

ДОДАТОК А

Перелік джерел посилання за науковими напрямками керівника та науковців
кафедри програмної інженерії

23. K. Smelyakov, A. Chupryna, O. Bohomolov and N. Hunko, "The Neural Network Models Effectiveness for Face Detection and Face Recognition," 2021 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2021, pp. 1-7, doi: 10.1109/eStream53087.2021.9431476

25. K. Smelyakov, D. Tovchyrechko, I. Ruban, A. Chupryna and O. Ponomarenko, "Local Feature Detectors Performance Analysis on Digital Image," 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 2019, pp. 644-648, doi: 10.1109/PICST47496.2019.9061331. (<https://ieeexplore.ieee.org/document/9061331>).

ДОДАТОК Б

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ



Ім'я користувача:
Кардаш Євген Вікторович каф.ПІ
Дата перевірки:
08.06.2024 17:18:32 EEST
Дата звіту:
08.06.2024 17:26:50 EEST

ID перевірки:
1016335657
Тип перевірки:
Doc vs Internet + Library
ID користувача:
100013622

Назва документа: 2024_М_ПІ_ІПЗм-22-6_Чайковський_С_О_скорочений

Кількість сторінок: 34 Кількість слів: 5198 Кількість символів: 39431 Розмір файлу: 1.52 MB ID файлу: 1016136338

3.64%
Схожість

Найбільша схожість: 0.58% з Інтернет-джерелом (<https://habr.com/ru/articles/448316>)

3.21% Джерела з Інтернету 88 Сторінка 36

1.79% Джерела з Бібліотеки 14 Сторінка 37

0% Цитат

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

0%
Вилучень

Немає вилучених джерел

ДОДАТОК В

Слайди презентації

ДОСЛІДЖЕННЯ МЕТОДІВ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ЗА КОНТЕНТОМ

Виконав:
ст. гр. ПЗМ-22-6 Чайковський С.О.

Керівник:
проф. Смяляков С.В.

Вступ

На сьогоднішній день проблема автоматизованого аналізу та класифікації зображень набуває вирішального значення. У даному дослідженні основна увага приділена тепличному господарству, яке відіграє вирішальну роль у задоволенні попиту протягом року, незалежно від кліматичних умов.

Важливою проблемою є підтримання оптимального здоров'я врожаю. Більше того, враховуючи новий Європейський кліматичний закон, який накладає цілі щодо зменшення використання та ризику хімічних пестицидів на 50% у 2030 році, оптимізація використання ресурсів для кожної одиниці культури стає обов'язковою.

У цій роботі представлено класифікатор гібридної нейронної мережі (HNN), керований штучним інтелектом, розроблений для ефективного виявлення хвороб і бур'янів у тепличному вирощуванні томатів. Цей автоматизований інструмент допоможе зробити виявлення хвороб і бур'янів своєчасним, економічно ефективним і безпечнішим для навколишнього середовища.

Опис дослідження

01

Предмет - процеси виявлення хвороб та бур'янів у теплицях при вирощуванні томатів за допомогою гібридного нейронно мережевого класифікатора.

02

Мета - експериментальна оцінка ефективності гібридного нейронно мережевого класифікатора для виявлення хвороб та росту бур'янів у теплиці томатів.

03

Актуальність - обумовлена постійним попитом на помідори в усьому світі та необхідністю оптимізації процесу їх вирощування та відповідає сучасним викликам в обробці великих обсягів візуальних даних.

Аналіз предметної галузі

Мета штучного інтелекту - розробка алгоритмів та програм, які дозволяють комп'ютерам аналізувати дані, розуміти контекст, вчитися з досвіду, приймати рішення та вирішувати завдання в реальному часі.

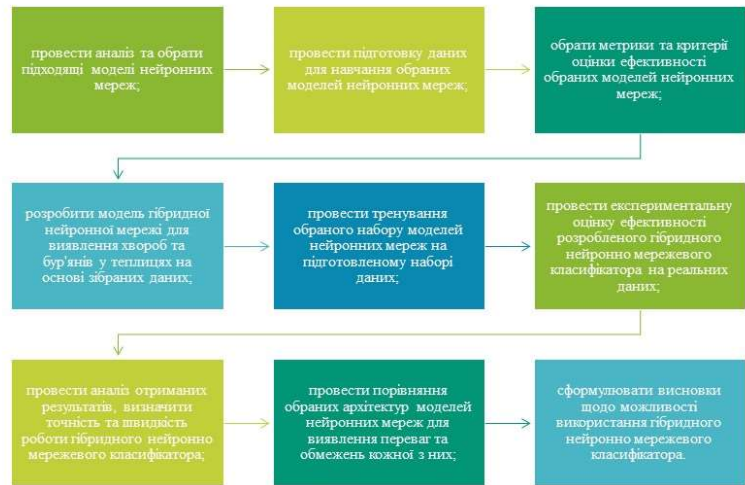
У даній роботі використані такі напрямки:

Згорткові нейронні мережі (CNN) – оптимізовані для обробки зображень та використовують згорткові шари для виявлення локальних характеристик.

Машинне навчання (ML) – підгалузь штучного інтелекту, яка дозволяє системам навчатися на основі даних, розпізнавати закономірності та приймати рішення без явного програмування.

Комп'ютерний зір (CV) – технологія, що надає комп'ютерам можливість розпізнавати, аналізувати та реагувати на візуальні дані, такі як зображення і відео.

Постановка задачі



Опис прийнятих проєктних рішень

Складові

- виявити листя рослин;
- виявити потенційні бур'яни;
- виявити потенційно можливі захворювання.

Реалізації

- YOLOv8
- VGG19;
- MobileNet.

Підготовка даних до проведення експерименту

Для виявлення листя та правильного обрізання зображення було взято 2000 зображень із спеціального набору даних із анотаціями. Він складається із зображень рослин у ґрунті. Набір даних складається з 1400 зображень поїздів і 600 зображень перевірки.

Для виявлення бур'янів використовувався набір даних V2 Plant Seedlings. Цей набір даних містить 5539 зображень сходів культур і бур'янів. Зображення згруповані в 12 класів, як показано на малюнках вище. Ці класи представляють поширені види рослин у сільському господарстві Данії. Кожен клас містить зображення RGB, які показують рослини на різних стадіях росту. Зображення мають різні розміри та формат PNG.

Для виявлення хвороб листя використовувався набір даних виявлення захворювань листя томатів. У даних є різні види хвороб листя томатів. Навчальний набір даних містить 10000 мічених зображень, а дані тестування/оцінки містять 1000 мічених зображень.

Вибір метрик та критеріїв порівняння

Точність - співвідношення правильно класифікованих зображень до загальної кількості зображень. Це добре працює для збалансованих наборів даних, де всі класи мають приблизно однакову кількість зразків.

Функції втрат - обчислює, наскільки прогнози моделі відхиляються від бажаного результату. Зводячи до мінімуму функцію втрат під час навчання, модель навчається коригувати свої внутрішні параметри, щоб покращити свою ефективність у виконанні завдання.

Засоби проведення дослідження

Дане магістерське дослідження буде виконане з використанням мови програмування Python, програмний код буде написаний за допомогою IDE Pycharm.

TensorFlow – відкрите програмне забезпечення для машинного навчання і глибокого навчання, розроблене Google.

Keras – високорівневий інтерфейс для роботи з нейронними мережами, який працює поверх бібліотеки глибокого навчання TensorFlow.

Roboflow Inference – це платформа з відкритим кодом, призначена для спрощення розгортання моделей комп'ютерного зору.

Планування експерименту

У нашому дослідженні ми використовували попередньо навчену модель YOLOv8 для виявлення листя.

Для класифікації зображень для виявлення бур'янів і хвороб ми навчили моделі VGG19 і MobileNet.

Обидва етапи навчання використовували попередньо підготовлені ваги ImageNet.

Після цього, використовуючи тестовий набір даних, ми виміряли точність, прецизійність, запам'ятовування та середню точність. Дані, отримані під час тренінгів, відображалися на графіках.

Опис програмної реалізації

```

# load a pre-trained yolov8n model
model = inference.get_roboflow_model(model_id="lettuce-weed-4/1")

# run inference on chosen image
results = model.infer(
    image,
    confidence=0.1,
    iou_threshold=0.0,
)

# load the results into the supervision Detections api
detections = sv.Detections.from_inference(results[0].dict(
    by_alias=True,
    exclude_none=True
))

train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    validation_split=0.2, # set the validation split
    rotation_range=15,
    shear_range=0.2,
    zoom_range=0.2,
    width_shift_range=0.1,
    height_shift_range=0.1,
    horizontal_flip=True)

base_model = VGG19(
    weights='imagenet',
    include_top=False,
    input_shape=(224, 224, 3))
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(1024, activation='relu')(x)
outputs = Dense(train_generator.num_classes,
    activation='softmax')(x)
model = Model(inputs=base_model.input, outputs=outputs)

opt = Nadam(learning_rate=0.001)
model.compile(optimizer=opt,
    loss='categorical_crossentropy',
    metrics=['accuracy'])
model.fit(
    train_generator,
    validation_data=validation_generator,
    epochs=10,
    verbose=1)

```

Архітектурна реалізація на основі хостингової платформи AWS

Для архітектури прикладного рішення було використано стиль мікросервісів, який забезпечує гнучкість, зручність обслуговування, масштабованість і можливість ефективного розміщення в хмарі.

В якості хостингової платформи було обрано AWS Cloud.

Для досягнення високої продуктивності було вибрано мультирегіональне налаштування.

Розгортання нової системи може бути реалізовано шляхом синьо-зеленого розгортання.

Прикладне рішення складається з кількох взаємопов'язаних мікросервісів, які працюють разом, щоб забезпечити ефективне завантаження та аналіз фотографій. Кожен мікросервіс відповідає за певний набір функцій і може бути розроблений, розгорнутий і масштабований незалежно.

інтерфейс зовнішнього користувача

служба керування об'єктовими записами користувачів

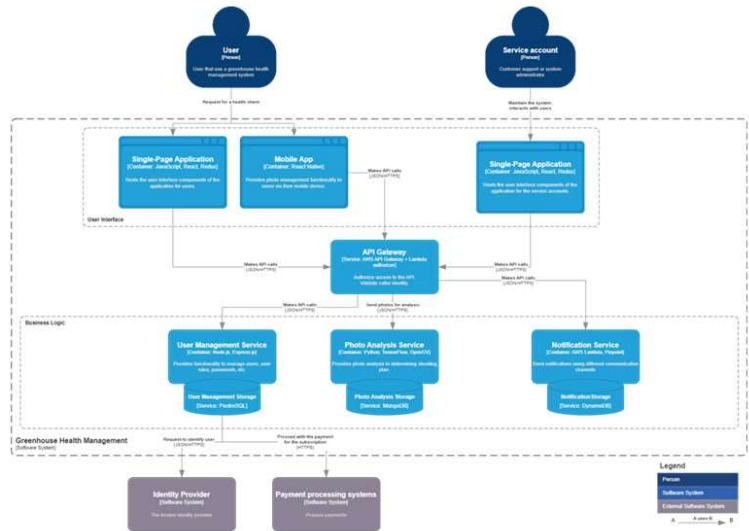
служба аналізу фотографій

служба імпорту/експорту

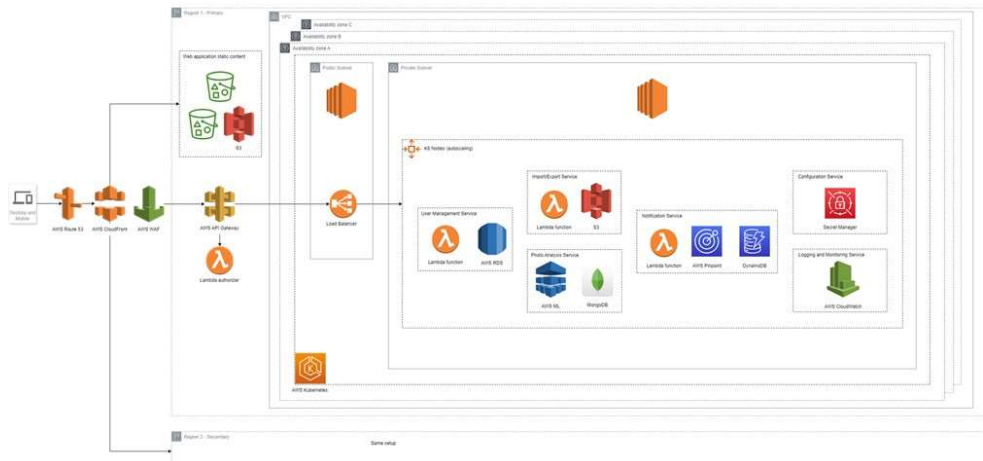
служба сповіщень

Зв'язок між мікросервісами виконаний через API або асинхронну передачу повідомлень

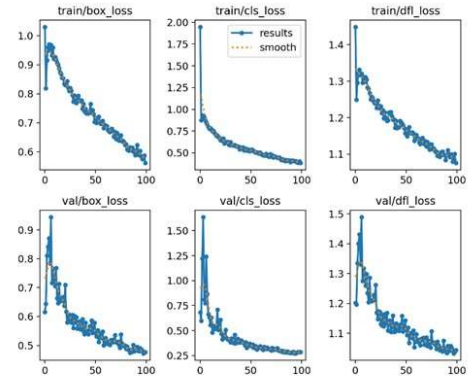
Діаграма логічних компонентів



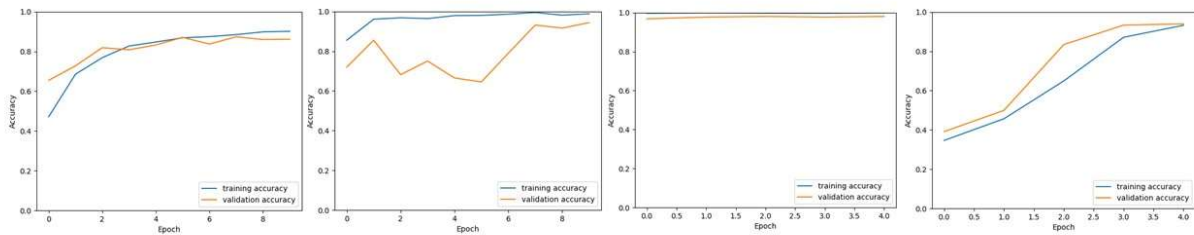
Діаграма розгортання компонентів



Аналіз результатів розпізнавання листя



Аналіз результатів класифікації бур'янів



Першим кроком до перенесення навчання є заморожування всіх шарів і навчання лише верхніх шарів.

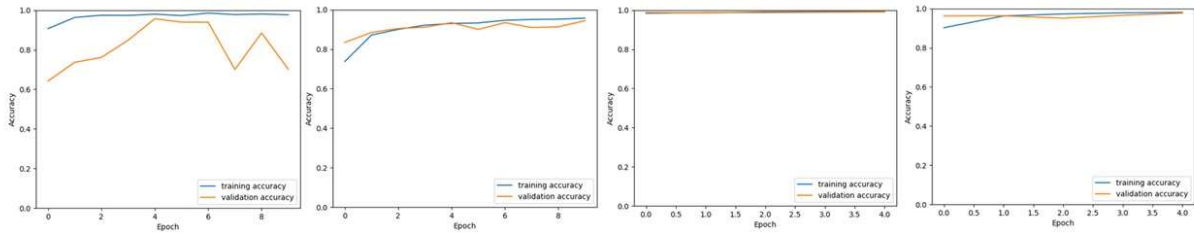
Моделі тренувалися на десяти епохах.

Другим кроком є розморожування 20 верхніх шарів і підгонка моделі з меншою швидкістю навчання.

Моделі тренувалися на п'яти епохах.

Images left to right: VGG19 перший крок, VGG19 другий крок, MobileNet перший крок, MobileNet другий крок.

Аналіз результатів класифікації захворювань



Першим кроком до перенесення навчання є заморожування всіх шарів і навчання лише верхніх шарів.

Моделі тренувалися на десяти епохах.

Другим кроком є розморожування 20 верхніх шарів і підгонка моделі з меншою швидкістю навчання.

Моделі тренувалися на п'яти епохах.

Images left to right: VGG19 перший крок, VGG19 другий крок, MobileNet перший крок, MobileNet другий крок.

Порівняння класифікаторів

Проаналізувавши результати навчання, ми дійшли висновку, що обидві моделі мають високі показники точності класифікації бур'янів.

Однак після експерименту з фотографією, яка містить і паросток бур'яну, і паросток помідора, стає очевидним, що модель MobileNet дає хибно позитивні результати. Ця проблема може призвести до неправильної загальної роботи класифікатора HNN.

Такий результат можна вважати очікуваним через різницю в кількості параметрів моделі.

FALSE TRUE RESULTS OF WEED CLASSIFIERS

Neural Network	Number of Parameters	Accuracy, %	Weed Type
VGG19	143.7M	37.32%	Black grass
MobileNet	4.3M	99.99%	Black grass

Порівняння класифікаторів

Проаналізувавши результати навчання, можна побачити, що модель VGG19 має вищу точність класифікації захворювань.

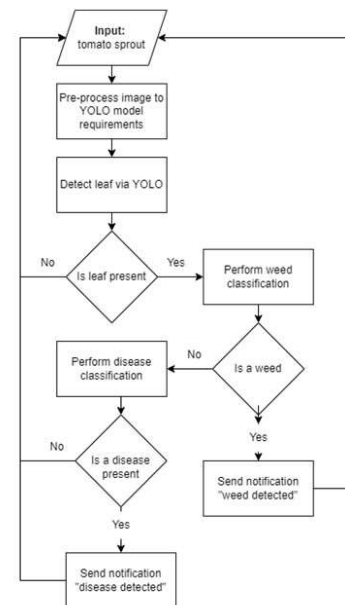
Цей висновок підтверджено результатом експерименту з класифікації листя помідора, ураженого хворобою листової плісняви.

RESULTS OF DISEASE CLASSIFIERS

Neural Network	Number of Parameters	Accuracy, %	Weed Type
VGG19	143.7M	99.99%	Leaf mold
MobileNet	4.3M	53.02%	Leaf mold

Алгоритм застосування класифікатора

- отримайте зображення листя рослини або ураженої ділянки за допомогою камери;
- попередньо стандартизуйте його формат для введення в моделі нейронної мережі;
- використовуйте попередньо навчену модель YOLOv8 для виявлення листя;
- використовуйте модель VGG19, навчену для класифікації бур'янів для виявлення бур'янів;
- оцініть результати класифікації, щоб визначити присутність бур'янів;
- використовуйте модель VGG19, навчену для класифікації захворювань;
- виявляйте та класифікуйте будь-які захворювання, присутні на листках;
- об'єднайте результати класифікації бур'янів і хвороб, щоб приймати обгрунтовані рішення щодо здоров'я рослин і стратегій управління.



Висновки



ХАРАКТЕРИСТИКИ НЕЙРОННИХ МЕРЕЖ

Neural network	Number of Parameters	Application size	ET till 100 epoch, weed classification	ET till 100 epoch, disease classification
VGG19	143.7M	549MB	33h	147h
MobileNet	4.3M	16MB	5.5h	31h

ДЯКУЮ ЗА УВАГУ

ДОДАТОК Г
Апробація результатів роботи



8th Open International Conference “Electrical,
Electronic and Information Sciences”

eStream 2024

CERTIFICATE

***Serhii Chaikovskiy and
Sergey Smelyakov***

Delivered an Oral Presentation

***HYBRID NEURAL NETWORK CLASSIFIER FOR
DETECTING WEEDS AND PLANT DISEASES IN
GREENHOUSES***

Chair of Science Committee

Prof. Dr Dalius Navakauskas

Section Chair

Assoc. Prof. Dr Dainius Udris

April 25, 2024
Vilnius

Hybrid Neural Network Classifier for Detecting Weeds and Plant Diseases in Greenhouses

Serhii Chaikovskiy
 Department of Software
 Engineering
 Kharkiv National University of
 Radio Electronics
 Ukraine, Kharkiv, Nauka Ave, 14
 serhii.chaikovskyi@nure.ua

Sergey Smelyakov
 Department of Software
 Engineering
 Kharkiv National University of
 Radio Electronics
 Ukraine, Kharkiv, Nauka Ave, 14
 serhii.smeliakov@nure.ua

Abstract — Greenhouse farming plays a crucial role in satisfying the demand throughout the year, regardless of climatic conditions. However, maintaining crop health in greenhouses is critical and challenging. This paper is devoted to analyzing the performance of a hybrid neural network (HNN) based classifier for effective and timely detection of weeds and tomato diseases in greenhouses. Each plant disease has unique features that can be recognized and classified. It is also possible to analyze images of plants in the early stages to detect weeds because weeds are usually significantly different from varietal plants. Thus, we can effectively analyze the condition and type of plants at each stage of cultivation. HNN models can be used to optimize the resource usage for growing a crop unit. It will also allow better monitoring of plant health. In addition, early detection of diseases will significantly reduce the excessive use of agrochemicals.

Keywords — *Hybrid Neural Networks, classifier, weeds detection, tomato diseases.*

I. INTRODUCTION

The cultivated tomato, *Solanum lycopersicum* L., is the world's most highly consumed vegetable due to its status as a basic ingredient in many raw, cooked, or processed foods [1]. Tomato can be grown in various geographical zones in open fields or greenhouses and can be harvested manually or mechanically [1, 2]. Predominantly treated as an annual crop, tomato is of significant commercial relevance globally [3].

Although it's challenging to quantify global vegetable production in greenhouses, some qualitative data can show tomato production rates. For example, imports from Mexico in 2017 accounted for almost 84 percent (1.8 billion pounds) of the greenhouse-tomato volume coming into the U.S. market. Meanwhile, imports of Canadian greenhouse-grown tomatoes have remained at about 300 million pounds [4].

Notably, investment in greenhouses continues to grow especially due to the COVID [5]. For example, there was a 23.9% rise in hectares between 2015 and 2020 in Canada, which led to greenhouse fruit and vegetable production capacity rising to 1,809 hectares in 2020. As a result, tomatoes led to sales growth of 12.1% [6].

The boom in greenhouse farming, however, is not devoid of challenges. A critical area of concern is maintaining optimal

crop health within these controlled environments [7]. More so, given the new European Climate Law that imposes targets to reduce the use and risk of chemical pesticides by 50% in 2030 [8], optimizing resource usage for each crop unit becomes imperative.

Recognizing these challenges and driven by recent technological advancements, this paper introduces an AI-driven Hybrid Neural Network (HNN) classifier tailored for the efficient detection of diseases and weeds in greenhouse tomato cultivation [2, 5]. This AI-automated tool will help to make disease and weed detection timely, cost-effective, and environmentally friendlier.

The purpose of this work is to experimentally evaluate the effectiveness of a Hybrid Neural Network Classifier to detect diseases and weed growth in a tomato greenhouse.

II. EXPERIMENT PLANNING

A. Initial Data

To detect leaves and perform proper image cropping, 2000 images were sampled from the custom dataset with image annotation. It consists of images of plants in the soil. The dataset consists of 1400 train images and 600 validation images [9].

To detect weeds, the V2 Plant Seedlings Dataset was used. This dataset contains 5,539 images of crop and weed seedlings. The images are grouped into 12 classes as shown in the above pictures. These classes represent common plant species in Danish agriculture. Each class contains RGB images that show plants at different growth stages. The images are in various sizes and are in PNG format [10].

To detect leaf diseases, the Tomato leaf disease detection dataset was used. The data has different types of diseases for tomato leaves. The training dataset contains 10000 labeled images and test/val data contains 1000 labeled images [11].

B. Selection of convolutional neural networks

The YOLOv8 [12] CNN was chosen for leaf detection due to its high speed, especially compared to RetinaNet [13], and its ability to work with video files.

YOLO or You Only Look Once is a very popular CNN architecture that is currently used to recognize multiple objects in an image. The main feature of this architecture in comparison with others is that most systems apply CNN several times to different regions of the image.

In YOLO CNN is applied once to the whole image at once. The network divides the image into a grid and predicts bounding boxes and the probability that there is a desired object in each area. The advantages of this approach are that the network looks at the whole image at once and considers the context in identifying and recognizing the object. Similarly, YOLO is 1000 times faster than R-CNN and about 100x faster than Fast R-CNN [14].

For weed and disease classification, we compared technologies such as VGG19 and MobileNet.

VGG19 is a variant of the VGG model, a Convolutional Neural Network (CNN) architecture introduced by the Visual Geometry Group from Oxford. VGG19 has 19 layers in total, including 16 convolutional layers, which allows this network to detect complex features in images. Specifically, the successive convolutional layers in VGG19 can extract low-level features in the initial layers, and more abstract, high-level features in the deeper layers. This makes VGG19 particularly suitable for our purpose of identifying subtle differences in the appearances of various weeds and tomato leaf diseases [15].

MobileNet is characterized by its small size and low latency, making it an attractive choice for applications that require rapid, real-time processing. It employs 'squeeze and excitation' optimization and hard swish activation to achieve high efficiency [16].

Both models were evaluated based on their computational efficiency, classification accuracy, and robustness against diverse and potentially challenging lighting, soil, and weather conditions in greenhouse settings. The chosen classifier would need to maintain high performance across all these parameters to effectively identify weeds and diseases, thereby preventing their spread and ensuring optimal crop health.

C. Metrics and comparison criteria

To measure the performance of the proposed HNN classifier and the specific neural network models we will use several key metrics and criteria designed to capture various aspects of effectiveness and efficiency. Critical comparison criteria included accuracy, precision, recall, and average precision. Each of these metrics provides a different perspective on the models' performance, and together they ensure a comprehensive evaluation.

Accuracy is one of the most straightforward metrics. It measures the proportion of correct predictions made by the models. While it gives an overall idea of the model's performance, it does not differentiate between false positives and false negatives, which leads us to precision and recall.

Precision is the ability of a model to identify only the relevant objects. It is the percentage of correct positive predictions and is given by [17].

$$Precision = \frac{TP}{AllDetections} \quad (1)$$

True Positive (TP): A correct detection. Detection with IOU \geq threshold. False Negative (FN): A ground truth not detected [17, 18].

False Positive (FP): A wrong detection. Detection with IOU $<$ threshold. Threshold: depending on the metric, it is usually set to 50%, 75% or 95% [17, 18].

Recall or Sensitivity measures the proportion of actual positives correctly identified. This is crucial for preventing disease and weed spread.

$$Recall = \frac{TP}{All\ Ground\ Truths} \quad (2)$$

The Precision x Recall curve is used to evaluate the performance of an object detector as the confidence is changed by plotting a curve for each object class [17].

Average Precision (AP) is a numerical metric and can also help us compare different detectors. In practice AP is the precision averaged across all recall values between 0 and 1. The general definition for the average precision is finding the area under the precision-recall curve [19].

$$\sum_{n=0}^{r_{n+1} - r_n} \rho_{interp}(r_n + 1), \quad (3)$$

where

$$\rho_{interp}(r_n + 1) = \max_{\bar{r}: \bar{r} \geq r_{n+1}} \rho(\bar{r}), \quad (4)$$

This is the measured precision at recall [20].

In this case, instead of using the precision observed at only few points, the AP is now obtained by interpolating the precision at each level, taking the maximum precision whose recall value is greater or equal than . This way we calculate the estimated area under the curve [19].

D. Experiment planning

In our research, we used the YOLOv8 pre-trained model to detect leaves trained on the custom dataset [9].

For the image classification to detect weeds and diseases, we trained VGG19 and MobileNet models on the weed dataset [10], and on the tomato disease dataset [11]. Both pieces of training used pre-trained ImageNet weights.

After that, using the test dataset, we measured accuracy, precision, recall, and average precision. The data obtained during trainings were displayed on graphs.

E. Tools and software

In our research, we used a Python programming language in the PyCharm development environment. The YOLOv8 model was taken from the Roboflow Inference [20]. To train VGG19 and MobileNet the TensorFlow [21] and Keras API [22] were used. No video card was used upon training.

III. EXPERIMENTAL RESULTS ANALYSIS

A. Leaf detection results

The YOLOv8 model was pre-trained to detect leaves and perform proper image cropping, 2000 images were sampled from the custom dataset with image annotation. As we can see on the training curve, the training of the neural network has been finished and further training does not make sense. The neural network is ready to use.

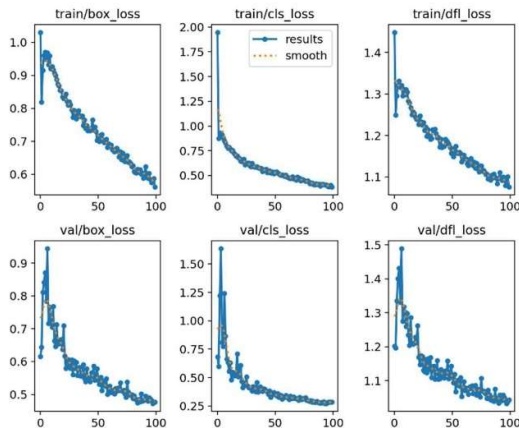


Fig. 1. Leaf detection training curves [9]

Below you can see the Precision, Recall, and Average Precision curves (Fig. 2, Fig. 3, Fig. 4)

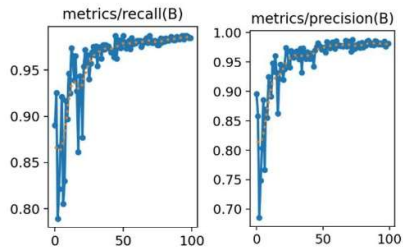


Fig. 2. YOLOv8 recall and precision curves [9]

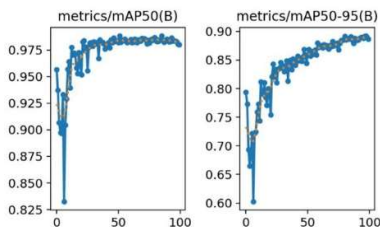


Fig. 3. YOLOv8 average precision curve [9]

The YOLOv8 showed high rates of image processing speed, as well as the accuracy of leaf detection.

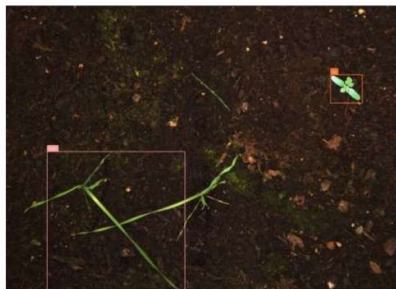


Fig. 4. YOLOv8 leaf detection [23]

B. Weed classification results

The VGG19 and MobileNet models were trained for weed classification on the V2 Plant Seedlings Dataset [10]. This dataset contains 5,539 images of crop and weed seedlings. We used pre-trained ImageNet weights. Training was performed in two steps.

The first step to transfer learning is to freeze all layers and train only the top layers. Models were trained on ten epochs.

Training time for the VGG19 model took at average 810 seconds per epoch with a total training time 3 hours for the first step. Training time for the MobileNet model took at average 200 seconds per epoch with a total training time 30 minutes for the first step.

The second step was to unfreeze top 20 layers and fit the model using smaller learning rate. Models were trained on five epochs.

Training time for the VGG19 model took at average 2050 seconds per epoch with a total training time 3 hours for the second step. Training time for the MobileNet model took at average 200 seconds per epoch with a total training time 30 minutes for the second step.

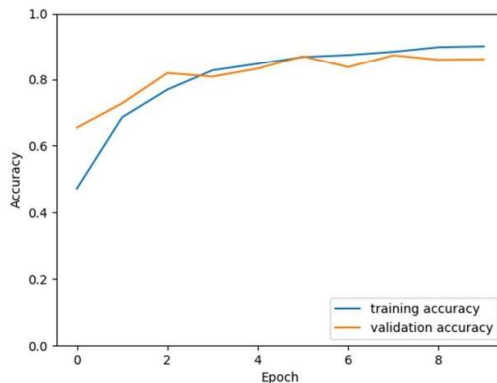


Fig. 5. VGG19 accuracy curve for the first step of the weed classification training [24]

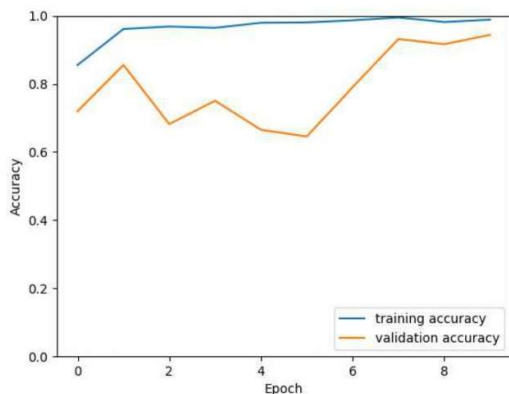


Fig. 6. MobileNet accuracy curve for the first step of the weed classification training [25]

After analyzing the training results, we concluded that both models have high accuracy rates of weed classification. However, after experimenting with a photo that contains both a weed sprout and a tomato sprout, it becomes obvious that the MobileNet model gives false positive results. This issue might lead to incorrect overall work of the HNN classifier.



Fig. 7. Tomato sprout cropped from the original photo [28]

Keras 3 provides such data about models: MobileNet has 4.3 million parameters, and VGG19 has 143.7 million parameters. Such a result can be considered as expected due to a difference in the number of model parameters.

TABLE I. FALSE TRUE RESULTS OF WEED CLASSIFIERS

Neural Network	Number of Parameters	Accuracy, %	Weed Type
VGG19	143.7M	37.32%	Black grass
MobileNet	4.3M	99.99%	Black grass

C. Disease classification results

The VGG19 and MobileNet models were trained for tomato disease classification on the Tomato leaf disease [11] detection dataset was used. This dataset contains 10000 labeled images and test/val data contains 1000 labeled images. We used pre-trained ImageNet weights.

The first step to transfer learning is to freeze all layers and train only the top layers. Models were trained on ten epochs.

Training time for the VGG19 model took at average 4700 seconds per epoch with a total training time 13 hours for the first

step. Training time for the MobileNet model took at average 1100 seconds per epoch with a total training time 3 hours for the first step.

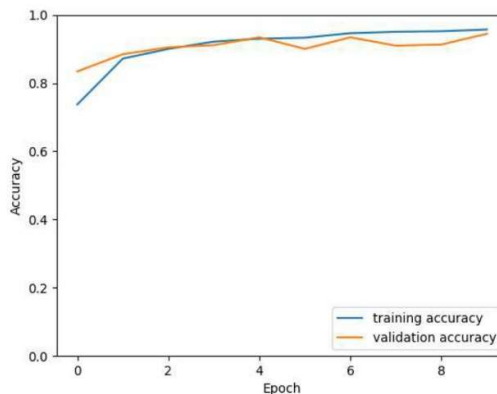


Fig. 8. VGG19 accuracy curve for the first step of the disease classification training [26]

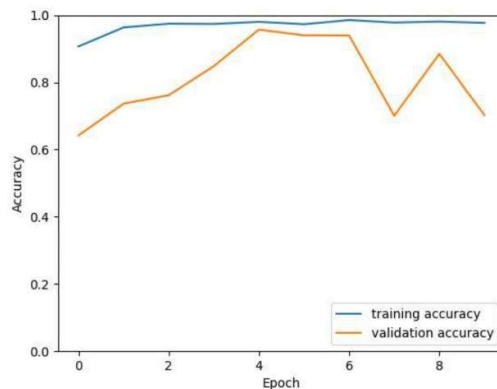


Fig. 9. MobileNet accuracy curve for the first step of the disease classification training [27]

The second step is to unfreeze top 20 layers and fit the model using smaller learning rate. Models were trained on five epochs.

Training time for the VGG19 model took at average 12000 seconds per epoch with a total training time 17 hours for the second step. Training time for the MobileNet model took at average 810 seconds per epoch with a total training time 1 hour for the second step.

After analyzing the training results, we can see that the VGG19 model has higher accuracy rate of disease classification.

This conclusion was confirmed by the result of an experiment on the classification of a tomato leaf affected by the leaf mold disease.



Fig. 10. Tomato leaf with leaf mold disease [29]

TABLE II. RESULTS OF DISEASE CLASSIFIERS

Neural Network	Number of Parameters	Accuracy, %	Weed Type
VGG19	143.7M	99.99%	Leaf mold
MobileNet	4.3M	53.02%	Leaf mold

IV. APPLICATION ALGORITHM

Considering the results of our research, we can deduce an instruction to effectively utilize the HNN classifier. We outline a step-by-step algorithm to leverage this AI technology for effective leaf detection, weed classification, and disease identification. The aim is to provide a comprehensive guide that can be followed to use the HNN classifier for plant health diagnosis and subsequent decision-making. This process involves several stages, including image acquisition, pre-processing, leaf detection using the YOLOv8 model, weed and disease classification using the VGG19 model, and finally, combining these results to make informed decisions about plant health management. The algorithm also emphasizes the importance of continuous model refinement for improved accuracy and performance.

The suggested algorithm of the HNN classifier application will be depicted later in this section.

A. Leaf detection

Obtain an image of the plant leaves or affected area of interest using a camera or other imaging devices.

Pre-process the acquired image to enhance its quality and standardize its format for input into the neural network models. This may involve resizing, normalization, and other relevant techniques to ensure consistency in the input data.

Utilize the pre-trained YOLOv8 model for leaf detection. The model automatically identifies and crops out the regions containing leaves from the input image, facilitating further analysis.

B. Weed classification

Employ the VGG19 model trained for weed classification. Pass the cropped leaf images into the models. The VGG19 model is particularly suited for detailed feature extraction. Evaluate the classification results to identify any presence of weeds.

C. Disease classification

Employ the VGG19 model trained for disease classification. Detect and classify any diseases present on the leaves by feeding the cropped images into the models.

D. Final decision

Combine the results from weed and disease classification to make informed decisions regarding plant health and management strategies.

Continuously update and refine the models based on new data and feedback from real-world applications to improve their accuracy and performance over time.

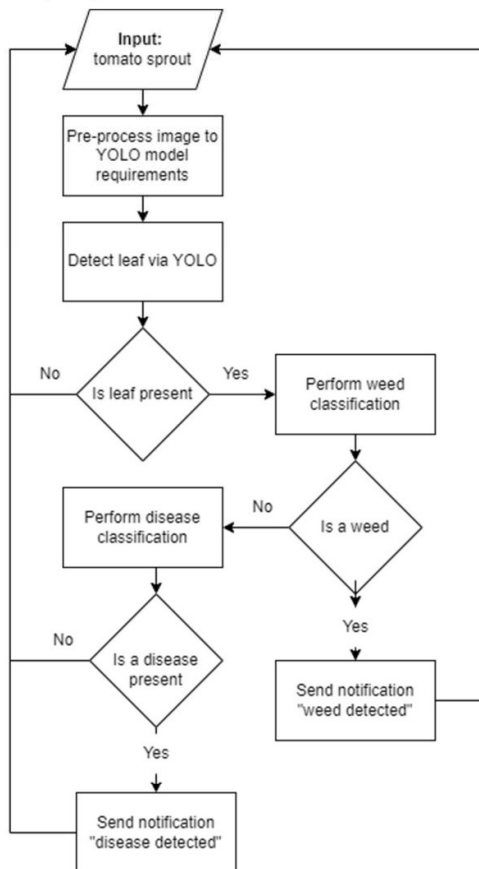


Fig. 11. Suggested application algorithm [30]

V. CONCLUSION

Based on the results of our research, the following main conclusions can be drawn.

In our research, we experimentally analyzed the most modern neural network models for solving problems such as leaf detection, weed classification, and tomato disease classification.

For leaf detection, we used a custom dataset and a pre-trained YOLOv8 model [9].

We used the weed dataset [6] for weed classification and the tomato disease dataset [11] for tomato disease classification.

VGG19 and MobileNet models were selected as candidates for weed classifiers. Models were trained with pre-trained ImageNet weights.

After trainings, we compared VGG19 and MobileNet models' accuracy for weed and tomato disease classification tasks.

The VGG19 model showed a high accuracy rate in weed prediction and disease classification tasks. The average rate for the VGG19 model on the disease classification task was 99.99%. The VGG19 model showed a low rate of false true results in the task of weed prediction - 37.32% for a test set of images. A downside of the VGG19 model is its training time - 33 hours for the fully trained weed classification model plus 147 hours for the fully trained disease classification model.

On the other hand, the MobileNet model offers faster training time - 5.5 hours for a fully trained weed classification model plus 31 hours for a fully trained disease classification model. The MobileNet model might be more suitable for resource-constrained environments due to the smaller size of the application - 16MB [31] compared to 549MB [31] for the VGG19 application. A downside of the MobileNet model is its high rate of false true results in the weed classification task - 99.99% for the test set of images. Also, the MobileNet model showed an insufficient accuracy in disease detection tasks - 53.02% on average compared to 99.99% for VGG19.

Since each model has its advantages and disadvantages, depending on the conditions of use, we should use the appropriate model. Use the VGG19 model when there are no resource constraints. For resource-constrained environment cases, it is possible to use the MobileNet model.

REFERENCES

- H. I. Peyal et al., "Plant Disease Classifier: Detection of Dual-Crop Diseases Using Lightweight 2D CNN Architecture," in *IEEE Access*, vol. 11, pp. 110627-110643, 2023, doi: 10.1109/ACCESS.2023.3320686. (<https://ieeexplore.ieee.org/document/10267978>)
- E. Özbilge, M. K. Ulukök, Ö. Toygar and E. Ozbilge, "Tomato Disease Recognition Using a Compact Convolutional Neural Network," in *IEEE Access*, vol. 10, pp. 77213-77224, 2022, doi: 10.1109/ACCESS.2022.3192428. (<https://ieeexplore.ieee.org/document/9832884>)
- Geisenberg, C. and K. Stewart (1986), "Field crop management", in Atherton, J.G. and J. Rudich (eds.), *The Tomato Crop: A Scientific Basis for Improvement*, Chapman & Hall, London, pp. 511-557.
- USDA 2019, "Imported Greenhouse Tomatoes From Mexico Illustrate the Growing Diversity in Fresh-Market Tomatoes". (<https://www.ers.usda.gov/data-products/chart-gallery/gallery/chart-detail/?chartId=93021>)
- C. Zhou, S. Zhou, J. Xing and J. Song, "Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network," in *IEEE Access*, vol. 9, pp. 28822-28831, 2021, doi: 10.1109/ACCESS.2021.3058947. (<https://ieeexplore.ieee.org/document/9353592>)
- Farm Credit Canada (2021), "Despite remarkable greenhouse sales in 2020 – uncertainty lies in the year ahead", Canada. (<https://www.fcc-fac.ca/en/knowledge/economics/greenhouse-sales>)
- P. D. Rosero-Montalvo, C. A. Gordillo-Gordillo and W. Hernandez, "Smart Farming Robot for Detecting Environmental Conditions in a Greenhouse," in *IEEE Access*, vol. 11, pp. 57843-57853, 2023, doi: 10.1109/ACCESS.2023.3283986. (<https://ieeexplore.ieee.org/document/10146280>)
- European Commission 2022, "Sustainable use of pesticides". (https://food.ec.europa.eu/plants/pesticides/sustainable-use-pesticides_en)
- Leaf detection dataset (<https://universe.roboflow.com/project-kunew/lettuce-weed-4/dataset/1>).
- V2 Plant Seedlings Dataset. (<https://www.kaggle.com/datasets/vbookshelf/v2-plant-seedlings-dataset?resource=download>)
- Tomato leaf disease detection. (<https://www.kaggle.com/datasets/kaustubhb999/tomatoleaf/data>)
- YOLOv8 (<https://blog.roboflow.com/whats-new-in-yolov8/>)
- RetinaNet (<https://paperswithcode.com/method/retinanet>)
- Getting Started with R-CNN, Fast R-CNN, and Faster R-CNN (<https://ch.mathworks.com/help/vision/ug/getting-started-with-r-cnn-fast-r-cnn-and-faster-r-cnn.html>)
- Very Deep Convolutional Networks for Large-Scale Image Recognition (<https://paperswithcode.com/paper/very-deep-convolutional-networks-for-large>)
- MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications (<https://arxiv.org/pdf/1704.04861.pdf>)
- K. Smelyakov, A. Chupryna, O. Bohomolov and N. Hunko, "The Neural Network Models Effectiveness for Face Detection and Face Recognition," 2021 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2021, pp. 1-7, doi: 10.1109/eStream53087.2021.9431476. (<https://ieeexplore.ieee.org/document/9431476>)
- Object Detection Metrics: (<https://github.com/rafaelpadilla/Object-Detection-Metrics>)
- K. Smelyakov, D. Tovchyrchko, I. Ruban, A. Chupryna and O. Ponomarenko, "Local Feature Detectors Performance Analysis on Digital Image," 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 2019, pp. 644-648, doi: 10.1109/PICST47496.2019.9061331. (<https://ieeexplore.ieee.org/document/9061331>)
- Roboflow Inference (https://inference.roboflow.com/quickstart/what_is_inference/)
- TensorFlow (<https://www.tensorflow.org/learn>)
- Keras (<https://keras.io/about/>)
- Picture YOLOv8 leaf detection result: https://drive.google.com/file/d/1d3qM_llqgXXaTB0BGBpIKaIGVCwmCq3X/view?usp=drive_link
- Picture VGG19 weed detection training first step: https://drive.google.com/file/d/1NmfrdO0Lo4sUb_hci-1h4wwsVL5yswY/view?usp=drive_link
- Picture MobileNet weed classification training first step: https://drive.google.com/file/d/1YHZVU-VglHSSZakTyx2DF_KaOhGi8huA/view?usp=drive_link
- Picture VGG19 disease classification training first step: https://drive.google.com/file/d/1BKCYD2YfupBTd4qaqd3kE8O1cxLNwBdm/view?usp=drive_link
- Picture MobileNet disease classification training first step: https://drive.google.com/file/d/1eegC3Pjo451qvhTapvflMBmvYJANGrJp/view?usp=drive_link
- Picture tomato sprout after leaf detection: https://drive.google.com/file/d/1BykvAEVQ7OUgtb-Wt9H1k8qL8o3Ys498/view?usp=drive_link
- Picture leaf mold sample: https://drive.google.com/file/d/1C-D70jheCucOhtCD-mrferAIBIEPncEP/view?usp=drive_link
- Application algorithm: <https://drive.google.com/file/d/19rYMSDWoatXSGER-WVUf8SUDm3YQZhVh/view?usp=sharing>
- Keras Application characteristics: <https://keras.io/api/applications/>

ДОДАТОК Д
Експертний висновок результатів перевірки кваліфікаційної роботи на
відповідність оформлення вимогам ДСТУ 3008: 2015

1

Експертний висновок результатів перевірки кваліфікаційної роботи

студент
(посада)

програмної інженерії
(кафедра)

ПЗМ-22-6
(група)

Чайковський Сергій Олександрович

(прізвище, ім'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.4 Нумерація розділів, підрозділів, пунктів, підпунктів	
	7.5 Рисунок	
	7.6 Таблиці	
	7.7 Переліки	
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні позначки	
	7.15 Додатки	

зауважень немає

Експерт

(підпис)

Олена ОЛІЙНИК

(прізвище, ініціали)

08.06.2024