

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KỲ
HỌC PHẦN: HỌC MÁY (INT3405 20)**

Giảng viên: TS. Trần Quốc Long

Sinh viên: Nguyễn Văn Hoàng

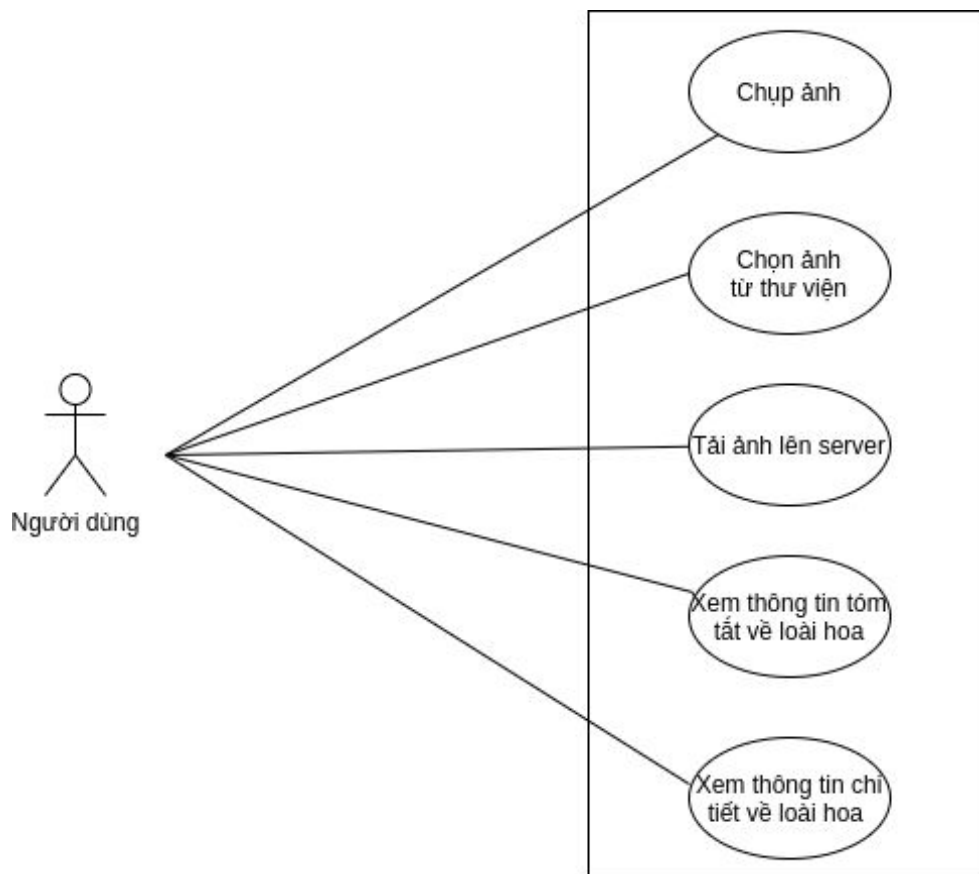
Hà Nội, 2020

1. Mục tiêu dự án

- Xây dựng ứng dụng di động giúp người dùng nhận diện loài hoa mà họ không biết bằng cách chụp ảnh loài hoa đó, hoặc chọn một bức ảnh lưu sẵn trong thư viện ảnh của máy;
- Hiển thị thông tin loài hoa mà mô hình nhận diện được cho người dùng. Nội dung hiển thị bao gồm tên loài hoa nhận diện được đi kèm với hình ảnh người dùng đã chụp/chọn, thông tin tóm tắt về loài hoa dựa trên thông tin lấy về từ 2 nguồn tham khảo, bao gồm **(1) Wikipedia** và **(2) Biology Online** (từ điển sinh học lớn nhất tính đến nay với hơn 70.000 định nghĩa).
- Cho phép người dùng đọc thông tin chi tiết hơn về loài hoa bằng việc hiển thị một tab trình duyệt trong ứng dụng sau khi người dùng nhấn nút **Show more/Xem thêm**.

2. Các use cases & tính năng đã cài đặt

2.1. Use Case Diagram



Use Case Diagram của ứng dụng Flower Detector.

2.2. Các tính năng đã cài đặt

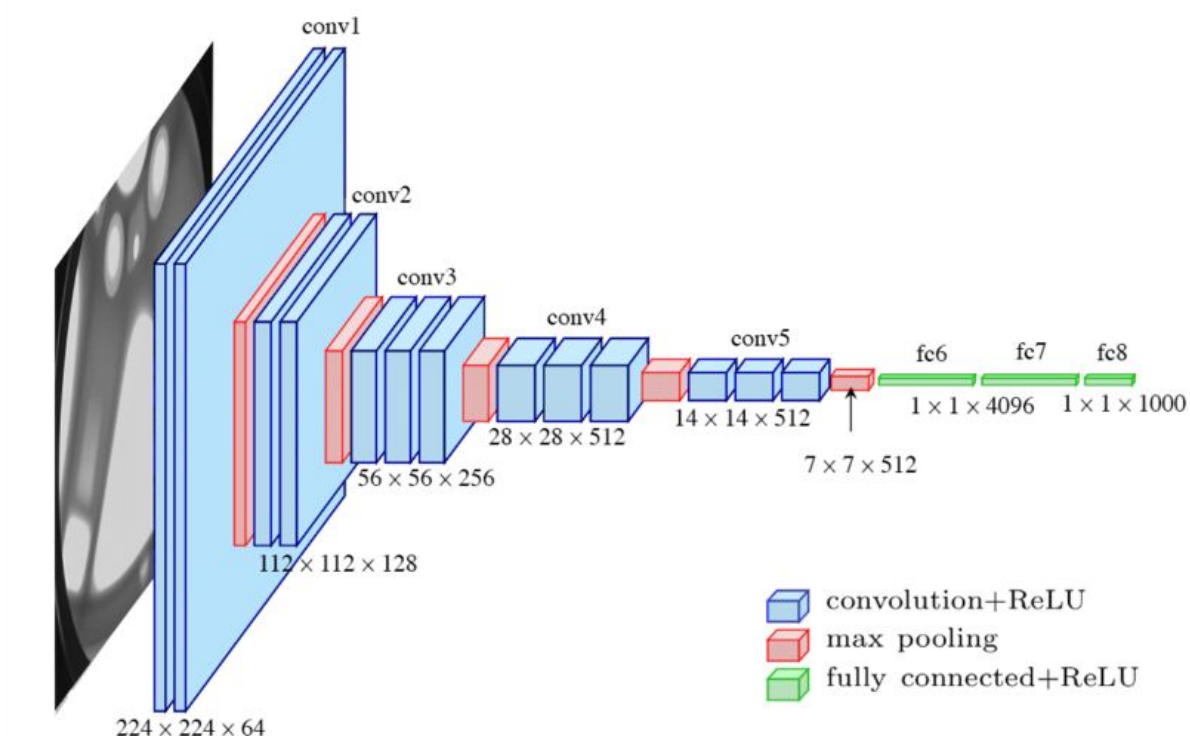
- Tính năng chụp ảnh;
- Tính năng chọn ảnh có từ trước được lưu trong thư viện ảnh;
- Tính năng upload ảnh lên server và nhận kết quả trả về;
- Hiển thị tóm tắt thông tin về loài hoa cho người dùng từ 2 nguồn Wikipedia và Biology Online. Nếu kết quả trả về là không phải là hoa thì hiển thị cho người dùng biết không có hoa trong bức ảnh.
- Cho phép người dùng xem thông tin chi tiết về loài hoa bằng việc mở tab trình duyệt với keyword là tên loài hoa đó.

3. Công nghệ, kỹ thuật đã sử dụng

3.1. Model

Kiến trúc model

Để xây dựng mô hình cho bài toán nhận diện loài hoa, em đã sử dụng mô hình VGG-16. Kiến trúc mô hình VGG-16 được thể hiện trong hình dưới đây:



Mạng VGG16 nhận đầu vào là ảnh kích thước $224 \times 224 \times 3$, ảnh đầu vào được truyền qua các lớp convolution. Các filters có kích thước là 3×3 với stride là 1, sử dụng

dùng padding để giữ nguyên chiều của đầu vào. Mạng sử dụng các lớp Max-Pooling với cửa sổ trượt là 2×2 , bước nhảy là 2. Theo sau các lớp convolution là 3 lớp fully-connected, 2 lớp đầu tiên mỗi lớp có 4096 kênh, lớp cuối cùng có số kênh bằng với số class phân loại trong bài toán. Trong bài toán phân loại ảnh hoa, em sử dụng lớp fully-connected bao gồm 10 kênh ở lớp cuối cùng vì bài toán có 10 class hoa.

3.2. Server

Với thành phần Server, em viết bằng ngôn ngữ python, với framework Flask. Flask có ưu điểm là nhẹ, có rất ít lỗi do ít bị phụ thuộc cũng như dễ dàng phát hiện và xử lý các lỗi bảo mật. Việc decode ảnh Base64 nhận từ client em dùng thư viện Base64, tính năng đọc/ghi ảnh sử dụng thư viện PIL.

3.3. Mobile application

Ứng dụng di động được phát triển cho các thiết bị chạy hệ điều hành Android sử dụng Android Studio. Ứng dụng hỗ trợ các thiết bị chạy Android 5.0 (API level 21) trở lên.

Về chi tiết cài đặt bên trong ứng dụng, có một số điểm đáng chú ý như sau:

- + Đối với tính năng chụp ảnh, em tự customize giao diện camera sử dụng thư viện CameraX trong bộ thư viện mới nhất Android Jetpack được cung cấp bởi Google nhằm nâng cao chất lượng ứng dụng.

- + Đối với tính năng upload ảnh lên server, em sử dụng thư viện Retrofit2 để đem lại tốc độ gửi/nhận nhanh nhất có thể.

- + Ảnh upload lên server được encoded theo format Base64.

- + Design pattern được sử dụng là MVP (Model-View-Presenter) nhằm tách biệt rõ các thành phần, dễ dàng cho việc bảo trì ứng dụng sau này.

4. Các khó khăn gặp phải & cách khắc phục

Khó khăn: Kích cỡ mô hình quá lớn ($> 300\text{MB}$) nên không thể cài đặt trực tiếp vào ứng dụng di động để chạy offline, vì nó sẽ khiến ứng dụng trở nên quá nặng và ảnh hưởng đến hiệu năng và trải nghiệm người dùng.

Giải pháp khắc phục: Để khắc phục khó khăn trên, em đã chuyển sang phát triển theo mô hình client - server. Cụ thể, ứng dụng di động sẽ đóng vai trò là client, cho

phép người dùng chụp/chọn ảnh và tải ảnh lên server. Thành phần server được phát triển sử dụng web framework Flask (ngôn ngữ Python), trong đó load vào mô hình đã train, khi server tiếp nhận ảnh được gửi lên bởi người dùng sẽ tiến hành gọi model để dự đoán trong ảnh có chứa loài hoa gì, sau đó trả về tên loài hoa cho client dưới định dạng JSON.

5. Đánh giá hiệu suất của chương trình (hoặc mô hình)

*** Mobile application:**

Ứng dụng di động của em được cài đặt thử nghiệm trên thiết bị di động cá nhân chạy Android 9 cho kết quả như sau:

- Thời gian trung bình upload ảnh lên và nhận kết quả trả về hiển thị cho người dùng: 3-5s

*** Mô hình:**

Quá trình huấn luyện được tiến hành trên môi trường Google Colab được cung cấp miễn phí bởi Google. Các thông số về môi trường phần cứng:

- Sử dụng Nvidia Tesla K80 GPU
- Ram 8GB

Bài toán phân loại được tiến hành xây dựng trên nền tảng:

- Tensorflow 1.13.1
- Keras 2.2.4

Mô hình được huấn luyện trên tập train bao gồm 3767 ảnh, tập val bao gồm 1098 ảnh. Các tham số của mô hình như sau: learning rate 0,0001, batch_size 32, 20 epochs. Kết quả thu được của mô hình sau khi huấn luyện đạt độ chính xác 90% trên tập val.