**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**TRƯỜNG ĐIỆN – ĐIỆN TỬ**

--□&□--

****

**ĐỒ ÁN TỐT NGHIỆP**

Sinh viên thực hiện

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Bùi Thanh Phong | 20193051 | Điện tử 05 – K64 |
|  |  |  |
|  |  |  |

Hà Nôi, 2/2024

# MỤC LỤC

[MỤC LỤC 1](#_Toc168277413)

[LỜI NÓI ĐẦU 2](#_Toc168277414)

[DANH MỤC HÌNH ẢNH 3](#_Toc168277415)

[I. GIỚI THIỆU MÔ HÌNH YOLO 3](#_Toc168277416)

[II. MÔ HÌNH YOLOFM 12](#_Toc168277417)

[1. Kiến trúc của mạng YOLOFM 12](#_Toc168277418)

[2. So sánh với các mô hình khác 13](#_Toc168277419)

[III. TRIỂN KHAI HỆ THỐNG 14](#_Toc168277420)

[1. Triển khai Web – Server 14](#_Toc168277421)

[1.1 Truyền hình ảnh và phát hiện cháy 15](#_Toc168277422)

[1.2 Tương tác điều khiển robot 15](#_Toc168277423)

[2. Triển khai phần cứng 16](#_Toc168277424)

[2.1 Sơ đồ mạch triển khai phần cứng 16](#_Toc168277425)

[2.2 Máy tính nhúng Raspberry Pi 4B & Camera 17](#_Toc168277426)

[2.3 Vi điều khiển Arduino UNO R3 21](#_Toc168277427)

[2.4 USB to TTL CP2102 23](#_Toc168277428)

[2.5 Các thành phần khác 24](#_Toc168277429)

[IV. KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ 32](#_Toc168277430)

[1. Model YOLOFM phát hiện cháy 32](#_Toc168277431)

[2. Phần điều khiển chuyển động và chữa cháy 32](#_Toc168277432)

[2.1 Kịch bản di chuyển tự động 32](#_Toc168277433)

[PHỤ LỤC 33](#_Toc168277434)

# LỜI NÓI ĐẦU

# DANH MỤC HÌNH ẢNH

[Hình 1. Kiến trúc mạng YOLOFM 12](#_Toc168277437)

[Hình 2. Sơ đồ khối của hệ thống 14](#_Toc168277438)

[Hình 3. Giao diện chung Web – Server 15](#_Toc168277439)

[Hình 4. Giao diện tương tác điều khiển robot 16](#_Toc168277440)

[Hình 5. Sơ đồ mạch triển khai phần cứng 16](#_Toc168277441)

[Hình 6. Mô hình phần cứng 17](#_Toc168277442)

[Hình 7. Raspberry Pi 4 18](#_Toc168277443)

[Hình 8. Raspberry Pi 4 Pinout 19](#_Toc168277444)

[Hình 9. GPIO Pin Raspberry Pi 4 - Terminal 20](#_Toc168277445)

[Hình 10. Module camera V2 cho Raspberry Pi 21](#_Toc168277446)

[Hình 11. Vi điều khiển Arduino UNO R3 22](#_Toc168277447)

[Hình 12. USB to TTL CP2102 23](#_Toc168277448)

[Hình 13. Module điều khiển động cơ L293D 24](#_Toc168277449)

[Hình 14. Động cơ DC Servo JGB37-520 26](#_Toc168277450)

[Hình 15. Động cơ DC Servo JGB37-520 26](#_Toc168277451)

[Hình 16. Bánh xe mecanum 27](#_Toc168277452)

[Hình 17. Pin sạc Lithium 18650 29](#_Toc168277453)

[Hình 18. Module hạ áp LM2596 29](#_Toc168277454)

[Hình 19. Nguồn UPS HAT Raspberry Pi 30](#_Toc168277455)

[Hình 20. Máy bơm mini MB370 31](#_Toc168277456)

# DANH MỤC TỪ VIẾT TẮT

YOLO

CNN

# GIỚI THIỆU CHUNG

# I. GIỚI THIỆU MÔ HÌNH YOLO

**YOLO – You only look once**

**Độ chính xác: không phải chính xác nhất nhưng tốc độ gần như realtime**

**Mục tiêu của mô hình: Có thể dự báo nhãn và xác định vị trí của vật thể (có thể nhiều nhãn trên một bức ảnh)**

**Lịch sử:**

**2015: YOLO**

**2016: YOLOv2**

**2018: YOLOv3**

**Yêu cầu:**

**Convolutional Neural Network (Mạng Nơ-ron tích chập): Kết hợp vs Maxpooling để trích xuất đặc trưng ảnh tốt hơn**

**Khái niệm về Bounding box: Khung hình bao quanh vật thể, và Anchor box: khung hình có kích thước xác định trước, có tác dụng dự đoán bounding box**

**Feature map: Một khối output mà ta sẽ chia thành một lưới ô vuông, phát hiện vật thể trên từng cell**

**Non-max Suppression: Phương pháp giúp giảm thiểu nhiều bounding box overlap về một bounding box có xác suất lớn nhất**

**Kiến trúc mạng YOLO**

A diagram of a diagram of a box

Description automatically generated**Bao gồm base network là các mạng convolution làm nhiệm vụ trích xuất đặc trưng. Phần phía sau mà những Extra Layers được áp dụng để phát hiện vật thể trên feature map của base network**

**Như trên hình, thành phần Darknet là base network, đầu ra của base network là feature map có kích thước 7x7x1024. Các feature map này là input của các Extra layer, có thể dự dự đoán được các nhãn và tọa độ bounding box của vật thể.**

**Base network của YOLO sử dụng chủ yếu là các lớp tích chập, trong đó, các lớp fully connected có tác dụng sự đoán xác xuất và rọa độ của đối tượng**

**Thuật toán của YOLO**

**- Residual block**

**Ảnh đầu vào được chia thành các ô có kích thước bằng nhau**

A group of people crossing a street

Description automatically generated

**- Bounding box regression**

**Bài toán dự đoán khi các ngõ ra có kích thước liên tục**

A car driving down a street

Description automatically generated

**Mỗi bounding box có các thuộc tính:**

**Chiều rộng bw**

**Chiều cao bh**

**bx,by là tâm của bounding box**

**c: nhãn**

**- Intersection over union (IOU) – là hàm đánh giá độ chính xác của Object detector trên tập dữ liệu cụ thể**

A diagram of an area of overlap

Description automatically generated

**Nếu IoU > 0.5 => khá tốt**

A cat lying down with a rectangle in the middle

Description automatically generated

**Xanh dương: Dự đoán**

**Xanh lá cây: Vật thật**

**Output của YOLO**

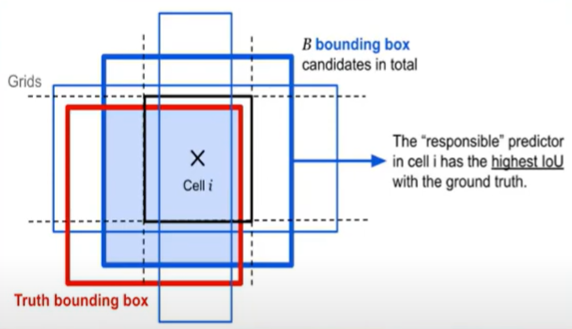
**ANCHOR BOX**

**Là 1 bounding cơ sở được xác định trước và bao quanh vật thể**

**Thuật toán Regression bounding box sẽ tinh chỉnh lại anchor box**

**Hình ảnh huấn luyện đc phân bố về một anchor box**

**Trường hợp có nhiều bounding box bao quanh vật thể, ta sẽ xác định anchor box có IoU với ground truth bounding box cao nhất**



**Như hình trên, Cell i xác định được 3 anchor box viền xanh, tuy nhiên chỉ anchor box có đường màu xanh dày nhất là được lựa chọn làm anchor box cho vật thể bới nó có IoU so với ground truth bounding box là cao nhất.**

**Như vậy, khi xác định vật thể, cần xác định 2 thành phần gắn liền vs nó là cell và anchor box**

**BOUNDING BOX**

**Là khung hình bao chứa vật thể được xác định trong quá trình huấn luyện**

**Để xác định 1 bounding box cho vật thể ta dựa trên 1 phép biến đổi từ anchor box và cell**

**YOLOv2 vs v3 dự đoán bounding box sao cho không lệch vị trí trung tâm quá nhiều.**

A diagram of a square with letters and numbers

Description automatically generated

**tx, ty :độ lệch so với góc trên cùng bên trái của cell**

**NON-MAX SUPPRESSION**

**Thuật toán YOLO dự đoán nhiều bounding box nên với những cell có vị trí gần nhau, tình trạng các khung hình bị overlap cao**

**=> Để làm giảm số lượng khung hình**

A truck driving on a street

Description automatically generated

A black truck on a street

Description automatically generated

**B1: Lọc bỏ các bounding box có xác suất chứa vật thể nhỏ hơn 1 ngưỡng nào đó (thường là 0.5)**

**B2: Đối với các bounding box giao nhau sẽ chọn cái có xác suất chứa vật thể lớn nhất, sau đó tính IoU với các bounding box còn lại. Nếu chỉ số này lón hơn ngưỡng chứng tỏ 2 2 bounding box đang giao nhau rất cao, ta giữ lại bounding box có xác suất cao nhất.**

# II. MÔ HÌNH YOLOFM

Trong phạm vi đồ án này, em sử dụng mô hình YOLOFM, một thuật toán phát hiện lửa và khói đã được cải thiện, dựa trên mô hình YOLOv5n. Cấu trúc, những ưu, nhược điểm của mô hình này sẽ được trình bày chi tiết ngay dưới đây.

## 1. Kiến trúc của mạng YOLOFM

Kiến trúc của mạng YOLOFM bao gồm 4 phần: Input, Backbone, Neck, Head, được thể hiện như hình dưới.

A diagram of a block diagram

Description automatically generated

Hình 1. Kiến trúc mạng YOLOFM

Sau khi tiền xử lý, hình ảnh đầu vào được thay đổi kích thước về (640, 640, 3). Sau đó, các hình ảnh đã tiền xử lý được chuyển đến mạng Backbone YOLOFM.

Ban đầu, thực hiện hai phép biến đổi tích chập thông thường trên các đặc trưng có hình dạng (640, 640, 3), tạo ra các đặc trưng với kích thước (320, 320, 12) và (320, 320, 64). Tiếp theo, thực hiện ba quá trình trích xuất đặc trưng FocalNext + CBS, tạo ra các đặc trưng với kích thước lần lượt là (160, 160, 128), (80, 80, 256) và (40, 40, 512). Tiếp theo, thực hiện phép biến đổi FocalNext+SPPF để có được một đặc trưng với kích thước (20, 20, 1024). Trong giai đoạn trích xuất đặc trưng, YOLOFM tạo ra nhiều lớp đặc trưng để phát hiện đối tượng, dẫn đến tổng cộng ba lớp đặc trưng. Ba lớp đặc trưng này nằm ở các vị trí khác nhau trong mạng Backbone, cụ thể là lớp giữa, lớp giữa-thấp, và lớp dưới cùng. Kích thước của ba lớp đặc trưng như sau: feature1 = (80, 80, 256), feature2 = (40, 40, 512), và feature3 = (20, 20, 1024).

Mạng FPN được hình thành trong mạng Neck sau khi có được ba lớp đặc trưng đã chứng minh là hữu ích. Cách xây dựng bao gồm thực hiện phép biến đổi tích chập 1 × 1 GhostConv trên lớp đặc trưng feature3, có kích thước (20, 20, 1024), để thay đổi kênh và tạo ra feature5. Feature5 sử dụng các phép biến đổi chéo để thực hiện phép lấy mẫu tăng và sau đó kết hợp kết quả với feature2, có kích thước (40, 40, 512). Tiếp theo, nó sử dụng QARepNeXt để trích xuất các đặc trưng, dẫn đến lớp đặc trưng feature5\_Transpose với kích thước (40, 40, 512). Feature5\_Transpose = (40, 40, 512) thực hiện phép biến đổi tích chập 1 × 1 GhostConv để thay đổi kênh và thu được feature4. Feature4 sau đó được lấy mẫu tăng và kết hợp với feature1 = (80, 80, 256) bằng cách sử dụng phép biến đổi chéo. Sau đó, QARepNeXt được sử dụng để trích xuất các đặc trưng, dẫn đến lớp đặc trưng feature3\_out = (80, 80, 256). Feature3\_out = (80, 80, 256) sử dụng phép biến đổi tích chập QARepVGGB 3 × 3 để lấy mẫu giảm, sau đó kết hợp feature3 đã lấy mẫu giảm với feature4. QARepNeXt sau đó được sử dụng để trích xuất các đặc trưng, thu được lớp đặc trưng feature2\_out với kích thước (40, 40, 512). Feature2\_out = (40, 40, 512) thực hiện phép biến đổi tích chập QARepVGG 3 × 3 để giảm kích thước và sau đó kết hợp nó với feature5 sau khi lấy mẫu giảm. QARepNeXt sau đó được sử dụng để trích xuất các đặc trưng, thu được lớp đặc trưng feature1\_out với kích thước (20, 20, 1024). Mạng FPN tạo ra ba đặc trưng quan trọng: (20, 20, 1024), (40, 40, 512), và (80, 80, 256). Sau đó, chúng tôi sử dụng ba đặc trưng này làm đầu vào cho mạng đầu YOLOFM để thu được kết quả dự đoán. Vì các loại của chúng tôi bị giới hạn trong "lửa" và "khói", hình dạng kết quả của ba lớp đặc trưng cuối cùng là (20, 20, 21), (40, 40, 21), và (80, 80, 21).

### 1.1 FocalNext network

FocalNext là một trong những thành phần quan trọng trong kiến trúc mạng Backbone của YOLOFM. Mục tiêu chính của FocalNext là cải thiện khả năng trích xuất thông tin đa tỷ lệ, giúp mạng có thể hiểu và nhận diện tốt hơn các vật thể trong những môi trường phức tạp. FocalNext sử dụng một module tiên tiến có tên là **FocalNextBlock**, được phát triển dựa trên kiến trúc CFNet. FocalNextBlock không chỉ tối ưu hóa khả năng tổng hợp thông tin từ các đặc trưng không gian và ngữ nghĩa, mà còn giảm đáng kể số lượng tham số, từ đó làm cho mô hình trở nên nhẹ hơn và dễ triển khai hơn trên các thiết bị tài nguyên hạn chế.

Điểm nổi bật của FocalNextBlock nằm ở khả năng tích hợp thông tin từ nhiều tầng đặc trưng khác nhau mà không làm mất đi thông tin quan trọng. Nó sử dụng các kỹ thuật gọn nhẹ để giảm bớt các phép tính dư thừa, từ đó tăng tốc độ xử lý và hiệu quả tổng thể của mô hình. Đồng thời, FocalNext kết hợp với **SPPF (Spatial Pyramid Pooling Fast)**, một phương pháp pooling tiên tiến. SPPF cho phép mạng gộp thông tin từ các vùng không gian khác nhau, giúp cải thiện khả năng nhận diện các vật thể có kích thước và hình dạng đa dạng. SPPF được thiết kế để giảm tải tính toán nhưng vẫn giữ nguyên độ phong phú của thông tin.

Quá trình trích xuất đặc trưng trong FocalNext tạo ra ba lớp đầu ra với kích thước lần lượt là (80, 80, 256), (40, 40, 512), và (20, 20, 1024). Những lớp đặc trưng này không chỉ cung cấp thông tin chi tiết về không gian của các vật thể mà còn mang lại các đặc trưng ngữ nghĩa, giúp cải thiện khả năng phát hiện và phân loại vật thể. Các lớp này sẽ là đầu vào quan trọng cho các thành phần tiếp theo trong mạng, bao gồm Neck và Head.

### 1.2 QAHARep-FPN network

QAHARep-FPN (Quantization-Aware and Hardware-Aware Feature Pyramid Network) là thành phần Neck của YOLOFM, chịu trách nhiệm hợp nhất các đặc trưng từ nhiều tầng khác nhau của Backbone. Điều này đảm bảo rằng mạng có thể tận dụng thông tin từ các đặc trưng đa tỷ lệ để đạt hiệu suất cao trong việc phát hiện và nhận dạng đối tượng. Điểm đặc biệt của QAHARep-FPN nằm ở việc nó được thiết kế để tối ưu hóa hiệu năng trên các thiết bị tài nguyên hạn chế, như các thiết bị IoT hoặc di động.

Đầu tiên, QAHARep-FPN sử dụng một kỹ thuật có tên **GhostConv**, một phép biến đổi tích chập 1×1. GhostConv giúp giảm số lượng kênh của các lớp đặc trưng mà không làm mất đi thông tin quan trọng. Điều này giúp mạng giảm đáng kể độ phức tạp tính toán, làm cho nó nhẹ hơn và nhanh hơn. Sau đó, các đặc trưng đã được xử lý sẽ trải qua một kỹ thuật có tên **Cross-Transformation**, trong đó thực hiện các phép lấy mẫu tăng (up-sampling) và lấy mẫu giảm (down-sampling). Cross-Transformation giúp các lớp đặc trưng có cùng kích thước, làm cho việc tích hợp giữa các tầng trở nên dễ dàng hơn.

QAHARep-FPN cũng tích hợp module **QARepNeXt**, một biến thể của ResNeXt, để trích xuất đặc trưng. QARepNeXt được thiết kế để tận dụng tối đa các ưu điểm của nhận thức lượng tử hóa (quantization-aware) và nhận thức phần cứng (hardware-aware), từ đó tối ưu hóa hiệu năng và giảm tải tính toán. Kết quả của QAHARep-FPN là ba lớp đặc trưng với kích thước cuối cùng là (20, 20, 1024), (40, 40, 512), và (80, 80, 256). Những lớp này đóng vai trò làm đầu vào cho Head của YOLOFM, cung cấp thông tin đầy đủ về không gian và ngữ nghĩa.

### NADH decoupled head

NADH (New Asymmetric Decoupled Head) là thành phần Head của YOLOFM, được thiết kế để đảm bảo việc xử lý các nhiệm vụ phát hiện đối tượng diễn ra hiệu quả và chính xác. Một trong những cải tiến quan trọng của NADH là khả năng phân tách hai nhiệm vụ chính của Head, bao gồm hồi quy khung giới hạn (bounding box regression) và phân loại (classification). Việc phân tách này không chỉ giúp tăng độ chính xác của dự đoán mà còn cải thiện hiệu quả tính toán.

Cấu trúc của NADH bao gồm hai nhánh chính. Nhánh đầu tiên là **Regression Head**, chịu trách nhiệm dự đoán tọa độ của các khung giới hạn. Để tăng tốc độ hội tụ và cải thiện độ chính xác, Regression Head sử dụng một hàm mất mát tiên tiến có tên **Focal-SIoU Loss**. Nhánh thứ hai là **Classification Head**, chịu trách nhiệm phân loại các đối tượng trong khung giới hạn. Classification Head sử dụng thông tin từ các đặc trưng không gian và ngữ nghĩa được cung cấp bởi Neck để dự đoán chính xác các lớp của đối tượng.

Ngoài ra, NADH tích hợp một logic nén phi đối xứng (asymmetric compression logic), giúp tối ưu hóa việc xử lý các đặc trưng không đồng nhất từ các tầng khác nhau của Neck. Kết quả đầu ra của NADH là ba lớp dự đoán với kích thước lần lượt là (20, 20, 21), (40, 40, 21), và (80, 80, 21), trong đó 21 đại diện cho số lượng lớp (bao gồm các lớp "lửa", "khói" và nền).

### 1.4 Focal-SioU loss

Hàm mất mát **Focal-SIoU Loss** là một cải tiến quan trọng trong YOLOFM, được thiết kế để thay thế hàm CIoU Loss truyền thống. Hàm này được xây dựng nhằm cải thiện tốc độ hội tụ, độ chính xác và khả năng xử lý các vật thể khó nhận dạng, đặc biệt là trong các môi trường phức tạp.

Focal-SIoU Loss kết hợp hai thành phần chính: **Focal Mechanism** và **Smooth Intersection over Union (SIoU)**. Focal Mechanism tăng trọng số cho các khung giới hạn khó dự đoán, giúp mạng tập trung vào các trường hợp phức tạp. Trong khi đó, SIoU là một biến thể cải tiến của IoU, không chỉ xét đến mức độ chồng lấp giữa khung dự đoán và thực tế, mà còn tính đến khoảng cách trung tâm giữa hai khung này. Điều này làm cho SIoU trở nên hiệu quả hơn trong việc tối ưu hóa vị trí và kích thước của khung giới hạn.

Công thức tổng quát của Focal-SIoU Loss được biểu diễn dưới dạng:

Focal-SIoU Loss=Focal Weight×Smooth IoU Loss

Hàm mất mát này cung cấp gradient ổn định hơn trong quá trình huấn luyện, từ đó giúp mô hình hội tụ nhanh hơn và đạt được độ chính xác cao hơn, đặc biệt đối với các vật thể nhỏ hoặc trong các cảnh có độ nhiễu cao.

## 2. So sánh với các mô hình khác

YOLOFM được thiết kế như một sự cải tiến từ YOLOv5n, với nhiều cải tiến nhằm tăng độ chính xác, tốc độ hội tụ và khả năng triển khai trên các thiết bị tài nguyên hạn chế. Dưới đây là sự so sánh giữa YOLOFM và các mô hình khác như YOLOv5, YOLOv8, Faster R-CNN, và các thuật toán phổ biến khác.

### 2.1 So với mô hình gốc YOLOv5n

YOLOFM được xây dựng dựa trên YOLOv5n, nhưng cải tiến đáng kể ở các thành phần như Backbone, Neck, và Head.

|  |  |  |
| --- | --- | --- |
| Tiêu chí | YOLOv5n | YOLOFM |
| Độ chính xác | Phù hợp với các nhiệm vụ đơn giản. | Cải thiện đáng kể (mAP50-95 tăng 7.9%). |
| Kích thước mô hình | Nhẹ, phù hợp với thiết bị di động. | Tăng nhẹ nhưng cân bằng hiệu suất. |
| Tốc độ | Nhanh trên các thiết bị hạn chế. | Tối ưu hóa tốc độ và hiệu suất. |
| Cải tiến | Dựa trên cấu trúc cũ. | FocalNext, QAHARep-FPN, NADH Head. |

YOLOFM sử dụng FocalNext để cải thiện khả năng tổng hợp thông tin, QAHARep-FPN để tăng hiệu suất hợp nhất đặc trưng và NADH Head để cải thiện phân loại và hồi quy. Điều này giúp YOLOFM vượt trội hơn YOLOv5 trong các tác vụ phức tạp.

# III. TRIỂN KHAI HỆ THỐNG

Sơ đồ khối chung của hệ thống được thể hiện như hình sau:

A diagram of a diagram

Description automatically generated with medium confidence

Hình 2. Sơ đồ khối của hệ thống

Hệ thống bao gồm các khối chính sau:

* Khối thu hình ảnh: Sử dụng camera để thu hình ảnh
* Khối xử lý trung tâm: Máy tính nhúng Raspberry Pi dùng để xử lý hình ảnh thu được và điều khiển robot
* Khối cảnh báo: Đưa ra cảnh báo nếu phát hiện được cháy
* Khối chuyển động và chữa cháy: Gồm động cơ để di chuyển và máy bơm để chữa cháy
* Web – Server: Hiển thị hình ảnh thu được, bao gồm cơ chế tương tác điều khiển robot di chuyển và chữa cháy

Việc triển khai các khối và phần cứng tương ứng được trình bày chi tiết ngay phần sau.

## 1. Triển khai Web – Server

Em sử dụng framework flask để xây dựng web – server. Flask là một micro framework web viết bằng Python, được biết đến với tính đơn giản, linh hoạt và khả năng mở rộng dễ dàng. Được phát triển bởi Armin Ronacher và ra mắt vào năm 2010, Flask cung cấp các công cụ cơ bản cần thiết để xây dựng các ứng dụng web mà không đi kèm với quá nhiều tính năng tích hợp sẵn, cho phép lập trình viên tự chọn các phần mở rộng phù hợp. Flask sử dụng Jinja2 cho hệ thống template và Werkzeug cho hệ thống WSGI, giúp dễ dàng quản lý route và tạo các trang web động. Nhờ tính nhẹ nhàng và dễ sử dụng, Flask rất phù hợp cho cả các dự án nhỏ lẫn các API backend phức tạp.

Giao diện chung của Web – Server được thể hiện như sau:

A screenshot of a video game

Description automatically generated

Hình 3. Giao diện chung Web – Server

Giao diện bao gồm 2 phần là hiển thị hình ảnh (bên trái) và tương tác điều khiển robot (bên phải). Chi tiết về từng phần được trình bày ngay sau đây.

### Truyền hình ảnh và phát hiện cháy

Khung video phát hiện cháy được hiển thị bên trái của giao diện Web – Server.

Đồng thời, khung video hiện thị FPS khi truyền video và thực hiện phát hiện cháy.

### 1.2 Tương tác điều khiển robot

Phần giao diện tương tác điều khiển robot bao gồm thanh trượt để chọn giữa 2 mode điều khiển là Munual và Auto và các nút nhấn dùng để điều khiển robot.

Ở chế độ Auto, robot hoạt động theo một kịch bản cho trước. Lúc này, các nút điều khiển robot bị vô hiệu hóa.

Ở chế độ Munual, kích hoạt lại các nút điều khiển robot để người sử dụng điều khiển robot.

A screenshot of a computer

Description automatically generated

Hình 4. Giao diện tương tác điều khiển robot

## 2. Triển khai phần cứng

### 2.1 Sơ đồ mạch triển khai phần cứng

Sơ đồ mạch tiển khai phần cứng được thể hiện như hình sau:

A diagram of a circuit board

Description automatically generated

Hình 5. Sơ đồ mạch triển khai phần cứng

Mạch triển khai phần cứng của đồ án bao gồm các linh kiện sau:

* Máy tính nhúng Raspberry Pi 4, camera module V2.1
* Arduino UNO R3, module điều khiển động cơ L293D
* Module điều khiển động cơ L298N, máy bơm
* USB to TTL CP2102
* Động cơ DC, bánh xe mecanum
* Nguồn, module hạ áp, UPS HAT
* Ngoài ra còn có khung xe robot để đặt các linh kiện lên trên.

Sau khi lắp ráp, em thu được mô hình phần cứng như hình sau:

A machine with yellow wheels and blue wires

Description automatically generated

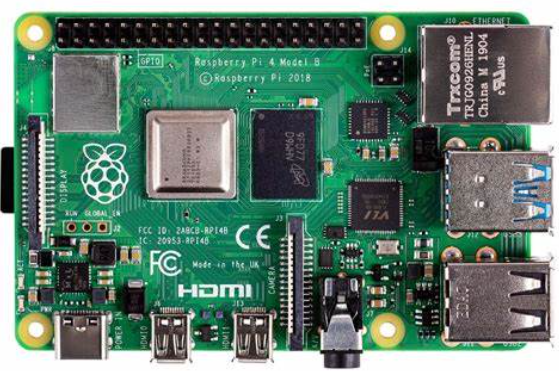
Hình 6. Mô hình phần cứng

### 2.2 Máy tính nhúng Raspberry Pi 4B & Camera

#### 2.2.1 Máy tính nhúng Raspberry Pi 4B

Raspberry Pi 4 là một trong những sản phẩm nổi bật của dòng máy tính nhúng Raspberry Pi, do Quỹ Raspberry Pi phát triển. Với sự cải tiến vượt bậc về cấu hình và khả năng kết nối, Raspberry Pi 4 được xem là một công cụ mạnh mẽ cho các dự án IoT (Internet of Things), tự động hóa, giáo dục và nghiên cứu.

Dưới đây là hình ảnh của máy tính nhúng Raspberry Pi 4:



Hình 7. Raspberry Pi 4

Máy tính nhúng Raspberry Pi 4 có cấu hình chi tiết như sau:

* Bộ vi xử lý (CPU) Broadcom BCM2711, 64-bit ARM Cortex-A73 (ARMv8), Quad-core, 28nm Processor SoC xung nhịp 1.5GHz, với nắp nhôm giúp tản nhiệt tốt hơn.
* Bộ nhớ RAM LPDDR4 4GB SDRAM
* 2 x USB3.0 + 2 x USB2.0
* 40 chân GPIO
* Video/Audio Out:

+ Jack 3.5mm 4 chấu

+ 2 cổng mMicro HDMI, hỗ trợ video 4K@60FPS

+ Hỗ trợ song song 2 màn hình

+ Cổng DSI

* Image/Camera input: Cổng CSI
* Lưu trữ: Thẻ MicroSD
* Cổng Gigabit Ethernet, >900 Mbps
* Hỗ trợ Power-over-Ethernet (PoE) (cần thêm PoE HAT)
* Wifi 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac
* Bluetooth 5.0 hỗ trợ BLE (Bluetooth Low Energy)
* Low-Level Peripherals:

+ 27 x GPIO

+ UART

+ I2C bus

+ SPI bus

+ 3.3V

+ 5V

+ Ground

* Yêu cầu nguồn 5V/3.0A cổng USB-C
* Kích thước: 85mm x 56mm x 17mm

Sơ đồ chân (PinOut) của máy tính nhúng Raspberry Pi 4 như hình sau:

A diagram of a computer

Description automatically generated

Hình 8. Raspberry Pi 4 Pinout

Ngoài ra, có thể xem trực tiếp bằng lệnh trên Terminal như hình sau (Yêu cầu cài thư viện WiringPi trên hệ điều hành Pi OS):

A screenshot of a computer

Description automatically generated

Hình 9. GPIO Pin Raspberry Pi 4 - Terminal

Từ sơ đồ chân trên, ta thấy, khi sử dụng các chân GPIO của Raspberry Pi cần phải chú ý phân biệt chân GPIO physical và BCM.

#### 2.2.2 Camera V2.1 Raspberry Pi

Raspberry Pi Camera Module V2.1 là một phụ kiện máy ảnh chất lượng cao dành cho Raspberry Pi, cho phép người dùng chụp ảnh và quay video với độ phân giải cao.

Camera có các thông số kỹ thuật như sau:

* Cảm biến hình ảnh:

+ Loại cảm biến: Sony IMX219

+ Độ phân giải: 8 megapixel

+ Kích thước pixel: 1.12 µm x 1.12 µm

+ Định dạng quang học: 1/4 inch

* Chất lượng hình ảnh:

+ Độ phân giải ảnh tĩnh: 3280 x 2464 pixels

+ Khả năng quay video: 1080p30, 720p60, và 640x480p90

+ Ống kính: Tiêu cự cố định, tiêu cự 3.04 mm, khẩu độ f/2.0

* Kết nối:

+ Giao diện: CSI (Camera Serial Interface)

+ Cáp kết nối: Đi kèm với cáp phẳng để kết nối với cổng CSI trên Raspberry Pi

* Kích thước và trọng lượng:

+ Kích thước: 23.86mm x 25.00mm x 9.50mm

+ Trọng lượng: Khoảng 3 gram

Bên cạnh các thông số kĩ thuật trên, camera còn có nhiều tính năng nổi bật như:

* Chất lượng hình ảnh cao: Cảm biến Sony IMX219 cung cấp hình ảnh và video với độ phân giải cao, màu sắc chân thực và độ nhạy sáng tốt.
* Tích hợp dễ dàng: Raspberry Pi Camera Module V2.1 được thiết kế để tương thích hoàn toàn với tất cả các model Raspberry Pi, có thể dễ dàng gắn vào thông qua cổng CSI.
* Tính năng đa dạng: Hỗ trợ nhiều chế độ quay video và chụp ảnh, bao gồm slow-motion và time-lapse, phù hợp cho nhiều ứng dụng khác nhau từ dự án học tập đến các ứng dụng thương mại.
* Khả năng mở rộng: Có thể kết hợp với các phụ kiện khác như đèn LED, case bảo vệ, hoặc sử dụng trong các dự án robot, an ninh giám sát, và các ứng dụng IoT.

A raspberry pi camera box

Description automatically generated

Hình 10. Module camera V2 cho Raspberry Pi

### 2.3 Vi điều khiển Arduino UNO R3

Do trong phạm vi đồ án, em cần sử dụng số lượng chân GPIO khá nhiều, em sử dụng thêm vi điều khiển Arduino UNO R3 để có thể đáp ứng đủ cho số lượng ngoại vi.

A close-up of a circuit board

Description automatically generated

Hình 11. Vi điều khiển Arduino UNO R3

Arduino UNO R3 là một trong những vi điều khiển phổ biến nhất trong dòng sản phẩm Arduino, được sử dụng rộng rãi trong các dự án điện tử, tự động hóa, và học tập lập trình nhúng. Được thiết kế với mục tiêu dễ sử dụng và mở rộng, Arduino UNO R3 là lựa chọn lý tưởng cho cả người mới bắt đầu và những người có kinh nghiệm.

Arduino UNO R3 được trang bị nhiều thành phần phần cứng để hỗ trợ lập trình và phát triển dự án:

* Vi điều khiển: ATmega328P của hãng Atmel, có 32 KB bộ nhớ flash, 2 KB SRAM và 1 KB EEPROM.
* Tần số hoạt động: 16 MHz.
* Nguồn điện: Có thể cấp nguồn qua cổng USB (5V) hoặc qua jack nguồn (7-12V).
* Cổng I/O: 14 chân digital I/O (trong đó 6 chân hỗ trợ PWM) và 6 chân analog input.
* Giao tiếp: Cổng USB type-B để kết nối với máy tính và nạp chương trình, một cổng UART (TX/RX), và giao tiếp I2C, SPI.
* Bộ điều khiển USB: ATmega16U2 dùng để chuyển đổi giữa giao tiếp USB và UART.
* Nút reset: Để khởi động lại vi điều khiển.

A circuit board with many different colored wires

Description automatically generated

Hình 12. Arduino UNO R3 Pinout

Vi điều khiển Arduino UNO R3 có nhiều ưu điểm như:

* Dễ sử dụng: Thiết kế đơn giản, có tài liệu và cộng đồng hỗ trợ phong phú, dễ dàng bắt đầu với lập trình nhúng.
* Tính mở rộng: Hỗ trợ nhiều shield (bảng mở rộng) để thêm chức năng như kết nối Wi-Fi, Ethernet, cảm biến, màn hình LCD, và nhiều hơn nữa.
* Phần mềm mã nguồn mở: Arduino IDE (Integrated Development Environment) là mã nguồn mở, hỗ trợ nhiều nền tảng như Windows, macOS, và Linux.

- Cộng đồng lớn: Có một cộng đồng người dùng lớn và nhiệt tình, cung cấp nhiều tài nguyên, thư viện và hỗ trợ kỹ thuật.

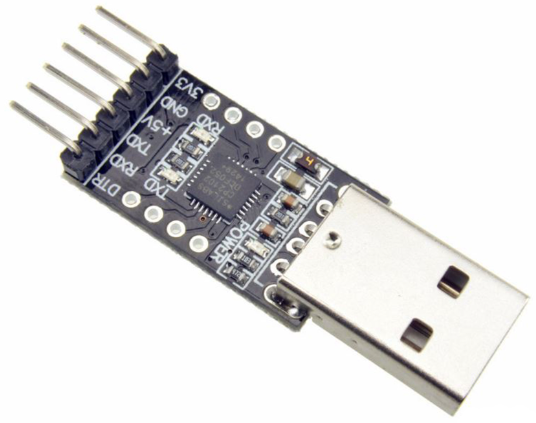
Tuy nhiên, Arduino UNO R3 cũng có một số nhược điểm sau:

* Hiệu suất giới hạn: Với tốc độ xử lý và bộ nhớ hạn chế, Arduino UNO R3 không phù hợp cho các ứng dụng yêu cầu hiệu suất cao.
* Khả năng lưu trữ hạn chế: Bộ nhớ flash chỉ 32 KB, không đủ cho các ứng dụng phức tạp hoặc chứa nhiều dữ liệu.

- Không tích hợp sẵn kết nối không dây: Phải sử dụng thêm các shield hoặc module để có kết nối Wi-Fi hoặc Bluetooth.

### 2.4 USB to TTL CP2102

Để giao tiếp giữa Raspberry Pi 4 và Arduino UNO R3, em sử dụng chuẩn giao tiếp UART (Universal Asynchronous Receiver/Transmitter) qua USB to TTL CP2102.



Hình 13. USB to TTL CP2102

CP2102 là một chip chuyển đổi giao tiếp USB sang UART (Universal Asynchronous Receiver/Transmitter) do Silicon Labs phát triển. Nó thường được sử dụng để kết nối các thiết bị có giao diện UART (như vi điều khiển hoặc các thiết bị nhúng khác) với máy tính hoặc các thiết bị có cổng USB. CP2102 được ưa chuộng vì tính đơn giản, dễ sử dụng và độ tin cậy cao.

CP2102 có những đặc điểm và tính năng chính sau:

* Tương thích USB 2.0: CP2102 hỗ trợ chuẩn USB 2.0 full-speed (12 Mbps).
* Tích hợp đầy đủ: Bao gồm bộ điều khiển USB, bộ điều khiển UART, bộ đệm dữ liệu, và các mạch điều khiển dòng dữ liệu.
* Tốc độ truyền dữ liệu: Hỗ trợ tốc độ truyền dữ liệu lên đến 1 Mbps cho UART.
* Dải điện áp rộng: Hoạt động ở các mức điện áp từ 1.8V đến 3.3V và có thể chịu được đầu vào lên đến 5V, cho phép kết nối với nhiều loại thiết bị.
* Driver hỗ trợ đa nền tảng: Cung cấp driver cho các hệ điều hành phổ biến như Windows, macOS, và Linux.

- Giao diện nhỏ gọn: Thường được đóng gói trong các module nhỏ gọn, dễ tích hợp vào các dự án DIY và các hệ thống nhúng.

### 2.5 Các thành phần khác

#### 2.5.1 Module điều khiển động cơ L293D & L298N

A blue circuit board with black and blue components

Description automatically generated

Hình 14. Module điều khiển động cơ L293D

L298N là module điều khiển động cơ trong các xe DC và động cơ bước, rất dễ sử dụng với các bộ vi điều khiển và cũng tương đối rẻ.

Module có một IC điều khiển động cơ L298N, một bộ điều chỉnh điện áp 5V 78M05, 5V Enable Jumper, đèn LED nguồn, tản nhiệt, điện trở và tụ điện, tất cả được tích hợp trong một mạch . Module L298N có thể điều khiển tối đa 4 động cơ DC hoặc 2 động cơ DC với khả năng điều khiển hướng và tốc độ.

Thông số kĩ thuật của module điều khiển động cơ L298N như sau:

* Module điều khiển: L298N
* Chip điều khiển: Cặp H-bridge L298N
* Công suất tối đa: 25W
* Điện áp tối đa cấp cho động cơ: 46V
* Dòng tối đa cấp cho động cơ: 2A
* Điện áp hoạt động của IC: 5-35V
* Dòng điện hoạt động IC: 2A

ENA điều khiển tốc độ của động cơ A và ENB điều khiển tốc độ của động cơ B. Nếu cả hai chân đều ở trạng thái logic CAO (5V), thì cả hai động cơ đều BẬT và quay ở tốc độ tối đa. Nếu cả hai chân đều ở trạng thái logic THẤP (tiếp đất), thì cả hai động cơ đều TẮT. Thông qua chức năng PWM, chúng ta cũng có thể kiểm soát tốc độ của động cơ.

|  |  |
| --- | --- |
| ENA Pin State | Motor Action |
| 1 (HIGH) | ON |
| 0 (LOW) | OFF |

Các chân điều khiển hướng là bốn chân đầu vào (IN1, IN2, IN3, IN4) với hai chân IN1 và IN2 là đầu vào của động cơ A, hai chân IN3 và IN4 là đầu vào của động cơ B.

#### 2.5.2 Động cơ DC Servo JGB37-520 & bánh xe mecanum

Trong phạm vi đồ án này, em sử dụng động cơ DC Servo JGB37-520 cho việc di chuyển khung xe robot.

A small metal electric motor with wires

Description automatically generated

Hình 15. Động cơ DC Servo JGB37-520

Động Cơ DC Servo JGB37-520 DC Geared Motor được tích hợp thêm Encoder hai kênh AB giúp đọc và điều kiển chính xác vị trí, chiều quay của động cơ trong các ứng dụng cần độ có chính xác cao: điều khiển PID, Robot tự hành,....

A diagram of a cable

Description automatically generated

Hình 16. Động cơ DC Servo JGB37-520

Động Cơ DC Servo JGB37-520 DC Geared Motor có cấu tạo bằng kim loại cho độ bền và độ ổn định cao, được sử dụng trong các mô hình robot, xe, thuyền,..., hộp giảm tốc của động cơ có nhiều tỉ số truyền giúp bạn dễ dàng lựa chọn giữa lực kéo và tốc độ (lực kéo càng lớn thì tốc độ càng chậm và ngược lại), động cơ sử dụng nguyên liệu chất lượng cao (lõi dây đồng nguyên chất, lá thép 407, nam châm từ tính mạnh,...) cho sức mạnh và độ bền vượt trội hơn các loại giá rẻ trên thị trường hiện nay (sử dụng lõi dây nhôm, nam châm từ tính yếu).

Thông số kỹ thuật chung của động cơ như sau:

* Điện áp sử dụng: 12VDC
* Đường kính động cơ: 37mm
* Đường kính trục động cơ: 6mmm
* Encoder: Cảm biến từ trường Hall, có 2 kênh AB lệch nhau giúp xác định chiều quay và vận tốc của động cơ, đĩa Encoder trả ra 11 xung/1 kênh/ 1 vòng (nếu đo tín hiệu đồng thời của cả hai kênh sẽ thu được tổng 22 xung / 1 vòng quay của Encoder).
* Điện áp cấp cho Encoder hoạt động: 3.3~5VDC

Trong đồ án này, em sử dụng loại 12VDC 333RPM với :

* Tỉ số truyền 30:1 (động cơ quay 30 vòng trục chính hộp giảm tốc quay 1 vòng).
* Dòng không tải: 120mA
* Dòng chịu đựng tối đa khi có tải: 1A
* Tốc độ không tải: 333RPM (333 vòng 1 phút)
* Tốc độ chịu đựng tối đa khi có tải: 250RPM (250 vòng 1 phút)
* Lực kéo Moment định mức: 3.5KG.CM
* Lực léo Moment tối đa: 5KG.CM
* Chiều dài hộp số L: 22mm
* Số xung Encoder mỗi kênh trên 1 vòng quay trục chính: 11 x 30 = 330 xung.

Sơ đồ chân của động cơ bao gồm các chân:

* M1: Dây cấp nguồn cho động cơ.
* GND: Dây cấp nguồn cho Encoder, 0VDC.
* C1/A: Kênh trả xung A
* C2/B: Kênh trả xung B
* VCC: Dây cấp nguồn cho Encoder 3.3~5VDC
* M2: Dây cấp nguồn cho động cơ

Để kết hợp với động cơ DC Servo JGB37-520, em lựa chọn bánh xe mecanum.Bánh xe Mecanum là một loại bánh xe đặc biệt được thiết kế để cho phép phương tiện di chuyển theo nhiều hướng khác nhau mà không cần thay đổi hướng của bánh xe. Điều này làm cho bánh xe Mecanum trở thành lựa chọn lý tưởng cho các hệ thống robot di động, xe tự hành, và các ứng dụng khác cần khả năng di chuyển linh hoạt trong không gian hẹp.

A yellow and black wheel

Description automatically generated

Hình 17. Bánh xe mecanum

Cấu tạo của bánh xe Mecanum bao gồm:

* Khung chính: Hình tròn, giống như bánh xe thông thường.
* Các con lăn: Gắn trên viền của bánh xe, thường là từ 8 đến 12 con lăn. Các con lăn này được bố trí chéo so với trục quay của bánh xe, thường là góc 45 độ.

Nguyên lý hoạt động của bánh xe mecanum:

* Khi bánh xe Mecanum quay, lực tác động từ các con lăn tạo ra các thành phần lực theo các hướng khác nhau.
* Bằng cách điều chỉnh tốc độ và hướng quay của từng bánh xe, có thể điều khiển phương tiện di chuyển theo mọi hướng: tiến, lùi, sang trái, sang phải, và xoay tại chỗ.

Bánh xe mecanum có những ưu điểm sau:

* Di chuyển linh hoạt: Có thể di chuyển theo mọi hướng mà không cần thay đổi hướng của bánh xe hoặc thân xe.
* Khả năng xoay tại chỗ: Có thể xoay quanh trục của chính nó, rất hữu ích trong các không gian hẹp.
* Ứng dụng rộng rãi: Thường được sử dụng trong các hệ thống robot, xe tự hành trong nhà kho, xe cứu hộ, và các ứng dụng công nghiệp khác.

Bên cạnh đó, bánh xe mecanum cũng có một số nhược điểm:

* Phức tạp trong điều khiển: Yêu cầu hệ thống điều khiển phức tạp để quản lý sự phối hợp giữa các bánh xe.
* Hiệu suất trên địa hình không bằng phẳng: Hiệu suất giảm khi di chuyển trên các bề mặt không đều hoặc trơn trượt.

Tuy nhiên, với phạm vi áp dụng mong muốn của đồ án này là trong những không gian có mặt sàn tương đối phẳng, việc lựa chọn bánh xe mecanum là việc hợp lý.

#### 2.5.3 Nguồn (Pin Lithium và module hạ áp)

Trong đồ án này, em sử dụng pin sạc Lithium 18650. Với điện áp mỗi viên pin trong khoảng 3,7-4 V, pin sạc 18650 phù hợp với nhiều loại ứng dụng điện tử khác nhau.

A close-up of a battery

Description automatically generated

Hình 18. Pin sạc Lithium 18650

Đồng thời, để có thể lấy ra được điện áp 3,3/5V , em sử dụng module hạ áp LM2596.

A close-up of a blue circuit board

Description automatically generated

Hình 19. Module hạ áp LM2596

Module có điện áp vào 3 - 40V DC (Khuyến cáo sử dụng điện áp đầu vào < 30V DC), điện áp ra 1,5 – 35V DC, dòng điện ra max 3A, điện áp đầu ra có thể điều chỉnh bằng biến trở. Với các thông số trên, module có thể kết hợp với pin 18650 để có thể lấy được điện áp và dòng điện đủ để cho mạch hoạt động.

Bên cạnh đó, do máy tính nhúng Raspberry Pi được đặt trên robot, nên cần sử dụng nguồn rời cho nó. Vì vậy, em sử dụng module nguồn UPS HAT. UPS HAT (Uninterruptible Power Supply Hardware Attached on Top) là một phụ kiện cung cấp nguồn dự phòng cho Raspberry Pi, giúp bảo vệ thiết bị khỏi mất điện đột ngột và đảm bảo rằng các dữ liệu quan trọng không bị mất mát.

Module UPS HAT có một số đặc điểm nổi bật sau:

* Có đầu nối mở rộng 40 chân GPIO tiêu chuẩn của Raspberry Pi. Bạn có thể dùng UPS cùng các thiết bị khác sử dụng chân GPIO
* Giao tiếp bus I2C, giám sát điện áp, dòng điện, nguồn và dung lượng còn lại của pin trong thời gian thực
* Nhiều mạch bảo vệ pin: bảo vệ quá tải / quá tải, bảo vệ quá dòng, bảo vệ ngắn mạch và bảo vệ ngược, cùng với tính năng sạc cân bằng, an toàn và ổn định hơn
* Bộ điều chỉnh 5V tích hợp, dòng đầu ra liên tục lên đến 2,5A
* Đầu ra USB 5V, thuận tiện để cấp nguồn cho các bo mạch khác
* Có cảnh báo pin, dễ dàng kiểm tra xem pin đã được kết nối đúng chưa

Module UPS HAT có các thông số kỹ thuật như sau:

* Điện áp đầu ra 5V
* Sạc 8.4V – 2A
* Control Bus I2C
* Kích thước 56x85mm
* Hỗ trợ pin Pin Lithium 18650
* Kích thước lỗ gắn với bo mạch 0.3mm

A close-up of a battery

Description automatically generated

Hình 20. Nguồn UPS HAT Raspberry Pi

Do máy tính nhúng Raspberry có sông suất tiêu thụ lớn hơn vi điều khiển thông thường, nên em sử dụng 2 viên pin Lithium dung lượng cao (12580mWh) để đảm bảo cung cấp đủ cho robot hoạt động trong thời gian dài.

#### 2.5.4 Máy bơm mini MB370 & ống nước

Để thực hiện việc điều khiển chữa cháy, em sử dụng máy bơm mini MB370, kết nói với bình nước đặt trên robot.

Máy bơm có một số thông số kĩ thuật sau:

* Điện áp: 3 - 6V DC
* Dòng: 0.3A
* Kích thước: 60x26mm
* Đầu hút nước vào: 5mm
* Đầu đẩy nước ra: 5mm
* Nhiệt độ nước: 0 độ C đến 60 độ C

A small electric motor with a white plastic cover

Description automatically generated

Hình 21. Máy bơm mini MB370

Máy bơm MB370 hoạt động dựa theo nguyên tắc áp lực, nước sẽ được chảy từ nơi có sức ép cao đến nơi có sức ép thấp hơn. Máy có kích thước nhỏ gọn, dễ dàng di chuyển và lắp đặt theo nhu cầu của người sử dụng. Chính vì vậy, em lựa chọn nó để dễ dàng đặt trên robot.

# IV. KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ

## Model YOLOFM phát hiện cháy

- Setting:

Mô hình được training với các thông sau:

Dataset:  
Phần cứng:

Thông số mô hình

Thời gian training:

Với các giá trị trên, kết quả thu được như sau:

A graph of a function

Description automatically generated with medium confidence

Hình 22. Kết quả training

**Dòng 1: Training Loss (Các biểu đồ đầu ra trong quá trình huấn luyện)**

1. **train/box\_loss**:

Đây là tổn thất (loss) liên quan đến việc dự đoán vị trí của các bounding boxes (hộp giới hạn đối tượng).

**Nhận xét**:

- Đường cong giảm đều từ khoảng 0.05 xuống gần 0.02 qua 200 epoch, điều này cho thấy mô hình học cách định vị bounding box ngày càng chính xác hơn.

- Giai đoạn đầu, tổn thất giảm nhanh (epoch 0-50) vì mô hình học được các đặc trưng cơ bản. Sau đó, tổn thất giảm chậm hơn (epoch 50-200) do mô hình dần đạt mức hội tụ.

**Giải thích**:

- Loss giảm nhanh ở giai đoạn đầu là bình thường, do các thông số mô hình được tối ưu ban đầu.

- Sau giai đoạn 50 epoch, mô hình cần tinh chỉnh các chi tiết nhỏ, nên tốc độ giảm tổn thất chậm hơn.

1. **train/obj\_loss**:

Tổn thất liên quan đến việc xác định liệu một vùng có chứa đối tượng hay không (objectness).

**Nhận xét**:

- Giá trị giảm từ 0.06 xuống gần 0.02 sau 200 epoch. Điều này cho thấy mô hình ngày càng tự tin hơn khi phân biệt giữa các vùng chứa đối tượng và vùng không chứa đối tượng.

**Giải thích**:

- Ban đầu, mô hình có nhiều sai sót trong việc dự đoán objectness, dẫn đến tổn thất cao.

- Tổn thất giảm cho thấy mô hình học được cách xác định chính xác các vùng chứa đối tượng, giảm false positives (dự đoán sai là có đối tượng).

1. **train/cls\_loss**:

Tổn thất liên quan đến việc phân loại đối tượng thành các lớp cụ thể.

**Nhận xét**:

- Giá trị giảm từ khoảng 0.02 xuống gần 0 sau 200 epoch. Điều này chỉ ra rằng mô hình đang cải thiện khả năng phân biệt các loại đối tượng (ví dụ: chó, mèo, xe).

**Giải thích**:

- Ban đầu, mô hình chưa học được đặc trưng phân loại, dẫn đến loss cao.

- Sau nhiều epoch, mô hình trở nên chính xác hơn trong việc dự đoán đúng nhãn của từng đối tượng, dẫn đến loss giảm mạnh.

**Dòng 2: Validation Loss (Tổn thất trên tập dữ liệu kiểm định)**

1. **val/box\_loss**:

Tổn thất trên tập validation liên quan đến dự đoán vị trí bounding box.

**Nhận xét**:

- Giá trị giảm từ khoảng 0.045 xuống gần 0.02. Đường cong này tương tự như train/box\_loss, nhưng không giảm sâu bằng.

**Giải thích**:

- Sự giảm này chỉ ra rằng mô hình không chỉ học tốt trên tập huấn luyện mà còn tổng quát hóa tốt trên dữ liệu validation.

- Chênh lệch nhỏ giữa train/box\_loss và val/box\_loss cho thấy mô hình không bị overfitting (quá khớp).

1. **val/obj\_loss**:

Tổn thất liên quan đến phát hiện objectness trên tập validation.

**Nhận xét**:

- Giá trị giảm từ 0.022 xuống khoảng 0.01, phản ánh rằng mô hình dần phát hiện tốt hơn các vùng chứa đối tượng trong tập validation.

**Giải thích**:

- Kết quả tốt trên tập validation cho thấy mô hình có khả năng tổng quát hóa và không bị phụ thuộc quá nhiều vào tập huấn luyện.

1. **val/cls\_loss**:

Tổn thất liên quan đến phân loại các đối tượng trên tập validation.

**Nhận xét**:

- Giá trị giảm từ 0.012 xuống gần 0 sau 200 epoch.

**Giải thích**:

- Sự giảm này cho thấy mô hình phân loại các đối tượng trên tập validation chính xác hơn, không chỉ giới hạn ở tập huấn luyện.

- Đường cong tương đồng với train/cls\_loss, cho thấy sự ổn định trong việc huấn luyện.

**Dòng 3: Metrics (Hiệu suất của mô hình)**

1. **metrics/precision**:

Độ chính xác của mô hình (tỷ lệ dự đoán đúng trên tổng dự đoán).

**Nhận xét**:

- Precision tăng từ khoảng 0.4 lên gần 0.95 sau 200 epoch.

- Đường cong hội tụ nhanh ở giai đoạn đầu, sau đó ổn định ở mức cao.

**Giải thích**:

- Precision tăng cho thấy mô hình ngày càng giảm các false positives (dự đoán sai là có đối tượng).

- Giá trị gần 1 phản ánh mô hình rất ít khi dự đoán nhầm đối tượng.

1. **metrics/recall**:

Độ bao phủ của mô hình (tỷ lệ đối tượng thực sự được phát hiện trên tổng đối tượng thực tế).

**Nhận xét**:

- Recall tăng từ khoảng 0.2 lên gần 0.9, nhưng thấp hơn precision.

**Giải thích**:

- Recall tăng cho thấy mô hình giảm false negatives (bỏ sót đối tượng).

- Giá trị thấp hơn precision có thể do mô hình vẫn bỏ sót một số đối tượng nhỏ hoặc khó phát hiện.

1. **metrics/mAP\_0.5**:

Mean Average Precision với IoU = 0.5 (một ngưỡng đơn giản).

**Nhận xét**:

- mAP tăng từ khoảng 0.2 lên gần 1.0, đạt hiệu suất rất cao sau 200 epoch.

**Giải thích**:

- Kết quả này cho thấy mô hình dự đoán tốt và bounding box của nó đủ chính xác khi IoU = 0.5.

1. **metrics/mAP\_0.5:0.95**:

Mean Average Precision với IoU từ 0.5 đến 0.95 (ngưỡng nghiêm ngặt hơn).

**Nhận xét**:

- mAP tăng từ khoảng 0.1 lên gần 0.7, thấp hơn mAP@0.5, nhưng vẫn đạt hiệu suất tốt.

**Giải thích**:

- Giá trị thấp hơn là do ngưỡng IoU cao đòi hỏi bounding box phải rất chính xác.

- Giá trị tăng ổn định cho thấy mô hình học tốt và có khả năng phát hiện chính xác trên các mức độ IoU khác nhau.

**Tổng quan:**

* **Loss giảm**: Cho thấy mô hình học ngày càng tốt hơn và hội tụ dần.
* **Precision và Recall tăng**: Mô hình giảm thiểu lỗi false positives và false negatives.
* **mAP cao**: Mô hình đạt hiệu suất tốt cả trên các ngưỡng dễ (IoU=0.5) và khó (IoU=0.95).
* **Ổn định**: Không có dao động lớn giữa tập train và validation, cho thấy mô hình được huấn luyện ổn định, không bị overfitting.

A graph of a curve

Description automatically generated with medium confidence

Hình 23. Precision-Confidence Curve

A graph of a curve

Description automatically generated

Hình 24. Recall-Confidence Curve

A graph of a graph

Description automatically generated with medium confidence

Hình 25. Precision-Recall Curve

A graph of a curve

Description automatically generated

Hình 26. F1-Confidence Curve

A blue squares with white text

Description automatically generated

Hình 27. Confusion Matrix

- Kết quả test thực tế

## 2. Phần điều khiển chuyển động và chữa cháy

### 2.1 Kịch bản di chuyển tự động

Trong phạm vi đồ án, robot được thiết kế để

# PHỤ LỤC

Trong phạm vi đồ án này, em có sử dụng các nguồn tài liệu sau:

* [(PDF) YOLOFM: an improved fire and smoke object detection algorithm based on YOLOv5n (researchgate.net)](https://www.researchgate.net/publication/378467729_YOLOFM_an_improved_fire_and_smoke_object_detection_algorithm_based_on_YOLOv5n)
* [Home - Ultralytics YOLO Docs](https://docs.ultralytics.com/)