

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC ỨNG DỤNG – BỘ MÔN VẬT LÝ KỸ THUẬT

__2022 – 2023__



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
THIẾT BỊ LUYỆN TẬP
ĐẮM BỐC THÔNG MINH

TP.HCM 06/2023

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC ỨNG DỤNG – BỘ MÔN VẬT LÝ KỸ THUẬT

__2022 – 2023__



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
THIẾT BỊ LUYỆN TẬP
ĐẮM BỐC THÔNG MINH

Ngành: Vật lý kỹ thuật

Sinh viên thực hiện
Bùi An Khang – 1913683

Giảng viên hướng dẫn
TS. Nguyễn Trung Hậu

TP.HCM 06/2023

ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA

CỘNG HÒA XÃ HỘI CHỦ NGHĨA
VIỆT NAM

Độc lập – Tự do – Hạnh phúc

KHOA KHOA HỌC ỨNG DỤNG
BỘ MÔN VẬT LÝ KỸ THUẬT Y SINH

Tp.Hồ Chí Minh, ngày ... tháng ... năm 2023

NHIỆM VỤ LUẬN VĂN TỐT NGHIỆP

HỌ VÀ TÊN

MSSV

Bùi An Khang

1913683

NGÀNH: VẬT LÝ KỸ THUẬT

LỚP: KU19VLY1

- Đề tài luận văn: Thiết bị luyện tập boxing thông minh.
- Nhiệm vụ:
 - Nghiên cứu các cú đấm và lực trong luyện tập boxing.
 - Tìm hiểu phương pháp đo lực.
 - Tìm hiểu các thiết bị đo lực trên thị trường.
 - Thiết kế và chế tạo các thiết bị đo lực và gia tốc.
 - Huấn luyện mô hình hồi quy dữ liệu thu được.
 - Đánh giá kết quả mô hình.
- Ngày giao nhiệm vụ luận văn: 01/01/2022
- Ngày hoàn thành nhiệm vụ: 23/05/2023
- Họ và tên người hướng dẫn: TS.Nguyễn Trung Hậu.

Tên đề tài và nội dung LVTN đã được thông qua Bộ môn.

NGƯỜI HƯỚNG DẪN

(Ký và ghi rõ họ tên)

CHỦ NHIỆM BỘ MÔN

(Ký và ghi rõ họ tên)

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn thầy cô bộ môn Kỹ thuật Y Sinh - Khoa Khoa Học Ứng Dụng - Đại học Bách Khoa TP HCM, thầy cô đã tận tụy và tạo mọi điều kiện để chúng em hoàn thành đồ án tốt nghiệp. Bên cạnh đó chúng em cũng gửi lời thầy Nguyễn Trung Hậu, người đã đồng hành cùng chúng em để thực hiện đồ án. Không thể thiếu là đội ngũ hùng hậu làm luận văn chung với em, những người đã đi trước em một vài lĩnh vực, những người đã hướng dẫn tận tình, chỉ dạy cho em những thứ mà nếu không có họ, em đã phải mất rất nhiều thời gian. Chúng em xin dành những lời cảm ơn chân thành, những lời tri ân sâu sắc nhất đến với thầy Huỳnh Quanh Linh và thầy Mai Hữu Xuân, người đã giúp đỡ em rất nhiều trong việc lên bản thiết kế và làm mô hình cơ khí cho các thiết bị. Nếu không nhờ vào sự dạy bảo tận tình, tấm lòng nhiệt huyết giúp chúng em có những kiến thức để thực hiện đồ án. Em chúc thầy cô luôn có nhiều sức khỏe, luôn tận tâm, nhiệt huyết, giữ được ngọn lửa để dìu dắt những thế hệ học sinh tiếp theo. Bài báo cáo sẽ có những thiếu sót là điều chắc chắn, em rất mong nhận được những ý kiến đóng góp quý báu của thầy cô để kiến thức của em được hoàn thiện hơn.

TÓM TẮT LUẬN VĂN

Hiện nay có nhiều nghiên cứu về lực đẩy với các tính toán cơ học và sử dụng cảm biến lực, gia tốc,... Tuy nhiên với sự phát triển mạnh mẽ của trí tuệ nhân tạo, đã tạo ra nhiều phương pháp khác và mang độ chính xác vượt trội so với các cách tính thông thường. Nội dung của luận văn hướng đến ứng dụng cho người sử dụng, bằng cách đo lực đẩy trong lúc luyện tập, các vận động viên sẽ biết được họ cần phải thay đổi gì để trở nên mạnh mẽ hơn. Ngoài ra luận văn còn so sánh mô hình hồi quy với các phương pháp khác như đo trực tiếp bằng loadcell hay tính bằng cơ học.

ABSTRACT

Currently, there are many studies on punching force that involve mechanical calculations and the use of force sensors, accelerometers, and other devices. However, with the powerful development of artificial intelligence, various methods have been created that offer superior accuracy compared to conventional approaches. The content of the thesis aims to apply these methods to benefit the users by measuring punching force during training. Athletes will then be able to determine what changes they need to make in order to become stronger. Additionally, the thesis also compares regression models with other methods such as direct measurement using load cells or mechanical calculations.

MỤC LỤC

LỜI CẢM ƠN.....	i
TÓM TẮT LUẬN VĂN.....	ii
ABSTRACT	iii
MỤC LỤC	iv
DANH SÁCH HÌNH VẼ.....	vii
DANH SÁCH BẢNG BIỂU	x
DANH MỤC TỪ VIẾT TẮT	xi
CHƯƠNG 1 GIỚI THIỆU	1
1.1 LÝ DO CHỌN ĐỀ TÀI.....	1
1.2 MỤC TIÊU NGHIÊN CỨU	1
1.3 ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU.....	1
CHƯƠNG 2 TỔNG QUAN	2
2.1 TỔNG QUAN VỀ BOXING	2
2.1.1 LỰC ĐÁM TRONG BOXING.....	2
2.1.2 LỊCH SỬ NGHIÊN CỨU LỰC ĐÁM TRONG BOXING	3
2.1.3 PHÂN TÍCH ĐỘNG LỰC HỌC CỦA BAO ĐÁM.....	4
2.1.4 ƯỚC LƯỢNG MOMENT QUÁN TÍNH CỦA BAO ĐÁM DỰA TRÊN HÌNH HỌC.....	6
2.2 TỔNG QUAN VỀ MẠNG NƠ-RON TÍCH CHẬP (CNN)	6
2.2.1 LỚP TÍCH CHẬP	7
2.2.2 LỚP GỘP (MAX POOLING).....	7
2.2.3 HÀM KÍCH HOẠT (ACTIVATION FUNCTION).....	8
2.2.4 HÀM KÍCH HOẠT RELU	8
2.2.5 HÀM KÍCH HOẠT SOFTMAX	9

2.2.6	CHUẨN HÓA DỮ LIỆU (NORMALIZATION)	10
CHƯƠNG 3	THIẾT BỊ PHẦN CỨNG VÀ PHẦN MỀM	11
3.1	THIẾT BỊ PHẦN CỨNG.....	11
3.1.1	VI ĐIỀU KHIỂN ESP32.....	12
3.1.2	CẢM BIẾN GIA TỐC LSM6DSO32	13
3.1.3	CẢM BIẾN LỰC DẠNG LOADCELL	16
3.1.4	MẠCH NGUỒN CHO ESP32 VÀ SẠC PIN LIPO	19
3.1.5	THIẾT KẾ CHI TIẾT THIẾT BỊ.....	20
3.1.6	THIẾT BỊ ĐO LỰC THAM CHIẾU	21
3.2	PHẦN MỀM (SOFTWARE).....	23
3.2.1	MÃ NGUỒN CỦA VI ĐIỀU KHIỂN	23
3.2.2	GIAO DIỆN THU DỮ LIỆU	26
3.2.3	GIAO DIỆN NGƯỜI DÙNG.....	28
CHƯƠNG 4	PHƯƠNG PHÁP ĐO VÀ XỬ LÝ TÍN HIỆU	31
4.1	THU DỮ LIỆU LỰC VÀ GIA TỐC	31
4.1.1	QUY TRÌNH THÍ NGHIỆM	31
4.2	XÂY DỰNG MÔ HÌNH HỒI QUY DỰ ĐOÁN LỰC	32
4.3	PHÂN LOẠI CÚ ĐÁM	33
4.3.1	PHƯƠNG PHÁP NGƯỠNG (THRESHOLD).....	34
4.3.2	XÂY DỰNG MÔ HÌNH PHÂN LOẠI (CLASSIFICATION)	35
4.4	DỰ ĐOÁN LỰC BẰNG MÔ HÌNH THỜI GIAN THỰC	37
CHƯƠNG 5	KẾT QUẢ VÀ ĐÁNH GIÁ.....	39
5.1	KẾT QUẢ	39
5.1.1	DỮ LIỆU GIA TỐC VÀ LỰC.....	39
5.1.2	PHÂN TÍCH DỮ LIỆU LỰC	40

5.1.3	ĐƯỜNG ĐẶC TÍNH CỦA LOADCELL	41
5.1.4	TẦN SỐ LẤY MẪU CỦA THIẾT BỊ.....	41
5.1.5	PHÂN TÍCH DỮ LIỆU GIA TỐC	42
5.1.6	SỰ TƯƠNG QUAN GIỮA GIA TỐC VÀ LỰC	43
5.1.7	MOMENT QUÁN TÍNH BAO ĐẮM	46
5.1.8	SO SÁNH LỰC ĐO BẰNG LOADCELL – DỰ ĐOÁN BẰNG MÔ HÌNH – TÍNH BẰNG LÝ THUYẾT	46
5.1.9	KẾT QUẢ MÔ HÌNH PHÂN LOẠI CÚ ĐẮM	49
5.2	ĐÁNH GIÁ.....	50
5.2.1	ĐÁNH GIÁ MÔ HÌNH DỰ ĐOÁN LỰC.....	50
5.2.2	ĐÁNH GIÁ GIỮA VIỆC SỬ DỤNG NGUỒN VÀ MÔ HÌNH PHÂN LOẠI TRONG VIỆC NHẬN DẠNG CÚ ĐẮM THỜI GIAN THỰC.....	51
5.2.3	ĐÁNH GIÁ PHẦN MỀM THỜI GIAN THỰC.....	51
	TÀI LIỆU THAM KHẢO	52

DANH SÁCH HÌNH VẼ

Hình 2.1 Các loại đấm cơ bản trong boxing, bao gồm jab, cross, hook, uppercut.....	2
Hình 2.2 Kích thước và vị trí treo của bao đấm.	4
Hình 2.3 Đồ thị hàm kích hoạt ReLu [5].....	9
Hình 3.1 Các khối chức năng cơ bản của ESP32 [8].	12
Hình 3.2 Sơ đồ chân vào ra của mạch Doit Devkit V1 [9].	13
Hình 3.3 Các chân ngoại vi và trục của cảm biến [6].	13
Hình 3.4 Các trục O_x , O_y , O_z của cảm biến gia tốc dài/góc LSM6DSO32.....	14
Hình 3.5 Loadcell 20kg [10].	16
Hình 3.6 Quy trình tìm đường đặc tính của loadcell.	18
Hình 3.7 Mạch nguồn cho vi điều khiển (VDK) ESP32.	19
Hình 3.8 Pin LIPO để cấp nguồn cho VDK ESP32.	19
Hình 3.9 Sơ đồ nguyên lý mạch nguồn cho VDK ESP32 [11].	19
Hình 3.10 Sơ đồ mạch nguyên lý của thiết bị.	20
Hình 3.11 PCB Mặt trước và mặt sau của thiết bị.....	20
Hình 3.12 Mạch thiết bị đo lực tham chiếu.	21
Hình 3.13 PCB Mặt trước và mặt sau của thiết bị loadcell tham chiếu.	22
Hình 3.14 Loadcell và tấm phẳng chịu lực có kích thước dài – rộng – cao lần lượt là $17 \times 14 \times 0.8 \text{ cm}$	22
Hình 3.15 Bảng vẽ kích thước của loadcell [10].	23
Hình 3.16 Phần mềm Arduino IDE [13].	23
Hình 3.17 Sơ đồ khối thu dữ liệu từ cảm biến.	24
Hình 3.18 Giao diện thu dữ liệu gia tốc dài và gia tốc góc trên bao đấm.	26
Hình 3.19 Giao diện thu dữ liệu lực tham chiếu bằng loadcell.	26
Hình 3.20 Mô hình máy trạng thái (finite state machine) của phần mềm thu dữ liệu gia tốc và lực.	26
Hình 3.21 Sơ đồ khối thu dữ liệu gia tốc và lực.....	27
Hình 3.22 Giao diện chính.....	28
Hình 3.23 Giao diện chọn port để kết nối tới Bluetooth.	29
Hình 3.24 Mô hình máy trạng thái của phần mềm.	30

Hình 4.1 Đặt vị trí 2 thiết bị trên bao đệm.....	31
Hình 4.2 Quy trình thu dữ liệu từ loadcell và gia tốc.....	32
Hình 4.3 Khác biệt về gia tốc dài giữa dữ liệu có đệm và không đệm.	34
Hình 4.4 Lưu đồ thuật toán phát hiện cú đầm sử dụng ngưỡng.....	35
Hình 4.5 Sơ đồ khối hoạt động của phần mềm.	38
Hình 5.1 Dữ liệu gia tốc và lực thu được từ cảm biến gai tốc và cảm biến lực loadcell.	39
Hình 5.2 Thời gian bao tay tiếp xúc với bao đệm trong một cú đầm.....	40
Hình 5.3 Đường đặc tính của loadcell trong đoạn 0 – 30kg.....	41
Hình 5.4 Đường đặc tính của loadcell trong đoạn 30 – 45kg.....	41
Hình 5.5 Thông tin một cú đầm theo phương Oy trong 100 ms với tần số lấy mẫu 1000Hz.....	42
Hình 5.6 Dữ liệu gia tốc dài khi tác dụng lực F theo phương Ox trong khoảng thời gian diễn ra một cú đầm (100 ms).	43
Hình 5.7 Dữ liệu gia tốc dài khi tác dụng lực F theo phương Oy trong khoảng thời gian diễn ra một cú đầm (100 ms).	43
Hình 5.8 Dữ liệu gia tốc góc khi tác dụng lực F theo phương Ox trong khoảng thời gian diễn ra một cú đầm (100 ms).	43
Hình 5.9 Dữ liệu gia tốc góc khi tác dụng lực F theo phương Oy trong khoảng thời gian diễn ra một cú đầm (100 ms).	43
Hình 5.10 Tương quan giữa gia tốc dài theo phương Ox và lực đạt 0.657057.....	45
Hình 5.11 Tương quan giữa gia tốc dài theo phương Oy và lực đạt 0.637418.....	45
Hình 5.12 Tương quan giữa gia tốc dài theo phương Oz và lực đạt 0.829592.	45
Hình 5.13 Tương quan giữa gia tốc góc theo phương Ox và lực đạt 0.617840.....	45
Hình 5.14 Tương quan giữa gia tốc góc theo phương Oy và lực đạt 0.563325.....	45
Hình 5.15 Tương quan giữa gia tốc góc theo phương Oz và lực đạt 0.651737.	45
Hình 5.16 Tương quan giữa gia tốc dài tổng hợp và lực đạt 0.926602.....	46
Hình 5.17 Tương quan giữa gia tốc góc tổng hợp và lực đạt 0.885718.....	46
Hình 5.18 So sánh lực giữa đo tham chiếu bằng loadcell và lực dự đoán từ mô hình.	47

Hình 5.19 Biểu đồ Bland–Altman giữa giá trị lực tham chiếu bằng loadcell và lực dự đoán bằng mô hình.	47
Hình 5.20 So sánh lực giữa tính bằng lý thuyết và lực dự đoán từ mô hình.	47
Hình 5.21 Biểu đồ Bland–Altman giữa giá trị lực được tính bằng lý thuyết và lực bằng mô hình.	47
Hình 5.22 Tương quan giữa lực đo bằng loadcell và lực đo bằng mô hình.	48
Hình 5.23 Tương quan giữa lực đo bằng lý thuyết và lực đo bằng mô hình.	48
Hình 5.24 Xác xuất nhận dạng cú đấm của mô hình theo thời gian thực: (a) Cửa sổ dữ liệu hoàn toàn không có cú đấm, xác xuất có đấm ~0.00. (b) Cửa sổ dữ liệu có một phần dữ liệu đầu cú đấm, xác xuất có đấm ~0.21. (c) Cửa sổ dữ liệu có toàn bộ dữ liệu cú đấm, xác xuất có đấm ~0.99. (d) Cửa sổ dữ liệu có một phần dữ liệu sau cú đấm, xác xuất có đấm ~0.00.	49
Hình 5.25 Chương trình chạy thời gian thực dựa trên giải thuật phân loại cú đấm sử dụng ngưỡng (threshold).	50

DANH SÁCH BẢNG BIỂU

Bảng 2.1 Thông tin chi tiết của bao đảm.....	4
Bảng 3.1 Các chân của cảm biến lsm6dso32 được sử dụng.....	14
Bảng 3.2 Thanh ghi WHO_AM_I.....	14
Bảng 3.3 Thanh ghi CTRL1_XL.....	14
Bảng 3.4 Thanh ghi CTRL2_G.	15
Bảng 3.5 Thanh ghi CTRL6_C chỉnh tần số lấy mẫu của gia tốc dài.	15
Bảng 3.6 Thanh ghi CTRL7_C chỉnh tần số lấy mẫu của gia tốc góc.	15
Bảng 3.7 Địa chỉ 12 thanh ghi gia tốc dài và gia tốc góc.....	15
Bảng 3.8 Các thông số cơ bản của loadcell [8].	16
Bảng 3.9 Từng tâm đo sẽ có giá trị mg/LSB hoặc mdps/LSB khác nhau [6].	25
Bảng 4.1 Các thông số của mô hình CNN được sử dụng cho bài toán hồi quy.....	32
Bảng 4.2 Giải thích các thông số của mô hình hồi quy.....	33
Bảng 4.3 Các thông số được sử dụng cho mô hình phân loại.	35
Bảng 4.4 Giải thích các thông số của mô hình phân loại [14].	36
Bảng 5.1 Đánh giá phương pháp sử dụng ngưỡng và mô hình phân loại.	51

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Từ đầy đủ
1	SR	Sampling Rate
2	CNN	Convolutional neural network
3	VDK	Vi điều khiển

CHƯƠNG 1 GIỚI THIỆU

1.1 LÝ DO CHỌN ĐỀ TÀI

Hiện nay trên thị trường có nhiều thiết bị giúp cho vận động viên luyện tập [1], các thiết bị không còn mới nữa, trên thế giới có những bài báo về phân loại cú đấm bằng cách sử dụng mô hình học máy [2]. Tuy nhiên sử dụng công nghệ trí tuệ nhân tạo trong việc giúp đỡ các vận động viên còn mới ở Việt Nam trong những năm gần đây. Học máy là một lựa chọn hiệu quả để dự đoán lực đánh trong quyền Anh vì chúng có khả năng học và trích xuất các đặc trưng từ đầu vào phức tạp và nhiều chiều như hình ảnh hoặc video. Trong trường hợp của quyền Anh, điều này có thể bao gồm các hình ảnh hoặc video của các võ sĩ tung ra những cú đấm, trong đó lực đánh liên quan đến các yếu tố như tư thế của võ sĩ, tốc độ của cú đấm, góc đánh và nhiều hơn nữa. Đó là lý do nhóm chúng em chọn đề tài này.

1.2 MỤC TIÊU NGHIÊN CỨU

Đề tài của nhóm chúng tôi sẽ tập trung vào việc thiết kế tạo nên một thiết bị giúp theo dõi quá trình tập luyện của vận động viên boxing để huấn luyện viên hay vận động viên đó có thể đánh giá được sự hiệu quả của quá trình tập luyện, từ đó cải thiện các chế độ tập luyện để đạt sự hiệu quả tốt hơn.

1.3 ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

Nghiên cứu hướng đến các vận động viên quyền Anh, các huấn luyện viên của các vận động viên quyền Anh.

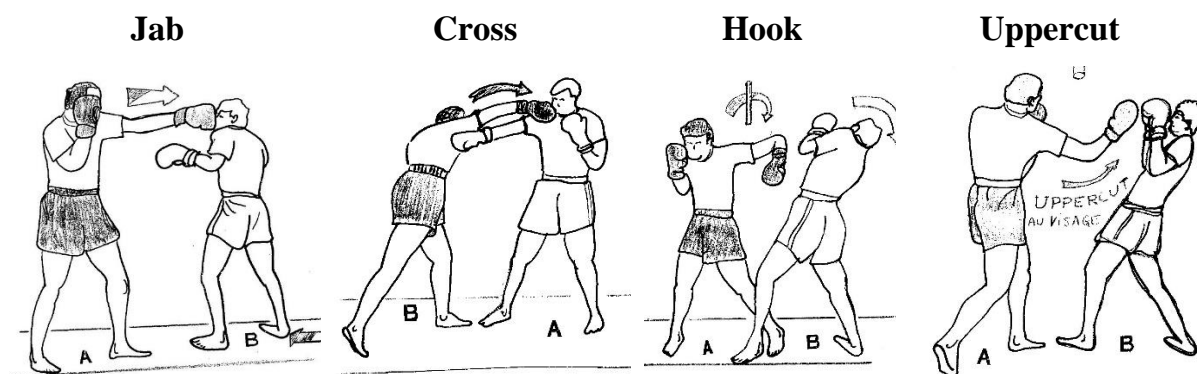
CHƯƠNG 2 TỔNG QUAN

2.1 TỔNG QUAN VỀ BOXING

Trong Quyền Anh chia thành Quyền Anh nghiệp dư và Quyền Anh chuyên nghiệp. Quyền Anh nghiệp dư là một nội dung thi đấu của nhiều đại hội thể thao trên toàn thế giới, trong đó có Thế vận hội. Đây là môn thi đấu tiêu chuẩn trong hầu hết các giải thể thao quốc tế - và môn này cũng có giải vô địch thế giới của riêng mình. Quyền Anh có hệ thống quy định riêng, được giám sát bởi trọng tài trong những khung thời gian nhất định, các khoảng thời gian từ một đến ba phút được gọi là các hiệp đấu. Trong trận đấu, người thắng cuộc có thể giành được chiến thắng trước khi kết thúc hiệp đấu khi trọng tài cho rằng đối phương không có khả năng tiếp tục trận đấu, truất quyền thi đấu của đối thủ hoặc đối thủ xin thua. Khi trận đấu kết thúc hiệp cuối cùng mà cả hai đối thủ vẫn đứng vững, phiếu ghi điểm của trọng tài sẽ xác định người chiến thắng. Trong trường hợp cả hai võ sĩ đạt được số điểm bằng nhau từ các trọng tài, thì cuộc đấu chuyên nghiệp được coi là hòa. Trong môn Quyền Anh ở Thế vận hội, vì phải tuyên bố người chiến thắng, không có trận hòa, nên các trọng tài sẽ quyết định người thắng dựa trên tiêu chí kỹ thuật.

2.1.1 LỰC ĐÁM TRONG BOXING

Trong Boxing được chia ra làm 4 loại đâm cơ bản: jab, cross, hook, uppercut.



Hình 2.1 Các loại đâm cơ bản trong boxing, bao gồm jab, cross, hook, uppercut.

Jab - Một cú đâm nhanh, thẳng được tung ra với tay đầu tiên từ tư thế bảo vệ. Jab được mở rộng từ bên cạnh thân hình và thường không vượt qua phía trước của nó. Nó

được kèm theo một sự xoay nhỏ theo chiều kim đồng hồ của thân hình và hông, trong khi nắm đấm quay 90 độ, trở thành ngang khi tác động. Khi cú đấm đạt đến độ kéo dài đầy đủ, vai tay đầu tiên có thể được đưa lên để bảo vệ cằm. Tay sau vẫn ở bên mặt để bảo vệ hàm. Sau khi tiếp xúc với mục tiêu, tay đầu tiên được thu hồi nhanh chóng để tiếp tục tư thế bảo vệ trước mặt.

Cross - Một cú đấm thẳng mạnh được tung ra với tay sau. Từ tư thế bảo vệ, tay sau được tung ra từ cằm, băng qua cơ thể và di chuyển về phía mục tiêu theo một đường thẳng. Vai tay sau được đẩy về phía trước và kết thúc chỉ chạm vào bên ngoài cằm. Đồng thời, tay đầu tiên được thu hồi và gấp vào mặt để bảo vệ bên trong cằm. Để tăng thêm sức mạnh, thân hình và hông được xoay ngược chiều kim đồng hồ khi tung cú đấm. Một độ đo của cú đấm thẳng mạnh hoàn hảo là vai của cánh tay đánh, đầu gối của chân trước và đầu gối của chân trước nằm trên cùng một mặt phẳng dọc.

Hook - Một cú đấm bán tròn được tung ra với tay đầu tiên vào bên cạnh đầu đối thủ. Từ tư thế bảo vệ, khuỷu tay được kéo lại với nắm đấm ngang (tránh hướng lên) tuy nhiên trong thời đại hiện đại, một tỷ lệ rộng các võ sĩ tung cú hook với nắm đấm dọc (tránh hướng về bên họ). Tay sau được gấp chắc chắn vào hàm để bảo vệ cằm. Thân hình và hông được xoay theo chiều kim đồng hồ, đẩy nắm đấm đi qua một vòng cung chặt theo chiều kim đồng hồ qua phía trước cơ thể và tiếp xúc với mục tiêu.

Uppercut - Một cú đấm thẳng dọc, nâng lên được tung ra với tay sau. Từ tư thế bảo vệ, thân hình dịch chuyển một chút sang phải, tay sau rơi xuống dưới mức ngực đối thủ và đầu gối được uốn cong một chút. Từ tư thế này, tay sau được đẩy lên theo một quỹ đạo nâng lên đến cằm hoặc thân đối thủ.

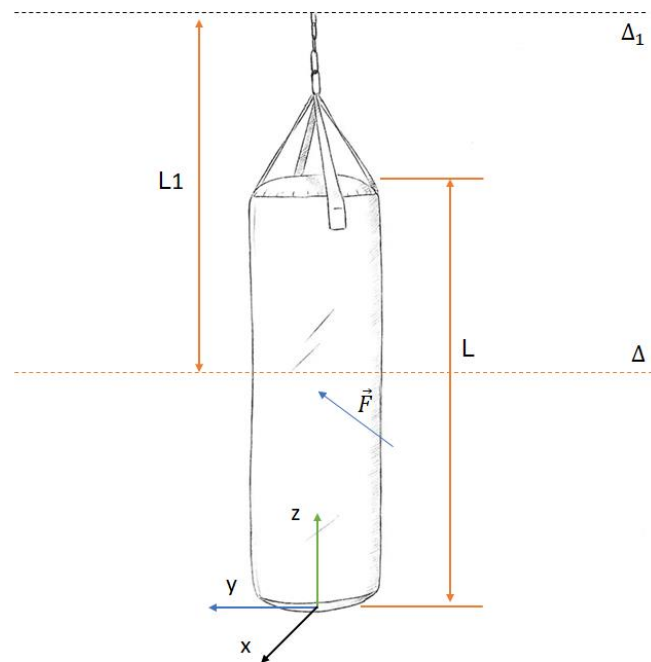
2.1.2 LỊCH SỬ NGHIÊN CỨU LỰC ĐẤM TRONG BOXING

Thống kê lực đấm từ bài báo [1], cho thấy với các vận động viên nam cho lực đấm trung bình với cú đấm cross vào khoảng 3364N, hook là 3115N và một cú jab đạt đến 2697N. Cường độ lực đánh trung bình cho các võ sĩ nam hàng đầu là 4.926 N cho cú đấm thẳng bằng tay trước và 5.248 N cho cú đấm thẳng bằng tay sau. Để đo được lực đấm, trong một nghiên cứu họ không dùng loadcell mà sử dụng một thiết bị tạo lực chuyên biệt để có thể tạo ra một lực đúng chính xác hơn [3]. Ngoài ra còn nhiều nghiên

cứu khác không sử dụng cảm biến gia tốc dài/góc, mà sử dụng camera với SR bằng 120Hz để đo được lực đâm của vận động viên lên bao đấm [4].

2.1.3 PHÂN TÍCH ĐỘNG LỰC HỌC CỦA BAO ĐẤM

Bao đấm được nhóm sử dụng trong đồ án lần này là bao đấm treo được thể hiện ở hình sau:



Hình 2.2 Kích thước và vị trí treo của bao đấm.

Bảng 2.1 Thông tin chi tiết của bao đấm

Viết tắt	Giải thích	Giá trị
$L1 (m)$	Độ dài từ nơi trần nhà đến nửa chiều dài của bao đấm.	1.2
$L (m)$	Chiều dài của bao đấm.	1.6
$M (kg)$	Khối lượng bao đấm.	80
$C (m)$	Chu vi bao đấm.	1.1
x	Trục Ox.	
y	Trục Oy.	
z	Trục Oz.	

Vì bao đầm có hình dạng phức tạp, nên nhóm sử dụng cảm biến để từ đó suy ngược về moment quán tính đối với trục quay.

Xét lực đầm bất kỳ tại của bao đầm, ta có:

$$\vec{F} = \vec{F}_x + \vec{F}_y + \vec{F}_z$$

Xem lực trên trục Oz là không có, nên từ đó phương trình được viết lại thành:

$$\vec{F} = \vec{F}_x + \vec{F}_y$$

Từ đó ta có lực F là cảm biến thu nhận được có giá trị bằng: $F = \sqrt{F_x^2 + F_y^2}$

Từ cảm biến gia tốc dài đã thu được từ cảm biến, ta thực hiện tìm moment quán tính của bao đầm như sau:

Từ a_x và a_y của cảm biến ta suy ra được gia tốc góc của bao đầm:

$$g_{P_x} = \frac{ax}{L}$$

$$g_{P_y} = \frac{ay}{L}$$

Từ giá trị lực trên bao đầm thu được ta tính được moment của lực đầm

$$\tau_F = F \times L1$$

(2.1)

Áp dụng định luật 2 Newton cho chuyển động quay:

$$\tau_F = I \times \beta$$

(2.2)

Với β được tính như sau:

$$\beta = \sqrt{g_{P_x}^2 + g_{P_y}^2}$$

Từ hai phương trình (2.1) và (2.2) ta suy được moment quán tính của bao cát như sau:

$$I = \frac{\tau_F}{\beta}$$

(2.3)

Trong đó:

- I là moment quán tính của bao đám ($kg \times m^2$).
- τ_F là lực moment quán tính. (Nm)
- β là gia tốc góc thu được từ cảm biến. (rad/s^2)

2.1.4 ƯỚC LƯỢNG MOMENT QUÁN TÍNH CỦA BAO ĐÁM DỰA TRÊN HÌNH HỌC

Vì bao đám có dạng gần giống so với hình trụ nên có moment quán tính đối với trục Δ (Hình 2.2) được tính như sau:

$$I_{\Delta} = \frac{1}{4}MR^2 + \frac{1}{12}ML^2$$

Ta tiến hành dời trục Δ về trục Δ_1 :

$$I_{\Delta_1} = I_{\Delta} + M \times L1^2$$

(2.4)

2.2 TỔNG QUAN VỀ MẠNG NƠ-RON TÍCH CHẬP (CNN)

Mạng nơ-ron tích chập (Convolutional Neural Network - CNN) là một loại thuật toán học sâu được sử dụng rộng rãi trong các ứng dụng thị giác máy tính và phân tích hình ảnh. CNN được thiết kế để phát hiện và trích xuất các đặc trưng từ một hình ảnh dựa trên các lớp tích chập. Lớp tích chập áp dụng bộ lọc để quét ảnh dữ liệu đầu vào để trích xuất các đặc trưng như cạnh, góc và hình dạng. Lớp gộp góp phần giảm độ phức tạp của mô hình. Lớp kết nối đầy đủ kết nối đầu ra của các neuron lớp trước đó với các neuron của nó.

2.2.1 LỚP TÍCH CHẬP

Một Convolutional layer trong CNN là khối xây dựng chính của mạng, thực hiện thao tác tích chập trên dữ liệu đầu vào. Thao tác tích chập được thực hiện bằng cách quét các bộ lọc qua toàn bộ ảnh đầu vào để lấy thông tin.

Mỗi bộ lọc trượt qua hình ảnh đầu vào, thực hiện phép nhân từng phần tử giữa trọng số của bộ lọc và giá trị pixel tương ứng trong hình ảnh đầu vào. Kết quả giá trị số được cộng lại để thu được một giá trị đầu ra duy nhất cho vị trí đó. Quá trình này được lặp lại cho tất cả các vị trí trong hình ảnh đầu vào, tạo ra một bản đồ đặc trưng ở các vị trí nổi bật cụ thể, chẳng hạn như cạnh, góc hoặc hình dạng.

Các bộ lọc trong lớp tích chập được học trong quá trình đào tạo của mạng, thông qua thuật toán lan truyền ngược. Bằng cách xếp nhiều lớp tích chập lên nhau, mạng có thể học các đặc trưng phức tạp và trừu tượng hơn từ dữ liệu đầu vào, cho phép nó thực hiện các tác vụ phân tích hình ảnh tinh vi như phát hiện hoặc nhận dạng đối tượng.

2.2.2 LỚP GỘP (MAX POOLING)

Lớp Pooling trong CNN là một loại lớp thường được đặt sau một hoặc nhiều lớp convolutional. Mục đích của nó là giảm chiều dữ liệu đầu vào trong khi giữ lại các đặc trưng quan trọng nhất.

Phép toán pooling có thể là max pooling, average pooling hoặc các loại pooling khác. Max pooling chọn giá trị lớn nhất trong mỗi vùng, trong khi average pooling tính giá trị trung bình. Kết quả của phép toán pooling là một feature map nhỏ hơn, nổi bật các đặc trưng quan trọng nhất của dữ liệu đầu vào.

Lợi ích chính của pooling là hai mặt. Thứ nhất, nó giảm chiều không gian của đầu vào, dẫn đến số lượng tham số ít hơn trong các lớp tiếp theo và giảm rủi ro quá khớp. Thứ hai, nó cung cấp tính dịch chuyển không đổi, có nghĩa là các dịch chuyển nhỏ của đầu vào sẽ không ảnh hưởng đến đầu ra của lớp pooling. Điều này có thể hữu ích cho các nhiệm vụ như phát hiện hoặc nhận dạng đối tượng, nơi vị trí của đối tượng trong ảnh đầu vào có thể thay đổi.

2.2.3 HÀM KÍCH HOẠT (ACTIVATION FUNCTION)

Hàm kích hoạt là một hàm được áp dụng vào đầu ra của mỗi neuron. Hàm kích hoạt giới thiệu tính phi tuyến vào mạng, cho phép nó học được các đặc trưng phức tạp và trừu tượng hơn từ dữ liệu đầu vào.

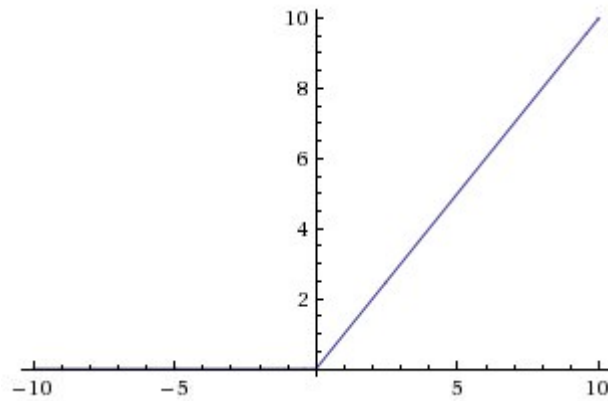
Hàm kích hoạt phổ biến nhất trong CNN là hàm đơn vị tuyến tính (Rectified Linear Unit - ReLU), trả về giá trị lớn nhất giữa đầu vào và 0. ReLU tính toán hiệu quả và được chứng minh cải thiện hiệu suất của CNN so với các hàm kích hoạt khác. Các hàm kích hoạt khác được sử dụng trong CNN bao gồm hàm tang hyperbol (tanh), sigmoid và softmax.

Lựa chọn hàm kích hoạt có thể ảnh hưởng đáng kể đến hiệu suất của CNN và phụ thuộc vào tính chất của tác vụ và đặc điểm của dữ liệu. Hàm ReLU thường là một điểm khởi đầu tốt cho hầu hết các tác vụ, nhưng nó có thể không phù hợp với tất cả các kịch bản.

2.2.4 HÀM KÍCH HOẠT RELU

Hàm kích hoạt ReLU (Rectified Linear Unit) là một hàm kích hoạt thường được sử dụng trong Mạng Nơ-ron Tích chập (Convolutional Neural Networks - CNNs). Nó nhận giá trị đầu vào và trả về giá trị lớn nhất giữa giá trị đầu vào và số không. Nói cách khác, nếu giá trị đầu vào lớn hơn không, đầu ra bằng giá trị đầu vào. Nếu giá trị đầu vào nhỏ hơn hoặc bằng không, đầu ra bằng số không.

ReLU là hàm kích hoạt được sử dụng phổ biến nhất trong mô hình học sâu. Hàm này trả về giá trị 0 nếu nhận được đầu vào âm, và trả lại giá trị dương nếu nhận được đầu vào dương. Do đó, nó có thể được viết thành $f(x) = \max(0, x)$.



Hình 2.3 Đồ thị hàm kích hoạt ReLU [5].

Hàm kích hoạt ReLU đơn giản, tính toán hiệu quả và đã được chứng minh là cải thiện hiệu suất của CNN so với các hàm kích hoạt khác. Nó giới thiệu tính phi tuyến vào mạng, cho phép nó học được các đặc trưng phức tạp và trừu tượng hơn từ dữ liệu đầu vào. Ngoài ra, ReLU có tính thưa thớt (sparsity property), có nghĩa là nhiều nơ-ron trong mạng không hoạt động, dẫn đến sử dụng tài nguyên mạng hiệu quả hơn.

Mặc dù có nhiều lợi ích, hàm kích hoạt ReLU có thể gặp vấn đề "dying ReLU", trong đó một số nơ-ron trở thành vĩnh viễn không hoạt động và ngừng học trong quá trình đào tạo. Để giải quyết vấn đề này, các biến thể của ReLU, chẳng hạn như leaky ReLU, đã được đề xuất, giới thiệu một độ dốc nhỏ cho đầu vào âm để ngăn chặn vấn đề dying ReLU.

2.2.5 HÀM KÍCH HOẠT SOFTMAX

Trong một Mạng Nơ-ron Tích chập (CNN), hàm Softmax là một hàm kích hoạt thường được sử dụng trong lớp đầu ra của mạng cho các tác vụ phân loại đa lớp. Nó nhận đầu vào là một vector số thực và trả về một vector cùng kích thước, trong đó mỗi phần tử của vector đầu ra là một xác suất cho việc đầu vào thuộc về một lớp cụ thể.

Hàm Softmax hoạt động bằng cách lấy lũy thừa của mỗi phần tử trong vector đầu vào và sau đó chuẩn hóa kết quả để thu được một phân bố xác suất. Cụ thể hơn, hàm Softmax áp dụng công thức sau đối với mỗi phần tử x_i của vector đầu vào:

$$\text{softmax}(x_i) = \frac{\exp(x_i)}{\sum (\exp(x_j))}$$

trong đó tổng được lấy trên tất cả các phần tử của vector đầu vào.

Hàm Softmax đảm bảo rằng các xác suất đầu ra cộng lại thành 1, điều này làm cho nó phù hợp với các tác vụ phân loại đa lớp. Trong quá trình huấn luyện, hàm Softmax thường được sử dụng kết hợp với hàm mất mát cross-entropy, đo lường sự khác biệt giữa nhãn lớp dự đoán và thực tế và cung cấp đạo hàm để cập nhật trọng số mạng.

2.2.6 CHUẨN HÓA DỮ LIỆU (NORMALIZATION)

Normalization trong CNN là việc chuẩn hóa dữ liệu, một kỹ thuật được sử dụng để tiền xử lý dữ liệu trước khi đưa vào mạng neural. Chuẩn hóa đề cập đến quá trình tỷ lệ hoá dữ liệu đầu vào để có trung bình bằng không và độ lệch chuẩn bằng một, giúp cho mạng neural học một cách hiệu quả hơn.

Trong CNN, chuẩn hóa thường được áp dụng cho dữ liệu đầu vào cũng như đầu ra của mỗi tầng. Chuẩn hóa có thể được thực hiện bằng nhiều cách khác nhau, bao gồm chuẩn hóa theo đặc trưng, chuẩn hóa theo mẫu hoặc chuẩn hóa theo lô (batch normalization). Đặc biệt, chuẩn hóa theo lô là một kỹ thuật được sử dụng rộng rãi trong CNNs, giúp chuẩn hóa đầu ra của một tầng bằng cách trừ đi trung bình và chia cho độ lệch chuẩn của lô dữ liệu đang được xử lý.

Chuẩn hóa giúp giảm thiểu ảnh hưởng của dữ liệu đầu vào có thể có tỷ lệ khác nhau, dẫn đến việc hội tụ chậm và hiệu suất của mô hình bị giảm. Bằng cách chuẩn hóa dữ liệu đầu vào, phạm vi giá trị đầu vào được chuẩn hóa, cho phép mạng neural học nhanh hơn và chính xác hơn.

CHƯƠNG 3 THIẾT BỊ PHẦN CỨNG VÀ PHẦN MỀM

3.1 THIẾT BỊ PHẦN CỨNG

Trong đề tài lần này, nhóm cần sử dụng các chức năng như sau của vi điều khiển:

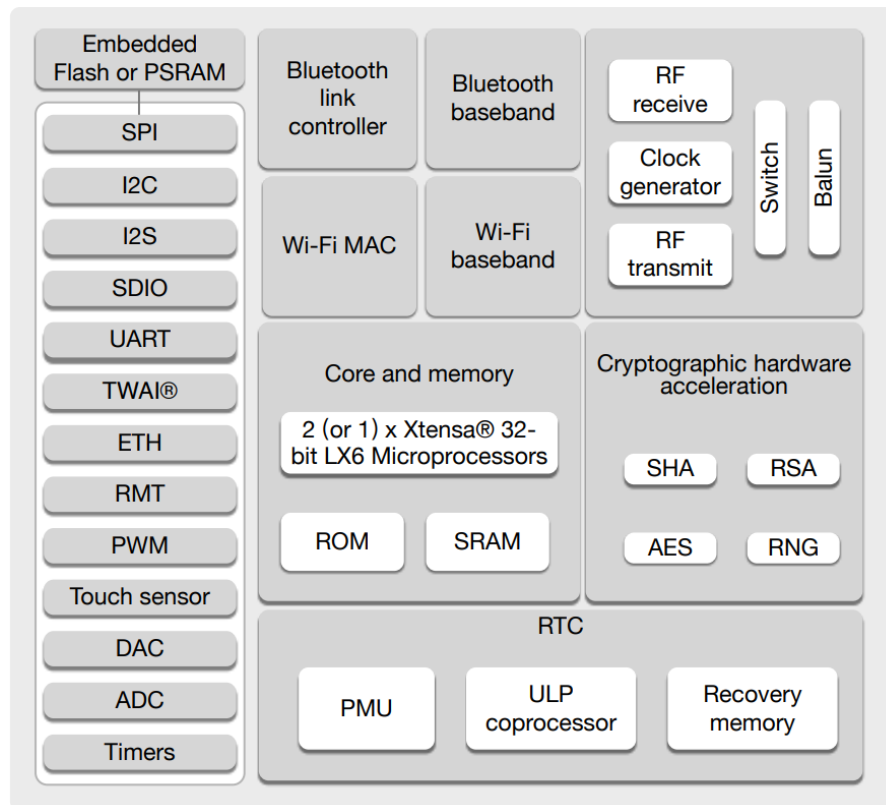
- Bluetooth.
- Analog to digital converter (ADC).
- Thư viện cung cấp chuẩn giao tiếp I2C hoặc SPI.

Tầm đo gia tốc góc vào khoảng $\pm 32g$ và gia tốc dài vào khoảng $\pm 2000dps$. Từ đó, nhóm sử dụng cảm biến gia tốc dài/góc LSM6DSO32. Trong cảm biến này đáp ứng các nhu cầu như sau [6]:

- Năng lượng tiêu thụ thấp: 0.55mA khi sử dụng chế độ high-performance .
- Có bộ hàng đợi (First In First Out) lên đến 9kbytes.
- Tầm đo của gia tốc dài $\pm 4/\pm 8/\pm 16/\pm 32$ g.
- Tầm đo của gia tốc góc $\pm 125/\pm 250/\pm 500/\pm 1000/\pm 2000dps$.
- Nguồn điện thế từ 1.71 V đến 3.6 V.

Quan trọng nhất, nhóm đã sử dụng qua các loại bộ chuyển đổi để thu được lực, từ tám RP-S40-ST đến loadcell, theo thống kê từ bài báo thì lực đầm trung bình của một vận động viên giao động khác nhau so với từng hạng cân [7]. Tuy nhiên nhóm sử dụng loại loadcell 20kg dễ tìm, dễ sử dụng có ở thị trường Việt Nam cho nghiên cứu lần này, vì cấu tạo của các loadcell như các bộ chuyển đổi (transducer) nên việc sử dụng loadcell 20kg sẽ tương tự với sử dụng các loại loadcell khác.

3.1.1 VI ĐIỀU KHIỂN ESP32

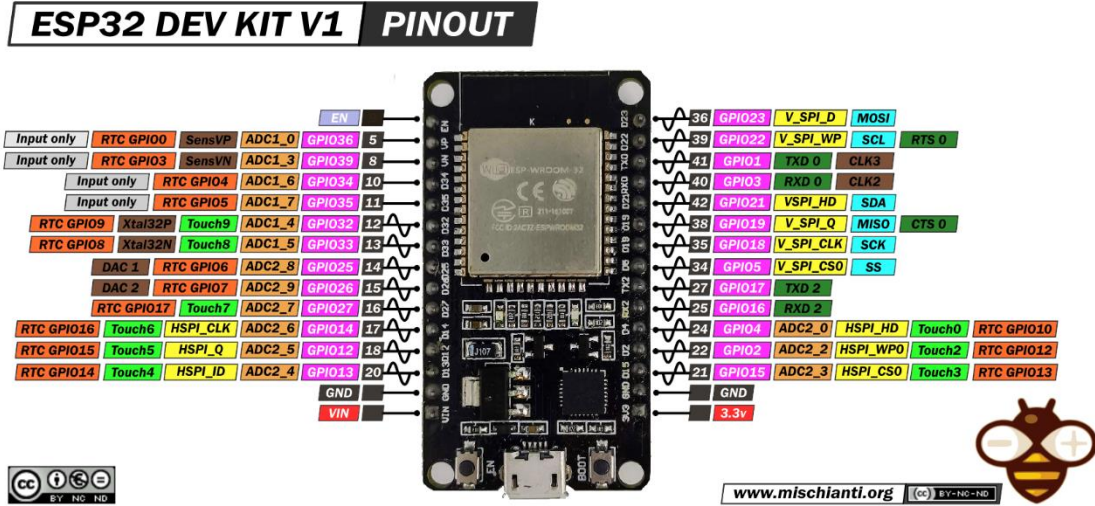


Hình 3.1 Các khối chức năng cơ bản của ESP32 [8].

Tổng quan về vi điều khiển ESP32:

- Vi xử lý:
 - CPU: Xtensa dual-core (or single-core) 32-bit LX6.
 - Vi xử lý tiết kiệm điện năng (Ultra low power).
- Bộ nhớ: 320KiB RAM, 448 KiB ROM.
- Kết nối không dây:
 - Wi-Fi: 802.11 b/g/n.
 - Bluetooth: v4.2 BR/EDR và BLE (chia sẻ sóng radio với Wi-Fi).
- Ngoại vi:
 - 32 chân GPOs.
 - 12-bit SAR ADS 18 kênh.
 - 2 x 8-bit DACs.
 - 4 x SPI.

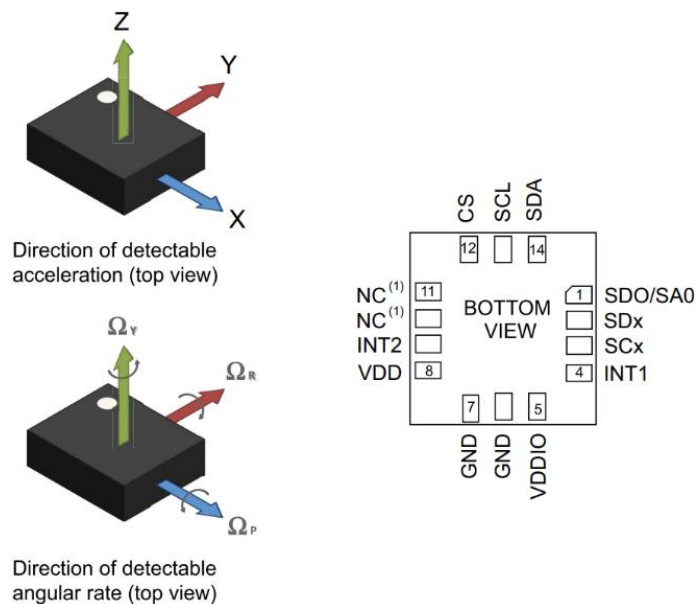
- 2 x I2S.
- 2 x I2C.
- 3 x UART.



Hình 3.2 Sơ đồ chân vào ra của mạch Doit Devkit V1 [9].

3.1.2 CẢM BIẾN GIA TỐC LSM6DSO32

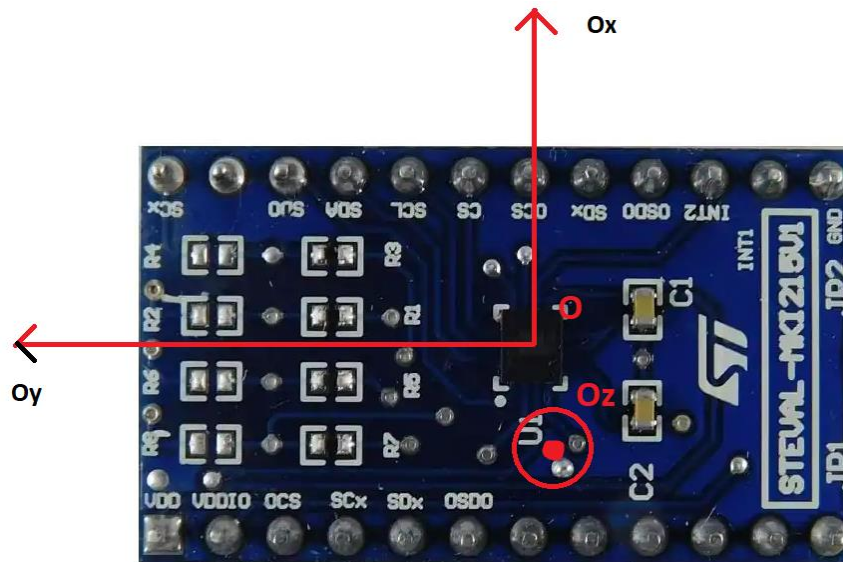
Cảm biến LSM6DSO32 rất nhỏ gọn với kích thước: $2.5\text{mm} \times 3\text{mm} \times 0.83\text{mm}$. Cảm biến có hỗ trợ chuẩn giao tiếp I2C và SPI.



Hình 3.3 Các chân ngoại vi và trục của cảm biến [6].

Bảng 3.1 Các chân của cảm biến lsm6dso32 được sử dụng

Kí hiệu	Mô tả (trong chuẩn giao thức SPI [6])
<i>SCL</i>	Được kết nối với chân <i>CSK</i>
<i>SDA</i>	Được kết nối với chân <i>MOSI</i>
<i>SDO</i>	Được kết nối với chân <i>MISO</i>
<i>CS</i>	Được kết nối với chân <i>CS</i>
<i>Vcc</i>	Chân nguồn
<i>GND</i>	Chân đất



Hình 3.4 Các trục Ox , Oy , Oz của cảm biến gia tốc dài/góc LSM6DSO32.

3.1.2.1 THANH GHI WHO_AM_I (0Fh)

WHO_AM_I là một giá trị quan trọng của cảm biến, nó trả về giá trị định danh mà trong quá trình sử dụng, vi điều khiển có thể phân biệt với các cảm biến khác.

Bảng 3.2 Thanh ghi WHO_AM_I.

0	1	1	0	1	1	0	0
---	---	---	---	---	---	---	---

3.1.2.2 THANH GHI CTRL1_XL (10h) và CTRL2_G

Sử dụng 2 thanh ghi này để kích hoạt cảm biến sử dụng hoặc gia tốc dài hoặc gia tốc góc.

Bảng 3.3 Thanh ghi CTRL1_XL.

ODL_XL3	ODR_XL2	ODR_XL1	ODR_XL0	FS1_XL	FS0_XL	LPF2_XL_EN	0
---------	---------	---------	---------	--------	--------	------------	---

Bảng 3.4 Thanh ghi CTRL2_G.

ODR_G3	ODR_G2	ODR_G1	ODR_G0	FS1_G	FS0_G	FS_125	0
--------	--------	--------	--------	-------	-------	--------	---

3.1.2.3 THANH GHI CTRL6_C (15h) VÀ CTRL7_C (16h)

Các chế độ power sẽ ảnh hưởng đến tần số lấy mẫu của cảm biến. Các thanh ghi này cho phép ta chỉnh được tần số lấy mẫu.

Bảng 3.5 Thanh ghi CTRL6_C chỉnh tần số lấy mẫu của gia tốc dài.

TRIG_EN	LVL1_EN	LVL2_EN	XL_HM_MODE	USR_HM_MODE	FTYPE_2	FTYPE_1	FTYPE_0
---------	---------	---------	------------	-------------	---------	---------	---------

Bảng 3.6 Thanh ghi CTRL7_C chỉnh tần số lấy mẫu của gia tốc góc.

G_HM_MODE	HP_EN_G	HPM1_G	HPM0_G	0	0	USR_OFF_ON_OUT	0
-----------	---------	--------	--------	---	---	----------------	---

3.1.2.4 BỘ THANH GHI GIA TỐC DÀI VÀ GIA TỐC GÓC

Quan trọng nhất là bộ 12 thanh ghi để ghi nhận giá trị gia tốc dài và gia tốc góc.

Bảng 3.7 Địa chỉ 12 thanh ghi gia tốc dài và gia tốc góc.

Tên	Loại	Địa chỉ thanh ghi		Chế độ mặc định
		Hex	Binary	
OUTX_L_G	R	22	00100010	Output
OUTX_H_G	R	23	00100011	Output
OUTY_L_G	R	24	00100100	Output
OUTY_H_G	R	25	00100101	Output
OUTZ_L_G	R	26	00100110	Output
OUTZ_H_G	R	27	00100111	Output

Tên	Loại	Địa chỉ thanh ghi		Chế độ mặc định
		Hex	Binary	
OUTX_L_A	R	28	00101000	Output
OUTX_H_A	R	29	00101001	Output
OUTY_L_A	R	2A	00101010	Output
OUTY_H_A	R	2B	00101011	Output
OUTZ_L_A	R	2C	00101100	Output
OUTZ_H_A	R	2D	00101101	Output

3.1.3 CẢM BIẾN LỰC DẠNG LOADCELL

Bộ chuyển đổi sử dụng LOADCELL tạo thành mạch cầu Strain Gauge, từ đó có sự thay đổi nhỏ về điện trở cho ta thay đổi về điện thế. Loadcell được sử dụng là loại 20kg có độ nhạy 20mV/V.



Hình 3.5 Loadcell 20kg [10].

Bảng 3.8 Các thông số cơ bản của loadcell [8].

Đo lực	Nén
Thang đo	20kg
Điện áp hoạt động	5 Vdc
Ngõ ra	2mV/V

Vật liệu vỏ	Nhôm
Loại điện cực	Nối dây
Chiều cao	12.7mm
Chiều rộng	80mm
Chiều sâu	12.7mm
Nhiệt độ min	-20°C
Nhiệt độ max	+65°C
Dây đỏ	Ngõ vào (+)
Dây đen	Ngõ vào (-)
Dây xanh lá	Ngõ ra (+)
Dây trắng	Ngõ ra (-)

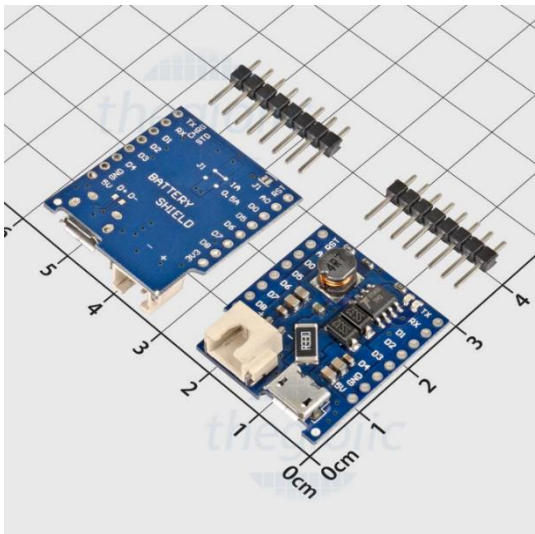
Điện thế output+, output- của loadcell không thể đọc được bằng ADC của ESP32. Nên từ đó nhóm đã sử dụng mạch khuếch đại (được nói đến trong phần 3.1.6) để có thể lấy được tín hiệu thô từ loadcell. Từ tín hiệu thô này nhóm bắt đầu quá trình căn chuẩn thiết bị (calibration).



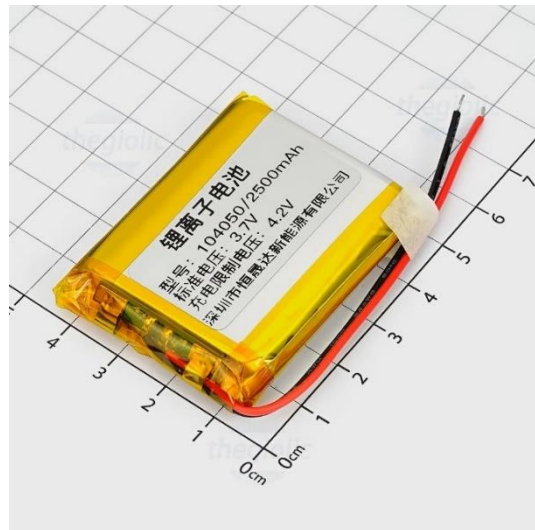
Hình 3.6 Quy trình tìm đường đặc tính của loadcell.

Quá trình tìm đường đặc tính của thiết bị loadcell như sau, đầu tiên ta đặt vật nặng với khối lượng biết trước lên loadcell, xét khối lượng của loadcell bằng một hằng số cho trước, từ đó ta có thể trừ đi giá trị cân nặng của loadcell để ra giá trị mà cân hiển thị. Tiếp đến ghi nhận giá trị ADC (analog) mà VDK ESP23 trả về. Vòng lặp tiếp tục như vậy đối với các khối lượng khác nhau.

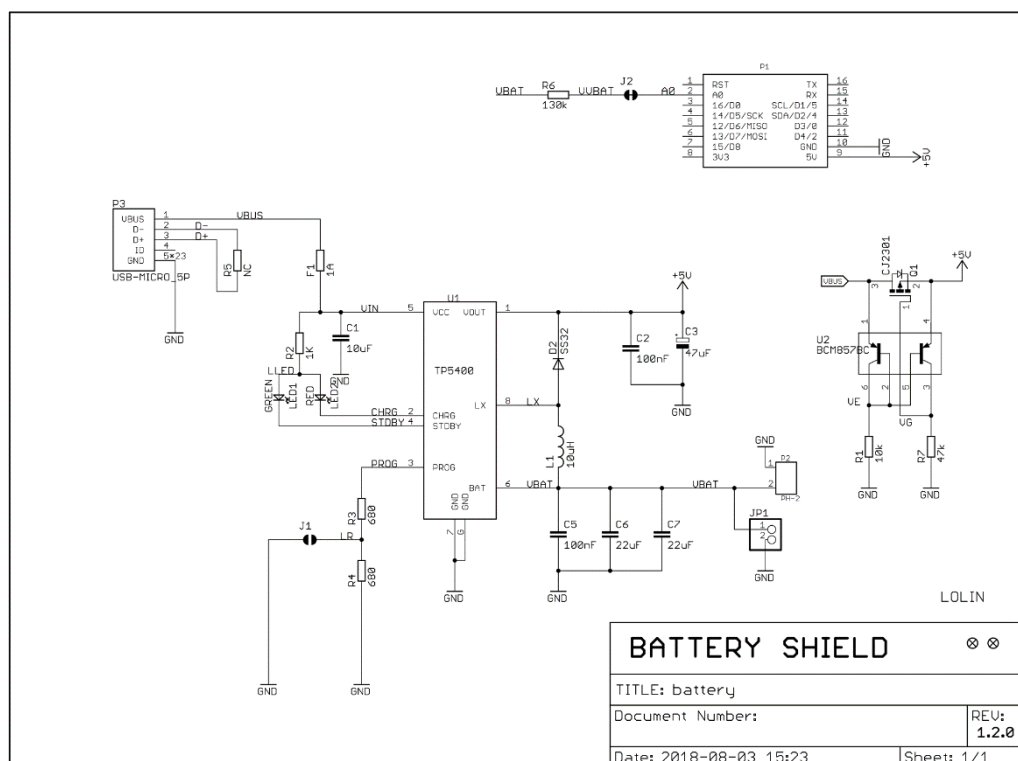
3.1.4 MẠCH NGUỒN CHO ESP32 VÀ SẠC PIN LIPO



Hình 3.7 Mạch nguồn cho vi điều khiển (VDK) ESP32.



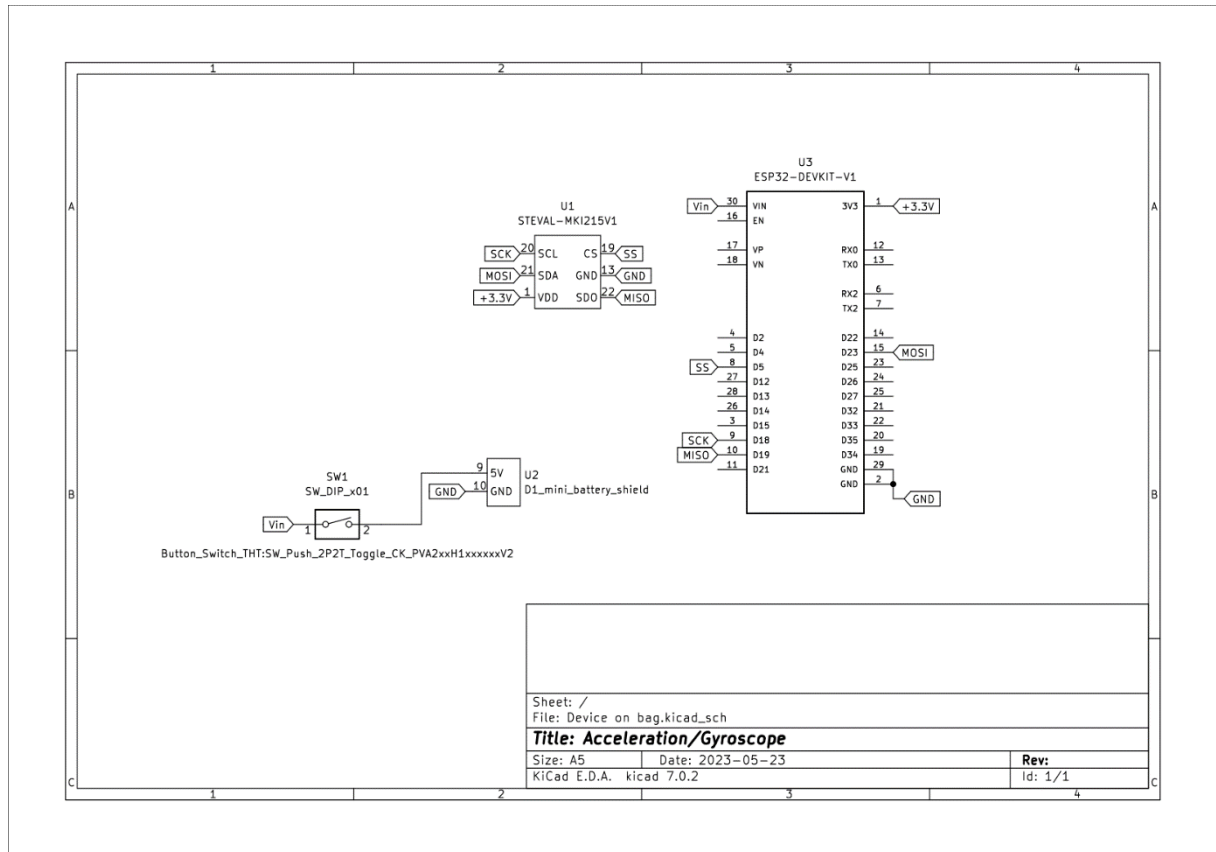
Hình 3.8 Pin LIPO để cấp nguồn cho VDK ESP32.



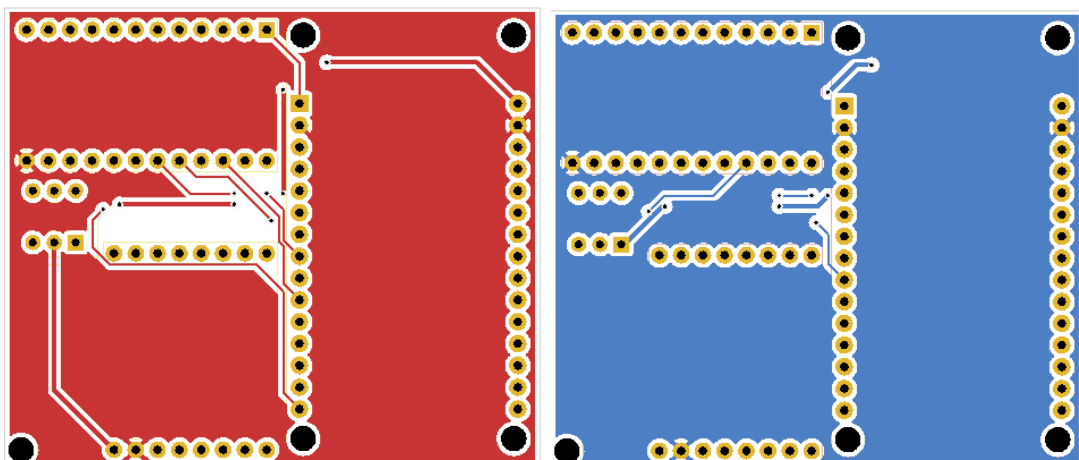
Hình 3.9 Sơ đồ nguyên lý mạch nguồn cho VDK ESP32 [11].

3.1.5 THIẾT KẾ CHI TIẾT THIẾT BỊ

Thiết bị đo gia tốc ở trên bao gồm là sự kết hợp của vi điều khiển ESP32 để đọc tín hiệu từ cảm biến gia tốc dài và gia tốc góc từ cảm biến LSM6DSO32, ngoài ra để cho các thiết bị này hoạt động thì cần đến mạch nguồn và pin LIPO.



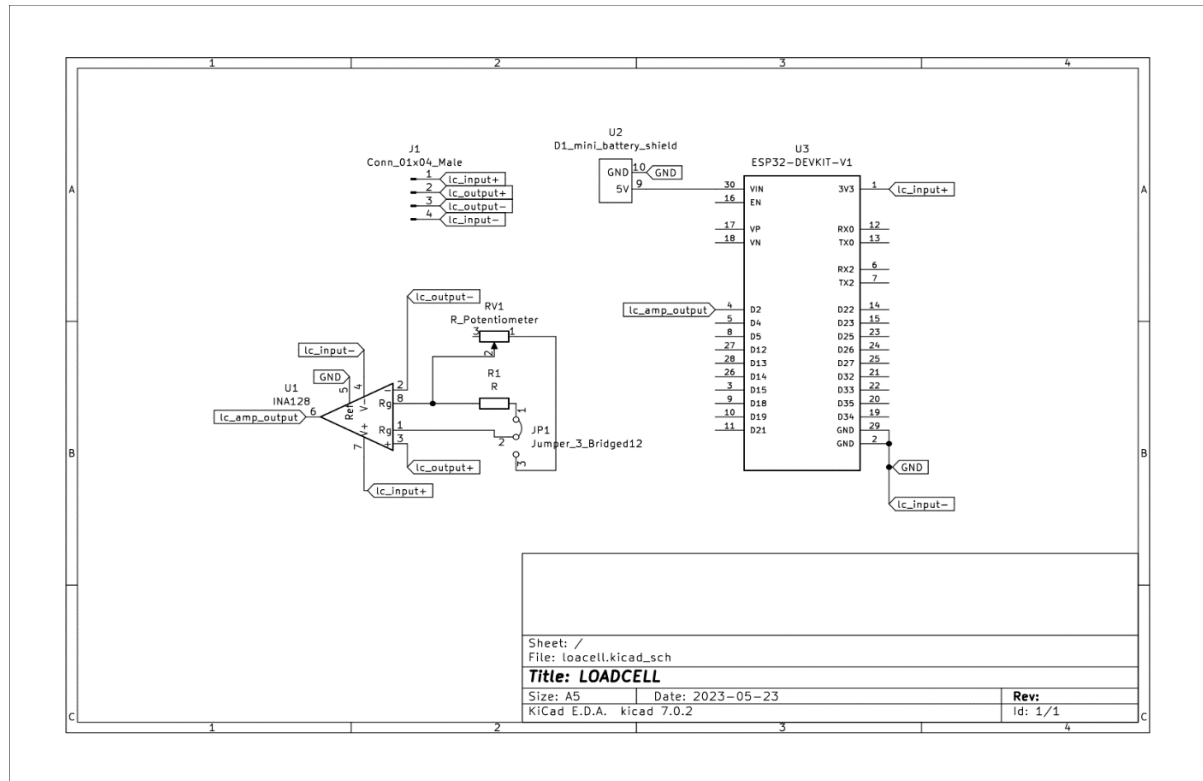
Hình 3.10 Sơ đồ mạch nguyên lý của thiết bị.



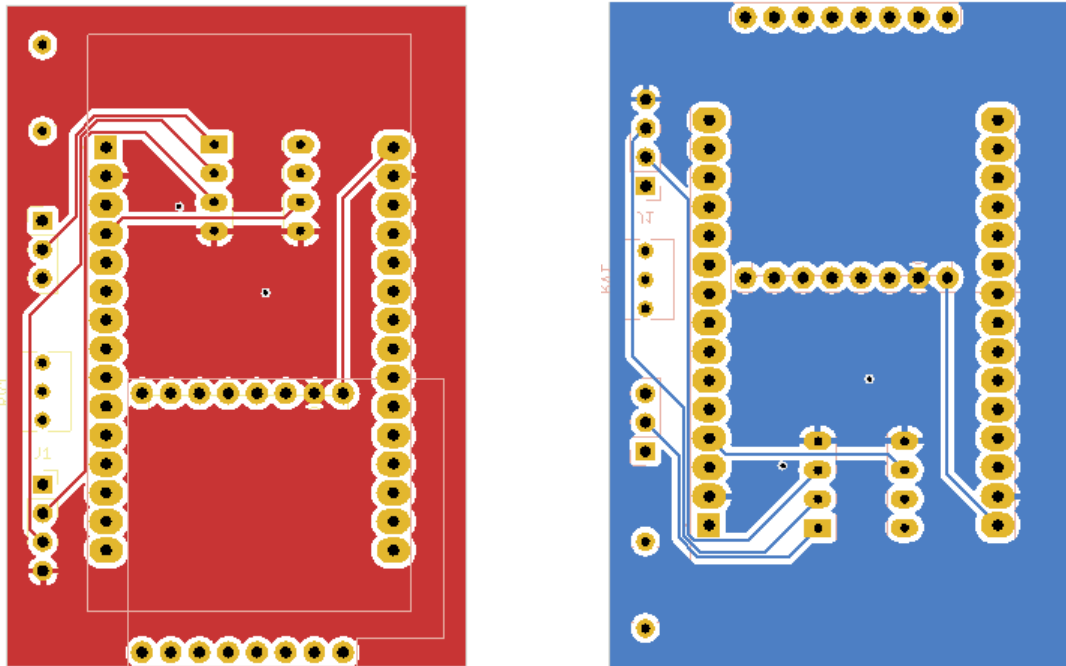
Hình 3.11 PCB Mặt trước và mặt sau của thiết bị.

3.1.6 THIẾT BỊ ĐO LỰC THAM CHIẾU

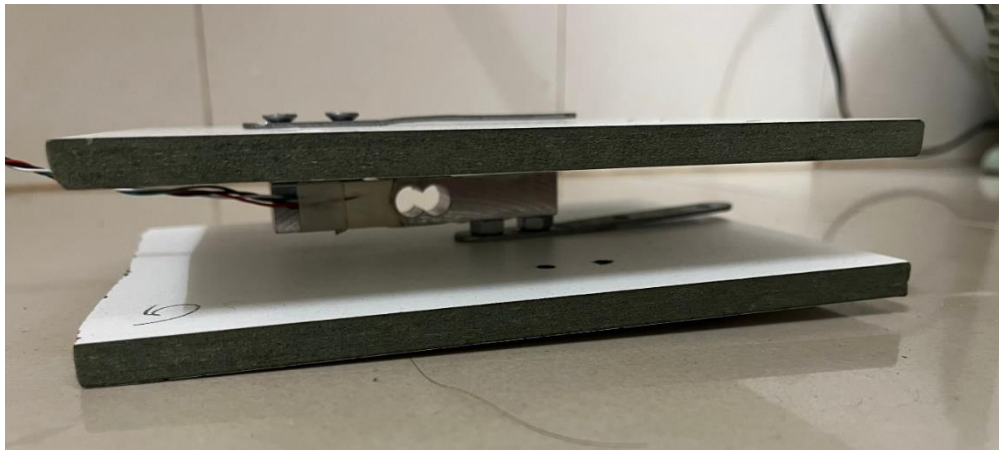
Vì loadcell cho ra độ thay đổi điện thế rất nhỏ mà các dòng vi xử lý khó có thể đo được. Nên tín hiệu của loadcell được khuếch đại khi đi qua INA128 [12]. Sau đó, tín hiệu đã được khuếch đại lên nhiều lần này được cho vào 1 chân của vi điều khiển ESP32 đọc và thu tín hiệu.



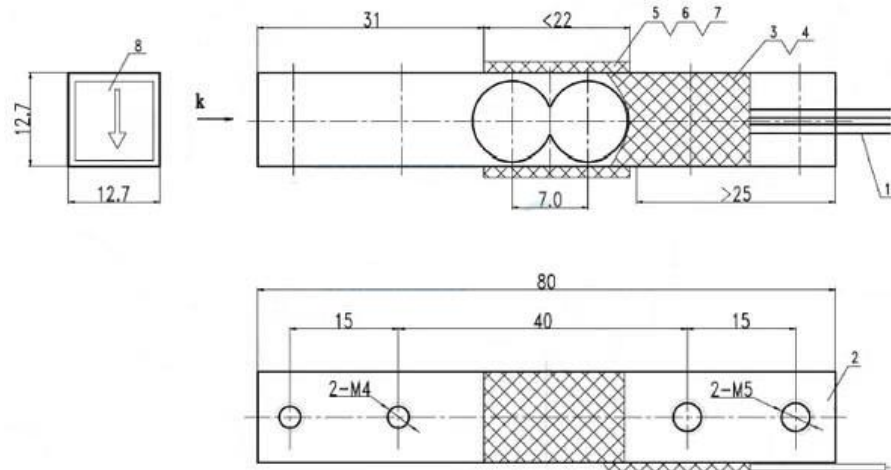
Hình 3.12 Mạch thiết bị đo lực tham chiếu.



Hình 3.13 PCB Mặt trước và mặt sau của thiết bị loadcell tham chiếu.



Hình 3.14 Loadcell và tấm phẳng chịu lực có kích thước dài – rộng – cao lần lượt là $17 \times 14 \times 0.8$ (cm).



Hình 3.15 Bảng vẽ kích thước của loadcell [10].

3.2 PHẦN MỀM (SOFTWARE)

3.2.1 MÃ NGUỒN CỦA VI ĐIỀU KHIỂN

Arduino IDE được dùng để soạn thảo mã nguồn cho vi điều khiển ESP32.

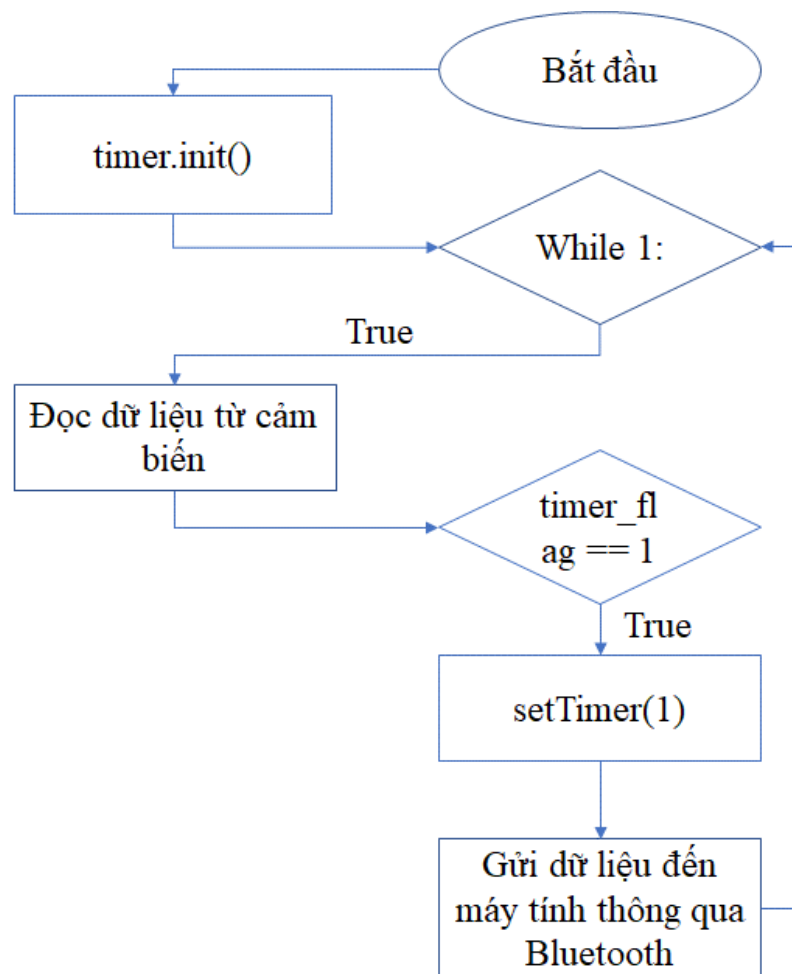


Hình 3.16 Phần mềm Arduino IDE [13].

Bộ vi điều khiển của ESP32 được trang bị tới 3 timer [8], với bộ timer0 đã được tích hợp vào bộ thư viện của Arduino để làm những tác vụ cơ bản (ADC, đọc/gửi qua cổng Serial,...). Để giảm tải nhẹ cho lượng timer0 sử dụng, nhóm chọn timer1 với 16-bit prescaler để hiện thực thời gian gửi dữ liệu qua Bluetooth.

Ban đầu, khi khởi tạo timer, nhóm cho timer clock tick mỗi 1ms, nghĩa là khi timer tick 1000 lần nghĩa là 1 giây. Sau mỗi lần tick, nó sẽ trừ vào một biến là timer_counter, timer_counter sẽ được set thông qua hàm setTimer(). Đến khi timer_counter nhỏ hơn 0, thì lúc đó (hết 1 giây) thì timer_flag sẽ được set bằng 1.

Trong tiến trình chính của vi điều khiển sẽ thấy timer_flag đúng (bằng 1) sẽ gửi dữ liệu đã thu trước đó đến cảm biến. Nhờ vậy nhóm đạt được tần số lấy mẫu mong muốn.



Hình 3.17 Sơ đồ khối thu dữ liệu từ cảm biến.

Bắt đầu quy trình, nhóm sẽ chỉnh timer1 trong vi điều khiển để xác định thời gian lấy mẫu, sau đó chương trình sẽ tiếp tục chạy vòng lặp vô tận, bắt đầu từ việc lấy giá trị từ cảm biến, sau đó xem đến thời gian gửi mẫu chưa, nếu $timer_flag = 1$ thì lúc này vi điều khiển sẽ gửi giá trị đã đọc được từ cảm biến đến với máy tính thông qua Bluetooth.

3.2.1.1 ĐỌC DỮ LIỆU TỪ CẢM BIẾN LSM6DSO32

Đối với cảm biến ESP32 được cung cấp 12 thanh ghi (read-only). Sau khi chỉnh các công suất nguồn (power), tần số lấy mẫu (SR – sampling rate), tầm đo (range) thì tiến hành đọc dữ liệu từ thanh ghi.

Đối với chuẩn giao tiếp SPI thì ta cần đầu tiên đọc thanh ghi ở mã hex 22 (tương ứng với bit LOW ở giá trị gx). Tạo một biến với kiểu dữ liệu là int_16 (2 bytes – 16bit) để lưu giá trị LOW của gx. Sau đó sử dụng phép dịch bit (shift bit operator <<) qua 8 bit để tiếp tục lưu giá trị ở bit HIGH của gx bằng phép OR. Tiếp tục như vậy cho tới khi kết thúc đọc ở bit az cuối cùng.

Tuy nhiên đây chỉ là giá trị raw data, để có thể biến từ LOW – HIGH các bit đã lưu vào một biến, ta cần giá trị trên 1 LSB tương ứng, từ đó suy ra được giá trị gia tốc dài (m/s²) hoặc gia tốc góc (rad/s²) cần tìm.

Bảng 3.9 Tầm tầm đo sẽ có giá trị mg/LSB hoặc mdps/LSB khác nhau [6].

Kí hiệu	Giải thích	Điều kiện chạy	Typ.	Unit
LA_So	Linear acceleration sensitivity	$FS = \pm 4g$	0.122	mg/LSB
		$FS = \pm 8g$	0.244	
		$FS = \pm 16g$	0.488	
		$FS = \pm 32g$	0.976	
G_So	Angular rate sensitivity	$FS = \pm 125dps$	4.375	mdps/LSB
		$FS = \pm 250dps$	8.75	
		$FS = \pm 500dps$	17.50	
		$FS = \pm 1000dps$	35	
		$FS = \pm 2000dps$	70	

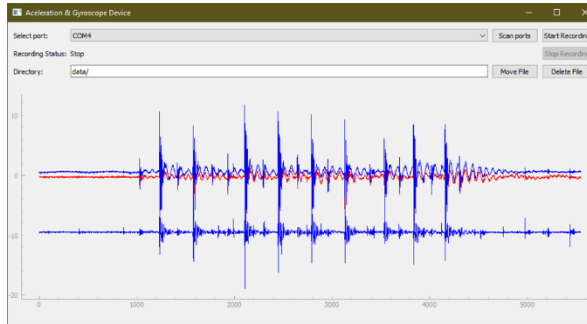
3.2.1.2 ĐỌC DỮ LIỆU TỪ LOADCELL

Vì đọc giá trị loadcell chỉ đơn thuần là độ thay đổi điện thế trong khoảng 0 – 3.3V vì đã được khuếch đại thông qua INA128 [12].

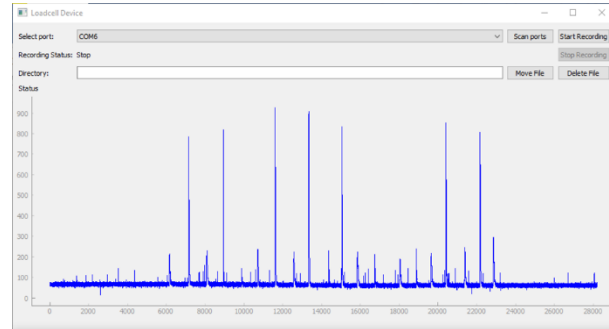
Từ lệnh `analogRead(LOADCELL_PIN)` mà thư viện Arduino đã cung cấp, ta không cần phải viết lại hàm đọc giá trị analog nữa.

3.2.2 GIAO DIỆN THU DỮ LIỆU

Để thu được tín hiệu từ các thiết bị đo gia tốc và đo lực, nhóm đã viết lên phần mềm cơ bản sau.

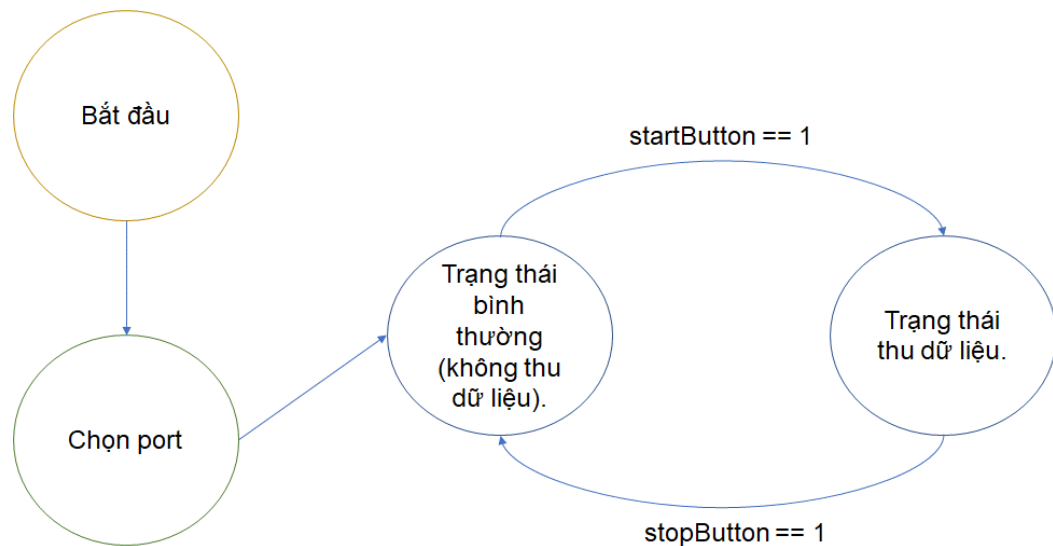


Hình 3.18 Giao diện thu dữ liệu gia tốc dài và gia tốc góc trên bao đệm.

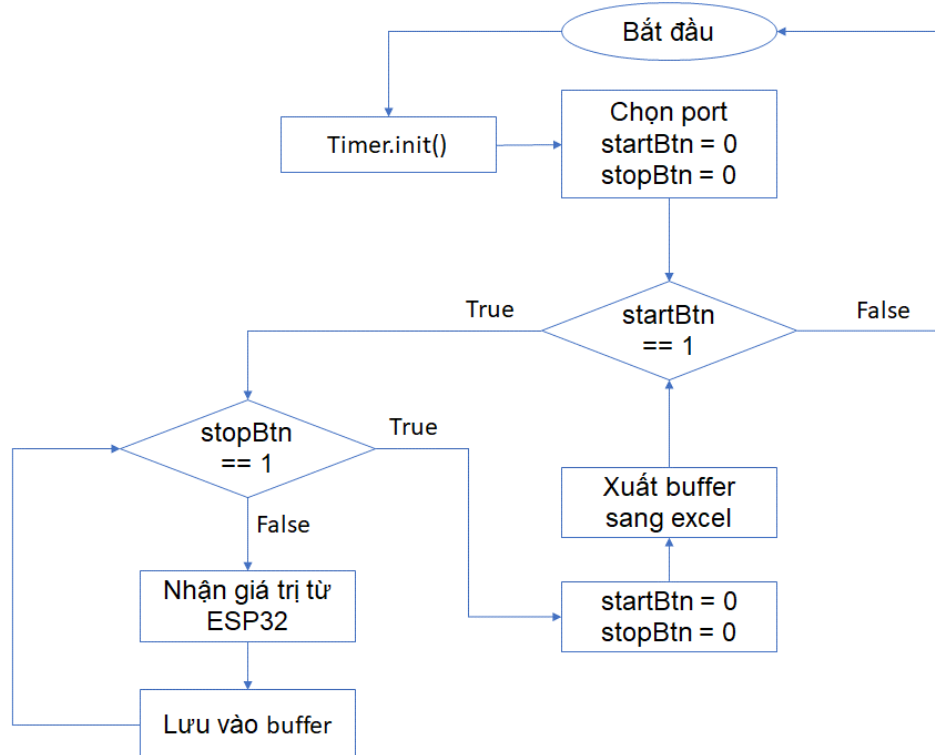


Hình 3.19 Giao diện thu dữ liệu lực tham chiếu bằng loadcell.

Do 2 phần mềm đều làm một công việc giống nhau, đó là thu tín hiệu từ ESP32 gửi đến, nên có cùng các trạng thái giống nhau.



Hình 3.20 Mô hình máy trạng thái (finite state machine) của phần mềm thu dữ liệu gia tốc và lực.

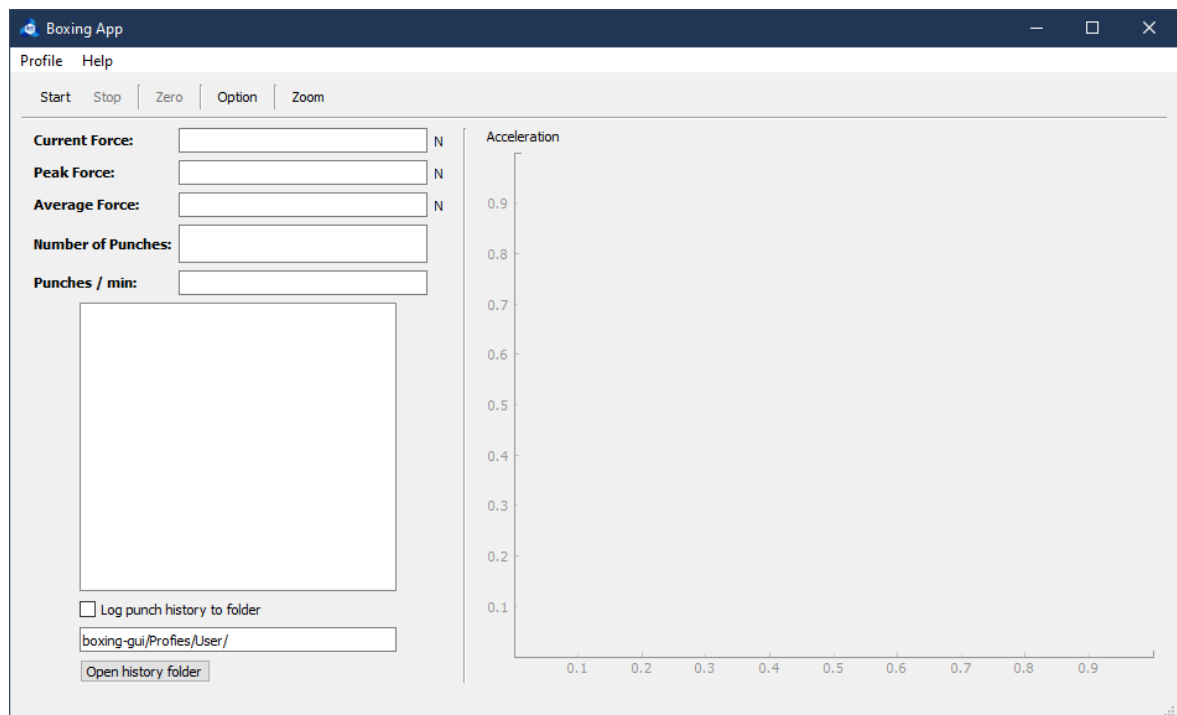


Hình 3.21 Sơ đồ khối thu dữ liệu gia tốc và lực.

Ở trạng thái đầu tiên, vi điều khiển sẽ khởi động và cài đặt timer được người sử dụng định nghĩa trước trong hàm *timer.init()*. Sau đó chương trình sẽ yêu cầu phải nhập port của thiết bị Bluetooth cần sử dụng và lắng nghe ở *startBtn* là có được nhấn chưa.

Nếu nhấn thì bắt đầu timer và chạy như hàm *while(1)*, và trong quá trình thu và gửi dữ liệu thông qua cổng Bluetooth thì nó vẫn sẽ lắng nghe người dùng có ấn dừng không. Nếu có nó sẽ ngắt timer và quay về trạng thái bình thường (chờ lắng nghe từ *startBtn*).

3.2.3 GIAO DIỆN NGƯỜI DÙNG



Hình 3.22 Giao diện chính.

Các chức năng của giao diện chính gồm có:

- Truy cập vào Profile người dùng
- Chạy thời gian thực.
- Tùy chọn chế độ của cảm biến (port, baudrate,...).
- Lưu lại lịch sử luyện tập bằng cách tick vào ô “Log punch history to folder”.
- Trong lúc luyện tập sẽ hiện số cú đấm, cú đấm được tung ra có lực cao nhất (peak force), các cú đấm hiện tại (current force).
- Lực đấm trung bình.
- Bên phải là biểu đồ hiển thị gia tốc thời gian thực.

Khi chọn chế độ Profile, người dùng cần nhập thông tin để từ đó phần mềm có thể căn chuẩn phù hợp với mô hình lực.

Các thông số người dùng cần nhập bao gồm:

- Đầy đủ họ tên (fullname).

- Tuổi (age).
- Chiều cao (height).
- Cân nặng (weight).

Ngoài ra người dùng có thể chọn hệ thống đo lường phù hợp (nhưng mặc định là metric):

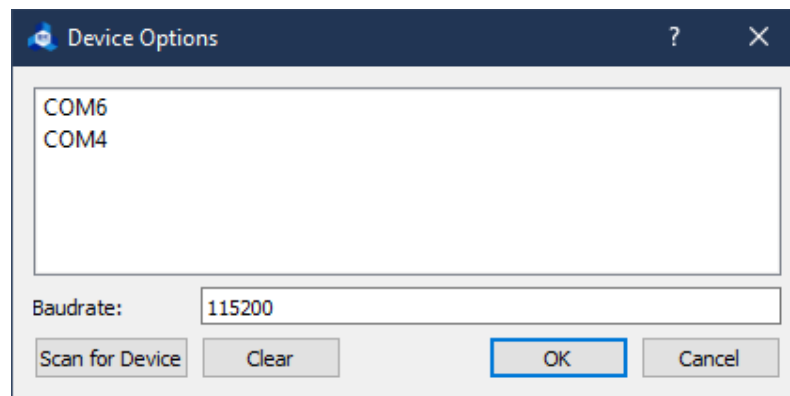
- Metric system.
- Imperial system.

Đối với bao đấm thì có 2 loại:

- Loại treo (Hanging bag).
- Loại đứng (Stangding bag).

Các thông số của bao đấm bao gồm:

- Cân nặng bao đấm.
- Chiều dài bao đấm.
- Khoảng cách treo.

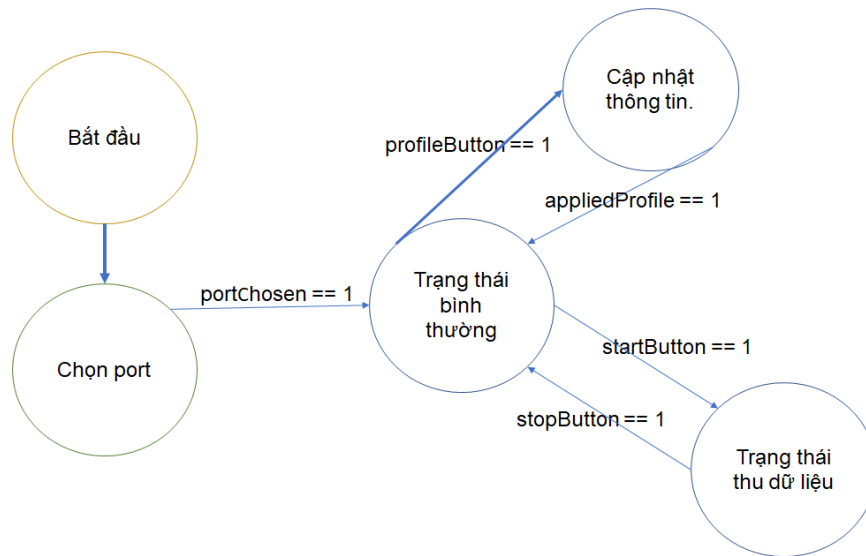


Hình 3.23 Giao diện chọn port để kết nối tới Bluetooth.

Trong giao diện chọn port sẽ có một số nút và cách sử dụng như sau:

- Scan for device là nút quét các thiết bị local trên máy (nghĩa là máy phải kết nối bluetooth với thiết bị trước khi ấn nút này).
- Clear (xóa các thiết bị đã quét).
- Ok chọn port và baudrate.

- Cancel không chọn gì hết.



Hình 3.24 Mô hình máy trạng thái của phần mềm.

Ở trạng thái bắt đầu, chương trình sẽ tự động thực hiện các thao tác ngầm như điều chỉnh timer, load mô hình đã được train,... Sau đó chương trình sẽ chuyển sang trạng thái chọn port, ở trạng thái này, người dùng sẽ liên kết thiết bị đặt ở dưới bao đảm đến tới máy tính bằng Bluetooth. Sau khi chọn liên kết được đến thiết bị đặt ở dưới bao đảm, chương trình sẽ tiếp tục chuyển sang trạng thái bình thường.

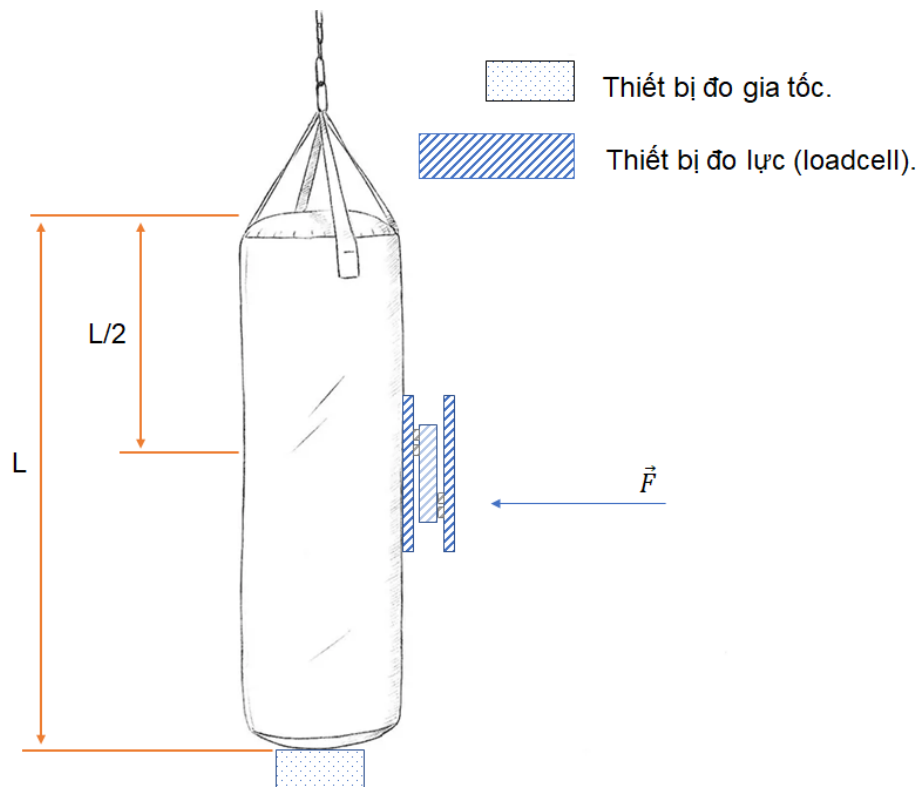
Ở trạng thái bình thường, chương trình sẽ tiếp tục lắng nghe người dùng ấn nút bắt đầu, tức chuyển sang trạng thái vừa thu, vừa xử lý dữ liệu. Ở trạng thái này chương trình sẽ tiếp tục lắng nghe người dùng ở nút ngừng (stopButton) và sau đó kết thúc và quay về trạng thái bình thường.

CHƯƠNG 4 PHƯƠNG PHÁP ĐO VÀ XỬ LÝ TÍN HIỆU

4.1 THU DỮ LIỆU LỰC VÀ GIA TỐC

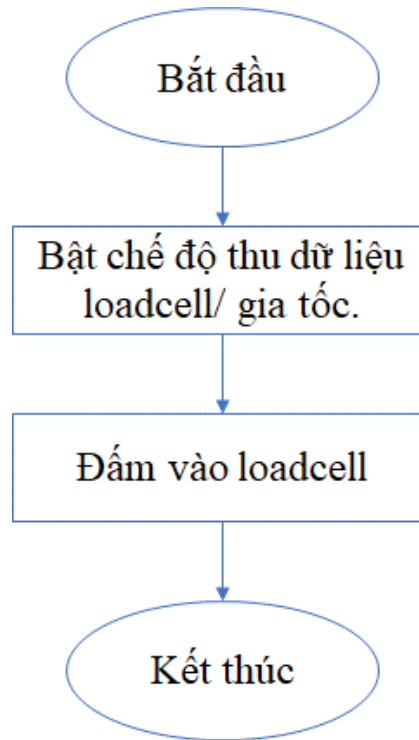
4.1.1 QUY TRÌNH THÍ NGHIỆM

Các thiết bị được đặt như Hình 4.1.



Hình 4.1 Đặt vị trí 2 thiết bị trên bao đàn.

Thiết bị đo lực được đặt với vị trí sao cho bằng với nửa chiều dài của bao đàn. Thiết bị đo gia tốc được đặt ở dưới đáy của bao đàn. Đầu tiên sẽ bật chế độ thu dữ liệu gia tốc và loadcell, sau đó đối tượng sẽ đâm vào loadcell nhiều mức độ lực khác nhau và cuối cùng là kết thúc quy trình. Quy trình được lặp đi lặp lại để thu được nhiều dữ liệu ở nhiều phương khác nhau của bao đàn hơn.



Hình 4.2 Quy trình thu dữ liệu từ loadcell và gia tốc.

Bắt quy trình thu dữ liệu, nhóm sẽ bật các chế độ thu dữ liệu từ các chương trình đã được viết ở 3.2.2. Sau đó đối tượng sẽ đeo găng bao tay và đắm vào loadcell ở các lực khác nhau. Thời gian giữa hai lần đắm là tự do.

4.2 XÂY DỰNG MÔ HÌNH HỒI QUY DỰ ĐOÁN LỰC

Mô hình mà nhóm hướng đến là mô hình hồi quy (regression). Nhóm sử dụng bộ thư viện Keras [14]. Input của mô hình là bộ các dữ liệu có sự tương quan giữa gia tốc và lực mà đã được nêu ở trong phần 5.1.6.

Bảng 4.1 Các thông số của mô hình CNN được sử dụng cho bài toán hồi quy.

Block	Layer	Activation
Function Block	Normalization	
	Flatten	
	Dense(64)	Relu
	Dense(1)	
Loss Function	Mean Square Error (mse)	

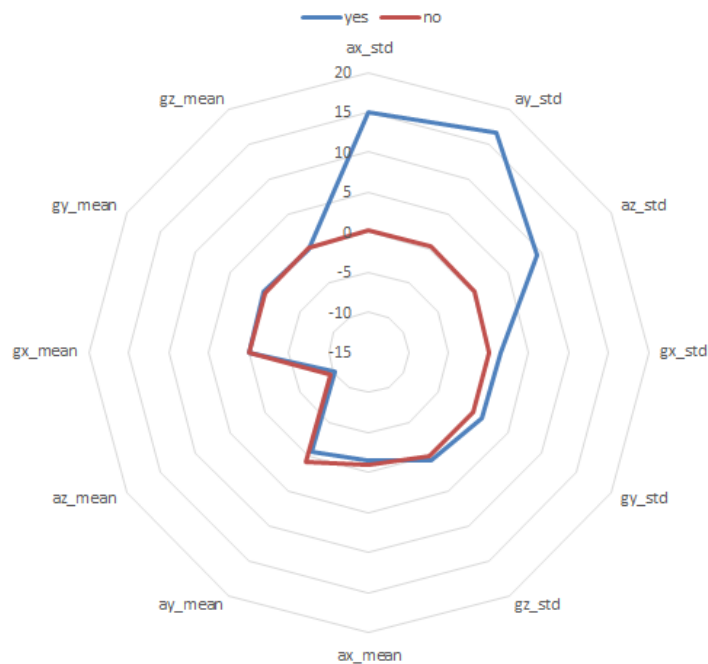
Block	Layer	Activation
Optimizer	Adam	

Bảng 4.2 Giải thích các thông số của mô hình hồi quy.

Thông số	Giải thích
Normalization	Chuẩn hóa dữ liệu đầu vào cho mô hình.
Flatten	Trải phẳng dữ liệu.
Dense(64), activation ReLu	Tạo ra một lớp ẩn với 64 neurons với hàm kích hoạt là ReLu.
Dense(1)	Tạo ra một lớp với số neuron output là một.
Loss Function Mean Square Error (mse)	Bình phương độ sai lệch giữa giá trị dự đoán và giá trị thực tế, sau đó lấy trung bình của các bình phương này.
Optimizer Adam	Adam (Adaptive Moment Estimation) là một thuật toán tối ưu hóa được sử dụng trong quá trình huấn luyện mô hình học máy để cập nhật các trọng số (weights) và điều chỉnh tốc độ học (learning rate) của mô hình.

4.3 PHÂN LOẠI CÚ ĐÁM

Từ các đoạn dữ liệu gia tốc thu được, ta có thể phân loại được đoạn dữ liệu có đám và không đám.



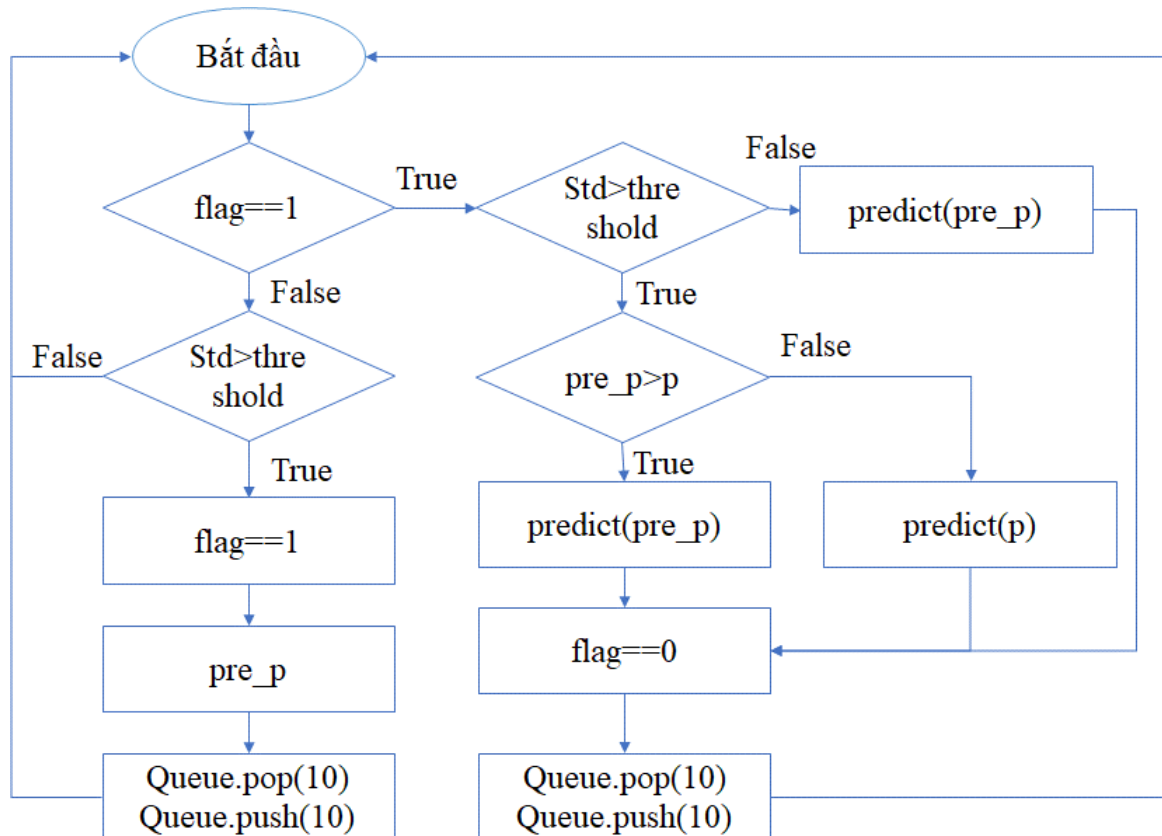
Hình 4.3 Khác biệt về gia tốc dài giữa dữ liệu có đấm và không đấm.

Qua quan sát ở Hình 4.3, nhóm có nhận xét rằng, có sự khác biệt rất lớn về gia tốc dài giữa có đấm và không có đấm. Đối với có đấm, các giá trị gia tốc dài theo phương xyz có sự thay đổi rất lớn, tuy nhiên các giá trị gia tốc góc cũng có sự thay đổi nhưng không đáng kể so với gia tốc dài.

Từ đó, nhóm nhận thấy dễ phân biệt dễ dàng giữa có đấm và không đấm, nhóm sử dụng gia tốc dài để phân biệt. Trong việc phân loại này, nhóm sử dụng hai phương pháp để có thể phân loại đoạn dữ liệu có đấm và đoạn dữ liệu không đấm. Hai phương pháp đó là sử dụng ngưỡng và sử dụng mô hình phân loại.

4.3.1 PHƯƠNG PHÁP NGƯỠNG (THRESHOLD)

Theo nhóm quan sát, giá trị a_std ảnh hưởng lớn đến giá lực, nên từ đó nhóm chọn a_std để làm giá trị ngưỡng, ngưỡng mà nhóm chọn ở đây là $a_std = 3$.



Hình 4.4 Lưu đồ thuật toán phát hiện cú đấm sử dụng ngưỡng.

Đầu tiên, chương trình sẽ kiểm tra các giá trị có vượt ngưỡng hay không, nếu không sẽ quay về bắt đầu, nếu có lần 1, thì chương trình sẽ lưu lại giá trị của cú đấm đó dưới dạng pre_p , sau đó chương trình sẽ kiểm tra tiếp cú đấm tiếp theo có vượt ngưỡng hay không, nếu không, giá trị pre_p sẽ được predict bởi mô hình đã được huấn luyện trước đó ($predict(pre_p)$). Nhưng ngược lại nếu chương trình phát hiện giá trị sau vượt ngưỡng, thì nó sẽ sử dụng giá trị này so sánh với giá trị pre_p trước đó. Nếu ngưỡng của $pre_p > p$ thì chương trình sẽ $predict(pre_p)$ còn không thì sẽ $predict(p)$. Sau đó quay trở lại trạng thái bắt đầu.

4.3.2 XÂY DỰNG MÔ HÌNH PHÂN LOẠI (CLASSIFICATION)

Bảng 4.3 Các thông số được sử dụng cho mô hình phân loại.

Block	Layer	Activation
Function Block	Normalization	
	Flatten	

Block	Layer	Activation
	Dense(16)	ReLu
	MaxPooling2D(3, 3)	
	Dense(32)	Relu
	Dense(64)	Relu
	AveragePooling2D(2, 2)	
	Dense(128)	Relu
	Dropout(0.5)	
	Dense(2)	Softmax
Loss Function	categorical_crossentropy	
Optimizer	Adam	

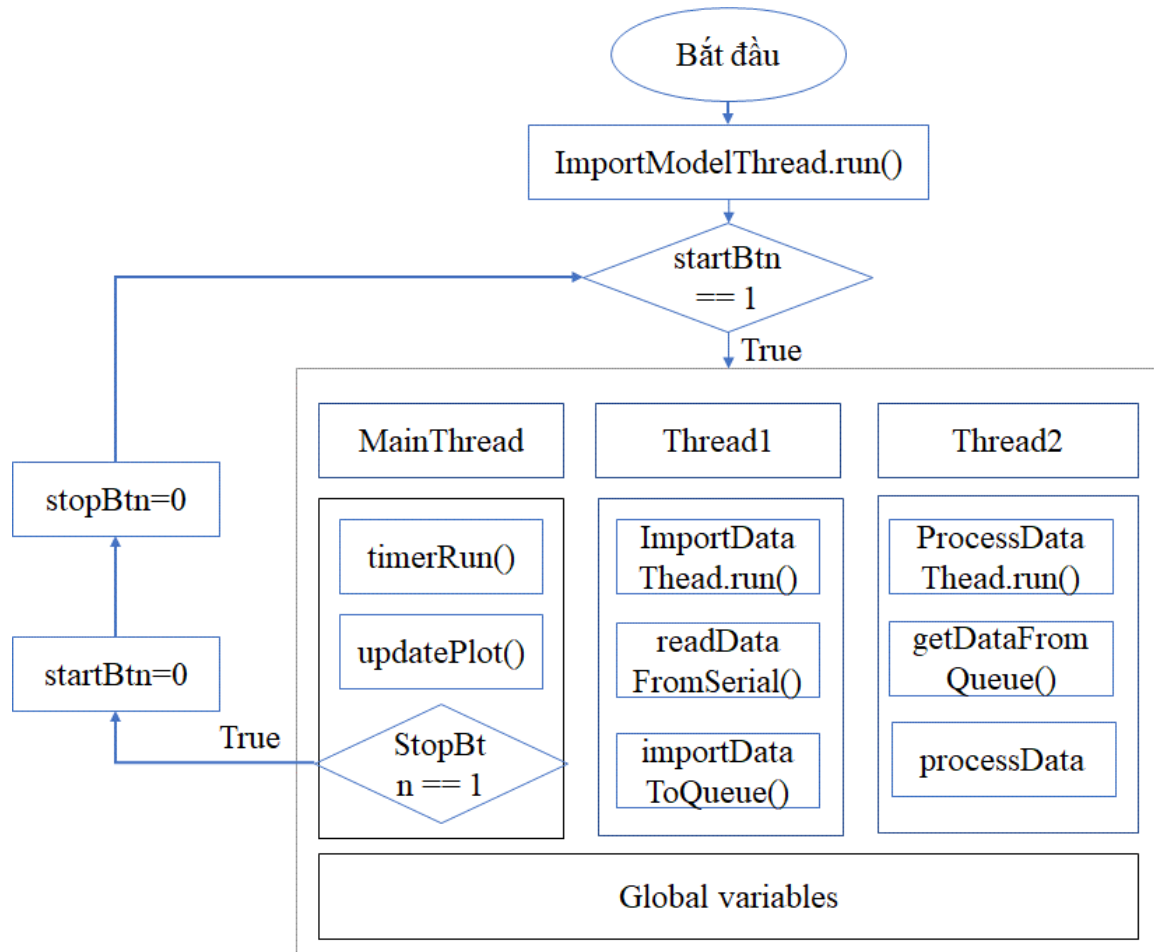
Bảng 4.4 Giải thích các thông số của mô hình phân loại [14].

Thông số	Giải thích
Normalization	Chuẩn hóa dữ liệu, dữ liệu được đưa vào window sẽ được chuẩn hóa trong một phạm vi từ (0, 1), bằng cách này, mô hình sẽ được tối ưu dữ liệu đầu vào hơn.
Flatten	Input window của mô hình là một dữ liệu sáu hàng mà hai mươi cột, dữ liệu này sẽ được trải phẳng ra.
Dense(16), activation ReLu	Tạo một lớp với 16 neurons. Với hàm kích hoạt là ReLu.
Dense(32), activation ReLu	Tạo một lớp với 32 neurons. Với hàm kích hoạt là ReLu.
Dense(64), activation ReLu	Tạo một lớp với 64 neurons. Với hàm kích hoạt là ReLu.
Dense(128), activation ReLu	Tạo một lớp với 128 neurons. Với hàm kích hoạt là ReLu.
Dense(2), activation Softmax	Tạo ra một lớp với 2 neurons. Với hàm kích hoạt là Softmax.

Thông số	Giải thích
MaxPooling2D(3, 3)	MaxPooling2D thực hiện phép tổng hợp (pooling) trên một vùng (window) 2D của đầu vào bằng cách lấy giá trị lớn nhất (max) từ vùng đó.
AveragePooling2D(2, 2)	AveragePooling2D thực hiện phép tổng hợp (pooling) trên một vùng (window) 2D của đầu vào bằng cách lấy giá trị trung bình (average) của các giá trị trong vùng đó.
Loss Function (categorical_crossentropy)	Hàm mất mát categorical crossentropy đo lường độ sai lệch giữa phân phối xác suất dự đoán và phân phối xác suất thực tế của các lớp đích. Nó được tính dựa trên hàm logarithm tự nhiên.
Optimizer adam	Adam (Adaptive Moment Estimation) là một thuật toán tối ưu hóa được sử dụng trong quá trình huấn luyện mô hình học máy để cập nhật các trọng số (weights) và điều chỉnh tốc độ học (learning rate) của mô hình.

4.4 DỰ ĐOÁN LỰC BẰNG MÔ HÌNH THỜI GIAN THỰC

Từ mô hình máy trạng thái được thể hiện ở Hình 3.24 nhóm xây dựng lên sơ đồ khối và hiện thực chương trình như sau:



Hình 4.5 Sơ đồ khởi hoạt động của phần mềm.

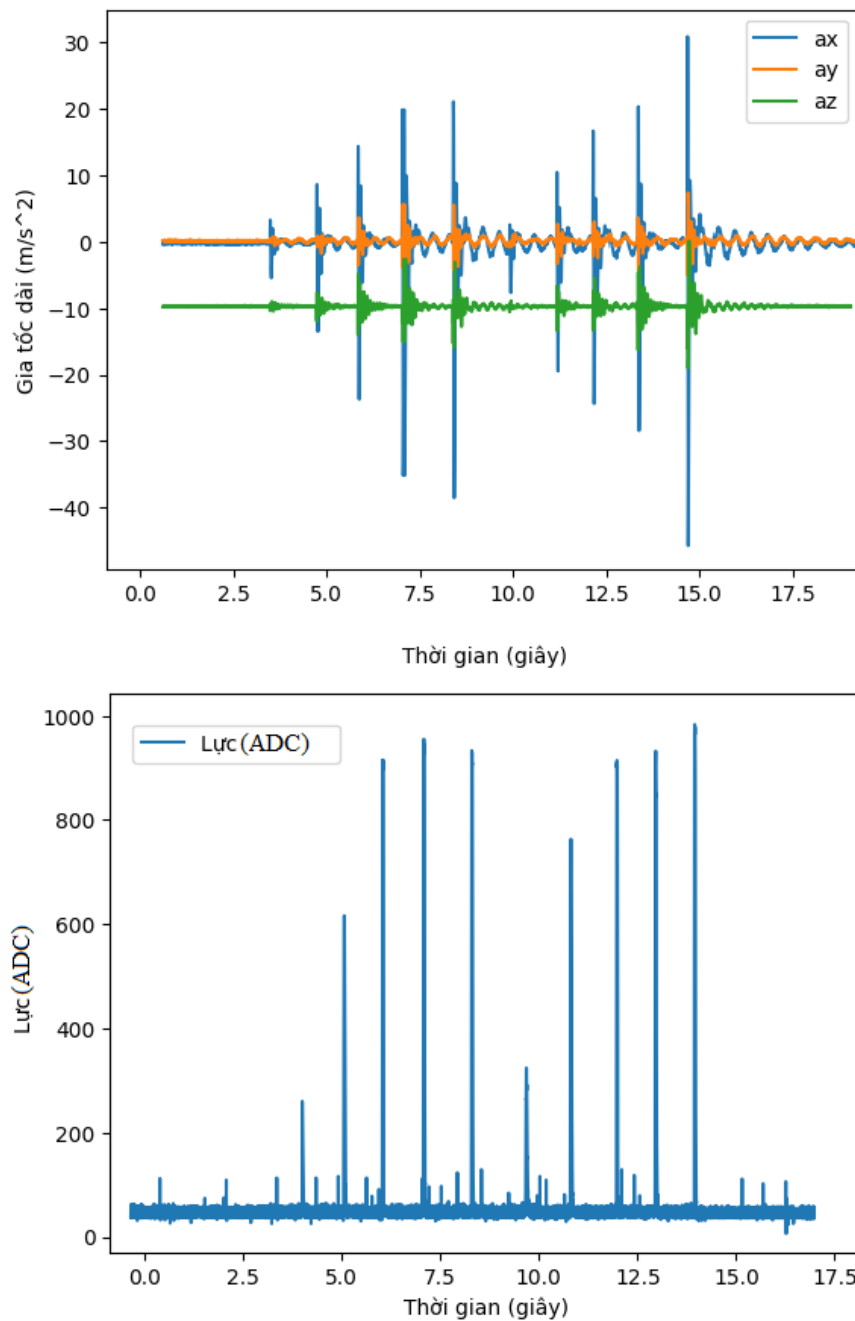
Khi khởi động, chương trình sẽ nhập vào mô hình đã được huấn luyện trước đó. Tiếp đến chương trình sẽ lắng nghe người dùng ấn phím bắt đầu (*startBtn*), sau khi ấn nút, chương trình sẽ tạo ra thêm 2 luồng (thread) mới bên cạnh luồng xử lý chính (mainThread). Bằng cách đó, chương trình sẽ nhận dữ liệu và đồng thời xử lý dữ liệu ở hai luồng khác nhau, như vậy sẽ giúp cho chương trình hoạt động ổn định hơn.

Ở luồng xử lý chính (mainThread), có nhiệm vụ cập nhật các giá trị lên giao diện, Ở luồng chính, nhóm sử dụng timer với độ trễ 1 giây. Bên cạnh đó luồng thứ nhất (Thread1) sẽ xử lý vấn đề nhập giá trị vào hàng đợi, sau đó các giá trị ở hàng đợi này sẽ được luồng thứ 2 (Thread2) lấy ra để xử lý.

CHƯƠNG 5 KẾT QUẢ VÀ ĐÁNH GIÁ

5.1 KẾT QUẢ

5.1.1 DỮ LIỆU GIA TỐC VÀ LỰC

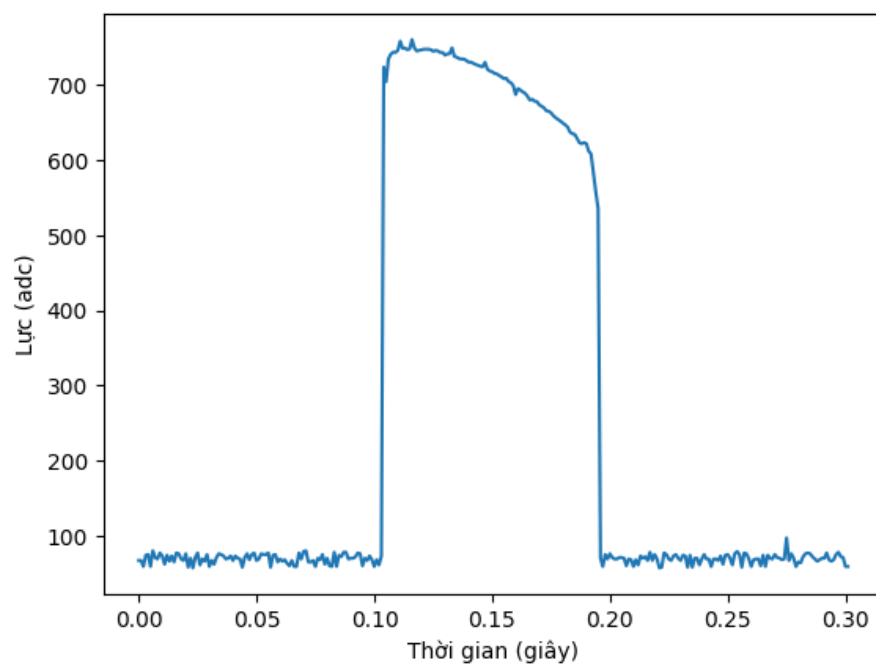


Hình 5.1 Dữ liệu gia tốc và lực thu được từ cảm biến gia tốc và cảm biến lực loadcell.

Trong khoảng thời gian 18 giây, đối tượng thực hiện các cú đâm vào loadcell, từ đó thu được các giá trị gia tốc và các giá trị lực như **Error! Reference source not found.** và Hình 5.1. Ta nhận thấy trong Hình 5.1, có nhiều giá trị bị nhiễu trong quá trình thu dữ liệu, các giá trị này không ảnh hưởng đến nhiều đến kết quả của giá trị lực.

5.1.2 PHÂN TÍCH DỮ LIỆU LỰC

Từ dữ liệu loadcell thu được ở Hình 5.1, nhóm tìm thời gian bao tay tiếp xúc với bao đệm như sau. Trích một đoạn dữ liệu từ một cú đâm, ta có:



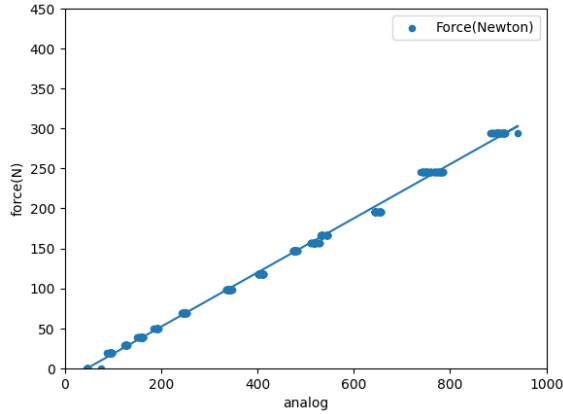
Hình 5.2 Thời gian bao tay tiếp xúc với bao đệm trong một cú đâm.

Từ Hình 5.2, nhóm nhận xét thời gian mà bao tay tiếp xúc với bao đệm có khoảng từ 0.1 giây đến 0.2 giây, có giá trị trung bình là 100 mili-giây (ms). Trong khoảng thời gian từ 0 – 0.1 ms và từ 0.2 – 0.3 ms thì không có giá trị cú đâm.

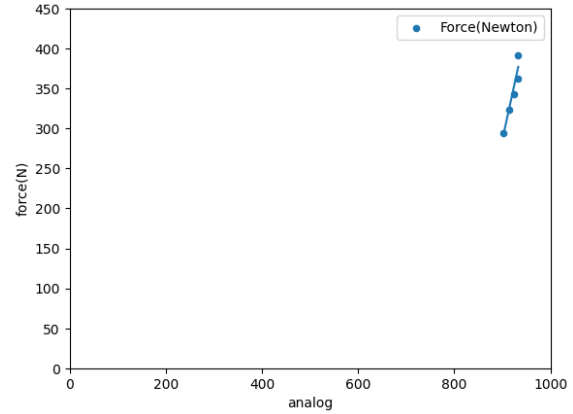
Giá trị này ảnh hưởng rất lớn đến SR của thiết bị và cũng nhờ thông số này, nhóm đã chọn ra window phù hợp để chạy thời gian thực.

5.1.3 ĐƯỜNG ĐẶC TÍNH CỦA LOADCELL

Sau khi cho vật nặng có khối lượng là M trải dài từ 0 – 45kg. Nhóm thu được đường đặc tính của loadcell như sau:



Hình 5.3 Đường đặc tính của loadcell trong đoạn 0 – 30kg.



Hình 5.4 Đường đặc tính của loadcell trong đoạn 30 – 45kg.

Sau khi sử dụng mô hình hồi quy tuyến tính để suy được đường đặc tính của loadcell, cuối cùng ta có phương trình giữa giá trị analog và lực như sau:

$$F = \begin{cases} 0.3392 \times adc - 15.9838 & (adc \leq 900) \\ 2.9296 \times adc - 2352.9 & (adc > 900) \end{cases}$$

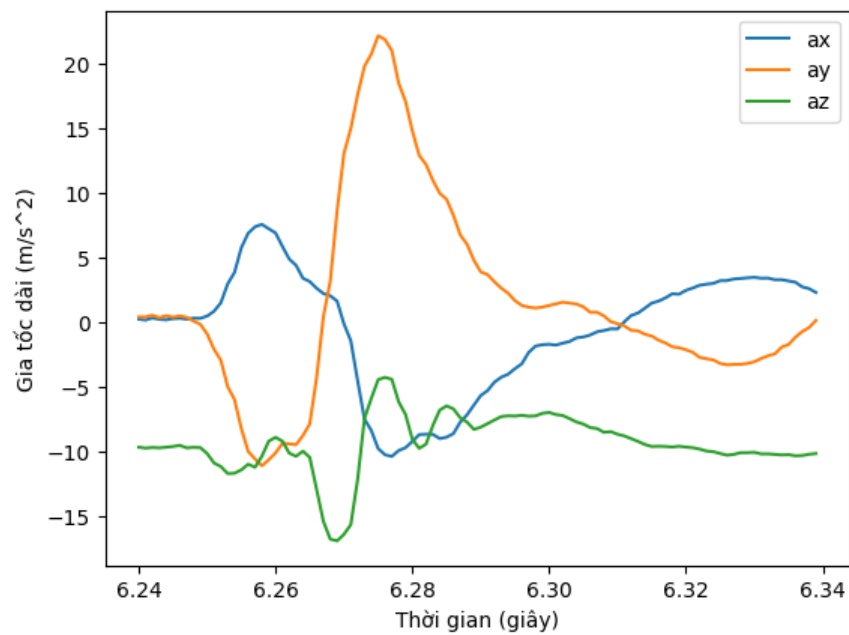
(5.1)

Trong đó:

- F : lực của cú đâm (Newton).
- adc : giá trị ADC mà ESP32 trả về.

5.1.4 TẦN SỐ LẤY MẪU CỦA THIẾT BỊ

Trong khảo sát cho thấy SR của cảm biến đóng vai trò quan trọng trong việc đo những hoạt động thể thao [15]. Thống kê trong bài báo cho thấy, hầu hết các cảm biến được sử dụng với SR lớn hơn 1000Hz, tuy nhiên có một vài ứng dụng cần SR đạt đến 400kHz [16]. Trong một nghiên cứu khác, chỉ sử dụng SR là 300Hz [2]. Tuy nhiên nhóm sẽ phân tích với dữ liệu 1 cú đâm với SR = 1000Hz và SR = 180Hz, từ đó đưa ra SR phù hợp nhất. Điều này giảm được tải hoạt động cho timer trong việc chạy thời gian thực.

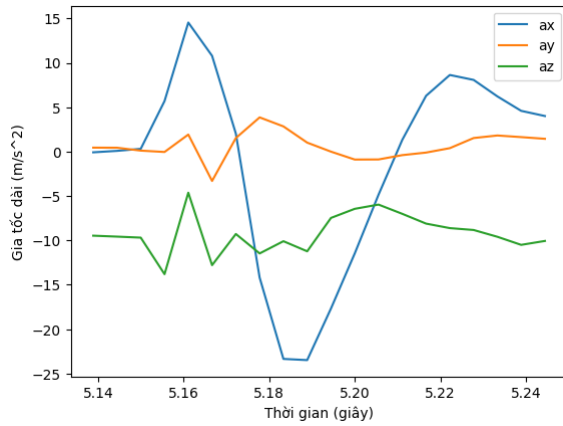


Hình 5.5 Thông tin một cú đấm theo phương Oy trong 100 ms với tần số lấy mẫu 1000Hz.

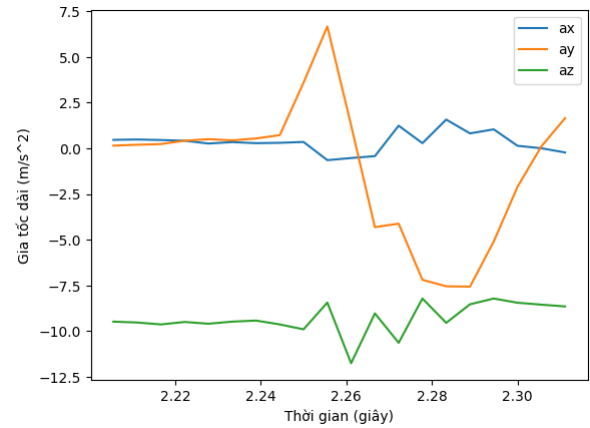
So sánh dữ liệu gia tốc dài ở Hình 5.5 và Hình 5.7 hầu như không có gì khác biệt ở hình dạng đồ thị. Từ đó, nhóm chọn sử dụng ở tần số 180Hz để cho sử dụng làm dữ liệu cho mô hình hồi quy và mô hình phân loại cú đấm.

5.1.5 PHÂN TÍCH DỮ LIỆU GIA TỐC

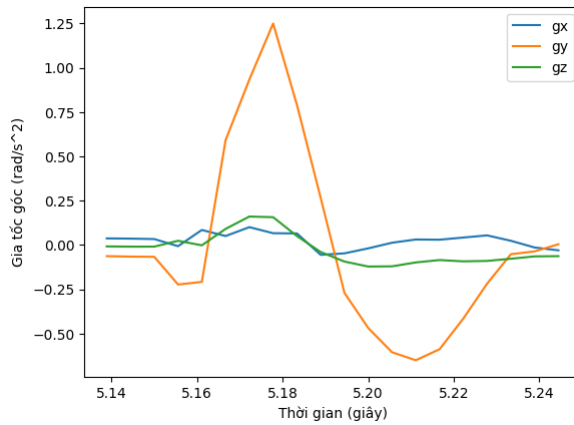
Với bộ dữ liệu gia tốc đạt được, nhóm đã trích xuất ra dữ liệu thu được từ 1 cú đấm theo các phương khác nhau.



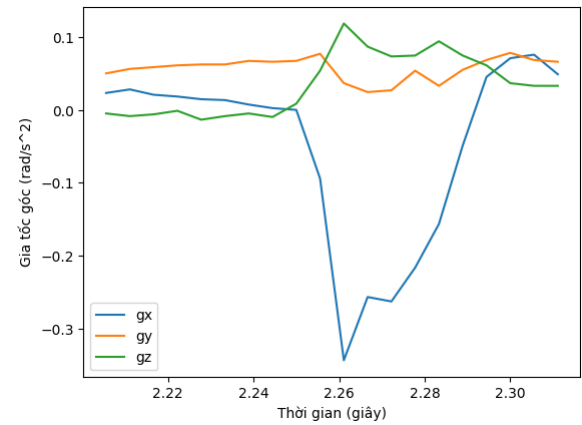
Hình 5.6 Dữ liệu gia tốc dài khi tác dụng lực F theo phương Ox trong khoảng thời gian diễn ra một cú đấm (100 ms).



Hình 5.7 Dữ liệu gia tốc dài khi tác dụng lực F theo phương Oy trong khoảng thời gian diễn ra một cú đấm (100 ms).



Hình 5.8 Dữ liệu gia tốc góc khi tác dụng lực F theo phương Ox trong khoảng thời gian diễn ra một cú đấm (100 ms).



Hình 5.9 Dữ liệu gia tốc góc khi tác dụng lực F theo phương Oy trong khoảng thời gian diễn ra một cú đấm (100 ms).

Khi cú đấm tác dụng với bao đấm theo phương Ox thì gia tốc dài Ox và gia tốc góc Oy có độ thay đổi. Ngược lại, khi tác dụng vào bao đấm theo phương Oy thì gia tốc dài Oy và gia tốc góc Ox có độ thay đổi lớn.

5.1.6 SỰ TƯƠNG QUAN GIỮA GIA TỐC VÀ LỰC

Từ 364 cú đấm thu được, nhóm nhận xét về thay đổi giữa gia tốc dài/góc như sau. Đối với gia tốc dài và gia tốc góc ở các phương $Oxyz$ có độ tương quan thấp với lực. Tuy nhiên, khi tổng hợp các gia tốc dài/góc này, thì có độ tương quan lớn ở Hình

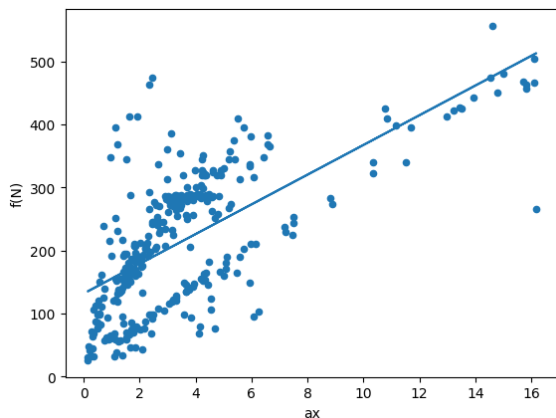
5.16 và Hình 5.17. Trong Hình 5.16 và Hình 5.17, gia tốc dài tổng hợp và gia tốc góc tổng hợp lần lượt có công thức như sau:

$$a = \sqrt{ax^2 + ay^2 + az^2}$$

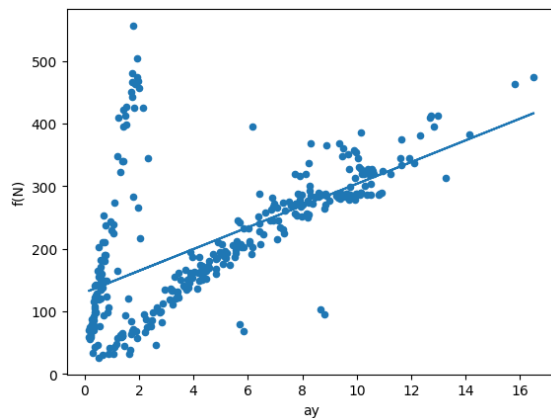
$$g = \sqrt{gx^2 + gy^2 + gz^2}$$

Trong đó:

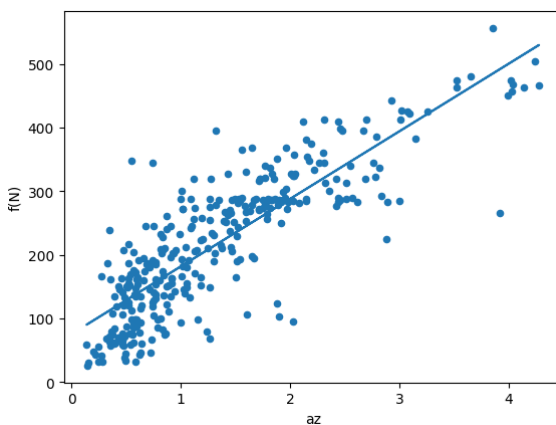
- a : gia tốc dài tổng hợp (m/s^2).
- g : gia tốc góc tổng hợp (rad/s^2).
- ax : gia tốc dài theo phương Ox (m/s^2).
- ay : gia tốc dài theo phương Oy (m/s^2).
- az : gia tốc dài theo phương Oz (m/s^2).
- gx : gia tốc dài theo phương Ox (rad/s^2).
- gy : gia tốc dài theo phương Oy (rad/s^2).
- gz : gia tốc dài theo phương Oz (rad/s^2).



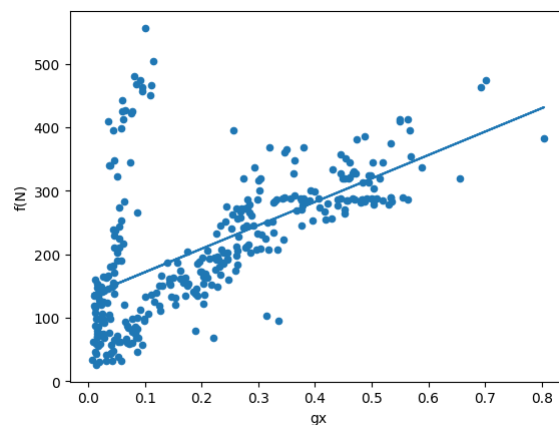
Hình 5.10 Tương quan giữa gia tốc dài theo phương Ox và lực đạt 0.657057.



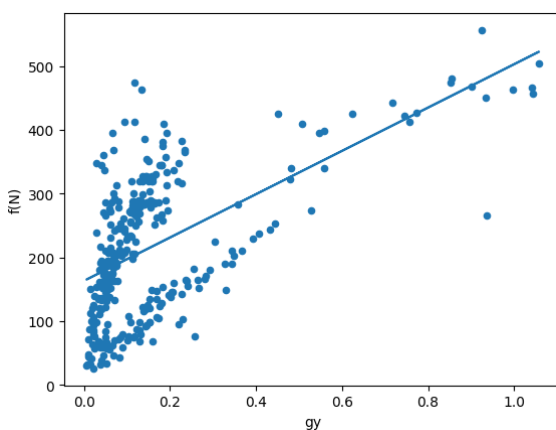
Hình 5.11 Tương quan giữa gia tốc dài theo phương Oy và lực đạt 0.637418.



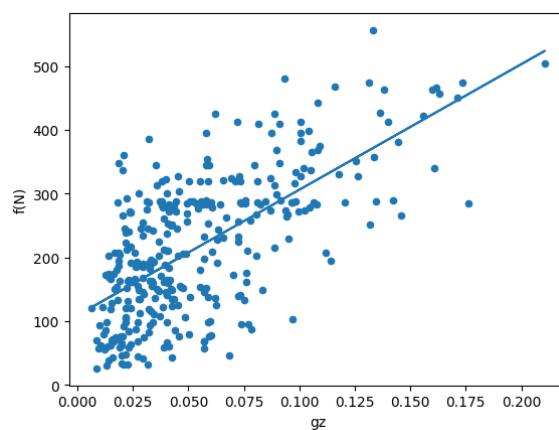
Hình 5.12 Tương quan giữa gia tốc dài theo phương Oz và lực đạt 0.829592.



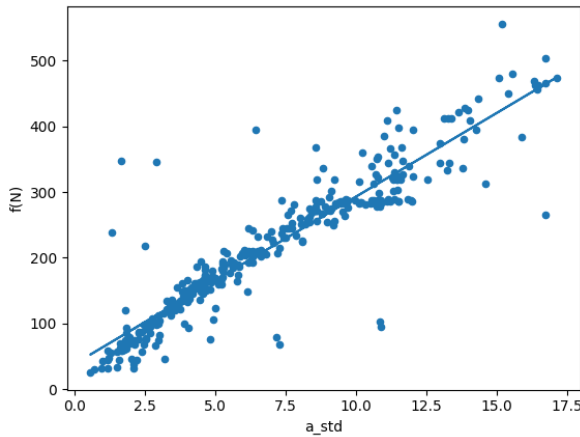
Hình 5.13 Tương quan giữa gia tốc góc theo phương Ox và lực đạt 0.617840.



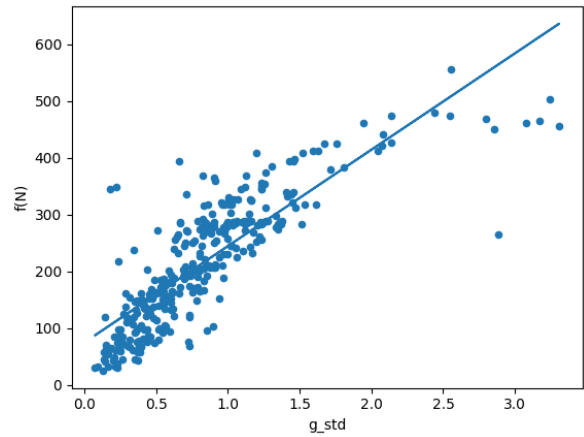
Hình 5.14 Tương quan giữa gia tốc góc theo phương Oy và lực đạt 0.563325.



Hình 5.15 Tương quan giữa gia tốc góc theo phương Oz và lực đạt 0.651737.



Hình 5.16 Tương quan giữa gia tốc dài tổng hợp và lực đạt 0.926602.



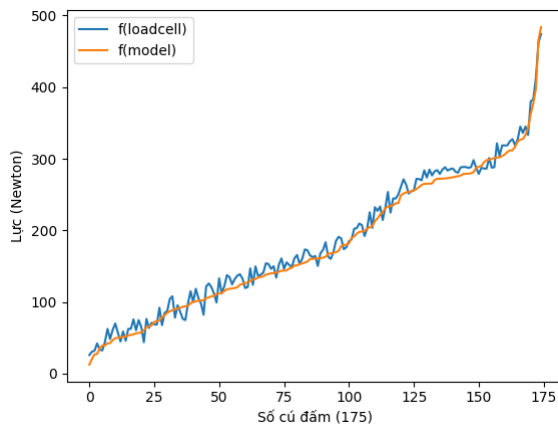
Hình 5.17 Tương quan giữa gia tốc góc tổng hợp và lực đạt 0.885718.

5.1.7 MOMENT QUÁN TÍNH BAO ĐẰM

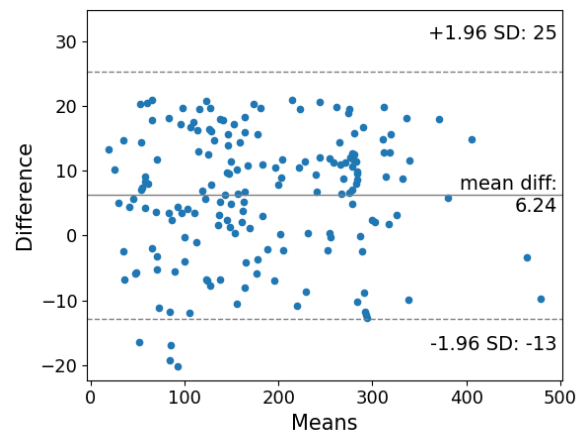
Từ phương trình (2.3), nhóm đã tiến hành trên 20 mẫu và thu được moment quán tính của bao đệm có giá trị như sau: $I_p = 218.479 (kg \times m^2)$. Trong khi đó, từ phương trình (2.4) nhóm tính được kết quả moment quán tính của bao đệm có giá trị là: $I_p = 222.41 (kg \times m^2)$. Vì kết quả của 2 phương thức tính này là khác nhau, nhưng cho kết quả có độ chênh lệch không đáng kể, nên nhóm lấy giá trị moment quán tính được tính bằng thực nghiệm để làm tham khảo cho lần này.

5.1.8 SO SÁNH LỰC ĐO BẰNG LOADCELL – DỰ ĐOÁN BẰNG MÔ HÌNH – TÍNH BẰNG LÝ THUYẾT

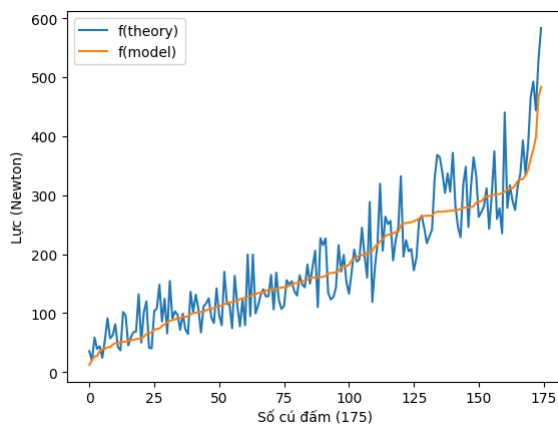
Trong 175 cú đâm, nhóm đã tiến hành và phân tích so sánh với các dữ liệu được đo trực tiếp bằng loadcell và được dự đoán bởi mô hình hồi quy được xây dựng ở 4.2.



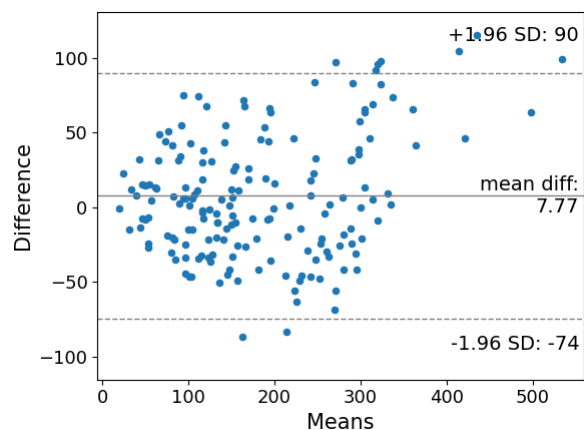
Hình 5.18 So sánh lực giữa đo tham chiếu bằng loadcell và lực dự đoán từ mô hình.



Hình 5.19 Biểu đồ Bland–Altman giữa giá trị lực tham chiếu bằng loadcell và lực dự đoán bằng mô hình.



Hình 5.20 So sánh lực giữa tính bằng lý thuyết và lực dự đoán từ mô hình.

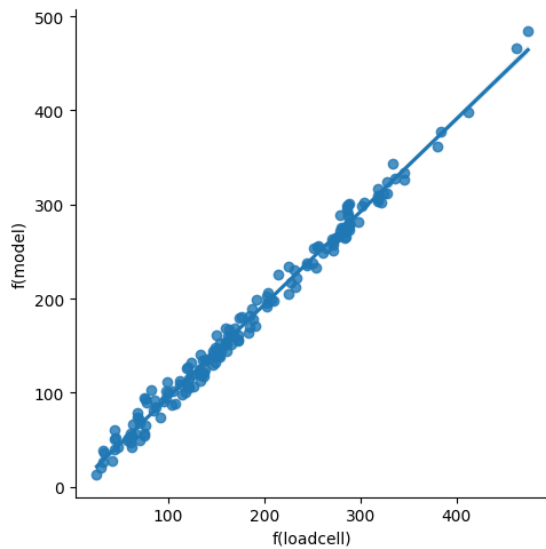


Hình 5.21 Biểu đồ Bland–Altman giữa giá trị lực được tính bằng lý thuyết và lực bằng mô hình.

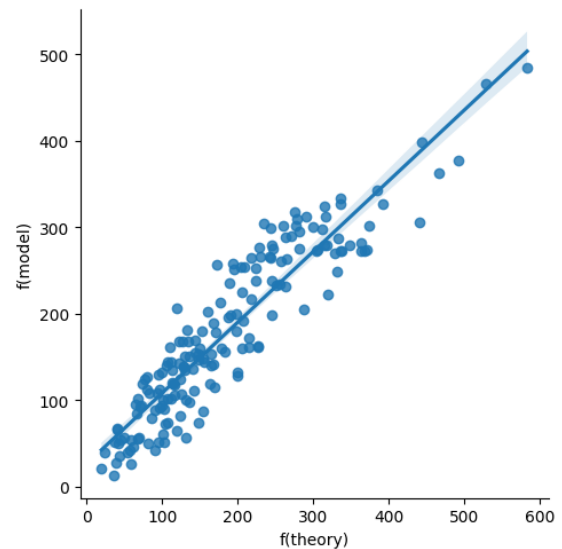
Dựa vào Hình 5.18 ta thấy, 2 đường lực mô hình dự đoán và lực tham chiếu bằng loadcell có sai lệch nhỏ. Nhưng so với tính bằng cơ học ở Hình 5.20 thì có độ sai lệch lớn.

Độ thay đổi trung bình ở Hình 5.20 là 6.24, std cận trên là 25 và std cận dưới là -13. Có nhiều outlier trong đoạn 100N. Độ thay đổi trung bình ở Hình 5.20 là 7.77, với std cận trên và cận dưới lần lượt là 90 và -74, có nhiều outlier ở đoạn 200 – 300N. Qua đây cho thấy, độ thay đổi trung bình giữa lực từ mô hình khi so với lực tham chiếu loadcell sẽ nhỏ hơn so với lực được tính bằng lý thuyết. Độ thay đổi cận trên và cận dưới của mô hình so với loadcell thì hẹp hơn khi so với khoảng lực được tính bằng lý

thuyết. Từ đó, rút ra kết luận lực dự đoán bằng mô hình sẽ cho kết quả sát với lực tham chiếu loadcell hơn và được tính bằng lý thuyết.



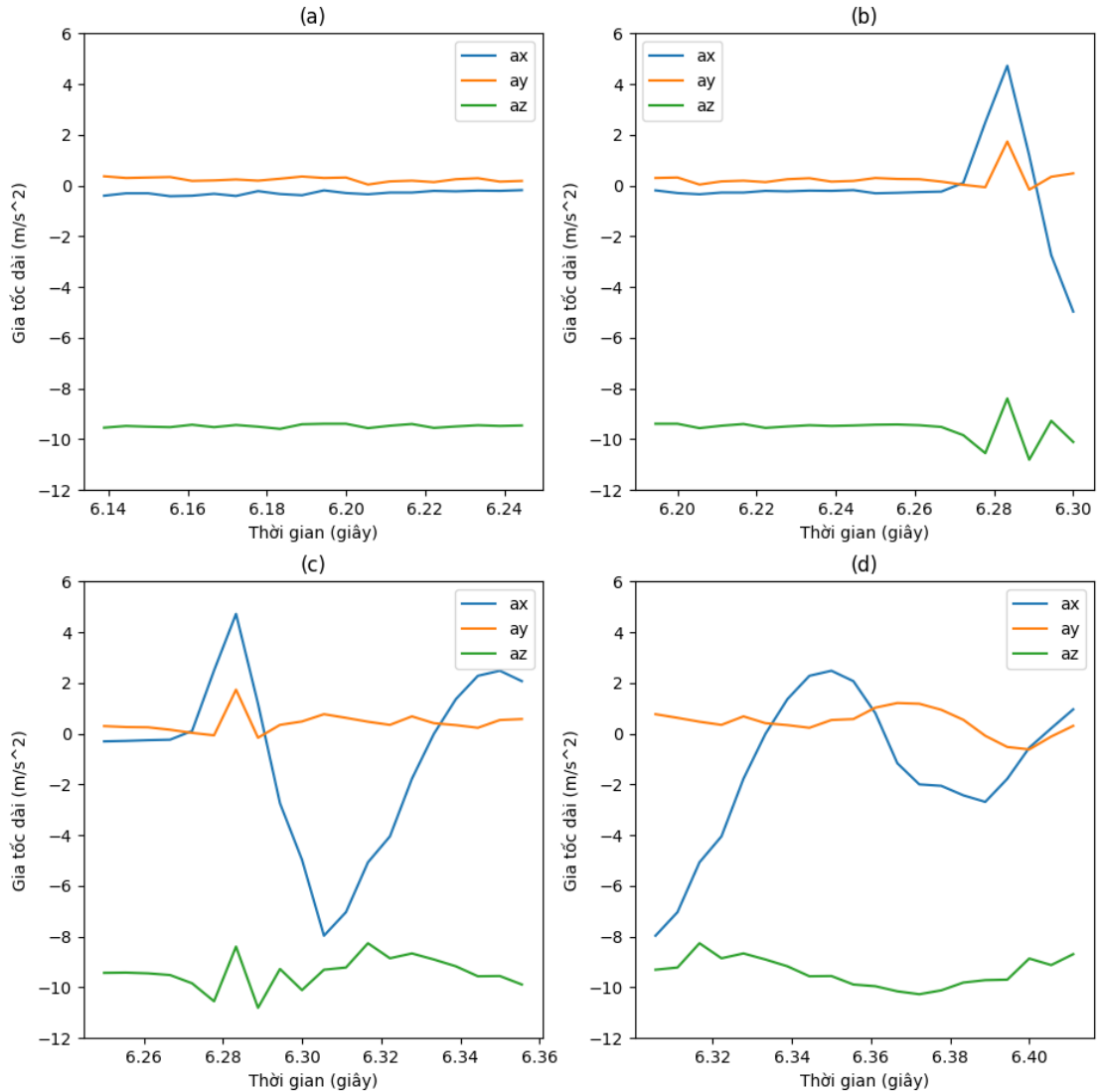
Hình 5.22 Tương quan giữa lực đo bằng loadcell và lực đo bằng mô hình.



Hình 5.23 Tương quan giữa lực đo bằng lý thuyết và lực đo bằng mô hình.

Dựa vào Hình 5.22 và Hình 5.23, nhóm nhận xét có sự tương quan mạnh giữa lực giữa lực từ mô hình – loadcell, tuy nhiên có sự tương quan kém giữa lực từ mô hình – lực tính bằng lý thuyết.

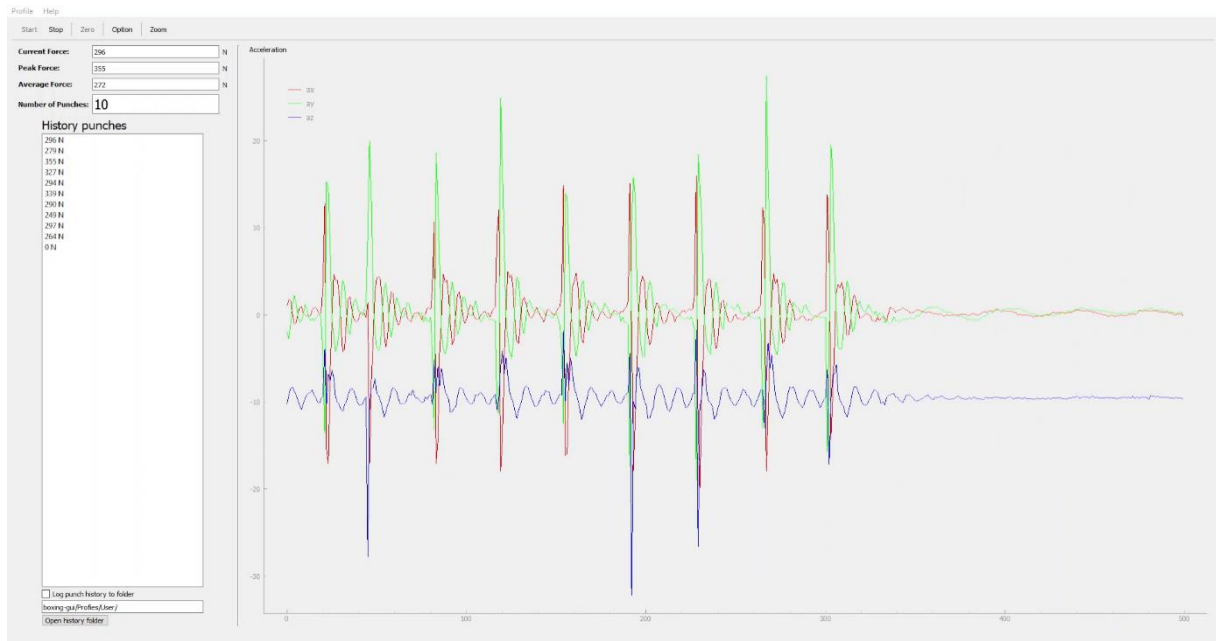
5.1.9 KẾT QUẢ MÔ HÌNH PHÂN LOẠI CÚ ĐÁM



Hình 5.24 Xác xuất nhận dạng cú đấm của mô hình theo thời gian thực: (a) Cửa sổ dữ liệu hoàn toàn không có cú đấm, xác suất có đấm ~ 0.00 . (b) Cửa sổ dữ liệu có một phần dữ liệu đầu cú đấm, xác suất có đấm ~ 0.21 . (c) Cửa sổ dữ liệu có toàn bộ dữ liệu cú đấm, xác suất có đấm ~ 0.99 . (d) Cửa sổ dữ liệu có một phần dữ liệu sau cú đấm, xác suất có đấm ~ 0.00 .

Với dữ liệu trung bình của một cú đấm là 100ms, nhóm cho các window khớp với từng phân đoạn của một cú đấm để xem được xác suất mà mô hình dự đoán. Trong Hình (a), dữ liệu input đầu vào không khớp với bất kỳ thông tin nào của một cú đấm, cho nên xác suất mà mô hình dự đoán là ~ 0.00 . Tuy nhiên trong Hình (b), window đã được offset và khớp được với 50ms đầu của một cú đấm, tuy nhiên chỉ khớp với 50ms đầu, mô hình

phần loại có xác suất vẫn rất thấp ~ 0.21 . Tốt hơn thế, trong Hình (c), toàn bộ thông tin của một cú đấm trong khoảng 100ms khớp với window cho trước, từ đó mô hình phân loại dự đoán có xác suất là cao nhất ~ 0.99 . Tiếp đến trong Hình (d) có thông tin 50ms sau của một cú đấm, tuy nhiên mô hình phân loại đưa ra xác suất rất thấp ~ 0.00 .



Hình 5.25 Chương trình chạy thời gian thực dựa trên giải thuật phân loại cú đấm sử dụng ngưỡng (threshold).

5.2 ĐÁNH GIÁ

5.2.1 ĐÁNH GIÁ MÔ HÌNH DỰ ĐOÁN LỰC

Ưu điểm:

- Các giá trị dự đoán gần khớp với các kết quả được đo ở loadcell và được tính bằng cơ học.
- Mô hình được đưa lên chạy thời gian thực nên cho kết quả khác với so với các dữ liệu được cắt bằng tay.

Nhược điểm:

- Lực không lớn ($< 500\text{N}$).

5.2.2 ĐÁNH GIÁ GIỮA VIỆC SỬ DỤNG NGƯỠNG VÀ MÔ HÌNH PHÂN LOẠI TRONG VIỆC NHẬN DẠNG CÚ ĐÁM THỜI GIAN THỰC.

Bảng 5.1 Đánh giá phương pháp sử dụng ngưỡng và mô hình phân loại.

Ngưỡng (threshold)	Mô hình phân loại
Dễ hiện thực.	Khó hiện thực.
Sử dụng giải thuật để phân loại cú đấm.	Sử dụng mô hình phân loại.
Tính toán ít.	Tính toán nhiều (do sử dụng mô hình).
Không cần tốn thời gian để load mô hình cho người dùng.	Tốn thời gian load mô hình.

Từ những yếu tố trên, nhóm đã quyết định chọn sử dụng ngưỡng dựa trên việc chọn threshold phù hợp để hiện thực cho giải thuật phân loại cú đấm.

5.2.3 ĐÁNH GIÁ PHẦN MỀM THỜI GIAN THỰC

Ưu điểm:

- Có tính ứng dụng vào đời sống.
- Giúp các vận động viên luyện tập biết được số cú đấm, lực đấm của mình.
- Lưu lại lịch sử luyện tập của vận động viên.

Nhược điểm:

- Giao diện chưa có nhiều tính năng hấp dẫn, chỉ có chế độ luyện tập.

TÀI LIỆU THAM KHẢO

- [1] K. A. R. ,. B. C. L. ,. R. J. G. a. C. M. P. John D Pierce, “Direct Measurement of Punch Force During,” *Journal of Quantitative Analysis in Sports*, 2006.
- [2] K. W. M. T. Sosopoulos, “IoT smart athletics: Boxing glove sensors implementing machine learning for an integrated training solution,” 2021 .
- [3] Z. S. M. S.-E. T. N. J. M.-R. P. Ł. M. J. G. M. G. KRZYSZTOF BUŚKO1, “Measuring the force of punches and kicks among combat sport athletes using a modified punching bag with an embedded accelerometer,” 2016.
- [4] C. B. O. T. M. A. Bergun Meric Bingul, “The effects of impact forces and kinematics of two,” *SCIENCE OF MARTIAL ARTS*, 2018.
- [5] DANB, “www.kaggle.com,” 2918. [Trực tuyến]. Available: <https://www.kaggle.com/code/dansbecker/rectified-linear-units-relu-in-deep-learning>.
- [6] STMicroelectronics, “LSM6DSO32,” 2020.
- [7] R. J. Gathercole, T. Stellingwerff và B. C. Sporer, “Effect of Acute Fatigue and Training Adaptation on Countermovement Jump Performance in Elite Snowboard Cross Athletes,” *Journal of Strength and Conditioning Research*, 2015.
- [8] Espressif, "ESP32," 2022.
- [9] Renzo, “www.mischianti.org,” 2023. [Trực tuyến]. Available: www.mischianti.org.
- [10] OEM, “thegioiic,” OEM, [Trực tuyến]. Available: <https://www.thegioiic.com/cam-bien-can-nang-load-cell-20kg-24bit-hx711>.

- [11] w. team, “wemos,” [Trực tuyến]. Available: https://www.wemos.cc/en/latest/_static/files/sch_battery_v1.3.0.pdf.
- [12] T. Instruments, “INA12x Precision, Low-Power Instrumentation Amplifiers”.
- [13] T. A. Team, “Arduino,” Arduino, [Trực tuyến]. Available: <https://www.arduino.cc>.
- [14] Keras, “Keras,” [Trực tuyến]. Available: https://keras.io/getting_started/.
- [15] *. F. 1. T. 2. M. B. 3. E. 2. G. 3. a. A. 1. Lin Zhou 1, “How We Found Our IMU: Guidelines to IMU Selection and a Comparison of Seven IMUs for Pervasive Healthcare Applications,” 2020.
- [16] I. Khasanshin, “Application of an Artificial Neural Network to Automate the Measurement of Kinematic Characteristics of Punches in Boxing,” 2021.
- [17] T. J. Suchomel, C. A. Bailey, C. J. Sole, J. L. Grazer và G. K. Beckham, “Using Reactive Strength Index-Modified as an Explosive Performance Measurement Tool in Division I Athletes,” *Journal of Strength and Conditioning Research*, 2015.
- [18] Legact, “Datasheet type: RP-S40-ST Force Sensitive Resistor,” [Trực tuyến]. Available: http://m.eleparts.co.kr/data/_gextends/good-pdf/201904/good-pdf-7495470-1.pdf.
- [19] J. Hrisko, “makersportal,” [Trực tuyến]. Available: <https://makersportal.com/blog/2020/5/24/force-sensitive-resistors-fsrs-arduino>.