

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC ĐÀ LẠT

BÁO CÁO TỔNG KẾT
ĐỀ TÀI KHOA HỌC SINH VIÊN NĂM 2024

XÂY DỰNG ỨNG DỤNG NHẬN DIỆN
NGÔN NGỮ KÝ HIỆU

Mã số đề tài: 04

Thuộc nhóm ngành khoa học: Công nghệ Thông Tin

Lâm Đồng, tháng 5 năm 2024

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC ĐÀ LẠT

BÁO CÁO TỔNG KẾT
ĐỀ TÀI KHOA HỌC SINH VIÊN NĂM 2024

Thuộc nhóm ngành khoa học: Công nghệ Thông Tin

Sinh viên thực hiện: Vũ Bá Đông

Nam, Nữ: Nam

Dân tộc: Kinh

Lớp, khoa: CTK45, Khoa CNTT

Năm thứ: 3/ Số năm đào tạo: 4

Ngành học: Kỹ thuật phần mềm

Giáo viên hướng dẫn: ThS. Phan Thị Thanh Nga

Lâm Đồng, tháng 5 năm 2024

MỤC LỤC

MỤC LỤC.....	1
DANH MỤC HÌNH ẢNH	4
DANH MỤC BẢNG.....	5
DANH MỤC VIẾT TẮT	6
THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI	7
1. Thông tin chung:.....	7
2. Mục tiêu đề tài:.....	7
2.1. Về lý thuyết.....	7
2.2. Về thực nghiệm	8
3. Giới thiệu đề tài.....	8
4. Kết quả nghiên cứu	8
5. Đóng góp về mặt kinh tế - xã hội, giáo dục và đào tạo, an ninh, quốc phòng và khả năng áp dụng của đề tài:	9
5.1 Hiệu quả xã hội:	9
5.2 Khả năng áp dụng của đề tài:.....	9
6. Công bố khoa học của sinh viên từ kết quả nghiên cứu của đề tài:.....	9
THÔNG TIN VỀ SINH VIÊN CHỊU TRÁCH NGHIỆM CHÍNH THỨC THỰC HIỆN ĐỀ TÀI	11
I. SƠ LƯỢC VỀ SINH VIÊN	11
II. QUÁ TRÌNH HỌC TẬP	11
MỞ ĐẦU.....	13
1.Tổng quan tình hình nghiên cứu của đề tài.....	13
2. Lý do chọn đề tài.....	15

3. Mục tiêu đề tài.....	15
3.1 Mục tiêu tổng quát.....	15
3.2 Mục tiêu cụ thể:	16
4. Phương pháp nghiên cứu.....	16
5. Đối tượng nghiên cứu	17
6. Phạm vi nghiên cứu.....	17
CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN	18
1.1. Máy học là gì?	18
1.2. Mạng nơ ron nhân tạo.....	19
1.2.1. Mạng nơ ron nhân tạo	19
1.2.2 Convolutional neural networks (CNNs)	20
1.2.5 Gated Recurrent Units (GRUs)	22
1.3. Các thư viện hỗ trợ xây dựng ứng dụng nhận diện ngôn ngữ ký hiệu	23
1.3.1 OpenCV	23
1.3.2 MediaPipe	23
1.3.3 Tensorflow.....	24
1.3.4 Kivy	26
CHƯƠNG 2. XÂY DỰNG MÔ HÌNH.....	27
2.1 Tập dữ liệu huấn luyện.....	27
2.2 Tiền xử lý dữ liệu	31
2.3 Trích xuất đặc trưng hành động	34
2.4. Tiến hành huấn luyện mô hình.....	34
2.5. Kết quả đánh giá và so sánh các mô hình	35
CHƯƠNG 3. TRIỂN KHAI CHƯƠNG TRÌNH	39

3.1 Phương pháp phát triển ứng dụng và công nghệ.....	39
3.1.1 Phương pháp phát triển.....	39
3.1.2 Công nghệ phát triển ứng dụng.....	40
3.2 Quy trình của chương trình.....	41
3.2.1 Phân tích chung.....	41
3.2.2 Ứng dụng phía Server.....	42
3.2.3 Ứng dụng phía Client.....	43
3.3 Kết quả triển khai.....	43
CHƯƠNG 4. KẾT LUẬN.....	46
DANH MỤC THAM KHẢO	48

DANH MỤC HÌNH ẢNH

Hình 1. Lịch sử máy học	18
Hình 2. Quy trình hình ảnh được xử lý của thư viện MediaPipe	23
Hình 3. Minh họa thực tế cho các điểm đánh dấu.....	24
Hình 4. Quy trình thu thập dữ liệu.....	27
Hình 5. Quy trình tạo tệp dữ liệu.....	28
Hình 6. Tổ chức tệp dữ liệu	31
Hình 7. Phương pháp làm mờ, phóng to, thay đổi hướng bàn tay	32
Hình 8. Tiền xử lý khung hình.....	34
Hình 9. Biểu đồ thể hiện tỷ lệ dự đoán chính xác giữa các mô hình.....	36
Hình 10. Biểu đồ so sánh giữa các mô hình.....	37
Hình 11. Hệ thống nhận diện ngôn ngữ ký hiệu.....	39
Hình 12. Hệ thống xử lý Server-Client.....	42
Hình 13. Xử lý chức năng phía Server	42
Hình 14. Xử lý chức năng phía Client.....	43
Hình 15. Ứng dụng nhận diện ngôn ngữ ký hiệu.....	45

DANH MỤC BẢNG

Bảng 1. Thống kê hình ảnh của tệp dữ liệu.....	30
Bảng 2. Thống kê số lượng video trong tập dữ liệu.....	33
Bảng 3. Thống kê thông tin mô hình	38
Bảng 4. Thống kê hành động dễ nhầm lẫn.....	44

DANH MỤC VIẾT TẮT

STT	Chữ viết tắt	Viết đầy đủ
1	AI	Artificial Intelligence
2	DL	Deep Learning
3	ML	Machine Learning
4	CNTT	Công nghệ thông tin
5	CNN	Convolution Neural Network
6	RNN	Recurrent Neural Networks
7	LSTM	Long Short-Term Memory Networks
8	GRU	Gated Recurrent Units

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC ĐÀ LẠT

THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI

1. Thông tin chung:

Tên đề tài: “Xây dựng ứng dụng nhận diện ngôn ngữ ký hiệu”

Sinh viên thực hiện:

STT	Họ và tên	MSSV	Lớp	Khoa	Năm thứ
1	Vũ Bá Đông	2111818	CTK45-PM	CNTT	3
2	Nguyễn Tân Tiến	2111896	CTK45-PM	CNTT	3
3	Tăng Thế Ngọc Song	2111881	CTK45-PM	CNTT	3

Giảng viên hướng dẫn: ThS. Phan Thị Thanh Nga

2. Mục tiêu đề tài:

2.1. Về lý thuyết

Tìm hiểu được lịch sử hình thành và phát triển cùng với các kiến thức cơ bản của Học máy và Học sâu, tìm hiểu được cơ sở toán học của các mô hình. Tìm hiểu các thư viện học máy nổi tiếng hiện nay, cách sử dụng thư viện để huấn luyện mô hình và triển khai xây dựng ứng dụng nhận diện trên nền tảng di động. Nắm rõ kiến thức về kiến trúc LSTM và GRU, tìm hiểu các mô hình học máy nổi tiếng những năm gần đây.

2.2. Về thực nghiệm

Đề tài hướng đến việc xây dựng mô hình huấn luyện và triển khai mô hình trên nền tảng di động, cho phép nhận diện ngôn ngữ ký hiệu từ camera thiết bị theo thời gian thực, dự đoán hành động và đưa ra những thông tin về hành động đó.

3. Giới thiệu đề tài

Trong đời sống, việc truyền đạt, ý nghĩ lời nói thông qua khẩu ngữ là việc rất tự nhiên, tuy nhiên đối với người khiếm thính và khiếm thanh là một việc gần như bất khả thi. Vì vậy ngôn ngữ ký hiệu là một phương tiện quan trọng trong việc truyền đạt ý nghĩa và tương tác của những người không may mắn này. Vì việc sử dụng thủ ngữ chỉ phổ biến cho một cộng đồng người khuyết tật này nên việc hiểu và sử dụng ngôn ngữ ký hiệu đôi khi gặp khó khăn đối với những người không được đào tạo.

Nghiên cứu này tập trung vào việc phát triển một ứng dụng thông minh có khả năng nhận diện và dịch ngôn ngữ ký hiệu thành văn bản. Mục tiêu chính của đề tài là tạo ra một ứng dụng hữu ích giúp người khiếm thính giao tiếp dễ dàng hơn trong các tình huống hàng ngày, cũng như giúp những người không quen thuộc với ngôn ngữ ký hiệu có thể hiểu và tương tác với cộng đồng này một cách hiệu quả. Hiện tại ứng dụng có khả năng nhận diện từng từ của ngôn ngữ ký hiệu, chưa kiểm tra hiệu suất hay đánh giá ứng dụng khi nhận diện từng câu.

4. Kết quả nghiên cứu

Thu thập được tập dữ liệu từ các đoạn phim lưu trữ của 11 chữ trong ngôn ngữ ký hiệu với số lượng bao gồm 3003 video.

Đề tài đã tiến hành tìm hiểu, xây dựng, huấn luyện thành công mô hình Máy học, so sánh các mô hình đã huấn luyện và chọn ra mô hình tối ưu nhất. Mô hình nhận diện ngôn ngữ ký hiệu có khả năng phân lớp hành động tốt 11 chữ trong thủ ngữ.

Triển khai ứng dụng di động sau khi huấn luyện và kết xuất mô hình, sử dụng mô hình đã kết xuất để áp dụng vào ứng dụng chạy trên nền tảng client-server. Xây dựng giao diện và các chức năng cho ứng dụng. Nhóm chúng tôi đã thử nghiệm ứng dụng nhận diện ngôn ngữ ký hiệu và nhận về kết quả tốt, với mô hình tự tạo có lớp LSTM + GRU có độ nhận diện hành động là 80,2% với ngôn ngữ ký hiệu Việt Nam

5. Đóng góp về mặt kinh tế - xã hội, giáo dục và đào tạo, an ninh, quốc phòng và khả năng áp dụng của đề tài:

5.1 Hiệu quả xã hội:

Hệ thống nhận diện cử chỉ và bàn tay cũng cấp cho người không biết ngôn ngữ ký hiệu một cách thức đơn giản và nhanh chóng trong việc giao tiếp với người khiếm thính với các từ thường nhật. Điều này góp phần cải thiện khả năng hòa nhập và giao tiếp với người khiếm thính.

5.2 Khả năng áp dụng của đề tài:

Có thể sử dụng kết quả nghiên cứu của đề tài có thể áp dụng rộng rãi trong mọi mặt đời sống hay ứng dụng trong các ngành khác. Ví dụ như trong việc tương tác với thiết bị thông qua cử chỉ, có thể áp dụng việc đánh máy dựa vào cử chỉ, dự đoán hành động của con người, ...

6. Công bố khoa học của sinh viên từ kết quả nghiên cứu của đề tài:

Chưa có công bố nào về kết quả nghiên cứu của đề tài với mô hình và ứng dụng xây dựng.

Ngày tháng năm

Sinh viên chịu trách nhiệm chính

thực hiện đề tài

(ký, họ và tên)

Nhận xét của người hướng dẫn về những đóng góp khoa học của sinh viên thực hiện đề tài (phần này do người hướng dẫn ghi)

- Nhóm sinh viên đã thực hiện khảo sát các ứng dụng và nghiên cứu liên quan tới đề tài, các mô hình tiền huấn luyện, các kiến trúc mạng học sâu, các thư viện học máy hỗ trợ.
- Thu thập và xây dựng được tập dữ liệu huấn luyện gồm các video ngắn, mỗi video gồm chuỗi các hành động (thủ ngữ) minh họa một từ.
- Tìm hiểu và áp dụng các phương pháp tiền xử lý dữ liệu, làm giàu dữ liệu để tăng số lượng của tập dữ liệu huấn luyện; kỹ thuật rút trích đặc trưng của bàn tay phù hợp.
- Tìm hiểu các mô hình tiền huấn luyện, xây dựng kiến trúc mạng sử dụng kết hợp LSTM và GRU từ đó tiến hành huấn luyện các mô hình trên tập dữ liệu thu thập. Nhóm sinh viên đã thực hiện so sánh hiệu quả nhận dạng của các mô hình này và đề xuất lựa chọn được mô hình phù hợp để xây dựng ứng dụng minh họa.

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC ĐÀ LẠT

THÔNG TIN VỀ SINH VIÊN

CHỊU TRÁCH NGHIỆM CHÍNH THỨC THỰC HIỆN ĐỀ TÀI

I. SƠ LƯỢC VỀ SINH VIÊN

Họ và tên: Vũ Bá Đông

Sinh ngày: 07/11/2002

Nơi sinh: Đà Lạt

Lớp: CTK45-PM

Khóa:45

Khoa: Công nghệ thông tin

Địa chỉ liên hệ: Lớp CTK45 - PM

Khoa Công nghệ Thông tin, Trường Đại học Đà Lạt

Điện thoại: 0325529551

Email:2111818@dlu.edu.vn

II. QUÁ TRÌNH HỌC TẬP

● Năm thứ 1:

Ngành học:

Khoa: Công nghệ thông tin

Kết quả xếp loại học tập: ĐTBTL: 3.57/4.0

● Năm thứ 2:

Ngành học:

Khoa: Công nghệ thông tin

Kết quả xếp loại học tập: ĐTBTL: 3.45/4.0

Sơ lược thành tích: Học bổng KKHT học kỳ I,II

● ***Năm thứ 3:***

Ngành học:

Khoa: Công nghệ thông tin

Kết quả xếp loại học tập: ĐTBTL: 3.57/4.0

Sơ lược thành tích: Học bổng KKHT học kỳ I

MỞ ĐẦU

1. Tổng quan tình hình nghiên cứu của đề tài

Hiện nay, việc giao tiếp của người khiếm thanh, khiếm thị đa phần đều dùng ngôn ngữ ký hiệu. Tuy nhiên không phải ai cũng biết loại ngôn ngữ đặc biệt này vì thế mà cách thứ hai để giao tiếp đó là viết ra giấy hoặc điện thoại mỗi khi giao tiếp với người không hiểu ngôn ngữ ký hiệu. Đây có thể là một rào cản khiến cho việc giao tiếp của họ chậm hơn. Theo điều tra Quốc gia, ở Việt Nam số lượng người khiếm thính là hơn 1,03 triệu người và con số này vẫn đang tiếp tục gia tăng, theo đà tăng của dân số thế giới và Việt Nam [1]. Hiện tại, có một vài ứng dụng được phát triển nhằm hỗ trợ cho người khiếm thính, một vài nghiên cứu được công bố tuy nhiên các ứng dụng chưa thể dịch các ngôn ngữ ký hiệu chính xác.

Một số sáng chế về công nghệ nhằm hỗ trợ người khiếm thính tính từ giai đoạn 2015-2022 đã đạt 20.695 về hỗ trợ, tăng cường và phục hồi chức năng trong đó bao gồm công nghệ nhận dạng giọng nói hoặc ngôn ngữ, xử lý giọng nói hoặc chuyển đổi âm thanh thành văn bản [1]. Một số ứng dụng nhằm hỗ trợ người khiếm thanh, khiếm thính có thể kể đến như Sound Transcribe (Google), SpeakLiz (Apple) giúp chuyển đổi giọng nói thành văn bản đồng thời lưu lại lịch sử dịch thuật. Hay tại Đại học Quốc tế Bách khoa HCM (BL_OISP) đã sáng tạo thiết bị hỗ trợ người khiếm thanh, khiếm thính với mô hình LSTM. Thiết bị Speak Your Mind và Sign2Speech nhận biết những câu thủ ngữ (ngôn ngữ ký hiệu) cơ bản. Nếu có thêm thời gian và được đầu tư đúng mức thì thiết bị có thể nhận biết được nhiều hơn. [2]

Deng (2023) trong nghiên cứu [3] đã chỉ ra phương hướng nhận diện các vật thể bằng phương pháp Single Shot Detector(SSD) áp dụng với mô hình YoLov5 để có thể thực hiện nhận diện, mô hình này nhận diện Các vật thể có kích thước nhỏ nên có thể phù hợp với việc nhận diện các cử chỉ của ngón tay khi thực hiện nhận diện. Nhóm tác giả Võ Hồng Khanh và Phạm Nguyên Khang lại sử dụng hệ thống cảm biến Kinect và máy học véc tơ hỗ trợ để trích xuất đặc trưng và sử dụng để huấn luyện mô hình nhận dạng ngôn ngữ ký hiệu với tổng 2.400 mẫu dữ liệu. Kết quả cho thấy hệ thống nhận dạng của nhóm tác giả đạt kết

quả 99.46% tuy nhiên việc sử dụng cảm biến Kinect tốn nhiều chi phí để thực hiện hơn tất cả [4]. Nhóm tác giả Lê Minh Thành và các cộng sự đã áp dụng mạng học sâu (Deep neural network) để nhận dạng cử chỉ bàn tay sử dụng mạng nơ-ron tích chập. Trong công trình này nhóm tác giả đã nhận dạng 6 cử chỉ bàn tay với ảnh đầu vào được chụp trực tiếp từ camera với tỷ lệ đúng trung bình lên tới 98.6% với tập kiểm tra gồm 920 ảnh. Công trình trên được lặp lại từ 100 đến 882 lần, cho thấy sau 882 lần huấn luyện thì tỷ lệ nhận dạng chính xác đã cải thiện từ 17,3% lên đến 99,17% ở lần huấn luyện thứ 882 [5].

Challa Neha và các cộng sự đã thực hiện một cuộc khảo sát về việc nhận diện ngôn ngữ ký hiệu trong lĩnh vực này. Nhóm tác giả V.H. Iyer và cộng sự đã thực hiện việc nhận dạng chuyển động của tay và khuôn mặt bằng việc áp dụng thư viện OpenCV, MediaPipe và mô hình LSTM để nhận diện ba ngôn ngữ ký hiệu là “Please”, “Goodbye” và “Love”. Nhóm tác giả đã thực hiện so sánh giữa các mô hình trong bài báo trước thông qua đó xác định sử dụng trên từng khung hình của một đoạn video để dự đoán hành động trong thời điểm đó. Bằng việc thay đổi phương thức huấn luyện mô hình đã thể hiện độ chính xác 90.8% trên tập dữ liệu huấn luyện và 87.5% trên tập dữ liệu thử nghiệm; kết quả này chứng tỏ mô hình này tốt hơn những mô hình được đề xuất trong những năm trước đó tính từ thời điểm công bố. Kết quả thực nghiệm của nghiên cứu này mang lại những kết quả khả quan. Cụ thể như sau: kết quả thực nghiệm được thực hiện trên tập dữ liệu gồm “Hello”, “Thanks” và “Iloveyou” sử dụng thư viện và phương pháp kể trên đạt 89.2% [6]. Shreyas Mhatre cùng các cộng sự đã xây dựng được mô hình sử dụng LSTM để phân loại các hành động trong ngôn ngữ ký hiệu. Việc xây dựng hệ thống này đã có thể phân biệt 7 cử chỉ với độ huấn luyện chính xác cao nhất từ 90% đến 96%. Mhatre (2022) trong nghiên cứu [7] đã kết hợp sử dụng các kỹ thuật xử lý ảnh kết hợp trích xuất đặc trưng và phân lớp học máy SVM để nhận diện dữ liệu. Với phương pháp này độ chính xác chỉ lên tới 100% trên cả tập dữ liệu kiểm thử và tập dữ liệu huấn luyện, tuy nhiên mô hình chỉ nhận diện được chữ cái không thể nhận diện từ hoặc cụm từ.

Pei (2022) sử dụng một số mô hình học máy đồng thời tận dụng phương pháp mới nhằm nhận diện các bàn tay trong ảnh như sử dụng kỹ thuật Transformer [8] để tách các lớp ảnh để nhận diện được nhiều bàn tay hơn, với Prof.Radha (2020) [9] lại sử dụng học

máy để phân biệt các chữ cái bằng cách dùng thuật toán Classification. Chen (2022) [10] lại nghiên cứu mô hình nhận diện bàn tay bằng cách tạo ba mô hình trong đó bao gồm mô hình nhận diện ảnh bàn tay, mô hình nhận diện ảnh bàn tay và đánh dấu chúng và mô hình chỉ đánh dấu bàn tay. Kết quả nhận được cho thấy với mô hình nhận diện ảnh bàn tay chỉ đạt 41.7%, mô hình nhận diện ảnh và đánh dấu bàn tay đạt 23.3% và mô hình chỉ đánh dấu bàn tay lại đạt 85%. Kothadiya (2022) [11] thực hiện việc nhận diện thông qua mô hình InceptionResNetV2 với kiến trúc mạng LSTM và GRU đối với khung hình trong video. Nhóm tác giả thực hiện việc trích xuất đặc trưng bằng mô hình tiền huấn luyện InceptionResNetV2, kết quả nhận được cho thấy với mô hình sử dụng lớp LSTM và GRU đạt tỷ lệ 97% đối với ngôn ngữ Ấn độ.

Trong nghiên cứu này, mô hình dự đoán được đề xuất dựa trên phương pháp học sâu (Deep Learning) sử dụng mô hình có kiến trúc mạng LSTM kết hợp GRU và thư viện MediaPipe để trích xuất vị trí của bàn tay nhằm nhận diện ngôn ngữ ký hiệu được thể hiện giúp người khiếm thính có thể giao tiếp dễ dàng hơn với những người xung quanh có độ chính xác cao.

2. Lý do chọn đề tài

Hiện nay việc nhận diện cử chỉ và bàn tay đã được phổ biến đối với nhiều ngành nghề, lĩnh vực khác nhau. Việc xây dựng một hệ thống nhận diện cử chỉ và bàn tay có thể đáp ứng được nhiều nhu cầu khác vẫn đang được mở rộng. Trong đó ngôn ngữ ký hiệu là một trong những bài toán có thể áp dụng tốt phương pháp này; việc xây dựng ứng dụng nhận diện ngôn ngữ ký hiệu có thể giúp cho việc giao tiếp giữa mọi người trở nên dễ dàng hơn đối với người không biết loại ngôn ngữ này.

3. Mục tiêu đề tài

3.1 Mục tiêu tổng quát

Xây dựng một ứng dụng trên di động cho phép nhận diện ngôn ngữ ký hiệu từ camera thiết bị, sử dụng thư viện TensorFlow để huấn luyện, dự đoán hành động và kết xuất ra ý nghĩa hành động để người dùng theo dõi.

3.2 Mục tiêu cụ thể:

Mục tiêu cụ thể của đề tài được trình bày trong bảng, theo nguyên tắc SMART:

Tính cụ thể (Specific)	Tìm hiểu một số phương pháp Học máy điển hình, áp dụng thuật toán trên tập dữ liệu thu thập, đánh giá kết quả của hai thuật toán để từ đó chọn lựa được một mô hình phù hợp với độ chính xác cao để triển khai lên thiết bị di động để người dùng tương tác
Tính đo lường (Measurable)	Thu thập hình ảnh của ngôn ngữ ký hiệu, với mỗi ngôn ngữ ký hiệu thu thập khoảng 50 - 90 đoạn phim để đưa vào quá trình huấn luyện, sau đó thực hiện tiền xử lý hình ảnh để kết quả dự đoán cao hơn 85%
Tính khả thi (Achievable)	Ứng dụng được xây dựng và có thể triển khai trên thiết bị di động để thử nghiệm nhận diện ngôn ngữ ký hiệu
Tính thực tế (Realistic)	Phạm vi nghiên cứu của đề tài này là phù hợp với trình độ của người thực hiện cũng như là kết quả mà đề tài mang lại là phù hợp với tình hình thực tế hiện nay
Tính thời gian (Timely)	Đề tài sẽ hoàn thành theo đúng tiến độ của nghiên cứu khoa học mà trường Đại học Đà Lạt đã công bố

4. Phương pháp nghiên cứu

Đề tài này sử dụng một vài phương pháp nghiên cứu nhưng chủ yếu là phương pháp phân tích và tổng kết kinh nghiệm. Cụ thể là từ những công trình nghiên cứu liên quan đến đề tài, tổng kết kinh nghiệm thông qua quá trình thực hiện đề tài và sự hỗ trợ từ thư viện máy học TensorFlow để đề xuất một cách tiếp cận mới trong giải quyết vấn đề đặt ra.

5. Đối tượng nghiên cứu

- Nghiên cứu về nhận dạng ngôn ngữ cử chỉ và bàn tay.
- Nghiên cứu và xây dựng bộ dữ liệu cho nhận dạng cử chỉ Tiếng Việt.

6. Phạm vi nghiên cứu

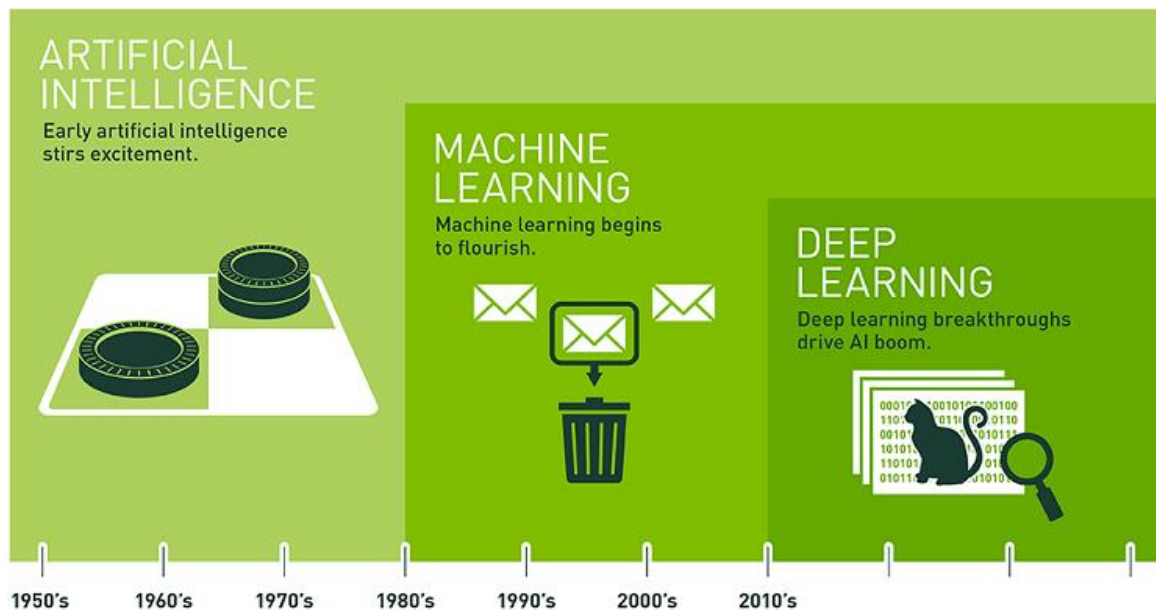
- Nghiên cứu về ngôn ngữ ký hiệu Tiếng Việt.
- Nghiên cứu về phương pháp thu nhận dữ liệu và xử lý ảnh.
- Nghiên cứu về nhận dạng hành động đối với ngôn ngữ ký hiệu Việt Nam.

CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN

1.1. Máy học là gì?

Công nghệ máy học (Machine Learning) là một nhánh quan trọng trong trí tuệ nhân tạo (AI) với khả năng tự động hóa và tối ưu hóa các quy trình trong nhiều lĩnh vực, từ sản xuất đến quản lý và đời sống hàng ngày như trong nhận diện khuôn mặt [12]. Machine Learning đặt trọng điểm vào việc phát triển các hệ thống, thiết bị có khả năng tự học mà không cần sự can thiệp cụ thể từ phía con người. Chúng có khả năng tự điều chỉnh và cải thiện qua thời gian dựa trên dữ liệu và kinh nghiệm mà chúng thu thập được trong quá trình học hỏi, tạo ra một quá trình học tương tự như con người, nhưng áp dụng trong môi trường máy móc.

Về định nghĩa khái niệm, máy học là một lĩnh vực thuộc Khoa học máy tính, đồng thời là một lĩnh vực con của *Trí tuệ nhân tạo (AI)*. Mối quan hệ giữa *Trí tuệ nhân tạo*, *Học máy* và *Học sâu* được thể hiện



Hình 1. Lịch sử máy học

Arthur Samuel đã định nghĩa vào năm 1959, “ML là một lĩnh vực nghiên cứu cung cấp cho máy vi tính khả năng học mà không cần phải lập trình rõ ràng” [13]. Một định

nghĩa thể hiện rõ ràng hơn vào năm 1997: “Một chương trình máy tính được cho là học hỏi từ kinh nghiệm E có liên quan với một vài nhiệm vụ T và hiệu suất đo lường P, nếu hiệu suất của nó trên T được đo bằng P cải thiện sau khi trải qua kinh nghiệm E” [14].

Ví dụ: AlphaGo là một chương trình máy tính do Google DeepMind phát triển đã học được cách chơi cờ vây và đánh bại cờ thủ nổi tiếng Lee Se-dol [15]. Theo định nghĩa của Tom Mitchell, ta có:

- E: Kinh nghiệm có được khi chơi với các đối thủ và tự tăng cường.
- T: Nhiệm vụ là chơi cờ vây.
- P: Khả năng mà AlphaGo sẽ thắng trong trận đấu tiếp theo.

Nhìn chung, để xác định được một vấn đề học tập rõ ràng cần phải nhận biết được ba đặc trưng sau: các nhiệm vụ, hiệu suất cần được cải thiện và nguồn gốc kinh nghiệm.

1.2. Mạng nơ ron nhân tạo

1.2.1. Mạng nơ ron nhân tạo

Theo [16] mạng nơ ron nhân tạo là các kỹ thuật học máy phổ biến theo cơ chế học trong các cơ thể sống. Hệ thống thần kinh của con người chứa các tế bào được gọi là nơ-ron. Các nơ-ron được kết nối với nhau, khi có tác nhân bên ngoài tác động làm thay đổi các kết nối tới nơ-ron. Sự thay đổi này là cách mà học tập diễn ra, cơ chế này được mô phỏng trong mạng nơ ron nhân tạo, mà bao gồm các đơn vị tính toán được gọi là nơ-ron. Các đơn vị tính toán được kết nối với nhau thông qua các trọng số, có cùng chức năng.

Mạng nơ-ron nhân tạo tính toán một hàm của các đầu vào bằng cách truyền giá trị được tính toán từ các nơ-ron đầu vào đến nơ-ron đầu ra và sử dụng các trọng số như các tham số trung gian. Việc học diễn ra bằng cách thay đổi các trọng số kết nối tới các nơ-ron, tác nhân bên ngoài trong mạng nơ-ron nhân tạo được cung cấp bởi dữ liệu huấn luyện chứa các ví dụ cặp đầu vào – đầu ra của hàm cần học. Dữ liệu huấn luyện có thể chứa các điểm ảnh của hình ảnh (đầu vào) và nhãn được chú thích của chúng. Dữ liệu được cung cấp phản hồi cho sự chính xác của các trọng số trong mạng nơ-ron dựa trên việc dự đoán đầu ra cho

một dự đoán đầu vào cụ thể có khớp với nhãn đầu ra được chú thích trong dữ liệu huấn luyện

Ví dụ: Tập dữ liệu chứa hình ảnh của chó và mèo, tách làm hai tập riêng biệt của 2 loài. Hình ảnh trong tập được trích xuất các điểm ảnh và gán nhãn theo đúng loại của tập đó. Dựa vào định nghĩa máy học ta có thể hiểu như sau:

E: Trọng số có được thông qua điểm ảnh gán nhãn.

T: Phân biệt chó hoặc mèo.

P: Khả năng dự đoán được chó hoặc mèo.

Mục tiêu của việc thay đổi trọng số là sửa đổi hàm tính toán để làm cho các dự đoán chính xác hơn trong các lần lặp sau. Bằng cách thay đổi trọng số giữa các nơ-ron qua nhiều cặp đầu vào-đầu ra, mô hình được huấn luyện với nhiều hình ảnh khác nhau, cuối cùng mô hình có khả năng nhận dạng một cặp đầu vào-đầu ra một cách chính xác trong một hình ảnh mà chưa từng thấy trước đây.

1.2.2 Convolutional neural networks (CNNs)

Convolutional neural networks (CNNs): Mạng nơ-ron tích chập (CNNs) là một lớp của mạng nơ-ron nhân tạo. Có khả năng mạnh mẽ trong quá trình phân tích, nhận diện hình ảnh, âm thanh hoặc một số dữ liệu khác. Bằng cách sử dụng các phép toán tích chập để trích xuất các đặc điểm và xác định các mẫu trong hình ảnh. CNNs sử dụng một loạt các lớp nhằm phát hiện các đặc điểm khác nhau của hình ảnh đầu vào. Tùy thuộc vào mức độ phức tạp của mục đích sử dụng mà số lớp có thể là hàng chục, trăm hay hàng nghìn lớp. Đối với CNNs đầu ra của lớp trước đó sẽ là đầu vào của lớp tiếp theo. Mạng CNNs thích hợp cho việc nhận diện hình ảnh

1.2.3 Recurrent Neural Networks (RNNs):

Recurrent Neural Networks (RNNs): Mạng nơ-ron tuần hoàn (RNN) là một lớp của mạng nơ-ron nhân tạo. Khi mạng NN gặp các bài toán dạng chuỗi như mô tả, hoàn thành câu,... thì mô hình NN không còn phù hợp vì những dự đoán tiếp theo như từ tiếp theo phụ

thuộc vào vị trí của nó trong câu và những từ đứng trước nó. Mạng nơ-ron tuần hoàn ra đời với mục đích lưu lại thông tin từ những lần dự đoán từ trước để đưa ra dự đoán chính xác cho bước hiện tại. Mạng nơ-ron tuần hoàn hoạt động dựa trên nguyên tắc sử dụng kết quả đầu ra của một lớp và sử dụng cho đầu vào mới để dự đoán đầu ra của lớp tiếp theo. Mạng nơ-ron tái phát và các biến thể của chúng được sử dụng trong nhiều ứng dụng khác nhau như học từ chuỗi đến chuỗi, mô tả hình ảnh, dịch máy và phân tích cảm xúc. RNN được thiết kế để xử lý dữ liệu chuỗi bằng cách lưu trọng số và sử dụng nó để xử lý thông tin từ cặp đầu vào-đầu vào trước đến cặp đầu vào-đầu ra sau. Điều này hỗ trợ cho việc xử lý video vì các khung hình trong video có thứ tự, nếu đảo lộn các ảnh thì có thể thay đổi nội dung video. Trong việc nhận diện hành động, nếu RNN được cung cấp thông tin từ các hành động trước đó, nó có xu hướng tạo ra các phản hồi đúng về ngôn ngữ ký hiệu đó hơn. Cách tiếp cận này làm cho mạng nơ-ron tuần hoàn phù hợp với nhận diện ngôn ngữ ký hiệu vì ngôn ngữ ký hiệu bao gồm nhiều hành động kết hợp theo trình tự để hiểu đầu vào từ video và đưa ra ý nghĩa đúng dựa vào trình tự thực hiện.

1.2.4 Long Short-Term Memory Networks (LSTM)

Mặc dù RNN được thiết kế để xử lý dữ liệu dạng chuỗi và có khả năng lưu trữ trọng số trước đó để dự đoán chuỗi tiếp theo nhưng có xu hướng mất thông tin khi chuỗi dài và không thể lưu trữ trọng số lâu dài. LSTM được giới thiệu như phương pháp cải tiến của RNN để giải quyết những hạn chế này. LSTM sử dụng các cổng để kiểm soát trọng số được truyền qua và lưu trữ trong quá trình xử lý dữ liệu chuỗi. Các cổng cho phép LSTM quyết định xem thông tin nào được giữ lại, loại bỏ và cập nhật.

- Cổng Quên (Forget Gate): Cổng quên quyết định trọng số nào sẽ được giữ lại và trọng số nào sẽ bị loại bỏ từ trạng thái lớp trước đó. Cổng này quyết định xem liệu trọng số nào là quan trọng và nên được duy trì trong thời gian dài hay không.
- Cổng Nhập (Input Gate): Cổng này quyết định trọng số mới nào sẽ được cập nhật vào lớp tiếp theo. Nó quyết định trọng số nào từ đầu vào mới sẽ được giữ lại.

- **Cổng Lấy Ra (Output Gate):** Cổng này quyết định trọng số nào từ trạng thái tế bào sẽ được đưa ra làm đầu ra của LSTM tại bước thời gian hiện tại. Nó kiểm soát trọng số được truyền từ lớp tới đầu ra.

Nhờ vào cấu trúc phức tạp và khả năng kiểm soát thông tin, LSTM đã trở thành một công cụ hữu ích trong nhiều ứng dụng thị giác máy tính, xử lý ngôn ngữ tự nhiên và dự đoán chuỗi. Đặc biệt, trong các tác vụ yêu cầu xử lý dạng chuỗi dài và giữ lại thông tin quan trọng qua nhiều lớp.

1.2.5 Gated Recurrent Units (GRUs)

Gated Recurrent Units (GRUs): được coi là một phiên bản đơn giản hóa của LSTM, mà không sử dụng lớp rõ ràng. GRU trực tiếp điều khiển trọng số được thay đổi trong lớp ẩn bằng cách sử dụng cổng đặt lại.

- **Cổng cập nhật (Update Gate):** GRU sử dụng cổng cập nhật để quyết định mức độ mà thông tin mới sẽ được cập nhật vào trạng thái ẩn. Cổng này quyết định xem thông tin mới sẽ được sử dụng (giữ nguyên) hay không, dựa trên thông tin từ thời điểm trước đó và đầu vào hiện tại.
- **Cổng đặt lại (Reset Gate):** Cổng đặt lại quyết định phần nào của trạng thái ẩn sẽ được "lãng quên" và cập nhật với thông tin mới. Điều này giúp mạng có khả năng "quên" thông tin không cần thiết từ quá khứ và tập trung vào thông tin mới quan trọng.

GRU sử dụng trạng thái lớp rõ ràng và sử dụng một cấu trúc đơn giản hóa hơn so với LSTM làm giảm độ phức tạp và số lượng tham số của mô hình. Điều này giúp GRU có hiệu suất tính toán tốt hơn so với LSTM trong các tác vụ không cần mức độ phức tạp cao.

1.3. Các thư viện hỗ trợ xây dựng ứng dụng nhận diện ngôn ngữ ký hiệu

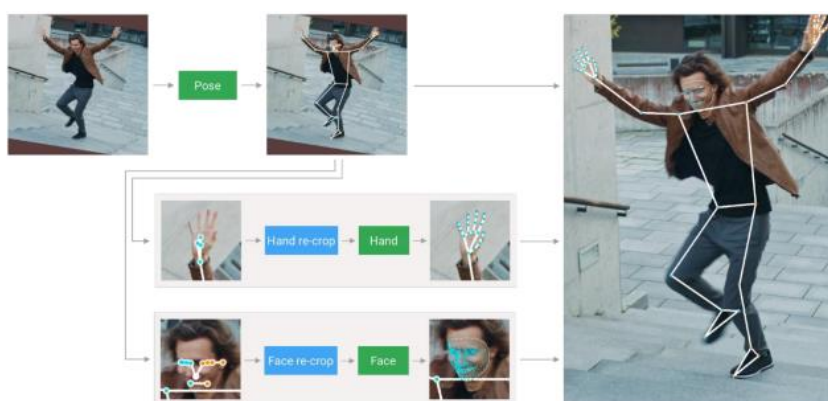
1.3.1 OpenCV

Thư viện OpenCV được thiết kế cho xử lý hình ảnh và video. OpenCV cung cấp các công cụ và chức năng đa dạng để xử lý và nhận diện hình ảnh, video và cả dữ liệu thời gian thực.

- Xử lý ảnh: OpenCV cung cấp các công cụ mạnh mẽ để xử lý ảnh, bao gồm đọc và ghi các định dạng ảnh khác nhau, biến đổi hình ảnh, lọc ảnh, phát hiện đối tượng, nhận dạng khuôn mặt, và nhiều thao tác khác.
- Thị giác máy tính: OpenCV hỗ trợ việc phát hiện đối tượng trong ảnh và video, phát hiện cạnh, tính toán đặc trưng, phân loại đối tượng và nhiều nhiệm vụ khác trong lĩnh vực thị giác máy tính.

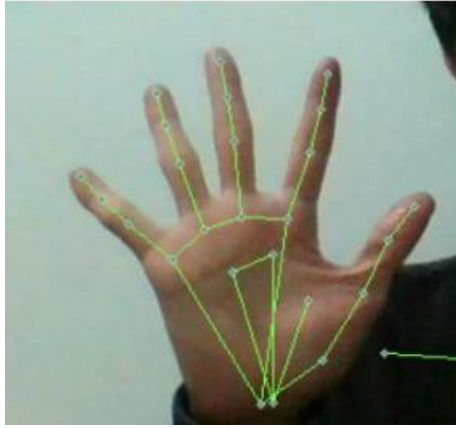
1.3.2 MediaPipe

Thư viện MediaPipe là một hệ thống giúp đồng thời nhận diện tư thế cơ thể, các điểm đặc trưng trên khuôn mặt và theo dõi động tác tay trên các thiết bị di động. Mô hình MediaPipe Holistic hoạt động bằng cách kết hợp cùng một lúc nhận diện tư thế cơ thể, các điểm đặc trưng trên khuôn mặt và theo dõi động tác tay trong thời gian thực.



Hình 2. Quy trình hình ảnh được xử lý của thư viện MediaPipe
(Nguồn: Google MediaPipe)

Mô hình MediaPipe Hands là một mô hình cung cấp khả năng theo dõi tay và ngón tay với độ chính xác cao bằng cách sử dụng học máy (Machine Learning) để suy luận 21 điểm chỉnh 3D của một bàn tay chỉ từ một khung hình duy nhất.



Hình 3. Minh họa thực tế cho các điểm đánh dấu

1.3.3 Tensorflow

Thư viện TensorFlow là một thư viện mã nguồn mở được phát triển bởi Google để xây dựng và huấn luyện các mô hình học sâu được đánh giá là thư viện học sâu phổ biến nhất và mạnh mẽ nhất. TensorFlow cung cấp một cách linh hoạt và mạnh mẽ để xây dựng, huấn luyện và triển khai các mô hình máy học, từ các mô hình đơn giản đến các mạng nơ-ron sâu phức tạp. Một trong những điểm nổi bật của TensorFlow là sự linh hoạt trong việc xây dựng các mô hình máy học, từ các mô hình hồi quy và phân loại đến các mạng nơ-ron sâu.

Hơn nữa, TensorFlow cũng cung cấp khả năng tích hợp với thư viện trực quan hóa như Matplotlib, giúp hiển thị dữ liệu một cách trực quan và dễ hiểu. Bằng cách sử dụng Matplotlib, người dùng có thể tạo ra các biểu đồ, đồ thị và hình ảnh để phân tích và hiển thị kết quả của mô hình một cách rõ ràng và sinh động.

a. Huấn luyện mô hình VGG-16 pretrained

VGG-16 là một trong những mạng CNN tiêu biểu bao gồm 16 lớp trong đó có 13 lớp tích chập 2 chiều và 3 lớp được kết nối đầy đủ. Kiến trúc của mô hình có một chồng các lớp tích chập, theo sau là các lớp tổng hợp tối đa, với độ sâu tăng dần. Thiết kế này cho phép mô hình tìm hiểu các cách biểu diễn phân cấp phức tạp của các tính năng trực quan, dẫn đến những dự đoán mạnh mẽ và chính xác. VGG-16 nổi tiếng vì sự đơn giản và hiệu quả cũng như khả năng đạt được hiệu suất mạnh mẽ

trong các tác vụ thị giác máy tính khác nhau, bao gồm phân loại hình ảnh và nhận dạng đối tượng.

b. Huấn luyện mô hình InceptionResNetV2 pretrained

Mạng InceptionResNetV2 là một trong những mô hình CNN tiên tiến và mạnh mẽ trong lĩnh vực thị giác máy tính. Nó được phát triển dựa trên ý tưởng của hai mô hình trước đó là Inception và ResNet.

Kiến trúc của InceptionResNetV2 kết hợp cả hai đặc điểm của Inception và ResNet để tạo ra một mô hình mạnh mẽ với hiệu suất cao. Mạng này bao gồm một chuỗi các module Inception được kết hợp với các kết nối rút gọn (shortcut connections) từ ResNet. Điều này cho phép mô hình học được các biểu diễn phân cấp phức tạp của các đặc trưng trong hình ảnh, đồng thời giảm thiểu hiện tượng mất mát thông tin và cải thiện tính ổn định của việc huấn luyện mạng.

Một trong những điểm nổi bật của InceptionResNetV2 là khả năng học được các biểu diễn chi tiết và sâu sắc từ hình ảnh, giúp mô hình đạt được hiệu suất cao trong các tác vụ thị giác máy tính như phân loại hình ảnh và nhận dạng đối tượng. Ngoài ra, mạng này cũng nổi tiếng với sự đa dạng và linh hoạt trong việc điều chỉnh các siêu tham số để phù hợp với nhiều loại dữ liệu và tác vụ khác nhau.

c. Huấn luyện mô hình sử dụng kiến trúc mạng LSTM và GRU

Trong đề tài này chúng tôi sử dụng kiến trúc mạng theo [11] bao gồm 3 lớp LSTM và 2 lớp GRU được cho là đạt độ nhận diện cao nhất:

Mô hình LSTM-GRU là một kiến trúc mạng nơ-ron sử dụng các lớp LSTM và GRU để học và tổng hợp thông tin từ chuỗi dữ liệu. Lớp LSTM giúp mô hình học được các mẫu dài hạn và duy trì thông tin trong thời gian dài, trong khi lớp GRU giúp giảm thiểu số lượng tham số cần huấn luyện và tăng tốc quá trình học.

Kiến trúc của mô hình bao gồm một chuỗi các lớp LSTM và GRU xen kẽ nhau. Mỗi lớp LSTM hoặc GRU sẽ nhận đầu vào từ lớp trước đó trong chuỗi và truyền dữ liệu đến lớp tiếp theo. Điều này cho phép mô hình học được các mẫu phức tạp trong chuỗi dữ liệu và tạo ra các dự đoán chính xác.

Các tham số của mô hình, bao gồm số lượng đơn vị trong mỗi lớp LSTM và GRU, có thể được điều chỉnh để phù hợp với yêu cầu cụ thể của bài toán. Sau đó, mô hình được huấn luyện thông qua việc đánh giá hiệu suất trên tập dữ liệu huấn luyện và điều chỉnh các tham số để tối ưu hóa kết quả.

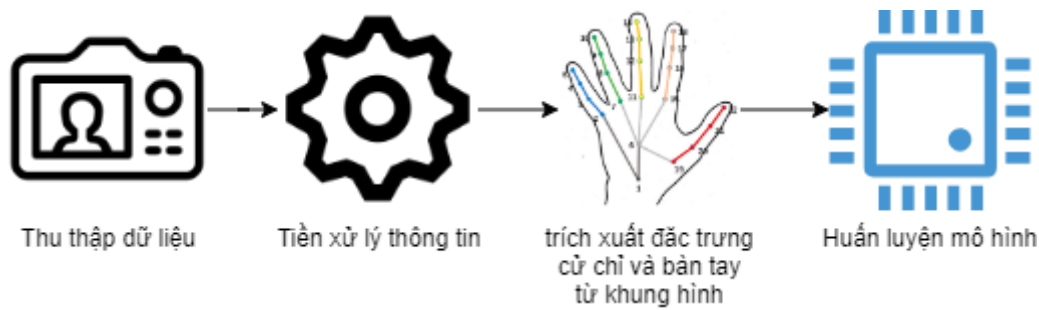
1.3.4 Kivy

Kivy là một thư viện mã nguồn mở được sử dụng để phát triển ứng dụng đa nền tảng bằng Python được thiết kế để xây dựng ứng dụng có giao diện người dùng trên nhiều nền tảng như Windows, IOS và Android. Với Kivy, có thể tạo ra các ứng dụng chạy trên nhiều hệ điều hành và thiết bị khác nhau, từ máy tính đến điện thoại di động và máy tính bảng.

Một trong những điểm mạnh của Kivy là khả năng xây dựng giao diện người dùng đẹp mắt và linh hoạt, cho phép tạo ra các ứng dụng có trải nghiệm người dùng tốt. Kivy cung cấp các thành phần giao diện đa dạng như các nút, ô nhập liệu, danh sách và nhiều hơn nữa, giúp dễ dàng tạo ra các ứng dụng phức tạp

CHƯƠNG 2. XÂY DỰNG MÔ HÌNH

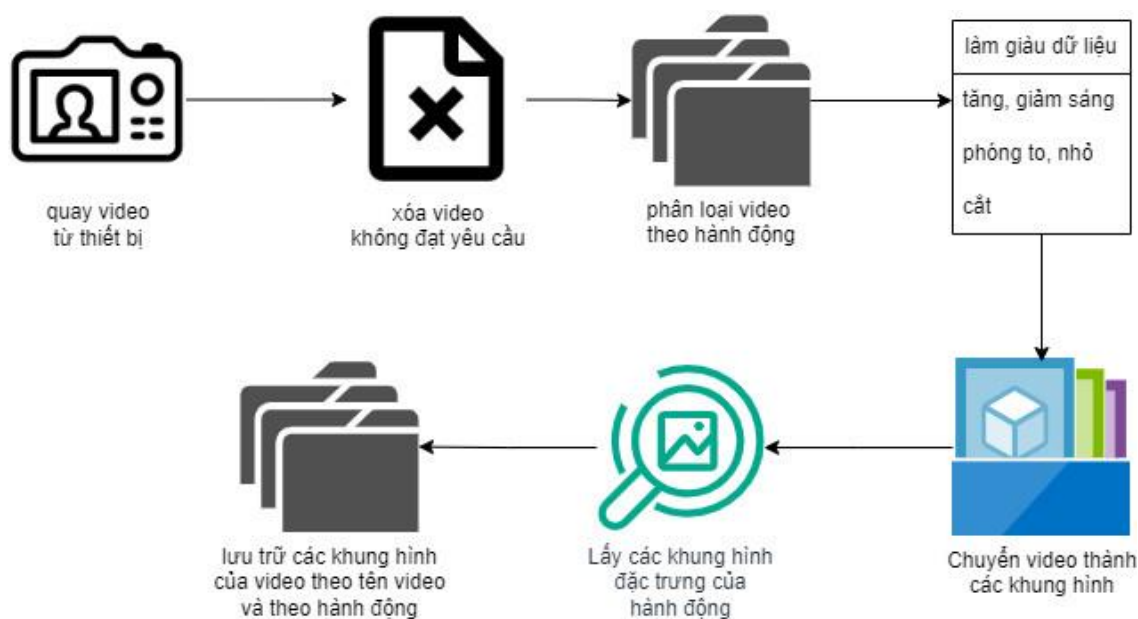
Tương tự như phát hiện cử chỉ và bàn tay, hiện nay có rất nhiều mô hình và thuật toán cho việc nhận diện ngôn ngữ ký hiệu. Để có thể nhận diện tốt ngôn ngữ ký hiệu, ta cần mô hình trích xuất hiệu quả được các điểm đặc trưng trên cơ thể và bàn tay cho nhiều người khác nhau. Đồng thời cũng nên kèm theo các quá trình tiền xử lý để cải thiện khả năng nhận diện cử chỉ và bàn tay chính xác của mô hình. Quy trình xây dựng và huấn luyện mô hình được bắt đầu từ công việc thu thập dữ liệu để huấn luyện, các dữ liệu thô được thu thập sẽ trải qua các bước tiền xử lý. Các thuật toán, mô hình Máy học sẽ được xây dựng và triển khai, việc huấn luyện mô hình và đánh giá mô hình sẽ được lặp đi lặp lại để điều chỉnh mô hình đến khi có được độ chính xác cao nhất trước khi kết xuất mô hình để sử dụng. Quy trình được minh họa



Hình 4. Quy trình huấn luyện mô hình

2.1 Tập dữ liệu huấn luyện

Việc thu thập và tiền xử lý hình ảnh là một bước quan trọng nhất trong quá trình huấn luyện mô hình dự đoán hành động ngôn ngữ ký hiệu. Việc thu thập dữ liệu được thực hiện bằng nhiều phương pháp kết hợp với các thiết bị khác nhau. Quá trình phát triển của hệ thống nhận diện ngôn ngữ ký hiệu như sau:



. Hình 5. Quy trình tạo tệp dữ liệu

Dữ liệu bao gồm 1001 video của 11 từ: “Xin chào”, “Ăn uống”, “Tạm biệt”, “Cảm ơn”, “Mẹ”, “Ngủ”, “Lặp lại”, “Nguy hiểm”, “Thêm”, “Xin lỗi”, “Tắm” được quay bằng thiết bị Samsung và Iphone với độ phân giải 1080x1920 (Full HD theo chiều dọc của màn hình) và tốc độ khung hình 30fps.

STT	Tên hành động	Hình ảnh
1	Xin chào	

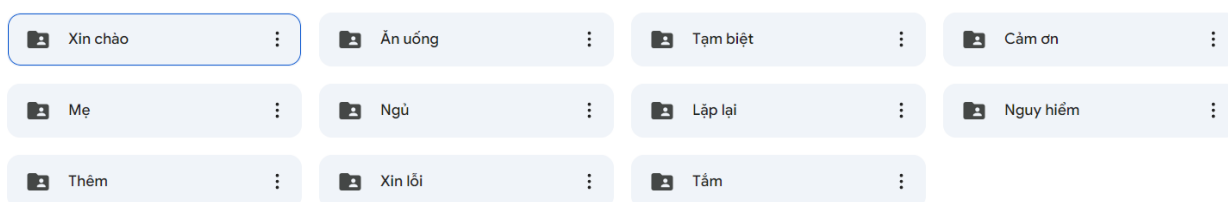
2	Ăn uống	
3	Tạm biệt	
4	Cảm ơn	
5	Mẹ	
6	Ngủ	

7	Lặp lại	
8	Nguy hiểm	
9	Thêm	
10	Xin lỗi	
11	Tắm	

Bảng 1. Thống kê hình ảnh của tập dữ liệu

2.2 Tiền xử lý dữ liệu

Mỗi hành động tương ứng với một thư mục, mỗi thư mục chứa video các ngôn ngữ ký hiệu cùng ý nghĩa tương ứng, các video đã được thay đổi kích thước ảnh nhỏ xuống còn 180x320 điểm ảnh để giảm thời gian huấn luyện, mỗi tệp video sau khi thay đổi kích thước sẽ có dung lượng khá bé trong khoảng từ 50 đến 100 MB (giảm 40 - 60 lần)



Hình 6. Tổ chức tập dữ liệu

Toán bộ dữ liệu này được chia ra làm 2 tập dữ liệu như sau:

Tập dữ liệu huấn luyện (training dataset) - 80%: Đây là dữ liệu đầu vào (*Input data*), dùng để điều chỉnh trọng số (*weight*) trong mô hình.

Tập dữ liệu thực nghiệm (test dataset) - 20%: Được sử dụng lần cuối, dùng để kiểm tra, đánh giá mức độ chính xác và mức độ sai lệch của mô hình đã được huấn luyện.

Tiếp đến là công đoạn làm giàu dữ liệu, được thực hiện bằng cách sinh ra nhiều dữ liệu hơn từ những dữ liệu sẵn có. Cụ thể hơn, ở đây chúng ta sẽ sinh ra nhiều hình ảnh hơn từ những khung hình đã được xử lý từ video, bằng các phương pháp làm mờ, phóng to..



Hình 7. Phương pháp làm mờ, phóng to, thay đổi hướng bàn tay

Dưới đây là số liệu thống kê số lượng video từ các phương pháp làm giàu dữ liệu:

Tên nhãn của video	Số lượng video gốc	Số lượng video sau khi làm giàu
Xin chào	80	240
Ăn uống	93	279
Tạm biệt	95	285
Cảm ơn	96	288
Mẹ	78	234
Ngủ	96	288

Lặp lại	95	285
Nguy hiểm	90	270
Thêm	94	282
Xin lỗi	94	282
Tắm	90	270
Tổng cộng	1001	3003

Bảng 2. Thống kê số lượng video trong tập dữ liệu

Với mỗi hình video sẽ sinh ra 3 video khác, như vậy với 1001 video của tập dữ liệu ban đầu, có thể tạo ra tập dữ liệu mới với 3003 video. Tuy nhiên, tập dữ liệu càng lớn sẽ khiến thời gian huấn luyện càng lâu và tiêu tốn nhiều bộ nhớ RAM hơn, tiến hành huấn luyện mô hình quá lớn có thể gây tràn bộ nhớ RAM khiến quá trình học gặp lỗi và dừng lại ngay lập tức.

Công đoạn *Chuẩn hóa ảnh (Image Normalization)* ảnh cũng rất quan trọng, quá trình này thực hiện chuyển đổi màu sắc ảnh đầu vào khi huấn luyện, và cả video được kiểm thử về cùng một giá trị màu đã định. Màu sắc hiển thị của hình ảnh kỹ thuật số được quyết định bởi giá trị của 3 màu chính: *đỏ (red)*, *xanh lá (green)* và *xanh dương (blue)*, gọi là hệ màu RGB.

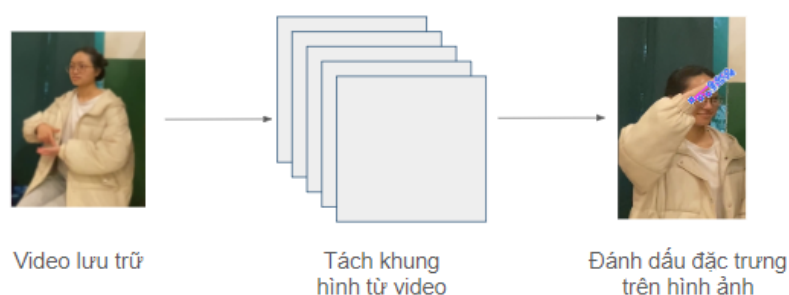
Các giá trị này giao động trong khoảng từ 0 đến 255, kết hợp với nhau tạo ra . Khi tiến hành chuẩn hóa hình ảnh sẽ giảm bớt các thông tin đặc trưng, đồng thời khiến cho màu sắc của ảnh khi huấn luyện và ảnh khi dự đoán về chung một vùng giá trị.

Công đoạn này giúp việc học và dự đoán của mô hình hiệu quả hơn đáng kể. Nhóm cũng đã tiến hành thực hiện huấn luyện và đánh giá thực nghiệm điều này, kết quả thu được

là độ chính xác của việc trích xuất đặc trưng trên hình ảnh có chuẩn hóa cao hơn khi chưa chuẩn hóa.

2.3 Trích xuất đặc trưng hành động

Sau khi thu thập được video về ngôn ngữ dữ liệu, video được tách thành các khung hình với số lượng là 60 khung hình cho video 2s có định dạng là 1080 x 980 với 29,9 FPS. Sau khi tách khung hình cần đánh dấu đặc trưng trên hình ảnh tùy nhiên với số lượng khung hình nhiều không nhất thiết phải lấy toàn bộ khung hình của tất cả video, trong trường hợp này chỉ lấy những hình ảnh thể hiện đặc điểm đặc trưng nhất của ngôn ngữ ký hiệu. Việc chỉ lấy những hình ảnh đặc trưng nhất giúp tăng khả năng nhận diện của mô hình lên đồng thời giảm tỷ lệ Overfitting với những thông tin không cần thiết trong hình ảnh.



Hình 8. Tiền xử lý khung hình

2.4. Tiến hành huấn luyện mô hình

Sau khi chuẩn bị đầy đủ về cơ sở hạ tầng và tệp dữ liệu đầu vào, ta tiến hành quá trình huấn luyện mô hình. Quá trình huấn luyện sẽ đưa vào 3 mô hình bao gồm InceptionResNetV2 (Pretrain), VGG16 (Pretrain) đã được tiền huấn luyện để tạo ra mô hình mới. Việc tận dụng một mô hình thuộc loại CNN đã được huấn luyện trên tập dữ liệu ImageNet với hàng triệu ảnh sẽ khiến quá trình huấn luyện diễn ra nhanh hơn và thường có kết quả tốt hơn. Ta tận dụng 1 phần hoặc toàn bộ các lớp, thêm/sửa/xóa 1 vài lớp để tạo ra 1 mô hình mới, quá trình này được gọi là *Fine-tuning*. Đồng thời xây dựng mô hình tự thiết kế từ các lớp LSTM và GRU.

Quá trình huấn luyện bao gồm :

Thực hiện huấn luyện lặp đi lặp lại (Training loop)

Trong quá trình huấn luyện mô hình, việc huấn luyện sẽ được lặp đi lặp lại nhiều lần trên tập dữ liệu đầu vào cho trước. Số lượng vòng lặp được quy định trong các siêu tham số khi huấn luyện, tham số quy định số lượng vòng lặp ở đây là epoch. Khởi mả công việc cho mỗi lần lặp như sau:

1. Lặp lại từng epoch. Mỗi epoch lại duyệt qua các dữ liệu trong bộ dữ liệu.
2. Với mỗi epoch như vậy, mô hình sẽ lặp lại qua từng hình ảnh trong bộ dữ liệu, rút trích ra các thông số đặc trưng (x) và nhãn, tức là đáp án tên của hành động (y).
3. Sử dụng các thông số đó của ảnh để so sánh với nhãn tương ứng. Đo lường tính dự đoán không chính xác, sử dụng số liệu này để tính toán tỷ lệ dự đoán đúng của mô hình.
4. Sử dụng một hàm tối ưu hóa (optimizer) để cập nhật các biến của mô hình.
5. Xuất ra một số các số liệu thống kê để dễ dàng hình dung.
6. Lặp lại các bước trên cho từng epoch.

Việc chọn đúng giá trị số lần lặp là cả kinh nghiệm lẫn thử nghiệm. Số lượng epoch quá ít sẽ khiến quá trình học không đủ, ngược lại quá nhiều có thể khiến quá trình huấn luyện diễn ra lâu. Nhóm lựa chọn số lần lặp là 1000 lần (epoch = 1000)

Quá trình huấn luyện mô hình sẽ mất nhiều hay ít thời gian phụ thuộc vào nhiều yếu tố như: kích thước tập huấn luyện và đánh giá, số vòng lặp, số lớp ẩn, kiến trúc sử dụng ...

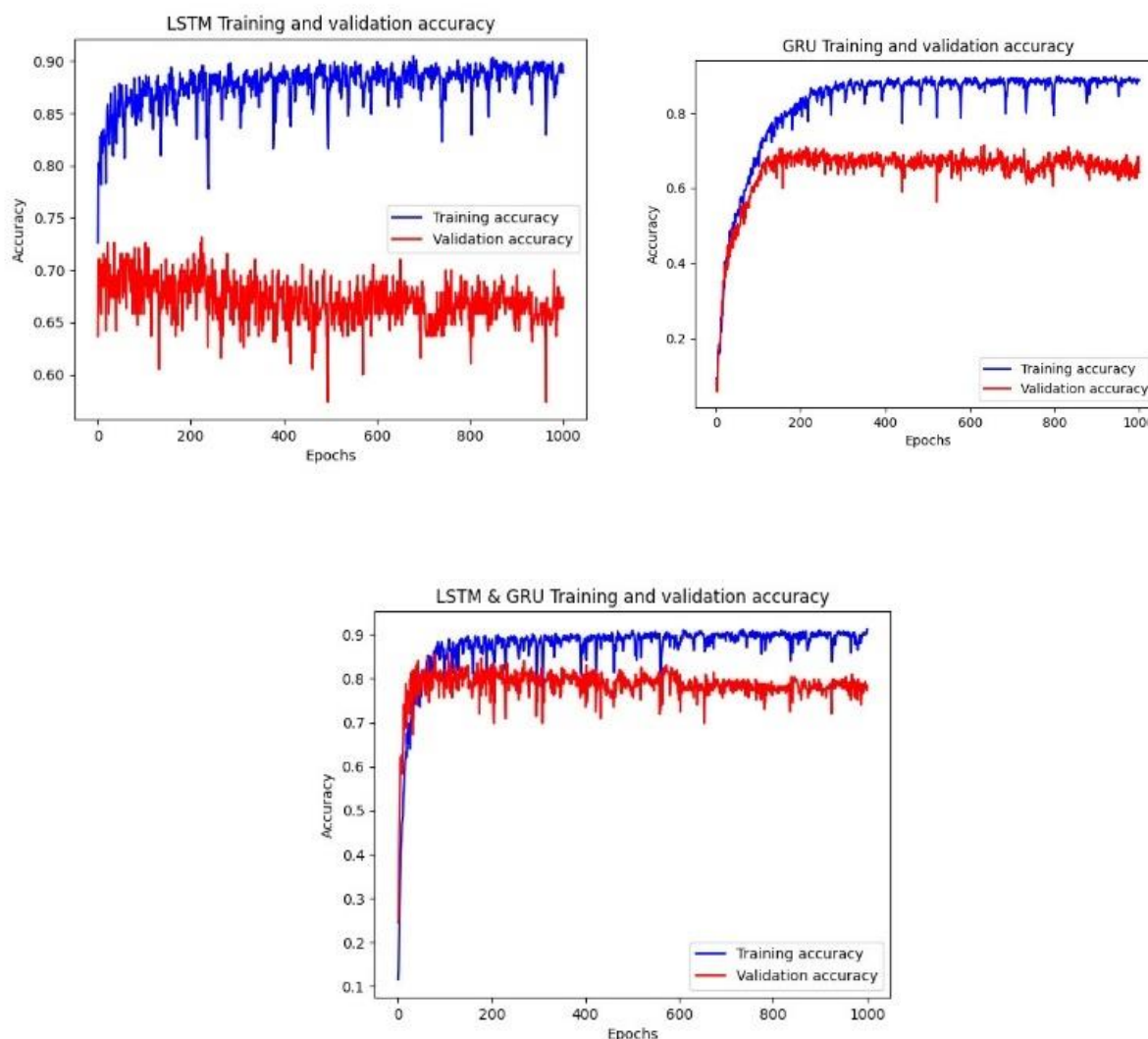
2.5. Kết quả đánh giá và so sánh các mô hình

Dựa vào bài báo [6] và [11], chúng tôi thực hiện ứng dụng mô hình đối với tập dữ liệu huấn luyện là Tiếng Việt với các lớp không thay đổi là ba lớp LSTM và hai lớp GRU. Đối với mô hình tự tạo chúng tôi thực hiện việc so sánh giữa các lớp như LSTM, GRU, LSTM + GRU.

Tập dữ liệu ban đầu với 3003 ảnh sẽ được chia ra làm tập:

Từ các tập dữ liệu chia ra để chia ra làm 2 tập: tập huấn luyện (80%) và tập đánh giá (20%), sau đó tiến hành huấn luyện.

Đối với mô hình tự tạo sau khi được huấn luyện được đánh giá hiệu suất thông qua chỉ số tỷ lệ dự đoán đúng của mô hình, đồng thời được triển khai vào thực tế nhằm đánh giá mô hình nào có khả năng phân lớp hành động tốt nhất. Dưới đây là biểu đồ thể hiện tỷ lệ dự đoán chính xác hành động của mô hình tự tạo có các lớp LSTM, GRU và LSTM + GRU



Hình 9. Biểu đồ thể hiện tỷ lệ dự đoán chính xác giữa các mô hình

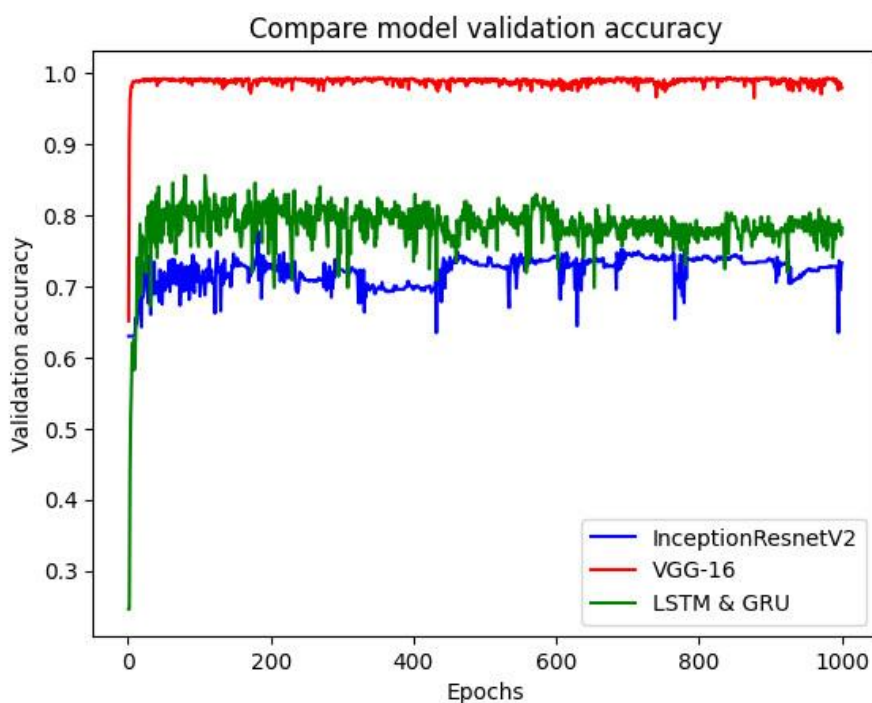
Trong phần này sẽ trình bày các kết quả đánh giá mô hình của các kiến trúc VGG16, InceptionResNetV2 và mô hình tự tạo theo [6] sau khi huấn luyện. Các mô hình này sử

dụng tập dữ liệu và các bước tiền xử lý dữ liệu giống nhau và được huấn luyện trên cùng một môi trường với cùng cấu hình để công bằng trong việc đánh giá các mô hình.

Dựa vào kết quả kiểm tra thử nhất, đúc kết được các thông tin sau về các mô hình huấn luyện được như sau:

- **Mô hình VGG16** có thời gian huấn luyện trung bình lâu nhất (3 tiếng), còn có **mô hình từ lớp LSTM + GRU** thời gian huấn luyện trung bình nhanh nhất (30 phút). Về độ chính xác, **mô hình InceptionResNetV2** có độ dự đoán chính xác thấp nhất (72,6%), còn có **mô hình VGG16** độ dự đoán chính xác cao nhất (99.57%).

Dưới đây là biểu đồ so sánh tỷ lệ dự đoán chính xác giữa các mô hình:



Hình 10. Biểu đồ so sánh giữa các mô hình

Qua quá trình huấn luyện và đánh giá mô hình chúng tôi rút ra một số đặc điểm của các kiến trúc mạng học sâu CNN và RNN như sau:

- Mô hình **VGG-16** nhìn chung được huấn luyện khá lâu, độ chính xác vẫn đảm bảo. Tuy nhiên việc triển khai thực tế lại cho thấy tỷ lệ nhận diện kém đi rất nhiều.

- Mô hình **từ lớp LSTM + GRU** có thời gian huấn luyện rất nhanh, độ chính xác cao, phù hợp với nhiều ứng dụng thực tế cần một mô hình có thể xử lý nhanh, hiệu quả. Và phù hợp với các vấn đề liên quan tới việc xử lý theo thời gian và tuần tự như video, văn bản.
- Mô hình **InceptionResNetV2** với kiến trúc CNN có độ chính xác thấp, thời gian huấn luyện ở mức trung bình.
- Đối với mạng CNN chỉ thích hợp cho việc giải quyết các vấn đề liên quan đến không gian như hình ảnh đơn lẻ. Vì khi nhận diện chuỗi hành động thì mạng nơ ron này sẽ làm mất các thông tin về sự di chuyển và thay đổi theo thời gian.
- Đối với mạng RNN thì ngược lại, thích hợp cho giải quyết các vấn đề liên quan đến chuỗi thời gian và văn bản.

Kiến trúc mạng	Tỷ lệ nhận diện	Thời gian huấn luyện
VGG16	99.57%	3 tiếng
LSTM+GRU	80,2%	30 phút
InceptionResNetV2	72,6%	1 tiếng 30 phút
GRU	65,26%	17 phút
LSTM	63,68%	16 phút

Bảng 3. Thống kê thông tin mô hình

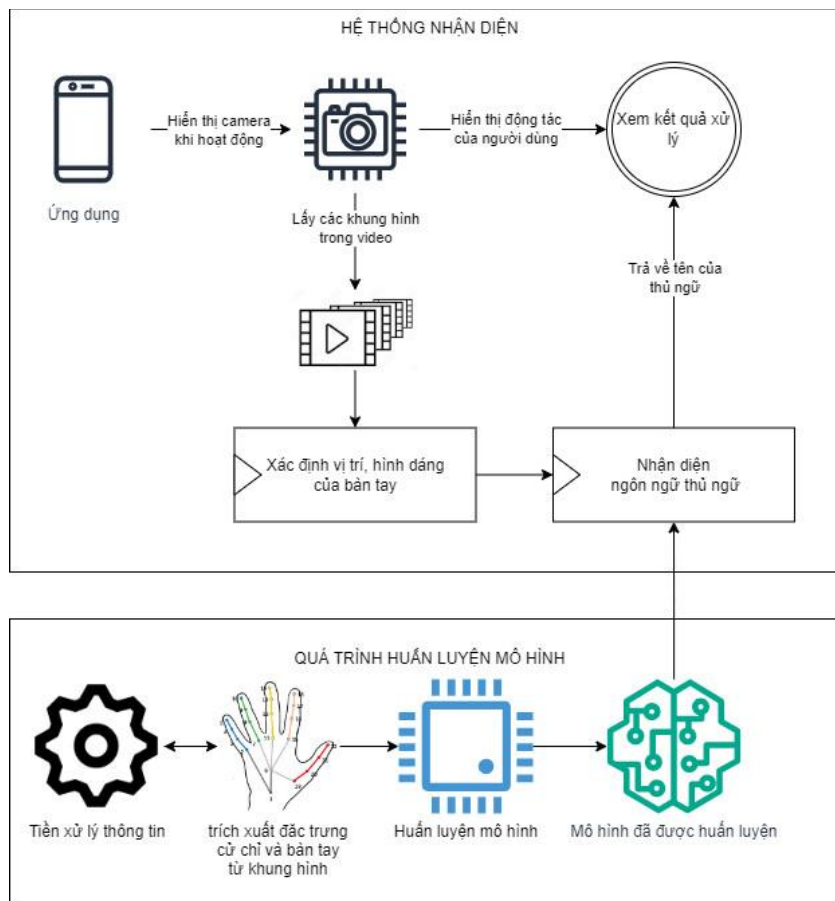
CHƯƠNG 3. TRIỂN KHAI CHƯƠNG TRÌNH

Hệ thống nhận diện cử chỉ và bàn tay có thể áp dụng trong nhiều trường hợp thực tiễn như điều khiển camera, ứng dụng, robot,... Trong bài báo cáo sẽ thiết kế một ứng dụng nhận diện ngôn ngữ ký hiệu đơn giản và triển khai chương trình lên trên thiết bị di động.

3.1 Phương pháp phát triển ứng dụng và công nghệ

3.1.1 Phương pháp phát triển

Như vậy, ta đã xây dựng được một mô hình về nhận diện ngôn ngữ ký hiệu. Vấn đề lúc này là cần phải triển khai ứng dụng có giao diện và thực hiện được những chức năng mà mô hình có. Với hệ thống nhận diện như sau:



Hình 11. Hệ thống nhận diện ngôn ngữ ký hiệu

Để giải quyết vấn đề này ta có hai cách:

- Cách thứ nhất: Lưu trữ toàn bộ mô hình đã được huấn luyện lên thiết bị của người dùng. Với cách này thì ưu điểm là nhanh, độ chính xác cao, dễ thực hiện nhưng nhược điểm là dung lượng thiết bị của người dùng, không phải thiết bị nào cũng có khả năng lưu trữ cả một mô hình AI, chưa kể đến là các tài nguyên phần cứng của thiết bị phải đủ tốt để chạy ứng dụng. Điều này sẽ gây trở ngại cho việc tiếp cận ứng dụng của người dùng.
- Cách thứ hai: Ta sẽ tiến hành triển khai ứng dụng theo mô hình Client – Server. Trong đó Server có nhiệm vụ tiếp nhận dữ liệu là các khung hình của ứng dụng phía Client trả về sau đó, nó xử lý dữ liệu đó qua mô hình AI, trả về kết quả cho Client. Server sẽ phải chấp nhận và xử lý được nhiều Client cùng một lúc. Ưu điểm của cách này là ứng dụng phía Client cho người dùng rất nhẹ, không yêu cầu tài nguyên phần cứng phải quá cao nhưng nhược điểm là độ trễ trả dữ liệu, yêu cầu kết nối Internet và có phần chưa đảm bảo về bảo mật. Hơn nữa, Server sẽ phải chịu gánh nặng nhiều hơn khi phải tiếp nhận quá nhiều công việc. Tuy nhiên những nhược điểm trên có thể hạn chế tối thiểu bằng cách xử lý dữ liệu truyền tốt, áp dụng các kỹ thuật giảm bit và mã hóa chúng trong quá trình truyền và với cách này người dùng sẽ dễ dàng tiếp cận ứng dụng hơn.

3.1.2 Công nghệ phát triển ứng dụng

Có rất nhiều khung thư viện để phát triển một ứng dụng mobile, trong dự án này khung thư viện được sử dụng sẽ là Kivy vì nó là một khung thư viện được xây dựng dựa trên Python là ngôn ngữ lập trình chủ yếu của mô hình nhận diện. Do đó, việc tích hợp ứng dụng và mô hình cũng sẽ dễ dàng và nhanh chóng hơn. Hơn nữa Kivy cũng là một thư viện đa nền tảng, nó có thể chạy ở hầu hết các hệ điều hành, cung cấp một kho lưu trữ các thành phần giao diện người dùng thân thiện để dùng và cơ chế xử lý sự kiện mạnh mẽ.

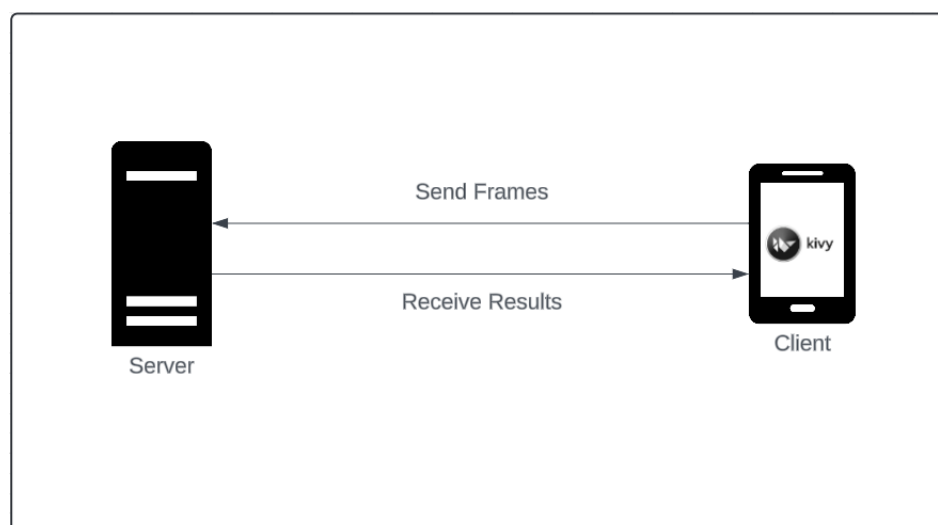
3.2 Quy trình của chương trình

3.2.1 Phân tích chung

Khi truyền dữ liệu qua mạng đặc biệt là dữ liệu về khung hình thì ta có rất nhiều phương thức để xây dựng như là RTMP, HLS, WebRTC nhưng khi phân tích sâu hơn thì các phương thức này bản chất cũng chỉ được phát triển dựa trên hai giao thức mạng phổ biến là TCP và UDP vì vậy ta có thể viết một chương trình trực tiếp sử dụng hai giao thức trên để tránh bị chậm trễ và tùy biến ứng dụng tốt hơn.

Với hai lựa chọn giao thức là TCP và UDP thì khi so sánh mỗi giao thức đều có ưu điểm và nhược điểm riêng, nếu ta lựa chọn phát triển bằng UDP thì độ trễ sẽ thấp, thông tin truyền tải sẽ nhanh hơn vì UDP không yêu cầu trả lại gói tin nhưng đổi lại dữ liệu gửi lên Server chưa chắc đã đủ để Server có thể xử lý, điều này sẽ gây nhầm lẫn thông tin và kết quả Server trả về có thể sai. Thay vào đó ta sẽ triển khai bằng TCP, dữ liệu sẽ luôn đảm bảo là gửi trọn vẹn hơn so với UDP vì TCP yêu cầu việc gửi lại gói tin khi gói tin đến được đích, nhưng tốc độ sẽ giảm tương đối so với UDP. Để xử lý vấn đề về tốc độ gửi của TCP ta sẽ dùng các kỹ thuật giảm kích thước gói tin, giảm khung hình hoặc đóng gói tin lớn và gửi chúng trong một lần chứ không phân đoạn gói tin, ta cũng có thể không yêu cầu Server phải xác nhận kết nối.

Việc gửi trả kết quả về cho Client, ta có thể sử dụng chính Socket đã mở trong việc truyền khung hình để giải quyết vấn đề này, điều này có thể đạt được thông qua việc xử lý bất đồng bộ hoặc đa luồng xử lý. Kết quả phân tích được thể hiện trong sơ đồ sau:

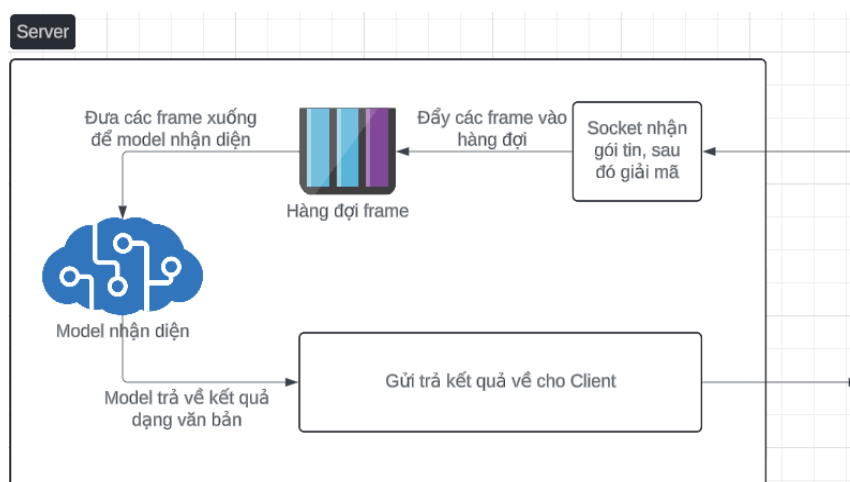


Hình 12. Hệ thống xử lý Server-Client

3.2.2 Ứng dụng phía Server

Khi khởi chạy Server, socket nhận kết nối cũng sẽ khởi tạo bằng địa chỉ host của server địa chỉ này là địa chỉ IP tĩnh, cùng với port được khởi tạo từ trước có giá trị là 9999. Server sẽ lắng nghe và chờ client kết nối.

Sau khi chấp nhận kết nối của client, server sẽ khởi tạo một luồng xử lý riêng cho client đó, sau đó quá trình truyền thông sẽ diễn ra. Server tiếp nhận gói tin và giải mã chúng thành các khung hình, các khung hình được chuyển vào một hàng đợi và các khung hình dần được đưa vào mô hình xử lý, mô hình xử lý từng khung hình và đồng thời nó cũng sẽ trả kết quả về cho client.

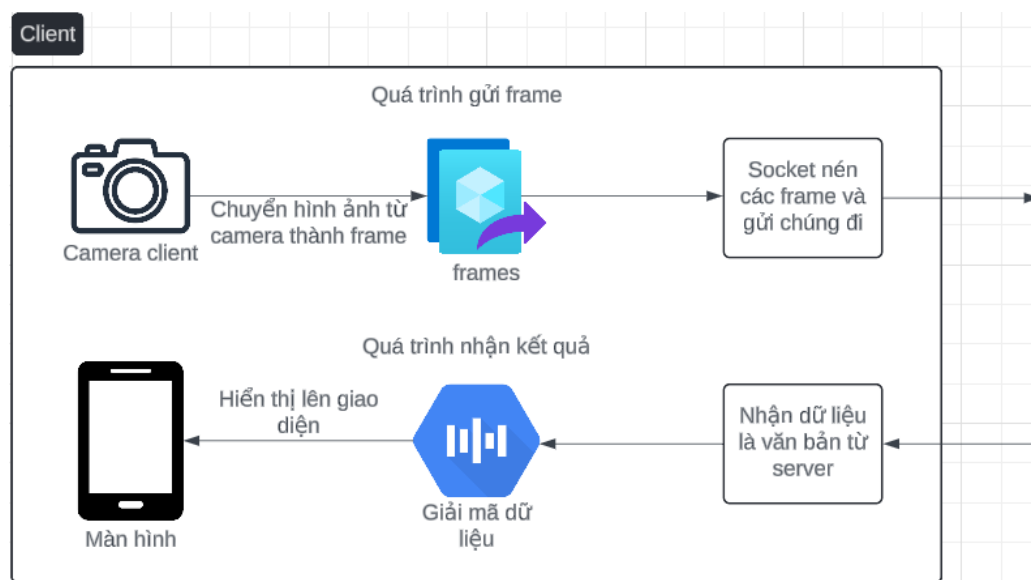


Hình 13. Xử lý chức năng phía Server

3.2.3 Ứng dụng phía Client

Trong quá trình hoạt động client sẽ thực hiện hai công việc:

- Công việc thứ nhất đó là gửi khung hình lên cho server. Quá trình này sẽ hoạt động từ việc lấy dữ liệu từ camera, chuyển dữ liệu đó thành các khung hình sau đó đóng gói chúng và gửi cho server.
- Công việc thứ hai đó là tiếp nhận kết quả trả về từ server là các kết quả nhận diện, giải mã chúng và hiển thị lên màn hình giao diện người dùng.



Hình 14. Xử lý chức năng phía Client

3.3 Kết quả triển khai

Từ kết quả chương 2.6, việc sử dụng hai mô hình VGG16 và mô hình có lớp LSTM + GRU cho việc triển khai ứng dụng là hợp lý, với số lần test là 250. Sau khi triển khai vào thực tế, chúng tôi đạt được kết quả như sau:

Mô hình VGG16 với tỷ lệ dự đoán đúng là 99.9% tuy nhiên khi vận dụng vào trong thực tế thì khả năng phân lớp hành động của mô hình không tốt. Mô hình chỉ có khả năng nhận diện ngôn ngữ ký hiệu khi thực hiện 1 ngôn ngữ ký hiệu lặp lại với tần suất nhất định.

Mô hình có lớp LSTM và GRU có khả năng nhận diện ngôn ngữ ký hiệu tốt khi vận dụng vào thực tế tuy nhiên khả năng phân lớp hành động vẫn còn nhầm lẫn vì một số ngôn ngữ ký hiệu có cử chỉ gần giống nhau. Một số hành động tương tự nhau như sau:

Tên hành động	Hình ảnh
Xin lỗi	
Thêm	
Lặp lại	

Bảng 4. Thống kê hành động dễ nhầm lẫn

Sau khi xác định được mô hình thích hợp cho việc nhận diện ngôn ngữ ký hiệu, chúng tôi thực hiện việc áp dụng mô hình vào ứng dụng trên kết nối Server-Client. Khi khởi tạo server, chương trình phía server sẽ khởi động luồng nhận kết nối và chờ cho client kết nối đến. Tuy nhiên lúc này có một vấn đề đó là khi client kết nối đến thì nó yêu cầu server phải trả dữ liệu ngay lập tức trong khi việc khởi tạo mô

hình chưa được kích hoạt. Vì vậy, sau khi chạy server ta sẽ chạy một chương trình kích hoạt mô hình trên server. Việc chạy chương trình kích hoạt chỉ cần thực hiện duy nhất một lần, khi tiến trình khởi tạo mô hình đã hoạt động thì các client sẽ kết nối đến bình thường mà không cần quan tâm đến việc này.



Hình 15. Ứng dụng nhận diện ngôn ngữ ký hiệu

CHƯƠNG 4. KẾT LUẬN

Dự án đã đạt được một số kết quả quan trọng như sau:

- **Về lý thuyết:** Chúng tôi đã khảo sát các phương pháp học máy hiện có, khảo sát các kỹ thuật đã và đang được sử dụng để xây dựng hệ thống nhận diện ngôn ngữ ký hiệu. Dựa trên kết quả khảo sát được, chúng tôi đã thực hiện đánh giá một số mô hình tiền huấn luyện dựa trên tỷ lệ dự đoán đúng được dùng cho mô hình nhận diện ngôn ngữ ký hiệu để đánh giá hiệu quả nhận diện hành động dựa trên tập dữ liệu của chúng tôi. Bên cạnh đó, chúng tôi đã đề xuất một phương pháp xây dựng ứng dụng nhận diện ngôn ngữ ký hiệu sử dụng các kỹ thuật như từ trích xuất đặc trưng từ cử chỉ và bàn tay và đề xuất áp dụng kiến trúc mạng sử dụng kết hợp LSTM + GRU để phân lớp hành động. Ngoài ra, chúng tôi cũng đã tìm hiểu được một số mô hình tiền huấn luyện được sử dụng cho việc nhận diện và phân biệt hành động như InceptionResNetV2, VGG16; đồng thời xây dựng một mô hình từ các lớp LSTM và GRU.
- **Về ứng dụng:** Chúng tôi đã xây dựng được tập dữ liệu gồm 3003 video thuộc tập 11 hành động, 15040 hành động cần trích xuất. Chúng tôi đã xây dựng thành công một ứng dụng nhận diện ngôn ngữ ký hiệu trên nền tảng Kivy, chạy và thực nghiệm với thực tế 11 hành động và cho kết quả tốt phân lớp tốt khi sử dụng mô hình kết hợp lớp LSTM và GRU. Kết quả thực nghiệm cho thấy, mô hình LSTM + GRU sử dụng MediaPipe đặt các điểm mốc trên bàn tay cho hiệu quả phân lớp hành động và trích xuất đặc trưng tốt nhất trong thực tế đạt 80,2%. Ứng dụng đã được triển khai trên các thiết bị di động như IOS, Android để phiên dịch ý nghĩa 11 từ thủ ngữ.

Bên cạnh đó, đề tài cũng không tránh khỏi một số hạn chế do gặp nhiều yếu tố khách quan như: số lượng video tập huấn luyện và khung hình cung cấp còn ít, một số ngôn ngữ ký hiệu có cử chỉ gần giống nhau như “Xin lỗi”, “Thêm” và “Lặp lại”, chưa triển khai chức năng thông báo về ý nghĩa của ngôn ngữ ký hiệu.

Phần tiếp theo của đề tài, chúng tôi dự định tăng kích thước tập dữ liệu huấn luyện đồng thời chuẩn hóa dữ liệu để có được kết quả đánh giá chính xác hơn về hiệu suất phân lớp hành động và trích xuất đặc trưng của các mô hình tiền huấn luyện từ ImageNet do tập dữ liệu hiện tại chỉ vận dụng trên các chữ riêng lẻ của thủ ngữ. Muốn triển khai thực tế, chúng tôi cần tăng số lượng chữ, tăng thông tin cần cung cấp cho người dùng. Bên cạnh, để đảm bảo độ chính xác cao cho ứng dụng, chúng tôi có thể đề xuất xem xét phương pháp sử dụng MediaPipe và mô hình từ các lớp LSTM kết hợp lớp GRU cho nhận diện ngôn ngữ ký hiệu bằng việc lấy khung hình trong video theo thời gian thực. Thêm vào đó, chúng tôi sẽ phát triển hiệu suất nhận diện của mô hình để có thể nhận diện nhiều từ liên tục từ đó có thể chuyển thành 1 câu hoàn chỉnh.

DANH MỤC THAM KHẢO

- [1] Sở Khoa học và công nghệ TP.HCM. (2023). *CESTI*. Truy cập ngày 16/11/2023. Từ: <http://thongke.cesti.gov.vn/phan-tich-thong-ke/ket-qua-thong-ke/1138-ho-tro-tang-cuong-phuc-hoi-chuc-nang-khiem-thinh-khiem-thi-p1>
- [2] Thiết bị giao tiếp thông minh dành cho người khiếm thanh, khiếm thính (Speak Your Mind - SYM) (2021). *Đại học Quốc gia Bách khoa HCM*. Truy cập ngày 17/11/2023. Từ <https://www.lib.hcmut.edu.vn/chuyen-doi-thu-ngu-sang-giong-noi-va-van-ban>
- [3]Deng, L., Bi, L., Li, H. et al.(2023). "*Lightweight aerial image object detection algorithm based on improved YOLOv5s*",. Sci Rep 13, 7817. DOI:10.1038/s41598-023-34892-4.
- [4] Khanh, V. H., & Khang, P. N. (2017). *NHẬN DẠNG NGÔN NGỮ DẤU HIỆU SỬ DỤNG CẢM BIẾN KINECT*. Đà Nẵng: Kỷ yếu Hội nghị Quốc gia lần thứ X về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR). 694-702. DOI: 10.15625/vap.2017.00082
- [5] Thành, L. M., Minh, L., et al.(2021) “Nhận dạng cử chỉ bàn tay dùng mạng Nơ-Ron chập”. Tạp Chí Khoa học Và Công nghệ - Đại học Đà Nẵng, vol 19, tr 53-56, Từ: <https://jst-ud.vn/jst-ud/article/view/7427>.
- [6] V. H. Iyer, U. M. Prakash, A. Vijay and P. Sathishkumar, (2022). "*Sign Language Detection using Action Recognition*",. Greater Noida, India, 2022,; 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). doi:10.1109/ICACITE53722.2022.9823484.
- [7] Mhatre, S., S. Joshi & Kulkarni H. B., (2022). Sign Language Detection using LSTM.*IEEE International Conference on Current Development in Engineering and Technology(CCET)*, Bhopal, India, 2022, pp.1-6, doi: 10.1109/CCET56606.2022.10080705

- [8] Pei, G., & Xu, C., (2022). Hand Detection and Gesture Classification based on Trasformer. *Center for Science Technology and Information (CESTI)*, 2022 China Automation Congress (CAC), Xiamen, China, 2022, pp. 276-281, doi: 10.1109/CAC57257.2022.10055588. IEEE Xplore.
- [9] Prof.Radha, S., Shirbhate, Vedant, M., Shinde, D., Ms.Sanam, A., Metkari, Pooja, M., Borkar, U., Ms.Mayuri, A., & Khandge (2020). Sign language Recognition Using Machine Learning Algorithm.. *International Research Journal of Engineering and Technology (IRJET)*. doi:<https://doi.org/10.1007/s42979-021-00592-x>
- [10] Chen, R. C., Manongga, E, W., Dewi, C., & Chen, -Y, H., "Automatic Digit Hand Sign Detection With Hand Landmark," 2022 International Conference on Machine Learning and Cybernetics (ICMLC), Japan, 2022, pp. 6-11, doi: 10.1109/ICMLC56445.2022.9941325.
- [11] Deep Kothadiya, C.Bhatt, K.Sapariya, K.Patel, A. Gill-Gonzalez, J. M.Cochardo. (2022).Deepsign: Sign Language Detection and Recognition Using Deep Learning. *Electronics* 2022.DOI:<https://doi.org/10.3390/electronics11111780>
- [12] Chauhan, K., et al., "Automated Machine Learning: The New Wave of Machine Learning," 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, 2020, pp. 205-212, doi: 10.1109/ICIMIA48430.2020.9074859.
- [13] M. Awad and R. Khanna, *Efficient learning machines: Theories, concepts, and applications for engineers and system designers*, no. April. 2015.
- [14] Mitchell, R, K., Agle, B, R., & Wood, J, D., "Toward a Theory of Stakeholder Identification and Salience: Defining the Principle of Who and What Really Counts," *Acad. Manag. Rev.*, vol. 22, no. 4, pp. 853–886, May 1997, doi: 10.2307/259247.

- [15] BBCNews, “Artificial intelligence: Google’s AlphaGo beats Go master Lee Se-dol,” 2016. <https://www.bbc.com/news/technology-35785875>.
- [16] Charu C. Aggarwal (2018) .Neural Networks and Deep Learning. DOI: <https://doi.org/10.1007/978-3-319-94463-0>.
- [17] Sharma, A, K., & Kim, K, K., "A Lightweight Detector for Small Objects," 2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS), Incheon, Korea, Republic of, 2022, pp. 360-361,