

**DẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN**  
**HỆ THỐNG PHÁT HIỆN VÀ PHÂN LOẠI**  
**NGƯỜI KHUYẾT TẬT CHÂN TAY**

**MÔN HỌC: TƯ DUY TÍNH TOÁN**

**LỚP: CS117.O22**

**GVHD: TS. NGÔ ĐỨC THÀNH**

**Nhóm sinh viên thực hiện:**

| <b>Họ và tên</b> | <b>MSSV</b> |
|------------------|-------------|
| Nguyễn Viết Đức  | 22520273    |
| Đoàn Văn Hoàng   | 22520459    |
| Huỳnh Nhật Minh  | 22520862    |
| Lê Thành Vinh    | 22521676    |
| Lê Nguyễn Hưng   | 22520507    |

# Mục lục

|       |  |    |
|-------|--|----|
| 0.1   | Thông tin thành viên . . . . .                         | 2  |
| 0.2   | Lý do chọn bài toán . . . . .                          | 2  |
| 0.3   | Phát biểu bài toán . . . . .                           | 2  |
| 0.4   | Xây dựng hệ thống phát hiện người khuyết tật . . . . . | 3  |
| 0.4.1 | Kịch bản xây dựng hệ thống . . . . .                   | 3  |
| 0.4.2 | Phân rã bài toán . . . . .                             | 4  |
| 0.4.3 | Kiến trúc YOLOv8 . . . . .                             | 5  |
| 0.4.4 | Flow chart Algorithm . . . . .                         | 6  |
| 0.5   | Kết quả thực nghiệm . . . . .                          | 7  |
| 0.5.1 | Bộ dữ liệu . . . . .                                   | 7  |
| 0.5.2 | Dộ đo đánh giá . . . . .                               | 8  |
| 0.5.3 | Dánh giá kết quả . . . . .                             | 8  |
| 0.5.4 | Demo . . . . .   | 10 |
| 0.6   | Tổng kết . . . . .                                     | 11 |

## 0.1 Thông tin thành viên

| STT | Họ và tên       | MSSV     | Đóng góp | Đánh giá |
|-----|-----------------|----------|----------|----------|
| 1   | Nguyễn Viết Đức | 22520273 | 100%     | 10/10    |
| 2   | Đoàn Văn Hoàng  | 22520459 | 100%     | 10/10    |
| 3   | Huỳnh Nhật Minh | 22520862 | 100%     | 10/10    |
| 4   | Lê Thành Vinh   | 22521676 | 100%     | 10/10    |
| 5   | Lê Nguyễn Hưng  | 22520507 | 100%     | 10/10    |

Bảng 1: Thông tin thành viên và đóng góp vào đồ án

## 0.2 Lý do chọn bài toán

Hiện trạng của việc phát hiện và hỗ trợ người khuyết tật hiện nay vẫn còn nhiều hạn chế và thách thức. Mặc dù đã có nhiều tiến bộ về mặt công nghệ và chính sách xã hội nhằm hỗ trợ người khuyết tật, vẫn còn nhiều vấn đề cần giải quyết.

Hiện nay, việc phát hiện và hỗ trợ người khuyết tật phần lớn dựa vào các biện pháp thủ công hoặc các hệ thống đơn giản, thiếu sự tự động hóa và thông minh. Điều này làm giảm hiệu quả và độ chính xác trong việc nhận diện và hỗ trợ kịp thời cho người khuyết tật.

Hệ thống phát hiện người khuyết tật có thể giúp đảm bảo an toàn cho họ trong các tình huống khẩn cấp như hỏa hoạn, động đất hoặc các sự cố khác, giúp lực lượng cứu hộ dễ dàng xác định và hỗ trợ kịp thời.



Hình 1: Hình ảnh người khuyết tật gặp khó khăn trong việc leo cầu thang

Hệ thống phát hiện người khuyết tật có thể được áp dụng trong nhiều lĩnh vực khác nhau như giao thông, y tế, giáo dục và dịch vụ công cộng. Điều này tạo ra tiềm năng lớn cho các nghiên cứu và ứng dụng trong tương lai.

Đề tài này đòi hỏi sự kết hợp của nhiều kỹ thuật và kiến thức trong lĩnh vực máy học, xử lý hình ảnh, và phát triển phần mềm. Điều này sẽ giúp các thành viên trong nhóm phát triển toàn diện các kỹ năng kỹ thuật và nghiên cứu của mình.

## 0.3 Phát biểu bài toán

### Input:

- Một bộ dữ liệu chứa các ảnh người khuyết tật được dán nhãn 1 trong 2 loại khuyết tật “ Tay”, “Chân”
- Một ảnh chứa người bị khuyết tật 1 trong 2 loại trên

### Output:

- Ảnh chứa bounding box tại vị trí kèm nhãn của của người khuyết tật từ ảnh đầu vào

### Requirements:

- Độ chính xác phân loại > 0.7
- Mô hình có thể phát hiện và phân loại nhanh chóng (từ 1-2 giây)

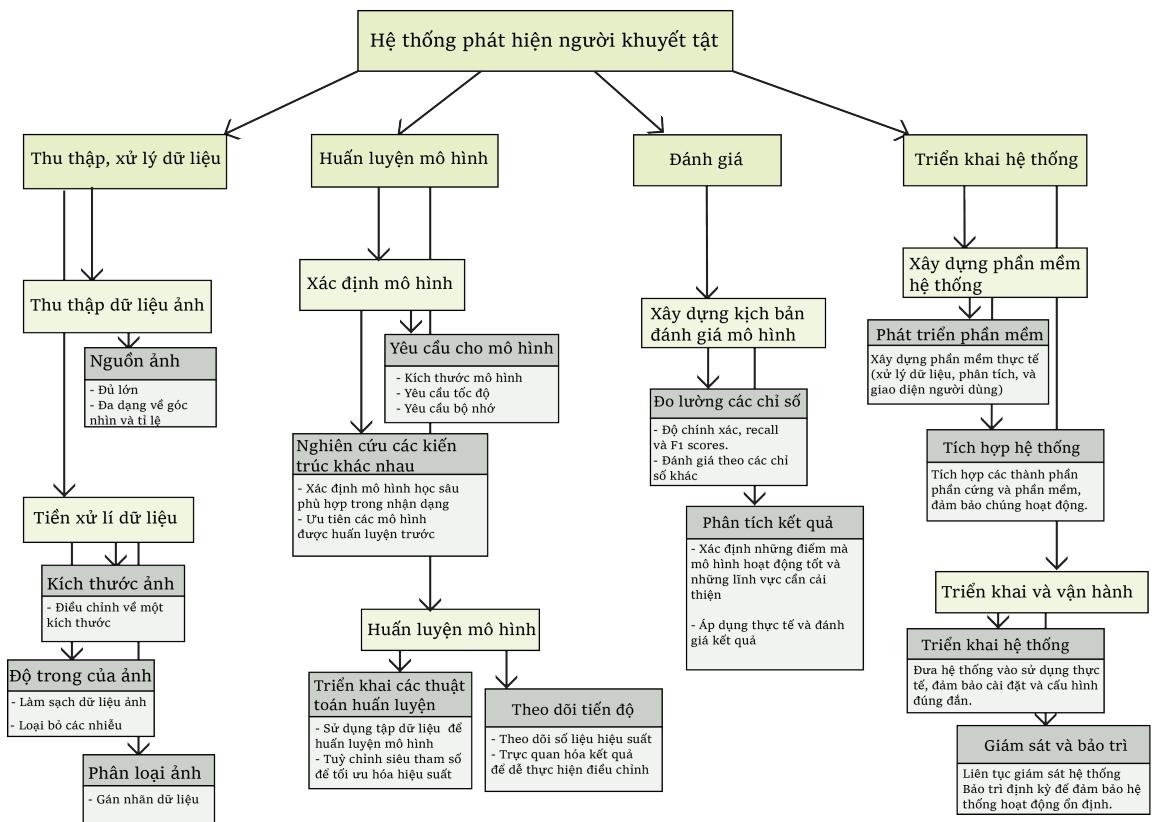
- Mô hình có hiệu suất cao
- Bounding box đề cập đến là bounding box tối thiểu bao phủ hoàn toàn đối tượng (người bị khuyết tật). Độ chính xác bounding box > 0.8
- Mô hình có thể phát hiện những người sử dụng thiết bị hỗ trợ di chuyển như xe lăn, nạng...
- Mô hình phải có khả năng nhận dạng hình ảnh người khuyết tật trong môi trường có nhiều người.

#### Constraints:

- Người khuyết tật tay, chân ở đây được định nghĩa là bị mất một phần hoặc mất hết tay, chân (từ 20% đến 100% độ dài tính từ bàn tay và bàn chân trở lên)
- Người bị liệt phải sử dụng xe lăn hoặc nạng được gán nhãn khuyết tật chân
- Hình ảnh được chụp phải có ít nhất một người khuyết tật tay hoặc khuyết tật chân
- Đối tượng cần xác định phải được chụp rõ bộ phận bị khuyết tật và không bị che khuất.
- Hình ảnh phải đủ sáng và đủ nét để có thể phát hiện vật thể.
- Hình ảnh người khuyết tật phải đủ lớn (chiếm từ 30% diện tích ảnh trở lên)
- Hình ảnh không chứa người khuyết tật mang quần áo che lấp đi bộ phận bị khuyết tật.

## 0.4 Xây dựng hệ thống phát hiện người khuyết tật

### 0.4.1 Kịch bản xây dựng hệ thống



Hình 2: Tổng quan hệ thống

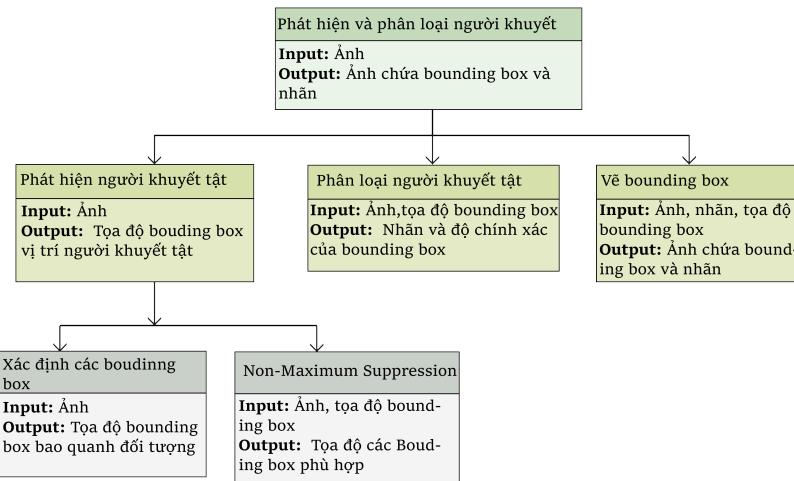
**Bước 1: Thu thập dữ liệu:** Thu thập các hình ảnh và video mẫu từ các nguồn khác nhau, đảm bảo bao gồm đủ các trường hợp của người khuyết tật (xe lăn, gãy chỏng, tay chân giả, v.v.). Dán nhãn cho dữ liệu video và hình ảnh đã thu thập để xác định rõ người khuyết tật trong các khung hình.

Bước 2: Xây dựng và huấn luyện mô hình : Sử dụng các mô hình học máy như YOLO (You Only Look Once), Faster R-CNN, hay SSD (Single Shot Multibox Detector) để phát hiện đối tượng dựa trên các yêu cầu Huấn luyện mô hình với bộ dữ liệu đã được gán nhãn. Theo dõi tiến độ mô hình, thực hiện điều chỉnh

Bước 3: Dánh giá Tính toán, đo lường các số liệu, trực quan kết quả. Phân tích kết quả

Bước 4: Triển khai hệ thống Phát triển phần mềm với giao diện người dùng. Tích hợp phần mềm với phần cứng. Chạy mô hình đã huấn luyện trên dữ liệu video thời gian thực. Liên tục thu thập dữ liệu mới và cải thiện mô hình thông qua việc huấn luyện lại định kỳ. Bảo trì và cập nhật hệ thống. Đảm bảo hệ thống hoạt động ổn định và chính xác.

#### 0.4.2 Phân rã bài toán



Hình 3: Decomposition

Chúng ta sẽ chi tiết hóa hệ thống phát hiện người khuyết tật thành ba bài toán phụ chính: Phát hiện, Phân loại, và Vẽ bounding box. Dưới đây là các chi tiết cho từng bài toán phụ, bao gồm input, output và các bước thực hiện.

##### 1. Bài Toán Phát Hiện (Detection)

- Mục tiêu: Xác định các bounding box chứa đối tượng quan tâm (người khuyết tật).
- Input: Hình ảnh cần phát hiện đối tượng.
- Output: Các bounding box (tọa độ x, y, width, height) cho mỗi đối tượng được phát hiện trong hình ảnh .

Các bước thực hiện:

- Xác định các bounding box: Sử dụng các mô hình phát hiện đối tượng như YOLO, SSD, Faster R-CNN để phát hiện vị trí các đối tượng trong hình ảnh. Mô hình sẽ đưa ra các bounding box có tọa độ và độ tin cậy (confidence score).
- Ngăn chặn không tối đa (Non-Maximum Suppression - NMS): Sau khi xác định các bounding box, sẽ có nhiều bounding box chồng chéo nhau cho cùng một đối tượng. Áp dụng kỹ thuật NMS để giữ lại bounding box có độ tin cậy cao nhất và loại bỏ các bounding box chồng chéo không cần thiết.

##### 2. Bài Toán Phân Loại (Classification)

- Mục tiêu: Phân loại đối tượng trong bounding box là người khuyết tật hay không. Nếu là người khuyết tật, xác định loại khuyết tật
- Input: Các bounding box đã được xác định từ bài toán phát hiện. Hình ảnh chứa các bounding box.
- Output: Nhãn (label) của đối tượng trong bounding box (ví dụ: người khuyết tật, xe lăn, gậy chống, tay chân giả, v.v.). Độ tin cậy (confidence score) cho mỗi nhãn.

Các bước thực hiện:

- Tiền xử lý dữ liệu: Cắt các vùng bounding box từ hình ảnh gốc để tạo thành các hình ảnh con chứa đối tượng cần phân loại.
- Phân loại đối tượng: Sử dụng các mô hình học sâu như Convolutional Neural Networks (CNN), ResNet, Inception để phân loại đối tượng trong bounding box. Mô hình sẽ đưa ra nhãn và độ tin cậy cho mỗi đối tượng.

### 3. Bài Toán Vẽ Bounding Box (Bounding Box Drawing)

- Mục tiêu: Vẽ bounding box đã xác định và phân loại lên hình ảnh gốc. Hiển thị thông tin bổ sung như loại khuyết tật, độ tin cậy của phân loại.
- Input: Hình ảnh gốc. Các bounding box đã xác định và phân loại.
- Output: Hình ảnh gốc với các bounding box được vẽ lên, kèm theo nhãn.

Các bước thực hiện:

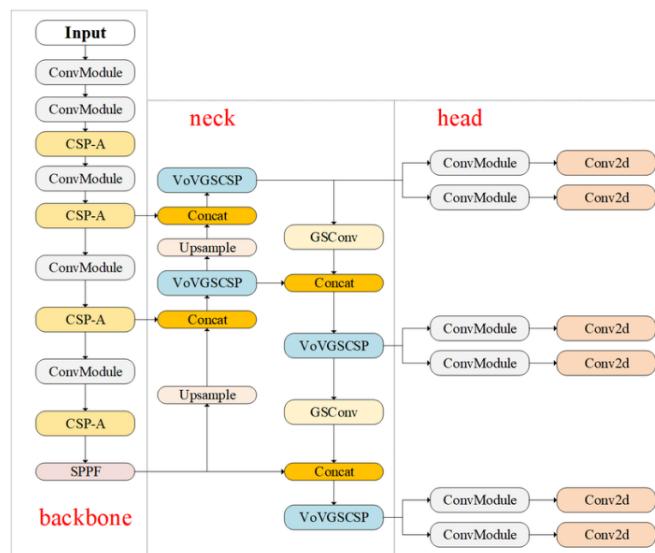
- Vẽ bounding box: Dánh giá theo độ tin cậy tùy vào ngưỡng(thresh-hold) của người sử dụng, sử dụng tọa độ bounding box để vẽ các hình chữ nhật lên hình ảnh gốc. bounding box sẽ được vẽ với màu sắc khác nhau để phân biệt các đối tượng.
- Hiển thị nhãn: Thêm nhãn của từng đối tượng lên hình ảnh, thường là phía trên bounding box.

Trong bài toán phát hiện người khuyết tật, nhóm sẽ sử dụng mô hình học máy YOLOv8 pre-train (You Only Look Once) để làm nhiệm vụ phát hiện và phân loại người khuyết tật được train bởi bộ dữ liệu riêng của nhóm.

#### 0.4.3 Kiến trúc YOLOv8

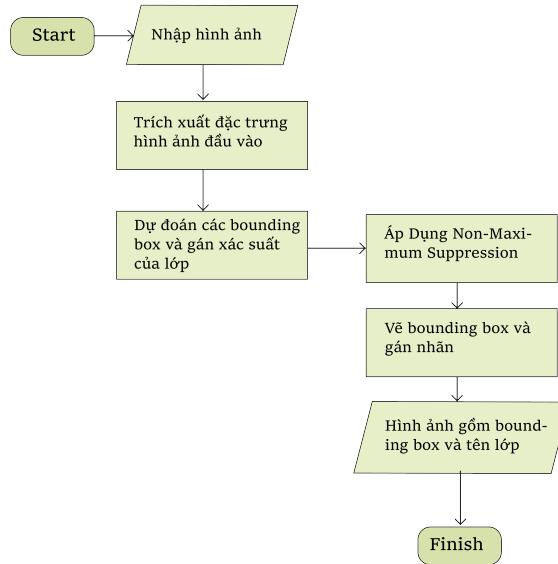
YOLO (You Only Look Once) là một mô hình mạng CNN (Convolutional Neural Network) được sử dụng cho việc phát hiện, nhận dạng và phân loại đối tượng trong ảnh. YOLO được tạo ra bằng cách kết hợp các lớp convolutional và fully-connected.

YOLOv8 (You Only Look Once version 8) là một phiên bản cải tiến của dòng mô hình YOLO, nổi tiếng trong lĩnh vực phát hiện đối tượng (object detection) và phân đoạn hình ảnh (image segmentation).



Hình 4: YOLOv8

#### 0.4.4 Flow chart Algorithm



Hình 5: Flow Chart Algorithm

Quy trình thuật toán chi tiết cho hệ thống phát hiện đối tượng sử dụng YOLOv8

1. Trích Xuất Đặc Trưng Hình Ảnh Đầu Vào: Lấy các đặc trưng từ hình ảnh đầu vào để làm cơ sở cho việc dự đoán bounding box và lớp của đối tượng.

- Backbone Network: YOLOv8 sử dụng một mạng neural sâu làm backbone để trích xuất các đặc trưng từ hình ảnh đầu vào. Backbone này thường là một mạng convolutional neural network (CNN) với nhiều lớp.
- Feature Maps: Kết quả của quá trình trích xuất đặc trưng là các feature maps, chứa thông tin quan trọng về các đối tượng trong hình ảnh.

2. Dự Đoán Các Bounding Box và Gán Xác Suất của Lớp: Dự đoán vị trí, kích thước của các bounding box và xác suất của mỗi lớp đối tượng trong các bounding box đó.

- Neck: Phần neck của mạng, như Feature Pyramid Network (FPN) hoặc Path Aggregation Network (PAN), giúp tổng hợp và nâng cao các đặc trưng từ các tầng khác nhau của backbone.
- Head: Phần head của YOLOv8 thực hiện các dự đoán cuối cùng về bounding box và lớp. Mỗi cell trong feature map dự đoán nhiều bounding box và xác suất của các lớp tương ứng với mỗi bounding box.
- Bounding Box Predictions: Mỗi bounding box được dự đoán bởi các tham số (x, y, w, h) đại diện cho tâm, chiều rộng và chiều cao của bounding box.
- Class Probability Predictions: Xác suất của mỗi lớp đối tượng được dự đoán cho mỗi bounding box.

3. Áp Dụng Non-Maximum Suppression (NMS): Loại bỏ các bounding box chồng chéo không cần thiết và chỉ giữ lại các bounding box có độ tin cậy cao nhất cho mỗi đối tượng.

- Score Thresholding: Đầu tiên, loại bỏ các bounding box có xác suất thấp hơn một ngưỡng đã định trước.
- NMS Algorithm: Sắp xếp các bounding box còn lại theo thứ tự giảm dần của xác suất. Chọn bounding box có xác suất cao nhất và loại bỏ tất cả các bounding box khác chồng chéo với nó vượt quá một ngưỡng IoU (Intersection over Union) đã định trước. Lặp lại quá trình cho đến khi không còn bounding box nào có thể loại bỏ.

4. Vẽ Bounding Box và Gán Nhãn: Hiển thị các bounding box đã phát hiện và gán nhãn cho chúng trên hình ảnh gốc.

- Drawing Bounding Boxes: Sử dụng tọa độ (x, y, w, h) của các bounding box đã qua NMS để vẽ các khung hình chữ nhật trên hình ảnh.
- Labeling: Gán nhãn cho mỗi bounding box bằng lớp đối tượng có xác suất cao nhất.

## 0.5 Kết quả thực nghiệm

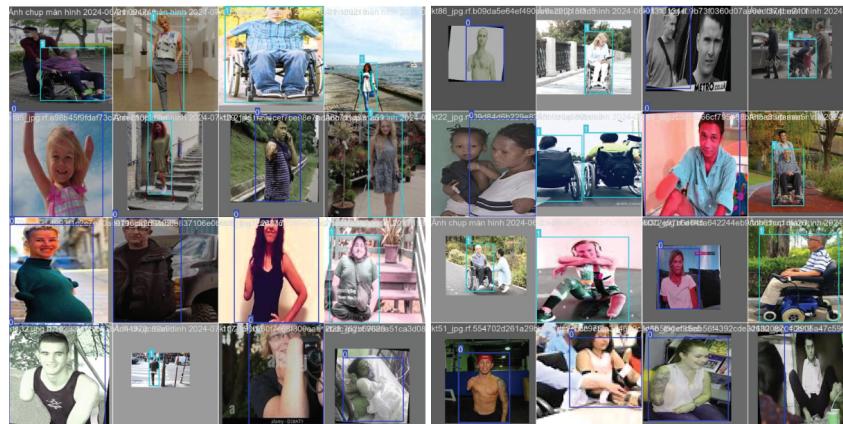
### 0.5.1 Bộ dữ liệu

Bộ dữ liệu sẽ bao gồm các bức ảnh chứa người bị khuyết tật được khoanh vùng và được gán nhãn 1 trong 2 loại khuyết tật “ Tay” hoặc “ Chân”.

Dữ liệu thô sẽ được nhóm thu thập và tổng hợp lại từ nhiều nguồn như các bức ảnh được chụp từ bộ dữ liệu liên quan hình ảnh của đối tượng khuyết tật có sẵn trên internet (chủ yếu là thu thập bằng tay) với các ràng buộc như: Ảnh phải chứa người khuyết tật; Ảnh phải đủ sáng; Người không bị vật cản che khuất; Vật thể không được quá nhỏ... đảm bảo tính đa dạng của dữ liệu, bao gồm người ở các độ tuổi khác nhau, mức độ khuyết tật khác nhau. Sau đó các bức ảnh sẽ được nhóm dán nhãn thủ công bằng công cụ gán nhãn có sẵn của trang web CVAT.

Bộ dữ liệu Disable People gồm 702 ảnh được chia thành các tập:

- Tập Train : 549 ảnh
- Tập Validation : 83 ảnh
- Tập Test : 70 ảnh



Hình 6: Tổng quan dữ liệu

## 0.5.2 Độ đo đánh giá

| Requirement                 | Metrics                                  |
|-----------------------------|--|
| Hiệu suất của mô hình cao   | Precision, Recall, mAP, Confusion Matrix |
| Độ chính xác Detection      | IoU (Intersection over Union)            |
| Độ chính xác Classification | Score                                    |
| Tốc độ xử lý mô hình        | Latency                                  |

Hình 7: Metrics

Chi tiết các metric sử dụng trong hệ thống:

**Precision** (Độ chính xác): Là tỷ lệ giữa số lượng kết quả đúng (true positives) và tổng số kết quả được mô hình dự đoán.

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

**Recall** (Độ bao phủ): Là tỷ lệ giữa số lượng kết quả đúng (true positives) và tổng số kết quả đúng có thể (true positives + false negatives).

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

- **TP** (True Positives) : là số mẫu được xác định đúng nhãn  $k$ .
- **FP** (False Positives) : là số mẫu được xác định là nhãn  $k$  nhưng thực tế không phải nhãn  $k$ .
- **FN** (False Negatives) : là số mẫu được xác định không phải là nhãn  $k$  nhưng thực tế là nhãn  $k$ .

**mAP** (Mean Average Precision): Đây là độ đo chính để đánh giá hiệu suất của mô hình phát hiện đối tượng.

- mAP tính trung bình của Average Precision (AP) trên tất cả các lớp (categories) trong tập dữ liệu.
- YOLOv8 thường sử dụng hai biến thể của mAP:
  - mAP@0.5: Tính mAP với ngưỡng IoU (Intersection over Union) là 0.5.
  - mAP@0.5:0.95: Tính mAP bằng cách lấy trung bình các giá trị mAP với các ngưỡng IoU từ 0.5 đến 0.95 (tăng 0.05 mỗi bước).

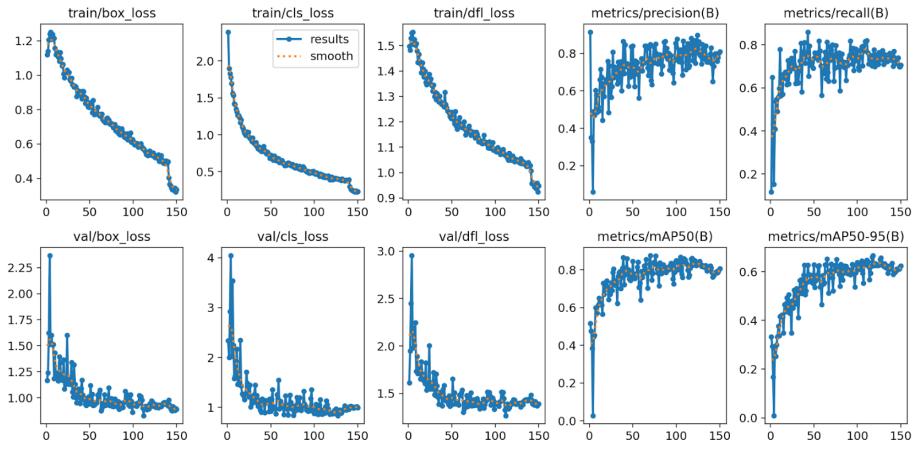
**IoU** được xác định theo công thức:

$$IoU = \frac{\text{Diện tích phần giao}}{\text{Diện tích phần hợp}} \quad (3)$$

- Diện tích phần giao là diện tích của phần chung giữa hai hình (thường là hai hình chữ nhật trong bài toán phát hiện đối tượng).
- Diện tích phần hợp là diện tích của phần bao gồm cả hai hình, không tính phần trùng lặp hai lần.

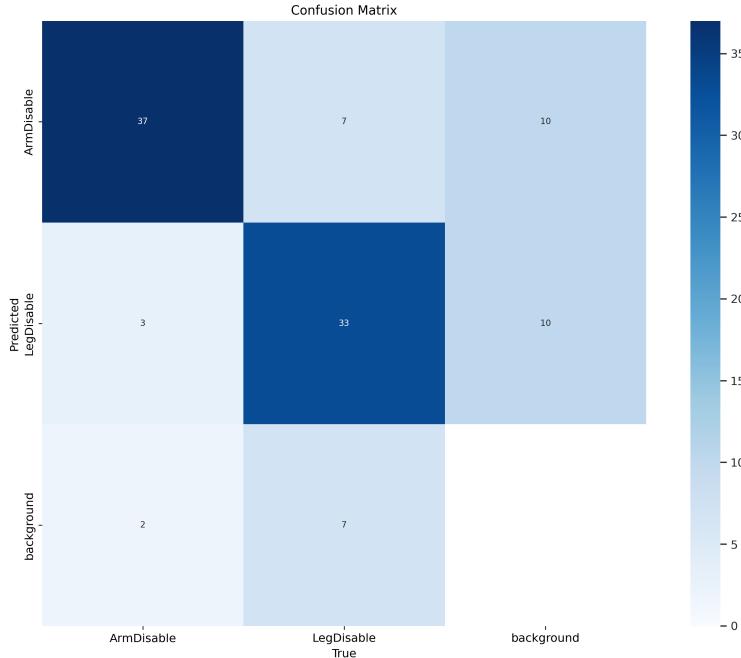
## 0.5.3 Đánh giá kết quả

Kết quả của model có kết quả training với mô hình pretrain-YOLOv8 tốt nhất là:  
Quá trình học của model:



Hình 8: Results

Confusion Matrix (Ma trận nhầm lẫn) cho phép đánh giá chi tiết hiệu suất của mô hình, bao gồm các chỉ số như True Positives, False Positives, True Negatives, và False Negatives cho từng lớp:



Hình 9: Confusion Matrix

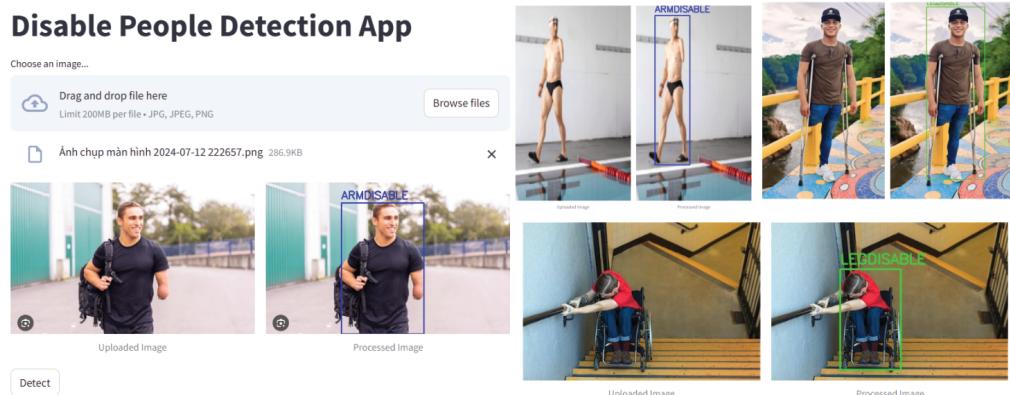
Từ Confusion Matrix ta có thể thấy:

- Mô hình đang hoạt động tốt trong việc dự đoán "ArmDisable" với số lượng kết quả đúng khá cao (37) và phân loại sai tương đối thấp.
- Dự đoán cho "LegDisable" cũng có nhiều kết quả tích cực (33), nhưng có những phân loại sai đáng chú ý, đặc biệt là với "background".
- Đối với "background", giá trị đúng (10) thấp hơn và có một số lượng đáng kể các phân loại sai là "ArmDisable" và "LegDisable".

Để cải thiện mô hình, nhóm sẽ cần tập trung vào việc giảm False Negative và True Negative, đặc biệt đối với những ảnh dự đoán không có người khuyết tật(background). Điều này có thể liên quan đến việc thiếu dữ liệu, chưa đủ feature để mô hình có thể dự đoán. Điều này nhóm sẽ cần phải điều chỉnh thêm mô hình, thu thập thêm dữ liệu hoặc áp dụng các kỹ thuật để xử lý sự mất cân bằng giữa các lớp.

#### 0.5.4 Demo

Demo hệ thống phát hiện người khuyết tật được thực hiện bằng streamlit



Hình 10: Demo

Cách cài đặt:

- Thử nghiệm trực tiếp trên trang web:  
<https://disabledetectionappapp-rpwm9p82k67mtvhbds8zrs.streamlit.app/>
- Cài đặt từ source code github:  
[https://github.com/duckviet/DisableDetection\\_StreamLitApp](https://github.com/duckviet/DisableDetection_StreamLitApp)

## 0.6 Tổng kết

Việc chọn đề tài "Hệ thống phát hiện người khuyết tật" cho đồ án không chỉ mang lại giá trị nhân văn sâu sắc mà còn đáp ứng nhu cầu cấp thiết của xã hội hiện đại. Hệ thống này giúp cải thiện chất lượng cuộc sống của người khuyết tật, đảm bảo an toàn và tạo điều kiện thuận lợi cho họ tham gia vào các hoạt động xã hội. Với sự phát triển mạnh mẽ của trí tuệ nhân tạo và máy học, đề tài này mở ra cơ hội áp dụng các công nghệ tiên tiến để giải quyết những vấn đề thực tiễn, đồng thời thách thức và phát triển toàn diện các kỹ năng kỹ thuật của sinh viên.

Nhìn chung, nhóm đã hoàn thành tốt các yêu cầu của đồ án môn Computational Thinking, mặc dù vẫn còn nhiều việc cần làm để hoàn thiện hơn. Tuy nhiên, trong quá trình thực hiện, nhóm cũng gặp phải một số khó khăn và thách thức:

Thu thập và xử lý dữ liệu:

- Khó khăn: Việc thu thập dữ liệu người khuyết tật để huấn luyện mô hình gặp nhiều trở ngại do tính riêng tư và nhạy cảm của thông tin. Dữ liệu không đủ đa dạng và phong phú có thể ảnh hưởng đến độ chính xác của hệ thống.
- Khắc phục: Nhóm có thể sử dụng các phương pháp tăng cường dữ liệu tổng hợp hoặc dữ liệu công khai có sẵn.

Phát triển và tối ưu hóa mô hình AI:

- Khó khăn: Lựa chọn và tối ưu hóa mô hình AI phù hợp là một thách thức lớn, do yêu cầu về độ chính xác cao và khả năng hoạt động trong nhiều điều kiện khác nhau.
- Khắc phục: Nhóm cần tiến hành thử nghiệm nhiều mô hình AI khác nhau, áp dụng các kỹ thuật tối ưu hóa hyperparameter tuning, và sử dụng các công cụ mạnh mẽ như TensorFlow, PyTorch để cải thiện hiệu suất của mô hình.

Tích hợp hệ thống và triển khai:

- Khó khăn: Việc tích hợp hệ thống phát hiện người khuyết tật vào các thiết bị và hệ thống hiện có đòi hỏi sự tương thích và tối ưu hóa cao.
- Khắc phục: Nhóm cần hợp tác chặt chẽ với các nhà phát triển phần cứng và phần mềm, kiểm tra tính tương thích của hệ thống trên nhiều nền tảng và thiết bị khác nhau, và đảm bảo rằng hệ thống được thiết kế linh hoạt để dễ dàng tích hợp và mở rộng.

Kiểm thử và đánh giá:

- Khó khăn: Việc kiểm thử hệ thống trong môi trường thực tế và đánh giá hiệu quả là một bước quan trọng nhưng đầy thách thức, do cần phải đảm bảo tính chính xác và an toàn.
- Khắc phục: Nhóm sẽ cần thiết lập các kịch bản kiểm thử chi tiết, thực hiện các thử nghiệm trên diện rộng và thu thập phản hồi từ người dùng để cải thiện hệ thống. Đồng thời, nhóm cũng cần xây dựng các chỉ số đánh giá rõ ràng để đo lường hiệu quả của hệ thống.

Bằng cách đổi mới và vượt qua những khó khăn này, nhóm không chỉ hoàn thiện kỹ năng kỹ thuật mà còn khẳng định vai trò của mình trong việc tạo ra các giải pháp công nghệ mang lại lợi ích cho cộng đồng.

# Tài liệu tham khảo

- [1] Streamlit Documentation. <https://docs.streamlit.io/>.
- [2] YOLOv8 - Ultralytics YOLO Documents. <https://docs.ultralytics.com/vi/models/yolov8/>.
- [3] ChatGPT. <https://chatgpt.com>.
- [4] YOLO Performance Metrics <https://docs.ultralytics.com/guides/yolo-performance-metrics/>