ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA TOÁN TIN HỌC

HỒ NGỌC ÂN - 20280001

HỎA NGỌC TÚ - 20280111

ỨNG DỤNG HỌC SÂU TRONG VIỆC PHÁT HIỆN GÃY XƯƠNG TRÊN ẢNH CHỤP X QUANG

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN KHOA HỌC

NGÀNH KHOA HỌC DỮ LIỆU

THÀNH PHỐ HỒ CHÍ MINH – 07/2024

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA TOÁN TIN HỌC

HỒ NGỌC ÂN - 20280001

HỎA NGỌC TÚ - 20280111

ỨNG DỤNG HỌC SÂU TRONG VIỆC PHÁT HIỆN GÃY XƯƠNG TRÊN ẢNH CHỤP X QUANG

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN KHOA HỌC

NGÀNH KHOA HỌC DỮ LIỆU

GIẢNG VIÊN HƯỚNG DẪN KHOA HỌC:

ThS. HUỲNH THANH SƠN

THÀNH PHỐ HỒ CHÍ MINH – 07/2024

LỜI CẢM ƠN

Luận văn được thực hiện tại khoa Toán Tin học – Trường Đại học Khoa học Tự Nhiên, dưới sự hướng dẫn của Ths. Huỳnh Thanh Sơn.

Trước tiên nhóm tôi bày tỏ lòng biết ơn sâu sắc đến Ths. Huỳnh Thanh Sơn đã giúp chúng tôi nghiên cứu đề tài này. Thầy đã tận tình hướng dẫn và hỗ trợ giúp nhóm tôi tiếp cận và đạt kết quả trong việc nghiên cứu đề tài của nhóm. Thầy luôn động viên, hỗ trợ và chỉ dẫn tận tình giúp nhóm tôi hoàn thành luận văn này.

Nhóm tôi xin bày tỏ lòng biết ơn tới các Thầy/Cô thuộc khoa Toán Tin học – Trường Đại học Khoa học Tự Nhiên đã tạo mọi điều kiện thuận lợi trong quá trình học tập và nghiên cứu tại trường.

Nhóm tôi xin chân thành cảm ơn Thầy/Cô trong hội đồng đánh giá luận văn tốt nghiệp đã đóng góp ý kiến quý báu giúp chúng tôi hoàn thiện luận văn.

TP. Hồ Chí Minh, ngày 1 tháng 7 năm 2024

Tác giả

Hồ Ngọc Ân,

Hỏa Ngọc Tú

MỤC LỤC

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

DANH SÁCH THUẬT NGỮ VÀ CHỮ VIẾT TẮT

Viết tắt	Tiếng anh
CNN	Convolutional Neural Networks
VGG	Visual geometry group
Resnet	Residual Network
ReLu	Rectified Linear Unit
GAP	Global average pooling
Densenet	Dense convolutional network
Regnet	Regularized network
Maxvit	Multi-Axis Vision Transformer
ViT	Vision transformer
YOLO	You Only Look Once
PAnet	Path Aggregation Network
NMS	Non-Maximum Suppression
PGI	Programmable Gradient Information
GELAN	Layer Aggregation Network

LỜI MỞ ĐẦU

1. Lý do chọn đề tài.

Ngày nay trong lĩnh vực y tế, đặc biệt là trong chẩn đoán hình ảnh, việc phát hiện sớm và chính xác các chấn thương xương như gãy xương là một vấn đề quan trọng trong việc đưa ra các phương pháp điều trị kịp thời và hiệu quả. Gãy xương là một chấn thương phổ biến có thể xảy ra do nhiều nguyên nhân khác nhau như tai nạn giao thông, té ngã, hoặc do các bệnh lý làm suy yếu cấu trúc xương. Việc chẩn đoán nhanh chóng và chính xác có thể giúp giảm thiểu thời gian điều trị, cải thiện kết quả phục hồi cho bệnh nhân, và giảm chi phí y tế.

Với sự phát triển mạnh mẽ của trí tuệ nhân tạo, đặc biệt là các kỹ thuật học sâu, việc ứng dụng những mô hình học sâu trong việc phân tích ảnh y tế đã trở nên khả thi và ngày càng phổ biến. AI có thể học từ các tập dữ liệu lớn để nhận diện các mẫu và đặc điểm mà mắt thường khó phát hiện. Các mô hình AI, đặc biệt là các mô hình học sâu, đã chứng minh hiệu quả vượt trội trong việc phân loại, phân đoạn và phát hiện các đặc điểm phức tạp trên ảnh X-quang. Việc phát triển một hệ thống tự động phân loại, phân đoạn và phát hiện gãy xương từ ảnh X-quang không chỉ có ý nghĩa trong việc cải thiện quy trình chẩn đoán mà còn mở ra cơ hội cho các ứng dụng trong hệ thống y tế thông minh. Các hệ thống này có thể hỗ trợ các bác sĩ chẩn đoán chính xác hơn, giảm tải công việc, và tối ưu hóa quy trình chăm sóc sức khỏe.

Với ý tưởng trên, và nhận được sự đồng ý của Ths. Huỳnh Thanh Sơn , nhóm tôi chọn đề tài "Ứng dụng học sâu trong việc phát hiện gãy xương trên ảnh chụp X quang", khi đề tài hoàn thành không những giúp cải thiện chất lượng chẩn đoán hình ảnh y tế mà còn đóng góp vào việc xây dựng nền tảng cho các nghiên cứu và ứng dụng tiếp theo trong lĩnh vực phát hiện bệnh qua hình ảnh. Kết quả của nghiên cứu có thể được chia sẻ và mở rộng trong cộng đồng y tế và khoa học, giúp cải thiện chất lượng chăm sóc sức khỏe cho cộng đồng.

2. Mục tiêu nghiên cứu.
    - Tìm hiểu bài toán phân loại hình ảnh, phân vùng ảnh, và phát hiện đối tượng gãy xương trong tập dữ liệu FracAtlas.
    - Nghiên cứu các mô hình giải quyết bài toán phân loại hình ảnh nhị phân.
    - Nghiên cứu các mô hình giải quyết bài toán phân vùng ảnh.
    - Nghiên cứu các mô hình giải quyết bài toán phát hiện đối tượng.
    - Tiến hành thực thi các mô hình và đánh giá kết quả thực nghiệm mô hình trên bộ dữ liệu FracAtlas.
3. Đối tượng và phạm vi nghiên cứu.
    a. Đối tượng.
        - Các mô hình phân loại hình ảnh (classification models)
        - Các mô hình phát hiện đối tượng (object detection)
        - Các mô hình phân vùng ảnh (image segmentation)
    b. Phạm vi nghiên cứu.
        - Bộ dữ liệu về ảnh chụp X-quang FracAtlas.
        - Các mô hình học sâu.
        - Ngôn ngữ lập trình Python.

CHƯƠNG 1. GIỚI THIỆU CHUNG.

Chụp X quang kỹ thuật số là một trong những tiêu chuẩn phổ biến và hiệu quả nhất để chẩn đoán gãy xương. Để có những chẩn đoán như vậy, cần có sự can thiệp của các bác sĩ và chuyên gia. Với sự phát triển nhanh chóng của thị giác máy tính, việc sử dụng mô hình hỗ trợ trong chuẩn đoán ngày càng được quan tâm. Dựa trên tập dữ liệu "FracAtlas: A Dataset for Fracture Classification, Localization and Segmentation of Musculoskeletal Radiographs", là một nguồn dữ liệu quan trọng trong lĩnh vực nghiên cứu phân loại, định vị và phân vùng hình ảnh về gãy xương. Đề tài tập trung vào nghiên cứu và phát triển mô hình sử dụng phương pháp học sâu để phân loại, định vị, và phân vùng chấn thương trong hình ảnh X quang. Góp phần cung cấp thông tin quan trọng hỗ trợ quyết định điều trị trong lĩnh vực y tế. Đó cũng chính là lý do nhóm tôi quyết định chọn đề tài này để nghiên cứu và phát triển.

1.1. Mạng nơ ron tích chập.

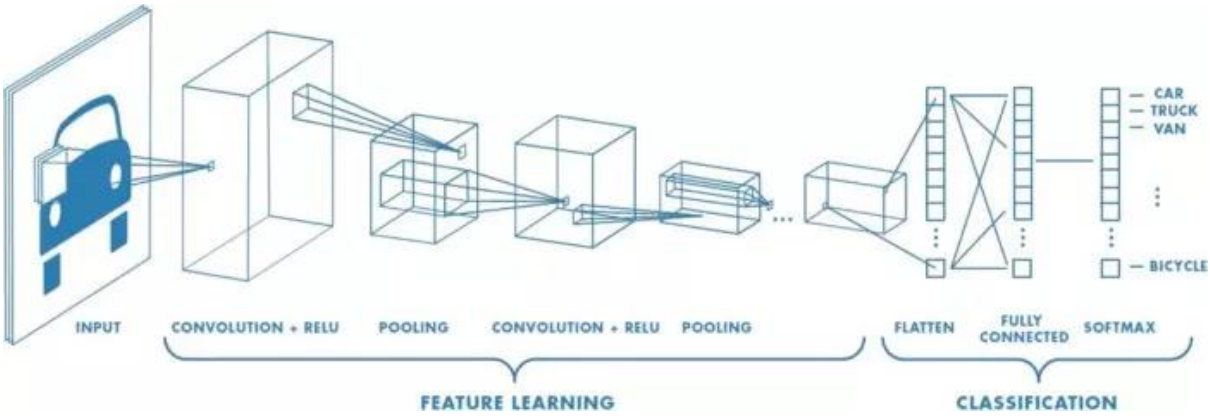
Mạng nơ-ron tích chập (CNN) là một trong những mô hình mạng nơ-ron truyền thẳng đặc biệt và tiên tiến nhất hiện nay. CNN được sử dụng rộng rãi trong các hệ thống nhận diện và xử lý ảnh nhờ vào tốc độ xử lý nhanh và độ chính xác cao. Trong khi các tầng của mạng nơ-ron truyền thống thường được coi là một chiều, CNN lại coi các tầng của mình là ba chiều, bao gồm chiều cao, chiều rộng và chiều sâu. CNN có hai khái niệm quan trọng: kết nối cục bộ và chia sẻ tham số, giúp giảm số lượng trọng số cần được huấn luyện, từ đó tăng tốc độ tính toán.

CNN là một trong những mô hình học sâu phổ biến và có ảnh hưởng nhất trong cộng đồng thị giác máy tính. CNN được ứng dụng trong nhiều bài toán như nhận dạng ảnh, phân tích video, ảnh MRI và các bài toán xử lý ngôn ngữ tự nhiên, và thường giải quyết tốt các bài toán này.

1.1.1. Kiến trúc mạng nơ ron tích chập.

Một kiến trúc CNN bao gồm các lớp: lớp tích chập (convolution layer), lớp gộp (pooling layer) và lớp kết nối đầy đủ (fully connected layer). Giữa các lớp tích chập và lớp gộp thường có các hàm kích hoạt phi tuyến. Khi ảnh được đưa vào mạng, nó

sẽ trải qua lớp tích chập, nơi các giá trị được tính toán và sau đó đi qua một hàm kích hoạt. Tiếp theo, các giá trị này sẽ được truyền qua lớp gộp. Cuối cùng, ảnh sẽ được đưa đến lớp kết nối đầy đủ và đi qua hàm kích hoạt Softmax. Kết quả cuối cùng thường là một vector chứa xác suất phần trăm thuộc về các lớp khác nhau trong bài toán phân loại. Hình 3 là một ví dụ minh họa về kiến trúc đầy đủ của một mạng nơ-ron tích chập:

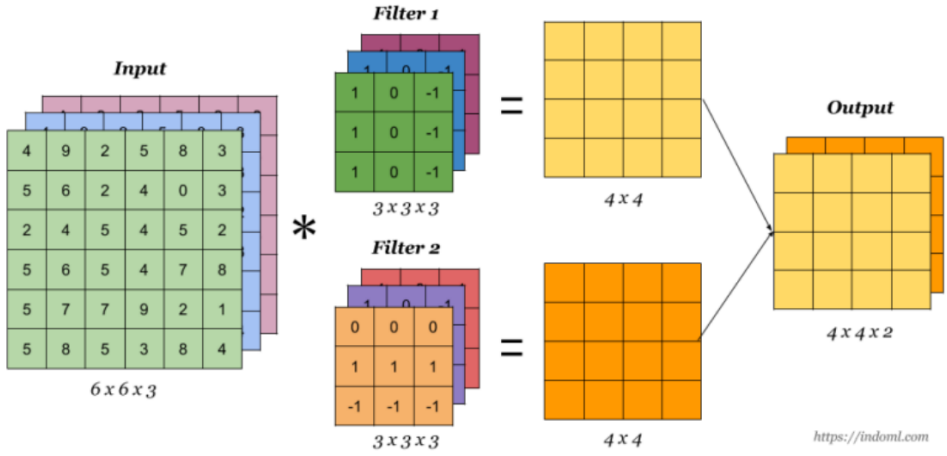


Hình 1: Minh họa kiến trúc mạng nơ-ron tích chập.

Lớp tích chập (convolution layer) là lớp quan trọng nhất và cũng là lớp đầu tiên của mô hình CNN. Chức năng chính của lớp này là phát hiện các đặc trưng không gian hiệu quả. Trong lớp này có bốn thành phần chính: ma trận đầu vào, bộ lọc (filters), trường tiếp nhận (receptive field), và bản đồ đặc trưng (feature map). Lớp tích chập nhận đầu vào là một ma trận ba chiều và một bộ lọc cần phải học. Bộ lọc này sẽ trượt qua từng vị trí trên ảnh để tính tích chập (convolution) giữa bộ lọc và phần tương ứng trên ảnh. Phần tương ứng này trên ảnh gọi là trường tiếp nhận, tức là vùng mà một neuron có thể nhìn thấy để đưa ra quyết định, và ma trận tạo ra bởi quá trình này được gọi là bản đồ đặc trưng.

Khi ảnh được đưa vào mạng, bộ lọc sẽ quét qua toàn bộ ảnh để phát hiện các đặc trưng cơ bản như góc, cạnh, màu sắc và texture, bất kể vị trí của chúng trong ảnh. Do đó, lớp tích chập được coi là một bộ trích xuất đặc trưng (feature detector) vì nó có chức năng chính là phát hiện các đặc trưng cụ thể của ảnh đầu vào.

Khi áp dụng phép tính tích chập cho xử lý hình ảnh, kỹ thuật này giúp biến đổi thông tin đầu vào thành các yếu tố đặc trưng, tương tự như bộ phát hiện đặc trưng như cạnh, hướng, v.v. Cụ thể hơn, tích chập trích xuất đặc trưng của ảnh đầu vào qua các vùng ảnh nhỏ gọi là trường tiếp nhận cục bộ (Local Receptive Field - LRF). Tích chập sẽ tính toán trên các LRF chồng lấp lên nhau, với độ chồng lắp phụ thuộc vào hệ số trượt (stride) của từng kiến trúc mạng cụ thể. Nếu sử dụng hệ số trượt S = α, LRF (bằng kích thước với kernel) sẽ dịch chuyển α đơn vị pixel sau mỗi lần tích chập.



Hình 2: Cách chập của một ảnh RGB

Ảnh đầu vào sau khi thực hiện quá trình tích chập sẽ thu được một bản đồ đặc trưng (feature map). Số lượng trường tiếp nhận cục bộ (LRF) trên ảnh đầu vào sẽ tương ứng với số lượng nơ-ron trên bản đồ đặc trưng, và các bộ lọc (kernel) sẽ là trọng số liên kết mỗi LRF với một nơ-ron trên bản đồ đặc trưng. Lớp tích chập có thể chứa một hoặc nhiều bản đồ đặc trưng. Nếu lớp tích chập có K bản đồ đặc trưng, thì độ sâu của lớp này là K.

Để minh họa rõ hơn quá trình này, hãy xem xét việc trích xuất đặc trưng từ một ảnh đầu vào cụ thể. Giả sử chúng ta có một ảnh với kích thước W1×H1×D1 (W1 và H1 lần lượt là chiều rộng và chiều cao của ảnh, và D1 là chiều sâu, tức là giá trị tại ba kênh màu của ảnh RGB). Một bộ lọc (filter) kích thước F×F sẽ trượt qua ảnh như một cửa sổ trượt (sliding window). Giả sử chúng ta sử dụng K bộ lọc trong trường

hợp này. Trong quá trình xử lý, mỗi bộ lọc sẽ được tính toán với tất cả các LRF trong ảnh với hệ số trượt S = α.

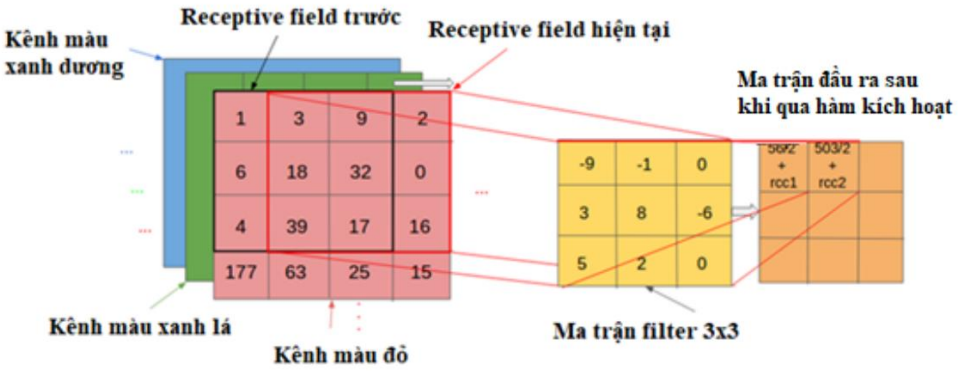
Để cân bằng giữa số bước di chuyển và kích thước của ảnh, đôi khi người ta chèn thêm P pixel với một giá trị màu nhất định (thông thường là 0) xung quanh viền của ảnh. Kết quả cuối cùng là một ma trận đầu ra (feature map) với kích thước W2×H2×D2.

Các Tham Số Của Lớp Tích Chập :

Các tham số cơ bản của lớp tích chập bao gồm kích thước bộ lọc (filter), stride và padding. Trong đó, kích thước bộ lọc là quan trọng nhất vì nó tỷ lệ thuận với số tham số cần học tại mỗi tầng tích chập và quyết định trường tiếp nhận (receptive field) của tầng đó. Kích thước bộ lọc phổ biến thường dùng là 3x3. Thông thường, nên chọn kích thước bộ lọc nhỏ vì những lý do sau:

- Rút trích đặc trưng cục bộ cao: Kích thước nhỏ cho phép mạng nơ-ron phát hiện các đặc trưng tinh tế và chi tiết hơn.
- Phát hiện đặc trưng nhỏ: Các bộ lọc nhỏ giúp phát hiện các chi tiết nhỏ và đặc trưng quan trọng trong ảnh.
- Rút trích đa dạng đặc trưng: Các bộ lọc nhỏ cho phép phát hiện nhiều loại đặc trưng khác nhau, hữu ích cho các tầng sau.
- Kích thước ảnh giảm chậm: Sử dụng bộ lọc nhỏ giúp giảm kích thước ảnh đầu ra từ từ, cho phép xây dựng kiến trúc mạng sâu hơn và học được nhiều hơn.
- Chia sẻ trọng số tốt: Bộ lọc nhỏ giúp chia sẻ trọng số tốt hơn, làm cho mạng nơ-ron tổng quát hóa tốt hơn.

Tham số stride cần lưu ý vì nó xác định số pixel mà bộ lọc sẽ dịch chuyển mỗi khi trượt qua ảnh. Tham số padding cũng rất quan trọng vì nó giúp giữ nguyên kích thước ma trận đầu ra của mỗi tầng tích chập, cho phép xây dựng kiến trúc mạng với số tầng tùy ý.



Hình 3: Minh họa bộ lọc filter.

Hàm Kích Hoạt:

Hàm kích hoạt là một hàm số nhận vào một giá trị đầu vào và kết quả là một giá trị nằm trong một khoảng xác định. Một số hàm kích hoạt phổ biến là Sigmoid, Tanh và ReLU. Hàm kích hoạt rất quan trọng vì nó tăng khả năng dự đoán của mạng neural và giúp mô hình học được các quan hệ phi tuyến phức tạp tiềm ẩn trong dữ liệu. Thông thường, hàm kích hoạt sử dụng ở giữa các tầng tích chập và gộp là hàm ReLU.

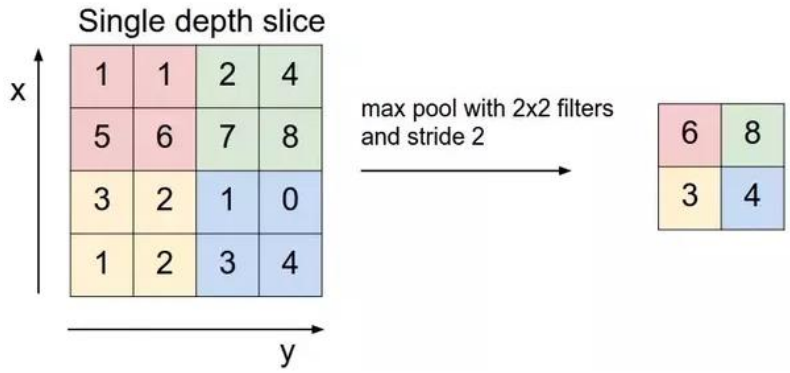
Hàm ReLU có công thức toán học là $f(x) = \max(0, x)$. Hàm ReLU được ưa chuộng vì tính toán đơn giản, giúp hạn chế tình trạng vanishing gradient và cho kết quả tốt hơn. ReLU gán giá trị âm bằng 0 và giữ nguyên giá trị dương của đầu vào, được đặt ngay sau tầng tích chập.

Lớp Gộp (Pooling):

Lớp Gộp được sử dụng sau lớp ReLU theo thiết kế lớp của Đại học Stanford. Pooling giúp giảm số lượng tham số, đơn giản hóa quá trình tính toán của CNN và góp phần giải quyết vấn đề overfitting khi huấn luyện mạng.

Có nhiều toán tử pooling như Sum-pooling, Max-pooling và L2-pooling, nhưng Max-pooling được sử dụng phổ biến nhất vì nó cho kết quả tốt hơn. Max-pooling giúp tạo ra tính bất biến dịch chuyển (translation invariance) cho đặc trưng, giúp mạng phân lớp chính xác dù đối tượng trong ảnh đầu vào có dịch chuyển nhỏ.

Max-pooling chọn giá trị lớn nhất tại mỗi vùng neural của lớp trước, tổng hợp thành lớp sau, giúp chọn ra đặc trưng tốt nhất để xử lý, duy trì khả năng phân lớp chính xác.



Hình 4: Minh họa kỹ thuật pooling trong mô hình CNN

Trong lớp tích chập, có thể có nhiều feature map, mỗi feature map sẽ có một lớp max-pooling. Ví dụ, với lớp đầu vào kích thước [28 × 28], ta thu được ba feature map kích thước [24 × 24] và ba lớp max-pooling kích thước [12 × 12], sử dụng kernel kích thước [5 × 5] và max-pooling lấy giá trị lớn nhất tại mỗi vùng [2 × 2] neural.

Lớp Chuẩn Hóa:

Lớp Chuẩn Hóa (Normalization) điều chỉnh dữ liệu đầu ra của các lớp trong CNN trước khi chuyển tiếp. Trong các kiến trúc CNN lớn và phức tạp, lớp này chuẩn hóa các giá trị của neural trước khi chúng được đưa vào hàm ReLU. Mặc dù hàm ReLU giúp giảm thời gian huấn luyện, nếu trọng số không được điều chỉnh đúng cách, có thể xảy ra hiện tượng "dying ReLU", làm chậm quá trình huấn luyện. Lớp Norm giúp tạo ra các giá trị đầu ra phù hợp để ngăn ReLU rơi vào giá trị 0, tránh hiện tượng gradient gần bằng 0, và giữ cho tốc độ học của mạng ổn định.

Lớp Kết Nối Đầy Đủ:

Tầng cuối cùng của mô hình CNN trong bài toán phân loại ảnh là tầng Fully Connected Layer. Tầng này chuyển ma trận đặc trưng ở tầng trước thành vector chứa xác suất của các đối tượng cần dự đoán. Ví dụ, trong một bài toán phân lớp có 10 lớp,

tầng Fully Connected Layer sẽ chuyển ma trận đặc trưng của tầng trước thành vector có 10 chiều thể hiện xác suất của 10 lớp tương ứng.

1.2. Bài toán phân loại ảnh.

Bài toán phân loại ảnh là một trong những thách thức quan trọng trong lĩnh vực trí tuệ nhân tạo và học máy. Nó yêu cầu hệ thống tự động nhận diện và gán nhãn cho các đối tượng trong hình ảnh, từ đó phân loại chúng vào các nhóm hoặc lớp khác nhau dựa trên các đặc điểm nhận dạng.

Ứng dụng của phân loại ảnh rất rộng rãi, bao gồm nhận diện khuôn mặt, chẩn đoán y khoa qua hình ảnh, phân loại thực vật và động vật, phân tích dữ liệu từ vệ tinh, và nhiều lĩnh vực khác. Để đạt được kết quả chính xác, các mô hình học sâu như mạng nơ-ron tích chập thường được sử dụng vì khả năng học và trích xuất các đặc trưng phức tạp từ dữ liệu hình ảnh.

Quá trình phân loại ảnh bắt đầu bằng việc thu thập và chuẩn bị dữ liệu, bao gồm các bước như tiền xử lý, tăng cường dữ liệu (data augmentation) để cải thiện đa dạng và chất lượng của tập dữ liệu huấn luyện. Sau đó, mô hình được huấn luyện trên dữ liệu này, tối ưu hóa các tham số để nhận diện đặc trưng của từng lớp. Cuối cùng, mô hình được đánh giá thông qua các thước đo như độ chính xác (Accuracy), độ chính xác (Precision), độ phủ (Recall) và chỉ số F1 (F1 score) để đảm bảo hiệu suất và độ tin cậy.

Với sự phát triển không ngừng của công nghệ và dữ liệu, bài toán phân loại ảnh không chỉ giúp cải thiện hiệu suất của các hệ thống tự động mà còn mở ra nhiều cơ hội mới trong nghiên cứu và ứng dụng thực tiễn.

1.3. Phân vùng hình ảnh.

Phân vùng hình ảnh (Image Segmentation) cũng là một nhánh quan trọng trong thị giác máy tính, so với phát hiện đối tượng, phân vùng hình ảnh đã thêm vào một nhiệm vụ là xác định ranh giới chính xác của các đối tượng bằng việc xác định từng đơn vị đầu vào thuộc về một lớp đối tượng hay nó thuộc về phông nền. Hiện nay có

ba loại phân đoạn hình ảnh là phân đoạn ngữ nghĩa, phân đoạn thể hiện và phân đoạn toàn cảnh.

Phân đoạn ngữ nghĩa đề cập đến việc chia các pixel trong một ảnh thành các vùng, các vùng này tương ứng với một lớp ngữ nghĩa nhưng không phân biệt sự khác nhau giữa các đối tượng trong một lớp.

Phân đoạn thể hiện lại có cách thực hiện khác với phân đoạn ngữ nghĩa khi mà nó phân đoạn các pixel trên ảnh thành các vùng và chi tiết đến từng đối tượng trong cùng một lớp, nghĩa là các đối tượng trong cùng một lớp được thể hiện riêng biệt rõ ràng.

Cuối cùng, phân loại toàn cảnh là phương pháp mới nhất với sự kết hợp của hai phương pháp phân đoạn nghữ nghĩa và phân đoạn thể hiện. Mục tiêu của nó là phân đoạn các pixel trên ảnh thành các vùng có ý nghĩa ngữ nghĩa và phân biệt các đối tượng riêng lẻ có trong các vùng này.

1.4. Phát hiện đối tượng.

Phát hiện đối tượng(Object detection) là sự kết hợp giữa phân loại và khoanh vùng bằng cách cung cấp nhãn của các lớp đối tượng và tọa độ hộp giới hạn của các vật thể. Ngày nay các phương pháp phát hiện đối tượng dựa trên học sâu là chủ yếu bởi vì tính chính xác và tốc độ tối ưu mà đó đem lại so với các phương pháp truyền thống. Mô hình mạng thần kinh sâu thường được sử dụng là mạng thần kinh tích chập (CNN) và chúng được chia thành hai loại bao gồm phương pháp phát hiện một giai đoạn và phương pháp phát hiện hai giai đoạn.

1.4.1. Phát hiện hai giai đoạn.

Các bộ phát hiện hai giai đoạn bao gồm hai phần tách biệt là Mạng đề xuất vùng (RPN) và phân loại. Các đặc trưng sau khi trích xuất từ vùng quan tâm (ROI) được truyền vào đầu phân loại để xác định nhãn lớp và đầu hồi quy để xác định các hộp giới hạn.

1.4.2. Phát hiện một giai đoạn.

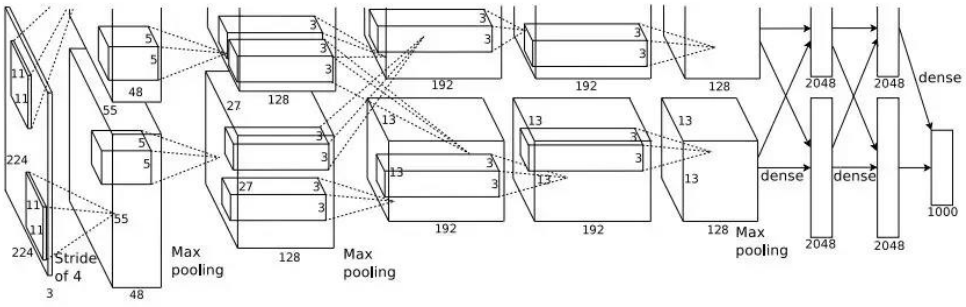
Khác với phát hiện hai giai đoạn, bộ phát hiện một giai đoạn chỉ bao gồm một mạng lan truyền xuôi đầu cuối đơn lẻ thực hiện việc phân loại xác định nhãn lớp và hồi quy xác định hộp giới hạn. Các bộ phát hiện này không có giai đoạn riêng cho việc xác định các vùng đề xuất và thay vào đó là xem tất cả các vị trí trên ảnh là đề xuất tiềm năng cho phát hiện đối tượng. Mỗi vị trí này được sử dụng để dự đoán xác suất lớp, vị trí hộp giới hạn và điểm số tin cậy. Trong đó điểm số tin cậy cho ta biết mức độ chắc chắc cho việc dự đoán lớp của mạng. Hai phân loại của phát hiện một giai đoạn bao gồm phát hiện dựa trên neo và phát hiện dựa trên điểm mấu chốt

CHƯƠNG 2. TỔNG QUAN CÁC MÔ HÌNH HỌC SÂU ÁP DỤNG CHO CÁC BÀI TOÁN PHÂN LOẠI, PHÂN VÙNG, PHÁT HIỆN ĐỐI TƯỢNG TRÊN ẢNH.

2.1. Mạng nơ-ron tích chập cho bài toán phân loại ảnh.

2.1.1. Mạng Alexnet.

Mạng Alexnet được giới thiệu vào năm 2012, mô hình này đạt giải nhất cuộc thi ImageNet Large Scale Visual Recognition Challenge. Kiến trúc mạng Alexnet được minh họa ở hình 5.



Hình 5: Minh họa kiến trúc mạng Alexnet

Trong tầng thứ nhất của AlexNet, kích thước cửa sổ tích chập là 11×11. Kích thước cửa sổ tích chập trong tầng thứ hai được giảm xuống còn 5×5 và sau đó là 3×3. Ngoài ra, theo sau các tầng chập thứ nhất, thứ hai và thứ năm là các tầng gộp cực đại với kích thước cửa sổ là 3×3 và sải bước bằng 2. Sau tầng tích chập cuối cùng là hai tầng kết nối đầy đủ với 4096 đầu ra.

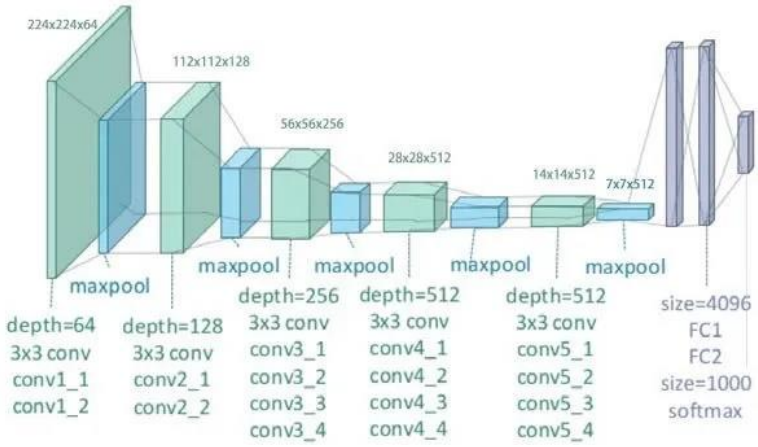
Đặc điểm nổi bật của mạng Alexnet:

- Các mô hình neural network trước khi bài báo ra đời thường sử dụng hàm Tanh làm hàm kích hoạt. Mô hình AlexNet không sử dụng hàm TanH mà giới thiệu một hàm kích hoạt mới là ReLU. ReLU giúp cho quá trình huấn luyện chạy nhanh hơn gấp 6 lần so với kiến trúc tương tự sử dụng Tanh.

- Trong mạng AlexNet sử dụng hàm chuẩn hóa là Local Response Normalization. Các lớp tích chập đầu tiên sử dụng LRN để làm nổi bật các neuron với phản hồi lớn hơn trong một vùng cục bộ, giúp cải thiện khả năng nhận dạng của mạng.
- Mạng AlexNet sử dụng pooling có chồng lấn thay vì non-overlapping pooling, giúp giảm kích thước của đầu ra mà vẫn giữ được nhiều thông tin đặc trưng hơn.
- Dropout được áp dụng tại các lớp fully connected để giảm thiểu quá khớp bằng cách ngẫu nhiên bỏ qua các neurons trong quá trình huấn luyện, giúp làm giảm sự phụ thuộc giữa các neurons.

AlexNet không chỉ nổi bật với thành tích đạt được trong cuộc thi ILSVRC 2012 mà còn tạo ra ảnh hưởng lâu dài trong lĩnh vực thị giác máy tính và học sâu. Các ý tưởng và kỹ thuật mà AlexNet giới thiệu, như việc sử dụng ReLU, LRN, và Dropout, đã trở thành cơ sở cho nhiều nghiên cứu và phát triển trong các mạng nơ-ron sâu hiện đại, thúc đẩy sự tiến bộ của trí tuệ nhân tạo trong nhiều lĩnh vực.

2.1.2. Mạng Visual Geometry Group.



Hình 6: Minh họa kiến trúc mạng VGG.

Trong mạng Visual Geometry Group (VGG), kích thước cửa sổ tích chập trong tất cả các tầng là 3×3 với sải bước bằng 1. Sau mỗi vài tầng tích chập là các tầng gộp cực đại với kích thước cửa sổ 2×2 và sải bước bằng 2. Cụ thể, VGG16 gồm 5 khối

tích chập, mỗi khối chứa từ 2 đến 3 lớp tích chập, và các khối này được nối tiếp nhau bởi các tầng gộp cực đại. Sau khối tích chập cuối cùng là ba tầng fully connected với 4096 đầu ra trong hai tầng đầu tiên và 1000 đầu ra cho tầng cuối cùng.

Đặc điểm nổi bật trong mạng VGG:

- Thiết kế nhất quán và đơn giản**:** Mạng VGG sử dụng kích thước bộ lọc nhỏ (3×3) cho tất cả các tầng tích chập với cùng một cấu hình, giúp đơn giản hóa thiết kế mạng và cho phép mở rộng độ sâu một cách dễ dàng mà vẫn hiệu quả.
- Tăng cường độ sâu của mạng**:** So với các kiến trúc mạng nông hơn, VGG tăng cường độ sâu của mạng lên đến 16 hoặc 19 tầng (trong các phiên bản VGG16 và VGG19), giúp mạng có khả năng học được các đặc trưng phức tạp và trừu tượng từ hình ảnh.
- Sử dụng các lớp pooling nhỏ (2×2)**:** Mạng VGG áp dụng các tầng gộp cực đại với kích thước 2×2 và sải bước bằng 2 sau mỗi vài tầng tích chập để giảm kích thước không gian của đầu ra một cách hiệu quả, đồng thời vẫn giữ được thông tin đặc trưng quan trọng.
- Fully Connected Layers lớn**:** VGG sử dụng các tầng fully connected với 4096 đơn vị, giúp tăng cường khả năng phân loại và học các mối quan hệ phi tuyến tính phức tạp giữa các đặc trưng đã được trích xuất.
- Thiết kế có tính mở rộng và tổng quát cao**:** Với cấu trúc đơn giản và nhất quán, VGG dễ dàng được áp dụng và mở rộng cho nhiều tác vụ thị giác máy tính khác nhau ngoài nhận dạng hình ảnh.

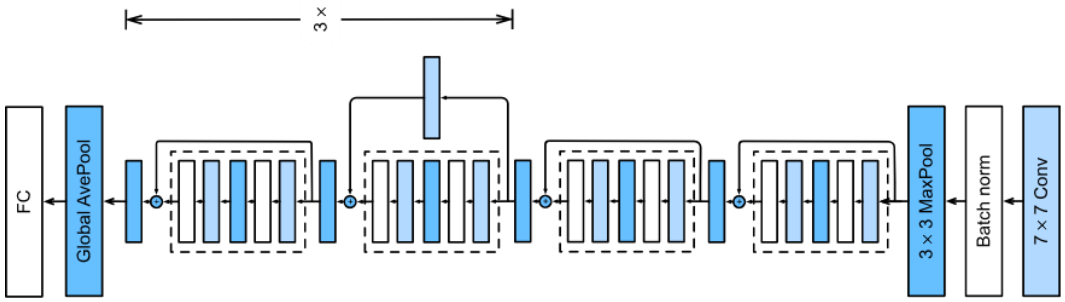
VGG19 là một biến thể mở rộng của mạng VGG, được phát triển với cấu trúc sâu hơn so với phiên bản VGG16. Cả hai mạng VGG16 và VGG19 đều có đặc trưng bởi việc sử dụng các lớp tích chập (convolutional layers) với bộ lọc kích thước nhỏ 3×3 và các lớp gộp cực đại (max pooling) có kích thước 2×2, tuy nhiên, VGG19 chứa nhiều tầng tích chập hơn.

Mạng VGG đã tạo ra ảnh hưởng sâu rộng trong lĩnh vực thị giác máy tính và học sâu. Kiến trúc mạng và các kỹ thuật tiên tiến mà VGG giới thiệu đã trở thành cơ

sở cho nhiều nghiên cứu và phát triển trong các mạng nơ-ron sâu hiện đại, thúc đẩy sự tiến bộ của trí tuệ nhân tạo trong nhiều lĩnh vực.

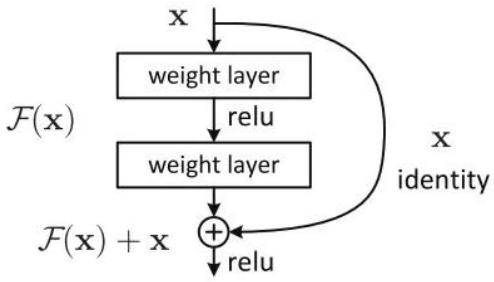
2.1.3. Mạng ResNet.

Mạng ResNet (Residual Network) ra đời vào năm 2015 và đã giành được vị trí thứ 1 trong cuộc thi ILSVRC 2015 với tỉ lệ lỗi top 5 chỉ 3.57%. Không những thế nó còn đứng vị trí đầu tiên trong cuộc thi ILSVRC and COCO 2015 với ImageNet Detection, ImageNet localization, Coco detection và Coco segmentation.Hiện tại có rất nhiều biến thể của kiến trúc ResNet với số lớp khác nhau như ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-152,...Với tên là ResNet theo sau là một số chỉ kiến trúc ResNet với số lớp nhất định. Kiến trúc mạng Resnet được minh họa ở hình 7.



Hình 7: Minh họa kiến trúc mạng Resnet.

Kiến trúc ResNet khác biệt so với các mạng nơ-ron truyền thống nhờ vào các khối Residual, giúp giảm thiểu vấn đề suy giảm gradient trong các mạng sâu. Trong ResNet, các khối Residual cho phép mạng học các đặc trưng ở nhiều độ sâu khác nhau mà không làm mất mát thông tin.



Hình 8: Minh họa khối Residual.

Đặc điểm nổi bật của mạng ResNet:

- Khối Residual (Residual Block): Mỗi khối Residual trong ResNet bao gồm các lớp tích chập, theo sau là một phép cộng (skip connection) giúp truyền thông tin thẳng qua các lớp mà không cần phải đi qua toàn bộ tầng tích chập. Skip connection giúp giảm vấn đề suy giảm gradient bằng cách duy trì thông tin đầu vào trong khi các lớp tích chập chỉ học phần sai khác (residual).
- Kiến trúc tầng: ResNet thường được cấu trúc thành nhiều tầng với các khối Residual. Có nhiều biến thể của ResNet với số lượng tầng khác nhau như ResNet-18, ResNet-34, ResNet-50, ResNet-101, và ResNet-152, thể hiện số lượng lớp tích chập trong mạng. Mỗi khối Residual thường bao gồm hai hoặc ba lớp tích chập nhỏ (3x3).
- Batch Normalization: ResNet sử dụng Batch Normalization sau mỗi lớp tích chập để chuẩn hóa dữ liệu, giúp tăng tốc độ huấn luyện và ổn định mạng.
- ReLU: ResNet sử dụng hàm kích hoạt ReLU (Rectified Linear Unit) để giới thiệu tính phi tuyến tính vào mạng, tương tự như AlexNet. ReLU giúp huấn luyện nhanh hơn và khắc phục vấn đề gradient biến mất.
- Pooling: Mạng ResNet sử dụng các lớp pooling để giảm kích thước đầu ra theo từng giai đoạn, tương tự như các mạng CNN truyền thống. Global Average Pooling (GAP) thường được sử dụng trước lớp fully connected cuối cùng để giảm kích thước của đầu ra thành một vector đặc trưng.

ResNet có nhiều biến thể khác nhau, được phân biệt dựa trên số lượng khối Residual và các tầng tích chập. Các biến thể phổ biến bao gồm ResNet-18, ResNet-34, ResNet-50, ResNet-101 và ResNet-152. Dưới đây là chi tiết về từng biến thể:

ResNet-18:

- Cấu trúc: Bao gồm 4 khối Residual, tổng cộng có 18 tầng tích chập.
- Đặc điểm: Đây là biến thể cơ bản, có cấu trúc nhẹ và tốc độ xử lý nhanh, phù hợp cho các ứng dụng yêu cầu tài nguyên tính toán hạn chế. ResNet-18 mang lại hiệu suất tốt trong nhiều bài toán thị giác máy tính cơ bản.

ResNet-34:

- Cấu trúc: Bao gồm 4 khối Residual với 34 tầng tích chập.
- Đặc điểm: Biến thể này sâu hơn ResNet-18, cho phép mạng học được nhiều đặc trưng phức tạp hơn. ResNet-34 là sự lựa chọn cân bằng giữa độ sâu và hiệu suất, thích hợp cho các bài toán thị giác phức tạp hơn mà không đòi hỏi quá nhiều tài nguyên tính toán.

ResNet-50:

- Cấu trúc: Bao gồm 4 khối Residual với 50 tầng tích chập, sử dụng các khối Bottleneck để tăng hiệu suất.
- Đặc điểm: Đây là biến thể được sử dụng rộng rãi, cân bằng giữa độ sâu và tài nguyên tính toán. Các khối Bottleneck giúp giảm số lượng tham số và tăng tốc độ huấn luyện mà vẫn duy trì khả năng học tập các đặc trưng phức tạp. ResNet-50 thường được áp dụng trong nhiều bài toán thị giác máy tính như nhận diện ảnh và phân loại đối tượng.

ResNet-101:

- Cấu trúc: Bao gồm 4 khối Residual với 101 tầng tích chập, sử dụng các khối Bottleneck.
- Đặc điểm: Với độ sâu lớn hơn, ResNet-101 có khả năng học tập tốt hơn trong các bài toán phức tạp, đặc biệt là những bài toán yêu cầu nhận diện chi tiết

cao. Tuy nhiên, điều này cũng đồng nghĩa với việc yêu cầu tài nguyên tính toán cao hơn so với ResNet-50.

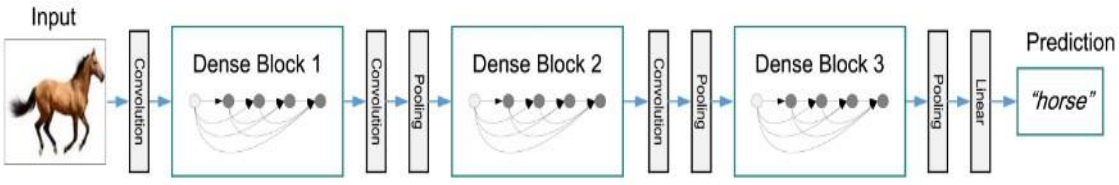
ResNet-152:

- Cấu trúc: Bao gồm 4 khối Residual với 152 tầng tích chập, sử dụng các khối Bottleneck.
- Đặc điểm: Đây là biến thể sâu nhất của ResNet, cho phép mạng học được rất nhiều đặc trưng phức tạp và chi tiết. ResNet-152 thường được sử dụng trong các bài toán thị giác máy tính tiên tiến và đòi hỏi tài nguyên tính toán lớn. Mạng này đặc biệt hữu ích trong các ứng dụng yêu cầu độ chính xác cao như phân loại ảnh phức tạp và nhận diện đối tượng trong các tình huống khó khăn.

ResNet đã tạo ra sự đột phá quan trọng trong việc huấn luyện các mạng nơ-ron sâu hơn, vượt qua các giới hạn trước đó nhờ vào thiết kế của các khối Residual. Nó đã mở ra nhiều hướng nghiên cứu mới và được áp dụng rộng rãi trong nhiều lĩnh vực của thị giác máy tính và học sâu, bao gồm nhận dạng hình ảnh, phân loại, và phát hiện đối tượng.

2.1.4. Mạng DenseNet.

Mạng DenseNet (Dense Convolutional Network) được giới thiệu vào năm 2017 và đã giành được nhiều sự chú ý trong cộng đồng thị giác máy tính. Kiến trúc mạng DenseNet được minh họa ở hình 9.



Hình 9: Kiến trúc DenseNet.

DenseNet khác biệt so với các mạng nơ-ron truyền thống nhờ vào việc kết nối trực tiếp tất cả các lớp với nhau trong từng khối Dense (Dense Block). Điều này có

nghĩa là mỗi lớp nhận đầu vào từ tất cả các lớp trước đó và truyền đầu ra đến tất cả các lớp tiếp theo trong cùng một khối.

Đặc điểm nổi bật của mạng DenseNet:

- Kết nối dày đặc (Dense Connections): Thay vì chỉ kết nối đầu ra của một lớp đến lớp tiếp theo, DenseNet kết nối đầu ra của một lớp đến tất cả các lớp tiếp theo trong cùng khối Dense. Các kết nối dày đặc này giúp tái sử dụng đặc trưng đã học qua nhiều lớp, giảm thiểu vấn đề biến mất gradient và cải thiện khả năng học tập của mạng.
- Khối Dense (Dense Block): Trong mỗi khối Dense, các lớp tích chập nhỏ (thường là 1x1 và 3x3) được sử dụng và kết nối trực tiếp đến tất cả các lớp tiếp theo trong khối. Các khối Dense giúp tăng tính lưu trữ và tái sử dụng thông tin, làm giảm số lượng tham số cần thiết.
- Bottleneck Layer và Transition Layer: Bottleneck Layer thường là lớp tích chập 1x1 được sử dụng trước các lớp tích chập lớn hơn để giảm số lượng đặc trưng và giảm chi phí tính toán. Transition Layer được sử dụng giữa các khối Dense, thường bao gồm một lớp tích chập 1x1 và một lớp pooling để giảm kích thước đặc trưng.

DenseNet có nhiều biến thể khác nhau, thường được phân biệt bằng số lượng khối Dense và các tầng:

DenseNet-121:

- Cấu trúc: Bao gồm 4 khối Dense với tổng cộng 121 tầng tích chập.
- Đặc điểm: Đây là một biến thể cơ bản, cân bằng giữa số lượng tầng và hiệu suất. DenseNet-121 thường được sử dụng cho các bài toán không quá phức tạp nhưng vẫn đòi hỏi độ chính xác cao.

DenseNet-169:

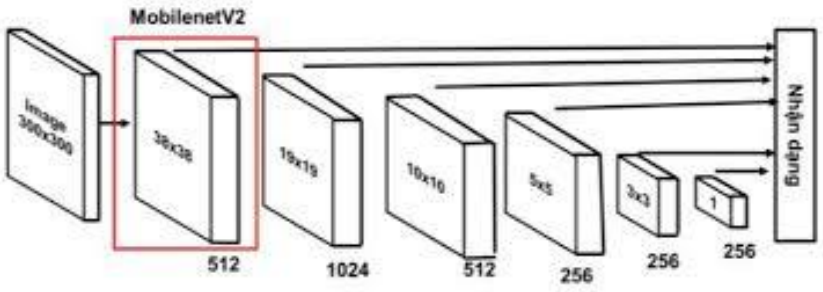
- Cấu trúc: Bao gồm 4 khối Dense với 169 tầng tích chập.

- Đặc điểm: Biến thể này có độ sâu lớn hơn DenseNet-121, cho phép mô hình học được nhiều đặc trưng hơn, từ đó cải thiện khả năng phân loại và nhận diện trong các bài toán phức tạp hơn.

DenseNet đã chứng minh sự hiệu quả trong việc tối ưu hóa quá trình học tập của các mạng nơ-ron sâu, giúp đạt được hiệu suất cao hơn với số lượng tham số ít hơn. Kiến trúc DenseNet đã được ứng dụng rộng rãi trong các lĩnh vực như nhận dạng hình ảnh, phân loại, và phát hiện đối tượng, tạo nền tảng cho nhiều mô hình tiên tiến trong lĩnh vực trí tuệ nhân tạo.

2.1.5. Mạng MobileNet

Mạng MobileNet được giới thiệu vào năm 2017, nổi bật với khả năng triển khai hiệu quả trên các thiết bị di động và nhúng. Kiến trúc mạng MobileNet được minh họa ở hình 10.



Hình 10: Kiến trúc mạng MobileNetV2.

MobileNet được thiết kế để đạt hiệu suất cao với yêu cầu tính toán thấp bằng cách sử dụng các tầng tích chập chuyên biệt, giúp giảm thiểu số lượng tham số và hoạt động tính toán cần thiết mà vẫn duy trì độ chính xác cao.

Đặc điểm nổi bật của mạng MobileNet:

- Tầng tích chập sâu tách biệt (Depthwise Separable Convolution): Tầng tích chập sâu tách biệt là cải tiến chính của MobileNet, chia một lớp tích chập thành hai giai đoạn: tích chập độ sâu (depthwise convolution) và tích chập điểm

(pointwise convolution). Tích chập độ sâu áp dụng một bộ lọc cho mỗi kênh đầu vào riêng lẻ, trong khi tích chập điểm sử dụng bộ lọc 1×1 để kết hợp các kênh đầu ra từ tích chập độ sâu. Điều này giúp giảm số lượng phép tính và tham số đáng kể so với tích chập truyền thống.

- Kiến trúc tầng: MobileNet xây dựng mạng từ các khối tích chập sâu tách biệt, nối tiếp bởi các lớp tích chập điểm để tạo ra một mạng hiệu quả về mặt tính toán. Các khối này được sử dụng tuần tự với các lớp chuẩn hóa và hàm kích hoạt để tạo ra toàn bộ mạng.
- Hệ số nhân chiều rộng (Width Multiplier): Hệ số nhân chiều rộng (α) điều chỉnh số lượng kênh trong từng lớp tích chập, giúp cân bằng giữa tốc độ tính toán và độ chính xác. Giá trị α nhỏ hơn 1 sẽ giảm số lượng kênh, dẫn đến giảm số lượng phép tính nhưng có thể ảnh hưởng đến độ chính xác.
- Hệ số nhân độ phân giải (Resolution Multiplier): Hệ số nhân độ phân giải (ρ) điều chỉnh độ phân giải của hình ảnh đầu vào, cho phép giảm kích thước của các đầu vào để tăng tốc độ tính toán, mặc dù có thể làm giảm độ chính xác.

MobileNet có nhiều biến thể khác nhau, thường được phân biệt bởi các phiên bản và cải tiến nhằm tối ưu hóa hơn nữa hiệu suất và độ chính xác:

MobileNetV1:

- Cấu trúc: MobileNetV1 sử dụng các khối tích chập chiều sâu (depthwise separable convolutions), giúp giảm đáng kể số lượng tham số và tính toán so với các mạng CNN truyền thống.
- Đặc điểm: MobileNetV1 tập trung vào hiệu quả tính toán và tài nguyên, làm cho nó phù hợp cho các ứng dụng trên thiết bị di động và nhúng. Các khối tích chập chiều sâu giúp mạng nắm bắt các đặc trưng quan trọng mà không tăng quá nhiều số lượng tham số.

MobileNetV2:

- Cấu trúc: MobileNetV2 tiếp tục sử dụng các khối tích chập chiều sâu nhưng thêm vào đó là các khối tích chập nhân rộng (inverted residuals) và linear bottlenecks. Kiến trúc này giúp duy trì lượng thông tin tốt hơn qua các tầng.
- Đặc điểm: Với các khối inverted residual và linear bottlenecks, MobileNetV2 cải thiện hiệu suất và độ chính xác so với MobileNetV1. Nó vẫn giữ được tính hiệu quả cao, phù hợp cho các ứng dụng thời gian thực và thiết bị có hạn chế về tài nguyên.

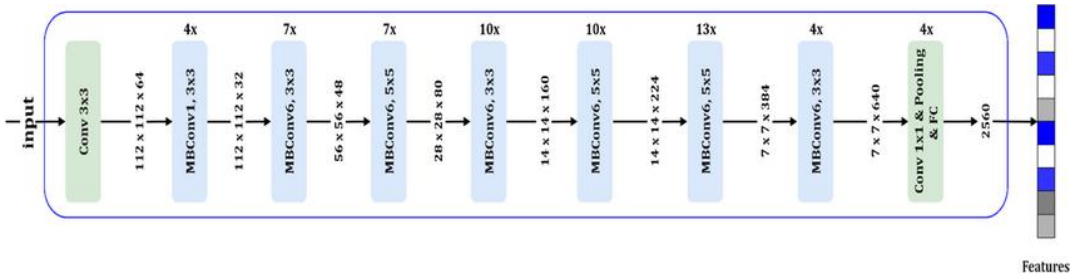
MobileNetV3:

- Cấu trúc: MobileNetV3 sử dụng một sự kết hợp giữa các khối tích chập nhân rộng và các kỹ thuật tối ưu hóa khác như squeeze-and-excitation modules và swish activation function.
- Đặc điểm: MobileNetV3 cải thiện thêm về hiệu suất và độ chính xác, cung cấp hai biến thể chính: MobileNetV3-Small và MobileNetV3-Large. MobileNetV3-Small được tối ưu hóa cho các ứng dụng yêu cầu tài nguyên cực kỳ thấp, trong khi MobileNetV3-Large hướng đến những ứng dụng yêu cầu hiệu suất cao hơn.

MobileNet đã chứng minh sự hiệu quả trong việc tối ưu hóa các mạng nơ-ron để triển khai trên các thiết bị di động và nhúng, nơi mà tài nguyên tính toán và bộ nhớ hạn chế. MobileNet đã mở ra một hướng đi mới cho việc triển khai các mô hình học sâu trên các thiết bị nhỏ gọn, thúc đẩy sự phát triển của học sâu và thị giác máy tính trong các ứng dụng thực tế.

2.1.6. Mạng EfficientNet.

Mạng EfficientNet được giới thiệu vào năm 2019 bởi Google AI, nổi bật với cách tiếp cận tiên tiến trong việc mở rộng kích thước mạng. Kiến trúc mạng EfficientNet được minh họa ở hình 13:



Hình 11: Kiến trúc mạng EfficientNet B7.

Đặc điểm nổi bật của mạng EfficientNet:

- Cách tiếp cận Compound Scaling: EfficientNet giới thiệu một cách tiếp cận mới gọi là compound scaling. Thay vì chỉ tăng kích thước độ sâu (depth), chiều rộng (width), hoặc độ phân giải đầu vào (resolution) của mạng riêng lẻ, EfficientNet mở rộng các chiều này theo cách đồng thời và có hệ thống. Công thức compound scaling là:

$$d = \alpha^{\emptyset}, w = \beta^{\emptyset}, r = \gamma^{\emptyset}$$

Trong đó $\emptyset$ là một biến số tự do để cân bằng tài nguyên tính toán, còn α , β, và γ là các hệ số để mở rộng độ sâu, chiều rộng, và độ phân giải.

- Kiến trúc Mobile Inverted Bottleneck: EfficientNet sử dụng các khối Mobile Inverted Bottleneck (MBConv) từ MobileNetV2, có các lớp tích chập 3x3 hoặc 5x5 với yếu tố mở rộng kênh và skip connection. Các khối MBConv giúp giảm số lượng tham số và cải thiện hiệu quả tính toán, đồng thời giúp duy trì các đặc trưng cần thiết cho việc học.
- Sử dụng Swish Activation: EfficientNet sử dụng hàm kích hoạt Swish (được định nghĩa là $x \cdot \sigma(x)$, trong đó σ là hàm sigmoid), thay vì ReLU. Swish giúp mô hình học các đặc trưng phức tạp hơn và cải thiện hiệu suất so với các hàm kích hoạt truyền thống như ReLU.
- Batch Normalization và Squeeze-and-Excitation (SE) Block: EfficientNet kết hợp Batch Normalization trong các khối MBConv để ổn định quá trình huấn

luyện. SE Block giúp cải thiện khả năng học của mạng bằng cách chú trọng vào các đặc trưng quan trọng và giảm thiểu các đặc trưng không cần thiết.

EfficientNet có nhiều biến thể khác nhau, thường được phân biệt bởi hệ số ∅ dùng để mở rộng các chiều của mạng:

EfficientNet-B0:

- Đây là mô hình cơ sở với các tham số được tối ưu hóa theo kiến trúc ban đầu.
- Phù hợp cho các ứng dụng cần cân bằng giữa hiệu suất và chi phí tính toán.

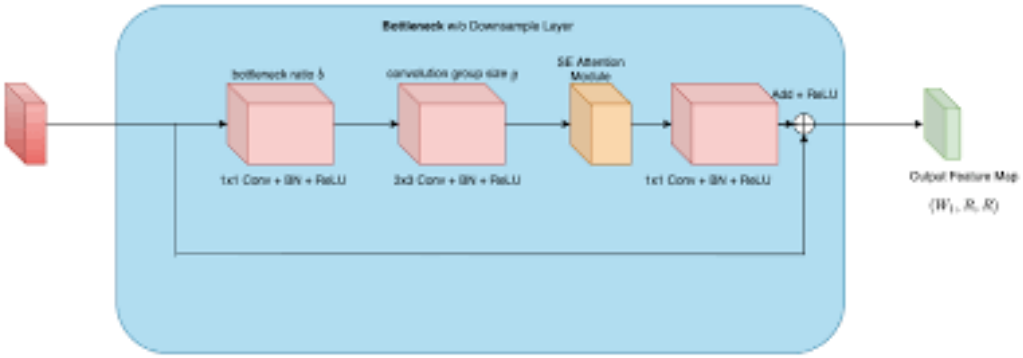
EfficientNet-B1 đến EfficientNet-B7:

- Các biến thể này tăng dần hệ số ∅, do đó mở rộng độ sâu, chiều rộng, và độ phân giải của mạng theo tỉ lệ hợp chất.
- Mô hình càng cao (như B7), càng có độ phức tạp và khả năng học cao hơn, thích hợp cho các bài toán yêu cầu độ chính xác cao và tài nguyên tính toán lớn.

EfficientNet đã chứng minh sự hiệu quả vượt trội trong việc tối ưu hóa hiệu suất mạng nơ-ron sâu, đặc biệt trong các tác vụ về nhận dạng hình ảnh và phân loại. Kiến trúc EfficientNet đạt được điểm số cao trên các tập dữ liệu chuẩn với số lượng tham số và tài nguyên tính toán thấp hơn đáng kể so với các mạng nơ-ron truyền thống. Với cách tiếp cận compound scaling, EfficientNet mở ra một hướng đi mới cho việc thiết kế các mô hình học sâu hiệu quả và mạnh mẽ hơn.

2.1.7. Mạng RegNet.

Mạng RegNet (Regularized Network) được giới thiệu vào năm 2020 bởi Facebook AI Research. Đây là một trong những mạng nơ-ron hiện đại với kiến trúc được tối ưu hóa nhằm đạt hiệu suất cao với khả năng mở rộng linh hoạt. Kiến trúc mạng RegNet được minh họa ở hình 12.



Hình 12: Minh họa kiến trúc RegNet.

RegNet được thiết kế dựa trên một quy trình tối ưu hóa có hệ thống, không phải chỉ dựa trên kinh nghiệm hoặc thử nghiệm trực tiếp. Điều này tạo nên các mạng có cấu trúc đều đặn, dễ điều chỉnh và có hiệu suất cao. RegNet đưa ra một cách tiếp cận gọi là Design Space Exploration, trong đó việc tìm kiếm và tối ưu hóa kiến trúc mạng được thực hiện một cách tự động dựa trên các nguyên tắc toán học.

Các đặc điểm nổi bật trong kiến trúc mạng RegNet:

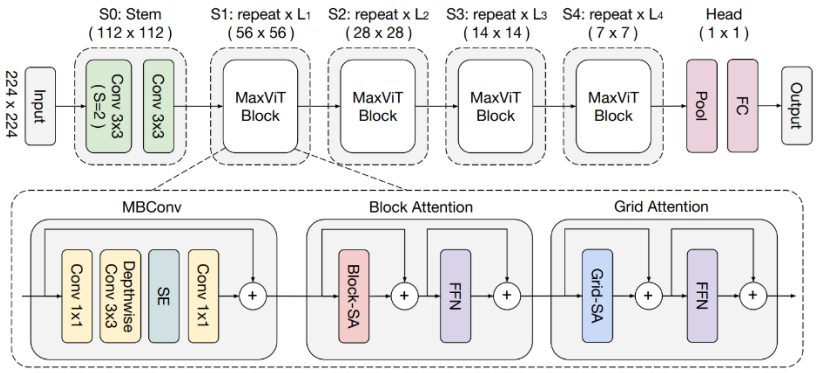
- Khối Xây dựng Basic Block: Mạng RegNet sử dụng các khối cơ bản với lớp tích chập 3x3, kèm theo các khối Bottleneck. Các khối này được sắp xếp theo một cấu trúc đơn giản nhưng hiệu quả. Các khối Basic Block giúp giảm số lượng tham số cần thiết, duy trì tính hiệu quả và khả năng học tốt của mạng.
- Linear Scaling Rules: RegNet áp dụng các quy tắc mở rộng tuyến tính (linear scaling rules) để mở rộng số lượng kênh, độ sâu và độ rộng của mạng. Điều này tạo ra một loạt các biến thể của RegNet, từ các mô hình nhỏ gọn đến các mô hình lớn và mạnh mẽ. Quy tắc này cho phép dễ dàng điều chỉnh cấu trúc mạng để phù hợp với các yêu cầu khác nhau về tài nguyên tính toán và độ chính xác.
- khối Squeeze-and-Excitation (SE) : RegNet có thể tích hợp các khối Bottleneck để giảm số lượng đặc trưng và giảm chi phí tính toán. Khối SE (Squeeze-and-Excitation) được sử dụng để tăng cường khả năng học tập của

mạng bằng cách chú trọng vào các đặc trưng quan trọng và giảm thiểu các đặc trưng không cần thiết.

RegNet đã chứng minh sự hiệu quả trong việc tạo ra các mạng nơ-ron với hiệu suất cao mà vẫn giữ được tính dễ điều chỉnh và tối ưu hóa. Với cách tiếp cận thiết kế dựa trên nguyên tắc, RegNet giúp các nhà nghiên cứu và phát triển dễ dàng tùy chỉnh cấu trúc mạng cho các ứng dụng cụ thể.

2.1.8. Mạng MaxVit.

Mạng MaxVit (Maximal Vision Transformer) được giới thiệu vào năm 2022 và đã thu hút nhiều sự chú ý trong cộng đồng thị giác máy tính. Kiến trúc mạng MaxVit được minh họa ở hình 2.8.



Hình 13: Kiến trúc mô hình MaxvitT

MaxVit khác biệt so với các mạng nơ-ron truyền thống bằng cách kết hợp các yếu tố tiên tiến từ các mô hình thị giác trước đó như Vision Transformer (ViT) và Swin Transformer, cùng với cơ chế Attention đa trục để tăng cường khả năng học và tổng quát hóa.

Đặc điểm nổi bật của mạng MaxVit:

- Attention Đa Trục (Multi-axis Attention): MaxVit áp dụng cơ chế Attention trên cả hai trục không gian và trục kênh (Channel). Điều này cho phép mạng mô hình hóa mối quan hệ không chỉ trong không gian hình ảnh mà còn giữa

các kênh khác nhau. Attention đa trục giúp mô hình tập trung vào các đặc trưng quan trọng từ cả không gian và kênh, cải thiện khả năng nhận diện và phân loại.

- Khối MaxViT (MaxViT Block): Trong mỗi khối MaxViT, mạng sử dụng cơ chế Attention kết hợp với tích chập để xử lý các đặc trưng ở nhiều tỉ lệ và mức độ chi tiết khác nhau. Các khối MaxViT kết hợp cơ chế Self-Attention của Transformer với các lớp tích chập để trích xuất và tổng hợp đặc trưng một cách hiệu quả.
- Lớp Bottleneck và Layer Transition: Lớp Bottleneck được sử dụng để giảm số lượng đặc trưng và chi phí tính toán trước khi áp dụng cơ chế Attention hoặc tích chập lớn hơn. Các lớp Transition được dùng để thay đổi kích thước của đặc trưng giữa các khối MaxViT, thường kết hợp tích chập và pooling để điều chỉnh kích thước không gian và số lượng kênh.
- Global Average Pooling (GAP): Trước khi đến lớp fully connected cuối cùng, MaxVit sử dụng Global Average Pooling để giảm kích thước của đầu ra thành một vector đặc trưng, giúp giảm số lượng tham số và khả năng quá khớp.

MatVit có nhiều biến thể khác nhau.

- ***MaxVit-T (Tiny):*** Bao gồm các khối MaxViT với cấu trúc gọn nhẹ, thích hợp cho các ứng dụng yêu cầu tài nguyên tính toán ít. Cân bằng giữa kích thước mô hình và hiệu suất, phù hợp với các thiết bị biên và ứng dụng thời gian thực.
- ***MaxVit-S (Small):*** Có cấu trúc phức tạp hơn MaxVit-T, cung cấp hiệu suất cao hơn, thích hợp cho các ứng dụng cần độ chính xác cao hơn. Được sử dụng rộng rãi trong các tác vụ yêu cầu xử lý thông tin lớn và chi tiết.
- ***MaxVit-B (Base):*** Là biến thể tiêu chuẩn với cấu trúc phức tạp hơn, MaxVit-B mang lại hiệu suất cao, thích hợp cho các ứng dụng trong công nghiệp và nghiên cứu. Được sử dụng trong các hệ thống lớn và các tác vụ yêu cầu hiệu suất mạnh mẽ.
- ***MaxVit-L (Large):*** Biến thể lớn nhất với cấu trúc sâu hơn, MaxVit-L có khả năng học tập và nắm bắt các đặc trưng chi tiết hơn. Đòi hỏi tài nguyên tính

toán cao nhưng mang lại hiệu suất vượt trội trong các bài toán phức tạp và dữ liệu lớn.

MaxVit đã chứng minh sự hiệu quả trong việc cải thiện khả năng nhận diện và phân loại của các mạng nơ-ron sâu, đạt được hiệu suất cao hơn với kiến trúc phức tạp và tối ưu hóa khả năng học tập. Kiến trúc MaxVit đã được áp dụng rộng rãi trong các lĩnh vực như nhận dạng hình ảnh, phân loại, và phát hiện đối tượng, đóng góp quan trọng vào sự phát triển của trí tuệ nhân tạo.

2.2. Bài toán phân vùng hình ảnh và phát hiện đối tượng.

2.2.1. YOLOv8.

YOLOv8 là một phiên bản cải tiến từ mô hình YOLOv5, được phát triển bởi Ultralytics và là mô hình YOLO mới nhất được phát hành bởi họ ra mắt vào tháng 1 năm 2023. YOLOv8 hỗ trợ cả tác vụ phát hiện đối tượng và phân đoạn hình ảnh, nó được phát triển với 5 biến thể gồm nano (n), small (s), medium (m), large (l) và extra large (x). Mỗi biến thể khác nhau ở số tham số và kích thước tăng dần từ nano đến extra large tương ứng với độ chính xác theo metric mAP50 trên tập dữ liệu MS COCO cũng tăng dần.

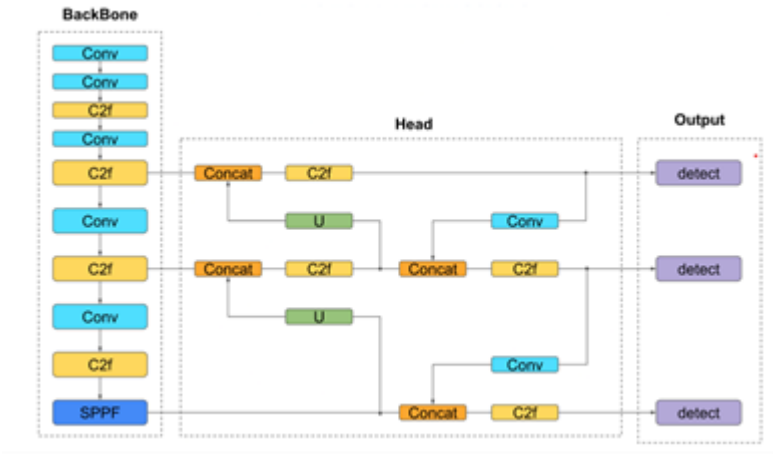
So với phiên bản YOLOv5, YOLOv8 sử dụng một backbone tương tự với một vài thay đổi trên module C2f. Kiến trúc của YOLOv8 gồm 3 phần chính:

- Backbone: sử dụng CSPDarknet53, một phiên bản được sửa đổi của kiến trúc Darknet. Kiến trúc CSPDarknet53 giới thiệu một kết nối Cross-Stage Partial (CSP) mới, tăng cường luồng thông tin giữa các giai đoạn khác nhau của mạng và cải thiện luồng gradient trong quá trình đào tạo, tăng cường khả năng học tập và hiệu quả.
- Neck: YOLOv8 giới thiệu PANet (Path Aggregation Network), một mạng kim tự tháp đặc trưng (feature pyramid network) tạo điều kiện cho luồng thông tin chảy qua các quy mô khác nhau. PANet tăng cường khả năng của mô hình trong việc xử lý các đối tượng có quy mô đa dạng một cách hiệu quả hơn.

- Head: YOLOv8 sử dụng một mô hình anchor-free với một đầu ra độc lập để xử lý độc lập các nhiệm vụ về tính khả thi của đối tượng, phân loại và hồi quy. Thiết kế này cho phép mỗi nhánh tập trung vào nhiệm vụ của mình và cải thiện độ chính xác tổng thể của mô hình. Trong lớp đầu ra của YOLOv8, họ sử dụng hàm sigmoid làm hàm kích hoạt cho điểm về tính khả thi của đối tượng, thể hiện xác suất rằng khung giới hạn chứa một đối tượng. Nó sử dụng hàm softmax cho các xác suất lớp, thể hiện xác suất các đối tượng thuộc vào từng lớp có thể có.

So với các phiên bản YOLO trước đó, YOLOv8 đem đến một số thay đổi mới bao gồm:

- Anchor-free detection: YOLOv8 đã loại bỏ việc sử dụng anchor box, dự đoán tâm của vật thể trực tiếp thay vì độ lệch từ một anchor box đã biết. Điều này giúp tăng tốc độ Non-Maximum Suppression(NMS) vì số lượng hộp dự đoán đã giảm đi
- Lớp convolutional mới: các lớp convolutional trong kiến trúc của YOLO giúp phát hiện các đặc trưng từ ảnh đầu vào và kết quả của nó sẽ được đưa vào các lớp tiếp theo cho việc tạo bounding box và dự đoán lớp cho mỗi đối tượng được phát hiện. YOLOv8 sử dụng một lớp convolutional mới là C2f, nó đã thay thế lớp C3 của YOLOv5. Lớp C2f này được nối với đầu ra của tất cả lớp Bottleneck.
- Tăng cường dữ liệu Mosaic: YOLOv8 hỗ trợ nhiều kỹ thuật tăng cường dữ liệu cho quá trình huấn luyện và kỹ thuật mới được giới thiệu là Mosaic. Với kỹ thuật này, bốn hình ảnh khác nhau được ghép lại và đưa vào đầu vào cho mô hình giúp mô hình học được các đối tượng ở vị trí mới và trong trường hợp bị che khuất.



Hình 14: Minh họa kiến trúc YOLOv8.

2.2.2. YOLOv9.

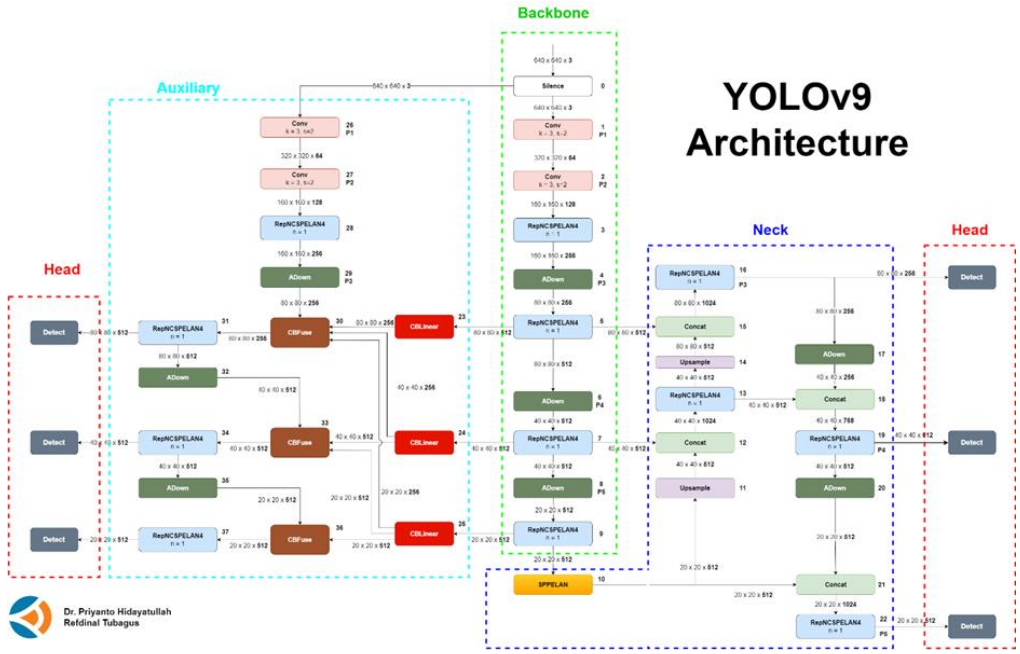
YOLOv9 tiếp tục là một phiên bản mới so với phiên bản v8 tiền nhiệm trước đó, ra mắt vào tháng 1 năm 2024 bao gồm 5 biến thể lần lượt là “t”, “s”, “m”, “c” và “e”. YOLOv9 đã cải tiến về tốc độ và độ chính xác trên bộ dữ liệu MS COCO, đồng thời giới thiệu về các kỹ thuật mới như PGI, GELAN.

Với lối kiến trúc vẫn gồm ba phần chính, YOLOV9 đã cải tiến thêm một số kỹ thuật mới:

- Backbone: sử dụng các khối RepNCSP-ELAN 4 để trích xuất đặc trưng đồng thời kết hợp các khối RepNBottleneck và RepNCSP để biểu diễn đặc trưng phân cấp. Bao gồm các khối Adown để down-sampling hiệu quả trong khi vẫn bảo toàn thông tin về mặt không gian.
- Neck: tích hợp các module PANet để tăng cường biểu diễn và tổng hợp các đặc trưng.
- Head: khởi tạo đầu ra cho các tác vụ phát hiện đối tượng bao gồm dự đoán tọa độ hộp giới hạn thông qua hồi quy, ước lượng xác xuất của đối tượng thuộc vào lớp tương ứng để dự đoán lớp cho đối tượng và xác định điểm số độ tin cậy cho sự thể hiện của đối tượng.

Bên cạnh đó, YOLOv9 giới thiệu tới một số thay đổi chính như:

- Programmable Gradient Information(PGI): để giải quyết vấn đề gradient không đáng tin cậy, YOLOv9 đề xuất PGI bao gồm 3 thành phần. Nhánh chính giúp loại bỏ sự cần thiết của thành phần phụ trợ trong quá trình suy luận, đảm bảo mô hình hoạt động hợp lý và hiệu quả đồng thời duy trì hiệu suất, giảm chi phí tính toán. Nhánh phụ có thể đảo ngược tạo ra các gradient đáng tin cậy và tạo điều kiện cho việc cập nhật trọng số chính xác, giảm thiểu việc mất và tắc nghẽn thông tin.Thông tin phụ trợ đa cấp sử dụng các mạng chuyên dụng kết hợp thông tin gradient của các lớp, giải quyết việc mất thông tin, đảm bảo mô hình hiểu được dữ liệu một cách đầy đủ.
- Generalized Efficient Layer Aggregation Network (GELAN): đây là kiến trúc kết hợp giữa CSPNet với đặc trưng lập kế hoạch đường đi của gradient và ELAN với đặc trưng tối ưu hóa tốc độ trong quá trình suy luận. Điều này giúp giảm nhẹ mô hình, tăng tốc độ suy luận và độ chính xác.



Hình 15: Minh họa kiến trúc YOLOv9.

2.2.3. YOLOv10.

Tính đến thời điểm hiện tại khi chúng tôi thực hiện khóa luận này, YOLOV10 chính là họ mô hình YOLO mới nhất được phát hành vào tháng 5 năm 2024 và chỉ mới hỗ trợ tác vụ phát hiện đối tượng. YOLOv10 được giới thiệu bao gồm 6 biến thể "n", "s", "m", "b", "l" và "x".

Cải tiến mới mà YOLOv10 giới thiệu bao gồm:

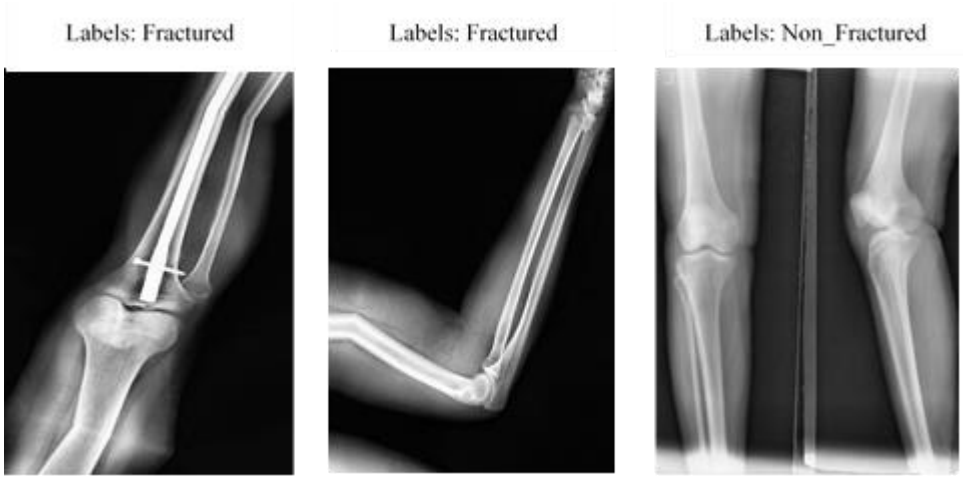
- Chiến lược huấn luyện NMS với việc gán nhãn kép: NMS là công việc xử lý để loại bỏ các hộp giới hạn dư thừa, đảm bảo mỗi đối tượng chỉ có một hộp giới hạn duy nhất và xuất hiện ở các phiên bản YOLO tiền nhiệm. Hạn chế của nó là làm tăng chi phí tính toán và thời gian suy luận nhất là khi có nhiều hộp giới hạn. YOLOv10 sử dụng phương pháp gán nhãn kép tận dụng ưu điểm của gán nhãn one-to-one và one-to-many. Trong quá trình huấn luyện sẽ sử dụng cả hai phương pháp để tận dụng điểm mạnh nhưng khi suy luận chỉ sử dụng one-to-one để dự đoán tránh việc xử lý NMS.
- Spatial-channel decoupled downspamling: các mô hình YOLO tiêu chuẩn thường dùng phép tích chập 3x3 với bước nhảy là 2 để downsampling về mặt không gian và tăng số kênh, việc này tạo ra chi phí tính toán và các tham số không đáng kể. YOLOv10 áp dụng Spatial-channel decoupled downspamling giúp giảm chi phí tính toán, số lượng tham số nhưng giữ lại nhiều thông tin dẫn đến tăng hiệu suất và giảm độ trễ.
- Đầu phân loại nhẹ**:** đầu phân loại và hồi quy của YOLO thường có chung kiến trúc nhưng các phân tích cho thấy đầu hồi quy quan trọng hơn đối với hiệu suất. YOLOv10 đã áp dụng kiến trúc nhẹ cho đầu phân loại giúp giảm chi phí tính toán nhưng không ảnh hưởng đến hiệu suất.
- Thiết kế khối được hướng dẫn theo thứ hạng: YOLOv10 giới thiệu cấu trúc khối đảo ngược(CIB) nhỏ gọn sử dụng phép tích chập theo chiều sâu để pha trộn không gian và phép tích chập theo điểm để tiết kiệm chi phí pha trộn kênh. Cấu trúc này được nhúng trong ELAN, tối hóa kiến trúc giúp hiệu quả cao

hơn. Việc thiết kế này đảm bảo các giai đoạn dư thừa được thay thế bằng các thiết kế nhỏ gọn mà không ảnh hưởng đến hiệu suất.

CHƯƠNG 3. THỰC NGHIỆM TRÊN BỘ DỮ LIỆU FRACATLAS.

3.1. Cơ sở dữ liệu.

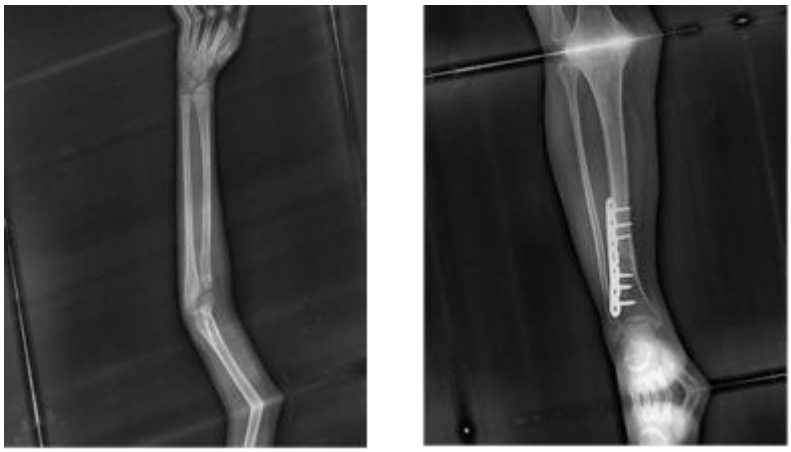
FracAtlas tập hợp các ảnh chụp X-quang được thu thập từ ba bệnh viện lớn ở Bangladesh. Tập dữ liệu bao gồm 4.083 hình ảnh đã được chú thích thủ công để thực hiện các tác vụ như phân loại, định vị và phân đoạn gãy xương với sự trợ giúp của 2 bác sĩ X-quang chuyên nghiệp và một bác sĩ chỉnh hình. Quá trình tạo ra bộ dữ liệu FracAtlas gồm nhiều giai đoạn, bắt đầu từ việc thu thập 14.068 hình ảnh X-quang, chuyển đổi từ định dạng DICOM sang JPG để đảm bảo quyền riêng tư. Tiếp theo được lọc để loại bỏ những bản quét không liên quan đến gãy xương ngoại trừ các khu vực tay, chân, hông và vai. Quá trình này kết thúc với một bộ dữ liệu cuối cùng gồm 4.083 hình ảnh. Trong đó có 717 hình ảnh gãy xương và 3366 hình ảnh xương bình thường.



Hình 16: Minh họa một số hình ảnh trong dữ liệu.

Ngoài ra, quan sát các hình ảnh trong tập Fractured (gãy xương) sẽ bao gồm cả ảnh trước điều trị: thấy được cấu trúc phần xương gãy. Ảnh sau khi điều trị: không còn cấu trúc phần xương gãy và thay vào đó là hình ảnh phản quan của các thiết bị cố định xương (đinh nội tủy, nẹp kim loại, đinh vít). Chi tiết này có thể xem là điểm

yếu cũng như thách thức của tập dữ liệu đối với việc phân loại vì có thể gây nhầm lẫn trong quá trình trích xuất đặc trưng và đào tạo mô hình.



Hình 17: Minh họa hình chụp X Quang trước và sau khi điều trị.

Bộ dữ liệu được cung cấp dưới giấy phép CC-BY 4.0, cho phép sử dụng rộng rãi trong cộng đồng nghiên cứu.

3.2. Tăng cường dữ liệu.

Để cải thiện hiệu quả của mô hình học sâu trong phân loại, chúng tôi áp dụng kỹ thuật tăng cường dữ liệu bằng cách xoay hình ảnh sang phải hoặc trái tối đa 10 độ. Việc này rất quan trọng vì kết quả của các nghiên cứu phân loại phụ thuộc mạnh mẽ vào chất lượng và số lượng dữ liệu huấn luyện. Bằng cách tăng số lượng dữ liệu qua data augmentation, chúng tôi có thể nâng cao hiệu suất huấn luyện mạng một cách đáng kể.

3.3. Tiền xử lý dữ liệu.

Đưa những hình ảnh ban đầu có độ phân giải và ba kênh khác nhau về cùng kích thước 320 × 320 × 3 pixel vì đây là độ phân giải tương thích nhất với các nghiên cứu trên tập dữ liệu tương tự.

Áp dụng các bước tiền xử lý dữ liệu để loại bỏ nhiễu, tránh tác động xấu đến quá trình phân loại của mô hình, giúp mô hình có được kết quả dự đoán chính xác hơn:

- Chuyển ảnh có định dạng *.jpg sang định dạng *.png
- Chuyển đổi CLAHE (chuyển đổi cân bằng biểu đồ thích ứng giới hạn độ tương phản): hình ảnh đầu vào được chia thành nhiều phần, mỗi phần chứa một biểu đồ riêng biệt được điều chỉnh dựa trên giới hạn cắt xén biểu đồ bởi người dùng. Kết quả đầu ra là một hình ảnh được cân bằng độ tương phản.
- Normalization và Standardization: Cuối cùng, chuẩn hóa các giá trị của hình ảnh về cùng một khoảng phạm vi thống nhất và giá trị trung bình của chúng là 0 và độ lệch chuẩn là 1.

3.4. Huấn luyện.

3.4.1. Huấn luyện các mô hình phân loại.

Chúng tôi sử dụng Google Colab với sự hỗ trợ của GPU T4 để đào tạo và thử nghiệm các mô hình học sâu mạng tích chập như ResNet, VGG, MobileNet, v.v., vốn đã được huấn luyện trước trên tập dữ liệu nổi tiếng ImageNet. Bằng cách áp dụng phương pháp học chuyển giao, chúng tôi sử dụng các trọng số của các mô hình đã được huấn luyện trước trên ImageNet và áp dụng chúng vào tập hình ảnh X-quang FracAtlas. Cấu trúc của lớp phân loại cuối cùng, ban đầu gồm 1000 lớp, sẽ được thay đổi thành 2 lớp để phù hợp với bài toán hiện tại. Bộ dữ liệu FracAtlas có chứa hình ảnh bao gồm cả gãy xương và không gãy xương, trong tác vụ phân loại hình ảnh, có 717 hình ảnh gãy xương và 3366 hình ảnh bình thường, sau đó được làm giàu dữ liệu . Các ảnh này được chia làm hai tập cho train và valid với tỉ lệ lần lượt là 70 : 30. Trong các thí nghiệm đào tạo mô hình, các siêu tham số được thiết lập phù hợp với từng mô hình.,số epochs huấn luyện là 20.

3.4.2. Huấn luyện các mô hình phân vùng và phát hiện đối tượng.

Chúng tôi sử dụng Google Colab với sự hỗ trợ của GPU T4 để đào tạo các mô hình và Kaggle để huấn luyện các biến thể có kích thước lớn hơn.

Bộ dữ liệu FracAtlas có chứa hình ảnh bao gồm cả gãy xương và không gãy xương, trong tác vụ phát hiện đối tượng và phân đoạn hình ảnh, chúng tôi chỉ lọc ra các ảnh gãy xương để thực hiện bao gồm 717 hình ảnh. Các ảnh này được chia làm ba tập cho train, test và valid với tỉ lệ lần lượt là 80%, 8% và 12%. Trong các thí nghiệm đào tạo mô hình YOLO, các siêu tham số được thiết lập theo mặc định theo tiêu chuẩn từ Ultralytics với kích thước ảnh đầu vào là 640, số epochs huấn luyện là 50 vì bộ dữ liệu có kích thước khá nhỏ. Chúng tôi cũng sử dụng tất cả các biến thể của từng phiên bản YOLO để thực hiện huấn luyện.

3.5. Độ đo đánh giá.

Để đánh giá hiệu quả của các phương pháp, chúng tôi tiến hành thử nghiệm với các mô hình học sâu đã được giới thiệu trước đó, sử dụng ba thước đo chính: Accuracy, Precision, Recall và F1-score giữa tập dự đoán và tập dữ liệu được gán nhãn.

Precision và Recall:

- Đối với các bộ dữ liệu không cân bằng, tức là số lượng dữ liệu của các lớp chênh lệch nhiều, Precision và Recall trở thành những thước đo quan trọng.
- Precision (Positive Predictive Value - PPV): Tỉ lệ dương tính đoán đúng.

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

- Precision được định nghĩa là tỉ lệ số điểm TP trong số các điểm được phân loại là dương tính (TP + FP).
- Precision cao: Độ chính xác của các điểm được dự đoán là dương tính cao.
- Recall (True Positive Rate - TPR): Độ nhạy, tỷ lệ dương tính thực.

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

- Recall được định nghĩa là tỉ lệ số điểm TP trong số các điểm thực sự là dương tính (TP + FN).
- Recall cao: Độ nhạy cao, tức là tỷ lệ bỏ sót các điểm thực sự dương tính thấp.

- Precision cao đồng nghĩa với việc báo động nhầm ít (tỉ lệ FP thấp), trong khi Recall cao đồng nghĩa với việc tỷ lệ bỏ sót thấp (tỉ lệ FN thấp).

F1-score:

- Chỉ số F1 là giá trị trung bình điều hòa của Precision và Recall.

$$F_1 = \frac{\text{Precision}.\,\text{Recall}}{\text{Precision} + \text{Recall}}$$

- F1-score có giá trị nằm trong khoảng (0, 1]. F1-score càng cao đồng nghĩa với bộ phân loại càng tốt. Khi cả Recall và Precision đều bằng 1 thì F1-score = 1. Khi cả Recall và Precision đều thấp thì F1-score tiến về 0.

mAP50:

Để đánh giá các tác vụ như phát hiện đối tượng hay phân đoạn hình ảnh, mAP (mean average precision) là một metric phổ biến cho việc này giúp đánh giá được hiệu suất của mô hình. Để tính được mAP ta cần có các kết quả của IoU (Intersect of Union) – một chỉ số thường được dùng để xác định độ chính xác việc dự đoán vị trí của đối tượng. IoU là tỉ số phép giao của kết quả dự đoán và hộp giới hạn thực tế so với sự hợp nhất của hai kết quả này. Ta có thể tính IoU như sau:

$$IoU = \frac{A \cap B}{A \cup B}$$

Trong đó A đại diện cho tập các kết quả dự đoán của hộp giới hạn và B là tập hộp giới hạn thực tế của đối tượng.

Nếu như IoU > 0.5, lúc này việc phát hiện được xem là True Positive (TP) ngược lại IoU < 0.5 là False Positve (FP). Các kết quả TP và FP được tính bằng IoU sẽ được dùng để tính Average Precision (AP) cho mỗi lớp đối tượng c như sau:

$$AP = \frac{TP(c)}{TP(c) + FP(c)}$$

Sau khi có kết quả của AP, ta sẽ tính mAP là trung bình của AP trên tất cả lớp, công thức tính mAP được thể hiện phía dưới:

$$mAP = \frac{1}{C}\sum_{c}^{C} AP(c)$$

Trong bài làm, chúng tôi sử dụng mAP50 (dựa trên các chỉ số IoU > 0.5) để làm metric đánh giá cho mô hình. Một mô hình phát hiện và phân đoạn tốt khi kết quả mAP50 có giá trị cao để ít gây ra sai xót trong quá trình thực nghiệm.

3.6. Kết quả thử nghiệm trên bộ dữ liệu FracAtlas.

3.6.1. Kết quả thử nghiệm trên các mô hình phân loại.

Đánh giá kết quả phân loại của mô hình một cách chính xác và đầy đủ thông qua các thông số: accuracy, precision, recall, F1-score. Dưới đây là kết quả phân loại theo lớp của các mô hình học sâu đã được sử dụng:

Models	Accuracy	Precision	Recall	F1Score	Labels
Alexnet	0.90	0.74	0.64	0.69	Fractured
		0.93	0.95	0.94	Nonfractured
VGG16	0.92	0.81	0.74	0.78	Fractured
		0.95	0.96	0.95	Nonfractured
VGG19	0.91	0.78	0.66	0.71	Fractured
		0.93	0.96	0.94	Nonfractured
ResNet-18	0.90	0.76	0.65	0.70	Fractured
		0.93	0.96	0.94	Nonfractured
ResNet-34	0.90	0.76	0.65	0.70	Fractured
		0.93	0.96	0.94	Nonfractured
ResNet-50	0.91	0.82	0.65	0.72	Fractured
		0.93	0.97	0.95	Nonfractured
ResNet-101	0.91	0.85	0.61	0.71	Fractured
		0.92	0.98	0.95	Nonfractured
ResNet-152	0.90	0.76	0.65	0.70	Fractured
		0.93	0.96	0.94	Nonfractured

Bảng 1: Kết quả các mô hình Alexnet, VGG và ResNet

Models	Accuracy	Precision	Recall	F1Score	Labels
DenseNet169	0.93	0.87	0.72	0.79	Fractured
		0.94	0.98	0.96	Nonfractured
DenseNet201	0.92	0.86	0.68	0.76	Fractured
		0.93	0.98	0.96	Nonfractured
MobileNetV2	0.93	0.82	0.75	0.79	Fractured
		0.95	0.97	0.96	Nonfractured
MobileNetV3_Small	0.91	0.84	0.62	0.71	Fractured
		0.92	0.97	0.95	Nonfractured

Bảng 2: Kết quả các mô hình DenseNet và MobileNet

Models	Accuracy	Precision	Recall	F1Score	Labels
EfficientNet_b7	0.93	0.86	0.72	0.78	Fractured
		0.94	0.97	0.96	Nonfractured
EfficientNet_b1	0.93	0.82	0.76	0.79	Fractured
		0.95	0.96	0.96	Nonfractured
EfficientNet_v2_s	0.90	0.76	0.67	0.71	Fractured
		0.93	0.96	0.94	Nonfractured
EfficientNet_b0	0.92	0.81	0.75	0.78	Fractured
		0.95	0.96	0.95	Nonfractured
RegNet_y_16gf	0.90	0.88	0.53	0.66	Fractured
		0.91	0.98	0.94	Nonfractured
MaxVit_T	0.91	0.77	0.72	0.75	Fractured
		0.94	0.95	0.95	Nonfractured
RepVGG_A0	0.90	0.76	0.65	0.70	Fractured
		0.93	0.96	0.94	Nonfractured

Bảng 3: Kết quả của các mô hình phân loại còn lại

Nhận xét:

- DenseNet169, EfficientNet_b1 và MobileNetV2 có F1-Score cao nhất cho phân loại vết nứt (Fractured) với 0.79, và cũng đạt hiệu suất cao cho phân loại xương không bị nứt (Nonfractured) với F1-Score là 0.96.

- VGG16 cũng có hiệu suất tốt với F1-Score cho Fractured là 0.78 và Nonfractured là 0.95.

3.6.2. Kết quả thử nghiệm các mô hình phân vùng và phát hiện đối tượng.

Sau khi huấn luyện các biến thể của các phiên bản YOLO, chúng tôi tổng hợp kết quả bao gồm precision, recall và mAP50.

Kết quả thử nghiệm các mô hình phân vùng ảnh được thể hiện ở bảng sau:

Models	Precision (Box)	Recall (Box)	mAP50 (Box)	Precision (Mask)	Recall (Mask)	mAP50 (Mask)
YOLOv8n-seg	0.64	0.493	0.561	0.711	0.505	0.536
YOLOv8s-seg	0.645	0.534	0.562	0.645	0.534	0.569
YOLOv8m-seg	0.529	0.57	0.548	0.589	0.479	0.521
YOLOv8l-seg	0.585	0.521	0.539	0.628	0.462	0.51
YOLOv8x-seg	0.689	0.411	0.462	0.732	0.438	0.502
YOLOv9c-seg	0.581	0.384	0.395	0.618	0.411	0.462
YOLOv9e-seg	0.688	0.288	0.398	0.604	0.301	0.363

Bảng 4: Kết quả của các mô hình YOLO cho tác vụ phân đoạn

Nhận xét:

- Mô hình YOLOv8s-seg có hiệu suất tổng thể tốt nhất với sự cân bằng giữa các chỉ số Precision, Recall, và mAP50 cho cả hộp và mặt nạ. Điều này cho thấy mô hình này có thể là lựa chọn tốt nhất cho việc phân đoạn ảnh X-quang trong tập dữ liệu này.
- YOLOv8x-seg nổi bật với Precision cao nhất cho cả hộp và mặt nạ, tuy nhiên, Recall của nó khá thấp, cho thấy nó có thể bỏ lỡ nhiều đối tượng.

- YOLOv8m-seg có Recall cao nhất cho hộp, nhưng Precision và mAP50 thấp hơn so với các mô hình khác, cho thấy nó có thể phát hiện nhiều đối tượng nhưng không chính xác.
- Các phiên bản YOLOv9 (c-seg và e-seg) cho thấy hiệu suất kém hơn so với các phiên bản YOLOv8, đặc biệt là về Recall và mAP50.

Trong các mô hình trên, mặc dù mỗi mô hình có những ưu và nhược điểm riêng, YOLOv8s-seg nổi bật nhất với hiệu suất cân bằng và ổn định, làm cho nó trở thành mô hình làm tốt nhiệm vụ phân vùng ảnh trông bộ dữ liệu này.

Kết quả thử nghiệm các mô hình phát hiện đối tượng được thể hiện ở bảng sau:

Models	Precision	Recall	mAP50
YOLOv8n	0.716	0.562	0.601
YOLOv8s	0.733	0.384	0.497
YOLOv8m	0.796	0.466	0.551
YOLOv8l	0.655	0.479	0.489
YOLOv8x	0.549	0.534	0.501
YOLOv9t	0.64	0.479	0.493
YOLOv9s	0.666	0.479	0.528
YOLOv9m	0.753	0.493	0.558
YOLOv9c	0.602	0.493	0.485
YOLOv9e	0.6	0.452	0.45
YOLOv10n	0.514	0.477	0.471

YOLOv10s	0.581	0.411	0.483
YOLOv10m	0.61	0.356	0.426
YOLOv10b	0.582	0.397	0.445
YOLOv10l	0.572	0.356	0.388

Bảng 5: Kết quả của các mô hình YOLO cho tác vụ phát hiện

Nhận xét:

- Mô hình YOLOv8m đạt giá trị Precision cao nhất với 0.796, cho thấy khả năng xác định chính xác đối tượng rất tốt.
- YOLOv10n có Precision thấp nhất với 0.514, cho thấy khả năng xác định đối tượng kém hơn.
- YOLOv8n đạt giá trị Recall cao nhất với 0.562, chứng tỏ khả năng phát hiện được nhiều đối tượng trong ảnh.
- YOLOv10l có Recall thấp nhất với 0.356, cho thấy khả năng phát hiện đối tượng hạn chế.
- YOLOv8n đạt mAP50 cao nhất với 0.601, cho thấy hiệu suất tổng thể tốt khi so sánh giữa các mô hình.
- YOLOv10l có mAP50 thấp nhất với 0.388, cho thấy hiệu suất kém hơn.
- Mô hình YOLOv9s và YOLOv9m có hiệu suất tổng thể tốt nhất, với sự cân bằng giữa các chỉ số Precision, Recall, và mAP50. Điều này cho thấy hai mô hình này là lựa chọn tốt nhất cho việc phát hiện đối tượng trong ảnh X-quang trong tập dữ liệu này.

CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.

4.1. Kết luận

Trong bài luận này, chúng tôi đã đánh giá hiệu suất của các mô hình YOLO mới nhất bao gồm YOLOv8, YOLOv9 và YOLOv10 cho tác vụ phát hiện đối tượng, phân đoạn hình ảnh. Thêm vào đó chúng tôi cũng đánh giá các mô hình mạng nơ-ron tích chập cho tác vụ phân loại nhị phân. Tất cả được thực hiện trên bộ dữ liệu FracAtlas. Từ kết quả đã thực hiện cho thấy:

- Các mô hình phân loại DenseNet169, EfficientNet_b1, MobileNetV2 và VGG16 cho các kết quả tốt nhất so với các mô hình còn lại.
- So với các biến thể giữa các mô hình YOLO, YOLOV8n là biến thể cho kết quả tốt nhất đối với tác vụ phát hiện đối tượng và YOLOv8s-seg cho kết quả tốt nhất đối với tác vụ phân đoạn hình ảnh.
- Trên cùng một tập dữ liệu, sự tăng dần về độ phức tạp, số lượng tham số và kích thước của các biến thể YOLO không cho kết quả tăng một cách tuyến tính, điều đó thể hiện qua các chỉ số độ đo tăng giảm không đồng đều.
- Nhìn chung, các mô hình cung cấp được kết quả tham khảo hỗ trợ bác sĩ chuẩn đoán được việc gãy xương của bệnh nhân thông qua ảnh chụp X-quang. Với mô hình phân loại hỗ trợ xác định liệu kết quả ảnh chụp X-quang có xương gãy hay không, mô hình phát hiện định vị được vị trí xương gãy thông qua hộp giới hạn và mô hình phân đoạn giúp làm nổi bật khoanh vùng vị trí cụ thể của xương gãy.
- Tuy nhiên các kết quả về phát hiện đối tượng và phân đoạn hình ảnh nhìn chung vẫn còn hạn chế khi mô hình không thể dự đoán trên một số ảnh, dự đoán sai hoặc thiếu. Điều này có thể gây một số trở ngại khi tham khảo các kết quả này cho việc chuẩn đoán gãy xương.

4.2. Hướng phát triển

Trong tương lai, việc phát triển các nghiên cứu cho các tác vụ phân loại, phát hiện và phân đoạn trên bộ dữ liệu FracAtlas có thể thực hiện tiếp bằng một số phương pháp:

- Cải thiện chất lượng hình ảnh cho tập train.
- Ứng dụng thêm một số mô hình khác để tăng thêm độ chính xác.
- Sử dụng một số mô hình được huấn luyện trước từ các bộ dữ liệu về ảnh chụp X-quang về xương để áp dụng lên bộ dữ liệu hiện tại.
- Áp dụng phương pháp Mixture of Expect kết hợp giữa các mô hình cho bài toán phân loại.

TÀI LIỆU THAM KHẢO

[1] Abedeen, I., Rahman, M.A., Prottyasha, F.Z. *et al.* (2023). FracAtlas: A Dataset for Fracture Classification, Localization and Segmentation of Musculoskeletal Radiographs. *Sci Data* 10, 521.

[2] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton (2012) - "Imagenet Classification with Deep Convolutional Neural Networks". Advances in Neural Information Processing Systems (NIPS).

[3] Simonyan, K., & Zisserman, A. (2014). "Very Deep Convolutional Networks for Large-Scale Image Recognition". *arXiv preprint arXiv:1409.1556*.

[4] He, K., Zhang, X., Ren, S., & Sun, J. (2016). "Deep Residual Learning for Image Recognition". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.

[5] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). "Densely Connected Convolutional Networks". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4700-4708.

[6] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). "MobileNetV2: Inverted Residuals and Linear Bottlenecks". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510-4520.

[7] Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. (2019). "Searching for MobileNetV3". *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1314-1324.

[8] Tan, M., & Le, Q. V. (2019). "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 6105-6114.

[9] Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., & Dollár, P. (2020). "Designing Network Design Spaces". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 10428-10436.

[10] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". *arXiv preprint arXiv:2010.11929*.

[11] Ding, X., Zhang, X., Han, J., & Ding, G. (2021). "RepVGG: Making VGG-style ConvNets Great Again". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 13728-13737.

[12] Yousaf, K., Nawaz, T. & Habib, A. (2024). "Using two-stream EfficientNet-BiLSTM network for multiclass classification of disturbing YouTube videos". Multimed Tools Appl 83, 36519–36546

[13] Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., & Li, Y. (2022). MaxViT: Multi-Axis Vision Transformer. arXiv preprint arXiv:2204.01697

[14] Sohan, M., Sai Ram, T., Rami Reddy, C.V. (2024). "A Review on YOLOv8 and Its Advancements". In: Jacob, I.J., Piramuthu, S., Falkowski-Gilski, P. (eds) Data Intelligence and Cognitive Informatics. ICDICI 2023. Algorithms for Intelligent Systems. Springer, Singapore

[15] Jocher, G., Chaurasia, A., Qiu, J. (2023)**.** "YOLOv8: Real-Time Flying Object Detection with YOLOv8". *arXiv preprint arXiv:2305.09972*.

[16] Wang, C.-Y., Yeh, I.-H., & Liao, H.-Y. M. (2023)**.** "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information". *arXiv preprint arXiv:* 2402.13616.

[17] Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., & Ding, G. YOLOv10: Real-Time End-to-End Object Detection. Tsinghua University. *arXiv preprint arXiv:* 2405.14458.

[18] https://vinbigdata.com/kham-pha/04-mo-hinh-pre-trained-cnn-giup-ban-giai-quyet-cac-bai-toan-thi-giac-may-tinh-voi-transfer-learning.html

[19] https://viblo.asia/p/deep-learning-tim-hieu-ve-mang-tich-chap-cnn-maGK73bOKj2

[20] https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/

[21] https://d2l.ai/chapter_convolutional-modern/resnet.html

[22] https://medium.com/visionwizard/simple-powerful-and-fast-regnet-architecture-from-facebook-ai-research-6bbc8818fb44

[23] https://article.stunningvisionai.com/yolov9-architecture