

Thiết Bị Phát Hiện Và Xử Lý Đám Cháy Sử Dụng Tâm Nhìn Máy Tính

Đỗ Văn Chí*, Nguyễn Phi Hùng¹, Bùi Quang Huy², Lê Văn Thanh Duy³

Giảng viên hướng dẫn: TS. Chu Bá Thành

Các thành viên thuộc về Trường đại học Sư phạm Kỹ thuật Hưng Yên, khoa công nghệ thông tin.

Tóm tắt: Nguy cơ từ các vụ cháy không chỉ gây ra nguy hiểm cho sức khỏe mà còn có thể tạo ra những tổn thất nặng nề. Để đánh giá và đối phó với sự nguy hiểm từ cháy, việc sử dụng hình ảnh trong việc phát hiện và xử lý đám cháy đã được xem xét như một công cụ quan trọng.

Nghiên cứu này tập trung vào việc áp dụng hình ảnh để phát hiện đám cháy. Chúng tôi đã sử dụng kiến trúc mạng CNNs để xây dựng mô hình để phân loại các hình ảnh có cháy và hình ảnh không có cháy. Kết quả nghiên cứu đã chỉ ra rằng mô hình sau khi được điều chỉnh và huấn luyện trên một tập dữ liệu lớn bao gồm hình ảnh các đám cháy từ nhiều trường hợp nguy hiểm từ cháy, có khả năng dự đoán một cách chính xác với độ chính xác cao và độ nhạy cao. Phương pháp này có thể giúp cho thiết bị của chúng tôi phát hiện đám cháy đúng lúc và kịp thời để phát thông báo vào xử lý đám cháy bằng hệ thống vòi nước tự động.

Từ khóa: Nguy hiểm từ cháy; CNNs.

1. Giới thiệu

Đám cháy không chỉ gây ra sự thiệt hại về tài sản mà còn tiềm ẩn nguy cơ lớn đối với tính mạng và sức khỏe của mọi người. Từ việc phá hủy tài sản đến việc gây ra thương tích và nguy hiểm đến tính mạng, hậu quả của một vụ cháy có thể lan rộng và tàn phá đến mức không thể lường trước. Cháy có thể lan nhanh và lan rộng, phá hủy mọi thứ trên đường đi của nó. Nguy cơ bị mắc kẹt, ngạt khói và chịu các vết thương từ cháy là rất cao. Những vết thương có thể kéo dài và ảnh hưởng nặng nề đến sức khỏe cả về thể chất và tinh thần. Không chỉ đối với con người, mà còn đối với môi trường xung quanh, đám cháy có thể gây ra các tác động tiêu cực lớn. Khói và khí độc hại từ cháy có thể gây hậu quả nghiêm trọng đến sức khỏe của các cộng đồng xung quanh, đặc biệt là đối với những người có vấn đề về hô hấp hoặc sức khỏe yếu.

Ngoài ra, đám cháy còn có khả năng lan rộng đến mức đe dọa không chỉ một ngôi làng, một thành phố mà còn có thể là cả các khu vực lớn. Hậu quả của việc di cư, mất đi nhà cửa, và sự mất mát về kinh tế có thể kéo dài trong thời gian dài sau vụ cháy ban đầu.

Do đó, việc phòng ngừa và ứng phó với đám cháy là rất quan trọng. Tăng cường các biện pháp an toàn, chuẩn bị cho các trường hợp khẩn cấp, và hỗ trợ cộng đồng trong việc học cách

đối phó với rủi ro của đám cháy là cần thiết để giảm thiểu những tác động nguy hiểm và tàn phá từ những sự cố này.

Chính vì những thiệt hại do cháy gây ra, trong năm 2023 có những vụ cháy gây thiệt hại lên tới hàng chục sinh mạng. Phá hủy tài sản lên đến hàng tỷ đồng, gây ra nhiều mất mát đau thương cho mọi người trong xã hội. Chúng tôi là những sinh viên Trường đại học sư phạm kỹ thuật Hưng Yên, khoa công nghệ thông tin muốn phát triển một thiết bị có thể thông báo và ngăn chặn các đám cháy giúp giảm đi đáng kể thiệt hại về người và của.

Hiện nay trên thị trường có nhiều thiết bị phát hiện đám cháy như là các cảm biến khói, hay hệ thống phun nước. Nhưng độ chính xác của chúng gần như còn rất kém, chỉ có thể thông báo không có chức năng trực tiếp can thiệp vào đám cháy. Thiết bị của chúng tôi sẽ trực tiếp can thiệp vào đám cháy khi nó phát hiện cháy, nó sẽ tự động thông báo cháy và điều hướng cho vòi xịt áp lực cao xịt thẳng vào đám cháy. Việc ngăn chặn đám cháy đúng lúc rất quan trọng vậy nên hệ thống của chúng tôi sẽ rất hiệu quả bởi vì nó sẽ ngăn chặn đám cháy khi từ khi khởi tạo.

2. Các nghiên cứu liên quan

2.1 Phát hiện cháy bằng phương pháp xử lý ảnh truyền thống

Trong phát hiện cháy truyền thống, nhiều nghiên cứu đã liên tục tập trung vào việc tìm ra các đặc điểm nổi bật của hình ảnh cháy. Chen [1] đã phân tích sự thay đổi của lửa bằng cách sử dụng mô hình màu RGB và HSI dựa trên sự khác biệt giữa các khung hình liên tiếp và đề xuất cách tiếp cận dựa trên quy tắc để quyết định lửa. Celik và Demirel [2] đã đề xuất cách phân loại pixel ngọn lửa dựa trên quy tắc chung bằng cách sử dụng mô hình màu YCbCr để tách các thành phần sắc độ khỏi thành phần độ chói. Ngoài ra, Wang [3] đã trích xuất vùng cháy tiềm năng trong ảnh bằng mô hình màu HSI và tính toán độ phân tán của màu ngọn lửa để xác định vùng cháy. Tuy nhiên, các phương pháp phát hiện cháy dựa trên màu sắc thường dễ bị ảnh hưởng bởi nhiều yếu tố môi trường như ánh sáng và bóng tối.

Borges và Izquierdo [4] đã áp dụng bộ phân loại Bayes để phát hiện các đám cháy dựa trên các đặc điểm bổ sung như diện tích, bề mặt và ranh giới của khu vực cháy để tô màu. Mueller [5] đề xuất phương pháp phát hiện cháy dựa trên mạng nơ-ron sử dụng luồng quang học cho khu vực cháy. Trong phương pháp này, hai mô hình dòng quang được kết hợp để phân biệt giữa lửa và các vật thể chuyển động động. Ngoài ra, Foggia [6] còn đề xuất một hệ thống đa chuyên gia kết hợp các kết quả phân tích về màu sắc, hình dạng và đặc điểm chuyển động của đám cháy. Mặc dù chưa đầy đủ nhưng các đặc điểm bổ sung cho màu sắc, bao gồm kết cấu, hình dạng và luồng quang học, có thể làm giảm khả năng phát hiện sai.

Haar Cascade[10] là một kỹ thuật dựa trên việc sử dụng phương pháp đặc trưng Haar, đó là một loại đặc trưng hình ảnh được sử dụng để nhận diện đối tượng. Các bộ lọc Haar được áp dụng trên các cửa sổ trượt qua hình ảnh để xác định vị trí của các đặc trưng quan trọng

trong ảnh. Các bộ lọc này có thể nhận diện các khuôn mẫu cụ thể, như khuôn mặt, mắt, hoặc đối tượng khác.

Mặc dù Haar Cascade là một phương pháp linh hoạt và có thể được sử dụng trong nhiều ứng dụng nhận diện đối tượng, nhưng nó thường không có khả năng học được như deep learning. Trong deep learning, mạng neural sâu được huấn luyện trên một lượng lớn dữ liệu để tự động học các đặc trưng và mối quan hệ phức tạp trong dữ liệu. Các mô hình deep learning thường có khả năng tự tối ưu hóa và chứa nhiều tầng ẩn để biểu diễn các mức độ trừu tượng khác nhau.

Tóm lại, những cách tiếp cận này đòi hỏi kiến thức về miền về các đám cháy trong các hình ảnh được chụp cần thiết để khám phá các tính năng thủ công và không thể phản ánh tốt thông tin liên quan đến không gian và thời gian trong môi trường cháy. Ngoài ra, hầu hết các phương pháp sử dụng phương pháp thông thường đều chỉ sử dụng ảnh tĩnh hoặc các cặp khung hình liên tiếp để phát hiện cháy. Vì vậy, họ chỉ xem xét đặc tính động học ngắn hạn của lửa, trong khi đám cháy có đặc tính động học dài hạn hơn.

2.2 Phát hiện cháy bằng tầm nhìn máy tính sử dụng Deep Learning

Ngày nay có nhiều mô hình deep learning ra đời phục vụ việc phân loại hình ảnh và tất nhiên cháy cũng là một loại hình ảnh có thể phân loại. Mô hình Yolov5[7] là một phương pháp mạnh mẽ sử dụng DL để giải quyết bài toán này. Trong bài báo nghiên cứu về việc phát hiện cháy sử dụng mô hình Yolov5 của tác giả *Shangjie Ge Zhang,1Fengxi Zhang,2Yuyang Ding,3and Yu Li* họ đã chỉ ra rằng việc sử dụng Yolov5 để phát hiện đám cháy có hiệu năng rất cao, họ chứng minh rằng sử dụng *Swin-YOLOv5* cao hơn 0,7% so với thuật toán gốc.

Ngoài ra có các kiến trúc deep learning đã được xây dựng sẵn giúp chúng ta có thể phát triển các mô hình DL một cách đơn giản và nhanh chóng như AlexNet, GoogleNet, VGG...

Tôi đã thống kê một số các mô hình và kết quả ở bảng dưới đây.

Models	Dataset	Author	Accurace(%)
Yolov5	> 1500 Images	spacewalk01[8]	98%
AlexNet	1370 Images	Chi	75%
ResNet-18	1370 Images	Chi	98%

Hình 4.1.1: So sánh giữa các mô hình

Với mô hình Yolov5 được đào tạo bởi bộ dữ liệu >1500 ảnh có thể thấy rằng độ chính xác rất cao khoảng 98%. Mô hình được xây dựng bởi kiến trúc AlexNet với bộ dữ liệu 1370 ảnh cho độ chính xác 75%, mô hình ResNet-18 cũng với bộ dữ liệu tương tự của AlexNet cho độ chính xác 98%.

3. Phương pháp đề xuất

3.1 Kiến trúc mạng học sâu ResNet-18

Chúng tôi đề xuất sử dụng mô hình được xây dựng dựa trên kiến trúc ResNet-18 kết hợp với phương pháp tranfer learning. Học chuyển tiếp(transfer learning)[9] là phương pháp học máy cho phép sử dụng lại các tham số của một mạng nơ ron đã huấn luyện trong một bài toán để giải quyết một bài toán mới. Điều này cũng tương tự như việc sử dụng các kiến thức về toán học để làm nền tảng cho việc học vật lý. Trên thực tế, phương pháp học chuyển tiếp khá hiệu quả trong nhiều bài toán nên trong bài toàn này chúng tôi đề xuất sử dụng cách này để xây dựng mô hình phân loại ảnh cháy và không cháy.

Do hiệu năng của các thiết bị IoT thường không được mạnh mẽ nên không thể sử dụng mô hình như Yolo để nhúng vào hệ thống, vậy nên qua quá trình nghiên cứu chúng tôi nhận thấy rằng mô hình ResNet-18 là phù hợp.

a. Dataset and data augmentation

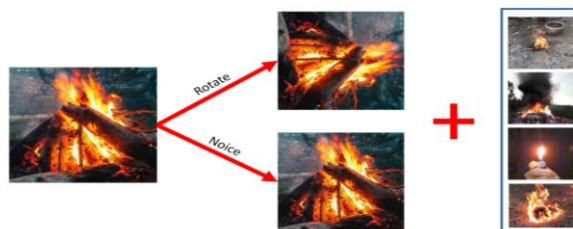
Chúng tôi đặt tên cho bộ dữ liệu là Dataset_B có tổng 1370 mẫu, được chia ra trong đó có 690 mẫu tích cực là những hình ảnh của đám cháy và 680 mẫu tiêu cực là những hình ảnh không chứa đám cháy. Bạn có thể thấy cấu trúc của bảng dữ liệu ở hình 3.1.1 và mẫu ở hình 3.1.2

- **Dataset_B** has 690 positive images and 680 negative images, many different environments.

Dataset	Class	Toal	Train	Test
Dataset_B	Fire	690	550	140
	No_Fire	680	550	130

Hình 3.1.1: Bộ dữ liệu của mô hình

Trong bộ dữ liệu này chúng tôi đã xử lý một cách rất cẩn thận, trong Deep Learning đó gọi là “Data augmentation”. Điều này giúp mô hình có cái nhìn tổng quan về các trường hợp mà nó sẽ phải gặp phải trong thực tế.



Hình 3.1.2: Các mẫu dữ liệu thu thập

Hình ảnh được tinh chỉnh bằng việc xoay ảnh, thêm nhiễu, lấy mẫu ở nhiều môi trường cháy khác nhau.

b. Kiến trúc của mô hình

Mô hình ResNet-18 là một trong những phiên bản của mạng nơ-ron tích chập (CNN) thuộc dòng ResNet (Residual Networks), được giới thiệu bởi Kaiming He và cộng sự trong bài báo "Deep Residual Learning for Image Recognition" vào năm 2015. ResNet nổi tiếng với cơ chế "residual learning", giúp giảm thiểu vấn đề biến mất gradient trong quá trình huấn luyện mạng sâu.

Dưới đây là kiến trúc của mô hình ResNet-18:

Đầu vào (Input): Hình ảnh có kích thước 224x224 pixel (RGB).

Convolutional Layer (Conv1): Convolution với 64 bộ lọc kích thước 7x7, và bước nhảy (stride) là 2. Sau đó là một lớp Batch Normalization và hàm kích hoạt ReLU.

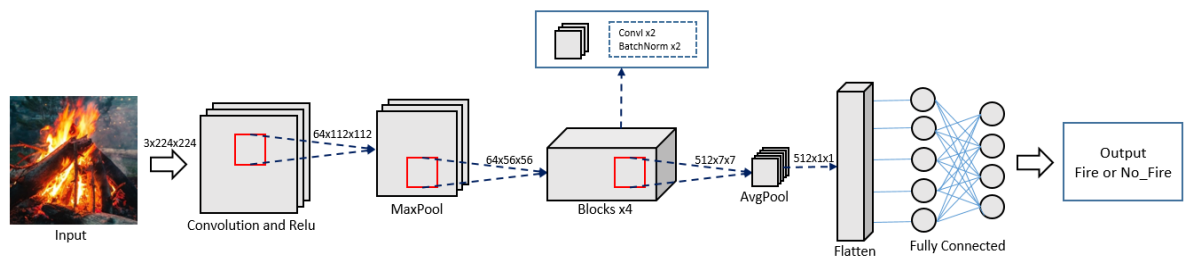
Lớp Max Pooling: Max pooling với kích thước cửa sổ 3x3 và bước nhảy là 2.

Residual Blocks (Basic Blocks): Bốn khối cơ bản (Basic Blocks) được sắp xếp liên tiếp. Mỗi khối bao gồm hai lớp convolution với số lượng bộ lọc ở mức trung bình (ví dụ, 64 hoặc 128 bộ lọc). Sử dụng kỹ thuật residual learning, trong đó đầu vào được thêm vào đầu ra của các lớp convolution.

Lớp Fully Connected (FC): Sau các khối cơ bản, một lớp trung gian đầy đủ kết nối với số nơ-ron ít hơn được sử dụng trước lớp softmax cuối cùng.

Lớp Softmax: Dùng để dự đoán xác suất của các lớp đối tượng.

Tóm lại, ResNet-18 có kiến trúc tổ chức các khối cơ bản theo cách cố định để tạo ra một mô hình sâu với khả năng học tốt. Cơ chế residual learning giúp đối mặt với vấn đề biến mất gradient và giúp việc huấn luyện mô hình trở nên hiệu quả hơn.

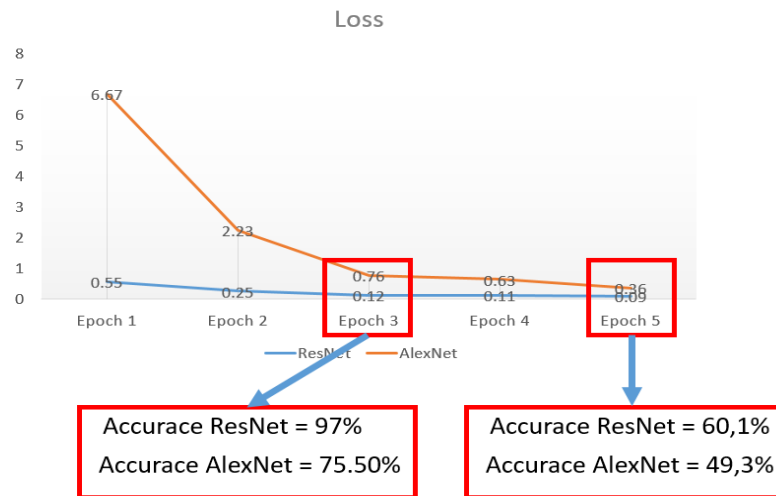


Hình 3.1.3: Kiến trúc của mô hình

Cuối cùng đầu ra của kiến trúc nó sẽ đưa cho bạn kết quả của hình ảnh đó là **cháy** hay **không cháy**.

c. Đào tạo mô hình

Để chứng minh cho việc mô hình ResNet là tốt hơn trong bài toán của chúng tôi. Chúng tôi đã xây dựng các mô hình hình phát hiện ảnh đám cháy dựa trên 2 kiến trúc là ResNet và AlexNet. Trong quá trình training mô hình chúng tôi đã ghi chép lại các giá trị hàm Loss của cả hai mô hình và biểu diễn chúng trên một biểu đồ phía dưới.



Hình 3.1.4: Biểu đồ thể hiện hàm Loss

Bạn có thể thấy với Epoch = 5 thì Loss của ResNet là 0.09 còn AlexNet là 0.36 với chỉ số này có thể thấy rằng độ lệch dự đoán của ResNet là thấp hơn AlexNet điều đó chứng tỏ mô hình ResNet có độ chính xác cao hơn. Khi đánh giá trên tập data_test kết quả cho ra đúng như dự đoán với ResNet đạt 60.1% và AlexNet đạt 49.3%.

Nhưng độ chính xác ấy vẫn còn quá thấp chúng tôi đã phải tinh chỉnh lại mô hình, dựa vào biểu đồ hàm Loss chúng tôi đã phát hiện tại điểm Epoch = 3 giá trị của hàm Loss đã gần như không có sự giảm mạnh nữa vậy nên chúng tôi đã xây dựng lại mô hình với Epoch=3. Điều này đã giúp mô hình tránh được tình trạng học quá sâu (overfitting) và tăng độ chính xác của mô hình lên 97% với ResNet và 75.50% với AlexNet. Qua quá trình xây dựng mô hình này đã chứng minh mô hình ResNet là tốt hơn với dữ liệu của chúng tôi.

d. Kết quả

Nhóm đã thực hiện xây dựng các mô hình trên cả hai kiến trúc là ResNet-18 và AlexNet, tất cả các kiến trúc được xây dựng dựa trên 2 bộ dữ liệu là dataset_A và dataset_B bộ dataset_A là bộ dữ liệu chưa tăng cường nên có thể thấy độ chính xác trên tập Test_P là thấp hơn. Bộ dữ liệu dataset_B là bộ dữ liệu đã được tăng cường nên độ chính xác đã đạt 98%.

Dưới đây là bảng tổng hợp các mô hình đã được xây dựng:

Model	Dataset	Epoch	Loss	Accurace/Test(%)	Accurace/Test_P(%)	Note
ResNet	Dataset_A	5	0.05	60.1	-	-
	Dataset_A	3	0.09	97	71	Không đạt
	Dataset_B	3	0.02	99.69	98.57	Đạt
AlexNet	Dataset_A	5	0.36	49.3	-	-
	Dataset_A	3	0.95	75.51	-	-

Hình 3.1.5: Bảng kết quả các mô hình

Mô hình ResNet với bộ dữ liệu B đã cho kết quả cao nhất, đây cũng sẽ là mô hình được áp dụng vào trong sản phẩm.

3.2 Cascade Classifier của OpenCV

Ngoài việc sử dụng mô hình ResNet-18 chúng tôi còn sử dụng thêm lớp phân loại của OpenCV có tên Cascade Classifier. Đây là phương pháp xử lý ảnh truyền thống giúp nhận diện các đối tượng dựa trên các đặc trưng của đối tượng. Phương pháp này có một số hạn chế như việc độ chính xác kém, dễ bị nhiễu bởi môi trường, không thể bao quát được tổng thể các trường hợp khác nhau. Nhưng nó có một ưu điểm đặc biệt đó là giúp hệ thống có thể tránh khỏi việc xác định các hình họa về đám cháy. Cụ thể nó sẽ bỏ qua các hình vẽ đám cháy được in trên áo, xe và chỉ xác nhận các điểm có khả năng là cháy thật. Điều mà các mô hình Deep Learning thường bị mắc phải.

Quá trình xử lý ảnh và phân loại sẽ được mô phỏng như sau:



Hình 3.2.1: Mô tả quá trình hoạt động

Hình ảnh từ camera sẽ được đưa vào bộ phân loại của openCV sau đó sẽ cắt ra vùng nghi là đám cháy và đưa vào mô hình ResNet-18 để phân loại. Việc áp dụng cả hai phương pháp vào bài toán giúp cho sản phẩm có độ chính xác cao hơn, tránh khỏi tình trạng bắt nhầm đối tượng.

4. Thành phần của thiết bị

Trong phần này chúng tôi sẽ giới thiệu các thành phần của hệ thống, bao gồm phần mềm và phần cứng.

4.1 Phần mềm

Chúng tôi sử dụng mô hình Deep Learning, có tên là ResNet-18 để phát triển mô hình phát hiện đám cháy từ hình ảnh được cung cấp bởi camera của hệ thống. Hình ảnh sẽ được thu từ nhiều góc độ khác nhau trong khu vực phạm vi hoạt động của thiết bị. Khi xảy ra cháy hình ảnh sẽ được xử lý qua mô hình và đưa ra kết quả là *cháy* hay *không cháy*.

4.1.1 Phần mềm cho máy tính nhúng Raspberry Pi 4

Máy tính nhúng Raspberry Pi 4 đóng vai trò quan trọng như một trung tâm xử lý trong hệ thống của chúng tôi. Sử dụng ngôn ngữ lập trình Python trên nền tảng Linux, chúng tôi tận dụng sức mạnh và linh hoạt của Raspberry Pi để thực hiện những chức năng quan trọng như:

Trích Xuất Đặc Trưng: Hình ảnh đầu vào được đưa qua mô hình ResNet-18 để trích xuất các đặc trưng quan trọng liên quan đến việc nhận diện đám cháy. Mô hình đã được huấn luyện trước với dữ liệu lớn để đảm bảo khả năng tổng quát hóa và độ chính xác cao.

Phân Loại Đám Cháy: Kết quả từ mô hình là một dự đoán về việc có hay không có đám cháy trong hình ảnh. Kết quả này được chuyển đến phần mềm điều khiển để thực hiện các hành động tiếp theo.

Quyết Định Hành Động: Dựa trên kết quả nhận diện, Raspberry Pi có khả năng đưa ra quyết định về việc cần phải điều chỉnh vị trí của camera để theo dõi đám cháy. Thông tin này được chuyển đến Arduino để điều chỉnh trục của 2 động cơ quay đến trung tâm của đám cháy.

Xử lý đám cháy: Khi Raspberry Pi xác nhận có đám cháy thông qua mô hình Deep Learning ResNet-18, nó sẽ gửi thông tin chi tiết tới Arduino để kích hoạt các biện pháp xử lý đám cháy một cách nhanh nhất.

4.1.2 Phần mềm cho vi điều khiển Arduino

Arduino, với ngôn ngữ lập trình C và môi trường Arduino IDE, chịu trách nhiệm trong việc điều khiển các trục của hệ thống. Điều này bổ sung chức năng cơ bản nhưng quan trọng để tối ưu hóa vị trí và góc quan sát của camera trong trường hợp có đám cháy.

Một trong những chức năng quan trọng của phần mềm trên Arduino là quản lý và điều khiển các trục của mô hình. Điều này có thể bao gồm việc di chuyển camera hoặc các thiết bị khác để đảm bảo góc nhìn tối ưu vào khu vực có đám cháy. Cụ thể:

Điều Chỉnh Góc Quan Sát: Phần mềm được lập trình để điều chỉnh góc quan sát của camera dựa trên dữ liệu từ Raspberry Pi 4 cung cấp. Khi có thông báo về đám cháy, Arduino có thể di chuyển camera để đảm bảo rằng đám cháy luôn nằm ở trung tâm của camera.

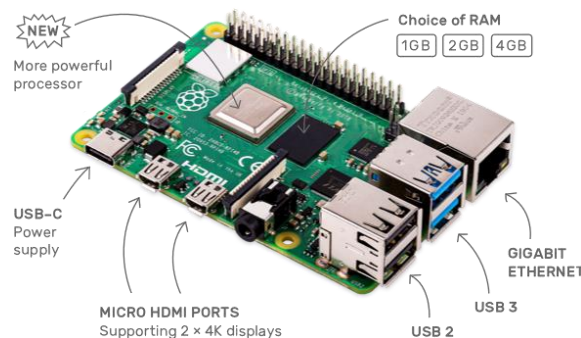
Quản Lý Vị Trí Đám Cháy: Arduino có khả năng xử lý cách thức di chuyển camera để duy trì tầm nhìn hiệu quả nhất. Điều này bao gồm việc quay trục để theo dõi chuyển động của đám cháy và điều chỉnh hướng để nhìn rõ hơn vào khu vực cần theo dõi. Nó sẽ quét và theo dõi một vùng có động rộng là 180 độ.

Xử lý đám cháy: Khi Arduino nhận được vị trí của đám cháy từ Raspberry Pi một quá trình tương tác chặt chẽ bắt đầu để điều chỉnh trục của mô hình sao cho vòi nước của hệ thống hướng đúng vào vị trí của đám cháy. Đồng thời, Arduino có nhiệm vụ bật công tắc máy bơm để triệt tiêu, dập tắt đám cháy.

4.2 Phần cứng

Chúng tôi đã triển khai một hệ thống với cấu trúc phần cứng đa dạng để đảm bảo hiệu suất và độ tin cậy trong quá trình phát hiện và xử lý đám cháy. Cụ thể, chúng tôi sử dụng các thành phần sau:

Máy Tính Nhúng: Raspberry Pi 4 là trung tâm xử lý của hệ thống, chịu trách nhiệm thu thập và xử lý hình ảnh từ camera.



Hình 4.2.1: Raspberry Pi4

Vi Điều Khiển: Arduino là bộ não của các hành động cơ khí và biện pháp phòng cháy. Nó nhận thông tin từ Raspberry Pi và thực hiện các hành động được yêu cầu.

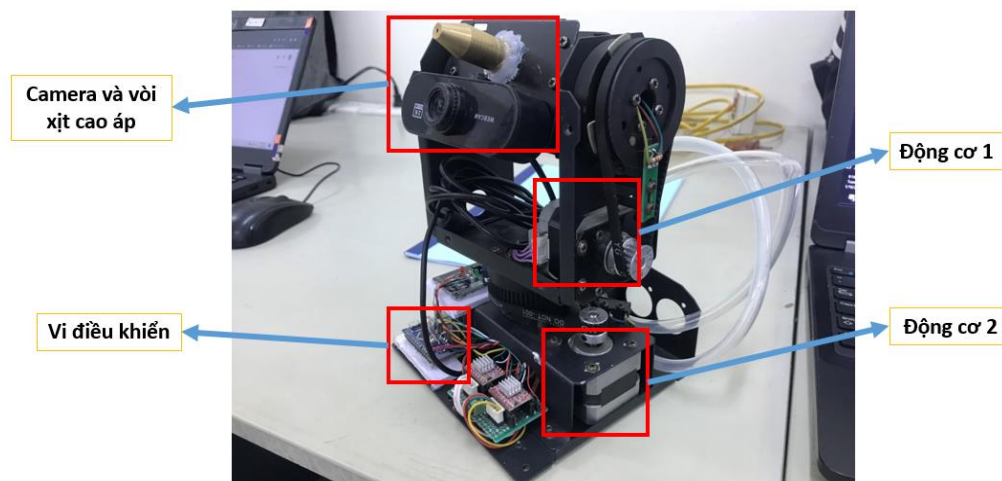
Thu Phát Hình Ảnh: Camera chất lượng cao với độ phân giải 2K giúp chính xác hóa quá trình nhận diện đám cháy.

Thiết Kế Đốc Đáo: Chúng tôi đã tự thiết kế một bộ khung đặc biệt để lắp đặt và giữ vững cả camera và các động cơ bước. Bộ khung này có khả năng đặt ở các vị trí cao và đảm bảo quan sát hiệu quả của khu vực cần bảo vệ.

Động Cơ Bước: Để thực hiện việc giám sát vùng cần bảo vệ cháy, và dẫn hướng cho vòi xịt cao áp để dập tắt đám cháy.

Vòi Nước Cao Áp: Được kích hoạt bởi Arduino khi cần thiết để triển khai biện pháp phòng cháy.

Bạn có thể nhìn ảnh của thiết bị ở hình dưới đây.



Hình 4.2.1: Thiết bị phần cứng

Thiết bị này sẽ được lắp đặt trên cao có thể là một góc tường hoặc bất cứ nơi nào có thể quan sát được khu vực mà bạn muốn bảo vệ. Vậy là tôi đã trình bày xong về phần cứng và phần mềm của thiết bị. Bây giờ chúng ta sẽ đi tới phần rất thú vị đó là chạy thử thiết bị.

5. Chạy thử thiết bị

Chúng tôi đã lắp đặt và chạy thử hệ thống tại một nhà bếp bỏ hoang, thuộc trường Đại học Sư phạm Kỹ Thuật Hưng Yên.

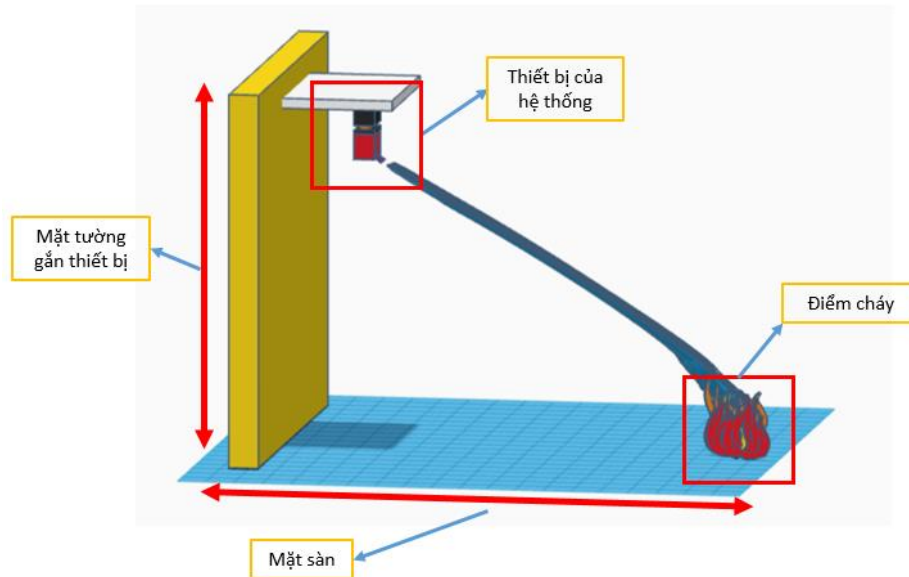
Phạm vi hoạt động của thiết bị(demo) theo như chúng tôi đo đạc là khoảng 3M bán kính. Nếu thiết bị được phát triển bởi những phần cơ khí tốt hơn, mạnh mẽ hơn, chính xác hơn thì phạm vi hoạt động của thiết bị có thể lên tới hàng chục mét bán kính.

Bởi phạm vi hoạt động của hệ thống này chủ yếu được tính bởi áp lực phụt của vòi phun nước. nếu vòi phun nước phụt càng xa thì phạm vi hoạt động sẽ càng cao.

Trong quá trình chạy thử hệ thống, hệ thống được lắp đặt trên một vách tường cách mặt đất 1,35m. Chiều cao này có thể thay đổi tùy thuộc vào mong muốn. Tối đa cho chiều cao này là khoảng 3m (với thiết bị demo) .

Chúng tôi đã mô tả lại quá trình hoạt động bằng một hình ảnh 3D.

Hình ảnh mô tả 3D của hệ thống:



Hình 3.2: Hình ảnh 3D của hệ thống

Như các bạn đã thấy trong hình ảnh 3D, thiết bị được lắp đặt tại một vị trí là gần tường sau đó khi nó phát hiện cháy, nó đã tự động xoay vòi nước sang hướng đó và phụt chính xác vào điểm cháy. Có thể với hình ảnh này không đủ để các bạn có thể thấy rõ được cách hoạt động và độ chính xác của mô hình, tôi sẽ đính kèm 1 link video (tại mục *liên hệ*) nếu bạn muốn xem video.

6. Kết luận

Kết luận của chúng tôi đề cập đến việc áp dụng mô hình deep learning trong việc phát hiện và xử lý đám cháy đóng vai trò quan trọng trong việc cải thiện hiệu suất và chính xác của các thiết bị phát hiện cháy.

Việc sử dụng deep learning, đặc biệt là các mô hình như CNN, đã đem lại nhiều cơ hội và tiềm năng trong việc nhận diện cháy thông qua hình ảnh và video. Sự tự động hóa quá trình tìm kiếm các đặc điểm quan trọng của cháy trong môi trường đồng thời với việc huấn luyện mô hình sử dụng một lượng lớn dữ liệu đã cải thiện đáng kể khả năng phát hiện và xử lý các tình huống cháy đa dạng.

Chúng tôi cũng đã chỉ ra rằng so với các phương pháp truyền thống sử dụng mô hình màu sắc hoặc các kỹ thuật thủ công, deep learning cung cấp khả năng nhận diện và xử lý cháy một cách hiệu quả hơn, không chỉ dựa trên màu sắc mà còn dựa vào cả thông tin không gian và thời gian. Điều này làm giảm thiểu các lỗi phát hiện giả mạo và cung cấp kết quả chính xác hơn trong việc xác định cháy và vùng nguy hiểm.

Tuy nhiên, còn một số thách thức cần được vượt qua như việc tăng cường khả năng phân loại các đối tượng không liên quan và hiệu suất của mô hình trong các điều kiện môi trường đa dạng. Sự tiếp tục nghiên cứu và cải thiện các mô hình deep learning có thể giúp giải quyết những thách thức này và tạo ra các thiết bị phát hiện cháy có khả năng phản ứng nhanh chóng và chính xác hơn trong các tình huống cấp bách.

Và tất nhiên với thiết bị của chúng tôi, chúng tôi hy vọng rằng việc sản xuất và lắp đặt hệ thống ở những nơi cần bảo vệ hỏa hoạn sẽ là rất hữu ích.

Liên hệ:

1. Trường Đại học Sư phạm Kỹ thuật Hưng Yên: <https://qldaotao.utehy.edu.vn/>
2. Email liên hệ nhóm nghiên cứu: dochiiot@gmail.com
3. Link video:
https://drive.google.com/file/d/1GiJTXpKeLTYPF65FViPxQDP4EjB00J1_/view?usp=sharing

Tài liệu tham khảo:

1. Kiến trúc ResNet: https://www.researchgate.net/figure/ResNet-18-architecture-a-basic-building-block-of-residual-learning-b-filter_fig3_334710759
2. Tài liệu tham khảo nước ngoài: <https://ieeexplore.ieee.org/document/9805875>
3. A Video-Based Fire Detection Using Deep Learning Models by Byoungjun KimORCID andJoonwhoan Lee * : <https://www.mdpi.com/2076-3417/9/14/2862>
4. PHÁT HIỆN ĐÁM CHÁY SỬ DỤNG PHƯƠNG PHÁP HỌC CHUYỂN TIẾP VỚI MẠNG EFFICIENTNETFIRE DETECTIONBY TRANSFER LEARNING WITH EFFICIENTNET NGUYỄN HỮU TUÂN*, TRỊNH THỊNGỌC HƯƠNG Khoa Công nghệThông tin, Trường Đại học Hàng hải Việt Nam: <https://jmst.vimaru.edu.vn/index.php/tckhcnhh/article/view/96>

Các nghiên cứu liên quan:

1. Chen, T.H.; Wu, P.H.; Chiou, Y.C. An early fire-detection method based on image processing. In Proceedings of the International Conference on Image Processing (ICIP), Singapore, 24–27 October 2004; pp. 1707–1710. [Google Scholar]
2. Celik, T.; Demirel, H.; Ozkaramanli, H.; Uyguroglu, M. Fire detection using statistical color model in video sequences. J. Vis. Commun. Image Represent. 2007, 18, 176–185. [Google Scholar] [CrossRef]
3. Wang, T.; Shi, L.; Hou, X.; Yuan, P.; Bu, L. A new fire detection method based on flame color dispersion and similarity in consecutive frames. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 151–156. [Google Scholar]
4. Borges, P.V.K.; Izquierdo, E. A probabilistic approach for vision-based fire detection in videos. IEEE Trans. Circuits Syst. Video Technol. 2010, 20, 721–731. [Google Scholar] [CrossRef]
5. Mueller, M.; Karasev, P.; Kolesov, I.; Tannenbaum, A. Optical flow estimation for flame detection in videos. IEEE Trans. Image Process. 2013, 22, 2786–2797. [Google Scholar] [CrossRef] [PubMed]

6. Foggia, P.; Saggese, A.; Vento, M. Real-time fire detection for video surveillance applications using a combination of experts based on color, shape, and motion. IEEE Trans. Circuits Syst. Video Technol. 2015, 25, 1545–1556. [Google Scholar] [CrossRef]
7. Swin-YOLOv5: Research and Application of Fire and Smoke Detection Algorithm Based on YOLOv5 by Shangjie Ge Zhang,¹Fengxi Zhang,²Yuyang Ding,³and Yu Li:
<https://www.hindawi.com/journals/cin/2022/6081680/>
8. <https://github.com/spacewalk01/yolov5-fire-detection>
9. D. Sarkar, R. Bali, and T. Ghosh, Hands-on transfer learning with Python: implement advanced deep learning and neural network models using TensorFlow and Keras. 2019.
10. Haar Cascade: https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html