BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC ĐÀ LẠT - LÂM ĐỒNG

BÁO CÁO TỔNG KẾT

ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN

Nghiên cứu và phát triển thuật toán đếm số bước đi của người dùng dựa trên cảm biến của điện thoại thông minh

Thuộc nhóm ngành khoa học: Kỹ thuật Điện tử - Viễn thông

Lâm Đồng, tháng 5 năm 2022

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC ĐÀ LẠT - LÂM ĐỒNG

BÁO CÁO TỔNG KẾT

ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN

Nghiên cứu và phát triển thuật toán đếm số bước đi của người dùng dựa trên cảm biến của điện thoại thông minh

Thuộc nhóm ngành khoa học: Kỹ thuật Điện tử - Viễn thông

Sinh viên thực hiện: Nguyễn Quốc Thịnh Nam

Dân tộc: Kinh

Lớp, khoa: VTK42- Khoa Vật Lý-KTHN Năm thứ: 4

Số năm đào tạo: 4.5 năm

Ngành học: Kỹ thuật Điện tử - Viễn Thông

Người hướng dẫn: TS. Nguyễn Hữu Khánh

Lâm Đồng, tháng 5 năm 2022

Mục lục

Mục lục 1
Danh mục hình 3
Danh mục bảng 4
Danh mục viết tắt 5
CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN VỀ MỤC ĐÍCH NGHIÊN CỨU VÀ HƯỚNG PHÁT TRIỂN 6
1.1. Đặt vấn đề 6
1.2. Tính cấp thiết của đề tài 7
1.3. Mục đích nghiên cứu 7
1.4. Cách tiếp cận, phương pháp nghiên cứu, phạm vi nghiên cứu: 8
CHƯƠNG 2: HỆ THỐNG ĐỊNH VỊ PDR VÀ CÁC NGHIÊN CỨU LIÊN QUAN 9
2.1. Một số hệ thống định vị hiện nay 9
2.2. Giới thiệu về hệ thống định vị PDR 10
2.2.1. Tổng quan về PDR 10
2.2.2. Hệ thống SHSs 10
A. Phát hiện bước (Step detection) 10
B. Ước tính độ dài bước (Step Length Estimation) 11
C. Ước tính hướng di chuyển (Heading Estimation) 12
2.3. Nhận dạng hoạt động của con người 12
2.4. Một số kỹ thuật học máy ML 14
2.4.1. K-nearest neighbors (KNN) 14
2.4.2. Decision tree (DT) 14
2.4.3. Random forest (RF) 14
2.4.4. Extra trees (ET) 17
2.4.5. So sánh các bộ phân loại 17
CHƯƠNG 3: PHƯƠNG PHÁP XÁC ĐỊNH SỐ BƯỚC ĐI 19
3.1. Sơ đồ khối 19
3.2. Áp dụng trên phần mềm PyCharm 24
3.2.1. CÁC TÍNH NĂNG CỦA PYCHARM 25

3.2.2. Giới thiệu ngôn ngữ python 26
CHƯƠNG 4: THU THẬP DỮ LIỆU NGƯỜI ĐI BỘ 28
4.1. Các thiết bị lấy mẫu dữ liệu 28
4.2. Thu thập dữ liệu 29
CHƯƠNG 5: KẾT QUẢ 31
CHƯƠNG 6: KẾT LUẬN 38
6.1 Kết luận. 38
6.2. Hạn chế. 38
6.3. Hướng phát triển. 38
TÀI LIỆU THAM KHẢO 39

Danh mục hình

Hình 1. Tổng quan sơ đồ các hệ thống PDR. 10
Hình 2. Ví dụ về KNN. 14
Hình 3. Ví dụ về DT. 15
Hình 4. Cấu trúc RF. 15
Hình 5. Mô tả ví dụ về phương pháp biểu quyết cứng. 16
Hình 6. Mô tả ví dụ về phương pháp biểu quyết mềm. 16
Hình 7. Sơ đồ khối 19
Hình 8. Dữ liệu thu được từ ba trục của cảm biến gia tốc. 20
Hình 9. Giá trị của cảm biến gia tốc (trục z) trước và sau khi áp dụng bộ lọc. .. 23
Hình 10. Kết quả các đỉnh lớn nhất / nhỏ nhất sau khi loại bỏ các đỉnh bị xác định sai. 24
Hình 11. Ảnh nền PyCharm. 25
Hình 12. Ảnh ngôn ngữ Python 26
Hình 13. Ảnh nền app SensorCollection. 28
Hình 14. Giao diện của app SensorCollection. 28
Hình 15. File CSV sau khi lấy dữ liệu. 29
Hình 16. Nội dung bên trong file CSV. 29
Hình 13. Quá trình thu thập dữ liệu người đi bộ. 29
Hình 17. Kết quả trung bình về độ chính xác trong việc xác định tốc độ bước đi chậm. 34
Hình 18. Kết quả trung bình về độ chính xác trong việc xác định tốc độ bước đi vừa. 34
Hình 19. Kết quả trung bình về độ chính xác trong việc xác định tốc độ bước đi nhanh. 35
Hình 20. Kết quả trung bình về độ chính xác đếm số bước đi ở ba tốc độ di chuyển. 37

Danh mục bảng

Bảng 1. So sánh ET và RF. 17
Bảng 2. So sánh DT, RF và ET. 17
Bảng 3. So sánh bộ các phân loại. 18
Bảng 4. Các đặc trưng được chọn. 20
Bảng 5. Bảng giá trị ngưỡng ΔA và ΔT theo từng tốc độ bước đi. 24
Bảng 6. Kết quả xác định tốc độ bước đi của người 1 (tốc độ chậm) 31
Bảng 7. Kết quả xác định tốc độ bước đi của người 1 (tốc độ vừa) 31
Bảng 8. Kết quả xác định tốc độ bước đi của người 1 (tốc độ nhanh) 31
Bảng 9. Kết quả xác định tốc độ bước đi của người 2 (tốc độ chậm) 32
Bảng 10. Kết quả xác định tốc độ bước đi của người 2 (tốc độ vừa) 32
Bảng 11. Kết quả xác định tốc độ bước đi của người 2 (tốc độ nhanh) 32
Bảng 12. Kết quả xác định tốc độ bước đi của người 3 (tốc độ chậm) 32
Bảng 13. Kết quả xác định tốc độ bước đi của người 3 (tốc độ vừa) 32
Bảng 14. Kết quả xác định tốc độ bước đi của người 3 (tốc độ nhanh) 33
Bảng 15. Kết quả xác định tốc độ bước đi của người 4 (tốc độ chậm) 33
Bảng 16. Kết quả xác định tốc độ bước đi của người 4 (tốc độ vừa) 33
Bảng 17. Kết quả xác định tốc độ bước đi của người 4 (tốc độ nhanh) 33
Bảng 18. Kết quả đếm số bước đi của người 1. 35
Bảng 19. Kết quả đếm số bước đi của người 2. 35
Bảng 20. Kết quả đếm số bước đi của người 3. 36
Bảng 21. Kết quả đếm số bước đi của người 4. 36
Bảng 22. Kết quả trung bình đếm số bước đi của người 1. 36
Bảng 23. Kết quả trung bình đếm số bước đi của người 2. 36
Bảng 24. Kết quả trung bình đếm số bước đi của người 3. 36
Bảng 25. Kết quả trung bình đếm số bước đi của người 4. 37

Danh mục viết tắt

CSV	Comma Separated Values
DT	Decision Tree
ET	Extra Tree
GPS	Global Positioning System
HAR	Human Activity Recognition
INSs	Inertial Navigation Systems
IPS	Indoor Positioning System
KNN	K Nearest Neighbors
ML	Machine Learning
PDR	Pedestrian Dead Reckoning
RF	Random Forest
RMSE	Root Mean Square Error
SHSs	Step-and-Heading Systems
SVM	Support Vector Machine

CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN VỀ MỤC ĐÍCH NGHIÊN CỨU VÀ HƯỚNG PHÁT TRIỂN

1.1. Đặt vấn đề

Định vị người di chuyển / đi bộ trong môi trường kín (indoors) là một trong các xu hướng công nghệ thu hút được nhiều sự chú ý bởi các ứng dụng thực tế của nó trong đời sống, ví dụ như xác định vị trí của người để thuận tiện cho việc giám sát cũng như hướng dẫn chỉ đường khi họ bước đi trong các tòa nhà hay trung tâm thương mại.

Trong môi trường mở (outdoors), hệ thống định vị toàn cầu GPS (Global Positioning System) đã hỗ trợ rất tốt và được sử dụng rộng khắp. Tuy nhiên, trong môi trường kín thì GPS không thể cung cấp thông tin vị trí của người đi bộ một cách chính xác do sự suy yếu về tín hiệu truyền nhận, mà theo nghiên cứu thì tại các thành phố lớn, một người thường sử dụng khoảng 90% thời gian trong một ngày để sinh sống và/hoặc làm việc trong môi trường kín. Do đó, hệ thống định vị người dùng trong môi trường kín cần được xây dựng và phát triển.

Hiện nay, các thiết bị thông minh như điện thoại di động hay máy tính bảng ngày càng trở nên phổ biến. Do đó, hầu như mỗi người đều có thể sở hữu cho mình một chiếc điện thoại thông minh và sử dụng nó như một vật bất ly thân. Một điểm thú vị là trên các thiết bị thông minh này thường tích hợp nhiều loại cảm biến khác nhau như cảm biến gia tốc (accelerometer), cảm biến con quay hồi chuyển (gyroscope), hay cảm biến từ trường (magnetometer). Các loại cảm biến này có thể được sử dụng cho các dịch vụ nhận biết ngữ cảnh (context-aware services) như dịch vụ xác định vị trí của người dùng. Vì vậy, việc sử dụng điện thoại để xây dựng các ứng dụng định vị là một lựa chọn hợp lý.

Pedestrian Dead Reckoning (PDR) là một trong những kỹ thuật phổ biến trong việc định vị dựa trên sự xác định bước đi và hướng di chuyển của người đi bộ trong môi trường kín. Kỹ thuật này bao gồm ba thành phần chính: xác định số bước đi, xác định độ dài mỗi bước đi, và xác định hướng di chuyển. Trong đó, sự sai lệch nhỏ từ mỗi thành phần sẽ dẫn tới sự sai lệch lớn dần trong việc định vị của hệ thống sử dụng kỹ thuật PDR. Vì vậy, việc giảm thiểu sự sai lệch trong tính toán của mỗi thành phần đều đóng góp vào việc duy trì và tăng cường hiệu năng định vị của PDR.

1.2. Tính cấp thiết của đề tài

Pedestrian Dead Reckoning là một trong những kỹ thuật phổ biến trong việc định vị và được áp dụng để xác định/theo dõi vị trí của bệnh nhân trong bệnh viện, xác định vị trí của khách hàng trong các trung tâm thương mại, hay đặc biệt trong các trường hợp khẩn cấp như cứu hỏa. Hơn thế nữa, do kỹ thuật PDR bao gồm cả việc nhận dạng các chuyển động bước đi của người dùng nên các thông tin thu thập từ quá trình bước đi này có thể giúp ích cho các ứng dụng quản lý sức khỏe người dùng. Trong suốt 20 năm qua, đã có nhiều hệ thống được nghiên cứu và thực hiện dựa trên kỹ thuật này. Mặc dù mỗi hệ thống đều có những điểm mạnh và yếu riêng, tuy nhiên hiện nay vẫn chưa có một hệ thống nào được thương mại hóa và có tầm ảnh hưởng lớn trong môi trường kín như tầm ảnh hưởng của GPS trong môi trường mở. Chính vì vậy, các nghiên cứu về việc sử dụng PDR trong việc định vị trong các tòa nhà vẫn được quan tâm và phát triển.

Hạn chế chính của việc định vị dựa trên kỹ thuật PDR là kết quả định vị bị sai lệch dần theo thời gian thực thi. Một trong các nguyên nhân là do sự xác định không chính xác số bước đi của người đang di chuyển. Trong suốt quá trình bước đi, người đi bộ có thể bước đi ở các tốc độ khác nhau (ví dụ như nhanh, bình thường hoặc chậm). Việc có thể xác định/phân biệt các tốc độ bước đi này của người đi bộ là rất cần thiết cho việc nâng cao độ chính xác cho quá trình xác định số bước đi. Vì vậy, nhóm nghiên cứu chọn lựa đề tài: **"Nghiên cứu và phát triển thuật toán đếm số bước đi của người dùng dựa trên các cảm biến của điện thoại thông minh"**.

1.3. Mục đích nghiên cứu

Nghiên cứu này hướng tới việc cải thiện hiệu năng của kỹ thuật PDR bằng cách xây dựng và phát triển thuật toán giúp nâng cao độ chính xác của việc xác định số bước đi, vốn là một thành phần quan trọng trong kỹ thuật PDR.

- Nghiên cứu và phát triển thuật toán học máy để phân biệt ba tốc độ bước đi khác nhau của người đi bộ, bao gồm: bước đi nhanh, bước đi bình thường/vừa, và bước đi chậm.
- Nghiên cứu và phát triển thuật toán xác định số bước đi của người đi bộ khi bước đi ở các tốc độ khác nhau dựa trên các cảm biến của điện thoại thông minh.

Sinh viên được hệ thống lại các kiến thức đã được học tại trường Đại học Đà Lạt, đặc biệt từ bộ môn Điện Tử - Viễn Thông (thuộc khoa Vật Lý – Kỹ Thuật Hạt Nhân), từ đó, sinh viên có thể áp dụng những kiến thức đã học vào thực tế và nâng cao về kiến thức chuyên môn cho bản thân.

1.4. Cách tiếp cận, phương pháp nghiên cứu, phạm vi nghiên cứu:

- Tìm hiểu kiến thức cơ bản thông qua sách báo, mạng Internet.

- Đề xuất thuật toán học máy để phân biệt ba tốc độ bước đi khác nhau của người đi bộ (sử dụng phần mềm Pycharm, ngôn ngữ Python).

- Đề xuất thuật toán xác định số bước đi của người đi bộ khi bước đi ở các tốc độ khác nhau (sử dụng phần mềm Pycharm, ngôn ngữ Python).

- Dữ liệu thực nghiệm từ các cảm biến trên thiết bị di động trong quá trình bước đi của người đi bộ với các tốc độ khác nhau sẽ được thu thập và phân tích để chứng minh cho hiệu năng của thuật toán.

CHƯƠNG 2: HỆ THỐNG ĐỊNH VỊ PDR VÀ CÁC NGHIÊN CỨU LIÊN QUAN

2.1. Một số hệ thống định vị hiện nay

Có 2 hệ thống định vị chủ yếu hiện nay:

- Trong môi trường mở (outdoors): hệ thống định vị toàn cầu GPS (Global Positioning System).

+ GPS hay còn gọi là hệ thống định vị toàn cầu, hệ thống này được phát triển và vận hành bởi Mỹ.

+ GPS là một hệ thống gồm rất nhiều vệ tinh bay trên quỹ đạo trái đất ở độ cao 20,200 km.

+ GPS có thể hoạt động tốt trong mọi điều kiện thời tiết cũng như bất kể nơi đâu trên Trái Đất.

+ Ứng dụng định vị GPS này có thể hoạt động liên tục suốt 24 giờ, hoàn toàn miễn phí đối với một số dịch vụ, và là hệ thống chính được áp dụng trên các thiết bị di động như điện thoại thông minh, laptop, tablet, ...

- Trong môi trường kín (indoors): hệ thống định vị trong nhà IPS (Indoor Positioning System).

+ Thay vì sử dụng vệ tinh như GPS, IPS dựa trên các công nghệ khác nhau.

+ Nó có thể được xây dựng bằng cách sử dụng nhiều điểm phát Wi-Fi trong một tòa nhà. IPS tính toán địa điểm của bạn bằng tín hiệu của Wi-Fi, ngoài ra nó còn có thể sử dụng Bluetooth hay các cảm biến tích hợp trên các thiết bị thông minh.

+ Các ứng dụng thực tế của IPS là:

. Tìm địa điểm trong các tòa nhà văn phòng lớn, các tòa nhà trường đại học, khu trung tâm, viện bảo tàng, bệnh viện.

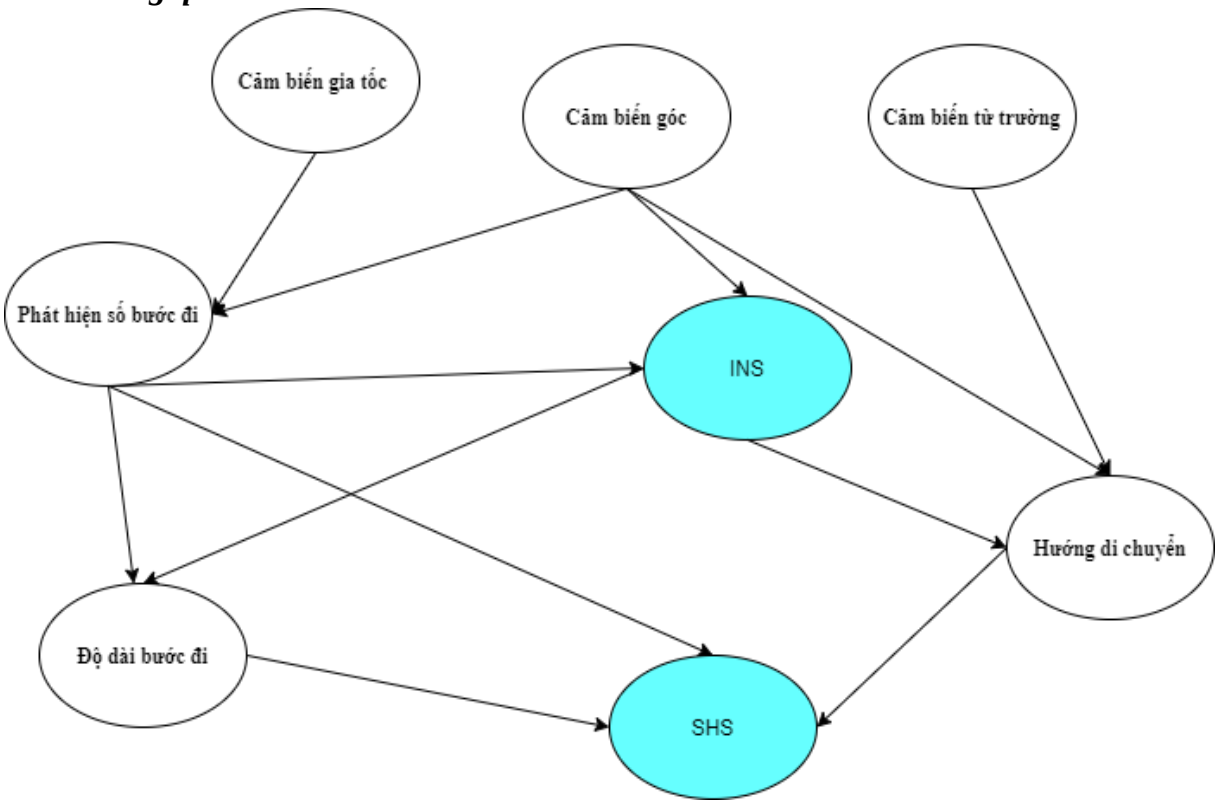
. Tình huống khẩn cấp: điều hướng cứu hộ và khoanh vùng tình huống khẩn cấp.

. Theo dõi người và tài sản - bệnh nhân, trẻ em, khách tham quan, du khách, ví dụ: theo dõi hành lý tại các sân bay; giao nhận, vận chuyển hàng hóa và theo dõi container trong kho, bến cảng, sân bay…; xác định vị trí thiết bị trong nhà máy, văn phòng và bệnh viện.

+ Ứng dụng xã hội: Tìm người hay tìm chỗ mua sắm, hỗ trợ đỗ xe trong nhà Ứng dụng trong lĩnh vực quảng cáo…

2.2. Giới thiệu về hệ thống định vị PDR

2.2.1. Tổng quan về PDR



Hình 1. Tổng quan sơ đồ các hệ thống PDR.

Hình 1 mô tả hai hệ thống định vị sử dụng các cảm biến tích hợp trên điện thoại thông minh, bao gồm hệ thống INSs (Inertial Navigation Systems) và SHSs (Step-and-Heading Systems). Hệ thống INS là hệ thống xác định vị trí trong không gian ba chiều được tính toán từ các cảm biến tại mỗi thời điểm xác định. Trong khi đó, hệ thống SHS xác định vị trí của người đang di chuyển dựa trên ba thành phần: số bước đi, độ dài bước, và hướng di chuyển. Nghiên cứu này tập trung vào hệ thống SHS với việc nâng cao độ chính xác trong việc xác định số bước đi.

2.2.2. Hệ thống SHSs

Hệ thống SHSs xác định vị trí mới dựa trên cặp thông tin (độ dài bước, hướng di chuyển) hoặc (độ dài bước, sự thay đổi hướng di chuyển), từ đó đưa ra vị trí mới trong không gian hai chiều.

A. Phát hiện bước (Step detection)

Nhiệm vụ đầu tiên của SHS là xác định các bước hoặc bước tiến dựa trên dữ liệu thu được từ các cảm biến. Trên thực tế, điều này thậm chí còn được sử dụng trong nhiều hệ thống INS. Có hai loại thuật toán chính:

- Phát hiện trạng thái: các thuật toán xác định các khoảng thời gian dữ liệu trong đó một chân nhất định được đặt trên sàn. Để làm điều này, cảm biến được gắn vào chân. Thông thường, chúng thích hợp để đếm bước nhưng cho đầu ra phân đoạn kém.

- Phát hiện chu kỳ bước: thuật toán phát hiện các chu kỳ trong dữ liệu cảm biến do chuyển động đi bộ là lặp đi lặp lại. Điều này có thể liên quan đến việc tìm kiếm các mẫu dữ liệu lặp lại hoặc các sự kiện lặp lại. Chúng rất phù hợp với phân đoạn theo bước.

Các thuật toán phát hiện bước đi điển hình là dựa trên ngưỡng. Nguyên tắc là cảm biến sẽ ổn định trong giai đoạn đứng (stance) và cảm biến gia tốc sẽ báo cáo sự thiếu hoạt động tương ứng mà ngưỡng có thể dễ dàng xác định. Hầu hết các thuật toán ngưỡng dựa trên cường độ của cảm biến gia tốc [1], [2], [3], ngưỡng dựa trên cảm biến vận tốc góc cũng được sử dụng [4], [5], [6], và ngưỡng kết hợp của cả hai loại cảm biến [7]. Ngay cả việc xác định ngưỡng từ cảm biến từ trường cũng có thể sử dụng được trong một số trường hợp [8].

Một số kỹ thuật được phát triển để xác định chính xác các sự kiện cụ thể trong quá trình bước đi của người đi bộ để phân đoạn dữ liệu, cũng như cho phép các cảm biến thu thập dữ liệu gắn lên các bộ phận khác nhau trên cơ thể.

- Phát hiện đỉnh (peak detection): Khi người đi bộ bước đi, tại thời điểm mà gót chân chạm vào mặt đất sẽ tạo ra những thay đổi mạnh đối với gia tốc theo phương thẳng đứng. Các thuật toán phát hiện đỉnh tiêu chuẩn có thể được sử dụng để xác định các bước đi. Lưu ý rằng mỗi tác động của chân có thể tạo ra nhiều đỉnh cục bộ [9], [10]. Điều này có thể làm tăng đáng kể độ phức tạp của thuật toán.

- Zero crossing: một cách dễ dàng hơn để sử dụng tính tuần hoàn trong một quá trình di chuyển (bước đi) bằng cách theo dõi giá trị gia tốc cho các điểm giao cắt mà tại đó giá trị của cảm biến bằng không [11].

Tất cả các kỹ thuật này thường được áp dụng cho tín hiệu từ cảm biến gia tốc, thường là sau khi áp dụng bộ lọc thông thấp để loại bỏ nhiễu. Các tần số cắt bộ lọc khoảng 20 Hz vẫn cho phép giữ nguyên các chu kỳ bước, mặc dù bộ lọc xuống 2 hoặc 3 Hz cũng đã được sử dụng [12]. Cần lưu ý rằng hầu hết các thử nghiệm chỉ sử dụng gia tốc theo phương thẳng đứng, nhưng không tính tới những thay đổi về trạng thái chung của cảm biến trong một bước. Thay vào đó, các hệ thống giả định rằng một trong các trục của cảm biến vẫn thẳng đứng trong suốt quá trình bước đi. Giả thiết này đúng nhất trong trường hợp các cảm biến gia tốc gắn vào thân người [13], [14], mặc dù nó đã được chứng minh là có thể áp dụng được khi gắn trên bàn chân [15].

B. Ước tính độ dài bước (Step Length Estimation)

Trên thực tế, cách tiếp cận đơn giản nhất để ước tính độ dài bước là giả định nó là một giá trị hằng số. Các nghiên cứu chỉ ra người đi bộ có tốc độ đi bộ tự nhiên với độ dài bước không đổi một cách đáng ngạc nhiên. Tuy nhiên, tốc độ đi bộ tự nhiên này được điều chỉnh khi bước nhanh, chạy nhanh hoặc đi bộ với người khác. Weinberg báo cáo rằng độ dài bước có thể thay đổi tới 40% giữa những người đi bộ đi cùng tốc độ và lên đến 50% trên phạm vi tốc độ đi bộ của một cá nhân [16]. Lưu ý rằng, trong các bài kiểm tra ngắn mà những người tham gia được yêu cầu đi bộ liên tục dọc theo một

tuyến đường cụ thể, họ có khả năng áp dụng tốc độ đi bộ tự nhiên của mình và điều này có thể dẫn đến độ chính xác của việc xác định độ dài bước đi.

Weinberg cũng mô tả một quy trình ước tính độ dài bước động dựa trên chuyển vị thẳng đứng tối đa của hông (còn gọi là "độ nảy"). Chiều dài sải chân khi bước đi được chỉ ra là một hàm của độ nảy và góc thẳng đứng giữa điểm cao nhất và thấp nhất của hông trong một sải chân duy nhất [16]. Góc này được coi là không đổi mặc dù nó thực sự liên quan đến chiều dài chân của người dùng. Độ chính xác của phương pháp này lên tới 92% trong việc xác định độ dài bước đi và được áp dụng rộng rãi trong rất nhiều hệ thống định vị PDR.

Một giải pháp khác là ước tính độ dài bước cơ sở dựa trên tần số bước quan sát, có thể được ước tính bằng cách sử dụng các kỹ thuật phát hiện bước khác nhau. Nghiên cứu đã chỉ ra sự kết hợp chặt chẽ giữa tần số bước và tốc độ đi bộ. Phương pháp này có thể đạt được sai số RMSE (Root Mean Square Error) là 5,6% [17].

Các biện pháp trực tiếp của từng bước cũng đã được sử dụng. Ví dụ, Saarinen đã sử dụng các cảm biến siêu âm gắn ở mặt trước và mặt sau của mỗi chiếc giày [18] và sử dụng các kỹ thuật xác định độ dài bước phổ biến. Các kỹ thuật này chắc chắn sẽ tạo ra các ước lượng dịch chuyển chất lượng cao hơn. Tuy nhiên, sự gia tăng độ chính xác thường không đáng kể vì độ lệch hướng di chuyển lớn. Cuối cùng, một số hệ thống ước tính độ dài bước lặp lại bằng cách đánh giá các đường đi được tạo ra bằng cách sử dụng các độ dài khác nhau và sử dụng bản đồ xây dựng để chọn độ dài tối ưu [19].

C. Ước tính hướng di chuyển (Heading Estimation)

Ước tính tiêu đề trong hệ thống SHS thường không khác so với hệ thống INS vì có rất ít cảm biến có sẵn làm đầu vào. Lấy tích phân một lần của giá trị cảm biến con quay hồi chuyển có thể xác định được hướng di chuyển của người đi bộ. Bởi vì SHS có thể tránh sử dụng tích phân để tính độ dài bước đi, độ lệch khi tính toán vị trí tăng theo tuyến tính. Ngoài ra, một số hệ thống chỉ sử dụng một con quay hồi chuyển duy nhất được gắn song song đối với thân, đưa ra giả định rằng nó vẫn (gần) thẳng đứng trong khi đi bộ. Cảm biến từ trường cũng có thể được sử dụng trực tiếp hoặc kết hợp với đầu ra của con quay hồi chuyển để ước tính sự thay đổi hướng di chuyển.

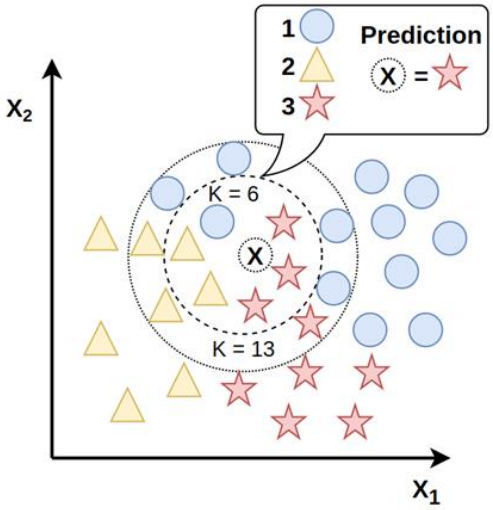
2.3. Nhận dạng hoạt động của con người

Trong lĩnh vực nhận dạng hoạt động của con người (Human Activity Recognition - HAR), có hai nhánh chính là dựa trên camera (vision) và dựa trên cảm biến (sensor). Công việc này chủ yếu tập trung vào việc sử dụng các loại cảm biến khác nhau được tích hợp trên điện thoại thông minh để phát hiện các hoạt động của người dùng như đi bộ, đi lên lầu, đi xuống cầu thang, …

Trong [20], Yin và các cộng sự đã sử dụng hai cảm biến phần cứng (là cảm biến gia tốc và con quay hồi chuyển) và hai cảm biến phần mềm trong hệ thống Android (là cảm biến gia tốc tuyến tính và cảm biến định hướng) để phân loại năm hoạt động bao gồm đi bộ, chạy, ngồi, đi lên lầu và đi xuống cầu thang. Các đặc tính khác nhau của miền thời gian như giá trị lớn nhất, giá trị nhỏ nhất, giá trị trung bình, độ lệch chuẩn, … sẽ được tính toán. Các thuật toán học máy ML (Machine Learning) khác nhau như Decision Tree, Support Vector Machine, Multilayer Perceptron được dùng để phân loại các hoạt động. Hai trường hợp được thiết lập để kiểm tra kết quả phân loại. Trong cả hai trường hợp, độ chính xác có thể đạt 99%. Trong tất cả các thuật toán ML, Decision Tree đạt được những kết quả tốt nhất. Tuy nhiên, hạn chế của nghiên cứu này là họ đã sử dụng nhiều đặc trưng (là 12) để phân loại ba hoạt động và điện thoại thông minh được giả định là chỉ bỏ vào túi. Trong các công trình nghiên cứu của Hung [21] và Suarez [2], họ đã kết hợp các tính năng được tính toán từ một cảm biến gia tốc và một cảm biến con quay hồi chuyển để nhận ra sáu loại hoạt động của con người trong khi điện thoại thông minh được gắn vào eo của người dùng. Tổng cộng có 45 đặc trưng trong [21] và 48 đặc trưng trong [8] từ cả miền thời gian và miền tần số được sử dụng cho ba bộ phân loại ML. Từ các phân loại này, thuật toán hồi quy tuyến tính đạt được độ chính xác tốt nhất với 96% và hơn 97% sử dụng Decision Tree. Vị trí cố định của điện thoại thông minh là hạn chế của các công trình nghiên cứu này. Để giải quyết hạn chế của hai công trình trên, Klein và cộng sự đã đề xuất một phương pháp không chỉ để phát hiện các chuyển động (là đi bộ bình thường và đi bộ nhanh) mà còn để phát hiện cách giữ điện thoại thông minh của người dùng (là đang gọi/nghe điện thoại, đong đưa, nhắn tin và bỏ túi). Các thuật toán ML khác nhau được áp dụng để phân loại các chuyển động và phong cách giữ điện thoại. Một số lượng lớn các đặc trưng (111) được sử dụng bao gồm các tính năng thống kê, tính năng miền thời gian và các tính năng cảm biến chéo. Gradient là thuật toán tăng cường có độ chính xác 95%, đây là kết quả tốt nhất so với các thuật toán khác. Sử dụng 111 đặc trưng là nhược điểm chính của thuật toán này, đặc biệt là khi áp dụng chúng vào hệ thống thời gian thực. Do đó, thuật toán này cố gắng giảm bớt số lượng tính năng nhưng vẫn đạt được độ chính xác phân loại hợp lý.

2.4. Một số kỹ thuật học máy ML

2.4.1. K-nearest neighbors (KNN)



Hình 2. Ví dụ về KNN.

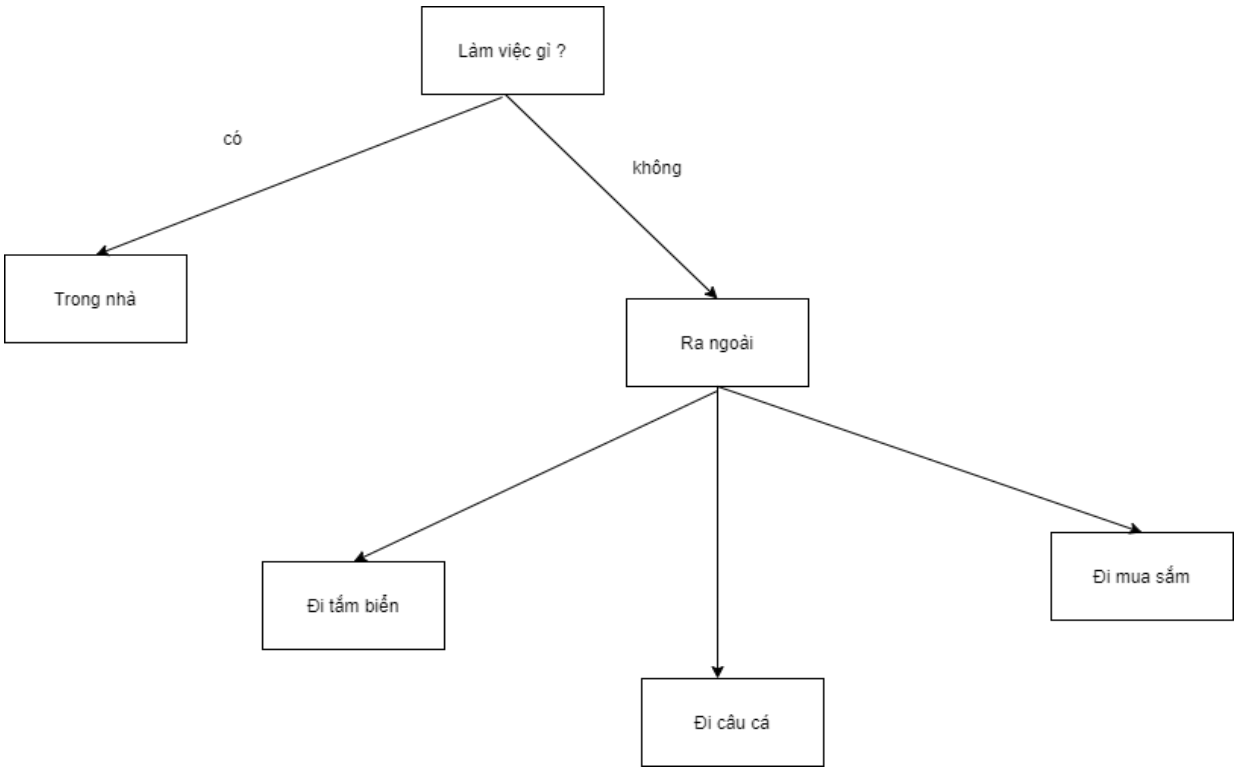
KNN được sử dụng để gán một điểm dữ liệu cho các cụm dựa trên phép đo tương đồng. "K" là số lượng điểm gần nhất. Để có được vùng lân cận cho mỗi dữ liệu, khoảng cách Euclide được áp dụng giữa các cặp dữ liệu. Hình 2 cho thấy một ví dụ về KNN. Nếu K = 6, các ngôi sao đỏ có số điểm lớn nhất, thì X sẽ được dự đoán là một ngôi sao đỏ. Nhưng nếu K tăng lên 13, vòng tròn màu xanh lam có được số điểm lớn nhất, thì X sẽ được dự đoán là một vòng tròn màu xanh lam.

2.4.2. Decision tree (DT)

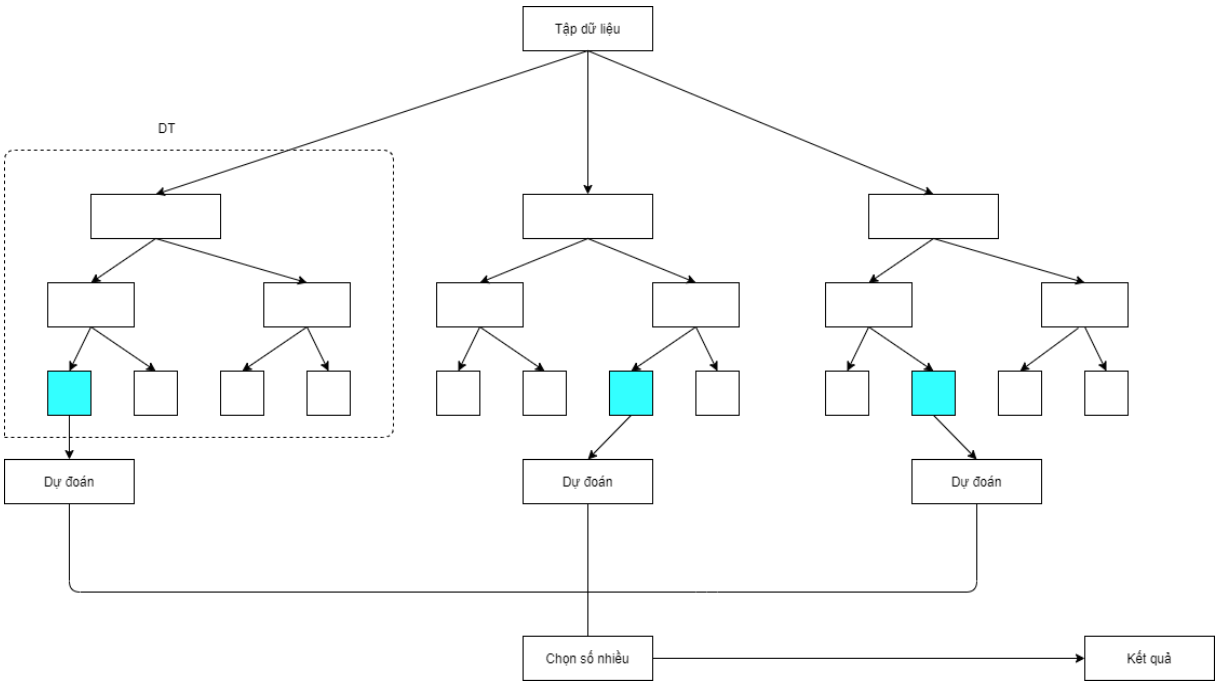
Cấu trúc của DT bao gồm các nút nội bộ, nút lá và nhánh. Mỗi nút nội bộ hiển thị một thuộc tính và nó được liên kết với một bài thử nghiệm có liên quan đến phân loại dữ liệu. Các nút lá cũng là phân loại. Các nhánh đại diện cho mỗi kết quả có thể có của các bài thử ngiệm được áp dụng. Một ví dụ về DT được thể hiện trong Hình 3, trong đó chỉ ra rõ các thành phần chính của DT.

2.4.3. Random forest (RF)

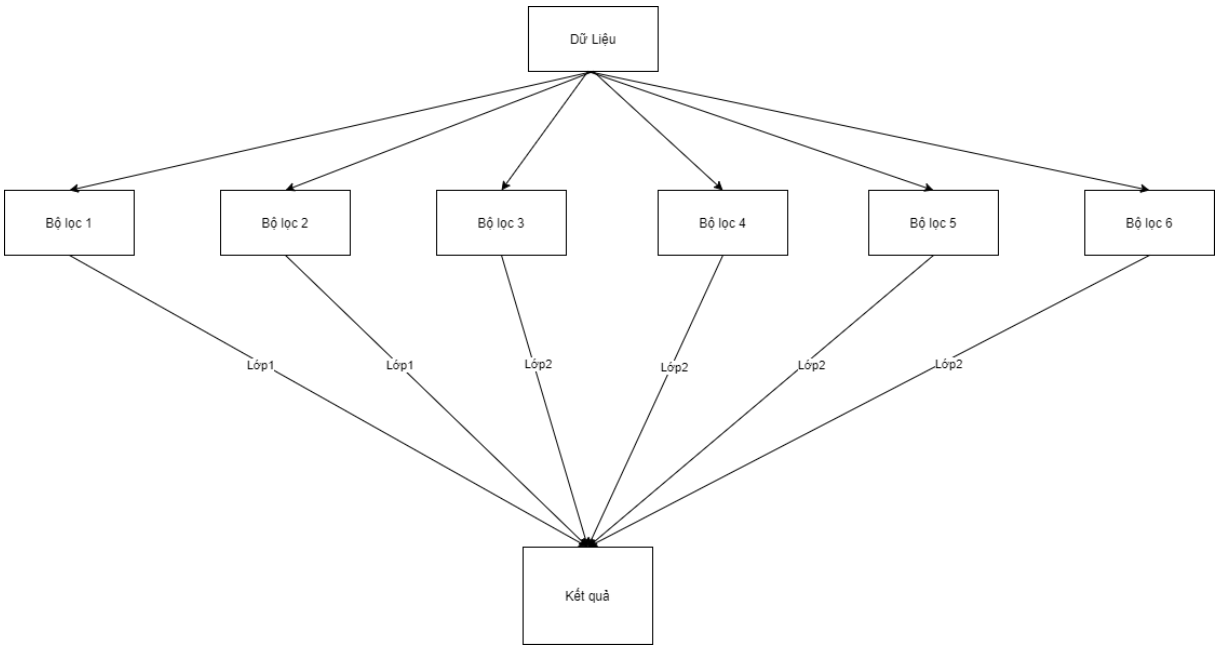
RF được đề xuất bởi Breiman [15]. Nó là một thuật toán phân loại hoạt động bằng cách sử dụng nhiều Decision Trees. Mỗi Decision Tree tìm hiểu các quy tắc đơn giản được trích xuất từ dữ liệu. Độ phức tạp sẽ tỷ lệ thuận với sự gia tăng (sâu hơn) của cây. Thuật toán này cố gắng khắc phục vấn đề về trang bị quá mức của DT cơ bản. RF phân loại các trường hợp dựa trên các quyết định của nhiều bộ phân loại, do đó, nó còn được gọi là tập hợp phân loại học tập. Phương pháp này sử dụng ý tưởng đóng bao để giảm phương sai mà không làm tăng sự thiên vị. Cấu trúc của RF được thể hiện trong hình 4. Phương pháp biểu quyết (voting) được sử dụng sau mỗi DT. Có hai loại biểu quyết là biểu quyết cứng (Hình 5) và biểu quyết mềm (Hình 6).



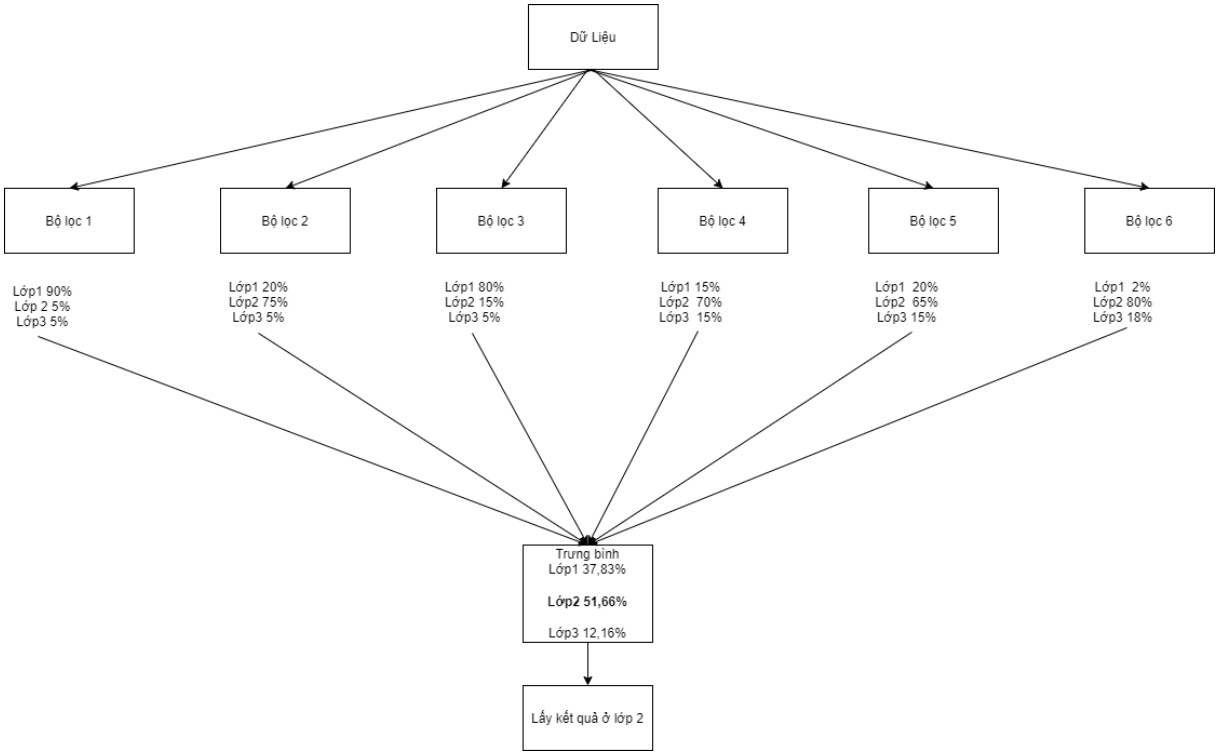
Hình 3. Ví dụ về DT.



Hình 4. Cấu trúc RF.



Hình 5. Mô tả ví dụ về phương pháp biểu quyết cứng.



Hình 6. Mô tả ví dụ về phương pháp biểu quyết mềm.

2.4.4. Extra trees (ET)

Bảng 1. So sánh ET và RF.

ET	RF
- Nhanh hơn. - Phương sai thấp. - Chọn một tách ngẫu nhiên để chia nút mẹ thành hai nút con ngẫu nhiên. - Các quan sát được rút ra mà không cần thay thế → không cần lặp lại các quan sát	- Chậm hơn. - Phương sai trung bình. - Chọn phân tách tốt nhất để chuyển đổi mẹ thành hai nút con đồng nhất nhất. - Các quan sát được rút ra với sự thay thế → lặp lại các quan sát được áp dụng

ET là một thuật toán phân loại về cơ bản sử dụng ý tưởng tương tự như RF [7]. Cả hai đều sử dụng nhiều Decision Trees quyết định để dự đoán kết quả cuối cùng. Bằng cách sử dụng một tập hợp con ngẫu nhiên của các tính năng, thuật toán này cho thấy sự cải thiện về hiệu suất so với DT. ET sử dụng toàn bộ các mẫu học tập (chứ không phải là một bản sao bootstrap) để xây dựng cây. Nó chọn các phần tách, cả thuộc tính và điểm cắt, hoàn toàn hoặc một phần ngẫu nhiên. Do đó, ET có một phương sai thấp. Bảng 1 so sánh ET và RF. Trong bảng này, sự phân chia là quá trình chuyển đổi một nút mẹ không đồng nhất thành 2 nút con đồng nhất.

2.4.5. So sánh các bộ phân loại

Bảng 2 so sánh giữa DT, RF và ET, cho thấy sự khác biệt giữa ba phương pháp. Bảng chỉ ra sự đơn giản của DT so với những phương pháp khác. Sự khác biệt chính giữa RF và ET là bootstrapping và cách tách tại mỗi nút quyết định.

Bảng 2. So sánh DT, RF và ET.

	DT	RF	ET
Số lượng cây	Một	Nhiều	Nhiều
Các tính năng được xem xét để phân chia tại mỗi nút quyết định	Tất cả các tính năng	Tập hợp ngẫu nhiên của các tính năng	Tập hợp ngẫu nhiên của các tính năng
Bootstrapping	Không	Có	Không
Tách	Chia tách tốt nhất	Chia tách tốt nhất	Chia tách ngẫu nhiên

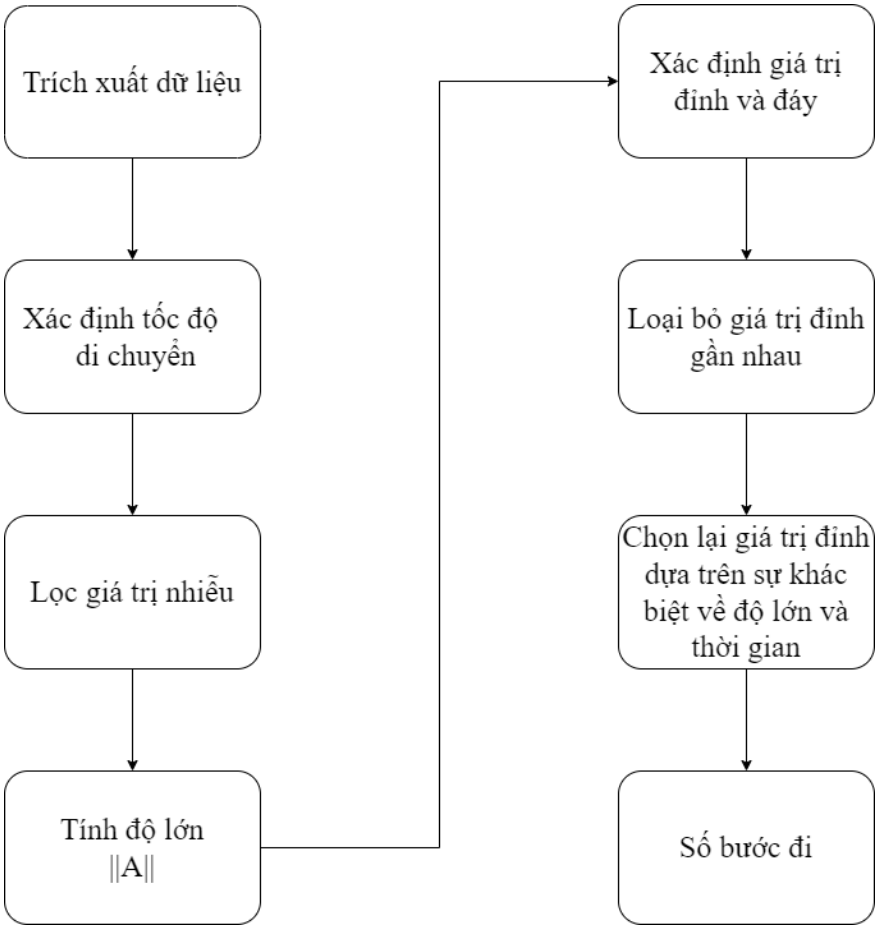
Bảng 3. So sánh bộ các phân loại.

	KNN	DT	RF	ET
Thực hiện dễ dàng	Có	Có	Không	Không
Trí nhớ	Lớn	Nhỏ	Lớn	Lớn
Phân biệt	Có	Có	Có	Có
Tăng hiệu suất với các tính năng tăng lên	Có	Có	Có	Có
Bootstrapping	Không	Không	Có	Không

Hơn nữa, việc so sánh các thuật toán trên được thể hiện trong Bảng 3. Có một số tiêu chí được đưa ra để so sánh các phân loại này với nhau chẳng hạn như sự dễ dàng thực hiện, chi phí tính toán hoặc phân biệt, … Hiệu suất tổng thể của các phân loại tăng từ trái sang phải. Theo hướng này, tất nhiên sự phức tạp của các phân loại cũng sẽ tăng lên trong khi mức độ dễ thực hiện sẽ giảm. Đối với các trường hợp DT, RF và ET, chỉ có bootstrapping và phương sai giữa dữ liệu thực và thử nghiệm được xem xét.

CHƯƠNG 3: PHƯƠNG PHÁP XÁC ĐỊNH SỐ BƯỚC ĐI

3.1. Sơ đồ khối

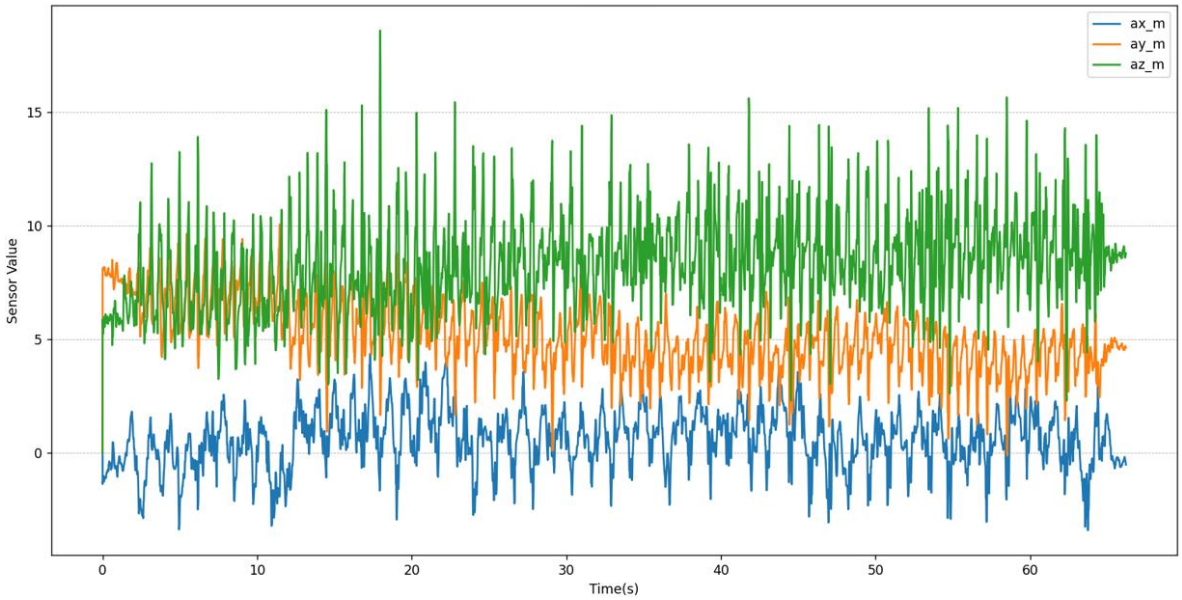


Hình 7. Sơ đồ khối

Hình 7 mô tả sơ đồ khối của phương pháp xác định bước đi, bao gồm việc xác định tốc độ bước đi và việc đếm số bước đi dựa trên các tốc độ đó. Các bước bao gồm:

1. Trích xuất dữ liệu

Lấy thông tin dữ liệu từ ba trục của cảm biến gia tốc, bao gồm ax_m, ay_m, az_m (Hình 8). Tần số lấy mẫu được sử dụng là 50 Hz. Dữ liệu được thu thập từ ứng dụng "SensorCollection" trên nền tảng Android và dữ liệu từ cảm biến được lưu lại dưới dạng file CSV (Comma Separated Values) để xử lý.



Hình 8. Dữ liệu thu được từ ba trục của cảm biến gia tốc.

2. Xác định tốc độ bước đi

Dự đoán tốc độ bước đi dựa trên các kỹ thuật học máy: Support Vector Machine (SVM), K Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), Extra Tree (ET). Với các kỹ thuật này, 50 đặc trưng được sử dụng để phân loại ba tốc độ bước đi: nhanh, vừa, và chậm. Bảng 4 mô tả các đặc trưng được chọn. Tỉ lệ xuất hiện tốc độ nào cao hơn sau khi phân loại thì bộ tham số tương ứng của tốc độ đó sẽ được áp dụng vào xác định số bước đi.

Bảng 4. Các đặc trưng được chọn.

STT	Mô tả	Ký hiệu	STT	Mô tả	Ký hiệu
			26	Phạm vi liên phần của trục y của cảm biến gia tốc	y_{IQR} (m/s^2)
1	Giá trị trung bình của trục x của cảm biến gia tốc	x_mean (m/s^2)	27	Phạm vi liên phần của trục z của cảm biến gia tốc	z_{IQR} (m/s^2)
2	Giá trị trung bình của trục y của cảm biến gia tốc	y_mean (m/s^2)	28	Đếm giá trị âm của trục x của cảm biến gia tốc	x_neg_count (m/s^2)
3	Giá trị trung bình của trục y của cảm biến gia tốc	z_mean (m/s^2)	29	Đếm giá trị âm của trục y của cảm biến gia tốc	y_neg_count (m/s^2)

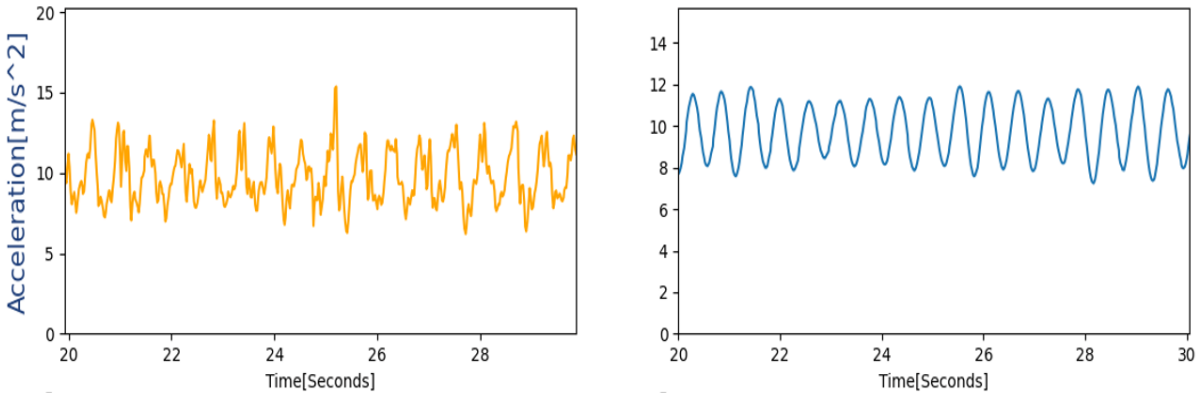
4	Độ lệch chuẩn của trục x của cảm biến gia tốc	x_std_{ax} (m/s^2)	30	Đếm giá trị âm của trục z của cảm biến gia tốc	z_neg_count (m/s^2)
5	Độ lệch chuẩn của trục y của cảm biến gia tốc	y_std (m/s^2)	31	Đếm giá trị dương của trục x của cảm biến gia tốc	x_pos_count (m/s^2)
6	Độ lệch chuẩn của trục z của cảm biến gia tốc	z_std (m/s^2)	32	Đếm giá trị dương của trục y của cảm biến gia tốc	y_pos_count (m/s^2)
7	Độ lệch tuyệt đối trung bình của trục x của cảm biến gia tốc	x_aad	33	Đếm giá trị dương của trục z của cảm biến gia tốc	z_pos_count (m/s^2)
8	Độ lệch tuyệt đối trung bình của trục y của cảm biến gia tốc	y_aad	34	Số giá trị trên trung bình của trục x của cảm biến gia tốc	x_above_mean (m/s^2)
9	Độ lệch tuyệt đối trung bình của trục z của cảm biến gia tốc	z_aad	35	Số giá trị trên trung bình của trục y của cảm biến gia tốc	y_above_mean (m/s^2)
10	Giá trị nhỏ nhất của trục x của cảm biến gia tốc	x_min (m/s^2)	36	Số giá trị trên trung bình của trục z của cảm biến gia tốc	z_above_mean (m/s^2)
11	Giá trị nhỏ nhất của trục y của cảm biến gia tốc	y_min (m/s^2)	37	Số đỉnh của trục x của cảm biến gia tốc	x_peak_count
12	Giá trị nhỏ nhất của trục z của cảm biến gia tốc	z_min (m/s^2)	38	Số đỉnh của trục y của cảm biến gia tốc	y_peak_count
13	Giá trị lớn nhất của trục x của cảm biến gia tốc	x_max (m/s^2)	39	Số đỉnh của trục z của cảm biến gia tốc	z_peak_count
14	Giá trị lớn nhất của trục y của cảm biến gia tốc	y_max (m/s^2)	40	Độ lệch của trục x của cảm biến gia tốc	x_skewness

15	Giá trị lớn nhất của trục z của cảm biến gia tốc	z_max (m/s^2)	41	Độ lệch của trục y của cảm biến gia tốc	y_skewness
16	Sự khác biệt giữa giá trị lớn nhất và nhỏ nhất của trục x của cảm biến gia tốc	$x_{maxmin_{diff}}$ (m/s^2)	42	Độ lệch của trục z của cảm biến gia tốc	z_skewness
17	Sự khác biệt giữa giá trị lớn nhất và nhỏ nhất của trục y của cảm biến gia tốc	$y_{maxmin_{diff}}$ (m/s^2)	43	Kurtosis của trục x của cảm biến gia tốc	x_kurtosis
18	Sự khác biệt giữa giá trị lớn nhất và nhỏ nhất của trục z của cảm biến gia tốc	$z_{maxmin_{diff}}$ (m/s^2)	44	Kurtosis của trục y của cảm biến gia tốc	y_kurtosis
19	Trung vị của trục x của cảm biến gia tốc	x_median (m/s^2)	45	Kurtosis của trục z của cảm biến gia tốc	z_kurtosis
20	Trung vị của trục y của cảm biến gia tốc	y_median (m/s^2)	46	Năng lượng của trục x của cảm biến gia tốc	x_energy
21	Trung vị của trục z của cảm biến gia tốc	z_median (m/s^2)	47	Năng lượng của trục y của cảm biến gia tốc	y_energy
22	Độ lệch tuyệt đối trung vị của trục x của cảm biến gia tốc	x_{mad} (m/s^2)	48	Năng lượng của trục z của cảm biến gia tốc	z_energy
23	Độ lệch tuyệt đối trung vị của trục y của cảm biến gia tốc	y_{mad} (m/s^2)	49	Kết qủa trung bình của cảm biến gia tốc	avg_result_accl
24	Độ lệch tuyệt đối trung vị của trục z của cảm biến gia tốc	z_{mad} (m/s^2)	50	Độ lớn tín hiệu khu vực của cảm biến gia tốc	sma

25	Phạm vi liên phần của trục x của cảm biến gia tốc	x_{IQR} (m/s^2)

3. *Lọc các giá trị nhiễu*

Dữ liệu thu được từ ba trục của cảm biến gia tốc sẽ bao gồm nhiễu, điều này gây khó khăn trong việc xác định số bước đi do các vị trí đỉnh bị méo dạng (không rõ ràng về hình dạng / vị trí đỉnh). Vì vậy, cần sử dụng bộ lọc thấp qua (Low pass filter). Trong đề tài này, bộ lọc Butterworth được áp dụng với tần số cắt là 2 Hz và bậc của bộ lọc là 7. Hình 9 mô tả giá trị của cảm biến gia tốc (trục z) trước và sau khi áp dụng bộ lọc.



Hình 9. Giá trị của cảm biến gia tốc (trục z) trước và sau khi áp dụng bộ lọc.

4. *Tính giá trị độ lớn* $\|\vec{A}\|$

Giá trị độ lớn từ ba trục của cảm biến gia tốc được tính theo công thức sau đây:

$$\|\vec{A}\| = \sqrt{ax_m^2 + ay_m^2 + az_m^2}$$

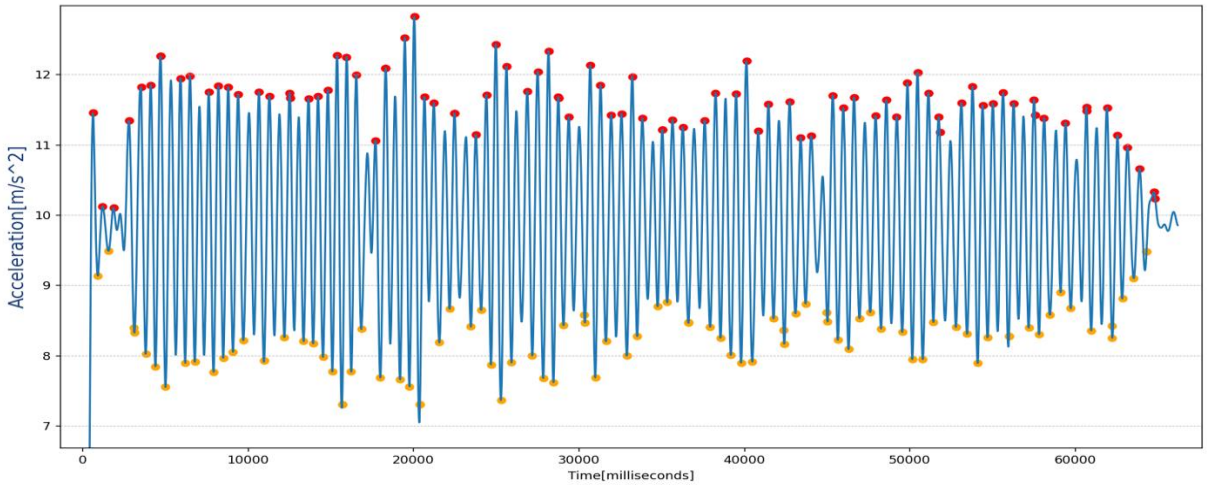
5. *Xác định giá trị đỉnh và đáy của* $\|\vec{A}\|$

Dựa trên giá trị độ lớn $\|\vec{A}\|$, các giá trị đỉnh lớn nhất (max) và nhỏ nhất (min) trong từng khung thời gian (window), độ trùng lắp 50%, được xác định.

6. *Loại bỏ các giá trị đỉnh gần nhau*

Các giá trị đỉnh lớn nhất / nhỏ nhất không thể xuất hiện liên tục trong thời gian quá ngắn. Ví dụ, với tốc độ bước đi bình thường, một người trong 1 giây chỉ đi được tối đa 2 bước, tức là khoảng cách thời gian giữa 2 đỉnh là 0.5 giây. Do đó, các đỉnh

xuất hiện mà khoảng cách thời gian giữa 2 đỉnh nhỏ hơn 0.5 giây cần được loại bỏ. Hình 10 mô tả kết quả sau khi loại bỏ các đỉnh bị xác định sai.



Hình 10. Kết quả các đỉnh lớn nhất / nhỏ nhất sau khi loại bỏ các đỉnh bị xác định sai.

7. Chọn lại giá trị đỉnh dựa trên sự khác biệt về độ lớn và thời gian

Các đỉnh lớn nhất/ nhỏ nhất được chọn lọc lại một lần nữa dựa trên hai tiêu chí: Sự khác biệt về độ lớn giữa đỉnh lớn nhất và nhỏ nhất gần kề nhau (ΔA) và sự khác biệt về thời gian giữa 2 đỉnh (ΔT). Hai giá trị ngưỡng ΔA và ΔT sẽ được thay đổi tùy vào tốc độ bước đi được xác định tại bước 2. Bảng tham số theo từng tốc độ được mô tả như sau.

Bảng 5. Bảng giá trị ngưỡng ΔA và ΔT theo từng tốc độ bước đi.

Tốc độ	Chậm	Vừa	Nhanh
Window	50	40	30
Overlap	40	35	25
ΔT	500	300	150
ΔA	2	2	2

8. Đếm số bước đi

Số bước đi được xác định sau khi đã chọn lọc lại các đỉnh lớn nhất thỏa các điều kiện tại bước 6 và bước 7.

3.2. Áp dụng trên phần mềm PyCharm

PyCharm là nền tảng được JetBrains phát triển như một IDE cho Python. Nó thường được sử dụng để phát triển Python ứng dụng. Một số ứng dụng lớn như Twitter, Facebook, Amazon và Pinterest cũng sử dụng PyCharm làm IDE Python của họ.



Hình 11. Ảnh nền PyCharm.

3.2.1. CÁC TÍNH NĂNG CỦA PYCHARM

Pycharm có thể chạy trên Windows, Linux, hoặc Mac OS. Ngoài ra, nó cũng chứa các Mô đun và các gói giúp các lập trình viên phát triển phần mềm bằng ngôn ngữ Python trong thời gian ngắn với ít công sức hơn. Hơn nữa, nó cũng có khả năng tùy chỉnh theo yêu cầu của nhà phát triển.

Khi sử dụng Pycharm, các lập trình viên có thể sử dụng một số tính năng sau:

***** Trình chỉnh sửa mã thông minh:

- Giúp các lập trình viên viết mã chất lượng cao.

- Bao gồm các lược đồ màu cho các từ khóa, lớp và hàm. Điều này giúp tăng khả năng đọc và hiểu mã.

- Xác định lỗi một cách dễ dàng.

- Cung cấp tính năng tự động hoàn thiện và hướng dẫn hoàn thiện mã.

***** Điều hướng mã:

- Giúp các nhà phát triển trong việc chỉnh sửa và nâng cao mã với ít nỗ lực và thời gian hơn.

- Với việc điều hướng mã, nhà phát triển có thể dễ dàng điều hướng một lớp, hàm hoặc tệp.

- LTV có thể xác định vị trí của một phần tử, một ký hiệu hoặc một biến trong mã nguồn trong thời gian ngắn khi sử dụng Pycharm.

- Bằng việc sử dụng chế độ thấu kính, nhà phát triển có thể kiểm tra và gỡ lỗi toàn bộ mã nguồn.

* Tái cấu trúc:

- Sử dụng Pycharm có lợi thế là thực hiện các thay đổi hiệu quả và nhanh chóng đối với cả biến cục bộ và biến toàn cục.

- Tái cấu trúc trong Pycharm cho phép các nhà phát triển cải thiện cấu trúc bên trong mà không thay đổi hiệu suất bên ngoài của mã.

- Nó cũng cho phép phân chia các lớp với các chức năng mở rộng hơn.

3.2.2. Giới thiệu ngôn ngữ python

Python là một ngôn ngữ lập trình bậc cao cho các mục đích lập trình đa năng, do Guido Van Rossum tạo ra và lần đầu ra mắt vào năm 1991. Python được thiết kế với ưu điểm mạnh là dễ đọc, dễ học và dễ nhớ. Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình và là ngôn ngữ lập trình dễ học; được dùng rộng rãi trong phát triển trí tuệ nhân tạo. Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu.

Python hoàn toàn tạo kiểu động và dùng cơ chế cấp phát bộ nhớ tự động; do vậy nó tương tự như Perl, Ruby, Scheme, Smalltalk, và Tcl. Python được phát triển trong một dự án mã mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý.

Ban đầu, Python được phát triển để chạy trên nền Unix. Nhưng rồi theo thời gian, Python dần mở rộng sang mọi hệ điều hành từ MS-DOS đến Mac OS, OS/2, Windows, Linux và các hệ điều hành khác thuộc họ Unix. Mặc dù sự phát triển của Python có sự đóng góp của rất nhiều cá nhân, nhưng Guido van Rossum hiện nay vẫn là tác giả chủ yếu của Python. Ông giữ vai trò chủ chốt trong việc quyết định hướng phát triển của Python.

Python luôn được xếp hạng vào những ngôn ngữ lập trình phổ biến nhất.



Hình 12. Ảnh ngôn ngữ Python

Các đặc điểm của Python:

Ngữ pháp đơn giản, dễ đọc.

- Vừa hướng thủ tục (procedural-oriented), vừa hướng đối tượng (object-oriented)

- Hỗ trợ module và hỗ trợ gói (package)

- Xử lý lỗi bằng ngoại lệ (Exception)

- Kiểu dữ liệu động ở mức cao.

- Có các bộ thư viện chuẩn và các module ngoài, đáp ứng tất cả các nhu cầu lập trình.

- Có khả năng tương tác với các module khác viết trên C /C++ (Hoặc Java cho Jython, hoặc .Net cho IronPython).

- Có thể nhúng vào ứng dụng như một giao tiếp kịch bản (scripting interface).

Python dễ dàng kết nối với các thành phần khác:

- Python có thể kết nối với các đối tượng COM, .NET (Ironpython, Python for .net), và CORBA, Java… Python cũng được hỗ trợ bởi Internet Communications Engine (ICE) và nhiều công nghệ kết nối khác.

- Có thể viết các thư viện trên C/C++ để nhúng vào Python và ngược lại.

- Python là ngôn ngữ có khả năng chạy trên nhiều nền tảng.

- Python có cho mọi hệ điều hành: Windows, Linux/Unix, OS/2, Mac, Amiga, và những hệ điều hành khác. Thậm chí có cả những phiên bản chạy trên .NET, máy ảo Java, và điện thoại di động (Nokia Series 60). Với cùng một mã nguồn sẽ chạy giống nhau trên mọi nền tảng.

- Python rất đơn giản và dễ học.

- Python có cộng đồng lập trình rất lớn, hệ thống thư viện chuẩn, và cả các thư viện mã nguồn mở được chia sẻ trên mạng.

- Python là ngôn ngữ mã nguồn mở.

- Cài đặt Python dùng giấy phép nguồn mở nên được sử dụng và phân tối tự do, ngay cả trong việc thương mại. Giấy phép Python được quản lý bởi Python Software Foundation.

CHƯƠNG 4: THU THẬP DỮ LIỆU NGƯỜI ĐI BỘ

4.1. Các thiết bị lấy mẫu dữ liệu

- Các thiết bị lấy dữ liệu gồm: Nexus 5, Galaxy S8+, Note 3, Galaxy A7. Các thiết bị đều sử dụng hệ điều hành Android.

- Số người tham gia lấy dữ liệu: 4 (người 1, người 2, người 3, người 4) cùng lấy dữ liệu ở cùng một thời điểm địa điểm giống nhau.

- Cài đặt app đo số liệu: Tốc độ lấy mẫu là 50 Hz.

- Tên app: SensorCollection, được viết trên nền tảng Android.



Hình 13. Ảnh nền app SensorCollection.

- Cách lấy dữ liệu:

+ Mở app SensorCollection, được giao diện như hình 14.



Hình 14. Giao diện của app SensorCollection.

+ Nhấn "Start" để bắt đầu lấy dữ liệu.

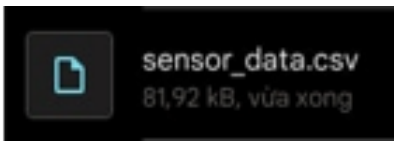
+ Nhấn "Save + Stop" để dừng và lưu dữ liệu thành file CSV (Hình 15). Các file CSV này được lưu vào thư mục "Download" trên điện thoại. Nội dung bên trong từng file được mô tả như trong Hình 16 với:

. Cột A: Time (ms) (thời gian).

. Cột B: ax_m (m/s^2) (giá trị của trục x của cảm biến gia tốc).

. Cột C: ay_m (m/s^2) (giá trị của trục y của cảm biến gia tốc).

. Cột D: az_m (m/s^2) (giá trị của trục z của cảm biến gia tốc).



Hình 15. File CSV sau khi lấy dữ liệu.

	A	B	C	D
1	Time	ax_m	ay_m	az_m
2	0	-0.10127041	7.457737	5.8497653
3	10	-0.20062548	7.7689705	4.6060314
4	47	-0.63036627	7.9353595	5.2452555
5	50	-0.58607537	7.9904246	6.224442
6	67	-0.39933568	8.019154	6.454275
7	87	-0.31793633	8.045488	5.813854
8	107	-0.37898585	7.986833	5.16625
9	127	-0.65430725	8.185543	5.153083
10	149	-0.63515437	8.143646	5.5193796
11	170	-0.62438095	8.17477	5.669011
12	189	-0.30117782	8.078612	5.208147
13	208	-0.33349797	7.904237	5.3266554
14	228	-0.4579911	7.65525	5.767169
15	249	-0.23893106	7.640686	5.86413

Hình 16. Nội dung bên trong file CSV.

4.2. Thu thập dữ liệu



Hình 13. Quá trình thu thập dữ liệu người đi bộ.

Dựa trên tìm hiểu về các kiểu cầm điện thoại nhóm thống nhất lấy mẫu dữ liệu với những điều kiện sau:

- Người lấy mẫu cầm điện thoại trên tay và để trước ngực và bước đi theo ba trường hợp.

  . Đi Chậm: 30 lần, 1 lần 100 bước.

  . Đi Vừa: 30 lần, 1 lần 100 bước.

  . Đi Nhanh: 30 lần, 1 lần 100 bước.

- Tất cả các trường hợp đều bước đi trên cùng một địa điểm, đi trên một đường thẳng.

- Ba tốc độ bước đi chậm, vừa, nhanh được ước tính như trong [22] và thể hiện trong bảng sau:

Kiểu di chuyển	Tốc độ ước tính (km/h)
Chậm	3.630±0.673
Vừa	4.886±0.618
Nhanh	6.370±0.729

- Các bước lấy dữ liệu:

  . Chạy ứng dụng đã cài đặt, chờ 5s để cho ứng dụng khởi động, gọi các giá trị cảm biến.

  . Bước đi theo phương song song với mặt đất có hướng tiến về phía trước đến vị trí được xác định bước thứ 100 rồi tiếp tục chờ 5s rồi bấm dừng ứng dụng để lưu kết quả vào điện thoại.

CHƯƠNG 5: KẾT QUẢ

Như đã trình bày ở trên, mỗi người đi bộ sẽ thu thập 30 files csv với mỗi tốc độ đi. Để áp dụng dữ liệu trên vào các mô hình học máy, 24 files của mỗi người cho mỗi tốc độ sẽ được dùng cho quá trình học (training), và 6 files còn lại sẽ được sử dụng để kiểm tra (testing). Từ các file kiểm tra, tốc độ nào xuất hiện nhiều lần nhất sẽ được sử dụng để chọn ra bộ tham số dùng cho việc xác định số bước đi. Kết quả xác định tốc độ bước đi của bốn người dựa trên năm kỹ thuật học máy được thể hiện trong các bảng từ Bảng 6 tới Bảng 17. Từ kết quả này có thể thấy ET cho kết quả tốt nhất cho cả ba tốc độ. Tốc độ bước đi vừa / bình thường có kết quả xác định với sai số lớn nhất. Lý do là trong quá trình bước đi với tốc độ này, sẽ có những lúc người đi bộ đi nhanh hơn bình thường, cũng có khi đi chậm hơn bình thường. Kết quả trung bình của việc xác định ba tốc độ bước đi dựa trên năm kỹ thuật học máy được biểu diễn trong Hình 14 tới Hình 16.

Bảng 6. Kết quả xác định tốc độ bước đi của người 1 (tốc độ chậm)

	accuracy (%)	f1-score
SVM	100.00	1.00
KNN	88.39	0.94
DT	91.91	0.93
RF	97.90	0.99
ET	97.15	0.99

Bảng 7. Kết quả xác định tốc độ bước đi của người 1 (tốc độ vừa)

	accuracy (%)	f1-score
SVM	16.43	1.00
KNN	66.49	0.94
DT	48.24	0.92
RF	46.67	0.99
ET	50.74	0.98

Bảng 8. Kết quả xác định tốc độ bước đi của người 1 (tốc độ nhanh)

	accuracy (%)	f1-score
SVM	90.01	0.94
KNN	87.39	0.96
DT	88.74	0.99
RF	89.59	0.98
ET	89.60	0.97

Bảng 9. Kết quả xác định tốc độ bước đi của người 2 (tốc độ chậm)

	accuracy (%)	f1-score
SVM	93.07	0.96
KNN	88.41	0.94
DT	84.23	0.91
RF	91.97	0.95
ET	90.87	0.95

Bảng 10. Kết quả xác định tốc độ bước đi của người 2 (tốc độ vừa)

	accuracy (%)	f1-score
SVM	30.76	0.46
KNN	59.25	0.74
DT	75.83	0.86
RF	83.40	0.91
ET	89.60	0.94

Bảng 11. Kết quả xác định tốc độ bước đi của người 2 (tốc độ nhanh)

	accuracy (%)	f1-score
SVM	94.43	0.81
KNN	94.69	0.81
DT	94.05	0.81
RF	96.11	0.81
ET	94.85	0.81

Bảng 12. Kết quả xác định tốc độ bước đi của người 3 (tốc độ chậm)

	accuracy (%)	f1-score
SVM	90.93	0.96
KNN	88.41	0.94
DT	84.23	0.94
RF	91.97	0.97
ET	90.87	0.96

Bảng 13. Kết quả xác định tốc độ bước đi của người 3 (tốc độ vừa)

	accuracy (%)	f1-score
SVM	73.59	0.84
KNN	68.77	0.81
DT	86.85	0.88
RF	94.21	0.97
ET	91.46	0.95

Bảng 14. Kết quả xác định tốc độ bước đi của người 3 (tốc độ nhanh)

	accuracy (%)	f1-score
SVM	97.74	0.99
KNN	96.38	0.98
DT	96.83	0.99
RF	97.74	0.99
ET	97.28	0.99

Bảng 15. Kết quả xác định tốc độ bước đi của người 4 (tốc độ chậm)

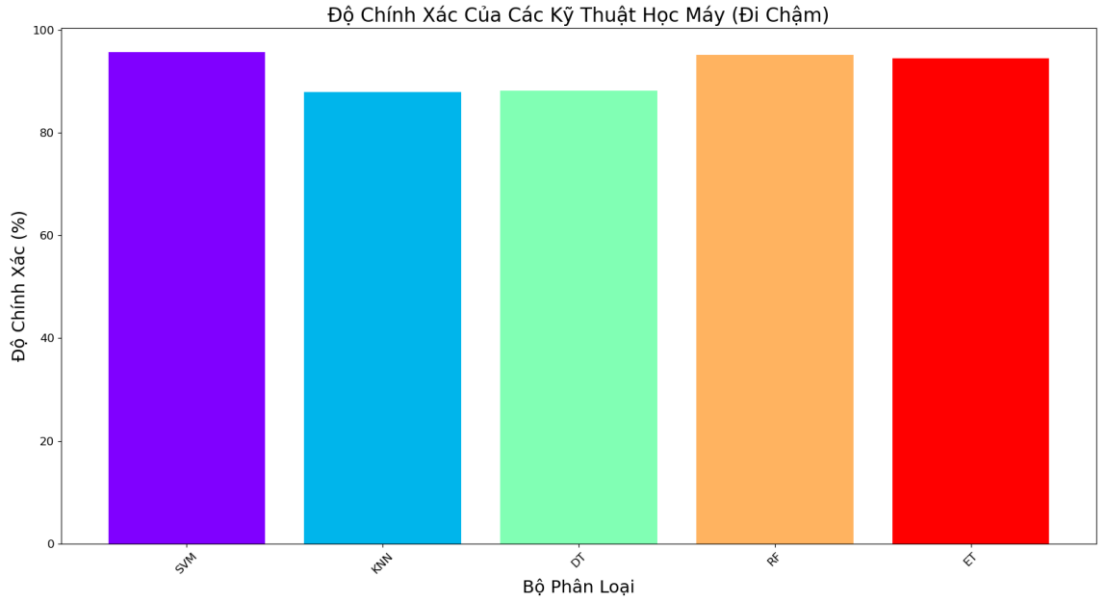
	accuracy (%)	f1-score
SVM	98.41	0.99
KNN	86.42	0.92
DT	92.16	0.97
RF	98.66	0.99
ET	98.93	1.00

Bảng 16. Kết quả xác định tốc độ bước đi của người 4 (tốc độ vừa)

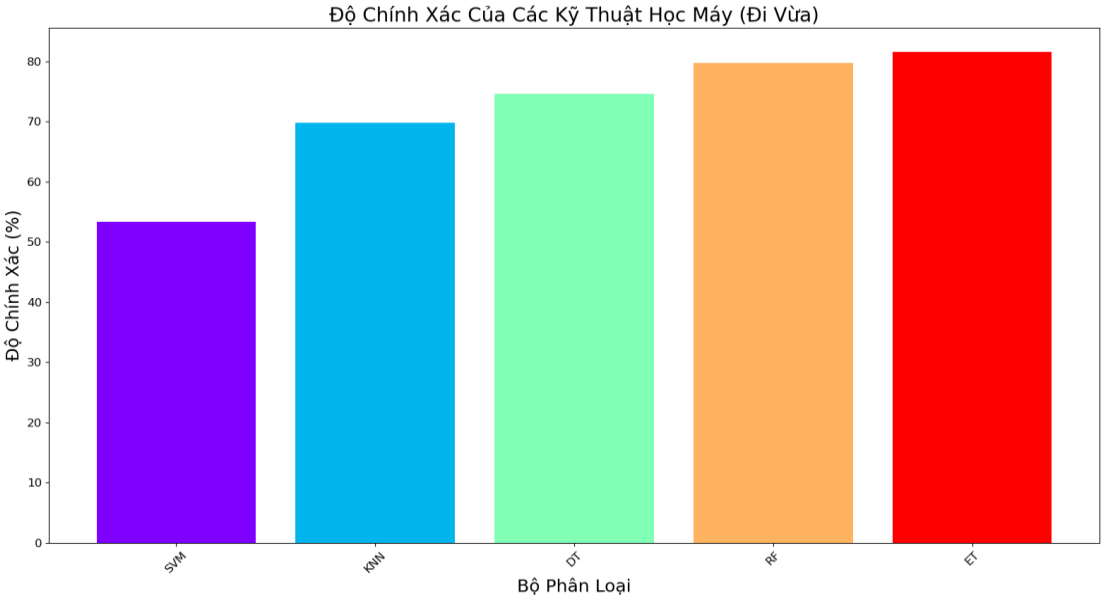
	accuracy (%)	f1-score
SVM	92.63	0.96
KNN	84.42	0.90
DT	87.47	0.91
RF	94.63	0.97
ET	94.21	0.96

Bảng 17. Kết quả xác định tốc độ bước đi của người 4 (tốc độ nhanh)

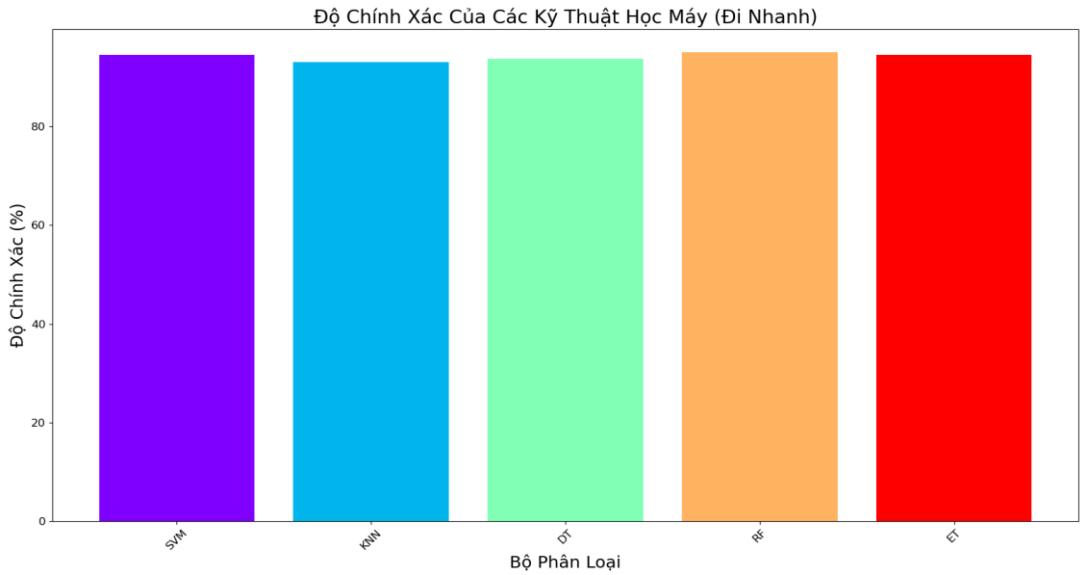
	accuracy (%)	f1-score
SVM	96.08	0.98
KNN	93.64	0.97
DT	95.57	0.98
RF	96.61	0.99
ET	96.60	0.98



Hình 17. Kết quả trung bình về độ chính xác trong việc xác định tốc độ bước đi chậm.



Hình 18. Kết quả trung bình về độ chính xác trong việc xác định tốc độ bước đi vừa.



Hình 19. Kết quả trung bình về độ chính xác trong việc xác định tốc độ bước đi nhanh.

Kết quả xác định số bước di chuyển của bốn người đi bộ với ba tốc độ bước đi khác nhau được thể hiện trong Bảng 18 tới Bảng 21. Kết quả trung bình của việc đếm số bước đi của bốn người được thể hiện trong Bảng 22 tới Bảng 25. Kết quả cho thấy phương pháp xác định số bước đi cho kết quả tốt với độ sai lệch trung bình dưới 3%. Hình 17 cho thấy sai số trung bình cho mỗi tốc độ bước đi của từng người đi bộ.

Bảng 18. Kết quả đếm số bước đi của người 1.

	Lần 1	Lần 2	Lần 3	Lần 4	Lần 5	Lần 6
Chậm	96	99	99	96	99	105
Vừa	95	99	94	98	96	98
Nhanh	103	97	98	98	100	99

Bảng 19. Kết quả đếm số bước đi của người 2.

	Lần 1	Lần 2	Lần 3	Lần 4	Lần 5	Lần 6
Chậm	100	100	101	95	101	96
Vừa	101	97	100	97	95	91
Nhanh	100	98	98	99	98	96

Bảng 20. Kết quả đếm số bước đi của người 3.

	Lần 1	Lần 2	Lần 3	Lần 4	Lần 5	Lần 6
Chậm	100	97	100	98	101	97
Vừa	101	103	98	103	98	96
Nhanh	95	101	101	102	99	96

Bảng 21. Kết quả đếm số bước đi của người 4.

	Lần 1	Lần 2	Lần 3	Lần 4	Lần 5	Lần 6
Chậm	95	100	98	99	100	97
Vừa	100	98	96	105	101	106
Nhanh	96	95	95	98	99	102

Bảng 22. Kết quả trung bình đếm số bước đi của người 1.

Tốc độ	Phát hiện bước	Số bước sai lệch	Tỷ lệ sai lệch (%)
Chậm	97/100	-3	3%
Vừa	97/100	-3	3%
Nhanh	98/100	-2	2%

Bảng 23. Kết quả trung bình đếm số bước đi của người 2.

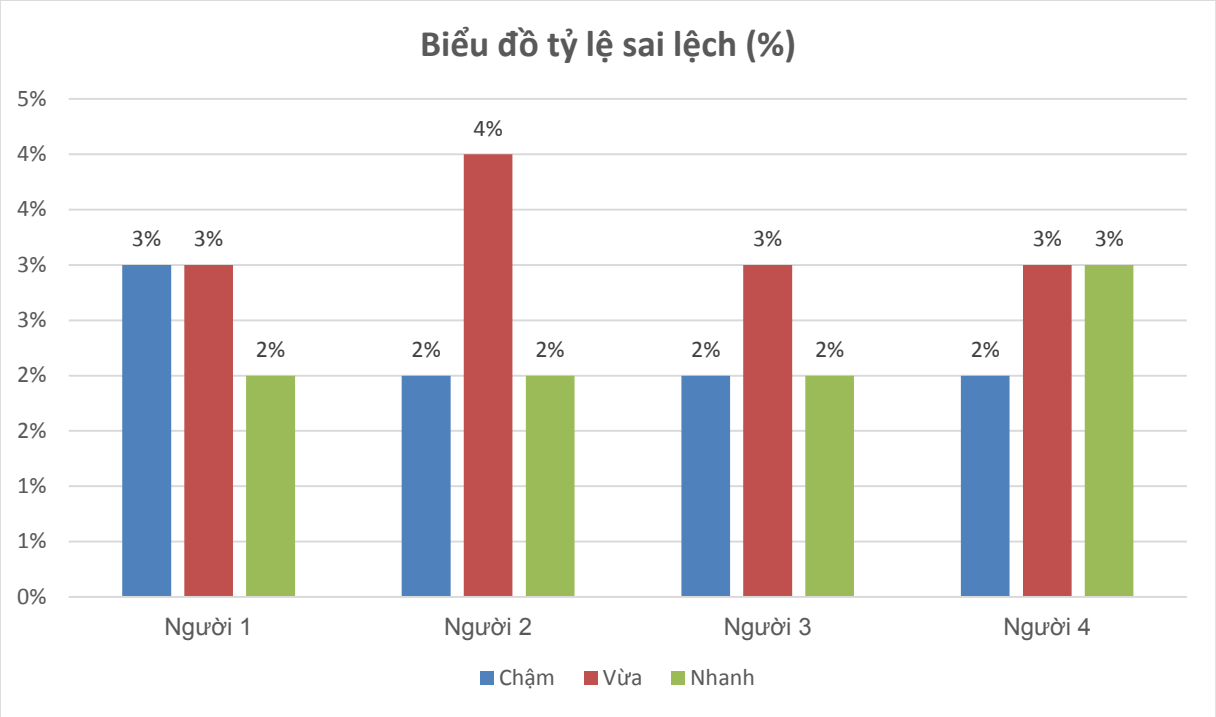
Tốc độ	Phát hiện bước	Số bước sai lệch	Tỷ lệ sai lệch (%)
Chậm	98/100	-2	2%
Vừa	96/100	-4	4%
Nhanh	98/100	-2	2%

Bảng 24. Kết quả trung bình đếm số bước đi của người 3.

Tốc độ	Phát hiện bước	Số bước sai lệch	Tỷ lệ sai lệch (%)
Chậm	98/100	-2	2%
Vừa	97/100	-3	3%
Nhanh	98/100	-2	2%

Bảng 25. Kết quả trung bình đếm số bước đi của người 4.

Tốc độ	Phát hiện bước	Số bước sai lệch	Tỷ lệ sai lệch (%)
Chậm	98/100	-2	2%
Vừa	97/100	-3	3%
Nhanh	97/100	-3	3%



Hình 20. Kết quả trung bình về độ chính xác đếm số bước đi ở ba tốc độ di chuyển.

CHƯƠNG 6: KẾT LUẬN

6.1 Kết luận.

Bằng cách sử dụng các giá trị của cảm biến gia tốc tích hợp trên điện thoại thông minh, đề tài đã giải quyết được hai vấn đề được đặt ra là:

- Áp dụng các kỹ thuật học máy để xác định các tốc độ di chuyển với kết quả chính xác trung bình hơn 90% với ba trên bốn người đi bộ.

- Xây dựng và phát triển thuật toán xác định số bước đi của người đi bộ dựa trên ba tốc độ bước đi khác nhau và đạt độ chính xác lên đến hơn 95%.

6.2. Hạn chế.

Trong quá trình thực nghiệm, một số trường hợp lấy dữ liệu bị lỗi, ví dụ như người bước đi không đủ 100 bước hoặc bước đủ nhưng bước quá nhẹ nên khi đi qua bộ lọc đã lược bỏ mất một số bước, do đó, những điều này làm ảnh hưởng tới kết quả đếm số bước đi.

6.3. Hướng phát triển.

- Tối ưu lại các đặc trưng sử dụng trong các kỹ thuật học máy để chọn ra các đặc trưng quan trọng nhất, từ đó giảm số lượng đặc trưng cần sử dụng.

- Tìm thuật toán xác định số bước chân tốt hơn để tìm ra chính xác hơn số bước chân của người đi bộ, đặc biệt trong điều kiện tốc độ bước đi thay đổi.

TÀI LIỆU THAM KHẢO

[1] R. Stirling, J. Collin, K. Fyfe, and G. Lachapelle, "An innovative shoe-mounted pedestrian navigation system," in *proceedings of European navigation conference GNSS*, 2003, vol. 110, no. 5: Citeseer.

[2] B. Krach and P. Roberston, "Cascaded estimation architecture for integration of foot-mounted inertial sensors," in *2008 IEEE/ION Position, Location and Navigation Symposium*, 2008, pp. 112-119.

[3] N. Castaneda and S. Lamy-Perbal, "An improved shoe-mounted inertial navigation system," in *2010 International Conference on Indoor Positioning and Indoor Navigation*, 2010, pp. 1-6: IEEE.

[4] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proceedings of the 10th international conference on Ubiquitous computing*, 2008, pp. 114-123.

[5] L. Ojeda and J. Borenstein, "Non-GPS navigation with the personal dead-reckoning system - art. no. 65610C," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 1, 05/01 2007.

[6] F. Cavallo, A. M. Sabatini, and V. Genovese, "A step toward GPS/INS personal navigation systems: real-time assessment of gait by foot inertial sensing," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 1187-1191.

[7] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, pp. 38-46, 2005.

[8] A. R. Jimenez, F. Seco, C. Prieto, and J. Guevara, "A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU," in *2009 IEEE International Symposium on Intelligent Signal Processing*, 2009, pp. 37-42.

[9] F. Lei *et al.*, "Design of a wireless assisted pedestrian dead reckoning system - the NavMote experience," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 6, pp. 2342-2358, 2005.

[10] S. Leonhardt, T. Falck, and P. Mähönen, *4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007): March 26-28, 2007 RWTH Aachen University, Germany*. Springer Science & Business Media, 2007.

[11] P. Goyal, V. J. Ribeiro, H. Saran, and A. Kumar, "Strap-down Pedestrian Dead-Reckoning system," in *2011 International Conference on Indoor Positioning and Indoor Navigation*, 2011, pp. 1-7.

[12] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer*, vol. 34, no. 8, pp. 57-66, 2001.

[13] Q. Ladetto, "On foot navigation: Continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering," *Proceedings of ION GPS,* 01/01 2000.

[14] M. Dippold, "Personal Dead Reckoning with Accelerometers," in *3rd International Forum on Applied Wearable Computing 2006*, 2006, pp. 1-6.

[15] C. Randell, C. Djiallis, and H. Muller, *Personal Position Measurement Using Dead Reckoning*. 2005, pp. 166-173.

[16] H. Weinberg, "Using the ADXL 202 in Pedometer and Personal Navigation Applications," 2002.

[17] S. Yang and Q. Li, "Ambulatory walking speed estimation under different step lengths and frequencies," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2010, pp. 658-663.

[18] J. Saarinen, J. Suomela, S. Heikkila, M. Elomaa, and A. Halme, "Personal navigation system," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, 2004, vol. 1, pp. 212-217 vol.1.

[19] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *the Proceedings of the 18th annual international conference on Mobile computing and networking*, Istanbul, Turkey, 2012.

[20] S. Godha, G. Lachapelle, and E. Cannon, "Integrated GPS/INS System for Pedestrian Navigation in a Signal Degraded Environment," *ION GNSS,* 01/01 2006.

[21] Widyawan, M. Klepal, and S. Beauregard, "A Backtracking Particle Filter for fusing building plans with PDR displacement estimates," in *2008 5th Workshop on Positioning, Navigation and Communication*, 2008, pp. 207-212.

[22] Zhang, K., et al. (2005). "Assessment of human locomotion by using an insole measurement system and artificial neural networks." *Journal of Biomechanics 38(11)*, pp. 2276-2287.