

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC PHENIKAA**



BÁO CÁO BÀI TẬP LỚN

HỌC PHẦN HỆ NHÚNG

**ĐỀ TÀI: THIẾT KẾ HỆ THỐNG GIÁM SÁT AN NINH TỰ ĐỘNG
BẰNG HÌNH ẢNH**

Nhóm: 05

Giảng viên hướng dẫn	: Phạm Ngọc Hưng
Lớp học phần	: Hệ nhúng 1-1-24(N06)
Danh sách thành viên	: Bùi Thị Anh Đào – 21012864
	Nguyễn Thị Ngọc Lan – 21012069
	Đỗ Hoàng Giang - 21012054

Hà Nội, Năm 2024

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC PHENIKAA



BÁO CÁO BÀI TẬP LỚN

HỌC PHẦN HỆ NHÚNG

ĐỀ TÀI: THIẾT KẾ HỆ THỐNG GIÁM SÁT AN NINH TỰ ĐỘNG
BẰNG HÌNH ẢNH

Nhóm: 05

Giảng viên hướng dẫn	: Phạm Ngọc Hưng
Lớp học phần	: Hệ nhúng 1-1-24(N06)
Danh sách thành viên	: Bùi Thị Anh Đào – 21012864
	Nguyễn Thị Ngọc Lan – 21012069
	Đỗ Hoàng Giang - 21012054

Hà Nội, Năm 2024

Mục lục

Bảng phân công công việc	1
Danh sách hình ảnh.....	2
Danh sách bảng biểu	3
1. Mở đầu	4
1.1. Lý do chọn đề tài	4
1.2. Mục tiêu và phạm vi của đề tài.....	4
1.2.1. Mục tiêu	4
1.2.2. Phạm vi đề tài.....	4
1.3. Phương pháp nghiên cứu	5
1.4. Cấu trúc của báo cáo.....	5
2. Kiến trúc hệ thống.....	6
2.1. Giới thiệu về hệ thống nhúng	6
2.2. Tổng quan về hệ thống giám sát an ninh tự động bằng hình ảnh.....	7
2.2.1. Sơ đồ khối của hệ thống.....	7
2.2.2. Luồng dữ liệu trong hệ thống.....	8
2.3. Các thành phần chính của hệ thống	9
2.3.1. Phần cứng.....	9
2.3.2. Phần mềm.....	11
3. Thiết kế và triển khai phần cứng.....	12
3.1. Phương pháp sử dụng máy tính cá nhân (PC/Laptop) kết hợp với webcam (triển khai chính)	12
3.1.1. Các thành phần của phần cứng	12
3.1.2. Triển khai phần cứng	13
3.1.3. Ưu và nhược điểm.....	14

3.2.	Phương pháp sử dụng thiết bị nhúng (kit Raspberry Pi) kết hợp với webcam (tham khảo)	15
3.2.1.	Các thành phần phần cứng	15
3.2.2.	Triển khai phần cứng	16
3.2.3.	Ưu và nhược điểm.....	17
3.3.	So sánh tổng quan hai phương pháp	17
3.4.	Các yếu tố để lựa chọn phương pháp phù hợp	19
4.	Thiết kế và triển khai phần mềm.....	21
4.1.	Mục tiêu thiết kế phần mềm	21
4.2.	Lựa chọn công nghệ và ngôn ngữ lập trình	21
4.3.	Thiết kế và triển khai phần mềm cho phương pháp 1 (máy tính với webcam – triển khai chính)	22
4.3.1.	Phần mềm và công cụ sử dụng	22
4.3.2.	Cài đặt và cấu hình phần mềm.....	22
4.4.	Thiết kế và triển khai phần mềm cho phương pháp 2 (Raspberry Pi với webcam – tham khảo).....	23
4.4.1.	Phần mềm và công cụ sử dụng	23
4.4.2.	Cài đặt và cấu hình phần mềm.....	23
4.5.	Phân tích và đánh giá phần mềm của hai phương pháp	24
5.	Kết luận	27
5.1.	Kết quả đã đạt được	27
5.2.	Những thách thức và giải pháp	28
5.3.	Định hướng phát triển trong tương lai.....	28

Bảng phân công công việc

Họ và tên	MSSV	Vai trò	Đánh giá
Bùi Thị Anh Đào	21012864	Trưởng nhóm Tổng hợp báo cáo, xây dựng và phát triển ứng dụng chính.	10/10
Đỗ Hoàng Giang	21012054	Hỗ trợ làm báo cáo, thuyết trình	10/10
Nguyễn Thị Ngọc Lan	21012069	Hỗ trợ làm báo cáo, làm powerpoint	10/10

Bảng 1: Bảng phân công công việc

Danh sách hình ảnh

Hình 1: Sơ đồ khối của hệ thống giám sát an ninh tự động bằng hình ảnh.....	8
Hình 2: Webcam	10
Hình 3: Thiết bị nhúng: Bộ kit Raspberry Pi.....	10

Danh sách bảng biểu

Bảng 1: Bảng phân công công việc	1
Bảng 2: Các thành phần phần cứng chính của hệ thống.....	10
Bảng 3: Công nghệ và thư viện phần mềm sử dụng.....	11
Bảng 4: Các thành phần phần cứng khi sử dụng máy tính và webcam.....	13
Bảng 5: Các thành phần phần cứng khi sử dụng Raspberry với webcam.....	16
Bảng 6: So sánh tổng quan hai phương pháp sử dụng máy tính & webcam với sử dụng Raspberry & webcam	18
Bảng 7: Đánh giá phần mềm hệ thống khi triển khai bằng hai phương pháp máy tính-webcam và Raspberry-webcam.....	25

1. Mở đầu

1.1. Lý do chọn đề tài

Trong thời đại công nghệ 4.0, an ninh và an toàn đã trở thành mối quan tâm hàng đầu của xã hội. Sự gia tăng về tội phạm và các hành vi vi phạm pháp luật đòi hỏi các giải pháp giám sát hiệu quả hơn để bảo vệ con người và tài sản. Hệ thống giám sát an ninh bằng hình ảnh đã chứng minh được vai trò quan trọng trong việc phát hiện, ngăn chặn và xử lý kịp thời các sự cố về người và của.

Các hệ thống giám sát truyền thống thường yêu cầu sự theo dõi liên tục từ con người, dẫn đến việc tiêu tốn nhiều nguồn lực và không đảm bảo hiệu quả tối ưu. Ngoài ra, khả năng phản ứng chậm và phụ thuộc vào yếu tố con người có thể gây ra những rủi ro không mong muốn. Tuy nhiên, sự phát triển mạnh mẽ của trí tuệ nhân tạo (AI) và thị giác máy tính đã mở ra cơ hội mới cho lĩnh vực giám sát an ninh. Các thuật toán nhận dạng và theo dõi đối tượng cho phép hệ thống tự động phân tích hình ảnh và video, giảm thiểu sự can thiệp của con người và nâng cao hiệu suất giám sát.

Chính vì vậy, chúng tôi chọn mô hình YOLO và thuật toán theo dõi chuyển động để thiết kế một hệ thống giám sát an ninh bằng hình ảnh. YOLO (You Only Look Once) là một trong những mô hình nhận dạng vật thể nhanh và chính xác nhất hiện nay, cho phép nhận diện đối tượng trong thời gian thực. Kết hợp với thuật toán theo dõi chuyển động như CentroidTracker, hệ thống có thể theo dõi liên tục các đối tượng di chuyển trong khung hình. Việc tích hợp hai thuật toán này trong một hệ thống nhúng sẽ tạo ra giải pháp giám sát an ninh hiệu quả, tự động và đáng tin cậy.

1.2. Mục tiêu và phạm vi của đề tài

1.2.1. Mục tiêu

Chúng tôi mong muốn có thể phát triển một hệ thống có khả năng nhận dạng và theo dõi các đối tượng trong khung hình video một cách tự động, không cần sự can thiệp của con người. Tận dụng ưu điểm của YOLO trong việc nhận dạng vật thể và CentroidTracker trong việc theo dõi chuyển động để nâng cao hiệu quả giám sát. Đồng thời tạo ra một giao diện ứng dụng trực quan, dễ sử dụng, hiển thị thông tin về video giám sát, thời gian và các thông báo liên quan một cách thuận tiện nhất.

1.2.2. Phạm vi đề tài

Về phần cứng: sử dụng camera (webcam hoặc camera IP) và máy tính cá nhân hoặc kit Raspberry Pi để chạy chương trình Python xử lý hình ảnh trong thời gian thực.

Về phần mềm: sử dụng ngôn ngữ lập trình Python và các thư viện hỗ trợ như OpenCV, Tkinter, ultralytics YOLO, NumPy và SciPy để xử lý hình ảnh, phát hiện và theo dõi chuyển động.

Về chức năng hệ thống: tập trung vào việc nhận dạng và theo dõi các đối tượng chuyển động trong video, không đi sâu vào các vấn đề về bảo mật dữ liệu, lưu trữ đám mây hoặc truyền thông mạng.

1.3. Phương pháp nghiên cứu

Chúng tôi tiến hành nghiên cứu nguyên lý hoạt động của YOLO trong nhận dạng vật thể và CentroidTracker trong theo dõi chuyển động, đồng thời so sánh các thuật toán và mô hình khác nhau để lựa chọn giải pháp phù hợp nhất cho hệ thống.

Xác định các chức năng chính của các thành phần trong hệ thống, lựa chọn và cấu hình các thiết bị phần cứng cần thiết, đảm bảo khả năng hoạt động ổn định của hệ thống. Lập trình các module xử lý hình ảnh, nhận dạng và theo dõi đối tượng, thiết kế giao diện người dùng một cách trực quan và thuận tiện.

1.4. Cấu trúc của báo cáo

Báo cáo đề tài Thiết kế hệ thống giám sát an ninh bằng hình ảnh được chia thành các phần chính sau:

- Phần 1: Mở đầu: Trình bày lý do chọn đề tài, mục tiêu, phạm vi và phương pháp nghiên cứu.
- Phần 2: Kiến trúc hệ thống: Mô tả tổng quan về kiến trúc hệ thống, bao gồm các thành phần phần cứng và phần mềm, cách chúng tương tác với nhau.
- Phần 3: Thiết kế và triển khai phần cứng: Trình bày quá trình lựa chọn, cấu hình và triển khai các thiết bị phần cứng sử dụng trong hệ thống.
- Phần 4: Thiết kế và triển khai phần mềm: Chi tiết hóa quá trình phát triển phần mềm, bao gồm mô tả mã nguồn, các thuật toán chính và giao diện người dùng.
- Phần 5: Kết luận: Tổng kết những kết quả đạt được, đánh giá hiệu suất của hệ thống và đề xuất hướng phát triển trong tương lai.

2. Kiến trúc hệ thống

2.1. Giới thiệu về hệ thống nhúng

Hệ thống nhúng (embedded system) là một hệ thống máy tính được thiết kế để thực hiện một hoặc một vài nhiệm vụ chuyên biệt trong một hệ thống lớn hơn. Khác với máy tính đa chức năng như PC, hệ thống nhúng thường có những yêu cầu về hiệu suất, độ tin cậy và hạn chế tài nguyên. Ví dụ, một hệ thống nhúng có thể là một bộ điều khiển trong lò vi sóng, máy giặt, xe ô tô, hoặc thiết bị y tế. Hệ thống nhúng thường không được sử dụng như một máy tính độc lập mà nó sẽ là một phần không thể tách rời của sản phẩm hoặc thiết bị mà nó điều khiển.

Các thành phần chính của một hệ thống nhúng bao gồm:

- Vi xử lý (Processor): Đây là thành phần chính thực hiện các lệnh và điều khiển toàn bộ hệ thống, có thể là vi điều khiển (microcontroller) hoặc vi xử lý (microprocessor), tùy thuộc vào yêu cầu của hệ thống.

- Bộ nhớ (Memory): Bao gồm bộ nhớ ROM (lưu trữ chương trình) và RAM (lưu trữ tạm thời dữ liệu trong quá trình xử lý).

- Thiết bị ngoại vi (Peripherals): Các thiết bị ngoại vi có thể là cảm biến, màn hình, nút nhấn, mô-đun giao tiếp không dây (Wi-Fi, Bluetooth) và các thiết bị đầu vào/đầu ra khác kết nối với vi xử lý.

- Phần mềm nhúng (Embedded Software): Phần mềm này được lập trình để điều khiển các thành phần của hệ thống và thực hiện nhiệm vụ chính của hệ thống nhúng. Phần mềm thường được tối ưu hóa về dung lượng và hiệu năng.

- Các thiết bị đầu vào/đầu ra (I/O Devices): Các cổng kết nối hoặc thiết bị cho phép hệ thống nhúng tương tác với môi trường xung quanh, như nút nhấn, relay, hoặc màn hình.

Các ứng dụng của hệ thống nhúng rất đa dạng và phong phú, có thể kể đến như:

- Thiết bị di động: Các điện thoại thông minh và máy tính bảng sử dụng hệ thống nhúng để quản lý chức năng và tương tác người dùng.

- Thiết bị gia dụng thông minh: Các sản phẩm như tủ lạnh, máy giặt, và điều hòa không khí ngày nay thường được trang bị hệ thống nhúng để tự động hóa và điều khiển thông minh.

- Hệ thống giám sát an ninh: Các camera và cảm biến sử dụng hệ thống nhúng để giám sát và phát hiện các sự kiện bất thường.
- Ô tô thông minh: Các hệ thống điều khiển trong ô tô như hệ thống phanh ABS, điều khiển động cơ, và hệ thống định vị GPS đều dựa vào công nghệ nhúng.
- Thiết bị y tế: Các máy đo huyết áp, máy siêu âm, và các thiết bị theo dõi sức khỏe sử dụng hệ thống nhúng để thu thập và phân tích dữ liệu.
- Hệ thống tự động hóa công nghiệp: Các robot và dây chuyền sản xuất tự động sử dụng hệ thống nhúng để điều khiển và giám sát quy trình sản xuất.
- Internet of Things (IoT): Các thiết bị IoT như cảm biến môi trường, đồng hồ thông minh và thiết bị gia đình kết nối mạng đều dựa vào hệ thống nhúng để thu thập và truyền tải dữ liệu.
- Robot: Các robot tự động và robot công nghiệp sử dụng hệ thống nhúng để thực hiện các tác vụ phức tạp và tương tác với môi trường xung quanh.

Đề tài “Thiết kế hệ thống giám sát an ninh tự động bằng hình ảnh” là một ví dụ cụ thể của hệ thống nhúng. Trong dự án này, chúng tôi sẽ sử dụng máy tính cá nhân và webcam để xây dựng triển khai hệ thống giám sát. Máy tính sẽ kết nối với webcam để thu thập hình ảnh và sử dụng phần mềm để xử lý dữ liệu. Webcam sẽ ghi lại hình ảnh trong thời gian thực, trong khi phần mềm được lập trình trên máy tính sẽ phân tích hình ảnh, phát hiện các hoạt động đáng ngờ và giao tiếp với người dùng qua giao diện ứng dụng để cung cấp thông tin cảnh báo. Tương lai, hệ thống cũng có thể lưu trữ dữ liệu hình ảnh và thông tin liên quan vào cơ sở dữ liệu hoặc các thẻ nhớ SD để phục vụ cho việc theo dõi và phân tích sau này.

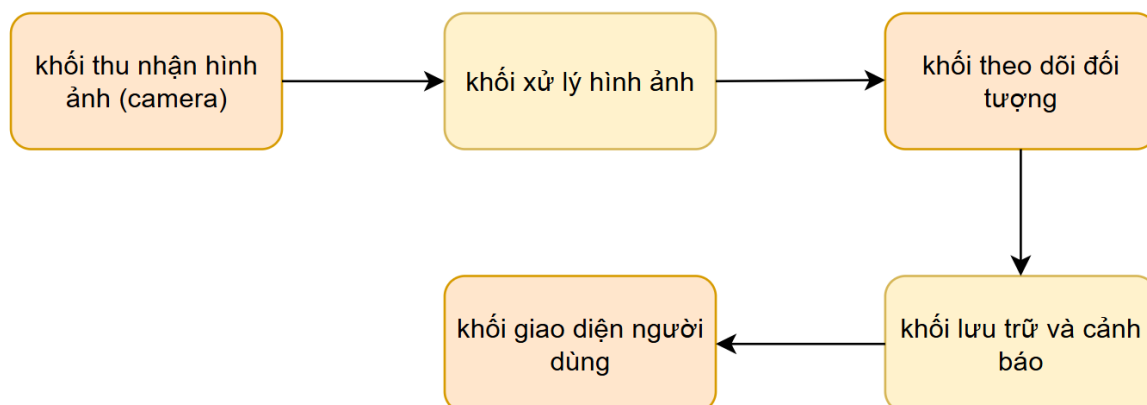
2.2. Tổng quan về hệ thống giám sát an ninh tự động bằng hình ảnh

2.2.1. Sơ đồ khối của hệ thống

Mô tả các khối trong sơ đồ:

- **Khối thu nhận hình ảnh:** sử dụng camera để thu nhận hình ảnh video từ môi trường giám sát.
- **Khối xử lý hình ảnh:** máy tính hoặc thiết bị nhúng (raspberry) nhận dữ liệu từ camera, thực hiện các thuật toán xử lý hình ảnh, bao gồm phát hiện chuyển động và nhận dạng vật thể.

- **Khối theo dõi đối tượng:** sử dụng thuật toán theo dõi để theo dõi các đối tượng đã được phát hiện trong khung hình.
- **Khối giao diện người dùng:** hiển thị hình ảnh video, thông tin thời gian và các thông báo liên quan trên giao diện.
- **Khối lưu trữ và cảnh báo:** lưu trữ dữ liệu cần thiết và gửi cảnh báo khi phát hiện có người/vật thể chuyển động trong khung hình.



Hình 1: Sơ đồ khối của hệ thống giám sát an ninh tự động bằng hình ảnh

2.2.2. Luồng dữ liệu trong hệ thống

Mô tả luồng dữ liệu:

- Thu nhận hình ảnh: camera liên tục quay video và gửi từng khung hình đến hệ thống.
- Xử lý hình ảnh: sử dụng bộ trừ nền (background subtraction) để phát hiện các khu vực có chuyển động trong khung hình. Áp dụng mô hình YOLO để nhận dạng các vật thể trong khung hình và lấy tọa độ bounding box.
- Theo dõi đối tượng: sử dụng các bounding box từ bước xử lý hình ảnh để theo dõi các đối tượng di chuyển trong khung hình bằng thuật toán theo dõi tọa độ trung tâm (CentroidTracker). Mỗi đối tượng được gán một ID duy nhất để theo dõi liên tục qua các khung hình.
- Cập nhật giao diện người dùng: khung hình sau khi được xử lý sẽ được hiển thị trên giao diện của ứng dụng, thời gian thực tế cũng được cập nhật liên tục, khi phát hiện đối tượng mới hoặc chuyển động thì hệ thống sẽ hiển thị thông báo

ngay trên giao diện, thông báo này đi kèm với thời gian mà đối tượng hoặc chuyển động được phát hiện.

- Lưu trữ và cảnh báo: thông tin về các đối tượng phát hiện được lưu trữ để phục vụ cho việc phân tích sau này, hệ thống cũng có thể được mở rộng để gửi cảnh báo qua email hoặc tin nhắn khi phát hiện sự kiện quan trọng (chức năng gửi cảnh báo qua email hoặc tin nhắn sẽ được nghiên cứu và triển khai trong tương lai)

Sự tương tác giữa các thành phần:

- Phần cứng (camera và máy tính/thiết bị nhúng): camera gửi dữ liệu hình ảnh đến máy tính/thiết bị nhúng để xử lý.
- Phần mềm (các thuật toán và giao diện người dùng): các thuật toán xử lý hình ảnh và theo dõi đối tượng hoạt động trên máy tính, kết quả được hiển thị trên giao diện người dùng.
- Người dùng: tương tác với hệ thống thông qua giao diện, xem thông báo và có thể thực hiện các hành động cần thiết (chức năng sẽ được cập nhật trong tương lai).

2.3. Các thành phần chính của hệ thống

2.3.1. Phần cứng

Danh sách các thiết bị phần cứng sử dụng:

Tên thiết bị	Chức năng	Cấu hình đề xuất	Lý do lựa chọn
Camera (webcam hoặc camera IP)	Thu thập hình ảnh video từ môi trường giám sát		Dễ dàng kết nối với máy tính, hỗ trợ độ phân giải cao và khả năng hoạt động liên tục.
Máy tính (PC/Laptop hoặc thiết bị nhúng/kit Raspberry)	Xử lý dữ liệu hình ảnh, chạy các thuật toán nhận dạng và theo dõi.	CPU: Intel Core i5 hoặc tương đương. RAM: Tối thiểu 8GB.	Đảm bảo khả năng xử lý hình ảnh real-time và chạy các

		GPU: Khuyến khích sử dụng GPU NVIDIA có hỗ trợ CUDA để tăng tốc xử lý (nếu sử dụng mô hình YOLO phiên bản lớn hơn).	mô hình AI một cách hiệu quả.
--	--	---	-------------------------------

Bảng 2: Các thành phần phần cứng chính của hệ thống



Hình 2: Webcam



Hình 3: Thiết bị nhúng: Bộ kit Raspberry Pi

2.3.2. Phần mềm

Các công nghệ và thư viện phần mềm sử dụng:

Tên của công nghệ/thư viện phần mềm	Lý do lựa chọn	Chức năng
Ngôn ngữ lập trình Python	Python là ngôn ngữ phổ biến trong lĩnh vực xử lý ảnh và trí tuệ nhân tạo, với cộng đồng lớn và nhiều thư viện hỗ trợ.	
Thư viện OpenCV	OpenCV là thư viện mạnh mẽ, hỗ trợ nhiều tính năng xử lý ảnh và video.	Xử lý ảnh và video, cung cấp các hàm hỗ trợ phát hiện chuyển động, xử lý hình ảnh.
Thư viện Tkinter	Tkinter là thư viện GUI tiêu chuẩn trong Python, dễ sử dụng và tích hợp.	Tạo giao diện người dùng đồ họa (GUI) cho ứng dụng.
Mô hình YOLO từ thư viện ultralytics	Mô hình YOLO cho phép nhận dạng đối tượng nhanh chóng và chính xác, phù hợp cho ứng dụng real-time.	Nhận dạng và theo dõi vật thể trong hình ảnh.
Thư viện NumPy và SciPy	NumPy và SciPy cung cấp các công cụ cần thiết cho xử lý dữ liệu trong Python	Xử lý các tính toán số học và khoa học, hỗ trợ các thao tác với mảng và ma trận
Thư viện PIL (Pillow)	PIL giúp dễ dàng chuyển đổi và hiển thị hình ảnh trong ứng dụng Python	Hỗ trợ xử lý hình ảnh, chuyển đổi định dạng hình ảnh để hiển thị trên giao diện

Bảng 3: Công nghệ và thư viện phần mềm sử dụng

3. Thiết kế và triển khai phần cứng

Trong phần này, chúng tôi sẽ trình bày chi tiết quá trình thiết kế và triển khai phần cứng cho hệ thống giám sát an ninh bằng hình ảnh. Hệ thống được xây dựng dựa trên hai phương pháp chính: Sử dụng máy tính cá nhân (PC/Laptop) kết hợp với webcam và Sử dụng thiết bị nhúng (Raspberry Pi) kết hợp với webcam. Do hạn chế về tài nguyên (thời gian, tài chính) nên hiện tại chúng tôi chỉ có thể tập trung xây dựng hệ thống giám sát an ninh với 2 phương pháp này.

3.1. Phương pháp sử dụng máy tính cá nhân (PC/Laptop) kết hợp với webcam (triển khai chính)

3.1.1. Các thành phần của phần cứng

Thành phần	Cấu hình tối thiểu	Cấu hình đề xuất
Máy tính cá nhân (PC/Laptop)	CPU: Intel Core i3 hoặc AMD Ryzen 3	CPU: Intel Core i5 hoặc AMD Ryzen 5 trở lên
	RAM: 4GB	RAM: 8GB hoặc cao hơn
	HDD/SSD: 128GB	SSD: 256GB hoặc hơn, ưu tiên SSD để tăng tốc độ truy xuất dữ liệu
	GPU: Tích hợp (Intel HD Graphics hoặc AMD Radeon Vega)	GPU: Card rời NVIDIA GeForce GTX 1050 hoặc AMD Radeon RX 560 trở lên (nếu cần xử lý hình ảnh nâng cao)
	Cổng kết nối: USB 2.0, cổng mạng Ethernet hoặc Wi-Fi	Cổng kết nối: USB 3.0/3.1, Ethernet Gigabit, Wi-Fi
	Hệ điều hành: Windows 7/8.1/10, Ubuntu Linux 18.04 trở lên hoặc macOS 10.12 trở lên	Hệ điều hành: Windows 10/11, Ubuntu Linux 20.04 trở lên hoặc macOS 10.15 trở lên
	Lưu ý nếu dùng laptop: cần đảm bảo có hệ thống tản nhiệt tốt để hoạt động liên tục, kết nối nguồn điện ổn định, tránh sử dụng	

	dùng pin trong thời gian dài, kiểm tra số lượng cổng USB và các kết nối cần thiết.	
Webcam	Độ phân giải: 720p (HD)	Độ phân giải: 720p (HD), 1080p (Full HD) hoặc cao hơn
	Tốc độ khung hình: 30fps	Tốc độ khung hình: 30fps hoặc 60fps (nếu cần ghi hình mượt mà)
	Kết nối: USB 2.0	Kết nối: USB 3.0
	Tính năng (cơ bản): Tự động cân bằng sáng, lấy nét cố định	Tính năng (cơ bản & nâng cao): Tự động cân bằng sáng, tự lấy nét, chống nhiễu (điều kiện ánh sáng yếu), góc nhìn rộng (wide-angle lens)
	Chú ý nếu sử dụng laptop: sử dụng camera có sẵn trên máy nên có thể bỏ qua các yêu cầu về cổng kết nối. Mẫu webcam đề xuất: Logitech C920/C922/C930e (chất lượng hình ảnh tốt, hỗ trợ nhiều tính năng), Microsoft LifeCam Studio (độ phân giải cao, tích hợp micro chất lượng).	

Bảng 4: Các thành phần phần cứng khi sử dụng máy tính và webcam

3.1.2. Triển khai phần cứng

- Bước 1: Chuẩn bị và kiểm tra thiết bị: kiểm tra cấu hình máy tính đáp ứng yêu cầu, cập nhật hệ điều hành và driver lên phiên bản mới nhất. Kiểm tra tình trạng vật lý của webcam (dây cáp, mắt cam,...), cài đặt driver nếu cần thiết (đối với Windows thường tự động nhận diện, với Linux cần kiểm tra hỗ trợ UVC).
- Bước 2: Kết nối và cài đặt:
 - + Kết nối webcam vào máy tính thông qua cổng USB (đối với PC, nếu dùng laptop có thể bỏ qua bước này).
 - + Cài đặt driver và phần mềm đi kèm (nếu có).
 - + Kiểm tra hoạt động của webcam bằng cách sử dụng ứng dụng Camera (trên Windows) hoặc Cheese (trên Linux), đảm bảo hình ảnh rõ nét, không bị mờ hay giật lag.

- Bước 3: Cài đặt phần mềm giám sát: cài đặt môi trường Python và các thư viện cần thiết cho việc chạy chương trình ứng dụng tùy chỉnh đã được thiết lập trên IDE (môi trường Python, các thư viện như OpenCV, Ultralytics, Tkinter,...)
- Bước 4: Lắp đặt và tối ưu hóa hệ thống:
 - + Đặt máy tính/laptop ở nơi khô ráo, thoáng mát, tránh tiếp xúc trực tiếp với ánh nắng hoặc nguồn nhiệt, gắn webcam ở vị trí có tầm nhìn bao quát khu vực cần giám sát (có thể sử dụng giá đỡ hoặc chân máy để cố định webcam).
 - + Khi sử dụng pc kết nối với webcam, cần sắp xếp dây cáp gọn gàng, tránh cản trở di chuyển cho người khác, có thể sử dụng ống luồn dây hoặc kẹp cáp để tổ chức dây dẫn.
 - + Chạy thử nghiệm trong khoảng thời gian nhất định để đảm bảo hệ thống hoạt động ổn định, đồng thời kiểm tra các chức năng phát hiện chuyển động và phát cảnh báo.

3.1.3. Ưu và nhược điểm

- Ưu điểm:
 - + Hiệu năng cao: với khả năng xử lý mạnh mẽ, phù hợp cho các ứng dụng đòi hỏi tính toán phức tạp và cần nhiều tài nguyên tính toán, có thể hỗ trợ nhiều camera cùng lúc mà không bị giảm hiệu suất.
 - + Dễ mở rộng và nâng cấp: máy tính có thể thêm được thiết bị ngoại vi, nâng cấp phần cứng (RAM, ổ cứng, GPU), hỗ trợ nhiều phần mềm và nền tảng khác nhau.
 - + Có tính linh hoạt: do sử dụng được cả PC và laptop, phù hợp với nhiều không gian triển khai hơn.
- Nhược điểm:
 - + Kích thước và tính di động: PC có thể chiếm diện tích và khó khăn trong việc di chuyển, laptop tuy nhỏ gọn và di động nhưng không được thiết kế cho việc hoạt động liên tục 24/7.
 - + Tiêu thụ điện năng cao: có thể tốn khá nhiều chi phí cho việc vận hành và duy trì

3.2. Phương pháp sử dụng thiết bị nhúng (kit Raspberry Pi) kết hợp với webcam (tham khảo)

3.2.1. Các thành phần phần cứng

Thành phần	Cấu hình tối thiểu	Cấu hình đề xuất
Thiết bị nhúng Raspberry Pi	Model: Raspberry Pi 3 Model B+	Model: Raspberry Pi 4 Model B
	CPU: Quad-core 1.4 GHz	CPU: Quad-core 64-bit 1.5GHz
	RAM: 1GB	RAM: 4GB hoặc 8GB
	Cổng kết nối: 4 x USB 2.0, HDMI, Ethernet, Wi-Fi	Cổng kết nối: <ul style="list-style-type: none"> - 2 x USB 3.0, 2 x USB 2.0. - 2 x micro-HDMI hỗ trợ 4K. - Ethernet Gigabit. - Wi-Fi, Bluetooth 5.0
Camera	Camera module V1: độ phân giải 5MP, hỗ trợ chụp ảnh và quay video HD 1080p. Hoặc: Webcam USB: độ phân giải 720p, kết nối USB 2.0.	Camera module V2 hoặc HQ Camera: độ phân giải 8MP (V2) hoặc 12.3MP (HQ Camera), hỗ trợ chụp ảnh và quay video Full HD, hỗ trợ ống kính thay đổi (với HQ Camera). Hoặc: Webcam USB: độ phân giải 1080p, kết nối USB 3.0 (trên Raspberry Pi 4)
Phụ kiện khác	Thẻ nhớ microSD: 16GB	Thẻ nhớ microSD: 32GB hoặc 64GB UHS-I/UHS-II để tăng tốc độ đọc/ghi

	Nguồn điện: Raspberry Pi 3 B+: Adapter 5V/2.5A micro-USB	Nguồn điện: Raspberry Pi 4: Adapter 5V/3A USB-C
	Vỏ bảo vệ, tản nhiệt: Vỏ có tích hợp quạt hoặc heatsink để tản nhiệt, đảm bảo hoạt động ổn định.	
	Màn hình, bàn phím, chuột (tùy chọn): sử dụng trong quá trình cài đặt và cấu hình ban đầu.	

Bảng 5: Các thành phần phần cứng khi sử dụng Raspberry với webcam

3.2.2. Triển khai phần cứng

- Bước 1: Chuẩn bị và cài đặt hệ điều hành:
 - + Có thể lựa chọn tải hệ điều hành Raspberry Pi OS Lite (không có giao diện đồ họa) nếu muốn tiết kiệm tài nguyên, hoặc Raspberry Pi OS Full (có giao diện đồ họa) nếu cần.
 - + Ghi hệ điều hành vào thẻ nhớ sử dụng Raspberry Pi Imager hoặc Etcher.
 - + Cấu hình truy cập từ xa: kích hoạt SSH bằng cách tạo file trống tên ‘ssh’ trong phân vùng boot, Cấu hình Wi-Fi bằng cách tạo file ‘wpa_supplicant.conf’ với thông tin mạng.
- Bước 2: Kết nối và khởi động Raspberry Pi: lắp thẻ nhớ vào Raspberry Pi, kết nối nguồn điện, màn hình, bàn phím (nếu cần), sau đó khởi động và thực hiện cài đặt các cấu hình ban đầu (đặt mật khẩu, cập nhật hệ thống, cài đặt ngôn ngữ, múi giờ,... nếu cần thiết).
- Bước 3: Kết nối camera:
 - + Nếu dùng Camera module: kết nối vào cổng CSI, đảm bảo cáp được cắm đúng chiều và chắc chắn. Sau đó kích hoạt camera bằng lệnh ‘sudo raspi-config’, chọn Interfacing Options -> Camera -> Enable.
 - + Nếu dùng Webcam USB: kết nối vào cổng USB, kiểm tra nhận diện với lệnh ‘lsusb’.
- Bước 4: Cài đặt phần mềm giám sát: có thể lựa chọn cài đặt phần mềm có sẵn như MotionEye (giao diện web cho phần mềm Motion, giám sát chuyển động của đối tượng), hoặc tự cài đặt và phát triển một ứng dụng giám sát tùy chỉnh bằng cách sử dụng ngôn ngữ lập trình và các thư viện liên quan.

- Bước 5: Lắp đặt và tối ưu hóa hệ thống:
 - + Đặt Raspberry Pi trong vỏ bảo vệ, tránh bụi bẩn và nhiệt độ cao, gắn camera ở vị trí phù hợp, sử dụng vỏ bảo vệ chống thời tiết nếu lắp đặt ngoài trời.
 - + Ngoài ra cần quản lý nhiệt độ bằng cách sử dụng quạt hoặc heatsink để tản nhiệt, tránh quá nhiệt gây giảm hiệu năng, khi lắp đặt cần sắp xếp dây gọn gàng, sử dụng dây cáp chất lượng tốt để đảm bảo tín hiệu ổn định.
 - + Chạy thử nghiệm, kiểm tra các chức năng và hiệu suất của hệ thống, điều chỉnh cấu hình nếu cần thiết.

3.2.3. Ưu và nhược điểm

- Ưu điểm:
 - + Kích thước nhỏ gọn và di động, dễ dàng lắp đặt ở nhiều vị trí, kể cả những nơi chật hẹp.
 - + Tiết kiệm năng lượng, tiêu thụ điện năng thấp, giảm chi phí vận hành.
 - + Chi phí thấp, đầu tư ban đầu ít, phù hợp với ngân sách hạn chế.
 - + Linh hoạt, có thể kết nối với nhiều loại cảm biến và module qua GPIO.
- Nhược điểm:
 - + Hiệu năng hạn chế, xử lý hình ảnh phức tạp có thể chậm hoặc không khả thi.
 - + Khả năng mở rộng hạn chế do số lượng cổng USB và băng thông hạn chế.
 - + Yêu cầu cao về kiến thức kỹ thuật, cần hiểu biết về Linux và lập trình nhúng.

3.3. So sánh tổng quan hai phương pháp

Để đưa ra quyết định lựa chọn phương pháp phù hợp cho hệ thống giám sát an ninh bằng hình ảnh, chúng ta có thể so sánh một số tiêu chí giữa hai phương pháp đã trình bày. Phần này sẽ giúp hiểu rõ hơn về ưu và nhược điểm của từng giải pháp, từ đó xác định được phương pháp tối ưu để sử dụng trong dự án.

Trước tiên chúng ta có bảng so sánh tổng quan giữa hai phương pháp như sau:

Tiêu chí	Máy tính với webcam	Raspberry với webcam
Hiệu năng xử lý	Cao, xử lý tác vụ phức tạp	Trung bình, phù hợp tác vụ đơn giản
Kích thước và khả năng di động	Lớn (desktop), vừa (laptop), khó di chuyển	Nhỏ gọn, dễ dàng di chuyển và lắp đặt

Tiêu thụ năng lượng	Cao (100W trở lên)	Thấp (~5W-15W)
Chi phí đầu tư	Cao (tùy cấu hình)	Thấp hơn đáng kể
Khả năng mở rộng	Dễ dàng, nhiều cổng kết nối	Hạn chế, cần thêm module nếu mở rộng
Độ phức tạp trong cài đặt	Dễ dàng với người dùng phổ thông	Cần kiến thức về Linux và lập trình nhúng
Tính ổn định	Cao, hoạt động liên tục	Tương đối, phụ thuộc vào cấu hình và cài đặt
Ứng dụng phù hợp	Hệ thống lớn, yêu cầu xử lý cao	Hệ thống nhỏ, di động, tiết kiệm chi phí

Bảng 6: So sánh tổng quan hai phương pháp sử dụng máy tính & webcam với sử dụng Raspberry & webcam

So sánh chi tiết:

- Hiệu năng xử lý:

Máy tính với webcam: Với cấu hình mạnh mẽ, máy tính cá nhân có khả năng xử lý các tác vụ phức tạp như phân tích video thời gian thực, nhận dạng khuôn mặt, phân tích hành vi. Điều này phù hợp cho các hệ thống giám sát an ninh quy mô lớn hoặc yêu cầu cao về phân tích dữ liệu.

Raspberry Pi với webcam: Hiệu năng hạn chế, phù hợp cho các tác vụ giám sát cơ bản như phát hiện chuyển động, ghi hình và gửi cảnh báo. Không thích hợp cho việc xử lý hình ảnh phức tạp do giới hạn về CPU và RAM.

- Kích thước và tính di động:

Máy tính với webcam: Kích thước lớn (đối với desktop) hoặc vừa (đối với laptop), khó lắp đặt ở những nơi hạn chế về không gian. Laptop dễ di chuyển hơn nhưng không được thiết kế để hoạt động liên tục 24/7.

Raspberry Pi với webcam: Kích thước nhỏ gọn, nhẹ, dễ dàng lắp đặt ở nhiều vị trí khác nhau, kể cả những nơi chật hẹp hoặc cần tính ẩn giấu.

- Tiêu thụ năng lượng:

Máy tính với webcam: Tiêu thụ điện năng cao (100W trở lên), dẫn đến chi phí vận hành lớn và không thân thiện với môi trường.

Raspberry Pi với webcam: Tiêu thụ điện năng thấp (~5W-15W), tiết kiệm chi phí vận hành, phù hợp cho hệ thống cần hoạt động liên tục và có nguồn điện hạn chế.

- Chi phí đầu tư:

Máy tính với webcam: Chi phí cao do phần cứng đắt tiền và có thể cần mua thêm phần mềm thương mại.

Raspberry Pi với webcam: Chi phí đầu tư ban đầu thấp, phù hợp với ngân sách hạn chế hoặc triển khai trên quy mô lớn.

- Khả năng mở rộng:

Máy tính với webcam: Dễ dàng mở rộng, kết nối nhiều thiết bị ngoại vi, nâng cấp phần cứng khi cần.

Raspberry Pi với webcam: Khả năng mở rộng hạn chế do số lượng cổng kết nối ít và băng thông hạn chế. Cần thêm module mở rộng nếu muốn kết nối nhiều thiết bị.

- Độ phức tạp trong cài đặt:

Máy tính với webcam: Dễ dàng cài đặt và sử dụng với người dùng phổ thông, giao diện thân thiện, hỗ trợ nhiều phần mềm.

Raspberry Pi với webcam: Yêu cầu kiến thức về Linux, lập trình nhúng, có thể gặp khó khăn trong quá trình cài đặt và cấu hình.

- Tính ổn định:

Máy tính với webcam: Hoạt động ổn định, có khả năng chịu tải cao và ít gặp sự cố khi được bảo trì đúng cách.

Raspberry Pi với webcam: Tính ổn định phụ thuộc vào cấu hình và cách cài đặt. Có thể gặp vấn đề về quá nhiệt hoặc quá tải nếu không được quản lý tốt.

- Ứng dụng phù hợp:

Máy tính với webcam: Thích hợp cho các hệ thống giám sát an ninh lớn, yêu cầu cao về xử lý dữ liệu và tính ổn định.

Raspberry Pi với webcam: Phù hợp cho các ứng dụng nhỏ, di động, tiết kiệm chi phí và không đòi hỏi cao về hiệu năng.

3.4. Các yếu tố để lựa chọn phương pháp phù hợp

Ngoài cách dựa trên các tiêu chí đã phân tích ở trên, việc lựa chọn giải pháp phù hợp cần dựa trên nhiều yếu tố cụ thể của dự án, các yếu tố có thể kể đến như:

- **Yêu cầu về hiệu năng và chức năng:** nếu dự án yêu cầu xử lý những hình ảnh phức tạp, hay quản lý và lưu trữ lượng lớn dữ liệu, tích hợp với hệ thống khác (hệ thống báo động, kiểm soát ra vào, quản lý trung tâm,...) thì phương pháp **sử dụng máy tính với webcam** sẽ phù hợp với dự án hơn do khả năng xử lý mạnh mẽ và có tính linh hoạt cao. Ngược lại, nếu dự án chỉ cần giám sát cơ bản, triển khai ở nhiều điểm với chi phí thấp và tiết kiệm năng lượng lẫn không gian lắp đặt, thì phương pháp **sử dụng Raspberry Pi với webcam** sẽ là lựa chọn hợp lý, đáp ứng đủ nhu cầu mà vẫn đảm bảo chi phí và tính linh hoạt.

- **Ngân sách và chi phí vận hành:** nếu dự án có kinh phí giới hạn, Raspberry Pi với webcam là lựa chọn tối ưu, giảm chi phí đầu tư ban đầu và chi phí vận hành nhờ tiêu thụ điện năng thấp. Nếu ngân sách không phải là vấn đề lớn và cần hiệu suất cao, máy tính với webcam có thể được đầu tư để đạt hiệu năng tối đa.

- **Không gian và môi trường triển khai:** Raspberry Pi có kích thước nhỏ gọn dễ dàng lắp đặt và di chuyển, phù hợp cho các vị trí khó tiếp cận, không gian hạn chế hoặc cần tính di động. Còn trong môi trường cố định, không hạn chế về không gian thì máy tính với webcam có thể được sử dụng, đảm bảo hiệu năng và khả năng mở rộng cho hệ thống.

- **Khả năng kỹ thuật của đội ngũ triển khai:** Máy tính với webcam sẽ dễ dàng cài đặt và sử dụng, ít cần đào tạo thêm đối với đội ngũ đã quen thuộc với Windows và các phần mềm thông dụng. Mặt khác, đối với đội ngũ đã có kinh nghiệm về Linux và lập trình nhúng, Raspberry Pi với webcam sẽ mang lại nhiều lợi ích và khả năng tùy biến cao, tận dụng và phát triển được kỹ năng sẵn có của họ.

4. Thiết kế và triển khai phần mềm

Phần mềm là thành phần quan trọng trong hệ thống giám sát an ninh bằng hình ảnh, giúp thu thập, xử lý và quản lý dữ liệu từ các camera. Trong phần này, chúng tôi sẽ trình bày chi tiết quá trình thiết kế và triển khai phần mềm cho hai phương pháp đã đề cập: sử dụng máy tính cá nhân (PC hoặc laptop) kết hợp với webcam và sử dụng máy tính nhúng Raspberry Pi kết hợp với webcam.

4.1. Mục tiêu thiết kế phần mềm

Mục tiêu chung của việc thiết kế phần mềm giám sát an ninh bằng hình ảnh chính là thu thập dữ liệu hình ảnh từ các camera, thông qua các bước xử lý hình ảnh để phát hiện chuyển động và nhận dạng đối tượng tùy thuộc vào yêu cầu của hệ thống, có thể quản lý và lưu trữ dữ liệu hình ảnh, video, quản lý dung lượng lưu trữ hiệu quả. Đồng thời có thể cung cấp giao diện người dùng thân thiện để thuận tiện cho việc theo dõi, cấu hình và nhận thông báo, cho phép nâng cấp, thêm tính năng mới vào hệ thống khi cần thiết.

4.2. Lựa chọn công nghệ và ngôn ngữ lập trình

Chúng tôi chọn **Python** làm ngôn ngữ lập trình chính sử dụng trong dự án này vì đó là một ngôn ngữ lập trình bậc cao, dễ học, cung cấp nhiều thư viện mạnh mẽ cho xử lý ảnh và phát triển giao diện người dùng.

Ngoài ra có các thư viện và công cụ hỗ trợ như:

- **OpenCV**: Thư viện mã nguồn mở mạnh mẽ dành cho xử lý ảnh và video, hỗ trợ nhiều thuật toán phát hiện và nhận diện đối tượng.
- **Tkinter**: Thư viện tích hợp sẵn trong Python để phát triển giao diện người dùng đồ họa (GUI).
- **PIL (Pillow)**: Thư viện xử lý ảnh, hỗ trợ chuyển đổi và hiển thị ảnh trong giao diện Tkinter.
- **Ultralytics YOLO**: Mô hình nhận diện đối tượng tiên tiến, cung cấp khả năng phát hiện và theo dõi nhiều loại đối tượng trong thời gian thực.
- **NumPy**: Thư viện hỗ trợ tính toán số học, cần thiết cho xử lý dữ liệu ảnh.
- **SciPy**: Cung cấp các hàm toán học và thuật toán nâng cao, bao gồm tính toán khoảng cách cho việc theo dõi đối tượng.

4.3. Thiết kế và triển khai phần mềm cho phương pháp 1 (máy tính với webcam – triển khai chính)

4.3.1. Phần mềm và công cụ sử dụng

- Hệ điều hành: Windows 10/11 hoặc Ubuntu Linux.
- Ngôn ngữ lập trình: Python 3.8 trở lên.
- Thư viện: OpenCV, Tkinter, PIL, Ultralytics YOLO, NumPy, SciPy.

4.3.2. Cài đặt và cấu hình phần mềm

- Bước 1: Cài đặt môi trường python và các thư viện cần thiết bằng pip: sử dụng Command Prompt (Windows) hoặc Terminal (Linux).
- Bước 2: Cài đặt mô hình YOLO: tải mô hình YOLO (ở đây chúng tôi dùng phiên bản yolov8n.pt) từ Ultralytics và lưu vào thư mục dự án.
- Bước 3: Sử dụng các thư viện đã cài đặt và mô hình YOLO để phát triển mã nguồn python cho ứng dụng giám sát tùy chỉnh. Trong đó:

+ Khởi tạo các thư viện và lớp CentroidTracker: lớp giúp theo dõi các đối tượng chuyển động dựa trên trọng tâm (centroid) của bounding boxes phát hiện được. Các phương thức chính bao gồm: khởi tạo các biến cần thiết (init()), đăng ký một đối tượng mới với centroid của nó (register()), xóa một đối tượng khỏi hệ thống theo dõi (deregister()), cập nhật danh sách các đối tượng đang theo dõi dựa trên các bounding boxes phát hiện được trong khung hình hiện tại (update()).

+ Khởi tạo giao diện và các thành phần UI.

+ Khởi tạo videoCapture và các biến quan trọng trong hệ thống.

+ Định nghĩa hàm xử lý sự kiện nhấn phím để dừng chương trình.

+ Định nghĩa hàm cập nhật khung hình và xử lý chuyển động (bao gồm thuật toán phát hiện chuyển động qua khung hình và phát hiện đối tượng bằng YOLO).

Chi tiết hoạt động của ứng dụng giám sát an ninh tùy chỉnh:

- Đọc khung hình từ webcam, kiểm tra nếu không đọc được khung hình thì dừng cập nhật.

- Xử lý chuyển động: áp dụng bộ trừ nền để tạo mặt nạ chuyển động, sau đó sử dụng ngưỡng để phân tách chuyển động và làm sạch ảnh bằng các phép biến đổi hình thái học như mở và dẫn, cuối cùng tìm các contour để xác định các vùng chuyển động và vẽ bounding boxes đỏ quanh chúng.

- Cập nhật đối tượng chuyển động: sử dụng CentroidTracker để theo dõi các đối tượng chuyển động, nếu phát hiện đối tượng mới, thêm thông báo vào giao diện ứng dụng và hiển thị.

- Nhận diện và theo dõi đối tượng với YOLO: sử dụng mô hình YOLO để phát hiện và vẽ bounding boxes cho các đối tượng trong khung hình, nếu phát hiện đối tượng mới, thêm thông báo vào giao diện ứng dụng và hiển thị.

- Cập nhật giao diện người dùng: chuyển đổi khung hình từ BGR sang RGB để hiển thị trong Tkinter, cập nhật thời gian hiện tại trên giao diện và lặp lại quá trình sau mỗi 10ms để tạo ra luồng video liên tục và thời gian thực.

4.4. Thiết kế và triển khai phần mềm cho phương pháp 2 (Raspberry Pi với webcam – tham khảo)

4.4.1. Phần mềm và công cụ sử dụng

- Hệ điều hành: Raspberry Pi OS (32-bit hoặc 64-bit).
- Ngôn ngữ lập trình: Python 3.7 trở lên.
- Thư viện: OpenCV, Tkinter, PIL, Ultralytics YOLO, NumPy, SciPy.
- Các công cụ hỗ trợ: Git để quản lý mã nguồn, SSH để truy cập từ xa nếu cần.

4.4.2. Cài đặt và cấu hình phần mềm

- Bước 1: Cài đặt Python, pip và các thư viện cần thiết bằng pip.
- Bước 2: Cài đặt mô hình YOLO (tải mô hình yolov8n.pt từ Ultralytics và lưu vào thư mục dự án trên Raspberry Pi).
- Bước 3: Triển khai ứng dụng giám sát tùy chỉnh: ứng dụng giám sát trên Raspberry Pi được triển khai tương tự như trên máy tính cá nhân. Tuy nhiên, có một số điều chỉnh nhỏ để tối ưu hóa hiệu năng trên thiết bị nhúng như:
 - + Tối ưu hóa kích thước khung hình bằng cách giảm kích thước.
 - + Giảm độ phân giải và tốc độ khung hình nếu cần để đảm bảo hoạt động mượt mà.
 - + Để ứng dụng tự động chạy khi Raspberry Pi bật, tạo một dịch vụ systemd.

+ Cấu hình mạng và truy cập từ xa (sử dụng SSH để truy cập Raspberry Pi từ máy tính khác nếu cần, cài đặt tường lửa để bảo mật hệ thống).

4.5. Phân tích và đánh giá phần mềm của hai phương pháp

Hai phương pháp *Sử dụng máy tính kết hợp với webcam* và *Sử dụng thiết bị nhúng Raspberry Pi kết hợp với webcam* trong thiết kế hệ thống giám sát an ninh tự động bằng hình ảnh đều có những đặc điểm riêng, phù hợp với các loại yêu cầu và tiêu chí khác nhau tùy thuộc vào bài toán và dự án thực tế. Một số tiêu chí nổi bật được nêu trong bảng sau:

Tiêu chí	Đánh giá		
		Máy tính với webcam	Raspberry với webcam
Khả năng phát triển và tùy biến	Ưu điểm	Dễ dàng sử dụng các phần mềm thương mại hoặc mã nguồn mở với giao diện thân thiện. Có thể phát triển ứng dụng tùy chỉnh với nhiều lựa chọn ngôn ngữ và công cụ. Ngoài ra còn nhận được hỗ trợ mạnh mẽ từ cộng đồng và nguồn tài liệu phong phú.	Khả năng tùy biến cao, nhưng cần kỹ năng kỹ thuật chuyên môn. Thích hợp cho việc học tập và nghiên cứu về lập trình nhúng và xử lý ảnh.
	Nhược điểm	Đòi hỏi phần mềm có thể tiêu thụ nhiều tài nguyên hệ thống, đặc biệt là khi xử lý nhiều camera cùng lúc.	Yêu cầu kiến thức về Linux và lập trình nhúng. Có giới hạn về tài nguyên phần cứng, ảnh hưởng đến khả năng mở rộng hệ thống.
Hiệu năng và khả năng xử lý	Ưu điểm	Hiệu năng cao, xử lý các tác vụ phức tạp một cách mượt mà, hỗ trợ đa nhiệm tốt, có thể chạy nhiều ứng dụng cùng lúc.	Tiêu thụ ít tài nguyên và năng lượng, đáp ứng tốt cho các tác vụ đơn giản như phát hiện chuyển

			động và nhận diện đối tượng cơ bản.
	Nhược điểm	Tiêu thụ nhiều tài nguyên và điện năng.	Hiệu năng hạn chế, không phù hợp cho các tác vụ xử lý hình ảnh phức tạp hoặc nhiều camera cùng lúc.
Chi phí và tài nguyên	Ưu điểm	Có thể sử dụng phần mềm miễn phí hoặc thương mại tùy vào nhu cầu.	Chi phí đầu tư ban đầu thấp hơn, phù hợp với ngân sách hạn chế. Sử dụng phần mềm mã nguồn mở có thể giảm được chi phí phần mềm.
	Nhược điểm	Chi phí đầu tư ban đầu cao hơn do cần phần cứng mạnh mẽ và có thể phải mua phần mềm thương mại. Tiêu thụ nhiều điện năng dẫn đến tăng chi phí vận hành.	Giới hạn về khả năng mở rộng và hiệu năng, đòi hỏi nhiều công sức để tối ưu hóa và bảo trì hệ thống.

Bảng 7: Đánh giá phần mềm hệ thống khi triển khai bằng hai phương pháp máy tính-webcam và Raspberry-webcam

Quá trình thiết kế và triển khai phần mềm cho hệ thống giám sát an ninh bằng hình ảnh đã cho thấy sự linh hoạt và hiệu quả của việc sử dụng các công nghệ hiện đại như Python, OpenCV, và YOLO. Hai phương pháp sử dụng máy tính cá nhân với webcam và Raspberry Pi với webcam đều có những ưu điểm và nhược điểm riêng, phù hợp với các nhu cầu và điều kiện triển khai khác nhau, trong đó:

Phương pháp 1 (Máy tính với Webcam): Thích hợp cho các hệ thống yêu cầu hiệu năng cao, khả năng xử lý phức tạp, và khả năng mở rộng lớn. Dù chi phí đầu tư ban đầu và chi phí vận hành cao hơn, phương pháp này đảm bảo độ ổn định và hiệu quả trong việc giám sát an ninh.

Phương pháp 2 (Raspberry Pi với Webcam): Phù hợp cho các ứng dụng nhỏ, di động, tiết kiệm chi phí và năng lượng. Mặc dù có giới hạn về hiệu năng và khả năng mở rộng, Raspberry Pi cung cấp một giải pháp linh hoạt và tiết kiệm cho các dự án giám sát cơ bản hoặc trong môi trường hạn chế về không gian và nguồn lực.

Trong dự án của chúng tôi, sau khi cân nhắc kỹ lưỡng các yếu tố về hiệu năng, chi phí, tính linh hoạt và khả năng kỹ thuật của đội ngũ, chúng tôi đã quyết định lựa chọn phương pháp sử dụng máy tính cá nhân kết hợp với webcam cho hệ thống giám sát an ninh. Lựa chọn này phù hợp với các yêu cầu đòi hỏi hiệu năng cao và khả năng xử lý phức tạp, đảm bảo độ ổn định và hiệu quả trong việc giám sát. Mặc dù chi phí đầu tư ban đầu và chi phí vận hành có thể cao hơn so với các phương pháp khác, nhưng sự mạnh mẽ và khả năng mở rộng lớn của máy tính cá nhân giúp chúng tôi đáp ứng tốt các nhu cầu giám sát an ninh nghiêm ngặt. Hơn nữa, hệ thống này dễ dàng mở rộng và nâng cấp trong tương lai, cho phép chúng tôi linh hoạt điều chỉnh và phát triển hệ thống theo nhu cầu thực tế.

5. Kết luận

Trong quá trình thực hiện đề tài “Thiết kế hệ thống giám sát an ninh tự động bằng hình ảnh”, chúng tôi đã đạt được những kết quả đáng khích lệ thông qua việc áp dụng các công nghệ hiện đại như Python, OpenCV, và YOLO. Hệ thống giám sát an ninh được thiết kế và triển khai trên nền tảng máy tính cá nhân kết hợp với webcam đã chứng minh được tính linh hoạt và hiệu quả trong việc phát hiện và theo dõi các đối tượng chuyển động, đồng thời hiện lên thông báo khi phát hiện ra một đối tượng hoặc chuyển động mới ngay trong thời gian thực.

5.1. Kết quả đã đạt được

Mục tiêu chính của đề tài là thiết kế một hệ thống giám sát an ninh tự động có khả năng phát hiện chuyển động, nhận diện vật thể, và theo dõi các đối tượng trong thời gian thực. Thông qua việc sử dụng mô hình YOLO để nhận diện vật thể và lớp CentroidTracker để theo dõi các đối tượng chuyển động, chúng tôi đã xây dựng được một hệ thống có khả năng xử lý hình ảnh nhanh chóng và chính xác. Giao diện người dùng được xây dựng bằng Tkinter cho phép hiển thị video trực tiếp, thời gian thực hiện giám sát và các thông báo về các sự kiện phát hiện, giúp người dùng dễ dàng theo dõi và quản lý hệ thống.

Khi kết hợp thuật toán nhận diện và theo dõi vật thể của YOLO với thuật toán phát hiện chuyển động CentroidTracker, hệ thống của chúng tôi đã có hiệu quả giám sát cao hơn trong việc nhận biết nhanh chóng và chính xác khi có sự xuất hiện của người hoặc vật thể quan trọng trong khung hình, các hành động cụ thể như hiện thông báo, ghi hình và giảm nhiễu cũng được tối ưu nhằm tăng độ tin cậy và chính xác cho hệ thống.

Sau khi phân tích và so sánh giữa hai phương pháp triển khai là sử dụng máy tính cá nhân với webcam và Raspberry Pi với webcam, chúng tôi đã quyết định chọn phương pháp sử dụng máy tính cá nhân kết hợp với webcam. Lựa chọn này dựa trên các yếu tố chính như hiệu năng, khả năng mở rộng và độ ổn định.

- Về hiệu năng: máy tính cá nhân cung cấp khả năng xử lý mạnh mẽ, phù hợp với các yêu cầu về xử lý hình ảnh phức tạp và chạy các mô hình nhận diện như YOLO một cách mượt mà.

- Về khả năng mở rộng: hệ thống có thể dễ dàng được nâng cấp về phần cứng hoặc phần mềm để đáp ứng nhu cầu giám sát ngày càng tăng mà không gặp nhiều hạn chế.
- Về độ ổn định và tin cậy: máy tính cá nhân thường có độ ổn định cao, giảm thiểu rủi ro gián đoạn trong quá trình giám sát, đảm bảo hệ thống hoạt động liên tục và hiệu quả.

5.2. Những thách thức và giải pháp

Mặc dù hệ thống đã đạt được các mục tiêu đề ra là phát hiện và cảnh báo khi có chuyển động, nhưng vẫn tồn tại một số hạn chế cần được khắc phục trong tương lai:

- Việc sử dụng máy tính cá nhân yêu cầu nguồn điện ổn định và không phù hợp với các ứng dụng cần di động hoặc lắp đặt tại những vị trí hạn chế về không gian, gây nên các giới hạn về phần cứng.
- Hệ thống có thể gặp khó khăn trong việc phân biệt các đối tượng khi môi trường giám sát có nhiều yếu tố gây nhiễu như ánh sáng thay đổi liên tục hoặc nhiều đối tượng di chuyển cùng lúc, khiến cho độ chính xác khi phát hiện chuyển động bị giảm đi trong các môi trường phức tạp.

Với những thách thức này, chúng tôi đã đưa ra và triển khai một số giải pháp khắc phục như tối ưu hóa mã nguồn, sử dụng các kỹ thuật xử lý song song khi có thể để đảm bảo hệ thống có thể xử lý và phản hồi nhanh chóng các sự kiện phát hiện, tinh chỉnh các tham số của mô hình YOLO và sử dụng các bộ dữ liệu đa dạng để huấn luyện mô hình để nâng cao độ chính xác trong việc nhận diện vật thể.

5.3. Định hướng phát triển trong tương lai

Để nâng cao hiệu quả và mở rộng khả năng của hệ thống giám sát an ninh tự động, chúng tôi đề xuất một số hướng phát triển trong tương lai như sau:

- Tích hợp các công nghệ AI tiên tiến hơn: áp dụng các mô hình học sâu mới hơn để cải thiện độ chính xác và tốc độ xử lý của hệ thống.
- Tối ưu hóa sử dụng tài nguyên: nghiên cứu và triển khai các phương pháp tối ưu hóa mã nguồn để giảm tải cho hệ thống, cho phép triển khai trên các nền tảng phần cứng nhẹ hơn mà vẫn đảm bảo hiệu năng.

- Mở rộng chức năng: thêm các tính năng như nhận diện khuôn mặt, phân tích hành vi, và cảnh báo tự động qua email hoặc SMS để tăng cường khả năng giám sát và phản hồi nhanh chóng.

Tóm lại, qua quá trình nghiên cứu và triển khai, chúng tôi đã thành công trong việc thiết kế và xây dựng một hệ thống giám sát an ninh tự động bằng hình ảnh với khả năng phát hiện và theo dõi đối tượng hiệu quả. Sự lựa chọn phương pháp sử dụng máy tính cá nhân kết hợp với webcam đã chứng minh được tính khả thi và hiệu quả, đáp ứng tốt các yêu cầu của dự án. Tuy còn một số hạn chế, nhưng với những cải tiến và phát triển trong tương lai, hệ thống có tiềm năng ứng dụng rộng rãi trong các lĩnh vực an ninh và giám sát hiện đại.