

Phát hiện bệnh phấn trắng và nấm mốc trên cây trồng

Trần Tiến Nam và Nguyễn Đức Minh

MSV: 22022594

MSV: 22022533

Viện trí tuệ nhân tạo - ĐH Công Nghệ - ĐHQGHN

Tóm tắt nội dung

Tóm tắt:

Bài nghiên cứu của chúng tôi tập trung về việc phát hiện bệnh trên lá của cây trồng sử dụng mô hình CNN. Mô hình của chúng tôi sẽ phân biệt hai loại bệnh phổ biến là nấm mốc và phấn trắng, đồng thời mô hình cũng có khả năng nhận biết lá xanh tốt, khỏe mạnh.

Từ khóa: CNN, phân loại, ResNet-18, nấm mốc, phấn trắng.

1 Đặt vấn đề

Trong những năm gần đây, các mô hình trí tuệ nhân tạo đã nổi lên như một công cụ mạnh mẽ cho các bài toán xử lý ảnh, bao gồm cả việc nhận dạng các đối tượng trong ảnh. Mô hình trí tuệ nhân tạo có khả năng học hỏi các đặc trưng phức tạp từ dữ liệu ảnh và phân loại các đối tượng với độ chính xác cao.

Theo thống kê của Sở nông nghiệp và phát triển nông thôn Tỉnh Bắc Giang, thiệt hại do sâu bệnh gây ra cho cây trồng, rau màu trong năm 2023 là khá nghiêm trọng.

Tổng diện tích nhiễm sâu bệnh trên các loại cây trồng khoảng 63.000 ha, nhiễm nặng 4.000 ha, diện tích phòng trừ 112.000 ha.

Các loại sâu bệnh hại phổ biến gồm sâu ăn tạp, sâu xanh đục trái, bọ trĩ, bệnh thối đọt, bệnh vàng lùn, bệnh lùn sọc đen, dịch rầy nâu, sâu keo mùa thu, bệnh khảm lá sắn...

Các yếu tố gây ra sự bùng phát của sâu bệnh hại bao gồm biến đổi khí hậu, điều kiện thời tiết nóng ẩm, trồng liên tục các loại cây thuộc cây ký chủ của sâu bệnh, phun thuốc trừ sâu hóa học quá nhiều, giảm số lượng thiên địch...

Bài nghiên cứu của chúng tôi cũng ứng dụng công nghệ AI trong nông nghiệp, cụ thể là huấn luyện mô hình học máy phát hiện những bệnh phổ biến trên lá của cây trồng để nông dân có thể phản ứng nhanh hơn trước các bệnh này từ đó tăng năng suất cây trồng. Đồng thời mô hình của chúng tôi cũng có khả năng nhận dạng được lá xanh tốt từ đó tăng khả năng thực tiễn.

Những bệnh chúng tôi tập trung chủ yếu là: nấm mốc và phấn trắng. Chúng tôi chọn hai loại bệnh này vì đây đều là bệnh thường gặp trên cây trồng và có ảnh hưởng rất lớn đến sản lượng của cây trồng.

2 Nghiên cứu liên quan

2.1 Thông tin về hai bệnh phấn trắng và nấm mốc

a) Nấm mốc:

Nguyên nhân: Bệnh thường do nấm gây ra, như nấm mốc đen (*Alternaria* spp.), nấm mốc trắng (*Phomopsis* spp.), hay các loại nấm khác. Những nấm này thường phát triển và lan truyền trong điều kiện ẩm ướt, khí hậu ẩm áp, và thường được tìm thấy trong đất hoặc các vật liệu hữu cơ phân hủy.

Triệu chứng: Các lá của cây trở nên mờ và xuất hiện các đốm màu đen, nâu hoặc trắng. Đốm có thể bị mục nát, làm cho lá trở nên héo, chết và rụng. Trên các trái cây, các đốm có thể lan rộng và gây ra sự hỏng hóc hoặc suy yếu của trái.

Quy mô: theo số liệu từ Bộ Nông nghiệp và Phát triển Nông thôn Việt Nam năm 2022, diện tích bị ảnh hưởng bởi nấm mốc trên cây trồng được ước tính lên đến khoảng 20.000 hecta. Tuy nhiên, con số này có thể thay đổi theo thời gian và vùng miền. Ví dụ, vào năm 2020, diện tích bị ảnh hưởng là 25.000 hecta.

Thiệt hại năng suất: nấm mốc có thể làm giảm năng suất cây trồng từ 10% đến 30%. Theo dữ liệu từ Bộ Nông nghiệp và Phát triển Nông thôn Việt Nam năm 2022, do tác động của nấm mốc, sản lượng cây trồng giảm 15%. Điều này tương đương với mất mát 300.000 tấn sản phẩm, gây thiệt hại kinh tế 300 tỷ đồng.

Mức độ ảnh hưởng: Nấm mốc ảnh hưởng đến tất cả các bộ phận của cây trồng, bao gồm lá, thân, cành, quả. Trường hợp nghiêm trọng có thể khiến cho cây trồng chết.

Một số loại cây trồng thường bị nấm mốc hại: các loại cây trồng phổ biến dễ bị nhiễm nấm mốc bao gồm cà chua, khoai tây, ớt chuông, bầu bí, và cây ăn quả.

b) Phấn trắng:

Nguyên nhân: Bệnh phấn trắng thường do nấm phấn trắng (*powdery mildew*) gây ra, chủ yếu thuộc các loài thuộc họ Erysiphaceae. Nấm này thường phát triển trong điều kiện ẩm ướt và khí hậu mát mẻ.

Triệu chứng: Mặt trên của lá bị phủ một lớp bột màu trắng hoặc màu xám nhạt, giống như bột mỳ. Dưới điều kiện ẩm ướt, nấm phấn trắng có thể phát triển nhanh chóng và lan ra khắp cây trồng.

Quy mô: theo số liệu từ Bộ Nông nghiệp và Phát triển Nông thôn Việt Nam vào năm 2022, diện tích cây

trồng bị bệnh phấn trắng ước tính lên đến 30.000 ha. Tuy nhiên, diện tích này có thể biến đổi tùy thuộc vào thời gian và vùng miền. Ví dụ, vào năm 2020, diện tích cây trồng bị bệnh phấn trắng là 35.000 ha.

Thiệt hại năng suất: bệnh phấn trắng có khả năng làm giảm năng suất của cây trồng từ 5% đến 20%. Theo thống kê của Bộ Nông nghiệp và Phát triển Nông thôn Việt Nam vào năm 2022, do tác động của bệnh phấn trắng, sản lượng cây trồng giảm 10%. Điều này tương đương với mất mát 200.000 tấn sản phẩm, gây thiệt hại kinh tế 200 tỷ đồng.

Mức độ ảnh hưởng: Bệnh phấn trắng chủ yếu tác động đến lá của cây trồng. Trong trường hợp nặng, lá cây có thể chuyển sang màu vàng và rụng, ảnh hưởng đến quá trình quang hợp và phát triển của cây.

Một số loại cây trồng thường bị bệnh phấn trắng: các loại cây trồng thường bị bệnh phấn trắng bao gồm cà chua, dưa chuột, bí ngô, nho và hoa hồng.

2.2 Các nghiên cứu liên quan về ứng dụng mô hình phân loại

- Nghiên cứu phương pháp phát hiện một số sâu bệnh trên lúa sử dụng đặc trưng SIFT. (Tác giả: Nguyễn Ngọc Tú, Bùi Thị Thanh Phương, Lê Hoàng Nam, Ngô Nam Thanh)[6]. Nghiên cứu này đề xuất một hệ thống tự động để phát hiện sâu bệnh trên lúa bằng cách sử dụng kỹ thuật xử lý ảnh và học máy. Hệ thống bao gồm hai giai đoạn chính: trích xuất đặc trưng và phân loại. Trong giai đoạn trích xuất đặc trưng, thuật toán SIFT được sử dụng để trích xuất các đặc trưng mô tả từ hình ảnh lá lúa. Sau đó, các đặc trưng này được sử dụng để huấn luyện mô hình phân loại SVM. Mô hình SVM được sử dụng để phân loại hình ảnh lá lúa thành hai lớp: lá lúa khỏe mạnh và lá lúa bị bệnh. Hệ thống được đánh giá trên một tập dữ liệu gồm 200 hình ảnh lá lúa, trong đó 100 hình ảnh lá lúa khỏe mạnh và 100 hình ảnh lá lúa bị bệnh. Kết quả cho thấy hệ thống có thể đạt được độ chính xác 95%.
- Tự động nhận dạng một số loại sâu bệnh trên lá bưởi sử dụng công nghệ ảnh (Tác giả: Nguyễn Minh Triết, Trương Quốc Bảo và Trương Quốc Định)[9]. Nghiên cứu này đã đề xuất một phương pháp tự động nhận diện các loại sâu bệnh trên lá bưởi bằng cách sử dụng công nghệ ảnh kết hợp với máy học. Phương pháp này bao gồm hai bước chính. Đầu tiên là phân đoạn ảnh sử dụng thuật toán Otsu, sau đó là việc tìm kiếm biên vùng ứng viên bằng cách kết hợp với phép xử lý hình thái. Tiếp theo, đặc trưng về màu sắc và hình dạng của các vùng ứng viên được rút trích để huấn luyện hệ thống nhận dạng bằng máy học vectơ hỗ trợ. Thí nghiệm được thực hiện trên 500 ảnh, và kết quả cho thấy hệ thống đạt độ chính xác 99,2% với thời gian xử lý trung bình khoảng 200 ms.
- Nghiên cứu "Phân loại và đánh giá mức độ bệnh trên lá lúa bằng bộ phân loại đa lớp (Tác giả: GL Anoop và C Nandini.)[1]. Nghiên cứu đề xuất một hệ thống để phát hiện và phân loại ba loại bệnh trên lá lúa:

bệnh đạo ôn, bệnh lem lép hạt và bệnh bạc lá. Hệ thống hoạt động theo hai giai đoạn:

- Phát hiện, trích xuất và phân vùng các vùng bệnh: sử dụng các kỹ thuật tiền xử lý ảnh như Wiener, Adaptive histogram để loại bỏ nhiễu và cải thiện chất lượng ảnh. Áp dụng thuật toán ngưỡng hóa Otsu để phân vùng các vùng bị bệnh trên lá.
 - Trích xuất đặc trưng, phân loại và đánh giá mức độ bệnh: sử dụng kỹ thuật phát hiện cạnh để trích xuất các đặc trưng hình ảnh liên quan đến bệnh. Sử dụng phân lớp SVM để phân loại ba loại bệnh. Sử dụng logic mờ để đánh giá mức độ nghiêm trọng của bệnh.
- Nhận dạng một số bệnh trên bông xoài sử dụng kỹ thuật học sâu chuyển tiếp (Tác giả: Phan Tấn Phước, Ngô Quang Hiếu và Trương Quốc Bảo)[7]. Bài báo này nghiên cứu về việc nhận dạng một số bệnh trên bông xoài sử dụng kỹ thuật học sâu chuyển tiếp. Các tác giả đã thu thập ảnh của bông xoài bị nhiễm bệnh và sử dụng một mô hình học sâu được huấn luyện trên tập dữ liệu ImageNet để phân loại các bệnh. Kết quả cho thấy mô hình học sâu chuyển tiếp có độ chính xác cao hơn so với các phương pháp truyền thống như SVM, KNN, và Random Forest. Bài báo cũng đề xuất một ứng dụng di động cho phép người dùng chụp ảnh bông xoài và nhận biết bệnh một cách nhanh chóng và dễ dàng.
 - Mô hình fast R-CNN cải tiến cho giải pháp nhận dạng, phát hiện trái dưa thời kỳ chín. (Tác giả: Trịnh Trung Hải, Nguyễn Hà Huy Cường, Ninh Khánh Duy)[5]. Trong bài báo này, nhóm tác giả trình bày giải pháp nhận dạng phát hiện trái dưa vào thời kỳ thu hoạch sử dụng mô hình Fast R-CNN cải tiến, thực nghiệm được thực hiện tại các vùng, vườn trồng dưa ở Việt Nam với sản lượng thu hoạch khá lớn. Từ đó xây dựng hệ thống ứng dụng dự đoán trái dưa chín chạy trên nền tảng di động nhằm hỗ trợ người nông dân đạt hiệu quả hiệu quả kinh tế cao đối với cây dưa và các cây nông nghiệp chủ lực của Việt Nam.
 - Rice Plant Disease Detection and Classification Techniques : A Survey [8]. Bài báo này tập trung vào việc nghiên cứu về các kỹ thuật nhận diện bệnh tật của cây lúa bằng cách sử dụng hình ảnh của chúng trong lĩnh vực nông nghiệp và công nghệ máy tính hiện đại. Bài báo cung cấp một cuộc khảo sát về các kỹ thuật khác nhau và thảo luận ngắn gọn về các khía cạnh quan trọng của các bộ phân loại và kỹ thuật khác nhau được sử dụng để phát hiện các bệnh tật của cây lúa. Các tài liệu trong thập kỷ qua được nghiên cứu kỹ lưỡng, bao gồm cả công việc về các bệnh tật của cây lúa khác nhau, và một cuộc khảo sát được trình bày dựa trên các khía cạnh cơ bản. Cuộc khảo sát tập trung vào việc phân biệt các phương pháp khác nhau dựa trên bộ phân loại được sử dụng. Cuộc khảo sát cung cấp cái nhìn về các kỹ thuật khác nhau được sử dụng để nhận dạng bệnh tật của cây lúa. Ngoài ra, một mô hình và prototype phần

cứng sử dụng Mạng Nơ-ron Tích chập (CNN) được đề xuất để phát hiện bệnh tật của cây lúa. Nó cũng xác định loại bệnh của cây lúa thành các loại bệnh như nổ lúa, nấm lúa, nấm đốm nâu, nấm lá, tungro và nấm đốm lá.

- g) Identification of rice diseases using artificial neural network [4]. Mục tiêu của nghiên cứu này là phát triển một mạng nơ-ron nhân tạo (ANN) để nhận dạng bệnh tật của cây lúa trong phòng thí nghiệm của Trường Đại học Nông nghiệp Bangladesh, Mymensingh, trong khoảng thời gian từ tháng 7 năm 2006 đến tháng 3 năm 2007. Lá lúa ở các giai đoạn khác nhau từ các cánh đồng khác nhau đã được thu thập, quét và lưu vào máy tính dưới dạng cường độ màu đỏ, xanh lá cây, xanh lam (RGB) và được chuyển đổi sang định dạng Joint Photographic Experts Group (JPEG). Bốn trong những bệnh phổ biến nhất trên lá lúa, như Nấm đốm nâu, Nổ lúa, Nấm lá và Khu vực chết, đã được phát hiện trên các cánh đồng. Một số 125 hình ảnh của lá lúa được sử dụng để học ANN và 250 hình ảnh khác được sử dụng để nhận dạng các bệnh tật. Mỗi bệnh từ hình ảnh, đã được cắt thành khu vực 16x48 pixel. Trong số tất cả các lá, Nấm đốm nâu, Nổ lúa, Nấm lá và Khu vực chết (khu vực lá bị hỏng không được nhận dạng là bệnh cụ thể) đã được học và kiểm tra bằng cách sử dụng (140, 493), (40, 58), (31,47) và (53,119) bệnh tương ứng. Tỷ lệ thành công tổng thể của việc nhận dạng bệnh là 88,65%, 79,71%, 81,17% và 76,21% cho các bệnh Nấm đốm nâu, Nổ lúa, Nấm lá và Khu vực chết lần lượt. Đặc điểm riêng biệt của bệnh đã đưa ra tỷ lệ thành công cao nhất trong việc phát hiện bệnh của lúa bằng ANN.

- h) Rice Crop Disease Identification and Classifier [2]. Các mục tiêu chính của nghiên cứu là:

- Xây dựng cơ sở dữ liệu để lưu trữ thông tin về bệnh tật của lúa.
- Tìm ra bệnh tật bị ảnh hưởng dựa trên hình ảnh của cây bị nhiễm bệnh sử dụng Deep CNN SVM classifier.
- Xây dựng một cơ sở dữ liệu cho các hội chứng bệnh tật của lúa và các phương pháp điều trị có thể.

- i) Identification and Classification of Rice Plant Diseases using Machine Learning [3]. Bài đánh giá này chủ yếu tập trung vào bệnh tật của cây lúa. Việc phát hiện dựa trên hình ảnh từ cây lúa bị bệnh sử dụng các thuật toán học máy và xử lý hình ảnh khác nhau. Hơn nữa, các khái niệm quan trọng về học máy và xử lý hình ảnh trong việc nhận diện và phân loại cây cỏ đã được đề cập đến. Probabilistic Neural Network (PNN), Evolutionary Techniques (GA) và Thuật toán k-Nearest Neighbor (K-Nearest Neighbor) là ba thuật toán phân loại. Thuật toán k-Nearest Neighbor (KNN) và Support Vector Machine (SVM) là hai thuật toán phân loại khác. Sự tin cậy của một đầu ra phụ thuộc vào dữ liệu đầu vào khi sử dụng trong một số ứng dụng nghiên cứu nông nghiệp. Do

đó, việc lựa chọn một phương pháp phân loại là một nhiệm vụ quan trọng. Nông nghiệp, nghiên cứu sinh học và cetera. Có các ngành công nghiệp khác sử dụng phân loại bệnh lá không? Nghiên cứu toàn diện về các bệnh của cây lúa, kích thước tập dữ liệu hình ảnh, các phương pháp xử lý và phân đoạn, hoặc các bộ phân loại là những biến số quan trọng cần xem xét.

3 Phương pháp

3.1 Thu thập dữ liệu

Đối với việc thu thập dữ liệu, chúng tôi đã lấy ảnh từ nhiều nguồn khác nhau, với ưu tiên dành cho các nguồn có chuyên môn cao trong lĩnh vực nông nghiệp như các bài báo và trang web chuyên về cây trồng. Mục đích của chúng tôi là để dữ liệu thu thập được phản ánh một cách chính xác nhất về các trường hợp thực tế. Các ảnh được thu thập trong môi trường có điều kiện ánh sáng tốt, và giai đoạn phát triển bệnh của lá là ngẫu nhiên để đảm bảo rằng mô hình học có thể nhận biết các biến thể của bệnh một cách khách quan.

3.2 Tiền xử lý dữ liệu

Đối với tập dữ liệu train, chúng tôi đồng nhất kích cỡ ảnh về 244x244, chuẩn hóa dữ liệu bằng cách trừ đi mean và chia cho standard deviation, đồng thời chúng tôi cũng lật ảnh theo chiều ngang và xoay ảnh đi một số góc để mô hình có thể phân biệt chính xác ở nhiều góc quay khác nhau.

Đối với tập dữ liệu test, chúng tôi sẽ chỉ đồng nhất kích cỡ ảnh và chuẩn hóa dữ liệu để đảm bảo độ chính xác của mô hình được đo lường trên các điều kiện thực tế.

Công thức chuẩn hóa theo z-score:

$$x_{\text{normalized}} = \frac{x - \mu}{\sigma},$$

trong đó:

x là giá trị ban đầu của mẫu dữ liệu.

μ là giá trị trung bình của tất cả các mẫu dữ liệu.

σ là độ lệch chuẩn của tất cả các mẫu dữ liệu.

Sau khi xử lý để dữ liệu đầu vào hợp lệ với mô hình, chúng tôi dữ liệu thành các batch, mỗi batch sẽ chứa 32 ảnh. Việc chia dữ liệu thành từng batch này sẽ giúp mô hình có thể tối ưu trọng số nhiều lần hơn. Trọng số sẽ được cập nhật sau mỗi batch chứ không phải sau mỗi epoch, giúp cải thiện độ chính xác của mô hình.

3.3 Phân tích dữ liệu

3.3.1 Phân tích ảnh dựa trên heatmap

Heatmap được sử dụng để biểu diễn cường độ của các điểm ảnh trên lá cây, giúp dễ dàng nhận diện các vùng bị bệnh. Các bước thực hiện bao gồm:

- Chuyển đổi ảnh gốc sang không gian màu phù hợp.
- Tính toán heatmap dựa trên cường độ màu sắc và các đặc trưng khác của ảnh.

Hiệu quả của phân tích heatmap:

Heatmap cung cấp một cái nhìn tổng quan về cường độ của các vùng bị bệnh, giúp dễ dàng nhận diện các vùng này. Tuy nhiên, heatmap có thể bị ảnh hưởng bởi các yếu tố như ánh sáng và màu sắc của lá cây, do đó cần có các biện pháp tiền xử lý ảnh phù hợp để đảm bảo độ chính xác.

3.3.2 Phân tích ảnh dựa trên edges

Phân tích edges được thực hiện để xác định các biên của vùng bị bệnh. Các bước bao gồm:

- Chuyển đổi ảnh gốc sang ảnh xám.
- Sử dụng thuật toán Canny để phát hiện biên.
- Sử dụng các phép toán hình thái để làm sạch và tách biệt các vùng biên.

Hiệu quả của phân tích edges:

Phân tích edges giúp xác định các biên của vùng bị bệnh một cách chính xác. Điều này rất hữu ích trong việc phân biệt các vùng bị bệnh với các vùng lá khỏe mạnh. Tuy nhiên, phân tích edges có thể gặp khó khăn khi các vùng bị bệnh có độ tương phản thấp hoặc khi ảnh bị nhiễu mạnh.

3.4 Mô hình huấn luyện

3.4.1 Mô hình GradientBoostingClassifier(GBC)

GradientBoostingClassifier là một thuật toán ensemble học máy, sử dụng phương pháp boosting để kết hợp nhiều mô hình cây quyết định yếu (weak learners) thành một mô hình mạnh (strong learner). Mục tiêu chính là giảm thiểu lỗi dự đoán bằng cách học từ các sai lầm của các mô hình trước đó. Quá trình huấn luyện diễn ra theo các bước lặp lại, mỗi bước nhằm sửa chữa lỗi của mô hình trước đó bằng cách thêm vào một cây quyết định mới.

Trong xử lý ảnh, GradientBoostingClassifier có thể được sử dụng để thực hiện các tác vụ phân loại ảnh, chẳng hạn như phân loại đối tượng, nhận dạng khuôn mặt, và phân loại các đặc trưng trong ảnh y khoa.

3.4.2 Mô hình LGBM Classifier

LGBM Classifier, một biến thể của GradientBoostingClassifier, là một công cụ mạnh mẽ cho các bài toán phân loại nhờ vào khả năng xử lý hiệu quả các tập dữ liệu lớn và phức tạp. Phương pháp này sử dụng kỹ thuật boosting để kết hợp nhiều cây quyết định yếu, nhằm tạo ra một mô hình học mạnh mẽ và chính xác, là một thuật toán tăng cường độ dốc gradient được thiết kế nhằm cải thiện hiệu suất và giảm thời gian huấn luyện so với các phương pháp truyền thống như XGBoost.

3.4.3 Mô hình KNeighborsClassifier

KNeighbors Classifier là một thuật toán học máy thuộc nhóm các phương pháp không tham số, dựa trên kỹ thuật láng giềng gần nhất (k-nearest neighbors, KNN). Thuật toán này hoạt động bằng cách tìm kiếm k điểm dữ liệu gần nhất trong không gian đặc trưng và sử dụng thông tin từ các điểm này để dự đoán nhãn cho điểm dữ liệu mới.

Trong lĩnh vực xử lý phân loại ảnh, KNeighborsClassifier có tác dụng quan trọng trong việc phân loại các đối tượng dựa trên sự tương đồng với các mẫu huấn luyện đã biết. Phương pháp này hoạt động tốt trong các bài toán mà ranh giới phân chia giữa các lớp không tuyến tính và phức tạp. Một trong những ưu điểm của KNeighborsClassifier là không cần giai đoạn huấn luyện phức tạp, mà tập trung vào tính toán khoảng cách và xác định láng giềng gần nhất trong không gian đặc trưng.

3.4.4 Mô hình RandomForestClassifier(RFC)

Random Forest Classifier là một thuật toán học máy thuộc nhóm các phương pháp ensemble, sử dụng kỹ thuật bagging để kết hợp nhiều cây quyết định (decision trees) thành một mô hình mạnh mẽ và chính xác hơn. Mỗi cây quyết định trong rừng (forest) được huấn luyện trên một mẫu ngẫu nhiên khác nhau của tập dữ liệu huấn luyện, và các cây này cùng nhau đưa ra dự đoán bằng cách bỏ phiếu đa số.

Trong lĩnh vực xử lý phân loại ảnh, RandomForestClassifier có tác dụng quan trọng trong việc phân loại các đối tượng trong ảnh một cách hiệu quả và chính xác. Phương pháp này tận dụng lợi thế của nhiều cây quyết định để giảm thiểu hiện tượng overfitting, một vấn đề thường gặp khi sử dụng một cây quyết định duy nhất. RandomForestClassifier cũng có khả năng xử lý các tập dữ liệu lớn và đa dạng, đồng thời tự động đánh giá tầm quan trọng của các đặc trưng đầu vào, giúp cải thiện độ chính xác và khả năng khái quát hóa của mô hình.

3.4.5 Mô hình Support Vector Machine(SVM)

Support Vector Machine (SVM) là một thuật toán học máy mạnh mẽ được sử dụng cho các bài toán phân loại và hồi quy. SVM hoạt động bằng cách tìm kiếm một siêu phẳng (hyperplane) tối ưu trong không gian đặc trưng để phân tách các lớp dữ liệu. Siêu phẳng này được tối ưu hóa sao cho khoảng cách (margin) giữa nó và các điểm dữ liệu gần nhất từ mỗi lớp là lớn nhất. Những điểm dữ liệu gần nhất này được gọi là các vector hỗ trợ (support vectors).

Trong lĩnh vực xử lý phân loại ảnh, SVM có tác dụng quan trọng trong việc phân loại các đối tượng và đặc trưng trong ảnh với độ chính xác cao. SVM đặc biệt hiệu quả khi các lớp dữ liệu có sự phân biệt rõ ràng trong không gian đặc trưng. Hơn nữa, SVM có thể được mở rộng để xử lý các bài toán không tuyến tính bằng cách sử dụng các hàm kernel như RBF (Radial Basis Function) để ánh xạ dữ liệu vào không gian chiều cao hơn, nơi siêu phẳng có thể được tìm thấy.

3.4.6 Mô hình Transfer Learning kết hợp Fine-tuning

Transfer Learning (học chuyển giao) là một kỹ thuật trong học máy và học sâu, trong đó một mô hình được huấn luyện trước trên một tập dữ liệu lớn và đa dạng (thường là trên các nhiệm vụ phổ biến như phân loại ảnh trên ImageNet) được chuyển giao và tinh chỉnh cho một nhiệm vụ cụ thể mới với tập dữ liệu nhỏ hơn. Phương pháp này giúp tận dụng kiến thức đã học được từ nhiệm vụ gốc để cải thiện hiệu suất và tốc độ huấn luyện cho nhiệm vụ mới.

Trong lĩnh vực xử lý phân loại ảnh, Transfer Learning có tác dụng quan trọng trong việc cải thiện hiệu suất mô hình, đặc biệt khi có ít dữ liệu huấn luyện. Bằng cách sử dụng các mô hình đã được huấn luyện trước (pre-trained models) như VGG, ResNet, hay Inception, các đặc trưng phức tạp và quan trọng từ các ảnh đã được học một cách hiệu quả. Sau đó, các lớp cuối cùng của mô hình được thay thế để phù hợp với bài toán của chúng tôi. Tiếp theo chúng tôi sử dụng Fine-tuning để tinh chỉnh các trọng số để phù hợp với nhiệm vụ phân loại tình trạng của lá, giúp tiết kiệm thời gian, tài nguyên huấn luyện và cho ra được kết quả tối ưu với bài toán của chúng tôi.

Mô hình resnet18 có các lớp là: lớp Convolutional, lớp MaxPool, lớp AveragePool, lớp ReLu, lớp BatchNorm, lớp Linear. Năm lớp đầu tiên sẽ là thành phần của các Block trong mô hình còn lớp Linear sẽ lớp cuối cùng để đưa ra output.

Chúng tôi sử dụng cấu trúc mạng cnn của resnet 18 làm mô hình huấn luyện, mô hình sẽ đi qua các block gồm các lớp Convo2D và BatchNorm để trích xuất đặc trưng và chuẩn hóa giá trị đầu vào của mỗi lớp trong mạng nơ ron. Mỗi block cũng sẽ có hàm kích hoạt ReLu để tăng khả năng biểu diễn đặc trưng cho các lớp. Sau đó chúng tôi sử dụng các lớp MaxPool và AveragePool để giảm kích thước các đặc trưng để tăng tốc độ tính toán và giữ lại các đặc trưng quan trọng nhất. Cuối cùng chúng tôi sử dụng lớp Linear để trích xuất ra output là xác suất cho mỗi tình trạng của lá. Mô hình sẽ chọn xác suất cao nhất để đưa ra dự đoán là lá mắc bệnh nấm đốm hoặc mắc bệnh phấn trắng hoặc xanh tốt.

Với bài toán multi classification, chúng tôi sử dụng hàm Multiclass cross-entropy để tính toán mất mát trong mỗi lần dự đoán và hàm SGD (Stochastic Gradient Descent) để tối ưu trọng số dựa trên hàm mất mát.

Cross-entropy là một phương pháp đo lường sự khác biệt giữa hai phân phối xác suất. Trong trường hợp của multiclass classification (phân loại nhiều lớp), cross-entropy thường được sử dụng để đo lường sự tương đồng giữa phân phối xác suất dự đoán và phân phối xác suất thực tế của các lớp.

Đối với bài toán multiclass classification, cross-entropy thường được sử dụng cùng với hàm softmax để đo lường sự không chắc chắn trong dự đoán của mô hình. Cụ thể, cross-entropy loss giảm khi dự đoán của mô hình tiệm cận với phân phối xác suất thực tế của các lớp.

Cross-entropy loss càng nhỏ khi dự đoán của mô hình càng gần với phân phối xác suất thực tế, và cao khi có sự chênh lệch lớn giữa dự đoán và thực tế.

Công thức Multiclass cross-entropy:

$$J(\theta) = \frac{1}{n} \sum_{i=1}^n -\ln g_{y_i}(x_i, \theta),$$

trong đó:

$g_{y_i}(x_i, \theta)$ là softmax.

Công thức softmax:

$$\text{softmax}(\mathbf{z}) \triangleq \frac{1}{\sum_{j=1}^M e^{z_j}} \begin{bmatrix} e^{z_1} \\ e^{z_2} \\ \vdots \\ e^{z_M} \end{bmatrix}$$

SGD (Stochastic Gradient Descent) là một thuật toán tối ưu hóa phổ biến được sử dụng để tìm giá trị tối ưu của một hàm mất mát trong các bài toán machine learning và deep learning. Thuật toán này là một dạng của Gradient Descent, nhưng thay vì cập nhật trọng số sau khi tính toán gradient của toàn bộ dữ liệu huấn luyện (batch gradient descent), SGD chỉ tính gradient của một mẫu dữ liệu đơn lẻ hoặc một minibatch (mini-batch gradient descent) và cập nhật trọng số dựa trên gradient đó.

Công thức SGD:

$$\theta = \theta - \eta \nabla J(\theta),$$

trong đó:

$\nabla J(\theta)$ là gradient của hàm mất mát,

$J(\theta)$ theo các tham số θ ,

η là tỷ lệ học (learning rate), quy định độ lớn của bước cập nhật.

3.4.7 Đánh giá mô hình

Để đánh giá mô hình chúng tôi sử dụng các độ đo là F1-score và Accuracy.

Hàm Accuracy (độ chính xác) là một thước đo đơn giản và phổ biến để đánh giá hiệu suất của mô hình phân loại. Nó cho biết tỷ lệ phần trăm các dự đoán đúng trên tổng số dự đoán.

Công thức Accuracy:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Hàm F1-score là một thước đo dùng để đánh giá hiệu quả của mô hình phân loại, đặc biệt là trong các bài toán có dữ liệu không cân bằng. F1-score là trung bình điều hòa giữa Precision (độ chính xác) và Recall (độ nhạy), giúp cung cấp một cái nhìn tổng quát hơn về hiệu suất của mô hình.

Công thức F1-score:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Công thức precision và recall:

$$\text{precision} = \frac{TP}{TP + FP}, \text{ recall} = \frac{TP}{TP + FN}$$

trong đó:

TP là True Positive,

TN là True Negative,

FP là False Positive,

FN là False Negative.

4 Thực nghiệm

4.1 Phương pháp Thu thập dữ liệu

Chúng tôi tập trung vào việc thu thập dữ liệu từ các nguồn uy tín như trang web chuyên về nông nghiệp và các bài báo có liên quan. Việc này giúp chúng tôi đảm bảo rằng dữ liệu thu thập được phản ánh một cách chính xác nhất tình trạng thực tế về bệnh hại của cây trồng.

4.2 Tiền xử lý dữ liệu

Dữ liệu thu thập được sau đó được chia thành hai tập: train_folder và test_folder. Tập train được sử dụng để huấn luyện mô hình học máy, trong khi tập test được dùng để đánh giá độ chính xác của mô hình. Mỗi tập được phân thành ba folder con tương ứng với ba đặc điểm phân loại chính: lá mắc bệnh phần trắng, lá mắc bệnh nấm mốc và lá xanh tốt.

Hiện tại, tập dữ liệu của chúng tôi bao gồm hơn 400 ảnh cho mỗi đặc điểm, đảm bảo tính đa dạng và đủ lớn để mô hình hóa các biến thể của bệnh hại cây trồng.

Ba đặc điểm của lá được gán nhãn lần lượt là 0 cho bình thường, 1 cho nấm mốc và 2 cho phần trắng.

Tỷ lệ cho tập train và test là 80%, 20%.

4.2.1 Tiền xử lý dữ liệu cho tập Train và tập Test

Trước khi đưa dữ liệu của tập train và tập test vào mô hình học máy, chúng tôi thực hiện các bước tiền xử lý sau:

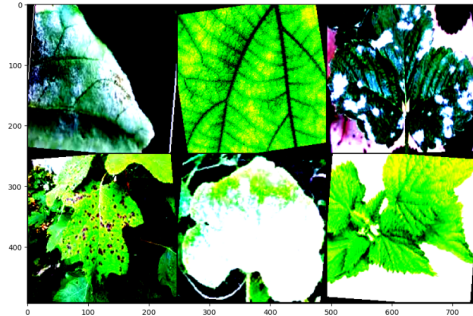
Tiền xử lý dữ liệu cho tập Train:

- Chỉnh kích thước của tất cả ảnh về 244x244 pixels để đảm bảo đồng nhất.
- Chuyển đổi ảnh thành dạng Tensor để mô hình có thể phân tích.
- Chuẩn hóa ảnh bằng cách trừ mean và chia cho standard deviation để giúp mô hình học máy hội tụ nhanh hơn.
- Lật ảnh và quay ảnh theo các góc 10 độ để mô hình không học dự đoán ảnh dựa trên một góc nhìn nhất định, mà có thể phân biệt bệnh ở các góc chụp khác nhau.

Tiền xử lý dữ liệu cho tập Test:

- Chỉnh kích thước của tất cả ảnh về 244x244 pixels để đảm bảo đồng nhất.
- Chuyển đổi ảnh thành dạng Tensor để mô hình có thể phân tích.
- Chuẩn hóa ảnh bằng cách trừ mean và chia cho standard deviation để giữ cho quá trình dự đoán trên tập test được thực hiện một cách nhất quán.
- Không thực hiện việc lật và quay góc ảnh để tập test có thể giống với điều kiện thực tế nhất, đồng thời đảm bảo tính khách quan trong việc đánh giá hiệu suất của mô hình.

Việc tiền xử lý dữ liệu đảm bảo rằng cả tập train và tập test đều được chuẩn bị một cách chu đáo và đồng nhất trước khi đưa vào mô hình học máy, từ đó tăng khả năng dự đoán chính xác của mô hình trên dữ liệu mới.



Hình 1: Ảnh sau khi đã xử lý.

Hình 1 là ảnh sau khi đã được xử lý và cũng là ảnh sẽ được đưa vào để huấn luyện mô hình.

4.2.2 Chia dữ liệu thành các Batch

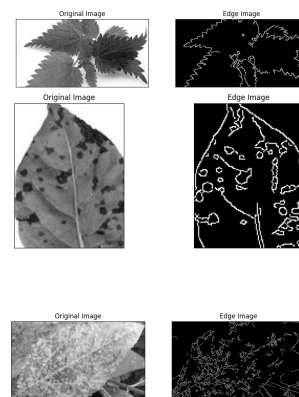
Việc dữ liệu chỉ cập nhật trọng số sau mỗi epoch sẽ làm tốc độ học của mô hình rất lâu và tốn kém tài nguyên. Thay vì đưa hết tất cả dữ liệu vào cùng một lần chúng tôi sẽ chia dữ liệu thành các batch, mỗi batch sẽ gồm 32 ảnh. Trọng số sẽ được cập sau mỗi batch giúp mô hình có tốc độ học nhanh hơn. Việc chia dữ liệu thành các batch như trên sẽ tăng tốc độ học máy, giảm lượng bộ nhớ cần thiết, và giảm bớt thời gian đào tạo mô hình.

4.3 Phân tích dữ liệu

Phân tích edges

Các bước thực hiện:

- Đọc ảnh dưới dạng ảnh xám (cv2.IMREAD_GRAYSCALE).
- Áp dụng Gaussian Blur để giảm nhiễu, giúp cải thiện độ chính xác của việc phát hiện biên.
- Sử dụng thuật toán Canny để phát hiện các biên của vùng bị bệnh.



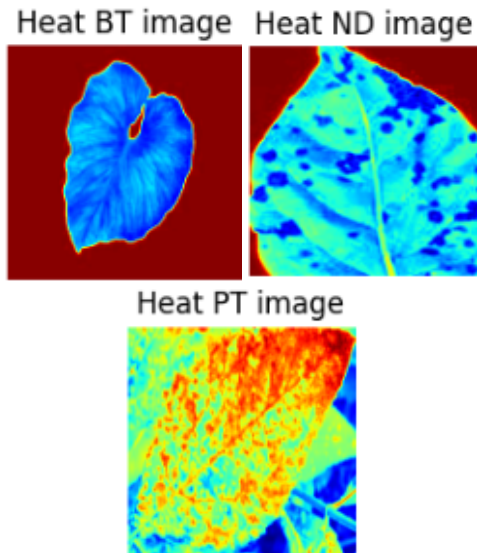
Hình 2: Phân tích edges.

Kết quả: Hình bên trái là ảnh gốc của lá cây, và hình bên phải là ảnh edges. Lá mắc bệnh nấm đốm bên trong sẽ có các đường tròn nhỏ phân bố rải rác. Lá mắc bệnh phần trắng bên trong có các đường thẳng bị gãy nứt và các đường này phân bố rất ngẫu nhiên. Còn lá khỏe mạnh bên trong sẽ có các đường vân đồng đều. Phân tích edges cung cấp một cách trực quan và hiệu quả để nhận diện các vùng bị bệnh, đặc biệt là khi các vùng này có độ tương phản cao so với phần còn lại của lá.

Phân tích heatmap

Các bước thực hiện:

- Sử dụng plt.subplots để tạo lưới 5x8 cho các hình ảnh.
- Lặp qua từng hình ảnh và đọc hình ảnh từ đường dẫn.
- Chuyển đổi hình ảnh thành dạng heatmap (đặt cmap='jet' trong phương thức imshow) và gán nhãn dự đoán trên subplot.



Hình 3: Phân tích heatmap.

Kết quả: Các vùng màu đỏ trên heatmap biểu diễn các vùng có cường độ cao, thường là các vùng bị bệnh. Đối với bệnh nấm đốm, các vùng đỏ hoặc xanh thường phân bố không đều và có dạng đốm tròn nhỏ. Đối với bệnh phần trắng, các vùng đỏ cam có xu hướng xuất hiện dưới dạng các đốm nhỏ hơn và tập trung thành các vùng. Còn đối với lá bình thường, gam màu đa số sẽ là màu lạnh, rất ít gam màu nóng như lá mắc bệnh phần trắng và nấm đốm. Heatmap cung cấp một cách trực quan để xác định các vùng bị bệnh, giúp người dùng dễ dàng nhận diện các vùng bị bệnh mà không cần phân tích kỹ thuật chi tiết.

Kết luận

Từ việc phân tích kết quả sử dụng heatmap và edges trong nhận diện bệnh trên lá cây, chúng tôi nhận thấy rằng hai loại bệnh phần trắng và nấm đốm đều có những đặc trưng rõ ràng, cụ thể. Chúng tôi dự đoán sử dụng các mô hình mạng nơ-ron tích chập (CNN) có hiệu quả phân

biệt cao hơn so với các mô hình truyền thống như SVM, Gradient Boosting Classifier, và KNN do khả năng trích xuất đặc trưng mạnh mẽ và xử lý thông tin không gian tốt của mô hình CNN. Trong khi các mô hình truyền thống như SVM, Gradient Boosting Classifier, và KNN có thể gặp khó khăn trong việc xử lý các đặc trưng phức tạp và không gian, CNN có thể tự động học các mẫu và đặc trưng từ dữ liệu hình ảnh, giúp nâng cao độ chính xác trong việc phân loại và nhận diện bệnh.

4.4 Huấn luyện mô hình

4.4.1 Mô hình GradientBoostingClassifier

Chúng tôi sử dụng trực tiếp mô hình GradientBoostingClassifier từ thư viện sklearn.ensemble. Đầu tiên, mô hình được huấn luyện với giá trị learning_rate = 0.1, thu được độ chính xác (accuracy) là 0.454 và điểm f1-score là 0.449. Sau khi điều chỉnh learning_rate xuống 0.01, độ chính xác vẫn giữ nguyên ở mức 0.454, tuy nhiên f1-score đã giảm xuống mức 0.447. Khi tăng giá trị learning_rate lên 0.05, độ chính xác (accuracy) và f1-score lần lượt đạt 0.491 và 0.484. Như vậy, sau nhiều lần thay đổi giá trị learning_rate, chúng tôi đã đạt được độ chính xác (accuracy) và f1-score cao nhất là 0.5 và 0.494, tương ứng với learning_rate = 0.07.

4.4.2 Mô hình LGBMClassifier

Chúng tôi sử dụng mô hình LGBMClassifier từ thư viện lightgbm của Python. Mô hình này được thử nghiệm với các giá trị learning_rate khác nhau và các kiểu boosting khác nhau để tìm ra cấu hình tốt nhất. Khi sử dụng learning_rate = 0.1 và ba kiểu boosting là gbdt, dart và goss, các cặp độ chính xác (accuracy) và f1-score lần lượt đạt được là (0.537, 0.536), (0.556, 0.557) và (0.546, 0.546). Thử nghiệm với các giá trị learning_rate khác ngoài 0.1 không mang lại kết quả tốt hơn. Do đó, cấu hình với độ chính xác và f1-score cao nhất là 0.556 và 0.557 tương ứng với learning_rate = 0.1 và boosting_type = 'dart'.

4.4.3 Mô hình KNeighborsClassifier

Với mô hình KNeighborsClassifier, chúng tôi sử dụng từ thư viện sklearn.neighbors. Trong quá trình huấn luyện, chúng tôi đã thử nghiệm với nhiều giá trị khác nhau của n_neighbors để tìm ra cấu hình tốt nhất. Với n_neighbors = 2, cả độ chính xác (accuracy) và f1-score lần lượt đạt 0.617 và 0.599. Khi tăng n_neighbors lên 3, độ chính xác (accuracy) bị giảm đi một chút là 0.608 và f1-score lại tăng lên là 0.605. Cuối cùng khi chúng tôi tăng n_neighbors = 20 thì đã thu được độ chính xác (accuracy) và f1-score với giá trị lần lượt là 0.65 và 0.638.

4.4.4 Mô hình RandomForestClassifier

Mô hình RandomForestClassifier được xây dựng sẵn trong thư viện sklearn.ensemble. Chúng tôi đã thử nghiệm với các hàm loss khác nhau như gini, entropy

và `log_loss`. Khi sử dụng hàm loss là `gini`, với nhiều giá trị `n_estimators` thì khi `n_estimators = 200`, độ chính xác (accuracy) và f1-score có giá trị cao nhất lần lượt là 0.725 và 0.72. Với hàm loss là `entropy`, khi huấn luyện mô hình với `n_estimators = 200`, chúng tôi đạt được độ chính xác (accuracy) và f1-score cao nhất là 0.742, 0.738. Tương tự, khi sử dụng hàm loss là `log_loss` và `n_estimators = 120`, độ chính xác và f1-score đạt được là 0.742, 0.739.

4.4.5 Mô hình Support Vector Machines

Chúng tôi sử dụng mô hình SVC từ thư viện `sklearn.svm`. Mô hình được huấn luyện với bốn kernel khác nhau là `rbf` (Radial Basis Function), `linear`, `polynomial` và `sigmoid`. Các cặp độ chính xác và f1-score thu được lần lượt là (0.758, 0.749) cho kernel `rbf`, (0.725, 0.729) cho kernel `linear`, (0.6, 0.581) cho kernel `polynomial`, và (0.467, 0.472) cho kernel `sigmoid`.

4.5 Mô hình Transfer Learning kết hợp Fine-tuning

Chúng tôi sử dụng cấu trúc mô hình `resnet18` để huấn luyện mô hình CNN. `ResNet` là viết tắt của `Residual Network`, là một loại mạng nơ-ron sâu (deep neural network) được thiết kế để huấn luyện các mô hình học sâu hiệu quả, đặc biệt trong bài toán nhận diện hình ảnh.

Mô hình bắt đầu với một lớp `convolutional (conv1)` với 64 bộ lọc, kích thước kernel `7x7`, và `stride 2x2` để giảm kích thước của ảnh đầu vào.

Sau đó là một lớp `Batch Normalization (bn1)` để chuẩn hóa các giá trị đầu vào trước khi áp dụng hàm kích hoạt.

Hàm kích hoạt được sử dụng ở đây là `ReLU` (Rectified Linear Unit) để thêm tính phi tuyến tính vào mạng.

Tiếp theo là một lớp `max pooling (maxpool)` với kernel size `3x3` và `stride 2x2` để giảm kích thước của đầu ra sau lớp `convolutional` đầu tiên.

Tiếp theo mô hình sẽ đi qua 4 layer, mỗi layer sẽ bao gồm 2 basic block. Mỗi `BasicBlock` chứa hai lớp `convolutional (conv1 và conv2)` và hai lớp `Batch Normalization (bn1 và bn2)`, với kích thước kernel là `3x3` và một lớp kích hoạt `ReLU`.

Các `BasicBlock` này được xếp chồng lên nhau trong các tầng của mô hình để tạo ra sự phức tạp và hiệu quả.

Để đảm bảo kích thước của đầu vào và đầu ra của các khối giống nhau, một phần `downsample` được sử dụng khi cần thiết để điều chỉnh kích thước của đầu vào.

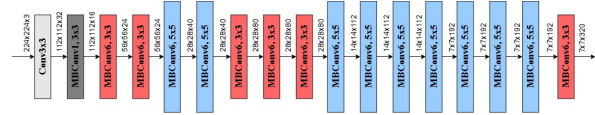
Phần `downsample` này thường là một chuỗi các lớp `convolutional` và `Batch Normalization` với kernel size là `1x1` và `stride 2x2`.

Trước khi đưa vào lớp `fully connected`, một lớp `adaptive average pooling (avgpool)` được sử dụng để chuyển từ tensor 3D sang tensor 2D có kích thước cố định (`1x1`).

Cuối cùng, một lớp `fully connected (fc)` được sử dụng để chuyển từ các đặc trưng hình ảnh đã được trích xuất sang các dự đoán cuối cùng.

Trong trường hợp này, lớp `fully connected` có kích thước đầu vào là 512 và đầu ra là số lượng lớp phân loại.

Như đã trình bày ở phần phương pháp thì hàm mất



Hình 4: Sơ đồ Resnet18.

mát và hàm tối ưu chúng tôi sử dụng là Multiclass cross-entropy và SGD (Stochastic Gradient Descent).

Quá trình huấn luyện mô hình diễn ra qua 300 epochs, mỗi epoch được chia thành các batch để giảm bớt thời gian tính toán và tăng khả năng học của mô hình. Mỗi vòng lặp của quá trình huấn luyện bao gồm các bước sau:

Dự đoán và tính toán hàm mất mát

- Dựa trên cấu trúc mô hình đã được xây dựng và các trọng số hiện có, một lô dữ liệu được đưa vào mô hình để dự đoán.
- Sau đó, sử dụng hàm mất mát Multiclass Cross-Entropy, chúng tôi tính toán giá trị mất mát giữa dự đoán của mô hình và nhãn thực tế của dữ liệu.

Tính toán đạo hàm và cập nhật trọng số

- Sử dụng thuật toán `backpropagation`, chúng tôi tính toán đạo hàm của hàm mất mát đối với các trọng số của mô hình.
- Sau đó, chúng tôi thực hiện bước cập nhật trọng số bằng cách sử dụng thuật toán tối ưu SGD để di chuyển theo hướng giảm giá trị của hàm mất mát.
- Số lần cập nhật trọng số trong mỗi epoch là tích của tổng số batch và tổng số epochs, đảm bảo rằng mô hình có đủ cơ hội học và điều chỉnh trọng số đến khi hội tụ.

Trong quá trình huấn luyện mô hình chúng tôi đã Fine-tuning các trọng số để mô hình Transfer learning phù hợp với bài toán của chúng tôi. Chúng tôi đã điều chỉnh giá trị của `learning rate`, `epochs`, `dropout rate`, `regularization` và `momentum`.

Tiêu chuẩn dừng Quá trình huấn luyện được lặp lại qua nhiều epochs cho đến khi đạt được tiêu chuẩn dừng đã được định trước hoặc khi không có sự cải thiện đáng kể trong hiệu suất của mô hình trên tập test.

4.6 Kiểm tra trên tập Test

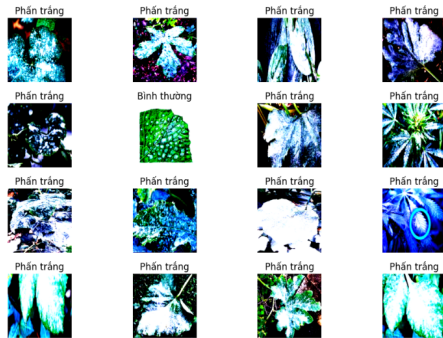
Các mô hình được đánh giá thông qua các độ đo như độ chính xác (accuracy) và f1 score trên tập test. Điều này giúp đảm bảo tính khách quan và độ tin cậy của mô hình khi triển khai vào thực tế.

Qua bảng 1 ta có thể thấy được độ chính xác và f1-score của mô hình Transfer learning kết hợp Fine-tuning cao hơn nhiều so với các mô hình khác. Điều này cũng đúng với dự đoán của chúng tôi trong phần phân tích dữ liệu về khả năng trích xuất đặc trưng của mạng CNN với bài toán phân loại của chúng tôi.

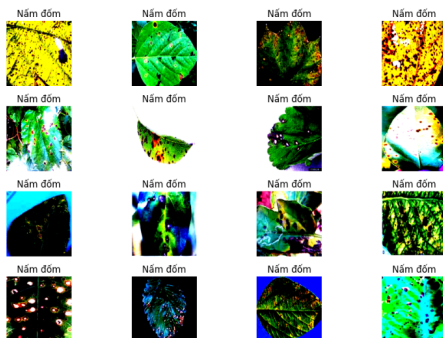
Bảng 1: Bảng so sánh Accuracy và F1-Score của một số mô hình

Mô hình	Accuracy	F1-Score
GradientBoostingClassifier	0.5	0.494
LGBMClassifier	0.556	0.557
KNeighborsClassifier	0.65	0.638
RandomForestClassifier	0.742	0.739
Support Vector Machines	0.758	0.749
Transfer Learning kết hợp Fine-tuning	0.986	0.991

Hình 5, 6 và 7 là một số ảnh trong tập test và dự đoán được đưa ra bởi mô hình Transfer learning.



Hình 5: Kiểm tra bệnh Phấn Trắng.



Hình 6: Kiểm tra bệnh Nấm Đốm.

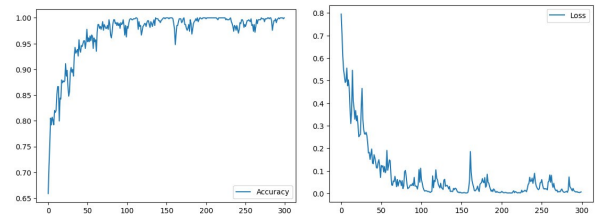


Hình 7: Kiểm tra lá Bình Thường.

5 Kết luận

Độ chính xác trên tập Test

- Sau quá trình huấn luyện và đánh giá, chúng tôi đã đạt được độ chính xác cao nhất là 98.(66)% trên tập test, sử dụng mô hình Transfer learning kết hợp Fine-tuning.
- Kết quả này là kết quả của việc đánh giá hiệu suất của mô hình trên dữ liệu mới và không được sử dụng trong quá trình huấn luyện, đảm bảo tính khách quan và đáng tin cậy của mô hình.



Hình 8: Sơ đồ biến thiên độ chính xác và hàm mất mát khi huấn luyện của mô hình Transfer Learning.

Đánh giá hiệu quả của mô hình ResNet, Multi-class Cross-Entropy, SGD

- Chúng tôi kết luận rằng mô hình ResNet đã thể hiện hiệu quả cao trong bài toán phân loại của chúng tôi.
- Sự kết hợp giữa kiến trúc mạng sâu và các kỹ thuật tiền xử lý dữ liệu đã giúp mô hình ResNet đạt được độ chính xác và độ tin cậy cao trong việc phân loại các loại bệnh trên lá cây.
- Hiệu suất của mô hình ResNet cung cấp một nền tảng mạnh mẽ cho việc ứng dụng trong thực tế và có thể cải thiện hiệu suất sản xuất và quản lý cây trồng hiệu quả hơn.
- Hàm mất mát Multiclass Cross-Entropy thường được sử dụng hiệu quả trong bài toán phân loại hình ảnh. Đặc biệt, tốc độ cải thiện độ chính xác qua mỗi vòng lặp của thuật toán đào tạo là rất đáng chú ý.
- Phương pháp tối ưu SGD cũng được chứng minh là hữu ích trong bài toán phân loại hình ảnh. Nó không chỉ giúp tiết kiệm tài nguyên mà còn có khả năng cập nhật trọng số gần với điểm cực tiểu toàn cục. Mặc dù có thể có sự giảm độ chính xác do việc cập nhật trọng số ngẫu nhiên trong SGD, nhưng xu hướng chung vẫn là tìm đến điểm cực tiểu toàn cục, như được minh họa trong Hình 10.

Trong tương lai, chúng tôi hy vọng có thể tiếp tục nghiên cứu và phát triển mô hình này để áp dụng vào các ứng dụng thực tiễn và giải quyết các vấn đề trong lĩnh vực nông nghiệp một cách hiệu quả và bền vững.

Tài liệu

- [1] GL Anoop and C Nandini. Rice crop yield prediction using multi-level machine learning techniques. *Journal of Computational and Theoretical Nanoscience*, 17(9-10):4280–4286, 2020.

- [2] V Bansod. Rice crop disease identification and classifier. *Int J Comput Sci Eng*, 7(11):45–48, 2019.
- [3] Jyoti Dinkar Bhosale and S Lomte Santosh. Identification and classification of rice plant diseases using machine learning. *International Journal of Computer Applications*, 975:8887.
- [4] MM Billah, MP Islam, and MG Rahman. Identification of rice diseases using artificial neural network. *Journal of the Bangladesh Society for Agricultural Science and Technology*, 4:189–194, 2007.
- [5] Trịnh Trung Hải, Nguyễn Hà Huy Cường, and Ninh Khánh Duy. Mô hình fast r-cnn cải tiến cho giải pháp nhận dạng, phát hiện trái dưa thời kỳ chín. *Tạp chí Khoa học và Công nghệ-Đại học Đà Nẵng*, pages 94–98, 2022.
- [6] Ngọc Tú Nguyễn, Thị Thanh Phương Bùi, Hoàng Nam Lê, and Nam Thanh Ngô. Nghiên cứu phương pháp phát hiện một số sâu bệnh trên lúa sử dụng đặc trưng sift. *Bản B của Tạp chí Khoa học và Công nghệ Việt Nam*, 61(8), 2019.
- [7] Ngô Quang Hiếu và Trương Quốc Bảo Phan Tấn Phước. Nhận dạng một số bệnh trên bông xoài sử dụng kỹ thuật học sâu chuyển tiếp. *Tạp chí Khoa học Đại học Cần Thơ*, 60(1), 2024.
- [8] Tejas Tawde, Kunal Deshmukh, Lobhas Verekar, Ajay Reddy, Shailendra Aswale, and Pratiksha Shetgaonkar. Rice plant disease detection and classification techniques: a survey. *international Journal of Engineering research & Technology*, 10(7):560–567, 2021.
- [9] Nguyễn Minh Triết, Trương Quốc Bảo, and Trương Quốc Định. Tự động nhận dạng một số loại sâu bệnh trên lá bưởi sử dụng công nghệ ảnh. *Tạp chí Khoa học Đại học Cần Thơ*, pages 88–95, 2017.