TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Hứa Phú Thành - Hà Minh Toàn -
Trần Ngọc Tịnh - Trần Luật Vy

THỰC TẬP DỰ ÁN TỐT NGHIỆP

ĐỒ ÁN TỐT NGHIỆP CỬ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

Tp. Hồ Chí Minh, tháng 06/2022

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Hứa Phú Thành - 18120563
Hà Minh Toàn - 18120597
Trần Ngọc Tịnh - 18120599
Trần Luật Vy - 18120656

PHÂN LOẠI ẢNH CÓ KHỐI U VÀ KHÔNG CÓ KHỐI U Ở PHỔI

ĐỒ ÁN TỐT NGHIỆP CỬ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

GIÁO VIÊN HƯỚNG DẪN
PGS.TS. Lê Hoàng Thái

Tp. Hồ Chí Minh, tháng 06/2022

Nhận xét hướng dẫn

Nhận xét phản biện

Lời cảm ơn

Sau một thời gian thực hiện đề tài với nhiều nỗ lực và cố gắng, thời điểm hoàn thành đồ án là lúc chúng em xin phép được bày tỏ lòng biết ơn chân thành với những người đã dạy dỗ, hướng dẫn và giúp đỡ em trong suốt thời gian qua.

Trước hết, chúng em xin chân thành bày tỏ lòng kính trọng và biết ơn sâu sắc tới thầy Lê Hoàng Thái đã hướng dẫn tận tình, động viên và giúp đỡ chúng em trong suốt quá trình thực hiện khóa luận.

Xin cảm ơn Ban giám hiệu, Phòng đào tạo, các phòng ban đã tạo mọi điều kiện giúp chúng em hoàn thành đồ án tốt nghiệp. Cám ơn các thầy cô trong khoa Công nghệ Thông tin - Khoa học Máy tính đã quan tâm, dìu dắt và truyền kiến thức cho chúng em trong 4 năm học vừa qua.

Cảm ơn những người bạn đã luôn bên cạnh hợp tác giúp đỡ nhau để có thể hoàn thành được khóa luận. Cảm ơn những giây phút đã tập trung nghiên cứu với nhau, những buổi trao đổi kiến thức bổ ích.

Xin chân thành cảm ơn tất cả mọi người!

Đề cương chi tiết

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

ĐỀ CƯƠNG THỰC TẬP DỰ ÁN TỐT NGHIỆP

PHÂN LOẠI ẢNH CÓ KHỐI U VÀ KHÔNG CÓ KHỐI U Ở PHỔI

(LUNG TUMOR CLASSIFICATION)

1 THÔNG TIN CHUNG

Người hướng dẫn:

– PGS. TS. Lê Hoàng Thái (Khoa Công nghệ Thông tin)

[Nhóm] Sinh viên thực hiện:

1. Hứa Phú Thành (MSSV: 18120563)
2. Hà Minh Toàn (MSSV: 18120597)
3. Trần Ngọc Tịnh (MSSV: 18120599)
4. Trần Luật Vy (MSSV: 18120656)

Loại đề tài: Nghiên cứu

Thời gian thực hiện: Từ *1/2022* đến *7/2022*

2 NỘI DUNG THỰC HIỆN

2.1 Giới thiệu về đề tài

Đối với mỗi con người chúng ta thì sức khỏe là thứ quý giá nhất và quan trọng

nhất. Các nguyên nhân ảnh hưởng đến sức khỏe hầu hết là do thói quen ăn uống và sinh hoạt của mỗi người. Nếu bạn có một chế độ dinh dưỡng tốt, cùng với việc luyện tập thể dục thể thao thường xuyên chắc chắn bạn sẽ có sức khỏe tốt. Và ngược lại, nếu chế độ ăn uống không lành mạnh, lười vận động, thường xuyên sử dụng các chất kích thích như: rượu, bia, thuốc lá, ... sẽ làm cho sức khỏe chúng ta giảm sút nghiêm trọng. Dẫn đến việc thường xuyên mắc bệnh và nguy hiểm nhất đó chính là bệnh ung thư.

Theo các tổ chức y tế thế giới, ung thư phổi là một trong những loại ung thư phổ biến nhất trên thế giới hiện nay. Triệu chứng thường xảy ra như ho, tức ngực hoặc ho ra máu. Tuy nhiên, cũng có rất nhiều trường hợp bệnh nhân không có bất kỳ triệu chứng lâm sàng nào hoặc khi có các triệu chứng thì ung thư đã vào giai đoạn cuối. Hiện nay, chúng ta có thể phát hiện các bệnh liên quan tới phổi, đặc biệt là ung thư bằng biện pháp chụp cắt lớp phổi (chụp CT phổi). Biện pháp này giảm đáng kể tỷ lệ tử vong do ung thư phổi nhờ việc phát hiện sớm và điều trị kịp thời.

Hiện nay với sự phát triển mạnh mẽ của trí tuệ nhân tạo, việc chẩn đoán, phát hiện các bệnh, các biểu hiện bất thường trên ảnh y khoa trở nên thuận lợi hơn. Ta có thể thu được kết quả tốt trong việc giải quyết các bài toán trên thông qua việc sử dụng các phương pháp học máy như: K-Mean, KNN, SVM, ... hoặc sử dụng các mô hình Deep learning (mô hình học sâu) như Convolutional neural network (CNN - mạng nơron tích chập). Thế nên, trong đồ án này, nhóm sẽ tập trung nghiên cứu để giải quyết bài toán "Phân loại ảnh có khối u hoặc không có khối u ở phổi" bằng việc ứng dụng trí tuệ nhân tạo.

2.2 Mục tiêu đề tài

Ngày nay, với sự phát triển nhanh chóng của công nghệ rất nhiều các thiết bị hiện đại được áp dụng vào lĩnh vực y tế như: hệ thống MRI 1.5 của Tesla, hệ thống CT Scanner 160 lát, hệ thống chụp mạch xóa nền (DSA), ... Và không thể không nhắc đến những ứng dụng của công nghệ thông tin trong y khoa như các ứng dụng theo dõi sức khỏe bệnh nhân, đặt lịch khám trực tuyến, hệ thống lưu

trữ và truyền hình ảnh (PACS), ... Bằng việc áp dụng công nghệ thông tin khối lượng dữ liệu trong lĩnh vực y tế được sản sinh ra vô cùng đáng kể và ảnh y khoa là một trong số đó. Với một khối lượng lớn hình ảnh y khoa (MRI, CT, X-quang, ...) đòi hỏi các bác sĩ phải tốn thêm nhiều thời gian trong việc chẩn đoán các bệnh, các biểu hiện bất thường trên ảnh. Thế nên việc xây dựng một mô hình hỗ trợ các bác sĩ chẩn đoán, phát hiện các biểu hiện bất thường trên ảnh y khoa, với độ chính xác cao là hết sức cần thiết, từ đó có thể đưa ra phác đồ điều trị tốt nhất cho bệnh nhân. Sau mỗi lần chụp CT, X-quang, ... các bác sĩ không còn loay hoay với việc xem các ảnh phim và bệnh nhân không còn tốn quá nhiều thời gian để chờ kết quả vì đã có các mô hình máy tính giúp ta làm những công việc trên.

Mục tiêu cụ thể của nhóm trong đồ án này là "Xây dựng mô hình phân loại ảnh có khối u hoặc không có khối u ở phổi". Bài toán được đặt ra ở đây là bài toán phân loại ảnh (Image classification). Phương pháp mà nhóm hướng đến là xây dựng mô hình phân lớp bằng mô hình học sâu. Ung thư là một trong những bệnh lý vô cùng nguy hiểm nếu không được phát hiện và điều trị kịp thời. Do đó, nhóm sử dụng các phương pháp nêu trên với hy vọng xây dựng một mô hình phân loại có độ chính xác cao, giảm tỷ lệ dương tính giả và âm tính giả của các mẫu được dự đoán.

2.3 Phạm vi của đề tài

Đối tượng nghiên cứu:

- Đề tài nghiên cứu lấy cảm hứng từ bài báo "Detection of Lung Nodules on CT Images based on the Convolutional Neural Network with Attention Mechanism" [1].

- Tập dữ liệu dùng để nghiên cứu: LIDC/IDRI [1]. Tập dữ liệu này được dùng trong cuộc thi mang tên LUNA16 [2] . Mục đích của cuộc thi này nhằm giải quyết 2 vấn đề chính: vấn đề thứ nhất là xây dựng mô hình phát hiện các nốt sần ở phổi, sau khi xác định được vị trí các nốt sần ta sẽ đi đến việc giải quyết vấn đề thứ hai là xây dựng mô hình phân loại nốt sần nào có chứa khối u và không chứa khối u,

[1] https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI
[2] https://luna16.grand-challenge.org/

nhằm giảm tỉ lệ dương tính giả. Nhóm tập trung nghiên cứu vào vấn đề thứ hai.

Phạm vi nghiên cứu:

- Tìm hiểu về các phương pháp tăng cường dữ liệu.
- Xây dựng mô hình phân loại kết hợp giữa mạng DenseNet và Channel attention module.
- Xây dựng mô hình phân loại bằng cách sử dụng mô hình EfficientNet.
- Tìm hiểu các phương pháp đánh giá mô hình phân loại.

2.4 Cách tiếp cận dự kiến

Phương pháp Data Augmentation (tăng cường dữ liệu)

Với những bài toán học sâu thì dữ liệu luôn là rất khan hiếm như đã đề cập ở trên. Bởi sự cần thiết rằng các thuật toán học máy đều coi Data là một phần rất quan trọng, càng nhiều data thì chất lượng đầu ra để ứng dụng sẽ càng được cải thiện. Một số phương pháp để nạp thêm dữ liệu như:

- Adding noise, cropping, flipping, rotation, scaling, ... (những phương pháp này thường được dùng trong computer vision) [2].
- Ngoài ra còn có phương pháp Data synthesis hay còn gọi là tạo dữ liệu giả. Tuy nhiên với đồ án hình ảnh này thì dữ liệu giả muốn tạo ra là rất khó áp dụng được.
- Và còn một cách bên cạnh các phương pháp cắt, xoay ảnh như trên là Collect more data, tức là sẽ thêm ảnh giống như bộ dữ liệu đã có bằng những ảnh được tìm kiếm trên mạng (được cho phép sử dụng trong nghiên cứu học tập).

Mô hình đề xuất thứ nhất: DenseNet kết hợp Channel attention module

Như ta đã biết, DenseNet là một trong những mạng phân loại khá hiện đại. Nó được xem như là một biến thể mở rộng của ResNet. DenseNet sẽ khác so với ResNet đó là chúng không cộng trực tiếp x và f(x) mà thay vào đó, các đầu ra của từng phép ánh xạ có dùng kích thước dài và rộng sẽ được concatenate với nhau thành một khối theo chiều sâu. Thành phần chính của DenseNet là các khối

DenseBlock và các tầng chuyển tiếp (translation layer). Một khối DenseBlock sẽ làm tăng thêm số lượng channel, nhưng việc thêm quá nhiều channel sẽ tạo nên một mô hình quá phức tạp. Do đó để giảm chiều dữ liệu chúng ta áp dụng tầng chuyển tiếp (translation layer). Tầng này dùng một tầng tích chập 1 x 1 để giảm số lượng channel, theo sau là tầng Pooling giúp giảm kích thước dài và rộng nhằm giúp giảm độ phức tạp của mô hình

Nhóm chúng tôi sử dụng mạng DenseNet để rút trích đặc trưng của dữ liệu kết quả thu được là các feature map. Kế đến các feature map này sẽ được đưa vào khối Channel attention. Channel Attention được thêm vào để cải thiện hiệu quả phân lớp của mô hình học sâu. Một bản đồ chú ý (attention map) sẽ được tạo ra bằng các khai thác các mối quan hệ giữa các kênh của các đối tượng. Mỗi kênh của feature map được xem như là một feature detector, nên channel attention sẽ tập trung vào những điểm có ý nghĩa của hình ảnh đầu vào nhằm xem xét phần thông tin nào quan trọng để nhấn mạnh hoặc phần thông tin nào ít quan trọng thì sự chú ý đối với chúng sẽ giảm đi.

Mô hình đề xuất thứ hai: EfficientNet

Các mạng Nơ ron tích chập được phát triển với tham số ban đầu là cố định, sau đó sẽ có thể tăng độ chính xác nếu đầu vào có được tham số lớn. Nhưng khi ta tăng kích thước của model đến một ngưỡng nào đấy thì độ chính xác trên tập dữ liệu sẽ bão hòa hoặc sẽ giảm. Đi kèm với sự gia tăng kích thước đó, chúng ta cần nhiều tài nguyên hơn để huấn luyện.

Nhận thấy các vấn đề trên, nhóm tác giả của phương pháp EfficientNet đã nghiên cứu và nhận thấy việc thu phóng cân bằng một cách có hệ thống (compound scaling) độ sâu, chiều rộng, độ phân giải (network depth, width, resolution) của một mạng có thể mang đến hiệu suất tốt hơn. Đây cũng một phướng phát triển mới cho các mô hình học sâu để tăng độ chính xác và cải thiện hiệu suất, minh chứng cho điều đó thì phương pháp này đã đạt được các thứ hạng cao trên các tập dữ liệu lớn như ImageNet.

Ngoài việc áp dụng cơ chế thu phóng mô hình, kiến trúc EfficientNet còn đặc biệt ở chỗ gồm nhiều khối Mobile inverted bottleneck. Đối với các mô hình thì độ

sâu quá lớn là một trong những nguyên nhân quan trọng dẫn đến việc số lượng tham số mô hình tăng cao. Khối MB Conv đã sử dụng tích chập tách biệt theo chiều sâu (depthwise separable convolution) giúp giảm lượng tham số đầu vào, chi phí tính toán cho mô hình.

Dữ liệu đầu vào sẽ được đưa qua một khối Preprocessing có nhiệm vụ chuẩn hóa dữ liệu cho phù hợp với đầu vào của mô hình EfficientNet. Tiếp đó, dữ liệu đã chuẩn hóa sẽ đi qua mạng EfficientNet và đầu ra sẽ được đưa qua hàm softmax và cho ra kết quả Positive hoặc Negative.

2.5 Kết quả dự kiến của đề tài

Kết quả dự kiến của đề tài: xây dựng được mô hình phân loại với độ chính xác cao từ các phương pháp đã đề xuất. Kết quả mô hình là một phiên bản tốt hơn về độ chính xác so với mô hình mà nhóm dựa vào để nghiên cứu từ những tác giả trước đó.

Sản phẩm đầu ra là 1 ứng dụng có chức năng như nhận ảnh đầu vào mà người dùng cần phân loại, sau đó trả về kết quả là Positive/Negative (là khối u hoặc không là khối u). Ngoài ra, mô hình còn có thể triển khai trên web-app.

2.6 Kế hoạch thực hiện

Bảng mô tả kế hoạch thực hiện đề tài:

STT	Công việc	Mốc thời gian	Thành viên thực hiện
1	Đọc và tìm hiểu các paper liên quan đến đề tài	1/1 - 1/2	Cả nhóm
2	Đề ra phương pháp tiếp cận Tìm hiểu các phần lý thuyết liên quan	1/2 - 1/3	Cả nhóm
3	Cài đặt mô hình bằng kỹ thuật Transfer learning Viết đề cương	1/3 - 1/4	Cả nhóm
4	Tìm hiểu về cơ chế attention Cài đặt mô hình	1/4 - 1/5	Cả nhóm
5	Tổng hợp kết quả Viết và chỉnh sửa báo cáo	1/5 - 1/7	Cả nhóm

Tài liệu

[1] K. D. Lai, T. T. Nguyen, and T. H. Le, "Detection of Lung Nodules on CT Images based on the Convolutional Neural Network with Attention Mechanism,"

Published by International Association of Educators and Researchers (IAER), 2021.

[2] A. Takimoglu, "Top data augmentation techniques: Ultimate guide for 2022." https://research.aimultiple.com/data-augmentation-techniques/.

[3] P. Baheti, "A newbie-friendly guide to transfer learning." https://www.v7labs.com/blog/transfer-learning-guide.

[4] Nttuan8, "Transfer learning data augmentation." https://nttuan8.com/bai-9-transfer-learning-va-data-augmentation/.

[5] M. Tripathi, "Image processing using cnn: A beginners guide." https://www.analyticsvidhya.com/blog/2021/06/image-processing-using-cnn-a-beginners-guide/.

XÁC NHẬN
CỦA NGƯỜI HƯỚNG DẪN
(Ký và ghi rõ họ tên)

TP. Hồ Chí Minh, ngày 4 /tháng 4 /năm 2022
NHÓM SINH VIÊN THỰC HIỆN
(Ký và ghi rõ họ tên)

Mục lục

Danh sách hình

Danh sách bảng

Tóm tắt

Hiện nay, ung thư phổi là một căn bệnh rất nguy hiểm và khó chữa trị, ảnh hưởng rất nhiều tới sức khỏe và cuộc sống của người bệnh. Có rất nhiều nghiên cứu được đặt ra về căn bệnh ung thư phổi, nhiều cách chẩn đoán và chửa trị được nghiên cứu từ các lĩnh vực khác nhau như y tế, hóa học, sinh học, ... Đặc biệt là việc ứng dụng các mô hình học sâu trong việc phân loại các nốt sần ở phổi. Điều này giúp cho các bác sĩ tiết kiệm được thời gian hơn và tăng độ chính xác khi chẩn đoán trên ảnh y khoa. Trong nội dung đồ án này, chúng tôi tiến hành nghiên cứu và đề xuất ra các mô hình phân loại ảnh có khối u hoặc không có khối u ở phổi. Nhóm chúng tôi xem xét và thử nghiệm trên hai mô hình. Mô hình thứ nhất, chúng tôi sử dụng mạng DenseNet kết hợp với khối Channel attention module. Mô hình thứ hai, chúng tôi sử dụng mạng EfficientNet B7, một trong những mạng phân loại khá hiện đại hiện nay. Kết quả tốt nhất chúng tôi thu được trên các độ đo như sau: **Precision: 0.950, Recall: 0.8759, Specificity: 0.9903**. Chúng tôi hy vọng việc áp dụng mô hình này trong lĩnh vực y tế kết hợp với kinh nghiệm của các bác sĩ, có thể tiết kiệm được thời gian trong việc chẩn đoán các nốt sần ở phổi có phải là khối u hay không và đưa ra các kết luận có tính chính xác cao hơn. Qua đó, có thể kịp thời chữa trị cho bệnh nhân hoặc giảm chi phí khám chữa bệnh trong việc chẩn đoán nhầm.

Chương 1

Giới thiệu

1.1 Lý do chọn đề tài

Đối với mỗi con người chúng ta thì sức khỏe là thứ quý giá nhất và quan trọng nhất. Các nguyên nhân ảnh hưởng đến sức khỏe hầu hết là do thói quen ăn uống và sinh hoạt của mỗi người. Nếu bạn có một chế độ dinh dưỡng tốt, cùng với việc luyện tập thể dục thể thao thường xuyên chắc chắn bạn sẽ có sức khỏe tốt. Và ngược lại, nếu chế độ ăn uống không lành mạnh, lười vận động, thường xuyên sử dụng các chất kích thích như: rượu, bia, thuốc lá, ... sẽ làm cho sức khỏe chúng ta giảm sút nghiêm trọng. Dẫn đến việc thường xuyên mắc bệnh và nguy hiểm nhất đó chính là bệnh ung thư.

Theo các tổ chức y tế thế giới, ung thư phổi là một trong những loại ung thư phổ biến nhất trên thế giới hiện nay. Triệu chứng thường xảy ra như ho, tức ngực hoặc ho ra máu. Tuy nhiên, cũng có rất nhiều trường hợp bệnh nhân không có bất kỳ triệu chứng lâm sàng nào hoặc khi có các triệu chứng thì ung thư đã vào giai đoạn cuối. Hiện nay, chúng ta có thể phát hiện các bệnh liên quan tới phổi, đặc biệt là ung thư bằng biện pháp chụp cắt lớp phổi (chụp CT phổi). Biện pháp này giảm đáng kể tỷ lệ tử vong do ung thư phổi nhờ việc phát hiện sớm và điều trị kịp thời.

Hiện nay với sự phát triển mạnh mẽ của trí tuệ nhân tạo, việc chẩn

đoán, phát hiện các bệnh, các biểu hiện bất thường trên ảnh y khoa trở nên thuận lợi hơn. Ta có thể thu được kết quả tốt trong việc giải quyết các bài toán trên thông qua việc sử dụng các phương pháp học máy như: K-Mean, KNN, SVM, ... hoặc sử dụng các mô hình Deep learning (mô hình học sâu) như Convolutional neural network (CNN - mạng nơron tích chập). Thế nên, trong đồ án này, nhóm sẽ tập trung nghiên cứu để giải quyết bài toán "Phân loại ảnh có khối u hoặc không có khối u ở phổi" bằng việc ứng dụng trí tuệ nhân tạo.

1.2 Mục đích nghiên cứu

Ngày nay, với sự phát triển nhanh chóng của công nghệ rất nhiều các thiết bị hiện đại được áp dụng vào lĩnh vực y tế như: hệ thống MRI 1.5 của Tesla, hệ thống CT Scanner 160 lát, hệ thống chụp mạch xóa nền (DSA), ... Và không thể không nhắc đến những ứng dụng của công nghệ thông tin trong y khoa như các ứng dụng theo dõi sức khỏe bệnh nhân, đặt lịch khám trực tuyến, hệ thống lưu trữ và truyền hình ảnh (PACS), ... Bằng việc áp dụng công nghệ thông tin khối lượng dữ liệu trong lĩnh vực y tế được sản sinh ra vô cùng đáng kể và ảnh y khoa là một trong số đó. Với một khối lượng lớn hình ảnh y khoa (MRI, CT, X-quang, ...) đòi hỏi các bác sĩ phải tốn thêm nhiều thời gian trong việc chẩn đoán các bệnh, các biểu hiện bất thường trên ảnh. Thế nên việc xây dựng một mô hình hỗ trợ các bác sĩ chẩn đoán, phát hiện các biểu hiện bất thường trên ảnh y khoa, với độ chính xác cao là hết sức cần thiết, từ đó có thể đưa ra phác đồ điều trị tốt nhất cho bệnh nhân. Sau mỗi lần chụp CT, X-quang, ... các bác sĩ không còn loay hoay với việc xem các ảnh phim và bệnh nhân không còn tốn quá nhiều thời gian để chờ kết quả vì đã có các mô hình máy tính giúp ta làm những công việc trên.

Mục tiêu cụ thể của nhóm trong đồ án này là "Xây dựng mô hình phân loại ảnh có khối u hoặc không có khối u ở phổi". Bài toán được đặt ra ở đây là bài toán phân loại ảnh (Image classification). Phương pháp mà

nhóm hướng đến là xây dựng mô hình phân lớp bằng mô hình học sâu. Ung thư là một trong những bệnh lý vô cùng nguy hiểm nếu không được phát hiện và điều trị kịp thời. Do đó, nhóm sử dụng các phương pháp nêu trên với hy vọng xây dựng một mô hình phân loại có độ chính xác cao, giảm tỷ lệ dương tính giả và âm tính giả của các mẫu được dự đoán.

1.3 Đối tượng nghiên cứu

Đề tài nghiên cứu lấy cảm hứng từ bài báo "Detection of Lung Nodules on CT Images based on the Convolutional Neural Network with Attention Mechanism" [12].

Tập dữ liệu dùng để nghiên cứu: LIDC/IDRI [6] . Tập dữ liệu này được dùng trong cuộc thi mang tên LUNA16. Mục đích của cuộc thi này nhằm giải quyết 2 vấn đề chính: vấn đề thứ nhất là xây dựng mô hình phát hiện các nốt sần ở phổi, sau khi xác định được vị trí các nốt sần ta sẽ đi đến việc giải quyết vấn đề thứ hai là xây dựng mô hình phân loại nốt sần nào có chứa khối u và không chứa khối u, nhằm giảm tỉ lệ dương tính giả. Nhóm tập trung nghiên cứu vào vấn đề thứ hai.

1.4 Phương pháp nghiên cứu

- Tìm hiểu về các phương pháp tăng cường dữ liệu.
- Xây dựng mô hình phân loại kết hợp giữa mạng DenseNet và Channel attention module.
- Xây dựng mô hình phân loại bằng cách sử dụng mô hình EfficientNet.
- Tìm hiểu các phương pháp đánh giá mô hình phân loại.

Trong đồ án này, nhóm sẽ nghiên cứu và sử dụng các mô hình học sâu, cài đặt mô hình để huấn luyện và phân lớp tập ảnh đầu vào và thực nghiệm

đánh giá kết quả dự trên bộ dữ liệu nhằm đánh giá hiệu năng của mô hình đã đề xuất.

1.5 Cấu trúc đề tài

Đề tại thực tập dự án tốt nghiệp này gồm 5 chương:

- Chương 1: Giới thiệu. Trong chương này nhóm sẽ trình bày một cách tổng quan về đề tài, mục tiêu nghiên cứu, phương pháp nghiên cứu về vấn đề mà đề tài tập trung giải quyết.
- Chương 2: Các công trình liên quan. Trong chương này nhóm sẽ giới thiệu về bài toán phân loại ảnh là gì, các công trình nghiên cứu trong và ngoài nước.
- Chương 3: Phương pháp đề xuất. Trong chương này nhóm sẽ trình bày cơ sở lý thuyết khi thực hiện đề tài, trình bày các mô hình mà nhóm đưa ra để giải quyết bài toán.
- Chương 4: Kết quả thực nghiệm. Trong chương này nhóm sẽ giới thiệu tập dữ liệu được sử dụng và đánh giá kết quả của các mô hình trên bộ dữ liệu.
- Chương 5: Kết luận và hướng phát triển.

Chương 2

Các công trình liên quan

2.1 Tổng quan về bài toán phân lớp

2.1.1 Khái niệm về bài toán phân lớp

Một trong những bài toán cơ bản nhất của lĩnh vực Khoa Học Máy Tính nói chung và Học Máy nói riêng, đó là bài toán phân lớp. Bài toán phân lớp là bài toán dự đoán sử dụng các đầu vào cho trước, có thể dự đoán các đầu vào của tương lai.
Thực tế đặt ra nhu cầu từ một cơ sở dữ liệu thông tin, con người có thể sử dụng để đưa ra quyết định thông minh dựa trên dữ liệu đã có. Công nghệ này cũng ứng dụng trong nhiều lĩnh vực khác nhau như thương mại, nhà băng, marketing, nghiên cứu thị trường, bảo hiểm, y tế, giáo dục...

2.1.2 Bài toán phân loại ảnh

Với sự phát triển của khoa học kĩ thuật, ngày càng có nhiều loại thiết bị và nhiều cách thức khác nhau để ghi lại hình ảnh hơn, điều này cũng dẫn đến nhu cầu về các tác vụ liên quan đến thị giác máy tính tăng lên trong những năm gần đây. Một trong các bài toán phổ biến nhất của thị giác máy tính đó là phân loại ảnh. Bài toán phân loại ảnh có mặt ở nhiều nơi ở trong đời sống của chúng ta, như nhận diện biển số xe, nhận biết đèn

xanh đỏ, phân loại ảnh y tế ...
Phân loại hình ảnh là tác vụ gán nhãn cho một nhóm các pixels hoặc véc-tơ trong một ảnh bằng các quy tắc cụ thể. Mục tiêu chính của bài toán phân loại ảnh là từ đầu vào là các ảnh, chúng ta có thể phân loại chúng ra theo như ý muốn. Ví dụ, với một tập dữ liệu hình ảnh các loại đồ vật cho trước, chúng ta có thể xây dựng được mô hình để đoán tên các hình ảnh tự động. Hiện nay phân loại ảnh đã và đang được nghiên cứu và ứng dụng rất rộng rãi, có nhiều phương pháp được nghiên cứu ra như dùng các mẫu nhị phân cục bộ, Fish Vector, SVM,.. Nhiều phương pháp học sâu có hiệu quả cao như: R-CNN, Resnet, VGG, Densenet...

2.1.3 Bài toán phân loại ảnh có khối u hoặc không có khối u ở phổi

Ung thư phổi là một trong những loại ung thư nghiêm trọng và phổ biến trên toàn thế giới, cả về số bệnh nhân mới và số trường hợp tử vong. ước tính có 1,8 triệu ca ung thư phổi mới trên toàn thế giới vào năm 2015 [19]. Tại Mỹ năm 2017, có tổng số 225.000 ca ung thư phổi mới và 155.870 ca tử vong được liệt kê, chiếm 26% tổng số ca tử vong do ung thư. .[BenhvienK] Tại Việt Nam ung thư phổi đứng hàng thứ nhất trong 10 loại ung thư thường gặp trên cả hai giới và là nguyên nhân gây tử vong hàng đầu. Nhưng nếu phát hiện sớm thì bạn có thể có phương pháp điều trị hiệu quả hơn rất nhiều. Ung thư phổi là loại ung thư thường gặp và ngày càng có xu hướng gia tăng. Gần đây, bệnh xuất hiện ở những người trẻ tuổi nhiều hơn, chiếm tỷ lệ khoảng 12% tổng số ung thư các loại tính chung trên toàn thế giới.
Khi một tế bào ở phổi phát triển và phân chia bất thường trái quy luật, có thể tích tạo thành các khối u. Các khối u này có thể là lành tính hoặc ác tính (ung thư). U lành tính ở phổi thường là những khối u phát triển chậm, không có dấu hiệu xâm lấn sang những tổ chức khác và không đe

dọa đến tính mạng người bệnh. Ngoài ra loại u này cũng có thể bắt nguồn từ việc biến đổi cấu trúc của phổi nhưng cũng không gây nguy hiểm. Một thuật ngữ khác cần được chú ý đến là nốt phổi. Hiện tượng này dễ dàng được quan sát trên các phim chụp X-quang hoặc CT, chúng thường đứng đơn lẻ một mình và cũng có khi hội tụ thành những đám san sát nhau. Theo Torre và các cộng sự của ông. [11], 55% tỷ lệ sống sót của bệnh nhân ung thư phổi có thể đạt được khi phát hiện tại chỗ (giai đoạn sớm). Do đó, cần phải kiểm tra và quan sát kỹ các khối u ở phổi khi chúng còn ở giai đoạn đầu.

Từ những điều đã đặt ra, nhóm sinh viên nhận thấy cần một giải pháp để hỗ trợ tăng độ chính xác của quá trình chẩn đoán các khối u ở phổi sử dụng các phương thức phân loại ảnh bằng học sâu. Bài toán được đặt ra là tìm cách giúp các bác sĩ có thể chẩn đoán khối u tốt hơn. Trước tiên các bác sĩ sẽ đánh dấu các khối u ở trên ảnh CT và sau đó được đưa qua mô hình của nhóm sinh viên để xác thực lại. Về bộ dữ liệu của bài toán sử dụng, bài toán sử dụng bộ dữ liệu LIDC/IDRI, gồm các ảnh CT và các bác sĩ đánh dấu các khối u trong phổi. Từ bộ giữ liệu này, chúng ta có thể huấn luyện được mô hình nhận biết cách để xác thực đâu chính xác là khối u, giúp cho các bác sĩ chẩn đoán khối u tốt hơn, giúp bệnh nhân nhận biết khối u hiệu quả qua đó tăng hiệu quả cho quá trình điệu trị của bệnh nhân và bệnh viện.

2.2 Tình hình nghiên cứu trong nước

Mạng tích chập hay tên tiếng anh là Convolutional Neural Network (gọi tắt là CNN) hiện đang là phương pháp tối ưu nhất để giải quyết các bài toán trong lĩnh vực hình ảnh. Đây là kiến trúc được thiết kế dựa trên thị giác sinh học, do đó rất có hiệu quả với các bài toán nhận diện ảnh, bất kể với kích thước hay tỉ lệ. Với nhiều phương pháp cải tiến và biến tấu khác nhau, CNN ngày càng được sử dụng rộng rãi hơn ở đa dạng lĩnh vực của ảnh.

Vào năm 2019, Giang Sơn Trần [8] cùng những cộng sự của mình đã sử dụng phương pháp học sâu CNN kết hợp với mất mát tiêu điểm (Focal Loss) để giải phân loại các ảnh có khối u ở phổi. Phương pháp này sử dụng các lớp học sâu sử dụng các khối tích chập và cuối cùng đi qua các tầng mất mát tiêu điểm. Qua đánh giá trên tập dữ liệu LIDC/IDRI kết quả thu được của mô hình này khá cao, với Accuracy 0,972 , Sensitivity 96%, Specificity 97,3%, AUC 98,2%.
Lai Đình Khải [12] và những công sự của mình đã đưa ra phương pháp khác để giải quyết bài toán đó là dùng mạng tích chập kết hợp với cơ chế đánh trọng (Attention Mechanism). Bằng phương pháp này, nhóm tác giả đã đạt kết quả rất ấn tượng với chỉ số auc là 0,992.

2.3 Tình hình nghiên cứu ngoài nước

Việc nghiên cứu về các khối u ở phổi là một vấn đề thú vị và mang lại rất nhiều lợi ích thiết thực cho nhân loại. Nghiên cứu này không chỉ được quan tâm tại Việt Nam, mà còn được sự quan tâm từ rất nhiều trường đại học và tổ chức trên thế giới. Cuộc thi LUNA16 [4] đề xuất ra một khung tiểu chẩn để đánh giá thuật toán phát hiện khối u tự động bằng cách sử dụng cơ sở dữ liệu lớn nhất công khai về ảnh chụp CT, bộ dữ liệu LIDC-IDRI [4]. Kết quả của cuộc thi này là mô hình phân loại đạt sensitivity ở mức 96,6%.
Kuruvilla và Gunavathi [10] đã đề xuất một phương pháp máy tính hỗ trợ để phân loại ung thư phổi bằng các ảnh CT. Các tác giả đã sử dụng các tham số thống kê như giá trị trung bình, độ lệch chẩn, độ lệch, kurtosis, thời điểm trung tâm thứ năm và thời điểm trung tâm thứ sáu làm các tính năng để phân loại. Mạng nơ-ron truyền ngược trở lại nguồn cấp dữ liệu được chứng minh là cho kết quả phân loại tốt hơn so với mạng nơ-ron truyền xuôi. Các thí nghiệm đã chứng minh rằng phương pháp này đạt độ chính xác 93,3 %, độ nhạy 91,4% và độ đặc hiệu là 100%. Kết quả cho

thấy, việc dự đoán các mẫu không phải là khối u rất chính xác.

Choi và Choi [7] đã giới thiệu phương pháp phát hiện nốt phổi bằng cách sử dụng phân loại khối theo cấp bậc (hierarchical block classification). Phương pháp này trước hết chia hình ảnh thành các khối ba chiều, sau đó áp dụng phương pháp phân tích entropy để chọn khối có thông tin hữu ích nhất. Cuối cùng, máy vectơ hỗ trợ (SVM) được sử dụng để phân loại các mẫu là khối u hay không. Đánh giá trên bộ dữ liệu LIDC cho thấy phương pháp này có độ chính xác rất tốt là 97,6%, độ nhạy là 95,2% và độ đặc hiệu là 96,2%.

Tại Trung tâm Y tế Đại học Radboud, Hà Lan, Setioet [3] sử dụng mạng tích chập đa chiều (ConvNets) để dự đoán các mẫu có khối u. Phương pháp này được huấn luyện trên bộ dữ liệu của cuộc thi LUNA16. Kết quả thực nghiệm thu được mức sensitivity là 90,1%.

Chương 3

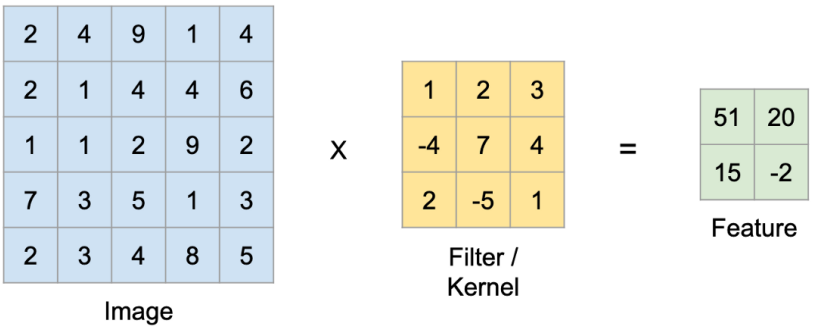
Phương pháp đề xuất

3.1 Cơ sở lý thuyết

3.1.1 Mạng nơ ron tích chập (Convolutional neural network)

Hay viết tắt là CNN là một giải pháp phổ biến cho các bài toán liên quan đến thị giác máy tính hiện nay. Ở một mức nào đó, ta có thể tưởng tượng chúng được lấy cảm hứng từ hoạt động của hệ thống thị giác của con người. Quá trình xử lí của CNN có thể được chia ra thành 5 bước: Convolution, Pooling, Flattening, Full Connection. Convolution là một thuật ngữ toán học nhằm đề cập đến một phép toán giữa hai ma trận. Là một khái niệm quan trọng và sử dụng nhiều nhất trong xử lý ảnh / thị giác máy tính. Chúng ta đơn giản nhân từng ma trận nhỏ của hình ảnh với một ma trận có tên là kernel, thông thường ma trận này là có kích thước là 3x3, 5x5.... Chúng ta sẽ có kết quả được gọi là Feature Map, nó sẽ có kích thước nhỏ hơn so với ảnh ban đầu. Kích thước này phụ thuộc vào các yếu tố như stride, padding, kích thước của kernel. Có một thắc mắc đó là kích thước sau khi đi qua nhỏ hơn sẽ có làm mất thông tin không, câu trả lời là không vì chúng ta sẽ sử dụng nhiều Feature Map. Convolution layer thực hiện phép toán tương quan chéo giữa đầu vào và kernel còn được gọi

với các cái tên khác như filter là một ma trận nhỏ có kích thước được định nghĩa trước. Trong quá trình huấn luyện mô hình chứa các tầng tích chấp, các kernel (filter) thường được tạo ngẫu nhiên, tương tự với các trọng số trong tầng kết nối đầy đủ (Fully Connected) Các kernel có thiết kế đặc biệt có thể xử lí hình ảnh cho các mục đích như làm mờ, sắc nét hay làm nổi bật các cạnh trong hình ảnh một cách hiệu quả và nhanh chóng. Các ứng dụng xử lý ảnh quen thuộc như photoshop chẳng hạn sẽ tìm ra các kernel đặc biệt cho các ứng dụng của mình. Trong các phương pháp cũ kernel được thiết kế thủ công dựa trên những kiến thức xử lý ảnh và thực nghiệm Mặc dù hiện nay có nhiều phương pháp mới được ra đời nhưng cốt lõi của deep learning trong xử lý ảnh đó là mạng convolutional neural network, các giá trị của kernel được học trong các lần lan truyền ngược sau khi tính toán hàm lỗi. Sau quá trình huấn luyện sẽ học được các bộ kernel tốt nhất cho việc trích xuất đặc trưng hình ảnh [5].

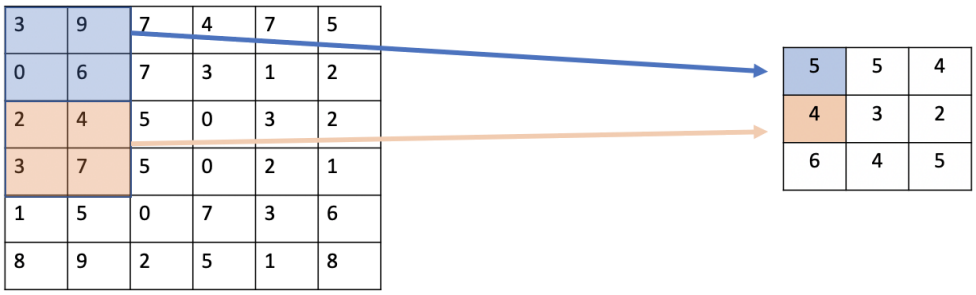


Hình 3.1: Convolution Operation. Nguồn: [18]

3.1.2 Lớp gộp (Pooling layer)

Trong mạng CNN, lớp pooling được áp dụng sau khi qua Convolutional layer. Mục đích chính của Pooling là giảm kích thước của các feature map, điều này sẽ giúp tăng tốc độ tính toán bởi vì số lượng tham số khi traning được giảm. Bên cạnh đó còn một mục đích mà em thấy rất ít tài liệu đề cập đó là Spatial Invariance. Ví dụ trong bức ảnh có một con mèo, ta có

thể hiểu con mèo có thể ở nhiều vị trí và nhiều hướng khác nhau trong bức ảnh. Pooling giúp mô hình có thể linh hoạt tìm ra con mèo ở vị trí nào. Bởi vì khi đi qua lớp Pooling, kết quả được giữ là các giá trị có gì trị lớn nhất hoặc là các giá trị được đại diện cho vùng đó. Điều này khá giống con người, bởi vì mắt con người không thể nhìn hết tất cả các vùng nhỏ nhặt hay nhiễu trong một bức ảnh, có xu hướng tập trung vào các đặc điểm chính và bỏ qua các chi tiết không quan trọng. Chúng ta có thể có nhiều loại Pooling khác nhau nhưng ở đây sẽ liệt kê 2 loại phổ biến là Max Pooling và Average Pooling sẽ lấy các giá trị lớn nhất và giá trị trung bình trong vùng cửa sổ. Trong cả hai trường hợp, giống như với Convolution, ta có thể xem cửa sổ bắt đầu từ phía trên bên trái và trượt từ trái sang phải và từ trên xuống dưới. Một điểm khá tương tự như Convolution nữa đó là padding và stride dùng để có thể thay đổi kích thước đầu ra. Với số lượng kênh đầu ra của Pooling thì sẽ bằng với số lượng kênh đầu vào.



Hình 3.2: Pooling Operation. Nguồn: [16]

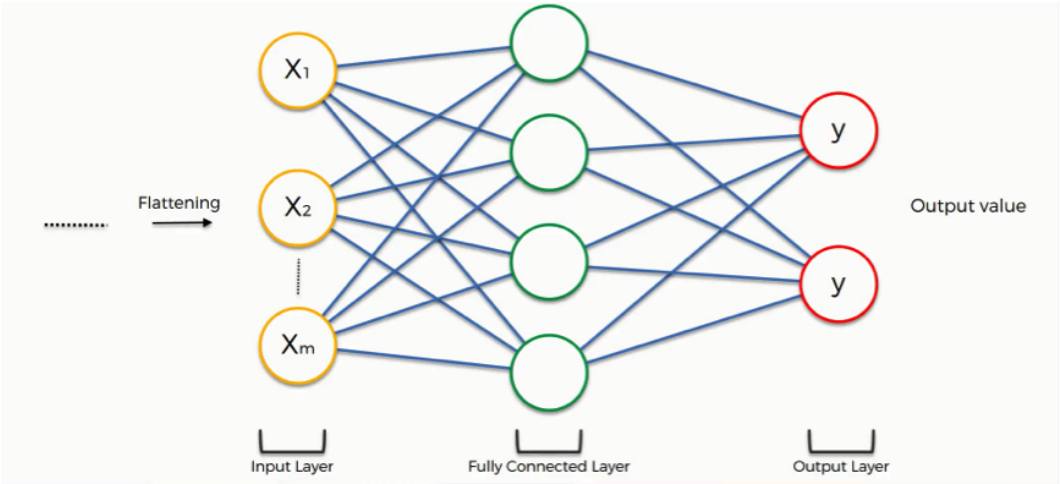
3.1.3 Hàm Relu

Đây là hàm phi tuyến được viết tắt của Rectified Linear Unit, đây là sự lựa chọn phổ biến nhất do tính đơn giản khi lập trình cũng như được chứng minh hiệu quả trong nhiều tác vụ. Hàm có công thức toán học là f(x) = max (0 ,x). Hay có thể hiểu một cách đơn giản, đó là hàm Relu

chỉ giữ lại các phần tử có trị dương và loại bỏ tất cả các phần tử có giá trị âm. Ngoài ra còn có một số biến thể của Relu như Leaky ReLU,ELU...

$$\text{ReLU}(z) = \max(z, 0). \tag{3.1}$$

3.1.4 Lớp kết nối đầy đủ (Fully connected layer)



Hình 3.3: Kiến trúc Fully Connected

Fully Connected nhận đầu vào là các dữ liệu đã được làm phẳng. Trong mô hành mạng CNNs, các tầng Fully Connected thường ở cuối mạng. Chúng kết hợp các features ở các bước trước một cách đa dạng nhắm tối ưu hóa độ chính xác của mạng.

3.1.5 Softmax Classifier

Softmax Classifier là một trong những thuật toán cơ bản và thường xuyên được sử dụng nhất để phân lớp các bài toán multi-class. Đặc biệt softmax còn thường được dùng để đưa ra output cuối trong các mạng neural network. Nhắc lại Logistic Regression, mặc dù tên nó có chứa Regression nhưng thật ra nó là một thuật toán phân loại 2 nhãn. Với các bài toán mà output nhiều hơn 2 nhãn, một số phương pháp được đưa ra như one

vs one, one vs rest... Các phương pháp này không được thống nhất vì ý tưởng là chúng ta sẽ so sánh xác suất đầu ra của từng lớp một so với tất cả các lớp còn lại, và nhiều khả năng tổng các xác suất này không bằng 1 [1].

$$\sigma(y_i) = \left(\frac{e^{y_i}}{\sum_j e^{y_j}} \right) j = 1, ..., n \tag{3.2}$$

Công thức của hàm sofmax giải quyết vấn đề trên, ngoài tổng xác suất các lớp bằng 1 thì chúng còn đảm bảo đánh mạnh vào các phần tử lớn hơn bằng việc exp.

3.1.6 Vấn đề quá khớp (Overfitting)

Đây là một hiện tượng khá phổ biến và rất dễ hay gặp với những người mới bắt đầu, lúc đó mô hình chỉ thể hiện tốt trên tập dữ liệu huấn luyện. Có nghĩa là mô hình sẽ học luôn các dữ liệu nhiễu, không bình thường trong quá trình xây dựng quy luật của mô hình. Những quy luật này quá khắt khe nên khi áp dụng cho một tập dữ liệu mới, mô hình sẽ hoạt động không tốt. Ta có thể nghĩ một ví dụ đơn giản trong đời sống như vấn đề học tủ khi thi cử, nếu đi thi chúng ta gặp một đề chúng ta đã được học thuộc thì sẽ được điểm cao. Nhưng nếu đề thi chỉ thay đổi vài con số so với đề ôn tập thì chúng ta sẽ làm sai. Để tránh việc xảy ra overfitting, có rất nhiều kĩ thuật được sử dụng như thu thập thêm dữ liệu, nếu không thể thu thập thêm chúng ta có thể sử dụng kĩ thuật tăng dữ liệu từ dữ liệu có sẵn như rotation, flip, skew... regularization, weight decay, dropout,early stoping...

3.1.7 Vấn đề chưa khớp (Underfitting)

Đây là hiện tượng xảy ra có thể nói là trái ngược so với Overfitting. Nó xảy ra khi mô hình quá đơn giản để có thể thể hiện mối quan hệ giữa input và output. Một số phương pháp để tránh Underfitting như giảm regularization, tăng thời gian training mô hình...

Việc train mô hình cũng giống như việc học trong đời sống, chúng ta không thể học tủ (overfitting) hay học qua loa (underfitting). Mà chúng ta cần phải học để có thể tổng quát hóa được kiến thức để có thể các trường hợp khác nhau.

3.1.8 Độ lỗi Binary cross entropy

Binary cross entropy (hay còn gọi là Log loss) sẽ so sánh từng xác suất được dự đoán với kết quả đầu ra thực tế của lớp được dự đoán, có thể là 0 hoặc 1. Sau đó, nó sẽ tính toán điểm phạt các xác suất dựa trên khoảng cách so với giá trị kỳ vọng, có nghĩa là gần hay xa giá trị thực tế. Log loss = -1N*i=1N(yi*log(pi)+(1-yi)*log(1-pi))

$$Logloss = -\frac{1}{N} * \sum_{i=1}^{n}(y_i * log(p_i) + (1 - y_i) * log(1 - p_i)) \tag{3.3}$$

Trong đó:

- N là tổng số mẫu quan sát.
- Yi là nhãn thực tế của mẫu thứ i.
- Pi là xác suất mẫu thuộc về lớp 1.
- 1 - pi là xác suất mẫu thuộc về lớn 0.

3.1.9 Thuật toán tối ưu Adam (Adaptive Moment Estimation)

Adam optimizer là một thuật toán tối ưu kết hợp giữa 2 kỹ thuật là RMSprop và momentum. Để dễ hiểu hơn ta sẽ hình dung theo hiện tượng vật lí thì momentum sẽ giúp hòn bi vượt qua các local minimum để tiến tới global minimum, nhưng khi gần đến đích thì nó lại dao động qua lại quanh vị trí đích rồi mới dừng hẳn, điều này có thể giải thích là do viên bi có quán tính. Còn đối với Adam như 1 viên bi nặng có ma sát, vì vậy nó dễ dàng vượt qua local minimum để đến global minimum và khi đến global minimum nó sẽ không tốn nhiều thời gian để dao động qua lại xung quanh điểm đích vì có ma sát nên dễ dừng lại hơn.[7]

3.1.10 Transfer learning (học chuyển giao)

Transfer learning là kỹ thuật khá quen thuộc trong lĩnh vực học sâu. Cụ thể hơn về Transfer learning, phương pháp này sẽ sử dụng kết quả/tri thức mà nó đã học được từ một vấn đề khác (pre-trained model) để áp dụng vào bài toán cần giải quyết hiện tại, tất nhiên sẽ là hai bài toán có liên quan đến nhau. Do đó kỹ thuật này giải quyết khá tốt trong những bài toán gặp vấn đề khan hiếm dữ liệu.

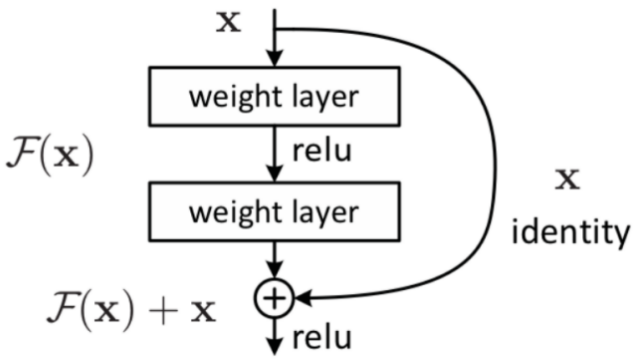
Ví dụ như nếu có một bài toán về nhận diện khối u phổi của một người khác làm với mô hình CNN, đồng thời họ đạt độ chính xác cao. Thì khi này ta áp dụng Transfer learning, nó sẽ tận dụng lại các đặc trưng được học từ mô hình của người đó với một mạng Based network (trích xuất một phần từ pre-trained model sau khi bỏ các top fully connected layers) mà có tác dụng trích lọc đặc trưng. Sau đó các lớp Fully Connected Layers giảm chiều dữ liệu, tính toán phân phối xác suất ở output.

Transfer learning có 2 loại: Feature extractor và Fine tuning. Sự khác biệt cơ bản giữa hai loại này là Fine tuning sau khi lấy ra các đặc trưng của ảnh thì sẽ thêm chi tiết (ví dụ như dùng ConvNet, để thêm những đặc

tính khác của bài toán nhận diện khuôn mặt, do khuôn mặt của mỗi vùng trên thế giới sẽ có những đặc điểm khác nhau). Còn Feature extractor sẽ lấy đặc trưng của ảnh và những đặc trưng đó sẽ trở thành input cho bài toán linear regression hay logistic regression.

3.2 Mạng ResNet

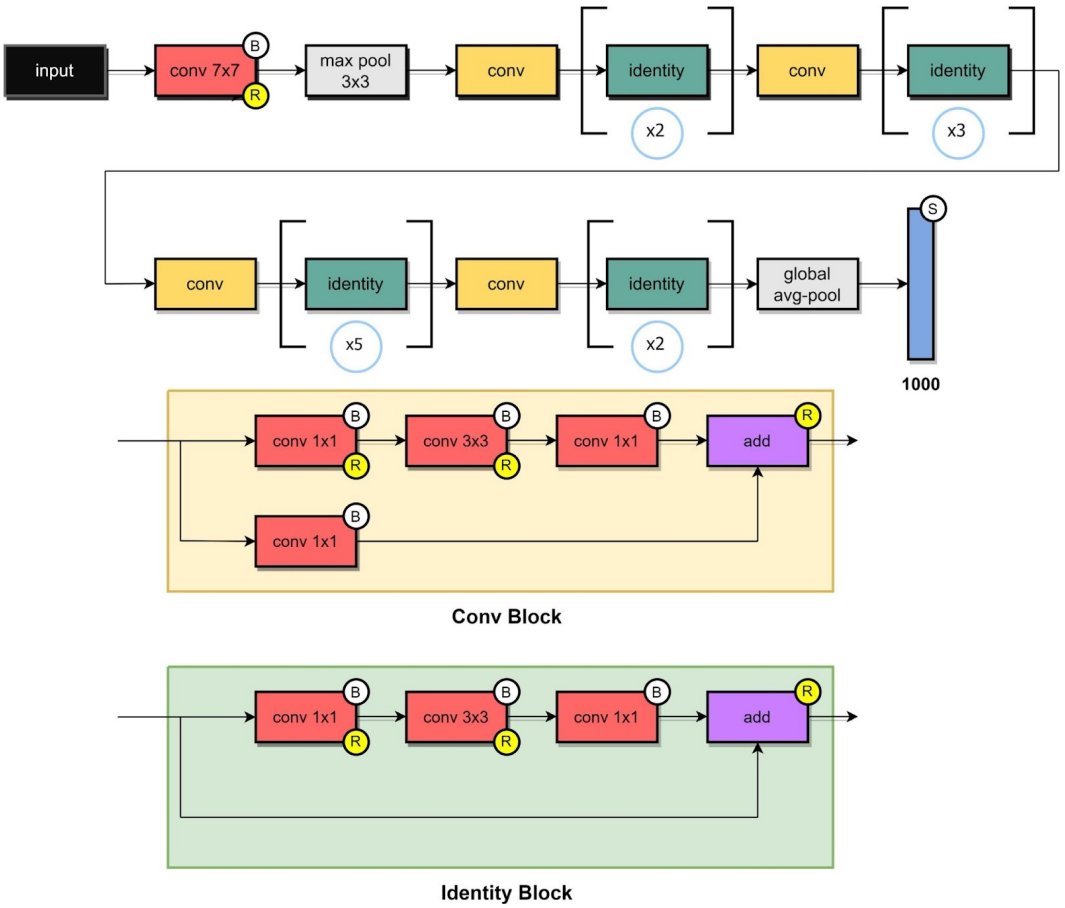
ResNet (Residual Network) là một trong những kiến trúc được sử dụng phổ biến nhất hiện nay. Những kiến trúc trước đây thường cải tiến độ chính xác bằng cách gia tăng chiều sâu của mạng CNN vì một mạng sâu có thể biểu diễn nhiều thuộc tính phức tạp. Tuy nhiên, mạng sâu rất khó huấn luyện vì gặp phải vấn đề về vanishing gradient. Đây là vấn đề xảy ra trong khi huấn luyện, giá trị đạo hàm là thông tin phản hồi của quá trình backpropagation. Trong thực tế các giá trị đạo hàm sẽ có giá trị nhỏ dần khi đi xuống các layer thấp hơn. Dẫn đến kết quả các cập nhật không làm thay đổi nhiều các trọng số của layer đó và làm chúng không thể hội tụ. Các nhà nghiên cứu của Microsoft đã giải quyết vấn đề này bằng cách sử dụng kết nối tắt (skip connection). [**resnetkientruc**]



Hình 3.4: Skip connection trong ResNet. Nguồn: [17]

Các kết nối tắt giúp giữ thông tin không bị mất bằng cách kết nối từ layer trước tới layer sau và bỏ các layer trung gian ở giữa, có nghĩa là đạo hàm có thể truyền trực tiếp thông qua skip connection từ layer trước đến layer sau bỏ qua một vài layer trung gian. Các kết này thường xuyên được

áp dụng trong các base network CNN của các mạng YOLO. Kiến trúc với ít tham số nhưng hiệu quả của ResNet đã mang lại chiến thắng trong cuộc thi ImageNet năm 2015. Kiến trúc ResNet gồm hai khối đặc trưng là khối tích chập (Conv Block) và khối xác định (Identity Block).[2]



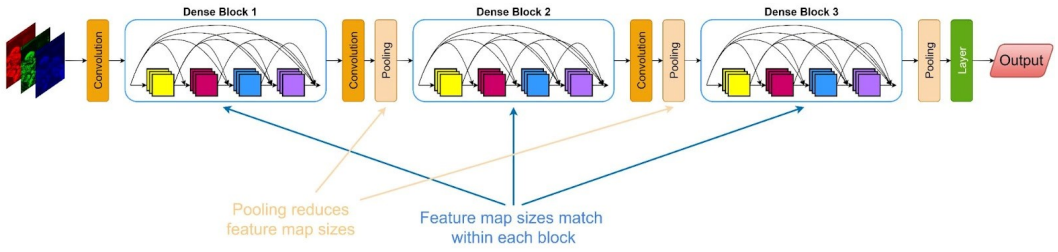
Hình 3.5: Kiến trúc ResNet. Nguồn: [2]

ResNet có khối tích chập (Conv block) sử dụng bộ lọc kích thước 3 x 3. Khối tích chập bao gồm 2 nhánh tích chập trong đó một nhánh áp dụng tích chập 1 x 1 trước khi cộng trực tiếp vào nhánh còn lại. Khối xác định (Indentity block) thì không áp dụng tích chập 1 x 1 mà cộng trực tiếp giá trị của nhánh đó vào nhánh còn lại. Sự khác nhau của hai khối trên đó, khối Conv block được sử dụng để bất cứ khi nào thì số chiều của input và output đều có thể bằng nhau để có thể cộng lại được với nhau. Có vài cách để thay đổi chiều của input như thay đổi giá trị stride. Ví dụ như trong một phép tính toán convolution, giả sử đầu vào là một ma

trận vuông. Thì số chiều của output có thể tìm được thông qua công thức (n+2p-f)/s+1, khi đó n là số chiều input, p là giá trị padding, f là số chiều của filter (kernel), s là stride. Từ công thức ta có thể thấy khi tăng giá trị stride có thể giảm chiều của output một cách hiệu quả. Do đó cần sử dụng khối Conv block để có sự bằng nhau giữa số chiều nhằm thực hiện phép add phía sau.[6]

3.3 Mạng DenseNet

DenseNet có thể được xem như là một biến thể mở rộng của ResNet. DenseNet sẽ khác so với ResNet đó là chúng không cộng trực tiếp x và f(x) mà thay vào đó, các đầu ra của từng phép ánh xạ có dùng kích thước dài và rộng sẽ được concatenate với nhau thành một khối theo chiều sâu. Thành phần chính của DenseNet là các khối DenseBlock và các tầng chuyển tiếp (translation layer). Một khối DenseBlock sẽ làm tăng thêm số lượng channel, nhưng việc thêm quá nhiều channel sẽ tạo nên một mô hình quá phức tạp. Do đó để giảm chiều dữ liệu chúng ta áp dụng tầng chuyển tiếp (translation layer)[2]. Tầng này dùng một tầng tích chập 1 x 1 để giảm số lượng channel, theo sau là tầng Pooling giúp giảm kích thước dài và rộng nhằm giúp giảm độ phức tạp của mô hình. Hình vẽ mô tả hoạt động của DenseNet:



Hình 3.6: Kiến trúc mạng DenseNet. Nguồn: [2]

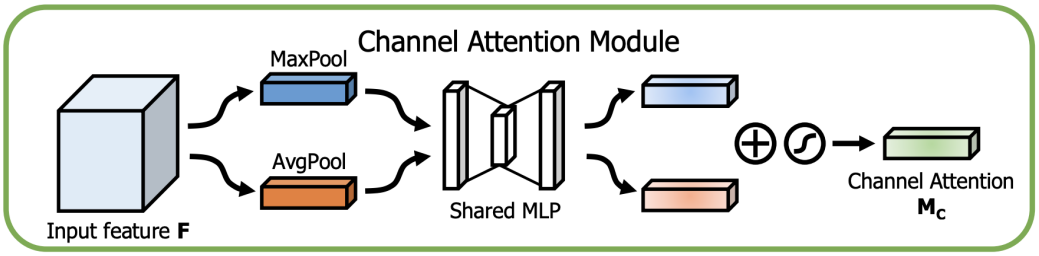
3.4 Channel Attention Module

Như ta đã biết Convolutional Block Attention Module có khả năng ứng dụng rộng rãi, đặc biệt với phân loại hình ảnh và phát hiện những vật thể.

CBAM gồm có hai module nhỏ hơn là: Channel Attention Module (CAM) và Spatial Attention Module (SAM).

Trong Convolutional Layer, input hình ảnh được coi là một tensor và đầu ra cũng sẽ là một tensor. Có 3 số liệu kích thước của tensor hình ảnh cần quan tâm là:

- h - chiều cao của mỗi feature map.
- w – chiều rộng của mỗi feature map.
- c – tổng số channels/tổng số feature maps/độ sâu của tensor.



Hình 3.7: Kiến trúc Channel attention module. Nguồn ảnh: [20]

Channel Attention Module (CAM) được tạo ra bằng cách khai thác mối quan hệ giữ các channel của các feature.

Đầu tiên, để tạo được CAM, cần tổng hợp thông tin không gian của Feature map bằng cách sử dụng hai pooling là average-pooling và max-pooling, tiếp đó hai biến khác nhau sẽ được tạo để mô tả không gian: F^c_{avg} và F^c_{max} biểu thị cho các feature average-pooling và max-pooling. Sau đó, cả hai biến được đẩy đến một mạng chung (shared network) để tạo ra Channel Attention Map là $M_c \in R^{C\times1\times1}$. Trong đó C là số lượng Channel.

Mạng chung cũng bao gồm nhiều MLP (multi-layer perceptron) với một hidden layer. Bên cạnh đó để giảm chi phí tham số, hidden activation size được gán bằng $R^{C/r\times1\times1}$, với r là tỉ lệ giảm.

Sau khi áp dụng mạng chung cho mỗi biến mô tả ở trên, tiến đến việc hợp nhất các vector feature bằng phép toán element-wise summation. Công thức chung để tạo nên Channel Attention sẽ được tính toán như sau:

$$\begin{aligned} M_c(F) &= \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \\ &= \sigma(W_1(W_0(F^c_{avg})) + W_1(W_0(F^c_{max}))) \end{aligned} \tag{3.4}$$

Trong đó, σ là hàm sigmoid, $W_0 \in R^{C/r \times C}, W_1 \in R^{C \times C/r}$

Channel Attention Module (CAM): giúp mô hình chú ý đến các đặc điểm quan trọng của đối tượng, đánh trọng số trên feature map nhằm phân loại xem thông tin nào quan trọng hay không quan trọng nhằm mục đích nhấn mạnh hoặc loại bỏ thông tin đó. Từ những việc trên, CAM giúp tăng hiệu suất tổng thể cho cả mô hình. Ngoài ra, CAM cũng là một module nhẹ và có thể dễ dàng kết nối với các network khác.

Lí do sử dụng cả CAM và SAM: CAM sẽ cho biết Feature map nào là quan trọng trong việc dùng để học và tăng cường. Trong khi đó SAM sẽ tiếp cận sâu hơn bên trong feature map cái nào sẽ cần thiết để học.

Tuy nhiên kết quả của đồ án khi thực hiện trên cả CAM và SAM cho kết quả khá kém. Có thể lý do ảnh đã detect cụ thể vùng cần phân loại, ảnh đã có kích thước rất nhỏ nhưng vẫn đánh trọng số dẫn đến ảnh thu được không còn rõ ràng, gây ra kết quả kém.

3.5 EfficientNet

3.5.1 Giới thiệu

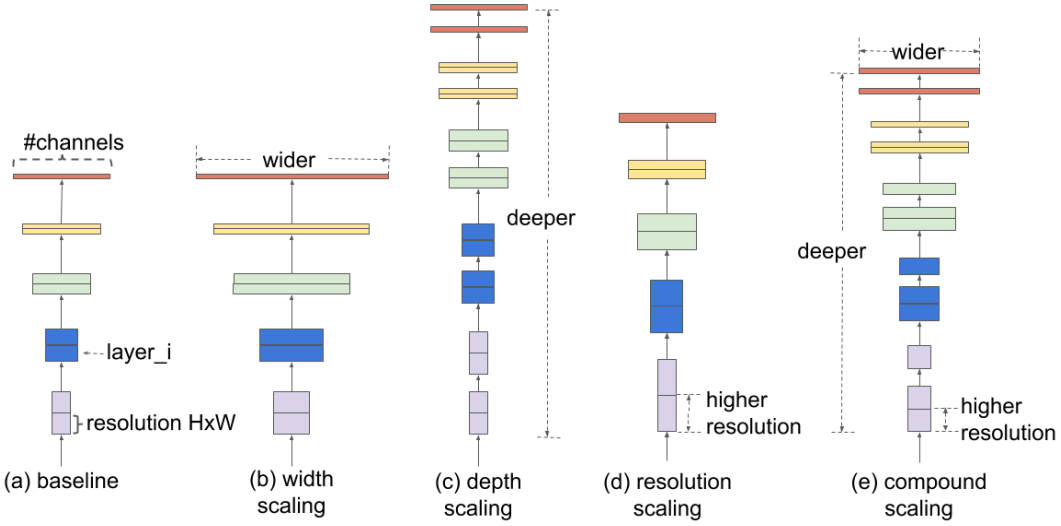
Các mạng Convolutional Neural Networks hay mạng Nơ ron tích chập (CNN) được phát triển với tham số ban đầu là cố định, sau đó sẽ có thể tăng độ chính xác nếu đầu vào có được lượng tham số đủ lớn. Nhưng khi ta tăng kích thước của model đến một ngưỡng nào đấy thì độ chính xác

Architecture	Year	Accuracy	Parameters
AlexNet	2012	56.55%	62M
GoogleNet	2014	74.8%	6.8M
SENet	2017	82.7%	145M
GPipe	2018	84.3%	557M

Hình 3.8: Độ chính xác và số lượng tham số của các mô hình. Nguồn ảnh: [13]

trên tập dữ liệu sẽ bão hòa hoặc sẽ giảm. Đi kèm với sự gia tăng kích thước đó, chúng ta cần nhiều tài nguyên hơn để huấn luyện cho mô hình.

Vì vậy, nhóm tác giả của phương pháp EfficientNet đã nghiên cứu và nhận thấy việc thu phóng cân bằng một cách có hệ thống (compound scaling) độ sâu, chiều rộng, độ phân giải (network depth, width, resolution) của một mạng có thể mang đến hiệu suất tốt hơn. Cụ thể hơn về việc thu phóng mô hình có thể xem ở hình sau:



Hình 3.9: Model Scaling - Mô tả mô hình được thu phóng theo các chiều khác nhau. Nguồn ảnh: [14]

Thu phóng theo chiều rộng (Width scaling): Ta có thể thực hiện Width

scaling bằng cách nạp thêm dữ liệu vào, cho phép các lớp tìm hiểu các feature chi tiết hơn, kĩ càng hơn. Tuy nhiên, tăng chiều rộng sẽ cản trợ mạng học các feature phức tạp, dẫn đến giảm độ chính xác.

Thu phóng theo chiều sâu (Depth scaling): là cách thông dụng nhất được sử dụng để thu phóng một mô hình CNN. Độ sâu có thể được thu phóng cũng như thu nhỏ bằng cách thêm hoặc bớt các lớp tương ứng. Như DenseNet121 (121 layers) tăng số lớp lên đến DenseNet201 (201 layers). Cũng như thu phóng theo chiều rộng, thu phóng theo chiều sâu cũng tăng sự tìm hiểu các feature. Tuy nhiên dùng quá nhiều tầng cho tập dữ liệu không thích hợp thì sẽ gây ra Over-fitting, kèm theo đó là hiện tượng vanishing gradients, khó đào tạo mô hình.

Thu phóng theo độ phân giải (Resolution Scaling): Độ phân giải ảnh cao thì sẽ cho thấy được nhiều chi tiết của ảnh hơn. Tuy nhiên áp dụng chỉ mỗi thu phóng theo độ phân giải thì cũng sẽ bị các hiện tượng tiêu cực như hai cách thu phóng trên.

3.5.2 Compound Model Scaling

3.5.2.1 Công thức hóa vấn đề

Với mỗi lớp ConvNet, có thể định nghĩa là một hàm. Ví dụ lớp i có hàm là $Y_i = F_i(X_i)$.

Trong đó Y_i là output tensor, F_i là toán tử, X_i là input tensor, với kích thước của Tensor được định nghĩa gồm $\langle H_i, W_i, C_i \rangle$, trong đó H_i và W_i là spatial dimension, còn C_i là channel dimension. ConvNet có thể được biểu diễn như sau:

$$N = F_k \odot F_{k-1} \odot F_{k-2} \odot \ldots \odot F_2 \odot F_1(X_1) = \bigodot_{j=1\ldots k} F_j(X_1) \qquad (3.5)$$

Các lớp ConvNet trong thực tế thường sẽ được phân chia thành nhiều stage và tất cả các lớp ở mỗi stage đều sẽ có kiến trúc giống nhau, ví dụ: ResNet có 5 stages và tất cả các lớp mỗi stage có cùng một kiểu phức hợp như

nhau, chỉ có lớp đầu tiên thực hiện giảm tần số lấy mẫu (Down Sampling). Do đó, chúng ta có thể định nghĩa Mạng ConvNet như sau:

$$N = \bigodot_{i=1\ldots s} F_i^{L_i}(X_{\langle H_i, W_i, C_i \rangle}) \tag{3.6}$$

Trong đó:

$F_i^{L_i}$ biểu thị cho lớp F_i mà được lặp lại L_i lần trong stage i,

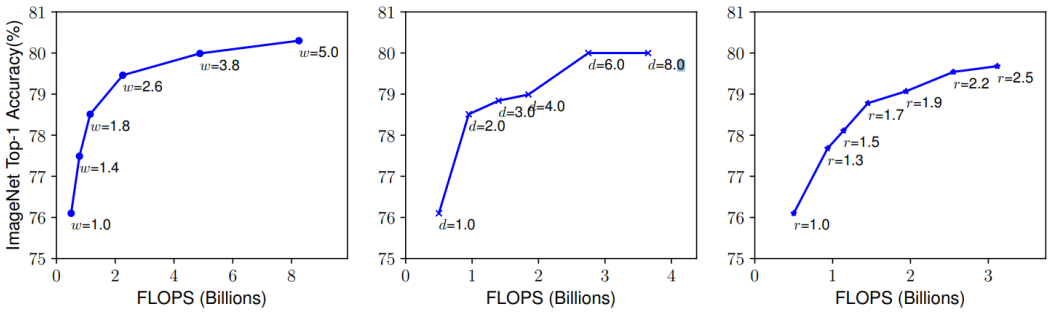
$\langle H_i, W_i, C_i \rangle$ biểu thị cho kích thước của tensor đầu vào X của lớp i.

Không giống như các thiết kế ConvNet thông thường mà chủ yếu tập trung vào việc tìm kiếm kiến trúc tốt nhất F_i, việc thu phóng mô hình cố gắng mở rộng chiều dài (L_i), chiều rộng (C_i) và/hoặc độ phân giải (H_i, W_i) riêng lẻ mà không thay đổi F_i được xác định trước đó trong mạng cơ sở. Bằng cách giữ nguyên hàm F_i, việc thu phóng mô hình sẽ giảm độ phức tạp cho vấn đề về thiết kế khi mà tài nguyên hạn chế. Tuy nhiên, còn rất nhiều trường hợp có thế diễn ra bởi chúng ta có thể thay đổi cả 3 chiều của mỗi lớp và mỗi chiều thì có mỗi mức độ khác nhau. Để giảm không gian cần phải tìm kiếm, nhóm tác giả hạn chế tất cả các lớp phải được thu phóng đồng nhất với tỷ lệ không đổi. Mục tiêu là tối đa hóa độ chính xác của mô hình cho bất kỳ hạn chế tài nguyên nào, vấn đề này có thể được xem là vấn đề phải tối ưu hóa như sau:

$$\begin{aligned} &\max_{d,w,r} \quad Accuracy((d, w, r)) \\ &s.t.(d, w, r) = \bigodot_{i=1\ldots s} \hat{F}_i^{d \cdot \hat{L}_i}(X_{\langle r \cdot \hat{H}_i, r \cdot \hat{W}_i, w \cdot \hat{C}_i \rangle}) \\ &Memory(N) \leq \texttt{target\_memory} \\ &FLOPS(N) \leq \texttt{target\_flops} \end{aligned} \tag{3.7}$$

Trong đó: w, d, r lần lượt là hệ số để chia chiều rộng, chiều sâu và độ phân giải của mạng. $\hat{F}_i, \hat{H}_i, \hat{W}_i, \hat{C}_i, \hat{L}_i$ đã được định nghĩa trước đó trong mạng cơ sở.

Nhóm tác giả thử thu phóng mô hình theo một chiều nhất định, nhận



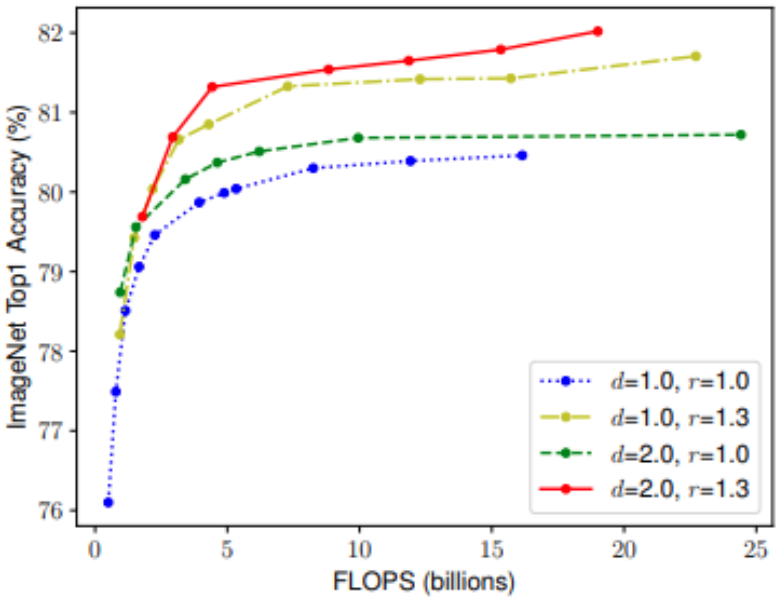
Hình 3.10: Accuracy và FLOPS khi thu phóng mô hình theo một chiều. Nguồn ảnh: [14]

thấy với những mạng lớn, độ thu phóng lớn theo một chiều thì độ chính xác tăng mạnh. Tuy nhiên tới ngưỡng 80% thì nhanh chóng bão hòa không tăng nữa. Nhận thấy điểm yếu của thu phóng mạng theo một chiều nhất định, do vậy, nhóm tác giả đã phối hợp và cân bằng các kích thước tỷ lệ khác nhau, thay vì chỉ là chia thu phóng theo tỉ lệ một chiều.

3.5.2.2 Compound Scaling

Bằng cách phối hợp và cân bằng các kích thước tỷ lệ khác nhau, thay vì chỉ là chia thu phóng theo tỉ lệ một chiều.

Từ đó nhận thấy để đạt được độ chính xác và hiệu quả tốt hơn, và điều quan trọn hơn cả là phải cân bằng được kích thước của chiều rộng, chiều sâu và cả độ phân giải của mạng khi thu phóng cho ConvNet. Nhóm tác giả đề xuất một phương pháp thu phóng phức hợp mới, sử dụng hệ số phức hợp để đồng nhất được chiều rộng, chiều sâu và độ phân giải theo một nguyên tắc nhất định:



Hình 3.11: So sánh việc thu phóng theo chiều rộng và theo các độ sâu cũng như độ phân giải mạng ở các mức độ khác nhau. Nguồn ảnh: [14]

$$
\begin{aligned}
&depth : d = \alpha^{\phi} \\
&width : w = \beta^{\phi} \\
&resolution : r = \gamma^{\phi} \\
&s.t. \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\
&\alpha \geq 1, \beta \geq 1, \gamma \geq 1
\end{aligned}
\tag{3.8}
$$

Trong đó:

α, β, γ là các hằng số mà có thể xác định được bằng small grid search.

ϕ là hệ số do người dùng sẽ chỉ định để kiểm soát số lượng tài nguyên khác có sẵn để thu phóng mô hình.

Trong khi α, β, γ là cách mà ta gán tài nguyên bổ sung này cho chiều rộng, chiều sâu và độ phân giải của mạng. Quan trọng hơn, FLOPS của một op tích hợp thông thường tỷ lệ với d, w^2, r^2, tức là nếu độ sâu mạng tăng gấp đôi thì sẽ kéo theo FLOPS cũng tăng gấp đôi, còn nếu độ rộng hoặc độ phân giải tăng gấp đôi thì sẽ tăng FLOPS lên bốn lần. Vì các hoạt động tích phân thường chiếm lượng lớn chi phí tính toán trong ConvNets, nên

việc thu phóng Mạng Conv với phương trình 3.8 sẽ làm tăng tổng FLOPS $(\alpha \cdot \beta^2 \cdot \gamma^2)^\phi$. Ràng buộc $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$ sao cho với bất kỳ ϕ mới nào, tổng FLOPS sẽ tăng xấp xỉ lên 2^ϕ.

3.5.3 Kiến trúc mạng EfficientNet

Nhóm tác giả phát triển mạng cơ sở của mình bằng cách tận dụng tìm kiếm kiến trúc nơron đa mục tiêu để tối ưu hóa cả hai giá trị Accuracy và FLOPS. Cụ thể hơn, nhóm tác giả sử dụng cùng một không gian với phương pháp được mô tả trong MnasNet: Platform-aware neural architecture search for mobile [15] và sử dụng $ACC(m) \times [FLOPS(m)/T]^w$ làm mục tiêu tối ưu hóa, trong đó $ACC(m)$ và $FLOPS(m)$ biểu thị Accuracy và FLOPS của mô hình m, T là mức FLOPS mục tiêu và w = -0,07 là hyper-parameter để kiểm soát sự cân bằng giữa Accuracy và FLOPS. Tại đây nhóm tác giả đã tối ưu hóa cho giá trị FLOPS thay vì là độ trễ bởi vì họ xác định sẽ thực hiện trên bất kỳ thiết bị phần cứng nào đó mà không phải một thiết bị cụ thể. Nhóm tác giả đã tạo ra một mạng mới gọi là EfficientNet-B0.

Stage i	Operator $\hat{\mathcal{F}}_i$	Resolution $\hat{H}_i \times \hat{W}_i$	#Channels $\hat{C}_i$	#Layers $\hat{L}_i$
1	Conv3x3	224×224	32	1
2	MBConv1, k3x3	112×112	16	1
3	MBConv6, k3x3	112×112	24	2
4	MBConv6, k5x5	56×56	40	2
5	MBConv6, k3x3	28×28	80	3
6	MBConv6, k5x5	14×14	112	3
7	MBConv6, k5x5	14×14	192	4
8	MBConv6, k3x3	7×7	320	1
9	Conv1x1 & Pooling & FC	7×7	1280	1

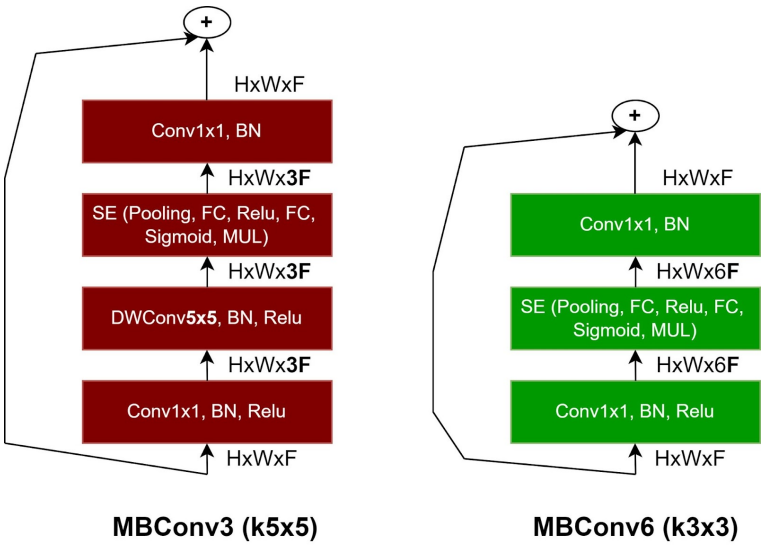
Hình 3.12: Bảng mô tả kiến trúc EfficientNet-B0 - Mỗi dòng mô tả một giai đoạn i với layer $\hat{L}_i$, với độ phân giải input $\langle \hat{H}_i, \hat{W}_i \rangle$ và output là các Channel $\hat{C}_i$. Nguồn ảnh: [14]

Ban đầu từ mô hình cơ sở EfficientNet-B0, nhóm tác giả áp dụng compound scaling để thu phóng mạng cơ sở từ đó có được EfficientNet-B1 đến EfficientNet-B7. Cụ thể với hai bước:

1. Đặt cố định giá trị của ϕ là 1, ta thu được bộ gồm các giá trị tối ưu là $\alpha = 1.2, \beta = 1.1, = 1.15$, theo sự ràng buộc của $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$.

2. Cố định những giá trị α, β, γ dưới dạng các hằng số và thu phóng mạng cơ sở với các giá trị ϕ khác nhau từ đó thu được EfficientNet-B1 đến EfficientNet-B7.

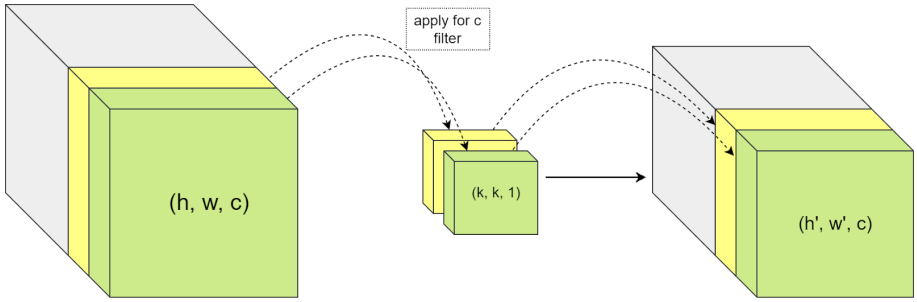
Điều đặc biệt ở kiến trúc EfficientNet là kiến trúc các khối mobile inverted bottleneck convolutional (MBConv).



Hình 3.13: kiến trúc các khối Mobile Inverted Bottleneck Convolutional (MBConv). Nguồn ảnh: [15]

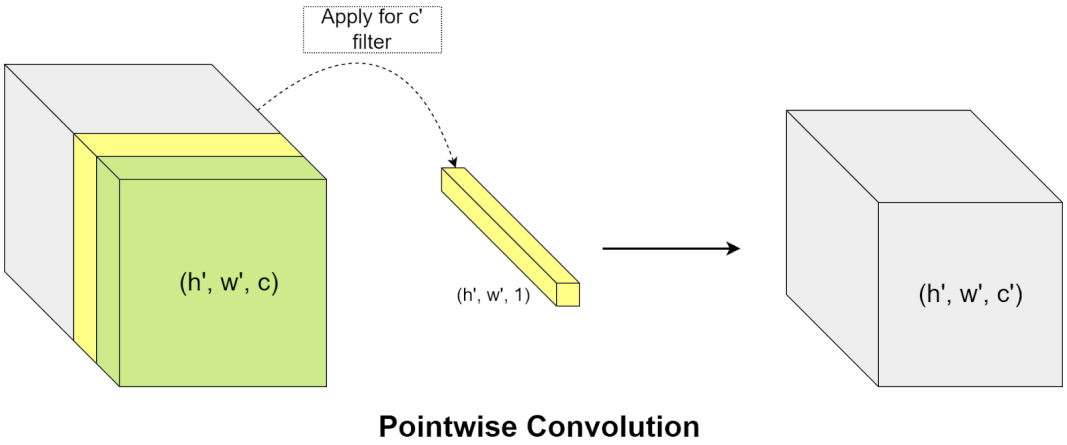
Đối với các mô hình, độ sâu là một trong những nguyên nhân quan trọng dẫn đến việc số lượng tham số mô hình tăng cao khi độ sâu lớn. Khối MB Conv đã sử dụng tích chập tách biệt theo chiều sâu (depthwise separable convolution). Tích chập này được thực hiện qua hai quá trình:

1. Tích chập theo chiều sâu (depthwise convolution): chia khối input tensor 3D thành những lát cắt ma trận theo độ sâu. Sau đó thực



Hình 3.14: Cách hoạt động của tích chập theo chiều sâu. Nguồn ảnh: [3]

hiện tích chập trên từng lát cắt. Mỗi channel sẽ áp một bộ lọc khác nhau và chúng đều hoàn toàn độc lập tham số với nhau. Điều này đem đến những tác dụng cho mô hình: giảm thiểu lượng tính toán, giảm thiếu số lượng tham số, nhận diện đặc trưng. Kết quả sau tích chập được nối lại với nhau (concatenate) lại theo độ sâu. Từ khối tensor 3D có kích thước (h, w, c) trở thành (h', w', c).

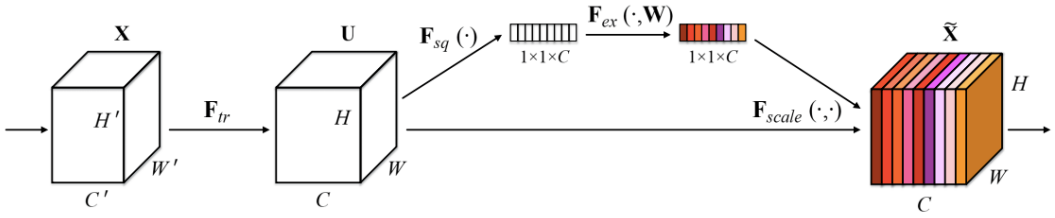


Hình 3.15: Cách hoạt động của tích chập điểm. Nguồn ảnh: [3]

2. Tích chập điểm (Pointwise Convolution): nhằm thay đổi độ sâu của output bước trên từ c thành c'. Chúng ta sẽ áp dụng bộ lọc kích thước $1 \times 1 \times c$. Giữ nguyên kích thước chiều rộng và chiều cao, chỉ độ sâu thay đổi. Số lượng của các tham số cần cho trường hợp này là $c' \times c$.

Tích chập tách biệt theo chiều sâu giúp giảm lượng tham số đầu vào, giảm chi phí tính toán cho mô hình.

Thêm vào đó, khối MBConV có một module gọi là SE, hay có tên đầy đủ là Squeeze-and-Excitation. Phiên bản MBConV đầu tiên chưa có module này, sau đó được thêm vào, mục đích của SE module nhằm để đưa ra trọng số cho từng kênh trong dữ liệu một cách phù hợp nhất, tăng cường thông tin giữa các kênh, bằng cách sử dụng toàn bộ thông tin sau đó nhấn mạnh có chọn lọc vào từng kênh có đặc trưng quan trọng và ít chú ý vào những kênh ít quan trọng hơn. SE khá tương đồng với self-attention, mục đích giúp mô hình đánh trọng số cho những phần quan trọng của bức ảnh, từ đó phân loại ảnh tốt hơn, cải thiện hiệu suất của mô hình.



Hình 3.16: Cách hoạt động của khối SE. Nguồn ảnh: [4]

Trong đó:

X: là ảnh đầu vào có kích thước $H' \times W' \times C'$

F_{tr}: tập hợp các phép biến đổi: một vài lớp convolution, 1 block trong ResNet, ...

U: feature map hay đặc trưng được trích xuất từ ảnh đầu vào bởi các phép biến đổi F_{tr}. U kích thước là $H \times W \times C$.

Cơ bản, ảnh đầu vào X đi qua một tập hợp các phép biến đổiF_{tr} trích xuất ra bản đồ đặc trưng (features map) U. Feature map U được đi qua hàm squeeze (global average pooling, ...) tạo ra một ma trận, ma trận đó sẽ miêu tả đặc trưng của từng kênh $(1 \times 1 \times C)$ bằng cách tổng hợp features map U theo hai chiều H và W. Tiếp đến, hàm excitation miêu tả mối liên hệ phụ thuộc giữa các kênh với nhau. Hàm nhận đầu vào là ma trận tổng hợp đặc trưng của từng kênh được tính toán từ bước 2 qua các

lớp biến đổi như convolution, hàm activation, ... cuối cùng là hàm gate để tạo ra trọng số chú ý cho từng kênh. Những trọng số này sẽ được nhân với U để tìm ra output của khối SE.

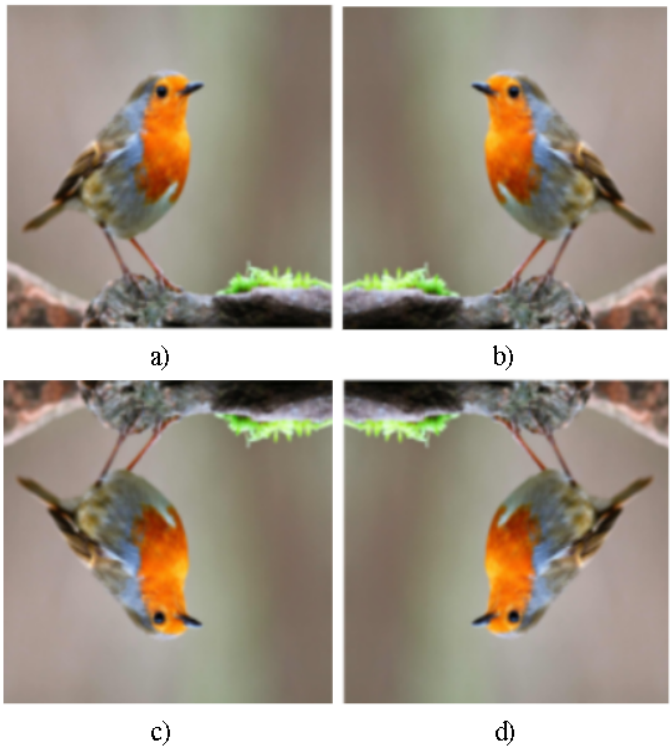
3.6 Mô hình đề xuất

3.6.1 Tăng cường dữ liệu

Trong thời đại công nghệ số hiện nay thì dữ liệu đóng vai trò khá quan trọng. Chẳng hạn như trong lĩnh vực bán hàng online, nếu ta có được càng nhiều dữ liệu của khách hàng thì việc phân tích thói quen, nhu cầu mua hàng của họ sẽ trở nên dễ dàng hơn. Từ đó, có thể đưa ra những chiến lược kinh doanh phù hợp nhằm tăng lợi nhuận cho doanh nghiệp. Trong lĩnh vực học sâu cũng vậy, nếu ta huấn luyện mô hình trên một bộ dữ liệu quá nhỏ, thì kết quả dự đoán khó có thể có độ chính xác cao. Thế nên, tăng cường dữ liệu là một khái niệm hết sức quan trọng để làm dồi dào nguồn dữ liệu đầu vào cho mô hình trong việc huấn luyện từ đó giúp mô hình có thể học tập được nhiều hơn và cho ra các kết quả dự đoán tốt hơn.

Trong học sâu có nhiều phương pháp tăng cường dữ liệu như: random crop (cắt ngẫu nhiên), noise addition (thêm nhiễu), contrast change (thay đổi độ tương phản), ... Có rất nhiều các phương pháp tăng cường dữ liệu khác nhau, ở đây nhóm chúng tôi sử dụng phương pháp tăng cường ảnh đơn giản là flip (lật).

Flip (lật): lật ảnh là một trong những phương pháp tăng cường dữ liệu đơn giản nhất để có thể cải thiện hiệu suất của mô hình. Thay vì nhìn vào đối tượng theo một hướng cụ thể, ta sẽ tiến hành lật ảnh theo chiều ngang hoặc chiều dọc hoặc kết hợp cả lật ảnh ngang và lật ảnh dọc từ đó sẽ tạo ra nhiều góc nhìn hơn về đối tượng. Từ đó cung cấp được nhiều thông tin hơn cho mô hình học sâu mà không cần trải qua quá trình thu thập và gán nhãn dữ liệu. Hình 3.17 một vài hình ảnh khi sử dụng các phương pháp lật ảnh.



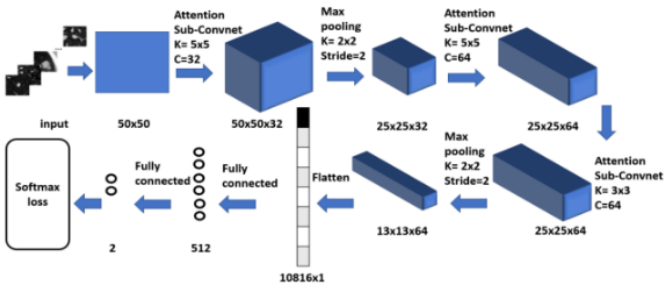
Hình 3.17: Kết quả thu được sau khi sử dụng lật ảnh (Hình a): ảnh gốc, hình b): áp dụng phương pháp lật ảnh ngang, hình c): áp dụng phương pháp lật ảnh dọc, hình d): áp dụng kết hợp lật ảnh dọc và lật ảnh ngang)

Hình 3.18: Lưu ý khi sử dụng phương pháp lật ảnh

Tuy nhiên, khi áp dụng phương pháp lật ảnh cho việc tăng cường dữ liệu cần xem xét kỹ đặc điểm của tập dữ liệu bạn đang sử dụng. Ví dụ, trong trường hợp bạn đang giải quyết bài toán nhận dạng các chữ số viết tay. Nếu áp dụng lật ảnh đối với số 6 thì nó có thể trở thành số 9, nên nhãn của chúng sẽ bị sai. Hình 3.18 minh họa một trường hợp cần lưu ý khi sử dụng phương pháp lật ảnh.

3.6.2 Mô hình đề xuất 1: DenseNet kết hợp Channel attention module

Lấy ý tưởng từ bài báo "Detection of Lung Nodules on CT Images based on the Convolutional Neural Network with Attention Mechanism" của nhóm tác giả Khai Dinh Lai, Thuy Thanh Nguyen và Thai Hoang Le [12] trong việc giải quyết vấn đề tương tự. Trong bài báo này, nhóm tác giả đã đề xuất ra một mô hình gồm các khối Attention sub-Convnet và Max pooling xếp chồng lên nhau. Khối Attention sub-Convnet do nhóm tác giả tự thiết kế sử dụng kết hợp mạng CNN và cơ chế attention. Các khối Attention sub-Convnet có nhiệm vụ rút trích các feature map và đánh trọng số trên các feature map bằng cơ chế attention. Hình 3.19 mô hình mà nhóm tác giả đề công bố.



Hình 3.19: Mô hình ASS (Attention sub-Convnet - Softmax - Softmax). Nguồn: [12]

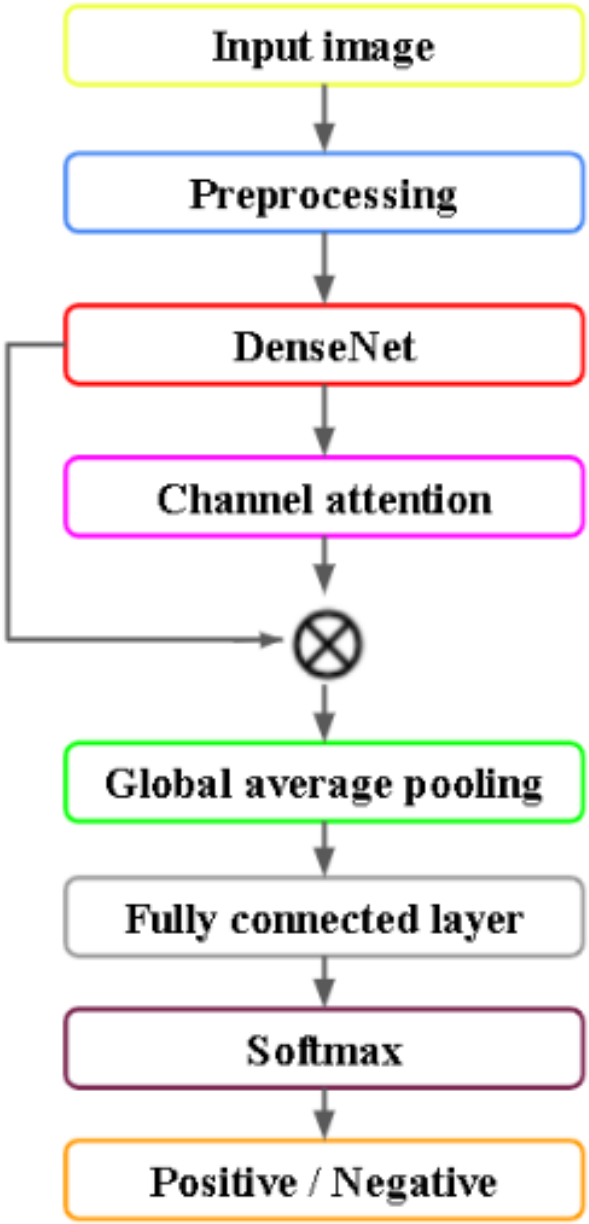
Từ bài báo trên, nhóm chúng tôi có ý tưởng thay vì dùng các Attention sub-Convnet xếp chồng lên nhau như trên, nhóm chúng tôi sẽ dùng một

mạng học sâu hiện đại để rút trích đặc trưng của dữ liệu, sau đó sẽ dùng attention đánh trọng số trên chúng.

Hình 3.20 minh họa mô hình nhóm chúng tôi đề xuất. Đầu vào của mô hình sẽ là ảnh mà chúng ta muốn dự đoán chúng có chứa nốt sần (Positive) hay không chứa nốt sần (Negative). Sau đó, ảnh đầu vào sẽ được chuyển đến bước tiền xử lý (Pre-processing). Công đoạn tiền xử lý sẽ đảm nhận các vai trò như resize kích thước và chuẩn hóa (normalize) các ảnh đầu vào, một số ảnh đầu vào sẽ được tăng cường bằng phương pháp lật ảnh ngang, lật ảnh dọc theo một xác suất mà chúng tôi quy định. Sau khi trải qua công đoạn tiền xử lý, các ảnh tiếp tục được đẩy vào mạng DenseNet. Mạng DenseNet có tác dụng rút trích các đặc trưng trong ảnh (feature extraction) thành các bản đồ đặc trưng (feature map). Sau đó, các bản đồ đặc trưng này được đưa qua khối Channel Attention để đánh trọng số, việc đánh trọng số trên các feature map nhằm nhấn mạnh xem các đối tượng nào quan trọng để tập trung chú ý vào chúng, các đối tượng nào ít quan trọng thì mức độ chú ý đến chúng sẽ thấp hơn. Đầu ra của khối Channel Attention sẽ được nhân với đầu ra của lớp DenseNet phía trước đó, sau đó đi qua các lớp global average pooling, fully connected, sigmoid và trả về xác suất của 2 lớp Positive và Negative.

Lý do chúng tôi chọn mạng DenseNet: Với các ưu điểm của DenseNet như: số lượng tham số cần thiết khi training ít hơn một nửa so với ResNet nhưng độ chính xác vẫn được giữ nguyên, đồng thời khả năng tránh over-fitting, tận dụng lại tối ưu các feature, ... là những lý do chính để chúng tôi chọn mạng DenseNet. Ngoài ra, DenseNet là một trong những mạng phân loại khá hiện đại, được thử nghiệm và đạt nhiều kết quả cao trên các tập dữ liệu lớn. Mô hình này đã được tiền huấn luyện trên các tập dữ liệu lớn nên chúng tôi có thể tận dụng lại các trọng số của mô hình và không mất nhiều chi phí về thời gia và tài nguyên trong việc huấn luyện cho mô hình.

Lý do chúng tôi sử dụng cơ chế Attention: Cơ chế Attention được thêm vào để cải thiện hiệu quả phân lớp của mô hình học sâu. Một bản đồ chú



Hình 3.20: Mô hình đề xuất thứ nhất là sự kết hợp giữa DenseNet và Channel attention module

ý (attention map) sẽ được tạo ra bằng các khai thác các mối quan hệ giữa các kênh của các đối tượng. Mỗi kênh của feature map được xem như là một feature detector, nên channel attention sẽ tập trung vào những điểm có ý nghĩa của hình ảnh đầu vào nhằm xem xét phần thông tin nào quan trọng để nhấn mạnh hoặc phần thông tin nào ít quan trọng thì sự chú ý đối với chúng sẽ giảm đi. Ngoài ra, Channel attention mà một module nhẹ, dễ dàng kết hợp với các mô hình khác.

3.6.3 Mô hình đề xuất 2: EfficientNet

Các mạng Nơ ron tích chập được phát triển với tham số ban đầu là cố định, sau đó sẽ có thể tăng độ chính xác nếu đầu vào có được tham số lớn. Nhưng khi ta tăng kích thước của model đến một ngưỡng nào đấy thì độ chính xác trên tập dữ liệu sẽ bão hòa hoặc sẽ giảm. Đi kèm với sự gia tăng kích thước đó, chúng ta cần nhiều tài nguyên hơn để huấn luyện.

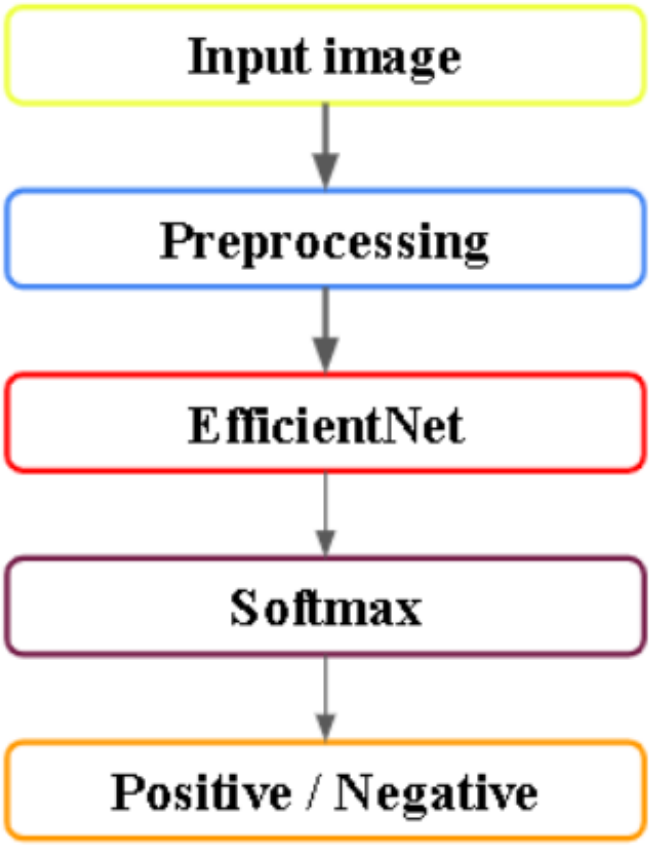
Nhận thấy các vấn đề trên, nhóm tác giả của phương pháp EfficientNet đã nghiên cứu và nhận thấy việc thu phóng cân bằng một cách có hệ thống (compound scaling) độ sâu, chiều rộng, độ phân giải (network depth, width, resolution) của một mạng có thể mang đến hiệu suất tốt hơn. Đây cũng một phướng phát triển mới cho các mô hình học sâu để tăng độ chính xác và cải thiện hiệu suất, minh chứng cho điều đó thì phương pháp này đã đạt được các thứ hạng cao trên các tập dữ liệu lớn như ImageNet.

Ngoài việc áp dụng cơ chế thu phóng mô hình, kiến trúc EfficientNet còn đặc biệt ở chỗ gồm nhiều khối Mobile inverted bottleneck. Đối với các mô hình thì độ sâu quá lớn là một trong những nguyên nhân quan trọng dẫn đến việc số lượng tham số mô hình tăng cao. Khối MB Conv đã sử dụng tích chập tách biệt theo chiều sâu (depthwise separable convolution) giúp giảm lượng tham số đầu vào, chi phí tính toán cho mô hình.

Thêm vào đó, khối MBConV có một module gọi là SE, hay có tên đầy đủ là Squeeze-and-excitation. Khối SE khá tương đồng với self attention, mục đích giúp mô hình đánh trọng số cho những phần quan trọng của bức

ảnh, từ đó phân loại ảnh tốt hơn, cải thiện hiệu suất của mô hình.

Mô hình này đã được tiền huấn luyện trên các tập dữ liệu lớn nên chúng tôi có thể tận dụng lại các trọng số của mô hình và không mất nhiều thời gian, tài nguyên trong việc huấn luyện cho mô hình. Vì những lý do trên chúng tôi sử đã sử dụng EfficientNet cho mô hình đề xuất này.



Hình 3.21: Mô hình đề xuất thứ hai sử dụng EfficientNet

Hình 3.21 minh họa mô hình nhóm chúng tôi đề xuất. Đầu vào của mô hình sẽ là ảnh mà chúng ta muốn dự đoán chúng có chứa nốt sần (Positive) hay không chứa nốt sần (Negative). Sau đó, ảnh đầu vào sẽ được chuyển đến bước tiền xử lý (Pre-processing). Công đoạn tiền xử lý sẽ đảm nhận các vai trò như resize kích thước và chuẩn hóa (normalize) các ảnh đầu vào, một số ảnh đầu vào sẽ được tăng cường bằng phương pháp lật ảnh ngang, lật ảnh dọc theo một xác suất mà chúng tôi quy định. Sau khi trải qua công đoạn tiền xử lý, các ảnh tiếp tục được đẩy vào mạng Efficient

B7, mạng này sẽ trả về kết quả là xác suất của 2 lớp Positive và Negative.

3.6.4 Độ đo đánh giá

True/False Positive/Negative

Phương pháp này thường được áp dụng để đánh giá cho các bài toán phân lớp có 2 lớp dữ liệu. Đặc biệt, trong 2 lớp cần được dự đoán sẽ có 1 lớp nghiêm trọng hơn lớp còn lại, nên cần được dự đoán chính xác.

Ví dụ: trong bài toán phân loại ảnh có khối u và không có khối u ở đồ án này có 2 lớp, lớp thứ nhất là Positive và lớp thứ hai là Negative. Lớp Positive là ảnh có chứa nốt sần là khối u nên sẽ nghiêm trọng hơn, cần được phân loại chính xác hơn.

Trong phương pháp này, người ta định nghĩa lớp dữ liệu quan trọng hơn cần được xác định đúng là lớp Positive (P-dương tính), lớp còn lại là Negative (N-âm tính). Ta định nghĩa True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) dựa trên confusion matrix chưa chuẩn hóa như sau:

Bảng 3.1: Định nghĩa True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) dựa trên confusion matrix chưa chuẩn hoá.

	Predicted: Positive	**Predicted: Negative**
Actual: Positive	True Positive (TP)	False Negative (FN)
Actual: Negative	False Positive (FP)	True Negative (TN)

- **TP**: số lượng các mẫu thuộc lớp Positive được mô hình phân loại dự đoán đúng
- **FN**: số lượng các mẫu thuộc lớp Positive nhưng được mô hình phân loại dự đoán là Negative.

- **TN**: số lượng các mẫu thuộc lớp Negative được mô hình phân loại đúng.
- **FP**: số lượng các mẫu thuốc lớp Negative nhưng được mô hình phân loại dự đoán là Positive.

Người ta thường quan tâm đến TPR, FNR, FPR, TNR (R - Rate) dựa trên normalized confusion matrix như sau:

Bảng 3.2: Tính toán các giá trị: TPR, FNR, FPR, TNR

	Predicted: Positive	**Predicted: Negative**
Actual: Positive	TPR	FNR
Actual: Negative	FPR	TNR

- **TPR (True Positive Rate)** - hay còn được gọi là recall hoặc sensitivity: thể hiện tỷ lệ phân loại đúng các mẫu thuộc lớp Positive trên tổng số tất cả các mẫu Positive. Tỷ lệ này dùng để đánh giá độ nhạy của mô hình.

$$TPR = \frac{TP}{TP + FN} \tag{3.9}$$

- **FPR (False Positive Rate)** - hay còn được gọi là tỷ lệ báo động nhầm (False Alarm Rate): thể hiện tỷ lệ phân loại nhầm các mẫu thuộc lớp Negative thuộc về lớp Positive.

$$FPR = \frac{FP}{FP + TN} \tag{3.10}$$

- **TNR (True Negative Rate)** - hay còn được gọi là Specificity: thể hiện tỷ lệ phân loại đúng các mẫu thuộc lớp Negative trên tổng số

tất cả các mẫu Negative. Tỷ lệ này dùng để đánh giá độ đặc hiệu của mô hình.

$$FNR = \frac{FN}{TP + FN} \tag{3.11}$$

- **FNR (False Negative Rate)** - hay còn được gọi là tỷ lệ bỏ sót (Miss Detection Rate): thể hiện tỷ lệ nhận phân loại nhầm các mẫu thuộc lớp Positive thuộc về lớp Negative.

$$TNR = \frac{TN}{FP + TN} \tag{3.12}$$

Precision

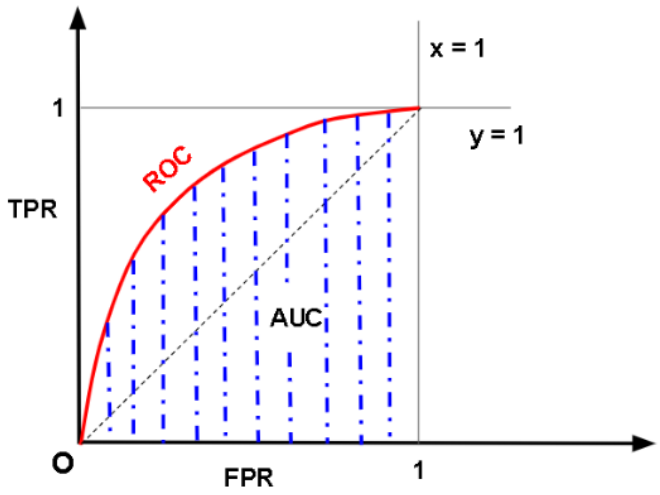
Precision là độ đo thể hiện khả năng phân loại Positive chính xác của mô hình. Nói cách khác, có bao nhiêu dự đoán "positive" là thật sự "true" trong thực tế.

$$Precision = \frac{TP}{TP + FP} \tag{3.13}$$

ROC - AUC

ROC (Receiver Operating Characteristics) là một đường cong biểu diễn khả năng phân loại của một mô hình phân loại tại các ngưỡng (threshold) khác nhau. Đường cong này dựa trên 2 chỉ số là TPR và FPR. Để hợp 2 chỉ số này thành 1 chỉ số duy nhất ta sử dụng đường cong ROC để biểu diễn từng cặp giá trị (TPR, FPR) cho các ngưỡng khác nhau. Với mỗi điểm trên đường cong biểu diễn 1 cặp (TPR, FPR) cho 1 ngưỡng. Ngưỡng phân loại tốt nhất của mô hình là 1 cặp (TPR, FPR) sao cho khoảng cách giữa cặp điểm này đến tọa độ (0, 1) là ngắn nhất.

AUC (Area Under The Curve) là chỉ số đánh giá khả năng phân loại của mô hình tốt như thế nào, được tính toán dựa trên đường cong ROC. Phần diện tích nằm dưới đường cong ROC và trên trục hoành là AUC,



Hình 3.22: Biểu diễn đường cong ROC và phần diện tích AUC

AUC có giá trị trong khoảng [0, 1]. Chỉ số AUC càng gần 1 thì khả năng phân loại của mô hình càng tốt, điều này đồng nghĩa với việc đường cong ROC sẽ có xu hướng tiệm cận đường thẳng y = 1. Khi chỉ số AUC = 0.5 đồng nghĩa với việc đường cong ROC sẽ trùng với đường thẳng đi qua hai điểm (0, 0) và (1, 1). Đây là trường hợp xấu nhất, mô hình hoàn toàn không có khả năng phân loại giữa 2 lớp (mô hình sẽ phân loại một cách ngẫu nhiên). Khi chỉ số AUC càng gần 0 thì mô hình có xu hướng phân loại ngược giữa 2 lớp Positive thành Negative và ngược lại Negative thành Positive.

Chương 4

Kết quả thực nghiệm

4.1 Giới thiệu tập dữ liệu

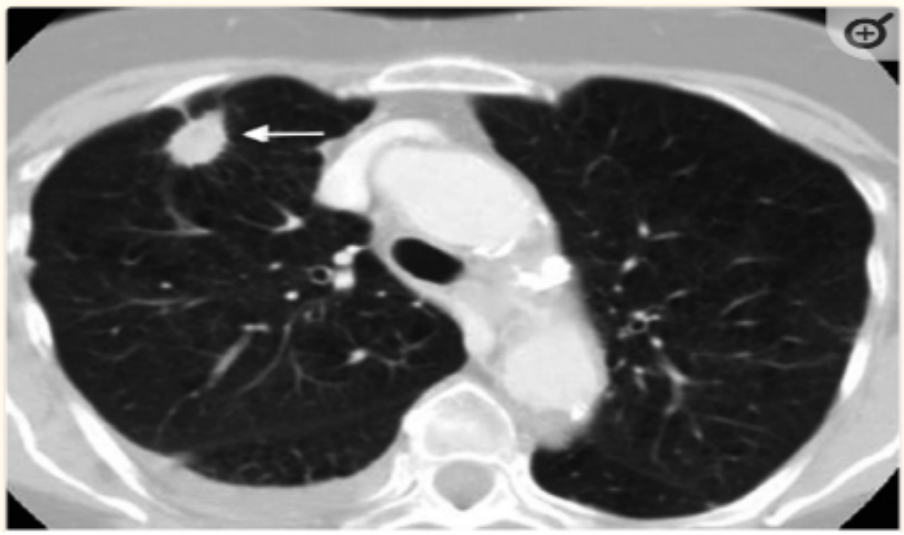
Dữ liệu gốc từ Hiệp hội Nghiên cứu Các bệnh Truyền nhiễm và Cơ sở dữ liệu về Phổi (the Association for Research on Infectious Diseases and Lung Database) [(LIDC / IDRI)]. Nhưng chúng tôi sử dụng một phiên bản khác là LUNA16. Bộ dữ liệu này bao gồm 888 bản chụp CT với các ghi chú mô tả tọa độ của vùng nốt sần và không nốt sần. Mỗi bản chụp CT có kích thước 512 x 512 x n, trong đó n là số lần quét, có khoảng 200 hình ảnh trong mỗi lần chụp CT.

Từ các lát cắt của ảnh CT ở phổi, 12 bác sĩ sẽ annotate tiến hành đánh dấu các khác thường trên các lát cắt ấy và phân chúng thành 3 nhóm chính: gồm nốt sần $\geq$ 3mm, nốt sần $<$ 3mm, các tổn thương khác.

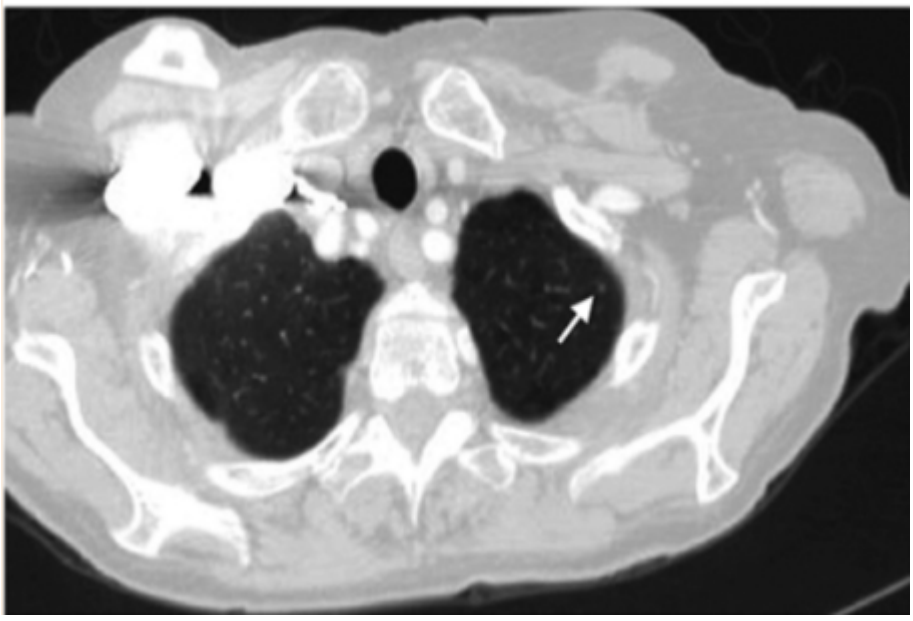
"Nodule $\geq$ 3 mm": được định nghĩa là bất kỳ tổn thương nào được xem là nốt sần có kích thước mặt phẳng lớn nhất trong khoảng 3 - 30 mm. (Hình 4.1 một ví dụ về loại nốt sần này)

"Nodule $<$ 3 mm": được định nghĩa là bất kỳ tổn thương nào được coi là nốt sần có kích thước mặt phẳng lớn nhất nhỏ hơn 3 mm mà không rõ ràng là lành tính. (Hình 4.2 một ví dụ về loại nốt sần này)

"Non-nodule $\geq$ 3 mm": được định nghĩa là bất kỳ các tổn thương phổi khác (chẳng hạn như sẹo đỉnh), có kích thước mặt phẳng lớn nhất lớn hơn

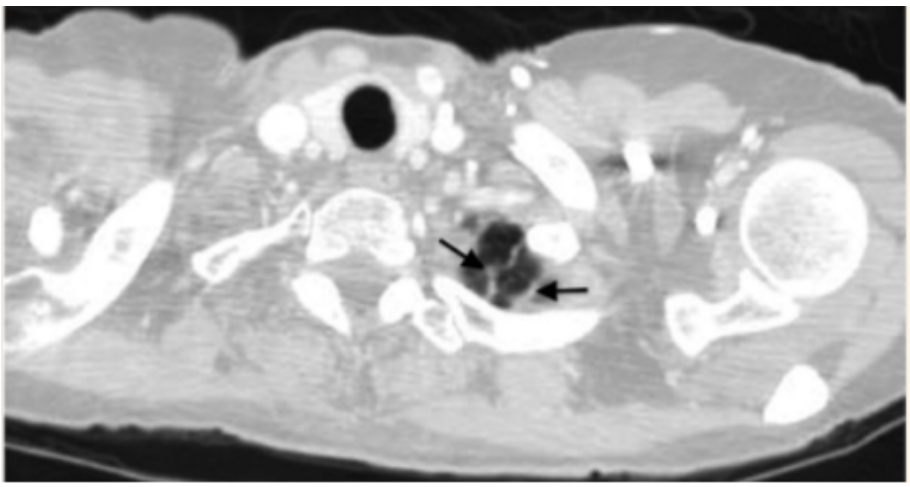


Hình 4.1: Ảnh chụp cắt lớp có chứa nốt sần có kích thước lớn hơn hoặc bằng 3mm. Nguồn: [5]



Hình 4.2: Ảnh chụp cắt lớp có chứa nốt sần có kích thước bé hơn 3mm. Nguồn: [5]

hoặc bằng 3 mm và không có các đặc điểm như nốt sần. (Hình 4.3 một ví dụ về khái niệm này)

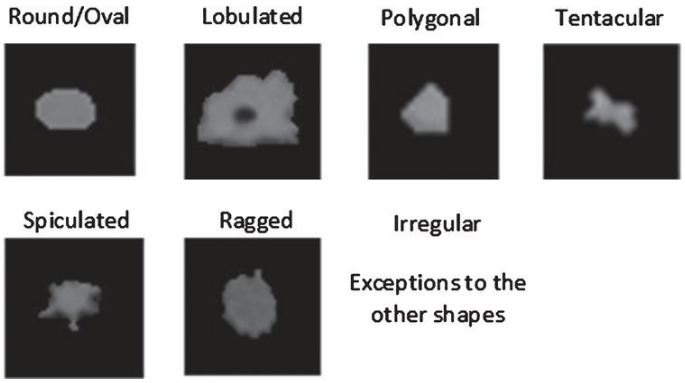


Hình 4.3: Ảnh chụp cắt lớp có chứa các tổn thương khác ở phổi không phải là nốt sần. Nguồn: [5]

Tập dữ liệu chúng ta sử dụng gồm 2 lớp: Positive và Negative.

- Positive: các nốt sần to, rõ, hình dáng rõ ràng (thường giống hình tròn).
- Negative: các nốt sần quá nhỏ, không rõ ràng và các tổn thương khác ở phổi.

Hình 4.4 gồm đặc điểm của các nốt sần trong lớp Positive:



Hình 4.4: Hình dạng của các nốt sần thuộc lớp Positive. Nguồn: [9]

Có tổng cộng 551065 chú thích. 1351 ghi chú được gán nhãn là dương tính (Positive) thường được gọi là nốt sần, trong khi phần còn lại được gán nhãn là âm tính (Negative), không phải là nốt sần.

Tác giả Swetha Subramanian [21] đã cắt hình ảnh xung quanh các tọa độ được cung cấp trong chú thích thành một tập các ảnh với kích thước 50x50 pixel trong thang màu xám dùng để training, validating và testing cho mô hình học sâu.

Tập dữ liệu cuối cùng mà chúng tôi sử dụng trong phạm vi đồ án này bao gồm 8106 hình ảnh có kích thước 50x50 pixel được chia thành 3 thư mục, cụ thể: Tập dữ liệu training gồm 5187 ảnh, trong đó có 845 ảnh thuộc lớp positive và 4342 ảnh thuộc lớp negative. Tập dữ liệu validating gồm 1297 ảnh, trong đó có 224 ảnh thuộc lớp positive và 1073 ảnh thuộc lớp negative. Tập dữ liệu testing gồm 1622 ảnh, trong đó có 282 ảnh thuộc lớp positive và 1340 ảnh thuộc lớp negative. Tỉ lệ các lớp là 20:80.

Bảng 4.1: Số lượng các lớp thuộc các tập train, validation và test

	Positve	**Negative**	**Sum**
Train	845	4342	5187
Validation	224	1073	1297
Test	282	1340	1622

4.2 Chi tiết quá trình thực nghiệm

4.2.1 Môi trường huấn luyện, ngôn ngữ và thư viện

Môi trường huấn luyện: chúng tôi tiến hành huấn luyện và kiểm thử mô hình trên máy tính của Kaggle cung cấp. Sở dĩ chúng tôi sử dụng máy tính của Kaggle là do ở đây có hỗ trợ GPU điều mà máy tính chúng tôi không có. Dùng GPU để quá trình huấn luyện mô hình được diễn ra nhanh hơn, qua đó có thể tiết kiệm được thời gian.

Ngôn ngữ sử dụng: chúng tôi sử ngôn ngữ Python cho đề tài này.

Thư viện sử dụng: hai thư viện chính mà chúng tôi sử dụng là Pytorch và Tensorflow trong việc huấn luyện các mô hình. Ngoài ra, chúng tôi còn sử dụng thêm các thư viện phụ như: numpy, pandas, sklearn, matplotlib để xử lý số liệu và trực quan hóa dữ liệu.

4.2.2 Phương pháp đánh giá

Chúng tôi sẽ so sánh và đánh giá 2 mô hình mà chúng tôi đã đề xuất trên cùng một tập dữ liệu như đã đề cập ở phần 4.1. Trong quá trình huấn luận ở mỗi epoch chúng tôi sẽ tính toán ra độ lỗi trong việc phân loại hình ảnh và các độ đo như: precision, recall, specificity.

4.2.3 Kết quả thực nghiệm

Mô hình 1: DenseNet kết hợp Channel attention module:

Chúng tôi sử dụng các tham số cho mô hình này như sau:

- **Mạng DenseNet:** phiên bản DenseNet 201.
- **Image size:** 256x256x3.
- **Số epoch huấn luyện:** 30.
- **Batch size:** 16.
- **Thuật toán tối ưu:** Adam.
- **Learning rate:** 0.0001.

Mô hình 2: EfficientNet:

Chúng tôi sử dụng các tham số cho mô hình này như sau:

- **Mạng EfficientNet:** phiên bản EfficientNet B7.

- **Image size:** 224x224x3.
- **Số epoch huấn luyện:** 30.
- **Batch size:** 8.
- **Thuật toán tối ưu:** Adam.
- **Learning rate:** 0.0001.

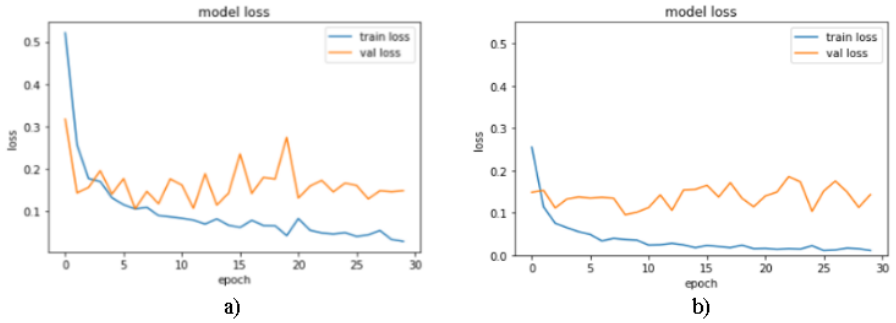
Kết quả thu được

Hình 4.5 biểu diễn trực quan độ lỗi của các mô hình trong quá trình huấn luyện.

Bảng 4.2 thể hiện kết quả của 2 mô hình trên tập test theo TP, FN, FP, TN.

Bảng 4.3 thể hiện kết quả của 2 mô hình trên tập test theo các độ đo Precision, Recall, Specificity.

Hình 4.6 biểu diễn đường cong AUC thể hiện hiệu suất phân loại của các mô hình.



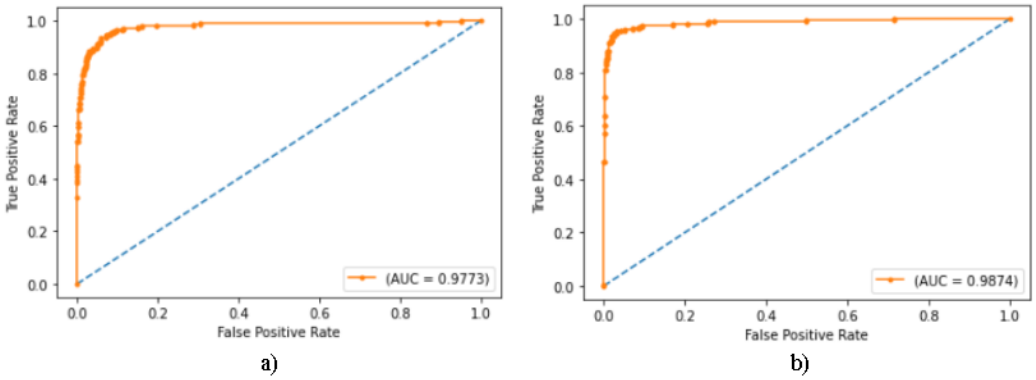
Hình 4.5: Biểu đồ thể hiện độ lỗi của các mô hình trong quá trình huấn luyện (Hình a: thể hiện độ lỗi của mô hình DenseNet201 kết hợp với Channel attention module. Hình b: thể hiện độ lỗi của mô hình EfficientNet B7.)

Bảng 4.2: So sánh kết quả của 2 mô hình trên tập test theo TP, FN, FP, TN.

Mô hình	**TP**	**FN**	**FP**	**TN**
DenseNet201 + CAM	232	50	27	1313
EfficientNet B7	255	27	16	1324

Bảng 4.3: So sánh kết quả của 2 mô hình trên tập test theo các độ đo Precision, Recall, Specificity.

Mô hình	**Precision**	**Recall**	**Specificity**
DenseNet201 + CAM	0.8958	0.8227	0.9799
EfficientNet B7	0.9410	0.9043	0.9881



Hình 4.6: Biểu đồ thể hiện chỉ số AUC của các mô hình (Hình a: thể hiện chỉ sô AUC của mô hình DenseNet201 kết hợp với Channel attention module. Hình b: thể hiện chỉ số AUC của mô hình EfficientNet B7.)

Chúng tôi nhận thấy mô hình Efficient B7 đã sử dụng cho kết quả vượt trội hơn so với mô hình còn lại. Đối với mô hình DenseNet201 kết hợp với Channel attention module, dù đã thêm khối Channel attention vào sau mạng DenseNet với mong muốn để cải thiện hiệu suất của DenseNet thế nhưng đó là chưa đủ để vượt qua EfficientNet B7. Nguyên nhân có thể là do EfficientNet B7 ngoài việc áp dụng cơ chế thu phóng mô hình thì kiến trúc của nó gồm các khối Mobile Inverted Botteleneck Convolutional(MBConv) xếp chồng lên nhau, khối MBConV có một module gọi là SE, hay có tên đầy đủ là Squeeze-and-excitation. Khối SE này khá tương đồng với self attention, mục đích giúp mô hình đánh trọng số cho những phần quan trọng của bức ảnh, từ đó phân loại ảnh tốt hơn, cải thiện hiệu suất của mô hình.

Khảo sát các phương pháp khác

Sau khi tìm ra được mô hình tốt nhất trong các mô hình mà nhóm chúng tôi đã đề xuất, chúng tôi tiến hành huấn luyện lại mô hình nhưng lần này là với toàn bộ dữ liệu và so sánh với kết quả của các tác giả khác trên cùng bộ dữ liệu.

Bảng 4.4 thể hiện kết quả của mô hình tốt nhất của chúng tôi so với kết quả của bài báo "Detection of Lung Nodules on CT Images based on

the Convolutional Neural Network with Attention Mechanism" [12] trên tập dữ liệu tương tự.

Bảng 4.4: So sánh kết quả của 2 mô hình trên tập test theo các độ đo Precision, Recall, Specificity.

Mô hình	**Precision**	**Recall**	**Specificity**	**Auc**
Paper [12]	0.950	0.864	0.982	0.992
EfficientNet B7	0.950	0.8759	0.9903	0.993

Chương 5

Kết luận và hướng phát triển

5.1 Kết luận

Để giải quyết bài toán về phân loại khối u trong phổi, một mô hình giải quyết bài toán phức tạp, với độ chính xác cao, tỉ lệ False-Positive thấp, đã được đưa ra. Mô hình dựa vào cách thu phóng cân bằng của kiến trúc học sâu CNN. Mô hình có thể dự đoán trực tiếp từ hình ảnh khối u đầu vào, không cần phải xử lí hay rút trích các đặc trưng từ ảnh. Kết quả thực nghiệm trên bộ dữ liệu LIDC-IDRI cho thấy các chỉ số auc, recall, đường cong ROC vượt qua tất cả các phương pháp được liệt kê đến trong báo cáo này, bao gồm các phương pháp học sâu và học máy truyền thống.
Kết quả của thực nghiệm còn cho thấy rằng việc áp dụng kiến trúc học sâu của EfficientNet vào bài toán phân loại khối u trong phổi đạt hiệu quả cao hơn kết quả của mô hình Densenet kết hợp CAM module.
Dù các thực nghiệm mà nhóm làm được có giới hạn, nhưng nhóm sinh viện nhận thấy rằng công trình của mình có khả năng phân loại khối u cao, có thể hỗ trợ bác sĩ dự đoán đúng hơn các khối u của bệnh nhân, qua đó có thể góp phần làm tăng hiệu quả của các bệnh viện cũng như giúp đỡ cho bệnh nhân.

5.2 Hướng phát triển

- Xây dựng hệ thống đưa được mô hình để thực hiện trên các Website, App...
- Xây dựng mô hình phân loại với độ chính xác cao hơn và thời gian xử lý ngắn hơn. Có thể áp dụng một số thuật toán, mô hình mới nhất để cải thiện kết quả.
- Xây dựng mô hình mà có thể phân loại được khối u lành tính hoặc ác tính, thậm chí là chính xác căn bệnh mà phổi đang mắc phải.

Tài liệu tham khảo

Tiếng Việt

[1] Nguyen Thanh Huyen. *[Paper explained] Some Face Recognition Approaches: Facenet, ArcFace, CosFace*. URL: https://viblo.asia/p/paper-explained-some-face-recognition-approaches-facenet-arcface-cosface-Do754zgLZM6 (visited on 06/23/2022).

[2] Pham Dinh Khanh. *Bài 38 - Các kiến trúc CNN hiện đại*. URL: https://phamdinhkhanh.github.io/2020/05/31/CNNHistory.html (visited on 06/23/2022).

[3] Pham Dinh Khanh. *Mobilenet model*. URL: https://phamdinhkhanh.github.io/2020/09/19/MobileNet.html (visited on 06/23/2022).

[4] Bui Quang Manh. *Những mô hình trợ thủ đắc lực trong các mô hình Deep learning [Phần 1]*. URL: https://viblo.asia/p/nhung-mo-hinh-tro-thu-dac-luc-trong-cac-mo-hinh-deep-learning-phan-1-WAyK8GO65xX (visited on 06/23/2022).

[5] Minh Nguyen. *Xử lý ảnh - Convolution là gì?* URL: https://minhng.info/tutorials/xu-ly-anh-convolution-la-gi.html (visited on 06/23/2022).

[6] Tran Ho Dat Phan Anh Cang Phan Thuong Cang. *PHÁT HIỆN TỔN THƯƠNG PHỔI BẰNG KỸ THUẬT HỌC SÂU TRONG MÔI TRƯỜNG XỬ LÝ SONG SONG SPARK*. URL: http://vap.ac.vn/

Portals/0/TuyenTap/2021/12/22/1ecec417207345d595e011cb434f7fe8/10_FAIR2021_paper_9.pdf (visited on 06/23/2022).

[7] Tran Trung Truc. *Optimizer- Hiểu sâu về các thuật toán tối ưu (GD,SGD,Adam,..)* URL: https://viblo.asia/p/optimizer-hieu-sau-ve-cac-thuat-toan-toi-uu-gdsgdadam-Qbq5QQ9E5D8 (visited on 06/23/2022).

Tiếng Anh

[3] A. A. A. Setio F. Ciompi, G. Litjens. "Pulmonary nodule detection in CT images: false positive reduction using multview convolutional networks". In: *IEE Transactions on Medical* (2016).

[4] A. A. A. Setio A. Traverso, T. de Bel. "Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge". In: *Medical Image Analysis* (2017).

[5] Armato, Samuel G, McLennan, Geoffrey, and Bidaut, Luc. *LUNA16 Dataset.* URL: https://luna16.grand-challenge.org/Data/ (visited on 06/23/2022).

[6] Armato, Samuel G, McLennan, Geoffrey, and Bidaut, Luc. *The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI).* URL: https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI#1966254f633413761b746ff9e49dd8f0d5b0 (visited on 06/23/2022).

[7] Choi, W.-J. and Choi, T.-S. "Automated pulmonary nodule detection system in computed tomography images: a hier-archical block classification approach". In: (2013).

[8] Giang Son Tran Thi Phuong Nghiem, Van Thi Nguyen Chi Mai Luong Jean-Christophe Burie. "Improving accuracy of lung nodule classification using deep learning with focal loss". In: *Journal of healthcare engineering* (2019).

[9] Huafenga, Wang et al. "A hybrid CNN feature model for pulmonary nodule malignancy risk differentiation". In: *Journal of X-Ray Science and Technology* (2018).

[10] Kuruvilla, J. and Gunavathi, K. "Lung cance classification using neural networks for CTimages". In: *Computer Methods and Programs in Biomedicine* (2014).

[11] L. A. Torre, R. L. Siegel and Jemal, A. "Lung cancer statistics". In: *Lung Cancer and Personalized Medicine* (2016).

[12] Lai, Khai Dinh, Nguyen, Thuy Thanh, and Le, Thai Hoang. "Detection of Lung Nodules on CT Images based on the Convolutional Neural Network with Attention Mechanism". In: *Published by International Association of Educators and Researchers (IAER)* (2021).

[13] Mallick, Satya. *EfficientNet: Theory + Code*. URL: https://learnopencv.com/efficientnet-theory-code/ (visited on 06/23/2022).

[14] Mingxing Tan, Quoc V. Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *International Conference on Machine Learning, 2019* (2019).

[15] Mingxing Tan Bo Chen, Ruoming Pang Vijay Vasudevan Mark Sandler Andrew Howard Quoc V. Le. "MnasNet: Platform-Aware Neural Architecture Search for Mobile". In: *CVPR 2019* (2019).

[16] Online. *Pooling Operation*. URL: https://programmathically.com/what-is-pooling-in-a-convolutional-neural-network-cnn-pooling-layers-explained/ (visited on 06/23/2022).

[17] Online. *Skip connection in Resnet*. URL: https://programmathically.com/an-introduction-to-residual-skip-connections-and-resnets/ (visited on 06/23/2022).

[18] Patel, Krut. *Convolution Operation*. URL: https://towardsdatascience.com/convolution-neural-networks-a-beginners-guide-implementing-a-mnist-hand-written-digit-8aa60330d022 (visited on 06/23/2022).

[19] Rebecca L Siegel Kimberly D Miller, Kimberly D Miller. "Cancer Statistics". In: *Cancer Journal for Clinicians* (2017).

[20] Sanghyun Woo Jongchan Park, Joon-Young Lee In So Kweon. "CBAM: Convolutional Block Attention Module". In: *ECCV 2018* (2018).

[21] Subramanian, Swetha. *LUNA16 Data Custom*. URL: https://github.com/swethasubramanian/LungCancerDetection?fbclid=IwAR1sUsGIs10APcMuSAF4tCp3M3Amg (visited on 06/23/2022).