**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

----🙡🕮🙣----

A yellow and red logo

Description automatically generated

BÁO CÁO THÍ NGHIỆM/THỰC NGHIỆM

HỌC PHẦN: TRÍ TUỆ NHÂN TẠO

ĐỀ TÀI: ỨNG DỤNG CNN ĐỂ XÂY DƯNG HỆ THỐNG PHÂN LOẠI GẠO

|  |  |
| --- | --- |
| GVHD: | ThS. Trần Thanh Huân |
| Nhóm - Lớp: | 10 - 20241IT6094003 |
| Thành viên: | 2022600279 – Nguyễn Minh Lộc |
|  | 2022600964 – Trần Thành Long |
|  | 2022602691 – Vương Thành Long |
|  | 2022606152 – Lưu Thị Ngọc Mai |

Hà Nội, Năm 2024

|  |  |
| --- | --- |
| BỘ CÔNG THƯƠNG  TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI | CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  Độc lập – tự do – hạnh phúc  -------o0o------- |

**PHIẾU GIAO ĐỀ TÀI**

**BÁO CÁO THỰC NGHIỆM/THÍ NGHIỆM**

Nhóm thực hiện: 1) Nguyễn Minh Lộc

2) Trần Thành Long

3) Vương Thành Long

4) Lưu Thị Ngọc Mai

Lớp học phần: 20241IT6094003 Khóa: K17 Khoa Công nghệ thông tin

Ngành học: Kỹ thuật phần mềm

Tên đề tài: Ứng dụng CNN để xây dựng hệ thống phân loại gạo

Mục đích: Áp dụng mạng nơ-ron tích chập (CNN) trong việc nhận dạng các loại gạo thông qua hình ảnh với độ chính xác và hiệu quả cao. Xây dựng được hệ thống phân loại gạo với mô hình CNN được lưu lại

Yêu cầu:- Hoàn thành các nhiệm vụ của đề tài

-Sử dụng kỹ năng làm việc nhóm trong quá trình thực hiện báo cáo

-Trình bày báo cáo đúng yêu cầu, sử dụng kỹ năng “Viết báo cáo” khi thực hiện đề tài

-Sử dụng kỹ năng viết tài liệu kỹ thuật và phi kỹ thuật trong phần mở đầu của báo cáo

Kết quả thu được: Bản báo cáo đề tài (bản cứng và bản mềm); sản phẩm đề tài

Ngày giao đề tài: 20/11/2024

Ngày hoàn thành: 20/12/2024

Giảng viên hướng dẫn: Trần Thanh Huân

Hà Nội, Ngày 20 Tháng 12 Năm 2024

GIẢNG VIÊN

Trần Thanh Huân

BÁO CÁO HỌC TẬP CÁ NHÂN/NHÓM

Tên lớp : 20241IT6094003 Khóa : K17

Nhóm 10, gồm 4 thành viên

1. Nguyễn Minh Lộc ( nhóm trưởng )
2. Vương Thành Long
3. Trần Thành Long
4. Lưu Thị Ngọc Mai

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tuần | Người thực hiện | | Nội dung công việc | Kết quả đạt được | Nhận xét của giảng viên |
| 1 | Lưu Thị Ngọc Mai | | Tìm hiểu về sơ đồ hướng giải quyết bài toán | Vẽ được sơ đồ hướng giải quyết bài toán | Không có |
| Trần Thành Long | | Tìm hiểu về CNN và lấy ví dụ | Tìm hiểu được về CNN và lấy được ví dụ |
| Nguyễn Minh Lộc | | Tổng hợp, chỉnh sửa nội dung, làm Powerpoint | Hoàn thiện nội dung, powerpoint |
| Vương Thành Long | | Tìm hiểu về CNN và lấy ví dụ | Tìm hiểu được về CNN và lấy được ví dụ |
|  |
| 2 | Lưu Thị Ngọc Mai | | Tìm hiểu về phương pháp nghiên cứu, đối tượng nghiên cứu | Tìm được phương pháp nghiên cứu, đối tượng nghiên cứu đối với đề tài | Không có |
| Trần Thành Long | | Tìm hiểu về phạm vi nghiên cứu ,lý do chọn đề tài, mục tiêu đề tài | Tìm được phạm vi nghiên cứu đối với đề tài,  hoàn thành nội dung lý do chọn đề tài, mục tiêu chọn đề tài trong quyển báo cáo |
| Nguyễn Minh Lộc | | Viết về những thuật toán có thể sử dụng để xử lý vấn đề của đề tài | Tìm được các thuật toán trích xuất đặc trưng trong xử lý ảnh như Mạng Nơ-ron tích chập, Support Vector Machine, Thuật toán K-Neareast Neighbors,... |
| Vương Thành Long | | Viết mở đầu của quyển báo cáo | Hoàn thành phần mở dầu của quyển báo cáo |
|  |
| 3 | Lưu Thị Ngọc Mai | | Phân tích hình ảnh sau khi qua lớp convolutional đầu tiên và qua lớp maxpooling đầu tiên | Hoàn thành phân tích hình ảnh sau khi qua lớp convolutional đầu tiên và qua lớp maxpooling đầu tiên | Không có |
| Trần Thành Long | | Phân tích hình ảnh sau khi qua lớp convolutional đầu tiên và qua lớp maxpooling đầu tiên | Hoàn thành phân tích hình ảnh sau khi qua lớp convolutional thứ hai và qua lớp maxpooling thứ hai |
| Nguyễn Minh Lộc | | Phân tích hình ảnh sau khi qua lớp convolutional thứ ba và qua lớp maxpooling thứ ba và code CNN | Hoàn thành phân tích hình ảnh sau khi qua lớp convolutional thứ ba và qua lớp maxpooling thứ ba và hoàn thành code CNN |
| Vương Thành Long | | Phân tích hình ảnh sau tiền xử lý | Hoàn thành phân tích hình ảnh sau tiền xử lý |
|
| 4 | Lưu Thị Ngọc Mai | | Phân tích tensor trước và flatten layer và vector sau flatten layer | Hoàn thành phân tích tensor trước và flatten layer và vector sau flatten layer | Không có |
|  | Trần Thành Long | | Tổng hợp hoàn chỉnh báo cáo | Tổng hợp hoàn chỉnh báo cáo |
|  | Nguyễn Minh Lộc | | Tổng hợp hoàn chỉnh báo cáo | Tổng hợp hoàn chỉnh báo cáo |
|  | Vương Thành Long | | Tổng hợp hoàn chỉnh báo cáo và lấy ví dụ tích chập CNN chi tiết | Tổng hợp hoàn chỉnh báo cáo,hoàn thành ví dụ |

**MỤC LỤC**

[DANH MỤC HÌNH ẢNH I](#_Toc184648343)

[DANH MỤC BẢNG BIỂU III](#_Toc184648344)

[DANH MỤC NHỮNG TỪ VIẾT TẮT IV](#_Toc184648345)

[LỜI MỞ ĐẦU V](#_Toc184648346)

[CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc184648347)

[1. Giới thiệu về đề tài 1](#_Toc184648348)

[2. Lý do lựa chọn đề tài 1](#_Toc184648349)

[3. Mục tiêu nghiên cứu 2](#_Toc184648350)

[5. Phương pháp nghiên cứu: 3](#_Toc184648351)

[6. Đối tượng nghiên cứu: 4](#_Toc184648352)

[7. Phạm vi nghiên cứu 4](#_Toc184648353)

[8. Kết luận chương I 4](#_Toc184648354)

[CHƯƠNG 2: CÁC THUẬT TOÁN TRÍCH XUẤT ĐẶC TRƯNG TRONG XỬ LÝ ẢNH 5](#_Toc184648355)

[2.1. Giới thiệu về Trích Xuất Đặc Trưng 5](#_Toc184648356)

[2.2. Các Thuật Toán Trích Xuất Đặc Trưng 5](#_Toc184648357)

[2.2.1. Mạng Nơ-ron Tích Chập (Convolutional Neural Network - CNN) 5](#_Toc184648358)

[2.2.2. Support Vector Machine (SVM) 8](#_Toc184648359)

[2.2.3. Thuật Toán K-Nearest Neighbors (KNN) 10](#_Toc184648360)

[2.2.4. Random Forest 13](#_Toc184648361)

[2.2.5. Transfer Learning 14](#_Toc184648362)

[2.2.6. Mạng Nơ-ron Xoắn Sâu (Deep CNN - DCNN) 15](#_Toc184648363)

[2.2.7. Các Phương Pháp Truyền Thống Khác (SIFT, SURF, HOG) 17](#_Toc184648364)

[2.3. So Sánh và Đánh Giá Các Thuật Toán 19](#_Toc184648365)

[2.4. Kết luận chương II 20](#_Toc184648366)

[CHƯƠNG III: ỨNG DỤNG CNN TRONG PHÂN LOẠI GẠO 21](#_Toc184648367)

[3.1. Quy trình áp dụng CNN trong hệ thống phân loại gạo 21](#_Toc184648368)

[3.1.1. Thuật toán CNN (Convolutional Neural Network) 21](#_Toc184648369)

[3.3.2. Áp dụng thuật toán CNN trong hệ thống phân loại gạo 33](#_Toc184648370)

[3.2. Kết quả đạt được và đánh giá 52](#_Toc184648371)

[3.2.1. Kết quả đạt được 52](#_Toc184648372)

[3.2.2. Đánh giá mô hình 53](#_Toc184648373)

[KẾT LUẬN VÀ KIẾN NGHỊ 56](#_Toc184648374)

[TÀI LIỆU THAM KHẢO 58](#_Toc184648375)

# DANH MỤC HÌNH ẢNH

[Hình 3.1. Mô hình tổng quát của CNN 22](#_Toc184648376)

[Hình 3.2. Hình minh họa cho quá trình trượt của bộ lọc để thu được Feature Map 23](#_Toc184648377)

[Hình 3.3. Hàm Relu 25](#_Toc184648378)

[Hình 3.4. Hình minh họa max pooling và average pooling 26](#_Toc184648379)

[Hình 3.5. Hình minh họa sử dụng fully connected layer 26](#_Toc184648380)

[Hình 3.6. Ma trận đầu vào và kernel 27](#_Toc184648381)

[Hình 3.7. Ma trận đầu vào với padding same 27](#_Toc184648382)

[Hình 3.8. Kernel với bias 27](#_Toc184648383)

[Hình 3.9. Bước đầu tiên của kernel 28](#_Toc184648384)

[Hình 3.10. Bước thứ hai của kernel 28](#_Toc184648385)

[Hình 3.11. Bước thứ ba của kernel 29](#_Toc184648386)

[Hình 3.12. Bước thứ tư của kernel 29](#_Toc184648387)

[Hình 3.13. Bước thứ năm của kernel 30](#_Toc184648388)

[Hình 3.14. Ma trận đầu ra sau khi nhân tích chập 30](#_Toc184648389)

[Hình 3.15. Ma trận đầu ra sau khi dùng hàm ReLU 31](#_Toc184648390)

[Hình 3.16. Sau khi thực hiện max pooling. 31](#_Toc184648391)

[Hình 3.17. Chuyển ma trận thành vector bằng flatten 31](#_Toc184648392)

[Hình 3.18. Sơ đồ giải quyết bài toán 33](#_Toc184648393)

[Hình 3.19. Cấu trúc thư mục dữ liệu 36](#_Toc184648394)

[Hình 3.20. Kết nối colab với drive 36](#_Toc184648395)

[Hình 3.21. Giải nén file dữ liệu 36](#_Toc184648396)

[Hình 3.22. Data Frame chứa các đường dẫn ảnh và nhãn tương ứng 37](#_Toc184648397)

[Hình 3.23. Kết quả Data Frame chứa đường dẫn ảnh và các nhãn tương ứng 37](#_Toc184648398)

[Hình 3.24. Code hiển thị 4 ảnh mỗi loại gạo lên màn hình 38](#_Toc184648399)

[Hình 3.25. Kết quả hiển thị 4 ảnh mỗi loại gạo 38](#_Toc184648400)

[Hình 3.26. Chia dữ liệu thành các tập và mã hóa các nhãn 39](#_Toc184648401)

[Hình 3.27. Tạo Image Data Generator để tăng độ linh hoạt của dữ liệu 40](#_Toc184648402)

[Hình 3.28. Áp dụng Image Data Generator vào các tệp dữ liệu 41](#_Toc184648403)

[Hình 3.29. Xây dựng mô hình CNN 42](#_Toc184648404)

[Hình 3.30. Thông số model CNN được xây dựng 42](#_Toc184648405)

[Hình 3.31. Hình ảnh hiển thị sau lớp conv2d đầu tiên 44](#_Toc184648406)

[Hình 3.32. Hình ảnh hiển thị sau lớp maxpooling đầu tiên 45](#_Toc184648407)

[Hình 3.33. Hình ảnh hiển thị sau lớp conv2d thứ 2 46](#_Toc184648408)

[Hình 3.34. Hình ảnh hiển thị sau lớp maxpooling thứ 2 47](#_Toc184648409)

[Hình 3.35. Hình ảnh hiển thị sau lớp conv2d thứ 3 48](#_Toc184648410)

[Hình 3.36. Hình ảnh hiển thị sau lớp maxpooling thứ 3 49](#_Toc184648411)

[Hình 3.37. Kích thước tensor sau lớp MaxPooling2D thứ ba (trước khi flatten) (1) 50](#_Toc184648412)

[Hình 3.38. Kích thước tensor sau lớp MaxPooling2D thứ ba (trước khi Flatten) (2) 51](#_Toc184648413)

[Hình 3.39. Kích thước tensor sau lớp MaxPooling2D thứ ba (trước khi Flatten) (3) 52](#_Toc184648414)

[Hình 3.40. Vector sau khi Flatten 52](#_Toc184648415)

[Hình 3.41. Chuẩn bị mô hình cho quá trình huấn luyện 52](#_Toc184648416)

[Hình 3.42. Huấn luyện mô hình 52](#_Toc184648417)

[Hình 3.43. Lưu mô hình lại 52](#_Toc184648418)

[Hình 3.44. Dùng mô hình dự đoán 53](#_Toc184648419)

[Hình 3.45. Kết quả dùng mô hình dự đoán 53](#_Toc184648420)

[Hình 3.46. Vẽ đồ thị đánh giá mô hình 54](#_Toc184648421)

[Hình 3.47. Đồ thị đánh giá mô hình qua mỗi epoch 54](#_Toc184648422)

[Hình 3.48. Độ chính xác của mô hình trên tập test 55](#_Toc184648423)

# DANH MỤC BẢNG BIỂU

[Bảng 2.1: So sánh và đánh giá các thuật toán 16](#_Toc184307442)

# DANH MỤC NHỮNG TỪ VIẾT TẮT

1. CNN: Convolutional Neural Network

2. HOG: Histogram of Oriented Gradients

3. KNN: K-Nearest Neighbors

4: SIFT: Scale-Invariant Feature Transform

5. SVM: Support Vector Machine

# LỜI MỞ ĐẦU

Trong bối cảnh phát triển không ngừng của ngành nông nghiệp và công nghệ, việc ứng dụng công nghệ tiên tiến nhằm cải thiện chất lượng và hiệu quả trong sản xuất và phân phối là một xu hướng tất yếu. Trong số đó, gạo – một trong những loại lương thực chính yếu trên toàn thế giới, đặc biệt tại Việt Nam – yêu cầu một quy trình kiểm định chất lượng chính xác và nhanh chóng để đáp ứng nhu cầu tiêu dùng ngày càng cao và khắt khe của thị trường.

Phân loại gạo thủ công truyền thống dựa trên quan sát và kinh nghiệm của con người không những tiêu tốn nhiều thời gian và công sức, mà còn dễ gặp sai sót do các yếu tố chủ quan như sự mệt mỏi hay hạn chế về mắt thường. Với sự phát triển mạnh mẽ của trí tuê nhân tạo (AI) và học máy (machine learning), các hệ thống thông minh có khả năng tự động nhận diện và phân loại hình ảnh đã trở thành công cụ hỗ trợ đắc lực trong nhiều lĩnh vực, từ y tế, công nghiệp cho đến nông nghiệp. Đăc biệt, mạng nơ-ron tích chập (Convolutional Neural Network – CNN) nổi lên như một công nghệ đột phá với khả năng xử lý và phân tích hình ảnh hiệu quả, được ứng dụng rộng rãi trong các bài toán nhận diện hình ảnh và phân loại đối tượng.

Đề tài “Ứng dụng CNN để xây dựng hệ thống phân loại gạo” nhằm nghiên cứu và phát triển một mô hình sử dụng CNN để phân loại các loại gạo khác nhau dựa trên hình ảnh chụp. Hệ thống này hứa hẹn sẽ mang lại nhiều lợi ích thiết thực cho ngành nông nghiệp và công nghiệp chế biến lương thực, góp phần nâng cao chất lượng và hiệu quả của quá trình sản xuất và kinh doanh gạo.  
(Cấu trúc báo cáo gồm 3 chương)

Chương 1: Tổng quan về đề tài

Lý do lựa chọn đề tài, mục tiêu đề tài

Phương pháp nghiên cứu, đối tượng nghiên cứu

Phạm vi nghiên cứu của đề tài

Chương 2: Tìm hiểu thuật toán

Trình bày các thuật toán có thể xử dụng để xử lý vấn đề của đề tài

Chương 3: Ứng dụng CNN trong phân loại gạo

Quy trình áp dụng CNN trong xử lý vấn đề

Kết quả đạt được và đánh giá

# CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI

## 1. Giới thiệu về đề tài

Tình hình nghiên cứu ở nước ngoài: Việc ứng dụng mạng nơ-ron tích chập (CNN) vào phân loại gạo đã trở thành một hướng đi nghiên cứu rất được quan tâm trong những năm gần đây ở nước ngoài. Với khả năng tự động trích xuất các đặc trưng hình ảnh một cách hiệu quả, CNN đã chứng minh được tiềm năng lớn trong việc phân biệt các loại gạo khác nhau dựa trên hình ảnh. Các hệ thống phân loại gạo dựa trên CNN đã được triển khai trong một số nhà máy chế biến gạo, giúp tăng năng suất và đảm bảo chất lượng sản phẩm.

Tình hình nghiên cứu ở trong nước: Việt Nam, với tư cách là một quốc gia có nền nông nghiệp phát triển mạnh mẽ, đặc biệt là sản xuất lúa gạo, việc ứng dụng công nghệ AI, cụ thể là CNN vào phân loại gạo đang là một hướng đi nghiên cứu rất được quan tâm. Mặc dù tiềm năng rất lớn, nhưng việc nghiên cứu và ứng dụng CNN vào phân loại gạo tại Việt Nam vẫn còn ở giai đoạn đầu.

## 2. Lý do lựa chọn đề tài

Lý do từ môn học: Môn học Trí tuệ nhân tạo yêu cầu việc thực hiện bài tập lớn nhằm giúp sinh viên hiểu rõ hơn về các kỹ năng cơ bản trong thực hiện và cách áp dụng các kiến thức về trí tuệ nhân tạo vào thực tế, đồng thời rèn luyện kỹ năng tự học, kỹ năng nghiên cứu, khả năng giao tiếp bằng văn bản kỹ thuật và phi kỹ thuật. Bài tập lớn này giúp sinh viên có thể tìm hiểu, nghiên cứu và áp dụng một phương pháp trong trí tuệ nhân tạo vào giải quyết bài toán trong thực tế.

Lý do lựa chọn xây dựng ứng dụng phân loại gạo: Gạo là một trong những thực phẩm được tiêu thụ nhiều nhất trên thế giới, được giao dịch nhiều nhất và là nền tảng cho nền kinh tế của các nước sản xuất. Việc phân loại gạo thủ công vốn rất tốn kém và mất nhiều thời gian. Vì vậy, trí tuệ nhân tạo được sử dụng trong lĩnh vực này có thể giúp phân loại lúa gạo cách dễ dàng. Phương pháp này mang lại nhiều lợi ích về mặt kinh tế nông nghiệp.

Lý do lựa chọn mô hình CNN trong xây dựng ứng dụng phân loại gạo: CNN có khả năng tự động học và trích xuất các đặc trưng hình ảnh phức tạp của hạt gạo như hình dạng, kích thước, màu sắc, vân bề mặt,... mà con người khó nhận biết. Nhờ khả năng học sâu, CNN có thể đạt được độ chính xác rất cao trong việc phân loại các loại gạo khác nhau, kể cả những loại gạo có hình dạng và màu sắc tương tự. Hơn thế nữa, đây là một phương pháp có tốc độ xử lý nhanh với sự hỗ trợ của các GPU, CNN có thể xử lý một lượng lớn ảnh gạo trong thời gian ngắn, giúp tăng năng suất và hiệu quả trong quá trình phân loại. Ngoài ra, CNN có thể dễ dàng được đào tạo lại để phân loại các loại gạo mới hoặc các đặc tính khác của gạo, giúp đáp ứng nhu cầu thay đổi của thị trường.

Do đó, việc ứng dụng CNN để xây dựng hệ thống phân loại gạo giúp áp dụng được các kiến thức về trí tuệ nhân tạo, cụ thể là phương pháp CNN vào thực tế, giải quyết bài toán phân loại gạo. Đây cũng là cơ hội để tìm hiểu sâu hơn về mô hình CNN và lĩnh vực sản xuất với quy trình phân loại gạo. Chúng em sẽ có cơ hội học hỏi từ thực tế, nâng cao, phát triển và hoàn thiện hơn kỹ năng và kiến thức của mình và tiếp thu các kỹ năng và kiến thức mới.

## 3. Mục tiêu nghiên cứu

Mục tiêu đề tài: Nghiên cứu cách nhập xuất file ảnh, cách tiền xử lý dữ liệu là ảnh. Nghiên cứu hoạt động của mô hình CNN, cấu trúc CNN gồm trích xuất đặc trưng và phân loại. Các thành phần trong phần trích xuất đặc trưng: lớp tích chập, lớp ReLU, lớp pooling. Các thành phần trong phần phân loại: flatten layer, fully connected layer, softmax layer.

Nghiên cứu quá trình huấn luyện mô hình với hàm optimizer và hàm mất mát. Số lần huấn luyện mô hình epoch, batch size.

Nghiên cứu quá trình nhận dạng hạt gạo, các bước chính trong hệ thống phân loại gạo. Xây dựng mô hình CNN với các lớp tích chập, ReLU, lớp pooling, flatten layer, fully connected layer, softmax layer. Huấn luyện mô hình và đánh giá độ chính xác.

Áp dụng mạng nơ-ron tích chập (CNN) trong việc nhận dạng các loại gạo thông qua hình ảnh với độ chính xác và hiệu quả cao. Xây dựng được hệ thống phân loại gạo với mô hình CNN được lưu lại và có thể sử dụng trong tương lai.

Rèn luyện khả năng làm việc nhóm và phối hợp giải quyết vấn đề, khả năng tìm tòi, học hỏi, tự nghiên cứu.

Mục tiêu kinh tế - xã hội: Mục tiêu kinh tế xã hội của đề tài là nâng cao chất lượng sản phẩm gạo, tăng năng suất và hiệu quả sản xuất trong phân loại gạo, sản phẩm gạo chấp lượng cao đáp ứng tiêu chuẩn quốc tế giúp mở rộng thị trường và tăng tính hiệu quả trong việc phân loại gạo.

Bảo vệ người sản xuất và người tiêu dùng khỏi các loại gạo không đúng như mong muốn. Giúp người tiêu dùng dễ dàng phân biệt các loại gạo khác nhau, từ đó lựa chọn được loại gạo phù hợp với nhu cầu và sở thích của mình, đảm bảo chất lượng bữa ăn. Ứng dụng giúp nhà sản xuất xây dựng hình ảnh thương hiệu chuyên nghiệp, minh bạch, tạo lòng tin cho người tiêu dùng.

Mục tiêu khoa học – công nghệ: Mục tiêu khoa học công nghệ của đề tài là xây dựng một mô hình CNN có khả năng phân loại chính xác các loại gạo khác nhau dựa trên đặc trưng hình ảnh. CNN có khả năng tự động trích xuất các đặc trưng hình ảnh nào quan trọng nhất trong việc phân loại gạo. Phát triển một hệ thống hoàn chỉnh, bao gồm phần thu thập dữ liệu, tiền xử lý dữ liệu, xây dựng và huấn luyện mô hình.

## 5. Phương pháp nghiên cứu:

Phân tích chủ đề: Xác định các nội dung và yêu cầu cụ thể cho việc xây dựng chương trình  phân loại gạo bằng mô hình mạng nơ-ron tích chập (CNN).

Nghiên cứu CNN và ứng dụng: Tìm hiểu về các mô hình CNN, các ưu điểm khi áp dụng trong nhận dạng và phân loại hình ảnh, đặc biệt là trong phân loại nông sản như gạo.

Tìm hiểu dữ liệu và chuẩn bị dữ liệu: Thu thập các hình ảnh về hạt gạo từ các nguồn khác nhau, đảm bảo tính đa dạng (gạo thường, gạo nứt, gạo bị hư hỏng, v.v.), và xử lý dữ liệu để phù hợp với mô hình CNN (chỉnh sửa kích thước, cân bằng dữ liệu).

Tìm hiểu Python và các thư viện liên quan: Tìm hiểu về Python và các thư viện hỗ trợ như TensorFlow, Keras để hỗ trợ việc phân loại ảnh và xây dựng CNN.

Xây dựng và huấn luyện mô hình: Phân tích các cấu trúc CNN phù hợp, xây dựng mô hình CNN và tiến hành huấn luyện với dữ liệu đã chuẩn bị.

Đánh giá hiệu suất mô hình: Sử dụng các chỉ số đánh giá độ chính xác, độ nhạy  để kiểm tra hiệu suất mô hình. Điều chỉnh các tham số nếu cần thiết.

Viết báo cáo và đánh giá: Tổng hợp kết quả, đánh giá độ hiệu quả của mô hình CNN trong việc phân loại gạo và ghi nhận các kết quả vào báo cáo.

## 6. Đối tượng nghiên cứu:

Bài toán sẽ tập trung vào việc phân loại 5 loại gạo khác nhau thông qua hình ảnh, áp dụng mô hình mạng nơ-ron tích chập CNN để phục vụ cho quá trình chẩn đoán chất lượng và phân loại gạo trong lĩnh vực nông nghiệp.

## 7. Phạm vi nghiên cứu

Áp dụng trong hệ thống phân loại gạo với các mẫu gạo Arborio, Basmati, Ipsala, Jasmine, Karacadag. Hệ thống phân loại gạo này có thể được ứng dụng trong các nhà máy sản xuất gạo hoặc cơ sở đóng gói để phân loại tự động theo loại gạo, đảm bảo chất lượng và sự đồng nhất.

## 8. Kết luận chương I

Chương I đã giới thiệu tổng quan về đề tài, làm rõ tầm quan trọng của việc ứng dụng công nghệ trí tuệ nhân tạo, đặc biệt là mô hình CNN, trong lĩnh vực phân loại và kiểm định chất lượng gạo. Phạm vi và mục tiêu nghiên cứu cùng được xác định rõ ràng để định hướng toàn bộ nội dung tiếp theo

# CHƯƠNG 2: CÁC THUẬT TOÁN TRÍCH XUẤT ĐẶC TRƯNG TRONG XỬ LÝ ẢNH

## 2.1. Giới thiệu về Trích Xuất Đặc Trưng

Trích xuất đặc trưng là một bước quan trọng trong quy trình xử lý ảnh, nhằm chuyển đổi thông tin từ ảnh thành các biểu diễn số mà các thuật toán máy học có thể hiểu và xử lý. Các đặc trưng này có thể là các thông tin về hình dạng, màu sắc, kết cấu và các yếu tố khác của ảnh. Áp dụng các thuật toán khác nhau để trích xuất đặc trưng giúp cải thiện hiệu suất của các mô hình phân loại và nhận diện trong nhiều ứng dụng khác nhau.

## 2.2. Các Thuật Toán Trích Xuất Đặc Trưng

### 2.2.1. Mạng Nơ-ron Tích Chập (Convolutional Neural Network - CNN)

1. Giới thiệu Mạng Nơ-ron Tích Chập (CNN)

Mạng Nơ-ron Tích Chập (Convolutional Neural Network - CNN) là một kiến trúc mạng nơ-ron nhân tạo được phát triển đặc biệt để xử lý dữ liệu dưới dạng hình ảnh hoặc các dữ liệu có cấu trúc dạng lưới (như tín hiệu hình ảnh hoặc tín hiệu âm thanh). CNN có khả năng tự động trích xuất các đặc trưng từ ảnh đầu vào và đặc biệt hiệu quả trong các bài toán nhận diện, phân loại và phân đoạn hình ảnh.

CNN khác với mạng nơ-ron truyền thống (Fully Connected Neural Network) ở chỗ không yêu cầu thủ công trích xuất các đặc trưng mà sử dụng các lớp tích chập để tự động học các đặc trưng này từ dữ liệu.

2. Cấu trúc Cơ Bản của CNN

CNN bao gồm một chuỗi các lớp xử lý chính, mỗi lớp đảm nhiệm vai trò khác nhau trong việc trích xuất và phân tích các đặc trưng từ ảnh đầu vào. Các lớp chính bao gồm:

Lớp Tích Chập (Convolutional Layer)

Lớp Gộp (Pooling Layer)

Lớp Kích Hoạt (Activation Layer)

Lớp Kết Nối Đầy Đủ (Fully Connected Layer)

Dưới đây là chi tiết về từng lớp này:

a. Lớp Tích Chập (Convolutional Layer)

Lớp tích chập là thành phần chính trong CNN, có nhiệm vụ trích xuất các đặc trưng từ hình ảnh đầu vào. Trong lớp này, các bộ lọc (filter hay kernel) sẽ trượt qua ảnh để tìm các đặc trưng (như cạnh, góc, kết cấu), từ đó tạo ra các ma trận đặc trưng.

Bộ lọc (Filter): Là một ma trận nhỏ (ví dụ: 3x3, 5x5) được sử dụng để tích chập với một vùng của ảnh.

Tích chập (Convolution): Bộ lọc được di chuyển qua ảnh và tại mỗi vị trí, tính tích chập giữa giá trị ảnh và giá trị bộ lọc. Kết quả là một ma trận đặc trưng (feature map).

Thông số Stride: Là khoảng cách di chuyển của bộ lọc trên ảnh.

Padding: Là việc thêm các giá trị (thường là 0) xung quanh biên của ảnh để đảm bảo kích thước của ma trận đặc trưng không bị giảm quá nhiều.

b. Lớp Gộp (Pooling Layer)

Lớp gộp có nhiệm vụ giảm độ phân giải của các ma trận đặc trưng, giúp giảm số lượng tham số và độ phức tạp tính toán. Điều này giúp CNN nhanh hơn và giảm nguy cơ quá khớp (overfitting). Phương pháp gộp phổ biến nhất là Max Pooling - chỉ lấy giá trị lớn nhất trong vùng gộp.

Max Pooling: Lấy giá trị lớn nhất từ mỗi vùng.

Average Pooling: Lấy giá trị trung bình của các giá trị trong vùng.

c. Lớp Kích Hoạt (Activation Layer)

Lớp kích hoạt áp dụng một hàm phi tuyến lên các ma trận đặc trưng để tăng khả năng học các đặc trưng phức tạp hơn. Hàm kích hoạt phổ biến nhất trong CNN là ReLU (Rectified Linear Unit), chuyển đổi các giá trị âm thành 0, giữ nguyên các giá trị dương.

ReLU: f(x)=max(0,x)

Leaky ReLU: Giữ các giá trị âm ở một giá trị nhỏ thay vì chuyển toàn bộ thành 0.

d. Lớp Kết Nối Đầy Đủ (Fully Connected Layer)

Lớp kết nối đầy đủ là lớp cuối cùng trong CNN, nơi các ma trận đặc trưng đã được làm phẳng (flattened) thành một vector và đưa vào một mạng nơ-ron truyền thống để thực hiện phân loại dựa trên các đặc trưng đã học được. Lớp này thực hiện quá trình ra quyết định và dự đoán nhãn cho ảnh đầu vào.

3. Cách CNN Hoạt Động trong Trích Xuất Đặc Trưng

Quy trình hoạt động của CNN trong trích xuất đặc trưng bao gồm các bước chính:

Bước 1: Ảnh đầu vào được đưa qua lớp tích chập, và các bộ lọc sẽ phát hiện các đặc trưng như cạnh và kết cấu trong ảnh.

Bước 2: Ảnh đặc trưng sau lớp tích chập được đưa qua lớp gộp để giảm độ phân giải.

Bước 3: Các đặc trưng phi tuyến và phức tạp hơn được học thông qua các lớp tích chập và lớp kích hoạt.

Bước 4: Ảnh được làm phẳng và đưa qua lớp kết nối đầy đủ để phân loại thành các nhãn tương ứng.

4. Ứng Dụng CNN trong Phân Loại Gạo

Trong bài toán phân loại gạo, CNN có thể tự động học các đặc trưng từ ảnh gạo (như hình dạng, màu sắc, kết cấu). Các đặc trưng này rất hữu ích để phân biệt các loại gạo khác nhau. Khi được huấn luyện với tập dữ liệu hình ảnh lớn và đa dạng, CNN sẽ học cách nhận diện và phân loại gạo chính xác.

5. Ưu Điểm và Nhược Điểm của CNN

Ưu điểm:

Tự động trích xuất đặc trưng từ ảnh mà không cần phải xử lý đặc trưng thủ công.

Hiệu quả và chính xác cao trong nhận diện và phân loại ảnh.

Có khả năng mở rộng với các mô hình sâu và kiến trúc tiên tiến hơn.

Nhược điểm:

Đòi hỏi tài nguyên tính toán lớn và thời gian huấn luyện lâu.

Dễ bị quá khớp nếu tập dữ liệu huấn luyện không đủ lớn.

Cần tối ưu hóa các tham số để đạt hiệu quả tốt nhất.

6. Một Số Kiến Trúc CNN Phổ Biến

LeNet: Kiến trúc đơn giản, thích hợp cho các bài toán ít phức tạp.

VGGNet: Sâu hơn và mạnh mẽ hơn với nhiều lớp tích chập, phù hợp cho các bài toán có yêu cầu cao về độ chính xác.

ResNet: Sử dụng các kết nối tắt, giúp học các đặc trưng phức tạp mà không gặp phải vấn đề mất mát gradient.

Inception: Sử dụng nhiều kích thước bộ lọc khác nhau, giúp phát hiện đặc trưng ở nhiều tỷ lệ khác nhau.

### 2.2.2. Support Vector Machine (SVM)

1. Giới thiệu

Support Vector Machine - SVM là một thuật toán học máy mạnh mẽ được sử dụng chủ yếu trong các bài toán phân loại và hồi quy. SVM nổi bật với khả năng xác định ranh giới phân loại tối ưu giữa các lớp dữ liệu, từ đó đạt độ chính xác cao cho các bài toán phân loại phức tạp.

2. Cách Hoạt Động của SVM

SVM hoạt động bằng cách tìm kiếm một siêu phẳng tối ưu (optimal hyperplane) trong không gian đặc trưng để phân biệt các lớp dữ liệu. Trong không gian hai chiều, siêu phẳng này là một đường thẳng, trong khi ở không gian cao hơn, nó có thể là một mặt phẳng hoặc một siêu phẳng.

Siêu phẳng tối ưu: SVM chọn một siêu phẳng sao cho lề (margin) giữa nó và các điểm dữ liệu gần nhất của mỗi lớp là lớn nhất, từ đó giúp phân tách các lớp một cách hiệu quả.

Support vectors: Các điểm dữ liệu gần siêu phẳng nhất trong mỗi lớp được gọi là support vectors. Đây là các điểm quyết định vị trí của siêu phẳng và khoảng cách giữa chúng.

3. SVM trong Trích Xuất Đặc Trưng Ảnh

Trong các ứng dụng xử lý ảnh, SVM thường được sử dụng để phân loại sau khi các đặc trưng từ ảnh đã được trích xuất. Điều này cho phép SVM không cần phân tích toàn bộ ảnh mà chỉ tập trung vào các đặc trưng nổi bật. Một phương pháp phổ biến là kết hợp SVM với các đặc trưng như:

HOG (Histogram of Oriented Gradients): HOG chia ảnh thành các vùng và tính toán histogram của các hướng gradient, giúp phát hiện các kết cấu và hình dạng trong ảnh.

SIFT (Scale-Invariant Feature Transform) và SURF (Speeded-Up Robust Features): Các đặc trưng này cũng thường được sử dụng để nhận diện hình dạng bất biến với kích thước hoặc xoay ảnh.

4. Ví dụ Ứng Dụng của SVM

SVM là lựa chọn phổ biến trong các bài toán xử lý ảnh và nhận diện đối tượng do độ chính xác cao. Một số ứng dụng nổi bật bao gồm:

Phân loại văn bản: SVM được ứng dụng nhiều trong phân loại văn bản, ví dụ như phân loại email spam và không spam.

Nhận diện chữ viết tay: SVM có thể nhận diện ký tự viết tay, ứng dụng trong các hệ thống nhận diện ký tự quang học (OCR).

Phân loại hình ảnh: SVM phân loại ảnh dựa trên các đặc trưng như HOG, giúp phân loại nhanh và chính xác.

Nhận diện khuôn mặt: SVM sử dụng các đặc trưng trên khuôn mặt để phân loại hình ảnh khuôn mặt, được ứng dụng trong xác thực danh tính.

5. Ưu Điểm và Nhược Điểm của SVM

Ưu điểm:

Khả năng phân loại tốt, đặc biệt trên các tập dữ liệu nhỏ và phân tách rõ ràng.

Độ chính xác cao trong các bài toán phân loại với dữ liệu không gian lớn.

Sử dụng hiệu quả với dữ liệu phi tuyến nhờ vào các hàm kernel (nhân), giúp biến đổi dữ liệu vào không gian có thể phân tách.

Nhược điểm:

Không hiệu quả trên các tập dữ liệu lớn do chi phí tính toán cao.

Khó xử lý khi dữ liệu không phân tách rõ ràng.

Việc chọn kernel phù hợp có thể phức tạp và yêu cầu thử nghiệm.

### 2.2.3. Thuật Toán K-Nearest Neighbors (KNN)

1. Giới Thiệu K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) là một thuật toán máy học đơn giản nhưng hiệu quả, chủ yếu được sử dụng cho các bài toán phân loại và hồi quy. KNN thuộc loại thuật toán dựa trên ví dụ (instance-based learning), nghĩa là nó không có giai đoạn huấn luyện trước mà thực hiện dự đoán trực tiếp dựa trên dữ liệu đầu vào. Do đó, KNN còn được gọi là thuật toán *lazy learning* vì nó không xây dựng mô hình trước khi có yêu cầu phân loại hoặc dự đoán.

2. Cách Hoạt Động của KNN

KNN dựa vào khoảng cách giữa các điểm dữ liệu trong không gian đặc trưng để đưa ra dự đoán. Dưới đây là các bước cơ bản để KNN hoạt động:

Bước 1: Xác định giá trị của K, tức là số lượng hàng xóm gần nhất mà thuật toán sẽ xét đến.

Bước 2: Tính khoảng cách giữa điểm cần dự đoán và tất cả các điểm trong tập dữ liệu. Phương pháp tính khoảng cách thường là khoảng cách Euclid​ hoặc các khoảng cách khác như khoảng cách Manhattan hay khoảng cách cosine.

Bước 3: Chọn K điểm gần nhất, tức là K điểm có khoảng cách nhỏ nhất với điểm cần phân loại.

Bước 4: Dự đoán lớp của điểm mới bằng cách lấy đa số phiếu từ các điểm hàng xóm gần nhất nếu là bài toán phân loại, hoặc trung bình giá trị nếu là bài toán hồi quy.

3. Ứng Dụng KNN trong Xử Lý Ảnh

Trong xử lý ảnh, KNN thường được sử dụng để phân loại ảnh hoặc nhận dạng đối tượng. Để thực hiện điều này:

Trước tiên, các đặc trưng của ảnh sẽ được trích xuất, chẳng hạn như các đặc trưng về màu sắc, hình dạng, hoặc các đặc trưng HOG (Histogram of Oriented Gradients).

Sau đó, KNN sẽ dựa trên các đặc trưng này để phân loại ảnh vào các nhóm khác nhau. Ví dụ, trong hệ thống phân loại gạo, các đặc trưng về hình dạng hạt gạo có thể được sử dụng làm dữ liệu đầu vào cho KNN.

4. Ví Dụ Ứng Dụng của KNN

KNN đã được ứng dụng trong nhiều lĩnh vực:

Nhận dạng ký tự chữ viết tay: KNN sử dụng hình dạng và cấu trúc của ký tự để phân loại các chữ cái viết tay.

Phân loại hình ảnh: Trong phân loại ảnh, KNN có thể phân nhóm các ảnh dựa trên đặc trưng trích xuất từ các đối tượng trong ảnh.

Hệ thống gợi ý: KNN còn được áp dụng trong các hệ thống gợi ý, ví dụ như đề xuất sản phẩm cho người dùng dựa trên hành vi mua hàng tương tự.

5. Ưu Điểm và Nhược Điểm của KNN

Ưu điểm:

KNN là thuật toán đơn giản, dễ triển khai, không cần giai đoạn huấn luyện.

Linh hoạt trong việc thay đổi giá trị Kvà có thể áp dụng với nhiều loại dữ liệu.

* Nhược điểm:

KNN chậm khi dữ liệu lớn, vì phải tính khoảng cách cho từng điểm trong tập dữ liệu.

Nhạy cảm với các đặc trưng không liên quan, vì tất cả các đặc trưng đều có trọng số như nhau.

Hiệu quả phụ thuộc vào giá trị K và phương pháp chuẩn hóa dữ liệu, yêu cầu tối ưu hóa trước khi đạt hiệu quả cao.

### 2.2.4. Random Forest

1. Giới thiệu

Random Forest là một thuật toán học máy mạnh mẽ, được sử dụng rộng rãi cho các bài toán phân loại và hồi quy. Nó hoạt động dựa trên nguyên tắc tập hợp (ensemble), kết hợp nhiều cây quyết định (decision trees) để cải thiện độ chính xác và khả năng tổng quát của mô hình.

2. Cách hoạt động

Tập hợp cây quyết định: Random Forest xây dựng một tập hợp gồm nhiều cây quyết định. Mỗi cây trong rừng được tạo ra từ một mẫu con ngẫu nhiên của dữ liệu huấn luyện.

Lựa chọn đặc trưng ngẫu nhiên: Trong quá trình tạo ra mỗi cây, thuật toán lựa chọn ngẫu nhiên một tập hợp con của các đặc trưng để tìm kiếm đặc trưng phân tách tốt nhất. Điều này giúp tăng cường tính đa dạng cho các cây và giảm thiểu hiện tượng overfitting.

Dự đoán: Đối với bài toán phân loại, Random Forest sử dụng phương pháp bỏ phiếu (voting) từ tất cả các cây để đưa ra dự đoán cuối cùng. Đối với bài toán hồi quy, nó tính giá trị trung bình từ các dự đoán của các cây.

3. Trích xuất đặc trưng

Random Forest có thể tự động chọn và đánh giá tầm quan trọng của các đặc trưng trong dữ liệu, giúp xác định các đặc trưng nào có ảnh hưởng lớn đến kết quả phân loại.

4. Ví dụ ứng dụng

Phân loại hình ảnh: Phân loại đối tượng trong các bức ảnh.

Chẩn đoán y tế: Dự đoán bệnh dựa trên các đặc trưng lâm sàng.

Phân tích tài chính: Dự đoán rủi ro tín dụng và phân loại khách hàng.

5. Ưu điểm

Khả năng xử lý tốt với dữ liệu lớn và nhiều đặc trưng.

Giảm thiểu hiện tượng overfitting nhờ vào tính chất ngẫu nhiên và phương pháp bỏ phiếu.

Độ chính xác cao và hiệu suất ổn định.

6. Nhược điểm

Mô hình có thể trở nên phức tạp và khó hiểu hơn so với một cây quyết định đơn lẻ.

Thời gian huấn luyện và dự đoán có thể kéo dài hơn so với một cây quyết định đơn giản.

### 2.2.5. Transfer Learning

1. Giới thiệu

Transfer Learning là một phương pháp trong học máy cho phép sử dụng các mô hình đã được huấn luyện sẵn trên các bộ dữ liệu lớn để giải quyết các bài toán khác nhau mà không cần phải huấn luyện lại từ đầu. Phương pháp này tiết kiệm thời gian và tài nguyên tính toán, đồng thời cải thiện độ chính xác của mô hình trong các bài toán cụ thể.

2. Nguyên tắc hoạt động

Mô hình đã được huấn luyện sẵn: Transfer Learning thường sử dụng các mô hình như mạng nơ-ron tích chập (CNN) đã được huấn luyện trên các bộ dữ liệu lớn như ImageNet, nơi chứa hàng triệu hình ảnh và hàng ngàn lớp khác nhau.

Tinh chỉnh mô hình: Sau khi tải mô hình đã huấn luyện, người dùng có thể tinh chỉnh (fine-tune) mô hình này bằng cách thêm các lớp mới và điều chỉnh trọng số cho các lớp đã có để phù hợp hơn với bài toán cụ thể. Việc này có thể được thực hiện với lượng dữ liệu huấn luyện nhỏ hơn nhiều so với việc huấn luyện từ đầu.

3. Lợi ích

Tiết kiệm thời gian: Transfer Learning giúp giảm thời gian huấn luyện mô hình vì bạn không cần phải bắt đầu từ đầu.

Độ chính xác cao hơn: Sử dụng mô hình đã được huấn luyện trên dữ liệu lớn giúp cải thiện độ chính xác trong các bài toán mà bạn có ít dữ liệu.

Giảm thiểu hiện tượng overfitting: Bằng cách sử dụng mô hình đã được huấn luyện sẵn, bạn có thể giảm thiểu nguy cơ overfitting, đặc biệt là khi làm việc với các tập dữ liệu nhỏ.

4. Ví dụ ứng dụng

Phân loại hình ảnh: Sử dụng CNN đã được huấn luyện trên ImageNet để phân loại các loại gạo khác nhau.

Nhận diện đối tượng: Ứng dụng Transfer Learning trong việc phát hiện và nhận diện các đối tượng trong các bức ảnh hoặc video.

Xử lý ngôn ngữ tự nhiên: Sử dụng các mô hình như BERT hoặc GPT đã được huấn luyện sẵn cho các tác vụ phân tích văn bản hoặc chatbot.

5. Công cụ và thư viện hỗ trợ

TensorFlow và Keras: Cung cấp nhiều mô hình đã được huấn luyện sẵn mà bạn có thể sử dụng cho học chuyển giao.

PyTorch: Cũng cung cấp các mô hình và cách dễ dàng để thực hiện tinh chỉnh.

### 2.2.6. Mạng Nơ-ron Xoắn Sâu (Deep CNN - DCNN)

1. Giới thiệu

Mạng nơ-ron xoắn sâu (Deep Convolutional Neural Network - DCNN) là một biến thể của mạng nơ-ron tích chập (CNN) với nhiều lớp hơn, cho phép mô hình học được các đặc trưng phức tạp hơn từ dữ liệu đầu vào. DCNN đã trở thành một trong những kiến trúc quan trọng nhất trong học sâu, đặc biệt là trong các ứng dụng xử lý ảnh.

2. Cấu trúc và Nguyên tắc hoạt động

Nhiều lớp tích chập: DCNN bao gồm nhiều lớp tích chập liên tiếp, mỗi lớp học được các đặc trưng khác nhau từ dữ liệu. Các lớp đầu tiên thường học các đặc trưng cơ bản như cạnh và hình dạng, trong khi các lớp sau học các đặc trưng phức tạp hơn.

Chất lượng đặc trưng: Sự gia tăng số lượng lớp cho phép DCNN khai thác được chiều sâu của dữ liệu, giúp cải thiện khả năng phân loại và phát hiện đối tượng trong hình ảnh.

Pooling: Các lớp pooling (thường là max pooling) được sử dụng để giảm kích thước của đặc trưng đầu ra từ các lớp tích chập, giúp giảm tính toán và ngăn ngừa hiện tượng overfitting.

3. Lợi ích của DCNN

Khả năng tổng quát tốt hơn: Với khả năng học các đặc trưng phức tạp từ dữ liệu, DCNN có khả năng tổng quát tốt hơn trong việc phân loại và nhận diện đối tượng.

Hiệu suất cao trong phân loại hình ảnh: DCNN đã chứng minh được hiệu suất vượt trội trong nhiều cuộc thi phân loại hình ảnh, chẳng hạn như ImageNet.

Khả năng xử lý dữ liệu lớn: DCNN có thể xử lý và học từ các tập dữ liệu lớn một cách hiệu quả.

4. Ứng dụng

Nhận diện và phân loại hình ảnh: DCNN được sử dụng rộng rãi trong nhận diện đối tượng, phân loại hình ảnh, và nhận diện khuôn mặt.

Phát hiện bất thường: Áp dụng trong các bài toán phát hiện bất thường trong dữ liệu hình ảnh, chẳng hạn như trong ngành y tế để phát hiện bệnh lý từ hình ảnh y khoa.

Xử lý video: DCNN cũng có thể được áp dụng trong việc phân tích video, chẳng hạn như nhận diện hoạt động trong video.

### 2.2.7. Các Phương Pháp Truyền Thống Khác (SIFT, SURF, HOG)

1. Giới thiệu

Các phương pháp trích xuất đặc trưng truyền thống như SIFT (Scale-Invariant Feature Transform), SURF (Speeded-Up Robust Features), và HOG (Histogram of Oriented Gradients) từng là những kỹ thuật phổ biến để trích xuất đặc trưng hình ảnh trước khi học sâu ra đời. Những phương pháp này vẫn được ứng dụng rộng rãi trong nhiều bài toán xử lý ảnh và máy tính thị giác nhờ khả năng hoạt động hiệu quả trong các tình huống nhất định và khả năng tính toán nhanh.

2. Các thuật toán

a) SIFT (Scale-Invariant Feature Transform)

Nguyên lý hoạt động: SIFT tìm kiếm các điểm đặc trưng (keypoints) trong hình ảnh không phụ thuộc vào thay đổi về kích thước, góc xoay, hay ánh sáng. Các điểm này sau đó được mô tả bằng một vector có kích thước cố định, giúp mô hình dễ dàng nhận diện các đặc trưng tương tự trong các ảnh khác nhau.

Quy trình:

Tìm kiếm các keypoints bằng cách áp dụng Gaussian Difference of Gaussian (DoG) để xác định các vị trí nổi bật.

Mô tả các keypoints bằng các vector đặc trưng để so sánh và nhận dạng.

Ứng dụng: SIFT thường được sử dụng trong các bài toán ghép ảnh, nhận diện đối tượng, và theo dõi chuyển động.

b) SURF (Speeded-Up Robust Features)

Nguyên lý hoạt động: SURF là một phiên bản tối ưu hóa của SIFT, được thiết kế để nhanh hơn và tính toán hiệu quả hơn, đặc biệt trong các ứng dụng thời gian thực. Thuật toán sử dụng bộ lọc Haar-wavelet và lọc Gaussian để phát hiện và mô tả các điểm đặc trưng.

Quy trình:

Xác định các keypoints bằng cách tính toán hàm Hessian và áp dụng bộ lọc Haar-wavelet để phân tích hướng của điểm ảnh.

Vector đặc trưng được xây dựng dựa trên giá trị hướng của mỗi keypoint.

Ứng dụng: SURF được ứng dụng trong nhiều lĩnh vực như nhận diện đối tượng, ghép ảnh, và ứng dụng xe tự lái nhờ vào tốc độ xử lý nhanh.

c) HOG (Histogram of Oriented Gradients)

Nguyên lý hoạt động: HOG trích xuất đặc trưng của hình ảnh bằng cách phân tích các hướng gradient trong ảnh. Thuật toán chia hình ảnh thành các ô (cells) và tính histogram của hướng gradient trong từng ô.

Quy trình:

Tính toán gradient của mỗi pixel trong ảnh.

Chia ảnh thành các ô nhỏ và tính histogram của hướng gradient trong mỗi ô.

Kết hợp các histogram thành một vector đặc trưng đại diện cho toàn bộ hình ảnh.

Ứng dụng: HOG là một phương pháp phổ biến trong nhận diện người và phân loại đối tượng trong ảnh.

3. So sánh

SIFT: Hoạt động tốt với các biến đổi về kích thước, xoay, và ánh sáng, nhưng tính toán khá tốn thời gian.

SURF: Tối ưu tốc độ hơn SIFT, phù hợp cho các ứng dụng thời gian thực.

HOG: Tập trung vào trích xuất hướng của gradient, hoạt động hiệu quả trong các bài toán nhận diện người.

4. Ứng dụng

Các phương pháp truyền thống này thường được sử dụng như bước tiền xử lý để trích xuất đặc trưng từ ảnh trước khi áp dụng các thuật toán học máy. Chúng cũng được dùng trong các hệ thống không yêu cầu độ phức tạp của mạng học sâu, hoặc các tình huống cần đặc trưng cố định.

## 2.3. So Sánh và Đánh Giá Các Thuật Toán

|  |  |  |  |
| --- | --- | --- | --- |
| Thuật toán | Ưu điểm | Nhược điểm | Ứng dụng điển hình |
| CNN | Tự động trích xuất đặc trưng, hiệu quả cao | Cần nhiều dữ liệu huấn luyện | Phân loại ảnh y tế, nhận diện khuôn mặt |
| SVM | Tốt với dữ liệu nhỏ | Khó mở rộng với dữ liệu lớn | Phân loại văn bản, nhận diện chữ viết tay |
| KNN | Đơn giản, dễ triển khai | Chậm với dữ liệu lớn | Phân loại ảnh động vật |
| Random Forest | Chính xác, ít bị overfitting | Cần nhiều thời gian huẩn luyện | Phân tích dữ liệu y tế |
| Transfer learning | Tiết kiệm thời gian huấn luyện | Phụ thuộc vào mô hình đã huấn luyện | Nhận diện đối tượng |
| DCNN | Trích xuất đặc trưng phức tạp | Tốn nhiều tài nguyên tính toán | Nhận diện và phân loại hình ảnh phức tạp |

Bảng 2.1: So sánh và đánh giá các thuật toán

## 2.4. Kết luận chương II

Chương II cung cấp nền tảng lý thuyết trích suất đặc trưng từ các phương pháp truyền thống đến sự khác biệt vượt trội của CNN. Những kiến thức này giúp lảm rõ lý do lựa chọn CNN làm công cụ chính trong nghiên cứu

# CHƯƠNG III: ỨNG DỤNG CNN TRONG PHÂN LOẠI GẠO

## 3.1. Quy trình áp dụng CNN trong hệ thống phân loại gạo

### 3.1.1. Thuật toán CNN (Convolutional Neural Network)

3.3.1.1 Tổng quan về CNN

Thuật toán CNN, hay Convolutional Neural Network (Mạng Nơ-ron Tích Chập). Đây là một loại mô hình học sâu rất mạnh mẽ trong lĩnh vực trí tuệ nhân tạo, đặc biệt là trong xử lý hình ảnh. CNN giúp xây dựng các hệ thống thông minh với độ chính xác cao, nhờ khả năng nhận diện và phân tích các đặc điểm quan trọng trong ảnh.

CNN hoạt động bằng cách xử lý dữ liệu hình ảnh thông qua các lớp tích chập, giúp trích xuất các đặc điểm nổi bật từ hình ảnh một cách hiệu quả. Ví dụ, thuật toán này thường được sử dụng để nhận diện khuôn mặt, phân loại đối tượng, và nhiều ứng dụng khác. Những nền tảng nổi tiếng như Facebook và Google cũng đã tích hợp CNN để cải thiện khả năng nhận diện hình ảnh trên các dịch vụ của họ.

Về mặt kỹ thuật, khi một hình ảnh được đưa vào hệ thống CNN, nó sẽ trải qua một loạt các bước. Đầu tiên, hình ảnh sẽ được xử lý qua các lớp tích chập với các bộ lọc để trích xuất các đặc điểm. Sau đó, dữ liệu sẽ đi qua các lớp kết nối đầy đủ và cuối cùng là lớp phân loại sử dụng hàm Softmax. Nó đưa ra xác suất cho các loại đối tượng khác nhau. Kết quả cuối cùng sẽ cho chúng ta biết khả năng thuộc về từng loại của đối tượng trong hình ảnh.

A diagram of a network

Description automatically generated

1. Mô hình tổng quát của CNN

3.3.1.2. Các lớp cơ bản của mạng CNN

1. Convolutional layer

Convolutional layer là lớp quan trọng nhất trong CNN, đảm nhiệm vai trò thực hiện các phép tính chính. Những yếu tố quan trọng của lớp này bao gồm stride, padding, filter map, và feature map.

Filter Map: Đây là các bộ lọc được áp dụng lên từng vùng của hình ảnh. Mỗi filter map là một ma trận 3 chiều chứa các tham số được biểu diễn dưới dạng số.

Stride: Đây là bước dịch chuyển của filter map trên hình ảnh, dịch từ trái sang phải theo từng pixel dựa trên giá trị đã xác định.

Padding: Đây là các giá trị 0 sẽ được thêm vào lớp input ở viền ảnh để giữ kích thước của ảnh không bị thay đổi.

Feature Map: Sau mỗi lần filter map quét qua input, một quá trình tính toán diễn ra. Và feature map chính là kết quả của quá trình này. Nó thể hiện các đặc trưng đã được trích xuất từ hình ảnh ban đầu.

Convolution (lớp tích chập) là lớp đầu tiên trích xuất các đặc tính từ hình ảnh. Tham số lớp này bao gồm một tập hợp các bộ lọc có thể học được. Các bộ lọc đều nhỏ thường có kích cỡ hai chiều đầu tiên khoảng 3x3 hoặc 5x5, .... và có độ sâu bằng với độ sâu của đầu vào đầu vào. Bằng cách trượt dần bộ lọc theo chiều ngang và dọc trên ảnh, chúng thu được một Feature Map chứa các đặc trưng được trích xuất từ trên hình ảnh đầu vào.

A screenshot of a computer

Description automatically generated

1. Hình minh họa cho quá trình trượt của bộ lọc để thu được Feature Map

Nhìn vào hình ví dụ ở trên, chúng ta có 2 filter map màu hồng là W0 và W1. Mỗi filter map có kích thước 3\*3\*3. Kích thước dài \* rộng ( 3\*3 ) là hyperparameter, còn chiều sâu (Depth) sẽ bằng với depth của input volumn, tức là bằng 3.

Stride: Dịch filter map theo pixel từ trái sang phải theo từng dòng. Mỗi lần dịch như thế sẽ dựa vào 1 giá trị gọi là stride.

Ví dụ: stride = 1, thì mỗi lần dịch filter map sẽ sang phải 1 pixel, khi hết cạnh biên phải thì xuống 1 dòng và dịch tiếp. Còn nếu stride = 2 thì mỗi lần dịch sẽ sang phải 2 pixel, khi hết cạnh thì xuống 2 dòng.

Padding: người ta sẽ thêm những giá trị 0 bao quanh lớp input. Ví dụ như ở trên hình, lớp input của chúng ta ban đầu có kích thước 5\*5\*3, nhưng vì giá trị padding = 1, nên được bao thêm 1 lớp 0 bên ngoài. Từ đó, kích thước của lớp input = 7\*7\*3.

Feature map: feature map thực chất là kết quả sau khi lớp input được filter map quét qua hết. Với mỗi lần filter map áp lên input, sẽ có quá trình tính toán xảy ra. Thực chất đó là quá trình nhân 2 ma trận. Ví dụ như trên hình, khi lớp trên cùng của filter map W0 áp vào lớp trên cùng của input volum, ma trận filtermap w0 3\*3 sẽ được nhân với ma trận 3\*3 của input sau đó cộng thêm bias=1, và đưa ra kết quả = 2 ở feature map. Mỗi lần 1 filter map trượt hết lên input volum sẽ cho ra 1 lớp của feature map. Vậy nên, depth của feature map sẽ bằng số lượng filter map.

Ví dụ như hình trên, depth của feature map = số lượng filter map = 2. Còn chiều dài và chiều rộng của feature map sẽ được tính bằng công thức: (W + 2P - F)/S +1 . Trong đó, W là kích thước input, P là padding, F là kích thước filter map, S là stride.  
 Ví dụ với hình, W = 5, P =1, F = 3, S =1 , ta tính được kích thước của feature map là 3.

2. Relu layer

Relu layer, hay còn gọi là hàm kích hoạt (activation function), đóng vai trò quan trọng trong mạng nơ-ron nhân tạo. Nó mô phỏng hoạt động của các neuron thần kinh bằng cách truyền tín hiệu qua axon. Ngoài Relu, còn có các hàm kích hoạt khác như Tanh, Sigmoid, Maxout, và Leaky Relu.

Relu layer được ứng dụng phổ biến trong quá trình huấn luyện mạng nơ-ron nhờ vào hiệu quả và các ưu điểm vượt trội, giúp mô hình học nhanh hơn và chính xác hơn.

f(x)=max(0,x). Hàm này lấy giá trị của x nếu x>0. Nếu x<0 0.



1. Hàm Relu

Thường thì sau khi feature map được tính ra, người ta sẽ xếp theo sau đó 1 lớp ReLU, lớp này sẽ áp dụng hàm ReLU lên tất cả các giá trị của feature map.

3. Pooling layer

Pooling layer giúp giảm kích thước đầu vào khi dữ liệu quá lớn. Nó được đặt giữa các lớp Convolutional layer để giảm số lượng tham số cần tính toán.

Có hai loại phổ biến là max pooling và average pooling. Max pooling lấy giá trị lớn nhất trong vùng dữ liệu, trong khi average pooling tính giá trị trung bình. Cả hai đều giúp giảm tải cho mô hình và tăng hiệu quả xử lý.

A diagram of a pool game

Description automatically generated

1. Hình minh họa max pooling và average pooling

4. Fully connected layer

Fully connected layer đảm nhiệm vai trò xuất kết quả sau khi ảnh đã được xử lý qua các lớp convolutional và pooling. Khi mô hình đã đọc được thông tin từ ảnh, lớp này tạo ra sự kết nối để sinh ra nhiều output hơn. Lập trình viên sử dụng fully connected layer để tổng hợp và xử lý dữ liệu cuối cùng. Ngoài ra, nếu lớp này nhận dữ liệu về hình ảnh, nó sẽ chuyển thành các mục phân loại nhằm phân tích sâu hơn.

Sau khi ảnh được truyền qua nhiều convolutional layer và pooling layer thì model đã học được tương đối các đặc điểm của ảnh (ví dụ mắt, mũi, khung mặt,…) thì tensor của output của layer cuối cùng, kích thước H\*W\*D, sẽ được chuyển về 1 vector kích thước (H\*W\*D)

A close-up of a map

Description automatically generated

1. Hình minh họa sử dụng fully connected layer

3.3.1.3. Ví dụ cụ thể cho các bước tích chập trong CNN

Xét input đầu vào là ma trận 4x4 và kernel 3x3 như sau:

A grid with numbers and letters

Description automatically generated with medium confidence

1. Ma trận đầu vào và kernel

Ta thực hiện thêm padding bằng 0 cho ma trận đầu vào, kích thước ma trận lúc này là 5x5.

A grid of numbers and letters

Description automatically generated

1. Ma trận đầu vào với padding same

Kernel với bias = 1.

A grid with numbers and letters

Description automatically generated

1. Kernel với bias

Stride = 1, thì mỗi lần dịch kernel sang phải 1 pixel, khi hết cạnh biên phải thì xuống 1 dòng và dịch tiếp.

Ta tính kích thước ma trận đầu ra theo công thức sau:

Output\_size = (Input\_size - Kernel\_size + 2 \* Padding) / Stride + 1

Input size là ma trận: 4x4. Ta được Input\_size = 4.

Kernel size là ma trận: 3x3. Ta được Kernel\_size = 3.

Padding 0 được thêm vào ma trận đầu vào. Ta được Padding = 1.

Stride = 1.

Như vậy: Output\_size = (4 – 3 + 2\*1)/1 + 1 = 4.

Sau khi nhân tích chập ta được ma trận đầu ra kích thước 4x4. Ta sẽ thực hiện nhân tích chập ma trận sau khi thêm padding với kernel có bias là 1. Bước đầu tiên của kernel:

A grid with numbers and symbols

Description automatically generated

1. Bước đầu tiên của kernel

Ta có: 0\*1 + 0\*1 + 0\*1 + 0\*0 + (–1)\*1 + 4\*0 + 0\*0 + (–1)\*1 + 4\*0 + 1 = –1.

Bước thứ hai của kernel sau khi trượt sang phải 1 pixel.

A grid with yellow squares and black squares

Description automatically generated

1. Bước thứ hai của kernel

Ta có: 0\*1 + 0\*1 + 0\*1 + (–1)\*0 + 4\*1 + (–4)\*0 + (–1)\*0 + 4\*1 + 3\*0 + 1 = 9.

Bước thứ ba của kernel sau khi trượt sang phải 1 pixel.

A grid with green squares and numbers

Description automatically generated

1. Bước thứ ba của kernel

Ta có: 0\*1 + 0\*1 + 0\*1 + 4\*0 + (–4)\*1 + (–4)\*0 + 4\*0 + 3\*1 + (–2)\*0 + 1 = 0.

Bước thứ tư của kernel sau khi trượt sang phải 1 pixel.

A grid with numbers and symbols

Description automatically generated

1. Bước thứ tư của kernel

Ta có: 0\*1 + 0\*1 + 0\*1 + (–4)\*0 + (–4)\*1 + 0\*0 + 3\*0 + (–2)\*1 + 0\*0 + 1 = –5.

Bước thứ năm của kernel do đã dịch chuyển đến cuối của ma trận nên ta di chuyển kernel xuống dòng và về vị trí biên bên trái để tiếp tục.

A grid with green squares and numbers

Description automatically generated

1. Bước thứ năm của kernel

Ta có: 0\*1 + (–1)\*1 + 4\*1 + 0\*0 + (–1)\*1 + 4\*0 + 0\*0 + 2\*1 + (–5)\*0 + 1 = 5.

Các bước tiếp theo của kernel ta làm tương tự, ta thu được ma trận đầu ra kích thước 4x4 như sau:

A grid of numbers and letters

Description automatically generated

1. Ma trận đầu ra sau khi nhân tích chập

Dùng hàm ReLU, f(x)=max(0,x). Pixel có giá trị lớn hơn hoặc bằng 0 thì sẽ giữ nguyên giá trị, pixel có giá trị nhỏ hơn 0 sẽ có giá trị bằng 0. Ta thu được kết quả sau khi dùng hàm ReLU.

A grid with numbers and letters

Description automatically generated

1. Ma trận đầu ra sau khi dùng hàm ReLU

Thực hiện max pooling trên ma trận này với kernel 2x2, ta sẽ lấy giá trị lớn nhất trong các ô được xét. Kết quả thu được như sau.

A grid with numbers and a grid with text

Description automatically generated with medium confidence

1. Sau khi thực hiện max pooling.

Với flatten ta chuyển đổi ma trận thành 1 vector.

A grid with different colored squares

Description automatically generated

1. Chuyển ma trận thành vector bằng flatten

3.3.1.3. Ứng dụng của CNN

Nhận diện khuôn mặt: CNN được sử dụng trong các hệ thống nhận diện khuôn mặt để xác định và phân tích các đặc điểm trên khuôn mặt người, giúp bảo mật, xác thực.

Phân loại hình ảnh: Trong các ứng dụng thương mại điện tử, CNN có thể phân loại các sản phẩm dựa trên hình ảnh, giúp người tiêu dùng dễ dàng tìm kiếm và lựa chọn.

Phân tích y tế: CNN được sử dụng để phân tích hình ảnh y tế như X-quang, MRI, giúp phát hiện sớm các bệnh lý như ung thư hoặc tổn thương.

Nhận diện đối tượng: Trong lĩnh vực ô tô tự lái, CNN giúp nhận diện và phân loại các đối tượng xung quanh như xe, người đi bộ, và biển báo giao thông.

Tìm kiếm hình ảnh: Các công cụ tìm kiếm hình ảnh sử dụng CNN để nhận diện và phân loại hình ảnh tương tự, giúp người dùng tìm kiếm thông tin dễ dàng hơn.

Tạo hình ảnh (Image Generation): CNN có thể được kết hợp với các mô hình khác để tạo ra hình ảnh mới từ dữ liệu đầu vào, ví dụ như trong nghệ thuật hoặc trò chơi.

Phát hiện hành vi:Trong video giám sát, CNN có thể phát hiện và phân tích hành vi của con người, giúp đảm bảo an ninh và quản lý hiệu quả.

Dịch ngôn ngữ: CNN cũng có thể được áp dụng trong các mô hình dịch hình ảnh thành văn bản, chẳng hạn như nhận diện chữ viết tay hoặc dịch biển báo.

### 3.3.2. Áp dụng thuật toán CNN trong hệ thống phân loại gạo

3.3.2.1. Sơ đồ giải quyết ài toán

A diagram of a diagram

Description automatically generated with medium confidence

1. Sơ đồ giải quyết bài toán

3.3.3.2. Quy trình giải quyết bài toán

1. Công cụ sử dụng

1.1 Ngôn ngữ

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML). Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.

Những lợi ích của Python bao gồm:

Dễ dàng đọc và hiểu một chương trình Python vì có cú pháp cơ bản giống tiếng Anh.

Cải thiện năng suất làm việc vì có thể sử dụng ít dòng mã hơn để viết một chương trình Python.

Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ.

Các nhà phát triển có thể dễ dàng sử dụng Python với các ngôn ngữ lập trình phổ biến khác như Java, C và C++.

Cộng đồng Python tích cực hoạt động bao gồm hàng triệu nhà phát triển nhiệt tình hỗ trợ trên toàn thế giới. Nếu gặp phải vấn đề, bạn sẽ có thể nhận được sự hỗ trợ nhanh chóng từ cộng đồng.

Trên Internet có rất nhiều tài nguyên hữu ích nếu bạn muốn học Python. Ví dụ: bạn có thể dễ dàng tìm thấy video, chỉ dẫn, tài liệu và hướng dẫn dành cho nhà phát triển.

Python có thể được sử dụng trên nhiều hệ điều hành máy tính khác nhau, chẳng hạn như Windows, macOS, Linux và Unix.

1.2 Thư viện

Matplotlib: Các nhà phát triển sử dụng Matplotlib để hiển thị dữ liệu dưới dạng đồ họa hai và ba chiều (2D và 3D) chất lượng cao. Thư viện này thường được sử dụng trong các ứng dụng khoa học. Với Matplotlib, bạn có thể trực quan hóa dữ liệu bằng cách hiển thị dữ liệu dưới dạng các biểu đồ khác nhau, chẳng hạn như biểu đồ cột và biểu đồ đường. Bạn cũng có thể hiển thị nhiều biểu đồ cùng lúc và các chi tiết đồ họa có thể được di chuyển qua mọi nền tảng.

Pandas: Pandas cung cấp cấu trúc dữ liệu được tối ưu hóa và linh hoạt mà bạn có thể sử dụng để thao tác với dữ liệu chuỗi thời gian và dữ liệu có cấu trúc, chẳng hạn như bảng và nhóm. Ví dụ, bạn có thể sử dụng Pandas để đọc, ghi, hợp nhất, lọc và nhóm dữ liệu. Thư viện này được nhiều người sử dụng cho các tác vụ khoa học dữ liệu, phân tích dữ liệu và ML.

NumPy: NumPy là một thư viện phổ biến mà các nhà phát triển sử dụng để dễ dàng tạo và quản lý nhóm, thao tác với các hình dạng logic và thực hiện các phép toán đại số tuyến tính. NumPy hỗ trợ tích hợp với nhiều ngôn ngữ như C và C ++.

OpenCV-Python: OpenCV-Python là một thư viện mà các nhà phát triển sử dụng để xử lý hình ảnh cho các ứng dụng thị giác máy tính. Thư viện này cung cấp nhiều hàm cho các tác vụ xử lý hình ảnh như đọc và ghi hình ảnh cùng lúc, xây dựng môi trường 3D từ môi trường 2D cũng như chụp và phân tích hình ảnh từ video.

Keras: Keras là thư viện mạng nơ-ron chuyên sâu của Python với khả năng hỗ trợ tuyệt vời cho việc xử lý dữ liệu, trực quan hóa và hơn thế nữa. Keras hỗ trợ nhiều mạng nơ-ron. Thư viện này có cấu trúc mô-đun mang lại sự linh hoạt cho việc lập trình các ứng dụng sáng tạo.

2. Dữ liệu thực nghiệm

Bộ dữ liệu được sử dụng chứa 5 loại gạo: Arborio, Basmati, Ipsala, Jasmine và Karacadag. Có 75000 hình ảnh hạt gạo với 15000 hình ảnh cho mỗi loại gạo. Bộ dữ liệu có 12 đặc điểm hình thái, 4 hình dạng và 90 đặc điểm màu sắc.

Cụ thể thông tin như sau:

Tên bộ dữ liệu: Rice Image Dataset.

Nguồn: <https://www.muratkoklu.com/datasets/>

Cấu trúc thư mục dữ liệu:

A screenshot of a computer

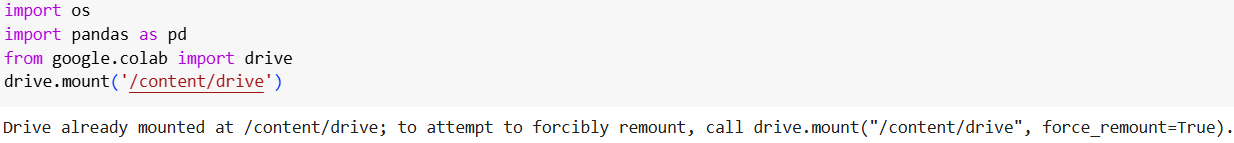
Description automatically generated

1. Cấu trúc thư mục dữ liệu

3. Áp dụng mô hình CNN phân loại gạo

3.1. Thu thập dữ liệu

Kết nối colab với drive.



1. Kết nối colab với drive

Giải nén file dữ liệu

A screenshot of a computer code

Description automatically generated

1. Giải nén file dữ liệu

Tạo data frame chứa đường dẫn đến ảnh và các nhãn của các ảnh.

A screenshot of a computer program

Description automatically generated

1. Data Frame chứa các đường dẫn ảnh và nhãn tương ứng

A screenshot of a computer

Description automatically generated

1. Kết quả Data Frame chứa đường dẫn ảnh và các nhãn tương ứng

Hiển thị dữ liệu 4 hình ảnh của mỗi loại gạo trong tệp dữ liệu lên màn hình.

A screenshot of a computer program

Description automatically generated

1. Code hiển thị 4 ảnh mỗi loại gạo lên màn hình

A collage of different types of white rice

Description automatically generated

1. Kết quả hiển thị 4 ảnh mỗi loại gạo

3.2 Tiền xử lý dữ liệu

A screenshot of a computer program

Description automatically generated

1. Chia dữ liệu thành các tập và mã hóa các nhãn

Ta sử dụng train\_test\_split của sklearn để chia dữ liệu thành các tập. Với test\_size = 0.4, ta chia dữ liệu thành 2 tập chứa 60% dữ liệu train (huấn luyện) và 40% dữ liệu gồm test (kiểm tra) và validation (giám sát). random\_state = 42, ta sẽ lấy dữ liệu theo một cách xác định để đảm bảo dữ liệu được lấy giống nhau ở mỗi lần chạy chương trình, đồng thời tránh các tập chứa dữ liệu trùng nhau.

Với test\_size = 0.5, ta tiếp tục 40% dữ liệu gồm test và validation trước đó thành 20% dữ liệu test và 20% dữ liệu validation. random\_state = 42 như trên. Sau khi chia dữ liệu xong, ta chuyển các dữ liệu sau khi chia thành data frame chứa đường dẫn ảnh và các nhãn tương ứng. Để thuận tiện cho việc tính toán của máy, ta sẽ mã hóa các nhãn ở các tệp bằng Label Encoder.

A screenshot of a computer

Description automatically generated

1. Tạo Image Data Generator để tăng độ linh hoạt của dữ liệu

Để mô hình có độ chính xác cao, ta cần biến đổi ảnh đa dạng để máy có thể nhận diện được loại gạo trong nhiều tính huống hơn. Chuẩn hóa dữ liệu mỗi pixel về khoảng [0, 1]. Ảnh được xoay ảnh ngẫu nhiên từ - 45 đến 45 độ. Dịch chuyển ảnh gốc một khoảng ngẫu nhiên tối đa là 20% chiều rộng của ảnh. Dịch chuyển ảnh gốc một khoảng ngẫu nhiên tối đa là 20% chiều cao của ảnh. Ảnh có hiệu ứng cắt xén ngẫu nhiên tối đa là 20 độ. Ảnh được phóng to hoặc thu nhỏ ngẫu nhiên tối đa là 20%. Lật ảnh gốc theo chiều ngang. Xử lý các pixel do các phép biến đổi trên bằng các giá trị gần kề.

A screenshot of a computer program

Description automatically generated

1. Áp dụng Image Data Generator vào các tệp dữ liệu

Xác định data frame tương ứng của các tập, xác định đường dẫn và nhãn. Kích thước ảnh được xác định sẽ sử dụng là 125px x 125px, batch size là 32. class\_model là sparse để có thể dùng được hàm mất mát là sparse cross entropy. Ta sẽ xáo trộn dữ liệu với tệp train để tăng độ linh hoạt cho dữ liệu khi máy học, tệp test và validation thì giữ nguyên vì tệp này chỉ dùng để kiểm tra nên việc xáo trộn là không cần thiết. Kết quả ta được 45000 ảnh đã biến đổi trong tệp train, 15000 ảnh đã biến đổi trong tệp validation và 15000 ảnh đã biến đổi trong tệp test, các tệp đều chứa đủ 5 lớp.

3.3 Xây dựng mô hình CNN

A screen shot of a computer code

Description automatically generated

1. Xây dựng mô hình CNN

Mô hình CNN với ảnh đầu vào có kích thước 125px x 125px, ảnh màu RGB. Lớp tích chập có các filter kích thước 3x3, padding same được sử dụng, activation là relu. Lớp max pooling sẽ được dùng sau lớp tích chập có kích thước 2x2. Lớp tích chập thứ nhất sử dụng 16 filter, lớp tích chập thứ hai là 32 filter và lớp tích chập thứ ba là 64 filter. Sau khi max pooling lần thứ 3 là lớp Flatten để chuyển thành các vector dùng cho lớp Fully-connected với 128 nơ ron. Cuối cùng là lớp softmax với 5 nơ ron tương ứng với xác suất tương ứng mỗi loại gạo.

A screenshot of a computer

Description automatically generated

1. Thông số model CNN được xây dựng

Tên lớp: Tên được đặt cho mỗi lớp để dễ phân biệt.

Output Shape: Hình dạng của đầu ra từ lớp đó.

Param #: Số lượng tham số cần phải học trong lớp.

Total Params: Tổng số tham số của toàn bộ mô hình.

Trainable params: Số lượng tham số có thể điều chỉnh trong quá trình huấn luyện.

Non-trainable params: Số lượng tham số không thể điều chỉnh (thường là các tham số của lớp nhúng).

Qua lớp tích chập thứ nhất từ ảnh 125px x 125px với 3 màu RGB, ta được (None, 125, 125, 16) nghĩa là batch size: none hay là kích thước batch size linh hoạt, do padding là same nên sau lớp tích chập ảnh vẫn là 125px x 125px, số feature map là 16. Param của lớp tích chập thứ nhất ((hình dạng chiều rộng của bộ lọc \* hình dạng chiều cao của bộ lọc \* số bộ lọc trong lớp trước đó) + 1) \* số bộ lọc) = ((3\*3\*3) + 1)\*16 = 448.

Sau khi max pooling thì kích thước ảnh giảm đi một nửa thành (None, 62, 62, 16) kích thước ảnh chỉ còn 62px x 62px. Lớp pooling không có việc học nên param là 0. Các lớp tích chập và pooling thứ 2 và 3 tương tự.

A collage of images of different types of filters

Description automatically generated

1. Hình ảnh hiển thị sau lớp conv2d đầu tiên

A collage of images of various shapes

Description automatically generated

1. Hình ảnh hiển thị sau lớp maxpooling đầu tiên

A collage of images of a cucumber

Description automatically generated

1. Hình ảnh hiển thị sau lớp conv2d thứ 2

A collage of images of a camera

Description automatically generated

1. Hình ảnh hiển thị sau lớp maxpooling thứ 2

A black and white image of a number of objects

Description automatically generated with medium confidence

1. Hình ảnh hiển thị sau lớp conv2d thứ 3

A black and white squares with white text

Description automatically generated

1. Hình ảnh hiển thị sau lớp maxpooling thứ 3

Tại flatten ta được số phần tử của vector là 15\*15\*64 = 14400. Param của lớp fully connected là ((lớp hiện tại c\*lớp trước p) + 1\*c) = ((128\*14400) + 1\*128) = 1843328. Param của lớp softmax là ((5\*128) + 1\*5) = 645.

A screenshot of a computer

Description automatically generated

1. Kích thước tensor sau lớp MaxPooling2D thứ ba (trước khi flatten) (1)

A screenshot of a computer code

Description automatically generated

1. Kích thước tensor sau lớp MaxPooling2D thứ ba (trước khi Flatten) (2)

A screenshot of a computer

Description automatically generated

1. Kích thước tensor sau lớp MaxPooling2D thứ ba (trước khi Flatten) (3)

A white background with black and red text

Description automatically generated

1. Vector sau khi Flatten

3.4 Huấn luyện mô hình

A screenshot of a computer code

Description automatically generated

1. Chuẩn bị mô hình cho quá trình huấn luyện

A close up of words

Description automatically generated

1. Huấn luyện mô hình

## 3.2. Kết quả đạt được và đánh giá

3.2.1. Kết quả đạt được

A close-up of a text

Description automatically generated

1. Lưu mô hình lại

A screen shot of a computer

Description automatically generated

1. Dùng mô hình dự đoán

A screenshot of a computer program

Description automatically generated

1. Kết quả dùng mô hình dự đoán

Ta sẽ sử dụng 10 bức ảnh của loại gạo Jasmine cho mô hình để xem mô hình có dự đoán có chính xác 10 bức ảnh này không. Và kết quả nhận được là 10 hình ảnh đều được mô hình phân loại chính các là loại gạo Jasmine.

3.2.2. Đánh giá mô hình

A screenshot of a computer code

Description automatically generated

1. Vẽ đồ thị đánh giá mô hình

A graph with a line

Description automatically generated

1. Đồ thị đánh giá mô hình qua mỗi epoch

Độ chính xác (accuracy) tăng dần theo số lượng epoch: Điều này cho thấy mô hình đang học được các đặc trưng từ dữ liệu huấn luyện và khả năng dự đoán đúng của mô hình đang cải thiện.

Độ chính xác trên tập validation (val\_accuracy) cũng tăng và gần sát với độ chính xác trên tập huấn luyện: Điều này cho thấy mô hình đang tổng quát hóa tốt trên dữ liệu mới, không bị overfitting (học quá mức) trên tập huấn luyện.

A screenshot of a computer

Description automatically generated

1. Độ chính xác của mô hình trên tập test

Mô hình huấn luyện có độ chính xác là 98,74% cho thấy đây là một mô hình CNN tốt dùng để phân loại gạo sẽ có kết quả rất sát với thực tế.

**3.3. Kết luận chương III**

Chương III trình bày chi tiết quá trình triển khai mô hình CNN vào bài toán phân loại gạo, từ xây dựng quy trình, xử lý dữ liệu, đến huấn luyện và đánh giá thấy tính hiệu quả và tiềm năng ứng dụng thực tế của phương pháp đề xuất

# KẾT LUẬN VÀ KIẾN NGHỊ

Báo cáo này đã tập trung nghiên cứu và ứng dụng mô hình CNN để giải quyết bài toán phân loại gạo, một vấn đề mang ý nghĩa thực tiễn cao trong lĩnh vực công nghiệp và công nghệ.

Kết quả nghiên cứu không chỉ chứng minh khả năng ứng dụng của CNN trong lĩnh vực phân loại gạo mà còn mở ra tiềm năng phát triển cho các bài toán tương tự trong nông nghiệp và công nghiệp chế biến.

Tuy nhiên, trong quá trình thực hiện, một số hạn chế đã được ghi nhận, như yêu cầu về dữ liệu lớn và thời gian huấn luyện dài. Đây sẽ là những thách thức cần được khắc phục trong các nghiên cứu tiếp theo. Trong tương lai, việc cải tiến mô hình, tối ưu hoá thời gian xử lý, và mở rộng phạm vi ứng dụng sang các lĩnh vực khác sẽ là những hướng đi tiềm năng để phát triển nghiên cứu này.

Báo cáo đã đạt được các mục tiêu đặt ra, đồng thời khẳng định sự cần thiết và hiệu quả của việc ứng dụng công nghệ hiện đại vào giải quyết các vấn đề thực tiễn trong đời sống.

Trong quá trình thực hiện đề tài “Ứng dụng CNN để xây dựng hệ thống phân loại gạo”, chúng em nhận thấy rằng việc áp dụng trí tuệ nhân tạo, đặc biệt là mạng nơ-ron tích chập, có tiềm năng rất lớn trong việc nâng cao hiệu quả và tự động hóa các quy trình sản xuất nông nghiệp. Tuy nhiên, hiện nay, nguồn dữ liệu phục vụ nghiên cứu trong lĩnh vực này tại Việt Nam còn hạn chế về số lượng và chất lượng. Vì vậy, chúng em kiến nghị các tổ chức và cơ quan nghiên cứu nên đầu tư xây dựng các bộ dữ liệu lớn, đa dạng và phù hợp với điều kiện thực tế tại Việt Nam để hỗ trợ các nghiên cứu trong tương lai.

Bên cạnh đó, việc triển khai và tối ưu hóa các mô hình học sâu yêu cầu tài nguyên tính toán mạnh mẽ. Chúng em kiến nghị nhà trường và các tổ chức liên quan nên có chính sách hỗ trợ sinh viên tiếp cận với các công cụ hiện đại như GPU hoặc các nền tảng điện toán đám mây, giúp nâng cao hiệu quả nghiên cứu và học tập.

Cuối cùng, chúng em hy vọng có thể thúc đẩy sự hợp tác giữa các trường đại học và các doanh nghiệp trong lĩnh vực nông nghiệp. Việc ứng dụng các mô hình trí tuệ nhân tạo vào thực tiễn không chỉ giúp cải thiện năng suất mà còn góp phần hiện đại hóa ngành nông nghiệp Việt Nam, mở ra nhiều hướng phát triển trong tương lai.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | www.scielo.br. 2024. SciELO - Brazil - Rice Classification and Quality Detection Success with Artificial Intelligence Technologies Rice Classification and Quality Detection Success with Artificial Intelligence Technologies. [ONLINE] Available at: https://www.scielo.br/j/babt/a/p5tvNxMPp9xjTVkkLkm7HBJ/. [Accessed 23 November 2024]. |
| [2] | Đ. Đ. T. N. T. T. T. L. Q. Trần Hồng Việt, "A FACE RECOGNITION SYSTEM USING MULTI-TASK CASCADED CONVOLUTIONAL NETWORKS AND FACENET MODEL," June 2021. [Online]. Available: https://khcn.haui.edu.vn/media/30/uffile-upload-no-title30656.pdf. |
| [3] | N. Q. Hoan, "Đề tài Ứng dụng mạng Neural tích chập trong nhận dạng hoa quả," 2021. [Online]. Available: https://fr.slideshare.net/slideshow/ti-ng-dng-mng-neural-tch-chp-trong-nhn-dng-hoa-qu/252784247. |
| [4] | D. Nelson, "Stackabuse," [Online]. Available: https://stackabuse.com/image-recognition-in-python-with-tensorflow-and-keras/. |
| [5] | c.-g. MarkDaoust, "https://www.tensorflow.org/tutorials," [Online]. Available: https://www.tensorflow.org/tutorials/images/classification?hl=vi. |
| [6] | "Datasets," [Online]. Available: https://www.muratkoklu.com/datasets/. |
| [7] | D. Đ. Trình, "Convolutional Neural Networks trong machine learning," 12 December 2021. [Online]. Available: https://websitehcm.com/convolutional-neural-networks-trong-machine-learning/. |
| [8] | "aws.amazon.com. 2024. Python là gì? - Giải thích về ngôn ngữ Python - AWS," [Online]. Available: https://aws.amazon.com/vi/what-is/python/. [Accessed 13 November 2024]. |
| [9] | "Introduction to Convolution Neural Network," [Online]. Available: https://www.geeksforgeeks.org/introduction-convolution-neural-network/. [Accessed October 2024]. |
| [10] | Sunil, "Learn How to Use Support Vector Machines (SVM) for Data Science," [Online]. Available: https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/. [Accessed 26 September 2024]. |
| [11] | "What is the k-nearest neighbors (KNN) algorithm?," [Online]. Available: https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN,used%20in%20machine%20learning%20today.. |
| [12] | T. Nguyễn, "Giới thiệu về thuật toán Random Forest," [Online]. Available: https://machinelearningcoban.com/tabml\_book/ch\_model/random\_forest.html. |
| [13] | "Học chuyển giao là gì?," [Online]. Available: https://aws.amazon.com/vi/what-is/transfer-learning/. |
| [14] | "Deep Convolutional Neural Networks," [Online]. Available: https://www.run.ai/guides/deep-learning-for-computer-vision/deep-convolutional-neural-networks. |
| [15] | Dzung, "Các ứng dụng của SIFT/SURF," 12 July 2015. [Online]. Available: https://viblo.asia/p/cac-ung-dung-cua-siftsurf-zNPVMaA8MQOk. |
| [16] | aws.amazon.com. 2024. Python là gì? - Giải thích về ngôn ngữ Python - AWS. [ONLINE] Available at: https://aws.amazon.com/vi/what-is/python/. [Accessed 23 November 2024]. |