**ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**A yellow and red sign with a red circle and a star

Description automatically generated**

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN: KHAI THÁC DỮ LIỆU VÀ ỨNG DỤNG**

**ĐỀ TÀI:** **XÂY DỰNG HỆ THỐNG NHẬN DẠNG BIỂN BÁO GIAO THÔNG BẰNG MẠNG NƠ-RON**

|  |  |  |
| --- | --- | --- |
| **Giảng viên hướng dẫn** | : | **ThS. Lê Thị Thuỷ** |
| **Lớp** | : | 20244IT6075001 |
| **Nhóm** | : | 7 |
| **Sinh viên thực hiện** | : | 1. Nguyễn Tuấn Minh - 2022607114 |
|  |  | 1. Trần Văn Nhã - 2022603089 |
|  |  | 1. Nguyễn Hoàng Tùng - 2022602302 |

**Hà Nội, tháng 7 năm 2025**

**ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**A yellow and red sign with a red circle and a star

Description automatically generated**

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN: KHAI THÁC DỮ LIỆU VÀ ỨNG DỤNG**

**ĐỀ TÀI:** **XÂY DỰNG HỆ THỐNG NHẬN DẠNG BIỂN BÁO GIAO THÔNG BẰNG MẠNG NƠ-RON**

|  |  |  |
| --- | --- | --- |
| **Giảng viên hướng dẫn** | : | **ThS. Lê Thị Thuỷ** |
| **Lớp** | : | 20244IT6075001 |
| **Nhóm** | : | 7 |
| **Sinh viên thực hiện** | : | 1. Nguyễn Tuấn Minh - 2022607114 |
|  |  | 1. Trần Văn Nhã - |
|  |  | 1. Nguyễn Hoàng Tùng - 2022602302 |

**Hà Nội, tháng 6 năm 2025**

PHIẾU HỌC TẬP CÁ NHÂN/NHÓM

**I. Thông tin chung**

Tên lớp học phần: 20244IT6075001 Khóa K17

Họ và tên thành viên trong nhóm:

1. Nguyễn Tuấn Minh
2. Trần Văn Nhã
3. Nguyễn Hoàng Tùng

Tên nhóm: Nhóm 7

**II.** **Nội dung học tập**

1. Tên chủ đề: Xây dựng hệ thống nhận dạng biển báo giao thộng bằng mạng nơ-ron
2. Hoạt động của sinh viên

* Hoạt động/Nội dung 1: *Khảo sát và phát biểu bài toán nhận diện da tổn thương..*

Mục tiêu/chuẩn đầu ra: *L2.*

* Hoạt động/Nội dung 2: *Nghiên cứu, mô tả một số kỹ thuật chính hiện có cho bài toán nhận diện da tổn thương.*

Mục tiêu/chuẩn đầu ra: *L2*

* Hoạt động/Nội dung 3: *Mô tả công nghệ nhận diện da tổn thương bằng CNNs và tiến hành thực nghiệm, phân tích, đánh giá kết quả thực nghiệm.*

Mục tiêu/chuẩn đầu ra: *L2*

* Hoạt động/Nội dung 4: *Viết báo cáo kỹ thuật.*

Mục tiêu/chuẩn đầu ra: L3

* Hoạt động/Nội dung 4: *Viết báo cáo phi kỹ thuật.*

Mục tiêu/chuẩn đầu ra: L4

* Hoạt động/Nội dung 4: *Thuyết trình kết quả.*

Mục tiêu/chuẩn đầu ra: L5

1. Sản phẩm nghiên cứu

* Báo cáo đúng theo mẫu quy định.
* Mã nguồn chương trình (nếu có).

**III. Nhiệm vụ học tập**

1. Hoàn thành toàn bộ nội dung được giao theo đúng thời gian quy định*.*
2. Trình bày sản phẩm nghiên cứu/thực nghiệm trước giảng viên và các sinh viên khác.

**IV. Học liệu sử dụng**

1. Tài liệu học tập:
2. Phương tiện, nguyên liệu thực hiện bài tập lớn: Máy tính cá nhân có kết nối Internet.

**KẾ HOẠCH THỰC HIỆN BÀI TẬP LỚN**

Tên lớp: 20244IT6075001. Khóa: K17.

Họ và tên sinh viên:

1. Nguyễn Tuấn Minh
2. Nguyễn Mạnh Niên
3. Nguyễn Hoàng Tùng

Tên nhóm: Nhóm 7

Tên chủ đề: Xây dựng hệ thống nhận dạng biển báo giao thông sử dụng mạng nơ ron.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tuần** | **Người thực hiện** | **Nội dung công việc** | **Kết quả đạt được** | **Phương pháp thực hiện** |
|  |  |  |  |  |

Ngày tháng năm 2025

**XÁC NHẬN CỦA GIẢNG VIÊN**

(Kí, ghi rõ họ tên)

Lê Thị Thuỷ

**BÁO CÁO HOẠT ĐỘNG NHÓM**

Tên lớp: 20244IT6075001. Khóa: K17.

Họ và tên sinh viên:

1. Nguyễn Tuấn Minh
2. Trần Văn Nhã
3. Nguyễn Hoàng Tùng

Tên nhóm: Nhóm 7

Tên chủ đề: Xây dựng hệ thống nhận dạng biển báo giao thông bằng mạng nơ-ron.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tuần** | **Người thực hiện** | **Nội dung công việc** | **Kết quả đạt được** | **Kiến nghị với giảng viên hướng dẫn** |
|  |  |  |  |  |

Ngày tháng năm 2025

**XÁC NHẬN CỦA GIẢNG VIÊN**

(Kí, ghi rõ họ tên)

Lê Thị Thuỷ

# **MỤC LỤC**

[MỤC LỤC 5](#_Toc1602904165)

[DANH MỤC HÌNH ẢNH 6](#_Toc1758026261)

[DANH MỤC BẢNG BIỂU 7](#_Toc2074933692)

[LỜI CẢM ƠN 8](#_Toc1999006971)

[LỜI NÓI ĐẦU 9](#_Toc1335321184)

[CHƯƠNG 1. TỔNG QUAN VỀ KHO DỮ LIỆU VÀ PHƯƠNG PHÁP KHAI PHÁ 11](#_Toc1720758073)

[1.1. Kiến thức tổng quan 12](#_Toc412841957)

[1.1.1. Trí tuệ nhân tạo 12](#_Toc91810052)

[1.1.2. Học máy 14](#_Toc2043271603)

[1.1.3. Xử lý ảnh số và thị giác máy tính 17](#_Toc1278976541)

[1.1.4. Mạng nơ-ron nhân tạo 20](#_Toc1605243831)

[1.1.5. Mạng tích chập 23](#_Toc614937758)

[1.2. Bài toán nhận diện 25](#_Toc190467196)

[1.2.1. Phương pháp nhận diện truyền thống 25](#_Toc1758483963)

[1.2.2. Phương pháp nhận diện ứng dụng công nghệ hiện đại 26](#_Toc1319145139)

[1.2.3. Lợi ích và tiềm năng 27](#_Toc45230360)

[1.3. Bài toán nhận dạng biển báo giao thông 29](#_Toc1269737818)

[1.3.1. Giới thiệu chung 29](#_Toc329967021)

[1.3.2. Mục đích và ý nghĩa của đề tài 30](#_Toc204712145)

[1.3.3. Đầu vào và đầu ra của bài toán 31](#_Toc597891915)

[1.3.4. Phạm vi của bài toán 32](#_Toc1480000262)

[1.3.5. Những thuận lợi, khó khăn 32](#_Toc287855229)

[1.3.6. Miền ứng dụng 33](#_Toc906488408)

[CHƯƠNG 2. CÁC PHƯƠNG PHÁP KHAI PHÁ DỮ LIỆU 34](#_Toc1369071890)

[2.1. Một số kỹ thuật xử lý ảnh 34](#_Toc797945344)

[2.1.1. Khái niệm cơ bản 34](#_Toc957101607)

[2.1.2. Các kỹ thuật xử lý ảnh 36](#_Toc654955444)

[2.2. Trích chọn và biểu diễn đặc trưng hình ảnh 38](#_Toc1935044777)

[2.2.1. Đặc trưng ảnh 38](#_Toc550548004)

[2.2.2. Một số phương pháp trích chọn đơn giản 39](#_Toc1983931452)

[2.3. Phương pháp phân đoạn ảnh 39](#_Toc2110542079)

[2.4. Mạng nơ-ron truyền thẳng 41](#_Toc322134165)

[2.4.1. Giới thiệu 42](#_Toc358524549)

[2.4.2. Cấu trúc 42](#_Toc122095383)

[2.4.3. Ưu điểm 45](#_Toc2058569121)

[2.4.4. Nhược điểm 45](#_Toc1837527413)

[2.4.5. Các kiến trúc MLP phổ biến 45](#_Toc369062056)

[2.5. Mạng nơ-ron hồi quy 46](#_Toc453871508)

[2.5.1. Giới thiệu 46](#_Toc402022522)

[2.5.2. Cấu trúc 46](#_Toc1920000568)

[2.5.3. Phân loại 47](#_Toc1914621982)

[2.5.4. Ưu điểm 48](#_Toc1526207169)

[2.5.5. Nhược điểm 48](#_Toc1568572833)

[2.5.6. Các kiến trúc RNN phổ biến 48](#_Toc1361555721)

[2.6. Mạng nơ-ron tích chập 49](#_Toc1621104623)

[2.6.1. Giới thiệu 49](#_Toc1352304098)

[2.6.2. Cấu trúc 49](#_Toc1017573829)

[2.6.3. Siêu tham số 53](#_Toc6818308)

[2.6.4. Các hàm kích hoạt thường gặp 57](#_Toc254254763)

[2.6.5. Ưu điểm 58](#_Toc2057988190)

[2.6.6. Nhược điểm 59](#_Toc2018736027)

[2.6.7. Các kiến trúc CNN phổ biến 60](#_Toc15255304)

[2.7. Vision Transformer (ViT) 61](#_Toc1608828005)

[2.7.1. Giới thiệu 61](#_Toc1800474627)

[2.7.2. Cấu trúc 61](#_Toc2102036046)

[2.7.3. Ưu điểm 63](#_Toc1838634005)

[2.7.4. Nhược điểm 64](#_Toc768890821)

[2.7.5. Các kiến trúc ViT phổ biến 64](#_Toc441918035)

[CHƯƠNG 3. THỰC NGHIỆM 65](#_Toc778415277)

[3.1. Kỹ thuật sử dụng 66](#_Toc551073825)

[3.2. Môi trường thực nghiệm 66](#_Toc807558154)

[3.3. Dữ liệu thực nghiệm 66](#_Toc566050621)

[3.3.1. Nguồn gốc và đặc điểm của bộ dữ liệu 66](#_Toc1078643971)

[3.3.2. Tiền xử lý dữ liệu 66](#_Toc1400736128)

[3.4. Xây dựng mô hình nơ-ron tích chập 66](#_Toc1037592502)

[3.4.1. Kiến trúc mạng nơ-rơn tích chập 66](#_Toc6657615)

[3.4.2. Tham số huấn luyện 66](#_Toc1198474811)

[3.4.3. Đánh giá hiệu suất mô hình 67](#_Toc1827371941)

[3.4.4. Lưu trữ mô hình 67](#_Toc800851081)

[CHƯƠNG 4. XÂY DỰNG ỨNG DỤNG CHO MÔ HÌNH NHẬN DẠNG BIỂN BÁO GIAO THÔNG 67](#_Toc2039476257)

[4.1. Giới thiệu chung 68](#_Toc103119387)

[4.2. Kiến trúc hệ thống 68](#_Toc1021180810)

[4.3. Các chức năng chính của hệ thống 68](#_Toc871571098)

[4.4. Kết quả và đánh giá 68](#_Toc1204809641)

[4.5. Hướng phát triển trong tương lai 68](#_Toc265793159)

[KẾT LUẬN 68](#_Toc137704770)

[TÀI LIỆU THAM KHẢO 69](#_Toc807392702)

DANH MỤC HÌNH ẢNH

[Hình 1.1. Quá trình nhận dạng đối tượng (Nguồn: Internet) 7](#_Toc201126683)

[Hình 1.2. Minh hoạ về phương pháp nhận diện da tổn thương qua ảnh (Nguồn: Internet) 14](#_Toc201126684)

[Hình 2.1. Ảnh ký tự chữ A và ma trận số của vùng chọn (Nguồn: Internet) 20](#_Toc201126685)

[Hình 2.2. Ma trận số biểu thị mức xám của các điểm ảnh (Nguồn: Internet) 21](#_Toc201126686)

[Hình 2.3. Lược đồ ảnh đa mức xám kích thước 64x64 pixels (Nguồn: Internet) 22](#_Toc201126687)

[Hình 2.4. Lược đồ cường độ sáng theo chiều ngang ảnh (Nguồn: Internet) 22](#_Toc201126688)

[Hình 2.5. Minh họa phép co ảnh trên ảnh nhị phân (Nguồn: Internet) 24](#_Toc201126689)

[Hình 2.6. Ảnh nhị phân ký tự “A” trước và sau khi co ảnh (Nguồn: Internet) 24](#_Toc201126690)

[Hình 2.7. Các phương pháp phân đoạn khác nhau trong xử lý ảnh y tế (Nguồn: Internet) 28](#_Toc201126691)

[Hình 2.8. Cấu trúc cơ bản của mạng nơ-ron truyền thẳng (Nguồn: Internet) 29](#_Toc201126692)

[Hình 2.9. Sơ đồ của một mạng nơ-ron hồi quy (Nguồn: Internet) 33](#_Toc201126693)

[Hình 2.10. Mô hình mạng nơ-ron tích chập (Nguồn: Internet) 35](#_Toc201126694)

[Hình 2.11. Mô tả cấu trúc mô hình CNN AlexNet (Nguồn: Internet) 37](#_Toc201126695)

[Hình 2.12. Hình ảnh mô tả lớp tích chập (Nguồn: Internet) 38](#_Toc201126696)

[Hình 2.13. Hình ảnh mô phỏng lớp kết nối đầy đủ (Nguồn: Internet) 39](#_Toc201126697)

[Hình 2.14. Các chiều của bộ lọc (Nguồn: Internet) 40](#_Toc201126698)

[Hình 2.15. Minh hoạ độ trượt (Nguồn: Internet) 40](#_Toc201126699)

[Hình 2.16. Tính tương tích của tham số trong tầng tích chập (Nguồn: Internet) 42](#_Toc201126700)

[Hình 2.17. Ví dụ minh hoạ trường thụ cảm (Nguồn: Internet) 43](#_Toc201126701)

[Hình 2.18. Minh hoạ kiến trúc GAN (Nguồn: Internet) 47](#_Toc201126702)

[Hình 2.19. Cấu trúc cơ bản của Vision Transformer (Nguồn: Internet) 48](#_Toc201126703)

[Hình 3.1. Hình ảnh một số tổn thương da (Nguồn: Internet). 55](#_Toc201126704)

[Hình 3.2. Đoạn mã đồng nhất kích thước hình ảnh bằng phương pháp nội suy bilinear 56](#_Toc201126705)

[Hình 3.3. Đoạn mã tăng cường dữ liệu ảnh 58](#_Toc201126706)

[Hình 3.4. Kiến trúc mạng Resnet 50 59](#_Toc201126707)

[Hình 3.5. Đoạn xây dựng kiến trúc Resnet50 60](#_Toc201126708)

[Hình 3.6. Đoạn mã thể hiện các tham huấn luyện của mô hình 60](#_Toc201126709)

[Hình 3.7. Đoạn mã huấn luyện và đánh giá mô hình 61](#_Toc201126710)

[Hình 3.8. Đoạn mã theo dõi quá trình huấn luyện mô hình 61](#_Toc201126711)

[Hình 3.9. Đoạn mã biểu diễn ra mô hình độ chính xác và tổn thất trong quá trình huấn luyện 62](#_Toc201126712)

[Hình 3.10. Biểu đồ thể hiện độ chính xác và tổn thất trong quá trình huấn luyện qua các epoch 62](#_Toc201126713)

[Hình 3.11. Đoạn mã biểu diễn mô hình độ chính xác và tổn thất trên tập kiểm tra 63](#_Toc201126714)

[Hình 3.12. Biểu đồ thể hiện độ chính xác và tổn thất trên tập kiểm tra qua các epoch 63](#_Toc201126715)

[Hình 3.13. Đoạn mã lưu trữ mô hình CNN 64](#_Toc201126716)

[Hình 3.14. Đoạn mã tải và xử lý ảnh mới từ một hình ảnh tổn thương da 64](#_Toc201126717)

[Hình 3.15. Đoạn mã dự đoán và in kết quả 64](#_Toc201126718)

[Hình 3.16. Kết quả chương trình dự đoán kết quả cho ảnh mới 65](#_Toc201126719)

[Hình 4.1. Giao diện người dùng hệ thống web nhận diện 67](#_Toc201126720)

[Hình 4.2. Giao diện người dùng sau khi nhận diện 67](#_Toc201126721)

[Hình 4.3. Giao diện thông tin chi tiết của bệnh 68](#_Toc201126722)

[Hình 4.4. Đoạn mã lớp xử lý ứng dụng 68](#_Toc201126723)

[Hình 4.5. Giao diện Report 70](#_Toc201126724)

[Hình 4.6. Giao diện khi hệ thống khi không thể nhận dạng 70](#_Toc201126725)

DANH MỤC BẢNG BIỂU

[Bảng 2.1. Bảng minh họa phương thức Max Pooling và Average Pooling 38](#_Toc191619450)

[Bảng 2.2. Bảng so sánh 3 phương pháp Zero-padding. 40](#_Toc191619451)

[Bảng 2.3. Bảng tính toán độ phức tạp của mô hình 42](#_Toc191619452)

[Bảng 2.4. Bảng tổng hợp biến thể của ReLU 44](#_Toc191619453)

LỜI CẢM ƠN

Đầu tiên cho phép chúng em gửi lời cảm ơn sâu sắc tới các thầy cô Trường Công Nghệ Thông Tin và Truyền Thông – Đại học Công nghiệp Hà Nội, những người đã hết mình truyền đạt và chỉ dẫn cho chúng em những kiến thức, những bài học quý báu và bổ ích. Đặc biệt, chúng em xin được bày tỏ sự tri ân và xin chân thành cảm ơn giảng viên Lê Thị Thuỷ, người đã trực tiếp hướng dẫn, chỉ bảo chúng em trong suốt quá trình học tập, nghiên cứu và hoàn thành được đồ án.

Trải qua thời gian nghiên cứu và làm đề tài, nhóm chúng em đã khám phá được rất nhiều công nghệ mới, tuy nhiên, do năng lực, kiến thức và trình độ chuyên môn còn hạn hẹp nên không thể tránh khỏi một số thiết sót. Vì vậy, chúng em mong nhận được sự đóng góp ý kiến từ các thầy cô để giúp nhóm chúng em hoàn thiện kiến thức và bổ sung thêm thông tin cần thiết cho báo cáo của mình.

Một lần nữa, chúng em xin chân thành cảm ơn thầy và tất cả những người đã giúp đỡ và đồng hành cùng nhóm chúng em trong chặng đường này.

Chúng em xin trân trọng cảm ơn!

|  |
| --- |
| ***Sinh viên thực hiện***  Nguyễn Tuấn Minh  Trần Văn Nhã  Nguyễn Hoàng Tùng |

LỜI NÓI ĐẦU

Trong bối cảnh công nghệ ngày càng phát triển mạnh mẽ, đặc biệt là sự tiến bộ vượt bậc của trí tuệ nhân tạo và học máy, việc ứng dụng các kỹ thuật này vào lĩnh vực giao thông thông minh đang mở ra nhiều hướng đi mới, góp phần nâng cao an toàn và hiệu quả khi tham gia giao thông. Một trong những hướng nghiên cứu quan trọng hiện nay là nhận dạng biển báo giao thông tự động bằng mạng nơ-ron nhân tạo, đặc biệt là mạng nơ-ron tích chập (Convolutional Neural Networks – CNN).

Biển báo giao thông đóng vai trò quan trọng trong việc điều tiết, hướng dẫn và cảnh báo cho người tham gia giao thông. Tuy nhiên, trong thực tế, việc nhận biết biển báo bằng mắt thường có thể bị ảnh hưởng bởi nhiều yếu tố như tốc độ di chuyển cao, điều kiện thời tiết, ánh sáng kém hoặc các vật cản làm che khuất biển báo. Điều này làm tăng nguy cơ xảy ra tai nạn giao thông do không nhận diện hoặc nhận diện sai biển báo.

Để khắc phục những hạn chế đó, việc xây dựng một hệ thống tự động nhận dạng biển báo giao thông từ hình ảnh thông qua mạng nơ-ron là một giải pháp tiềm năng và cần thiết. Hệ thống này có thể được tích hợp vào các phương tiện giao thông thông minh, hệ thống xe tự hành hoặc các ứng dụng giám sát giao thông, góp phần nâng cao mức độ an toàn và giảm thiểu tai nạn.

Với tầm quan trọng và tính thực tiễn cao, đề tài ***"Xây dựng hệ thống nhận dạng biển báo giao thông bằng mạng nơ-ron"***được lựa chọn nhằm nghiên cứu và triển khai một mô hình nhận dạng biển báo chính xác, hiệu quả, có khả năng hoạt động tốt trong các điều kiện môi trường khác nhau, từ đó mở rộng khả năng ứng dụng của trí tuệ nhân tạo vào lĩnh vực giao thông hiện đại. Nội dung của báo cáo sẽ bao gồm các chương như sau:

***Chương 1. Tổng quan về bài toán nhận dạng biển báo giao thông:***Chương này sẽ trình bày tổng quan về các kiến thức, da tổn thương, nhu cầu nhận diện, cũng như các khó khăn hiện tại trong quá trình nhận diện, đồng thời cũng xác định mục tiêu và phạm vi của bài toán.

***Chương 2. Cơ sở lý thuyết:*** Trình bày các kỹ thuật học máy và mạng nơ-ron tích chập hiện có, phân tích ưu nhược điểm của từng kỹ thuật và lựa chọn phương pháp phù hợp nhất để giải quyết bài toán nhận diện bệnh ngoài da qua ảnh.

***Chương 3. Một số kết quả thực nghiệm:*** Chương này mô tả chi tiết quá trình thu thập dữ liệu, tiền xử lý dữ liệu và các bước thực hiện thực nghiệm. Kết quả đạt được từ các mô hình sẽ được trình bày và đánh giá.

***Chương 4. Xây dựng ứng dụng cho mô hình nhận dạng biển báo giao thông:*** Sử dụng các công cụ và thư viện lập trình để xây dựng hệ thống nhận diện tự động, bao gồm cả phần giao diện người dùng và phần xử lý ảnh. Các kỹ thuật triển khai và vận hành hệ thống cũng sẽ được mô tả chi tiết.

***Phần kết luận:*** Tổng hợp lại những kết quả đạt được, đưa ra những nhận xét và đánh giá về hiệu quả của hệ thống, đồng thời đề xuất những hướng phát triển tiếp theo cho nghiên cứu.

Thông qua việc thực hiện đề tài, người thực hiện có cơ hội tiếp thu thêm nhiều kiến thức mới về trí tuệ nhân tạo và ứng dụng của nó trong lĩnh vực y học, đồng thời mong muốn đóng góp vào việc nâng cao chất lượng chăm sóc sức khỏe cộng đồng. Đề tài hy vọng sẽ mang lại giá trị thực tiễn và mở ra nhiều cơ hội mới cho các ứng dụng công nghệ trong y tế.

# TỔNG QUAN VỀ KHO DỮ LIỆU VÀ PHƯƠNG PHÁP KHAI PHÁ

## Kiến thức tổng quan

### Trí tuệ nhân tạo

Trí tuệ nhân tạo (Artificial Intelligience) là một nhánh của khoa học máy tính nghiên cứu về các hệ thống có khả năng thực hiện những hành vi thông minh như con người, bao gồm học tập, suy luận, giải quyết vấn đề, hiểu ngôn ngữ tự nhiên và nhận thức môi trường.

#### Lịch sử phát triển

Vào năm 1943, Warren McCulloch và Walter Pitts đã đề xuất mô hình mạng nơ-ron nhân tạo đầu tiên, dựa trên việc mô phỏng hoạt động của nơ-ron sinh học. Họ xây dựng mô hình toán học mô tả một nơ-ron có hai trạng thái “bật” và “tắt”, cùng khả năng biểu diễn logic. Đây được xem là nền tảng cho nhiều nghiên cứu AI sau này.

Năm 1950, Alan Turing đề xuất “Turing Test” – một phép kiểm định đánh giá liệu một cỗ máy có thể biểu hiện hành vi thông minh như con người hay không.

Năm 1956, tại hội thảo Dartmouth Summer Research Project on Artificial Intelligence, John McCarthy đã chính thức đặt tên cho lĩnh vực này là Artificial Intelligence. Ông cũng phát triển ngôn ngữ lập trình Lisp, được sử dụng rộng rãi trong các hệ thống AI thời kỳ đầu.

Trong những năm 1960–1970, AI phát triển mạnh với sự ra đời của ngôn ngữ Prolog, các hệ chuyên gia đầu tiên như MYCIN, DENDRAL, có khả năng suy luận từ tập luật để đưa ra chẩn đoán.

Đến thập niên 1980, các hệ chuyên gia trở thành xu hướng chủ đạo, được ứng dụng trong y học, công nghiệp, tài chính... Tuy nhiên, AI cũng vấp phải những giới hạn do thiếu dữ liệu lớn và sức mạnh tính toán, dẫn đến thời kỳ gọi là "AI Winter".

Từ đầu những năm 2000, với sự phát triển của phần cứng, dữ liệu lớn (big data), và các phương pháp học máy hiện đại, đặc biệt là học sâu (deep learning), AI quay trở lại mạnh mẽ và trở thành một trong những công nghệ cốt lõi của cuộc cách mạng công nghiệp 4.0.

#### Các lĩnh vực của trí tuệ nhân tạo

**Lập luận và suy diễn tự động:** Cho phép hệ thống đưa ra kết luận từ cơ sở tri thức đã cho. Cần kết hợp giữa biểu diễn tri thức và các thuật toán suy luận logic.

**Biểu diễn tri thức:** Là cách thức lưu trữ và tổ chức tri thức cho máy tính. Các mô hình điển hình gồm mạng ngữ nghĩa (semantic networks), luật IF–THEN, khung (frames), và ontology.

**Lập kế hoạch (Planning):** Xây dựng chuỗi hành động để đạt mục tiêu nhất định trong điều kiện ràng buộc về môi trường.

**Xử lý ngôn ngữ tự nhiên (NLP):** Giúp máy hiểu và tạo ra ngôn ngữ con người, bao gồm các tác vụ như phân tích cú pháp, dịch tự động, nhận dạng tiếng nói, chatbot,...

**Hệ chuyên gia (Expert Systems):** Mô phỏng tri thức và tư duy của chuyên gia con người để hỗ trợ ra quyết định. Ví dụ: MYCIN (y học), DENDRAL (hóa học).

**Thị giác máy tính (Computer Vision):** Nhận dạng và phân tích hình ảnh từ thế giới thực (ví dụ: phát hiện đối tượng, nhận dạng khuôn mặt, biển báo giao thông…).

**Robot học (Robotics):** Phát triển các hệ thống máy móc thông minh có khả năng tương tác vật lý với môi trường.

#### Một số ứng dụng của trí tuệ nhân tạo

**Lĩnh vực giáo dục**: Mô hình trường học thông minh hay học trực tuyến ngày một phát triển nhờ ứng dụng AI vào quá trình tương tác học tập giữa nhà trường và học sinh, sinh viên tạo ra một hệ sinh thái về giáo dục mà ở đó nhà trường và học sinh tương tác với nhau một cách thuận lợi và nhanh chóng thông qua hệ sinh thái này.

**Lĩnh vực y tế**: Chẩn đoán, điều trị và theo dõi bệnh giúp nâng cao chăm sóc sức khỏe con người, giảm chi phí chữa bệnh cho người dân, trong đó kể cả các dự án như điều trị ung thư bằng AI, robot chăm sóc y tế…

**Công nghiệp**: Nhận dạng hình ảnh, giọng nói, robot thông minh, xe tự hành, hệ thống tương tác thực ảo… đã phát triển mạnh mẽ nhờ sự phát triển của mạng Neural học sâu, hệ thống xử lý phân tán song song (Parallel Distributed processing).

**Giao thông:** Nhận dạng biển báo, điều khiển phương tiện tự hành, dự báo luồng giao thông.

**Tài chính – ngân hàng:** Phát hiện gian lận, tư vấn đầu tư tự động, đánh giá rủi ro tín dụng.

Trong số các lĩnh vực của trí tuệ nhân tạo, **học máy (machine learning)** – đặc biệt là học sâu với các mạng nơ-ron – đóng vai trò cốt lõi trong việc giúp hệ thống có khả năng **tự học** từ dữ liệu. Trong các bài toán thị giác máy tính như nhận diện biển báo giao thông, các mô hình học sâu như **CNN** đã cho thấy hiệu quả vượt trội so với các phương pháp truyền thống. Phần tiếp theo sẽ trình bày chi tiết hơn về học máy và các kỹ thuật liên quan.

### Học máy

Học máy (Machine Learning – ML) là một nhánh của trí tuệ nhân tạo chuyên nghiên cứu và phát triển các thuật toán cho phép máy tính **tự động học** từ dữ liệu, nhận diện mẫu, và cải thiện hiệu suất theo thời gian mà không cần lập trình tường minh cho từng tác vụ cụ thể.

#### Các phương pháp học máy

Bao gồm 2 hướng tiếp cận chính

**Học quy nạp (inductive learning):** Máy tính học cách phân biệt hoặc khái quát hóa các khái niệm dựa trên **các ví dụ cụ thể (dữ liệu đã thu thập)**. Đây là hướng tiếp cận phổ biến trong học máy hiện đại nhờ vào sự phát triển của dữ liệu lớn (Big Data).

**Học suy diễn (deductive learning):** Máy tính học thông qua việc **suy diễn logic từ các luật và tri thức chuyên ngành** đã được xác định. Phương pháp này ít được sử dụng hơn trong học máy hiện đại, nhưng có vai trò quan trọng trong các hệ chuyên gia.

#### Các loại học máy chính

Machine Learning được phân chia thành ba nhóm chính:

**Học có giám sát (Supervised Learning):**

Dữ liệu đầu vào đã được gán nhãn rõ ràng. Mục tiêu là huấn luyện mô hình học được hàm ánh xạ từ đầu vào đến đầu ra (nhãn).  
Ví dụ: phân loại ảnh biển báo, nhận diện chữ viết, dự đoán giá nhà.  
Các thuật toán phổ biến: Mạng nơ-ron nhân tạo (ANN), cây quyết định (Decision Trees), máy vector hỗ trợ (SVM), rừng ngẫu nhiên (Random Forest), Boosting (AdaBoost, Gradient Boosting)

**Học không giám sát (Unsupervised Learning):**

Dữ liệu không có nhãn. Mục tiêu là tìm ra cấu trúc tiềm ẩn, phân cụm hoặc giảm chiều dữ liệu.  
Ví dụ: phân nhóm khách hàng, phát hiện bất thường.  
Các thuật toán thường gặp: K-means, DBSCAN (gom cụm), phân tích thành phần chính (PCA), phân tích thành phần độc lập (ICA)

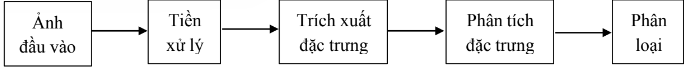
**Học tăng cường (Reinforcement Learning):**

Máy học thông qua tương tác với môi trường. Tại mỗi bước, hệ thống chọn hành động và nhận được phần thưởng hoặc hình phạt. Mục tiêu là tối đa hóa phần thưởng theo thời gian.  
Ứng dụng: chơi game, điều khiển robot, xe tự hành...

#### Ứng dụng trong bài toán nhận dạng

Trong lĩnh vực xử lý ảnh, bài toán phân loại mẫu và nhận dạng đối tượng được hiểu là việc phân loại, xác định các mẫu thuộc lớp nào đó một cách chính xác dựa vào việc đo lường về đối tượng dựa trên các đặc trưng của đối tượng. Nói một cách khác, nhận dạng đối tượng nhằm mục đích phân loại các mẫu dựa trên kiến thức có trước về đối tượng hoặc dựa vào thông tin thống kê, đo lường được trích rút ra từ các mẫu trong tập dữ liệu huấn luyện. Ngày nay, nhận dạng được ứng dụng rộng rãi trong nhiều hệ thống như nhận dạng người trong các hệ thống tương tác người máy; nhận dạng khuôn mặt và định danh cá thể, nhận dạng phương tiện giao thông trong các hệ thống giao thông thông minh, hỗ trợ lái xe; nhận dạng hành động trong các hệ thống giám sát thông minh; nhận dạng chữ viết... Lĩnh vực nhận dạng đối tượng liên quan đến các phương pháp, kỹ thuật xử lý ảnh và học máy. Để huấn luyện mô hình và nhận dạng đối tượng, các mẫu dữ liệu thường được số hóa và biểu diễn thành các vector đặc trưng trong một không gian đa chiều tương ứng, được gọi là không gian đặc trưng. Nhờ vào các vector đặc trưng này để có thể phân tích, đánh giá nhằm xây dựng được mô hình đối tượng phục vụ phân loại, nhận dạng chúng. Trong nhận dạng, các phương pháp học máy có giám sát và không giám sát đều được nghiên cứu và ứng dụng. Các kỹ thuật học máy có giám sát thường được sử dụng như cây quyết định, mạng neural, SVM, Boosting, rừng ngẫu nhiên (random forest). Các kỹ thuật học máy không giám sát như gom cụm (Clustering Algorithms), phân tích thành phần chính (Principal Component Analysis - PCA), phân tích thành phần độc lập (Independent Component Analysis).

Quá trình nhận dạng đối tượng được mô tả bằng sơ đồ tổng quát như sau:



Hình 1.7. Quá trình nhận dạng đối tượng (Nguồn: Internet)

Sự phát triển mạnh mẽ của học máy, đặc biệt là học sâu (deep learning), đã mở ra khả năng vượt trội trong các bài toán thị giác máy tính, trong đó có nhận dạng biển báo giao thông. Phần tiếp theo sẽ trình bày về xử lý ảnh số và thị giác máy tính – nền tảng kỹ thuật quan trọng cho các hệ thống nhận diện đối tượng từ ảnh chụp.

### Xử lý ảnh số và thị giác máy tính

Xử lý ảnh số (Digital Image Processing) và Thị giác máy tính (Computer Vision) là hai lĩnh vực quan trọng trong khoa học máy tính, có vai trò nền tảng trong nhiều công nghệ hiện đại. Trong khi xử lý ảnh số tập trung vào việc phân tích, cải biến và trích xuất thông tin từ hình ảnh dưới dạng số, thì thị giác máy tính hướng tới việc giúp máy “hiểu” nội dung thị giác một cách tương tự như con người, từ đó đưa ra quyết định hoặc hành động phù hợp.

Sự kết hợp giữa xử lý ảnh số và thị giác máy tính đã tạo nên những đột phá trong nhiều lĩnh vực như y tế, giao thông thông minh, công nghiệp tự động hóa, an ninh giám sát, thực tế tăng cường và trí tuệ nhân tạo tổng quát.

#### Xử lý ảnh số

Xử lý ảnh số là quá trình áp dụng các phương pháp tính toán để biến đổi hoặc trích xuất thông tin từ dữ liệu ảnh. Một hình ảnh số được biểu diễn dưới dạng ma trận các pixel, trong đó mỗi pixel chứa thông tin về màu sắc hoặc cường độ sáng. Các kỹ thuật xử lý ảnh số cơ bản bao gồm:

* **Lọc ảnh:** Loại bỏ nhiễu hoặc cải thiện chất lượng ảnh bằng các bộ lọc như Gaussian, Median hoặc Sobel.
* **Tăng cường ảnh:** Điều chỉnh độ tương phản, độ sáng hoặc màu sắc để làm nổi bật các chi tiết quan trọng.
* **Phân đoạn ảnh:** Chia ảnh thành các vùng có ý nghĩa, ví dụ như tách nền và vật thể.
* **Nén ảnh:** Giảm kích thước dữ liệu ảnh mà vẫn giữ được chất lượng, sử dụng các phương pháp như JPEG hoặc PNG.
* **Nhận dạng mẫu:** Xác định các đặc trưng hình học hoặc cấu trúc như cạnh, đường thẳng, góc hoặc hình dạng cụ thể trong ảnh.

Các kỹ thuật trên được hiện thực qua các thư viện và công cụ mạnh mẽ như OpenCV, MATLAB, PIL hoặc scikit-image, giúp xử lý ảnh hiệu quả trong thời gian thực.

#### Thị giác máy tính

Thị giác máy tính là lĩnh vực nghiên cứu và phát triển các phương pháp cho phép máy tính diễn giải và hiểu nội dung hình ảnh hoặc video. Mục tiêu không chỉ là nhận biết mà còn phân tích ngữ cảnh và tương tác với môi trường thị giác.

Một số nhiệm vụ chính trong thị giác máy tính bao gồm:

* **Nhận diện đối tượng (Object Detection/Recognition):**  
   Xác định vị trí và phân loại các đối tượng trong ảnh, ví dụ như xe cộ, người đi bộ, biển báo...
* **Phân tích chuyển động (Motion Analysis):**  
   Theo dõi chuyển động của đối tượng trong chuỗi khung hình, ứng dụng trong giám sát, giao thông và robot.
* **Nhận dạng khuôn mặt (Face Recognition):**  
   So sánh và xác định danh tính cá nhân dựa trên đặc điểm khuôn mặt.
* **Xây dựng mô hình 3D (3D Reconstruction):**  
   Tái tạo mô hình không gian ba chiều từ ảnh 2D, ứng dụng trong thực tế ảo (VR) hoặc thị giác robot.
* **Phân tích ngữ cảnh (Scene Understanding):**  
   Xác định môi trường và tình huống xảy ra trong ảnh, như cảnh đông người, giao thông, trong nhà hay ngoài trời.

Thị giác máy tính **thường tích hợp các phương pháp học máy**, đặc biệt là học sâu (deep learning), để nâng cao hiệu quả nhận dạng và phân tích.

#### Học sâu trong xử lý ảnh và thị giác máy tính

Sự phát triển của **học sâu (Deep Learning)**, đặc biệt là **mạng nơ-ron tích chập (Convolutional Neural Networks - CNN)**, đã mang lại bước tiến vượt bậc cho thị giác máy tính. Các kiến trúc nổi bật bao gồm:

* **CNN truyền thống:** VGG, ResNet, Inception – chuyên về nhận diện đối tượng và phân đoạn ảnh.
* **Phát hiện đối tượng thời gian thực:** YOLO, SSD – đạt tốc độ và độ chính xác cao trong phát hiện nhanh nhiều đối tượng.
* **Phân đoạn ngữ nghĩa (Semantic Segmentation):** U-Net, DeepLab – phân vùng ảnh theo từng loại đối tượng.
* **Mô hình mới dựa trên Transformer:** Vision Transformer (ViT), Swin Transformer – đạt hiệu quả cao trong các bài toán phân loại ảnh và nhận dạng nâng cao.

Việc kết hợp học sâu với **GPU/TPU hiệu năng cao** và **dữ liệu lớn** (ImageNet, COCO, Pascal VOC...) đã giúp thị giác máy tính đạt hiệu suất gần với con người trong nhiều tác vụ.

#### Ứng dụng thực tiễn

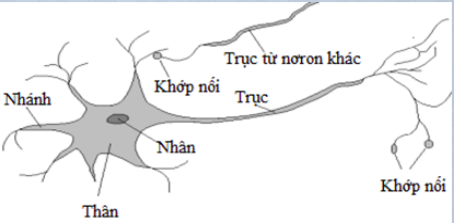
Xử lý ảnh và thị giác máy tính hiện diện trong hầu hết các lĩnh vực của đời sống hiện đại:

* **Y tế:** Phân tích ảnh chẩn đoán (X-quang, CT, MRI), phát hiện khối u, theo dõi tiến trình điều trị, hỗ trợ phẫu thuật bằng robot.
* **Giao thông:** Nhận diện biển báo giao thông, phát hiện phương tiện, hỗ trợ lái xe, phát triển xe tự hành.
* **Công nghiệp:** Giám sát dây chuyền sản xuất, phát hiện lỗi sản phẩm bằng camera, phân tích thị giác cho robot công nghiệp.
* **An ninh và giám sát:** Nhận diện khuôn mặt, phân tích hành vi, theo dõi đối tượng nghi vấn qua camera an ninh.
* **Giải trí và truyền thông:** Thực tế tăng cường (AR), chỉnh sửa ảnh/video thông minh, tạo hiệu ứng thị giác trong game và phim.

Trong tương lai, các nghiên cứu có xu hướng tập trung vào: Mô hình nhẹ và tiết kiệm năng lượng (EfficientNet, MobileNet) để triển khai trên thiết bị di động, IoT hoặc hệ thống nhúng. Học không giám sát và bán giám sát (Unsupervised/Semi-supervised Learning): Giảm phụ thuộc vào dữ liệu gán nhãn. Học chuyển giao (Transfer Learning): Tái sử dụng mô hình đã huấn luyện trên tập dữ liệu lớn cho bài toán mới có ít dữ liệu. Tích hợp thị giác máy tính với các lĩnh vực khác: Xử lý ngôn ngữ tự nhiên (NLP) mô tả ảnh tự động, hỏi đáp từ video, giúp robot tương tác và điều hướng trong môi trường thực tế, hệ thống AI đa phương thức (multimodal) xử lý đồng thời ảnh, âm thanh, văn bản để hiểu sâu hơn về ngữ cảnh.

### Mạng nơ-ron nhân tạo

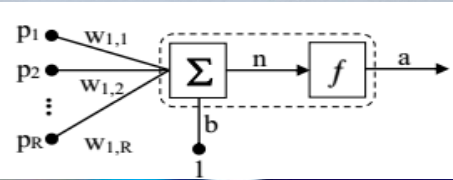
Mạng nơ-ron nhân tạo (ANN) là một mô hình tính toán lấy cảm hứng từ cấu trúc và hoạt động của não bộ con người. ANN bao gồm nhiều "nơ-ron" nhân tạo (artificial neurons), được tổ chức thành các lớp (layers), kết nối với nhau thông qua các trọng số (weights). Các mô hình ANN là nền tảng quan trọng trong học sâu (deep learning) và được ứng dụng rộng rãi trong các bài toán phức tạp như phân loại ảnh, nhận diện giọng nói, dịch máy, và đặc biệt là nhận dạng biển báo giao thông từ ảnh chụp.



Hình 1.2. Hình ảnh minh ohạ nơ ron thần kinh

#### Cấu trúc cơ bản của một nơ ron thông thường bao gồm 3 lớp:

* **Lớp đầu vào (Input layer):** Nhận dữ liệu đầu vào từ thế giới thực (ví dụ: hình ảnh các biển báo giao thông).
* **Lớp ẩn (Hidden layers):** Thực hiện các phép biến đổi tuyến tính và phi tuyến trên dữ liệu.
* **Lớp đầu ra (Output layer):** Trả về kết quả đầu ra, ví dụ như nhãn biển báo.

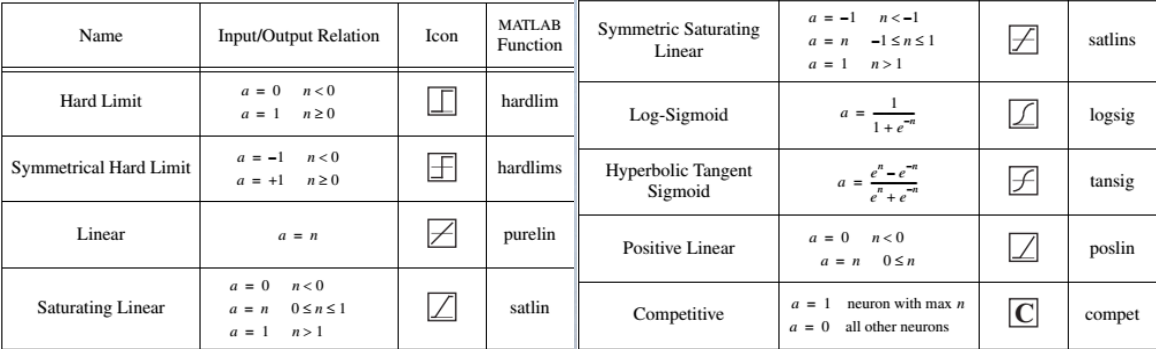


Hình 1.3: Mô hình cơ bản của mạng nơ ron

#### Nơ-ron và hàm kích hoạt

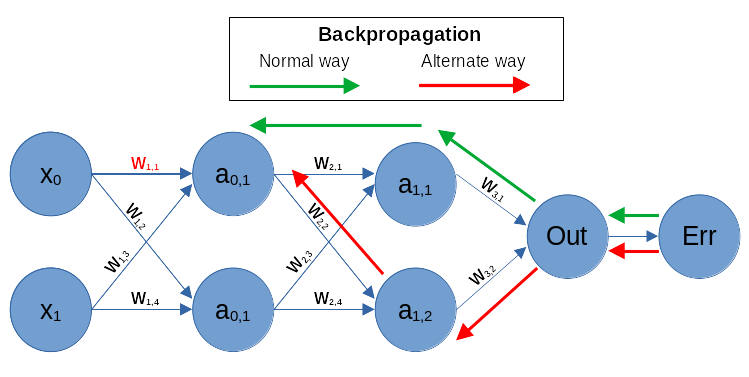
Một nơ-ron tổng hợp tín hiệu thông qua bộ công có trọng số và đưa kết quả qua một hàm kích hoạt phi tuyến như:

* Sigmoid: Thường dùng trong bài toán phân loại nhị phân.
* Tanh: Mạnh hơn sigmoid, giúp mô hình học tốt hơn.
* ReLU (Rectified Linear Unit): Rất phổ biến trong các mạng sâu vì đơn giản và hiệu quả.
* Ngoài ra còn có các hàm kích hoạt khác như: hardlim, purelin, satlin, poslin, compet, ....

Hình 1.4.: Một số hàm kích hoạt phổ biến

#### Quá trình huấn luyện mạng

Mạng nơ-ron được huấn luyện thông qua quá trình lan truyền ngược (Backpropagation) và sử dụng thuật toán tối ưu hóa như Gradient Descent để điều chỉnh trọng số, giảm sai số giữa đầu ra dự đoán và đầu ra thực tế.



Hình 1.5. Quá trình học lan truyền ngược (Backpropagation)

#### Các kỹ thuật hỗ trợ và ứng dụng

Regularization: Giảm overfitting, điển hình như L1, L2 regularization.

Dropout: Ngẫu nhiên loại bỏ một số nơ-ron trong quá trình huấn luyện để tránh mạng học thuộc dữ liệu.

Batch Normalization: Giúp tăng tốc độ huấn luyện và cải thiện hiệu suất mạng.

Trong các bài toán thị giác máy tính, ANN là nền tảng để xây dựng các mô hình phức tạp hơn như mạng tích chập (CNN). Với các bài toán phân loại hình ảnh như nhận diện biển báo, mạng nơ-ron giúp mô hình học được các đặc trưng phân biệt giữa các loại biển hiệu khác nhau một cách tự động và chính xác.

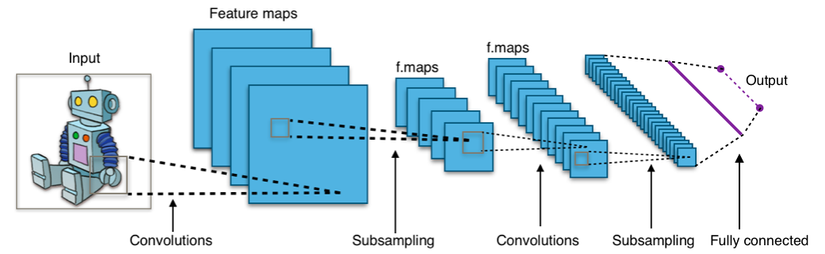
### Mạng tích chập

Mạng nơ-ron tích chập (CNN) là một kiến trúc mạng nơ-ron đặc biệt, thiết kế dành riêng cho các bài toán xử lý ảnh và thị giác máy tính. CNN có khả năng tự động học các đặc trưng từ ảnh mà không cần trích xuất thủ công như các phương pháp truyền thống.

#### Cấu trúc tổng quan

Một mạng CNN (Convolutional Neural Networks) cơ bản gồm ba lớp chính:

* Lớp tích chập (Convolutional Layer): Dùng các bộ lọc (kernel) để quét qua ảnh và trích xuất đặc trưng cục bộ như đường viền, góc cạnh, hình dạng.
* Lớp gộp (Pooling Layer): Thường là MaxPooling, giúp giảm kích thước dữ liệu và tăng tính kháng nhiễu.
* Lớp fully connected: Kết nối đầy đủ như mạng MLP để đưa ra dự đoán cuối cùng.

Hình 1.6: Kiến trúc mạng CNN

Ngoài ra, các lớp như Dropout, BatchNorm, và Flatten cũng được sử dụng để cải thiện hiệu quả và độ ổn định của mạng.

Trong mô hình CNN có 2 khía cạnh cần quan tâm là tính bất biến (Location Invariance) và tính kết hợp (Compositionality). Với cùng một đối tượng, nếu đối tượng này được chiếu theo các gốc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể. Pooling layer sẽ cho bạn tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling). Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các filter. Chính vì vậy, CNN cho ra mô hình huấn luyện với độ chính xác khá cao.

Một số kiến trúc CNN tiêu biểu:

* LeNet-5 (1998): Kiến trúc đầu tiên được ứng dụng trong nhận dạng chữ viết tay.
* AlexNet (2012): Giúp CNN bùng nổ nhờ chiến thắng cuộc thi ImageNet.
* VGGNet: Mạng sâu hơn, với các lớp tích chập 3×3 đơn giản nhưng hiệu quả.
* ResNet: Sử dụng kết nối tắt (skip connection) để tránh hiện tượng mất mát gradient khi huấn luyện mạng sâu.

#### Ưu điểm, hạn chế

Bảng 1.1: So sánh ưu điểm, hạn chế của CNN

|  |  |
| --- | --- |
| **Ưu điểm** | **Hạn chế** |
| Hiệu quả cao trong xử lý ảnh và nhận dạng đối tượng.  Khả năng tự động trích xuất đặc trưng (features).  Giảm số lượng tham số nhờ dùng tích chập (convolution). | Cần nhiều dữ liệu huấn luyện.  Tốn tài nguyên tính toán.  Dễ bị overfitting nếu mô hình lớn và dữ liệu ít. |

#### Ứng dụng trong bài toán nhận dạng biển giao thông

CNN được xem là giải pháp hiệu quả nhất hiện nay trong bài toán nhận dạng biển báo từ ảnh. Nhờ khả năng học đặc trưng tự động và khả năng tổng quát hóa cao, CNN có thể phân biệt được các loại biển báo có hình dạng, màu sắc, và biểu tượng khác nhau, ngay cả khi ảnh chụp bị nhiễu, mờ hoặc có góc chụp không thuận lợi.

## Bài toán nhận diện

Nhận diện là quá trình phân tích có hệ thống nhằm xác định bản chất, đặc điểm hoặc mức độ của một đối tượng, hiện tượng hay tình huống cụ thể. Đây là một bước nền tảng trong nhiều lĩnh vực như y học, kỹ thuật, công nghiệp, khoa học dữ liệu và trí tuệ nhân tạo, với mục tiêu cung cấp thông tin đáng tin cậy để hỗ trợ ra quyết định chính xác.

Quá trình nhận diện đòi hỏi khả năng phân biệt giữa các đặc trưng tương đồng, xử lý dữ liệu phức tạp, đa dạng và đưa ra kết luận logic từ các tín hiệu đầu vào. Các phương pháp hiện nay bao gồm cả cách tiếp cận truyền thống dựa trên kinh nghiệm con người và các giải pháp hiện đại ứng dụng công nghệ tiên tiến, với đầu vào đầu ra linh hoạt, phục vụ nhiều mục tiêu và bối cảnh khác nhau.

### Phương pháp nhận diện truyền thống

Các phương pháp truyền thống chủ yếu dựa vào sự quan sát, phân tích và kinh nghiệm chuyên môn của con người để nhận diện đối tượng hoặc hiện tượng. Thông tin đầu vào thường bao gồm:

* Dữ liệu trực quan: hình dạng, màu sắc, kích thước, kết cấu;
* Thông tin bổ trợ: dữ liệu lịch sử, bối cảnh môi trường, mô tả từ người dùng, hồ sơ hoặc kết quả xét nghiệm.

Các công cụ hỗ trợ thường là thiết bị đo lường đơn giản, như kính lúp, thước đo, hoặc biểu mẫu đánh giá thủ công.

Đầu ra thường là nhận định sơ bộ hoặc kết luận mang tính chủ quan, chẳng hạn như chẩn đoán ban đầu trong y tế, phân loại mức độ hư hỏng của thiết bị, hoặc đánh giá nguy cơ trong môi trường.

Tuy nhiên, phương pháp này phụ thuộc mạnh vào yếu tố con người, dễ bị ảnh hưởng bởi thiếu sót trong tri thức, kinh nghiệm không đồng đều, hoặc thiên kiến nhận thức, đặc biệt khi đối tượng có đặc điểm gần giống nhau hoặc dữ liệu bị nhiễu, thiếu hụt. Điều này làm nổi bật nhu cầu ứng dụng công nghệ để tăng cường hiệu quả và độ tin cậy trong nhận diện.

### Phương pháp nhận diện ứng dụng công nghệ hiện đại

Với sự phát triển của trí tuệ nhân tạo (AI), học máy (machine learning) và xử lý dữ liệu lớn (big data), các hệ thống nhận diện hiện đại có khả năng:

* Tiếp nhận đầu vào đa dạng: hình ảnh, video, dữ liệu vector, tín hiệu từ cảm biến, hồ sơ điện tử…
* Phân tích sâu bằng thuật toán: như mạng nơ-ron tích chập (CNN), mạng Transformer, hoặc mô hình học sâu đa phương thức.

Ví dụ:

* Hình ảnh y khoa được phân tích để xác định dấu hiệu bất thường.
* Dữ liệu cảm biến trong công nghiệp phát hiện dao động bất thường ở máy móc.
* Video giám sát nhận diện hành vi khả nghi trong thời gian thực.

Các hệ thống hiện đại có khả năng:

* Tự động trích xuất đặc trưng từ dữ liệu mà không cần lập trình thủ công;
* Đưa ra đầu ra định lượng hoặc định tính, như xác suất thuộc một lớp cụ thể, cảnh báo tự động, hoặc đề xuất hành động;
* Học hỏi và cải thiện theo thời gian nhờ cơ chế huấn luyện lặp lại với dữ liệu mới.

Đặc biệt, học sâu cho phép xử lý các mẫu phức tạp mà con người khó nhận biết, vượt qua giới hạn của kinh nghiệm chủ quan và tăng độ chính xác ngay cả trong điều kiện dữ liệu nhiễu hoặc chưa rõ ràng.

### Lợi ích và tiềm năng

So với phương pháp truyền thống, các giải pháp nhận diện hiện đại mang lại nhiều lợi ích vượt trội, bao gồm:

* Tăng độ chính xác và độ tin cậy trong nhiều bối cảnh;
* Giảm thiểu sai sót và thiên kiến con người;
* Tăng tốc độ xử lý, đặc biệt trong các hệ thống thời gian thực;
* Phát hiện sớm và ngăn chặn rủi ro trong các lĩnh vực nhạy cảm như y tế, an ninh, hoặc công nghiệp nặng;
* Khả năng mở rộng linh hoạt, phù hợp với dữ liệu lớn hoặc môi trường thay đổi liên tục.

Bảng 1.2: Ứng dụng vào bài toán nhận diện

|  |  |
| --- | --- |
| Y tế | Nhận diện khối u từ ảnh CT/MRI, hỗ trợ bác sĩ chẩn đoán bệnh tự động, tầm soát ung thư ở giai đoạn sớm. |
| Công nghiệp | Phát hiện lỗi sản phẩm, kiểm tra độ chính xác lắp ráp, giám sát dây chuyền sản xuất bằng thị giác máy tính. |
| Giao thông | Nhận diện biển báo, người đi bộ, phương tiện trong hệ thống xe tự hành hoặc quản lý giao thông đô thị. |
| An ninh | Nhận diện khuôn mặt, phát hiện hành vi bất thường, giám sát an toàn tại sân bay, khu vực công cộng. |
| Tiện ích người dùng | Ứng dụng di động dùng camera để nhận diện da, cây trồng, thực phẩm, mã QR, gợi ý hành động tự động. |

Đặc biệt quan trọng, các hệ thống nhận diện dựa trên AI có thể triển khai tại những khu vực thiếu chuyên gia, thiếu hạ tầng, giúp tăng cường khả năng tự đánh giá và hỗ trợ người dùng phổ thông một cách hiệu quả, nhanh chóng và tiết kiệm chi phí.

Tuy nhiên, vẫn còn tồn tại một số thách thức sau:

Bảng 1.2: Thách thức trong việc ứng dụng vào bài toán nhận dạng

|  |  |
| --- | --- |
| Chất lượng và số lượng dữ liệu | Dữ liệu không đầy đủ hoặc thiếu,  Dữ liệu nhiễu, không đồng nhất |
| Độ phức tạp của đặc trưng đối tượng | Nhiều đối tượng có hình dạng, màu sắc hoặc kết cấu tương tự, gây khó phân biệt.  Đặc trưng của đối tượng biến đổi theo điều kiện ánh sáng, góc nhìn, hoặc môi trường |
| Chi phí và hạ tầng tính toán | Các mô hình học sâu như CNN, Transformer đòi hỏi tài nguyên tính toán lớn  Chi phí huấn luyện, triển khai và bảo trì hệ thống có thể cao, đặc biệt với doanh nghiệp nhỏ. |
| Overfitting | Mô hình có thể học quá kỹ trên dữ liệu huấn luyện nhưng kém hiệu quả trên dữ liệu mới, do không tổng quát hóa tốt |
| Tính minh bạch và khả năng giải thích | Các mô hình học sâu thường được xem là “hộp đen”, khó lý giải tại sao đưa ra quyết định cụ thể nào đó.  Khó khăn khi cần kiểm tra, chứng minh hoặc giải trình trong các lĩnh vực yêu cầu sự minh bạch như y tế, pháp lý. |
| Thay đổi môi trường hoặc điều kiện đầu vào | Hệ thống nhận diện có thể mất hiệu quả nếu môi trường thay đổi |
| An toàn và bảo mật | Mô hình có thể bị tấn công, nguy cơ rò rỉ dữ liệu cá nhân nếu không xử lý đúng cách, đặc biệt trong các ứng dụng như nhận diện khuôn mặt. |
| Rào cản triển khai thực tế | Khó tích hợp với hệ thống hiện tại (legacy systems), thiếu nhân lực chuyên môn |

## Bài toán nhận dạng biển báo giao thông

### Giới thiệu chung

Bài toán nhận dạng biển báo giao thông là một dạng đặc thù của bài toán thị giác máy tính, trong đó hệ thống có nhiệm vụ phát hiện và phân loại các biển báo xuất hiện trong một bức ảnh đầu vào. Mỗi biển báo cần được xác định về vị trí (tọa độ bounding box), nhãn phân loại (label) tương ứng với loại biển báo, cùng với độ tin cậy (confidence score) thể hiện xác suất mà mô hình tin vào dự đoán của nó.

Cụ thể, khi một người dùng tải ảnh lên hệ thống – có thể là ảnh chụp từ camera hành trình, điện thoại di động hoặc hình ảnh giao thông khác – mô hình sẽ:

* Tìm và khoanh vùng các biển báo trong ảnh (object detection).
* Phân loại mỗi biển báo thuộc loại nào (classification).
* Trả về kết quả gồm: nhãn, bounding box và độ chính xác.

Bài toán được triển khai dưới dạng hệ thống web đơn giản, cho phép kiểm thử, kiểm chứng hiệu suất của mô hình học sâu trên các tình huống thực tế.

### Mục đích và ý nghĩa của đề tài

#### Mục đích

Phát triển một mô hình học sâu có khả năng nhận dạng chính xác các loại biển báo giao thông đường bộ Việt Nam trong ảnh tĩnh.

Xây dựng một hệ thống đơn giản, dễ sử dụng, cho phép người dùng tải ảnh lên và nhận kết quả nhận dạng trực quan.

Thử nghiệm ứng dụng công nghệ thị giác máy tính vào một bài toán gần gũi với đời sống thực tiễn tại Việt Nam.

#### Ý nghĩa khoa học

Cung cấp một bài toán nghiên cứu ứng dụng học sâu và thị giác máy tính vào lĩnh vực giao thông.

Là cơ hội thử nghiệm và đánh giá hiệu quả của các mô hình hiện đại như YOLOv8, EfficientDet, hoặc Vision Transformers trong điều kiện thực tế tại Việt Nam.

Góp phần mở rộng nghiên cứu về phát hiện đối tượng, xử lý ảnh có điều kiện ánh sáng, vật thể che khuất, nhiễu ảnh thực tế.

#### Ý nghĩa thực tiễn

Với người dân: Giúp tra cứu ý nghĩa biển báo, nhất là các biển ít phổ biến hoặc trong các khu vực mới chưa quen thuộc.

Với cơ quan chức năng: Hỗ trợ kiểm tra, đánh giá tình trạng biển báo trên hạ tầng giao thông, tiết kiệm thời gian so với kiểm tra thủ công.

Trong giáo dục: Là công cụ hỗ trợ học viên học lái xe làm quen với biển báo nhanh hơn, thông qua hình ảnh thực tế.

Trong giao thông thông minh: Là bước đầu tiên trong chuỗi xử lý để xây dựng hệ thống hỗ trợ lái xe hoặc xe tự hành.

### Đầu vào và đầu ra của bài toán

#### Đầu vào

Ảnh hoặc video thu được từ camera gắn trên xe hoặc thiết bị di động.

Dữ liệu ảnh được chuẩn hóa và gán nhãn từ các bộ dữ liệu được gán nhãn theo chuẩn Việt Nam.

#### Đầu ra

**Label (Nhãn biển báo)**: Loại biển được mô hình nhận dạng, ví dụ “Cấm đi ngược chiều”, “Giới hạn tốc độ 50 km/h”. Có tổng cộng 55 classes

**Bounding box**: Tọa độ vị trí của biển báo trong ảnh dưới dạng (x\_min, y\_min, x\_max, y\_max).

**Confidence score**: Độ tin cậy của mô hình cho nhãn dự đoán, biểu thị bằng số thực từ 0 đến 1 (hoặc %).

**Biểu đồ trực quan về độ chính xác**: Biểu đồ thể hiện mức confidence hoặc tỉ lệ dự đoán đúng/sai theo từng loại biển báo, giúp đánh giá tổng quan hiệu suất mô hình.

**Thông tin chi tiết của biển báo**: Mô tả ngắn gọn về ý nghĩa, cách xử lý khi gặp loại biển đó trên đường (dựa trên dữ liệu bổ sung), hỗ trợ người dùng không chỉ nhận diện mà còn hiểu rõ nội dung.

### Phạm vi của bài toán

Bài toán nhận dạng biển báo giao thông trong đề tài này được giới hạn trong phạm vi như sau:

Loại biển báo: Tập trung vào các loại biển báo giao thông đường bộ chính thức tại Việt Nam, bao gồm:

* + Biển báo cấm
  + Biển báo nguy hiểm
  + Biển báo hiệu lệnh
  + Biển báo chỉ dẫn

Dữ liệu đầu vào: Chủ yếu là ảnh tĩnh được chụp từ camera hành trình hoặc thiết bị di động, ở nhiều điều kiện môi trường khác nhau. Hoặc video được quay lại.

Xử lý và đầu ra: Hệ thống chỉ thực hiện nhận dạng biển báo hiện diện trong ảnh, đầu ra bao gồm: loại biển (label), vị trí (bounding box), và độ tin cậy (confidence score).

Không xử lý các trường hợp:

* + Biển báo bị hư hỏng nặng, che khuất hoàn toàn
  + Ảnh mờ nhoè đến mức không thể phân biệt bằng mắt thường
  + Nhận dạng biển trong video thời gian thực (có thể là mở rộng trong tương lai)

### Những thuận lợi, khó khăn

Bảng 1.4. Thuận lợi, khó khăn khi giải quyết bài toán

|  |  |
| --- | --- |
| **Thuận lợi** | **Khó khăn** |
| Có tiêu chuẩn rõ ràng về hình dạng, màu sắc, nội dung biển báo giao thông. | Biển báo có thể bị mờ, nghiêng, che khuất hoặc hư hỏng trong thực tế. |
| Các mô hình học sâu hiện đại như CNN, YOLO, Transformer hoạt động hiệu quả. | Góc chụp không cố định, biển bị biến dạng hình học gây khó nhận diện. |
| Dữ liệu ảnh có thể thu thập từ nhiều nguồn như camera hành trình, di động. | Thiếu bộ dữ liệu gán nhãn đầy đủ, chuẩn hóa theo quy định tại Việt Nam. |
| Khả năng triển khai hệ thống trên nhiều nền tảng (web, mobile, xe hơi,...). | Yêu cầu mô hình nhẹ, nhanh để chạy trên thiết bị tài nguyên hạn chế. |
| Hỗ trợ nâng cao an toàn giao thông và xây dựng hệ thống giao thông thông minh. | Biển báo có nhiều loại tương tự nhau dễ gây nhầm lẫn (ví dụ: các biển giới hạn tốc độ khác nhau). |

### Miền ứng dụng

Bài toán nhận dạng biển báo giao thông có thể được ứng dụng trong nhiều lĩnh vực thực tiễn:

Giao thông thông minh (ITS): Hệ thống cảnh báo biển báo cho người lái xe trong thời gian thực; phát hiện và xử lý vi phạm giao thông dựa trên camera giám sát

Xe tự hành / hỗ trợ lái nâng cao (ADAS): Nhận dạng biển để điều chỉnh tốc độ, hành vi điều khiển tự động

Ứng dụng di động / học tập: Hỗ trợ học viên lái xe luyện tập nhận diện biển báo thông qua ảnh. Website tra cứu biển báo tức thời từ ảnh chụp

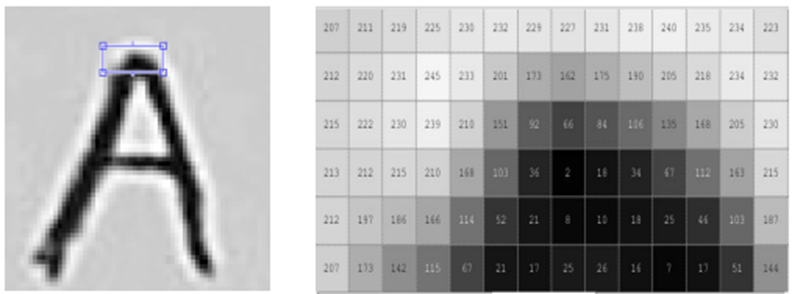
Phân tích dữ liệu giao thông: Thống kê phân bố biển báo theo vùng miền, phát hiện các khu vực thiếu hoặc sai biển

# CÁC PHƯƠNG PHÁP KHAI PHÁ DỮ LIỆU

## Một số kỹ thuật xử lý ảnh

### Khái niệm cơ bản

* **Ảnh số**: Ảnh số là tập hợp hữu hạn các điểm ảnh với mức xám phù hợp dùng để mô tả ảnh gần với ảnh thật. Ảnh số bao gồm một tập hữu hạn các phần tử được biểu diễn bởi giá trị số. Ảnh số có thể được biểu diễn dưới dạng ma trận hai chiều, mỗi phần tử của ảnh số gọi là điểm ảnh (pixel). Số điểm ảnh xác định độ phân giải của ảnh. Ảnh có độ phân giải càng cao thì càng thể hiện rõ nét các đặt điểm của tấm ảnh và càng làm cho tấm ảnh trở nên thực và sắc nét hơn.



Hình 2.1. Ảnh ký tự chữ A và ma trận số của vùng chọn (Nguồn: Internet)

* **Điểm ảnh**: Điểm ảnh (Pixel) là một phần tử của ảnh số tại tọa độ (x, y) với độ xám hoặc màu nhất định. Kích thước và khoảng cách giữa các điểm ảnh đó được chọn thích hợp sao cho mắt người cảm nhận sự liên tục về không gian và mức xám (hoặc màu) của ảnh số gần như ảnh thật. Mỗi phần tử trong ma trận được gọi là một phần tử ảnh. Trong hình 2.1 mỗi điểm ảnh là một ô mang một giá trị số biểu thị mức xám từ 0 đến 255.
* **Mức xám của ảnh**: Là kết quả của sự biến đổi tương ứng 1 giá trị độ sáng của 1 điểm ảnh với một giá trị nguyên dương. Thông thường nó xác định trong tuỳ thuộc vào giá trị mà mỗi điểm ảnh được biểu diễn. Các thang giá trị mức xám thông thường: 2, 16, 32, 64, 128. Ảnh đa mức xám thường dùng là 256, như vậy mức xám thường xác định trong khoảng tùy thuộc vào giá trị mà mỗi điểm ảnh được biểu diễn.

A table with numbers and a few squares

Description automatically generated with medium confidence

Hình 2.2. Ma trận số biểu thị mức xám của các điểm ảnh (Nguồn: Internet)

* **Độ phân giải ảnh***:* Độ phân giải (Resolution) của ảnh là mật độ điểm ảnh được ấn định trên một ảnh số được hiển thị. Theo định nghĩa, khoảng cách giữa các điểm ảnh phải được chọn sao cho mắt người vẫn thấy được sự liên tục của ảnh. Việc lựa chọn khoảng cách thích hợp tạo nên một mật độ phân bổ, đó chính là độ phân giải và được phân bố theo trục x và y trong không gian hai chiều.

Một số dạng ảnh:

* **Ảnh màu***:* thường là các ảnh chứa thông tin về đối tượng được biểu diễn dưới dạng màu sắc mà mắt thường có thể quan sát được. Mỗi điểm ảnh có cấu trúc gồm nhiều kênh khác nhau, thông thường trong máy tính, nó biểu diễn 3 lớp màu cơ bản RGB (Red, Green, Blue).
* **Ảnh đa mức xám***:* thường biểu diễn thông tin liên quan đến cường độ đa xám của đối tượng trong không gian mà không được thể hiện bởi màu sắc thực của nó.
* **Ảnh nhị phân***:* Giá trị xám của tất cả các điểm ảnh chỉ nhận giá trị 1 hoặc 0 như vậy mỗi điểm ảnh trong ảnh nhị phân được biểu diễn bởi 1 bit. Ảnh nhị phân thường được dùng để biểu diễn, phân biệt sự xuất hiện đối tượng và nền trong mỗi bức ảnh.

### Các kỹ thuật xử lý ảnh

#### Lược đồ ảnh (Histogram)

Lược đồ ảnh là một đồ thị biểu diễn tần số xuất hiện của cường độ sáng điểm ảnh theo các mức hay nói cách khác lược đồ Histogram của một hình ảnh biểu diễn lược đồ về sự phân bố các mức cường độ xám của một bức ảnh.

A comparison of a person's face

Description automatically generated

Hình 2.3. Lược đồ ảnh đa mức xám kích thước 64x64 pixels (Nguồn: Internet)

Lược đồ ảnh thể hiện mức cường độ sáng theo chiều ngang ảnh là kết quả của việc tính tổng giá trị của các điểm ảnh theo từng cột của ảnh. Ví dụ: cho ảnh xám đầu vào có kích thước 600x200 pixels, khi đó lược đồ sẽ có 600 cột và độ cao mỗi cột là tổng giá trị (từ 0 đến 255) của 200 điểm ảnh tương ứng với 200 dòng.

A black and white image of a line

Description automatically generated

Hình 2.4. Lược đồ cường độ sáng theo chiều ngang ảnh (Nguồn: Internet)

#### Nhị phân hoá ảnh

Để chuyển sang ảnh nhị phân, trước tiên cần chuyển ảnh về dạng ảnh xám với các mức xám có giá trị từ 0 đến 255 dựa trên ba giá trị red, green, blue theo công thức:

Sử dụng ngưỡng thích hợp để chuyển các điểm ảnh về dạng 0 (màu đen) hoặc 255 (màu trắng) theo nguyên tắc:

nếu và nếu

**Trong đó:** là giá trị kết quả tại điểm ảnh có tọa độ .

là giá trị điểm ảnh của ảnh cần xử lý.

là ngưỡng.

#### Làm mịn ảnh

Mịn ảnh được thực hiện dựa trên bộ lọc trơn (Smoothing filter) nhằm loại nhiễu, bước này dùng trong quá trình tiền xử lý (Preprocessing) khi phải giảm bớt một số chi tiết không cần thiết của một đối tượng nào đó trong ảnh. Một hướng áp dụng phổ biến để giảm nhiễu là lọc tuyến tính, những bộ lọc tuyến tính theo hướng này được biết đến như là lọc thông thấp.

Ý tưởng cho những bộ lọc thông thấp là thay thế giá trị mức sáng của mọi điểm ảnh bằng giá trị mức sáng trung bình của các hàng xóm, định nghĩa theo mặt nạ lọc. Kết quả trên dẫn tới ảnh số văn bản mất đi những chi tiết nhiễu, ma trận của một bộ lọc làm mịn ảnh thường sử dụng có các hệ số như sau:

hoặc

#### Phép co ảnh – Erosion

Xét tập hợp A và tập hợp B (phần tử cấu trúc), phép co ảnh nhị phân của tập hợp A bởi phần tử cấu trúc B được ký hiệu là A ⊝ B và viết dưới dạng công thức như sau:

**Trong đó:** : Ma trận điểm ảnh của ảnh nhị phân

: Phần tử cấu trúc

A number and number equation

Description automatically generated with low confidence

Hình 2.5. Minh họa phép co ảnh trên ảnh nhị phân (Nguồn: Internet)

A black and white image of a person

Description automatically generated

Hình 2.6. Ảnh nhị phân ký tự “A” trước và sau khi co ảnh (Nguồn: Internet)

## Trích chọn và biểu diễn đặc trưng hình ảnh

### Đặc trưng ảnh

#### Đặc trưng màu sắc

Màu sắc là một đặc trưng nổi bật và được sử dụng phổ biến nhất trong tìm kiếm ảnh theo nội dung. Mỗi một điểm ảnh (thông tin màu sắc) có thể được biểu diễn như một điểm trong không gian màu sắc ba chiều. Các không gian màu sắc thường dùng là: RGB, Munsell, CIE, HSV.

#### Đặc trưng kết cấu

Kết cấu hay còn gọi là vân (texture) là một đối tượng dùng để phân hoạch ảnh ra thành những vùng được quan tâm và để phân lớp những vùng đó. Vân cung cấp thông tin sự sắp xếp về mặt không gian của màu sắc và cường độ của một ảnh.

#### Đặc trưng hình dạng

Hình dạng của một ảnh hay một vùng là một đặc trưng quan trọng trong việc xác định và phân biệt ảnh trong nhận dạng mẫu. Mục tiêu chính của biểu diễn hình dạng trong nhận dạng mẫu là đo thuộc tính hình học của một đối tượng được dùng trong phân lớp, so sánh và nhận dạng đối tượng.

### Một số phương pháp trích chọn đơn giản

#### Trích chọn đặc trưng ảnh dựa trên màu sắc

* **Lược đồ màu (Histogram)***:* Là đại lượng đặc trưng cho phân bố màu cục bộ của ảnh. Độ đo tính tương tự về màu sắc được tính bằng phần giao của 2 lược đồ màu ảnh truy vấn H(IQ) và ảnh trong cơ sở dữ liệu ảnh H(ID). Kết quả sẽ là một lược đồ màu thể hiện độ giống nhau giữa 2 ảnh trên. Tuy nhiên vì lược đồ màu chỉ thể hiện tính phân bố màu toàn cục của ảnh mà không xét đến tính phân bố cục bộ của điểm ảnh nên có thể có 2 ảnh xem rất khác nhau nhưng lại có cùng lược đồ màu.
* **Vector liên kết màu (Color Coherence Vector)**: Là lược đồ tinh chế lược đồ màu, chia mỗi ô màu thành 2 nhóm điểm ảnh: nhóm liên kết màu (coherence pixels) và nhóm không liên kết màu (noncoherent pixels). Vector liên kết màu còn giúp giải quyết khuyết điểm về tính không duy nhất của lược đồ màu đối với ảnh. Hai ảnh có thể có chung lược đồ màu nhưng khác nhau hoàn toàn, đây là khuyết điểm của lược đồ màu. Nhưng với tìm kiếm theo đặc trưng vector liên kết màu thì nó sẽ giải quyết được khuyết điểm không duy nhất này.

#### Trích chọn đặc trưng ảnh dựa trên kết cấu

Kết cấu hay còn gọi là vân (texture) là một đối tượng dùng để phân hoạch ảnh ra thành những vùng được quan tâm và để phân lớp những vùng đó. Vân cung cấp những thông tin sự sắp xếp về mặt không gian của màu sắc và cường độ của một ảnh.

## Phương pháp phân đoạn ảnh

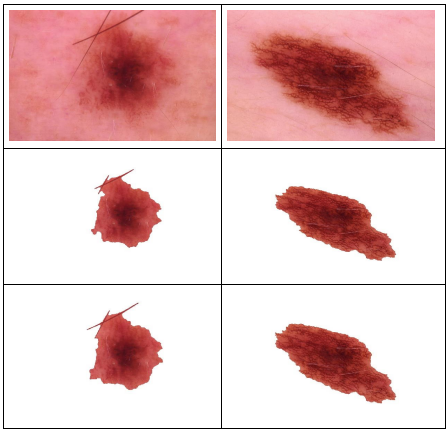
Trong lĩnh vực xử lý ảnh, phân đoạn ảnh (segmentation) được hiểu là quá trình nhóm điểm ảnh thành các vùng sao cho các điểm ảnh trong cùng một vùng có chung đặc điểm, tính chất nào đó, ví dụ như sự tương tự về màu sắc, cường độ hoặc kết cấu. Nói cách khác, phân đoạn ảnh là một quá trình chia một ảnh số thành nhiều vùng khác nhau, mỗi vùng gồm tập hợp các điểm ảnh (có thể gọi là superpixels hoặc region) có cùng tính chất. Trong một số tài liệu, phân đoạn hình ảnh còn được gọi là phân vùng ảnh. Mục tiêu của việc phân đoạn là để đơn giản hóa hoặc thay đổi cách biểu diễn một ảnh theo tiêu chuẩn nào đó nhằm làm cho nó ý nghĩa hơn và dễ dàng trong phân tích, trích xuất đặc trưng. Kết quả của quá trình phân đoạn ảnh là tập hợp các vùng (region) hay còn gọi là các phân đoạn (segments) có thể bao gồm toàn bộ ảnh hoặc tập hợp các vùng đối tượng được trích xuất ra từ ảnh.

Phân đoạn ảnh có ý nghĩa quan trọng trong việc phát hiện các đặc tính và cấu trúc đối tượng trong ảnh vì mỗi vùng thường miêu tả cấu trúc tương ứng với đối tượng trong vùng đó. Ví dụ số lượng các vùng trong ảnh thể hiện rằng trong ảnh đó có chứa nhiều đối tượng khác nhau và ngược lại mỗi đối tượng chứa các vùng trong nó tương ứng với các phần khác nhau của cùng đối tượng. Phân đoạn ảnh có nhiều ứng dụng quan trọng trong nhiều lĩnh vực khác nhau như phân tích hình ảnh y học (medical imaging), phát hiện và nhận dạng đối tượng, phát hiện hành động trong các hệ thống camera giám sát thông minh, hệ thống điều khiển và giao thông thông minh, xe không người lái, xử lý hình ảnh vệ tinh…

Phân đoạn ảnh được xem là quá trình tiền xử lý, có ý nghĩa quan trọng trong hệ thống xử lý ảnh. Phân đoạn tốt sẽ giúp cho quá trình xử lý ở các bước tiếp theo đạt hiệu quả cao hơn, nâng cao độ chính xác và cũng như tiết kiệm chi phí tính toán.

Các vùng phân tách được dùng trong nhiều ngữ cảnh khác nhau và đặc biệt có thể được sử dụng làm đại diện thay thế cho đối tượng và biểu diễn dưới dạng nhiều hình thức khác nhau. Việc lựa chọn mô tả vùng dữ liệu có nhiều cách thức khác nhau tùy thuộc vào tính chất, mục tiêu và đảm bảo phù hợp cho từng ứng dụng cụ thể. Nhiều ứng dụng không chỉ đòi hỏi mô tả các vùng riêng lẻ mà còn cần thể hiện được mối quan hệ với nhau giữa các vùng trong ảnh.

Ngưỡng (threshold) là một khái niệm phổ biến trong lĩnh vực phân đoạn đối tượng của kỹ thuật xử lý ảnh. Ngưỡng được dùng để chỉ một giá trị nào đó mà ta dùng để phân tách một tập hợp các điểm ảnh thành các vùng phân biệt. Ví dụ sử dụng một ngưỡng để phân đoạn ảnh thành các vùng có cường độ sáng lớn hơn và các vùng còn lại có giá trị cường độ sáng nhỏ hơn hoặc bằng . Giá trị ngưỡng thường được xác định dựa vào giá trị đặc biệt như giá trị trung bình, giá trị trung vị hoặc dựa vào khảo sát thống kê theo kinh nghiệm. Có nhiều cách phân loại khác nhau, tuy nhiên nếu căn cứ vào số lượng ngưỡng dùng cho phân đoạn thì có phương pháp phân đoạn sử dụng ngưỡng đơn, phân đoạn ngưỡng kép hoặc phân đoạn đa ngưỡng. Nếu căn cứ vào tính chất biến thiên của giá trị ngưỡng thì có thể phân thành phương pháp dùng ngưỡng cố định (constant threshold) và ngưỡng động hay ngưỡng thích ứng (adaptive threshold). Trong đó, ngưỡng động được hiểu theo nghĩa giá trị của ngưỡng sẽ thay đổi tùy thuộc vào sự biến thiên của tập dữ liệu ảnh theo không gian và thời gian. Thông thường giá trị ngưỡng được xác định bằng cách áp dụng phương pháp phân tích thống kê.



Hình 2.7. Các phương pháp phân đoạn khác nhau trong xử lý ảnh y tế (Nguồn: Internet)

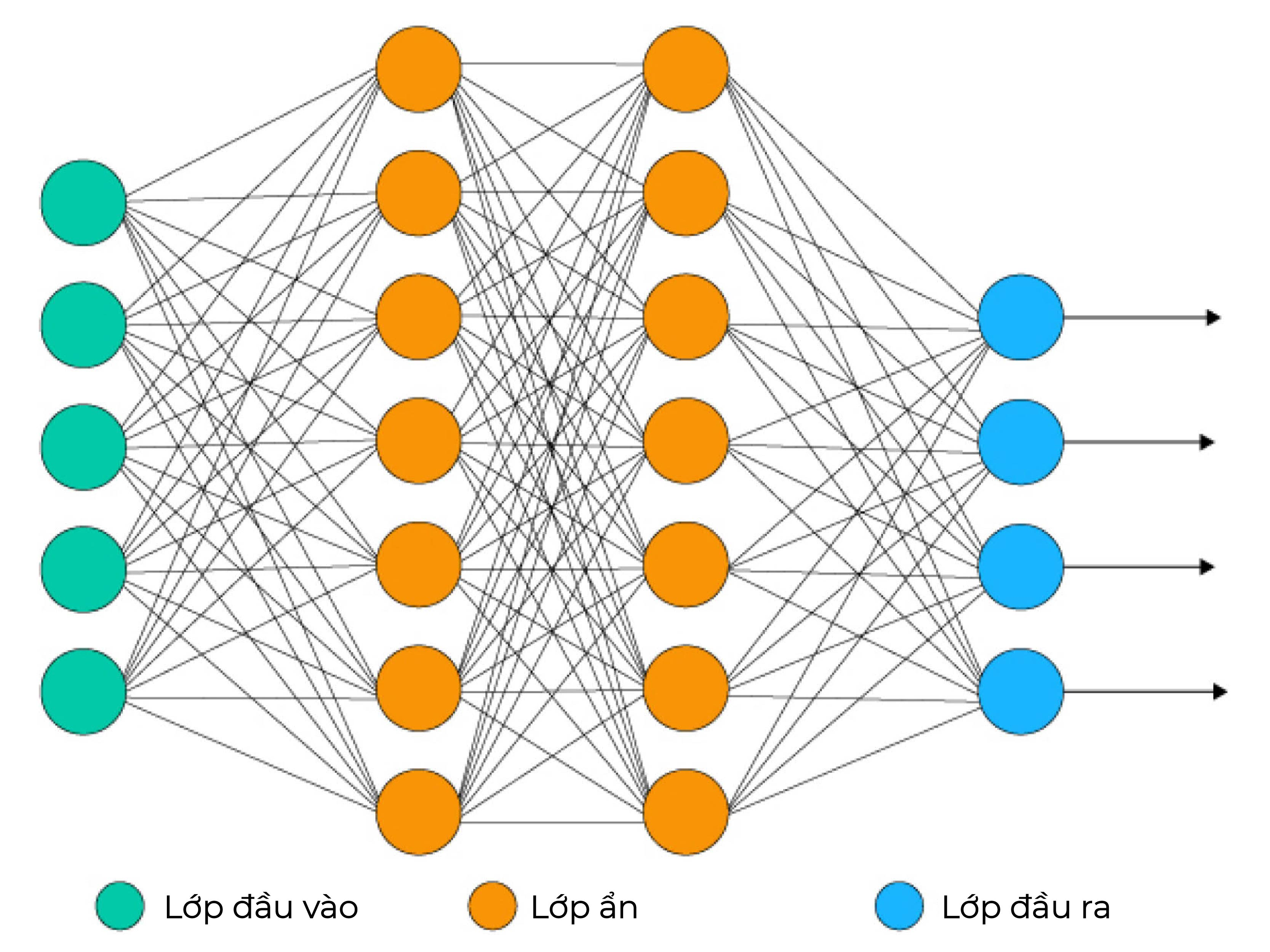
## Mạng nơ-ron truyền thẳng

### Giới thiệu

Mạng nơ-ron truyền thẳng, hay Multi-Layer Perceptron (MLP), là một kiến trúc cơ bản của mạng nơ-ron nhân tạo thuộc loại feedforward, trong đó dữ liệu được xử lý theo một hướng từ lớp đầu vào qua các lớp ẩn đến lớp đầu ra, không có kết nối ngược. MLP đóng vai trò nền tảng trong học máy, đặc biệt trong các bài toán phân loại, hồi quy, và nhận dạng mẫu, bao gồm ứng dụng nhận diện da tổn thương từ ảnh chụp. Dù không tối ưu như mạng nơ-ron tích chập (CNN) trong xử lý hình ảnh, MLP vẫn được sử dụng nhờ tính đơn giản và khả năng học các đặc trưng phi tuyến tính từ dữ liệu. MLP được sử dụng rộng rãi trong các bài toán phân loại, hồi quy, và nhận dạng mẫu, là nền tảng cho nhiều mô hình học sâu phức tạp hơn như CNN và RNN.

### Cấu trúc

Mạng MLP được thiết kế để xử lý dữ liệu theo một hướng duy nhất – từ đầu vào qua các tầng ẩn (nếu có) đến đầu ra – mà không có vòng lặp hay kết nối ngược.



Hình 2.8. Cấu trúc cơ bản của mạng nơ-ron truyền thẳng (Nguồn: Internet)

#### Thành phần chính:

Mạng MLP bao gồm ba thành phần chính

* Lớp đầu vào (Input Layer):
* Nhận dữ liệu ảnh dưới dạng vector đặc trưng (ví dụ, giá trị pixel hoặc đặc trưng trích xuất như màu sắc, kết cấu da).
* Số nơ-ron bằng số đặc trưng đầu vào (ví dụ, ảnh 28x28 pixel grayscale sẽ có 784 nơ-ron).
* Lớp ẩn (Hidden Layers):
* Gồm một hoặc nhiều lớp, thực hiện tính toán phi tuyến để học các mẫu phức tạp (như tổn thương da, vết loét).
* Mỗi nơ-ron trong lớp ẩn nhận đầu vào từ tất cả các nơ-ron của lớp trước đó, thực hiện phép nhân trọng số, cộng bias và áp dụng hàm kích hoạt.
* Các hàm kích hoạt phổ biến gồm:
  + - **ReLU (Rectified Linear Unit):** Giảm gradient vanishing, phù hợp cho dữ liệu ảnh lớn.
    - **Sigmoid:** Dùng trong phân loại nhị phân (bệnh/không bệnh).
    - **Tanh:** Đầu ra chuẩn hóa trong , ít dùng hơn trong hình ảnh.
* Trọng số và bias của từng nơ-ron được cập nhật trong quá trình huấn luyện để tối ưu hóa đầu ra.
* Lớp đầu ra (Output Layer):
* Chuyển đổi đầu ra từ lớp ẩn thành kết quả mong muốn.
* Số lượng nơ-ron trong lớp đầu ra phụ thuộc vào loại bài toán:
  + - **Phân loại nhị phân:** Một nơ-ron với hàm kích hoạt sigmoid.
    - **Phân loại nhiều lớp:** Số lượng nơ-ron bằng số lớp, sử dụng softmax để tính xác suất.
    - **Hồi quy:** Một nơ-ron với hàm kích hoạt tuyến tính để dự đoán giá trị thực.

#### Quá trình tính toán trong MLP

* Truyền tín hiệu tiến (Forward Propagation):
* Dữ liệu đầu vào được nhân với trọng số, cộng bias và áp dụng hàm kích hoạt để truyền qua từng lớp.
* Công thức tổng quát cho từng nơ-ron:

Trong đó: là đầu ra của lớp.

là ma trận trọng số.

là vector bias.

là hàm kích hoạt.

* Tính toán lỗi (Loss Calculation):
* Sai số giữa đầu ra dự đoán và giá trị thực tế được tính toán bằng các hàm mất mát:
  + - **Cross-Entropy Loss:** Dùng cho bài toán phân loại.
    - **Mean Squared Error (MSE):** Dùng cho bài toán hồi quy.
* Lan truyền ngược (Backpropagation):
* Sử dụng thuật toán lan truyền ngược để tính gradient của hàm mất mát theo trọng số.
* Cập nhật trọng số bằng thuật toán tối ưu như Stochastic Gradient Descent (SGD) hoặc Adam.

### Ưu điểm

* **Dễ triển khai**: MLP là mô hình cơ bản và dễ hiểu trong học sâu.
* **Tổng quát hóa tốt**: Có thể mô hình hóa các hàm phi tuyến tính nhờ vào các hàm kích hoạt như ReLU, Sigmoid hoặc Tanh.
* **Khả năng mở rộng**: Dễ dàng mở rộng với nhiều lớp ẩn để học được các đặc trưng phức tạp hơn.

### Nhược điểm

* **Không hiệu quả với dữ liệu lớn**: Khi số lượng đặc trưng tăng, MLP gặp khó khăn trong việc học và dễ bị overfitting.
* **Không khai thác được đặc trưng không gian**: Không hiệu quả bằng CNN khi xử lý dữ liệu hình ảnh.
* **Cần nhiều dữ liệu**: MLP hoạt động tốt khi có đủ dữ liệu huấn luyện, nhưng dễ bị ảnh hưởng khi dữ liệu hạn chế.

### Các kiến trúc MLP phổ biến

* **MLP với một lớp ẩn**: Cấu trúc đơn giản nhất, phù hợp với bài toán hồi quy tuyến tính và phân loại đơn giản.
* **MLP nhiều lớp ẩn**: Được sử dụng trong các bài toán phức tạp hơn như nhận diện chữ viết tay (MNIST), xử lý ngôn ngữ tự nhiên (NLP).
* **MLP với dropout**: Kết hợp dropout để giảm overfitting, giúp mô hình tổng quát hóa tốt hơn.
* MLP là một phần quan trọng trong học sâu và được sử dụng kết hợp với các mô hình phức tạp khác để cải thiện hiệu suất.

## Mạng nơ-ron hồi quy

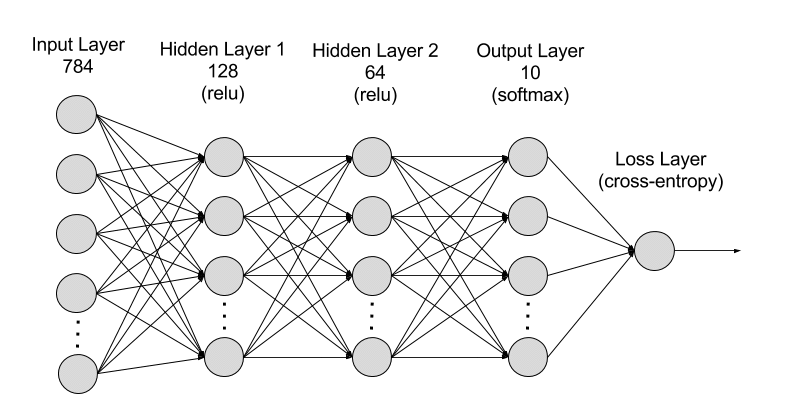
### Giới thiệu

Mạng nơ-ron hồi quy (RNN) là một mô hình học sâu được đào tạo để xử lý và chuyển đổi đầu vào dữ liệu tuần tự thành đầu ra dữ liệu tuần tự cụ thể. Dữ liệu tuần tự là dữ liệu, chẳng hạn như từ, câu hoặc dữ liệu chuỗi thời gian, trong đó các thành phần tuần tự tương quan với nhau dựa trên ngữ nghĩa phức tạp và quy tắc cú pháp. RNN là một hệ thống phần mềm gồm nhiều thành phần được kết nối với nhau theo cách con người thực hiện chuyển đổi dữ liệu tuần tự, chẳng hạn như dịch văn bản từ ngôn ngữ này sang ngôn ngữ khác. Phần lớn RNN đang được thay thế bằng trí tuệ nhân tạo (AI) dựa trên công cụ biến đổi và các mô hình ngôn ngữ lớn (LLM), hiệu quả hơn nhiều trong việc xử lý dữ liệu tuần tự.

### Cấu trúc

RNN khác với MLP ở chỗ nó có một trạng thái ẩn (**hidden state**) giúp lưu trữ thông tin từ các bước trước đó. Cấu trúc chính của RNN bao gồm:

* **Lớp đầu vào (Input Layer)**: Nhận một chuỗi dữ liệu đầu vào theo từng thời điểm.
* **Lớp ẩn (Hidden Layer)**: Có kết nối vòng lặp để lưu trữ trạng thái ẩn từ bước trước.
* **Lớp đầu ra (Output Layer)**: Dự đoán kết quả cho mỗi bước thời gian hoặc cho toàn bộ chuỗi.



Hình 2.9. Sơ đồ của một mạng nơ-ron hồi quy (Nguồn: Internet)

RNN hoạt động bằng cách lần lượt truyền dữ liệu tuần tự nhận được đến các lớp ẩn. Tuy nhiên, RNN cũng có quy trình làm việc tự lặp lại hay *hồi quy*: lớp ẩn có thể ghi nhớ và sử dụng các đầu vào trước đó cho các dự đoán trong tương lai trong một thành phần bộ nhớ ngắn hạn. Quy trình này sử dụng đầu vào hiện tại và bộ nhớ đã lưu trữ để dự đoán chuỗi tiếp theo.

### Phân loại

RNN thường có đặc trưng là kiến trúc một-một: một chuỗi đầu vào được liên kết với một đầu ra. Tuy nhiên, bạn có thể điều chỉnh linh hoạt thành các cấu hình khác nhau cho các mục đích cụ thể. Sau đây là một số loại RNN phổ biến.

#### ****Một-nhiều****

Loại RNN này dẫn một đầu vào đến một số đầu ra. Loại này tạo điều kiện cho các ứng dụng ngôn ngữ như chú thích hình ảnh bằng cách tạo một câu từ một từ khóa duy nhất.

#### ****Nhiều-nhiều****

Mô hình sử dụng nhiều đầu vào để dự đoán nhiều đầu ra. Ví dụ: bạn có thể tạo một công cụ dịch ngôn ngữ bằng RNN, với khả năng phân tích câu và cấu trúc chính xác các từ trong một ngôn ngữ khác.

#### ****Nhiều-một****

Một số đầu vào được ánh xạ đến một đầu ra. Loại này rất hữu ích trong các ứng dụng như phân tích cảm xúc, trong đó mô hình dự đoán cảm xúc của khách hàng như tích cực, tiêu cực và trung lập từ lời chứng thực đầu vào.

### Ưu điểm

* **Phù hợp với dữ liệu tuần tự**: RNN có thể ghi nhớ trạng thái trước đó, hữu ích cho NLP và chuỗi thời gian.
* **Có thể xử lý dữ liệu có độ dài biến đổi**: Không yêu cầu đầu vào có kích thước cố định.

### Nhược điểm

* **Vấn đề vanishing gradient**: Gradient có thể trở nên rất nhỏ trong quá trình lan truyền ngược.
* **Khó song song hóa**: Do tính chất phụ thuộc tuần tự, RNN khó được tăng tốc trên GPU như CNN hay Transformer.

### Các kiến trúc RNN phổ biến

* **RNN cơ bản**: Mô hình gốc, dễ triển khai nhưng gặp vấn đề vanishing gradient.
* **BRNN**: Mạng nơ-ron hồi quy hai chiều (BRNN) xử lý các chuỗi dữ liệu với các lớp tiến và lùi của các nút ẩn. Lớp tiến hoạt động tương tự như RNN, lưu trữ đầu vào trước đó ở trạng thái ẩn và sử dụng đầu vào đó để dự đoán đầu ra tiếp theo. Trong khi đó, lớp lùi hoạt động theo hướng ngược lại bằng cách lấy cả đầu vào hiện tại và trạng thái ẩn trong tương lai để cập nhật trạng thái ẩn hiện tại. Kết hợp cả hai lớp cho phép BRNN cải thiện độ chính xác của dự đoán bằng cách xem xét bối cảnh trong quá khứ và tương lai.
* **LSTM**: Bộ nhớ dài-ngắn hạn (LSTM) là một biến thể RNN cho phép mô hình mở rộng dung lượng bộ nhớ để phù hợp với dòng thời gian dài hơn. RNN chỉ có thể nhớ đầu vào vừa qua tức thì. Nhưng RNN không thể sử dụng đầu vào từ một số chuỗi trước đó để cải thiện dự đoán.
* **GRU**: Đơn vị hồi quy có cổng (GRU) là một RNN cho phép lưu giữ bộ nhớ có chọn lọc. Mô hình thêm một bản cập nhật và quên cổng vào lớp ẩn của mô hình, điều này có thể lưu trữ hoặc xóa thông tin trong bộ nhớ.

## Mạng nơ-ron tích chập

### Giới thiệu

Mạng nơ-ron tích chập (Convolutional Neural Network - CNN) là một loại mạng nơ-ron nhân tạo (Artificial Neural Network - ANN) được thiết kế đặc biệt để xử lý dữ liệu có cấu trúc dạng lưới, ví dụ như hình ảnh. CNN được lấy cảm hứng từ cấu trúc của vỏ não thị giác của động vật, nơi các tế bào thần kinh được sắp xếp theo một cấu trúc phân cấp để xử lý thông tin thị giác.

Diagram of a diagram of a network

Description automatically generated with medium confidence

Hình 2.10. Mô hình mạng nơ-ron tích chập (Nguồn: Internet)

### Cấu trúc

Mạng nơ-ron tích chập thường bao gồm 3 lớp chính:

* **Lớp tích chập (Convolutional layer)**:Thực hiện phép tích chập giữa bộ lọc (kernel) và dữ liệu đầu vào để trích xuất các đặc trưng cục bộ. Tầng tích chập (CONV) sử dụng các bộ lọc để thực hiện phép tích chập khi đưa chúng đi qua đầu vào II theo các chiều của nó. Các siêu tham số của các bộ lọc này bao gồm kích thước bộ lọc FF và độ trượt (stride) SS. Kết quả đầu ra OO được gọi là feature map hay activation map.
* **Lớp gộp (Pooling layer)**: Tầng pooling (POOL) là một phép downsampling, thường được sử dụng sau tầng tích chập, giúp tăng tính bất biến không gian. Cụ thể, max pooling và average pooling là những dạng pooling đặc biệt, mà tương ứng là trong đó giá trị lớn nhất và giá trị trung bình được lấy ra.
* **Lớp kết nối đầy đủ (Fully connected layer)**: Tầng kết nối đầy đủ (FC) nhận đầu vào là các dữ liệu đã được làm phẳng, mà mỗi đầu vào đó được kết nối đến tất cả nơ-ron. Trong mô hình mạng CNNs, các tầng kết nối đầy đủ thường được tìm thấy ở cuối mạng và được dùng để tối ưu hóa mục tiêu của mạng ví dụ như độ chính xác của lớp.

A screenshot of a computer

Description automatically generated

Hình 2.11. Mô tả cấu trúc mô hình CNN AlexNet (Nguồn: Internet)

#### Lớp tích chập (Convolutional Layer)

Lớp này là nơi thể hiện ý tưởng ban đầu của CNN. Thay vì kết nối toàn bộ điểm ảnh, layer này sẽ sử dụng một tập các bộ lọc (filters) có kích thước nhỏ so sánh với ảnh (thường là 5x5 hoặc 3x3) áp  vào một vùng trong ảnh và tiến hành tính tích chập giữa bộ lọc và giá trị điểm ảnh trong vùng cục bộ đó. Bộ lọc sẽ lần lượt được dịch chuyển theo một giá trị bước trượt (stride) chạy dọc theo ảnh và quét toàn bộ ảnh.

A close-up of a diagram

Description automatically generated

Hình 2.12. Hình ảnh mô tả lớp tích chập (Nguồn: Internet)

Như vậy, trong quá trình tích chập, filter sẽ quét qua toàn bộ ảnh, từng phần một và tạo ra một feature map mới, trích xuất các đặc trưng quan trọng từ ảnh gốc. Số lượng feature maps phụ thuộc vào số lượng filter trong lớp tích chập. Ví dụ, nếu có 10 filter thì sẽ tạo ra 10 feature maps. Những ảnh này sau đó được truyền vào các lớp tiếp theo của mô hình để tiếp tục xử lý. Ban đầu, các trọng số trong filter được khởi tạo ngẫu nhiên và sau đó được điều chỉnh trong quá trình huấn luyện, giúp mô hình học cách nhận diện các đặc trưng chính xác hơn từ dữ liệu. Một hình minh họa của quá trình tích chập với filter 5×5 sẽ thể hiện rõ cách mà từng phần của bức ảnh gốc tương tác với filter để tạo ra feature map tương ứng.

#### Lớp gộp (Polling Layer)

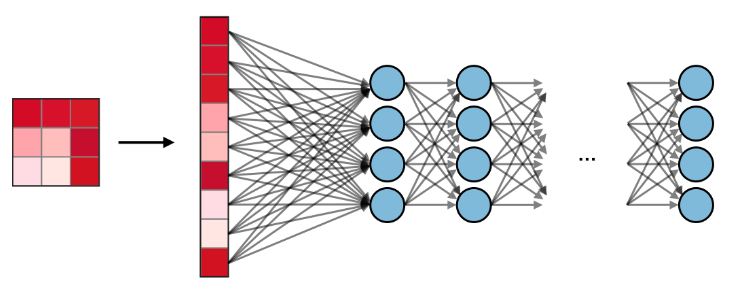
Layer này sử dụng một cửa sổ trượt để quét qua toàn bộ ảnh dữ liệu, mỗi lần trượt theo một bước trượt (stride) cho trước. Khác với layer Convolution, layer Pooling không tính tích chập mà tiến hành lấy mẫu (subsampling). Khi cửa sổ trượt trên ảnh, chỉ có một giá trị được xem là giá trị đại diện cho thông tin ảnh tại vùng đó (giá trị mẫu) được giữ lại. Các phương thức lấy phổ biến trong layer Pooling là MaxPooling (lấy giá trị lớn nhất) và Average Pooling (lấy giá trị trung bình).

Bảng 2.1. Bảng minh họa phương thức Max Pooling và Average Pooling

|  |  |  |
| --- | --- | --- |
| **Kiểu** | Max pooling | Average Pooling |
| **Chức năng** | Từng phép pooling chọn giá trị lớn nhất trong khu vực mà nó đang được áp dụng. | Từng phép pooling tính trung bình các giá trị trong khu vực mà nó đang được áp dụng. |
| **Minh họa** |  |  |
| **Nhận xét** | * Bảo toàn các đặc trưng đã phát hiện. * Được sử dụng thường xuyên. | * Giảm kích thước feature map. * Được sử dụng trong mạng LeNet. |

#### Lớp kết nối đầy đủ (Fully Connected)

Tầng kết nối đầy đủ (FC) nhận đầu vào là các dữ liệu đã được làm phẳng, mà mỗi đầu vào đó được kết nối đến tất cả nơ-ron. Trong mô hình mạng CNN, các tầng kết nối đầy đủ thường được tìm thấy ở cuối mạng và được dùng để tối ưu hóa mục tiêu của mạng ví dụ như độ chính xác của lớp.

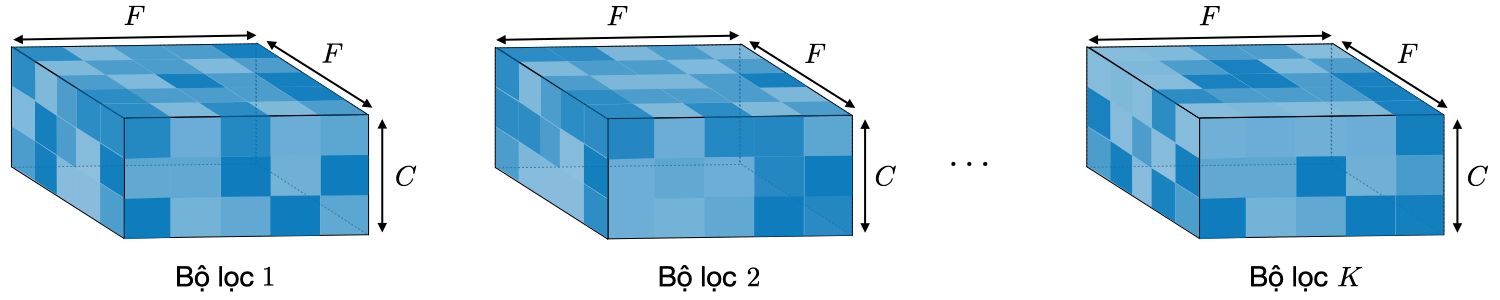


Hình 2.13. Hình ảnh mô phỏng lớp kết nối đầy đủ (Nguồn: Internet)

### Siêu tham số

#### Các siêu tham số của bộ lọc

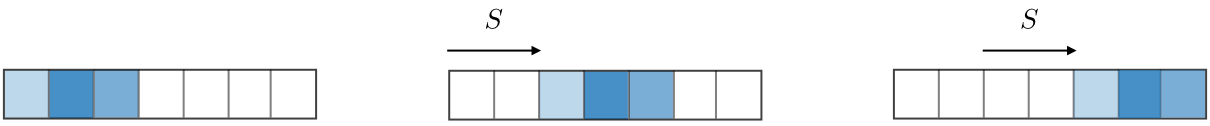
* **Các chiều của một bộ lọc**: Một bộ lọc kích thước áp dụng lên đầu vào chứa kênh (channels) thì có kích thước tổng kể là thực hiện phép tích chập trên đầu vào kích thước và cho ra một feature map (hay còn gọi là activation map) có kích thước .



Hình 2.14. Các chiều của bộ lọc (Nguồn: Internet)

**Lưu ý**: Việc áp dụng bộ lọc có kích thước cho ra một feature map có kích thước .

* **Stride**: Đối với phép tích chập hoặc phép pooling, độ trượt ký hiệu số pixel mà cửa sổ sẽ di chuyển sau mỗi lần thực hiện phép tính.



Hình 2.15. Minh hoạ độ trượt (Nguồn: Internet)

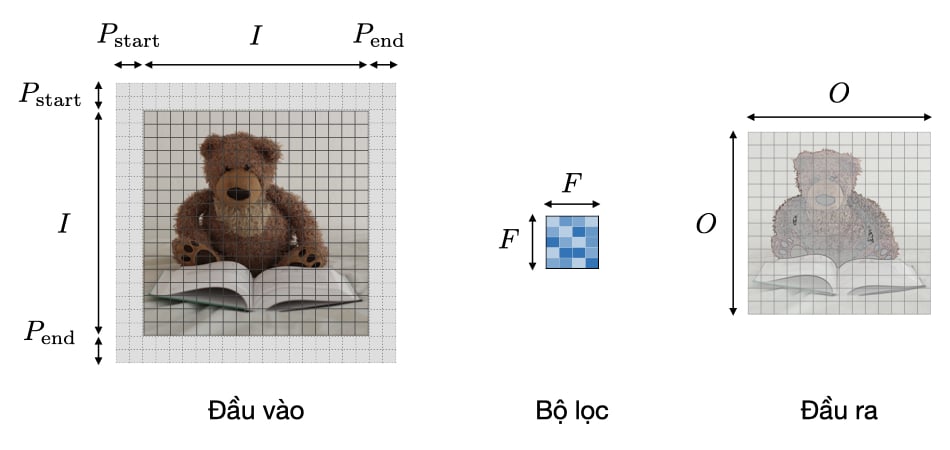
* **Zero-padding**: Zero-padding là tên gọi của quá trình thêm số không vào các biên của đầu vào. Giá trị này có thể được lựa chọn thủ công hoặc một cách tự động bằng một trong ba những phương pháp mô tả bên dưới.

Bảng 2.2. Bảng so sánh 3 phương pháp Zero-padding.

|  |  |  |  |
| --- | --- | --- | --- |
| **Phương pháp** | **Valid** | **Same** | **Full** |
| **Giá trị** |  |  |  |
| **Minh hoạ** | Padding valid | Padding same |  |
| **Mục đích** | * Không sử dụng padding * Bỏ phép tích chập cuối nếu số chiều không khớp | * Sử dụng padding để làm cho feature map có kích thước . * Kích thước đầu ra thuận lợi về mặt toán học. * Còn được gọi là “half” padding. | * Padding tối đa sao cho các phép tích chập có thể được sử dụng tại các rìa của đầu vào. * Bộ lọc “thấy” được đầu vào từ đầu đến cuối. |

#### Điều chỉnh siêu tham số

* **Tính tương thích của tham số trong tầng tích chập**: Bằng cách ký hiệu là độ dài kích thước đầu vào, là độ dài của bộ lọc, là số lượng zero-padding, là độ trượt, ta có thể tính được độ dài của feature map theo một chiều bằng công thức:



Hình 2.16. Tính tương tích của tham số trong tầng tích chập (Nguồn: Internet)

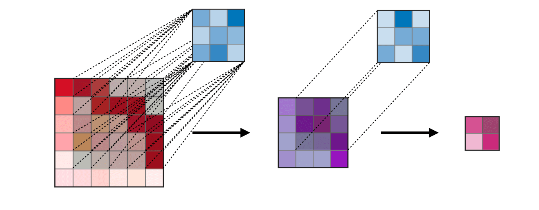
* **Hiểu về độ phức tạp của mô hình**: Để đánh giá độ phức tạp của một mô hình, cách hữu hiệu là xác định số tham số mà mô hình đó sẽ có. Trong một tầng của mạng neural tích chập, nó sẽ được tính toán như sau.

Bảng 2.3. Bảng tính toán độ phức tạp của mô hình

|  |  |  |  |
| --- | --- | --- | --- |
|  | **CONV** | **POOL** | **FC** |
| **Minh họa** |  |  |  |
| **Kích thước đầu vào** |  |  |  |
| **Kích thước đầu ra** |  |  |  |
| **Số lượng tham số** |  |  |  |
| **Lưu ý** | * Một tham số bias với mỗi bộ lọc. * Trong đa số trường hợp, . * Một lựa chọn phổ biến cho là . | * Phép pooling được áp dụng lên từng kênh (channel-wise). * Trong đa số trường hợp, . | * Đầu vào được làm phẳng. * Mỗi nơ-ron có một tham số bias. * Số nơ-ron trong một tầng FC phụ thuộc vào ràng buộc kết cấu. |

* **Trường thụ cảm**: Trường thụ cảm (receptive field) tại tầng là vùng được ký hiệu ​ của đầu vào mà những pixel của activation map thứ kk có thể "nhìn thấy". Bằng cách gọi ​ là kích thước bộ lọc của tầng và ​ là giá trị độ trượt của tầng và để thuận tiện, ta mặc định , trường thụ cảm của tầng được tính toán bằng công thức:

Trong ví dụ bên dưới, ta có và , nên cho ra được .



Hình 2.17. Ví dụ minh hoạ trường thụ cảm (Nguồn: Internet)

### Các hàm kích hoạt thường gặp

* **Rectified Linear Unit**: Tầng rectified linear unit (ReLU) là một hàm kích hoạt được sử dụng trên tất cả các thành phần. Mục đích của nó là tăng tính phi tuyến tính cho mạng. Những biến thể khác của ReLU được tổng hợp ở bảng dưới.

Bảng 2.4. Bảng tổng hợp biến thể của ReLU

|  |  |  |
| --- | --- | --- |
| **ReLU** | **Leaky ReLU** | **ELU** |
| **)** | với | với |
|  | Leaky ReLU | ELU |
| Độ phức tạp phi tuyến tính có thể thông dịch được về mặt sinh học | Gán vấn đề ReLU chết cho những giá trị âm | Khả vi tại mọi nơi |

* **Softmax**: Bước softmax có thể được coi là một hàm logistic tổng quát lấy đầu vào là một vector chứa các giá trị và cho ra là một vector gồm các xác suất thông qua một hàm softmax ở cuối kiến trúc. Nó được định nghĩa như sau:

với

### Ưu điểm

Mạng nơ-ron tích chập (CNN) có nhiều ưu điểm vượt trội, đặc biệt trong các bài toán xử lý ảnh và thị giác máy tính. Trước tiên, CNN có khả năng tự động trích xuất đặc trưng mà không cần can thiệp thủ công. Các bộ lọc tích chập giúp mô hình tự học được các đặc điểm quan trọng trong ảnh như đường viền, góc cạnh và kết cấu ở nhiều cấp độ khác nhau. Điều này giúp CNN hoạt động hiệu quả hơn so với các phương pháp truyền thống vốn phải dựa vào kỹ thuật trích xuất đặc trưng thủ công.

Một ưu điểm quan trọng khác là CNN có khả năng xử lý dữ liệu không gian tốt nhờ cấu trúc tầng tích chập. Mô hình tận dụng tính cục bộ và tính bất biến dịch chuyển, giúp nhận diện đối tượng ngay cả khi chúng bị thay đổi vị trí hoặc có nhiễu nhỏ. Hơn nữa, nhờ cơ chế chia sẻ tham số trong các bộ lọc, CNN giảm đáng kể số lượng tham số cần học so với mạng nơ-ron truyền thống (fully connected network), giúp tiết kiệm bộ nhớ và cải thiện khả năng tổng quát hóa.

CNN cũng có hiệu suất cao trong các bài toán thực tế nhờ khả năng học sâu. Khi được huấn luyện trên tập dữ liệu lớn, mô hình có thể đạt độ chính xác cao trong các tác vụ như nhận diện khuôn mặt, phân loại hình ảnh và phát hiện vật thể. Ngoài ra, CNN có thể mở rộng dễ dàng bằng cách tăng số lượng tầng hoặc sử dụng các kiến trúc tiên tiến như ResNet, VGG, và EfficientNet để tối ưu hóa hiệu suất.

CNN đã được ứng dụng rộng rãi trong nhiều lĩnh vực như y tế, xe tự hành, an ninh và thị giác máy tính. Nhờ khả năng học sâu và tối ưu hóa đặc trưng tự động, CNN trở thành nền tảng cốt lõi trong nhiều công nghệ trí tuệ nhân tạo hiện đại, giúp cải thiện độ chính xác và hiệu suất của các hệ thống nhận diện và phân tích hình ảnh.

### Nhược điểm

Mạng nơ-ron tích chập (CNN) có nhiều ưu điểm trong xử lý ảnh nhưng cũng tồn tại một số nhược điểm đáng kể. Trước tiên, CNN đòi hỏi tài nguyên tính toán lớn, đặc biệt là GPU, để xử lý lượng lớn tham số và phép toán tích chập. Điều này làm tăng chi phí triển khai và gây khó khăn cho các thiết bị có cấu hình thấp. Ngoài ra, CNN cần một lượng dữ liệu lớn để hoạt động hiệu quả. Nếu tập dữ liệu không đủ phong phú, mô hình dễ bị overfitting, dẫn đến khả năng tổng quát hóa kém.

Một nhược điểm khác của CNN là nó chỉ học được các đặc trưng cục bộ thông qua các bộ lọc nhỏ, nên gặp khó khăn khi xử lý mối quan hệ toàn cục trong ảnh. Các mô hình như Vision Transformer đã được phát triển để khắc phục hạn chế này. Bên cạnh đó, CNN dễ bị tấn công bởi adversarial attacks, khi những thay đổi nhỏ không đáng kể với con người có thể khiến mô hình dự đoán sai hoàn toàn. Điều này gây rủi ro lớn trong các ứng dụng quan trọng như nhận diện khuôn mặt hay xe tự hành.

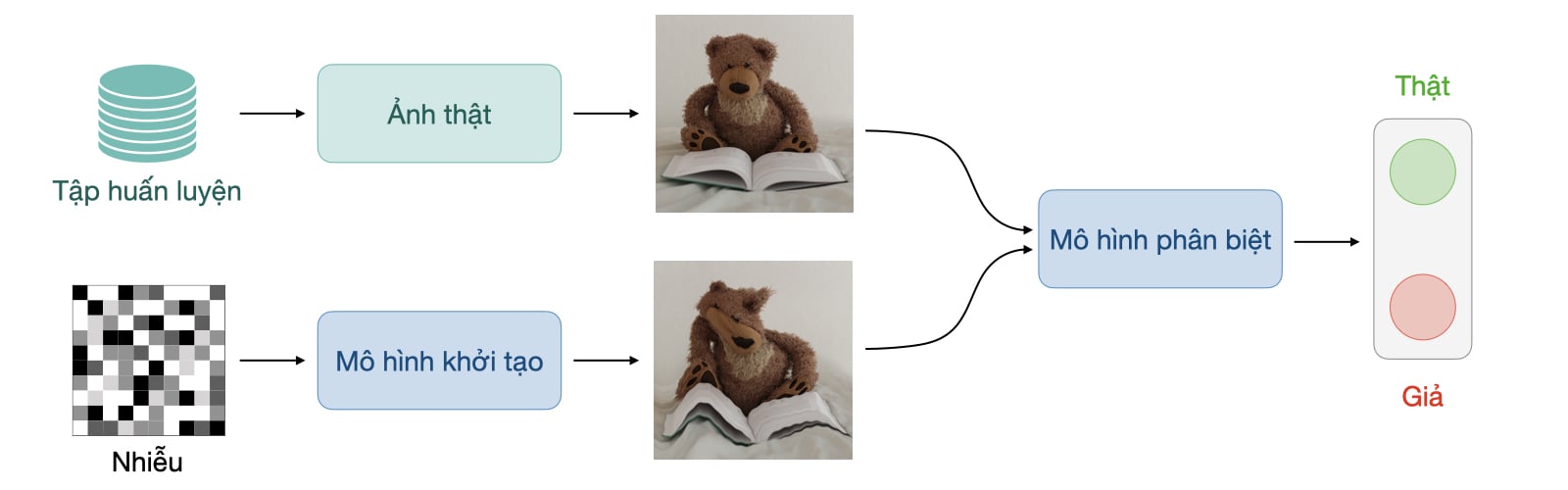
CNN cũng gặp khó khăn trong việc bảo toàn cấu trúc đối xứng và quan hệ hình học. Khi một vật thể bị xoay hoặc thay đổi vị trí, mô hình có thể không nhận diện chính xác nếu không được huấn luyện với đủ dữ liệu biến đổi. Ngoài ra, quá trình tinh chỉnh siêu tham số của CNN rất phức tạp, đòi hỏi nhiều thử nghiệm để tìm ra kiến trúc tối ưu.

Cuối cùng, CNN không tối ưu cho các bài toán yêu cầu xử lý theo thời gian thực. Các mô hình lớn có thể gây độ trễ đáng kể, làm giảm hiệu suất trong các ứng dụng như drone hay robot. Một số phương pháp tối ưu như pruning, quantization có thể giúp giảm kích thước mô hình nhưng lại ảnh hưởng đến độ chính xác. Vì vậy, tùy vào bài toán cụ thể, có thể cần kết hợp CNN với các mô hình khác để cải thiện hiệu suất.

### Các kiến trúc CNN phổ biến

Một số kiến trúc CNN phổ biến được sử dụng trong bài toán nhận diện da tổn thương:

* + **AlexNet**:Một trong những CNN đầu tiên đạt được kết quả vượt trội trong bài toán phân loại hình ảnh.
  + **VGGNet**: Kiến trúc sâu hơn AlexNet, sử dụng nhiều lớp tích chập nhỏ để trích xuất đặc trưng.
  + **GoogLeNet (Inception)**:Kiến trúc này sử dụng những inception module và hướng tới việc thử các tầng tích chập khác nhau để tăng hiệu suất thông qua sự đa dạng của các feature. Cụ thể, kiến trúc này sử dụng thủ thuật tầng tích chập để hạn chế gánh nặng tính toán.
  + **ResNet**:Kiến trúc Residual Network (hay còn gọi là ResNet) sử dụng những khối residual (residual blocks) cùng với một lượng lớn các tầng để giảm lỗi huấn luyện. Những khối residual có những tính chất sau đây:
  + **Generative Adversarial Network**: Generative adversarial networks, hay còn được gọi là GAN, là sự kết hợp giữa mô hình khởi tạo và mô hình phân biệt, khi mà mô hình khởi tạo cố gắng tạo ra hình ảnh đầu ra chân thực nhất, sau đó được đưa vô mô hình phân biệt, mà mục tiêu của nó là phân biệt giữa ảnh được tạo và ảnh thật.



Hình 2.18. Minh hoạ kiến trúc GAN (Nguồn: Internet)

**Lưu ý**: có nhiều loại GAN khác nhau bao gồm từ văn bản thành ảnh, sinh nhạc và tổ hợp.

## Vision Transformer (ViT)

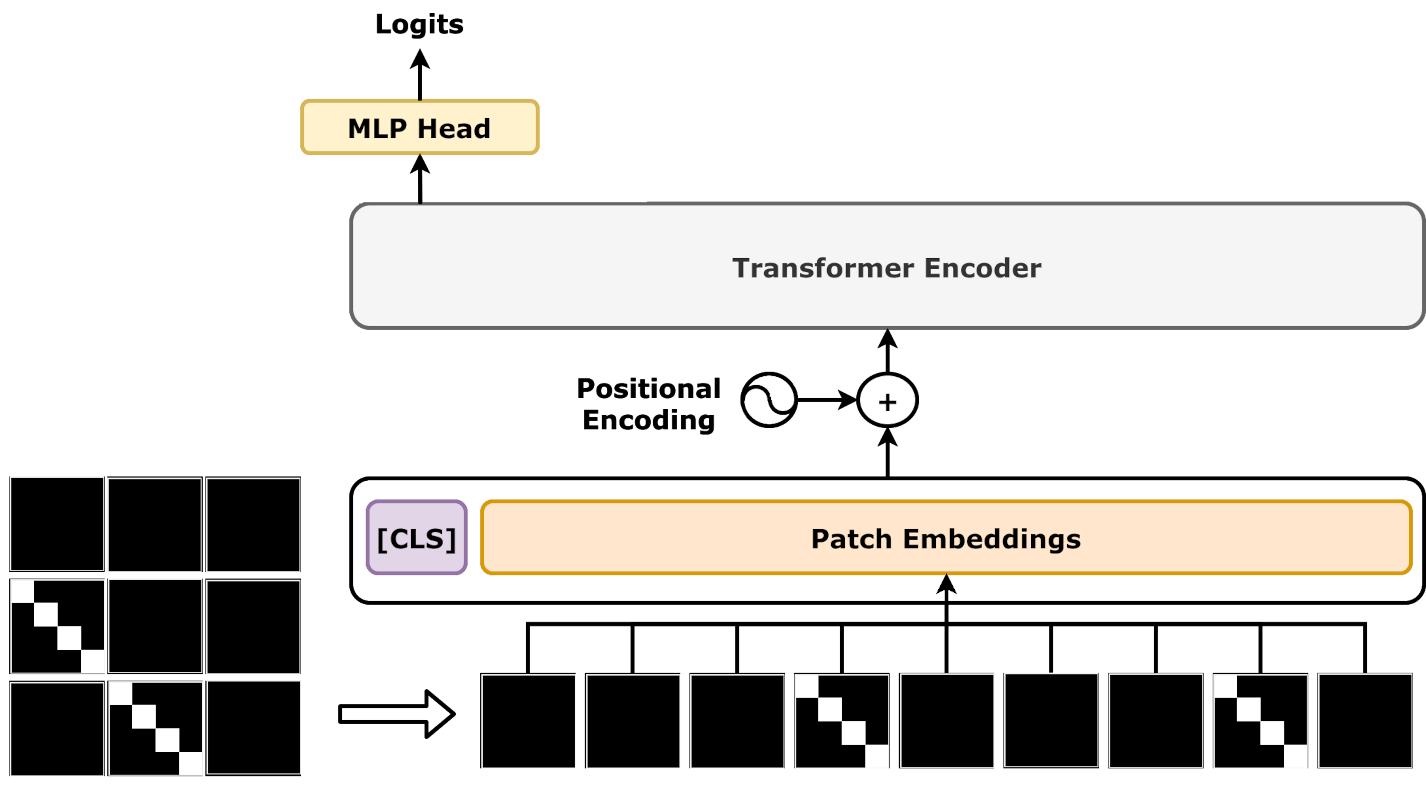
### Giới thiệu

Vision Transformer, hay ViT, là một mô hình phân loại hình ảnh sử dụng kiến trúc giống Transformer áp dụng trên các mảng (patches) của hình ảnh. Một hình ảnh được chia thành các mảng có kích thước cố định, mỗi mảng sau đó được nhúng tuyến tính, thêm các nhúng vị trí, và chuỗi vectơ kết quả được đưa vào một bộ mã hóa Transformer tiêu chuẩn. Để thực hiện phân loại, phương pháp tiêu chuẩn là thêm một “token phân loại” bổ sung có thể học được vào chuỗi vectơ này được sử dụng.

Kiến trúc của Vision Transformer (ViT) là một bước tiến đột phá trong việc áp dụng mô hình Transformer – vốn ban đầu được thiết kế cho xử lý ngôn ngữ tự nhiên (NLP) – vào các tác vụ thị giác máy tính, đặc biệt là phân loại hình ảnh.

### Cấu trúc

Thay vì sử dụng các mạng nơ-ron tích chập (CNN) như cách tiếp cận truyền thống trong thị giác máy tính, ViT chia nhỏ hình ảnh thành các "mảnh" (patches) và xử lý chúng như một chuỗi tuần tự, tương tự cách Transformer xử lý các từ trong một câu. Sau đó, nó sử dụng cơ chế tự chú ý (self-attention) để tìm ra mối quan hệ giữa các mảnh này, từ đó hiểu được nội dung của toàn bộ hình ảnh.



Hình 2.19. Cấu trúc cơ bản của Vision Transformer (Nguồn: Internet)

#### Chia hình ảnh thành patches

* Hình ảnh đầu vào (thường là ảnh RGB) được chia thành các mảnh nhỏ có kích thước cố định, ví dụ 16x16 pixel.
* Mỗi mảnh được "duỗi phẳng" (flattened) thành một vector. Giả sử ảnh có kích thước và mỗi mảnh là , ta sẽ có 196 mảnh .
* Các vector này sau đó được đưa qua một lớp tuyến tính (linear projection) để chuyển thành các embedding có kích thước đồng nhất, gọi là patch embeddings.

#### Thêm thông tin vị trí (Positional Encoding)

* Vì Transformer không có khái niệm thứ tự tự nhiên như CNN, ViT thêm thông tin vị trí (positional embeddings) vào mỗi patch embedding. Điều này giúp mô hình biết được vị trí tương đối của từng mảnh trong hình ảnh gốc.
* Các embedding vị trí này có thể được cố định hoặc học được trong quá trình huấn luyện.

#### Token đặc biệt [CLS]

Một token đặc biệt, gọi là [CLS], được thêm vào đầu chuỗi các patch embeddings. Token này đóng vai trò tổng hợp thông tin từ toàn bộ hình ảnh và thường được sử dụng để đưa ra dự đoán cuối cùng trong tác vụ phân loại.

#### Transformer Encoder

* Chuỗi gồm [CLS] và các patch embeddings được đưa vào một chuỗi các khối Transformer Encoder (giống như trong mô hình BERT).
* Mỗi khối bao gồm:
* **Multi-Head Self-Attention**: Cơ chế chú ý đa đầu giúp mô hình tập trung vào các mối quan hệ giữa các mảnh khác nhau, ví dụ như liên kết giữa mắt và mũi trong ảnh khuôn mặt.
* **Feed-Forward Network (FFN):** Một mạng nơ-ron tiến thẳng áp dụng cho từng embedding.
* **Layer Normalization và Residual Connections**: Giúp ổn định và tăng hiệu quả huấn luyện.
* Quá trình này được lặp lại qua nhiều tầng (layers), thường là 12 hoặc 24 tùy phiên bản.

#### Đầu ra

* Sau khi qua các tầng Transformer, embedding của token [CLS] được lấy ra.
* Embedding này được đưa qua một lớp tuyến tính đơn giản (MLP head) để dự đoán lớp của hình ảnh.

### Ưu điểm

* **Khả năng nắm bắt mối quan hệ toàn cục (Global Context)**: ViT có thể hiểu được mối quan hệ giữa các vùng xa nhau trong hình ảnh ngay từ đầu, không cần phải xây dựng dần như CNN qua các tầng tích chập.
* **Tính linh hoạt**:ViT có thể dễ dàng thích nghi với các kích thước đầu vào khác nhau (bằng cách thay đổi số lượng patches) và áp dụng cho nhiều tác vụ thị giác khác nhau (phân loại, phân đoạn, phát hiện đối tượng) mà không cần thay đổi cấu trúc cơ bản.
* **Hiệu quả cao khi có dữ liệu lớn**: Khi được huấn luyện trên các tập dữ liệu khổng lồ (như JFT-300M), ViT thường vượt trội so với CNN về độ chính xác, đặc biệt trong các bài toán phân loại hình ảnh.
* **Đơn giản hóa kiến trúc**: Không cần các phép toán tích chập phức tạp hay các tầng gộp (pooling), ViT dựa hoàn toàn vào Transformer, giúp dễ hiểu và triển khai hơn trong một số trường hợp.

### Nhược điểm

* **Yêu cầu dữ liệu huấn luyện lớn**: ViT không có các giả định cục bộ như CNN (như tính bất biến với dịch chuyển), nên nó cần rất nhiều dữ liệu để học được các đặc trưng hình ảnh hữu ích. Nếu chỉ huấn luyện trên tập dữ liệu nhỏ (như CIFAR-10), hiệu suất thường kém hơn CNN.
* **Tốn tài nguyên tính toán**: Cơ chế tự chú ý có độ phức tạp tính toán bậc hai với số lượng patches, khiến ViT tiêu tốn nhiều bộ nhớ và thời gian huấn luyện hơn, đặc biệt với hình ảnh độ phân giải cao.
* **Hiệu quả thấp với hình ảnh nhỏ**: Khi số lượng patches ít (ví dụ: ảnh kích thước nhỏ), ViT không tận dụng được hết sức mạnh của self-attention, dẫn đến hiệu suất không tối ưu.
* **Thiếu tính bất biến cục bộ**:Không giống CNN, vốn tự nhiên có khả năng nhận diện đặc trưng bất kể vị trí (translation invariance), ViT cần học điều này từ dữ liệu, đôi khi làm giảm hiệu quả trong các tác vụ cần sự nhạy cảm với cấu trúc cục bộ.

### Các kiến trúc ViT phổ biến

* **ViT gốc (Vision Transformer)**: Mô hình gốc do Google đề xuất, hoạt động tốt trên dữ liệu lớn. Các phiên bản phổ biến: ViT-B (Base), ViT-L (Large), ViT-H (Huge), khác nhau về số tầng và kích thước embedding.
* **DeiT (Data-efficient ViT):** Cải tiến ViT để hoạt động tốt hơn với dữ liệu nhỏ hơn (như ImageNet-1k) bằng cách sử dụng kỹ thuật tăng cường dữ liệu (data augmentation) và một token "distillation" học từ mô hình thầy (teacher model). Tối ưu hóa việc huấn luyện với ít dữ liệu hơn.
* **Swin Transformer**: Chia ảnh thành các cửa sổ (windows) nhỏ và chỉ áp dụng self-attention trong từng cửa sổ, sau đó dùng cơ chế "shifted windows" để kết nối thông tin giữa các cửa sổ. Sử dụng cơ chế attention phân cấp để cải thiện hiệu suất.

# THỰC NGHIỆM

## Kỹ thuật sử dụng

## Môi trường thực nghiệm

Toàn bộ các thực nghiệm được thực hiện trên nền tảng Google Colab, một môi trường notebook dựa trên đám mây phù hợp cho các tác vụ học máy và khoa học dữ liệu.

* **Phần cứng**: Intel Xeon 2vCPUs 2.20 Ghz, 13GB Ram,100GB SSD, GPU NVIDIA T4 16GB VRAM.
* **Phần mềm**: Ubuntu LTS.
* **Môi trường lập trình**:Python 3.10 cùng với các thư viện: tensorflow 2.10,numpy, matplotlib, scikit-learn, pillow.

## Dữ liệu thực nghiệm

### Nguồn gốc và đặc điểm của bộ dữ liệu

#### Nguồn gốc của bộ dữ liệu

#### Đặc điểm của bộ dữ liệu

### Tiền xử lý dữ liệu

#### Đồng nhất kích thước ảnh

#### Tăng cường dữ liệu

## Xây dựng mô hình nơ-ron tích chập

### Kiến trúc mạng nơ-rơn tích chập

### Tham số huấn luyện

### Đánh giá hiệu suất mô hình

### Lưu trữ mô hình

# XÂY DỰNG ỨNG DỤNG CHO MÔ HÌNH NHẬN DẠNG BIỂN BÁO GIAO THÔNG

## Giới thiệu chung

## Kiến trúc hệ thống

## Các chức năng chính của hệ thống

## Kết quả và đánh giá

## Hướng phát triển trong tương lai

KẾT LUẬN

TÀI LIỆU THAM KHẢO

[1]: CS 230 – Mạng nơ-ron tích chập cheatsheet. URL: <https://stanford.edu/~shervine/l/vi/teaching/cs-230/cheatsheet-convolutional-neural-networks>. Lần truy cập gần nhất ngày: 01/06/2025.

[2]: Gonzalez, R. C. và Woods, R. E.: *Digital Image Processing.* Pearson, 2018.

[3]: Nguyễn Văn Hiếu: *Xử lý ảnh số*. Nhà xuất bản Đại học Quốc gia TP.HCM, 2013.

[4]: Nguyễn Đức Tùng: *Nhập môn xử lý ảnh.* Nhà xuất bản Bách Khoa, 2019.

[6]: Nguyễn Hoàng Anh: *Nhập môn học máy.* Nhà xuất bản Đại học Quốc gia TP.HCM, 2020.

[7]: Nguyễn Thanh Bình: *Học sâu trong thị giác máy tính.* Nhà xuất bản Khoa học và Kỹ thuật, 2021.

[8]: Phạm Huy Thông: *Slide bài giảng môn Xử lý ảnh số và thị giác máy tính.* Đại học Công Nghiệp Hà Nội, 2025.

[9]: Lê Thị Thuỷ: *Slide bài giảng môn Khai thác dữ liệu và ứng dụng.* Đại học Công Nghiệp Hà Nội, 2025.