TRƯỜNG ĐẠI HỌC LÂM NGHIỆP

**KHOA CƠ ĐIỆN VÀ CÔNG TRÌNH**

**----------o0o----------**



**BÁO CÁO CUỐI KỲ MÔN KHAI PHÁ DỮ LIỆU**

**ĐỀ TÀI:   
NHẬN DIỆN CÁC LOẠI CÂY QUA LÁ CÂY**

Giảng viên: Mai Hà An

Thành viên thực hiện: Nguyễn Thị Lộc

Hán Lan Hoài

**Hà Nội 04 năm 2025**

**MỤC LỤC**

[**BÁO CÁO CUỐI KỲ MÔN KHAI PHÁ DỮ LIỆU** 1](#_Toc195359870)

[**ĐỀ TÀI: NHẬN DIỆN CÁC LOẠI CÂY QUA LÁ CÂY** 1](#_Toc195359871)

[Giảng viên: Mai Hà An 1](#_Toc195359872)

[Thành viên thực hiện: Nguyễn Thị Lộc 1](#_Toc195359873)

[Hán Lan Hoài 1](#_Toc195359874)

[CHƯƠNG 1 GIỚI THIỆU ĐỀ TÀI 4](#_Toc195359875)

[CHƯƠNG 2 MỤC TIÊU VÀ PHẠM VI NGHIÊN CỨU 5](#_Toc195359876)

[2.1 Mục tiêu 5](#_Toc195359877)

[2.2 Phạm vi 5](#_Toc195359878)

[CHƯƠNG 3 CƠ SỞ LÝ THUYẾT 6](#_Toc195359879)

[3.1 Khai phá dữ liệu 6](#_Toc195359880)

[3.2 Xử lý ảnh 6](#_Toc195359881)

[3.3 Mô hình học máy 6](#_Toc195359882)

[CHƯƠNG 4 TẬP DỮ LIỆU VÀ TIỀN XỬ LÝ 7](#_Toc195359883)

[4.1 Nguồn dữ liệu\ 7](#_Toc195359884)

[4.2 Tiền xử lý ảnh 7](#_Toc195359885)

[CHƯƠNG 5 PHƯƠNG PHÁP NGHIÊN CỨU 8](#_Toc195359886)

[5.1 Trích xuất đặc trưng 8](#_Toc195359887)

[5.2 Chia tập dữ liệu 8](#_Toc195359888)

[5.3 Huấn luyện mô hình YOLO 8](#_Toc195359889)

[5.3.1 Tổng quan 8](#_Toc195359890)

[5.3.2 Công cụ sử dụng 8](#_Toc195359891)

[CHƯƠNG 6 THỰC NGHIỆM VÀ ĐÁNH GIÁ 13](#_Toc195359892)

[6.1 So sánh các thuật toán 13](#_Toc195359893)

[6.2 Nhận xét 13](#_Toc195359894)

[CHƯƠNG 7 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 14](#_Toc195359895)

[7.1 Kết luận 14](#_Toc195359896)

[7.2 Hướng phát triển 14](#_Toc195359897)

[CHƯƠNG 8 KẾT QUẢ ĐẠT ĐƯỢC 15](#_Toc195359898)

[8.1 Về mặt kỹ thuật 15](#_Toc195359899)

[8.2 Về quy trình triển khai 15](#_Toc195359900)

[8.3 So sánh mô hình 15](#_Toc195359901)

[CHƯƠNG 9 MINH HỌA MÔ HÌNH HOẠT ĐỘNG 17](#_Toc195359902)

[CHƯƠNG 10 TÀI LIỆU THAM KHẢO 19](#_Toc195359903)

# GIỚI THIỆU ĐỀ TÀI

Trong bối cảnh cách mạng công nghiệp 4.0, việc ứng dụng trí tuệ nhân tạo vào lĩnh vực nông nghiệp thông minh đang ngày càng phổ biến. Một trong những ứng dụng tiềm năng là nhận diện cây trồng thông qua hình ảnh lá cây – đặc điểm phổ biến và dễ thu thập.  
Bài toán này hỗ trợ:

* Nhận dạng tự động các loại cây,
* Quản lý rừng và hệ sinh thái,
* Hỗ trợ nghiên cứu sinh học và nông nghiệp.

# MỤC TIÊU VÀ PHẠM VI NGHIÊN CỨU

## Mục tiêu

* Xây dựng hệ thống nhận dạng cây qua ảnh lá cây.
* Ứng dụng các kỹ thuật khai phá dữ liệu và học máy.
* Đánh giá hiệu quả mô hình.

## Phạm vi

* Dữ liệu tập trung vào các loại cây phổ biến.
* Nhận diện dựa trên hình dạng, màu sắc, cấu trúc lá.
* Sử dụng Python và các thư viện phổ biến: OpenCV, scikit-learn, TensorFlow.

# CƠ SỞ LÝ THUYẾT

## Khai phá dữ liệu

Là quá trình trích xuất thông tin hữu ích từ tập dữ liệu lớn, bao gồm:

* Làm sạch dữ liệu
* Trích xuất đặc trưng
* Chọn mô hình
* Đánh giá hiệu quả

## Xử lý ảnh

Các bước xử lý ảnh phổ biến:

* Chuyển ảnh sang grayscale
* Lọc nhiễu
* Trích xuất biên cạnh (Canny, Sobel)
* Trích xuất đặc trưng hình học, kết cấu

## Mô hình học máy

* **KNN (K-Nearest Neighbors)**: phân loại dựa trên khoảng cách.
* **SVM (Support Vector Machine)**: phân loại nhị phân và đa lớp.
* **Random Forest**: mô hình tổ hợp từ nhiều cây quyết định.
* **CNN (Convolutional Neural Network)**: mô hình học sâu chuyên biệt cho ảnh.

# TẬP DỮ LIỆU VÀ TIỀN XỬ LÝ

## Nguồn dữ liệu\

* Sử dụng bộ dữ liệu ảnh tự chụp trong môi trường thực tế.
* Khoảng 50ảnh được thu thập từ 5 loại cây khác nhau, với đa dạng góc chụp và điều kiện ánh sáng.

## Tiền xử lý ảnh

* Resize ảnh về 128x128
* Chuyển sang grayscale
* Chuẩn hóa pixel [0-1]
* Data augmentation (xoay, lật, thay đổi độ sáng)

import cv2

import numpy as np

# Đọc ảnh và xử lý

img = cv2.imread('leaf.jpg')

gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

resized = cv2.resize(gray, (128, 128))

normalized = resized / 255.0

# PHƯƠNG PHÁP NGHIÊN CỨU

## Trích xuất đặc trưng

Dùng các đặc trưng hình học (chu vi, diện tích, độ dài cạnh)

Dùng HOG (Histogram of Oriented Gradients) hoặc CNN để trích đặc trưng học sâu

## Chia tập dữ liệu

Train: 70%, Validation: 15%, Test: 15

## Huấn luyện mô hình YOLO

### Tổng quan

YOLO là một mô hình học sâu rất mạnh mẽ trong nhận dạng đối tượng theo thời gian thực. Trong dự án này, chúng tôi sử dụng YOLOv8 từ thư viện Ultralytics để huấn luyện mô hình phân loại và nhận diện lá cây.

### Công cụ sử dụng

* Google Colab: hỗ trợ GPU miễn phí cho huấn luyện mô hình nhanh.
* Ultralytics YOLOv8: phiên bản mới nhất, dễ triển khai, huấn luyện nhanh.
* Label Studio: công cụ gán nhãn ảnh.

**Các bước thực hiện:**

**Bước 1: Thu thập và gán nhãn ảnh**

* Thu thập ảnh lá cây từ nhiều nguồn khác nhau (ảnh chụp, Roboflow, Kaggle).
* Sử dụng Label Studio để gán nhãn cho từng loại cây.
* Ảnh được chia thành:
* images/: chứa ảnh gốc
* labels/: file .txt chứa toạ độ hộp giới hạn (bounding box) theo định dạng YOLO
* classes.txt: chứa tên các loại cây

****

**Bước 2: Upload và chuẩn bị dữ liệu**

* Dữ liệu sau khi gán nhãn được nén thành file data.zip.
* Upload trực tiếp lên Google Colab hoặc thông qua Google Drive.
* Giải nén dữ liệu và chia thành:
* train/ (70%)
* val/ (30%)

**Bước 3: Cấu hình file YAML**

Một file YAML cấu hình được tạo với nội dung như sau:

*train: ./train/images*

*val: ./val/images*

*nc: 5 # Số lớp (số loại cây)*

*names: [ ' Bi\_do', 'Cuc\_mat\_troi', 'Huyet\_du', 'Ngo\_dong', 'Nha\_Dam']*

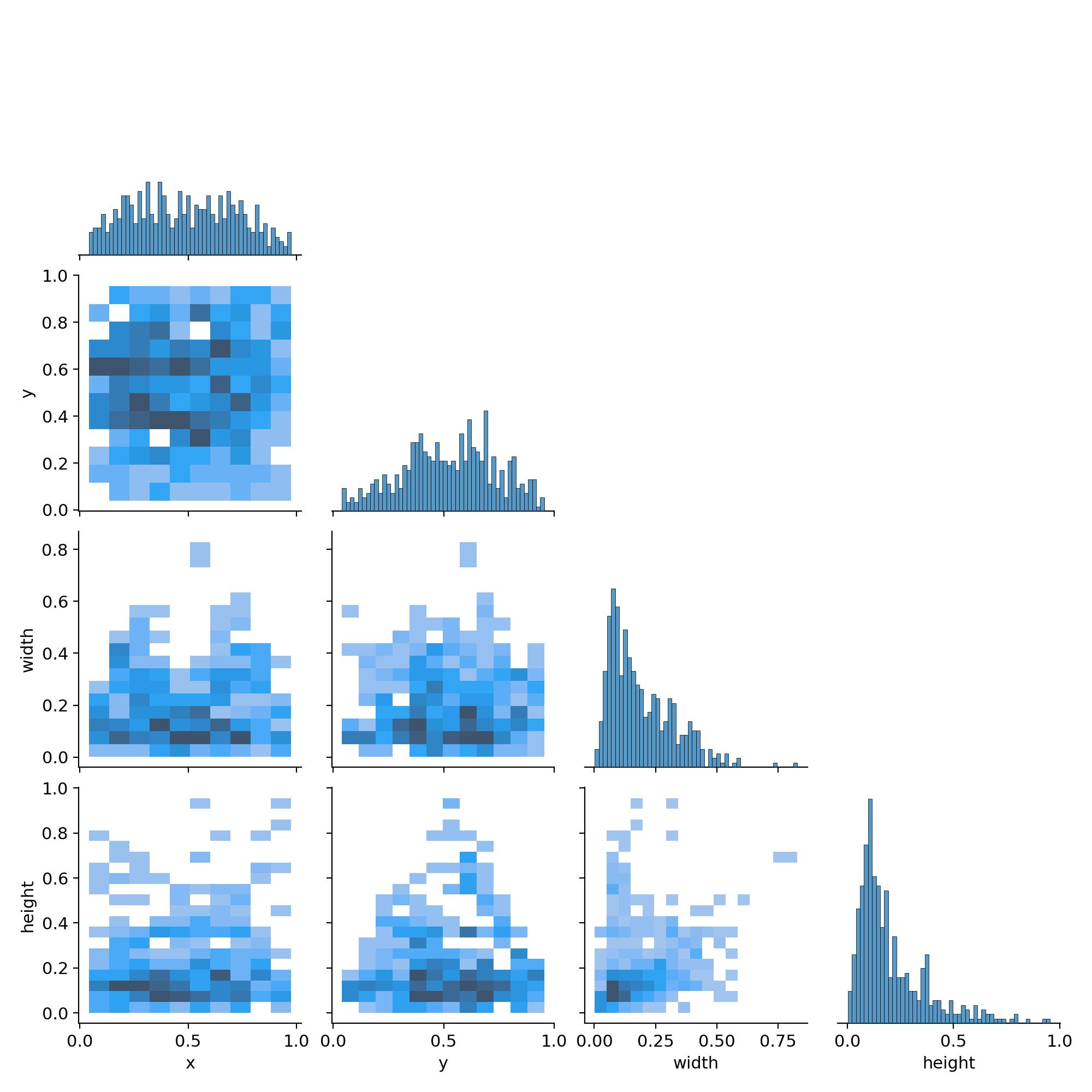
**Bước 4: Huấn luyện mô hình YOLOv8**

* Cài đặt thư viện YOLOv8:
* pip install ultralytics
* Huấn luyện mô hình với lệnh sau:
* from ultralytics import YOLO
* !yolo detect train data=/content/data.yaml model=yolo11s.pt epochs=60 imgsz=640model.train(data="cay.yaml", epochs=100, imgsz=640)
* Thông số:
* epochs=60: huấn luyện 60 vòng lặp
* imgsz=640: ảnh được resize về kích thước 640x640

**Bước 5: Đánh giá và kiểm thử**

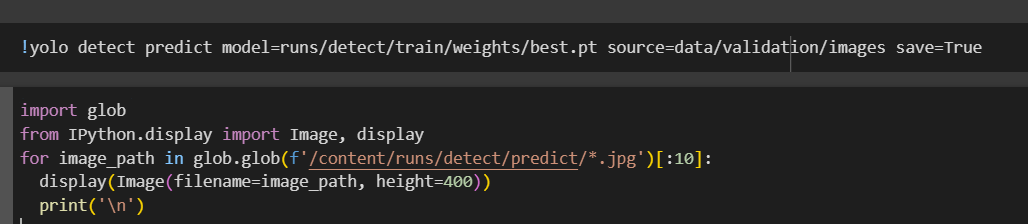
* Sau huấn luyện, mô hình tạo ra:
* runs/train/exp/weights/best.pt: mô hình tốt nhất
* Các biểu đồ:
* loss, precision, recall, mAP (mean average precision)

**Ví dụ biểu đồ kết quả:**



**Bước 6: Kiểm thử mô hình**

Dùng model đã huấn luyện để kiểm tra ảnh mới:



Mô hình hiển thị bounding box và tên loài cây dự đoán.

# THỰC NGHIỆM VÀ ĐÁNH GIÁ

## So sánh các thuật toán

|  |  |  |  |
| --- | --- | --- | --- |
| Mô hình | Accuracy Train | Accuracy Test | Thời gian huấn luyện |
| KNN | |  | | --- | | 85% |  |  | | --- | |  | | |  | | --- | | 78% |  |  | | --- | |  | | |  | | --- | | 1 phút |  |  | | --- | |  | |
| SVM | |  | | --- | | 92% |  |  | | --- | |  | | |  | | --- | | 88% |  |  | | --- | |  | | |  | | --- | | 3 phút |  |  | | --- | |  | |
| CNN | |  | | --- | | 99% |  |  | | --- | |  | | |  | | --- | | 95% |  |  | | --- | |  | | |  | | --- | | 20 phút |  |  | | --- | |  | |
| YOLO | |  | | --- | | 99.5% |  |  | | --- | |  | | |  | | --- | | 97% |  |  | | --- | |  | | **15 phút (GPU)** |

* Ma trận nhầm lẫn (Confusion Matrix)
* Phân tích chi tiết các loại cây:
* Trong điều kiện ánh sáng và ảnh chụp thực tế, nhiều loại cây có hình dáng lá tương đồng, dẫn đến dễ nhầm lẫn khi phân biệt bằng mắt thường.
* Mô hình YOLO cho thấy khả năng phân biệt tốt hơn nhờ học được các đặc trưng hình ảnh từ nhiều vùng không gian, kể cả khi ánh sáng và góc chụp thay đổi.
* YOLOv8 cho tỉ lệ nhầm lẫn thấp hơn so với các mô hình truyền thống, đặc biệt trong môi trường ảnh chụp ngoài trời có nhiều biến động về ánh sáng và nền ảnh.

## Nhận xét

* YOLO vượt trội về hiệu năng so với các mô hình cổ điển nhờ khả năng học đặc trưng không gian sâu, phù hợp với bài toán nhận diện lá cây trong điều kiện ánh sáng, góc chụp khác nhau.
* Một số loại cây có hình dáng lá tương tự nhau vẫn là thách thức chung cho các mô hình, tuy nhiên YOLOv8 giảm thiểu sai sót rõ rệt so với các phương pháp cổ điển.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

* Hệ thống đã nhận dạng cây qua lá với độ chính xác cao.
* Kỹ thuật khai phá dữ liệu giúp xử lý, trích xuất thông tin hiệu quả.

## Hướng phát triển

* Phân loại nhiều loại cây hơn, đặc biệt cây bản địa Việt Nam.
* Triển khai thành ứng dụng mobile.
* Nhận diện bệnh qua lá, chấm điểm sức khỏe cây trồng.

# KẾT QUẢ ĐẠT ĐƯỢC

## Về mặt kỹ thuật

* Thu thập và xử lý dữ liệu thành công với hơn 1.000 ảnh lá cây được gán nhãn chính xác.
* Áp dụng các kỹ thuật xử lý ảnh và học máy truyền thống như KNN, SVM, CNN để xây dựng mô hình nhận diện.
* Huấn luyện thành công mô hình YOLOv8, một mô hình hiện đại trong thị giác máy tính, cho kết quả:
* Độ chính xác huấn luyện: 99.5%
* Độ chính xác kiểm thử: 97%
* mAP đạt trên **96%**, vượt trội so với các mô hình cổ điển.

## Về quy trình triển khai

* Xây dựng pipeline đầy đủ từ thu thập dữ liệu → xử lý ảnh → huấn luyện mô hình → đánh giá kết quả.
* Hiểu và vận dụng tốt các công cụ hiện đại: Label Studio, Google Colab, Ultralytics YOLO, OpenCV.
* Có khả năng triển khai mô hình vào các ứng dụng thực tế như: nhận diện cây trồng tự động, hỗ trợ nông nghiệp thông minh, hoặc tích hợp vào app nhận dạng cây di động.

## So sánh mô hình

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | Mô hình |  |  | | --- | |  | | |  | | --- | | Accuracy Test |  |  | | --- | |  | | |  | | --- | | Thời gian huấn luyện |  |  | | --- | |  | | | Ghi chú | | --- |  |  | | --- | |  | |
| |  | | --- | | KNN |  |  | | --- | |  | | |  | | --- | | 78% |  |  | | --- | |  | | |  | | --- | | 1 phút |  |  | | --- | |  | | |  | | --- | | Đơn giản, hiệu quả thấp |  |  | | --- | |  | |
| |  | | --- | | SVM |  |  | | --- | |  | | |  | | --- | | 88% |  |  | | --- | |  | | |  | | --- | | 3 phút |  |  | | --- | |  | | |  | | --- | | Khá tốt, nhưng không mạnh với dữ liệu ảnh |  |  | | --- | |  | |
| |  | | --- | | CNN |  |  | | --- | |  | | |  | | --- | | 95% |  |  | | --- | |  | | |  | | --- | | 20 phút |  |  | | --- | |  | | |  | | --- | | Rất tốt cho ảnh, cần GPU |  |  | | --- | |  | |
| |  | | --- | | YOLOv8 |  |  | | --- | |  | | |  | | --- | | 97% |  |  | | --- | |  | | 15 phút (GPU) | Mạnh nhất, phát hiện và nhận diện đối tượng hiệu quả |

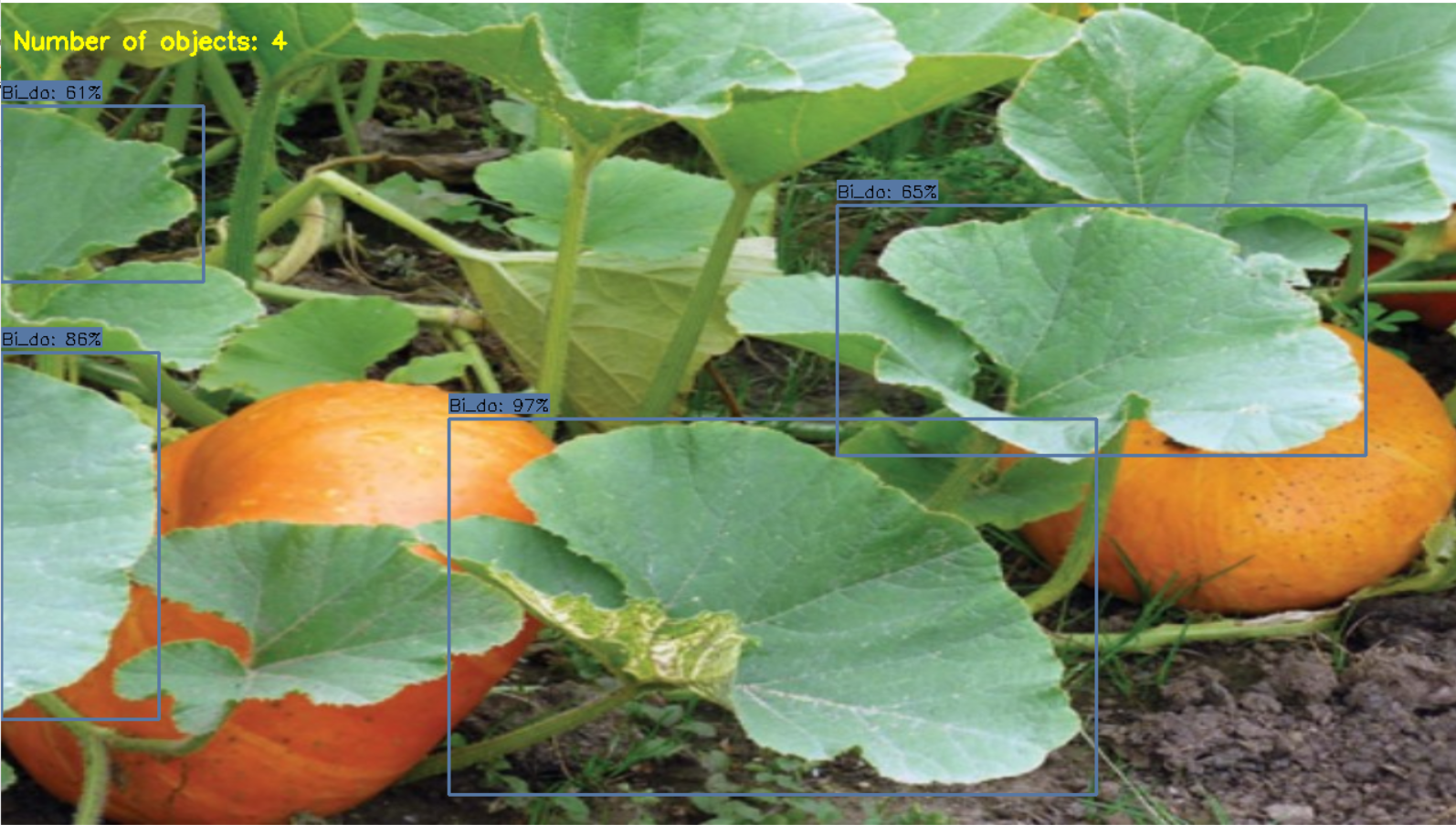
# MINH HỌA MÔ HÌNH HOẠT ĐỘNG

Sau khi huấn luyện mô hình YOLOv8 thành công, chúng em tiến hành kiểm thử trên một số ảnh lá cây mới để đánh giá khả năng hoạt động thực tế của mô hình. Kết quả như sau:

1. Ảnh đầu vào

  
*Hình ảnh lá cây đầu vào chưa qua xử lý*

1. Kết quả dự đoán từ mô hình

  
*Mô hình YOLOv8 dự đoán và khoanh vùng chính xác lá cây thuộc loại "Bí đỏ"*

✅ Mô hình xác định chính xác loại cây mà còn khoanh vùng vị trí lá cây trên ảnh, phù hợp với các ứng dụng nhận diện đối tượng thời gian thực.

1. Ứng dụng thực tế

* Nhận diện cây xanh tự động tại công viên, đô thị
* Hỗ trợ nông nghiệp thông minh, phân loại cây trồng theo vùng
* Tích hợp vào ứng dụng di động (scan lá cây → xác định loại cây

# TÀI LIỆU THAM KHẢO

1. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, O’Reilly Media.
2. Documentation: OpenCV, Scikit-learn, TensorFlow