



ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN MÔN HỌC:

CS231.M21.KHCL

NHẬP MÔN THỊ GIÁC MÁY TÍNH

**NHẬN DIỆN NGÔN NGỮ KÍ HIỆU
(ALPHABET)**

Hồ Chí Minh , ngày 24 tháng 6 năm 2022

THÔNG TIN CHUNG

Đề tài:	Nhận diện ngôn ngữ kí hiệu (Alphabet)
Môn học:	CS231 – Nhập môn thị giác máy tính
Lớp:	CS231.M21.KHCL
Giảng viên hướng dẫn:	TS.Mai Tiến Dũng
Thời gian thực hiện:	Học kỳ 2 Năm học: 2021 – 2022
Sinh viên thực hiện:	Huỳnh Đình Kim Điền – 20521183 Nguyễn Hoàng Hải - 20521279
Nội dung đề tài:	Nhận diện ngôn ngữ kí hiệu cho Alphabet , chúng ta sẽ giúp máy tính hiểu được các ngôn ngữ kí hiệu của bảng chữ cái , cụ thể sẽ giúp máy tính hiểu được hình ảnh cử chỉ bàn tay thể hiện cho một kí tự chữ cái nào đó nằm trong bảng chữ cái từ A đến Z bằng cách phân loại ảnh và gán nhãn label.

<p>Kế hoạch thực hiện:</p>	<p>Tuần 1 : Hội ý , chuẩn bị đồ án (cụ thể bàn bạc về tên chủ đề , ý định làm chủ đề như thế nào)</p> <p>Tuần 2 , 3 : Tìm hiểu về chủ đề mình cần làm(cụ thể chủ đề đó là gì , những khó khăn và thách thức khi làm đề tài đó)</p> <p>Tuần 4 , 5, 6 : Tìm hiểu về tập dữ liệu dataset liên quan đến đề tài mình làm.</p> <p>Tuần 7 , 8: Tìm hiểu mô hình Machine Learning/Deeplearning có thể phù hợp với hướng bài toán mình cần giải quyết</p> <p>Tuần 9 : Thiết kế mô hình máy học / học sâu phù hợp cho việc huấn luyện và kiểm thử.</p> <p>Tuần 10: Tạo bộ dataset</p> <p>Tuần 11: Huấn luyện ,kiểm thử bộ dataset trên mô hình đã chọn</p> <p>Tuần 12: Kiểm thử thực tế bằng camera máy tính</p> <p>Tuần 13 : Tổng kết</p>
-----------------------------------	--

LỜI CẢM ƠN

Nhóm xin chân thành gửi lời cảm ơn đến TS.Mai Tiến Dũng– Giảng viên khoa Khoa học máy tính, Trường Đại học Công nghệ thông tin, Đại học Quốc gia thành phố Hồ Chí Minh, đồng thời là giảng viên giảng dạy lớp CS231.M21.KHCL -Nhập môn thị giác máy tính , trong thời gian đã tận tình hướng dẫn và định hướng cho nhóm trong suốt quá trình thực hiện và hoàn thành đồ án.

Trong quá trình thực hiện đồ án , nhóm đã cố gắng rất nhiều để hoàn thành đồ án một cách tốt nhất và hoàn thiện nhất, song cũng sẽ không tránh khỏi được những sai sót

ngoài ý muốn. Nhóm mong rằng sẽ nhận được những lời nhận xét và những lời góp ý chân thành từ quý thầy/cô và các bạn trong quá trình thực hiện đề tài của nhóm để đề tài ngày càng hoàn thiện hơn. Mọi thắc mắc cũng như mọi góp ý của mọi người xin gửi email về một trong các địa chỉ email sau: 20521183@gm.uit.edu.vn (Huỳnh Đình Kim Điền), 20521279@gm.uit.edu.vn (Nguyễn Hoàng Hải). Mỗi ý kiến đóng góp sẽ là một nguồn động lực to lớn đối với nhóm để nhóm có thể cố gắng cải tiến chương trình ngày càng hoàn thiện và phát triển đồ án lên một mức cao hơn, nhóm cũng sẽ dựa vào đó để phát triển hơn những ưu điểm và cải thiện được phần nào đó những nhược điểm của đề tài. Hy vọng đề tài “Nhận diện ngôn ngữ kí hiệu dành cho bảng chữ cái” do nhóm thực hiện sẽ trở thành một công cụ hữu ích và có thể ứng dụng được trong lĩnh vực Thị Giác Máy Tính , cũng như giúp được những người khuyết tật(câm , điếc) trong cuộc sống này.

Thành phố Hồ Chí Minh, tháng 6 năm 2022

Nhóm sinh viên thực hiện

Huỳnh Đình Kim Điền, Nguyễn Hoàng Hải

MỤC LỤC

Chương 1. TỔNG QUAN

1. Giới thiệu.....-7
2. Đặt vấn đề.....-9
3. Hướng giải quyết.....-10
4. Đối tượng và phạm vi bài toán.....-10
5. Các yêu cầu của bài toán.....-11
6. Các thách thức của bài toán.....-11

Chương 2. CƠ SỞ LÝ THUYẾT

- I. Học đặc trưng.....12
 1. Giới thiệu.....-12
 2. Hướng tiếp cận.....-12
- II. Phân loại hình ảnh.....-15
 1. Giới thiệu.....-15
 2. Hướng tiếp cận.....-16
- III. Tổng kết.....-18

Chương 3. TẬP DỮ LIỆU

1. Cơ sở xây dựng.....-19
2. Xây dựng tập dữ liệu cho bài toán.....-19

Chương 4. GIẢI QUYẾT BÀI TOÁN

1. Input và Output.....-23
2. Mô hình Convolutional Neural Network(CNN).....-23
 - 2.1 Giới thiệu về CNN.....-23
 - 2.2 Lí do lựa chọn CNN.....-24
 - 2.3 Xây dựng và đánh giá mô hình CNN.....-24
3. Framework của bài toán.....-28

Chương 5. CÀI ĐẶT THỰC NGHIỆM

1. Thiết kế chương trình cài đặt.....-29
2. Kết quả thực nghiệm.....-31

Chương 6. KẾT LUẬN

1. Bài học kinh nghiệm.....-34
2. Tài liệu tham khảo.....-35

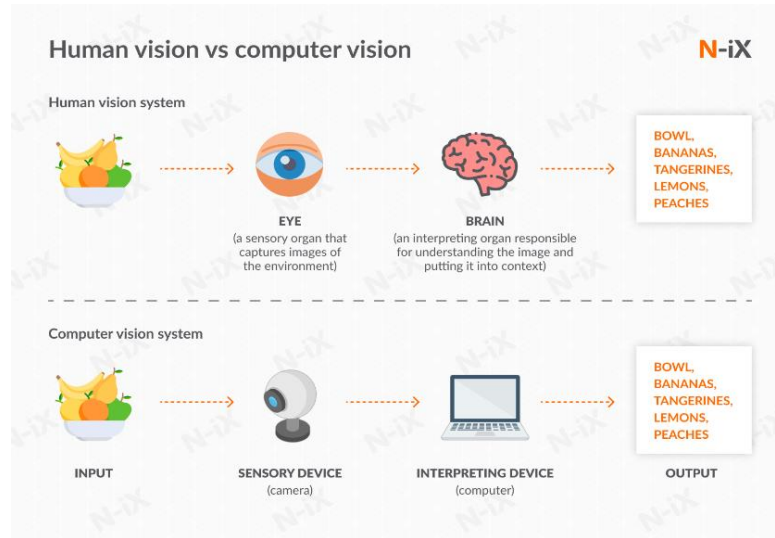
Chương 1 : TỔNG QUAN

1. Giới thiệu:

1.1. Giới thiệu về thị giác máy tính:

Thị giác máy tính là một lĩnh vực đã và đang phát triển mạnh mẽ trên thế giới. Khái niệm thị giác máy tính có liên quan tới nhiều ngành học và có nhiều hướng nghiên cứu khác nhau. Kể từ những năm 70 của thế kỉ 20 khi mà khả năng tính toán của máy tính trở nên mạnh mẽ hơn, nó có thể giải quyết được các công việc yêu cầu tốc độ cao xử lí các tệp ảnh hay đoạn video mà thị giác máy tính được nhắc đến, nghiên cứu và phát triển đến nay. Thị giác máy tính là một vấn đề rất mới mẻ đối với các nhà nghiên cứu tại Việt Nam. Việc nghiên cứu và phát triển các ứng dụng thị giác máy tính mở ra nhiều tiềm năng và khả năng ứng dụng thực tiễn mạnh mẽ cho chặng đường nghiên cứu tại Việt Nam.

Thị giác máy tính là một lĩnh vực bao gồm các phương pháp thu nhận , xử lý ảnh kỹ thuật số , phân tích và nhận dạng các hình ảnh , nói chung là dữ liệu đa chiều từ thế giới thực để cho ra các thông tin số hoặc biểu tượng. Nói một cách đơn giản hơn , thị giác máy tính nhằm tạo ra một hệ thống nhân tạo có thể tiếp nhận các hình ảnh thu được hay tập dữ liệu đa chiều để phân tích , xử lý nó theo ý muốn.



Thị giác máy tính bao gồm các lĩnh vực sau:

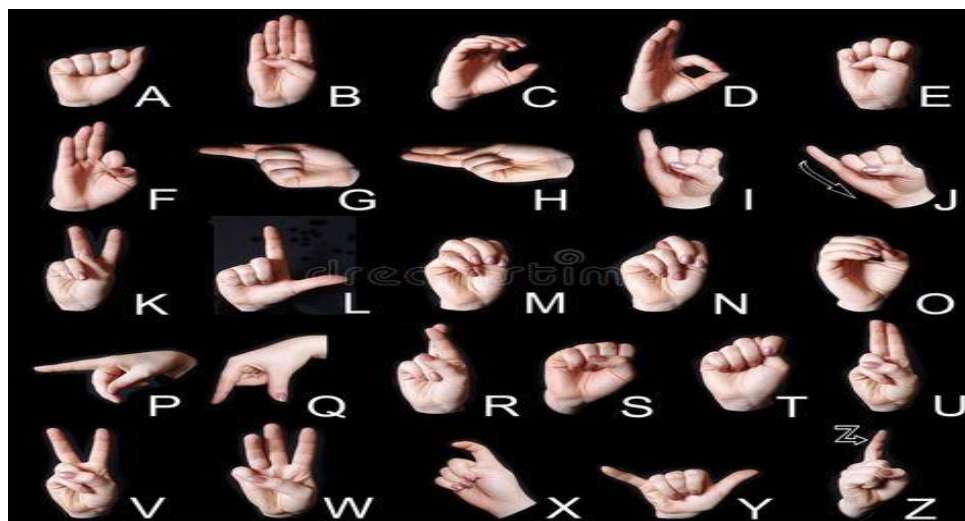
- **Xử lý hình ảnh** : Phát triển các thuật toán xử lý ảnh như tăng/giảm chất lượng ảnh , lọc nhiễu....
- **Nhận dạng mẫu** : Giải thích các kỹ thuật , mô hình khác nhau để phân loại mẫu.
- **Quang trắc** : Liên quan đến việc thu thập các số đo chính xác từ hình ảnh.

1.2. Giới thiệu về ngôn ngữ kí hiệu:

Ngôn ngữ kí hiệu là ngôn ngữ dùng những biểu hiện của bàn tay thay cho âm thanh của tiếng nói . Ngôn ngữ kí hiệu do người khuyết tật(câm , điếc) tạo ra nhằm giúp họ có thể giao tiếp với nhau trong cộng đồng của mình và tiếp thu tri thức của xã hội. Ngôn ngữ kí hiệu được nghiên cứu và phát triển từ rất lâu đời cho đến nay.

Cũng như ngôn ngữ nói, ngôn ngữ ký hiệu của từng quốc gia, thậm chí là từng khu vực trong một quốc gia rất khác nhau. Điều đó là do mỗi quốc gia, khu vực có lịch sử, văn hóa, tập quán khác nhau nên ký hiệu để biểu thị sự vật hiện tượng cũng khác nhau. Chẳng hạn, cùng chỉ tính từ màu hồng thì ở Hà Nội người ta xoa vào má (má hồng), còn tại Thành phố Hồ Chí Minh lại chỉ vào môi (môi hồng). Điều tương tự cũng diễn ra khi có sự khác biệt lớn hơn trên tầm quốc gia, dẫn tới sự khác biệt của hệ thống từ vựng và ngữ pháp ngôn ngữ ký hiệu giữa các nước. Ngôn ngữ kí hiệu có rất nhiều mảng dùng để mô tả , miêu tả nhiều khía cạnh trong đời sống như : quan hệ gia đình , quan hệ xã hội hoặc hành động trong đời sống hàng ngày,... Trong đề tài này , nhóm em chỉ nhận diện ngôn ngữ kí hiệu về mảng bảng chữ cái cho người

khuyết tật. Dưới đây là bảng chữ cái tương ứng với từng chữ cái là các cử chỉ thể hiện khác nhau.



2. Đặt vấn đề:

Hiện nay trên thế giới , tỉ lệ người khuyết tật (câm , điếc) ngày càng gia tăng mỗi năm .Không chỉ riêng ở Việt Nam mà còn ở nhiều nước khác trên thế giới đang phải đối diện với tình trạng trẻ từ 3 tuổi trở lên bị câm hoặc điếc .Điều này dẫn đến nhiều khó khăn và thách thức đối với xã hội trong việc phát triển và giáo dục con em mình. Điều này cũng khiến cho các bậc làm cha , làm mẹ phải chịu nhiều sự đau buồn khi con mình bị các bệnh bẩm sinh như vậy.Đồng thời , họ cũng gặp nhiều trở ngại trong việc nuôi dạy trẻ và đưa trẻ hoà nhập với xã hội.

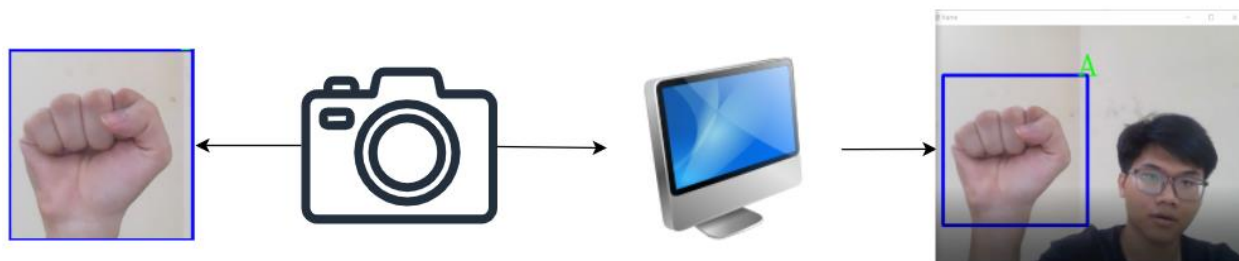
Nắm bắt và hiểu được những khó khăn trên , nhóm chúng em đã cùng nhau thực hiện đề tài “Nhận diện ngôn ngữ kí hiệu (bảng chữ cái)” với hi vọng có thể giúp ích cho cộng đồng người khuyết tật (câm , điếc) và mọi người có thể hiểu được lẫn nhau , dễ dàng trong việc giao tiếp và tiếp thu kiến thức.

Đề tài “Nhận diện ngôn ngữ kí hiệu(bảng chữ cái)” là đề tài nhận diện ngôn ngữ kí hiệu của người khuyết tật(câm , điếc) về mảng “bảng chữ cái”.



3.Hướng giải quyết:

Bài toán “Nhận diện ngôn ngữ kí hiệu về mảng bảng chữ cái” là đề tài khá hay và hữu ích . Ở đây , nhóm chúng em sẽ giới thiệu về hướng giải quyết tổng quát của bài toán .Nhận diện ngôn ngữ kí hiệu là chúng ta sẽ đi lấy hình ảnh được thu thập từ camera , sau đó chuyển hình ảnh này vào máy tính và tiếp theo , chúng ta sẽ giúp máy tính hiểu được nội dung của ảnh , từ đó đưa ra những nội dung đúng đắn về cử chỉ bản tay đó.



Quy trình thực hiện nhận diện ngôn ngữ kí hiệu tổng quát

4.Đối tượng và phạm vi nghiên cứu:

Đối tượng mà chúng ta quan tâm , đề cập trong đề tài”Nhận diện ngôn ngữ kí hiệu” bao gồm người khuyết tật(câm , điếc) và những ai có mong muốn biết đến ngôn ngữ kí hiệu dành cho người khuyết tật.

Phạm vi nghiên cứu của bài toán xoay quanh việc nhận diện cử chỉ tay từ người dùng và cho biết nội dung của cử chỉ đó tương ứng với chữ cái nào trong bảng chữ cái.Mục đích của việc nhận diện này giúp cho người khuyết tật có thể giao tiếp với mọi người thông qua hệ thống nhận diện vừa giúp người khuyết tật dễ dàng trong việc giao tiếp và giúp mọi người hiểu được điều mà ngườiếm thính họ muốn

truyền tải. Và những ai có mong muốn biết đến ngôn ngữ kí hiệu , muốn học nó và hiểu nó để dễ dàng giao tiếp với người khiếm thính đều có thể tự luyện tập thông qua hệ thống nhận diện này.

5.Yêu cầu của bài toán:

Với mục tiêu như đã nêu ở trên , đề tài này cần đặt ra những yêu cầu cần thiết , giúp cho đầu ra đúng như mong đợi.

Một số yêu cầu như sau:

- Chính xác cao
- Dễ sử dụng
- Nâng cấp và sửa chữa linh hoạt

6. Thách thức của bài toán:

Khi thực hiện đề tài này , nhóm em đã gặp không ít khó khăn và trở ngại về vấn đề dữ liệu hình ảnh thu thập từ người dùng và bộ dữ liệu ảnh huấn luyện.

6.1)Thách thức của dữ liệu ảnh thu thập từ người dùng :

Ảnh được lấy từ camera do người dùng cung cấp phụ thuộc vào nhiều yếu tố như : sự thay đổi về góc nhìn , sự chói lóa của ảnh , điều kiện ánh sáng , sự che khuất vật thể trong ảnh , sự biến dạng của vật thể trong ảnh , sự lộn xộn về background . Chính vì sự đa dạng về mặt hình ảnh như vậy , nên cũng dẫn đến nhiều khó khăn trong việc xây dựng bộ dữ liệu huấn luyện.

6.2)Thách thức của bộ dữ liệu huấn luyện:

Chính vì những khó khăn về sự đa dạng hình ảnh thu thập từ người dùng , nên cũng ảnh hưởng đáng kể đến việc xây dựng và đặt ra những quy định về bộ dữ liệu hình ảnh mà chúng ta phải xây dựng.

Nhóm đã thử làm và phát hiện ra 3 khó khăn tồn đọng ảnh hưởng đến quá trình xây dựng bộ ảnh như sau:

- Thời gian thu thập hình ảnh
- Dung lượng lưu trữ (dung lượng bộ nhớ)
- Phần cứng máy tính(card đồ họa)

Chính vì gặp nhiều khó khăn ảnh hưởng đến quá trình giải quyết bài toán ,nên nhóm chúng em đã cùng nhau bàn luận để đưa ra những quy định về mặt hình ảnh người dùng cung cấp.Vừa đảm bảo được mục tiêu , yêu cầu của bài toán và vừa có thể giải quyết bài toán một cách thuận lợi.Phần quy định ràng buộc sẽ được đề cập ở Chương Tập dữ liệu.

Chương 2. CƠ SỞ LÝ THUYẾT

I. Học đặc trưng:

1. Giới thiệu:

Trong học máy, học tính năng hoặc học đại diện là một tập hợp các kỹ thuật cho phép hệ thống tự động khám phá các biểu diễn cần thiết để phát hiện hoặc phân loại tính năng từ dữ liệu thô. Điều này thay thế kỹ thuật tính năng thủ công và cho phép một máy vừa tìm hiểu các tính năng vừa học thực hiện một tác vụ cụ thể.

2. Hướng tiếp cận:

Học đặc trưng được học thông qua các mô hình được giám sát hoặc không được giám sát:

- Trong học đặc trưng có giám sát, các đặc trưng được học bằng cách sử dụng dữ liệu đầu vào được gắn nhãn. Các ví dụ bao gồm mạng nơ-ron được giám sát, perceptron nhiều lớp và học từ điển (có giám sát).
- Trong học đặc trưng không có giám sát, các đặc trưng được học với dữ liệu đầu vào không được gắn nhãn. Các ví dụ bao gồm học từ điển, phân tích thành phần độc lập, tự động mã hóa, phân tích nhân tử ma trận và các hình thức phân cụm khác nhau.

2.1) Học giám sát:

Học đặc trưng được giám sát là học các đặc trưng từ dữ liệu được gắn nhãn. Nhãn dữ liệu cho phép hệ thống tính toán thuật ngữ lỗi, mức độ mà hệ thống không tạo ra nhãn, sau đó có thể được sử dụng làm phản hồi để sửa quá trình học (giảm / giảm thiểu lỗi). Dưới đây là các mô hình học có giám sát sử dụng kỹ thuật học đặc trưng.

Học từ điển có giám sát

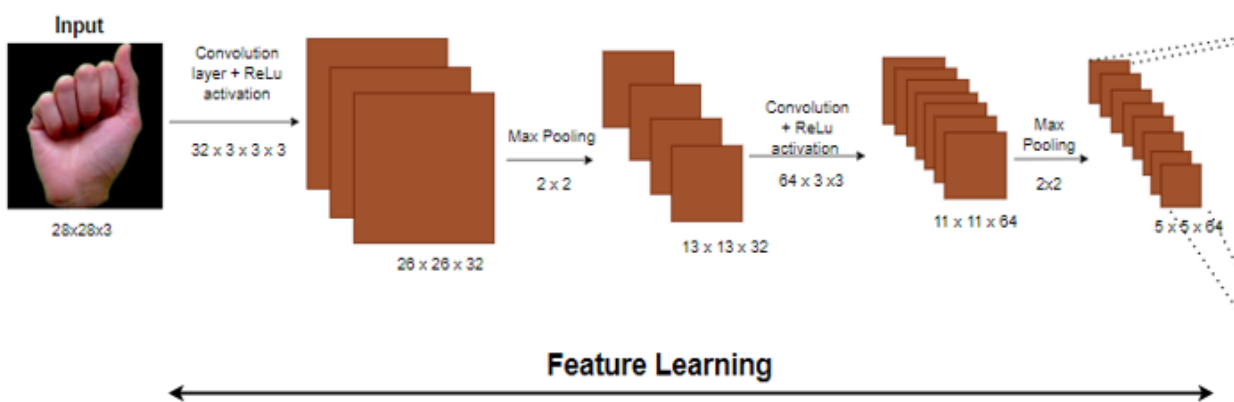
Học từ điển phát triển một tập hợp (từ điển) các phần tử đại diện từ dữ liệu đầu vào sao cho mỗi điểm dữ liệu có thể được biểu diễn dưới dạng tổng trọng số của các phần tử đại diện. Các phần tử từ điển và trọng số có thể được tìm thấy bằng cách giảm thiểu sai số biểu diễn trung bình (trên dữ liệu đầu vào), cùng với sự chính xác hóa L1 trên các trọng số để tạo ra sự thưa thớt (tức là, biểu diễn của mỗi điểm dữ liệu chỉ có một vài trọng số khác không).

Học từ điển có giám sát khai thác cả cấu trúc bên dưới dữ liệu đầu vào và nhân để tối ưu hóa các phần tử từ điển. Ví dụ, kỹ thuật học từ điển có giám sát này áp dụng việc học từ điển vào các vấn đề phân loại bằng cách cùng nhau tối ưu hóa các phần tử từ điển, trọng số để biểu diễn các điểm dữ liệu và các tham số của bộ phân loại dựa trên dữ liệu đầu vào. Cụ thể, một bài toán tối thiểu hóa được xây dựng, trong đó hàm mục tiêu bao gồm lỗi phân loại, lỗi biểu diễn, chính quy L1 trên các trọng số đại diện cho mỗi điểm dữ liệu (để cho phép biểu diễn dữ liệu thưa thớt) và chính quy L2 trên các tham số của trình phân loại.

Mạng nơ-ron

Mạng nơ-ron là một họ các thuật toán học sử dụng một "mạng" bao gồm nhiều lớp các nút được kết nối với nhau. Nó được lấy cảm hứng từ hệ thần kinh động vật, nơi các nút được xem như tế bào thần kinh và các cạnh được xem như khớp thần kinh. Mỗi cạnh có trọng số liên quan và mạng xác định các quy tắc tính toán để truyền dữ liệu đầu vào từ lớp đầu vào của mạng đến lớp đầu ra. Một chức năng mạng được liên kết với mạng nơ-ron đặc trưng cho mối quan hệ giữa các lớp đầu vào và đầu ra, được tham số hóa bởi các trọng số. Với các chức năng mạng được xác định phù hợp, các tác vụ học tập khác nhau có thể được thực hiện bằng cách giảm thiểu hàm chi phí trên chức năng mạng (trọng số).

Mạng nơ-ron nhiều lớp có thể được sử dụng để thực hiện việc học đặc trưng, vì chúng học cách biểu diễn đầu vào của chúng ở (các) lớp ẩn, sau đó được sử dụng để phân loại hoặc hồi quy ở lớp đầu ra. Kiến trúc mạng phổ biến nhất của loại này là mạng Siamese.



Quá trình học đặc trưng của mô hình CNN

2.2) Học không giám sát:

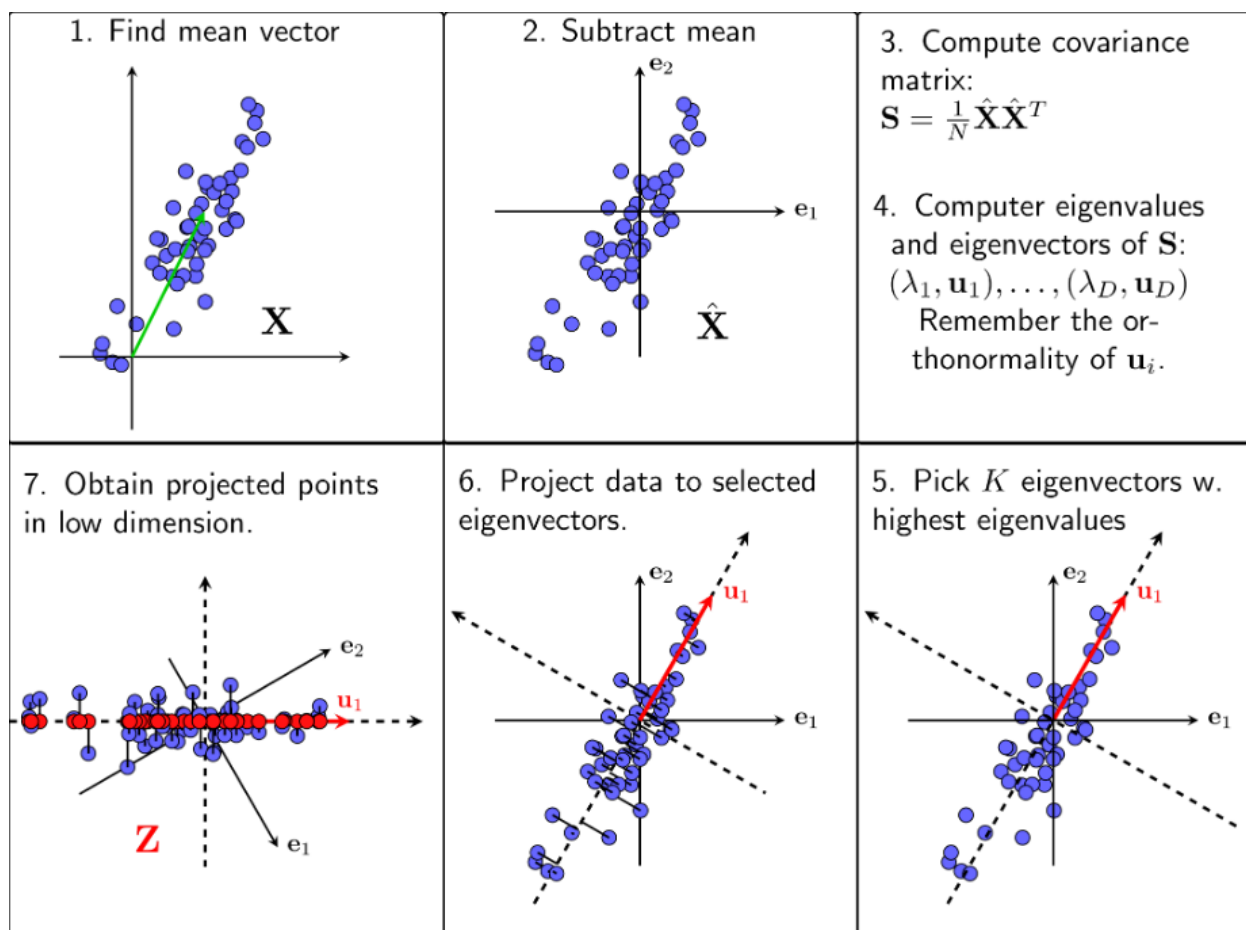
Học đặc trưng không giám sát là học các đặc trưng từ dữ liệu không được gán nhãn. Khi việc học đặc trưng được thực hiện theo cách không được giám sát, nó sẽ cho phép một hình thức học tập bán giám sát trong đó các đặc trưng được học từ tập dữ liệu chưa được gán nhãn sau đó sẽ được sử dụng để cải thiện hiệu suất trong cài đặt được giám sát với dữ liệu được gán nhãn. Dưới đây là mô hình học không giám sát có sử dụng kỹ thuật học đặc trưng:

Phân tích thành phần chính(PCA):

Phân tích thành phần chính (PCA) thường được sử dụng để giảm kích thước. Cho một tập hợp n vector dữ liệu đầu vào không được gán nhãn, PCA tạo ra p (nhỏ hơn nhiều so với kích thước của dữ liệu đầu vào) vector đơn lẻ bên phải tương ứng với giá trị kỳ dị p lớn nhất của ma trận dữ liệu, trong đó hàng thứ k của ma trận dữ liệu là vector dữ liệu đầu vào thứ k được dịch chuyển bằng giá trị trung bình mẫu của đầu vào (tức là trừ giá trị trung bình mẫu cho vector dữ liệu). Tương tự, các vector đơn lẻ này là các ký hiệu riêng tương ứng với p giá trị riêng lớn nhất của ma trận hiệp phương sai mẫu của các vector đầu vào. P vector số ít này là các vector đặc trưng được học từ dữ liệu đầu vào và chúng đại diện cho các hướng mà dữ liệu có các biến thể lớn nhất.

PCA là một phương pháp học đặc trưng tuyến tính vì các vector số ít p là các hàm tuyến tính của ma trận dữ liệu. Các vector số ít có thể được tạo ra thông qua một thuật toán đơn giản với p lần lặp. Trong lần lặp thứ i , phép chiếu của ma trận dữ liệu trên ký hiệu riêng thứ $(i-1)$ bị trừ đi và vector số một thứ i được tìm thấy là vector số ít bên phải tương ứng với số ít lớn nhất của ma trận dữ liệu dư.

PCA có một số hạn chế. Đầu tiên, nó giả định rằng các hướng có phương sai lớn được quan tâm nhất, điều này có thể không đúng. PCA chỉ dựa vào các phép biến đổi trực giao của dữ liệu gốc và nó chỉ khai thác các khoảng khắc bậc nhất và bậc hai của dữ liệu, điều này có thể không đặc trưng cho việc phân phối dữ liệu. Hơn nữa, PCA có thể giảm thứ nguyên một cách hiệu quả chỉ khi các vector dữ liệu đầu vào có tương quan với nhau (dẫn đến một vài giá trị riêng chi phối).



Quy trình học đặc trưng của PCA

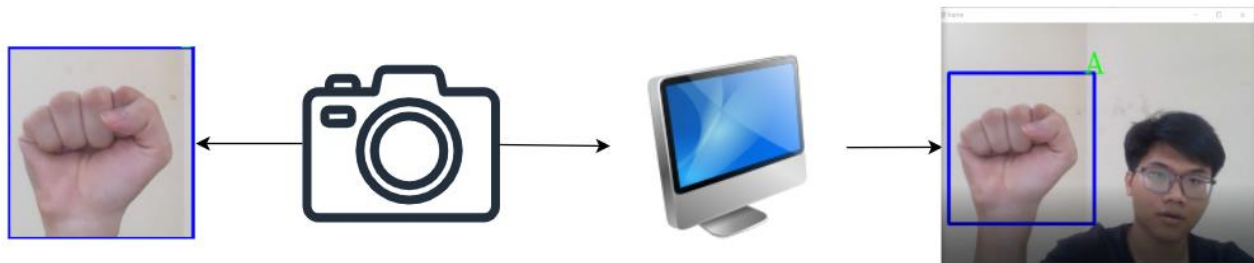
II. Phân loại hình ảnh:

1. Giới thiệu:

Phân loại hình ảnh (Image classification) hay Nhận dạng hình ảnh (Image recognition) là một trong những tác vụ của thị giác máy tính, ở đó thuật toán xem xét và dán nhãn cho hình ảnh từ một bộ dữ liệu được xác định và đào tạo trước.

Thực tế, thị giác góp phần tạo nên 80-85% nhận thức của con người về thế giới. Hàng ngày, mỗi người phải thực hiện phân loại trên bất kỳ dữ liệu hình ảnh nào mà chúng ta bắt gặp. Do đó, mô phỏng nhiệm vụ phân loại với sự trợ giúp của mạng nơ-ron là một trong những ứng dụng đầu tiên của thị giác máy tính mà các nhà nghiên cứu nghĩ đến.

Dưới đây là ảnh mô phỏng quá trình nhận dạng tổng quát:



Quy trình thực hiện nhận diện ngôn ngữ kí hiệu tổng quát

2. Hướng tiếp cận:

Có nhiều thuật toán khác nhau được ứng dụng trong việc phân loại hình ảnh. Các thuật toán này được chia thành hai nhóm chính là học có giám sát (supervised learning) và học không giám sát (unsupervised learning).

2.1) Phân loại có giám sát:

Trong học máy có giám sát, thuật toán được huấn luyện trên một tập hình ảnh đã được dán nhãn. Từ dữ liệu mẫu này, thuật toán có thể trích xuất thông tin, phục vụ phân loại ngay cả những hình ảnh chưa từng nhìn thấy trước đó.

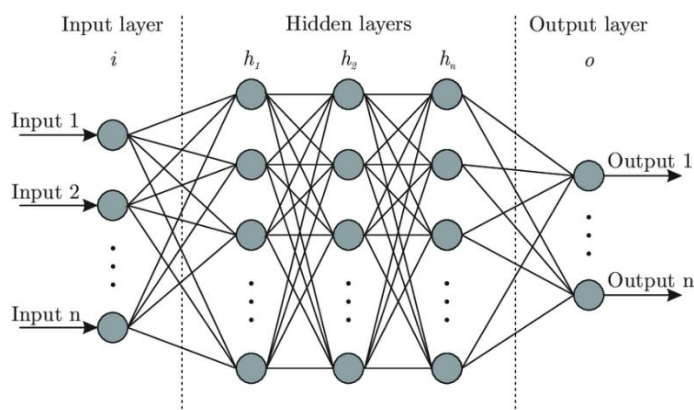
Xuyên suốt quá trình đào tạo, đặc điểm của ma trận hình ảnh sẽ được trích xuất dưới dạng dữ liệu quan trọng để đưa vào xử lý. Các đặc điểm này đại diện cho hình ảnh trong không gian chiều thấp (lower-dimensional feature space) và là cơ sở để thuật toán tiến hành phân loại.

Trong quá trình đánh giá, các đặc điểm của ảnh thử nghiệm được thu thập và tái phân loại với sự hỗ trợ của mạng thần kinh nhân tạo. Hệ thống lúc này đã có thể nhận biết các đặc điểm điển hình của mọi lớp hình ảnh mà nó được đào tạo.

Các phương pháp phân loại phổ biến dựa trên học có giám sát bao gồm:

- Support Vector Machines
- Decision Trees
- K Nearest Neighbors

Các mạng nơ-ron thường được sử dụng để phân loại hình ảnh có giám sát bao gồm AlexNet, ResNet, DenseNet và Inception.



Phân loại hình ảnh sử dụng ANN

Đối với phân loại có giám sát, việc dán nhãn dữ liệu đóng vai trò quan trọng. Độ chính xác của dữ liệu được dán nhãn quyết định phần lớn hiệu suất của mô hình học máy. Các thuật toán phân loại có giám sát có thể được chia thành hai mục nhỏ hơn dựa trên nhãn dữ liệu.

Phân loại nhãn đơn:

Phân loại nhãn đơn (Single-label classification) là tác vụ phổ biến nhất trong phân loại ảnh có giám sát. Theo đó, mỗi hình ảnh được đại diện bởi một nhãn/chú thích (a single label or annotation). Mô hình xuất ra một giá trị hoặc dự đoán duy nhất cho mỗi hình ảnh mà nó xử lý.

Đầu ra từ mô hình là mã hóa One-hot (từng giá trị được biến đổi thành các đặc trưng nhị phân chỉ chứa giá trị 1 hoặc 0). Mã hóa One-hot có độ dài bằng số lớp và giá trị biểu thị xác suất hình ảnh thuộc về lớp này.

Hàm Softmax được sử dụng để đảm bảo các xác suất tổng bằng một và xác suất tối đa được chọn làm đầu ra của mô hình. Mặc dù Softmax không có giá trị về mặt dự đoán, nhưng nó giúp ràng buộc đầu ra giữa 1 và 0, nhờ vậy, có thể đánh giá độ tin cậy của mô hình từ điểm Softmax.

Một số ví dụ về bộ dữ liệu phân loại nhãn đơn bao gồm MNIST, SVHN, ImageNet, v.v.

Phân loại nhãn đơn có thể được xếp vào phân loại đa lớp (Multiclass classification) hoặc phân loại nhị phân (binary classification).

Phân loại đa nhãn:

Phân loại đa nhãn là một tác vụ phân loại trong đó mỗi hình ảnh có thể chứa nhiều hơn một nhãn hoặc một số hình ảnh chứa đồng thời tất cả các nhãn.

Phân loại đa nhãn xuất hiện phổ biến trong lĩnh vực xử lý hình ảnh y tế, khi một bệnh nhân có thể được chẩn đoán mắc nhiều bệnh dựa trên dữ liệu chụp X-quang.

Phân loại không giám sát

Trong học máy không giám sát, thuật toán chỉ sử dụng dữ liệu thô để đào tạo. Các nhãn phân loại thường không xuất hiện trong kiểu học này và mô hình học bằng cách nhận dạng các mẫu trong tập dữ liệu huấn luyện.

Giống như phân loại có giám sát, các phương pháp dựa trên không giám sát cũng liên quan đến bước trích xuất đặc điểm với các thông tin chi tiết nhất về hình ảnh. Các đặc điểm này sau đó được xử lý bằng các phương pháp phân cụm tham số (Gaussian Mixture Models) và phi tham số (K-means for Clustering) hoặc các thuật toán học không giám sát khác.

Các thuật toán và kỹ thuật phân loại của thị giác máy tính không chỉ giới hạn trong dữ liệu hình ảnh 2D đơn giản mà còn mở rộng ra dưới dạng Video và ảnh 3D.

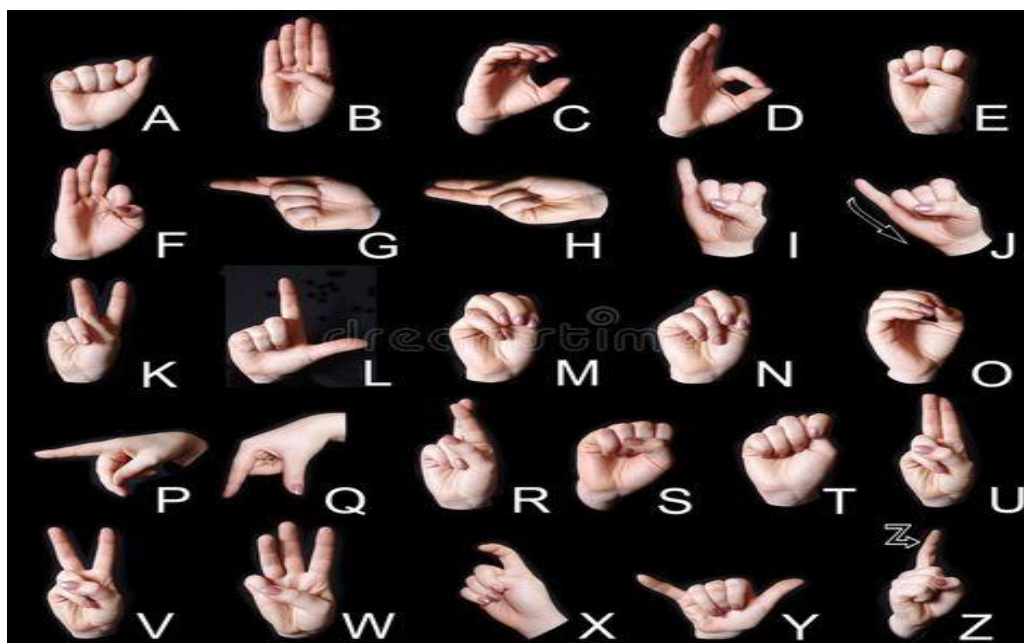
III. Tổng kết:

Với đề tài “Nhận diện ngôn ngữ kí hiệu cho bảng chữ cái”, như đã bàn luận sơ lược ở các chương trước, bộ dữ liệu chúng ta xây dựng là bộ dữ liệu đã được gán nhãn sẵn. Dựa vào các mục cơ sở lý thuyết đã bàn luận ở phía trên về Học đặc trưng và Phân loại hình ảnh, chúng ta kết luận với đề tài “Nhận diện ngôn ngữ kí hiệu cho bảng chữ cái”, sẽ được tiến hành tiếp cận Học đặc trưng và Phân loại hình ảnh theo hướng có giám sát. Điều này cũng dễ dàng chứng minh được là bởi vì bộ dữ liệu chúng ta xây dựng là bộ dữ liệu đã có nhãn và sau khi huấn luyện, chúng ta sẽ kiểm tra xem liệu mô hình của chúng ta đã dự đoán chính xác hay chưa? Và việc lựa chọn tiếp cận bài toán này theo hướng học có giám sát là hoàn toàn hợp lý vì để đảm bảo yêu cầu “Tính chính xác” của đề tài như đã được trình bày ở chương 1 mục 6. Sau khi cùng nhau tìm hiểu về cơ sở lý thuyết giúp ta tiếp cận bài toán này, chúng ta cùng đến chương tiếp theo của đề tài thú vị này nhé.

Chương 3 : TẬP DỮ LIỆU

1.Cơ sở xây dựng tập dữ liệu:

Với đề tài “Nhận diện ngôn ngữ kí hiệu cho bảng chữ cái” , như đã bàn luận ở các chương trên , chúng ta sẽ dựa trên việc học giám sát để xây dựng bộ dữ liệu gán nhãn sẵn , cùng với đó dựa trên đề tài làm về ngôn ngữ kí hiệu cho bảng chữ cái , chúng ta sẽ thu thập hình ảnh từng cử chỉ tay tương ứng với từng chữ cái khác nhau.



2.Xây dựng tập dữ liệu cho bài toán:

Ở đề tài này, nhóm chúng em sẽ chủ động triển khai thu thập dữ liệu theo 2 cách như sau:

- Bộ dữ liệu có sẵn
- Dữ liệu cá nhân hoá

2.1 Quy định về hình ảnh:

Như đã bàn luận các thách thức của đề tài này ở chương 1 mục 6 , nhằm giải quyết các khó khăn trên , nhóm em đã chủ động triển khai xây dựng một số quy định về dữ liệu hình ảnh được cung cấp bởi người dùng như sau:

- Hình ảnh phải có chứa bàn tay của người (bàn tay ở đây được tính từ phần cổ tay đến đỉnh đầu ngón tay dài nhất, bàn tay đầy đủ 5 ngón, bàn tay phải đưa rõ ràng).
- Điều kiện ánh sáng: không quá tối, không quá sáng dẫn đến ảnh sẽ bị choá sáng
- Nền ảnh: Khuyến khích người dùng cung cấp ảnh có background nền trơn, không nên cung cấp ảnh có background lộn xộn dẫn đến sự sai lệch trong việc nhận diện.

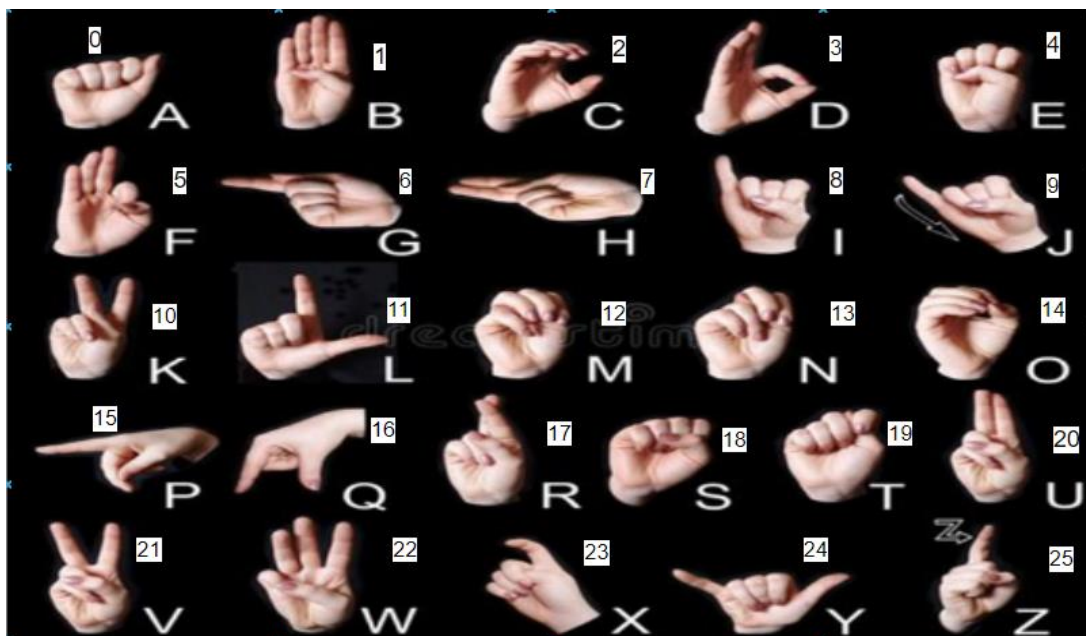
2.2 Bộ dữ liệu có sẵn:

Bộ dữ liệu có sẵn là bộ dữ liệu dạng file csv đã được chuyển từ hình ảnh cử chỉ tay sang, bộ dữ liệu này được thu thập và xây dựng bởi 1 nhóm tác giả, họ thu thập về ảnh cử chỉ tay tương ứng với từng chữ cái. Bộ dữ liệu bao gồm 2 tập dữ liệu huấn luyện và kiểm thử được tạo ra từ việc chuyển các ảnh sang ma trận ảnh số.



Minh hoạ về chuyển ảnh sang ma trận

Mỗi ma trận ảnh có 784 pixels (được chuyển từ ảnh có kích thước 28x28) thể hiện kèm với đó là label tương ứng được gán cho. Mỗi label này được quy ước theo thứ tự của từng chữ cái trong bảng chữ cái.



Link của bộ dữ liệu có sẵn: <https://pjreddie.com/projects/mnist-in-csv/>

	label	pixel1	pixel2	pixel3	pixel4
1	3	107	118	127	134
2	6	155	157	156	156
3	2	187	188	188	187
4	2	211	211	212	212
5	13	164	167	170	172
6	16	161	168	172	173
7	8	134	134	135	135
8	22	114	42	74	99
9	3	169	174	176	180
10	3	189	189	189	190
11	18	133	135	141	146
12	10	0	25	38	40
13	16	87	91	99	116
14	22	80	98	121	39
15	20	127	127	128	130
16	16	86	87	89	93
17	17	118	120	128	135
18	13	223	225	226	227
19	13	189	193	195	197
20	19	173	174	176	177
21	18	149	150	150	150

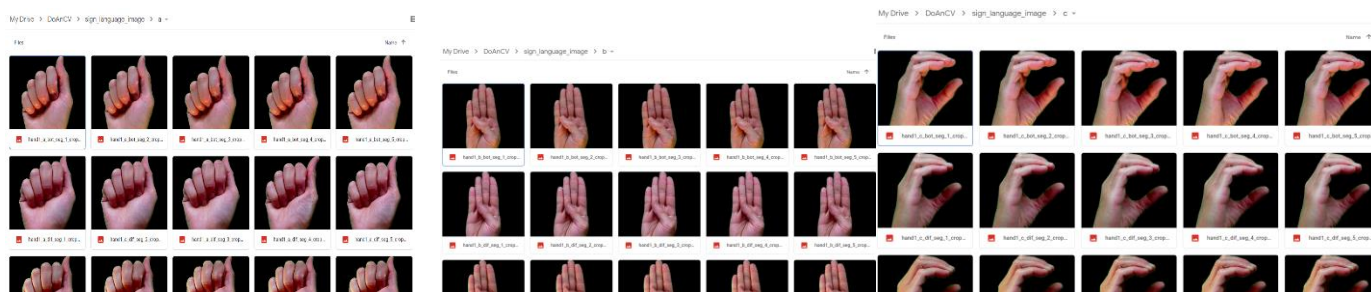
Minh hoạ ảnh bằng ma trận ảnh số từ bộ dữ liệu có sẵn.



Kết luận , với những quy định về mặt hình ảnh được nêu ở trên , thì việc lựa chọn bộ dữ liệu này cho việc huấn luyện mô hình và kiểm thử là hoàn toàn hợp lí.

2.2 Dữ liệu cá nhân hoá:

Dữ liệu cá nhân hoá là bộ dữ liệu sẽ được thu thập , lấy trực tiếp từ người dùng. Mỗi người dùng sẽ tự cung cấp ảnh , hệ thống từ đó sẽ tự tổng hợp lại và huấn luyện , kiểm thử dựa trên bộ dữ liệu được cung cấp .Như vậy , hệ thống sẽ sử dụng bộ dữ liệu được cung cấp từ mỗi người.Điều này , sẽ có mặt lợi về nhận dạng , giúp đảm bảo được yêu cầu đề ra ở chương 1 mục 5.Người dùng khi cung cấp dữ liệu cho hệ thống cũng phải tuân thủ về quy định hình ảnh đã được nêu ở trên.



Hình ảnh người dùng cung cấp ,được thu thập và lưu trữ trên Drive

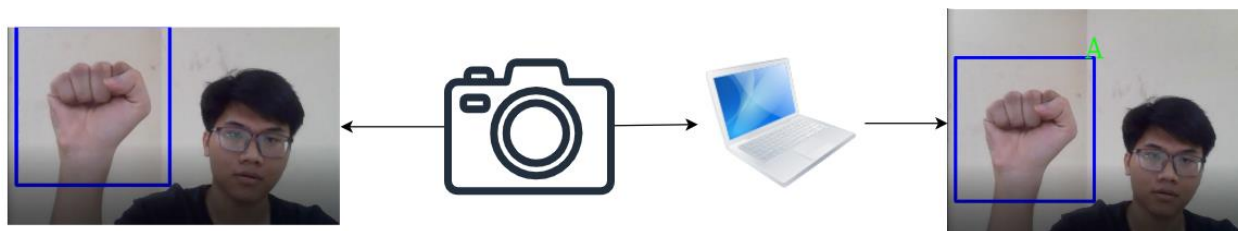
Chương 4 : GIẢI QUYẾT BÀI TOÁN

1.Input và Output của bài toán:

Đầu vào và đầu ra là một khái niệm khá quan trọng khi chúng ta thực hiện giải quyết 1 vấn đề nào đó. Có thể chưa bàn đến chuyện giải pháp của một vấn đề, nhưng để giải quyết được vấn đề, chúng ta cần định hình được đầu vào và đầu ra của bài toán chúng ta phải giải. Ở đề tài này, nhóm em xin được giới thiệu về đầu vào và đầu ra của đề tài “Nhận diện ngôn ngữ kí hiệu (bảng chữ cái)”.

Input: Hình ảnh được lấy từ camera người dùng, lưu ý ảnh phải có chứa bàn tay của người dùng.

Output: Kí tự chữ cái tương ứng với cử chỉ tay người dùng thể hiện.



2.Mô hình Convolutional Neural Network(CNN):

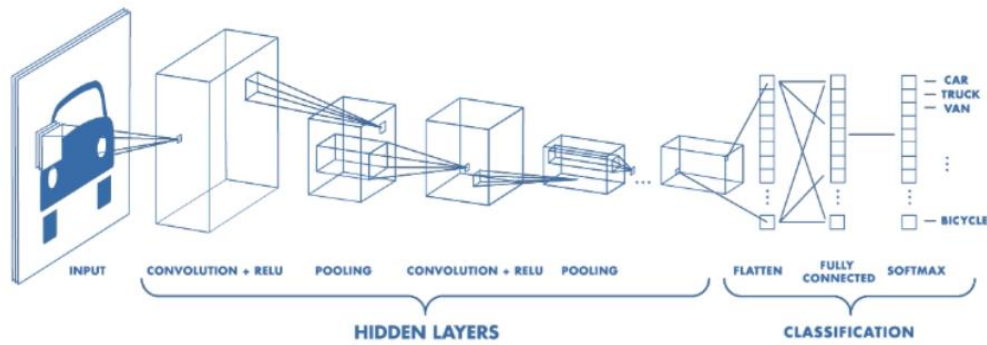
2.1 Giới thiệu về CNN:

Convolutional Neural Network là mạng nơ-ron tích chập nhiều lớp, thuộc kiểu mạng học sâu, là mô hình học có giám sát. Trong CNN, người ta phát triển nhiều kiểu mạng khác như LeNet, ResNet, AlexNet, GoogleNet,.... Các mạng này được xây dựng dựa trên cấu trúc của mạng CNN, nhằm phục vụ cho nhiều bài toán có tính chất khác nhau.

Mạng CNN gồm 2 phần chính: học đặc trưng và phân loại mẫu.

Một số lớp phổ biến thường được sử dụng trong mô hình Convolutional Neural Network:

- Convolutional layer + Activation function
- Pooling layer
- Flatten layer
- Fully connected layers(input layer, hidden layer, output layer)
- Dropout layer



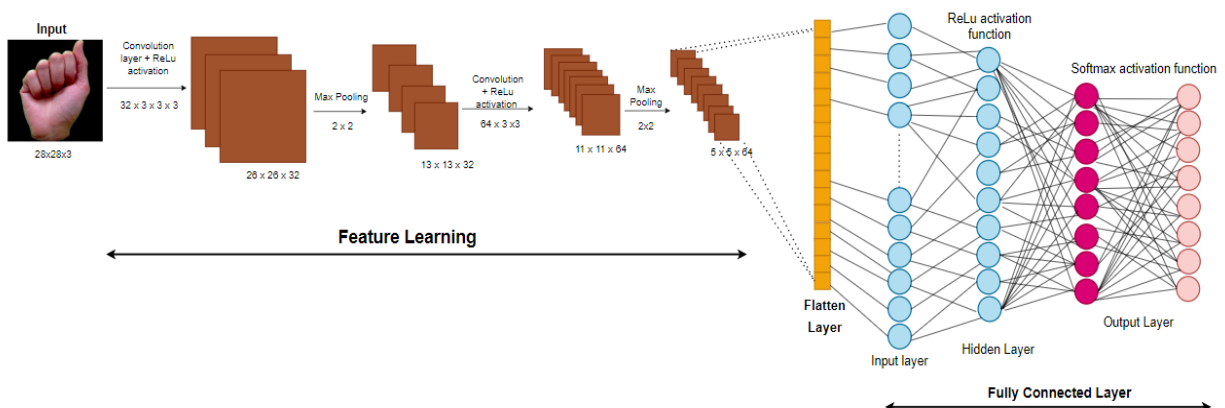
Quá trình học đặc trưng và phân loại

2.2 Lí do lựa chọn CNN:

Như đã được giới thiệu về CNN ở phía trên kèm với các cách tiếp cận bài toán theo hướng **học có giám sát** đã được bàn luận ở chương 2. Nhóm chúng em đã quyết định lựa chọn CNN là mô hình duy nhất được sử dụng trong đề tài này. Mặc dù có nhiều mô hình học máy có giám sát khác như SVM, K-nearest Neighbor, Logistic Regression đều có các phần học đặc trưng và phân loại khá tốt nhưng sau khi tìm hiểu kỹ lưỡng thì nhóm em vẫn quyết định lựa chọn CNN vì thấy nó phù hợp với các bài toán nhận diện, phân loại ảnh.

2.3 Xây dựng và đánh giá mô hình CNN:

Ở phần này, nhóm em sẽ tự xây dựng, thiết kế một mô hình CNN cho đề tài của mình (mô hình được áp dụng cho cả 2 cách thức lấy dữ liệu (dữ liệu có sẵn và dữ liệu cá nhân hoá)).



Ở đây, chắc hẳn sẽ có nhiều thắc mắc và câu hỏi về thông số của từng lớp tích chập hoặc thứ tự thực hiện từng lớp tích chập. Nhằm giải đáp các thắc mắc trên

nhóm em cũng đã cố gắng đi thực nghiệm và kiểm tra độ chính xác của mô hình tương ứng với từng thông số nhất định.

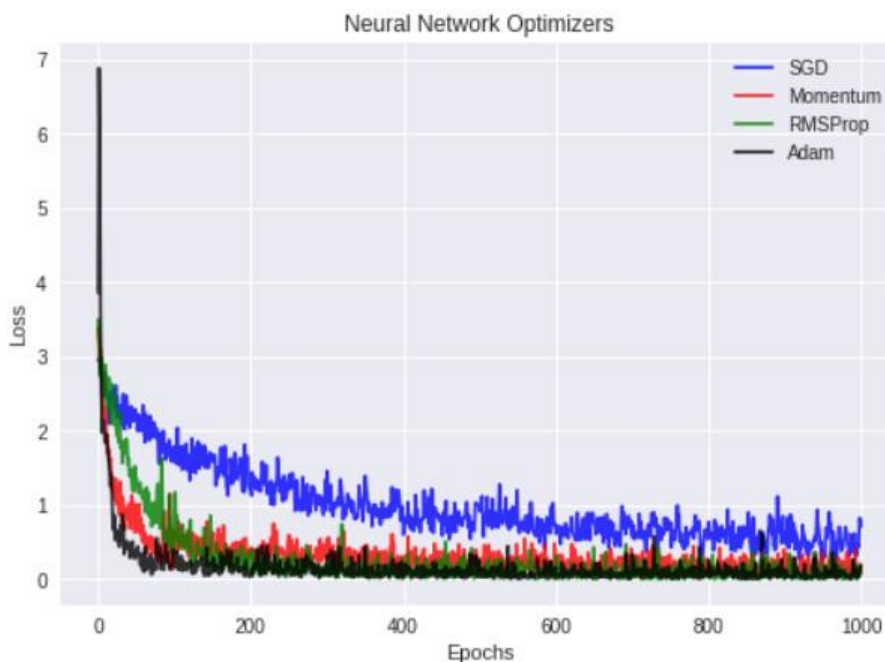
Để có thể thực hiện việc chọn lọc các thông số phù hợp , chúng ta cùng xét đến các hàm , thông số và thuật toán phục vụ cho việc đánh giá mô hình CNN này.

- Hàm mất mát : vì đề tài này chúng ta phải đi phân loại từng chữ chỉ tương ứng từng chữ cái khác nhau nên sẽ có khá nhiều label để phân loại .Vì vậy nhóm em đã sử dụng categorical crossentropy loss function , phù hợp với bài toán phân loại nhiều label với mục đích tính toán được sự mất mát của việc phân loại.

$$\text{Loss} = - \sum_{i=1}^{\text{output size}} y_i \cdot \log \hat{y}_i$$

Công thức của hàm mất mát

- Thuật toán tối ưu: nhóm em sẽ sử dụng thuật toán tối ưu Adam bởi vì , đây là thuật toán cho sự mất mát thấp nhất sau nhiều lần duyệt.Tham khảo qua hình dưới đây.



- Accuracy_score: Độ chính xác là thông số phổ biến hầu hết đều được áp dụng , phục vụ cho việc đánh giá các mô hình Machine Learning.Thông số

này sẽ cho ta biết được mức độ chính xác về việc học của mô hình trên bộ dữ liệu ta có.

Sau đây là một số kết quả thực nghiệm cho việc lựa chọn thông số ở lớp hidden layer và lựa chọn hàm activation function cho việc tính xác suất khả năng rơi vào label đó là bao nhiêu.

-Hidden layer : 128 nơ-ron , activation function: softmax

```
#Build model
model = Sequential()
model.add(Conv2D(32, kernel_size=(3,3), activation = 'relu', input_shape=(28, 28 , 3)))
model.add(MaxPooling2D(pool_size = (2,2)))

model.add(Conv2D(64, kernel_size=(3,3), activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))

#Input Layer
model.add(Flatten())
#Hidden Layer
model.add(Dense(128,activation = 'relu'))
model.add(Dropout(0.1))
#Output Layer
model.add(Dense(num_classes, activation = 'softmax'))

#get accuracy score
from sklearn.metrics import accuracy_score
accuracy_score(test_labels, y_pred.round())
```

0.925404350250976

-Hidden layer : 64 nơ-ron , activation function: softmax

```
#Build model
model = Sequential()
model.add(Conv2D(32, kernel_size=(3,3), activation = 'relu', input_shape=(28, 28 , 3)))
model.add(MaxPooling2D(pool_size = (2,2)))

model.add(Conv2D(64, kernel_size=(3,3), activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))

#Input Layer
model.add(Flatten())
#Hidden Layer
model.add(Dense(64,activation = 'relu'))
model.add(Dropout(0.1))
#Output Layer
model.add(Dense(num_classes, activation = 'softmax'))
```

```
#get accuracy score
from sklearn.metrics import accuracy_score
accuracy_score(test_labels, y_pred.round())
```

0.9138315672058004

-Hidden layer : 256 no-ron , activation function: softmax

```
#Build model
model = Sequential()
model.add(Conv2D(32, kernel_size=(3,3), activation = 'relu', input_shape=(28, 28 , 3)))
model.add(MaxPooling2D(pool_size = (2,2)))

model.add(Conv2D(64, kernel_size=(3,3), activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
```

```
#Input Layer
model.add(Flatten())
#Hidden Layer
model.add(Dense(256,activation = 'relu'))
model.add(Dropout(0.1))
#Output Layer
model.add(Dense(num_classes, activation = 'softmax'))
```

```
model.compile(loss = 'categorical_crossentropy', optimizer = Adam(), metrics = ['accuracy'])
```

```
#get accuracy score
from sklearn.metrics import accuracy_score
accuracy_score(test_labels, y_pred.round())
```

0.9223368655883993

-Hidden layer : 128 no-ron , activation function: sigmoid

```
#Build model
model = Sequential()
model.add(Conv2D(32, kernel_size=(3,3), activation = 'relu', input_shape=(28, 28 , 3)))
model.add(MaxPooling2D(pool_size = (2,2)))

model.add(Conv2D(64, kernel_size=(3,3), activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
```

```
#Input Layer
model.add(Flatten())
#Hidden Layer
model.add(Dense(128,activation = 'relu'))
model.add(Dropout(0.1))
#Output Layer
model.add(Dense(num_classes, activation = 'sigmoid'))
```

```
#get accuracy score
from sklearn.metrics import accuracy_score
accuracy_score(test_labels, y_pred.round())
```

0.3580591187953151

-Hidden layer : 64 nơ-ron , activation function: sigmoid

```
#Build model
model = Sequential()
model.add(Conv2D(32, kernel_size=(3,3), activation = 'relu', input_shape=(28, 28 , 3)))
model.add(MaxPooling2D(pool_size = (2,2)))

model.add(Conv2D(64, kernel_size=(3,3), activation = 'relu'))
model.add(MaxPooling2D(pool_size = (2,2)))

#Input Layer
model.add(Flatten())
#Hidden Layer
model.add(Dense(64,activation = 'relu'))
model.add(Dropout(0.1))
#Output Layer
model.add(Dense(num_classes, activation = 'sigmoid'))
```

```
#get accuracy score
from sklearn.metrics import accuracy_score
accuracy_score(test_labels, y_pred.round())
```

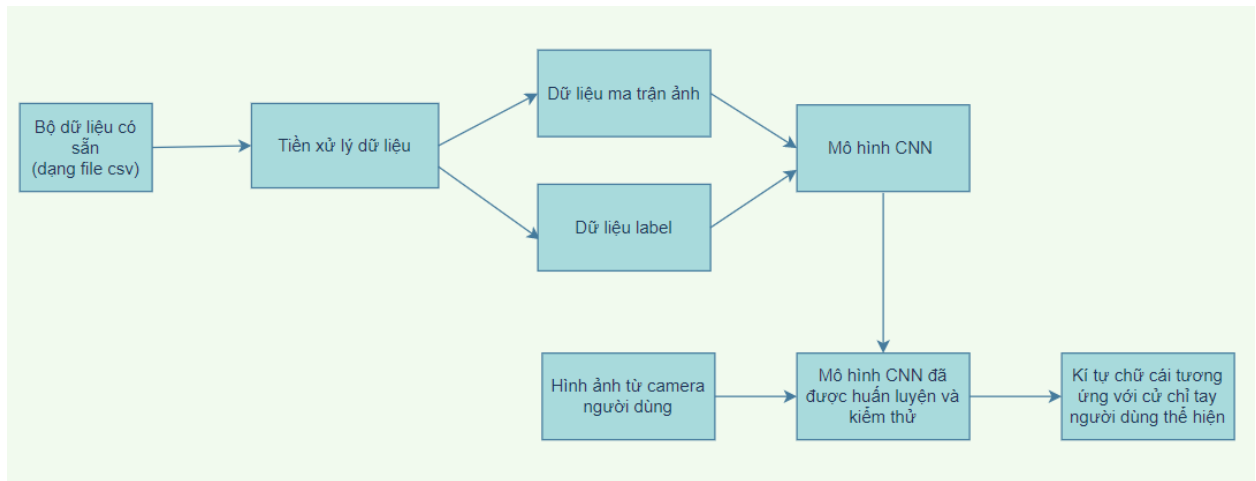
0.27732849972113777

Nhận xét:

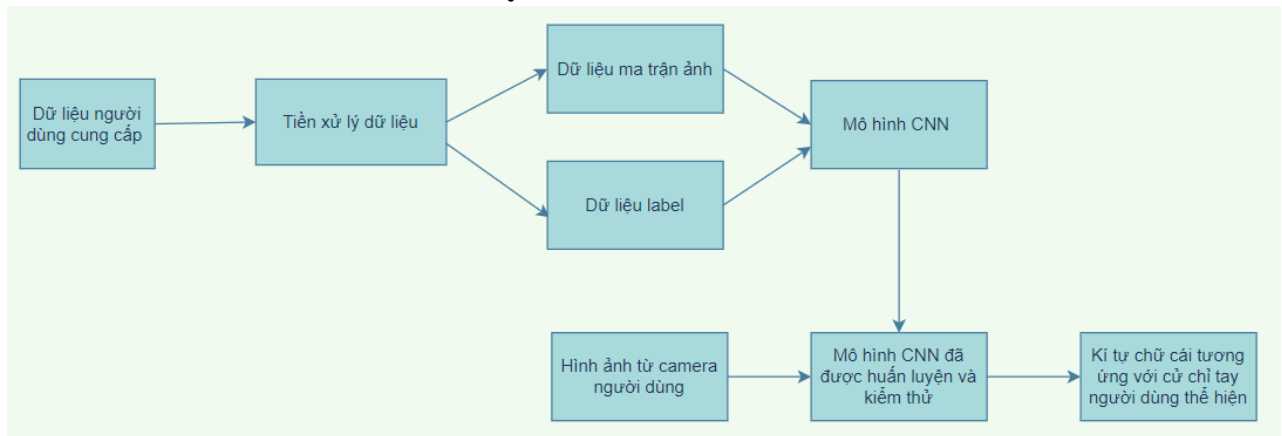
Sự thay đổi rõ rệt về độ chính xác nằm ở việc chúng ta lựa chọn hàm activation function cho mô hình , còn sự thay đổi về số nơ-ron là không đáng kể.Như vậy , nhóm em đã quyết định lựa chọn ở lớp hidden layer sẽ chứa 128 nơ-ron và hàm tính xác suất là softmax function cho mô hình CNN dùng cho đề tài này.

3. Framework của bài toán:

3.1 Framework với bộ dữ liệu có sẵn:



3.2 Framework với dữ liệu cá nhân hoá:



Chương 5. CÀI ĐẶT THỰC NGHIỆM

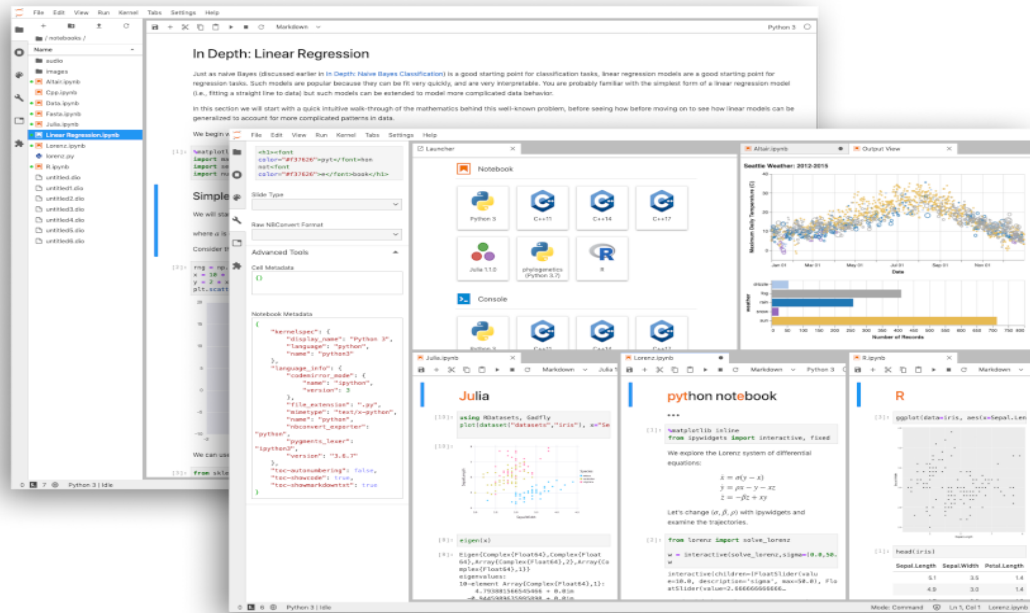
1. Thiết kế chương trình cài đặt:

Trong đề này, nhóm em sẽ tiến hành code và chạy demo trên Jupyterlab, ngoài Jupyterlab, chúng ta có thể sử dụng nhiều nền tảng khác như Googlecolab, Pycharm, Spider,... Đây là các nền tảng có hỗ trợ biên dịch ngôn ngữ Python và đề này cũng được viết bằng ngôn ngữ Python.

Giới thiệu về Jupyterlab:

JupyterLab là môi trường phát triển tương tác dựa trên web mới nhất dành cho sổ ghi chép, mã và dữ liệu. Giao diện linh hoạt của nó cho phép người dùng

định cấu hình và sắp xếp quy trình công việc trong khoa học dữ liệu, khoa học máy tính. Thiết kế mô-đun kết hợp với các tiện ích mở rộng để mở rộng và làm phong phú thêm chức năng.



Giới thiệu về thư viện được sử dụng:

- Thư viện Numpy: là một thư viện toán học phổ biến và mạnh mẽ của Python. Cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng “core Python” đơn thuần.
- Thư viện Seaborn: Seaborn là một thư viện để tạo đồ họa thống kê bằng Python. Nó được xây dựng dựa trên matplotlib và tích hợp chặt chẽ với các cấu trúc dữ liệu.
- Thư viện Matplotlib: là thư viện giúp chúng ta trực quan hoá dữ liệu sang nhiều dạng như ảnh, số liệu, hỗ trợ tối đa trong việc thống kê dữ liệu.
- Thư viện Tensorflow: là một thư viện mã nguồn mở, hỗ trợ mạnh giúp ta phát triển, triển khai các mô hình học máy, học sâu hiệu quả.
- Thư viện Open-cv: là một thư viện mã nguồn mở hàng đầu cho xử lý về thị giác máy tính, machine learning, xử lý ảnh. OpenCV được viết bằng C/C++, vì vậy có tốc độ tính toán rất nhanh, có thể sử dụng với các ứng dụng liên quan đến thời gian thực.

- Thư viện Keras: là một thư viện phần mềm mã nguồn mở cung cấp giao diện Python cho các mạng nơ-ron nhân tạo. Keras hoạt động như một giao diện cho thư viện TensorFlow.
- Thư viện Pandas: là một thư viện phần mềm được viết cho ngôn ngữ lập trình Python để thao tác và phân tích dữ liệu. Đặc biệt, nó cung cấp các cấu trúc dữ liệu và các phép toán để thao tác với các bảng số và chuỗi thời gian.
- Thư viện Skimage: là một thư viện mã nguồn mở Python, được dùng cho việc xử lý hình ảnh.
- Thư viện Python Imaging: Thư viện hình ảnh Python là một thư viện bổ sung mã nguồn mở và miễn phí cho ngôn ngữ lập trình Python, hỗ trợ thêm việc mở, thao tác và lưu nhiều định dạng tệp hình ảnh khác nhau.

Về phần code, nhóm em sẽ đính kèm trong tệp drive để thầy/cô có thể xem qua cách tụi em xây dựng viết chương trình cho đề tài “Nhận diện ngôn ngữ kí hiệu(bảng chữ cái)”.

2.Kết quả thực nghiệm:

Sau khi hoàn thành xong chương trình, nhóm em đã tiến hành thực nghiệm thời gian thực trên Jupyterlab.

2.1 Kết quả thực nghiệm với bộ dữ liệu có sẵn:

Trong tệp trên drive, nhóm em có đính kèm video “PP1” là video về kết quả thực nghiệm trên bộ dữ liệu có sẵn. Kết quả cho ta thấy được, mô hình đã học khá tốt dựa trên bộ dữ liệu có sẵn khi nhận diện được hầu hết các cử chỉ tương ứng với từng chữ cái. Tuy vậy, vẫn còn nhiều chữ cái mà chưa thể nhận diện được ngay.

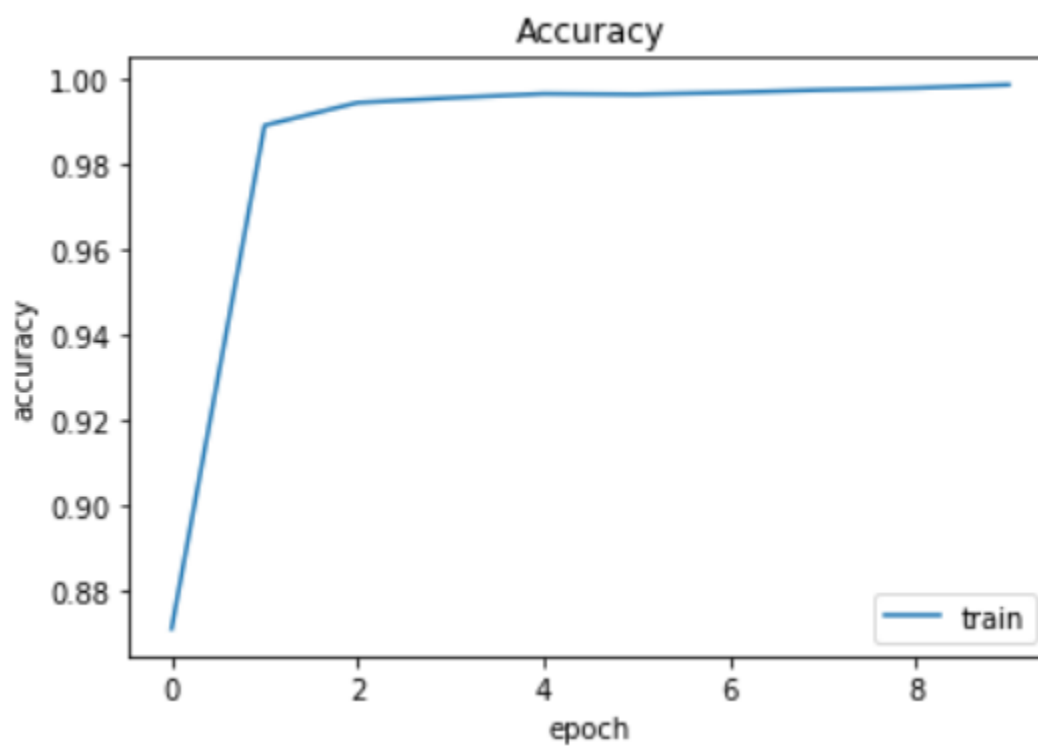
Tóm tắt mô hình CNN sau khi được huấn luyện:

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	896
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
conv2d_1 (Conv2D)	(None, 11, 11, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 5, 5, 64)	0
flatten (Flatten)	(None, 1600)	0
dense (Dense)	(None, 128)	204928
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 24)	3096

=====
 Total params: 227,416
 Trainable params: 227,416
 Non-trainable params: 0

Biểu đồ độ chính xác sau khi huấn luyện CNN trên tập data_train:



Độ chính xác của mô hình trên tập data_test:


```
#get accuracy score
from sklearn.metrics import accuracy_score
accuracy_score(test_labels, y_pred.round())
```

0.9167596207473508

Đó là tất cả kết quả thực nghiệm trên tập dữ liệu(train , test) với 25 cử chỉ tương ứng với 25 chữ cái.

2.2 Kết quả thực nghiệm với dữ liệu cá nhân hoá:

Trong tệp trên drive , nhóm em có đính kèm video “PP2” là video về kết quả thực với dữ liệu cá nhân hoá.Kết quả cho ta thấy được , mô hình đã học rất tốt dựa trên bộ dữ liệu lấy trực tiếp từ mỗi người dùng khi nhận diện được tất cả cử chỉ tương ứng với từng chữ cái mà người dùng đưa vào.

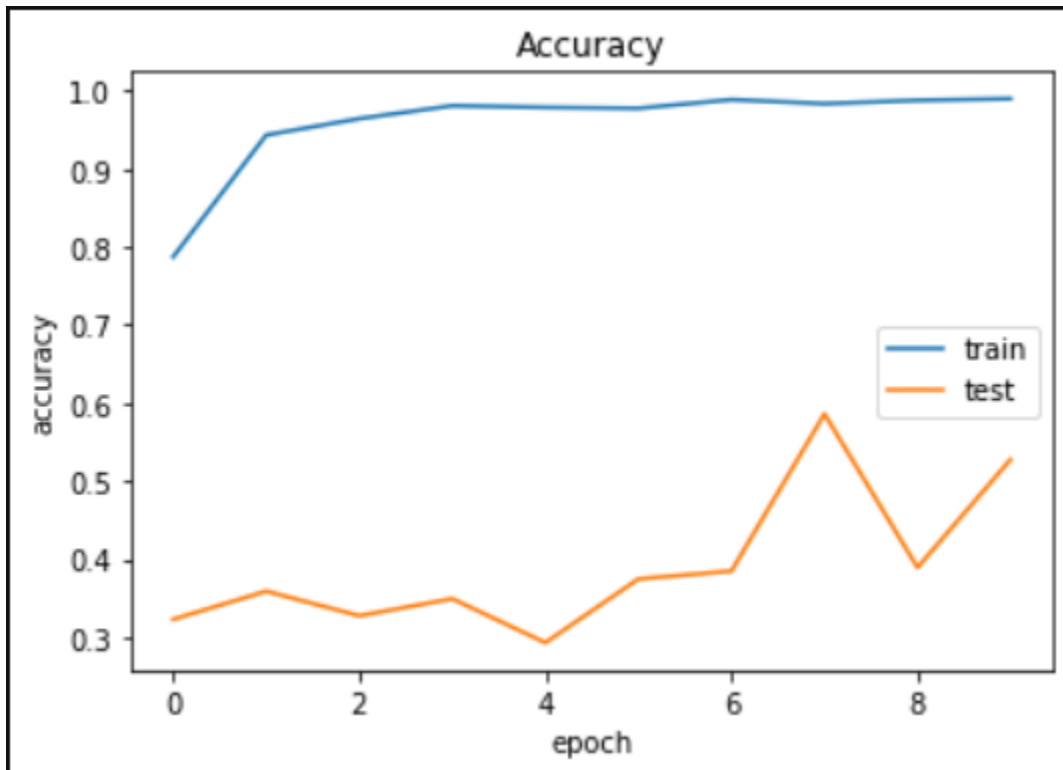
Tóm tắt mô hình CNN sau khi được huấn luyện:

Model: "sequential_4"

Layer (type)	Output Shape	Param #
conv2d_11 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_11 (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_12 (Conv2D)	(None, 29, 29, 64)	18496
max_pooling2d_12 (MaxPooling2D)	(None, 14, 14, 64)	0
flatten_4 (Flatten)	(None, 12544)	0
dense_8 (Dense)	(None, 128)	1605760
dropout_4 (Dropout)	(None, 128)	0
dense_9 (Dense)	(None, 5)	645

=====
Total params: 1,625,797
Trainable params: 1,625,797
Non-trainable params: 0

Biểu đồ độ chính xác của mô hình trên tập dữ liệu train và test:



Đó là tất cả kết quả thực nghiệm trên tập dữ liệu (train , test) với 5 chữ cái. Độ chính xác có thể thay đổi nếu chúng ta cung cấp số lượng cử chỉ đầu vào khác nhau.

Chương 6 : BÀI HỌC KINH NGHIỆM

Sau khi hoàn thành xong đề tài này, các thành viên trong nhóm chúng em đã đều có khả năng trả lời các câu hỏi liên quan tới Ảnh số là gì , các bước Xử lý ảnh cơ bản như thế nào , Phân loại ảnh là gì và đặc biệt hiểu thêm về mô hình Convolutional Neural Network. Đây cũng là đề án chuyên ngành đầu tiên của nhóm tụi em ,là thách thức đầu tiên trong lĩnh vực Thị giác máy tính .Dù khi thực hiện đề án , nhóm tụi em còn gặp nhiều khó khăn về dữ liệu và mô hình , đặc biệt là tập dữ liệu , nhóm tụi em đã gặp nhiều khó khăn trong việc thu thập dữ liệu sao cho phù hợp với đề tài , khó khăn trong việc đưa ra những quy định về dữ liệu hình ảnh .Nhưng nhờ có sự hỗ trợ từ thầy/cô và đọc thêm nhiều tài liệu tham khảo , nhóm tụi em cũng đã rút ra được một số kinh nghiệm quý báu khi xây dựng bộ dữ liệu . Về phần lựa chọn mô hình , nhóm em đã mắc lỗi vì chưa kịp thử nghiệm bộ dữ liệu trên nhiều mô hình học giám sát nhất có thể .Đây là bài học mà tụi em cần rút ra cho các lần làm đề án sau này.Sau khi làm hoàn thành xong đề tài này , nhóm tụi em đã rút ra một câu hỏi cần nên được đặt ra khi bước vào làm một đề tài nào đó “Ta cần phải hiểu được đề tài của chúng ta phân dữ liệu và mô hình cái nào quan trọng

hơn và cần tập trung vào phần nào để làm cho đề tài được tốt hơn?”. Và đặc biệt là lời nhận xét của thầy/cô, khiến em ghi nhớ mãi “Cái quan trọng không phải là độ chính xác bao nhiêu, mà là tụi em phải hiểu được tại sao lại có được độ chính xác như vậy, tại sao lại thấp, tại sao lại cao?”. Hi vọng sau khi thực hiện đề tài này, nhóm tụi em sẽ tích lũy được nhiều kinh nghiệm quý báu, nâng cao kỹ năng làm việc nhóm, kỹ năng thuyết trình, kỹ năng bảo vệ quan điểm của mình.

Cuối cùng, nhóm em xin chân thành gửi lời cảm ơn đến thầy/cô đã luôn nhiệt tình hỗ trợ, giúp đỡ trong suốt quá trình thực hiện đồ án này.

TÀI LIỆU THAM KHẢO

- [1] <https://www.youtube.com/watch?v=NQPV2344cGE>
- [2] https://www.youtube.com/watch?v=6Bn0PY_ouBY
- [3] https://www.youtube.com/watch?v=V0Pk_dPU2IY
- [4] <https://towardsdatascience.com/understanding-cnn-convolutional-neural-network-69fd626ee7d4>
- [5] https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_k%C3%BD_hi%E1%BB%87u
- [6] Dropout: A Simple Way to Prevent Neural Networks from Overfitting by Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, Department of Computer Science, University of Toronto, Journal of Machine Learning Research 15 (2014) 1929-1958

