TRƯỜNG ĐẠI HỌC NGOẠI NGỮ-TIN HỌC TP. HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN

KHÓA LUẬN TỐT NGHIỆP

ĐỀ TÀI: XÁC ĐỊNH VĂN BẢN TRONG ẢNH SỬ DỤNG PHƯƠNG PHÁP MÁY HỌC

GVHD: PGS.TS. Nguyễn Thanh Bình
SVTH: 21DH111919 - Ngô Trọng Tín
21DH113466 - Lương Gia Ân

TP. HỒ CHÍ MINH – THÁNG 8 – NĂM 2024

LỜI CẢM ƠN

Lời đầu tiên, nhóm em xin được gửi lời cảm ơn đến thầy PGS.TS Nguyễn Thanh Bình – giảng viên hướng dẫn trực tiếp. Trong quá trình tìm hiểu và thực hiện, thầy đã giúp đỡ, hướng dẫn cho chúng em để có thể hoàn thành khóa luận tốt nghiệp một cách thuận lợi.

Cùng đó, nhóm em cũng xin được trân trọng cảm ơn thầy cô cùng ban giám hiệu khoa Công nghệ thông tin – trường Đại học Ngoại ngữ - Tin học TPHCM (HUFLIT) đã giúp cho chúng em có thể hoàn thành khóa luận tốt nghiệp trong điều kiện và môi trường học tập tốt.

Nhóm em trong quá trình tìm hiểu – thực hiện – hoàn thành khóa luận khó tránh khỏi những thiếu sót do hạn chế về mặt kiến thức và thiếu kinh nghiệm thực tiễn. Chúng em kính mong nhận được những lời góp ý, đánh giá từ thầy cô để khóa luận có thể hoàn thiện hơn cũng như là góp thêm kinh nghiệm cho chúng em trong hành trình sắp tới.

Chúng em xin chân thành cảm ơn!

LỜI CAM KẾT

Chúng em xin cam đoan rằng đề tài "Xác định văn bản trong ảnh sử dụng phương pháp máy học" thuộc về quyền sở hữu của nhóm, bao gồm 2 sinh viên: 21DH111919 – Ngô Trọng Tín và 21DH113466 – Lương Gia Ân, dưới sự hướng dẫn của thầy PGS.TS. Nguyễn Thanh Bình.

Mọi tài liệu tham khảo đã được trích dẫn trong mục Tài liệu tham khảo. Mọi kết quả, số liệu nghiên cứu đều được nhóm hiện thực và ghi nhận lại, không sao chép từ nguồn khác.

Chúng em xin chịu mọi hình phạt và kỉ luật nếu như lời cam đoan không chính xác.

MỤC LỤC

DANH MỤC HÌNH ẢNH

Chương 1: GIỚI THIỆU

1.1. Giới thiệu đề tài

Xác định văn bản trong ảnh là một lĩnh vực của thị giác máy tính với mục tiêu bao gồm: xác định văn bản trong ảnh và trích xuất nội dung văn bản trong ảnh thành một tập văn bản.

Trong lĩnh vực thị giác máy tính, ứng dụng về nhận diện văn bản trong ảnh rất đa dạng: số hóa tài liệu, bảng báo cáo; ứng dụng dịch tự động; phân loại các hình ảnh; tìm kiếm hình ảnh dựa trên nội dung văn bản;…

Luận văn tập trung nghiên cứu và phát triển mô hình xác định văn bản trong ảnh, sử dụng phương pháp máy học. Các mô hình và tập dữ liệu trong đề tài trên được huấn luyện để đánh giá độ chính xác dựa trên thang đo đánh giá, trích xuất văn bản từ ảnh thành tập văn bản (có đuôi là .txt).

1.2. Mục tiêu và nội dung đề tài

- **Mục tiêu** luận văn nghiên cứu và phát triển mô hình học sâu để nhận diện và trích xuất văn bản từ các hình ảnh dựa trên tập dữ liệu đã chọn ra. Áp dụng các cải tiến mô hình của lĩnh vực học máy để chọn ra những mô hình phù hợp, có khả năng hoàn thiện và đưa ra kết quả tốt so với các mô hình còn lại. Mô hình có khả năng ứng dụng vào các bài toán thực tế như là trích xuất văn bản, xác định văn bản trong ảnh.
- **Nội dung đề tài** thực hiện các việc sau:
  i. Tìm hiểu thông tin cách thức hoạt động của đề tài.
  ii. Nghiên cứu các mô hình phù hợp đề tài.
  iii. Thu thập các tập dữ liệu văn bản.
  iv. Đề xuất những mô hình nhận diện văn bản.
  v. Thực nghiệm các mô hình đề xuất và so sánh.

1.3. Giới hạn đề tài

Trong đề tài này, nhóm chọn ra 3 mô hình CNN, DenseNet121 và VGG19 vì nhận thấy khả năng nhận diện và xác định văn bản của 3 mô hình trên thích hợp với tập dữ liệu đã được chọn.

1.4. Cấu trúc báo cáo

Bài báo cáo bao gồm 5 mục chính:

- **Chương 1: GIỚI THIỆU ĐỀ TÀI**

  Giới thiệu về đề tài luận văn, mục tiêu và nội dung thực hiện đề tài, giới hạn nghiên cứu luận văn đặt cho đề tài.

- **Chương 2: CƠ SỞ LÝ THUYẾT LIÊN QUAN**

  Đưa ra các cơ sở lý thuyết đã tìm hiểu về các thông tin mô hình và tập dữ liệu đã sử dụng để xác định văn bản trong ảnh bằng phương pháp máy học.

- **Chương 3: PHƯƠNG PHÁP ĐỀ XUẤT**

  Mô tả bài toán của luận văn. Luận văn chọn mô hình phương pháp học sâu và thực nghiệm các phương pháp mà nhóm đề xuất thực nghiệm. Trong mục còn đề cập đến thang đo đánh giá các mô hình.

- **Chương 4: THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ**

  Đưa ra được các yêu cầu phần cứng và phần mềm khi thực hiện phương pháp. Áp dụng mô hình và so sánh kết quả thực hiện.

- **Chương 5: KẾT LUẬN**

  Trình bày về những gì luận văn đã đạt được trong quá trình thực hiện, bao gồm các lỗi đã mắc phải và ưu – nhược điểm của phương pháp và định hướng phát triển trong tương lai. Rút ra được kinh nghiệm từ những lỗi sai trong quá trình làm việc.

Chương 2: CƠ SỞ LÝ THUYẾT LIÊN QUAN

2.1. Cơ sở lý thuyết

Xác định văn bản trong ảnh (Optical Character Recognition - OCR) là một lĩnh vực nghiên cứu quan trọng trong xử lý ảnh và thị giác máy tính. Việc nhận dạng và trích xuất thông tin từ văn bản trong ảnh mở ra nhiều ứng dụng trong đời sống như số hóa tài liệu, nhận dạng biển số xe, và hỗ trợ người khuyết tật thị giác.

Có 3 phương pháp xác định văn bản trong ảnh:

- **Phương pháp truyền thống:** Sử dụng các thuật toán như k-means clustering, connected component analysis, và các kỹ thuật phân loại dựa trên đặc trưng thủ công.
- **Phương pháp dựa trên học máy (Machine Learning-based Methods)**: Sử dụng các mô hình học máy cổ điển như Support Vector Machines (SVMs), Random Forests, hoặc các thuật toán phân loại khác để nhận diện ký tự.
- **Phương pháp học sâu (Deep Learning-based Methods)**: Sử dụng các mô hình học sâu, đặc biệt là mạng nơ-ron tích chập (CNNs), mạng nơ-ron lặp (RNNs), và các mô hình Transformer. Các mô hình như VGG, ResNet, DenseNet, LSTM, và Transformer đã được áp dụng rộng rãi trong OCR và cho kết quả vượt trội so với các phương pháp truyền thống và học máy cổ điển.

Trong đó, phương pháp học sâu (Deep Learning-based Methods) được đánh giá cao trong việc nhận dạng, xác định văn bản có trong ảnh.

2.2. Giới thiệu về nhận diện văn bản (Text Recognition)

2.2.1 Lịch sử phát triển của OCR

Nhận diện ký tự quang học (Optical Character Recognition - OCR) là một công nghệ giúp nhận diện và chuyển đổi các ký tự văn bản từ hình ảnh thành dữ liệu kỹ thuật số có thể chỉnh sửa và tìm kiếm được. Quá trình phát triển của OCR đã trải qua nhiều giai đoạn với những cải tiến vượt bậc cả về mặt kỹ thuật lẫn ứng dụng.

2.2.1.1 Giai đoạn đầu tiên: Từ ý tưởng đến thực tế

Công nghệ OCR bắt đầu xuất hiện từ những năm đầu thế kỷ 20 với những ý tưởng đầu tiên về việc nhận diện ký tự một cách tự động. Những nỗ lực ban đầu chủ yếu tập trung vào việc xây dựng các máy móc cơ học có thể đọc và nhận diện các ký tự in ấn.

- **Năm 1914:** Emanuel Goldberg, một nhà phát minh người Đức, đã phát triển một thiết bị có khả năng chuyển đổi các ký tự in thành mã Morse, được xem là một trong những nguyên mẫu đầu tiên của OCR.
- **Năm 1931:** Goldberg tiếp tục phát triển một thiết bị khác có khả năng quét và chuyển đổi các ký tự in trên phim ảnh thành tín hiệu điện tử, đây là một bước tiến quan trọng trong việc nhận diện ký tự tự động.

2.2.1.2 Sự phát triển trong thế kỷ 20: Từ cơ học đến điện tử

Trong những năm 1950, với sự phát triển của công nghệ điện tử, các hệ thống OCR bắt đầu chuyển từ cơ học sang điện tử, mang lại hiệu quả cao hơn và khả năng ứng dụng rộng rãi hơn.

- **Năm 1951:** David H. Shepard, một kỹ sư người Mỹ, đã phát minh ra thiết bị nhận diện ký tự đầu tiên có thể chuyển đổi văn bản in sang tín hiệu điện tử. Thiết bị này sau đó được sử dụng trong hệ thống nhận diện ký tự quang học đầu tiên của Hoa Kỳ.
- **Năm 1955:** Công ty Reader's Digest sử dụng hệ thống OCR đầu tiên để xử lý số lượng lớn phiếu đăng ký của độc giả, giúp giảm thiểu đáng kể thời gian và công sức cho việc nhập liệu thủ công.

2.2.1.3 Giai đoạn hiện đại: Sự bùng nổ của công nghệ số

Vào những năm 1980 - 1990, với sự phát triển mạnh mẽ của máy tính và công nghệ số, OCR đã có những bước tiến vượt bậc, từ việc nhận diện ký tự in ấn đơn giản đến khả năng nhận diện văn bản viết tay và đa ngôn ngữ.

- **Năm 1974:** Công ty Kurzweil Computer Products, do Raymond Kurzweil sáng lập, đã phát triển hệ thống OCR đầu tiên có khả năng nhận diện nhiều phông chữ và ký tự khác nhau, mở ra một kỷ nguyên mới cho công nghệ này.

- **Năm 1993:** Adobe giới thiệu phần mềm OCR đầu tiên tích hợp trong Adobe Acrobat, cho phép người dùng chuyển đổi tài liệu PDF thành văn bản có thể tìm kiếm được, đánh dấu một bước ngoặt quan trọng trong ứng dụng OCR.

2.2.1.4 OCR trong thế kỷ 21: Trí tuệ nhân tạo và Học máy

Bước vào thế kỷ 21, sự phát triển của trí tuệ nhân tạo (AI) và học máy (Machine Learning) đã mang lại những cải tiến đáng kể cho OCR, từ khả năng nhận diện chính xác hơn đến việc xử lý văn bản trong các môi trường phức tạp.

- **Năm 2006:** Google giới thiệu Google Books, một dự án sử dụng OCR để số hóa hàng triệu cuốn sách trên toàn thế giới. Dự án này không chỉ giúp bảo tồn tri thức nhân loại mà còn làm tăng đáng kể khả năng truy cập thông tin cho người dùng.
- **Năm 2015:** DeepMind, một công ty thuộc Google, đã phát triển các mô hình học sâu (Deep Learning) cho OCR, giúp cải thiện đáng kể độ chính xác trong việc nhận diện văn bản từ các hình ảnh phức tạp, bao gồm cả văn bản viết tay và văn bản bị biến dạng.

2.2.2 Các ứng dụng thực tiễn của OCR trong đời sống và công nghiệp

OCR đã được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau của đời sống và công nghiệp, mang lại nhiều lợi ích to lớn cho con người và xã hội.

2.2.2.1 Số hóa tài liệu

Một trong những ứng dụng phổ biến nhất của OCR là số hóa tài liệu. Thông qua OCR, các tài liệu giấy có thể được chuyển đổi thành văn bản kỹ thuật số, giúp dễ dàng lưu trữ, tìm kiếm và chia sẻ.

- **Lưu trữ và quản lý tài liệu:** OCR giúp các tổ chức và doanh nghiệp số hóa kho tài liệu khổng lồ của mình, từ đó giảm thiểu không gian lưu trữ vật lý và tăng cường khả năng truy xuất thông tin. Các thư viện, trường đại học và các cơ quan nhà nước đã và đang áp dụng OCR để bảo tồn và quản lý tài liệu của mình.

- **Truyền tải và chia sẻ thông tin:** Trong thời đại số hóa, việc truyền tải và chia sẻ thông tin trở nên dễ dàng hơn bao giờ hết nhờ OCR. Các tài liệu có thể được gửi đi và truy cập từ bất cứ đâu, giúp tăng cường khả năng hợp tác và làm việc từ xa.

2.2.2.2 Hỗ trợ người khuyết tật

OCR đã mở ra nhiều cơ hội mới cho người khuyết tật, đặc biệt là người khiếm thị hoặc suy giảm thị lực. Các thiết bị và ứng dụng dựa trên OCR giúp họ tiếp cận thông tin một cách dễ dàng hơn.

- **Đọc văn bản bằng giọng nói:** Các ứng dụng như KNFB Reader sử dụng OCR để đọc to văn bản trên tài liệu giấy hoặc màn hình, giúp người khiếm thị có thể "đọc" sách, báo và các tài liệu khác mà không cần sự trợ giúp từ người khác.
- **Chuyển đổi văn bản sang chữ nổi:** OCR cũng được sử dụng trong các hệ thống chuyển đổi văn bản thành chữ nổi (Braille), giúp người khiếm thị có thể đọc tài liệu một cách độc lập và chủ động.

2.2.2.3 Tự động hóa quá trình nhập liệu

OCR đã mang lại sự tự động hóa cho nhiều quy trình trong doanh nghiệp và công nghiệp, từ việc nhập liệu đến xử lý văn bản.

- **Xử lý hóa đơn và biên lai:** Nhiều doanh nghiệp sử dụng OCR để tự động hóa quá trình xử lý hóa đơn và biên lai, giúp tiết kiệm thời gian và giảm thiểu lỗi do con người gây ra. Các hệ thống này có thể tự động quét và nhập liệu từ hàng ngàn hóa đơn chỉ trong vài phút.
- **Tự động hóa quy trình kế toán:** Trong lĩnh vực kế toán, OCR giúp tự động hóa việc nhập dữ liệu từ các chứng từ kế toán như phiếu thu, chi, và báo cáo tài chính, giảm thiểu thời gian và công sức cho các nhân viên kế toán.

2.2.2.4 Quản lý giao thông và an ninh

OCR cũng được ứng dụng rộng rãi trong quản lý giao thông và an ninh, từ nhận diện biển số xe đến kiểm soát an ninh tại các sân bay và cơ sở hạ tầng quan trọng.

- **Nhận diện biển số xe:** Các hệ thống nhận diện biển số xe tự động (ANPR) sử dụng OCR để đọc và lưu trữ thông tin biển số xe, giúp cảnh sát và cơ quan chức năng quản lý giao thông và xử lý vi phạm một cách hiệu quả.
- **Kiểm soát an ninh:** Tại các sân bay và cơ sở hạ tầng quan trọng, OCR được sử dụng để quét và nhận diện thông tin từ hộ chiếu và giấy tờ tùy thân, giúp tăng cường an ninh và giảm thiểu thời gian chờ đợi cho hành khách.

2.2.3 Thách thức trong nhận diện văn bản từ hình ảnh

Mặc dù OCR đã đạt được nhiều thành tựu nhưng công nghệ này vẫn đối mặt với nhiều thách thức cần được giải quyết để nâng cao hiệu suất và khả năng ứng dụng,

2.2.3.1. Chất lượng hình ảnh

Một trong những thách thức lớn nhất đối với OCR là chất lượng hình ảnh. Hình ảnh có thể bị mờ, nhiễu hoặc có độ tương phản thấp, gây khó khăn cho quá trình nhận diện.

- **Hình ảnh bị mờ hoặc nhiễu:** Hình ảnh mờ hoặc bị nhiễu có thể khiến OCR không thể nhận diện chính xác ký tự. Điều này thường xảy ra khi tài liệu được chụp bằng các thiết bị di động có chất lượng thấp hoặc trong điều kiện ánh sáng không tốt.
- **Văn bản bị biến dạng:** Văn bản trên các tài liệu cũ bị gập hoặc uốn cong cũng gây khó khăn cho OCR, đặc biệt là khi các ký tự bị biến dạng hoặc mất nét.

2.2.3.2. Đa dạng ngôn ngữ và phông chữ

Sự đa dạng ngôn ngữ và phông chũ cũng là một thách thức đáng kể đối với OCR, đặc biệt là trong các ứng dụng đa quốc gia hoặc liên văn hóa.

- **Ngôn ngữ và ký tự đặc biệt:** OCR cần được huấn luyện để nhận diện các ngôn ngữ và ký tự khác nhau. Từ các ngôn ngữ Latinh như tiếng Anh hoặc tiếng Pháp, đến các ngôn ngữ tượng hình như tiếng Trung Quốc hoặc tiếng Nhật. Điều này đòi hỏi một lượng lớn dữ liệu huấn luyện đa dạng và phong phú.

- **Phông chữ và kích thước khác nhau:** Các phông chữ không phổ biến hoặc kích thước chữ quá nhỏ hoặc quá lớn cũng có thể gây khó khăn cho OCR dẫn đến việc nhận diện sai hoặc không chính xác.

2.2.3.3. Văn bản viết tay

Văn bản viết tay là một trong những thách thức lớn nhất đối với OCR. Mỗi người có một phong cách viết tay riêng biệt và thậm chí văn bản của cùng một người cũng có thể tay đổi theo thời gian hoặc tình huống.

- **Sự biến đổi trong văn bản viết tay:** Sự khác biệt trong độ dày nét bút, góc nghiêng của chữ viết và khoảng cách giữa các ký tự có thể khiến OCR gặp khó khăn trong việc nhận diện chính xác.
- **Thiếu dữ liệu huấn luyện:** Việc huấn luyện OCR để nhận diện văn bản viết tay đòi hỏi một lượng lớn dữ liệu huấn luyện. Điều này có thể khó khăn trong việc thu thập và xử lý.

2.2.3.4. Kết luận

Phần giới thiệu này đã cung cấp một cái nhìn tổng quan về lịch sử phát triển của OCR, các ứng dụng thực tiễn của nó trong đời sống và công nghiệp cũng như các thách thức mà công nghệ này phải đối mặt. Mặc dù OCR đã có những bước tiến lớn trong những thập kỷ qua nhưng vẫn còn nhiều vấn đề cần được giải quyết để công nghệ này có thể phát huy tối đa tiềm năng của mình trong tương lai. Những thách thức này cũng chính là cơ hội để các nhà nghiên cứu và các công ty công nghệ tiếp tục cải tiến và phát triển các giải pháp OCR ngày càng mạnh mẽ và chính xác hơn.

2.3. Phân loại và tổng quan về OCR.

2.3.1. OCR truyền thống và OCR dựa trên học sâu.

Nhận diện ký tự quang học (OCR) đã trải qua nhiều giai đoạn phát triển, từ các phương pháp truyền thống đến việc áp dụng các mô hình học sâu tiên tiến. Cả hai phương pháp này đều có những ưu điểm và hạn chế riêng, phản ánh sự tiến bộ của công nghệ trong việc nhận diện và xử lý văn bản từ hình ảnh.

2.3.1.1. OCR truyền thống

OCR truyền thống dựa trên các thuật toán và kỹ thuật xử lý hình ảnh cơ bản để nhận diện các ký tự trong văn bản. Các phương pháp này thường bao gồm nhiều bước tiền xử lý và phân tích hình ảnh để trích xuất các đặc trưng từ ký tự, sau đó so sánh chúng với các mẫu ký tự đã được định nghĩa trước.

- **Phương pháp dựa trên mẫu (Template Matching):** Đây là một trong những phương pháo đơn giản và phổ biến nhất trong OCR truyền thống. Hệ thống sẽ so sánh các ký tự trong hình ảnh với các mẫu ký tự đã lưu trữ trong cơ sở dữ liệu. Phương pháp này hoạt động tốt với các văn bản in ấn có định đạng chuẩn nhưng gặp khó khăn với các văn bản viết tay hoặc phông chữ phức tạp.
- **Phương pháp phân tích đặc trưng (Feature Extraction):** Phương pháp này trích xuất các đặc trưng cơ bản của ký tự như cạnh, góc và đường cong. Sau đó sử dụng các thuật toán như k-nearest neighbors (k-NN) hoặc support vector machines (SVM) để phân loại các ký tự. Phương pháp này chính xác hơn so với template matching nhưng vẫn bị giới hạn khi xử lý các văn bản không đồng nhất.
- **Nhận diện ký tự quang học dựa trên đồ thị (Graph – based OCR):** Một số hệ thống OCR truyền thống sử dụng cấu trúc đô thị để biểu diễn các ký tự. Sau đó áp dụng các thuật toán tìm kiếm để nhận diện văn bản. Cách tiếp cận có khả năng xử lý tốt các ký tự hình dạng phức tạp nhưng đòi hỏi nhiều tính toán và không hiệu quả với các hình ảnh bị nhiễu.
- **Hạn chế của OCR truyền thống:** Mặc dù đã đạt được một số thành công, OCR truyền thống có những hạn chế nhất định như khó khăn trong việc nhận diện văn bản viết tay, xử lý các phông chữ lạ và hiệu suất giảm khi hình ảnh bị nhiễu hoặc biến dạng. Ngoài ra, các phương pháp này cũng đòi hỏi nhiều bước tiền xử lý và không có khả năng học hỏi và thích ứng với dữ liệu mới.

2.3.1.2. OCR dựa trên học sâu

Với sự ra đời của trí tuệ nhân tạo (AI) và học sâu (Deep Learning), OCR đã có những bước tiến vượt bậc. Các mô hình học sâu không chỉ cải thiện độ chính

xác mà còn có khả năng xử lý các văn bản phức tạp hơn, bao gồm cả văn bản viết tay và các ngôn ngữ tượng hình.

- **Mô hình mạng nơ – ron tích chập (Convolutional Neural Networks – CNNs):** CNNs là nền tảng của nhiều hệ thống OCR hiện đại. Chúng có khả năng tự động trích xuất các đặc trưng từ hình ảnh mà không cần sự can thiệp của con người. CNNs đã chứng minh được hiệu suất vượt trội trong việc nhận diện văn bản từ hình ảnh, đặc biệt là trong các môi trường phức tạp và đa dạng.
- **Mô hình LSTM và CRNN:** Để xử lý các văn bản viết tay hoặc văn bản có độ dài không cố định, các mô hình Long Short – Term Memory (LSTM) và Convolutional Recurrent Neural Networks (CRNN) được sử dụng rộng rãi. LSTM có khả năng ghi nhớ và xử lý các chuỗi ký tự liên tục, trong khi CRNN kết hợp sức mạnh của CNNs và RNNs để nhận diện văn bản trong các chuỗi dài.
- **Mô hình Transformer:** Transformer là một cải tiến mới trong OCR, sử dụng cơ chế attention để tập trung vào các phần quan trọng của hình ảnh và nhận diện văn bản một cách chính xác hơn. Transformer đã được chứng minh là có hiệu quả vượt trội trong nhiều tác vụ liên quan đến ngôn ngữ tự nhiên và hiện đang được áp dụng rộng rãi trong OCR.
- **Ưu điểm của OCR dựa trên học sâu:** OCR dựa trên học sâu có khả năng xử lý tốt hơn các hình ảnh phức tạp, nhận diện chính xác hơn trong các điều kiện biến dạng hoặc nhiễu, và đặc biệt là có khả năng học hỏi và thích ứng với dữ liệu mới. Các mô hình này có thể nhận diện nhiều ngôn ngữ và ký tự khác nhau, bao gồm cả các ngôn ngữ tượng hình như tiếng Trung Quốc và tiếng Nhật.
- **Thách thức và hạn chế:** Mặc dù có nhiều ưu điểm, OCR dựa trên học sâu cũng đối mặt với các thách thức như yêu cầu về tài nguyên tính toán lớn, thời gian huấn luyện dài, và cần nhiều dữ liệu huấn luyện chất lượng cao. Ngoài ra, việc giải thích kết quả từ các mô hình học sâu vẫn còn là một thách thức lớn.

2.3.2. Các phương pháp phân loại OCR theo ngôn ngữ, ký tự và ngữ cảnh.

OCR không chỉ được phân loại dựa trên phương pháp kỹ thuật mà còn theo ngôn ngữ, ký tự và ngữ cảnh sử dụng. Dưới đây là một số phương pháp phân loại phổ biến:

2.3.2.1. Phân loại theo ngôn ngữ

- **OCR Đơn ngữ**: Hệ thống OCR được thiết kế để nhận diện và xử lý văn bản trong một ngôn ngữ duy nhất. Các hệ thống này thường có độ chính xác cao hơn vì chúng được huấn luyện chuyên biệt cho một ngôn ngữ cụ thể.
- **OCR Đa ngôn ngữ:** Các hệ thống OCR đa ngôn ngữ có khả năng nhận diện và xử lý văn bản trong nhiều ngôn ngữ khác nhau. Đây là một thách thức lớn vì mỗi ngôn ngữ có cấu trúc ký tự, cú pháp và phong cách viết riêng biệt. Ví dụ, một hệ thống OCR đa ngôn ngữ có thể nhận diện tiếng Anh, tiếng Pháp, tiếng Trung và tiếng Ả Rập cùng một lúc.
- **OCR Tượng hình:** Đây là các hệ thống OCR được thiết kế để nhận diện các ngôn ngữ sử dụng ký tự tượng hình như tiếng Trung, tiếng Nhật, và tiếng Hàn. Các ngôn ngữ này có cấu trúc ký tự phức tạp và đòi hỏi các mô hình OCR phải có khả năng xử lý các chi tiết nhỏ và độ phân giải cao.

2.3.2.2. Phân loại theo loại ký tự.

- **OCR Nhận diện ký tự in:** Hệ thống OCR chuyên xử lý các văn bản in ấn với các ký tự chuẩn và dễ nhận diện như Times New Roman, Arial, hoặc Helvetica. Các hệ thống này thường có độ chính xác cao với văn bản in từ các tài liệu, sách, báo và tài liệu văn phòng.
- **OCR Nhận diện ký tự viết tay:** Đây là một trong những loại OCR phức tạp nhất vì văn bản viết tay có nhiều biến thể và phong cách khác nhau. Các hệ thống OCR viết tay thường sử dụng các mô hình học sâu như LSTM hoặc CRNN để nhận diện các chuỗi ký tự liên tục và không đồng nhất.
- **OCR Nhận diện ký tự đặc biệt:** Một số hệ thống OCR được thiết kế để nhận diện các ký tự đặc biệt hoặc ký hiệu, chẳng hạn như các ký hiệu toán học, ký tự Unicode, hoặc các biểu tượng đặc biệt trong ngành kỹ thuật. Các hệ thống này đòi hỏi phải được huấn luyện với các tập dữ liệu chuyên biệt và phức tạp.

2.3.2.3. Phân loại theo ngữ cảnh sử dụng:

- **OCR Trong văn phòng và công nghiệp:** Các hệ thống OCR được sử dụng trong môi trường văn phòng và doanh nghiệp thường được tối ưu hóa để xử lý các tài liệu hành chính, báo cáo, và thư từ. Những hệ thống này cần có khả năng xử lý nhanh chóng và chính xác, đồng thời hỗ trợ nhiều định dạng tài liệu khác nhau.
- **OCR Trong công nghiệp:** Trong môi trường công nghiệp, OCR được sử dụng để kiểm tra chất lượng sản phẩm, đọc mã vạch, và nhận diện các ký tự trên sản phẩm hoặc linh kiện. Các hệ thống này phải đáp ứng được các yêu cầu về độ bền,tốc độ và khả năng làm việc trong các điều kiện khắc nghiệt.
- **OCR Trong y tế:** Các ứng dụng OCR trong y tế bao gồm nhận diện thông tin từ hồ sơ bệnh án, đơn thuốc, và các kết quả xét nghiệm. Độ chính xác là yếu tố quyết định trong lĩnh vực này vì nó ảnh hưởng trực tiếp đến sức khỏe và sự an toàn của bệnh nhân.
- **OCR Trong giáo dục:** OCR được sử dụng trong giáo dục để số hóa sách giáo khoa, tài liệu học tập, và bài kiểm tra. Hệ thống này giúp việc truy cập tri thức trở nên dễ dàng hơn, đồng thời hỗ trợ các công cụ học tập trực tuyến và giáo dục từ xa.

2.3.3. Tổng quan về các công cụ và phần mềm OCR hiện nay.

Hiện nay, có rất nhiều công cụ và phần mềm OCR được phát triển và ứng dụng rộng rãi trong các lĩnh vực khác nhau. Dưới đây là tổng quan về một số công cụ OCR phổ biến và được đánh giá cao:

2.3.3.1. Tesseract OCR

- **Giới Thiệu:** Tesseract OCR là một trong những công cụ OCR mã nguồn mở phổ biến nhất, được phát triển bởi Google. Tesseract hỗ trợ nhiều ngôn ngữ và có khả năng nhận diện văn bản từ hình ảnh một cách chính xác.
- **Ưu Điểm:**
  - **Hỗ trợ đa ngôn ngữ:** Teseract hỗ trợ hơn 100 ngôn ngữ, bao gồm cả các ngôn ngữ tượng hình như tiếng Trung Quốc và tiếng Nhật.

- **Tính linh hoạt:** Tesseract có thể tích hợp với nhiều ngôn ngữ lập trình như Python, C++ và Java, giúp các nhà phát triển dễ dàng xây dựng các ứng dụng OCR tùy chỉnh.
- **Công đồng lớn:** Vì là mã nguồn mở, Tesseract có một cộng đồng phát triển lớn và năng động, giúp cung cấp tài liệu, hỗ trợ và các bản cập nhật thường xuyên.

- **Nhược Điểm:**
  - **Hiệu suất:** Mặc dù Tesseract hoạt động tốt với các văn bản in ấn chuẩn nhưng nó gặp khó khăn với các hình ảnh chất lượng thấp hoặc văn bản viết tay
  - **Cần cấu hình:** Để đạt được kết quả tốt nhất, Tesseract cần được cấu hình và tinh chỉnh phù hợp với từng loại văn bản và hình ảnh cụ thể.

2.3.3.2. Google Cloud Vision OCR

- **Giới Thiệu:** Google Cloud Vision OCR là một dịch vụ OCR dựa trên đám mây, cung cấp bởi Google Cloud. Công cụ này sử dụng các mô hình học sâu tiên tiến để nhận diện văn bản từ hình ảnh và tài liệu một cách chính xác và nhanh chóng.
- **Ưu Điểm:**
  - **Khả năng mở rộng:** Là một dịch vụ đám mây, Google Cloud Vision OCR có khả năng xử lý hàng triệu tài liệu một cách nhanh chóng và hiệu quả, mà không cần đến cơ sở hạ tầng phần cứng phức tạp.
  - **Tích hợp AI:** Google Cloud Vision OCR không chỉ nhận diện văn bản mà còn có thể phân loại hình ảnh, nhận diện khuôn mặt, và phát hiện các đối tượng trong hình ảnh, mang lại nhiều ứng dụng hơn cho người dùng.
  - **Hỗ trợ đa ngôn ngữ:** Dịch vụ này hỗ trợ nhiều ngôn ngữ và có khả năng nhận diện văn bản trong các ngôn ngữ tượng hình và phức tạp.
- **Nhược Điểm:**
  - **Chi phí:** Sử dụng dịch vụ đám mây có thể phát sinh chi phí đáng kể, đặc biệt là khi xử lý số lượng lớn tài liệu.

- **Yêu cầu kết nối Internet:** Vì là dịch vụ đám mây, Google Cloud Vision OCR yêu cầu kết nối Internet ổn định để hoạt động, điều này có thể gây khó khăn trong các môi trường làm việc không có kết nối mạng.

2.3.3.3. ABBYY FineReader.

- **Giới Thiệu:** ABBYY FineReader là một phần mềm OCR thương mại được phát triển bởi ABBYY, nổi tiếng với khả năng nhận diện văn bản chính xác và dễ sử dụng.
- **Ưu Điểm:**
  - **Độ chính xác cao:** ABBYY FineReader được biết đến với độ chính xác cao trong việc nhận diện văn bản từ các tài liệu in ấn, đặc biệt là các tài liệu phức tạp như sách, báo, và các tài liệu nhiều cột.
  - **Giao diện thân thiện:** Phần mềm này có giao diện người dùng thân thiện, dễ sử dụng, với các tính năng tùy chỉnh phong phú, phù hợp cho cả người dùng chuyên nghiệp và cá nhân.
  - **Chức năng số hóa PDF:** ABBYY FineReader cung cấp các công cụ mạnh mẽ để số hóa và chỉnh sửa tài liệu PDF, giúp người dùng dễ dàng chuyển đổi và xử lý các tài liệu số.
- **Nhược Điểm:**
  - **Chi phí:** ABBYY FineReader là một phần mềm thương mại, do đó người dùng cần phải trả phí để sử dụng, điều này có thể là một rào cản đối với một số cá nhân và doanh nghiệp nhỏ.
  - **Tùy biến hạn chế:** Mặc dù ABBYY FineReader có nhiều tính năng mạnh mẽ, nhưng nó không linh hoạt như các giải pháp OCR mã nguồn mở hoặc tùy chỉnh.

2.3.3.4. Adobe Acrobat OCR.

- **Giới Thiệu:** Adobe Acrobat là một phần mềm quản lý tài liệu nổi tiếng, trong đó bao gồm chức năng OCR để nhận diện văn bản từ các tài liệu PDF.
- **Ưu Điểm:**
  - **Tích hợp tốt với PDF:** Adobe Acrobat OCR được tối ưu hóa cho các tài liệu PDF, giúp người dùng dễ dàng số hóa, chỉnh sửa và tìm kiếm văn bản trong các tài liệu PDF.

- **Chất lượng hình ảnh:** Adobe Acrobat OCR cung cấp các công cụ để cải thiện chất lượng hình ảnh trước khi nhận diện văn bản, đảm bảo kết quả nhận diện tốt nhất.
- **Khả năng tìm kiếm:** Sau khi OCR được thực hiện, người dùng có thể tìm kiếm văn bản trong các tài liệu PDF một cách dễ dàng, tăng cường hiệu quả làm việc và quản lý tài liệu.

- **Nhược Điểm:**
  - **Chi phí cao:** Adobe Acrobat là một phần mềm thương mại, và chi phí của nó có thể là một rào cản lớn đối với một số người dùng.
  - **Giới hạn trong PDF:** Mặc dù mạnh mẽ trong việc xử lý PDF, Adobe Acrobat OCR có thể không phù hợp cho các ứng dụng OCR khác ngoài việc xử lý tài liệu PDF.

2.3.4. Kết luận

Phần này đã cung cấp một cái nhìn tổng quan và chi tiết về sự phân loại của OCR. Từ các phương pháp truyền thống đến các mô hình học sâu hiện đại cùng với các phương pháp phân loại theo ngôn ngữ, ký tự và ngữ cảnh. Ngoài ra, chúng ta cũng đã khám phá một số công cụ và phần mềm OCR phổ biến hiện nay. Từ các giải pháp mã nguồng mở như Tesseract đến các phần mềm thương mại như Adobe Acrobat và ABBYY FineReader. Mỗi công cụ và phương pháp đều có những ưu điểm và hạn chế riêng, việc lựa chọn giải pháp OCR phù hợp vào nhu cầu cụ thể của từng ứng dụng.

2.4. Các phương pháp truyền thống trong OCR.

Nhận diện ký tự quang học (OCR) đã phát triển qua nhiều thập kỷ với sự ra đời của các phương pháp truyền thống, bao gồm phương pháp dựa trên mẫu, kỹ thuật phân đoạn và phân loại ký tự và các thuật toán cổ điển như Support Vector Machine (SVM) và k – Nearest Neighbors (KNN). Mặc dù các phương pháp này hiện này đã bị thay thế phần lớn bởi vì các mô hình học sâu tiên tiến nhưng chúng vẫn đóng vai trò nền tảng trong lịch sử phát triển của OCR và là cơ sở cho nhiều nghiên cứu hiện đại.

2.4.1. Phương pháp dựa trên mẫu (Template Matching)

Phương pháp dựa trên mẫu, hay còn gọi là Template Matching, là một trong những phương pháp OCR đầu tiên và đơn giản nhất. Phương pháp này dựa trên việc so sánh trực tiếp hình ảnh của ký tự cần nhận diện với với một tập hợp các mẫu ký tự đã được định nghĩa trước trong cơ sở dữ liệu.

2.4.1.1. Nguyên lý hoạt động.

Nguyên lý hoạt động của Template Matching rất đơn giản: mỗi ký tự trong văn bản cần nhận diện được so sánh với các mẫu ký tự tương ứng trong cơ sở dữ liệu. Ký tự được nhận diện là ký tự có sự tương đồng cao nhất với mẫu trong cơ sở dữ liệu. Quá trình này thường bao gồm các bước sau:

- **Tiền xử lý hình ảnh:** Trước khi thực hiện Template Matching, hình ảnh văn bản cần được xử lý để loại bỏ nhiễu và chuẩn hóa kích thước, màu sắc. Điều này giúp cải thiện độ chính xác khi so sánh với các mẫu ký tự.
- **So sánh mẫu (Template Matching):** Mỗi ký tự trong hình ảnh được cắt ra và so sánh với các mẫu ký tự trong cơ sở dữ liệu. So sánh này thường dựa trên các biện pháp như phép đo khoảng cách (Euclidean distance) hoặc phép đo tương quan (cross-correlation)..
- **Phân loại ký tự:** Ký tự trong hình ảnh sẽ được phân loại dựa trên mẫu có mức độ tương đồng cao nhất. Ví dụ, nếu một ký tự trong hình ảnh có sự tương đồng cao nhất với mẫu chữ "A" trong cơ sở dữ liệu, nó sẽ được nhận diện là chữ "A".

2.4.1.2. Ưu điểm và nhược điểm.

- Ưu Điểm:

- **Đơn giản và dễ hiểu:** Template Matching là phương pháp rất đơn giản và dễ hiểu, không yêu cầu kiến thức sâu về toán học hay học máy.
- **Hiệu quả với văn bản chuẩn:** Phương pháp này hoạt động rất tốt với các văn bản in ấn có định dạng chuẩn và phông chữ đồng nhất, nơi các ký tự ít biến dạng.

- Nhược Điểm:

- **Không hiệu quả với văn bản viết tay hoặc phông chữ phức tạp:** Template Matching hoạt động kém hiệu quả với các văn bản viết tay, phông chữ lạ,

hoặc các hình ảnh văn bản bị biến dạng, do sự khác biệt lớn giữa ký tự trong hình ảnh và các mẫu trong cơ sở dữ liệu.

- **Không khả thi với đa ngôn ngữ:** Đối với các hệ thống yêu cầu nhận diện nhiều ngôn ngữ, việc sử dụng Template Matching trở nên phức tạp vì cần phải lưu trữ một lượng lớn mẫu ký tự từ các ngôn ngữ khác nhau.

2.4.1.3. Ứng dụng thực tế

Template Matching từng được sử dụng rộng rãi trong các hệ thống OCR thương mại đầu tiên, chủ yếu để nhận diện văn bản in từ các tài liệu hành chính, sách báo, và các biểu mẫu. Tuy nhiên, với sự phát triển của các kỹ thuật học máy hiện đại, Template Matching dần bị thay thế bởi các phương pháp tiên tiến hơn, nhưng vẫn đóng vai trò quan trọng trong việc phát triển các hệ thống nhận diện văn bản đơn giản và chuyên dụng.

2.4.2. Phân đoạn và phân loại ký tự

Một trong những bước quan trọng trong quá trình OCR là phân đoạn hình ảnh văn bản thành các ký tự riêng lẻ, sau đó phân loại chúng. Đây là một quá trình phức tạp, đòi hỏi phải xử lý nhiều vấn đề liên quan đến chất lượng hình ảnh, kích thước ký tự, và sự biến dạng của văn bản.

2.4.2.1. Phân đoạn ký tự

Phân đoạn ký tự là quá trình tách từng ký tự riêng lẻ từ một dòng văn bản hoặc từ một khối văn bản. Quá trình này thường bắt đầu bằng việc xác định các đường viền xung quanh các ký tự, sau đó cắt chúng ra khỏi nền.

- **Phân đoạn dòng và từ:** Trước khi phân đoạn ký tự, hình ảnh văn bản cần được phân đoạn thành các dòng và từ. Việc này thường được thực hiện bằng cách tìm các khoảng trắng giữa các dòng và từ, sau đó sử dụng các phương pháp phát hiện cạnh để xác định ranh giới.
- **Phân đoạn ký tự:** Sau khi phân đoạn từ, từng từ sẽ được tiếp tục phân đoạn thành các ký tự riêng lẻ. Quá trình này có thể phức tạp hơn nếu các ký tự chồng chéo hoặc dính liền nhau, đặc biệt trong các văn bản viết tay hoặc văn bản bị biến dạng.

- **Xử lý biến dạng**: Trong trường hợp các ký tự bị biến dạng hoặc chồng chéo, cần sử dụng các kỹ thuật bổ sung như xử lý hình thái (morphological processing) hoặc phát hiện đường viền nâng cao để phân đoạn chính xác.

2.4.2.2. Phân loại ký tự.

Sau khi phân đoạn, bước tiếp theo là phân loại các ký tự dựa trên các đặc trưng trích xuất từ hình ảnh của chúng. Đây là bước quan trọng trong quá trình OCR, quyết định độ chính xác của hệ thống nhận diện.

- **Trích xuất đặc trưng:** Quá trình phân loại bắt đầu bằng việc trích xuất các đặc trưng từ hình ảnh ký tự, chẳng hạn như các đường viền, góc, cạnh, và cấu trúc hình học của ký tự. Các đặc trưng này giúp phân biệt các ký tự khác nhau và phục vụ cho quá trình phân loại.
- **Phân loại bằng phương pháp thống kê:** Các phương pháp thống kê như k-Nearest Neighbors (KNN) hoặc Support Vector Machines (SVM) thường được sử dụng để phân loại ký tự dựa trên các đặc trưng trích xuất. Những phương pháp này sẽ so sánh các đặc trưng của ký tự cần nhận diện với các đặc trưng của ký tự trong cơ sở dữ liệu để xác định ký tự tương ứng.
- **Phân loại bằng phương pháp dựa trên quy tắc**: Ngoài các phương pháp thống kê, phân loại ký tự cũng có thể được thực hiện bằng cách sử dụng các quy tắc dựa trên hình dạng và kích thước của ký tự. Ví dụ, các ký tự như "O" và "Q" có hình dạng tròn, trong khi "L" và "T" có các đường thẳng đặc trưng.

2.4.2.3. Thách thức trong phân đoạn và phân loại ký tự

Quá trình phân đoạn và phân loại ký tự gặp nhiều thách thức, đặc biệt khi xử lý các hình ảnh văn bản không đồng nhất hoặc bị biến dạng.

- **Văn bản viết tay:** Phân đoạn và phân loại văn bản viết tay là một trong những thách thức lớn nhất do sự không đồng nhất trong cấu trúc và kích thước của các ký tự.
- **Văn bản chồng chéo:** Trong một số trường hợp, các ký tự có thể chồng chéo lên nhau, gây khó khăn cho việc phân đoạn chính xác. Điều này thường xảy ra trong các tài liệu cũ hoặc hình ảnh bị biến dạng.

- **Chất lượng hình ảnh thấp:** Hình ảnh văn bản có chất lượng thấp, bị mờ hoặc nhiễu cũng ảnh hưởng đáng kể đến quá trình phân đoạn và phân loại, làm giảm độ chính xác của hệ thống OCR.

2.4.3. Các thuật toán cổ điển như SVM, KNN trong OCR.

Các thuật toán cổ điển như Support Vector Machine (SVM) và k-Nearest Neighbors (KNN) đã đóng vai trò quan trọng trong việc phát triển các hệ thống OCR truyền thống. Mặc dù hiện nay các mô hình học sâu đã thay thế phần lớn các phương pháp này, nhưng SVM và KNN vẫn là nền tảng cho nhiều nghiên cứu và ứng dụng OCR.

2.4.3.1. Support Vector Machine (SVM)

Support Vector Machine (SVM) là một thuật toán học máy được sử dụng rộng rãi trong OCR để phân loại các ký tự dựa trên các đặc trưng trích xuất từ hình ảnh.

- **Nguyên lý hoạt động**: SVM hoạt động bằng cách tìm một siêu phẳng (hyperplane) trong không gian đặc trưng để phân chia các mẫu dữ liệu thuộc các lớp khác nhau. SVM sẽ tối ưu hóa siêu phẳng này sao cho khoảng cách giữa các mẫu dữ liệu gần nhất (các vector hỗ trợ) và siêu phẳng là lớn nhất, giúp tối đa hóa độ chính xác phân loại.
- **Ưu điểm:** SVM có khả năng phân loại chính xác ngay cả khi dữ liệu không tuyến tính, bằng cách sử dụng các hàm kernel để ánh xạ dữ liệu từ không gian gốc sang không gian cao hơn, nơi mà siêu phẳng có thể phân tách được các lớp dữ liệu.
- **Ứng dụng trong OCR:** SVM được sử dụng để phân loại các ký tự sau khi đã trích xuất các đặc trưng từ hình ảnh. SVM hoạt động tốt với các tập dữ liệu có kích thước vừa và nhỏ, và có thể được huấn luyện để nhận diện nhiều loại ký tự khác nhau.
- **Nhược điểm:** SVM không hiệu quả với các tập dữ liệu lớn do thời gian huấn luyện dài và yêu cầu tính toán cao. Ngoài ra, SVM cần được điều chỉnh các tham số một cách cẩn thận để đạt được hiệu suất tốt nhất.

2.4.3.2. k – Nearest Neighbors (KNN).

k-Nearest Neighbors (KNN) là một thuật toán học máy dựa trên khoảng cách, được sử dụng để phân loại các ký tự trong OCR.

- **Nguyên lý hoạt động:** KNN hoạt động dựa trên nguyên lý "hàng xóm gần nhất", trong đó một mẫu mới được phân loại dựa trên nhãn của k mẫu gần nhất trong không gian đặc trưng. Khoảng cách giữa các mẫu thường được tính bằng khoảng cách Euclidean.
- **Ưu điểm:** KNN rất đơn giản và dễ hiểu, không yêu cầu nhiều tính toán khi áp dụng. KNN có khả năng hoạt động tốt trên các tập dữ liệu nhỏ và không yêu cầu quá trình huấn luyện phức tạp như SVM.
- **Ứng dụng trong OCR:** KNN được sử dụng trong OCR để phân loại các ký tự dựa trên các đặc trưng trích xuất. Vì tính đơn giản của nó, KNN thường được sử dụng như một chuẩn mực để so sánh với các thuật toán phân loại khác.
- **Nhược điểm:** KNN trở nên kém hiệu quả khi số lượng mẫu tăng lên, vì nó phải tính toán khoảng cách cho tất cả các mẫu trong tập huấn luyện, dẫn đến chi phí tính toán lớn. Ngoài ra, KNN cũng nhạy cảm với sự phân bố của dữ liệu và có thể bị ảnh hưởng bởi các mẫu nhiễu.

2.4.4. Kết luận

Phần này đã trình bày các phương pháp truyền thống trong OCR, bao gồm phương pháp dựa trên mẫu, kỹ thuật phân đoạn và phân loại ký tự, cùng với các thuật toán cổ điển như SVM và KNN. Mặc dù các phương pháp này đã bị thay thế phần lớn bởi các mô hình học sâu hiện đại, nhưng chúng vẫn đóng vai trò quan trọng trong lịch sử phát triển của OCR và là nền tảng cho nhiều nghiên cứu hiện nay. Sự hiểu biết về các phương pháp truyền thống này không chỉ giúp chúng ta đánh giá được những thành tựu đã đạt được, mà còn giúp định hướng cho các nghiên cứu và cải tiến trong tương lai.

2.5. Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) là một trong những mô hình học sâu (Deep Learning) phổ biến và mạnh mẽ nhất trong việc xử lý dữ liệu hình ảnh. CNN đã mang lại những đột phá quan trọng trong nhiều lĩnh vực khác nhau, đặc

biệt là trong nhận diện văn bản, nhận diện hình ảnh, và phân loại đối tượng. Dưới đây, chúng ta sẽ đi sâu vào cấu trúc của một mạng CNN cơ bản, các loại lớp trong CNN, và vai trò của CNN trong nhận diện văn bản.

2.5.1. Cấu trúc của một mạng cnn cơ bản

Một mạng CNN cơ bản thường bao gồm một chuỗi các lớp xử lý dữ liệu được thiết kế để học các đặc trưng từ dữ liệu đầu vào (thường là hình ảnh). Cấu trúc của CNN thường được chia thành hai phần chính: phần trích xuất đặc trưng (feature extraction) và phần phân loại (classification).

2.5.1.1. Phần trích xuất đặc trưng

Phần trích xuất đặc trưng của CNN bao gồm một loạt các lớp Convolutional và Pooling. Những lớp này hoạt động cùng nhau để trích xuất các đặc trưng quan trọng từ hình ảnh đầu vào, chẳng hạn như cạnh, góc, và cấu trúc hình học.

- **Convolutional Layers:** Đây là các lớp đầu tiên trong CNN, chịu trách nhiệm trích xuất các đặc trưng cơ bản từ hình ảnh, như cạnh, góc, và các mẫu hình học đơn giản. Convolutional layers sử dụng các bộ lọc (filters) để thực hiện phép tích chập (convolution) trên hình ảnh đầu vào, tạo ra các bản đồ đặc trưng (feature maps).
- **Pooling Layers:** Sau các lớp convolutional, các lớp pooling được sử dụng để giảm kích thước không gian của các bản đồ đặc trưng, giúp giảm thiểu số lượng tham số và tính toán trong mạng. Các lớp pooling cũng giúp mạng CNN trở nên ít nhạy cảm hơn với các biến đổi vị trí của đối tượng trong hình ảnh.

2.5.1.2. Phần phân loại

Sau khi các đặc trưng đã được trích xuất và giảm kích thước bởi các lớp convolutional và pooling, phần phân loại của CNN sẽ sử dụng các đặc trưng này để đưa ra dự đoán cuối cùng. Phần này thường bao gồm các lớp Fully Connected và một lớp đầu ra (output layer).

- **Fully Connected Layers:** Đây là các lớp kết nối hoàn toàn, nơi mỗi nơ-ron trong lớp này kết nối với tất cả các nơ-ron trong lớp trước đó. Các lớp fully connected hoạt động như một mạng nơ-ron truyền thống, chịu trách nhiệm kết hợp các đặc trưng đã trích xuất để phân loại đối tượng.

- **Output Layer:** Lớp cuối cùng của CNN là lớp đầu ra, sử dụng các hàm kích hoạt (activation function) như softmax hoặc sigmoid để đưa ra xác suất cho từng lớp phân loại. Ví dụ, trong nhận diện văn bản, lớp đầu ra sẽ đưa ra xác suất ký tự thuộc về một trong những lớp ký tự cụ thể.

2.5.2. Các loại lớp (layers) trong CNN

Một mạng CNN bao gồm nhiều loại lớp khác nhau, mỗi lớp đảm nhiệm một vai trò cụ thể trong quá trình học và trích xuất đặc trưng từ dữ liệu đầu vào. Dưới đây là các loại lớp phổ biến nhất trong CNN.

2.5.2.1. Convolutional Layer

Convolutional Layer là thành phần cốt lõi của một mạng CNN, chịu trách nhiệm trích xuất các đặc trưng cơ bản từ hình ảnh đầu vào.

- **Phép tích chập (Convolution Operation):** Tại mỗi lớp convolutional, một tập hợp các bộ lọc (filters) được sử dụng để thực hiện phép tích chập trên hình ảnh đầu vào. Mỗi bộ lọc có nhiệm vụ phát hiện một đặc trưng cụ thể, chẳng hạn như cạnh, góc, hoặc đường cong. Kết quả của phép tích chập là các bản đồ đặc trưng (feature maps), biểu diễn sự hiện diện của các đặc trưng tại các vị trí khác nhau trong hình ảnh.
- **Kích thước bộ lọc:** Kích thước của các bộ lọc thường là 3x3, 5x5, hoặc 7x7, và số lượng bộ lọc có thể thay đổi tùy thuộc vào độ phức tạp của mô hình. Số lượng bộ lọc càng lớn thì số lượng đặc trưng trích xuất càng nhiều, nhưng cũng đồng nghĩa với việc số lượng tham số và chi phí tính toán tăng lên.
- **Đệm (Padding) và Bước (Stride):** Padding là kỹ thuật thêm các pixel giá trị 0 xung quanh biên của hình ảnh để kiểm soát kích thước đầu ra. Stride xác định số lượng bước mà bộ lọc di chuyển trên hình ảnh, với stride lớn hơn dẫn đến kích thước bản đồ đặc trưng nhỏ hơn.
- **Vai trò của Convolutional layer:** Convolutional Layer đóng vai trò quan trọng trong việc trích xuất các đặc trưng từ dữ liệu đầu vào, giúp CNN học cách phát hiện các mẫu phức tạp trong hình ảnh, từ đó phân biệt các đối tượng khác nhau.

2.5.2.2. Pooling Layer

Pooling Layer là lớp được sử dụng để giảm kích thước không gian của các bản đồ đặc trưng, đồng thời giữ lại các thông tin quan trọng.

- **Phép gộp (Pooling Operation):** Phép gộp phổ biến nhất là Max Pooling, trong đó lớp pooling chọn giá trị lớn nhất từ một cửa sổ (window) của các pixel trong bản đồ đặc trưng. Một loại pooling khác là Average Pooling, nơi giá trị trung bình của các pixel trong cửa sổ được tính toán.
- **Kích thước cửa sổ (Pooling Window):** Kích thước cửa sổ thường là 2x2 hoặc 3x3, và phép gộp được thực hiện trên toàn bộ bản đồ đặc trưng. Kết quả là kích thước của bản đồ đặc trưng giảm xuống, giúp giảm thiểu số lượng tham số và chi phí tính toán trong mạng.
- **Stride và Padding trong Pooling:** Stride trong pooling xác định khoảng cách di chuyển của cửa sổ trên bản đồ đặc trưng. Stride lớn hơn dẫn đến kích thước đầu ra nhỏ hơn. Padding trong pooling ít phổ biến hơn, nhưng có thể được sử dụng để kiểm soát kích thước đầu ra.
- **Vai trò của Pooling Layer:** Pooling Layer giúp giảm kích thước không gian của các bản đồ đặc trưng, làm cho mạng CNN ít nhạy cảm hơn với các biến đổi vị trí và tăng cường khả năng khái quát hóa của mô hình.

2.5.2.3. Fully Connected Layer

Fully Connected Layer là lớp mà mỗi nơ-ron trong lớp này kết nối với tất cả các nơ-ron trong lớp trước đó, tương tự như trong các mạng nơ-ron truyền thống.

- **Kết nối toàn phần:** Các lớp fully connected kết nối tất cả các đặc trưng đã trích xuất từ các lớp trước đó với các lớp đầu ra, giúp kết hợp các đặc trưng này để đưa ra dự đoán cuối cùng.
- **Hàm kích hoạt:** Lớp fully connected thường sử dụng các hàm kích hoạt như ReLU (Rectified Linear Unit) để giới thiệu tính phi tuyến trong mô hình, và softmax hoặc sigmoid cho lớp đầu ra để tính toán xác suất cho từng lớp phân loại.

- **Vai trò của Fully Connected Layer:** Lớp fully connected đóng vai trò kết hợp các đặc trưng đã trích xuất và đưa ra quyết định phân loại cuối cùng, từ đó xác định đối tượng trong hình ảnh hoặc nhận diện văn bản.

2.5.3. Vai trò của CNN trong nhận diện văn bản

CNN đã chứng minh được hiệu quả vượt trội trong nhiều nhiệm vụ liên quan đến xử lý hình ảnh, và nhận diện văn bản là một trong số đó. Dưới đây là những vai trò quan trọng của CNN trong nhận diện văn bản.

2.5.3.1. Trích xuất đặc trưng tự động

CNN có khả năng trích xuất các đặc trưng từ hình ảnh một cách tự động và hiệu quả, điều này rất quan trọng trong nhận diện văn bản. Thay vì phải sử dụng các kỹ thuật trích xuất đặc trưng thủ công, CNN có thể học và trích xuất các đặc trưng một cách trực tiếp từ dữ liệu huấn luyện.

- **Đặc trưng cấp thấp và cấp cao:** Trong các lớp đầu tiên của CNN, các đặc trưng cấp thấp như cạnh, góc, và đường cong được trích xuất. Ở các lớp sâu hơn, các đặc trưng phức tạp hơn, như các mẫu chữ cái và từ, được hình thành, giúp CNN nhận diện văn bản chính xác hơn.
- **Khả năng xử lý đa dạng ngôn ngữ và phông chữ:** CNN có thể học và trích xuất đặc trưng từ các phông chữ và ngôn ngữ khác nhau, từ đó giúp nhận diện văn bản trong các ngữ cảnh đa ngôn ngữ và đa dạng văn bản.

2.5.3.2. Khả Năng Tổng Quát Hóa Tốt

CNN có khả năng tổng quát hóa tốt, nghĩa là nó có thể nhận diện văn bản từ các hình ảnh mới chưa từng gặp trong quá trình huấn luyện.

- **Học từ dữ liệu đa dạng:** Bằng cách huấn luyện trên một tập dữ liệu lớn và đa dạng, CNN có thể học các mẫu phức tạp và phát hiện các đặc trưng chung của văn bản, từ đó nhận diện chính xác các văn bản trong các điều kiện khác nhau.
- **Khả năng đối phó với biến đổi**: CNN có khả năng xử lý các biến đổi trong hình ảnh, chẳng hạn như thay đổi vị trí, xoay, và biến dạng, nhờ vào các lớp pooling và các bộ lọc trong các lớp convolutional.

2.5.3.3. Hiệu Suất Cao Trong Các Nhiệm Vụ Phức Tạp

CNN đã chứng minh được hiệu suất cao trong các nhiệm vụ nhận diện văn bản phức tạp, bao gồm nhận diện văn bản viết tay, văn bản in từ nhiều nguồn khác nhau, và văn bản trong các hình ảnh có độ phân giải thấp hoặc bị nhiễu.

- **Nhận diện văn bản viết tay:** Một trong những ứng dụng khó nhất của OCR là nhận diện văn bản viết tay, nơi mà các ký tự không đồng nhất và có nhiều biến thể. CNN có khả năng học các mẫu và biến thể trong văn bản viết tay, từ đó nhận diện chính xác các ký tự.
- **Nhận diện văn bản trong hình ảnh phức tạp:** CNN cũng được sử dụng để nhận diện văn bản trong các hình ảnh phức tạp, chẳng hạn như biển báo giao thông, văn bản trên các vật thể, và văn bản trong các điều kiện ánh sáng kém. Khả năng trích xuất đặc trưng mạnh mẽ của CNN giúp nó đối phó tốt với các thách thức này.

2.5.3.4. Tích Hợp Trong Các Ứng Dụng Thực Tiễn

CNN đã được tích hợp vào nhiều ứng dụng OCR thực tiễn, từ các phần mềm nhận diện văn bản trên điện thoại di động đến các hệ thống tự động hóa công nghiệp.

- **Ứng dụng trong công nghiệp**: CNN được sử dụng để nhận diện văn bản trên các sản phẩm, bao bì, và nhãn hiệu trong các dây chuyền sản xuất. Khả năng xử lý nhanh và chính xác của CNN giúp tăng cường hiệu suất và độ chính xác của các hệ thống tự động hóa.
- **Ứng dụng trong y tế:** Trong lĩnh vực y tế, CNN được sử dụng để nhận diện văn bản từ các hồ sơ bệnh án, đơn thuốc, và kết quả xét nghiệm. Điều này giúp giảm thiểu lỗi do con người gây ra và cải thiện chất lượng chăm sóc y tế.

2.5.4. Kết luận

Phần này đã cung cấp một cái nhìn tổng quan và chi tiết về Convolutional Neural Networks (CNN), từ cấu trúc cơ bản của mạng đến các loại lớp trong CNN, và vai trò quan trọng của CNN trong nhận diện văn bản. CNN đã trở thành một công cụ mạnh mẽ và không thể thiếu trong lĩnh vực nhận diện văn bản, giúp cải thiện đáng kể độ chính xác và hiệu suất của các hệ thống OCR. Hiểu rõ về

CNN không chỉ giúp bạn đánh giá được những tiến bộ trong lĩnh vực này, mà còn cung cấp nền tảng vững chắc cho các nghiên cứu và ứng dụng OCR trong tương lai.

2.6. Các mô hình CNN tiên tiến.

Convolutional Neural Networks (CNN) đã phát triển mạnh mẽ với sự ra đời của nhiều mô hình tiên tiến, giúp cải thiện đáng kể hiệu suất trong các nhiệm vụ xử lý hình ảnh và nhận diện văn bản. Trong số các mô hình CNN tiên tiến, DenseNet và VGG là hai kiến trúc nổi bật, đã chứng minh hiệu quả trong nhiều ứng dụng khác nhau. Dưới đây, chúng ta sẽ khám phá chi tiết về hai mô hình này.

2.6.1. DenseNet: Kiến Trúc và Lợi Ích

2.6.1.1. Giới thiệu về DenseNet

DenseNet (Densely Connected Convolutional Networks) là một mô hình CNN tiên tiến được giới thiệu bởi Gao Huang và các đồng nghiệp vào năm 2017. DenseNet nổi bật với kiến trúc đặc biệt, trong đó mỗi lớp được kết nối trực tiếp với tất cả các lớp trước đó trong mạng. Điều này trái ngược với các mạng CNN truyền thống, nơi mà mỗi lớp chỉ nhận đầu vào từ lớp liền kề trước đó.

- **Dense Blocks:** Kiến trúc của DenseNet bao gồm nhiều "Dense Blocks", mỗi block bao gồm một tập hợp các lớp convolutional. Trong mỗi block, đầu ra của mỗi lớp convolutional được kết nối trực tiếp với tất cả các lớp sau đó, thông qua phép nối (concatenation). Điều này giúp tăng cường sự truyền thông tin và làm giảm vấn đề mất mát thông tin trong mạng sâu.
- **Transition Layers:** Giữa các Dense Blocks, DenseNet sử dụng các lớp chuyển tiếp (Transition Layers) để giảm kích thước không gian của các đặc trưng, thông qua các phép pooling và convolution. Điều này giúp kiểm soát số lượng tham số trong mạng, đồng thời duy trì hiệu suất tính toán.

2.6.1.2. Lợi Ích của DenseNet

DenseNet mang lại nhiều lợi ích so với các mô hình CNN truyền thống, bao gồm khả năng tăng cường truyền thông tin, giảm thiểu số lượng tham số, và cải thiện khả năng khái quát hóa của mô hình.

- **Giảm thiểu số lượng tham số:** Bằng cách kết nối mỗi lớp với tất cả các lớp trước đó, DenseNet có thể giảm thiểu số lượng tham số cần thiết, vì mỗi lớp có thể tái sử dụng các đặc trưng đã học từ các lớp trước đó. Điều này không chỉ giảm thiểu tình trạng overfitting mà còn làm cho mô hình hiệu quả hơn trong việc học các đặc trưng phức tạp.
- **Giảm thiểu vấn đề Vanishing Gradient:** Trong các mạng sâu, vấn đề vanishing gradient thường xảy ra, khiến cho các lớp đầu tiên không được huấn luyện đúng cách. DenseNet giải quyết vấn đề này bằng cách cung cấp đường truyền gradient ngắn hơn giữa các lớp thông qua kết nối trực tiếp, giúp cải thiện quá trình huấn luyện.
- **Cải thiện khả năng tái sử dụng đặc trưng:** DenseNet khuyến khích việc tái sử dụng các đặc trưng đã học từ các lớp trước đó, giúp mạng học được nhiều đặc trưng hơn mà không cần tăng kích thước của mạng. Điều này đặc biệt hữu ích trong các ứng dụng nhận diện văn bản, nơi cần nhận diện các đặc trưng phức tạp từ hình ảnh.
- **Tăng cường truyền thông tin**: Kết nối giữa tất cả các lớp giúp thông tin được truyền đi một cách hiệu quả hơn, giảm thiểu sự mất mát thông tin trong quá trình huấn luyện.

2.6.1.3. Ứng Dụng của DenseNet

DenseNet đã được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau, bao gồm nhận diện hình ảnh, phân loại đối tượng, và đặc biệt là trong nhận diện văn bản.

- **Nhận diện văn bản:** DenseNet đã được sử dụng trong nhiều hệ thống nhận diện văn bản để trích xuất các đặc trưng phức tạp từ hình ảnh văn bản. Khả năng kết nối dày đặc của DenseNet giúp mạng học được các đặc trưng từ các phần khác nhau của văn bản, từ đó cải thiện độ chính xác của hệ thống OCR.
- **Y tế:** DenseNet cũng được áp dụng trong lĩnh vực y tế, chẳng hạn như trong chẩn đoán hình ảnh y khoa, nơi cần phân loại các bệnh dựa trên các đặc trưng nhỏ và tinh tế trong ảnh chụp.

2.6.2. VGG: Cấu trúc mạng và các phiên bản khác nhau (VGG16, VGG19)

2.6.2.1. Giới thiệu về VGG

VGG (Visual Geometry Group) là một trong những mô hình CNN nổi tiếng nhất, được phát triển bởi các nhà nghiên cứu tại Đại học Oxford vào năm 2014. VGG nổi bật với kiến trúc đơn giản và dễ hiểu, sử dụng các lớp convolutional với kích thước bộ lọc nhỏ (3x3) và các lớp pooling xen kẽ.

- **VGG16 và VGG19:** Hai phiên bản phổ biến nhất của VGG là VGG16 và VGG19, được đặt tên dựa trên số lượng lớp trọng số trong mạng. VGG16 có 16 lớp trọng số (13 lớp convolutional và 3 lớp fully connected), trong khi VGG19 có 19 lớp trọng số (16 lớp convolutional và 3 lớp fully connected). Cả hai phiên bản này đều đã đạt được kết quả ấn tượng trong nhiều bài toán xử lý hình ảnh, bao gồm phân loại ảnh và nhận diện đối tượng.

2.6.2.2. Cấu trúc mạng của VGG

VGG sử dụng một kiến trúc rất đặc biệt với các lớp convolutional liên tiếp sử dụng kích thước bộ lọc 3x3, nhằm trích xuất các đặc trưng từ hình ảnh một cách chi tiết và chính xác.

- **Kích thước bộ lọc nhỏ:** Việc sử dụng các bộ lọc nhỏ (3x3) trong tất cả các lớp convolutional giúp VGG có thể học được các đặc trưng phức tạp từ hình ảnh mà không cần quá nhiều tham số. Kích thước bộ lọc nhỏ cũng giúp giảm thiểu số lượng tham số và tăng cường khả năng tổng quát hóa của mô hình.
- **Lớp Pooling:** Sau mỗi khối convolutional, VGG sử dụng các lớp pooling để giảm kích thước không gian của các bản đồ đặc trưng, đồng thời giữ lại các thông tin quan trọng. Việc này giúp giảm thiểu số lượng tham số và chi phí tính toán trong mạng.
- **Lớp Fully Connected:** Sau khi các đặc trưng đã được trích xuất và giảm kích thước, các lớp fully connected sẽ được sử dụng để kết hợp các đặc trưng này và đưa ra dự đoán cuối cùng. Lớp fully connected trong VGG thường sử dụng hàm kích hoạt ReLU để tăng cường tính phi tuyến trong mô hình, và softmax cho lớp đầu ra để tính toán xác suất cho từng lớp phân loại.

- **Tính đơn giản và hiệu quả:** Mặc dù có kiến trúc đơn giản, VGG đã chứng minh hiệu quả vượt trội trong nhiều nhiệm vụ xử lý hình ảnh. Đơn giản trong thiết kế nhưng mạnh mẽ trong khả năng học, VGG đã trở thành một chuẩn mực cho nhiều nghiên cứu và ứng dụng trong lĩnh vực học sâu.

2.6.2.3. Các phiên bản khác nhau của VGG

VGG có nhiều phiên bản khác nhau, trong đó VGG16 và VGG19 là hai phiên bản phổ biến nhất. Sự khác biệt chính giữa chúng nằm ở số lượng lớp trọng số và độ sâu của mạng.

- **VGG16:** VGG16 bao gồm 13 lớp convolutional và 3 lớp fully connected, với tổng số 16 lớp trọng số. Mạng này đã đạt được kết quả ấn tượng trong cuộc thi ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2014, và trở thành một trong những mô hình CNN phổ biến nhất trong lĩnh vực xử lý hình ảnh.
- **VGG19:** VGG19 là phiên bản mở rộng của VGG16, với 16 lớp convolutional và 3 lớp fully connected, tổng cộng 19 lớp trọng số. VGG19 có độ sâu lớn hơn, giúp mô hình học được nhiều đặc trưng phức tạp hơn, nhưng đồng thời cũng yêu cầu nhiều tài nguyên tính toán hơn.
- **So sánh VGG16 và VGG19:** VGG16 thường được ưa chuộng hơn trong các ứng dụng yêu cầu sự cân bằng giữa hiệu suất và tài nguyên tính toán, trong khi VGG19 được sử dụng khi cần hiệu suất cao hơn và có sẵn tài nguyên tính toán mạnh mẽ.

2.6.2.4. Ứng Dụng của VGG

VGG đã được áp dụng rộng rãi trong nhiều lĩnh vực khác nhau, từ nhận diện hình ảnh đến nhận diện văn bản, và thậm chí là trong các ứng dụng y tế.

- **Nhận diện văn bản:** VGG đã được sử dụng để phát triển các hệ thống OCR tiên tiến, có khả năng nhận diện văn bản từ các hình ảnh có độ phức tạp cao. Độ sâu và khả năng trích xuất đặc trưng của VGG giúp cải thiện đáng kể độ chính xác của các hệ thống OCR, đặc biệt là trong nhận diện văn bản viết tay và văn bản từ các ngôn ngữ khác nhau.

- **Phân loại ảnh:** VGG đã đạt được những kết quả xuất sắc trong các cuộc thi phân loại ảnh, chẳng hạn như ImageNet, và trở thành một trong những mô hình CNN được sử dụng rộng rãi nhất trong lĩnh vực này.
- **Y tế:** Trong lĩnh vực y tế, VGG đã được áp dụng để phân loại và phát hiện các bệnh từ ảnh chụp X-quang, MRI, và các hình ảnh y khoa khác. Khả năng học sâu của VGG giúp phát hiện các đặc trưng tinh tế trong các ảnh y tế, từ đó hỗ trợ chẩn đoán và điều trị bệnh.

2.6.3. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết và toàn diện về hai mô hình CNN tiên tiến: DenseNet và VGG. Cả hai mô hình này đã chứng minh hiệu quả vượt trội trong nhiều nhiệm vụ xử lý hình ảnh và nhận diện văn bản, và chúng tiếp tục đóng vai trò quan trọng trong sự phát triển của các ứng dụng học sâu hiện đại. DenseNet với kiến trúc kết nối dày đặc, và VGG với sự đơn giản nhưng mạnh mẽ, đều mang lại những lợi ích độc đáo và là nguồn cảm hứng cho nhiều nghiên cứu và cải tiến trong lĩnh vực học sâu.

2.7. Các kỹ thuật tăng hiệu suất trong CNN.

Trong quá trình huấn luyện các mô hình Convolutional Neural Networks (CNN), việc đảm bảo mô hình đạt hiệu suất cao là điều rất quan trọng. Điều này đòi hỏi các kỹ thuật và chiến lược phù hợp để cải thiện khả năng học và giảm thiểu các vấn đề như overfitting, vanishing gradient, hay thiếu dữ liệu. Dưới đây là các kỹ thuật tăng cường hiệu suất phổ biến trong CNN, bao gồm Regularization, Batch Normalization, và Data Augmentation, đặc biệt trong bối cảnh OCR.

2.7.1. Regularization (Dropout, L2 Regularization)

Regularization là một trong những kỹ thuật quan trọng nhất để giảm thiểu hiện tượng overfitting trong quá trình huấn luyện mạng CNN. Overfitting xảy ra khi mô hình quá phù hợp với dữ liệu huấn luyện và không thể tổng quát hóa tốt trên dữ liệu mới, dẫn đến hiệu suất kém trên tập kiểm tra. Regularization giúp giảm thiểu overfitting bằng cách hạn chế khả năng của mô hình trong việc học quá mức các đặc trưng không quan trọng từ dữ liệu huấn luyện.

2.7.1.1. Dropout

Dropout là một kỹ thuật Regularization rất hiệu quả được giới thiệu bởi Srivastava và các đồng nghiệp vào năm 2014. Dropout hoạt động bằng cách "tắt" ngẫu nhiên một số lượng nơ-ron trong mỗi lớp mạng trong quá trình huấn luyện.

- **Nguyên lý hoạt động:** Trong mỗi lần huấn luyện (iteration), một tỷ lệ phần trăm ngẫu nhiên các nơ-ron trong một lớp được "tắt" (không tham gia vào quá trình tính toán). Điều này buộc mạng phải học cách dựa vào nhiều nơ-ron khác nhau để đưa ra dự đoán, thay vì dựa vào một số nơ-ron cụ thể.
- **Tỷ lệ Dropout:** Thông thường, tỷ lệ dropout được đặt trong khoảng từ 0.2 đến 0.5, nghĩa là từ 20% đến 50% số lượng nơ-ron sẽ bị "tắt" trong mỗi lần huấn luyện. Trong giai đoạn kiểm tra (test), tất cả các nơ-ron đều được kích hoạt, nhưng đầu ra của mỗi nơ-ron được nhân với tỷ lệ dropout để đảm bảo tính nhất quán với quá trình huấn luyện.
- **Lợi ích của Dropout:** Dropout giúp giảm thiểu overfitting bằng cách buộc mô hình trở nên "dự trữ" hơn, học các đặc trưng chung hơn thay vì các đặc trưng cụ thể cho dữ liệu huấn luyện. Điều này giúp mô hình tổng quát hóa tốt hơn trên các dữ liệu mới.

2.7.1.2. L2 Regularization

L2 Regularization, còn được gọi là weight decay, là một kỹ thuật khác để giảm thiểu overfitting bằng cách thêm một "hình phạt" vào hàm mất mát của mô hình dựa trên độ lớn của các trọng số.

- **Nguyên lý hoạt động:** Trong L2 Regularization, một thành phần bổ sung tỉ lệ thuận với bình phương của các trọng số được thêm vào hàm mất mát. Điều này khuyến khích các trọng số nhỏ hơn, giúp mạng tránh việc học quá mức từ các đặc trưng không quan trọng.
- **Hàm mất mát có Regularization:** Hàm mất mát mới sẽ có dạng: L(θ)=L0(θ)+λ∑iθi2L(\theta)=L_0(\theta)+\lambda\sum_i\theta_i^2L(θ)=L0(θ)+λi∑θi2

Trong đó, L0(θ)L_0(\theta)L0(θ) là hàm mất mát gốc (ví dụ như cross-entropy), θi\theta_iθi là các trọng số của mạng, và λ\lambdaλ là hệ số Regularization (thường là một số nhỏ như 0.0001).

- **Lợi ích của L2 Regularization:** L2 Regularization giúp tránh overfitting bằng cách kiểm soát độ lớn của các trọng số, đảm bảo rằng mô hình không quá phụ thuộc vào các trọng số lớn và do đó có khả năng tổng quát hóa tốt hơn.

2.7.2. Batch normalization và tác động đến quá trình huấn luyện

Batch Normalization là một kỹ thuật được giới thiệu bởi Sergey Ioffe và Christian Szegedy vào năm 2015 để giải quyết vấn đề vanishing gradient và tăng tốc độ huấn luyện mạng sâu. Kỹ thuật này đã trở thành một thành phần không thể thiếu trong các mô hình CNN hiện đại.

2.7.2.1. Nguyên lý hoạt động của Batch Normalization

Batch Normalization hoạt động bằng cách chuẩn hóa đầu vào của mỗi lớp trong mạng dựa trên trung bình và độ lệch chuẩn của nó trong một batch nhỏ. Cụ thể, tại mỗi lớp, đầu vào sẽ được chuẩn hóa bằng cách:

- **Chuẩn hóa:** Đầu vào xix_ixi sẽ được chuẩn hóa bằng cách trừ đi giá trị trung bình μB\mu_BμB của batch và chia cho độ lệch chuẩn σB\sigma_BσB:x^i=xi−μBσB\hat{x}_i=\frac{x_i\mu_B}{\sigma_B}x^i=σBxi−μB
- **Tính toán giá trị mới:** Sau khi chuẩn hóa, một giá trị mới được tính toán dựa trên các tham số học γ\gammaγ và β\betaβ:yi=γx^i+βy_i = \gamma \hat{x}_i + \betayi=γx^i+β
- **Giá trị học được:** Các tham số γ\gammaγ và β\betaβ cho phép mạng khôi phục lại phân phối ban đầu của đầu vào nếu cần thiết, mang lại sự linh hoạt trong quá trình huấn luyện.

2.7.2.2. Lợi Ích Của Batch Normalization

Batch Normalization mang lại nhiều lợi ích trong quá trình huấn luyện mô hình CNN:

- **Tăng tốc độ huấn luyện**: Bằng cách chuẩn hóa đầu vào của mỗi lớp, Batch Normalization giúp giảm thiểu sự thay đổi của phân phối đầu vào (internal covariate shift), cho phép mô hình hội tụ nhanh hơn và giảm số lượng epochs cần thiết.

- **Giảm thiểu Vanishing Gradient:** Batch Normalization giúp ổn định gradient trong suốt quá trình huấn luyện, giảm thiểu vấn đề vanishing gradient trong các mạng sâu, từ đó giúp cải thiện hiệu suất huấn luyện.
- **Giảm thiểu sự phụ thuộc vào Initial Weights:** Batch Normalization giúp giảm sự phụ thuộc vào các giá trị khởi tạo trọng số ban đầu, cho phép mô hình đạt được hiệu suất cao hơn mà không cần điều chỉnh quá nhiều các tham số ban đầu.
- **Tăng cường Regularization:** Batch Normalization có tác dụng như một dạng Regularization nhẹ, giúp giảm thiểu overfitting bằng cách thêm nhiễu vào quá trình huấn luyện thông qua sự dao động của các batch nhỏ.

2.7.3. Data Augmentation Trong OCR và Vai Trò Của Nó

Data Augmentation là một kỹ thuật quan trọng trong học sâu, đặc biệt là khi làm việc với các tập dữ liệu hình ảnh. Kỹ thuật này giúp tăng cường độ đa dạng của dữ liệu huấn luyện bằng cách tạo ra các phiên bản khác nhau của dữ liệu hiện có thông qua các phép biến đổi như xoay, lật, cắt, và thay đổi độ sáng. Trong bối cảnh OCR, Data Augmentation đóng vai trò quan trọng trong việc cải thiện hiệu suất của mô hình.

2.7.3.1. Data Augmentation Trong OCR

Trong OCR, Data Augmentation thường được áp dụng để tăng cường sự đa dạng của các hình ảnh văn bản huấn luyện, từ đó giúp mô hình học được các đặc trưng của văn bản trong nhiều điều kiện khác nhau.

- **Xoay (Rotation):** Xoay hình ảnh văn bản giúp mô hình học được các đặc trưng của văn bản khi nó xuất hiện ở các góc độ khác nhau. Điều này đặc biệt hữu ích khi nhận diện văn bản từ các tài liệu bị nghiêng hoặc biến dạng.
- **Lật (Flipping):** Lật ngang hoặc lật dọc hình ảnh giúp mô hình học cách nhận diện văn bản trong các trường hợp đối xứng hoặc ngược.
- **Cắt (Cropping):** Cắt hình ảnh văn bản để tạo ra các vùng quan trọng, giúp mô hình tập trung vào các phần quan trọng của văn bản và học được các đặc trưng chi tiết hơn.

- **Thay đổi độ sáng và Độ tương phản:** Điều chỉnh độ sáng và độ tương phản của hình ảnh giúp mô hình học cách nhận diện văn bản trong các điều kiện ánh sáng khác nhau, từ đó tăng cường khả năng khái quát hóa của mô hình.
- **Nhiễu (Noise):** Thêm nhiễu vào hình ảnh giúp mô hình học cách nhận diện văn bản trong các điều kiện không lý tưởng, chẳng hạn như văn bản bị mờ, nhiễu, hoặc có chất lượng thấp.

2.7.3.2. Vai Trò Của Data Augmentation

Data Augmentation mang lại nhiều lợi ích cho quá trình huấn luyện mô hình OCR:

- **Tăng cường số lượng dữ liệu:** Data Augmentation giúp tăng cường số lượng dữ liệu huấn luyện một cách nhân tạo mà không cần phải thu thập thêm dữ liệu mới, giúp mô hình học được nhiều đặc trưng hơn từ dữ liệu.
- **Cải thiện khả năng khái quát hóa:** Bằng cách giới thiệu nhiều biến thể của dữ liệu huấn luyện, Data Augmentation giúp mô hình trở nên mạnh mẽ hơn trước các biến đổi trong hình ảnh, từ đó cải thiện khả năng khái quát hóa và hiệu suất trên dữ liệu kiểm tra.
- **Giảm thiểu Overfitting:** Data Augmentation giúp giảm thiểu overfitting bằng cách tăng độ đa dạng của dữ liệu huấn luyện, khiến mô hình khó học quá mức các đặc trưng cụ thể từ dữ liệu huấn luyện ban đầu.

2.7.4. Kết luận

Phần này đã cung cấp một cái nhìn toàn diện về các kỹ thuật tăng cường hiệu suất trong CNN, bao gồm Regularization (Dropout, L2 Regularization), Batch Normalization, và Data Augmentation. Những kỹ thuật này không chỉ giúp cải thiện hiệu suất của mô hình mà còn giúp mô hình trở nên mạnh mẽ hơn trước các thách thức như overfitting, vanishing gradient, và thiếu dữ liệu. Hiểu rõ và áp dụng đúng các kỹ thuật này sẽ giúp bạn xây dựng các mô hình CNN hiệu quả và đạt được kết quả tốt hơn trong các ứng dụng thực tế, đặc biệt là trong nhận diện văn bản.

2.8. Học sâu không giám sát và bán giám sát trong OCR.

Trong lĩnh vực nhận diện văn bản (OCR), học sâu không giám sát và bán giám sát đang ngày càng trở nên quan trọng. Những kỹ thuật này giúp cải thiện hiệu quả của các mô hình OCR, đặc biệt khi đối mặt với các thách thức như thiếu nhãn dữ liệu và nhu cầu xử lý lượng lớn dữ liệu chưa được gán nhãn. Dưới đây, chúng ta sẽ tìm hiểu chi tiết về các mô hình học sâu không giám sát như Autoencoders và GANs, và cách mà học sâu bán giám sát có thể được áp dụng trong OCR.

2.8.1. Các mô hình học sâu không giám sát

Học sâu không giám sát là một lĩnh vực trong học sâu mà ở đó các mô hình được huấn luyện mà không cần các nhãn dữ liệu rõ ràng. Điều này có nghĩa là mô hình phải tự học cách biểu diễn dữ liệu một cách có ý nghĩa mà không dựa vào các nhãn do con người cung cấp. Trong OCR, các mô hình học sâu không giám sát có thể được sử dụng để trích xuất các đặc trưng từ dữ liệu văn bản mà không cần biết trước nhãn của các ký tự hoặc từ. Dưới đây là hai mô hình học sâu không giám sát phổ biến: Autoencoders và GANs.

2.8.1.1. Autoencoders

Autoencoders là một loại mạng nơ-ron được thiết kế để học cách mã hóa và giải mã dữ liệu đầu vào sao cho đầu ra gần giống với đầu vào nhất có thể. Mục tiêu của Autoencoders là học một biểu diễn nén (compressed representation) của dữ liệu, có thể được sử dụng để phát hiện các đặc trưng quan trọng hoặc giảm thiểu nhiễu trong dữ liệu.

- **Cấu trúc của Autoencoders:** Một Autoencoder bao gồm hai phần chính: bộ mã hóa (encoder) và bộ giải mã (decoder). Bộ mã hóa học cách nén dữ liệu đầu vào thành một mã số (code) có kích thước nhỏ hơn, trong khi bộ giải mã học cách tái tạo dữ liệu đầu vào từ mã số đó.
- **Học các biểu diễn ẩn:** Autoencoders học cách mã hóa dữ liệu sao cho các biểu diễn ẩn (latent representations) chứa các đặc trưng quan trọng nhất của dữ liệu. Các biểu diễn này có thể được sử dụng để trích xuất các đặc trưng từ hình ảnh văn bản trong OCR mà không cần nhãn.

- **Ứng dụng trong OCR:** Autoencoders có thể được sử dụng để làm sạch dữ liệu văn bản bị nhiễu hoặc giảm thiểu kích thước của dữ liệu mà vẫn giữ lại các thông tin quan trọng. Ví dụ, một Autoencoder có thể được sử dụng để học cách trích xuất các đặc trưng của văn bản từ các hình ảnh có chất lượng thấp hoặc bị nhiễu, từ đó cải thiện độ chính xác của mô hình OCR.
- **Biến thể của Autoencoders:** Có nhiều biến thể của Autoencoders như Denoising Autoencoders (DAE) dùng để loại bỏ nhiễu trong dữ liệu, và Variational Autoencoders (VAE) được sử dụng để tạo ra các biểu diễn ẩn có cấu trúc, giúp mô hình học các đặc trưng có tính thống kê cao.

2.8.1.2. Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) là một loại mô hình học sâu không giám sát được giới thiệu bởi Ian Goodfellow và các đồng nghiệp vào năm 2014. GANs bao gồm hai mạng nơ-ron đối kháng nhau: một mạng sinh (generator) và một mạng phân biệt (discriminator). Mục tiêu của GANs là tạo ra các dữ liệu tổng hợp (synthetic data) có chất lượng cao mà không cần nhãn.

- **Nguyên lý hoạt động**: Trong GANs, mạng sinh học cách tạo ra dữ liệu giả từ nhiễu ngẫu nhiên, trong khi mạng phân biệt học cách phân biệt giữa dữ liệu thật và dữ liệu giả. Hai mạng này học cùng nhau trong một trò chơi tổng bằng không (zero-sum game), nơi mạng sinh cố gắng tạo ra dữ liệu giả khó phân biệt với dữ liệu thật nhất, và mạng phân biệt cố gắng phát hiện dữ liệu giả một cách chính xác nhất.
- **Ứng dụng trong OCR:** GANs có thể được sử dụng để tạo ra các hình ảnh văn bản tổng hợp để huấn luyện các mô hình OCR khi dữ liệu thực tế bị thiếu hụt. Các hình ảnh văn bản tổng hợp này có thể được sử dụng để mở rộng tập dữ liệu huấn luyện, từ đó cải thiện hiệu suất của mô hình OCR.
- **Image-to-Image Translation:** GANs cũng có thể được sử dụng trong các bài toán chuyển đổi hình ảnh (image-to-image translation), chẳng hạn như chuyển đổi văn bản viết tay thành văn bản in hoặc cải thiện chất lượng hình ảnh văn bản bị mờ.
- **Biến thể của GANs:** Có nhiều biến thể của GANs như Conditional GANs (cGANs) cho phép điều kiện hóa quá trình sinh dữ liệu dựa trên nhãn hoặc

các đặc trưng cụ thể, và CycleGANs được sử dụng để chuyển đổi giữa hai miền hình ảnh mà không cần dữ liệu gán nhãn.

2.8.2. Học sâu bán giám sát Trong OCR

Học sâu bán giám sát là một kỹ thuật học máy kết hợp giữa học có giám sát và học không giám sát, trong đó một phần dữ liệu được gán nhãn và phần còn lại không được gán nhãn. Kỹ thuật này rất hữu ích trong các bài toán OCR khi chỉ có một lượng nhỏ dữ liệu được gán nhãn, trong khi hầu hết dữ liệu là chưa được gán nhãn.

2.8.2.1. Tầm quan trọng của học sâu bán giám sát

Trong nhiều ứng dụng OCR, việc thu thập và gán nhãn cho tất cả dữ liệu là rất tốn kém và mất thời gian. Tuy nhiên, có thể có rất nhiều dữ liệu chưa được gán nhãn có sẵn, và học sâu bán giám sát cho phép tận dụng dữ liệu này để cải thiện hiệu suất của mô hình.

- **Học từ dữ liệu thiếu nhãn:** Học sâu bán giám sát cho phép mô hình học từ một lượng nhỏ dữ liệu gán nhãn và sử dụng dữ liệu chưa được gán nhãn để cải thiện quá trình huấn luyện. Điều này giúp tiết kiệm chi phí và thời gian so với việc gán nhãn tất cả dữ liệu.
- **Cải thiện khả năng khái quát hóa:** Bằng cách sử dụng dữ liệu chưa được gán nhãn, mô hình có thể học được các đặc trưng chung của dữ liệu, giúp cải thiện khả năng khái quát hóa và hiệu suất trên tập kiểm tra.

2.8.2.2. Các phương pháp học sâu bán giám sát trong OCR

Có nhiều phương pháp khác nhau để áp dụng học sâu bán giám sát trong OCR, dưới đây là một số phương pháp phổ biến.

- **Pseudo-Labeling:** Trong phương pháp này, mô hình ban đầu được huấn luyện trên dữ liệu gán nhãn, sau đó được sử dụng để gán nhãn cho dữ liệu chưa được gán nhãn (gọi là pseudo-labels). Dữ liệu có pseudo-labels sau đó được sử dụng để tiếp tục huấn luyện mô hình.
- **Consistency Regularization:** Phương pháp này yêu cầu mô hình phải đưa ra các dự đoán nhất quán cho cùng một dữ liệu đầu vào, ngay cả khi đầu vào bị biến đổi (chẳng hạn như lật, xoay, hoặc thêm nhiễu). Điều này giúp mô hình học được các đặc trưng ổn định hơn từ dữ liệu.

- **Generative Models:** Các mô hình sinh, chẳng hạn như GANs hoặc VAEs, có thể được sử dụng để tạo ra các biểu diễn ẩn cho dữ liệu chưa được gán nhãn, từ đó giúp cải thiện quá trình học của mô hình OCR.
- **MixMatch:** Đây là một phương pháp bán giám sát tiên tiến kết hợp giữa pseudo-labeling, consistency regularization, và augmentation để tối ưu hóa quá trình huấn luyện. MixMatch đã cho thấy hiệu quả vượt trội trong nhiều bài toán xử lý hình ảnh và OCR.

2.8.2.3. Ứng dụng thực tiễn của học sâu bán giám sát trong OCR

Học sâu bán giám sát có rất nhiều ứng dụng trong OCR, đặc biệt là trong các tình huống khi chỉ có một lượng nhỏ dữ liệu gán nhãn hoặc khi dữ liệu gán nhãn rất khó thu thập.

- **Nhận diện văn bản từ dữ liệu thực tế:** Học sâu bán giám sát có thể được sử dụng để huấn luyện các hệ thống OCR từ dữ liệu thực tế, chẳng hạn như văn bản viết tay hoặc văn bản từ các ngôn ngữ ít phổ biến. Bằng cách sử dụng dữ liệu chưa được gán nhãn, các hệ thống OCR có thể học được các đặc trưng của văn bản một cách hiệu quả hơn.
- **Tăng cường tập dữ liệu huấn luyện**: Học sâu bán giám sát cho phép tăng cường tập dữ liệu huấn luyện bằng cách sử dụng dữ liệu chưa được gán nhãn, từ đó cải thiện độ chính xác và độ tin cậy của các hệ thống OCR.

2.8.3. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về học sâu không giám sát và bán giám sát trong OCR, bao gồm các mô hình không giám sát như Autoencoders và GANs, và các phương pháp bán giám sát như Pseudo-Labeling và Consistency Regularization. Những kỹ thuật này không chỉ giúp cải thiện hiệu suất của các hệ thống OCR mà còn giúp xử lý các thách thức liên quan đến dữ liệu thiếu nhãn và tăng cường khả năng tổng quát hóa của mô hình. Hiểu rõ và áp dụng đúng các kỹ thuật này sẽ giúp bạn phát triển các mô hình OCR hiệu quả hơn, đặc biệt trong các tình huống khi dữ liệu gán nhãn là hạn chế.

2.9. Recurrent Neural Networks (RNN) và LSTM trong OCR

2.9.1. Tổng quan về RNN và LSTM

2.9.1.1 Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNN) là một loại mạng nơ-ron được thiết kế để xử lý dữ liệu tuần tự, nơi mà mỗi phần tử của chuỗi dữ liệu không chỉ phụ thuộc vào đầu vào hiện tại mà còn vào các phần tử trước đó trong chuỗi.

- **Cấu trúc cơ bản của RNN:** Trong một RNN, đầu ra của mỗi nơ-ron tại một thời điểm sẽ trở thành đầu vào cho nơ-ron tại thời điểm tiếp theo. Điều này cho phép RNN lưu giữ thông tin từ các bước trước đó trong chuỗi dữ liệu, giúp nó học được các mẫu tuần tự.
- **Vấn đề Vanishing Gradient:** Mặc dù RNN có khả năng lưu giữ thông tin tuần tự, nhưng nó gặp phải vấn đề vanishing gradient khi học các chuỗi dài. Điều này xảy ra khi gradient biến mất hoặc trở nên quá nhỏ, khiến mô hình không thể học được các mẫu trong các chuỗi dài. Vấn đề này được giải quyết bằng cách sử dụng các biến thể của RNN như LSTM.

2.9.1.2. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) là một biến thể của RNN được thiết kế để giải quyết vấn đề vanishing gradient và giúp mô hình có khả năng học được các mối quan hệ dài hạn trong chuỗi dữ liệu.

- **Cấu trúc của LSTM:** LSTM sử dụng các "cell states" và các "gate" (cửa) để điều khiển dòng thông tin bên trong mạng. Cấu trúc cơ bản của một LSTM bao gồm ba loại cửa chính: Forget Gate, Input Gate, và Output Gate.
  - Forget Gate: Quyết định thông tin nào cần được loại bỏ từ cell state dựa trên đầu vào hiện tại và trạng thái ẩn từ bước trước đó.
  - Input Gate: Quyết định thông tin nào cần được thêm vào cell state.
  - Output Gate: Quyết định thông tin nào từ cell state sẽ được sử dụng để tạo ra đầu ra hiện tại.
- **Khả năng lưu trữ thông tin dài hạn:** Với cấu trúc này, LSTM có thể duy trì thông tin từ các bước trước đó trong chuỗi dữ liệu lâu hơn so với RNN truyền thống, giúp mô hình học được các mối quan hệ dài hạn trong dữ liệu tuần tự.

2.9.2. Ứng dụng của RNN và LSTM trong nhận diện văn bản tuần tự (Chuỗi Ký Tự)

RNN và LSTM được ứng dụng rộng rãi trong OCR, đặc biệt là khi cần nhận diện văn bản tuần tự, chẳng hạn như chuỗi ký tự trong văn bản viết tay hoặc văn bản in. Dưới đây là các ứng dụng cụ thể của RNN và LSTM trong nhận diện văn bản tuần tự.

2.9.2.1. Nhận diện văn bản viết tay

Nhận diện văn bản viết tay là một trong những ứng dụng nổi bật của LSTM trong OCR. Văn bản viết tay thường có sự thay đổi lớn về hình dạng, kích thước, và khoảng cách giữa các ký tự, khiến việc nhận diện trở nên phức tạp. LSTM giúp giải quyết vấn đề này bằng cách học các mối quan hệ tuần tự giữa các ký tự trong chuỗi.

- **LSTM trong nhận diện văn bản viết tay:** LSTM có thể học được cách mà các ký tự viết tay kết nối với nhau trong chuỗi văn bản, từ đó giúp mô hình nhận diện được văn bản viết tay một cách chính xác hơn. Bằng cách sử dụng các cell state và gate, LSTM có thể duy trì thông tin về các ký tự trước đó trong chuỗi, giúp tăng cường khả năng nhận diện của mô hình.
- **Chuỗi ký tự biến đổi:** Văn bản viết tay thường không theo một khuôn mẫu cố định, với các ký tự và từ ngữ có thể biến đổi về hình dạng và kích thước. LSTM giúp mô hình OCR học được cách nhận diện các biến đổi này, từ đó cải thiện độ chính xác của việc nhận diện văn bản viết tay.

2.9.2.2. Nhận diện văn bản từ hình ảnh in

Nhận diện văn bản từ hình ảnh in cũng là một ứng dụng quan trọng của RNN và LSTM. Mặc dù văn bản in thường có cấu trúc cố định hơn so với văn bản viết tay, nhưng nó vẫn đòi hỏi mô hình phải có khả năng nhận diện các chuỗi ký tự tuần tự trong các ngữ cảnh khác nhau.

- **RNN và LSTM trong nhận diện văn bản in:** RNN và LSTM có thể học được cách các ký tự liên kết với nhau trong văn bản in, từ đó giúp mô hình OCR nhận diện văn bản một cách hiệu quả. Điều này đặc biệt quan trọng trong các ứng dụng như đọc văn bản từ sách, báo, tài liệu kỹ thuật, hoặc văn bản từ biển báo.

- **Xử lý chuỗi dài:** Một trong những lợi thế lớn của LSTM là khả năng xử lý các chuỗi dài mà không gặp vấn đề về vanishing gradient, giúp mô hình nhận diện được các văn bản dài và phức tạp hơn.

2.9.3. Cơ chế Attention và cải tiến trong các mô hình OCR sử dụng RNN/LSTM

Cơ chế Attention là một trong những cải tiến quan trọng trong các mô hình OCR sử dụng RNN và LSTM. Attention cho phép mô hình tập trung vào các phần quan trọng của dữ liệu đầu vào khi đưa ra dự đoán, thay vì xử lý toàn bộ dữ liệu một cách đồng đều.

2.9.3.1. Cơ Chế Attention

Attention là một cơ chế cho phép mô hình học cách tập trung vào các phần quan trọng của chuỗi đầu vào khi thực hiện các dự đoán, thay vì xử lý toàn bộ chuỗi một cách đồng đều. Điều này đặc biệt hữu ích khi làm việc với các chuỗi dài hoặc phức tạp.

- **Nguyên lý hoạt động của Attention:** Attention hoạt động bằng cách gán một trọng số (weight) cho mỗi phần tử của chuỗi đầu vào, dựa trên tầm quan trọng của nó đối với đầu ra hiện tại. Những phần tử quan trọng hơn sẽ được gán trọng số cao hơn, giúp mô hình tập trung nhiều hơn vào chúng khi đưa ra dự đoán.
- **Self-Attention và Multi-Head Attention:** Self-Attention là một biến thể của Attention, trong đó mỗi phần tử của chuỗi đầu vào tương tác với tất cả các phần tử khác trong chuỗi để xác định tầm quan trọng của chúng. Multi-Head Attention là một kỹ thuật mở rộng, cho phép mô hình học nhiều mối quan hệ khác nhau giữa các phần tử của chuỗi đầu vào thông qua nhiều "đầu" Attention.

2.9.3.2. Cải tiến trong các mô hình OCR sử dụng RNN/LSTM

Cơ chế Attention đã mang lại nhiều cải tiến trong các mô hình OCR sử dụng RNN và LSTM, giúp cải thiện đáng kể hiệu suất của các hệ thống nhận diện văn bản.

- **Attention trong nhận diện văn bản:** Khi sử dụng trong OCR, Attention giúp mô hình tập trung vào các ký tự hoặc từ quan trọng trong văn bản, từ đó

cải thiện độ chính xác của việc nhận diện. Điều này đặc biệt hữu ích khi nhận diện văn bản trong các hình ảnh phức tạp, nơi mà các ký tự có thể bị che khuất hoặc biến dạng.

- **Kết hợp Attention với LSTM:** Sự kết hợp giữa Attention và LSTM đã tạo ra các mô hình mạnh mẽ hơn trong OCR, với khả năng xử lý hiệu quả các chuỗi ký tự dài và phức tạp. Attention giúp LSTM tập trung vào các phần quan trọng của chuỗi văn bản, trong khi LSTM giúp mô hình lưu giữ thông tin dài hạn.
- **Ứng dụng thực tiễn:** Các mô hình OCR sử dụng Attention và LSTM đã được áp dụng thành công trong nhiều ứng dụng thực tiễn, bao gồm nhận diện văn bản từ tài liệu kỹ thuật, nhận diện văn bản từ các biển báo giao thông, và nhận diện văn bản từ các tài liệu viết tay hoặc văn bản in phức tạp.

2.9.4. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về vai trò của RNN và LSTM trong OCR, cũng như cơ chế Attention và các cải tiến mà nó mang lại. RNN và LSTM, với khả năng xử lý dữ liệu tuần tự, đã chứng minh được hiệu quả cao trong việc nhận diện văn bản tuần tự, đặc biệt là trong các ứng dụng như nhận diện văn bản viết tay và văn bản in. Cơ chế Attention đã giúp nâng cao khả năng của các mô hình OCR sử dụng RNN và LSTM, mang lại những cải tiến đáng kể trong độ chính xác và hiệu suất. Hiểu rõ và áp dụng đúng các kỹ thuật này sẽ giúp bạn xây dựng các mô hình OCR mạnh mẽ và hiệu quả hơn trong các ứng dụng thực tế.

2.10. Mô hình Transformer và Vision Transformer (ViT) trong OCR

Mô hình Transformer đã trở thành một trong những đột phá quan trọng nhất trong lĩnh vực học sâu, đặc biệt trong các bài toán xử lý ngôn ngữ tự nhiên (NLP). Transformer đã mang lại những cải tiến lớn trong nhiều lĩnh vực, bao gồm OCR (Optical Character Recognition). Gần đây, Vision Transformer (ViT) đã được giới thiệu và chứng minh khả năng mạnh mẽ trong việc xử lý các bài toán thị giác máy tính, bao gồm nhận diện văn bản từ ảnh.

2.10.1. Nguyên lý hoạt động của Transformer

2.10.1.1. Giới thiệu về Transformer

Transformer là một kiến trúc mạng nơ-ron được giới thiệu trong bài báo "Attention is All You Need" của Vaswani et al. vào năm 2017. Transformer đã cách mạng hóa cách xử lý dữ liệu tuần tự trong NLP và đã được áp dụng rộng rãi trong nhiều lĩnh vực khác nhau.

- **Self-Attention:** Một trong những đặc điểm nổi bật của Transformer là cơ chế Self-Attention. Self-Attention cho phép mô hình học cách tập trung vào các phần quan trọng của chuỗi đầu vào khi đưa ra dự đoán, thay vì xử lý toàn bộ chuỗi một cách đồng đều. Điều này giúp Transformer hiệu quả hơn trong việc xử lý các chuỗi dài và phức tạp.
- **Kiến trúc Transformer:** Transformer bao gồm hai thành phần chính: Bộ mã hóa (Encoder) và bộ giải mã (Decoder). Cả hai đều bao gồm nhiều lớp Self-Attention và Feedforward Neural Network.
    - **Encoder:** Bộ mã hóa nhận đầu vào là một chuỗi và chuyển nó thành một biểu diễn ngữ cảnh phong phú, với mỗi phần tử trong chuỗi được mã hóa với thông tin từ toàn bộ chuỗi.
    - **Decoder:** Bộ giải mã sử dụng biểu diễn ngữ cảnh từ bộ mã hóa và thông tin từ chuỗi đích để tạo ra chuỗi đầu ra.
- **Position Encoding:** Vì Transformer không có khả năng tự nhiên để nhận biết thứ tự của các phần tử trong chuỗi, một vector vị trí (Position Encoding) được thêm vào đầu vào để cung cấp thông tin về thứ tự.

2.10.1.2. Lợi Ích của Transformer

Transformer mang lại nhiều lợi ích so với các kiến trúc RNN truyền thống:

- **Xử lý song song:** Khác với RNN, Transformer không yêu cầu xử lý tuần tự từng phần tử trong chuỗi mà có thể xử lý song song, giúp tăng tốc độ huấn luyện và dự đoán.
- **Học mối quan hệ dài hạn:** Cơ chế Self-Attention giúp Transformer học được các mối quan hệ dài hạn trong chuỗi một cách hiệu quả, mà không gặp phải vấn đề vanishing gradient như trong RNN.

- **Tính linh hoạt:** Transformer có thể dễ dàng mở rộng hoặc điều chỉnh để phù hợp với nhiều loại dữ liệu và bài toán khác nhau.

2.10.2. Ứng Dụng của Transformer trong OCR

2.10.2.1. Transformer trong Nhận Diện Văn Bản

Transformer đã được áp dụng thành công trong nhiều bài toán OCR, đặc biệt là trong nhận diện văn bản tuần tự và dịch ngôn ngữ từ ảnh.

- **Nhận diện văn bản tuần tự:** Transformer có khả năng xử lý các chuỗi ký tự trong văn bản, đặc biệt là trong các ngôn ngữ có cú pháp phức tạp. Nó giúp nhận diện các ký tự theo thứ tự chính xác trong một chuỗi văn bản, ngay cả khi có sự khác biệt về phông chữ hoặc kích thước ký tự.
- **Dịch ngôn ngữ từ ảnh:** Trong các ứng dụng như dịch ngôn ngữ từ biển báo hoặc tài liệu, Transformer giúp chuyển đổi chuỗi văn bản từ một ngôn ngữ sang ngôn ngữ khác bằng cách sử dụng bộ mã hóa để hiểu ngôn ngữ gốc và bộ giải mã để tạo ra chuỗi văn bản trong ngôn ngữ đích.
- **Phân loại văn bản:** Transformer cũng được sử dụng để phân loại các đoạn văn bản từ ảnh dựa trên nội dung hoặc ngữ cảnh, giúp tự động hóa quá trình xử lý văn bản từ tài liệu hoặc hình ảnh kỹ thuật số.

2.10.2.2. Các Mô Hình Transformer Hiện Đại trong OCR

Một số mô hình Transformer hiện đại đã được phát triển để tối ưu hóa quá trình nhận diện văn bản, bao gồm:

- **TrOCR (Transformer OCR):** TrOCR là một mô hình OCR dựa trên Transformer, được thiết kế đặc biệt để nhận diện văn bản từ hình ảnh. TrOCR sử dụng bộ mã hóa để xử lý hình ảnh đầu vào và bộ giải mã để tạo ra chuỗi văn bản tương ứng. TrOCR đã chứng minh hiệu suất vượt trội trong nhiều bài toán OCR so với các mô hình truyền thống.
- **DocFormer:** Đây là một mô hình OCR mới được phát triển bởi Microsoft, kết hợp giữa Transformer và CNN để nhận diện văn bản từ tài liệu và hình ảnh phức tạp. DocFormer sử dụng CNN để trích xuất đặc trưng từ hình ảnh và Transformer để xử lý các đặc trưng này, giúp nhận diện chính xác các ký tự và từ ngữ trong các tài liệu có cấu trúc phức tạp.

2.10.3. Vision Transformer (vit) và sự cải tiến trong nhận diện văn bản từ ảnh

Vision Transformer (ViT) là một cải tiến của mô hình Transformer, được thiết kế để xử lý các bài toán thị giác máy tính, bao gồm cả nhận diện văn bản từ ảnh. ViT đã mở ra một hướng đi mới trong lĩnh vực thị giác máy tính, với khả năng xử lý hiệu quả các dữ liệu hình ảnh.

2.10.3.1. Giới thiệu về Vision Transformer (ViT)

Vision Transformer là một mô hình học sâu được thiết kế để áp dụng cơ chế Self-Attention của Transformer vào các bài toán xử lý hình ảnh.

- **Patch Embedding:** Thay vì xử lý toàn bộ hình ảnh cùng một lúc, ViT chia hình ảnh thành các mảnh nhỏ (patches) và chuyển chúng thành các vector embedding. Mỗi vector này đại diện cho một phần của hình ảnh, tương tự như cách mà Transformer xử lý các token trong chuỗi văn bản.
- **Self-Attention trên các Patches:** Sau khi embedding các mảnh, ViT áp dụng cơ chế Self-Attention để học mối quan hệ giữa các mảnh này, cho phép mô hình hiểu được ngữ cảnh toàn cầu của hình ảnh.
- **Lợi ích của ViT:** ViT không yêu cầu các lớp convolutional truyền thống để trích xuất đặc trưng từ hình ảnh, mà thay vào đó sử dụng hoàn toàn cơ chế Attention. Điều này giúp ViT trở nên linh hoạt hơn trong việc học các đặc trưng phức tạp từ hình ảnh.

2.10.3.2. Ứng dụng của Vision Transformer trong OCR

ViT đã được chứng minh là rất hiệu quả trong các bài toán OCR, đặc biệt là trong nhận diện văn bản từ các hình ảnh có cấu trúc phức tạp.

- **Nhận diện văn bản từ hình ảnh:** ViT có thể nhận diện văn bản từ các hình ảnh có độ phân giải cao hoặc chứa nhiều chi tiết phức tạp. Khả năng học mối quan hệ giữa các mảnh của hình ảnh giúp ViT hiểu được ngữ cảnh và vị trí của các ký tự trong văn bản, từ đó nhận diện chính xác hơn.
- **Xử lý hình ảnh đa ngữ:** ViT có thể được áp dụng để nhận diện văn bản từ các hình ảnh chứa nhiều ngôn ngữ khác nhau, với khả năng xử lý các ký tự và từ ngữ từ nhiều hệ chữ khác nhau.

- **Cải thiện độ chính xác:** So với các mô hình OCR truyền thống, ViT mang lại sự cải thiện đáng kể về độ chính xác, đặc biệt trong các trường hợp văn bản bị biến dạng hoặc nằm trong các điều kiện ánh sáng không lý tưởng.

2.10.3.3. ViT kết hợp với CNN và Transformer

Một số nghiên cứu gần đây đã kết hợp ViT với CNN và các mô hình Transformer khác để tạo ra các hệ thống OCR mạnh mẽ hơn.

- **ViT-CNN Hybrid:** Một số mô hình kết hợp giữa ViT và CNN được phát triển để tận dụng lợi thế của cả hai kiến trúc. CNN được sử dụng để trích xuất các đặc trưng cơ bản từ hình ảnh, sau đó ViT xử lý các đặc trưng này để học mối quan hệ ngữ cảnh trong toàn bộ hình ảnh.
- **ViT và Self-Supervised Learning:** ViT cũng được áp dụng trong các bài toán học tự giám sát (self-supervised learning), nơi mà mô hình có thể học các đặc trưng từ hình ảnh mà không cần nhãn dữ liệu. Điều này đặc biệt hữu ích trong các bài toán OCR nơi mà dữ liệu gán nhãn khó thu thập.

2.10.4. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về mô hình Transformer và Vision Transformer (ViT) trong OCR. Transformer, với cơ chế Self-Attention mạnh mẽ, đã mở ra những cải tiến lớn trong xử lý ngôn ngữ tự nhiên và thị giác máy tính, bao gồm nhận diện văn bản từ ảnh. Vision Transformer, với cách tiếp cận mới mẻ trong xử lý hình ảnh, đã chứng minh được tiềm năng to lớn trong OCR, mang lại những cải tiến đáng kể về độ chính xác và hiệu suất. Hiểu rõ và áp dụng đúng các kỹ thuật này sẽ giúp bạn xây dựng các mô hình OCR hiện đại và hiệu quả hơn trong các ứng dụng thực tế.

2.11. Transfer Learning trong OCR.

Transfer Learning đã trở thành một phương pháp quan trọng trong học sâu, đặc biệt khi đối mặt với các bài toán yêu cầu nhiều tài nguyên như OCR (Optical Character Recognition). Phương pháp này cho phép tái sử dụng các mô hình đã được huấn luyện trên các tập dữ liệu lớn, từ đó giảm thiểu thời gian và công sức cần thiết để xây dựng các mô hình OCR hiệu quả.

2.11.1. Khái niệm và lợi ích của Transfer Learning

2.11.1.1. Khái niệm Transfer Learning

Transfer Learning là một kỹ thuật trong học sâu, nơi mà một mô hình đã được huấn luyện trên một nhiệm vụ cụ thể (ví dụ như phân loại ảnh) được tái sử dụng hoặc tinh chỉnh để thực hiện một nhiệm vụ mới (ví dụ như nhận diện văn bản). Thay vì huấn luyện mô hình từ đầu, Transfer Learning cho phép tận dụng kiến thức mà mô hình đã học được để áp dụng vào các nhiệm vụ mới, đặc biệt là khi tập dữ liệu mới không đủ lớn hoặc đa dạng.

- **Mô hình gốc (Pretrained Model):** Mô hình gốc là mô hình đã được huấn luyện trước đó trên một tập dữ liệu lớn, chẳng hạn như ImageNet. Các đặc trưng mà mô hình học được từ tập dữ liệu lớn này có thể hữu ích cho nhiều nhiệm vụ khác nhau, bao gồm OCR.
- **Fine-Tuning:** Fine-Tuning là quá trình tinh chỉnh các trọng số của mô hình gốc để phù hợp với nhiệm vụ mới. Trong OCR, điều này có thể bao gồm việc tinh chỉnh các lớp cuối cùng của mô hình để nó học cách nhận diện các ký tự và văn bản từ hình ảnh.

2.11.1.2. Lợi ích của Transfer Learning

Transfer Learning mang lại nhiều lợi ích quan trọng trong OCR:

- **Tiết kiệm thời gian và tài nguyên:** Transfer Learning giúp giảm thiểu thời gian và tài nguyên cần thiết để huấn luyện một mô hình từ đầu. Thay vì cần một lượng lớn dữ liệu và thời gian tính toán, Transfer Learning cho phép tái sử dụng các mô hình đã được huấn luyện và chỉ cần tinh chỉnh nhẹ để phù hợp với nhiệm vụ mới.
- **Cải thiện hiệu suất:** Các mô hình pretrained đã học được nhiều đặc trưng từ tập dữ liệu lớn, giúp cải thiện hiệu suất của mô hình trong các nhiệm vụ mới, đặc biệt là khi dữ liệu huấn luyện cho nhiệm vụ mới hạn chế về số lượng và đa dạng.
- **Khả năng tổng quát hóa tốt hơn:** Bằng cách sử dụng một mô hình đã được huấn luyện trên một tập dữ liệu đa dạng, mô hình mới có khả năng tổng quát hóa tốt hơn khi đối mặt với các dữ liệu không quen thuộc trong nhiệm vụ mới.

- **Ứng dụng linh hoạt:** Transfer Learning có thể được áp dụng cho nhiều loại dữ liệu và nhiệm vụ khác nhau, từ nhận diện ký tự đơn giản đến nhận diện văn bản trong các ngữ cảnh phức tạp như văn bản viết tay hoặc văn bản từ biển báo.

2.11.2. Các mô hình Pretrained phổ biến trong OCR

Có nhiều mô hình pretrained đã được phát triển và chứng minh hiệu quả cao trong OCR. Dưới đây là một số mô hình phổ biến nhất:

2.11.2.1. ImageNet Pretrained Models

ImageNet là một trong những tập dữ liệu lớn nhất và phổ biến nhất cho các nhiệm vụ phân loại ảnh. Các mô hình được huấn luyện trên ImageNet đã học được một lượng lớn các đặc trưng hữu ích, bao gồm cả các đặc trưng về hình dạng, kết cấu, và màu sắc, giúp chúng trở nên rất hiệu quả trong nhiều nhiệm vụ thị giác máy tính, bao gồm OCR.

- **ResNet:** ResNet (Residual Networks) là một trong những mô hình nổi tiếng nhất được huấn luyện trên ImageNet. Với kiến trúc đặc biệt giúp giảm thiểu vấn đề vanishing gradient, ResNet đã trở thành một trong những lựa chọn hàng đầu cho các tác vụ OCR khi áp dụng Transfer Learning.
- **Inception:** Mô hình Inception (GoogleNet) với cấu trúc phức tạp cho phép xử lý các đặc trưng ở nhiều mức độ khác nhau trong hình ảnh. Điều này làm cho Inception trở thành một công cụ mạnh mẽ khi được sử dụng cho OCR, đặc biệt là trong các nhiệm vụ yêu cầu nhận diện văn bản từ hình ảnh có độ phân giải cao.
- **VGG:** VGG, với cấu trúc đơn giản nhưng hiệu quả, đã được chứng minh là rất hữu ích khi áp dụng Transfer Learning trong OCR. Đặc biệt, các biến thể như VGG16 và VGG19 thường được sử dụng để nhận diện các ký tự trong văn bản in hoặc văn bản từ các tài liệu số hóa.

2.11.2.2. Các mô hình OCR đặc biệt

Ngoài các mô hình ImageNet, còn có những mô hình được thiết kế đặc biệt cho OCR và có thể được sử dụng với Transfer Learning:

- **CRNN (Convolutional Recurrent Neural Network):** CRNN kết hợp giữa CNN và RNN để nhận diện văn bản tuần tự. Mô hình này đặc biệt hữu ích trong việc nhận diện văn bản viết tay hoặc văn bản liên tục mà không cần phân đoạn từng ký tự.
- **TrOCR (Transformer OCR):** TrOCR là một mô hình OCR hiện đại dựa trên Transformer, đã được huấn luyện trước trên các tập dữ liệu văn bản lớn. TrOCR có khả năng nhận diện chính xác văn bản từ hình ảnh và là một lựa chọn tuyệt vời khi áp dụng Transfer Learning cho các nhiệm vụ OCR.

2.11.3. Cách tinh chỉnh mô hình (Fine-Tuning) cho các tác vụ OCR cụ thể

2.11.3.1. Các bước Fine-Tuning mô hình

Fine-Tuning là quá trình điều chỉnh các mô hình pretrained để phù hợp với nhiệm vụ OCR cụ thể. Dưới đây là các bước cơ bản để thực hiện Fine-Tuning:

1. **Chọn mô hình Pretrained:** Lựa chọn mô hình pretrained phù hợp nhất với nhiệm vụ OCR cụ thể, chẳng hạn như ResNet, Inception, hoặc TrOCR. Đảm bảo rằng mô hình được lựa chọn đã được huấn luyện trên tập dữ liệu có liên quan đến nhiệm vụ mới.
2. **Thay đổi lớp cuối cùng:** Tùy chỉnh lớp cuối cùng của mô hình để phù hợp với số lượng lớp cần thiết cho nhiệm vụ mới. Ví dụ, nếu bạn đang thực hiện nhận diện ký tự, lớp cuối cùng cần có số lượng đầu ra bằng số lượng ký tự cần nhận diện.
3. **Freeze các lớp đầu**: Trong quá trình Fine-Tuning, các lớp đầu tiên của mô hình thường được "đóng băng" (không cập nhật trọng số) để giữ lại các đặc trưng đã học từ nhiệm vụ trước. Chỉ các lớp cuối cùng được huấn luyện lại để phù hợp với nhiệm vụ mới.
4. **Huấn luyện lại mô hình:** Sử dụng dữ liệu OCR cụ thể để huấn luyện lại các lớp cuối cùng. Có thể cần điều chỉnh tốc độ học (learning rate) để đảm bảo quá trình Fine-Tuning diễn ra ổn định.
5. **Đánh giá và Tinh chỉnh thêm: Sau** khi huấn luyện lại, đánh giá hiệu suất của mô hình trên tập kiểm tra và thực hiện các tinh chỉnh cần thiết để tối ưu hóa hiệu suất.

2.11.3.2. Lợi ích của Fine-Tuning trong OCR

Fine-Tuning mang lại nhiều lợi ích khi áp dụng cho các nhiệm vụ OCR cụ thể:

- **Hiệu suất cao hơn:** Bằng cách tinh chỉnh các lớp cuối cùng của mô hình pretrained, bạn có thể đạt được hiệu suất cao hơn trên nhiệm vụ OCR cụ thể so với việc huấn luyện mô hình từ đầu.
- **Thời gian huấn luyện ngắn hơn:** Fine-Tuning giúp giảm đáng kể thời gian huấn luyện so với việc huấn luyện toàn bộ mô hình từ đầu, đặc biệt khi sử dụng các mô hình lớn như ResNet hoặc Transformer.
- **Tận dụng dữ liệu có hạn:** Khi dữ liệu OCR cụ thể hạn chế, Fine-Tuning giúp tối đa hóa hiệu suất bằng cách tận dụng các đặc trưng đã học từ tập dữ liệu lớn và đa dạng hơn.
- **Ứng dụng linh hoạt:** Fine-Tuning cho phép áp dụng các mô hình pretrained vào nhiều loại dữ liệu và nhiệm vụ OCR khác nhau, từ nhận diện ký tự đơn giản đến nhận diện văn bản trong các hình ảnh phức tạp.

2.11.4. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về Transfer Learning trong OCR, bao gồm khái niệm, lợi ích, các mô hình pretrained phổ biến, và cách tinh chỉnh mô hình cho các nhiệm vụ OCR cụ thể. Transfer Learning là một phương pháp mạnh mẽ và linh hoạt, giúp tiết kiệm thời gian và tài nguyên, đồng thời cải thiện hiệu suất của các mô hình OCR trong các ứng dụng thực tế. Hiểu rõ và áp dụng đúng các kỹ thuật Transfer Learning sẽ giúp bạn xây dựng các hệ thống OCR hiệu quả hơn và nhanh chóng đạt được kết quả tốt.

2.12. Preprocessing và xử lý trước dữ liệu trong OCR

Preprocessing (Xử lý trước dữ liệu) là một bước quan trọng trong quy trình OCR (Optical Character Recognition). Chất lượng của dữ liệu đầu vào có ảnh hưởng lớn đến hiệu suất của mô hình OCR. Do đó, việc áp dụng các kỹ thuật preprocessing để làm sạch và chuẩn bị dữ liệu trước khi đưa vào mô hình là cần thiết để đạt được độ chính xác cao trong nhận diện văn bản.

2.12.1. Các kỹ thuật làm sạch dữ liệu ảnh

2.12.1.1. Noise Reduction (giảm nhiễu)

Noise Reduction là quá trình loại bỏ các nhiễu (noise) không mong muốn trong hình ảnh, giúp làm sạch dữ liệu và cải thiện độ rõ ràng của văn bản. Noise có thể xuất hiện do nhiều nguyên nhân, chẳng hạn như chất lượng in ấn kém, ánh sáng không đều, hoặc quá trình số hóa hình ảnh.

- **Gaussian Blur:** Gaussian Blur là một trong những kỹ thuật giảm nhiễu phổ biến, sử dụng một bộ lọc Gaussian để làm mờ hình ảnh. Bộ lọc này giúp loại bỏ các điểm nhiễu nhỏ mà không làm mất đi các chi tiết quan trọng trong văn bản.
- **Median Filtering:** Median Filtering là một kỹ thuật giảm nhiễu khác, hoạt động bằng cách thay thế mỗi pixel trong hình ảnh bằng giá trị trung vị của các pixel lân cận. Phương pháp này hiệu quả trong việc loại bỏ nhiễu muối tiêu (salt-and-pepper noise), thường xuất hiện trong các hình ảnh kém chất lượng.
- **Bilateral Filtering:** Bilateral Filtering là một kỹ thuật nâng cao hơn, kết hợp giữa Gaussian Blur và Median Filtering. Phương pháp này giữ lại các cạnh rõ nét trong hình ảnh trong khi vẫn loại bỏ được nhiễu, giúp duy trì độ rõ ràng của văn bản.

2.12.1.2. Binarization (Nhị phân hóa)

Binarization là quá trình chuyển đổi hình ảnh từ dạng xám (grayscale) sang dạng nhị phân (binary), nơi mà các pixel chỉ nhận một trong hai giá trị: đen hoặc trắng. Binarization giúp làm nổi bật văn bản trong hình ảnh, loại bỏ các yếu tố gây nhiễu và tăng độ tương phản giữa văn bản và nền.

- **Thresholding:** Thresholding là kỹ thuật cơ bản nhất trong binarization, hoạt động bằng cách gán một ngưỡng (threshold) cố định để phân loại các pixel trong hình ảnh. Các pixel có giá trị lớn hơn ngưỡng sẽ được gán là trắng, và các pixel có giá trị nhỏ hơn ngưỡng sẽ được gán là đen.
- **Otsu's Method:** Otsu's Method là một kỹ thuật thresholding nâng cao, tự động xác định ngưỡng tối ưu bằng cách giảm thiểu phương sai trong nội bộ

các lớp được phân loại. Phương pháp này đặc biệt hiệu quả khi có sự thay đổi độ sáng trong hình ảnh.

- **Adaptive Thresholding:** Adaptive Thresholding sử dụng một ngưỡng khác nhau cho từng vùng nhỏ của hình ảnh, thay vì áp dụng một ngưỡng cố định cho toàn bộ hình ảnh. Phương pháp này giúp xử lý tốt hơn các hình ảnh có độ sáng không đồng đều hoặc có các vùng có mức độ ánh sáng khác nhau.

2.12.2. Vai trò của Preprocessing trong cải thiện hiệu suất ocr

Preprocessing đóng vai trò quan trọng trong việc cải thiện hiệu suất của các hệ thống OCR. Dưới đây là các lý do chính giải thích vì sao preprocessing là cần thiết:

2.12.2.1. Tăng độ chính xác của mô hình

Một trong những mục tiêu chính của preprocessing là cải thiện độ chính xác của mô hình OCR. Các kỹ thuật như giảm nhiễu và nhị phân hóa giúp làm sạch hình ảnh đầu vào, từ đó giảm thiểu các lỗi nhận diện do nhiễu hoặc độ tương phản kém. Hình ảnh sạch và rõ ràng giúp mô hình OCR nhận diện văn bản một cách chính xác hơn.

- **Loại bỏ nhiễu:** Noise có thể gây ra nhiều lỗi trong quá trình nhận diện, đặc biệt khi các ký tự bị biến dạng hoặc bị che khuất bởi các điểm nhiễu. Bằng cách áp dụng các kỹ thuật giảm nhiễu, hình ảnh đầu vào trở nên rõ ràng hơn, giúp mô hình OCR nhận diện các ký tự một cách chính xác.
- **Tăng độ tương phản:** Nhị phân hóa giúp tăng độ tương phản giữa văn bản và nền, làm cho văn bản nổi bật hơn trong hình ảnh. Điều này giúp các mô hình OCR dễ dàng tách biệt và nhận diện các ký tự trong văn bản, đặc biệt trong các hình ảnh có nền phức tạp.

2.12.2.2. Giảm thiểu Overfitting

Overfitting xảy ra khi mô hình học quá mức các đặc trưng cụ thể của dữ liệu huấn luyện và không thể tổng quát hóa tốt trên dữ liệu mới. Preprocessing giúp giảm thiểu overfitting bằng cách loại bỏ các yếu tố gây nhiễu không cần thiết, giúp mô hình tập trung vào các đặc trưng quan trọng thực sự của văn bản.

- **Loại bỏ các chi tiết không quan trọng:** Preprocessing giúp loại bỏ các chi tiết không quan trọng hoặc gây nhiễu trong hình ảnh, từ đó giúp mô hình

OCR tập trung vào việc nhận diện các ký tự một cách chính xác hơn mà không bị phân tâm bởi các yếu tố không liên quan.

- **Tăng độ đa dạng của dữ liệu:** Các kỹ thuật preprocessing như augmentation có thể được sử dụng để tạo ra các biến thể của dữ liệu huấn luyện, giúp mô hình học cách nhận diện văn bản trong các điều kiện khác nhau. Điều này giúp giảm thiểu overfitting và cải thiện khả năng tổng quát hóa của mô hình.

2.12.2.3. Cải thiện tốc độ xử lý

Preprocessing không chỉ giúp cải thiện độ chính xác mà còn giúp tăng tốc độ xử lý của các hệ thống OCR. Bằng cách chuẩn hóa và làm sạch dữ liệu đầu vào, mô hình OCR có thể xử lý dữ liệu nhanh hơn và hiệu quả hơn.

- **Giảm kích thước dữ liệu:** Các kỹ thuật như giảm nhiễu và nhị phân hóa giúp giảm kích thước dữ liệu đầu vào bằng cách loại bỏ các yếu tố không cần thiết. Điều này giúp mô hình OCR xử lý dữ liệu nhanh hơn mà không làm giảm độ chính xác.
- **Tối ưu hóa dữ liệu đầu vào:** Bằng cách chuẩn hóa dữ liệu đầu vào (chẳng hạn như điều chỉnh kích thước hình ảnh hoặc cân bằng màu sắc), quá trình xử lý của mô hình OCR trở nên ổn định và nhất quán hơn, từ đó cải thiện hiệu suất tổng thể.

2.12.3. Phân Tích Các Phương Pháp Xử Lý Dữ Liệu Tiên Tiến

Bên cạnh các kỹ thuật preprocessing truyền thống, có nhiều phương pháp xử lý dữ liệu tiên tiến đã được phát triển để đáp ứng các nhu cầu ngày càng phức tạp trong OCR. Dưới đây là một số phương pháp tiên tiến đáng chú ý:

2.12.3.1. Deep Learning-Based Preprocessing

Deep Learning-Based Preprocessing sử dụng các mô hình học sâu để tự động học cách xử lý và cải thiện chất lượng của hình ảnh trước khi đưa vào hệ thống OCR. Các mô hình này có thể học cách loại bỏ nhiễu, cải thiện độ phân giải, và tăng độ tương phản của hình ảnh một cách tự động mà không cần phải xác định các tham số cụ thể.

- **Super-Resolution:** Super-Resolution là một kỹ thuật học sâu được sử dụng để cải thiện độ phân giải của hình ảnh. Bằng cách tăng cường độ phân giải

của hình ảnh, Super-Resolution giúp các hệ thống OCR nhận diện chính xác hơn các ký tự nhỏ hoặc mờ.

- **Denoising Autoencoders:** Denoising Autoencoders là một loại mạng nơ-ron tự mã hóa (autoencoder) được sử dụng để loại bỏ nhiễu khỏi hình ảnh. Autoencoder học cách tái tạo hình ảnh sạch từ các hình ảnh nhiễu, giúp cải thiện chất lượng đầu vào cho hệ thống OCR.
- **Generative Adversarial Networks (GANs):** GANs có thể được sử dụng để tạo ra các phiên bản cải thiện của hình ảnh đầu vào, chẳng hạn như loại bỏ nhiễu hoặc cải thiện độ sáng. Mô hình GAN học cách tạo ra hình ảnh có chất lượng cao từ hình ảnh đầu vào có chất lượng thấp, giúp cải thiện hiệu suất của OCR.

2.12.3.2. Adaptive Preprocessing

Adaptive Preprocessing là một phương pháp tiên tiến trong đó các kỹ thuật preprocessing được điều chỉnh linh hoạt dựa trên các đặc điểm cụ thể của từng hình ảnh. Thay vì áp dụng cùng một bộ kỹ thuật preprocessing cho tất cả các hình ảnh, Adaptive Preprocessing sử dụng các thuật toán để xác định kỹ thuật nào là phù hợp nhất cho từng hình ảnh cụ thể.

- **Adaptive Binarization:** Adaptive Binarization tự động chọn ngưỡng nhị phân hóa dựa trên độ sáng và độ tương phản của từng vùng trong hình ảnh. Điều này giúp xử lý tốt hơn các hình ảnh có độ sáng không đồng đều hoặc chứa các vùng có độ sáng khác nhau.
- **Context-Aware Preprocessing:** Context-Aware Preprocessing sử dụng các thông tin ngữ cảnh từ hình ảnh để điều chỉnh quá trình xử lý. Ví dụ, nếu hình ảnh chứa các ký tự nhỏ hoặc văn bản nằm trên nền phức tạp, thuật toán có thể tăng cường độ tương phản hoặc áp dụng các kỹ thuật làm sạch nâng cao để cải thiện chất lượng văn bản.

2.12.4. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về Preprocessing và Xử lý Trước Dữ liệu trong OCR, bao gồm các kỹ thuật làm sạch dữ liệu ảnh như Noise Reduction và Binarization, vai trò của preprocessing trong cải thiện hiệu suất OCR, và phân tích các phương pháp xử lý dữ liệu tiên tiến. Preprocessing là một

bước không thể thiếu trong quy trình OCR, giúp cải thiện độ chính xác, giảm thiểu overfitting, và tăng tốc độ xử lý của mô hình. Hiểu rõ và áp dụng đúng các kỹ thuật preprocessing sẽ giúp bạn xây dựng các hệ thống OCR hiệu quả và đạt được kết quả tốt hơn trong các ứng dụng thực tế.

2.13. Nghiên cứu về các dữ liệu OCR tiêu chuẩn

Trong lĩnh vực OCR (Optical Character Recognition), việc sử dụng các bộ dữ liệu tiêu chuẩn đóng vai trò quan trọng trong việc phát triển và đánh giá các mô hình nhận diện văn bản. Các bộ dữ liệu này cung cấp một nền tảng chung để so sáng hiệu suất của các thuật toán và mô hình khác nhau. Dưới dây, chúng ta sẽ tìm hiểu về một số bộ dữ liệu tiêu chuẩn phổ biến trong OCR, phân tích và so sánh chúng, cũng như thảo luận về cách lựa chọn và sử dụng bổ dữ liệu phù hợp cho các tác vụ OCR.

2.13.1. Các bộ dữ liệu chuẩn cho OCR

2.13.1.1. MNIST (Modified National Institute of Standards and Technology)

MNIST là một trong những bộ dữ liệu phổ biến nhất và được sử dụng rộng rãi trong lĩnh vực học sâu, đặc biệt là trong các bài toán phân loại chữ số viết tay. Bộ dữ liệu này bao gồm 70,000 hình ảnh chữ số viết tay từ 0 đến 9, được chia thành 60,000 hình ảnh cho tập huấn luyện và 10,000 hình ảnh cho tập kiểm tra.

- **Đặc điểm:** Mỗi hình ảnh trong MNIST có kích thước 28x28 pixel, và mỗi pixel chứa một giá trị từ 0 đến 255 đại diện cho cường độ màu xám. Các chữ số trong MNIST đã được chuẩn hóa về kích thước và đặt ở trung tâm của mỗi hình ảnh.
- **Ứng dụng:** MNIST thường được sử dụng làm bài toán "Hello World" cho các mô hình học sâu, nơi các nhà nghiên cứu và kỹ sư kiểm tra và đánh giá các thuật toán mới. Nó cũng được sử dụng để kiểm tra khả năng của các mô hình trong việc nhận diện các ký tự đơn giản trước khi chuyển sang các bài toán phức tạp hơn.

2.13.1.2. Chars74K

Chars74K là một bộ dữ liệu lớn và đa dạng, được thiết kế để nhận diện ký tự và chữ cái viết tay trong nhiều ngữ cảnh khác nhau. Bộ dữ liệu này bao gồm hơn

74,000 hình ảnh của các ký tự từ nhiều ngôn ngữ, bao gồm cả chữ cái Latin và các chữ cái từ các ngôn ngữ khác như Kannada và Tamil.

- **Đặc điểm:** Chars74K bao gồm các hình ảnh của các ký tự được viết tay, in ấn, và được chụp từ nhiều ngữ cảnh khác nhau. Bộ dữ liệu này cung cấp một thách thức lớn hơn so với MNIST, vì nó bao gồm nhiều loại ký tự và phong cách viết khác nhau, cũng như các biến thể về kích thước, độ nghiêng, và độ phân giải.
- **Ứng dụng:** Chars74K được sử dụng để phát triển và đánh giá các mô hình OCR có khả năng nhận diện ký tự trong nhiều điều kiện khác nhau, bao gồm cả văn bản viết tay và văn bản in từ nhiều ngôn ngữ. Bộ dữ liệu này đặc biệt hữu ích cho các nghiên cứu liên quan đến nhận diện đa ngôn ngữ.

2.13.1.3. IIIT5K

IIIT5K (Indian Institute of Information Technology, 5,000 Words Dataset) là một bộ dữ liệu dành riêng cho nhận diện từ ngữ trong hình ảnh. Bộ dữ liệu này bao gồm 5,000 hình ảnh chứa các từ ngữ từ nhiều ngữ cảnh khác nhau, chẳng hạn như biển báo, tài liệu, và văn bản từ hình ảnh thực tế.

- **Đặc điểm:** Các hình ảnh trong IIIT5K có độ phân giải và chất lượng đa dạng, với các từ ngữ xuất hiện trong nhiều phong cách và phông chữ khác nhau. Bộ dữ liệu này không chỉ thách thức các mô hình OCR trong việc nhận diện các ký tự đơn lẻ, mà còn trong việc hiểu và giải mã các từ ngữ hoàn chỉnh.
- **Ứng dụng:** IIIT5K được sử dụng để phát triển các mô hình OCR có khả năng nhận diện và hiểu các từ ngữ trong nhiều ngữ cảnh khác nhau, bao gồm cả các từ ngữ từ các biển báo đường phố và văn bản từ các tài liệu không chuẩn.

2.13.2. Phân tích và so sánh các bộ dữ liệu

2.13.2.1. MNIST vs. Chars74K

MNIST là một bộ dữ liệu lý tưởng cho những người mới bắt đầu với OCR, nhờ vào sự đơn giản và chuẩn hóa của nó. Các hình ảnh trong MNIST có kích thước nhỏ và ít biến thể, giúp các mô hình học sâu dễ dàng nhận diện và đạt được độ chính xác cao. Tuy nhiên, hạn chế của MNIST là nó chỉ bao gồm các chữ số đơn giản, không phản ánh đầy đủ các thách thức thực tế trong OCR.

Chars74K ngược lại, là một bộ dữ liệu phức tạp hơn, với nhiều biến thể về ngôn ngữ, phong cách viết, và ngữ cảnh. Chars74K yêu cầu các mô hình OCR phải có khả năng xử lý các biến thể phức tạp trong văn bản, làm cho nó trở thành một bộ dữ liệu lý tưởng cho các nghiên cứu OCR tiên tiến. Tuy nhiên, việc sử dụng Chars74K đòi hỏi các mô hình phức tạp hơn và có thể gặp khó khăn hơn trong việc đạt được độ chính xác cao.

2.13.2.2. Chars74K vs. IIIT5K

Chars74K và IIIT5K đều là những bộ dữ liệu phức tạp, nhưng chúng tập trung vào các khía cạnh khác nhau của OCR. Trong khi Chars74K tập trung vào nhận diện ký tự từ nhiều ngôn ngữ khác nhau, IIIT5K tập trung vào nhận diện từ ngữ trong các ngữ cảnh thực tế.

Chars74K phù hợp cho các nghiên cứu đa ngôn ngữ, nơi mà khả năng nhận diện ký tự từ nhiều hệ chữ khác nhau là quan trọng. Trong khi đó, IIIT5K cung cấp một thách thức về việc nhận diện và hiểu các từ ngữ trong các hình ảnh thực tế, giúp phát triển các mô hình OCR có khả năng xử lý văn bản trong nhiều ngữ cảnh và điều kiện khác nhau.

2.13.2.3. MNIST vs. IIIT5K

MNIST và IIIT5K đại diện cho hai cấp độ khác nhau của OCR. Trong khi MNIST cung cấp một môi trường đơn giản để thử nghiệm các mô hình học sâu với các ký tự đơn giản, IIIT5K đặt ra một thách thức lớn hơn với các từ ngữ trong ngữ cảnh thực tế.

MNIST phù hợp cho việc xây dựng và kiểm thử các mô hình cơ bản, trong khi IIIT5K yêu cầu các mô hình phải phức tạp và tinh vi hơn để xử lý các từ ngữ trong các điều kiện khác nhau. Sử dụng IIIT5K có thể giúp đánh giá khả năng tổng quát hóa của mô hình OCR trong các bài toán thực tế.

2.13.3. Cách lựa chọn và sử dụng bộ dữ liệu phù hợp cho ocr

2.13.3.1. Xác định mục tiêu nghiên cứu

Trước khi lựa chọn một bộ dữ liệu OCR, điều quan trọng là phải xác định rõ mục tiêu của nghiên cứu. Bạn cần xác định xem liệu mô hình của bạn sẽ được sử dụng để nhận diện các ký tự đơn giản như chữ số, hay để nhận diện các từ ngữ phức tạp trong các ngữ cảnh thực tế.

- **Nếu mục tiêu là nhận diện chữ số:** MNIST là lựa chọn lý tưởng để bắt đầu. Nó cung cấp một môi trường dễ dàng và chuẩn hóa để thử nghiệm các mô hình học sâu và hiểu rõ cách thức chúng hoạt động.
- **Nếu mục tiêu là nhận diện ký tự đa ngôn ngữ:** Chars74K là một bộ dữ liệu thích hợp, giúp phát triển các mô hình có khả năng nhận diện ký tự từ nhiều ngôn ngữ khác nhau và trong nhiều ngữ cảnh khác nhau.
- **Nếu mục tiêu là nhận diện văn bản trong ngữ cảnh thực tế:** IIIT5K là lựa chọn phù hợp, giúp đánh giá khả năng của mô hình trong việc nhận diện và hiểu các từ ngữ trong các hình ảnh thực tế.

2.13.3.2. Chuẩn bị và xử lý dữ liệu

Sau khi lựa chọn bộ dữ liệu phù hợp, bước tiếp theo là chuẩn bị và xử lý dữ liệu. Điều này bao gồm các bước như preprocessing (xử lý trước dữ liệu), augmentation (tăng cường dữ liệu), và phân chia dữ liệu thành các tập huấn luyện, kiểm tra, và kiểm định.

- **Preprocessing:** Áp dụng các kỹ thuật xử lý trước dữ liệu như giảm nhiễu, nhị phân hóa, và chuẩn hóa kích thước hình ảnh để đảm bảo rằng dữ liệu đầu vào sạch và nhất quán.
- **Augmentation:** Sử dụng kỹ thuật augmentation để tạo ra các biến thể của dữ liệu huấn luyện, giúp mô hình OCR học cách nhận diện văn bản trong nhiều điều kiện khác nhau.
- **Phân chia dữ liệu:** Đảm bảo rằng dữ liệu được phân chia hợp lý thành các tập huấn luyện, kiểm tra, và kiểm định để đánh giá hiệu suất của mô hình một cách khách quan.

2.13.3.3. Đánh giá hiệu suất mô hình

Cuối cùng, sau khi huấn luyện mô hình OCR trên bộ dữ liệu đã chọn, bạn cần đánh giá hiệu suất của mô hình. Điều này bao gồm việc so sánh kết quả với các mô hình khác và đánh giá khả năng tổng quát hóa của mô hình trên dữ liệu mới.

- **Sử dụng bộ dữ liệu kiểm tra:** Đánh giá mô hình trên tập kiểm tra để đo lường độ chính xác và hiệu suất tổng thể của mô hình.

- **So sánh với các mô hình khác:** So sánh kết quả của mô hình với các mô hình OCR khác trên cùng một bộ dữ liệu để đánh giá hiệu suất tương đối.
- **Kiểm tra khả năng tổng quát hóa:** Sử dụng dữ liệu mới hoặc dữ liệu từ các bộ dữ liệu khác để kiểm tra khả năng tổng quát hóa của mô hình, đảm bảo rằng mô hình có thể hoạt động tốt trong nhiều ngữ cảnh khác nhau.

2.13.4. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về các bộ dữ liệu tiêu chuẩn cho OCR, phân tích và so sánh chúng, và thảo luận về cách lựa chọn và sử dụng bộ dữ liệu phù hợp cho các tác vụ OCR. Sự lựa chọn đúng đắn và sử dụng hiệu quả các bộ dữ liệu tiêu chuẩn là yếu tố quan trọng để phát triển các mô hình OCR mạnh mẽ và đạt được hiệu suất cao trong các ứng dụng thực tế.

2.14. Ứng dụng của OCR trong các lĩnh vực đặc thù.

OCR (Optical Character Recognition) không chỉ đơn thuần là công nghệ nhận diện văn bản từ hình ảnh mà đã trở thành một công cụ quan trọng trong nhiều lĩnh vực khách nhau. Dưới đây là một cái nhìn sâu hơn về cách OCR đang được ứng dụng trong các lĩnh vực y tế, an ninh và tài chính.

2.14.1. OCR trong lĩnh vực y tế: số hóa hồ sơ bệnh án, đơn thuốc

2.14.1.1. Số hóa hồ sơ bệnh án

Trong y tế, việc quản lý và lưu trữ hồ sơ bệnh án là một thách thức lớn do khối lượng dữ liệu khổng lồ. OCR đã mang lại một giải pháp hiệu quả bằng cách số hóa hồ sơ bệnh án, giúp việc truy cập, lưu trữ và quản lý thông tin bệnh nhân trở nên dễ dàng và nhanh chóng hơn.

- **Tiết kiệm thời gian:** Trước đây, việc tìm kiếm và xử lý thông tin từ hồ sơ bệnh án thường tốn nhiều thời gian. Với OCR, các tài liệu giấy có thể được quét và chuyển đổi thành dạng số, giúp bác sĩ và nhân viên y tế truy cập thông tin một cách nhanh chóng.
- **Cải thiện độ chính xác:** Hồ sơ bệnh án số hóa giúp giảm thiểu lỗi do nhập liệu thủ công, đảm bảo rằng các thông tin quan trọng như chẩn đoán và lịch sử điều trị được ghi nhận chính xác.

- **Bảo mật tốt hơn:** OCR kết hợp với các công nghệ bảo mật hiện đại giúp bảo vệ thông tin y tế nhạy cảm, ngăn chặn truy cập trái phép và đảm bảo rằng chỉ những người có quyền mới có thể truy cập dữ liệu.

2.14.1.2. Nhận diện đơn thuốc

OCR cũng được sử dụng để tự động hóa quá trình nhận diện và xử lý đơn thuốc, giúp các nhà thuốc và bệnh viện cải thiện hiệu suất và giảm thiểu sai sót.

- **Xử lý đơn thuốc nhanh chóng:** OCR có thể tự động nhận diện và chuyển đổi các đơn thuốc viết tay hoặc in ấn thành văn bản số, giúp nhà thuốc và bác sĩ dễ dàng kiểm tra và xác nhận thông tin thuốc.
- **Giảm thiểu sai sót:** Sai sót trong việc đọc và xử lý đơn thuốc có thể gây ra hậu quả nghiêm trọng. OCR giúp giảm thiểu những sai sót này bằng cách tự động nhận diện và đối chiếu thông tin thuốc với cơ sở dữ liệu.
- **Tích hợp với hệ thống quản lý y tế:** OCR giúp đơn thuốc được lưu trữ và quản lý hiệu quả trong hệ thống thông tin y tế, từ đó giúp bác sĩ và nhân viên y tế dễ dàng theo dõi lịch sử kê đơn và điều trị của bệnh nhân.

2.14.2. OCR trong an ninh: nhận diện biển số xe, giấy tờ tùy thân

2.14.2.1. Nhận diện biển số xe

OCR đã trở thành một công cụ đắc lực trong lĩnh vực an ninh, đặc biệt là trong việc nhận diện biển số xe, hỗ trợ cho các hệ thống giám sát giao thông và quản lý an ninh.

- **Giám sát giao thông:** Hệ thống nhận diện biển số xe (Automatic License Plate Recognition - ALPR) sử dụng OCR để tự động nhận diện biển số từ các camera giám sát giao thông, giúp phát hiện các vi phạm như vượt đèn đỏ, chạy quá tốc độ, hoặc xe bị đánh cắp.
- **Quản lý bãi đỗ xe:** OCR giúp tự động hóa việc quản lý bãi đỗ xe bằng cách nhận diện và ghi lại biển số xe khi vào và ra khỏi bãi, từ đó giảm thiểu thời gian chờ và tăng cường an ninh.
- **Hỗ trợ điều tra:** OCR giúp cảnh sát và các cơ quan an ninh nhanh chóng tìm kiếm và xác định các xe liên quan đến các vụ án bằng cách quét và so sánh biển số từ camera với cơ sở dữ liệu.

2.14.2.2. Nhận diện giấy tờ tùy thân

OCR cũng được áp dụng rộng rãi trong việc nhận diện và xử lý thông tin từ các giấy tờ tùy thân như chứng minh nhân dân, hộ chiếu, và bằng lái xe.

- **Xác minh danh tính:** OCR giúp tự động nhận diện và xác minh thông tin cá nhân từ giấy tờ tùy thân, từ đó hỗ trợ các quy trình đăng ký, xác minh danh tính trong ngân hàng, sân bay, và các dịch vụ công.
- **Tự động hóa quy trình:** OCR giúp tự động hóa quy trình nhập liệu từ giấy tờ tùy thân vào các hệ thống quản lý, giảm thiểu lỗi do nhập liệu thủ công và tăng cường hiệu suất làm việc.
- **Cải thiện bảo mật:** OCR kết hợp với các công nghệ nhận diện khuôn mặt và vân tay để tạo ra các hệ thống xác minh danh tính chính xác và bảo mật hơn, giúp ngăn chặn các hành vi gian lận và giả mạo giấy tờ.

2.14.3. OCR trong tài chính: số hóa tài liệu, nhận diện chữ ký

2.14.3.1. Số hóa tài liệu

Trong lĩnh vực tài chính, việc quản lý và xử lý tài liệu là một phần quan trọng của hoạt động hàng ngày. OCR đã giúp các tổ chức tài chính số hóa tài liệu, từ đó cải thiện hiệu quả làm việc và giảm thiểu rủi ro.

- **Số hóa hóa đơn và Chứng từ:** OCR giúp tự động quét và số hóa các hóa đơn, chứng từ tài chính, giúp việc lưu trữ và tìm kiếm thông tin trở nên dễ dàng hơn. Điều này cũng giúp giảm thiểu rủi ro mất mát hoặc hư hỏng tài liệu giấy.
- **Xử lý tự động:** Các hệ thống tài chính sử dụng OCR để tự động xử lý các tài liệu như đơn vay, hợp đồng, và sao kê ngân hàng, từ đó giảm thiểu lỗi do nhập liệu thủ công và tăng cường độ chính xác.
- **Hỗ trợ tuân thủ quy định:** OCR giúp các tổ chức tài chính tuân thủ các quy định về lưu trữ và quản lý dữ liệu, đảm bảo rằng tất cả các tài liệu đều được lưu trữ đúng cách và có thể truy cập khi cần thiết.

2.14.3.2. Nhận diện chữ ký

OCR cũng được sử dụng để nhận diện và xác minh chữ ký, một yếu tố quan trọng trong các giao dịch tài chính.

- **Xác minh chữ ký:** OCR giúp tự động nhận diện và so sánh chữ ký từ các tài liệu tài chính với mẫu chữ ký đã được lưu trữ, từ đó hỗ trợ quá trình xác minh danh tính trong các giao dịch quan trọng.
- **Chữ ký điện tử:** Với sự phát triển của chữ ký điện tử, OCR đóng vai trò quan trọng trong việc xác minh và quản lý chữ ký điện tử, đảm bảo rằng các giao dịch tài chính được thực hiện an toàn và hợp pháp.
- **Giảm thiểu gian lận:** OCR giúp ngăn chặn các hành vi gian lận chữ ký bằng cách tự động phát hiện các dấu hiệu bất thường hoặc giả mạo trong chữ ký trên các tài liệu tài chính.

2.14.4. Kết luận

OCR đã chứng minh được giá trị của mình trong nhiều lĩnh vực đặc thù, từ y tế, an ninh đến tài chính. Việc số hóa và tự động hóa quy trình bằng OCR không chỉ giúp tăng cường hiệu suất làm việc mà còn cải thiện độ chính xác, giảm thiểu rủi ro và đảm bảo an ninh. Với sự phát triển không ngừng của công nghệ, OCR tiếp tục mở rộng ứng dụng của mình, đóng góp tích cực vào nhiều khía cạnh của cuộc sống hiện đại.

2.15. Các hướng phát triển mới trong OCR.

Công nghệ OCR đã trải qua nhiều giai đoạn phát triển và vẫn đang tiếp tục tiến hóa. Với sự xuất hiện của các công nghệ mới như Quantum Computing, AI và IoT. OCR hứa hẹn sẽ có những bước tiến lớn trong tương lai. Dưới đây là một số xu hướng và hướng phát triển mới trong lĩnh vực OCR.

2.15.1. Xu hướng phát t riển các mô hình OCR trong tương lai

2.15.1.1. Deep Learning và OCR

Deep Learning đã và đang là một yếu tố chủ chốt trong sự phát triển của OCR. Các mô hình dựa trên CNN (Convolutional Neural Networks), RNN (Recurrent Neural Networks), và Transformer đã cải thiện đáng kể hiệu suất của OCR, đặc biệt trong các nhiệm vụ phức tạp như nhận diện văn bản viết tay hoặc văn bản từ các hình ảnh chất lượng thấp.

- **Mô hình Hybrid:** Sự kết hợp giữa các mô hình CNN và RNN trong các mô hình OCR đã giúp cải thiện khả năng nhận diện văn bản liên tục và tuần tự.

Trong tương lai, các mô hình hybrid này có thể tiếp tục phát triển để xử lý các ngữ cảnh phức tạp hơn.

- **Transformer và Vision Transformer:** Các mô hình Transformer đã chứng minh hiệu quả trong nhiều lĩnh vực, bao gồm cả OCR. Vision Transformer (ViT) là một hướng phát triển mới, kết hợp giữa Transformer và CNN, hứa hẹn mang lại những cải tiến vượt bậc trong OCR, đặc biệt là trong việc nhận diện văn bản từ hình ảnh có độ phân giải cao hoặc phức tạp.
- **Self-Supervised Learning:** Học sâu không giám sát (Self-Supervised Learning) là một xu hướng mới, giúp các mô hình OCR tự học từ dữ liệu mà không cần gán nhãn. Điều này mở ra cơ hội lớn cho OCR trong các ứng dụng yêu cầu xử lý lượng dữ liệu lớn mà không cần đầu tư nhiều vào việc gán nhãn.

2.15.1.2. OCR đa ngôn ngữ

Với sự phát triển của các mô hình đa ngôn ngữ như mBERT (Multilingual BERT) và các biến thể của nó, OCR đang mở rộng khả năng nhận diện và xử lý văn bản từ nhiều ngôn ngữ khác nhau. Điều này đặc biệt quan trọng trong các ứng dụng toàn cầu, nơi mà việc nhận diện và xử lý văn bản từ nhiều ngôn ngữ khác nhau là cần thiết.

- **OCR cho ngôn ngữ thiểu số:** Sự phát triển của OCR đa ngôn ngữ cũng tạo điều kiện cho việc nghiên cứu và phát triển các mô hình OCR cho các ngôn ngữ thiểu số, giúp bảo tồn và số hóa các tài liệu văn hóa và lịch sử.
- **Tích hợp ngôn ngữ và ngữ nghĩa:** Các mô hình OCR trong tương lai có thể không chỉ nhận diện văn bản mà còn hiểu được ngữ nghĩa và ngữ cảnh của văn bản đó, giúp cải thiện khả năng dịch và xử lý ngôn ngữ tự nhiên (NLP).

2.15.2. Ứng dụng của công nghệ mới như Quantum Computing trong OCR

Quantum Computing là một công nghệ đang nổi lên, hứa hẹn mang lại những cải tiến lớn cho nhiều lĩnh vực, bao gồm cả OCR.

- **Tăng tốc độ xử lý:** Quantum Computing có khả năng xử lý dữ liệu với tốc độ vượt trội so với các hệ thống máy tính truyền thống. Điều này có thể giúp

cải thiện đáng kể tốc độ xử lý của các hệ thống OCR, đặc biệt là trong các ứng dụng yêu cầu xử lý dữ liệu lớn và phức tạp.

- **Tối ưu hóa thuật toán:** Các thuật toán OCR có thể được tối ưu hóa bằng cách sử dụng các thuật toán lượng tử, giúp cải thiện độ chính xác và hiệu suất của OCR trong các bài toán khó như nhận diện văn bản từ hình ảnh chất lượng thấp hoặc văn bản bị che khuất.
- **Khả năng học tập nâng cao:** Quantum Computing có thể mở ra khả năng học tập nâng cao cho các mô hình OCR, giúp chúng học hỏi và thích nghi nhanh chóng hơn với các ngữ cảnh mới và dữ liệu mới.

2.15.3. Tích hợp OCR với các công nghệ khác (như AI, IoT) để mở rộng ứng dụng

Sự tích hợp OCR với các công nghệ khác như AI và IoT đang mở ra những cơ hội mới cho việc mở rộng ứng dụng của OCR trong nhiều lĩnh vực khác nhau.

2.15.3.1. AI Và OCR

AI không chỉ giúp cải thiện hiệu suất của OCR mà còn mở rộng phạm vi ứng dụng của nó. Việc tích hợp AI vào OCR giúp các hệ thống có thể học hỏi từ dữ liệu, cải thiện độ chính xác theo thời gian và tự động hóa các quy trình phức tạp.

- **Tự động hóa quy trình:** AI kết hợp với OCR có thể tự động hóa các quy trình xử lý tài liệu, từ nhận diện văn bản đến phân loại và lưu trữ, giúp tiết kiệm thời gian và chi phí cho doanh nghiệp.
- **Phân tích dữ liệu:** OCR kết hợp với AI có thể tự động phân tích và trích xuất thông tin từ các tài liệu, giúp doanh nghiệp nhanh chóng đưa ra quyết định dựa trên dữ liệu.

2.15.3.2. IoT và OCR

IoT (Internet of Things) đang kết nối hàng tỷ thiết bị trên toàn thế giới, và OCR có thể đóng vai trò quan trọng trong việc thu thập và xử lý dữ liệu từ các thiết bị này.

- **Giám sát thông minh:** Các thiết bị IoT có thể sử dụng OCR để nhận diện và xử lý thông tin từ môi trường xung quanh, chẳng hạn như nhận diện biển số

xe, đọc thông tin từ đồng hồ đo điện, hoặc nhận diện nhãn sản phẩm trong kho hàng.

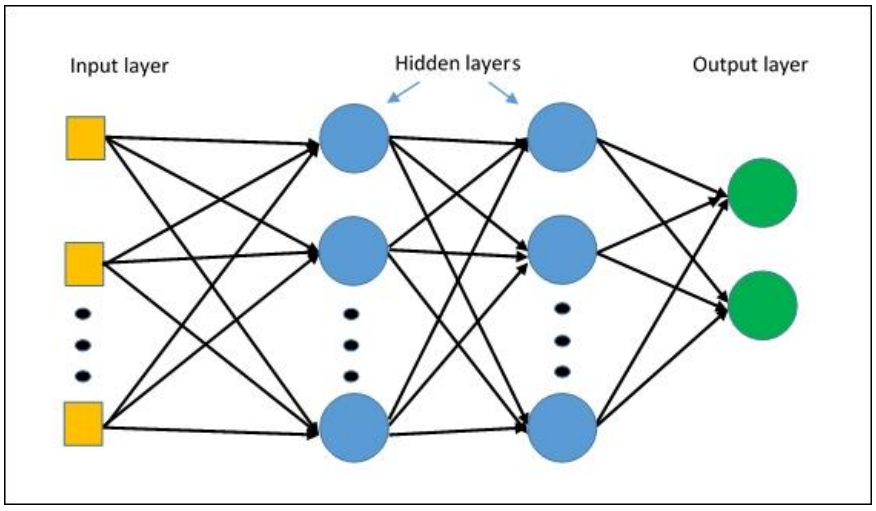
- **Tích hợp dữ liệu:** OCR có thể giúp tích hợp và chuẩn hóa dữ liệu từ các thiết bị IoT, từ đó giúp tạo ra các hệ thống giám sát và quản lý thông minh, đáp ứng nhu cầu của các ngành công nghiệp hiện đại.

2.15.4. Kết luận

Phần này đã cung cấp một cái nhìn chi tiết về các xu hướng và hướng phát triển mới trong OCR, bao gồm sự phát triển của các mô hình dựa trên học sâu, ứng dụng của Quantum Computing, và sự tích hợp của OCR với các công nghệ khác như AI và IoT. OCR đang tiếp tục tiến hóa và mở rộng phạm vi ứng dụng của mình, từ việc cải thiện hiệu suất đến mở ra những cơ hội mới trong các ngành công nghiệp và cuộc sống hàng ngày. Hiểu rõ và theo kịp những xu hướng này sẽ giúp bạn xây dựng các hệ thống OCR hiện đại và hiệu quả hơn trong tương lai.

2.16. Multilayer Perceptrons (MLP)

2.16.1 Kiến trúc MLP



Hình 2.1: Kiến trúc MLP. [1]

Các thành phần chính của kiến trúc mạng MLP như sau:

1. **Tầng nhập (Input layer):** Chứa các đặc trưng đầu vào của dữ liệu. Đây là nơi mà dữ liệu được đưa vào mạng để xử lý.
2. **Tầng ẩn (Hidden layer):** Bao gồm các neuron được xếp chồng lên nhau. Các tầng ẩn có thể có nhiều lớp và đóng vai trò quan trọng trong việc học và trích xuất các đặc trưng từ dữ liệu đầu vào.

3. **Các cạnh (Edges):** Mỗi cạnh đại diện cho một trọng số kết nối giữa các neuron trong các tầng khác nhau. Trọng số này được học trong quá trình huấn luyện để tối ưu hóa mô hình.
4. **Tầng xuất (Output layer):** Cung cấp kết quả dự đoán của mô hình đến từ tầng xuất. Kết quả đầu ra cuối cùng có thể là một số thực (cho các bài toán hồi quy) hoặc một tập hợp các xác suất (cho các bài toán phân loại). Điều này được xác định bởi hàm kích hoạt mà chúng ta sử dụng ở tầng xuất.

2.16.2 Hidden layer

Các tầng ẩn (hidden layers) trong mạng nơ-ron thực hiện các nhiệm vụ học khác nhau từ dữ liệu. Các tầng ẩn ban đầu tập trung vào việc phát hiện các mẫu đơn giản và học các đặc trưng cơ bản của dữ liệu. Các tầng ẩn tiếp theo sẽ tiếp tục phát hiện các mẫu phức tạp hơn từ các đặc trưng đã học, cho phép mô hình học các đặc trưng và hình dáng phức tạp hơn.

Trong mạng nơ-ron, các tầng ẩn được xếp chồng lên nhau để mô hình có thể học được các đặc trưng phức tạp và các mối quan hệ giữa chúng. Khi thiết kế mạng nơ-ron, nếu mô hình chưa đạt hiệu suất mong muốn hoặc chưa phù hợp với dữ liệu, một giải pháp có thể là thêm một số tầng ẩn để tăng cường khả năng học và trích xuất các đặc trưng phức tạp hơn từ dữ liệu..

2.16.3 Số lượng hidden layer, số lượng unit

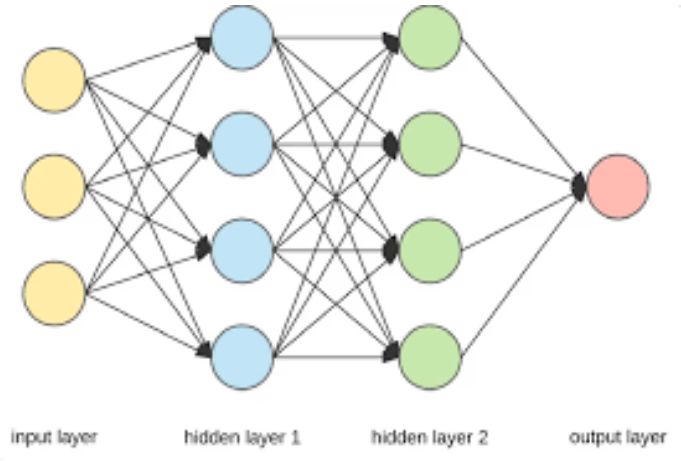
Theo quy tắc chung, nếu mạng nơ-ron càng sâu thì mô hình càng phù hợp với dữ liệu huấn luyện. Tuy nhiên, nếu mạng quá sâu sẽ gây nên hiện tượng quá khớp (overfitting) và làm tăng thời gian tính toán.Do đó, khi xây dựng mô hình mạng neuron, một phương pháp thực tiễn là bắt đầu với ba đến năm lớp tầmh ẩn (đặc biệt khi huấn luyện trên GPU) và quan sát hiệu suất của mô hình.

Nếu hiế suất không đạt yêu cầu và có dấu hiệu của hiện tượng chưa khớp (underfitting), tức là mô hình không học đủ đặc trưng của dữ liệu, có thể cần thêm nhiều tầng ẩn. Ngược lại, nếu mô hình xuất hiện hiện tượng quá khớp (overfitting), tức là mô hình học quá kỹ dữ liệu huấn luyện nhưng kém với dữ liệu kiểm tra, ta có thể giảm số lượng tầng ẩn để cải thiện khả năng tổng quát của mô hình.

2.16.4 Fully connected layers (FCN)

Trong kiến trúc mạng MLP (Multilayer Perceptron) cổ điển, các tầng được kết nối đầy đủ với tầng ẩn (hidden layer) tiếp theo. Mỗi nút trong một tầmh được kết nối với tất cả các nút trong layer trước đó. Kiến trúc này được gọi là mạng nơ-ron kết nối đầy đủ (Fully Connected Network).

Trong mạng nơ-ron kết nối đầy đủ, các cạnh giữa các nút đại diện cho trong số (weights), thể hiện tầm quan trọng của nút đối với các giá trị đầu ra. Trọng số này được học trong quá trình huấn luyện để tối ưu hóa mô hình và cải thiện khả năng dự đoán.



Hình 2.2: Fully Connected Network. [2]

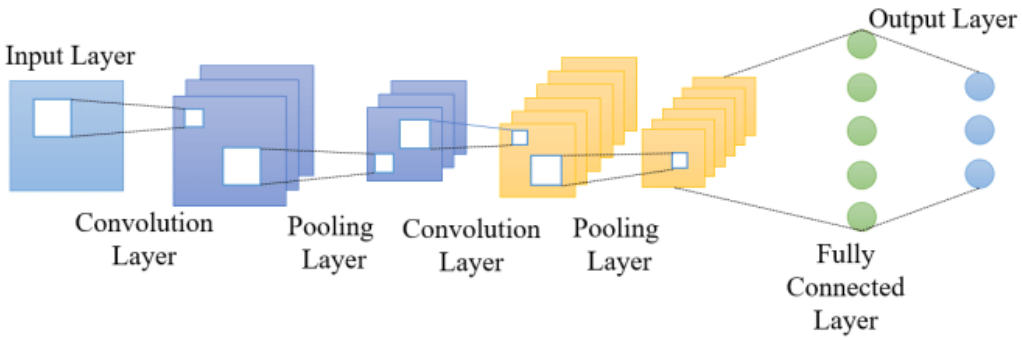
Giả sử ta có một mạng MLP gồm 2 lớp ẩn (hidden layer), mỗi lớp ẩn có 4 neuron(như hình 1.3):

- **Weights_0_1:** Từ lớp đầu vào (input layer) đến lớp ẩn 1 (hidden layer 1). Lớp đầu vào có 3 nơ-ron, lớp ẩn 1 có 4 nơ-ron. Sô trọng số là 3 x 4 = 12. Thêm 4 trọng số bias (mỗi nơ-ron trong lớp ẩn 1 có 1 trọng số bias), tổng là 12 + 4 = 16.
- **Weights_1_2:** Từ lớp ẩn 1 đến lớp ẩn 2. Cả hai đều có 4 nơ-ron. Số trọng số là 4 x 4 = 16. Thêm 4 trọng số bias (mỗi nơ-ron trong lớp ẩn 2 có 1 trọng số bias), tổng số là 16 + 4 = 20.
- **Weights_2_output:** Từ lớp ẩn 2 đến lớp đầu ra (output layer). Lớp ấn 2 có 4 nơ-ron, lớp đầu ra có 1 nơ-ron. Số trọng số là 4 x 1= 4. Thêm 1 trọng số bias cho nơ-ron đầu ra, tổng số là 4 + 1 = 5.
- **Tổng số cạnh (trọng số) trong mạng:** 16 + 20 + 5 = 41.

Với kiến trúc mạng như trên, ta có tổng 41 trọng số. Giá trị của các trọng số được khởi tạo ngẫu nhiên và sau đó mạng sẽ thực hiện quá trình feedforward (lan truyền tiến) và backpropagation (lan truyền ngược) để điều chỉnh giá trị của các trọng số.

2.17. Kiến trúc mạng nơ-ron tích chập – Convolutional Neural Network (CNN)

Mạng nơ-ron tích chập (CNN) bao gồm nhiều lớp khác nhau, trong đó mỗi lớp có khả năng học các đặc trưng ngày càng phức tạp. Lớp đầu tiên thường tập trung vào việc nhận diện các đặc trưng đơn giản như cạnh và đường thẳng. Các lớp kế tiếp sẽ học cách nhận diện các hình dạng phức tạp hơn như hình tròn và hình vuông. Ở các lớp sâu hơn, mô hình có thể nhận diện được những đặc trưng rất phức tạp, chẳng hạn như các bộ phận của khuôn mặt hoặc bánh xe.



Hình 2.3: Cấu trúc mạng CNN. [3]

2.17.1 Các thành phần cơ bản

CNN gồm 3 kiểu layer chính:

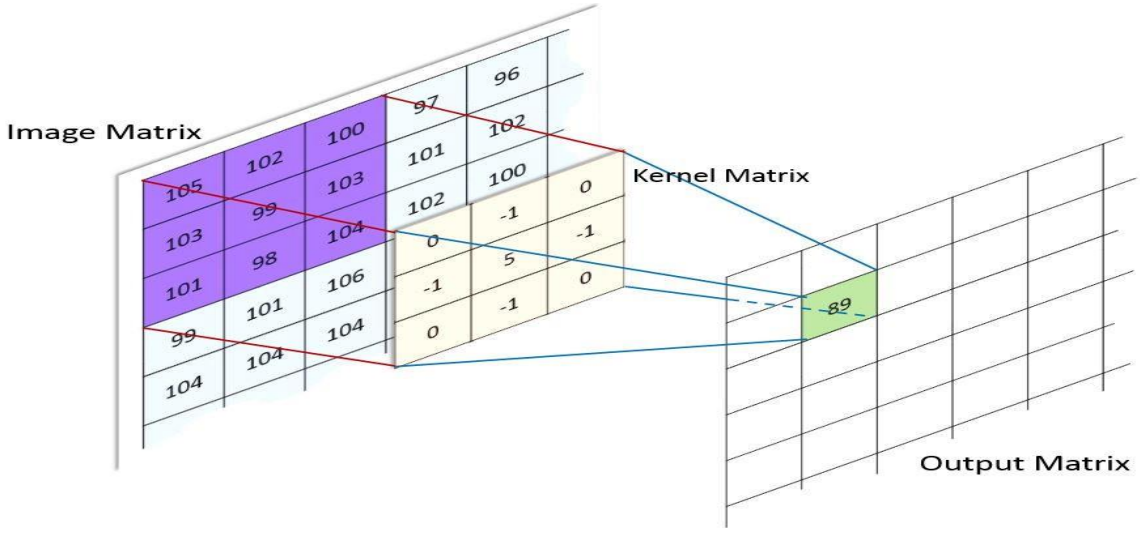
1. Lớp tích chập (Convolutional Layer - CONV)
2. Lớp gộp (Pooling layer - POOL)
3. Lớp kết nối đầy đủ (Fully Connected Layer - FC)

2.17.2 Lớp tích chập (Convolution Layer)

Convolution layer hoạt động như là một cửa sổ trượt qua từng pixel của ảnh để trích xuất các đặc trưng có ý nghĩa cho việc nhận dạng các đối tượng ảnh.

Bằng cách trượt bộ lọc tích chập (Convolutional Filter) qua ảnh input, mạng chia hình ảnh thành các thành phần nhỏ và xử lý các phần đó riêng lẻ .Kết quả là

một bản đồ đặc trưng (feature map), nơi các đặc trưng của ảnh đã được trích xuất và biểu diễn theo cách có nghĩa.

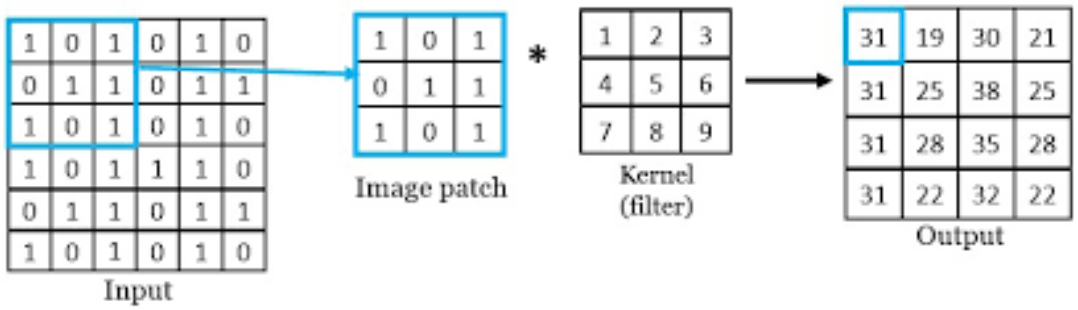


Hình 2.4: Một bộ lọc kích thước 3x3 đang trượt qua ảnh input. [4]

Ở (Hình 2.4), ta có các thành phần:

- Ma trận 3x3 ở giữa là Convoution Filer (bộ lọc tích chập) hay còn gọi là Kernel.
- Kernel trượt qua từng pixel của hình ảnh gốc và thực hiện một số phép tính toán học để tạo nên một hình ảnh mới với các giá trị pixel mới, gọi là feature map hay activation map.
- Vùng của ảnh mà Kernel đi qua gọi là Receptive Field (trường tiếp nhận).

Trong CNN, mỗi kernal là một tập hợp các trọng số. Điều này có nghĩa chúng sẽ được khởi tạo với các giá ngẫu nhiên và điều chỉnh trong quá trình huấn luyện. Bằng cách nhân mỗi pixel trong Receptive Field tương ứng với các pixcel trong kernal và cộng tất cả lại với nhau, ta sẽ thu được giá trị của pixel trong ảnh mới.



Hình 2.5: Kernal trượt qua ảnh ban đầu để tạo nên feature map. [5]

Input đầu vào là một ma trận 6x6 đại diện cho một phần của hình ảnh. Một phần nhỏ của dữ liệu đầu vào, gọi là miếng ảnh hoặc 'local receptive field,' được chọn để áp dụng tích chập. Trong hình, miếng ảnh này là một ma trận 3x3 nằm ở góc trên bên trái của dữ liệu đầu vào. Một ma trận 3x3 khác, gọi là kernel hoặc filter, được sử dụng để trích xuất các đặc trưng từ miếng ảnh. Kernel được trượt qua toàn bộ dữ liệu đầu vào, và tại mỗi vị trí, thực hiện phép nhân từng phần tử (element-wise multiplication) giữa miếng ảnh và kernel, sau đó cộng các kết quả lại với nhau để cho ra một giá trị duy nhất. Giá trị này được tính bằng cách nhân các giá trị tương ứng trong miếng ảnh và kernel, rồi cộng lại để cho ra kết quả là 31. Kết quả sau khi thực hiện tích chập tại mỗi vị trí của kernel sẽ trở thành một phần của bản đồ đặc trưng (feature map) ở đầu ra.

2.17.3 Kích thước của kernel

Kernel là một ma trận nhỏ chứa các trọng số, được sử dụng để quét qua dữ liệu đầu vào nhằm trích xuất các đặc trưng. Kích thước của kernel thường được xác định bởi hai yếu tố chính: **kích thước không gian (chiều cao và chiều rộng)** và **số lượng kênh (chiều sâu)**.

Kernel là một ma trận vuông với kích thước (2x2), (3x3),... Kích thước của kernel càng lớn thì khả năng mất mát các thông tin quan trong từ ảnh càng cao. Điều này bởi vì kernel lớn sẽ lấy mẫu thông tin từ một vùng rộng hơn của ảnh đầu vào, có thể dẫn đến việc làm mất các chi tiết nhỏ hoặc chính xác.

Công thức tính kích thước của ouput:

Khi ta áp dụng kernel có kích thước (F,F) và input có kích thước (I,I):

$$\theta = I - F + 1 \quad (1)$$

Giải thích công thức (1):

- θ: Kích thước của output (chiều rộng/chiều cao).
- I: Kích thước của input (chiều rộng/chiều cao).
- F: Kích thước của kernel/filter.

2.17.4 Strides

Strides là số lượng pixel mà kernal trượt trên ảnh. Ví dụ, để trượt kernel một pixel tại một thời điểm, thì giá trị strides = 1. Nếu kernel di chuyển hai pixel cùng một lúc, thì ta đặt strides = 2. Các giá trị strides lớn ơn hai (như ba trở lên) là không phổ biến và hiếm khi gặp trong thực tế.

Công thức tính kích thước của output:

Khi ta áp dụng kernel có kích thước (F,F) vào input có kích thước (I,I) và có thêm giá trị strides = S, kích thước của đầu ra (output size) được tính bằng công thức:

$$\theta = \frac{I - F}{S} + 1 \qquad (2)$$

Giải thích kí hiệu công thức (2):

- θ: Kích thước của output (output size).Đây là chiều cao (hoặc chiều rộng) của đầu ra, giả sử đầu ra hình vuông.
- I : Kích thước của input (input size). Đây là chiều cao (hoặc chiều rộng) của input, giả sử input là hình vuông.
- F: Kích thước của kernel (filter size). Đây là chiều cao (hoặc chiều rộng) của kernel, giả sử kernel là hình vuông.
- S: Giá trị strides. Đây là số pixel mà kernel di chuyển trên input tại mỗi bước.

2.17.5 Padding

Khi ta áp dụng kernel vào ảnh input sẽ đẫn đến các vấn đề như.Làm giảm kích thước của ảnh input và có thể mất các thông tin quan trọng từ ảnh. Để khắc phục vấn đề này, chúng ta sử dụng padding để khắc phục vấn đề đề đó.

Padding thường được gọi là zero-padding bởi vì ta thêm các số 0 xung quanh đường viền của hình ảnh. Với padding = 1 , ta thêm một đường viền gồm các số 0 vào ảnh, padding = 2, ta thêm hai đường viền gồm các số 0 vào ảnh. Padding

được sử dụng phổ biến để duy trì kích thước không gian của đầu vào, đảm bảo chiều rộng và chiều cao tương ứng với đầu vào nếu cần.

Padding có 2 dạng chính :

- **Valid:** Không có padding hay nói cách khác là padding = 0. Lúc này công thức tính output của ta được sử dụng như công thức (2)
- **Same:** Thêm các viền số 0 sao cho kích thước output bằng với kích thước input. Lúc này ta có công thức tổng quát mới :

$$\theta = \frac{I - F + 2 * P}{S} + 1 \qquad (3)$$

Giải thích kí hiệu công thức (3):

- θ: Kích thước của output (output size).Đây là chiều cao (hoặc chiều rộng) của đầu ra, giả sử đầu ra là hình vuông
- I: Kích thước của input (input size). Đây là chiều cao (hoặc chiều rộng) của ảnh đầu vào.
- F: Kích thước của kernel (filter size). Đây là chiều cao (hoặc chiều rộng) của kernel.
- P: Giá trị padding. Đây là số lớp số 0 thêm vào xung quanh ảnh đầu vào.
- S: Giá trị strides. Đây là số pixel mà kernel di chuyển trên ảnh đầu vào tại mỗi bước.

2.17.6 Pooling layer

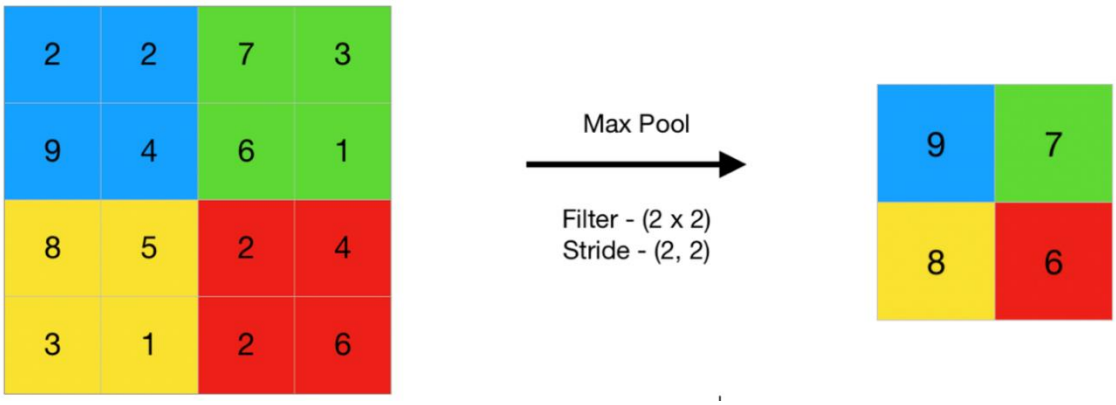
Khi số lượng lớp tích chập (convolutional layer) tăng lên, đồng nghĩa với việc ảnh càng sâu hơn, dẫn đến việc gia tăng số lượng các tham số mà mô hình cần phải tối ưu. Lớp gộp (pooling layer) có nhiệm vụ làm giảm kích thước của các feature map được sinh ra sau mỗi lớp tích chập. Trong thực thế, pooling layer được thêm vào sau mỗi một hay hai lớp tích chập.

Khi các feature map của lớp tích chập (convolutional layer) đi qua lớp gộp (pooling layer), kích thước chiều rộng và chiều cao của chúng sẽ giảm , nhưng độ sâu vẫn giữ nguyên.

Ở Pooling layer, ta không có bất kì tham số nào để mô hình học. Tương tự như kernel của lớp tích chập, lớp gộp cũng có kích thước và giá trị strides để di chuyển qua ảnh.

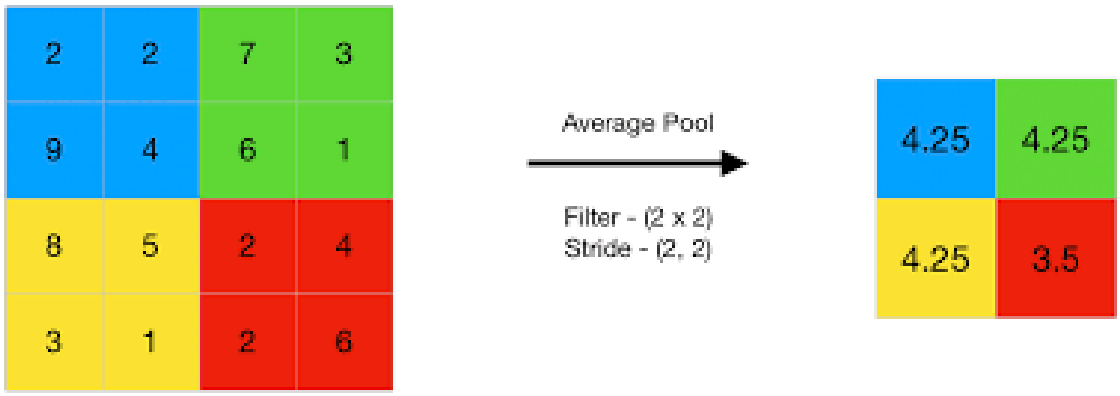
Có hai kiểu chính của pooling layer là: Max Pooling và Average Pooling.

- **Max Pooling**: Max Pooling sẽ trượt qua feature map được tạo ra từ Convolutional Layer trước nó và chọn giá trị pixel lớn nhất trong vùng con của feature map. Các giá trị pixel còn lại trong vùng con được loại bỏ.



Hình 2.6: Max Pooling. [6]

- **Average Pooling:** Average Pooling trượt qua feature map nhưng không chọn giá trị lớn nhất như Max Pooling mà sẽ tính giá trị trung bình của tất cả các pixel trong mỗi vùng con, sau đó sử dụng giá trị trung bình này làm giá trị đại diện cho vùng đó. Nếu giá trị trung bình không phải là một số nguyên, thì sẽ làm tròn về số nguyên gần nhất.



Hình 2.7: Average Pooling. [6]

2.17.7 Fully-Connected layer

Sau khi đã có được các feature map được tạo ra từ các lớp tích chập (convolutional layer) và lớp gộp (pooling layer), bước tiếp theo là chuyển đổi feature map thành vector đặc trưng. Điều này được thực hiền bằng cách làm phẳng (flatten) các feature map.

Vector đặc trưng sau khi flatten sẽ được đưa vào lớp kết nối đầy đủ (fully connected layer). Trong lớp kết nối đầy đủ, mỗi nơ-ron của lớp sẽ kết nối với tất cả các nơ-ron của lớp trước đó. Đây là giai đoạn mà mạng nơ-ron thực hiện các phép toán phân loại hoặc hồi quy dựa trên các đặc trưng đã được trích xuất và flatten.

Lớp kết nối đầy đủ đóng vai trò quan trọng trong việc đưa ra dự đoán cuối cùng của mạng, cho phép mạng nơ-ron kết hợp với các đặc trưng học được từ các lớp trước đó để thực hiện nhiệm vụ như phân loại, hồi quy, hay nhận diện đối tượng.

2.17.8 Hàm kích hoạt (Activation Function)

Hàm kích hoạt còn được biết đến như là hàm chuyển đổi (Transfer Function) hoặc hàm phi tuyến (Nonlinear Function) vì nó chuyển giá trị tuyến tính của hàm tổng trọng số (weighted sum) sang giá trị phi tuyến tính.

Mục đích của hàm kích hoạt là để đưa tính phi tuyến vào mạng nơ-ron. Nếu không có hàm kích hoạt, mạng nơ-ron nhiều lớp (MLP) sẽ hoạt động như một perceptron đơn lớp, bất kết số lượng lớp có bao nhiêu.

Hàm truyền (hàm kích hoạt hay hàm chuyển đổi) tính toán đầu ra của một nơ-ron để chuyển đến lớp tiếp theo trong mạng nơ-ron. Hàm kích hoạt phi tuyến

được sử dụng vì mạng chỉ sử dụng các hàm kích hoạt tuyến tính , mô hình có thể lược giản hóa qua các biến đổi đại số thành mô hình perceptron một lớp (là mô hình mạng nơ-ron đơn giản nhất, không có lớp ẩn).

Một số hàm kích hoạt phi tuyến thường dùng là ReLU (Rectified Linear Unit), Sigmoid, Logistic, Gaussian, Tanh, Softmax (Hình 1.13).

Hàm ReLU:

$$f(x) = \begin{cases} 0\,, x < 0 \\ x\,, x \geq 0 \end{cases} \quad (4)$$

Hàm Sigmoid:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

Hàm Tanh:

$$f(x) = \frac{1}{1 + e^{-2x}} - 1 \quad (6)$$

Kết quả xử lý đầu ra hàm tổng (weighted sum) của nơ-ron đôi khi quá lớn. Để xử lý đầu ra này trước khi chuyển đến lớp tiếp theo, hàm kích hoạt thường được sử dụng. Hàm kích hoạt giúp biến đổi giá trị đầu ra của nơ-ron thành một giá trị phù hợp với yêu cầu của mạng nơ-ron, đồng thời cung cấp tính phi tuyến cần thiết cho mô hình.

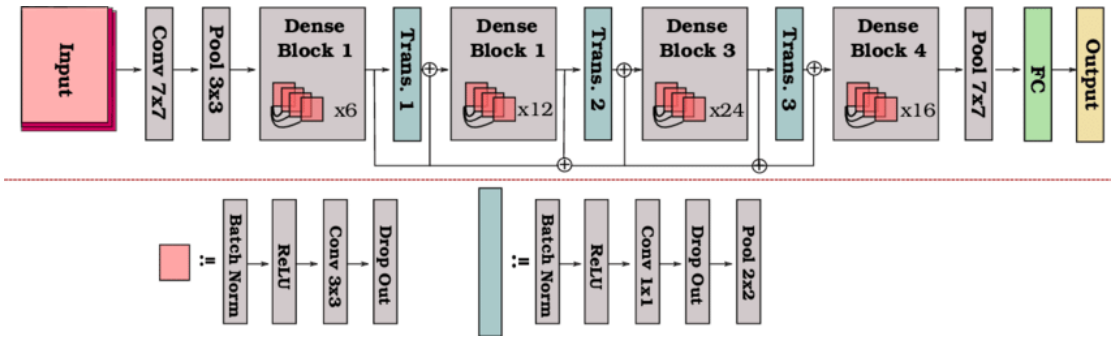
2.18. DenseNet121

DenseNet121 là một trong những mô hình của Dense Convolutional Network (DenseNet), một loại mạng nơ-ron tích chập (CNN) được giới thiệu bởi Gao Huang và cộng sự trong bài báo "Densely Connected Convolutional Networks" vào năm 2017.

2.18.1 Kiến trúc mạng DenseNet121

1. **Densely Connected Blocks:** Trong DenseNet, mỗi lớp nhận được đầu vào từ tất cả các lớp trước đó. Điều này có nghĩa là các đặc trưng từ tất cả các lớp trước đều được truyền thẳng tới các lớp tiếp theo, giúp cải thiện sự truyền thông tin và gradient.

2. **Composite Function:** Mỗi lớp trong một Dense Block bao gồm ba hoạt động chính: Batch Normalization (BN), followed by ReLU activation, and then a Convolutional layer. Công thức này được lặp lại nhiều lần.
3. **Bottleneck Layers:** Để giảm số lượng tham số và tính toán, DenseNet sử dụng các lớp bottleneck, nơi mà một số lượng lớn các kênh đầu vào được giảm xuống nhờ một lớp tích chập 1x1 trước khi áp dụng một lớp tích chập 3x3.
4. **Transition Layers:** Để giảm chiều kích thước của đặc trưng và số lượng kênh, giữa các Dense Block có các lớp chuyển đổi (Transition layers) với tích chập 1x1 và Pooling 2x2.
5. **Growth Rate:** Một tham số quan trọng trong DenseNet là "tốc độ tăng trưởng" (growth rate), định nghĩa số lượng kênh được thêm vào mỗi lớp. Với DenseNet121, growth rate thường là 32.



Hình 2.8: Sơ đồ minh họa mô hình DenseNet121. [7]

2.18.2 Các phiên bản của DenseNet

- **DenseNet121**: Với 121 lớp, là phiên bản nhẹ nhất và ít tham số hơn, phù hợp cho các thiết bị có khả năng tính toán hạn chế.
- **DenseNet169**: Có 169 lớp, cung cấp một cân bằng tốt giữa độ sâu và hiệu quả tính toán.
- **DenseNet201**: Với 201 lớp, cung cấp độ sâu lớn hơn, giúp nắm bắt các đặc trưng phức tạp hơn.
- **DenseNet264**: Phiên bản sâu nhất trong các phiên bản phổ biến, với 264 lớp.

2.18.3 Cải tiến và Ưu điểm

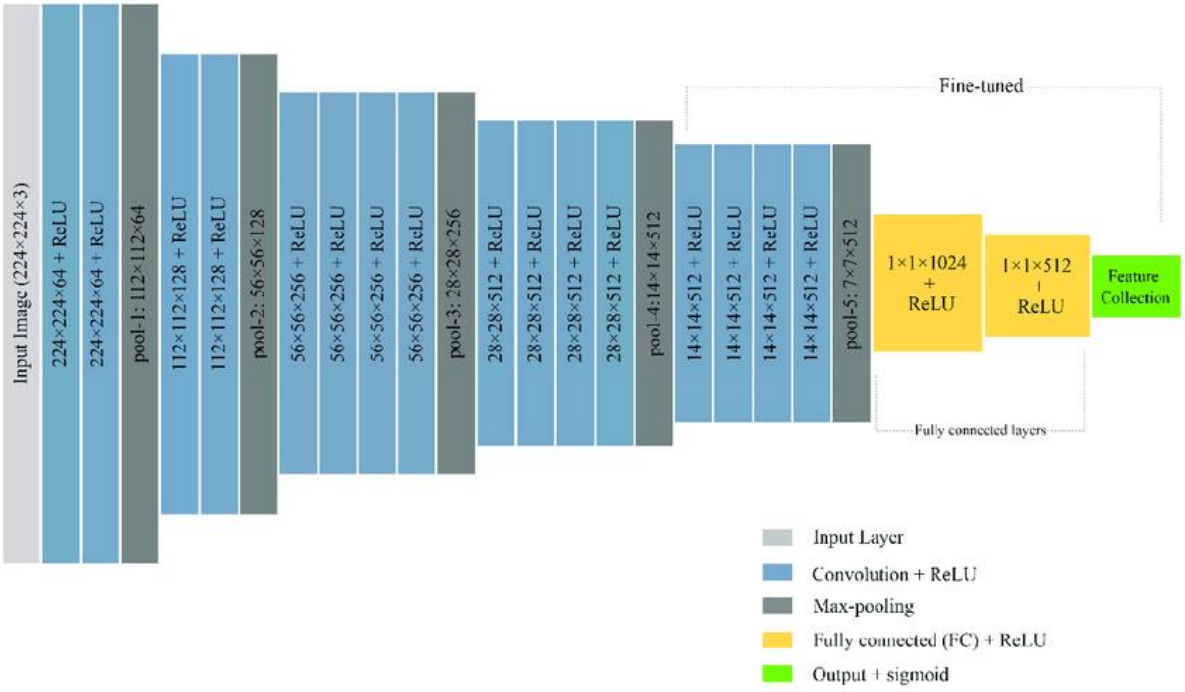
1. **Khả năng Tái sử dụng Đặc trưng:** DenseNet giúp tái sử dụng đặc trưng rất hiệu quả, giảm số lượng tham số cần thiết và cải thiện việc huấn luyện.
2. **Truyền Gradient Tốt Hơn:** Cấu trúc Dense giúp truyền gradient tốt hơn, khắc phục vấn đề gradient vanishing thường gặp trong các mạng sâu.
3. **Tiết Kiệm Tham Số:** So với các kiến trúc khác có cùng độ chính xác, DenseNet sử dụng ít tham số hơn do các lớp tích chập 1x1 và cấu trúc kết nối dày đặc.

2.19. VGG-19

Mạng VGG19 là một trong những mạng nơ-ron tích chập (Convolutional Neural Network - CNN) nổi tiếng, được phát triển bởi các nhà nghiên cứu từ Visual Geometry Group (VGG) của Đại học Oxford. Mạng này được giới thiệu lần đầu tiên trong bài báo "Very Deep Convolutional Networks for Large-Scale Image Recognition" năm 2014.

2.19.1 Kiến trúc mạng VGG19

Kiến trúc của VGG19 bao gồm 19 tầng, trong đó có 16 tầng tích chập (convolutional) và 3 tầng kết nối đầy đủ (fully connected).



Hình 2.9: Mô hình VGG19 đơn giản [8]

Như ta thấy được từ Hình 2.5, Mô hình VGG19 xử lý dầu vào thông qua sau khối (blocks) chức năng, mỗi khối đóng một vai trò quang trọng trong quá trình trichx xấut và học đặc trưng (features). Block1, với hai lớp tích chập (convolutional layers) 3x3 và 64 bộ lọc (filters), tập trung vào nhận diện các đặc trưng cơ bản nhất từ cạnh (edges) và góc (corners). Lớp Max Pooling kết thúc khồi này giảm kích thước không gian, tập trung trung vào những thông tin quan trọng. Block2, cũng với 2 lớp tích chập nhưng tăng lên về 128 bộ lọc, bắt đầu kết hợp các đặc trưng cơ bản thành các mẫu phức tạp hơn. Tiến sâu hơn Block3 mở rộng lên bốn lớp tích chập với 256 bộ lọc, cho phép mạng học các đặc trưng trừu tượng hơn, có khả năng nhận diện các phần của đối tượng.Block4,5 mỗi block có bốn lớp tích chập với 512 bộ lọc, đẩy mạnh khả năng trích xuất các đặc trưng cao cấp (high-level features), có thể đại diện cho các phần lớn hoặc toàn bộ đối tượng trong ảnh. Cuối cùng, block6 với 3 lớp funlly connected (FC layers) đóng vai trò quyết định.Hai lớp fully connected đầu tiên, mỗi lớp với 4096 đơn vị (units), nằm trích hợp thông tin từ tất cả các đặc trưng không gian (spatial feature) được trích xuất ở các block trước đó, trong khi lớp fully connected cuối cùng với 1000 unit ánh xạ các đặc trưng này vào 1000 lớp đối tượng cần phải phân loại (classification). Đây là lớp output của mạng, cho ra kết quả dự đoán lớp (class)

của ảnh đầu vào. Nếu đối với bài toán phát hiện vật thể, ta cần thêm các lớp tùy chỉnh như Region Proposal Network (RPN) và các lớp để dự đoán bounding box và phân loại vật thể trong bounding box. Đối với một bài toán khác là phân đoạn hình ảnh ta thêm các lớp deconvolutional để tăng kích thước không gian đầu ra trở lại kích thước gốc.

2.19.2 Các phiên bản của VGG

VGG có một số mô hình khác nhau như VGG11, VGG13, VGG16 và VGG19. Mô hình nhỏ nhất là VGG11, có độ chính xác thấp nhất trên tập ImageNet so với các phiên bản VGG khác. Ngược lại, mô hình lớn nhất VGG19, đạt độ chính xác cao nhất trên tập ImageNet.

Về tốc độ: Do VGG19 có nhiều lớp hơn nên VGG19 yêu cầu thời gian và tài nguyên tính hơn để huấn luyện so với VGG11,VGG13 và VGG16. So với VGG16 ra mắt trước đó thì VGG16 có thể huấn luyện nhanh hơn và yêu cầu ít tài nguyên hơn so với VGG19.

Độ chính xác: VGG19 thường có độ chính xác cao hơn trên các tác vụ phân loại hình ảnh so với VGG16, do kiến trúc sâu hơn giúp mô hình học được nhiều đặc trưng phức tạp hơn. VGG16 mặc dù có độ chính xác thấp hơn VGG19 nhưng vẫn là một mô hình rất mạnh và thường được sử dụng rộng rãi hơn.

2.19.3 Những cải tiến của VGG19 so với VGG16

So với VGG16 thì VGG19 có một cố cải tiến về số lớp convolutional. VGG19 có 19 lớp convolutional, nhiều hơn 3 lớp convolutional so với VGG16, vốn có 16 lớp convolutional. Việc tăng số lớp convolutional cho phép VGG19 học được các đặc trưng phức tạp và chi tiết hơn so với VGG16. Điều này có thể dẫn đến hiệu suất tốt hơn trên các tập dữ liệu phức tạp.Trong các thử nghiệm với tập dữ liệu Imagenet, VGG19 thường đạt hiệu suất tốt hơn so với VGG16 do khả năng học sâu hơn và trích xuất đặc trưng chi tiết hơn.

2.20. Transfer Learning

2.20.1 Định nghĩa Transfer Learning

Bình thường khi chúng ta train (huấn luyện) từ đầu một mô hình (model) nhận diện ảnh thì phải có một số lượng dữ liệu đủ lớn, cũng như cấu hình máy

tính thật mạnh mẽ để có thể tiến hành train, chưa kể đến việc chúng ra phải viết code từ đầu có khi lại không xây dựng và chạy được mô hình. Thế nên người ta nghĩ đến phương pháp Transfer Learning.

Học chuyển giao (transfer learning) là quá trình khai thác, tái sử dụng lại các tri thức của một mô hình đã được học tập trên một tập dữ liệu trước đó để sử dụng vào 1 bài toán mới mà không cần phải xây dựng lại cả một mô hình huấn luyện từ đầu.

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

Ví dụ khi ta học tiếng Pháp: ban đầu, ta dành thời gian học tiếng Pháp và trở nên thành thạo về ngữ pháp, từ vựng, cách phát âm, và các cấu trúc câu phức tạp. Khi ta chuyển sang học tiếng Tây Ban Nha, sẽ có nhiều từ vựng, cấu trúc ngữ pháp, và cách phát âm tương đồng với tiếng Pháp. Nhờ đó, ta sẽ rút ngắn được thời gian học và đạt hiệu quả cao hơn một cách nhanh chóng.

2.20.2 Phân loại Transfer Learning

Có 2 loại Transfer Learning:

- **Feature Extractor (trích xuất đặc trưng):** là một kỹ thuật transfer learning trong đó ta sử dụng một mô hình đã được huấn luyện trước đó trên một tập dữ liệu lớn (thường được gọi là mô hình cơ sở hoặc pre-trained model) như một bộ trích đặc trưng cố định. Phương pháp này dựa trên giả định rằng các đặc trưng hoc được từ tập dữ liệu gốc có thể hữu ích cho tác vụ mới, mặc dù tác vụ mới có thể khác biệt đáng kể.
- **Fine turning (tinh chỉnh):** là một kỹ thuật transfer learning nâng cao hơn, trong đó ta không chỉ sử dụng mô hình cơ sở như một bộ trích xuất đặc trưng cố định, mà còn điều chỉnh các trọng số của nó để phù

hợp hơn với tác vụ mới. Phương pháp này cho phép mô hình thích nghi tốt hơn với đặc thù của tác vụ mới, đồng thời vẫn vận dụng được kiến thức đã được học tập từ tập dữ liệu gốc.

2.20.3 Khi nào nên dùng Transfer Learning ?

Khi không có đủ dữ liệu: khi tập dữ liệu cho tác vụ mới quá nhỏ, không đủ để huấn luyện một mô hình hiệu quả từ đầu. Transfer learning giúp tận dụng kiến thức từ mô hình đã huấn luyện trước đó, cải thiện kết quả dù ít dữ liệu

Không đủ tài nguyên phần cứng: nếu phần cứng không đủ mạnh mẽ hoặc thười gian để huấn luyện quá phức tạp. Transfer learning giảm đáng kể yêu cầu về tài nguyên và thời gian huấn luyện.

Cải thiện chất lượng mô hình: việc sử dụng transfer learning giúp cải thiện hiệu suất và độ chính xác của mô hình, đặc biệt khi làm việc với dữ liệu hạn chế hoặc trong lĩnh vực chuyên biệt.

Thích ứng nhanh: khi cần nhanh chóng áp dụng mô hình cho tác vụ mới, transfer learning cho phép điều chỉnh mô hình hiện có thay vì bắt đàu từ đầu.

Lĩnh vực đặc thù: trong các ngành như y học hay khoa học, nơi việc thu tập dữ liệu khó khăn và tốn kém, transfer learning mở ra khả năng áp dụng học máy hiệu quả.

2.21. Các nghiên cứu liên quan

Để tăng cường độ chính xác trong các bài toán liên quan đến xử lý ảnh và nhận diện văn bản trong ảnh, các nhà nghiên cứu đã không ngừng nỗ lực phát triển và cải tiến các phương pháp mới. Các phương pháp này bao gồm kỹ thuật nhận dạng mẫu sử dụng mạng nơ-ron tích chập (CNN), học sâu không giám sát, và các mô hình học sâu tiên tiến. Những phương pháp này đã giúp cải thiện đáng kể hiệu suất trong nhiều ứng dụng xử lý ảnh khác nhau.

- Nghiên cứu "Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles" (2016) của Mehdi Noroozi và Paolo Favaro giới thiệu phương pháp học không giám sát thông qua việc giải quyết bài toán xếp hình. Mô hình này đạt độ chính xác 51.8% cho bài toán phát hiện và 68.6% cho bài toán phân loại.

- Nghiên cứu "Scene Text Recognition with Convolutional Neural Networks" (2014) của Wang và cộng sự đã trình bày một phương pháp sử dụng CNN để nhận diện văn bản trong các bối cảnh thực tế. Mô hình của họ đạt độ chính xác cao trên tập dữ liệu ICDAR 2011 và SVT.
- Tiếp theo, nghiên cứu "Reading Text in the Wild with Convolutional Neural Networks" (2016) của Jaderberg và cộng sự đã giới thiệu một hệ thống nhận diện văn bản trong ảnh sử dụng mạng nơ-ron tích chập kết hợp với các phương pháp học sâu khác. Hệ thống này đạt kết quả tốt trên nhiều tập dữ liệu khó như ICDAR 2013 và IIIT 5K-Words.
- Nghiên cứu "Synthetic Data for Text Localisation in Natural Images" (2016) của Gupta và cộng sự đã đề xuất việc sử dụng dữ liệu tổng hợp để huấn luyện các mô hình nhận diện văn bản. Phương pháp này giúp cải thiện đáng kể hiệu suất nhận diện văn bản trong các bối cảnh khác nhau bằng cách sử dụng các hình ảnh được tạo ra một cách tự động.

Trong lĩnh vực nhận diện văn bản, các nhà nghiên cứu phải đối mặt với nhiều thách thức và khó khăn đáng kể. Một trong những vấn đề chính là việc trích xuất các đặc trưng có tính phân biệt cao từ hình ảnh, đặc biệt là trong các tập dữ liệu lớn và đa dạng. Khó khăn này trở nên phức tạp hơn khi phải xử lý các hình ảnh có độ phức tạp cao, chứa nhiều đối tượng, hoặc có sự thay đổi về góc nhìn, ánh sáng và tỷ lệ. Ngoài ra, việc thu thập và gán nhãn cho một lượng lớn dữ liệu huấn luyện chất lượng cao cũng là một thách thức lớn, đòi hỏi nhiều thời gian và nguồn lực.

Một khó khăn khác là cân bằng giữa độ chính xác và tốc độ xử lý, đặc biệt là trong các ứng dụng thời gian thực. Các mô hình học sâu phức tạp thường đòi hỏi tài nguyên tính toán lớn, gây khó khăn cho việc triển khai trên các thiết bị có tài nguyên hạn chế. Cuối cùng, việc tạo ra các mô hình có khả năng tổng quát hóa tốt, có thể hoạt động hiệu quả trên nhiều loại dữ liệu và tình huống khác nhau, vẫn là một thách thức lớn trong lĩnh vực này.

Nghiên cứu hiện tại nhằm mục đích giải quyết những thách thức này bằng cách đề xuất một chiến lược học chuyển giao (transfer learning) mới dựa trên các

mô hình CNN đã được huấn luyện trước, cụ thể là VGG19 và DenseNet121, để đạt được hiệu quả cao cho bài toán nhận diện văn bản trong ảnh. Phương pháp này hướng đến việc tận dụng kiến thức đã học từ cả hai mô hình: khả năng trích xuất đặc trưng mạnh mẽ của VGG19 và khả năng biểu diễn học sâu của DenseNet121.

Các đóng góp chính của nghiên cứu có thể được tóm tắt lại như sau:

a) Đề xuất một chiến lược học sâu mới: Nghiên cứu đề xuất một mô hình kết hợp VGG19 và DenseNet121, tận dụng ưu điểm của cả hai kiến trúc để cải thiện hiệu quả trích xuất đặc trưng và giảm sự suy giảm gradient.

b) Áp dụng các kỹ thuật tăng cường dữ liệu tiên tiến: Sử dụng các phương pháp tăng cường dữ liệu để cải thiện khả năng tổng quát hóa của mô hình, giúp mô hình hoạt động hiệu quả trên tập dữ liệu đa dạng và lớn.

c) Tối ưu hóa mô hình cho các thiết bị hạn chế tài nguyên: Áp dụng các kỹ thuật lượng tử hóa mô hình và cắt tỉa mạng nơ-ron để đảm bảo mô hình có thể triển khai hiệu quả trên các thiết bị có tài nguyên hạn chế.

d) Xây dựng một giải pháp nhận diện văn bản hiệu quả và có thể áp dụng rộng rãi: Mục tiêu cuối cùng của nghiên cứu là tạo ra một hệ thống nhận diện văn bản có độ chính xác cao, hiệu quả và có thể được ứng dụng rộng rãi trong thực tế.

Chương 3: PHƯƠNG PHÁP ĐỀ XUẤT

3.1. Mô tả bài toán

Bài toán xác định văn bản trong ảnh bằng phương pháp máy học là nghiên cứu quan trọng trong thị giác máy tính, xử lý hình ảnh và truy xuất thông tin. Mục đích của bài toán là phát hiện và trích xuất đặc trưng các văn bản. Trong thực tế, xác định văn bản trong ảnh có nhiều ứng dụng hữu ích, như trích xuất thông tin từ các loại tài liệu, hỗ trợ dịch thuật tự động, tìm kiếm và lưu trữ ảnh có chứa văn bản, phân tích nội dung ảnh.

Trong bài toán này, luận văn sử dụng phương pháp quy trình nghiên cứu được chia thành 2 phần: phần nhận dạng ký tự và phần xử lý văn bản. Mỗi phần được chia thành nhiều giai đoạn. Trong phần nhận dạng ký tự, quy trình bao gồm: thu thập dữ liệu ký tự, tăng cường dữ liệu, huấn luyện mô hình và đánh giá mô hình. Trong phần xử lý văn bản cảnh, quy trình bao gồm: thu thập dữ liệu văn bản, xử lý hình ảnh, phát hiện và cắt bounding box cũng như đánh giá mô hình trên dữ liệu văn bản.

- Trong phần nhận dạng ký tự, luận văn sử dụng dữ liệu ký tự bảng chữ cái để huấn luyện và đánh giá, dữ liệu được tải vào và áp dụng tăng cường sinh để mở rộng loại dữ liệu và chuẩn hóa. Dữ liệu huấn luyện sau đó được đào tạo bằng để tạo mô hình học sâu để nhận dạng với hiệu suất cao. Mô hình được đánh giá bằng dữ liệu kiểm tra và xác thực. Kết quả đánh giá được phân tích để cải thiện độ chính xác.
- Trong phần xử lý văn bản, luận văn sử dụng dữ liệu văn bản để đánh giá mô hình đã được đào tạo trước đó. Dữ liệu văn bản được xử lý bằng kỹ thuật xử lý hình ảnh để hỗ trợ phát hiện bounding box. Sau khi xử lý ảnh, chúng ta xác định khung bounding box của từng ký tự bằng cách tính toán đường viền. Bounding box được cắt và lưu dưới dạng hình ảnh mới, sử dụng làm dữ liệu thử nghiệm và đánh giá mô hình. Kết quả đánh giá được phân tích để đánh giá độ chính xác của các mô hình.

Để giải quyết bài toán này, nhóm sử dụng các phương pháp máy học theo trình tự:

- **Chuẩn bị dữ liệu ký tự:** Thu thập và chuẩn bị dữ liệu. Xử lý dữ liệu, chuẩn hóa kích thước, chia tập dữ liệu, tăng sinh ảnh, chia tập dữ liệu thành 3 phần mới huấn luyện (train), kiểm tra (test), xác thực (validation).
- **Huấn luyện mô hình:** Sử dụng tập dữ liệu đã xử lý, tiếp tục xây dựng các mô hình học sâu luận văn chọn là CNN, DenseNet, VGG19. Quá trình này bao gồm:
    - Biên dịch mô hình với các trọng số đã được huấn luyện trước.
    - Áp dụng các kỹ thuật tối ưu hóa: điều chỉnh tỷ lệ huấn luyện, điều chỉnh tham số.
- **Đánh giá, lưu mô hình:** Đánh giá độ chính xác mô hình, vẽ biểu đồ đánh giá mất mát và độ chính xác của từng mô hình. Lưu từng mô hình để áp dụng vào việc xác định văn bản trong tập dữ liệu văn bản. Hàm mất mát sử dụng là categorical crossentropy, và chỉ số đánh giá là accuracy.
- **Chuẩn bị dữ liệu văn bản:** Sử dụng mô hình để nhận dạng văn bản.
- **Đánh giá kết quả:** Sử dụng đánh giá độ chính xác ký tự (character accuracy), đánh giá độ chính xác của mô hình nhận dạng ký tự quang học (OCR) trên tập dữ liệu văn bản. Phương pháp này so sánh từng ký tự trong chuỗi dự đoán với ký tự tương ứng trong chuỗi thực tế và tính toán tỷ lệ phần trăm các ký tự dự đoán đúng so với tổng số ký tự.
    - **Input:** Dữ liệu đầu vào là tập ảnh, mỗi ảnh chứa một chuỗi ký tự, dùng mô hình để xác định văn bản trong ảnh.
    - **Output:** Trả về là kết quả của từng mô hình và in kết quả sau khi xác định văn bản của một hình ảnh ngẫu nhiên.

3.2. Phương pháp đề xuất

3.2.1 Mô hình học sâu Convolutional Neural Network (CNN)

CNN (Convolutional Neural Network) là một kiến trúc mạng neural được thiết kế để hiệu quả trong việc xử lý dữ liệu có cấu trúc lưới, chẳng hạn như ảnh ảnh. CNN bao gồm các tầng chính:

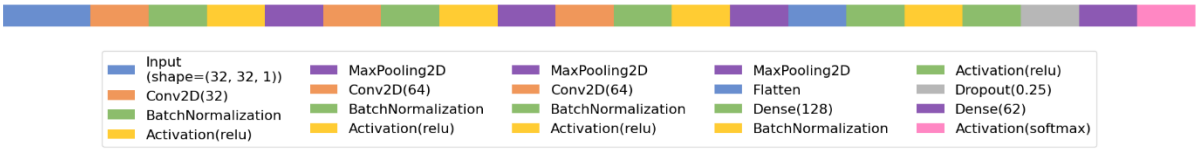
- **Tầng Convolution:** Thực hiện phép tích chập để trích xuất các đặc trưng cục bộ từ dữ liệu đầu vào.
- **Tầng Pooling:** Giảm kích thước không gian của các đặc trưng, giữ lại thông tin quan trọng.
- **Tầng Fully Connected:** Kết nối tầng cuối cùng để thực hiện phân loại hoặc hồi quy.

CNN rất phù hợp cho nhiệm vụ xác định văn bản trong ảnh vì: tầng Convolution có thể trích xuất các đặc trưng cục bộ như các nét, cạnh, góc trong ảnh văn bản; tầng Pooling giúp giảm kích thước đặc trưng trong khi vẫn giữ được thông tin quan trọng; tầng Fully Connected có thể học cách phân loại các mẫu văn bản từ đặc trưng được trích xuất.

CNN là một lựa chọn rất phù hợp cho bài toán nhận dạng văn bản trong ảnh, với khả năng trích xuất đặc trưng tự động và phân loại hiệu quả. Luận văn đã nghiên cứu và tinh chỉnh cho mô hình hoàn thiện, thích hợp để áp dụng cho tập dữ liệu.

- **Với mô hình xử lý ảnh xám:**

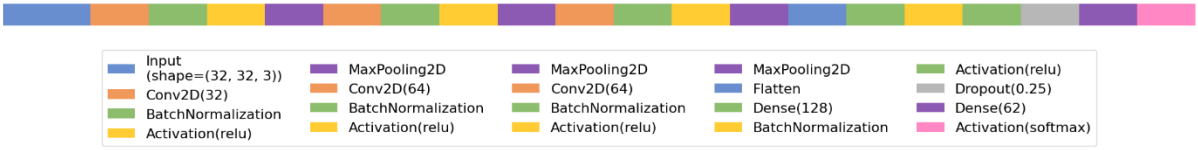
Nhóm sử dụng mô hình Sequential để xây dựng kiến trúc CNN bao gồm các lớp sau: input đầu vào sẽ là (32,32,1) với 32x32 là kích thước ảnh và 1 là số kênh màu (số kênh màu của ảnh xám chỉ có 1), lớp convolutional sử dụng các lớp Conv2D với 32 và 64 filters, kích thước kernel 3x3, và padding 'same', mỗi lớp convolutional đều có BatchNormalization và Activation('relu'); lớp pooling sử dụng các lớp MaxPooling2D với kích thước pool 2x2 để giảm kích thước đặc trưng; lớp flatten: Chuyển đổi từ tensor sang vector để đưa vào các lớp fully connected; lớp fully connected sử dụng lớp Dense với 128 neuron, BatchNormalization, Activation ('relu'), và Dropout (0.25) để giảm thiểu hiện tượng overfitting; lớp output có lớp Dense với 62 neuron, sử dụng Activation ('softmax') để phân loại thành 62 lớp. Mô hình được biên dịch với optimizer Adam có learning rate là 0.001.



Hình 3.1: Cấu trúc mô hình CNN cải tiến (Grayscale).

- **<u>Với mô hình xử lý ảnh màu:</u>**

Mô hình xử lý ảnh màu nhóm cũng sẽ biên dịch giống với mô hình xử lý ảnh xám, nhưng input đầu vào sẽ là (32,32,3): 32x32 vẫn là kích thước ảnh và thay đổi kênh màu thành 3 (RGB) là đỏ, lục, xanh.



Hình 3.2: Cấu trúc mô hình CNN cải tiến (RGB).

3.2.2 Mô hình học sâu DenseNet121

DenseNet121 là một kiến trúc mạng neural sử dụng các kết nối dày đặc (dense connections) giữa các tầng, giúp thông tin được truyền dễ dàng giữa các tầng. Mô hình có ít tham số hơn các kiến trúc khác nhờ sử dụng các kết nối dày đặc và có khả năng tái sử dụng đặc trưng tốt, giúp hiệu suất cao hơn.

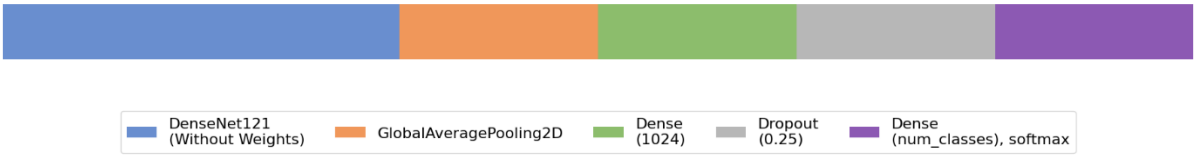
Mô hình DenseNet121 rất phù hợp cho bài toán nhận dạng văn bản trong ảnh vì: các kết nối dày đặc giúp thông tin được truyền dễ dàng từ các tầng thấp lên các tầng cao, tối ưu hóa việc trích xuất đặc trưng; khả năng tái sử dụng đặc trưng tốt giúp DenseNet121 có thể học được các đặc trưng quan trọng cho nhận dạng văn bản như các nét, cạnh, góc...; số lượng tham số ít giúp DenseNet121 có khả năng xác định văn bản tốt, hiệu suất cao trên các tập dữ liệu văn bản.

DenseNet121 là một lựa chọn rất phù hợp và hiệu quả cho bài toán nhận dạng văn bản trong ảnh, nhờ các đặc điểm nổi bật về kiến trúc và khả năng trích xuất đặc trưng. Nhóm loại bỏ các lớp đầu ra của DenseNet121 và chỉ giữ lại phần mô hình cơ sở và tinh chỉnh lại các siêu tham số để trích xuất đặc trưng.

- **<u>Với mô hình xử lý ảnh xám:</u>**

Nhóm đã thêm vào các lớp tùy chỉnh để thích ứng với bài toán nhận dạng văn bản, bao gồm: Lớp GlobalAveragePooling2D để thay thế Flatten nhằm giảm kích thước không gian của đầu ra từ phần convolutional base của DenseNet121,

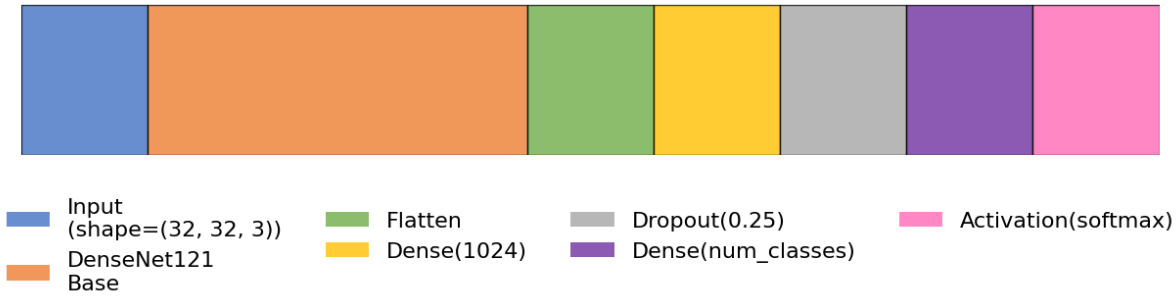
giúp giảm số lượng tham số và ngăn chặn overfitting; lớp Dense với 1024 neuron và hàm kích hoạt ReLU để học các đặc trưng phi tuyến tính từ đầu ra của lớp GlobalAveragePooling2D; lớp Dropout với tỷ lệ dropout 0.25 để giảm thiểu hiện tượng overfitting bằng cách ngẫu nhiên bỏ qua 25% các neurons trong quá trình huấn luyện; lớp đầu ra Dense với số lượng neuron bằng với số lượng lớp cần phân loại, sử dụng hàm kích hoạt softmax để trả về xác suất cho từng lớp. Mô hình đóng băng các lớp của mô hình DenseNet121 để không huấn luyện lại các trọng số của chúng trong quá trình fine-tuning.



Hình 3.3: Mô hình DenseNet121 cải tiến (Grayscale).

- **<u>Với mô hình xử lý ảnh màu:</u>**

Nhóm đã thêm vào các lớp tùy chỉnh để thích ứng với bài toán nhận dạng văn bản, bao gồm: lớp Flatten để chuyển đổi ma trận đặc trưng thành vector; lớp Dense với 1024 neuron và hàm kích hoạt ReLU; lớp Dropout với tỉ lệ dropout 0.25 để giảm thiểu hiện tượng overfitting; lớp đầu ra Dense với số lượng neuron bằng với số lượng lớp cần phân loại, sử dụng hàm kích hoạt softmax. Đóng băng các lớp của mô hình DenseNet121 để không huấn luyện lại các trọng số của chúng trong quá trình fine-tuning. Điều này giúp giảm thiểu thời gian huấn luyện và tận dụng tốt nhất các đặc trưng học được từ ImageNet. Mô hình được biên dịch với optimizer SGD có learning rate là 0.001 và momentum là 0.9.



Hình 3.4: Mô hình DenseNet121 cải tiến (RBG).

3.2.3 Mô hình học sâu VGG19

VGG19 là một kiến trúc mạng neural sử dụng nhiều lớp convolution và pooling để trích xuất đặc trưng từ ảnh. VGG19 có 19 lớp sâu, bao gồm 16 lớp convolution và 3 lớp fully connected, sử dụng các bộ lọc convolution nhỏ (3x3) nhưng nhiều lớp để tăng độ sâu của mạng. Mô hình đã được huấn luyện trên tập dữ liệu ImageNet rất lớn, có thể tái sử dụng cho các bài toán khác.

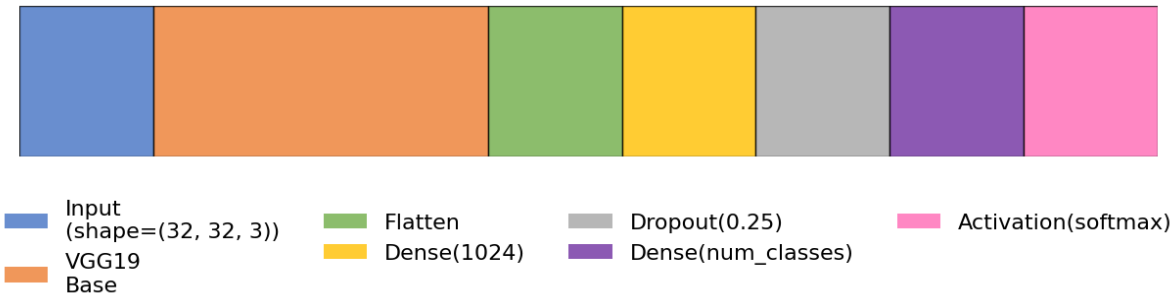
Mô hình VGG19 rất phù hợp để sử dụng vì tương thích với kiến trúc mạng convolution, có khả năng trích xuất các đặc trưng phức tạp như các nét, góc, khu vực văn bản... từ ảnh. VGG19 đã được huấn luyện trên tập dữ liệu lớn ImageNet, có thể tái sử dụng và fine-tune cho bài toán nhận dạng văn bản, có khả năng generalize tốt và hiệu suất cao trên nhiều tập dữ liệu khác.

VGG19 là một lựa chọn tuyệt vời cho bài toán nhận dạng văn bản trong ảnh, nhờ vào kiến trúc phù hợp, khả năng trích xuất đặc trưng mạnh mẽ và hiệu suất cao trên các tập dữ liệu. Mô hình VGG19 được tải về với các trọng số đã được huấn luyện trên bộ dữ liệu ImageNet. Luận văn loại bỏ các lớp đầu ra của VGG19 và chỉ giữ lại phần mô hình cơ sở và tinh chỉnh lại các siêu tham số để trích xuất đặc trưng.

VGG19 hoạt động tốt ở xử lý ảnh màu, ảnh xám cho ra kết quả không như mong muốn. Vì vậy, nhóm chỉ đề xuất mô hình VGG19 với xử lý ảnh màu.

Mô hình nhóm tinh chỉnh thêm vào các lớp tùy chỉnh để thích ứng với bài toán nhận dạng văn bản, bao gồm: lớp Flatten để chuyển đổi ma trận đặc trưng thành vector, lớp Dense với 1024 neuron và hàm kích hoạt ReLU, lớp Dropout với tỉ lệ dropout 0.25 để giảm thiểu hiện tượng overfitting, lớp đầu ra Dense với số lượng neuron bằng với số lượng lớp cần phân loại, sử dụng hàm kích hoạt

softmax. Nhóm thực hiện đóng băng các lớp của mô hình VGG19 để không huấn luyện lại các trọng số của chúng trong quá trình fine-tuning. Mô hình được biên dịch với optimizer SGD có learning rate là 0.001 và momentum là 0.9.



Hình 3.5: Mô hình VGG19 cải tiến.

3.3. Phương pháp đánh giá

3.3.1 Các phương pháp đánh giá liên quan

Phương pháp đánh giá mà nhóm đang sử dụng trong bài toán là categorical cross-entropy loss kết hợp với độ chính xác (accuracy). Đây là cách phổ biến để đánh giá các mô hình phân loại đa lớp (multi-class classification) như các mô hình đang huấn luyện.

Categorical Cross-Entropy Loss: Đây là một hàm mất mát (loss function) được lớp. sử dụng phổ biến trong các bài toán phân loại đa Hàm mất mát này đo lường sự khác biệt giữa phân phối xác suất dự đoán của mô hình và phân phối xác suất thực tế. Cụ thể, hàm mất mát này được định nghĩa như sau:

$$L = -\Sigma_1 y_i \log(\hat{y}i) \quad (7)$$

Trong đó:

- y_i: là nhãn thực tế (ground truth) cho lớp *i*.
- $\hat{y}i$: là xác suất dự đoán cho lớp *i*.

Accuracy: Đây là tỷ lệ giữa số lượng dự đoán đúng và tổng số mẫu dữ liệu. Độ chính xác là một chỉ số trực quan và dễ hiểu, cho biết tỷ lệ phần trăm dự đoán của mô hình là chính xác.

Mô hình được biên dịch với hàm mất mát là categorical_crossentropy và độ chính xác là một trong các chỉ số đánh giá (metrics). ModelCheckpoint để lưu lại trọng số mô hình tốt nhất dựa trên val_accuracy (độ chính xác trên tập validation).

Ngoài ra, bạn cũng sử dụng ReduceLROnPlateau để giảm tốc độ học khi hàm mất mát không cải thiện, giúp mô hình học tốt hơn trong quá trình huấn luyện.

Kết hợp các phương pháp này giúp bạn tối ưu hóa quá trình huấn luyện mô hình và cải thiện hiệu suất của mô hình trên dữ liệu kiểm tra.

3.3.2 Đánh giá mô hình

Đầu tiên, nhóm đánh giá mô hình trên tập huấn luyện để kiểm tra hiệu suất của mô hình trên dữ liệu mà nó đã được huấn luyện. Quá trình này giúp xác định khả năng học của mô hình từ dữ liệu huấn luyện.

Tiếp theo, mô hình được đánh giá trên tập xác nhận (validation set). Điều này cho phép kiểm tra khả năng của mô hình trong việc tổng quát hóa và dự đoán trên dữ liệu mới mà mô hình chưa từng thấy trong quá trình huấn luyện. Kết quả trên tập xác nhận cung cấp một chỉ số quan trọng để phát hiện và ngăn chặn overfitting.

Cuối cùng, đánh giá mô hình trên tập kiểm tra (test set). Đây là bước quan trọng để xác định hiệu suất thực sự của mô hình trên dữ liệu hoàn toàn mới. Tập kiểm tra không được sử dụng trong quá trình huấn luyện và xác nhận, do đó kết quả trên tập này phản ánh khả năng thực sự của mô hình trong môi trường thực tế. Sau khi đánh giá, mô hình hoàn chỉnh được lưu lại để sử dụng sau này. Việc lưu mô hình giúp dễ dàng tải lại và sử dụng mô hình mà không cần huấn luyện lại từ đầu.

Để hiểu rõ hơn về quá trình huấn luyện và hiệu suất của mô hình, nhóm vẽ biểu đồ loss và accuracy trên cả tập huấn luyện và tập xác nhận. Các biểu đồ này giúp nhận diện các vấn đề tiềm ẩn như overfitting hoặc underfitting.

3.3.3 Đánh giá kết quả trên tập dữ liệu văn bản

Độ chính xác ký tự (Character Accuracy) là một thước đo quan trọng để đánh giá hiệu suất của mô hình nhận dạng ký tự quang học (OCR). Độ chính xác ký tự được tính bằng cách so sánh các ký tự dự đoán với các ký tự thực tế và tính toán tỷ lệ ký tự đúng trên tổng số ký tự. Đánh giá độ chính xác của mô hình OCR là quá trình xác định mức độ chính xác của mô hình trong việc nhận dạng và dự đoán các ký tự từ hình ảnh.

Nhóm đã thực hiện hàm calculate_accuracy được sử dụng để tính toán độ chính xác ký tự bằng cách lặp qua các chuỗi ký tự dự đoán và thực tế, so sánh

từng ký tự tương ứng, và đếm số ký tự đúng. Cụ thể, công thức được tính như sau:

$$accuracy = \frac{correct\_chars}{total\_chars} \quad (8)$$

Trong đó:

- accuracy: độ chính xác mô hình
- correct_chars: tổng số ký tự được nhận dạng đúng.
- total_chars: tổng số ký tự trong tất cả các chuỗi nhãn thực tế

Phương pháp đánh giá này cung cấp một cách tiếp cận toàn diện để xác định hiệu suất của mô hình OCR trên tập dữ liệu kiểm tra thực tế. Độ chính xác ký tự là một chỉ số quan trọng để đánh giá mức độ chính xác của mô hình trong việc nhận dạng các ký tự riêng lẻ, trong khi các biểu đồ và phương pháp trực quan giúp kiểm tra và xác định các vấn đề tiềm ẩn trong quá trình huấn luyện và đánh giá mô hình.

Chương 4: THỰC NGHIỆM – ĐÁNH GIÁ KẾT QUẢ

4.1. Đọc dữ liệu và cấu hình phần cứng

4.1.1 Giới thiệu phần cứng

Google Colab là một nền tảng cung cấp môi trường phát triển và chạy mã Python trực tuyến với nhiều tiện ích hỗ trợ học máy và trí tuệ nhân tạo. Một trong những điểm mạnh của Google Colab là cung cấp khả năng sử dụng phần cứng mạnh mẽ, bao gồm cả GPU và TPU, giúp tăng tốc quá trình huấn luyện mô hình học sâu. Dưới đây là mô tả chi tiết về các tùy chọn phần cứng trên Google Colab:

- **CPU:** Môi trường mặc định của Google Colab là sử dụng CPU. Với CPU, có thể chạy các đoạn mã và huấn luyện các mô hình nhỏ hoặc thử nghiệm nhanh các ý tưởng. Tuy nhiên, khi làm việc với các mô hình lớn hoặc dữ liệu phức tạp, việc sử dụng GPU hoặc TPU sẽ mang lại hiệu suất tốt hơn.
- **GPU:** Google Colab cung cấp GPU miễn phí, bao gồm các loại như NVIDIA Tesla K80, T4, P4, và P100. GPU giúp tăng tốc đáng kể quá trình huấn luyện mô hình học sâu bằng cách thực hiện các phép tính song song trên hàng ngàn lõi.

4.1.2 Tập dữ liệu Chars74K

Tập dữ liệu Chars74K là một tập dữ liệu được phát triển để phục vụ nghiên cứu trong lĩnh vực nhận dạng ký tự quang học (OCR) và học sâu. Tập dữ liệu này được tạo ra bởi các nhà nghiên cứu tại Trung tâm Visvesvaraya về Công nghệ và Thông tin, Đại học Bangalore, Ấn Độ. Nó được giới thiệu lần đầu tiên trong bài báo "Character Recognition in Natural Images" của Thiagarajan Ravindran S và V. Balasubramanian vào năm 2009. Nguồn tài liệu: “The Chars74K image dataset - Character Recognition in Natural Images.” https://info-ee.surrey.ac.uk/CVSSP/demos/chars74k/.

Mục đích chính của việc phát triển Chars74K là để cung cấp một bộ dữ liệu đa dạng và phong phú phục vụ cho việc nghiên cứu và phát triển các hệ thống nhận dạng ký tự tự động trong các hình ảnh tự nhiên. Tập dữ liệu này đặc biệt

hữu ích trong việc huấn luyện và đánh giá các mô hình học máy và học sâu nhằm nâng cao độ chính xác trong nhận dạng ký tự.

Tập dữ liệu Chars74K bao gồm ba tập con chính:

- EnglishFnt (Font): Bao gồm 62 lớp ký tự tiếng Anh (chữ hoa, chữ thường, và số) được tạo ra bằng cách sử dụng các phông chữ khác nhau. Tập con này bao gồm tổng cộng 74.000 hình ảnh.
- EnglishImg (Image): Bao gồm 62 lớp ký tự tiếng Anh được trích xuất từ các hình ảnh tự nhiên. Tập con này bao gồm khoảng 64.000 hình ảnh, đại diện cho các ký tự được chụp từ các biển báo, áp phích, tài liệu viết tay và các nguồn khác.
- KannadaImg: Bao gồm các ký tự tiếng Kannada được trích xuất từ các hình ảnh tự nhiên, hỗ trợ nghiên cứu trong ngôn ngữ địa phương.

Tập dữ liệu bao gồm nhiều phông chữ, kích thước, màu sắc và điều kiện ánh sáng khác nhau, giúp cho việc huấn luyện các mô hình OCR trở nên mạnh mẽ và chính xác hơn. Chars74K đã được sử dụng rộng rãi trong các nghiên cứu và các bài báo khoa học về nhận dạng ký tự và học sâu, trở thành một nguồn dữ liệu quan trọng trong lĩnh vực này.

Tập dữ liệu Chars74K có thể được truy cập và tải xuống từ nhiều nguồn trên internet, bao gồm cả trang chủ của các tác giả hoặc các kho lưu trữ dữ liệu mở.

Hình 4.1: Hình ảnh minh họa thư mục chữ A.

Bộ dữ liệu này có chứa thêm tập nhãn. Một nhãn là một tệp văn bản tương ứng với một hình ảnh, nghĩa là cấu trúc và số lượng của tập nhãn giống tập hình ảnh. Bộ dữ liệu này chủ yếu tập trung vào việc nhận dạng các ký tự mà vốn rất

khó xử lý bằng các kỹ thuật của mã nguồn mở OCR. Với tập dữ liệu Chars74k này, luận văn sử dụng trực tiếp từ nguồn trên Internet.

Trong bài toán này, nhóm thực hiện chia lại tập dữ liệu thành 3 tập dữ liệu với 62 lớp: 6164 ảnh trong tập huấn luyện (train), 771 ảnh trong tập kiểm tra (test) và 770 ảnh trong tập xác thực (validation).

4.1.3 Tập dữ liệu IIIT 5K_coco

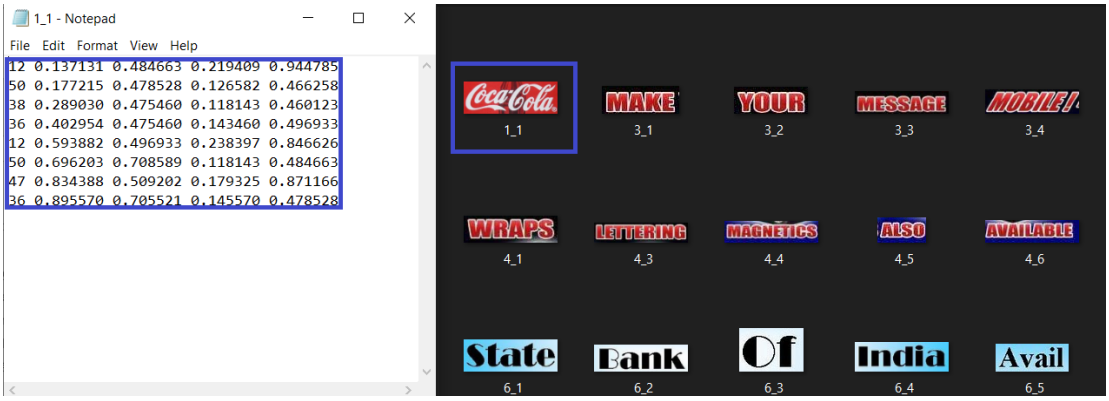
Tập dữ liệu IIIT5K-COCO là một tập dữ liệu được phát triển để phục vụ nghiên cứu trong lĩnh vực nhận dạng văn bản trong hình ảnh và học sâu. Tập dữ liệu này được tạo ra bởi các nhà nghiên cứu từ Viện Công nghệ Thông tin Ấn Độ, Hyderabad. Mục đích của tập dữ liệu này là kết hợp các đặc điểm của tập dữ liệu IIIT5K với các hình ảnh từ tập dữ liệu COCO để cung cấp một nguồn dữ liệu phong phú cho các bài toán nhận dạng văn bản. Tập dữ liệu IIIT5K-COCO có thể được truy cập và tải xuống từ kho lưu trữ GitHub. Nguồn tài liệu: Adumrewal, “GitHub - adumrewal/iiit-5k-word-coco-dataset: IIIT5K dataset converted to coco format along with python readable original label files. Original dataset is in matlab format, which might have been an issue for some potential users, hence this repository.,” *GitHub*. https://github.com/adumrewal/iiit-5k-word-coco-dataset.

Mục đích chính của việc phát triển IIIT5K-COCO là để cung cấp một bộ dữ liệu đa dạng và phong phú, hỗ trợ cho việc nghiên cứu và phát triển các hệ thống nhận dạng văn bản tự động trong các hình ảnh tự nhiên. Tập dữ liệu này đặc biệt hữu ích trong việc huấn luyện và đánh giá các mô hình học máy và học sâu, nhằm nâng cao độ chính xác trong nhận dạng văn bản.

Tập dữ liệu IIIT5K-COCO bao gồm các hình ảnh từ tập dữ liệu COCO (Common Objects in Context) được gán nhãn với văn bản từ tập dữ liệu IIIT5K. Các hình ảnh này chứa các từ được chụp trong các ngữ cảnh tự nhiên khác nhau, từ đó cung cấp các thách thức thực tế cho các mô hình nhận dạng văn bản.

Tập dữ liệu bao gồm các hình ảnh từ nhiều ngữ cảnh khác nhau, bao gồm các biển báo, áp phích, sách, và các tài liệu khác. IIIT5K-COCO đã được sử dụng rộng rãi trong các nghiên cứu và các bài báo khoa học về nhận dạng văn bản và học sâu, trở thành một nguồn dữ liệu quan trọng trong lĩnh vực này.

Hình 4.2: Hình ảnh minh họa dữ liệu IIIT 5K-coco.



Hình 4.3: Hình ảnh minh họa nhãn của một ảnh.

Nhãn được định dạng theo phong cách YOLO (You Only Look Once), một cách định nghĩa các bounding box cho các ký tự trong một bức ảnh. Mỗi dòng trong nhãn đại diện cho một bounding box cho một ký tự cụ thể. Cấu trúc của mỗi dòng như sau:

1. **Class ID (ID của lớp):** Con số đầu tiên trong dòng (ví dụ: 12, 50, 38, ...) là ID của lớp, tức là ký tự mà bounding box đang bao quanh. Ví dụ, 12 có thể đại diện cho một ký tự cụ thể (chẳng hạn như 'C' nếu hệ thống của bạn đã gán 'C' cho ID này).
2. **x_center (tọa độ tâm theo trục x):** Con số thứ hai (ví dụ: 0.137131, 0.177215, ...) là tọa độ x của tâm bounding box, được chuẩn hóa theo chiều rộng của hình ảnh. Tọa độ này là giá trị tương đối (từ 0 đến 1), tính từ mép trái của ảnh.
3. **y_center (tọa độ tâm theo trục y):** Con số thứ ba (ví dụ: 0.484663, 0.478528, ...) là tọa độ y của tâm bounding box, được chuẩn hóa theo chiều

cao của hình ảnh. Tọa độ này cũng là giá trị tương đối (từ 0 đến 1), tính từ mép trên của ảnh.

4. **width (chiều rộng của bounding box):** Con số thứ tư (ví dụ: 0.219409, 0.126582, ...) là chiều rộng của bounding box, được chuẩn hóa theo chiều rộng của hình ảnh.
5. **height (chiều cao của bounding box):** Con số cuối cùng (ví dụ: 0.944785, 0.466258, ...) là chiều cao của bounding box, được chuẩn hóa theo chiều cao của hình ảnh.

   Ví dụ: Dòng đầu tiên: 12 0.137131 0.484663 0.219409 0.944785

   - **12**: Class ID, xác định ký tự mà bounding box này đang bao quanh.
   - **0.137131**: Tọa độ x của tâm bounding box.
   - **0.484663**: Tọa độ y của tâm bounding box.
   - **0.219409**: Chiều rộng của bounding box (tương đối với chiều rộng của hình ảnh).
   - **0.944785**: Chiều cao của bounding box (tương đối với chiều cao của hình ảnh).

Các thông số này giúp định vị và xác định kích thước của bounding box, từ đó mô hình có thể dự đoán ký tự hoặc đối tượng được chứa trong hộp giới hạn đó.

Tổng tập dataset có 5000 ảnh bao gồm 2000 train và 3000 test. Sau đó, nhằm tránh tình trạng tràn bộ nhớ trên google colab, nhóm đã tiến hành phân chia lại tập dữ liệu theo tỉ lệ 4 - 3.4 -2.4 (40% train - 2000 ảnh, nhãn; 34% test - 1700 ảnh, nhãn; 24% val - 1300 ảnh, nhãn). Tiếp đó, nhóm tiến hành đó độ chính xác của mô hình trên tập test.

4.2. Kết quả thực nghiệm và đánh giá

4.2.1 Xử lý dữ liệu ảnh xám (Grayscale)

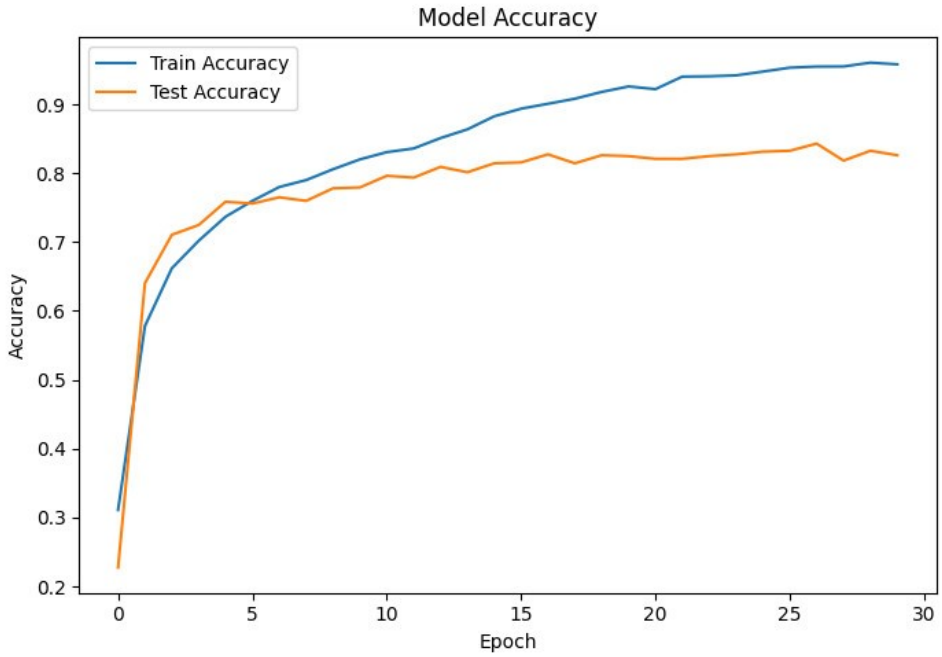
ImageDataGenerator là một công cụ mạnh mẽ trong Keras để tạo ra các batch ảnh từ thư mục, đồng thời hỗ trợ các kỹ thuật tăng cường dữ liệu (data augmentation).

Việc chuyển ảnh màu sang ảnh xám mang tới một vài lợi ích cho việc xác định văn bản trong ảnh, có thể kể đến như: giảm số kênh màu, tăng tốc độ xử lý

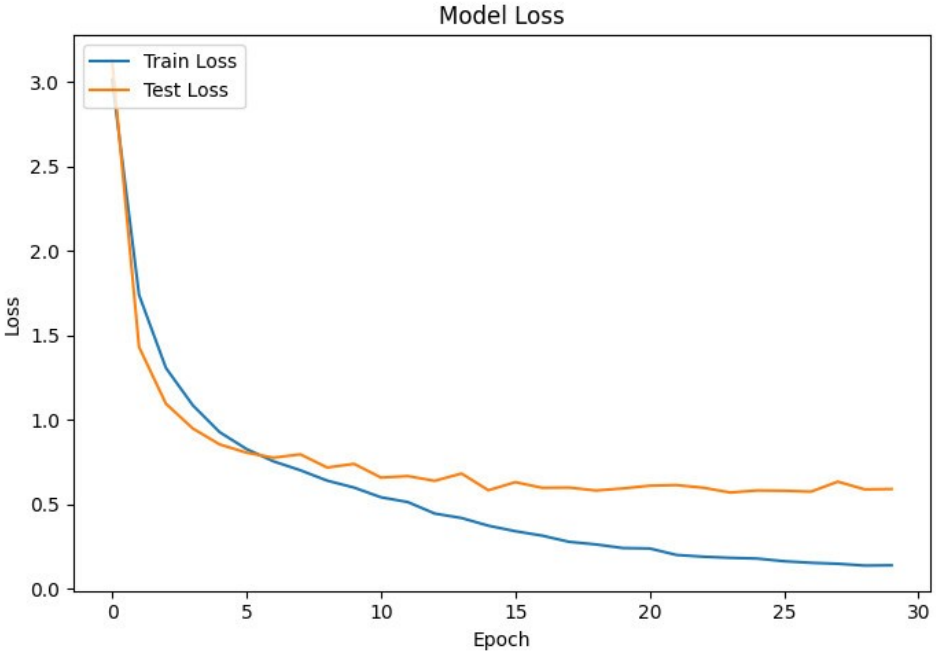
hình ảnh, giảm độ phức tạp của mô hình, và có khả năng giữ lại các đặc trưng quan trọng cần thiết cho việc xác định văn bản.

Mô hình VGG19 sẽ không xuất hiện ở mô hình xử lý dữ liệu ảnh xám vì kết quả cho ra không tốt như nhóm mong đợi.

1. **Mô hình CNN:**

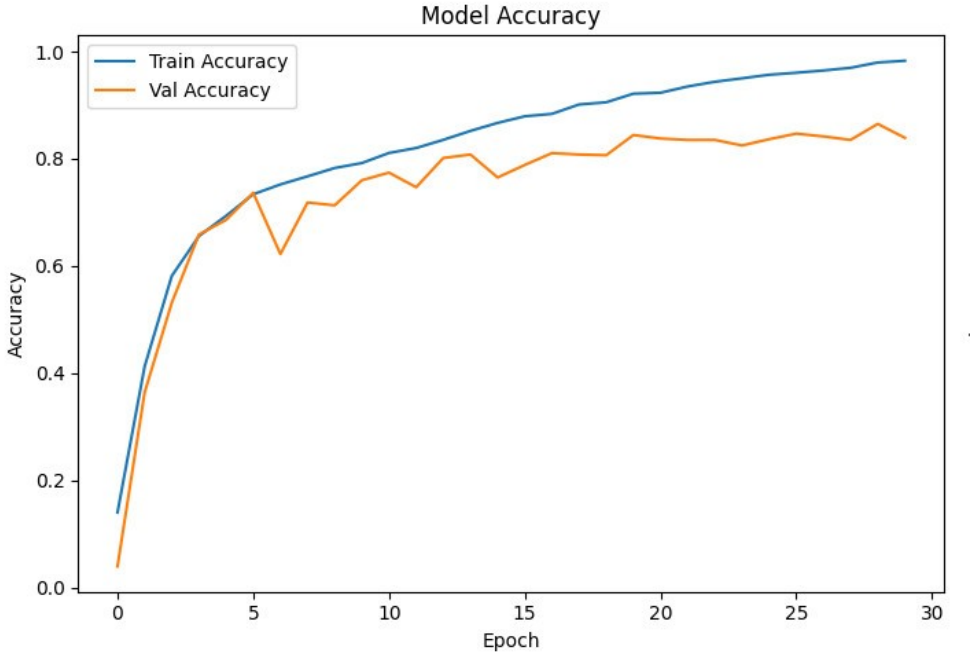


Hình 4.4: Kết quả hàm Accuracy khi huấn luyện mô hình CNN (Grayscale).

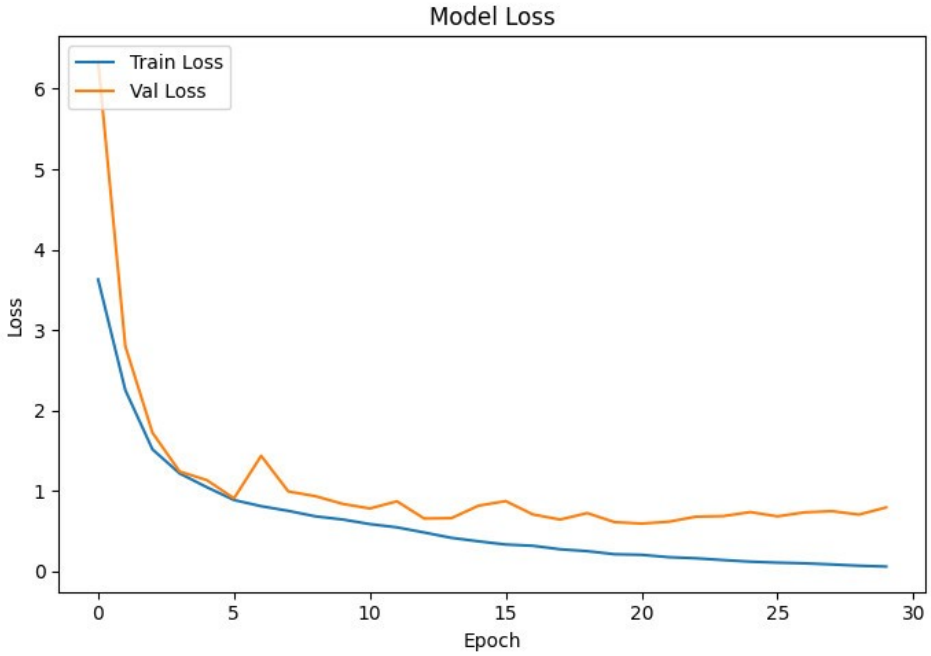


Hình 4.5: Kết quả hàm Loss khi huấn luyện mô hình CNN (Grayscale).

2. **Mô hình DenseNet121:**



Hình 4.6: Kết quả hàm Accuracy huấn luyện mô hình DenseNet121 (Grayscale).



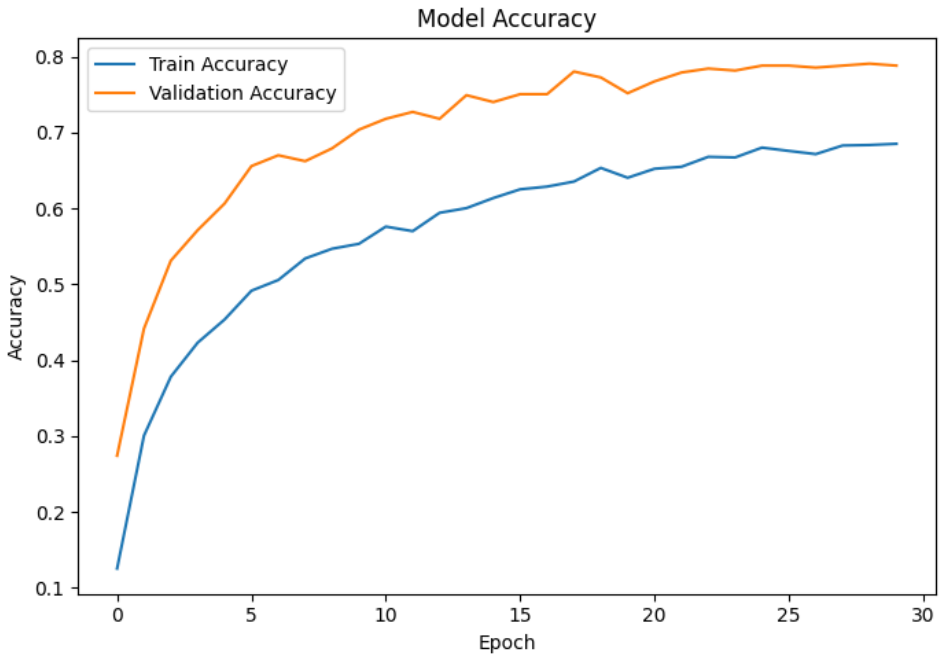
Hình 4.7: Kết quả hàm Loss khi huấn luyện mô hình DenseNet121 (Grayscale).

4.2.2 Xử lý dữ liệu ảnh màu (RGB)

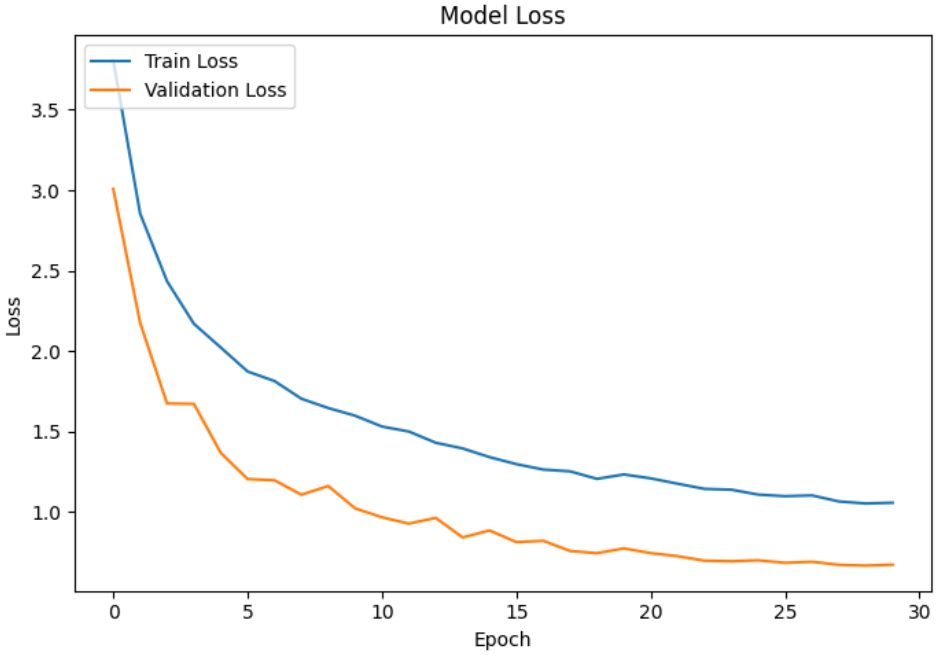
Ảnh được chuẩn bị dưới dạng màu (RGB), với các thông số cấu hình trong ImageDataGenerator. Việc sử dụng ảnh màu giúp đọc được thêm nhiều thông tin ảnh xám không thể đọc được, giúp mô hình nhận diện ký tự có thêm ngữ cảnh về màu sắc và độ sáng, có thể cải thiện độ chính xác của mô hình.

Sử dụng ảnh màu giúp mô hình có thể tận dụng thông tin màu sắc để cải thiện quá trình nhận dạng ký tự. Các phép biến đổi dữ liệu (augmentation) giúp tăng cường dữ liệu huấn luyện, làm cho mô hình mạnh mẽ hơn và ít bị overfitting.

1. **Mô hình CNN:**

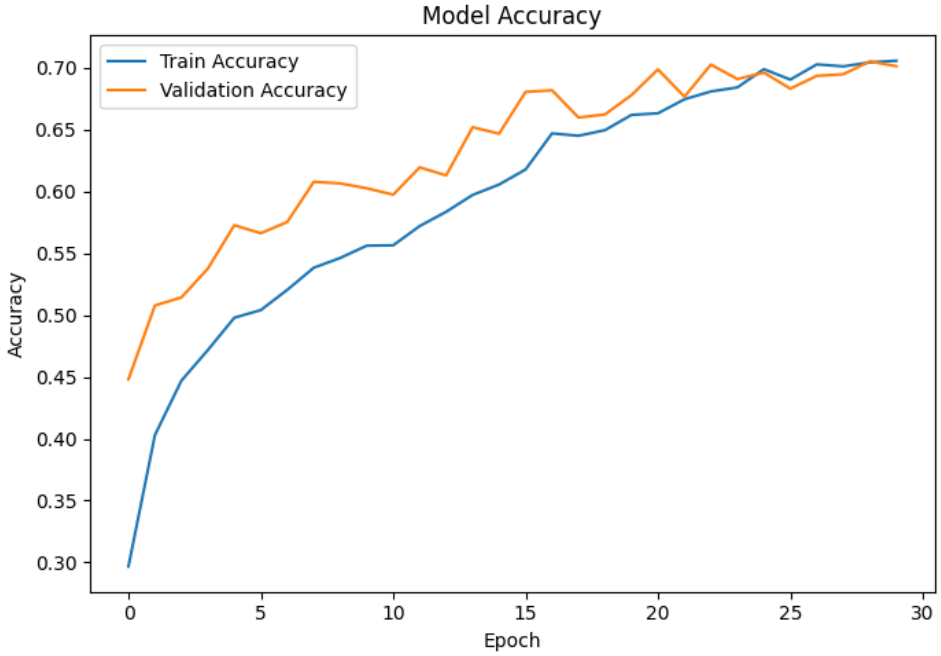


Hình 4.8: Kết quả hàm Accuracy khi huấn luyện mô hình CNN (RGB).

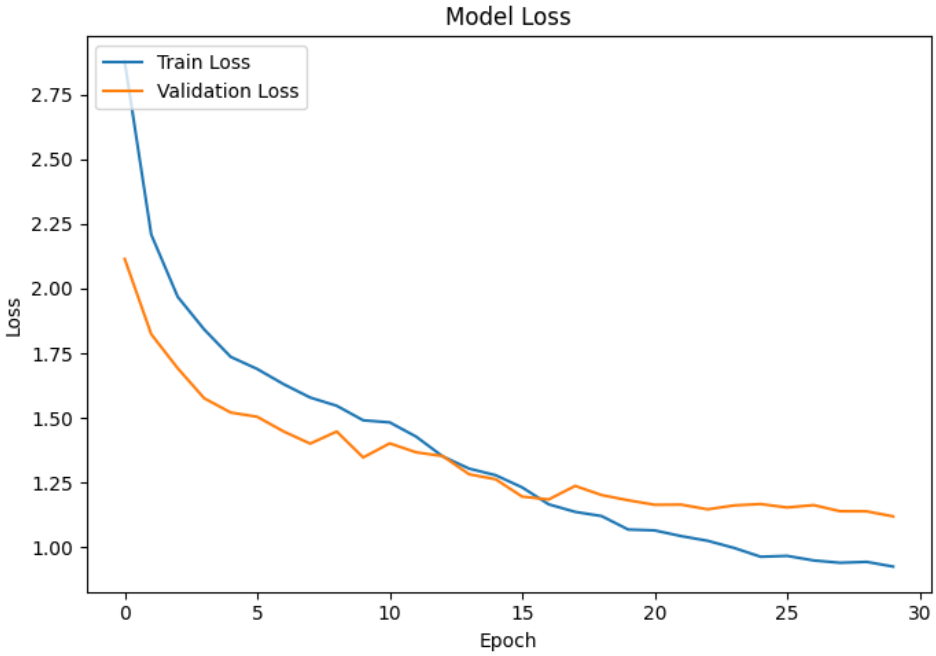


Hình 4.9: Kết quả hàm Loss khi huấn luyện mô hình CNN (RGB).

2. Mô hình DenseNet121:

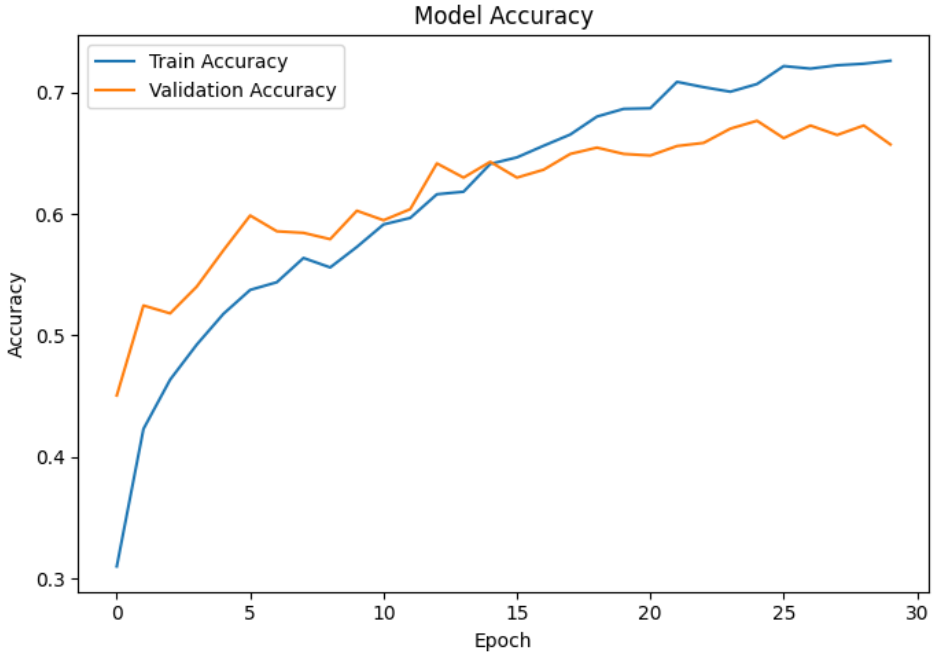


Hình 4.10: Kết quả hàm Accuracy khi huấn luyện mô hình DenseNet121 (RGB).

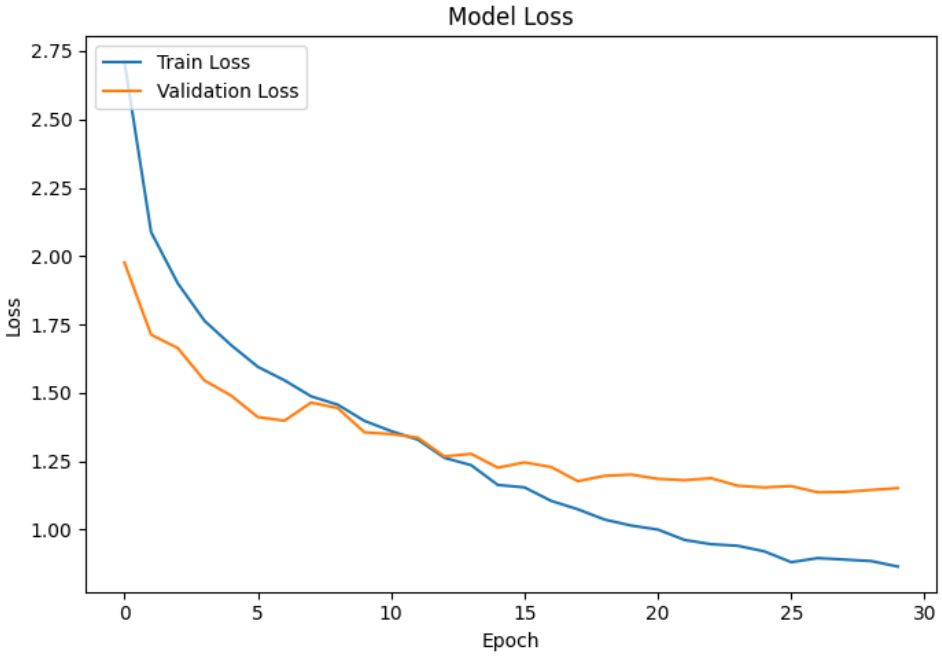


Hình 4.11: Kết quả hàm Loss khi huấn luyện mô hình DenseNet121 (RGB).

3. Mô hình VGG19:



Hình 4.12: Kết quả hàm Accuracy khi huấn luyện mô hình VGG19.



Hình 4.13: Kết quả hàm Loss khi huấn luyện mô hình VGG19.

4.2.3 Bảng kết quả hiện thực các mô hình

	CNN		VGG19		DENSENET121	
	Grayscale	RGB	Grayscale	RGB	Grayscale	RGB
Train Accuracy	96.05%	78.3%	40.75%	47.99%	98.58%	80.43%
Train loss	0.1343	0.7023	2.1310	1.8073	0.0472	0.6553
Validation Accuracy	82.6%	80.54%	4.86%	59.48%	83.9%	68.22%
Validation Loss	0.5893	0.6697	13.1669	1.4768	0.7901	1.1853
Độ chính xác trên tập dữ liệu IIIT5K	65.37%	61.68%	3.17%	35.89%	66.55%	45.52%

So sánh và phân tích các mô hình xử lý ảnh RGB và Grayscale

1. **CNN**

- **RGB:**

CNN cải tiến là một mô hình đơn giản nhưng hiệu quả, với kiến trúc gồm ba lớp convolutional kèm BatchNormalization, activation (ReLU), và MaxPooling. Mô hình đạt được độ chính xác huấn luyện (Train Accuracy) là 78.3% và độ chính xác kiểm tra (Validation Accuracy) là 80.54%. Tuy nhiên, độ chính xác trên tập dữ liệu IIIT5K chỉ đạt 61.68%, cho thấy mô hình có thể bị hạn chế khi xử lý dữ liệu phức tạp. Train loss và Validation loss cao cũng cho thấy mô hình có thể bị overfitting.

- **Grayscale:**

CNN cải tiến cho dữ liệu grayscale có kiến trúc tương tự như CNN cho RGB nhưng với ảnh đầu vào là grayscale (1 kênh). Mô hình này đạt được Train Accuracy là 96.05% và Validation Accuracy là 82.6%, độ chính xác trên tập dữ liệu IIIT5K là 65.37%. Mô hình sử dụng BatchNormalization giúp ổn định quá trình huấn luyện và Dropout để giảm overfitting. Tuy nhiên, Train loss và Validation loss vẫn cao hơn so với DenseNet121, cho thấy mô hình có thể bị overfitting.

2. DenseNet121

- **RGB:**

DenseNet121 là một mô hình mạnh mẽ với khả năng truyền thông tin và gradient tốt, giúp cải thiện hiệu suất huấn luyện. Mô hình này sử dụng các trọng số pre-trained từ ImageNet, giúp tăng tốc độ và hiệu quả huấn luyện. DenseNet121 đạt được Train Accuracy là 80.43% và Validation Accuracy là 68.22%, nhưng độ chính xác trên tập dữ liệu IIIT5K chỉ là 45.52%. Validation loss cao cho thấy mô hình chưa tối ưu hóa tốt cho dữ liệu mới.

- **Grayscale:**

DenseNet121 cho dữ liệu grayscale là mô hình đạt hiệu suất cao nhất với Train Accuracy là 98.58%, Validation Accuracy là 83.9% và độ chính xác trên tập dữ liệu IIIT5K là 66.55%. Kiến trúc DenseNet cho phép truyền thông tin hiệu quả qua các lớp, giúp cải thiện độ chính xác và giảm overfitting. Mô hình này không sử dụng các trọng số pre-trained từ ImageNet, mà huấn luyện từ đầu, giúp nó thích nghi tốt hơn với dữ liệu cụ thể của bài toán. Sử dụng GlobalAveragePooling2D giúp giảm số lượng tham so và tránh overfitting tốt hơn. Dropout giúp giảm overfitting, cải thiện khả năng tổng quát hóa của mô hình.

3. VGG19

- **RGB:**

VGG19 là mô hình phổ biến và mạnh mẽ trong lĩnh vực nhận dạng ảnh. Tuy nhiên, mô hình này có Train Accuracy và Validation Accuracy

lần lượt chỉ đạt 47.99% và 59.48%. Độ chính xác trên tập dữ liệu IIIT5K thấp nhất trong ba mô hình, chỉ đạt 35.89%, cho thấy VGG19 chưa học tốt từ dữ liệu và hiệu suất chưa cao.

4. Kết Luận Lý do chọn CNN (Grayscale)

CNN cho dữ liệu grayscale là mô hình tốt nhất vì các lý do sau:

- **Độ chính xác cao:** Mô hình đạt Train Accuracy 96.05% và Validation Accuracy 82.6%, cho thấy khả năng học tốt từ dữ liệu và khả năng tổng quát hóa mạnh mẽ.
- **Độ chính xác trên tập dữ liệu thực tế:** Mô hình đạt độ chính xác 65.37% có mức ổn định giữa ảnh màu và ảnh xám, chứng tỏ tính tổng quát hóa tốt và hiệu quả trong việc nhận dạng văn bản trên dữ liệu thực tế.
- **Giảm overfitting:** Việc sử dụng BatchNormalization và Dropout giúp giảm overfitting, đảm bảo rằng mô hình không chỉ học thuộc lòng mà còn có khả năng dự đoán chính xác trên dữ liệu mới.
- **Kiến trúc đơn giản nhưng hiệu quả:** Với kiến trúc gồm ba lớp convolutional đơn giản nhưng mạnh mẽ, mô hình CNN grayscale tối ưu hóa được độ chính xác trong khi vẫn duy trì hiệu suất tính toán.

Dựa trên các phân tích và số liệu cụ thể, CNN (Grayscale) là lựa chọn tốt nhất cho bài toán nhận dạng văn bản nhờ vào hiệu suất và độ chính xác vượt trội so với các mô hình khác.

Chương 5: KẾT LUẬN

5.1. Kết quả đạt được

Mô hình CNN cải tiến đã đạt được độ chính xác cao nhất trên cả tập dữ liệu huấn luyện và kiểm tra, đặc biệt là khi xử lý ảnh xám. Điều này chứng tỏ CNN cải tiến rất phù hợp với dữ liệu văn bản trong ảnh xám. Đặc biệt, mô hình này cũng đạt độ chính xác cao nhất trên tập dữ liệu IIIT5K, với 65.37% cho ảnh xám và 61.68% cho ảnh màu. Trong khi đó, mô hình VGG19 có độ chính xác thấp nhất trong số ba mô hình, đặc biệt là khi xử lý ảnh xám. Điều này có thể do VGG19 là một mô hình khá phức tạp và cần nhiều dữ liệu hơn để huấn luyện tốt. Tuy nhiên, VGG19 đã cải thiện đáng kể khi xử lý ảnh màu, đạt độ chính xác 35.89% trên tập dữ liệu IIIT5K. Mô hình DenseNet121 cũng có hiệu suất huấn luyện tốt, đặc biệt là trên ảnh xám với độ chính xác huấn luyện đạt 98.58%, cho thấy khả năng mạnh mẽ của DenseNet121 trong việc học các đặc trưng từ dữ liệu văn bản trong ảnh xám. Trên tập kiểm tra và tập dữ liệu IIIT5K, DenseNet121 cũng đạt độ chính xác khá cao, chỉ xếp sau CNN cải tiến.

Mô hình CNN cải tiến là mô hình có hiệu suất tốt nhất trên cả tập huấn luyện, tập kiểm tra và tập dữ liệu IIIT5K, đặc biệt là với ảnh xám. DenseNet121 cũng có hiệu suất khá cao, chỉ xếp sau CNN cải tiến, đặc biệt là với ảnh xám. Trong khi đó, VGG19 có hiệu suất thấp nhất nhưng đã có cải thiện đáng kể khi tinh chỉnh mô hình.

5.2. Ưu – Nhược điểm của phương pháp đề xuất

5.2.1 Ưu điểm:

- **Tính đa dạng của mô hình:**
    - CNN cải tiến: Dễ dàng tùy chỉnh và tối ưu hóa cho nhiệm vụ cụ thể. Nó có thể phù hợp hơn với dữ liệu của bạn nếu được thiết kế tốt.
    - DenseNet121: Có khả năng chia sẻ các đặc trưng rất hiệu quả nhờ các kết nối dày đặc giữa các lớp, giúp giảm thiểu vấn đề vanishing gradient và cải thiện quá trình học.
    - VGG19: Được biết đến với cấu trúc đơn giản và hiệu quả cao trong nhiều nhiệm vụ nhận dạng hình ảnh, dễ dàng áp dụng và tinh chỉnh.

- **Sử dụng nhiều mô hình:** Việc sử dụng nhiều mô hình giúp tăng khả năng chính xác và độ tin cậy của kết quả nhận dạng bằng cách kết hợp các đặc trưng học được từ các mô hình khác nhau.
- **Nhận dạng ảnh màu và ảnh xám:** Phương pháp này linh hoạt khi làm việc với cả ảnh màu và ảnh xám, phù hợp với nhiều loại dữ liệu đầu vào.
- **Tận dụng mô hình đã huấn luyện:** Việc sử dụng các mô hình đã được huấn luyện trước giúp giảm thời gian và tài nguyên cần thiết cho việc huấn luyện từ đầu.

5.2.2 Khuyết điểm:

- **Tốn nhiều tài nguyên**: Việc sử dụng nhiều mô hình khác nhau có thể đòi hỏi nhiều tài nguyên tính toán hơn, đặc biệt là khi xử lý một lượng lớn dữ liệu.
- **Phức tạp trong việc quản lý và triển khai**: Sử dụng và quản lý nhiều mô hình cùng lúc có thể phức tạp hơn, đòi hỏi sự phối hợp tốt và kỹ năng trong việc triển khai.
- **Khả năng tổng hợp kết quả**: Cần có phương pháp tốt để tổng hợp kết quả từ các mô hình khác nhau một cách hiệu quả. Điều này có thể đòi hỏi thêm các bước xử lý hậu kỳ và có thể làm tăng độ phức tạp của hệ thống.
- **Khả năng overfitting**: Nếu không được quản lý tốt, việc huấn luyện và tinh chỉnh nhiều mô hình có thể dẫn đến overfitting, đặc biệt là khi dữ liệu huấn luyện hạn chế.

5.3. Hướng mở rộng tương lai

Mở rộng tập dữ liệu bằng việc thu thập thêm nhiều ảnh từ các nguồn khác nhau như sách báo, tư liệu, biển hiệu đường phố, và các ảnh chụp thực tế. Dữ liệu đa dạng giúp mô hình nhận diện văn bản đáp ứng các điều kiện khác nhau về độ tương phản, góc chụp, và chất lượng ảnh khác nhau.

Khám phá và triển khai các kỹ thuật mô hình kết hợp để kết hợp các mô hình khác nhau như CNN cải tiến, DenseNet121 và VGG19. Mô hình kết hợp có thể sử dụng phương pháp voting, averaging hoặc stacking để tổng hợp kết quả dự đoán từ các mô hình riêng lẻ, từ đó nâng cao độ chính xác và độ tin cậy của hệ thống.

Áp dụng kỹ thuật học sâu chuyển dịch từ các mô hình đã được huấn luyện trên các tập dữ liệu lớn khác để cải thiện khả năng nhận dạng. Việc sử dụng các mô hình pre-trained như Transformer hoặc các mô hình mới như Vision Transformers (ViT) có thể mang lại hiệu quả cao hơn trong việc nhận dạng văn bản trong ảnh.

Phát triển mô hình học đa nhiệm để nhận dạng không chỉ văn bản mà còn các đặc trưng khác của hình ảnh như phong cách viết, màu sắc, và các đối tượng liên quan khác. Điều này giúp mô hình trở nên toàn diện hơn và có thể áp dụng cho nhiều loại dữ liệu khác nhau.

Khám phá việc áp dụng học tăng cường để tối ưu hóa quá trình nhận dạng văn bản. Học tăng cường có thể giúp mô hình học cách điều chỉnh các tham số và chiến lược nhận dạng để đạt được hiệu suất tốt nhất trên các tập dữ liệu đa dạng.

Nghiên cứu các phương pháp tối ưu hóa hiệu suất để giảm thời gian huấn luyện và dự đoán, như sử dụng phần cứng chuyên dụng (GPU, TPU) và các kỹ thuật tối ưu hóa mô hình (quantization, pruning).

Xây dựng hệ thống triển khai linh hoạt và hiệu quả, đảm bảo rằng mô hình có thể hoạt động tốt trong môi trường thực tế với yêu cầu về tốc độ và tài nguyên.

Theo dõi và tích hợp các công nghệ mới như Augmented Reality (AR) và Virtual Reality (VR) để cung cấp các giải pháp nhận dạng văn bản trong các ứng dụng thực tế tăng cường và thực tế ảo.

Nghiên cứu và ứng dụng các công nghệ xử lý ngôn ngữ tự nhiên (NLP) để cải thiện khả năng hiểu và xử lý văn bản sau khi nhận dạng.

TÀI LIỆU THAM KHẢO

[1] GeeksforGeeks, “Text Detection and Extraction using OpenCV and OCR," *GeeksforGeeks*, Jul. 30, 2024. https://www.geeksforgeeks.org/text-detection-and-extraction-using-opencv-and-ocr/ (mở lần cuối 2/8/2024)

[2] GeeksforGeeks, “ML Transfer Learning with Convolutional Neural Networks,” *GeeksforGeeks*, Mar. 20, 2024. https://www.geeksforgeeks.org/ml-transfer-learning-with-convolutional-neural-networks/?itm_source=auth&itm_medium=contributions&itm_campaign=articles (mở lần cuối 2/8/2024)

[3] Adumrewal, “GitHub - adumrewal/iiit-5k-word-coco-dataset: IIIT5K dataset converted to coco format along with python readable original label files. Original dataset is in matlab format, which might have been an issue for some potential users, hence this repository.,” *GitHub*. https://github.com/adumrewal/iiit-5k-word-coco-dataset (mở lần cuối 2/8/2024)

[4] M. N. Huy, “Công nghệ xử lý hình ảnh và chuyển đổi hì ảnh thành văn bản,” *Viblo*, Aug. 05, 2024. https://viblo.asia/p/cong-nghe-xu-ly-hinh-anh-va-chuyen-doi-hinh-anh-thanh-van-ban-5OXLAYbZLGr (mở lần cuối 2/8/2024)

[5] P. Đ. Khánh, “Khoa học dữ liệu,” *Khanh’s Blog*, Aug. 13, 2020. https://phamdinhkhanh.github.io/2020/08/13/ModelMetric.html (mở lần cuối 2/8/2024)

[6] Rizky, A. F., Yudistira, N., & Santoso, E. (n.d.). *"Text recognition on images using pre-trained CNN*." Arxiv.org. https://arxiv.org/pdf/2302.05105 (mở lần cuối 2/8/2024)

[7] Wydyanto, W., Mat Nayan, N., Sulaiman, R., Dewi, D. A., & Kurniawan,

T. B. (2024). A hybrid approach to detect and identify text in picture. *"Emerging Science Journal"*, *8*(1), 218–238. https://doi.org/10.28991/esj-2024-08-01-016 (mở lần cuối 2/8/2024)

[8] Mathur, G., & Rikhari, M. S. (2017). Text Detection in Document Images: Highlight on using FAST algorithm. *"International Journal of Advanced Engineering Research and Science*, *4"*(3), 275–284. https://doi.org/10.22161/ijaers.4.3.43 (mở lần cuối 2/8/2024)

[9] Coates, A., Carpenter, B., Case, C., Satheesh, S., Suresh, B.,Wang, T., Ng, A. Y. (n.d.). *"Text detection and character recognition in scene images with unsupervised feature learning"*. Stanford.edu. https://cs.stanford.edu/~cbcase/papers/CoatesEtAl-icdar11.pdf (mở lần cuối 2/8/2024)

[10] Kotappa Y G, Krushika M, M Ravichandra, & Pranitha, M. (2022). A review paper on computer vision and image processing. *"International Journal of Advanced Research in Science, Communication and Technology"*, 68–72. https://doi.org/10.48175/ijarsct-2822 (mở lần cuối 2/8/2024)

[11] Huang, G., Liu, Z., & van der Maaten, L. (n.d.). *"Densely connected convolutional networks"*. Arxiv.org. http://arxiv.org/abs/1608.06993 (mở lần cuối 2/8/2024)