**Shape, square

Description automatically generated**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TR** **ƯỜNG ĐẠI HỌC SƯ PHẠM KĨ THUẬT TPHCM**

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO CUỐI KỲ**

**LẬP TRÌNH DI ĐỘNG**

**Đề tài:**

**ỨNG DỤNG MACHINE LEARNING TRONG XÂY DỰNG ỨNG DỤNG ANDROID**

**GVHD: Th.S Trương Thị Ngọc Phượng**

**NHÓM SINH VIÊN THỰC HIỆN:**

**1. Nguyễn Gia Huy 20110103**

**2. Vũ Trung Hiếu 20110482**

**3. Lương Nhất Phi 20110547**

**Tp. Hồ Chí Minh, Tháng 6 Năm 2023**

**NHẬN XÉT CỦA GIẢNG VIÊN**

Tp Hồ Chí Minh, ngày … tháng 6 năm 2023

**(**Giảng viên hưỡng dẫn)

**Th.S Trương Thị Ngọc Phượng**

***LỜI CẢM ƠN***

*Kính gửi Ths. Trương Thị Ngọc Phượng*

*Chúng em muốn bày tỏ lòng biết ơn sâu sắc đến cô vì sự hướng dẫn, truyền cảm hứng và giúp đỡ trong suốt quá trình học tập của mình. Trong năm qua, cô đã dành cho chúng em rất nhiều thời gian và công sức để giúp Chúng em tiếp thu những kiến thức và kỹ năng cần thiết trong lĩnh vực mà chúng em đang theo đuổi.*

*Chúng em rất trân trọng tinh thần tận tâm, chu đáo và nhiệt tình của cô trong việc giảng dạy và truyền đạt kiến thức cho chúng em. Các bài giảng, tài liệu và hoạt động giảng dạy được cô chuẩn bị với sự cẩn trọng và chuyên nghiệp, đã giúp chúng em tiếp thu và ứng dụng kiến thức một cách hiệu quả.*

*Đặc biệt, chúng em rất cảm kích sự động viên và hỗ trợ của cô khi chúng em gặp khó khăn trong quá trình học tập. Cô đã luôn sẵn sàng trả lời các câu hỏi của chúng em, hướng dẫn và giúp chúng em vượt qua các thử thách để đạt được mục tiêu học tập của mình.*

*Những kinh nghiệm, kỹ năng và kiến thức mà chúng em học được từ cô sẽ là tài sản vô giá của mình trong sự nghiệp và cuộc sống. Chúng em sẽ luôn cảm nhận và trân trọng những gì cô đã truyền đạt và tôn trọng công lao của cô.*

*Một lần nữa, chúng em xin gửi đến cô lời cảm ơn chân thành nhất vì những gì cô đã dành cho chúng em trong suốt quá trình học tập. Chúng em hy vọng sẽ có cơ hội tiếp tục được học tập và phát triển trong môi trường học tập tuyệt vời do cô tạo ra.*

**MỤC LỤC**

[CHƯƠNG 1. MỞ ĐẦU 1](#_Toc136758798)

[**1. Phạm vi và đối tượng 1**](#_Toc136758799)

[**2. Mô tả bài toán 1**](#_Toc136758800)

[**3. Mục đích thực hiện bài toán và yêu cầu 2**](#_Toc136758801)

[**3.1. Mục đích thực hiện 2**](#_Toc136758802)

[**3.2. Yêu cầu bài toán 3**](#_Toc136758803)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 4](#_Toc136758804)

[**1. Công cụ và môi trường để lập trình 4**](#_Toc136758805)

[**2. Ngôn ngữ và các thư viện 5**](#_Toc136758806)

[**2.1. Ngôn ngữ 5**](#_Toc136758807)

[**2.2. Các thư viện được sử dụng 5**](#_Toc136758808)

[CHƯƠNG 3. THIẾT KẾ ỨNG DỤNG 8](#_Toc136758809)

[**1. Thiết kế và xây dựng giao diện của ứng dụng 8**](#_Toc136758810)

[**1.1. Nhận diện ngôn ngữ ký hiệu 8**](#_Toc136758811)

[**1.2. Nhận diện cảm xúc khuôn mặt 9**](#_Toc136758812)

[**1.3. Nhận diện số viết tay 11**](#_Toc136758813)

[**2. Xây dựng Logic cho ứng dụng 13**](#_Toc136758814)

[**2.1. Chuẩn bị các Model 13**](#_Toc136758815)

[**2.2. Chuẩn bị các thư viện 13**](#_Toc136758816)

[**2.3. Lấy dữ liệu đầu vào 14**](#_Toc136758817)

[**2.4. Xử lý dữ liệu 15**](#_Toc136758818)

[CHƯƠNG 4. KẾT LUẬN 17](#_Toc136758819)

[**1. Kết quả đạt được 17**](#_Toc136758820)

[**1.1. Kiến thức 17**](#_Toc136758821)

[**1.2. Kỹ năng 17**](#_Toc136758822)

[**1.3. Sản phẩm 17**](#_Toc136758823)

[**2. Ưu điểm 17**](#_Toc136758824)

[**3. Nhược điểm 18**](#_Toc136758825)

[**4. Hướng phát triển 18**](#_Toc136758826)

[TÀI LIỆU THAM KHẢO 19](#_Toc136758827)

**MỤC LỤC HÌNH**

[Hình 1: Hình thiết kế chức năng nhận diện ngôn ngữ ký hiệu 8](#_Toc136758786)

[Hình 2: Hình chức năng nhận diện ngôn ngữ ký hiệu 9](#_Toc136758787)

[Hình 3: Hình thiết kế chức năng nhận diện cảm xúc khuôn mặt 10](#_Toc136758788)

[Hình 4: : Hình thiết kế chức năng nhận diện cảm xúc khuôn mặt 11](#_Toc136758789)

[Hình 5: Hình thiết kế chức năng nhận diện số viết tay 12](#_Toc136758790)

[Hình 6: Hình chức năng nhận diện số viết tay 13](#_Toc136758791)

[Hình 7: Hình khai báo các thư viện được sử dụng 14](#_Toc136758792)

[Hình 8: Hình khai báo quyền sử dụng Camera 14](#_Toc136758793)

[Hình 9: Hình đoạn code yêu cầu quyền sử dụng Camera 15](#_Toc136758794)

[Hình 10: Hình đoạn code khởi tạo Detector 15](#_Toc136758795)

[Hình 11: Hình đoạn code xử lý ảnh 16](#_Toc136758796)

1. MỞ ĐẦU
   1. Phạm vi và đối tượng

Phạm vi của bài toán là xây dựng một ứng dụng android có khả năng nhận diện ngôn ngữ ký hiệu (ASL - American Sign Language) và nhận diện cảm xúc của khuôn mặt bằng camera của thiết bị. Bên cạnh đó ứng dụng cũng có thể nhận diện được số mà người dùng viết.

Ứng dụng sẽ nhận diện hình ảnh bằng Camera của thiết bị, sau đó ứng dụng sẽ phân tích và trả kết quả trực tiếp lên trên hình ảnh.

Phạm vi của bài toán bao gồm các bước chính sau đây:

**- Machine Learning**: Hiểu về nguyên tắc và các thuật toán Machine Learning, bao gồm học có giám sát và học không giám sát. Sử dụng các thư viện và framework phổ biến như TensorFlow, PyTorch hoặc Scikit-learn để triển khai các mô hình Machine Learning trên Android.

**- Computer Vision**: Hiểu về xử lý ảnh và khái niệm cơ bản trong Computer Vision như mô hình màu, lọc thông tin, trích xuất đặc trưng và phân đoạn ảnh. Có kiến thức về các phương pháp nhận dạng đối tượng và phát hiện khuôn mặt trong ảnh.

**- Android Development**: Có kiến thức về lập trình Android, bao gồm Java hoặc Kotlin. Hiểu về cấu trúc ứng dụng Android, giao diện người dùng và cách tương tác với các thành phần khác nhau của hệ thống Android.

* 1. Mô tả bài toán

Bài toán thuộc lĩnh vực học máy, xử lý ảnh và Android và có ứng dụng rất nhiều trong nhiều trong các lĩnh vực khác nhau như: công nghệ thông tin, ngân hàng, y tế, giáo dục,…

Bài toán sẽ có ba nhiệm vụ chính:

* **Nhận diện ngôn ngữ ký hiệu:**

+ Ứng dụng sẽ có khả năng nhận dạng và giải mã các ngôn ngữ ký hiệu của người dùng.

+ Đầu vào của ứng dụng sẽ là hình ảnh nhận được từ Camera của thiết bị.

+ Ứng dụng sẽ sử dụng các Model của Tensorflow Lite để xử lý các dữ liệu đầu vào sau đó trả về kết quả.

* **Nhận diện cảm xúc trên khuôn mặt:**

+ Ứng dụng sẽ có khả năng nhận dạng được khuôn mặt của người dùng.

+ Đầu vào của ứng dụng sẽ là hình ảnh nhận được từ Camera của thiết bị.

+ Ứng dụng sẽ sử dụng thư viện Smiley Rating để xử lý các dữ liệu đầu vào sau đó trả về kết quả.Các chức năng của ứng dụng.

* **Nhận diện số viết tay:**

+ Ứng dụng có khả năng nhận diện số của người dùng.

+ Đầu vào của ứng dụng sẽ là số người dùng viết trực tiếp vào màn hình của thiết bị.

+ Ứng dụng sẽ sử dụng thư viện FingerPaintView để thu thập hình số mà người dùng viết, sau đó sẽ sử dụng các Model của Tensorflow Lite để xử lý và trả về kết quả.

* 1. Mục đích thực hiện bài toán và yêu cầu
     1. Mục đích thực hiện

Nghiên cứu và áp dụng các kỹ thuật xử lý ảnh số, học máy và lập trình Android để phát triển một ứng dụng nhận diện ngôn ngữ ký hiệu, cảm xúc khuôn mặt và số viết tay với độ chính xác cao.

Tìm hiểu về các phương pháp xử lý ảnh số và thuật toán máy học để phân tích và nhận diện ngôn ngữ ký hiệu, cảm xúc khuôn mặt và số viết tay.

Cải thiện kỹ năng và tư duy lập trình trong việc xây dựng và phát triển ứng dụng Android.

Áp dụng các kiến thức và kỹ năng học được trong môn học vào thực tiễn và ứng dụng thực tế.

Đóng góp vào việc phát triển các ứng dụng công nghệ mới với mục đích hỗ trợ con người trong các công việc liên quan đến nhận diện ngôn ngữ ký hiệu, cảm xúc khuôn mặt và số viết tay.

* + 1. Yêu cầu bài toán

Nghiên cứu và đánh giá các phương pháp xử lý ảnh số và thuật toán máy học để phân tích và nhận diện ngôn ngữ ký hiệu, cảm xúc khuôn mặt và số viết tay.

Xây dựng và phát triển một ứng dụng Android có đầy đủ các chức năng theo yêu cầu. Đồng thời các chức năng phải hoạt động ổn định và có độ chính xác cáo

1. CƠ SỞ LÝ THUYẾT
   1. Công cụ và môi trường để lập trình

Để hoàn thành được đề tài, nhóm sẽ sử dụng Android Studio. Android Studio là một môi trường phát triển tích hợp (Integrated Development Environment - IDE) mạnh mẽ và chính thức được phát triển bởi Google dành cho việc phát triển ứng dụng di động Android. Với một giao diện người dùng dễ sử dụng và tích hợp các công cụ mạnh mẽ, Android Studio là công cụ ưu tiên của các nhà phát triển Android.

Android Studio cung cấp một trình soạn thảo mã thông minh và mạnh mẽ. Với tính năng autocompletion, kiểm tra lỗi cú pháp và gợi ý mã, việc viết mã trở nên nhanh chóng và dễ dàng hơn. Ngoài ra, trình soạn thảo mã cũng hỗ trợ các tính năng như refactor, tìm kiếm và thay thế nhanh chóng, giúp tăng hiệu suất và chất lượng mã nguồn.

Android Studio cung cấp một bộ công cụ gỡ lỗi (debugger) mạnh mẽ. Với bộ công cụ này, chúng ta có thể theo dõi và kiểm tra giá trị của biến, thực hiện gỡ lỗi từng bước và xem stack trace để tìm ra và sửa lỗi. Điều này giúp chúng ta nhanh chóng và hiệu quả trong việc khắc phục các vấn đề trong mã nguồn.

Android Studio đi kèm với Android Emulator, cho phép chúng ta mô phỏng các thiết bị Android để kiểm tra và chạy ứng dụng. Chúng ta có thể kiểm tra ứng dụng trên nhiều thiết bị và các phiên bản Android khác nhau, giúp đảm bảo tính tương thích và chất lượng của ứng dụng trên nhiều môi trường khác nhau.

Ngoài ra, Android Studio cung cấp nhiều tính năng hỗ trợ khác như quản lý phiên bản, phân tích hiệu năng và tạo giao diện người dùng một cách trực quan. Tất cả những điều này giúp nhà phát triển tạo ra các ứng dụng Android chất lượng cao, tương thích rộng rãi và có hiệu suất tốt. Với sự hỗ trợ liên tục và cập nhật từ Google, Android Studio là công cụ tuyệt vời để phát triển ứng dụng di động Android.

* 1. Ngôn ngữ và các thư viện
     1. Ngôn ngữ

Nhóm chọn ngôn ngữ lập trình Java để xây dựng ứng dụng của mình. Vì các thành viên trong nhóm đã sử dụng Java để hoàn thành các bài tập trong các tuần học, cũng như Java là một ngôn ngữ hỗ trợ rất tốt cho việc xây dựng các ứng dụng Android và các lĩnh vực liên quan đến trí tuệ nhân tạo, học máy.

* + 1. Các thư viện được sử dụng
       1. TensorFlow Lite

TensorFlow Lite là một phiên bản nhẹ và tối ưu hóa của thư viện máy học TensorFlow, được phát triển bởi Google. Nó là một công cụ rất hữu ích cho việc triển khai mô hình máy học trên các thiết bị di động và nhúng, nơi tài nguyên tính toán và bộ nhớ thường có giới hạn.

Với TensorFlow Lite, nhà phát triển có thể chuyển đổi và chạy các mô hình máy học trực tiếp trên thiết bị, giúp giảm thiểu sự phụ thuộc vào kết nối mạng và tăng cường quyền riêng tư dữ liệu. Nó cung cấp các tính năng tối ưu hóa để đảm bảo hiệu suất cao và tiêu thụ năng lượng thấp, như quantization, pruning và khối đồng bộ. Các kỹ thuật này giúp giảm kích thước mô hình và tăng tốc độ thực thi, đáp ứng yêu cầu của các thiết bị di động và nhúng.

TensorFlow Lite hỗ trợ nhiều nền tảng và ngôn ngữ lập trình, bao gồm Android, iOS, Raspberry Pi và các hệ điều hành nhúng. Nó cung cấp các API dễ sử dụng để tải, chuyển đổi và chạy mô hình máy học trên các thiết bị này. Điều này cho phép nhà phát triển tận dụng sức mạnh của máy học trực tiếp trên thiết bị, giảm thiểu sự phụ thuộc vào kết nối mạng và bảo vệ dữ liệu.

* + - 1. FingerPaintView

Thư viện FingerPaintView là một thư viện mã nguồn mở dành cho Android, được sử dụng để tạo giao diện vẽ bằng ngón tay trên màn hình. Với thư viện này, nhà phát triển có thể cho phép người dùng vẽ, tô màu và tương tác trực tiếp trên màn hình bằng cách sử dụng ngón tay của họ.

Thư viện FingerPaintView cung cấp một tập hợp các tính năng và chức năng mạnh mẽ để tạo giao diện vẽ đơn giản và linh hoạt. Chúng ta có thể tùy chỉnh kích thước, màu sắc và độ trong suốt của bút vẽ, cho phép người dùng tạo ra các hiệu ứng vẽ đa dạng. Ngoài ra, chúng ta cũng có thể xóa, hoặc lưu lại bức tranh được vẽ bằng cách chụp ảnh màn hình.

Thư viện này được xây dựng dựa trên công nghệ cảm ứng đa điểm của Android, cho phép người dùng vẽ bằng nhiều ngón tay cùng một lúc. Điều này mang lại trải nghiệm vẽ mượt mà và tự nhiên, tạo điểm nhấn độc đáo cho các ứng dụng vẽ và sáng tạo.

* + - 1. Smiley Rating

Đây là một thư viện mã nguồn mở được sử dụng trong phát triển ứng dụng di động trên nền tảng Android. Thư viện này cung cấp một giao diện người dùng tương tác đơn giản và hấp dẫn để đánh giá hoặc đánh giá các mục tiêu khác nhau bằng cách sử dụng biểu tượng mặt cười.

Smiley Rating cho phép chúng ta hiển thị một tập hợp các biểu tượng mặt cười hoặc hình ảnh tương tự, biểu thị các mức độ khác nhau của sự hài lòng, đánh giá hoặc cảm xúc. Người dùng có thể chạm vào biểu tượng mặt cười để chọn mức độ tương ứng với cảm xúc của họ. Thư viện này cung cấp một giao diện tương tác mượt mà và thân thiện, tạo ra trải nghiệm người dùng tốt.

Smiley Rating cho phép tùy chỉnh các yếu tố như số lượng biểu tượng mặt cười, biểu tượng được chọn mặc định, kích thước, màu sắc và các hiệu ứng hình ảnh khác. Chúng ta có thể tùy chỉnh giao diện của Smiley Rating để phù hợp với thiết kế và phong cách của ứng dụng của chúng ta.

1. THIẾT KẾ ỨNG DỤNG
   1. Thiết kế và xây dựng giao diện của ứng dụng
      1. Nhận diện ngôn ngữ ký hiệu

A screen shot of a computer

Description automatically generated with medium confidence

Hình : Hình thiết kế chức năng nhận diện ngôn ngữ ký hiệu

A hand with a green border

Description automatically generated with low confidence

Hình : Hình chức năng nhận diện ngôn ngữ ký hiệu

* + 1. Nhận diện cảm xúc khuôn mặt

A screenshot of a face recognition system

Description automatically generated with low confidence

Hình : Hình thiết kế chức năng nhận diện cảm xúc khuôn mặt

A selfie of a child

Description automatically generated with medium confidence

Hình : : Hình thiết kế chức năng nhận diện cảm xúc khuôn mặt

* + 1. Nhận diện số viết tay

A screenshot of a computer

Description automatically generated with medium confidence

Hình : Hình thiết kế chức năng nhận diện số viết tay

A black number on a white background

Description automatically generated with low confidence

Hình : Hình chức năng nhận diện số viết tay

* 1. Xây dựng Logic cho ứng dụng
     1. Chuẩn bị các Model

Nhóm sẽ tải xuống các Model được huấn luyện sẵn của Tensorflow từ các nguồn như: TensorFlow Hub, TensorFlow Model Zoo. Sau đó sẽ chuyển đổi các Model thành dạng .tflite để tương thích với ứng dụng của mình.

Để có thể sử dụng TensorFlow Lite và triển khai model trong đề tài, nhóm sẽ tích hợp các Model vào thứ mục “ml” trong dự án Android của mình.

* + 1. Chuẩn bị các thư viện

Để có thể hoàn thành được đề tài, nhóm sẽ cần sử dụng các thư viện: Tensorflow Lite, FingerPaintView, …

A screen shot of a computer program

Description automatically generated with medium confidence

Hình : Hình khai báo các thư viện được sử dụng

* + 1. Lấy dữ liệu đầu vào

Khai báo quyền sử dụng Camera tại file AndroidManifest.xml

A screen shot of a computer code

Description automatically generated with low confidence

Hình : Hình khai báo quyền sử dụng Camera

Kiểm tra và yêu cầu quyền sử dụng Camera

A picture containing text, screenshot, font

Description automatically generated

Hình : Hình đoạn code yêu cầu quyền sử dụng Camera

* + 1. Xử lý dữ liệu

Khởi tạo Detector từ các Model

A screen shot of a computer

Description automatically generated with low confidence

Hình : Hình đoạn code khởi tạo Detector

Tiến hành xử lý ảnh nhận vào

A picture containing text, screenshot

Description automatically generated

Hình : Hình đoạn code xử lý ảnh

1. KẾT LUẬN
   1. Kết quả đạt được
      1. Kiến thức

Các thành viên trong nhóm đều đã tích lũy cho mình những kiến thức về lập trình Android và các kiến thức về học máy, đồng thời biết cách áp dụng những kiến thức ấy vào trong sản phẩm của mình.

Bên cạnh đó, các thành viên cũng đã củng cố kiến thức về các Layout trong thiết kế giao diện cho ứng dụng Android.

* + 1. Kỹ năng

Bên cạnh các kiến thức, các thành viên trong nhóm đều phát triển được các kỹ năng cần thiết như: thiết kế và xây dựng ứng dụng Android theo yêu cầu đặt ra. Nâng cao khả năng sử dụng các công cụ mới như: Android Studio,.. Gia tăng khả năng lập trình. Đông thời các kỹ năng mềm như: quản lý dụ án, xử lý tình huống và làm việc nhóm..

* + 1. Sản phẩm

Bằng những kiến thức trong môn học và một số môn học liên quan như: học máy, xử lý ảnh số,... Nhóm đã hoàn thiện việc xây dựng ứng dụng Android với các chức năng nhận diện ngôn ngữ ký hiệu, nhận diện cảm xúc khuôn mặt, nhận diện số viết tay. Đồng thời các chức năng của ứng dụng hoạt động ổn định và có độ chính xác cao.

* 1. Ưu điểm
* Ứng dụng hiệu quả marchine learning vào ứng dụng Android.
* Đề tài đã hoàn thành tốt các yêu cầu đặt ra.
* Giao diện trang ứng dụng đơn giản, dễ dàng sử dụng
* Ứng dụng hoạt động ổn định.
* Kết quả nhận diện có độ chính xác tương đối cao.
  1. Nhược điểm
* Vì Dataset còn hạn chế nên trong một số trường hợp ứng dụng sẽ cho ra kết quả sai.
* Chỉ nhận diện dược ký tự đơn lẻ.
  1. Hướng phát triển
* Gia tăng Dataset để train cho ứng dụng có thể nhận diện được chính xác hơn.
* Gia tăng khả năng nhận diện của của ứng dụng lên nhiều ký tự cùng lúc.

# **TÀI LIỆU THAM KHẢO**

<https://developer.android.com/>

<https://www.geeksforgeeks.org/>

<https://howkteam.vn/>

https://khoapham.vn/

Và một số nguồn khác trên Internet.