HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA VIỄN THÔNG II

ĐỀ TÀI: TỪ ĐIỂN THỰC VẬT

Mã đề tài: 61 – SV – 2018 – VT2

BÁO CÁO NGHIÊN CỨU KHOA HỌC

Chuyên Ngành: Điện Tử Truyền Thông

Giảng viên hướng dẫn: PGS.TS. Võ Nguyễn Quốc Bảo

Sinh viên thực hiện: Nguyễn Hoài Nam

Lớp: D15CQVT01-N

Mã sinh viên: N15DCVT036

Tháng 12/2018 - Hồ Chí Minh

LỜI CẢM ƠN

Đầu tiên, cho phép em gửi lời cảm ơn chân thành đến thầy hướng dẫn nghiên cứu khoa học **PGS.TS. Võ Nguyễn Quốc Bảo**. Thầy luôn hướng dẫn em tận tình trong suốt quá trình thực hiện đề tài nghiên cứu khoa học. Thầy đã tận tình chỉ bảo, giúp đỡ em giải quyết các vướng mắc, khó khăn để có thể hoàn thành đề tài nghiên cứu khoa học này.

Tiếp đến, em xin gửi lời cảm ơn đến thầy **PGS.TS. Võ Nguyễn Quốc Bảo** - trưởng khoa Viễn Thông 2 đã tạo điều kiện và hỗ trợ hoạt động nghiên cứu khoa học. Và các Thầy Cô đã và đang giảng dạy tại trường Học Viện Công Nghệ Bưu Chính Viễn Thông cơ sở TP. Hồ Chí Minh đã giúp em có những kiến thức bổ ích để thực hiện đề tài nghiên cứu khoa học này. Kính chúc các các Thầy Cô dồi dào sức khỏe, thành đạt và ngày càng thành công hơn nữa trên con đường sự nghiệp giáo dục của mình.

Ngoài ra, em xin gửi lời cảm ơn sâu sắc đến tập thể phòng **LAB B02** đã không ngại bỏ thời gian, công sức giúp đỡ em thực hiện đề tài nghiên cứu khoa học này. Cuối cùng, em cũng xin cảm ơn các anh chị, bạn bè, gia đình đã luôn quan tâm, động viên và giúp đỡ em trong suốt khoảng thời gian thực hiện.

Em xin chân thành cảm ơn tất cả mọi người!

Tp. Hồ Chí Minh, tháng 10 năm 2018.

(Sinh viên thực hiện)

Nguyễn Hoài Nam

Mục lục

Danh sách hình vẽ

Danh sách bảng

Thuật ngữ

Machine Learning: Máy học hay học máy.

Tensorflow: Thư viện mã nguồn mở hỗ trợ ứng dụng có liên quan đến học máy, mạng thần kinh nhân tạo.

API: Application Programming Interface là phương thức kết nối giữa thư viện và các ứng dụng.

GPU: Graphics Processing Unit - Đơn vị xử lý đồ họa.

Anaconda: Phần mềm quản lý gói, quản lý môi trường, thư viện khoa học bổ sung hỗ trợ cho các ứng dụng học máy và khoa học dữ liệu.

Training: Quá trình huấn luyện. Background: Ảnh nền. Foreground: Ảnh trước. Anchor: RPN xếp khung khu vực. ROI: Region of Interest - Vùng quan tâm. XML: Ngôn ngữ đánh dấu mở rộng. ID: Chỉ số nhận diện. FPS: Frame per second - Khung hình trên giây.

Chương 1

TỔNG QUAN

1.1 Tổng quan về Machine Learning

1.1.1 Toàn cảnh

Học máy (Machine learning)[1], hay còn gọi là máy học là một thuật ngữ được giới công nghệ được nhắc đến khá nhiều trong thời gian gần đây. Học máy tương tự việc chúng ta "dạy" cho máy móc biết tự học hỏi và cập nhật kiến thức, để chúng ngày càng trở nên thông minh hơn.

Machine Learning có khả năng nhân diện tự động, đưa ra những dự đoán mô hình dư liệu mà không cầ phải lập trình tường tận.

1.1.2 Lịch sử

Trước nay, máy móc là những thiết bị do con người tạo ra để phục vụ một mục đích hoặc công việc hay một công đoạn cụ thể nào đó trong một quy trình sản xuất, giúp tăng năng suất lao động, kể cả việc thay thế cho con người trong những điều kiện làm việc khắc nghiệt và nguy hiểm. Về cơ bản, các máy móc này được lập trình sẵn để chúng cứ tự hoạt động, và chỉ thay đổi quy trình làm việc khi chúng ta thay đổi chương trình hoạt động cho chúng mà thôi.

Ý tưởng về học máy đã bắt đầu xuất hiện ngay từ những năm 50, 60 thế kỷ trước, sau khi con người có được các máy móc tự động. Ở đây, các nhà khoa học mong muốn các

máy móc cũng phải biết "tự cập nhật kiến thức" để thông minh hơn, phù hợp với điều kiện và hoàn cảnh mới, thay vì ta phải loại bỏ chương trình cũ để đưa vào các chương trình phần mềm khác.

1.1.3 Phân loại

Học máy chủ yếu dựa trên 4 thao tác: Phân loại/liệt kê thông tin; dự đoán những sự kiện nhất định trên cơ sở các mô hình đã được nhận dạng; phát hiện/nhận dạng các mô hình chưa được biết đến và sự phụ thuộc giữa chúng; phát hiện sự bất thường và các sự kiện chưa được dự đoán.

Học máy được chia làm 4 kỹ thuật điển hình:

1. **Học máy có giám sát**

   Học có giám sát (Supervised Learning)[3]. Đây là trường hợp các máy tự học trên cơ sở các ví dụ. Điều này giống như các học trò nhận được từ khóa cho bài thử nghiệm và được yêu cầu tìm lời giải. Dữ liệu đầu vào được sử dụng để tìm các mối liên quan nhằm giải quyết một vấn đề cụ thể. Nếu xác lập thành công một mô hình, thì mô hình đó được sử dụng trong các trường hợp tương tự.

   Có thể nêu ra các ví dụ về ứng dụng dạng này, như phát hiện sự quá tải, cá nhân hóa tương tác, nhận dạng giọng nói, văn bản, hình ảnh, . . .

2. **Học máy nửa giám sát**

   Học nửa giám sát (Semi- Supervised Learning)[2]. Trong trường hợp này, máy móc nhận được cả dữ liệu đầu vào được gắn nhãn (tức dự đoán các dữ liệu tương ứng ở đầu ra với các ví dụ cụ thể), cũng như dữ liệu đầu vào không được gắn nhãn (tức đòi hỏi sắp xếp cho dữ liệu đầu ra và tìm kiếm câu trả lời).

   Kiểu học máy này được sử dụng trong trường hợp khi mà một cơ quan, tổ chức có quá nhiều dữ liệu, hoặc khi các thông tin khác biệt đến mức không thể sắp xếp câu trả lời cho mỗi thông tin, đòi hỏi hệ thống tự đề xuất câu trả lời và có thể tạo ra những mô hình chung. Ví dụ về ứng dụng dạng này có thể liệt kê như: Nhận dạng giọng nói và hình ảnh, phân loại các trang web,. . .

3. **Học máy không giám sát**

   Học không giám sát (Unsupervised Learning). Trong trường hợp này, máy móc không có “chìa khóa" trả lời và buộc phải tự phân tích dữ liệu, tìm kiếm mô hình và tìm mối tương quan có liên quan để đưa ra đáp án.

   Kiểu học máy này khiến người ta liên tưởng tới cách hoạt động của bộ não con người: Bộ não rút ra kết luận trên cơ sở quan sát tự phát và trực giác. Thêm vào đó, cùng với sự gia tăng các tệp dữ liệu, kết luận đầu ra sẽ trở nên ngày càng chính xác hơn. Ví dụ về ứng dụng dạng này như: Phân tích gói hàng, phát hiện sự bất thường, nhận dạng đối tượng tương tự,...

4. **Học máy tăng cường**

   Học tăng cường (Reinforcement Learning). Trong trường hợp này máy móc nhận được bộ thao tác và những quy định được phép từ trước. Hoạt động trong khuôn khổ đó, máy móc tiến hành phân tích và quan sát kết quả các thao tác để tự cập nhật và đưa ra những kết quả ngày càng tốt hơn.

   Có thể so sánh kiểu học máy này với học chơi bóng rổ. Các quy định về bước chân, lỗi hay bóng ra ngoài sân,...là không thay đổi. Ngược lại, cách mà một đội bóng giành điểm (người chơi ném bóng từ khoảng cách xa, chạy đến gần rổ hay chuyền bóng cho người chơi khác) phụ thuộc vào quyết định tức thời của người chơi. Ví dụ về ứng dụng này như: Hoa tiêu (lựa chọn hành trình dựa trên thông tin về sự gia tăng mật độ giao thông trên đường), trò chơi điện tử, robot học,...

1.1.4 Ứng dụng [1]

- **Các dịch vụ tài chính**

  Ngân hàng và những doanh nghiệp hoạt động trong lĩnh vực tài chính sử dụng công nghệ Machine Learning với mục đích: xác định lỗi trong dữ liệu và ngăn chặn lừa đảo. Từ đó sẽ biết được các cơ hội đầu tư hoặc thông báo đến nhà đầu tư thời điểm giao dịch hợp lý. Tìm hiểu, khai thác dữ liệu để có thể tìm được những khách hàng đang có hồ sơ rủi ro cao hoặc sử dụng giám sát mạng để chỉ rõ những tín hiệu lừa đảo.

- **Chính phủ**

  Các tổ chức chính phủ hoạt động về an ninh cộng đồng hoặc tiện ích xã hội sở hữu rất nhiều nguồn dữ liệu có thể khai thác. Ví dụ, khi phân tích dữ liệu cảm biến, chính phủ sẽ tăng mức độ hiệu quả của dịch vụ và tiết kiệm chi phí. Machine Learning còn hỗ trợ phát hiện gian lận và giảm thiểu khả năng trộm cắp danh tính.

- **Chăm sóc sức khỏe**

  Machine Learning là 1 xu hướng phát triển nhanh chóng trong ngành chăm sóc sức khỏe, nhờ vào sự ra đời của các thiết bị và máy cảm ứng đeo được sử dụng dữ liệu để đánh giá tình hình sức khỏe của bệnh nhân trong thời gian thực (real-time). Công nghệ Machine Learning còn giúp các chuyên gia y tế xác định những xu hướng hoặc tín hiệu để cải thiện khả năng điều trị, chẩn đoán bệnh.

- **Marketing và Sales**

  Dựa trên hành vi mua hàng, các trang web sử dụng Machine Learning phân tích lịch sử mua hàng, từ đó giới thiệu những vật dụng mà bạn có thể sẽ quan tâm và yêu thích. Khả năng tiếp nhận dữ liệu, phân tích và sử dụng những dữ liệu đó để cá nhân hóa trải nghiệm mua sắm (hoặc thực hiện chiến dịch Marketing) chính là tương tai của ngành bán lẻ.

- **Giao thông**

  Hệ thống giám sát giao thông, đưa ra dự đoán từ đó cảnh báo đến người điều khiển phương tiện liên quan đến các vấn đề tiềm tàng thông qua dữ liệu về lưu lượng xe, tình trạng mặt đường, cơ sở hạ tầng tại đoạn đường phương tiện sắp đí qua.

Máy móc có thể tự học được nhiều thứ, nhưng không phải là tất cả. Mặc dù khả năng hoàn thiện học máy là rất lớn, nhưng công nghệ này vẫn có một số hạn chế nhất định. Một trong những hạn ché đó là máy móc không có khả năng tư duy sáng tạo và không đưa ra được các dự đoán, giả định nếu thiếu dữ liệu thích hợp.

Ngoài ra, máy móc cũng không thể thu nhận các kích thích mới, chưa được biết đến, bởi mỗi một sự thay đổi dữ liệu đều gây ảnh hưởng đến công việc của máy móc. Điều đó có nghĩa là có thể gây ảnh hưởng (có chủ đích hoặc do nhầm lẫn) đối với các kết quả được thể hiện, thông qua việc thay đổi các thông tin cung cấp cho hệ thống.

1.2 Tổng quan đề tài

1.2.1 Lý do chọn đề tài

Trong bối cảnh cuộc "Cách mạng công nghiệp 4.0" xuất hiện ngày càng nhiều các ứng dụng có liên quan đến Machine Learning (Máy học), một trong những khía cạnh liên quan được quan tâm là khoa học máy tính nói chung và xử lý hình ảnh (hay Computer Vision) nói riêng. Trong phạm vi đề tài này ta sử dụng thư viện Tensorflow trong việc xử lý cũng như nhận dạng vật thể[4], các đối tượng nhận dạng được tổng hợp. Từ hướng nghiên cứu này phát triển thành một ứng dụng tra cứu thông tin về thực vật, cụ thể là từ điển về thực vật. Bên cạnh đó đề tài này sẽ giới thiệu về nguyên lý, cách thức hoạt động của thư viện được sử dụng. Cùng với quy trình trong việc huấn luyện máy học những đối tượng cụ thể.

Tình hình trong nước:

- Ứng dụng tra cứu cây thuốc Việt Nam của nhóm tác giả thuộc ĐHQG Hà Nội. Ứng dụng nhận dạng hình ảnh lá cây thuốc, từ đó đưa ra thông tin về công dụng, các đặc điểm sinh học của cây. Hạn chế về kho dữ liệu, chỉ nhận diện được một bộ phận cây thuốc.

- Cơ sở dữ liệu của của nhóm tác giả trang web botanyvn, tổng hợp thông tin về khoa học và ứng dụng tài nguyên Thực vật Việt Nam. Ngoài ra còn là một thư viện mở trang bị cho tất cả các đối tượng đặc biệt là học sinh nhằm nâng cao hiểu biết và đem đến những điều thú vị khác trong thế giới thực vật.

Tình hình thế giới:

- Ứng dụng Leafsnap của nhóm nghiên cứu thuộc đại học Columbia, theo đó, điện thoại sẽ chụp lại hình ảnh của chiếc lá, thông qua hệ thống nhận diện, thông tin được hiển thị sau đó trên smartphone.

- Cở sở dữ liệu tổng hợp Garden.org là nơi tổng hợp lưu trữ thông tin về các loài thực vật, hoạt động như một thư viện mở về thực vật trên toàn thế giới.

Vấn đề còn hạn chế:

- Hệ thống kho dữ liệu của ứng dụng này chưa nhiều, đông thời ứng dụng chỉ phân biệt hình dạng đặc trưng của đối tượng.
- Những ứng dụng này chưa thật sự tiện lợi khi sử dụng. Cụ thể, người dùng phải biết và nhập tên đối tượng vào ô tìm kiếm như truyền thống.
- Tốc độ xử lý còn chưa hạn chế, độ chính xác mang tính tương đối.
- Chưa được tập trung phát triển chức năng xử lý hình ảnh.

1.2.2 Giới thiệu đề tài

Đề tài sử dụng thư viện Tensorflow bằng ngôn ngữ Python để tiến hành nhận dạng dựa trên dữ liệu để được huấn lyện trước đó, thông qua camera, chương trình sẽ tiến hành nhận diện đối tượng mà camera quan sát được. Dữ liệu về thực vật được huấn luyện trước đó, từ đó dựa vào dữ liệu đã huấn luyện, phần mềm sẽ đưa ra dự đoán các đối tượng sẽ nhận dạng trong tương lai.

1.2.3 Khả năng ứng dụng của đề tài

Từ điển thực vật là một trong những ứng dụng thư viện Tensorflow trong việc nhận biết đối tượng thực vật. Tạo ra ứng dụng tra cứu thông tin nhanh chóng, đầy đủ và chính xác.

Pham vi ứng dụng rộng rãi có thể kể đến là trong trường học, trung tâm đào tạo, nghiên cứu. Đối tượng hướng đến là học sinh giúp khơi dậy được niềm đam mê thực vật, cũng có thể là những nhà nghiên cứu thực vật, với giao diện trực quan và nhanh chóng có thể xác định được đối tượng một cách nhanh chóng

1.2.4 Hướng phát triển của đề tài

Sử dụng trong các tính năng nhận diện, đưa ra dự đoán từ dữ liệu được cung cấp có thể nhận dạng nhiều đối tượng khác nhau, mở rộng thêm kho dữ liệu, giúp chuyên môn hóa tính năng nhận dạng. Không dừng lại ở từ điển thực vật mà còn có thể mở rộng ra thành từ điển bách khoa với cơ sở dữ liệu khổng lồ. Ngoài ra còn có thể sử dụng tính năng nhận

dạng vật thể phát triển camera thông minh, giám sát cảnh báo thông tin cho người dùng trước những sự việc xảy ra trong tương lai. Thêm nữa, không chỉ sử dụng đơn lẻ, từ điển thực vật còn có thể được đưa vào các ứng dụng học tập ở các môn Sinh học, thực vật, tự nhiên, . . . làm phong phú thêm cho các tính năng trong hệ sinh thái ứng dụng giáo dục.

Chương 2

THƯ VIỆN, MÔI TRƯỜNG

2.1 Tensorflow

Tensorflow là thư viện mã nguồn mở hỗ trợ lập trình các luồng dữ liệu, dựa trên thư viện toán học, dùng cho các ứng dụng liên quan đến học máy như mạng thần kinh nhân tạo sử dụng để nghiên cứu và phát triển sản phẩm của Google.

Tensorflow còn là một "bộ" phần mềm, một hệ sinh thái với mục đích phát triển mô hình học sâu. Thư viện gồm các công cụ hỗ trợ xây dựng cũng như triển khai ứng dụng cụ thể, gồm:

1. **Tensorflow(API):** Thành phần này của Tensorflow cung cấp API định nghĩa mô hình, huấn luyện mô hình từu dữ liệu được cung cấp.

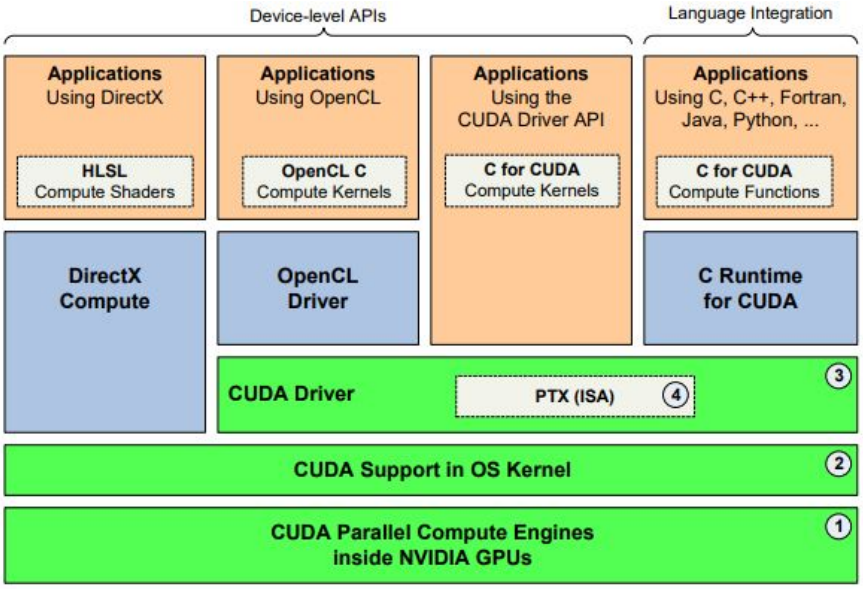
2. **TensorFlow Serving:** Thành phần giúp triển khai mô hình huấn luyện trước. Tensorflow Serving có thể dễ dàng thay đổi từ mô hình cũ sang mô hinh mới bất cứ lúc nào.

3. **TensorBoard:** Thành phần thứ ba trong hệ sinh thái là Tensorboard. Hỗ trợ trong việc phân tích, hình dung và sữa lỗi đồ thị Tensorflow.

2.2 Nhân CUDA

CUDA (Compute Unified Device Architecture) là động cơ tính toán trong các GPU của hãng NVIDIA. Người dùng cài đặt các thuật toán chạy trên GPU. Kiến trúc CUDA

hỗ trợ mọi chức năng tính toán thông qua ngôn ngữ C. Các GPU của NVIDIA được trang bị kiến trúc tính toán song song CUDA, bao gồm cả trình điều khiển thiết bị CUDA vốn được nhúng bên trong trình điều khiển thiết bị đồ họa do NVIDIA cung cấp.



Hình 2.1: Cấu trúc nhân CUDA

Trong đề tài này, thông qua nhân CUDA để thực hiện tính toán trong phương thức nhận dạng đối tượng.

2.3 Bộ dữ liệu training

Tensorflow cung cấp mô hình nhận dạng vật thể có thể hiểu là bước tiền xử lý, phân loại đặc trưng của cấu trúc mạng thần kinh. Đó là bộ dữ liệu CoCo.

Một trong nhiều thành phần quan trọng nhất trong học sâu là bộ dữ liệu, bộ dữ liệu tốt sẽ đóng góp vào mô hình làm tăng độ chính xác cao. CoCo như là môt thư viện cung cấp thông tin về bộ dữ liệu được chuẩn bị trước chứa dữ liệu về các ảnh nền liên quan đến cuộc sông thường ngày, những đối tượng này được phân biệt, gán nhãn để phân biệt. Trong quá trình training, thông qua bộ dữ liệu được sử dụng, chương trình có thể lọc ra được ảnh nền và dễ dàng tách riêng đối tượng, giúp độ chính xác trong việc nhận dạng được tăng lên.

2.4 Thiết lập môi trường

CUDA Toolkit cung cấp môi trường phát triển tăng tốc hiển thị GPU. Với CUDA Toolkit, có thể phát triển, tối ưu hóa, triển khai ứng dụng trong xử lý ảnh và video.

Cài đặt môi trường làm việc với Anaconda. Tại đây, phần mềm Anaconda giúp người dùng quản lý môi trường, các gói thư viện liên quan trong chương trình nhận dạng.

Chương 3

NGUYÊN LÝ

3.1 Đối tượng nhận dạng

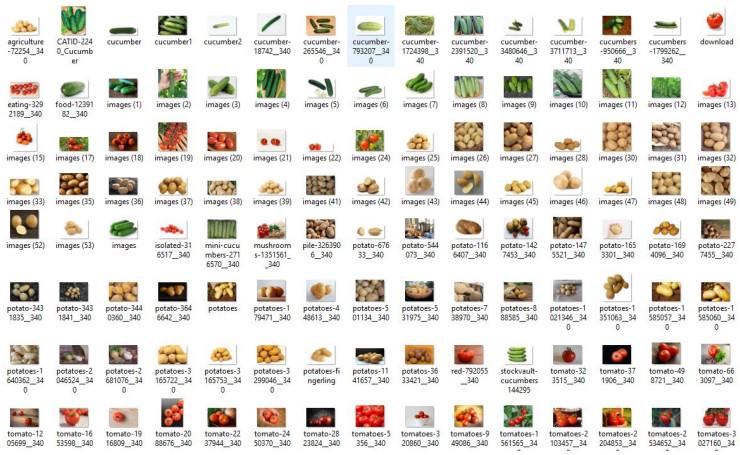
3.1.1 Thu thập hình ảnh

Đối tượng nhận dạng trong đề tài này được lưu trữ dưới dạng hình ảnh, hình ảnh của các đối tượng sẽ được phân loại thành 2 nhóm: đối tượng nhận dạng và ảnh chứa đối tượng.

Tensorflow cần số lượng ảnh rất lớn để quá trinh nhận dạng có thể diễn ra chính xác nhất, để việc nhận dạng trở nên mạnh mẽ hơn, quá trình training ảnh nên có những đối tượng ngẫu nhiên, trong những đối tương cần nhận dạng. Đồng thời, cần phải đa dạng về background với những điều kiện về ánh sáng hay độ tương phản.

Tổng số ảnh tùy thuộc vào số đối tượng tiến hành nhận dạng, tối thiểu cho một đối tượng là 50 ảnh, tổng số ảnh được chia là 20% cho ảnh đối tượng và 80% còn lại cho background.

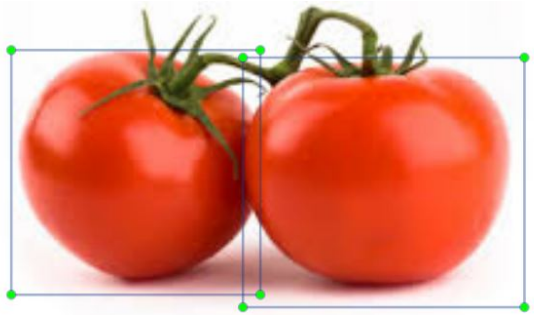
Trong phạm vi đề tài, có 300 tấm ảnh của đối tượng được trainig, giới hạn nhận dạng 3 đối tượng chính là: Cà chua, Dưa leo và Khoai tây. Những quy định về kích thước ảnh, dung lượng sẽ ảnh hướng đến quá trình training. Cụ thể ảnh có kích thước lớn, dung lượng ảnh cao sẽ mất nhiều thời gian trong giai đoạn training.



Hình 3.1: Hình ảnh cho quá trình training

3.1.2 Gán nhãn

Hình ảnh sau khi được tập hợp, bước tiếp theo là gán nhãn, hay xác định đối tượng nhận diện.



Hình 3.2: Gán nhãn cho đối tượng

Sau khi gán nhãn cho các đối tượng, mỗi lần gán nhãn và lưu từng ảnh, sẽ có một file .xml được tạo ra, file này đóng vai trò một trong những thành phần ngõ vào quá trình training (file TFRecord).

3.2 Tạo bảng đồ các nhãn và cấu hình training

3.2.1 Bản đồ nhãn

Bản đồ các nhãn cho chương trình training biết được từng đối tượng được định nghĩa theo tên và số id.

```
item {
id: 1
name: 'cucumber'
}
item {
  id: 2
  name: 'potato'
}
item {
  id: 3
  name: 'tomato'
}
```

File labelmap có phần mở rộng là pbtxt. Đây là phần mở rộng của Protocol Buffers. Đây là cấu trúc dữ liệu ngôn ngữ tương tự như xml nhưng với tốc độ nhanh hơn, dung lượng nhỏ, dễ dàng bổ sung thêm các trường mà không phá vỡ cấu trúc lớp.

3.2.2 Cấu hình training

Mô hình CNN

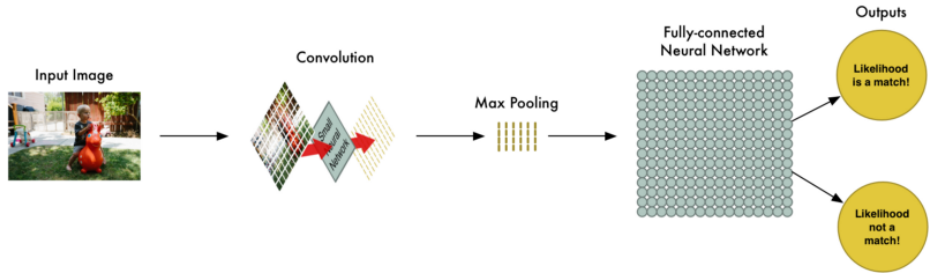
CNN là một thuật toán để tìm kiếm vật thể trong ảnh. Thuật toán này sẽ có đầu ra là những điểm khớp hoặc không khớp với vật thể ban đầu.

Quá trình nhận dạng diễn ra như sau:

- Bắt đầu với ý tưởng chia ảnh có chứa đối tượng thành một mạng lưới chồng chéo nhau.
- Tiến hành đưa lần lượt từng mảnh ảnh đã chia vào mạng nơ-ron, vì có cùng trọng số mà đầu vào khác nhau nên đầu ra khác nhau. Mục đích này là tìm kiếm những phần nhỏ đặc biệt của bức ảnh. Trong một bức ảnh lớn, ta không thể tìm ra vật thể. Giải pháp là tách ảnh ra để tìm các chi tiết cấu thành nên ảnh bởi vì vật thể nắm bất kì chỗ nào trong ảnh vẫn là vật thể đó. Vật thể ở góc hay ở giữa ảnh, vẫn có đầy đủ các chi tiết như thế. Dữ liệu ngõ ra được lưu trữ thành mảng cho từng mảnh.

- Ở ngõ ra nhận được chuỗi mảng rất lớn, cần phải giảm đi kích thước của các mảng này. Để giảm chiều của mảng, sử dụng thuật toán tách lọc. Với mỗi ô trống 2x2 của mảng giá trị, chỉ lưu lại giá trị lớn nhất - đặc trưng rõ ràng nhất của ảnh.

- Sau khi có được chuỗi mảng được tách lọc.Tiến hành kết nối tất cả đặc trưng đó lại tạo thành Fully-connected Neural Network - mạng nơron kết nối hoàn chỉnh. Gía trị của mạng mới sẽ dùng làm đầu vào cho mạng nơron phân nhóm để tìm ra kết quả cuối cùng. Kết nối mạng giống như khi tìm được tay, chân, đầu, ta cần lắp ghép lại thành 1 người hoàn chỉnh. Vì phương pháp này không phụ thuộc vào vị trí của vật thể trong ảnh, mà phụ thuộc vị trí tương đối của các chi tiết đặc trưng với nhau, nên dù đối tượng ở vị trí nào cũng sẽ nhận biết được.



Hình 3.3: Nhận dạng bằng CNN

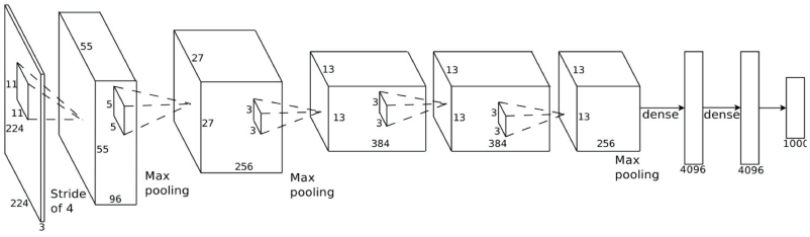
Quá trình nhận diện đối tượng là một chuỗi các bước: tích chập, giảm mẫu và kết nối mạng hoàn chỉnh.

Khi giải quyết vấn đề thực tế, những bước này có thể được kết nối rất nhiều lần. Bạn có thể có 2, 3 thậm chí 10 lớp tích chập. Bạn có thể chèn giảm mẫu vào bất cứ vị trí nào bạn muốn. Ý tưởng là bắt đầu từ ảnh lớn, từng bước một chúng trích xuất đặc trưng, giảm chiều để có kết quả cuối cùng. Càng nhiều bước tích chập, mô hình càng có thể nhận diện đặc trưng phức tạp.

Việc có nhiều lớp tích chập giúp ta tìm được các đặc trưng nhỏ hơn của vật thể. Giờ chúng ta nhận diện các chi tiết vật thể. Nhiều lớp tích chập hơn nữa giúp ta nhận diện các chi tiết nhỏ hơn. Cứ tiếp tục như thế, ta có thể tìm vật thể bằng cách tìm đặc trưng ở cấp độ phân tử. Cũng giống như nhận diện đối tượng qua phân tử (tưởng tượng như vậy), hầu hết các con số trích xuất từ học máy là những đặc trưng mà có thể con người không

hình dung được, nhưng máy tính lại biết được. Người ta thường coi đó là hộp đen thần kì của học máy. Hiện nay, cũng đã tồn tại một số hệ thống nhận diện ảnh tốt hơn con người.

Vì nhận dạng ở cấp độ nhỏ hay rất nhỏ, nhận diện giữa gà hay chó, giữa ô tô hay xe máy đều sẽ có một số quá trình tách lọc đặc trưng giống nhau. Nên mạng nơ - ron dùng để phân biệt động vật có thể được tái sử dụng (transfer learning) để phân biệt xe cộ, các loại hoa, đồ vật... và chỉ cần tinh chỉnh trọng số của mạng ở những lớp cuối cùng.



Hình 3.4: Mạng lưới CNN trong thực tế

Mô hình Faster R-CNN

Faster RCNN là một thuật toán để tìm kiếm vị trí của vật thể trong ảnh. Thuật toán này sẽ có ngõ ra là những hình hộp, cùng với vật thể bên trong hộp đó. Phiên bản đầu tiên của Faster RCNN là RCNN, với nguyên lý khá đơn giản.

Tác giả sử dụng một thuật toán gọi là selective search để đưa ra các bounding boxes, hay còn gọi là region proposals, chứa các vùng có thể có vật thể ở trong. Sử dụng các mạng đã được huấn luyện sẵn để tính toán feed-forward các regions thu được convolutional features của mỗi region, sau đó huấn luyện SVM để xác định được vật thể nào được chứa trong region proposal đó. Sử dụng Linear Regression để hiệu chỉnh các giá trị (vị trí các đỉnh) của region proposer

Sử dụng các mạng huấn luyện sẵn để feed-forward các region proposals, sẽ tốn nhiều thời gian bởi với mỗi ảnh thuật toán selective search sẽ cho ra hàng nghìn region proposals. Ta sẽ chỉ feed-forward một lần đối với ảnh gốc, thu được convolutional features của ảnh đó. Ví dụ với một hình ảnh có kích thước 600x600x3, ta sẽ thu được convolutional features với kích thước 37x37x512. Kích thước của features bị giảm nhỏ khoảng 16 lần. Dựa vào kích thước cùng vị trí của các region proposals đối với ảnh gốc, ta sẽ tính toán

được vị trí của region proposal trong convolutional features. Sửa dụng giá trị convolutional features của region proposal, ta dự đoán được vị trí các đỉnh của bounding box cũng như vật thể nằm trong bounding box là gì.

Đối với Fast RCNN , do chia sẻ tính toán giữa các region trong ảnh, tốc độ thực thực thi của thuật toán đã được giảm từ 120s mỗi ảnh xuống 2s. Phần tính toán gây ra nghẽn chính là phần đưa ra các region proposal đầu vào, chỉ có thể thực thi tuần tự trên CPU. Faster RCNN giải quyết vấn đề này bằng cách sử dụng DNN để tính toán các region proposals này.

RPN

RPN giải quyết các vấn đề về thời gian thực thi bằng cách huấn luyện mạng neural network để đảm nhận thay vai trò của các thuật toán như selective search vốn rất chậm chạp.

Một Region Proposal Network nhận đầu vào là ảnh với kích thước bất kì và cho đầu ra là region proposal (tập vị trí của các hình chữ nhật có thể chứa vật thể), cùng với xác suất chứa vật thể của hình chữ nhật tương ứng. Faster R-CNN gồm có 2 mạng chính:

Mạng đề nghị khu vực(Region Proposal Network) tạo ra khu vực đề nghị cà mạng sử dụng khu vực này để nhận dạng vật thể. Điểm khác biệt chính ở đây với Fast R-CNN là sử dụng selective search để tạo khu vực đề nghị. Thời gian thiết lập ngắn hơn là selective search khi mà RPN chia sẻ phần lớn bước tính toán với mạng nhận diện vật thể. RPN xếp hạng khu vực khung (gọi là anchors) và đề nghị một cái chứa vật thể. Anchor đóng vai trò quan trọng trong Faster R-CNN. Anchors trong khung, trong cấu hình mặc định của Faster R-CNN có cái anchor

Ngõ ra của RPN là một nhóm các khung, sẽ được kiểm tra, phân loại, được hồi quy để tìm các vật thể chính xác. Để chính xác hơn, PRN sẽ đự đoán, ảnh nền và ảnh trước và lọc anchor.

Phân loại background và foreground: Đầu tiên là lấy bộ dữ liệu, bộ dữ liệu training là những anchors lấy từ tiến trình trước đó và những vùng thực địa. Vấn đề cần giải quyết là sử dụng khung thực địa để gán nhãn các anchors. Ý tưởng cơ bản là gán nhãn những anchor vùng chồng chéo cao hơn là foreground, thấp hơn là background. Câu hỏi

thứ 2 ở đây là đặc tính của các anchor là gì Mỗi vị trí trong bản đồ đặc tính có 9 anchors, mỗi anchor có 2 nhãn (background và foreground). Nếu bản đồ đặc tính có 18 anchors và đưa vào hàm hồi quy logic, sẽ dự đoán được nhãn.

Lược bỏ ROI

Sau khi có được vùng đề xuất với nhiều kích thước khác nhau, kích thước khác nhau gây khó khăn cho việc xử lý. Do đó cần phải đưa các đặc trưng về cùng một kích thước. Lược bỏ vùng quan tâm điều chỉnh kích thước các điểm đặc trưng về một kích thước chung, đồng thời ngõ ra cũng theo kích cỡ của ngõ vào.

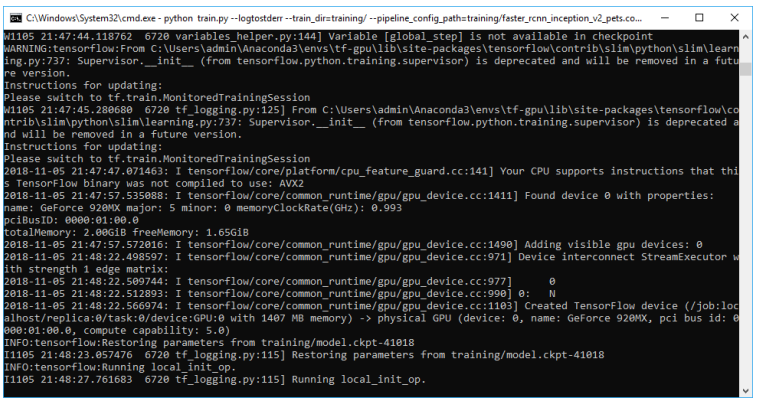
Cuối cũng sẽ đưa các ngõ ra vào các mô hình phân loại hoặc hồi quy.

3.3 Training

Quá trình huấn luyện sử dụng mô hình nhận dạng vật thể được tiến hành với các file ngõ vào quy định về kích thước, dung lượng, số lượng nhận dạng, số lượng đối tượng sử dụng trong training, các tỉ số học, . . .

3.3.1 Khởi tạo tài nguyên

Chương trình training được khởi chạy trên môi trường đã được cài đặt, chương trình sẽ, khởi động lần lượt các thành phần phần cứng liên quan. Lúc này, CPU và GPU sẽ làm việc, nhân cuda trong GPU sẽ hoạt động trong suốt quá trình training.

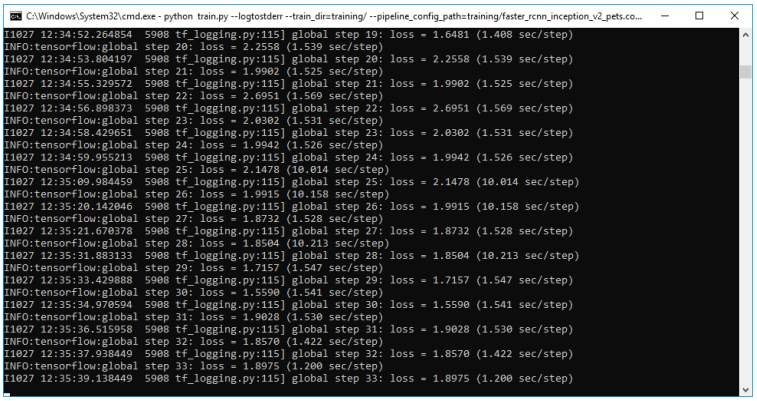


Hình 3.5: Khởi động tiến trình training

Thông tin về GPU sẽ được hiển thị chi tiết trong quá trình khởi chạy tiến trình này gồm tên, dung lượng dữ liệu còn trống cho quá trình. Trong đề tài, sử dụng GPU Gerforce 920MX, và dung lượng trống 1.6 GiB/2 GiB.

3.3.2 Thông số huấn luyện

Sau khi hoàn thành việc khởi tạo các tài nguyên phục vụ cho quá trình training. Từng bước (step) sẽ được hiển thì khi hoàn thành, ngoài ra còn có thông tin về mất mát (loss), thời gian hoàn thành 1 bước training.



Hình 3.6: Tiến trình training

Mỗi bước training sẽ được ghi lại thông tin về sự mất mát, giá trị này sẽ giảm dần trong tiến trình huấn luyện. Trong quá trình huấn luyện này trên mô hình training Faster RCNN, giá trị loss ban đầu là 3.0 và nhanh chóng giảm xuống 0.8 ở những bước tiếp theo, giá trị chuẩn cho tiến trình này là 0.05. Khi này ta có thể kết thúc quá trình training. Giá trị loss quy định số bước trong quá trình training, thời gian training liên quan đến kích thước, dung lượng ảnh và số bước huấn luyện.Ngoài ra, phần cứng (CPU và GPU cũng ảnh hưởng rất nhiều đến thời gian training.

Giá trị mất mát

Giá trị mất mát (loss) kết quả tính toán trên việc huấn luyện và xác thực. Không như giá trị chính xác, mất mát không phải là phần trăm. Giá trị mất mát càng lớn, mô hình càng tốt.

Công thức tính giá trị mất mát được biểu diễn như sau:

$$\mathcal{L}(w) = \frac{1}{2}\sum_{i=1}^{N}(y_i - \overline{x_i}w)^2 \tag{3.1}$$

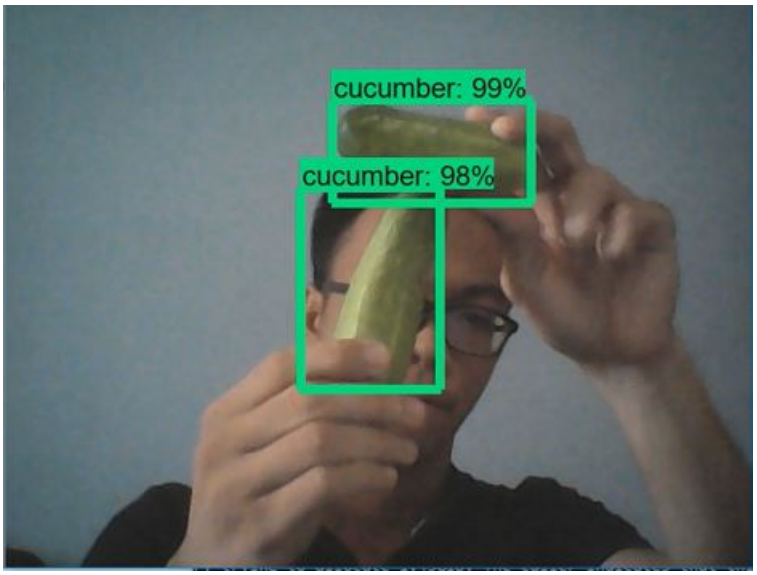
Hàm số $\mathcal{L}(w)$ được gọi là hàm mất mát (loss function). Chúng ta luôn mong muốn rằng sự mất mát (sai số) là nhỏ nhất, điều đó đồng nghĩa với việc tìm vector hệ số w sao cho giá trị của hàm mất mát này càng nhỏ càng tốt. Hàm mất mát trả về một số thực không âm thể hiện sự chênh lệch giữa hai đại lượng: $\hat{y}$, label được dự đoán và y, label đúng. Khi y là một đại lượng rời rạc chỉ nhận giá trị trong một tập label hữu hạn rời rạc nào đó.

Chương 4

KẾT QUẢ

4.1 Đánh giá kết quả

Sau quá trình training, thu được các thông tin huấn luyện trong từng bước, với số lượng đối tượng nhận dạng trong phạm vi đề tài này là 3 thì số bước thực hiện trong quá trình huấn luyện là 40000 bước, thời gian trung bình cho mỗi bước là 1.6 giây, tổng thời gian cho toàn bộ quá trình training ước tính là 17 giờ (hệ thống training dùng Intel Core i3 6th gen, và GPU Geforce 920 MX).



Hình 4.1: Giao diện nhận dạng

Số đối tượng	Số bước (step)	Thời gian (giây)	Mất mát (loss)
3	40597	1.6	0.03

Bảng 4.1: Kết quả training

Số đối tượng	Số bước (step)	Thời gian (giây)	Mất mát (loss)
10	90000	1.6	< 0.05

Bảng 4.2: Kết quả ước lượng

Ước lượng

Đối với nhận dạng 10 đối tượng thực vật thì thời gian ước lượng là 22 giờ, với số bước là 90000 (steps), vẫn duy trì trung bình 1.6 giây cho mỗi bước huấn luyện.

4.2 Hạn chế

Thời gian trong quá trình huấn luyện khá là dài (trung bình 1.6 giây) cho mỗi bước. Nguyên nhân là do kích thước ảnh và khả năng phần cứng. Vấn đề này sẽ ảnh hưởng rất lớn khi áp dụng vào hệ thống nhận diện lớn hơn, đòi hỏi cần phải có mô hình huấn luyện tốt, cũng như là tối ưu hóa cho tiến trình training.

Tốc độ khung ảnh trên giây (FPS) còn rất thấp, không thể chạy tăng lên được đối với khả năng hiện tại của phần cứng. Do đó hình ảnh trong lúc nhận dạng có hiện tượng rung, giật. Một phần nguyên nhân cũng do độ phân giải của camera trong quá trình nhận diện do sử dụng camera webcam nên không thể tránh khỏi lỗi này. Một phần nguyên nhân khác là do tính trễ trong hệ thống nhận dạng.

4.3 Hướng mở rộng đề tài

Theo xu hướng ngày nay thì các thiết bị di dộng trờ nên rất phổ biến, khi mà sự tiện lợi được đưa lên hàng đầu. Từ đó mở ra hướng phát triển của đề tài, từ điển thực vật sẽ được đưa lên thiết bị di động, giúp một phần nào đó tối ưu được khả năng tiện dụng của đề tài.

Đối với các thiết bị di động hay các hệ thống nhúng, ta dùng đến thư viện Tensorflow Lite do hạn chế về bộ nhớ thiết bị và khả năng xử lý bị hạn chế.

Mô hình sử dụng để training cũng sẽ khác đi, mô hình được cung cấp phổ biến hiện tại là Mobilenet - Inception được Google cung cấp. Hệ thống chạy trên thiết bị di động sẽ trở nên gọn nhẹ hơn, tốc độ được cải thiện hơn, do chương trình phải đảm bảo chạy được trên phần lớn các thiêt bị. Tuy nhiên sẽ hạn chế về độ chính xác cũng như là số lượng đối tượng nhận diện.Từ điển thực vật sử dụng ngay trên thiết bị di động hứa hẹn sẽ tối ưu chức năng, phát huy nhiều ưu điểm đặc trưng về tốc độ, tính chính xác và độ tiện lợi.

Chương 5

KẾT LUẬN

Với việc sử dụng máy học trong kỹ thuật xử lý ảnh vào đề tài từ điển thực vật việc tra cứu thông tin của đối tượng trở nên nhanh chóng và tiện lợi hơn, đáp ứng được nhu cầu người sử dụng trong thời đại hiện nay khi mà tốc độ được yêu cầu ngày càng gắt gao hơn.

Ngày nay, người ta vẫn tiếp tục nghiên cứu và phát triển các ứng dụng tra cứu kết hợp nhiều công nghệ hiện đại: máy học hay trí tuệ nhân tạo… Góp phần nâng tầm vai trò của khoa học công nghệ lên tầm cao mới.

Tài liệu tham khảo

[1] Sheena Angra and Sachin Ahuja. *Machine learning and its applications: a review*. 2017, pp. 57–60.

[2] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]". In: *IEEE Transactions on Neural Networks* 20.3 (2009), pp. 542–542.

[3] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python". In: *Journal of machine learning research* 12.Oct (2011), pp. 2825–2830.

[4] Rasika Phadnis, Jaya Mishra, and Shruti Bendale. "Objects Talk-Object Detection and Pattern Tracking Using TensorFlow". In: *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*. IEEE. 2018, pp. 1216–1219.