ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN ĐẮC THÀNH

NHẬN DẠNG VÀ PHÂN LOẠI HOA QUẢ TRONG ẢNH MÀU

LUẬN VĂN THẠC SĨ KỸ THUẬT PHẦN MỀM

Hà Nội – 2017

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN ĐẮC THÀNH

NHẬN DẠNG VÀ PHÂN LOẠI HOA QUẢ TRONG ẢNH MÀU

Ngành: Công nghệ thông tin
Chuyên ngành: Kỹ thuật phần mềm
Mã số: 60480103

LUẬN VĂN THẠC SĨ KỸ THUẬT PHẦN MỀM

NGƯỜI HƯỚNG DẪN KHOA HỌC: PGS. TS. LÊ THANH HÀ

NGƯỜI ĐỒNG HƯỚNG DẪN KHOA HỌC: TS. TRẦN QUỐC LONG

Hà Nội – 2017

Lời cam đoan

Tôi xin cam đoan đây là công trình nghiên cứu khoa học của riêng tôi và được sự hướng dẫn khoa học của PGS. TS. Lê Thanh Hà và TS. Trần Quốc Long. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong luận văn còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc. Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung luận văn của mình.

Học viên Cao học

Nguyễn Đắc Thành

Lời cảm ơn

Trước tiên, tôi xin bày tỏ sự biết ơn chân thành và sâu sắc nhất tới PGS. TS. Lê Thanh Hà – Giáo viên hướng dẫn trực tiếp và TS. Trần Quốc Long – Giáo viên đồng hướng dẫn của tôi, những người đã hết lòng hỗ trợ và giúp đỡ tôi trong quá trình nghiên cứu và hoàn thiện luận văn thạc sĩ của mình. Đồng thời tôi cũng gửi lời cám ơn chân thành đến Trần Tuấn Linh, thành viên nhóm đề tài, đã hỗ trợ tôi rất nhiều trong thời gian xây dựng cơ sở dữ liệu cũng như phát triển và cài đặt giải pháp cho bài toán trong luận văn này.

Tôi cũng xin gửi lời cảm ơn chân thành tới các thầy, các cô là giảng viên của trường Đại học Công nghệ đã tận tình dạy dỗ và hướng dẫn cho tôi trong suốt quá trình học tập thạc sĩ tại trường.

Và tôi cũng xin gửi lời cảm ơn tới bố mẹ và những người thân trong gia đình vì đã nuôi nấng, dạy dỗ, chăm lo cho tôi, động viên tôi hoàn thành thật tốt khóa học thạc sĩ này.

Mặc dù đã hết sức cố gắng hoàn thành luận văn nhưng chắc chắn sẽ không tránh khỏi những sai sót. Kính mong nhận được sự cảm thông, chỉ bảo tận tình của các quý thầy cô và các bạn.

Tôi xin chân thành cảm ơn!

Mục lục

Danh mục hình vẽ

Danh mục bảng biểu

Danh mục từ viết tắt

STT	Từ viết tắt	Ý nghĩa
1	CSDL	Cơ sở dữ liệu
2	CNN	Convolutional Neural Network – Mạng nơ ron tích chập
3	ReLU	Rectified Linear Unit – Tinh chỉnh đơn vị tuyến tính
4	GPU	Graphics Processing Unit – Bộ vi xử lý đồ họa

MỞ ĐẦU

1. Tính cấp thiết của đề tài luận văn

Hiện nay, ở nước ta nói riêng và ở các nước đang phát triển có nền nông nghiệp là một trong các ngành sản xuất chủ yếu, quá trình thu hoạch, phân loại và đánh giá chất lượng các loại sản phẩm nông nghiệp, đặc biệt là các loại hoa quả, chủ yếu còn phải thực hiện bằng các phương pháp thủ công. Đây là công việc không quá khó, nhưng tiêu tốn nhiều thời gian, công sức của con người và là rào cản đối với mở rộng phát triển quy mô sản xuất nông nghiệp. Do đó, nhiều phương pháp tự động hóa công việc thu hoạch, nhận dạng và đánh giá chất lượng hoa quả đã được nghiên cứu và đưa vào ứng dụng thực tế, trong đó sử dụng chủ yếu các phương pháp Xử lý ảnh đơn thuần. Tuy nhiên, các phương pháp này vẫn chưa thực sự thỏa mãn yêu cầu về khả năng nhận dạng một số lượng lớn các loại hoa quả với độ chính xác cao do bị hạn chế bởi các đặc trưng của bài toán nhận dạng hoa quả: số lượng chủng loại lớn với nhiều loại hoa quả hết sức tương tự nhau, sự biến thiên về hình dạng, màu sắc, chi tiết trong từng loại quả cũng rất khó dự đoán trước…

Trong thời gian gần đây, nhờ có sự phát triển mạnh mẽ về khả năng tính toán của các thế hệ máy tính hiện đại cũng như sự bùng nổ về dữ liệu thông qua mạng lưới Internet trải rộng, ta đã chứng kiến nhiều sự đột phá trong lĩnh vực Học máy, đặc biệt là trong lĩnh vực Thị giác máy tính. Sự quay lại và phát triển vượt bậc của các phương pháp Học sâu đã giúp Thị giác máy tính đạt được những thành tựu đáng kể trong lĩnh vực Nhận dạng ảnh, trong đó có bài toán nhận dạng hoa quả. Đề tài nghiên cứu "Nhận dạng và phần loại hoa quả trong ảnh màu" đã được đưa ra với hy vọng có thể ứng dụng thành công các mô hình học sâu hiện đại để xây dựng một hệ thống nhận dạng hoa quả tự động, đặc biệt là đối với các loại hoa quả phổ biến tại nước ta.

2. Mục tiêu của luận văn

Do thời gian hạn chế trong thời gian thực hiện nghiên cứu, luận văn trước hết tập trung nghiên cứu, tìm hiểu và so sánh các phương pháp Học máy truyền thống với phương pháp Học sâu, đồng thời thực hiện cài đặt một mô hình huấn luyện về nhận dạng ảnh trong Học sâu với số lượng hoa quả được hạn chế, và sử dụng chúng làm bộ nhận dạng cơ sở cho ứng dụng hỗ trợ nhận dạng hoa quả trên điện thoại thông minh.

2.1. Cơ sở dữ liệu ảnh hoa quả

Bộ cơ sở dữ liệu ảnh là một trong các thành phần quan trọng hàng đầu trong các phương pháp Học máy nói chung, được sử dụng để phục vụ cho quá trình tính toán tham số và huấn luyện, tinh chỉnh các mô hình. Thông thường, bộ dữ liệu càng lớn và càng được chọn lọc tỉ mỉ cẩn thận thì độ chính xác của mô hình càng được cải thiện, nhưng

trong phạm vi luận văn này kích thước CSDL sẽ được hạn chế, cả về số lượng loại hoa quả sẽ nhận dạng cũng như số lượng ảnh chụp cho mỗi loại hoa quả đó. Cụ thể:

- Số lượng hoa quả sẽ nhận dạng: 40 loại hoa quả phổ biến ở nước ta như nho, táo, chuối, thanh long…
- Số lượng ảnh gốc cho mỗi loại quả: 500-1000 ảnh, bao gồm các ảnh chụp hoa quả ở các góc độ khác nhau với nền tùy ý, có thể lấy từ nguồn trên mạng hoặc tự chụp bằng thiết bị camera cá nhân.

Sau khi đã thu thập đủ số lượng ảnh gốc cho các loại hoa quả, ta sẽ sử dụng các thuật toán chỉnh sửa ảnh, như làm nghiêng ảnh, chèn thêm nhiễu hoặc ghép ảnh với nền khác, để tạo thêm ảnh mới nhằm tăng cường kích thước cơ sở dữ liệu.

2.2. Bộ huấn luyện nhận dạng hoa quả

Để đưa ra đánh giá tổng quát và so sánh độ chính xác tương đối giữa các phương pháp Học máy truyền thống với phương pháp Học sâu, luận văn thực hiện cài đặt một mạng huấn luyện nơ-ron nhân tạo truyền thống và một mạng huấn luyện nơ-ron tích chập trong Học sâu, sau khi thực hiện huấn luyện trên cùng bộ cơ sở dữ liệu ảnh và so sánh kết quả.

Đối với phương pháp Học máy truyền thống: nghiên cứu, tìm hiểu các phương pháp đã được trình bày trong các bài báo, công trình khoa học và thống kê ra các đặc trưng thường được sử dụng và cho kết quả huấn luyện tốt nhất. Các đặc trưng này thể hiện thông tin của hoa quả về màu sắc, hình dạng và kết cấu, và được đưa vào bộ tính toán, trích chọn đặc trưng của mạng nơ-ron nhân tạo.

Đối với mạng nơ-ron tích chập thuộc nhóm Học sâu: tìm hiểu và chọn một trong các mô hình huấn luyện phổ biến trong lĩnh vực Nhận dạng ảnh trên thế giới để thực hiện cài đặt và so sánh kết quả với bộ nhận dạng truyền thống.

2.3. Ứng dụng nhận dạng hoa quả

Một trong các mục tiêu của luận văn là xây dựng thành công một ứng dụng đơn giản trên điện thoại thông minh nhằm hỗ trợ người dùng nhận dạng hoa quả. Nguyên nhân chọn điện thoại thông minh làm nền tảng cho ứng dụng vì sự phổ biến cũng như tính cơ động của thiết bị, điều này giúp cho ứng dụng dễ dàng được phổ biến hơn từ đó hỗ trợ việc thu thập ảnh chụp cho cơ sở dữ liệu từ các cộng tác viên sử dụng ứng dụng.

Hệ thống nhận dạng hoa quả - Fruit Recognition System - ngoài ứng dụng client trên điện thoại thông minh còn có một máy chủ server để thực hiện tất cả các bước huấn luyện và nạp mô hình nhận dạng, các bước tính toán nhận dạng loại hoa quả dựa trên ảnh chụp nhận được từ ứng dụng client. Việc đặt mọi tính toán xử lý trên máy chủ nhằm mục đích quản lý tập trung, tăng hiệu năng tính toán cũng như đơn giản hóa ứng dụng client trên điện thoại thông minh, giúp ứng dụng không bị hạn chế bởi các nền tảng, môi trường khác nhau.

3. Cấu trúc của luận văn

Dựa trên mục tiêu cụ thể đã trình bày trong phần trước, luận văn được tổ chức thành năm chương với các nội dung cụ thể như sau:

Chương 1: Trong chương tổng quan này, ta sẽ có ra cái nhìn tổng quan về các hướng tiếp cận và giải pháp đã được ứng dụng trong bài toán nhận dạng phân loại hoa quả, từ các phương pháp thuần tính toán xử lý ảnh tương đối thô sơ cho tới các phương pháp Học máy truyền thống và cuối cùng là các phương pháp Học sâu - một nhánh đặc biệt trong Học máy.

Chương 2: Chương này sẽ đi sâu hơn vào một mạng huấn luyện trong Học sâu thường được sử dụng trong lĩnh vực Nhận dạng ảnh - mạng nơ-ron tích chập, và tìm hiểu chìa khóa giải quyết bài toán nhận dạng ảnh với bộ dữ liệu huấn luyện có kích thước tương đối nhỏ.

Chương 3: Trong chương tiếp theo, ta sẽ đi vào phần mô tả tổng quan Hệ thống nhận dạng hoa quả tự động, với các mô đun chính như máy chủ, máy trạm, bộ huấn luyện và nhận dạng … Ngoài ra, cách thức thu thập, chỉnh sửa cơ sở dữ liệu ảnh và cách cài đặt triển khai môi trường huấn luyện cho mô hình mạng nơ-ron tích chập đã chọn trong chương 2 cũng sẽ được trình bày cụ thể tại đây.

Chương 4: Chương 4 tập trung trình bày về kết quả thực nghiệm, bao gồm kết quả so sánh độ chính xác giữa các phương pháp Học máy truyền thống với phương pháp Học sâu, cùng với các đánh giá về độ hiệu quả của bộ tạo dữ liệu ảnh nhiễu cũng như các ảnh chụp thực tế khi được sử dụng trong thực tế. Dựa trên các kết quả thực nghiệm này, ta sẽ đưa ra một số phân tích và kết luận về điểm mạnh và điểm hạn chế của mô hình huấn luyện Học sâu đã chọn.

Chương 5: Cuối cùng, chương 5 sẽ tổng kết các nội dung đã trình bày trong luận văn, từ đó đề xuất các phương hướng nghiên cứu tiếp theo để tiếp tục cải thiện chất lượng nhận dạng của hệ thống.

Chương 1. Giới thiệu tổng quan

1.1. Bài toán nhận dạng và phân loại hoa quả

Nhận dạng vật thể trong ảnh được coi là bài toán cơ bản nhất trong lĩnh vực Thị giác máy tính, là nền tảng cho rất nhiều bài toán mở rộng khác như bài toán phân lớp, định vị, tách biệt vật thể.... Tuy bài toán cơ bản này đã tồn tại hàng thế kỷ nhưng con người vẫn chưa thể giải quyết nó một cách triệt để, do tồn tại rất nhiều khó khăn để máy tính có thể hiểu được các thông tin trong một bức ảnh. Trong đó, những khó khăn tiêu biểu [3] phải kể đến:

- Sự đa dạng trong điểm nhìn – Viewpoint: Cùng một vật thể nhưng có thể có rất nhiều vị trí và góc nhìn khác nhau, dẫn đến các hình ảnh thu được về vật thể đó sẽ không giống nhau. Việc huấn luyện để máy tính có thể hiểu được điều này thực sự là một thách thức khó khăn.

- Sự đa dạng trong kích thước: Các bức ảnh không có cách nào thể hiện trường thông tin về kích thước của vật thể trong đời thực, và máy tính cũng chỉ có thể tính toán được tỉ lệ tương đối của vật thể so với bức ảnh bằng cách đếm theo số lượng các điểm ảnh vật thể đó chiếm trong ảnh.

- Các điều kiện khác nhau của chiếu sáng: Ánh sáng có ảnh hưởng mạnh mẽ đến thông tin thể hiện trong một bức ảnh, đặc biệt là ở mức độ thấp như mức độ điểm ảnh.

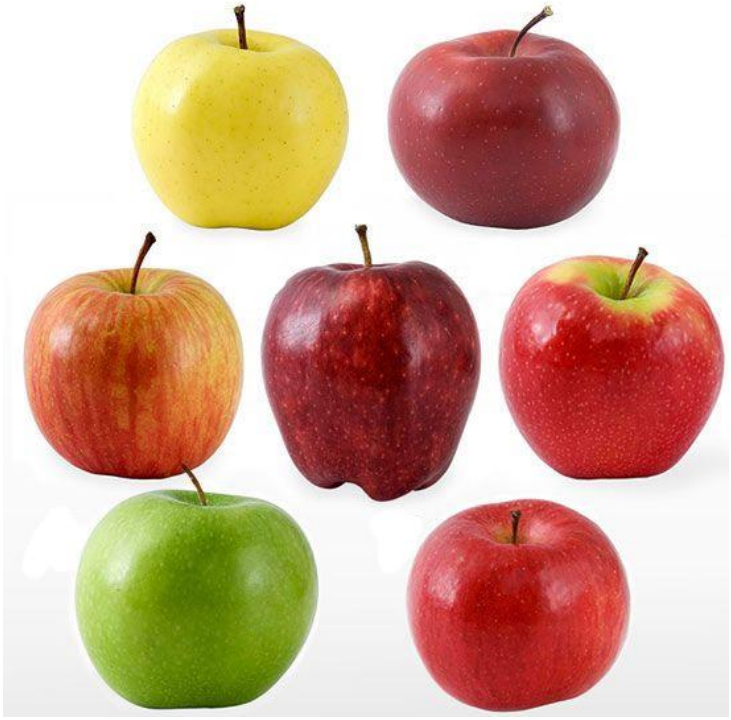
- Sự ẩn giấu một phần của vật thể sau các đối tượng khác trong ảnh: Trong các bức ảnh, vật thể không nhất định phải xuất hiện với đầy đủ hình dạng mà có thể bị che lấp một phần nào đó bởi nền hoặc các vật thể xung quanh. Sự không đầy đủ về hình dạng của vật thể sẽ dẫn đến việc thiếu thông tin, đặc trưng và càng làm bài toán nhận dạng khó khăn hơn.

- Sự lộn xộn phức tạp của nền: Trong nhiều trường hợp, vật thể cần nhận dạng bị lẫn gần như hoàn toàn vào nền của bức ảnh, sự lẫn lộn về màu sắc, họa tiết giữa vật thể và nền khiến cho việc nhận dạng trở nên vô cùng khó khăn, kể cả với thị giác con người.

- Sự đa dạng về chủng loại vật thể: Vật thể cần nhận dạng có thể bao gồm nhiều chủng loại khác nhau, với hình dạng, màu sắc, kết cấu vô cùng khác biệt. Đây chính là một thách thức nữa với bài toán nhận dạng, đó là làm thế nào để các mô hình nhận dạng của máy tính có thể nhận biết được các biến thể về chủng loại của vật thể, ví dụ các loại ghế khác nhau, trong khi vẫn tách biệt được đâu là các vật thể khác loại, ví dụ phân biệt bàn với ghế...

Hình 1.1: Các khó khăn trong bài toán nhận dạng vật thể trong ảnh

Là một trường hợp cụ thể của bài toán nhận dạng và phân lớp, bài toán nhận dạng hoa quả kế thừa các khó khăn vốn có của bài toán gốc, và kèm theo là các khó khăn riêng của chính nó, như: số lượng khổng lồ về chủng loại hoa quả theo mùa, vùng miền, địa hình… với vô số loại hoa quả có hình dáng, màu sắc, kết cấu giống nhau, dải biến thiên màu sắc theo chu kỳ phát triển của quả từ lúc còn xanh đến lúc chín, hay sự đa dạng về hình dạng của cùng một loại quả do ảnh hưởng của thời tiết, điều kiện thổ nhưỡng và chế độ dinh dưỡng…

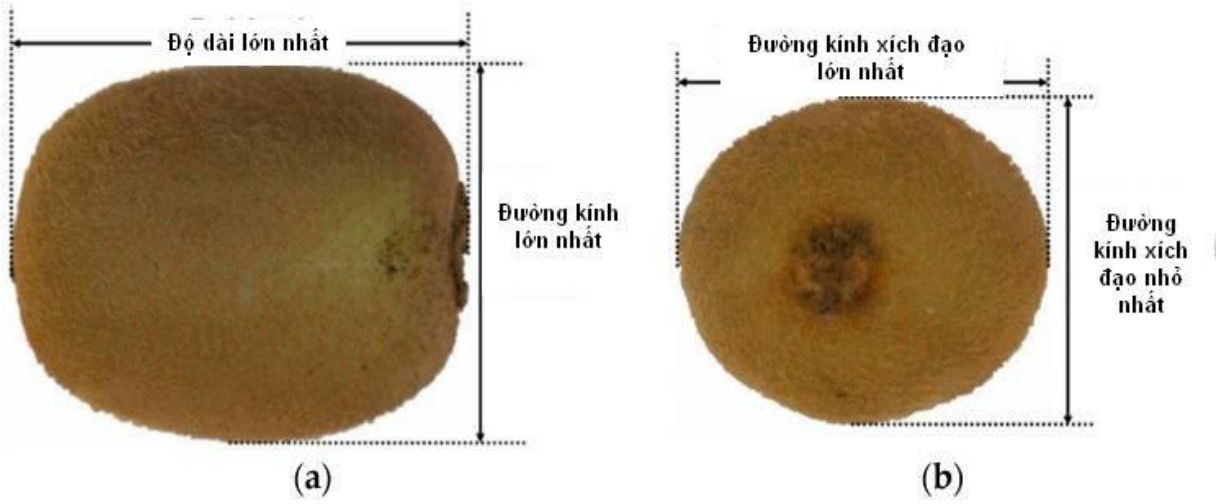


Hình 1.2: Sự đa dạng về chủng loại của một loại hoa quả

1.2. Các hướng tiếp cận và giải quyết bài toán

Bài toán tự động nhận dạng hoa quả đã xuất hiện từ lâu và đã có rất nhiều bài báo, công trình khoa học được đưa ra nhằm đề xuất hoặc cải tiến các thuật toán nhận dạng. Trong đó, xuất hiện sớm nhất là các phương pháp Xử lý ảnh – Image Processing,

các phương pháp này tập trung vào phát triển các thuật toán nhằm trích xuất thông tin, ví dụ các tham số về màu sắc, hình dạng, kết cấu, kích thước…, từ bức ảnh đầu vào để nhận dạng hoa quả [4, 5]. Do chỉ đơn thuần xử lý trên một vài ảnh đầu vào trong khi sự biến thiên về màu sắc, hình dạng, kích thước… của hoa quả quá phức tạp, kết quả đạt được của các phương pháp này không được cao và phạm vi áp dụng trên số lượng loại hoa quả cũng bị hạn chế.

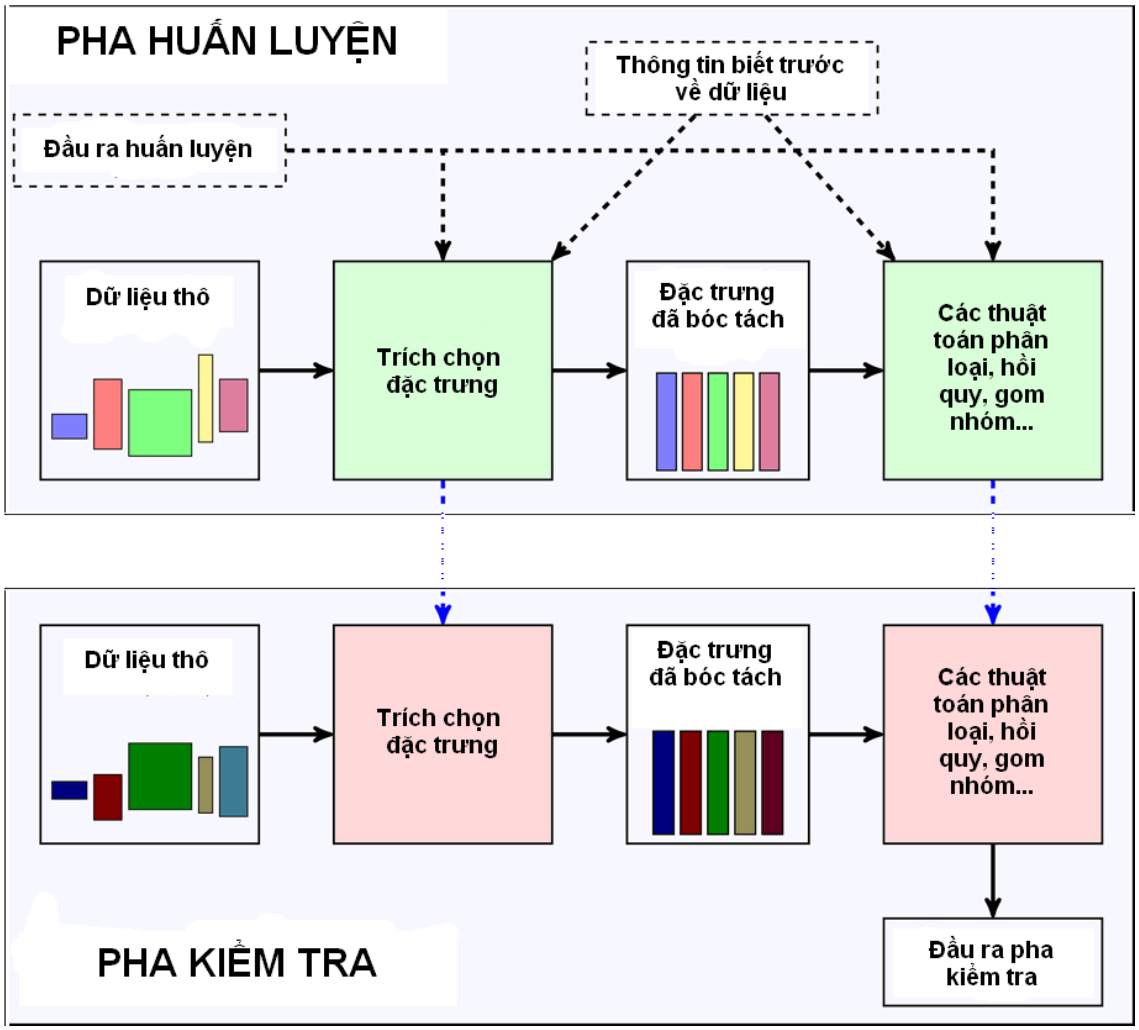


Hình 1.3: Các thông tin về hình học được tính toán bởi các thuật toán Xử lý ảnh

Bắt đầu từ những năm 2000s, sau khi xuất hiện một bài bài báo khoa học đề xuất áp dụng phương pháp Học máy - Machine Learning - vào bài toán nhận dạng hoa quả với độ chính xác cao [6], hướng giải quyết bài toán đã tập trung vào ứng dụng và cải tiến các thuật toán Học máy, cụ thể là nghiên cứu, thử nghiệm trích chọn các đặc trưng phù hợp nhất để đưa vào huấn luyện bộ nhận dạng tự động [7-9]. Kết quả thu được tương đối khả quan, khả năng nhận dạng hoa quả tự động đã được cải thiện với số lượng loại hoa quả được mở rộng và độ chính xác của nhận dạng cao hơn nhiều so với các phương pháp thuần Xử lý ảnh ban đầu. Nối tiếp sự phát triển của Học máy, trong những năm gần đây, nhờ sự phát triển vượt bậc về sức mạnh tính toán của các máy tính cũng như sự bùng nổ dữ liệu trên Internet, một nhánh đặc biệt trong Học máy là Học sâu - Deep Learning đã đạt được nhiều thành tựu đáng kể, đặc biệt là trong lĩnh vực Xử lý ảnh và ngôn ngữ tự nhiên. Học sâu cũng đã được áp dụng rất thành công vào bài toán nhận dạng hoa quả, trong các thử nghiệm với phạm vi hạn chế về số lượng loại hoa quả cần nhận dạng, phương pháp này đã đạt được kết quả rất cao. Sau đây ta sẽ tìm hiểu sâu hơn về hai tiếp cận chính hiện nay để giải quyết bài toán nhận dạng hoa quả nói riêng và nhận dạng vật thể trong ảnh nói chung: phương pháp Học sâu và các phương pháp Học máy truyền thống không sử dụng Học sâu.

1.2.1. Phương pháp Học máy truyền thống

Mô hình hoạt động chung của các phương pháp Học máy truyền thống được thể hiện trong Hình 1.4 dưới đây [2]:



Hình 1.4: Mô hình hoạt động chung của các phương pháp Học máy [2]

Từ hình ta có thể thấy Học máy gồm hai giai đoạn chính là Huấn luyện – Training và Thử nghiệm – Testing, trong mỗi giai đoạn đều sử dụng hai thành phần quan trọng nhất do người xử lý bài toán thiết kế, đó là Trích chọn đặc trưng – Feature Engineering (hay còn gọi là Feature Extraction) và Thuật toán phân loại, nhận dạng… - Algorithms. Hai thành phần này có ảnh hưởng trực tiếp đến kết quả bài toán, vì thế được thiết kế rất cẩn thận, tốn nhiều thời gian, đòi hỏi người thiết kế phải có kiến thức chuyên môn và nắm rõ đặc điểm của bài toán cần xử lý.

1.2.1.1. Trích chọn đặc trưng

Trong các bài toán thực tế, ta chỉ có được những dữ liệu thô chưa qua chọn lọc xử lý, và để có thể đưa các dữ liệu này vào huấn luyện ta cần có những phép biến đổi để biến các dữ liệu thô thành dữ liệu chuẩn, với khả năng biểu diễn dữ liệu tốt hơn. Các phép biến đổi bao gồm loại bỏ dữ liệu nhiễu và tính toán để lưu lại các thông tin đặc trưng, có ý nghĩa từ dữ liệu thô ban đầu. Các thông tin đặc trưng này là khác nhau với từng loại dữ liệu và bài toán cụ thể, vì thế trong từng trường hợp phép biến đổi này cần phải được tùy biến một cách thích hợp để cải thiện độ chính xác của mô hình dự đoán. Quá trình này được gọi là *Trích chọn đặc trưng* – Feature Engineering, là một thành phần rất quan trọng trong các phương pháp Học máy truyền thống.

- **Đầu vào**: Toàn bộ thông tin của dữ liệu, không có quy chuẩn về dạng thông tin (véc tơ, ma trận…) hay kích thước các chiều thông tin. Đồng thời, do chứa toàn bộ thông tin, gồm cả thông tin nhiễu và không có giá trị nên kích thước lưu trữ thường lớn và không có lợi cho tính toán sau này.
- **Đầu ra**: Các thông tin hữu ích đã được tính toán, rút ra từ dữ liệu đầu vào, trong đó không còn các thành phần nhiễu hay vô nghĩa. Kích thước dữ liệu đầu ra đã được rút gọn rất nhiều so với kích thước dữ liệu đầu vào, giúp cho việc tính toán về sau trở nên nhanh gọn, thuận tiện hơn rất nhiều.
- **Thông tin biết trước về dữ liệu**: Đây là thành phần tùy chọn, không bắt buộc với mọi bài toán, mà chỉ xuất hiện trong một số trường hợp cụ thể với những thông tin rõ ràng về đặc trưng hữu ích với mô hình dự đoán. Các thông tin biết trước này giúp người thiết kế có thể lựa chọn được những đặc trưng tốt nhất và các phương pháp tính toán phù hợp nhất để ra được mô hình dự đoán với độ chính xác cao.

1.2.1.2. Thuật toán

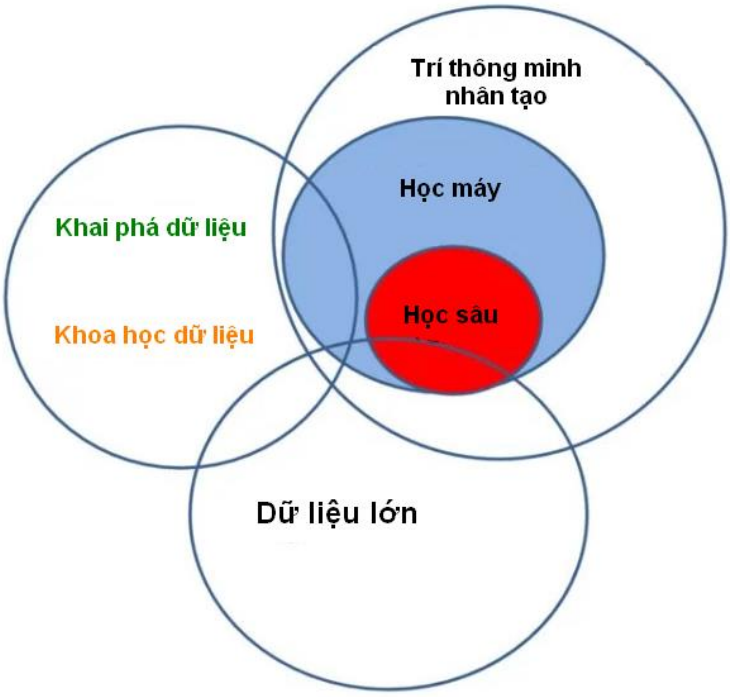
Sau quá trình trích chọn đặc trưng ở bước trước, ta có được các đặc trưng, được lưu trữ ở định dạng chuẩn về kiểu dữ liệu, kích thước dữ liệu…, và các thông tin đặc trưng này có thể được sử dụng cùng với các thông tin biết trước về dữ liệu (nếu có) để xây dựng ra các mô hình dự đoán phù hợp bằng các thuật toán khác nhau. Các thuật toán trong Học máy thường được phân loại theo hai cách phổ biến là theo phương thức học hoặc theo chức năng của thuật toán, ví dụ như:

- Phân nhóm theo phương thức học: Học giám sát và Học không giám sát (Supervised và Unsupervised Learning)
- Phân nhóm theo chức năng: Các thuật toán hồi quy, phân loại, gom nhóm…

Một đặc điểm nổi bật của các phương pháp Học máy truyền thống là độ chính xác của mô hình dự đoán phụ thuộc rất nhiều vào chất lượng các đặc trưng được lựa chọn, các đặc trưng này càng phù hợp với bài toán đưa ra thì kết quả thu được càng tốt. Đây là điểm mạnh, và cũng là điểm yếu của các phương pháp này, bởi việc trích chọn đặc trưng chính là sự đóng góp của bản tay con người trong việc cải tiến các mô hình, nó yêu cầu sự hiểu biết thấu đáo về bài toán cần giải quyết, các thuật toán sử dụng và các thông số trong mô hình huấn luyện. Các đặc trưng được thiết kế riêng cho từng bài toán khác biệt, do vậy hiếm khi chúng có thể được tái sử dụng với các bài toán mới mà cần phải được cải thiện hay thay thế bởi các đặc trưng khác.

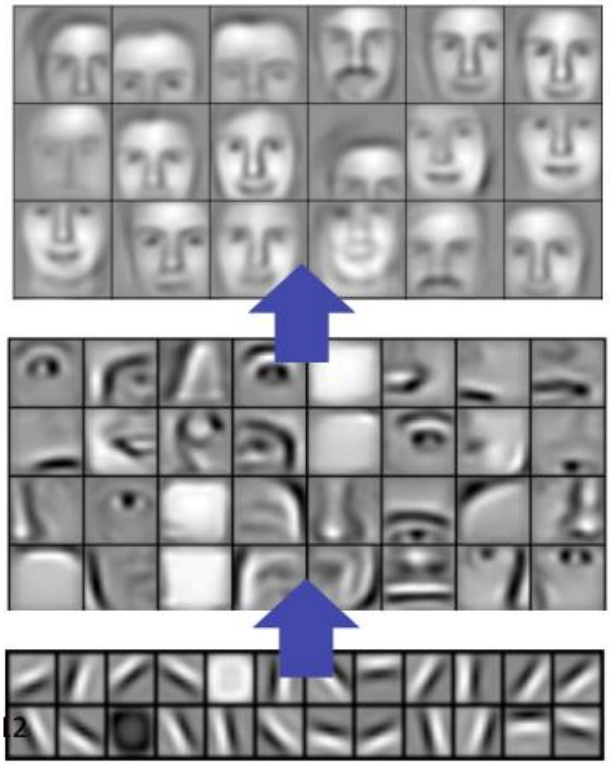
1.2.2. Phương pháp Học sâu

Học sâu là một nhánh đặc biệt của ngành Học máy, và bắt đầu trở nên phổ biến trong thập kỷ gần đây do các nhà khoa học đã có thể tận dụng khả năng tính toán mạnh mẽ của các máy tính hiện đại cũng như khối lượng dữ liệu khổng lồ (hình ảnh, âm thanh, văn bản,…) trên Internet. Ta có thể thấy rõ mối quan hệ giữa Học sâu với Học máy cũng như các lĩnh vực liên quan khác qua hình ảnh mô tả bên dưới (Hình 1.5) [10]:



Hình 1.5: Mối quan hệ của Học sâu với các lĩnh vực liên quan

Các mạng huấn luyện theo phương pháp Học sâu còn được gọi với cái tên khác là mạng nơ-ron sâu (Deep Neural Network) do cách thức hoạt động của chúng. Về cơ bản, các mạng này bao gồm rất nhiều lớp khác nhau, mỗi lớp sẽ phân tích dữ liệu đầu vào theo các khía cạnh khác nhau và theo mức độ trừu tượng nâng cao dần (xem Hình 1.6).



Hình 1.6: Mức độ trừu tượng tăng dần qua các tầng học của Học sâu [11]

Cụ thể, với một mạng Học sâu cho nhận dạng ảnh, các lớp đầu tiên trong mạng chỉ làm nhiệm vụ rất đơn giản là tìm kiếm các đường thẳng, đường cong, hoặc đốm màu trong ảnh đầu vào. Các thông tin này sẽ được sử dụng làm đầu vào cho các lớp tiếp theo, với nhiệm vụ khó hơn là từ các đường, các cạnh đó tìm ra các thành phần của vật thể trong ảnh. Cuối cùng, các lớp cao nhất trong mạng huấn luyện sẽ nhận nhiệm vụ phát hiện ra vật thể trong ảnh.

Với cách thức học thông tin từ ảnh lần lượt qua rất nhiều lớp, nhiều tầng khác nhau như vậy, các phương pháp này có thể giúp cho máy tính hiểu được những dữ liệu phức tạp bằng nhiều lớp thông tin đơn giản qua từng bước phân tích. Đó cũng là lý do chúng được gọi là các phương pháp Học sâu.

Tuy có nhiều điểm ưu việt trong khả năng huấn luyện máy tính cho các bài toán phức tạp, Học sâu vẫn còn rất nhiều giới hạn khiến nó chưa thể được áp dụng vào giải quyết mọi vấn đề. Điểm hạn chế lớn nhất của phương pháp này là yêu cầu về kích thước dữ liệu huấn luyện, mô hình huấn luyện Học sâu đòi hỏi phải có một lượng khổng lồ dữ liệu đầu vào để có thể thực hiện việc học qua nhiều lớp với một số lượng lớn nơ-ron và tham số. Đồng thời, việc tính toán trên quy mô dữ liệu và tham số lớn như vậy cũng yêu cầu đến sức mạnh xử lý của các máy tính server cỡ lớn. Quy trình chọn lọc dữ liệu cũng như huấn luyện mô hình đều tốn nhiều thời gian và công sức, dẫn đến việc thử nghiệm các tham số mới cho mô hình là công việc xa xỉ, khó thực hiện. Tuy nhiên, nhờ các phương pháp Học tập chuyển giao, hiện nay điểm hạn chế lớn nhất này đã không còn là vấn đề quá nghiêm trọng như trước – điều này sẽ được trình bày cụ thể trong các chương sau.

Ngoài hạn chế về kích thước dữ liệu đầu vào, Học sâu còn chưa đủ thông minh để nhận biết và hiểu được các logic phức tạp như con người, các tác vụ do chúng thực hiện vẫn tương đối máy móc và cần cải thiện để "thông minh" hơn nữa. Trong ví dụ Hình 1.7, ta có thể nhận thấy sự vô lý trong bức ảnh về quả tạ hai đầu mà mạng Học sâu tạo ra sau khi được huấn luyện với hàng loạt ảnh mẫu. Bức ảnh có chứa các phần ảnh về cánh tay con người, là thành phần không phải thuộc về quả tạ. Việc hình ảnh cánh tay xuất hiện trong phần lớn các ảnh mẫu đã dẫn đến sự nhầm lẫn của mô hình dự đoán này.

Hình 1.7: Bức ảnh quả tạ hai đầu sinh ra bởi mô hình dự đoán Học sâu

Như đã trình bày trong phần mở đầu, mục đích của luận văn là tìm hiểu và ứng dụng một mô hình Học sâu vào bài toán nhận dạng, phân loại hoa quả, nguyên nhân chính khiến Học sâu được chọn làm giải pháp là bởi khả năng mạnh mẽ vượt trội của nó đối với các phương pháp Học máy truyền thống khi áp dụng vào các bài toán nhận dạng vật thể, trong đó vật thể là các đối tượng rất khó chọn lọc đặc trưng phù hợp, cụ thể với trường hợp này là các loại hoa quả. Để chứng minh cho nhận định này, luận văn đã thực hiện phép so sánh độ chính xác của hai mô hình nhận dạng, được huấn luyện lần lượt bởi hai phương pháp trên với cùng bộ dữ liệu đầu vào. Kết quả cụ thể sẽ được trình bày trong Chương 4 – Kết quả thực nghiệm và Đánh giá.

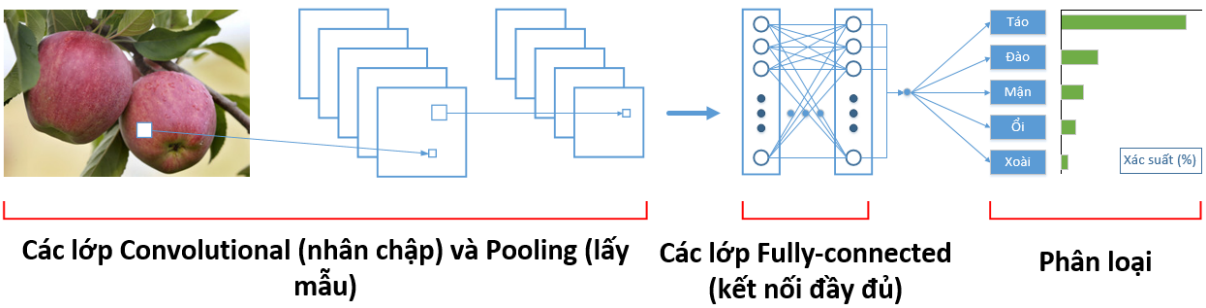
Chương 2. Mạng nơ-ron tích chập

Mạng nơ-ron tích chập (CNN - Convolutional Neural Network) là một trong những mô hình mạng Học sâu phổ biến nhất hiện nay, có khả năng nhận dạng và phân loại hình ảnh với độ chính xác rất cao, thậm chí còn tốt hơn con người trong nhiều trường hợp. Mô hình này đã và đang được phát triển, ứng dụng vào các hệ thống xử lý ảnh lớn của Facebook, Google hay Amazon… cho các mục đích khác nhau như các thuật toán tagging tự động, tìm kiếm ảnh hoặc gợi ý sản phẩm cho người tiêu dùng.

Sự ra đời của mạng CNN là dựa trên ý tưởng cải tiến cách thức các mạng nơ-ron nhân tạo truyền thống học thông tin trong ảnh. Do sử dụng các liên kết đầy đủ giữa các điểm ảnh vào node, các mạng nơ-ron nhân tạo truyền thẳng (Feedforward Neural Network) bị hạn chế rất nhiều bởi kích thước của ảnh, ảnh càng lớn thì số lượng liên kết càng tăng nhanh và kéo theo sự bùng nổ khối lượng tính toán. Ngoài ra sự liên kết đầy đủ này cũng là sự dư thừa khi với mỗi bức ảnh, các thông tin chủ yếu thể hiện qua sự phụ thuộc giữa các điểm ảnh với những điểm xung quanh nó mà không quan tâm nhiều đến các điểm ảnh ở cách xa nhau. Mạng CNN ra đời với kiến trúc thay đổi, có khả năng xây dựng liên kết chỉ sử dụng một phần cục bộ trong ảnh kết nối đến node trong lớp tiếp theo thay vì toàn bộ ảnh như trong mạng nơ-ron truyền thẳng.

2.1. Kiến trúc Mạng nơ-ron tích chập

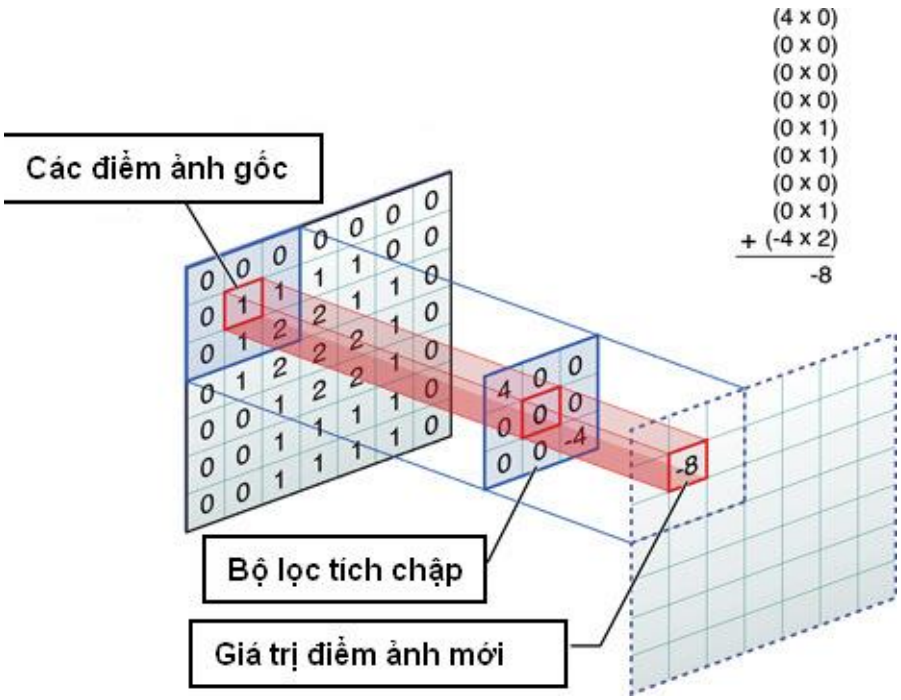
Các lớp cơ bản trong một mạng CNN bao gồm: Lớp tích chập (Convolutional), Lớp kích hoạt phi tuyến ReLU (Rectified Linear Unit), Lớp lấy mẫu (Pooling) và Lớp kết nối đầy đủ (Fully-connected), được thay đổi về số lượng và cách sắp xếp để tạo ra các mô hình huấn luyện phù hợp cho từng bài toán khác nhau.



Hình 2.1: Kiến trúc cơ bản của một mạng tích chập

- **Lớp tích chập**:

Đây là thành phần quan trọng nhất trong mạng CNN, cũng là nơi thể hiện tư tưởng xây dựng sự liên kết cục bộ thay vì kết nối toàn bộ các điểm ảnh. Các liên kết cục bộ này được tính toán bằng phép tích chập giữa các giá trị điểm ảnh trong một vùng ảnh cục bộ với các bộ lọc – filters – có kích thước nhỏ.



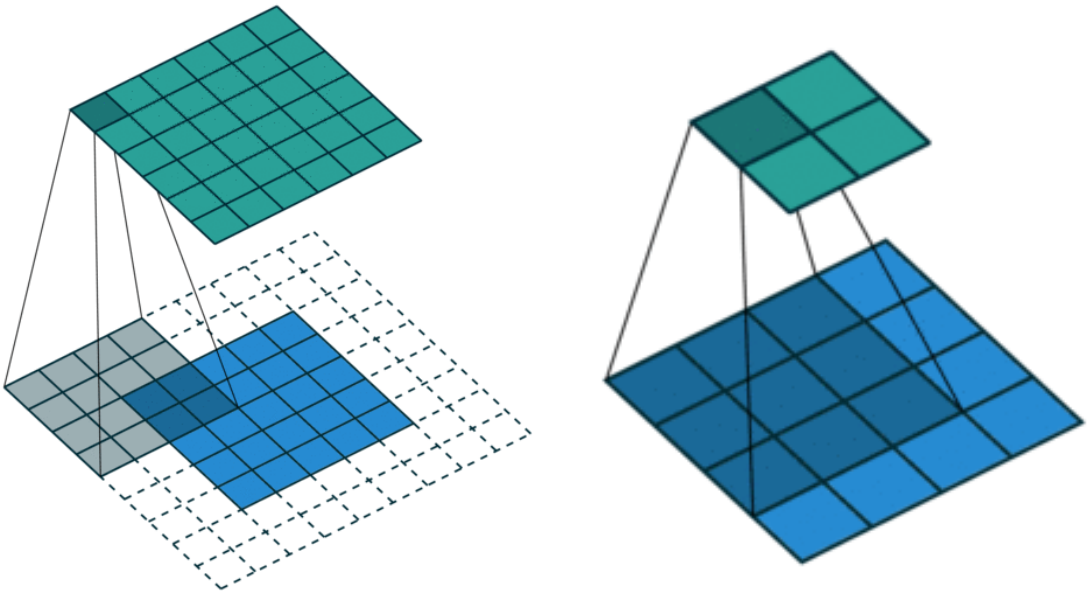
Hình 2.2: Ví dụ bộ lọc tích chập được sử dụng trên ma trận điểm ảnh

Trong ví dụ ở Hình 2.2 [12], ta thấy bộ lọc được sử dụng là một ma trận có kích thước 3x3. Bộ lọc này được dịch chuyển lần lượt qua từng vùng ảnh đến khi hoàn thành quét toàn bộ bức ảnh, tạo ra một bức ảnh mới có kích thước nhỏ hơn hoặc bằng với kích thước ảnh đầu vào. Kích thước này được quyết định tùy theo kích thước các khoảng trắng được thêm ở viền bức ảnh gốc và được tính theo công thức (1) [13]:

$$o = \frac{i+2*p-k}{s} + 1 \quad (1)$$

Trong đó:

- o: kích thước ảnh đầu ra
- i: kích thước ảnh đầu vào
- p: kích thước khoảng trắng phía ngoài viền của ảnh gốc
- k: kích thước bộ lọc
- s: bước trượt của bộ lọc



Hình 2.3: Trường hợp thêm/không thêm viền trắng vào ảnh khi tích chập

Như vậy, sau khi đưa một bức ảnh đầu vào cho lớp Tích chập ta nhận được kết quả đầu ra là một loạt ảnh tương ứng với các bộ lọc đã được sử dụng để thực hiện phép tích chập. Các trọng số của các bộ lọc này được khởi tạo ngẫu nhiên trong lần đầu tiên và sẽ được cải thiện dần xuyên suốt quá trình huấn luyện.

- **Lớp kích hoạt phi tuyến ReLU**:

Lớp này được xây dựng với ý nghĩa đảm bảo tính phi tuyến của mô hình huấn luyện sau khi đã thực hiện một loạt các phép tính toán tuyến tính qua các lớp Tích chập. Lớp Kích hoạt phi tuyến nói chung sử dụng các hàm kích hoạt phi tuyến như ReLU hoặc sigmoid, tanh… để giới hạn phạm vi biên độ cho phép của giá trị đầu ra. Trong số các hàm kích hoạt này, hàm ReLU được chọn do cài đặt đơn giản, tốc độ xử lý nhanh mà vẫn đảm bảo được tính toán hiệu quả. Cụ thể, phép tính toán của hàm ReLU chỉ đơn giản là chuyển tất cả các giá trị âm thành giá trị 0.

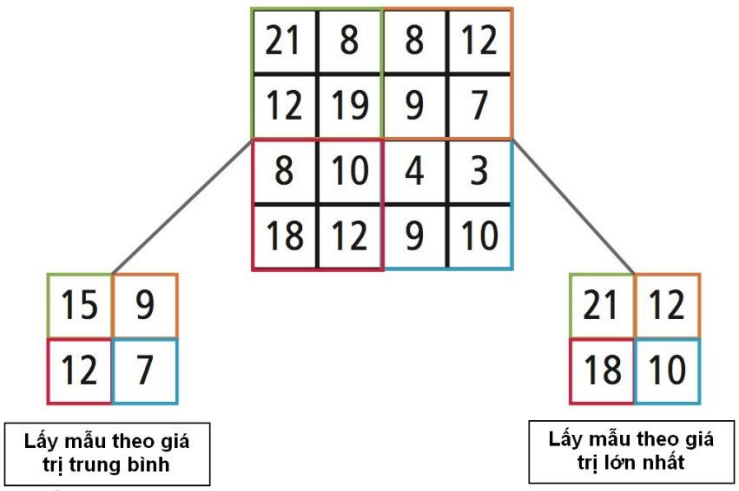
$$f(x) = max(0, x) \quad (2)$$

Thông thường, lớp ReLU được áp dụng ngay phía sau lớp Tích chập, với đầu ra là một ảnh mới có kích thước giống với ảnh đầu vào, các giá trị điểm ảnh cũng hoàn toàn tương tự trừ các giá trị âm đã bị loại bỏ.

- **Lớp lấy mẫu**:

Một thành phần tính toán chính khác trong mạng CNN là lấy mẫu (Pooling), thường được đặt sau lớp Tích chập và lớp ReLU để làm giảm kích thước kích thước ảnh đầu ra trong khi vẫn giữ được các thông tin quan trọng của ảnh đầu vào. Việc giảm kích thước dữ liệu có tác dụng làm giảm được số lượng tham số cũng như tăng hiệu quả tính toán. Lớp lấy mẫu cũng sử dụng một cửa sổ trượt để quét toàn bộ các vùng trong ảnh tương tự như lớp Tích chập, và thực hiện phép lấy mẫu thay vì phép tích chập – tức là ta sẽ chọn lưu lại một giá trị duy nhất đại diện cho toàn bộ thông tin của vùng ảnh đó.

Hình 2.4 thể hiện các phương thức lấy mẫu thường được sử dụng nhất hiện nay, đó là Max Pooling (lấy giá trị điểm ảnh lớn nhất) và Avarage Pooling (lấy giá trị trung bình của các điểm ảnh trong vùng ảnh cục bộ) [14].



Hình 2.4: Phương thức Avarage Pooling và Max Pooling

Như vậy, với mỗi ảnh đầu vào được đưa qua lấy mẫu ta thu được một ảnh đầu ra tương ứng, có kích thước giảm xuống đáng kể nhưng vẫn giữ được các đặc trưng cần thiết cho quá trình tính toán sau này.

- **Lớp kết nối đầy đủ**:

Lớp kết nối đầy đủ này được thiết kế hoàn toàn tương tự như trong mạng nơ-ron truyền thống, tức là tất cả các điểm ảnh được kết nối đầy đủ với node trong lớp tiếp theo. So với mạng nơ-ron truyền thống, các ảnh đầu vào của lớp này đã có kích thước được giảm bớt rất nhiều, đồng thời vẫn đảm bảo các thông tin quan trọng cho việc nhận dạng. Do vậy, việc tính toán nhận dạng sử dụng mô hình truyền thẳng đã không còn phức tạp và tốn nhiều thời gian như trong mạng nơ ron truyền thống.

2.2. Học chuyển giao và tinh chỉnh mô hình huấn luyện

Trong thời gian đầu khi các phương pháp Học sâu mới đạt được nhiều thành tựu và được áp dụng phổ biến, trong cộng đồng Học sâu trên thế giới đã tồn tại một quan niệm không chính xác nhưng hết sức phổ biến: nếu bạn không có lượng dữ liệu huấn luyện khổng lồ, bạn không thể tạo ra một mô hình Học sâu hiệu quả. Nói chính xác hơn, đây đã từng là một quan niệm đúng và hợp lý, bởi mỗi mô hình huấn luyện này đều sử dụng rất nhiều các lớp ẩn, với hàng nghìn nơ-ron và hàng triệu tham số. Đồng thời quá trình huấn luyện mô hình cũng được gắn liền với các kiến thức riêng và bài toán phân tích, nhận dạng… cụ thể, và nếu cố gắng áp dụng mô hình đó với một CSDL khác, chắc chắn độ chính xác sẽ bị suy giảm đáng kể. Tuy nhiên, trong thời gian sau đó, một phương pháp học mới được đưa ra và đã giải quyết được điểm hạn chế này của Học sâu, đó chính là Học chuyển giao – Transfer Learning [15].

Học chuyển giao là quá trình khai thác, tái sử dụng các tri thức đã được học tập bởi một mô hình huấn luyện trước đó vào giải quyết một bài toán mới mà không phải xây dựng một mô hình huấn luyện khác từ đầu. Đây được coi là một trong những kỹ thuật được xếp mức độ quan trọng hàng đầu trong cộng đồng khoa học dữ liệu, nhằm hướng tới mục đích chung là phát minh ra một thuật toán học tự động mạnh mẽ.

Hiện nay, phương pháp phổ biến thường được áp dụng khi huấn luyện mô hình với một bộ CSDL tương đối nhỏ là sử dụng Học chuyển giao để tận dụng một mạng CNN đã được huấn luyện trước đó với bộ dữ liệu rất lớn như ImageNet (1,2 triệu ảnh với 1.000 nhãn đánh dấu). Phương pháp này sử dụng mạng CNN theo hai cách chính như sau:

- Mạng CNN này sẽ chỉ được sử dụng như một bộ trích chọn đặc trưng cho bộ CSDL huấn luyện mới, bằng cách thay thế các lớp Fully-connected ở cuối mạng và giữ cố định các tham số cho toàn bộ các lớp còn lại của mạng.
- Không chỉ thay thế và huấn luyện lại bộ nhận dạng cuối cùng của mạng CNN, mà đồng thời ta thực hiện tối ưu, tinh chỉnh (Fine-tune) một vài hoặc tất cả các lớp trong mạng.

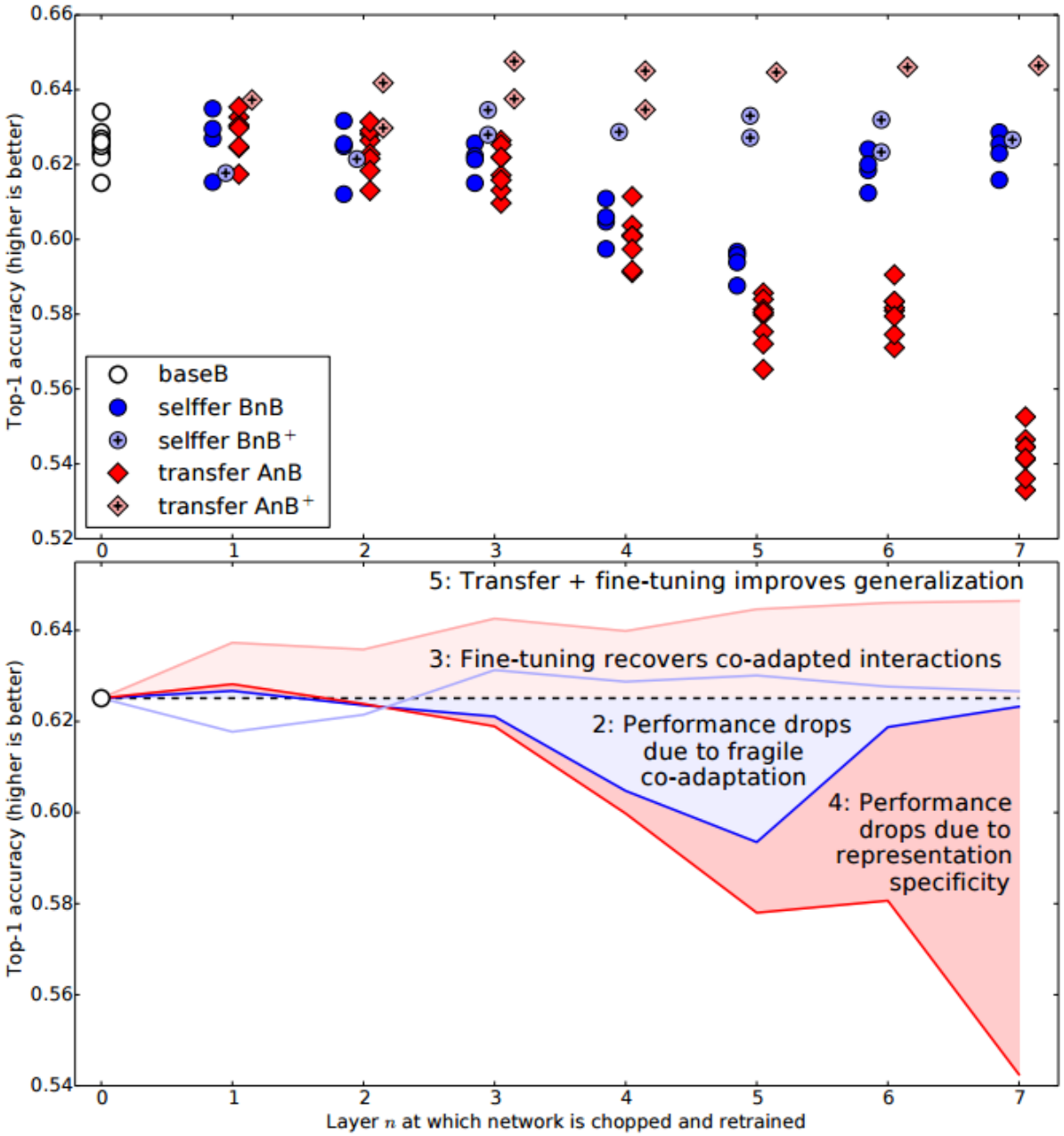
Ý tưởng của việc tái sử dụng mạng CNN là dựa trên nhận định rằng các đặc trưng được học trong các lớp đầu của mạng là các đặc trưng chung nhất, hữu dụng với phần lớn bài toán, ví dụ: đặc trưng về cạnh, hình khối hay các khối màu… Các lớp sau đó của mạng CNN sẽ nâng dần độ cụ thể, riêng biệt của các chi tiết phục vụ cho bài toán nhận dạng cần giải quyết. Do đó, ta hoàn toàn có thể tái sử dụng lại các lớp đầu của mạng CNN mà không phải mất nhiều thời gian và công sức huấn luyện từ đầu.

Có khá nhiều bài báo, công trình khoa học được đưa ra để chứng minh cho khả năng chuyển giao của những đặc trưng trong mạng Học sâu [16]. Cụ thể, để tìm ra mức độ "chung" của các đặc trưng theo từng lớp của mạng AlexNet, các tác giả của bài báo đã thực hiện một phương pháp so sánh tốn nhiều thời gian và công sức để thu được kết quả cụ thể, rõ ràng:

1) Chia đôi bộ dữ liệu của ImageNet, mỗi nhóm có khoảng 645.000 ảnh.
2) Huấn luyện lại mạng AlexNet trên từng nhóm để được 2 mạng cơ sở, gọi là mạng baseA và baseB.
3) Copy lần lượt n lớp đầu tiên (n = 1, 2...7) của từng mạng baseA, baseB, đồng thời cố định hoặc cho phép tinh chỉnh các tham số của các lớp này để được các mạng huấn luyện khác nhau (AnB, AnB+).
4) Thực hiện huấn luyện trên từng mạng và so sánh kết quả để thể hiện khả năng chuyển giao của các đặc trưng qua từng lớp của mạng AlexNet.

Từ kết quả thực nghiệm trong hình dưới, kết luận quan trọng được rút ra: sự chuyển giao các đặc trưng có thể cải thiện hiệu năng của mô hình, tuy nhiên chất lượng

chuyển giao này chịu ảnh hưởng bởi hai yếu tố chính là sự thích nghi lẫn nhau dễ bị phá vỡ tại các lớp nằm ở giữa mạng và sự riêng biệt hóa tại các lớp cấp cao của mạng.



Hình 2.5: Kết quả thực nghiệm theo số lượng lớp mạng CNN được chuyển giao [16]

Một bài báo khoa học khác cũng đã chứng minh được hiệu quả của Học chuyển giao khi giải quyết một bài toán mới bằng cách tinh chỉnh một mô hình CNN đã được huấn luyện trước đó với bộ cơ sở dữ liệu ảnh ImageNet. Bài toán được đưa ra là nhận dạng 102 loại hoa khác nhau sử dụng bộ dữ liệu ảnh hoa Oxford có kích thước nhỏ (~6.000 ảnh huấn luyện và ~1.000 ảnh test), nhóm nghiên cứu đã tùy chỉnh các lớp Fully-connected của mạng AlexNet để số lượng đầu ra là 102, tương ứng với 102 loại hoa cần nhận dạng [17]. Bằng cách giảm tỉ lệ học toàn cục và tăng tỉ lệ học cục bộ tại các lớp Fully-connected so với các lớp khác, mạng AlexNet (được trình bày trong mục

2.3) đã được tinh chỉnh thành công với độ chính xác cao: tỉ lệ lỗi chỉ còn 7% trên bộ test 1.000 ảnh.

```
I0215 15:28:06.417726  6585 solver.cpp:246] Iteration 50000, loss = 0.000120038
I0215 15:28:06.417789  6585 solver.cpp:264] Iteration 50000, Testing net (#0)
I0215 15:28:30.834987  6585 solver.cpp:315]     Test net output #0: accuracy = 0.9326
I0215 15:28:30.835072  6585 solver.cpp:251] Optimization Done.
I0215 15:28:30.835083  6585 caffe.cpp:121] Optimization Done.
```

Hình 2.6: Kết quả huấn luyện sau khi tinh chỉnh mạng AlexNet [17]

2.3. Mạng huấn luyện AlexNet

Mạng huấn luyện AlexNet là công trình đầu tiên phổ biến mạng CNN trong lĩnh vực Thị giác máy tính, cũng là một trong những mạng huấn luyện CNN nổi tiếng nhất nhờ thành tích ấn tượng mà nó đạt được trong cuộc thi nhận dạng ảnh quy mô lớn tổ chức vào năm 2012. Cuộc thi này có tên chính thức là ILSVRC – ImageNet Large Scale Visual Recognition Challenge [18], được ImageNet - một hãng CSDL ảnh - tổ chức thường niên và được coi là cuộc thi Olympics quy mô thế giới trong lĩnh vực Thị giác máy tính. Mục đích của cuộc thi là nhằm thử nghiệm các công nghệ mới giúp cho máy tính có thể hiểu, phân tích, phát hiện và nhận dạng các vật thể trong một bức ảnh.

Cụ thể hơn, nhiệm vụ chính của cuộc thi năm 2012 đặt ra mà các đội tham gia phải giải quyết là bài toán nhận dạng, với bộ dữ liệu huấn luyện lên đến 1,2 triệu ảnh được gán nhãn cho 1.000 hạng mục khác nhau. Nhóm SuperVision, gồm các thành viên Alex Krizhevsky, Ilya Sutskever và Geoff Hinton, cùng với mạng AlexNet của họ đã đạt được kết quả đáng kinh ngạc là chiến thắng áp đảo nhóm đứng thứ hai với độ chính xác chênh lệch đến hơn 10% (15,31% và 26,17%) [19]. Điều đặc biệt là mạng huấn luyện này chỉ nhận dữ liệu đầu vào là các giá trị điểm ảnh thô và không hề áp dụng bất kỳ phương pháp trích chọn đặc trưng nào, trong khi mọi hệ thống nhận dạng thị giác truyền thống đều phải gồm nhiều giai đoạn trích chọn đặc trưng hết sức tỉ mỉ, cẩn thận, thậm chí phải áp dụng nhiều mẹo để cải thiện chất lượng nhận dạng. Thiết kế kiến trúc mạng huấn luyện gần như một hộp đen, cộng với khả năng tự học các đặc trưng thông qua các lớp ẩn, đã khiến CNN nói riêng và Học sâu nói chung trở thành giải pháp mạnh mẽ nhất cho bài toán nhận dạng và phân loại vật thể cho tới bây giờ.

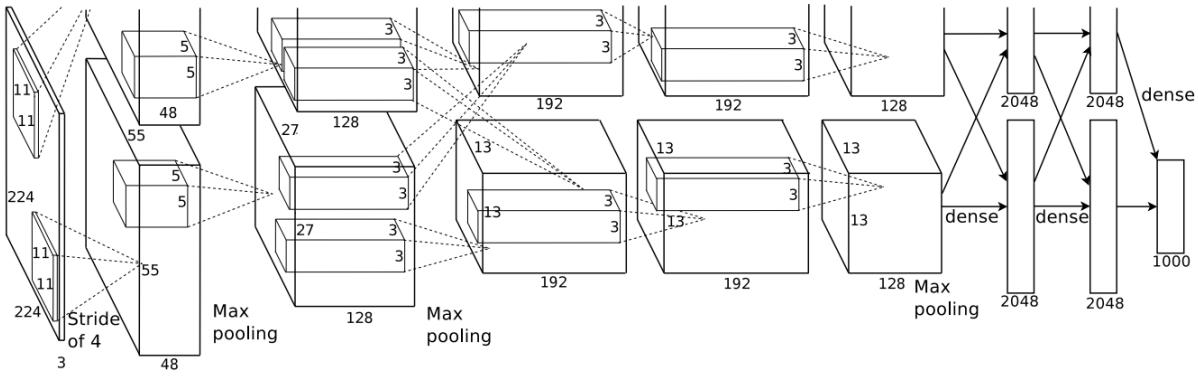
Từ năm 2012, mạng CNN trở thành cái tên gắn liền với cuộc thi và đã có rất nhiều mạng CNN nổi bật khác xuất hiện trong những năm sau đó VGG, GoogleNet hay Microsoft ResNet… Các mạng CNN càng ngày càng đạt độ chính xác cao hơn, tuy nhiên chúng có độ phức tạp và độ sâu lớn hơn rất nhiều, ví dụ mạng CNN có thể coi là tốt nhất hiện nay – ResNet – đã sử dụng đến 152 lớp tính toán. Sự phức tạp này yêu cầu khả năng tính toán lớn, thời gian huấn luyện lâu, và gây nhiều khó khăn trong việc cài đặt triển khai hệ thống, do đó mạng AlexNet đã được chọn làm cơ sở phát triển phiên bản

thử nghiệm ban đầu và việc cài đặt các mạng huấn luyện khác nhằm nâng cao chất lượng nhận dạng của hệ thống sẽ được thử nghiệm trong tương lai.

Trong phần tiếp theo ta sẽ tìm hiểu kỹ hơn về kiến trúc tổng thể của mạng AlexNet cũng như cách thức ứng dụng nó vào bài toán nhận dạng hoa quả sử dụng phương pháp Học chuyển giao.

2.3.1. Kiến trúc mạng AlexNet

Nhóm của Alex Krizhevsky đã công bố một bài báo với tiêu đề "ImageNet Classification with Deep Convolutional Networks" [20], đưa ra mô tả cụ thể về kiến trúc của mạng AlexNet cũng như cách thức cài đặt và sử dụng các lớp trong mạng để huấn luyện mô hình với bộ dữ liệu ảnh của ImageNet. Mạng có cấu trúc tương đối đơn giản nếu so với các mạng CNN hiện đại gần đây, bao gồm 5 lớp Tích chập và 3 lớp kết nối đầy đủ với các lớp giữa là các lớp lấy mẫu và ReLU, được huấn luyện song song trên hai card đồ họa GPU.



Hình 2.7: Kiến trúc mạng AlexNet [20]

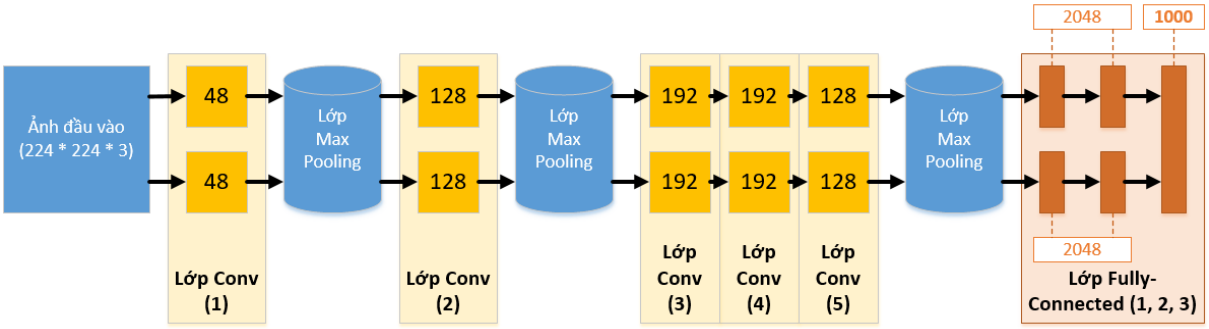
Hình 2.7 thể hiện rõ kiến trúc tổng thể của mạng AlexNet, trong đó:

- **Lớp 1** (Tích chập):
  - Đầu vào: Ảnh với kích thước 224 x 224 x 3 (3 là con số tương ứng với 3 màu đỏ, xanh lục, xanh lam trong hệ màu RGB thông thường)
  - Số bộ lọc: 96
  - Kích thước bộ lọc: 11 x 11 x 3
  - Bước trượt (Stride): 4
  - Đầu ra: (224/4) x (224/4) x 96 = 55 x 55 x 96, chia đều cho hai GPU
- Lớp chuyển tiếp sang lớp 2 (Lấy mẫu tối đa):
  - Đầu vào: 55 x 55 x 96
  - Đầu ra: (55/2) x (55/2) x 96 = 27 x 27 x 96
- **Lớp 2** (Tích chập):
  - Đầu vào: 27 x 27 x 96
  - Số bộ lọc: 256

- Kích thước bộ lọc: 5 x 5 x 48
- Đầu ra: 27 x 27 x 256, chia đều cho hai GPU

- **Lớp 3, 4, 5**: Tương tự như với lớp 1 và lớp 2 với các kích thước bộ lọc lần lượt là 3 x 3 x 256, 3 x 3 x 384 và 3 x 3 x 384. Toàn bộ các lớp tính toán này đều được chia đều cho hai GPU để tăng tốc độ xử lý. Đầu ra cuối cùng qua lớp Tích chập thứ 5 là dữ liệu với kích thước 13 x 13 x 128, dữ liệu này sau khi đi qua một lớp Lấy mẫu tối đa cuối cùng sẽ được dùng làm đầu vào cho các lớp sau đó là các lớp Kết nối đầy đủ.
- **Lớp 6** (Kết nối đầy đủ):
  - Đầu vào: 6 x 6 x 256
  - Số nơ-ron: 4096
- **Lớp 7** (Kết nối đầy đủ): Tương tự lớp 6.
- Lớp 8 (Kết nối đầy đủ): Lớp cuối cùng trong mạng AlexNet này có 1000 nơ-ron, tương ứng với 1000 lớp khác nhau mà bộ huấn luyện cần nhận dạng.

Ta có thể nhìn rõ hơn kiến trúc mạng AlexNet ở dạng phẳng như trong Hình 2.8:



Hình 2.8: Kiến trúc mạng AlexNet ở dạng phẳng

2.3.2. Ứng dụng mạng AlexNet vào bài toán Nhận dạng, phân loại hoa quả

Từ kết luận rút ra trong phần 2.2 về hiệu quả của Học chuyển giao với các mô hình CNN trong việc giải quyết trường hợp bài toán mới với kích thước bộ cơ sở dữ liệu tương đối nhỏ, luận văn đề xuất phương hướng giải quyết bài toán nhận dạng hoa quả như sau:

1) Cài đặt mạng AlexNet với một mô hình đã được huấn luyện trước với bộ ảnh của ImageNet.
2) Xây dựng bộ CSDL ảnh huấn luyện cho 40 loại hoa quả với ảnh được chọn lựa theo tiêu chuẩn về kích thước, màu sắc cũng như độ rõ nét, đồng thời được gán nhãn cẩn thận.
3) Tinh chỉnh lại mô hình để giải quyết bài toán nhận dạng 40 loại hoa quả. Dựa theo kết luận được chứng minh bởi các bài báo khoa học đã trình bày trong phần trước, dù kích thước CSDL ảnh không quá lớn độ chính xác của mô hình

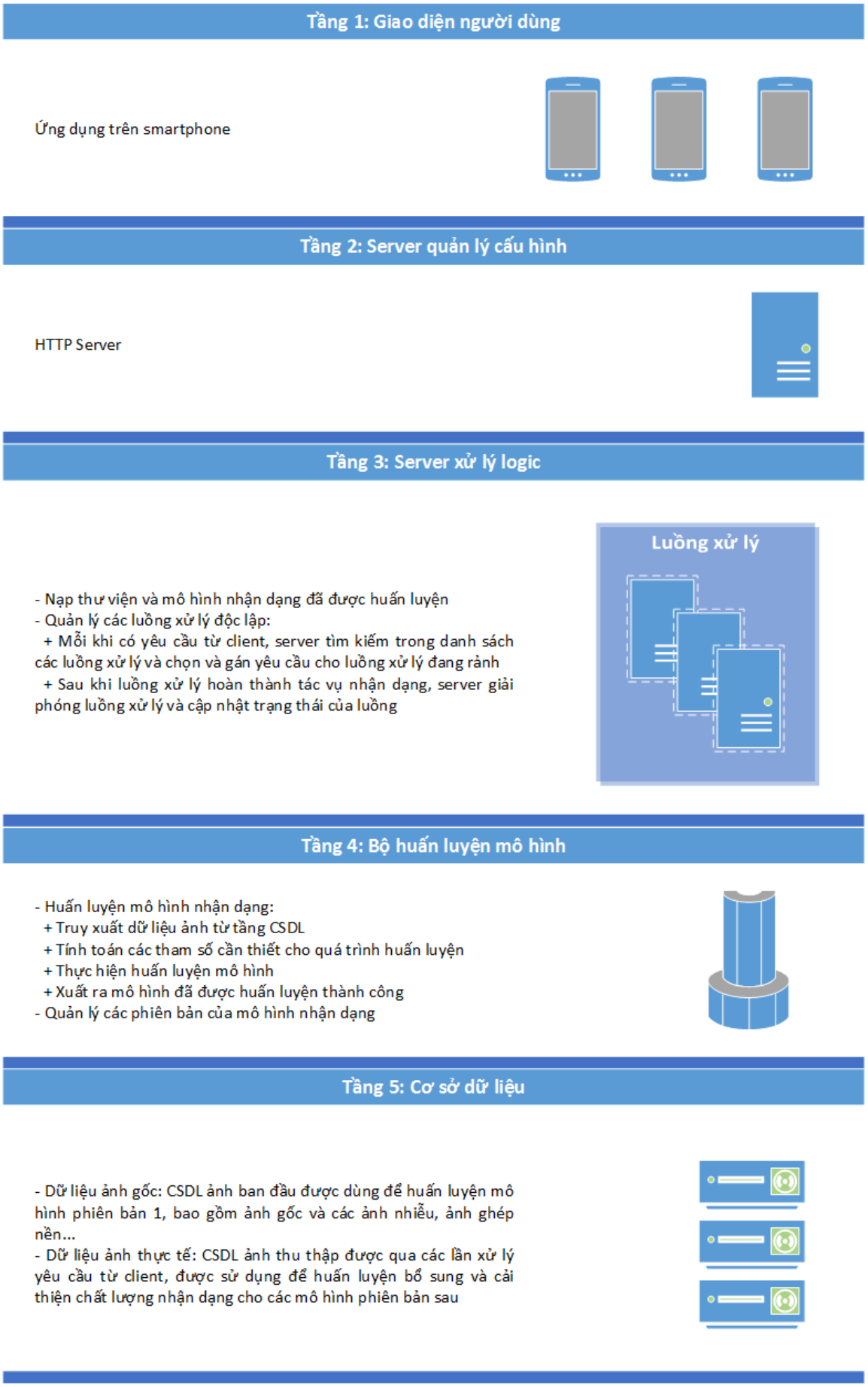
nhận dạng vẫn được đảm bảo nhờ khả năng trích chọn đặc trưng tự động của mạng AlexNet.

Chương 3. Hệ thống phần mềm nhận dạng hoa quả

3.1. Tổng quan hệ thống

Hệ thống phần mềm Nhận dạng hoa quả – Fruit Recognition System – được thiết kế theo kiến trúc Client/Server năm tầng (xem Hình 3.1), trong đó:

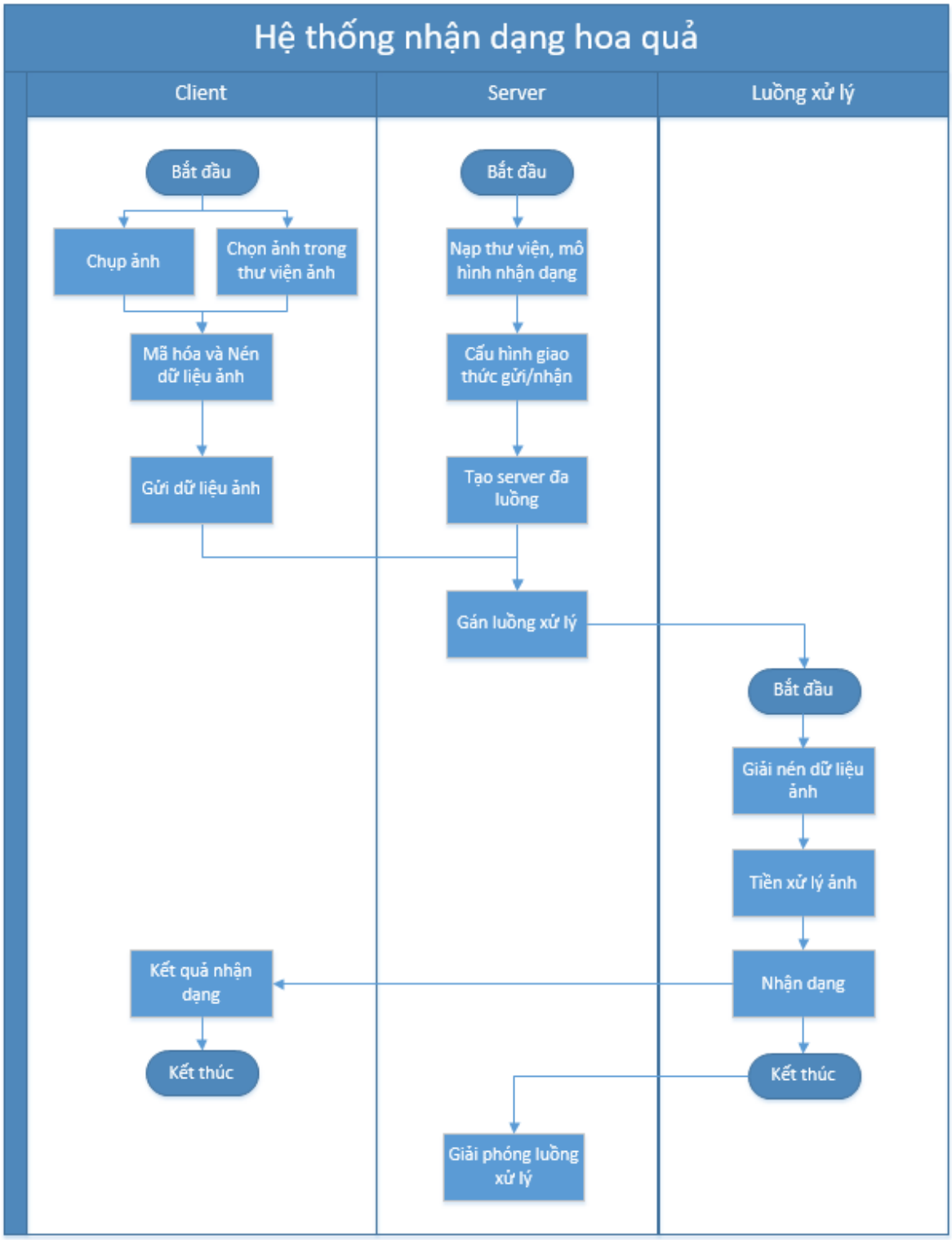
- **Tầng thứ nhất** là tầng giao diện người dùng, cụ thể là ứng dụng client trên điện thoại thông minh, quản lý tương tác người dùng với ứng dụng như chụp ảnh, chọn ảnh gửi lên server… và hiển thị kết quả nhận dạng do server gửi về.
- **Tầng thứ hai** là tầng server quản lý cấu hình hệ thống, ví dụ cấu hình giao thức gửi/nhận dữ liệu với client, cụ thể giao thức được sử dụng trong hệ thống là giao thức HTTP.
- **Tầng thứ ba** là tầng server thực hiện logic xử lý các yêu cầu từ client, như quản lý và phân phối các luồng xử lý độc lập, đảm bảo hiệu năng và chất lượng tính toán nhận dạng cho nhiều client trong cùng một thời điểm.
- **Tầng thứ tư** là tầng đảm nhiệm xây dựng, tinh chỉnh và quản lý các phiên bản mô hình nhận dạng cho hệ thống, với bộ ảnh huấn luyện được lấy từ tầng quản lý dữ liệu bên dưới.
- **Tầng cuối cùng** là tầng quản lý dữ liệu, bao gồm CSDL ảnh phục vụ cho việc huấn luyện mô hình, CSDL ảnh đã xử lý từ các client nhằm mục đích bổ sung sự đa dạng của CSDL ảnh và cải thiện độ chính xác của mô hình nhận dạng. Các bộ ảnh trên được lưu tách biệt để thuận tiện cho việc quản lý và đánh giá độ chính xác của các phiên bản mô hình huấn luyện cũng như mức độ ảnh hưởng của bộ ảnh huấn luyện lên chất lượng mô hình.



Hình 3.1: Kiến trúc Client-Server n tầng

Luồng hoạt động chính của hệ thống được thể hiện trong Hình 3.2, trong đó các bước thực hiện của server và client từ lúc khởi động ban đầu tới lúc kết thúc như sau:

- Client (ứng dụng trên điện thoại thông minh):
    1) Người dùng khởi động ứng dụng.
    2) Người dùng thực hiện chụp ảnh hoa quả bằng camera của điện thoại, hoặc chọn ảnh đã chụp trước đó được lưu trong Thư viện ảnh.
    3) Ảnh chụp được mã hóa, nén lại và gửi tới máy chủ.
    4) Ứng dụng đợi nhận kết quả nhận dạng từ máy chủ gửi về và hiển thị cho người dùng.
- Chương trình Server:
    1) Chương trình được khởi động và nạp các thư viện cần thiết.
    2) Chương trình nạp mô hình nhận dạng đã được huấn luyện trước đó.
    3) Giao thức gửi, nhận dữ liệu giữa ứng dụng phía client và chương trình server được cấu hình.
    4) Một loại các luồng xử lý được khởi tạo, đặt trạng thái ban đầu là trạng thái rỗi.
    5) Khi có ứng dụng client kết nối tới, chương trình kiểm tra trong danh sách các luồng xử lý và chọn một luồng đang ở trạng thái rỗi để nhận và tính toán dữ liệu do client gửi tới.
    6) Trong luồng xử lý:
        - Bắt đầu quá trình tính toán nhận dạng, cờ trạng thái là “bận”.
        - Thực hiện giải nén dữ liệu thành dữ liệu ảnh gốc.
        - Sử dụng mô hình đã nạp để nhận dạng loại hoa quả.
        - Trả kết quả nhận dạng về cho ứng dụng client.
        - Kết thúc quá trình tính toán.
    7) Khi luồng xử lý đã hoàn thành quá trình tính toán nhận dạng, chương trình giải phóng luồng xử lý bằng cách cập nhật lại trạng thái hiện tại của luồng.



Hình 3.2: Luồng hoạt động chính của hệ thống

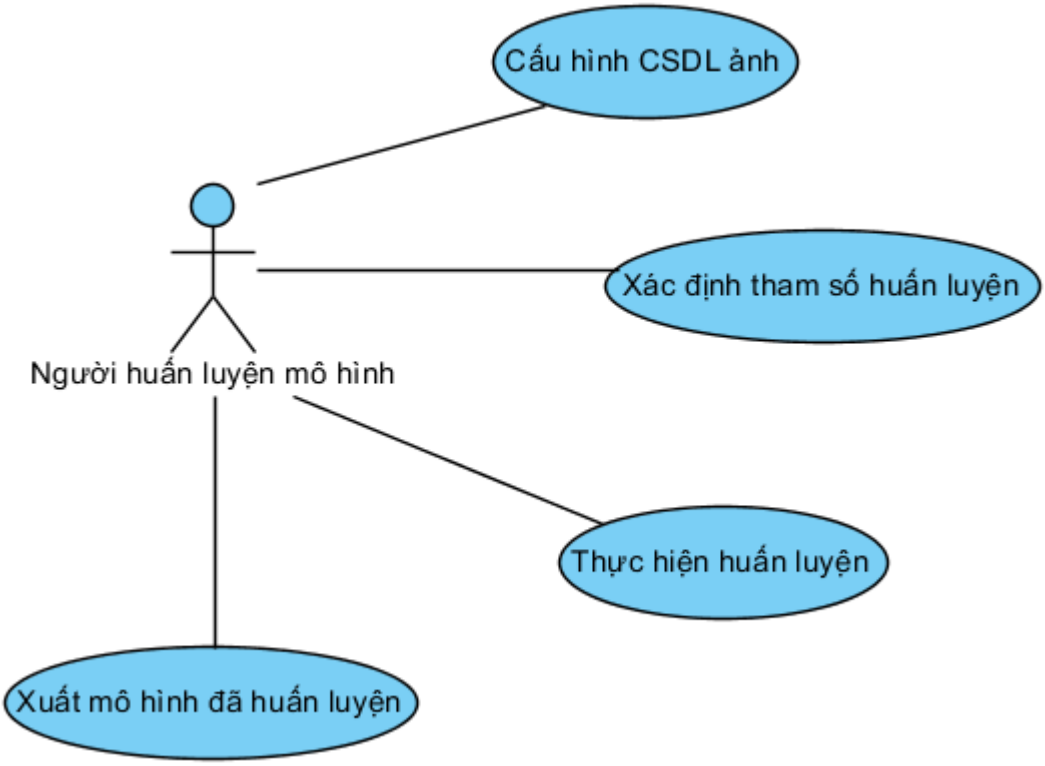
3.2. Mô đun quản lý cơ sở dữ liệu

Bộ CSDL ảnh phục vụ cho huấn luyện và tinh chỉnh các mô hình nhận dạng trong các thuật toán Học sâu nói riêng và Học máy nói chung là thành phần vô cùng quan trọng, quyết định chủ yếu đến độ chính xác mà mô hình đạt được. Do vậy, chúng cần được lưu trữ và quản lý một cách khoa học. Trong hệ thống lưu trữ, bộ CSDL ảnh huấn luyện được chia thành các thư mục riêng biệt:

- **Thư mục ảnh gốc**: Là bộ ảnh ban đầu được sử dụng để xây dựng phiên bản mô hình nhận dạng đầu tiên, gồm các thư mục con là ảnh gốc, ảnh lọc nền, ảnh chiếu nghiêng, ảnh thêm nhiễu và ảnh ghép nền.
- **Thư mục ảnh thực tế chưa duyệt**: Là các ảnh chụp thực tế bởi ứng dụng, do điện thoại thông minh của người dùng gửi lên để thực hiện nhận dạng, được chia thành các thư mục con tương ứng với 40 loại hoa quả được huấn luyện. Các ảnh này chưa được kiểm duyệt và chưa được sử dụng để tăng cường cho CSDL ảnh huấn luyện.
- **Thư mục ảnh thực tế đã duyệt**: Bao gồm các ảnh thực tế đã được kiểm duyệt đảm bảo chất lượng tốt và loại hoa quả trong ảnh là hợp lệ, những ảnh này đã được gán nhãn đúng, chuyển tới các thư mục con tương ứng và đã được sử dụng để huấn luyện tăng cường cho mô hình nhận dạng ban đầu. Các thư mục ảnh này đều được đặt trong các thư mục cha được đánh số ứng với phiên bản mô hình được huấn luyện bổ sung, nhằm đảm bảo không có sự nhầm lẫn giữa các phiên bản với nhau.

3.3. Bộ huấn luyện mô hình

Nằm ở tầng thứ tư trong kiến trúc n tầng của hệ thống, bộ huấn luyện mô hình là thành phần có vai trò quan trọng hàng đầu, chịu toàn bộ trách nhiệm về các mô hình nhận dạng từ giai đoạn khởi tạo đến tinh chỉnh và hoàn thiện, cũng như quản lý và đánh giá độ chính xác các phiên bản khác nhau của mô hình. Bộ huấn luyện được cài đặt và triển khai thành một mô đun hoàn toàn tách biệt với các thành phần còn lại của server, giúp cho việc nâng cấp hay thay thế có thể thực hiện độc lập mà không gây ảnh hưởng đến hoạt động thông thường của server. Các ca sử dụng chính của mô đun bao gồm: Cấu hình CSDL ảnh để huấn luyện, Xác định các tham số cho mô hình huấn luyện, Thực hiện huấn luyện và Xuất ra mô hình đã huấn luyện xong theo phiên bản tương ứng với bộ CSDL ảnh đã sử dụng (xem Hình 3.3).



Hình 3.3: Biểu đồ ca sử dụng của Bộ huấn luyện mô hình

Đặc tả biểu đồ ca sử dụng:

Cấu hình CSDL ảnh:

- Mục đích: Cấu hình các thông tin cơ bản về CSDL ảnh cho bộ huấn luyện mô hình, như: đường dẫn thư mục lưu ảnh, phiên bản huấn luyện hiện tại, số lượng ảnh huấn luyện và ảnh test…
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống, hoặc người quản trị mô đun huấn luyện mô hình nhận dạng.
    - Mô tả chung: Người quản trị khi muốn bắt đầu huấn luyện mới, hoặc huấn luyện bổ sung cho mô hình nhận dạng thì trước hết cần cấu hình thông tin bộ CSDL ảnh phục vụ cho huấn luyện.
- Luồng sự kiện chính: Người quản trị cập nhật thông tin về bộ CSDL ảnh trong file cấu hình cho bộ huấn luyện, tạo mới các file ghi lại đường dẫn đến ảnh huấn luyện, ảnh test và nhãn đánh dấu tương ứng.
- Luồng thay thế: Không.
- Các yêu cầu cụ thể: Thông tin bộ CSDL ảnh phải chính xác, đường dẫn đến vị trí ảnh huấn luyện và ảnh test phải hợp lệ.
- Điều kiện trước: Bộ CSDL ảnh huấn luyện phải có sẵn trong hệ thống lưu trữ, các ảnh đã được duyệt và đặt đúng thư mục tương ứng.
- Điều kiện sau: Không.

Tính toán tham số huấn luyện:

- Mục đích: Tính toán các thông số cần thiết từ bộ CSDL ảnh và xác định các tham số nhằm định nghĩa mô hình và cách thức huấn luyện mô hình.
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống, hoặc người quản trị mô đun huấn luyện mô hình nhận dạng.
    - Mô tả chung: Người quản trị khi muốn bắt đầu thực hành huấn luyện mới, hoặc huấn luyện bổ sung cho mô hình nhận dạng thì cần phải xác định các tham số định nghĩa quá trình huấn luyện cũng như các giá trị cần thiết liên quan đến bộ CSDL ảnh.
- Luồng sự kiện chính: Người quản trị gọi file thực thi các hàm tính toán giá trị liên quan đến bộ CSDL ảnh đầu vào, sửa đổi cập nhật tham số trong các file định nghĩa huấn luyện mô hình.
- Luồng thay thế: File thực thi tính toán thông báo lỗi khi không thể tính toán thành công trên bộ CSDL ảnh đã cấu hình.
- Các yêu cầu cụ thể: Đầu ra của file thực thi tính toán phải là các file dữ liệu theo định dạng chuẩn, các tham số định nghĩa mô hình phải phù hợp với mục đích huấn luyện.
- Điều kiện trước: Các thông tin liên quan đến bộ CSDL ảnh phải được cấu hình hợp lệ trước đó.
- Điều kiện sau: Thông báo tính toán thành công giá trị cần thiết từ bộ CSDL ảnh.

Thực hiện huấn luyện:

- Mục đích: Huấn luyện, tinh chỉnh mô hình nhận dạng cho hệ thống sử dụng bộ CSDL ảnh trên nền một mô hình đã huấn luyện trước. Ảnh được sử dụng để huấn luyện có thể là các ảnh ban đầu, gồm ảnh gốc và ảnh sinh tự động, hoặc là các ảnh được thu thập, lưu trữ trong quá trình người dùng gửi yêu cầu nhận dạng lên server.
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống, hoặc người quản trị mô đun huấn luyện mô hình nhận dạng.
    - Mô tả chung: Người quản trị khi đã hoàn thành việc thu thập ảnh, cấu hình các thông tin liên quan đến CSDL ảnh cũng như tính toán, xác định các tham số cần thiết thì có thể bắt đầu thực hiện huấn luyện mô hình nhận dạng cho hệ thống.
- Luồng sự kiện chính: Người quản trị gọi file thực thi các câu lệnh cần thiết để bắt đầu huấn luyện mô hình. Các câu lệnh được chia thành hai loại: Câu lệnh bắt đầu một phiên huấn luyện mới và Câu lệnh tiếp tục phiên huấn luyện bị tạm dừng trước đó.

- Luồng thay thế: File thực thi thông báo lỗi khi không thể thực hiện huấn luyện với các tham số đầu vào đã cấu hình, gồm tham số về file định nghĩa mô hình, mô hình được huấn luyện trước, lựa chọn sử dụng card đồ họa GPU, hoặc file trạng thái huấn luyện tại thời điểm tạm dừng (trong trường hợp tiếp tục phiên huấn luyện chưa hoàn thành)…
- Các yêu cầu cụ thể: Đầu ra của quá trình huấn luyện là một mô hình nhận dạng và các file ghi lại nhật ký huấn luyện, gồm các thông tin, cảnh báo hoặc lỗi xảy ra trong quá trình huấn luyện để người quản trị có thể truy vết nếu cần thiết. Ngoài ra, thông tin về phiên bản của mô hình nhận dạng được huấn luyện cũng được lưu lại.
- Điều kiện trước: Thông tin cấu hình CSDL ảnh và tham số định nghĩa mô hình huấn luyện phải chính xác. Mô hình được huấn luyện trước và file trạng thái huấn luyện tại thời điểm tạm dừng phải hợp lệ.
- Điều kiện sau: Thông báo huấn luyện thành công mô hình, với một số thông tin cơ bản của phiên huấn luyện như phiên bản hiện tại của mô hình và độ chính xác đạt được.

Xuất mô hình đã huấn luyện:

- Mục đích: Xuất ra mô hình đã được huấn luyện thành công, làm đầu vào cho mô đun tính toán nhận dạng của server.
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống, hoặc người quản trị mô đun huấn luyện mô hình nhận dạng.
    - Mô tả chung: Sau khi người quản trị đã hoàn thành việc huấn luyện mô hình, để mô hình mới có thể được sử dụng vào quá trình tính toán nhận dạng thực tế người quản trị phải xuất mô hình ra và thay thế cho mô hình cũ.
- Luồng sự kiện chính: Người quản trị gọi file thực thi câu lệnh xuất mô hình đã huấn luyện ra thư mục lưu trữ (đã cài đặt trong file cấu hình chung của hệ thống). Mô hình cũ đang được sử dụng được chuyển sang thư mục lưu các phiên bản không còn sử dụng.
- Luồng thay thế: File thực thi thông báo lỗi trong quá trình xuất mô hình đã huấn luyện và chuyển mô hình phiên bản cũ sang thư mục để lưu trữ.
- Các yêu cầu cụ thể: Không.
- Điều kiện trước: Mô hình đã được huấn luyện phải là mô hình hoàn thiện. Các thông tin cấu hình về thư mục mô hình hiện tại và thư mục lưu trữ các mô hình với phiên bản thấp hơn phải hợp lệ.
- Điều kiện sau: Thông báo xuất mô hình thành công.

Về thành phần cấu tạo của mô đun Bộ huấn luyện mô hình, ta có hai thành phần chính: chương trình huấn luyện (sử dụng phương pháp Học sâu, cụ thể là một mạng nơ-

ron tích chập CNN, và framework Caffe trên Windows) và thành phần quản lý phiên bản mô hình nhận dạng. Ta sẽ đi vào mô tả chi tiết các thành phần này trong các mục tiếp theo.

3.3.1. Môi trường huấn luyện

Môi trường được sử dụng để huấn luyện mô hình nhận dạng hoa quả là Windows 10, ngôn ngữ Python phiên bản 2.7.12 với framework chuyên dùng cho Học sâu là Caffe.

Caffe [21] là một framework mã nguồn mở cho Học sâu, phát triển với Berkeley Vision and Learning Center (BVLC), được viết bởi ngôn ngữ C++, CUDA C++ cùng với các bộ gói wrapper cho các ngôn ngữ khác như Python hay Matlab. Điểm mạnh của framework này là cho phép người dùng tùy chọn huấn luyện thuật toán Học sâu trên CPU hay trên card đồ họa GPU, dễ dàng thực hiện quá trình huấn luyện trên bộ dữ liệu ảnh cá nhân chỉ với các câu lệnh đơn giản. Bên cạnh đó, Caffe cũng cho phép người dùng tái sử dụng lại các mô hình đã được huấn luyện sẵn và được chia sẻ bởi cộng đồng nghiên cứu trên khắp thế giới.

Hình 3.4: Các framework Học sâu nổi tiếng trên thế giới

Trong các framework Học sâu phổ biến nhất, ngoài Caffe người dùng còn có các lựa chọn khác như Theano, Torch7 hoặc TensorFlow... (xem Hình 3.4). Mỗi framework đều hỗ trợ rất mạnh mẽ trong việc huấn luyện các mô hình nhận dạng trong Học sâu, cũng như có các điểm mạnh riêng phù hợp với các mục đích sử dụng khác nhau. Theano [22] là một trong các framework Học sâu ra đời sớm nhất, là giải pháp tốt cho những người dùng muốn lập trình lại toàn bộ thuật toán, hoặc tinh chỉnh riêng một vài thành phần tối ưu tính toán để giải quyết cho các vấn đề riêng biệt. Theano đặc biệt phù hợp với các bài toán hoặc các hệ thống không có sự cài đặt, triển khai mạng huấn luyện theo một tiêu chuẩn cụ thể nào. Với framework Torch7 [23], đây cũng là một framework ở mức cấp thấp (low-level) gần giống với Theano nhưng có cung cấp thêm một số thuật toán và logic cơ bản giúp người dùng giảm bớt việc lập trình toàn bộ thuật toán từ đầu. Tuy có khá nhiều dự án mã nguồn mở sử dụng Torch7, đặc biệt là các dự án từ Facebook

AI, và ngôn ngữ Lua của Torch7 cũng là ngôn ngữ lập trình phổ biến nhưng người dùng gặp khá nhiều khó khăn trong việc tìm các tài liệu hướng dẫn hay tham chiếu. Đây cũng là điểm hạn chế lớn nhất của framework này. Một framework Học sâu phổ biến khác được tạo ra bởi Google để thay thế cho Theano là TensorFlow (TF) [24], TF không phải mã nguồn mở hoàn toàn, cũng như không đơn thuần chỉ phục vụ cho Học sâu mà còn hỗ trợ các công cụ tính toán cho học tăng cường (Reinforcement Learning) và khá nhiều thuật toán khác. Trong nhiều bài báo về đánh giá hiệu năng của các framework Học sâu phổ biến, TF thường đạt kết quả không cao trong phần lớn các bài test. Tuy vậy, TF có ưu điểm mạnh về số lượng công cụ bổ trợ, đặc biệt là cho việc gỡ lỗi (debug), và sự đảm bảo hỗ trợ liên tục từ Google.

Sau quá trình tìm hiểu và so sánh các framework phổ biến, tôi đã quyết định chọn Caffe làm công cụ cài đặt triển khai ứng dụng bởi một số ưu điểm nổi trội của nó đối với bài toán nhận dạng ảnh: Caffe là framework rất mạnh về xử lý ảnh, cho phép người dùng dễ dàng tinh chỉnh mô hình mạng đã được huấn luyện trước cũng như thực hiện các bước huấn luyện mà không cần quá trình lập trình phức tạp.

3.3.2. Cấu hình mạng huấn luyện AlexNet

Các mô hình huấn luyện và các phép tinh chỉnh của mạng AlexNet nói riêng và mạng CNN nói chung đều được framework Caffe thể hiện bằng cấu trúc văn bản thuần, nhằm tạo ra sự minh bạch rõ ràng khi định nghĩa các phép biến đổi ảnh hay các lớp trong mô hình và sự dễ dàng khi triển khai hoặc chuyển giao một mô hình huấn luyện. Ta có thể tham khảo một vài ví dụ về cách thức định nghĩa một lớp trong mạng huấn luyện CNN như lớp dữ liệu đầu vào hay lớp tích chập Tích chập như trong Hình 3.5:

```
layer {
  name: "data"
  type: "ImageData"
  top: "data"
  top: "label"
  include {
    phase: TRAIN
  }
  transform_param {
    mirror: true
    crop_size: 227
    mean_value: 104
    mean_value: 117
    mean_value: 123
  }
  image_data_param {
    source: "/path/to/file/train.txt"
    batch_size: 32
    shuffle: 1
  }
}
```

```
layers {
  name: "conv1"
  type: CONVOLUTION
  bottom: "data"
  top: "conv1"
  convolution_param {
    num_output: 20
    kernel_size: 5
    stride: 1
  }
}
```

Hình 3.5: Cách thức framework Caffe định nghĩa một lớp trong mạng CNN

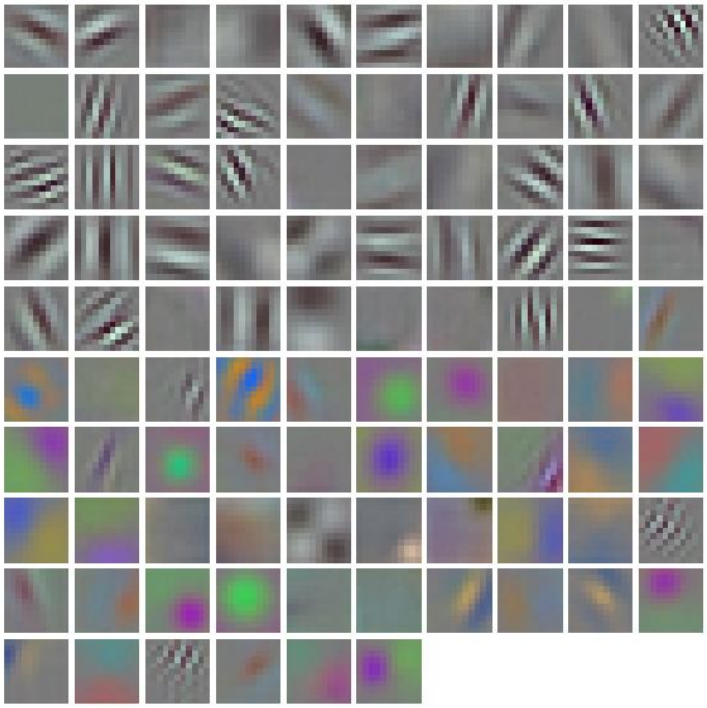
Các thông số của từng lớp trong mạng huấn luyện được định nghĩa rõ ràng trong file cấu hình, để giúp framework thực hiện việc huấn luyện mô hình một cách chính xác. Các lớp này yêu cầu người huấn luyện tùy chỉnh khá nhiều thông số theo từng bài toán cụ thể cần giải quyết, đặc biệt khi phải huấn luyện mô hình từ đầu. Tuy nhiên đối với trường hợp này, việc tinh chỉnh mô hình đã được huấn luyện trước chỉ cần ta quan tâm tới các tham số chính như sau:

- Thông tin ảnh đầu vào: Thông tin này được định nghĩa trong lớp "data", giúp framework có thể thực hiện tốt các phép tiền xử lý cần thiết, cũng như điều chỉnh lại kích cỡ ảnh đầu vào cho phù hợp với các lớp tính toán tích chập ở phía sau. Các phép tiền xử lý thường được sử dụng là các phép cắt ảnh, đối xứng gương và thay đổi tỉ lệ, là các cách đơn giản để giúp tăng thêm cơ sở ảnh trước khi thực hiện huấn luyện.
- Thông số tỉ lệ học: Tỉ lệ học được quyết định tại từng lớp, đối với các trường hợp huấn luyện từ đầu thì hầu hết tỉ lệ học tại các lớp là như nhau, với giá trị vừa phải để giúp cho các đặc trưng theo từng lớp được tính toán chuẩn xác. Với trường hợp tinh chỉnh mô hình thì sự khác biệt nằm ở việc các giá trị này được điều chỉnh về rất thấp, nhằm đảm bảo việc tính toán các đặc trưng không bị ảnh hưởng bởi bài toán mới. Đồng thời, tỉ lệ học cũng được tăng cường tại các lớp Kết nối đầy đủ tại phía sau cùng của mạng, từ đó việc huấn luyện cho mô hình mới sẽ nhanh chóng đạt được kết quả.
- Số lượng lớp nhận dạng đầu ra: Số lượng kết quả đầu ra cần tính toán sẽ được thay đổi tương ứng với số lượng lớp cần nhận dạng, con số này được định nghĩa trong lớp cuối cùng của mạng – lớp Kết nối đầy đủ. Cụ thể trong trường hợp nhận dạng hoa quả ta sẽ đặt thông số này là 40.

3.3.3. Một số hình ảnh về đặc trưng do mạng AlexNet tính toán

Như đã trình bày trong chương trước, các mạng CNN nói chung đều có thể được sử dụng như một bộ trích chọn đặc trưng làm đầu vào cho các bài toán phân loại, nhận dạng khác. Các lớp đầu tiên trong mạng huấn luyện của CNN có thể được coi là bộ bóc

tách các đặc trưng cơ bản, chung nhất cho mọi tác vụ nhận dạng ảnh như các đường thẳng, đường cong hay các đốm ảnh màu… Ta có thể thấy điều này trong Hình 3.6, với hình ảnh các đặc trưng thường gặp của lớp tích chập thứ nhất trong mạng AlexNet [25].



Hình 3.6: Các đặc trưng tiêu biểu của lớp tích chập đầu tiên [25]

Hình ảnh hóa các lớp trong mạng CNN là một trong những cách tiếp cận giúp người nghiên cứu hiểu thêm về cách thức mạng CNN nâng cao dần mức độ trừu tượng của kiến thức nó học được qua từng lớp trong mạng. Trong đó, phương pháp trực tiếp nhất là hình ảnh hóa các đặc trưng trong các lớp đầu của mạng do các đặc trưng này có khả năng chuyển giao tốt nhất. Đồng thời, độ nét và mịn của các đặc trưng cũng thể hiện cho mức độ huấn luyện của mạng, nếu mạng chưa được huấn luyện tốt, với kích thước CSDL ảnh lớn và thời gian huấn luyện đủ lâu, thì hình ảnh các đặc trưng sẽ bị nhiễu.

Ngoài ra ta cũng có thể hình ảnh hóa kết quả tính toán của các lớp nhân chập với một ảnh đầu vào cụ thể để có cái nhìn rõ hơn về thông tin mạng AlexNet có được sau các bước tính toán (xem Hình 3.7).

Hình 3.7: Kết quả ảnh đầu ra qua các lớp tích chập

3.4. Các mô đun phía Server

Chương trình phía server được cấu thành bởi các nhiều mô đun, đảm nhiệm các vai trò nhiệm vụ khác nhau liên quan đến huấn luyện, quản lý mô hình nhận dạng, cấu hình giao thức giao tiếp giữa client và server hay xử lý logic đa luồng, đảm bảo khả năng tính toán cho nhiều yêu cầu cùng lúc… Các ca sử dụng tương ứng với các nhiệm vụ này được thể hiện trong Hình 3.8 cùng với phần mô tả cụ thể bên dưới.

Hình 3.8: Biểu đồ ca sử dụng của Server

Đặc tả biểu đồ ca sử dụng:

Nạp mô hình nhận dạng:

- Mục đích: Nạp vào hệ thống mô hình nhận dạng phiên bản mới nhất được xuất ra bởi mô đun Bộ huấn luyện mô hình, phục vụ cho việc xử lý các yêu cầu nhận được từ ứng dụng phía client sau đó.
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống.
    - Mô tả chung: Để hệ thống có thể thực hiện tính toán và nhận dạng hoa quả trong ảnh do ứng dụng client gửi lên, mô hình nhận dạng cần phải được nạp trước vào hệ thống. Việc nạp mô hình này chỉ thực hiện một lần tại thời điểm bắt đầu một phiên chạy của server.
- Luồng sự kiện chính: Người quản trị khởi động chương trình server, hàm khởi tạo của chương trình tự động gọi câu lệnh thực thi việc nạp mô hình nhận dạng.
- Luồng thay thế: Chương trình server thông báo không nạp mô hình nhận dạng thành công, với thông tin cụ thể về lỗi xảy ra, như file định dạng chế độ triển khai của mô hình hoặc mô hình nhận dạng không hợp lệ.
- Các yêu cầu cụ thể: Phiên bản mô hình nhận dạng là phiên bản hoàn thiện mới nhất. Các thông tin cấu hình cho chương trình server phải hợp lệ.
- Điều kiện trước: Các thư viện cần thiết cho chương trình server đã được cài đặt đầy đủ và đúng phiên bản được khuyến cáo.
- Điều kiện sau: Thông báo nạp mô hình thành công.

Cấu hình giao thức gửi/nhận dữ liệu:

- Mục đích: Cấu hình các thông tin quyết định giao thức gửi, nhận dữ liệu giữa chương trình server và ứng dụng phía client, ví dụ: giao thức HTTP, cổng kết nối…
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống.
    - Mô tả chung: Người quản trị khi muốn khởi động chương trình server để nhận các yêu cầu từ ứng dụng client và gửi trả kết quả nhận dạng thì phải cấu hình giao thức gửi, nhận dữ liệu để thống nhất cách thức giao tiếp giữa hai thành phần server và client. Việc cấu hình này này chỉ thực hiện một lần tại thời điểm bắt đầu một phiên chạy của server.
- Luồng sự kiện chính: Người quản trị khởi động chương trình server, hàm khởi tạo của chương trình tự động gọi câu lệnh thực thi việc cấu hình các thông số cho giao thức gửi, nhận dữ liệu giữa server và ứng dụng phía client.
- Luồng thay thế: Chương trình server thông báo không thể cấu hình được giao thức gửi, nhận dữ liệu, với thông tin cụ thể về lỗi xảy ra.

- Các yêu cầu cụ thể: Các thông tin của giao thức gửi, nhận dữ liệu phải hợp lệ, cổng giao tiếp phải ở trạng thái tự do, không bị tranh chấp với các chương trình khác.
- Điều kiện trước: Các thư viện cần thiết cho chương trình server đã được cài đặt đầy đủ và đúng phiên bản được khuyến cáo. Chương trình server đã nạp thành công mô hình nhận dạng.
- Điều kiện sau: Thông báo cấu hình thành công giao thức gửi, nhận dữ liệu với các thông tin cụ thể của giao thức.

Tạo danh sách luồng xử lý:

- Mục đích: Khởi tạo trước một loạt các luồng xử lý, nhằm phục vụ quá trình tách riêng tính toán và nhận dạng cho từng yêu cầu phía client trong suốt phiên chạy của chương trình server.
- Tác nhân, Mô tả chung:
    - Tác nhân: Người quản trị hệ thống.
    - Mô tả chung: Trong quá trình chạy, để đảm bảo chương trình không bị chậm trễ khi đồng thời có nhiều yêu cầu từ các client khác nhau, chương trình server cần thực hiện việc xử lý này theo phương pháp đa luồng. Nghĩa là: mỗi yêu cầu từ phía client được xử lý trên một luồng riêng, không bị ảnh hưởng và không gây ảnh hưởng đến các luồng xử lý khác. Để thuận tiện cho việc quản lý và tránh tình trạng tràn bộ nhớ (leak-mem) do quản lý luồng không tốt, chương trình khởi tạo trước danh sách một loạt các luồng xử lý và sử dụng cờ trạng thái để giao việc cũng như giải phóng luồng.
- Luồng sự kiện chính: Người quản trị khởi động chương trình server, hàm khởi tạo của chương trình tự động gọi câu lệnh thực thi việc khởi tạo danh sách một loạt các luồng xử lý. Số lượng luồng xử lý được lưu dưới dạng hằng số trong file cấu hình chung của hệ thống.
- Luồng thay thế: Chương trình server thông báo không thể khởi tạo thành công các luồng xử lý.
- Các yêu cầu cụ thể: Không.
- Điều kiện trước: Các thư viện cần thiết cho chương trình server đã được cài đặt đầy đủ và đúng phiên bản được khuyến cáo. Chương trình server đã hoàn thành các bước Nạp mô hình huấn luyện và Cấu hình giao thức gửi, nhận dữ liệu.
- Điều kiện sau: Thông báo tạo thành công danh sách các luồng xử lý.

Gán luồng xử lý cho yêu cầu từ phía Client:

- Mục đích: Gán việc tính toán xử lý cho mỗi yêu cầu từ ứng dụng phía client cho một luồng xử lý đang ở trạng thái rảnh rỗi, nhằm hạn chế tối đa khả năng gây chậm trễ cho ứng dụng khi phải xử lý cùng lúc nhiều yêu cầu.
- Tác nhân, Mô tả chung:
    - Tác nhân: Thành phần xử lý logic quản lý luồng trong chương trình server.
    - Mô tả chung: Mỗi khi có yêu cầu mới từ phía client, đầu tiên server phải kiểm tra trạng thái tính toán của các luồng xử lý trong danh sách được khởi tạo từ ban đầu. Nếu có một luồng xử lý đang ở trạng thái rỗi, server gán luồng xử lý này cho yêu cầu mới nhận để luồng xử lý thực hiện các phép tính toán, nhận dạng và trả về kết quả cho client tương ứng. Nếu toàn bộ các luồng xử lý này đều ở trạng thái đang tính toán thì yêu cầu được đưa vào một hàng đợi, đợi đến khi có một luồng xử lý được giải phóng và đặt trạng thái rảnh rỗi.
- Luồng sự kiện chính: Thành phần xử lý logic quản lý luồng xử lý kiểm tra danh sách các luồng xử lý, gán yêu cầu mới nhận được cho một luồng xử lý rảnh rỗi.
- Luồng thay thế: Nếu không có luồng xử lý nào trong danh sách đang ở trạng thái rảnh rỗi, yêu cầu được đưa vào hàng đợi và sẽ được xử lý tiếp khi có một luồng xử lý hoàn thành việc tính toán trước đó.
- Các yêu cầu cụ thể: Trong danh sách luồng xử lý còn ít nhất một luồng ở trạng thái rảnh rỗi.
- Điều kiện trước: Các luồng xử lý phải được khởi tạo thành công từ khi bắt đầu chạy chương trình server.
- Điều kiện sau: Thông báo gán luồng xử lý thành công.

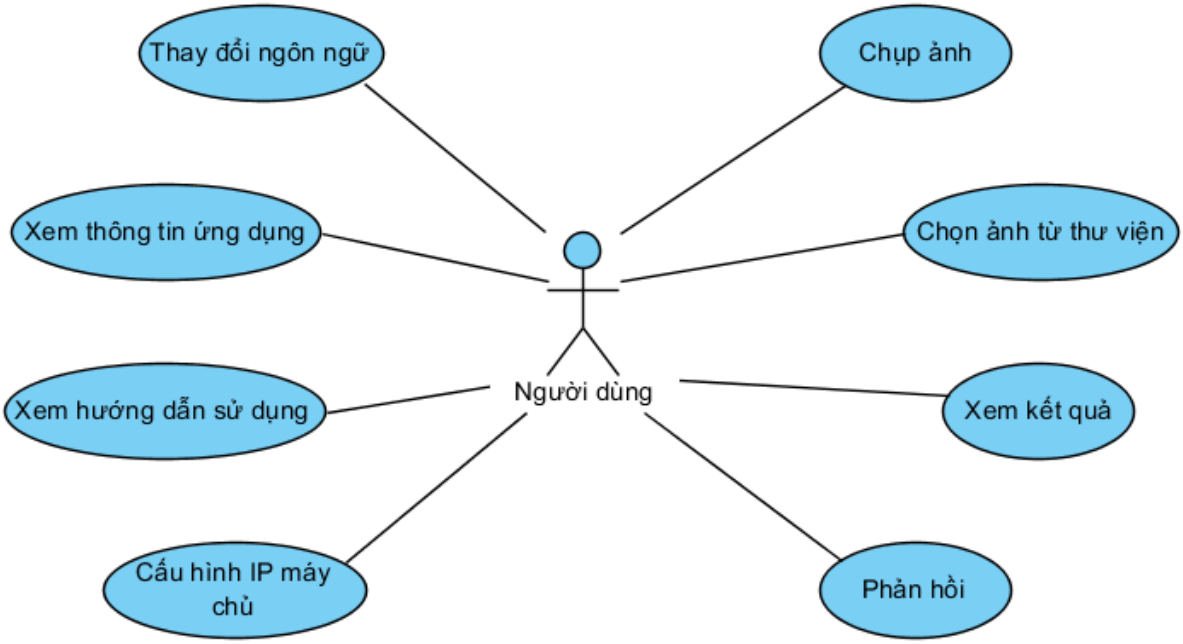
Giải phóng luồng xử lý:

- Mục đích: Cập nhật trạng thái của luồng xử lý khi đã hoàn thành việc tính toán và nhận dạng cho yêu cầu được giao.
- Tác nhân, Mô tả chung:
    - Tác nhân: Thành phần xử lý logic quản lý luồng trong chương trình server.
    - Mô tả chung: Khi luồng xử lý hoàn thành việc tính toán, nhận dạng và gửi kết quả về cho phía client, luồng cần được giải phòng bằng cách cập nhật lại trạng thái của luồng thành trạng thái rảnh rỗi. Luồng xử lý tiếp tục đợi đến khi được giao cho một yêu cầu nhận dạng mới từ server.

- Luồng sự kiện chính: Thành phần xử lý logic quản lý luồng xử lý cập nhật trạng thái của một luồng thành trạng thái rỗi khi luồng đó thông báo hoàn thành việc xử lý yêu cầu được giao trước đó.
- Luồng thay thế: Không.
- Các yêu cầu cụ thể: Thành phần xử lý logic quản lý luồng xử lý cần phải có cơ chế để nhận được thông báo hoàn thành tính toán từ các luồng trong danh sách.
- Điều kiện trước: Các luồng xử lý phải được khởi tạo thành công từ khi bắt đầu chạy chương trình server. Luồng xử lý đã hoàn thành tính toán và đưa ra thông báo tới thành phần quản lý logic luồng trong chương trình.
- Điều kiện sau: Thông báo luồng xử lý đã hoàn thành tính toán. Thành phần quản lý luồng xử lý cập nhật trạng thái của luồng thành trạng thái rảnh rỗi. Sau đó, thành phần quản lý luồng tiếp tục kiểm tra hàng đợi các yêu cầu từ client chưa được xử lý, nếu hàng đợi không rỗng thì lấy yêu cầu đầu tiên từ hàng đợi và gán cho luồng xử lý đang rảnh rỗi.

3.5. Ứng dụng phía Client

Ứng dụng phía Client [1] là ứng dụng trên điện thoại thông minh, là một thành phần trong hệ thống đảm nhiệm vai trò thu thập ảnh đầu vào để nhận dạng, gồm các chức năng chính sau đây: Chụp ảnh, Chọn ảnh từ thư viện và Xem kết quả nhận dạng do mô đun phía Server trả về. Ngoài ra, ứng dụng còn có các chức năng phụ khác, như: Phàn hồi kết quả, Thay đổi ngôn ngữ hiển thị, Xem thông tin về ứng dụng hoặc hướng dẫn sử dụng ứng dụng. Chức năng Cấu hình địa chỉ IP máy chủ là chức năng được sử dụng trong phiên bản thử nghiệm của hệ thống, khi các mô đun phía Server vẫn đang trong giai đoạn phát triển và kiểm thử chứ chưa đưa vào triển khai thực tế.



Hình 3.9: Biểu đồ ca sử dụng của Client

Hình 3.9 đã cho thấy các ca sử dụng của ứng dụng phía client, sau đây ta sẽ đi vào phần đặc tả chi tiết của ba ca sử dụng chính trong số đó:

Chụp ảnh:

- Mục đích: Chụp ảnh hoa quả và gửi ảnh mới chụp lên cho server thực hiện tính toán và nhận dạng.
- Tác nhân, Mô tả chung:
  - Tác nhân: Người dùng ứng dụng.
  - Mô tả chung: Khi người dùng muốn thực hiện nhận dạng một loại hoa quả nào đó, người dùng có thể chọn chức năng chụp ảnh trực tiếp của ứng dụng.
- Luồng sự kiện chính: Người dùng thực hiện chụp ảnh hoa quả, ứng dụng thực hiện mã hóa và nén dữ liệu ảnh rồi gửi lên server, đồng thời hiển thị thông báo cho người dùng chờ kết quả nhận dạng.
- Luồng thay thế: Ứng dụng thông báo lỗi không thể sử dụng camera của điện thoại, hoặc thông báo lỗi không thể kết nối đến chương trình server theo địa chỉ IP đã cấu hình.
- Các yêu cầu cụ thể: Không.
- Điều kiện trước: Ứng dụng đã được khởi động thành công, các thông tin cấu hình chung cho ứng dụng trên điện thoại được nạp thành công.
- Điều kiện sau: Thông báo người dùng chờ trong lúc chương trình server thực hiện tính toán nhận dạng.

Chọn ảnh trong thư viện ảnh:

- Mục đích: Lấy ra một ảnh đã chụp trong thư viện ảnh trên máy điện thoại để gửi lên cho server, yêu cầu tính toán nhận dạng.
- Tác nhân, Mô tả chung:
    - Tác nhân: Người dùng ứng dụng.
    - Mô tả chung: Khi người dùng muốn thực hiện nhận dạng một loại hoa quả nào đó, nhưng tại thời điểm đó người dùng không có kết nối mạng để kết nối tới server, người dùng có thể chụp và lưu lại ảnh hoa quả trong thư viện ảnh của máy điện thoại và thực hiện việc nhận dạng sau đó.
- Luồng sự kiện chính: Người dùng chọn một ảnh đã lưu trong thư viện ảnh, ứng dụng thực hiện mã hóa và nén dữ liệu ảnh rồi gửi lên server, đồng thời hiển thị thông báo cho người dùng chờ kết quả nhận dạng.
- Luồng thay thế: Ứng dụng không thể truy cập vào thư mục thư viện ảnh của điện thoại, hoặc thông báo lỗi không thể kết nối đến chương trình server theo địa chỉ IP đã cấu hình.
- Các yêu cầu cụ thể: Không.
- Điều kiện trước: Ứng dụng đã được khởi động thành công, các thông tin cấu hình chung cho ứng dụng trên điện thoại được nạp thành công.
- Điều kiện sau: Thông báo người dùng chờ trong lúc chương trình server thực hiện tính toán nhận dạng.

Xem kết quả:

- Mục đích: Hiển thị kết quả nhận dạng hoa quả nhận được từ chương trình server. Kết quả hiển thị cho người dùng bao gồm một loại hoa quả chính, với kết quả nhận dạng cao nhất, và bốn loại hoa quả với kết quả nhận dạng thấp hơn. Việc hiển thị nhiều loại hoa quả thay vì chỉ một loại chính là giúp người dùng tham khảo các loại hoa quả tương tự, đồng thời phục vụ cho tính năng phản hồi kết quả khi thông tin nhận dạng bị sai lệch.
- Tác nhân, Mô tả chung:
    - Tác nhân: Ứng dụng phía client.
    - Mô tả chung: Sau khi chương trình server nhận được yêu cầu từ phía client, server thực hiện tính toán trên luồng xử lý và trả về kết quả cho ứng dụng phía client. Lúc này ứng dụng sẽ hiển thị kết quả nhận được cho người dùng.
- Luồng sự kiện chính: Ứng dụng hiển thị cho người dùng kết quả nhận dạng, gồm danh sách năm loại quả có kết quả nhận dạng cao nhất.

- Luồng thay thế: Ứng dụng hiển thị kết quả là các chuỗi ký tự vô nghĩa, nguyên nhân do quá trình nhận và bóc tách dữ liệu từ server bị lỗi. Hoặc
- Các yêu cầu cụ thể: Không.
- Điều kiện trước: Ứng dụng đã gửi ảnh lên server và đang ở trạng thái đợi kết quả tính toán nhận dạng từ server.
- Điều kiện sau: Ứng dụng sẽ hiển thị thành công kết quả nhận được cho người dùng.

Để kết quả nhận dạng được tốt, các ảnh đầu vào cần phải thỏa mãn một số ràng buộc chính, các ràng buộc này nhằm đảm bảo sự tương tự nhất định giữa ảnh đầu vào và bộ ảnh dữ liệu được sử dụng để huấn luyện, từ đó nâng cao tỉ lệ nhận dạng chính xác của mô hình. Các ràng buộc cụ thể như sau:

1) Trong ảnh đầu vào chỉ có duy nhất một loại quả.
2) Hình ảnh của quả trong ảnh phải chiếm tỉ lệ nhất định trong ảnh, nếu hình ảnh của quả quá nhỏ sẽ dẫn đến khó khăn trong tính toán đặc trưng, gây nhầm lẫn giữa quả và nền, từ đó gây ra kết quả sai lệch.
3) Hình ảnh của quả trong ảnh không bị che lấp quá nhiều bởi vật thể khác, do yếu tố này có ảnh hưởng lớn đến kết quả tính toán đặc trưng của các lớp trong mạng.
4) Ảnh chụp đầu vào không bị quá nhòe hoặc điều kiện ánh sáng quá kém.

Hiện tại, ứng dụng phía Client mới chỉ được phát triển trên nền tảng điện thoại thông minh, hệ điều hành Android, và trong tương lai sẽ tiếp tục được phát triển trên các hệ điều hành cũng nền tảng khác nhau. Việc mở rộng này không quá phức tạp do các tính toán nhận dạng phức tạp đều dược phía Server xử lý, tính năng phía Client đã được đơn giản hóa và không có nhiều sự phụ thuộc vào nền tảng bên dưới.

Chương 4. Kết quả thử nghiệm và đánh giá

4.1. So sánh với phương pháp Học máy truyền thống

Thực nghiệm với phương pháp Học máy truyền thống:

- **Bước 1:** Xây dựng CSDL ảnh hoa quả cho 20 loại quả.
- **Bước 2**: Tiền xử lý ảnh trong CSDL (lọc nền) và gán nhãn. Đặc trưng của bộ CSDL ảnh này là các ảnh đều được thu thập bằng cách chụp thủ công, nhằm đảm bảo các ảnh có chất lượng cao, có cùng kích thước và tỉ lệ ảnh, với nền đã bị loại bỏ hoàn toàn.

Hình 4.1: Một số ảnh đã lọc nền trong bộ CSDL 20 loại quả

- **Bước 3:** Chọn lọc đặc trưng, cụ thể:
  - *Về màu sắc*:

Sử dụng 16 đặc trưng về số lượng các điểm ảnh với giá trị màu tính theo hệ màu HSI (Hue-Saturation-Intensity). Ta không sử dụng hệ màu thường gặp nhất là RGB bởi sau khi chuyển sang hệ màu HSI, ta đã có thể tách biệt được thông tin màu sắc với những thành phần khác như độ sáng, sự bão hòa…

Cụ thể hơn, ta chia dải màu Hue thành 12 đoạn tương ứng với 12 dải màu chính (đỏ, vàng, xanh lục…) và chia dải giá trị độ thuần khiết màu sắc Saturation thành 4 đoạn, sau đó thống kê số điểm ảnh có giá trị điểm màu nằm trong các dải này để thu được 16 giá trị đặc trưng về màu sắc cho mỗi ảnh đầu vào.

- *Về hình dạng*:

Sử dụng 4 đặc trưng về hình dạng của hoa quả trong ảnh là chu vi, diện tích, độ dài lớn nhất, độ rộng lớn nhất của hoa quả trong ảnh.

- *Về kết cấu*:

Sử dụng 10 đặc trưng về kết cấu, là 10 tham số trong bộ ma trận GLCM (Grey Level Co-occurrence Matrix) – một ma trận tính toán đặc trưng kết cấu phổ biến trong lĩnh vực Xử lý ảnh, VD một số tham số được sử dụng như: Entropy, Energy, Homogenity, Contrast, Correlation…

Tổng kết lại, với mỗi ảnh đầu vào ta sẽ tính toán được 30 giá trị đại diện cho 30 đặc trưng về màu sắc, hình dạng và kết cấu. Những đặc trưng này được chọn lựa sau quá trình tìm hiểu các bài báo, công trình khoa học về sử dụng Học máy trong bài toán nhận dạng hoa quả và thống kê các đặc trưng được sử dụng nhiều nhất, đạt hiệu quả tốt nhất.

- **Bước 4:** Huấn luyện mô hình nhận dạng hoa quả từ CSDL ảnh đã xây dựng. Bộ CSDL ảnh này chỉ để so sánh tương đối độ chính xác của mô hình truyền thống so với mô hình học sâu tiên tiến bây giờ, do đó số lượng loại hoa quả được hạn chế chỉ còn 20 loại, với số lượng ảnh cho mỗi loại là 400-600 ảnh.
- **Bước 5:** Thống kê độ chính xác với tỉ lệ bộ training/test là 75/25. Kết quả đạt được không cao, chỉ đạt ~74.5% trên bộ dữ liệu test 2.600 ảnh, và khi thử nghiệm thực tế cũng gặp phải sai số lớn (do ảnh chụp thực tế có chất lượng không cao và sự khác biệt lớn so với bộ CSDL ảnh để huấn luyện).

Thực nghiệm với phương pháp Học sâu:

- **Bước 1:** Xây dựng CSDL ảnh hoa quả cho 20 loại quả.
- **Bước 2**: Tiền xử lý ảnh trong CSDL (lọc nền) và gán nhãn. Hai bước đầu tiên này chỉ cần thực hiện một lần khi xây dựng bộ CSDL ảnh huấn luyện cho phương pháp Học máy truyền thống.
- **Bước 3:** Thực hiện các bước tính toán cần thiết để ứng dụng mô hình AlexNet.
- **Bước 4:** Huấn luyện mô hình nhận dạng hoa quả từ CSDL ảnh đã xây dựng.
- **Bước 5:** Thống kê độ chính xác với tỉ lệ bộ training/test là 75/25. Xem hình kết quả ta có thể thấy độ chính xác đạt được là rất cao, ~98.8%, vượt trội so với với độ chính xác của mô hình huấn luyện sử dụng phương pháp Học máy truyền thống.

Đánh giá kết quả:

Với kết quả thu được từ hai mô hình huấn luyện sử dụng hai phương pháp khác nhau trên cùng một bộ CSDL ảnh chất lượng tốt và đã được tiền xử lý cũng như gán nhãn cẩn thận, ta có thể rút ra kết luận như sau: Với các bài toán nhận dạng và phân loại đối tượng nói chung, trong đó rất khó có thể chọn được các đặc trưng hiệu quả, thì Học sâu là phương pháp có ưu thế vượt trội so với các phương pháp Học máy truyền thống. Học sâu giúp đơn giản hóa quá trình huấn luyện mô hình nhận dạng khi không yêu cầu sự tham gia của người huấn luyện trong quá trình trích chọn đặc trưng, đồng thời cho phép tái sử dụng các mô hình đã huấn luyện trước để giảm thời gian cài đặt giải pháp cho các bài toán nhận dạng mới.

Thông tin tổng quan về bộ CSDL ảnh và quá trình huấn luyện cũng như kết quả đạt được của hai phương pháp cũng được tóm lược trong bảng bên dưới:

Bảng 4.1: So sánh sơ bộ kết quả huấn luyện của 2 phương pháp

	Bộ CSDL ảnh	**Thời gian huấn luyện**	**Độ chính xác**
Học máy truyền thống	- Số lượng hoa quả cần nhận dạng: 20 loại - Số lượng ảnh trung bình cho mỗi loại quả: 400-600 ảnh - Tổng số ảnh được sử dụng để huấn luyện: 10.400 ảnh	120 phút	74.50%
Học sâu		360 phút	98.76%

4.2. So sánh kết quả với bộ CSDL được sinh tự động

Trong mục 2.3, ta đã chứng minh được khả năng của Học chuyển giao trong việc giữ được độ chính xác cao của mô hình huấn luyện chỉ với bộ dữ liệu có kích thước không lớn. Tuy nhiên khi thực hiện cài đặt và tinh chỉnh mô hình, ta vẫn phải liên tục tăng cường, bổ sung CSDL ảnh để mô hình huấn luyện ngày càng hiệu quả, các tham số và các đặc trưng cũng được cải thiện, riêng biệt hóa cho bài toán nhận dạng hoa quả. Để kiểm chứng sự ảnh hưởng của kích thước bộ CSDL ảnh lên độ chính xác của mô hình nhận dạng, ta thực hiện huấn luyện mô hình hai lần riêng biệt với bộ dữ liệu chỉ gồm ảnh gốc và với bộ dữ liệu bao gồm cả ảnh gốc cùng với các ảnh được tự động sinh thêm nhờ các thuật toán xử lý ảnh.

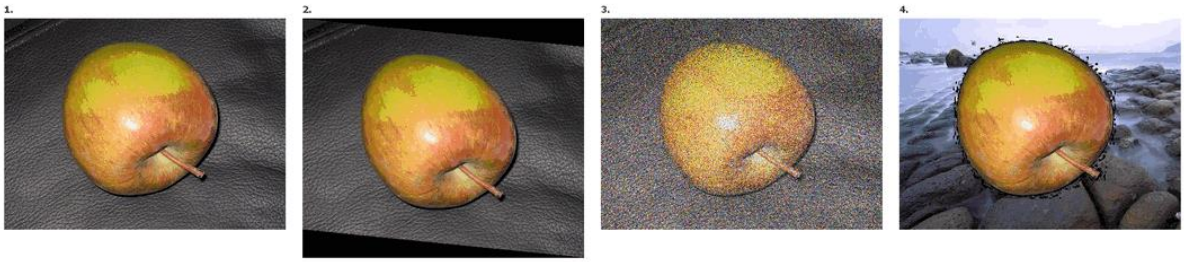
Do kết quả phần thực nghiệm này sẽ được ứng dụng vào chương trình server để sử dụng trong thực tế, bộ CSDL ảnh được sử dụng là bộ CSDL ảnh đầy đủ, gồm các ảnh được thu thập và gán nhãn cho 40 loại hoa quả khác nhau.

Thực nghiệm với bộ CSDL ảnh gốc:

- Số lượng ảnh gốc cho mỗi loại quả: 500-1000 ảnh
- Tổng số ảnh dùng để huấn luyện: 21.000 ảnh
- Tổng số ảnh dùng để test: 7.000 ảnh
- Thời gian huấn luyện cho 20.000 lượt: 5 tiếng
- Độ chính xác: 65,49%

Thực nghiệm với bộ CSDL ảnh được sinh tự động từ ảnh gốc:

Từ mỗi ảnh gốc, sau khi sử dụng các thuật toán xử lý ảnh như chiếu nghiêng (skew), thêm nhiễu và ghép nền khác ta sẽ thu được 9 ảnh mới để tăng cường cho bộ CSDL ảnh huấn luyện.



Hình 4.2: Ảnh hoa quả gốc và các ảnh được sinh tự động

- Số lượng ảnh gốc cho mỗi loại quả: 500-1000 ảnh
- Số lượng ảnh sinh thêm từ một ảnh gốc: 9 ảnh
- Tổng số ảnh dùng để huấn luyện: 210.000 ảnh
- Tổng số ảnh dùng để test: 70.000 ảnh
- Thời gian huấn luyện cho 20.000 lượt: 30 tiếng
- Độ chính xác: 98,67%

Đánh giá kết quả:

Sự cải thiện rõ rệt trong độ chính xác của mô hình nhận dạng sau khi tăng cường CSDL ảnh huấn luyện đã cho thấy hiệu quả thực tế của các phép sinh ảnh tự động sử dụng các phương pháp xử lý ảnh cơ bản. Chất lượng nhận dạng của ứng dụng trong thực tế cũng được tăng lên do các ảnh được sinh tự động giúp mô phỏng quá trình chụp ảnh trong đời thực, như các góc chụp khác nhau, các nhiễu sinh ra do môi trường, chất lượng máy ảnh… cũng như sự đa dạng của nền mà người dùng sử dụng để chụp ảnh. Việc tăng cường CSDL ảnh cũng là một giải pháp cho trường hợp khó thu thập ảnh để huấn luyện mô hình, tuy nhiên cũng cần phải chú ý đến mặt trái của việc lạm dụng phương pháp tăng cường ảnh này, đó là nguy cơ gây ra trạng thái "overfit" dữ liệu (mô hình nhận dạng quá khớp với dữ liệu huấn luyện mà bị sai lệch với dữ liệu thực tế).

Bảng 4.2 tóm lược lại kết quả so sánh độ chính xác của mô hình nhận dạng được huấn luyện với hai bộ CSDL khác nhau: một bộ CSDL ảnh gốc và một bộ có bổ sung thêm các ảnh được sinh tự động bởi thuật toán Xử lý ảnh.

Bảng 4.2: Ảnh hưởng của bộ ảnh sinh tự động với chất lượng mô hình nhận dạng

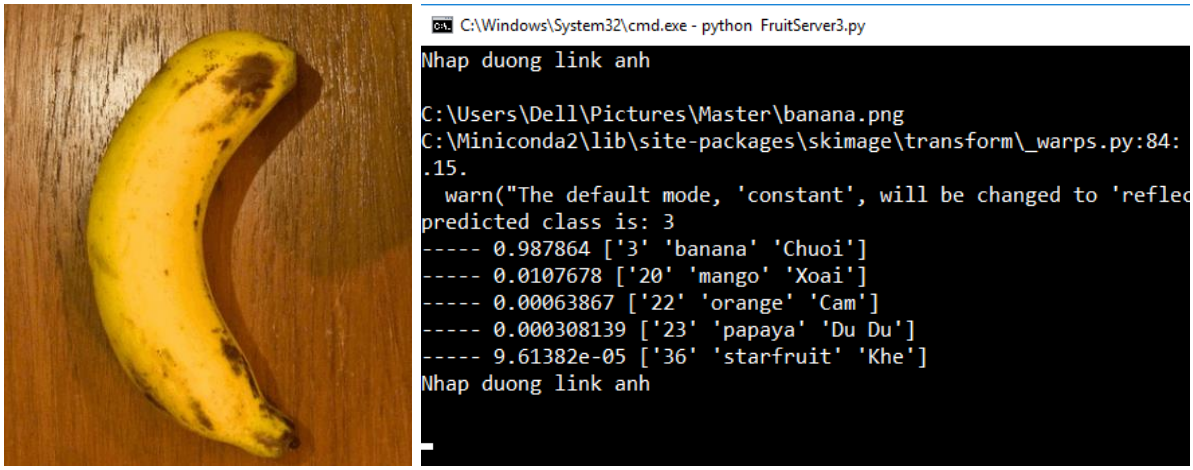
	Bộ CSDL ảnh	**Thời gian huấn luyện**	**Độ chính xác**
Bộ CSDL ảnh gốc	- Số lượng hoa quả cần nhận dạng: 40 loại - Số lượng ảnh trung bình cho mỗi loại quả: 500-1000 ảnh - Tổng số ảnh được sử dụng để huấn luyện: 28.000 ảnh	5 tiếng	65,49%
Bộ CSDL ảnh gốc, bổ sung thêm ảnh sinh tự động	- Số lượng hoa quả cần nhận dạng: 40 loại	30 tiếng	98,67%

	- Số lượng ảnh trung bình cho mỗi loại quả: 5.000-10.000 ảnh - Tổng số ảnh được sử dụng để huấn luyện: 280.000 ảnh		

4.3. Thử nghiệm ứng dụng trong thực tế

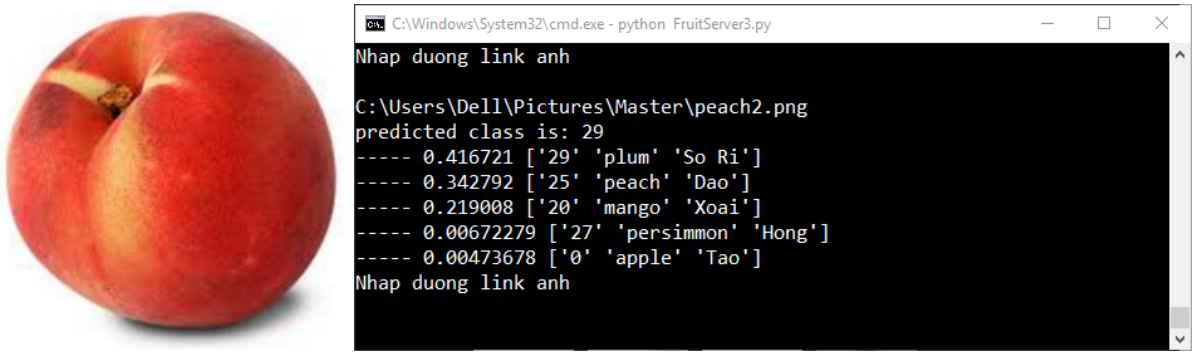
Ứng dụng Nhận dạng hoa quả - Fruit Recognition - đã được thử nghiệm thực tế với nhiều mẫu hoa quả khác nhau, được chia thành hai nhóm chính: Nhóm đã được huấn luyện nhận dạng và nhóm chưa được huấn luyện. Kết quả đạt được tương đối tốt, cụ thể như sau:

- Nhóm hoa quả nằm trong danh sách 40 quả được chọn để xây dựng bộ dữ liệu nhận dạng: Kết quả nhận dạng đạt độ chính xác khá cao, đặc biệt là với những loại quả có nét đặc trưng về màu sắc hoặc hình dạng như chuối, thanh long, chôm chôm…



Hình 4.3: Kết quả nhận dạng tốt với loại quả có đặc trưng riêng biệt

Đối với những loại quả có nhiều nét tương đồng lẫn nhau, kết quả nhận dạng của ứng dụng còn đôi lúc bị nhầm lẫn, đặc biệt trong các trường hợp ảnh được chụp theo góc nhìn chưa tốt dẫn đến ảnh không thể hiện được các đặc trưng riêng của quả. Những sai sót này là không thể tránh khỏi vì trong nhiều trường hợp, mắt người cũng không dễ dàng phân biệt được chúng nếu chỉ dựa vào một hình ảnh chụp mà không có sự hỗ trợ của các giác quan khác như khứu giác hay vị giác.



Hình 4.4: Kết quả nhận dạng chưa tốt với loại quả không có đặc trưng riêng biệt

Có thể thấy trong hình trên, hình ảnh quả đào được chụp ở góc độ chưa tốt, khiến cho hệ thống nhận dạng nhầm lẫn. Tuy nhiên ta có thể thấy được điều này qua thông số thể hiện độ chính xác khi nhận dạng mà mô hình đưa ra: tỉ lệ nhận dạng đúng của quả mận chỉ đạt 41,67%, không cao hơn nhiều so với quả đào là 34,28%, và quá thấp so với tỉ lệ nhận dạng thông thường (lớn hơn 90%).

- Nhóm hoa quả nằm ngoài danh sách 40 quả: Hệ thống sẽ tính toán và trả về kết quả nhận dạng là một trong 40 loại hoa quả có tỉ lệ giống nhất với loại quả cần nhận dạng. Độ tương đồng giữa hai loại quả này ta có thể nhận thấy rất rõ ràng:

Hình 4.5: Kết quả nhận dạng với loại quả không được huấn luyện

Trong trường hợp như hình trên, khi ta yêu cầu hệ thống nhận dạng quả bòn bon, do bòn bon không có trong danh sách 40 quả được huấn luyện nhận dạng nên kết quả trả về là loại quả có sự tương đồng cao nhất, quả nhãn.

Ngoài ra, kết quả thực nghiệm thu được cho thấy hệ thống nhận dạng đạt được kết quả tương đối chuẩn xác với các trường hợp hình ảnh quả trong ảnh đầu vào bị che khuất một phần, điều kiện ánh sáng không thực sự tốt cũng như các trường hợp ảnh bị biến dạng nhẹ. Đây chính là các khó khăn đối với bài toán nhận dạng vật thể nói chung mà ta đã đề cập tới trong phần mở đầu của luận văn, lý giải cho điều này là do trong quá trình thu thập ảnh ban đầu cũng như sinh ảnh tự động từ các ảnh gốc, mô hình nhận dạng đã được huấn luyện để nhận ra các trường hợp tương tự. Khả năng dự đoán mạnh mẽ này đã giúp cho các phương pháp Học sâu, đặc biệt là mạng huấn luyện no ron tích chập CNN trở thành giải pháp mạnh mẽ nhất trong lĩnh vực nhận dạng ảnh bây giờ.

Chương 5. Kết luận

Luận văn đã nghiên cứu, tìm hiểu bài toán tự động nhận dạng và phân loại hoa quả trong ảnh màu, và thực hiện phát triển, cài đặt phương án giải quyết cho bài toán dựa trên sự thống kê các hướng tiếp cận đã được công bố qua rất nhiều bài báo, công trình khoa học trên thế giới. Các kết quả chính mà luận văn đã đạt được, tương ứng với các mục tiêu đề ra ban đầu như sau:

- Hoàn thiện xây dựng bộ cơ sở dữ liệu ảnh phục vụ huấn luyện nhận dạng cho 40 loại hoa quả phổ biến ở nước ta, với số lượng ảnh gốc trung bình cho mỗi loại quả là từ 500-1000 ảnh.
- Thống kê các đặc trưng thường được sử dụng để huấn luyện bộ nhận dạng hoa quả trong các phương pháp Học máy truyền thống, bao gồm các đặc trưng về màu sắc, hình dạng và kết cấu. Từ đó làm cơ sở xây dựng một mạng nơ-ron nhân tạo truyền thống và so sánh kết quả với một mạng nơ-ron tích chập thuộc nhóm phương pháp Học sâu.
- Cài đặt và tinh chỉnh một mạng nơ-ron tích chập đã được huấn luyện trước, ứng dụng vào bài toán nhận dạng hoa quả. Đồng thời xây dựng hệ thống tự động nhận dạng hoa quả Fruit Recognition System với ứng dụng client trên điện thoại thông minh.

Thực nghiệm với bộ dữ liệu test và trong thực tế đã cho kết quả khá tốt, nguyên nhân chính là do phạm vi số lượng hoa quả để nhận dạng đã được hạn chế chỉ còn 40 loại – một con số rất khiêm tốn so với số lượng hoa quả ở Việt Nam nói riêng và cả thế giới nói chung. Hệ thống tự động nhận dạng hoa quả còn cần rất nhiều cải thiện, đặc biệt là về khả năng mở rộng phạm vi loại hoa quả cũng như kích thước, chất lượng của bộ CSDL ảnh huấn luyện. Trong tương lai, để có thể cải thiện độ chính xác của mô hình nhận dạng, tôi đề xuất cài đặt thử nghiệm và đánh giá các loại mô hình mạng Học sâu đã được huấn luyện trước, đặc biệt là các mạng đã đạt được kết quả cao trong cuộc thi Nhận dạng ảnh quy mô lớn do ImageNet tổ chức thường niên như: ZF Net (2013), VGG Net (2014), GoogleNet và Microsoft ResNet (2015)…

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Trần Tuấn Linh. (2017). Ứng dụng nhận dạng hoa quả cho điện thoại thông minh dựa trên hình ảnh.

[2] Vũ Hữu Tiệp. (2017). Machine Learning cơ bản. http://machinelearningcoban.com/general/2017/02/06/featureengineering/

Tiếng Anh

[3] Andrej Karpathy. CS231n Convolutional Neural Networks for Visual Recognition - Image Classification. http://cs231n.github.io/classification/

[4] Sadrnia, H., Rajabipour, A., Jafary, A., Javadi, A., & Mostofi, Y. (2007). Classification and analysis of fruit shapes in long type watermelon using image processing. *Int J Agric Biol*, *9*(1), 68–70.

[5] Fu, L., Sun, S., Li, R., & Wang, S. (2016). Classification of kiwifruit grades based on fruit shape using a single camera. *Sensors (Switzerland)*, *16*(7), 1–14.

[6] Seng, W. C., & Mirisaee, S. H. (2009). A new method for fruits recognition system. *Proceedings of the 2009 International Conference on Electrical Engineering and Informatics, ICEEI 2009*, *1*, 130–134.

[7] Arivazhagan, S., Shebiah, R. N., Nidhyanandhan, S. S., & Ganesan, L. (2010). Fruit Recognition using Color and Texture Features. *Information Sciences*, *1*(2), 90–94.

[8] Zhang, Y., & Wu, L. (2012). Classification of fruits using computer vision and a multiclass support vector machine. *Sensors (Switzerland)*, *12*(9), 12489–12505.

[9] Naskar, S. (2015). A Fruit Recognition Technique using Multiple Features and Artificial Neural Network, *116*(20), 23–28.

[10] GilPress. (2016). Visually Linking AI, Machine Learning, Deep Learning, Big Data and Data Science | What's The Big Data? https://whatsthebigdata.com/2016/10/17/visually-linking-ai-machine-learning-deep-learning-big-data-and-data-science/

[11] Lee, H., Grosse, R., Ranganath, R., & Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, 1–8.

[12] Huew Engineering. (2015). Introduction to Convolution Neural Networks – Huew Engineering. https://engineering.huew.co/introduction-to-convolution-neural-networks-18981d1cd09a

[13] Dumoulin, V., & Visin, F. (2016). A guide to convolution arithmetic for deep learning.

[14] Samer, C. H., Rishi, K., & Rowen. (2015). Image Recognition Using Convolutional Neural Networks. *Cadence Whitepaper*, 1–12.

[15] Andrej Karpathy. (n.d.). CS231n Convolutional Neural Networks for Visual Recognition - Transfer Learning. http://cs231n.github.io/transfer-learning/

[16] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems 27 (Proceedings of NIPS)*, *27*, 1–9.

[17] Jimmie Goode. (2015). Classifying images in the Oxford 102 flower dataset with CNNs – Jimmie Goode. http://jimgoo.com/flower-power/

[18] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, *115*(3), 211–252.

[19] ImageNet Large Scale Visual Recognition Competition 2012 (ILSVRC2012). http://image-net.org/challenges/LSVRC/2012/results.html

[20] Krizhevsky, A., Sutskever, I., & Geoffrey E., H. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems 25 (NIPS2012)*, 1–9.

[21] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. http://arxiv.org/abs/1408.5093

[22] Welcome — Theano 0.9.0 documentation. http://deeplearning.net/software/theano/

[23] Torch | Tutorials for learning Torch. http://torch.ch/docs/tutorials.html

[24] TensorFlow. https://www.tensorflow.org/

[25] Andrej Karpathy. (n.d.). CS231n Convolutional Neural Networks for Visual Recognition - Visualizing what ConvNets learn. http://cs231n.github.io/understanding-cnn/