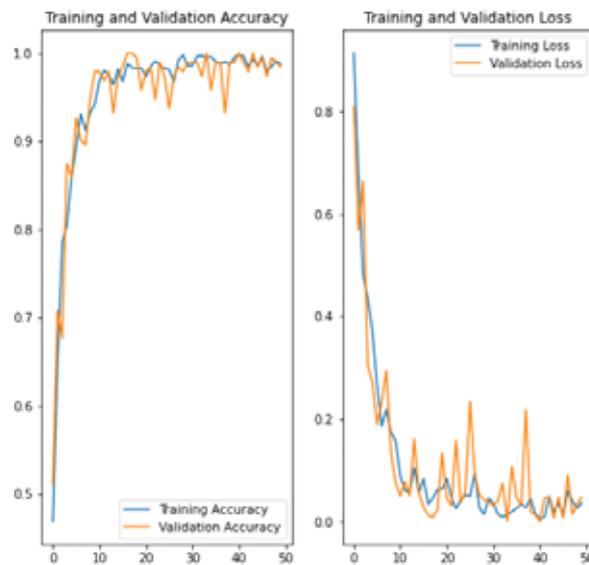


Рис. 10. Динаміка точності навчання та валідації

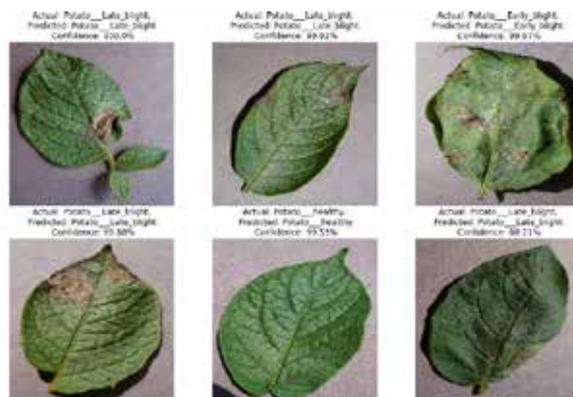


Рис. 11. Зразки передбачення стану листка

Як видно, модель правильно і впевнено класифікує захворювання листків, супроводжуючи кожне передбачення високим коефіцієнтом впевненості.

Таким чином, розроблена модель досягає точності 95 %, демонструючи здатність правильно класифікувати дані у широкому спектрі сценаріїв. Крім того, модель демонструє високу розподільчу здатність, успішно розрізняючи різні класи об'єктів, що підтверджує її ефективність у складних завданнях класифікації.

**ВИСНОВКИ**

У ході дослідження розроблено модель ідентифікації стану рослин на основі зображень їх листя із використанням згорткової нейронної мережі (CNN). Застосування CNN обумовлено її високою ефективністю в задачах обробки зображень, здатністю до автоматичного виділення ознак та адаптивністю до різних задач класифікації.

Для оптимізації процесу навчання моделі застосовано алгоритм Adam, який забезпечує динамічну зміну швидкості навчання та зменшення впливу шуму завдяки використанню експоненційно зважених середніх значень. Розроблену модель інтегровано у веб-додаток, який, після незначної модифікації, може забезпечити автоматизований моніторинг стану рослин. Це створює передумови для своєчасного виявлення хвороб і впровадження профілактичних заходів.

Результати дослідження демонструють ефективність глибокого навчання та комп'ютерного зору для вирішення прикладних завдань в аграрному секторі. Розроблене рішення має перспективи масштабування на інші культури та може бути основою для створення систем моніторингу стану посівів у режимі реального часу, що сприятиме оперативному реагуванню на загрози та підвищенню урожайності.

## REFERENCES

- [1] Kumar, N., Belhumeur, P. N., Biswas, A. et al. (2012). Leafsnap: a computer vision system for automatic plant species identification. *Computer Vision – ECCV*, Springer, pp. 502–516.
- [2] Hrabar, O. I., Lysogor, Yu. I., Skachkov, V. O. (2020). Analiz alhorytmiv vizualnoi systemy rozpoznavannia sortiv roslyn dlia stvorennia veb-servisu [Analysis of algorithms of a visual system for plant variety recognition for the creation of a web service]. *Vcheni zapysky TNU imeni V. I. Vernadskoho. Serii: Tekhnichni nauky*, vol. 31 (70), part. 1, № 2, pp. 80–84. DOI: <https://doi.org/10.32838/2663-5941/2020.2-1/12>.
- [3] Picek, L., Šulc, M., Patel, Y., Matas, J. (2022). Plant recognition by AI: Deep neural nets, transformers, and kNN in deep embedding. *Frontiers in Plant Science*, vol. 13., 16 pp. DOI: <https://doi.org/10.3389/fpls.2022.787527>.
- [4] Lee, S. H., Chang, Y. L., Chan, C. S., Remagnino, P. (2016). Plant Identification System based on a Convolutional Neural Network for the LifeClef 2016 Plant Classification Task. *Conference and Labs of the Evaluation Forum*. Retrieved from: <https://ceur-ws.org/Vol-1609/16090502.pdf>.
- [5] Sutaji, D., Rosyid, H. (2022). Convolutional Neural Network (CNN) Models for Crop Diseases Classification. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, vol. 7, № 2, pp. 187–196. DOI: <https://doi.org/10.22219/kinetik.v7i2.1443>.
- [6] Sutaji, D., Yildiz, O., Rosyid, H., Chotijah, U. (2024). WECNN-PDP: Weighted Ensemble Convolutional Neural Networks Models to Improve the Plant Disease Prediction. *E3S Web of Conferences*, vol. 482. 8 pp. DOI: <https://doi.org/10.1051/e3sconf/202448203003>.
- [7] Humeniuk, R., Popovych, I. (2024). Doslidzhennia metodiv diahnozyky zakhvoriuvan roslyn za dopomohoiu hlybokoho navchannia [Research of plant disease diagnostics methods using deep learning]. *Komp'uterni systemy proiektuvannia. Teoriia i praktyka*, vol. 6, № 1, pp. 37–48. DOI: <https://doi.org/10.23939/cds2024.01.037>.
- [8] CNN in Deep Learning: Algorithm and Machine Learning Uses. Retrieved from: <https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network>.
- [9] PlantVillage. Retrieved from: <https://plantvillage.psu.edu>
- [10] iNaturalist. Retrieved from: <https://www.inaturalist.org>
- [11] PlantVillage Dataset. Dataset of diseased plant leaf images and corresponding labels. Retrieved from: <https://www.kaggle.com/datasets/emmarex/plantdisease>

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] Kumar, N., Belhumeur, P. N., Biswas, A. та ін. (2012) Leafsnap: a computer vision system for automatic plant species identification. *Computer Vision – ECCV*. Springer. Pp. 502–516.
- [2] Грабар, О. І., Лисогор, Ю. І., Скачков, В. О. (2020) Аналіз алгоритмів візуальної системи розпізнавання сортів рослин для створення веб-сервісу. *Вчені записки ТНУ імені В. І. Вернадського. Серія: Технічні науки*. Т. 31(70). Ч. 1. № 2. С. 80–84. DOI: <https://doi.org/10.32838/2663-5941/2020.2-1/12>.
- [3] Picek, L., Šulc, M., Patel, Y., Matas, J. (2022) Plant recognition by AI: Deep neural nets, transformers, and kNN in deep embedding. *Frontiers in Plant Science*. Vol. 13. 16 pp. DOI: <https://doi.org/10.3389/fpls.2022.787527>.
- [4] Lee, S. H., Chang, Y. L., Chan, C. S., Remagnino, P. (2016) Plant Identification System based on a Convolutional Neural Network for the LifeClef 2016 Plant Classification Task. *Conference and Labs of the Evaluation Forum*. URL: <https://ceur-ws.org/Vol-1609/16090502.pdf>.
- [5] Sutaji, D., Rosyid, H. (2022) Convolutional Neural Network (CNN) Models for Crop Diseases Classification. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*. Vol. 7. № 2. Pp. 187–196. DOI: <https://doi.org/10.22219/kinetik.v7i2.1443>.
- [6] Sutaji, D., Yildiz, O., Rosyid, H., Chotijah, U. (2024) WECNN-PDP: Weighted Ensemble Convolutional Neural Networks Models to Improve the Plant Disease Prediction. *E3S Web of Conferences*. 482. P. 8. DOI: <https://doi.org/10.1051/e3sconf/202448203003>.
- [7] Гуменюк, Р., Попович, І. (2024) Дослідження методів діагностики захворювань рослин за допомогою глибокого навчання. *Комп'ютерні системи проєктування. Теорія і практика*. Т. 6. № 1. С. 37–48. DOI: <https://doi.org/10.23939/cds2024.01.037>.

- [8] CNN in Deep Learning: Algorithm and Machine Learning Uses. URL: <https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network>.
- [9] PlantVillage. URL: <https://plantvillage.psu.edu>.
- [10] iNaturalist. URL: <https://www.inaturalist.org>.
- [11] PlantVillage Dataset. Dataset of diseased plant leaf images and corresponding labels. URL: <https://www.kaggle.com/datasets/emmarex/plantdisease>.

---

© Гайтан О. М., Альошин С. П., Луаттар І.

Дата надходження статті до редакції: 02.05.2025

Дата затвердження статті до друку: 15.05.2025