

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**



**UIT**  
**TRƯỜNG ĐẠI HỌC**  
**CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO CUỐI KỲ**

**CS231.N22.KHCL – NHẬP MÔN THỊ GIÁC MÁY TÍNH**

**ĐỒ ÁN: PEDESTRIAN DETECTION**

**(PHÁT HIỆN NGƯỜI ĐI BỘ)**

**Giảng viên:**

**TS. Mai Tiến Dũng**

**Sinh viên thực hiện:**

**21522110 – Bùi Mạnh Hùng**

**21522057 – Hồ Thị Khánh Hiền**

**Địa chỉ email:**

**[21522110@gm.uit.edu.vn](mailto:21522110@gm.uit.edu.vn)**

**[21522057@gm.uit.edu.vn](mailto:21522057@gm.uit.edu.vn)**

*TP. Hồ Chí Minh, ngày 30 tháng 6 năm 2023*

## This image shows a full page of white paper with horizontal dashed lines. The lines are evenly spaced and run across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the paper.

***Giảng viên***

# MỤC LỤC

<b>Phần 1: Mở đầu</b>	4
<b>I. Tổng quan</b>	4
<b>II. Lý do chọn đề án</b>	4
<b>III. Cài đặt</b>	4
<b>Phần 2: Nội dung</b>	5
<b>I. Phát biểu bài toán</b>	5
<b>II. Các công trình nghiên cứu liên quan</b>	5
1. Histogram of Oriented Gradients (HOG)	5
2. Support Vector Machine (SVM)	6
3. Faster RCNN (Faster Region Convolutional Neural Network)	7
4. Độ đo IoU (Intersection over Union)	10
5. Non Maximum Suppression (NMS)	11
<b>III. Phương pháp sử dụng</b>	11
1. Phương pháp máy học có giám sát	11
2. Phương pháp học sâu	15
<b>IV. Dataset</b>	17
1. Tổng quan	17
2. Chi tiết	17
<b>V. Đánh giá</b>	19
1. mAP (mean Average Precision)	19
<b>Phần 3: Kết luận</b>	21
<b>TÀI LIỆU THAM KHẢO</b>	22

# Phần 1: Mở đầu

## I. Tổng quan

Phát hiện người đi bộ là một lĩnh vực quan trọng và phổ biến trong lĩnh vực thị giác máy tính. Nó được sử dụng trong nhiều ứng dụng thực tế từ giám sát an ninh, hỗ trợ tự động lái xe và nhiều lĩnh vực khác

## II. Lý do chọn đề án

Phát hiện người đi bộ có thể được triển khai tại các lối băng qua đường để cải thiện an toàn cho người đi bộ và người lái xe. Chẳng hạn, camera có thể được cài đặt để giám sát vạch băng qua đường. Khi phát hiện người đi bộ, hệ thống giám sát có thể kích hoạt biển cảnh báo hoặc âm thanh để người lái xe nhường đường cho người đi bộ. Qua đó làm giảm thiểu số lượng tai nạn giao thông không mong muốn xảy ra.

## III. Cài đặt

IDE: Google Colab.

Ngôn ngữ lập trình: Python

## Phần 2: Nội dung

### I. Phát biểu bài toán

#### 1. Input

Trong nghiên cứu này, đầu vào là ảnh có chứa người đi bộ.



Hình 1. Ví dụ về ảnh đầu vào của bài toán

#### 2. Output

Đầu ra của bài toán là vị trí của người đi bộ trong ảnh đang xét được thể hiện thông qua khung hình chữ nhật, hay còn gọi là bounding box.



Hình 2. Ví dụ về ảnh đầu ra của bài toán

### II. Các công trình nghiên cứu liên quan

#### 1. Histogram of Oriented Gradients (HOG)

Trích xuất đặc trưng là một quá trình chuyển đổi dữ liệu đầu vào phức tạp thành một cách đơn giản hơn để biểu diễn dữ liệu, phù hợp hơn cho việc học máy. Trong quá trình giải nén, dữ liệu được loại bỏ dư thừa và giữ lại dữ liệu hữu ích cho bài toán.

HOG là một phương pháp mô tả đặc trưng dùng trong lĩnh vực xử lý ảnh và nhận dạng đối tượng. Phương pháp HOG được sử dụng để trích xuất đặc trưng từ một hình ảnh bằng cách tính toán độ dốc hướng của các điểm ảnh trong hình ảnh.

Nguyên lý hoạt động của phương pháp HOG sử dụng hai ma trận của độ lớn gradient (gradient magnitude) và phương gradient (gradient direction) để mô tả hình dạng của một vật thể cục bộ. Các toán tử HOG được cài đặt bằng cách chia nhỏ một bức ảnh thành các vùng con, được gọi là các ô (cells). Với mỗi cell, tính toán một histogram về các hướng của gradients cho các điểm nằm trong cell. Để cải thiện hiệu năng nhận dạng, các histogram cục bộ có thể được chuẩn hóa về độ tương phản bằng cách tính một ngưỡng cường độ trong một vùng lớn hơn cell, gọi là các khối (blocks) và sử dụng giá trị ngưỡng đó để chuẩn hóa tất cả các cell trong khối. Phép chuẩn hóa này nhằm tạo ra sự bất biến tốt hơn đối với những thay đổi trong chiếu sáng và đổ bóng.

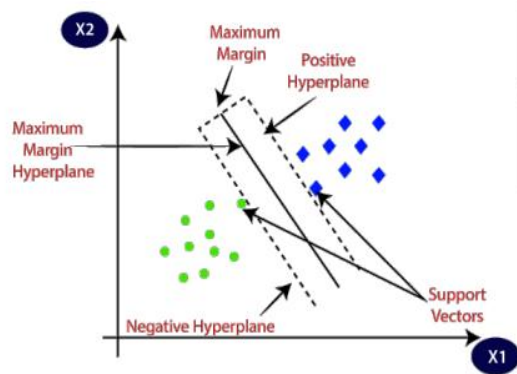
Có 5 bước cơ bản để xây dựng một vector HOG cho hình ảnh: (1) resize kích thước tất cả hình ảnh trong tập dữ liệu về một kích thước chung, (2) tính Gradient theo hai hướng  $O_x$  và  $O_y$ , (3) thu được các vote có trọng số trong các ô có định hướng và không gian, (4) chuẩn hóa lại tương phản các ô không gian, (5) kết hợp tất cả các vector trong một block để tạo ra một vector đặc trưng HOG cho toàn bộ hình ảnh.



**Hình 3.** Các bước cơ bản xây dựng một vector HOG cho hình ảnh

## 2. Support Vector Machine (SVM)

SVM là một thuật toán học máy có giám sát dùng để phân chia dữ liệu thành các nhóm riêng biệt. Phương pháp máy vector hỗ trợ SVM ra đời từ lý thuyết học thống kê do Vapnik và Chervonekis xây dựng năm 1995. Mục tiêu của SVM là xây dựng một siêu phẳng giữa hai lớp sao cho khoảng cách từ nó tới các điểm gần siêu phẳng nhất của hai lớp là cực đại.



**Hình 4.** Thuật toán máy hỗ trợ (SVM)

Các vector đặc trưng HOG thường được sử dụng làm đầu vào cho bộ phân loại SVM. Với dữ liệu huấn luyện được gán nhãn, SVM học cách phân chia các vector đặc trưng của các lớp khác nhau bằng cách tìm một siêu phẳng tốt nhất để tách chúng. Siêu phẳng này được chọn để tối đa hóa khoảng cách từ các vector đặc trưng đến siêu phẳng.

Kernel là một phần quan trọng trong SVM vì nó cho phép SVM xử lý các bài toán phân lớp phi tuyến bằng cách ánh xạ dữ liệu từ không gian gốc sang một không gian mới có số chiều cao hơn. Thay vì tìm một siêu phẳng tuyến tính trong không gian gốc, SVM sẽ tìm một siêu phẳng tuyến tính trong không gian mới được ánh xạ bằng kernel.

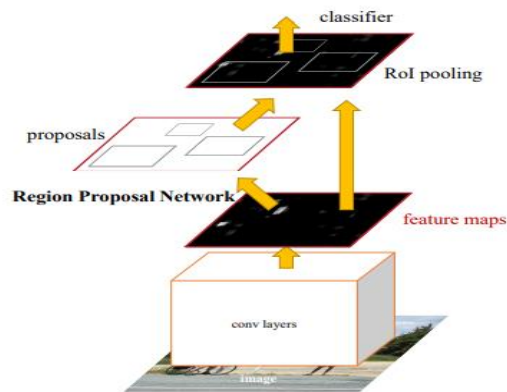
### 3. Faster RCNN (Faster Region Convolutional Neural Network)

#### 3.1. Giới thiệu

- Công bố vào năm 2015
- Các tác giả: Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun

#### 3.2. Chi tiết phương pháp

##### 3.2.1. Kiến trúc Faster R-CNN



**Hình 5.** Kiến trúc Faster R-CNN

Faster R-CNN bao gồm 2 module. Module đầu tiên là một mạng deep fully convolutional có khả năng đề xuất các vùng, và module thứ hai là Fast R-CNN detector sử dụng các vùng đã đề xuất. Toàn bộ hệ thống là một mạng đơn nhất dùng để phát hiện đối tượng. Ở phương pháp này cải tiến hơn so với hai phương pháp trước (R-CNN, Fast R-CNN) là sử dụng Region Proposal NetWork để đề xuất vùng thay vì sử dụng thuật toán Selective Search.

### 3.2.2. Region Proposal Networks

Region proposal network(RPN) là một phần quan trọng trong mô hình Faster R-CNN, có nhiệm vụ chính là đề xuất các vùng tiềm năng chứa đối tượng trong ảnh.

Trước khi có RPN, các phương pháp truyền thống thường sử dụng sliding window hoặc selective search để đề xuất các vùng quan tâm. Tuy nhiên các phương pháp này tốn nhiều thời gian và không hiệu quả trong việc xác định các vùng quan trọng. Còn với RPN có thể tạo ra các vùng quan tâm một cách nhanh chóng và chính xác.

Region proposal network(RPN) lấy đầu vào là một hình ảnh( với bất kỳ kích thước) và trả về một tập hợp hình chữ nhật các vùng đề xuất, mỗi đề xuất có điểm số về tính đối tượng.

Để tạo ra các vùng đề xuất, chúng ta trượt một mạng nhỏ qua bản đồ đặc trưng tích chập (conv feature map). Mạng nhỏ này nhận đầu vào là một cửa sổ không gian  $n \times n$  của bản đồ đặc trưng tích chập. Mỗi cửa sổ trượt được ánh xạ thành một đặc trưng chiều thấp hơn (ví dụ: 256 chiều cho ZF, hay 512 chiều cho VGG). Sau đó, đặc trưng này được đưa vào hai lớp liên kết đầy đủ (fully

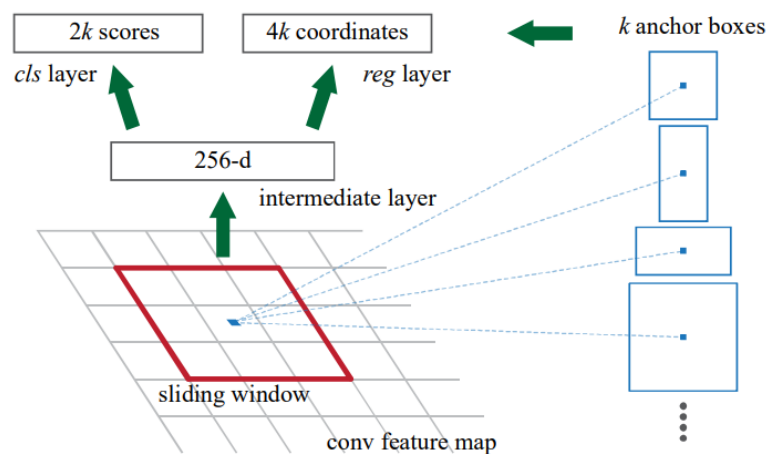


connected layers) – một lớp hồi quy (box-regression layer) và một lớp phân loại (box-classification layer)

## Anchors

Tại mỗi vị trí cửa sổ trượt, chúng ta đồng thời dự đoán nhiều đề xuất vùng, trong đó số đề xuất vùng tối đa tại mỗi vị trí được ký hiệu là  $k$ . Do đó lớp hồi quy đầu ra (box-regression layer) có  $4k$  đầu ra tương ứng với tọa độ của  $k$  hộp (mỗi tọa độ có 4 giá trị: min, max, height, width) và lớp hồi quy có  $2k$  điểm số đầu ra để ước tính xác suất của đối tượng hoặc không phải đối tượng cho mỗi vùng đề xuất.

Tham số  $k$  là liên quan đến  $k$  hộp tham chiếu, gọi là các anchors. Mỗi anchor được đặt tại tâm của cửa sổ trượt đang xét. Mặc định, thường người ta sẽ chọn 3 tỷ lệ, và 3 kích thước, nên  $k = 9$  anchors tại mỗi vị trí trượt.



**Hình 6.** Region proposal network

(256-d: ví dụ đang sử dụng model ZF nên mỗi cửa sổ trượt được ánh xạ thành một vector 256-d )

### 3.2.3. Sharing Features for RPN and Faster R-CNN

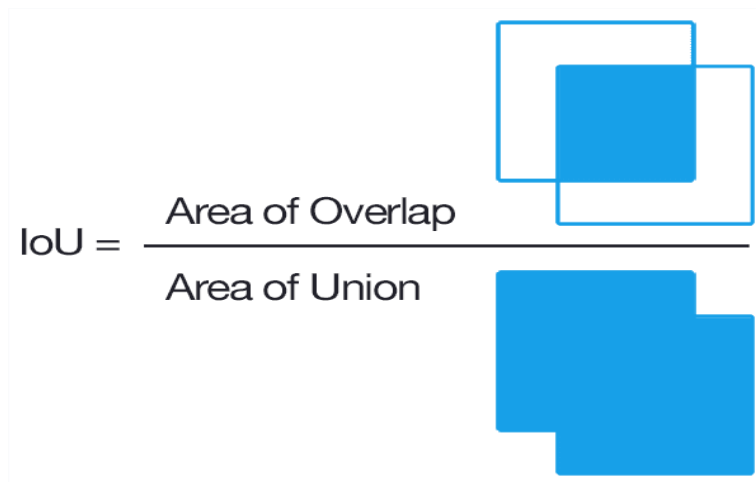
Đây là kỹ thuật được sử dụng trong mô hình Faster R-CNN (Hình 5) để chia sẻ thông tin đặc trưng giữa RPN và Fast RCNN, nhằm tăng cường hiệu suất và giảm độ phức tạp tính toán. (Fast RCNN được sử dụng trong Faster RCNN để phân loại và dự đoán vị trí chính xác của các đối tượng trong các vùng quan tâm được đề xuất bởi RPN).

Mạng CNN được sử dụng để trích xuất đặc trưng từ ảnh gốc và tạo ra các bản đồ đặc trưng(feature map).Các bản đồ đặc trưng này được chia sẻ giữa RPN và Fast RCNN. Điều này có nghĩa là cả hai module này sử dụng cùng bộ trọng số và thông tin đặc trưng để thực hiện nhiệm vụ của nó.

Việc chia sẻ đặc trưng giúp cải thiện khả năng tổng quát hóa và chia sẻ thông tin quan trọng về đối tượng trên toàn bộ mô hình. Đồng thời, nó cũng giảm số lượng tham số cần huấn luyện và giảm độ phức tạp tính toán, từ đó tăng tốc độ thực hiện các quy trình.

#### 4. Độ đo IoU (Intersection over Union)

Độ đo IoU là một phương pháp đo lường độ chồng lấp giữa hai vùng quan tâm trong lĩnh vực thị giác máy tính. Được sử dụng phổ biến trong các bài toán như phát hiện đối tượng và đo đặc hiệu suất của các mô hình, IoU cung cấp một phép đánh giá chính xác về mức độ tương đồng giữa hai vùng quan tâm.


$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

**Hình 7.** Công thức tính độ đo IoU

Như công thức tính IoU ở **Hình 7**, giá trị IoU nằm trong khoảng từ 0 đến 1. Khi hai vùng quan tâm không có sự chồng lấp, IoU sẽ bằng 0, và khi hai vùng quan tâm hoàn toàn trùng nhau, IoU sẽ bằng 1. Giá trị IoU càng gần 1, thì mức độ tương đồng giữa hai vùng quan tâm càng cao.

Độ đo IoU là một công cụ quan trọng để đánh giá chất lượng của các bài toán phát hiện đối tượng, đặc biệt trong các tác vụ có sự chồng lấp giữa các đối tượng. Đối với một hệ thống phát hiện đối tượng, việc sử dụng ngưỡng IoU nhất định cho phép xác định xem

một bounding box được coi là chính xác hay không. Các giá trị ngưỡng IoU phổ biến thường là 0.5 hoặc 0.7, tùy thuộc vào yêu cầu của bài toán cụ thể.

## 5. Non Maximum Suppression (NMS)

Phương pháp Non-Maximum Suppression là một kỹ thuật quan trọng được sử dụng trong lĩnh vực thị giác máy tính để xử lý vấn đề trùng lặp của các đối tượng trong quá trình phát hiện đối tượng. NMS giúp loại bỏ các bounding box trùng lặp và chỉ chọn ra những bounding box tốt nhất để đại diện cho mỗi đối tượng.



Hình 8. Ví dụ về NMS

Nguyên tắc hoạt động của NMS là xác định mức độ chồng lặp giữa các bounding box dự đoán (có thể sử dụng độ đo IoU). Bắt đầu bằng việc sắp xếp các bounding box theo thứ tự giảm dần của độ tin cậy dự đoán. Tiếp theo lấy bounding box có độ tin cậy cao nhất làm bounding box chính (đại diện) và loại bỏ tất cả các bounding box có chồng lặp cao hơn một ngưỡng xác định với bounding box chính. Ngưỡng chồng lặp thường được đặt trong khoảng từ 0,5 đến 0.7 tùy thuộc vào yêu cầu của bài toán cụ thể. Nếu chồng lặp giữa hai bounding box vượt qua ngưỡng, bounding box có độ tin cậy thấp hơn sẽ bị loại bỏ.

Phương pháp NMS giúp cải thiện độ chính xác và hiệu suất của hệ thống phát hiện đối tượng. NMS được sử dụng phổ biến trong các nhiệm vụ như nhận dạng đối tượng, phát hiện khuôn mặt,... hay phát hiện người đi bộ trong đồ án này.

## III. Phương pháp sử dụng

### 1. Phương pháp máy học có giám sát

Trong nghiên cứu này, chúng em sử dụng các đặc trưng HOG cho đối tượng là người đi bộ trong ảnh. Phương pháp này sẽ có hai giai đoạn: huấn luyện (training) và phát hiện (detecting).

## 1.1. Giai đoạn huấn luyện:

### 1.1.1. Tiền xử lý dữ liệu

Bước đầu tiên, chúng em tải xuống tập dữ liệu Penn-Fudan Ped và cắt ra thành những ảnh chứa người đi bộ (positive) và ảnh không chứa người đi bộ (negative) trước khi trích xuất đặc trưng. Vì SVM là một thuật toán dùng để phân chia dữ liệu thành các nhóm riêng biệt nên phải sử dụng mẫu positive và mẫu negative để mô hình có thể học được cách phân biệt đặc trưng HOG giữa hai lớp này và tìm ra siêu phẳng tốt nhất để phân loại các ảnh mới.

Trong nghiên cứu này, mẫu dữ liệu dùng để huấn luyện có kích thước 64x128 pixels.

Có tổng số 953 hình ảnh với 667 mẫu cho training và 286 mẫu cho testing.

### 1.1.2. Trích xuất đặc trưng

HOG được sử dụng để trích xuất đặc trưng, các tham số trong nghiên cứu này là pixel trên mỗi ô (8,8), ô trên mỗi khối (2,2) và 9 hướng. Vector đặc trưng được tính toán và thu được 3780 đặc trưng cho từng mẫu dữ liệu huấn luyện. Sau đó gán nhãn 1 đối với mẫu positive và 0 đối với mẫu negative.

### 1.1.3. Phân lớp

Bước tiếp theo trong giai đoạn huấn luyện này là phân loại. Chúng em sử dụng SVM để xây dựng mô hình phân loại dự đoán đối tượng có phải là người đi bộ hay không. SVM xây dựng một siêu phẳng hoặc một tập hợp các siêu phẳng trong không gian đa chiều được sử dụng cho phân loại.

Mô hình SVM có một số siêu tham số như kernel, gamma, C không thể học trực tiếp từ dữ liệu. Chúng thường được lựa chọn dựa trên một số trực giác hoặc quan sát từ thực nghiệm. Mục đích của GridSearchCV là tạo một lưới các siêu tham số và thử tất cả các kết hợp của chúng để xem thông số nào hoạt động tốt nhất. Tìm các thông số tốt nhất do GridSearchCV tìm thấy thông qua thuộc tính `best_params_` và công cụ ước tính tốt nhất thông qua thuộc tính `best_estimator_`. Trong nghiên cứu này, các siêu tham số tốt nhất được tìm thấy là **C = 0.1, gamma = 1, kernel= 'poly'**.

### 1.1.4. Đánh giá:

Sau khi tạo ra mô hình, chúng em sẽ đánh giá mô hình dựa trên 4 loại độ đo: accuracy score, precision score, recall score và F1 score.

- Accuracy score: khả năng dự đoán đúng của mô hình trên tổng các mẫu trong tập testing.
- Precision score: thể hiện sự chuẩn xác của việc dự đoán đúng, tỷ lệ càng cao thì mô hình nhận các positive càng chuẩn.
- Recall score: thể hiện khả năng phát hiện tất cả các positive, tỷ lệ càng cao thì khả năng bỏ sót các positive càng thấp.
- F1 score: là số dung hòa giữa recall score và precision score.

```
Classification report for classifier SVC(C=0.1, gamma=1, kernel='poly'):
```

	precision	recall	f1-score	support
0	0.97	0.98	0.98	200
1	0.95	0.93	0.94	86
accuracy			0.97	286
macro avg	0.96	0.96	0.96	286
weighted avg	0.96	0.97	0.96	286

**Hình 9.** Độ chính xác của mô hình trên tập dữ liệu test

## 1.2. Giai đoạn phát hiện

Trong giai đoạn phát hiện, trước khi tính toán gradient và phân loại dựa trên mô hình đã được huấn luyện, ảnh gốc được chuyển đổi về ảnh có kích thước 400x256 để áp dụng hai kỹ thuật trong xử lý ảnh là: Sliding window và Gaussian pyramid.

### 1.2.1. Sliding window

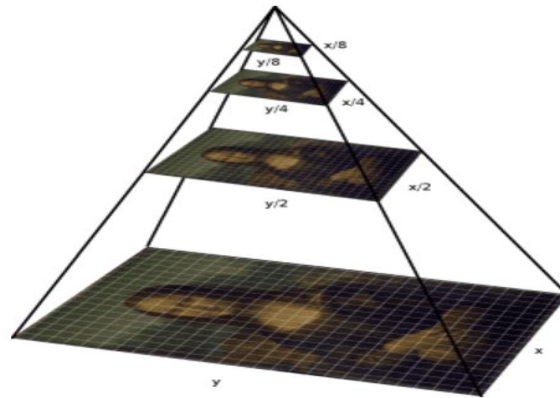
Sliding window (cửa sổ trượt) là kỹ thuật dùng một cửa sổ để trượt trên mỗi pixel của ảnh đang xét từ trái qua phải, từ trên xuống dưới. Bằng cách này sẽ bao phủ toàn bộ hình ảnh.

Khi áp dụng sliding window, một cửa sổ có kích thước cố định được di chuyển qua từng vị trí trong hình ảnh. Ở mỗi vị trí, cửa sổ sẽ được áp dụng cho bức ảnh để kiểm tra xem có sự xuất hiện của đối tượng là người đi bộ hay không.

Trong nghiên cứu này sẽ sử dụng cửa sổ có kích thước 64x128, mỗi lần cửa sổ trượt sẽ nhảy qua 8 pixels theo hai chiều ngang và dọc.

### 1.2.2. Gaussian pyramid

Gaussian pyramid là một kĩ thuật được sử dụng trong xử lý ảnh để tạo ra 1 chuỗi các ảnh được giảm kích thước theo tỉ lệ nhất định. Kĩ thuật này biểu diễn hình ảnh với nhiều tỉ lệ khác nhau.



**Hình 10.** Minh họa kim tự tháp ảnh trong Gaussian pyramid

Ở đáy là ảnh với kích thước ban đầu. Ở mỗi layer tiếp theo là ảnh đã được resize lại có kết hợp với làm mờ (Gaussian blurring). Khi nào kích thước của ảnh nhỏ hơn kích thước cửa sổ thì không giảm nữa.

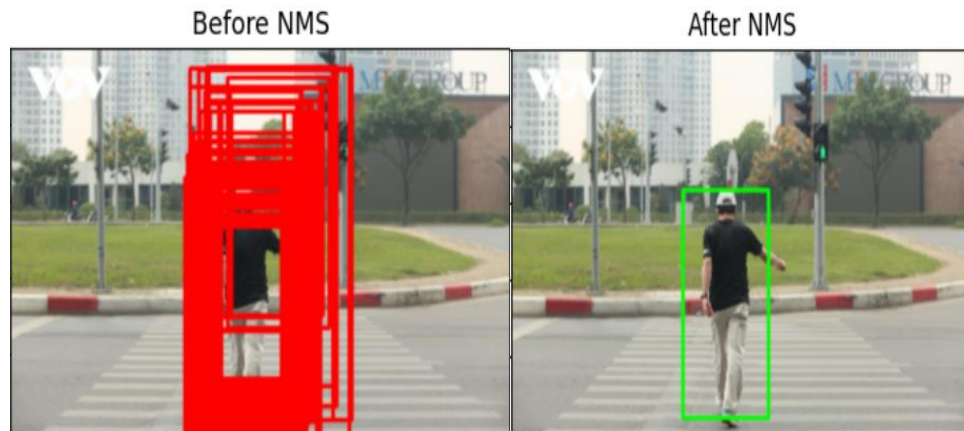
Sử dụng Gaussian pyramid cho phép tìm thấy vật thể với nhiều kích thước khác nhau trong ảnh. Kết hợp với sliding window có thể tìm được vật thể ở các vị trí khác nhau.

#### 1.2.3. Các bước thực hiện:

- Mỗi ảnh sau khi sử dụng Gaussian pyramid sẽ dùng sliding window. Do sliding window được thiết kế với kích thước cố định nên việc xử lý trên ảnh với nhiều kích thước khác nhau sẽ phát hiện người đi bộ trên nhiều kích thước khác nhau.
- Tại mỗi cửa sổ trượt trên mỗi kích thước ảnh khác nhau, tiến hành rút trích đặc trưng HOG cho cửa sổ trượt đó.
- Phân loại lớp đối tượng dựa trên mô hình đã huấn luyện.
- Từ kết quả phân loại chúng em biết được cửa sổ trượt đang xét có nhãn là 0 hay 1. Nếu cửa sổ đang xét có nhãn là 1 thì vị trí của cửa sổ đó cũng chính là vị trí bounding box của người đi bộ.

- Output sẽ có rất nhiều bounding box cho cùng một người đi bộ, gây ra sự dư thừa thông tin. Chúng em sẽ sử dụng NMS để loại bỏ đi các bounding box chồng chéo cho cùng một đối tượng trong ảnh.

Ngưỡng chồng chéo được thiết lập dựa trên độ tin cậy của mô hình trong việc dự đoán thông qua hàm `decision_function()`. Nếu sự chồng chéo của hai bounding box vượt ngưỡng chồng chéo đặt ra, bounding box có độ tin cậy thấp hơn bị loại bỏ, ngược lại bounding box có độ tin cậy cao hơn sẽ được giữ lại.



Hình 11. So sánh trước và sau khi dùng NMS

## 2. Phương pháp học sâu

Trong nghiên cứu này, chúng em sử dụng phương pháp Faster R-CNN cho bài toán phát hiện người đi bộ trong ảnh.

### 2.1. Thông tin mô hình sử dụng

- Sử dụng một mạng nơ ron tích chập(CNN): Resnet-50
  - Resnet-50 là một kiến trúc mạng nơ-ron sâu với 50 layers, được đào tạo trước trên tập dữ liệu ImageNet
- Sử dụng Feature Pyramid Network(FPN)
  - FPN được sử dụng để tạo ra các bản đồ đặc trưng có độ phân giải đa cấp trong quá trình trích xuất đặc trưng.

### 2.2. Giai đoạn huấn luyện

#### 2.2.1. Chuẩn bị dữ liệu

Từ bộ dữ liệu Penn-Fudan ban đầu, chia ra thành hai tập train-test với tỉ lệ tương ứng 8/2.

### 2.2.2. Training mô hình

Trong quá trình thực nghiệm, đào tạo lại toàn bộ mô hình (đã sử dụng pretrained) trong 15 epoch.

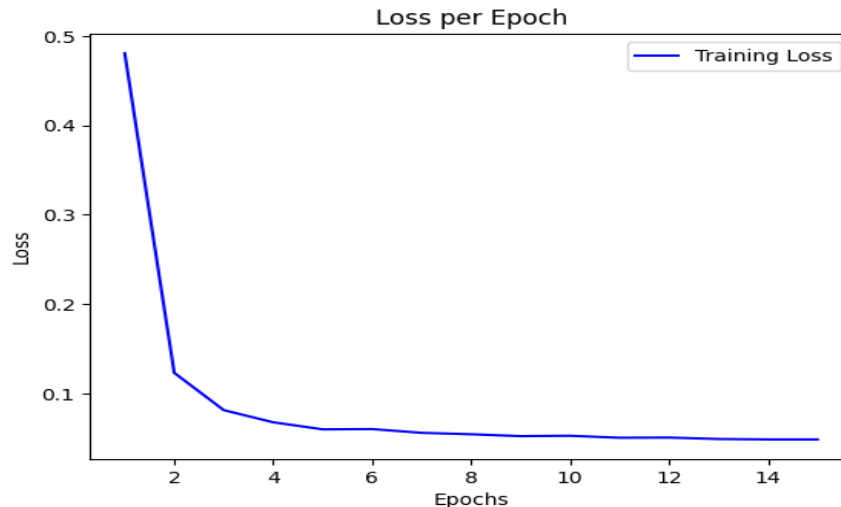
Có 4 hàm mất mát (loss function) được sử dụng, chia thành 2 hàm trong RPN và 2 hàm trong Fast R-CNN

- Trong RPN:
  - Hàm mất mát về phân loại (Classification loss): Đo lường sai số trong việc phân loại các vùng quan tâm(regions proposal) là chứa đối tượng hay không (positive hoặc negative)
  - Hàm mất mát về dự đoán vị trí (Bounding box Regression Loss): Đo lường sai số trong việc dự đoán và điều chỉnh vị trí của các bounding box đề xuất
- Trong Fast R-CNN detector:
  - Hàm mất mát đa dạng (Classification Loss của Fast R-CNN): Đo lường sai số trong việc phân loại các vùng quan tâm vào các lớp đối tượng.
  - Hàm mất mát dự đoán vị trí (Bounding box Regression Loss của Fast R-CNN): Đo lường sai số trong việc dự đoán và điều chỉnh vị trí của các bounding box cho các vùng quan tâm.

Khi train mạng Region Proposal (RPN), việc tạo ra nhiều anchors trên mỗi vị trí của sổ trượt làm cho số lượng anchors khá lớn. Nên ta sẽ phân loại các anchors dựa trên độ đo IoU vào 3 lớp: Positive, Neutral, Negative. Sau đó chỉ lấy những anchors có nhãn là positive hoặc negative vào giai đoạn tiếp theo.

### 2.3. Kết quả huấn luyện





**Hình 12.** Kết quả huấn luyện sau 15 epochs

Sau 15 epochs, giá trị loss đã giảm xuống rất thấp, nên em dừng huấn luyện mô hình để tránh bị overfitting.

#### IV. Dataset

##### 1. Tổng quan

- Tên bộ dữ liệu: Penn-Fudan Database.
- Số lượng: 170 ảnh.
- Trong đó, 96 bức ảnh chụp xung quanh Đại học Pennsylvania, 74 bức chụp xung quanh Đại học Fudan.
- Mỗi bức ảnh có ít nhất 1 người đi bộ.

##### 2. Chi tiết

Chuẩn bị dữ liệu cho quá trình Train SVM:

Vì trong tập dữ liệu chỉ có bounding box của người đi bộ trong bức ảnh, để tạo ra tập dữ liệu gồm ảnh chứa người đi bộ và ảnh chứa người đi bộ, ta thực hiện như sau:

- Đối với ảnh chứa người đi bộ
  - Dựa vào bounding box dữ liệu đã có để cắt thành những ảnh chứa người đi bộ
- Đối với ảnh không chứa người đi bộ

- Trong mỗi bức ảnh, sử dụng một window có kích thước 100x200 (kích thước này được định nghĩa dựa vào đánh giá về kích thước bức ảnh và trung bình kích thước của các bounding box trong tập dữ liệu) trượt trên toàn bộ bức ảnh. Nếu như window đó không giao với bất kỳ bounding box(chứa người đi bộ) nào trong bức ảnh thì lấy tọa độ của window tại đó để cắt ảnh.
- Để tránh việc tạo ra quá nhiều ảnh không chứa người đi bộ làm mất cân bằng giữa 2 tập dữ liệu, nên mỗi bước trượt của window theo trục x là 20 và theo trục y là 10.



**Hình 13.** Ví dụ về cắt ảnh (bounding box màu đỏ là window)

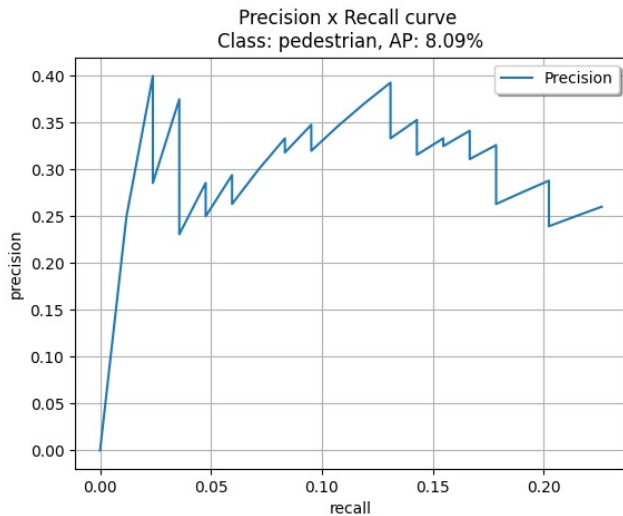
## V. Đánh giá

### 1. mAP (mean Average Precision)

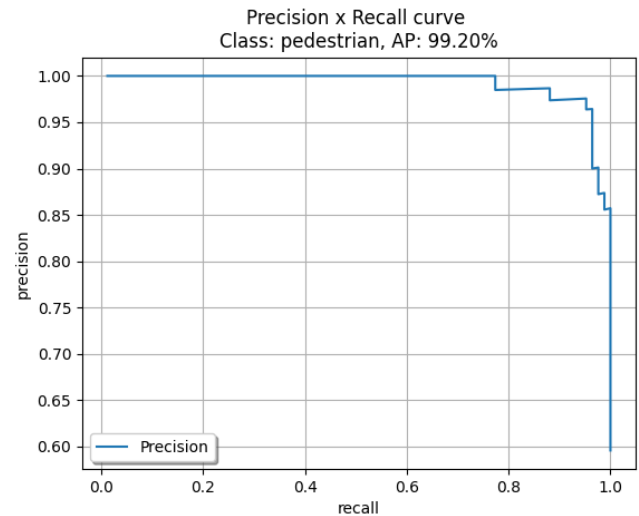
Để đánh giá 2 mô hình HOG+ SVM và Faster R-CNN, nhóm đã sử dụng độ đo mAP. Về độ đo mAP: Tính toán độ chính xác của việc phát hiện đối tượng bằng cách so sánh bounding box dự đoán và ground truth có sẵn trong tập dữ liệu test.

Phương pháp	Đánh giá
HOG+SVM	8.09%
Faster R-CNN	99.20%

**Hình 14.** Bảng tổng hợp kết quả mAP của 2 mô hình



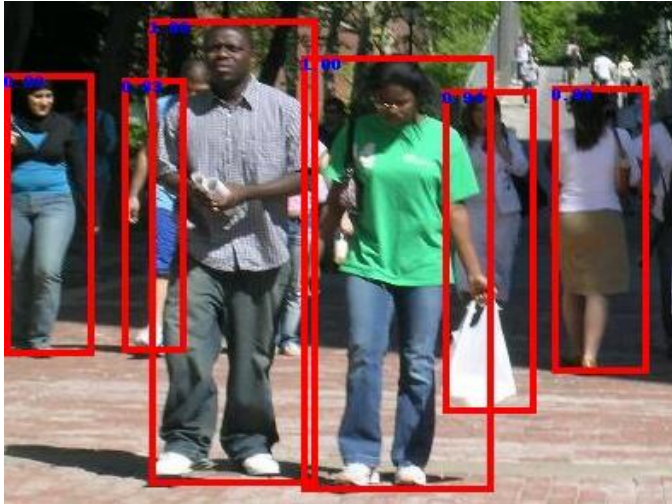
**Hình 15.** PRC của HOG+SVM



**Hình 16.** PRC của Faster R-CNN

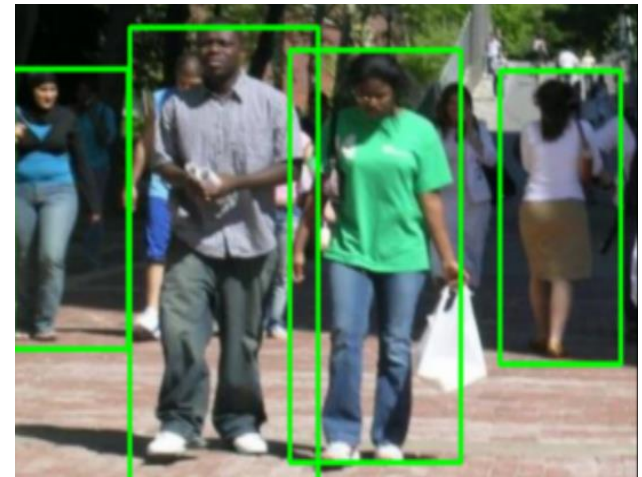
### 2. Thực nghiệm

- Đối với Faster R-CNN



**Hình 17.** Thực nghiệm của Faster R-CNN

- Đối với HOG + SVM



**Hình 18.** Thực nghiệm của HOG + SVM

## Phần 3: Kết luận

### Mô hình SVM

Một số nghiên cứu gần đây đã đề xuất các công nghệ hiện đại để đem lại hiệu quả tính toán và độ chính xác cao hơn, nhưng mục đích của việc sử dụng phương pháp đặc trưng HOG kết hợp với bộ phân loại SVM sẽ là cơ sở để so sánh giữa phương pháp truyền thống và phương pháp học sâu.

### Mô hình Faster RCNN

Trong quá trình huấn luyện, ta thấy độ chính xác đạt được khá cao, đặc biệt chỉ số mAP = 99,20 %. Nhưng trong quá trình thực nghiệm với dữ liệu bên ngoài, mô hình không chỉ nhận diện người đi bộ mà còn nhận diện người đi xe đạp, mô tô. Một phần là do sử dụng mô hình Faster R-CNN đã được đào tạo trước trên tập dữ liệu lớn sử dụng cho phát hiện người. Còn lý do khác, do có thể tập dữ liệu huấn luyện đang còn khá ít cho một mô hình mạng học sâu.

### Kết luận

Phương pháp truyền thống có thể hiệu quả trong những trường hợp đơn giản, với các dữ liệu có phân khối tách biệt rõ ràng và yêu cầu tính toán thấp. Tuy nhiên, nó có thể không đạt hiệu quả cao trong những bài toán phức tạp, có độ biến đổi lớn và yêu cầu xử lý một lượng lớn các đặc trưng.

Phương pháp học sâu thường đạt hiệu quả cao trong các bài toán phức tạp và có độ biến đổi lớn. Với khả năng học các đặc trưng từ dữ liệu đầu vào thông qua mạng neural sâu, học sâu có thể mang lại khả năng nhận diện tốt hơn so với phương pháp truyền thống.

## TÀI LIỆU THAM KHẢO

- [1] MAY THU, NIKOM SUVONVORN, MONTRI KARNJANADECHA, “Pedestrian Detection using Linear SVM Classifier with HOG Features”, in 2018 Oct.30 – Nov.2 Asia Pacific Conference on Robot IoT System Development and Platform (APRIS).
- [2] T. Surasak, I. Takahiro, C. Cheng, C. Wang, and P. Sheng, “Histogram of oriented gradients for human detection in video,” in 2018 5th International Conference on Business and Industrial Research (ICBIR).
- [3] R. Girshick, “Fast R-CNN,” in IEEE International Conference on Computer Vision (ICCV), 2015
- [4] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015.