

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC NÔNG LÂM TP HCM
KHOA CÔNG NGHỆ THÔNG TIN



LUẬN VĂN TỐT NGHIỆP
NGHIÊN CỨU VISION TRANSFORMER ỨNG
DỤNG NHẬN DIỆN SẢN PHẨM TRONG THANH
TOÁN Ở CỬA HÀNG TIỆN LỢI

Ngành : CNTT

Niên khoá : 2020 – 2024

Lớp : DH20DTB

Sinh viên thực hiện: Nguyễn Hà Phước Hậu

Nguyễn Ngọc Huy

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC NÔNG LÂM TP HCM
KHOA CÔNG NGHỆ THÔNG TIN



LUẬN VĂN TỐT NGHIỆP
NGHIÊN CỨU VISION TRANSFORMER ỨNG
DỤNG NHẬN DIỆN SẢN PHẨM TRONG THANH
TOÁN Ở CỬA HÀNG TIỆN LỢI

Giảng viên hướng dẫn:

TS. Nguyễn Văn Dũ

Sinh viên thực hiện:

Nguyễn Hà Phước Hậu – 20130254

Nguyễn Ngọc Huy – 20130282

TP.HỒ CHÍ MINH, tháng 09 năm 2024

Nghiên cứu Vision Transformer ứng dụng nhận diện sản phẩm trong
thanh toán ở cửa hàng tiện lợi

Năm
2024

CÔNG TRÌNH HOÀN TẤT TẠI TRƯỜNG ĐẠI HỌC NÔNG LÂM TP HCM

Cán bộ hướng dẫn: TS. Nguyễn Văn Dũ

Cán bộ phản biện: TS. Nguyễn Thị Phương Trâm

Nhận xét của Cán bộ hướng dẫn:

[illegible]

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

NHIỆM VỤ LUẬN VĂN CỬ NHÂN

Họ tên sinh viên: **NGUYỄN HÀ PHƯỚC HẬU**

Phái: Nam

Ngày tháng năm sinh: 04/10/2002

Nơi sinh: Bà Rịa

Chuyên ngành: Công Nghệ Thông Tin

Ngành: Công Nghệ Thông Tin

Họ tên sinh viên: **NGUYỄN NGỌC HUY**

Phái: Nam

Ngày tháng năm sinh: 02/02/2002

Nơi sinh: Bình Định

Chuyên ngành: Công Nghệ Thông Tin

Ngành: Công Nghệ Thông Tin

I. TÊN ĐỀ TÀI: Nghiên cứu Vision Transformer ứng dụng nhận diện sản phẩm trong thanh toán ở cửa hàng tiện lợi

II. NHIỆM VỤ VÀ NỘI DUNG

- Nhiệm vụ: Nghiên cứu kỹ thuật Vision Transformer để áp dụng vào bài toán phân loại sản phẩm. Sau đó, áp dụng nhận diện sản phẩm hỗ trợ quá trình thanh toán cho các cửa hàng tiện lợi.

- Nội dung:

- + Nghiên cứu về mô hình Vision Transformer.
- + Xây dựng mô hình máy học phân loại hình ảnh sản phẩm.
- + Xây dựng phần mềm thanh toán cho cửa hàng tiện lợi có hỗ trợ nhận diện sản phẩm bằng mô hình Vision Transformer.

III. NGÀY GIAO NHIỆM VỤ: 20/03/2024

IV. NGÀY HOÀN THÀNH NHIỆM VỤ: 30/08/2024

V. HỌ VÀ TÊN CÁN BỘ HƯỚNG DẪN: TS. Nguyễn Văn Dũ

Ngày / /
CÁN BỘ HƯỚNG DẪN
(Ký và ghi rõ họ tên)

Ngày / /
CÁN BỘ PHẢN BIỆN
(Ký và ghi rõ họ tên)

Ngày / /
KHOA CNTT
(Ký và ghi rõ họ tên)

LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn các thầy cô khoa Công nghệ thông tin trường Đại học Nông Lâm TP.Hồ Chí Minh, với những kiến thức quý báu và sự tận tâm, nhiệt huyết mà thầy cô đã truyền đạt cho chúng em trong suốt những năm đại học.

Đặc biệt, chúng em xin chân thành cảm ơn người Thầy Nguyễn Văn Dũ đã tận tình hướng dẫn, chỉ bảo và giúp đỡ chúng em trong suốt quá trình thực hiện đề tài nghiên cứu này. Cảm ơn Thầy vì Thầy đã truyền tải cho chúng em không chỉ về kiến thức chuyên ngành mà còn cả về cách sống, cách ứng xử, cách suy nghĩ giúp chúng em trưởng thành hơn.

Bên cạnh đó, em xin cảm ơn tất cả những bạn bè đã chia sẻ kiến thức và tận tình giúp đỡ chúng tôi hoàn thành đề tài này. Trong quá trình thực hiện đề tài nghiên cứu, mặc dù chúng em đã có những cố gắng nỗ lực thực hiện nhưng chúng em không thể tránh được những sai sót nhất định. Kính mong sự thông cảm và tận tình chỉ bảo của quý Thầy Cô. Xin chân thành cảm ơn mọi người.

Sinh viên thực hiện

Nguyễn Hà Phước Hậu

Nguyễn Ngọc Huy

DANH SÁCH CHỮ VIẾT TẮT

CNN Convolutional Neural Network

Mạng Neural tích chập

ViT Vision Transformer

Biến đổi Thị giác

DL Deep Learning

Học sâu

ML Machine Learning

Học máy

COCO Common Objects in Context

MLP Multi-Layer Perceptron

DANH MỤC CÁC HÌNH

Hình 1. Kiến trúc ViT [1].....	6
Hình 2. Chia hình ảnh ra thành các patch	7
Hình 3. Tại sao cần phải thêm vị trí cho patch.....	9
Hình 4. So sánh khi Fine-Tuning 2 model trên cùng 1 dataset.....	14
Hình 5. Hình ảnh sơ đồ thực hiện phân tích dữ liệu	26
Hình 6. Sơ đồ quy trình thu thập dữ liệu.....	31
Hình 7. Sơ đồ Phân tích phân phối dữ liệu huấn luyện.....	33
Hình 8. Cấu trúc mô hình VIT	34
Hình 9. Biểu đồ loss và accuracy của mô hình	37
Hình 10. Confusion matrix.....	38
Hình 12. Biểu đồ nhiệt mô hình CNN.....	42
Hình 13. Sơ đồ Hệ thống.....	44
Hình 14. Lược đồ UseCase	45
Hình 15. Lược đồ tuần tự chức năng nhận diện sản phẩm.....	46
Hình 16. Lược đồ tuần tự chức năng thêm sản phẩm	47
Hình 17. Lược đồ tuần tự chức năng sửa sản phẩm.....	48
Hình 18. Lược đồ tuần tự chức năng xóa sản phẩm.....	49
Hình 19. Lược đồ tuần tự chức năng quản lý đơn hàng.....	50
Hình 20. Hình ảnh minh họa chức năng nhận diện sản phẩm(Bước 1)	53
Hình 21. Hình ảnh minh họa chức năng nhận diện sản phẩm(Bước 2)	53
Hình 22. Hình ảnh minh họa chức năng nhận diện sản phẩm(Bước 3)	54
Hình 23. Hình ảnh minh họa chức năng thanh toán(Bước 1)	55
Hình 24. Hình ảnh minh họa chức năng thanh toán(Bước 3)	55
Hình 25. Hình ảnh minh họa chức năng thanh toán(Bước 4)	56
Hình 26. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 1).....	57
Hình 27. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 2).....	57
Hình 28. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 3).....	58
Hình 29. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 4).....	59
Hình 30. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 1)	59
Hình 31. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 2)	60
Hình 32. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 3)	60
Hình 33. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 4)	61
Hình 34. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 1)	61
Hình 35. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 2)	62
Hình 36. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 3)	62
Hình 37. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 4)	63
Hình 38. Hình ảnh minh họa chức năng quản lý đơn hàng(Bước 1)	63
Hình 39. Hình ảnh minh họa chức năng quản lý đơn hàng(Bước 2)	64

Hình 40. Hình ảnh minh họa chức năng xem đơn hàng chi tiết.....	64
---	----

DANH MỤC CÁC BẢNG

Bảng 1. So sánh giữa hai model ViT và CNN	16
Bảng 2. Bảng so sánh hai phương pháp nhận dạng sản phẩm	25
Bảng 3. Tham số huấn luyện mô hình tốt nhất	36
Bảng 4. Bảng kết quả khi dự đoán trên dữ liệu thực tế	39
Bảng 5. Tham số thực nghiệm mô hình CNN	41
Bảng 6. Bảng kết quả khi dự đoán trên dữ liệu thực tế với mô hình CNN	41
Bảng 7. So sánh kết quả thực nghiệm ViT và CNN	42

TÓM TẮT

Trong những năm gần đây, trí tuệ nhân tạo đã có những bước tiến vượt bậc và được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau. Công nghệ AI không chỉ cải thiện hiệu suất làm việc mà còn mang lại nhiều lợi ích thiết thực trong đời sống hàng ngày. Đặc biệt, để nâng cao trải nghiệm mua sắm của khách hàng, việc tự động hóa quá trình nhận diện sản phẩm và thanh toán trở nên cực kỳ quan trọng. Đề tài này tập trung vào nghiên cứu và ứng dụng Vision Transformer cho bài toán phân loại hình ảnh và áp dụng vào hỗ trợ thanh toán tại các cửa hàng tiện lợi bằng cách nhận diện hình ảnh sản phẩm thay vì sử dụng hệ thống mã code trên sản phẩm như truyền thống.

ABSTRACT

In recent years, AI has made remarkable strides and has been widely applied in various fields. AI technology improves work efficiency and brings many practical benefits to daily life. In particular, to enhance the shopping experience for customers, automating the product recognition and payment process has become extremely important. This thesis investigates the Vision Transformer technique and its application to image classification problems supporting the payment process at convenience stores, replacing the traditional barcode system on products.

MỤC LỤC

CHƯƠNG 1. MỞ ĐẦU	1
1.1 Lý do chọn đề tài.....	1
2.1 Mục đích, đối tượng và phạm vi nghiên cứu	2
2.1.1 Mục tiêu nghiên cứu	2
2.1.2 Đối tượng nghiên cứu.....	2
2.1.3 Phạm vi nghiên cứu	3
3.1 Kết quả cần đạt.....	3
CHƯƠNG 2: NỘI DUNG NGHIÊN CỨU	5
2.1. Tổng quan về Vision Transformer	5
2.1.1. Giới thiệu về Vision Transformer	5
2.1.2. Kiến trúc của Vision Transformer.....	5
2.1.3. Sự khác biệt giữa Vision Transformer và các mô hình CNN truyền thống	13
2.1.4. Lợi ích và hạn chế của Vision Transformer	18
2.2. Ứng dụng của Vision Transformer trong nhận diện sản phẩm	19
2.2.1. Giới thiệu về nhận diện sản phẩm	19
2.2.2. Các phương pháp nhận diện sản phẩm truyền thống	20
2.2.3. Ứng dụng Vision Transformer trong nhận diện sản phẩm.....	21
2.2.4. Các nghiên cứu và ứng dụng liên quan	21
CHƯƠNG 3. BÀI TOÁN PHÂN LOẠI SẢN PHẨM	24
3.1. Phát biểu bài toán	24
3.2. Các phương pháp giải quyết bài toán.....	27
3.3. Xây dựng mô hình Vision Transformer	28
3.3.1. Giới thiệu dataset.....	28
3.2.2. Thu thập và tiền xử lý dữ liệu	29
3.2.3. Phân tích phân phối dữ liệu huấn luyện	32
3.2.4. Xây dựng và huấn luyện mô hình	33
3.2.5. Kết quả thực nghiệm và đánh giá.....	36

CHƯƠNG 4. NGHIÊN CỨU VÀ PHÁT TRIỂN	38
4.1. Mô tả yêu cầu và chức năng của hệ thống	44
4.1.1. Sơ đồ hệ thống.....	44
4.1.2. Lược đồ Use Case.....	44
4.1.3. Lược đồ tuần tự (Sequence Diagram)	45
4.1.4. Công nghệ sử dụng.....	51
4.4.1. Mô tả các chức năng.....	52
CHƯƠNG 5. Kiến luận và kiến nghị.....	65
4.1: Kết luận chung	65
5.2. Ưu điểm.....	65
5.3. Nhược điểm	66

CHƯƠNG 1. MỞ ĐẦU

1.1 Lý do chọn đề tài

Hiện nay, cuộc Cách mạng Công nghiệp 4.0 đã và đang tác động sâu rộng vào tất cả các lĩnh vực của cuộc sống từ giáo dục, y tế, ngân hàng, nông nghiệp,... Thực vậy, nhiều tổ chức, doanh nghiệp đã ứng dụng các tiến bộ khoa học công nghệ từ cuộc Cách mạng Công nghiệp 4.0 này trong tổ chức sản xuất và quản lý nhằm nâng cao hiệu quả. Có thể thấy, quá trình đổi mới này đã mang đến nhiều chuyển biến tích cực trong đời sống, đặc biệt là trong thói quen mua sắm của người dân. Trước đây, mọi người thường ra chợ để mua sắm, nhưng trong xã hội hiện đại, khi con người ngày càng bận rộn, việc mua sắm tại chợ không còn hấp dẫn. Người dân cần những nơi có thể đáp ứng nhu cầu mua sắm với hàng hóa phong phú, dễ tìm, thời gian mở cửa dài hơn, và giá cả ổn định hơn, đặc biệt trong các dịp lễ. Chính vì vậy, các siêu thị và cửa hàng tiện lợi đã ra đời để đáp ứng nhu cầu này. Trong số đó, cửa hàng tiện lợi là lựa chọn tối ưu vì thời gian mở cửa 24/24, có thể đặt tại nhiều địa điểm khác nhau, không nhất thiết phải ở trung tâm thành phố như siêu thị. Một số cửa hàng tiện lợi còn có chương trình khuyến mãi hoặc khu vực bán thực phẩm ăn nhanh tại chỗ. Hơn nữa, cửa hàng tiện lợi cung cấp đa dạng mặt hàng và dịch vụ, giúp khách hàng tiết kiệm thời gian, không cần phải chờ đợi lâu như khi mua sắm tại siêu thị. Tuy nhiên, dù nhanh chóng và tiện lợi, cả cửa hàng tiện lợi và siêu thị đều gặp phải một nhược điểm chung là thời gian thanh toán vẫn còn chậm, do nhân viên phải quét mã vạch cho từng sản phẩm. Nếu như siêu thị với diện tích lớn có thể mở nhiều quầy thanh toán để phục vụ nhiều khách hàng cùng lúc, thì đối với các cửa hàng tiện lợi, đây vẫn là một vấn đề cần được giải quyết.

Với sự phát triển mạnh mẽ và tốc độ chóng mặt của Cách mạng Công nghiệp 4.0, nhu cầu của xã hội đối với các hệ thống phần mềm thông minh ngày càng tăng cao. Trong đó, các phần mềm chuyên về phân tích và xử lý hình ảnh đang dần trở thành xu hướng tất yếu và nổi bật trong những năm gần đây. Điều này dễ dàng nhận thấy khi hàng ngày, hàng triệu người trên thế giới sử dụng các ứng dụng điện thoại để chụp

ảnh với các hiệu ứng làm đẹp tích hợp AI, hoặc khi các công ty, tập đoàn sử dụng hệ thống nhận dạng khuôn mặt để chấm công hoặc chẩn đoán sức khỏe qua hình ảnh, như hệ thống VINDR của VinGroup. Nhiều phần mềm AI cũng đã được phát triển để ứng dụng trong kinh tế, tiêu biểu như việc quét mã QR để thanh toán hay mã vạch để phân biệt hàng hóa.

Tuy nhiên, việc quét mã vạch sản phẩm vẫn chưa tận dụng hết sức mạnh của AI. Nếu chúng ta có thể sử dụng AI để nhận dạng khuôn mặt, tại sao không áp dụng công nghệ này để nhận dạng nhiều sản phẩm cùng lúc khi thanh toán, nhằm tiết kiệm thời gian thay vì quét từng mã vạch một cách thủ công? Nếu vấn đề này được giải quyết, nó sẽ mang lại hiệu quả kinh tế lớn như tiết kiệm thời gian trong quá trình thanh toán, giảm công sức cho nhân viên, rút ngắn thời gian chờ đợi của khách hàng, từ đó cải thiện trải nghiệm mua sắm và tăng tỷ lệ khách hàng quay lại. Với mong muốn góp phần giải quyết vấn đề này, đề tài ***“NGHIÊN CỨU VISION TRANSFORMER ỨNG DỤNG NHẬN DIỆN SẢN PHẨM TRONG THANH TOÁN Ở CỬA HÀNG TIỆN LỢI”*** hướng đến việc phát triển một phần mềm có khả năng nhận dạng nhiều sản phẩm cùng lúc, giúp quá trình thanh toán trở nên đơn giản hơn và tiện lợi hơn.

2.1 Mục đích, đối tượng và phạm vi nghiên cứu

2.1.1 Mục tiêu nghiên cứu

- Nghiên cứu về mô hình Vision Transformer.
- Xây dựng mô hình máy học phân loại hình ảnh sản phẩm.
- Xây dựng hệ thống thanh toán bằng cách nhận diện sản phẩm thông qua hình ảnh.

2.1.2 Đối tượng nghiên cứu

Đối tượng nghiên cứu là kỹ thuật Vision Transformer và bài toán phân loại hình ảnh có áp dụng kỹ thuật này để hỗ trợ quá trình thanh toán tại các cửa hàng tiện lợi. Một số công việc thực hiện như sau:

- Chuẩn bị và tiền xử lý dữ liệu để xây dựng mô hình.

- Phân tích cấu trúc và hoạt động: Nghiên cứu chi tiết cách mà Vision Transformer xử lý thông tin hình ảnh, tập trung vào cơ chế self-attention và việc ánh xạ các phần nhỏ của hình ảnh.

- Hiệu suất của Vision Transformer: Đánh giá khả năng của mô hình trong việc nhận diện và phân loại hình ảnh trên các bộ dữ liệu thử nghiệm khác nhau.

- Tối ưu hóa và ổn định hóa: Tìm hiểu cách tối ưu hóa mô hình, điều chỉnh tham số để cải thiện độ chính xác và ổn định hóa quá trình huấn luyện.

- Xây dựng phần mềm thanh toán bằng nhận diện sản phẩm demo.

2.1.3 Phạm vi nghiên cứu

Nội dung nghiên cứu và sản phẩm của luận văn trong phạm vi như sau:

- Ứng dụng này sẽ được áp dụng tại các cửa hàng tiện ích hoặc là siêu thị nhỏ.

- Sử dụng camera điện thoại kết nối với máy tính. camera được đặt sao cho có thể chụp từ trên xuống mặt phẳng sản phẩm một cách hiệu quả.

- Nhận diện được các loại như chai các chai, lon... không hỗ trợ các thực phẩm tươi như rau, củ, quả, thịt...

- Các sản phẩm phải nằm trên bề mặt phẳng, nhãn hiệu phải hướng lên trên và không bị chồng và che bởi các vật khác, hình ảnh phải rõ nét không bị mờ.

- Không xem xét đến kích thước của sản phẩm.

3.1 Kết quả cần đạt

- Phần mềm thanh toán bằng nhận diện sản phẩm từ thành quả nghiên cứu trên bao gồm các chức năng:

- + Nhận diện sản phẩm thông qua hình ảnh khi thanh toán.

Tính ra số tiền từ sản phẩm được nhận dạng và hiển thị hóa đơn cho khách hàng.

- Tài liệu báo cáo về Vision Transformer để xử lý dữ liệu đầu vào của hình ảnh, giải quyết các vấn đề về nhận diện.

CHƯƠNG 2: NỘI DUNG NGHIÊN CỨU

2.1. Tổng quan về Vision Transformer

2.1.1. Giới thiệu về Vision Transformer

Vision Transformer là một kiến trúc mạng nơ-ron đột phá, tái hiện cách chúng ta xử lý và hiểu hình ảnh. Mô hình ViT được giới thiệu vào năm 2021 trong bài báo nghiên cứu tại hội nghị ICLR 2021 với tiêu đề “An Image is Worth 16*16 Words: Transformers for Image Recognition at Scale” [1]. Lấy cảm hứng từ thành công của các mô hình Transformer trong xử lý ngôn ngữ tự nhiên, ViT giới thiệu một cách mới để phân tích hình ảnh bằng cách chia chúng thành các mảng nhỏ và tận dụng các cơ chế tự chú ý (self-attention). Điều này cho phép mô hình nắm bắt cả các mối quan hệ cục bộ và toàn cục trong hình ảnh, dẫn đến hiệu suất ấn tượng trong các nhiệm vụ thị giác máy tính khác nhau.

Vision Transformer đã trở thành một đối thủ cạnh tranh với CNN, đang là công nghệ hàng đầu trong lĩnh vực thị giác máy tính và được sử dụng rộng rãi cho các nhiệm vụ nhận dạng hình ảnh khác nhau.

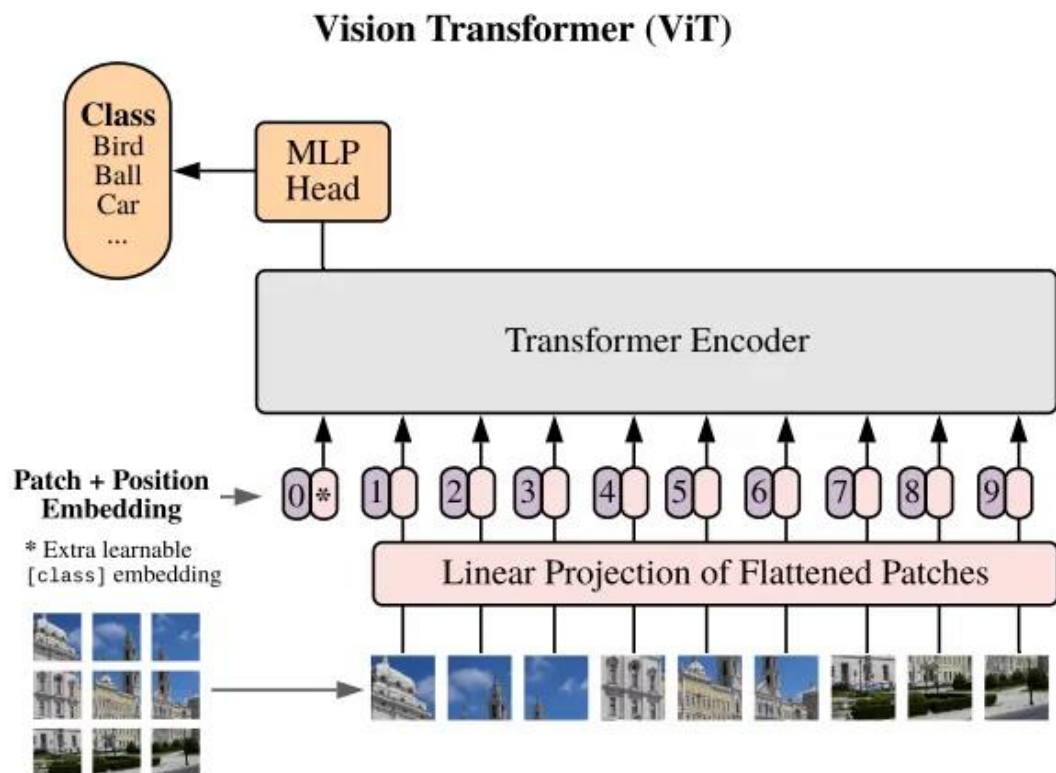
Mặc dù mạng neural tích chập đã thống trị lĩnh vực thị giác máy tính trong nhiều năm, nhưng các mô hình ViT mới đã cho thấy khả năng đáng kinh ngạc, đạt được hiệu suất tương đương hoặc thậm chí vượt trội hơn so với CNN trên nhiều bài toán. Khi được huấn luyện trước trên một lượng lớn dữ liệu và chuyển sang các bộ chuẩn nhận diện hình ảnh có kích thước vừa hoặc nhỏ (như ImageNet, CIFAR-100, VTAB, v.v.), ViT đạt được kết quả xuất sắc so với các mạng tích chập tiên tiến trong khi yêu cầu ít tài nguyên tính toán hơn đáng kể để huấn luyện.

2.1.2. Kiến trúc của Vision Transformer

Vision Transformer có ứng dụng rộng rãi trong các bài toán nhận dạng hình ảnh phổ biến như object detection, image segmentation, image classification và action recognition. Ngoài ra, ViT còn được áp dụng trong các mô hình generative và các bài

toán đa mô hình như visual grounding, visual-question answering, and visual reasoning,...

Mô hình ViT chia nhỏ hình ảnh thành các patches, sau đó chuyển các phân vùng này thành các chuỗi và đưa chúng vào một mô hình transformer. Các phân vùng này được biến đổi thành các vector thông qua một lớp tuyến tính trước khi được đưa vào mô hình Transformer. Mô hình Transformer sau đó sẽ học cách ánh xạ các vector đầu vào này thành các vector đầu ra tương ứng với các đặc trưng quan trọng trong hình ảnh.



Hình 1. Kiến trúc ViT [1]

Dựa vào kiến trúc trên ta có thể tóm gọn cách hoạt động của ViT như sau: Embedding, Transformer Encoder, MLP Head.

-Embedding:

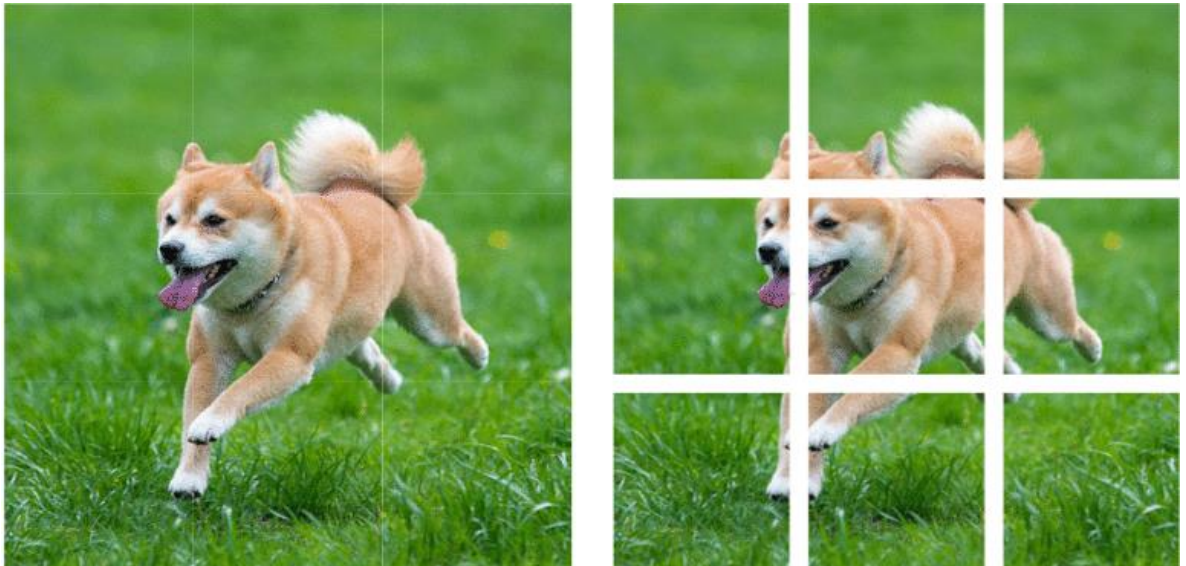
Đóng vai trò quan trọng trong kiến trúc ViT và là bước đầu tiên trong quá trình xử lý hình ảnh. Đây là giai đoạn chuyển đổi hình ảnh từ dạng pixel thành các vector

nhúng có thể xử lý được bởi mô hình transformer. Các bước chi tiết của quá trình embedding trong ViT bao gồm:

Chia nhỏ hình ảnh: Việc chia nhỏ hình ảnh thành các mảnh nhỏ là một bước quan trọng trong ViT, giúp mô hình xử lý hình ảnh theo cách hiệu quả và linh hoạt hơn. Đầu tiên, hình ảnh đầu vào được chia thành một lưới các mảnh nhỏ, mỗi mảnh có kích thước bằng nhau. Điều này tương tự như việc chia nhỏ một bức tranh thành các mảnh ghép nhỏ để dễ dàng quản lý và xử lý. Giả sử chúng ta có một hình ảnh kích thước $H \times W$ và chúng ta muốn chia hình ảnh này thành các mảnh nhỏ kích thước $P \times P$.

$$\text{Patches} = \frac{H \times W}{P^2}$$

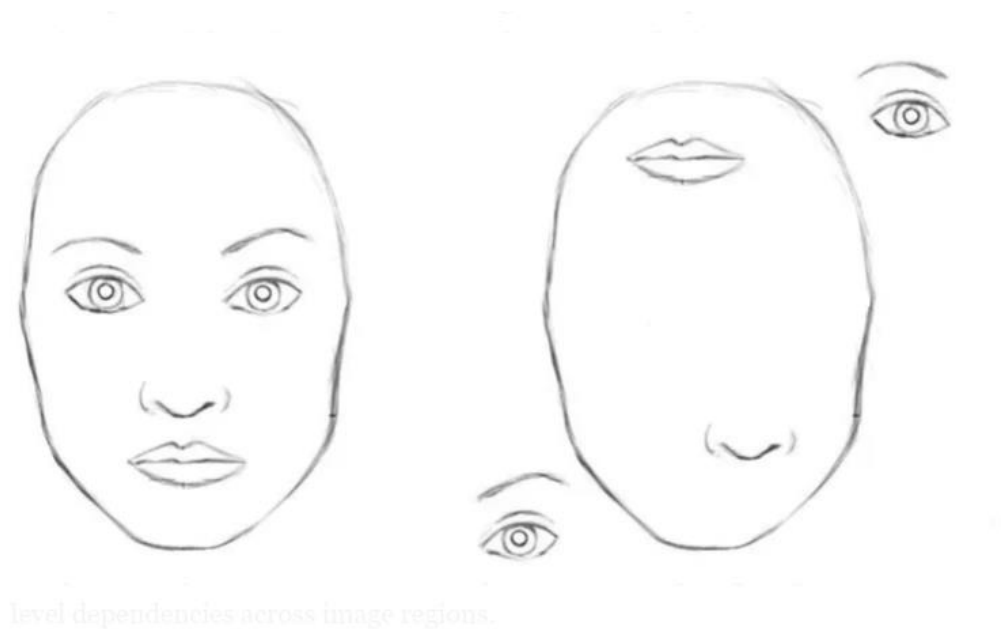
Ví dụ, một hình ảnh 224x224 có thể được chia thành 14 mảnh nhỏ, mỗi mảnh kích thước 16x16 kết quả thu được (14,16,16,3). Quá trình này giúp đơn giản hóa việc xử lý hình ảnh bằng cách tạo ra các đơn vị nhỏ hơn và dễ xử lý hơn.



Hình 2. Chia hình ảnh ra thành các patch

Nhúng các mảnh hình ảnh: Mỗi mảnh hình ảnh sau đó được biến đổi thành một vector có chiều cố định. Quá trình này tương tự như việc chuyển đổi các từ trong xử lý ngôn ngữ tự nhiên thành các vector nhúng. Các mảnh hình ảnh được trải phẳng và sau đó được chiếu vào không gian vector thông qua một lớp Dense. Điều này giúp chuyển đổi thông tin từ dạng pixel thành dạng vector tương ứng với mỗi patch là một từ trong câu và có thể xử lý bởi mô hình transformer.

Thêm position embeddings: Trong ViT nó đóng vai trò quan trọng vì chúng cung cấp thông tin về vị trí của các patch trong hình ảnh, điều mà mô hình transformer vốn không tự động nắm bắt. Mô hình transformer, vốn được phát triển cho các bài toán xử lý ngôn ngữ tự nhiên, không có khả năng nắm bắt thông tin về thứ tự hoặc vị trí của các token đầu vào vì nó hoạt động dựa trên cơ chế attention. Trong ngữ cảnh văn bản, thứ tự của các từ là quan trọng, nhưng các mô hình transformer không tự động nhận diện điều này mà không có thêm thông tin bên ngoài. Khi áp dụng transformers vào hình ảnh, hình ảnh đầu vào thường được chia thành các patch nhỏ (ví dụ, 16x16 pixel). Mỗi patch được chuyển đổi thành một vector nhúng, nhưng các mô hình transformer không tự hiểu được vị trí của từng patch trong toàn bộ hình ảnh. Thông tin về vị trí của các patch là quan trọng vì mối quan hệ không chỉ giữa các patch mà còn giữa các vị trí cụ thể trong hình ảnh cũng ảnh hưởng đến việc hiểu và phân tích hình ảnh.



Hình 3. Tại sao cần phải thêm vị trí cho patch

Để khắc phục điều này, ViT thêm các position embeddings vào các embeddings của các patch. Position embeddings là các vector đặc biệt được gán cho từng vị trí của patch trong hình ảnh và được cộng vào các embeddings của patch. Điều này cho phép mô hình học được thông tin về vị trí tương đối và tuyệt đối của các patch trong hình ảnh, từ đó nắm bắt được mối quan hệ không gian giữa các phần của hình ảnh. Việc này rất quan trọng để đảm bảo rằng mô hình hiểu được ngữ cảnh không gian của các mảnh hình ảnh, cho phép nó nắm bắt được cả thông tin địa phương và toàn cục của hình ảnh.

- Transformer Encoder

Phần cốt lõi của Vision Transformer bao gồm nhiều lớp transformer encoder, mỗi lớp chứa hai thành phần chính: multi-head attention và mạng nơ-ron feedforward.

Cơ chế self-attention nắm bắt các mối quan hệ giữa các mảnh khác nhau trong chuỗi đầu vào. Đối với mỗi mảnh nhúng (patch embedding), self-attention tính toán tổng có trọng số của tất cả các mảnh nhúng, trong đó trọng số được xác định bởi mức độ liên quan của mỗi mảnh đối với mảnh hiện tại. Cơ chế này cho phép mô hình tập

trung vào các mảnh quan trọng trong khi xem xét cả ngữ cảnh địa phương và toàn cục. Multi-head attention sử dụng nhiều bộ tham số học được (attention heads) để nắm bắt các loại mối quan hệ khác nhau.

Công thức tính:

$$Head_i = Attention_i(Q, K, V) = softmax\left(\frac{Q_i \times K_i^T}{\sqrt{d_k}}\right) V$$

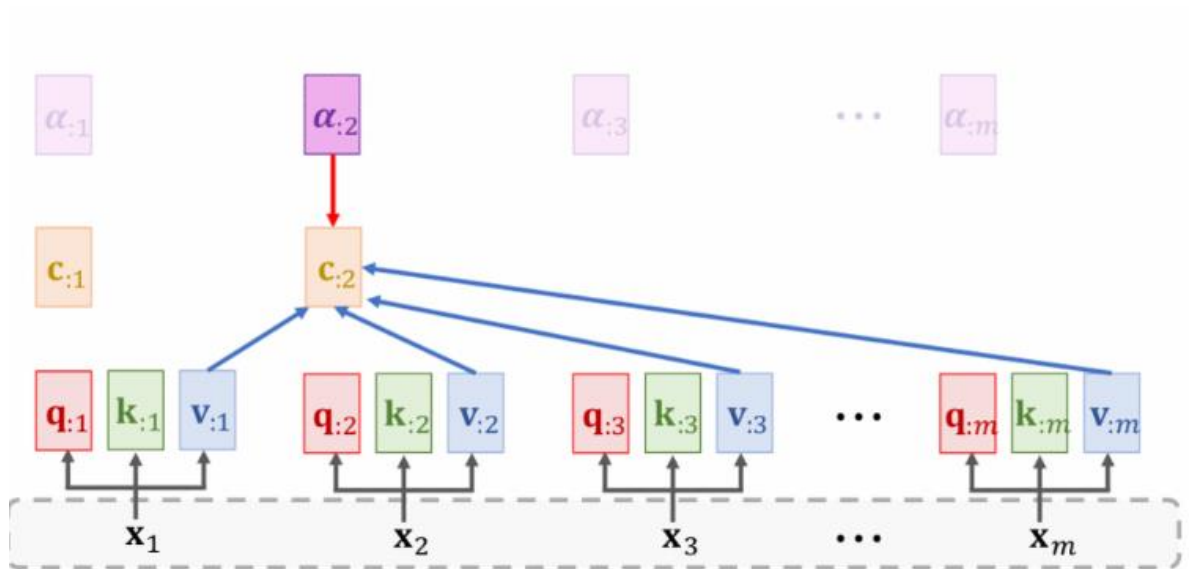
Trong đó:

Q: vector truy vấn.

K: vector khóa.

V: vector giá trị.

d_k : kích thước của vector khóa.



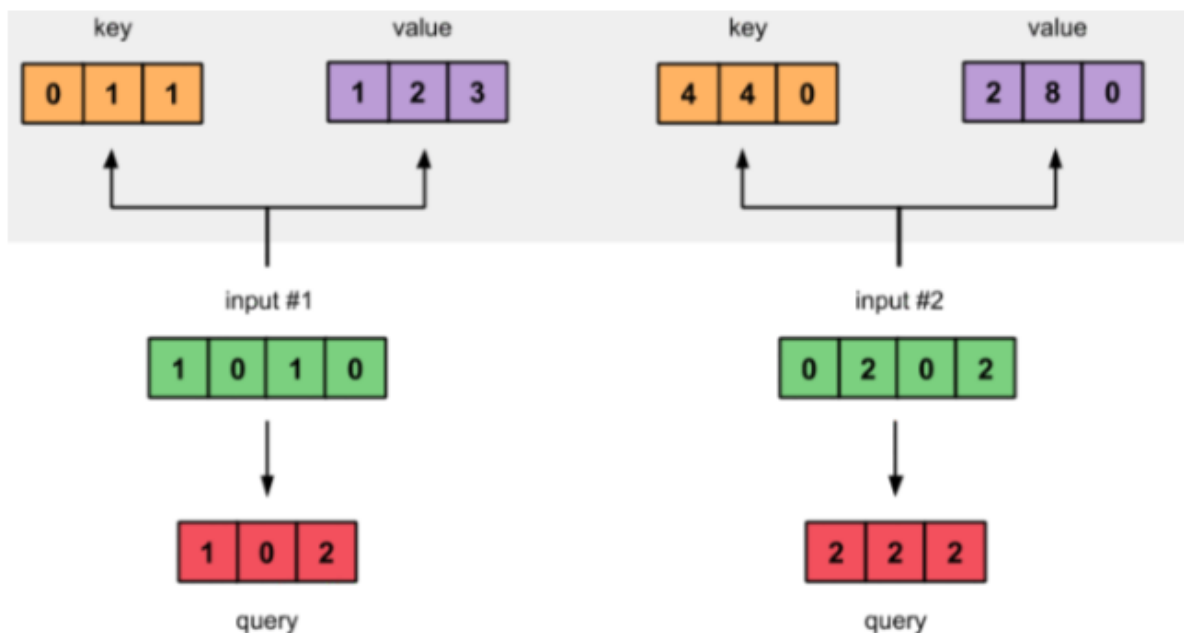
Hình 4. Hình ảnh minh họa cho công thức Self-attention

Trong mỗi self-attention nhận vào các input là các patch có kích thước mỗi patch ví dụ là (4x4). Trong mỗi patch sẽ khởi tạo một khóa K, giá trị V và truy vấn Q được khởi tạo ngẫu nhiên sử dụng một trong số các phân phối Gaussian, Xavier, Kaiming... Truy vấn Q sẽ tìm kiếm trong các giá trị khóa K của tất cả các patch khác

để tìm ra ứng viên cung cấp ngữ cảnh cho nó (thông qua tích vô hướng). Cuối cùng, ta có ma trận đầu ra của lớp attention:

Làm rõ một self-attention trong đó dễ hiểu hơn về cơ chế hoạt động của nó:

Ta ví dụ có input hình ảnh sao khi chia thành các 4 patch và làm phẳng kèm đánh vị trí khi làm phẳng mỗi patch là vector 4 chiều và số chiều d_k ta lấy ví dụ là 3 khi vào self-attention sẽ được khởi tạo kèm theo ma trận trọng số Key, Query, Value với kích thước (4×3) lần lượt tính ra được Key, Query, Value đạt được bằng cách nhân ma trận input với trọng số tương ứng của chúng kết quả là vector có chiều bằng với chiều output.



Hình 2.1: Mô tả quan hệ giữa các patch và Key, Query, Value

Khi có giá trị của Key, Query, Value chúng ta thực hiện tính ra điểm của mỗi input bằng cách lấy Query của input đó nhân với Key của tất cả input còn lại kể cả chính nó:

$$scores = \frac{Q \times K^T}{\sqrt{d_k}}$$

Ta có được score Input 1 với các Input còn lại:

$$(1 \ 0 \ 2) \times \begin{pmatrix} 0 & 4 \\ 1 & 4 \\ 1 & 0 \end{pmatrix} \times \frac{1}{\sqrt{3}} = (1.1, 2.3)$$

Ta có được score Input 2 với các Input còn lại:

$$(2 \ 2 \ 2) \times \begin{pmatrix} 0 & 4 \\ 1 & 4 \\ 1 & 0 \end{pmatrix} \times \frac{1}{\sqrt{3}} = (2.3, 9.2)$$

Bước tiếp theo ta đưa kết quả này qua hàm softmax được kết quả với input 1: [0.2, 0.7] ta làm tương tự với input 2.

Sau khi có softmax của input 1 ta lấy nhân với các value ta được các Attention Output có chiều (số lượng input, số chiều value) và nó cũng là output của một Head trong Multi Head Attention.

Sau khi có được các Output của từng Head, ta sẽ tổng hợp chúng lại thành 1 Output duy nhất.

$$MultiHead(Q, K, V) = Concat(Head_1, Head_2, \dots, Head_h)W^0$$

W^0 là ma trận có chiều rộng bằng với chiều rộng của ma trận Input, mục đích sử dụng của nó là để đưa kích thước của Output về bằng với kích thước của Input.

Một patch có thể có nhiều kết nối với các patch khác theo những cách rất đa dạng, do đó mỗi patch có thể có nhiều bộ Q-K-V liên kết với nó. Kiến trúc này có tên gọi là multi-headed attention. Mỗi ma trận self-attention được tính toán riêng rẽ trên toàn bộ để học ra những tầng ngữ nghĩa khác nhau trước khi được kết hợp với nhau bởi một ma trận trọng số.

Sau multi-head attention, đầu ra từ cơ chế chuyển qua một mạng nơ-ron feedforward. Mạng này thường bao gồm một lớp fully connected tiếp theo là một hàm kích hoạt như ReLU (Rectified Linear Unit) hoặc GELU (Gaussian Error Linear Unit). Mục đích của mạng feedforward là giới thiệu tính phi tuyến tính và cho phép mô hình học các mối quan hệ phức tạp giữa các mảnh.

Cả đầu ra của cơ chế self-attention và mạng feedforward đều được theo sau bởi layer normalization và residual connections. Layer normalization giúp ổn định và tăng tốc độ huấn luyện bằng cách chuẩn hóa các đầu vào cho mỗi lớp con. Residual connections, còn được gọi là skip connections, thêm các nhúng đầu vào ban đầu vào đầu ra của mỗi lớp con. Điều này giúp dòng gradient trong quá trình huấn luyện và ngăn chặn vấn đề vanishing gradient.

- MLP Head

MLP Head là một cấu trúc mạng neural đa lớp, thường được sử dụng để chuyển đổi đầu ra của mô hình transformer thành dự đoán cuối cùng. MLP Head bao gồm một hoặc nhiều lớp fully connected với các chức năng kích hoạt nhằm học các đặc trưng phân loại từ embeddings.

● Các Thành Phần Chính của MLP Head:

- Fully Connected Layers: MLP Head thường bao gồm một hoặc nhiều lớp fully connected. Mỗi lớp fully connected được nối với tất cả các neuron của lớp trước đó và học các trọng số để thực hiện phân loại.
- Activation Functions: Sau mỗi lớp fully connected, thường có một lớp kích hoạt ví dụ như relu hoặc gelu để thêm tính phi tuyến vào mô hình, giúp cải thiện khả năng học.
- Output Layer: Lớp đầu ra của MLP Head là một lớp fully connected với kích thước bằng số lượng lớp phân loại. Lớp này sẽ cung cấp các logits hoặc giá trị không chuẩn hóa cho các lớp phân loại.

2.1.3. Sự khác biệt giữa Vision Transformer và các mô hình CNN truyền thống

Khi so sánh ViT với CNN, chúng ta có thể thấy rõ sự khác biệt trong cách mà mỗi mô hình xử lý hình ảnh và học từ dữ liệu.

CNN đã được sử dụng rộng rãi trong nhiều bài toán phân loại hình ảnh trong nhiều năm. CNN hoạt động bằng cách sử dụng các lớp tích chập để trích xuất đặc

trung từ hình ảnh. Các lớp này áp dụng các bộ lọc nhỏ qua toàn bộ hình ảnh để học các đặc trưng địa phương như cạnh và kết cấu. Bằng cách kết hợp các lớp pooling, CNN có thể giảm kích thước không gian của hình ảnh, giữ lại các đặc trưng quan trọng và xây dựng các hierarchies của đặc trưng từ thấp đến cao. Cuối cùng, các đặc trưng này được chuyển đến các lớp fully connected để thực hiện phân loại. CNN thường yêu cầu ít tài nguyên tính toán hơn và có thể đạt được hiệu suất cao khi tinh chỉnh cho các bài toán cụ thể. Tuy nhiên, CNN có thể gặp khó khăn khi cần nắm bắt các mối quan hệ toàn cục trong hình ảnh, đặc biệt là khi xử lý hình ảnh có kích thước lớn hoặc yêu cầu hiểu biết toàn cục sâu hơn.

Ngược lại, ViT là một cách tiếp cận mới hơn, áp dụng transformer để xử lý hình ảnh. Thay vì áp dụng bộ lọc cục bộ, ViT chia hình ảnh thành các patch nhỏ và ánh xạ chúng thành các vector nhúng. Các vector này sau đó được xử lý qua các lớp transformer, trong đó cơ chế attention giúp học các mối quan hệ toàn cục giữa các patch. Điều này cho phép ViT nắm bắt các đặc trưng toàn cục của hình ảnh tốt hơn.

Mặc dù ViT có khả năng mở rộng tốt hơn và có thể làm việc hiệu quả với dữ liệu lớn, nó yêu cầu nhiều tài nguyên tính toán hơn, đặc biệt đối với các mô hình lớn và phức tạp. ViT có thể vượt trội trong các bài toán phân loại hình ảnh khi có lượng dữ liệu lớn và tài nguyên tính toán đủ.

Version	Tuning	CNN Val Accuracy	ViT Val Accuracy
#11	Base + 3 Layers Head with Normalization and low Data Augmentation	0.7305	0.8828
#23	Added Dropout Layers	0.7344	0.8828
#33	Increased Data Augmentation, Increased Batch Size and Increased Epochs	0.7090	0.9219
#44	Increase Learning Rate	0.7158	0.9326
#45	Learning Rate back to default, Increased Epochs (100)	0.7510	0.9463

Hình 4. So sánh khi Fine-Tuning 2 model trên cùng 1 dataset

Khi Fine-Tuning 2 model trên cùng 1 dataset tác giả đã cho ra được dữ liệu trên, chúng ta có thể thấy rõ sự vượt trội của ViT so với CNN trong nhiều điều kiện khác nhau.

Khi bắt đầu với phiên bản cơ bản, ViT đã thể hiện ưu thế rõ rệt với độ chính xác đạt 0.8828, cao hơn đáng kể so với CNN với độ chính xác 0.7305. Điều này cho

thấy ViT có khả năng học và phân tích hình ảnh tốt hơn ngay từ đầu, ngay cả khi không áp dụng nhiều kỹ thuật tinh chỉnh.

Khi chúng tôi thêm các lớp dropout vào CNN ở phiên bản #23, mặc dù độ chính xác của CNN đã tăng lên một chút, đạt 0.7344, ViT vẫn duy trì độ chính xác ổn định ở mức 0.8828. Điều này chứng tỏ rằng, mặc dù cải tiến nhỏ có thể nâng cao hiệu suất của CNN, nhưng ViT vẫn giữ được lợi thế vượt trội về tổng thể.

Tiến đến phiên bản #33, khi áp dụng các kỹ thuật tối ưu hóa như tăng cường dữ liệu, tăng kích thước batch và số lượng epochs, độ chính xác của CNN lại giảm xuống còn 0.7090, trong khi ViT đạt độ chính xác 0.9219. Điều này cho thấy các thay đổi lớn hơn trong quá trình huấn luyện có thể làm giảm hiệu suất của CNN, trong khi ViT vẫn duy trì hiệu suất cao.

Ở phiên bản #44, với việc tăng learning rate, CNN đạt được một mức độ chính xác cải thiện lên 0.7158. Tuy nhiên, ViT tiếp tục cho thấy sự vượt trội với độ chính xác 0.9326, cho thấy rằng việc điều chỉnh learning rate có tác động tích cực đến CNN nhưng vẫn không thể sánh với sự hiệu quả của ViT.

Cuối cùng, khi chúng tôi điều chỉnh learning rate trở lại giá trị mặc định và tăng số lượng epochs lên 100 ở phiên bản #45, độ chính xác của CNN tăng lên 0.7510. Mặc dù có sự cải thiện này, ViT vẫn đạt được hiệu suất ấn tượng nhất với độ chính xác lên tới 0.9463. Điều này chứng tỏ rằng, ngay cả khi tối ưu hóa các tham số và tăng cường huấn luyện, ViT vẫn giữ được ưu thế đáng kể về hiệu suất.

Nhìn chung, bảng so sánh cho thấy rõ rằng ViT không chỉ thể hiện hiệu suất vượt trội hơn so với CNN mà còn duy trì sự ổn định và hiệu quả cao trong nhiều điều kiện khác nhau. Điều này nhấn mạnh rằng ViT có thể là lựa chọn ưu việt hơn cho các bài toán phân loại hình ảnh, đặc biệt khi các điều kiện và kỹ thuật tinh chỉnh được áp dụng.

Dựa vào thông tin trên ta có được so sánh trực quan về ViT và CNN:

Bảng 1. So sánh giữa hai model ViT và CNN

Tiêu chí	CNN	ViT
Khả năng học đặc trưng	Tốt trong việc học các đặc trưng cục bộ nhờ các lớp tích chập. Khả năng học các hierarchies đặc trưng từ thấp đến cao.	Tốt trong việc học các mối quan hệ toàn cục nhờ cơ chế attention, giúp nắm bắt các đặc trưng toàn diện hơn.
Yêu cầu dữ liệu	Hiệu quả với tập dữ liệu vừa và nhỏ. Cần ít dữ liệu để đạt được hiệu suất tốt, nhờ vào khả năng trích xuất đặc trưng mạnh mẽ từ các lớp tích chập.	Cần nhiều dữ liệu hơn để đạt được hiệu suất tối ưu. Hiệu quả hơn với các tập dữ liệu lớn nhờ khả năng học toàn cục.
Tài nguyên tính toán	Thường yêu cầu ít tài nguyên tính toán hơn so với ViT, đặc biệt đối với các mô hình nhỏ và trung bình.	Yêu cầu nhiều tài nguyên tính toán hơn do số lượng lớn các tham số và tính toán attention.
Khả năng mở rộng	Có thể gặp khó khăn khi mở rộng với hình ảnh có kích thước lớn hoặc yêu cầu hiểu biết toàn cục phức tạp.	Tốt hơn trong việc mở rộng với các tập dữ liệu lớn và hình ảnh có kích thước lớn nhờ khả năng học toàn cục.
Khả năng tổng quát	Hiệu suất tốt với các bài toán phân loại hình ảnh truyền thống và có thể đạt kết quả cao với các mô hình đã tinh chỉnh tốt.	Cung cấp khả năng tổng quát tốt hơn khi có lượng dữ liệu lớn, giúp mô hình học các đặc trưng phức tạp hơn.

Đặc điểm học tập	Học đặc trưng cục bộ nhanh chóng và hiệu quả với số lượng lớp và bộ lọc hạn chế.	Cần nhiều epoch và tài nguyên tính toán hơn để học các mối quan hệ toàn cục phức tạp.
-------------------------	--	---

Dưới đây là một số điểm cần lưu ý về ViT:

- ViT đã chứng minh hiệu quả của mình đối với các nhiệm vụ thị giác máy tính; các mô hình vision transformer đã nhận được sự chú ý đáng kể và làm suy yếu sự thống trị của CNN trong lĩnh vực thị giác máy tính.
- Vì Transformers yêu cầu một lượng lớn dữ liệu để đạt độ chính xác cao, quá trình thu thập dữ liệu có thể kéo dài thời gian dự án. Trong trường hợp có ít dữ liệu, CNN thường cho kết quả tốt hơn so với Transformers.
- Thời gian huấn luyện của Transformer nhanh hơn so với CNN. So sánh theo hiệu suất tính toán và độ chính xác, Transformers có thể được lựa chọn nếu thời gian huấn luyện mô hình bị giới hạn.
- Cơ chế tự chú ý (self-attention) có thể mang lại nhận thức sâu hơn cho mô hình. Rất khó hiểu được các điểm yếu của mô hình CNN, nhưng attention maps có thể được trực quan hóa và giúp tìm ra cách cải tiến mô hình. Quá trình này khó hơn đối với các mô hình CNN.
- Mặc dù có một số framework cho Transformers, triển khai các mô hình CNN vẫn ít phức tạp hơn.
- Sự xuất hiện của Vision Transformers cũng cung cấp một nền tảng quan trọng cho việc phát triển các mô hình computer vision. Mô hình vision lớn nhất là ViT-MoE của Google, có 15 tỷ tham số, đã lập kỷ lục mới trong việc phân loại ImageNet-1K.

ViT đã trở thành một trong những tiến bộ quan trọng trong lĩnh vực xử lý hình ảnh và tiếp tục được nghiên cứu và phát triển để cải thiện hiệu suất và khả năng ứng dụng trong tương lai.

2.1.4. Lợi ích và hạn chế của Vision Transformer

Khi nói về ViT, chúng ta có thể thấy rõ ràng những lợi ích và hạn chế của mô hình này, đặc biệt là khi so sánh với các phương pháp học sâu truyền thống như CNN.

Một trong những điểm mạnh nổi bật của Vision Transformer là khả năng nắm bắt các mối quan hệ toàn cục trong hình ảnh. Khác với các CNN, vốn chủ yếu tập trung vào các đặc trưng cục bộ qua các lớp tích chập, ViT sử dụng cơ chế attention để học các mối liên hệ giữa các patch của hình ảnh. Điều này giúp mô hình hiểu rõ hơn về cấu trúc tổng thể của hình ảnh. Ví dụ, trong các bài toán phân loại hình ảnh phức tạp như nhận diện cảnh vật hoặc phân tích chi tiết các đối tượng trong bức tranh lớn, ViT có thể cung cấp sự hiểu biết sâu hơn và chính xác hơn về toàn bộ bức tranh.

Thêm vào đó, ViT cho thấy hiệu suất cao khi được đào tạo trên các tập dữ liệu lớn. Nghiên cứu của Google Brain cho thấy ViT đạt hiệu suất ấn tượng trên các bộ dữ liệu quy mô lớn như ImageNet khi có đủ tài nguyên và dữ liệu để huấn luyện. Điều này cho thấy rằng ViT có khả năng khai thác tối đa thông tin từ lượng dữ liệu phong phú.

ViT cũng có khả năng mở rộng tốt với kích thước mô hình và dữ liệu. Khi mô hình và dữ liệu lớn hơn, ViT thường tiếp tục cải thiện hiệu suất, điều này khác biệt so với các CNN truyền thống, nơi việc mở rộng kích thước mô hình có thể gặp nhiều thách thức. Chẳng hạn, khi áp dụng ViT cho các bài toán phân tích hình ảnh trong các ứng dụng yêu cầu độ phân giải cao hoặc số lượng lớp lớn, mô hình này cho thấy sự gia tăng hiệu suất rõ rệt.

Tuy nhiên, Vision Transformer không phải là không có hạn chế. Một trong những thách thức lớn nhất là yêu cầu về tài nguyên tính toán. ViT yêu cầu nhiều tài nguyên tính toán hơn so với các CNN, đặc biệt khi xử lý các mô hình lớn. Điều này có thể tạo ra khó khăn trong việc triển khai trên các hệ thống có tài nguyên hạn chế, như các thiết bị di động hoặc máy tính cá nhân. Ví dụ, việc huấn luyện ViT trên các GPU cấu hình cao hoặc các cụm máy chủ mạnh mẽ có thể làm tăng chi phí và thời gian huấn luyện.

Bên cạnh đó, ViT cũng cần một lượng dữ liệu lớn để đạt hiệu suất tối ưu. Trong các tình huống với dữ liệu hạn chế, ViT có thể không đạt được kết quả tốt như mong muốn, điều này trái ngược với các CNN có thể hoạt động tốt hơn trên các tập dữ liệu nhỏ hơn nhờ khả năng học các đặc trưng cục bộ hiệu quả hơn.

Cuối cùng, mặc dù ViT có kiến trúc đơn giản hơn so với các CNN truyền thống, điều này không đồng nghĩa với việc nó luôn dễ dàng trong việc triển khai hoặc tối ưu hóa. ViT có thể cần sự điều chỉnh tinh vi và tối ưu hóa đặc biệt để đạt hiệu suất tốt nhất trong các ứng dụng thực tế.

Nhìn chung, ViT cung cấp một cách tiếp cận mới mẻ và mạnh mẽ cho các bài toán phân loại hình ảnh, nhưng nó cũng đi kèm với các yêu cầu về tài nguyên và dữ liệu cần được cân nhắc khi áp dụng trong thực tế.

2.2. Ứng dụng của Vision Transformer trong nhận diện sản phẩm

2.2.1. Phát biểu bài toán

Bài toán Nhận diện sản phẩm là một bài toán trong lĩnh vực xử lý ảnh và thị giác máy tính, nơi mà mục tiêu là xác định và phân loại các sản phẩm khác nhau từ hình ảnh:

Input: Một hình ảnh hình ảnh chứa sản phẩm cần nhận diện. Có thể bao gồm hình ảnh chụp từ các góc độ khác nhau, trong điều kiện ánh sáng khác nhau hoặc các bối cảnh phức tạp.

Output: Sản phẩm được nhận diện trong hình ảnh để phục vụ cho việc thanh toán.

Có thể thấy, nhận diện sản phẩm là một lĩnh vực quan trọng trong công nghệ thông tin, đặc biệt khi áp dụng vào quy trình thanh toán tại các cửa hàng tiện lợi hoặc siêu thị. Mục tiêu của nhận diện sản phẩm trong bối cảnh này là xác định các sản phẩm từ hình ảnh, giúp tự động hóa và tối ưu hóa quy trình thanh toán.

Trong các cửa hàng truyền thống, quy trình thanh toán thường yêu cầu nhân viên thu ngân phải quét mã vạch hoặc nhập mã sản phẩm bằng tay, điều này không

chỉ tốn thời gian mà còn có nguy cơ sai sót. Với sự phát triển của các mô hình như ViT, hệ thống nhận diện sản phẩm có khả năng phân tích hình ảnh nhanh chóng, xác định sản phẩm và tính toán giá trị tổng cộng mà không cần đến sự can thiệp của con người. Quá trình này giúp giảm thiểu thời gian chờ đợi của khách hàng, đồng thời tăng cường hiệu quả hoạt động của cửa hàng..

Nhìn chung, nhận diện sản phẩm đóng vai trò then chốt trong việc hiện đại hóa quy trình thanh toán, mang lại lợi ích to lớn cho cả khách hàng và doanh nghiệp. Với sự hỗ trợ của các công nghệ tiên tiến như VIT, quy trình thanh toán không chỉ trở nên nhanh chóng và tiện lợi hơn, mà còn tạo ra một trải nghiệm mua sắm thông minh và hiệu quả.

2.2.2. Các phương pháp nhận diện sản phẩm truyền thống

Các phương pháp nhận diện sản phẩm truyền thống thường dựa vào các kỹ thuật học máy và thị giác máy tính, bao gồm nhiều phương pháp khác nhau:

- **Phát hiện và mô tả đặc trưng [10]:** Những kỹ thuật như SIFT (Scale-Invariant Feature Transform) và SURF (Speeded-Up Robust Features) đã được sử dụng để phát hiện các điểm đặc trưng trong hình ảnh và mô tả chúng một cách hiệu quả. Tuy nhiên, những phương pháp này thường yêu cầu quá trình tiền xử lý và điều chỉnh tham số tốn thời gian.
- **Phân loại dựa trên các đặc trưng [11]:** Sau khi phát hiện, các đặc trưng được sử dụng để huấn luyện các mô hình học máy như SVM (Support Vector Machine) hoặc k-NN (k-Nearest Neighbors) nhằm phân loại sản phẩm. Mặc dù những phương pháp này đã đạt được một số thành công nhất định, nhưng chúng thường không đủ mạnh mẽ để xử lý các hình ảnh phức tạp trong môi trường thực tế.
- **Deep Learning với CNNs [12][13]:** Mạng nơ-ron tích chập (Convolutional Neural Networks - CNN) đã trở thành tiêu chuẩn vàng trong nhận diện hình ảnh. CNN tự động học các đặc trưng từ dữ liệu hình ảnh mà không cần phải thiết kế thủ công, giúp tăng cường độ chính xác và khả năng tổng quát của mô

hình. Tuy nhiên, CNN vẫn gặp một số hạn chế trong việc nắm bắt các mối quan hệ không gian phức tạp giữa các đối tượng trong hình ảnh.

2.2.3. Ứng dụng Vision Transformer trong nhận diện sản phẩm

ViT là một kiến trúc mạng nơ-ron mới, được phát triển dựa trên khái niệm Transformer, vốn được áp dụng thành công trong lĩnh vực xử lý ngôn ngữ tự nhiên.

ViT xử lý hình ảnh bằng cách chia chúng thành các patch nhỏ và xem xét chúng như một chuỗi, điều này cho phép mô hình nắm bắt được các mối quan hệ không gian phức tạp hơn so với các phương pháp truyền thống.

ViT đã chứng minh khả năng vượt trội trong việc nhận diện hình ảnh, đặc biệt là trong các nhiệm vụ yêu cầu xử lý hình ảnh lớn và đa dạng.

Ví dụ: Trong một dự án nhận diện sản phẩm, quy trình áp dụng VIT có thể được thực hiện như sau:

- **Chuẩn bị dữ liệu:** Hình ảnh sản phẩm được chia thành các patch nhỏ, với kích thước bằng nhau. Điều này cho phép mô hình phân tích chi tiết từng phần của hình ảnh và phát hiện các đặc trưng quan trọng.
- **Huấn luyện mô hình:** ViT sẽ được huấn luyện trên một tập dữ liệu lớn gồm nhiều hình ảnh sản phẩm khác nhau, nhằm học các đặc trưng quan trọng và nhận diện các mẫu trong dữ liệu. Quá trình này thường bao gồm việc tối ưu hóa các tham số của mô hình để đạt được độ chính xác cao nhất.
- **Nhận diện:** Sau khi huấn luyện, mô hình có thể sử dụng để dự đoán nhãn sản phẩm cho hình ảnh đầu vào mới. Điều này không chỉ giúp tăng cường độ chính xác mà còn giảm thời gian xử lý trong các ứng dụng thực tế.

2.2.4. Các nghiên cứu và ứng dụng liên quan

Nhiều nghiên cứu và ứng dụng đã chỉ ra khả năng và tiềm năng to lớn của ViT trong lĩnh vực nhận diện sản phẩm. Dưới đây là một số nghiên cứu tiêu biểu và các ứng dụng liên quan:

Trong nghiên cứu của Dosovitskiy et al. (2020) [1], nhóm tác giả đã giới thiệu mô hình ViT, một mô hình mới dựa trên Transformer, thay vì sử dụng CNN truyền thống. Họ đã chia hình ảnh thành patches nhỏ có kích thước 16x16 pixel, tương đương với các từ (tokens) trong ngôn ngữ tự nhiên. Các patches này sau đó được đưa qua kiến trúc Transformer để học các mối quan hệ toàn cục giữa các vùng khác nhau của hình ảnh. Kết quả thực nghiệm cho thấy mô hình ViT đã đạt hiệu suất tương đương hoặc thậm chí vượt trội hơn so với các mô hình CNN hiện đại khi được huấn luyện trên các tập dữ liệu lớn như ImageNet. Điều này chứng tỏ rằng các mô hình Transformer, ban đầu được thiết kế cho ngôn ngữ tự nhiên, có thể được mở rộng để xử lý hình ảnh với độ chính xác cao hơn trong nhiều trường hợp.

Trong nghiên cứu về phân loại hình ảnh tinh vi, Fine-Grained Image Classification Using Vision Transformers (2021) [5], các tác giả đã áp dụng Vision Transformer vào việc phân loại các đối tượng có sự khác biệt rất nhỏ về đặc điểm. Nghiên cứu đã thử nghiệm ViT trên các tập dữ liệu phân loại giống chim, cây, và các sản phẩm thương mại điện tử với những đặc điểm tinh tế khác nhau. Kết quả thực nghiệm cho thấy ViT có khả năng học các đặc trưng chi tiết và tinh vi của các đối tượng, giúp mô hình phân loại chính xác các đối tượng có sự khác biệt nhỏ. Điều này chứng tỏ ViT có tiềm năng ứng dụng cao trong các ngành như thời trang, nhận diện động vật, và phân loại sản phẩm công nghệ.

Trong nghiên cứu về ứng dụng Vision Transformer trong thương mại điện tử, Transforming E-Commerce with Vision Transformers (2022) [6], các tác giả đã tập trung vào việc tích hợp ViT vào hệ thống gợi ý sản phẩm và quy trình kiểm kê hàng hóa. Nghiên cứu cho thấy ViT được sử dụng để nhận diện và phân loại sản phẩm từ hình ảnh trong quản lý tồn kho và hệ thống gợi ý sản phẩm. Kết quả thực nghiệm cho thấy việc áp dụng ViT đã cải thiện đáng kể độ chính xác trong việc nhận diện sản phẩm, nâng cao hiệu quả của hệ thống gợi ý và quản lý tồn kho, từ đó giúp các công ty tối ưu hóa chuỗi cung ứng và giảm thiểu sai sót trong kiểm kê hàng hóa.

Trong nghiên cứu về phát hiện đối tượng, End-to-End Object Detection with Transformers (2022) [7], các tác giả đã giới thiệu cách Vision Transformer có

thể được áp dụng cho nhiệm vụ phát hiện đối tượng. Nghiên cứu thử nghiệm ViT trên các tập dữ liệu phát hiện đối tượng, nơi mà các sản phẩm có thể bị che khuất hoặc khó phân biệt. Kết quả thực nghiệm cho thấy ViT đã cải thiện đáng kể khả năng phát hiện đối tượng so với các mô hình trước đây, cho phép nhận diện các đối tượng ngay cả trong các tình huống khó khăn như sản phẩm bị xếp chồng lên nhau hoặc có màu sắc và kích thước gần giống nhau.

Trong nghiên cứu về ứng dụng Vision Transformer trong bán lẻ, Deep Learning for Retail Inventory Management (2021) [8], các tác giả đã áp dụng ViT vào việc theo dõi và quản lý hàng hóa trong môi trường bán lẻ. Nghiên cứu cho thấy việc sử dụng ViT trong các hệ thống theo dõi tự động từ nhập hàng, quản lý tồn kho đến bán sản phẩm đã giúp tối ưu hóa quá trình kiểm kê và quản lý hàng hóa. Kết quả thực nghiệm cho thấy ViT đã giúp các cửa hàng theo dõi chính xác lượng hàng tồn kho, giảm thiểu sai sót và nâng cao trải nghiệm khách hàng nhờ việc cung cấp thông tin hàng tồn kho chính xác hơn.

Trong nghiên cứu về ứng dụng Vision Transformer trong y tế, Applications of Vision Transformers in Medical Imaging (2023) [9], các tác giả đã áp dụng ViT vào việc phân tích các hình ảnh y tế như ảnh chụp X-quang và MRI. Nghiên cứu cho thấy ViT có khả năng phát hiện các bất thường hoặc dấu hiệu bệnh từ các hình ảnh y tế với độ chính xác cao. Kết quả này không chỉ hỗ trợ các bác sĩ trong việc phát hiện sớm bệnh tật mà còn mở ra khả năng ứng dụng của ViT trong các ngành công nghiệp khác như kiểm tra chất lượng sản phẩm hoặc giám sát sản xuất.

Những nghiên cứu và ứng dụng này không chỉ chứng minh được hiệu quả của Vision Transformer trong lĩnh vực nhận diện sản phẩm mà còn mở ra nhiều cơ hội và hướng đi mới cho việc phát triển công nghệ trong tương lai.

Việc tham khảo các tài liệu này sẽ giúp bạn có cái nhìn sâu sắc hơn về cách ViT đang thay đổi cách chúng ta nhận diện và phân loại sản phẩm trong nhiều lĩnh vực khác nhau.

CHƯƠNG 3. BÀI TOÁN PHÂN LOẠI SẢN PHẨM

3.1. Phát biểu bài toán

Bài toán nhận diện sản phẩm trong cửa hàng tiện lợi là một nhánh trong Thị giác máy tính, nhằm xác định và phân loại các sản phẩm dựa trên hình ảnh của chúng. Điều này giúp nhận diện các sản phẩm trong cửa hàng, quản lý hàng hóa hiệu quả, và cải thiện trải nghiệm mua sắm của khách hàng bằng cách tự động hóa các quy trình nhận diện và phân loại sản phẩm.

Các loại nhận dạng sản phẩm:

Phân loại sản phẩm (Product classification): Xác định loại sản phẩm dựa trên các danh mục đã định sẵn (ví dụ: đồ uống, thực phẩm đóng gói, vật dụng cá nhân).

Phát hiện đối tượng (Object detection): Xác định vị trí chính xác của sản phẩm trong hình ảnh thông qua hộp giới hạn (bounding box).

Nhận dạng nhãn hiệu (Brand recognition): Nhận diện và phân loại các nhãn hiệu cụ thể của sản phẩm.

Nhận dạng chi tiết (Fine-grained recognition): Phân biệt giữa các sản phẩm có hình dạng và đặc điểm tương tự nhưng thuộc các loại hoặc nhãn hiệu khác nhau. Có hai phương pháp chính để giải quyết bài toán nhận dạng sản phẩm:

- **Có hai phương pháp chính để giải quyết bài toán nhận dạng sản phẩm:**

Dựa trên đặc trưng thủ công (Hand-crafted features): Sử dụng các kỹ thuật xử lý ảnh truyền thống để trích xuất đặc trưng như màu sắc, hình dạng, kết cấu, sau đó áp dụng các thuật toán phân loại.

Dựa trên học sâu (Deep Learning based): Xây dựng mô hình mạng nơ-ron sâu để học các đặc trưng từ dữ liệu huấn luyện, từ đó tiến hành nhận dạng sản phẩm. Một số kiến trúc phổ biến được sử dụng như CNN, YOLO, Faster R-CNN và gần đây ViT đã góp phần lớn để giải quyết bài toán nhận dạng.

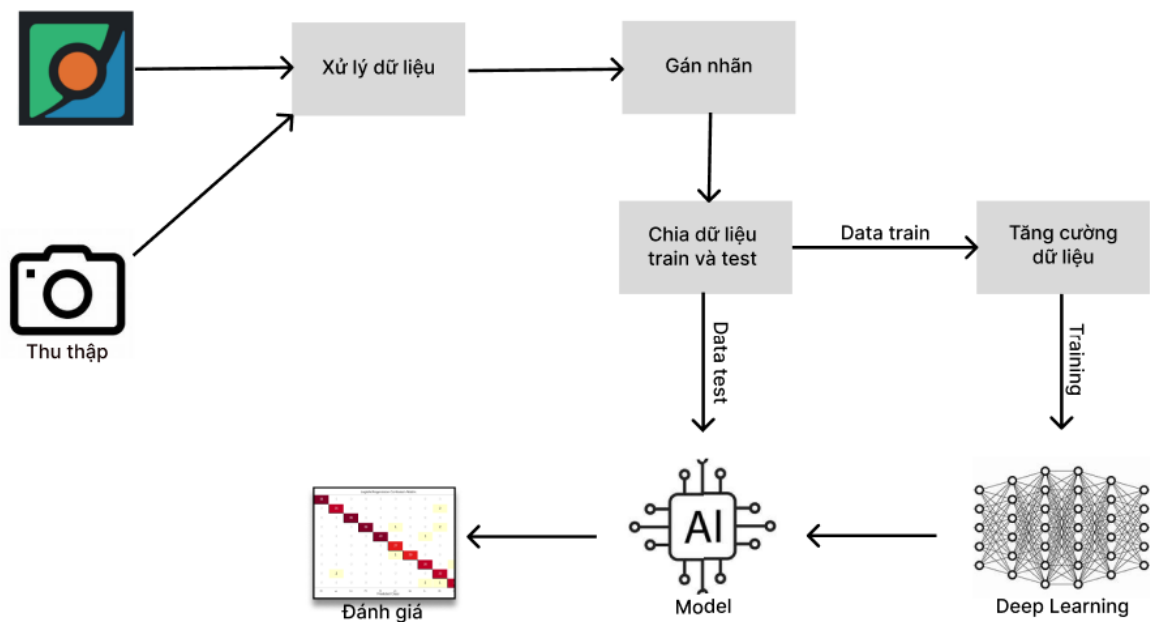
Bảng 2. Bảng so sánh hai phương pháp nhận dạng sản phẩm

Đặc điểm	Dựa trên đặc trưng thủ công	Dựa trên học sâu
Cách tiếp cận	<ul style="list-style-type: none"> - Sử dụng kỹ thuật xử lý ảnh truyền thống - Trích xuất đặc trưng thủ công (màu sắc, hình dạng, kết cấu) - Áp dụng thuật toán phân loại cổ điển 	<ul style="list-style-type: none"> - Xây dựng mô hình mạng nơ-ron sâu - Tự động học đặc trưng từ dữ liệu - Sử dụng kiến trúc như CNN, YOLO, Faster R-CNN
Hiệu suất	<ul style="list-style-type: none"> - Hiệu quả với bộ dữ liệu nhỏ và đơn giản - Có thể gặp khó khăn với dữ liệu phức tạp hoặc đa dạng 	<ul style="list-style-type: none"> - Hiệu quả cao với bộ dữ liệu lớn và phức tạp - Khả năng tổng quát hóa tốt hơn
Yêu cầu dữ liệu	<ul style="list-style-type: none"> - Yêu cầu ít dữ liệu huấn luyện hơn - Có thể hoạt động tốt với bộ dữ liệu nhỏ 	<ul style="list-style-type: none"> - Yêu cầu lượng lớn dữ liệu huấn luyện - Cần dữ liệu đa dạng để tránh overfitting
Thời gian và tài nguyên	<ul style="list-style-type: none"> - Ít tốn thời gian và tài nguyên để xây dựng mô hình - Quá trình suy luận nhanh hơn 	<ul style="list-style-type: none"> - Tốn nhiều thời gian và tài nguyên để huấn luyện mô hình - Có thể yêu cầu phần cứng chuyên dụng (GPU)
Khả năng tùy chỉnh	<ul style="list-style-type: none"> - Dễ dàng điều chỉnh và tối ưu hóa cho từng trường hợp cụ thể - Có thể kết hợp kiến thức chuyên môn 	<ul style="list-style-type: none"> - Khó điều chỉnh chi tiết, phụ thuộc vào dữ liệu huấn luyện - Hoạt động như "hộp đen", khó giải thích

Ứng dụng	<ul style="list-style-type: none"> - Phù hợp cho các bài toán đơn giản hoặc có ràng buộc về tài nguyên - Hữu ích trong các môi trường có ít dữ liệu 	<ul style="list-style-type: none"> - Phù hợp cho các bài toán phức tạp, cần độ chính xác cao - Hiệu quả trong môi trường có nhiều dữ liệu và đa dạng sản phẩm
-----------------	---	---

Trong bài nghiên cứu này, nhóm chúng tôi thực hiện xây dựng mô hình nhận dạng sản phẩm **Phân loại sản phẩm (Product classification)** với chín loại sản phẩm (**Blue Lays, Coca Cola, Colgate, Fanta, Laybuoy, Lays Wawy, Safeguard, Sunsil, YẾN**), sử dụng **Deep Learning** cụ thể là ViT và áp dụng nó vào nhận dạng sản phẩm ở cửa hàng tiện lợi giúp đơn giản hóa việc thanh toán.

Sơ đồ tổng quát quá trình hiện thực như sau:



Hình 5. Hình ảnh sơ đồ thực hiện phân tích dữ liệu

Bước 1: Thu thập dữ liệu từ COCO Dataset và tự thu thập thêm từ thực tế.

- Tải các phần cần thiết như ảnh, tệp gán nhãn (annotations) cho các nhiệm vụ như phát hiện đối tượng, phân đoạn, và nhận diện đối tượng phù hợp với sản phẩm.

- Thu thập thêm data từ các cửa hàng tiện lợi để nhận được thêm dữ liệu thực tế hơn.

Bước 2: Gán nhãn cho dữ liệu nếu chưa có.

- Đối với dữ liệu thu thập chúng ta cần đảm bảo nhãn phải phù hợp với sản phẩm tránh trường hợp sai nhãn.
- Khi thu thập từ nhiều COCO dataset thì cần phải điều chỉnh nhãn cho phù hợp với dữ liệu thu thập.

Bước 3: Xử lý dữ liệu đầu vào phù hợp hệ thống.

- Xử lý dữ liệu đầu vào tránh bị nhiễu hay bị lệnh quá nhiều.
- Điều chỉnh kích thước phù hợp với input của mô hình.

Bước 4: Chia dữ liệu thành dùng để huấn luyện và kiểm tra.

- Tập huấn luyện (Training Set): Sử dụng phần lớn dữ liệu để huấn luyện mô hình (thường là 80% của tổng số dữ liệu).
- Tập kiểm tra (Testing Set): Sử dụng phần còn lại của dữ liệu để đánh giá mô hình (20%).

Bước 5: Tăng cường dữ liệu dùng để huấn luyện bằng cách sử dụng các layers có sẵn của TensorFlow giúp dữ liệu đa dạng hơn.

Bước 6: Xây dựng model và huấn luyện trên data train.

- Xây dựng mô hình phù hợp với yêu cầu.
- Sử dụng tập dữ liệu huấn luyện để cập nhật trọng số và điều chỉnh mô hình qua các epoch.
- Quan sát quá trình huấn luyện qua độ mất mát và độ chính xác để điều chỉnh các tham số nếu cần.

Bước 7: Dùng data test để đánh giá mô hình sau huấn luyện để kiểm tra khả năng tổng quát trên dữ liệu chưa thấy.

3.2. Các phương pháp giải quyết bài toán

Dựa vào ưu và nhược điểm khi so sánh 2 phương pháp có thể xử lý bài toán nhận diện như ở trên cùng đó là bài toán này cần đòi hỏi độ chính xác cao ta thấy được rằng phương pháp học sâu là phù hợp nhất đối với bài toán này. Về vấn đề nhận

dạng sản phẩm thông qua hình ảnh hiện nay phát triển khá nhiều và phương pháp phổ biến nhất CNN nó gần như là lựa chọn hàng đầu khi nhắc đến xử lý ảnh.

Tuy nhiên hiện nay còn có một phương pháp khác tốt hơn và có thể thay thế được là ViT thông qua bài báo **AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE** [1] ở đây nó chỉ ra được khả năng vượt trội của ViT so với CNN.

ViT tận dụng sức mạnh của Transformer, một mô hình đã thành công vang dội trong xử lý ngôn ngữ tự nhiên, và áp dụng vào xử lý hình ảnh. Ưu điểm chính của ViT so với CNN nằm ở khả năng nhận dạng các mẫu phức tạp và khả năng tổng quát hóa cao hơn, đặc biệt khi làm việc với các hình ảnh có độ phân giải cao. Điều này rất quan trọng đối với bài toán nhận dạng sản phẩm của cửa hàng tiện lợi, nơi yêu cầu độ chính xác cực kỳ cao.

Với những lợi thế đó, chúng tôi tin rằng việc áp dụng ViT thay vì CNN truyền thống cho bài toán nhận dạng sản phẩm trong cửa hàng tiện lợi sẽ mang lại hiệu quả cao hơn, góp phần nâng cao chất lượng dịch vụ và tối ưu hóa quá trình nhận diện sản phẩm.

3.3. Xây dựng mô hình Vision Transformer

3.3.1. Giới thiệu dataset

Trong nghiên cứu này, chúng tôi sử dụng từ nhiều nguồn COCO dataset và tự thu thập thêm data để huấn luyện mô hình Vision Transformer nhận dạng sản phẩm cửa hàng tiện lợi. COCO là một bộ dữ liệu phổ biến trong lĩnh vực thị giác máy tính, cung cấp hình ảnh đa dạng với hơn 1.5 triệu đối tượng được gán nhãn và phân đoạn chi tiết với các mask pixel cho từng đối tượng.

COCO dataset bao gồm các thành phần chính sau:

- Hình ảnh: Chứa thông tin về các hình ảnh như ID, đường dẫn file, kích thước.
- Chú thích (Annotations): Chứa thông tin về các đối tượng trong hình ảnh, bao gồm bounding box, ID hình ảnh, và ID category.

- **Danh mục (Categories):** Chứa thông tin về các loại đối tượng, bao gồm ID và tên.
- **Giấy phép (Licenses):** Chứa thông tin về giấy phép sử dụng của hình ảnh.

Hệ thống thu thập dữ liệu tự động: Nhóm chúng tôi đã xây dựng một hệ thống tiên tiến nhằm mục đích thu thập dữ liệu nhanh chóng và hiệu quả hơn. Hệ thống này sử dụng máy ảnh để nhận diện và thu thập hình ảnh đầu vào. Khi hình ảnh được nhập vào hệ thống, nó sẽ được xử lý để sinh thêm các biến thể khác nhau, từ đó tạo ra một tập hợp dữ liệu phong phú hơn.

Quá trình hoạt động của hệ thống bao gồm:

- **Nhận diện và thu thập hình ảnh:** Máy ảnh của điện thoại được sử dụng để chụp hình ảnh đầu vào, có thể là từ các nguồn khác nhau .
- **Xử lý hình ảnh:** Sau khi hình ảnh được thu thập, hệ thống sẽ tiến hành xử lý để sinh thêm các biến thể. Điều này có thể bao gồm việc thay đổi độ sáng, độ tương phản, hoặc áp dụng các bộ lọc khác để tạo ra nhiều phiên bản khác nhau của cùng một hình ảnh.
- **Tạo và lưu trữ dữ liệu:** Các hình ảnh biến thể được tạo ra sẽ được lưu trữ theo cấu trúc dataset đã được quy định trước đó. Cấu trúc này đảm bảo rằng dữ liệu được tổ chức một cách hệ thống và dễ dàng truy cập, phân tích trong tương lai.

Ở đây ta có 9 label tập trung vào các sản phẩm của một cửa hàng tiện lợi được phân đoạn hình ảnh và gán nhãn cụ thể cho từng sản phẩm ta có được 10583 data train và 1375 data dùng để test.

3.2.2. Thu thập và tiền xử lý dữ liệu

Để xử lý dữ liệu từ nhiều COCO dataset, chúng tôi đã phát triển lớp **COCOParser** với các chức năng chính đọc dữ liệu từ hình từ và label của dataset và thực hiện chuyển đổi label của nó thành label theo quy định riêng của chúng tôi thông

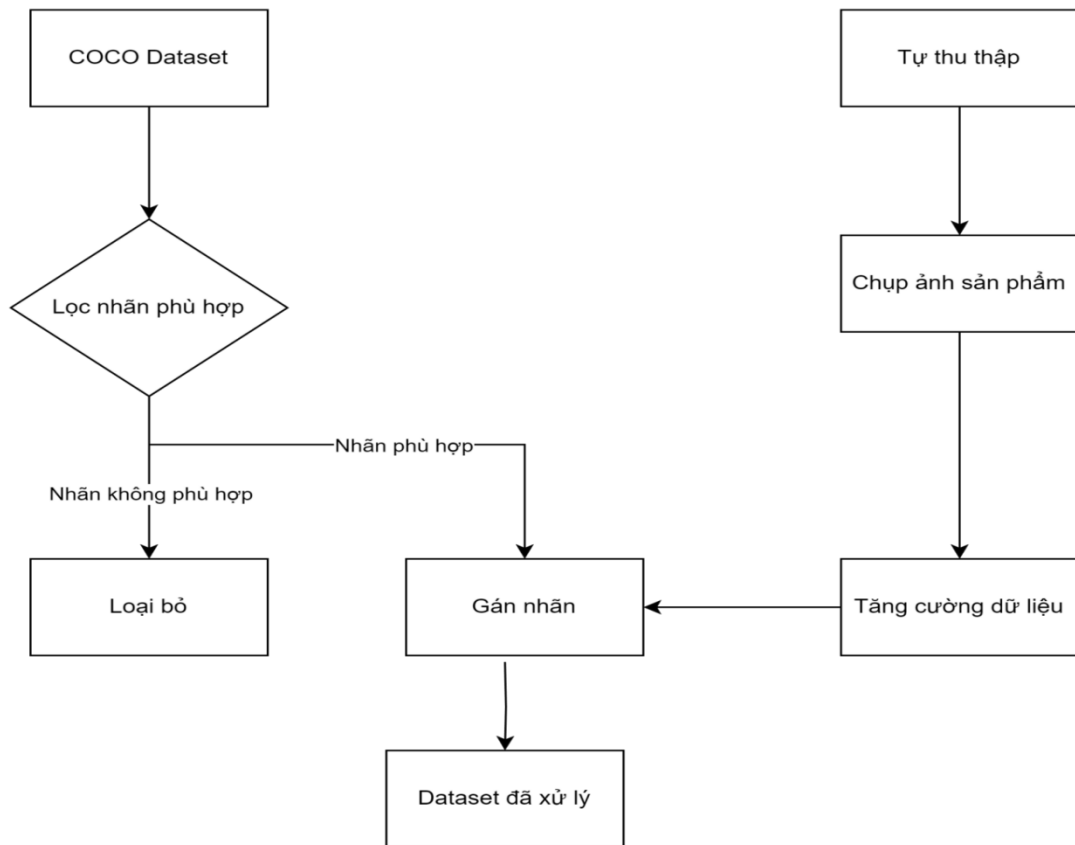
qua **JoinDataset** class này phát triển giúp chúng tôi có thể chuyển đổi các sản phẩm của các dataset về chung một label mà chúng tôi quy định từ trước.

Bằng cách chúng tôi thêm vào các thư mục chứa dataset của COCO 1 file mapping.csv có tác dụng chuyển đổi các quy định sẵn các label cần đổi lại cho phù hợp với dataset của chúng tôi. Như thế này khi tôi có 3 bộ dataset khác nhau ở dataset đầu tiên sản phẩm Fanta có label là 4 nhưng ở dataset khác Fanta được đánh là 10 thì dựa vào file mapping tôi có thể map Fanta của các bộ dataset về thành label là 8 được tôi quy định.

Bên cạnh nhóm chúng tôi đã thu thập thêm dữ liệu từ thực tế, sử dụng máy ảnh để nhận diện và thu thập hình ảnh đầu vào. Khi hình ảnh được nhập vào hệ thống, nó sẽ được xử lý để sinh thêm các biến thể khác nhau, từ đó tạo ra một tập hợp dữ liệu phong phú hơn.

Quá trình hoạt động của hệ thống bao gồm:

- Nhận diện và thu thập hình ảnh: Máy ảnh được sử dụng để chụp hình ảnh đầu vào, có thể là từ các nguồn khác nhau như tài liệu, sản phẩm, hoặc các đối tượng cần phân tích.
- Xử lý hình ảnh: Sau khi hình ảnh được thu thập, hệ thống sẽ tiến hành xử lý để sinh thêm các biến thể. Điều này có thể bao gồm việc thay đổi độ sáng, độ tương phản, hoặc áp dụng các bộ lọc khác để tạo ra nhiều phiên bản khác nhau của cùng một hình ảnh.
- Tạo và lưu trữ dữ liệu: Các hình ảnh biến thể được tạo ra sẽ được lưu trữ theo cấu trúc dataset đã được quy định trước đó. Cấu trúc này đảm bảo rằng dữ liệu được tổ chức một cách hệ thống và dễ dàng truy cập, phân tích trong tương lai.



Hình 6. Sơ đồ quy trình thu thập dữ liệu

- Hình ảnh được chuẩn hóa về kích thước cố định (được chỉ định bởi tham số size).
- Giá trị pixel được chuẩn hóa về khoảng $[0, 1]$.
- Mỗi hình ảnh được gán nhãn là ID của category tương ứng.
- Thực hiện tăng cường dữ liệu giúp đa dạng hóa tập dữ liệu huấn luyện

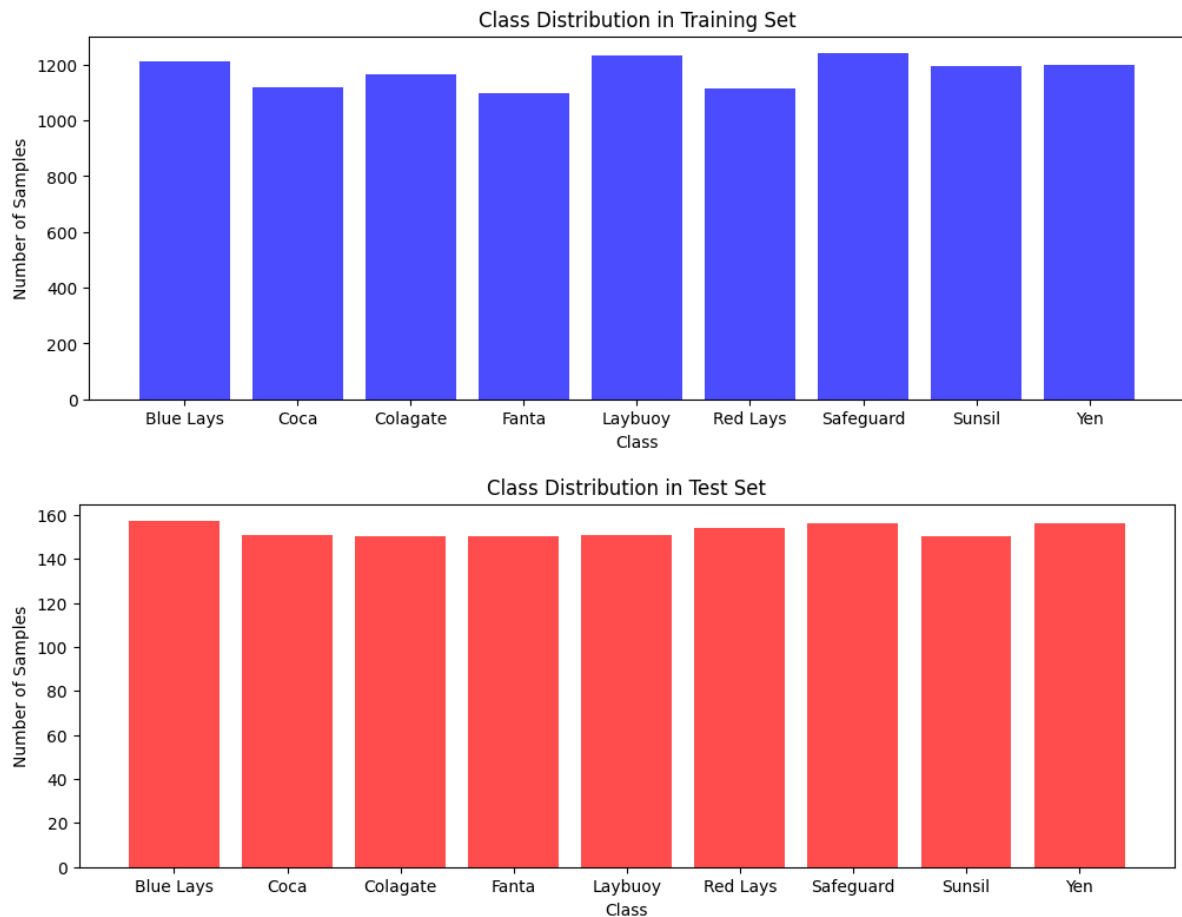
Việc xử lý dữ liệu này cho phép chúng tôi chuẩn bị một bộ dữ liệu phù hợp để huấn luyện mô hình Vision Transformer, đảm bảo tính nhất quán về kích thước và định dạng của dữ liệu đầu vào.

3.2.3. Phân tích phân phối dữ liệu huấn luyện

Để hiểu rõ hơn về sự phân bố của các lớp trong tập dữ liệu, chúng tôi đã tiến hành phân tích và trực quan hóa số lượng mẫu cho mỗi lớp. Kết quả được thể hiện thông qua biểu đồ cột như sau:

Bảng 1: Bảng mô tả tập dữ liệu

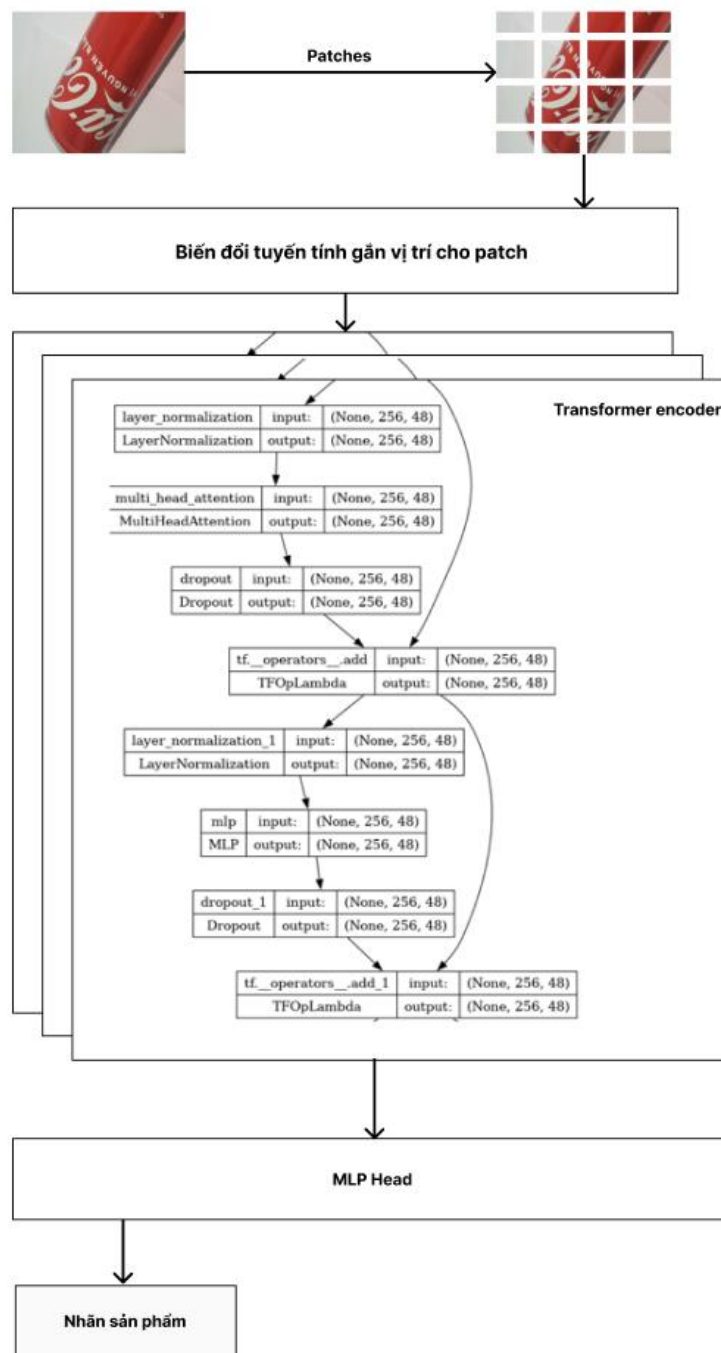
Thuộc tính	Mô tả
Số lượng ảnh	11.958
Số lớp	9
Kích thước ảnh	64x64 pixels
Số lượng tập huấn luyện	10.583
Số lượng tập kiểm tra	1.375
Tên lớp	0: Blue Lays, 1: Coca Cola, 2: Colgate, 3: Fanta, 4: Laybuoy, 5: Lays Wawy, 6: Safeguard, 7: Sunsil, 8: Yen



Hình 7. Sơ đồ Phân tích phân phối dữ liệu huấn luyện

3.2.4. Xây dựng và huấn luyện mô hình

Trong nghiên cứu này, chúng tôi đã xây dựng một mô hình ViT tùy chỉnh để nhận dạng sản phẩm cửa hàng tiện lợi. Dưới đây là chi tiết mô hình về cấu trúc và quá trình xây dựng:



Hình 8. Cấu trúc mô hình ViT

Mô hình được biên dịch với các cấu hình sau:

- **Optimizer**: AdamW với learning rate ban đầu là $1e-3$ là một biến thể của optimizer Adam, có khả năng tự điều chỉnh learning rate và thường cho kết quả tốt với các mô hình Transformer.

- **Loss function:** SparseCategoricalCrossentropy phù hợp với bài toán phân loại nhiều lớp, đặc biệt khi labels không được one-hot encoded.
- **Metric:** Accuracy là metric trực quan để đánh giá hiệu suất của mô hình phân loại.
- **Callbacks:** Sử dụng ModelCheckpoint và EarlyStopping
 - ModelCheckpoint là một callback dùng để lưu trữ trạng thái của mô hình trong quá trình huấn luyện. Điều này rất hữu ích khi muốn lưu lại các mô hình trong suốt quá trình huấn luyện để có thể tiếp tục huấn luyện hoặc sử dụng mô hình tốt nhất sau khi hoàn tất huấn luyện.
 - EarlyStopping là một callback giúp dừng huấn luyện sớm khi mô hình không còn cải thiện được nữa. Điều này giúp ngăn ngừa hiện tượng overfitting và giảm thời gian huấn luyện không cần thiết.

Trong quá trình phát triển mô hình, chúng tôi đã thử nghiệm với nhiều cấu hình khác nhau của các tham số trên. Một số nhận xét:

- Giảm **patch_size** giúp mô hình học được các đặc trưng chi tiết hơn, nhưng cũng làm tăng đáng kể độ phức tạp tính toán.
- Điều chỉnh **mlp_head_units** và **transformer_layers** ảnh hưởng đến khả năng học các biểu diễn phức tạp của mô hình.
- Tăng dropout giúp giảm overfitting nhưng nếu quá cao có thể làm giảm khả năng học của mô hình.

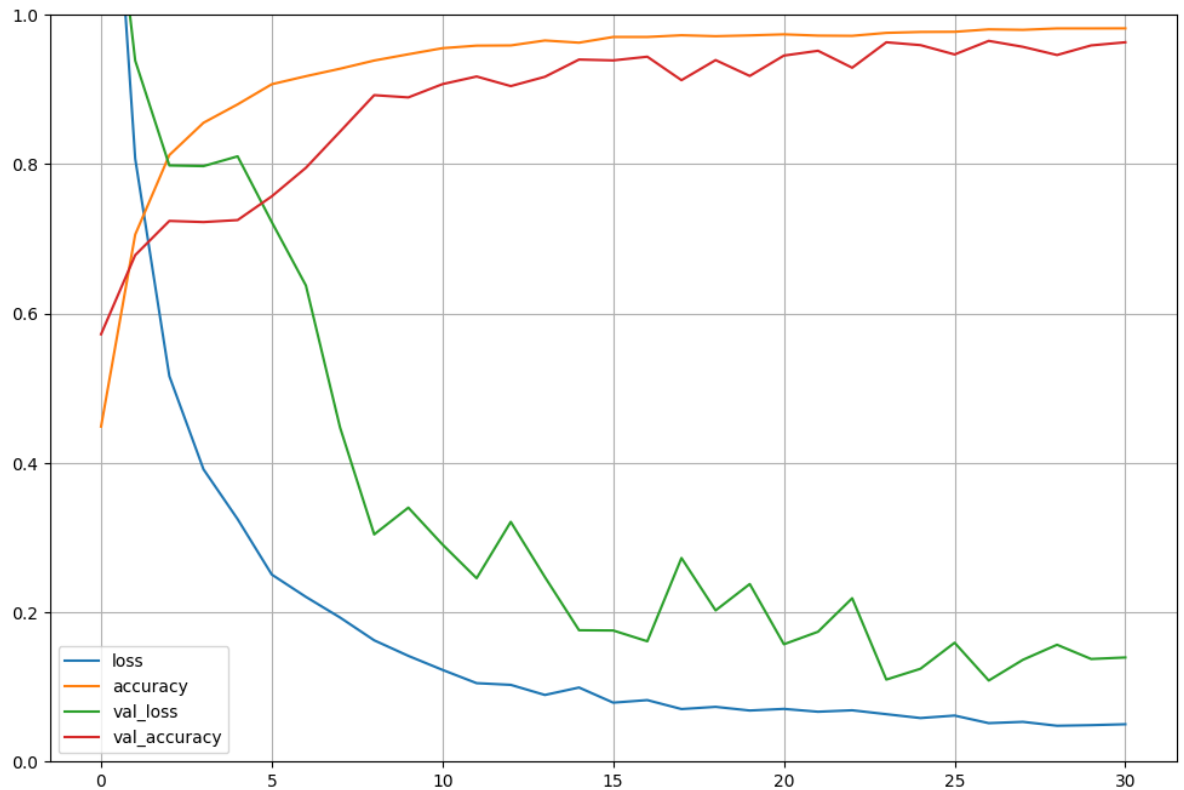
Ta thu được kết quả tốt nhất khi model huấn luyện trên dataset trên với các tham số sau:

Bảng 3. Tham số huấn luyện mô hình tốt nhất

Param	Giá trị
input_shape (Kích thước ảnh đầu vào)	64x64x3
patch_size (Kích thước của mỗi patch)	4
num_heads (Số lượng heads trong Multi-Head Attention)	10
transformer_layers (Số lượng lớp Transformer Encoder)	6
mlp_head_units (Số lượng units trong MLP Head)	[1024, 512]

3.2.5. Kết quả thực nghiệm và đánh giá

Sau khi hoàn tất quá trình huấn luyện mô hình Vision Transformer, chúng tôi tiến hành đánh giá mô hình dựa trên các chỉ số hiệu suất chính và biểu đồ đường học để hiểu rõ hơn về khả năng hoạt động và hiệu quả của mô hình.



Hình 9. Biểu đồ loss và accuracy của mô hình

Kết quả huấn luyện mô hình Vision Transformer cho thấy hiệu suất ấn tượng và cải thiện đáng kể qua các epoch. Trong suốt quá trình huấn luyện, mô hình đã đạt được những chỉ số quan trọng như sau:

- Hiệu suất Huấn luyện:** Mô hình bắt đầu với độ chính xác khoảng 41.5% và mất mát 1.58 trên tập huấn luyện ở epoch đầu tiên. Qua các epoch tiếp theo, mô hình đã cải thiện đáng kể, với độ chính xác cuối cùng đạt 97.05% và mất mát giảm xuống còn 0.0693. Điều này cho thấy mô hình đã học tốt các đặc trưng của dữ liệu huấn luyện và đạt được hiệu suất cao trong việc phân loại.
- Hiệu suất Xác thực:** Mô hình cũng cho thấy khả năng tổng quát tốt khi đánh giá trên tập xác thực. Độ chính xác trên tập xác thực bắt đầu từ 55.5% và tăng lên 96.02% ở epoch cuối cùng. Mất mát trên tập xác thực giảm từ 1.1260 xuống 0.14, chứng tỏ mô hình không chỉ học tốt từ dữ liệu huấn luyện mà còn duy trì hiệu suất cao trên dữ liệu chưa thấy.

Để đánh giá công bằng và chính xác hơn hiệu quả của các mô hình trên dữ liệu mất cân bằng ta có thể cân nhắc tới một số đánh giá thay thế như precision, recall, f1-score, ...

		Actual		
		Positive	Negative	
Predicted	Positive	True Positive (TP)	False Positive (FP) Type I error	Precision = $TP/(TP+FP)$
	Negative	False Negative (FN) Type II error	True Negative (TN)	
		Recall = $TP/(TP+FN)$	False positive rate (FPR) = $FP/(FP+TN)$	

Hình 10. Confusion matrix

Trong đó:

Actual là lớp của dữ liệu từ tập test và Predicted class là lớp mà mô hình dự đoán cho dữ liệu từ tập test.

- Accuracy: Được dùng để đo độ chính xác của mô hình phân lớp

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

- **Precision:** Là tỉ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm mô hình dự đoán là Positive. Hay nói cách khác, đây là giá trị thể hiện khả năng phát hiện tất cả các Positive. Precision càng cao, tức là số điểm mô hình dự đoán là positive đều là positive càng nhiều. Precision = 1, tức là tất cả số điểm mô hình dự đoán là Positive đều đúng, hay không có điểm nào có nhãn là Negative mà mô hình dự đoán nhầm là Positive.

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

- **Recall:** Là tỉ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm thật sự là Positive (hay tổng số điểm được gán nhãn là Positive ban đầu). Nói cách khác giá trị này thể hiện sự chuẩn xác của việc phát triển các điểm Positive. Recall càng cao, tức là số điểm là positive bị bỏ sót càng ít. Recall = 1, tức là tất cả số điểm có nhãn là Positive đều được mô hình nhận ra.

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

- **F1 - score:** Là một giá trị trung bình điều hòa (harmonic mean) của các tiêu chí Precision và Recall. F1 có giá trị lớn nếu cả 2 giá trị Precision và Recall đều lớn. Hàm F1 có giá trị càng cao chứng minh mô hình phân lớp càng tốt. Khi lý tưởng nhất là $F1 = 1$ (Khi đó $Recall - Precision = 1$).

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

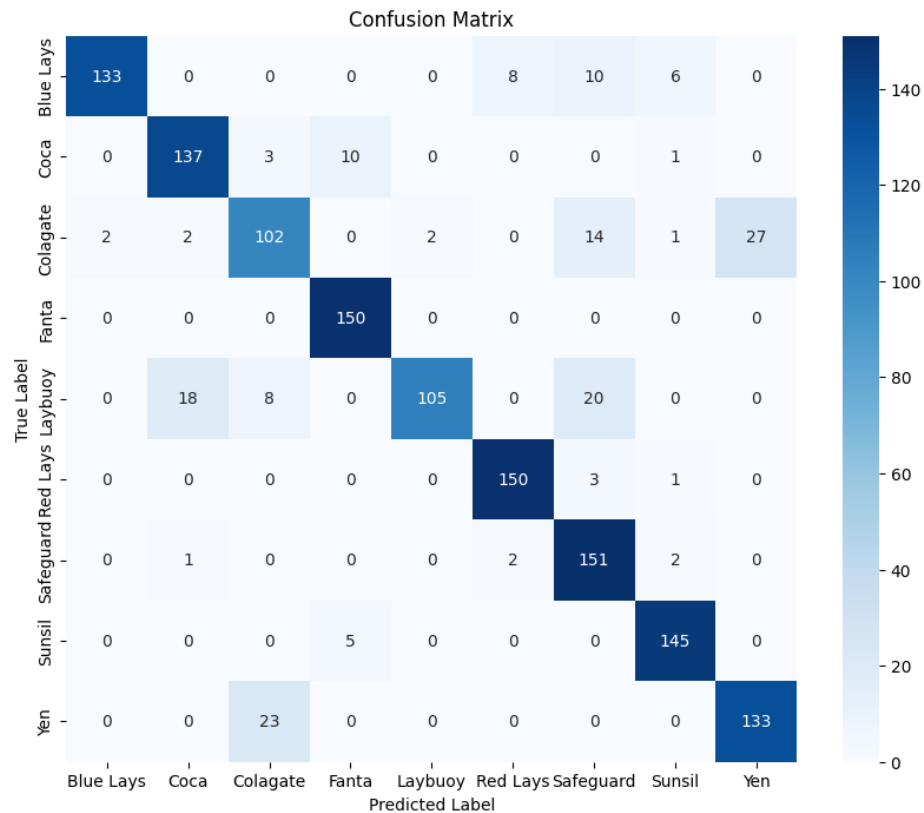
Ở đây chúng tôi tiến hành kiểm tra mô hình trên tập dữ liệu kiểm tra để đánh giá hiệu suất tôi tạo ma trận nhầm để trực quan hóa các dự đoán sai của mô hình trên dữ liệu mà nó chưa nhìn thấy.

Bảng 4. Bảng kết quả khi dự đoán trên dữ liệu thực tế

Class	Precision	Recall	F1-Score
Blue Lays	0.99	0.85	0.91
Coca	0.87	0.91	0.89
Colagate	0.75	0.68	0.71
Fanta	0.91	1.00	0.95
Lifebuoy	0.98	0.70	0.81
Red Lays	0.94	0.97	0.96
Safeguard	0.76	0.97	0.85
Sunsil	0.93	0.97	0.95
Yen	0.83	0.85	0.84
Accuracy	-	-	0.88
Macro Avg	0.88	0.88	0.88
Weighted Avg	0.88	0.88	0.88

Accuracy, Precision và Recall là những thước đo quan trọng trong việc đánh giá mô hình phân loại. Tuy nhiên, có sự khác biệt rõ ràng giữa các chỉ số này, và mỗi chỉ số

mang đến một góc nhìn riêng về hiệu suất của mô hình. Chúng tôi thể hiện nó bằng dấu (-)



Hình 11: Kết quả mô hình trên dữ liệu thực tế

Dựa trên kết quả phân loại được cung cấp, có thể thấy mô hình Vision Transformer đã thể hiện hiệu suất đáng kể trong bài toán phân loại sản phẩm tiêu dùng này. Với độ chính xác tổng thể đạt 88%, mô hình cho thấy khả năng phân biệt tốt giữa 9 lớp sản phẩm khác nhau. Các chỉ số macro average và weighted average đồng nhất ở mức 0.88 cho precision, recall và f1-score, phản ánh sự cân bằng tương đối tốt trong hiệu suất của mô hình trên các lớp. Điều này chứng tỏ ViT có khả năng học và trích xuất các đặc trưng quan trọng từ hình ảnh sản phẩm, bất kể sự đa dạng trong thiết kế và đặc điểm của chúng. Đánh giá tổng quát các kết quả trên cho thấy mô hình Vision Transformer có khả năng học tốt và tổng quát mạnh mẽ, với độ chính xác và sự giảm mất mát ổn định qua thời gian. Mô hình đã chứng minh được khả năng phân loại hình ảnh hiệu quả, với hiệu suất cao không chỉ trên tập huấn luyện mà còn trên tập xác thực. Điều này cho thấy mô hình có thể áp dụng thành công vào các bài toán phân loại hình ảnh thực tế.

Bên cạnh đó chúng tôi có thực hiện thực nghiệm thêm khi sử dụng bộ dataset này và thực hiện train trên mô hình CNN để thực nghiệm với các tham số:

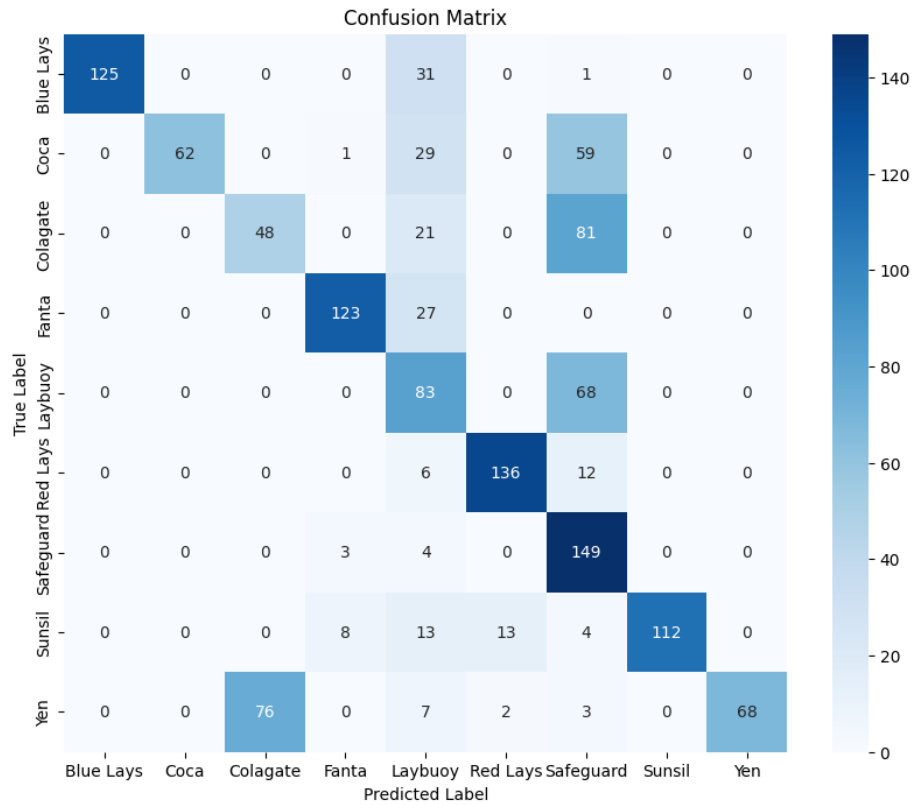
Bảng 5. Tham số thực nghiệm mô hình CNN

Param	Giá trị
Input Size (Kích thước ảnh đầu vào)	64x64x3
Convolutional Layers (Số lớp tích chập)	4
Fully Connected Layers (Lớp kết nối đầy đủ)	[1024, 512]

Sau khi được huấn luyện trên bộ dataset trên với 31 epoch chúng tôi nhận được kết quả

Bảng 6. Bảng kết quả khi dự đoán trên dữ liệu thực tế với mô hình CNN

Class	Precision	Recall	F1-Score
Blue Lays	1.00	0.80	0.89
Coca	1.00	0.41	0.58
Colagate	0.39	0.32	0.35
Fanta	0.91	0.82	0.86
Lifebuoy	0.38	0.55	0.45
Red Lays	0.90	0.88	0.89
Safeguard	0.40	0.96	0.56
Sunsil	1.00	0.75	0.85
Yen	1.00	0.44	0.61
Accuracy	-	-	0.66
Macro Avg	0.78	0.66	0.67
Weighted Avg	0.78	0.66	0.67



Hình 11. Biểu đồ nhiệt mô hình CNN

Dựa trên kết quả từ hai bảng so sánh về mô hình CNN và ViT ta có thể tạo ra một bảng so sánh để làm rõ sự khác biệt và ưu nhược điểm giữa chúng. Dưới đây là cách so sánh:

Bảng 7. So sánh kết quả thực nghiệm ViT và CNN

	CNN	ViT
Accuracy	0.66	0.88
Precision	0.77	0.88
Recall	0.66	0.88
F1-Score	0.67	0.88

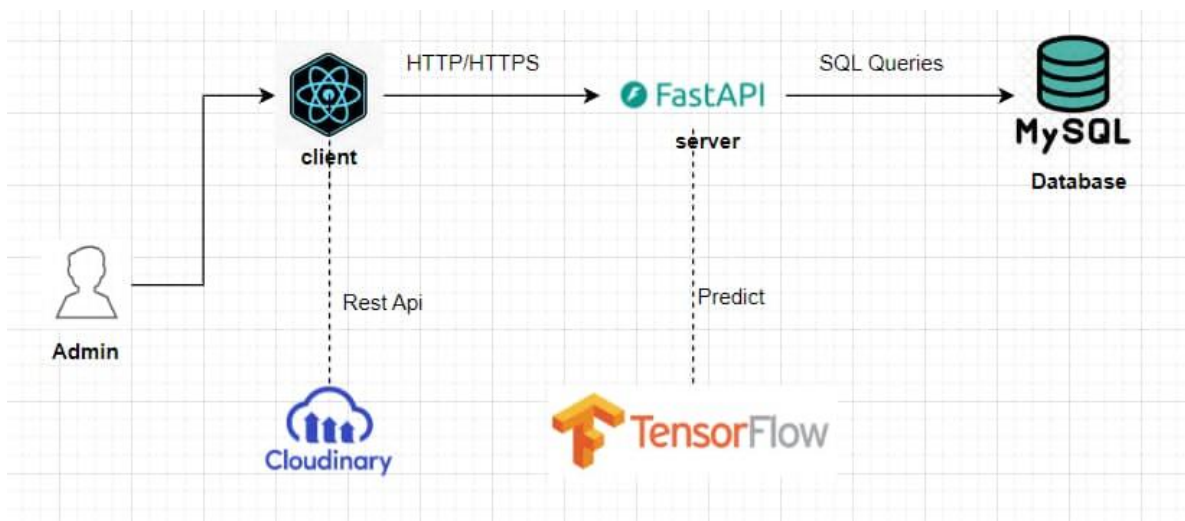
Mô hình ViT cho thấy hiệu suất vượt trội hơn so với mô hình CNN trên tất cả các chỉ số đánh giá. ViT đạt được độ chính xác, precision, recall và F1-score đều là 0.88, trong khi CNN chỉ đạt được các chỉ số tương ứng là 0.66, 0.77, 0.66 và 0.67. Đặc biệt, ViT có độ chính xác và recall cao hơn CNN 22%, đồng thời F1-score của ViT cũng cao hơn 21% so với CNN. Điều này cho thấy ViT có khả năng nhận dạng

và phân loại sản phẩm chính xác hơn, đồng thời cũng ít bỏ sót các mẫu cần phát hiện. Từ kết quả này, có thể kết luận rằng mô hình ViT là lựa chọn phù hợp hơn cho nhiệm vụ nhận dạng sản phẩm của cửa hàng tiện lợi, do nó mang lại hiệu suất tổng thể tốt hơn đáng kể so với mô hình CNN.

CHƯƠNG 4. NGHIÊN CỨU VÀ PHÁT TRIỂN

4.1. Mô tả yêu cầu và chức năng của hệ thống

4.1.1. Sơ đồ hệ thống



Hình 12. Sơ đồ Hệ thống

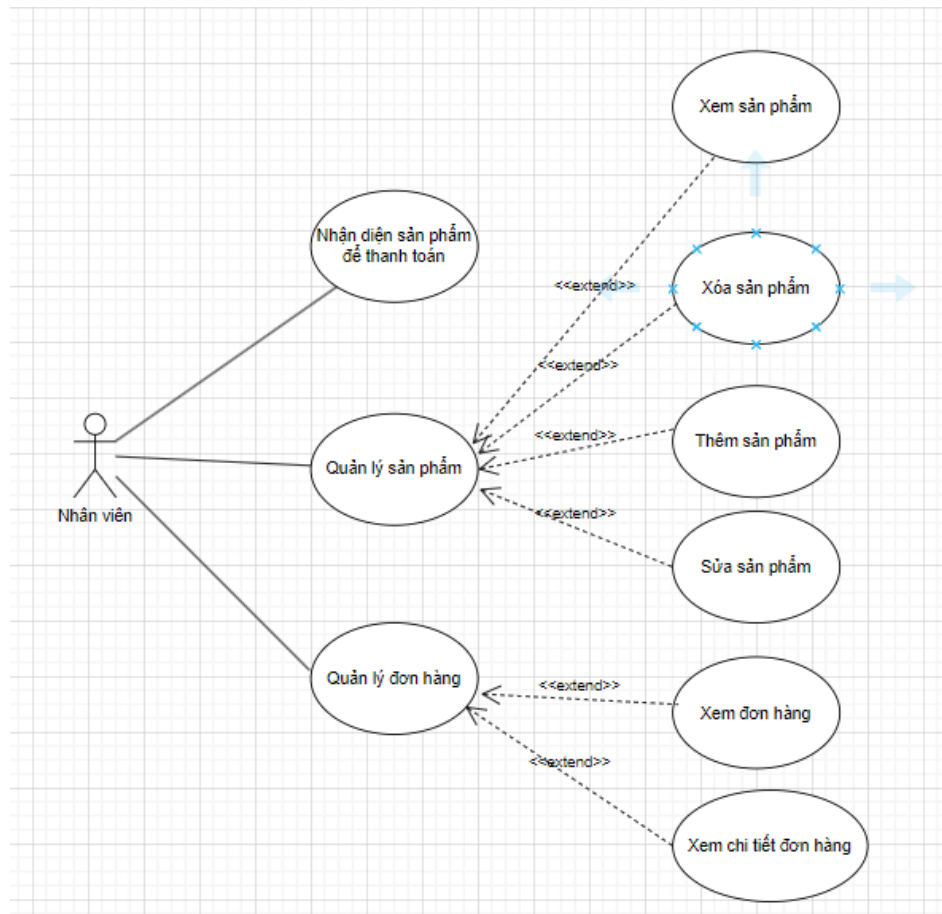
- Mô tả:

- + Admin: Người quản lý hệ thống.
- + Client: Ứng dụng phía người dùng, sử dụng HTTP/HTTPS để giao tiếp với hệ thống.
- + Cloudinary: Dịch vụ lưu trữ và xử lý hình ảnh.
- + Tensorflow: Thư viện học máy được sử dụng để dự đoán/nhận diện sản phẩm.
- + MySQL Database: Cơ sở dữ liệu lưu trữ thông tin sản phẩm và giao dịch.
- + FastAPI: Cung cấp API để tương tác với hệ thống.

- Quy trình hoạt động:

- + B1: Client gửi yêu cầu lên hệ thống thông qua FastAPI.
- + B2: Hệ thống sử dụng ViT được xây dựng dựa trên Tensorflow để dự đoán/nhận diện sản phẩm từ hình ảnh.
- + B3: Thông tin sản phẩm và giao dịch được lưu trữ trong cơ sở dữ liệu MySQL.

4.1.2. Lược đồ Use Case



Hình 13. Lược đồ UseCase

- Mô tả chi tiết từng use case:

+ Nhận diện sản phẩm để thanh toán: Chức năng này là cốt lõi của hệ thống. Cho phép nhân viên có thể tiếp nhận sản phẩm của khách hàng. Từ đó tiến hành chụp hình ảnh của sản phẩm để lấy thông tin sản phẩm và phục vụ cho quá trình thanh toán.

+ Quản lý sản phẩm: Chức năng cho phép nhân viên thực hiện các tác vụ như xem, thêm, sửa, xóa thông tin sản phẩm.

+ Quản lý đơn hàng: Chức năng cho phép nhân viên theo dõi, cập nhật trạng thái của các đơn hàng.

+ Xem sản phẩm: Chức năng cho phép nhân viên xem thông tin chi tiết của các sản phẩm.

+ Thêm sản phẩm: Chức năng cho phép nhân viên thêm mới sản phẩm vào hệ thống.

+ Sửa sản phẩm: Chức năng cho phép nhân viên chỉnh sửa thông tin của các sản phẩm.

+ Xóa sản phẩm: Chức năng cho phép nhân viên xóa bỏ các sản phẩm khỏi hệ thống.

+ Xem đơn hàng: Chức năng cho phép nhân viên xem thông tin chi tiết của các đơn hàng.

+ Xem chi tiết đơn hàng: Chức năng cho phép nhân viên xem các thông tin chi tiết về một đơn hàng cụ thể.

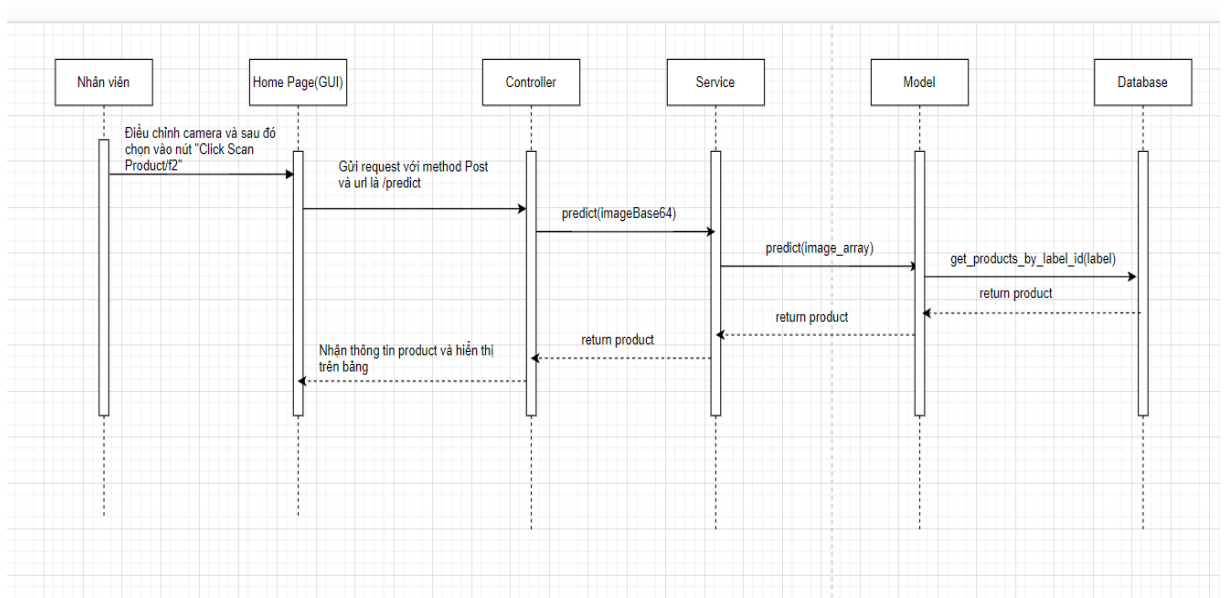
- Giải thích mối quan hệ giữa các use case:

+ Các chức năng như Xem sản phẩm, Thêm sản phẩm, Sửa sản phẩm, Xóa sản phẩm được "extend" từ Quản lý sản phẩm.

+ Các chức năng như Xem đơn hàng, Xem chi tiết đơn hàng được "extend" từ Quản lý đơn hàng.

4.1.3. Lược đồ tuần tự (Sequence Diagram)

- Chức năng nhận diện sản phẩm

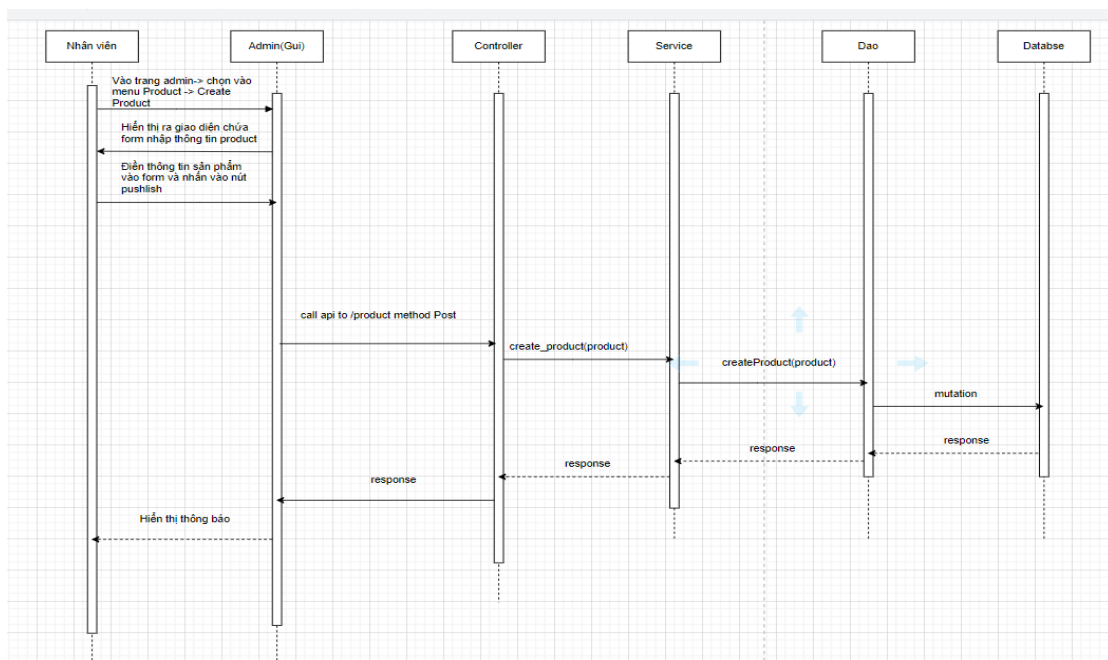


Hình 14. Lược đồ tuần tự chức năng nhận diện sản phẩm

● Mô tả

- **Bước 1:** Người dùng (Home Page/GUI) gửi yêu cầu nhận diện sản phẩm kèm theo hình ảnh sản phẩm đến ứng dụng.
- **Bước 2:** Ứng dụng tiếp nhận yêu cầu và hình ảnh, sau đó gọi hàm `predict_image()` của mô-đun Service để thực hiện nhận diện sản phẩm.
- **Bước 3:** Mô-đun Service sẽ sử dụng mô hình học máy đã được huấn luyện trước đó để dự đoán loại sản phẩm dựa trên hình ảnh đầu vào.
- **Bước 4:** Kết quả dự đoán sẽ được trả về ứng dụng.
- **Bước 5:** Ứng dụng nhận kết quả từ mô-đun Service, sau đó gọi hàm `get_product_by_label()` của mô-đun Database để lấy thông tin chi tiết về sản phẩm.
- **Bước 6:** Mô-đun Database trả về thông tin sản phẩm tương ứng với kết quả dự đoán.
- **Bước 7:** Ứng dụng nhận thông tin sản phẩm từ mô-đun Database và trả về cho Người dùng (Home Page/GUI).

- Chức năng thêm sản phẩm(Quản lý sản phẩm)

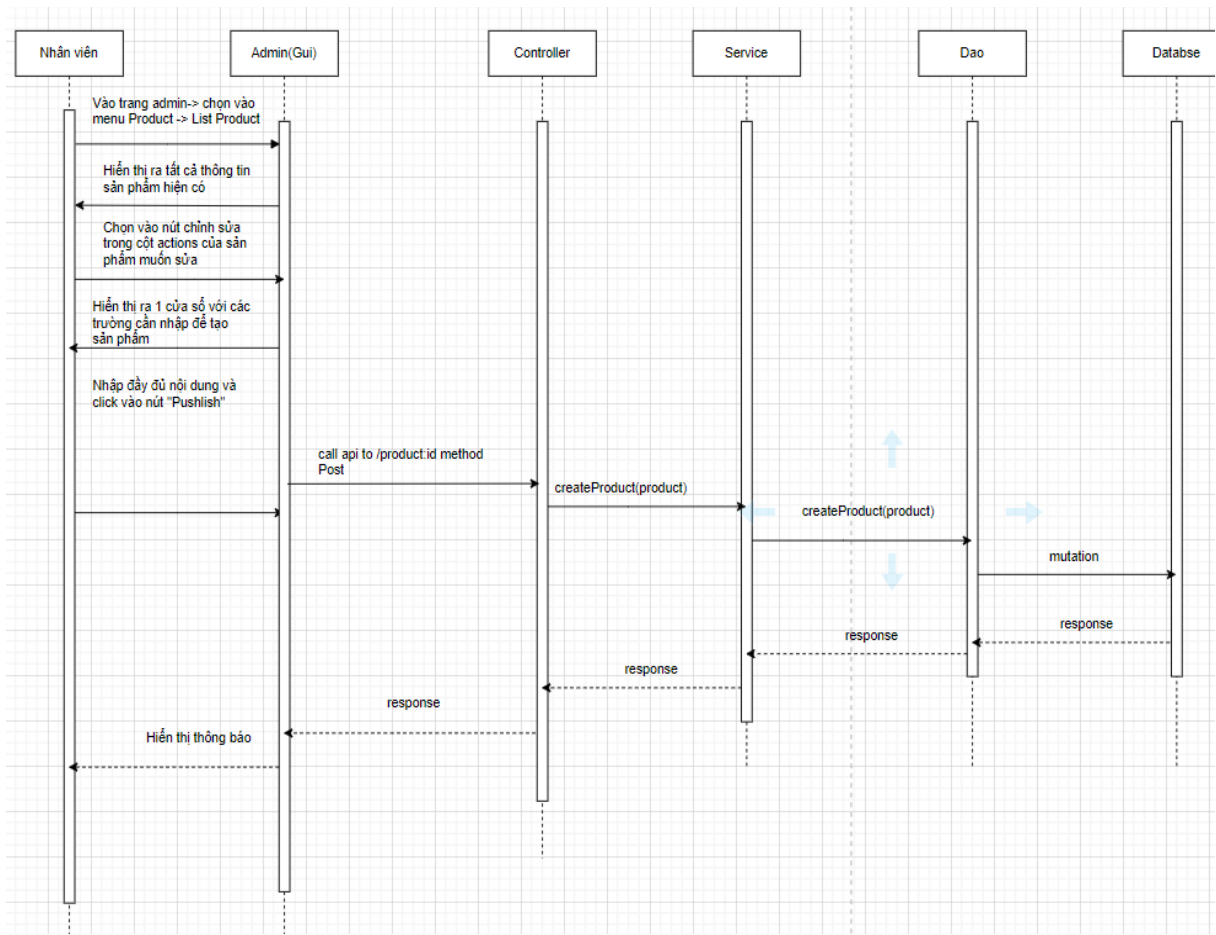


Hình 15. Lược đồ tuần tự chức năng thêm sản phẩm

● Mô tả

- **Bước 1:** Người dùng (Home Page/GUI) gửi yêu cầu thêm sản phẩm cùng với thông tin sản phẩm đến ứng dụng (Controller).
- **Bước 2:** Controller tiếp nhận yêu cầu và thông tin sản phẩm, sau đó gọi hàm `create_product()` của mô-đun Service để tạo mới sản phẩm.
- **Bước 3:** Mô-đun Service thực hiện các xử lý cần thiết để tạo mới sản phẩm, chẳng hạn như lưu thông tin sản phẩm vào cơ sở dữ liệu.
- **Bước 4:** Kết quả tạo sản phẩm sẽ được trả về Controller.
- **Bước 5:** Controller nhận kết quả từ mô-đun Service và trả về cho Người dùng (Home Page/GUI).

- Chức năng sửa sản phẩm(Quản lý sản phẩm)

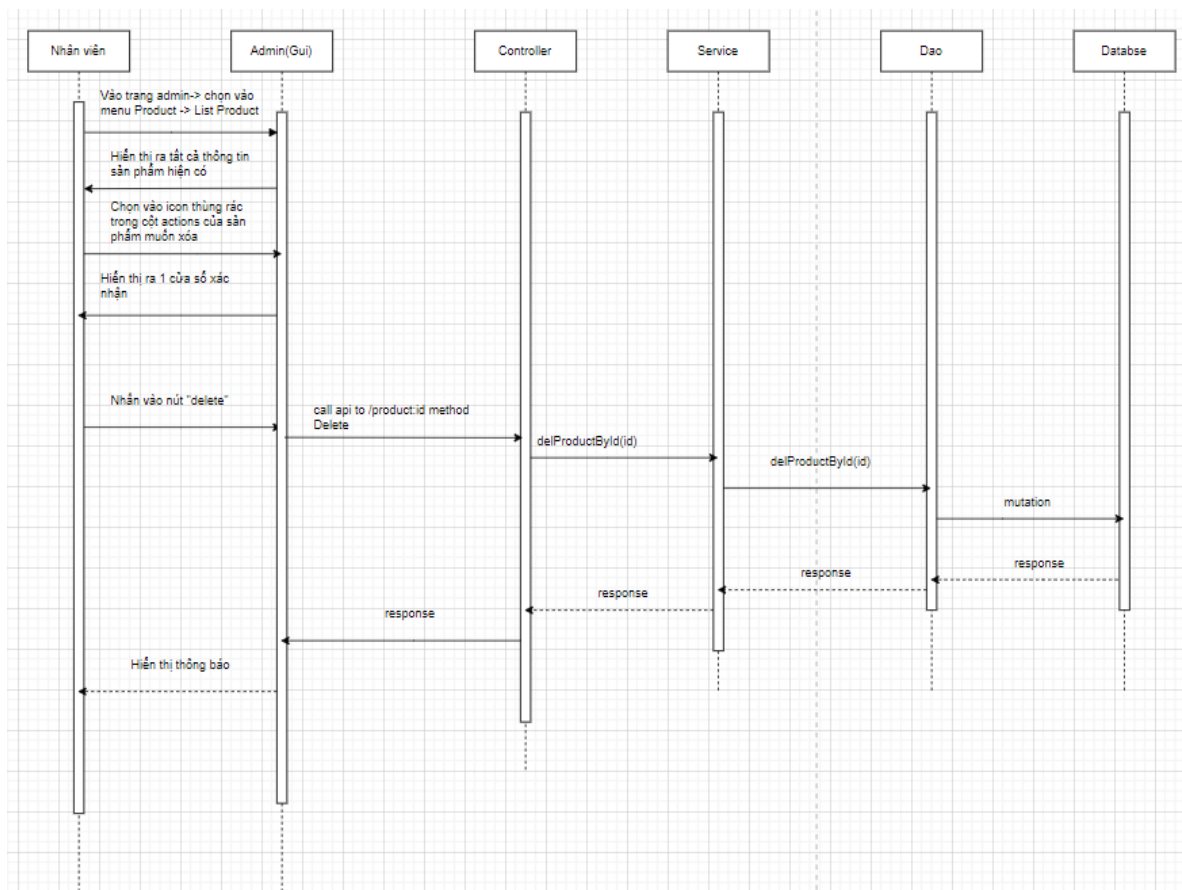


Hình 16. Lược đồ tuần tự chức năng sửa sản phẩm

● Mô tả

- **Bước 1:** Người dùng (Home Page/GUI) gửi yêu cầu sửa đổi thông tin sản phẩm đến ứng dụng (Controller).
- **Bước 2:** Controller tiếp nhận yêu cầu và thông tin sản phẩm cần sửa đổi, sau đó gọi hàm `update_product()` của mô-đun Service để cập nhật thông tin sản phẩm.
- **Bước 3:** Mô-đun Service sẽ thực hiện các xử lý cần thiết để cập nhật thông tin sản phẩm, chẳng hạn như lưu thông tin mới vào cơ sở dữ liệu.
- **Bước 4:** Kết quả cập nhật sản phẩm sẽ được trả về Controller.
- **Bước 5:** Controller nhận kết quả từ mô-đun Service và trả về cho Người dùng (Home Page/GUI).

- Chức năng xóa sản phẩm(Quản lý sản phẩm)

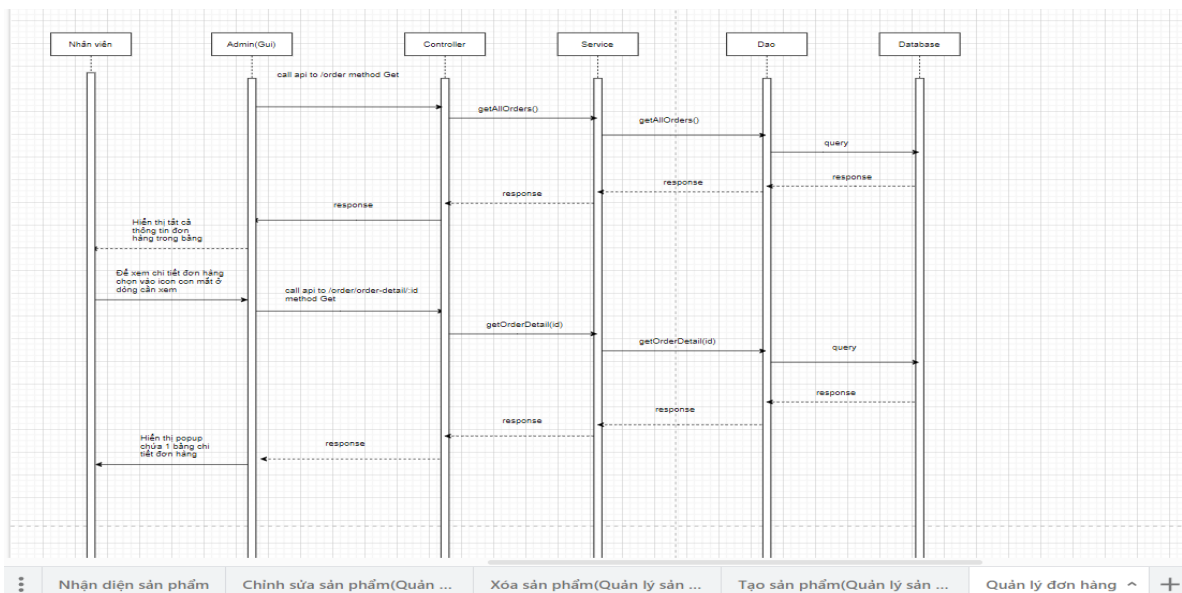


Hình 17. Lược đồ tuần tự chức năng xóa sản phẩm

● Mô tả

- **Bước 1:** Người dùng (Home Page/GUI) gửi yêu cầu xóa sản phẩm đến ứng dụng (Controller).
- **Bước 2:** Controller tiếp nhận yêu cầu xóa sản phẩm, sau đó gọi hàm `delete_product()` của mô-đun Service để xóa sản phẩm.
- **Bước 3:** Mô-đun Service sẽ thực hiện các xử lý cần thiết để xóa sản phẩm, chẳng hạn như xóa thông tin sản phẩm khỏi cơ sở dữ liệu.
- **Bước 4:** Kết quả xóa sản phẩm sẽ được trả về Controller.
- **Bước 5:** Controller nhận kết quả từ mô-đun Service và trả về cho Người dùng (Home Page/GUI).

- Chức năng xem đơn hàng(Quản lý đơn hàng)



Hình 18. Lược đồ tuần tự chức năng quản lý đơn hàng

● Mô tả

- **Bước 1:** Người dùng (Home Page/GUI) gửi yêu cầu xem đơn hàng (bao gồm cả xem chi tiết đơn hàng) đến ứng dụng (Controller).

- **Bước 2:** Controller tiếp nhận yêu cầu và kiểm tra tính hợp lệ của dữ liệu đầu vào (ví dụ: kiểm tra xem người dùng có quyền truy cập vào chức năng này hay không).
- **Bước 3:** Nếu dữ liệu đầu vào hợp lệ, Controller sẽ gọi hàm `get_order_info()` của mô-đun Service để lấy thông tin về các đơn hàng, bao gồm cả chi tiết của từng đơn hàng.
- **Bước 4:** Mô-đun Service sẽ thực hiện các xử lý cần thiết để lấy thông tin về các đơn hàng và chi tiết đơn hàng từ cơ sở dữ liệu.
- **Bước 5:** Kết quả lấy thông tin đơn hàng và chi tiết đơn hàng sẽ được trả về Controller.
- **Bước 6:** Controller nhận kết quả từ mô-đun Service, có thể thực hiện các xử lý bổ sung (ví dụ: format lại dữ liệu) và trả về cho Người dùng (Home Page/GUI) để hiển thị.
- **Bước 7:** Người dùng (Home Page/GUI) nhận và hiển thị thông tin đơn hàng, bao gồm cả chi tiết đơn hàng.

4.1.4. Công nghệ sử dụng

- Frontend

- React
 - Mô tả: React là một thư viện JavaScript dùng để xây dựng giao diện người dùng. Nó được phát triển bởi Facebook và hiện nay được sử dụng rộng rãi trong cộng đồng phát triển web.
 - Lý do sử dụng: React cho phép xây dựng các thành phần giao diện tái sử dụng, giúp việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn. Nó cung cấp cơ chế ảo hóa DOM, giúp tối ưu hóa hiệu suất ứng dụng.
- TypeScript

- Mô tả: TypeScript là một ngôn ngữ lập trình phát triển từ JavaScript, bổ sung thêm kiểu dữ liệu tĩnh. Nó được phát triển bởi Microsoft và giúp các lập trình viên viết mã rõ ràng và dễ bảo trì hơn.
- Lý do sử dụng: TypeScript giúp phát hiện lỗi ngay trong quá trình viết code, giảm thiểu lỗi runtime và cải thiện tính minh bạch của mã nguồn. Nó cũng cung cấp tính năng autocompletion và refactoring hiệu quả.

- Backend

● Tensorflow

- Mô tả: tensorflow là một thư viện mã nguồn mở dành cho học máy và học sâu. Nó được phát triển bởi Google và hiện nay là một trong những thư viện học máy phổ biến nhất.
- Lý do sử dụng: tensorflow cung cấp các công cụ mạnh mẽ để xây dựng và triển khai các mô hình học sâu cho việc nhận diện sản phẩm. Nó hỗ trợ nhiều loại mô hình khác nhau và có khả năng mở rộng cao.

● Fastapi

- Mô tả: fastapi là một framework web hiện đại và nhanh chóng cho Python. Nó được thiết kế để dễ dàng xây dựng các API RESTful với hiệu suất cao.
- Lý do sử dụng: fastapi cho phép xây dựng các API hiệu suất cao và dễ dàng, hỗ trợ đầy đủ các tính năng như xác thực, quản lý phiên làm việc và tài liệu tự động. Nó cũng tương thích tốt với các công nghệ async hiện đại.

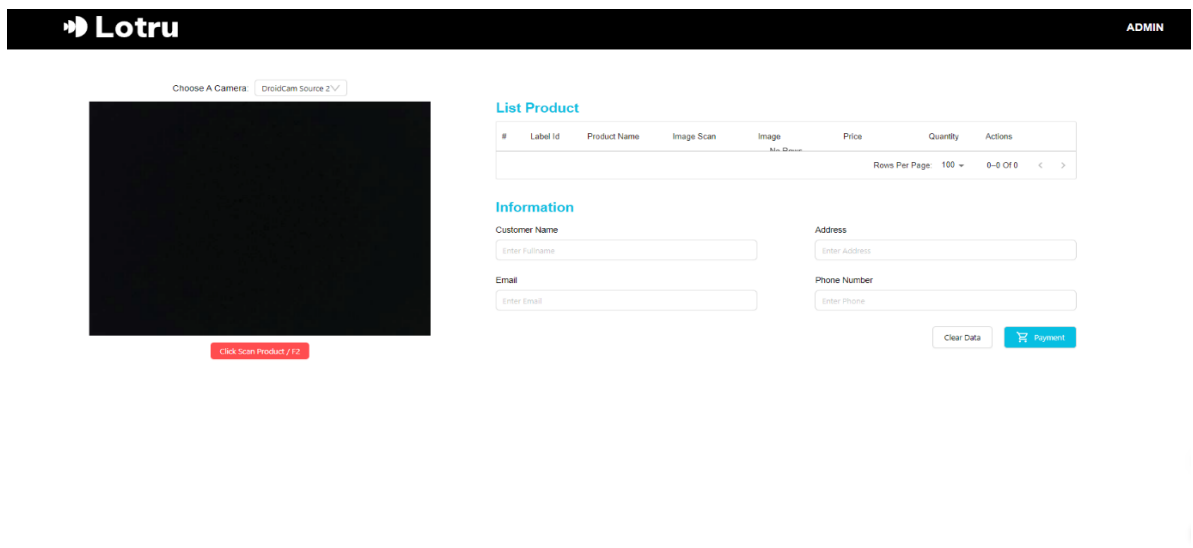
4.4.1. Mô tả các chức năng

Hệ thống bao gồm các chức năng chính như nhận diện sản phẩm, thanh toán, quản lý sản phẩm, và xem các đơn hàng. Các chức năng này được chi tiết như sau:

● Nhận diện sản phẩm

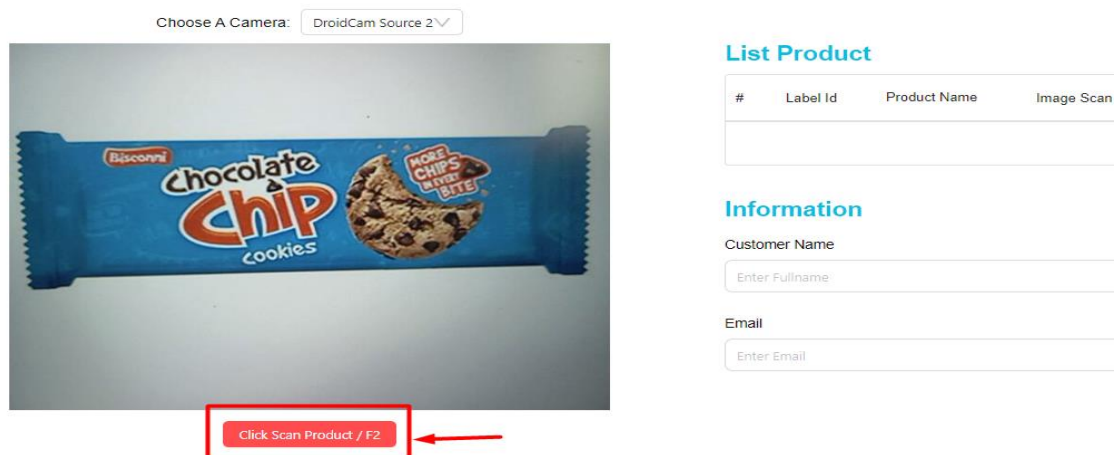
- Mô tả: Chức năng này cho phép hệ thống tự động nhận diện sản phẩm thông qua hình ảnh. Người dùng có thể chụp ảnh sản phẩm hoặc tải lên hình ảnh từ thư viện để hệ thống nhận diện.
- Quy trình:

Bước 1: Chủ cửa hàng truy cập vào website và truy cập vào phần nhận diện sản phẩm ở home page



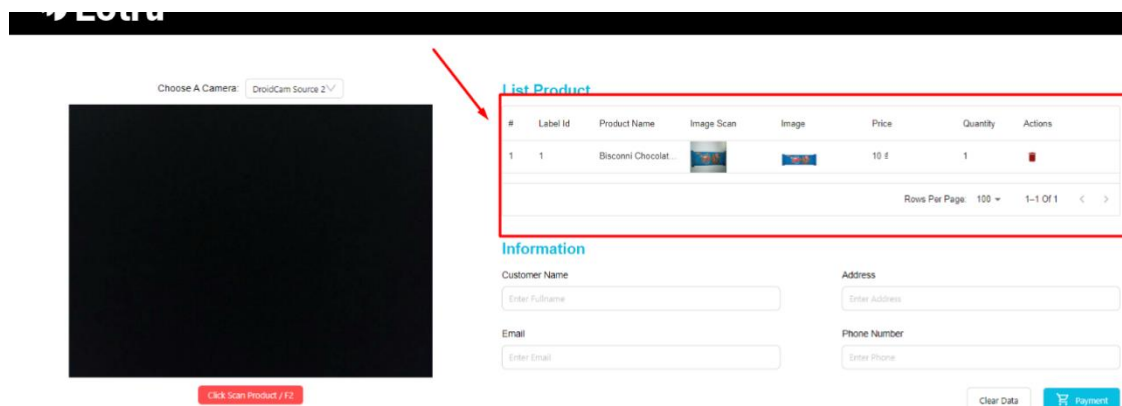
Hình 19. Hình ảnh minh họa chức năng nhận diện sản phẩm(Bước 1)

Bước 2: chụp hình ảnh của sản phẩm.



Hình 20. Hình ảnh minh họa chức năng nhận diện sản phẩm(Bước 2)

Bước 3: Nhấn vào nút click scan product/f2. Hệ thống xử lý và phân tích hình ảnh thông qua mô hình ViT.



Hình 21. Hình ảnh minh họa chức năng nhận diện sản phẩm(Bước 3)

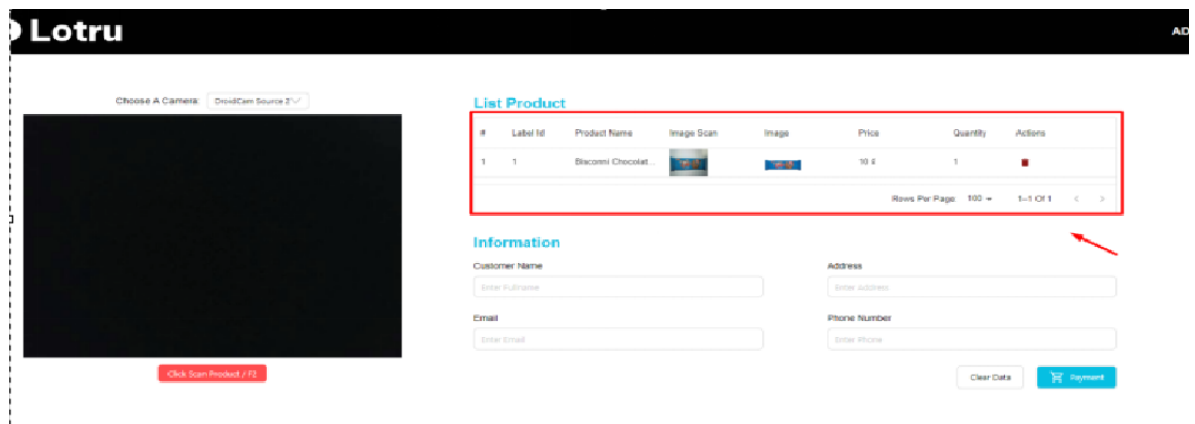
- + Kết quả nhận diện được hiển thị, bao gồm tên sản phẩm, giá cả, và thông tin chi tiết khác. Những thông tin này sẽ được hiển thị bên trong bảng trong trang để dễ dàng tiến hành thanh toán
- + Kỹ thuật: Sử dụng mô hình ViT để xử lý và phân tích hình ảnh. Mô hình này đã được huấn luyện trên một tập dữ liệu lớn của các sản phẩm từ cửa hàng tiện lợi..

● Thanh toán

Mô tả: Chức năng thanh toán cho phép người bán hàng có thể hoàn tất quá trình mua sắm thông qua website. Người dùng có thể xem lại các sản phẩm đã nhận diện và thanh toán

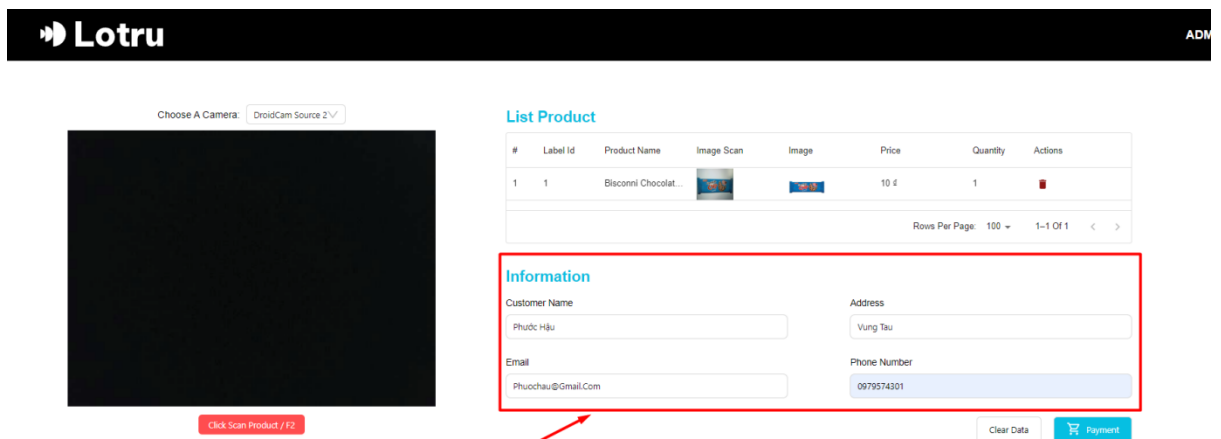
– Quy trình:

Bước 1: Người dùng xem lại các sản phẩm đã nhận diện.



Hình 22. Hình ảnh minh họa chức năng thanh toán(Bước 1)




Bước 2: Điền đầy đủ thông tin của khách hàng vào form



Hình 23. Hình ảnh minh họa chức năng thanh toán(Bước 3)

Bước 3: Nhấn vào Nút Payment bên dưới form để tiến hành thanh toán.

List Product

#	Label Id	Product Name	Image Scan	Image	Price	Quantity	Actions
1	1	Bisconni Chocolat...			10 đ	1	

Rows Per Page: 100 ▾ 1-1 Of 1 < >

Information

Customer Name

Phước Hậu

Address

Vung Tau

Email

Phuchoau@Gmail.Com

Phone Number

0979574301

Clear Data

 Payment

Hình 24. Hình ảnh minh họa chức năng thanh toán(Bước 4)

● Quản lý sản phẩm

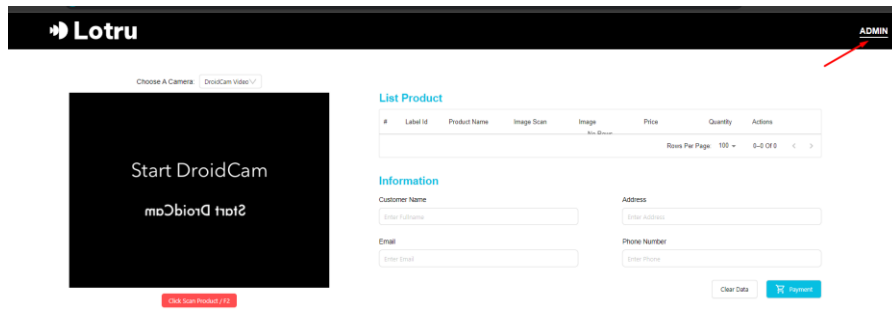
- Mô tả: Chức năng quản lý sản phẩm dành cho quản trị viên của hệ thống, cho phép họ thêm, sửa, hoặc xóa sản phẩm khỏi cơ sở dữ liệu. Dưới đây là mô tả chi tiết về từng phần trong quá trình thêm, sửa, và xóa sản phẩm:

- Thêm mới sản phẩm

- + Mô tả: Chức năng này cho phép quản trị viên thêm mới các sản phẩm vào cơ sở dữ liệu. Quản trị viên cần nhập đầy đủ thông tin sản phẩm và tải lên hình ảnh minh họa.

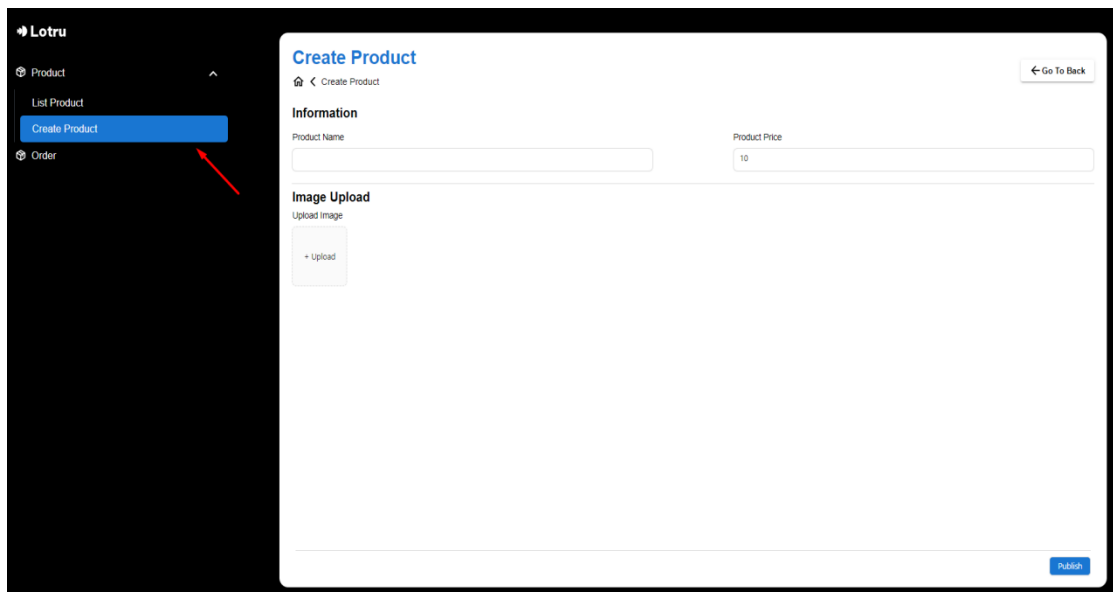
+ Quy trình:

Bước 1: Truy cập vào trang web. Trên thanh menu nhấn vào menu admin.



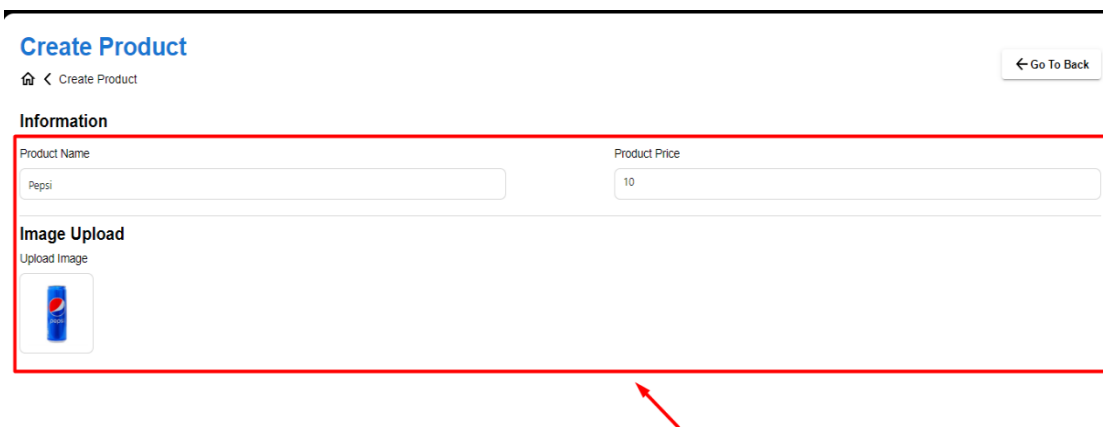
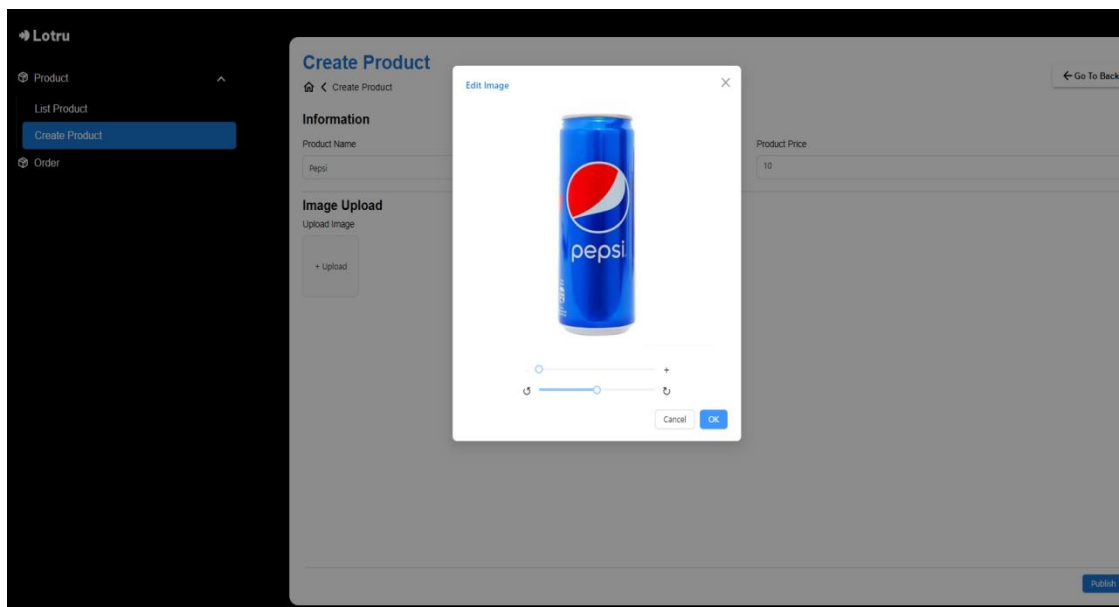
Hình 25. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 1)

Bước 2: Giao diện page admin sẽ hiện ra. Nhấn chọn vào Product bên trong sidebar bên trái -> nhấn vào menu create product trên sidebar



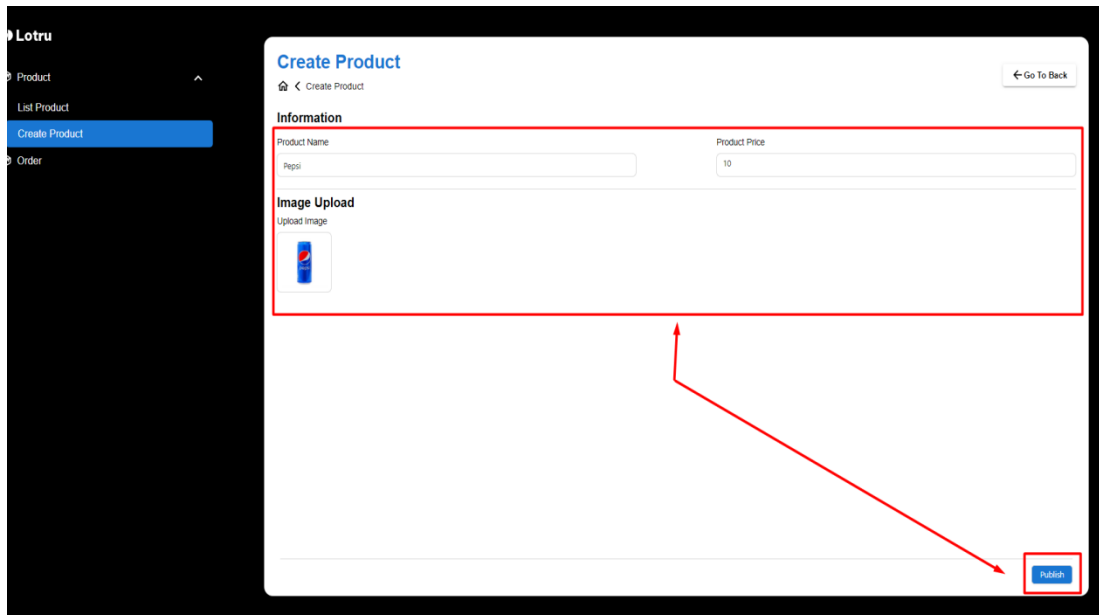
Hình 26. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 2)

Bước 3: Nhập đầy đủ thông tin của product bao gồm tên product, hình ảnh...



Hình 27. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 3)

Bước 4: Nhấn vào nút “Pushlish” để thêm sản phẩm.

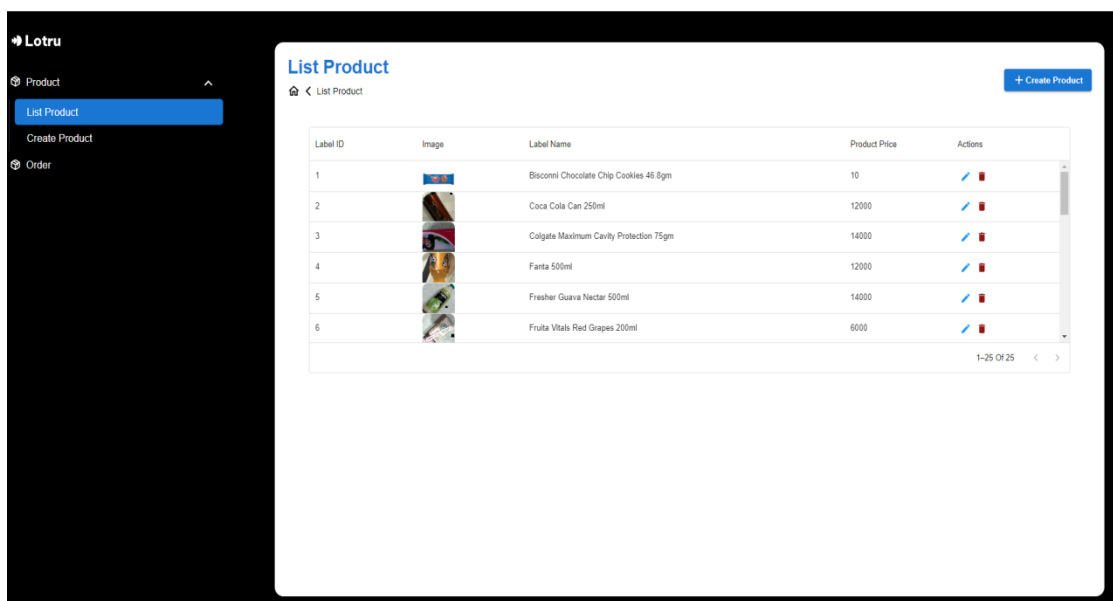


Hình 28. Hình ảnh minh họa chức năng thêm sản phẩm(Bước 4)

– Sửa thông tin sản phẩm

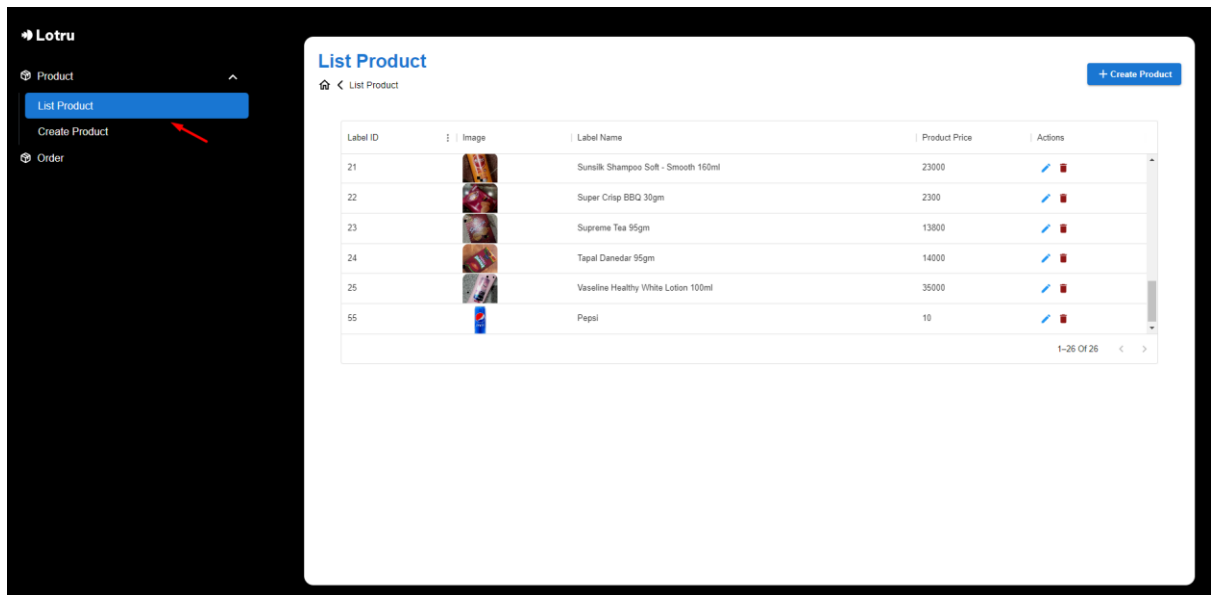
- + Mô tả: Chức năng này cho phép quản trị viên chỉnh sửa thông tin của các sản phẩm đã có trong cơ sở dữ liệu.
- + Quy trình:

Bước 1: Truy cập vào trang web. Trên thanh menu nhấn vào menu admin.




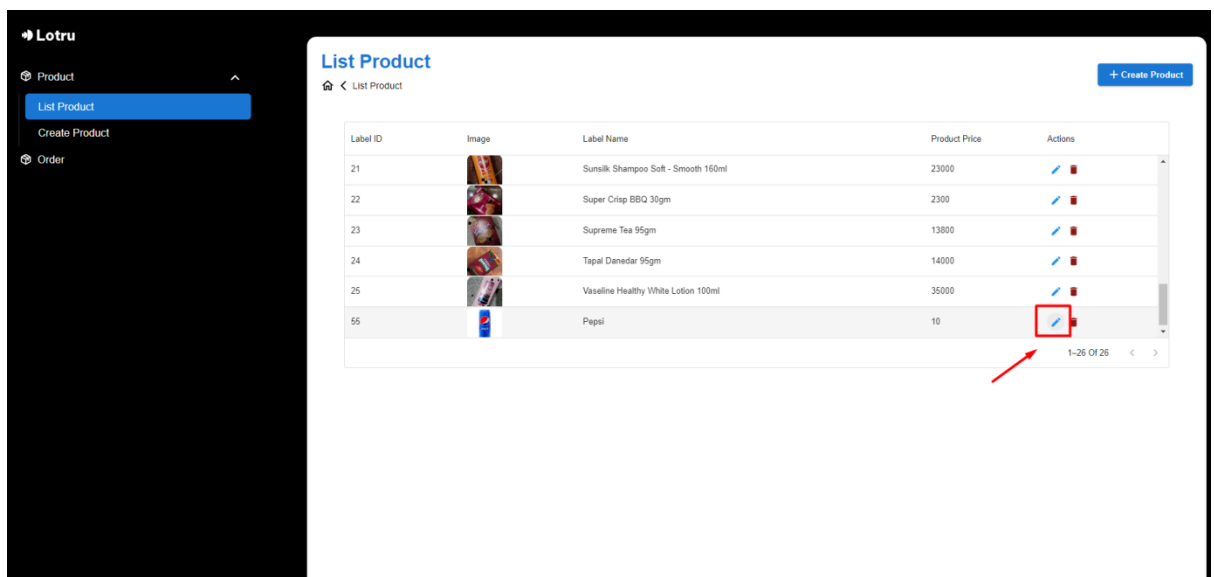
Hình 29. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 1)

Bước 2: Giao diện page admin sẽ hiện ra. Nhấn chọn vào Product bên trong sidebar bên trái -> nhấn vào menu “List Product” trên sidebar



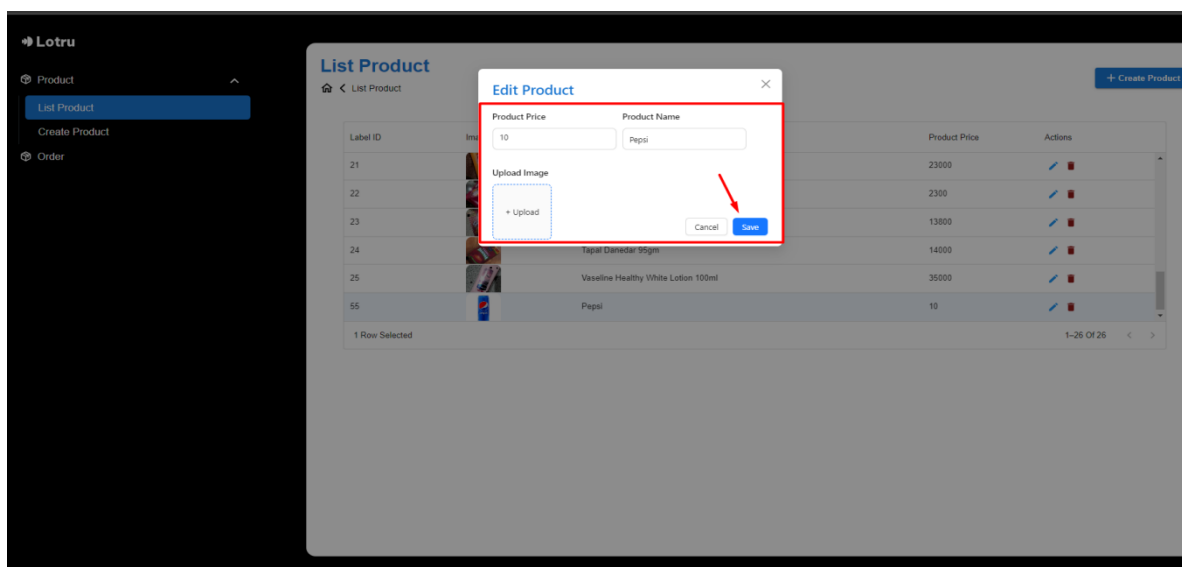
Hình 30. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 2)

Bước 3: Ở đây sẽ hiển thị tất cả những sản phẩm có trong cửa hàng trong bảng. Chọn vào biểu tượng  trên cột “Actions” sản phẩm nào muốn chỉnh sửa.



Hình 31. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 3)

Bước 4: Một cửa sổ sẽ được hiện lên chứa form thông tin của product. Nhập vào các thông tin cần chỉnh sửa và nhấn vào nút “save” để hoàn tất chỉnh sửa.



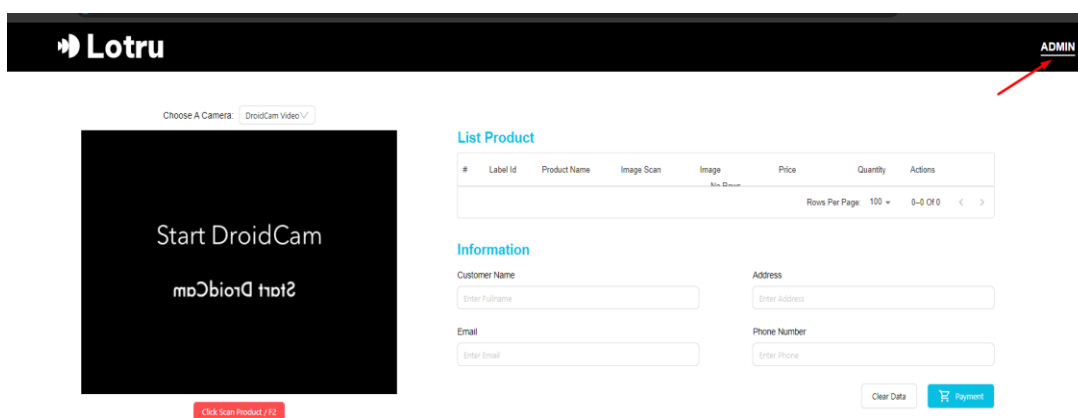
Hình 32. Hình ảnh minh họa chức năng sửa sản phẩm(Bước 4)

- Xóa sản phẩm

+ Mô tả: Chức năng này cho phép quản trị viên xóa các sản phẩm khỏi cơ sở dữ liệu khi không còn cần thiết.

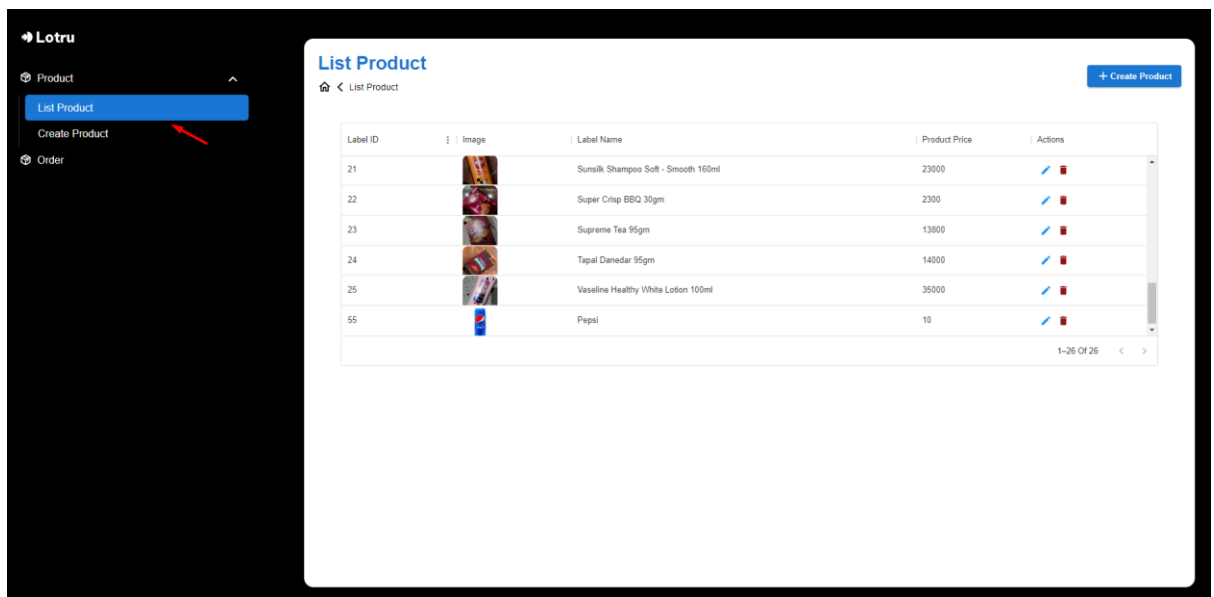
+ Quy trình:

Bước 1: Truy cập vào trang web. Trên thanh menu nhấn vào menu admin.




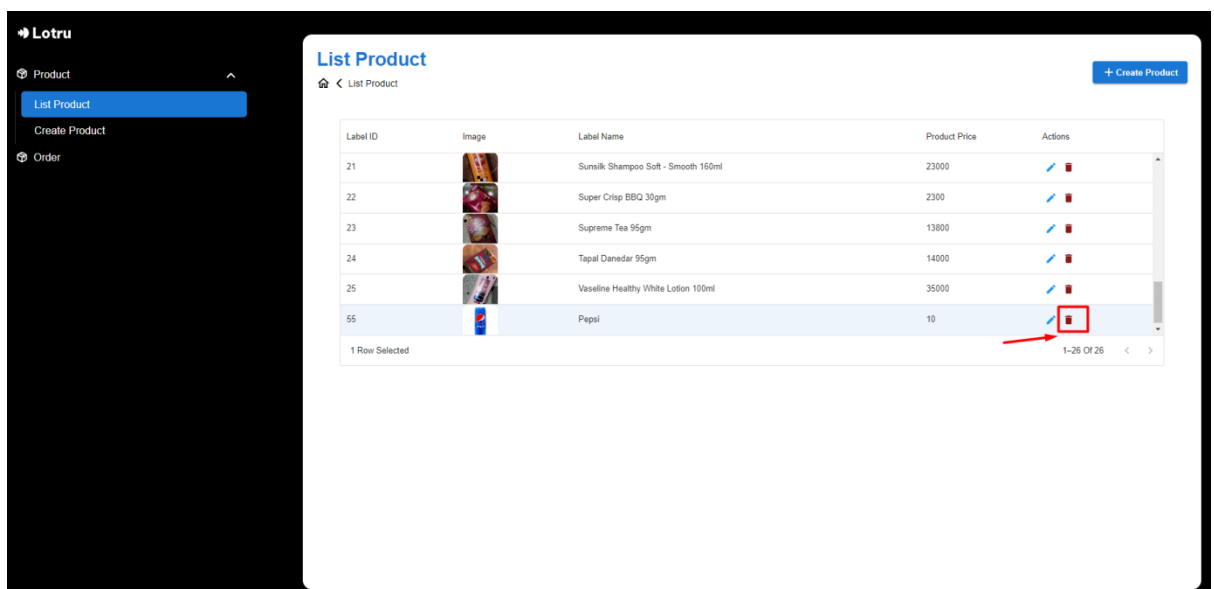
Hình 33. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 1)

Bước 2: Giao diện page admin sẽ hiện ra. Nhấn chọn vào Product bên trong sidebar bên trái -> nhấn vào menu “List Product” trên sidebar



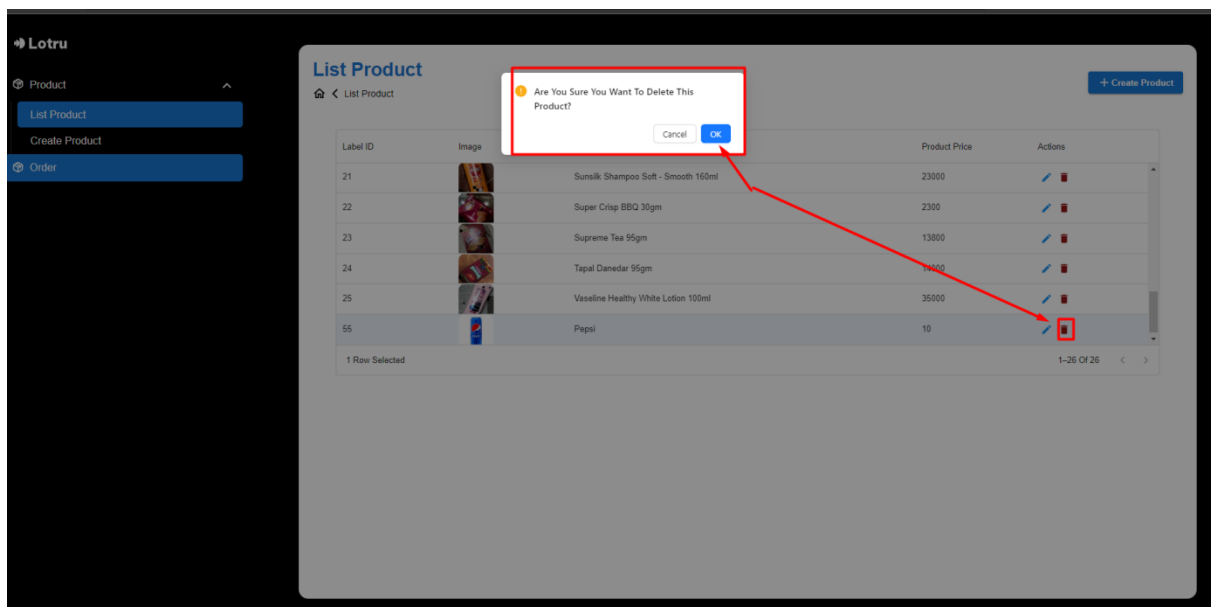
Hình 34. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 2)

Bước 3: Ở đây sẽ hiển thị tất cả những sản phẩm có trong cửa hàng trong bảng. Chọn vào biểu tượng  trên cột “Actions” đối với sản phẩm nào muốn xóa.



Hình 35. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 3)

Bước 4: Một cửa sổ sẽ được hiện lên để xác nhận rằng muốn xóa sản phẩm đó hay không. Nếu đồng ý sản phẩm sẽ được xóa ra khỏi cơ sở dữ liệu.

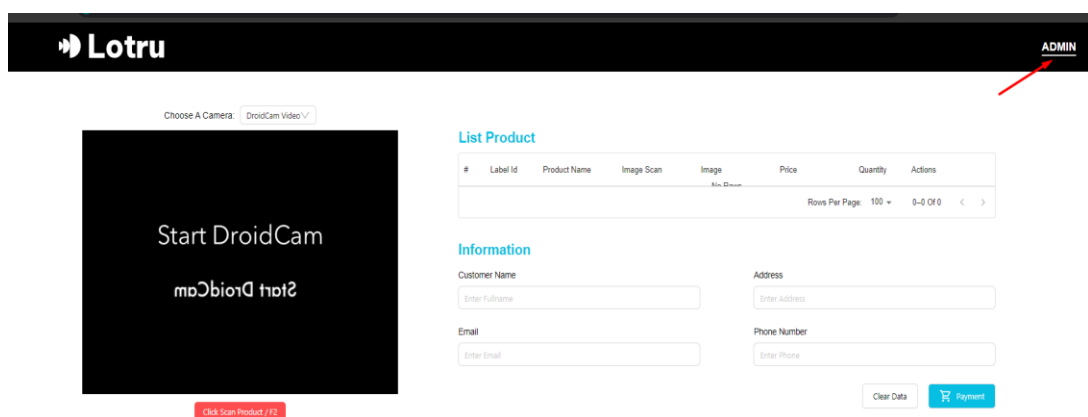


Hình 36. Hình ảnh minh họa chức năng xóa sản phẩm(Bước 4)

● Quản lý đơn hàng

- Mô tả: Chức năng này cho phép quản trị viên xem lại các đơn hàng đã đặt của khách hàng có thể kiểm tra trạng thái đơn hàng, trong khi quản trị viên có thể quản lý và xử lý các đơn hàng.
- Quy trình:

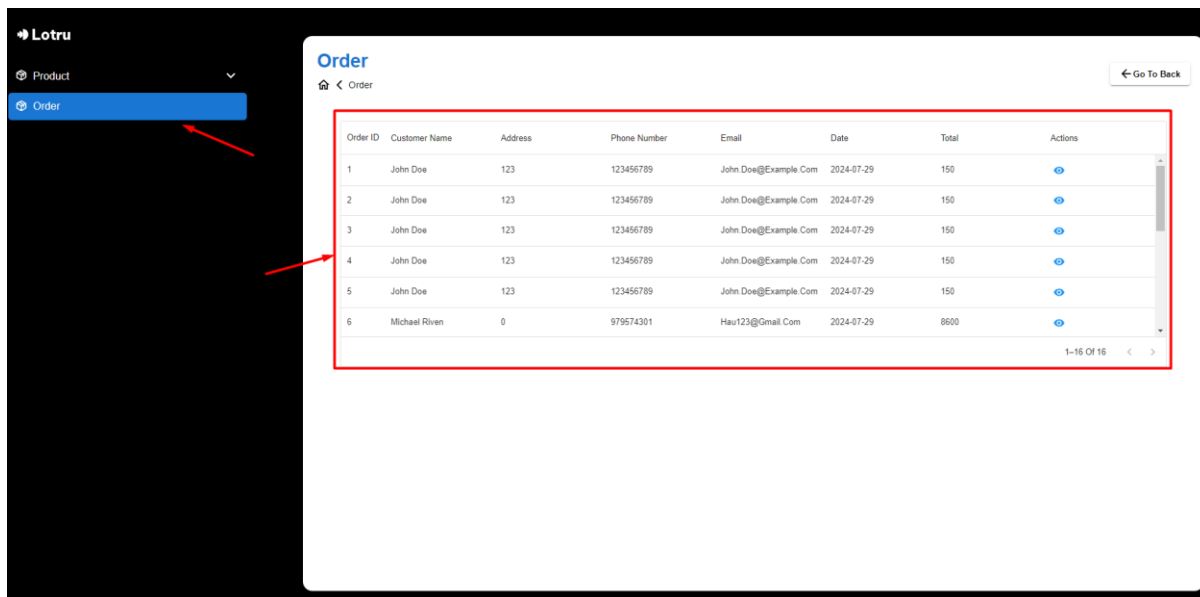
Bước 1: Truy cập vào trang web. Trên thanh menu nhấn vào menu admin.




Hình 37. Hình ảnh minh họa chức năng quản lý đơn hàng(Bước 1)

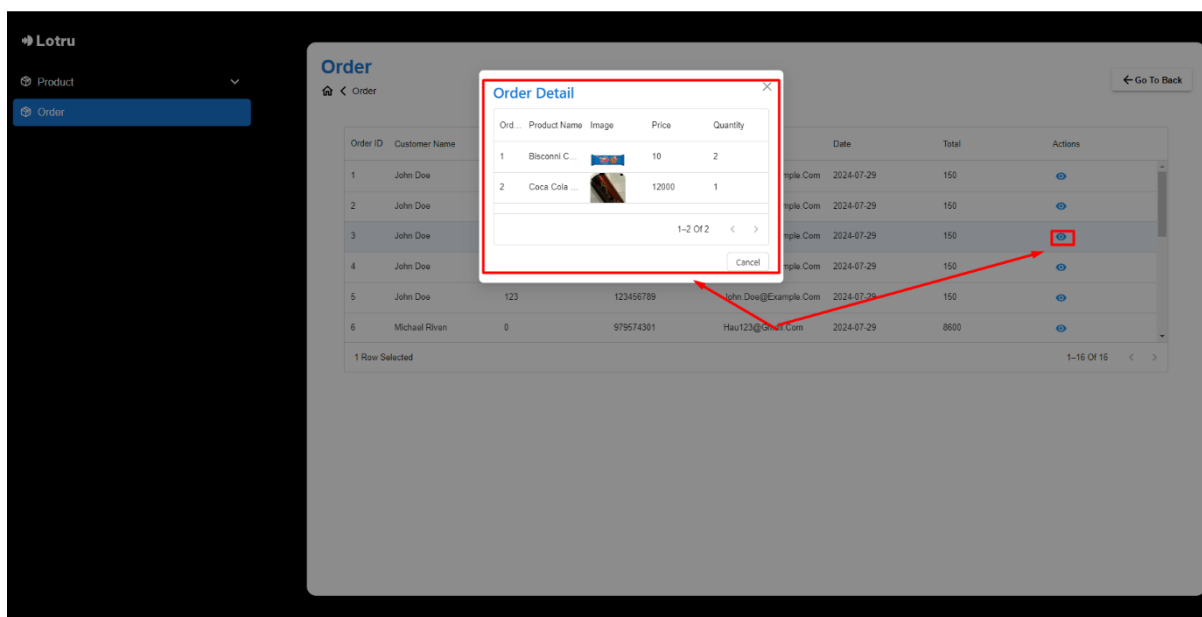
Bước 2: Giao diện page admin sẽ hiện ra. Nhấn chọn vào Order bên trong sidebar

để vào giao diện quản lý đơn hàng



Hình 38. Hình ảnh minh họa chức năng quản lý đơn hàng(Bước 2)

+ Ở đây sẽ hiển thị ra các order đã được đặt trước đó. Để xem chi tiết các đơn hàng có thể nhấn vào con mắt  trên 1 hàng chứa đơn hàng bất kì.



Hình 39. Hình ảnh minh họa chức năng xem đơn hàng chi tiết

CHƯƠNG 5. KIẾN LUẬN VÀ KIẾN NGHỊ

4.1: Kết luận chung

- **Về kiến thức:** Thuật toán ViT và bài toán nhận dạng sản phẩm và thanh toán tự động có ý nghĩa lớn trong việc tối ưu hóa quy trình bán hàng và nâng cao trải nghiệm khách hàng, đặc biệt là trong bối cảnh công nghệ AI ngày càng phát triển và được ứng dụng rộng rãi trong nhiều lĩnh vực, bao gồm cả bán lẻ và thương mại điện tử. Trong quá trình nghiên cứu và phát triển, chúng tôi đã sử dụng thư viện TensorFlow để xây dựng mô hình Vision Transformer, áp dụng nó vào bài toán nhận dạng sản phẩm tại cửa hàng tiện lợi. Chúng tôi cũng đã so sánh mô hình này với một số thuật toán khác như CNN và thử nghiệm trên dữ liệu thực tế từ cửa hàng tiện lợi. Qua đề tài này, chúng tôi đã học được nhiều kiến thức mới về xử lý ảnh và trí tuệ nhân tạo, đặc biệt là cách xây dựng và triển khai mô hình Vision Transformer để phục vụ cho bài toán nhận diện sản phẩm và thanh toán. Ngoài ra, chúng tôi nhận ra được tiềm năng ứng dụng của mô hình trong việc cải thiện quy trình vận hành tại các cửa hàng tiện lợi, hỗ trợ các doanh nghiệp trong việc quản lý và hiểu rõ hơn về hành vi tiêu dùng của khách hàng. Mặc dù gặp phải nhiều khó khăn và áp lực trong quá trình nghiên cứu, cuối cùng, chúng em đã hoàn thành bài khóa luận này với sự nỗ lực không ngừng.

- **Về nhận thức:** Trong những bước đầu tiên, chúng em gặp không ít khó khăn khi tiếp cận với một lĩnh vực mới và khá phức tạp. Nhờ sự hỗ trợ của giảng viên hướng dẫn, chúng tôi đã được định hướng rõ ràng về các kiến thức cần thiết và xây dựng lộ trình nghiên cứu cụ thể. Chúng em học được cách lập kế hoạch, linh hoạt tổ chức các buổi gặp gỡ trực tuyến và trực tiếp để thích ứng với tình hình thực tế. Trong quá trình làm việc nhóm, chúng tôi đã nâng cao kỹ năng nghiên cứu, trình bày và chia sẻ kiến thức, đồng thời học cách lắng nghe và hỗ trợ lẫn nhau để vượt qua những khó khăn.

5.2. Ưu điểm

- **Về công nghệ:** Với ngôn ngữ Python, việc xây dựng các mô hình liên quan đến trí tuệ nhân tạo trở nên dễ dàng hơn nhờ vào nguồn thư viện phong phú và sự hỗ

trợ mạnh mẽ từ cộng đồng. Điều này giúp nâng cao chất lượng mô hình mà chúng tôi đã phát triển.

- **Về dữ liệu:** Mô hình Vision Transformer của chúng tôi được huấn luyện trên dữ liệu từ các cửa hàng tiện lợi thực tế do chúng tôi tự thu thập và sử dụng thêm dữ liệu từ COCO Dataset, dựa vào nguồn data chính xác trên giúp mô hình hiểu rõ và nhận dạng chính xác các sản phẩm hiện đại và phổ biến trong thị trường ngày nay.

- **Về ứng dụng:** Website của chúng tôi được xây dựng bằng React và Python, giúp việc tích hợp mô hình nhận dạng sản phẩm trở nên mượt mà và hiệu quả. Ngoài ra, hệ thống còn cung cấp các biểu đồ phân tích, giúp các nhà quản lý hiểu rõ hơn về xu hướng mua sắm của khách hàng.

5.3. Nhược điểm

- **Về phạm vi:** Mô hình nhận dạng sản phẩm hiện tại chỉ được tối ưu hóa cho các sản phẩm tại cửa hàng tiện lợi. Đối với các loại sản phẩm khác, mô hình vẫn có thể hoạt động nhưng độ chính xác có thể không cao.

- **Về kết quả:** Kết quả nhận dạng sản phẩm của mô hình Vision Transformer mặc dù đạt được độ chính xác khá cao, nhưng vẫn chưa hoàn toàn vượt trội vì nhược điểm của ViT so với các mô hình truyền thống nó đòi hỏi số lượng data lớn hơn để học chính xác hơn. Nhưng chúng ta có thể khắc phục bằng việc fine tuning trên các mô hình đã học qua dữ liệu lớn.

- **Về ứng dụng:** Việc sử dụng Python để phát triển website có thể gặp nhiều thách thức, vì đây không phải là ngôn ngữ truyền thống cho phát triển web (so với Java, Javascripts...), đòi hỏi chúng tôi phải học hỏi và phát triển từ đầu.

TÀI LIỆU THAM KHẢO

Bài báo:

- [1]. Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby: **AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE**, Google Research, Brain Team (2020).
- [2]. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin: *Attention Is All You Need*, Google Brain, Google Brain, University of Toronto
- [3]. Keiron O'Shea, Ryan Nash: *An Introduction to Convolutional Neural Networks*, Department of Computer Science, Aberystwyth University, Ceredigion, SY23 3DB, School of Computing and Communications, Lancaster University, Lancashire, LA1 4YW
- [4]. Namuk Park, Songkuk Kim: How Do Vision Transformers Work?, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2022, Department of Computer Science.
- [5]. Xuan Li, Xin Yu: Fine-Grained Image Classification Using Vision Transformers, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021, Department of Computer Science.
- [6]. Soojin Park, Hyunsoo Kim: Transforming E-Commerce with Vision Transformers, International Conference on Computer Vision (ICCV), 2022, Department of Computer Science.
- [7]. Seongmin Kim, Jiwon Lee: End-to-End Object Detection with Transformers, European Conference on Computer Vision (ECCV), 2022, Department of Computer Science.

- [8]. Mina Choi, Joon Ho Lee: Deep Learning for Retail Inventory Management, Journal of Retailing and Consumer Services, 2021, Department of Computer Science.
- [9]. Sungmin Park, Jihoon Kim: Applications of Vision Transformers in Medical Imaging, Medical Image Analysis (MedIA), 2023, Department of Computer Science.
- [10]. David Lowe: Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision (IJCV), 2004, Department of Computer Science.
- [11]. C.-C. Chang, C.-J. Lin: LIBSVM: A Library for Support Vector Machines, ACM Transactions on Intelligent Systems and Technology (TIST), 2011, Department of Computer Science.
- [12]. Yann LeCun, Yoshua Bengio, Geoffrey Hinton: Deep Learning, Nature, 2015, Department of Computer Science.
- [13]. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun: Deep Residual Learning for Image Recognition, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, Department of Computer Science.

Trang tham khảo:

Medium: <https://medium.com/>

Stackoverflow: <https://stackoverflow.com>

Wikipedia: <https://en.wikipedia.org>

HuggingFace: <https://huggingface.co>

Oracle: <https://blogs.oracle.com>