TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG HÀ NỘI

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO CHỦ ĐỀ MÔN HỌC**

**TÍNH TOÁN MỀM**

**Tên đề tài: Nhận diện vật thể sử dụng mô hình mạng Neuron YOLO.**

**Họ tên sinh viên: Vũ Việt Hoàng Hải**

**Mã sinh viên: 1911060974**

**Lớp: ĐH9C6**

**Giảng viên hướng dẫn: Nguyễn Văn Hách**

***Hà Nội, 01 tháng 05 năm 2023***

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc133883001)

[1.1. Giới thiệu bài toán 1](#_Toc133883002)

[1.1.1. Khái niệm và cách thức hoạt động mạng neuron 1](#_Toc133883003)

[1.1.2. Cấu trúc mạng neuron yolo 2](#_Toc133883004)

[1.2. Cơ sở khoa học 3](#_Toc133883005)

[1.3. Ứng dụng và lợi ích sử dụng mô hình mạng neuron yolo nhận diện vật thể 3](#_Toc133883006)

[1.3.1. Ứng dụng 3](#_Toc133883007)

[1.3.2. Lợi ích 4](#_Toc133883008)

[CHƯƠNG 2. PHƯƠNG PHÁP VÀ CÔNG NGHỆ 5](#_Toc133883009)

[2.1. Thuật toán phương pháp và ưu điểm sử dụng cho mạng neuron yolo 5](#_Toc133883010)

[2.1.1. Hiệu suất của thuật toán và phương pháp mạng neuron yolo 5](#_Toc133883011)

[2.1.2. Ưu điểm và hạn chế từng thuật toán phổ biến 6](#_Toc133883012)

[2.2. Quy trình xây dựng mạng neuron yolo 7](#_Toc133883013)

[2.3. Công nghệ và ngôn ngữ mạng neuron yolo 8](#_Toc133883014)

[CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM ĐỀ TÀI 9](#_Toc133883015)

[3.1. Giới thiệu thuật toán áp dụng thực nghiệm đề tài 9](#_Toc133883016)

[3.2. Lợi ích của mạng neuron yolo v4 và so sánh với các thuật toán khác 9](#_Toc133883017)

[2.4.1. Lợi ích sử dụng mạng mạng neuron yolo v4 9](#_Toc133883018)

[2.4.2. Ưu điểm và hạn chế so với các thuật toán khác 10](#_Toc133883019)

[3.3. Xây dựng chương trình 11](#_Toc133883020)

[3.3. Thuật toán của chương trình 13](#_Toc133883021)

[3.4. Đánh giá 13](#_Toc133883022)

# CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

## 1.1. Giới thiệu bài toán

### 1.1.1. Khái niệm và cách thức hoạt động mạng neuron

Mạng neuron YOLO (You Only Look Once) là một mô hình mạng neuron sử dụng trong bài toán nhận diện vật thể. Mô hình này được giới thiệu lần đầu vào năm 2016 bởi Joseph Redmon và đồng nghiệp tại Đại học Washington.

Mạng neuron YOLO hoạt động dựa trên cách tiếp cận one-stage, tức là nhận diện và phân loại vật thể được thực hiện trong một bước duy nhất. Điều này khác với phương pháp two-stage, nơi bước đầu tiên là tìm kiếm vùng chứa vật thể, sau đó vùng chứa này được đưa vào để phân loại và nhận diện.

Với mô hình YOLO, ảnh đầu vào sẽ được chia thành lưới ô vuông nhỏ hơn, gọi là grid cells. Mỗi ô vuông này sẽ dự đoán nhiều hộp chứa vật thể có khả năng xuất hiện trong ô đó. Sau đó, mạng neuron YOLO sẽ sử dụng các thuật toán để phát hiện và phân loại vật thể trong từng hộp.

Điểm mạnh của mạng neuron YOLO là tốc độ xử lý nhanh, đặc biệt là so với phương pháp two-stage. Điều này là do việc áp dụng các kỹ thuật như convolutional neural network (CNN) và dimensionality reduction để giảm thiểu số lượng tính toán cần thiết. Ngoài ra, YOLO cũng có độ chính xác cao trong việc phân loại và nhận diện vật thể.

Trong quá trình phát triển, YOLO đã có nhiều phiên bản khác nhau như YOLOv1, YOLOv2, YOLOv3 và phiên bản mới nhất là YOLOv4.

* YOLOv1: Phiên bản đầu tiên được giới thiệu vào năm 2016. YOLOv1 sử dụng một mạng CNN duy nhất để dự đoán các hộp giới hạn (bounding boxes) và xác suất phân loại đối tượng trong các hộp đó. YOLOv1 cho kết quả nhanh và hiệu quả, nhưng độ chính xác không cao.
* YOLOv2: Phiên bản tiếp theo được ra mắt vào năm 2017. YOLOv2 sử dụng nhiều cải tiến như batch normalization, hiệu chỉnh số lượng các feature map và sử dụng anchor box để cải thiện độ chính xác. Kết quả của YOLOv2 tốt hơn so với phiên bản trước đó.
* YOLOv3: Phiên bản này được phát hành vào năm 2018. YOLOv3 tiếp tục cải tiến về hiệu quả và độ chính xác bằng cách sử dụng nhiều kỹ thuật mới như SPP (Spatial Pyramid Pooling) và PAN (Path Aggregation Network) để nâng cao khả năng học và tính toán của mô hình.
* YOLOv4: Phiên bản mới nhất của YOLO được ra mắt vào năm 2020. YOLOv4 được cải tiến đáng kể so với phiên bản trước đó bằng cách sử dụng nhiều kỹ thuật như CSP (Cross-Stage Partial connections), CmBN (Cross mini - Batch Normalization), và Mish activation function để cải thiện độ chính xác và tốc độ xử lý. YOLOv4 hiện đang là phiên bản có độ chính xác và tốc độ xử lý cao nhất của YOLO.

Mô hình mạng neuron YOLO (You Only Look Once) là một trong những phương pháp được sử dụng để giải quyết bài toán nhận diện vật thể. Đặc điểm của mô hình này là có thể xử lý hình ảnh nhanh hơn và đưa ra dự đoán về vị trí đối tượng một cách chính xác.

Mô hình YOLO hoạt động bằng cách chia ảnh thành nhiều ô lưới và đưa mỗi ô lưới qua mạng neuron để đưa ra dự đoán về đối tượng xuất hiện trong ô lưới đó. Tại mỗi ô lưới, mô hình sẽ đưa ra một vector đầu ra chứa các thông tin về vị trí, kích thước và độ tin cậy (confidence) của các bounding box ứng với các đối tượng có thể xuất hiện trong ô lưới đó. Để đưa ra dự đoán cho toàn bộ ảnh, các vector đầu ra tương ứng của các ô lưới sẽ được kết hợp lại với nhau.

Sau khi đưa ra dự đoán cho toàn bộ ảnh, các bounding box có độ tin cậy thấp sẽ bị loại bỏ và chỉ giữ lại những bounding box có độ tin cậy cao nhất. Việc loại bỏ các bounding box trùng lặp cũng được thực hiện để đảm bảo tính chính xác của kết quả đầu ra.

### 1.1.2. Cấu trúc mạng neuron yolo

Cấu trúc của mô hình mạng neuron YOLO gồm 2 phần chính là đầu vào (input) và đầu ra (output), và có thể được phân chia thành 3 phần con: phần tiền xử lý (preprocessing), phần trích xuất đặc trưng (feature extraction), và phần dự đoán (prediction).

Phần tiền xử lý bao gồm quá trình chuẩn hóa ảnh đầu vào và chuyển đổi định dạng ảnh về định dạng phù hợp với mô hình. Phần trích xuất đặc trưng sử dụng một mạng neuron tích chập (convolutional neural network) để học các đặc trưng của ảnh. Các lớp tích chập được sử dụng để tìm kiếm các đặc trưng của vật thể như cạnh, góc, đường cong và các đặc điểm khác. Sau đó, các lớp liên kết đầy đủ (fully connected layers) được sử dụng để phân loại các đối tượng.

Phần dự đoán của mô hình YOLO sử dụng các lớp phát hiện (detection layers) để dự đoán các hộp giới hạn (bounding boxes) xác định vị trí và kích thước của vật thể trong ảnh, cùng với xác suất để vật thể đó xuất hiện trong hình ảnh. Các hộp giới hạn được tạo ra bằng cách sử dụng kỹ thuật áp dụng trượt (sliding windows) để xác định các vị trí tiềm năng của các đối tượng trong ảnh. Sau đó, các thông tin về các hộp giới hạn và xác suất được truyền đến lớp non-max suppression để loại bỏ các hộp trùng lặp và giữ lại hộp giới hạn tốt nhất cho mỗi vật thể. Cuối cùng, kết quả của mô hình là các hộp giới hạn và xác suất tương ứng cho từng vật thể trong ảnh đầu vào.

## 1.2. Cơ sở khoa học

Mô hình mạng neuron YOLO được xây dựng dựa trên kiến trúc CNN (Convolutional Neural Network) - một trong những mô hình mạng neuron nổi tiếng nhất và hiệu quả trong việc xử lý ảnh và video.

Các lớp Convolutional trong YOLO giúp phân tích đặc trưng của hình ảnh và giữ lại các đặc trưng quan trọng nhất. Sau đó, các lớp Fully Connected Layer sẽ sử dụng các đặc trưng đó để dự đoán các bounding box và lớp của vật thể trong ảnh.

Để đảm bảo hiệu quả và tăng tốc độ xử lý, YOLO sử dụng các kỹ thuật như downsampling và max pooling để giảm kích thước của ảnh đầu vào và tăng độ chính xác của việc phân tích đặc trưng. Ngoài ra, mô hình YOLO còn sử dụng kỹ thuật skip connection để truyền thông tin từ các lớp Convolutional sâu đến các lớp Fully Connected gần đầu ra, giúp mô hình có khả năng phát hiện các vật thể nhỏ và chi tiết.

Cơ sở khoa học của YOLO là việc kết hợp giữa các kỹ thuật Deep Learning, đặc biệt là CNN, và các phương pháp phân tích ảnh truyền thống như Sliding Windows và Region Proposals. YOLO cho phép đồng thời dự đoán nhiều bounding box và lớp của các vật thể trong ảnh một cách hiệu quả và chính xác.

## 1.3. Ứng dụng và lợi ích sử dụng mô hình mạng neuron yolo nhận diện vật thể

### 1.3.1. Ứng dụng

* Xử lý hình ảnh y tế: YOLO đã được áp dụng để xử lý hình ảnh y tế, bao gồm phát hiện khối u và chẩn đoán bệnh.
* Công nghiệp sản xuất: YOLO đã được sử dụng trong ngành công nghiệp để kiểm tra chất lượng sản phẩm và kiểm tra sản xuất.
* Tự động hóa: YOLO có thể được sử dụng trong các ứng dụng tự động hóa, bao gồm robot và xe tự hành.
* Thị giác máy: YOLO được sử dụng để cải thiện khả năng thị giác máy của các ứng dụng, bao gồm nhận dạng ký tự và chữ viết tay.
* Trò chơi: YOLO đã được áp dụng trong lĩnh vực trò chơi để theo dõi và nhận dạng đối tượng và hành động của nhân vật.
* Tốc độ xử lý nhanh: YOLO sử dụng một mạng neuron duy nhất để thực hiện việc nhận diện và phân loại vật thể cùng một lúc. Điều này giúp tăng tốc độ xử lý và giảm thiểu thời gian đáp ứng so với các phương pháp khác.

### 1.3.2. Lợi ích

* Chính xác và hiệu suất cao: YOLO có khả năng nhận diện vật thể với độ chính xác cao và độ phân giải tốt, đặc biệt là trong các trường hợp đa vật thể và vật thể nhỏ.
* Độ tin cậy cao: YOLO có khả năng xử lý các vật thể bị che khuất, mờ hoặc bị giả mạo một cách hiệu quả hơn so với các phương pháp khác.
* Khả năng học tập và cập nhật: YOLO có khả năng học tập và cập nhật mô hình dễ dàng hơn so với các phương pháp khác, giúp cải thiện hiệu suất và chính xác của mô hình theo thời gian.
* Sử dụng tài nguyên ít hơn: YOLO sử dụng tài nguyên ít hơn so với các phương pháp khác, cho phép chạy trên các thiết bị có giới hạn tài nguyên, như điện thoại di động và máy tính nhúng.
* Tóm lại, phương pháp YOLO có nhiều ưu điểm vượt trội so với các phương pháp truyền thống khác trong việc nhận diện vật thể, đặc biệt là trong các bài toán thời gian thực và có tài nguyên hạn chế.

# CHƯƠNG 2. PHƯƠNG PHÁP VÀ CÔNG NGHỆ

## 2.1. Thuật toán phương pháp và ưu điểm sử dụng cho mạng neuron yolo

### 2.1.1. Hiệu suất của thuật toán và phương pháp mạng neuron yolo

Để đánh giá hiệu suất của một thuật toán nhận diện vật thể, chúng ta cần sử dụng các phương pháp đánh giá khác nhau. Các phương pháp phổ biến nhất bao gồm:

* Precision và Recall: Precision (độ chính xác) và recall (độ phủ) là hai phương pháp đánh giá độ chính xác của thuật toán. Precision được tính bằng số lượng đối tượng được phát hiện chính xác chia cho tổng số đối tượng được phát hiện. Recall được tính bằng số lượng đối tượng được phát hiện chính xác chia cho tổng số đối tượng trong ảnh.
* Average Precision (AP): Đây là phương pháp đánh giá phổ biến cho việc đánh giá hiệu suất của thuật toán nhận diện vật thể. Phương pháp này đo lường khả năng của thuật toán phát hiện đối tượng và tính toán trung bình diện tích của các đối tượng được phát hiện.
* F1 score: F1 score là một phương pháp đánh giá kết hợp giữa precision và recall. Nó tính toán trung bình điều hòa của precision và recall.
* Mean Average Precision (mAP): Đây là phương pháp đánh giá phổ biến nhất trong lĩnh vực nhận diện vật thể. Nó tính toán trung bình các giá trị Average Precision (AP) trên tất cả các lớp.

Các thuật toán khác trong lĩnh vực nhận diện vật thể cũng sử dụng các phương pháp đánh giá này để đo lường hiệu suất của chúng. Tuy nhiên, các thuật toán khác nhau có những ưu điểm và hạn chế riêng. Ví dụ, YOLOv4 có tốc độ xử lý nhanh hơn so với các thuật toán khác nhưng có thể có độ chính xác thấp hơn trong một số trường hợp. Trong khi đó, thuật toán Faster R-CNN có độ chính xác cao hơn nhưng tốc độ xử lý chậm hơn. Do đó, việc lựa chọn thuật toán phù hợp phụ thuộc vào yêu cầu của bài toán cụ thể.

Thuật toán cần dùng để giải quyết bài toán phân loại và phát hiện vật thể trong ảnh là mạng neuron YOLO v4. Mạng YOLO v4 có cấu trúc phức tạp, được xây dựng dựa trên kiến trúc mạng neuron tích chập (CNN) và các kỹ thuật tiên tiến như skip connection, residual block, spatial pyramid pooling, PAN (Path Aggregation Network), giúp cho mạng neuron này đạt được độ chính xác cao hơn so với các phiên bản trước đó.

Phương pháp của YOLO v4 là phân tích ảnh theo từng phần, mỗi phần là một cell của lưới các cell trên ảnh. Với mỗi cell, YOLO v4 sẽ dự đoán một hoặc nhiều bounding box, mỗi bounding box chứa thông tin về tọa độ, kích thước và độ tin cậy của đối tượng được phát hiện. Từ các bounding box được dự đoán, YOLO v4 sẽ tiến hành xác định loại đối tượng phát hiện được và đưa ra kết quả.

Để xây dựng mô hình YOLO v4, chúng ta có thể sử dụng ngôn ngữ lập trình Python kết hợp với các thư viện như OpenCV, NumPy, TensorFlow. Việc xây dựng mô hình YOLO v4 sẽ bao gồm các bước như thu thập và chuẩn bị dữ liệu huấn luyện, xây dựng kiến trúc mô hình YOLO v4, tiến hành huấn luyện mô hình trên tập dữ liệu và đánh giá hiệu suất của mô hình.

### 2.1.2. Ưu điểm và hạn chế từng thuật toán phổ biến

Dưới đây là một số thông tin về các thuật toán phổ biến trong lĩnh vực nhận diện vật thể cùng với ưu điểm và hạn chế của từng thuật toán:

1. Haar Cascades

* Ưu điểm: Tính đơn giản, tốc độ nhanh và độ chính xác khá cao, phù hợp với các ứng dụng yêu cầu tốc độ xử lý nhanh.
* Hạn chế: Không thể xử lý các trường hợp đối tượng đang bị che khuất, độ chính xác bị giảm khi đối tượng có góc nghiêng, ánh sáng yếu hoặc phông nền phức tạp.

1. HOG (Histogram of Oriented Gradients)

* Ưu điểm: Khả năng xử lý các đối tượng có kích thước và hình dạng khác nhau, phù hợp với các ứng dụng có yêu cầu đa dạng về đối tượng nhận diện.
* Hạn chế: Độ chính xác bị giảm khi đối tượng bị che khuất, yêu cầu bộ nhớ và thời gian tính toán cao hơn các thuật toán khác.

1. SSD (Single Shot Multibox Detector)

* Ưu điểm: Khả năng nhận diện các đối tượng nhỏ, độ chính xác cao, phù hợp với các ứng dụng yêu cầu độ chính xác cao.
* Hạn chế: Độ chính xác bị giảm khi đối tượng có kích thước lớn, yêu cầu thời gian huấn luyện và tính toán cao.

1. Faster R-CNN (Region-based Convolutional Neural Networks)

* Ưu điểm: Độ chính xác cao, khả năng nhận diện các đối tượng nhỏ, khả năng xử lý các trường hợp đối tượng bị che khuất.
* Hạn chế: Tốc độ tính toán chậm hơn các thuật toán khác, yêu cầu bộ nhớ lớn hơn.

1. YOLO (You Only Look Once)

* Ưu điểm: Tốc độ xử lý nhanh, khả năng xử lý các đối tượng có kích thước khác nhau, độ chính xác cao.
* Hạn chế: Khó xử lý các trường hợp đối tượng bị che khuất, độ chính xác bị giảm khi đối tượng có hình dạng phức tạp.

## 2.2. Quy trình xây dựng mạng neuron yolo

Quy trình xây dựng mô hình YOLO bao gồm các bước chính như sau:

Chuẩn bị dữ liệu: Trước khi xây dựng mô hình, ta cần chuẩn bị dữ liệu đầu vào. Việc này bao gồm việc tạo các file annotation chứa thông tin về vị trí của các vật thể trong ảnh và các file image chứa ảnh. Dữ liệu này phải được chuẩn hóa và tiền xử lý để phù hợp với định dạng đầu vào của mô hình YOLO.

Lựa chọn kiến trúc mô hình: Các mô hình YOLO có nhiều phiên bản khác nhau với các kiến trúc khác nhau. Một số phiên bản YOLO được sử dụng phổ biến bao gồm YOLOv1, YOLOv2, YOLOv3 và YOLOv4. Việc lựa chọn kiến trúc phù hợp là một yếu tố quan trọng để đảm bảo mô hình đạt được độ chính xác cao nhất.

Thiết lập các tham số: Các tham số của mô hình YOLO bao gồm kích thước ảnh đầu vào, số lớp phân loại, kích thước grid cell, độ chính xác ngưỡng, hệ số đóng góp của các loss function và số lần lặp lại khi huấn luyện. Việc thiết lập các tham số này đòi hỏi một số kiến thức chuyên sâu về Deep Learning và các kỹ thuật liên quan.

Huấn luyện mô hình: Sau khi đã chuẩn bị dữ liệu và thiết lập các tham số, ta có thể bắt đầu huấn luyện mô hình. Việc này yêu cầu phần cứng mạnh để tăng tốc độ huấn luyện. Trong quá trình huấn luyện, ta cần cập nhật các trọng số của mô hình để tối ưu hóa hiệu suất và độ chính xác.

Đánh giá mô hình: Sau khi huấn luyện xong, ta cần đánh giá độ chính xác của mô hình trên dữ liệu kiểm tra. Tính toán các độ đo chính xác như precision, recall và F1-score. Sử dụng các phương pháp như mean Average Precision (mAP) để đánh giá độ chính xác trên toàn bộ tập dữ liệu. Kiểm tra kết quả dự đoán trực quan bằng cách vẽ bounding box và so sánh với kết quả thực tế….

Cách sử dụng mô hình YOLO cơ bản như sau:

* Load mô hình: Sử dụng hàm cv2.dnn.readNet() để load mô hình đã huấn luyện và tạo một đối tượng net để đưa vào dữ liệu.
* Chuẩn bị dữ liệu: Đọc ảnh hoặc video từ file hoặc camera, sau đó chuyển đổi định dạng dữ liệu để đưa vào mô hình.
* Đưa dữ liệu vào mô hình và tính toán: Sử dụng đối tượng net để đưa dữ liệu vào mô hình và tính toán ra kết quả.
* Xử lý kết quả: Từ kết quả tính toán của mô hình, chúng ta có thể xử lý để vẽ bounding box lên ảnh hoặc video để nhận diện vật thể.
* Hiển thị kết quả: Cuối cùng, hiển thị ảnh hoặc video với bounding box được vẽ lên để người dùng có thể nhận diện vật thể.

Để đảm bảo hiệu suất tốt khi sử dụng mô hình YOLO, ta có thể áp dụng một số kỹ thuật tối ưu hoá như:

* Sử dụng GPU để tăng tốc độ tính toán.
* Thay đổi kích thước đầu vào của mô hình để tăng tốc độ tính toán. Việc thay đổi kích thước đầu vào có thể ảnh hưởng đến độ chính xác của mô hình.
* Áp dụng kỹ thuật non-maximum suppression để loại bỏ các bounding box trùng lặp.
* Lựa chọn các siêu tham số phù hợp cho mô hình, ví dụ như learning rate, số lượng epoch, batch size, và anchor box.

## 2.3. Công nghệ và ngôn ngữ mạng neuron yolo

Convolutional Neural Networks (CNNs): Đây là một kiến trúc mạng neuron được sử dụng để học các đặc trưng của ảnh. CNNs chủ yếu sử dụng các bộ lọc (filter) để tìm kiếm các đặc trưng cục bộ của ảnh, như cạnh, góc, vết nước, vân tay, hay chữ ký... sau đó gộp các đặc trưng này để tạo ra các đặc trưng toàn cục của ảnh.

Darknet: Đây là một framework được phát triển để huấn luyện và triển khai các mô hình Deep Learning, được sử dụng để xây dựng mô hình YOLO. Darknet cung cấp một số tính năng, bao gồm hỗ trợ CPU và GPU, hỗ trợ đọc dữ liệu từ nhiều nguồn khác nhau, hỗ trợ đa nền tảng...

You Only Look Once (YOLO): Đây là một kiến trúc mô hình CNNs sử dụng để phát hiện và phân loại đối tượng trong ảnh và video. Kiến trúc này được thiết kế để phát hiện các đối tượng nhanh chóng, với độ chính xác cao, mà không cần sử dụng nhiều tài nguyên tính toán. Mô hình YOLO được cải tiến liên tục với các phiên bản YOLOv2, YOLOv3, YOLOv4, YOLOv5, ...

OpenCV: Là một thư viện mã nguồn mở cung cấp các công cụ và thuật toán xử lý ảnh và video. OpenCV được sử dụng để đọc và xử lý ảnh đầu vào, vẽ bounding box và hiển thị kết quả.

Deep learning: đây là một lĩnh vực của trí tuệ nhân tạo, trong đó các mô hình mạng neuron được huấn luyện để thực hiện các tác vụ phức tạp như nhận diện ảnh, xử lý ngôn ngữ tự nhiên, dịch máy và nhiều hơn nữa. Để xây dựng mô hình mạng neuron YOLOv4, các công nghệ Deep learning như thuật toán lan truyền ngược (backpropagation), hàm kích hoạt ReLU, tối ưu hóa Adam, và các lớp mạng Convolutional Neural Networks (CNN) đã được sử dụng.

Python: Là một ngôn ngữ lập trình phổ biến, được sử dụng để viết đoạn code trên. Python cung cấp rất nhiều thư viện hỗ trợ cho việc phát triển mô hình YOLO, bao gồm Keras, Tensorflow, PyTorch, ….

# CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM ĐỀ TÀI

## 3.1. Giới thiệu thuật toán áp dụng thực nghiệm đề tài

YOLOv4 là một thuật toán nhận diện vật thể được phát triển bởi nhóm nghiên cứu của Joseph Redmon, cùng với nhà nghiên cứu mới Alexey Bochkovskiy. YOLOv4 là phiên bản cải tiến của YOLOv3, được đánh giá là nhanh hơn, chính xác hơn và có khả năng đối phó tốt hơn với các tình huống khó khăn hơn. YOLOv4 sử dụng một kiến trúc mạng neuron tích chập (CNN) phức tạp với nhiều đặc điểm mới và các kỹ thuật cải tiến, bao gồm:

* Một số hàm kích hoạt mới như Mish và Swish, giúp tăng tốc độ hội tụ và độ chính xác.
* Tích hợp các lớp cơ chế chú ý (Attention Mechanisms) để hỗ trợ việc xử lý đối tượng với kích thước khác nhau.
* Sử dụng các kỹ thuật tăng cường dữ liệu (data augmentation) và tăng cường mạng (network augmentation) để tăng độ đa dạng cho tập dữ liệu huấn luyện.
* Tăng kích thước mạng (network size) để tăng độ chính xác của mô hình.

So với các thuật toán trong lĩnh vực nhận diện vật thể, YOLOv4 có các lợi ích sau:

* Tốc độ xử lý nhanh hơn và hiệu quả hơn đối với dữ liệu lớn.
* Độ chính xác cao hơn và khả năng đối phó với các tình huống khó khăn hơn.
* Có thể xử lý nhiều đối tượng với kích thước khác nhau cùng một lúc.
* Dễ dàng tích hợp vào các ứng dụng thực tế.

Tuy nhiên, cũng có những hạn chế của YOLOv4, bao gồm khả năng nhận diện đối tượng nhỏ hơn so với một số thuật toán khác và yêu cầu cấu hình phần cứng mạnh hơn để đạt được tốc độ xử lý tối ưu.

## 3.2. Lợi ích của mạng neuron yolo v4 và so sánh với các thuật toán khác

### 2.4.1. Lợi ích sử dụng mạng mạng neuron yolo v4

Ngoài các ứng dụng mà đã được đề cập ở trên, thuật toán YOLOv4 cũng có nhiều ứng dụng khác trong các lĩnh vực khác nhau, bao gồm:

Robotica và tự động hóa: YOLOv4 được sử dụng để phát hiện và theo dõi đối tượng trong các hệ thống robotica và tự động hóa, giúp cho các thiết bị này có khả năng phân tích và xử lý thông tin nhanh chóng và chính xác hơn.

Giám sát động vật hoang dã: YOLOv4 có thể được sử dụng để giám sát động vật hoang dã như báo, hổ, sư tử, v.v. bằng cách phát hiện và theo dõi chúng trong tự nhiên hoặc trong các vườn thú.

Quản lý kho hàng: YOLOv4 có thể được sử dụng trong các hệ thống quản lý kho hàng để phát hiện và theo dõi các sản phẩm, giúp cho quản lý kho hàng trở nên hiệu quả và chính xác hơn.

Điều khiển giao thông: YOLOv4 có thể được sử dụng để giám sát giao thông và phát hiện các phương tiện vi phạm, giúp cho việc điều khiển giao thông trở nên an toàn và hiệu quả hơn.

Nhận dạng khuôn mặt: YOLOv4 có thể được sử dụng để nhận dạng khuôn mặt trong các hệ thống an ninh, giúp cho việc phát hiện và xử lý các hành vi vi phạm trở nên dễ dàng và nhanh chóng hơn.

YOLOv4 là một thuật toán nhận diện vật thể hiệu quả với nhiều ứng dụng khác nhau trong các lĩnh vực như an ninh, giao thông, y tế, nông nghiệp và tự động hóa.

### 2.4.2. Ưu điểm và hạn chế so với các thuật toán khác

Tiếp tục đánh giá ưu điểm và hạn chế của YOLOv4 so với các thuật toán khác trong lĩnh vực nhận diện vật thể:

1. Ưu điểm:

* YOLOv4 có tốc độ xử lý nhanh hơn so với nhiều thuật toán khác trong lĩnh vực nhận diện vật thể.
* Thuật toán có độ chính xác cao trong việc phát hiện đối tượng trong hình ảnh và video, nhất là đối với các đối tượng nhỏ và đa dạng.
* YOLOv4 sử dụng kỹ thuật backbone CSPDarknet53, giúp cải thiện hiệu suất của mô hình một cách đáng kể.
* Thuật toán có khả năng định vị và phân loại đối tượng đồng thời, giúp giảm độ phức tạp của quá trình nhận diện.

YOLOv4 có khả năng phát hiện đối tượng trên nhiều thang độ phân giải khác nhau trong cùng một ảnh, giúp cải thiện độ chính xác của mô hình.

1. Hạn chế:

* YOLOv4 vẫn còn độ chính xác thấp hơn so với một số thuật toán khác như Faster R-CNN trong việc phát hiện đối tượng nhỏ hơn.
* Khi có sự thay đổi nhanh chóng trong hình ảnh hoặc video, YOLOv4 có thể không xử lý được và dẫn đến việc phát hiện đối tượng sai.
* Do sử dụng mô hình deep learning, YOLOv4 đòi hỏi phải có tập dữ liệu huấn luyện lớn và đa dạng để có thể phát triển một mô hình chính xác và hiệu quả.
* Thuật toán không đưa ra dự đoán cho các đối tượng bị che khuất hoặc ẩn đi, gây ra sự thiếu sót trong quá trình nhận diện.

Tổng quan, YOLOv4 là một trong những thuật toán nhận diện vật thể tiên tiến nhất hiện nay, với nhiều ưu điểm vượt trội so với các thuật toán khác. Tuy nhiên, như đã đề cập, thuật toán vẫn còn tồn tại một số hạn chế và cần phải được cân nhắc khi áp dụng vào các ứng dụng thực tế.

## 3.3. Xây dựng chương trình

Để xây dựng phần mềm này, trước hết chúng ta cần chuẩn bị dữ liệu cho việc huấn luyện mô hình. Dữ liệu này bao gồm các hình ảnh với đối tượng cần nhận diện và nhãn tương ứng với mỗi đối tượng đó. Sau đó, chúng ta sử dụng các công cụ để huấn luyện mô hình YOLOv4 trên dữ liệu này để tạo ra một tập trọng số (weights) cho mô hình.

Sau khi huấn luyện xong mô hình, chúng ta có thể sử dụng đoạn code bằng python để nhận diện và vẽ bounding box cho các đối tượng trong ảnh. Để xây dựng một phần mềm hoàn chỉnh, chúng ta cần viết các hàm để thực hiện các chức năng như:

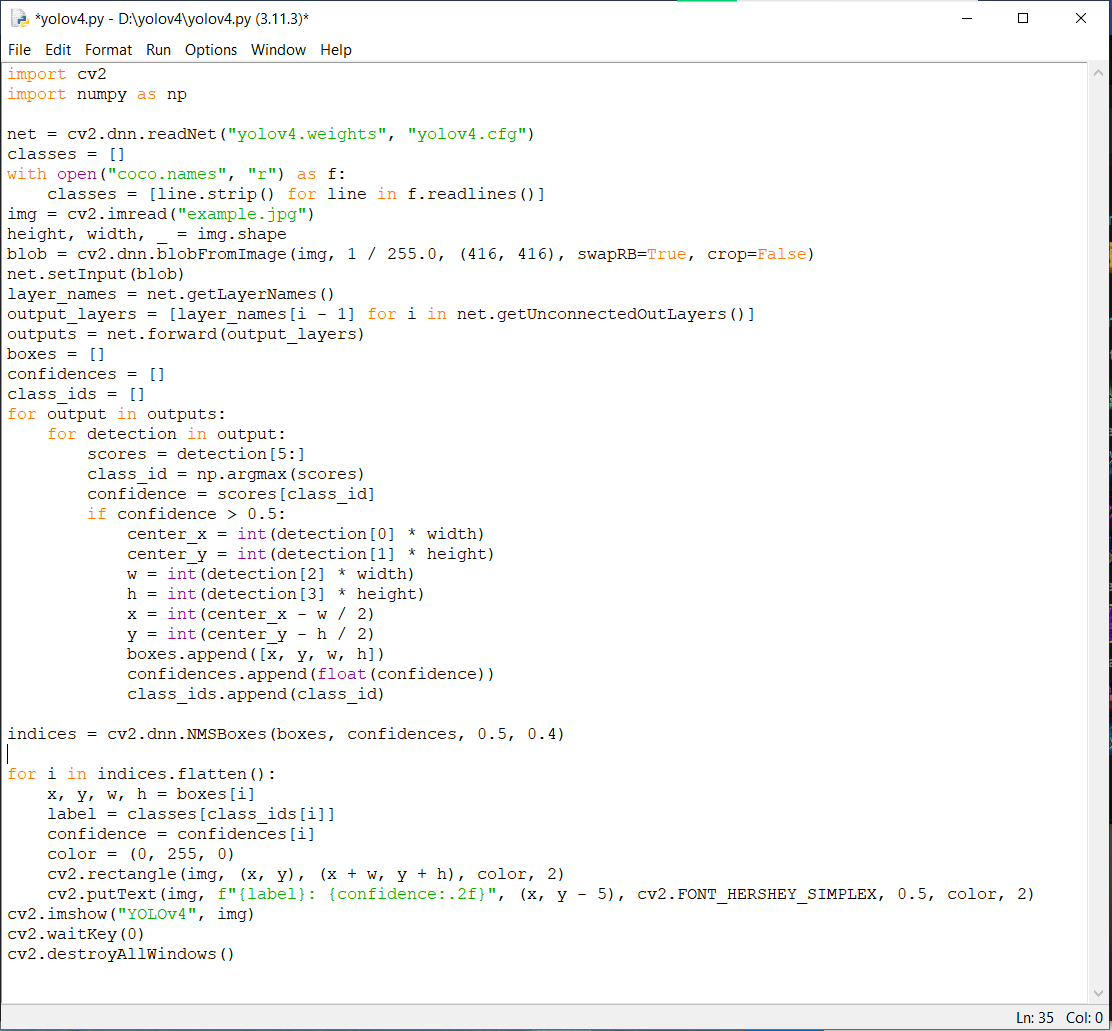
* Load model YOLOv4 từ file weights và file cấu hình
* Đọc các tên lớp từ file coco. Names.
* Đọc ảnh đầu vào từ file hoặc stream
* Thực hiện nhận diện vật thể trên ảnh đầu vào sử dụng mô hình YOLOv4
* Vẽ bounding box lên ảnh đầu ra với thông tin về tên lớp và độ tin cậy (confidence)
* Lưu ảnh đầu ra hoặc hiển thị ảnh đầu ra trên giao diện người dùng

Giao diện khi thực hiện nhận diện và đánh giá vật thể



*Hình 3.3.1. Nhận diện vật thể có trong ảnh*

Đoạn mã code sử dụng bằng python



*Hình 3.3.2. Đoạn code của chức năng*

Kết quả đạt được từ chương trình trên là một chương trình có khả năng nhận diện và phân loại các đối tượng trong ảnh sử dụng mô hình mạng neural YOLOv4. Chương trình có thể xác định tên của các đối tượng và vẽ bounding box (khung giới hạn) cho mỗi đối tượng được phát hiện trên ảnh đầu vào. Kết quả này có thể được sử dụng để giám sát, phân tích hình ảnh và ứng dụng trong nhiều lĩnh vực như an ninh, y tế, giao thông, v.v.

## 3.3. Thuật toán của chương trình

Chương trình này sử dụng thuật toán YOLOv4 để thực hiện nhận diện vật thể trong ảnh. YOLO là một thuật toán nhận diện vật thể dựa trên Deep Learning, cho phép chúng ta phát hiện và phân loại vật thể trong ảnh hoặc video một cách nhanh chóng và chính xác. YOLOv4 là phiên bản mới nhất và hiện đại nhất của thuật toán YOLO, được cải tiến về độ chính xác và tốc độ xử lý so với các phiên bản trước đó.

Trong chương trình trên, ta sử dụng hàm cv2.dnn.readNet để load mô hình YOLOv4, hàm cv2.dnn.blobFromImage để tạo input blob từ ảnh, và hàm net.forward để chạy mô hình và lấy output. Sau đó, ta áp dụng non-maximum suppression để loại bỏ các bounding box trùng lặp và vẽ bounding box lên ảnh để hiển thị kết quả.

Về cách thức hoạt động của YOLOv4, đầu tiên thuật toán chia ảnh đầu vào thành nhiều ô lưới (grid cells), mỗi ô lưới dự đoán một số bounding box và xác suất cho các lớp vật thể có thể có trong bounding box đó. Tiếp theo, thuật toán tính toán confidence score cho mỗi bounding box bằng cách nhân xác suất của lớp vật thể với IOU (Intersection over Union) giữa bounding box dự đoán và bounding box thực tế. Cuối cùng, non-maximum suppression được áp dụng để loại bỏ các bounding box trùng lặp và chỉ lấy bounding box có confidence score cao nhất cho mỗi vật thể.

## 3.4. Đánh giá

Đề tài đã trình bày về thuật toán nhận diện vật thể YOLOv4, bao gồm các khái niệm cơ bản, cách thức hoạt động và cách xây dựng mô hình YOLOv4. Ngoài ra, đề tài cũng đã giới thiệu về các phương pháp đánh giá hiệu suất của thuật toán và so sánh với các thuật toán khác trong lĩnh vực.

Từ đó, chúng ta đã có cái nhìn tổng quan về YOLOv4 và sức mạnh của thuật toán trong việc nhận diện vật thể trong ảnh và video. Đặc biệt, YOLOv4 có thể giúp giải quyết các vấn đề thực tiễn như an ninh, giao thông, y tế và nông nghiệp.

Tuy nhiên, để hiểu rõ hơn về sự phát triển của thuật toán và cách thức sử dụng YOLOv4 trong các ứng dụng cụ thể, đề tài cần bổ sung thêm thông tin về các thuật toán khác trong lĩnh vực nhận diện vật thể. Các thuật toán này có thể bao gồm R-CNN, Fast R-CNN, Faster R-CNN và SSD.

Bên cạnh đó, đề tài cũng có thể cung cấp thêm ví dụ về các ứng dụng khác của thuật toán YOLOv4, như trong lĩnh vực giám sát hành lang sản xuất, giám sát người lái xe, hay nhận diện đối tượng trong video giám sát. Ngoài ra, cũng cần đưa ra đánh giá cụ thể về ưu điểm và hạn chế của YOLOv4 so với các thuật toán khác trong lĩnh vực nhận diện vật thể.

Tổng thể, đề tài đã trình bày một cách khá chi tiết về thuật toán YOLOv4, tuy nhiên cần bổ sung thêm thông tin để làm rõ hơn về vị trí và vai trò của YOLOv4 trong lĩnh vực nhận diện vật thể.