

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA QUẢN LÝ DỰ ÁN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: QUẢN LÝ CÔNG NGHIỆP

ĐỀ TÀI:
PHÁT TRIỂN HỆ THỐNG GIÁM SÁT TỰ ĐỘNG
NHẪM NÂNG CAO CHẤT LƯỢNG QUY TRÌNH
LẮP RÁP SẢN PHẨM TẠI CÔNG TY TNHH
DAIWA VIỆT NAM

GVHD : TS. Nguyễn Thị Phương Quyên
SVTH : Mai Văn Thuận
LỚP : 21QLCN2
MSSV : 118210203

Đà Nẵng, tháng 6 năm 2025

TÓM TẮT ĐỒ ÁN

Tên đề tài: Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam.

Sinh viên thực hiện: Mai Văn Thuận

Số thẻ sinh viên: 118210203

Lớp: 21QLCN2

Đồ án được thực hiện với mục tiêu xây dựng một giải pháp công nghệ nhằm hỗ trợ giám sát thao tác công nhân trong công đoạn lắp ráp O.Slider tại bộ phận V159 – Công ty TNHH Daiwa Việt Nam. Thực trạng cho thấy các lỗi thao tác như lắp thiếu linh kiện và lắp sai vị trí đang chiếm tỷ lệ lớn, gây ảnh hưởng đáng kể đến chất lượng sản phẩm và chi phí sản xuất.

Để giải quyết vấn đề trên, đề tài đã ứng dụng mô hình Yolo11 trong nhận diện thao tác, kết hợp với thư viện OpenCV xử lý hình ảnh theo thời gian thực. Phần mềm giám sát được xây dựng bằng Python sử dụng Tkinter để tạo giao diện, cùng với Arduino làm thiết bị cảnh báo khi phát hiện thao tác sai. Dữ liệu hình ảnh được thu thập tại xưởng, dán nhãn bằng Label Studio và huấn luyện trên nền tảng Google Colab.

Kết quả triển khai cho thấy hệ thống có khả năng phát hiện sớm thao tác sai với độ chính xác cao, giúp giảm thiểu tỷ lệ NG của công đoạn từ 0.54% giảm còn 0.08%. Việc ứng dụng hệ thống không chỉ mang lại hiệu quả tài chính thông qua tiết kiệm chi phí sửa lỗi, mà còn nâng cao nhận thức thao tác cho công nhân, góp phần xây dựng môi trường sản xuất chuyên nghiệp và ổn định hơn.

LỜI CẢM ƠN

Trước tiên, em xin bày tỏ lòng biết ơn sâu sắc đến TS. Nguyễn Thị Phương Quyên -Giảng viên bộ môn Quản lý Công nghiệp, người đã đồng hành, tận tâm hướng dẫn và chỉ bảo em trong suốt quá trình thực hiện đề tài. Những đóng góp quý báu từ cô, từ việc định hướng nghiên cứu, gợi mở ý tưởng cho đến từng nhận xét sâu sắc trong quá trình triển khai, đã giúp em từng bước hoàn thiện đề tài một cách khoa học, logic và có tính thực tiễn cao. Bên cạnh đó, sự nghiêm túc, cẩn trọng và tâm huyết trong phong cách làm việc của cô cũng là tấm gương để em học hỏi, hình thành cho mình phương pháp tư duy và tác phong làm việc chuyên nghiệp hơn trên hành trình phát triển sự nghiệp sau này.

Tiếp đến, em xin chân thành cảm ơn quý thầy cô trong Khoa Quản lý Dự án – Trường Đại học Bách khoa, Đại học Đà Nẵng. Nhờ sự giảng dạy, truyền đạt kiến thức tận tâm của quý thầy cô trong suốt quá trình học tập mà em đã được trang bị những nền tảng kiến thức chuyên môn vững chắc về quản lý công nghiệp, hệ thống sản xuất, cũng như các công cụ quản lý chất lượng, giúp em có đủ năng lực để tiếp cận, triển khai và hoàn thiện đề tài tốt nghiệp này.

Em xin gửi lời cảm ơn chân thành đến anh Nguyễn Văn Phú – Giám đốc Công ty TNHH Daiwa Việt Nam, người đã tạo điều kiện cho em được tiếp cận thực tế tại doanh nghiệp, đồng thời luôn hỗ trợ, sắp xếp công việc một cách thuận lợi, tạo môi trường thực tập lý tưởng để em thu thập dữ liệu và kiểm nghiệm mô hình. Sự hỗ trợ quý báu của anh là yếu tố quan trọng giúp em có thể áp dụng lý thuyết vào thực tiễn và hoàn thiện đề tài một cách thiết thực hơn.

Em cũng xin cảm ơn các anh chị tại bộ phận V159 của công ty đã luôn nhiệt tình hỗ trợ, tạo điều kiện cho em trong quá trình thu thập dữ liệu, thử nghiệm hệ thống và xử lý những phát sinh thực tế tại hiện trường.

Sau cùng, em xin gửi lời cảm ơn chân thành đến gia đình, người thân, bạn bè – những người luôn là nguồn động viên to lớn, đồng hành, sẻ chia cùng em trong suốt chặng đường học tập và hoàn thành đề tài. Chính sự ủng hộ và khích lệ của mọi người đã tiếp thêm cho em sức mạnh vượt qua mọi khó khăn trong quá trình học tập và nghiên cứu.

Một lần nữa, em xin chân thành cảm ơn tất cả mọi người đã đồng hành cùng em trong suốt chặng đường vừa qua.

LỜI CAM ĐOAN

Tôi tên là Mai Văn Thuận, sinh viên lớp 21QLCN2 xin cam đoan:

- Đồ án tốt nghiệp là thành quả của quá trình nghiên cứu học hỏi dựa trên cơ sở lý thuyết, số liệu thực tế tại Công ty thu thập được, thực hiện theo sự hướng dẫn của giảng viên hướng dẫn.
- Đồ án được thực hiện hoàn toàn mới, là kết quả công sức của cá nhân tôi.
- Mọi sự tham khảo trong đồ án được trích nguồn và nằm trong danh mục tài liệu tham khảo.
- Mọi sao chép không hợp lệ, vi phạm quy chế nhà trường, tôi xin hoàn toàn chịu trách nhiệm.

Đà Nẵng, ngày 25 tháng 6 năm 2025

Sinh viên thực hiện

(Ký, ghi rõ họ tên)

MỤC LỤC

TÓM TẮT ĐỒ ÁN.....	i
LỜI CẢM ƠN	ii
LỜI CAM ĐOAN	iii
MỤC LỤC	iv
DANH MỤC BẢNG	viii
DANH MỤC HÌNH ẢNH	ix
CHƯƠNG 1: GIỚI THIỆU	1
1.1 Lý do chọn đề tài.....	1
1.2 Mục tiêu nghiên cứu	1
1.3 Phạm vi nghiên cứu	2
1.4 Công cụ nghiên cứu	2
1.5 Cấu trúc luận văn	3
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	5
2.1 Giới thiệu về trí tuệ nhân tạo.....	5
2.1.1 Trí tuệ nhân tạo	5
2.1.2 Học máy.....	6
2.1.3 Học sâu.....	7
2.2 Quy trình triển khai học máy có giám sát	7
2.2.1 Thu thập và xử lý dữ liệu	8
2.2.2 Lựa chọn mô hình	8
2.2.3 Đánh giá và tối ưu hóa mô hình	9
2.2.4 Triển khai mô hình	9
2.2.5 Giám sát và bảo trì mô hình	9
2.3 Yolo	10
2.3.1 Yolo.....	10
2.3.2 Yolo11n.....	11
2.4 Thư viện chính trong phần mềm.....	12
2.4.1 OpenCv.....	13
2.4.2 Arduino.....	13
2.4.3 Tkinter	14

2.4.4 Pandas	14
2.5 Các chỉ số đánh giá mô hình	15
2.5.1 Precision	16
2.5.2 Recall	16
2.5.3 F1- Score	16
2.6 Công cụ quản lý chất lượng	16
2.6.1 Biểu đồ Pareto.....	16
2.6.2 5Why.....	17
2.7 Đánh giá hiệu quả tài chính	18
2.7.1 Dòng tiền	18
2.7.2 Lợi tức đầu tư (ROI).....	18
CHƯƠNG 3: TỔNG QUAN VỀ CÔNG TY.....	20
3.1 Giới thiệu công ty	20
3.1.1 Lĩnh vực hoạt động.....	20
3.1.2 Quy mô sản xuất.	21
3.2 Cơ cấu tổ chức sản xuất.	22
3.2.1 Bộ phận sản xuất cần câu nhà máy Rod.....	22
3.2.2 Bộ phận sản xuất guồng quay cước nhà máy Reel.....	23
3.2.3 Bộ phận văn phòng.....	24
3.2.4 Bộ phận Quality Assurance QA	25
3.3 Phân tích hiện trạng tại bộ phận V159	25
3.4 Thống kê các dạng sai lỗi trong quá trình lắp ráp O.Slider	26
3.5 Phân tích chi phí cho thực trạng hiện tại.	27
3.5.1 Chi phí sửa chữa.....	28
3.5.2 Chi phí hư hỏng.....	28
3.6 Phân tích nguyên nhân	29
CHƯƠNG 4: XÂY DỰNG HỆ THỐNG GIÁM SÁT TỰ ĐỘNG CHO QUY TRÌNH LẮP RÁP SẢN PHẨM	31
4.1 Thu thập dữ liệu.....	32
4.1.1 Xác định quy trình lắp O.Slider	32
4.1.2 Xác định các thao tác chính.....	34
4.1.3 Thu thập dữ liệu hình ảnh	34
4.2 Xử lý dữ liệu	37

4.2.1 Tải dữ liệu	38
4.2.2 Chọn loại nhãn	38
4.2.3 Dán nhãn.....	38
4.2.4 Kiểm tra và xuất file.....	40
4.3 Huấn luyện mô hình	42
4.4 Đánh giá mô hình.....	46
4.4.1 Đánh giá Model theo Precision, Recall và F1.....	46
4.4.2 Phân tích F1-Confidence Curve	47
4.4.3 Phân tích Recall-Confidence Curve	48
4.4.4 Phân tích Precision-Confidence Curve	49
4.4.5 Phân tích Precision-Recall Curve.....	50
4.5 Đánh giá trên tập kiểm thử (Test set)	50
4.5.1 Tập dữ liệu kiểm thử (Test set)	50
4.5.2 Kết quả đánh giá	51
4.6 Kiến trúc tổng thể	52
4.7 Thuật toán của phần mềm	54
4.7.1 Khởi tạo chương trình:.....	55
4.7.2 Thiết lập giao diện và nút bấm.....	55
4.7.3 Tải mô hình và bắt đầu nhận diện.....	56
4.7.4 Xử lý logic chu trình:.....	56
4.7.5 Kết thúc nhận diện và lưu kết quả	57
4.7.6 Hiển thị kết quả:	57
4.8 Giao diện phần mềm.....	57
4.8.1 Chức năng của từng phần trong giao diện:	58
4.8.2 Cách sử dụng phần mềm.....	61
4.9 Kết quả thử nghiệm	64
4.9.1 Kết quả sản phẩm OK.....	65
4.9.2 Kết quả sản phẩm NG.....	68
CHƯƠNG 5: PHÂN TÍCH HIỆU QUẢ CỦA HỆ THỐNG	70
5.1 Chi phí đầu tư triển khai.....	70
5.1.1 Chi phí phần cứng	70
5.1.2 Chi phí phần mềm và đào tạo.....	70
5.1.3 Chi phí vận hành và bảo trì.....	71
5.1.4 Tổng chi phí đầu tư.....	71

5.2 Phân tích lợi ích	71
5.2.1 Hiệu quả trong giảm NG.....	72
5.2.2 Phân tích lợi ích tài chính.....	73
5.2.3 Phân tích dòng tiền.....	74
5.3 Giá trị tổ chức và vận hành	75
5.3.1 Nâng cao nhận thức và tác phong làm việc của công nhân.....	76
5.3.2 Hồ bộ phận QA trong công tác giám sát	76
5.3.3 Nâng cao hình ảnh doanh nghiệp	76
CHƯƠNG 6: KẾT LUẬN.....	77
6.1 Kết luận.....	77
6.2 Hạn chế	78
6.3 Đề xuất	78
TÀI LIỆU THAM KHẢO	80

DANH MỤC BẢNG

Bảng 3-1: Thống kê các dạng lỗi.....	26
Bảng 3-2: Thống kê các dạng lỗi của O.Slider AS 6 tháng cuối năm 2024.....	26
Bảng 3-3: Nguyên nhân chính gây ra lỗi lắp ráp thiếu linh kiện và sai vị trí.....	29
Bảng 4-1: Mô tả dataset cần thu thập cho các thao tác	35
Bảng 4-2: Kết quả thu được sau khi huấn luyện model.	44
Bảng 4-3: Thống kê kết quả đo được từ model	46
Bảng 4-4: Bảng kết quả của bộ dữ liệu test set	52
Bảng 4-5: Kết quả thống kê quá trình kiểm thử hệ thống	65
Bảng 5-1: Thống kê chi phí cho dự án phát triển hệ thống giám sát (đơn vị: VNĐ)....	71
Bảng 5-2: Thống kê trước và sau khi áp dụng hệ thống	72
Bảng 5-3: Phân tích dòng tiền trong 5 năm sau cải tiến (đơn vị: 1000 VNĐ)	75

DANH MỤC HÌNH ẢNH

Hình 2-1: Sơ đồ tổng quan về lịch sử hình thành trí tuệ nhân tạo.....	5
Hình 2-2: Mô tả quá cách thực học có giám sát.....	8
Hình 2-3: Sơ đồ kiến trúc mạng YOLO.....	11
Hình 2-4: Quan hệ giữa độ chính xác và độ trễ của các phiên bản YOLO.....	12
Hình 2-5: Confusion Matrix.....	15
Hình 3-1: Công ty TNHH Daiwa Việt Nam (DVN).....	20
Hình 3-2: Fishing Rod do Daiwa sản xuất.....	21
Hình 3-3: Fishing Reel do Daiwa sản xuất.....	21
Hình 3-4: Layout nhà máy Rod.....	23
Hình 3-5: Layout nhà máy REEL.....	24
Hình 3-6: Biểu đồ Pareto của các dạng lỗi.....	27
Hình 3-7: Biểu đồ chi phí mất mát do lỗi NG.....	28
Hình 4-1: Quy trình thực hiện xây dựng hệ thống.....	32
Hình 4-2: Quy trình lắp ráp O.Slider As.....	33
Hình 4-3: Quy trình dán nhãn.....	37
Hình 4-4: Giao diện thao tác chọn nhãn trong label-studio.....	39
Hình 4-5: Các loại nhãn trong bài toán nhận diện.....	39
Hình 4-6: Mô tả thao tác khoanh vùng trong Label Studio.....	40
Hình 4-7: Minh họa quá trình dán nhãn lắp ráp O.slider.....	41
Hình 4-8: Kết quả thu được sau khi dán nhãn.....	41
Hình 4-9: Đường dẫn của tập Dataset.....	42
Hình 4-10: Biểu đồ tương quan giữa F1-Confidence.....	47
Hình 4-11: Biểu đồ tương quan giữa Recall và Confidence.....	48
Hình 4-12: Biểu đồ tương quan giữa Precision và Confidence.....	49
Hình 4-13: Biểu đồ tương quan giữa Precision và Recall.....	50
Hình 4-14: Minh họa thuật toán để kiểm thử.....	51
Hình 4-15: Sơ đồ kiến trúc tổng thể.....	54
Hình 4-16: Giao diện phần mềm.....	58
Hình 4-17: Giao diện webcam của hệ thống.....	59
Hình 4-18: Giao diện lưu log của phần mềm giám sát.....	60
Hình 4-19: Giao diện hiển thị đồ thị.....	61
Hình 4-20: Quy trình thao tác phần mềm.....	62
Hình 4-21: Mô tả giao diện biểu đồ trong phần mềm.....	64

Hình 4-22: Minh họa hiển thị kết quả NG.....	68
Hình 5-1: Biểu đồ so sánh trước và sau khi áp dụng hệ thống.....	73

CHƯƠNG 1: GIỚI THIỆU

1.1 Lý do chọn đề tài

Trong bối cảnh yêu cầu ngày càng cao về chất lượng và tiến độ giao hàng trong ngành sản xuất, việc kiểm soát chất lượng không còn là một lựa chọn, mà là yếu tố bắt buộc để duy trì năng lực cạnh tranh. Tại Công ty TNHH DAIWA Việt Nam, bộ phận tiền lắp ráp V159 đóng vai trò quan trọng trong chuỗi sản xuất cần cầu, đặc biệt là công đoạn lắp ráp các chi tiết nhỏ có độ chính xác cao như O.slider. Tuy nhiên, thực trạng trong giai đoạn gần đây cho thấy vẫn còn nhiều hạn chế trong việc kiểm soát lỗi ngay tại công đoạn thao tác.

Theo số liệu thống kê nội bộ, chỉ riêng trong 6 tháng cuối năm 2024, tại bộ phận V159 đã ghi nhận 1305 lỗi phát sinh từ công đoạn lắp ráp O.slider. Đây là một con số đáng báo động, phản ánh sự thiếu ổn định trong quá trình thao tác và chưa có cơ chế giám sát kịp thời tại thời điểm thực hiện. Đáng chú ý, hầu hết các lỗi này không được phát hiện ngay tại chỗ, mà chỉ được nhận biết ở các công đoạn sau hoặc qua kiểm tra cuối cùng. Hệ quả là sản phẩm lỗi không những gây gián đoạn chu trình sản xuất mà còn làm tăng chi phí xử lý lại, tốn thời gian và ảnh hưởng đến tiến độ giao hàng. Trong khi đó, phương pháp kiểm tra hiện tại vẫn chủ yếu dựa vào sự giám sát thủ công của con người, vốn tiềm ẩn nhiều hạn chế như thiếu chính xác, dễ bỏ sót và không thể theo dõi liên tục theo thời gian thực.

Trước yêu cầu cấp thiết đó, việc ứng dụng các giải pháp công nghệ như thị giác máy tính, trí tuệ nhân tạo và tự động hóa để xây dựng một hệ thống giám sát thông minh là hướng đi phù hợp. Hệ thống này không chỉ hỗ trợ phát hiện kịp thời các sai lệch trong thao tác mà còn giúp doanh nghiệp ghi nhận và phân tích dữ liệu phục vụ cải tiến quy trình sản xuất lâu dài.

Đề tài “Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam” được thực hiện với mục tiêu hiện đại hóa công tác giám sát tại bộ phận tiền lắp ráp thông qua việc tích hợp công nghệ nhận diện hình ảnh và mô hình học máy. Qua đó, góp phần giảm thiểu sai sót trong thao tác, nâng cao chất lượng sản phẩm và tối ưu hiệu quả vận hành của dây chuyền sản xuất.

1.2 Mục tiêu nghiên cứu

Đề tài tập trung vào việc phát triển hệ thống giám sát tự động nhằm hỗ trợ kiểm soát chất lượng trong công đoạn lắp ráp O.Slider tại bộ phận V159 – Công ty TNHH Daiwa Việt Nam. Cụ thể, hệ thống được xây dựng để đáp ứng các tính năng chính sau:

- Nhận diện thao tác lắp ráp theo thời gian thực:
- Phát hiện lỗi lắp ráp: Tìm ra các lỗi trong quá trình lắp ráp và cảnh báo sớm cho công nhân
- Ghi nhận và lưu trữ dữ liệu: Lưu lại thông tin về các chu trình lắp ráp, số lượng lỗi phát sinh nhằm phục vụ phân tích, đánh giá hiệu quả sản xuất.
- Cảnh báo tức thời cho công nhân vận hành: Thông qua thiết bị Arduino điều khiển còi báo động và đèn tín hiệu khi phát hiện lỗi.
- Hiển thị thông tin giám sát trên giao diện người dùng: Phần mềm giao diện thân thiện được phát triển bằng Python, giúp công nhân và quản lý dễ dàng theo dõi quá trình vận hành.

1.3 Phạm vi nghiên cứu

Đề tài tập trung nghiên cứu và xây dựng hệ thống giám sát tự động dành riêng cho công đoạn lắp ráp O.Slider tại bộ phận V159 – Công ty TNHH Daiwa Việt Nam.

Phạm vi nghiên cứu cụ thể bao gồm:

- Nghiên cứu thực trạng phát sinh lỗi trong quá trình lắp ráp O.Slider.
- Thu thập dữ liệu hình ảnh thực tế tại khu vực lắp ráp phục vụ cho việc xây dựng bộ dữ liệu huấn luyện mô hình nhận diện thao tác.
- Ứng dụng mô hình học sâu yolo11 kết hợp với công nghệ thị giác máy tính để phát hiện thao tác lắp ráp sai trong thời gian thực.
- Xây dựng phần mềm giám sát hoàn chỉnh bao gồm chức năng nhận diện, lưu trữ dữ liệu, cảnh báo tức thời và hiển thị giao diện giám sát cho người vận hành.
- Đánh giá, phân tích hiệu quả kinh tế, kỹ thuật và phi tài chính của hệ thống sau khi áp dụng thực tế vào dây chuyền sản xuất.

1.4 Công cụ nghiên cứu

Để triển khai đề tài, việc lựa chọn công cụ nghiên cứu phù hợp đóng vai trò quan trọng nhằm đảm bảo tính khả thi và hiệu quả khi áp dụng vào môi trường sản xuất thực tế tại công ty. Cụ thể các công cụ được sử dụng trong nghiên cứu được trình bày như sau:

- Label Studio: Là công cụ hỗ trợ dán nhãn dữ liệu hình ảnh thu thập từ xưởng sản xuất. Việc tạo nhãn chính xác là yếu tố đầu vào quan trọng cho quá trình huấn luyện mô hình học sâu, giúp hệ thống nhận diện chính xác các thao tác lắp ráp.
- Google Colab: Nền tảng tính toán đám mây được sử dụng để huấn luyện mô hình Yolo11 dựa trên dữ liệu đã được dán nhãn. Đây là công cụ thuận tiện trong xử lý các tác vụ học máy đòi hỏi tài nguyên tính toán cao.

- Python: Ngôn ngữ lập trình chính cho toàn bộ hệ thống phần mềm, bao gồm xây dựng mô hình nhận diện, xử lý hình ảnh, điều khiển thiết bị ngoại vi và thiết kế giao diện người dùng.
- Các thư viện hỗ trợ phát triển hệ thống như OpenCV (xử lý ảnh số), Tkinter (giao diện người dùng), cùng với nền tảng Arduino (giao tiếp phần cứng cảnh báo) được kết hợp đồng bộ nhằm đảm bảo khả năng nhận diện, hiển thị và cảnh báo thời gian thực trong suốt quá trình lắp ráp sản phẩm.
- Excel: Công cụ hỗ trợ xử lý, phân tích và lưu trữ dữ liệu thu thập được trong quá trình thử nghiệm mô hình.

Việc phối hợp linh hoạt giữa các công cụ trên giúp đảm bảo tính đồng bộ, chính xác trong toàn bộ quy trình nghiên cứu từ khâu thu thập dữ liệu, huấn luyện mô hình, triển khai hệ thống, cho đến giai đoạn đánh giá và phân tích hiệu quả.

1.5 Cấu trúc luận văn

Để tiếp cận vấn đề một cách khoa học và toàn diện, luận văn được bố cục thành 6 chương, lần lượt đi từ nền tảng lý thuyết đến xây dựng hệ thống và đánh giá hiệu quả thực tế.

Chương 1 (Giới thiệu): Trình bày bối cảnh, lý do chọn đề tài từ thực trạng lỗi phát sinh tại công đoạn lắp ráp O.Slider của Công ty Daiwa Việt Nam. Xác định mục tiêu nghiên cứu, phạm vi, đối tượng và các công cụ sử dụng. Đây là cơ sở định hướng cho toàn bộ nội dung nghiên cứu ở các chương tiếp theo.

Chương 2 (Cơ sở lý thuyết): Trình bày các kiến thức nền tảng phục vụ nghiên cứu: trí tuệ nhân tạo, học máy, học sâu, phương pháp học có giám sát. Phân tích chi tiết về Yolo11n - mô hình nhận diện ảnh được áp dụng. Giới thiệu các công cụ lập trình (Python, OpenCV, Arduino...), phương pháp đánh giá mô hình (Precision, Recall, F1-score), công cụ quản lý chất lượng (Pareto, 5Why) và các chỉ tiêu phân tích hiệu quả tài chính (dòng tiền, ROI).

Chương 3 (Tổng quan về công ty): Giới thiệu tổng quan về Công ty Daiwa Việt Nam: lịch sử, quy mô, tổ chức sản xuất. Trọng tâm là bộ phận V159 - nơi xảy ra nhiều lỗi lắp ráp. Chương cũng phân tích dữ liệu lỗi, nguyên nhân và chi phí liên quan, làm cơ sở thực tiễn cho việc xây dựng hệ thống giám sát.

Chương 4 (Xây dựng hệ thống giám sát tự động cho quy trình lắp ráp sản phẩm): Mô tả quy trình xây dựng hệ thống: thu thập, xử lý dữ liệu, dán nhãn, huấn luyện mô hình nhận diện bằng Yolo11n. Trình bày kiến trúc tổng thể, thuật toán phần mềm, cách

vận hành giao diện, cơ chế cảnh báo phần cứng Arduino. Hệ thống giám sát lỗi thao tác thời gian thực.

Chương 5 (Phân tích hiệu quả của hệ thống): Phân tích hiệu quả tài chính thông qua chi phí đầu tư, khả năng giảm lỗi NG, tiết kiệm chi phí sửa lỗi, tính toán thời gian hoàn vốn và ROI. Đồng thời đánh giá các lợi ích phi tài chính khác.

Chương 6 (Kết luận): Tóm tắt kết quả đạt được, khẳng định giá trị thực tiễn của hệ thống giám sát tự động. Nêu rõ các hạn chế và đồng thời đề xuất hướng nghiên cứu mở rộng sang các công đoạn khác và ứng dụng AI tiên tiến hơn.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

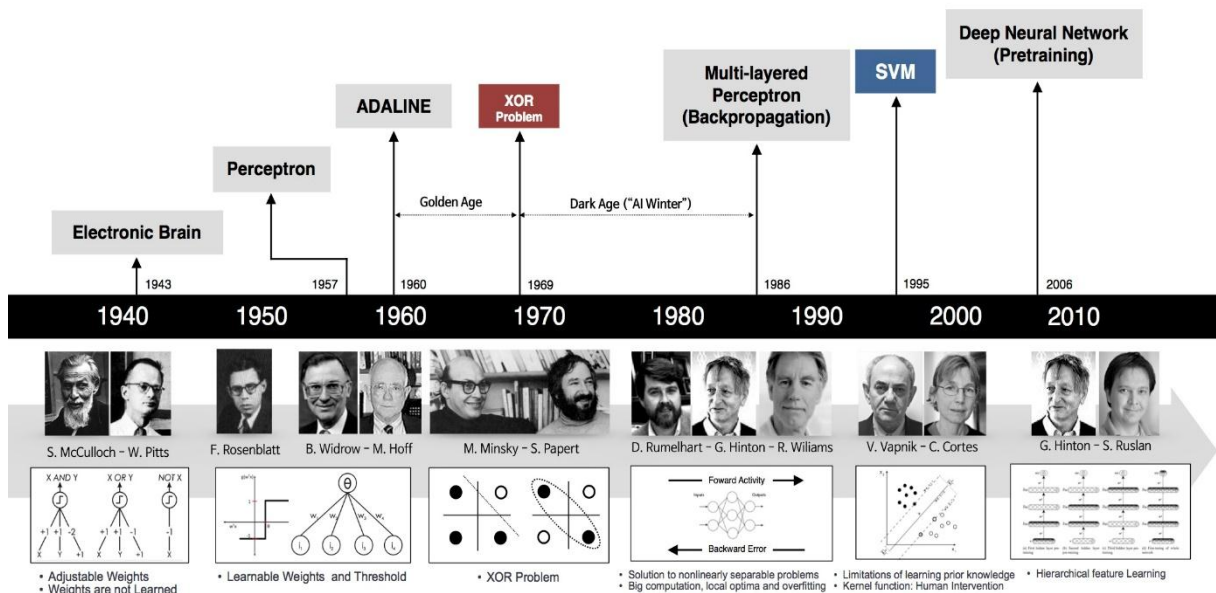
Trong chương này, đồ án sẽ trình bày nội dung chi tiết về cách tiếp cận bài toán sử dụng phương pháp cổ điển và cách áp dụng mô hình học sâu vào bài toán nhận diện thao tác, từ đó xây dựng hệ thống giám sát thao tác trong lắp ráp qua các phần:

- Giới thiệu về trí tuệ nhân tạo
- Quá trình triển khai học máy có giám sát
- Giới thiệu về YOLO
- Thư viện chính trong phần mềm
- Các chỉ số đánh giá
- Công cụ quản lý chất lượng
- Đánh giá hiệu quả tài chính

2.1 Giới thiệu về trí tuệ nhân tạo

2.1.1 Trí tuệ nhân tạo

Trí tuệ nhân tạo (AI) [1] là một nhánh quan trọng của khoa học máy tính, tập trung vào việc phát triển những hệ thống có khả năng mô phỏng trí thông minh con người. Những hệ thống này có thể học hỏi từ dữ liệu, thích nghi với các tình huống mới, và thực hiện những nhiệm vụ phức tạp như nhận diện ngôn ngữ, phân tích hình ảnh đưa ra quyết định tối ưu.



Hình 2-1: Sơ đồ tổng quan về lịch sử hình thành trí tuệ nhân tạo

Lịch sử hình thành AI bắt nguồn từ giữa thế kỷ XX, với những nghiên cứu tiên phong về mạng nơ-ron và thuật toán học máy. Tuy nhiên, phải đến những năm gần đây, cùng với sự bùng nổ của dữ liệu lớn (Big Data) và sức mạnh điện toán, AI mới đạt được Sinh viên thực hiện: Mai văn Thuận

những bước tiến vượt bậc. Ngày nay, AI không chỉ giới hạn trong phòng thí nghiệm mà đã len lỏi vào đời sống, từ những chatbot hỗ trợ khách hàng đến hệ thống chẩn đoán y khoa chính xác cao.

Ứng dụng của trí tuệ nhân tạo trải rộng trên nhiều ngành nghề, bao gồm: Y tế, giao thông, giáo dục, tài chính. Với tốc độ phát triển nhanh chóng, AI đang dần trở thành công nghệ nền tảng, định hình tương lai của xã hội hiện đại.

2.1.2 Học máy

Học máy [2](Machine Learning - ML) là một lĩnh vực quan trọng của trí tuệ nhân tạo, giúp máy tính có khả năng tự học từ dữ liệu mà không cần lập trình cụ thể. Theo Arthur Samuel, ML cho phép máy tính tự cải thiện hiệu suất thông qua kinh nghiệm từ dữ liệu đầu vào. Tom Mitchell bổ sung định nghĩa này bằng cách nhấn mạnh ba yếu tố cốt lõi: kinh nghiệm (E), nhiệm vụ (T) và thước đo hiệu suất (P). Về bản chất, ML không chỉ đơn thuần là áp dụng các quy tắc có sẵn mà là quá trình khám phá các mẫu hình ẩn trong dữ liệu, từ đó đưa ra dự đoán chính xác cho các tình huống mới. Các phương pháp ML phổ biến bao gồm học có giám sát (như phân loại dữ liệu), học không giám sát (như phân cụm) và học tăng cường (như huấn luyện robot). Ngày nay, ML được ứng dụng rộng rãi trong nhiều lĩnh vực như y tế (chẩn đoán bệnh), tài chính (đánh giá rủi ro) và công nghệ (xe tự lái), mang lại những giải pháp thông minh và hiệu quả cho các vấn đề phức tạp.

Có bốn loại chính của học máy:

1. Học máy giám sát (Supervised Learning): Trong loại này, mô hình được huấn luyện trên một tập dữ liệu có nhãn, nghĩa là mỗi mẫu dữ liệu đã được gán nhãn với đầu ra mong muốn. Mục tiêu là học cách ánh xạ từ dữ liệu đầu vào đến các nhãn đầu ra để sau đó có thể tự động dự đoán đầu ra cho các mẫu mới mà không cần nhãn.
2. Học máy không giám sát (Unsupervised Learning): Trong loại này, mô hình được huấn luyện trên một tập dữ liệu không có nhãn. Mục tiêu chính của học máy không giám sát là tìm ra cấu trúc, mối quan hệ, hoặc phân nhóm dữ liệu trên toàn bộ tập dữ liệu mà không cần biết đầu ra mong muốn trước.
3. Học máy bán giám sát (Semi-supervised Learning): Là một hình thức trung gian giữa học máy giám sát và học máy không giám sát, loại này sử dụng một tập dữ liệu lớn không có nhãn và một số ít dữ liệu có nhãn để huấn luyện mô hình.
4. Học tăng cường (Reinforcement Learning): Là một hình thức mà máy tính được thiết kế để tìm hiểu và cải thiện hiệu suất của mô hình thông qua trải nghiệm và tương tác với môi trường. Mục tiêu của mô hình là tối ưu hóa giá trị nhận được (reward) tại

mỗi hành động mà mô hình thực hiện trong một môi trường nhất định nào đó. Khác với học máy truyền thống, học tăng cường không chỉ tập trung vào việc giải quyết các vấn đề cụ thể, mà còn đề cao khả năng học và ra quyết định tối ưu trong môi trường động và không chắc chắn.

Trong suốt thời gian phát triển, học máy đã đạt được nhiều thành công trong nhiều ứng dụng thực tế như nhận dạng hình ảnh, xử lý ngôn ngữ tự nhiên, dự đoán chuỗi thời gian, v.v. Nói cách khác, học máy đã làm thay đổi cách chúng ta xử lý thông tin và ra quyết định, mang lại sự tự động hóa và hiệu suất đáng kể. Tuy nhiên, logic tuyến tính giữa đầu vào và đầu ra trong học máy là cản trở để phương pháp này hoạt động tốt với các loại dữ liệu phức tạp, đòi hỏi tính phi tuyến.

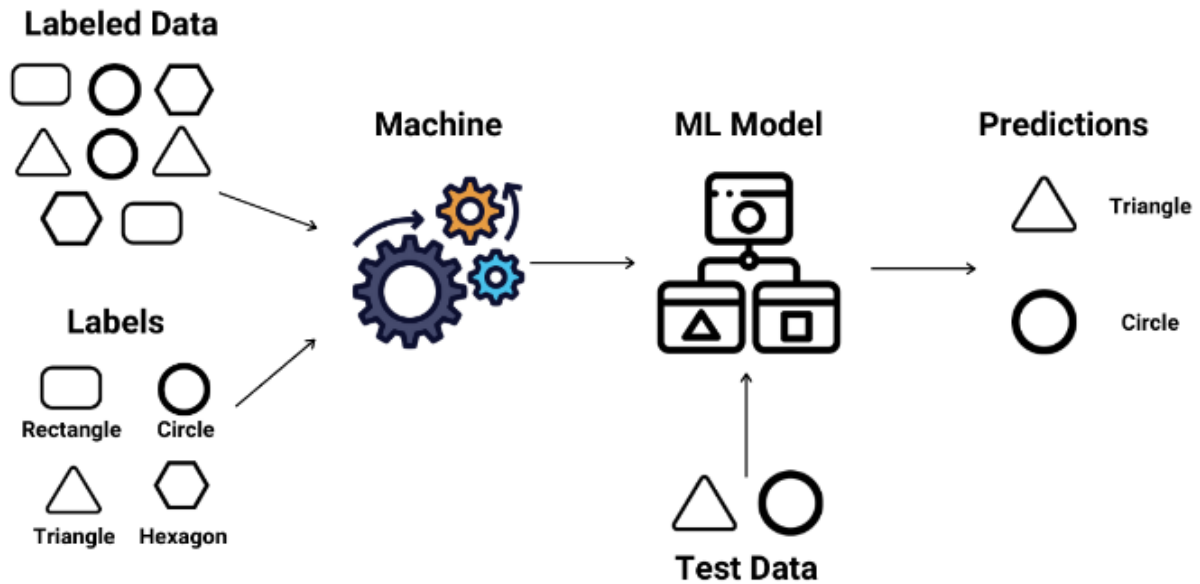
2.1.3 Học sâu.

Học sâu (Deep Learning - DL) [3] là một tập con của học máy trong lĩnh vực trí tuệ nhân tạo. Học sâu tập trung vào phát triển và huấn luyện các mô hình máy học sâu có khả năng tự động học và thực hiện các nhiệm vụ phức tạp có tính chất phi tuyến. Các mô hình này, hay còn được gọi là mạng nơ-ron nhân tạo, được thiết kế với nhiều lớp thần kinh như bộ não con người. Qua quá trình huấn luyện, mạng nơ-ron dần đạt được kinh nghiệm thông qua hàm đánh giá, có khả năng lưu giữ kinh nghiệm học được và sử dụng để thực hiện dự đoán cho các dữ liệu chưa biết sau này

2.2 Quy trình triển khai học máy có giám sát

Trong nghiên cứu này, phương pháp học máy có giám sát được ứng dụng nhằm xây dựng mô hình có khả năng dự đoán kết quả đầu ra dựa trên dữ liệu đầu vào đã được gán nhãn. Bằng cách huấn luyện mô hình trên tập dữ liệu mẫu, trong đó mỗi phần tử đều có thông tin đầu vào và nhãn tương ứng, hệ thống có thể học được mối quan hệ giữa các đặc trưng và kết quả, từ đó đưa ra dự đoán chính xác cho các dữ liệu mới chưa từng gặp.

Supervised Learning



Hình 2-2: Mô tả quá trình học có giám sát

Quy trình thực hiện được mô tả 5 bước sau:

2.2.1 Thu thập và xử lý dữ liệu

Dữ liệu đóng vai trò then chốt trong học có giám sát, vì chất lượng dữ liệu quyết định trực tiếp đến chất lượng mô hình. Việc xử lý dữ liệu bao gồm:

Thu thập dữ liệu: Dữ liệu cần được lấy từ các nguồn đáng tin cậy, đảm bảo mỗi mẫu có gán nhãn đúng. Dữ liệu có thể ở dạng số, văn bản, ảnh hoặc tín hiệu.

Làm sạch dữ liệu: Xử lý giá trị thiếu, loại bỏ nhiễu, trùng lặp hoặc dữ liệu sai lệch.

Chuyển đổi và chuẩn hóa:

- Mã hóa các biến phân loại (label encoding hoặc one-hot encoding).
- Chuẩn hóa dữ liệu về cùng một thang đo (normalization, standardization).

Chia tập dữ liệu: Dữ liệu được phân chia thành:

- Tập huấn luyện (training set): 70-80%.
- Tập kiểm tra (test set): 20-30%.

2.2.2 Lựa chọn mô hình

Sau khi có dữ liệu đã xử lý, bước tiếp theo là lựa chọn mô hình học máy phù hợp:

Lựa chọn thuật toán:

- Bài toán phân loại: sử dụng các thuật toán như Logistic Regression, SVM, KNN, Decision Tree, Random Forest, v.v.
- Bài toán hồi quy: áp dụng Linear Regression, Ridge, Lasso, SVR, v.v.

Khởi tạo mô hình: Thiết lập các siêu tham số ban đầu phù hợp với dữ liệu và yêu cầu bài toán.

Huấn luyện mô hình:

- Mô hình học từ tập huấn luyện thông qua việc tối ưu hàm mất mát.
- Cập nhật trọng số và tham số để mô hình dần cải thiện độ chính xác.

2.2.3 Đánh giá và tối ưu hóa mô hình

Đánh giá mô hình là bước quan trọng để kiểm tra khả năng tổng quát hóa của mô hình trên dữ liệu chưa từng thấy.

Đánh giá bằng tập kiểm tra:

- Với bài toán phân loại: sử dụng các chỉ số như Accuracy, Precision, Recall, F1-score.
- Với bài toán hồi quy: dùng MAE, MSE, RMSE, R^2 .

Tối ưu hóa mô hình:

- Thử nghiệm và điều chỉnh siêu tham số (Grid Search, Random Search, Bayesian Optimization).
- Áp dụng kỹ thuật giảm overfitting như Regularization, Dropout hoặc Early Stopping.

Phân tích lỗi: Sử dụng ma trận nhầm lẫn (confusion matrix) và kiểm tra các trường hợp dự đoán sai để cải thiện mô hình.

2.2.4 Triển khai mô hình

Khi mô hình đạt độ chính xác mong muốn, nó được đưa vào sử dụng thực tế:

Lưu mô hình: Sử dụng các định dạng như .pkl, .joblib, .h5 hoặc .onnx.

Tích hợp hệ thống:

- Đưa mô hình vào hệ thống phần mềm, dịch vụ, hoặc tạo API để người dùng truy cập.
- Đảm bảo mô hình hoạt động ổn định, thời gian phản hồi nhanh và hiệu suất phù hợp với yêu cầu thực tế.

2.2.5 Giám sát và bảo trì mô hình

Trong môi trường thực tế, mô hình cần được theo dõi liên tục để đảm bảo hiệu quả duy trì theo thời gian:

Giám sát hiệu suất: Phát hiện tượng suy giảm do dữ liệu thay đổi (data drift, concept drift).

Cập nhật mô hình định kỳ: Tái huấn luyện với dữ liệu mới để mô hình luôn chính xác.

Lưu trữ toàn bộ quy trình: Ghi lại cấu hình mô hình, tập dữ liệu, kết quả huấn luyện nhằm phục vụ cho tái tạo hoặc bảo trì trong tương lai.

2.3 Yolo

2.3.1 Yolo

YOLO [4] (You Only Look Once) là một trong những phương pháp nhận diện và phát hiện đối tượng (object detection) tiên tiến, được giới thiệu lần đầu bởi Joseph Redmon và các cộng sự vào năm 2016. Điểm nổi bật của YOLO là khả năng thực hiện phát hiện đối tượng trong thời gian thực (real-time) với tốc độ cao và độ chính xác đáng kể. YOLO khác biệt so với các phương pháp truyền thống như R-CNN (Region-based Convolutional Neural Networks) ở chỗ nó chỉ sử dụng một mạng nơ-ron duy nhất để dự đoán cả vị trí (bounding boxes) và lớp đối tượng trong một lần forward pass, thay vì chia quá trình thành nhiều giai đoạn.

Lịch sử phát triển của YOLO:

YOLOv1 (2015): Mô hình phát hiện đối tượng thời gian thực đầu tiên, do Joseph Redmon và Ali Farhadi phát triển. Xử lý hình ảnh trong một lần truyền qua mạng, đạt tốc độ cao nhưng hạn chế với đối tượng nhỏ và độ chính xác thấp hơn các phiên bản sau.

YOLOv2 (2016): Cải tiến từ phiên bản gốc với chuẩn hóa theo lô, sử dụng hộp neo và cụm kích thước để nâng cao hiệu suất.

YOLOv3 (2018): Tăng cường hiệu quả bằng mạng xương sống tối ưu, nhiều mô neo hơn và kim tự tháp đặc trưng không gian.

YOLOv4 (2020): Giới thiệu tăng cường dữ liệu Mosaic, đầu phát hiện không cần neo và hàm mất mát cải tiến.

YOLOv5: Nâng cao hiệu suất với tối ưu hóa siêu tham số, tích hợp theo dõi thử nghiệm và hỗ trợ xuất sang nhiều định dạng phổ biến.

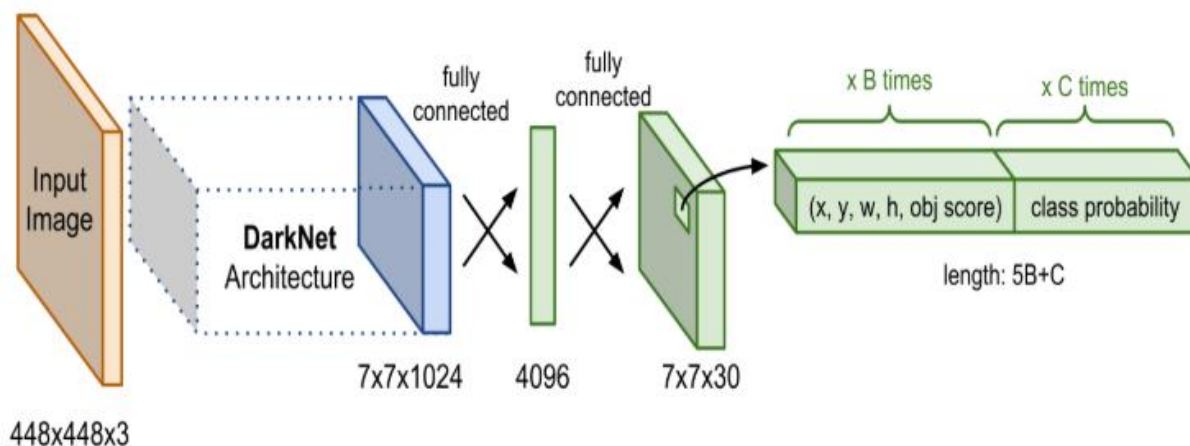
YOLOv6 (2022): Được Meituan mở mã nguồn, ứng dụng trong các robot giao hàng tự động. YOLOv7: Bổ sung tác vụ ước lượng tư thế trên tập dữ liệu COCO Keypoints.

YOLOv8 (2023): Do Ultralytics phát hành, mang đến các cải tiến về hiệu suất, tính linh hoạt và hỗ trợ đầy đủ các tác vụ thị giác AI.

YOLOv9: Tích hợp các phương pháp mới như Thông tin Gradient có thể lập trình (PGI) và Mạng tổng hợp lớp hiệu quả tổng quát (GELAN).

YOLOv10: Được phát triển bởi Đại học Thanh Hoa dựa trên gói Ultralytics Python, cải tiến phát hiện đối tượng thời gian thực bằng cách loại bỏ yêu cầu NMS.

YOLO11 (Mới nhất): Mô hình mới nhất của Ultralytics, đạt hiệu suất hàng đầu (SOTA) cho nhiều tác vụ như phát hiện, phân đoạn, ước lượng tư thế, theo dõi và phân loại, phù hợp với đa dạng ứng dụng AI.



Hình 2-3: Sơ đồ kiến trúc mạng YOLO

Nguyên lý cơ bản của YOLO:

Chia lưới (Grid Division): Hình ảnh đầu vào được chia thành một lưới (grid) kích thước $S \times S$ (ví dụ: 7×7 hoặc 13×13). Mỗi ô lưới chịu trách nhiệm dự đoán một số hộp giới hạn (bounding boxes) và xác suất thuộc về các lớp đối tượng.

Dự đoán đồng thời: Mỗi ô lưới dự đoán:

Tọa độ và kích thước của các bounding box (x, y, w, h).

Độ tin cậy (confidence score) của bounding box, biểu thị khả năng có đối tượng và độ chính xác của hộp.

Xác suất thuộc về từng lớp đối tượng (class probabilities).

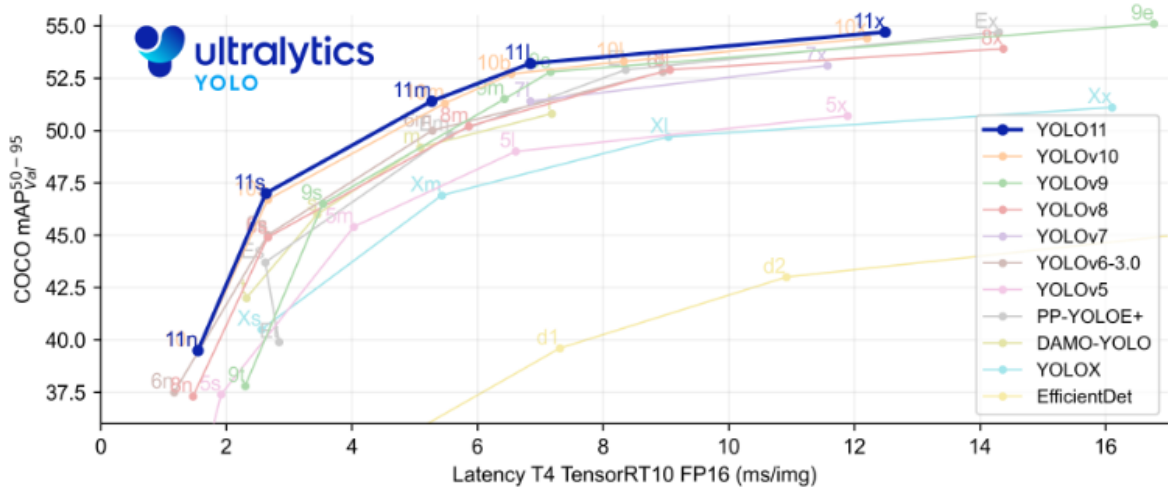
2.3.2 Yolo11n

YOLO11n (You Only Look Once version 11 Nano) là phiên bản nhẹ nhất trong dòng mô hình YOLO11, được phát triển và công bố bởi nhóm Ultralytics vào cuối năm 2024. Đây là một phần trong chuỗi cải tiến liên tục của YOLO (từ v1 đến v11) nhằm giải quyết các thách thức trong bài toán nhận diện đối tượng thời gian thực, đồng thời đảm bảo tốc độ suy luận nhanh, kích thước mô hình nhỏ và hiệu suất tương đối cao trên các thiết bị giới hạn phần cứng như Raspberry Pi, Jetson Nano, hoặc webcam trên dây chuyền sản xuất.

YOLO11n là phiên bản "Nano", tức là rút gọn và tối ưu kiến trúc để phục vụ các ứng dụng Edge AI, nơi hiệu năng xử lý và bộ nhớ là giới hạn quan trọng.

YOLO11n thừa hưởng các đặc điểm cốt lõi của dòng YOLO:

- Một bước duy nhất (single-shot): Mô hình phân chia ảnh thành các ô lưới (grid cells) và dự đoán bounding boxes, objectness và class trực tiếp trong một lần truyền (forward pass).
- Dự đoán toàn cục (global prediction): Mỗi ô lưới chịu trách nhiệm phát hiện đối tượng trong vùng bao quanh, giúp tăng tốc độ suy luận.
- Không cần vùng đề xuất (no region proposals): Khác với R-CNN hoặc Faster R-CNN, YOLO không cần giai đoạn sinh vùng nghi ngờ → nhanh hơn.



Hình 2-4: Quan hệ giữa độ chính xác và độ trễ của các phiên bản YOLO

YOLO11n (điểm 11n) có độ trễ khoảng 2-3ms và mAP khoảng 40-42%, phù hợp cho các ứng dụng cần tốc độ cao trên thiết bị hạn chế.

Hiệu suất và độ chính xác

mAP@0.5:95 trên tập COCO: Khoảng 39.5% Điều này có nghĩa là YOLOv11n đạt độ chính xác trung bình khá tốt khi phát hiện đối tượng với các ngưỡng IoU từ 0.5 đến 0.95.

mAP@0.5: Có thể đạt tới 54-55% (IoU=0.5), cho thấy khả năng phát hiện đối tượng với độ chính xác cao ở ngưỡng IoU tiêu chuẩn.

Tốc độ suy luận (Inference Latency): Khoảng 2.2ms mỗi hình ảnh trên GPU T4 với TensorRT FP16. Đây là tốc độ rất nhanh, phù hợp cho các ứng dụng thời gian thực như giám sát dây chuyền sản xuất.

2.4 Thư viện chính trong phần mềm

Trong quá trình phát triển hệ thống giám sát, việc lựa chọn các thư viện phần mềm phù hợp giữ vai trò then chốt, đảm bảo khả năng xử lý dữ liệu hiệu quả, duy trì tính ổn định khi vận hành và dễ dàng tích hợp với phần cứng ngoại vi. Mỗi thư viện được sử

dụng đều đóng góp những chức năng chuyên biệt, tạo nên sự đồng bộ giữa các thành phần mềm, giúp hệ thống vận hành chính xác và đáp ứng tốt các yêu cầu giám sát thời gian thực tại môi trường sản xuất. Cụ thể các thư viện chính sẽ được trình bày dưới đây.

2.4.1 OpenCv

OpenCv [5](Open Source Computer Vision Library) là một thư viện lập trình mã nguồn mở chuyên cung cấp các công cụ và thuật toán phục vụ cho xử lý ảnh số (image processing) và thị giác máy tính (computer vision). Được phát triển bởi Intel từ năm 1999, OpenCV có thể hoạt động trên nhiều hệ điều hành như Windows, Linux, macOS và hỗ trợ các ngôn ngữ như C++, Python, Java.

Thư viện này hỗ trợ các kỹ thuật quan trọng như:

- Biến đổi ảnh: làm mờ, làm sắc nét, chuyển sang ảnh thang xám, cân bằng histogram.
- Phân tích hình dạng: phát hiện cạnh, contour, phát hiện góc, segment ảnh.
- Nhận diện vật thể: khuôn mặt, biển số xe, tay người, chuyển động,...
- Xử lý video: đọc, ghi, hiển thị khung hình theo thời gian thực.
- Tích hợp thiết bị ngoại vi: như camera, webcam, giúp thu thập dữ liệu trực tiếp

Trong đề tài này, OpenCV đóng vai trò là cầu nối giữa hệ thống camera giám sát và phần mềm hiển thị. Thư viện đảm nhiệm việc:

- Mở và xử lý luồng hình ảnh video từ webcam theo thời gian thực.
- Hiển thị hình ảnh kèm bounding box từ mô hình nhận diện (YOLO).
- Giao tiếp với giao diện Tkinter để đồng bộ hóa hình ảnh và dữ liệu điều khiển.

2.4.2 Arduino

Arduino [6] là nền tảng phần cứng mã nguồn mở, sử dụng vi điều khiển (microcontroller) như ATmega328P để điều khiển các thiết bị điện tử thông qua lập trình. Arduino có các dòng phổ biến như Uno, Mega, Nano,... và được lập trình bằng ngôn ngữ C/C++ thông qua Arduino IDE.

Vi điều khiển Arduino có khả năng:

- Đọc tín hiệu từ cảm biến (sensor) như ánh sáng, nhiệt độ, lực, v.v.
- Điều khiển thiết bị đầu ra như LED, relay, còi, motor,...
- Giao tiếp với máy tính qua cổng USB (Serial).

Trong hệ thống giám sát sản xuất, Arduino được sử dụng để:

- Nhận tín hiệu cảnh báo từ phần mềm giám sát thông qua giao tiếp Serial.
- Điều khiển đèn báo hoặc còi để cảnh báo khi phát hiện sản phẩm lỗi.
- Đồng bộ với phần mềm theo thời gian thực để hỗ trợ công nhân thao tác

2.4.3 Tkinter

Tkinter [6] là thư viện giao diện đồ họa tiêu chuẩn (GUI toolkit) đi kèm với Python, được xây dựng dựa trên bộ công cụ Tk GUI toolkit của Tcl/Tk. Đây là công cụ đơn giản, nhẹ, dễ học, hỗ trợ tạo ra các cửa sổ, nút nhấn, nhãn văn bản, khung nhập liệu và các yếu tố tương tác khác một cách nhanh chóng.

Các thành phần cơ bản của Tkinter:

Widget (phần tử giao diện): gồm Label, Button, Entry, Canvas, Frame, Text, v.v.

Layout (bố cục): sử dụng các phương thức pack(), grid() hoặc place() để dàn layout.

Sự kiện (event): xử lý các hành vi như nhấn nút, di chuyển chuột, đóng cửa sổ.

Vòng lặp giao diện: root.mainloop() giữ cho ứng dụng hoạt động liên tục và cập nhật giao diện theo thời gian thực.

Trong đề tài này, Tkinter có các vai trò sau:

- Hiển thị video từ camera kết hợp với bounding box từ OpenCV.
- Cập nhật thông số như số lỗi NG, số chu trình, nhãn đang được nhận diện.
- Tích hợp các nút chức năng.

Giao diện này giúp người vận hành dễ dàng giám sát quy trình, đồng thời tăng tính tương tác giữa con người và hệ thống.

2.4.4 Pandas

Pandas [6] là một thư viện mã nguồn mở được phát triển cho ngôn ngữ lập trình Python, chuyên dùng để thao tác, phân tích và xử lý dữ liệu bảng (dạng bảng giống Excel hoặc SQL). Tên “Pandas” xuất phát từ cụm từ Panel Data, là thuật ngữ trong thống kê kinh tế, dùng để chỉ các tập dữ liệu đa chiều.

Pandas cung cấp hai cấu trúc dữ liệu chính:

Series: Một mảng một chiều (tương tự như một cột).

DataFrame: Một bảng hai chiều (gồm nhiều cột và nhiều dòng), tương tự như bảng dữ liệu trong Excel.

Trong đề tài này, Pandas cho phép:

- Nhập và xuất dữ liệu từ nhiều định dạng (CSV, Excel, SQL, JSON,...).
- Lọc, sắp xếp, nhóm, thống kê và xử lý dữ liệu.
- Tính toán nhanh trên cột/dòng với khả năng vector hóa cao.
- Ghi dữ liệu và phân tích dữ liệu lịch sử một cách khoa học.

2.5 Các chỉ số đánh giá mô hình

Trong các bài toán học máy có giám sát (Supervised Learning), đặc biệt là các bài toán phân loại và nhận diện đối tượng (Object Detection), việc đánh giá mô hình là bước không thể thiếu nhằm xác định hiệu quả hoạt động của hệ thống trên tập dữ liệu chưa từng thấy. Đặc biệt đối với bài toán trong đề tài này – phát hiện thao tác lắp ráp sai trong quá trình sản xuất – việc đo lường độ chính xác và khả năng phát hiện lỗi là tối quan trọng.

Ba chỉ số phổ biến và quan trọng nhất trong đánh giá hiệu suất mô hình là Precision (Độ chính xác), Recall (Độ bao phủ) và F1-score (Chỉ số tổng hợp) [7]. Các chỉ số này được tính dựa trên ma Confusion Matrix, bao gồm các thành phần:

True class (Edgels from the ground truth)	
Predicted class (Edgels from algorithm)	TP (True Positive)
	FP (False Positive)
	FN (False Negative)
	TN (True Negative)

Hình 2-5: Confusion Matrix

True Positive (TP): Dự đoán đúng dương tính – mô hình phát hiện đúng một đối tượng/thao tác là “có”.

False Positive (FP): Dự đoán sai dương tính – mô hình cho là “có” nhưng thực tế không có.

False Negative (FN): Dự đoán sai âm tính – mô hình cho là “không có” nhưng thực tế lại có.

True Negative (TN): Dự đoán đúng âm tính – mô hình dự đoán đúng rằng “không có”.

2.5.1 Precision

Precision phản ánh tỷ lệ giữa số lượng dự đoán dương tính đúng (TP) và tổng số lần mô hình dự đoán là dương tính (TP + FP).

$$Precision = \frac{TP}{TP + FP}$$

Precision càng cao thì tỷ lệ cảnh báo sai càng thấp, tức là hệ thống ít báo nhầm thao tác sai, điều này rất quan trọng trong các ứng dụng công nghiệp – nơi việc báo động sai có thể gây gián đoạn không cần thiết.

2.5.2 Recall

Recall đo lường khả năng của mô hình trong việc phát hiện đầy đủ các trường hợp dương tính thực sự. Nói cách khác, Recall cho biết trong số tất cả thao tác lỗi thật sự, mô hình đã phát hiện được bao nhiêu phần trăm.

$$Recall = \frac{TP}{TP + FN}$$

2.5.3 F1- Score

F1-score là trung bình điều hòa giữa Precision và Recall, nhằm cung cấp cái nhìn cân bằng giữa tính chính xác và khả năng bao phủ của mô hình. Chỉ số này đặc biệt hữu ích trong các tình huống mà việc bỏ sót lỗi (FN) hay báo nhầm lỗi (FP) đều gây hậu quả nghiêm trọng.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

F1-score có giá trị nằm trong khoảng từ 0 đến 1, với 1 là giá trị tốt nhất. Khi Precision và Recall đều cao, F1-score cũng sẽ cao. Ngược lại, nếu một trong hai chỉ số thấp, F1-score sẽ giảm theo.

2.6 Công cụ quản lý chất lượng

2.6.1 Biểu đồ Pareto

Biểu đồ Pareto [8] là một công cụ phân tích chất lượng dựa trên nguyên lý Pareto (80/20), được phát triển bởi nhà kinh tế học người Ý Vilfredo Pareto và sau đó được áp dụng trong quản lý chất lượng bởi Tiến sĩ Joseph Juran. Nguyên lý Pareto chỉ ra rằng khoảng 80% các vấn đề hoặc kết quả thường được gây ra bởi 20% nguyên nhân chính. Trong sản xuất, điều này có nghĩa là phần lớn các lỗi hoặc khuyết tật thường xuất phát từ một số ít nguyên nhân quan trọng.

Biểu đồ Pareto là sự kết hợp giữa biểu đồ cột và đường cong, trong đó:

Biểu đồ cột: Thể hiện tần suất hoặc mức độ ảnh hưởng của từng nguyên nhân (lỗi) theo thứ tự giảm dần.

Đường cong tích lũy: Hiện thị tỷ lệ phần trăm tích lũy của các nguyên nhân, giúp xác định các yếu tố chiếm phần lớn vấn đề (thường là 80%).

Quá trình xây dựng biểu đồ Pareto bao gồm các bước sau:

1. Thu thập dữ liệu: Ghi nhận và phân loại các lỗi xảy ra trong quy trình.
2. Sắp xếp dữ liệu: Sắp xếp các loại lỗi theo tần suất hoặc mức độ ảnh hưởng, từ cao đến thấp.
3. Tính tỷ lệ tích lũy: Xác định phần trăm đóng góp của từng loại lỗi và tính tổng tích lũy.
4. Vẽ biểu đồ:
 - Biểu đồ cột thể hiện tần suất của từng loại lỗi.
 - Đường cong tích lũy biểu thị tỷ lệ phần trăm tích lũy của các lỗi.
5. Phân tích: Xác định các lỗi chiếm khoảng 80% vấn đề để ưu tiên xử lý.

2.6.2 5Why

Phương pháp 5 Why [9] dựa trên triết lý Kaizen (cải tiến liên tục) và tư duy hệ thống của Toyota, nhấn mạnh việc giải quyết vấn đề một cách khoa học và có hệ thống. Nó xuất phát từ nhận thức rằng các vấn đề thường không chỉ có nguyên nhân bề mặt mà còn liên quan đến các yếu tố sâu hơn trong quy trình, con người, hoặc hệ thống. Bằng cách đặt câu hỏi "Tại sao?" nhiều lần, phương pháp này giúp bóc tách các lớp nguyên nhân để tìm ra bản chất thực sự của vấn đề.

Nguyên tắc cơ bản của phương pháp:

- Tư duy phân tích nguyên nhân gốc rễ: 5 Why tập trung vào việc đi sâu vào nguyên nhân thực sự thay vì chỉ dừng lại ở các triệu chứng bề mặt. Ví dụ, nếu một máy móc hỏng, nguyên nhân bề mặt có thể là "máy bị lỗi", nhưng nguyên nhân gốc rễ có thể là "bảo trì không đúng lịch trình".
- Đơn giản và trực quan: Phương pháp này không yêu cầu công cụ phức tạp hay kiến thức chuyên sâu, chỉ cần tư duy logic và sự kiên nhẫn để đặt câu hỏi đúng.
- Số lần "Tại sao" không cố định: Mặc dù được gọi là "5 Why", số lần hỏi có thể ít hơn hoặc nhiều hơn 5, tùy thuộc vào độ phức tạp của vấn đề. Mục tiêu là tìm ra nguyên nhân gốc rễ, không phải chỉ dừng lại ở con số 5.

Quy trình thực hiện:

1. Xác định vấn đề: Mô tả rõ ràng vấn đề đang xảy ra,

2. Hỏi "Tại sao?" lần đầu tiên: Xác định nguyên nhân trực tiếp của vấn đề.
3. Lặp lại câu hỏi "Tại sao?": Với mỗi câu trả lời, tiếp tục hỏi "Tại sao?" để đào sâu hơn.
4. Xác định nguyên nhân gốc rễ: Khi không thể hỏi "Tại sao?" thêm nữa hoặc khi câu trả lời chỉ ra một nguyên nhân có thể giải quyết được, đó là nguyên nhân gốc rễ.
5. Đề xuất giải pháp: Dựa trên nguyên nhân gốc rễ, đưa ra các biện pháp khắc phục để ngăn vấn đề tái diễn.

2.7 Đánh giá hiệu quả tài chính

2.7.1 Dòng tiền

Khái niệm: Dòng tiền [10] là số tài chính phản ánh lượng tiền mặt chuyển vào và ra khỏi một dự án, doanh nghiệp hoặc tài khoản đầu tư trong một khoảng thời gian nhất định. Dòng tiền bao gồm các tài khoản thu (lợi ích tài chính) và các tài khoản chi (chi phí đầu tư, vận hành, bảo trì). Phân tích dòng tiền giúp đánh giá khả năng sinh lời, thời gian hoàn vốn và tính khả thi của một dự án.

Cơ sở lý thuyết: Dòng tiền được chia thành dòng tiền vào (dòng tiền vào, ví dụ: doanh thu, lợi ích tài chính từ giảm chi phí) và dòng tiền ra (dòng tiền ra, ví dụ: chi phí đầu tư, chi phí vận hành hành).

Dòng tiền ròng (Net Cash Flow) được tính bằng công thức:

$$\text{Dòng tiền ròng} = \text{Dòng tiền vào} - \text{Chi phí đầu tư}$$

Dòng tiền lũy kế (Dòng tiền tích lũy) là tổng dòng tiền ròng cộng dồn qua các kỳ, được sử dụng để xác định thời gian hoàn vốn:

$$\text{Dòng tiền tích lũy tại năm } N = \sum_{i=1}^N \text{Dòng tiền năm thứ } i$$

Trong dự án này, dòng tiền ròng được tính dựa trên lợi ích tài chính từ việc giảm sản phẩm lỗi (NG) trừ đi chi phí phát triển và chi phí duy trì hàng năm. Dòng tiền tăng tốc giúp xác định thời điểm dự án thu hồi vốn (khi dòng tiền tăng tốc chuyển từ âm sang dương).

2.7.2 Lợi tức đầu tư (ROI)

Khái niệm: ROI là số tài chính chính đo lường tỷ lệ lợi nhuận thu được với chi phí đầu tư của một dự án. Nó phản ánh hiệu quả sử dụng vốn đầu tư và được sử dụng rộng rãi để so sánh tính hấp dẫn của các tài khoản đầu tư khác nhau.

Công thức:

$$\text{Lợi tức đầu tư} = \frac{\text{Tổng lợi nhuận ròng}}{\text{Tổng chi phí}} * 100\%$$

Trong đó:

Tổng lợi nhuận ròng = Tổng dòng tiền ròng trong thời gian phân tích trong suốt (tỷ lợi ích tài chính trừ tổng chi phí).

Tổng chi phí đầu tư = Tổng chi phí phát triển và duy trì trong toàn bộ dự án.

ROI được tính toán dựa trên nguyên tắc so sánh lợi ích tài chính được tính bằng số vốn đầu tư ban đầu và chi phí duy trì.

CHƯƠNG 3: TỔNG QUAN VỀ CÔNG TY

3.1 Giới thiệu công ty

Công ty TNHH Daiwa Việt Nam là một doanh nghiệp có vốn đầu tư 100% từ Nhật Bản, trực thuộc tập đoàn Globeride. Với bề dày hoạt động hơn 17 năm tại thị trường Việt Nam, công ty đã khẳng định vị thế là một trong những đơn vị chuyên sản xuất cần câu hàng đầu tại Đà Nẵng, đồng thời duy trì đà tăng trưởng ổn định và bền vững. Theo số liệu thống kê, chỉ trong 9 tháng đầu năm 2020, doanh thu của công ty đã vượt mốc 1.300 tỷ đồng, trong đó hoạt động xuất nhập khẩu đóng góp hơn 36 triệu USD. Bên cạnh đó, tổng số tiền nộp ngân sách nhà nước đạt hơn 7,1 tỷ đồng, phản ánh hiệu quả hoạt động tài chính tích cực.

Nhờ môi trường làm việc năng động và hội nhập sâu rộng, Daiwa Việt Nam có cơ hội hợp tác trực tiếp với nhiều thị trường lớn như Nhật Bản, Mỹ, Châu Âu và các quốc gia trong khu vực Châu Á. Điều này không chỉ mở rộng mạng lưới kinh doanh mà còn củng cố vị thế của công ty như một trong những doanh nghiệp ổn định và có tiềm năng phát triển lâu dài trong ngành.



Hình 3-1: Công ty TNHH Daiwa Việt Nam (DVN)

3.1.1 Lĩnh vực hoạt động

Công ty chuyên sản xuất dụng cụ câu cá cao cấp, bao gồm:

Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam

Cần câu (Fishing Rods): Đa dạng mẫu mã từ cần câu biển, cần câu nước ngọt đến cần câu thể thao.



Hình 3-2: Fishing Rod do Daiwa sản xuất

Guồng quay (Fishing Reels): Công nghệ chống ăn mòn, độ bền cao, phục vụ câu cá chuyên nghiệp.



Hình 3-3: Fishing Reel do Daiwa sản xuất

Thị trường xuất khẩu chính: Nhật Bản, Mỹ, Châu Âu, Australia và các nước ASEAN.

Chứng nhận chất lượng: ISO 9001, ISO 14001, tuân thủ tiêu chuẩn Nhật Bản (JIS).

3.1.2 Quy mô sản xuất.

- Tổng diện tích nhà xưởng: 117.000 m² (gồm 3 nhà máy):
- Nhà máy REEL No.1 (60.000 m²): Sản xuất guồng quay cao cấp, dây chuyên tự động hóa 80%.

Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam

- Nhà máy REEL No.2 (17.000 m²): Lắp ráp linh kiện và sản xuất phụ tùng.
- Nhà máy ROD (40.000 m²): Chuyên về cần câu composite, công nghệ ép sợi carbon.

Công suất sản xuất:

Hơn 2 triệu sản phẩm/năm (2023).

Đáp ứng 30% nhu cầu thị trường toàn cầu của thương hiệu Daiwa.

Hệ thống quản lý:

Áp dụng 5S, Kaizen, ERP trong vận hành.

3.2 Cơ cấu tổ chức sản xuất.

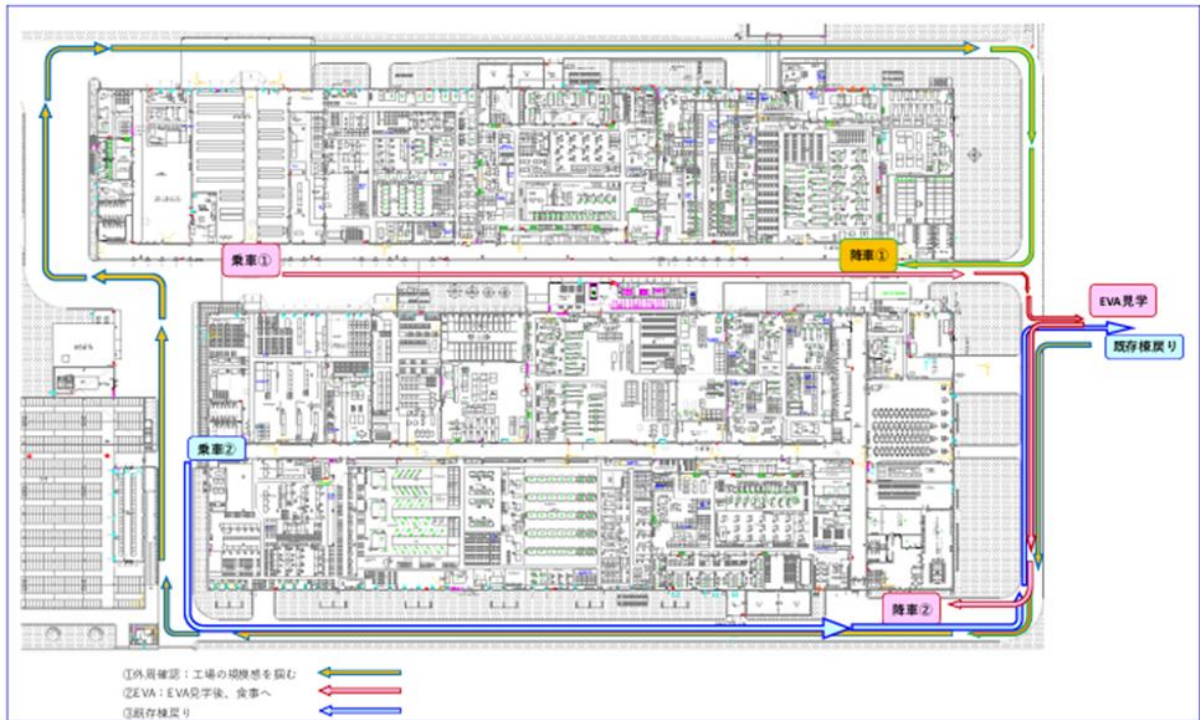
3.2.1 Bộ phận sản xuất cần câu nhà máy Rod

Đặc điểm tổ chức: Nhà máy Rod được thiết kế theo mô hình sản xuất chuyên biệt hóa với hai phân hệ chính:

- Phân hệ kỹ thuật:
 - V291: Bộ phận kỹ thuật sản xuất
 - V293: Bộ phận kỹ thuật sản phẩm
 - V295: Bộ phận thiết kế
- Phân hệ sản xuất:
 - V220-V270: Chuỗi giá trị sản xuất tích hợp từ nguyên liệu thô đến thành phẩm, bao gồm:
 - Xử lý vật liệu composite (V220)
 - Hoàn thiện bề mặt (V230)
 - Hệ thống phun sơn tự động (V240)
 - Dây chuyền lắp ráp module (V260)
 - Hệ thống đóng gói tự động (V270)

Nhận định: Mô hình tổ chức này thể hiện nguyên tắc phân công lao động chuyên môn hóa kết hợp đảm bảo tính hiệu quả trong sản xuất các sản phẩm đòi hỏi độ chính xác cao.

Mặt bằng tổng thể nhà máy sản xuất cần câu cá được thể hiện ở hình sau:



Hình 3-4: Layout nhà máy Rod

3.2.2 Bộ phận sản xuất guồng quay cước nhà máy Reel

Tổng quan nhà máy: Nhà máy Reel thuộc Daiwa Việt Nam là đơn vị chuyên biệt hóa trong lĩnh vực sản xuất và lắp ráp các linh kiện guồng quay cước. Để đáp ứng nhu cầu sản xuất quy mô lớn và mật độ đơn hàng dày đặc, công ty đã tiến hành mở rộng thành hai nhà máy độc lập là Reel 1 và Reel 2.

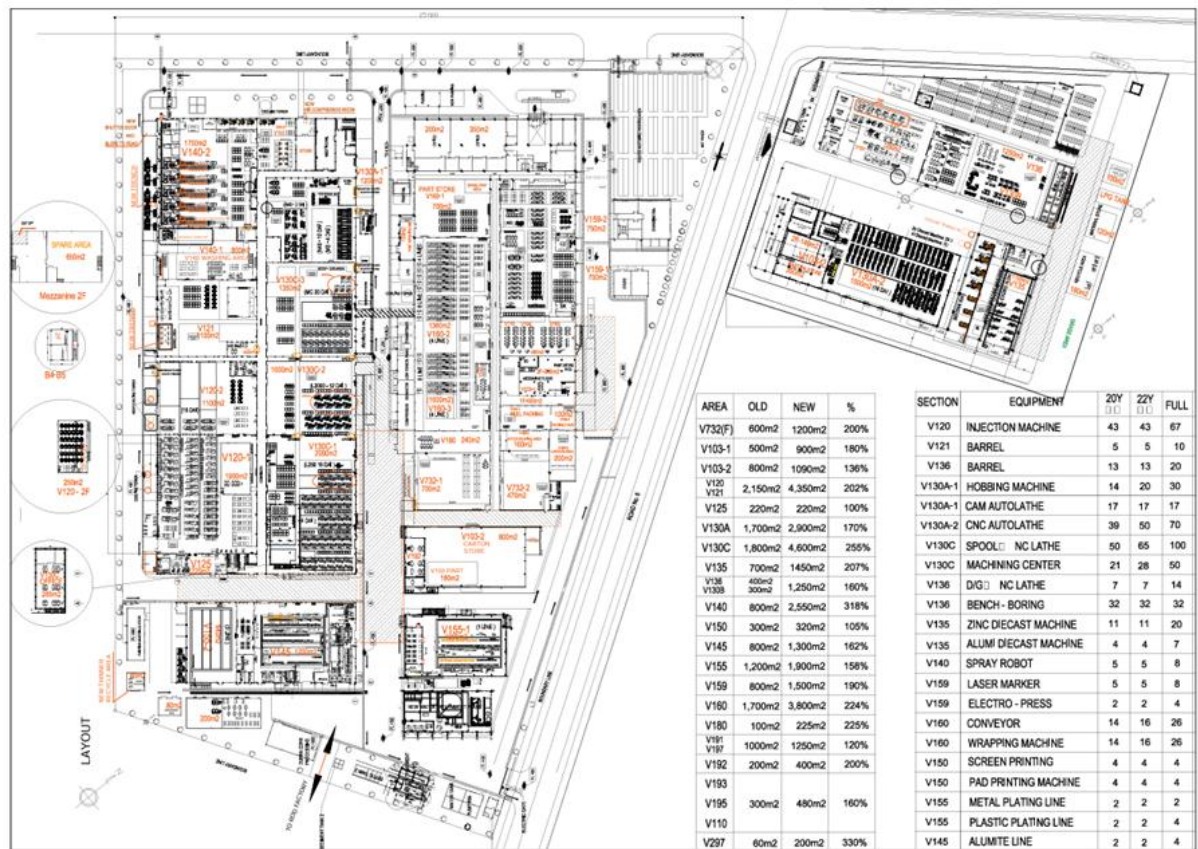
Cơ cấu tổ chức sản xuất: Nhà máy vận hành theo mô hình tổ chức phân cấp rõ rệt, bao gồm hai hệ thống chính:

- Bộ phận gián tiếp (Hỗ trợ kỹ thuật):
 - V110: Phòng quản lý sản xuất
 - V180: Bộ phận đảm bảo chất lượng (QA)
 - V191: Kỹ thuật sản xuất
 - V192: Kỹ thuật máy móc thiết bị
 - V193: Kỹ thuật sản phẩm
 - V194: Kỹ thuật linh kiện
 - V195: Kỹ thuật thiết kế
 - V196: Bộ phận thiết kế 3D
 - V197: Kỹ thuật khuôn
- Bộ phận trực tiếp sản xuất:
 - V120: Khu vực đúc nhựa

Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam

- V121: Hoàn thiện sản phẩm đúc nhựa
- V125: Bộ phận dập
- V130: Khu gia công cơ khí
- V135: Phòng đúc
- V136: Hoàn thiện sản phẩm đúc
- V140: Khu vực sơn
- V145: Bộ phận anot hóa
- V150: Khu vực in ấn
- V155: Bộ phận mạ nhựa
- V159: Khu vực tiền lắp ráp
- V160: Dây chuyền lắp ráp giường hoàn chỉnh

Mặt bằng tổng thể nhà máy sản xuất giường quay cước được thể hiện như hình sau:



Hình 3-5: Layout nhà máy REEL

3.2.3 Bộ phận văn phòng

Là doanh nghiệp 100% vốn Nhật Bản, Daiwa Việt Nam áp dụng mô hình quản lý theo phong cách Nhật với hệ thống phân công chức năng rõ ràng và chuyên môn hóa cao. Bộ phận văn phòng được tổ chức thành các phòng ban chuyên môn sau:

- Phòng Tổng hợp:
 - Đảm nhiệm công tác hành chính tổng hợp
 - Quản lý nhân sự và các vấn đề pháp lý
 - Phòng Mua hàng:
 - Quản lý chuỗi cung ứng nguyên vật liệu
 - Đàm phán và quản lý nhà cung cấp
 - Phòng Kế toán:
 - Quản lý tài chính doanh nghiệp
 - Kiểm soát chi phí và báo cáo thuế
 - Phòng Sales:
 - Quản lý quan hệ khách hàng
 - Phát triển thị trường và chăm sóc đơn hàng
 - Phòng Kho:
 - Quản lý tồn kho nguyên vật liệu và thành phẩm
 - Kiểm soát xuất nhập hàng hóa
 - Phòng IT:
 - Bảo trì hệ thống công nghệ thông tin
 - Phát triển phần mềm quản lý nội bộ
- Nhận xét: Các bộ phận có sự phối hợp nhịp nhàng để đảm bảo hoạt động kinh doanh sản xuất của công ty.

3.2.4 Bộ phận Quality Assurance QA

Với mục tiêu và sứ mệnh Daiwa luôn đặt chất lượng của sản phẩm lên hàng đầu. Vì vậy, từ các quy trình sản xuất, linh kiện, lắp ráp cần câu cá và guồng quay cước luôn được các bộ phận kiểm tra chất lượng nghiêm ngặt. Đặc biệt hơn hết là bộ phận QA luôn phải đảm bảo chất lượng cho các linh kiện, sản phẩm cần câu cá và guồng quay cước, ngăn chặn hàng lỗi đến tay khách hàng.

Gồm có các bộ phận:

- QA Reel V180: Kiểm soát chất lượng sản phẩm guồng quay của nhà máy Reel.
- QA Rod V280: Kiểm soát chất lượng sản phẩm cần câu của nhà máy Rod.

3.3 Phân tích hiện trạng tại bộ phận V159

Công ty Daiwa Việt Nam là một doanh nghiệp chuyên sản xuất các sản phẩm dụng cụ câu cá, bao gồm Reel (guồng quay cước) và Rod (cần câu). Quy trình sản xuất Reel đòi hỏi phải trải qua nhiều công đoạn phức tạp, trong đó, khâu lắp ráp đóng vai trò quan trọng, ảnh hưởng trực tiếp đến chất lượng thành phẩm.

Tương tự như nhiều doanh nghiệp sản xuất khác, Công ty Daiwa Việt Nam cũng gặp phải những thách thức trong quá trình lắp ráp, khi các lỗi kỹ thuật thường xuyên phát sinh, làm giảm hiệu suất và độ tin cậy của sản phẩm.

Hiện tại, bộ phận V159 (tiền lắp ráp) sản xuất của công ty có tổng cộng 10 quy trình lắp ráp Reel. Trong đó, hai quy trình lắp ráp Rotor Un và O.Slider được xác định là có tỷ lệ lỗi cao nhất, đòi hỏi phải có giải pháp khắc phục kịp thời để nâng cao chất lượng sản xuất.

Trong khuôn khổ luận văn này, nghiên cứu sẽ tập trung phân tích chi tiết quy trình lắp ráp O.Slider, một công đoạn quan trọng trong dây chuyền sản xuất Reel tại Công ty Daiwa Việt Nam.

3.4 Thống kê các dạng sai lỗi trong quá trình lắp ráp O.Slider

Qua khảo sát thực tế và phân tích dữ liệu sản xuất, các lỗi tại quy trình lắp ráp O.Slider được phân loại thành 5 nhóm chính và được định nghĩa dưới bảng sau:

Bảng 3-1: Thống kê các dạng sai lỗi

Nhóm lỗi	Định Nghĩa
Lắp ráp sai vị trí	Lắp sai linh kiện hoặc đặt linh kiện không đúng vị trí/kích thước theo thiết kế.
Lắp ráp thiếu linh kiện	Sản phẩm hoàn thiện thiếu một hoặc nhiều bộ phận cần thiết.
Bị trầy xước	Linh kiện hoặc sản phẩm bị trầy xước trong quá trình lắp ráp
Hư hỏng linh kiện	Linh kiện bị lỗi hoặc hư hỏng trước khi đưa vào dây chuyền lắp ráp.
Siết lực không đúng	Vít, đai ốc siết quá chặt hoặc quá lỏng, ảnh hưởng đến độ bền.

Dựa trên dữ liệu được thu thập từ hệ thống đo lường và báo cáo hiện tại bộ phận V159 (tiền lắp ráp) trong 6 tháng cuối năm 2024, kết quả thống kê các dạng lỗi phát sinh trong quy trình lắp ráp O.Slider được tổng hợp và thể hiện qua bảng sau:

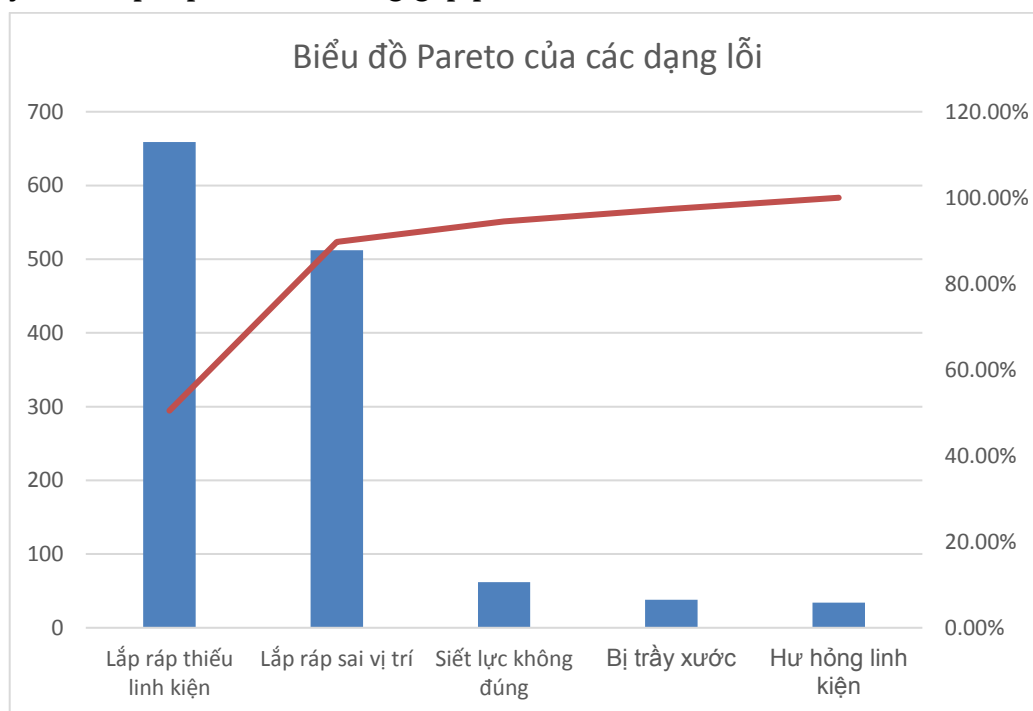
Bảng 3-2: Thống kê các dạng sai lỗi của O.Slider AS 6 tháng cuối năm 2024

Tên lỗi	Bảng thống kê lỗi O.SLIDER 6 tháng cuối năm 2024					
	T7	T8	T9	T10	T11	T12
Lắp ráp sai vị trí	86	87	86	81	91	81
Lắp ráp thiếu linh kiện	102	112	121	108	99	117
Bị trầy xước	8	5	7	6	9	3

Hư hỏng linh kiện	5	7	9	5	5	3
Siết lực không đúng	10	12	14	6	11	9
Tổng số NG	211	223	237	206	215	213
Tổng sản lượng	40951	41956	39861	40213	41002	38092
Tỉ lệ NG	0.52%	0.53%	0.59%	0.51%	0.52%	0.56%

Từ bảng trên nhận thấy tỉ lệ NG trong 6 tháng cuối năm 2024 dao động từ 0.51% đến 0.59%, và tỉ lệ NG trung bình là 0.54%. Số lượng lỗi thấp nhất vào tháng 10 với 206 thành phẩm NG, và cao nhất trong tháng 9 là 237 thành phẩm NG. Tổng số lỗi 6 tháng cuối năm 2024 là: 1305 thành phẩm

Dưới đây là biểu đồ Pareto nhằm tìm ra các dạng lỗi nào ảnh hưởng lớn đến sản phẩm mà quy trình lắp ráp O.Slider đang gặp phải:



Hình 3-6: Biểu đồ Pareto của các dạng lỗi

Từ biểu đồ Pareto trên nhận thấy 2 dạng lỗi quan trọng nhất ảnh hưởng lớn đến lắp ráp O.Slider lần lượt là: Lắp ráp thiếu linh kiện và Lắp ráp sai vị trí chiếm khoảng 90% tổng số lỗi gây ra hàng NG. Cần phải tìm ra nguyên nhân và giải pháp nhằm giải quyết 2 dạng lỗi trên đầu tiên.

3.5 Phân tích chi phí cho thực trạng hiện tại.

Theo thống kê thu được từ công ty thì có khoảng 1305 O.slider bị NG trong 6 tháng cuối năm 2024 của công đoạn lắp ráp O.slider tại bộ phận V159. Từ đó ước tính cả năm số lỗi NG trung bình là: 2610 bán thành phẩm.

3.5.1 Chi phí sửa chữa

Ước tính có khoảng 2349 (90%) thành phẩm NG có thể sửa chữa.

Chi phí sửa chữa bao gồm chi phí thời gian nhân công dành ra để sửa chữa, và các linh kiện thay thế nếu có.

Quy tắc tính chi phí thời gian sửa chữa:

- Ước tính lương cho 1 giờ là 35.000 VNĐ/giờ
- Thời gian sửa chữa trung bình cho 1 bán thành phẩm là 5 phút.
- Tổng thời gian sửa chữa cho 2.349 bán thành phẩm là $= 5 * 2349 = 11.745$ phút (tương đương 197.75 giờ)
- Chi phí cho thời gian sửa chữa $= 197.45 * 35.000 = 6.851.250$ VNĐ

Chi phí linh kiện thay thế nếu có ước tính trung bình cho mỗi sản phẩm là 5.000 VNĐ. Tổng chi phí linh kiện thay thế $= 2349 * 5.000 = 11.745.000$ VNĐ

Tổng chi phí sửa chữa $= 6.851.250 + 11.745.000 = 18.596.250$ VNĐ

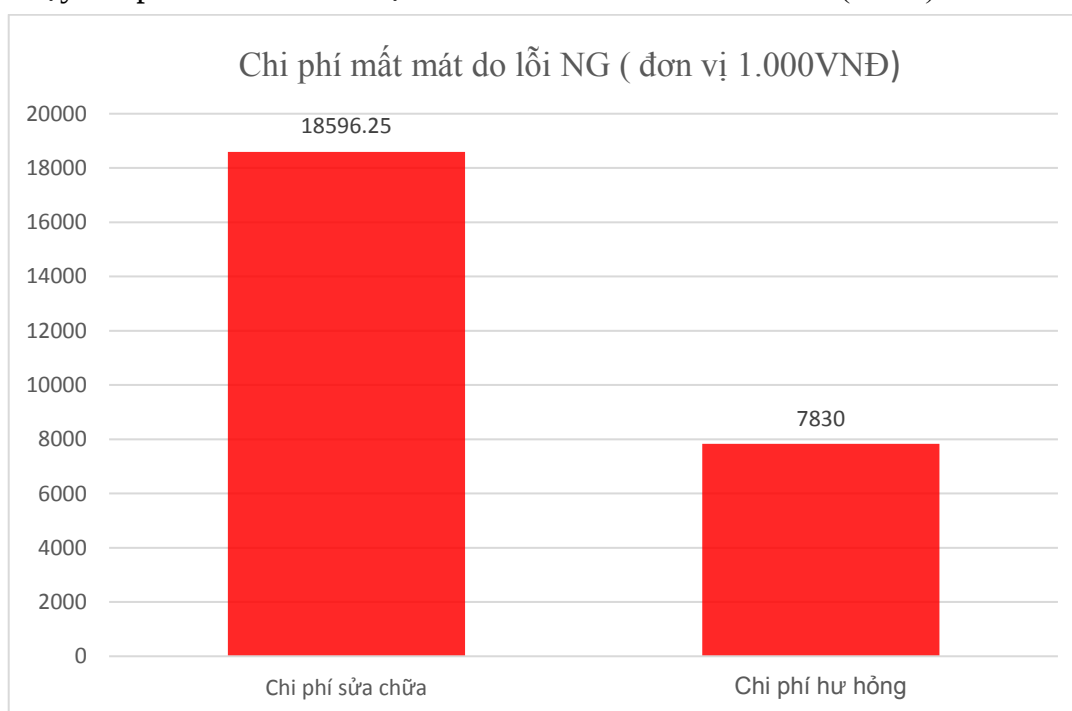
3.5.2 Chi phí hư hỏng

Ước tính có khoảng 261 (10%) bán thành phẩm NG không thể sử dụng và chi phí cho mỗi bán thành phẩm là: 30.000 VNĐ cho 1 bán thành phẩm.

Tổng chi phí hư hỏng $= 261 * 30.000 = 7.830.000$ VNĐ

Tổng chi phí mất mát mà lỗi NG gây ra $=$ Tổng chi phí sửa chữa + Tổng chi phí hư hỏng $= 18.596.250 + 7.830.000 = 26.426.250$ VNĐ/ năm

Vậy chi phí mất mát đơn vị $= 26.426.250 / 2.610 = 10.125$ (VNĐ)



Hình 3-7: Biểu đồ chi phí mất mát do lỗi NG

Hình 3-7 minh họa phân bố chi phí mất mát do lỗi NG trong 6 tháng cuối năm

Nguyên nhân chính	Diễn giải
Con người	Thiếu đào tạo: nhân viên không được đào đầy đủ về quy trình lắp ráp. Mệt mỏi/Áp lực: làm việc quá tải khiến nhân viên quên và bỏ bước. Thiếu tập trung: Sai sót do thói quen, hoặc do môi trường gây xao nhãng.
Quy trình thiếu chuẩn hóa	Thiếu hệ thống giám sát: không sử dụng cảnh báo tự động hoặc công nghệ hỗ trợ.
Linh kiện	Đặt linh kiện sai vị trí: gây ra ảnh hưởng lấy nhầm linh kiện
Môi trường	Tiếng ồn: Ảnh hưởng đến mất tập trung công nhân

2024. Nhận thấy, phần lớn chi phí ($\approx 70\%$) đến từ công đoạn sửa chữa sản phẩm lỗi, trong khi khoảng 30% đến từ việc loại bỏ hoàn toàn sản phẩm hư hỏng. Điều này cho thấy nếu giảm được lỗi NG, doanh nghiệp sẽ tiết kiệm chủ yếu ở khâu tái xử lý sản phẩm, giúp tối ưu chi phí và nhân lực.

3.6 Phân tích nguyên nhân

Sử dụng phương pháp 5 Why đặt câu hỏi cho công nhân, nhân viên và quản lý trong bộ phận thu được những nguyên nhân gây ra lỗi lắp ráp thiếu linh kiện và lắp ráp sai vị trí được thể hiện bảng sau:

Bảng 3-3: Nguyên nhân chính gây ra lỗi lắp ráp thiếu linh kiện và sai vị trí.

Từ kết quả phân tích bảng số liệu, có thể nhận thấy yếu tố con người đóng vai trò chủ yếu trong việc phát sinh hàng không đạt chất lượng (NG). Thực trạng này phản ánh một vấn đề phổ biến tại các môi trường sản xuất chủ yếu dựa vào thủ công hoặc thiếu sự hỗ trợ của các giải pháp công nghệ hiện đại.

Để khắc phục tình trạng trên, việc ứng dụng hệ thống giám sát thông minh vào quy trình sản xuất là một giải pháp tối ưu. Hệ thống này không chỉ giúp kiểm soát hiệu quả các yếu tố liên quan đến quy trình và môi trường sản xuất mà còn hỗ trợ đắc lực cho người vận hành, từ đó giảm thiểu đáng kể những sai sót phát sinh do thiếu tập trung hoặc áp lực công việc. Mặc dù đòi hỏi nguồn vốn đầu tư ban đầu, giải pháp này hứa hẹn mang lại hiệu quả bền vững thông qua việc nâng cao năng suất, giảm chi phí khắc phục lỗi và cải thiện chất lượng sản phẩm một cách toàn diện.

*Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty
TNHH Daiwa Việt Nam*

Việc triển khai hệ thống giám sát thông minh do đó không chỉ là một giải pháp công nghệ mà còn là một chiến lược dài hạn nhằm nâng cao năng lực cạnh tranh của doanh nghiệp trong bối cảnh cách mạng công nghiệp 4.0.

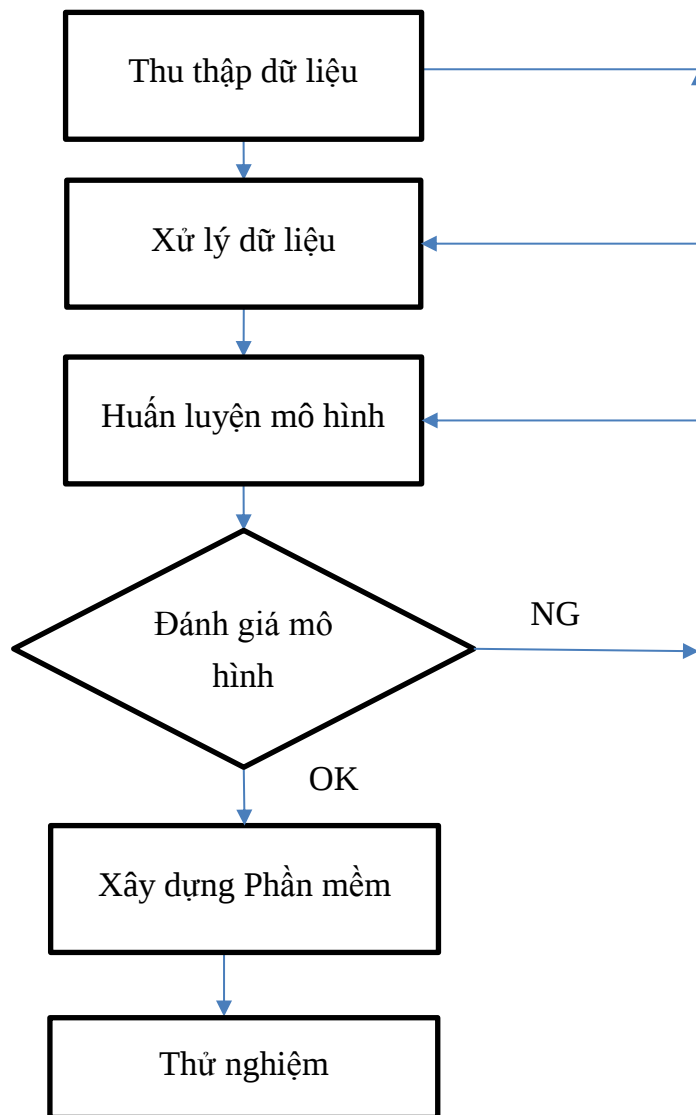
CHƯƠNG 4: XÂY DỰNG HỆ THỐNG GIÁM SÁT TỰ ĐỘNG CHO QUY TRÌNH LẮP RÁP SẢN PHẨM

Trong bối cảnh sản xuất công nghiệp hiện đại, việc đảm bảo chất lượng sản phẩm là yếu tố then chốt để duy trì uy tín và năng lực cạnh tranh của doanh nghiệp. Tại công ty Daiwa chuyên sản xuất Reel (guồng quay cước), quy trình lắp ráp O.slider đóng vai trò quan trọng trong việc hoàn thiện sản phẩm. Tuy nhiên, tỷ lệ hàng không đạt tiêu chuẩn (NG) trong công đoạn này vẫn còn tồn tại, gây lãng phí nguyên vật liệu, thời gian và gia tăng chi phí sản xuất.

Nguyên nhân chính dẫn đến tình trạng này thường xuất phát từ các lỗi kỹ thuật trong quá trình lắp ráp, sai sót trong kiểm tra thủ công hoặc thiếu hệ thống giám sát chặt chẽ. Do đó, việc ứng dụng hệ thống giám sát thông minh vào dây chuyền lắp ráp O.slider là giải pháp cấp thiết nhằm giảm thiểu hàng NG, nâng cao hiệu quả sản xuất và đảm bảo chất lượng đầu ra.

Hệ thống giám sát được xây dựng với ba thành phần chính: camera, phần mềm giám sát, và thiết bị thông báo (đèn tín hiệu), tạo thành một quy trình hoạt động hiệu quả. Camera ghi lại hình ảnh và video từ khu vực giám sát, truyền dữ liệu đến phần mềm giám sát. Phần mềm này xử lý, phân tích dữ liệu và cung cấp khả năng xem trực tiếp hoặc xem lại. Khi phát hiện sự kiện bất thường, phần mềm kích hoạt đèn tín hiệu để cảnh báo kịp thời cho công nhân gia công.

Để thực hiện xây dựng hệ thống trên thì dưới đây là Quy trình thực hiện được tiến hành:

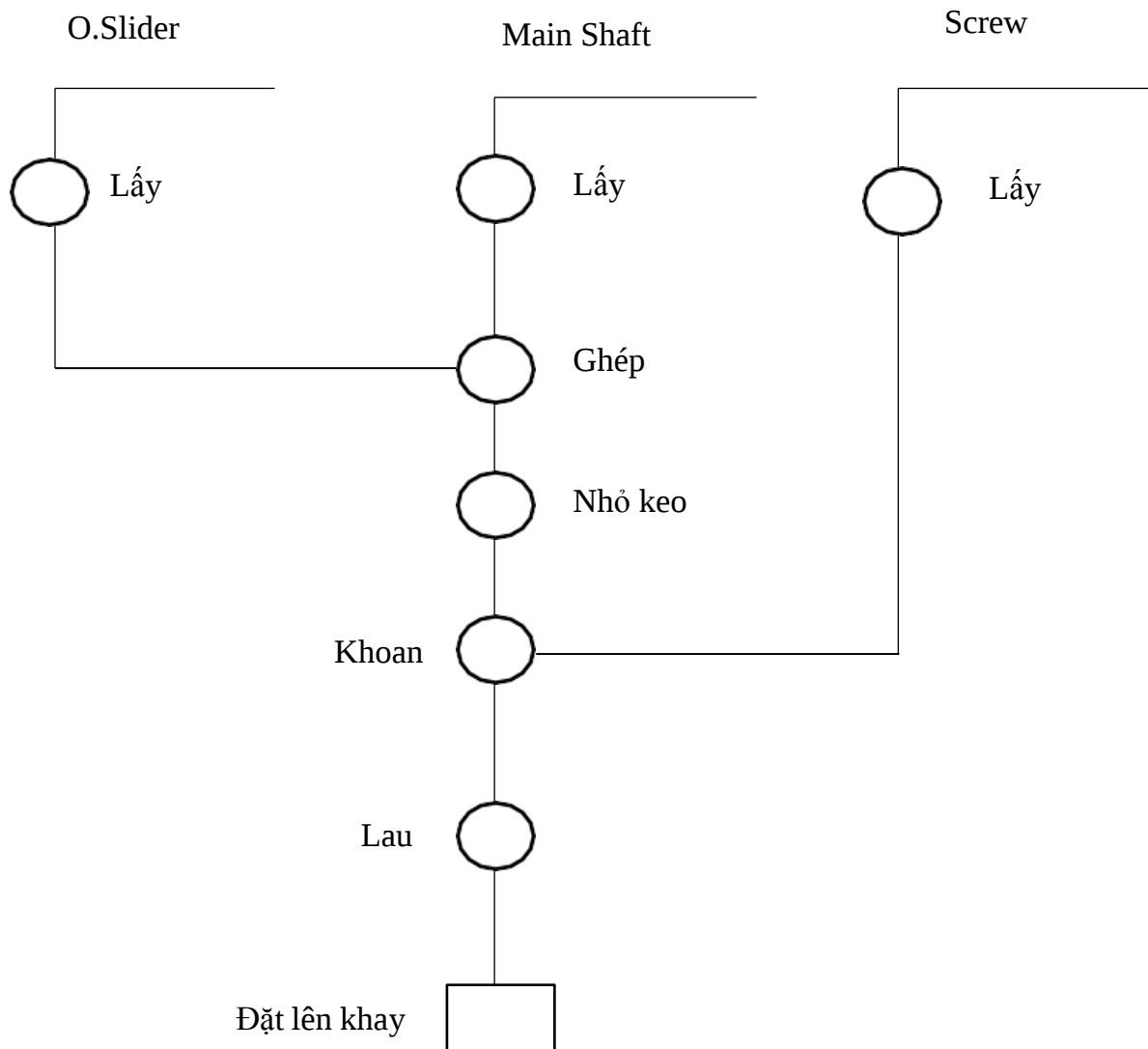


Hình 4-1: Quy trình thực hiện xây dựng hệ thống

4.1 Thu thập dữ liệu

4.1.1 Xác định quy trình lắp O.Slider

O..Slider là cụm lắp ráp quan trọng, bao gồm ba thành phần chính: (1) O.Slider, (2) Main Shaft, và (3) Screw, được kết hợp theo một trình tự chuẩn nhằm đảm bảo độ chính xác và độ bền của sản phẩm. Quy trình lắp ráp được minh họa chi tiết thông qua sơ đồ công nghệ (Hình 4-2).



Hình 4-2: Quy trình lắp ráp O.Slider As

- O.Slider As được công nhân thực hiện bằng máy bán tự động. Đầu tiên công nhân tiến hành đổ keo vào máy nhỏ keo. Và chuẩn bị đầy đủ linh kiện trước khi lắp ráp.
- Sau đó công nhân sẽ lấy O.Slider và Main Shaft ghép lại với nhau, sau đó sẽ đặt lên đồ gá. Tiếp đó công nhân tiến hành nhỏ keo lên vị trí đóng Screw theo bảng chỉ đạo công việc.
- Tiếp đến công nhân sẽ dùng Driver điện (khoan), để khoan Screw vào Main Shaft.
- Cuối cùng là lau sạch keo bị lem ra ở các bề mặt. Và đặt lên khay chứa.

4.1.2 Xác định các thao tác chính

Qua khảo sát quy trình lắp ráp O.Slider bằng máy bán tự động, nhận thấy các thao tác then chốt bao gồm: “Ghép O.Slider và Main Shaft”, “Lấy O.Slider”, “Lấy Screw”, “Khoan bằng Driver điện”.

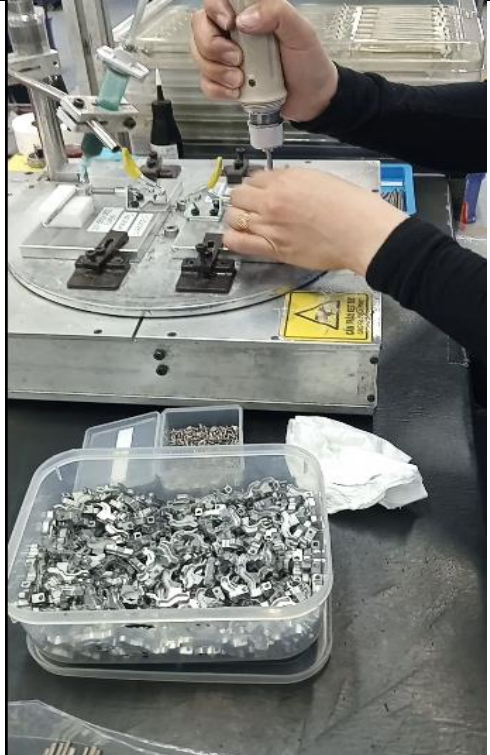
Các thao tác trên đặc trưng bởi chuỗi hành động lặp lại, có thể lượng hóa thông qua dữ liệu cảm biến hoặc hình ảnh. Do đó, chúng được lựa chọn làm tập dữ liệu huấn luyện (training dataset) nhằm xây dựng mô hình nhận diện phục vụ giám sát tự động hoặc hỗ trợ công nhân.

4.1.3 Thu thập dữ liệu hình ảnh

Sau khi hoàn thiện việc xác định các thao tác cần nhận diện, nghiên cứu tiến hành thu thập dữ liệu hình ảnh để xây dựng bộ dữ liệu (dataset) phục vụ cho quá trình huấn luyện và đánh giá mô hình. Dữ liệu thu thập được chia thành ba tập chính theo tỷ lệ phổ biến trong học máy: 80% dành cho tập huấn luyện (training set), 15% cho tập kiểm định (validation set) và 5% cho tập kiểm thử (test set). Việc phân chia này đảm bảo mô hình được đánh giá khách quan trên dữ liệu chưa từng xuất hiện trong quá trình huấn luyện.

Bảng 4-1: Mô tả dataset cần thu thập cho các thao tác

Thao tác	Training set	Validation set	Ảnh minh họa
Ghép O.Slider và Main Shaft	48	13	
Lấy O.Slider	30	8	

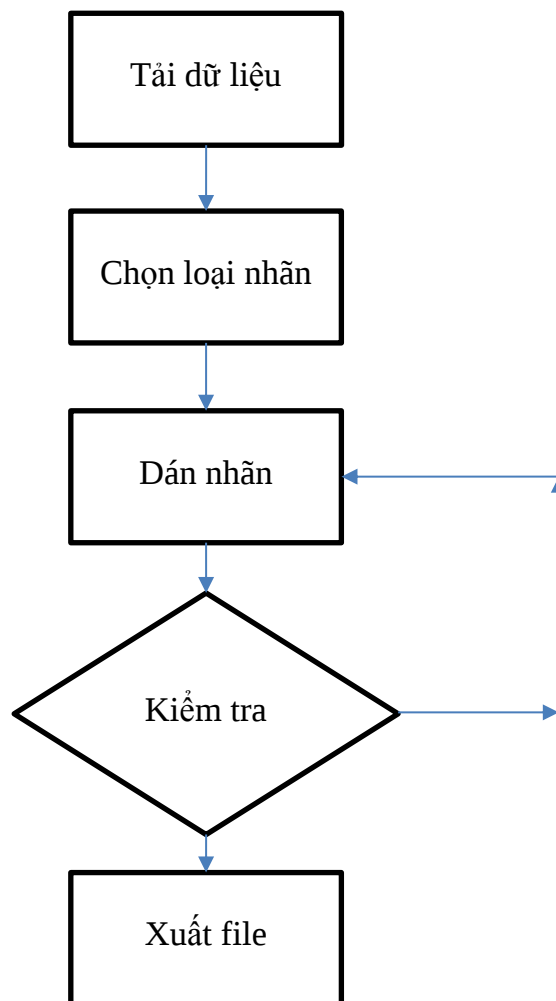
Lấy Screw	38	11	
Khoan bằng Driver điện	45	10	

Bảng dữ liệu trên trình bày số lượng mẫu được phân chia trong các tập dữ liệu khác nhau (Training set, Validation set) cho 4 thao tác: "Ghép O.Slider và Main Shaft", "Lấy O.Slider", "Lấy Screw", "Khoan bằng Driver điện". Với tổng số dữ liệu cho quá trình Training model là: 161 , số dữ liệu quá trình Validation là: 42.

4.2 Xử lý dữ liệu

Dán nhãn dữ liệu (Data Labeling) là một bước quan trọng trong quá trình xây dựng mô hình học máy, đặc biệt với các bài toán học có giám sát (supervised learning). Chất lượng nhãn ảnh hưởng trực tiếp đến hiệu suất của mô hình, do đó cần được thực hiện cẩn thận và nhất quán. Trong nghiên cứu này, dữ liệu đầu vào được gán nhãn thủ công bằng công cụ **Label Studio**. Các nhãn được kiểm tra chéo (cross-verification) bởi nhiều người để giảm thiểu sai sót, đồng thời tuân thủ các tiêu chuẩn đã định nghĩa trước (ví dụ: bounding box phải bao phủ toàn bộ đối tượng). Kết quả dữ liệu sau gán nhãn được lưu trữ dưới định dạng TXT để phù hợp với đầu vào của mô hình. Bước này đòi hỏi nhiều thời gian và chi phí, nhưng là nền tảng thiết yếu để mô hình học được các đặc trưng chính xác từ dữ liệu.

Quy trình dán nhãn cho bộ dữ liệu trong nghiên cứu này được thiết kế tỉ mỉ nhằm đảm bảo tính nhất quán và độ tin cậy của yếu tố then chốt quyết định hiệu suất mô hình học máy. Cụ thể, quy trình được trình bày bằng flow chart (hình 4-3) sau:



Hình 4-3: Quy trình dán nhãn

4.2.1 Tải dữ liệu

Label Studio [11] được lựa chọn làm công cụ chính để dán nhãn dữ liệu nhờ những ưu điểm vượt trội về tính linh hoạt, khả năng hỗ trợ đa dạng định dạng dữ liệu (hình ảnh, văn bản, âm thanh), và giao diện trực quan phù hợp cho cả người dùng không chuyên. Bên cạnh đó, công cụ này cho phép tùy chỉnh workflow dán nhãn theo yêu cầu của từng dự án, đồng thời hỗ trợ xuất dữ liệu ở nhiều định dạng phổ biến (JSON, CSV), thuận tiện cho quá trình huấn luyện mô hình học máy.

Về quy trình tải dữ liệu lên Label Studio, hệ thống cho phép tải trực tiếp từ máy tính cá nhân. Dữ liệu sau khi được tải lên sẽ được tự động tổ chức thành các project, giúp quản lý và phân công công việc dán nhãn một cách hiệu quả.

4.2.2 Chọn loại nhãn

Trong nghiên cứu này, kiểu nhãn **Object Detection** được lựa chọn để gán nhãn dữ liệu hình ảnh, vì những lý do sau.

Thứ nhất, Object Detection cho phép xác định đồng thời cả vị trí không gian và danh mục của các đối tượng trong hình ảnh thông qua việc sử dụng các hộp giới hạn (bounding box) và nhãn lớp, cung cấp thông tin toàn diện để huấn luyện các mô hình học sâu. Điều này đặc biệt phù hợp với các ứng dụng thực tiễn như nhận diện phương tiện giao thông hoặc giám sát an ninh, nơi yêu cầu phát hiện chính xác vị trí và loại đối tượng.

Thứ hai, dữ liệu được gán nhãn theo kiểu Object Detection hỗ trợ các thuật toán học máy trong việc học các đặc trưng không gian và ngữ nghĩa, từ đó cải thiện độ chính xác trong việc phát hiện các đối tượng phức tạp trong các bối cảnh đa dạng.

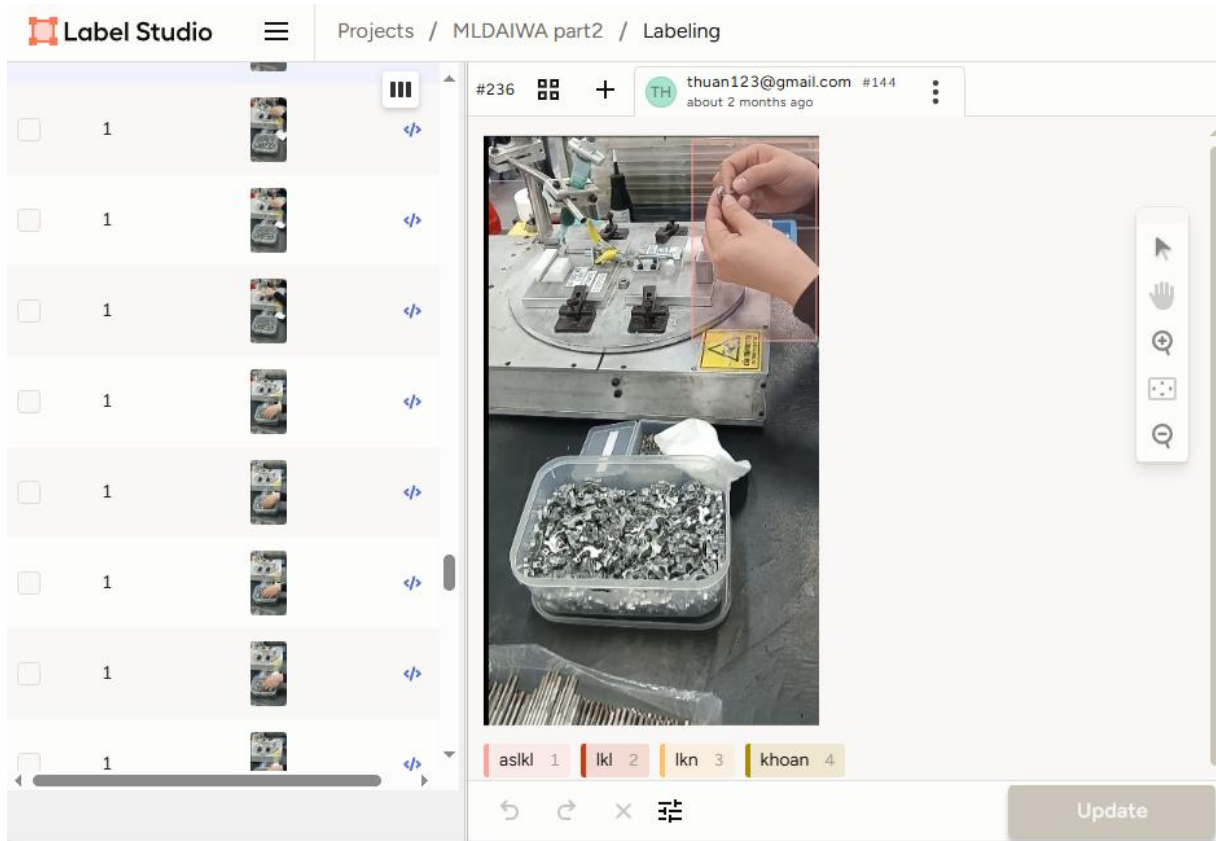
Thứ ba, kiểu nhãn này tương thích với nhiều kiến trúc mô hình hiện đại như YOLO, Faster R-CNN, và SSD, đảm bảo tính linh hoạt và khả năng mở rộng của nghiên cứu.

Cuối cùng, việc sử dụng Object Detection trong gán nhãn dữ liệu giúp tối ưu hóa quá trình huấn luyện bằng cách cung cấp tập dữ liệu có cấu trúc, giảm thiểu sự mơ hồ và nâng cao hiệu quả của mô hình trong các tác vụ thị giác máy tính.

4.2.3 Dán nhãn

Quy trình dán nhãn dữ liệu hình ảnh được thực hiện theo ba bước tuần tự.

Đầu tiên, cần lựa chọn hình ảnh cần gán nhãn từ bộ dữ liệu đầu vào. Được mô tả trong hình 4-4 sau:



Hình 4-4: Giao diện thao tác chọn nhãn trong label-studio

Tiếp theo, xác định loại nhãn phù hợp với đối tượng trong ảnh, đảm bảo tính nhất quán trong việc phân loại. Dưới đây là ảnh minh họa cho các loại nhãn trong bài toán nhận diện .

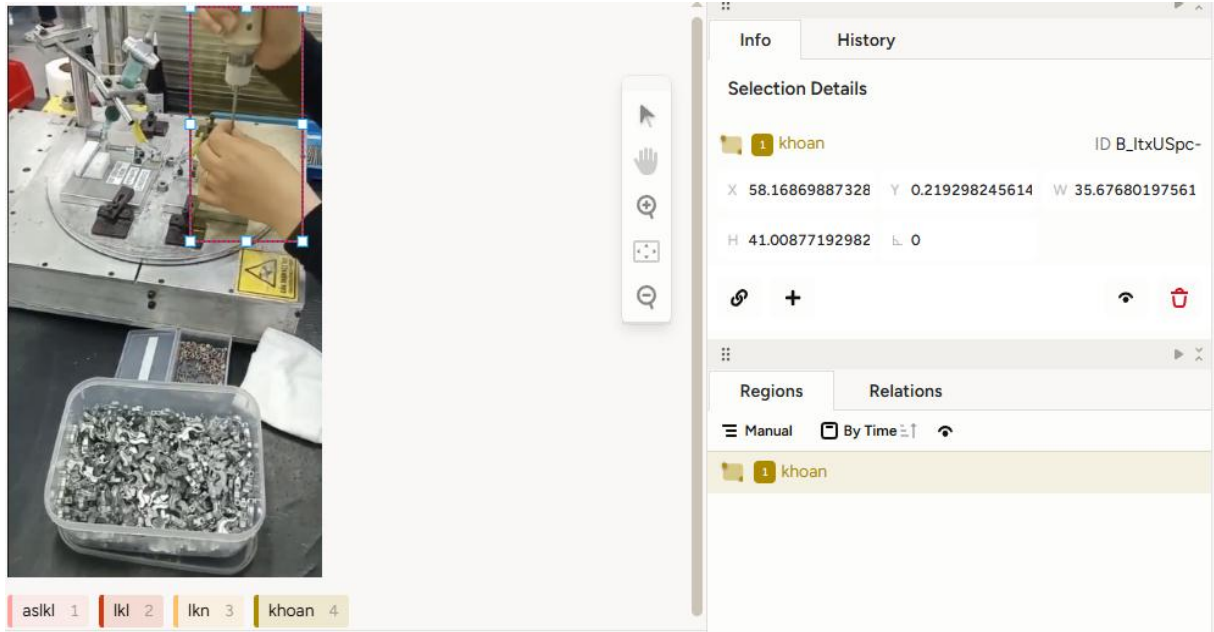


Hình 4-5: Các loại nhãn trong bài toán nhận diện

Các nhãn trên tương ứng với các thao tác lắp ráp cụ thể như sau:

- Nhãn “aslkl” tương ứng với thao tác: Ghép O.Slider với trục chính (Main Shaft).
- Nhãn “lkl” tương ứng với thao tác: Lấy linh kiện O.Slider.
- Nhãn “lkn” tương ứng với thao tác: Lấy linh kiện Screw.
- Nhãn “khoan” tương ứng với thao tác: Thực hiện công đoạn khoan.

Cuối cùng, thao tác khoan vùng đối tượng cần đánh dấu được thực hiện, sau đó người dùng xác nhận để hoàn tất quá trình gán nhãn. Quy trình này đảm bảo độ chính xác và hiệu quả trong việc chuẩn bị dữ liệu cho mô hình

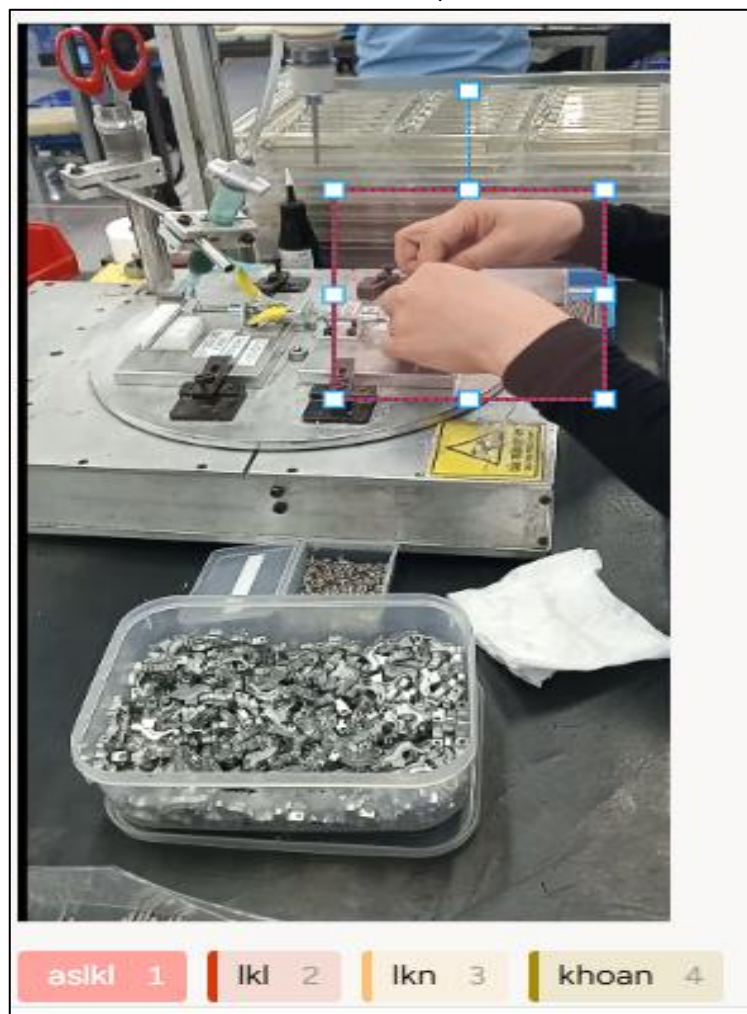


Hình 4-6: Mô tả thao tác khoan vùng trong Label Studio.

Hình 4-6 minh họa quá trình dán nhãn cho công đoạn khoan, vùng label được chọn bao phủ hoàn toàn thao tác khoan và máy khoan. Nhằm mục đích phục vụ quá trình train model nhận diện.

4.2.4 Kiểm tra và xuất file

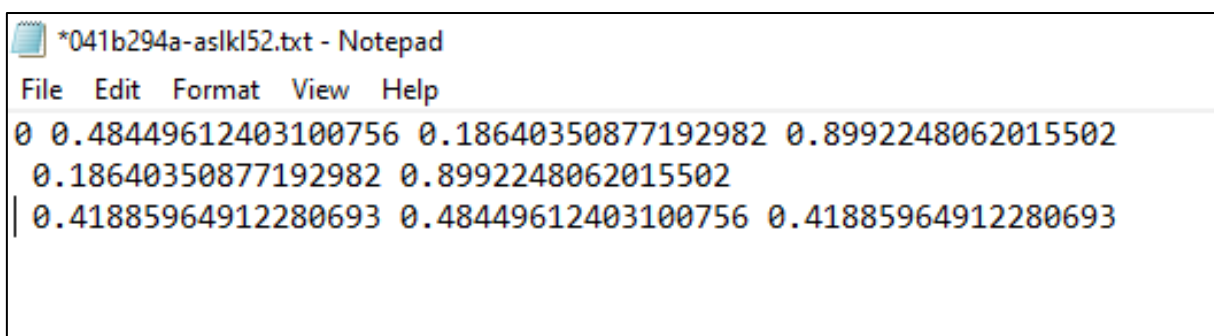
Sau khi hoàn thành quá trình gán nhãn dữ liệu, bước kiểm tra lại các nhãn đóng vai trò quan trọng nhằm đảm bảo tính chính xác và nhất quán của tập dữ liệu. Thao tác này bao gồm việc rà soát từng nhãn đã được gán, phát hiện và chỉnh sửa các lỗi sai (nếu có), đồng thời xác minh tính tương thích giữa nhãn và vùng đối tượng tương ứng trên dữ liệu hình ảnh. Sau khi hoàn tất kiểm tra, dữ liệu được xuất thành file txt theo cấu tạo độ.



Hình 4-7: Minh họa quá trình dán nhãn lắp ráp O.slider

Hình 4-7: Minh họa quá trình dán nhãn lắp ráp O.Slider, cho thấy giao diện của Label Studio khi khoanh vùng đối tượng (bounding box) và gắn nhãn lớp.

Ví dụ, một hình ảnh có thể hiển thị công nhân đang thực hiện thao tác “lấy O.Slider” với bounding box bao quanh tay và linh kiện.



Hình 4-8: Kết quả thu được sau khi dán nhãn

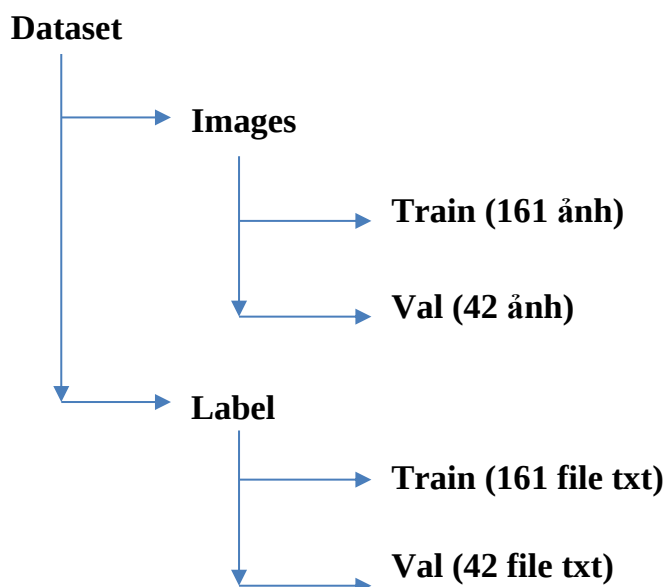
Hình 4-8 thể hiện kết quả cuối cùng của quá trình gắn nhãn mô tả tọa độ cần nhận diện tương ứng, với các bounding box và nhãn lớp được hiển thị trên hình ảnh, cùng với

dữ liệu được xuất ra dưới dạng file TXT theo chuẩn YOLO. Đây là bước chuẩn bị dữ liệu để huấn luyện mô hình.

Định dạng đầu ra dữ liệu được xuất dưới dạng file TXT theo chuẩn YOLO, chứa thông tin về:

- Nhãn lớp (class label): Ví dụ, “lkl” (lấy O.Slider), “lkn” (lấy Screw), “khoan” (khoan bằng Driver điện), “aslk” (ghép O.Slider và Main Shaft).
- Tọa độ bounding box: Bao gồm các giá trị (x_center, y_center, width, height) được chuẩn hóa theo kích thước hình ảnh.
- Mỗi hình ảnh có một file TXT tương ứng, chứa thông tin về tất cả các đối tượng được gán nhãn trong hình ảnh đó.

File sau khi được xuất ra sẽ được sắp xếp lại và được trình bày theo các đường dẫn được mô tả như hình 4-9:



Hình 4-9: Đường dẫn của tập Dataset

Việc chia dataset như trong Hình 4-9: Đường dẫn của tập Dataset (gồm Train (161 ảnh) và Val (42 ảnh), cùng các file nhãn tương ứng) nhằm phục vụ cho quá trình train mô hình một cách logic và khoa học.

Tập Train (80%): 161 ảnh được sử dụng để huấn luyện mô hình,.

Tập Val (20%): 42 ảnh được dùng để kiểm tra và đánh giá hiệu suất mô hình trong quá trình huấn luyện, tránh hiện tượng overfitting và đảm bảo khả năng tổng quát hóa.

4.3 Huấn luyện mô hình

Trong nghiên cứu này, Google Colab được sử dụng làm nền tảng huấn luyện mô hình nhờ khả năng mở rộng, tiết kiệm chi phí và hỗ trợ tài nguyên GPU/TPU miễn phí (Tesla T4). Môi trường tích hợp sẵn thư viện AI/ML (TensorFlow, PyTorch) cùng khả năng

năng đồng bộ hóa dữ liệu qua Google Drive giúp tối ưu thời gian triển khai, phù hợp cho giai đoạn prototype. Đặc biệt, YOLO11n – phiên bản Nano của YOLO11 (Ultralytics) – được lựa chọn nhờ hiệu suất cao và khả năng triển khai trên thiết bị tài nguyên hạn chế, đáp ứng yêu cầu nhận diện thao tác người trong thời gian thực cho các ứng dụng như tương tác người-máy.

Quá trình huấn luyện mô hình YOLO11n được triển khai trên Google Colab như sau:

```
!pip install ultralytics
from ultralytics import YOLO
!yolo task=detect mode=train epochs=100 model='/content/yolo11n.pt'
data='/content/drive/MyDrive/dataset/mydata.yaml'
```

Mô hình YOLO11n được huấn luyện trong 100 epochs trên nền tảng Google Colab, sử dụng bộ dữ liệu gồm 200 mẫu.

Mô hình best.pt sau khi huấn luyện đã được kiểm thử nghiêm trên bộ mẫu dữ liệu hoàn toàn mới không xuất hiện trong tập huấn luyện, thu được kết quả như sau:

Bảng 4-2: Kết quả thu được sau khi huấn luyện model.

Công đoạn	Ảnh minh họa
Ghép O.Slider và Main Shaft	
Lấy O.Slider	

Lấy Screw



**Khoan
bằng
Driver điện**



Kết quả kiểm thử trên bộ mẫu ảnh mới không có trong tập dữ liệu test set cho thấy mô hình best.pt đã có khả năng nhận diện chính xác các thao tác then chốt trong quá trình lắp ráp O.Slider như:

- Lấy linh kiện O.slider có label “lkl” và confidence là 0.92.
- Lấy linh kiện Screw có label “lkn” ” và confidence là 0.89..

- Khoan có label “khoan” ” và confidence là 0.88.
- Ghép O.Slider và Main Shaft có label “as1kl” ” và confidence là 0.91.

Hầu hết các thao tác đều được nhận diện với độ chính xác cao, bounding box được vẽ đúng vị trí và nhãn được gán chính xác. Đây là minh chứng cho thấy mô hình đã học được đặc trưng hành động từ tập dữ liệu nhỏ nhưng được dán nhãn chất lượng.

Tuy nhiên, để đảm bảo tính tổng quát trong thực tế, mô hình cần được đánh giá sâu hơn về độ chính xác (precision), độ bao phủ (recall),...Những nội dung này sẽ được trình bày chi tiết trong phần 4.4.

4.4 Đánh giá mô hình

Trong bối cảnh phát triển mạnh mẽ của các mô hình học máy và trí tuệ nhân tạo, việc đánh giá hiệu suất của các model đóng vai trò quan trọng nhằm đảm bảo tính khả thi và ứng dụng thực tế. Bài đánh giá này sẽ tập trung phân tích các chỉ số quan trọng như độ chính xác dương (precision), độ đo F1 (F1-score), cùng với khả năng xử lý dữ liệu nhiễu và tốc độ suy luận của model. Thông qua đó, chúng ta có thể nhận định rõ ưu điểm, hạn chế và đề xuất hướng tối ưu để nâng cao hiệu quả mô hình trong các bài toán cụ thể.

4.4.1 Đánh giá Model theo Precision, Recall và F1.

Bảng 4-3 được thống kê từ kết quả train model thu thập được biểu dưới đây:

Bảng 4-3: Thống kê kết quả đo được từ model

Class	Precision	Recall	F1-score
All	0.993	0.975	0.984
as1kl	0.993	1.000	0.996
khoan	0.990	1.000	0.995
lkl	0.994	0.900	0.945
lkn	0.995	1.000	0.998

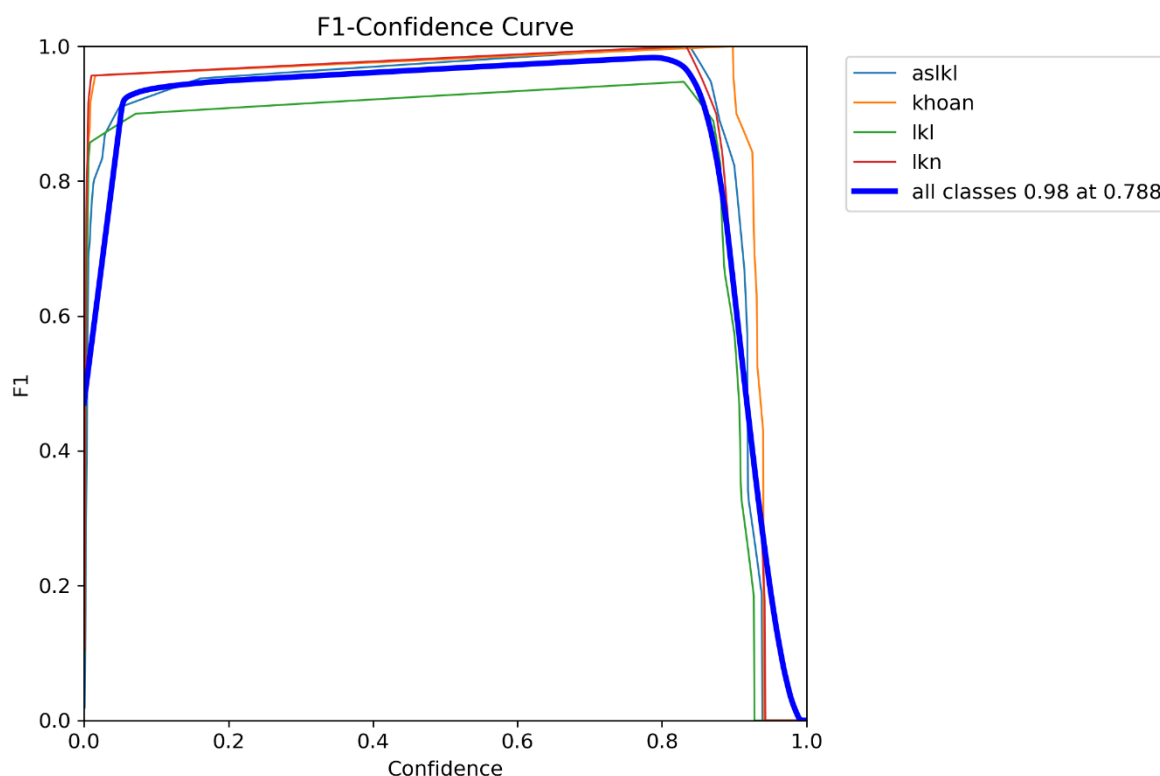
Từ Bảng 4-3 trên nhận thấy rằng:

Precision phản ánh tỷ lệ dự đoán đúng trong số các dự đoán mô hình đưa ra. Mô hình đạt Precision rất cao trên toàn bộ các lớp, với giá trị trung bình là 0.993. Các lớp as1kl, khoan, lkl và lkn đều có Precision xấp xỉ hoặc vượt 0.99, cho thấy tỷ lệ báo sai (false positives) là rất thấp

Recall đo lường khả năng phát hiện đầy đủ các đối tượng thực tế. Mô hình đạt Recall trung bình là 0.975. Các lớp as1kl, khoan, và lkn đều đạt Recall tuyệt đối (1.000), phản ánh rằng mô hình không bỏ sót đối tượng nào thuộc các lớp này. Tuy nhiên, lớp lkl chỉ đạt Recall 0.900, cho thấy vẫn còn một số đối tượng bị bỏ sót trong lớp này.

F1-score là chỉ số tổng hợp giữa Precision và Recall, giúp đánh giá hiệu quả cân bằng giữa độ chính xác và độ bao phủ. Giá trị trung bình của F1-score toàn bộ là 0.984, cho thấy mô hình hoạt động rất tốt và ổn định. Các lớp có F1-score vượt 0.99 (as1kl, khoan, lkn) khẳng định tính nhất quán của mô hình. Riêng lớp lkl có F1 thấp hơn (0.945)

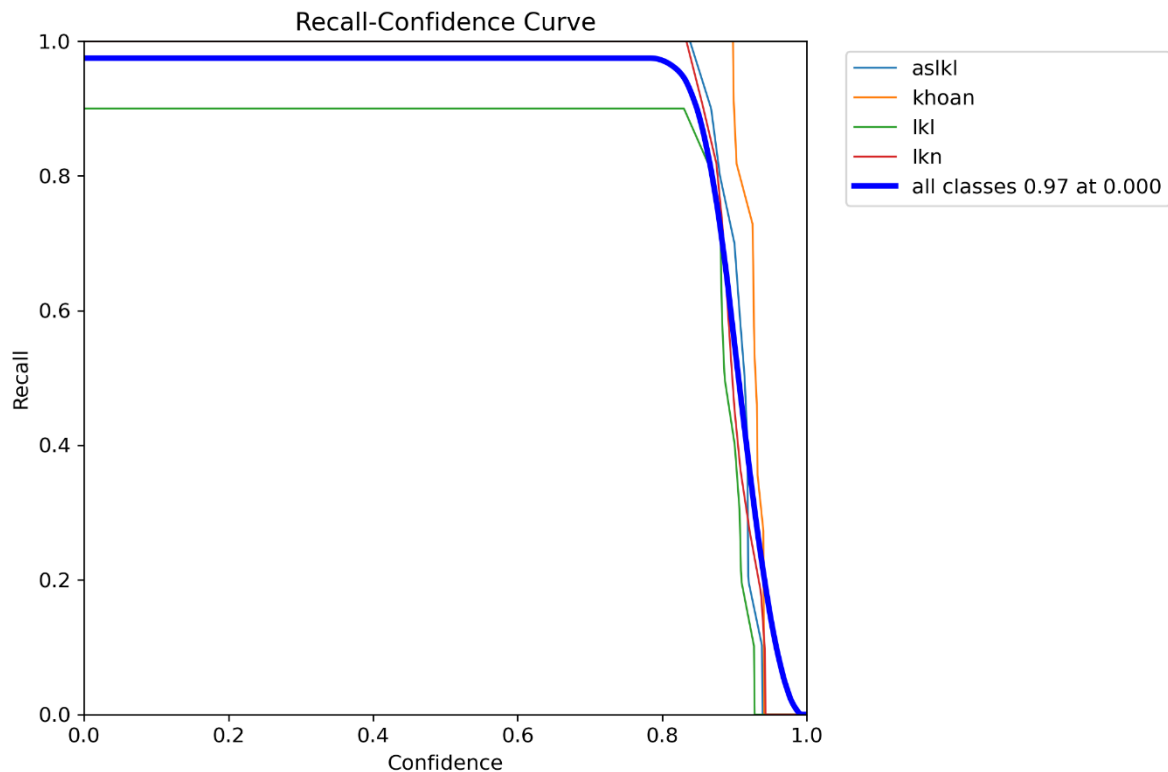
4.4.2 Phân tích F1-Confidence Curve



Hình 4-10: Biểu đồ tương quan giữa F1-Confidence

Biểu đồ F1-Confidence Curve cho thấy F1-score của tập hợp tất cả các lớp (all classes) đạt giá trị cao nhất là 0.98 tại ngưỡng confidence 0.788. Các lớp riêng lẻ (as1kl, khoan, lkl, lkn) đều duy trì F1-score trên 0.9 trong khoảng confidence từ 0 đến 0.8, với xu hướng giảm nhẹ sau ngưỡng này, đặc biệt lớp lkl giảm nhanh hơn. Điều này cho thấy mô hình đạt hiệu suất tối ưu về mặt cân bằng giữa precision và recall tại ngưỡng confidence 0.788, nhưng hiệu suất có thể giảm khi yêu cầu độ tin cậy cao hơn.

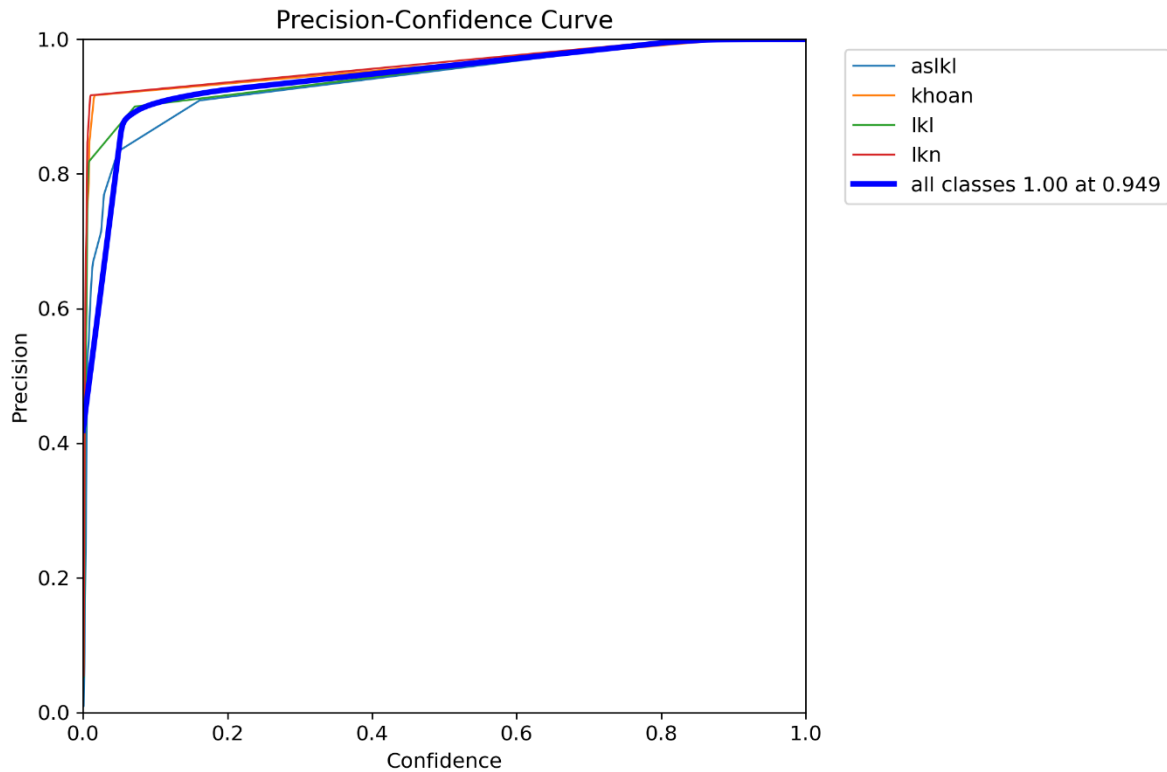
4.4.3 Phân tích Recall-Confidence Curve



Hình 4-11: Biểu đồ tương quan giữa Recall và Confidence

Recall của tất cả các lớp đạt mức 0.97 tại ngưỡng confidence 0.0 và duy trì trên 0.9 cho đến ngưỡng 0.6, sau đó giảm mạnh xuống gần 0 tại ngưỡng confidence 1.0. Trong đó, lớp lkl có mức giảm recall rõ rệt nhất (xuống dưới 0.2 tại confidence 1.0), trong khi các lớp khác (askl, khoan, lkn) duy trì recall tốt hơn ở ngưỡng trung bình. Kết quả này chỉ ra rằng mô hình có khả năng phát hiện tốt các mẫu dương tính ở ngưỡng confidence thấp, nhưng hiệu suất recall giảm đáng kể khi ngưỡng tin cậy tăng cao, đặc biệt đối với lớp lkl.

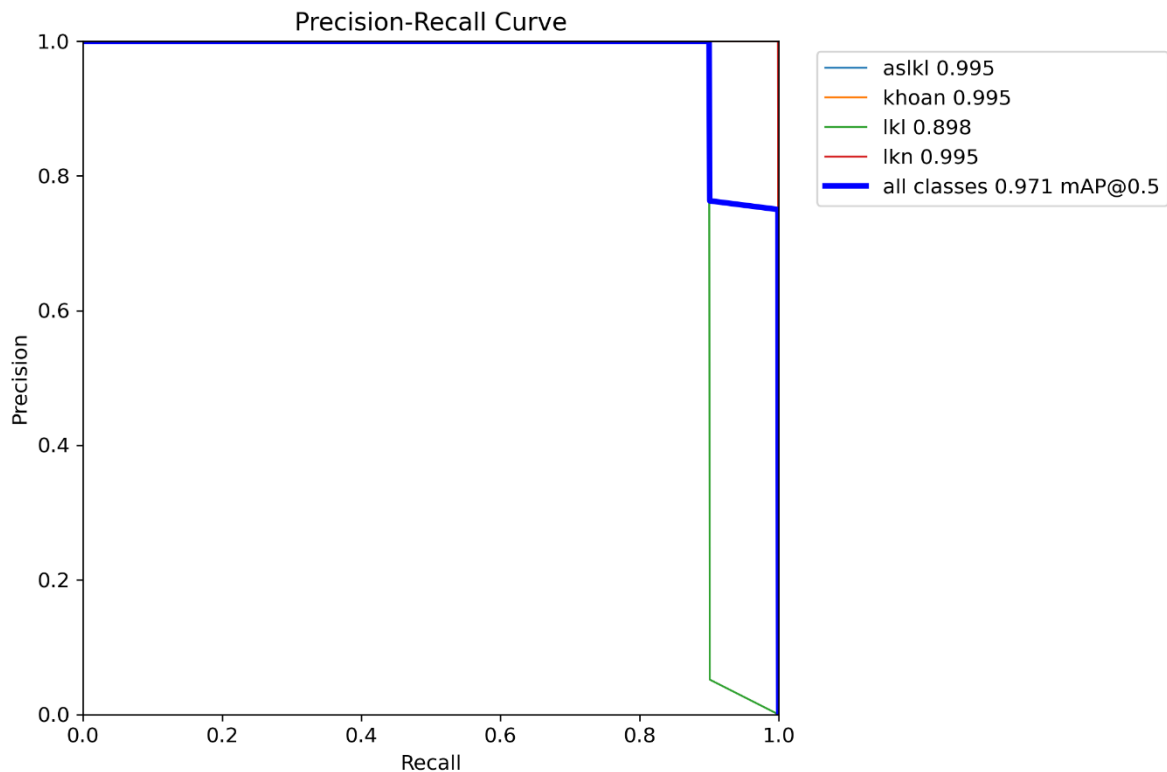
4.4.4 Phân tích Precision-Confidence Curve



Hình 4-12: Biểu đồ tương quan giữa Precision và Confidence

Precision của tất cả các lớp đạt giá trị tối đa 1.0 tại ngưỡng confidence 0.949 và duy trì ở mức cao (trên 0.9) trên toàn dải confidence. Các lớp riêng lẻ ghi nhận precision ấn tượng: askl, khoan và lkn đạt 0.995, trong khi lkl đạt 0.898. Precision tăng dần theo confidence, chứng minh rằng mô hình có khả năng dự đoán chính xác các mẫu dương tính, đặc biệt ở ngưỡng tin cậy cao, với tỷ lệ sai sót loại I (false positives) rất thấp.

4.4.5 Phân tích Precision-Recall Curve



Hình 4-13: Biểu đồ tương quan giữa Precision và Recall

Biểu đồ Precision-Recall Curve cho thấy precision duy trì ở mức 1.0 cho đến khi recall đạt khoảng 0.8, sau đó giảm nhẹ. Giá trị mAP@0.5 của tất cả các lớp là 0.971, với các lớp riêng lẻ đạt: askl (0.995), khoan (0.995), lkn (0.995), và lkl (0.898). Kết quả này phản ánh khả năng cân bằng tốt giữa precision và recall, với hiệu suất tổng thể cao. Tuy nhiên, lớp lkl có mAP thấp hơn, cho thấy hiệu suất chưa đồng đều giữa các lớp.

4.5 Đánh giá trên tập kiểm thử (Test set)

Để đảm bảo tính tổng quát hóa và hiệu quả thực tế của mô hình best.pt trong việc giám sát quá trình lắp ráp O.Slider tại bộ phận tiền lắp ráp V159, Công ty TNHH Daiwa Việt Nam, việc đánh giá mô hình trên tập kiểm thử (test set) là một bước quan trọng. Phần này sẽ trình bày chi tiết cách thức xây dựng, thực hiện và phân tích kết quả đánh giá trên tập kiểm thử, nhằm xác minh khả năng nhận diện thao tác của mô hình trong các điều kiện thực tế.

4.5.1 Tập dữ liệu kiểm thử (Test set)

Test set là tập dữ liệu hoàn toàn độc lập so với dữ liệu huấn luyện (train) và xác thực (validation). Mục tiêu của test set là giúp kiểm tra khách quan hiệu năng cuối cùng của mô hình sau khi đã huấn luyện xong. Đây là bước quan trọng để đảm bảo rằng mô

hình không chỉ học thuộc dữ liệu huấn luyện, mà còn có thể áp dụng tốt cho các tình huống thực tế chưa từng gặp.

Tập test bao gồm 20 ảnh và label tương ứng, trong đó mỗi ảnh chứa các tình huống thực tế trong dây chuyền lắp ráp. Các lỗi trong ảnh đã được gán nhãn thủ công chính xác, tuân theo định dạng YOLO. Bộ dữ liệu này được khai báo trong tệp new.yaml chứa thông tin đường dẫn ảnh và phân loại các lớp lỗi

```
from ultralytics import YOLO
import os

# 1. Load model YOLO đã train
model_path='/content/drive/MyDrive/ML_DAIWA/runs/runs/detect/train/weights/best.pt'

model = YOLO(model_path)

# 2. Validation trên tập test
yaml_path = '/content/drive/MyDrive/ML_DAIWA/new.yaml'
results = model.val(data=yaml_path)

# 3. Tính các chỉ số tổng quát
mp = results.box.mp
```

Hình 4-14: Minh họa thuật toán để kiểm thử

Trong đoạn mã kiểm thử, trước tiên, thư viện ultralytics được gọi để sử dụng các chức năng có sẵn của mô hình YOLO11n. Sau đó, mô hình đã huấn luyện (best.pt) được nạp vào chương trình thông qua lớp YOLO. Lệnh model.val(data=yaml_path) được sử dụng để thực hiện đánh giá mô hình trên tập test. Hàm này sẽ tự động chạy mô hình trên toàn bộ ảnh kiểm thử, so sánh các kết quả dự đoán với nhãn thật, và tính toán các chỉ số đánh giá quan trọng như Precision, Recall, F1

4.5.2 Kết quả đánh giá

Sau khi hoàn tất quá trình huấn luyện và kiểm thử mô hình nhận diện, hệ thống được tiến hành đánh giá trên tập dữ liệu kiểm thử (test set) nhằm xác định mức độ chính xác và độ tin cậy trong việc nhận diện thao tác lắp ráp. Việc đánh giá dựa trên các chỉ số hiệu suất chính như Precision, Recall, và F1-score – là những chỉ số phản ánh khả năng mô hình phát hiện đúng thao tác cần nhận diện và hạn chế báo sai.

Kết quả thu được từ quá trình đánh giá không chỉ phản ánh chất lượng của mô hình trên dữ liệu mới mà còn là cơ sở quan trọng để xác định mức độ phù hợp khi triển khai

thực tế trong môi trường sản xuất tại bộ phận V159. Dưới đây là các chỉ số chi tiết đo được từ mô hình khi áp dụng lên tập kiểm thử.

Bảng 4-4: Bảng kết quả của bộ dữ liệu test set

Class	Precision	Recall	F1-Score
All	0.9879	0.9583	0.9729
Aslkl	0.9898	0.8333	0.9048
Khoan	0.9839	1.0000	0.9919
Lkl	0.9872	1.0000	0.9937
Lkn	0.9907	1.0000	0.9951

Từ bảng 4-4 thống kê thu được từ quá trình đánh giá kết quả trên tập dữ liệu testset nhận thấy:

Mô hình đạt được hiệu suất tổng thể rất cao trên tập test, với:

Precision = 0.9879, chứng tỏ khả năng hạn chế tối đa dự đoán sai.

Recall = 0.9583, cho thấy mô hình phát hiện được phần lớn các đối tượng mục tiêu trong dữ liệu.

F1-Score = 0.9729, phản ánh sự cân bằng tốt giữa độ chính xác và độ bao phủ, phù hợp với các ứng dụng thực tế yêu cầu độ tin cậy cao.

Lớp Khoan, Lkl, Lkn: Đạt Recall = 1.0000 và F1-Score > 0.99, chứng tỏ mô hình gần như hoàn hảo trong việc nhận diện các lớp này.

Lớp Aslkl: Recall thấp hơn (0.8333), dù Precision rất cao (0.9898), cho thấy mô hình có xu hướng bỏ sót một số trường hợp thuộc lớp này.

Từ các kết quả đánh giá trên tập kiểm thử, có thể nhận thấy rằng mô hình đạt hiệu suất rất tốt, với các chỉ số precision, recall và F1-score đều vượt mức 0.97 ở cấp độ tổng thể. Điều này cho thấy mô hình không chỉ có khả năng nhận diện chính xác các thao tác then chốt trong quá trình lắp ráp O.Slider, mà còn đảm bảo độ tin cậy cao khi triển khai trong môi trường thực tế. Như vậy, mô hình có thể dễ dàng tích hợp vào hệ thống giám sát thời gian thực tại dây chuyền sản xuất, với kỳ vọng góp phần giảm thiểu lỗi thao tác, nâng cao chất lượng sản phẩm và tối ưu hóa quy trình vận hành.

4.6 Kiến trúc tổng thể

Việc xây dựng phần mềm giám sát đóng vai trò trung tâm trong hệ thống nhận diện thao tác lắp ráp nhằm nâng cao chất lượng quá trình sản xuất tại công ty Daiwa Việt Nam. Phần mềm này không đơn thuần chỉ là một công cụ hiển thị hình ảnh mà còn tích hợp chặt chẽ các thành phần như mô hình trí tuệ nhân tạo, thư viện xử lý hình ảnh và

giao tiếp thiết bị phần cứng (Arduino), nhằm tạo thành một hệ thống tự động hóa, hỗ trợ giám sát thông minh và cảnh báo lỗi theo thời gian thực.

a. Các chức năng chính:

- Nhận luồng hình ảnh trực tiếp từ webcam và xử lý theo thời gian thực.
- Tích hợp mô hình được huấn luyện từ yolo11n để phát hiện các thao tác then chốt trong lắp ráp.
- Cảnh báo khi phát hiện thao tác sai hoặc có khả năng dẫn đến lỗi sản phẩm.
- Giao tiếp với vi điều khiển Arduino để kích hoạt đèn/còi cảnh báo.
- Giao diện thân thiện, trực quan với người dùng.
- Ghi log dữ liệu và thống kê các lỗi được phát hiện.

b. Thành phần Input và Output

Phần mềm giám sát sản xuất được phát triển nhằm tự động hóa quá trình kiểm tra chất lượng sản phẩm thông qua các thành phần chính sau:

Input:

Đường dẫn file mô hình YOLO (best.pt), ảnh giao diện.

Dữ liệu từ webcam.

Cổng kết nối Arduino.

Output:

Giao diện Tkinter hiển thị trạng thái, số sản phẩm hoàn thành, điểm, lỗi NG.

File Excel `thong_ke.xlsx` chứa log dữ liệu.

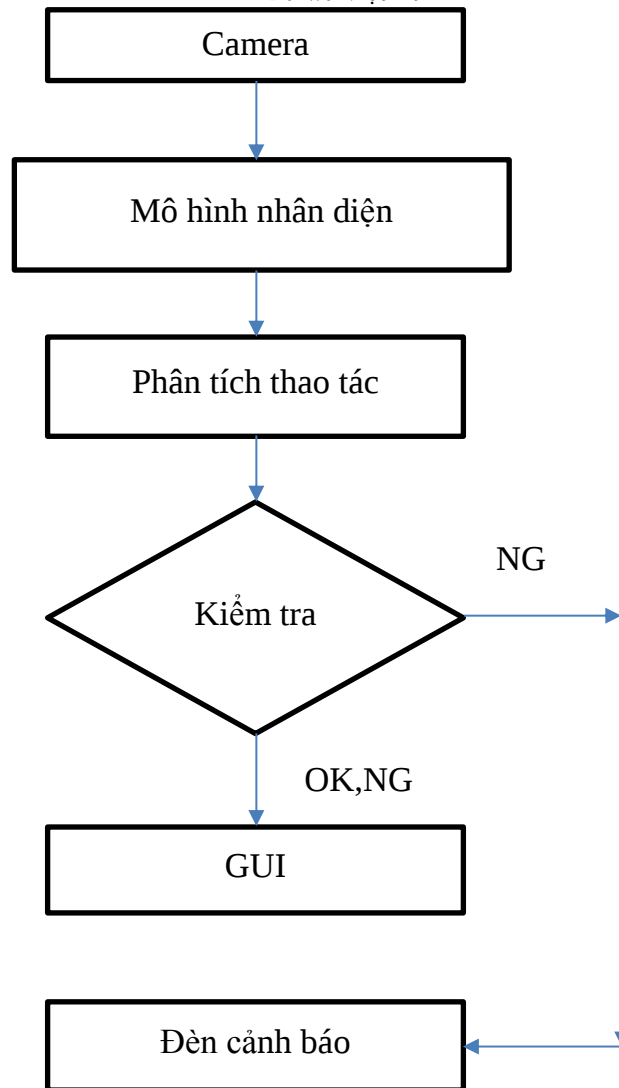
Biểu đồ cột thống kê số sản phẩm và lỗi NG trong ngày

Tín hiệu điều khiển đèn NG qua Arduino.

c. Cấu trúc phần mềm

Phần mềm được xây dựng theo mô hình phân lớp rõ ràng nhằm đảm bảo tính mở rộng, dễ bảo trì và tích hợp với phần cứng. Cấu trúc tổng thể bao gồm 3 lớp chính:

- Lớp giao diện (GUI): sử dụng Tkinter để hiển thị video, trạng thái hệ thống, nút điều khiển
- Lớp xử lý logic: mô hình YOLOv11n xử lý ảnh, xác định nhãn, kiểm tra quy trình logic thao tác
- Lớp giao tiếp phần cứng: dùng PySerial để gửi tín hiệu điều khiển đèn tới Arduino



Hình 4-15: Sơ đồ kiến trúc tổng thể

Sơ đồ trên mô tả quy trình hoạt động chính của hệ thống. Dữ liệu hình ảnh được thu nhận từ camera, sau đó xử lý qua mô hình nhận diện để trích xuất các nhãn thao tác. Thông tin từ mô hình được kiểm tra theo logic chu trình. Nếu phát hiện sai sót (NG), phần mềm sẽ gửi tín hiệu cảnh báo xuống thiết bị phần cứng (Arduino) để hiển thị bằng đèn hoặc còi. Đồng thời, dữ liệu và trạng thái cũng được cập nhật trên giao diện người dùng để dễ theo dõi và vận hành.

4.7 Thuật toán của phần mềm

Để xây dựng phần mềm giám sát quy trình sản xuất với chức năng nhận diện sản phẩm, đếm chu trình và phát hiện lỗi NG, thuật toán được triển khai theo các bước sau:

- Khởi tạo chương trình: Thiết lập giao diện Tkinter, biến trạng thái và kết nối phần cứng (Arduino).

- Thiết lập giao diện : Tạo nền, logo, các nút điều khiển và hiển thị thông số thời gian thực.
 - Tải mô hình và nhận diện : Sử dụng Yolo (best.pt) để phát hiện đối tượng từ webcam, xử lý khung hình với tốc độ ổn định.
 - Xử lý logic chu trình: Tính sản phẩm, kiểm tra lỗi NG, cập nhật dữ liệu và ghi log.
 - Kết thúc và lưu kết quả: Giải phóng tài nguyên, xuất dữ liệu thống kê ra file Excel.
 - Hiển thị kết quả: Vẽ biểu đồ thống kê sản phẩm và lỗi trong ngày.
- Các phần dưới đây sẽ trình bày rõ hơn về thuật toán trong phần mềm

4.7.1 Khởi tạo chương trình:

Tạo cửa sổ Tkinter, đặt tiêu đề: "Phần mềm giám sát sản xuất".

Tải và đặt biểu tượng (2.png) cho cửa sổ; nếu thất bại, in lỗi.

Khởi tạo biến trạng thái:

cap: Webcam (None).

running: Trạng thái chạy (False).

cycle_points: Điểm chu trình (0).

cycle_count: Số sản phẩm hoàn thành (0).

ng_count: Số lỗi NG (0).

ng_triggered: Cờ lỗi NG (False).

seen_labels_in_cycle: Tập hợp nhãn trong chu trình (rỗng).

score_map: nhãn (lkl, askl, lkn, khoan) sang điểm (1).

log_data: Danh sách log (rỗng).

Kết nối Arduino qua cổng COM3:

Nếu thành công, in thông báo kết nối.

Nếu thất bại, đặt arduino = None, in lỗi.

4.7.2 Thiết lập giao diện và nút bấm

Tải hình nền (1.png), resize về 800x600, đặt làm nền; nếu thất bại, dùng nền trắng.

Tải logo (2.png), resize về 100x100, đặt ở góc trên trái; nếu thất bại, in lỗi.

Tạo nhãn hiển thị:

Trạng thái: "Trạng thái: Đang dừng" (x=300, y=50).

Số sản phẩm: "Số Sản Phẩm Hoàn Thành: 0" (x=300, y=100).

Điểm: "Point: 0" (x=300, y=150).

Lỗi NG: "NG: 0" (x=300, y=200).

"Bắt đầu" (x=300, y=300, màu xanh, gọi hàm phát hiện).

"Dừng" (x=500, y=300, màu đỏ, ban đầu vô hiệu hóa, gọi hàm dừng).

"Lưu Log" (x=400, y=380, màu xanh dương, gọi hàm lưu log).

"Hiển thị đồ thị" (x=400, y=460, màu tím, gọi hàm vẽ biểu đồ).

4.7.3 Tải mô hình và bắt đầu nhận diện

Tải mô hình từ best.pt.

Nếu thất bại, hiển thị lỗi và vô hiệu hóa nút "Bắt đầu".

Nếu chưa chạy (running = False):

Mở webcam (640x480).

Nếu không mở được, hiển thị lỗi và thoát.

Đặt running = True, vô hiệu hóa nút "Bắt đầu", kích hoạt nút "Dừng".

Cập nhật nhãn trạng thái: "Trạng thái: Đang chạy".

Chạy luồng phát hiện riêng.

Khởi tạo frame_skip = 2, frame_count = 0.

Trong khi running = True:

Đọc khung hình từ webcam; nếu thất bại, hiển thị lỗi, dừng và thoát.

Tăng frame_count. Nếu frame_count chia hết cho frame_skip:

Chạy YOLO trên khung hình

Tạo tập hợp labels_in_frame cho nhãn phát hiện.

Với mỗi hộp chỉ lấy (confidence $\geq$ 0.788):

Lấy nhãn, vẽ hộp và nhãn lên khung hình, thêm nhãn vào labels_in_frame.

4.7.4 Xử lý logic chu trình:

Nếu nhãn lkl: Đặt cycle_points = 1, seen_labels_in_cycle = {'lkl'}, ng_triggered = False, ghi log.

Với mỗi nhãn trong labels_in_frame (thuộc score_map, chưa có trong seen_labels_in_cycle):

Tăng cycle_points theo score_map, thêm nhãn vào seen_labels_in_cycle, ghi log.

Nếu chưa có lỗi NG:

Nhãn aslkl và cycle_points != 2: Tăng ng_count, đặt ng_triggered = True, ghi log, bật đèn NG (gửi '1' qua Arduino, tắt sau 10 giây).

Nhãn lkn và cycle_points != 3: Tăng ng_count, đặt ng_triggered = True, ghi log, bật đèn NG.

Nếu cycle_points $\geq$ 4:

Tăng `cycle_count`, ghi log, đặt lại `cycle_points = 0`, xóa `seen_labels_in_cycle`, `ng_triggered = False`.

Cập nhật giao diện: Số sản phẩm, điểm, lỗi NG.

Hiển thị trên khung hình: Số sản phẩm (nếu > 0), điểm, lỗi NG (nếu > 0).

Hiển thị khung hình qua OpenCV ("WEBCAM").

Nếu nhấn phím 'q', dừng chương trình.

Điều chỉnh tốc độ vòng lặp (~30 FPS).

4.7.5 Kết thúc nhận diện và lưu kết quả

Đặt `running = False`, kích hoạt nút "Bắt đầu", vô hiệu hóa nút "Dừng".

Cập nhật nhãn trạng thái: "Trạng thái: Đang dừng".

Giải phóng webcam, đóng cửa sổ OpenCV.

Tạo DataFrame từ `log_data` với cột: ["Time", "Process", "Label", "Point", "Cycle_Count", "NG_Count"].

Lưu vào `thong_ke.xlsx` (engine `openpyxl`).

Hiển thị thông báo thành công hoặc lỗi.

4.7.6 Hiển thị kết quả:

Đọc `thong_ke.xlsx`. Nếu thiếu cột hoặc file, hiển thị lỗi.

Tính số sản phẩm và lỗi NG trong ngày (giá trị cuối trừ giá trị trước ngày).

Vẽ biểu đồ cột:

Trục x: ["Số sản phẩm hoàn thành", "Số lỗi NG"].

Trục y: [số sản phẩm, số lỗi NG].

Tiêu đề: "Tổng kê trong ngày"

Thêm lưới trục y, hiển thị biểu đồ.

Nếu có lỗi, hiển thị thông báo phù hợp.

4.8 Giao diện phần mềm

Giao diện phần mềm giám sát quá trình lắp ráp O.Slider tại bộ phận tiền lắp ráp V159, Công ty TNHH Daiwa Việt Nam, đóng vai trò trung tâm trong việc hiện thực hóa hệ thống giám sát thông minh, nhằm nâng cao chất lượng sản phẩm và giảm thiểu lỗi trong sản xuất. Được xây dựng dựa trên thư viện Tkinter của Python, giao diện phần mềm không chỉ cung cấp khả năng hiển thị luồng video trực tiếp từ webcam với các nhãn thao tác được nhận diện bởi mô hình, mà còn tích hợp các chức năng điều khiển, cập nhật trạng thái thời gian thực, và ghi log dữ liệu. Hệ thống giao tiếp với vi điều khiển Arduino để kích hoạt cảnh báo lỗi, đồng thời đảm bảo tính trực quan và thân thiện với

người vận hành. Phần này sẽ trình bày chi tiết về thiết kế, các thành phần, và quy trình hoạt động của giao diện phần mềm, nhấn mạnh vai trò của nó trong việc hỗ trợ giám sát tự động.



Hình 4-16: Giao diện phần mềm

4.8.1 Chức năng của từng phần trong giao diện:

Khu vực hiển thị trạng thái (Trung tâm phía trên):

Trạng thái: Đang dừng/Đang chạy:

- Chức năng: Hiển thị trạng thái hiện tại của hệ thống.
- Ý nghĩa: Giúp người vận hành xác định hệ thống có đang hoạt động hay không.

Số Sản Phẩm Hoàn Thành:

- Chức năng: Hiển thị tổng số chu kỳ lắp ráp hoàn thành (cycle_count).
- Ý nghĩa: Theo dõi năng suất sản xuất.

Point:

- Chức năng: Hiển thị điểm của chu trình hiện tại (cycle_points).
- Ý nghĩa: Phản ánh tiến độ lắp ráp trong một chu trình.

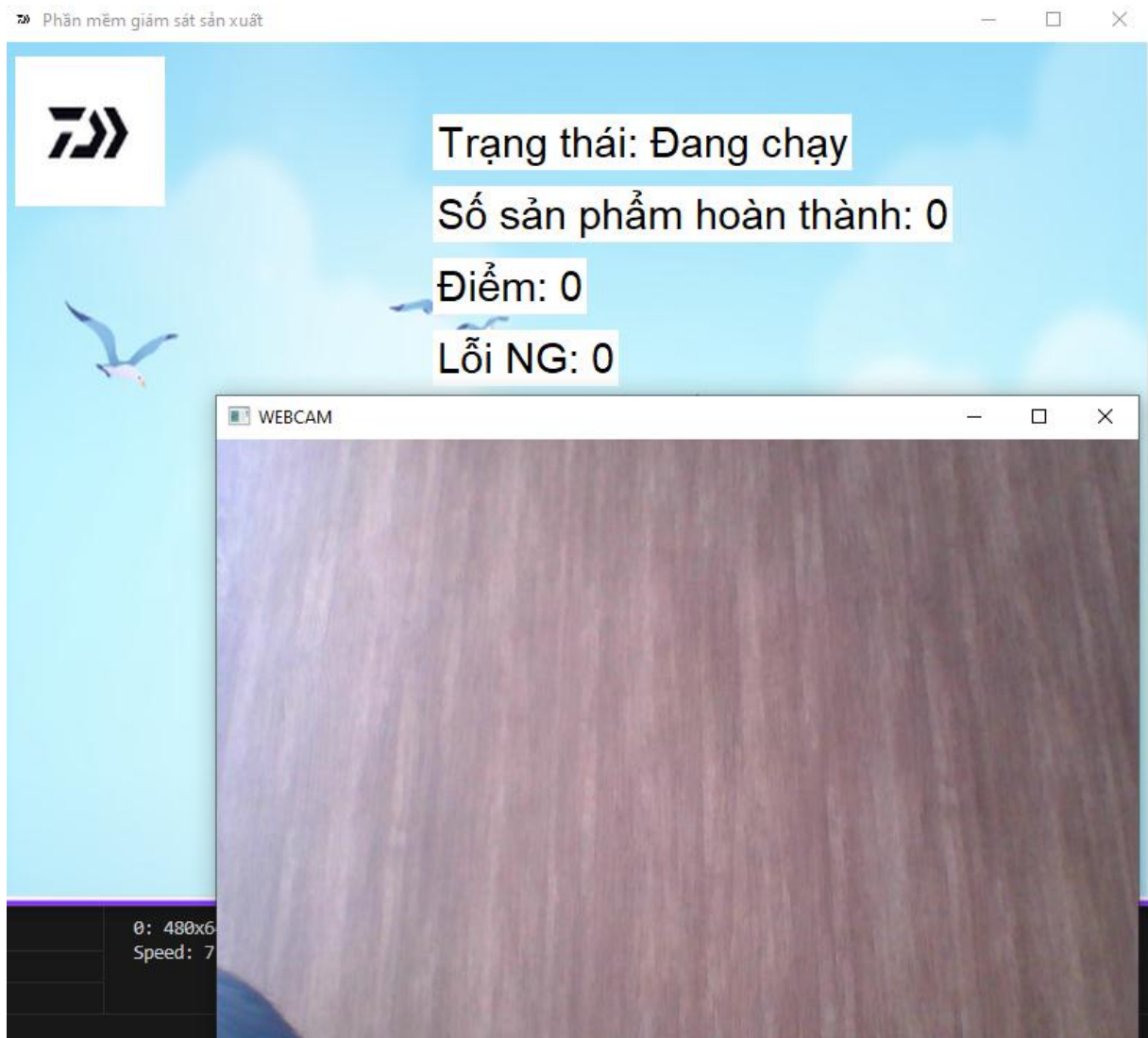
NG:

- Chức năng: Hiển thị số lỗi NG phát hiện được (ng_count).
- Ý nghĩa: Cung cấp thông tin về tỷ lệ lỗi để xử lý kịp thời.

Trong phần trình bày tiếp theo, nghiên cứu sẽ phân tích và làm rõ đặc điểm giao diện của từng nút điều khiển trong phần mềm. Cụ thể, mỗi thành phần tương tác sẽ được mô tả chi tiết về vị trí, chức năng, nhằm cung cấp cơ sở khoa học cho việc đánh giá trải nghiệm người dùng

Nút "Bắt đầu" (Màu xanh lá):

Khi bấm vào nút Bắt đầu sẽ hiện thị giao diện webcam được mô tả ở hình 4-17:



Hình 4-17: Giao diện webcam của hệ thống

Chức năng: Khởi động hệ thống giám sát, mở webcam, kích hoạt mô hình best.pt để nhận diện thao tác, và bắt đầu kiểm tra logic chu trình.

Ý nghĩa: Cho phép bắt đầu giám sát tự động, đảm bảo phát hiện lỗi thời gian thực.

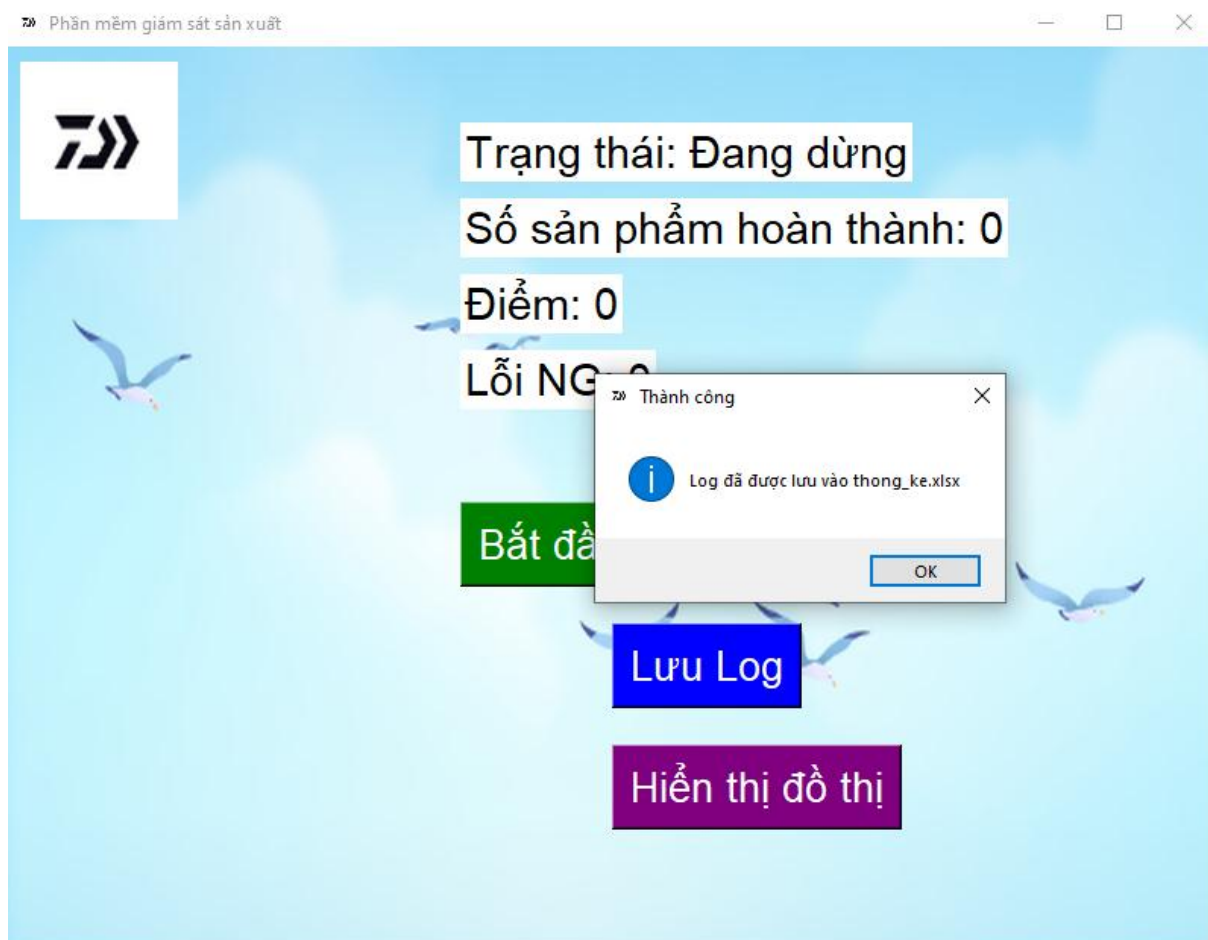
Nút "Dừng" (Màu đỏ):

Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam

Chức năng: Tạm dừng hệ thống, đóng webcam, dừng nhận diện, và cập nhật trạng thái "Đang dừng".

Nút "Lưu Log" (Màu xanh dương):

Khi bấm vào nút Lưu Log sẽ hiện thị giao diện lưu log được mô tả ở hình 4-18:



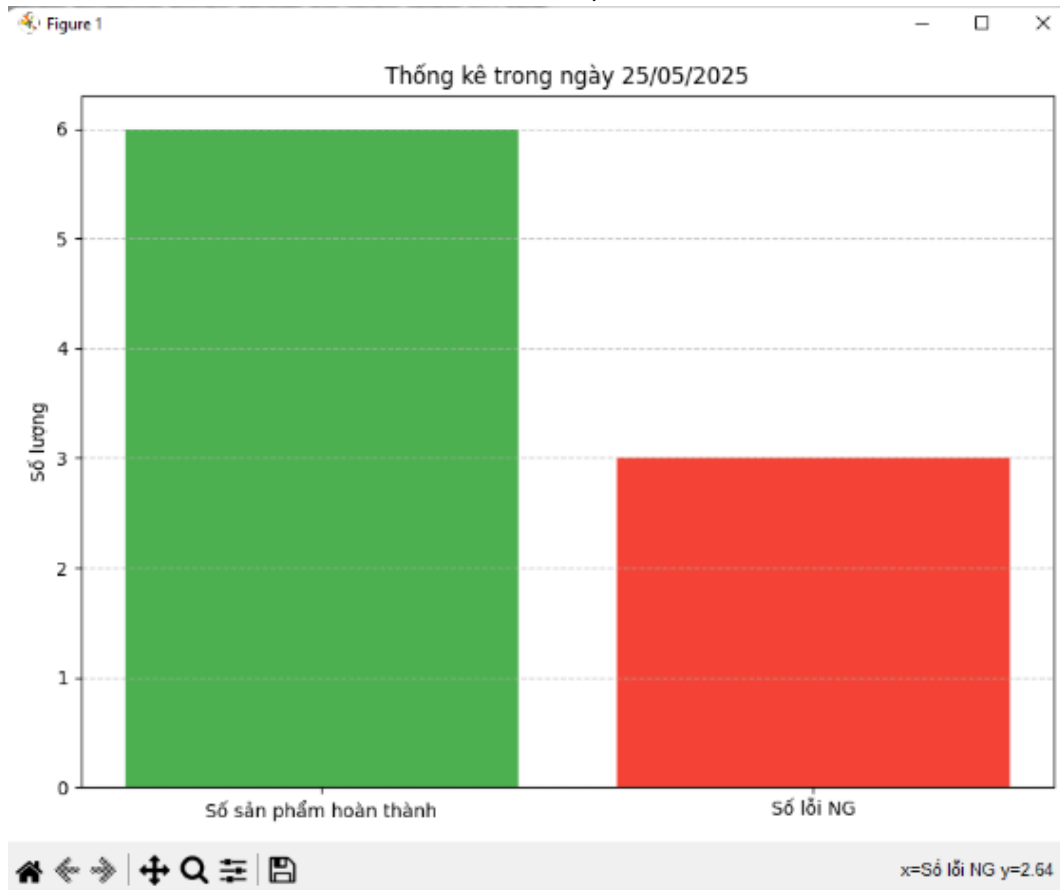
Hình 4-18: Giao diện lưu log của phần mềm giám sát.

Hình 4-18. Mô tả chức năng Ghi dữ liệu log (thời gian, sự kiện, nhãn, điểm, số chu kỳ, số lỗi) vào file Excel (thong_ke.xlsx) bằng Pandas.

Ý nghĩa: Lưu trữ dữ liệu để phân tích và cải tiến quy trình.

Nút "Hiện thị đồ thị" (Màu tím):

Khi bấm vào nút Hiện thị đồ thị sẽ hiện thị giao diện kết quả được trực quan hóa được mô tả ở hình 19:

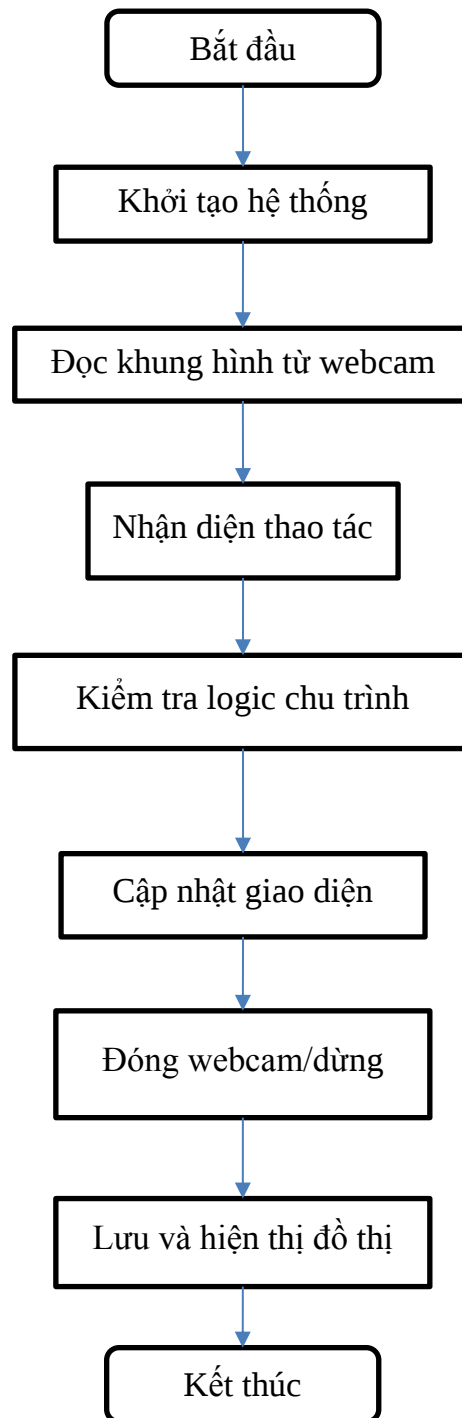


Hình 4-19: Giao diện hiển thị đồ thị

Chức năng: Hiển thị biểu đồ cột thống kê số sản phẩm hoàn thành và số lỗi NG trong ngày, sử dụng Matplotlib.

Ý nghĩa: Trực quan hóa dữ liệu để hỗ trợ quản lý và đánh giá hiệu suất sản xuất.

4.8.2 Cách sử dụng phần mềm



Hình 4-20: Quy trình thao tác phần mềm

Bắt đầu: Khởi chạy phần mềm.

Khởi tạo hệ thống: Tải mô hình YOLO11n (best.pt).

- Kết nối webcam và Arduino.
- Khởi tạo biến: cycle_count, cycle_points, ng_count, ng_triggered, seen_labels_in_cycle.

- Thiết lập giao diện Tkinter với logo, hình nền, và các thành phần GUI.

Nhấn nút "Bắt đầu": Mở webcam, bắt đầu nhận diện.

Đọc khung hình từ webcam:

- Sử dụng OpenCV để đọc video.
- Kiểm tra: Nếu không đọc được, hiển thị lỗi và dừng.

Nhận diện thao tác bằng mô hình:

- Xử lý khung hình, nhận diện nhãn thao tác ("lkl", "aslkl", "lkn", "khoan") với ngưỡng độ tin cậy 0.788.
- Vẽ bounding box và nhãn lên khung hình.

Kiểm tra logic chu trình:

- Nếu phát hiện "lkl": Khởi tạo chu trình (cycle_points = 1).
- Nếu nhãn hợp lệ: Cộng điểm (cycle_points += score_map[label]).
- Nếu phát hiện lỗi Tăng ng_count, kích hoạt đèn/còi qua Arduino.
- Nếu cycle_points >= 4: Tăng cycle_count, reset chu trình.

Cập nhật giao diện:

- Hiển thị video với bounding box trên cửa sổ OpenCV.
- Cập nhật số liệu trên Tkinter (cycle_count, cycle_points, ng_count).

Nhấn "Dừng" hoặc "q"?:

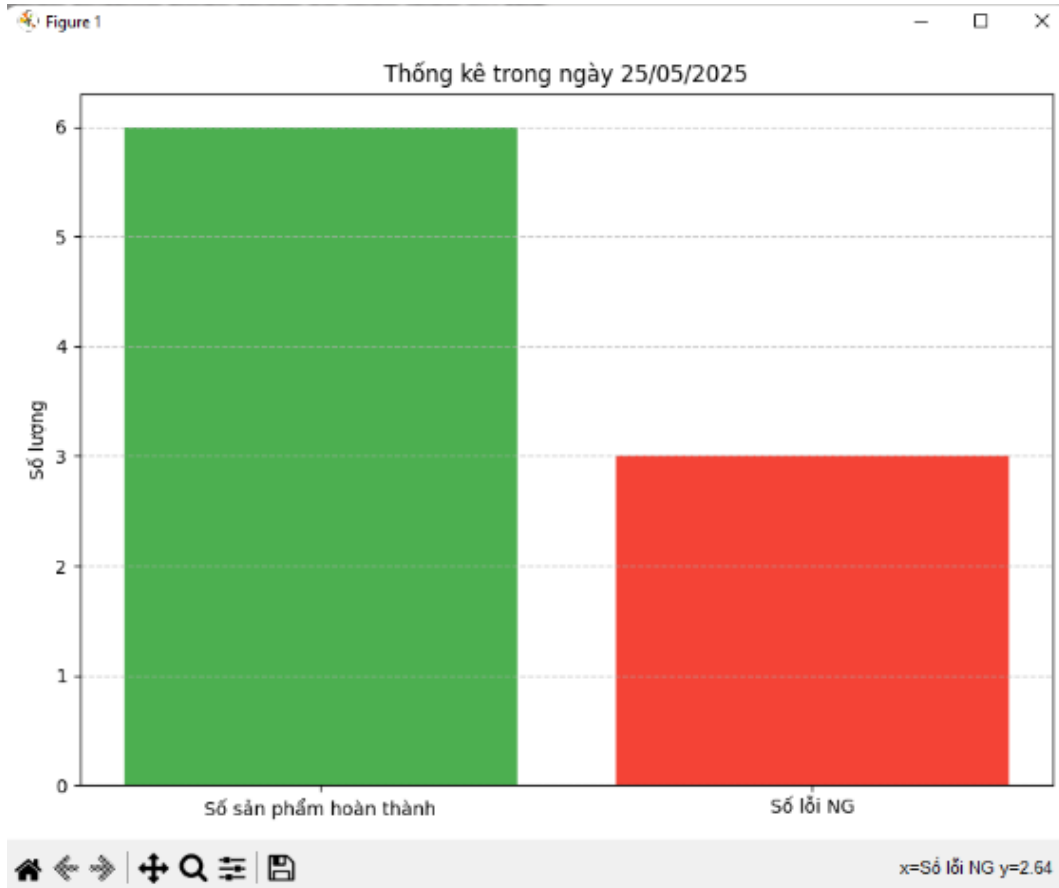
- Nếu có: Dừng hệ thống, đóng webcam.
- Nếu không: Tiếp tục đọc khung hình.

Lưu log và hiển thị đồ thị:

- Khi nhấn "Lưu Log": Ghi dữ liệu vào thong_ke.xlsx.
- Khi nhấn "Hiển thị đồ thị": Vẽ biểu đồ cột cho ngày 25/05/2025.

Kết thúc: Đóng webcam, ngắt kết nối Arduino, thoát chương trình.

Ngoài giao diện chính ra còn có giao diện biểu đồ.



Hình 4-21: Mô tả giao diện biểu đồ trong phần mềm

Chức năng hiển thị biểu đồ được thiết kế để cung cấp cái nhìn trực quan về hiệu suất sản xuất trong ngày. Biểu đồ cột sẽ thể hiện tổng số sản phẩm hoàn thành và số lỗi NG, giúp người dùng dễ dàng đánh giá tình hình sản xuất một cách nhanh chóng và hiệu quả.

4.9 Kết quả thử nghiệm

Nhằm đánh giá mức độ chính xác và ổn định của hệ thống giám sát tự động khi xử lý dữ liệu hình ảnh từ video thực tế, sẽ tiến hành xây dựng kịch bản kiểm thử với nguồn dữ liệu đầu vào là các video quay sẵn tại khu vực lắp ráp O.Slider. Việc kiểm thử với video giúp mô phỏng chân thực các tình huống thao tác lắp ráp xảy ra trong quá trình sản xuất thực tế, đồng thời cho phép hệ thống được kiểm thử lặp lại nhiều lần trong điều kiện chuẩn hóa dữ liệu.

Thông qua kiểm thử, hệ thống cần chứng minh khả năng:

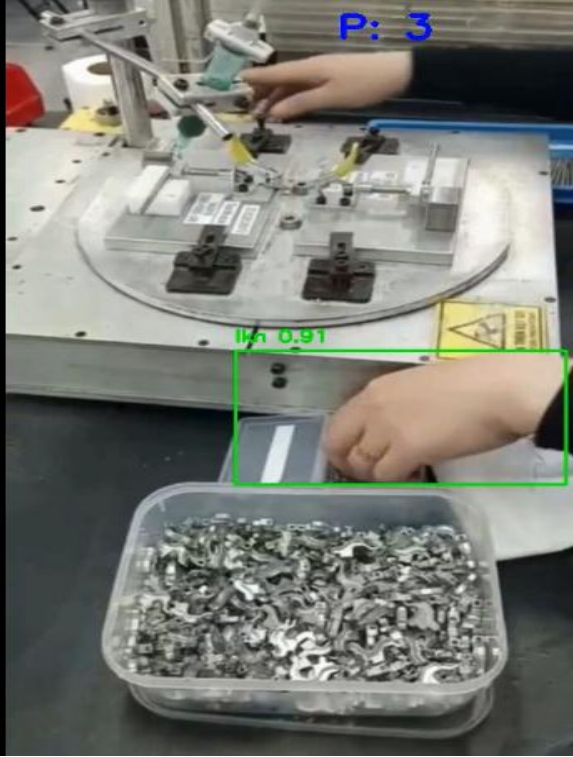
- Phát hiện chính xác các thao tác lỗi (lắp thiếu linh kiện, lắp sai vị trí).
- Đảm bảo tốc độ xử lý video liên tục, ổn định.
- Ghi nhận kết quả nhận diện để phục vụ phân tích đánh giá các chỉ số mô hình.

4.9.1 Kết quả sản phẩm OK

Kết quả kiểm thử sẽ được tổng hợp và trình bày dưới dạng bảng thống kê sau.

Bảng 4-5: Kết quả thống kê quá trình kiểm thử hệ thống

Công đoạn	Ảnh minh họa kết quả
Lấy O.Slider	
Ghép O.Slider và Main Shaft	
Lấy Screw	

	
Khoan bằng Driver điện	

Quy trình logic kiểm soát chu trình lắp ráp được thiết kế dựa trên việc theo dõi liên tục các trạng thái thao tác của công nhân thông qua giá trị P, trong đó P đại diện cho từng bước của chu trình lắp ráp O.Slider được thể hiện qua bảng 4-5. Và được diễn giải cụ thể như sau:

- Khi bắt đầu một chu trình mới, công nhân thực hiện thao tác lấy chi tiết O.Slider đầu tiên. Tại thời điểm này, hệ thống nhận diện và gán giá trị $P = 1$, đồng thời xác định rằng chu trình mới đã được khởi tạo.
- Tiếp theo, công nhân tiến hành lắp ráp linh kiện O.Slider vào vị trí. Khi thao tác này được thực hiện chính xác, hệ thống ghi nhận $P = 2$, thể hiện bước lắp ráp linh kiện hoàn tất.
- Sau đó, công nhân thực hiện thao tác lấy Screw. Khi phát hiện thao tác này, hệ thống cập nhật giá trị $P = 3$.
- Cuối cùng, công nhân thực hiện thao tác khoan hoàn thiện sản phẩm. Lúc này hệ thống ghi nhận $P = 4$, đồng nghĩa với việc toàn bộ chu trình lắp ráp đã hoàn thành đầy đủ.
- Ngay khi giá trị P đạt đến 4, hệ thống sẽ tự động tăng chỉ số sản phẩm hoàn thiện (SP) lên 1 đơn vị, đồng thời thực hiện lệnh reset để đưa P về trạng thái khởi đầu ($P = 1$), sẵn sàng cho chu trình tiếp theo.

Đáng lưu ý, trong thực tế quá trình vận hành, tại thời điểm công nhân quay trở lại thực hiện thao tác lấy O.Slider mới, hệ thống tiếp tục nhận diện và thiết lập lại giá trị $P = 1$, khởi động lại vòng lặp kiểm soát chu trình.

Cơ chế vận hành này cho phép hệ thống không những theo dõi được chính xác từng bước thao tác mà còn đảm bảo tính liên tục, đồng bộ giữa quá trình giám sát và thao tác thực tế của công nhân trên dây chuyền sản xuất. Và để xác định khi nào NG thì hệ thống sẽ được tuân sẽ thuật toán logic.

Thuật toán logic kiểm soát chu trình lắp ráp được xây dựng dựa trên nguyên tắc kiểm tra đồng thời hai yếu tố: số lượng thao tác hoàn thành (P) Cụ thể như sau:

- Khi giá trị P đạt mức 4, điều này đồng nghĩa với việc công nhân đã hoàn tất đầy đủ bốn bước chính của quy trình lắp ráp O.Slider. Do đó, hệ thống xác nhận linh kiện đã được lắp ráp đầy đủ, không xảy ra lỗi thiếu linh kiện.
- Khi $P = 2$ và đồng thời hiện thị công đoạn ghép O.slider với main shaft, hệ thống đánh giá thao tác lắp ráp đang tuân thủ đúng thứ tự quy trình, đảm bảo không có lỗi lắp sai vị trí tại bước này.
- Tương tự, khi P lần lượt đạt các giá trị 3 và 4, hệ thống tiếp tục đối chiếu với các điểm chu trình tương ứng. Nếu các giá trị này khớp nhau tại từng bước, hệ thống xác nhận công nhân đã thực hiện đúng thứ tự lắp ráp, không có sai sót về vị trí lắp linh kiện.

4.9.2 Kết quả sản phẩm NG

Ngoài ra, trong quá trình vận hành, khi hệ thống phát hiện trường hợp sản phẩm NG (lỗi), thuật toán xử lý sẽ thực hiện thêm nhánh kiểm tra và lưu trữ dữ liệu lỗi. Cụ thể, khi có NG xảy ra, luồng xử lý của hệ thống sẽ được mô phỏng dưới dạng sơ đồ như hình sau:



Hình 4-22: Minh họa hiển thị kết quả NG

Tại hình 4-22, khi công nhân thực hiện thao tác lắp ráp O.Slider, hệ thống dự kiến ghi nhận trạng thái $P = 2$. Tuy nhiên, tại thời điểm kiểm tra, giá trị P chỉ đạt 1, không tuân thủ logic chu trình, do đó hệ thống xác định thao tác sai và hiển thị NG.

- Sau khi phát hiện lỗi NG tại bất kỳ bước thao tác nào trong chu trình, hệ thống sẽ thực hiện:

- Dừng tăng chỉ số sản phẩm hoàn thiện (SP), không cộng thêm sản phẩm vào tổng số sản phẩm đạt chuẩn.
- Gửi tín hiệu cảnh báo thông qua thiết bị ngoại vi (đèn báo, còi báo động thông qua Arduino).
- Đồng thời, toàn bộ thông tin liên quan đến chu trình phát sinh lỗi sẽ được hệ thống tự động ghi nhận vào file dữ liệu thống kê có tên *thong_ke.xlsx*. File dữ liệu này chứa các thông tin chi tiết như: thời điểm phát sinh lỗi, loại lỗi phát hiện (thiếu linh kiện, sai vị trí), trạng thái các bước P tại thời điểm lỗi xảy ra, giúp phục vụ công tác thống kê, phân tích và đánh giá hiệu quả sản xuất sau này.

Việc kết hợp đồng bộ giữa cảnh báo thời gian thực và lưu trữ dữ liệu lỗi giúp hệ thống vừa hỗ trợ cảnh báo ngay tại dây chuyền sản xuất, vừa cung cấp cơ sở dữ liệu đầy đủ cho bộ phận quản lý chất lượng (QA) trong quá trình cải tiến liên tục hoạt động sản xuất.

CHƯƠNG 5: PHÂN TÍCH HIỆU QUẢ CỦA HỆ THỐNG

5.1 Chi phí đầu tư triển khai

Phân tích chi phí - lợi ích (Cost-Benefit Analysis) được thực hiện nhằm đánh giá tính khả thi kinh tế của hệ thống giám sát thông minh ứng dụng công nghệ nhận diện hình ảnh tại bộ phận tiền lắp ráp V159 của Công ty TNHH Daiwa Việt Nam. Mục tiêu chính là:

- Xác định tổng chi phí đầu tư ban đầu và chi phí vận hành cho hệ thống giám sát.
- Định lượng lợi ích kinh tế từ việc giảm tỷ lệ lỗi NG, tăng năng suất sản xuất và tối ưu hóa quy trình.
- Cung cấp cơ sở khoa học để thuyết phục ban lãnh đạo công ty về giá trị lâu dài của hệ thống.

Phân tích này không chỉ tập trung vào khía cạnh tài chính mà còn xem xét các lợi ích phi tài chính như cải thiện chất lượng sản phẩm, tăng uy tín thương hiệu và nâng cao hiệu quả quản lý.

5.1.1 Chi phí phần cứng

Hệ thống giám sát yêu cầu các thiết bị phần cứng để thu thập dữ liệu, xử lý và phát tín hiệu cảnh báo. Chi phí ước tính cho 3 bàn gia công O.slider bao gồm:

- Camera công nghiệp: Sử dụng camera công nghiệp để ghi hình thao tác lắp ráp O.Slider. Giá ước tính: 800.000 VNĐ/camera x 3 camera = 2.400.000 VNĐ.
- Vi điều khiển Arduino: Sử dụng Arduino Nano để điều khiển đèn/còi cảnh báo. Giá ước tính: 300.000 VNĐ/bộ x 3 bộ = 900.000 VNĐ.
- Máy tính (Edge Device): Sử dụng Raspberry Pi 4 (4GB RAM) để chạy mô hình và phần mềm giám sát. Giá ước tính: 2.500.000 VNĐ/thiết bị x 1 = 2.500.000 VNĐ.
- Phụ kiện (đèn LED, còi, cáp kết nối): Chi phí linh kiện phụ trợ như đèn báo, còi, dây cáp và nguồn điện. Giá ước tính: 1.000.000 VNĐ.

Tổng chi phí phần cứng: $3.000.000 + 900.000 + 2.500.000 + 1.000.000 = 6.800.000$ VNĐ

5.1.2 Chi phí phần mềm và đào tạo

Google Colab: Được sử dụng để huấn luyện mô hình YOLO. Chi phí thuê Colab (GPU Tesla T4) ước tính: 250.000 VNĐ/tháng x 2 tháng phát triển = 500.000 VNĐ.

Chi phí đào tạo công nghệ: cho khóa tập huấn: 20.000.000 VNĐ/1 khóa

Phần mềm Label Studio: Công cụ dán nhãn dữ liệu miễn phí (phiên bản mã nguồn mở), không phát sinh chi phí.

Tổng chi phí phần mềm và đào tạo: $500.000 + 20.000.000 = 20.500.000$ VNĐ

5.1.3 Chi phí vận hành và bảo trì

Chi phí ước tính cho nhân viên: nhân viên kỹ thuật tại bộ phận đã được đào tạo nên miễn phí.

Phụ cấp cho nhân viên: Chi phí phụ cấp cho 2 nhân viên tại bộ phận 800.000 VNĐ/tháng (tương đương 9.600.000 VNĐ/năm)

Bảo trì phần cứng: Kiểm tra và thay thế camera, Arduino định kỳ (1 lần/năm). Chi phí ước tính: 1.000.000 VNĐ/năm.

Tái huấn luyện mô hình: Cập nhật mô hình với dữ liệu mới (1 lần/năm). Chi phí ước tính: $250.000 \text{ VNĐ} \times 1 = 250.000 \text{ VNĐ/năm}$.

Chi phí vận hành điện: Ước tính: 3.000.000 VNĐ/năm.

Tổng chi phí vận hành và bảo trì = $9.600.000 + 1.000.000 + 250.000 + 3.000.000 = 13.850.000$ VNĐ/năm

5.1.4 Tổng chi phí đầu tư

Tổng chi phí phần cứng = 6.800.000 VNĐ

Tổng chi phí phần mềm và đào tạo = 20.500.000 VNĐ

⇒ Tổng chi phí phát triển = 6.800.000 (phần cứng) + 20.500.000 (phần mềm và đào tạo) = 27.300.000 VNĐ

Tổng chi phí duy trì hàng năm = Tổng chi phí vận hành và bảo trì = 13.850.000 VNĐ/năm

Từ những thông kê trên thu được bảng 5-1

Bảng 5-1: Thống kê chi phí cho dự án phát triển hệ thống giám sát (đơn vị: VNĐ)

Đơn vị (VNĐ)	Tổng chi phí phát triển	Tổng chi phí duy trì	Tổng chi phí
Năm 1	27.300.000	13.850.000	41.150.000
Năm 2	0	13.850.000	13.850.000
Năm 3	0	13.850.000	13.850.000

Từ bảng 5-1 thấy tổng chi phí của dự án cho năm đầu là VNĐ, và của 3 năm là: VNĐ

5.2 Phân tích lợi ích

Việc triển khai hệ thống giám sát thông minh tại công đoạn lắp ráp O.Slider mang lại nhiều lợi ích rõ rệt, không chỉ về mặt chi phí mà còn ở khía cạnh chất lượng sản phẩm, năng suất lao động và cải tiến dài hạn. Các lợi ích cụ thể như sau:

5.2.1 Hiệu quả trong giảm NG

Giả định thực tế dựa trên dữ liệu thu thập cho thấy hệ thống giám sát thông minh có thể giúp giảm đáng kể số lượng sản phẩm lỗi (NG) tại công đoạn lắp ráp O.Slider. Cụ thể, với việc tập trung vào hai nguyên nhân chính gây lỗi – “lắp ráp thiếu linh kiện” và “lắp sai vị trí” – vốn chiếm khoảng 80% tổng số NG, hệ thống được thiết kế để cảnh báo ngay khi phát hiện thao tác sai, nhờ đó công nhân có thể điều chỉnh tức thì.

Dưới đây là bảng thống kê cho giả định của hệ thống giám sát.

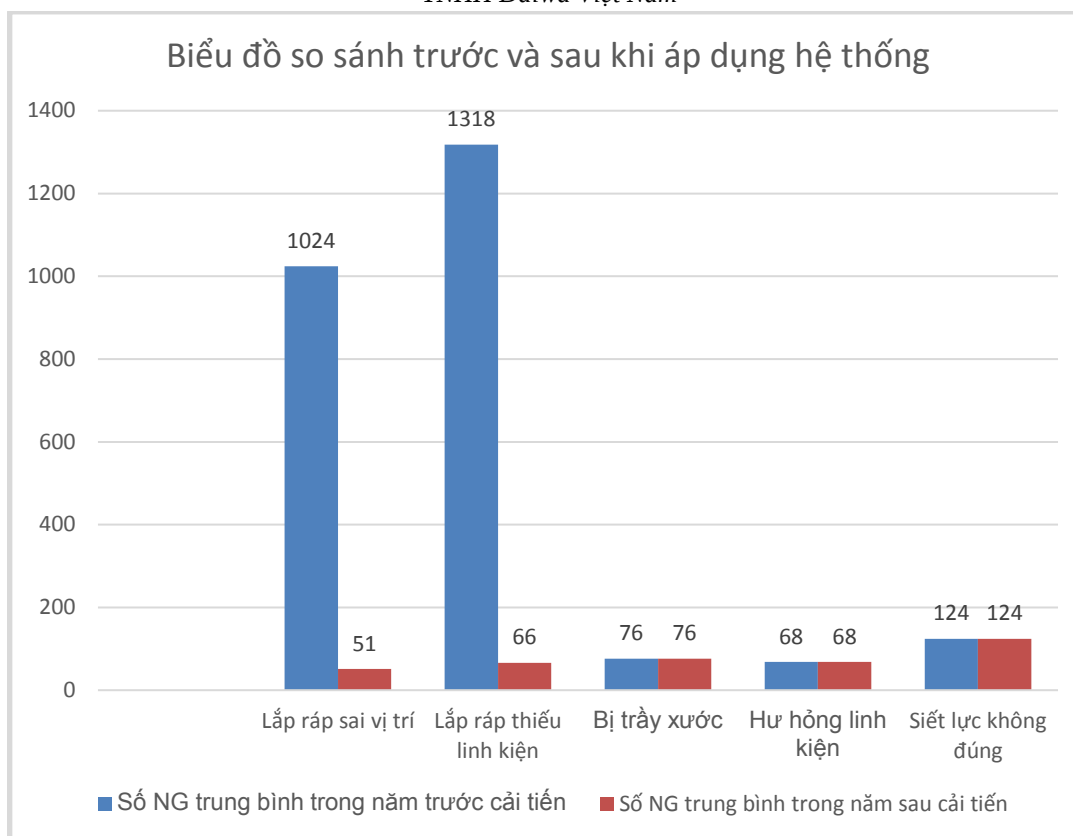
Bảng 5-2: Thống kê trước và sau khi áp dụng hệ thống

Tên lỗi	Số NG trung bình trong năm trước cải tiến	Số NG trung bình trong năm sau cải tiến	Số NG giảm được
Lắp ráp sai vị trí	1024	51	973
Lắp ráp thiếu linh kiện	1318	66	1252
Bị trầy xước	76	76	0
Hư hỏng linh kiện	68	68	0
Siết lực không đúng	124	124	0
Tổng	2610	385	2225

Từ bảng 5-2 thống kê có thể thấy, hai lỗi chiếm tỷ lệ lớn nhất là “lắp ráp sai vị trí” và “lắp ráp thiếu linh kiện” có số lượng NG lần lượt là 1.024 và 1.318 sản phẩm trong một năm, chiếm tổng cộng 80.1% tổng số NG (2.610 sản phẩm). Đây cũng chính là hai nguyên nhân chính được hệ thống giám sát thông minh hướng đến để cải thiện.

Sau khi áp dụng hệ thống, số lượng NG từ hai nguyên nhân này được giả định sẽ giảm 95%, lần lượt còn lại 51 sản phẩm và 66 sản phẩm, giúp tiết kiệm tổng cộng 2225 sản phẩm lỗi mỗi năm. Và tỷ lệ NG sau cải tiến giảm từ 0.54% xuống còn khoảng 0.08% Những lỗi còn lại như trầy xước, hư hỏng linh kiện và siết lực không đúng không nằm trong phạm vi can thiệp của hệ thống nên số lượng không thay đổi.

Hình 5-1 dưới đây thể hiện trực quan sự thay đổi về số lượng lỗi NG trung bình trong năm trước và sau khi triển khai hệ thống giám sát thông minh tại công đoạn lắp ráp O.Slider:



Hình 5-1: Biểu đồ so sánh trước và sau khi áp dụng hệ thống

Từ biểu đồ có thể nhận thấy, sau khi áp dụng hệ thống giám sát, số lượng lỗi NG ở hai nguyên nhân trọng yếu giảm lần lượt từ 1.024 xuống còn 51 sản phẩm và từ 1.318 xuống còn 66 sản phẩm, tương ứng với mức giảm 95%. Trong khi đó, các lỗi khác không bị ảnh hưởng bởi hệ thống vẫn giữ nguyên

5.2.2 Phân tích lợi ích tài chính

Dựa trên bảng 5-2, sau khi áp dụng hệ thống giám sát thông minh vào công đoạn lắp ráp O.Slider, số lỗi NG do hai nguyên nhân chính là “lắp sai vị trí” và “lắp thiếu linh kiện” đã giảm đáng kể:

Lỗi lắp sai vị trí: giảm từ 1.024 xuống còn 51 sản phẩm

Lỗi lắp thiếu linh kiện: giảm từ 1.318 xuống còn 66 sản phẩm

Tổng cộng, hệ thống giúp giảm 2.225 sản phẩm NG/năm, chiếm hơn 85% tổng số lỗi NG trong quy trình.

Chi phí NG đơn vị trung bình: 10.125 VNĐ/bán thành phẩm (bao gồm cả sửa chữa và hư hỏng)

Lợi ích tài chính hằng năm thu được.

Tổng lợi ích tài chính = $2.225 \times 10.125 = 22.528.125$ VNĐ/ năm

5.2.3 Phân tích dòng tiền

Phân tích tài chính là một bước quan trọng trong việc đánh giá tính khả thi và hiệu quả của dự án phát triển khai hệ thống giám sát thông minh tại công đoạn lắp ráp O.Slider. Phần này tập trung vào việc đánh giá chi phí đầu tư ban đầu, chi phí duy trì hàng năm, lợi ích tài chính từ việc giảm sản phẩm lỗi (NG), và dòng tiền ròng trong 5 năm và ROI của dự án. Mục tiêu là xác định thời điểm thu hồi vốn và đánh giá mức độ sinh lời của dự án, từ đó cung cấp cơ sở để quyết định đầu tư. Phân tích được trình bày chi tiết thông tin qua các bảng, bao gồm chi phí phát triển, chi phí duy trì, lợi ích tài chính, dòng tiền tăng tốc kế hoạch, và năm thu hồi vốn. sau khi áp dụng hệ thống.

Dòng tiền ròng mỗi năm được tính bằng cách lấy lợi ích tài chính trừ đi tổng chi phí (chi phí phát triển + chi phí duy trì):

Năm 1:

Lợi ích tài chính: 22.528.125 VNĐ

Chi phí: 27.300.000 (phát triển) + 13.850.000 (duy trì) = 41.150.000 VNĐ

Dòng tiền ròng = 22.528.125 - 41.150.000 = -18.621.875 VNĐ

Năm 2:

Lợi ích tài chính: 22.528.125 VNĐ

Chi phí: 13.850.000 VNĐ

Dòng tiền ròng = 22.528.125 - 13.850.000 = +8.678.125 VNĐ

Năm 3:

Lợi ích tài chính: 22.528.125 VNĐ

Chi phí: 13.850.000 VNĐ

Dòng tiền ròng = 22.528.125 - 13.850.000 = +8.678.125 VNĐ

Năm 4:

Lợi ích tài chính: 22.528.125 VNĐ

Chi phí: 13.850.000 VNĐ

Dòng tiền ròng = 22.528.125 - 13.850.000 = +8.678.125 VNĐ

Năm 5:

Lợi ích tài chính: 22.528.125 VNĐ

Chi phí: 13.850.000 VNĐ

Dòng tiền ròng = 22.528.125 - 13.850.000 = +8.678.125 VNĐ

Dòng tiền tích lũy kế tiếp:

Năm 1: -18.621.875 VNĐ

Năm 2: -18.621.875 + 8.678.125 = -9.943.750 VNĐ

Năm 3: -9.943.750 + 8.678.125 = -1.265.625 VNĐ

Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty
TNHH Daiwa Việt Nam

Năm 4: $-1.265.625 + 8.678.125 = +7.412.500$ VNĐ

Năm 5: $+7.412.500 + 8.678.125 = +16.090.625$ VNĐ

Tổng chi phí 5 năm:

Chi phí phát triển: 27.300.000 VNĐ

Chi phí duy trì: $13.850.000 \times 5 = 69.250.000$ VNĐ

Tổng chi phí = $27.300.000 + 69.250.000 = 96.550.000$ VNĐ

Tổng lợi ích tài chính 5 năm: $22.528.125 \times 5 = 112.640.625$ VNĐ

Bảng 5-3 dưới đây sẽ thống kê lại dòng tiền từ tính toán trên:

Bảng 5-3: Phân tích dòng tiền trong 5 năm sau cải tiến (đơn vị: 1000 VNĐ)

Năm	Chi phí phát triển	Chi phí duy trì	Tổng chi phí	Lợi ích tài chính	Dòng tiền ròng	Dòng tiền lũy kế
Năm 1	27.300	13.850	41.150	22.528	-18.621	-18.621
Năm 2	0	13.850	13.850	22.528	8.678	-9.943
Năm 3	0	13.850	13.850	22.528	8.678	-1.265
Năm 4	0	13.850	13.850	22.528	8.678	7.412
Năm 5	0	13.850	13.850	22.528	8.678	16.090
Tổng	27.300	69.250	96.550	112.640	16.090	

Từ Bảng 5-3 trên nhận thấy thời tiền dương ở năm thứ 4. Nên thời gian hoàn vốn sẽ là trước năm thứ 3.

Thời gian hoàn vốn = $3 + \frac{1.265}{8.678} = 3 + 0.145 = 3.145$ (3 năm, 2 tháng)

Tỷ lệ hoàn vốn ROI đo lường trên tổng chi phí đầu tư, được tính bằng công thức:

$$ROI = \frac{\text{Tổng lợi nhuận ròng}}{\text{Tổng chi phí}} = \frac{16.090}{96.550} = 16.66 \%$$

Kết luận: Dự án này có lợi nhuận đầu tư (ROI) là 16.66% và sau 3 năm 2 tháng sẽ thu hồi vốn.

5.3 Giá trị tổ chức và vận hành

Bên cạnh các lợi ích tài chính có thể đo lường được, việc triển khai hệ thống giám sát thông minh còn mang lại nhiều giá trị thiết thực cho tổ chức, đặc biệt trong việc nâng cao hiệu quả vận hành, ổn định quy trình và cải thiện văn hóa sản xuất tại bộ phận V159. Những giá trị này tuy khó lượng hóa bằng con số cụ thể, nhưng lại đóng vai trò then chốt trong việc xây dựng một môi trường sản xuất hiện đại và bền vững.

5.3.1 Nâng cao nhận thức và tác phong làm việc của công nhân

Khi hệ thống giám sát được đưa vào sử dụng, toàn bộ quá trình thao tác của công nhân đều được theo dõi và lưu trữ. Điều này góp phần làm thay đổi thói quen làm việc theo hướng tích cực hơn. Công nhân có xu hướng chú ý đến độ chính xác trong từng thao tác, hạn chế việc bỏ bước hoặc làm tắt quy trình. Đồng thời, việc biết rằng mọi thao tác đều được ghi nhận cũng khiến công nhân nâng cao tinh thần trách nhiệm cá nhân, từ đó tạo ra một môi trường làm việc nghiêm túc và kỷ luật hơn.

5.3.2 Hỗ trợ bộ phận QA trong công tác giám sát

Thay vì phải liên tục theo dõi bằng mắt thường hoặc kiểm tra ngẫu nhiên sau khi thao tác hoàn tất, nhân viên QA có thể dựa vào dữ liệu hình ảnh và cảnh báo từ phần mềm để nắm bắt tình hình thực tế. Điều này giúp tiết kiệm đáng kể thời gian giám sát, đồng thời nâng cao tính khách quan trong đánh giá thao tác của công nhân. Hệ thống đóng vai trò như một "trợ lý kỹ thuật số", hỗ trợ con người trong việc kiểm tra, đánh giá và phản hồi nhanh chóng khi có vấn đề phát sinh.

5.3.3 Nâng cao hình ảnh doanh nghiệp

Việc áp dụng công nghệ tiên tiến vào dây chuyền sản xuất thể hiện định hướng phát triển hiện đại, không ngừng cải tiến và đề cao chất lượng. Điều này giúp nâng cao uy tín với khách hàng, đặc biệt là các đối tác quốc tế yêu cầu tiêu chuẩn cao về chất lượng và kiểm soát quá trình. Đồng thời, đây cũng là điểm cộng lớn trong công tác tuyển dụng và giữ chân nhân sự kỹ thuật chất lượng cao.

CHƯƠNG 6: KẾT LUẬN

6.1 Kết luận

Sau quá trình nghiên cứu và phát triển, đề tài “Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty TNHH Daiwa Việt Nam” đã đạt được nhiều kết quả, đóng góp cả về mặt kỹ thuật, tài chính lẫn giá trị tổ chức cho Công ty TNHH Daiwa Việt Nam, đặc biệt tại bộ phận tiền lắp ráp V159.

Về mặt kỹ thuật, hệ thống giám sát được xây dựng dựa trên mô hình Yolo11 kết hợp cùng thư viện trong Python đã cho thấy hiệu quả vượt trội trong việc giám sát thao tác lắp ráp theo thời gian thực. Trong quá trình thử nghiệm, hệ thống đã tập trung phát hiện sớm và xử lý hiệu quả hai nhóm lỗi chủ yếu là “lắp ráp sai vị trí” và “thiếu linh kiện” – vốn chiếm gần 80% tổng số lỗi NG. Kết quả thực tế cho thấy, hệ thống giúp giảm được 95% số lỗi thuộc hai nhóm nguyên nhân chính này, từ đó góp phần giảm tỷ lệ NG của công đoạn lắp ráp O.slider từ 0.54% xuống còn 0.08% so với trước khi triển khai hệ thống.

Về hiệu quả tài chính, việc giảm lỗi NG mang lại lợi ích trực tiếp thông qua việc tiết kiệm chi phí sửa chữa và loại bỏ bán thành phẩm hư hỏng. Cụ thể, lợi ích tài chính ước tính đạt 16.703.250 VNĐ, trong khi tổng chi phí đầu tư ban đầu cho hệ thống là 27.000.000 VNĐ. Với mức tiết kiệm này, thời gian hoàn vốn của dự án chỉ khoảng 3 năm 2 tháng, và lợi tức đầu tư (ROI) đạt 16,66% trong vòng 5 năm. Đây là con số khả quan, chứng minh tính hiệu quả kinh tế rõ rệt của việc đầu tư hệ thống giám sát tự động hóa tại công đoạn then chốt trong sản xuất.

Bên cạnh lợi ích tài chính, hệ thống còn tạo ra nhiều giá trị phi tài chính quan trọng. Trước hết là nâng cao ý thức, kỷ luật và sự tập trung trong thao tác của công nhân, khi quá trình làm việc được giám sát một cách khách quan và minh bạch. Thứ hai, bộ phận QA giảm tải trong công tác theo dõi, phát hiện sai lỗi, nhờ đó cải thiện tốc độ phản hồi và nâng cao chất lượng kiểm soát nội bộ. Cuối cùng, việc ứng dụng công nghệ thị giác máy tính và trí tuệ nhân tạo còn góp phần nâng cao hình ảnh doanh nghiệp hiện đại, chuyên nghiệp, từng bước hiện thực hóa mục tiêu chuyển đổi số trong sản xuất công nghiệp theo xu hướng toàn cầu.

Tổng kết lại, đề tài đã hoàn thành tốt các mục tiêu đặt ra cả về mặt công nghệ và thực tiễn ứng dụng. Hệ thống không chỉ mang lại hiệu quả rõ ràng trong việc giảm lỗi, tiết kiệm chi phí mà còn giúp thay đổi tư duy vận hành truyền thống sang hướng chủ động – số hóa – minh bạch

6.2 Hạn chế

Mặc dù đề tài đã bước đầu xây dựng được một hệ thống giám sát thao tác lắp ráp sử dụng công nghệ thị giác máy tính và trí tuệ nhân tạo, tuy nhiên trong quá trình nghiên cứu và triển khai thực tế vẫn còn tồn tại một số hạn chế sau:

Hệ thống mới chỉ được triển khai thử nghiệm tại một quy trình lắp ráp đơn giản – cụ thể là công đoạn O.Slider có số lượng chi tiết và thao tác hạn chế. Trong khi đó, tại doanh nghiệp còn rất nhiều quy trình lắp ráp phức tạp hơn, bao gồm các sản phẩm với hàng chục linh kiện nhỏ và thao tác chồng lắp, yêu cầu kiểm soát nhiều bước liên tục.

Giới hạn về môi trường thử nghiệm: Hệ thống mới chỉ được kiểm thử trong điều kiện mô phỏng tại phòng thí nghiệm và chưa được triển khai rộng rãi trong môi trường sản xuất thực tế tại xưởng, do đó chưa thể đánh giá đầy đủ hiệu quả khi vận hành lâu dài.

Dữ liệu huấn luyện chưa đa dạng: Bộ dữ liệu hình ảnh dùng để huấn luyện mô hình chủ yếu được thu thập trong khoảng thời gian ngắn, chưa bao phủ hết các tình huống thao tác đa dạng có thể xảy ra trong quá trình sản xuất thực tế

Hệ thống phần mềm vẫn còn đơn giản: Giao diện người dùng chưa trực quan, chưa tích hợp chức năng truy xuất lỗi theo thời gian, thống kê hiệu suất dài hạn, hoặc cảnh báo nâng cao cho quản lý cấp cao.

Phần cứng triển khai còn ở mức cơ bản: Việc kết nối Arduino chỉ mới phục vụ chức năng cảnh báo đơn giản.

Chưa có đánh giá chính thức từ doanh nghiệp: Do chưa được thử nghiệm chính thức tại công ty Daiwa nên hệ thống chưa có đánh giá từ bộ phận QA hoặc lãnh đạo sản xuất về mức độ phù hợp với thực tế vận hành.

6.3 Đề xuất

Để hoàn thiện hệ thống và nâng cao tính ứng dụng trong thực tiễn sản xuất, đề tài đề xuất tiếp tục phát triển phần mềm theo hướng trực quan và thân thiện hơn với người dùng. Cụ thể, phần mềm cần bổ sung chức năng cảnh báo tự động, xuất báo cáo định kỳ, hiển thị thống kê lỗi theo thời gian và theo ca làm việc nhằm hỗ trợ công tác quản lý và cải tiến quy trình.

Bên cạnh đó, nghiên cứu đề xuất triển khai thử nghiệm hệ thống tại một dây chuyền thực tế trong nhà máy nhằm đánh giá hiệu quả hoạt động, thu thập phản hồi từ người dùng và tiếp tục điều chỉnh, tối ưu hệ thống. Việc áp dụng thực tế là bước quan trọng để chứng minh tính khả thi và giá trị ứng dụng của hệ thống trong môi trường công nghiệp.

*Phát triển hệ thống giám sát tự động nhằm nâng cao chất lượng quy trình lắp ráp sản phẩm tại công ty
TNHH Daiwa Việt Nam*

Mở rộng phạm vi ứng dụng sang các công đoạn phức tạp hơn: Nghiên cứu triển khai hệ thống cho các quy trình lắp ráp có số lượng linh kiện nhiều hơn, thao tác chồng lắp và yêu cầu cao về thứ tự lắp ráp.

TÀI LIỆU THAM KHẢO

- [1] N. & S. A. Nathani, "Introduction to AI Techniques for Renewable Energy Systems,," in *Foundations of Machine Learning*, 2001.
- [2] V. H. Tiệp, “Tổng quan về Machine Learning, các khái niệm cơ bản,” trong *Machine Learning Cơ Bản*.
- [3] Y. B. Y. & H. G. LeCun, trong *Deep Learning*, 2015, pp. 521, 436-444..
- [4] Joseph Redmon và cộng sự, “Ultralytic,” [Trực tuyến]. Available: <https://docs.ultralytics.com/vi/>.
- [5] N. V. H. Đỗ Văn Tiến, Xử Lý Ảnh Với OpenCV.
- [6] Bùi Việt Hà, “Python cơ bản,” ĐẠI HỌC QUỐC GIA HÀ NỘI.
- [7] V. H. Tiệp, “Đánh giá mô hình,” trong *Machine Learning Cơ Bản*.
- [8] A. B. G. Joseph M. Juran, Juran’s Quality Handbook.
- [9] J. Liker, The Toyota Way, 2004.
- [10] T. N. M. Kiều, Phân tích Báo cáo Tài chính.
- [11] Nikolai Liubimov, Maxim Tkachenko, Michael Malyuk, “Label studio,” Heartex, 2019 . [Trực tuyến]. Available: <https://labelstud.io/>.