**Danh sách từ viết tắt**

**DCN** ………… ….. Mạng tích chập sâu

**DCN** …….…………Mạng tích chập sâu

**ASL** …………… ….. Ngôn ngữ ký hiệu của Mỹ

**HCI** ……………… Tương tác giữa người và máy tính

**ONEIOS** ……… Hệ điều hành Robot thông minh thần kinh-điện tử hoàn thiện mở

**RNN** …………… ….. Mạng lưới thần kinh tái phát

**Mục lục**

[CHƯƠNG I](#_Toc133053795)  [1](#_Toc133053795)

[GIỚI THIỆU](#_Toc133053796)  [1](#_Toc133053796)

[Bối cảnh](#_Toc133053797)  [1](#_Toc133053797)

[Mục tiêu](#_Toc133053798)  [2](#_Toc133053798)

[Ứng dụng](#_Toc133053799)  [2](#_Toc133053799)

[1.5.](#_Toc133053800)  [Phân tích khả thi](#_Toc133053800)  [3](#_Toc133053800)

[1.5.1.](#_Toc133053801)  [Tính khả thi về kinh tế](#_Toc133053801)  [3](#_Toc133053801)

[1.5.2.](#_Toc133053802)  [Tính khả thi kỹ thuật](#_Toc133053802)  [3](#_Toc133053802)

[1.5.3.](#_Toc133053803)  [Tính khả thi hoạt động](#_Toc133053803)  [3](#_Toc133053803)

[1.5.4.](#_Toc133053804)  [Tiến độ dự án](#_Toc133053804)  [3](#_Toc133053804)

[1.6.](#_Toc133053805)  [Yêu cầu hệ thống](#_Toc133053805)  [4](#_Toc133053805)

[1.7.](#_Toc133053808)  [Công nghệ được sử dụng](#_Toc133053808)  [4](#_Toc133053808)

[CHƯƠNG II](#_Toc133053809)  [11](#_Toc133053809)

[BÌNH LUẬN VĂN HỌC](#_Toc133053810)  [11](#_Toc133053810)

[2.1.](#_Toc133053812)  [Công trình liên quan](#_Toc133053812)  [11](#_Toc133053812)

[CHƯƠNG III](#_Toc133053813)  [14](#_Toc133053813)

[THIẾT KẾ VÀ KIẾN TRÚC HỆ THỐNG](#_Toc133053814)  [14](#_Toc133053814)

[3.1.](#_Toc133053815)  [Sơ đồ khối](#_Toc133053815)  [14](#_Toc133053815)

[3.2.](#_Toc133053816)  [Sơ đồ ca sử dụng](#_Toc133053816)  [15](#_Toc133053816)

[3.3.](#_Toc133053817)  [Sơ đồ luồng dữ liệu](#_Toc133053817)  [16](#_Toc133053817)

[3.4.](#_Toc133053818)  [Sơ đồ tuần tự](#_Toc133053818)  [17](#_Toc133053818)

[CHƯƠNG IV](#_Toc133053819)  [18](#_Toc133053819)

[PHƯƠNG PHÁP](#_Toc133053820)  [18](#_Toc133053820)

[4.1.](#_Toc133053821)  [Bộ sưu tập dữ liệu](#_Toc133053821)  [18](#_Toc133053821)

[4.2.](#_Toc133053822)  [Thuật toán được sử dụng](#_Toc133053822)  [19](#_Toc133053822)

[4.3.](#_Toc133053823)  [Kiểm tra và xác minh](#_Toc133053823)  [22](#_Toc133053823)

[CHƯƠNG V](#_Toc133053824)  [23](#_Toc133053824)

[KẾT QUẢ VÀ THẢO LUẬN](#_Toc133053825)  [23](#_Toc133053825)

[5.1.](#_Toc133053826)  [Đầu ra](#_Toc133053826)  [23](#_Toc133053826)

[5.2.](#_Toc133053827)  [Công việc đã hoàn thành](#_Toc133053827)  [26](#_Toc133053827)

[5.3.](#_Toc133053828)  [Hạn chế](#_Toc133053828)  [26](#_Toc133053828)

[5.4.](#_Toc133053829)  [Vấn đề phải đối mặt](#_Toc133053829)  [26](#_Toc133053829)

[CHƯƠNG VI](#_Toc133053830)  [27](#_Toc133053830)

[KẾT LUẬN VÀ NHỮNG CẢI TIẾN TRONG TƯƠNG LAI](#_Toc133053831)  [27](#_Toc133053831)

[6.1.](#_Toc133053833)  [Kết luận](#_Toc133053833)  [27](#_Toc133053833)

[6.2.](#_Toc133053834)  [Những cải tiến trong tương lai](#_Toc133053834)  [27](#_Toc133053834)

[CHƯƠNG VII](#_Toc133053835)  [28](#_Toc133053835)

[TÀI LIỆU THAM](#_Toc133053836)  [KHẢO 28](#_Toc133053836)

# CHƯƠNG I

# GIỚI THIỆU

## Lý lịch

Thế giới không thể tồn tại nếu không có sự tương ứng, cho dù nó có xuất hiện dưới dạng liên hệ, diễn ngôn hay khớp nối trực quan hay không. Văn bản và cách phát âm hình ảnh có tác dụng tương ứng giữa người khó nghe và người yên tĩnh. Bàn tay và điểm nhấn trên khuôn mặt đặc biệt quan trọng trong việc đưa ra quan điểm của con người trong thư từ bí mật. Nhiều nâng cấp cơ khí và nhiều cuộc kiểm tra đã nhằm mục đích giúp đỡ những người không quá nhạy bén. Khả năng học tập sâu sắc và tầm nhìn PC cũng có thể được sử dụng để tác động đến lý do này.

Nếu một cá nhân không thể nói hoặc nghe, giao tiếp dựa trên cử chỉ là phương pháp chính để họ tiếp cận thư từ. Đánh vần bằng ngón tay là một công cụ quan trọng trong ngôn ngữ ký hiệu vì nó cho phép giao tiếp bằng tên, địa chỉ và các từ khác không mang ý nghĩa ở cấp độ liên kết từ [1] .

Giấy phép giao tiếp dựa trên cử chỉ đã kích thích mọi người đưa ra quan điểm và tình cảm của họ. Trong bài báo này, một chiến lược chứng minh khả năng nhận biết giao tiếp dựa trên cử chỉ đặc biệt nhằm phân biệt các bộ chữ cái và chuyển động trong giao tiếp thông qua ký hiệu được đề xuất.

Vấn đề chúng tôi đang điều tra là nhận dạng ngôn ngữ ký hiệu thông qua việc học tính năng không giám sát. Nhiều hệ thống được phát triển để nhận dạng ngôn ngữ ký hiệu và không có hệ thống chính xác để nhận dạng các dấu hiệu hoàn chỉnh [2]. Khả năng nhận dạng ngôn ngữ ký hiệu là một vấn đề thú vị về thị giác máy tính đồng thời cực kỳ hữu ích cho người khiếm thính khi tương tác với những người không biết cách hiểu Ngôn ngữ ký hiệu của Mỹ (ASL) [3].

* 1. **Báo cáo vấn đề**

Có nhiều người câm và người điếc có hình thức thể hiện cảm xúc của mình thông qua các dấu hiệu. Người bình thường rất khó hiểu chính xác nội dung biểu đạt tượng trưng của những người này. Đó là một công việc đầy thách thức và đã tạo ra rào cản giao tiếp trong cuộc sống thực. Công nghệ phát triển rất nhanh và đáng kinh ngạc, tuy nhiên vẫn chưa có nhiều sự phát triển và cải tiến công nghệ dành cho người câm điếc. Vì vậy, mục đích của dự án của chúng tôi là nhằm ngăn chặn những quan niệm sai lầm và tăng cường giao tiếp và hòa hợp giữa các loại người khác nhau

## Khách quan

Mục tiêu của dự án của chúng tôi là phân tích các cử chỉ ký hiệu khác nhau nhằm xóa bỏ rào cản giao tiếp giữa người câm điếc và người bình thường.

## Các ứng dụng

Khả năng hiểu cử chỉ tay của máy tính hoặc máy móc là chìa khóa để mở ra vô số ứng dụng tiềm năng. Các miền ứng dụng tiềm năng của hệ thống nhận dạng cử chỉ như sau:

1. Nhận dạng ngôn ngữ ký hiệu—Phương tiện giao tiếp dành cho người điếc. Nó bao gồm một số loại: đánh vần bằng ngón tay, các từ riêng biệt, từ vựng của từ và các dấu hiệu liên tục.
2. Robot và Tele-robot—Các bộ truyền động và chuyển động của cánh tay, chân robot và các bộ phận khác của robot có thể được di chuyển bằng cách mô phỏng hành động của con người.
3. Trò chơi và thực tế ảo—Thực tế ảo cho phép tương tác thực tế giữa người dùng và môi trường ảo. Nó mô phỏng chuyển động của người dùng và chuyển chuyển động đó sang thế giới 3D.
4. Tương tác giữa người và máy tính (HCI )—Bao gồm ứng dụng điều khiển cử chỉ trong lĩnh vực quân sự, y tế, thao tác đồ họa, công cụ thiết kế, chú thích hoặc chỉnh sửa tài liệu

## Phân tích tính khả thi

### Nền kinh tế khả thi

Chúng tôi có tất cả các nguồn lực sẵn có để thực hiện dự án do đó không cần thêm chi phí.

### Tính khả thi về mặt kỹ thuật

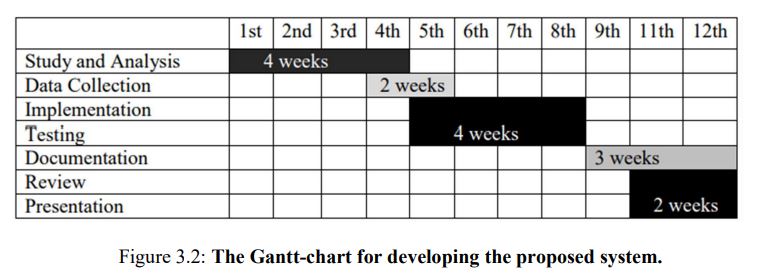
Về mặt kỹ thuật, việc phát triển các hệ thống nhận dạng ngôn ngữ ký hiệu là khả thi về mặt kỹ thuật. Một số phương pháp đã thành công trong việc đạt được kết quả tốt trong nhận dạng ngôn ngữ ký hiệu, bao gồm việc sử dụng kỹ thuật thị giác máy tính, thuật toán học máy và thiết bị đeo được.

### Tính khả thi hoạt động

Nó liên quan đến khả năng vận hành của hệ thống. Dự án này yêu cầu một máy tính có khả năng tính toán tốt. Cần ít nhất 8 Gigabyte RAM và CPU hiện đại.

### Dòng thời gian dự án

Thông thường, tính khả thi của Lịch trình là ước tính về thời gian phát triển ứng dụng của hệ thống. Dự án được lên kế hoạch khả thi nếu nó có thể được hoàn thành bằng cách sử dụng một số phương pháp như thời gian hoàn vốn.



Hình: 1.1: Biểu đồ Gannt cho quá trình hoàn thành tổng thể dự án

## yêu cầu hệ thống

1. Đặc điểm phần cứng

* 3 GB ổ đĩa miễn phí
* RAM 8GB
* CPU/GPU lõi tứ

1. Đặc tả phần mềm

* Hệ điều hành – Windows/MAC/LINUX
* Python - Để tạo mã nguồn và thuật toán

## Công nghệ được sử dụng

**Python**

Python là ngôn ngữ lập trình thông dịch cấp cao, được biết đến vì tính đơn giản, dễ đọc và linh hoạt. Nó được Guido van Rossum phát hành lần đầu tiên vào năm 1991 và từ đó trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới. Python được sử dụng cho nhiều ứng dụng, bao gồm phát triển web, phân tích dữ liệu, học máy, trí tuệ nhân tạo, tính toán khoa học, v.v. Python hỗ trợ nhiều mô hình lập trình, bao gồm lập trình hướng đối tượng, lập trình thủ tục và lập trình chức năng. Nó có một hệ thống kiểu động và quản lý bộ nhớ tự động, giúp viết và gỡ lỗi mã rất dễ dàng .

Python có thể chạy trên nhiều nền tảng, bao gồm Windows, macOS, Linux và nhiều nền tảng khác. Nó cũng có một số lượng lớn các thư viện và framework của bên thứ ba giúp bạn dễ dàng thực hiện các tác vụ phức tạp với mã hóa tối thiểu. Một số thư viện và framework phổ biến bao gồm NumPy, Pandas, Django, Flask, PyTorch , TensorFlow và nhiều thư viện khác.

**OpenCV**

OpenCV ( Thư viện thị giác máy tính nguồn mở ) là một tập hợp các khả năng lập trình được thiết kế chủ yếu cho tầm nhìn PC đang diễn ra. Lúc đầu, do Intel sản xuất, sau đó nó được Willow Garage và Itseez (được Intel mua lại) hỗ trợ. Thư viện có tính tự chủ theo từng giai đoạn và được phép sử dụng theo giấy phép nguồn mở BSD.

Các vùng ứng dụng của OpenCV bao gồm:

Ngăn chứa công cụ phần tử 2D và 3D

Đánh giá bản ngã

Khung nhận dạng khuôn mặt

Xác nhận tín hiệu

Hiệp hội con người-PC (HCI)

Công nghệ cơ khí đa dạng

Phong trào nhận được nó

Bằng chứng nhận dạng đối tượng

Phân chia và ghi nhận Tầm nhìn hệ thống âm thanh Stereopsis: cái nhìn sâu sắc từ 2 camera

Cấu trúc từ chuyển động (SFM).

Chuyển động theo sau

OpenCV chính thức được bắt đầu vào năm 1999 dưới dạng chương trình Nghiên cứu của Intel nhằm thúc đẩy các ứng dụng tập trung vào CPU, như một tính năng của tiến trình thử nghiệm bao gồm việc theo dõi chùm tia không đổi và bộ chia trình bày 3D. Nhiều chuyên gia tinh giản khác nhau từ Intel Nga và Nhóm Thư viện Hiệu suất của Intel nằm trong số những người ủng hộ thiết yếu cho liên doanh này. Trong khoảng thời gian dài đầu tiên của OpenCV, các mục tiêu của liên doanh được thể hiện như sau:

Nâng cao nghiên cứu tầm nhìn bằng cách cung cấp mã cho nền tảng tầm nhìn quan trọng không chỉ đơn giản là mở mà còn được sắp xếp hợp lý hơn. Đừng bận tâm lãng phí thời gian.

Cung cấp một cơ sở tiêu chuẩn để các nhà thiết kế mở rộng, để mã đó nhanh chóng hợp lý và dễ thích ứng hơn, từ đó truyền bá nhận thức về mục tiêu.

Nâng cao các ứng dụng kinh doanh dựa trên tầm nhìn bằng cách làm cho mã nhỏ gọn, nâng cao khả năng thực thi có thể truy cập được mà không cần hạn chế — với giấy phép không cần bận tâm đến mã nguồn để được mở hoặc miễn phí.

OpenCV được tạo bằng C++ và điểm kết nối cơ bản của nó là trong C++, tuy nhiên , nó cũng chứa một điểm kết nối C lớn nhưng ít đầy đủ hơn. Mỗi tiến trình và tính toán mới đều được phản ánh trong giao diện C++. Có các mối quan hệ Python, Java và MATLAB/OCTAVE. Bạn có thể theo dõi API cho các điểm kết nối này trong tài liệu dựa trên web. Các lớp phủ bằng nhiều phương ngữ lập trình khác nhau đã được tạo ra để phù hợp với mục đích sử dụng nổi bật hơn. Trong biến thể 3.4, các liên kết JavaScript cho một tập hợp con các khả năng OpenCV đã có thể truy cập được đối với các nền tảng web dưới dạng OpenCV.js.

**máy ảnh**

Keras dựa trên Python là một thư viện mã nguồn mở dành cho các tổ chức trí não. TensorFlow, Microsoft Cognitive Toolkit, R, Theano và PlaidML đều khả thi với nó. Tính dễ sử dụng, chất lượng được đo lường và khả năng mở rộng được tập trung vào việc thúc đẩy quá trình thử và sai nhanh chóng với các tổ chức trí não sâu sắc. Nó được tạo ra như một tính năng của liên doanh nghiên cứu ONEIROS (Hệ điều hành Robot thông minh thần kinh-điện tử hoàn thiện mở) và kỹ sư Francois Chollet của Google là người khởi xướng và bảo trì thiết yếu của nó. Chollet cũng là người thiết kế mô hình XCeption của một tổ chức trí não sâu sắc. Điểm nổi bật: Keras chứa nhiều cách thực thi khác nhau của các khối xây dựng mạng não được sử dụng thường xuyên, chẳng hạn như lớp, mục tiêu, khả năng thực thi và trình tăng cường, cũng như nhiều loại công cụ giúp làm việc với thông tin hình ảnh và văn bản dễ dàng hơn để cải thiện trên 11 mã hóa dự kiến để soạn mã mạng não sâu sắc. Mã nguồn có thể truy cập được trên GitHub và các cuộc thảo luận hỗ trợ tại khu vực địa phương là trang lỗi GitHub và kênh Slack. Ngoài việc hỗ trợ các tổ chức não phức tạp và không liên tục, Keras còn hỗ trợ các tổ chức não mẫu mực. Các lớp tiện ích tiêu chuẩn như bỏ học, tiêu chuẩn hóa nhóm và gộp chung đều được duy trì. Keras cho phép gửi các mô hình chuyên sâu trên điện thoại di động (iOS và Android), web và Máy ảo Java. Hơn nữa, nó cho phép chuẩn bị phổ biến các mô hình học tập sâu trên các nhóm GPU và TPU, thường liên quan đến CUDA. Mô -đun ứng dụng Keras được sử dụng để cung cấp các mô hình được chuẩn bị trước cho các tổ chức trí não sâu sắc. Các mô hình Keras được sử dụng để xác định, làm nổi bật quá trình trích xuất và hiệu chỉnh. Phần này giải thích cách sử dụng cấu trúc ứng dụng Keras . Một mô hình đã chuẩn bị bao gồm Kiến trúc mô hình và Trọng lượng mô hình. Vì tải mô hình là những bản ghi lớn nên chúng ta nên tải xuống tập dữ liệu ImageNet và tập trung vào thành phần đó. Sau đây là những mô hình được chuẩn bị trước phổ biến nhất.

ResNet

VGG16

Mạng di động

Khởi đầuResNetV2

Khởi đầuV3

Gói nguồn mở Keras cung cấp điểm kết nối Python cho các tổ chức não giả. Keras đóng vai trò là điểm kết nối của thư viện TensorFlow.

Keras đã hỗ trợ nhiều chương trình phụ trợ trước biến thể 2.3, bao gồm TensorFlow, Microsoft Cognitive Toolkit, Theano và PlaidML . Kể từ phiên bản 2.4, hỗ trợ chỉ giới hạn ở TensorFlow. Nó tập trung vào tính dễ sử dụng, tách biệt và khả năng mở rộng để làm việc với việc thử và sai nhanh chóng với các tổ chức não sâu sắc. Nó được tạo ra như một phần của dự án nghiên cứu ONEIROS (Hệ điều hành Robot thông minh thần kinh-điện tử hoàn thiện mở) và nhà sản xuất và bảo trì quan trọng của nó là kỹ sư Francois Chollet của Google. Chollet cũng là nhà sản xuất mô hình mạng não sâu Xception .

**Numpy**

NumPy là một thư viện Python bổ sung hỗ trợ cho các hệ thống và lưới khổng lồ, phức tạp, cũng như vô số khả năng số cấp độ không thể phủ nhận để xử lý các cụm này. Jim Hugunin và nhiều người khác đã xây dựng nên tổ tiên của NumPy, Numeric, phiên bản 12, trước hết. Năm 2005, Travis Oliphant kết hợp NumPy bằng cách trộn các phần của thư viện Numarray của đối thủ vào thư viện Numeric. NumPy là một chương trình nguồn mở được một số người ủng hộ. Điểm nổi bật: NumPy tập trung vào việc thực thi tham chiếu CPython của Python, một trình trung gian không tăng cường mã byte. Thông thường, các phép tính số dành cho biến thể Python này thực hiện chậm hơn đáng kể so với các phép tính được tập hợp của chúng. NumPy giải quyết vấn đề chậm chạp ở một mức độ nào đó bằng cách cung cấp các cụm, chức năng và quản trị viên đa diện hoạt động thực sự trên các chương trình, yêu cầu sửa đổi mã cụ thể, chủ yếu là các vòng tròn bên trong, được viết bằng các ngôn ngữ khác nhau. NumPy trong Python cung cấp khả năng giống như MATLAB vì cả hai đều được giải mã và cho phép khách hàng xây dựng các dự án nhanh chóng miễn là hầu hết các hoạt động được thực hiện trên các cụm hoặc khung thay vì vô hướng. NumPy thường được kết hợp với Python, một ngôn ngữ lập trình hiện đại và hoàn chỉnh hơn, trong khi MATLAB có rất nhiều bộ công cụ tăng cường, hầu hết là Simulink. SciPy là một thư viện bổ sung thêm khả năng giống MATLAB, trong khi Matplotlib là một thiết bị vẽ đồ thị mang lại tính hữu ích cho việc vẽ đồ thị giống MATLAB. Cả MATLAB và NumPy đều phụ thuộc vào BLAS và LAPACK cho các bài toán đa thức thẳng. Các triển lãm NumPy được sử dụng rộng rãi bởi các mối quan hệ Python của OpenCV lập trình thị giác PC được sử dụng rộng rãi để lưu trữ và kiểm soát thông tin. Vì các hình ảnh có một số kênh được sắp xếp thành các cụm ba lớp nên việc sắp xếp, cắt và ẩn các đối tượng khác nhau là những phương pháp cực kỳ hiệu quả để truy cập các pixel cụ thể trong một hình ảnh.

Cụm NumPy là cấu trúc thông tin phổ biến trong OpenCV cho hình ảnh, các tiêu điểm nổi bật đã được loại bỏ, các phần kênh và một số loại thông tin khác hoạt động triệt để trong quá trình lập trình và khắc phục sự cố. Hạn chế: Việc chèn hoặc thêm các thành phần vào một chương trình khó hơn việc sử dụng danh sách của Python. Hệ thống tăng cường triển lãm np. pad(...) tạo các cụm mới với cấu trúc và giá trị đệm được xác định trước, di chuyển cụm đã cho vào phần trưng bày mới và trả về kết quả. Np. chiến lược concatenate([a 1,a 2]) trong NumPy không thực sự liên kết hai cụm mà thay vào đó trả về một triển lãm khác với các thành phần từ hai cụm trong 13 lần liên tiếp. Các thành phần của một cụm có thể được định hình lại bằng cách sử dụng np. reshape(...) nếu số lượng cá thể trưng bày không thay đổi. Do cách hiển thị các cụm NumPy trên các đệm bộ nhớ liên tục nên sẽ xuất hiện một số trường hợp nhất định.

**Mạng lưới thần kinh**

Mạng lưới thần kinh là một tập hợp các phép tính nhằm theo dõi các kết nối ẩn của thông tin bằng cách lặp lại tâm trí con người. Mạng lưới não trong bối cảnh này ám chỉ đến khung thần kinh tự nhiên hoặc giả mạo. Các tổ chức trí não có thể điều chỉnh theo nhiều nguồn thông tin khác nhau, cho phép họ mang lại kết quả tốt nhất mà không cần xem xét lại các tiêu chuẩn kết quả. Bắt đầu với khả năng suy luận được máy tính hóa, các mạng não đang ngày càng nổi tiếng nhờ sự phát triển của các hệ thống trao đổi. Mạng não trông giống như mạng não được ghi lại trong não người. Một "tế bào thần kinh" trong mạng não là một khả năng số tập hợp và sắp xếp đầu vào theo một thiết kế được xác định trước. Thật ngạc nhiên, tổ chức này trông giống như các kỹ thuật có thể đo lường được, chẳng hạn như điều tra độ cong và tái phát.

Một mạng lưới thần kinh được tạo thành từ các lớp trung tâm được kết nối. Giống như nhiều lần tái phát trực tiếp khác, mỗi hub đều là một perceptron. Perceptron xử lý tín hiệu có được bằng cách lặp lại thẳng vào công việc khởi đầu phi tuyến tính. Perceptron trong Perceptron nhiều mặt (MLP) được đặt trong các lớp được kết nối với nhau. Các thiết kế đầu vào được thu thập bởi lớp thông tin. Các ví dụ đầu vào có thể được lên kế hoạch cho sự sắp xếp hoặc tín hiệu kết quả của lớp kết quả. Lượng thông tin được tải được điều chỉnh thông qua các lớp bí mật cho đến khi khả năng sai sót của tổ chức não là rất ít. Người ta đề xuất rằng các lớp bí mật ngoại suy các thuộc tính thông tin thông tin dễ thấy có tiềm năng mang lại lợi nhuận. Điều này có ý nghĩa đối với việc trích xuất phần tử, đáp ứng nhu cầu tương tự như các phương pháp có thể đo lường được như kiểm tra phần đầu. Không gian sử dụng sắp tới là một phần ứng dụng của ANN. Nó suy ra rằng các sự kiện và ứng dụng của ANN sử dụng cách tiếp cận đa ngành. Khả năng nhận biết của diễn ngôn Lời nói có vai trò quan trọng trong giao tiếp của con người. Vì vậy, thông thường mọi người mong đợi giao tiếp bằng giọng nói với PC. Mọi người sử dụng các phương ngữ phức tạp, khó học để nói về sự đổi mới trong thời đại ngày nay. Việc sử dụng ngôn ngữ giao tiếp hợp lý bằng máy có thể là một phương pháp trực tiếp để khắc phục trở ngại về thư từ này. Sự tiến bộ quan trọng đã đạt được ở đây, mặc dù các hệ thống như vậy gặp khó khăn với các thuật ngữ hoặc cú pháp hạn chế, cũng như khó khăn trong việc đào tạo lại hệ thống cho các loa và cài đặt đã thay đổi. ANN đảm nhận vai trò quan trọng trong lĩnh vực này.

**Học kĩ càng**

Học tập sâu sắc: Mạng não có khả năng tăng cường sâu được nhận dạng từ mạng não một lớp được xếp gọn bởi mức độ sâu sắc của chúng hoặc số lượng đầu vào của lớp trung tâm sẽ được chuyển qua trong quy trình xác nhận thiết kế nhiều bước. Các phiên bản trước đây của các tổ chức não, chẳng hạn như các perceptron sớm nhất , chỉ bao gồm một lớp bí mật duy nhất giữa lớp thông tin và lớp kết quả. Học tập sâu sắc bao gồm nhiều lớp (đếm thông tin và kết quả). Điều quan trọng không chỉ đơn giản là một cách diễn đạt phổ biến được sử dụng để tạo ấn tượng rằng PC đọc Sartre và chú ý đến các nhóm đám mây. Một biểu thức có định nghĩa chính xác sẽ kết nối với nhiều lớp được bao phủ khác nhau. Trong các tổ chức học tập sâu, mỗi lớp trung tâm được chuẩn bị dựa trên một loạt đặc điểm cụ thể dựa trên kết quả của lớp trước. Bạn càng thêm nhiều lớp vào tổ chức não thì các đặc tính mà trung tâm của nó có thể cảm nhận được càng phức tạp hơn, bởi vì mỗi lớp kết hợp và kết hợp lại thông tin từ lớp bên dưới. Bao gồm tiến trình có thứ tự là một hệ thống lũy tiến phức tạp và phản ánh ngày càng tăng. Nó trao quyền cho các tổ chức học tập sâu sắc để xử lý các chỉ mục thông tin khổng lồ, có nhiều lớp với hàng tỷ ranh giới phi tuyến tính.

Đặc biệt, mạng lưới não giả có thể tìm thấy các thiết kế được cất giấu bên trong thông tin không có nhãn mác, không có cấu trúc, đóng khung phần lớn thông tin của thế giới. Thông tin phi cấu trúc, bao gồm các tài khoản hình ảnh, văn bản, video và âm thanh, được coi là phương tiện truyền thông thô sơ. Học sâu đặc biệt có kỹ năng kiểm tra và thu thập các phương tiện truyền thông thô sơ, không nhãn mác của thế giới, tìm ra những đặc điểm và đặc thù chung trong tài liệu chưa bao giờ được sắp xếp trong một tập dữ liệu xã hội hoặc được đặt tên bởi một cá nhân. Ví dụ: việc học sâu có thể tập hợp 1.000.000 bức ảnh dựa trên những điểm giống nhau của chúng: những con mèo ở một quận, những người bắt đầu cuộc trò chuyện ở một quận khác và những bức ảnh của bà của bạn ở một quận thứ ba. Điều này đặt ra sự chuẩn bị cho các bộ sưu tập ảnh thông minh. Các tổ chức học tập sâu sắc, không giống với hầu hết các phép tính AI thông thường, loại bỏ các điểm nổi bật một cách tự nhiên mà không cần sự can thiệp của con người. Xét rằng việc trích xuất điểm nổi bật có thể khiến các nhóm nhà nghiên cứu thông tin phải mất nhiều năm mới đạt được, học sâu là một phương pháp để khắc phục tình trạng thiếu kiến thức chuyên môn. Nó nâng cao năng lực của các nhóm khoa học thông tin nhỏ, những nhóm không thể mở rộng quy mô theo bản chất thực tế của họ. Trong khi chuẩn bị các thông tin không được gắn nhãn, mỗi lớp trung tâm trong một tổ chức sâu sắc sẽ tự nhận ra những điểm nổi bật bằng cách liên tục nỗ lực làm lại đóng góp mà nó lấy ví dụ và hạn chế sự khác biệt giữa kỳ vọng của nó và khả năng truyền tải thông tin. Ví dụ, trong thiết kế này, các máy Boltzmann có số lượng hạn chế tạo ra các sản phẩm tái tạo được cho là. Đồng thời, các mạng não này tìm ra cách tìm ra mối quan hệ giữa các quan điểm chính cụ thể và kết quả tối ưu - chúng đặt ra các mối liên hệ giữa các tín hiệu nổi bật và những gì các tín hiệu đó hiển thị, cho dù bản sao đầy đủ hoặc thông tin được đặt tên có đang được sử dụng hay không. Do đó, một mạng lưới học tập sâu sắc đã được chuẩn bị dựa trên thông tin được đánh dấu có thể được áp dụng cho thông tin phi cấu trúc, giúp nó có khả năng đóng góp nhiều hơn không thể phủ nhận so với các tổ chức AI.

# CHƯƠNG II

# BÌNH LUẬN VĂN HỌC

Ngôn ngữ ký hiệu là ngôn ngữ dựa trên hình ảnh sử dụng sự kết hợp của nhiều hình ảnh khác nhau như hình dạng và cử chỉ bàn tay, định hướng, vị trí, chuyển động của bàn tay và cơ thể, chuyển động của môi và nét mặt. Giống như ngôn ngữ nói, các biến thể ngôn ngữ ký hiệu theo khu vực cũng tồn tại, ví dụ: Ngôn ngữ ký hiệu của Ấn Độ (ISL), Ngôn ngữ ký hiệu của Mỹ (ASL) và Ngôn ngữ ký hiệu của Bồ Đào Nha. Có ba loại ngôn ngữ ký hiệu: đánh vần từng bảng chữ cái bằng ngón tay, từ vựng ký hiệu cho các từ, sử dụng cử động tay và cơ thể, nét mặt và cử động môi. Ngôn ngữ ký hiệu cũng có thể bị cô lập cũng như liên tục. Trong ngôn ngữ ký hiệu biệt lập, mọi người giao tiếp bằng cử chỉ của một từ duy nhất, trong khi ngôn ngữ ký hiệu liên tục là một chuỗi cử chỉ tạo ra một câu có ý nghĩa. Danh sách một số công việc được thực hiện trong lĩnh vực dịch ngôn ngữ ký hiệu sang văn bản như sau:



## Tác phẩm liên quan

* Dịch ngôn ngữ ký hiệu sang văn bản bằng Kinect cho Windows v2. Mô hình này đề xuất phương pháp nhận dạng và dịch các cử chỉ động của Ngôn ngữ ký hiệu tiếng Đức (Deutsche Gebärdensprache , DGS) sang văn bản bằng Microsoft Kinect for Windows v2. Hai phương pháp đã được sử dụng cho quá trình nhận dạng cử chỉ: khớp trình tự bằng thuật toán Dynamic Time Warping và kết hợp Trình tạo cử chỉ trực quan cùng với Dynamic Time Warping. Vì mục đích đo điểm chuẩn, 11 cử chỉ DGS do một người dùng chuyên nghiệp đến từ Đức cung cấp đã được lấy làm tập dữ liệu mẫu. Các phương pháp đề xuất được so sánh dựa trên chi phí tính toán và độ chính xác của các cử chỉ này. Thời gian tính toán cho Dynamic Time Warping tăng đều đặn khi số lượng cử chỉ trong tập dữ liệu ngày càng tăng trong khi đó trong trường hợp Visual Gesture Builder với Dynamic Time Warping, thời gian tính toán gần như không đổi. Tuy nhiên, độ chính xác của Visual Gesture Builder với Dynamic Time Warping chỉ là 20,42% trong khi độ chính xác của Dynamic Time Warping là 65,45%. Dựa trên kết quả, chúng tôi đề xuất thuật toán Dynamic Time Warping cho các tập dữ liệu nhỏ và Visual Gesture Builder với Dynamic Time Warping cho các tập dữ liệu lớn [4].
* Dịch ngôn ngữ ký hiệu của Mỹ bằng cách sử dụng tính năng phát hiện cạnh và tương tác chéo Dự án này nhằm triển khai một hệ thống dịch tự động có khả năng dịch ASL sang văn bản tiếng Anh bằng cách sử dụng các môi trường máy tính phổ biến như máy tính và webcam chung. Trong dự án này, hệ thống nhận dạng cử chỉ tay theo thời gian thực sử dụng kết hợp các phương thức xử lý hình ảnh được triển khai. Một ứng dụng giao diện người dùng đồ họa nguyên mẫu để thu thập, xử lý, thu thập và phân tích dấu hiệu ASL được trình bày. Cách tiếp cận này bao gồm giai đoạn trích xuất cử chỉ, sau đó là giai đoạn nhận dạng cử chỉ. Cơ sở dữ liệu cử chỉ hình ảnh được thu thập thông qua ứng dụng và được sử dụng làm thông tin đào tạo để sử dụng trong giai đoạn nhận dạng cử chỉ. Mô hình này nhằm mục đích cung cấp hai mô hình dịch thuật khác nhau:

◦ Ký tự tiếng Anh (bảng chữ cái)

◦ Hoàn thành các từ hoặc cụm từ

Trong phương pháp nhận dạng ký tự riêng lẻ, ảnh cử chỉ tay được xử lý bằng cách kết hợp phân đoạn ảnh và dò tìm cạnh để trích xuất thông tin hình thái và sau đó được xử lý bằng giai đoạn phát hiện cử chỉ nhận dạng chữ cái trong bảng chữ cái tương ứng.

Trong giai đoạn lựa chọn tính năng này, một tập hợp con các khung có thể đại diện cho một từ hoặc cụm từ cụ thể sẽ được chọn. Sau đó, tập hợp các khung biểu thị một từ hoặc một cụm từ sẽ được xử lý bằng kỹ thuật đa phương thức được sử dụng để xử lý các ký tự riêng lẻ. Cuối cùng, giai đoạn nhận dạng cử chỉ được áp dụng cho cả hai phương pháp sử dụng sơ đồ dựa trên hệ số tương quan chéo để phát hiện biểu thức [5].

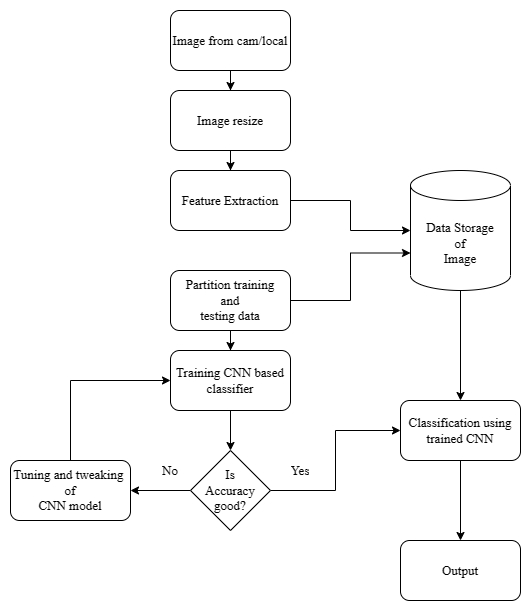
* Ronchetti và cộng sự. [6] đã thảo luận về một phương pháp dựa trên xử lý hình ảnh để trích xuất các bộ mô tả, sau đó là phân loại hình dạng bàn tay bằng cách sử dụng ProbSom , một phương pháp điều chỉnh có giám sát của các bản đồ tự tổ chức. Sử dụng kỹ thuật này, họ có thể đạt được độ chính xác trên 90% trong Ngôn ngữ ký hiệu Argentina. Kỹ thuật phân loại mà họ sử dụng dựa trên khoảng cách Euclide có trọng số riêng. Họ đã xác định được 24 bảng chữ cái khác nhau của Ngôn ngữ ký hiệu Ấn Độ với độ chính xác 96%. Kumud và Neha [8] đề xuất một phương pháp nhận dạng cử chỉ từ video chứa nhiều cử chỉ của Ngôn ngữ ký hiệu Ấn Độ. Họ trích xuất khung hình chính, dựa trên độ dốc, để chia video thành các cử chỉ riêng biệt độc lập. Các tính năng được trích xuất từ cử chỉ bằng cách áp dụng Biểu đồ định hướng và Phân tích thành phần chính.

Lionel và cộng sự. [6 ] đề xuất một hệ thống nhận dạng cử chỉ ngôn ngữ ký hiệu của Ý. Họ đã sử dụng Microsoft Kinect và mạng nơ-ron tích chập (CNN) được tăng tốc thông qua bộ xử lý đồ họa (GPU). Họ đã đạt được độ chính xác xác thực chéo khoảng 92% trên tập dữ liệu bao gồm 20 cử chỉ của Ý. Rajat và cộng sự. [7] đã đề xuất một thiết bị di động được tinh chỉnh như một giải pháp nhằm giảm bớt vấn đề giảm thiểu khoảng cách giao tiếp giữa những người bình thường và những người có khuyết tật khác.

# CHƯƠNG III

# THIẾT KẾ VÀ KIẾN TRÚC HỆ THỐNG

## Sơ đồ khối



Hình: 3.1: Sơ đồ khối thu thập, kiểm tra, huấn luyện và phân loại dữ liệu

## Sử dụng sơ đồ trường hợp

Hình: 3.2: Sơ đồ use-case của Nhận dạng ngôn ngữ ký hiệu

## Sơ đồ luồng dữ liệu

Hình: 3.3: Sơ đồ luồng dữ liệu nhận dạng ngôn ngữ ký hiệu

## Sơ đồ trình tự

Hình: 3.4: Sơ đồ trình tự nhận dạng ngôn ngữ ký hiệu

# CHƯƠNG IV

# PHƯƠNG PHÁP

## Bộ sưu tập tập dữ liệu

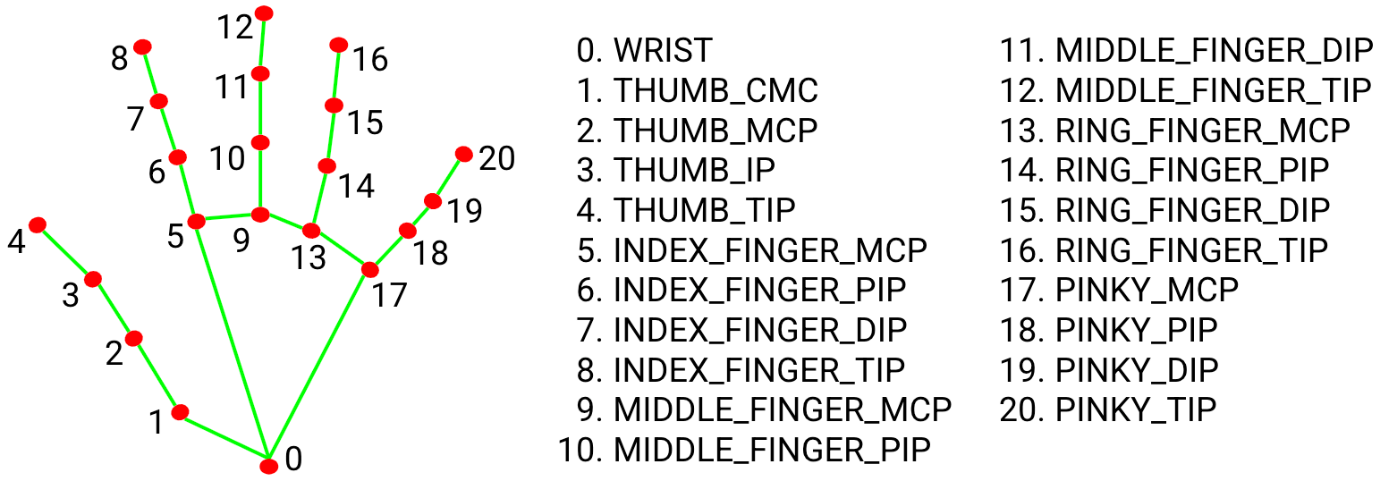
Tập dữ liệu về nhận dạng Ngôn ngữ ký hiệu của Mỹ (ASL) có được bằng cách chụp ảnh cử chỉ tay đại diện cho mọi bảng chữ cái bằng thư viện theo dõi tay Mediapipe . Thư viện theo dõi bàn tay cung cấp một cách nhanh chóng và đáng tin cậy để phát hiện các mốc trên bàn tay và theo dõi chuyển động của bàn tay trong thời gian thực. Sau đó, các cử chỉ tay được ghi lại sẽ được ánh xạ lên hình ảnh nền trắng, cung cấp nền trung tính và nhất quán để xử lý và phân loại hình ảnh. Tập dữ liệu kết quả có thể được sử dụng để đào tạo các mô hình học máy cho các ứng dụng công nghệ hỗ trợ và nhận dạng ASL.

Mediapipe sử dụng phương pháp tiếp cận dựa trên deep learning để phát hiện và theo dõi bàn tay con người trong thời gian thực. Thư viện này dựa trên mô hình mạng thần kinh tích chập (CNN) hạng nhẹ, được đào tạo trên hàng triệu hình ảnh bàn tay có chú thích để tìm hiểu các mốc trên bàn tay và mối liên hệ của chúng. Mạng được tối ưu hóa về hiệu quả và có thể chạy trên thiết bị di động và máy tính để bàn.

Trước tiên, thư viện sẽ phát hiện các vùng bàn tay trong hình ảnh đầu vào bằng thuật toán hồi quy hộp giới hạn. Sau đó, nó đưa các vùng bàn tay được phát hiện vào mô hình mốc bàn tay để ước tính các mốc của mỗi bàn tay. Các điểm mốc trên bàn tay là một tập hợp gồm 21 điểm 2D tượng trưng cho các khớp và đầu ngón tay của bàn tay.

Mô hình điểm mốc bàn tay là một mạng lưới thần kinh chuyển tiếp nguồn cấp dữ liệu, lấy vùng bàn tay được phát hiện làm đầu vào và tạo ra các điểm mốc làm đầu ra. Mô hình được đào tạo bằng cách sử dụng kết hợp dữ liệu tổng hợp và dữ liệu thực để khái quát hóa các hình dạng bàn tay, màu da và điều kiện ánh sáng khác nhau.

Sau khi phát hiện được các mốc của bàn tay, thư viện có thể sử dụng chúng cho nhiều ứng dụng khác nhau, chẳng hạn như nhận dạng cử chỉ, ước tính tư thế bàn tay và thực tế tăng cường. Thư viện cũng cung cấp một bộ tiện ích để trực quan hóa và xử lý các mốc bàn tay, chẳng hạn như vẽ chú thích bàn tay, tính toán các đặc điểm bàn tay và lọc các điểm mốc nhiễu.



Hình: 4.1: Hệ thống mốc của Mediapipe

## Thuật toán được sử dụng

**Mạng thần kinh chuyển đổi (CNN):**

Mạng thần kinh chuyển đổi ( ConvNet /CNN) là một thuật toán Deep Learning có thể lấy hình ảnh đầu vào, gán tầm quan trọng (trọng số và độ lệch có thể học được) cho các khía cạnh/đối tượng khác nhau trong hình ảnh và có thể phân biệt cái này với cái kia. Yêu cầu xử lý trước trong ConvNet thấp hơn nhiều so với các thuật toán phân loại khác. Kiến trúc của ConvNet tương tự như kiểu kết nối của các nơ-ron trong não người và được lấy cảm hứng từ tổ chức của Visual Cortex. Các tế bào thần kinh riêng lẻ chỉ phản ứng với các kích thích trong một vùng hạn chế của trường thị giác được gọi là Trường tiếp nhận. Một tập hợp các trường như vậy chồng lên nhau để bao phủ toàn bộ khu vực thị giác. Mạng nơ-ron tích chập hay ConvNet rất giỏi trong việc nắm bắt các mô hình không gian cục bộ trong dữ liệu. Họ rất giỏi trong việc tìm kiếm các mẫu và sau đó sử dụng chúng để phân loại hình ảnh. ConvNets giả định rõ ràng rằng đầu vào mạng sẽ là hình ảnh. CNN, do có các lớp tổng hợp, không nhạy cảm với việc xoay hoặc dịch hai hình ảnh tương tự nhau; tức là một hình ảnh và hình ảnh được xoay của nó sẽ được phân loại là cùng một hình ảnh. Do những lợi thế to lớn của CNN trong việc trích xuất các đặc điểm không gian của hình ảnh, chúng tôi đã sử dụng mô hình Inception-v3 [8] của thư viện TensorFlow [9], một ConvNet sâu để trích xuất các đặc điểm không gian từ các khung của chuỗi video. Inception là một mô hình phân loại hình ảnh khổng lồ với hàng triệu tham số để phân loại hình ảnh.

**Lớp chập:**

Trong lớp tích chập, tôi đã lấy một kích thước cửa sổ nhỏ [thường có chiều dài 5\*5] kéo dài đến độ sâu của ma trận đầu vào.

Lớp này bao gồm các bộ lọc có thể học được về kích thước cửa sổ. Trong mỗi lần lặp, tôi trượt cửa sổ theo kích thước sải chân [thường là 1] và tính tích số chấm của các mục nhập bộ lọc và giá trị đầu vào tại một vị trí nhất định.

Khi tôi tiếp tục, quá trình này sẽ tạo ra ma trận kích hoạt 2 chiều cung cấp phản hồi của ma trận đó ở mọi vị trí không gian.

Nghĩa là, mạng sẽ tìm hiểu các bộ lọc kích hoạt khi chúng nhìn thấy một số loại tính năng trực quan, chẳng hạn như một cạnh của một số hướng hoặc một đốm màu nào đó.



Hình: 4.2: Các lớp chập khác nhau

**Lớp gộp:**

Chúng tôi sử dụng lớp tổng hợp để giảm kích thước của ma trận kích hoạt và cuối cùng là giảm các tham số có thể học được.

Có hai loại gộp:

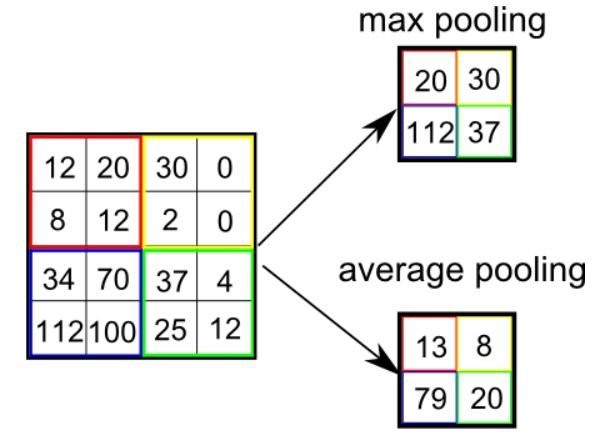
**Một. Tổng hợp tối đa:**

Trong tổng hợp tối đa, chúng tôi lấy kích thước cửa sổ [ví dụ: cửa sổ có kích thước 2\*2] và chỉ lấy tối đa 4 giá trị.

Chà, hãy đóng cửa sổ này lại và tiếp tục quá trình này, vậy là cuối cùng chúng ta sẽ nhận được ma trận kích hoạt bằng một nửa Kích thước ban đầu của nó.

**b. Tổng hợp trung bình:**

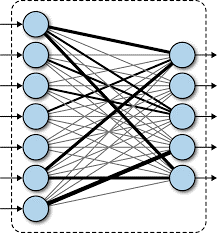
Trong tổng hợp trung bình, chúng tôi lấy trung bình của tất cả các Giá trị trong một cửa sổ.



Hình: 4.3: Tổng hợp trung bình và tổng hợp tối đa

**Lớp được kết nối đầy đủ:**

Trong lớp tích chập, các nơ-ron chỉ được kết nối với một vùng cục bộ, trong khi ở vùng được kết nối đầy đủ, kết nối tốt tất cả các đầu vào với các nơ-ron.



Hình: 4.4: Lớp dày đặc hoặc được kết nối đầy đủ

**c. Lớp đầu ra cuối cùng** :

Trong lớp tích chập, các nơ-ron chỉ được kết nối với một vùng cục bộ, trong khi ở vùng được kết nối đầy đủ, hãy kết nối tốt tất cả các đầu vào với các nơ-ron.

Sau khi nhận được các giá trị từ lớp được kết nối đầy đủ, hãy kết nối chúng với lớp nơ-ron cuối cùng [ có số lượng bằng tổng số lớp], điều này sẽ dự đoán xác suất mỗi hình ảnh thuộc các lớp khác nhau.

## Kiểm tra và xác minh

Một phần của tập dữ liệu sẽ được sử dụng để thử nghiệm và xác minh. Sau khi CNN đã được đào tạo, nó sẽ được kiểm tra trên một tập dữ liệu riêng biệt (được gọi là bộ kiểm tra) để đánh giá hiệu suất của nó trên dữ liệu mà nó chưa từng thấy trước đây. Điều này giúp đảm bảo rằng CNN có thể khái quát hóa kiến thức của mình cho dữ liệu mới

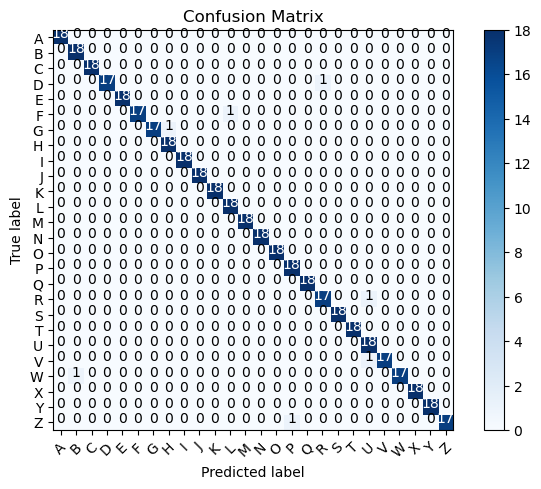
# CHƯƠNG V

# KẾT QUẢ VÀ THẢO LUẬN

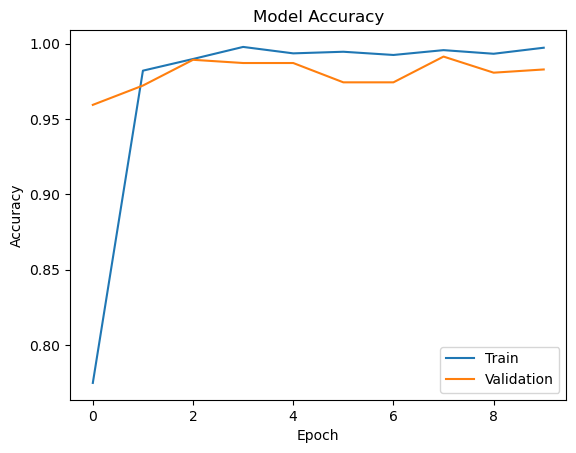
## đầu ra

Đầu ra của dự án là một mô hình học máy có thể nhận dạng bảng chữ cái Ngôn ngữ ký hiệu của Mỹ (ASL) từ hình ảnh cử chỉ tay và chuyển văn bản thành giọng nói kèm theo gợi ý từ. Mô hình đạt độ chính xác 95% trên tập dữ liệu thử nghiệm.

Biểu đồ được tạo trong quá trình đào tạo với ma trận Độ chính xác và nhầm lẫn của Mô hình tương ứng như trong hình



Hình: 5.1: Ma trận nhầm lẫn



Hình: 5.2: Độ chính xác của mô hình

## Công việc đã hoàn thành

Các nhiệm vụ sau đã được hoàn thành trong quá trình thực hiện dự án:

* Thu thập dữ liệu: Chúng tôi đã chụp ảnh cử chỉ tay cho tất cả 26 bảng chữ cái ASL bằng máy ảnh và thư viện theo dõi tay Mediapipe .
* Tiền xử lý dữ liệu: Các hình ảnh được xử lý trước để loại bỏ nền và chuẩn hóa kích thước và vị trí bàn tay.
* Đào tạo mô hình: Chúng tôi đã đào tạo mạng lưới thần kinh tích chập (CNN) trên các hình ảnh được xử lý trước để nhận dạng bảng chữ cái ASL.
* Đánh giá mô hình: Chúng tôi đã đánh giá mô hình trên tập dữ liệu thử nghiệm để đo độ chính xác của nó.
* Chuyển văn bản thành giọng nói: Đã thêm hỗ trợ cho tính năng chuyển văn bản thành giọng nói
* Trình tạo từ: Đã thêm hỗ trợ cho trình tạo từ từ các chữ cái riêng lẻ

## Hạn chế

Một hạn chế của dự án là nó chỉ nhận dạng bảng chữ cái ASL và không hỗ trợ nhận dạng từ hoặc cụm từ. Một hạn chế khác là độ chính xác của mô hình có thể giảm trong điều kiện ánh sáng yếu hoặc tư thế tay khó khăn.

## Vấn đề phải đối mặt

Một số thách thức phải đối mặt trong quá trình thực hiện dự án bao gồm:

* Khó chụp ảnh cử chỉ tay chất lượng cao trong điều kiện ánh sáng khó khăn
* Độ phức tạp trong quá trình xử lý trước hình ảnh để loại bỏ nền và chuẩn hóa kích thước và vị trí bàn tay
* Quá trình đào tạo và tối ưu hóa mô hình CNN tốn thời gian

# CHƯƠNG VI

# KẾT LUẬN VÀ NHỮNG CẢI TIẾN TRONG TƯƠNG LAI



## Phần kết luận

Trong dự án này, chúng tôi đã trình bày một hệ thống nhận dạng Ngôn ngữ ký hiệu của Mỹ (ASL) bằng cách sử dụng phương pháp tiếp cận dựa trên học tập sâu. Hệ thống của chúng tôi sử dụng mạng thần kinh tích chập (CNN) được đào tạo trên tập dữ liệu về cử chỉ tay được ánh xạ tới bảng chữ cái ASL. Chúng tôi đã chứng minh rằng hệ thống của chúng tôi đạt được độ chính xác cao trong việc nhận dạng cử chỉ ASL trong thời gian thực bằng cách sử dụng nguồn cấp dữ liệu video trực tiếp hoặc video đã ghi. Hệ thống của chúng tôi có thể được sử dụng để hỗ trợ những người khiếm thính và khiếm ngôn trong giao tiếp với người khác.

## Những cải tiến trong tương lai

Có rất nhiều phạm vi cho những cải tiến trong tương lai đối với hệ thống nhận dạng ASL của chúng tôi. Đầu tiên, chúng tôi có thể mở rộng tập dữ liệu để bao gồm số lượng cử chỉ tay lớn hơn nhằm nhận dạng nhiều cụm từ và câu ASL hơn. Thứ hai, chúng ta có thể khám phá việc sử dụng các mô hình học sâu tiên tiến hơn như mạng thần kinh tái phát (RNN) và mô hình dựa trên sự chú ý để cải thiện độ chính xác. Ngoài ra, chúng tôi có thể điều tra việc sử dụng các nguồn đầu vào đa phương thức như âm thanh và nét mặt để nâng cao khả năng nhận dạng của hệ thống của chúng tôi. Cuối cùng, chúng ta có thể khám phá việc sử dụng các kỹ thuật học chuyển tiếp để điều chỉnh hệ thống của mình nhằm nhận dạng các ngôn ngữ ký hiệu khác được sử dụng trên khắp thế giới.

# CHƯƠNG VII

**REFERENCES**

[1] Dhiman, M., 2021. Summer Research Fellowship Programme of India's Science Academies

2017. [online] AuthorCafe.

[2] T D, Sajanraj & M V, Beena. (2018). Indian Sign Language Numeral Recognition

Using Region of Interest Convolutional Neural Network.

[3] Chen, J., 2021. CS231A Course Project Final Report Sign Language Recognition

[4] P.Amatya,k.Sergieva & G. Meixener," Translation of Sign Language Into Text Using

Kinect for Windows v2".

[5] A. Joshi, H. Sierra & E. Arzuaga," American sign language translation using edge detection

and cross-correlation".

[6]. Pigou, L., Dieleman, S., Kindermans, P.-J., Schrauwen, B.: Sign language recognition

using convolutional neural networks. In: Workshop at the European Conference on Computer

Vision 2014, pp. 572–578. Springer International Publishing (2014).

[7]. Sharma, R., Bhateja, V., Satapathy, S.C., Gupta, S.: Communication device for differentlyabled people: a prototype model. In: Proceedings of the International Conference on DataEngineering and Communication Technology, pp. 565–575. Springer, Singapore (2017)

[8]Thomas Noltey, Hans Hansson, Lucia Lo Belloz,” Communication Buses for Automotive

Applications” In Proceedings of the 3rd Information Survivability Workshop (ISW-2007),

Boston, Massachusetts, USA, October 2007. IEEE Computer Society.

[9] R. S. Pressman, Software Engineering (3rd Ed.): A Practitioner’s Approach. New York,NY, USA: McGraw-Hill, Inc., 1992.