



TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO
PBL5 – DỰ ÁN KỸ THUẬT MÁY TÍNH

TÊN ĐỀ TÀI
HỆ THỐNG HỖ TRỢ NGƯỜI KHIẾM THỊ

Giảng viên hướng dẫn: Ts. Bùi Thị Thanh Thanh

NHÓM SINH VIÊN THỰC HIỆN	LỚP HỌC PHẦN
Đặng Phúc Long	22Nh12.B
Thái Viết Quốc Hưng	22Nh12.B
Phạm Nhật Thành	22Nh12.B

Đà Nẵng, 06/2025

TÓM TẮT ĐỒ ÁN

Người khiếm thị thường xuyên đối mặt với những thách thức đáng kể trong cuộc sống hàng ngày, từ việc di chuyển an toàn đến khả năng định vị và nhận diện vật cản. Để giải quyết những khó khăn này và mang lại sự độc lập cao hơn, chúng tôi đã phát triển một **Hệ thống hỗ trợ người khiếm thị**. Hệ thống tích hợp hai thiết bị bao gồm gậy thông minh và thắt lưng cảnh báo té ngã, được thiết kế để nâng cao khả năng di chuyển và đảm bảo an toàn tối đa cho người dùng.

Gậy thông minh không chỉ là một dụng cụ hỗ trợ di chuyển thông thường, mà còn là một giải pháp công nghệ tiên tiến, mang lại sự an toàn và tự tin cho người khiếm thị. Phát hiện và nhận diện chướng ngại vật nhờ tích hợp trí tuệ nhân tạo (AI) hiện đại, gậy có khả năng nhận biết chính xác các vật cản phổ biến trên đường đi, giúp người dùng tránh va chạm và di chuyển mượt mà hơn. Cảnh báo âm thanh tức thì ngay khi phát hiện chướng ngại vật, giúp người dùng phản ứng nhanh chóng, mang đến sự độc lập và an tâm cho người dùng.

Thắt lưng cảnh báo té ngã tích hợp định vị là một phần quan trọng của hệ thống, giúp bảo vệ người khiếm thị khỏi những tai nạn tiềm ẩn và giữ kết nối với người thân. Kết hợp cảm biến gia tốc và công nghệ học máy (Machine Learning), thắt lưng có thể phân tích chuyển động cơ thể để nhận diện các tư thế mất thăng bằng hoặc thay đổi đột ngột, từ đó phát hiện sớm nguy cơ té ngã và gửi một cảnh báo khẩn cấp sẽ đến ứng dụng trên điện thoại của người thân, giúp họ phản ứng kịp thời. Bên cạnh đó, module GPS tích hợp cho phép xác định vị trí chính xác của người dùng, hiển thị trực quan trên bản đồ ứng dụng. Điều này không chỉ mang lại sự an tâm cho người khiếm thị mà còn giúp gia đình dễ dàng theo dõi và hỗ trợ khẩn cấp khi cần thiết.

Hệ thống hỗ trợ người khiếm thị của chúng tôi đã trải qua quá trình thử nghiệm và đạt được những kết quả vượt trội, khẳng định hiệu quả trong việc nâng cao sự an toàn và độc lập cho người dùng. Khả năng phát hiện và nhận diện vật cản đạt độ chính xác cao, trong khi tính năng cảnh báo té ngã hoạt động kịp thời, giúp người thân có thể can thiệp ngay lập tức. Đặc biệt, chức năng định vị vị trí hoạt động liên tục và chính xác, cung cấp thông tin vị trí rõ ràng trên ứng dụng điện thoại. Ứng dụng này được thiết kế với giao diện thân thiện, trực quan và dễ sử dụng, giúp người thân dễ dàng theo dõi và hỗ trợ, mang lại sự an tâm tối đa cho cả người khiếm thị và gia đình họ.

Nhóm xin chân thành cảm ơn cô TS. Bùi Thị Thanh Thanh đã nhiệt tình hướng dẫn, gợi ý các giải pháp để nhóm có thể hoàn thành dự án này.

BẢNG PHÂN CÔNG NHIỆM VỤ

Sinh viên thực hiện	Các nhiệm vụ	Tự đánh giá
Đặng Phúc Long	<ul style="list-style-type: none">- Tìm hiểu, triển khai mô hình YOLO cho nhận diện vật cản.- Tìm hiểu, triển khai mô hình SVM cho nhận diện té ngã.- Tìm hiểu, triển khai deploy mô hình SVM vào ESP32.- Viết Server AI nhận diện vật cản.- Lắp mạch.	Đã hoàn thành tất cả các nhiệm vụ
Thái Việt Quốc Hưng	<ul style="list-style-type: none">- Viết ứng dụng Mobile.- Thu thập dữ liệu.- Đánh nhãn dữ liệu.- Lắp mạch.	Đã hoàn thành tất cả các nhiệm vụ
Phạm Nhật Thành	<ul style="list-style-type: none">- Viết code Arduino.- Thu thập dữ liệu.- Đánh nhãn dữ liệu.- Lắp mạch.	Đã hoàn thành tất cả các nhiệm vụ

MỤC LỤC

I.GIỚI THIỆU	9
1. Thực trạng.....	9
2. Các vấn đề cần giải quyết.....	9
3. Đề xuất giải pháp tổng quan	10
II. GIẢI PHÁP	12
1. Giải pháp phần cứng.....	12
1.1 Module ESP32-CAM	15
1.2 Module ESP32	16
1.3 Module DF Player Mini.....	16
1.4 Module GPS NEO- 6M	16
1.5 Cảm biến gia tốc GY-521 6DOF IMU MPU6050	17
2. Giải pháp truyền thông	18
2.1 Giao thức UART	18
2.2 Giao thức I2C	19
2.3. Giao thức HTTP	20
2.4. Giao thức Websocket.....	21
3. Giải pháp AI/KHDL	22
3.1. Giải pháp nhận diện vật cản	22
3.2 Giải pháp nhận diện té ngã	28
4. Giải pháp phần mềm.....	33
4.1. Phát biểu bài toán	33
4.2. Công nghệ sử dụng	33
4.3. Giải pháp sử dụng Firebase	33
4.4. Sơ đồ hoạt động các chức năng của ứng dụng	36
4.5. Sơ đồ tuần tự.....	38
III. KẾT QUẢ.....	40
1. Mô hình nhận diện vật cản YOLO11	40
1.1 Kết quả thu thập dữ liệu	40

1.2 Tiến hành huấn luyện mô hình	41
1.3 Đánh giá mô hình	42
2. Mô hình phát hiện té ngã SVM	45
2.1 Kết quả thu thập dữ liệu	45
2.4 Tiến hành huấn luyện mô hình	46
2.3 Đánh giá mô hình	47
3. Ứng dụng di động	48
3.1 Giao diện chức năng đăng nhập, đăng ký	48
3.2 Giao diện trang chủ và cài đặt	49
3.3 Giao diện thêm thiết bị và hiển thị vị trí của thiết bị đó	50
3.4 Giao diện xem lịch sử vị trí của thiết bị	51
3.5 Giao diện thông tin cá nhân và đổi mật khẩu	52
4. Kết quả mô hình phần cứng	53
IV. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	55
4.1 Kết luận	55
4.2 Hướng phát triển	55
V. TÀI LIỆU THAM KHẢO	56

DANH MỤC HÌNH ẢNH

Hình 1: Sơ đồ tổng quan gậy thông minh.....	10
Hình 2: Sơ đồ tổng quan thắt lưng cảnh báo té ngã.....	11
Hình 3: Sơ đồ lắp mạch gậy thông minh	12
Hình 4: Sơ đồ lắp mạch thắt lưng cảnh báo té ngã.....	13
Hình 5: Sơ đồ khái quy trình hoạt động của module DF Player Mini	16
Hình 6: Sơ đồ khái quy trình hoạt động của module GPS NEO 6M	17
Hình 7: Sơ đồ khái quy trình hoạt động của cảm biến MPU6050	18
Hình 8: Minh họa giao thức UART	18
Hình 9: Giao tiếp UART giữa ESP32-CAM với DF Player Mini và ESP32 với GPS Neo6M19	
Hình 10: Minh họa giao thức I2C.....	19
Hình 11: Giao tiếp I2C giữa ESP32 với MPU6050	20
Hình 12: Minh họa giao thức truyền thông HTTP	20
Hình 13: Minh họa quá trình gửi tọa độ và cảnh báo té ngã từ ESP đến server GPS	21
Hình 14: Minh họa giao thức truyền thông Websocket.....	21
Hình 15: Minh họa quá trình gửi ảnh realtime từ ESP32-Cam đến Server AI.....	22
Hình 16: Sơ đồ giải pháp nhận diện vật cản.....	22
Hình 17: Sơ đồ quy trình xây dựng mô hình nhận diện vật cản	23
Hình 18: Minh họa cách mô hình YOLO hoạt động	24
Hình 19: Đầu ra của mô hình YOLO	24
Hình 20: Quá trình hoạt động cơ bản của Greedy NMS (loại phỏ biến nhất).....	25
Hình 21: Hàm loss function của mô hình YOLO	26
Hình 22: Sơ đồ giải pháp nhận diện té ngã.....	28
Hình 23: Sơ đồ quy trình xây dựng mô hình nhận diện té ngã.....	29
Hình 24: Dịch vụ hỗ trợ nhiều phương thức đăng nhập	34
Hình 25: Mô hình hoạt động của Firebase Realtime Database	35
Hình 26: Sơ đồ hoạt động chức năng đăng nhập.....	36
Hình 27: Sơ đồ hoạt động chức năng hiển thị vị trí thiết bị	36
Hình 28: Sơ đồ hoạt động chức năng xem lịch sử vị trí của thiết bị	37
Hình 29: Sơ đồ tuần tự chức năng đăng nhập	38
Hình 30: Sơ đồ tuần tự chức năng xem vị trí của thiết bị.....	39

Báo cáo đồ án PBL5 – Dự án kỹ thuật máy tính

Hình 31: Sơ đồ tuân tự chức năng xem lịch sử vị trí thiết bị.....	39
Hình 32: Biểu đồ thống kê số lượng instances của mỗi class	40
Hình 33: Ma nhận nhầm lẩn chuẩn hóa mô hình YOLO	42
Hình 34: Ma trận nhầm lẩn mô hình YOLO	43
Hình 35: Biểu đồ đường cong đánh giá độ tự tin của mô hình YOLO	44
Hình 36: Kết quả dự đoán trên tập validation	44
Hình 37: Biểu đồ thống kê dữ liệu Té ngã	45
Hình 38: Ma trận nhầm lẩn mô hình SVM trong nhận diện té ngã	47
Hình 39: Giao diện chức năng đăng nhập, đăng ký.....	48
Hình 40: Giao diện trang chủ và cài đặt	49
Hình 41: Giao diện thêm thiết bị và hiển thị vị trí của thiết bị đó	50
Hình 42: Giao diện xem lịch sử vị trí của thiết bị	51
Hình 43: Giao diện thông tin cá nhân và đổi mật khẩu	52
Hình 44: Mô hình phần cứng gậy thông minh.....	53
Hình 45: Mô hình phần cứng Thắt lưng cảnh báo té ngã	54
Hình 46: Mô hình Hệ thống hỗ trợ người khiếm thị thực tế	54

DANH MỤC BẢNG BIỂU

Bảng 1: Bảng đề xuất giải pháp.....	12
Bảng 2: Bảng danh sách các linh kiện sử dụng	15
Bảng 3: Bảng danh sách tham số huấn luyện	41

I.GIỚI THIỆU

Người khiếm thị thường xuyên đối mặt với những thách thức đáng kể trong cuộc sống hàng ngày, từ việc di chuyển an toàn đến khả năng định vị và nhận diện vật cản. Để giải quyết những khó khăn này và mang lại sự độc lập cao hơn, chúng tôi đã phát triển một Hệ thống hỗ trợ người khiếm thị toàn diện. Hệ thống tích hợp hai thiết bị thông minh bao gồm gậy thông minh và thắt lưng cảnh báo té ngã, được thiết kế để nâng cao khả năng di chuyển và đảm bảo an toàn tối đa cho người dùng.

1. Thực trạng

Hiện tại, trên thị trường trong nước và quốc tế đã có một số sản phẩm hỗ trợ người khiếm thị. Một số sản phẩm nổi bật như **SmartCane**, **UltraCane**, **WeWALK** đều cho thấy mức độ hoàn thiện cao và tích hợp nhiều công nghệ tiên tiến, mang lại hiệu quả sử dụng đáng kể trong việc phát hiện vật cản.

Tuy nhiên, điểm chung của các giải pháp hiện tại là chúng thường chỉ tập trung vào một khía cạnh cụ thể, chưa cung cấp một hệ thống hỗ trợ toàn diện. Ví dụ, nhiều loại gậy thông minh phổ biến chỉ dừng lại ở việc phát hiện vật cản cơ bản thông qua cảm biến siêu âm hoặc hồng ngoại. Khả năng nhận diện chi tiết loại vật cản hoặc cảnh báo té ngã do mất thăng bằng vẫn còn hạn chế. Các ứng dụng định vị trên điện thoại tuy hữu ích nhưng lại đòi hỏi người dùng phải thao tác liên tục và có thể không hiệu quả trong các tình huống cần phản ứng nhanh. Hơn nữa, giá thành đắt đỏ của nhiều sản phẩm công nghệ cao trên thị trường quốc tế có thể là rào cản lớn đối với người khiếm thị, vốn thường có điều kiện kinh tế phụ thuộc vào gia đình. Bên cạnh đó, việc thiếu một hệ thống kết nối liền mạch để người thân có thể theo dõi và hỗ trợ kịp thời cũng là một điểm yếu chung của các giải pháp hiện tại.

2. Các vấn đề cần giải quyết

Từ thực trạng trên, chúng tôi nhận thấy có ba vấn đề chính cần được giải quyết để nâng cao chất lượng cuộc sống và sự an toàn cho người khiếm thị:

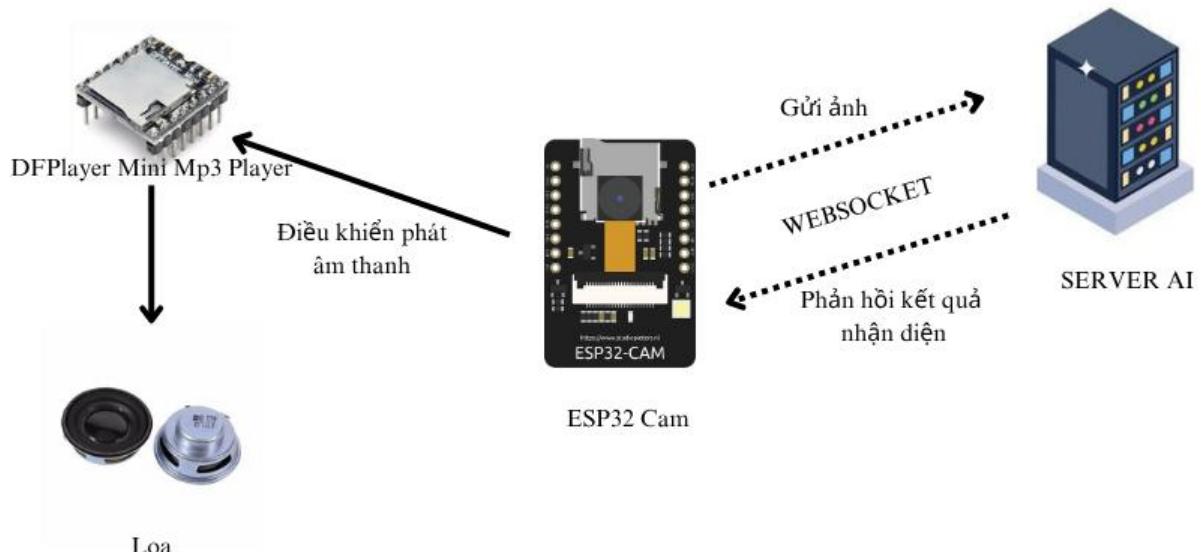
- **Hạn chế trong phát hiện và nhận diện vật cản phức tạp:** Các thiết bị hiện có thường chỉ cảnh báo sự hiện diện của vật cản mà chưa thể nhận diện chi tiết, dẫn đến việc người dùng khó đưa ra quyết định di chuyển tối ưu.
- **Thiếu khả năng cảnh báo té ngã và mất thăng bằng tức thì:** Nguy cơ té ngã là mối lo ngại lớn đối với người khiếm thị. Các giải pháp hiện tại ít khi tích hợp tính năng phát hiện và cảnh báo té ngã một cách chủ động và kịp thời.

- **Khó khăn trong việc định vị và hỗ trợ khẩn cấp:** Trong trường hợp khẩn cấp, việc người thân hoặc đội ngũ hỗ trợ nhanh chóng xác định vị trí của người khiếm thị là vô cùng quan trọng.

3. Đề xuất giải pháp tổng quan

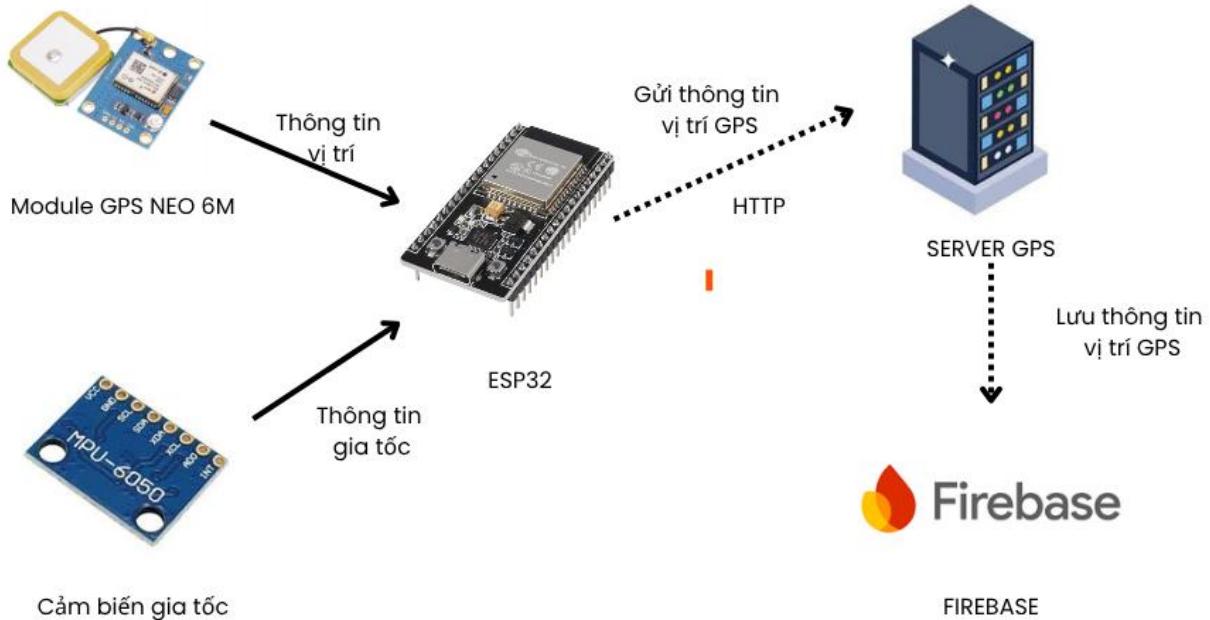
Để giải quyết những vấn đề trên, chúng tôi đề xuất phát triển một **Hệ thống hỗ trợ người khiếm thị** dựa trên sự kết hợp của công nghệ **Trí tuệ nhân tạo (AI)** và **Học máy (Machine Learning)**. Hệ thống sẽ bao gồm hai thiết bị thông minh chính, hoạt động phối hợp để đảm bảo an toàn tối đa và tăng cường sự độc lập cho người khiếm thị:

Gậy thông minh tích hợp AI: Thiết bị này sẽ được trang bị các thiết bị tiên tiến và sử dụng AI để phát hiện, nhận diện chi tiết các loại chướng ngại vật (ví dụ: bậc thang, cửa, ghế, cây cối, người, ...) và cảnh báo bằng âm thanh tức thì. Điều này giúp người dùng chủ động tránh va chạm và di chuyển an toàn hơn.



Hình 1: Sơ đồ tổng quan gậy thông minh

Thắt lưng cảnh báo té ngã tích hợp định vị: Chiếc thắt lưng này sẽ kết hợp cảm biến gia tốc và công nghệ học máy để phân tích chuyển động cơ thể, từ đó phát hiện sớm các dấu hiệu mất thăng bằng hoặc nguy cơ té ngã. Khi phát hiện nguy hiểm, thắt lưng sẽ tự động gửi cảnh báo khẩn cấp đến ứng dụng điện thoại của người thân. Ngoài ra, module GPS tích hợp sẽ liên tục cập nhật và hiển thị vị trí chính xác của người dùng trên bản đồ, cho phép người thân theo dõi và hỗ trợ kịp thời trong mọi tình huống khẩn cấp.



Hình 2: Sơ đồ tổng quan thắt lưng cảnh báo té ngã

Vấn đề	Giải pháp đề xuất
Phần cứng	ESP32 Cam ESP32 Module GPS NEO 6M Cảm biến gia tốc GY-521 6DOF IMU MPU6050 Mạch phát âm thanh DFPlayer Mini Loa 3W
Mô hình nhận diện vật cản	Sử dụng mô hình YOLO11
Mô hình nhận diện té ngã	Sử dụng mô hình SVM (Support Vector Machine)
Ứng dụng điện thoại	Sử dụng Native Android với Java
Cơ sở dữ liệu lưu tọa độ GPS	Firebase Realtime Database
Server nhận diện vật cản	Sử dụng Framework Flask

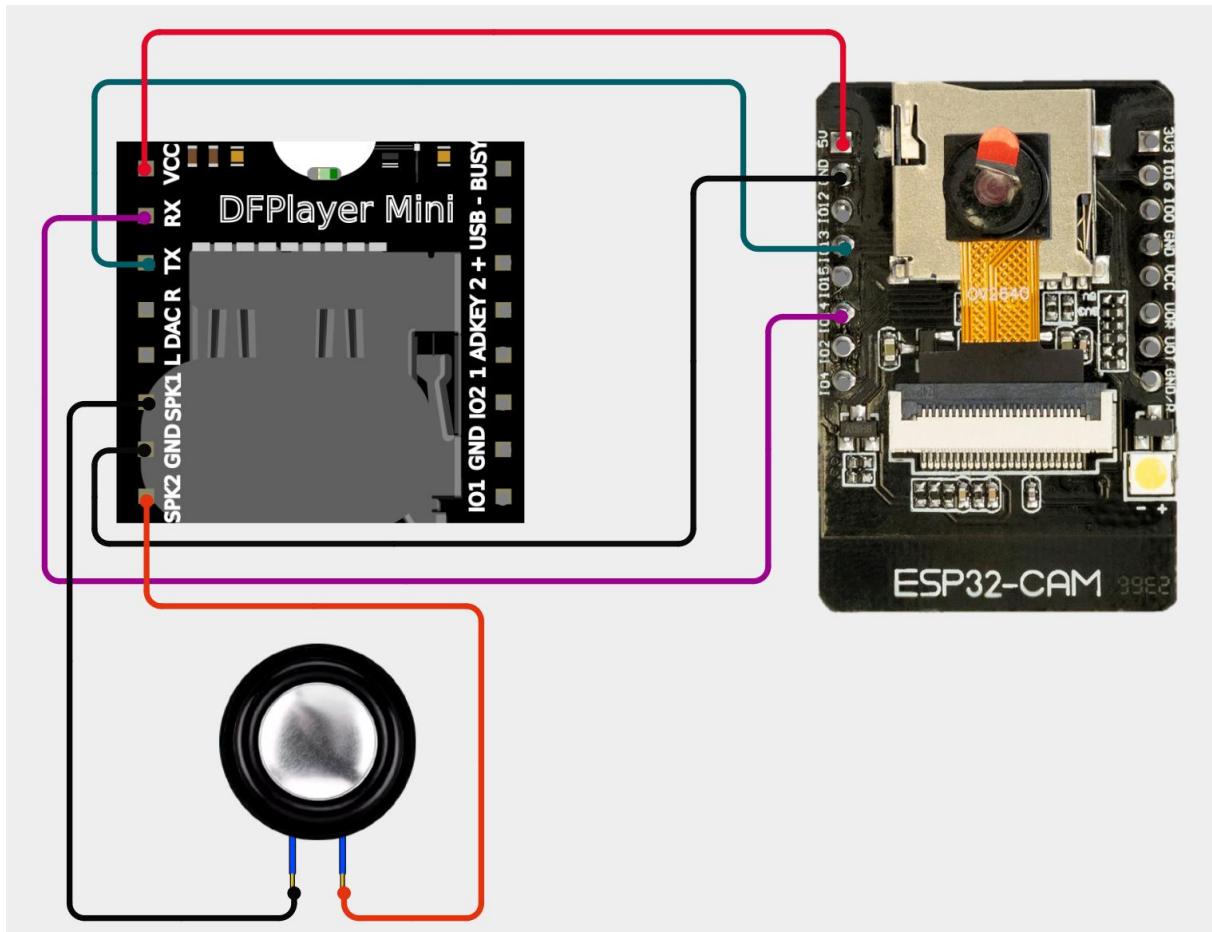
Server xử lý dữ liệu tọa độ GPS

Sử dụng Framework Express Nodejs

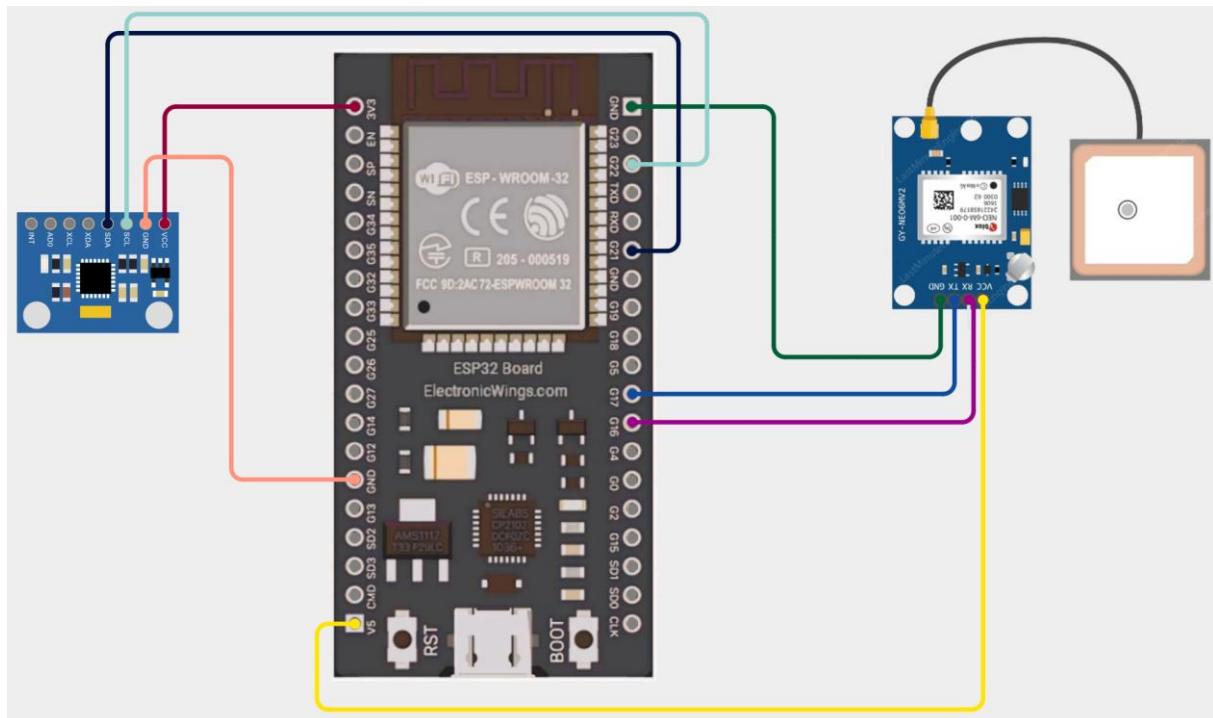
Bảng 1: Bảng đề xuất giải pháp

II. GIẢI PHÁP

1. Giải pháp phần cứng



Hình 3: Sơ đồ lắp mạch gậy thông minh



Hình 4: Sơ đồ lắp mạch thắt lưỡng cảnh báo té ngã

Tên linh kiện	Thông số kỹ thuật	Chi phí
ESP32-CAM	<ul style="list-style-type: none"> - Điện áp cung cấp: 5V - SPI Flash mặc định: 32MB - RAM: 520KB SRAM, 4MB PSRAM - WiFi: 802.11 b/g/n - Interface: UART, SPI, I2C, PWM - Hỗ trợ camera OV2640 - Transmit Power: 802.11b: 17 ± 2 dBm [1] 	155.000
DF Player Mini	<ul style="list-style-type: none"> - Điện áp sử dụng: 3.2~5VDC - Chuẩn giao tiếp: UART - Tốc độ lấy mẫu (Khz) ; 8 / 11.025 / 12 / 16 / 22.05 / 24 / 32 / 44.1 / 48 - Ngõ ra: 24bit [2] 	30.000

Báo cáo đồ án PBL5 – Dự án kỹ thuật máy tính

 <p>Loa 3W</p>	<ul style="list-style-type: none"> - Công suất: 3W Trở kháng: 4Ω - Tần số hoạt động: $0,13 \sim 18(\text{kHz})$ - SNR: $85 \pm 3(\text{dB})$ [3] 	24.000
 <p>ESP32</p>	<ul style="list-style-type: none"> - RAM: 520 KB SRAM - Flash: 4MB – 16MB tùy module - Wi-Fi: 2.4 GHz IEEE 802.11 b/g/n - Bluetooth: Bluetooth v4.2 (Classic + BLE) - GPIO: Lên đến 34 chân đa chức năng - ADC / DAC: 12-bit ADC (18 kênh), 2 kênh DAC - Điện áp hoạt động: 3.0 – 3.3V - Khoảng cách Wi-Fi: Lên đến 100m (môi trường mở) [4] 	125.000
 <p>GPS NEO-6M</p>	<ul style="list-style-type: none"> - Nguồn hoạt động: 3.3-5.5V - Dòng hoạt động bình thường: 50 mA - Dòng hoạt động tiết kiệm: 30 mA - Giao tiếp: UART/TTL - Tốc độ cập nhật: Mặc định 1Hz (tối đa 5Hz) - Độ chính xác vị trí: ~2.5m ngoài trời, có thể tốt hơn với anten chất lượng cao [5] 	125.000
<p>Cảm biến gia tốc GY-521 6DOF IMU MPU6050</p>	<ul style="list-style-type: none"> - Điện áp sử dụng: 3~5VDC - Chuẩn giao tiếp: I2C - Tần số cập nhật dữ liệu: lên tới 1kHz 	48.000

	<ul style="list-style-type: none"> - Dải đo góc quay: $\pm 250, \pm 500, \pm 1000, \pm 2000$ độ/s - Dải đo gia tốc: $\pm 2g, \pm 4g, \pm 8g, \pm 16g$ [6] 	
Anten GPS 1575.42Mhz SMA 	<ul style="list-style-type: none"> - Tần số hoạt động: 1575.42 MHz - Trở kháng: 50Ω - Độ rộng băng thông: ± 10 MHz - Độ khuếch đại: 28 ± 2 dB [7] 	85.000
Tổng		507.000

Bảng 2: Bảng danh sách các linh kiện sử dụng

1.1 Module ESP32-CAM

Module ESP32-CAM là một board phát triển tích hợp dựa trên chip ESP32-S, nổi bật với khả năng xử lý mạnh mẽ và camera OV2640 tích hợp. Module này được thiết kế đặc biệt cho các ứng dụng liên quan đến xử lý hình ảnh, truyền video trực tuyến, và các dự án IoT cần thu thập dữ liệu hình ảnh. ESP32-CAM được trang bị lõi kép 32-bit, hỗ trợ Wi-Fi và Bluetooth, cùng khe cắm thẻ MicroSD để lưu trữ dữ liệu. [8]

Module này có kích thước nhỏ gọn, dễ dàng giao tiếp với các vi điều khiển khác qua các giao thức như UART, SPI, và I2C. Với khả năng xử lý ảnh và video, ESP32-CAM được sử dụng phổ biến trong các ứng dụng như camera giám sát, nhận diện khuôn mặt, hoặc các dự án DIY liên quan đến thị giác máy tính. Chi phí thấp và khả năng lập trình dễ dàng thông qua các nền tảng như Arduino hoặc ESP-IDF khiến ESP32-CAM trở thành lựa chọn lý tưởng cho các nhà phát triển.

1.2 Module ESP32

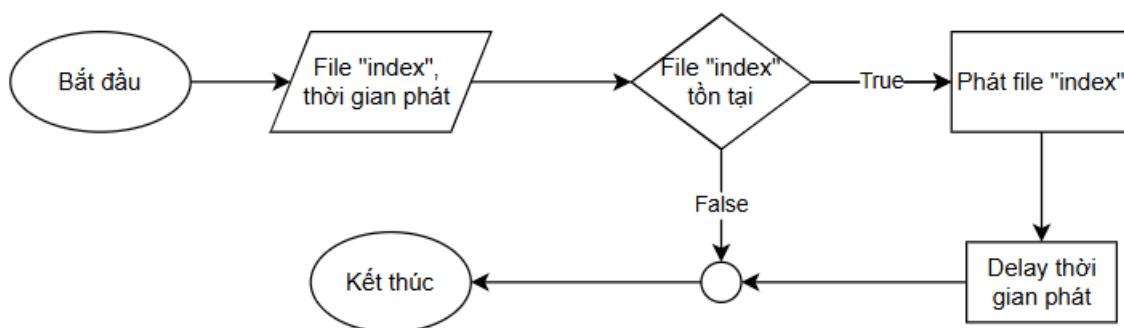
Module ESP32 là một board phát triển mạnh mẽ dựa trên chip ESP32 của Espressif Systems, tích hợp bộ vi xử lý lõi kép 32-bit, hỗ trợ kết nối Wi-Fi và Bluetooth 4.2 (BLE). [9] Với hiệu suất cao, ESP32 phù hợp cho các ứng dụng IoT, từ điều khiển thiết bị thông minh, thu thập dữ liệu cảm biến, đến các hệ thống nhúng phức tạp.

ESP32 cung cấp nhiều giao diện giao tiếp như UART, SPI, I2C, và PWM, cho phép kết nối dễ dàng với các cảm biến, module ngoại vi, và thiết bị khác. Module này có thể được lập trình thông qua các nền tảng phổ biến như Arduino, MicroPython, hoặc ESP-IDF, mang lại sự linh hoạt cho các nhà phát triển. Với chi phí hợp lý và khả năng xử lý đa nhiệm, ESP32 được ứng dụng rộng rãi trong các dự án như nhà thông minh, tự động hóa, và hệ thống giám sát từ xa.

1.3 Module DF Player Mini

DFPlayer Mini là một module phát nhạc MP3 nhỏ gọn, lý tưởng cho các dự án điện tử cần tích hợp âm thanh. Nó có khả năng phát các tệp MP3 lưu trữ trên thẻ nhớ MicroSD (hỗ trợ FAT16, FAT32, tối đa 32GB). Module này đặc biệt dễ sử dụng nhờ giao tiếp qua UART với các vi điều khiển, cho phép điều khiển phát nhạc, chuyển bài, và điều chỉnh âm lượng chỉ bằng các lệnh đơn giản. [10]

Module này được ứng dụng rộng rãi trong hệ thống thông báo, thiết bị IoT và các dự án DIY khác. Với kích thước nhỏ, chi phí thấp, và khả năng kết nối trực tiếp với loa công suất nhỏ, DFPlayer Mini mang đến một giải pháp phát âm thanh hiệu quả và tiện lợi.



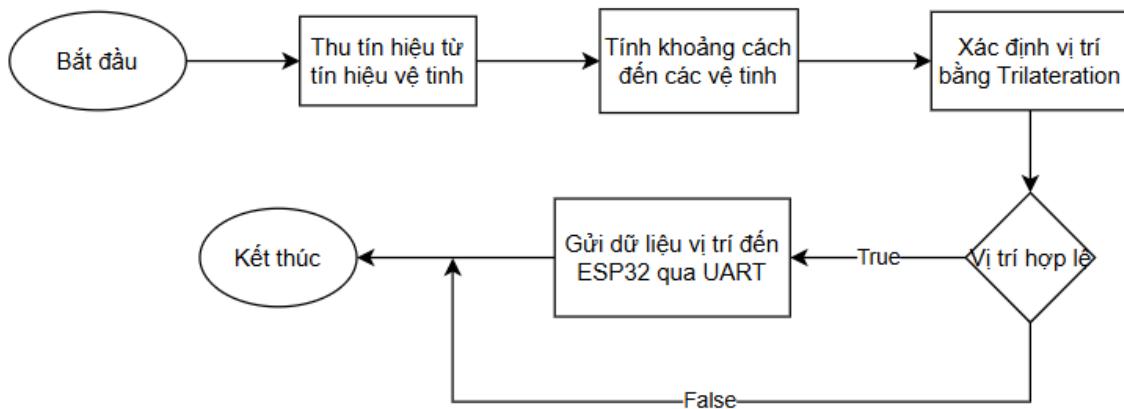
Hình 5: Sơ đồ khái quát quy trình hoạt động của module DF Player Mini

1.4 Module GPS NEO- 6M

Module GPS NEO-6M là một module định vị toàn cầu nhỏ gọn, được sử dụng rộng rãi để xác định vị trí địa lý. Nó hoạt động bằng cách nhận tín hiệu từ các vệ tinh GPS, sau đó tính

toán kinh độ, vĩ độ, độ cao, và thời gian chính xác. Module này thường giao tiếp với vi điều khiển thông qua giao thức UART (nối tiếp), giúp việc tích hợp vào các dự án nhúng trở nên dễ dàng.

Với độ chính xác tốt và chi phí phải chăng, NEO-6M rất phù hợp cho nhiều ứng dụng như theo dõi và xác định vị trí. Nó thường xuất ra dữ liệu ở định dạng NMEA, một chuẩn phổ biến dễ dàng phân tích bằng các thư viện có sẵn cho Arduino, ESP32 và các nền tảng khác.

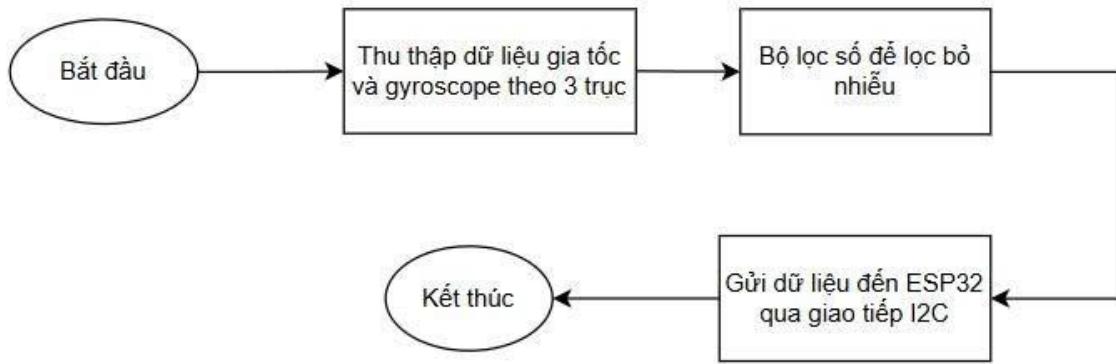


Hình 6: Sơ đồ khái quát quy trình hoạt động của module GPS NEO 6M

1.5 Cảm biến gia tốc GY-521 6DOF IMU MPU6050

Cảm biến gia tốc GY-521 là module IMU 6DOF tích hợp chip MPU6050, cung cấp dữ liệu về gia tốc 3 trục và con quay hồi chuyển 3 trục. Điều này cho phép nó đo lường cả chuyển động thẳng và xoay, mang lại thông tin toàn diện về định hướng và chuyển động của vật thể trong không gian. [11]

Giao tiếp qua I2C, GY-521 dễ dàng kết nối với các vi điều khiển như Arduino hay ESP32. Module này được ứng dụng rộng rãi trong cảnh báo té ngã, máy bay không người lái và các hệ thống nhận diện cử chỉ, là lựa chọn lý tưởng cho các dự án cần dữ liệu chuyển động chi tiết và chính xác.



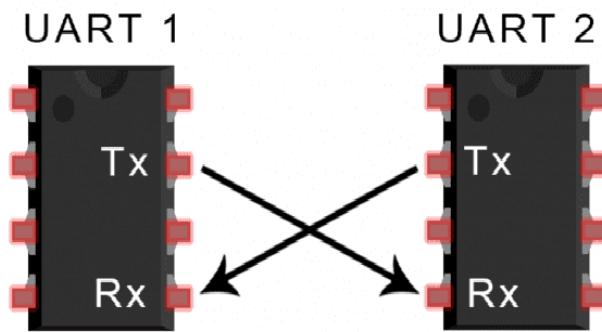
Hình 7: Sơ đồ khối quy trình hoạt động của cảm biến MPU6050

2. Giải pháp truyền thông

2.1 Giao thức UART

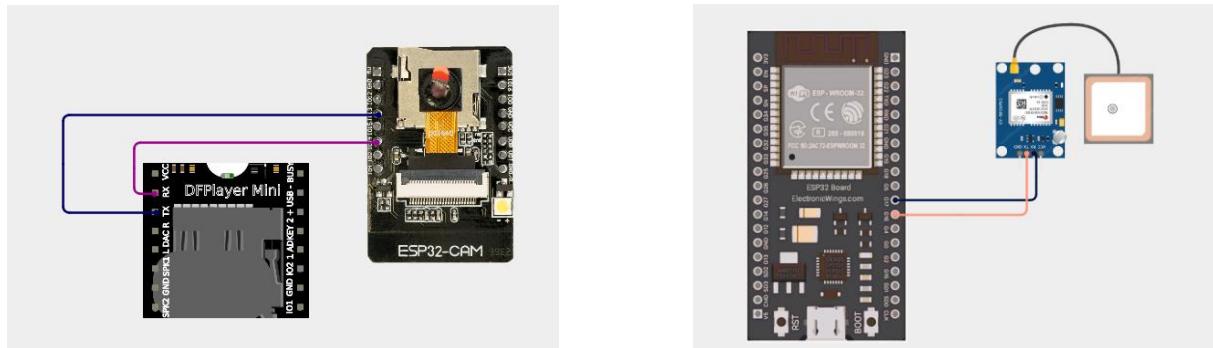
UART (Universal Asynchronous Receiver-Transmitter) là một giao thức truyền thông nối tiếp không đồng bộ rất phổ biến, dùng để trao đổi dữ liệu giữa các vi điều khiển và các module ngoại vi. Nó hoạt động với hai đường truyền độc lập: TX (truyền) để gửi dữ liệu đi và RX (nhận) để tiếp nhận dữ liệu đến. Dữ liệu được gửi đi dưới dạng các khung (dataframe) với cấu trúc chuẩn, bao gồm bit bắt đầu, các bit dữ liệu, bit kiểm tra chẵn lẻ (tùy chọn), và một hoặc nhiều bit dừng. [12]

Tốc độ truyền của UART thường được đặt ở các giá trị chuẩn như 9600, 19200, 115200 baud, định nghĩa số lượng bit được truyền mỗi giây.



Hình 8: Minh họa giao thức UART

Nhóm sử dụng giao thức UART để thực hiện truyền thông giữa ESP32-CAM với DF Player Mini và ESP32 với module GPS Neo6M, tận dụng sự đơn giản và hiệu quả của UART để các module này có thể "nói chuyện" với nhau.

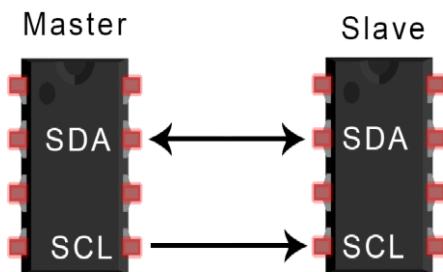


Hình 9: Giao tiếp UART giữa ESP32-CAM với DF Player Mini và ESP32 với GPS Neo6M

2.2 Giao thức I2C

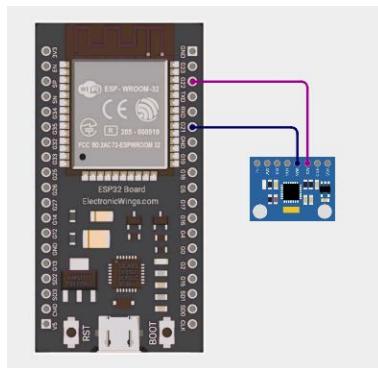
Giao thức I2C (Inter-Integrated Circuit) là một giao thức truyền thông nối tiếp đồng bộ phổ biến. Nó cho phép vi điều khiển giao tiếp với các thiết bị ngoại vi sử dụng chỉ hai đường dây tín hiệu: SDA (đữ liệu) và SCL (đồng hồ). Điều này giúp đơn giản hóa đáng kể việc kết nối dây và giảm số lượng chân cần dùng trên vi điều khiển.

Về cách truyền, I2C hoạt động theo mô hình Master-Slave. Thiết bị Master (thường là vi điều khiển) sẽ tạo xung nhịp trên đường SCL để đồng bộ hóa và điều khiển việc truyền dữ liệu. Dữ liệu được gửi từng bit một trên đường SDA, với mỗi bit được đồng bộ bởi một xung trên SCL. Mỗi thiết bị Slave có một địa chỉ duy nhất, cho phép Master chọn chính xác thiết bị cần giao tiếp trong số nhiều Slave cùng kết nối trên một bus. [13]



Hình 10: Minh họa giao thức I2C

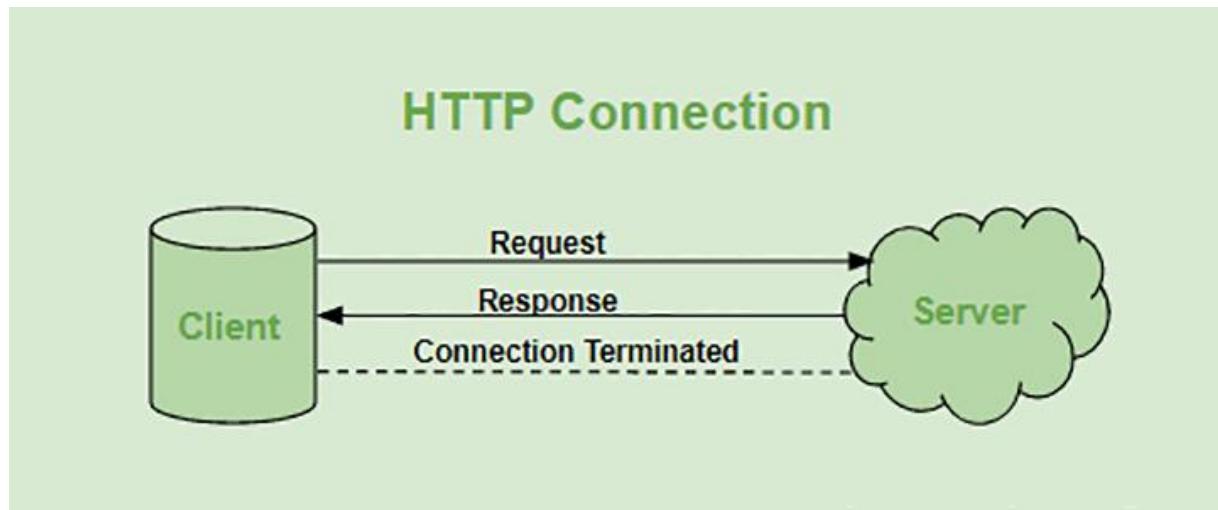
Nhóm sử dụng giao thức I2C để thực hiện truyền thông giữa ESP32 với cảm biến giá tốc GY-521.



Hình 11: Giao tiếp I2C giữa ESP32 với MPU6050

2.3. Giao thức HTTP

HTTP (Hypertext Transfer Protocol) là một giao thức ứng dụng cốt lõi cho việc truyền tải dữ liệu trên World Wide Web, và nó cũng đóng vai trò quan trọng trong các hệ thống IoT. HTTP thường được sử dụng theo mô hình yêu cầu-phản hồi (request-response). Giao thức này cho phép các thiết bị IoT (client) gửi yêu cầu và nhận phản hồi từ máy chủ (server), giúp chúng trao đổi thông tin để thực hiện các chức năng như gửi dữ liệu cảm biến, nhận lệnh điều khiển, hoặc cập nhật trạng thái.



Hình 12: Minh họa giao thức truyền thông HTTP

Nhóm sử dụng giao thức HTTP cho việc thực hiện truyền thông giữa ESP32 với Server GPS để gửi tọa độ và cảnh báo té ngã:

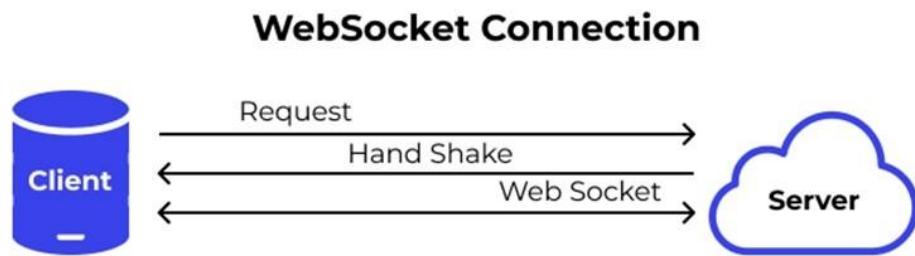


Hình 13: Minh họa quá trình gửi tọa độ và cảnh báo té ngã từ ESP đến server GPS

2.4. Giao thức Websocket

WebSocket là một giao thức truyền thông hai chiều (full-duplex) qua một kết nối TCP duy nhất, được thiết kế để khắc phục hạn chế của HTTP về việc giao tiếp liên tục. Không giống như HTTP hoạt động theo mô hình yêu cầu-phản hồi từng lượt, WebSocket cho phép máy chủ và client thiết lập một kênh kết nối duy trì sau khi bắt tay (handshake) ban đầu. Điều này giúp hai bên có thể gửi và nhận dữ liệu theo thời gian thực mà không cần phải gửi lại các yêu cầu HTTP lặp đi lặp lại.

Điểm mạnh của WebSocket là khả năng truyền dữ liệu theo thời gian thực và giảm độ trễ đáng kể. Một khi kết nối được thiết lập, cả thiết bị IoT và máy chủ đều có thể gửi/nhận dữ liệu chủ động mà không cần yêu cầu lại. Điều này cực kỳ lý tưởng cho các ứng dụng IoT cần cập nhật liên tục như giám sát, điều khiển tức thì, đảm bảo dữ liệu luôn được truyền tải ngay lập tức.



Hình 14: Minh họa giao thức truyền thông Websocket

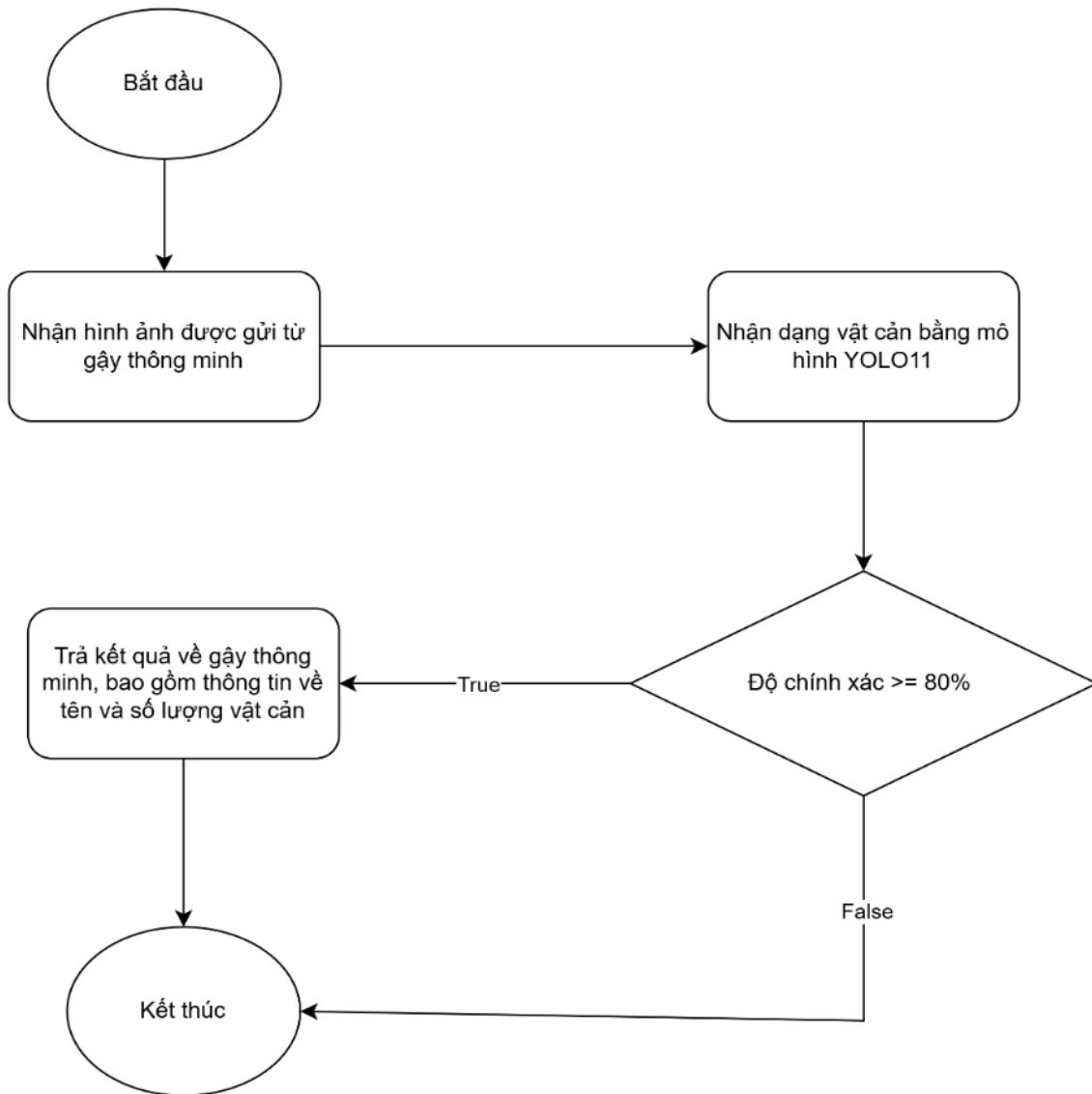
Nhóm sử dụng giao thức Websocket cho việc thực hiện truyền thông giữa ESP32- CAM với Server AI để gửi ảnh realtime:



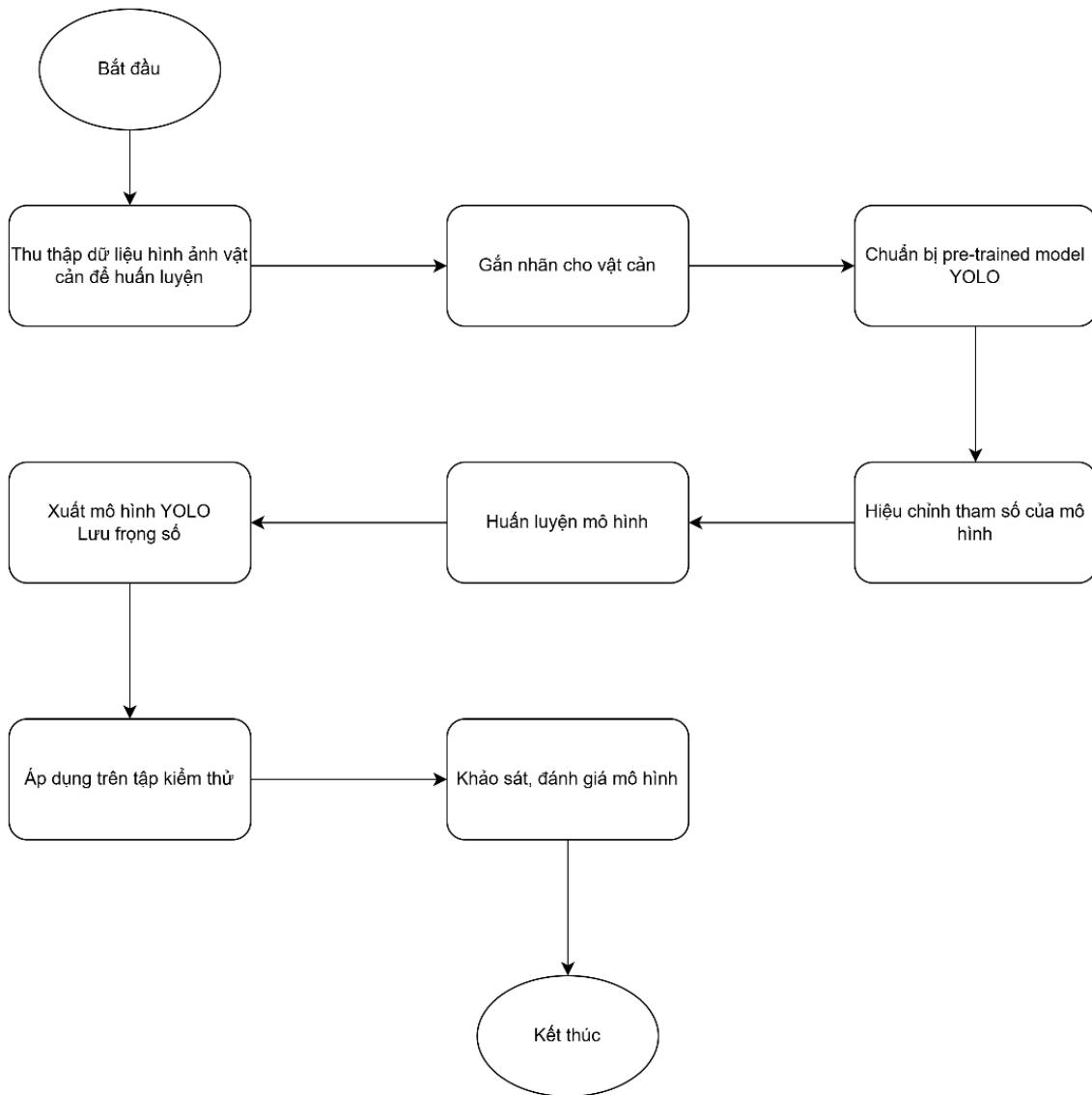
Hình 15: Minh họa quá trình gửi ảnh realtime từ ESP32-Cam đến Server AI

3. Giải pháp AI/KHDL

3.1. Giải pháp nhận diện vật cản



Hình 16: Sơ đồ giải pháp nhận diện vật cản



Hình 17: Sơ đồ quy trình xây dựng mô hình nhận diện vật cản

3.1.1 Giới thiệu mô hình YOLO11

❖ Tổng quan mô hình YOLO11

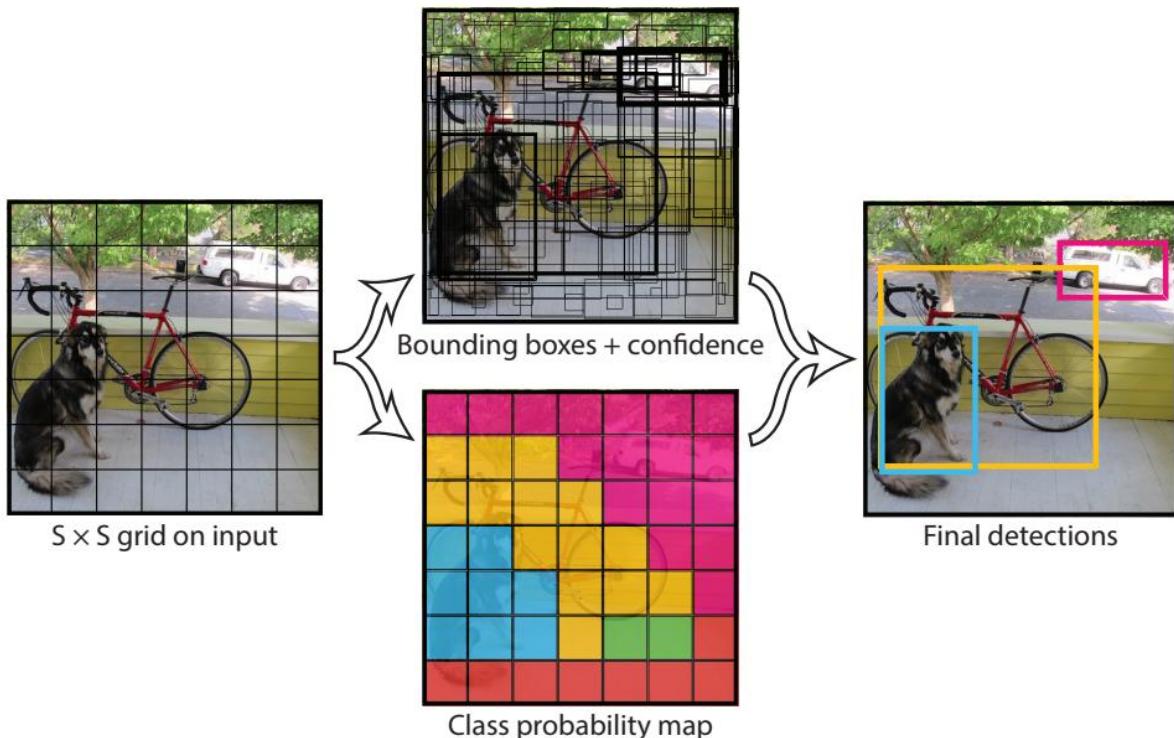
YOLO11 là một trong những phiên bản mới nhất trong loạt mô hình YOLO của Ultralytics, được thiết kế để phát hiện đối tượng theo thời gian thực. Mô hình này đặt mục tiêu định nghĩa lại khả năng với độ chính xác, tốc độ và hiệu quả tiên tiến. Dựa trên những tiến bộ của các phiên bản YOLO trước đó, YOLO11 mang đến những cải tiến đáng kể về kiến trúc và phương pháp đào tạo. Điều này làm cho YOLO11 trở thành một lựa chọn linh hoạt cho nhiều tác vụ thị giác máy tính. [14]

❖ Các cải tiến chính

Trích xuất tính năng nâng cao: YOLO11 được tăng cường khả năng trích xuất đặc trưng để phát hiện đối tượng chính xác hơn và thực hiện các tác vụ phức tạp hơn. [14]

Được tối ưu hóa cho hiệu quả và tốc độ: Mô hình được thiết kế kiên trúc tinh tế và quy trình đào tạo được tối ưu hóa, mang lại tốc độ xử lý nhanh hơn đồng thời duy trì sự cân bằng tối ưu giữa độ chính xác và hiệu suất. [14]

Khả năng thích ứng trong nhiều môi trường: YOLO11 có thể được triển khai liền mạch trên nhiều môi trường khác nhau, bao gồm các thiết bị biên, nền tảng đám mây và hệ thống hỗ trợ NVIDIA GPU. [14]



Hình 18: Minh họa cách mô hình YOLO hoạt động

❖ Đầu ra của YOLO11

Phát hiện đối tượng (Object Detection) có đầu ra bao gồm hộp giới hạn cho mỗi đối tượng được phát hiện, cùng với điểm tin cậy và lớp của đối tượng.

$$y^T = [p_0, \underbrace{\langle t_x, t_y, t_w, t_h \rangle}_{\text{bounding box}}, \underbrace{\langle p_1, p_2, \dots, p_c \rangle}_{\text{scores of } c \text{ classes}}]$$

Hình 19: Đầu ra của mô hình YOLO

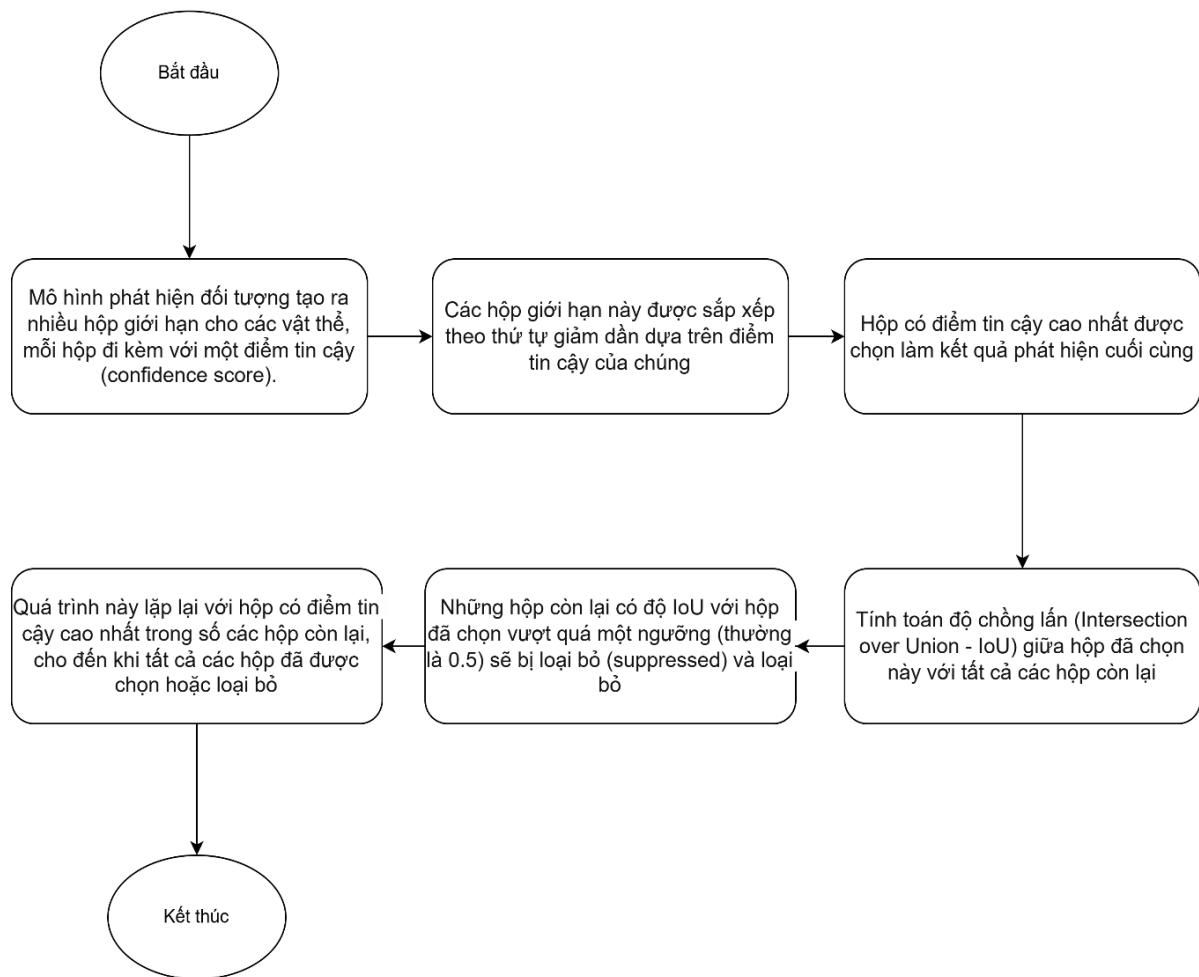
Trong đó:

- p_0 là xác suất dự báo vật thể xuất hiện trong bounding box.

- tx, ty, tw, th giúp xác định bounding box, tx, ty là tọa độ tâm và tw, th là kích thước rộng, dài của bounding box.
- $p_1, p_2, p_3, \dots, p_c$ là vector phân phối xác suất dự báo của các classes.

❖ Thuật toán Non-Maximum Suppression

Non-Maximum Suppression (NMS) là một phương pháp được sử dụng trong lĩnh vực phát hiện đối tượng (object detection). Mục đích của nó là để loại bỏ các hộp giới hạn (bounding boxes) dư thừa được phát hiện xung quanh cùng một đối tượng. Khi một mô hình phát hiện đối tượng dự đoán nhiều hộp giới hạn khác nhau cho cùng một vật thể, NMS sẽ giữ lại hộp tốt nhất và loại bỏ những hộp còn lại. Điều này giúp đảm bảo rằng mỗi đối tượng chỉ được đếm một lần.



Hình 20: Quá trình hoạt động cơ bản của Greedy NMS (loại phổ biến nhất)

❖ Loss function

$$L_{total} = L_{classification} + L_{localization} + L_{confidence}$$

Hình 21: Hàm loss function của mô hình YOLO

Hàm lỗi trong YOLO được tính trên việc dự đoán và nhãn mô hình để tính. Cụ thể hơn nó là tổng độ lỗi của 3 thành phần sau :

- Độ lỗi của việc dự đoán loại nhãn của object - Classification loss.
- Độ lỗi của dự đoán tọa độ tâm, chiều dài, rộng của boundary box (x, y ,w, h) - Localization loss.
- Độ lỗi của việc dự đoán bounding box đó chứa object so với nhãn thực tế tại ô vuông đó - Confidence loss.

3.1.2 Các bước huấn luyện mô hình

❖ Thu thập dữ liệu

- Lập danh sách các vật thể cần nhận dạng để thu thập hình ảnh.
- Ảnh được chụp trên thiết bị điện thoại cá nhân, sử dụng hình ảnh đã có sẵn trên Roboflow và ảnh chụp từ ESP32 Cam.

❖ Gắn nhãn dữ liệu với Roboflow

Quy tắc khi gắn nhãn :

- Bounding box chính xác: Bounding box phải ôm gọn vật thể, không để dư không gian trống hoặc thiếu phần vật thể.
- Khoảng cách gán nhãn: Đối với hệ thống hỗ trợ người mù, khoảng cách vật thể không nhất thiết phải giới hạn, nhưng cần đảm bảo vật thể rõ ràng trong hình ảnh.
- Bỏ qua vật thể bị che khuất: Không gán nhãn cho những vật thể bị che khuất hơn 40% diện tích. Vì các vật thể này có thể mất đi những đặc điểm quan trọng, làm cho mô hình học sai.

Đối với mỗi ảnh sau khi gán nhãn, sẽ tạo ra 1 file txt. Gọi là file annotation. Các thành phần của 1 file annotation như sau:

- Mỗi dòng là thông tin của 1 bounding box:
- Id: Thứ tự của class do mình định nghĩa.
- Tọa độ tâm và chiều rộng, chiều dài của bounding box theo chiều ngang và dọc của bức ảnh (X_center, Y_center, Width, Height).

❖ Quá trình huấn luyện

Các tham số là các biến ảnh hưởng đến cách mô hình học và khái quát hóa các mẫu trong dữ liệu. Các tham số này có thể có tác động đáng kể đến hiệu suất, độ chính xác và thời gian đào tạo của mô hình:

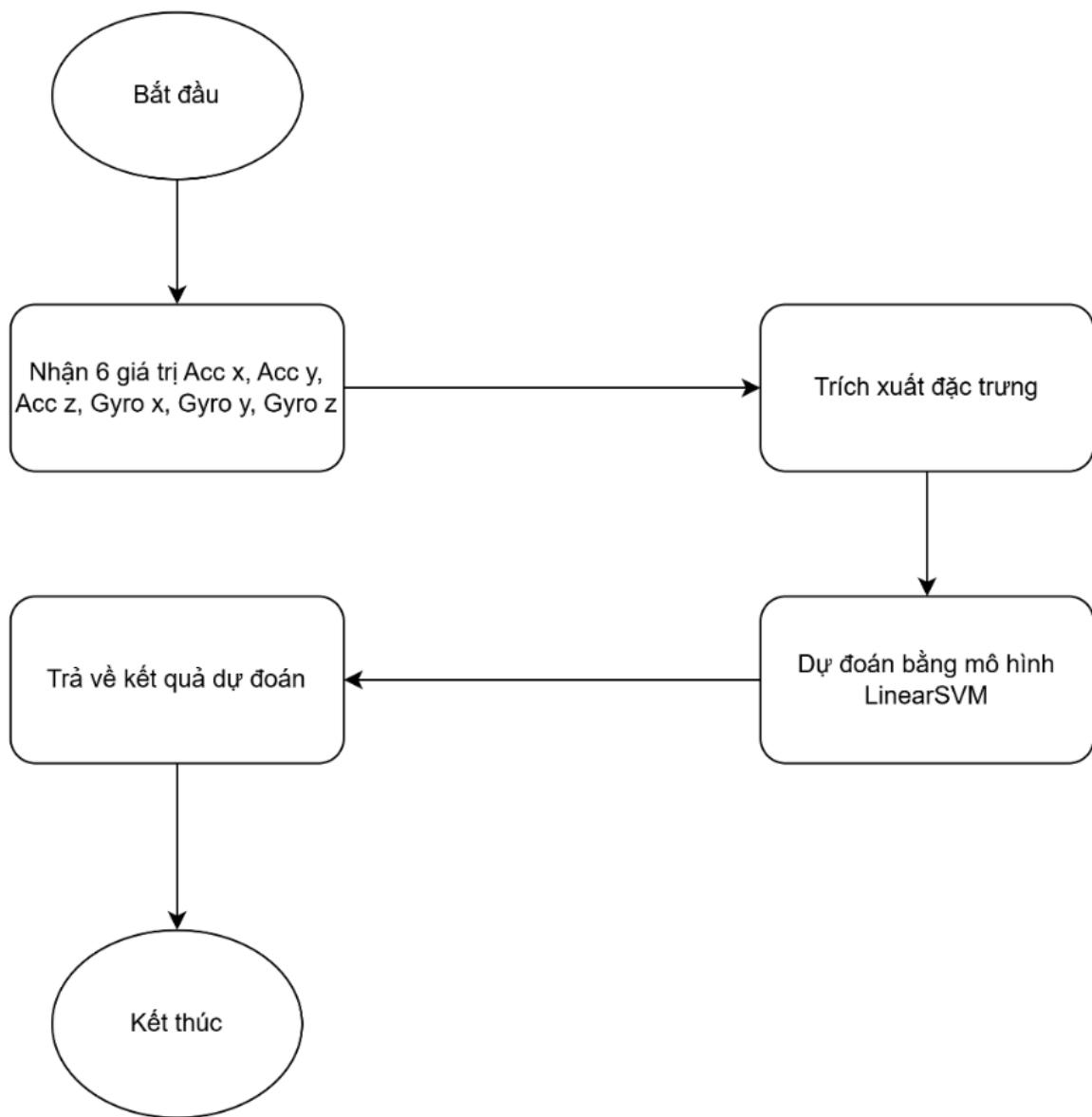
- Batch size: Là số lượng mẫu dữ liệu huấn luyện được xử lý trong một lần lặp. Batch size lớn giúp quá trình học nhanh hơn nhưng yêu cầu nhiều bộ nhớ hơn; Batch size nhỏ có thể giúp mô hình tổng quát hóa tốt hơn.
- Learning rate: Là kích thước bước mà các tham số của mô hình được cập nhật trong quá trình đào tạo.
 - Learning rate lớn: Học nhanh nhưng dễ bỏ qua điểm tối ưu.
 - Learning rate nhỏ: Học chậm nhưng có thể đạt được độ chính xác và ổn định tốt hơn.
- Epochs: là số lần mô hình đi qua toàn bộ tập dữ liệu huấn luyện. Mục tiêu của việc chạy nhiều Epoch là liên tục điều chỉnh trọng số của mô hình để cải thiện độ chính xác trên toàn bộ tập dữ liệu.

❖ Đánh giá mô hình

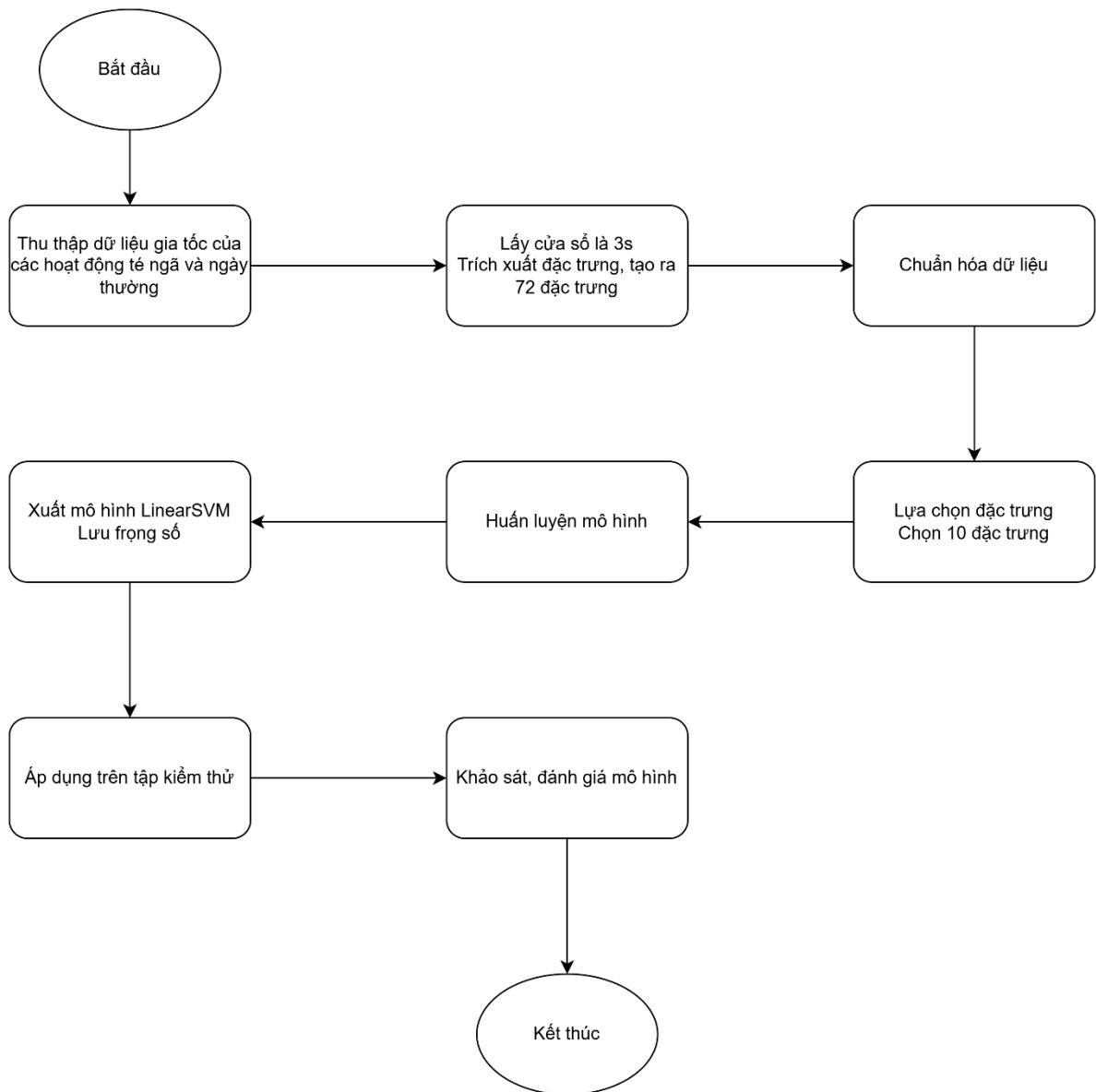
Một số chỉ số được sử dụng để đánh giá hiệu suất của mô hình như:

- Độ chính xác (Accuracy): Tỷ lệ tổng số dự đoán đúng trên tổng số dự đoán.
- Độ đúng (Precision): Tỷ lệ các dự đoán dương đúng trong tổng số dự đoán là dương.
- Độ nhạy (Recall): Tỷ lệ các dự đoán dương đúng trong tổng số mẫu dương thực tế.
- Điểm F1 (F1-score): Là trung bình điều hòa của Precision và Recall. F1-score đặc biệt hữu ích khi các lớp dữ liệu bị mất cân bằng, vì nó cân bằng giữa Precision và Recall.
- mAP (mean Average Precision): Đây là chỉ số quan trọng nhất trong phát hiện đối tượng. mAP tính toán giá trị Trung bình của Độ chính xác Trung bình (Average Precision - AP) trên tất cả các lớp đối tượng. AP lại được tính dựa trên đường cong Precision-Recall. mAP phản ánh khả năng phát hiện đúng đối tượng và định vị chúng một cách chính xác. Thường có các biến thể như mAP@0.5 (mAP với ngưỡng IoU là 0.5) và mAP@0.5:0.95 (mAP trung bình trên các ngưỡng IoU từ 0.5 đến 0.95 với bước nhảy 0.05).

3.2 Giải pháp nhận diện té ngã



Hình 22: Sơ đồ giải pháp nhận diện té ngã



Hình 23: Sơ đồ quy trình xây dựng mô hình nhận diện té ngã

3.2.1 Giới thiệu mô hình SVM

❖ Tổng quan mô hình SVM

Support Vector Machine (SVM) là một thuật toán học có giám sát cực kỳ hiệu quả, được sử dụng rộng rãi trong các bài toán phân loại nhị phân và dự báo. Điểm mạnh nổi bật của SVM nằm ở khả năng hoạt động tốt với các tập dữ liệu kích thước lớn và thường mang lại kết quả vượt trội so với nhiều thuật toán học có giám sát khác. [15]

Những ưu điểm nổi bật của SVM:

- Hiệu quả với không gian cao chiều (high-dimensional spaces): SVM xử lý xuất sắc các dữ liệu có nhiều đặc trưng, giúp bạn khai thác tối đa thông tin từ các tập dữ liệu phức tạp. [15]
- Tiêu tốn ít bộ nhớ: Thuật toán chỉ sử dụng một tập hợp nhỏ các điểm dữ liệu quan trọng, được gọi là vectơ hỗ trợ (support vectors), để xây dựng hàm quyết định. Điều này giúp tối ưu hóa việc sử dụng bộ nhớ và tăng tốc độ dự báo. [15]
- Linh hoạt với các hàm kernel: Bạn có thể tạo ra nhiều hàm quyết định khác nhau bằng cách sử dụng các hàm kernel (ví dụ: Linear, Polynomial, Radial Basis Function - RBF). Việc lựa chọn kernel phù hợp có thể cải thiện đáng kể hiệu suất của mô hình. [15]

Với tính hiệu quả vượt trội, SVM được ứng dụng rộng rãi trong nhiều lĩnh vực:

- Chẩn đoán bệnh: Từ các chỉ số xét nghiệm lâm sàng, SVM có thể dự báo nguy cơ mắc các bệnh như tiểu đường, suy thận, máu nhiễm mỡ, v.v.
- Phân loại tin tức và văn bản: Xác định chủ đề của một đoạn văn bản, phân loại cảm xúc (tích cực/tiêu cực), hay lọc thư rác một cách chính xác.
- Phát hiện té ngã: Dựa trên dữ liệu cảm biến, SVM có thể phát hiện và cảnh báo các sự kiện té ngã, đặc biệt hữu ích trong việc chăm sóc người cao tuổi hoặc bệnh nhân khiếm thị.

3.2.2 Các bước huấn luyện mô hình SVM

Để xây dựng một mô hình SVM mạnh mẽ, chúng ta cần tuân thủ một quy trình chặt chẽ bao gồm các bước sau:

❖ Thu thập dữ liệu

Dữ liệu tự thu thập thông qua cảm biến gia tốc MPU6050. Giả lập các hoạt động té ngã (ngã tới trước, ngã sang hai bên) và các hoạt động thường ngày (đứng yên, đi bộ, chạy bộ, ngồi, đi cầu thang) để có được một tập dữ liệu đa dạng.

❖ Gán nhãn

Gán nhãn cho hoạt động thường ngày: Mỗi hoạt động thường ngày sẽ được lưu dưới dạng file csv, mỗi hàng tương ứng 6 giá trị (Gia tốc kép trực X, Y, Z và Con quay hồi chuyển trực X, Y, Z). Dữ liệu được lấy mẫu với cửa sổ thời gian là 3 giây và độ chồng lấn (overlap) là 1.5 giây. Điều này giúp thu thập các mẫu liên tục, đảm bảo tính đầy đủ của dữ liệu.

Gán nhãn cho hoạt động té ngã: Mỗi hoạt động té ngã sẽ được lưu dưới dạng file csv đại diện cho một hoạt động té ngã duy nhất, mỗi hàng tương ứng 6 giá trị (Gia tốc kép trục X, Y, Z và Con quay hồi chuyển trục X, Y, Z). Dữ liệu được lấy mẫu với cửa sổ là 3 giây (tức 60 mẫu). Chọn điểm ban đầu có tổng bình phương của 3 giá trị Gia tốc kép trục X, Y, Z lớn nhất, sau đó lấy phía trước 30 mẫu và lấy phía sau 30 mẫu.

❖ Trích xuất đặc trưng

Tạo đặc trưng: Đối với mỗi giá trị từ các cảm biến:

- Gia tốc kép: Trục X, Y, Z (đơn vị: g)
- Con quay hồi chuyển: Trục X, Y, Z (đơn vị: rad/s)

Chúng tôi tạo ra 12 đặc trưng thống kê, bao gồm:

- Mean (Giá trị trung bình)
- StandardDeviation (Độ lệch chuẩn)
- RootMeanSquare (Căn bậc hai trung bình bình phương)
- MaximalAmplitude (Biên độ tối đa)
- MinimalAmplitude (Biên độ tối thiểu)
- Median (Trung vị)
- Number of zero-crossing (Số lần vượt qua điểm 0)
- Skewness (Độ xiên)
- Kurtosis (Độ nhọn)
- First Quartile (Tứ phân vị thứ nhất)
- Third Quartile (Tứ phân vị thứ ba)
- Autocorrelation (Tự tương quan)

Tổng cộng, chúng ta thu được 72 đặc trưng cho mỗi mẫu dữ liệu.

Lựa chọn đặc trưng quan trọng: Để tối ưu hóa hiệu suất và giảm thiểu nhiễu, chúng ta sử dụng hai mô hình **ExtraTreesClassifier** và **RFE (Recursive Feature Elimination)**. Lựa chọn 10 đặc trưng quan trọng nhất để đưa vào huấn luyện mô hình.

❖ Quá trình huấn luyện

➤ Chia tách dữ liệu

Để đảm bảo mô hình được đánh giá một cách khách quan và không bị "nhớ" dữ liệu huấn luyện (overfitting), chúng tôi chia tập dữ liệu đã chuẩn bị thành hai phần theo tiêu chí “người – hoạt động”:

- Tập huấn luyện (Training set): Chiếm 70% dữ liệu. Mô hình sẽ học từ tập này để nhận diện các mối quan hệ và mẫu hình trong dữ liệu.
 - Tập kiểm tra (Testing set): Chiếm 30% dữ liệu còn lại. Tập này được sử dụng để đánh giá hiệu suất cuối cùng của mô hình sau khi huấn luyện, đảm bảo mô hình có thể tổng quát hóa tốt trên dữ liệu mới chưa từng thấy.
- Lựa chọn và tinh chỉnh siêu tham số (Hyperparameter Tuning)

Mô hình SVM có một số **siêu tham số** cần được điều chỉnh để đạt hiệu suất tối ưu. Hai siêu tham số quan trọng nhất của SVM là:

Tham số C (Regularization parameter): Kiểm soát sự đánh đổi giữa việc phân loại đúng các điểm dữ liệu huấn luyện và việc giữ cho biên phân chia càng rộng càng tốt.

- C nhỏ: Biên rộng hơn, ít bị phạt khi có lỗi phân loại, dẫn đến mô hình đơn giản hơn, có thể bị thiên lệch (bias) nhưng ít bị phuông sai (variance) (để generalize hơn).
- C lớn: Biên hẹp hơn, phạt nặng hơn khi có lỗi phân loại, dẫn đến mô hình phức tạp hơn, dễ bị overfitting nhưng có thể đạt độ chính xác cao hơn trên tập huấn luyện.

Hàm Kernel (Kernel function): Xác định cách các điểm dữ liệu được biến đổi vào không gian chiều cao hơn để có thể tìm được biên phân chia tuyến tính. Các hàm kernel phổ biến bao gồm:

- Linear (Tuyến tính): Phù hợp khi dữ liệu có thể tách biệt tuyến tính.
- Polynomial (Đa thức): Tạo ra biên phân chia cong, phù hợp với dữ liệu phức tạp hơn.
- RBF (Radial Basis Function / Gaussian Kernel): Thường là lựa chọn mặc định và hoạt động hiệu quả với nhiều loại dữ liệu phi tuyến tính, tạo ra biên phân chia phức tạp.

Để tìm ra bộ siêu tham số tốt nhất, chúng tôi sử dụng kỹ thuật **Tối ưu hóa Bayes (Bayesian Optimization):** Sử dụng các mô hình xác suất để tìm kiếm hiệu quả hơn trong không gian siêu tham số.

❖ Đánh giá mô hình

- Ma trận nhầm lẫn (Confusion Matrix)

Đây là bảng tóm tắt hiệu suất của mô hình phân loại, hiển thị số lượng các dự đoán đúng và sai cho từng lớp. Ma trận nhầm lẫn bao gồm bốn giá trị cơ bản:

- True Positive (TP): Số lượng mẫu dương được dự đoán đúng là dương.
- True Negative (TN): Số lượng mẫu âm được dự đoán đúng là âm.
- False Positive (FP): Số lượng mẫu âm bị dự đoán sai là dương (Lỗi loại I).
- False Negative (FN): Số lượng mẫu dương bị dự đoán sai là âm (Lỗi loại II).

➤ Các chỉ số đánh giá chính

Từ ma trận nhầm lẫn, chúng ta có thể tính toán các chỉ số quan trọng sau:

- Độ chính xác (Accuracy): Tỷ lệ tổng số dự đoán đúng trên tổng số dự đoán.
- Độ đúng (Precision): Tỷ lệ các dự đoán dương đúng trong tổng số dự đoán là dương.
- Độ nhạy (Recall): Tỷ lệ các dự đoán dương đúng trong tổng số mẫu dương thực tế.
- Điểm F1 (F1-score): Là trung bình điều hòa của Precision và Recall. F1-score đặc biệt hữu ích khi các lớp dữ liệu bị mất cân bằng, vì nó cân bằng giữa Precision và Recall.

➤ Đường cong ROC

- Đường cong ROC (Receiver Operating Characteristic): Là một đồ thị biểu diễn mối quan hệ giữa Tỷ lệ Đúng dương (True Positive Rate - TPR, chính là Recall) và Tỷ lệ Sai dương (False Positive Rate - FPR) ở các ngưỡng phân loại khác nhau.

4. Giải pháp phần mềm

4.1. Phát biểu bài toán

Xây dựng ứng dụng di động dành cho người thám hoặc người quản lý để hỗ trợ việc theo dõi vị trí của người khiếm thị. Ứng dụng tập trung vào chức năng tìm kiếm vị trí hiện tại của người khiếm thị nhằm giúp người thám theo dõi và cung cấp sự hỗ trợ khi cần thiết. Ngoài ra, còn hỗ trợ các chức năng như thêm thiết bị, truy xuất thông tin lịch sử vị trí của thiết bị.

4.2. Công nghệ sử dụng

- Mã nguồn: Native Android bằng Java
- Database: Firebase Realtime Database
- Nền tảng: Android

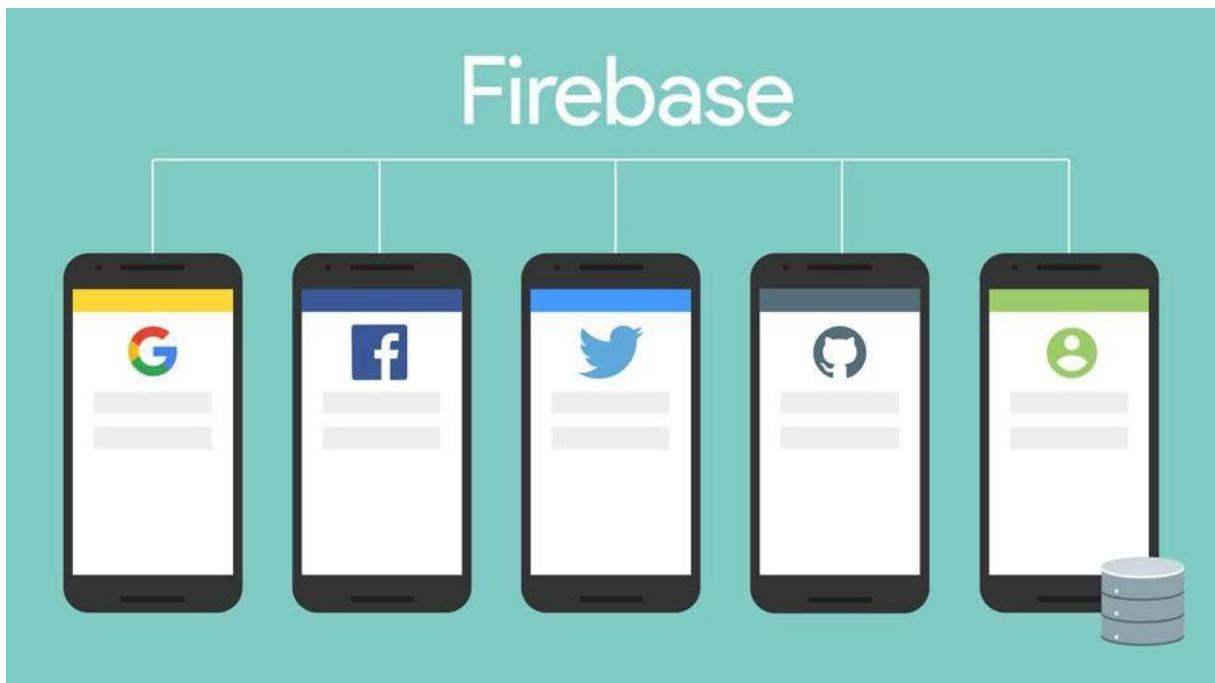
4.3. Giải pháp sử dụng Firebase

Firebase là một nền tảng phát triển ứng dụng di động và web do Google phát triển, cung cấp một bộ công cụ mạnh mẽ giúp các nhà phát triển xây dựng ứng dụng nhanh chóng và dễ

dàng. Firebase bao gồm các dịch vụ như cơ sở dữ liệu thời gian thực, xác thực người dùng, lưu trữ đám mây, và các chức năng đám mây, tất cả đều có thể tích hợp một cách liền mạch vào ứng dụng của bạn.

4.3.1 Firebase Authentication

Firebase Authentication cung cấp các phương pháp xác thực người dùng an toàn và đơn giản. Dịch vụ này hỗ trợ nhiều phương thức đăng nhập, bao gồm email và mật khẩu, số điện thoại, cũng như các tài khoản bên thứ ba như Google, Facebook, và Twitter. [16]



Hình 24: Dịch vụ hỗ trợ nhiều phương thức đăng nhập

❖ Ưu điểm:

- Bảo mật cao: Firebase Authentication tuân thủ các tiêu chuẩn bảo mật cao, giúp bảo vệ thông tin người dùng.
- Đa dạng phương thức đăng nhập: Hỗ trợ nhiều phương thức đăng nhập khác nhau, giúp người dùng có nhiều lựa chọn và trải nghiệm tốt hơn.
- Dễ dàng tích hợp: Firebase Authentication có thể dễ dàng tích hợp vào ứng dụng với các SDK và API có sẵn.

Ứng dụng trong giải pháp truyền thông: Firebase Authentication giúp quản lý người dùng hiệu quả, đảm bảo chỉ có những người dùng được xác thực mới có thể truy cập vào các tính năng truyền thông của ứng dụng, từ đó tăng cường tính bảo mật và độ tin cậy.

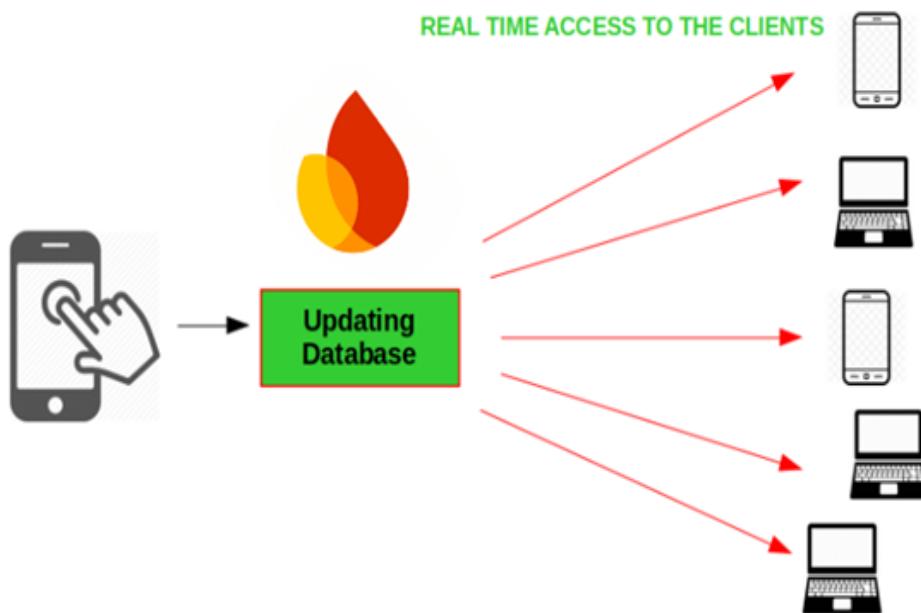
4.3.2 Firebase Realtime Database

Firebase lưu trữ dữ liệu database dưới dạng JSON và thực hiện đồng bộ database tới tất cả các client theo thời gian thực. [17] Cụ thể hơn là bạn có thể xây dựng được client đa nền tảng (cross-platform client) và tất cả các client này sẽ cùng sử dụng chung 1 database đến từ Firebase và có thể tự động cập nhật mỗi khi dữ liệu trong database được thêm mới hoặc sửa đổi.

❖ Ưu điểm:

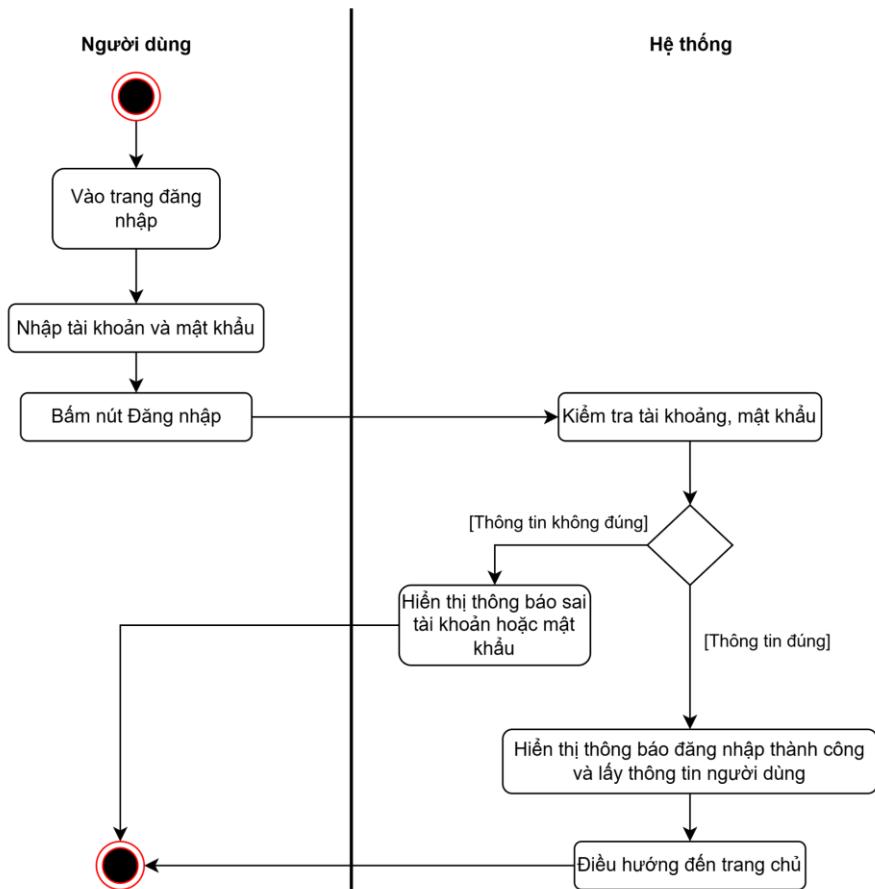
- Đồng bộ thời gian thực: Dữ liệu được cập nhật ngay lập tức đến tất cả các thiết bị kết nối khi có thay đổi.
- Hỗ trợ nhiều nền tảng: Có thể sử dụng với Android, iOS, và web.
- Tính mở rộng: Phù hợp với các ứng dụng quy mô nhỏ đến trung bình với yêu cầu thời gian thực cao.

Firebase Realtime Database có thể được sử dụng để lưu trữ và quản lý thông tin người dùng, thiết bị, và vị trí thiết bị trong thời gian thực. Điều này giúp đảm bảo tất cả người dùng luôn nhận được dữ liệu mới nhất, hỗ trợ giao tiếp mượt mà giữa các bên

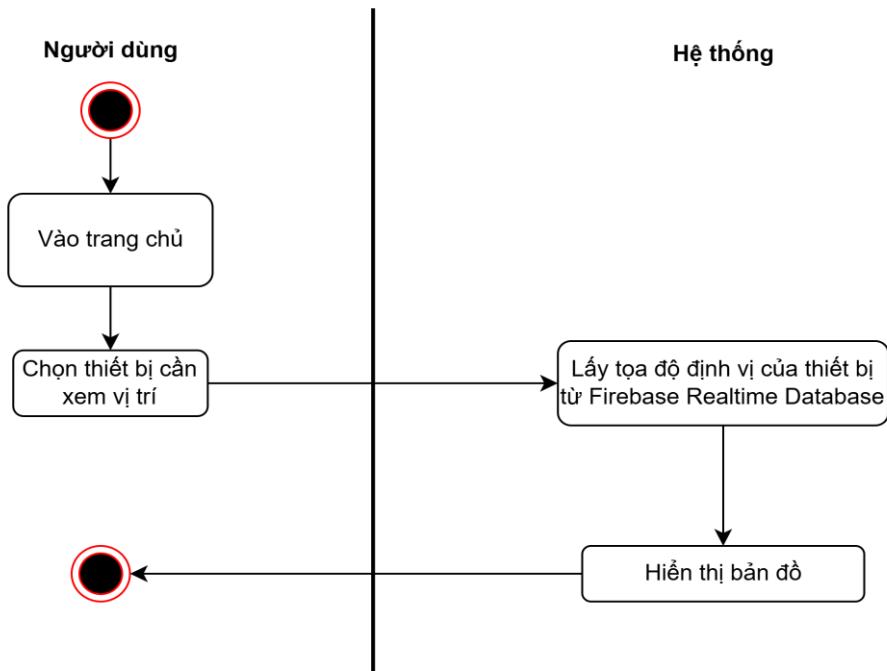


Hình 25: Mô hình hoạt động của Firebase Realtime Database

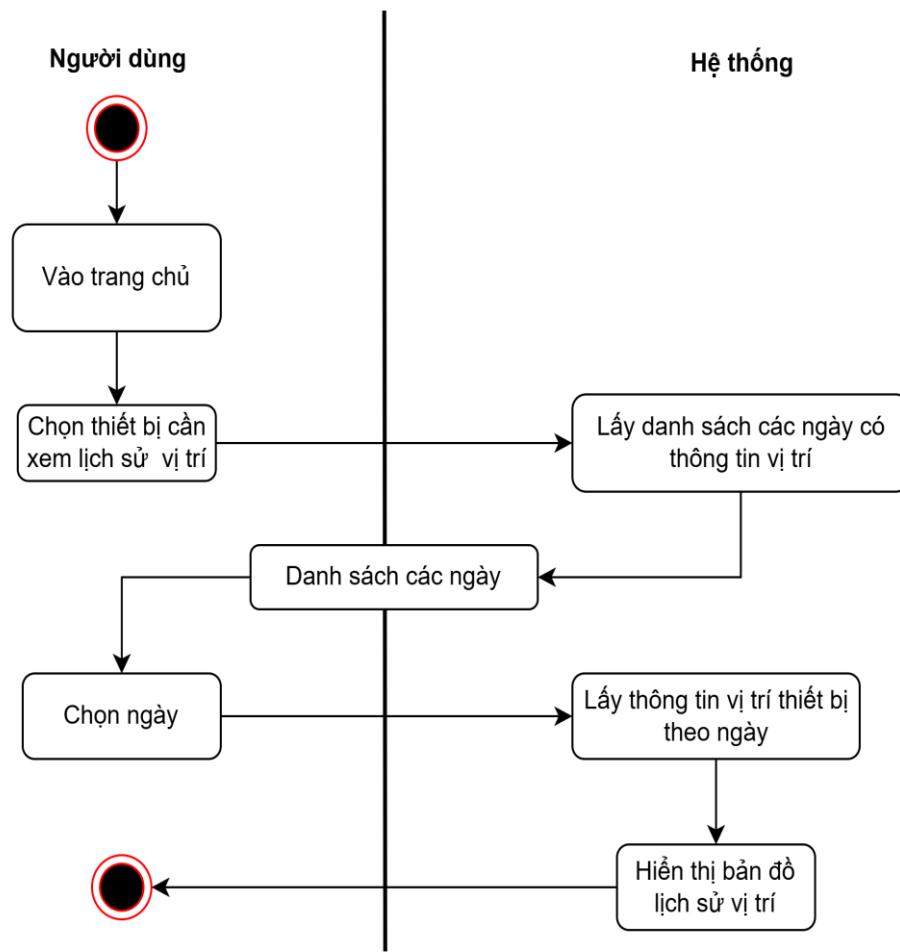
4.4. Sơ đồ hoạt động các chức năng của ứng dụng



Hình 26: Sơ đồ hoạt động chức năng đăng nhập

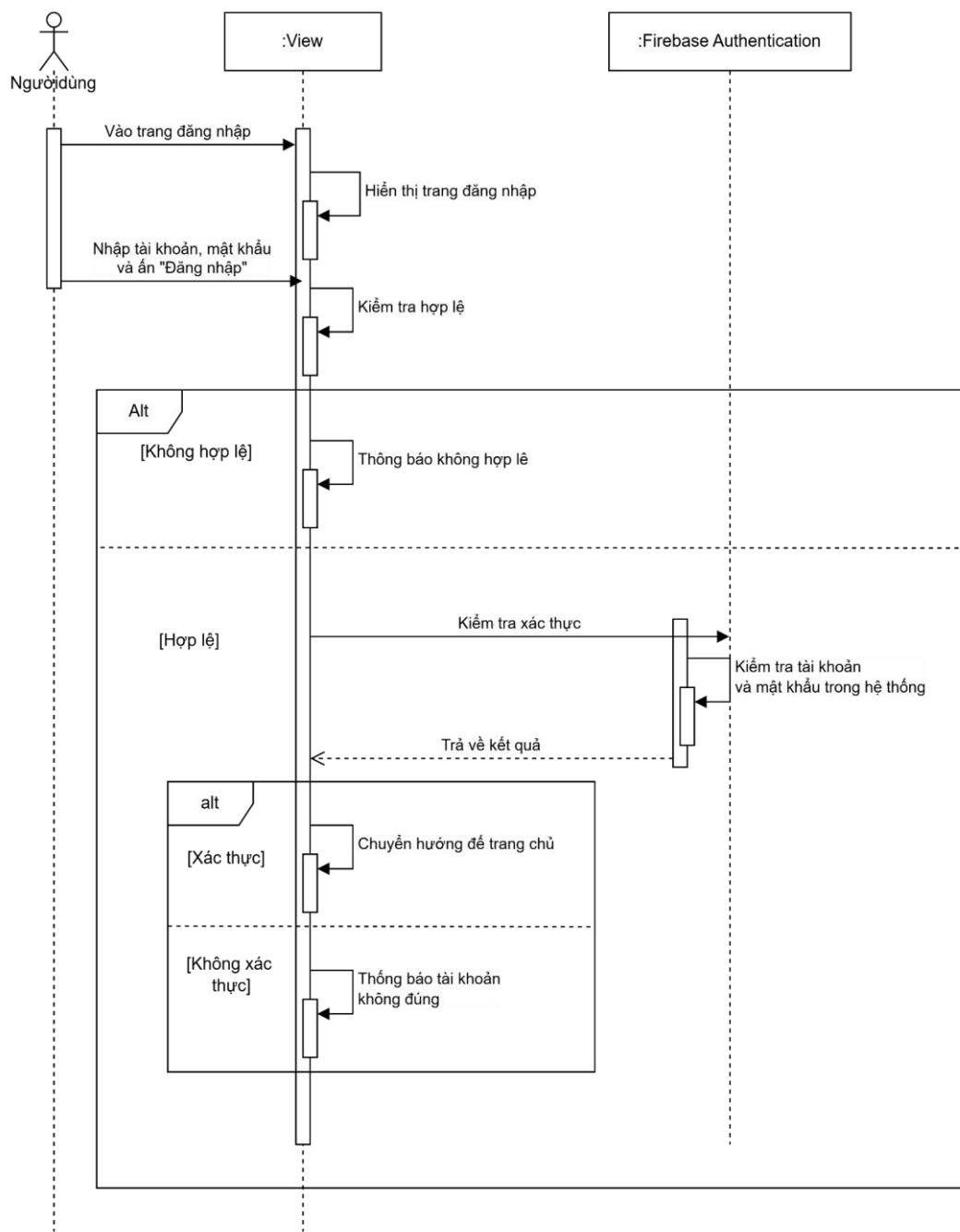


Hình 27: Sơ đồ hoạt động chức năng hiển thị vị trí thiết bị



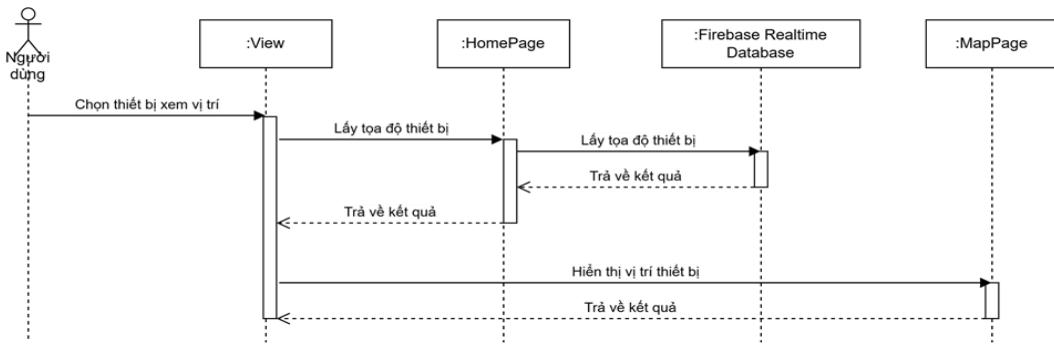
Hình 28: Sơ đồ hoạt động chức năng xem lịch sử vị trí của thiết bị

4.5. Sơ đồ tuần tự

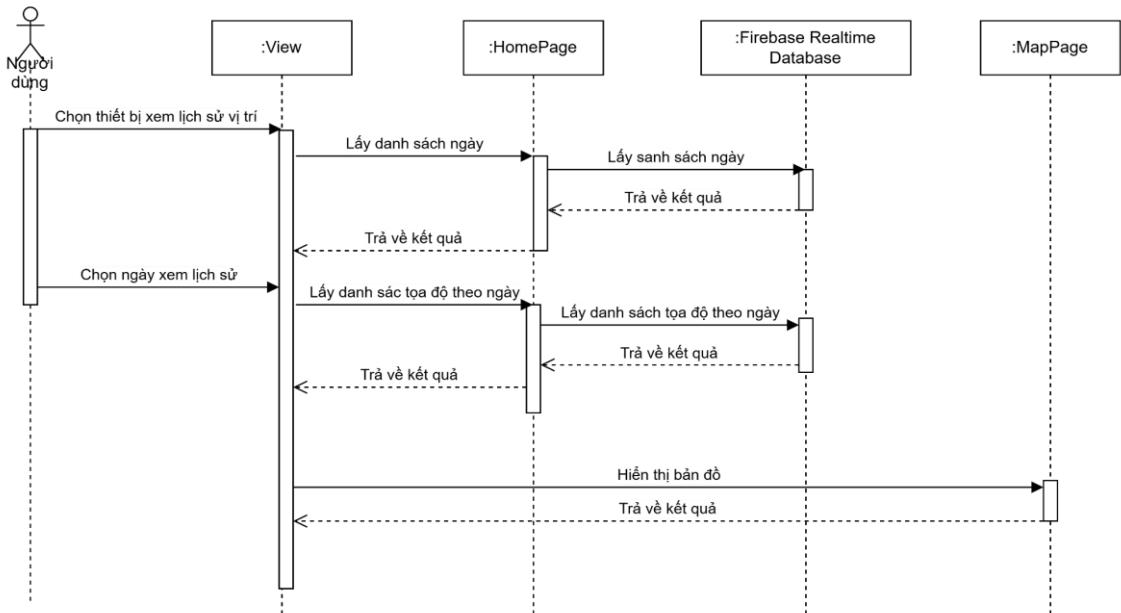


Hình 29: Sơ đồ tuần tự chức năng đăng nhập

Báo cáo đồ án PBL5 – Dự án kỹ thuật máy tính



Hình 30: Sơ đồ tuần tự chức năng xem vị trí của thiết bị

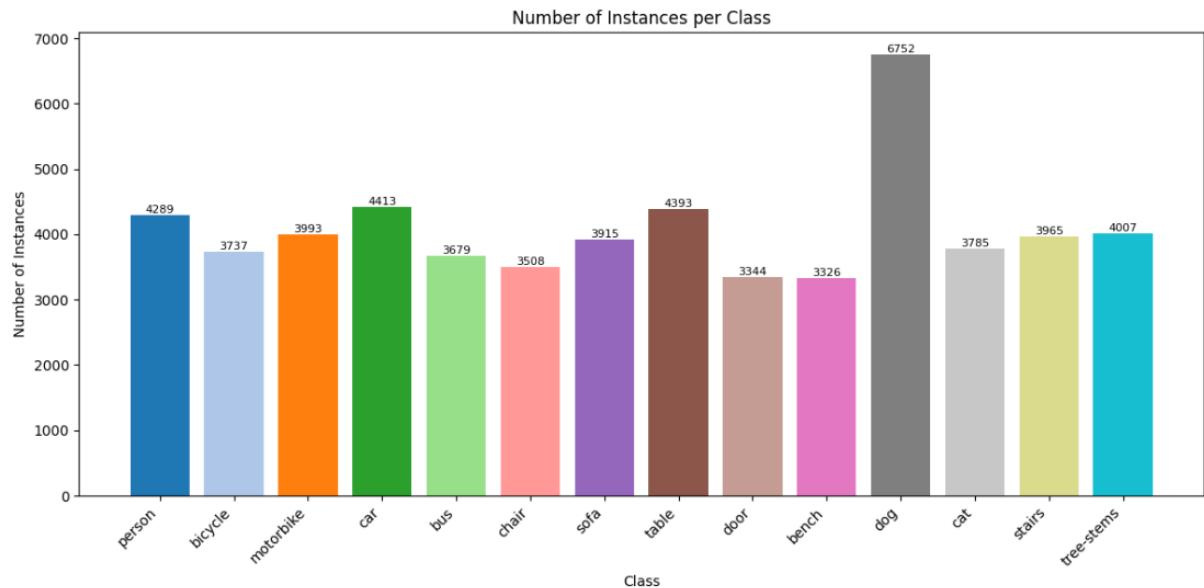


Hình 31: Sơ đồ tuần tự chức năng xem lịch sử vị trí thiết bị

III. KẾT QUẢ

1. Mô hình nhận diện vật cản YOLO11

1.1 Kết quả thu thập dữ liệu



Hình 32: Biểu đồ thống kê số lượng instances của mỗi class

Tập dữ liệu có đặc điểm sau:

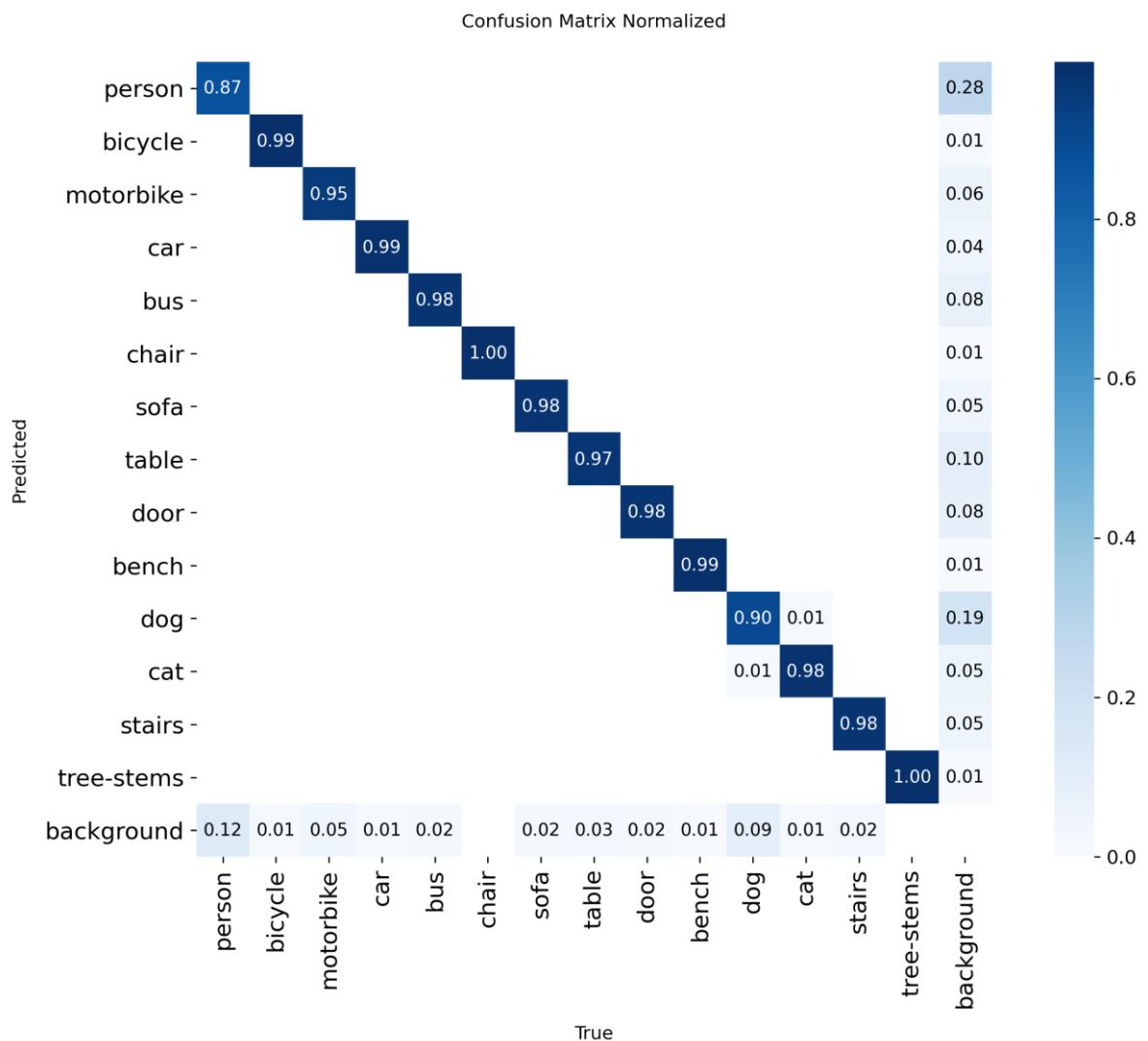
- Class có nhiều instance nhất là Dog trên 6000 instance.
- Tất cả class đều trên 3000 instance.
- Số lượng các ảnh của class Dog khá lớn vì có áp dụng Data Argumentation.

1.2 Tiến hành huấn luyện mô hình

Tham số	Giá trị	Ghi chú
epochs	200	Tổng số epoch mong muốn
imgsz	320	Kích thước ảnh huấn luyện và đánh giá
batch	16	Kích thước batch huấn luyện
lr0	0.01	Học suất ban đầu
lrf	0.01	Hệ số giảm học suất
momentum	0.937	Momentum cho SGD
box	7.5	Hệ số loss cho boading box
cls	0.5	Hệ số loss cho lớp
dfl	1.5	Hệ số loss cho dfl
hsv_h	0.015	Augmentation màu sắc (Hue)
hsv_s	0.7	Augmentation màu sắc (Saturation)
hsv_v	0.4	Augmentation màu sắc (Value)
degrees	16.74	Augmentation xoay
translate	0.1	Augmentation dịch chuyển
scale	0.5	Augmentation tỉ lệ
fliplr	0.5	Lật ảnh theo chiều ngang
mosaic	0.75	Augmentation mosaic
mixup	0	Augmentation mixup

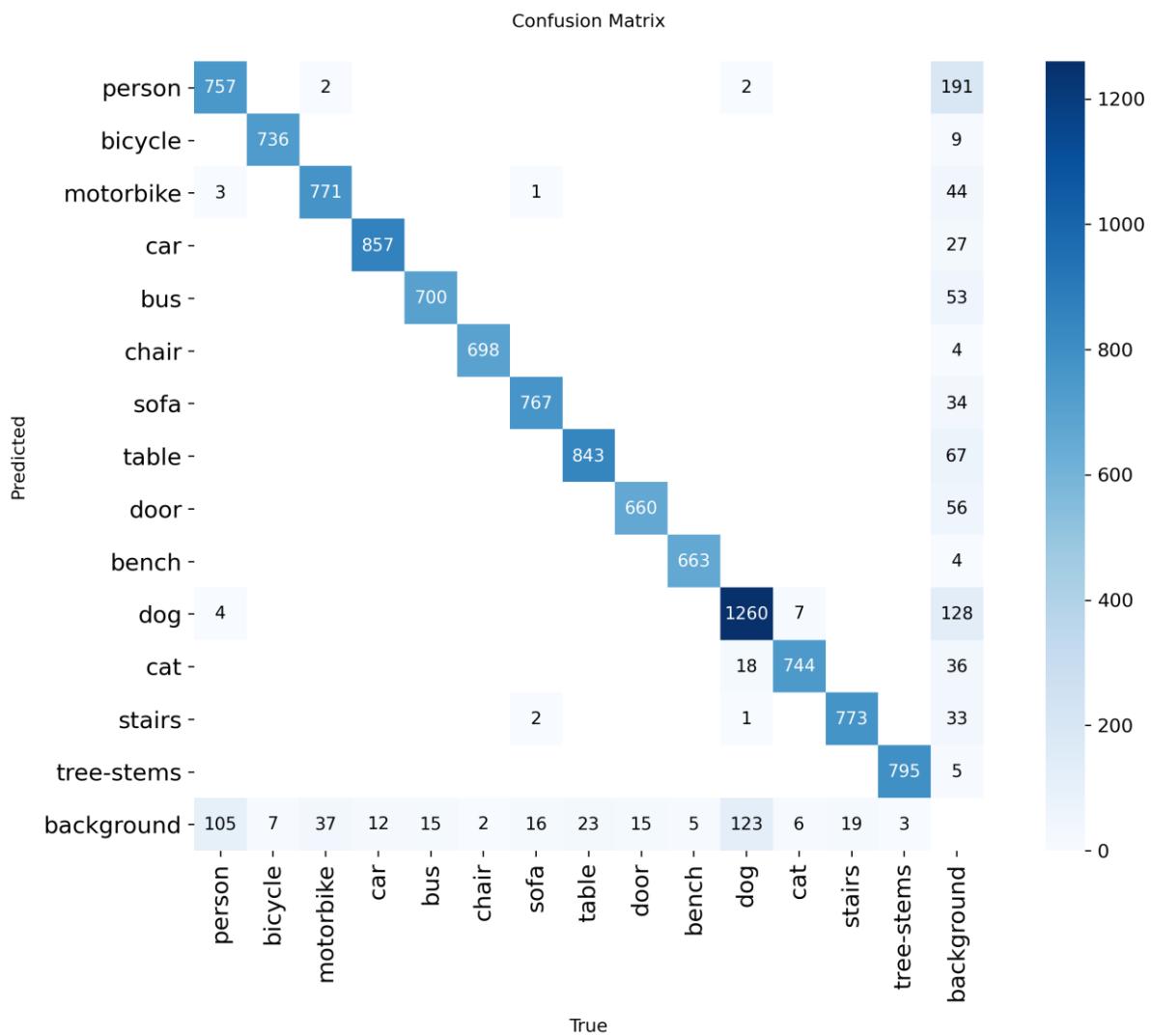
Bảng 3: Bảng danh sách tham số huấn luyện

1.3 Đánh giá mô hình



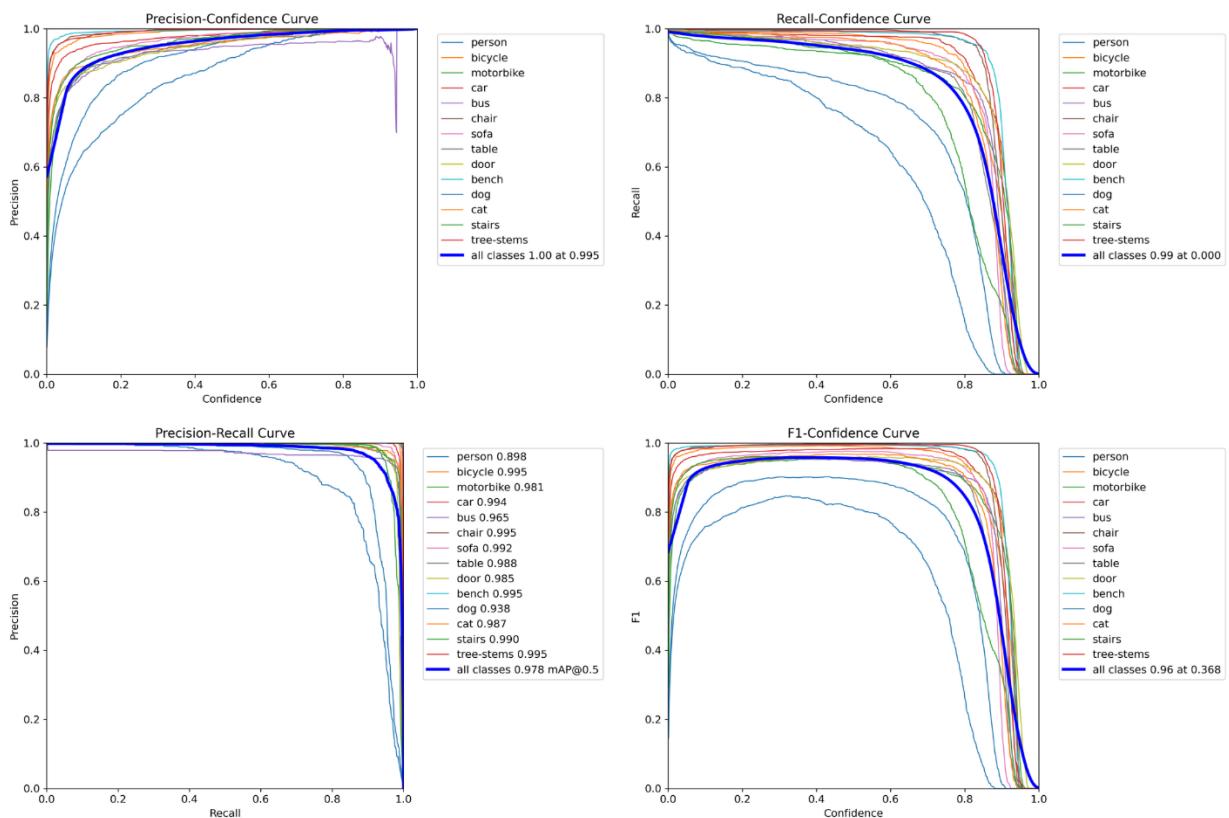
Hình 33: Ma trận nhầm lẫn chuẩn hóa mô hình YOLO

Báo cáo đỗ án PBL5 – Dự án kỹ thuật máy tính

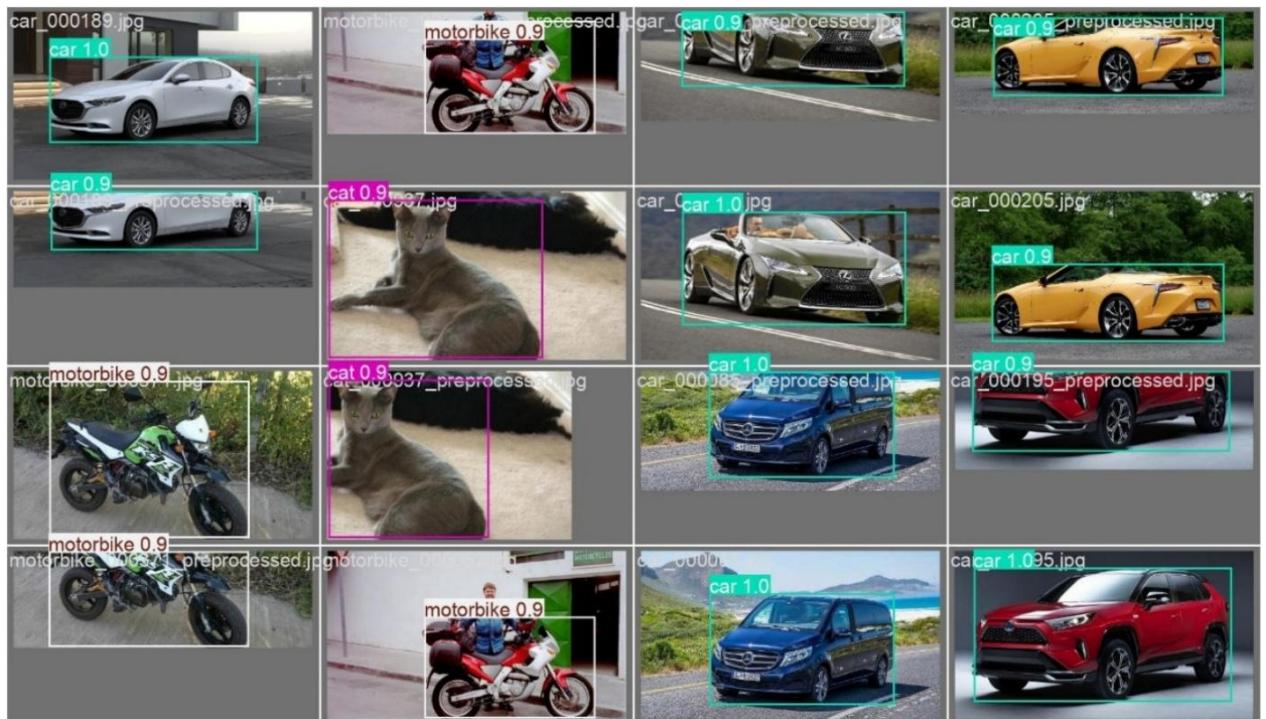


Hình 34: Ma trận nhầm lẫn mô hình YOLO

Báo cáo đồ án PBL5 – Dự án kỹ thuật máy tính



Hình 35: Biểu đồ đường cong đánh giá độ tin cậy của mô hình YOLO



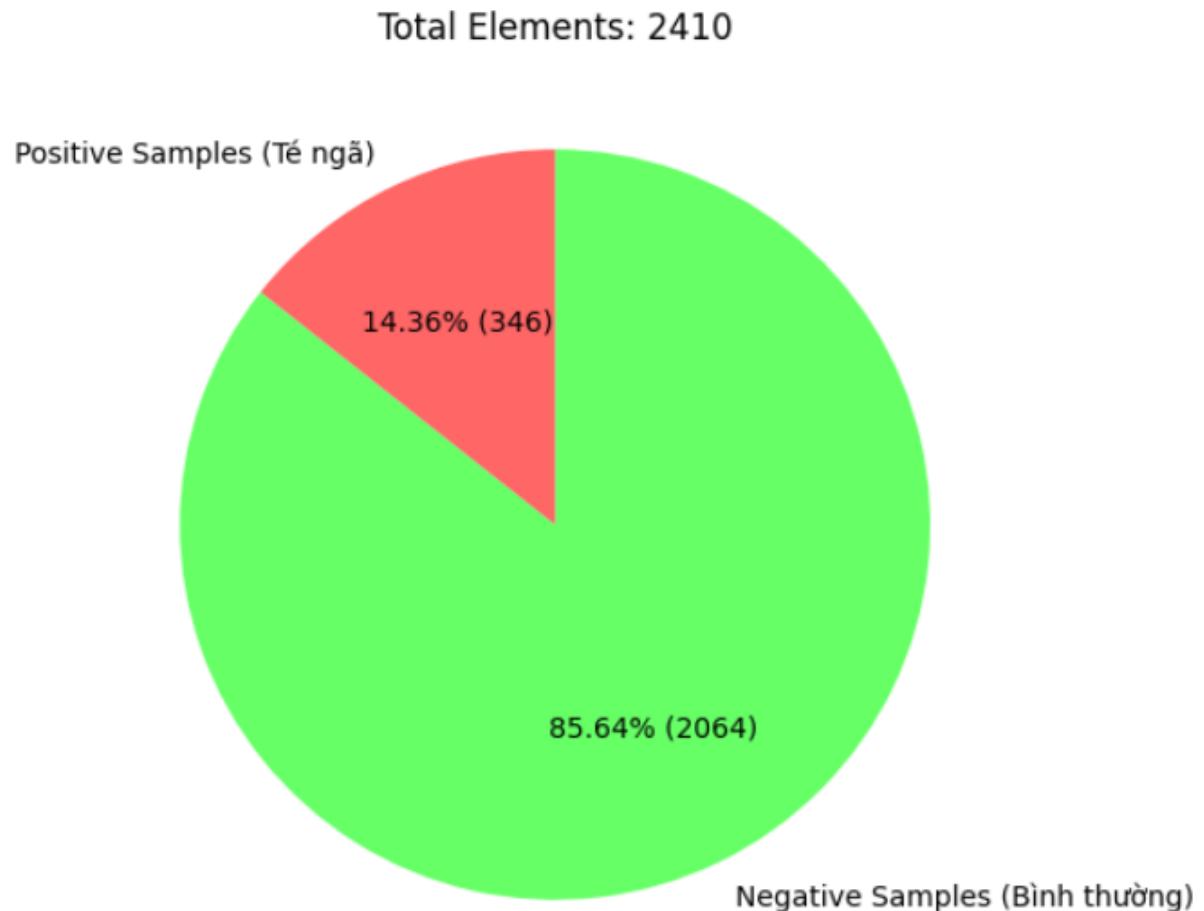
Hình 36: Kết quả dự đoán trên tập validation

Nhận xét: Độ chính xác trung bình đạt mức 87% trên tập xác thực. Mô hình đã học và khai

quát được nhiều đặc trưng hơn của các vật thể tương đối tốt các điều kiện ánh sáng khác nhau. Tuy nhiên, độ chính xác nhận dạng với các vật bị che khuất vẫn còn bị hạn chế, đối với các tập dữ liệu ảnh kiểm thử có chất lượng độ phân giải ảnh thấp cũng dẫn đến kết quả độ chính xác kiểm thử thấp.

2. Mô hình phát hiện té ngã SVM

2.1 Kết quả thu thập dữ liệu



Hình 37: Biểu đồ thống kê dữ liệu Té ngã

2.4 Tiến hành huấn luyện mô hình

Các đặc trưng quan trọng – 10 đặc trưng:

- Accelerometer: x-axis (g) RootMeanSquare
- Accelerometer: x-axis (g) MaximalAmplitude
- Accelerometer: x-axis (g) MinimalAmplitude
- Accelerometer: x-axis (g) Median
- Accelerometer: x-axis (g) Number of zero-crossing
- Accelerometer: x-axis (g) Kurtosis
- Accelerometer: x-axis (g) First Quartile
- Accelerometer: y-axis (g) Mean
- Gyroscope: z-axis (rad/s) Kurtosis
- Gyroscope: z-axis (rad/s) Autocorrelation

Best Parameter C: 0.007686918758498276.

Best Kernel: Linear

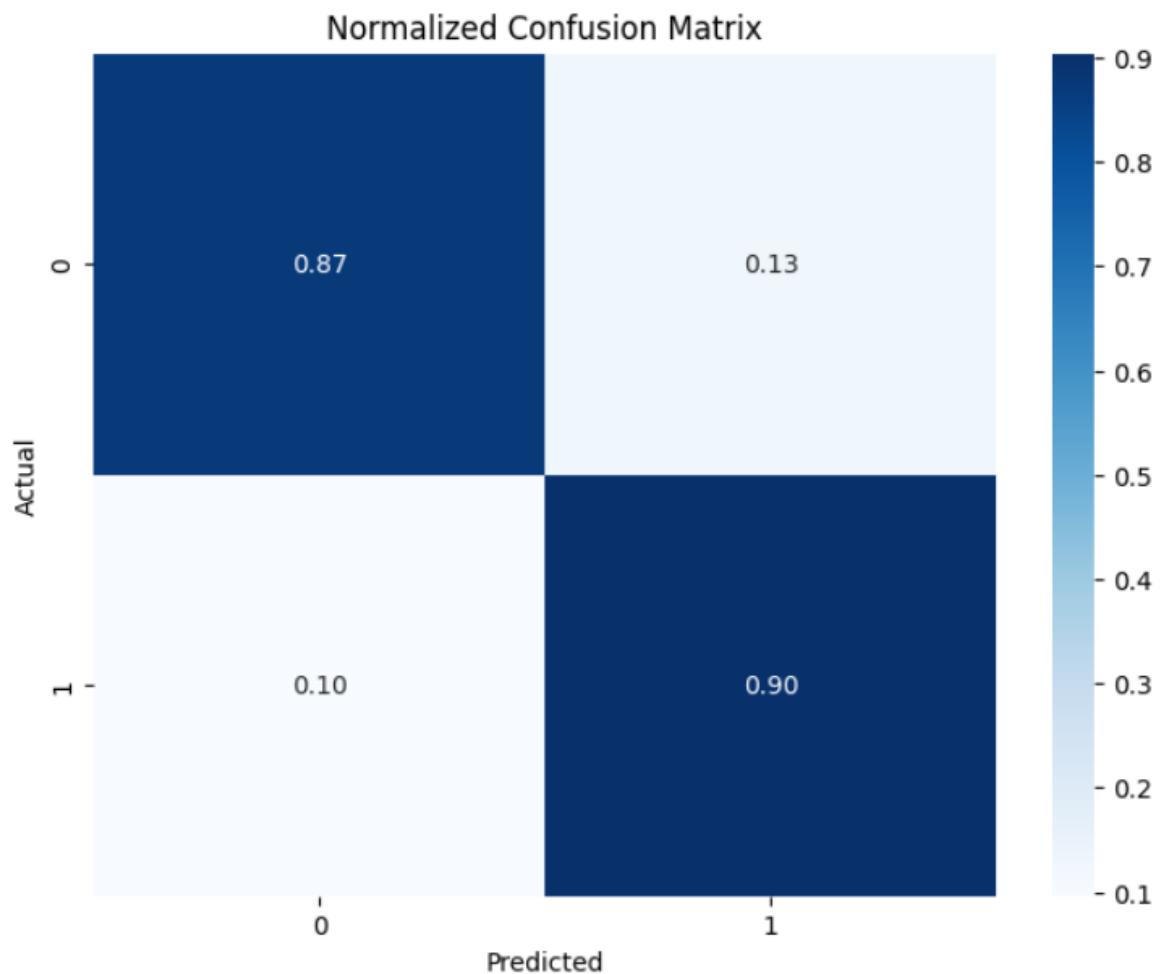
❖ Nhận xét:

Việc chọn kernel tuyến tính để phù hợp cho việc triển khai trên ESP32:

- Tính toán nhẹ: Giảm tải xử lý, giúp mô hình chạy nhanh hơn trên tài nguyên hạn chế.
- Kích thước nhỏ: Mô hình gọn nhẹ, tiết kiệm bộ nhớ cho thiết bị.
- Dễ triển khai: Đơn giản hóa việc chuyển đổi sang mã C/C++ phù hợp với môi trường nhúng.

Tham số C = 0.007686918758498276 để ưu tiên một biên phân cách mềm để tránh overfitting, đồng thời giúp mô hình tổng quát và nhẹ hơn.

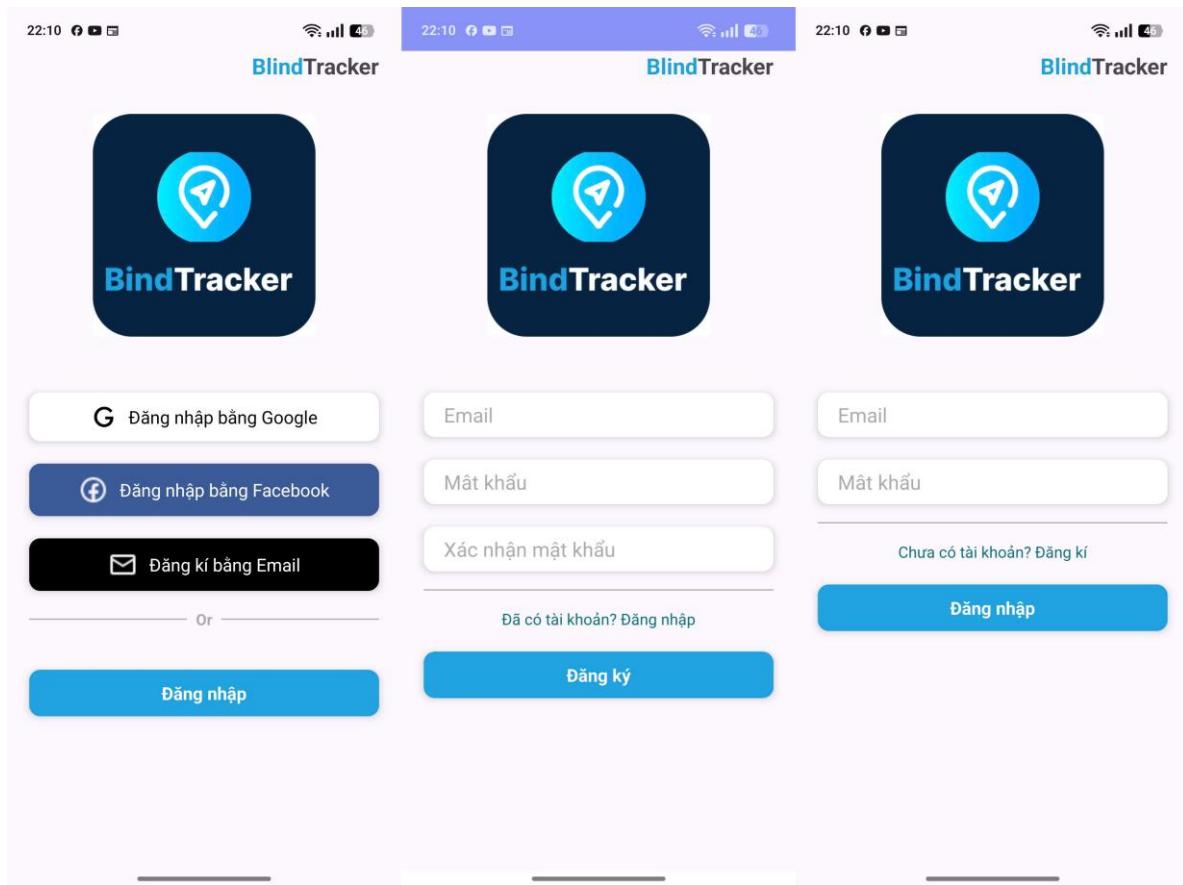
2.3 Đánh giá mô hình



Hình 38: Ma trận nhầm lẫn mô hình SVM trong nhận diện té ngã

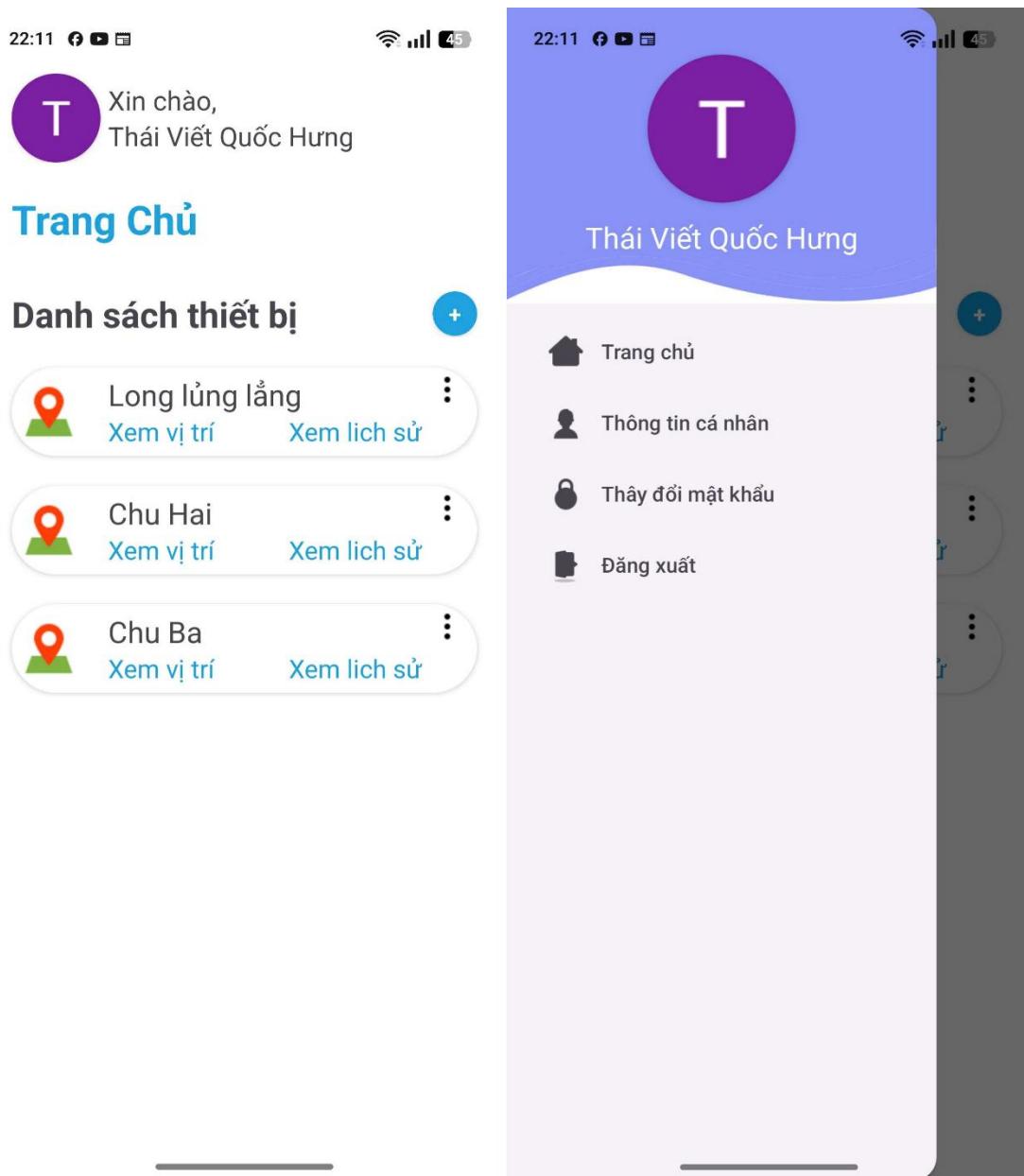
3. Ứng dụng di động

3.1 Giao diện chức năng đăng nhập, đăng ký



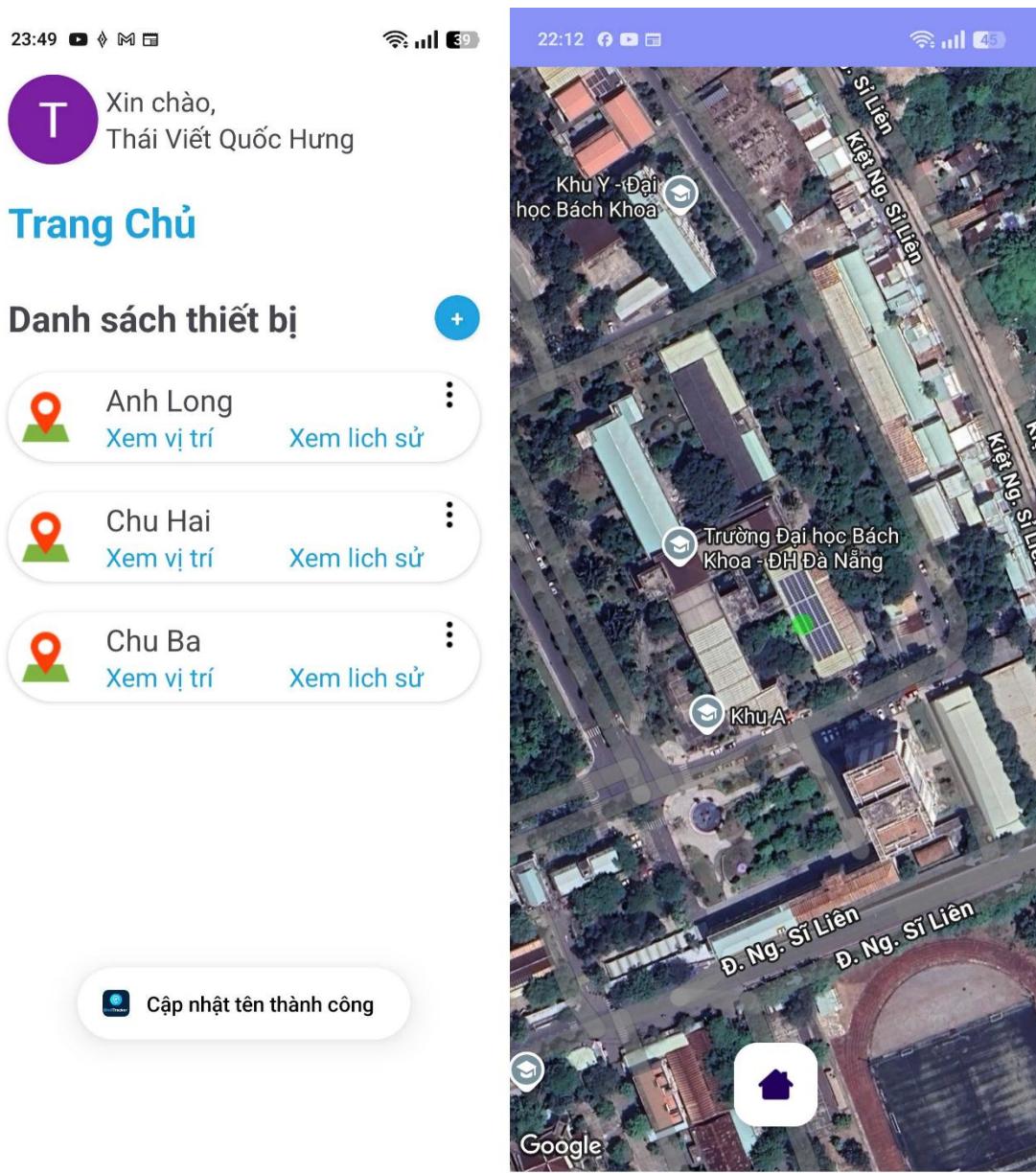
Hình 39: Giao diện chức năng đăng nhập, đăng ký

3.2 Giao diện trang chủ và cài đặt



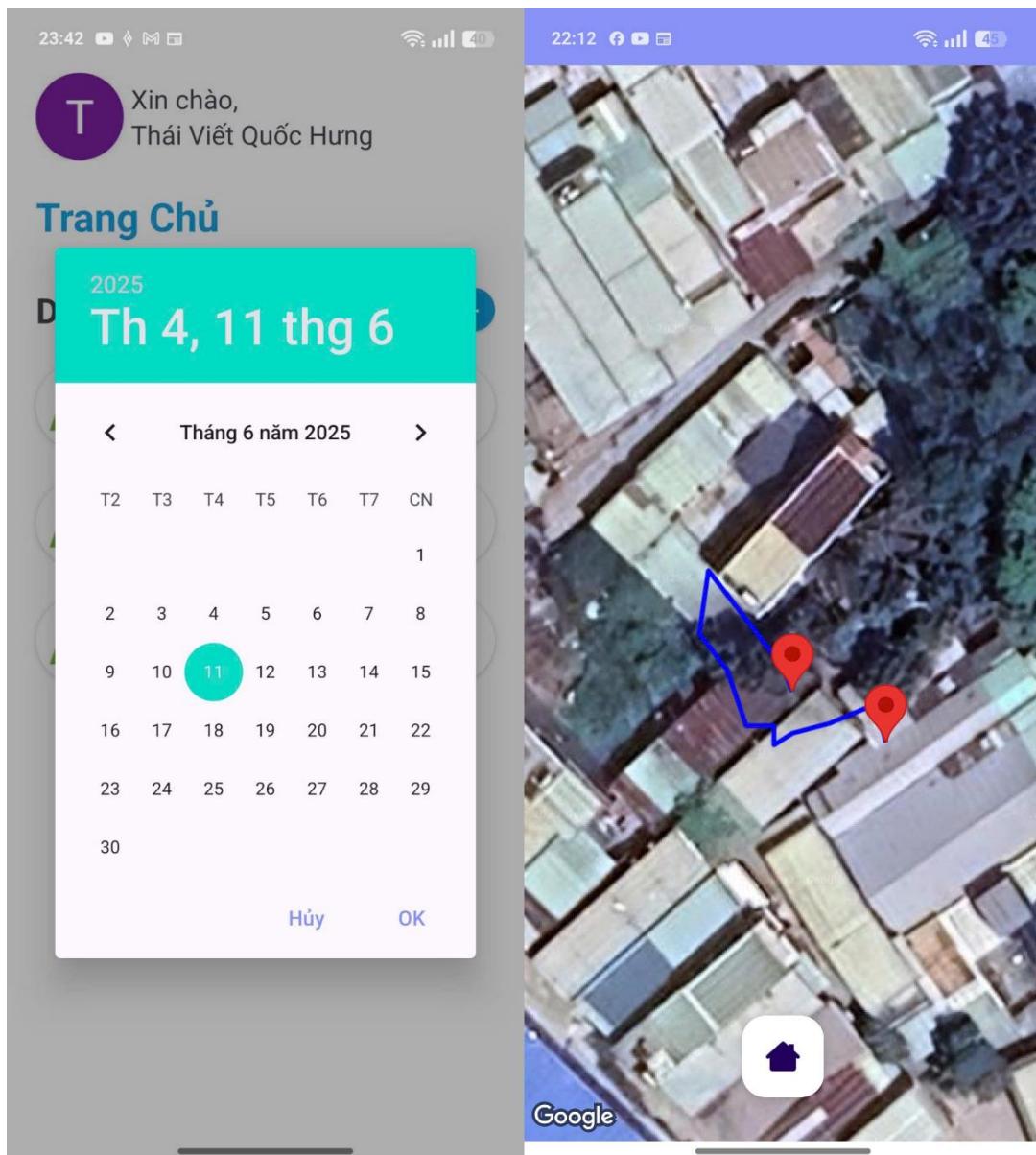
Hình 40: Giao diện trang chủ và cài đặt

3.3 Giao diện thêm thiết bị và hiển thị vị trí của thiết bị đó



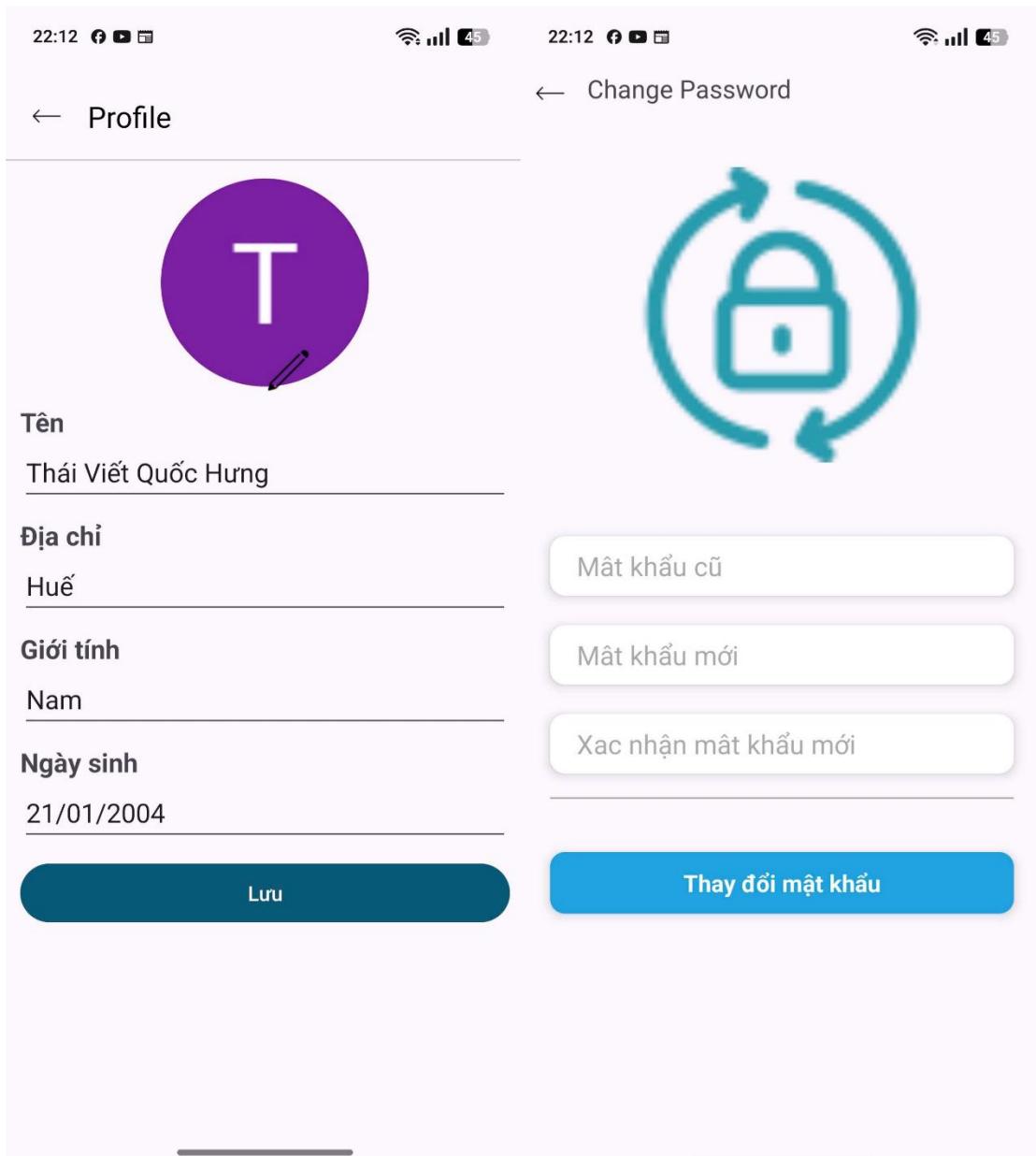
Hình 41: Giao diện thêm thiết bị và hiển thị vị trí của thiết bị đó

3.4 Giao diện xem lịch sử vị trí của thiết bị



Hình 42: Giao diện xem lịch sử vị trí của thiết bị

3.5 Giao diện thông tin cá nhân và đổi mật khẩu



Hình 43: Giao diện thông tin cá nhân và đổi mật khẩu

4. Kết quả mô hình phần cứng



Hình 44: Mô hình phần cứng gậy thông minh



Hình 45: Mô hình phần cứng Thắt lưng cảnh báo té ngã



Hình 46: Mô hình Hệ thống hỗ trợ người khiếm thị thực tế

IV. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1 Kết luận

Hệ thống hỗ trợ người khiếm thị toàn diện của chúng tôi, với sự tích hợp thông minh giữa gậy thông minh và thắt lưng cảnh báo té ngã, đã chứng minh được hiệu quả vượt trội trong việc giải quyết những thách thức lớn mà người khiếm thị phải đối mặt hàng ngày.

- Nâng cao khả năng di chuyển an toàn: Gậy thông minh, với khả năng phát hiện và nhận diện chuồng ngại vật bằng AI cùng cảnh báo âm thanh tức thì. Nó giúp người dùng tránh va chạm, di chuyển mượt mà và tự tin hơn rất nhiều trong môi trường xung quanh.
- Đảm bảo an toàn tối đa: Thắt lưng cảnh báo té ngã, được trang bị cảm biến gia tốc và công nghệ học máy, cung cấp giải pháp bảo vệ chủ động bằng cách phát hiện sớm nguy cơ té ngã và gửi cảnh báo khẩn cấp tức thì đến người thân. Kết hợp với module GPS định vị chính xác, hệ thống mang lại sự an tâm tuyệt đối cho cả người khiếm thị và gia đình họ.
- Hiệu quả đã được kiểm chứng: Qua quá trình thử nghiệm, hệ thống đã đạt được độ chính xác cao trong việc nhận diện vật cản và cảnh báo té ngã kịp thời. Chức năng định vị hoạt động liên tục và chính xác, cùng với ứng dụng điện thoại thân thiện, trực quan, cung cấp thêm giá trị và sự tiện lợi của giải pháp này.

Chúng tôi tin rằng, Hệ thống hỗ trợ người khiếm thị này không chỉ là một sản phẩm công nghệ mà còn là một bước tiến quan trọng hướng tới việc kiến tạo một cuộc sống độc lập, an toàn và hòa nhập hơn cho người khiếm thị.

4.2 Hướng phát triển

Mặc dù hệ thống hiện tại đã rất hiệu quả, tiềm năng phát triển trong tương lai là vô cùng lớn để mang lại những trải nghiệm tốt hơn nữa cho người dùng:

- Nâng cao độ chính xác của mô hình: Tăng cường huấn luyện AI với dữ liệu đa dạng để nhận diện vật cản trong mọi điều kiện ánh sáng và che khuất.
- Hỗ trợ giao tiếp và nhận diện khuôn mặt: Tích hợp camera nhỏ trên gậy và AI để nhận dạng khuôn mặt người quen, hỗ trợ tương tác xã hội.
- Hệ thống dẫn đường chính xác: Kết hợp GPS với các công nghệ định vị tiên tiến khác để đảm bảo dẫn đường chuẩn xác ở mọi môi trường, kể cả trong nhà.
- Phần cứng bền bỉ, tích hợp AI: Nâng cấp thiết bị để có chất lượng hình ảnh và dữ liệu cảm biến tốt hơn, đồng thời tích hợp trực tiếp mô hình AI vào thiết bị. Thiết kế sẽ đảm bảo gọn nhẹ, chắc chắn, chịu mọi điều kiện thời tiết và bền bỉ theo thời gian.

V. TÀI LIỆU THAM KHẢO

- [1] IC ĐÂY RỒI, "Module ESP32-CAM WiFi + bluetooth Camera OV2640," [Online]. Available:<https://icdayroi.com/module-esp32-cam-wifi-bluetooth-camera-ov2640>. [Accessed 29 5 2025].
- [2] IC ĐÂY RỒI, "Mạch Phát Âm Thanh MP3 Kết Hợp Amply DFPlayer Mini," [Online]. Available:<https://icdayroi.com/mach-phat-am-thanh-mp3-ket-hop-amply-dfplayer-mini>. [Accessed 29 5 2025].
- [3] IC ĐÂY RỒI, "Loa 4Ohm 3W," [Online]. Available:<https://icdayroi.com/loa-4ohm-3w>. [Accessed 29 5 2025].
- [4] Linh Kien Viet, "Bo mạch ESP32 DevKit V1 / module ESP-WROOM-32 - S1H15," [Online]. Available:<https://linhkienvietnam.vn/module-node-mcu-32s-esp32-devkitc-dung-module-esp-wroom-32>. [Accessed 29 5 2025].
- [5] E360, "Module GPS NEO-6M V2 Định Vị Chính Xác Cao - Thiết Bị GPS Dẫn Đường Cho Arduino ESP32 STM32," [Online]. Available: <https://dientu360.com/mach-dinh-vi-gps-neo-6m-v2>. [Accessed 29 5 2025].
- [6] HSHOP, "Cảm biến GY-521 6DOF IMU MPU6050," [Online]. Available: <https://hshop.vn/cam-bien-6-dof-bac-tu-do-gy-521-mpu6050>. [Accessed 29 5 2025].
- [7] IC ĐÂY RỒI, "Antenna GPS 1575.42Mhz," [Online]. Available: <https://icdayroi.com/antenna-gps-1575-42mhz>. [Accessed 29 5 2025].
- [8] Last Minute Engineers, "Getting Started With ESP32-CAM," [Online]. Available: <https://lastminuteengineers.com/getting-started-with-esp32-cam>. [Accessed 29 5 2025].
- [9] Random Nerd Tutorials, "Getting Started with the ESP32 Development Board," [Online]. Available: <https://randomnerdtutorials.com/getting-started-with-esp32>. [Accessed 29 5 2025].
- [10] DF Robot, "DFPlayer - A Mini MP3 Player," [Online]. Available: <https://www.dfrobot.com/product-1121.html>. [Accessed 30 5 2025].
- [11] NShopVN, "Cảm Biến gia tốc GY-521 6DOF IMU MPU6050," [Online]. Available: <https://nshopvn.com/product/cam-bien-gia-toc-gy-521-6dof-imu-mpu6050>. [Accessed 30 5 2025].
- [12] BKaii, "Tìm hiểu về truyền thông UART: khái niệm, nguyên lý và ứng dụng," [Online]. Available: <https://bkaii.com.vn/tin-tuc/1099-tim-hieu-ve-truyen-thong-uart-khai-niem-nguyen-ly-va-ung-dung>. [Accessed 29 5 2025].
- [13] Dien Tu Tuong Lai, "Chuẩn giao tiếp I2C là gì," [Online]. Available: <https://dientutuonglai.com/chuan-giao-tiep-i2c-la-gi.html>. [Accessed 29 5 2025].
- [14] Ultralytics, "Ultralytics YOLO11," [Online]. Available: <https://docs.ultralytics.com/vi/models/yolo11>. [Accessed 15 1 2025].
- [15] Deep AI KhanhBlog, "Giới thiệu về SVM," [Online]. Available: https://phamduinhkhanh.github.io/deepai-book/index_SVM.html. [Accessed 20 4 2025].
- [16] Google, "Firebase Authentication," [Online]. Available: <https://firebase.google.com/docs/auth?hl=vi>. [Accessed 29 5 2025].
- [17] Google, "Firebase Documentation," [Online]. Available: <https://firebase.google.com/docs>. [Accessed 29 05 2025].
- [18] "Last Minute Engineers," [Online]. Available: <https://lastminuteengineers.com/getting-started-with-esp32-cam/>. [Accessed 29 5 2025].
- [19] "Random Nerd Tutorials," [Online]. Available: <https://randomnerdtutorials.com/getting-started-with-esp32/>. [Accessed 29 5 2025].

