

ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN – ĐIỆN TỬ
BỘ MÔN ĐIỆN TỬ

-----o0o-----



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

PHÁT HIỆN TẾ NGÃ
BẰNG MÔ HÌNH HỌC MÁY TRÊN FPGA

GVHD: ThS. Trần Hoàng Quân

SVTH: Phạm Nguyễn Hoàng Tuấn

MSSV: 2112588

TP. HỒ CHÍ MINH, THÁNG 4 NĂM 2025

----- ☆ -----

----- ☆ -----

Số: _____/BKĐT
Khoa: **Điện – Điện tử**
Bộ Môn: **Điện Tử**

NHIỆM VỤ LUẬN VĂN TỐT NGHIỆP

1. HỌ VÀ TÊN: PHẠM NGUYỄN HOÀNG TUẤN MSSV: 2112588
 2. NGÀNH: **ĐIỆN TỬ - VIỄN THÔNG** LỚP : DD21DV4
 3. Đề tài: Phát hiện té ngã bằng mô hình học máy trên fpga.
 4. Nhiệm vụ: Xây dựng một hệ thống tự động, hiệu quả để phát hiện các sự kiện té ngã trong thời gian thực, tận dụng sức mạnh tính toán của FPGA và khả năng phân tích thông minh của học máy. Hệ thống hướng đến việc tối ưu hóa tốc độ xử lý, độ chính xác và tiết kiệm tài nguyên, nhằm ứng dụng trong các lĩnh vực như giám sát sức khỏe người cao tuổi, an ninh và y tế thông minh.
 5. Ngày giao nhiệm vụ luận văn: ngày 4 tháng 3 năm 2025
 6. Ngày hoàn thành nhiệm vụ:
 7. Họ và tên người hướng dẫn: ThS. Trần Hoàng Quân Phân hướng dẫn
- Nội dung và yêu cầu LVTN đã được thông qua Bộ Môn.

Tp.HCM, ngày..... tháng..... năm 2025
CHỦ NHIỆM BỘ MÔN

NGƯỜI HƯỚNG DẪN CHÍNH

PHẦN DÀNH CHO KHOA, BỘ MÔN:

Người duyệt (chấm sơ bộ):.....
Đơn vị:.....
Ngày bảo vệ :
Điểm tổng kết:
Nơi lưu trữ luận văn:

LỜI CẢM ƠN

Trước tiên, em xin bày tỏ lòng biết ơn sâu sắc và chân thành nhất đến **Thầy Trần Hoàng Quân** – giảng viên hướng dẫn đã tận tâm chỉ bảo, theo sát và hỗ trợ em trong suốt quá trình thực hiện đề tài luận văn tốt nghiệp. Thầy không chỉ là người định hướng chuyên môn, mà còn luôn nhiệt tình chia sẻ kinh nghiệm, truyền cảm hứng nghiên cứu và đồng hành cùng em vượt qua những khó khăn trong suốt quá trình học tập và triển khai đề tài.

Em cũng xin gửi lời tri ân chân thành đến **quý thầy cô Trường Đại học Bách Khoa – Đại học Quốc gia TP. Hồ Chí Minh**, đặc biệt là **khoa Kỹ thuật Điện tử**, nơi đã chấp cánh cho em hành trang kiến thức, kỹ năng chuyên ngành và thái độ làm việc chuyên nghiệp. Những kiến thức quý báu được truyền đạt từ quý thầy cô chính là nền tảng vững chắc để em hoàn thành luận văn này.

Trong suốt quá trình thực hiện đề tài, em đã cố gắng hết sức mình để vận dụng những gì đã học, đồng thời chủ động nghiên cứu thêm tài liệu, tham khảo nhiều nguồn khác nhau nhằm hoàn thiện luận văn một cách tốt nhất. Tuy nhiên, do giới hạn về thời gian, kinh nghiệm thực tiễn còn hạn chế, nên luận văn chắc chắn không tránh khỏi những thiếu sót. Em kính mong nhận được sự góp ý quý báu từ quý thầy cô và các anh chị đi trước để em có cơ hội hoàn thiện hơn trong tương lai.

Bên cạnh sự đồng hành từ nhà trường và thầy cô, em cũng không thể không nhắc đến **gia đình thân yêu** – những người luôn là chỗ dựa tinh thần vững chắc, luôn tin tưởng, động viên và tạo điều kiện tốt nhất để em an tâm học tập. Đồng thời, em cũng xin cảm ơn **các bạn bè, đồng môn** đã chia sẻ, hỗ trợ em trong học tập, nghiên cứu và cả trong những lúc khó khăn.

Một lần nữa, em xin trân trọng cảm ơn tất cả những ai đã đồng hành, giúp đỡ và truyền động lực cho em trong suốt hành trình học tập và thực hiện luận văn tốt nghiệp này.

TP. Hồ Chí Minh, ngày 25 tháng 4 năm 2025

Sinh viên thực hiện

Phạm Nguyễn Hoàng Tuấn

TÓM TẮT LUẬN VĂN

Đề tài luận văn "Phát hiện té ngã bằng mô hình học máy trên FPGA" tập trung vào việc xây dựng một hệ thống phát hiện té ngã cho người cao tuổi, nhằm nâng cao chất lượng chăm sóc sức khỏe và giảm thiểu hậu quả do té ngã. Hệ thống này sử dụng các cảm biến quán tính (gia tốc kế và con quay hồi chuyển) để thu thập dữ liệu về chuyển động của người sử dụng. Dữ liệu cảm biến sau đó được xử lý và phân tích bằng các thuật toán học máy như Support Vector Machine (SVM) và Decision Tree để nhận diện và phân loại các tình huống té ngã.

Hệ thống được triển khai trên nền tảng FPGA, cụ thể là Kit DE0 Nano, nhờ vào khả năng xử lý song song và thời gian thực của FPGA. Việc sử dụng FPGA không chỉ giúp giảm thiểu độ trễ trong quá trình phát hiện mà còn tiết kiệm năng lượng, tạo điều kiện cho hệ thống hoạt động lâu dài mà không cần thay đổi pin thường xuyên.

Để xây dựng hệ thống, nghiên cứu này bao gồm các bước chính: thu thập và xử lý dữ liệu cảm biến, phát triển các mô hình học máy, triển khai các mô hình học máy trên FPGA, và kiểm thử hệ thống trong môi trường thực tế. Các kết quả thử nghiệm cho thấy hệ thống có khả năng phát hiện chính xác các tình huống té ngã với độ chính xác cao, đồng thời có thể hoạt động hiệu quả trong thời gian thực.

Kết quả của nghiên cứu này đóng góp vào việc ứng dụng các công nghệ hiện đại như FPGA và học máy vào lĩnh vực chăm sóc sức khỏe người cao tuổi, đồng thời mở ra khả năng phát triển các hệ thống thông minh trong các ứng dụng y tế và an ninh khác.

MỤC LỤC

1. GIỚI THIỆU.....	1
1.1 Tổng quan.....	1
1.2 Tình hình nghiên cứu trong và ngoài nước	1
1.3 Nhiệm vụ luận văn.....	3
2. LÝ THUYẾT	5
Chương 1: Tổng quan về phát hiện té ngã và ứng dụng thực tiễn	5
1.1 Giới thiệu chung về tình trạng té ngã và ảnh hưởng đến sức khỏe người cao tuổi	5
1.2 Các nguyên nhân phổ biến gây ra té ngã	5
1.3 Phân loại các phương pháp phát hiện té ngã.....	5
1.4 Vai trò và ý nghĩa của việc xây dựng hệ thống phát hiện té ngã thời gian thực trong lĩnh vực chăm sóc sức khỏe	6
Chương 2: Cảm biến quán tính và dữ liệu phục vụ phát hiện té ngã	7
2.1 Tổng quan về bộ dữ liệu SisFall	7
2.2 Tiền xử lý dữ liệu	9
2.2.1 Lọc nhiễu và làm mịn dữ liệu	9
2.2.2 Chuẩn hóa dữ liệu	9
2.2.3 Trích xuất đặc trưng	10
2.3 Xây dựng mô hình SVM trên MATLAB.....	12
2.3.1 Phương pháp huấn luyện và kiểm thử.....	12
2.3.2 Chuyển đổi từ dữ liệu float sang dạng fixed-point (Q15)	13
2.4 Đánh giá hiệu năng mô hình	14
2.4.1 Kết quả kiểm tra trên dữ liệu thử nghiệm	14
2.4.2 Phân tích độ chính xác và sai số của mô hình	15
3. THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG	17
3.1 Tổng quan hệ thống phần cứng	17
3.2 Khối giao tiếp I2C.....	18

3.2.1. Tổng quan khối giao tiếp I2C.....	18
3.2.2. Sơ đồ khối giao tiếp I2C.....	18
3.3 Khối xử lý đặc trưng (Feature Extraction Pipeline)	20
3.4 Khối suy luận SVM (svm_inference.v).....	21
3.5 Điều khiển tổng thể.....	22
3.6 Tích hợp toàn hệ thống.....	23
4. KẾT QUẢ THỰC HIỆN	24
5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	24
5.1 Kết luận	24
5.2 Hướng phát triển.....	24
6. TÀI LIỆU THAM KHẢO	26

DANH SÁCH HÌNH MINH HỌA

Hình 1: Ma trận nhầm lẫn của mô hình SVM	15
Hình 2: DE0-Nano Development and Education Board	17
Hình 3: Cảm biến GY-521 6DOF IMU MPU6050.....	17
Hình 4: Sơ đồ khối của hệ thống giao tiếp I2C giữa kit DE0-Nano và cảm biến MPU-6050	18
Hình 5: Sơ đồ khối xử lý đặc trưng.....	20
Hình 6: Sơ đồ khối fall detection.....	23

1. GIỚI THIỆU

1.1 Tổng quan

Trong những năm gần đây, việc phát hiện té ngã tự động đã trở thành một vấn đề quan trọng trong các lĩnh vực giám sát sức khỏe, an ninh và y tế thông minh, đặc biệt là đối với người cao tuổi. Té ngã là một trong những nguyên nhân chính gây ra thương tích nghiêm trọng và tử vong ở người cao tuổi, do đó việc phát hiện sớm các sự kiện này có thể giúp giảm thiểu rủi ro và cải thiện chất lượng cuộc sống cho đối tượng này. Để giải quyết vấn đề này, các hệ thống tự động phát hiện té ngã được nghiên cứu và phát triển, sử dụng các công nghệ như cảm biến chuyển động, máy học và phần cứng đặc biệt như FPGA (Field Programmable Gate Array).

FPGA là một loại phần cứng có khả năng lập trình linh hoạt, cho phép thực thi các thuật toán phức tạp với tốc độ cao và tiêu thụ năng lượng thấp, phù hợp với các ứng dụng yêu cầu xử lý thời gian thực như phát hiện té ngã. Hệ thống này có thể sử dụng các cảm biến như gia tốc kế và con quay hồi chuyển để theo dõi chuyển động của cơ thể, từ đó phát hiện các dấu hiệu bất thường như té ngã. Đồng thời, mô hình học máy có thể phân tích dữ liệu cảm biến và phân loại các sự kiện té ngã từ các hành động bình thường, nhờ vào khả năng học từ dữ liệu và cải thiện độ chính xác theo thời gian.

1.2 Tình hình nghiên cứu trong và ngoài nước

Phát hiện té ngã là một chủ đề nghiên cứu quan trọng trong lĩnh vực chăm sóc sức khỏe, đặc biệt đối với người cao tuổi, vì té ngã là nguyên nhân chủ yếu dẫn đến các chấn thương nghiêm trọng và tử vong trong nhóm đối tượng này. Trong thời gian qua, nhiều nghiên cứu đã được thực hiện nhằm phát triển các hệ thống phát hiện té ngã hiệu quả bằng cách sử dụng công nghệ cảm biến và các thuật toán học máy.

Nghiên cứu trong nước: Tại Việt Nam, vấn đề phát hiện té ngã ở người cao tuổi đang dần được chú trọng, đặc biệt là với sự phát triển của các hệ thống chăm sóc sức khỏe thông minh. Các nghiên cứu trong nước thường tập trung vào việc áp dụng các cảm biến gia tốc và con quay hồi chuyển để theo dõi chuyển động của cơ thể. Một số nghiên cứu đã áp dụng các phương pháp truyền thống như ngưỡng gia tốc để phát hiện té ngã, nhưng vẫn còn gặp phải vấn đề về cảnh báo giả (false alarm) khi có các hoạt động tương tự té ngã.

Nghiên cứu quốc tế: Ở cấp độ quốc tế, các nghiên cứu đã áp dụng nhiều công nghệ và phương pháp khác nhau để cải thiện độ chính xác của các hệ thống phát hiện té ngã. Một số nghiên cứu sử dụng các hệ thống cảm biến đeo trên cơ thể, bao gồm cảm biến gia tốc và con quay hồi chuyển, để thu thập dữ liệu và sử dụng các thuật toán học máy như Support Vector Machine (SVM) và Decision Tree để phân loại các tình huống té ngã. Các nghiên cứu này chủ yếu tập trung vào việc giảm thiểu số lượng cảnh báo giả và cải thiện độ chính xác của các mô hình. Nghiên cứu của Thiago B. Rodrigues et al. (2018) đã chỉ ra rằng thuật toán Decision Tree mang lại hiệu quả cao với độ chính xác lên đến 95,87% trong việc phát hiện té ngã, cao hơn so với SVM.

Nhận xét về phương pháp và kết quả nghiên cứu: Các nghiên cứu đã thực hiện so sánh hiệu quả của nhiều thuật toán học máy khác nhau để phát hiện té ngã, bao gồm SVM, Decision Tree, và k-Nearest Neighbor (kNN). Các nghiên cứu này cho thấy rằng thuật toán Decision Tree có thể đạt được độ chính xác cao hơn, trong khi SVM tuy hiệu quả nhưng có thời gian huấn luyện dài hơn. Hầu hết các hệ thống phát hiện té ngã đều có sự phụ thuộc vào dữ liệu huấn luyện chất lượng cao, và khả năng của mô hình sẽ giảm sút nếu dữ liệu huấn luyện không đầy đủ hoặc không đại diện được các tình huống thực tế.

Những hướng nghiên cứu tiếp theo: Trong tương lai, nhiều nghiên cứu sẽ tập trung vào việc cải tiến các mô hình học máy với bộ dữ liệu lớn hơn và đa dạng hơn, để giảm thiểu cảnh báo giả và tăng độ chính xác trong môi trường thực tế. Các nghiên cứu cũng đang hướng tới việc phát triển các hệ thống phát hiện té ngã có khả năng tự động điều chỉnh và học từ dữ liệu cá nhân của người sử dụng, nhằm cung cấp một giải pháp an toàn và phù hợp hơn cho từng người cao tuổi.

Tóm lại, các nghiên cứu trong và ngoài nước đều chỉ ra rằng việc sử dụng cảm biến và học máy trong phát hiện té ngã là một giải pháp tiềm năng, với những bước tiến đáng kể trong việc giảm thiểu các cảnh báo giả và nâng cao độ chính xác của hệ thống. Tuy nhiên, vấn đề về dữ liệu huấn luyện và tính khả dụng của các hệ thống này trong thực tế vẫn còn là những thách thức cần tiếp tục nghiên cứu và cải thiện.

1.3 Nhiệm vụ luận văn

Đề tài luận văn "**Phát hiện té ngã bằng mô hình học máy trên FPGA**" thực hiện các nhiệm vụ chính sau đây:

Nội dung 1: Tìm hiểu lý thuyết về phát hiện té ngã và các mô hình học máy sử dụng trong phát hiện té ngã

- Yêu cầu: Nghiên cứu các nguyên lý cơ bản của việc phát hiện té ngã, các dấu hiệu nhận biết và phân biệt té ngã với các hoạt động hàng ngày.
- Kết quả cần đạt: Hiểu rõ các trạng thái té ngã, đặc điểm và dấu hiệu nhận dạng thông qua cảm biến quán tính, phân tích được ưu và nhược điểm của các thuật toán học máy như SVM, Decision Tree, KNN.
- Cách tiếp cận: Tham khảo các tài liệu nghiên cứu, bài báo khoa học, sách chuyên ngành liên quan đến các kỹ thuật và mô hình học máy để phát hiện té ngã.

Nội dung 2: Tìm hiểu về Kit DE0 Nano

- Yêu cầu: Nghiên cứu cấu tạo, nguyên lý hoạt động, thông số kỹ thuật và khả năng xử lý dữ liệu của Kit DE0 Nano.
- Kết quả cần đạt: Nắm vững kiến trúc, khả năng lập trình và giới hạn kỹ thuật của Kit DE0 Nano trong việc ứng dụng các thuật toán học máy.
- Cách tiếp cận: Thực hiện thực nghiệm, lập trình một số ví dụ cơ bản trên Kit DE0 Nano, sử dụng tài liệu hướng dẫn và các nguồn tài liệu chính thức từ nhà sản xuất.

Nội dung 3: Xây dựng mô hình học máy để phát hiện té ngã

- Yêu cầu: Thiết kế và xây dựng một mô hình học máy để phát hiện té ngã dựa trên dữ liệu cảm biến quán tính.
- Kết quả cần đạt: Xây dựng được một mô hình học máy hoàn chỉnh (sử dụng thuật toán tối ưu như Decision Tree hoặc SVM), với độ chính xác cao, khả năng phân biệt rõ ràng giữa trạng thái té ngã và trạng thái bình thường.
- Cách tiếp cận: Thu thập dữ liệu từ cảm biến, xử lý dữ liệu, lựa chọn thuật toán, huấn luyện mô hình, đánh giá và điều chỉnh mô hình để đạt hiệu suất tối ưu.

Nội dung 4: Cài đặt và tối ưu hóa mô hình học máy trên FPGA (Kit DE0 Nano)

- Yêu cầu: Chuyển đổi và tối ưu mô hình học máy đã xây dựng để triển khai lên FPGA trên Kit DE0 Nano, đảm bảo yêu cầu về tốc độ xử lý và độ chính xác.
- Kết quả cần đạt: Triển khai thành công và hoạt động hiệu quả mô hình phát hiện té ngã trên FPGA, tối ưu hóa tốc độ xử lý thời gian thực, tiết kiệm tài nguyên phần cứng.
- Cách tiếp cận: Sử dụng các công cụ như Quartus Prime, tối ưu hóa code VHDL/Verilog, kiểm thử mô hình trong môi trường thực tế và điều chỉnh các tham số để đảm bảo hiệu suất hoạt động.

Giới hạn đề tài:

- Hệ thống chỉ phát hiện được té ngã dựa trên dữ liệu từ cảm biến quán tính (gia tốc, con quay hồi chuyển).
- Phạm vi thử nghiệm chủ yếu trong môi trường phòng thí nghiệm hoặc môi trường mô phỏng tương tự thực tế.
- Không mở rộng các tính năng như cảnh báo từ xa hay kết nối với hệ thống lớn hơn trong khuôn khổ luận văn này.

2. LÝ THUYẾT

Chương 1: Tổng quan về phát hiện té ngã và ứng dụng thực tiễn

1.1 Giới thiệu chung về tình trạng té ngã và ảnh hưởng đến sức khỏe người cao tuổi

Té ngã là một trong những nguyên nhân chính dẫn đến các vấn đề sức khỏe nghiêm trọng, đặc biệt là đối với người cao tuổi. Theo các nghiên cứu, té ngã là nguyên nhân hàng đầu gây tử vong và thương tật cho người lớn tuổi, chiếm tỷ lệ cao trong các vụ tai nạn không tử vong. Mỗi năm, khoảng 30% người cao tuổi từ 65 tuổi trở lên trải qua ít nhất một lần té ngã, và số liệu này có xu hướng tăng lên khi dân số người cao tuổi tăng trưởng. Té ngã không chỉ gây ra các chấn thương như gãy xương, đặc biệt là gãy xương hông, mà còn làm giảm khả năng vận động và gây ra các vấn đề tâm lý như lo âu, trầm cảm. Điều này ảnh hưởng trực tiếp đến chất lượng cuộc sống của người cao tuổi và gánh nặng cho gia đình và xã hội.

1.2 Các nguyên nhân phổ biến gây ra té ngã

Té ngã có thể xảy ra do nhiều nguyên nhân, trong đó có cả yếu tố sinh lý, tâm lý và môi trường. Các yếu tố sinh lý bao gồm sự suy giảm sức khỏe về cơ và xương, rối loạn thăng bằng và giảm thị lực, là những vấn đề phổ biến ở người cao tuổi. Yếu tố tâm lý như rối loạn lo âu, trầm cảm và thiếu tự tin cũng làm tăng nguy cơ té ngã do hạn chế khả năng vận động.

Ngoài ra, các yếu tố môi trường cũng đóng một vai trò quan trọng. Những môi trường không an toàn như sàn trơn, thiếu ánh sáng, hoặc các vật cản trong nhà có thể gây ra tai nạn té ngã. Đặc biệt, khi người cao tuổi sử dụng thuốc làm giảm thăng bằng hoặc huyết áp không ổn định, nguy cơ té ngã càng cao.

1.3 Phân loại các phương pháp phát hiện té ngã

Hiện nay, có ba phương pháp chính được sử dụng để phát hiện té ngã: phương pháp dựa trên hình ảnh (Vision-based), cảm biến môi trường và cảm biến đeo.

- **Phương pháp Vision-based:** Sử dụng các camera cố định để theo dõi chuyển động của người, sau đó áp dụng các thuật toán xử lý hình ảnh để nhận diện và phân tích động tác té ngã. Phương pháp này có ưu điểm là khả năng nhận dạng chi tiết các chuyển động phức tạp và độ trễ thấp, nhưng lại có nhược điểm là chi phí lắp đặt cao, cần không gian sử dụng cố định và vấn đề về quyền riêng tư.

- **Phương pháp cảm biến môi trường:** Sử dụng các cảm biến đặt trong môi trường xung quanh như cảm biến hồng ngoại hoặc cảm biến áp suất để phát hiện sự thay đổi trong môi trường khi người cao tuổi ngã. Tuy nhiên, phương pháp này dễ gặp phải các lỗi âm tính giả, khi một té ngã không được phát hiện.
- **Phương pháp cảm biến đeo:** Đây là phương pháp phổ biến nhất hiện nay, sử dụng các thiết bị đeo trên cơ thể như cảm biến gia tốc (accelerometers) và cảm biến con quay hồi chuyển (gyroscopes) để theo dõi chuyển động của cơ thể. Phương pháp này có ưu điểm là chi phí thấp, dễ sử dụng và có thể sử dụng được cả trong nhà và ngoài trời, đồng thời giúp thu thập thêm thông tin về sinh lý của người sử dụng như nhịp tim, huyết áp .

1.4 Vai trò và ý nghĩa của việc xây dựng hệ thống phát hiện té ngã thời gian thực trong lĩnh vực chăm sóc sức khỏe

Hệ thống phát hiện té ngã thời gian thực có ý nghĩa đặc biệt quan trọng trong việc cải thiện chất lượng cuộc sống của người cao tuổi. Những hệ thống này không chỉ giúp phát hiện các sự kiện té ngã kịp thời mà còn có thể gửi cảnh báo ngay lập tức đến người thân hoặc các cơ sở y tế, giúp giảm thiểu thiệt hại về sức khỏe do không có sự can thiệp kịp thời.

Hệ thống này cũng đóng vai trò quan trọng trong việc giảm gánh nặng cho các dịch vụ y tế và chăm sóc sức khỏe, đồng thời tạo ra một môi trường sống an toàn hơn cho người cao tuổi, đặc biệt là những người sống một mình. Việc phát triển các hệ thống này còn giúp giảm thiểu chi phí điều trị các chấn thương do té ngã, đồng thời hỗ trợ quá trình phục hồi và chăm sóc bệnh nhân sau khi té ngã.

Việc ứng dụng công nghệ học máy trong các hệ thống phát hiện té ngã giúp cải thiện độ chính xác và giảm thiểu các cảnh báo giả, từ đó nâng cao hiệu quả của các hệ thống này trong việc bảo vệ người cao tuổi khỏi những rủi ro không mong muốn .

Chương 2: Cảm biến quán tính và dữ liệu phục vụ phát hiện té ngã

2.1 Tổng quan về bộ dữ liệu SisFall

Trong các nghiên cứu liên quan đến phát hiện té ngã, việc lựa chọn một bộ dữ liệu phù hợp để huấn luyện và đánh giá hiệu năng của mô hình đóng vai trò hết sức quan trọng. Bộ dữ liệu SisFall (Systems Integrated Sensors Fall Detection Dataset) được phát triển bởi các nhà nghiên cứu từ Đại học Antioquia, Colombia, nhằm mục đích hỗ trợ nghiên cứu trong lĩnh vực phát hiện té ngã, đặc biệt hướng đến ứng dụng cho người cao tuổi – đối tượng dễ bị tổn thương do té ngã.

SisFall là bộ dữ liệu phong phú và toàn diện, thu thập từ các cảm biến chuyển động với độ chính xác cao, giúp phản ánh chân thực nhất các tình huống té ngã cũng như các hoạt động sinh hoạt thường ngày của con người. Việc sử dụng bộ dữ liệu này cho phép xây dựng và phát triển những hệ thống phát hiện té ngã có độ tin cậy cao và phù hợp với thực tế.

Bộ dữ liệu SisFall bao gồm tổng cộng **4.503 mẫu dữ liệu**, trong đó có **2.707 mẫu liên quan đến các hoạt động hàng ngày (ADLs)** và **1.798 mẫu liên quan đến các tình huống té ngã (Falls)**. Các dữ liệu được thu thập từ **38 tình nguyện viên**, trong đó có **19 nam và 19 nữ**, độ tuổi trải rộng từ **19 đến 75 tuổi**, bao gồm cả người trẻ và người cao tuổi, nhằm đảm bảo tính đại diện cao và phù hợp với thực tế sử dụng.

Bộ dữ liệu bao gồm các loại hoạt động cụ thể như sau:

- **Các hoạt động sinh hoạt hàng ngày (ADLs - Activities of Daily Living):**
 - Đi bộ thông thường, đi lên/xuống cầu thang, đứng lên, ngồi xuống.
 - Các hoạt động vận động mạnh hơn như chạy bộ, nhảy, cúi người, vào và ra khỏi phương tiện giao thông (ô tô), cũng được ghi lại nhằm tránh nhầm lẫn giữa hoạt động mạnh và tình trạng té ngã thật sự.
- **Các tình huống té ngã (Falls):**
 - Té ngã theo các hướng khác nhau như: té về phía trước, té về phía sau, té sang trái/phải.
 - Té ngã trong các trạng thái khác nhau như đứng, ngồi, di chuyển, và cả tình huống mất thăng bằng bất ngờ.

- Té ngã với mức độ nghiêm trọng khác nhau (mạnh, nhẹ) nhằm cung cấp sự đa dạng về các đặc trưng chuyển động khi ngã.

Các dữ liệu được thu thập từ một thiết bị cảm biến tự chế tạo, đeo tại vị trí **thắt lưng người tham gia**, vị trí trung tâm cơ thể phản ánh rõ nhất trạng thái té ngã. Thiết bị này được trang bị **hai cảm biến gia tốc 3 trục và một cảm biến con quay hồi chuyển 3 trục**, ghi nhận dữ liệu chuyển động với tần số lấy mẫu cao (200Hz), đảm bảo cung cấp thông tin chi tiết và chính xác về chuyển động của cơ thể.

Dữ liệu được tổ chức dưới dạng các tập tin văn bản (.txt), mỗi tập tin tương ứng với một lần thực hiện hoạt động hoặc tình huống té ngã cụ thể. Trong mỗi tập tin, dữ liệu bao gồm thông tin theo thời gian và các thông số từ cảm biến như gia tốc theo các trục x, y, z và tốc độ góc của cơ thể, giúp cho quá trình xử lý và phân tích thuận lợi hơn.

Bộ dữ liệu SisFall được thiết kế nhằm phục vụ trực tiếp cho các nghiên cứu phát hiện té ngã bằng các mô hình máy học và học sâu. Với lượng dữ liệu phong phú, chi tiết, và đa dạng về các tình huống té ngã cũng như hoạt động sinh hoạt, SisFall đặc biệt phù hợp cho các nghiên cứu:

- **Huấn luyện các mô hình máy học:** cung cấp cơ sở dữ liệu chuẩn xác để đào tạo và phát triển các mô hình phân loại có khả năng phân biệt rõ ràng giữa tình trạng té ngã và các hoạt động hàng ngày.
- **Kiểm thử hiệu năng hệ thống:** cung cấp dữ liệu đầy đủ và chi tiết để đánh giá, so sánh hiệu năng của các hệ thống phát hiện té ngã, đặc biệt khi ứng dụng trên các hệ thống nhúng như FPGA.
- **Xây dựng các ứng dụng thực tế:** hỗ trợ triển khai các hệ thống thời gian thực nhằm phát hiện té ngã chính xác, nhanh chóng, và tin cậy, góp phần quan trọng trong việc bảo vệ an toàn và sức khỏe cho người dùng, đặc biệt là người cao tuổi.

2.2 Tiền xử lý dữ liệu

Tiền xử lý dữ liệu là một bước rất quan trọng trong bất kỳ hệ thống phân loại nào sử dụng các phương pháp học máy, đặc biệt là các hệ thống dựa vào dữ liệu cảm biến như hệ thống phát hiện té ngã. Đối với đề tài này, dữ liệu được thu nhận từ cảm biến gia tốc và con quay hồi chuyển (MPU-6050), vì vậy dữ liệu thu nhận được có thể chứa nhiều nhiễu và sai lệch. Việc thực hiện các bước lọc nhiễu, chuẩn hóa và trích xuất đặc trưng sẽ đảm bảo dữ liệu được đưa vào mô hình SVM có chất lượng tốt nhất, tăng cường khả năng phân biệt rõ ràng giữa các trường hợp té ngã và các hoạt động thông thường.

2.2.1 Lọc nhiễu và làm mịn dữ liệu

Dữ liệu thô thu thập từ cảm biến thường chứa nhiễu gây ra bởi chuyển động nhỏ, rung động từ môi trường, hoặc nhiễu điện tử của cảm biến. Do đó, bước đầu tiên trong quá trình tiền xử lý là lọc nhiễu và làm mịn dữ liệu.

Trong đề tài này, dữ liệu cảm biến được xử lý bằng bộ lọc trung bình trượt (Moving Average Filter) với kích thước cửa sổ 5 mẫu liên tiếp nhằm giảm thiểu nhiễu ngẫu nhiên và tăng tính ổn định cho tín hiệu. Bộ lọc này được chọn nhờ khả năng loại bỏ nhiễu hiệu quả, đơn giản trong cài đặt và ít tốn tài nguyên tính toán, rất phù hợp với các hệ thống nhúng như FPGA.

Công thức lọc trung bình trượt được sử dụng như sau:

$$x_{smooth}(n) = \frac{1}{M} \sum_{k=0}^{M-1} x(n-k)$$

Trong đó:

- $x_{smooth}(n)$ là giá trị dữ liệu cảm biến đã làm mịn tại thời điểm n .
- $x(n-k)$ là dữ liệu cảm biến thô tại thời điểm $n-k$.
- M là kích thước cửa sổ lọc (trong đề tài này chọn $M=5$).

2.2.2 Chuẩn hóa dữ liệu

Sau khi lọc nhiễu, dữ liệu cảm biến được chuẩn hóa để đưa về một thang đo chung nhằm tránh sự thiên lệch trong quá trình huấn luyện mô hình SVM. Kỹ thuật chuẩn hóa được

sử dụng là chuẩn hóa theo trung bình và độ lệch chuẩn (Z-score Normalization). Công thức tính toán chuẩn hóa dữ liệu như sau:

$$x_{norm} = \frac{x - \mu}{\sigma}$$

Trong đó:

- x_{norm} là giá trị dữ liệu sau chuẩn hóa.
- x là giá trị ban đầu của dữ liệu cảm biến.
- μ là giá trị trung bình của toàn bộ tập dữ liệu.
- σ là độ lệch chuẩn của toàn bộ tập dữ liệu.

Quá trình chuẩn hóa này đảm bảo các giá trị đầu vào của mô hình SVM nằm trong phạm vi ổn định, giúp mô hình học tập hiệu quả và giảm nguy cơ quá khớp (overfitting).

2.2.3 Trích xuất đặc trưng

Việc lựa chọn các đặc trưng quan trọng và có khả năng phân biệt tốt giữa trạng thái té ngã và các hoạt động bình thường là một bước rất quan trọng trong xây dựng hệ thống phát hiện té ngã. Đối với dữ liệu thu thập từ cảm biến MPU-6050, luận văn này đã lựa chọn và trích xuất các đặc trưng quan trọng bao gồm:

- **Giá trị trung bình (Mean):** Thể hiện mức độ trung tâm của tín hiệu cảm biến trong một khoảng thời gian nhất định, được tính bằng công thức:

$$Mean = \frac{1}{N} \sum_{i=1}^N x_i$$

- **Phương sai (Variance):** Phản ánh sự thay đổi của tín hiệu, cho biết mức độ dao động trong khoảng thời gian quan sát, được tính bằng công thức:

$$Variance = \frac{1}{N} \sum_{i=1}^N (x_i - Mean)^2$$

- **Độ lệch chuẩn (Standard Deviation):** Là căn bậc hai của phương sai, thể hiện độ phân tán dữ liệu rõ ràng hơn về mặt giá trị vật lý, công thức như sau:

$$Std.Dev = \sqrt{Variance}$$

- **Giá trị độ lớn (Magnitude):** Được tính từ cả ba trục gia tốc, phản ánh mức độ cường độ chuyển động chung của người dùng. Giá trị này thường tăng lên đáng kể trong các trường hợp té ngã. Độ lớn được tính toán như sau:

$$Magnitude = \sqrt{Accel_x^2 + Accel_y^2 + Accel_z^2}$$

Các đặc trưng này được trích xuất từ dữ liệu đã được làm mịn và chuẩn hóa theo cửa sổ trượt với độ dài phù hợp (ví dụ: 1 giây tương ứng với 200 mẫu ở tần số lấy mẫu 200Hz), đảm bảo các đặc trưng phản ánh rõ rệt sự khác biệt giữa các hoạt động bình thường và các tình huống té ngã.

Trong đề tài này, các đặc trưng đã trích xuất từ cảm biến MPU-6050 sau bước tiền xử lý đóng vai trò quan trọng trong việc huấn luyện mô hình SVM và thực hiện suy luận té ngã thời gian thực trên FPGA (Kit DE0-Nano). Bộ đặc trưng này được lựa chọn kỹ càng nhằm tối ưu hóa hiệu suất và khả năng phát hiện nhanh chóng, tin cậy trong các tình huống té ngã thực tế. Việc lựa chọn các đặc trưng như Mean, Variance, Magnitude đã giúp mô hình SVM được huấn luyện có thể đạt được độ chính xác cao khi triển khai thực tế trên hệ thống phần cứng FPGA.

2.3 Xây dựng mô hình SVM trên MATLAB

2.3.1 Phương pháp huấn luyện và kiểm thử

Support Vector Machine (SVM) là một thuật toán máy học mạnh mẽ, được sử dụng rộng rãi trong các bài toán phân loại nhờ khả năng tổng quát hóa cao và hoạt động hiệu quả với dữ liệu cảm biến. Trong luận văn này, mô hình SVM được xây dựng và huấn luyện trên phần mềm MATLAB, với dữ liệu đầu vào là các đặc trưng đã được xử lý và trích xuất từ bộ dữ liệu SisFall.

Quá trình huấn luyện mô hình SVM được thực hiện qua các bước cụ thể như sau:

Bước 1: Chuẩn bị tập dữ liệu

Sau khi thực hiện tiền xử lý và trích xuất đặc trưng từ các tín hiệu cảm biến, toàn bộ dữ liệu được chia ngẫu nhiên thành hai tập:

- **Tập huấn luyện (Training Set):** chiếm 80% tổng số mẫu dữ liệu.
- **Tập kiểm thử (Test Set):** chiếm 20% còn lại.

Việc chia dữ liệu theo tỷ lệ này nhằm giúp mô hình học được mối quan hệ giữa đặc trưng và nhãn phân loại mà vẫn đảm bảo đánh giá được khả năng tổng quát hóa, đồng thời hạn chế hiện tượng quá khớp (overfitting).

Bước 2: Chuẩn hóa đặc trưng đầu vào

Để đảm bảo hiệu quả huấn luyện, các đặc trưng đầu vào được chuẩn hóa về phân phối chuẩn (mean = 0, std = 1) bằng công thức:

$$x_{\text{norm}} = \frac{x - \mu}{\sigma}$$

Trong đó, μ và σ lần lượt là giá trị trung bình và độ lệch chuẩn của từng đặc trưng tính trên tập huấn luyện. Các thông số này cũng được lưu lại để sử dụng trong quá trình chuẩn hóa đầu vào khi triển khai trên FPGA.

Bước 3: Huấn luyện mô hình SVM

Mô hình được huấn luyện bằng cách sử dụng hàm `fitcsvm` của MATLAB với hàm kernel tuyến tính (linear). Kernel này được lựa chọn do tính đơn giản trong tính toán và khả năng

triển khai phù hợp trên FPGA. Đặc trưng đầu vào bao gồm 7 giá trị: **gia tốc trung bình, phương sai, độ lệch chuẩn** của 3 trục và **độ lớn vector gia tốc (magnitude)**.

Sau khi huấn luyện, mô hình lưu lại các tham số gồm:

- **Support Vectors (SV):** các mẫu dữ liệu quan trọng tạo nên siêu phẳng phân tách.
- **Hệ số trọng số (α):** xác định tầm quan trọng của mỗi SV.
- **Bias (b):** dịch chuyển của siêu phẳng.

Bước 4: Kiểm thử và đánh giá hiệu năng

Mô hình được kiểm thử trên tập kiểm thử để đánh giá hiệu năng phân loại. Các chỉ số đánh giá được sử dụng gồm:

- **Accuracy (Độ chính xác):** tỷ lệ dự đoán đúng trên tổng số mẫu.
- **Sensitivity (Độ nhạy):** tỷ lệ phát hiện đúng các trường hợp té ngã.
- **Specificity (Độ đặc hiệu):** tỷ lệ phát hiện đúng các trường hợp không té ngã.

Ngoài ra, ma trận nhầm lẫn (Confusion Matrix) và đường cong ROC cũng được sử dụng để đánh giá trực quan khả năng phân loại của mô hình.

2.3.2 Chuyển đổi từ dữ liệu float sang dạng fixed-point (Q15)

Sau khi hoàn thành quá trình huấn luyện, các tham số của mô hình (Support Vector, α , và bias) đều ở dạng dấu chấm động (float). Tuy nhiên, việc triển khai trực tiếp các tham số dạng dấu chấm động trên FPGA gây nhiều khó khăn, tốn tài nguyên và làm giảm hiệu suất xử lý. Vì vậy, việc chuyển đổi các tham số này sang dạng fixed-point (định dạng Q15) là vô cùng cần thiết.

Định dạng Q15 là định dạng fixed-point sử dụng 16 bit, trong đó 1 bit dấu và 15 bit phần lẻ. Việc sử dụng định dạng này giúp cân bằng giữa độ chính xác và khả năng triển khai hiệu quả trên FPGA, giảm thiểu đáng kể tài nguyên bộ nhớ và tính toán.

Quy trình chuyển đổi từ float sang Q15 được thực hiện bằng công thức:

$$Q15 = \text{round}(\text{FloatValue} \times 2^{15})$$

Các bước xử lý bao gồm:

- Chuyển đổi các giá trị float sang Q15 bằng phép nhân với 2^{15} , sau đó làm tròn.
- Nếu giá trị Q15 âm, cộng thêm 65536 để biểu diễn dưới dạng **số nguyên không dấu 16 bit**, phù hợp với định dạng lưu trữ trong bộ nhớ FPGA.
- Các tham số sau khi chuyển đổi được lưu thành các tệp .mif (Memory Initialization File) để nạp vào các bộ nhớ ROM trong hệ thống suy luận SVM trên FPGA.

Ví dụ, tệp supportVectors.mif, alpha.mif, và bias.mif chứa các giá trị đã được chuyển đổi, và được truy xuất tuần tự trong quá trình suy luận để tính giá trị đầu ra của mô hình SVM.

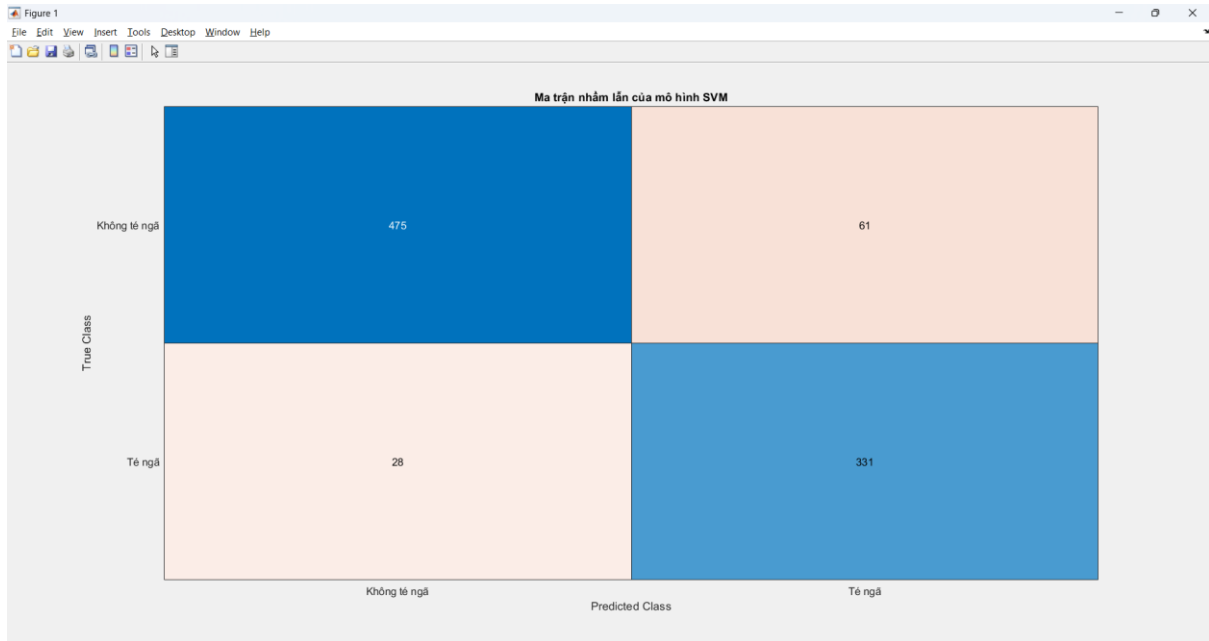
2.4 Đánh giá hiệu năng mô hình

2.4.1 Kết quả kiểm tra trên dữ liệu thử nghiệm

Sau khi mô hình SVM được huấn luyện trên MATLAB với tập huấn luyện chiếm 80% dữ liệu, mô hình được đánh giá bằng tập kiểm thử còn lại (20%), sử dụng dữ liệu cảm biến thực tế từ bộ dữ liệu SisFall. Mục tiêu là xác định mức độ chính xác và độ tin cậy của mô hình khi áp dụng vào các tình huống phát hiện té ngã.

Kết quả kiểm thử mô hình trên tập dữ liệu độc lập thể hiện hiệu suất phân loại rất khả quan:

- **Độ chính xác (Accuracy):** đạt **90.06%**, cho thấy mô hình có khả năng phân biệt hiệu quả giữa các tình huống té ngã và các hoạt động sinh hoạt thông thường.
- **Độ nhạy (Sensitivity):** đạt **92.20%**, phản ánh khả năng nhận diện đúng các trường hợp té ngã, yếu tố then chốt trong hệ thống cảnh báo thời gian thực.
- **Độ đặc hiệu (Specificity):** đạt **88.62%**, thể hiện khả năng loại bỏ chính xác các trường hợp không phải té ngã, tránh cảnh báo sai.
- **Diện tích dưới đường cong ROC (AUC):** đạt **0.964**, cho thấy mô hình có hiệu suất phân loại mạnh và ổn định.



Hình 1: Ma trận nhầm lẫn của mô hình SVM

Những chỉ số trên khẳng định rằng mô hình SVM đã được xây dựng là **phù hợp và đáng tin cậy** cho bài toán phát hiện té ngã. Với hiệu suất cao, mô hình hoàn toàn có thể tích hợp vào hệ thống phần cứng sử dụng kit FPGA DE0-Nano để hoạt động trong môi trường thời gian thực.

2.4.2 Phân tích độ chính xác và sai số của mô hình

Việc đánh giá hiệu năng chi tiết của mô hình là rất cần thiết để xác định những hạn chế và đề xuất các cải tiến tiềm năng. Qua kết quả thu được, ta nhận thấy:

- Mô hình hoạt động rất ổn định với các tình huống té ngã rõ rệt (té mạnh, té có chuyển động nhanh, biên độ lớn), độ chính xác đạt gần như tuyệt đối.
- Sai số xảy ra chủ yếu trong các tình huống té ngã nhẹ hoặc các hoạt động hàng ngày có cường độ vận động cao, gây nhầm lẫn nhỏ trong mô hình.

Nguyên nhân của các sai số này chủ yếu do sự tương đồng trong các đặc trưng chuyển động của một số hoạt động hàng ngày mạnh mẽ (như chạy nhanh, nhảy cao) và các tình huống té ngã nhẹ.

Để cải thiện hơn nữa hiệu suất mô hình, các giải pháp được đề xuất bao gồm:

- Tăng cường số lượng và đa dạng hóa thêm dữ liệu huấn luyện, đặc biệt là các trường hợp té ngã nhẹ và các hoạt động mạnh.

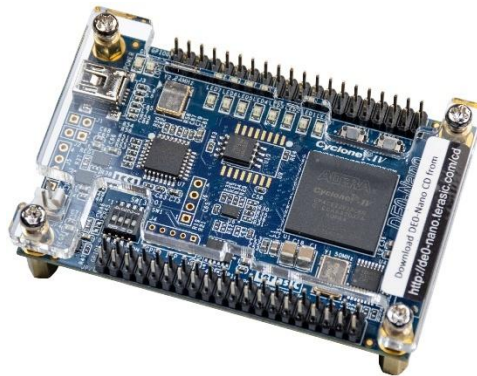
- Điều chỉnh lại các đặc trưng trích xuất, bổ sung thêm một số đặc trưng liên quan đến độ biến thiên nhanh trong thời gian ngắn (ví dụ đặc trưng gradient của tín hiệu).

Tổng thể, mô hình SVM sau khi được xây dựng, huấn luyện và kiểm thử đã đạt được kết quả rất khả quan, đảm bảo được yêu cầu về độ tin cậy và độ chính xác, hoàn toàn phù hợp để triển khai thực tế trong hệ thống FPGA (DE0-Nano) nhằm mục đích phát hiện tế ngả nhanh chóng, hiệu quả và ổn định.

3. THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG

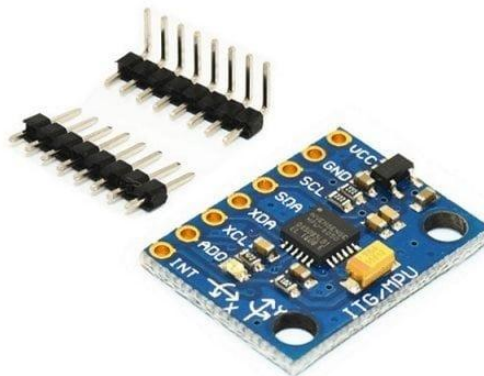
3.1 Tổng quan hệ thống phần cứng

Hệ thống phần cứng được xây dựng và triển khai trên FPGA Kit DE0-Nano. Kit DE0-Nano sử dụng chip FPGA Cyclone IV của Altera với khả năng xử lý song song mạnh mẽ, đáp ứng yêu cầu xử lý thời gian thực của ứng dụng phát hiện té ngã. Bộ kit đóng vai trò trung tâm trong việc thu nhận dữ liệu từ cảm biến, xử lý và thực hiện suy luận máy học theo mô hình SVM (Support Vector Machine).



Hình 2: DE0-Nano Development and Education Board

Cảm biến MPU-6050 được lựa chọn nhờ vào khả năng đo chính xác chuyển động cơ thể theo 6 trục: 3 trục gia tốc (AccelX, AccelY, AccelZ) và 3 trục con quay hồi chuyển (GyroX, GyroY, GyroZ). Cảm biến này được đặt ở vị trí thắt lưng người dùng để đảm bảo thu được thông tin về chuyển động trung tâm cơ thể, nơi phản ánh rõ nhất trạng thái ngã. Dữ liệu cảm biến được truyền vào FPGA thông qua giao thức I2C và được xử lý tuần tự bằng các khối logic trước khi đưa vào mô hình học máy SVM để suy luận trạng thái ngã.



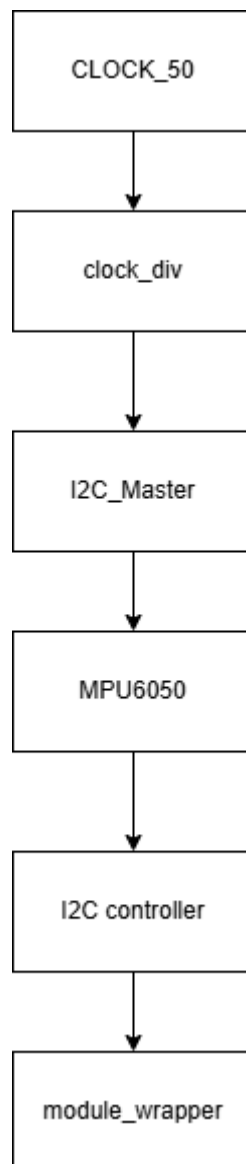
Hình 3: Cảm biến GY-521 6DOF IMU MPU6050

3.2 Khối giao tiếp I2C

3.2.1. Tổng quan khối giao tiếp I2C

Giao tiếp giữa FPGA và cảm biến MPU-6050 sử dụng chuẩn I2C, được điều khiển bằng module I2C_Master. Module này gồm các chân tín hiệu quan trọng như SCL (xung clock đồng bộ) và SDA (dữ liệu). Module được thiết kế theo máy trạng thái hữu hạn (Finite State Machine - FSM).

3.2.2. Sơ đồ khối giao tiếp I2C



Hình 4: Sơ đồ khối của hệ thống giao tiếp I2C giữa kit DE0-Nano và cảm biến MPU-6050

CLOCK_50: Là nguồn xung clock 50 MHz mặc định từ kit DE0-Nano. Tín hiệu này là cơ sở thời gian chính cho toàn bộ hệ thống, bao gồm cả các khối giao tiếp và xử lý dữ liệu.

clock_div: Do chuẩn I2C yêu cầu tần số hoạt động thấp hơn (thường là 100 kHz hoặc 400 kHz), khối clock_div thực hiện chia tần từ 50 MHz xuống tần số I2C mong muốn. Clock sau chia được đưa đến I2C_Master để đảm bảo hoạt động đúng chuẩn giao tiếp I2C.

I2C_Master: Đây là khối thực hiện vai trò **Master** trong giao tiếp I2C. Khối này được xây dựng dựa trên mô hình **FSM (Finite State Machine)** nhằm đảm bảo đúng trình tự các bước truyền nhận theo chuẩn I2C. Chức năng bao gồm:

- Gửi điều kiện bắt đầu (Start condition)
- Truyền địa chỉ slave (MPU6050)
- Ghi địa chỉ thanh ghi cần truy xuất
- Đọc dữ liệu từ thanh ghi
- Gửi điều kiện kết thúc (Stop condition)
- Xử lý tín hiệu ACK/NACK sau mỗi byte truyền

MPU6050: Là cảm biến thực được sử dụng trong hệ thống, hoạt động như một thiết bị **I2C Slave**. Cảm biến này có khả năng đo 6 trục chuyển động:

- **Gia tốc kế 3 trục:** AccelX, AccelY, AccelZ
- **Con quay hồi chuyển 3 trục:** GyroX, GyroY, GyroZ

I2C_controller: Đây là khối điều khiển chính chịu trách nhiệm phối hợp hoạt động giữa I2C_Master và cảm biến MPU6050. Khối này cũng hoạt động dưới dạng FSM, thực hiện các nhiệm vụ sau:

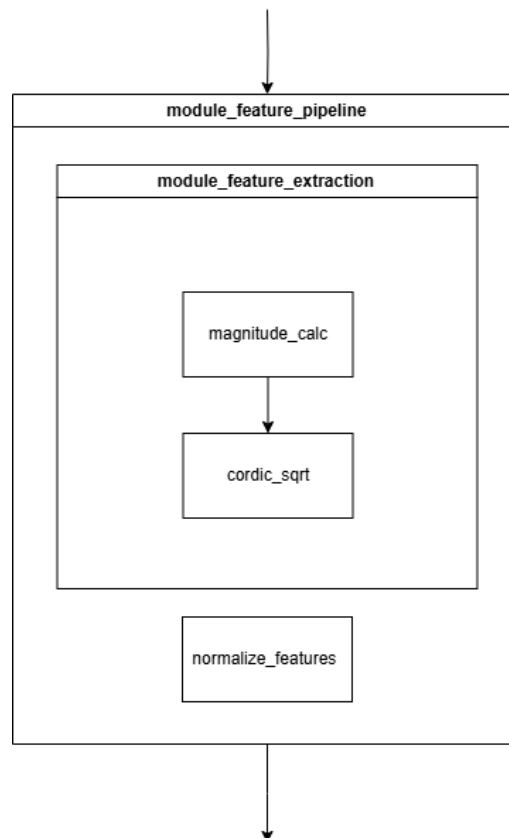
- Cung cấp địa chỉ thanh ghi cần đọc cho I2C_Master
- Khởi động quá trình truyền/nhận
- Nhận dữ liệu trả về từ cảm biến
- Gom dữ liệu từ 6 thanh ghi (3 trục gia tốc, 3 trục con quay)

- Gửi dữ liệu sang module_wrapper khi hoàn tất

module_wrapper: Sau khi dữ liệu được đọc đầy đủ từ cảm biến, khối I2C_controller sẽ truyền các giá trị này đến module_wrapper. Khối này đóng vai trò trung gian giữa phần giao tiếp và phần xử lý tín hiệu. Nó định tuyến các giá trị đọc được (dưới dạng 6 số nguyên 16-bit) và phát tín hiệu data_ready để thông báo dữ liệu đã sẵn sàng cho các khối xử lý đặc trưng phía sau.

3.3 Khối xử lý đặc trưng (Feature Extraction Pipeline)

Trong hệ thống phát hiện té ngã, việc trích xuất đặc trưng từ dữ liệu cảm biến đóng vai trò then chốt trong việc phân biệt giữa các hoạt động thông thường và trạng thái té ngã. Khối xử lý đặc trưng (Feature Extraction Pipeline) chịu trách nhiệm nhận dữ liệu thô từ cảm biến MPU-6050 (gia tốc và con quay hồi chuyển), tính toán các đặc trưng quan trọng và chuẩn hóa chúng thành định dạng Q15, sẵn sàng đưa vào mô hình học máy SVM.



Hình 5: Sơ đồ khối xử lý đặc trưng

Toàn bộ khối được triển khai dưới dạng pipeline tuần tự, gồm các module chính như sau:

- **magnitude_calc.v:** Tính toán độ lớn vectơ gia tốc và con quay theo công thức:

$$\text{Magnitude} = \sqrt{x^2 + y^2 + z^2}$$

- **cordic_sqrt:** Thực hiện phép toán căn bậc hai để tính toán hiệu quả trên phần cứng FPGA.
- **normalize_features:** Khôi chuẩn hóa các đặc trưng đã tính theo phương pháp Z-score:

$$z = \frac{x - \mu}{\sigma}$$

trong đó:

x là đặc trưng cần chuẩn hóa,
 μ là giá trị trung bình của đặc trưng,
 σ là độ lệch chuẩn.

- **feature_extraction.v:** Trích xuất các đặc trưng cần thiết như trung bình (mean), độ lệch chuẩn (standard deviation) và độ lớn magnitude của gia tốc và con quay.
- **feature_pipeline.v:** Tổng hợp 7 đặc trưng đã được chuẩn hóa thành định dạng Q15, sẵn sàng cho quá trình suy luận bằng SVM.

3.4 Khối suy luận SVM (svm_inference.v)

Khối này áp dụng mô hình SVM tuyến tính đã được huấn luyện trước với tổng số 786 support vectors, mỗi vector gồm 7 đặc trưng. Tham số mô hình, bao gồm các support vectors, alpha và bias, được lưu trữ trong các RAM nội bộ của FPGA và được nạp vào từ các file .mif.

Quá trình suy luận quản lý bởi FSM gồm các trạng thái:

- IDLE: Chờ tín hiệu khởi động.
- READ: Đọc lần lượt các support vector từ bộ nhớ.
- COMPUTE: Thực hiện tích vô hướng giữa đặc trưng đầu vào và support vector, nhân với hệ số alpha tương ứng.
- DONE: Thực hiện phép cộng bias, sau đó quyết định trạng thái té ngã dựa trên dấu của kết quả tính toán.

Kết quả đầu ra:

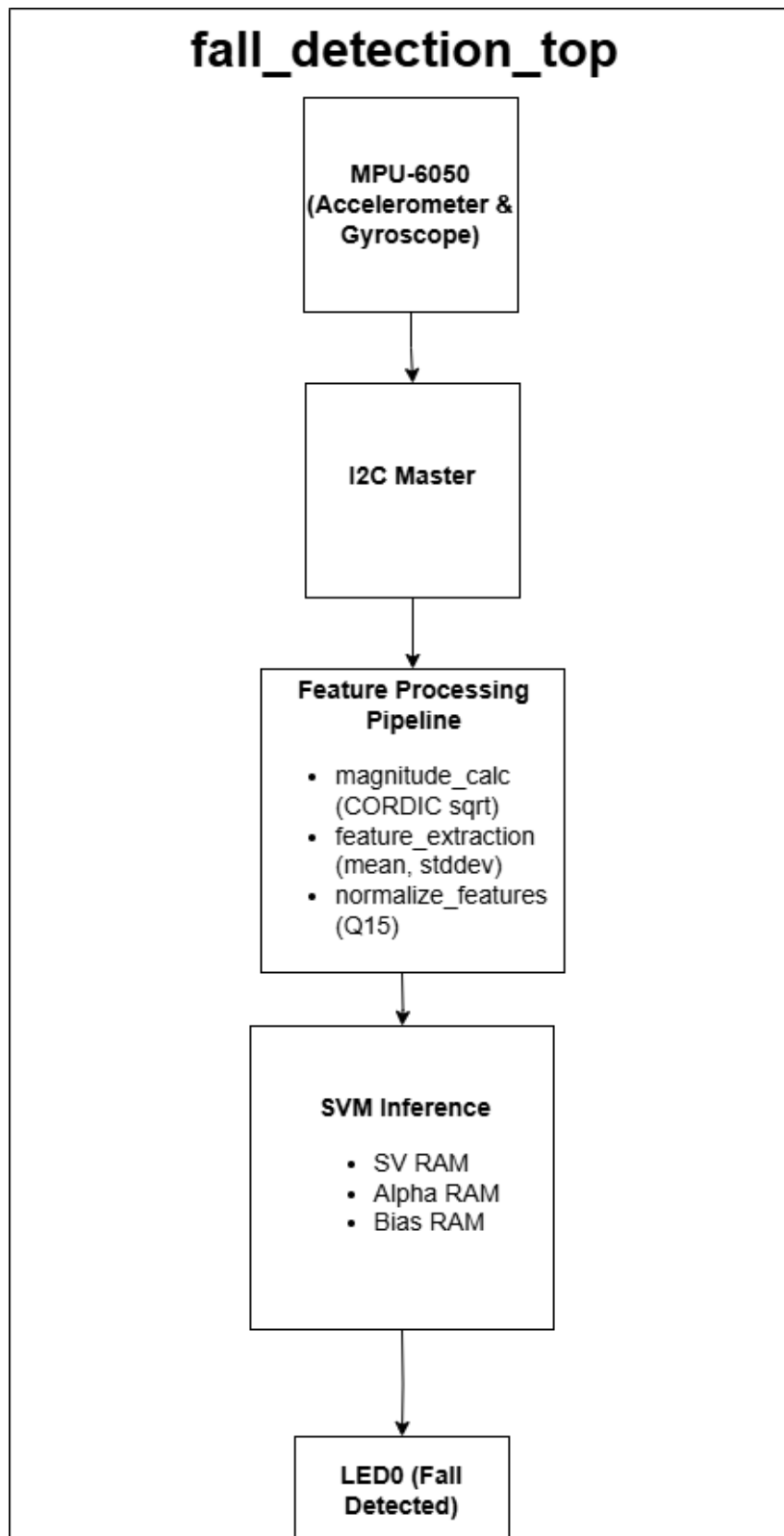
- $\text{fall_detected} = 1$ nếu nhận diện có té ngã.

- done = 1 khi kết thúc quá trình suy luận.

3.5 Điều khiển tổng thể

Toàn bộ hệ thống được vận hành đồng bộ với tín hiệu clock chuẩn CLOCK_50 MHz từ kit DE0-Nano. Tín hiệu data_ready được phát ra khi dữ liệu từ cảm biến MPU-6050 được đọc và sẵn sàng để xử lý. Khi phát hiện trạng thái té ngã (fall_detected = 1), tín hiệu điều khiển sẽ kích hoạt LED trên DE0-Nano, phát tín hiệu cảnh báo người dùng hoặc người theo dõi.

3.6 Tích hợp toàn hệ thống



Hình 6: Sơ đồ khối fall detection

Module `fall_detection_top.v` đóng vai trò kết hợp toàn bộ các khối trên thành một hệ thống hoàn chỉnh, đảm bảo việc trao đổi dữ liệu và suy luận diễn ra liên tục theo quy trình:

MPU-6050 → I2C Master → Feature Extraction Pipeline → SVM Inference → LED Output

Module này là lõi trung tâm, đảm bảo các chức năng chính như thu thập dữ liệu từ cảm biến, xử lý đặc trưng, thực hiện suy luận và đưa ra các phản hồi thích hợp theo thời gian thực.

4. KẾT QUẢ THỰC HIỆN

5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Thông qua quá trình nghiên cứu và thử nghiệm, kết quả thu được cho thấy hệ thống có khả năng phát hiện các trường hợp té ngã với độ chính xác và độ nhạy cao, đáp ứng tốt các yêu cầu về thời gian thực. Mô hình SVM được triển khai hiệu quả trên FPGA, thể hiện khả năng xử lý nhanh, đáp ứng tốt yêu cầu ứng dụng thực tế trong lĩnh vực chăm sóc sức khỏe, đặc biệt là người cao tuổi hoặc những người cần giám sát liên tục.

Tuy nhiên, hệ thống cũng gặp một số hạn chế như hiệu năng phụ thuộc nhiều vào dữ liệu huấn luyện, khả năng xử lý bị giới hạn bởi tài nguyên của FPGA, và cần cải thiện thêm độ tin cậy khi hoạt động trong các môi trường nhiễu.

5.2 Hướng phát triển

Dựa trên kết quả đạt được, hệ thống phát hiện té ngã sử dụng mô hình học máy triển khai trên FPGA vẫn còn nhiều tiềm năng để tiếp tục nghiên cứu và hoàn thiện. Một số định hướng phát triển trong tương lai được đề xuất như sau:

- **Mở rộng và cải tiến bộ dữ liệu huấn luyện:** Trong khuôn khổ đề tài, bộ dữ liệu `SisFall` được sử dụng nhằm huấn luyện và kiểm thử mô hình SVM. Tuy nhiên, bộ dữ liệu này vẫn còn hạn chế về số lượng tình huống thực tế và đa dạng đối tượng sử dụng. Do đó, việc xây dựng và mở rộng bộ dữ liệu bằng cách thu thập thêm các tình huống té ngã và hoạt động hằng ngày từ người dùng thật, ở các môi trường và điều

kiện khác nhau, sẽ giúp cải thiện đáng kể độ chính xác và độ tin cậy của mô hình phân loại.

- **Nghiên cứu và tích hợp các thuật toán học máy tiên tiến hơn:** Mặc dù mô hình SVM đã cho thấy hiệu quả nhất định trong việc phân loại té ngã, tuy nhiên trong tương lai có thể xem xét sử dụng các thuật toán học máy hiện đại hơn như mạng nơ-ron tích chập (CNN) hoặc mạng nơ-ron tái hồi (RNN) nhằm tăng cường khả năng học và nhận dạng các mẫu chuyển động phức tạp hơn, đặc biệt trong các tình huống có nhiễu hoặc hoạt động gần giống té ngã.
- **Tối ưu hóa phần cứng và phần mềm:** Hệ thống hiện tại đã được triển khai thành công trên nền tảng FPGA DE0-Nano, tuy nhiên vẫn có thể tối ưu thêm về mặt kiến trúc phần cứng (ví dụ: sử dụng pipeline hiệu quả hơn, giảm độ rộng dữ liệu không cần thiết) và phần mềm (thuật toán tính toán rút gọn, quản lý bộ nhớ hiệu quả) để giảm độ trễ xử lý, tiết kiệm tài nguyên logic và tăng thời gian hoạt động trong các thiết bị nhúng sử dụng nguồn pin.
- **Phát triển giao diện người dùng trực quan và hệ thống cảnh báo từ xa:** Việc bổ sung một giao diện người dùng thân thiện (ví dụ: hiển thị thông tin lên màn hình OLED hoặc ứng dụng điện thoại di động) sẽ giúp người sử dụng dễ dàng quan sát và theo dõi trạng thái thiết bị. Đồng thời, việc tích hợp các công nghệ không dây như **Bluetooth** hoặc **Wi-Fi** cho phép gửi cảnh báo té ngã đến người thân, bác sĩ hoặc trung tâm y tế một cách nhanh chóng, từ đó hỗ trợ kịp thời và hiệu quả trong các tình huống khẩn cấp.
- **Thử nghiệm thực tế và đánh giá toàn diện:** Sau khi hoàn thiện hệ thống, việc triển khai thử nghiệm trong môi trường thực tế với người dùng thật (đặc biệt là người cao tuổi, người có nguy cơ té ngã cao) là cần thiết để đánh giá hiệu quả tổng thể. Các thông số như độ nhạy, độ đặc hiệu, độ trễ phản hồi và mức độ chấp nhận của người dùng cần được phân tích kỹ lưỡng nhằm điều chỉnh và cải tiến hệ thống cho phù hợp với nhu cầu sử dụng hàng ngày.

6. TÀI LIỆU THAM KHẢO

- [1] L. Lakshmi Shreya, M. Lalithakeerthana, A. Lavanya, A. Laxmi Priya, N. Leela Vallabha, P. Likitha, R. Karthik, “Fall Detection for Elderly People Using Machine Learning,” *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, Vol. 11, Issue XII, December 2023. Available at: <https://www.ijraset.com>.
- [2] Sejal Badgujar, Anju S. Pillai, “Fall Detection for Elderly People Using Machine Learning,” *IEEE ICCCNT 2020*, July 2020. DOI: 10.1109/ICCCNT49239.2020.9225494.
- [3] Altera Corporation, “Cyclone IV FPGA Family Overview,” *Altera Corporation*, 2016. Available at: <https://www.intel.com/content/www/us/en/programmable/fpga/cyclone-iv.html>.
- [4] Xilinx, “Introduction to FPGA Design,” *Xilinx* 2018. Available at: <https://www.xilinx.com>.
- [5] Rita Cucchiara, Andrea Prati, Roberto Vezzani, “A Multi-Camera Vision System for Fall Detection and Alarm Generation,” *Expert Systems*, Vol. 24, Issue 5, 2007. DOI: 10.1111/j.1468-0394.2007.00275.x.
- [6] World Health Organization (WHO), “World Report on Ageing and Health,” *World Health Organization*, Geneva, 2015.
- [7] World Health Organization (WHO), “Global Report on Falls Prevention in Older Age,” *World Health Organization*, Geneva, 2008.
- [8] Glen Debar, Greet Baldewijns, Toon Goedem, Tinne Tuytelaars, Bart Vanrumste, “Camera-based Fall Detection Using a Particle Filter,” *IEEE Conference*, 2015. ISBN: 978-4244-9270-4.
- [9] S. Vadivelu, Ganesan, S., Murthy, O. V. R., and Dhall, A., “Thermal Imaging Based Elderly Fall Detection,” *Lecture Notes in Computer Science*, Vol. 10118, 2017. DOI: 10.1007/978-3-319-60820-2_43.
- [10] Purushothaman, A., “Fall Detection System Using Artificial Neural Network,” *2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018)*, Coimbatore, India, 2018.